RM0456
リファレンスマニュアル

STM32U5シリーズArm® Cortex®-M32bitマイクロコントローラ

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32U5シリーズマイクロコントローラのメモリやペリフェラルを使用する方法について、詳しく説明しています。

注文情報、機械的および電気的なデバイスの特性については、対応するデータシートを参照してください。

Arm® Cortex®-Mxコアについては、対応するArm®テクニカルリファレンスマニュアルを参照してください（http://infocenter.arm.com）。

STM32U5シリーズマイクロコントローラでは、STの最先端の特許技術を活用しています。

関連ドキュメント

- STM32U535xxデータシート（DS14217）
- STM32U545xxデータシート（DS14216）
- STM32U535xxおよびSTM32U545xxデバイスの正誤表シート（ES0587）
- STM32U575xxデータシート（DS13737）
- STM32U585xxデータシート（DS13086）
- STM32U575xxおよびSTM32U585xxデバイスの正誤表シート（ES0499）
- STM32U59xxxデータシート（DS13633）
- STM32U5Axxxデータシート（DS13543）
- STM32U59xxxおよびSTM32U5Axxxデバイスの正誤表シート（ES0553）
- STM32U5Fxxxデータシート（DS14395）
- STM32U5Gxxxデータシート（DS14102）
- STM32U5FxxxおよびSTM32U5Gxxxデバイスの正誤表シート（ES0595）
- STM32 Cortex®-M33マイクロコントローラプログラミングマニュアル（PM0264）

目次

1 このマニュアルにおける表記の規則 123

1.1 一般情報 123

1.2 レジスタに関する略記 123

1.3 用語 124

1.4 使用可能なペリフェラル 124

2 メモリとバスのアーキテクチャ 125

2.1 システムアーキテクチャ 125

2.1.1 高速 C バス 127

2.1.2 低速 C バス 127

2.1.3 S- バス 128

2.1.4 DCACHE S バス 128

2.1.5 GPDMA バス 128

2.1.6 OTG_HS バス 128

2.1.7 LTDC バス 128

2.1.8 GPU2D バス 128

2.1.9 GFXMMU バス 129

2.1.10 SDMMC1 および SDMMC2 コントローラ DMA バス 129

2.1.11 バスマトリックス 129

2.1.12 AHB/APB ブリッジ 129

2.1.13 SmartRun ドメイン（SRD） 129

2.2 Arm TrustZone セキュリティアーキテクチャ 130

2.2.1 デフォルトの Arm TrustZone セキュリティ状態 131

2.2.2 Arm TrustZone ペリフェラルの分類 132

2.3 メモリ構成 136

2.3.1 概要 136

2.3.2 メモリマップとレジスタ境界アドレス 137

2.3.3 内蔵 SRAM 146

2.3.4 Flash メモリの概要 146

3 システムセキュリティ 147

3.1 主なセキュリティ機能 147

3.2 セキュアインストール 148

3.3 セキュア･ブート 148
3.3.1 一意なブートエントリおよび BOOT_LOCK 149
3.3.2 システム Flash メモリの不変の信頼の起点 149
3.4 セキュア更新 149
3.5 TrustZone を使用したリソースの分離 150
3.5.1 TrustZone セキュリティアーキテクチャ 150
3.5.2 Cortex-M33 の Armv8-M セキュリティ拡張 151
3.5.3 IDAU/SAU を使用したメモリおよびペリフェラルの割り当て 151
3.5.4 GTZC を使用したメモリおよびペリフェラルの割り当て 153
3.5.5 TrustZone 対応ペリフェラルのセキュリティの管理 157
3.5.6 TrustZone セキュリティの有効化 164
3.5.7 TrustZone セキュリティの無効化 165
3.6 その他のリソースの分離 165
3.6.1 セキュア非表示保護（HDP）を使用した時間的分離 165
3.6.2 Cortex 特権モードを使用したリソース分離 166
3.7 セキュアな実行 170
3.7.1 メモリ保護ユニット（MPU） 170
3.7.2 内蔵 Flash メモリ書込み保護 171
3.7.3 タンパの検出と対応 171
3.8 セキュアストレージ 174
3.8.1 ハードウェア秘密キー管理 175
3.8.2 ユニーク ID 175
3.9 暗号化エンジン 175
3.9.1 暗号化エンジンの特徴 176
3.9.2 セキュア AES コプロセッサ（SAES） 177
3.9.3 オンザフライ復号エンジン（OTFDEC） 177
3.10 製品ライフサイクル 178
3.10.1 読出し保護（RDP）によるライフサイクル管理 179
3.10.2 推奨されるオプションバイトの設定 182
3.11 アクセスコントロールされたデバッグ 182
3.11.1 読出し保護（RDP）対応のデバッグ保護 182
3.12 ソフトウェア知的財産保護と共同開発 183
3.12.1 RDP によるソフトウェア知的財産保護 184
3.12.2 OTFDEC によるソフトウェア知的財産保護 184
3.12.3 その他のソフトウェア知的財産保護 186

4 ブートモード 187

5 グローバル TrustZone コントローラ（GTZC） 190

5.1 GTZC の概要 190

5.2 GTZC の主な機能 190

5.3 GTZC の実装 192

5.4 GTZC の機能詳細 195

5.4.1 GTZC ブロック図 195

5.4.2 不正アクセスの定義 196

5.4.3 TrustZone セキュリティ･コントローラ（TZSC） 196

5.4.4 ブロックベースのメモリ保護コントローラ（MPCBB） 198

5.4.5 TrustZone 不正アクセスコントローラ（TZIC） 198

5.4.6 パワーオン / リセット状態 198

5.5 GTZC 割込み 199

5.6 GTZC1 TZSC レジスタ 200

5.6.1 GTZC1 TZSC 制御レジスタ（GTZC1_TZSC_CR） 200

5.6.2 GTZC1 TZSC セキュア設定レジスタ 1（GTZC1_TZSC_SECCFGR1） 200

5.6.3 GTZC1 nnTZSC セキュア設定レジスタ 2（GTZC1_TZSC_SECCFGR2） . . . 203

5.6.4 GTZC1 TZSC セキュア設定レジスタ 3（GTZC1_TZSC_SECCFGR3） 204

5.6.5 GTZC1 TZSC 特権設定レジス 1（GTZC1_TZSC_PRIVCFGR1） 208

5.6.6 GTZC1 TZSC 特権設定レジスタ 2（GTZC1_TZSC_PRIVCFGR2） 210

5.6.7 GTZC1 TZSC 特権設定レジスタ 3（GTZC1_TZSC_PRIVCFGR3） 212

5.6.8 GTZC1 TZSC メモリ x サブ領域 z ウォーターマーク設定レジスタ（GTZC1_TZSC_MPCWMxzCFGR）（z = A から B） 215

5.6.9 GTZC1 TZSC メモリ x サブ領域 A ウォーターマークレジスタ（GTZC1_TZSC_MPCWMxAR） 216

5.6.10 GTZC1 TZSC メモリ x サブ領域 B ウォーターマークレジスタ（GTZC1_TZSC_MPCWMxBR） 217

5.6.11 GTZC1 TZSC レジスタのマップ 218

5.7 GTZC1 TZIC レジスタ 220

5.7.1 GTZC1 TZIC 割込み有効化レジスタ 1（GTZC1_TZIC_IER1） 220

5.7.2 GTZC1 TZIC 割込み有効化レジスタ 2（GTZC1_TZIC_IER2） 222

5.7.3 GTZC1 TZIC 割込み有効レジスタ 3（GTZC1_TZIC_IER3） 224

5.7.4 GTZC1 TZIC 割込み有効レジスタ 4（GTZC1_TZIC_IER4） 227

5.7.5 GTZC1 TZIC ステータス･レジスタ 1（GTZC1_TZIC_SR1） 229

5.7.6 GTZC1 TZIC ステータスレジスタ 2（GTZC1_TZIC_SR2） 232

5.7.7 GTZC1 TZIC ステータス･レジスタ 3（GTZC1_TZIC_SR3） 233

5.7.8 GTZC1 TZIC ステータスレジスタ 4（GTZC1_TZIC_SR4） 236

5.7.9 GTZC1 TZIC フラグ･クリア･レジスタ 1（GTZC1_TZIC_FCR1）. 239
5.7.10 GTZC1 TZIC フラグクリアレジスタ 2（GTZC1_TZIC_FCR2）. 241
5.7.11 GTZC1 TZIC フラグ･クリア･レジスタ 3（GTZC1_TZIC_FCR3）. 242
5.7.12 GTZC1 TZIC フラグクリアレジスタ 4（GTZC1_TZIC_FCR4）. 246
5.7.13 GTZC1 TZIC レジスタ･マップ 248
5.8 GTZC1 MPCBBz レジスタ（z = 1、2、3、5、6）. 250
5.8.1 GTZC1 SRAMz MPCBB 制御レジスタ（GTZC1_MPCBBz_CR）（z = 1、2、3、5、6）. 250
5.8.2 GTZC1 SRAMz MPCBB 設定ロックレジスタ 1（GTZC1_MPCBBz_CFGLOCKR1）（z = 1、2、3、5、6）. 251
5.8.3 GTZC1 SRAMz MPCBB 設定ロックレジスタ 2（GTZC1_MPCBBz_CFGLOCKR2）（z = 1、2、3、5、6）. 251
5.8.4 スーパーブロック x レジスタの GTZC1 SRAMz MPCBB セキュリティ設定（GTZC1_MPCBBz_SECCFGRx）（z = 1、2、3、5、6）. 252
5.8.5 スーパーブロック x レジスタの GTZC1 SRAMz MPCBB 特権設定（GTZC1_MPCBBz_PRIVCFGRx）（z = 1、2、3、5、6）. 252
5.8.6 GTZC1 MPCBBz レジスタマップ（z = 1、2、3、5、6）. 253
5.9 GTZC2 TZSC レジスタ 254
5.9.1 GTZC2 TZSC 制御レジスタ（GTZC2_TZSC_CR）. 254
5.9.2 GTZC2 TZSC セキュア設定レジスタ 1（GTZC2_TZSC_SECCFGR1）. 254
5.9.3 GTZC2 TZSC 特権設定レジスタ 1（GTZC2_TZSC_PRIVCFGR1）. 256
5.9.4 GTZC2 TZSC レジスタマップ 257
5.10 GTZC2 TZIC レジスタ 258
5.10.1 GTZC2 TZIC 割込み有効化レジスタ 1（GTZC2_TZIC_IER1）. 258
5.10.2 GTZC2 TZIC 割込み有効レジスタ 2（GTZC2_TZIC_IER2）. 259
5.10.3 GTZC2 TZIC ステータス･レジスタ 1（GTZC2_TZIC_SR1）. 260
5.10.4 GTZC2 TZIC ステータスレジスタ 2（GTZC2_TZIC_SR2）. 262
5.10.5 GTZC2 TZIC フラグ･クリア･レジスタ 1（GTZC2_TZIC_FCR1）. 263
5.10.6 GTZC2 TZIC フラグクリアレジスタ 2（GTZC2_TZIC_FCR2）. 264
5.10.7 GTZC2 TZIC レジスタマップ 266
5.11 GTZC2 MPCBB4 レジスタ 267
5.11.1 GTZC2 SRAM4 MPCBB 制御レジスタ（GTZC2_MPCBB4_CR）. 267
5.11.2 GTZC2 SRAM4 MPCBB 設定ロックレジスタ 1（GTZC2_MPCBB4_CFGLOCKR1）. 268
5.11.3 スーパーブロック 0 の GTZC2 SRAM4 MPCBB セキュリティ設定レジスタ（GTZC2_MPCBB4_SECCFGR0）. 268
5.11.4 スーパーブロック 0 の GTZC2 SRAM4 MPCBB 特権設定レジスタ（GTZC2_MPCBB4_PRIVCFGR0）. 269
5.11.5 GTZC2 MPCBB4 レジスタマップ 269

6 RAM 設定コントローラ（RAMCFG）. 270

6.1 RAMCFG の概要 270

6.2 RAMCFG の主な機能 270

6.3 RAMCFG の機能詳細 270

6.3.1 内部 SRAM 機能 270

6.3.2 エラーコード訂正（SRAM2、SRAM3、BKPSRAM）. 271

6.3.3 書込み保護（SRAM2）. 274

6.3.4 読出しアクセスの遅延 274

6.3.5 ソフトウェア消去 274

6.4 RAMCFG 低電力モード 275

6.5 RAMCFG 割込み 275

6.6 RAMCFG レジスタ 276

6.6.1 RAMCFG メモリ x 制御レジスタ (RAMCFG_MxCR) 276

6.6.2 RAMCFG メモリ x 割込みイネーブルレジスタ（RAMCFG_MxIER）. 277

6.6.3 RAMCFG メモリ割込みステータスレジスタ (RAMCFG_MxISR) 278

6.6.4 RAMCFG メモリ xECC シングルエラーアドレスレジスタ（RAMCFG_MxSEAR）. 278

6.6.5 RAMCFG メモリ xECC ダブルエラーアドレスレジスタ（RAMCFG_MxDEAR）. 279

6.6.6 RAMCFG メモリ x 割込みクリアレジスタ x(RAMCFG_MxICR) 279

6.6.7 RAMCFG メモリ 2 書込み保護レジスタ 1（RAMCFG_M2WPR1）. 280

6.6.8 RAMCFG メモリ 2 書込み保護レジスタ 2（RAMCFG_M2WPR2）. 280

6.6.9 RAMCFG メモリ xECC キーレジスタ（RAMCFG_MxECCKEYR）. 281

6.6.10 RAMCFG メモリ x 消去キーレジスタ（RAMCFG_MxERKEYR）. 281

6.6.11 RAMCFG レジスタマップ 282

7 内蔵 Flash メモリ（FLASH）. 285

7.1 FLASH の概要 285

7.2 Flash の主な機能 285

7.3 Flash の機能説明 286

7.3.1 Flash メモリの構成 286

7.3.2 エラーコード訂正（ECC）. 288

7.3.3 読出しアクセスの遅延 289

7.3.4 バンク·パワーダウン·モード 290

7.3.5 Flash メモリプログラムおよび消去操作 291

7.3.6 Flash メインメモリの消去シーケンス 292

7.3.7 Flash メインメモリのプログラミングシーケンス 293

7.3.8 Flash メモリの書換え耐性 295
7.3.9 Flash メモリ･エラー･フラグ 295
7.3.10 書込み中読出し（RWW） 297
7.3.11 FLASH のプログラム操作または消去操作中のパワーダウン 298
7.3.12 FLASH のプログラム操作または消去操作中のリセット 298
7.4 Flash オプションバイト 299
7.4.1 オプションバイトの説明 299
7.4.2 オプションバイトのプログラミング 300
7.5 FLASH TrustZone のセキュリティと特権の保護 301
7.5.1 TrustZone のセキュリティ保護 301
7.5.2 ウォーターマークベースのセキュア Flash メモリの領域保護 302
7.5.3 セキュア非表示保護（HDP） 303
7.5.4 ブロックベースのセキュア Flash メモリの領域保護 304
7.5.5 Flash のセキュリティ属性状態 305
7.5.6 ブロックベースの特権 Flash メモリの領域保護 305
7.5.7 Flash メモリ･レジスタの特権および非特権モード 306
7.5.8 バンクスワップの際の Flash メモリバンク属性 306
7.6 Flash メモリの保護 308
7.6.1 書込み保護（WRP） 308
7.6.2 読出し保護（RDP） 309
7.7 Flash メモリと FLASH レジスタのアクセス制御 318
7.8 FLASH 割込み 322
7.9 Flash レジスタ 322
7.9.1 Flash アクセス制御レジスタ（Flash_ACR） 322
7.9.2 FLASH 非セキュアキーレジスタ（FLASH_NSKEYR） 324
7.9.3 FLASH セキュアキーレジスタ（FLASH_SECKEYR） 324
7.9.4 Flash オプションキーレジスタ（Flash_OPTKEYR） 325
7.9.5 FLASH バンク 1 パワーダウンキーレジスタ（FLASH_PDKEY1R） 325
7.9.6 FLASH バンク 2 パワーダウンキーレジスタ（FLASH_PDKEY2R） 326
7.9.7 FLASH 非セキュア･ステータス･レジスタ（FLASH_NSSR） 326
7.9.8 FLASH セキュア･ステータス･レジスタ（Flash_SECSR） 329
7.9.9 FLASH 非セキュア制御レジスタ（FLASH_NSCR） 330
7.9.10 Flash セキュア制御レジスタ（Flash_SECCR） 332
7.9.11 FlashECC レジスタ（Flash_ECCR） 334
7.9.12 FLASH 操作ステータスレジスタ（FLASH_OPSR） 335
7.9.13 Flash オプションレジスタ（Flash_OPTR） 336
7.9.14 FLASH 非セキュアブートアドレス 0 レジスタ（FLASH_NSBOOTADD0R） . . . 338
7.9.15 FLASH 非セキュアブートアドレス 1 レジスタ（FLASH_NSBOOTADD1R） . . . 339

7.9.16 FLASH セキュアブートアドレス 0 レジスタ（FLASH_SECBOOTADD0R）....340
7.9.17 FLASH のセキュアウォーターマーク 1 レジスタ 1（FLASH_SECWM1R1）...341
7.9.18 FLASH のセキュアウォーターマーク 1 レジスタ 2（FLASH_SECWM1R2）...342
7.9.19 FLASH の WRP1 領域 A のアドレスレジスタ（FLASH_WRP1AR）.....343
7.9.20 FLASH の WRP1 領域 B のアドレスレジスタ（FLASH_WRP1BR）.....344
7.9.21 FLASH のセキュアウォーターマーク 2 レジスタ 1（FLASH_SECWM2R1）...345
7.9.22 FLASH のセキュアウォーターマーク 2 レジスタ 2（FLASH_SECWM2R2）...346
7.9.23 FLASH の WRP2 領域 A のアドレスレジスタ（FLASH_WRP2AR）.....347
7.9.24 FLASH の WPR2 領域 B のアドレスレジスタ（FLASH_WRP2BR）.....348
7.9.25 FLASHOEM1 キーレジスタ 1（FLASH_OEM1KEYR1）.....349
7.9.26 FLASHOEM1 キーレジスタ 2（FLASH_OEM1KEYR2）.....349
7.9.27 FLASHOEM2 キーレジスタ 1（FLASH_OEM2KEYR1）.....350
7.9.28 FLASHOEM2 キーレジスタ 2（FLASH_OEM2KEYR2）.....350
7.9.29 ブロックベースの FLASH セキュアバンク 1 レジスタ x（FLASH_SECBB1Rx）.....351
7.9.30 ブロックベースの FLASH セキュアバンク 2 レジスタ x（FLASH_SECBB2Rx）.....351
7.9.31 FLASH セキュア HDP 制御レジスタ（FLASH_SECHDPCR）.....352
7.9.32 FLASH の特権設定レジスタ（FLASH_PRIVCFGR）.....352
7.9.33 ブロックベースの FLASH 特権バンク 1 レジスタ x（FLASH_PRIVBB1Rx）.....353
7.9.34 ブロックベースの FLASH 特権バンク 2 レジスタ x（FLASH_PRIVBB2Rx）...354
7.9.35 Flash レジスタマップ 354

8 命令キャッシュ (ICACHE) 358

8.1 ICACHE の概要 358
8.2 ICACHE の主な機能 358
8.3 ICACHE の実装 359
8.4 ICACHE の機能説明 360
8.4.1 ICACHE ブロック図 360
8.4.2 ICACHE のリセットおよびクロック 361
8.4.3 ICACHE TAG メモリ 361
8.4.4 ダイレクトマップ ICACHE(1 ウェイキャッシュ) 363
8.4.5 ICACHE 有効化 363
8.4.6 キャッシュ格納可および不可のトラフィック 364
8.4.7 アドレス再配置 365
8.4.8 キャッシュ可能アクセス 367
8.4.9 デュアルマスタキャッシュ 367
8.4.10 ICACHE のセキュリティ 368

8.4.11 ICACHE のメンテナンス 368
8.4.12 ICACHE 性能の監視 368
8.4.13 ICACHE ブート 369
8.5 ICACHE 低電力モード 369
8.6 ICACHE のエラー管理と割込み 369
8.7 ICACHE レジスタ 370
8.7.1 ICACHE 制御レジスタ（ICACHE_CR） 370
8.7.2 ICACHE ステータスレジスタ（ICACHE_SR） 371
8.7.3 ICACHE 割込み有効化レジスタ（ICACHE_IER） 371
8.7.4 ICACHE フラグクリアレジスタ（ICACHE_FCR） 372
8.7.5 ICACHE ヒットモニタレジスタ（ICACHE_HMONR） 372
8.7.6 ICACHE ミスモニタレジスタ（ICACHE_MMONR） 373
8.7.7 ICACHE 領域 x 設定レジスタ（ICACHE_CRRx） 373
8.7.8 ICACHE レジスタマップ 374

9 データキャッシュ (DCACHE) 376
9.1 DCACHE の概要 376
9.2 DCACHE の主な機能 376
9.3 DCACHE の実装 377
9.4 DCACHE の機能説明 378
9.4.1 DCACHE のブロック図 378
9.4.2 DCACHE のリセットおよびクロック 379
9.4.3 DCACHE TAG メモリ 379
9.4.4 DCACHE 有効化 381
9.4.5 キャシュ格納可および不可のトラフィック 381
9.4.6 キャッシュ可能アクセス 382
9.4.7 DCACHE のセキュリティ 384
9.4.8 DCACHE のメンテナンス 384
9.4.9 DCACHE 性能の監視 386
9.4.10 DCACHE ブート 386
9.5 DCACHE 低電力モード 386
9.6 DCACHE のエラー管理と割込み 386
9.7 DCACHE レジスタ 387
9.7.1 DCACHE 制御レジスタ (DCACHE_CR) 387
9.7.2 DCACHE ステータスレジスタ (DCACHE_SR) 389
9.7.3 DCACHE 割込みイネーブルレジスタ (DCACHE_IER) 390
9.7.4 DCACHE フラグクリアレジスタ (DCACHE_FCR) 390

9.7.5 DCACHE 読出しヒットモニタレジスタ (DCACHE_RHMONR) 391
9.7.6 DCACHE 読出しミスモニタレジスタ (DCACHE_RMMONR) 391
9.7.7 DCACHE 書込みヒットモニタレジスタ (DCACHE_WHMONR) 392
9.7.8 DCACHE 書込みミスモニタレジスタ (DCACHE_WMMONR) 392
9.7.9 DCACHE コマンド範囲開始アドレスレジスタ（DCACHE_CMDRSADDRR） 392
9.7.10 DCACHE コマンド範囲終了アドレスレジスタ（DCACHE_CMDREADDRR） 393
9.7.11 DCACHE レジスタマップ 393

10 電源制御（PWR） 395

10.1 PWR の概要 395
10.2 PWR の主な機能 395
10.3 PWR ピンおよび内部信号 396
10.4 PWR 電源供給と電源ドメイン 398
10.4.1 外部電源 399
10.4.2 内部レギュレータ 400
10.4.3 パワーアップおよびパワーダウン電源シーケンス 400
10.4.4 独立したアナログペリフェラルの電源 400
10.4.5 独立 I/O 電源レール 401
10.4.6 独立した USB トランシーバの電源 401
10.4.7 バッテリバックアップドメイン 401
10.5 PWR システム電源電圧レギュレーション 403
10.5.1 SMPS および LDO 内蔵レギュレータ 403
10.5.2 LDO および SMPS に対するリセット、電圧スケーリング、および低電力モード 403
10.5.3 LDO および SMPS ステップダウンコンバータの高速起動 403
10.5.4 ダイナミック電圧スケーリングの管理 404
10.6 PWR 電源供給の監視 405
10.6.1 ブラウンアウトリセット（BOR） 405
10.6.2 プログラム可能な電圧検出器（PVD） 406
10.6.3 ペリフェラル電圧モニタ（PVM） 407
10.6.4 バックアップドメイン電圧および温度監視 408
10.7 PWR 電源管理 409
10.7.1 電源モード 409
10.7.2 自律型ペリフェラルと低消費電力バックグラウンド自律モード（LPBAM） 416
10.7.3 RUN モード 418
10.7.4 低電力モード 418

10.7.5 SLEEP モード 419
10.7.6 STOP0 モード 420
10.7.7 STOP1 モード 422
10.7.8 STOP2 モード 423
10.7.9 STOP3 モード 425
10.7.10 STANDBY モード 427
10.7.11 SHUTDOWN モード 429
10.7.12 低電力モードでの USB パワー・マネージメント（STM32U59x/5Ax/5Fx/5Gx のみ） 431
10.7.13 電力モード出力ピン 432
10.8 PWR セキュリティおよび特権保護 433
10.8.1 PWR セキュリティ保護 433
10.8.2 WR 特権保護 434
10.9 PWR 割込み 435
10.10 PWR レジスタ 436
10.10.1 PWR 制御レジスタ 1（PWR_CR1） 436
10.10.2 PWR 制御レジスタ 2（PWR_CR2） 438
10.10.3 PWR 制御レジスタ 3（PWR_CR3） 441
10.10.4 PWR 電圧スケーリングレジスタ（PWR_VOSR） 442
10.10.5 PWR 電源電圧モニタの制御レジスタ（PWR_SVMCR） 444
10.10.6 PWR ウェイクアップ制御レジスタ 1（PWR_WUCR1） 445
10.10.7 PWR ウェイクアップ制御レジスタ 2（PWR_WUCR2） 446
10.10.8 PWR ウェイクアップ制御レジスタ 3（PWR_WUCR3） 447
10.10.9 PWR バックアップドメイン制御レジスタ 1（PWR_BDCR1） 449
10.10.10 PWR バックアップドメイン制御レジスタ 2（PWR_BDCR2） 450
10.10.11 PWR ディセーブルバックアップドメインレジスタ（PWR_DBPR） 450
10.10.12 PWR UCPD レジスタ（PWR_UCPDR） 451
10.10.13 PWR セキュリティ設定レジスタ（PWR_SECCFGR） 452
10.10.14 PWR 特権制御レジスタ（PWR_PRIVCFGR） 453
10.10.15 PWR ステータスレジスタ（PWR_SR） 454
10.10.16 PWR 電源電圧モニタのステータスレジスタ（PWR_SVMSR） 455
10.10.17 PWR バックアップドメインステータスレジスタ（PWR_BDSR） 456
10.10.18 PWR ウェイクアップステータスレジスタ（PWR_WUSR） 456
10.10.19 PWR ウェイクアップステータスクリアレジスタ (PWR_WUSCR) 458
10.10.20 PWR で適用されるプル設定レジスタ（PWR_APCR） 459
10.10.21 PWR ポート A プルアップ制御レジスタ（PWR_PUCRA） 459
10.10.22 PWR ポート A プルダウン制御レジスタ（PWR_PDCRA） 460
10.10.23 PWR ポート B プルアップ制御レジスタ（PWR_PUCRB） 460

10.10.24 PWR ポート B プルダウン制御レジスタ（PWR_PDCRB）. 461
10.10.25 PWR ポート C プルアップ制御レジスタ（PWR_PUCRC）. 461
10.10.26 PWR ポート C プルダウン制御レジスタ（PWR_PDCRC）. 462
10.10.27 PWR ポート D プルアップ制御レジスタ（PWR_PUCRD）. 462
10.10.28 PWR ポート D プルダウン制御レジスタ（PWR_PDCRD）. 463
10.10.29 PWR ポート E プルアップ制御レジスタ（PWR_PUCRE）. 463
10.10.30 PWR ポート E プルダウン制御レジスタ（PWR_PDCRE）. 464
10.10.31 PWR ポート F プルアップ制御レジスタ（PWR_PUCRF）. 464
10.10.32 PWR ポート F プルダウン制御レジスタ（PWR_PDCRF）. 465
10.10.33 PWR ポート G プルアップ制御レジスタ（PWR_PUCRG）. 465
10.10.34 PWR ポート G プルダウン制御レジスタ（PWR_PDCRG）. 466
10.10.35 PWR ポート H プルアップ制御レジスタ（PWR_PUCRH）. 466
10.10.36 PWR ポート H プルダウン制御レジスタ（PWR_PDCRH）. 467
10.10.37 PWR ポート I プルアップ制御レジスタ（PWR_PUCRI）. 467
10.10.38 PWR ポート I プルダウン制御レジスタ（PWR_PDCRI）. 468
10.10.39 PWR ポート J プルアップ制御レジスタ（PWR_PUCRJ）. 468
10.10.40 PWR ポート J プルダウン制御レジスタ（PWR_PDCRJ）. 469
10.10.41 PWR 制御レジスタ 4（PWR_CR4）. 470
10.10.42 PWR 制御レジスタ 5（PWR_CR5）. 471
10.10.43 PWR レジスタマップ 471

11 リセットおよびクロック制御（RCC）. 475

11.1 RCC の概要 475
11.2 RCC ピンおよび内部信号 475
11.3 RCC リセット機能の説明 475
11.3.1 電源リセット 475
11.3.2 システムリセット 476
11.3.3 バックアップドメインリセット 477
11.4 RCC クロックの機能説明 478
11.4.1 HSE クロック 480
11.4.2 HSI16 クロック 481
11.4.3 MSI（MSIS および MSIK）クロック 481
11.4.4 HSI48 クロック 484
11.4.5 SHSI クロック 484
11.4.6 PLL 484
11.4.7 LSE クロック 488
11.4.8 LSI クロック 488
11.4.9 システムクロック（SYSCLK）の選択 489

11.4.10 クロックソースの周波数と電圧スケーリング 489
11.4.11 クロックセキュリティシステム（CSS） 490
11.4.12 LSE のクロックセキュリティシステム 490
11.4.13 ADC および DAC クロック 491
11.4.14 RTC および TAMP クロック 491
11.4.15 タイマクロック 491
11.4.16 ウォッチドッグクロック 491
11.4.17 OCTOSPI クロック 492
11.4.18 HSPI1 クロック 492
11.4.19 OTG_HS クロック 492
11.4.20 DSI クロック 492
11.4.21 LTDC クロック 492
11.4.22 クロック信号出力 492
11.4.23 TIM15/TIM16/TIM17 を使用した内部／外部クロックの測定 493
11.4.24 ペリフェラルクロックゲーティングおよび自律モード 494
11.5 RCC セキュリティおよび特権機能の説明 496
11.5.1 RCC TrustZone セキュリティ保護モード 496
11.5.2 RCC 特権保護モード 499
11.6 RCC 低電力モード 500
11.7 RCC 割込み 501
11.8 RCC レジスタ 503
11.8.1 RCC クロック制御レジスタ（RCC_CR） 503
11.8.2 RCC 内部クロックソース較正レジスタ 1（RCC_ICSCR1） 507
11.8.3 RCC 内部クロックソース較正レジスタ 2（RCC_ICSCR2） 509
11.8.4 RCC 内部クロックソース較正レジスタ 3（RCC_ICSCR3） 510
11.8.5 RCC クロックリカバリ RC レジスタ（RCC_CRRCR） 510
11.8.6 RCC クロック設定レジスタ 1(RCC_CFGR1) 511
11.8.7 RCC クロック設定レジスタ 2(RCC_CFGR2) 513
11.8.8 RCC クロック設定レジスタ 3(RCC_CFGR3) 515
11.8.9 RCCPLL1 設定レジスタ（RCC_PLL1CFGR） 516
11.8.10 RCCPLL2 設定レジスタ（RCC_PLL2CFGR） 518
11.8.11 RCC PLL3 設定レジスタ（RCC_PLL3CFGR） 519
11.8.12 RCCPLL1 分周器レジスタ（RCC_PLL1DIVR） 521
11.8.13 RCC PLL1 小数分周器レジスタ（RCC_PLL1FRACR） 522
11.8.14 RCCPLL2 分周器設定レジスタ（RCC_PLL2DIVR） 523
11.8.15 RCCPLL2 小数分周器レジスタ（RCC_PLL2FRACR） 524
11.8.16 RCCPLL3 分周器設定レジスタ（RCC_PLL3DIVR） 525
11.8.17 RCC PLL3 小数分周器レジスタ（RCC_PLL3FRACR） 526

11.8.18 RCC クロック割込み有効化レジスタ（RCC_CIER）.....527
11.8.19 RCC クロック割込みフラグレジスタ（RCC_CIFR）.....528
11.8.20 RCC クロック割込みクリアレジスタ（RCC_CICR）.....530
11.8.21 RCCAHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）.....531
11.8.22 RCC AHB2 ペリフェラルリセットレジスタ 1（RCC_AHB2RSTR1）.....533
11.8.23 RCC AHB2 ペリフェラルリセットレジスタ 2（RCC_AHB2RSTR2）.....536
11.8.24 RCC AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）.....537
11.8.25 RCCAPB1 ペリフェラルリセットレジスタ 1（RCC_APB1RSTR1）.....538
11.8.26 RCC APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）.....540
11.8.27 RCCAPB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）.....541
11.8.28 RCC APB3 ペリフェラルリセットレジスタ（RCC_APB3RSTR）.....543
11.8.29 RCC AHB1 ペリフェラルクロック有効化レジスタ（RCC_AHB1ENR）.....544
11.8.30 RCCAHB2 ペリフェラルクロック有効化レジスタ 1（RCC_AHB2ENR1）...547
11.8.31 RCCAHB2 ペリフェラルクロック有効化レジスタ 2（RCC_AHB2ENR2）...551
11.8.32 RCCAHB3 ペリフェラルクロック有効化レジスタ（RCC_AHB3ENR）.....552
11.8.33 RCC APB1 ペリフェラルクロック有効化レジスタ 1（RCC_APB1ENR1）...554
11.8.34 RCCAPB1 ペリフェラルクロック有効化レジスタ 2（RCC_APB1ENR2）....556
11.8.35 RCCAPB2 ペリフェラルクロック有効化レジスタ（RCC_APB2ENR）.....557
11.8.36 RCC APB3 ペリフェラルクロック有効化レジスタ（RCC_APB3ENR）.....559
11.8.37 SLEEP モードおよび STOP モードにおける RCC AHB1 ペリフェラル クロックの有効化レジスタ（RCC_AHB1SMENR）.....561
11.8.38 SLEEP モードおよび STOP モードにおける RCC AHB2 ペリフェラル クロック有効レジスタ 1（RCC_AHB2SMENR1）.....564
11.8.39 SLEEP モードおよび STOP モードにおける RCC AHB2 ペリフェラル クロック有効レジスタ 2（RCC_AHB2SMENR2）.....568
11.8.40 SLEEP モードおよび STOP モードにおける RCC AHB3 ペリフェラル クロックの有効化レジスタ（RCC_AHB3SMENR）.....570
11.8.41 SLEEP モードおよび STOP モードにおける RCCAPB1 ペリフェラル クロック有効レジスタ 1（RCC_APB1SMENR1）.....571
11.8.42 SLEEP モードおよび STOP モードにおける RCCAPB1 ペリフェラル クロック有効レジスタ 2（RCC_APB1SMENR2）.....574
11.8.43 SLEEP モードおよび STOP モードにおける RCC APB2 ペリフェラル クロック有効レジスタ（RCC_APB2SMENR）.....576
11.8.44 SLEEP モードおよび STOP モードにおける RCC APB3 ペリフェラル クロックの有効化レジスタ（RCC_APB3SMENR）.....578
11.8.45 RCC SmartRun ドメインペリフェラル自律モードレジスタ（RCC_SRDAMR）.....580
11.8.46 RCC ペリフェラル独立クロック設定レジスタ 1（RCC_CCIPR1）.....583
11.8.47 RCC ペリフェラル独立クロック設定レジスタ 2（RCC_CCIPR2）.....586
11.8.48 RCC ペリフェラル独立クロック設定レジスタ 3（RCC_CCIPR3）.....589
11.8.49 RCC バックアップドメイン制御レジスタ（RCC_BDCR）.....591

11.8.50 RCC 制御／ステータスレジスタ（RCC_CSR）.....594
11.8.51 RCC セキュア設定レジスタ（RCC_SECCFGR）.....596
11.8.52 RCC 特権設定レジスタ (RCC_PRIVCFGR).....598
11.8.53 RCC レジスタ･マップ.....599

12 クロックリカバリシステム（CRS）.....605
12.1 概要.....605
12.2 CRS の主な機能.....605
12.3 CRS の実装.....606
12.4 CRS の機能詳細.....607
12.4.1 CRS ブロック図.....607
12.4.2 同期入力.....607
12.4.3 周波数誤差測定.....608
12.4.4 周波数誤差評価および自動トリミング.....609
12.4.5 CRS の初期化と設定.....609
12.5 CRS 低電力モード.....610
12.6 CRS 割込み.....610
12.7 CRS レジスタ.....611
12.7.1 CRS 制御レジスタ（CRS_CR）.....611
12.7.2 CRS 設定レジスタ（CRS_CFGR）.....612
12.7.3 CRS 割込みおよびステータスレジスタ（CRS_ISR）.....613
12.7.4 CRS 割込みフラグクリアレジスタ（CRS_ICR）.....615
12.7.5 CRS レジスタマップ.....616

13 汎用 I/O（GPIO）.....617
13.1 はじめに.....617
13.2 GPIO の主な機能.....617
13.3 GPIO の機能説明.....617
13.3.1 汎用 I/O（GPIO）.....619
13.3.2 I/O ピンオルタネート機能マルチプレクサと配置.....620
13.3.3 I/O ポート制御レジスタ.....620
13.3.4 I/O ポートデータレジスタ.....621
13.3.5 I/O データのビット単位の操作.....621
13.3.6 GPIO ロック機構.....621
13.3.7 I/O オルタネート機能の入力／出力.....622
13.3.8 外部割込み／ウェイクアップライン.....622
13.3.9 入力設定.....622

13.3.10 出力設定 623
13.3.11 オルタネート機能設定 623
13.3.12 アナログ設定 624
13.3.13 HSE または LSE オシレータのピンを GPIO として使用 625
13.3.14 GPIO ピンを RTC 供給ドメインで使用 625
13.3.15 PH3 を GPIO として使用 625
13.3.16 PA11 および PA12 を GPIO として使用（STM32U59x/5Ax/5Fx/5Gx のみ） . . . 625
13.3.17 OPAMPx_VINM 専用ピン 625
13.3.18 TrustZone セキュリティ 625
13.3.19 特権および非特権モード 627
13.3.20 ハイスピード低電圧モード（HSLV） 627
13.3.21 I/O 補正セル 627
13.4 GPIO レジスタ 628
13.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x = A ~ J） 628
13.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPER）（x = A ~ J） 629
13.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ~ J） 629
13.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A ~ J） 630
13.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A ~ J） 630
13.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A~J） 631
13.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A~J） 631
13.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ~ J） 632
13.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A~J） 633
13.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ~ J） 634
13.4.11 GPIO ポートビットリセットレジスタ（GPIOx_BRR）（x = A ~ J） 635
13.4.12 GPIO ハイスピード低電圧レジスタ（GPIOx_HSLVR）（x = A~J） 635
13.4.13 GPIO セキュア設定レジスタ（GPIOx_SECCFGR）（x = A~J） 636
13.4.14 GPIO レジスタマップ 637

14 低消費電力汎用 I/O（LPGPIO） 639
14.1 はじめに 639
14.2 LPGPIO の主な機能 639
14.3 LPGPIO の機能詳細 639
14.3.1 LPGPIO および GPIO の設定 639
14.3.2 LPGPIO 制御レジスタ 639
14.3.3 LPGPIO I/O データレジスタ 639
14.3.4 LPGPIO I/O データのビット単位の操作 640

14.3.5 セキュリティ保護 640
14.3.6 セキュアクロックおよびリセット管理 640
14.4 LPGPIO レジスタ 641
14.4.1 LPGPIO ポートモードレジスタ（LPGPIO_MODER） 641
14.4.2 LPGPIO ポート入力データレジスタ（LPGPIO_IDR） 641
14.4.3 LPGPIO ポート出力データレジスタ（LPGPIO_ODR） 642
14.4.4 LPGPIO ポートビットセット／リセットレジスタ（LPGPIO_BSRR） 642
14.4.5 LPGPIO ポートビットリセットレジスタ（LPGPIO_BRR） 643
14.4.6 LPGPIO レジスタマップ 643

15 システム設定コントローラ（SYSCFG） 644

15.1 SYSCFG の主な機能 644
15.2 SYSCFG の機能説明 644
15.2.1 I/O 補正セルの管理 644
15.2.2 SYSCFG TrustZone のセキュリティと特権 645
15.2.3 OTG_HS PHY の設定（STM32U59x/5Ax/5Fx/5Gx の場合のみ） 647
15.2.4 HSPI 供給容量の調整（STM32U59x/5Ax/5Fx/5Gx の場合のみ） 647
15.2.5 DCACHE2 による内部 SRAM キャッシュ機能（STM32U59x/5Ax/5Fx/5Gx の場合のみ） 647
15.3 SYSCFG レジスタ 648
15.3.1 SYSCFG セキュア設定レジスタ（SYSCFG_SECCFGR） 648
15.3.2 SYSCFG 設定レジスタ 1（SYSCFG_CFGR1） 649
15.3.3 SYSCFG FPU 割込みマスクレジスタ（SYSCFG_FPUIMR） 651
15.3.4 SYSCFG CPU 非セキュアロックレジスタ（SYSCFG_CNSLCKR） 651
15.3.5 SYSCFG CPU セキュアロックレジスタ（SYSCFG_CSLCKR） 652
15.3.6 SYSCFG 設定レジスタ 2（SYSCFG_CFGR2） 653
15.3.7 SYSCFG メモリ消去ステータスレジスタ（SYSCFG_MESR） 654
15.3.8 SYSCFG 補正セル制御／ステータスレジスタ（SYSCFG_CCCSR） 655
15.3.9 SYSCFG 補正セル値レジスタ（SYSCFG_CCVR） 657
15.3.10 SYSCFG 補正セルコードレジスタ（SYSCFG_CCCR） 658
15.3.11 SYSCFG RSS コマンドレジスタ（SYSCFG_RSSCMDR） 659
15.3.12 SYSCFG OTG_HS PHY レジスタ（SYSCFG_OTGHSPHYCR） 659
15.3.13 SYSCFG OTG_HS PHY 調整レジスタ 2（SYSCFG_OTGHSPHYTUNER2） 660
15.3.14 SYSCFG レジスタマップ 661

16 ペリフェラル相互接続マトリックス 663
16.1 相互接続マトリックスの概要 663
16.2 接続の一覧 664
16.3 相互接続の詳細 666
16.3.1 タイマのマスタ / スレーブ相互接続 666
16.3.2 ADC へのトリガ 666
16.3.3 タイマのトリガとしての ADC アナログウォッチドッグ 668
16.3.4 DAC へのトリガ 668
16.3.5 MDF1 または ADF1 のトリガ 669
16.3.6 MDF1 からのタイマブレーク 669
16.3.7 タイマへのクロックソース 670
16.3.8 低電力タイマへのトリガ 671
16.3.9 コンパレータへのブランキングソース 671
16.3.10 タイマへの入力としての RTC ウェイクアップ 672
16.3.11 タイマへのトリガとしての OTG_FS/OTG_HS SOF 672
16.3.12 タイマへの入力、トリガ、またはブレーク信号としてのコンパレータ 672
16.3.13 タイマへのブレーク信号としてのシステムエラー 674
16.3.14 IRTIM 信号を生成するタイマ 674
16.3.15 通信ペリフェラルのためのトリガ 674
16.3.16 GPDMA/LPDMA へのトリガ 675
16.3.17 アナログペリフェラルへの内部アナログ信号 676
16.3.18 MDF1 による ADC データのフィルタリング 676
16.3.19 DAC サンプルおよびホールドモードのクロックソース 677
16.3.20 グラフィックインタフェースからタイマへのトリガ 677
16.3.21 内部タンパソース 678
16.3.22 タンパから RTC への出力 678
16.3.23 AES/SAES への暗号キー 678

17 汎用ダイレクト･メモリ･アクセス･コントローラ（GPDMA） 679
17.1 GPDMA の概要 679
17.2 GPDMA の主な機能 679
17.3 GPDMA の実装 680
17.3.1 GPDMA チャネル 680
17.3.2 低消費電力モードにおける GPDMA 自律モード 681
17.3.3 GPDMA リクエスト 681
17.3.4 GPDMA ブロック･リクエスト 685
17.3.5 GPDMA トリガ 685

17.4 GPDMA の機能説明 687
17.4.1 GPDMA ブロック図 687
17.4.2 GPDMA チャネル状態とリンクリストなしの直接プログラミング 688
17.4.3 GPDMA チャネルのサスペンドとレジューム 689
17.4.4 GPDMA チャネルのアボートとリスタート 690
17.4.5 GPDMA のリンクリスト･データ構造 691
17.4.6 リンクリスト･アイテムの転送実行 694
17.4.7 GPDMA チャネル状態と run-to-completion モードでのリンクリスト･プログラミング 694
17.4.8 リンク･ステップ･モードでの GPDMA チャネル状態およびリンクリスト･プログラミング 698
17.4.9 GPDMA チャネル状態およびリンクリスト･プログラミング 705
17.4.10 GPDMA FIFO を使用した転送 707
17.4.11 GPDMA 転送リクエストとアービトレーション 714
17.4.12 GPDMA トリガ転送 718
17.4.13 リンクリスト･プログラミングでの GPDMA サーキュラ･バッファリング 720
17.4.14 GPDMA セキュア／非セキュア･チャネル 721
17.4.15 GPDMA 特権／非特権チャネル 722
17.4.16 GPDMA エラー管理 722
17.4.17 GPDMA 自律モード 724
17.5 デバッグ･モードでの GPDMA 725
17.6 低消費電力モードでの GPDMA 725
17.7 GPDMA 割込み 726
17.8 GPDMA レジスタ 727
17.8.1 GPDMA セキュア設定レジスタ（GPDMA_SECCFGR） 727
17.8.2 GPDMA 特権設定レジスタ（GPDMA_PRIVCFGR） 728
17.8.3 GPDMA 設定ロック･レジスタ（GPDMA_RCFGLOCKR） 729
17.8.4 GPDMA 非セキュア･マスク割込みステータス･レジスタ（GPDMA_MISR） . . 729
17.8.5 GPDMA セキュア･マスク割込みステータス･レジスタ（GPDMA_SMISR） . . . 730
17.8.6 GPDMA チャネル x のリンクリスト･ベース･アドレス･レジスタ（GPDMA_CxLBAR） 731
17.8.7 GPDMA チャネル x フラグ･クリア･レジスタ（GPDMA_CxFCR） 731
17.8.8 GPDMA チャネル x ステータス･レジスタ（GPDMA_CxSR） 732
17.8.9 GPDMA チャネル x 制御レジスタ（GPDMA_CxCR） 734
17.8.10 GPDMA チャネル x 転送レジスタ 1（GPDMA_CxTR1） 736
17.8.11 GPDMA チャネル x 転送レジスタ 2（GPDMA_CxTR2） 739
17.8.12 GPDMA チャネル x ブロック･レジスタ 1（GPDMA_CxBR1） 743
17.8.13 GPDMA チャネル x オルタネート･ブロック･レジスタ 1（GPDMA_CxBR1） 745

17.8.14 GPDMA チャネル x 転送元アドレスレジスタ（GPDMA_CxSAR）.....747
17.8.15 GPDMA チャネル x 転送先アドレス･レジスタ（GPDMA_CxDAR）.....748
17.8.16 GPDMA チャネル x 転送レジスタ 3（GPDMA_CxTR3）.....749
17.8.17 GPDMA チャネル x ブロック･レジスタ 2（GPDMA_CxBR2）.....750
17.8.18 GPDMA チャネル x のリンクリスト･アドレス･レジスタ（GPDMA_CxLLR）.....751
17.8.19 GPDMA チャネル x オルタネート･リンクリスト･アドレス･レジスタ（GPDMA_CxLLR）.....753
17.8.20 GPDMA レジスタマップ 755

18 低電力ダイレクトメモリアクセスコントローラ（LPDMA）..... 757

18.1 LPDMA の概要 757
18.2 LPDMA の主な機能 757
18.3 LPDMA の実装 758
18.3.1 LPDMA チャネル 758
18.3.2 低電力モードでの LPDMA 自律モード 758
18.3.3 LPDMA リクエスト 759
18.3.4 LPDMA ブロック･リクエスト 759
18.3.5 LPDMA トリガ 760
18.4 LPDMA の機能説明 761
18.4.1 LPDMA ブロック図 761
18.4.2 LPDMA チャネル状態とリンクリストなしの直接プログラミング 761
18.4.3 LPDMA チャネルのサスペンドとレジューム 763
18.4.4 LPDMA チャネルのアボートとリスタート 764
18.4.5 LPDMA のリンクリスト･データ構造 765
18.4.6 リンクリスト･アイテムの転送実行 766
18.4.7 LPDMA チャネル状態と run-to-completion モードでのリンクリスト･プログラミング 767
18.4.8 リンク･ステップ･モードでの LPDMA チャネル状態およびリンクリスト･プログラミング 771
18.4.9 LPDMA チャネル状態およびリンクリスト･プログラミング 777
18.4.10 LPDMA 直接転送 779
18.4.11 LPDMA 転送リクエストとアービトレーション 781
18.4.12 LPDMA トリガ転送 785
18.4.13 リンクリスト･プログラミングでの LPDMA サーキュラ･バッファリング 786
18.4.14 LPDMA セキュア／非セキュア･チャネル 788
18.4.15 LPDMA 特権／非特権チャネル 788
18.4.16 LPDMA エラー管理 789
18.4.17 LPDMA 自律モード 790

18.5 デバッグ･モードでの LPDMA 791
18.6 低消費電力モードでの LPDMA 791
18.7 LPDMA 割込み 792
18.8 LPDMA レジスタ 793
18.8.1 LPDMA セキュア設定レジスタ（LPDMA_SECCFGR） 793
18.8.2 LPDMA 特権設定レジスタ（LPDMA_PRIVCFGR） 794
18.8.3 LPDMA 設定ロック･レジスタ（LPDMA_RCFGLOCKR） 795
18.8.4 LPDMA 非セキュアマスク割込みステータス･レジスタ（LPDMA_MISR） 795
18.8.5 LPDMA セキュア･マスク割込みステータス･レジスタ（LPDMA_SMISR） . . . 796
18.8.6 LPDMA チャネル x のリンクリスト･ベースのアドレス･レジスタ（LPDMA_CxLBAR） 797
18.8.7 LPDMA チャネル x フラグ･クリア･レジスタ（LPDMA_CxFCR） 797
18.8.8 LPDMA チャネル x ステータス･レジスタ（LPDMA_CxSR） 798
18.8.9 LPDMA チャネル x 制御レジスタ（LPDMA_CxCR） 800
18.8.10 LPDMA チャネル x 転送レジスタ 1（LPDMA_CxTR1） 802
18.8.11 LPDMA チャネル x 転送レジスタ 2（LPDMA_CxTR2） 804
18.8.12 LPDMA チャネル x ブロック･レジスタ 1（LPDMA_CxBR1） 806
18.8.13 LPDMA チャネル x 転送元アドレス･レジスタ（LPDMA_CxSAR） 807
18.8.14 LPDMA チャネル x 転送先アドレス･レジスタ（LPDMA_CxDAR） 808
18.8.15 LPDMA チャネル x のリンクリスト･アドレス･レジスタ（LPDMA_CxLLR） . . . 809
18.8.16 LPDMA レジスタマップ 810

19 Chrom-Art Accelerator コントローラ（DMA2D） 812

19.1 DMA2D の概要 812
19.2 DMA2D の主な機能 812
19.3 DMA2D の機能詳細 813
19.3.1 DMA2D ブロック図 813
19.3.2 DMA2D 内部信号 814
19.3.3 DMA2D 制御 815
19.3.4 DMA2D フォアグラウンドおよびバックグラウンド FIFO 815
19.3.5 DMA2D フォアグラウンドおよびバックグラウンドのピクセルフォーマットコンバータ（PFC） 815
19.3.6 DMA2D フォアグラウンドおよびバックグラウンド CLUT インタフェース 817
19.3.7 DMA2D ブレンダ 819
19.3.8 DMA2D 出力 PFC 819
19.3.9 DMA2D 出力 FIFO 820
19.3.10 DMA2D 出力 FIFO バイトの順序変更 820

19.3.11 DMA2D AHB マスタポートタイマ 822
19.3.12 DMA2D トランザクション 822
19.3.13 DMA2D 設定 823
19.3.14 DMA2D 転送制御（開始、サスペンド、アボート、完了） 826
19.3.15 ウォーターマーク 826
19.3.16 エラー管理 827
19.3.17 AHB デッドタイム 827
19.4 DMA2D 割込み 827
19.5 DMA2D レジスタ 828
19.5.1 DMA2D 制御レジスタ（DMA2D_CR） 828
19.5.2 DMA2D 割込みステータスレジスタ（DMA2D_ISR） 830
19.5.3 DMA2D 割込みフラグクリアレジスタ（DMA2D_IFCR） 830
19.5.4 DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR） 831
19.5.5 DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR） 831
19.5.6 DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR） 832
19.5.7 DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR） 832
19.5.8 DMA2D フォアグラウンド PFC 制御レジスタ（DMA2D_FGPFCCR） 833
19.5.9 DMA2D フォアグラウンドカラーレジスタ（DMA2D_FGCOLR） 834
19.5.10 DMA2D バックグラウンド PFC 制御レジスタ（DMA2D_BGPFCCR） 835
19.5.11 DMA2D バックグラウンドカラーレジスタ（DMA2D_BGCOLR） 837
19.5.12 DMA2D フォアグラウンド CLUT メモリアドレスレジスタ（DMA2D_FGCMAR） 838
19.5.13 DMA2D バックグラウンド CLUT メモリアドレスレジスタ（DMA2D_BGCMAR） 838
19.5.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR） 839
19.5.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR） 840
19.5.16 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR） 840
19.5.17 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR） 841
19.5.18 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR） 841
19.5.19 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR） 842
19.5.20 DMA2D 出力オフセットレジスタ（DMA2D_OOR） 842
19.5.21 DMA2D ライン数レジスタ（DMA2D_NLR） 843
19.5.22 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR） 843
19.5.23 DMA2D AHB マスタタイマ設定レジスタ（DMA2D_AMTCR） 844
19.5.24 DMA2D フォアグラウンド CLUT（DMA2D_FGCLUTx） 844
19.5.25 DMA2D バックグラウンド CLUT（DMA2D_BGCLUTx） 845
19.5.26 DMA2D レジスタマップ 845

20 Chrom-Art Accelerator コントローラ（DMA2D）. 847

20.1 DMA2D の概要 847

20.2 DMA2D の主な機能 847

20.3 DMA2D の機能詳細 848

20.3.1 概要 848

20.3.2 DMA2D 内部信号 849

20.3.3 DMA2D 制御 850

20.3.4 DMA2D フォアグラウンドおよびバックグラウンド FIFO 850

20.3.5 DMA2D フォアグラウンドおよびバックグラウンド PFC 850

20.3.6 DMA2D フォアグラウンドおよびバックグラウンド CLUT インタフェース 853

20.3.7 DMA2D ブレンダ 854

20.3.8 DMA2D 出力 PFC 854

20.3.9 DMA2D 出力 FIFO 855

20.3.10 DMA2D 出力 FIFO バイトの順序変更 855

20.3.11 DMA2DAHB マスタポートタイマ 857

20.3.12 DMA2D トランザクション 857

20.3.13 DMA2D 設定 857

20.3.14 YCbCr サポート 861

20.3.15 DMA2D 転送制御（開始、一時停止、強制終了、完了） 862

20.3.16 ウォーターマーク 862

20.3.17 エラー管理 862

20.3.18 AHB デッドタイム 862

20.4 DMA2D 割込み 863

20.5 DMA2D レジスタ 864

20.5.1 DMA2D 制御レジスタ（DMA2D_CR） 864

20.5.2 DMA2D 割込みステータスレジスタ（DMA2D_ISR） 866

20.5.3 DMA2D 割込みフラグクリアレジスタ（DMA2D_IFCR） 866

20.5.4 DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR） 867

20.5.5 DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR） 867

20.5.6 DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR） 868

20.5.7 DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR） 868

20.5.8 DMA2D フォアグラウンド PFC 制御レジスタ（DMA2D_FGPFCCR） 869

20.5.9 DMA2D フォアグラウンドカラーレジスタ（DMA2D_FGCOLR） 871

20.5.10 DMA2D バックグラウンド PFC 制御レジスタ（DMA2D_BGPFCCR） 871

20.5.11 DMA2D バックグラウンドカラーレジスタ（DMA2D_BGCOLR） 873

20.5.12 DMA2D フォアグラウンド CLUT メモリアドレスレジスタ（DMA2D_FGCMAR） 873

20.5.13 DMA2D バックグラウンド CLUT メモリアドレスレジスタ（DMA2D_BGCMAR） 874

20.5.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）.....874
20.5.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）.....875
20.5.16 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR）.....876
20.5.17 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR）.....876
20.5.18 DMA2D 出力カラーレジスタ [オルタネート]（DMA2D_OCOLR）.....877
20.5.19 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）.....877
20.5.20 DMA2D 出力オフセットレジスタ（DMA2D_OOR）.....878
20.5.21 DMA2D ライン数レジスタ（DMA2D_NLR）.....878
20.5.22 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）.....879
20.5.23 DMA2DAHB マスタタイマ設定レジスタ（DMA2D_AMTCR）.....879
20.5.24 DMA2D フォアグラウンド CLUT（DMA2D_FGCLUTx）.....880
20.5.25 DMA2D バックグラウンド CLUT（DMA2D_BGCLUTx）.....880
20.5.26 DMA2D レジスタマップ 881

21 Chrom-GRC（GFXMMU）.....883

21.1 はじめに 883
21.2 GFXMMU の主な機能 883
21.3 GFXMMU の実装 883
21.4 GFXMMU の機能およびアーキテクチャに関する説明 884
21.4.1 仮想メモリ 884
21.4.2 MMU アーキテクチャ 887
21.4.3 キャッシュとプリフェッチ機構 889
21.4.4 アドレスのキャッシュ 891
21.5 GFXMMU 割込み 892
21.6 GFXMMU レジスタ 893
21.6.1 GFXMMU 設定レジスタ（GFXMMU_CR）.....893
21.6.2 GFXMMU のステータスレジスタ（GFXMMU_SR）.....895
21.6.3 GFXMMU フラグクリアレジスタ（GFXMMU_FCR）.....895
21.6.4 GFXMMU キャッシュ制御レジスタ（GFXMMU_CCR）.....896
21.6.5 GFXMMU デフォルト値レジスタ（GFXMMU_DVR）.....897
21.6.6 GFXMMU バッファ 0 設定レジスタ（GFXMMU_B0CR）.....897
21.6.7 GFXMMU バッファ 1 設定レジスタ（GFXMMU_B1CR）.....898
21.6.8 GFXMMU バッファ 2 設定レジスタ（GFXMMU_B2CR）.....898
21.6.9 GFXMMU バッファ 3 設定レジスタ（GFXMMU_B3CR）.....899
21.6.10 GFXMMULUT 入力 x、ロー（GFXMMU_LUTxL）.....899
21.6.11 GFXMMU LUT 入力 x、ハイ（GFXMMU_LUTxH）.....900
21.6.12 GFXMMU レジスタマップ 901

22 ネスト化されたベクタ割込みコントローラ（NVIC）. 902

22.1 NVIC の主な機能 902

22.2 SysTick 較正値レジスタ 902

22.3 割込みベクタと例外ベクタ 903

23 拡張割込み／イベントコントローラ（EXTI）. 908

23.1 EXTI の主な機能 908

23.2 EXTI ブロック図 908

23.2.1 ペリフェラルと CPU 間の EXTI 接続 910

23.2.2 EXTI 割込み／イベントのマッピング 910

23.3 EXTI の機能説明 911

23.3.1 EXTI の設定可能なイベント入力によるウェイクアップ 911

23.3.2 EXTI マルチプレクサの選択 912

23.4 EXTI の機能動作 913

23.5 EXTI イベント保護 913

23.5.1 EXTI セキュリティ保護 914

23.5.2 EXTI 特権保護 914

23.6 EXTI レジスタ 915

23.6.1 EXTI 立上がりトリガ選択レジスタ（EXTI_RTSR1）. 915

23.6.2 EXTI 立下がりトリガ選択レジスタ（EXTI_FTSR1）. 916

23.6.3 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER1）. 917

23.6.4 EXTI 立上がりエッジペンディングレジスタ（EXTI_RPR1）. 917

23.6.5 EXTI 立下がりエッジペンディングレジスタ（EXTI_FPR1）. 918

23.6.6 EXTI セキュリティ設定レジスタ（EXTI_SECCFGR1）. 919

23.6.7 EXTI 特権設定レジスタ（EXTI_PRIVCFGR1）. 919

23.6.8 EXTI 外部割込み選択レジスタ（EXTI_EXTICRm）. 920

23.6.9 EXTI ロックレジスタ（EXTI_LOCKR）. 922

23.6.10 割込みマスクによる EXTI CPU ウェイクアップレジスタ（EXTI_IMR1）. . . . 923

23.6.11 イベントマスクによる EXTI CPU ウェイクアップレジスタ（EXTI_EMR1）. . . 923

23.6.12 EXTI レジスタ･マップ 924

24 巡回冗長検査計算ユニット（CRC） 926

24.1 概要 926

24.2 CRC の主な機能 926

24.3 CRC の機能説明 927

24.3.1 CRC ブロック図 927

24.3.2 CRC 内部信号 927

24.3.3 CRC 操作 927

24.4 CRC レジスタ 929

24.4.1 CRC データレジスタ（CRC_DR） 929

24.4.2 CRC 独立型データレジスタ (CRC_IDR) 929

24.4.3 CRC 制御レジスタ（CRC_CR） 930

24.4.4 CRC の初期値（CRC_INIT） 931

24.4.5 CRC 多項式（CRC_POL） 931

24.4.6 CRC レジスタマップ 932

25 CORDIC コプロセッサ（CORDIC） 933

25.1 CORDIC の概要 933

25.2 CORDIC の主な機能 933

25.3 CORDIC の機能詳細 933

25.3.1 概要 933

25.3.2 CORDIC 関数 934

25.3.3 固定小数点表現 940

25.3.4 スケーリングファクタ 940

25.3.5 Precision 940

25.3.6 ゼロオーバーヘッドモード 944

25.3.7 ポーリングモード 945

25.3.8 割込みモード 945

25.3.9 DMA モード 946

25.4 CORDIC レジスタ 947

25.4.1 CORDIC 制御／ステータスレジスタ（CORDIC_CSR） 947

25.4.2 CORDIC 引数レジスタ（CORDIC_WDATA） 949

25.4.3 CORDIC 結果レジスタ（CORDIC_RDATA） 949

25.4.4 CORDIC レジスタマップ 950

26 フィルタ演算アクセラレータ（FMAC）. 951

26.1 FMAC の概要 951

26.2 FMAC の主な機能 951

26.3 FMAC の機能詳細 952

26.3.1 概要 952

26.3.2 ローカルメモリとバッファ 953

26.3.3 入力バッファ 954

26.3.4 出力バッファ 956

26.3.5 初期化機能 958

26.3.6 フィルタ機能 959

26.3.7 固定小数点表現 963

26.3.8 FMAC を用いた FIR フィルタの実装 963

26.3.9 FMAC を用いた IIR フィルタの実装 965

26.3.10 フィルタ初期化の例 967

26.3.11 フィルタ演算の例 968

26.3.12 フィルタ設計のヒント 970

26.4 FMAC レジスタ 971

26.4.1 FMAC X1 バッファ設定レジスタ（FMAC_X1BUFCFG）. 971

26.4.2 FMAC X2 バッファ設定レジスタ（FMAC_X2BUFCFG）. 971

26.4.3 FMAC Y バッファ設定レジスタ（FMAC_YBUFCFG）. 972

26.4.4 FMAC パラメータ･レジスタ（FMAC_PARAM）. 973

26.4.5 FMAC 制御レジスタ（FMAC_CR）. 974

26.4.6 FMAC ステータス･レジスタ（FMAC_SR）. 975

26.4.7 FMAC ライト･データ･レジスタ（FMAC_WDATA）. 977

26.4.8 FMAC リード･データ･レジスタ（FMAC_RDATA）. 977

26.4.9 FMAC レジスタマップ 978

27 フレキシブル･スタティック･メモリ･コントローラ（FSMC）. 979

27.1 概要 979

27.2 FMC の主な特徴 979

27.3 FMC のブロック図 980

27.4 AHB インタフェース 981

27.4.1 サポートされるメモリおよびトランザクション 981

27.5 外部デバイスアドレスマッピング 982

27.5.1 NOR/PSRAM アドレスマッピング 983

27.5.2 NAND Flash メモリ･アドレス･マッピング 983

27.6 NOR 型 Flash ／ PSRAM コントローラ 984
27.6.1 外部メモリインタフェース信号 985
27.6.2 サポートされるメモリおよびトランザクション 987
27.6.3 一般的なタイミング規則 988
27.6.4 NOR Flash/PSRAM コントローラ非同期トランザクション 989
27.6.5 同期トランザクション 1006
27.6.6 NOR/PSRAM コントローラレジスタ 1012
27.7 NAND 型 Flash コントローラ 1020
27.7.1 外部メモリインタフェース信号 1020
27.7.2 NAND 型 Flash によってサポートされるメモリとトランザクション 1021
27.7.3 NAND Flash メモリのタイミング図 1022
27.7.4 NAND Flash 動作 1022
27.7.5 NAND Flash のプリウェイト機能 1023
27.7.6 NAND 型 Flash メモリ内のエラー訂正コード（ECC）の計算 1025
27.7.7 NAND Flash コントローラ･レジスタ 1026
27.7.8 FMC レジスタマップ 1031

28 Octo-SPI インタフェース（OCTOSPI） 1033

28.1 概要 1033
28.2 OCTOSPI の主な機能 1033
28.3 OCTOSPI の実装 1034
28.4 OCTOSPI の機能詳細 1035
28.4.1 OCTOSPI ブロック図 1035
28.4.2 OCTOSPI ピンおよび内部信号 1038
28.4.3 OCTOSPI インタフェースを用いたメモリモード 1039
28.4.4 OCTOSPI レギュラコマンドプロトコル 1039
28.4.5 OCTOSPI レギュラコマンドプロトコル信号インタフェース 1042
28.4.6 HyperBus プロトコル 1046
28.4.7 特有の機能 1050
28.4.8 OCTOSPI の動作モードの概要 1051
28.4.9 OCTOSPI インダイレクトモード 1052
28.4.10 OCTOSPI 自動ステータスポーリングモード 1053
28.4.11 OCTOSPI メモリマップドモード 1054
28.4.12 OCTOSPI の設定の概要 1055
28.4.13 OCTOSPI のシステム設定 1055
28.4.14 OCTOSPI のデバイス設定 1055
28.4.15 OCTOSPI レギュラコマンドモードの設定 1056

28.4.16 OCTOSPI HyperBus プロトコルの設定 1059
28.4.17 OCTOSPI エラー管理 1059
28.4.18 OCTOSPI の BUSY およびアボート 1060
28.4.19 OCTOSPI の再設定または無効化 1060
28.4.20 NCS の動作 1061
28.5 アドレスのアライメントとデータ数 1062
28.6 OCTOSPI 割込み 1063
28.7 OCTOSPI レジスタ 1064
28.7.1 OCTOSPI 制御レジスタ（OCTOSPI_CR） 1064
28.7.2 OCTOSPI デバイス設定レジスタ 1（OCTOSPI_DCR1） 1066
28.7.3 OCTOSPI デバイス設定レジスタ 2（OCTOSPI_DCR2） 1068
28.7.4 OCTOSPI デバイス設定レジスタ 3（OCTOSPI_DCR3） 1069
28.7.5 OCTOSPI デバイス設定レジスタ 4（OCTOSPI_DCR4） 1070
28.7.6 OCTOSPI ステータスレジスタ（OCTOSPI_SR） 1070
28.7.7 OCTOSPI フラグクリアレジスタ（OCTOSPI_FCR） 1071
28.7.8 OCTOSPI データ長レジスタ（OCTOSPI_DLR） 1072
28.7.9 OCTOSPI アドレスレジスタ（OCTOSPI_AR） 1072
28.7.10 OCTOSPI データレジスタ（OCTOSPI_DR） 1073
28.7.11 OCTOSPI ポーリングステータスマスクレジスタ（OCTOSPI_PSMKR） . . . 1074
28.7.12 OCTOSPI ポーリングステータス一致レジスタ（OCTOSPI_PSMAR） 1074
28.7.13 OCTOSPI ポーリングインターバルレジスタ（OCTOSPI_PIR） 1075
28.7.14 OCTOSPI 通信設定レジスタ（OCTOSPI_CCR） 1075
28.7.15 OCTOSPI タイミング設定レジスタ（OCTOSPI_TCR） 1077
28.7.16 OCTOSPI 命令レジスタ（OCTOSPI_IR） 1078
28.7.17 OCTOSPI オルタネートバイトレジスタ（OCTOSPI_ABR） 1078
28.7.18 OCTOSPI 低電力タイムアウトレジスタ（OCTOSPI_LPTR） 1079
28.7.19 OCTOSPI ラップ通信設定レジスタ（OCTOSPI_WPCCR） 1079
28.7.20 OCTOSPI ラップタイミング設定レジスタ（OCTOSPI_WPTCR） 1081
28.7.21 OCTOSPI ラップ命令レジスタ（OCTOSPI_WPIR） 1082
28.7.22 OCTOSPI ラップオルタネートバイトレジスタ（OCTOSPI_WPABR） 1082
28.7.23 OCTOSPI 書込み通信設定レジスタ（OCTOSPI_WCCR） 1083
28.7.24 OCTOSPI 書込みタイミング設定レジスタ（OCTOSPI_WTCR） 1085
28.7.25 OCTOSPI 書込み命令レジスタ（OCTOSPI_WIR） 1085
28.7.26 OCTOSPI 書込みオルタネートバイトレジスタ（OCTOSPI_WABR） 1086
28.7.27 OCTOSPI HyperBus 遅延設定レジスタ（OCTOSPI_HLCR） 1086
28.7.28 OCTOSPI レジスタマップ 1087

29 OCTOSPI の I/O マネージャ（OCTOSPIM）. 1090

29.1 概要 1090

29.2 OCTOSPIM の主な機能 1090

29.3 OCTOSPIM の実装 1090

29.4 OCTOSPIM の機能説明 1091

29.4.1 OCTOSPIM ブロック図 1091

29.4.2 OCTOSPIM の入出力ピン 1091

29.4.3 OCTOSPIM マトリックス 1092

29.4.4 OCTOSPIM マルチプレクス･モード 1092

29.5 OCTOSPIM レジスタ 1094

29.5.1 OCTOSPIM 制御レジスタ（OCTOSPIM_CR） 1094

29.5.2 OCTOSPIM ポート n 設定レジスタ（OCTOSPIM_PnCR） 1094

29.5.3 OCTOSPIM レジスタマップ 1096

30 Hexadeca-SPI インタフェース（HSPI）. 1097

30.1 概要 1097

30.2 HSPI の主な特徴 1097

30.3 HSPI の実装 1098

30.4 HSPI 機能詳細 1099

30.4.1 HSPI ブロック図 1099

30.4.2 HSPI ピンおよび内部信号 1102

30.4.3 メモリに対する HSPI インタフェースのモード 1103

30.4.4 HSPI レギュラコマンドプロトコル 1103

30.4.5 HSPI レギュラコマンドプロトコル信号インタフェース 1107

30.4.6 HyperBus プロトコル 1111

30.4.7 特有の機能 1116

30.4.8 HSPI 動作モードの概要 1117

30.4.9 HSPI インダイレクトモード 1117

30.4.10 HSPI 自動ステータスポーリングモード 1119

30.4.11 HSPI メモリマップドモード 1120

30.4.12 HSPI の設定の概要 1120

30.4.13 HSPI システムの設定 1121

30.4.14 HSPI デバイスの設定 1121

30.4.15 HSPI レギュラコマンドモードの設定 1122

30.4.16 HSPIHyperBus プロトコルの設定 1124

30.4.17 HSPI エラー管理 1125

30.4.18 HSPI ハイスピードインタフェースと較正 1125

30.4.19 HSPI の BUSY および ABORT 1126
30.4.20 HSPI の再設定または無効化 1127
30.4.21 NCS の動作 1127
30.5 アドレスのアライメントとデータ数 1128
30.6 HSPI 割込み 1130
30.7 HSPI レジスタ 1130
30.7.1 HSPI 制御レジスタ（HSPI_CR）. 1130
30.7.2 HSPI デバイス設定レジスタ 1（HSPI_DCR1）. 1133
30.7.3 HSPI デバイス設定レジスタ 2（HSPI_DCR2）. 1134
30.7.4 HSPI デバイス設定レジスタ 3（HSPI_DCR3）. 1135
30.7.5 HSPI デバイス設定レジスタ 4（HSPI_DCR4）. 1136
30.7.6 HSPI ステータスレジスタ（HSPI_SR）. 1136
30.7.7 HSPI フラグクリアレジスタ（HSPI_FCR）. 1137
30.7.8 HSPI データ長レジスタ（HSPI_DLR）. 1138
30.7.9 HSPI アドレスレジスタ（HSPI_AR）. 1139
30.7.10 HSPI データレジスタ（HSPI_DR）. 1139
30.7.11 HSPI ポーリングステータスマスクレジスタ（HSPI_PSMKR）. 1140
30.7.12 HSPI ポーリングステータス一致レジスタ（HSPI_PSMAR）. 1140
30.7.13 HSPI ポーリングインターバルレジスタ（HSPI_PIR）. 1141
30.7.14 HSPI 通信設定レジスタ（HSPI_CCR）. 1141
30.7.15 HSPI タイミング設定レジスタ（HSPI_TCR）. 1143
30.7.16 HSPI 命令レジスタ（HSPI_IR）. 1144
30.7.17 HSPI オルタネートバイトレジスタ（HSPI_ABR）. 1144
30.7.18 HSPI 低電力タイムアウトレジスタ（HSPI_LPTR）. 1144
30.7.19 HSPI ラップ通信設定レジスタ（HSPI_WPCCR）. 1145
30.7.20 HSPI ラップタイミング設定レジスタ（HSPI_WPTCR）. 1147
30.7.21 HSPI ラップ命令レジスタ（HSPI_WPIR）. 1148
30.7.22 HSPI ラップオルタネートバイトレジスタ（HSPI_WPABR）. 1148
30.7.23 HSPI 書込み通信設定レジスタ（HSPI_WCCR）. 1149
30.7.24 HSPI 書込みタイミング設定レジスタ（HSPI_WTCR）. 1151
30.7.25 HSPI 書込み命令レジスタ（HSPI_WIR）. 1151
30.7.26 HSPI ライトオルタネートバイトレジスタ（HSPI_WABR）. 1152
30.7.27 HSPIHyperBus 遅延設定レジスタ（HSPI_HLCR）. 1152
30.7.28 HSPI フルサイクル較正の設定（HSPI_CALFCR）. 1153
30.7.29 HSPIDLL マスタ較正の設定（HSPI_CALMR）. 1153
30.7.30 HSPIDLL スレーブ出力較正の設定 （HSPI_CALSOR）. 1154
30.7.31 HSPIDLL スレーブ入力較正の設定（HSPI_CALSIR）. 1155
30.7.32 HSPI レジスタマップ 1156

31 セキュアデジタル入力／出力 MultiMediaCard インタフェース（SDMMC）. 1159

31.1 SDMMC の主な機能 1159
31.2 SDMMC の実装 1159
31.3 SDMMC バストポロジ 1160
31.4 SDMMC 動作モード 1162
31.5 SDMMC の機能説明 1163
31.5.1 SDMMC ブロック図 1163
31.5.2 SDMMC ピンおよび内部信号 1164
31.5.3 概要 1164
31.5.4 SDMMC アダプタ 1166
31.5.5 SDMMC AHB スレーブインタフェース 1186
31.5.6 SDMMC AHB マスタインタフェース 1186
31.5.7 AHB と SDMMC_CK のクロック関係 1189
31.6 カード機能詳細 1190
31.6.1 SD I/O モード 1190
31.6.2 CMD12 送信タイミング 1198
31.6.3 SLEEP（CMD5） 1201
31.6.4 割込みモード（Wait-IRQ） 1202
31.6.5 ブート操作 1203
31.6.6 レスポンス R1b の処理 1206
31.6.7 リセットとカードサイクルパワー 1207
31.7 ハードウェアフロー制御 1208
31.8 ウルトラハイスピードフェーズ I（UHS-I）の電圧スイッチ 1209
31.9 SDMMC 割込み 1212
31.10 SDMMC レジスタ 1213
31.10.1 SDMMC 電源制御レジスタ（SDMMC_POWER） 1213
31.10.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR） 1214
31.10.3 SDMMC 引数レジスタ（SDMMC_ARGR） 1216
31.10.4 SDMMC コマンドレジスタ（SDMMC_CMDR） 1216
31.10.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMDR） 1218
31.10.6 SDMMC レスポンス x レジスタ（SDMMC_RESPxR） 1218
31.10.7 SDMMC データタイマレジスタ（SDMMC_DTIMER） 1219
31.10.8 SDMMC データ長レジスタ（SDMMC_DLENR） 1220
31.10.9 SDMMC データ制御レジスタ（SDMMC_DCTRL） 1220
31.10.10 SDMMC データカウンタレジスタ（SDMMC_DCNTR） 1222
31.10.11 SDMMC ステータスレジスタ（SDMMC_STAR） 1223

31.10.12 SDMMC 割込みクリアレジスタ（SDMMC_ICR）. 1225
31.10.13 SDMMC マスクレジスタ（SDMMC_MASKR）. 1228
31.10.14 SDMMC 確認応答タイマレジスタ（SDMMC_ACKTIMER）. 1231
31.10.15 SDMMC データ FIFO レジスタ x（SDMMC_FIFORx）. 1231
31.10.16 SDMMC DMA 制御レジスタ（SDMMC_IDMACTRLR）. 1232
31.10.17 SDMMC IDMA バッファサイズレジスタ（SDMMC_IDMABSIZER）. 1232
31.10.18 SDMMC IDMA バッファベースアドレスレジスタ（SDMMC_IDMABASER）. 1233
31.10.19 SDMMC IDMA リンクリストのアドレスレジスタ（SDMMC_IDMALAR）. . . 1233
31.10.20 SDMMC IDMA リンクリストメモリのベースレジスタ（SDMMC_IDMABAR）. 1234
31.10.21 SDMMC レジスタマップ 1235

32 遅延ブロック（DLYB）. 1238

32.1 概要 1238
32.2 DLYB の主な機能 1238
32.3 DLYB の実装 1238
32.4 DLYB の機能説明 1238
32.4.1 DLYB 図 1238
32.4.2 DLYB ピンおよび内部信号 1239
32.4.3 概要 1239
32.4.4 遅延ライン長さ設定手順 1240
32.4.5 出力クロック位相の設定手順 1241
32.5 DLYB レジスタ 1241
32.5.1 DLYB 制御レジスタ（DLYB_CR）. 1241
32.5.2 DLYB 設定レジスタ（DLYB_CFGR）. 1242
32.5.3 DLYB レジスタマップ 1242

33 アナログデジタルコンバータ（ADC12）. 1243

33.1 概要 1243
33.2 ADC の主な機能 1243
33.3 A/D コンバータの実装 1245
33.4 ADC の機能詳細 1246
33.4.1 ADC のブロック図 1246
33.4.2 A/D コンバータのピンおよび内部信号 1247
33.4.3 A/D コンバータのクロック 1249
33.4.4 ADC 接続性 1251
33.4.5 スレーブ AHB インタフェース 1253

33.4.6 ADC ディープ･パワー･ダウン･モード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）....1253
33.4.7 シングルエンドおよび差動入力のチャネル 1253
33.4.8 較正（ADCAL、ADCALLIN、ADC_CALFACT）....1254
33.4.9 ADC オン / オフ制御（ADEN、ADDIS、ADRDY）....1257
33.4.10 ADC 制御ビット書込み時の制約 1258
33.4.11 チャネルの選択（SQRx、JSQRx）....1259
33.4.12 チャネル事前選択レジスタ（ADC_PCSEL）....1259
33.4.13 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）....1260
33.4.14 シングル変換モード（CONT = 0）....1261
33.4.15 連続変換モード（CONT = 1）....1262
33.4.16 変換の開始（ADSTART、JADSTART）....1263
33.4.17 タイミング 1264
33.4.18 実行中の変換の停止（ADSTP、JADSTP）....1264
33.4.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN[1:0]、JEXTSEL、JEXTEN[1:0]）....1266
33.4.20 インジェクトチャネルの管理 1267
33.4.21 不連続モード（DISCEN、DISCNUM、JDISCEN）....1269
33.4.22 プログラム可能な分解能（RES）- 高速変換モード 1270
33.4.23 変換の終了とサンプリングフェーズの終了（EOC、JEOC、EOSMP）....1270
33.4.24 変換シーケンスの終了（EOS、JEOS）....1271
33.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）....1271
33.4.26 低周波数トリガモード（LFTRIG）....1272
33.4.27 データ管理 1273
33.4.28 MDF を使用した変換の管理 1281
33.4.29 動的低電力機能 1281
33.4.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）....1286
33.4.31 オーバーサンプリング回路 1290
33.4.32 デュアル ADC モード 1295
33.4.33 温度センサ 1310
33.4.34 VBAT 電源監視 1311
33.4.35 内部電圧基準の監視 1312
33.5 ADC 割込み 1314
33.6 ADC レジスタ（ADC ごと）....1315
33.6.1 ADC 割込みおよびステータスレジスタ（ADC_ISR）....1315
33.6.2 ADC 割込み有効レジスタ（ADC_IER）....1317

33.6.3 ADC 制御レジスタ（ADC_CR）.....1319
33.6.4 ADC 設定レジスタ（ADC_CFGR1）.....1322
33.6.5 ADC 設定レジスタ 2（ADC_CFGR2）.....1326
33.6.6 ADC サンプル時間レジスタ 1（ADC_SMPR1）.....1329
33.6.7 ADC サンプル時間レジスタ 2（ADC_SMPR2）.....1329
33.6.8 ADC チャネル事前選択レジスタ（ADC_PCSEL）.....1330
33.6.9 ADC レギュラシーケンスレジスタ 1（ADC_SQR1）.....1331
33.6.10 ADC レギュラシーケンスレジスタ 2（ADC_SQR2）.....1332
33.6.11 ADC レギュラシーケンスレジスタ 3（ADC_SQR3）.....1333
33.6.12 ADC レギュラシーケンスレジスタ 4（ADC_SQR4）.....1334
33.6.13 ADC レギュラデータレジスタ（ADC_DR）.....1334
33.6.14 ADC インジェクトシーケンスレジスタ（ADC_JSQR）.....1335
33.6.15 ADC オフセット y レジスタ（ADC_OFRy）.....1336
33.6.16 ADC ゲイン補正レジスタ（ADC_GCOMP）.....1337
33.6.17 ADC インジェクトデータレジスタ（ADC_JDRy）.....1338
33.6.18 A/D コンバータアナログウォッチドッグ 2 設定レジスタ（ADC_AWD2CR）.....1338
33.6.19 A/D コンバータアナログウォッチドッグ 3 設定レジスタ（ADC_AWD3CR）.....1339
33.6.20 ADC ウォッチドッグ閾値レジスタ 1（ADC_LTR1）.....1339
33.6.21 ADC ウォッチドッグ閾値レジスタ 1（ADC_HTR1）.....1340
33.6.22 A/D コンバータウォッチドッグ低閾値レジスタ 2（ADC_LTR2）.....1340
33.6.23 A/D コンバータウォッチドッグ高閾値レジスタ 2（ADC_HTR2）.....1341
33.6.24 A/D コンバータウォッチドッグ低閾値レジスタ 3（ADC_LTR3）.....1341
33.6.25 A/D コンバータウォッチドッグ高閾値レジスタ 3（ADC_HTR3）.....1342
33.6.26 A/D コンバータ差動モード選択レジスタ（ADC_DIFSEL）.....1342
33.6.27 ADC クロック制御レジスタ（ADC_CALFACT）.....1343
33.6.28 ADC 較正係数レジスタ（ADC_CALFACT2）.....1344
33.7 ADC 共通レジスタ 1345
33.7.1 ADC 共通ステータスレジスタ（ADC12_CSR）.....1345
33.7.2 ADC 共通制御レジスタ（ADC12_CCR）.....1347
33.7.3 デュアルモード用 ADC 共通レギュラデータレジスタ（ADC12_CDR）.....1350
33.7.4 32bit デュアルモード用 ADC 共通レギュラデータレジスタ（ADC12_CDR2）.....1351
33.8 ADC レジスタマップ 1352

34 アナログデジタルコンバータ（ADC4） 1355
34.1 概要 1355
34.2 ADC の主な機能 1355
34.3 A/D コンバータの実装 1356
34.4 ADC の機能説明 1358
34.4.1 ADC のブロック図 1358
34.4.2 A/D コンバータのピンおよび内部信号 1358
34.4.3 ADC 電圧レギュレータ（ADVREGEN） 1360
34.4.4 較正（ADCAL） 1360
34.4.5 ADC オン / オフ制御（ADEN、ADDIS、ADRDY） 1362
34.4.6 ADC クロック（PRESC[3:0]） 1363
34.4.7 ADC 接続性 1365
34.4.8 ADC の設定 1366
34.4.9 チャネル選択 (CHSEL, SCANDIR, CHSELRMOD) 1366
34.4.10 プログラマブルサンプリング時間 (SMPx[2:0]) 1367
34.4.11 シングル変換モード（CONT = 0） 1368
34.4.12 連続変換モード（CONT = 1） 1368
34.4.13 変換の開始（ADSTART） 1369
34.4.14 タイミング 1370
34.4.15 実行中の変換の停止（ADSTP） 1371
34.4.16 外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN） 1371
34.4.17 不連続モード（DISCEN） 1372
34.4.18 プログラム可能な分解能（RES）- 高速変換モード 1372
34.4.19 変換の終了、サンプリングフェーズの終了（EOC、EOSMP フラグ） 1373
34.4.20 変換シーケンスの終了（EOS フラグ） 1374
34.4.21 タイミング図の例（シングル／連続モード、ハードウェア／ソフトウェアトリガ） 1374
34.4.22 低周波数トリガモード 1376
34.4.23 データ管理 1377
34.4.24 低電力機能 1380
34.4.25 アナログウィンドウウォッチドッグ 1384
34.4.26 オーバーサンプリング回路 1388
34.4.27 温度センサと内部基準電圧 1391
34.4.28 バッテリ電圧監視 1394
34.4.29 別の ADC による同時動作 1394
34.5 ADC 低電力モード 1395
34.6 ADC 割込み 1395

34.7 ADC レジスタ 1397
34.7.1 ADC 割込みおよびステータスレジスタ（ADC_ISR）. 1397
34.7.2 ADC 割込み有効レジスタ（ADC_IER）. 1399
34.7.3 ADC 制御レジスタ（ADC_CR）. 1401
34.7.4 ADC 設定レジスタ 1（ADC_CFGR1）. 1403
34.7.5 ADC 設定レジスタ 2（ADC_CFGR2）. 1406
34.7.6 ADC サンプリング時間レジスタ（ADC_SMPR）. 1408
34.7.7 ADC ウォッチドッグ閾値レジスタ（ADC_AWD1TR）. 1409
34.7.8 ADC ウォッチドッグ閾値レジスタ（ADC_AWD2TR）. 1409
34.7.9 ADC チャネル選択レジスタ［オルタネート］（ADC_CHSELR）. 1410
34.7.10 ADC チャネル選択レジスタ［オルタネート］（ADC_CHSELR）. 1411
34.7.11 ADC ウォッチドッグ閾値レジスタ（ADC_AWD3TR）. 1413
34.7.12 ADC データレジスタ（ADC_DR）. 1413
34.7.13 ADC 電源制御レジスタ（ADC_PWRR）. 1414
34.7.14 ADC アナログウォッチドッグ 2 設定レジスタ（ADC_AWD2CR）. 1415
34.7.15 ADC アナログウォッチドッグ 3 設定レジスタ（ADC_AWD3CR）. 1415
34.7.16 ADC 較正係数（ADC_CALFACT）. 1416
34.7.17 ADC オプションレジスタ（ADC_OR）. 1416
34.7.18 ADC 共通設定レジスタ ADC_CCR）. 1417
34.8 ADC レジスタ･マップ 1418

35 D/A コンバータ（DAC）. 1420
35.1 概要 1420
35.2 DAC の主な機能 1420
35.3 DAC の実装 1421
35.4 DAC の機能説明 1422
35.4.1 DAC ブロック図 1422
35.4.2 DAC ピンおよび内部信号 1423
35.4.3 DAC クロック 1424
35.4.4 DAC チャネルイネーブル 1424
35.4.5 DAC データフォーマット 1425
35.4.6 DAC 変換 1426
35.4.7 DAC 出力電圧 1427
35.4.8 DAC トリガ選択 1428
35.4.9 DMA リクエスト 1428
35.4.10 ノイズ生成 1429
35.4.11 三角波生成 1430

35.4.12 D/A コンバータチャネルモード 1431
35.4.13 DAC 出力バッファ較正 1434
35.4.14 デュアル D/A コンバータチャネル変換モード（デュアルチャネルを使用できる場合） 1435
35.4.15 DAC 自律モード 1439
35.5 低消費電力モードの DAC 1440
35.6 DAC 割込み 1441
35.7 DAC レジスタ 1442
35.7.1 DAC 制御レジスタ（DAC_CR） 1442
35.7.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR） 1445
35.7.3 DAC チャネル 1 の 12bit 右詰めデータ保持レジスタ（DAC_DHR12R1） . . . 1446
35.7.4 DAC チャネル 1 の 12bit 左詰めデータ保持レジスタ（DAC_DHR12L1） 1446
35.7.5 DAC チャネル 1 の 8bit 右詰めデータ保持レジスタ（DAC_DHR8R1） 1447
35.7.6 DAC チャネル 2 の 12bit 右詰めデータ保持レジスタ（DAC_DHR12R2） . . . 1447
35.7.7 DAC チャネル 2 の 12bit 左詰めデータ保持レジスタ（DAC_DHR12L2） 1448
35.7.8 DAC チャネル 2 の 8bit 右詰めデータ保持レジスタ（DAC_DHR8R2） 1448
35.7.9 デュアル DAC 12bit 右詰めデータ保持レジスタ（DAC_DHR12RD） 1449
35.7.10 デュアル DAC 12bit 左詰めデータ保持レジスタ（DAC_DHR12LD） 1449
35.7.11 デュアル DAC 8bit 右詰めデータ保持レジスタ（DAC_DHR8RD） 1450
35.7.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1） 1450
35.7.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2） 1451
35.7.14 DAC ステータスレジスタ（DAC_SR） 1451
35.7.15 DAC 較正制御レジスタ（DAC_CCR） 1453
35.7.16 DAC モード制御レジスタ（DAC_MCR） 1453
35.7.17 DAC チャネル 1 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR1） 1455
35.7.18 DAC チャネル 2 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2） 1455
35.7.19 DAC サンプルおよびホールド時間レジスタ（DAC_SHHR） 1456
35.7.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR） 1456
35.7.21 DAC 自律モード制御レジスタ（DAC_AUTOCR） 1457
35.7.22 DAC レジスタマップ 1458

36 電圧基準バッファ（VREFBUF）. 1460

36.1 概要 1460

36.2 VREFBUF の実装 1460

36.3 VREFBUF の機能説明 1460

36.4 VREFBUF のトリミング 1461

36.5 VREFBUF レジスタ 1462

36.5.1 VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR）. 1462

36.5.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR） 1463

36.5.3 VREFBUF レジスタマップ 1463

37 コンパレータ（COMP）. 1464

37.1 概要 1464

37.2 COMP の主な機能 1464

37.3 COMP の実装 1464

37.4 COMP の機能説明 1465

37.4.1 COMP のブロック図 1465

37.4.2 COMP ピンおよび内部信号 1465

37.4.3 コンパレータのロック機構 1467

37.4.4 ウィンドウコンパレータ 1467

37.4.5 ヒステリシス 1468

37.4.6 コンパレータの出力のブランキング機能 1468

37.4.7 COMP 電力とスピードモード 1469

37.4.8 スケーラ機能 1469

37.5 COMP 低消費電力モード 1470

37.6 COMP 割込み 1471

37.7 COMP レジスタ 1471

37.7.1 COMP1 制御およびステータスレジスタ（COMP1_CSR）. 1471

37.7.2 COMP2 制御およびステータスレジスタ（COMP2_CSR）. 1473

37.7.3 COMP レジスタマップ 1474

38 オペアンプ（OPAMP）. 1475

38.1 OPAMP の概要 1475

38.2 OPAMP の主な機能 1475

38.3 OPAMP の機能説明 1475

38.3.1 OPAMP のリセットおよびクロック 1475

38.3.2 初期設定 1476

38.3.3 信号接続 1476

38.3.4 OPAMP モード 1477

38.3.5 較正 1480

38.4 OPAMP 低消費電力モード 1482

38.5 OPAMP レジスタ 1482

38.5.1 OPAMP1 制御／ステータス･レジスタ（OPAMP1_CSR） 1482

38.5.2 通常モードでの OPAMP1 オフセットトリミングレジスタ（OPAMP1_OTR） 1484

38.5.3 低消費電力モードでの OPAMP1 オフセットトリミングレジスタ（OPAMP1_LPOTR） 1484

38.5.4 OPAMP2 制御／ステータスレジスタ（OPAMP2_CRS） 1485

38.5.5 通常モードでの OPAMP2 オフセットトリミングレジスタ（OPAMP2_OTR） 1486

38.5.6 低消費電力モードでの OPAMP2 オフセットトリミングレジスタ（OPAMP2_LPOTR） 1487

38.5.7 OPAMP レジスタマップ 1488

39 多機能デジタル･フィルタ（MDF）. 1489

39.1 はじめに 1489

39.2 MDF の主な機能 1490

39.3 MDF の実装 1491

39.4 MDF の機能説明 1492

39.4.1 MDF ブロック図 1492

39.4.2 MDF ピンおよび内部信号 1493

39.4.3 シリアル入力インタフェース（SITF） 1495

39.4.4 ADC スレーブ･インタフェース（ADCITF） 1500

39.4.5 クロック･ジェネレータ（CKGEN） 1501

39.4.6 ビットストリーム･マトリックス（BSMX） 1504

39.4.7 短絡検出回路（SCD） 1505

39.4.8 デジタル･フィルタ処理（DFLT） 1506

39.4.9 限界外検出回路（OLD） 1517

39.4.10 デジタル･フィルタ取得モード 1519

39.4.11 開始シーケンスの例 1529
39.4.12 ブレーク･インタフェース 1531
39.4.13 メモリへのデータ転送 1531
39.4.14 自律モード 1536
39.4.15 レジスタの保護 1537
39.5 MDF 低電力モード 1538
39.6 MDF 割込み 1539
39.7 MDF アプリケーション情報 1540
39.7.1 オーディオ･キャプチャ用の MDF 設定の例 1540
39.7.2 プログラミングの例 1541
39.7.3 接続の例 1543
39.7.4 グローバル周波数応答 1544
39.7.5 総 MDF ゲイン 1545
39.8 MDF レジスタ 1549
39.8.1 MDF グローバル制御レジスタ（MDF_GCR） 1549
39.8.2 MDFMDF クロック･ジェネレータ制御レジスタ（MDF_CKGCR） 1550
39.8.3 MDF シリアル･インタフェース制御レジスタ x (MDF_SITFxCR) 1553
39.8.4 MDF ビットストリーム･マトリックス制御レジスタ x（MDF_BSMXxCR） 1554
39.8.5 MDF デジタル･フィルタ制御レジスタ x（MDF_DFLTxCR） 1555
39.8.6 MDF デジタル･フィルタ設定レジスタ x（MDF_DFLTxCICR） 1557
39.8.7 MDF 再形成フィルタ設定レジスタ x（MDF_DFLTxRSFR） 1559
39.8.8 MDF 積分器設定レジスタ x（MDF_DFLTxINTR） 1560
39.8.9 MDF 限界外検出回路制御レジスタ x（MDF_OLDxCR） 1561
39.8.10 MDF OLDx の閾値下限レジスタ x（MDF_OLDxTHLR） 1562
39.8.11 MDF OLDx の閾値上限レジスタ x（MDF_OLDxTHHR） 1563
39.8.12 MDF 遅延制御レジスタ x（MDF_DLYxCR） 1563
39.8.13 MDF 短絡検出回路の制御レジスタ x（MDF_SCDxCR） 1564
39.8.14 MDFMDF DFLT0 割込み有効化レジスタ 0 （MDF_DFLT0IER） 1565
39.8.15 MDF DFLTx 割込み有効化レジスタ x（MDF_DFLTxIER） 1566
39.8.16 MDF DFLT0 割込みステータス･レジスタ 0 （MDF_DFLT0ISR） 1567
39.8.17 MDF DFLTx 割込みステータス･レジスタ x（MDF_DFLTxISR） 1569
39.8.18 MDF オフセット･エラー補正制御レジスタ x（MDF_OECxCR） 1571
39.8.19 MDF スナップショット･データ･レジスタ x（MDF_SNPSxDR） 1572
39.8.20 MDF デジタル･フィルタのデータ･レジスタ x（MDF_DFLTxDR） 1572
39.8.21 MDF レジスタ･マップ 1573

40 オーディオ･デジタル･フィルタ（ADF）. 1575
40.1 概要 1575
40.2 ADF の主な機能 1575
40.3 ADF の実装 1576
40.4 ADF の機能説明 1577
40.4.1 ADF ブロック図 1577
40.4.2 ADF ピンおよび内部信号 1577
40.4.3 シリアル入力インタフェース（SITF）. 1578
40.4.4 ADC スレーブ･インタフェース（ADCITF） 1583
40.4.5 クロック･ジェネレータ（CKGEN）. 1583
40.4.6 ビットストリーム･マトリックス（BSMX）. 1586
40.4.7 デジタル･フィルタ処理（DFLT）. 1586
40.4.8 デジタル･フィルタ取得モード 1596
40.4.9 開始シーケンスの例 1604
40.4.10 サウンド･アクティビティ検出（SAD）. 1605
40.4.11 メモリへのデータ転送 1613
40.4.12 自律モード 1616
40.4.13 レジスタの保護 1616
40.5 ADF 低電力モード 1617
40.6 ADF 割込み 1618
40.7 ADF アプリケーション情報 1619
40.7.1 オーディオ･キャプチャ用の ADF 設定の例 1619
40.7.2 プログラミングの例 1620
40.7.3 接続の例 1623
40.7.4 グローバル周波数応答 1624
40.7.5 総 ADF ゲイン 1625
40.7.6 SAD 閾値の計算方法 1628
40.8 ADF レジスタ 1633
40.8.1 ADF グローバル制御レジスタ（ADF_GCR） 1633
40.8.2 ADF クロック･ジェネレータ制御レジスタ（ADF_CKGCR）. 1634
40.8.3 ADF シリアル･インタフェース制御レジスタ 0（ADF_SITF0CR）. 1636
40.8.4 ADF ビットストリーム･マトリックス制御レジスタ 0（ADF_BSMX0CR）. . . 1638
40.8.5 ADF デジタル･フィルタ制御レジスタ 0（ADF_DFLT0CR）. 1638
40.8.6 ADF デジタル･フィルタ設定レジスタ 0（ADF_DFLT0CICR）. 1640
40.8.7 ADF 再形成フィルタ設定レジスタ 0（ADF_DFLT0RSFR）. 1642
40.8.8 ADF 遅延制御レジスタ 0（ADF_DLY0CR） 1643
40.8.9 ADF DFLT0 割込み有効化レジスタ（ADF_DFLT0IER）. 1643

40.8.10 ADF DFLT0 割込みステータス･レジスタ 0 （ADF_DFLT0ISR）. 1644
40.8.11 ADF SAD 制御レジスタ（ADF_SADCR）. 1646
40.8.12 ADF SAD 設定レジスタ（ADF_SADCFGR）. 1648
40.8.13 ADF SAD サウンド･レベル･レジスタ（ADF_SADSDLVR）. 1649
40.8.14 ADF SAD 環境雑音レベル･レジスタ（ADF_SADANLVR）. 1649
40.8.15 ADF デジタル･フィルタ･データ･レジスタ 0（ADF_DFLT0DR）. 1650
40.8.16 ADF レジスタマップ 1650

41 デジタルカメラインタフェース（DCMI）. 1653

41.1 概要 1653
41.2 DCMI の主な機能 1653
41.3 DCMI の機能説明 1653
41.3.1 DCMI のブロック図 1654
41.3.2 DCMI ピンおよび内部信号 1654
41.3.3 DCMI クロック 1655
41.3.4 DCMI DMA インタフェース 1655
41.3.5 DCMI 物理インタフェース 1655
41.3.6 DCMI の同期 1657
41.3.7 DCMI キャプチャモード 1659
41.3.8 DCMI クロップ機能 1660
41.3.9 DCMI JPEG フォーマット 1661
41.3.10 DCMI の FIFO 1661
41.3.11 DCMI データフォーマットの説明 1661
41.4 DCMI 割込み 1664
41.5 DCMI レジスタ 1664
41.5.1 DCMI 制御レジスタ（DCMI_CR）. 1664
41.5.2 DCMI ステータスレジスタ（DCMI_SR）. 1667
41.5.3 DCMI raw 割込みステータスレジスタ（DCMI_RIS）. 1668
41.5.4 DCMI 割込み有効レジスタ（DCMI_IER）. 1669
41.5.5 DCMI マスク割込みステータスレジスタ（DCMI_MIS）. 1670
41.5.6 DCMI 割込みクリアレジスタ（DCMI_ICR）. 1671
41.5.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）. 1672
41.5.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR）. 1673
41.5.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT）. 1674
41.5.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE）. 1674
41.5.11 DCMI データレジスタ（DCMI_DR）. 1675
41.5.12 DCMI レジスタマップ 1676

42 パラレル同期スレーブインタフェース（PSSI）. 1677

42.1 はじめに 1677

42.2 PSSI の主な機能 1677

42.3 PSSI の機能説明 1677

42.3.1 PSSI ブロック図 1678

42.3.2 PSSI ピンおよび内部信号 1678

42.3.3 PSSI クロック 1679

42.3.4 PSSI データ管理 1679

42.3.5 PSSI オプション制御信号 1681

42.4 PSSI 割込み 1684

42.5 PSSI レジスタ 1685

42.5.1 PSSI 制御レジスタ（PSSI_CR）. 1685

42.5.2 PSSI ステータスレジスタ（PSSI_SR）. 1687

42.5.3 PSSI raw 割込みステータスレジスタ（PSSI_RIS）. 1687

42.5.4 PSSI 割込み有効レジスタ（PSSI_IER）. 1688

42.5.5 PSSI マスク割込みステータスレジスタ（PSSI_MIS）. 1688

42.5.6 PSSI 割込みクリアレジスタ（PSSI_ICR）. 1689

42.5.7 PSSI データレジスタ（PSSI_DR）. 1690

42.5.8 PSSI レジスタマップ 1691

43 LCD-TFT ディスプレイコントローラ（LTDC）. 1692

43.1 はじめに 1692

43.2 LTDC の主な機能 1692

43.3 LTDC の機能詳細 1693

43.3.1 LTDC のブロック図 1693

43.3.2 LTDC ピンおよび内部信号 1693

43.3.3 LTDC のリセットとクロック 1694

43.4 LTDC のプログラム可能なパラメータ 1696

43.4.1 LTDC グローバル設定パラメータ 1696

43.4.2 レイヤのプログラム可能なパラメータ 1698

43.5 LTDC 割込み 1702

43.6 LTDC プログラミング手順 1703

43.7 LTDC レジスタ 1704

43.7.1 LTDC 同期サイズ設定レジスタ（LTDC_SSCR）. 1704

43.7.2 LTDC バックポーチ設定レジスタ（LTDC_BPCR）. 1704

43.7.3 LTDC アクティブ幅設定レジスタ（LTDC_AWCR）. 1705

43.7.4 LTDC 全幅設定レジスタ（LTDC_TWCR）. 1706
43.7.5 LTDC グローバル制御レジスタ（LTDC_GCR）. 1706
43.7.6 LTDC シャドウ再ロード設定レジスタ（LTDC_SRCR）. 1708
43.7.7 LTDC バックグラウンドカラー設定レジスタ（LTDC_BCCR）. 1708
43.7.8 LTDC 割込み有効レジスタ（LTDC_IER）. 1709
43.7.9 LTDC 割込みステータスレジスタ（LTDC_ISR）. 1710
43.7.10 LTDC 割込みクリアレジスタ（LTDC_ICR）. 1710
43.7.11 LTDC ライン割込み位置設定レジスタ（LTDC_LIPCR）. 1711
43.7.12 LTDC 現在位置ステータスレジスタ（LTDC_CPSR）. 1711
43.7.13 LTDC 現在表示ステータスレジスタ（LTDC_CDSR）. 1712
43.7.14 LTDC レイヤ x 制御レジスタ（LTDC_LxCR）. 1712
43.7.15 LTDC レイヤ x ウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）. . . . 1713
43.7.16 LTDC レイヤ x ウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）. . . . 1714
43.7.17 LTDC レイヤ x カラーキーイング設定レジスタ（LTDC_LxCKCR）. 1715
43.7.18 LTDC レイヤ x ピクセルフォーマット設定レジスタ（LTDC_LxPFCR）. . . . 1715
43.7.19 LTDC レイヤ x 定数アルファ設定レジスタ（LTDC_LxCACR）. 1716
43.7.20 LTDC レイヤ x デフォルト色設定レジスタ（LTDC_LxDCCR）. 1716
43.7.21 LTDC レイヤ x ブレンディング係数設定レジスタ（LTDC_LxBFCR）. 1717
43.7.22 LTDC レイヤ x カラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）. 1718
43.7.23 LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）. . . 1718
43.7.24 LTDC レイヤ x カラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）. 1719
43.7.25 LTDC レイヤ x CLUT 書込みレジスタ（LTDC_LxCLUTWR）. 1719
43.7.26 LTDC レジスタマップ 1720

44 DSI ホスト（DSI）. 1723

44.1 概要 1723
44.2 規格および参照資料 1723
44.3 DSI ホストの主な機能 1724
44.4 DSI ホストの機能説明 1725
44.4.1 概要 1725
44.4.2 DSI ホストのピンと内部信号 1726
44.4.3 サポートされている解像度およびフレームレート 1726
44.4.4 システム·レベルのアーキテクチャ 1727
44.5 機能説明：LTDC インタフェースのビデオ·モード 1729
44.5.1 ビデオ送信モード 1731
44.5.2 ビデオ·モードでの LTDC インタフェース設定の更新 1732

44.6 機能説明：LTDC インタフェースのアダプテッド･コマンド･モード 1734
44.7 機能説明：APB スレーブ汎用インタフェース 1738
44.7.1 汎用インタフェースを使ったパケットの送信 1738
44.8 機能説明：タイムアウトカウンタ 1742
44.8.1 競合エラー検出タイムアウトカウンタ 1742
44.8.2 ペリフェラルレスポンスタイムアウトカウンタ 1743
44.9 機能説明：コマンドの送信 1748
44.9.1 ビデオ･モードでのコマンドの送信 1748
44.9.2 低電力モードでのコマンドの送信 1749
44.9.3 ハイスピードでのコマンドの送信 1753
44.9.4 読出しコマンドの送信 1753
44.9.5 低電力モードでのクロック･レーン 1754
44.10 機能説明：仮想チャネル 1756
44.11 機能説明：ビデオ･モード･パターン･ジェネレータ 1757
44.11.1 カラーバー･パターン 1757
44.11.2 カラー･コーディング 1759
44.11.3 BER テスト･パターン 1759
44.11.4 ビデオ･モード･パターン･ジェネレータの解像度 1760
44.12 機能説明：D-PHY の管理 1761
44.12.1 D-PHY の設定 1761
44.12.2 D-PHY の HS2LP および LP2HS の時間 1761
44.12.3 D-PHY の特殊動作 1762
44.12.4 DSI PLL の制御 1762
44.12.5 D-PHY のバイアス制御 1763
44.13 機能説明：割込みとエラー 1763
44.13.1 DSI ラッパー割込み 1763
44.13.2 DSI ホストの割込みとエラー 1764
44.14 プログラミング手順 1771
44.14.1 プログラミング手順の概要 1771
44.14.2 D-PHY パラメータの設定 1772
44.14.3 DSI ホストのタイミングの設定 1772
44.14.4 フロー制御と DBI インタフェースの設定 1773
44.14.5 DSI ホストの LTDC インタフェースの設定 1773
44.14.6 ビデオ･モードの設定 1774
44.14.7 アダプテッド･コマンド･モードの設定 1776
44.14.8 ビデオ･モード･パターン･ジェネレータの設定 1777
44.14.9 ULPM の管理 1778

44.15 DSI ホスト・レジスタ 1781
44.15.1 DSI ホスト・バージョン・レジスタ（DSI_VR）. 1781
44.15.2 DSI ホスト制御レジスタ（DSI_CR）. 1781
44.15.3 DSI ホスト・クロック制御レジスタ（DSI_CCR）. 1781
44.15.4 DSI ホスト LTDC VCID レジスタ（DSI_LVCIDR）. 1782
44.15.5 DSI ホスト LTDC カラー・コーディング・レジスタ（DSI_LCOLCR）. 1782
44.15.6 DSI ホスト LTDC 極性設定レジスタ（DSI_LPCR）. 1783
44.15.7 DSI ホスト低電力モード設定レジスタ（DSI_LPMCR）. 1784
44.15.8 DSI ホスト・プロトコル設定レジスタ（DSI_PCR）. 1784
44.15.9 DSI ホスト汎用 VCID レジスタ（DSI_GVCIDR）. 1785
44.15.10 DSI ホスト・モード設定レジスタ（DSI_MCR）. 1786
44.15.11 DSI ホスト・ビデオ・モード設定レジスタ（DSI_VMCR）. 1786
44.15.12 DSI ホスト・ビデオ・パケット設定レジスタ（DSI_VPCR）. 1788
44.15.13 DSI ホスト・ビデオ・チャンク設定レジスタ（DSI_VCCR）. 1788
44.15.14 DSI ホスト・ビデオ null パケット設定レジスタ（DSI_VNPCR）. 1789
44.15.15 DSI ホスト・ビデオ HSA 設定レジスタ（DSI_VHSACR）. 1789
44.15.16 DSI ホスト・ビデオ HBP 設定レジスタ（DSI_VHBPCR）. 1789
44.15.17 DSI ホスト・ビデオ・ライン設定レジスタ（DSI_VLCR）. 1790
44.15.18 DSI ホスト・ビデオ VSA 設定レジスタ（DSI_VVSACR）. 1790
44.15.19 DSI ホスト・ビデオ VBP 設定レジスタ（DSI_VVBPCR）. 1790
44.15.20 DSI ホスト・ビデオ VFP 設定レジスタ（DSI_VVFPCR）. 1791
44.15.21 DSI ホスト・ビデオ VA 設定レジスタ（DSI_VVACR）. 1791
44.15.22 DSI ホスト LTDC コマンド設定レジスタ（DSI_LCCR）. 1791
44.15.23 DSI ホスト・コマンド・モード設定レジスタ（DSI_CMCR）. 1792
44.15.24 DSI ホスト汎用ヘッダ設定レジスタ（DSI_GHCR）. 1793
44.15.25 DSI ホスト汎用ペイロード・データ・レジスタ（DSI_GPDR）. 1794
44.15.26 DSI ホスト汎用パケット・ステータスレジスタ（DSI_GPSR）. 1794
44.15.27 DSI ホスト・タイムアウトカウンタ設定レジスタ 0（DSI_TCCR0）. 1796
44.15.28 DSI ホスト・タイムアウトカウンタ設定レジスタ 1（DSI_TCCR1）. 1796
44.15.29 DSI ホスト・タイムアウトカウンタ設定レジスタ 2（DSI_TCCR2）. 1797
44.15.30 DSI ホスト・タイムアウトカウンタ設定レジスタ 3（DSI_TCCR3）. 1797
44.15.31 DSI ホスト・タイムアウトカウンタ設定レジスタ 4（DSI_TCCR4）. 1798
44.15.32 DSI ホスト・タイムアウトカウンタ設定レジスタ 5（DSI_TCCR5）. 1798
44.15.33 DSI ホスト・クロック・レーン設定レジスタ（DSI_CLCR）. 1799
44.15.34 DSI ホスト・クロック・レーン・タイマ設定レジスタ（DSI_CLTCR）. 1799
44.15.35 DSI ホスト・データ・レーン・タイマ設定レジスタ（DSI_DLTCR）. 1800
44.15.36 DSI ホスト PHY 制御レジスタ（DSI_PCTLR）. 1800
44.15.37 DSI ホスト PHY 設定レジスタ（DSI_PCONFR）. 1801

44.15.38 DSI ホスト PHY ULPS 制御レジスタ（DSI_PUCR）.....1801
44.15.39 DSI ホスト PHY TX トリガ設定レジスタ（DSI_PTTCR）.....1802
44.15.40 DSI ホスト PHY ステータスレジスタ（DSI_PSR）.....1802
44.15.41 DSI ホスト割込みおよびステータスレジスタ 0（DSI_ISR0）.....1803
44.15.42 DSI ホスト割込みおよびステータスレジスタ 1（DSI_ISR1）.....1804
44.15.43 DSI ホスト割込みイネーブル･レジスタ 0（DSI_IER0）.....1806
44.15.44 DSI ホスト割込みイネーブル･レジスタ 1 （DSI_IER1）.....1808
44.15.45 DSI ホスト強制割込みレジスタ 0（DSI_FIR0）.....1810
44.15.46 DSI ホスト強制割込みレジスタ 1（DSI_FIR1）.....1811
44.15.47 DSI ホスト･データ･レーン･タイマ読出し設定レジスタ（DSI_DLTRCR）.....1812
44.15.48 DSI ホスト･ビデオ･シャドウ制御レジスタ（DSI_VSCR）.....1812
44.15.49 DSI ホスト LTDC 現在 VCID レジスタ（DSI_LCVCIDR）.....1813
44.15.50 DSI ホスト LTDC 現在カラー･コーディング･レジスタ（DSI_LCCCR）.....1813
44.15.51 DSI ホスト低電力モード現在設定レジスタ（DSI_LPMCCR）.....1814
44.15.52 DSI ホスト･ビデオ･モード現在設定レジスタ（DSI_VMCCR）.....1815
44.15.53 DSI ホスト･ビデオ･パケット現在設定レジスタ（DSI_VPCCR）.....1816
44.15.54 DSI ホスト･ビデオ･チャンク現在設定レジスタ（DSI_VCCCR）.....1816
44.15.55 DSI ホスト･ビデオ null パケット現在設定レジスタ（DSI_VNPCCR）.....1817
44.15.56 DSI ホスト･ビデオ HSA 現在設定レジスタ（DSI_VHSACCR）.....1817
44.15.57 DSI ホスト･ビデオ HBP 現在設定レジスタ（DSI_VHBPCCR）.....1817
44.15.58 DSI ホスト･ビデオ･ライン現在設定レジスタ（DSI_VLCCR）.....1818
44.15.59 DSI ホスト･ビデオ VSA 現在設定レジスタ（DSI_VVSACCR）.....1818
44.15.60 DSI ホスト･ビデオ VBP 現在設定レジスタ（DSI_VVBPCCR）.....1818
44.15.61 DSI ホスト･ビデオ VFP 現在設定レジスタ（DSI_VVFPCCR）.....1819
44.15.62 DSI ホスト･ビデオ VA 現在設定レジスタ（DSI_VVACCR）.....1819
44.15.63 DSI ホスト FIFO およびバッファ･ステータスレジスタ（DSI_FBSR）.....1819
44.16 DSI ラッパー･レジスタ 1822
44.16.1 DSI ラッパー設定レジスタ（DSI_WCFGR）.....1822
44.16.2 DSI ラッパー制御レジスタ（DSI_WCR）.....1823
44.16.3 DSI ラッパー割込みイネーブル･レジスタ（DSI_WIER）.....1824
44.16.4 DSI ラッパー割込みおよびステータスレジスタ（DSI_WISR）.....1824
44.16.5 DSI ラッパー割込みフラグクリアレジスタ（DSI_WIFCR）.....1825
44.16.6 DSI ラッパー PHY 設定レジスタ 0（DSI_WPCR0）.....1826
44.16.7 DSI ラッパー･レギュレータおよび PLL 制御レジスタ（DSI_WRPCR）.....1827
44.16.8 DSI ラッパー PLL チューニング･レジスタ （DSI_WPTR）.....1828
44.17 DSI バイアス･レジスタ 1829
44.17.1 DSI バイアス設定レジスタ（DSI_BCFGR）.....1829

44.18 D-PHY レジスタ 1829
44.18.1 DSI D-PHY クロック帯域制御レジスタ（DSI_DPCBCR） 1829
44.18.2 DSI D-PHY クロック･スルーレート制御レジスタ（DSI_DPCSRCR） 1830
44.18.3 DSI D-PHY データ･レーン 0 HS オフセット制御レジスタ（DSI_DPDL0HSOCR） 1830
44.18.4 DSI D-PHY データ･レーン 0 HS LPX オフセット制御レジスタ（DSI_DPDL0LPXOCR） 1831
44.18.5 DSI D-PHY データ･レーン 0 帯域制御レジスタ（DSI_DPDL0BCR） 1832
44.18.6 DSI D-PHY データ･レーン 0 スルーレート制御レジスタ（DSI_DPDL0SRCR） 1832
44.18.7 DSI D-PHY データ･レーン 1 HS オフセット制御レジスタ（DSI_DPDL1HSOCR） 1833
44.18.8 DSI D-PHY データ･レーン 1 HS LPX オフセット制御レジスタ（DSI_DPDL1LPXOCR） 1833
44.18.9 DSI D-PHY データ･レーン 1 帯域制御レジスタ（DSI_DPDL1BCR） 1834
44.18.10 DSI D-PHY データ･レーン 1 スルーレート制御レジスタ（DSI_DPDL1SRCR） 1834
44.18.11 DSI レジスタマップ 1835

45 Neo-Chrom グラフィック･プロセッサ（GPU2D） 1841

45.1 GPU2D の概要 1841
45.2 GPU2D の主な機能 1841
45.3 GPU2D の実装 1842
45.4 GPU2D の概要 1843
45.4.1 GPU2D ブロック図 1843
45.4.2 GPU2D のピンおよび内部信号 1843

46 JPEG コーデック（JPEG） 1844

46.1 はじめに 1844
46.2 JPEG コーデックの主な機能 1844
46.3 JPEG コーデックブロック の機能説明 1845
46.3.1 概要 1845
46.3.2 JPEG 内部信号 1845
46.3.3 JPEG デコード手順 1846
46.3.4 JPEG エンコード手順 1848
46.4 JPEG コーデック割込み 1850
46.5 JPEG コーデックレジスタ 1851
46.5.1 JPEG コーデック制御レジスタ（JPEG_CONFR0） 1851
46.5.2 JPEG コーデック設定レジスタ 1（JPEG_CONFR1） 1852

46.5.3 JPEG コーデック設定レジスタ 2（JPEG_CONFR2）.....1853
46.5.4 JPEG コーデック設定レジスタ 3（JPEG_CONFR3）.....1853
46.5.5 JPEG コーデック設定レジスタ x（JPEG_CONFRx）.....1854
46.5.6 JPEG 制御レジスタ（JPEG_CR）.....1855
46.5.7 JPEG ステータスレジスタ（JPEG_SR）.....1856
46.5.8 JPEG クリアフラグレジスタ（JPEG_CFR）.....1857
46.5.9 JPEG データ入力レジスタ（JPEG_DIR）.....1858
46.5.10 JPEG データ出力レジスタ（JPEG_DOR）.....1858
46.5.11 JPEG 量子化メモリ x（JPEG_QMEMx_y）.....1859
46.5.12 JPEG ハフマン最小（JPEG_HUFFMINx_y）.....1860
46.5.13 JPEG ハフマン最小 x（JPEG_HUFFMINx_y）.....1860
46.5.14 JPEG ハフマンベース（JPEG_HUFFBASEx）.....1861
46.5.15 JPEG ハフマンシンボル（JPEG_HUFFSYMBx）.....1861
46.5.16 JPEG DHT メモリ（JPEG_DHTMEMx）.....1862
46.5.17 JPEG ハフマンエンコード ACx（JPEG_HUFFENC_ACx_y）.....1863
46.5.18 JPEG ハフマンエンコード DCx（JPEG_HUFFENC_DCx_y）.....1863
46.5.19 JPEG コーデックレジスタマップ.....1864

47 タッチセンシングコントローラ（TSC）.....1866

47.1 概要.....1866
47.2 TSC の主な機能.....1866
47.3 TSC の機能詳細.....1867
47.3.1 TSC ブロック図.....1867
47.3.2 表面電荷移動取得の概要.....1867
47.3.3 リセットおよびクロック.....1870
47.3.4 電荷移動取得シーケンス.....1870
47.3.5 スペクトル拡散機能.....1871
47.3.6 最大カウントエラー.....1872
47.3.7 サンプリングコンデンサ I/O および チャネル I/O のモード選択.....1872
47.3.8 取得モード.....1873
47.3.9 I/O ヒステリシスおよびアナログスイッチの制御.....1873
47.4 TSC 低電力モード.....1874
47.5 TSC 割込み.....1874
47.6 TSC レジスタ.....1875
47.6.1 TSC 制御レジスタ（TSC_CR）.....1875
47.6.2 TSC 割込み有効レジスタ（TSC_IER）.....1877
47.6.3 TSC 割込みクリアレジスタ（TSC_ICR）.....1878

47.6.4 TSC 割込みステータスレジスタ（TSC_ISR）. 1878
47.6.5 TSC I/O ヒステリシス制御レジスタ（TSC_IOHCR）. 1879
47.6.6 TSC I/O アナログスイッチ制御レジスタ（TSC_IOASCR）. 1879
47.6.7 TSC I/O サンプリング制御レジスタ（TSC_IOSCR）. 1880
47.6.8 TSC I/O チャネル制御レジスタ（TSC_IOCCR）. 1880
47.6.9 TSC I/O グループ制御ステータスレジスタ（TSC_IOGCSR）. 1881
47.6.10 TSC I/O グループ x カウンタレジスタ（TSC_IOGxCR）. 1881
47.6.11 TSC レジスタマップ 1882

48 真性乱数発生器（RNG）. 1884

48.1 概要 1884
48.2 RNG の主な機能 1884
48.3 RNG の機能説明 1885
48.3.1 RNG のブロック図 1885
48.3.2 RNG 内部信号 1885
48.3.3 乱数の生成 1886
48.3.4 RNG 初期化 1888
48.3.5 RNG 動作 1890
48.3.6 RNG クロック供給 1891
48.3.7 エラー管理 1891
48.3.8 RNG の低消費電力時の取り扱い 1892
48.4 RNG 割込み 1893
48.5 RNG 処理時間 1893
48.6 RNG エントロピーソース検証 1894
48.6.1 概要 1894
48.6.2 検証条件 1894
48.6.3 データ収集 1894
48.7 RNG レジスタ 1895
48.7.1 RNG 制御レジスタ（RNG_CR）. 1895
48.7.2 RNG ステータスレジスタ（RNG_SR）. 1897
48.7.3 RNG データレジスタ（RNG_DR）. 1898
48.7.4 RNG 健全性テスト制御レジスタ（RNG_HTCR）. 1898
48.7.5 RNG レジスタマップ 1899

49 AES ハードウェアアクセラレータ（AES） 1900

49.1 概要 1900

49.2 AES の主な特徴 1900

49.3 AES の実装 1901

49.4 AES 機能詳細 1901

49.4.1 AES ブロック図 1901

49.4.2 AES 内部信号 1902

49.4.3 AES 暗号コア 1902

49.4.4 暗号操作を行うための AES の手順 1908

49.4.5 AES 復号化ラウンドキーの準備 1910

49.4.6 AES 暗号文借用およびデータパディング 1910

49.4.7 AES タスクのサスペンドとレジューム 1911

49.4.8 AES 基本連鎖モード（ECB、CBC） 1912

49.4.9 AES カウンタ（CTR）モード 1916

49.4.10 AES ガロア／カウンタモード（GCM） 1918

49.4.11 AES ガロアメッセージ認証コード（GMAC） 1924

49.4.12 AESCBC-MAC 付きカウンタ（CCM） 1925

49.4.13 共有キーを使った AES の動作 1930

49.4.14 AES データレジスタおよびデータスワッピング 1931

49.4.15 AES キーレジスタ 1933

49.4.16 AES 初期化ベクタレジスタ 1933

49.4.17 AES DMA インタフェース 1934

49.4.18 AES エラー管理 1935

49.5 AES 割込み 1936

49.6 AES 処理遅延時間 1937

49.7 AES レジスタ 1938

49.7.1 AES 制御レジスタ（AES_CR） 1938

49.7.2 AES ステータスレジスタ（AES_SR） 1940

49.7.3 AES データ入力レジスタ（AES_DINR） 1942

49.7.4 AES データ出力レジスタ（AES_DOUTR） 1943

49.7.5 AES キーレジスタ 0（AES_KEYR0） 1943

49.7.6 AES キーレジスタ 1（AES_KEYR1） 1944

49.7.7 AES キーレジスタ 2（AES_KEYR2） 1944

49.7.8 AES キーレジスタ 3（AES_KEYR3） 1944

49.7.9 AES 初期化ベクタレジスタ 0（AES_IVR0） 1945

49.7.10 AES 初期化ベクタレジスタ 1（AES_IVR1） 1945

49.7.11 AES 初期化ベクタレジスタ 2（AES_IVR2） 1946

49.7.12 AES 初期化ベクタレジスタ 3（AES_IVR3）. 1946
49.7.13 AES キーレジスタ 4（AES_KEYR4）. 1946
49.7.14 AES キーレジスタ 5（AES_KEYR5）. 1947
49.7.15 AES キーレジスタ 6（AES_KEYR6）. 1947
49.7.16 AES キーレジスタ 7（AES_KEYR7）. 1947
49.7.17 AES サスペンドレジスタ（AES_SUSPxR）. 1948
49.7.18 AES 割込み有効レジスタ（AES_IER）. 1948
49.7.19 AES 割込みステータスレジスタ（AES_ISR）. 1949
49.7.20 AES 割込みクリアレジスタ（AES_ICR）. 1950
49.7.21 AES レジスタマップ 1951

50 セキュア AES コプロセッサ（SAES）. 1953

50.1 概要 1953
50.2 SAES の主な特徴 1954
50.3 SAES の実装 1954
50.4 SAES 機能詳細 1955
50.4.1 SAES ブロック図 1955
50.4.2 SAES 内部信号 1955
50.4.3 SAES 暗号コア 1956
50.4.4 暗号操作を行うための SAES の手順 1958
50.4.5 SAES 復号ラウンド·キーの準備 1961
50.4.6 SAES 暗号文借用およびデータパディング 1961
50.4.7 SAES タスクのサスペンドとレジューム 1962
50.4.8 SAES 基本連鎖モード（ECB、CBC）. 1962
50.4.9 SAES ラップキーを使った の動作 1967
50.4.10 共有キーを使った SAES の動作 1970
50.4.11 SAES データレジスタおよびデータスワッピング 1972
50.4.12 SAES キーレジスタ 1974
50.4.13 SAES 初期化ベクタレジスタ 1975
50.4.14 SAES DMA インタフェース 1976
50.4.15 SAES エラー管理 1977
50.5 SAES 割込み 1979
50.6 SAES 処理遅延時間 1980
50.7 SAES レジスタ 1980
50.7.1 SAES 制御レジスタ（SAES_CR）. 1980
50.7.2 SAES ステータスレジスタ（SAES_SR）. 1983
50.7.3 SAES データ入力レジスタ（SAES_DINR）. 1985

50.7.4 SAES データ出力レジスタ（SAES_DOUTR）.....1986
50.7.5 SAES キーレジスタ 0（SAES_KEYR0）.....1987
50.7.6 SAES キーレジスタ 1（SAES_KEYR1）.....1987
50.7.7 SAES キーレジスタ 2（SAES_KEYR2）.....1988
50.7.8 SAES キーレジスタ 3（SAES_KEYR3）.....1988
50.7.9 SAES 初期化ベクタレジスタ 0（SAES_IVR0）.....1988
50.7.10 SAES 初期化ベクタレジスタ 1（SAES_IVR1）.....1989
50.7.11 SAES 初期化ベクタレジスタ 2（SAES_IVR2）.....1989
50.7.12 SAES 初期化ベクタレジスタ 3（SAES_IVR3）.....1989
50.7.13 SAES キーレジスタ 4（SAES_KEYR4）.....1990
50.7.14 SAES キーレジスタ 5（SAES_KEYR5）.....1990
50.7.15 SAES キーレジスタ 6（SAES_KEYR6）.....1990
50.7.16 SAES キーレジスタ 7（SAES_KEYR7）.....1991
50.7.17 SAES 割込み有効レジスタ（SAES_IER）.....1991
50.7.18 SAES 割込みステータスレジスタ（SAES_ISR）.....1992
50.7.19 SAES 割込みクリアレジスタ（SAES_ICR）.....1993
50.7.20 SAES レジスタマップ.....1994

51 ハッシュプロセッサ（HASH）.....1996

51.1 概要.....1996
51.2 HASH の主な機能.....1996
51.3 ハッシュの実装.....1997
51.4 HASH の機能説明.....1997
51.4.1 HASH ブロック図.....1997
51.4.2 HASH 内部信号.....1997
51.4.3 セキュアハッシュアルゴリズムについて.....1998
51.4.4 メッセージデータの供給.....1998
51.4.5 メッセージダイジェストの計算.....1999
51.4.6 メッセージのパッディング.....2001
51.4.7 HMAC 操作.....2002
51.4.8 HASH サスペンド／レジューム操作.....2004
51.4.9 HASH DMA インタフェース.....2006
51.4.10 HASH エラー管理.....2006
51.5 HASH 割込み.....2006
51.6 HASH 処理時間.....2007
51.7 HASH レジスタ.....2008
51.7.1 HASH 制御レジスタ（HASH_CR）.....2008

51.7.2 HASH データ入力レジスタ（HASH_DIN）. 2010
51.7.3 HASH スタートレジスタ（HASH_STR）. 2011
51.7.4 HASH ダイジェストレジスタ 2012
51.7.5 HASH 割込み有効レジスタ（HASH_IMR）. 2014
51.7.6 HASH ステータスレジスタ（HASH_SR）. 2014
51.7.7 HASH コンテキストスワップレジスタ 2016
51.7.8 HASH レジスタマップ 2017

52 オンザフライ復号化エンジン（OTFDEC）. 2019

52.1 概要 2019
52.2 OTFDEC の主な機能 2019
52.3 OTFDEC の機能説明 2020
52.3.1 OTFDEC ブロック図 2020
52.3.2 OTFDEC 内部信号 2020
52.3.3 OTFDEC オンザフライ復号化 2021
52.3.4 OTFDEC を使用したカウンタモードでの AES 復号化 2022
52.3.5 フロー制御管理 2023
52.3.6 OTFDEC エラー管理 2023
52.4 OTFDEC 割込み 2024
52.5 OTFDEC アプリケーション情報 2025
52.5.1 OTFDEC 初期化処理 2025
52.5.2 OTFDEC および電力管理 2026
52.5.3 OTFDEC での暗号化 2026
52.5.4 OTFDEC 鍵 CRC のソースコード 2028
52.6 OTFDEC レジスタ 2029
52.6.1 OTFDEC 制御レジスタ（OTFDEC_CR）. 2029
52.6.2 OTFDEC 特権アクセス制御設定レジスタ（OTFDEC_PRIVCFGR）. 2030
52.6.3 OTFDEC 領域 x 設定レジスタ（OTFDEC_RxCFGR）. 2030
52.6.4 OTFDEC 領域 x 開始アドレスレジスタ（OTFDEC_RxSTARTADDR）. 2032
52.6.5 OTFDEC 領域 x 終了アドレスレジスタ（OTFDEC_RxENDADDR）. 2033
52.6.6 OTFDEC 領域 x ノンスレジスタ 0（OTFDEC_RxNONCER0）. 2033
52.6.7 OTFDEC 領域 x ノンスレジスタ 1（OTFDEC_RxNONCER1）. 2034
52.6.8 OTFDEC 領域 x 鍵レジスタ 0（OTFDEC_RxKEYR0）. 2034
52.6.9 OTFDEC 領域 x 鍵レジスタ 1（OTFDEC_RxKEYR1）. 2035
52.6.10 OTFDEC 領域 x 鍵レジスタ 2（OTFDEC_RxKEYR2）. 2035
52.6.11 OTFDEC 領域 x 鍵レジスタ 3（OTFDEC_RxKEYR3）. 2036
52.6.12 OTFDEC 割込みステータスレジスタ（OTFDEC_ISR）. 2036

52.6.13 OTFDEC 割込みクリアレジスタ（OTFDEC_ICR）.....2037
52.6.14 OTFDEC 割込み有効レジスタ（OTFDEC_IER）.....2038
52.6.15 OTFDEC レジスタマップ.....2039

53 公開鍵アクセラレータ（PKA）..... 2042
53.1 概要.....2042
53.2 PKA の主な機能.....2042
53.3 PKA の機能詳細.....2043
53.3.1 PKA ブロック図.....2043
53.3.2 PKA 内部信号.....2043
53.3.3 PKA のリセットおよびクロック.....2043
53.3.4 PKA 公開鍵の高速化.....2044
53.3.5 PKA の標準的なアプリケーション.....2045
53.3.6 演算を行うための PKA の手順.....2048
53.3.7 PKA エラー管理.....2049
53.4 PKA 演算モード.....2049
53.4.1 概要.....2049
53.4.2 モンゴメリパラメータ計算.....2050
53.4.3 剰余加算.....2051
53.4.4 剰余減算.....2051
53.4.5 剰余乗算とモンゴメリ乗算.....2051
53.4.6 べき剰余.....2053
53.4.7 モジュラ逆数.....2054
53.4.8 モジュラリダクション.....2055
53.4.9 算術加算.....2055
53.4.10 算術減算.....2055
53.4.11 算術乗算.....2056
53.4.12 算術比較.....2056
53.4.13 RSA CRT べき乗.....2056
53.4.14 楕円曲線 Fp 上の点の確認.....2057
53.4.15 ECC Fp スカラー乗算.....2058
53.4.16 ECDSA 署名.....2059
53.4.17 ECDSA 検証.....2060
53.4.18 ECC 完全加算.....2061
53.4.19 ECC ダブルベースラダー.....2062
53.4.20 ECC 射影からアフィンへ.....2063

53.5 設定と処理時間の例 2063
53.5.1 サポートされている楕円曲線 2063
53.5.2 計算時間 2065
53.6 PKA 割込み 2067
53.7 PKA レジスタ 2067
53.7.1 PKA 制御レジスタ （PKA_CR） 2067
53.7.2 PKA ステータスレジスタ （PKA_SR） 2069
53.7.3 PKA クリアフラグレジスタ（PKA_CLRFR） 2070
53.7.4 PKA RAM 2070
53.7.5 PKA レジスタマップ 2071

54 高機能制御タイマ（TIM1/TIM8） 2072

54.1 TIM1/TIM8 の概要 2072
54.2 TIM1/TIM8 の主な特徴 2072
54.3 TIM1/TIM8 機能詳細 2073
54.3.1 ブロック図 2073
54.3.2 TIM1/TIM8 ピンおよび内部信号 2074
54.3.3 タイムベースユニット 2079
54.3.4 カウンタモード 2081
54.3.5 繰返しカウンタ 2093
54.3.6 外部トリガ入力 2095
54.3.7 クロック選択 2095
54.3.8 キャプチャ／比較チャネル 2099
54.3.9 入力キャプチャモード 2102
54.3.10 PWM 入力モード 2103
54.3.11 強制出力モード 2104
54.3.12 出力比較モード 2104
54.3.13 PWM モード 2106
54.3.14 非対称 PWM モード 2114
54.3.15 組み合わせ PWM モード 2115
54.3.16 組合せ 3 相 PWM モード 2116
54.3.17 相補出力とデッドタイム挿入 2117
54.3.18 ブレーク機能の使用 2120
54.3.19 双方向ブレーク入力 2126
54.3.20 外部イベントによる tim_ocxref 信号のクリア 2128
54.3.21 6 ステップ PWM 生成 2129
54.3.22 ワンパルスモード 2130

54.3.23 再トリガ可能なワンパルスモード 2132
54.3.24 比較モードでのパルス 2133
54.3.25 エンコーダインタフェースモード 2135
54.3.26 方向ビットの出力 2153
54.3.27 UIF ビットの再配置 2153
54.3.28 タイマ入力 XOR 機能 2153
54.3.29 ホールセンサとのインタフェース 2154
54.3.30 タイマの同期 2156
54.3.31 ADC トリガ 2159
54.3.32 DMA バーストモード 2160
54.3.33 TIM1/TIM8DMA リクエスト 2161
54.3.34 デバッグモード 2161
54.4 TIM1/TIM8 低消費電力モード 2161
54.5 TIM1/TIM8 割込み 2162
54.6 TIM1/TIM8 レジスタ 2163
54.6.1 制御レジスタ 1（TIMx_CR1）（x = 1、8） 2163
54.6.2 制御レジスタ 2（TIMx_CR2）（x = 1、8） 2165
54.6.3 TIMx スレーブモード制御レジスタ (TIMx_SMCR)（x = 1、8） 2168
54.6.4 DMA ／割込み有効レジスタ（TIMx_DIER）（x = 1、8） 2172
54.6.5 TIMx ステータスレジスタ（TIMx_SR）（x = 1、8） 2173
54.6.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 1、8） 2176
54.6.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）（x = 1、8） 2177
54.6.8 TIMx キャプチャ／比較モード･レジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 1、8） 2179
54.6.9 TIMx キャプチャ / 比較モードレジスタ 2（TIMx_CCMR2）（x = 1、8） 2181
54.6.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 1、8） 2182
54.6.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8） 2185
54.6.12 TIMx カウンタ（TIMx_CNT）（x = 1、8） 2189
54.6.13 TIMx プリスケーラ（TIMx_PSC）（x = 1、8） 2189
54.6.14 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 1、8） 2190
54.6.15 TIMx 繰返しカウンタレジスタ（TIMx_RCR）（x = 1、8） 2190
54.6.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）（x = 1、8） 2191
54.6.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）（x = 1、8） 2192
54.6.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）（x = 1、8） 2193
54.6.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）（x = 1、8） 2194
54.6.20 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 1、8） . . . 2195
54.6.21 TIMx キャプチャ／比較レジスタ 5 （TIMx_CCR5）（x = 1、8） 2199

54.6.22 TIMx キャプチャ／比較レジスタ 6 （TIMx_CCR6）（x = 1、8） 2200
54.6.23 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）（x = 1、8） 2201
54.6.24 TIMx タイマデッドタイムレジスタ 2（TIMx_DTR2）（x = 1、8） 2202
54.6.25 TIMx タイマエンコーダ制御レジスタ（TIMx_ECR）（x = 1、8） 2203
54.6.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 1、8） 2204
54.6.27 TIMx オルタネート機能オプションレジスタ 1 （TIMx_AF1）（x = 1、8） . . .2205
54.6.28 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 1、8） 2208
54.6.29 DMA 制御レジスタ（TIMx_DCR）（x = 1、8） 2211
54.6.30 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 1、8） 2212
54.6.31 TIMx レジスタマップ 2213

55 汎用タイマ（TIM2/TIM3/TIM4/TIM5） 2216

55.1 TIM2/TIM3/TIM4/TIM5 の概要 2216
55.2 TIM2/TIM3/TIM4/TIM5 の主な特徴 2216
55.3 TIM2/TIM3/TIM4/TIM5 の実装 2217
55.4 TIM2/TIM3/TIM4/TIM5 機能詳細 2218
55.4.1 ブロック図 2218
55.4.2 TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号 2219
55.4.3 タイムベースユニット 2223
55.4.4 カウンタモード 2225
55.4.5 クロック選択 2235
55.4.6 キャプチャ／比較チャネル 2239
55.4.7 入力キャプチャモード 2241
55.4.8 PWM 入力モード 2242
55.4.9 強制出力モード 2243
55.4.10 出力比較モード 2243
55.4.11 PWM モード 2245
55.4.12 非対称 PWM モード 2253
55.4.13 組み合わせ PWM モード 2254
55.4.14 外部イベントによる tim_ocxref 信号のクリア 2255
55.4.15 ワンパルスモード 2257
55.4.16 再トリガ可能なワンパルスモード 2258
55.4.17 比較モードでのパルス 2259
55.4.18 エンコーダインタフェースモード 2261
55.4.19 方向ビットの出力 2279
55.4.20 UIF ビットの再配置 2279
55.4.21 タイマ入力 XOR 機能 2279
55.4.22 タイマと外部トリガの同期 2279

55.4.23 タイマの同期 2283
55.4.24 ADC トリガ 2288
55.4.25 DMA バーストモード 2288
55.4.26 TIM2/TIM3/TIM4/TIM5DMA リクエスト 2289
55.4.27 デバッグモード 2290
55.4.28 TIM2/TIM3/TIM4/TIM5 低消費電力モード 2290
55.4.29 TIM2/TIM3/TIM4/TIM5 割込み 2290
55.5 TIM2/TIM3/TIM4/TIM5 レジスタ 2291
55.5.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 2 〜 5） 2291
55.5.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 2 〜 5） 2293
55.5.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）（x = 2 〜 5） 2294
55.5.4 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 2 〜 5） 2297
55.5.5 TIMx ステータスレジスタ（TIMx_SR）（x = 2 〜 5） 2298
55.5.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 2 〜 5） 2300
55.5.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）（x = 2 〜 5） . . . 2301
55.5.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 2 〜 5） 2303
55.5.9 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）（x = 2 〜 5） . . . 2305
55.5.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 2 〜 5） 2306
55.5.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 2 〜 5） 2307
55.5.12 TIMx カウンタ（TIMx_CNT）（x = 2 〜 5） 2309
55.5.13 TIMx プリスケーラ（TIMx_PSC）（x = 2 〜 5） 2309
55.5.14 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 2 〜 5） 2310
55.5.15 TIMx キャプチャ／比較レジスタ 1（TIMx_CCR1）（x = 2 〜 5） 2311
55.5.16 TIMx キャプチャ／比較レジスタ 2（TIMx_CCR2）（x = 2 〜 5） 2312
55.5.17 TIMx キャプチャ／比較レジスタ 3（TIMx_CCR3）（x = 2 〜 5） 2313
55.5.18 TIMx キャプチャ／比較レジスタ 4（TIMx_CCR4）（x = 2 〜 5） 2314
55.5.19 TIMx タイマエンコーダ制御レジスタ（TIMx_ECR）（x = 2 〜 5） 2315
55.5.20 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 2 〜 5） 2316
55.5.21 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 2 〜 5） 2317
55.5.22 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 2 〜 5） 2318
55.5.23 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 2 〜 5） 2319
55.5.24 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 2 〜 5） 2320
55.5.25 TIMx レジスタマップ 2321

56 汎用タイマ（TIM15/TIM16/TIM17） 2323

56.1 TIM15/TIM16/TIM17 の概要 2323

56.2 TIM15 の主な機能 2323

56.3 TIM16/TIM17 の主な特長 2324

56.4 TIM15/TIM16/TIM17 機能詳細 2325

56.4.1 ブロック図 2325

56.4.2 TIM15/TIM16/TIM17 ピンおよび内部信号 2326

56.4.3 タイムベースユニット 2329

56.4.4 カウンタモード 2331

56.4.5 繰返しカウンタ 2335

56.4.6 クロック選択 2336

56.4.7 キャプチャ／比較チャネル 2338

56.4.8 入力キャプチャモード 2341

56.4.9 PWM 入力モード（TIM15 の場合のみ） 2342

56.4.10 強制出力モード 2343

56.4.11 出力比較モード 2343

56.4.12 PWM モード 2345

56.4.13 組合せ PWM モード（TIM15 のみ） 2350

56.4.14 相補出力とデッドタイム挿入 2351

56.4.15 ブレーク機能の使用 2354

56.4.16 双方向ブレーク入力 2358

56.4.17 外部イベントによる tim_ocxref 信号のクリア 2359

56.4.18 6 ステップ PWM 生成 2360

56.4.19 ワンパルスモード 2361

56.4.20 再トリガ可能なワンパルスモード（TIM15 のみ） 2362

56.4.21 UIF ビットの再配置 2363

56.4.22 タイマ入力 XOR 機能（TIM15 のみ） 2363

56.4.23 外部トリガ同期（TIM15 のみ） 2364

56.4.24 スレーブモード - リセットモードとトリガモードの組合せ（TIM15 のみ） . . . 2366

56.4.25 スレーブモード - リセットモードとゲートモードの組合せ（TIM15 のみ） . . . 2366

56.4.26 タイマ同期（TIM15 のみ） 2368

56.4.27 タイマ出力を他のタイマのトリガとして使用（TIM16/TIM17 のみ） 2368

56.4.28 ADC トリガ（TIM15 のみ） 2368

56.4.29 DMA バーストモード 2368

56.4.30 TIM15/TIM16/TIM17DMA リクエスト 2369

56.4.31 デバッグモード 2369

56.5 TIM15/TIM16/TIM17 低消費電力モード 2370

56.6 TIM15/TIM16/TIM17 割込み 2370
56.7 TIM15 レジスタ 2371
56.7.1 TIM15 制御レジスタ 1 （TIM15_CR1）. 2371
56.7.2 TIM15 制御レジスタ 2（TIM15_CR2） 2372
56.7.3 TIM15 スレーブモード制御レジスタ（TIM15_SMCR）. 2374
56.7.4 TIM15 DMA ／割込み有効レジスタ（TIM15_DIER） 2376
56.7.5 TIM15 ステータスレジスタ（TIM15_SR）. 2377
56.7.6 TIM15 イベント生成レジスタ（TIM15_EGR） 2379
56.7.7 TIM15 キャプチャ／比較モードレジスタ 1（TIM15_CCMR1） 2380
56.7.8 TIM15 キャプチャ／比較モード･レジスタ 1 [オルタネート] （TIM15_CCMR1） 2382
56.7.9 TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）. 2384
56.7.10 TIM15 カウンタ（TIM15_CNT） 2387
56.7.11 TIM15 プリスケーラ（TIM15_PSC）. 2387
56.7.12 TIM15 の自動再ロードレジスタ（TIM15_ARR）. 2388
56.7.13 TIM15 繰返しカウンタレジスタ（TIM15_RCR）. 2388
56.7.14 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1） 2389
56.7.15 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2） 2390
56.7.16 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）. 2391
56.7.17 TIM15 タイマデッドタイムレジスタ 2（TIM15_DTR2）. 2394
56.7.18 TIM15 入力選択レジスタ（TIM15_TISEL）. 2395
56.7.19 TIM15 オルタネート機能レジスタ 1 （TIM15_AF1） 2396
56.7.20 TIM15 オルタネート機能レジスタ 2 （TIM15_AF2） 2398
56.7.21 TIM15 DMA 制御レジスタ（TIM15_DCR）. 2399
56.7.22 完全転送の TIM15 DMA アドレス（TIM15_DMAR）. 2400
56.7.23 TIM15 レジスタマップ 2400
56.8 TIM16/TIM17 レジスタ 2403
56.8.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 16 ～ 17） 2403
56.8.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 16 ～ 17） 2404
56.8.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 16 ～ 17） 2405
56.8.4 TIMx ステータスレジスタ（TIMx_SR）（x = 16 ～ 17） 2406
56.8.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 16 ～ 17） 2407
56.8.6 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1） （x = 16 ～ 17） 2408
56.8.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート] （TIMx_CCMR1）（x = 16 ～ 17） 2409
56.8.8 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16 ～ 17） 2411
56.8.9 TIMx カウンタ（TIMx_CNT）（x = 16 ～ 17） 2413
56.8.10 TIMx プリスケーラ（TIMx_PSC）（x = 16 ～ 17） 2414

56.8.11 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 16 ～ 17）. 2414
56.8.12 TIMx 繰返しカウンタレジスタ（TIMx_RCR）（x = 16 ～ 17）. 2415
56.8.13 TIMx キャプチャ／比較レジスタ 1（TIMx_CCR1）（x = 16 ～ 17）. 2415
56.8.14 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16 ～ 17）. 2417
56.8.15 TIMx タイマデッドタイムレジスタ 2（TIMx_DTR2）（x = 16 ～ 17）. 2420
56.8.16 TIMx 入力選択レジスタ（TIMx_TISEL）（x = 16 ～ 17）. 2421
56.8.17 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 16 ～ 17）. 2421
56.8.18 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 16 ～ 17）. 2424
56.8.19 TIMx オプションレジスタ 1（TIMx_OR1）（x = 16 ～ 17）. 2424
56.8.20 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 16 ～ 17）. 2425
56.8.21 TIM16/TIM17 完全転送用の DMA アドレス（TIMx_DMAR）（x = 16 ～ 17）. 2426
56.8.22 TIM16/TIM17 レジスタマップ 2427

57 基本タイマ（TIM6/TIM7）. 2429

57.1 TIM6/TIM7 の概要 2429
57.2 TIM6/TIM7 の主な特徴 2429
57.3 TIM6/TIM7 機能詳細 2430
57.3.1 TIM6/TIM7 ブロック図 2430
57.3.2 TIM6/TIM7 内部信号 2430
57.3.3 TIM6/TIM7 クロック 2431
57.3.4 タイムベースユニット 2431
57.3.5 カウントモード 2433
57.3.6 UIF ビットの再配置 2440
57.3.7 ADC トリガ 2441
57.3.8 TIM6/TIM7DMA リクエスト 2441
57.3.9 デバッグモード 2441
57.3.10 TIM6/TIM7 低消費電力モード 2441
57.3.11 TIM6/TIM7 割込み 2441
57.4 TIM6/TIM7 レジスタ 2442
57.4.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 6 ～ 7）. 2442
57.4.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 6 ～ 7）. 2444
57.4.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 6 ～ 7）. 2444
57.4.4 TIMx ステータスレジスタ（TIMx_SR）（x = 6 ～ 7）. 2445
57.4.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 6 ～ 7）. 2445
57.4.6 TIMx カウンタ（TIMx_CNT）（x = 6 ～ 7）. 2446
57.4.7 TIMx プリスケーラ（TIMx_PSC）（x = 6 ～ 7）. 2446

57.4.8 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 6 ～ 7）.....2447
57.4.9 TIMx レジスタマップ.....2448

58 低消費電力タイマ（LPTIM）..... 2449

58.1 概要..... 2449
58.2 LPTIM の主な機能..... 2449
58.3 LPTIM の実装..... 2450
58.4 LPTIM の機能説明..... 2451
58.4.1 LPTIM ブロック図.....2451
58.4.2 LPTIM ピンおよび内部信号.....2452
58.4.3 LPTIM 入力およびトリガマッピング.....2454
58.4.4 LPTIM のリセットとクロック.....2455
58.4.5 グリッチフィルタ.....2456
58.4.6 プリスケーラ.....2457
58.4.7 トリガマルチプレクサ.....2457
58.4.8 動作モード.....2458
58.4.9 タイムアウト機能.....2460
58.4.10 波形生成.....2460
58.4.11 レジスタの更新.....2461
58.4.12 カウンタモード.....2462
58.4.13 タイマ有効.....2462
58.4.14 タイマカウンタのリセット.....2463
58.4.15 エンコーダモード.....2463
58.4.16 繰返しカウンタ.....2465
58.4.17 キャプチャ／比較チャネル.....2466
58.4.18 入力キャプチャモード.....2466
58.4.19 PWM モード.....2468
58.4.20 自律モード.....2470
58.4.21 DMA リクエスト.....2471
58.4.22 デバッグモード.....2472
58.5 LPTIM 低消費電力モード..... 2472
58.6 LPTIM 割込み..... 2473
58.7 LPTIM レジスタ..... 2474
58.7.1 LPTIM4 割込みおよびステータスレジスタ（LPTIM4_ISR）.....2474
58.7.2 LPTIMx 割込みとステータスレジスタ [オルタネート]（LPTIMx_ISR）（x = 1 ～ 3）.....2475
58.7.3 LPTIMx 割込みとステータスレジスタ [オルタネート]（LPTIMx_ISR）（x = 1 ～ 3）.....2477

58.7.4 LPTIM4 割込みクリアレジスタ（LPTIM4_ICR）.....2479
58.7.5 LPTIMx 割込みクリアレジスタ [オルタネート]（LPTIMx_ICR）（x = 1 ～ 3）.....2480
58.7.6 LPTIMx 割込みクリアレジスタ [オルタネート]（LPTIMx_ICR）（x = 1 ～ 3）.....2482
58.7.7 LPTIM4 割込み有効レジスタ（LPTIM4_DIER）.....2483
58.7.8 LPTIMx 割込み有効レジスタ [オルタネート]（LPTIMx_DIER）（x = 1 ～ 3）.....2484
58.7.9 LPTIMx 割込み有効レジスタ [オルタネート]（LPTIMx_DIER）（x = 1 ～ 3）.....2486
58.7.10 LPTIM 設定レジスタ（LPTIM_CFGR）.....2488
58.7.11 LPTIM 制御レジスタ（LPTIM_CR）.....2491
58.7.12 LPTIM 比較レジスタ 1（LPTIM_CCR1）.....2492
58.7.13 LPTIM 自動再ロードレジスタ（LPTIM_ARR）.....2493
58.7.14 LPTIM カウンタレジスタ（LPTIM_CNT）.....2493
58.7.15 LPTIM 設定レジスタ 2（LPTIM_CFGR2）.....2494
58.7.16 LPTIM 繰返しレジスタ（LPTIM_RCR）.....2495
58.7.17 LPTIM キャプチャ／比較モードレジスタ 1（LPTIM_CCMR1）.....2495
58.7.18 LPTIM 比較レジスタ 2（LPTIM_CCR2）.....2498
58.7.19 LPTIM レジスタマップ.....2499

59 グラフィック･タイマ（GFXTIM）.....2501

59.1 概要.....2501
59.2 GFXTIM の主な機能.....2501
59.3 GFXTIM の機能詳細.....2501
59.3.1 ブロック図.....2501
59.3.2 GFXTIM ピンおよび内部信号.....2502
59.3.3 クロックジェネレータ.....2503
59.3.4 クロックジェネレータ設定の例.....2505
59.3.5 絶対タイマ.....2509
59.3.6 相対タイマ.....2510
59.3.7 ティアリング効果の検出.....2511
59.3.8 イベント･ジェネレータ.....2511
59.3.9 ウォッチドッグ･タイマ.....2512
59.4 GFXTIM 割込み.....2513
59.5 GFXTIM レジスタ.....2514
59.5.1 GFXTIM 設定レジスタ（GFXTIM_CR）.....2514
59.5.2 GFXTIM クロックジェネレータ設定レジスタ（GFXTIM_CGCR）.....2515
59.5.3 GFXTIM タイマ設定レジスタ（GFXTIM_TCR）.....2517

59.5.4 GFXTIM タイマ無効化レジスタ（GFXTIM_TCR）.....2518
59.5.5 GFXTIM イベント制御レジスタ（GFXTIM_EVCR）.....2519
59.5.6 GFXTIM イベント選択レジスタ（GFXTIM_EVCR）.....2520
59.5.7 GFXTIM ウォッチドッグ･タイマ設定レジスタ（GFXTIM_WDGTCR）.....2522
59.5.8 GFXTIM 割込みステータスレジスタ（GFXTIM_ISR）.....2523
59.5.9 GFXTIM 割込みクリアレジスタ（GFXTIM_ICR）.....2525
59.5.10 GFXTIM 割込み有効レジスタ（GFXTIM_IER）.....2527
59.5.11 GFXTIM タイマ･ステータス･レジスタ（GFXTIM_ISR）.....2529
59.5.12 GFXTIM ライン･クロック･カウンタの再ロードレジスタ（GFXTIM_LCCRR）.....2530
59.5.13 GFXTIM フレーム･クロック･カウンタの再ロードレジスタ（GFXTIM_FCCRR）.....2530
59.5.14 GFXTIM 絶対時間レジスタ（GFXTIM_ATR）.....2531
59.5.15 GFXTIM 絶対フレーム･カウンタ･レジスタ（GFXTIM_AFCR）.....2531
59.5.16 GFXTIM 絶対ライン･カウンタ･レジスタ（GFXTIM_ALCR）.....2532
59.5.17 GFXTIM 絶対フレーム･カウンタ比較 1 レジスタ（GFXTIM_AFCC1R）.....2532
59.5.18 GFXTIM 絶対ライン･カウンタ比較 1 レジスタ（GFXTIM_ALCC1R）.....2533
59.5.19 GFXTIM 絶対ライン･カウンタ比較 2 レジスタ（GFXTIM_ALCC2R）.....2533
59.5.20 GFXTIM 相対フレーム･カウンタ 1 レジスタ（GFXTIM_RFC1R）.....2533
59.5.21 GFXTIM 相対フレーム･カウンタ 1 の再ロード･レジスタ（GFXTIM_RFC1RR）.....2534
59.5.22 GFXTIM 相対フレーム･カウンタ 2 レジスタ（GFXTIM_RFC2R）.....2534
59.5.23 GFXTIM 相対フレーム･カウンタ 2 の再ロード･レジスタ（GFXTIM_RFC2RR）.....2535
59.5.24 GFXTIM ウォッチドッグ･カウンタ･レジスタ（GFXTIM_WDGCR）.....2535
59.5.25 GFXTIM ウォッチドッグ再ロード･レジスタ（GFXTIM_WDGRR）.....2536
59.5.26 GFXTIM ウォッチドッグ･プリアラーム･レジスタ（GFXTIM_WDGPAR）..2536
59.5.27 GFXTIM レジスタマップ.....2537

60 赤外線インタフェース（IRTIM）..... 2539

61 独立型ウォッチドッグ（IWDG）..... 2540

61.1 概要.....2540
61.2 IWDG の主な機能.....2540
61.3 IWDG の実装.....2540
61.4 IWDG の機能説明.....2541
61.4.1 IWDG ブロック図.....2541
61.4.2 IWDG 内部信号.....2542
61.4.3 ソフトウェアおよびハードウェアウォッチドッグモード.....2542

61.4.4 ウィンドウオプション 2543
61.4.5 デバッグ 2545
61.4.6 レジスタのアクセス保護 2545
61.5 IWDG 低電力モード 2546
61.6 IWDG 割込み 2546
61.7 IWDG レジスタ 2548
61.7.1 IWDG キーレジスタ（IWDG_KR） 2548
61.7.2 IWDG プリスケーラレジスタ（IWDG_PR） 2549
61.7.3 IWDG 再ロードレジスタ（IWDG_RLR） 2550
61.7.4 IWDG ステータスレジスタ（IWDG_SR） 2550
61.7.5 IWDG ウィンドウレジスタ（IWDG_WINR） 2551
61.7.6 IWDG 早期ウェイクアップ割込みレジスタ（IWDG_EWCR） 2552
61.7.7 IWDG レジスタマップ 2553

62 システムウィンドウ型ウォッチドッグ（WWDG） 2554
62.1 概要 2554
62.2 WWDG の主な機能 2554
62.3 WWDG の実装 2554
62.4 WWDG の機能説明 2555
62.4.1 WWDG ブロック図 2555
62.4.2 WWDG 内部信号 2555
62.4.3 ウォッチドッグの有効化 2556
62.4.4 ダウンカウンタの制御 2556
62.4.5 ウォッチドッグタイムアウトをプログラムする方法 2556
62.4.6 デバッグモード 2557
62.5 WWDG 割込み 2558
62.6 WWDG レジスタ 2559
62.6.1 WWDG 制御レジスタ（WWDG_CR） 2559
62.6.2 WWDG 設定レジスタ（WWDG_CFR） 2560
62.6.3 WWDG ステータスレジスタ（WWDG_SR） 2560
62.6.4 WWDG レジスタマップ 2561

63 リアルタイムクロック（RTC）. 2562
63.1 概要 2562
63.2 RTC の主な機能 2562
63.3 RTC の機能説明 2562
63.3.1 RTC のブロック図 2562
63.3.2 RTC ピンおよび内部信号 2564
63.3.3 RTC および TAMP によって制御される GPIO 2565
63.3.4 RTC セキュア保護モード 2567
63.3.5 RTC 特権保護モード 2568
63.3.6 クロックとプリスケーラ 2570
63.3.7 リアルタイムクロックとカレンダ 2571
63.3.8 カレンダ超低消費電力モード 2571
63.3.9 プログラム可能なアラーム 2571
63.3.10 周期的自動ウェイクアップ 2572
63.3.11 RTC の初期化と設定 2573
63.3.12 カレンダの読出し 2575
63.3.13 RTC のリセット 2576
63.3.14 RTC の同期 2577
63.3.15 RTC リファレンスクロック検出 2577
63.3.16 RTC の高精度デジタル較正 2578
63.3.17 タイムスタンプ機能 2580
63.3.18 較正クロック出力 2581
63.3.19 タンパおよびアラーム出力 2581
63.4 RTC 低消費電力モード 2582
63.5 RTC 割込み 2582
63.6 RTC レジスタ 2584
63.6.1 RTC 時刻レジスタ（RTC_TR）. 2584
63.6.2 RTC 日付レジスタ（RTC_DR）. 2585
63.6.3 RTC サブセカンドレジスタ（RTC_SSR）. 2586
63.6.4 RTC 初期化制御およびステータスレジスタ（RTC_ICSR）. 2587
63.6.5 RTC プリスケーラレジスタ（RTC_PRER）. 2589
63.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）. 2590
63.6.7 RTC 制御レジスタ（RTC_CR）. 2591
63.6.8 RTC 特権モード制御レジスタ（RTC_PRIVCFGR）. 2594
63.6.9 RTC セキュア設定レジスタ（RCC_SECCFGR）. 2595
63.6.10 RTC 書込み保護レジスタ（RTC_WPR）. 2596
63.6.11 RTC 較正レジスタ（RTC_CALR）. 2597

63.6.12 RTC シフト制御レジスタ（RTC_SHIFTR）. 2598
63.6.13 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）. 2599
63.6.14 RTC タイムスタンプ日付レジスタ（RTC_TSDR）. 2600
63.6.15 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）. 2601
63.6.16 RTC アラーム A レジスタ（RTC_ALRMAR）. 2601
63.6.17 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）. 2603
63.6.18 RTC アラーム B レジスタ（RTC_ALRMBR）. 2604
63.6.19 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）. 2605
63.6.20 RTC ステータスレジスタ（RTC_SR）. 2606
63.6.21 RTC 非セキュアマスク済み割込みステータスレジスタ（RTC_MISR）. 2607
63.6.22 RTC セキュアマスク済み割込みステータスレジスタ（RTC_SMISR）. 2608
63.6.23 RTC ステータスクリアレジスタ（RTC_SCR）. 2609
63.6.24 RTC アラーム A バイナリモードレジスタ（RTC_ALRABINR）. 2610
63.6.25 RTC アラーム B バイナリモードレジスタ（RTC_ALRBBINR）. 2611
63.6.26 RTC レジスタマップ 2612

64 タンパおよびバックアップレジスタ（TAMP）. 2614

64.1 概要 2614
64.2 TAMP の主な機能 2614
64.3 TAMP の実装 2615
64.4 TAMP の機能説明 2616
64.4.1 TAMP ブロック図 2616
64.4.2 TAMP ピンおよび内部信号 2617
64.4.3 RTC および TAMP によって制御される GPIO 2619
64.4.4 TAMP レジスタ書込み保護 2619
64.4.5 TAMP セキュア保護モード 2620
64.4.6 バックアップレジスタ保護ゾーン 2620
64.4.7 TAMP 特権保護モード 2621
64.4.8 ブート･ハードウェア･キー（BHK）. 2621
64.4.9 タンパ検出 2622
64.4.10 TAMP バックアップレジスタとその他のデバイス機密情報の消去 2622
64.4.11 タンパ検出設定と初期化 2624
64.5 TAMP 低消費電力モード 2629
64.6 TAMP 割込み 2629
64.7 TAMP レジスタ 2629
64.7.1 TAMP 制御レジスタ 1（TAMP_CR1）. 2630
64.7.2 TAMP 制御レジスタ 2（TAMP_CR2）. 2632

64.7.3 TAMP 制御レジスタ 3（TAMP_CR3）. 2634
64.7.4 TAMP フィルタ制御レジスタ（TAMP_FLTCR）. 2636
64.7.5 TAMP アクティブタンパ制御レジスタ 1（TAMP_ATCR1）. 2637
64.7.6 TAMP アクティブタンパシードレジスタ（TAMP_ATSEEDR）. 2639
64.7.7 TAMP アクティブタンパ出力レジスタ（TAMP_ATOR）. 2640
64.7.8 TAMP アクティブタンパ制御レジスタ 2（TAMP_ATCR2）. 2641
64.7.9 TAMP セキュア設定レジスタ（TAMP_SECCFGR）. 2644
64.7.10 TAMP 特権設定レジスタ（TAMP_PRIVCFGR）. 2646
64.7.11 TAMP 割込み有効レジスタ（TAMP_IER）. 2647
64.7.12 TAMP ステータスレジスタ（TAMP_SR）. 2649
64.7.13 TAMP 非セキュアマスク済み割込みステータスレジスタ（TAMP_MISR）. . .2651
64.7.14 TAMP セキュアマスク済み割込みステータスレジスタ（TAMP_SMISR）. . .2652
64.7.15 TAMP ステータスクリアレジスタ（TAMP_SCR）. 2654
64.7.16 TAMP 単調カウンタ 1 レジスタ（TAMP_COUNT1R）. 2656
64.7.17 TAMP 消去設定レジスタ（TAMP_ERCFGR）. 2656
64.7.18 TAMP バックアップ x レジスタ（TAMP_BKPxR）. 2657
64.7.19 TAMP レジスタマップ 2658

65 I²C（Inter-integrated circuit）インタフェース 2660

65.1 はじめに 2660
65.2 I²C の主な機能 2660
65.3 I²C の実装 2661
65.4 I²C の機能詳細 2662
65.4.1 I²C ブロック図 2663
65.4.2 I²C ピンおよび内部信号 2664
65.4.3 I²C クロックの要件 2665
65.4.4 モード選択 2666
65.4.5 I²C の初期化 2667
65.4.6 ソフトウェアリセット 2671
65.4.7 データ転送 2671
65.4.8 I²C スレーブモード 2673
65.4.9 I²C マスタモード 2682
65.4.10 I2C_TIMINGR レジスタの設定例 2694
65.4.11 SMBus 固有の機能 2695
65.4.12 SMBus 初期化 2697
65.4.13 SMBus：I2C_TIMEOUTR レジスタの設定例 2699
65.4.14 SMBus スレーブモード 2700
65.4.15 自律モード 2706

65.4.16 エラー条件 2708
65.4.17 DMA リクエスト 2709
65.4.18 デバッグモード 2710
65.5 I2C 低消費電力モード 2711
65.6 I2C 割込み 2712
65.7 I2C レジスタ 2713
65.7.1 I2C 制御レジスタ 1（I2C_CR1） 2713
65.7.2 I2C 制御レジスタ 2（I2C_CR2） 2716
65.7.3 I2C Own Address 1 レジスタ（I2C_OAR1） 2718
65.7.4 I2C Own Address 2 レジスタ（I2C_OAR2） 2719
65.7.5 I2C タイミングレジスタ（I2C_TIMINGR） 2720
65.7.6 I2C タイムアウトレジスタ（I2C_TIMEOUTR） 2721
65.7.7 I2C 割込みおよびステータスレジスタ（I2C_ISR） 2722
65.7.8 I2C 割込みクリアレジスタ（I2C_ICR） 2724
65.7.9 I2C PEC レジスタ（I2C_PECR） 2725
65.7.10 I2C 受信データレジスタ（I2C_RXDR） 2726
65.7.11 I2C 送信データレジスタ（I2C_TXDR） 2726
65.7.12 I2C 自律モード制御レジスタ（I2C_AUTOCR） 2727
65.7.13 I2C レジスタマップ 2728

66 Universal synchronous/asynchronous receiver transmitter (USART/UART) 2730

66.1 概要 2730
66.2 USART の主な機能 2730
66.3 USART の拡張機能 2731
66.4 USART の実装 2731
66.5 USART の機能説明 2733
66.5.1 USART ブロック図 2733
66.5.2 USART ピンおよび内部信号 2733
66.5.3 USART クロック 2736
66.5.4 USART キャラクタの説明 2736
66.5.5 USART の FIFO と閾値 2738
66.5.6 USART トランスミッタ 2738
66.5.7 USART レシーバ 2742
66.5.8 USART ボーレート生成 2748
66.5.9 クロック偏差に対する USART レシーバの許容誤差 2750
66.5.10 USART 自動ボーレート検出 2751

66.5.11 USART マルチプロセッサ通信 2752
66.5.12 USART Modbus 通信 2754
66.5.13 USART パリティ制御 2755
66.5.14 USART LIN（Local Interconnection Network）モード 2756
66.5.15 USART 同期モード 2758
66.5.16 USART 単線半二重通信 2762
66.5.17 USART レシーバタイムアウト 2762
66.5.18 USART スマートカードモード 2763
66.5.19 USART IrDA SIR ENDEC ブロック 2767
66.5.20 USART および DMA を使用した連続通信 2769
66.5.21 RS232 ハードウェアフロー制御および RS485 ドライバ有効 2772
66.5.22 USART 自律モード 2774
66.6 低消費電力モードの USART 2775
66.7 USART 割込み 2776
66.8 USART レジスタ 2778
66.8.1 USART 制御レジスタ 1（USART_CR1） 2778
66.8.2 USART 制御レジスタ 1[オルタネート]（USART_CR1） 2782
66.8.3 USART 制御レジスタ 2（USART_CR2） 2786
66.8.4 USART 制御レジスタ 3（USART_CR3） 2790
66.8.5 USART ボーレートレジスタ（USART_BRR） 2794
66.8.6 USART ガード時間およびプリスケーラレジスタ（USART_GTPR） 2794
66.8.7 USART レシーバタイムアウトレジスタ（USART_RTOR） 2795
66.8.8 USART リクエストレジスタ（USART_RQR） 2796
66.8.9 USART 割込みおよびステータスレジスタ（USART_ISR） 2797
66.8.10 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR） 2803
66.8.11 USART 割込みフラグクリアレジスタ（USART_ICR） 2808
66.8.12 USART レシーバデータレジスタ（USART_RDR） 2809
66.8.13 USART トランスミッタデータレジスタ（USART_TDR） 2809
66.8.14 USART プリスケーラレジスタ（USART_PRESC） 2810
66.8.15 USART 自律モード制御レジスタ（USART_AUTOCR） 2811
66.8.16 USART レジスタマップ 2812

67 低消費電力ユニバーサル非同期レシーバトランスミッタ（LPUART） 2814
67.1 概要 2814
67.2 LPUART の主な機能 2814
67.3 LPUART の実装 2815
67.4 LPUART 機能の詳細 2817
67.4.1 LPUART ブロック図 2817
67.4.2 LPUART ピンおよび内部信号 2818
67.4.3 LPUART クロック 2819
67.4.4 LPUART キャラクタの説明 2820
67.4.5 LPUART の FIFO と閾値 2822
67.4.6 LPUART トランスミッタ 2822
67.4.7 LPUART レシーバ 2825
67.4.8 LPUART ボーレート生成 2829
67.4.9 クロック偏差に対する LPUART レシーバの許容誤差 2831
67.4.10 LPUART マルチプロセッサ通信 2832
67.4.11 LPUART パリティ制御 2834
67.4.12 LPUART 単線半二重通信 2835
67.4.13 DMA および LPUART を使用した連続通信 2835
67.4.14 RS232 ハードウェアフロー制御および RS485 ドライバ有効 2838
67.4.15 LPUART 自律モード 2840
67.5 低消費電力モードの LPUART 2841
67.6 LPUART 割込み 2842
67.7 LPUART レジスタ 2843
67.7.1 LPUART 制御レジスタ 1（LPUART_CR1） 2843
67.7.2 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1） 2846
67.7.3 LPUART 制御レジスタ 2（LPUART_CR2） 2849
67.7.4 LPUART 制御レジスタ 3（LPUART_CR3） 2851
67.7.5 LPUART ボーレートレジスタ（LPUART_BRR） 2853
67.7.6 LPUART リクエストレジスタ（LPUART_RQR） 2854
67.7.7 LPUART 割込みおよびステータスレジスタ（LPUART_ISR） 2855
67.7.8 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR） . . . 2859
67.7.9 LPUART 割込みフラグクリアレジスタ（LPUART_ICR） 2862
67.7.10 LPUART 受信データレジスタ（LPUART_RDR） 2863
67.7.11 LPUART 送信データレジスタ（LPUART_TDR） 2863
67.7.12 LPUART プリスケーラレジスタ（LPUART_PRESC） 2864
67.7.13 LPUART 自律モード制御レジスタ（LPUART_AUTOCR） 2865
67.7.14 LPUART レジスタマップ 2866

68 シリアルペリフェラルインタフェース（SPI） 2868
68.1 概要 2868
68.2 SPI の主な機能 2868
68.3 SPI の実装 2869
68.4 SPI の機能詳細 2870
68.4.1 SPI のブロック図 2870
68.4.2 SPI ピンおよび内部信号 2871
68.4.3 SPI 通信の一般的側面 2873
68.4.4 マスタとスレーブの 1 対 1 の通信 2873
68.4.5 標準のマルチスレーブ通信 2876
68.4.6 マルチマスタ通信 2877
68.4.7 スレーブ選択（SS）ピンの管理 2877
68.4.8 レディピン（RDY）の管理 2881
68.4.9 通信フォーマット 2881
68.4.10 SPI の設定 2883
68.4.11 SPI の有効化 2884
68.4.12 SPI のデータ送受信手順 2884
68.4.13 SPI の無効化 2888
68.4.14 データパッキング 2890
68.4.15 DMA（ダイレクトメモリアクセス）を使用する通信 2892
68.4.16 自律モード 2893
68.5 SPI の特定モードおよび制御 2894
68.5.1 TI モード 2894
68.5.2 SPI エラーフラグ 2895
68.5.3 CRC 計算 2898
68.6 SPI 低電力モード 2899
68.7 SPI 割込み 2899
68.8 SPI レジスタ 2901
68.8.1 SPI 制御レジスタ 1（SPI_CR1） 2901
68.8.2 SPI 制御レジスタ 2（SPI_CR2） 2903
68.8.3 SPI 設定レジスタ 1（SPI_CFG1） 2903
68.8.4 SPI 設定レジスタ 2（SPI_CFG2） 2906
68.8.5 SPI 割込み有効レジスタ（SPI_IER） 2908
68.8.6 SPI ステータスレジスタ（SPI_SR） 2909
68.8.7 SPI 割込み / ステータスフラグクリアレジスタ（SPI_IFCR） 2912
68.8.8 SPI 自律モード制御レジスタ（SPI_AUTOCR） 2913
68.8.9 SPI 送信データレジスタ（SPI_TXDR） 2914

68.8.10 SPI 受信データレジスタ（SPI_RXDR）. 2914
68.8.11 SPI 多項式レジスタ（SPI_CRCPOLY）. 2915
68.8.12 SPI トランスミッタ CRC レジスタ（SPI_TXCRC）. 2916
68.8.13 SPI レシーバ CRC レジスタ（SPI_RXCRC）. 2917
68.8.14 SPI アンダーランデータレジスタ（SPI_UDRDR）. 2917
68.8.15 SPI レジスタマップ 2918

69 シリアルオーディオインタフェース（SAI）. 2919

69.1 概要 2919
69.2 SAI の主な機能 2919
69.3 SAI の実装 2920
69.4 SAI の機能説明 2921
69.4.1 SAI ブロック図 2921
69.4.2 SAI ピンおよび内部信号 2922
69.4.3 主要な SAI モード 2923
69.4.4 SAI 同期モード 2924
69.4.5 オーディオデータサイズ 2925
69.4.6 フレーム同期 2925
69.4.7 スロットの設定 2928
69.4.8 SAI クロックジェネレータ 2930
69.4.9 内部 FIFO 2933
69.4.10 PDM インタフェース 2935
69.4.11 AC’97 リンクコントローラ 2943
69.4.12 SPDIF 出力 2945
69.4.13 特有の機能 2947
69.4.14 エラーフラグ 2951
69.4.15 SAI の無効化 2954
69.4.16 SAI DMA インタフェース 2954
69.5 SAI 割込み 2955
69.6 SAI レジスタ 2956
69.6.1 SAI グローバル設定レジスタ（SAI_GCR）. 2956
69.6.2 SAI 設定レジスタ 1（SAI_ACR1）. 2957
69.6.3 SAI 設定レジスタ 1（SAI_BCR1）. 2960
69.6.4 SAI 設定レジスタ 2（SAI_ACR2）. 2963
69.6.5 SAI 設定レジスタ 2（SAI_BCR2）. 2965
69.6.6 SAI フレーム設定レジスタ（SAI_AFRCR）. 2967
69.6.7 SAI フレーム設定レジスタ（SAI_BFRCR）. 2968

69.6.8 SAI スロットレジスタ（SAI_ASLOTR）. 2969
69.6.9 SAI スロットレジスタ（SAI_BSLOTR）. 2970
69.6.10 SAI 割込みマスクレジスタ（SAI_AIM）. 2971
69.6.11 SAI 割込みマスクレジスタ（SAI_BIM）. 2973
69.6.12 SAI ステータスレジスタ（SAI_ASR）. 2974
69.6.13 SAI ステータスレジスタ（SAI_BSR）. 2976
69.6.14 SAI クリアフラグレジスタ（SAI_ACLRFR）. 2978
69.6.15 SAI クリアフラグレジスタ（SAI_BCLRFR）. 2979
69.6.16 SAI データレジスタ（SAI_ADR）. 2980
69.6.17 SAI データレジスタ（SAI_BDR）. 2980
69.6.18 SAI PDM 制御レジスタ（SAI_PDMCR）. 2981
69.6.19 SAI PDM ディレイレジスタ（SAI_PDMDLY）. 2982
69.6.20 SAI レジスタマップ. 2985

70 FD コントローラエリアネットワーク（FDCAN）. 2987

70.1 概要. 2987
70.2 FDCAN の主な機能. 2989
70.3 FDCAN の機能詳細. 2990
70.3.1 ビットタイミング. 2991
70.3.2 動作モード. 2992
70.3.3 メッセージ RAM. 3000
70.3.4 FIFO 確認応答の処理. 3008
70.3.5 FDCAN Rx FIFO 要素. 3009
70.3.6 FDCAN の Tx バッファ要素. 3011
70.3.7 FDCAN Tx イベント FIFO 要素. 3013
70.3.8 FDCAN 標準メッセージ ID フィルタ要素. 3014
70.3.9 FDCAN 拡張メッセージ ID フィルタ要素. 3015
70.4 FDCAN レジスタ. 3016
70.4.1 FDCAN コア開放レジスタ（FDCAN_CREL）. 3016
70.4.2 FDCAN エンディアンレジスタ（FDCAN_ENDN）. 3016
70.4.3 FDCAN データビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP）. 3017
70.4.4 FDCAN テストレジスタ（FDCAN_TEST）. 3018
70.4.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD）. 3019
70.4.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）. 3019
70.4.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）. 3021
70.4.8 FDCAN タイムスタンプカウンタ設定レジスタ（FDCAN_TSCC）. 3022

70.4.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV）.....3023
70.4.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）.....3023
70.4.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）.....3024
70.4.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR）.....3025
70.4.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）.....3025
70.4.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）.....3028
70.4.15 FDCAN 割込みレジスタ（FDCAN_IR）.....3028
70.4.16 FDCAN 割込み有効レジスタ（FDCAN_IE）.....3031
70.4.17 FDCAN 割込みライン選択レジスタ（FDCAN_ILS）.....3033
70.4.18 FDCAN 割込みライン有効レジスタ（FDCAN_ILE）.....3034
70.4.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_RXGFC）.....3034
70.4.20 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）.....3036
70.4.21 FDCAN 高優先メッセージステータスレジスタ（FDCAN_HPMS）.....3036
70.4.22 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）.....3037
70.4.23 CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）.....3038
70.4.24 FDCAN Rx FIFO 1 ステータスレジスタ（FDCAN_RXF1S）.....3038
70.4.25 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）.....3039
70.4.26 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）.....3039
70.4.27 FDCAN Tx FIFO ／キューステータスレジスタ（FDCAN_TXFQS）.....3040
70.4.28 FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）.....3040
70.4.29 FDCAN Tx バッファ追加リクエストレジスタ（FDCAN_TXBAR）.....3041
70.4.30 FDCAN Tx バッファキャンセルリクエストレジスタ（FDCAN_TXBCR）.....3042
70.4.31 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）.....3042
70.4.32 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）.....3043
70.4.33 FDCAN Tx バッファ送信割込み有効レジスタ（FDCAN_TXBTIE）.....3043
70.4.34 FDCAN Tx バッファ･キャンセル終了割込みイネーブル･レジスタ（FDCAN_TXBCIE）.....3044
70.4.35 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）.....3044
70.4.36 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）.....3045
70.4.37 FDCAN CFG クロック分周回路レジスタ（FDCAN_CKDIV）.....3046
70.4.38 FDCAN レジスタマップ 3047

71 ユニバーサルシリアルバスフルスピードホスト / デバイスインタフェース（USB） 3050

71.1 概要 3050

71.2 USB の主な機能 3050

71.3 USB の実装 3050

71.4 USB の機能説明 3051

71.4.1 デバイスモードとホストモードの両方で使用される USB ブロックの説明 . . . 3053

71.4.2 ホストモードに固有のホストフレームスケジューラ（HFS）の説明 3054

71.5 デバイスモードおよびホストモードでのプログラミングの考慮事項 3055

71.5.1 汎用 USB デバイスのプログラミング 3055

71.5.2 システムリセットとパワーオンリセット 3055

71.5.3 ダブルバッファエンドポイントとデバイスモードでの使用 3061

71.5.4 ダブルバッファチャネル：ホストモードでの使用 3064

71.5.5 デバイスモードでのアイソクロナス転送 3065

71.5.6 ホストモードでのアイソクロナス転送 3066

71.5.7 サスペンド / レジュームイベント 3067

71.6 USB および USB SRAM レジスタ 3070

71.6.1 共通レジスタ 3070

71.6.2 バッファディスクリプタテーブル 3088

71.6.3 USB レジスタマップ 3093

72 USB On-The-Go フルスピード （OTG_FS） 3095

72.1 概要 3095

72.2 OTG_FS の主な機能 3096

72.2.1 標準機能 3096

72.2.2 ホストモード機能 3097

72.2.3 ペリフェラルモード機能 3097

72.3 OTG_FS の実装 3098

72.4 OTG_FS の機能説明 3099

72.4.1 OTG_FS のブロック図 3099

72.4.2 OTG_FS ピン および内部信号 3100

72.4.3 OTG_FS コア 3100

72.4.4 OTG-FS に接続された内蔵フルスピード OTG PHY 3101

72.4.5 OTG の検出 3101

72.5 OTG_FS デュアルロールデバイス（DRD） 3102

72.5.1 ID ラインの検出 3102

72.5.2 HNP デュアルロールデバイス 3102

72.5.3 SRP デュアルロールデバイス 3103
72.6 USB ペリフェラルとしての OTG_FS 3103
72.6.1 SRP 対応ペリフェラル 3104
72.6.2 ペリフェラルの状態 3104
72.6.3 ペリフェラルエンドポイント 3105
72.7 USB ホストとしての OTG_FS 3107
72.7.1 SRP 対応ホスト 3108
72.7.2 USB ホストの状態 3108
72.7.3 ホストチャネル 3110
72.7.4 ホストスケジューラ 3111
72.8 OTG_FS SOF トリガ 3112
72.8.1 ホストの SOF 3112
72.8.2 ペリフェラルの SOF 3112
72.9 OTG_FS 低電力モード 3113
72.10 OTG_FS OTG_HFIR レジスタの動的更新 3114
72.11 OTG_FS データ FIFO 3114
72.11.1 ペリフェラル FIFO アーキテクチャ 3115
72.11.2 ホスト FIFO アーキテクチャ 3116
72.11.3 FIFO RAM の割り当て 3117
72.12 OTG_FS システムのパフォーマンス 3119
72.13 OTG_FS 割込み 3119
72.14 OTG_FS 制御／ステータスレジスタ 3121
72.14.1 CSR メモリマップ 3121
72.15 OTG_FS レジスタ 3125
72.15.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL）. 3126
72.15.2 OTG 割込みレジスタ（OTG_GOTGINT）. 3129
72.15.3 OTG AHB 設定レジスタ（OTG_GAHBCFG）. 3130
72.15.4 OTG USB 設定レジスタ（OTG_GUSBCFG）. 3131
72.15.5 OTG リセットレジスタ（OTG_GRSTCTL）. 3133
72.15.6 OTG コア割込みレジスタ（OTG_GINTSTS）. 3135
72.15.7 OTG 割込みマスクレジスタ（OTG_GINTMSK）. 3139
72.15.8 OTG 受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR）. 3142
72.15.9 OTG 受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR）. 3143
72.15.10 OTG ステータス読出しおよびポップレジスタ（OTG_GRXSTSP）. 3144
72.15.11 OTG ステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）3145

72.15.12 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）.....3146
72.15.13 OTG ホスト非周期的送信 FIFO サイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）.....3146
72.15.14 OTG 非周期的送信 FIFO ／キューステータスレジスタ（OTG_HNPTXSTS）.....3147
72.15.15 OTG 一般コア設定レジスタ（OTG_GCCFG）.....3148
72.15.16 OTG コア ID レジスタ（OTG_CID）.....3149
72.15.17 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）.....3150
72.15.18 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）.....3154
72.15.19 OTG デバイス IN エンドポイント送信 FIFO x サイズレジスタ（OTG_DIEPTXFx）.....3154
72.15.20 ホストモードレジスタ.....3155
72.15.21 OTG ホスト設定レジスタ（OTG_HCFG）.....3155
72.15.22 OTG ホストフレームインターバルレジスタ（OTG_HFIR）.....3156
72.15.23 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）...3156
72.15.24 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）.....3157
72.15.25 OTG ホスト全チャネル割込みレジスタ（OTG_HAINT）.....3158
72.15.26 OTG ホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）.....3159
72.15.27 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）.....3159
72.15.28 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）.....3162
72.15.29 OTG ホストチャネル x 割込みレジスタ（OTG_HCINTx）.....3163
72.15.30 OTG ホストチャネル x 割込みマスクレジスタ（OTG_HCINTMSKx）.....3164
72.15.31 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）.....3165
72.15.32 デバイスモードレジスタ.....3166
72.15.33 OTG デバイス設定レジスタ（OTG_DCFG）.....3166
72.15.34 OTG デバイス制御レジスタ（OTG_DCTL）.....3167
72.15.35 OTG デバイスステータスレジスタ（OTG_DSTS）.....3169
72.15.36 OTG デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）.....3170
72.15.37 OTG デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）.....3171
72.15.38 OTG デバイス全エンドポイント割込みレジスタ（OTG_DAINT）.....3172
72.15.39 OTG 全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）.....3173
72.15.40 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）.....3173
72.15.41 OTG デバイス V_{BUS} パルス時間レジスタ（OTG_DVBUSPULSE）.....3174
72.15.42 OTG デバイス IN エンドポイント FIFO エンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）.....3174
72.15.43 OTG デバイス制御 IN エンドポイント 0 制御レジスタ（OTG_DIEPCTL0）.....3175
72.15.44 OTG デバイス IN エンドポイント x 制御レジスタ（OTG_DIEPCTLx）.....3176

72.15.45 OTG デバイス IN エンドポイント x 割込みレジスタ（OTG_DIEPINTx）....3179
72.15.46 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）....3180
72.15.47 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）....3181
72.15.48 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）....3182
72.15.49 OTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）....3183
72.15.50 OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）....3184
72.15.51 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）....3186
72.15.52 OTG デバイス OUT エンドポイント x 制御レジスタ（OTG_DOEPCTLx）....3187
72.15.53 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）....3189
72.15.54 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）....3190
72.15.55 OTG_FS レジスタマップ 3191
72.16 OTG_FS プログラミングモデル 3199
72.16.1 コアの初期化 3199
72.16.2 ホストの初期化 3200
72.16.3 デバイスの初期化 3201
72.16.4 ホストプログラミングモデル 3201
72.16.5 デバイスプログラミングモデル 3222
72.16.6 最悪ケースの応答時間 3242
72.16.7 OTG プログラミングモデル 3243

73 USB On-The-Go ハイスピード（OTG_HS） 3249

73.1 概要 3249
73.2 OTG_HS の主な特徴 3250
73.2.1 標準機能 3250
73.2.2 ホストモード機能 3251
73.2.3 ペリフェラルモード機能 3251
73.3 OTG_HS の実装 3251
73.4 OTG_HS 機能詳細 3252
73.4.1 OTG_HS ブロック図 3252
73.4.2 OTG_HS ピンおよび内部信号 3253
73.4.3 OTG_HS コア 3253
73.4.4 OTG の検出 3254

73.4.5 OTG-FS に接続されたハイスピード OTG PHY 3254
73.5 OTG_HS デュアルロールデバイス（DRD） 3254
73.5.1 ID ラインの検出 3254
73.6 USB ペリフェラルとしての OTG_HS 3255
73.6.1 ペリフェラルの状態 3255
73.6.2 ペリフェラルエンドポイント 3256
73.7 USB ホストとしての OTG_HS 3259
73.7.1 USB ホストの状態 3259
73.7.2 ホストチャネル 3261
73.7.3 ホストスケジューラ 3262
73.8 OTG_HS SOF トリガ 3263
73.8.1 ホストの SOF 3263
73.8.2 ペリフェラルの SOF 3263
73.9 OTG_HS 低消費電力モード 3264
73.10 OTG_HSOTG_HFIR レジスタの動的更新 3265
73.11 OTG_HS データ FIFO 3265
73.11.1 ペリフェラル FIFO アーキテクチャ 3266
73.11.2 ホスト FIFO アーキテクチャ 3267
73.11.3 FIFO RAM の割り当て 3268
73.12 **OTG_HS** 割込み 3270
73.13 **OTG_HS** 制御レジスタおよびステータスレジスタ 3272
73.13.1 CSR メモリマップ 3272
73.14 **OTG_HS** レジスタ 3277
73.14.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL） 3277
73.14.2 OTG 割込みレジスタ（OTG_GOTGINT） 3279
73.14.3 OTG AHB 設定レジスタ（OTG_GAHBCFG） 3280
73.14.4 OTG USB 設定レジスタ（OTG_GUSBCFG） 3281
73.14.5 OTG リセットレジスタ（OTG_GRSTCTL） 3283
73.14.6 OTG コア割込みレジスタ [オルタネート]（OTG_GINTSTS） 3285
73.14.7 OTG コア割込みレジスタ [オルタネート]（OTG_GINTSTS） 3289
73.14.8 OTG 割込みマスクレジスタ [オルタネート]（OTG_GINTMSK） 3294
73.14.9 OTG 割込みマスクレジスタ [オルタネート]（OTG_GINTMSK） 3295
73.14.10 OTG 受信ステータスデバッグ読出しレジスタ [オルタネート]（OTG_GRXSTSR） 3297
73.14.11 OTG 受信ステータスデバッグ読出しレジスタ [オルタネート]（OTG_GRXSTSR） 3298
73.14.12 OTG ステータス読出しおよびポップレジスタ（OTG_GRXSTSP） 3299

73.14.13 OTG ステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）3300
73.14.14 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）. 3301
73.14.15 OTG ホスト非周期的送信 FIFO サイズレジスタ [オルタネート]（OTG_HNPTXFSIZ）. 3302
73.14.16 エンドポイント 0 送信 FIFO サイズ [オルタネート]（OTG_DIEPTXF0）. . .3302
73.14.17 OTG 非周期的送信 FIFO ／キューステータスレジスタ（OTG_HNPTXSTS）. 3303
73.14.18 OTG 一般コア設定レジスタ（OTG_GCCFG）. 3304
73.14.19 OTG コア ID レジスタ（OTG_CID）. 3305
73.14.20 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）. 3306
73.14.21 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）. 3310
73.14.22 OTG デバイス IN エンドポイント送信 FIFO x サイズレジスタ（OTG_DIEPTXFx）. 3310
73.14.23 ホストモードレジスタ 3310
73.14.24 OTG ホスト設定レジスタ（OTG_HCFG）. 3311
73.14.25 OTG ホストフレームインターバルレジスタ（OTG_HFIR）. 3311
73.14.26 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）. . .3312
73.14.27 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）. 3313
73.14.28 OTG ホスト全チャネル割込みレジスタ（OTG_HAINT）. 3314
73.14.29 OTG ホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）. 3314
73.14.30 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）. 3315
73.14.31 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）. 3317
73.14.32 OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）. 3319
73.14.33 OTG ホストチャネル x 割込みレジスタ（OTG_HCINTx）. 3320
73.14.34 OTG ホストチャネル x 割込みマスクレジスタ（OTG_HCINTMSKx）. 3321
73.14.35 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）. 3322
73.14.36 OTG ホストチャネル x DMA アドレスレジスタ（OTG_HCDMAx）. 3323
73.14.37 デバイスモードレジスタ 3323
73.14.38 OTG デバイス設定レジスタ（OTG_DCFG）. 3323
73.14.39 OTG デバイス制御レジスタ（OTG_DCTL）. 3325
73.14.40 OTG デバイスステータスレジスタ（OTG_DSTS）. 3327
73.14.41 OTG デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）. 3328
73.14.42 OTG デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）. 3329
73.14.43 OTG デバイス全エンドポイント割込みレジスタ（OTG_DAINT）. 3330
73.14.44 OTG 全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）. 3331
73.14.45 OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）. 3331

73.14.46 OTG デバイス IN エンドポイント FIFO エンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）. 3332
73.14.47 OTG デバイス IN エンドポイント x 制御レジスタ [オルタネート]（OTG_DIEPCTLx）. 3333
73.14.48 OTG デバイス IN エンドポイント x 制御レジスタ [オルタネート]（OTG_DIEPCTLx）. 3335
73.14.49 OTG デバイス IN エンドポイント x 割込みレジスタ（OTG_DIEPINTx）. . . . 3337
73.14.50 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）. 3338
73.14.51 OTG デバイス IN エンドポイント x DMA アドレスレジスタ（OTG_DIEPDMAx）. 3339
73.14.52 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）. 3339
73.14.53 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）. 3340
73.14.54 OTG デバイスコントロール OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）3341
73.14.55 OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）. 3342
73.14.56 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）. 3344
73.14.57 OTG デバイス OUT エンドポイント x DMA アドレスレジスタ（OTG_DOEPDMAx）. 3345
73.14.58 OTG デバイス OUT エンドポイント x 制御レジスタ [オルタネート]（OTG_DOEPCTLx）. 3345
73.14.59 OTG デバイス OUT エンドポイント x 制御レジスタ [オルタネート]（OTG_DOEPCTLx）. 3348
73.14.60 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）. 3350
73.14.61 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）. 3351
73.14.62 OTG パワーおよびクロックゲーティング制御レジスタ 1（OTG_PCGCCTL1）. 3352
73.14.63 OTG_HS レジスタマップ 3353
73.15 **OTG_HS** プログラミングモデル 3360
73.15.1 コアの初期化 3360
73.15.2 ホストの初期化 3361
73.15.3 デバイスの初期化 3362
73.15.4 DMA モード 3362
73.15.5 ホストプログラミングモデル 3362
73.15.6 デバイスプログラミングモデル 3395
73.15.7 最悪ケースの応答時間 3416
73.15.8 OTG プログラミングモデル 3417

74 USB Type-C®/USB Power Delivery インタフェース（UCPD）. 3418

74.1 概要 3418
74.2 UCPD の主な機能 3418
74.3 UCPD の実装 3419
74.4 UCPD の機能詳細 3419
74.4.1 UCPD ブロック図 3420
74.4.2 UCPD のリセットおよびクロック 3421
74.4.3 物理層のプロトコル 3422
74.4.4 UCPD BMC トランスミッタ 3429
74.4.5 UCPD BMC レシーバ 3430
74.4.6 UCPD Type-C のプルアップ（Rp）およびプルダウン（Rd） 3432
74.4.7 UCPD Type-C 電圧監視とデバウンス 3432
74.4.8 UCPD ファストロールスワップ（FRS） 3433
74.4.9 UCPD DMA インタフェース 3433
74.4.10 STOP モードからのウェイクアップ 3433
74.5 UCPD プログラミングシーケンス 3433
74.5.1 初期化フェーズ 3434
74.5.2 Type-C ステートマシンの処理 3434
74.5.3 USB PD の送信 3436
74.5.4 USB PD の受信 3436
74.5.5 UCPD ソフトウェアのトリミング 3437
74.6 UCPD 低消費電力モード 3438
74.7 UCPD 割込み 3439
74.8 UCPD レジスタ 3440
74.8.1 UCPD 設定レジスタ 1（UCPD_CFGR1） 3440
74.8.2 UCPD 設定レジスタ 2（UCPD_CFGR2） 3442
74.8.3 UCPD 設定レジスタ 3（UCPD_CFGR3） 3443
74.8.4 UCPD 制御レジスタ（UCPD_CR） 3443
74.8.5 UCPD 割込みマスクレジスタ（UCPD_IMR） 3446
74.8.6 UCPD ステータスレジスタ（UCPD_SR） 3447
74.8.7 UCPD 割込みクリアレジスタ（UCPD_ICR） 3450
74.8.8 UCPD Tx 順序集合タイプレジスタ（UCPD_TX_ORDSETR） 3451
74.8.9 UCPD Tx ペイロードサイズレジスタ（UCPD_TX_PAYSZR） 3452
74.8.10 UCPD Tx データレジスタ（UCPD_TXDR） 3452
74.8.11 UCPD Rx 順序集合レジスタ（UCPD_RX_ORDSETR） 3453
74.8.12 UCPD Rx ペイロードサイズレジスタ（UCPD_RX_PAYSZR） 3454
74.8.13 UCPD 受信データレジスタ（UCPD_RXDR） 3454

74.8.14 UCPD Rx オーダード･セット拡張レジスタ 1（UCPD_RX_ORDEXTR1）...3455
74.8.15 UCPD Rx 順序集合拡張レジスタ 2（UCPD_RX_ORDEXTR2）.....3455
74.8.16 UCPD レジスタ･マップ 3456

75 デバッグサポート（DBG） 3458

75.1 DBG の概要 3458
75.2 DBG の機能説明 3459
75.2.1 DBG ブロック図 3459
75.2.2 DBG ピンおよび内部信号 3459
75.2.3 DBG のリセットおよびクロック 3460
75.2.4 DBG 電力ドメイン 3460
75.2.5 デバッグおよび低電力モード 3460
75.2.6 セキュリティ 3460
75.3 シリアルワイヤ／ JTAG デバッグポート（SWJ-DP） 3462
75.3.1 JTAG デバッグポート 3463
75.3.2 シリアルワイヤデバッグポート 3465
75.3.3 デバッグポートレジスタ 3467
75.3.4 デバッグポートレジスタマップ 3473
75.4 アクセスポート 3474
75.4.1 アクセスポートレジスタ 3474
75.4.2 アクセスポートレジスタマップ 3479
75.5 ROM テーブル 3480
75.5.1 MCU ROM テーブルレジスタ 3482
75.5.2 MCU ROM テーブルレジスタマップ 3487
75.5.3 プロセッサ ROM テーブルレジスタ 3488
75.5.4 プロセッサ ROM テーブルレジスタマップ 3493
75.6 データウォッチポイントおよびトレースユニット（DWT） 3494
75.6.1 DWT レジスタ 3495
75.6.2 DWT レジスタマップ 3509
75.7 計装トレースマクロセル（ITM） 3511
75.7.1 ITM レジスタ 3511
75.7.2 ITM レジスタマップ 3520
75.8 ブレークポイントユニット（BPU） 3521
75.8.1 BPU レジスタ 3522
75.8.2 BPU レジスタマップ 3529

75.9 組み込みトレースマクロセル（ETM） 3530
75.9.1 ETM レジスタ 3530
75.9.2 ETM レジスタマップ 3555
75.10 トレースポートインタフェースユニット（TPIU） 3559
75.10.1 TPIU レジスタ 3560
75.10.2 TPIU レジスタマップ 3569
75.11 クロストリガインタフェース（CTI） 3571
75.11.1 CTI レジスタ 3572
75.11.2 CTI レジスタマップ 3583
75.12 マイクロコントローラデバッグユニット（DBGMCU） 3585
75.12.1 デバイス ID 3585
75.12.2 低電力モードエミュレーション 3585
75.12.3 ペリフェラルクロックフリーズ 3585
75.12.4 DBGMCU レジスタ 3588
75.12.5 DBGMCU レジスタマップ 3599
75.13 参照 3601

76 デバイス電子署名 3602
76.1 ユニークデバイス ID レジスタ（96bit） 3602
76.2 Flash サイズデータレジスタ 3603
76.3 パッケージデータレジスタ 3604

77 セキュリティ上の重要通知 3605

78 改版履歴 3606

表の一覧

表 1. STM32U5 シリーズへのマスタの実装 . . . 125
表 2. STM32U5 シリーズへのスレーブの実装 . . . 126
表 3. SAU 設定可能領域に対するメモリマップセキュリティ属性の例 . . . 130
表 4. TZSC によってセキュリティ保護可能なペリフェラル . . . 132
表 5. TrustZone を認識するペリフェラル . . . 134
表 6. メモリマップとペリフェラルのレジスタ境界アドレス . . . 141
表 7. SRAM サイズ . . . 146
表 8. IDAU および SAU によるセキュリティ属性の設定 . . . 153
表 9. MPCWMx リソース . . . 155
表 10. MPCBBx リソース . . . 155
表 11. DMA チャネルの使用（セキュリティ） . . . 159
表 12. ペリフェラルと割り当てられた I/O 間のセキュアオルタネート機能 . . . 162
表 13. セキュア I/O に接続されていない非セキュアペリフェラル機能 . . . 162
表 14. セキュア I/O に接続可能な非セキュアペリフェラル機能 . . . 163
表 15. TrustZone を認識する DBGMCU アクセス管理 . . . 164
表 16. DMA チャネルの使用（特権） . . . 168
表 17. TAMP の内部タンパ . . . 172
表 18. 低消費電力モードが TAMP に与える影響 . . . 173
表 19. 加速された暗号操作 . . . 176
表 20. 主な製品ライフサイクル遷移 . . . 178
表 21. 一般的な製品ライフサイクルの フェーズ . . . 179
表 22. OEM1/2RDP アンロック方法 . . . 181
表 23. RDP によるデバッグ保護 . . . 182
表 24. RDP によるソフトウェア知的財産保護 . . . 184
表 25. TrustZone が無効な場合のブートモード（TZEN = 0） . . . 187
表 26. TrustZone が有効な場合のブートモード（TZEN = 1） . . . 188
表 27. RDP 保護レベルに対するブート領域 . . . 188
表 28. GTZC の機能 . . . 192
表 29. GTZC1 サブブロックのアドレスオフセット . . . 193
表 30. GTZC2 サブブロックのアドレスオフセット . . . 193
表 31. MPCWM リソースの割当 . . . 193
表 32. STM32U535/545 に対する MPCBB リソースの割当 . . . 194
表 33. に対する MPCBB リソースの割当 . . . 194
表 34. に対する MPCBB リソースの割当 . . . 194
表 35. STM32U5Fx/5Gx に対する MPCBB リソースの割当 . . . 194
表 36. サブ領域 A および B のセキュア属性 . . . 197
表 37. サブ領域 A および B の特権属性 . . . 197
表 38. GTZC 割込みリクエスト . . . 199
表 39. GTZC1 TZSC レジスタのマップとリセット値 . . . 218
表 40. GTZC1 TZIC レジスタマップとリセット値 . . . 248
表 41. GTZC1 MPCBBz レジスタマップおよびリセット値（z = 1、2、3、5、6） . . . 253
表 42. GTZC2 TZSC レジスタマップとリセット値 . . . 257
表 43. GTZC2 TZIC レジスタのマップとリセット値 . . . 266
表 44. GTZC2 MPCBB4 レジスタマップとリセット値 . . . 269
表 45. SRAM の構造 . . . 270
表 46. 内部 SRAM 機能 . . . 271
表 47. HCLK 周波数と電圧スケーリング範囲に対するウェイトステートの数 . . . 274
表 48. 低消費電力モードが RAMCFG に与える影響 . . . 275
表 49. RAMCFG 割込みリクエスト . . . 275

表 50. RAMCFG レジスタのマップとリセット値 282
表 51. STM32U535/545 向けの Flash モジュール 512KB デュアルバンク構成 286
表 52. STM32U575/585 向けの Flash モジュール 2MB デュアルバンク構成 287
表 53. STM32U59x/5Ax/5Fx/5Gx 向けの Flash モジュール 4MB デュアルバンク構成 287
表 54. CPU クロック（HCLK）周波数によるウェイトステート数（LPM=0） 289
表 55. CPU クロック（HCLK）周波数によるウェイトステート数（LPM = 1） 289
表 56. システムリセットによる Flash 操作の中断 298
表 57. ユーザオプションバイト構成マッピング 299
表 58. TZEN 有効化後のデフォルトのセキュアオプションバイト 302
表 59. ウォーターマークベースセキュア領域 303
表 60. セキュア非表示保護 304
表 61. セキュアおよび HDP 保護 304
表 62. Flash のセキュリティ状態 305
表 63. WRP 保護 309
表 64. Flash メモリ読出し保護ステータス（TZEN = 0） 310
表 65. アクセス状態に対する保護レベルと実行モード（TZEN = 0 の場合） 311
表 66. Flash メモリ読出し保護ステータス（TZEN = 1） 312
表 67. アクセス状態に対する保護レベルと実行モード（TZEN = 1 の場合） 313
表 68. TrustZone が有効（TZEN = 1）な場合の Flash メモリアクセスに対する RDP レベル 318
表 69. TrustZone が無効（TZEN = 0）な場合の Flash メモリアクセスに対する RDP レベル 319
表 70. TrustZone が有効（TZEN = 1）な場合の Flash メモリ全体消去に対する RDP レベル 319
表 71. Flash システムメモリの OTP および RSS アクセス 319
表 72. Flash レジスタへのアクセス 320
表 73. Flash ページアクセスに対する特権モード 320
表 74. Flash 全体消去に対する特権モード 320
表 75. TrustZone が有効（TZEN = 1）な場合の SECyBBRx レジスタへのアクセス 321
表 76. TrustZone が有効（TZEN = 1）な場合の PRIVyBBRx レジスタへのアクセス 321
表 77. TrustZone が無効（TZEN = 0）な場合の PRIVyBBRx レジスタへのアクセス 321
表 78. Flash 割込みリクエスト 322
表 79. Flash レジスタマップとリセット値 354
表 80. STM32U535/545/575/585 の ICACHE 機能 359
表 81. STM32U59x/5Ax/5Fx/5Gx の ICACHE 機能 359
表 82. n ウェイセットアソシアティブ動作モード（デフォルト）の TAG メモリディメンショニングパラメータ 362
表 83. ダイレクトマップキャッシュモードの TAG メモリディメンショニングパラメータ 363
表 84. AHB トランザクションの ICACHE のキャッシュ可能性 364
表 85. メモリ設定 364
表 86. ICACHE 再配置領域のサイズ、ベースアドレス、および再配置アドレス 366
表 87. ICACHE 割込み 369
表 88. ICACHE レジスタマップとリセット値 374
表 89. STM32U535/545/575/585 の DCACHE 機能 377
表 90. STM32U59x/5Ax/5Fx/5Gx の DCACHE 機能 377
表 91. TAG メモリディメンショニングパラメータ 380
表 92. AHB トランザクションの DCACHE のキャッシュ可能性 382
表 93. DCACHE 割込み 387
表 94. DCACHE レジスタのマップとリセット値 393
表 95. PWR の入出力ピン 396
表 96. PWR 内部入力 / 出力信号 396
表 97. PWR ウェイクアップソース選択 397
表 98. PVM の機能 407
表 99. 低消費電力モードの概要 410
表 100. 動作モードに応じた機能 412

表 101. SLEEP モード 420
表 102. STOP0 モード 421
表 103. STOP1 モード 422
表 104. STOP2 モード 424
表 105. STOP3 モード 427
表 106. STANDBY モード 429
表 107. SHUTDOWN モード 430
表 108. 電力モード出力状態対 MCU 電力モード 432
表 109. PWR セキュリティ設定一覧 433
表 110. PWR 割込みリクエスト 435
表 111. PWR レジスタマップとリセット値 471
表 112. パッケージのピンまたはボールに接続された RCC スレーブ入力／出力信号 475
表 113. 内部 MSIRC ごとの MSIS および MSIK レンジ 482
表 114. バスの最大周波数 489
表 115. クロックソースの最大周波数 489
表 116. 自律型ペリフェラル 495
表 117. RCC セキュリティ設定一覧 497
表 118. 割込みソースとその制御 501
表 119. RCC レジスタマップとリセット値 599
表 120. CRS の機能 606
表 121. STM32U535/545/575/585 用の CRS 内部入力／出力信号 606
表 122. STM32U59x/5Ax/5Fx/5Gx 用の CRS 内部入力／出力信号 606
表 123. 低電力モードが CRS に与える影響 610
表 124. 割込み制御ビット 610
表 125. CRS レジスタマップとリセット値 616
表 126. ポートビット設定 618
表 127. GPIO セキュアビット 627
表 128. GPIO レジスタマップとリセット値 637
表 129. LPGPIO レジスタマップとリセット値 643
表 130. TrustZone セキュリテと特権レジスタアクセス 646
表 131. BOOSTEN および ANASWVDD のセット／リセット 650
表 132. SYSCFG レジスタマップとリセット値 661
表 133. ペリフェラル相互接続マトリックス 664
表 134. GPDMA1 チャネル実装 680
表 135. 低電力モードでの GPDMA1 自律モードとウェイクアップ 681
表 136. プログラムされた GPDMA1 リクエスト 681
表 137. ブロック･リクエストとしてプログラムされた GPDMA1 リクエスト 685
表 138. プログラムされた GPDMA1 トリガ 685
表 139. プログラムされた GPDMA ソース／デスティネーション･バースト 707
表 140. プログラムされたデータ･ハンドリング 712
表 141. 低消費電力モードが GPDMA に与える影響 725
表 142. GPDMA 割込みリクエスト 726
表 143. GPDMA レジスタのマップとリセット値 755
表 144. LPDMA1 チャネル実装 758
表 145. 低電力モードでの LPDMA1 自律モードとウェイクアップ 758
表 146. プログラムされた LPDMA1 リクエスト 759
表 147. ブロック･リクエストとしてプログラムされた LPDMA1 リクエスト 759
表 148. プログラムされた LPDMA1 トリガ 760
表 149. プログラムされた LPDMA ソース／デスティネーション･シングル 779
表 150. プログラムされたデータ･ハンドリング 780
表 151. 低消費電力モードが LPDMA に与える影響 791
表 152. LPDMA 割込みリクエスト 792

表 153. LPDMA レジスタマップとリセット値 810
表 154. DMA2D 内部信号 814
表 155. DMA2D トリガ相互接続 814
表 156. 入力でサポートされているカラーモード 815
表 157. メモリ内のデータ順序 816
表 158. アルファモード設定 817
表 159. サポートされている CLUT カラーモード 818
表 160. システムメモリ内の CLUT データ順序 818
表 161. 出力でサポートされているカラーモード 819
表 162. メモリ内のデータ順序 820
表 163. メモリ内の標準データ順序 821
表 164. 出力 FIFO バイトの順序変更ステップ 822
表 165. DMA2D 割込みリクエスト 827
表 166. DMA2D レジスタマップとリセット値 845
表 167. DMA2D 内部信号 849
表 168. DMA2D トリガ相互接続 849
表 169. 入力でサポートされているカラーモード 850
表 170. メモリ内のデータ順序 852
表 171. アルファモード設定 852
表 172. サポートされている CLUT カラーモード 853
表 173. メモリ内の CLUT データ順序 853
表 174. 出力でサポートされているカラーモード 854
表 175. メモリ内のデータ順序 855
表 176. メモリ内の標準データ順序 855
表 177. 出力 FIFO バイトの順序変更ステップ 856
表 178. メモリ内の MCU 順序 861
表 179. DMA2D 割込みリクエスト 863
表 180. DMA2D レジスタマップとリセット値 881
表 181. GFXMMU の実装 883
表 182. GFXMMU 割込みリクエスト 892
表 183. GFXMMU レジスタのマップとリセット値 901
表 184. STM32U5 シリーズベクタテーブル 903
表 185. EXTI 信号 909
表 186. EVG 信号 909
表 187. EXTI ラインの接続 910
表 188. マスクの機能 913
表 189. レジスタ保護の概要 914
表 190. EXTI レジスタマップセクション 915
表 191. EXTI レジスタマップとリセット値 924
表 192. CRC 内部入力／出力信号 927
表 193. CRC レジスタマップとリセット値 932
表 194. CORDIC 関数 934
表 195. 余弦パラメータ 934
表 196. 正弦パラメータ 935
表 197. 位相パラメータ 935
表 198. モジュラス·パラメータ 936
表 199. 逆正接パラメータ 937
表 200. 双曲線余弦パラメータ 937
表 201. 双曲線正弦パラメータ 938
表 202. 双曲線逆正接パラメータ 938
表 203. 自然対数パラメータ 939
表 204. 自然対数のスケーリングファクタと対応範囲 939

表 205. 平方根パラメータ 939
表 206. 平方根のスケーリング·ファクタと対応範囲 940
表 207. 精度対繰返し回数 943
表 208. CORDIC レジスタのマップとリセット値 950
表 209. 読出し手法と書込み手法の有効な組合せ 964
表 210. FMAC レジスタマップとリセット値 978
表 211. NOR/PSRAM バンク選択 983
表 212. NOR/PSRAMbit 外部メモリアドレス 983
表 213. NAND メモリマッピングおよびタイミングレジスタ 983
表 214. NAND バンク選択 983
表 215. プログラム可能な NOR/PSRAM のアクセスパラメータ 985
表 216. 非マルチプレクス I/O NOR Flash メモリ 985
表 217. 16bit マルチプレクス I/O NOR Flash メモリ 986
表 218. 非マルチプレクス I/O PSRAM/SRAM 986
表 219. 16bit マルチプレクス I/O PSRAM 987
表 220. NOR Flash ／ PSRAM：サポートされているメモリおよびトランザクションの例 987
表 221. FMC_BCRx ビットフィールド（モード 1） 990
表 222. FMC_BTRx ビットフィールド（モード 1） 991
表 223. FMC_BCRx ビットフィールド（モード A） 992
表 224. FMC_BTRx ビットフィールド（モード A） 993
表 225. FMC_BWTRx ビットフィールド（モード A） 993
表 226. FMC_BCRx ビットフィールド（モード 2/B） 995
表 227. FMC_BTRx ビットフィールド（モード 2/B） 996
表 228. FMC_BWTRx ビットフィールド（モード 2/B） 996
表 229. FMC_BCRx ビットフィールド（モード C） 998
表 230. FMC_BTRx ビットフィールド（モード C） 998
表 231. FMC_BWTRx ビットフィールド（モード C） 999
表 232. FMC_BCRx ビットフィールド（モード D） 1000
表 233. FMC_BTRx ビットフィールド（モード D） 1001
表 234. FMC_BWTRx ビットフィールド（モード D） 1001
表 235. FMC_BCRx ビットフィールド（マルチプレクス·モード） 1003
表 236. FMC_BTRx ビットフィールド（マルチプレクス·モード） 1004
表 237. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード） 1009
表 238. FMC_BTRx ビットフィールド（同期マルチプレクス読出しモード） 1009
表 239. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード） 1010
表 240. FMC_BTRx ビットフィールド（同期マルチプレクス書込みモード） 1011
表 241. プログラム可能な NAND Flash のアクセス·パラメータ 1020
表 242. 8bit NAND 型 Flash 1020
表 243. 16bit NAND 型 Flash 1021
表 244. サポートされるメモリおよびトランザクション 1021
表 245. ECC 結果関連ビット 1030
表 246. FMC レジスタマップとリセット値 1031
表 247. STM32U5 シリーズデバイス上のインスタンス 1034
表 248. OCTOSPI/HSPI の実装 1034
表 249. OCTOSPI の入出力ピン
（STM32U535/545 デバイスを除く STM32U5 シリーズ） 1038
表 250. OCTOSPI 入出力ピン（STM32U535/545 デバイスのみ） 1038
表 251. OCTOSPI 内部信号 1038
表 252. コマンド／アドレスフェーズの説明 1047
表 253. アドレスアライメントのケース 1062
表 254. OCTOSPI の割込みリクエスト 1063
表 255. OCTOSPI レジスタマップとリセット値 1087

表 256. OCTOSPIM の実装 1090
表 257. OCTOSPIM の入出力ピン 1091
表 258. OCTOSPIM レジスタマップとリセット値 1096
表 259. STM32U5 シリーズデバイス上のインスタンス 1098
表 260. HSPI の実装 1098
表 261. HSPI の入出力ピン 1102
表 262. HSPI 内部信号 1102
表 263. コマンド／アドレスフェーズの説明 1112
表 264. アドレスアライメントのケース 1128
表 265. HSPI 割込みリクエスト 1130
表 266. HSPI レジスタマップとリセット値 1156
表 267. SDMMC の機能 1159
表 268. SDMMC 動作モード SD および SDIO 1162
表 269. SDMMC 動作モード e•MMC 1162
表 270. SDMMC 内部入力／出力信号 1164
表 271. SDMMC ピン 1164
表 272. SDMMC コマンドおよびデータの位相選択 1165
表 273. コマンドトークンフォーマット 1171
表 274. CRC ありのショートレスポンスのトークンフォーマット 1172
表 275. CRC なしのショートレスポンスのトークンフォーマット 1172
表 276. CRC ありのロングレスポンスのトークンフォーマット 1172
表 277. 固有コマンドの概要 1173
表 278. コマンドパスステータスフラグ 1173
表 279. コマンドパスのエラー処理 1174
表 280. データトークンフォーマット 1181
表 281. データパスステータスフラグとクリアビット 1181
表 282. データパスエラー処理 1182
表 283. データ FIFO アクセス 1183
表 284. 送信 FIFO ステータスフラグ 1184
表 285. 受信 FIFO ステータスフラグ 1185
表 286. AHB と SDMMC_CK のクロック周波数の関係 1189
表 287. SDIO 特殊動作制御 1190
表 288. 4bit モードのスタート、割込み、CRC ステータスの信号検出 1194
表 289. CMD12 の使用事例 1198
表 290. SDMMC 割込み 1212
表 291. レスポンスタイプと SDMMC_RESPxR レジスタ 1219
表 292. SDMMC レジスタマップ 1235
表 293. STM32U5 シリーズ機能 1238
表 294. DLYB 内部入力／出力信号 1239
表 295. 遅延ブロック制御 1240
表 296. DLYB レジスタマップとリセット値 1242
表 297. A/D コンバータの機能 1245
表 298. 温度センサ較正値のメモリ位置 1245
表 299. 内部基準電圧センサの較正値のメモリ位置 1245
表 300. A/D コンバータの各入出力ピン 1247
表 301. ADC 内部入力／出力信号 1247
表 302. ADC1/ADC12 相互接続 1247
表 303. ADC1/ADC12 のレギュラチャネル用外部トリガ 1247
表 304. ADC1/ADC12 のインジェクトチャネルの外部トリガ 1248
表 305. 較正係数のインデックス 1256
表 306. レギュラ外部トリガのトリガ極性の設定 1266
表 307. インジェクト外部トリガのトリガ極性の設定 1266

表 308. TSAR タイミングは分解能に依存 1270
表 309. オフセット計算対データ分解能 1273
表 310. 14bit データフォーマット 1276
表 311. 16bit フォーマットでの数値の例 1277
表 312. アナログウォッチドッグチャネル選択 1286
表 313. アナログウォッチドッグ 1、2、3 の比較 1287
表 314. オーバーサンプリング回路の動作モードの概要 1295
表 315. ADC 割込み 1314
表 316. DELAY ビット対 ADC 分解能 1349
表 317. ADC グローバルレジスタマップ 1352
表 318. ADC レジスタマップと各 ADC のリセット値（マスタ ADC のオフセットは 0x000、スレーブ ADC のオフセットは 0x100）1352
表 319. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット = 0x300 1354
表 320. ADC の主な機能 1356
表 321. 温度センサ較正値のメモリ位置 1357
表 322. 内部基準電圧センサの較正値のメモリ位置 1357
表 323. A/D コンバータの各入出力ピン 1358
表 324. ADC 内部入力／出力信号 1359
表 325. ADC 相互接続 1359
表 326. トリガから変換開始までの遅延 1364
表 327. トリガ極性の設定 1371
表 328. t_{SAR} タイミングは分解能に依存 1372
表 329. アナログウォッチドッグ比較 1384
表 330. アナログウォッチドッグ 1 チャネル選択 1385
表 331. 最大出力結果対 N と M。グレイの値は切り詰めを示す 1389
表 332. 低電力モードが A/D コンバータに与える影響 1395
表 333. ADC のウェイクアップおよび割込みリクエスト 1396
表 334. ADC レジスタマップとリセット値 1418
表 335. DAC の機能 1421
表 336. DAC の入出力ピン 1423
表 337. DAC 内部入力／出力信号 1423
表 338. DAC の相互接続 1424
表 339. データフォーマット（12bit データの場合） 1426
表 340. HFSEL の説明 1427
表 341. サンプルおよびリフレッシュタイミング 1432
表 342. チャネル出力モードの概要 1434
表 343. 低消費電力モードが DAC に与える影響 1440
表 344. DAC 割込み 1441
表 345. DAC レジスタのマップとリセット値 1458
表 346. VREFBUF の標準値 1460
表 347. VREF バッファモード 1461
表 348. VREFBUF レジスタマップとリセット値 1463
表 349. COMP 機能 1464
表 350. COMP1 非反転入力割当て 1465
表 351. COMP1 反転入力割当て 1466
表 352. COMP2 非反転入力割当て 1466
表 353. COMP2 反転入力割当て 1466
表 354. COMP1 出力ブランキング PWM 割当て 1466
表 355. COMP2 出力ブランキング PWM 割当て 1467
表 356. 低消費電力モードでのコンパレータの動作 1470
表 357. 割込み制御ビット 1471

表 358. COMP レジスタマップとリセット値 1474
表 359. オペアンプの可能な接続 1476
表 360. 動作モードと較正 1481
表 361. 低消費電力モードが OPAMP に与える影響 1482
表 362. OPAMP レジスタマップとリセット値 1488
表 363. ADF/MDF の機能 1491
表 364. MDF 外部ピン 1493
表 365. MDF 内部信号 1493
表 366. MDF トリガ接続 1494
表 367. MDF ブレーク接続 1494
表 368. MDF ADC データ接続 1494
表 369. 共通クロック生成の制御 1502
表 370. 受信ストリームに関するクロックの制約 1503
表 371. CIC の次数と CIC のデシメーション値に従ったデータ･サイズ 1510
表 372. 最大デシメーション比と次数、入力データ･サイズ 1511
表 373. 使用可能なゲイン値 1512
表 374. 推奨の最大ゲイン値と CIC のデシメーション比 1514
表 375. 一般的なマイクロフォンの設定 1514
表 376. HPF 3dB 遮断周波数の例 1516
表 377. レジスタ保護のまとめ 1537
表 378. 低消費電力モードが MDF に与える影響 1538
表 379. MDF 割込みリクエスト 1540
表 380. マイクロ･フォンキャプチャ用の MDF 設定の例 1541
表 381. プログラミング･シーケンス 1541
表 382. 出力信号レベル 1548
表 383. MDF レジスタマップとリセット値 1573
表 384. ADF/MDF の機能 1576
表 385. ADF 外部ピン 1577
表 386. ADF 内部信号 1577
表 387. ADF トリガ接続 1578
表 388. 共通クロック生成の制御 1584
表 389. 受信ストリームに関するクロックの制約 1585
表 390. CIC の次数と CIC のデシメーション値に従ったデータ･サイズ 1590
表 391. 使用可能なゲイン値 1591
表 392. 推奨の最大ゲイン値と CIC のデシメーション比 1593
表 393. 一般的なマイクロフォンの設定 1594
表 394. HPF 3dB 遮断周波数の例 1596
表 395. FRSIZE とサンプリング･レートに対する ANSLP 値 1609
表 396. SNTHR に従った閾値 1610
表 397. レジスタ保護のまとめ 1616
表 398. 低消費電力モードが ADF に与える影響 1617
表 399. ADF 割込みリクエスト 1619
表 400. マイクロ･フォンキャプチャ用の ADF 設定の例 1620
表 401. プログラミング･シーケンス（CIC4） 1620
表 402. プログラミング･シーケンス（CIC5） 1622
表 403. 出力信号レベル 1627
表 404. ADF レジスタのマップとリセット値 1650
表 405. DCMI の入出力ピン 1654
表 406. DCMI 内部入力／出力信号 1654
表 407. 32bit ワード（8bit 幅）でキャプチャされたデータバイトの配置 1656
表 408. 32bit ワード（10bit 幅）でキャプチャされたデータバイトの配置 1656
表 409. 32bit ワード（12bit 幅）でキャプチャされたデータバイトの配置 1656

表 410. 32bit ワード（14bit 幅）でキャプチャされたデータバイトの配置 1657
表 411. モノクロプログレッシブビデオフォーマットでのデータ保存 1662
表 412. RGB プログレッシブビデオフォーマットでのデータ保存 1663
表 413. YCbCr プログレッシブビデオフォーマットでのデータ保存 1663
表 414. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード 1663
表 415. DCMI 割込み 1664
表 416. DCMI レジスタマップとリセット値 1676
表 417. PSSI の入出力ピン 1679
表 418. PSSI 内部入力／出力信号 1679
表 419. 32bit ワード（8bit 幅）でキャプチャされたデータバイトの配置 1680
表 420. 32bit ワード（16bit 幅）でキャプチャされたデータバイトの配置 1681
表 421. PSSI 割込みリクエスト 1684
表 422. PSSI レジスタマップとリセット値 1691
表 423. LTDC 外部ピン 1693
表 424. LTDC 内部信号 1694
表 425. LTDC トリガの相互接続 1694
表 426. 各レジスタのクロックドメイン 1694
表 427. ピクセルデータマッピングとカラーフォーマット 1699
表 428. LTDC 割込みリクエスト 1702
表 429. LTDC レジスタマップとリセット値 1720
表 430. DSI ピン 1726
表 431. DSI 内部入力／出力信号 1726
表 432. LTDC インタフェースのカラー･コンポーネントの位置 1729
表 433. 各データ型におけるピクセル単位のペイロードサイズに対する倍数 1730
表 434. 競合検出タイムアウトカウンタの設定 1742
表 435. PRESP_TO カウンタのさまざまなカテゴリのイベントの一覧 1743
表 436. PRESP_TO カウンタの設定 1746
表 437. フレーム要件設定レジスタ 1757
表 438. RGB の成分 1759
表 439. スルーレートおよび遅延の調整 1761
表 440. レーンのカスタム設定 1761
表 441. 帯域周波数（MHz）に対する HS2LP および LP2HS の値 1761
表 442. DSI ラッパー割込みリクエスト 1763
表 443. エラーの原因と回復方法 1765
表 444. DSI レジスタマップとリセット値 1835
表 445. GPU2D の実装 1842
表 446. GPU2D 内部入力／出力信号 1843
表 447. GPU2D トリガ接続 1843
表 448. JPEG 内部信号 1845
表 449. JPEG トリガ接続 1846
表 450. JPEG コーデック割込み 1851
表 451. JPEG コーデックレジスタマップとリセット値 1864
表 452. 取得シーケンスの概要 1869
表 453. スペクトル拡散偏差と AHB クロック周波数 1871
表 454. モードおよび IODEF ビットの値に応じた I/O の状態 1872
表 455. 低電力モードが TSC に与える影響 1874
表 456. 割込み制御ビット 1874
表 457. TSC レジスタマップとリセット値 1882
表 458. RNG 内部入力／出力信号 1885
表 459. RNG 割込みリクエスト 1893
表 460. RNG の設定 1894
表 461. RNG レジスタマップとリセット値 1899

表 462. AES/SAES の機能 1901
表 463. AES 内部入力／出力信号 1902
表 464. CTR モード初期化ベクタの定義 1917
表 465. GCM 最終ブロックの定義 1919
表 466. GCM モードでの AES_IVRx レジスタの初期化 1920
表 467. CCM モードでの AES_IVRx レジスタの初期化 1927
表 468. AES_KEYRx レジスタでのキーエンディアン（128bit または 256bit のキー長） 1933
表 469. AES 割込みリクエスト 1937
表 470. ECB、CBC および CTR での処理遅延時間 1937
表 471. GCM および CCM での処理遅延時間（クロックサイクル数） 1937
表 472. AES レジスタマップとリセット値 1951
表 473. AES/SAES の機能 1954
表 474. SAES 内部入力／出力信号 1955
表 475. SAES_KEYRx レジスタでのキーのエンディアン設定（128bit または 256bit のキー長） 1974
表 476. SAES 割込みリクエスト 1979
表 477. ECB、CBC の処理遅延時間 1980
表 478. SAES レジスタマップとリセット値 1994
表 479. HASH 内部入力／出力信号 1997
表 480. ハッシュプロセッサ出力 2001
表 481. HASH 割込みリクエスト 2007
表 482. 処理時間（クロックサイクル数） 2007
表 483. HASH レジスタのマップとリセット値 2017
表 484. OTFDEC 内部入力／出力信号 2020
表 485. OTFDEC 割込みリクエスト 2024
表 486. OTFDEC レジスタマップとリセット値 2039
表 487. 内部入力／出力信号 2043
表 488. PKA 整数算術機能リスト 2044
表 489. PKA 素体（Fp）楕円曲線機能リスト 2045
表 490. モンゴメリパラメータ計算 2050
表 491. 剰余加算 2051
表 492. 剰余減算 2051
表 493. モンゴメリ乗算 2052
表 494. べき剰余（通常モード） 2053
表 495. べき剰余（高速モード） 2053
表 496. べき剰余（保護モード） 2054
表 497. モジュラ逆数 2054
表 498. モジュラリダクション 2055
表 499. 算術加算 2055
表 500. 算術減算 2055
表 501. 算術乗算 2056
表 502. 算術比較 2056
表 503. CRT べき乗 2057
表 504. 楕円曲線 Fp 上の点の確認 2057
表 505. ECC Fp スカラー乗算 2058
表 506. ECDSA 署名 - 入力 2059
表 507. ECDSA 署名 - 出力 2059
表 508. 拡張 ECDSA 署名 - 追加出力 2060
表 509. ECDSA 検証 - 入力 2060
表 510. ECDSA 検証 - 出力 2061
表 511. ECC 完全加算 2061
表 512. ECC ダブルベースラダー 2062
表 513. ECC 射影からアフィンへ 2063

表 514. ECC 演算でサポートされる曲線群 2064
表 515. べき剰余 2065
表 516. ECC スカラー乗算 2065
表 517. ECDSA 署名の平均計算時間 2066
表 518. ECDSA 検証の平均計算時間 2066
表 519. ECC ダブルベースラダーの平均計算時間 2066
表 520. ECC 射影からアフィンへの平均計算時間 2066
表 521. ECC 完全加算の平均計算時間 2066
表 522. 楕円曲線 Fp 上の点の確認の平均計算時間 2066
表 523. モンゴメリパラメータの平均計算時間 2066
表 524. PKA 割込みリクエスト 2067
表 525. PKA レジスタマップとリセット値 2071
表 526. TIM の入出力ピン 2074
表 527. TIM 内部入力／出力信号 2074
表 528. tim_ti1 入力マルチプレクサへの相互接続 2075
表 529. tim_ti2 入力マルチプレクサへの相互接続 2076
表 530. tim_ti3 入力マルチプレクサへの相互接続 2076
表 531. tim_ti4 入力マルチプレクサへの相互接続 2076
表 532. 内部トリガ接続 2076
表 533. STM32U535/545/575/585 向けの tim_etr 入力マルチプレクサへの相互接続 2077
表 534. STM2U59x/5Ax/5Fx/5Gx 向けの tim_etr 入力マルチプレクサへの相互接続 2077
表 535. タイマブレーク相互接続 2078
表 536. タイマ break2 相互接続 2078
表 537. システムブレーク相互接続 2078
表 538. ocref_clr 入力マルチプレクサへの相互接続 2079
表 539. CCR および ARR レジスタの変更のディザリングパターン 2113
表 540. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン 2114
表 541. タイマ出力と tim_brk/tim_brk2 入力の動作 2125
表 542. ブレーク保護解除条件 2127
表 543. カウント方向とエンコーダ信号（CC1P = CC2P = 0） 2136
表 544. カウント方向 対 エンコーダ信号および極性設定 2140
表 545. DMA リクエスト 2161
表 546. 低消費電力モードが TIM1/TIM8 に与える影響 2161
表 547. 割込みリクエスト 2162
表 548. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット 2188
表 549. TIMx レジスタマップとリセット値 2213
表 550. STM32U5 シリーズ汎用タイマ 2217
表 551. TIM の入出力ピン 2219
表 552. TIM 内部入力／出力信号 2219
表 553. tim_ti1 入力マルチプレクサへの相互接続 2220
表 554. tim_ti2 入力マルチプレクサへの相互接続 2220
表 555. tim_ti3 入力マルチプレクサへの相互接続 2220
表 556. tim_ti4 入力マルチプレクサへの相互接続 2221
表 557. TIMx 内部トリガ接続 2221
表 558. STM32U535/545/575/585 向けの tim_etr 入力マルチプレクサへの相互接続 2221
表 559. STM32U59x/5Ax/5Fx/5Gx 向けの tim_etr 入力マルチプレクサへの相互接続 2222
表 560. tim_ocref_clr 入力マルチプレクサへの相互接続 2223
表 561. CCR および ARR レジスタの変更のディザリングパターン 2252
表 562. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン 2253
表 563. カウント方向とエンコーダ信号（CC1P = CC2P = 0） 2262
表 564. カウント方向 対 エンコーダ信号および極性設定 2266
表 565. DMA リクエスト 2289

表 566. 低消費電力モードが TIM2/TIM3/TIM4/TIM5 に与える影響 2290
表 567. 割込みリクエスト 2290
表 568. 標準 tim_ocx チャネルの出力制御ビット 2308
表 569. TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値 2321
表 570. TIM の入出力ピン 2326
表 571. TIM 内部入力／出力信号 2327
表 572. tim_ti1 入力マルチプレクサへの相互接続 2327
表 573. tim_ti2 入力マルチプレクサへの相互接続 2328
表 574. TIMx 内部トリガ接続 2328
表 575. タイマブレーク相互接続 2328
表 576. システムブレーク相互接続 2329
表 577. ocref_clr 入力マルチプレクサへの相互接続 2329
表 578. CCR および ARR レジスタの変更のディザリングパターン 2349
表 579. ブレーク保護解除条件 2358
表 580. DMA リクエスト 2369
表 581. 低消費電力モードが TIM15/TIM16/TIM17 に与える影響 2370
表 582. 割込みリクエスト 2370
表 583. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット（TIM15） 2386
表 584. TIM15 レジスタマップとリセット値 2400
表 585. ブレーク機能を持つ相補 tim_oc1 および tim_oc1n チャネルの出力制御ビット（TIM16/TIM17） 2413
表 586. TIM16/TIM17 レジスタマップとリセット値 2427
表 587. TIM 内部入力／出力信号 2430
表 588. TIMx_ARR レジスタの変更のディザリングパターン 2440
表 589. DMA リクエスト 2441
表 590. 低消費電力モードが TIM6/TIM7 に与える影響 2441
表 591. 割込みリクエスト 2441
表 592. TIMx レジスタマップとリセット値 2448
表 593. STM32U5 シリーズ LPTIM 機能 2450
表 594. LPTIM1/2/3 の入出力ピン 2452
表 595. LPTIM4 の入出力ピン 2452
表 596. LPTIM1/2/3 内部信号 2453
表 597. LPTIM4 内部信号 2453
表 598. LPTIM1/2/3/4 外部トリガ接続 2454
表 599. LPTIM1/2/3/4 入力 1 接続 2454
表 600. LPTIM1/2 入力 2 接続 2454
表 601. LPTIM1/2/3 入力キャプチャ 1 接続 2454
表 602. LPTIM1 入力キャプチャ 2 接続 2455
表 603. LPTIM2 入力キャプチャ 2 接続 2455
表 604. LPTIM3 入力キャプチャ 2 接続 2455
表 605. プリスケーラ分周比 2457
表 606. エンコーダのカウントシナリオ 2464
表 607. 入力キャプチャのグリッチフィルタの遅延（カウンタステップ単位） 2467
表 608. 低消費電力モードが LPTIM に与える影響 2472
表 609. 割込みイベント 2473
表 610. LPTIM レジスタのマップとリセット値 2499
表 611. GFXTIM の入出力ピン 2502
表 612. GFXTIM 内部信号 2502
表 613. GFXTIM トリガの相互接続 2503
表 614. グラフィック･タイマ割込みリクエスト 2513
表 615. GFXTIM レジスタマップとリセット値 2537
表 616. STM32U5 シリーズ IWDG の機能 2540

表 617. IWDG 内部入力／出力信号 2542
表 618. 低消費電力モードが IWDG に与える影響 2546
表 619. IWDG 割込みリクエスト 2548
表 620. IWDG レジスタマップとリセット値 2553
表 621. WWDG の機能 2554
表 622. WWDG 内部入力／出力信号 2555
表 623. WWDG 割込みリクエスト 2558
表 624. WWDG レジスタのマップとリセット値 2561
表 625. RTC の入出力ピン 2564
表 626. RTC 内部入力／出力信号 2564
表 627. RTC 相互接続 2565
表 628. RTC ピン PC13 の設定 2565
表 629. RTC_OUT の配置 2567
表 630. 低消費電力モードが RTC に与える影響 2582
表 631. RTC ピン機能（モード共通） 2582
表 632. 非セキュア割込みリクエスト 2583
表 633. セキュア割込みリクエスト 2583
表 634. RTC レジスタマップとリセット値 2612
表 635. TAMP の入出力ピン 2617
表 636. TAMP 内部入力／出力信号 2617
表 637. TAMP 相互接続 2618
表 638. デバイスリソース x のタンパ保護 2623
表 639. アクティブタンパ出力の変更周期 2625
表 640. 最低 ATPER 値 2626
表 641. アクティブタンパのフィルタパルス時間 2627
表 642. 低消費電力モードが TAMP に与える影響 2629
表 643. TAMP ピン機能（モード共通） 2629
表 644. 割込みリクエスト 2629
表 645. TAMP レジスタマップとリセット値 2658
表 646. STM32U535/545/575/585 の I^2C の実装 2661
表 647. STM32U59x/5Ax/5Fx/5Gx の I2C の実装 2662
表 648. I2C の入出力ピン 2664
表 649. I2C 内部入力／出力信号 2664
表 650. I2C1、I2C2、I2C4、I2C5、I2C6 相互接続 2664
表 651. I2C3 相互接続 2665
表 652. アナログフィルタとデジタルフィルタの比較 2667
表 653. I2C-SMBus 仕様のデータのセットアップおよびホールド時間 2669
表 654. I2C 設定 2673
表 655. I2C-SMBus 仕様のクロックタイミング 2684
表 656. fI2CCLK = 8 MHz でのタイミング設定の例 2694
表 657. fI2CCLK = 16 MHz でのタイミング設定の例 2694
表 658. SMBus タイムアウト仕様 2696
表 659. SMBus の PEC 設定 2698
表 660. TIMEOUTA の設定例（最大 $t_{TIMEOUT}$ = 25ms） 2699
表 661. TIMEOUTB の設定例 2699
表 662. TIMEOUTA の設定例（最大 t_{IDLE} = 50µs） 2699
表 663. 低消費電力モードが I2C に与える影響 2711
表 664. I2C 割込みリクエスト 2712
表 665. I^2C レジスタマップとリセット値 2728
表 666. STM32U5 シリーズでの USART/UART/LPUART の実装 2731
表 667. USART/LPUART の機能 2731
表 668. USART の入出力ピン 2734

表 669. USART 内部入力／出力信号 2735
表 670. USART の相互接続（USART1/2/3/6 および UART4/5） 2735
表 671. サンプリングされたデータからのノイズ検出 2747
表 672. BRR [3:0] = 0000 のときの USART レシーバの許容誤差 2750
表 673. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差 2751
表 674. USART フレームのフォーマット 2755
表 675. 低消費電力モードが USART に与える影響 2775
表 676. USART 割込みリクエスト 2776
表 677. USART レジスタマップとリセット値 2812
表 678. STM32U5 シリーズでの USART/UART/LPUART の実装 2815
表 679. USART/LPUART の機能 2815
表 680. LPUART の入出力ピン 2818
表 681. LPUART 内部入力／出力信号 2818
表 682. LPUART の相互接続（LPUART1） 2819
表 683. lpuart_ker_ck_pres = 32.768kHz でプログラムされたボーレートのエラー計算 2830
表 684. fCK = 100MHz でプログラムされたボーレートのエラー計算 2830
表 685. LPUART レシーバの許容誤差 2831
表 686. LPUART フレームのフォーマット 2834
表 687. 低消費電力モードが LPUART に与える影響 2841
表 688. LPUART 割込みリクエスト 2842
表 689. LPUART レジスタマップとリセット値 2866
表 690. SPI の機能 2869
表 691. SPI の入出力ピン 2871
表 692. SPI 内部入力／出力信号 2872
表 693. SPI の相互作用（SPI1 および SPI2） 2872
表 694. SPI の相互接続（SPI3） 2873
表 695. 低電力モードが SPI に与える影響 2899
表 696. SPI のウェイクアップおよび割込みリクエスト 2900
表 697. SPI レジスタマップとリセット値 2918
表 698. SAI の機能 2920
表 699. SAI 内部入力／出力信号 2922
表 700. SAI の入出力ピン 2922
表 701. 外部同期の選択 2924
表 702. MCLK_x の有効化条件 2930
表 703. クロックジェネレータのプログラミング例 2933
表 704. TDM 設定 2940
表 705. TDM フレーム設定の例 2942
表 706. SOPD パターン 2946
表 707. パリティビットの計算 2946
表 708. オーディオサンプリング周波数とシンボルレート 2947
表 709. SAI 割込みソース 2955
表 710. SAI レジスタマップとリセット値 2985
表 711. CAN サブシステム I/O 信号 2987
表 712. FDCAN での DLC コーディング 2994
表 713. フレーム送信の可能な設定 3006
表 714. Rx FIFO 要素 3009
表 715. Rx FIFO 要素の説明 3009
表 716. Tx バッファおよび FIFO 要素 3011
表 717. Tx バッファ要素の説明 3011
表 718. Tx イベント FIFO 要素 3013
表 719. Tx イベント FIFO 要素の説明 3013
表 720. 標準メッセージ ID フィルタ要素 3014

表 721. 標準メッセージ ID フィルタ要素フィールドの説明 3014
表 722. 拡張メッセージ ID フィルタ要素. 3015
表 723. 拡張メッセージ ID フィルタ要素フィールドの説明 3015
表 724. FDCAN レジスタのマップとリセット値 3047
表 725. STMU535/545 での USB 実装 3050
表 726. ダブルバッファリングバッファフラグの定義. 3062
表 727. バルクダブルバッファリングメモリバッファの用途（デバイスモード） 3063
表 728. バルクダブルバッファリングメモリバッファの用途（ホストモード） 3064
表 729. アイソクロナスメモリバッファの使用 3066
表 730. アイソクロナスメモリバッファの使用 3067
表 731. レジュームイベント検出 3068
表 732. ホスト用のレジュームイベント検出 3069
表 733. 受信ステータスエンコード 3087
表 734. エンドポイント / チャネルタイプのエンコード 3087
表 735. エンドポイント / チャネルの種類の意味. 3087
表 736. 送信ステータスエンコード 3087
表 737. 割り当てられるバッファメモリの定義 3090
表 738. USB レジスタのマップとリセット値 3093
表 739. サポートされる OTG_FS 速度 3095
表 740. OTG_FS の実装. 3098
表 741. OTG_FS の入出力ピン 3100
表 742. OTG_FS 入力／出力信号 3100
表 743. STM32 の低消費電力モードの OTG との両立性. 3113
表 744. コアグローバル制御およびステータスレジスタ（CSR） 3121
表 745. ホストモード制御およびステータスレジスタ （CSR). 3122
表 746. デバイスモード制御およびステータスレジスタ 3123
表 747. データ FIFO（DFIFO）アクセススレジスタマップ 3125
表 748. パワーおよびクロックゲーティング制御およびステータスレジスタ 3125
表 749. TRDT 値. 3132
表 750. ソフト切断のための最短時間 3168
表 751. OTG_FS レジスタマップとリセット値. 3191
表 752. サポートされる OTG_HS の速度 3250
表 753. OTG_HS の実装 3251
表 754. OTG_HS の入出力ピン 3253
表 755. OTG_HS 入出力信号. 3253
表 756. STM32 の低消費電力モードの OTG との両立性. 3264
表 757. コアグローバル制御およびステータスレジスタ（CSR） 3272
表 758. ホストモード制御およびステータスレジスタ （CSR). 3274
表 759. デバイスモード制御およびステータスレジスタ 3275
表 760. データ FIFO（DFIFO）アクセススレジスタマップ 3276
表 761. パワーおよびクロックゲーティング制御およびステータスレジスタ 3277
表 762. TRDT 値. 3282
表 763. ソフト切断のための最短時間 3326
表 764. レジスタマップとリセット値 3353
表 765. UCPD の実装. 3419
表 766. UCPD ソフトウェアの trim データ 3419
表 767. ピンの UCPD 信号. 3420
表 768. UCPD 内部信号 3421
表 769. 4b5b 記号エンコード表 3423
表 770. 順序集合 3424
表 771. 順序集合の検証 3425
表 772. データサイズ 3425

表 773. ANAMODE と ANASUBMODE のコーディングおよび TYPEC_VSTATE_CCx とのリンク 3434
表 774. Type-C シーケンス（転送元：3A）、ケーブル／シンク接続済み（CC1 では Rd、CC2 では Ra） 3435
表 775. 低電力モードが UCPD に与える影響 3438
表 776. UCPD 割込みリクエスト 3439
表 777. UCPD レジスタマップとリセット値 3456
表 778. JTAG ／シリアルワイヤデバッグポートのピン 3459
表 779. トレースポートピン 3459
表 780. シングルワイヤトレースポートのピン 3459
表 781. 認証信号の状態 3461
表 782. JTAG-DP データレジスタ 3464
表 783. パケットリクエスト 3466
表 784. ACK 応答 3466
表 785. データ転送 3466
表 786. デバッグポートのレジスタマップとリセット値 3473
表 787. アクセスポートのレジスタマップとリセット値 3479
表 788. MCU ROM テーブル 3480
表 789. プロセッサ ROM テーブル 3480
表 790. MCU ROM テーブルレジスタのマップとリセット値 3487
表 791. CPU ROM テーブルレジスタのマップとリセット値 3493
表 792. DWT レジスタのマップとリセット値 3509
表 793. ITM レジスタのマップとリセット値 3520
表 794. BPU レジスタのマップとリセット値 3529
表 795. ETM レジスタのマップとリセット値 3555
表 796. TPIU レジスタのマップとリセット値 3569
表 797. CTI 入力 3571
表 798. CTI 出力 3572
表 799. CTI レジスタのマップとリセット値 3583
表 800. ペリフェラルクロックフリーズ制御ビット 3585
表 801. デバッグモードでのペリフェラルの動作 3586
表 802. フリーズレジスタビットへのデバッガのアクセス 3587
表 803. DBGMCU レジスタマップとリセット値 3599
表 804. 文書改版履歴 3606

図の一覧

図 1. システムアーキテクチャ 127
図 2. SmartRun ドメイン·アーキテクチャ 130
図 3. STM32U535/545 の IDAU マッピングに基づくメモリマップ 137
図 4. STM32U575/585 の IDAU マッピングに基づくメモリマップ 138
図 5. STM32U59x/5Ax の IDAU マッピングに基づくメモリマップ 139
図 6. STM32U5Fx/5Gx の IDAU マッピングに基づくメモリマップ 140
図 7. TrustZone テクノロジーを使用したセキュア / 非セキュアなパーティショニング 150
図 8. セキュア状態の CPU と非セキュア状態の CPU 間のメモリマップの共有 152
図 9. セキュアワールド遷移とメモリパーティショニング 152
図 10. グローバル TrustZone フレームワークと TrustZone の認識 154
図 11. Flash メモリの TrustZone 保護 158
図 12. Flash メモリのセキュア HDP 領域 166
図 13. キー管理原則 174
図 14. デバイスライフサイクルセキュリティ 178
図 15. RDP レベル遷移図 180
図 16. 共同開発の原則 183
図 17. SFI を使用した外部 Flash メモリの保護 185
図 18. Armv8-M サブシステム内の GTZC のブロック図 192
図 19. GTZC ブロック図 195
図 20. ウォーターマークベースのメモリ保護コントローラ（領域 x/ サブ領域 A および B） 197
図 21. MPCBB ブロック図 198
図 22. SRAM1、ECC 付きの SRAM2、および ECC 付きの SRAM3 メモリマップ 273
図 23. バンクスワップがない場合の Flash メモリのセキュリティ属性および保護（SWAP_BANK = 0） 307
図 24. バンクスワップがある場合の Flash メモリのセキュリティ属性および保護（SWAP_BANK = 1） 307
図 25. TrustZone が無効（TZEN = 0）な場合の RDP レベル遷移図 315
図 26. TrustZone が有効（TZEN = 1）な場合の RDP レベル遷移図 316
図 27. ICACHE ブロック図 360
図 28. ICACHE タグおよびデータメモリの機能図 362
図 29. ICACHE 再配置アドレスのメカニズム 365
図 30. DCACHE のブロック図 378
図 31. DCACHE タグおよびデータメモリの機能図 381
図 32. 電源の概要 398
図 33. ブラウンアウトリセット波形 406
図 34. PVD の閾値 406
図 35. 簡略化されたリセット回路図 476
図 36. STM32U5 シリーズ用のクロックツリー 479
図 37. HSE/LSE クロックソース 480
図 38. MSI ブロック図 482
図 39. PLL ブロック図 485
図 40. PLL 初期化フロー 487
図 41. CRS のブロック図 607
図 42. CRS カウンタの動作 608
図 43. 3V または 5V トレラント GPIO（TT または FT）の構造 618
図 44. 入力フローティング／プルアップ／プルダウン設定 622
図 45. 出力設定 623
図 46. オルタネート機能設定 624
図 47. ハイインピーダンスアナログ設定 624

図 48. I/O 補正セルのブロック図 645
図 49. GPDMA ブロック図 687
図 50. リンクリストを用いない GPDMA チャネル･ダイレクト･プログラミング（GPDMA_CxLLR = 0） 688
図 51. GPDMA チャネルのサスペンドとレジュームのシーケンス 689
図 52. GPDMA チャネルのアボートとリスタートのシーケンス 690
図 53. チャネル x のリニア･アドレス指定の静的リンクリスト･データ構造（すべて Uxx = 1） 692
図 54. チャネル x の 2D ･アドレス指定の静的リンクリスト･データ構造（すべて Uxx = 1） 693
図 55. チャネル x のリニア･アドレス指定の GPDMA 動的リンクリスト･データ構造 693
図 56. チャネル x の 2D ･アドレス指定の GPDMA 動的リンクリスト･データ構造 694
図 57. run-to-completion モード (GPDMA_CxCR.LSM = 0) での GPDMA チャネル実行およびリンクリスト･プログラミング 696
図 58. 補助 GPDMA チャネル y による LLIn の挿入 698
図 59. リンク･ステップ･モード (GPDMA_CxCR.LSM = 1) での GPDMA チャネル実行およびリンクリスト･プログラミング 700
図 60. LLIn+1 の設定：リンク･ステップ･モードでの GPDMA 動的リンクリスト 701
図 61. リンク･ステップ･モードでのレジスタ･ファイル内の新しい LLIn' による置換 702
図 62. リンク･ステップ･モードでのメモリ内の新しい LLIn' および LLIn+1' による置換（オプション 1） 703
図 63. リンク･ステップ･モードでのメモリ内の新しい LLIn' および LLIn+1' による置換（オプション 2） 704
図 64. GPDMA チャネル実行およびリンクリスト･プログラミング 706
図 65. プログラムされた 2D アドレス指定 709
図 66. GPDMA アービトレーション･ポリシー 716
図 67. トリガのヒット、メモリ書込み、オーバーラン波形 719
図 68. GPDMA サーキュラ･バッファ･プログラミング：リニア･アドレス指定によるメモリ開始アドレスの更新 720
図 69. サーキュラ･バッファのある共有 GPDMA チャネル：リニア･アドレス指定によるメモリ開始アドレスの更新 721
図 70. LPDMA ブロック図 761
図 71. リンクリストを用いない LPDMA チャネルの直接プログラミング（LPDMA_CxLLR = 0） 762
図 72. LPDMA チャネルのサスペンドとレジュームのシーケンス 763
図 73. LPDMA チャネルのアボートとリスタートのシーケンス 764
図 74. チャネル x の静的リンクリスト･データ構造（すべての Uxx = 1） 765
図 75. チャネル x のリニア･アドレス指定の LPDMA 動的リンクリスト･データ構造. 766
図 76. run-to-completion モード（LPDMA_CxCR.LSM = 0）での LPDMA チャネル実行およびリンクリスト･プログラミング 768
図 77. 補助 LPDMA チャネル y による LLIn の挿入 770
図 78. リンク･ステップ･モード（LPDMA_CxCR.LSM = 1）での LPDMA チャネル実行およびリンクリスト･プログラミング 772
図 79. LLIn+1 の設定：リンク･ステップ･モードでの LPDMA 動的リンクリスト. 773
図 80. リンク･ステップ･モードでのレジスタ･ファイル内の新しい LLIn' による置換 774
図 81. リンク･ステップ･モードでのメモリ内の新しい LLIn' および LLIn+1' による置換（オプション 1） 775
図 82. リンク･ステップ･モードでのメモリ内の新しい LLIn' および LLIn+1' による置換（オプション 2） 776
図 83. LPDMA チャネル実行およびリンクリスト･プログラミング. 778
図 84. LPDMA アービトレーション･ポリシー. 782
図 85. トリガのヒット、メモリ書込み、オーバーラン波形 786
図 86. LPDMA サーキュラ･バッファのプログラミング：メモリの開始アドレスの更新 787
図 87. サーキュラ･バッファリングが実行される共有 LPDMA チャネル：メモリの開始アドレスの更新 787
図 88. DMA2D ブロック図 814

図 89. Intel 8080 16bit モード（RGB565）....821
図 90. Intel 8080 18/24bit モード（RGB888）....821
図 91. DMA2D ブロック図....849
図 92. Intel 8080 16bit モード（RGB565）....856
図 93. Intel 8080 18/24bit モード（RGB888）....856
図 94. GFXMMU ブロック図....884
図 95. 仮想バッファ....885
図 96. 仮想バッファと物理バッファのメモリマップ....886
図 97. MMU ブロック図....887
図 98. ブロックの検証／コンパレータの実装....888
図 99. EXTI ブロック図....909
図 100. 設定可能なイベントトリガロジックによる CPU ウェイクアップ....911
図 101. EXTI マルチプレクサの GPIO 選択....912
図 102. CRC 計算ユニットのブロック図....927
図 103. 三角関数に対する CORDIC の収束....941
図 104. 双曲線関数に対する CORDIC の収束....942
図 105. 平方根に対する CORDIC の収束....943
図 106. ブロック図....952
図 107. 入力バッファ領域....954
図 108. サーキュラ入力バッファ....955
図 109. サーキュラ入力バッファの操作....956
図 110. サーキュラ出力バッファ....957
図 111. サーキュラ出力バッファの操作....958
図 112. FIR フィルタの構造....960
図 113. IIR フィルタの構造（直接型 1）....962
図 114. X1 バッファの初期化....967
図 115. フィルタ例 1....968
図 116. フィルタ例 2....969
図 117. FMC ブロック図....980
図 118. FMC メモリバンク....982
図 119. モード 1 読出しアクセス波形....989
図 120. モード 1 書込みアクセス波形....990
図 121. モード A 読出しアクセス波形....991
図 122. モード A 書込みアクセス波形....992
図 123. モード 2 およびモード B 読出しアクセス波形....994
図 124. モード 2 書込みアクセス波形....994
図 125. モード B 書込みアクセス波形....995
図 126. モード C 読出しアクセス波形....997
図 127. モード C 書込みアクセス波形....997
図 128. モード D 読出しアクセス波形....999
図 129. モード D 書込みアクセス波形....1000
図 130. マルチプレクス読出しアクセス波形....1002
図 131. マルチプレクス書込みアクセス波形....1003
図 132. 読出しアクセス中の非同期ウェイト波形....1005
図 133. 書込みアクセス中の非同期ウェイト波形....1006
図 134. 待ち設定波形....1008
図 135. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM）....1008
図 136. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM）....1010
図 137. 共通メモリ･アクセスの NAND Flash コントローラ波形....1022
図 138. 「CE don’t care」でない NAND Flash へのアクセス....1024
図 139. OCTOSPI ブロック図（オクタル構成）
（STM32U535/545 デバイスを除く STM32U5 シリーズ）....1035

図 140. OCTOSPI ブロック図（クワッド構成）（STM32U535/545 デバイスを除く STM32U5 シリーズ）....1035
図 141. OCTOSPI ブロック図（デュアルクワッド構成）（STM32U535/545 デバイスを除く STM32U5 シリーズ）....1036
図 142. OCTOSPI ブロック図（オクタル構成）（STM32U535/545 デバイスのみ）....1036
図 143. OCTOSPI ブロック図（クワッド構成）（STM32U535/545 デバイスのみ）....1037
図 144. OCTOSPI ブロック図（デュアルクワッド構成）（STM32U535/545 デバイスのみ）....1037
図 145. オクタル構成における SDR 読出しコマンド....1039
図 146. Octal-SPI モードにおける DQS による DTR 読出し（Macronix モード）の例....1042
図 147. Octo-SPI モードにおける SDR 書込みコマンドの例....1044
図 148. Octal-SPI モードにおける DTR 書込み（Macronix モード）の例....1045
図 149. HyperBus 読出し動作の例....1046
図 150. 初期遅延を設けた HyperBus 書込み動作....1048
図 151. 追加の遅延を設けた HyperBus 読出し動作....1048
図 152. 追加の遅延を設けた HyperBus 書込み動作....1049
図 153. 遅延なしの HyperBus 書込み動作（レジスタ書込み）....1049
図 154. 遅延を設けた HyperBus のページまたがり時の読出し動作....1050
図 155. CKMODE = 0 のときの NCS（T = CLK 周期）....1061
図 156. CKMODE = 1 で、SDR モードのときの NCS（T = CLK 周期）....1061
図 157. CKMODE=1 で、DTR モードのときの NCS（T=CLK 周期）....1061
図 158. CKMODE = 1 で、アボートが発生したときの NCS（T=CLK 周期）....1062
図 159. OCTOSPIM ブロック図....1091
図 160. 16bit 構成の HSPI ブロック図....1099
図 161. デュアルオクタル構成の HSPI ブロック図....1100
図 162. オクタル構成の HSPI ブロック図....1100
図 163. クワッド構成の HSPI ブロック図....1101
図 164. デュアルクワッド構成の HSPI ブロック図....1102
図 165. 16bit 構成での SDR 読出しコマンド....1104
図 166. Octal-SPI モードにおける DQS による DTR 読出し（Macronix モード）の例....1107
図 167. octal-SPI モードにおける SDR 書込みコマンドの例....1109
図 168. Octal-SPI モードにおける DTR 書込み（Macronix モード）の例....1109
図 169. HyperBus 読出し動作の例（8bit データモード）....1112
図 170. 初期遅延を設けた HyperBus 書込み動作（8bit データモード）....1113
図 171. 追加の遅延を設けた HyperBus 読出し動作（8bit データモード）....1113
図 172. 追加の遅延を設けた HyperBus 書込み動作（8bit データモード）....1114
図 173. 遅延なしの HyperBus 書込み動作（レジスタ書込み）....1114
図 174. 遅延を設けた HyperBus のページ跨がり時の読出し動作（8bit データモード）....1115
図 175. 初期遅延を設けた HyperBus 書込み動作（16bit モード）....1116
図 176. CKMODE = 0 のときの NCS（T = CLK 周期）....1127
図 177. CKMODE = 1 で、SDR モードのときの NCS（T = CLK 周期）....1127
図 178. CKMODE = 1 で、DTR モードのときの NCS（T=CLK 周期）....1128
図 179. CKMODE = 1 で、アボートが発生したときの NCS（T = CLK 周期）....1128
図 180. SDMMC「レスポンス無し」と「データなし」操作....1160
図 181. SDMMC（複数）ブロック読出し操作....1160
図 182. SDMMC（複数）ブロック書込み操作....1161
図 183. SDMMC（シーケンシャル）ストリーム読出し操作....1161
図 184. SDMMC（シーケンシャル）ストリーム書込み操作....1161
図 185. SDMMC のブロック図....1163
図 186. SDMMC コマンドおよびデータの位相関係....1165
図 187. 制御ユニット....1167

図 188. コマンド / レスポンスパス 1168
図 189. コマンドパスステートマシン（CPSM） 1169
図 190. データパス 1174
図 191. DDR モードのデータパケットクロック供給 1175
図 192. DDR モードの CRC ステータス／ブート確認応答クロック供給 1175
図 193. データパスステートマシン（DPSM） 1176
図 194. CLKMUX ユニット 1185
図 195. リンクリストの構造 1188
図 196. 非同期割込みの生成 1191
図 197. 同期割込み期間のデータ読出し 1191
図 198. 同期割込み期間のデータ書込み 1192
図 199. 非同期割込み期間のデータ読出し 1193
図 200. 非同期割込み期間のデータ書込み 1193
図 201. DS、ハイスピード、SDR12、SDR25 向け SDMMC_CK でのクロック停止 1196
図 202. DDR50、SDR50、SDR104 向け SDMMC_CK でのクロック停止 1196
図 203. 50MHz 未満の SDMMC_CK での読出しウェイト 1197
図 204. 50MHz 以上の SDMMC_CK での読出しウェイト 1197
図 205. CMD12 ストリームタイミング 1199
図 206. CMD5 SLEEP AWAKE 手順 1201
図 207. 通常ブートモード動作 1203
図 208. 代替ブートモード動作 1204
図 209. コマンドレスポンス R1b ビジー信号 1206
図 210. SDMMC の状態制御 1207
図 211. カードサイクルパワー／パワーアップの図 1208
図 212. CMD11 信号電圧スイッチシーケンス開始 1209
図 213. 電圧スイッチトランシーバの標準アプリケーション 1211
図 214. DLYB ブロック図 1239
図 215. ADC ブロック図 1246
図 216. ADC クロック構成 1249
図 217. ADC1 接続性 1251
図 218. ADC2 接続性（STM32U59x/5Ax /5Fx/5Gx） 1252
図 219. ADC 較正 1255
図 220. ADC の有効化 / 無効化 1258
図 221. バルブモードのタイミング図 1261
図 222. シングル変換でのアナログデジタル変換の時間 1264
図 223. 実行中のレギュラ変換の停止 1265
図 224. 実行中のレギュラ変換とインジェクト変換の停止 1265
図 225. ADC マスタと ADC スレーブ間でトリガを共有 1267
図 226. インジェクト変換の遅延時間 1268
図 227. シーケンスのシングル変換、ソフトウェアトリガ 1271
図 228. シーケンスの連続変換、ソフトウェアトリガ 1271
図 229. シーケンスのシングル変換、ハードウェアトリガ 1272
図 230. シーケンスの連続変換、ハードウェアトリガ 1272
図 231. 右詰め（オフセット無効、符号なしの値） 1274
図 232. 右詰め（オフセット有効、符号付きの値） 1275
図 233. 左詰め（オフセット無効、符号なしの値） 1275
図 234. 左詰め（オフセット有効、符号付きの値） 1276
図 235. オーバーラン（OVRMOD = 0）の例 1279
図 236. オーバーラン（OVRMOD = 1）の例 1279
図 237. AUTDLY = 1、連続モードのレギュラ変換、ソフトウェアトリガ 1283
図 238. AUTODLY = 1、インジェクト変換によって中断されたレギュラハードウェア変換
（DISCEN = 0、JDISCEN = 0） 1283

図 239. AUTODLY = 1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN = 1、JDISCEN = 1）. 1284
図 240. AUTDLY = 1、インジェクト変換によって中断されたレギュラ連続変換. 1285
図 241. 自動インジェクトモードの AUTDLY = 1（JAUTO = 1）. 1285
図 242. アナログウォッチドッグによって保護される領域 1286
図 243. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）. 1288
図 244. ADC_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合）. . . . 1288
図 245. ADC_AWDx_OUT 信号の生成（1 つのレギュラチャネル）. 1289
図 246. ADC_AWDx_OUT 信号の生成（すべてのインジェクトチャネル）. 1289
図 247. 10bit 右シフトと丸めを行う 14bit 結果のオーバーサンプリング 1290
図 248. オーバーサンプリングモードのトリガ（TROVS ビット =1）. 1292
図 249. レギュラオーバーサンプリングモード（4x の比率）. 1293
図 250. レギュラおよびインジェクトオーバーサンプリングモードの同時使用. 1293
図 251. インジェクションでのトリガレギュラオーバーサンプリング 1294
図 252. 自動インジェクトモードでのオーバーサンプリング 1294
図 253. デュアル ADC ブロック図 (1) 1297
図 254. 4 チャネルのインジェクト同時モード：デュアル ADC モード 1298
図 255. 16 チャネルのレギュラ同時モード：デュアル ADC モード 1300
図 256. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 1301
図 257. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 1302
図 258. インジェクションによるインタリーブ変換 1302
図 259. オルタネートトリガ：各 ADC のインジェクトグループ 1303
図 260. オルタネートトリガ：不連続モードでの 4 つのインジェクトチャネル（各 ADC）. 1304
図 261. オルタネート + レギュラ同時 1305
図 262. インジェクト変換中にトリガが発生した場合. 1305
図 263. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0 1306
図 264. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割込み優先. 1306
図 265. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割込み優先 1306
図 266. レギュラ同時モードにおける DMA リクエスト（DAMDF[1:0] = 00）. 1307
図 267. レギュラ同時モードにおける DMA リクエスト（DAMDF[1:0] = 10）. 1308
図 268. インタリーブモードにおける DMA リクエスト（DAMDF[1:0] = 10）. 1308
図 269. 温度センサチャネルのブロック図. 1310
図 270. VBAT チャネルのブロック図 1312
図 271. VREFINT チャネルのブロック図 1312
図 272. ADC ブロック図 1358
図 273. ADC 較正 1361
図 274. 較正係数によって強制される 1362
図 275. ADC の有効化 / 無効化 1363
図 276. ADC クロック構成. 1363
図 277. ADC4 接続性 1365
図 278. アナログ / デジタル変換時間. 1370
図 279. ADC 変換タイミング 1370
図 280. 実行中の変換の停止 1371
図 281. シーケンスのシングル変換、ソフトウェアトリガ 1374
図 282. シーケンスの連続変換、ソフトウェアトリガ. 1375
図 283. シーケンスのシングル変換、ハードウェアトリガ 1375
図 284. シーケンスの連続変換、ハードウェアトリガ. 1376
図 285. データの配置と分解能（オーバーサンプリング無効：OVSE = 0）. 1377
図 286. オーバーラン（OVR）の例. 1378
図 287. ウェイト変換モード（連続モード、ソフトウェアトリガ）. 1380
図 288. オートオフモードの状態図 1381

図 289. WAIT = 0 かつ AUTOFF = 1 での ADC の動作 1382
図 290. WAIT = 1 かつ AUTOFF = 1 での ADC の動作 1382
図 291. 自律モードの状態図 1383
図 292. アナログウォッチドッグによって保護される領域 1384
図 293. ADC_AWDx_OUT 信号生成 1386
図 294. ADC_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合） 1386
図 295. ADC_AWDx_OUT 信号生成（1 つのチャネル） 1387
図 296. アナログウォッチドッグ閾値更新 1387
図 297. 20bit から 16bit への結果の切詰め 1388
図 298. 5bit シフトと丸めの数値例 1388
図 299. トリガオーバーサンプリングモード（TOVS ビット = 1） 1390
図 300. 温度センサおよび VREFINT チャネルのブロック図 1392
図 301. VBAT チャネルブロック図 1394
図 302. デュアルチャネル DAC ブロック図 1422
図 303. シングル DAC チャネルモードのデータレジスタ 1425
図 304. デュアル DAC チャネルモードのデータレジスタ 1426
図 305. トリガ無効（TEN = 0）時の変換タイミング図 1427
図 306. DAC LFSR レジスタ計算アルゴリズム 1429
図 307. LFSR 波形生成による DAC 変換（SW トリガ有効） 1430
図 308. DAC 三角波生成 1430
図 309. 三角波生成による DAC 変換（SW トリガ有効） 1431
図 310. D/A コンバータサンプルおよびホールドモードフェーズの図 1433
図 311. VREFBUF のブロック図 1460
図 312. コンパレータのブロック図 1465
図 313. ウィンドウモード 1467
図 314. コンパレータヒステリシス 1468
図 315. コンパレータ出力のブランキング 1469
図 316. スケーラ 1470
図 317. スタンドアロンモード：外部ゲイン設定モード 1477
図 318. フォロア設定 1478
図 319. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力 1479
図 320. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力 1480
図 321. MDF ブロック図 1492
図 322. SITFx の概要 1496
図 323. SPI タイミングの例 1497
図 324. マンチェスタのタイミング例（SITFMOD = 11） 1499
図 325. CKGEN の概要 1501
図 326. BSMX の概要 1504
図 327. SCD の機能図 1505
図 328. SCD タイミングの例 1506
図 329. DFLT の概要 1507
図 330. プログラム可能な遅延 1508
図 331. デシメーション比 = 32 での CIC3 および CIC5 周波数応答 1510
図 332. 正規化された再形成フィルタの周波数応答（FRS / 2 = 1） 1515
図 333. 限界外検出回路の閾値 1518
図 334. DFLT および CKGEN のためのトリガ･ロジック 1520
図 335. 非同期連続モード（ACQMOD[2:0] = 0） 1521
図 336. 非同期シングルショット･モード（ACQMOD[2:0] = 001） 1522
図 337. 同期連続モード（ACQMOD[2:0] = 010） 1523
図 338. 同期シングルショット･モード（ACQMOD[2:0] = 011） 1524
図 339. ウィンドウ連続モード（ACQMOD[2:0] = 100） 1526
図 340. スナップショット･モードの例 1527
図 341. 破棄機能の例 1528

図 342. 連続モード、オーディオ設定での、DFLTEN による開始シーケンス 1529
図 343. 連続モード、モータ設定での、トリガ入力による開始シーケンス 1530
図 344. ブレーク･インタフェースの概略図.....1531
図 345. MDF_DFLTxDR データ･フォーマット 1531
図 346. データの再同期化 1532
図 347. インタリーブ転送モードでのデータ転送 1534
図 348. インタリーブ転送モードと独立転送モードのためのデータ･パス 1535
図 349. オーバーフローとメモリへの転送の例 1536
図 350. MDF 割込みインタフェース 1539
図 351. センサの接続例 1544
図 352. グローバル周波数応答 1544
図 353. 周波数応答の詳細 1545
図 354. DFLT の概略とゲイン情報.....1547
図 355. ADF ブロック図 1577
図 356. SITF の概要 1579
図 357. SPI タイミングの例 1580
図 358. マンチェスタのタイミング例（SITFMOD = 11）.....1581
図 359. CKGEN の概要 1584
図 360. BSMX の概要 1586
図 361. DFLT の概要 1587
図 362. プログラム可能な遅延.....1588
図 363. デシメーション比 = 32 または 16 での CIC4 および CIC5 周波数応答 1589
図 364. 正規化された再形成フィルタの周波数応答（FRS / 2 = 1）.....1595
図 365. DFLT および CKGEN のためのトリガ･ロジック 1597
図 366. 非同期連続モード（ACQMOD[2:0] = 0）.....1598
図 367. 非同期シングルショット･モード（ACQMOD[2:0] = 001）.....1599
図 368. 同期連続モード（ACQMOD[2:0] = 010）.....1600
図 369. 同期シングルショット･モード（ACQMOD[2:0] = 011）.....1601
図 370. ウィンドウ連続モード（ACQMOD[2:0] = 100）.....1602
図 371. 破棄機能の例.....1603
図 372. 連続モード、オーディオ設定での、DFLTEN による開始シーケンス 1604
図 373. SAD ブロック図.....1605
図 374. SAD のフロー図.....1607
図 375. SAD のタイミング図の例 1613
図 376. ADF_DFLTxDR データ･フォーマット 1613
図 377. データの再同期化 1614
図 378. オーバーフローとメモリへの転送の例 1615
図 379. ADF 割込みインタフェース 1618
図 380. センサの接続例 1623
図 381. グローバル周波数応答.....1624
図 382. 周波数応答の詳細 1624
図 383. DFLT の概略とゲイン情報.....1627
図 384. SADMOD = 01 で機能する SAD の例 1630
図 385. SADMOD = 1x で機能する SAD の例 1633
図 386. DCMI のブロック図 1654
図 387. DCMI 信号の波形.....1655
図 388. タイミング図.....1657
図 389. スナップショットモードにおけるフレームキャプチャ波形.....1659
図 390. 連続グラブモードにおけるフレームキャプチャ波形 1660
図 391. クロップ後のウィンドウの座標とサイズ 1660
図 392. データキャプチャ波形.....1661
図 393. ピクセルラスタースキャンの順序.....1662
図 394. PSSI ブロック図 1678

図 395. 最上位ブロック図 1678
図 396. 受信モードでのデータイネーブルの波形図（CKPOL=0）. 1682
図 397. 送信モードでのデータイネーブルの波形図（CKPOL=0）. 1682
図 398. 受信モードでのレディの波形図（CKPOL=0）. 1683
図 399. 双方向 PSSI_DE/PSSI_RDY の波形 1684
図 400. 双方向 PSSI_DE/PSSI_RDY の接続図 1684
図 401. LTDC ブロック図 1693
図 402. LTDC 同期タイミング 1696
図 403. レイヤウィンドウのプログラム可能なパラメータ 1698
図 404. 2 つのレイヤとバックグラウンドとのブレンディング 1701
図 405. 割込みイベント 1702
図 406. DSI ブロック図 1725
図 407. DSI ホストのアーキテクチャ 1727
図 408. シャドウレジスタを使用して LTDC インタフェース設定を更新するフロー 1732
図 409. 即時更新手順 1733
図 410. フレーム送信中の設定更新 1733
図 411. アダプテッド･コマンド･モードの使用フロー. 1735
図 412. 24 bpp の APB ピクセルからバイトへの構成 1739
図 413. 18 bpp の APB ピクセルからバイトへの構成 1740
図 414. 16 bpp の APB ピクセルからバイトへの構成 1740
図 415. 12 bpp の APB ピクセルからバイトへの構成 1741
図 416. 8 bpp の APB ピクセルからバイトへの構成 1741
図 417. バスターンアラウンド後の PRESP_TO のタイミング. 1744
図 418. 読出しリクエスト後の PRESP_TO のタイミング（HS または LP）. 1745
図 419. 書込みリクエスト後の PRESP_TO のタイミング（HS または LP）. 1746
図 420. Presp モードが 1 のときの効果. 1747
図 421. 画像領域内のコマンド送信期間 1748
図 422. フレームの最終ラインでのコマンドの送信 1748
図 423. 同期パルスでの非バーストの LPSIZE. 1750
図 424. バーストまたは同期イベントでの非バーストの LPSIZE 1750
図 425. 同期パルスでの非バーストの VLPSIZE 1751
図 426. 同期イベントでの非バーストの VLPSIZE. 1751
図 427. バースト･モードの VLPSIZE. 1752
図 428. 画像領域での LPSIZE と VLPSIZE の位置. 1753
図 429. HS でのクロック･レーンとデータ･レーン 1755
図 430. HS でのクロック･レーンと LP でのデータ･レーン 1755
図 431. LP でのクロック･レーンとデータ･レーン 1755
図 432. 汎用インタフェースによるコマンドの送信 1756
図 433. 垂直カラーバー･モード. 1758
図 434. 水平カラーバー･モード. 1758
図 435. RGB888 の BER テスト･パターン 1759
図 436. 垂直パターン（103x15）. 1760
図 437. 水平パターン（103x15）. 1760
図 438. PLL ブロック図 1762
図 439. エラー発生源 1765
図 440. ビデオ･パケット送信の設定フロー図 1775
図 441. テストパターンを送信するためのプログラミング･シーケンス 1777
図 442. フレーム設定レジスタ 1778
図 443. GPU2D ブロック図 1843
図 444. JPEG コーデックブロック図 1845
図 445. TSC ブロック図. 1867
図 446. 表面電荷移動アナログ I/O グループの構造. 1868
図 447. サンプリングコンデンサの電圧の変化 1869

図 448. 電荷移動取得シーケンス 1870
図 449. スペクトル拡散の変動の原理 1871
図 450. RNG のブロック図 1885
図 451. NIST SP800-90B エントロピーソースモデル 1886
図 452. RNG 初期化の概要 1889
図 453. AES ブロック図 1901
図 454. ECB の暗号化と復号化の原理 1903
図 455. CBC の暗号化と復号化の原理 1904
図 456. CTR の暗号化と復号化の原理 1905
図 457. GCM の暗号化と認証の原理 1906
図 458. GMAC の認証の原理 1906
図 459. CCM の暗号化と認証の原理 1907
図 460. サスペンドモードの管理例 1911
図 461. ECB 暗号化 1912
図 462. ECB 復号化 1912
図 463. CBC 暗号化 1913
図 464. CBC 復号化 1913
図 465. ECB/CBC 暗号化（モード 1） 1914
図 466. ECB/CBC 復号化（モード 3） 1915
図 467. CTR モードでのメッセージ構造 1916
図 468. CTR 暗号化 1917
図 469. CTR 復号化 1917
図 470. GCM でのメッセージ構造 1919
図 471. GCM 認証暗号化 1920
図 472. GMAC モードでのメッセージ構造 1924
図 473. GMAC 認証モード 1924
図 474. CCM モードでのメッセージ構造 1925
図 475. CCM モード認証暗号化 1927
図 476. データスワッピングに関する 128bit ブロックの構成 1932
図 477. 入力フェーズでの 128bit データブロックの DMA 転送 1934
図 478. 出力フェーズでの 128bit データブロックの DMA 転送 1935
図 479. SAES ブロック図 1955
図 480. ECB の暗号化と復号化の原理 1957
図 481. CBC の暗号化と復号化の原理 1958
図 482. サスペンドモードの管理例 1962
図 483. ECB 暗号化 1963
図 484. ECB 復号 1963
図 485. CBC 暗号化 1964
図 486. CBC 復号化 1964
図 487. ECB/CBC 暗号化（モード 1） 1965
図 488. ECB/CBC 復号化（モード 3） 1966
図 489. ラップ･キーを使った動作 1968
図 490. 共有キーモードの使用 1970
図 491. データ･スワッピングに関する 128bit ･ブロックの構成 1973
図 492. キー保護メカニズム 1975
図 493. 入力フェーズでの 128bit ･データ･ブロックの DMA 転送 1976
図 494. 出力フェーズでの 128bit データブロックの DMA 転送 1977
図 495. HASH ブロック図 1997
図 496. メッセージデータスワッピング機能 1999
図 497. HASH サスペンド／レジュームメカニズム 2004
図 498. OTFDEC ブロック図 2020
図 499. SoC での OTFDEC の標準的な使用方法 2021
図 500. AES CTR 復号化フロー 2022

図 501. OTFDEC フロー制御の概要（デュアルバースト読出しリクエスト）.....2023
図 502. PKA のブロック図.....2043
図 503. 高機能制御タイマのブロック図.....2073
図 504. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図.....2080
図 505. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図.....2081
図 506. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....2082
図 507. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....2083
図 508. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....2083
図 509. 内部クロック分周比が N の場合のカウンタのタイミング図.....2084
図 510. ARPE=0 の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロードされない）.....2084
図 511. ARPE=1（TIMx_ARR はプリロード）の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロード）.....2085
図 512. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....2086
図 513. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....2087
図 514. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....2087
図 515. 内部クロック分周比が N の場合のカウンタのタイミング図.....2088
図 516. 繰返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図.....2088
図 517. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図.....2090
図 518. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....2090
図 519. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図.....2091
図 520. 内部クロック分周比が N の場合のカウンタのタイミング図.....2091
図 521. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図.....2092
図 522. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図.....2093
図 523. モードと TIMx_RCR レジスタの設定に応じた更新レートの例.....2094
図 524. 外部トリガ入力ブロック.....2095
図 525. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図.....2096
図 526. tim_ti2 外部クロックの接続例.....2096
図 527. 外部クロックモード 1 の制御回路.....2097
図 528. 外部トリガ入力ブロック.....2098
図 529. 外部クロックモード 2 の制御回路.....2099
図 530. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）.....2099
図 531. キャプチャ／比較チャネル 1 メイン回路.....2100
図 532. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3、および 4）の出力ステージ.....2101
図 533. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ.....2101
図 534. PWM 入力モードタイミング.....2103
図 535. 出力比較モード、tim_oc1 の反転.....2105
図 536. エッジアライン PWM 波形（ARR=8）.....2107
図 537. センターアライン PWM 波形（ARR=8）.....2108
図 538. ディザリング原理.....2109
図 539. ディザリングモードでのデータフォーマットとレジスタのコーディング.....2110
図 540. PWM 分解能 対 周波数.....2111
図 541. PWM ディザリングのパターン.....2112
図 542. センターアライン PWM モードでのデューティサイクルへのディザリングの影響.....2113
図 543. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成.....2115
図 544. チャネル 1 および 3 における組合せ PWM モード.....2116
図 545. 周期ごとの複数トリガパルスを持つ組合せ 3 相 PWM 信号.....2117
図 546. 対称のデッドタイム挿入のある相補出力.....2118
図 547. 非対称のデッドタイム.....2119
図 548. 負のパルスより長い遅延があるときのデッドタイムの波形.....2119
図 549. 正のパルスより長い遅延があるときのデッドタイムの波形.....2119
図 550. ブレークおよびブレーク 2 回路の概要.....2122
図 551. tim_brk（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作.....2124

図 552. tim_brk および tim_brk2 のアサート後の PWM 出力状態（OSSI=1）. 2125
図 553. tim_brk アサート後の PWM 出力状態（OSSI=0）. 2126
図 554. 出力先変更（tim_brk2 リクエストは表示なし）. 2127
図 555. tim_ocref_clr 入力選択マルチプレクサ 2128
図 556. TIMx tim_ocxref のクリア 2129
図 557. 6 ステップ生成 COM の例（OSSR=1）. 2130
図 558. ワンパルスモードの例. 2131
図 559. 再トリガ可能なワンパルスモード. 2132
図 560. パルス発生回路 2133
図 561. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成 2134
図 562. 同時トリガの場合の拡張されたパルス幅 2135
図 563. エンコーダインタフェースモードにおけるカウンタの動作例 2137
図 564. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例 2137
図 565. 直交エンコーダのカウントモード. 2138
図 566. 方向プラスクロックエンコーダモード 2139
図 567. 方向クロックエンコーダモード（CC1P = CC2P = 0）. 2139
図 568. 方向クロックエンコーダモード（CC1P = CC2P = 1）. 2140
図 569. インデックスゲーティングオプション 2141
図 570. ジッタのあるインデックス信号 2141
図 571. IPOS[1:0] = 11 の場合のインデックス生成. 2142
図 572. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値 2142
図 573. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値 2143
図 574. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値 2143
図 575. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11）. 2144
図 576. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07）. 2145
図 577. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01）. 2146
図 578. インデックスの方向検知 2146
図 579. FIDX ビット設定によって変わるカウンタリセット 2147
図 580. インデックスのブランキング 2148
図 581. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1）. 2148
図 582. 方向クロックモードでのインデックスの動作（IPOS[0] = 1）. 2149
図 583. 直交エンコードされた信号の状態図 2149
図 584. アップカウントでのエンコーダエラー検出 2150
図 585. ダウンカウントでのエンコーダエラー検出 2151
図 586. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）. 2152
図 587. 3 つの信号上のエッジ間の時間間隔の測定. 2153
図 588. ホールセンサインタフェースの例. 2155
図 589. リセットモードの制御回路 2156
図 590. ゲートモードの制御回路. 2157
図 591. トリガモードの制御回路. 2158
図 592. 外部クロックモード 2 ＋トリガモードの制御回路. 2159
図 593. 汎用タイマのブロック図. 2218
図 594. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 2224
図 595. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 2224
図 596. 内部クロック分周比が 1 の場合のカウンタのタイミング図. 2225
図 597. 内部クロック分周比が 2 の場合のカウンタのタイミング図. 2226
図 598. 内部クロック分周比が 4 の場合のカウンタのタイミング図. 2226
図 599. 内部クロック分周比が N の場合のカウンタのタイミング図 2227
図 600. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図. 2227
図 601. ARPE=1（TIMx_ARR はプリロードされる）のときの更新イベント時のカウンタのタイミング図. 2228
図 602. 内部クロック分周比が 1 の場合のカウンタのタイミング図. 2229

図 603. 内部クロック分周比が 2 の場合のカウンタのタイミング図 2229
図 604. 内部クロック分周比が 4 の場合のカウンタのタイミング図 2230
図 605. 内部クロック分周比が N の場合のカウンタのタイミング図 2230
図 606. 更新イベント時のカウンタのタイミング図 2231
図 607. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 2232
図 608. 内部クロック分周比が 2 の場合のカウンタのタイミング図 2233
図 609. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図 2233
図 610. 内部クロック分周比が N の場合のカウンタのタイミング図 2234
図 611. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図 2234
図 612. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 2235
図 613. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 2236
図 614. tim_ti2 外部クロックの接続例 2236
図 615. 外部クロックモード 1 の制御回路 2237
図 616. 外部トリガ入力ブロック 2238
図 617. 外部クロックモード 2 の制御回路 2239
図 618. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 2239
図 619. キャプチャ／比較チャネル 1 メイン回路 2240
図 620. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3、および 4）の出力ステージ 2240
図 621. PWM 入力モードタイミング 2243
図 622. 出力比較モード、tim_oc1 の反転 2244
図 623. エッジアライン PWM 波形（ARR=8） 2246
図 624. センターアライン PWM 波形（ARR=8） 2247
図 625. ディザリング原理 2248
図 626. ディザリングモードでのデータフォーマットとレジスタのコーディング 2249
図 627. PWM 分解能 対 周波数（16bit モード） 2250
図 628. PWM 分解能 対 周波数（32bit モード） 2250
図 629. PWM ディザリングのパターン 2251
図 630. センターアライン PWM モードでのデューティサイクルへのディザリングの影響 2252
図 631. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 2254
図 632. チャネル 1 および 3 における組み合わせ PWM モード 2255
図 633. OCREF_CLR 入力選択マルチプレクサ 2256
図 634. TIMx tim_ocxref のクリア 2256
図 635. ワンパルスモードの例 2257
図 636. 再トリガ可能なワンパルスモード 2259
図 637. パルス発生回路 2259
図 638. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成 2260
図 639. 同時トリガの場合の拡張されたパルス幅 2261
図 640. エンコーダインタフェースモードにおけるカウンタの動作例 2263
図 641. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例 2263
図 642. 直交エンコーダのカウントモード 2264
図 643. 方向プラスクロックエンコーダモード 2265
図 644. 方向クロックエンコーダモード（CC1P = CC2P = 0） 2265
図 645. 方向クロックエンコーダモード（CC1P = CC2P = 1） 2266
図 646. インデックスゲーティングオプション 2267
図 647. ジッタのあるインデックス信号 2267
図 648. IPOS[1:0] = 11 の場合のインデックス生成 2268
図 649. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値 2268
図 650. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値 2269
図 651. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値 2269
図 652. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11） 2270
図 653. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07） 2271
図 654. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01） 2272
図 655. インデックスの方向検知 2272

図 656. FIDX ビット設定によって変わるカウンタリセット 2273
図 657. インデックスのブランキング 2274
図 658. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1） 2274
図 659. 方向クロックモードでのインデックスの動作（IPOS[0] = 1） 2275
図 660. 直交エンコードされた信号の状態図 2275
図 661. アップカウントでのエンコーダエラー検出 2276
図 662. ダウンカウントでのエンコーダエラー検出 2277
図 663. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0） 2278
図 664. リセットモードの制御回路 2280
図 665. ゲートモードの制御回路 2281
図 666. トリガモードの制御回路 2281
図 667. 外部クロックモード 2 ＋トリガモードの制御回路 2283
図 668. マスタ／スレーブタイマの例 2283
図 669. チャネルが 1 つしかないタイマーによるマスタ／スレーブ接続の例 2284
図 670. TIM_mstr の tim_oc1ref による TIM_slv のゲート操作 2285
図 671. TIM_mstr の有効化による TIM_slv のゲート操作 2286
図 672. TIM_mstr の更新による TIM_slv のトリガ 2286
図 673. TIM_mstr の有効化による TIM_slv のトリガ 2287
図 674. TIM_mstr の tim_ti1 入力による TIM_mstr と TIM_slv のトリガ 2288
図 675. TIM15 ブロック図 2325
図 676. TIM16/TIM17 ブロック図 2326
図 677. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 2330
図 678. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 2331
図 679. 内部クロック分周比が 1 の場合のカウンタのタイミング図 2332
図 680. 内部クロック分周比が 2 の場合のカウンタのタイミング図 2333
図 681. 内部クロック分周比が 4 の場合のカウンタのタイミング図 2333
図 682. 内部クロック分周比が N の場合のカウンタのタイミング図 2334
図 683. ARPE=0 の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARR はプリロードされない） 2334
図 684. ARPE=1（TIMx_ARR はプリロード）の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARR はプリロード） 2335
図 685. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 2336
図 686. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 2337
図 687. tim_ti2 外部クロックの接続例 2337
図 688. 外部クロックモード 1 の制御回路 2338
図 689. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 2339
図 690. キャプチャ／比較チャネル 1 メイン回路 2339
図 691. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 2340
図 692. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ 2340
図 693. PWM 入力モードタイミング 2343
図 694. 出力比較モード、tim_oc1 の反転 2345
図 695. エッジアライン PWM 波形（ARR=8） 2346
図 696. ディザリング原理 2347
図 697. ディザリングモードでのデータフォーマットとレジスタのコーディング 2347
図 698. PWM 分解能 対 周波数 2348
図 699. PWM ディザリングのパターン 2349
図 700. チャネル 1 および 2 における組合せ PWM モード 2351
図 701. 対称のデッドタイム挿入のある相補出力。 2352
図 702. 非対称のデッドタイム 2353
図 703. 負のパルスより長い遅延があるときのデッドタイムの波形 2353
図 704. 正のパルスより長い遅延があるときのデッドタイムの波形 2353
図 705. ブレーク回路の概要 2355
図 706. tim_brk でのブレークイベントに対する出力の動作 2357

図 707. 出力先変更 2359
図 708. tim_ocref_clr 入力選択マルチプレクサ 2359
図 709. 6 ステップ生成 COM の例（OSSR=1） 2360
図 710. ワンパルスモードの例 2361
図 711. 再トリガ可能なワンパルスモード 2363
図 712. 2 つの信号上のエッジ間の時間間隔の測定 2363
図 713. リセットモードの制御回路 2364
図 714. ゲートモードの制御回路 2365
図 715. トリガモードの制御回路 2366
図 716. 基本タイマブロック図 2430
図 717. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 2431
図 718. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 2432
図 719. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 2433
図 720. 内部クロック分周比が 1 の場合のカウンタのタイミング図 2434
図 721. 内部クロック分周比が 2 の場合のカウンタのタイミング図 2434
図 722. 内部クロック分周比が 4 の場合のカウンタのタイミング図 2435
図 723. 内部クロック分周比が N の場合のカウンタのタイミング図 2435
図 724. ARPE = 0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 2436
図 725. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 2437
図 726. ディザリング原理 2438
図 727. ディザリングモードでのデータフォーマットとレジスタのコーディング 2438
図 728. FCnt 分解能 対 周波数 2439
図 729. PWM ディザリングのパターン 2439
図 730. LPTIM1/2/3 タイマブロック図 2451
図 731. LPTIM4 タイマブロック図 2452
図 732. グリッチフィルタのタイミング図 2456
図 733. LPTIM 出力波形、シングルカウントモードの設定（繰返しレジスタの内容が 0 ではない場合（かつ PRELOAD = 1）） 2458
図 734. LPTIM 出力波形、シングルカウントモードの設定、セットワンスモードのアクティブ化（WAVE ビットがセットされている場合） 2459
図 735. LPTIM 出力波形、連続カウントモードの設定 2459
図 736. 波形生成 2461
図 737. エンコーダモードのカウントシーケンス 2464
図 738. 繰返しレジスタ LPTIM_RCR が 0 ではない場合の連続カウントモード（PRELOAD = 1） 2465
図 739. キャプチャ／比較入力ステージ（チャネル 1） 2466
図 740. キャプチャ／比較出力ステージ（チャネル 1） 2466
図 741. エッジアライン PWM モード（PRELOAD = 1） 2469
図 742. エッジアライン PWM 波形（ARR = 8、CCxP = 0） 2469
図 743. 即時更新とプリロード更新の PWM モード 2470
図 744. GFXTIM ブロック図 2502
図 745. クロックジェネレータ 2504
図 746. スタンドアロンの波形 2506
図 747. スタンドアロンでアクティブなカウンタおよび信号 2506
図 748. 外部 HSYNC および VSYNC の波形 2506
図 749. 外部 HSYNC のみの波形 2507
図 750. 外部 HSYNC のみのアクティブなカウンタおよび信号 2507
図 751. 外部 VSYNC のみの波形 2507
図 752. 外部 VSYNC のみのアクティブなカウンタ 2508
図 753. 外部 VSYNC のみの場合のプリスケーリング 2508
図 754. 外部 CSYNC のみの波形 2508
図 755. 外部 CSYNC のみのアクティブなカウンタおよび信号 2509

図 756. 外部 CSYNC のみの場合のプリスケーリング 2509
図 757. ティアリング効果の構成 2511
図 758. ウォッチドッグ·タイマ 2512
図 759. TIM16 および TIM17 との IRTIM 内部ハードウェア接続 2539
図 760. 独立型ウォッチドッグのブロック図 2541
図 761. タイムアウトによるリセットのタイミング 2543
図 762. 許可されていない領域でのリフレッシュによるリセットのタイミング 2544
図 763. 独立型ウォッチドッグ割込みのタイミング図 2547
図 764. ウォッチドッグのブロック図 2555
図 765. ウィンドウ型ウォッチドッグのタイミング図 2557
図 766. RTC のブロック図 2563
図 767. TAMP ブロック図 2616
図 768. バックアップレジスタ保護ゾーン 2620
図 769. アクティブタンパフィルタ 2627
図 770. I^2C ブロック図 2663
図 771. I2C バスプロトコル 2666
図 772. セットアップおよびホールドタイミング 2668
図 773. I^2C 初期化フロー 2670
図 774. データ受信 2671
図 775. データ送信 2672
図 776. スレーブ初期化フロー 2675
図 777. I^2C スレーブトランスミッタの転送シーケンスフロー（NOSTRETCH = 0） 2677
図 778. I^2C スレーブトランスミッタの転送シーケンスフロー（NOSTRETCH = 1） 2678
図 779. I^2C スレーブトランスミッタの転送バス図（必須イベントのみ） 2679
図 780. スレーブレシーバの転送シーケンスフロー（NOSTRETCH = 0） 2680
図 781. スレーブレシーバの転送シーケンスフロー（NOSTRETCH = 1） 2681
図 782. I^2C スレーブレシーバの転送バス図（必須イベントのみ） 2681
図 783. マスタクロック生成 2683
図 784. マスタ初期化フロー 2685
図 785. HEAD10R = 0 のときの 10bit アドレス読出しアクセス 2685
図 786. HEAD10R = 1 のときの 10bit アドレス読出しアクセス 2686
図 787. N ≤ 255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフロー 2687
図 788. N > 255 バイトの場合の I^2C マスタトランスミッタの転送シーケンスフロー 2688
図 789. I^2C マスタトランスミッタの転送バス図（必須イベントのみ） 2689
図 790. N ≤ 255 バイトの場合の I^2C マスタレシーバの転送シーケンスフロー 2691
図 791. N > 255 バイトの場合の I^2C マスタレシーバの転送シーケンスフロー 2692
図 792. I^2C マスタレシーバの転送バス図（必須イベントのみ） 2693
図 793. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔 2697
図 794. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフロー 2700
図 795. SMBus スレーブトランスミッタの転送バス図（SBC = 1） 2701
図 796. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフロー 2702
図 797. SMBus スレーブレシーバのバス転送図（SBC = 1） 2703
図 798. SMBus マスタトランスミッタのバス転送図 2704
図 799. SMBus マスタレシーバのバス転送図 2706
図 800. USART ブロック図 2733
図 801. ワード長のプログラミング 2737
図 802. 設定可能なストップビット 2739
図 803. 送信時の TC/TXE の動作 2741
図 804. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出 2742
図 805. usart_ker_ck クロック分周回路のブロック図 2745
図 806. データサンプリング（16 倍のオーバーサンプリング） 2746
図 807. データサンプリング（8 倍のオーバーサンプリング） 2747
図 808. アイドルライン検出を使用したミュートモード 2753

図 809. アドレスマーク検出を使用したミュートモード 2754
図 810. LIN モードでのブレーク検出（11bit ブレーク長、LBDL=1） 2757
図 811. LIN モードでのブレーク検出とフレーミングエラー検出 2758
図 812. USART の同期マスタ送信の例 2759
図 813. 同期マスタモードでの USART データクロックタイミング図（M ビット = 00） 2759
図 814. 同期マスタモードでの USART データクロックタイミング図（M ビット = 01） 2760
図 815. 同期スレーブモードでの USART データクロックタイミング図（M ビット = 00） 2761
図 816. ISO 7816-3 非同期プロトコル 2763
図 817. ストップビット 1.5 個を使用したパリティエラー検出 2765
図 818. IrDA SIR ENDEC ブロック図 2768
図 819. IrDA データ変調（3/16）- 通常モード 2769
図 820. DMA を使用した送信 2770
図 821. DMA を使用した受信 2771
図 822. 2 つの USART 間のハードウェアフロー制御 2772
図 823. RS232 RTS フロー制御 2772
図 824. RS232 CTS フロー制御 2773
図 825. LPUART ブロック図 2817
図 826. LPUART ワード長のプログラミング 2821
図 827. 設定可能なストップビット 2823
図 828. 送信時の TC/TXE の動作 2825
図 829. lpuart_ker_ck クロック分周回路のブロック図 2828
図 830. アイドルライン検出を使用したミュートモード 2833
図 831. アドレスマーク検出を使用したミュートモード 2834
図 832. DMA を使用した送信 2836
図 833. DMA を使用した受信 2837
図 834. 2 つの LPUART 間のハードウェアフロー制御 2838
図 835. RS232 RTS フロー制御 2838
図 836. RS232 CTS フロー制御 2839
図 837. SPI のブロック図 2870
図 838. 全二重シングルマスタ／シングルスレーブアプリケーション 2874
図 839. 半二重シングルマスタ／シングルスレーブアプリケーション 2874
図 840. 単方向シングルマスタ／シングルスレーブのアプリケーション
（送信専用モードのマスタ / 受信専用モードのスレーブ） 2875
図 841. スター接続形態で接続されたマスタと 3 つの独立したスレーブ 2876
図 842. マルチマスタアプリケーション 2877
図 843. SS 制御ロジックの仕組み 2879
図 844. データフロータイミング制御（SSOE = 1、SSOM = 0、SSM = 0） 2879
図 845. SS によるデータ間パルスのインターリーブ（SSOE = 1、SSOM = 1、SSM = 0） 2880
図 846. データクロックのタイミング図 2882
図 847. データサイズが 8、16、または 32bit と等しくない場合のデータ配置 2883
図 848. 全機能がセットされたインスタンスで、FIFO での送受信用のデータのパッキング 2891
図 849. TI モードでの転送 2894
図 850. スレーブ検出中のアンダーラン状態に関するオプションの設定 2896
図 851. SAI 機能ブロック図 2921
図 852. オーディオフレーム 2925
図 853. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1） 2927
図 854. FS の機能はフレーム開始（FSDEF = 0） 2928
図 855. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合） 2929
図 856. 先頭ビットオフセット 2929
図 857. オーディオブロッククロックジェネレータの概要 2931
図 858. PDM 標準接続とタイミング 2935
図 859. PDM インタフェースの詳細ブロック図 2936
図 860. 起動シーケンス 2937

図 861. TDM の SAI_ADR フォーマット（32bit スロット幅）. 2938
図 862. TDM の SAI_ADR フォーマット（16bit スロット幅）. 2939
図 863. TDM の SAI_ADR フォーマット（8bit スロット幅）. 2940
図 864. AC’97 オーディオフレーム 2943
図 865. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC’97 設定の代表例 2944
図 866. SPDIF フォーマット 2945
図 867. SAI_xDR レジスタの順序 2946
図 868. SAI オーディオブロックのデータコンパンディングハードウェア. 2949
図 869. 非アクティブスロットにおける SD 出力ラインのトライステート化. 2950
図 870. I2S のようなプロトコルにおける出力データラインのトライステート化 2951
図 871. オーバーラン検出エラー. 2952
図 872. FIFO アンダーランイベント 2952
図 873. CAN サブシステム. 2988
図 874. FDCAN ブロック図 2990
図 875. ビットタイミング 2991
図 876. トランシーバ遅延測定. 2996
図 877. バスモニタモードでのピン制御 2997
図 878. ループバックモードでのピン制御. 2999
図 879. メッセージ RAM 設定 3000
図 880. 標準メッセージ ID フィルタパス. 3003
図 881. 拡張メッセージ ID フィルタパス. 3004
図 882. USB ペリフェラルのブロック図 3051
図 883. パケットバッファ領域とバッファ記述テーブルの位置の例 3057
図 884. OTG_FS フルスピードブロック図 3099
図 885. OTG_FS A-B デバイス接続. 3102
図 886. OTG_FS ペリフェラル専用接続 3104
図 887. OTG_FS ホスト専用接続 3108
図 888. SOF 配線（TIM および ITR1 接続への SOF トリガ出力). 3112
図 889. OTG_HFIR の動的更新（RLDCTRL=1）. 3114
図 890. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング. 3115
図 891. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 3116
図 892. 割込みの階層 3120
図 893. 送信 FIFO 書込みタスク 3203
図 894. 受信 FIFO 読出しタスク 3204
図 895. 通常のバルク／制御 OUT/SETUP. 3205
図 896. バルク／制御 IN トランザクション. 3209
図 897. 通常割込み OUT 3211
図 898. 通常割込み IN 3216
図 899. アイソクロナス OUT トランザクション 3218
図 900. アイソクロナス IN トランザクション 3221
図 901. 受信 FIFO パケット読出し 3225
図 902. SETUP パケットの処理. 3227
図 903. バルク OUT トランザクション 3234
図 904. TRDT 最大タイミングのケース. 3243
図 905. A デバイスの SRP. 3244
図 906. B デバイスの SRP. 3245
図 907. A デバイスの HNP. 3246
図 908. B デバイスの HNP. 3247
図 909. OTG_HS ハイスピードブロック図 3252
図 910. OTG_HS A-B デバイス接続. 3254
図 911. OTG_HS ペリフェラル専用接続. 3255
図 912. OTG_HS ホスト専用接続 3259

図 913. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力）.....3263
図 914. OTG_HFIR の動的更新（RLDCTRL=1）.....3265
図 915. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング.....3266
図 916. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 3267
図 917. 割込みの階層.....3271
図 918. 送信 FIFO 書込みタスク 3365
図 919. 受信 FIFO 読出しタスク 3366
図 920. 通常のバルク／コントロール OUT/SETUP.....3368
図 921. バルク／コントロール IN トランザクション.....3372
図 922. 通常のインタラプト OUT 3374
図 923. 通常のインタラプト IN 3379
図 924. アイソクロナス OUT トランザクション.....3381
図 925. アイソクロナス IN トランザクション.....3384
図 926. 通常のバルク／コントロール OUT/SETUP トランザクション - DMA 3386
図 927. 通常のバルク／コントロール IN トランザクション - DMA.....3388
図 928. 通常のインタラプト OUT トランザクション - DMA モード.....3390
図 929. 通常のインタラプト IN トランザクション - DMA モード.....3391
図 930. 通常のアイソクロナス OUT/IN トランザクション - DMA モード.....3392
図 931. 通常のアイソクロナス IN トランザクション - DMA モード.....3393
図 932. 受信 FIFO パケット読出し 3399
図 933. SETUP パケットの処理.....3401
図 934. バルク OUT トランザクション 3408
図 935. TRDT 最大タイミングのケース.....3417
図 936. UCPD のブロック図 3420
図 937. クロック分周とタイミング要素 3422
図 938. K コード送信.....3424
図 939. さまざまなデータサイズの送信順.....3426
図 940. パケットフォーマット.....3426
図 941. Hard Reset のラインフォーマット 3427
図 942. Cable Reset のラインフォーマット 3427
図 943. BIST テストデータフレーム 3428
図 944. BIST キャリアモード 2 フレーム.....3429
図 945. UCPD BMC トランスミッタアーキテクチャ 3429
図 946. UCPD BMC レシーバのアーキテクチャ.....3431
図 947. デバッグサポートインフラストラクチャのブロック図 3459
図 948. JTAG TAP ステートマシン.....3463
図 949. CoreSight トポロジ.....3481
図 950. トレースポートインタフェースユニット（TPIU）.....3559
図 951. 組み込みクロストリガ.....3571

1 このマニュアルにおける表記の規則

1.1 一般情報

STM32U5シリーズデバイスには、Arm® TrustZone®搭載 Arm®(a) Cortex®-M33コアがあります。

1.2 レジスタに関する略記

レジスタの説明では、次の略記(b) が使用されます。

読出し/書込み（rw）	このビットは、ソフトウェアによる読出しと書込みが可能です。
読出し専用（r）	このビットは、ソフトウェアによる読出しのみが可能です。
書込み専用（w）	このビットは、ソフトウェアによる書込みのみが可能です。このビットを読み出すと、リセット値が返されます。
読出し/クリア書込み0（rc_w0）	このビットは、ソフトウェアによって読み出すことができ、“0”を書き込むことによってクリアできます。“1”を書き込んでも、ビットの値は変化しません。
読出し/クリア書込み1（rc_w1）	このビットは、ソフトウェアによって読み出すことができ、“1”を書き込むことによってクリアできます。“0”を書き込んでも、ビットの値は変化しません。
読出し/クリア書込み（rc_w）	このビットは、ソフトウェアによって読み出すことができ、レジスタへの書込みによってクリアできます。このビットに書き込まれた値は重要ではありません。
読出し/読出しによるクリア（rc_r）	このビットは、ソフトウェアによって読み出すことができます。このビットを読み出すと、自動的に“0”にクリアされます。このビットを書き込んでも、ビットの値は変化しません。
読出し/読出しによるセット（rs_r）	このビットは、ソフトウェアによって読み出すことができます。このビットを読み出すと、自動的に“1”にセットされます。このビットを書き込んでも、ビットの値は変化しません。
読出し/セット（rs）	このビットは、ソフトウェアによって読出しとセットができます。“0”を書き込んでも、ビットの値は変化しません。
読出し/1回だけ書込み（rwo）	このビットは、ソフトウェアによって1回だけ書き込むことができ、いつでも読み出すことができます。1回リセットするだけで、このビットをそのリセット値に戻すことができます。
反転（t）	このビットは、ソフトウェアによって“1”を書き込むことで反転できます。0を書き込んでも、ビットの値は変化しません。
読出し専用書込みトリガ（rt_w1）	このビットは、ソフトウェアによって読み出すことができます。“1”を書き込むと、イベントがトリガされますが、ビットの値は変化しません。
予約済み（Res.）	予約済みビットであり、リセット値に保持する必要があります。

a. Armは、米国内およびその他の地域にあるArm Limited（またはその子会社）の登録商標です。

b. これはSTMicroelectronicsマイクロコントローラに適用されるすべての略語の包括的リストですが、本書では使用されていないものもあります。

1.3 用語

このセクションでは、本書で用いられる略語についての定義の概要を掲載しています。

- **ワード**：32bit長のデータ。
- **ハーフワード**：16bit長のデータ。
- **バイト**：8bit長のデータ。
- **AHB**：アドバンストハイパフォーマンスバス。
- **APB**：アドバンストペリフェラルバス。

1.4 使用可能なペリフェラル

特定のデバイスでの使用可能なペリフェラルについては、表 6：メモリマップとペリフェラルのレジスタ境界アドレスを参照してください。

すべての販売タイプで使用可能なペリフェラルとその型番（製品ラインとパッケージにより異なる）について詳しくは、デバイスのデータシートを参照してください。

2 メモリとバスのアーキテクチャ

2.1 システムアーキテクチャ

STM32U5シリーズアーキテクチャは、内蔵Flashメモリに直接アクセスできる命令キャッシュによって実行に関して最適化されたArm Cortex-M33コアに依存しています。

このアーキテクチャは、次の表に示すような、マスタおよびスレーブを相互接続する32bitマルチレイヤAHBバスマトリックスも備えています。

表1.STM32U5シリーズへのマスタの実装

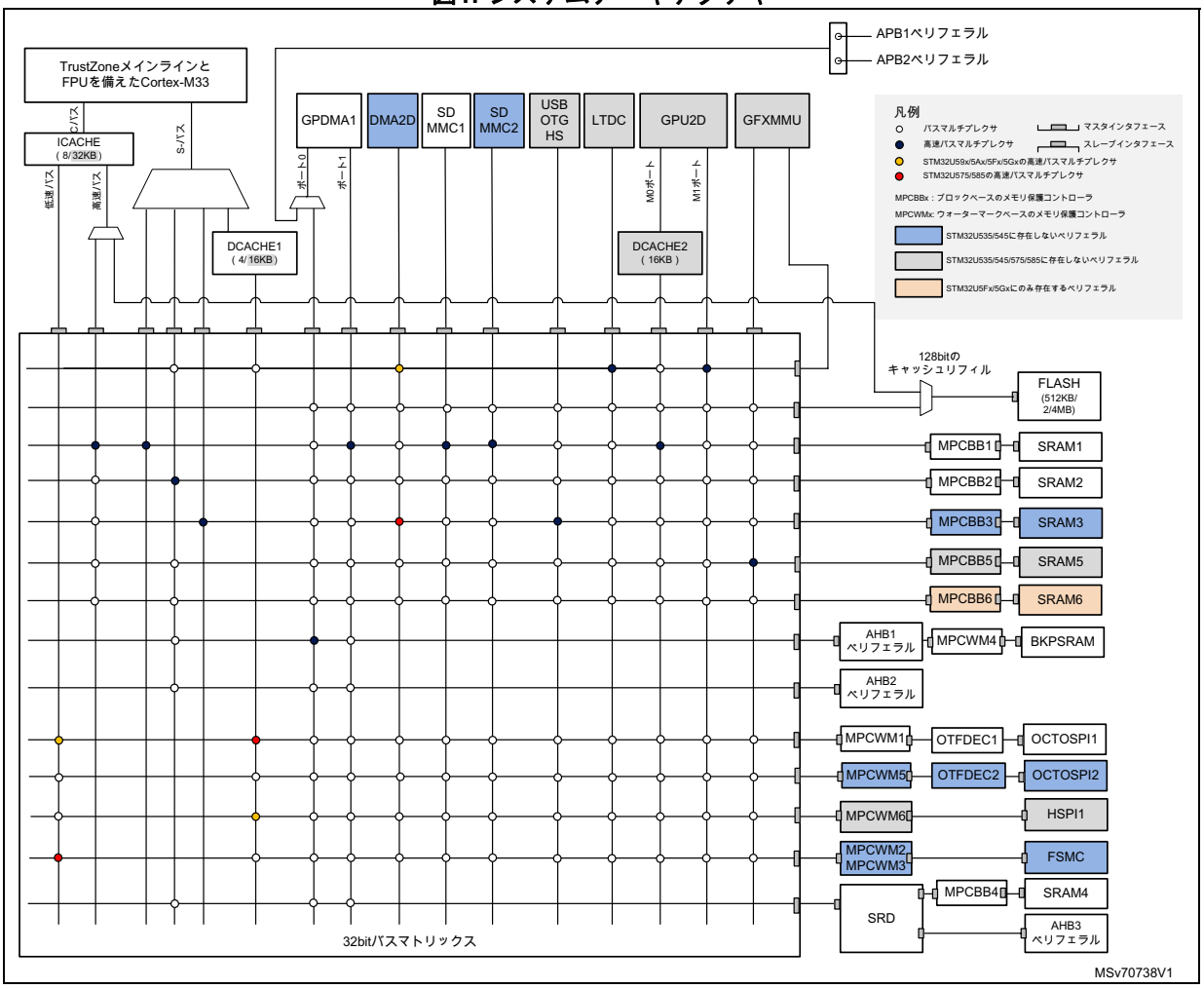
マスタ	コメント	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
Cortex-M33高速Cバス	ICACHEを経由したCortex-M33（Arm TrustZone®メインラインおよび FPU搭載）の内部SRAMおよびFlashメモリへの接続。	X	X	X	X
Cortex-M33低速Cバス	ICACHEを経由したCortex-M33（Arm TrustZoneメインラインおよび FPU搭載）の外部メモリへの接続。	X	X	X	X
Cortex-M33 Sバス	遅延なしのCortex-M33（Arm TrustZoneメインラインおよびFPU搭載）の内部SRAMへの接続。	2マスタ	3マスタ		
	DCACHE2を経由したCortex-M33（Arm TrustZoneメインラインおよび FPU搭載）の外部メモリへの接続。	X	X	X	X
GPDMA1	-	2マスタ			
DMA2D	-	-	X	X	X
SDMMC1	-	X	X	X	X
SDMMC2	-	-	X	X	X
LTDC	-	-	-	X	X
GPU2D M1ポート	グラフィック処理装置M1ポート	-	-	X	X
GPU2D M0ポート	DCACHE2を経由したグラフィック処理装置M0ポート	-	-	X	X
GFXMMUマスタポート	-	-	-	X	X
OTG_HS	-	-	-	X	X

表2.STM32U5シリーズへのスレーブの実装

スレーブ	コメント	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
内部Flashメモリ	-	X	X	X	X
内蔵SRAM1	-	X	X	X	X
内蔵SRAM2	-	X	X	X	X
内蔵SRAM3	-	-	X	X	X
内蔵SRAM5	-	-	-	X	X
内蔵SRAM6	-	-	-	-	X
AHB1	ペリフェラル、AHB－APBブリッジを含む2KBのBKPSRAMおよびAPBペリフェラル（APB1およびAPB2に接続）	X	X	X	X
AHB2	ペリフェラル	X	X	X	X
FSMC（フレキシブルスタティックメモリコントローラ）	-	-	X	X	X
OCTOSPI1	-	X	X	X	X
OCTOSPI2	-	-	X	X	X
HSPI1	-	-	-	X	X
GFXMMUスレーブポート	-	-	-	X	X
AHB3	AHB－APBブリッジを含むSRDペリフェラルおよびSRAM4、およびAPBペリフェラル（APB3に接続）	X	X	X	X

複数の高速ペリフェラルが同時動作中であっても、バスマトリクスによって、マスタからスレーブへのアクセスが可能になるとともに、同時アクセスや効率的な動作を行うことが可能になっています。このアーキテクチャを次の図に示します。

図1. システムアーキテクチャ



2.1.1 高速Cバス

このバスによって、Cortex-M33コアのCバスが命令キャッシュ経由で内部Flashメモリとバスマトリックスに接続されます。このバスは、命令フェッチおよびコード領域にマップされた内部メモリへのデータアクセスに使用されます。このバスは、内部Flashメモリと内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）を対象としています。

SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6は、連続マッピングにより、このバスでアクセス可能です。

2.1.2 低速Cバス

このバスによって、Cortex-M33コアのCバスが命令キャッシュ経由でバスマトリックスに接続されます。このバスは、命令フェッチおよびコード領域にマップされた外部メモリへのデータアクセスに使用されます。このバスは、外部メモリ（FSMC、HSPI1、およびOCTOSPI）を対象としています。

2.1.3 S-バス

このバスによって、Cortex-M33コアのシステムバスがバスマトリックスに接続されます。コアがこのバスを使用してペリフェラルまたはSRAM領域内のデータにアクセスします。このバスは、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM4、SRAM5、SRAM6、およびBKPSRAM）、APB1およびAPB2ペリフェラルを含むAHB1ペリフェラル、AHB2ペリフェラル、およびSRDペリフェラルを対象としています。

SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6は、連続マッピングにより、このバスでアクセス可能です。

注： **バスマトリックスがSRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6にアクセスするときは、ゼロ遅延です。**

2.1.4 DCACHE Sバス

このバスによって、Cortex-M33コアのシステムバスがデータキャッシュ経由でバスマトリックスに接続されます。このバスは、命令フェッチおよびデータ領域にマップされた外部メモリへのデータアクセスに使用されます。このバスは、外部メモリ（FSMC、HSPI1、GFXMMU、およびOCTOSPI）を対象としています。

注： **このバス経由で命令をフェッチするのは、低速Cバス経由で命令をフェッチするより非効率的です。**

2.1.5 GPDMAバス

これらのバスによって、GPDMAの2つのAHBマスタインタフェースがバスマトリックスに接続されます。これらのバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM4、SRAM5、SRAM6、およびBKPSRAM）、APB1およびAPB2ペリフェラルを含むAHB1ペリフェラル、AHB2ペリフェラル、SRDペリフェラル、およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリをターゲットとします。

2.1.6 OTG_HSバス

このバスによって、OTG_HSマスタインタフェースがバスマトリックスに接続されます。このバスは、OTG_HSによってデータをメモリからロードするか、データをメモリに保存するためにのみ使用されます。このバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリなどのデータメモリを対象としています。

2.1.7 LTDCバス

このバスによって、LTDCマスタインタフェースがバスマトリックスに接続されます。このバスは、データをメモリからロードするためにのみ使用されます。このバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリなどのデータメモリに加え、GFXMMUを対象としています。

2.1.8 GPU2Dバス

これらのバスによって、GPU2Dマスタインタフェースがバスマトリックスに接続されます。これらのバスは、GPU2Dによってデータをメモリからロードするか、データをメモリに保存するためにのみ使用されます。これらのバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリなどのデータメモリに加え、GFXMMUを対象としています。性能向上のため、GPU2D M0バスには16KBのデータキャッシュ（DCACHE2）があります。

2.1.9 GFXMMUバス

このバスによって、GFXMMUマスタインタフェースがバスマトリックスに接続されます。このバスは、GFXMMUによってデータをメモリからロードするか、データをメモリに保存するためにのみ使用されます。このバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリなどのデータメモリを対象としています。GFXMMUには、グラフィカル·ペリフェラル·マスタ·バスがアクセスするスレーブバス接続もあります。

2.1.10 SDMMC1およびSDMMC2コントローラDMAバス

これらのバスによって、SDMMC1およびSDMMC2 DMAマスタインタフェースがバスマトリックスに接続されます。これらのバスは、SDMMC1およびSDMMC2 DMAによってデータをメモリからロードするか、データをメモリに保存するためにのみ使用されます。これらのバスは、内部Flashメモリ、内部SRAM（SRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6）およびFSMC、HSPI1、またはOCTOSPIを経由した外部メモリなどのデータメモリを対象としています。

2.1.11 バスマトリックス

このバスマトリックスによって、マスタ間のアクセス調停を管理します。調停には、ラウンドロビン方式を使用します。このバスマトリックスは、各マスタを任意のスレーブに遅延なく接続するための高速バスマルチプレクサを備えています（図1 を参照）。同じマスタの場合、他のスレーブでは、新しいアクセスごとに少なくとも1サイクルの遅延が発生します。

2.1.12 AHB/APBブリッジ

3つのAHB/APBブリッジがAHBバスとAPBバスの間の完全同期接続を提供するため、ペリフェラル周波数の柔軟な選択が可能です。

これらのブリッジに接続されたペリフェラルのアドレスマッピングについては、セクション 2.3.2: メモリマップとレジスタ境界アドレスを参照してください。

各デバイスのリセット後、すべてのペリフェラルクロックは無効になります（内部SRAMおよびFlashメモリインタフェースは除きます）。ペリフェラルを使用する前に、RCC_AHBxENR および RCC_APBxENRレジスタでクロックを有効にする必要があります。

注： **APBレジスタに8bitまたは16bitアクセスが行われるときには、アクセスは32bitアクセスに変換されます。すなわち、ブリッジが8bitまたは16bitのデータを複製して、32bitのベクタを供給します。**

2.1.13 SmartRunドメイン（SRD）

SRDアーキテクチャは、STOP2以上で動作する低電力モード中の自律動作を可能にするDMAに依存しています。

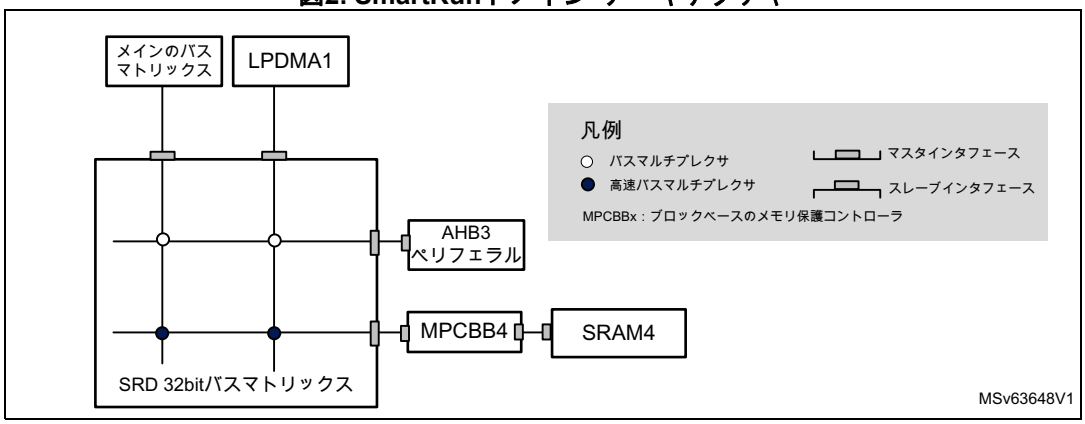
このアーキテクチャは、以下を相互接続する32bitAHBバスマトリックスも備えています。

- 2つのマスタ：
  - メインのAHBバスマトリックス
  - LPDMA1（1つのマスタポートを備えた低電力DMA）
- 2つのスレーブ：
  - APB3に接続されたAHB－APBブリッジを含むAHB3ペリフェラル
  - 内部SRAM4

注： **SRAM4は、LPDMA1によってアクセス可能な唯一のSRAMです。**

このアーキテクチャを次の図に示します。

図2. SmartRunドメイン·アーキテクチャ



2.2 Arm TrustZoneセキュリティアーキテクチャ

セキュリティアーキテクチャは、Armv8-Mメインライン拡張を備えたArm TrustZoneに基づきます。

TrustZoneセキュリティはFlash_OPTRレジスタのTZENオプションビットで有効化されます。

TrustZoneが有効なとき、SAU（security-attribution　unit：セキュリティ属性ユニット）とIDAU（implementation-defined-attribution unit：実装定義属性ユニット）は、セキュア状態と非セキュア状態に基づいてアクセス許可を定義します。

- SAU：セキュリティ属性には、最大8つのSAU設定可能領域が利用できます。
- IDAU：非セキュアまたは非セキュア呼出し可能属性として、1番目のメモリパーティションを提供します。IDAUメモリマップパーティションは設定することができず、ハードウェア実装により固定されます（セクション 2.3.2：メモリマップとレジスタ境界アドレスの図3から図6を参照）。そして、SAUセキュリティ属性からの結果と組み合わされ、より高度なセキュリティ状態が選択されます。

IDAUセキュリティ属性に基づいて、Flashメモリ、システムSRAM、およびペリフェラルのメモリ空間は、セキュア状態と非セキュア状態についてエイリアスが2回作成されます。外部のメモリ空間にはエイリアスは作成されません。

次の表に、IDAU領域に基づく典型的な8つのSAU領域マッピングの例を示します。必要に応じて、外部メモリ用にセキュア領域、非セキュア領域、またはNSC領域を分割し、選択することができます。

表3.SAU設定可能領域に対するメモリマップセキュリティ属性の例

領域の説明	アドレス範囲	IDAU セキュリティ属性	SAUセキュリティ 属性の標準設定	最終的な セキュリティ属性
コード - 外部メモリ	0x0000_0000：0x07FF_FFFF	非セキュア	セキュア、非セキュア、またはNSC(1)	セキュア、非セキュア、またはNSC
コード - FlashメモリおよびSRAM	0x0800_0000：0x0BFF_FFFF	非セキュア	非セキュア	非セキュア
	0x0C00_0000：0x0FFF_FFFF	NSC	セキュア/NSC	セキュア/NSC

表3.SAU設定可能領域に対するメモリマップセキュリティ属性の例 （続き）

領域の説明	アドレス範囲	IDAU セキュリティ属性	SAUセキュリティ 属性の標準設定	最終的な セキュリティ属性
コード - 外部メモリ	0x1000_0000 : 0x17FF_FFFF	非セキュア	非セキュア	
	0x1800_0000 : 0x1FFF_FFFF			
SRAM	0x2000_0000 : 0x2FFF_FFFFF	非セキュア		
	0x3000_0000 : 0x3FFF_FFFFF	NSC	セキュア/NSC	セキュア/NSC
ペリフェラル	0x4000_0000 : 0x4FFF_FFFFF	非セキュア	非セキュア	非セキュア
	0x5000_0000 : 0x5FFF_FFFFF	NSC	セキュア/NSC	セキュア/NSC
外部メモリ	0x6000_0000 : 0xDFFF_FFFF	非セキュア	セキュア、非セキュア、またはNSC	セキュア、非セキュア、またはNSC

1. NSC = 非セキュア呼出し可能

2.2.1 デフォルトのArm TrustZoneセキュリティ状態

TrustZoneセキュリティがFLASH_OPTRのTZENオプションビットで有効化される場合、各システムのデフォルトのセキュリティ状態は以下のとおりです。

- CPU：
  - Cortex-M33は、リセット後はセキュアな状態です。ブートアドレスは、セキュアなアドレスである必要があります。
- メモリマップ：
  - SAUは、リセット後は完全にセキュアです。そのため、すべてのメモリマップは完全にセキュアです。セキュリティ属性には、最大8つのSAU設定可能領域が利用できます。
- Flashメモリ：
  - Flashメモリのセキュリティ領域は、ウォーターマークユーザオプションによって定義されます。
  - Flashのブロックベースのセキュリティ属性は、リセット後は非セキュアです。
- SRAM：
  - すべてのSRAMはリセット後はセキュアです。MPCBBx（block-based memory protection controller：ブロックベースのメモリ保護コントローラ）はセキュアです。
- 外部メモリ：
  - FSMC、HSPI1、およびOCTOSPIバンクは、リセット後はセキュアです。MPCWMx（watermark-based memory protection controller：ウォーターマークベースのメモリ保護コントローラ）はセキュアです。
- ペリフェラル（セキュリティ保護可能なペリフェラルおよびTrustZone対応ペリフェラルのリストについては、表4および表5を参照）
  - セキュリティ保護可能なペリフェラルは、リセット後は非セキュアです。
  - TrustZoneを認識するペリフェラルは、リセット後は非セキュアです。それらのセキュア設定レジスタはセキュアです。
- すべてのGPIOはリセット後はセキュアです。
- 割込み：
  - NVIC：すべての割込みはリセット後はセキュアです。NVIC はセキュアな状態と非セキュアな状態でバンク切替えを行います。
  - TZIC：すべての不正アクセス割込みはリセット後に無効になります（GTZCのTrustZoneシステムアーキテクチャを参照）。

2.2.2 Arm TrustZoneペリフェラルの分類

TrustZoneセキュリティが有効な場合、ペリフェラルは、次のように、セキュリティ保護可能タイプまたはTrustZone対応タイプのいずれかです。

- セキュリティ保護可能：セキュリティ特性を定義するために、TZSCコントローラから制御されるAHB/APBファイアウォールゲートによって保護されるペリフェラル
- TrustZone対応：AHBまたはAPBバスに直接接続され、セキュアなレジスタのサブセットなど、特定のTrustZone動作を実装するペリフェラル。

詳細については、GTZCのTrustZoneシステムアーキテクチャを参照してください。

次の表は、システム内のセキュリティ保護可能ペリフェラルとTrustZone対応ペリフェラルのリストです。

表4.TZSCによってセキュリティ保護可能なペリフェラル

バス	ペリフェラル	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB3	ADF1	X	X	X	X
	DAC1	X	X	X	X
	ADC4	X	X	X	X
AHB2	OCTOSPI2レジスタ	-	X	X	X
	OCTOSPI1レジスタ	X	X	X	X
	HSPI1レジスタ	-	-	X	X
	FSMCレジスタ	-	X	X	X
	SDMMC1	X	X	X	X
	SDMMC2	-	X	X	X
	OCTOSPIM	-	X	X	X
	SAES	X	X	X	X
	PKA	X	X	X	X
	RNG	X	X	X	X
	HASH	X	X	X	X
	AES	X	X	X	X
	USB	X	-	-	-
	OTG_FS	-	X	-	-
	OTG_HS	-	-	X	X
	DCMI	X	X	X	X
	ADC12	X	X	X	X

表4.TZSCによってセキュリティ保護可能なペリフェラル （続き）

バス	ペリフェラル	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB1	GFXMMUレジスタ	-	-	X	X
	GPU2Dレジスタ	-	-	X	X
	JPEG	-	-	-	X
	DCACHE2レジスタ	-	-	X	X
	DCACHE1レジスタ(1)	X	X	X	X
	ICACHEレジスタ (1)	X	X	X	X
	DMA2D	-	X	X	X
	TSC	X	X	X	X
	CRC	X	X	X	X
	RAMCFG	X	X	X	X
	MDF1	X	X	X	X
	FMAC	X	X	X	X
	CORDIC	X	X	X	X
APB3	VREFBUF	X	X	X	X
	COMP(2)	X	X	X	X
	OPAMP	X	X	X	X
	LPTIM4	X	X	X	X
	LPTIM3	X	X	X	X
	LPTIM1	X	X	X	X
	I2C3	X	X	X	X
	LPUART1	X	X	X	X
	SPI3	X	X	X	X
APB2	DSI	-	-	X	X
	LTDC	-	-	X	X
	GFXTIM	-	-	-	X
	SAI2	-	X	X	X
	SAI1	X	X	X	X
	TIM17	X	X	X	X
	TIM16	X	X	X	X
	TIM15	X	X	X	X
	USART1	X	X	X	X
	TIM8	X	X	X	X
	SPI1	X	X	X	X
	TIM1	X	X	X	X

表4.TZSCによってセキュリティ保護可能なペリフェラル （続き）

バス	ペリフェラル	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
APB1	I2C6	-	-	X	X
	I2C5	-	-	X	X
	USART6	-	-	X	X
	UCPD1	-	X	X	X
	FDCAN1	X	X	X	X
	LPTIM2	X	X	X	X
	I2C4	X	X	X	X
	CRS	X	X	X	X
	I2C2	X	X	X	X
	I2C1	X	X	X	X
	UART5	X	X	X	X
	UART4	X	X	X	X
	USART3	X	X	X	X
	USART2	-	X	X	X
	SPI2	X	X	X	X
	IWDG	X	X	X	X
	WWDG	X	X	X	X
	TIM7	X	X	X	X
	TIM6	X	X	X	X
	TIM5	X	X	X	X
	TIM4	X	X	X	X
	TIM3	X	X	X	X
	TIM2	X	X	X	X

1. ICACHEおよびDCACHE1は、それらのレジスタへのアクセスがTZSCによってセキュリティ保護されているかどうかに関わらず、TrustZoneを認識するペリフェラル。
2. STM32U535/545にあるCOMPは1つのみ。

表5.TrustZoneを認識するペリフェラル

バス	ペリフェラル	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB3	GTZC2	X	X	X	X
	EXTI	X	X	X	X
	LPDMA1	X	X	X	X
	RCC	X	X	X	X
	PWR	X	X	X	X
	LPGPIO1	X	X	X	X

表5.TrustZoneを認識するペリフェラル （続き）

バス	ペリフェラル	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB2	OTFDEC1(1)	X	X	X	X
	OTFDEC2 (1)	-	X	X	X
	GPIOJ	-	-	X	X
	GPIOI	-	X	X	X
	GPIOH	X	X	X	X
	GPIOG	X	X	X	X
	GPIOF	-	X	X	X
	GPIOE	X	X	X	X
	GPIOD	X	X	X	X
	GPIOC	X	X	X	X
	GPIOB	X	X	X	X
	GPIOA	X	X	X	X
AHB1	GTZC1	X	X	X	X
	FLASH	X	X	X	X
	GPDMA1	X	X	X	X
APB3	TAMP	X	X	X	X
	RTC	X	X	X	X
	SYSCFG	X	X	X	X

1. TZEN = 1のときは常にセキュア。

2.3 メモリ構成

2.3.1 概要

プログラムメモリ、データメモリ、レジスタ、およびI/Oポートは、同じ4GBのリニアなアドレス空間に配置されています。

バイトは、メモリ内でリトルエンディアン形式でコード化されます。ワード内で最も小さな番号のバイトがワードの最下位バイトとみなされ、最も大きな番号のバイトが最上位バイトとみなされます。

2.3.2 メモリマップとレジスタ境界アドレス

図3. STM32U535/545のIDAUマッピングに基づくメモリマップ

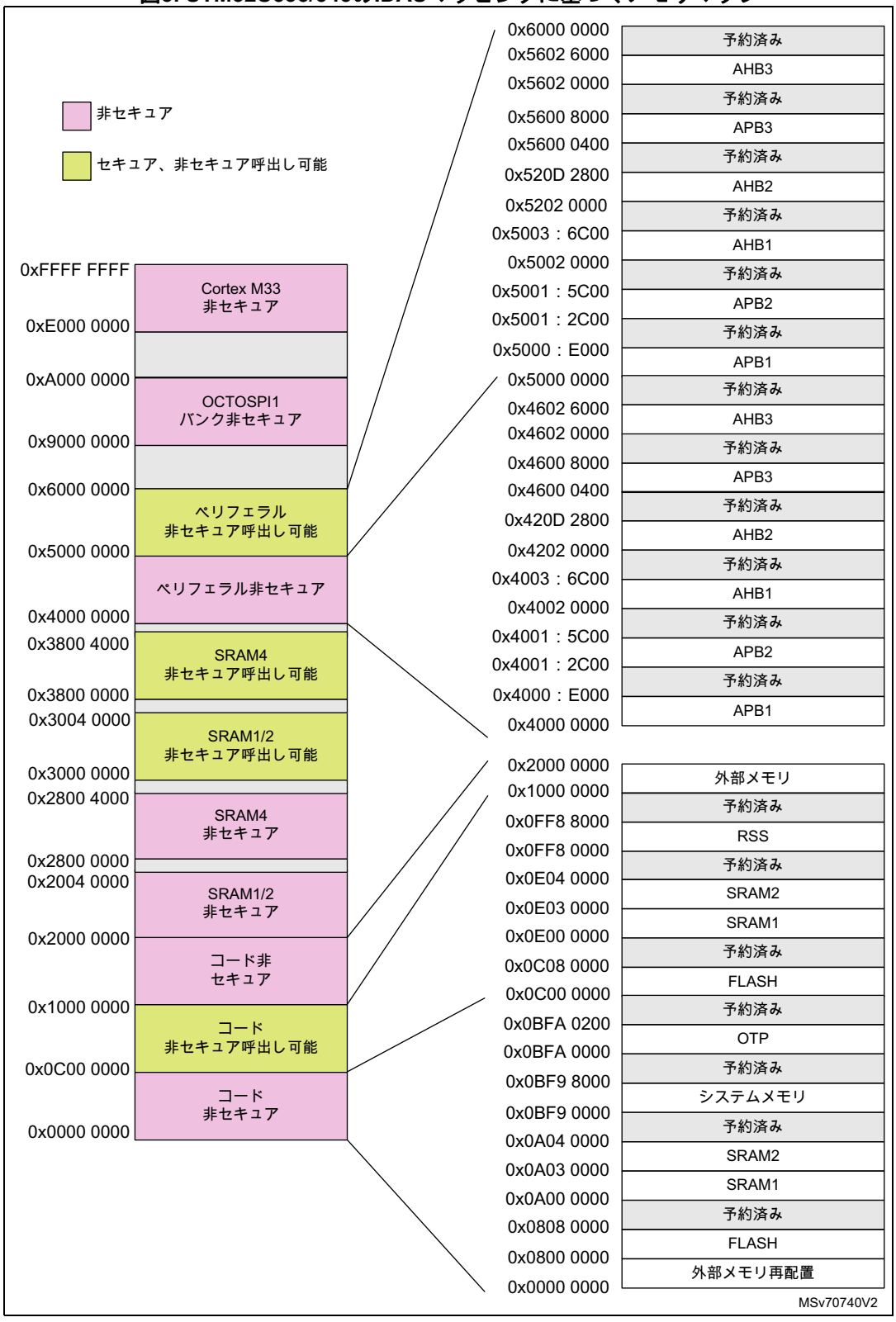


図4. STM32U575/585のIDAUマッピングに基づくメモリマップ

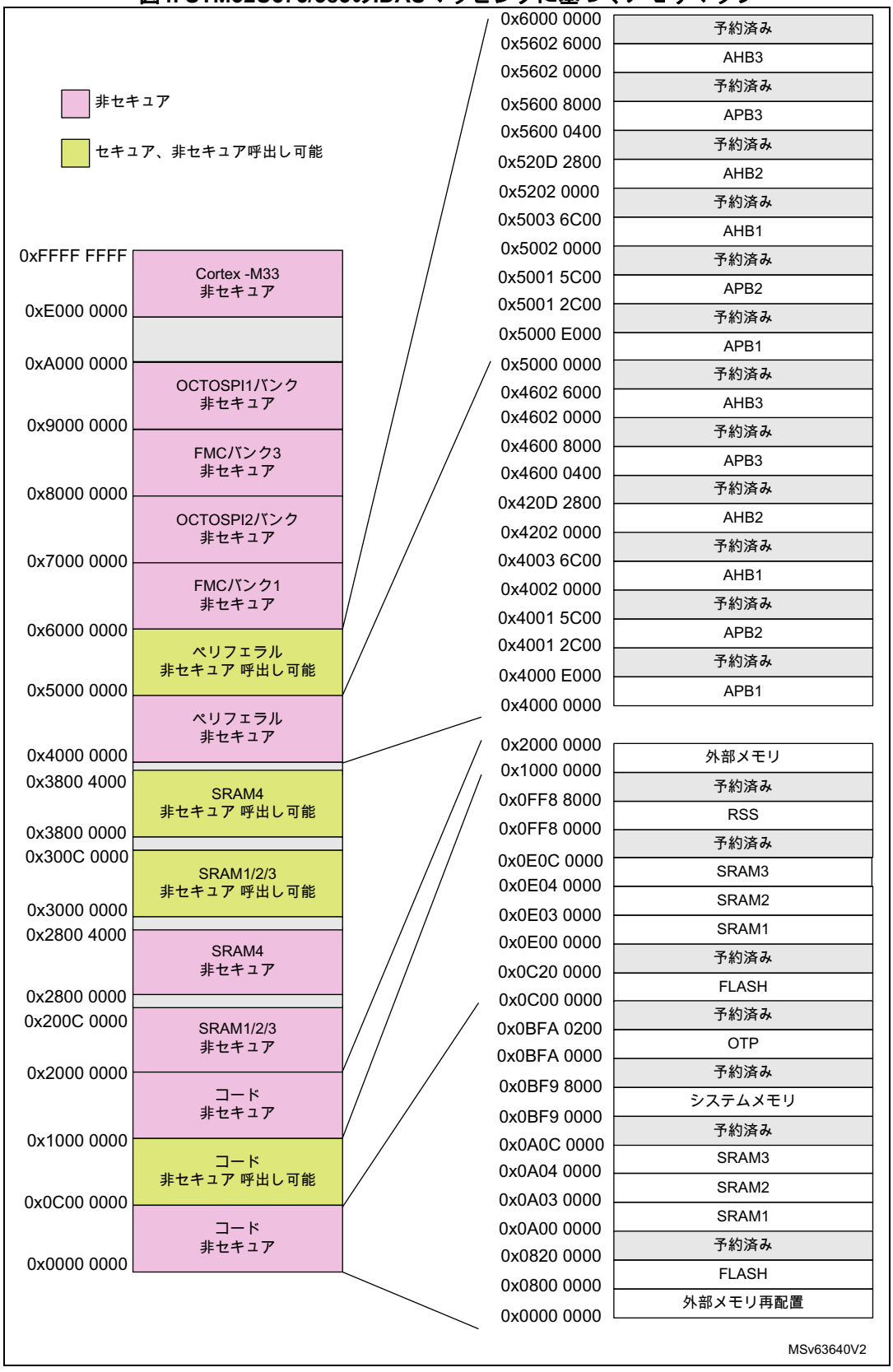


図5. STM32U59x/5AxのIDAUマッピングに基づくメモリマップ

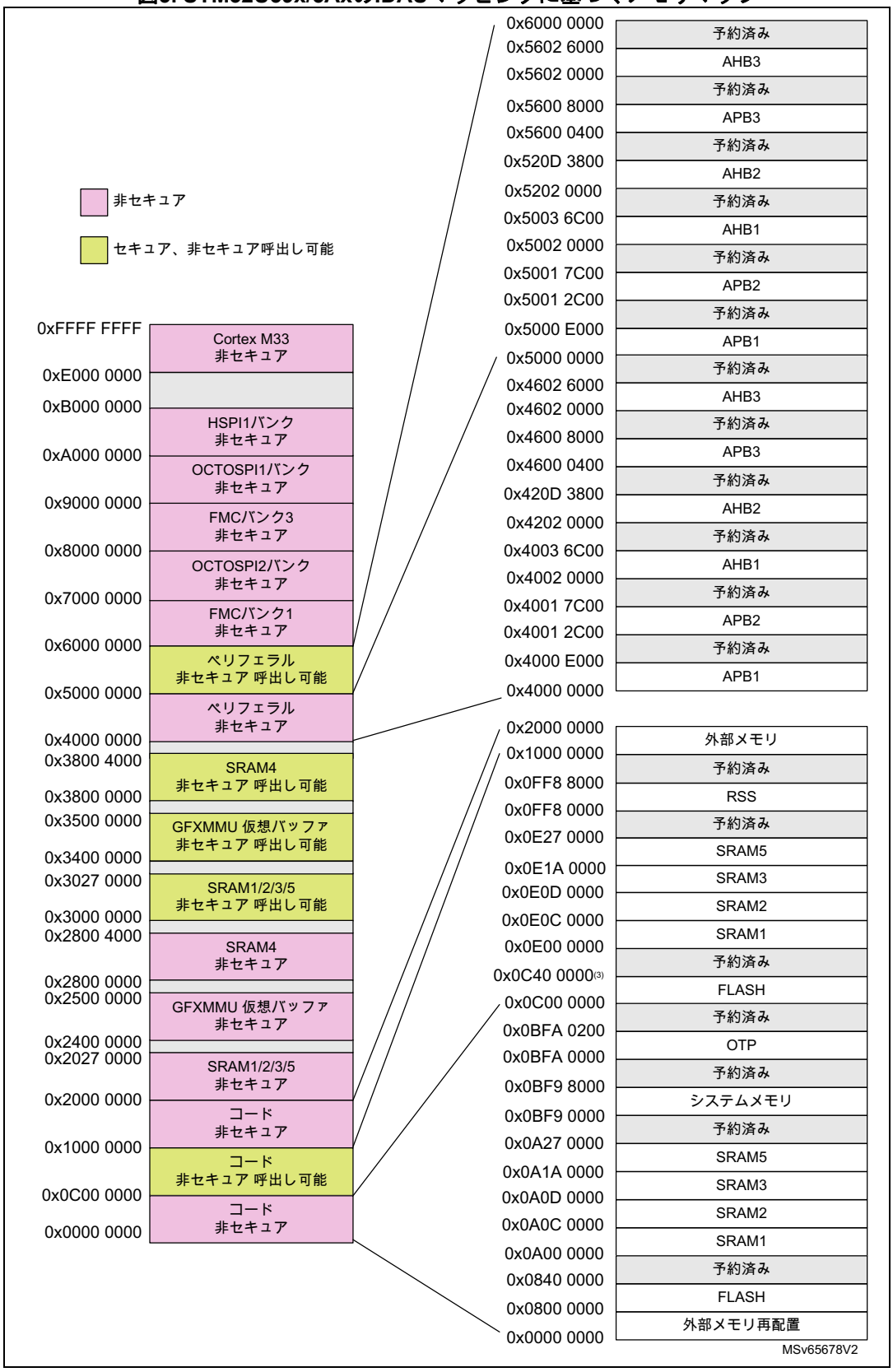
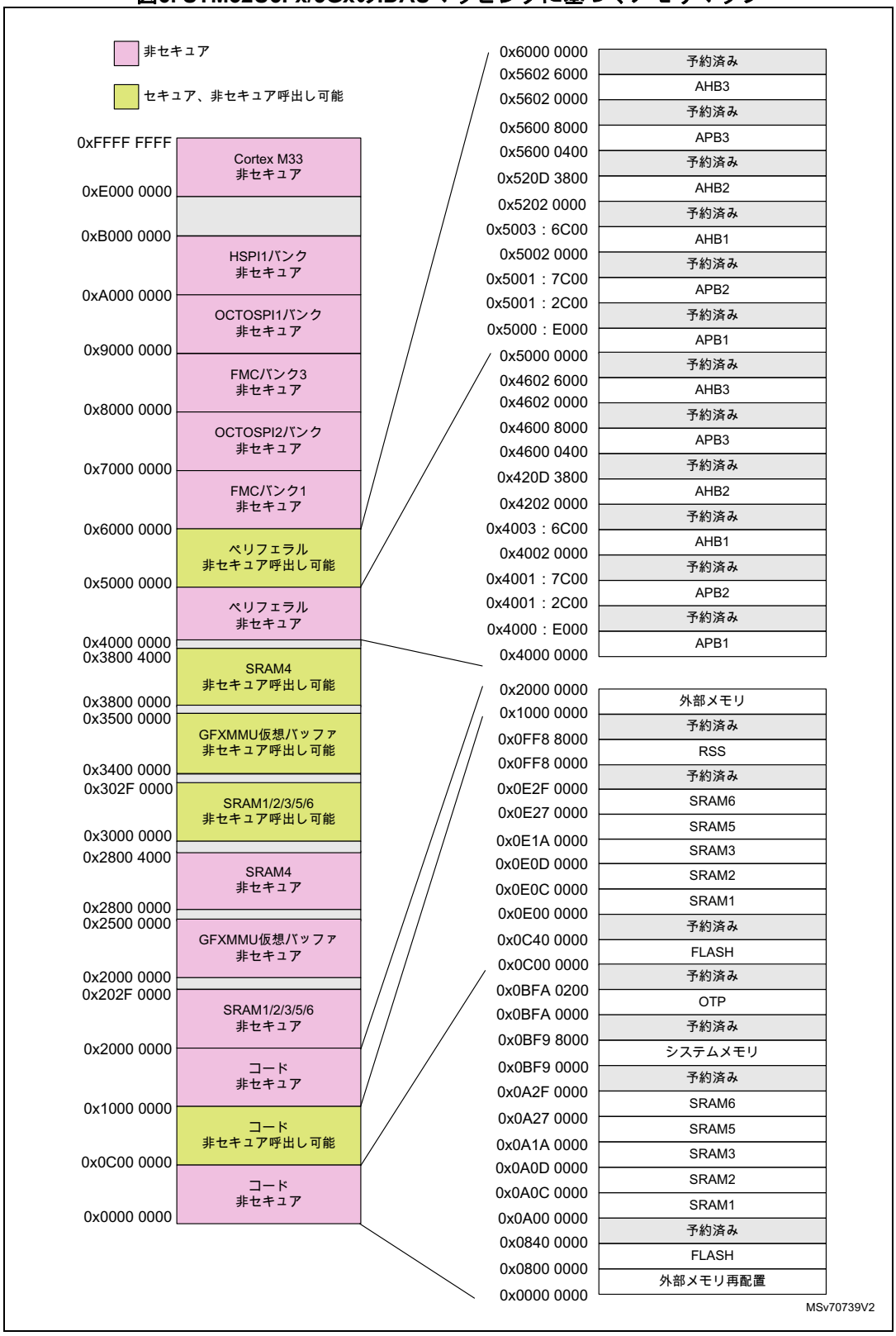


図6. STM32U5Fx/5GxのIDAUマッピングに基づくメモリマップ



オンチップメモリとペリフェラルに割り当てられていないメモリマップ領域はすべて、「予約済み」とみなされます。次の表に、デバイスで使用可能なペリフェラルの境界アドレスを示します。

表6.メモリマップとペリフェラルのレジスタ境界アドレス

バス	セキュアな境界アドレス	非セキュアな境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB3	0x5602 6000 - 0x5FFF FFFF	0x4602 6000 - 0x4FFF FFFF	164 M	予約済み	-	-	-	-	-
	0x5602 5000 - 0x5602 5FFF	0x4602 5000 - 0x4602 5FFF	4 K	LPDMA1	LPDMAレジスタマップ	X	X	X	X
	0x5602 4000 - 0x5602 4FFF	0x4602 4000 - 0x4602 4FFF	4 K	ADF1	ADFレジスタマップ	X	X	X	X
	0x5602 3C00 - 0x5602 3FFF	0x4602 3C00 - 0x4602 3FFF	1 K	予約済み	-	-	-	-	-
	0x5602 3800 - 0x5602 3BFF	0x4602 3800 - 0x4602 3BFF	1 K	GTZC2_MPCBB4	GTZC2 MPCBB4 レジスタマップ	X	X	X	X
	0x5602 3400 - 0x5602 37FF	0x4602 3400 - 0x4602 37FF	1 K	GTZC2_TZIC	GTZC2 TZICレジスタマップ	X	X	X	X
	0x5602 3000 - 0x5602 33FF	0x4602 3000 - 0x4602 33FF	1 K	GTZC2_TZSC	GTZC2 TZSCレジスタマップ	X	X	X	X
	0x5602 2400 - 0x5602 2FFF	0x4602 2400 - 0x4602 2FFF	3 K	予約済み	-	-	-	-	-
	0x5602 2000 - 0x5602 23FF	0x4602 2000 - 0x4602 23FF	1 K	EXTI	EXTIレジスタ·マップ	X	X	X	X
	0x5602 1C00 - 0x5602 1FFF	0x4602 1C00 - 0x4602 1FFF	1 K	予約済み	-	-	-	-	-
	0x5602 1800 - 0x5602 1BFF	0x4602 1800 - 0x4602 1BFF	1 K	DAC1	DACレジスタマップ	X	X	X	X
	0x5602 1400 - 0x5602 17FF	0x4602 1400 - 0x4602 17FF	1 K	予約済み	-	-	-	-	-
	0x5602 1000 - 0x5602 13FF	0x4602 1000 - 0x4602 13FF	1 K	ADC4	ADCレジスタ·マップ	X	X	X	X
	0x5602 0C00 - 0x5602 0FFF	0x4602 0C00 - 0x4602 0FFF	1 K	RCC	RCCレジスタ·マップ	X	X	X	X
	0x5602 0800 - 0x5602 0BFF	0x4602 0800 - 0x4602 0BFF	1 K	PWR	PWRレジスタマップ	X	X	X	X
	0x5602 0400 - 0x5602 07FF	0x4602 0400 - 0x4602 07FF	1 K	予約済み	-	-	-	-	-
	0x5602 0000 - 0x5602 03FF	0x4602 0000 - 0x4602 03FF	1 K	LPGPIO1	LPGPIOレジスタマップ	X	X	X	X
APB3	0x5600 8000 - 0x5601 FFFF	0x4600 8000 - 0x4601 FFFF	96 K	予約済み	-	-	-	-	-
	0x5600 7C00 - 0x5600 7FFF	0x4600 7C00 - 0x4600 7FFF	1 K	TAMP	TAMPレジスタマップ	X	X	X	X
	0x5600 7800 - 0x5600 7BFF	0x4600 7800 - 0x4600 7BFF	1 K	RTC	RTCレジスタマップ	X	X	X	X
	0x5600 7400 - 0x5600 77FF	0x4600 7400 - 0x4600 77FF	1 K	VREFBUF	VREFBUFレジスタマップ	X	X	X	X
	0x5600 5800 - 0x5600 73FF	0x4600 5800 - 0x4600 73FF	7 K	予約済み	-	-	-	-	-
	0x5600 5400 - 0x5600 57FF	0x4600 5400 - 0x4600 57FF	1 K	COMP(1)	COMPレジスタマップ	X	X	X	X
	0x5600 5000 - 0x5600 53FF	0x4600 5000 - 0x4600 53FF	1 K	OPAMP	OPAMPレジスタマップ	X	X	X	X
	0x5600 4C00 - 0x5600 4FFF	0x4600 4C00 - 0x4600 4FFF	1 K	LPTIM4	LPTIMレジスタマップ	X	X	X	X
	0x5600 4800 - 0x5600 4BFF	0x4600 4800 - 0x4600 4BFF	1 K	LPTIM3		X	X	X	X
	0x5600 4400 - 0x5600 47FF	0x4600 4400 - 0x4600 47FF	1 K	LPTIM1		X	X	X	X
	0x5600 2C00 - 0x5600 43FF	0x4600 2C00 - 0x4600 43FF	6 K	予約済み	-	-	-	-	-
	0x5600 2800 - 0x5600 2BFF	0x4600 2800 - 0x4600 2BFF	1 K	I2C3	I^2Cレジスタマップ	X	X	X	X
	0x5600 2400 - 0x5600 27FF	0x4600 2400 - 0x4600 27FF	1 K	LPUART1	LPUARTレジスタマップ	X	X	X	X
	0x5600 2000 - 0x5600 23FF	0x4600 2000 - 0x4600 23FF	1 K	SPI3	SPIレジスタマップ	X	X	X	X
	0x5600 0800 - 0x5600 1FFF	0x4600 0800 - 0x4600 1FFF	6 K	予約済み	-	-	-	-	-
	0x5600 0400 - 0x5600 07FF	0x4600 0400 - 0x4600 07FF	1 K	SYSCFG	SYSCFGレジスタマップ	X	X	X	X
AHB2	0x520D 3800 - 0x5600 03FF	0x420D 3800 - 0x4600 03FF	64.3 M	予約済み	-	-	-	-	-
	0x520D 3400 - 0x520D 37FF	0x420D 3400 - 0x420D 37FF	1 K	HSPI1	HSPI レジスタマップ	-	-	X	X
	0x520D 2800 - 0x520D 33FF	0x420D 2800 - 0x420D 33FF	3 K	予約済み	-	-	-	-	-
	0x520D 2400 - 0x520D 27FF	0x420D 2400 - 0x420D 27FF	1 K	OCTOSPI2レジスタ	OCTOSPIレジスタマップ	-	X	X	X
	0x520D 1800 - 0x520D 23FF	0x420D 1800 - 0x420D 23FF	3 K	予約済み	-	-	-	-	-
	0x520D 1400 - 0x520D 17FF	0x420D 1400 - 0x420D 17FF	1 K	OCTOSPI1レジスタ	OCTOSPIレジスタマップ	X	X	X	X
	0x520D 0800 - 0x520D 13FF	0x420D 0800 - 0x420D 13FF	3 K	予約済み	-	-	-	-	-

表6.メモリマップとペリフェラルのレジスタ境界アドレス （続き）

バス	セキュアな境界アドレス	非セキュアな境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB2（続き）	0x520D 0400 - 0x520D 07FF	0x420D 0400 - 0x420D 07FF	1 K	FSMCレジスタ	FMC レジスタマップ	-	X	X	X
	0x520C F800 - 0x520D 03FF	0x420C F800 - 0x420D 03FF	3K	予約済み	-	-	-	-	-
	0x520C F400 - 0x520C F7FF	0x420C F400 - 0x420C F7FF	1 K	DLYBOS2	DLYBレジスタマップ	-	X	X	X
	0x520C F000 - 0x520C F3FF	0x420C F000 - 0x420C F3FF	1 K	DLYBOS1		X	X	X	X
	0x520C 9000 - 0x520C EFFF	0x420C 9000 - 0x420C EFFF	24 K	予約済み	-	-	-	-	-
	0x520C 8C00 - 0x520C 8FFF	0x420C 8C00 - 0x420C 8FFF	1 K	SDMMC2	SDMMCレジスタマップ	-	X	X	X
	0x520C 8800 - 0x520C 8BFF	0x420C 8800 - 0x420C 8BFF	1 K	DLYBSD2	DLYBレジスタマップ	-	X	X	X
	0x520C 8400 - 0x520C 87FF	0x420C 8400 - 0x420C 87FF	1 K	DLYBSD1		X	X	X	X
	0x520C 8000 - 0x520C 83FF	0x420C 8000 - 0x420C 83FF	1 K	SDMMC1	SDMMCレジスタマップ	X	X	X	X
	0x520C 5800 - 0x520C 7FFF	0x420C 5800 - 0x420C 7FFF	10 K	予約済み	-	-	-	-	-
	0x520C 5400 - 0x520C 57FF	0x420C 5400 - 0x420C 57FF	1 K	OTFDEC2	OTFDECレジスタマップ	-	X	X	X
	0x520C 5000 - 0x520C 53FF	0x420C 5000 - 0x420C 53FF	1 K	OTFDEC1		X	X	X	X
	0x520C 4400 - 0x520C 4FFF	0x420C 4400 - 0x420C 4FFF	3 K	予約済み	-	-	-	-	-
	0x520C 4000 - 0x520C 43FF	0x420C 4000 - 0x420C 43FF	1 K	OCTOSPIM	OCTOSPIMレジスタマップ	-	X	X	X
	0x520C 2000 - 0x520C 3FFF	0x420C 2000 - 0x420C 3FFF	8 K	PKA	PKAレジスタマップ	X	X	X	X
	0x520C 1000 - 0x520C 1FFF	0x420C 1000 - 0x420C 1FFF	4 K	予約済み	-	-	-	-	-
	0x520C 0C00 - 0x520C 0FFF	0x420C 0C00 - 0x420C 0FFF	1 K	SAES	SAES レジスタマップ	X	X	X	X
	0x520C 0800 - 0x520C 0BFF	0x420C 0800 - 0x420C 0BFF	1 K	RNG	RNGレジスタマップ	X	X	X	X
	0x520C 0400 - 0x520C 07FF	0x420C 0400 - 0x420C 07FF	1 K	HASH	HASHレジスタマップ	X	X	X	X
	0x520C 0000 - 0x520C 03FF	0x420C 0000 - 0x420C 03FF	1 K	AES	AES レジスタマップ	X	X	X	X
	0x5204 0000 - 0x5205 FFFF	0x4204 0000 - 0x4205 FFFF	128 K	OTG_HS	OTG_HS レジスタマップ	-	-	X	X
	0x5204 0000 - 0x520B FFFF	0x4204 0000 - 0x420B FFFF	512 K	OTG_FS	OTG_FS レジスタマップ	-	X	-	-
	0x5202 C800 - 0x5203 FFFF	0x4202 C800 - 0x4203 FFFF	78 K	予約済み	-	-	-	-	-
	0x5202 C400 - 0x5202 C7FF	0x4202 C400 - 0x4202 C7FF	1 K	PSSI	PSSIレジスタマップ	X	X	X	X
	0x5202 C000 - 0x5202 C3FF	0x4202 C000 - 0x4202 C3FF	1 K	DCMI	DCMIレジスタマップ	X	X	X	X
	0x5202 8400 - 0x5202 BFFF	0x4202 8400 - 0x4202 BFFF	15 K	予約済み	-	-	-	-	-
	0x5202 8000 - 0x5202 83FF	0x4202 8000 - 0x4202 83FF	1 K	ADC12(2)	ADCレジスタマップ	X	X	X	X
	0x5202 2800 - 0x5202 7FFF	0x4202 2800 - 0x4202 7FFF	22 K	予約済み	-	-	-	-	-
	0x5202 2400 - 0x5202 27FF	0x4202 2400 - 0x4202 27FF	1 K	GPIOJ	GPIOレジスタマップ	-	-	X	X
	0x5202 2000 - 0x5202 23FF	0x4202 2000 - 0x4202 23FF	1 K	GPIOI		-	X	X	X
	0x5202 1C00 - 0x5202 1FFF	0x4202 1C00 - 0x4202 1FFF	1 K	GPIOH		X	X	X	X
	0x5202 1800 - 0x5202 1BFF	0x4202 1800 - 0x4202 1BFF	1 K	GPIOG		X	X	X	X
	0x5202 1400 - 0x5202 17FF	0x4202 1400 - 0x4202 17FF	1 K	GPIOF		-	X	X	X
	0x5202 1000 - 0x5202 13FF	0x4202 1000 - 0x4202 13FF	1 K	GPIOE		X	X	X	X
	0x5202 0C00 - 0x5202 0FFF	0x4202 0C00 - 0x4202 0FFF	1 K	GPIOD		X	X	X	X
	0x5202 0800 - 0x5202 0BFF	0x4202 0800 - 0x4202 0BFF	1 K	GPIOC		X	X	X	X
	0x5202 0400 - 0x5202 07FF	0x4202 0400 - 0x4202 07FF	1 K	GPIOB		X	X	X	X
	0x5202 0000 - 0x5202 03FF	0x4202 0000 - 0x4202 03FF	1 K	GPIOA		X	X	X	X
AHB1	0x5003 6C00 - 0x5201 FFFF	0x4003 6C00 - 0x4201 FFFF	32.7 M	予約済み	-	-	-	-	-
	0x5003 6400 - 0x5003 6BFF	0x4003 6400 - 0x4003 6BFF	2 K	BKPSRAM	-	X	X	X	X
	0x5003 4000 - 0x5003 63FF	0x4003 4000 - 0x4003 63FF	9 K	予約済み	-	-	-	-	-

表6.メモリマップとペリフェラルのレジスタ境界アドレス （続き）

バス	セキュアな境界アドレス	非セキュアな境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
AHB1（続き）	0x5003 3C00 - 0x5003 3FFF	0x4003 3800 - 0x4003 3BFF	1 K	GTZC1_MPCBB6	GTZC1 MPCBBz レジスタマップ（z = 1、2、3、5、6）	-	-	-	X
	0x5003 3800 - 0x5003 3BFF	0x4003 3C00 - 0x4003 3FFF	1 K	GTZC1_MPCBB5		-	-	X	X
	0x5003 3400 - 0x5003 37FF	0x4003 3400 - 0x4003 37FF	1 K	GTZC1_MPCBB3		-	X	X	X
	0x5003 3000 - 0x5003 33FF	0x4003 3000 - 0x4003 33FF	1 K	GTZC1_MPCBB2		X	X	X	X
	0x5003 2C00 - 0x5003 2FFF	0x4003 2C00 - 0x4003 2FFF	1 K	GTZC1_MPCBB1		X	X	X	X
	0x5003 2800 - 0x5003 2BFF	0x4003 2800 - 0x4003 2BFF	1 K	GTZC1_TZIC	GTZC1 TZICレジスタ·マップ	X	X	X	X
	0x5003 2400 - 0x5003 27FF	0x4003 2400 - 0x4003 27FF	1 K	GTZC1_TZSC	GTZC1 TZSC レジスタのマップ	X	X	X	X
	0x5003 1C00 - 0x5003 23FF	0x4003 1C00 - 0x4003 23FF	2 K	予約済み	-	-	-	-	-
	0x5003 1800 - 0x5003 1BFF	0x4003 1800 - 0x4003 1BFF	1 K	DCACHE2	DCACHEレジスタマップ	-	-	X	X
	0x5003 1400 - 0x5003 17FF	0x4003 1400 - 0x4003 17FF	1 K	DCACHE1		X	X	X	X
	0x5003 0800 - 0x5003 13FF	0x4003 0800 - 0x4003 13FF	3 K	予約済み	-	-	-	-	-
	0x5003 0400 - 0x5003 07FF	0x4003 0400 - 0x4003 07FF	1 K	ICACHE	ICACHEレジスタマップ	X	X	X	X
	0x5003 0000 - 0x5003 03FF	0x4003 0000 - 0x4003 03FF	18 K	予約済み	-	-	-	-	-
	0x5002 F000 - 0x5002 FFFF	0x4002 F000 - 0x4002 FFFF	1 K	GPU2D	-	-	-	X	X
	0x5002 C000 - 0x5002 EFFF	0x4002 C000 - 0x4002 EFFF	1 K	GFXMMU	GFXMMUレジスタマップ	-	-	X	X
	0x5002 BC00 - 0x5002 BFFF	0x4002 BC00 - 0x4002 BFFF	18 K	予約済み	-	-	-	-	-
	0x5002 B000 - 0x5002 BBFF	0x4002 B000 - 0x4002 BBFF	3 K	DMA2D	DMA2Dレジスタマップ	-	X	X	X
	0x5002 A000 - 0x5002 AFFF	0x4002 A000 - 0x4002 AFFF	4 K	JPEG	JPEGコーデックレジスタマップ	-	-	-	X
	0x5002 7000 - 0x5002 AFFF	0x4002 7000 - 0x4002 AFFF	16 K	予約済み	-	-	-	-	-
	0x5002 6000 - 0x5002 6FFF	0x4002 6000 - 0x4002 6FFF	4 K	RAMCFG	RAMCFGレジスタマップ	X	X	X	X
	0x5002 5000 - 0x5002 5FFF	0x4002 5000 - 0x4002 5FFF	4 K	MDF1(3)	MDFレジスタ·マップ	X	X	X	X
	0x5002 4400 - 0x5002 4FFF	0x4002 4400 - 0x4002 4FFF	3 K	予約済み	-	-	-	-	-
	0x5002 4000 - 0x5002 43FF	0x4002 4000 - 0x4002 43FF	1 K	TSC	TSCレジスタマップ	X	X	X	X
	0x5002 3400 - 0x5002 3FFF	0x4002 3400 - 0x4002 3FFF	3 K	予約済み	-	-	-	-	-
	0x5002 3000 - 0x5002 33FF	0x4002 3000 - 0x4002 33FF	1 K	CRC	CRCレジスタマップ	X	X	X	X
	0x5002 2400 - 0x5002 2FFF	0x4002 2400 - 0x4002 2FFF	3 K	予約済み	-	-	-	-	-
	0x5002 2000 - 0x5002 23FF	0x4002 2000 - 0x4002 23FF	1 K	Flashレジスタ	Flashレジスタマップ	X	X	X	X
	0x5002 1800 - 0x5002 1FFF	0x4002 1800 - 0x4002 1FFF	2 K	予約済み	-	-	-	-	-
	0X5002 1400 - 0x5002 17FF	0X4002 1400 - 0x4002 17FF	1 K	FMAC	FMACレジスタマップ	X	X	X	X
	0X5002 1000 - 0x5002 13FF	0X4002 1000 - 0x4002 13FF	1 K	CORDIC	CORDICレジスタマップ	X	X	X	X
	0x5002 0000 - 0x5002 0FFF	0x4002 0000 - 0x4002 0FFF	4 K	GPDMA1	GPDMAレジスタマップ	X	X	X	X

表6.メモリマップとペリフェラルのレジスタ境界アドレス （続き）

バス	セキュアな境界アドレス	非セキュアな境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
APB2	0x5001 7C00 - 0x5001 FFFF	0x4001 7C00 - 0x4001 FFFF	33 K	予約済み	-	-	-	-	-
	0x5001 6C00 - 0x5001 7BFF	0x4001 6C00 - 0x4001 7BFF	4 K	DSI	DSIレジスタマップ	-	-	X	X
	0x5001 6800 - 0x5001 6BFF	0x4001 6800 - 0x4001 6BFF	1 K	LTDC	LTDCレジスタマップ	-	-	X	X
	0x5001 6400 - 0x5001 67FF	0x4001 6400 - 0x4001 67FF	1 K	GFXTIM	GFXTIMレジスタマップ	-	-	-	X
	0x5001 6400 - 0x5001 6BFF	0x4001 6400 - 0x4001 6BFF	2 K	USB RAM	-	X	-	-	-
	0x5001 6000 - 0x5001 63FF	0x4001 6000 - 0x4001 63FF	1 K	USB	USBレジスタマップ	X	-	-	-
	0x5001 5C00 - 0x5001 5FFF	0x4001 5C00 - 0x4001 5FFF	1 K	予約済み	-	-	-	-	-
	0x5001 5800 - 0x5001 5BFF	0x4001 5800 - 0x4001 5BFF	1 K	SAI2	SAIレジスタマップ	-	X	X	X
	0x5001 5400 - 0x5001 57FF	0x4001 5400 - 0x4001 57FF	1 K	SAI1		X	X	X	X
	0x5001 4C00 - 0x5001 53FF	0x4001 4C00 - 0x4001 53FF	2 K	予約済み	-	-	-	-	-
	0x5001 4800 - 0x5001 4BFF	0x4001 4800 - 0x4001 4BFF	1 K	TIM17	TIM16/TIM17 レジスタマップ	X	X	X	X
	0x5001 4400 - 0x5001 47FF	0x4001 4400 - 0x4001 47FF	1 K	TIM16		X	X	X	X
	0x5001 4000 - 0x5001 43FF	0x4001 4000 - 0x4001 43FF	1 K	TIM15	TIM15レジスタマップ	X	X	X	X
	0x5001 3C00 - 0x5001 3FFF	0x4001 3C00 - 0x4001 3FFF	1 K	予約済み	-	-	-	-	-
APB2	0x5001 3800 - 0x5001 3BFF	0x4001 3800 - 0x4001 3BFF	1 K	USART1	USARTレジスタマップ	X	X	X	X
	0x5001 3400 - 0x5001 37FF	0x4001 3400 - 0x4001 37FF	1 K	TIM8	TIMx レジスタマップ	X	X	X	X
	0x5001 3000 - 0x5001 33FF	0x4001 3000 - 0x4001 33FF	1 K	SPI1	SPIレジスタマップ	X	X	X	X
	0x5001 2C00 - 0x5001 2FFF	0x4001 2C00 - 0x4001 2FFF	1 K	TIM1	TIMx レジスタマップ	X	X	X	X

表6.メモリマップとペリフェラルのレジスタ境界アドレス （続き）

バス	セキュアな境界アドレス	非セキュアな境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
APB1	0x5000 E000 - 0x5001 2BFF	0x4000 E000 - 0x4001 2BFF	19 K	予約済み	-	-	-	-	-
	0x5000 DC00 - 0x5000 DFFF	0x4000 DC00 - 0x4000 DFFF	1 K	UCPD1	UCPDレジスタ･マップ	-	X	X	X
	0x5000 B000 - 0x5000 DBFF	0x4000 B000 - 0x4000 DBFF	11 K	予約済み	-	-	-	-	-
	0x5000 AC00 - 0x5000 AFFF	0x4000 AC00 - 0x4000 AFFF	1 K	FDCAN1 RAM	-	X	X	X	X
	0x5000 A800 - 0x5000 ABFF	0x4000 A800 - 0x4000 ABFF	1 K	予約済み	-	-	-	-	-
	0x5000 A400 - 0x5000 A7FF	0x4000 A400 - 0x4000 A7FF	1 K	FDCAN1	FDCANレジスタマップ	X	X	X	X
	0x5000 A000 - 0x5000 A3FF	0x4000 A000 - 0x4000 A3FF	1 K	予約済み	-	-	-	-	-
	0x5000 9C00 - 0x5000 9FFF	0x4000 9C00 - 0x4000 9FFF	1 K	I2C6	I^2Cレジスタマップ	-	-	X	X
	0x5000 9800 - 0x5000 9BFF	0x4000 9800 - 0x4000 9BFF	1 K	I2C5		-	-	X	X
	0x5000 9400 - 0x5000 97FF	0x4000 9400 - 0x4000 97FF	1 K	LPTIM2	LPTIMレジスタマップ	X	X	X	X
	0x5000 8800 - 0x5000 93FF	0x4000 8800 - 0x4000 93FF	3 K	予約済み	-	-	-	-	-
	0x5000 8400 - 0x5000 87FF	0x4000 8400 - 0x4000 87FF	1 K	I2C4	I^2Cレジスタマップ	X	X	X	X
	0x5000 6800 - 0x5000 83FF	0x4000 6800 - 0x4000 83FF	8 K	予約済み	-	-	-	-	-
	0x5000 6400 - 0x5000 67FF	0x4000 6400 - 0x4000 67FF	1 K	USART6	USARTレジスタマップ	-	-	X	X
	0x5000 6000 - 0x5000 63FF	0x4000 6000 - 0x4000 63FF	1 K	CRS	CRSレジスタマップ	X	X	X	X
	0x5000 5C00 - 0x5000 5FFF	0x4000 5C00 - 0x4000 5FFF	1 K	予約済み	-	-	-	-	-
	0x5000 5800 - 0x5000 5BFF	0x4000 5800 - 0x4000 5BFF	1 K	I2C2	I^2Cレジスタマップ	X	X	X	X
	0x5000 5400 - 0x5000 57FF	0x4000 5400 - 0x4000 57FF	1 K	I2C1		X	X	X	X
	0x5000 5000 - 0x5000 53FF	0x4000 5000 - 0x4000 53FF	1 K	UART5	USARTレジスタマップ	X	X	X	X
	0x5000 4C00 - 0x5000 4FFF	0x4000 4C00 - 0x4000 4FFF	1 K	UART4		X	X	X	X
	0x5000 4800 - 0x5000 4BFF	0x4000 4800 - 0x4000 4BFF	1 K	USART3		X	X	X	X
	0x5000 4400 - 0x5000 47FF	0x4000 4400 - 0x4000 47FF	1 K	USART2		-	X	X	X
	0x5000 3C00 - 0x5000 43FF	0x4000 3C00 - 0x4000 43FF	2 K	予約済み	-	-	-	-	-
	0x5000 3800 - 0x5000 3BFF	0x4000 3800 - 0x4000 3BFF	1 K	SPI2	SPIレジスタマップ	X	X	X	X
	0x5000 3400 - 0x5000 37FF	0x4000 3400 - 0x4000 37FF	1 K	予約済み	-	-	-	-	-
	0x5000 3000 - 0x5000 33FF	0x4000 3000 - 0x4000 33FF	1 K	IWDG	IWDGレジスタマップ	X	X	X	X
	0x5000 2C00 - 0x5000 2FFF	0x4000 2C00 - 0x4000 2FFF	1 K	WWDG	WWDGレジスタマップ	X	X	X	X
	0x5000 1800 - 0x5000 2BFF	0x4000 1800 - 0x4000 2BFF	5 K	予約済み	-	-	-	-	-
	0x5000 1400 - 0x5000 17FF	0x4000 1400 - 0x4000 17FF	1 K	TIM7	TIMxレジスタマップ	X	X	X	X
	0x5000 1000 - 0x5000 13FF	0x4000 1000 - 0x4000 13FF	1 K	TIM6		X	X	X	X
	0x5000 0C00 - 0x5000 0FFF	0x4000 0C00 - 0x4000 0FFF	1 K	TIM5	TIMxレジスタマップ	X	X	X	X
	0x5000 0800 - 0x5000 0BFF	0x4000 0800 - 0x4000 0BFF	1 K	TIM4		X	X	X	X
	0x5000 0400 - 0x5000 07FF	0x4000 0400 - 0x4000 07FF	1 K	TIM3		X	X	X	X
	0x5000 0000 - 0x5000 03FF	0x4000 0000 - 0x4000 03FF	1 K	TIM2		X	X	X	X

1. STM32U535/545にあるCOMPは1つのみ。
2. STM32U535/545/575/585にデュアルADCモードなし。
3. MDFの機能は、STM32U535/545にある2つのフィルタのみ。

2.3.3 内蔵SRAM

表7.SRAMサイズ

SRAM	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
SRAM1	192KB		768KB	
SRAM2	64KB			
SRAM3	-	512KB	832KB	
SRAM4	16KB			
SRAM5	-	-	832KB	
SRAM6	-	-	-	512KB
BKSRAM	2KB			
合計	274KB	786KB	2514KB	3026KB

これらのSRAMには、バイト、ハーフワード（16bit）、またはフルワード（32bit）によるアクセスが可能です。これらのメモリは、CPUとDMAの両方によるアドレス指定が可能です。

CPUは、選択したアドレスに応じて、システムバスまたはCバス経由でSRAM1、SRAM2、SRAM3、SRAM5、およびSRAM6にアクセスできます。CPUは、システムバス経由でのみSRAM4およびBKPSRAMにアクセスできます。

TrustZoneセキュリティが有効な場合、すべてのSRAMはリセット後はセキュアです。SRAMは、ブロック単位で非セキュアとしてプログラムできます。詳細については、セクション 5：グローバルTrustZoneコントローラ（GTZC）を参照してください。

SRAM機能の詳細については、セクション 6.3.1：内部SRAM機能を参照してください。

2.3.4 Flashメモリの概要

Flashメモリは2つの別個の物理領域で構成されます。

- アプリケーションプログラムとユーザデータを含むメインFlashメモリブロック
- 以下の部分で構成される情報ブロック：
  - ハードウェアとメモリ保護のユーザ設定のオプションバイト。
  - ST独自仕様コードを含むシステムメモリ
  - OTP（一度だけプログラム可能な）領域。

Flashインタフェースは、AHBプロトコルに基づいて命令アクセスとデータアクセスを実装します。また、Flashレジスタとセキュリティ·アクセス制御機能によって制御されるFlashメモリ操作（プログラム/消去）の実行に必要なロジックも実装します。詳細については、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。

3 システムセキュリティ

STM32U5シリーズデバイスは、セキュリティの機能を統合させたもので設計され、その一部は標準のArm TrustZoneテクノロジーに基づいています。

これらのセキュリティ機能は、IoTデバイスをセキュリティ標準に照らして評価するプロセスを簡素化します。また、再利用を容易にし、相互運用性を高め、APIの断片化を最小限に抑えるため、OEMおよびサードパーティ開発者にとって、ソフトウェア開発のコストと複雑性が大幅に軽減します。

このセクションでは、STM32U5シリーズデバイスで使用可能なさまざまなセキュリティ機能について説明します。

3.1 主なセキュリティ機能

- セキュリティ保護可能なI/O、メモリ、およびペリフェラルまで拡張された、Cortex-M33の特権モードとArmv8-Mメインラインセキュリティ拡張を用いたリソースの分離
- デバイス固有の暗号鍵ペアによるセキュアなファームウェアインストール（SFI）
  - USART、USB、I²C、SPI、FDCAN、およびJTAG経由のイメージのダウンロードをサポートする、オンチップで不変のブートローダの利用
- 一意なブートエントリ機能と非表示保護領域（HDP）機構によるセキュア·ブート
- 以下の特徴を持つセキュアストレージ：
  - セキュアなHDP領域で保護された、オンチップの不揮発性セキュアストレージ
  - タンパ発生時に自動的に消去される、バッテリ駆動の揮発性セキュアストレージ
  - AESエンジンの書込み専用キーレジスタ
  - デバイスの96bitのユニークIDとJTAGの32bitのデバイス固有ID
  - ハードウェアの秘密の不揮発性の導出されたハードウェア·ユニーク·キー（DHUK）およびアプリケーション定義の揮発性のブート·ハードウェア·キー（BHK）を用いて、オンチップによって強化されたストレージ技術、キーは両方ともハードウェアによって耐DPA SAESエンジンにロード可能
- 汎用暗号化アクセラレーション：
  - ECB、CBC、CTR、GCM、およびCCM連鎖モードをサポートするAES 256bitエンジン
  - サイドチャネル対策と軽減策を備えた、ECB および CBC 連鎖モードをサポートする、セキュアAES 256bitセキュリティコプロセッサ
  - MD5/SHA-1チェックサムとSHA-2セキュアハッシュをサポートするHASHプロセッサ
  - 機密情報を処理するときのサイドチャネル対策と軽減策を実装する、RSA/DH（最大4096bit）およびECC（最大640bit）向け公開鍵アクセラレータ（PKA）
  - 真の乱数発生器（RNG）、NIST SP800-90B認証済み
- OCTOSPI経由で接続された外部Flashメモリに保存された暗号化済イメージのオンザフライ復号：
  - 標準NOR Flashメモリで遅延がほぼゼロ
  - デバイス固有の秘密鍵を使用してイメージを暗号化するために使用可能
  - タンパ発生時の自動的な鍵消去
- 製品のデコミッショニング（自動消去）対応の読出し保護（RDP）された柔軟性の高いライフ·サイクル·スキーム
  - RDPレベルに応じたデバッグ保護
  - オプションのパスワードに基づいた、RDPレベル2用を含むRDPレベル回帰

- TrustZone、オンザフライ復号、およびRDPレベル0.5を使用した保護されたファームウェアの配布スキーム
- 温度、電圧、および周波数攻撃に対するアクティブタンパおよび保護
  - すべての電力モードで使用可能な8つのアクティブな入力と8つのアクティブな出力タンパピン

3.2 セキュアインストール

セキュア・ファームウェア・インストール（SFI）はSTMicroelectronicsのセキュアサービスで、デバイスに保存された不変のRSSコードによって認証され、復号化されています。SFIにより、信頼できない製造環境（OEM契約メーカーなど）へのOEMファームウェアのセキュアで数量管理されたインストールが可能です。

内部Flashメモリに、または外部Flashメモリに暗号化されて書き込まれるインストールイメージの秘匿性もAESを使用して保護されます。

SFIネイティブサービスは、以下のハードウェアセキュリティ機能を利用します。

- セキュア・ブート（セクション 3.3を参照）
- TrustZoneを使用したリソース分離（セクション 3.5を参照）
- 非表示保護を使用した時間的分離（セクション 3.6.1を参照）
- セキュア実行（セクション 3.7を参照）
- 関連する暗号化エンジンのあるセキュア・ストレージ（セクション 3.8およびセクション 3.9を参照）

詳細については、アプリケーション・ノート「STM32 MCUs secure firmware install (SFI) overview」（AN4992）を参照してください。

3.3 セキュア・ブート

セキュア・ブートは、システムリセット後に常に実行される不変コードです。信頼の起点として、このコードは、デバイスの静的保護を確認し、使用可能なデバイスのランタイム保護を有効にして、プラットフォームで無効または悪意のあるコードが実行されるリスクを軽減します。信頼の起点として、セキュア・ブートは、実行前に、次のレベルのファームウェアの完全性と認証性もチェックします。

セキュア・ブートの実際の機能は、TrustZone機能の可用性と、デバイスの保存されているファームウェアによって異なります。セキュア・ブートは、一般にセキュアストレージを初期化し、オンザフライ復号化キーをOTFDECにインストールします。これは、外部Flashメモリに保存された暗号化されたファームウェアを使用するためです。

デバイスのTrustedFirmware-M（TFM）アプリケーションは、STM32開発エコシステムでサポートされており、セキュア・ブート機能を含む信頼の起点のソリューションを提供します。詳細については、ユーザマニュアル（UM2851）を参照してください。

デバイスでは、セキュア・ブートは、次のようなハードウェアセキュリティ機能を利用します。

- TrustZoneを使用したリソース分離（セクション 3.5を参照）
- 非表示保護を使用した時間的分離（セクション 3.6.1を参照）
- セキュア実行（セクション 3.7を参照）
- セキュアインストールおよび更新（セクション 3.2およびセクション 3.4を参照）
- 使用可能な場合、関連する暗号化エンジンのあるセキュア・ストレージ（セクション 3.8およびセクション 3.9を参照）

このセクションでは、特にセキュア・ブートのために設計された機能について説明します。

3.3.1 一意なブートエントリおよびBOOT_LOCK

TrustZoneが有効で（TZEN = 1）、BOOT_LOCKセキュア·オプション·ビットがクリアされているとき、アプリケーションは、システムFlashメモリ（次のセクションを参照）またはセキュアユーザFlashメモリのいずれかの、SECBOOTADD0オプションバイトによって定義されたアドレスにあるブートエントリポイントを選択します。

TrustZoneが有効で（TZEN = 1）、BOOT_LOCKセキュア·オプション·ビットがセットされているとき、デバイスの一意なブートエントリは、SECBOOTADD0オプションバイトによって定義された変更不能なセキュアアドレスです。これらのオプションバイトはすべて、BOOT_LOCKがセットされている場合は、アプリケーションで変更できません。

注： **BOOT_LOCK オプションビットは、クリアされている限り、何の制約もなくセットできます。ただし、いったんセットすると、BOOT_LOCKオプションビットはRDPレベルが0より大きいときに、クリアできなくなります。**

ブートメカニズムの詳細については、セクション 4：ブートモードを参照してください。

3.3.2 システムFlashメモリの不変の信頼の起点

システムFlashメモリに保存される不変の信頼の基点コードは、まず、SFIを開始するために使用されます。これにより、信頼できない製造環境（OEM契約メーカーなど）へのOEMファームウェアのセキュアで数量管理されたインストールが可能になります。

STMicroelectronicsの不変コードには、セキュア·アプリケーションがシステムリセットをトリガする前にSYSCFG_RSSCMDRレジスタを非null値にセットしたときに、実行時に呼出し可能なセキュア·ランタイム·サービスも含まれています。このランタイム機能は、BOOT_LOCKセキュア·オプション·ビットがセットされると無効化され、SECBOOTADD0によって定義されたセキュアアドレスがセキュアユーザFlashメモリにセットされます。

3.4 セキュア更新

セキュアファームウェア更新は、セキュア·ブート後に実行するセキュアサービスです。実際の機能は、TrustZone機能の可用性と、デバイスに保存されているファームウェアによって異なります。

デバイスのTrusted Firmware-M（TFM）アプリケーションは、STM32開発エコシステムでサポートされており、マイクロコントローラ内蔵プログラムを更新して、新しい機能を追加したり潜在的な問題を修正したりできます。更新プロセスは安全な方法で実行され、不正な更新やデバイス上の機密データへの不正アクセスを防ぎます。

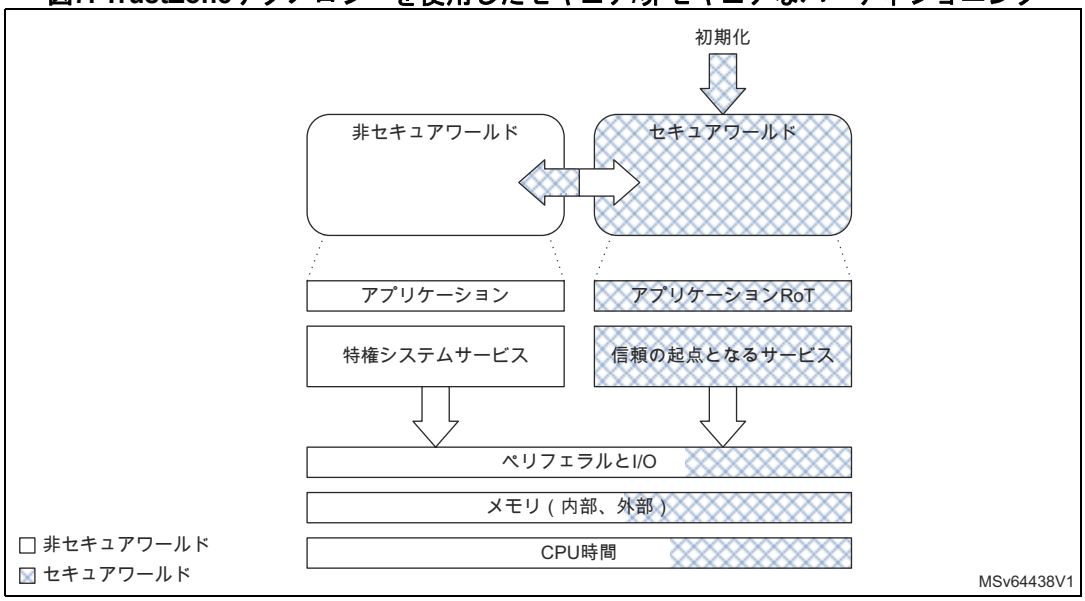
ファームウェアの更新は、セキュア部分と非セキュア部分の両方を含む単一のファームウェアイメージ、またはファームウェアイメージのセキュア（非セキュア）部分で個別に行うことができます。

デバイスでは、セキュア更新アプリケーションは、セクション 3.2で説明されているファームウェアインストールと同じハードウェアセキュリティを利用します。詳細については、ユーザマニュアル『Getting started with STM32CubeU5 TFM application』（UM2851）を参照してください。

3.5 TrustZoneを使用したリソースの分離

STM32U5シリーズのデバイスでは、ハードウェアとソフトウェアのリソースをセキュアワールドまたは非セキュアワールドのいずれかに配置できるよう、下図に示すように分けることができます。

図7. TrustZoneテクノロジーを使用したセキュア/非セキュアなパーティショニング



注： **プラットフォームの初期パーティショニングは、デバイスのリセット後に実行されるセキュア・ファームウェアの管理下で行われます。**

このリソース分離テクノロジーは、セキュアワールドを使用して、非セキュアワールドで実行されている、より公開度の高いコードからの意図的または偶発的なタンパに対して重要なコードを保護できます。

注： **セキュアコードは、一般に小さく、ほとんど変更されませんが、非セキュアコードは、より公開度が高く、ファームウェアの更新が多い傾向があります。**

3.5.1 TrustZoneセキュリティアーキテクチャ

Armv8-M TrustZoneテクノロジーは、包括的なハードウェアアーキテクチャであり、プロセッサおよびシステム全体に及ぶ包括的でホリスティックな保護を開発者に提示します。デバイスのTrustZoneハードウェア機能は、次のとおりです。

- 関連するセキュア割込みにより、新しいプロセッサのセキュア状態を有効にする、Cortex-M33のArmv8-Mメインラインセキュリティ拡張
- Cortex-M33の8つのセキュリティ属性ユニット（SAU）領域を使用した、メモリおよびペリフェラルのTrustZoneへの動的割り当て
- TrustZone 保護を Cortex-M33 以外のシステムのマスタから着信するトランザクションに対して拡張する、グローバルTrustZoneフレームワーク（GTZC）
- TrustZone対応の内蔵Flashメモリおよびペリフェラル

注： **TrustZoneセキュリティはFlash_OPTRレジスタのTZENオプションビットで有効化されます。**

3.5.2 Cortex-M33のArmv8-Mセキュリティ拡張

Cortex-M33のArmセキュリティ拡張は、変革ではなく進化です。Cortex-M4など、以前のCortex-Mサブファミリのプログラマモデルを使用します。Armv8-MとArmv7-Mは、同じ32bitアーキテクチャと、MPUで保護された同じメモリ·マップ·リソースを使用しているため、構造上似ています。Armv8-Mはネスト化されたベクタ割込みコントローラ（NVIC）も使用しています。

STM32U5シリーズデバイスでのArmv8-MTrustZone実装は、以下の機能で構成されています。

- セキュア/非セキュアワールドの変更をトリガするために専用の例外ルーチンを使用するArmv8-ATrustZoneとは異なり、追加のコード/サイクルのオーバーヘッドがほとんどない、新しいプロセッサ状態
- 共有4GBのアドレス空間の2つのメモリマップビュー
- セキュアと非セキュアの両方のドメインの低割込み遅延と、セキュリティのグループ化と優先設定用の新しい割込み設定
- セキュアおよび非セキュア例外用の個別の例外ベクタテーブル
- マイクロコード化されたコンテキスト保持
- スタック限界チェッカー付きのスタックポインタを含む、セキュア/非セキュア状態にまたがる特定のレジスタのバンキング
- 次のプログラム可能なCortex-M33コンポーネントのバンキング（セキュアと非セキュアの独立した2つのユニット）:
  - SysTickタイマ
  - MPU設定レジスタ（セキュアまたは非セキュアな、それぞれ8つのMPU領域）
  - システム制御ブロック（SCB）レジスタの一部
- セキュリティ違反を処理するための新しいシステム例外（SecureFault）
- セクション 3.11で定義されている設定可能なデバッグサポート

詳細については、（PM0264）を参照してください。

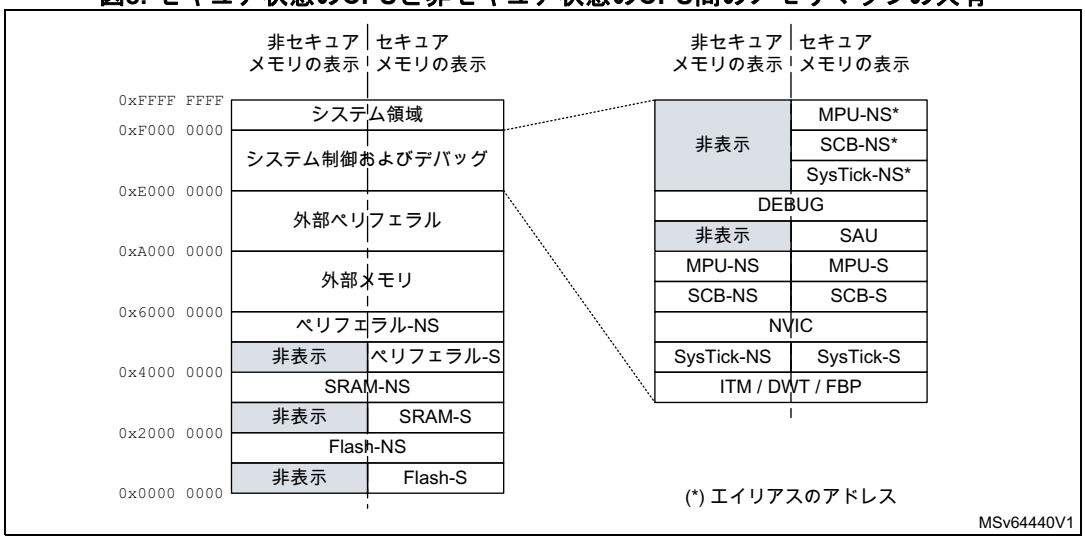
3.5.3 IDAU/SAUを使用したメモリおよびペリフェラルの割り当て

セキュリティ属性

図8に示されているように、Armv8-M非セキュアメモリの表示はArmv7-M（Cortex-M4で見られる）と同様ですが、セキュアメモリが非表示である点が異なります。セキュアメモリの表示は、Cortexプロセッサがセキュア状態で実行されているときにのみアクセス可能なFlashメモリ、SRAM、およびペリフェラルを示しています。

下の図は、セキュアコードによるSAU設定後に表示される32bitのアドレス空間を示しています。

図8. セキュア状態のCPUと非セキュア状態のCPU間のメモリマップの共有

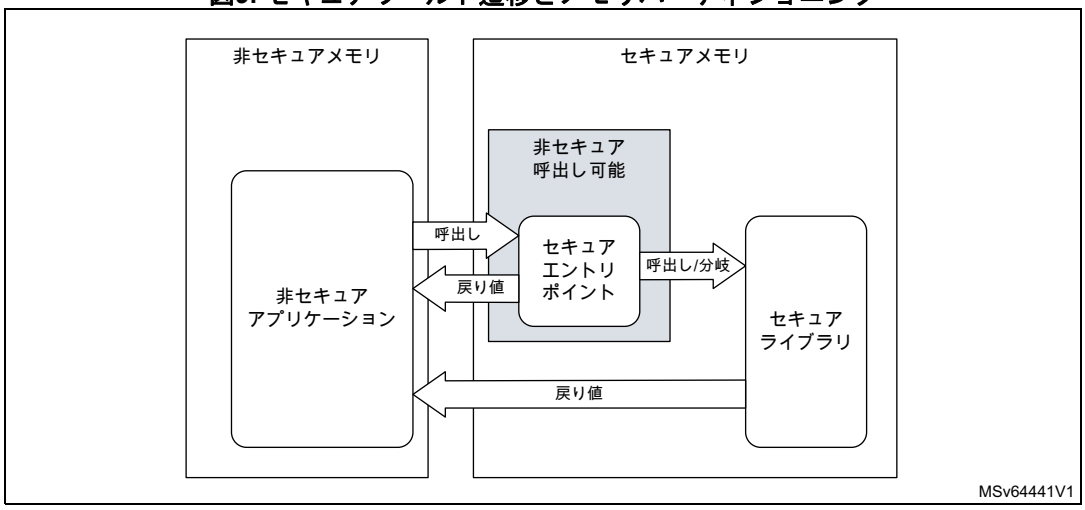


Cortexプロセッサの状態（および関連する権利）は、それが実行されるメモリ領域に割り当てられたセキュリティ属性に依存します。

- 非セキュア状態のプロセッサは、非セキュア（NS）なプログラムメモリからのみ実行されますが、セキュア状態のプロセッサは、セキュア（S）なプログラムメモリからのみ実行されます。
- セキュア状態での実行中、プロセッサはSとNSの両方のメモリのデータにアクセスできます。非セキュア状態で実行する際、CPUは非セキュアメモリに制限されます。

セキュアワールドへの遷移を管理するため、開発者は、セキュアライブラリへの有効なエントリ･ポイントを含む、非セキュア呼出し可能（NSC）領域を作成する必要があります。これらのエントリ･ポイントの最初の命令は、安全な機能を呼び出すために非セキュアコードが使用する、新しいセキュアゲート（SG）命令でなければなりません（下の図を参照）。

図9. セキュアワールド遷移とメモリパーティショニング



セキュリティ属性のプログラミング

Cortex-M33では、静的実装によって定義された属性ユニット（IDAU）は、プログラム可能なセキュリティ属性ユニット（SAU）と連携して動作して、次の表に示すように、特定のセキュリティ属性（S、NS、またはNSC）を特定のアドレスに割り当てます。

表8.IDAUおよびSAUによるセキュリティ属性の設定

IDAUセキュリティ属性	SAUセキュリティ属性(1)	最終的なセキュリティ属性
非セキュア	セキュア	セキュア
	セキュア-NSC	セキュア-NSC
	非セキュア	非セキュア
セキュア-NSC	セキュア	セキュア
	非セキュア	セキュア-NSC

1. 定義された領域は32バイト境界に揃えられます。

SAUは、セキュア特権状態のCortex-M33でのみ設定できます。TrustZoneが有効なとき、SAUは、デフォルトですべてのアドレスをセキュア（S）として定義します。セキュア･ブート･アプリケーションはその後、前の表に示しているように、SAUをプログラムして、NSCまたはNS領域を作成できます。

注： **SAU/IDAU設定はCortex-M33にのみ適用されます。DMAなどの他のマスタは、これらのポリシーの影響を受けません。**

SAU領域でカバーされないメモリ空間は、セキュアなものとして固定されます。

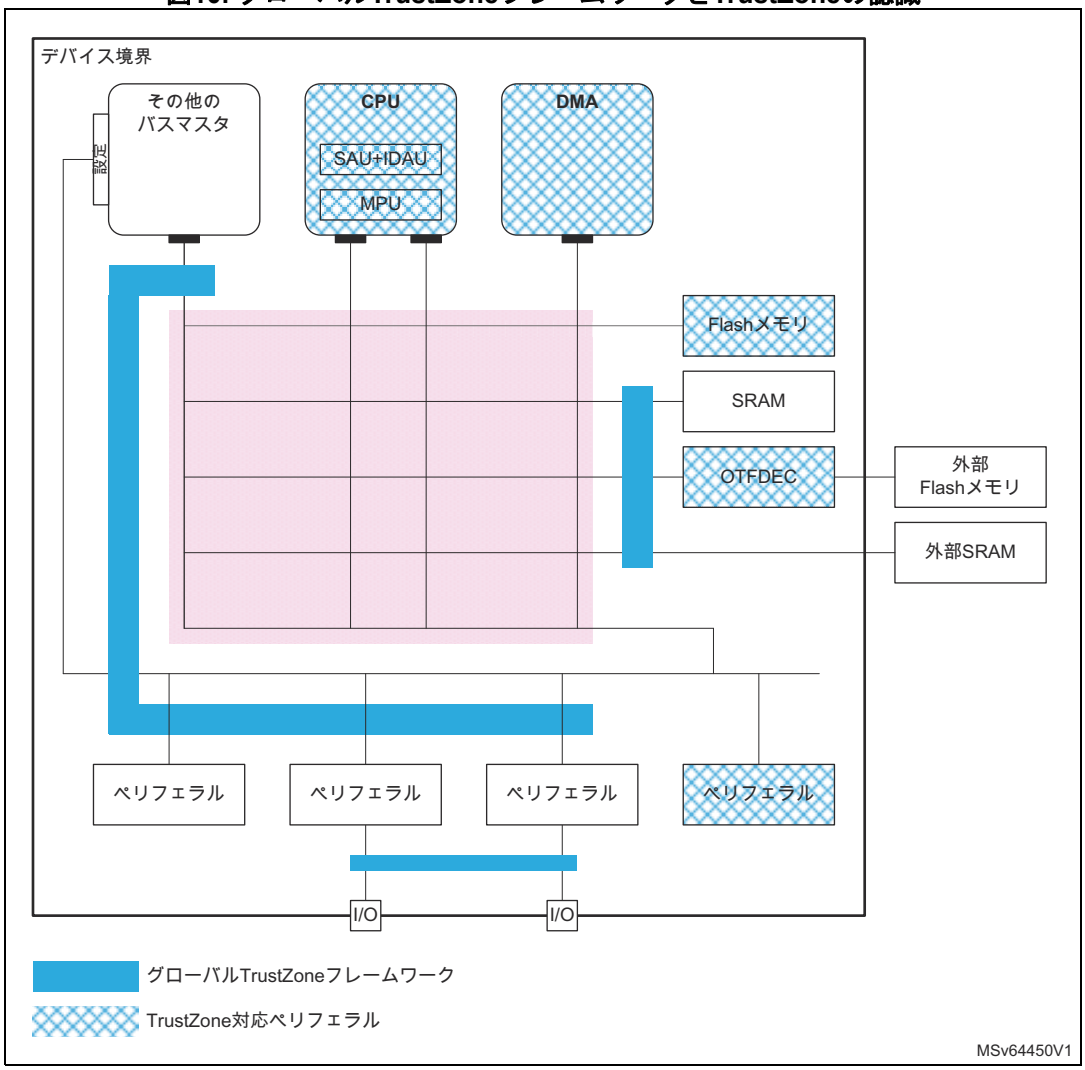
IDAU/SAUを使用したメモリのセキュリティ属性の詳細については、アプリケーション･ノート「TrustZone features on STM32L5 and STM32U5 Series」（AN5347）を参照してください。

3.5.4 GTZCを使用したメモリおよびペリフェラルの割り当て

グローバルTrustZoneフレームワークアーキテクチャ

デバイスは、Cortex-M33のArmv8-M TrustZoneセキュリティ拡張の上に、セキュアワールドと非セキュアワールドの分離を柔軟に補強する、相補的なセキュリティ機能を内蔵しています。SAU/IDAUとは異なり、GTZCは、Cortex-M33以外のマスタから着信する非セキュアなトランザクションから、レガシー･メモリとペリフェラルを保護できます。

図10. グローバルTrustZoneフレームワークとTrustZoneの認識



TZSCによるペリフェラルのセキュリティ保護

TrustZoneセキュリティがアクティブな場合、前の図に示すように、ペリフェラルは、GTZCのTZSCを通じてセキュリティ保護可能か、ネイティブでTrustZoneを認識するかのいずれかです。

- セキュリティ保護可能なペリフェラルまたはメモリは、TrustZone　セキュリティコントローラ（TZSC）によって制御されるAHB/APBファイアウォールゲートによって保護されます。
- TrustZoneを認識するペリフェラルまたはメモリは、AHBまたはAPB相互接続に直接接続され、セキュアレジスタのサブセットやセキュアメモリ領域などの、TrustZone固有の動作を実装します。

セキュリティ保護可能なペリフェラルがGTZCによりセキュア専用にされているとき、このペリフェラルが相互接続のマスタの場合は（SDMMCなど）、セキュアなトランザクションを自動的に発行します。SDMMCは、セキュリティ保護可能なマスタの例です。Cortex-M33やDMAなど、TrustZone対応AHBマスタは、GTZCとは無関係に、セキュリティモードに従って、AHB相互接続でセキュア信号を駆動します。

注：　**TrustZoneのように、ペリフェラルをTZSCで特権専用にできます（セクション 3.6.2を参照）。この場合、このペリフェラルが相互接続のマスタであれば、特権トランザクションは自動的に発行されます。**

TZSCおよびMPCBBによるメモリのセキュリティ保護

GTZCのTZSCブロックは、MPCWMリソースをプログラミングして、セキュリティ保護可能な外部メモリのすべてのセキュリティと特権を管理する機能を備えています。

次の表に、実装例を示します。STM32U5シリーズデバイスのその他すべての実装については、セクション 5.3：GTZCの実装を参照してください。

表9.MPCWMxリソース

メモリ	MPCリソース	フィルタリングのタイプ	領域数	デフォルトセキュリティ	オンザフライ復号(1)
OCTOSPI1	MPCWM1	非セキュアな特権または非特権領域（ウォーターマーク）	2	セキュア特権(2)	あり
FMC_NORバンク	MPCWM2		2		なし
FMC_NANDバンク	MPCWM3		1		
バックアップSRAM（BKPSRAM）	MPCWM4		1		
OCTOSPI2	MPCWM5		2		あり
HSPI1	MPCWM6		2		なし

1. OTFDECの使用。
2. デバイスでTrustZoneが有効であることを前提とし、そうでない場合は、非セキュア非特権。

GTZC内のMPCBBリソースは、内蔵SRAMブロックのセキュリティと特権を設定する機能を備えています。

次の表に、実装例を示します。STM32U5シリーズデバイスのその他すべての実装については、セクション 5.3：GTZCの実装を参照してください。

表10.MPCBBxリソース

メモリ	MPCリソース	フィルタリングのタイプ	メモリサイズ（KB）	ブロックサイズ（バイト）	スーパーブロックの数	デフォルトセキュリティ
SRAM1	GTZC1_MPCBB1	セキュリティと特権のブロックベースの管理	768	512(1)	48	セキュア特権(2)
SRAM2	GTZC1_MPCBB2		64		4	
SRAM3	GTZC1_MPCBB3		832		52	
SRAM4	GTZC2_MPCBB4		16		1	
SRAM5	GTZC1_MPCBB5		832		52	
SRAM6	GTZC1_MPCBB6		512		32	

1. ブロックは、32の連続したブロックから成るスーパーブロックにグループ化され、設定ロックを管理します。
2. デバイスでTrustZoneが有効であることを前提とし、そうでない場合は、非セキュア非特権。

GTZC設定の適用

TZSCおよびMPCBBブロックは、次のいずれかの方法で使用できます。

- セキュア·ブート時に静的にプログラムされ、ロックされて、その後は変更されない
- 特定のアプリケーションコードまたはリアルタイム·カーネルを使用して動的に再プログラムされる

動的オプションが選択され、設定がロックされていないとき：

- MPCBB セキュアブロックまたは MPCWM 非セキュア領域サイズは、セキュアソフトウェアによって変更できます。このソフトウェアはMPCWMに対する特権を持つ必要があり、特定のブロックが特権専用でない場合は、非特権になります。
- 各ペリフェラルのセキュア（それぞれ特権）状態は、GTZC_TZSC_SECCFRGx（それぞれGTZC_TZSC_PRIVCFGRx）レジスタに書き込むことで変更できます。

TZSCによるペリフェラルのセキュリティ保護

GTZC内のTZSCブロックは、セキュリティ保護可能なすべてのペリフェラルのセキュリティと特権を管理する機能を備えています。これらのペリフェラルのリストは、セクション 5: グローバルTrustZoneコントローラ（GTZC）にあります。

注： **TrustZoneが無効なときでも、リソース分離ハードウェアGTZCを使用して、ペリフェラルを特権コード専用に分離できます（セクション 3.6.2を参照）。**

TrustZoneが有効なときは、ペリフェラルはリセット後に非セキュアおよび非特権として設定されます。

TrustZone対応ペリフェラル

デバイスには、以下のTrustZone対応ペリフェラルが含まれます。

- LPGPIOオルタネート機能で設定されているかどうかにかかわらず、GPIOA~GPIOJ
- GTZCx_MPCBB、GTZCx_TZIC、およびGTZCx_TZSC（GTZCブロック）
- TZEN = 1の場合、セキュアでのみ書込み可能なOTFDEC1/2
- EXTI
- Flashメモリ
- RCCおよびPWR
- GPDMAおよびLPDMA
- SYSCFGレジスタ
- RTCおよびTAMP
- MCUデバッグユニットDBGMCU
- ICACHEおよびDCACHE1

それらのペリフェラルへの不正アクセスをTZICレジスタを介して監視する方法は、セクション 5 : グローバルTrustZoneコントローラ（GTZC）で説明されています。

詳細については、セクション3.5.5を参照してください。

TrustZone不正アクセスコントローラ（TZIC）

GTZC内のTZICブロックは、GTZCまたはTrustZoneを認識するペリフェラルによって保護されているソースから発生したすべての不正アクセスイベントを収集して、NVICに対して1つのグローバル·セキュア割込みを生成します。

TZICは、システムのTrustZoneが有効（TZEN = 1）なときのみ使用可能です。TZICレジスタへのアクセスはすべて、セキュアおよび特権である必要があります。

各不正イベントソースについて、ステータスフラグとクリアビットがあります。各不正イベントをマスクして、NVICに対する割込みを生成しないようにできます。

注： **デフォルトでは、すべてのイベントがマスクされています。**

3.5.5 TrustZone対応ペリフェラルのセキュリティの管理

このセクションでは、前のセクションでリストされたTrustZone対応ペリフェラルでのセキュリティの実装方法について詳しく説明します。

内蔵Flashメモリ

TrustZoneセキュリティがオプションバイト（TZEN = 1）によって有効化されている場合、Flashメモリ全体がリセット後にセキュアになり、下の図に示す以下の保護がアプリケーションで使用可能になります。

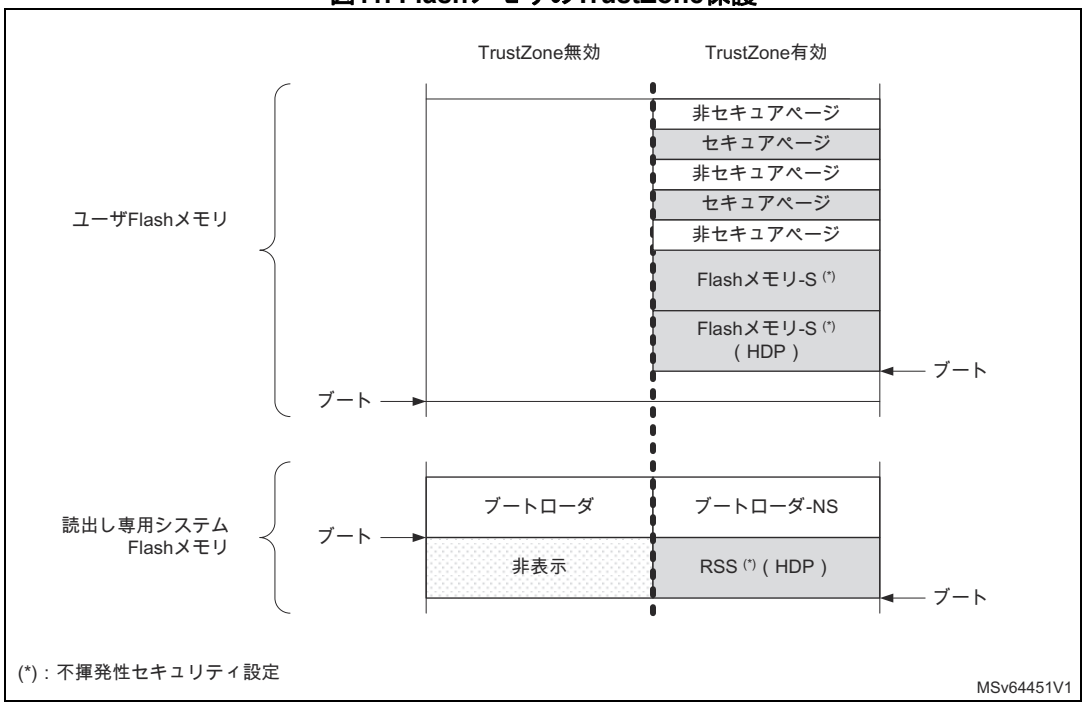
- 不揮発性ユーザセキュア領域、不揮発性セキュアユーザオプションバイトで定義
  - ウォーターマークベースのセキュア専用領域（x2）
  - セキュア非表示保護（HDP）領域、ブート後は厳格に非表示（x2）
- 揮発性ユーザセキュアページ、揮発性セキュアレジスタで定義（リセット後は失われる）
  - 非セキュアとして設定されたページ（ウォーターマークベースのセキュア専用領域外など）は、ブロックベースの設定レジスタを使用して、オンザフライでセキュアとして設定できます。

注： **すべての領域がFlashメモリのページ単位で配置されます。**

Flashメモリ領域は、Cortex-M33 IDAU/SAUで非セキュアとしてタグ付けされていても、セキュアとして設定できます。この場合、CPUによるFlashメモリへの非セキュアアクセスは拒否されます。

消去またはプログラム操作は、セキュア（もしくは非セキュア）コードで利用可能です。このコードは、セキュア（もしくは非セキュア）ページまたはメモリでのみ使用されます。少なくとも1つのページがセキュアの場合、Flashメモリはセキュアとみなされます。

図11. FlashメモリのTrustZone保護



上記のように、TrustZoneが有効なときは（TZEN = 1）、アプリケーションコードは、Flashメモリのウォーターマークベースのセキュア領域の一部であるHDP領域を使用できます。実際、アプリケーションがHDPx_ACCDISビットをセットすると、このHDP領域でのデータの読取り、書込み、および命令フェッチは、次のシステムリセットまで拒否されます。

たとえば、セキュアFlashメモリのHDP領域内のソフトウェアコードは一度だけ実行でき、この領域へのその後のアクセスは、次のシステムリセットまで拒否されます。また、アクティブなHDP領域に属するFlashメモリのページは、それ以上消去できなくなります。

TrustZoneが無効なときは（TZEN = 0）、揮発性/不揮発性セキュア領域機能が無効化され、すべてのセキュアレジスタはRAZ/WIです。

詳細は、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。

オンザフライ暗号化/復号（OTFDEC）

TrustZoneセキュリティが有効なとき（TZEN = 1）、OTFDECはセキュアアプリケーションによってのみ初期化できます。4つの暗号化領域のそれぞれは、設定が確定されると、次のパワーオンリセットまで、書込みロックされます。

注： **アプリケーション（セキュアまたは非セキュア）は、ペリフェラルレジスタを読み取ることにより、各OTFDEC領域の初期化コンテキスト（キーのCRCを含む）を確認できます。**

各OTFDEC内のキーレジスタは書込み専用です。

この暗号化エンジンの詳細については、セクション3.9.3を参照してください。

ダイレクトメモリアクセスコントローラ（LPDMAおよびGPDMA）

DMAチャネルxがセキュアとして定義されているとき（LP/GPDMA_SECCFGRのSECx = 1）、セキュア·アプリケーションは、LP/GPDMA_CxTR1のSSECおよびDSECビットを使用して、ソースまたはデスティネーション転送をセキュアまたは非セキュアとして個別に設定できます。

次の表に、各DMAチャネルで使用可能なセキュリティオプションの概要を示します。

表11. DMAチャネルの使用（セキュリティ）(1)

デスティネーションタイプ	セキュアDMAチャネルx（SECx = 1）		非セキュアDMAチャネルy（SECy = 0）	
	セキュアソース	非セキュア·ソース	セキュアソース	非セキュア·ソース
セキュアデスティネーション	OK	OK(2)	転送はブロックされる	
非セキュア·デスティネーション	OK(3)	OK(4)	転送はブロックされる	OK

1. 転送がブロックされると、転送は完了しますが、対応する書込みは無視され、読取りは0を返します。また、TZICへの不正アクセスイベントは、ソースまたはデスティネーションとして使用されるメモリ/ペリフェラルによって自動的にトリガされます。
2. ソースがメモリの場合、転送はSSEC = 0の場合のみ可能であり、そうでない場合はブロックされます。
3. デスティネーションがメモリの場合、転送はDSEC = 0の場合のみ可能であり、そうでない場合はブロックされます。
4. 転送がメモリ間の場合、転送はSSEC = 0でDSEC = 0の場合のみ可能であり、そうでない場合はブロックされます。

チャネルがセキュアとして設定されているとき：

- このチャネルに割り当てられたレジスタ（LP/GPDMA_SECCFGR、LP/GPDMA_PRIVCFGR、およびLP/GPDMA_RCFGLOCKRを除く）は、0として読み出されます。書込みは非セキュアアクセスの場合無視されます。セキュア不正アクセスイベントは、TZICペリフェラルに対してトリガされることもあります。
  - LP/GPDMA_SECCFGR および LP/GPDMA_RCFGLOCKR への書込みはセキュアである必要があります。LP/GPDMA_PRIVCFGR内の各ビットについて、LP/GPDMA_SECCFGR内の対応するビットがセットされている場合、書込みはセキュアでなければなりません。
- リンクリストモードでは、メモリからの次のリンクリストデータ構造のロードは、セキュア転送で行われます。
- 非セキュア状態に切り替えるとき、セキュア·アプリケーションは、切り替えを行う前に、チャネルをアボートするか、セキュアチャネルが完了するまで待つ必要があります。

注： **DMAセキュアチャネルは、TrustZoneが無効なときには使用できません。**

チャネルが非セキュアとして設定されているとき、リンクリスト·モードでは、メモリからの次のリンクリスト·データ構造のロードは、非セキュア転送で行われます。

詳細については、セクション 18：低電力ダイレクトメモリアクセスコントローラ（LPDMA）およびセクション 17：汎用ダイレクト·メモリ·アクセス·コントローラ（GPDMA）を参照してください。

電源制御（PWR）

TrustZoneセキュリティが有効なとき（TZEN = 1）、選択したPWRレジスタはPWR_SECCFGRによって保護することができ、それにより以下のPWR機能を保護できます。

- 低電力モードのセットアップ
- ウェイクアップ（WKUP）ピン
- 電圧の検出と監視
- バックアップドメイン制御

他のPWR設定ビットは、次のときにセキュアになります。

- システムクロックの選択がRCCでセキュアな場合：電圧スケーリング（VOS）とレギュレータブースタ（BOOSTEN）の設定がセキュアになります。
- GPIOがセキュアとして設定されている場合：STANDBYモードのプルアップ/プルダウン設定の対応するビットがセキュアになります。
- USB Type-C/USB Power Deliveryインタフェース（UCPD）がTZSCでセキュアとして設定されている場合：PWR_UCPDRレジスタがセキュアになります。

詳細については、セクション 10：電源制御（PWR）を参照してください。

セキュアクロックおよびリセット（RCC）

TrustZoneセキュリティが有効で（TZEN = 1）、RCCでセキュリティが有効なとき、ペリフェラルクロックおよびリセットを制御するビットはTrustZoneを認識するようになります。

- ペリフェラルがセキュリティ保護可能で、TZSC でセキュアとしてプログラムされている場合、ペリフェラルクロックおよびリセットビットはセキュアになります。
- ペリフェラルがTrustZoneを認識するようになると、ペリフェラルクロックとリセットビットは、1つ以上の機能がペリフェラル内でセキュアに設定され次第、セキュアになります。

注： **セキュリティ保護可能でTrustZoneを認識するペリフェラルのリストについては、セクション3.5.4を参照してください。**

SHSI設定とステータスビットは、SAESがセキュアに設定されると、セキュアになります。

また、以下の設定は、RCC_SECCFGRを使用してセキュア専用にできます。

- 外部クロック（HSEまたはLSEなど）、内部オシレータ（HSI、MSI、またはLSIなど）
- メインPLLおよびAHBプリスケーラ
- システムクロックソース選択
- リセットフラグのクリア
- 自動内部オシレータウェイクアップ設定

詳細については、セクション 11：リセットおよびクロック制御（RCC）を参照してください。

リアルタイム･クロック（RTC）

すべてのTrustZoneを認識するペリフェラルと同様に、セキュアなRTCレジスタへの非セキュア読出し/書込みアクセスはRAZ/WIです。RTC不正アクセスイベントがTZICで有効化されたときにセキュア不正アクセス割込みをトリガする不正アクセスインベントも生成されます。

バックアップドメインのパワーオンリセット後、すべてのRTCレジスタは、セキュアモードと非セキュアモードの両方で読み書きできます。セキュア･ブート･コードは、RTC_SECCFGRを使用してこのセキュリティ･セットアップを変更し、アラームA、アラームB、ウェイクアップタイマ、およびタイムスタンプのレジスタをセキュアまたは非セキュアにできます。

SECビットがセキュア専用RTC_SECCFGRでセットされているとき：

- RTCレジスタへの書込みはセキュアモードでのみ可能です。
- RTC_SECCFGR、RTC_PRIVCFGR、RTC_MISR、RTC_TR、RTC_DR、RTC_SSR、RTC_PRER、およびRTC_CALRの読出しは、セキュアモードと非セキュアモードで常に可能です。その他のすべてのRTCレジスタは、セキュアモードでのみ読出しできます。

SECがセキュア専用RTC_SECCFGRでクリアされた場合、専用の制御ビットINITSEC、CALSEC、TSSEC、WUTSEC、ALRASEC、およびALRBSECをセットすることで、一部のRTCレジスタへのセキュアモードでのアクセスを制限できます。

注： **RTCセキュリティ設定は、システムリセットの影響を受けません。**

詳細は、セクション 63：リアルタイムクロック（RTC）を参照してください。

タンパおよびバックアップレジスタ（TAMP）

すべてのTrustZoneを認識するペリフェラルと同様に、セキュアなTAMPレジスタへの非セキュア読出し/書込みアクセスはRAZ/WIです。TAMP不正アクセスイベントがTZICで有効化されたときにセキュア不正アクセス割込みをトリガする不正アクセスインベントも生成されます。

バックアップドメインのパワーオンリセット後、すべてのTAMPレジスタは、セキュアモードと非セキュアモードの両方で読み書きできます。セキュア·ブート·コードは、TAMP_SECCFGRレジスタを使用してこのセキュリティ·セットアップを変更し、一部のレジスタを必要に応じてセキュアまたは非セキュアにできます。

TAMP_SECCFGRでTAMPSECがセットされているとき：

- TAMPレジスタへの書込みはセキュアモードでのみ可能です。バックアップレジスタには専用の書込み保護があります（下記を参照）。
- アクセスが非セキュアな場合、TAMP レジスタ（TAMP_SECCFGR、TAMP_PRIVCFGR、およびTAMP_MISRを除く）を読み出すと、ゼロが返されます。バックアップレジスタには専用の読出し保護があります（下記を参照）。

アプリケーションは、以下のこともできます。

- TAMP_SECCFGRセキュアレジスタのCNT1SECビットをセットすることにより、TAMP_COUNTRレジスタを読出しおよび書込みセキュア専用にできます。
- TAMP_SECCFGR の BKPRWSEC[7:0] および BKPWSEC[7:0] ビットフィールドを使用して設定された3つの保護ゾーンのうちの2つについて、バックアップレジスタのセキュリティを強化します。
  - 保護ゾーン1は、読出し非セキュア、書込み非セキュアです。
  - 保護ゾーン2は、読出し非セキュア、書込みセキュアです。
  - 保護ゾーン3は、読出しセキュア、書込みセキュアです。

注： **TAMPセキュリティ設定は、システムリセットの影響を受けません。**

詳細は、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照してください。

汎用I/O（GPIO）

TrustZoneセキュリティが有効なとき（TZEN = 1）、GPIOポートの各I/Oピンは、GPIOx_SECCFGRレジスタによって個別にセキュアとして設定できます。セキュアアプリケーションのみが、GPIOx_SECCFGRレジスタに書き込むことができます。ブート後、GPIOの各I/Oピンはセキュアとして設定されます。

I/Oピンがセキュアとして設定されているとき、オルタネート機能（AF）、モード選択（MODE）、およびI/Oデータの対応する設定ビットは、非セキュアアクセスの場合、RAZ/WIです。

セキュアペリフェラルによって管理されたI/Oとの間で転送されるデータを保護するためにデジタル·オルタネート機能が使用されるとき（入力/出力モード）、デバイスはペリフェラルと割り当てられたI/Oの間のパスに、セキュア·オルタネート機能ゲートを追加します。

- ペリフェラルがセキュアな場合、データの入出力を可能にするために、I/O ピンもセキュアである必要があります。
- ペリフェラルがセキュアでない場合、I/Oピンの状態に関係なく、接続は許されます。

次の表に、オルタネート機能として使用される、GPIOポート周辺のTrustZoneを認識するロジックの概要を示します。

表12.ペリフェラルと割り当てられたI/O間のセキュアオルタネート機能

セキュリティ設定		オルタネート機能ロジック		コメント
ペリフェラル	割り当てられたI/Oピン	入力	出力	
セキュア	セキュア	I/Oデータ	ペリフェラルデータ	-
非セキュア				リセット解除設定
セキュア	非セキュア	ゼロ	ゼロ	-
非セキュア		I/Oデータ	ペリフェラルデータ	

アナログスイッチ付きのアナログ機能が使用されるとき、次の表にリストされたペリフェラルへの接続は、ペリフェラルが非セキュアでI/Oがセキュアな場合に、ハードウェアによってブロックされます。

表13.セキュアI/Oに接続されていない非セキュアペリフェラル機能

ペリフェラル	アナログ機能(1)	入力	出力	ペリフェラルまたは機能をセキュアとして設定する方法
ADC12	ADC12_INy（y = 1?17）	X	-	GTZC1_TZSC_SECCFGR3のADC12SECをセット
ADC4	ADC4_INy（y = 1~19）			GTZC2_TZSC_SECCFGR1のADC4SECをセット
OPAMPx（x = 1、2）	OPAMPx_VINy（x = 1、2/y = 1、2）			GTZC2_TZSC_SECCFGR1のOPAMPSECをセット
COMPx（x=1、2）	COMPx_INy（x = 1/y = 1~3） COMPx_INy（x = 2/y = 1、2）			GTZC2_TZSC_SECCFGR1のCOMPSECをセット

1. パッケージ上の信号/機能に対応するI/Oを見つけるために使用されます（製品データシートを参照）。

最後に、GPIOとセキュリティに関して、次の表に、TrustZoneにリンクされたハードウェア保護のないペリフェラル機能のリストをまとめます。リストされている信号（入力または出力）は、I/Oがセキュアとして設定され、関連するペリフェラルが非セキュアのときにはブロックされません。

たとえば、セキュア·アプリケーションがPA4をLPTIM2_OUTとして使用するためにセキュアとして設定したとき、DACが非セキュアの場合はデータをPA4に出力するようプログラムできますが、これによりセキュア·アプリケーションが誤動作することがあります。

同様に、セキュア·アプリケーションがPA0をUART4_TXとして使用するためにセキュアとして設定したとき、TAMPが非セキュアの場合はUSART入力トラフィックをTAMP_IN機能によってキャプチャするようにプログラムできます。

次の表で説明されている各ケースについて、セキュア·アプリケーションが、データの整合性や機密性への潜在的な影響が重大かどうかを判断することが重要です。たとえば、上記で説明されているUSART状況が許容されない場合（セキュアUSARTでのデータ遷移は機密である）、セキュア·アプリケーションは、使用しない場合でも、TAMPをセキュアとして設定する必要があります。

注：　ペリフェラルをセキュアにする方法の概要を、次の表の右欄に示します。

表14.セキュアI/Oに接続可能な非セキュアペリフェラル機能

ペリフェラル	信号(1)	入力	出力	ペリフェラルまたは機能をセキュアとして設定する方法
DAC	DAC1_OUTx（x = 1、2）	-	X	GTZC2_TZSC_SECCFGR1のDAC1SECをセットします。
OTG_FSOTG_HS	OTG_FS_VBUS OTG_HS_VBUS	X	X	GTZC1_TZSC_SECCFGR3のOTGSECをセットします。
UCPD	UCPD1_CCx（x = 1、2）	X	X	GTZC1_TZSC_SECCFGR1のUCPD1SECをセットします。
	UCPD1_DBx（x = 1、2）	X	-	
TAMP	TAMP_INx（x = 1?8）	X	-	TAMP_SECCFGRのTAMPSECをセットします。
	TAMP_OUTx（x = 1?8）	-	X	
RTC	RTC_OUTx（x = 1、2）	-	X	RTC_SECCFGRのSECをセットします。
	RTC_TS	X	-	RTC_SECCFGRのTSSECをセットします。
PWR	WKUPx（x = 1?8）	X	-	PWR_SECCFGRのWUPxSECをセットします。
RCC	LSCO	-	X	RCC_SECCFGRのLSESECをセットします。
EXTI	EXTIx（x = 0~22）	X	-	EXTI_SECCFGRのSECxビットをセットします。

1. パッケージ上の信号/機能に対応するI/Oを見つけるには、製品データシートを参照してください。

詳細については、セクション 13：汎用I/O（GPIO）を参照してください。

拡張割込みとイベントコントローラ（EXTI）

TrustZoneセキュリティが有効なとき（TZEN = 1）、EXTIはイベントレジスタビットが非セキュアアクセスによって変更されないように保護することができます。保護は、入力イベントごとにEXTI_SECCFGR1のレジスタビットによって個別に有効化できます。入力イベントがセキュアとして設定されているとき、セキュア·アプリケーションのみが設定を変更でき（該当する場合はセキュリティを含む）、この入力イベントのマスクを変更したり、ステータスをクリアしたりできます。

EXTI_SECCFGR1のセキュリティ設定は、EXTI_LOCKRでリセット後にグローバルにロックできます。

詳細は、セクション 23：拡張割込み／イベントコントローラ（EXTI）を参照してください。

システム設定コントローラ（SYSCFG）

すべてのTrustZoneを認識するペリフェラルと同様に、TrustZoneセキュリティが有効なとき（TZEN = 1）、セキュアなSYSCFGレジスタへの非セキュア読出し/書込みアクセスはRAZ/WIです。このようなアクセスにより、SYSCFG不正アクセスイベントがTZICでマスクされないときにセキュア不正アクセス割込みをトリガする不正アクセスインベントも生成されます。

詳細は、セクション 15：システム設定コントローラ（SYSCFG）を参照してください。

マイクロコントローラデバッグユニット（DBGMCU）

MCUデバッグコンポーネント（DBGMCU）は、デバッガによる以下のサポート機能を支援します。

- デバッグ中の低電力モード動作
- I^2C、IWDG、WWDG、タイマ、低電力タイマ、および LP/GPDMA チャネルに適用可能な、デバッグ中のペリフェラルフリーズ

DBGCMUはTrustZoneを認識するペリフェラルであり、次の表で説明されているように、制御レジスタへのアクセスを管理します。

表15.TrustZoneを認識するDBGMCUアクセス管理

デバッグプロファイル	ペリフェラルのステータス(1)	DBG_xx_STOP制御ビット	
		書込みアクセス	読出しアクセス
非セキュアな侵入（SPIDEN = 0）	NS	はい（S(2) またはNS）	はい（SまたはNS）
	S	なし（SまたはNS）	
セキュアな侵入（SPIDEN = 1）	NS	はい（SまたはNS）	
	S	はい（Sのみ）	

1. GTZCによって報告されるTrustZone対応ペリフェラルまたはDMAチャネル。
2. デバッガからのセキュアアクセスは、デバイスで非セキュアアクセスに変換されます。

詳細については、セクション 75.12 : マイクロコントローラデバッグユニット（DBGMCU）を参照してください。

3.5.6 TrustZoneセキュリティの有効化

TrustZoneは、すべてのSTM32U5シリーズデバイスでデフォルトで無効です。RDPレベルが0のとき、FLASH_OPTRのTZENユーザオプションビットを設定して有効化できます。一度TZENを0から1に変更すると、リセット後のデフォルトのセキュリティ状態は、必ず以下になります。

- CPUサブシステム
  - Cortex-M33はセキュア状態でリセットを抜けるため、ブートアドレスはセキュアメモリ領域を指すはずです。
  - すべての割込みソースはセキュアです（NVICで）。
  - IDAU/SAU経由でCPUによって表示されるメモリマップは完全にセキュアです。
- 内蔵Flashメモリ
  - Flashメモリの不揮発性セキュア領域（HDPゾーン付き）は、不揮発性レジスタFLASH_SECWMxR（x = 1、2）で定義されます。デフォルトのセキュアオプションバイトの設定は、すべてユーザFlashセキュアであり、HDP領域は定義されていません。
  - Flash　メモリのブロックベースの揮発性セキュリティ属性は非セキュアです。セキュア·ブート·コードは、このセットアップを変更し、ブロックをセキュアにできます。
- 内蔵SRAMメモリ
  - すべてのSRAMは、GTZC/MPCBBで定義されるようにセキュアです（セクション3.5.4を参照）。セキュア·ブート·コードは、このセキュリティ·セットアップを変更し、ブロックをセキュアまたは非セキュアにできます。
- 外部メモリ
  - FSMCおよびOCTOSPIに接続されたすべてのメモリデバイスは、GTZC/MPCWMで定義されるように、セキュアです（セクション3.5.4を参照）。セキュア·ブート·コードは、このセキュリティ·セットアップを変更し、コンポーネントをセキュアまたは非セキュアにできます。
- すべてのGPIOはセキュアです。
- すべてのLP/GPDMAチャネルは非セキュアです。
- バックアップレジスタは非セキュアです。

- ペリフェラルとGTZC：
  - セキュリティ保護可能なペリフェラルは、非セキュアかつ非特権です。
  - TrustZoneを認識するペリフェラルは、そのセキュア設定レジスタがセキュアなとき、非セキュアです。
  - GTZC/TZICにおけるすべての不正アクセス割込みは無効です。

注： **セキュリティ保護可能でTrustZoneを認識するペリフェラルのリストについては、セクション3.5.4を参照してください。**

3.5.7 TrustZoneセキュリティの無効化

一度TrustZoneを有効にすると、無効化はレベル0へのRDP回帰の間にのみ行えます。

注： **そのようなRDP回帰は内蔵メモリ（SRAM2、Flashメモリ）の消去と、OTFDECおよびすべての暗号化エンジンを含む、すべてのペリフェラルのリセットをトリガします。**

TrustZoneの無効化後、セクション3.5で言及されたほとんどの機能は使用できなくなります。

- 内蔵Flashメモリの不揮発性セキュア領域は、HDP領域も含めて無効になります。
- NVICは、非セキュア割込みのみを管理します。
- TrustZone対応ペリフェラルのすべてのセキュアレジスタはRAZ/WIです。

注： **TrustZoneが無効化されても、特権を使用するリソース分離は使用可能なままです（詳細については、セクション3.6.2を参照してください）。**

詳細については、アプリケーション·ノート（AN5347）を参照してください。

3.6 その他のリソースの分離

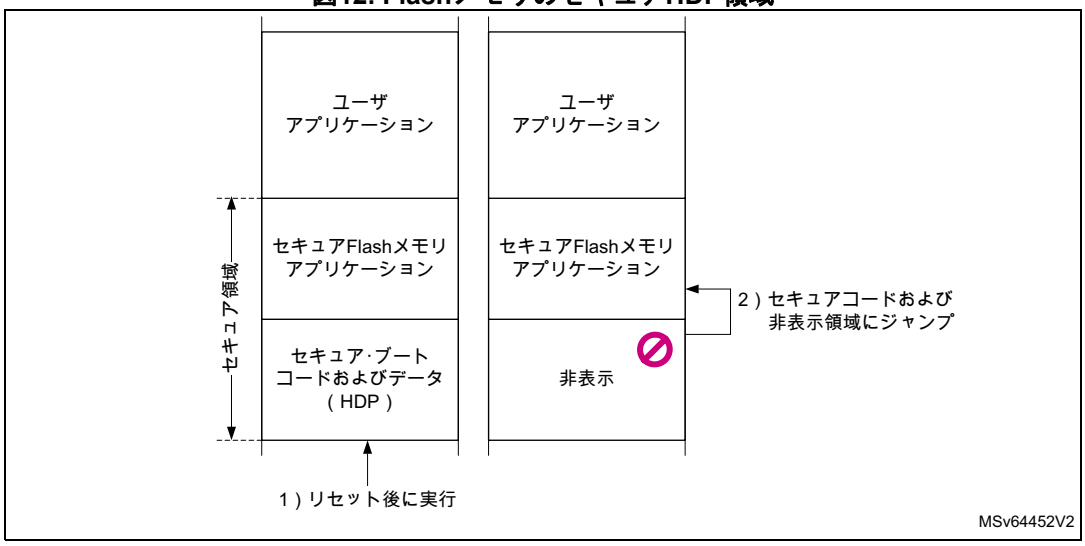
これらは、TrustZoneテクノロジーの上にもう1つの分離レベルを提供するハードウェアメカニズムです。

3.6.1 セキュア非表示保護（HDP）を使用した時間的分離

TrustZoneセキュリティが有効なとき（TZEN = 1）、内蔵Flashメモリでは各バンク（8KBのページ単位）のウォーターマークを付けたセキュア領域ごとにHDP領域を定義できます。このHDP領域で実行されるコードは、ブート後は次のシステムリセットまで、関連データやキーとともに、非表示にできます。

非表示保護の原理を、下の図に示します。

図12. Flashメモリのセキュアhdp領域



HDPxENとHDPx_ACCDISビット（x = 1、2）をセットすると、SECWMx_PSTRTおよびHDPx_PENDオプションバイトで定義された領域のデータの読出し、書込み、および命令フェッチが、次のデバイスのリセットまで拒否されます。

注： **バンク消去に書込み保護領域（WRP またはHDP 領域）が含まれていると、バンク消去はアボートされます。**

HDP領域は、非表示でない場合、およびRDPレベル≠2の場合に、セキュア·アプリケーションでサイズ変更できます。

3.6.2 Cortex特権モードを使用したリソース分離

セクション3.5で説明されているTrustZone分離と並行して、STM32U5シリーズデバイスのハードウェアとソフトウェアのリソースを分けることで、それらをCortex特権モードで実行しているソフトウェアに制限できます。

このハードウェア分離テクノロジーは、TrustZoneが無効なときでも（TZEN = 0）使用できるため、より公開度の高い非特権コードからの意図的または偶発的なタンパに対して重要なコードやデータを保護できます。

MPUを使用したメモリおよびペリフェラルの特権割当て

Cortex-M33 MPUにより、統合されたメモリが、それぞれ32バイトの倍数に調整された8つの領域に分けられます。各メモリ領域は、非特権ソフトウェアによって不適切にアクセスされたときに、フォールトを生成するようにプログラムできます。

GTZCを使用したメモリおよびペリフェラルの特権割当て

Cortex-M33マスタの場合、MPUによって提供される簡易分離を補足するために、GTZCは、ペリフェラルと選択されたメモリについて、特権タスクと非特権タスク間の分離を柔軟に補強します。

Cortex-M33以外のマスタについては、GTZCはそれらを非特権イニシエータとして割り当てて、特権として定義されたリソースをこのマスタから自動的に保護します。

- TZSCによるペリフェラルのセキュリティ保護（特権のみ）

  デバイスでは、ペリフェラルは、GTZCを通じてセキュリティ保護可能な特権専用か、ネィティブに特権対応かのいずれかです。

  – セキュリティ保護可能な特権専用のペリフェラルまたはメモリは､TZSCによって制御されるAHB/APBファイアウォールゲートによって保護されます。

  – 特権を認識するペリフェラルまたはメモリは、AHB または APB 相互接続に直接接続され、レジスタのサブセットやメモリ領域を特権専用とするなどの、固有の動作を実装します。

    そのようなペリフェラルがGTZCにより特権専用にされ、相互接続のマスタである場合（SDMMC）、特権トランザクションを自動的に発行します。GPDMAまたはLPDMAなどの特権を認識するマスタは、GTZCとは無関係に、内部特権モードに従って、AHB相互接続で特権信号を駆動します。

    セキュリティ保護可能なペリフェラルのリストは、セクション 5 : グローバルTrustZoneコントローラ（GTZC）にあります。

- TZSCおよびMPCBBによるメモリのセキュリティ保護（特権専用）

  GTZCのTZSCロジックは､セキュリティ保護可能な外部メモリのすべての特権レベルを管理し、セクション3.5.4で定義されているように、MPCWMリソースをプログラムする機能を備えています。

  同様に、GTZCは、セクション3.5.4で定義されているMPCBBリソースをプログラムし、内蔵SRAMブロックの特権レベルを設定する機能を備えています。

- エラー管理（特権専用）

  – 特権リソースへのアクセスを試みる非特権トランザクションはすべて不正アクセスとみなされます。不正非特権読出しおよび書込みアクセスについては、不正アクセスイベントは生成されません。

  – アドレス指定されたリソースはサイレントフェイル動作に従い、読出しにはすべてゼロデータを返し、書込みはすべて無視します。

  – 不正な非特権アクセスが発生すると、このアクセスが特権メモリまたはペリフェラルレジスタにアクセスする命令フェッチである場合を除き、バスエラーは生成されません。

特権対応ペリフェラルのセキュリティの管理

TrustZone対応ペリフェラルは、特権専用アクセスモードも実装します。特権保護は、TZEN = 0のときのみ有効です。

- 内蔵Flashメモリ

  デフォルトでは、すべての内蔵Flashレジスタは、特権モードと非特権モードの両方で読出しまたはプログラムが可能です。

  FLASH_PRIVCFGRのセキュア特権ビットSPRIVがセットされているとき、Flashセキュアレジスタの読み書きは、特権モードでのみ可能です。このビットへの書込みアクセスは、TrustZoneが無効な場合（TZEN = 0）、無視されます。

  FLASH_PRIVCFGRの非セキュア特権ビットNSPRIVがセットされているとき、Flash非セキュアレジスタの読み書きは、特権モードでのみ可能です。

  内蔵Flashメモリの特権保護に関して、デバイスは以下の機能を提供します。

  – システムFlashメモリは、特権モードと非特権モードの両方でアクセスできます。

  – 各ウォーターマークベースのセキュア領域は、セキュアHDP領域も含め、セキュア特権モードとセキュア非特権モードでアクセス可能です（該当する場合）。

  – 内蔵Flashメモリの各8KBのページは、ブロックベースの特権設定レジスタFLASH_PRIV1BBRxおよびFLASH_PRIV2BBRxを使用して、オンザフライで特権専用としてプログラムできます。非特権ページは、特権アクセスまたは非特権アクセスでアクセス可能です。

注： **ページの特権から非特権への切り替えによって、ページの内容が消去されることはありません。**

該当する場合、消去またはプログラム操作は、特権コードに対しては常に使用可能であり、非特権コードに対しては、非特権ページまたは非特権メモリについてのみ使用可能です。

- オンザフライ暗号化/復号（OTFDEC）

  OTFDEC_PRIVCFGRの権限ビットPRIVがセットされたとき、OTFDECは権限アプリケーションによってのみ初期化できます。

注： **OTFDEC_PRIVCFGRは、特権コードと非特権コードの両方によって読み出すことができます。**

- ダイレクトメモリアクセスコントローラ（LPDMAおよびGPDMA）

  DMAチャネルxが特権として定義されているとき（LP/GPDMA_PRIVCFGRのPRIVx = 1）、特権/非特権ソースまたはデスティネーションにアクセスするときには、特別な規則が適用されます。次の表に、これらの規則の概要を示します。

表16.DMAチャネルの使用（特権）

転送先	特権DMAチャネルx（PRIVx = 1）		非特権DMAチャネルy（PRIVy = 0）	
	特権ソース	非特権ソース	特権ソース	非特権ソース
特権	OK		転送はブロックされる(1)	
非特権			転送はブロックされる	OK

1. 転送がブロックされると、転送は完了しますが、対応する書込みは無視され、読取りは0を返します。

詳細については、セクション 18 : 低電力ダイレクトメモリアクセスコントローラ（LPDMA）およびセクション 17 : 汎用ダイレクト･メモリ･アクセス･コントローラ（GPDMA）を参照してください。

- 電源制御（PWR）

  デフォルトでは、パワーオンまたはシステムリセット後、PWR_PRIVCFGR以外のすべてのPWRレジスタは、特権モードと非特権モードの両方で読み書きできます。

  PWR_PRIVCFGRのセキュア特権ビットSPRIVがセットされているとき、PWRセキュリティ保護可能レジスタの読み書きは、特権モードでのみ可能です。このビットへの書込みアクセスは、TrustZoneが無効な場合（TZEN = 0）、無視されます。

  PWR_PRIVCFGRの非セキュア特権ビットNSPRIVがセットされているとき、PWR非セキュアレジスタの読み書きは、特権モードでのみ可能です。

  詳細については、セクション 10：電源制御（PWR）を参照してください。

- セキュアクロックおよびリセット（RCC）

  デフォルトでは、パワーオンまたはシステムリセット後、RCC_PRIVCFGR以外のすべてのRCCレジスタは、特権モードと非特権モードの両方で読み書きできます。

  RCC_PRIVCFGRのセキュア特権ビットSPRIVがセットされているとき、RCCセキュリティ保護可能なビットの読み書きは、特権モードでのみ可能です。このビットへの書込みアクセスは、TrustZoneが無効な場合（TZEN = 0）、無視されます。

  RCC_PRIVCFGRの非セキュア特権ビットNSPRIVがセットされているとき、RCC非セキュアビットの読み書きは、特権モードでのみ可能です。

  詳細については、セクション 11：リセットおよびクロック制御（RCC）を参照してください。

- リアルタイム·クロック（RTC）

  デフォルトでは、バックアップドメインリセット後、RTC_PRIVCFGR以外のすべてのRTCレジスタは、特権モードと非特権モードの両方で読み書きできます。

  PRIVビットが特権専用RTC_PRIVCFGRでセットされているとき：

  – RTCレジスタへの書込みは特権モードでのみ可能です。
  – RTC_SECCFGR、RTC_PRIVCFGR、RTC_TR、RTC_DR、RTC_SSR、RTC_PRER、およびRTC_CALRの読出しは、特権モードと非特権モードで常に可能です。

  その他すべてのRTCレジスタは、特権モードでのみ読み出せます。

  PRIV ビットが特権専用 RTC_PRIVCFGR レジスタでクリアされているときでも、専用のINITPRIV、CALPRIV、TSPRIV、WUTPRIV、ALRAPRV、またはALRBPRIVの制御ビットをセットすることで、一部のRTCレジスタの特権モードへのアクセスを制限することが可能です。

  詳細については、第セクション 63：リアルタイムクロック（RTC）章を参照してください。

- タンパおよびバックアップレジスタ（TAMP）

  デフォルトでは、任意のバックアップドメインリセット後、TAMP_PRIVCFGR以外のすべてのTAMPレジスタは、特権モードと非特権モードの両方で読み書きできます。

  PRIVビットが特権専用TAMP_PRIVCFGRでセットされているとき：

  – TAMP レジスタへの書込みは、独自の保護設定を持つバックアップレジスタと単調カウンタを除き、特権モードでのみ可能です。
  – TAMP_SECCFGRまたはTAMP_PRIVCFGRの読出しは、特権モードと非特権モードで常に可能です。他のTAMPレジスタはすべて、独自の保護設定を持つバックアップレジスタと単調カウンタを除き、特権モードでのみ読み出すことができます。

  アプリケーションは、以下のこともできます。

  – TAMP_PRIVCFGRのCNTPRIVビットをセットすることにより、TAMP_COUNT1Rレジスタの読み書きを特権専用にできます。

– TAMP_PRIVCFGRのBKPRWPRIVおよびBKPWPRIVビットを使用して、バックアップレジスタの3つの保護ゾーンのうちの2つについてセキュリティを高めます。

- 保護ゾーン1を読出し特権、書込み特権にします。
- 保護ゾーン2を読出し特権または非特権、書込み特権にします。
- 保護ゾーン3は、常に読み書き特権または非特権です。

- 汎用I/O（GPIO）

  すべてのGPIOレジスタは、セキュリティ状態（セキュアまたは非セキュア）に関係なく、特権および非特権アクセスで読み書きできます。

- 拡張割込みとイベントコントローラ（EXTI）

  EXTIペリフェラルは、イベントレジスタビットを非特権アクセスによる変更から保護できます。保護は、特権専用のEXTI_PRIVCFGR1レジスタのレジスタビットから入力イベントごとに個別に有効化できます。入力イベントが特権として設定されているとき、特権アプリケーションのみが設定（該当する場合はセキュリティを含む）を変更でき、この入力イベントのマスクを変更したり、ステータスをクリアしたりできます。

  EXTI_PRIVCFGR1のセキュリティ設定は、EXTI_LOCKRでリセット後にグローバルにロックできます。

  詳細は、セクション 23：拡張割込み／イベントコントローラ（EXTI）を参照してください。

- システム設定コントローラ（SYSCFG）

  すべてのSYSCFGレジスタは、特権モードと非特権モードの両方で読み書きできますが、以下は例外です。

  – SYSCFG_SECCFGRレジスタのFPUSECビット（特権専用）

  – CPU 設定用の SYSCFG レジスタ：SYSCFG_CSLCKR、SYSCFG_FPUIMR、およびSYSCFG_CNSLCKR

  詳細は、セクション 15：システム設定コントローラ（SYSCFG）を参照してください。

3.7 セキュアな実行

ソフトウェアとハードウェアの特別な機能の組み合わせにより、デバイスは、プログラマのエラー、ネットワークアクセスを通じたソフトウェア攻撃、またはコード実行をタンパするローカルな試みに対して、機能の正常な動作を確保します。

このセクションでは、セキュアな実行のために特に設計されたハードウェアについて説明します。

3.7.1 メモリ保護ユニット（MPU）

Cortex-M33にはメモリ保護ユニット（MPU）が含まれ、これは、以下の1つ以上のパラメータに基づいて、メモリ領域（ペリフェラルにマップされた領域も含む）への読出しおよび書込みアクセスを制限できます。

- Cortex-M33動作モード（特権、非特権）
- データ/命令フェッチ

非セキュアおよびセキュアMPUのメモリマップとプログラミングは、メモリを領域（MPUごとに最大8つ）に分割します。セキュアMPUは、TrustZoneが有効なときにのみ使用できます。

3.7.2 内蔵Flashメモリ書込み保護

内蔵Flashメモリ書込み保護（WRP）は、内蔵Flashメモリユーザ領域の特別なセクションへの不正または望ましくない書込み/消去を防止します（システム領域は永続的に書込み保護されています）。

書込み保護領域は、開始アドレスと終了アドレスを書き込むオプションバイトによって定義されます。バンクごとに2つの書込み保護領域を8KBのページ単位で定義できます。

WRP領域は、オプションバイトの変更によって修正できますが、対応するFLASH_WRPxA/BRのUNLOCKオプションビットがクリアされている場合（ROMエミュレーションを意味する）を除きます。UNLOCKは、RDPのレベル1からレベル0に回帰する場合にのみセットできます。

注： **バンク消去に書込み保護領域（WRP またはHDP 領域）が含まれていると、バンク消去はアボートされます。**

3.7.3 タンパの検出と対応

原則

デバイスは、温度、電圧、および周波数攻撃に対する重要なセキュリティアセットのアクティブ保護を含んでいて、次のような機能を備えています。

- タンパ検出時のデバイスシークレットの消去
- 次のような、CPUと関連のセキュリティペリフェラルの安全な実行の保証の向上。
  - 範囲外の電圧（例：V_{BAT}、V_{DDA}）、温度、およびクロック供給（LSE）検出
  - 内部オシレータLSIによってクロックされるセキュリティウォッチドッグIWDG
  - 内部オシレータHSIをシステムクロックとして選択可能
- 電源装置の保護
  - V_{DD}またはV_{BAT}によって自動的に電力供給されるRTC/TAMPドメイン

詳細は、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照してください。

タンパ検出ソース

デバイスは8つのアクティブ入出力ピンをサポートしているため、4つの独立したアクティブタンパメッシュ、または同じ出力ピンが複数の入力ピンで共有される場合は最大7つのメッシュが可能です（合計8つのアクティブタンパI/O）。アクティブタンパ·ボールは、POS市場で使用可能なパッケージの中心にマップされます（WLCSP90など）。

アクティブ·ピンはLSEによってクロックされ、すべてのシステムの動作モード（RUN、SLEEP、STOP、STANDBY、またはSHUTDOWN）とV_{BAT}モードで機能します。

検出時間はプログラム可能であり、デジタルフィルタリングが使用可能です（連続する4つの比較サンプルで2つの比較が偽となったときにのみタンパがトリガされます）。

注： **タンパイベントが発生すると、タイムスタンプが自動的に生成されます。**

内部タンパ·ソースを、次の表にリストします。

表17.TAMPの内部タンパ

タンパ入力	TAMP_CR3のNOERビット番号	タンパソース
itamp1	0	V_{BAT}モードで機能するバックアップドメイン電圧の連続監視
itamp2	1	V_{BAT}モードで機能する温度監視
itamp3	2	V_{BAT}モードで機能するLSE監視(1)
itamp4、10	-	未使用
itamp5	4	RTCカレンダオーバーフロー（rtc_calovf）
itamp6	5	RDP > 0のときのJTAG/SWDアクセス
itamp7、12、13	6、11、12	Stop2モード以上で動作するADCアナログウォッチドッグによる電圧監視（V_{CORE}、V_{REF+}）
itamp8	7	単調カウンタオーバーフロー（内部生成）
itamp9	8	暗号化ペリフェラルのフォールト生成（SAES、PKA、AES、RNG）
itamp11	10	IWDGタイムアウトおよび潜在的タンパ（少なくとも1つの有効タンパフラグがセットされると、IWDGリセット）

1. LSE喪失またはオーバー周波数検出（> 2Mhz）、グリッチフィルタ（> 2MHz）。

タンパへの対応

デバイス内のタンパの各ソースは、以下のイベントをトリガするように設定できます。

- デバイスをStopおよびStandbyモードからウェイクアップできる割込みを生成（TAMP_CR2レジスタのTAMPxMSKビットを参照）。
- 低電力タイマのハードウェアトリガを生成。
- 対応するTAMPxNOERビットがTAMP_CR2（タンパピンの場合）またはTAMP_CR3（内部タンパの場合）でクリアされている場合、デバイスシークレットを消去。消去可能なシークレットは、次のとおりです。
  - バックアップレジスタ（x32）、SAES、AES、HASH、およびOTFDECに保存された対称鍵（Flashメモリの暗号化領域は0として読み出される）
  - PKASRAMに保存された非対称鍵、V_{DD}が存在するときに消去。
  - SRAM2およびCPU命令キャッシュメモリに保存されたその他の機密情報（SRAM2はV_{DD}が存在するときに消去される）
  - SRAM2が完全に消去されるまで、SAESのDHUKを導出するために使用される不揮発性情報は0
  - V_{DD}が存在するときに消去された2KBのバックアップSRAM（設定ビットによる）
  - V_{DD}が存在するときに消去されたICACHEおよびDCACHE1

これらすべてのシークレットに対するソフトウェアからの読出し/書込みアクセスは、TAMP_CR2のBKBLOCKビットをセットすることによってブロックできます。デバイスシークレットへのアクセスは、BKBLOCKがクリアされ、有効なタンパソースのいずれについてもタンパフラグがセットされていないときのみ可能です。

V_{DD}が存在しない場合、V_{DD}が存在するときに消去されるシークレットは、次のV_{DD}パワーオン時にのみ消去されます。

注： **デバイス機密情報の消去は、TAMP_CR2のBKERASEをセットしたり、セクション3.10.1で定義されているRDP回帰を実行したりしてもトリガされます。**

デバイス機密情報は、システムリセットやSTANDBYモードからのウェイクアップではリセットされません。

ソフトウェアフィルタリングメカニズム

各タンパ·ソースは、即時消去を開始しないように設定できます。これには、TAMP_CR2（外部タンパピンの場合）またはTAMP_CR3（内部タンパの場合）の対応するTAMPxNOERビットをセットします。

そのような状況では、タンパフラグが立てられると、すべてのタンパフラグがクリアされるまで、以下の機密情報へのアクセスはブロックされます。

- SAESのDHUK：ダミー値に固定
- バックアップレジスタ、バックアップSRAM、SRAM2：0として読出し、書込みは無視
- AES、SAES、およびHASHペリフェラル：RCCによって自動的にリセット
- PKAペリフェラル：リセットされ、メモリ使用はブロック（PKAは使用不可）

タンパイベントの通知を受けたアプリケーションが状況を分析した場合、次の2つのケースがありえます。

- アプリケーションはソフトウェアコマンドによってシークレットの消去を開始します(確認されたタンパ)。
- アプリケーションはフラグをクリアして、シークレットのブロックを解除します（偽タンパ）。

注： **タンパ·ソフトウェアがそのようなタンパフラグに反応できなかった場合、IWDGリセットにより機密情報の消去が自動的にトリガされます。**

タンパ検出と低電力モード

次の表に、タンパ検出時の低電力モードの影響の概要を示します。

表18.低消費電力モードがTAMPに与える影響

モード	説明
SLEEP	タンパ検出機能への影響なし TAMP割込みによって、デバイスはSLEEPモードを終了する。
STOP	クロックソースがLSEまたはLSIである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、タンパ検出機能への影響なし タンパイベントによって、デバイスはSTOPモードを終了する。
STANDBY	クロックソースがLSEまたはLSIである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、タンパ検出機能への影響なし タンパイベントによって、デバイスはSTANDBYモードを終了する。
SHUTDOWN	クロックソースがLSEである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、タンパ検出機能への影響なし タンパイベントによって、デバイスはSHUTDOWNモードを終了する。

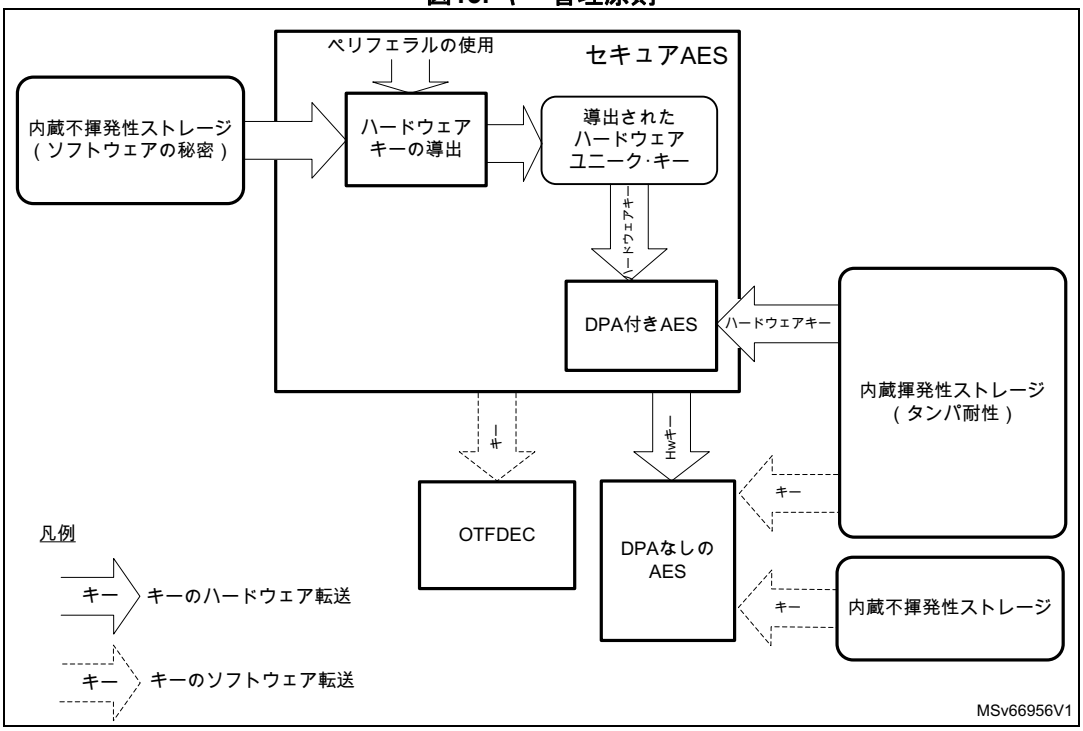
3.8 セキュアストレージ

すべてのセキュリティシステムで重要な機能は、ロングタームキーの保存、保護、およびプロビジョニングの方法です。そのようなキーは、一般に、ブートイメージのロードや重要なユーザデータの処理に使用されます。

図13 に、キー管理サービス·アプリケーションが、たとえば外部イメージ復号化キーを計算するために、どのようにAESエンジンを使用できるかを示しています。不揮発性のキーは、内蔵されたセキュアHDP領域（セクション3.6.1を参照）に保存できる一方、揮発性のキーのストレージは、バッテリ駆動のタンパ保護SRAMまたはTrustZoneを認識するTAMPのレジスタで構成されます。

図13 は、ソフトウェアによって操作されるキー（OTFDECキーなど）や、ハードウェアによってのみ管理されるキー（DHUKなど）も示しています。これらのハードウェアキーの詳細については、セクション3.8.1を参照してください。

図13. キー管理原則



タンパ保護の詳細についてはセクション3.7.3を参照してください。TAMP TrustZone機能についてはセクション3.5.5で簡単に説明しています。

3.8.1 ハードウェア秘密キー管理

前の図に示されているように、デバイスはハードウェア秘密キーを使用した、アプリケーションキーの保護の向上を提案します。これらのAESキーは、クリアテキスト（暗号化されていない）で公開しなくても、アプリケーションにとって使用可能にできます。タンパが発生した場合は、そのようなキーを直ちに使用不可にすることもできます。、

ハードウェア秘密キーには3つの異なるソースがあります。

- DHUK：Flash メモリ内の 256bit の不揮発性、デバイス一意シークレットに基づいて導出されたキー。このキーの生成は、TrustZoneの状態とキーの使用状態（KMOD）を考慮しています。

  注：DHUKは、RDP = 0（デバッグ/開発モード）の場合、すべてのデバイスに対して同じです。
- BHK：TAMPの耐タンパ揮発性ストレージに保存された256bitのアプリケーションキー

  このキーは、ブート時に書き込まれ、その後は次のリセットまで、アプリケーションに対して読み書きがロックされます。
- XORK：BHKとDHUKのXORの結果

これらのキーの用途は以下のとおりです。

- 通常のキーとして、書込み専用キーレジスタにロードして使用（ソフトウェア･キー･モード）
- 別のキーの暗号化/復号化キーとして、DPA抵抗SAESで使用（ラップ･キー･モード）
- 別のキーの暗号化/復号化キーとして、高速なAESエンジンで使用（共有鍵モード）

3.8.2 ユニークID

デバイスは、各デバイスに一意な96bitのIDを保存します（セクション 76.1：ユニークデバイスIDレジスタ（96bit）を参照）。

アプリケーションサービスは、この一意IDキーを使用して、クラウドネットワーク製品を識別したり、偽造デバイスまたはクローンが、信頼できないデータをネットワークに注入するのを困難にしたりできます。

または、256bitのデバイス一意キー（DHUK）を使用することもできます（セクション3.8.1を参照）。

3.9 暗号化エンジン

デバイスは最新の暗号アルゴリズムを実装しています。これらは、米国のNIST、ドイツのBSI、フランスのANSSIなど、各国の安全保障機関によって推奨されているキー長とコンピューティング保護が特徴です。これらのアルゴリズムは、プライバシー、認証、整合性、エントロピー、および身元確認のサポートに使用されます。

STM32に内蔵された暗号化エンジンは、重要な暗号化機能の実装における弱点を減らし、たとえば弱い暗号アルゴリズムやキー･サイズの使用を防止します。また、暗号操作を実行する際の処理時間の短縮と電力消費の低減も可能にし、これらの計算負荷をCortex-M33から取り除きます。特に、非対称暗号の場合はそうです。

製品認証の目的から、STは、これらのセキュリティ機能の実装方法と検証方法に関する認証済みのデバイス情報を提供できます。

暗号化エンジンの処理時間の詳細については、リファレンスマニュアルの該当セクションを参照してください。

3.9.1 暗号化エンジンの特徴

表19に、デバイスで使用可能な高速化暗号操作を示します。2つのAESアクセラレータが使用可能です（どちらもセキュアアプリケーション専用に予約できます）。

注： **付加的な操作は、ファームウェアを使用して追加できます。**

PKAは、非対称暗号操作を加速できます（キーペア生成、ECCスカラー乗算、曲線上の点の確認など）。詳細は、セクション 53：公開鍵アクセラレータ（PKA）を参照してください。

表19.加速された暗号操作

動作	アルゴリズム	仕様	キー長（ビット単位）	モード
エントロピーの取得	RNG	NIST SP800-90B(1)	N/A	並列実行するソフトウェアおよびハードウェア(2) モード
暗号化、復号	AES	FIPS PUB 197 NIST SP800-38A	128、256	ECB、CBC、CTR(3)
認証済みの暗号化または復号		NIST SP800-38C NIST SP800-38D	128、256	GCM、CCM
暗号ベースのメッセージ認証コード		NIST SP800-38D	128、256	GMAC
チェックサム	MD5	IETF RFC 1321	N/A	ダイジェスト128bit
	SHA-1	FIPS PUB 180-4		ダイジェスト160bit
暗号化ハッシュ	SHA-2			SHA-224,SHA-256
メッセージ認証のための鍵付きハッシング	HMAC	FIPS PUB 198-1 IETF RFC 2104	ショート、ロング（> 64バイト）	-
暗号化/復号化キーペア生成(4)	RSA	IETF RFC 8017 NIST SP800-56B	最大4160	RSAES-OAEP
ハッシング付き署名 (4) 署名の検証	RSA	IETF RFC 8017 FIPS PUB 186-4	最大4160	PKCS1-v1_5、PSS
	ECDSA	ANSI X9.62 IETF RFC 7027 FIPS PUB 186-4	最大640	詳細については、セクション 53：公開鍵アクセラレータ（PKA）の「ECC操作でサポートされる曲線群」の表を参照。
鍵の合意	ECDH	ANSI X9.42		

1. STMicroelectronicsレビュードキュメントを使用して認定可能
2. 専用ハードウェアバスを使用して、SAESおよびPKAに乱数を分散
3. ECBおよびCBC連鎖モードをサイドチャネルやSAESのタイミング攻撃から保護（セクション3.9.2を参照）。
4. 私有鍵暗号をサイドチャネルおよびタイミング攻撃から保護。

注： **バイナリ曲線、エドワーズ曲線、およびCurve25519は、PKAではサポートされていません。**

3.9.2 セキュアAESコプロセッサ（SAES）

デバイスは、電力および電磁サイドチャネル攻撃に対する対策と軽減策を実装する、追加のオンチップ·ハードウェアAESの暗号化および復号化エンジンを備えています。

また、専用の48Mhz SHSIクロックにより、SAESはクラス最高のサイドチャネル攻撃からの保護機能を提供するために、AESより遅くなります。SAESエンジンは、電子コードブック（ECB）または暗号ブロック連鎖（CBC）モードで128bitまたは256bitのキーをサポートしています。

セクション3.8に示されているように、SAESは機密情報の超セキュアなオンチップストレージに使用できます。セキュア専用にすることもできます。

詳細については、セクション 50：セキュアAESコプロセッサ（SAES）を参照してください。

3.9.3 オンザフライ復号エンジン（OTFDEC）

OTFDECのTrustZoneを認識するペリフェラルは、OCTOSPI経由で接続された外部Flashメモリに保存された暗号化済イメージのオンザフライ復号を提示します。この復号化処理により、標準NOR Flashメモリが使用されたときに、追加のサイクルのオーバーヘッドはほとんど導入されません。OTFDECは、デバイス上のFlashメモリイメージを暗号化するためにも使用できます（デバイス一意秘密キーによる暗号化など）。

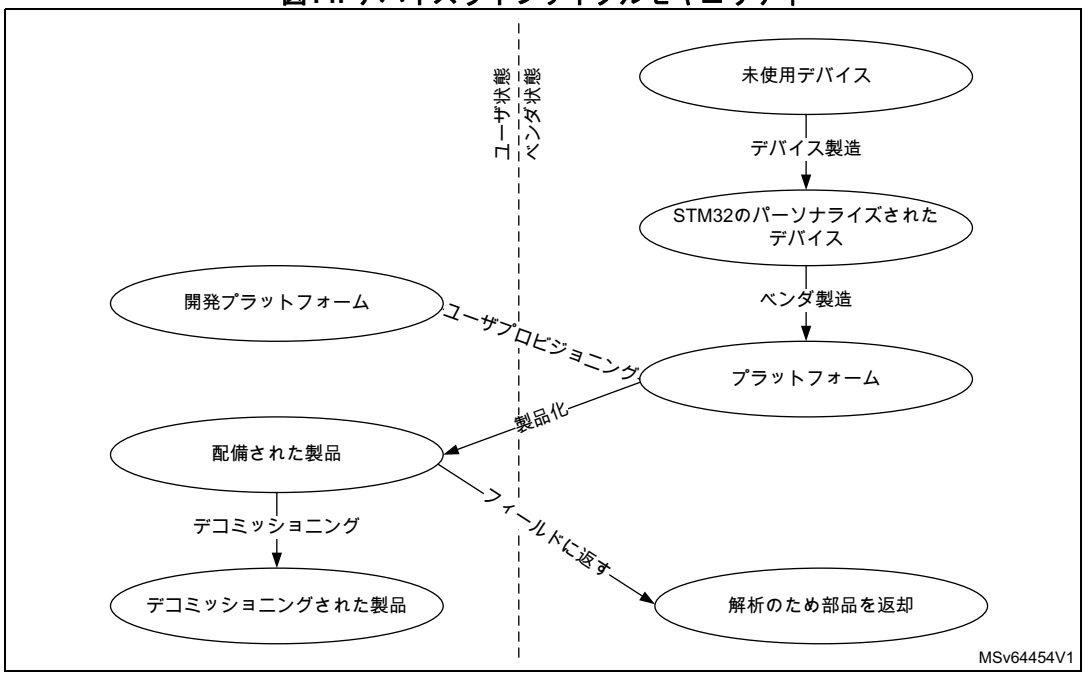
TAMPでタンパイベントが確定されると、すべてのOTFDECキーは消去され、OTFDECが再び正しく初期化されるまで、暗号化領域は0として読み出されます。

OTFDECの一般的な用途については、セクション3.12.2を参照してください。ペリフェラルプログラミングの詳細については、セクション 52：オンザフライ復号化エンジン（OTFDEC）を参照してください。

3.10 製品ライフサイクル

次の表に、標準的なIoTデバイスのライフサイクルの概要を示します。各ステップについて、デバイスはハードウェアに内蔵されたセキュアなライフサイクル管理メカニズムを提示します。

図14. デバイスライフサイクルセキュリティ



次の表に、ベンダまたはエンドユーザ構内で見られるさまざまなフェーズと関連する遷移の詳しい概要を示します。

表20.主な製品ライフサイクル遷移

遷移	説明
デバイス製造	STMicroelectronicsは新しいSTM32デバイスを作成し、常に製造上の欠陥がないか確認します。このプロセス中、STM32にはROMファームウェア、セキュア·ファームウェア·インストール（SFI）の一意のキーペア、および公開IDがプロビジョニングされます。
ベンダ製造	1つ（または複数）のベンダが、エンドユーザに提供される前のプラットフォームのアセンブリ、初期化、およびプロビジョニングを担当します。このエンド·ユーザは、ソフトウェア開発に最終製品を使用するか(「製品化」遷移)、プラットフォームを使用できます(「ユーザプロビジョニング」遷移)。
量産	エンドユーザは、すぐに使える製品を入手します。プラットフォームのすべてのセキュリティ機能は有効化され、デバッグ/テスト機能は制限/無効化され、不変コードへの一意なブートエントリが行われます。
ユーザプロビジョニング	プラットフォームベンダは、生産クラウドネットワークに接続されない、開発用の個別プラットフォームを準備します。
フィールドに返す、またはデコミッショニング	これらは単方向遷移で、デバイスはユーザ構内に保管されるかメーカーに返されます。いずれの場合も、ユーザデータを含むすべてのデータが破壊され、それによりデバイスは安全に動作する能力を失います（管理されたIoTネットワークへの接続など）。

以下に説明する機能は、デバイスライフサイクルのセキュリティ保護に貢献します。

3.10.1 読出し保護（RDP）によるライフサイクル管理

読出し保護メカニズム（正式なハードウェアの機能）は、次の表に要約されるように、デバイスのデバッグ、テスト、およびプロビジョニングされた機密情報へのアクセスを制御します。

表21.一般的な製品ライフサイクルの フェーズ

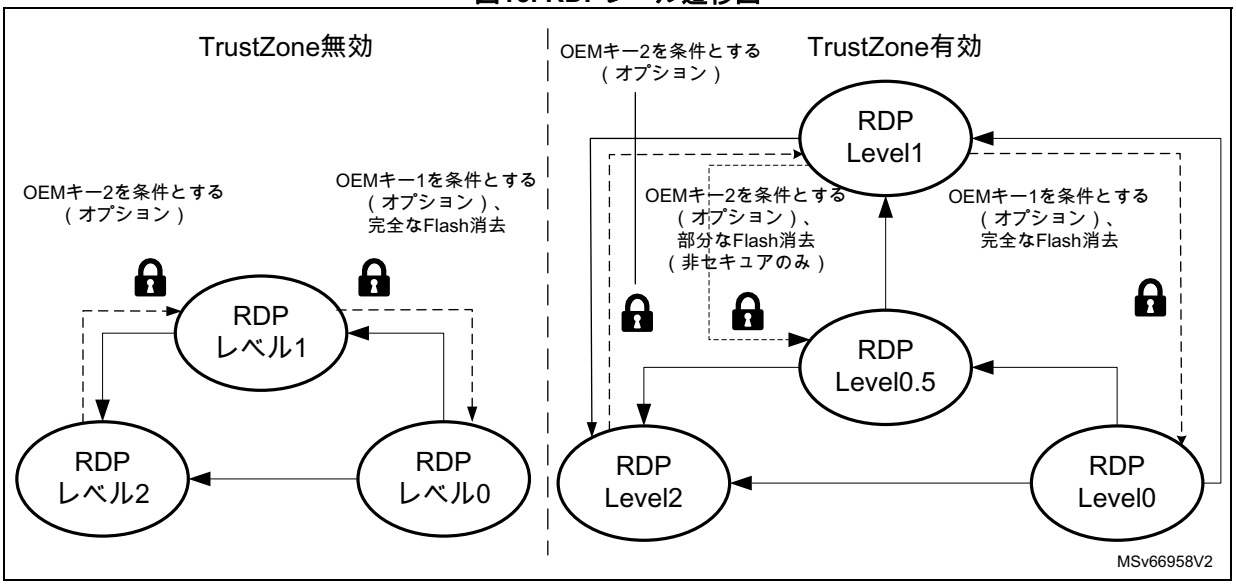
RDP保護レベル		デバッグ	コメント
レベル0	デバイスオープン	セキュア(1) および 非セキュア	TrustZoneが有効なとき、ブートアドレスはセキュア領域を対象とする必要があります（システムFlashメモリのセキュアSRAM、セキュアFlashメモリ、RSS）。 OEM1およびOEM2のアンロック·キーは、両方ともFlashメモリのユーザオプションでプロビジョニングできます。SAESペリフェラルのDHUKは、すべてのデバイスに対して同じです。
レベル0.5(2)	部分的なデバイスクローズ （クローズセキュア）	非セキュアのみ	TrustZoneが有効なとき、ブートアドレスはセキュア領域を対象とする必要があります（セキュア·ユーザまたはシステムのFlashメモリ）。SRAMでのブートは許されません。 デバッグに接続している場合、非セキュアFlashメモリにアクセスできます。 OEM1およびOEM2のアンロック·キーは、両方ともFlashメモリのユーザオプションでプロビジョニングできます。DHUKはデバイスごとに異なります。
レベル1	デバイスメモリ保護	非セキュアのみ （条件付き）	ブートアドレスは、セキュアユーザFlashメモリを対象とする必要があります。 デバッグに接続している場合、非セキュアFlashメモリ、暗号化されたFlashメモリ(3)、SRAM2、およびバックアップレジスタへはアクセス**できません**。 OEM1およびOEM2のアンロック·キーは、両方ともFlashメモリのユーザオプションでプロビジョニングできます。DHUKはデバイスごとに異なります。
レベル2	デバイスクローズ	なし （JTAGヒューズ）	ブートアドレスは、ユーザFlashメモリを対象とする必要があります（TZEN = 1の場合セキュア）。 オプションバイトは読出し専用です。そのため、OEM2のアンロック·キーが有効になるまで、RDPレベル2は変更できません（表22 を参照）。DHUKはデバイスごとに異なります。

1. RSSコードを実行しているときには、デバッグはできません。
2. 製品のTrustZoneセキュリティが有効な場合のみ適用可能です。
3. 外部Flashメモリ領域は、OTFDECによってオンザフライで復号されます。

注： OEM1KEYオプションバイトは、OEM1LOCK = 0（RDP = 0.5または1のみ）の場合に変更できます。
OEM2KEYオプションバイトは、OEM2LOCK = 0（RDP = 1のみ）の場合に変更できます。

デバッグインタフェースまたはシステムブートローダを介して、次の図に要約されたサポートされている遷移を要求できます（使用可能な場合）。

図15. RDPレベル遷移図



次の図に示すように、RDP1からのRDP回帰中に、ユーザFlashメモリの一部または全体が自動的に消去されます。これらの回帰は、OEMによってプロビジョニングされている場合は、専用の64bitのパスワード･キーに合わせて調整できます（次のサブセクションを参照）。RDPレベル1からRDPレベル0.5までの回帰中、非セキュア内蔵Flashメモリのみが消去され、セキュア･ブートやセキュアファームウェア更新などの機能は保持されます。レベル1からのすべての回帰において、FlashメモリのOTP領域は保持され、すべてのSRAMSと対象デバイスの機密情報は消去されます。そのため、OTPに機密情報を保存しないでください。RDPレベル0への回帰後に明かされます。タンパへの対応として消去されるこれらの機密情報は、セクション3.7.3に定義されています。

注： **オプションバイトTZENを使用したTrustZoneの有効化は、RDPレベルが0の場合のみ可能です。**

RDPの詳細については、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。

RDPアンロックシーケンス

最後の図で説明した2つのOEMパスワード･キーの使用法の詳細について、以下に示します。

注： **デバイスは、レベル1への永続的なRDPレベル2（レガシーモード）またはパスワードベースのRDPレベル2回帰の両方をサポートしています。このレベル2回帰によってアプリケーションコードは消去されず、またその場でRDPレベル1保護は変更されません。**

パスワードベースの回帰の詳細については、次の表で確認できます。

表22.OEM1/2RDPアンロック方法

OEM1パスワード･オプション			OEM2パスワード･オプション		
OEM1 LOCK	初期RDP レベル	RDP回帰	OEM2 LOCK	初期RDP レベル	RDP回帰
1	1	レベル 0 への回帰は、OEM1アンロック･シーケンスによってのみ可能（下記を参照）	1	1	レベル0.5への回帰は、OEM2.1アンロック･シーケンスによってのみ可能（下記を参照）
				2	レベル1への自動回帰は、正常なOEM2.2アンロック･シーケンスによってのみトリガされる（下記を参照）
0		レベル0への回帰を常に許可する	0	1	レベル0.5への回帰を常に許可する
				2	レベル1への回帰を常に許可しない RDPは永続的な状態を維持する。

- RDPレベル1から開始するOEM1アンロック･シーケンス：
  - JTAG/SWDを通じてパスワード･キーをシフトします（次の注を参照）。
  - このキーがデバイス内でプロビジョニングされたOEM1KEYと一致する場合、アプリケーションはレベル0への回帰シーケンスをトリガできます。回帰完了後、内蔵Flashメモリとデバイスの機密情報がすべて消去されます。OTP領域は消去されません。
  - 不一致の値がある場合、RDP回帰はブロックされ、次のパワーオンリセットまでRDPレベル1保護が強制されます。
- RDPレベル1から開始するOEM2.1アンロック･シーケンス：
  - リセット中に、JTAG/SWDを通じてパスワード･キーをシフトします（次の注を参照）。
  - このキーがデバイス内でプロビジョニングされたOEM2KEYと一致する場合、アプリケーションはレベル0.5への回帰シーケンスをトリガできます。回帰完了後、非セキュア内蔵Flashメモリとデバイスの機密情報が消去されます。OTP領域は消去されません。
  - 不一致の値がある場合、RDP回帰はブロックされ、次のパワーオンリセットまでRDPレベル1保護が強制されます。
- RDPレベル2から開始するOEM2.2アンロック･シーケンス：
  - リセット中に、JTAG/SWDを通じてパスワード･キーをシフトします（次の注を参照）。
  - このキーがデバイス内でプロビジョニングされたOEM2KEYと一致する場合、デバイスは自動的にレベル1への回帰シーケンスをトリガできます。回帰完了後、ユーザはパワーオンリセットを行う必要があります。
  - 不一致の値がある場合、RDP回帰はブロックされ、次のパワーオンリセットまでRDPレベル2保護が強制されます。

注： **パスワードでのデバイスのアンロックは、パワーサイクルごとに1回だけ行うことができます。**

JTAG/SWDを通じたパスワード･キーのシフトは、2つの32bitのキーのワード書込みに対応します。DBGMCU_DBG_AUTH_HOSTレジスタのAUTH_KEY[31:0]とAUTH_KEY[63:32]です。

JTAG 32bitのデバイス固有ID

JTAGポートが無効（OEM2LOCK = 0およびRDPレベル = 2）でない限り、32bitのデバイス固有の量は、いつでもJTAGポートを通じて読出し可能です。この情報はDBGMCU_DBG_AUTH_DEVICEに保存されます。

OEMは、この32bit情報を使用して、この特定のデバイスをアンロックするために、予期されたOEMパスワード･キーを導出できます。

3.10.2 推奨されるオプションバイトの設定

ほとんどの場合、ユーザの脅威モデルは主にソフトウェア攻撃に焦点を置いています。この場合、RDPレベル1をデバイス保護として保持すれば十分です。

より攻撃的な脅威モデルの場合で、ユーザがSTM32デバイスへの物理攻撃を恐れている場合は、RDPレベル2を設定してセキュリティのレベルを最適化することを推奨します。

推奨設定について以下に詳述します。

- TrustZoneが無効な場合（TZEN = 0）
  - RDPレベル2
  - ユーザFlashメモリにセットされた非セキュアブートアドレスのオプションバイト
- TrustZoneが有効な場合（TZEN = 1）
  - RDPレベル2
  - BOOT_LOCK = 1
  - ユーザセキュアFlashメモリにセットされたセキュアブートアドレスのオプションバイト

前のセクションで説明されているように、顧客は、OEM Key2が正常にプロビジョニングされ、OEM2LOCKオプションビットがセットされている場合に限り、任意のRDPレベル2部分をRDPレベル1に回帰させることができます。

3.11 アクセスコントロールされたデバッグ

デバイスは、無許可のデバッグおよびトレース機能の使用から顧客のアセットの機密性を保証するために、内蔵デバッグ機能へのアクセスを制限します。

3.11.1 読出し保護（RDP）対応のデバッグ保護

セクション3.10.1で説明されているように、ハードウェアのRDPメカニズムは、デバイスのデバッグおよびテストへのアクセスを自動的に制御します。これらのデバッグ機能の保護は、次の表に定義されています。使用可能なパスワードベースの回帰については、セクション3.10.1で説明します。

表23.RDPによるデバッグ保護

RDP保護レベル		デバッグ機能保護
レベル0	デバイスオープン	任意のデバッグ(1)
レベル0.5(2)	部分的なデバイスクローズ	セキュアデバッグは使用できなくなりました。
レベル1	デバイスメモリ保護	非セキュアデバッグは、内蔵Flashメモリ、暗号化された外部Flashメモリ(3)、SRAM2、およびバックアップレジスタに保存されたコードやデータをデバッグできなくなりました。
レベル2	デバイスクローズ	JTAGは、パスワード・キー・インジェクション（OEM2LOCK = 1）のためだけに動作可能にしている場合を除き、物理的に無効になります。詳細については、セクション3.10.1を参照してください。

1. フィールドに返すために使用されるSTのエンジニアリング・テスト・モードを含む。
2. 製品のTrustZoneセキュリティが有効な場合のみ適用可能です。
3. 外部Flashメモリ領域は、OTFDECによってオンザフライで復号されます。

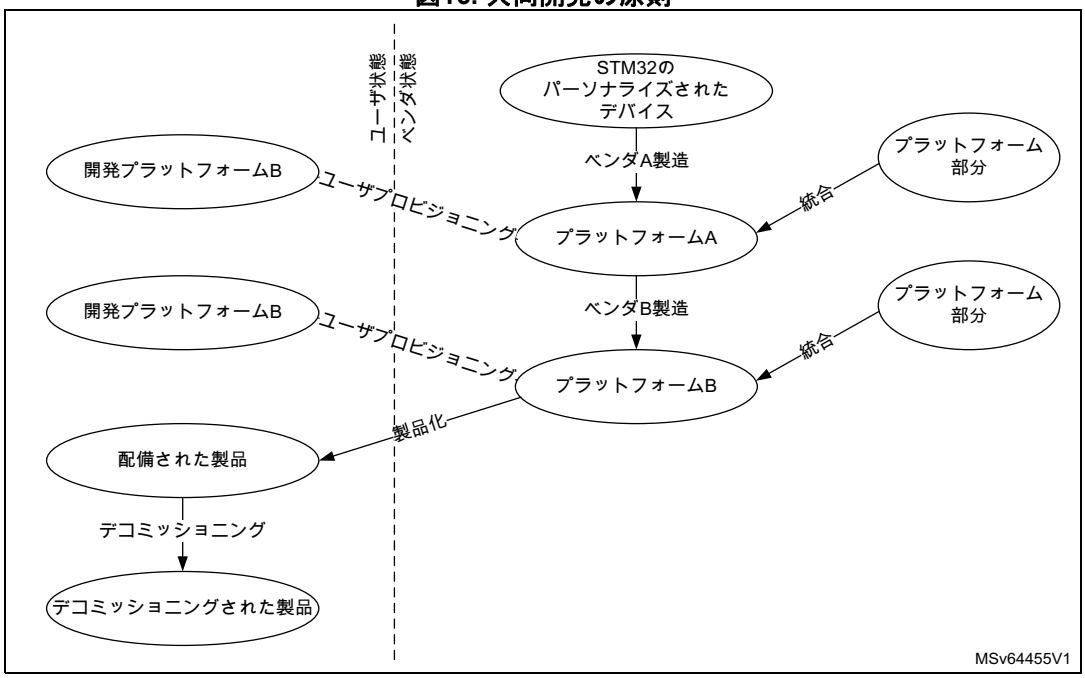
3.12 ソフトウェア知的財産保護と共同開発

ソフトウェア知的財産保護と共同モデルにより、デバイスは革新的なサードパーティライブラリを統合するソリューションの設計を可能にします。

次の表に、共同開発の概要を示します。STMicroelectronicsによって販売されたパーソナライズされたデバイスから始めて、ベンダAはハードウェアとソフトウェアの一部をプラットフォームAで統合できます。それをベンダBが使用して、同じことを行ってから、最終製品をエンドユーザにデプロイします。

注： **各プラットフォームベンダは、生産クラウドネットワークに接続されない開発用の個別プラットフォームをプロビジョニングできます（「開発プラットフォームX」）。**

図16. 共同開発の原則



以下で説明されている機能は、そのような共同開発におけるソフトウェア知的財産のセキュリティ保護に貢献します。

3.12.1 RDPによるソフトウェア知的財産保護

セクション3.10.1で説明されているように、ハードウェアのRDPメカニズムは、デバイスでプロビジョニングされた機密情報へのアクセスを自動的に制御します。

これらの機密情報の保護は、次の表に定義されています。

表24.RDPによるソフトウェア知的財産保護

RDP保護レベル		シークレット保護
レベル0	デバイスオープン	特別な保護なし。
レベル0.5(1)	部分的なデバイスクローズ	セキュア·ブート時にセキュアとしてマップされたすべてのペリフェラルとメモリを、ダンプ、デバッグ、またはトレースできません。
レベル1	デバイスメモリ保護	内蔵Flashメモリ、暗号化された外部Flashメモリ(2)、SRAM2、およびバックアップレジスタに保存されたデータやコードに、デバッガからアクセスできなくなりました。
レベル2	デバイスクローズ	デバイスに保存され、または外部Flashメモリに暗号化されたすべてのデータとコードは、クリアテキストのダンプ、デバッグ、またはトレースができません。

1. 製品のTrustZoneセキュリティが有効な場合のみ適用可能です。
2. 外部Flashメモリ領域は、OTFDECペリフェラルによってオンザフライで復号されます。

3.12.2 OTFDECによるソフトウェア知的財産保護

セクション3.9.3で説明されているように、OCTOSPIに関連付けられたOTFDECは、外部SPI Flashメモリに保存された暗号化済みイメージをオンザフライで復号できます。

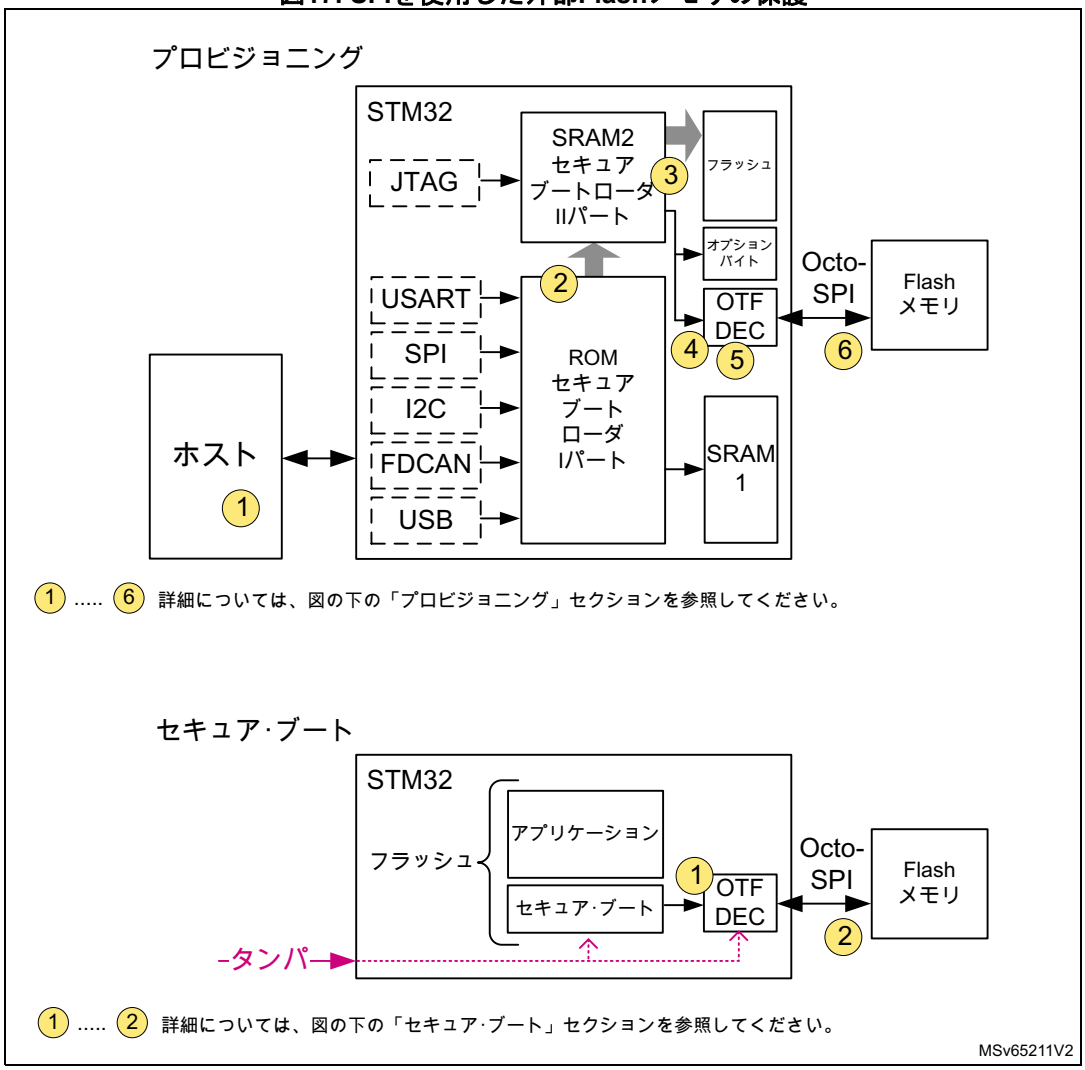
この機能により、デバイスは次のいずれかの方法で知的財産のインストールができます。

- 無線で。デバイスにプロビジョニングされたキーですでに暗号化されたイメージで。
- 信頼できる、または信頼できない環境/設備にあるプロビジョニングホストを通じて。

図17 は、この最後のケースを示しています。信頼できない環境での、内蔵Flashメモリと外部 SPI Flashメモリ（暗号化されている）の両方のソフトウェア知的財産のプロビジョニング。

注： **OTFDECはカウンタモード（CTR）のAESを使用して、可能な限り低い遅延を達成するため、1つの暗号化領域の内容が変更されるたびに、対応する暗号コンテキスト（キーまたは初期化ベクトル）を変更する必要があります。この制約により、OTFDECは、外部NOR Flashメモリに保存された読出し専用データまたはコードの復号に適しています。**

図17. SFIを使用した外部Flashメモリの保護



プロビジョニング

デバイスが新品であると仮定して、最初のステップは、次のように両方のメモリをプロビジョニングすることです。

1. ユーザは、以下で構成されるSFIイメージを作成します。
   - 暗号化された内部ファームウェアおよびデータ（外部Flashメモリのドライバを含む）
   - 暗号化された外部ファームウェアおよびデータAESキー（最大4）
   - 暗号化された外部ファームウェアおよびデータイメージ
2. システムメモリに保存されたセキュア·ブートローダは、サポートされる通信ポート（USART、SPI、I²C、FDCAN、USB、およびJTAG）を通じて、セキュア·ブートローダの2番目の部分をSRAMにロードします。この2番目の部分はセキュアSRAMで実行され、SFIプロセスの実行を担当し、上記のサポートされる通信ポートを通じて受信したコマンドによってSFIプロトコルを適用します。

3. 内部 Flash メモリは、復号されたオプションバイト、内部ファームウェアおよびデータ、外部ファームウェアおよびデータAESキーでプログラムされます。または、そのようなグローバルキーの代わりに、デバイス一意外部ファームウェアAESキーを使用することもできます。
4. OTFDECは、対応する外部ファームウェアおよびデータ AES キーを含め、暗号化された領域情報で適切に初期化されます。
5. SFI プロセスの実行により、暗号化された外部ファームウェアおよびデータイメージのチャンクは、デバイスで復号されてから、OTFDECで再び暗号化されます。
6. OTFDECによるチャンクの再暗号化後、ユーザ外部Flashメモリのプログラマは、OCTOSPIを通じて最後の暗号化チャンクを外部SPI Flashメモリにプログラムします。

または、OTFDECの外部ファームウェアおよびデータAESキーは、署名ツールで生成されない場合、デバイスで生成できます。

セキュア·ブート

プロビジョニング後、デバイスが信頼できるファームウェアで初期化するたびに、以下のアクションが必要です。

1. セキュアブートファームウェアが実行して、外部ファームウェアおよびデータAESキーを、その他の必要な情報とともに、OTFDEC書込み専用キーレジスタにプログラムします。
2. アプリケーションは、暗号化された外部ファームウェアおよびデータをOCTOSPIを通じてメモリマップモードで読み出し、または実行しますが、タンパイベントが検出された場合を除きます。この場合、すべてのOTFDECキーは消去され、OTFDECが再び正しく初期化されるまで、暗号化領域は0として読み出されます。

デバイスのSFIソリューションの詳細については、アプリケーション·ノート「STM32 MCUs secure firmware install (SFI) overview」（AN4992）を参照してください。

3.12.3 その他のソフトウェア知的財産保護

ソフトウェア知的財産に対するデバイスのその他の保護としては、以下のものがあります。

- 物理的タンパまたは摂動などの侵入攻撃は、検出後のデバイスのデコミッショニングによって、検出された攻撃が成功する前に対応されます。
- サイドチャネル攻撃などの非侵襲攻撃には、タイミング、電力、EM放射などのサイドチャネル経由で、機密情報を漏えいさせないことで対抗します。

4 ブートモード

起動時に、BOOT0ピン、NBOOT0、およびNSBOOTADDx[24:0]/SECBOOTADD0[24:0]オプションバイトを使用して、以下を含むブートメモリアドレスを選択します。

- ユーザFlashメモリ内の任意のアドレスからのブート
- システムメモリ（ブートローダ）からのブート
- 内蔵SRAM内の任意のアドレスからのブート
- ルートセキュリティサービス（RSS）からのブート

BOOT0の値は、PH3-BOOT0ピンまたは、必要に応じてGPIOパッドを解放するために、ユーザオプションビットの値から来る場合があります。

システムメモリに配置されたブートローダは、USART、I2C、SPI、FDCAN、またはUSB/OTG_FS/OTG_HSのデバイスモードでDFU（デバイスファームウェアアップグレード）を使用して、Flashメモリを再プログラムするために使用します。

表25および表26には、TrustZoneが無効または有効な場合のブートモードについて詳しく示されています。

表25.TrustZoneが無効な場合のブートモード（TZEN = 0）

NBOOT0 FLASH_ OPTR[27]	BOOT0 ピン PH3	NSWBOOT0 FLASH_ OPTR[26]	ブート･アドレスの オプション･バイトの選択	ブート領域	STによってプログラムされた デフォルト値
-	0	1	NSBOOTADD0 [24:0]	ユーザオプションバイト NSBOOTADD0[24:0]で定義されたブートアドレス	Flashメモリ：0x0800 0000
-	1	1	NSBOOTADD1 [24:0]	ユーザオプションバイト NSBOOTADD1[24:0]で定義されたブートアドレス	ブートローダ：0x0BF9 0000
1	-	0	NSBOOTADD0 [24:0]	ユーザオプションバイト NSBOOTADD0[24:0]で定義されたブートアドレス	Flashメモリ：0x0800 0000
0	-	0	NSBOOTADD1 [24:0]	ユーザオプションバイト NSBOOTADD1[24:0]で定義されたブートアドレス	ブートローダ：0x0BF9 0000

TrustZoneがTZENオプションビットをセットすることによって有効化される場合、ブート領域はセキュア領域になければなりません。SECBOOTADD0[24:0]オプション･バイトは、セキュア･ブート･メモリ･アドレスを選択するために使用します。

一意なブートエントリオプションは、BOOT_LOCKオプションビットを設定することによって選択できます。その他すべてのブートオプションは無視されます。

表26.TrustZoneが有効な場合のブートモード（TZEN = 1）

BOOT_LOCK	NBOOT0 FLASH_OPTR[27]	BOOT0 ピン PH3	NSWBOOT0 FLASH_OPTR[26]	RSS コマンド	ブート･アドレスのオプション･バイトの選択	ブート領域	STによってプログラムされたデフォルト値
0	-	0	1	0	SECBOOTADD0 [24:0]	ユーザオプションバイト SECBOOTADD0[24:0] で定義されたセキュアブートアドレス	Flashメモリ：0x0C00 0000
	-	1	1	0	N/A	RSS	RSS：0x0FF8 0000
	1	-	0	0	SECBOOTADD0 [24:0]	ユーザオプションバイト SECBOOTADD0[24:0] で定義されたセキュアブートアドレス	Flashメモリ：0x0C00 0000
	0	-	0	0	N/A	RSS	RSS：0x0FF8 0000
	-	-	-	≠ 0	N/A	RSS	RSS：0x0FF8 0000
1	-	-	-	-	SECBOOTADD0 [24:0]	ユーザオプションバイト SECBOOTADD0[24:0] で定義されたセキュアブートアドレス	Flashメモリ：0x0C00 0000

ブートアドレスのオプションバイトは、任意のブートメモリアドレスをプログラムするために使用されます。ただし、許容されるアドレス空間はFlashメモリ読出し保護（RDP）のレベルに依存します。

RDPレベルが0.5以上のときにプログラムされたブートメモリアドレスが許容されるメモリマップ領域外の場合、デフォルトのブートフェッチアドレスは、次の表で説明されているように、TrustZoneセキュリティオプションに応じて、強制的にセキュアまたは非セキュアFlashメモリに設定されます。

表27.RDP保護レベルに対するブート領域

RDP	TZEN = 1	TZEN = 0
0	任意のブートアドレス	任意のブートアドレス
0.5	RSS専用のブートアドレス：0x0FF80000またはセキュアFlashメモリ： – STM32U535/545では0x0C00 0000-0x0C07 FFFF – STM32U575/585では0x0C00 0000-0x0C1F FFFF – STM32U59x/5Ax/5Fx/5Gxでは0x0C00 0000-0x0C3F FFFF これ以外の場合、強制されるブートアドレスは0x0FF8 0000です。	N/A
1		任意のブートアドレス
2		Flashメモリ専用のブートアドレス： – STM32U535/545では0x0800 0000-0x0807 FFFF – STM32U575/585では0x0800 0000-0x081F FFFF – STM32U59x/5Ax/5Fx/5Gxでは0x0800 0000-0x083F FFFF これ以外の場合、強制されるブートアドレスは0x0800 0000です。

BOOT0の値（ピンまたはオプションビットから来る）は、リセット解除と同時にラッチされます。要求されるブートモードを選択するためにnBOOT0を設定するか、BOOT0の値を設定するかは、ユーザ次第です。

BOOT0ピンまたはユーザオプションビット（Flash_OPTRのNSWBOOT0に依存）は、STANDBYモードを終了するときにも再度サンプリングされます。結果として、BOOT0ピンまたはユーザオプションビットは、STANDBYモードで必要とされるブートモード設定を保持する必要があります。起動遅延時間後、プロセッサのリセットが解除される前に、ブート領域の選択が行われます。

PH3/BOOT0 GPIOは次のように設定されます。

- FLASH_OPTRのNSWBOOT0がセットされているときは、リセット完了フェーズ中は入力モードに設定され、リセット解除後に自動的にアナログモードに切り替わります（BOOT0ピン）。
- FLASH_OPTRのNSWBOOT0がクリアされているときは（BOOT0値はオプションビットからの値）、リセットフェーズからオプションバイトのロード完了までの間入力モードで、リセットフェーズが未完でも自動的にアナログモードに切り替わります。

内蔵ブートローダ

内蔵ブートローダは、システムメモリに配置され、生産時にSTによってプログラムされています。アプリケーションノート「STM32 microcontroller system memory boot mode」（AN2606）を参照してください。

内蔵のルート･セキュリティ･サービス（RSS）

内蔵RSSは、セキュア情報ブロックに配置されるよう生産時にSTによってプログラムされています。アプリケーションノート「STM32 MCUs secure firmware install (SFI) overview」（AN4992）を参照してください。

5 グローバルTrustZoneコントローラ（GTZC）

5.1 GTZCの概要

このセクションでは、次のようなサブブロックを含むグローバルTrustZoneコントローラ（GTZC）ブロックについて説明します。

- **TZSC**：TrustZoneのセキュリティコントローラ

  このサブブロックは、スレーブペリフェラルのセキュア/特権状態を定義します。また、ウォーターマークメモリペリフェラルコントローラ（MPCWM）のサブ領域の領域サイズと属性を制御します。TZSCは、セキュリティ保護可能な各ペリフェラルのセキュア状態をRCCおよびI/Oロジックと共有することで、いくつかのペリフェラル（RCC、GPIOなど）に通知します。
- **MPCBB**：ブロックベースのメモリ保護コントローラ

  このサブブロックでは、プログラム可能なセキュリティおよび特権属性を備えたセグメント化されたSRAM（512バイトのページ）を持つTrustZoneシステム製品の内蔵RAMを設定します。
- **TZIC**：TrustZone不正アクセスコントローラ

  このサブブロックは、システム内のすべての不正アクセスイベントを集め、NVICへのセキュア割込みを生成します。

これらのサブブロックは、以下に挙げるプログラム可能なセキュリティおよび特権属性を備えたバスエージェントを持つ製品において、TrustZoneシステムセキュリティを設定するために使用します。

- プログラム可能なセキュア/特権ブロック（ページ）を持つオンチップRAM（内蔵RAM）
- プログラム可能なセキュア/特権アクセスモードを持つAHBおよびAPBペリフェラル
- セキュア/特権領域を持つオフチップメモリ（外付けメモリ）

5.2 GTZCの主な機能

GTZCの主な機能を以下に列記します。

- TZSC、TZIC、およびMPCBB用の3つの独立した32bit AHBインタフェース
- セキュアなトランザクションでのみTZICへのアクセスが可能
- TZSCおよびMPCBBの特権および非特権部分への、セキュアおよび非セキュアアクセスに対応
- 一連のレジスタを使用して製品のセキュリティ設定を定義
  - 内蔵SRAM用のセキュア/特権ブロック
  - 外部メモリおよび内部バックアップSRAM用のセキュア/特権領域
  - セキュリティ保護可能なペリフェラルのセキュア/特権アクセスモード
  - セキュリティ保護可能なマスタのセキュア/特権アクセスモード

GTZCのTrustZoneシステムアーキテクチャ

Armv8-Mは、次の2つのワールドを分離することで、TrustZone-Mモデル単位のセキュリティをサポートします。

- 通常、セキュリティに厳しいアプリケーションが実行されたり、重要なリソースが配置されたりするセキュアなワールド
- 非セキュアワールドまたはパブリックワールド（通常の非セキュアなオペレーティングシステムやユーザ空間など）

TrustZoneアーキテクチャは、以下の手段によりAHBおよびArmv8-Mの範囲外まで拡張されます。

- APB エージェントに対するセキュア / 非セキュアおよび特権 / 非特権トランザクションをブロックまたは伝搬するためのセキュアゲートとして、AHB/APBブリッジを用いる
- AHB エージェントに対するセキュア / 非セキュアおよび特権 / 非特権トランザクションをブロックまたは伝搬するためのセキュアゲートとして、PPC（ペリフェラル保護コントローラ）を用いる
- 内部SRAMへのセキュア/非セキュア、特権/非特権アクセスをフィルタするためのセキュアゲートとして、ブロックベースのTrustZone MPCファイアウォールを用いる
- 外部メモリへのセキュア/非セキュア、特権/非特権アクセスをフィルタするためのセキュアゲートとして、ウォーターマークベースのTrustZone MPCファイアウォールを用いる

AHBおよびAPBペリフェラルは以下のように分類することができます。

- **特権ペリフェラル**：特権属性を定義するために、TZSC コントローラから制御される AHB/APB ファイアウォールスタブによって保護されているペリフェラル。
- **セキュアペリフェラル**：AHB/APB ファイアウォール･スタブによって常時保護されているペリフェラル。これらのペリフェラルは常にセキュアです（TZICなど）。
- **セキュリティ保護可能なペリフェラル**：セキュリティ属性を定義するために、TZSCから制御されるAHB/APBファイアウォール･スタブによって保護されているペリフェラル（オプション）。
- **非セキュアおよび非特権ペリフェラル**：一切のセキュアゲートなしで直接AHB/APBに相互接続されているペリフェラル。
- **TrustZone対応ペリフェラル**：AHBまたはAPBバスに直接接続され、TrustZone固有の動作（セキュアなレジスタのサブセットなど）を実装しているペリフェラル。TrustZone対応AHBマスタは、常にそのセキュリティモードに合ったHNONSEC信号をドライブします（Armv8-Mコアまたは DMAなど）。

AHBによるセキュリティ保護可能なマスタは、TZSCでセキュア/非セキュアや特権/非特権に設定することができます。

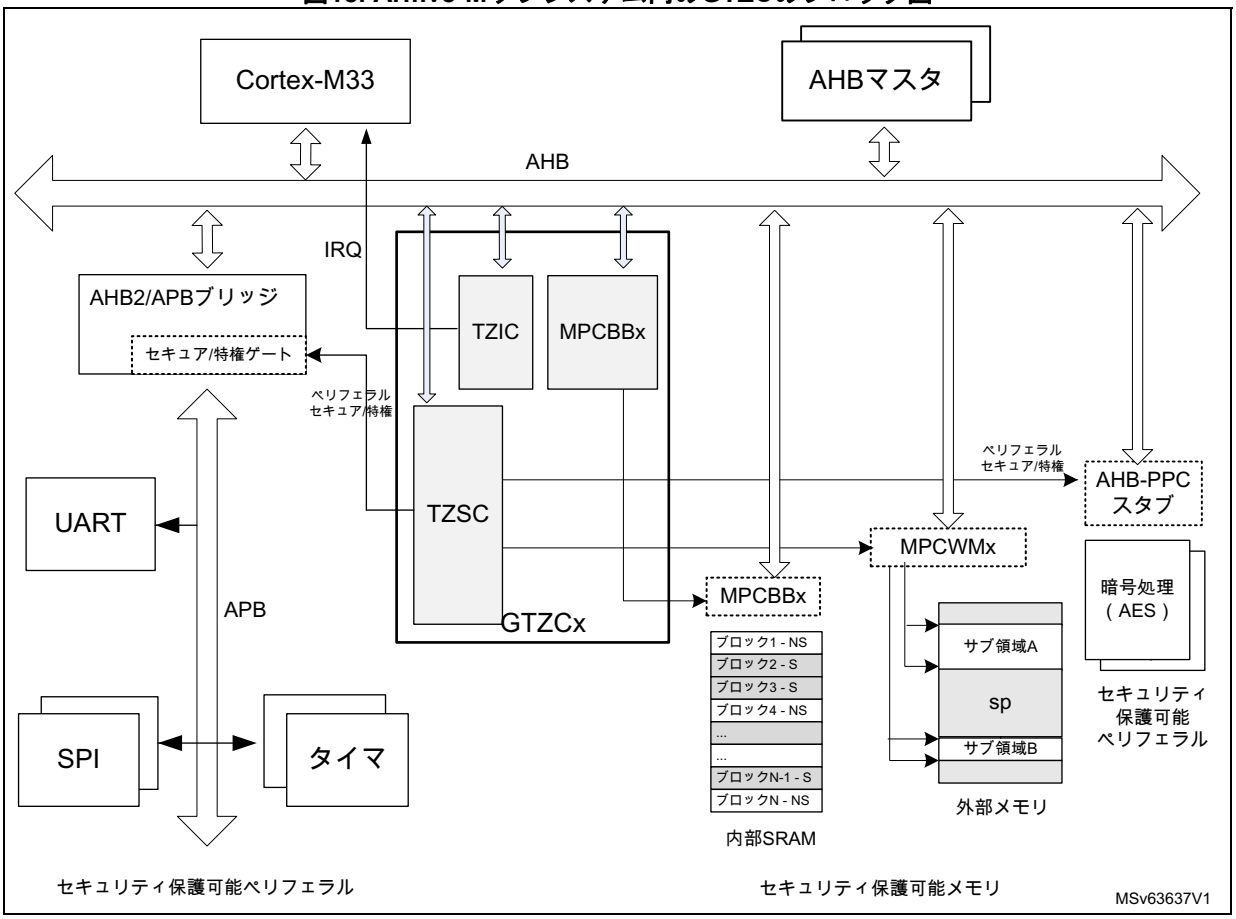
アプリケーション情報

TZSC、MPCBB、およびTZICは次のいずれかの方法で使用できます。

- 必ずセキュア･ブート中にプログラムし、その後はロックした状態で変更しません。
- 特定のアプリケーションコードまたはセキュアカーネル（マイクロバイザ）の使用時に動的に再プログラムします。ロックされていない場合、MPCセキュアブロックまたは領域のサイズは、セキュアFLASH領域またはセキュアSRAMから実行するセキュアソフトウェアによって変更できます。各ペリフェラルのセキュア/特権状態を定義するGTZC_TZSC_SECCFGRx/PRIVCFGRxレジスタにも同じことが言えます。

セキュア、セキュリティ保護可能、およびTrustZoneを認識するペリフェラルを備えたArmv8-Mセキュリティアーキテクチャを下の図に示します。

図18. Armv8-Mサブシステム内のGTZCのブロック図



5.3 GTZCの実装

STM32U5シリーズデバイスには、GTZCのインスタンスが2つ内蔵されています。

表28.GTZCの機能

GTZCサブブロック	GTZC1	GTZC2
TZSC	X	X
TZIC	X	X
STM32U535/545のMPCBBサブブロック	MPCBB1/2	MPCBB4
STM32U575/585のMPCBBサブブロック	MPCBB1/2/3	
STM32U59x/5AxのMPCBBサブブロック	MPCBB1/2/3/5	
STM32U5Fx/5GxのMPCBBサブブロック	MPCBB1/2/3/5/6	

次の表に、GTZCベースアドレスに対するGTZCサブブロックのアドレスオフセットを示します（GTZC1およびGTZC2ベースアドレスについては、セクション2.3を参照）。

表29.GTZC1サブブロックのアドレスオフセット

GTZC1サブブロック	アドレスオフセット
GTZC1_TZSC	0x0
GTZC1_TZIC	0x400
GTZC1_MPCBB1	0x800
GTZC1_MPCBB2	0xC00
GTZC1_MPCBB3	0x1000
GTZC1_MPCBB5	0x1400
GTZC1_MPCBB6	0x1800

表30.GTZC2サブブロックのアドレスオフセット

GTZC2サブブロック	アドレスオフセット
GTZC2_TZSC	0x0
GTZC2_TZIC	0x400
GTZC2_MPCBB4	0x800

次の表に、使用可能なMPCWMの特性を示します。

表31.MPCWMリソースの割当

GTZC	MPC	対象メモリインタフェース	セキュア/非セキュアおよび特権/非特権領域の数	ウォーターマークの粒度（バイト）	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
GTZC1	MPCWM1	OCTOSPI1	2	128 K	X	X	X	X
	MPCWM2	FSMC_NORバンク	2	128 K	-	X	X	X
	MPCWM3	FSMC_NANDバンク	1	128 K	-	X	X	X
	MPCWM4	BKPSRAM	1	32	X	X	X	X
	MPCWM5	OCTOSPI2	2	128 K	-	X	X	X
	MPCWM6	HSPI1	2	128 K	-	-	X	X

次の表に、使用可能なMPCBBの特性を示します。

表32.STM32U535/545 に対するMPCBBリソースの割当

GTZC	MPC	リソース	メモリサイズ（KB）	ブロック･サイズ（バイト）	ブロック数	スーパーブロックの数
GTZC1	MPCBB1	SRAM1	192	512	384	12
	MPCBB2	SRAM2	64		128	4
GTZC2	MPCBB4	SRAM4	16		32	1

表33. に対するMPCBBリソースの割当

GTZC	MPC	リソース	メモリサイズ（KB）	ブロック･サイズ（バイト）	ブロック数	スーパーブロックの数
GTZC1	MPCBB1	SRAM1	192	512	384	12
	MPCBB2	SRAM2	64		128	4
	MPCBB3	SRAM3	512		1024	32
GTZC2	MPCBB4	SRAM4	16		32	1

表34. に対するMPCBBリソースの割当

GTZC	MPC	リソース	メモリサイズ（KB）	ブロック･サイズ（バイト）	ブロック数	スーパーブロックの数
GTZC1	MPCBB1	SRAM1	768	512	1536	48
	MPCBB2	SRAM2	64		128	4
	MPCBB3	SRAM3	832		1664	52
GTZC2	MPCBB4	SRAM4	16		32	1
GTZC1	MPCBB5	SRAM5	832		1664	52

表35.STM32U5Fx/5Gx に対するMPCBBリソースの割当

GTZC	MPC	リソース	メモリサイズ（KB）	ブロック･サイズ（バイト）	ブロック数	スーパーブロックの数
GTZC1	MPCBB1	SRAM1	768	512	1536	48
	MPCBB2	SRAM2	64		128	4
	MPCBB3	SRAM3	832		1664	52
GTZC2	MPCBB4	SRAM4	16		32	1
GTZC1	MPCBB5	SRAM5	832		1664	52
	MPCBB6	SRAM6	512		1024	32

5.4 GTZCの機能詳細

5.4.1 GTZCブロック図

TZSC、MPCBB、およびTZICの機能を組み合わせたものを下の図に示します。各サブブロックは、独自のAHB設定ポートで制御されます。

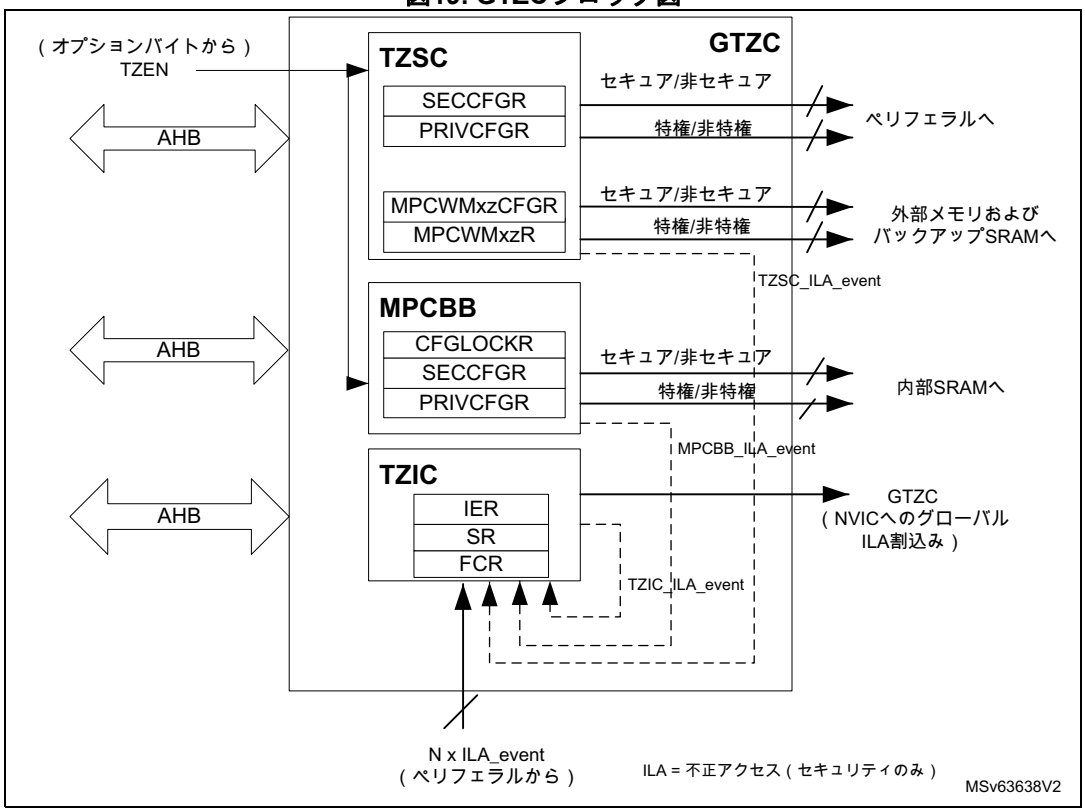
TZSCはセキュア、特権、およびその両方であるペリフェラルを定義します。ペリフェラルの特権設定ビットは、そのペリフェラルがセキュアとして設定されている場合はセキュア特権トランザクションで変更可能です。それ以外の場合は、特権トランザクション（非セキュア）で十分です。

反対に、ペリフェラルのセキュア設定ビットは、そのペリフェラルが特権に設定されている場合、セキュア特権トランザクションによってのみ変更可能です。そうでない場合、（特権ではない）セキュアトランザクションで十分です。

ある特定のRAMブロックのセキュア設定ビットは、そのRAMブロックが特権に設定されている場合、セキュア特権トランザクションでのみ変更が可能です。そうでない場合、（特権ではない）セキュアトランザクションで十分です。

TZICは不正アクセスが検出された場合、システム内に生成された不正イベントを収集します。TZICはその後必要に応じて、CPUに対しセキュアな割込みを生成します。

図19. GTZCブロック図



5.4.2 不正アクセスの定義

次の3種類の不正アクセスが存在します。

- 不正な非セキュアアクセス

  セキュアリソースの書込みを試みる非セキュアなトランザクションは不正アクセスとみなされます。そのため、そのアドレス指定されたリソースから不正書込みアクセスの不正アクセス割込み、および不正フェッチアクセスのバスエラーが生成されます。ただし、セキュアおよび特権設定レジスタには例外があり、、これらの設定レジスタはセキュアレジスタへの非セキュア読出しアクセスを認可します（GTZC_TZSC_SECCFGRxおよびGTZC_TZSC_PRIVCFGRxを参照）。

- 不正なセキュアアクセス

  ブロックベース内部SRAMまたはウォーターマークベースメモリにおいて非セキュアブロックへのアクセスを試みるセキュアトランザクションは、不正アクセスとみなされます。

  TZICの正しい設定は、関連イベントのキャプチャを可能にし、さらにNVICへのGTZC_IRQn割込みを生成します。これには、書込み、読出し、および実行アクセスが該当します。

  MPCBBコントローラについては、GTZC_MPCBBz_CRレジスタのSRWILADISビットをセットすることによって非セキュアSRAMブロックへのセキュアなデータ読出し/書込みアクセスを無視するという選択肢があります。その場合、非セキュアSRAMブロックに対するセキュアなデータ読出しおよび書込みトランザクションは許可されますが、セキュアな実行アクセスは許可されないままです。

  非セキュアペリフェラルレジスタへのアクセスを試みるセキュア実行トランザクションは不正アクセスとみなされ、バスエラーが生成されます。

- 不正な非特権アクセス

  特権リソースへのアクセスを試みる非特権トランザクションはすべて不正アクセスとみなされます。不正読出しおよび書込みアクセスに対して生成される不正アクセスイベントはありません。アドレス指定されたリソースはサイレントフェイル動作に従い、読出しにはすべてゼロデータを返し、書込みはすべて無視します。バスエラーは生成されません。非特権実行トランザクションが特権メモリへのアクセスを試みた場合、バスエラーが生成されます。

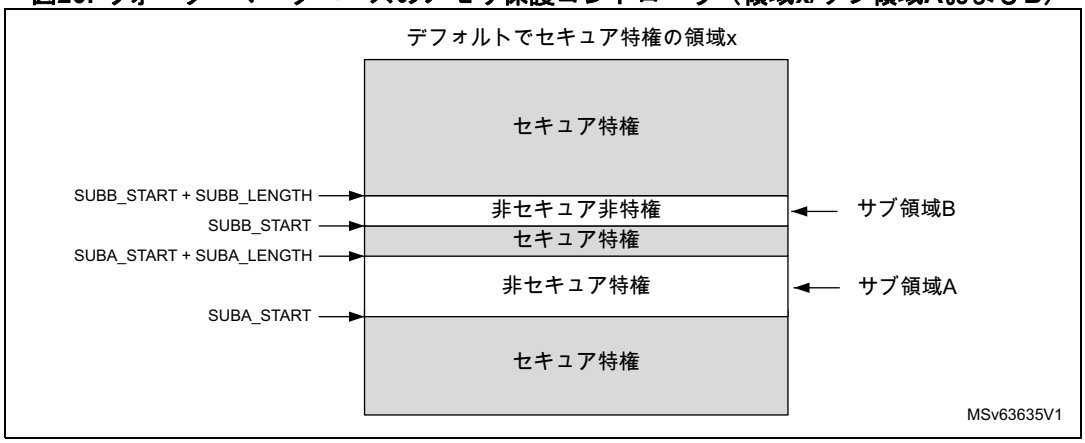
5.4.3 TrustZoneセキュリティ・コントローラ（TZSC）

TZSCは、設定可能な一連のレジスタで構成され、以下の機能を提供します。

- 次のレジスタによる、すべてのペリフェラルのセキュアおよび特権状態の制御。
  - セキュリティ保護可能なペリフェラルの AHB/APB ファイアウォールスタブを制御するためのGTZC_TZSC_SECCFGRxレジスタ
  - 特権ペリフェラルのAHB/APBファイアウォールスタブを制御するための GTZC_TZSC_PRIVCFGRxレジスタ
- ウォーターマークベースのメモリ保護コントローラ（外部メモリおよびバックアップSRAM）には2つの独立した領域を定義することができ、次のフィールドを使ってプログラムをします。
  - 外部メモリ/バックアップSRAMの1番目の保護されたサブ領域の開始： SUBA_START[10:0]
  - 外部メモリ/バックアップSRAMの1番目の保護されたサブ領域の長さ： SUBA_LENGTH[11:0]
  - 外部メモリ/バックアップSRAMの2番目の保護されたサブ領域の開始： SUBA_START[10:0]
  - 外部メモリ/バックアップSRAMの2番目の保護されたサブ領域の長さ： SUBB_LENGTH[11:0]

各サブ領域の制御レジスタは、ウォーターマークベースのメモリ保護コントローラを有効化/無効化するほか、各サブ領域の適切な属性の定義に使用できます。

図20. ウォーターマークベースのメモリ保護コントローラ（領域x/サブ領域AおよびB）



上図において、領域xは外部メモリまたはバックアップSRAM領域（FSMCバンク、OCTOSPI1、OCTOSPI2、HSPI1、またはBKPSRAMなど）を表します。サブ領域AとBのセキュアおよび特権属性は、個別に設定可能です。領域xにおいてサブ領域が定義または有効化されていない場合、領域xのデフォルト属性は「セキュア特権」としてセットされます。

次の表に、非セキュア領域がオーバーラップしている場合の、サブ領域AとBの共通領域のセキュア/特権特性について説明します。

表36.サブ領域AおよびBのセキュア属性

サブ領域A	サブ領域B	領域AとBのオーバーラップ領域の属性
非セキュア	非セキュア	非セキュア
非セキュア	セキュア	非セキュア
セキュア	非セキュア	非セキュア
セキュア	セキュア	セキュア

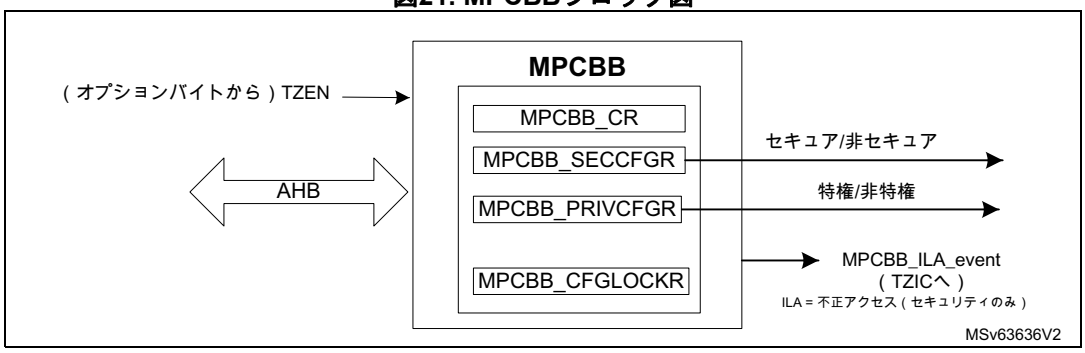
表37.サブ領域AおよびBの特権属性

サブ領域A	サブ領域B	領域AとBのオーバーラップ領域のプロパティ
非特権	非特権	非特権
非特権	特権	非特権
特権	非特権	非特権
特権	特権	特権

5.4.4 ブロックベースのメモリ保護コントローラ（MPCBB）

MPCBBは一連の設定可能なレジスタで構成され、内蔵SRAMメモリのセキュリティと特権ポリシーの定義を可能にします。セキュリティおよび特権ポリシーは、SRAMの各512バイト･ブロックごとに個別に設定が可能です。

図21. MPCBBブロック図



MPCBBを設定するには、以下の操作が必要です（たとえばブート時）。

- セキュアファームウェアは、GTZC_MPCBBz_SECCFGRx に正しいビットをセットすることによって、どのメモリブロックがセキュアであるかを定義する必要があります。
- 特権ファームウェアは、GTZC_MPCBBz_PRIVCFGRxに正しいビットをセットすることによって、どのメモリブロックが特権であるかを定義する必要があります。

MPCBBスーパーブロックは32の連続したブロックからなります。セキュア･アプリケーションは、各スーパーブロックに対し、GTZC_MPCBBz_CFGLOCKR1/2の正しいビットを使用して、すべての関連するセキュリティ /特権ビットをロックできます。このロックは次のシステム･リセットまで有効です。

注： **ブロック･サイズは512バイトです。スーパーブロック･サイズは512*32 = 16KBです。**

5.4.5 TrustZone不正アクセスコントローラ（TZIC）

TZICは、すべての不正アクセスのソースとなるイベントを集中管理します。システムのTrustZone機能（TZEN = 1）が有効な場合のみ使用します。

TZICは、セキュア不正アクセス割込みをトリガしたイベントの追跡（フラグ）を許可します。不要なイベントをフィルタするには、レジスタマスク（GTZC_TZIC_IERx）を使用できます。マスクされていない不正イベントでは、TZICがNVICに対してGTZC_IRQn割込みを生成します。

不正イベントソースごとに、ステータスフラグとクリアビットが存在します（それぞれGTZC_TZIC_SRxおよびGTZC_TZIC_FCRx内に）。マスクレジスタ（GTZC_TZIC_IERx）のリセット値では、すべてのイベントがマスクされていなければなりません。

5.4.6 パワーオン/リセット状態

TZSCのパワーオンおよびリセット状態は、GTZC_TZSC_SECCFGRxおよびGTZC_TZSC_PRIVCFGRxのすべてのビットをクリアして0にします。これは、すべてのセキュリティ保護可能ペリフェラルが、それぞれ非セキュアおよび非特権にセットされることを意味します。

内部SRAMx（x = 1から6）の場合、すべてのGTZC_MPCBBz_SECCFGRx and GTZC_MPCBBz_PRIVCFGRxが次のようにセットされます。

- 0xFFFF FFFF：TrustZoneセキュリティがシステム･レベルで有効になっている場合（TZEN = 1）、これらの内部メモリのブロックをデフォルトでセキュアおよび特権にします。

- 0x0000 0000 : TrustZoneセキュリティがシステム･レベルで無効になっている場合（TZEN = 0）、これらの内部メモリのブロックをデフォルトで非セキュアおよび非特権にします。

外部メモリおよびバックアップSRAMの場合 :

- GTZC_TZSC_MPCWMxzRレジスタはすべて0x0000　0000にセットされ、TrustZoneセキュリティがシステムレベルで有効化されたときに（TZEN = 1）、これらのメモリをデフォルトでセキュアおよび特権にします。
- GTZC_TZSC_MPCWMxzRレジスタにはアクセスできず、TrustZoneセキュリティがシステムレベルで無効化されたときに（TZEN = 0）、これらのメモリはデフォルトで非セキュアかつ非特権です。

次に、セキュア･ブート･コードによりセキュリティ設定をプログラムし、必要に応じてコンポーネントをセキュアまたは非セキュアにすることができます。

5.5 GTZC割込み

TZICはセキュアペリフェラルなので、非セキュア･アクセスが行われると、自動的に不正アクセスイベントを生成します。MPCBBとTZSCはTrustZoneを認識するペリフェラルであり、ペリフェラル内にセキュアレジスタと非セキュアレジスタが混在しています。

表38.GTZC割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
GTZC	不正アクセス	GTZC_TZIC_SRxのすべてのフラグ	GTZC_TZIC_IERxのすべてのビット	ビットGTZC_TZIC_FCRxに1を書き込む	あり	あり	なし

5.6 GTZC1 TZSCレジスタ

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.6.1 GTZC1 TZSC制御レジスタ（GTZC1_TZSC_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCK
															rs

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 LCK : GTZC1_TZSC_SECCFGRxおよびGTZC1_TZSC_PRIVCFGRxレジスタの設定を次のリセットまでロック

このビットはデフォルトではクリアされており、一旦、設定すると、システムリセットまでリセットできません。

0：ロックされていないすべてのGTZC1_TZSC_SECCFGRxおよびGTZC1_TZSC_PRIVCFGRxレジスタの設定

1：ロックされているすべてのGTZC1_TZSC_SECCFGRxおよびGTZC1_TZSC_PRIVCFGRxレジスタの設定

5.6.2 GTZC1 TZSCセキュア設定レジスタ1（GTZC1_TZSC_SECCFGR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

書込みセキュアアクセスのみ。

このレジスタへの書込みは、対応するGTZC1_TZSC_PRIVCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のPRIVビットが設定されていない場合、相当するSECビットはセキュア非特権トランザクションによって書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6SEC	I2C5SEC	USART6SEC	Res.	UCPD1SEC	FDCAN1SEC	LPTIM2SEC	I2C4SEC
								rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRSSEC	I2C2SEC	I2C1SEC	UART5SEC	UART4SEC	USART3SEC	USART2SEC	SPI2SEC	IWDGSEC	WWDGSEC	TIM7SEC	TIM6SEC	TIM5SEC	TIM4SEC	TIM3SEC	TIM2SEC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **I2C6SEC**：I2C6のセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **I2C5SEC**：I2C5のセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット21 **USART6SEC**：USART6のセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **UCPD1SEC**：UCPD1のセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **FDCAN1SEC**：FDCAN1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット17 **LPTIM2SEC**：LPTIM2のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット16 **I2C4SEC**：I2C4のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット15 **CRSSEC**：CRSのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット14 **I2C2SEC**：I2C2のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット13 **I2C1SEC**：I2C1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット12 **UART5SEC**：UART5のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット11 **UART4SEC**：UART4のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット10 **USART3SEC**：USART3のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット9 **USART2SEC**：USART2のセキュアアクセスモード
0：非セキュア
1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット8 **SPI2SEC**：SPI2のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット7 **IWDGSEC**：IWDGのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット6 **WWDGSEC**：WWDGのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット5 **TIM7SEC**：TIM7のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット4 **TIM6SEC**：TIM6のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット3 **TIM5SEC**：TIM5のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット2 **TIM4SEC**：TIM4のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット1 **TIM3SEC**：TIM3のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット0 **TIM2SEC**：TIM2のセキュアアクセスモード
0：非セキュア
1：セキュア

5.6.3 GTZC1 nnTZSCセキュア設定レジスタ2（GTZC1_TZSC_SECCFGR2）

アドレスオフセット：0x014

リセット値：0x0000 0000

書込みセキュアアクセスのみ。

このレジスタへの書込みは、対応するGTZC1_TZSC_PRIVCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のPRIVが設定されていない場合、相当するSECビットはセキュア非特権トランザクションによって書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	GFXTIMSEC	DSISEC	LTDCUSBSEC	SAI2SEC	SAI1SEC	TIM17SEC	TIM16SEC	TIM15SEC	USART1SEC	TIM8SEC	SPI1SEC	TIM1SEC
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **GFXTIMSEC**：GFXTIMのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **DSISEC**：DSIのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **USBFSSEC**：LTDCまたはUSBのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U59x/5Ax/5Fx/5GxのLTDCでセキュアです。STM32U535/545のUSBでセキュアです。STM32U575/585では予約済みです。

ビット8 **SAI2SEC**：SAI2のセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット7 **SAI1SEC**：SAI1のセキュアアクセスモード

0：非セキュア

1：セキュア

ビット6 **TIM17SEC**：TIM7のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット5 **TIM16SEC**：TIM6のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット4 **TIM15SEC**：TIM5のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット3 **USART1SEC**：USART1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット2 **TIM8SEC**：TIM8のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット1 **SPI1SEC**：SPI1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット0 **TIM1SEC**：TIM1のセキュアアクセスモード
0：非セキュア
1：セキュア

5.6.4 GTZC1 TZSCセキュア設定レジスタ3（GTZC1_TZSC_SECCFGR3）

アドレスオフセット：0x018

リセット値：0x0000 0000

書込みセキュアアクセスのみ。

このレジスタへの書込みは、対応するGTZC1_TZSC_PRIVCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のPRIVが設定されていない場合、相当するSECビットはセキュア非特権トランザクションによって書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	JPEGSEC	DCACHE2_REGSEC	HSPI1_REGSEC	GFXMMU_REGSEC	GFXMMUSEC	GPU2DSEC	RAMCFGSEC	OCTOSPI2_REGSEC	OCTOSPI1_REGSEC	FSMC_REGSEC	SDMMC2SEC	SDMMC1SEC	OCTOSPIMSEC
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAESSEC	PKASEC	RNGSEC	HASHSEC	AESSEC	OTGSEC	DCMISEC	ADC12SEC	DCACHE1_REGSEC	ICACHE_REGSEC	DMA2DSEC	TSCSEC	CRCSEC	FMACSEC	CORDICSEC	MDF1SEC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **JPEGSEC**：JPEGのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **DCACHE2_REGSEC**：DCACHE2レジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット26 **HSPI1_REGSEC**：HSPI1レジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット25 **GFXMMU_REGSEC**：GFXMMUレジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット24 **GFXMMUSEC**：GFXMMUのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット23 **GPU2DSEC**：GPU2Dのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **RAMCFGSEC**：RAMCFGのセキュアアクセスモード

0：非セキュア

1：セキュア

ビット21 **OCTOSPI2_REGSEC**：OCTOSPI2レジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 **OCTOSPI1_REGSEC**：OCTOSPI1レジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

ビット19 **FSMC_REGSEC**：FSMCレジスタのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **SDMMC2SEC**：SDMMC1のセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット17 **SDMMC1SEC**：SDMMC2のセキュアアクセスモード

0：非セキュア

1：セキュア

ビット16 **OCTOSPIMSEC**：OCTOSPIMのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット15 **SAESSEC**：SAESのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット14 **PKASEC**：PKAのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット13 **RNGSEC**：RNGのセキュアアクセスモード

0：非セキュア

1：セキュア

ビット12 **HASHSEC**：HASHのセキュアアクセスモード

0：非セキュア

1：セキュア

ビット11 **AESSEC**：AESのセキュアアクセスモード

0：非セキュア

1：セキュア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **OTGSEC**：OTG_FSまたはOTG_HSのセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **DCMISEC**：DCMIおよびPSSIのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット8 **ADC12SEC**：ADC1およびADC2のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット7 **DCACHE1_REGSEC**：DCACHE1レジスタのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット6 **ICACHE_REGSEC**：ICACHEレジスタのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット5 **DMA2DSEC**：DMA2Dのレジスタのセキュアアクセスモード
0：非セキュア
1：セキュア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット4 **TSCSEC**：TSCのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット3 **CRCSEC**：CRCのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット2 **FMACSEC**：FMACのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット1 **CORDICSEC**：CORDICのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット0 **MDF1SEC**：MDF1のセキュアアクセスモード
0：非セキュア
1：セキュア

5.6.5 GTZC1 TZSC特権設定レジス1（GTZC1_TZSC_PRIVCFGR1）

アドレスオフセット：0x020

リセット値：0x0000 0000

書込み特権アクセスのみ。

このレジスタへの読出しまたは書込みは、対応するGTZC1_TZSC_SECCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のSECビットが設定されていない場合、相当するPRIVビットは非セキュア特権トランザクションによって読出しや書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6PRIV	I2C5PRIV	USART6PRIV	Res.	UCPD1PRIV	FDCAN1PRIV	LPTIM2PRIV	I2C4PRIV
								rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRSPRIV	I2C2PRIV	I2C1PRIV	UART5PRIV	UART4PRIV	USART3PRIV	USART2PRIV	SPI2PRIV	IWDGPRIV	WWDGPRIV	TIM7PRIV	TIM6PRIV	TIM5PRIV	TIM4PRIV	TIM3PRIV	TIM2PRIV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **I2C6PRIV**：I2C6の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **I2C5PRIV**：I2C5の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット21 **USART6PRIV**：USART6の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **UCPD1PRIV**：UCPD1の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **FDCAN1PRIV**：FDCAN1の特権アクセスモード
0：非特権
1：特権

ビット17 **LPTIM2PRIV**：LPTIM2の特権アクセスモード
0：非特権
1：特権

ビット16 **I2C4PRIV**：I2C4の特権アクセスモード
0：非特権
1：特権

ビット15 **CRSPRIV**：CRSの特権アクセスモード
0：非特権
1：特権

ビット14 **I2C2PRIV**：I2C2の特権アクセスモード
0：非特権
1：特権

ビット13 **I2C1PRIV**：I2C1の特権アクセスモード
0：非特権
1：特権

ビット12 **UART5PRIV**：UART5の特権アクセスモード
0：非特権
1：特権

ビット11 **UART4PRIV**：UART4の特権アクセスモード
0：非特権
1：特権

ビット10 **USART3PRIV**：USART3の特権アクセスモード
0：非特権
1：特権

ビット9 **USART2PRIV**：USART2の特権アクセスモード
0：非特権
1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット8 **SPI2PRIV**：SPI2の特権アクセスモード
0：非特権
1：特権

ビット7 **IWDGPRIV**：IWDGの特権アクセスモード
0：非特権
1：特権

ビット6 **WWDGPRIV**：WWDGの特権アクセスモード
0：非特権
1：特権

ビット5 **TIM7PRIV**：TIM7の特権アクセスモード
0：非特権
1：特権

ビット4 **TIM6PRIV**：TIM6の特権アクセスモード
0：非特権
1：特権

ビット3 **TIM5PRIV**：TIM5の特権アクセスモード
0：非特権
1：特権

ビット2 **TIM4PRIV**：TIM4の特権アクセスモード
0：非特権
1：特権

ビット1 **TIM3PRIV**：TIM3の特権アクセスモード
0：非特権
1：特権

ビット0 **TIM2PRIV**：TIM2の特権アクセスモード
0：非特権
1：特権

5.6.6 GTZC1 TZSC特権設定レジスタ2（GTZC1_TZSC_PRIVCFGR2）

アドレスオフセット：0x024

リセット値：0x0000 0000

書込み特権アクセスのみ。

このレジスタへの読出しまたは書込みは、対応するGTZC1_TZSC_SECCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のSECビットが設定されていない場合､相当するPRIVビットは非セキュア特権トランザクションによって読出しや書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	GFXTIMPRIV	DSIPRIV	LTDCUSBPRIV	SAI2PRIV	SAI1PRIV	TIM17PRIV	TIM16PRIV	TIM15PRIV	USART1PRIV	TIM8PRIV	SPI1PRIV	TIM1PRIV
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **GFXTIMPRIV**：GFXTIMの特権アクセスモード
0：非特権
1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **DSIPRIV**：DSIの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **LTDCUSBPRIV**：LTDCまたはUSBの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U59x/5Ax/5Fx/5GxのLTDCに特権を与えます。また、STM32U535/545のUSBに特権を与えます。STM32U575/585では予約済みです。

ビット8 **SAI2PRIV**：SAI2の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット7 **SAI1PRIV**：SAI1の特権アクセスモード

0：非特権

1：特権

ビット6 **TIM17PRIV**：TIM17の特権アクセスモード

0：非特権

1：特権

ビット5 **TIM16PRIV**：TIM16の特権アクセスモード

0：非特権

1：特権

ビット4 **TIM15PRIV**：TIM15の特権アクセスモード

0：非特権

1：特権

ビット3 **USART1PRIV**：USART1の特権アクセスモード

0：非特権

1：特権

ビット2 **TIM8PRIV**：TIM8の特権アクセスモード

0：非特権

1：特権

ビット1 **SPI1PRIV**：SPI1PRIVの特権アクセスモード

0：非特権

1：特権

ビット0 **TIM1PRIV**：TIM1の特権アクセスモード

0：非特権

1：特権

5.6.7 GTZC1 TZSC特権設定レジスタ3（GTZC1_TZSC_PRIVCFGR3）

アドレスオフセット：0x028

リセット値：0x0000 0000

書込み特権アクセスのみ。

このレジスタへの読出しまたは書込みは、対応するGTZC1_TZSC_SECCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のSECビットが設定されていない場合、相当するPRIVビットは非セキュア特権トランザクションによって読出しや書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	JPEGPRIV	DCACHE2_REGPRIV	HSPI1_REGPRIV	GFXMMU_REGPRIV	GFXMMUPRIV	GPU2DPRIV	RAMCFGPRIV	OCTOSPI2_REGPRIV	OCTOSPI1_REGPRIV	FSMC_REGPRIV	SDMMC2PRIV	SDMMC1PRIV	OCTOSPIMPRIV
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAESPRIV	PKAPRIV	RNGPRIV	HASHPRIV	AESPRIV	OTGPRIV	DCMIPRIV	ADC12PRIV	DCACHE1_REGPRIV	ICACHE_REGPRIV	DMA2DPRIV	TSCPRIV	CRCPRIV	FMACPRIV	CORDICPRIV	MDF1PRIV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **JPEGPRIV**：JPEGの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **DCACHE2_REGPRIV**：DCACHE2レジスタの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット26 **HSPI1_REGPRIV**：HSPI1レジスタの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット25 **GFXMMU_REGPRIV**：GFXMMUレジスタの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット24 **GFXMMUPRIV**：GFXMMUの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット23 **GPU2DPRIV**：GPU2Dの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **RAMCFGPRIV**：RAMCFGの特権アクセスモード

0：非特権

1：特権

ビット21 **OCTOSPI2_REGPRIV**：OCTOSPI2の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 **OCTOSPI1_REGPRIV**：OCTOSPI1の特権アクセスモード

0：非特権

1：特権

ビット19 **FSMC_REGPRIV**：FSMCレジスタの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **SDMMC2PRIV**：SDMMC1の特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット17 **SDMMC1PRIV**：SDMMC2の特権アクセスモード

0：非特権

1：特権

ビット16 **OCTOSPIMPRIV**：OCTOSPIMの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット15 **SAESPRIV**：SAESの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット14 **PKAPRIV**：PKAの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット13 **RNGPRIV**：RNGの特権アクセスモード

0：非特権

1：特権

ビット12 **HASHPRIV**：HASHの特権アクセスモード

0：非特権

1：特権

ビット11 **AESPRIV**：AESの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **OTGPRIV**：OTG_FSまたはOTG_HSの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **DCMIPRIV**：DCMIおよびPSSIの特権アクセスモード

0：非特権

1：特権

ビット8 **ADC12PRIV**：ADC1およびADC2の特権アクセスモード

0：非特権

1：特権

ビット7 **DCACHE1_REGPRIV**：DCACHE1レジスタの特権アクセスモード

0：非特権

1：特権

ビット6 **ICACHE_REGPRIV**：ICACHEレジスタの特権アクセスモード

0：非特権

1：特権

ビット5 **DMA2DPRIV**：DMA2Dのレジスタの特権アクセスモード

0：非特権

1：特権

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット4 **TSCPRIV**：TSCの特権アクセスモード
0：非特権
1：特権

ビット3 **CRCPRIV**：CRCの特権アクセスモード
0：非特権
1：特権

ビット2 **FMACPRIV**：FMACの特権アクセスモード
0：非特権
1：特権

ビット1 **CORDICPRIV**：CORDICの特権アクセスモード
0：非特権
1：特権

ビット0 **MDF1PRIV**：MDF1の特権アクセスモード
0：非特権
1：特権

5.6.8 GTZC1 TZSCメモリxサブ領域zウォーターマーク設定レジスタ（GTZC1_TZSC_MPCWMxzCFGR）（z = AからB）

アドレスオフセット：ブロックA0x40 + 0x10 * (x-1)(x = 1から6)
アドレスオフセット：ブロックB 0x48 + 0x10 * (x - 1) (x = 1, 2, 5, 6)
リセット値：0x0000 0000

セキュア特権アクセスのみ。

注： **一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
						rw	rw							rs	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **PRIV**：ベース領域xの特権サブ領域z
このビットはSRENが設定されている場合のみ考慮されます。
0：特権および非特権アクセスは、サブ領域zで許可されます。
1：特権アクセスのみが、領域xのサブ領域zで許可されます。

ビット8 **SEC**：ベース領域xのセキュア･サブ領域z
このビットはSRENが設定されている場合のみ考慮されます。
0：非セキュアデータアクセスのみが、領域xのサブ領域zで許可されます。
1：セキュアデータアクセスのみが、領域xのサブ領域zで許可されます。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SRLOCK**：サブ領域zのロック

このビットが一旦設定されると、システムリセットでしかクリアできません。

0：GTZC1_TZSC_MPCWMxCFGR、GTZC1_TZSC_MPCWMxAR、およびGTZC1_TZSC_MPCWMxBRは書込み可能です。

1：GTZC1_TZSC_MPCWMxCFGR、GTZC1_TZSC_MPCWMxAR、およびGTZC1_TZSC_MPCWMxBRへの書込みは無視されます。

ビット0 **SREN**：サブ領域zの有効化

0：サブ領域zは無効です。ベース領域xのアクセス制御が、このサブ領域の開始アドレスと終了アドレスの間のあらゆるアクセスに適用されます。

1：領域xのサブ領域zは有効です。GTZC1_TZSC_MPCWMx_CFGRで定義されたアクセス制御が、GTZC1_TZSC_MPCWMxARとGTZC1_TZSC_MPCWMxBRの両方で定義された、このサブ領域の開始アドレスと終了アドレスの間のあらゆるアクセスに適用されます。

注： ウォーターマーク制御される外部メモリは、TZEN = 0のとき、リセット時点で完全に非セキュア/非特権な状態を開始します。TZEN = 1のとき、外部メモリは完全なセキュア状態/完全な特権状態を開始します（リセット値の反転）。

5.6.9 GTZC1 TZSCメモリxサブ領域Aウォーターマークレジスタ（GTZC1_TZSC_MPCWMxAR）

アドレスオフセット：0x44 + 0x10 *（x - 1）（x = 1から6）
リセット値：0x0000 0000
セキュア特権アクセスのみ。
SUBA_START + SUBA_LENGTHがメモリの最大許容サイズより大きい場合、自動的にSUBA_LENGTHの飽和が適用されます。
サブ領域AとBにオーバーラップが存在する場合、両方のサブ領域のセキュア/特権属性が共通部分に適用されます（セクション5.4.3を参照）。

注： 一部のレジスタは、STM32U5シリーズの一部のデバイスでのみ使用できます。
関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **SUBA_LENGTH[11:0]**：領域xのサブ領域Aの長さ

このフィールドはサブ領域Aの長さを定義し、これに表31で定義された粒度を掛け算します。
SUBA_START + SUBA_LENGTHがメモリに許されている最大許容サイズより大きい場合、SUBA_LENGTHの飽和が自動的に適用されます。
SUBA_LENGTH = 0の場合、サブ領域Aが無効になります（GTZC1_TZSC_MPCMWxACFGRのSRENビットがクリアされます）。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **SUBA_START[10:0]**：領域xのサブ領域Aの開始

領域xの開始に対して、このフィールドでサブ領域Aのアドレスオフセットを定義し、表31で定義された粒度を掛け算します。
ウォーターマーク制御される外部メモリは、TZEN = 0のとき、リセット時点で完全に非セキュアな状態を開始します。TZEN = 1のとき、外部メモリは完全にセキュアな状態を開始します（リセット値の反転）。

5.6.10 GTZC1 TZSCメモリxサブ領域Bウォーターマークレジスタ（GTZC1_TZSC_MPCWMxBR）

アドレスオフセット：0x4C + 0x10 *（x - 1）（x = 1、2、5、6）
リセット値：0x0000 0000
セキュア特権アクセスのみ。
SUBB_START + SUBB_LENGTHがメモリの最大許容サイズより大きい場合、自動的にSUBB_LENGTHの飽和が適用されます。
サブ領域AとBにオーバーラップが存在する場合、両方のサブ領域のセキュア/特権属性が共通部分に適用されます（セクション5.4.3を参照）。

注：　一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SUBB_LENGTH[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SUBB_START[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **SUBA_LENGTH[11:0]**：領域xのサブ領域Bの長さ

このフィールドはサブ領域Bの長さを定義し、これに表31で定義された粒度を掛け算します。
SUBB_START + SUBB_LENGTHがメモリに許されている最大許容サイズより大きい場合、SUBB_LENGTHの飽和が自動的に適用されます。
SUBB_LENGTH = 0の場合、サブ領域Bが無効になります（GTZC1_TZSC_MPCMWxBCFGRのSRENビットがクリアされます）。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **SUBA_START[10:0]**：領域xのサブ領域Bの開始

領域xの開始に対して、このフィールドでサブ領域Bのアドレスオフセットを定義し、表31で定義された粒度を掛け算します。
ウォーターマーク制御される外部メモリは、TZEN = 0のとき、リセット時点で完全に非セキュアな状態を開始します。TZEN = 1のとき、外部メモリは完全にセキュアな状態を開始します（リセット値の反転）。

5.6.11 GTZC1 TZSCレジスタのマップ

表39.GTZC1 TZSCレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC1_TZSC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCK
	リセット値																																0
0x004 - 0x00C	予約済み	予約済み																															
0x010	GTZC1_TZSC_SECCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6SEC	I2C5SEC	USART6SEC	Res.	UCPD1SEC	FDCAN1SEC	LPTIM2SEC	I2C4SEC	CRSSEC	I2C2SEC	I2C1SEC	UART5SEC	UART4SEC	USART3SEC	USART2SEC	SPI2SEC	IWDGSEC	WWDGSEC	TIM7SEC	TIM6SEC	TIM5SEC	TIM4SEC	TIM3SEC	TIM2SEC
	リセット値									0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	GTZC1_TZSC_SECCFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXTMSEC	DSISEC	LTDCUSBSEC	SAI2SEC	SAI1SEC	TIM17SEC	TIM16SEC	TIM15SEC	USART1SEC	TIM8SEC	SPI1SEC	TIM1SEC
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x018	GTZC1_TZSC_SECCFGR3	Res.	Res.	Res.	JPEGSEC	DCACHE2_REGSEC	HSPI1_REGSEC	GFXMMU_REGSEC	GFXMMUSEC	GPU2DSEC	RAMCFGSEC	OCTOSPI2_REGSEC	OCTOSPI1_REGSEC	FSMC_REGSEC	SDMMC2SEC	SDMMC1SEC	OCTOSPIMSEC	SAESSEC	PKASEC	RNGSEC	HASHSEC	AESSEC	OTGSEC	DCMISEC	ADC12SEC	DCACHE1_REGSEC	ICACHE_REGSEC	DMA2DSEC	TSCSEC	CRCSEC	FMACSEC	CORDICSEC	MDF1SEC
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	予約済み	予約済み																															
0x020	GTZC1_TZSC_PRIVCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6PRIV	I2C5PRIV	USART6PRIV	Res.	UCPD1PRIV	FDCAN1PRIV	LPTIM2PRIV	I2C4PRIV	CRSPRIV	I2C2PRIV	I2C1PRIV	UART5PRIV	UART4PRIV	USART3PRIV	USART2PRIV	SPI2PRIV	IWDGPRIV	WWDGPRIV	TIM7PRIV	TIM6PRIV	TIM5PRIV	TIM4PRIV	TIM3PRIV	TIM2PRIV
	リセット値									0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	GTZC1_TZSC_PRIVCFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXTIMPRIV	DSIPRIV	LTDCUSBPRIV	SAI2PRIV	SAI1PRIV	TIM17PRIV	TIM16PRIV	TIM15PRIV	USART1PRIV	TIM8PRIV	SPI1PRIV	TIM1PRIV
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x028	GTZC1_TZSC_PRIVFGR3	Res.	Res.	Res.	JPEGPRIV	DCACHE2_REGPRIV	HSPI1_REGPRIV	GFXMMU_REGPRIV	GFXMMUPRIV	GPU2DPRIV	RAMCFGPRIV	OCTOSPI2_REGPRIV	OCTOSPI1_REGPRIV	FSMC_REGPRIV	SDMMC2PRIV	SDMMC1PRIV	OCTOSPIMPRIV	SAESPRIV	PKAPRIV	RNGPRIV	HASHPRIV	AESPRIV	OTGPRIV	DCMIPRIV	ADC12PRIV	DCACHE1_REGPRIV	ICACHE_REGPRIV	DMA2DPRIV	TSCPRIV	CRCPRIV	FMACPRIV	CORDICPRIV	MDF1PRIV
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C - 0x03C	予約済み	予約済み																															
0x040	GTZC1_TZSC_MPCWM1ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x44	GTZC1_TZSC_MPCWM1AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0	0
0x048	GTZC1_TZSC_MPCWM1BCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x04C	GTZC1_TZSC_MPCWM1BR	Res.	Res.	Res.	Res.	SUBB_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBB_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0	0

表39.GTZC1 TZSCレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x050	GTZC1_TZSC_MPCWM2ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x054	GTZC1_TZSC_MPCWM2AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x058	GTZC1_TZSC_MPCWM2BCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x05C	GTZC1_TZSC_MPCWM2BR	Res.	Res.	Res.	Res.	SUBB_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBB_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x060	GTZC1_TZSC_MPCWM3ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x64	GTZC1_TZSC_MPCWM3AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x068 - 0x06C	予約済み	予約済み																															
0x070	GTZC1_TZSC_MPCWM4ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x074	GTZC1_TZSC_MPCWM4AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x078 ~0x07C	予約済み	予約済み																															
0x080	GTZC1_TZSC_MPCWM5ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x084	GTZC1_TZSC_MPCWM5AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x088	GTZC1_TZSC_MPCWM5BCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x08C	GTZC1_TZSC_MPCWM5BR	Res.	Res.	Res.	Res.	SUBB_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBB_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x090	GTZC1_TZSC_MPCWM6ACFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x094	GTZC1_TZSC_MPCWM6AR	Res.	Res.	Res.	Res.	SUBA_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBA_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x098	GTZC1_TZSC_MPCWM6BCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	SEC	Res.	Res.	Res.	Res.	Res.	Res.	SRLOCK	SREN
	リセット値																							0	0							0	0
0x09C	GTZC1_TZSC_MPCWM6BR	Res.	Res.	Res.	Res.	SUBB_LENGTH[11:0]												Res.	Res.	Res.	Res.	Res.	SUBB_START[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0

表 29：GTZC1サブブロックのアドレスオフセットを参照してください。

5.7 GTZC1 TZICレジスタ

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.7.1 GTZC1 TZIC割込み有効化レジスタ1（GTZC1_TZIC_IER1）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

このレジスタは、不正アクセスの割込みを有効にするために使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6IE	I2C5IE	USART6IE	Res.	UCPD1IE	FDCAN1IE	LPTIM2IE	I2C4IE
								rw	rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRSIE	I2C2IE	I2C1IE	UART5IE	UART4IE	USART3IE	USART2IE	SPI2IE	IWDGIE	WWDGIE	TIM7IE	TIM6IE	TIM5IE	TIM4IE	TIM3IE	TIM2IE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **I2C6IE**：I2C6の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **I2C5IE**：I2C5の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット21 **USART6IE**：USART6の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **UCPD1IE**：UCPD1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **FDCAN1IE**：FDCAN1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット17 **LPTIM2IE**：LPTIM2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット16 **I2C4IE**：I2C4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット15 **CRSIE**：CRSの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット14 **I2C2IE**：I2C2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット13 **I2C1IE**：I2C1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット12 **UART5IE**：UART5の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット11 **UART4IE**：UART4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット10 **USART3IE**：USART3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット9 **USART2IE**：USART2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット8 **SPI2IE**：SPI2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット7 **IWDGIE**：IWDGの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット6 **WWDGIE**：WWDGの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット5 **TIM7IE**：TIM7の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット4 **TIM6IE**：TIM6の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット3 **TIM5IE**：TIM5の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット2 **TIM4IE**：TIM4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット1 **TIM3IE**：TIM3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット0 **TIM2IE**：TIM2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

5.7.2 GTZC1 TZIC割込み有効化レジスタ2（GTZC1_TZIC_IER2）

アドレスオフセット：0x004

リセット値：0x0000 0000

セキュア特権アクセスのみ。

このレジスタは、不正アクセスの割込みを有効にするために使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	GFXTIMIE	DSIIE	LTDCUSBIE	SAI2IE	SAI1IE	TIM17IE	TIM16IE	TIM15IE	USART1IE	TIM8IE	SPI1IE	TIM1IE
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **GFXTIMIE**：GFXTIMの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **DSIIE**：DSIの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **LTDCUSBIE**：LTDCまたはUSBの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U59x/5Ax/5Fx/5GxのLTDCを制御します。また、STM32U535/545のUSBを制御します。STM32U575/585では予約済みです。

ビット8 **SAI2IE**：SAI2の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット7 **SAI1IE**：SAI1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット6 **TIM17IE**：TIM7の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット5 **TIM16IE**：TIM6の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット4 **TIM15IE**：TIM5の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット3 **USART1IE**：USART1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット2 **TIM8IE**：TIM8の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット1 **SPI1IE**：SPI1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット0 **TIM1IE**：TIM1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

5.7.3 GTZC1 TZIC割込み有効レジスタ3（GTZC1_TZIC_IER3）

アドレスオフセット：0x008

リセット値：0x0000 0000

セキュア特権アクセスのみ。

このレジスタは、不正アクセスの割込みを有効にするために使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	JPEGIE	DCACHE2_REGIE	HSPI1_REGIE	GFXMMU_REGIE	GFXMMUIE	GPU2DIE	RAMCFGIE	OCTOSPI2_REGIE	OCTOSPI1_REGIE	FSMC_REGIE	SDMMC2IE	SDMMC1IE	OCTOSPIMIE
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAESIE	PKAIE	RNGIE	HASHIE	AESIE	OTGIE	DCMIIE	ADC12IE	DCACHE1_REGIE	ICACHE_REGIE	DMA2DIE	TSCIE	CRCIE	FMACIE	CORDICIE	MDF1IE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **JPEGIE**：JPEGの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **DCACHE2_REGIE**：DCACHE2レジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット26 **HSPI1_REGIE**：HSPI1レジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット25 **GFXMMU_REGIE**：GFXMMUレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット24 **GFXMMUIE**：GFXMMUの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット23 **GPU2DIE**：GPU2Dの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **RAMCFGIE**：RAMCFGの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット21 **OCTOSPI2_REGIE**：OCTOSPI2レジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 **OCTOSPI1_REGIE**：OCTOSPI1レジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット19 **FSMC_REGIE**：FSMCレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **SDMMC2IE**：SDMMC1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット17 **SDMMC1IE**：SDMMC2の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット16 **OCTOSPIMIE**：OCTOSPIMの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット15 **SAESIE**：SAESの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット14 **PKAIE**：PKAの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット13 **RNGIE**：RNGの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット12 **HASHIE**：HASHの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット11 **AESIE**：AESの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **OTGIE**：OTG_FSまたはOTG_HSの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **DCMIIE**：DCMIおよびPSSIの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット8 **ADC12IE**：ADC1またはADC2の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット7 **DCACHE1_REGIE**：DCACHE1レジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット6 **ICACHE_REGIE**：ICACHEレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット5 **DMA2DIE**：DMA2Dのレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット4 **TSCIE**：TSCの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット3 **CRCIE**：CRCの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット2 **FMACIE**：FMACの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット1 **CORDICIE**：CORDICの不正アクセス割込みの有効化
0：割込みは無効です。
1：割込みは有効です。

ビット0 **MDF1IE**：MDF1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

5.7.4 GTZC1 TZIC割込み有効レジスタ4（GTZC1_TZIC_IER4）

アドレスオフセット：0x00C

リセット値：0x0000 0000

セキュア特権アクセスのみ。

このレジスタは、不正アクセスの割込みを有効にするために使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MPCBB5_REGIE	SRAM5IE	MPCBB3_REGIE	SRAM3IE	MPCBB2_REGIE	SRAM2IE	MPCBB1_REGIE	SRAM1IE	Res.	Res.	Res.	HSPI1_MEMIE	OCTOSPI2_MEMIE	BKPSRAMIE	FSMC_MEMIE	OCTOSPI1_MEMIE
rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TZIC1IE	TZSC1IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTFDEC2IE	OTFDEC1IE	FLASHIE	Flash_REGIE	GPDMA1IE
rw	rw										rw	rw	rw	rw	rw

ビット31 **MPCBB5_REGIE**：MPCBB5レジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット30 **SRAM5IE**：SRAM5の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット29 **MPCBB3_REGIE**：MPCBB3レジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット28 **SRAM3IE**：SRAM3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **MPCBB2_REGIE**：MPCBB2レジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット26 **SRAM2IE**：SRAM2の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット25 **MPCBB1_REGIE**：MPCBB1レジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット24 **SRAM1IE**：SRAM1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **HSPI1_MEMIE**：HSPI1メモリバンクの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット19 **OCTOSPI2_MEMIE**：OCTOSPI2メモリバンクの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **BKPSRAMIE**：MPCWM3（BKPSRAM）メモリバンクの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット17 **FSMC_MEMIE**：MPCWM2（FSMC NAND）およびMPCWM3（FSMC NOR）の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット16 **OCTOSPI1_MEMIE**：MPCWM1（OCTOSPI1）メモリ･バンクの不正アクセス割込みの有効化
0：割込みは無効です。
1：割込みは有効です。

ビット15 **TZIC1IE**：GTZC1 TZICレジスタの不正アクセス割込みの有効化
0：割込みは無効です。
1：割込みは有効です。

ビット14 **TZSC1IE**：GTZC1 TZSCレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット13:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OTFDEC2IE**：OTFDEC2の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット3 **OTFDEC1IE**：OTFDEC1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット2 **FLASHIE**：Flashメモリの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット1 **Flash_REGIE**：Flashレジスタの不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット0 **GPDMA1IE**：GPDMA1の不正アクセス割込み有効化

0：割込みは無効です。

1：割込みは有効です。

5.7.5 GTZC1 TZICステータス・レジスタ1（GTZC1_TZIC_SR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6F	I2C5F	USART6F	Res.	UCPD1F	FDCAN1F	LPTIM2F	I2C4F
								r	r	r		r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRSF	I2C2F	I2C1F	UART5F	UART4F	USART3F	USART2F	SPI2F	IWDGF	WWDGF	TIM7F	TIM6F	TIM5F	TIM4F	TIM3F	TIM2F
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **I2C6F**：I2C6の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **I2C5F**：I2C5の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット21 **USART6F**：USART6の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **UCPD1F**：UCPD1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **FDCAN1F**：FDCAN1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット17 **LPTIM2F**：LPTIM2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット16 **I2C4F**：I2C4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット15 **CRSF**：CRSの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット14 **I2C2F**：I2C2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット13 **I2C1F**：I2C1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット12 **UART5F**：UART5の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット11 **UART4F**：UART4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット10 **USART3F**：USART3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット9 **USART2F**：USART2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット8 **SPI2F**：SPI2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット7 **IWDGF**：IWDGの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット6 **WWDGF**：WWDGの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット5 **TIM7F**：TIM7の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット4 **TIM6F**：TIM6の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット3 **TIM5F**：TIM5の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **TIM4F**：TIM4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **TIM3F**：TIM3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **TIM2F**：TIM2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.7.6 GTZC1 TZICステータスレジスタ2（GTZC1_TZIC_SR2）

アドレスオフセット：0x014

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	GFXTIMF	DSIF	LTDCUSBF	SAI2F	SAI1F	TIM17F	TIM16F	TIM15F	USART1F	TIM8F	SPI1F	TIM1F
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **GFXTIMF**：GFXTIMの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **DSIF**：DSIの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **LTDCUSBF**：LTDCまたはUSBの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U59x/5Ax/5Fx/5GxのLTDCにフラグを立てます。また、STM32U535/545のUSBにフラグを立てます。STM32U575/585では予約済みです。

ビット8 **SAI2F**：SAI2の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット7 **SAI1F**：SAI1の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

ビット6 **TIM17F**：TIM7の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

ビット5 **TIM16F**：TIM6の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

ビット4 **TIM15F**：TIM5の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット3 **USART1F**：USART1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **TIM8F**：TIM8の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **SPI1F**：SPI1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **TIM1F**：TIM1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.7.7 GTZC1 TZICステータス･レジスタ3（GTZC1_TZIC_SR3）

アドレスオフセット：0x018

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	JPEGF	DCACHE2_REGF	HSPI1_REGF	GFXMMU_REGF	GFXMMUF	GPU2DF	RAMCFGF	OCTOSPI2_REGF	OCTOSPI1_REGF	FSMC_REGF	SDMMC2F	SDMMC1F	OCTOSPIMF
			r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAESF	PKAF	RNGF	HASHF	AESF	OTGF	DCMIF	ADC12F	DCACHE1_REGF	ICACHE_REGF	DMA2DF	TSCF	CRCF	FMACF	CORDICF	MDF1F
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **JPEGF**：JPEGの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **DCACHE2_REGF**：DCACHE2レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット26 **HSPI1_REGF**：HSPI1レジスタの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット25 **GFXMMU_REGF**：GFXMMUレジスタの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット24 **GFXMMUF**：GFXMMUの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット23 **GPU2DF**：GPU2Dの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **RAMCFGF**：RAMCFGの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

ビット21 **OCTOSPI2_REGF**：OCTOSPI2レジスタの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 **OCTOSPI1_REGF**：OCTOSPI1レジスタの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

ビット19 **FSMC_REGF**：FSMCレジスタの不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **SDMMC2F**：SDMMC1の不正アクセスフラグ

0：不正アクセスイベントなし

1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット17 **SDMMC1F**：SDMMC2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット16 **OCTOSPIMF**：OCTOSPIMの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット15 **SAESF**：SAESの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット14 **PKAF**：PKAの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット13 **RNGF**：RNGの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット12 **HASHF**：HASHの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット11 **AESF**：AESの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **OTGF**：OTG_FSまたはOTG_HSの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **DCMIF**：DCMIおよびPSSIの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット8 **ADC12F**：ADC1およびADC2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット7 **DCACHE1_REGF**：DCACHE1レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット6 **ICACHE_REGF**：ICACHEレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット5 **DMA2DF**：DMA2Dのレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット4 **TSCF**：TSCの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット3 **CRCF**：CRCの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **FMACF**：FMACの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **CORDICF**：CORDICの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **MDF1F**：MDF1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.7.8 GTZC1 TZICステータスレジスタ4（GTZC1_TZIC_SR4）

アドレスオフセット：0x01C

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MPCBB5_REGF	SRAM5F	MPCBB3_REGF	SRAM3F	MPCBB2_REGF	SRAM2F	MPCBB1_REGF	SRAM1F	Res.	Res.	Res.	HSPI1_MEMF	OCTOSPI2_MEMF	BKPSRAMF	FSMC_MEMF	OCTOSPI1_MEMF
r	r	r	r	r	r	r	r				r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TZIC1F	TZSC1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTFDEC2F	OTFDEC1F	FLASHF	Flash_REGF	GPDMA1F
r	r										r	r	r	r	r

ビット31 **MPCBB5_REGF**：MPCBB5レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット30 **SRAM5F**：SRAM5の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット29 **MPCBB3_REGF**：MPCBB3レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット28 **SRAM3F**：SRAM3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **MPCBB2_REGF**：MPCBB2レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット26 **SRAM2F**：SRAM2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット25 **MPCBB1_REGF**：MPCBB1レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット24 **SRAM1F**：SRAM1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **HSPI1_MEMF**：HSPI1メモリバンクの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット19 **OCTOSPI2_MEMF**：OCTOSPI2メモリバンクの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **BKPSRAMF**：MPCWM3（BKPSRAM）メモリバンクの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット17 **FSMC_MEMF**：MPCWM2（FSMC NAND）およびMPCWM3（FSMC NOR）の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット16 **OCTOSPI1_MEMF**：MPCWM1（OCTOSPI1）メモリ･バンクの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット15 **TZIC1F**：GTZC1 TZICレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット14 **TZSC1F**：GTZC1 TZSCレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット13:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OTFDEC2F**：OTFDEC2の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット3 **OTFDEC1F**：OTFDEC1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **FLASHF**：Flashメモリの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **Flash_REGF**：Flashレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **GPDMA1F**：GPDMA1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.7.9 GTZC1 TZICフラグ･クリア･レジスタ1（GTZC1_TZIC_FCR1）

アドレスオフセット：0x020

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CI2C6F	CI2C5F	CUSART6F	Res.	CUCPD1F	CFDCAN1F	CLPTIM2F	CI2C4F
								w	w	w		w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCRSF	CI2C2F	CI2C1F	CUART5F	CUART4F	CUSART3F	CUSART2F	CSPI2F	CIWDGF	CWWDGF	CTIM7F	CTIM6F	CTIM5F	CTIM4F	CTIM3F	CTIM2F
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **CI2C6F**：I2C6の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **CI2C5F**：I2C5の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット21 **CUSART6F**：USART6の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CUCPD1F**：UCPD1の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **CFDCAN1F**：FDCAN1の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット17 **CLPTIM2F**：LPTIM2の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット16 **CI2C4F**：I2C4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット15 **CCRSF**：CRSの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット14 **CI2C2F**：I2C2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット13 **CI2C1F**：I2C1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット12 **CUART5F**：UART5の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット11 **CUART4F**：UART4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット10 **CUSART3F**：USART3の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット9 **CUSART2F**：USART2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット8 **CSPI2F**：SPI2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット7 **CIWDGF**：IWDGの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット6 **CWWDGF**：WWDGの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット5 **CTIM7F**：TIM7の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット4 **CTIM6F**：TIM6の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット3 **CTIM5F**：TIM5の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット2 **CTIM4F**：TIM4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット1 **CTIM3F**：TIM3の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット0 **CTIM2F**：TIM2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

5.7.10 GTZC1 TZICフラグクリアレジスタ2（GTZC1_TZIC_FCR2）

アドレスオフセット：0x024

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CGFXTIMF	CDSIF	CLTDCUSBF	CSAI2F	CSAI1F	CTIM17F	CTIM16F	CTIM15F	CUSART1F	CTIM8F	CSPI1F	CTIM1F
				w	w	w	w	w	w	w	w	w	w	w	w

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **CGFXTIMF**：GFXTIMの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **CDSIF**：DSIの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **CLTDCUSBF**：LTDCまたはUSBの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

注： このビットは、STM32U59x/5Ax/5Fx/5GxのLTDCを制御します。また、STM32U535/545のUSBを制御します。STM32U575/585では予約済みです。

ビット8 **CSAI2F**：SAI2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット7 **CSAI1F**：SAI1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット6 **CTIM17F**：TIM7の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット5 **CTIM16F**：TIM6の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット4 **CTIM15F**：TIM5の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット3 **CUSART1F**：USART1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット2 **CTIM8F**：TIM8の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット1 **CSPI1F**：SPI1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット0 **CTIM1F**：TIM1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

5.7.11 GTZC1 TZICフラグ･クリア･レジスタ3（GTZC1_TZIC_FCR3）

アドレスオフセット：0x028

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CJPEGF	CDCACHE2_REGF	CHSPI1_REGF	CGFXMMU_REGF	CGFXMMUF	CGPU2DF	CRAMCFGF	COCTOSPI2_REGF	COCTOSPI1_REGF	CFSMC_REGF	CSDMMC2F	CSDMMC1F	COCTOSPIMF
			w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSAESF	CPKAF	CRNGF	CHASHF	CAESF	COTGF	CDCMIF	CADC12F	CDCACHE1_REGF	CICACHE_REGF	CDMA2DF	CTSCF	CCRCF	CFMACF	CCORDICF	CMDF1F
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **CJPEGF**：JPEGの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **CDCACHE2_REGF**：DCACHE2レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット26 **CHSPI1_REGF**：HSPI1レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット25 **CGFXMMU_REGF**：GFXMMUレジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット24 **CGFXMMUF**：GFXMMUの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット23 **CGPU2DF**：GPU2Dの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット22 **CRAMCFGF**：RAMCFGの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット21 **COCTOSPI2_REGF**：OCTOSPI2レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット20 **COCTOSPI1_REGF**：OCTOSPI1レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット19 **CFSMC_REGF**：FSMCレジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **CSDMMC2F**：SDMMC1の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット17 **CSDMMC1F**：SDMMC2の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット16 **COCTOSPIMF**：OCTOSPIMの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット15 **CSAESF**：SAESの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット14 **CPKAF**：PKAの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット13 **CRNGF**：RNGの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット12 **CHASHF**：HASHの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット11 **CAESF**：AESの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット10 **COTGF**：OTG_FSまたはOTG_HSの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット9 **CDCMIF**：DCMIおよびPSSIの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット8 **CADC12F**：ADC1およびADC2の不正アクセスフラグのクリア

0：効果なし

1：ステータスフラグのクリア

ビット7 **CDCACHE1_REGF**：DCACHE1レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット6 **CICACHE_REGF**：ICACHEレジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット5 **CDMA2DF**：DMA2Dのレジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット4 **CTSCF**：TSCの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット3 **CCRCF**：CRCの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット2 **CFMACF**：FMACの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット1 **CCORDICF**：CORDICの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット0 **CMDF1F**：MDF1の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

5.7.12 GTZC1 TZICフラグクリアレジスタ4（GTZC1_TZIC_FCR4）

アドレスオフセット：0x02C

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMPCBB5_REGF	CSRAM5F	CMPCBB3_REGF	CSRAM3F	CMPCBB2_REGF	CSRAM2F	CMPCBB1_REGF	CSRAM1F	Res.	Res.	Res.	CHSPI1_MEMF	COCTOSPI2_MEMF	CBKPSRAMF	CFSMC_MEMF	COCTOSPI1_MEMF
w	w	w	w	w	w	w	w				w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTZIC1F	CTZSC1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COTFDEC2F	COTFDEC1F	CFLASHF	CFLASH_REGF	CGPDMA1F
w	w										w	w	w	w	w

ビット31 **CMPCBB5_REGF**：MPCBB5レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット30 **CSRAM5F**：SRAM5の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット29 **CMPCBB3_REGF**：MPCBB3レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット28 **CSRAM3F**：SRAM3の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット27 **CMPCBB2_REGF**：MPCBB2レジスタの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット26 **CSRAM2F**：SRAM2の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット25 **CMPCBB1_REGF**：MPCBB1レジスタの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット24 **CSRAM1F**：SRAM1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **CHSPI1_MEMF**：HSPI1メモリバンクの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット19 **COCTOSPI2_MEMF**：OCTOSPI2メモリバンクの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット18 **CBKPSRAMF**：MPCWM3（BKPSRAM）メモリバンクの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット17 **CFSMC_MEMF**：MPCWM2（FSMC NAND）およびMPCWM3（FSMC NOR）の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット16 **COCTOSPI1_MEMF**：MPCWM1（OCTOSPI1）メモリ･バンクの不正アクセスフラグのクリア
0：効果なし
1：ステータスフラグのクリア

ビット15 **CTZIC1F**：GTZC1 TZICレジスタの不正アクセスフラグのクリア
0：効果なし
1：ステータスフラグのクリア

ビット14 **CTZSC1F**：GTZC1 TZSCレジスタの不正アクセスフラグのクリア
0：効果なし
1：ステータスフラグのクリア

ビット13:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **COTFDEC2F**：OTFDEC2の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値を保持してください。

ビット3 **COTFDEC1F**：OTFDEC1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット2 **CFLASHF**：Flashメモリの不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

ビット1 **CFLASH_REGF**：Flashレジスタの不正アクセスフラグのクリア

0：効果なし

1：ステータスフラグのクリア

ビット0 **CGPDMA1F**：GPDMA1の不正アクセスフラグクリア

0：効果なし

1：ステータスフラグのクリア

5.7.13 GTZC1 TZICレジスタ・マップ

表40.GTZC1 TZICレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC1_TZIC_IER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6IE	I2C5IE	USART6IE	Res.	UCPD1IE	FDCAN1IE	LPTIM2IE	I2C4IE	CRSIE	I2C2IE	I2C1IE	UART5IE	UART4IE	USART3IE	USART2IE	SPI2IE	IWDGIE	WWDGIE	TIM7IE	TIM6IE	TIM5IE	TIM4IE	TIM3IE	TIM2IE
	リセット値									0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	GTZC1_TZIC_IER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXTIMIE	DSIIE	LTDCUSBIE	SAI2IE	SAI1IE	TIM17IE	TIM16IE	TIM15IE	USART1IE	TIM8IE	SPI1IE	TIM1IE
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x008	GTZC1_TZIC_IER3	Res.	Res.	Res.	JPEGIE	DCACHE2_REGIE	HSPI1_REGIE	GFXMMU_REGIE	GFXMMUIE	GPU2DIE	RAMCFGIE	OCTOSPI2_REGIE	OCTOSPI1_REGIE	FSMC_REGIE	SDMMC2IE	SDMMC1IE	OCTOSPIMIE	SAESIE	PKAIE	RNGIE	HASHIE	AESIE	OTGIE	DCMIIE	ADC12IE	DCACHE1_REGIE	ICACHE_REGIE	DMA2DIE	TSCIE	CRCIE	FMACIE	CORDICIE	MDF1IE
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	GTZC1_TZIC_IER4	MPCBB5_REGIE	SRAM5IE	MPCBB3_REGIE	SRAM3IE	MPCBB2_REGIE	SRAM2IE	MPCBB1_REGIE	SRAM1IE	Res.	Res.	Res.	HSPI1_MEMIE	OCTOSPI2_MEMIE	BKPSRAMIE	FSMC_MEMIE	OCTOSPI1_MEMIE	TZIC1IE	TZSC1IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTFDEC2IE	OTFDEC1IE	FLASHIE	Flash_REGIE	GPDMA1IE
	リセット値	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0											0	0	0	0	0
0x010	GTZC1_TZIC_SR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C6F	I2C5F	USART6F	Res.	UCPD1F	FDCAN1F	LPTIM2F	I2C4F	CRSF	I2C2F	I2C1F	UART5F	UART4F	USART3F	USART2F	SPI2F	IWDGF	WWDGF	TIM7F	TIM6F	TIM5F	TIM4F	TIM3F	TIM2F
	リセット値									0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	GTZC1_TZIC_SR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXTIMF	DSIF	LTDCUSBF	SAI2F	SAI1F	TIM17F	TIM16F	TIM15F	USART1F	TIM8F	SPI1F	TIM1F
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x018	GTZC1_TZIC_SR3	Res.	Res.	Res.	JPEGF	DCACHE2_REGF	HSPI1_REGF	GFXMMU_REGF	GFXMMUF	GPU2DF	RAMCFGF	OCTOSPI2F	OCTOSPI1F	FSMCF	SDMMC2F	SDMMC1F	OCTOSPIMF	SAESF	PKAF	RNGF	HASHF	AESF	OTGF	DCMIF	ADC12F	DCACHE1F	ICACHEF	DMA2DF	TSCF	CRCF	FMACF	CORDICF	MDF1F
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表40.GTZC1 TZICレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	GTZC1_TZIC_SR4	MPCBB5_REGF	SRAM5F	MPCBB3_REGF	SRAM3F	MPCBB2_REGF	SRAM2F	MPCBB1_REGF	SRAM1F	Res.	Res.	Res.	HSPI1_MEMIF	OCTOSPI2_MEMIF	BKPSRAMIF	FSMC_MEMFE	OCTOSPI1_MEMFE	TZIC1F	TZSC1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTFDEC2F	OTFDEC1F	FLASHF	Flash_REGF	GPDMA1F
	リセット値	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0										0	0	0	0	0
0x020	GTZC1_TZIC_FCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CI2C6F	CI2C5F	CUSART6F	Res.	CUCPD1F	CFDCAN1F	CLPTIM2F	CI2C4F	CCRSF	CI2C2F	CI2C1F	CUART5F	CUART4F	CUSART3F	CUSART2F	CSPI2F	CIWDGF	CWWDGF	CTIM7F	CTIM6F	CTIM5F	CTIM4F	CTIM3F	CTIM2F
	リセット値									0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	GTZC1_TZIC_FCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CGFXTIMF	CDSIF	CLTDCUSBF	CSAI2F	CSAI1F	CTIM17F	CTIM16F	CTIM15F	CUSART1F	CTIM8F	CSPI1F	CTIM1F
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x028	GTZC1_TZIC_FCR3	Res.	Res.	Res.	CJPEGF	CDCACHE2_REGF	CHSPI1_REGF	CGFXMMU_REGF	CGFXMMUF	CGPU2DF	CRAMCFGF	COCTOSPI2F	COCTOSPI1F	CFSMCF	CSDMMC2F	CSDMMC1F	COCTOSPIMF	CSAESF	CPKAF	CRNGF	CHASHF	CAESF	COTGF	CDCMIF	ADC12F	CDCACHE1F	CICACHEF	CDMA2DF	CTSCF	CCRCF	CFMACF	CCORDICF	CMDF1F
					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	リセット値																																
0x02C	GTZC1_TZIC_FCR4	CMPCBB5_REGF	CSRAM5F	CMPCBB3_REGF	CSRAM3F	CMPCBB2_REGF	CSRAM2F	CMPCBB1_REGF	CSRAM1F	Res.	Res.	Res.	CHSPI1_MEMIF	COCTOSPI2_MEMIF	CBKPSRAMIF	CFSMC_MEMFE	COCTOSPI1_MEMFE	CTZIC1F	CTZSC1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COTFDEC2F	COTFDEC1F	CFLASHF	CFLASH_REGF	CGPDMA1F
	リセット値	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0										0	0	0	0	0

表 29：GTZC1サブブロックのアドレスオフセットを参照してください。

5.8 GTZC1 MPCBBzレジスタ（z = 1、2、3、5、6）

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.8.1 GTZC1 SRAMz MPCBB制御レジスタ（GTZC1_MPCBBz_CR）（z = 1、2、3、5、6）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

注： 一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRWILADIS	INVSECSTATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLOCK
															rs

ビット31 SRWILADIS：セキュア読出し/書込み不正アクセス無効化

このビットは、セキュア読出し/書込みトランザクションがブロックベースSRAMの非セキュアブロックにアクセスするとき、不正アクセスの検出を無効にします（非セキュアブロックにおけるセキュアフェッチは常に不正とみなされます）。

0：有効、非セキュアSRAMブロックへのセキュア読出し/書込みアクセスは不可

1：無効、非セキュアSRAMブロックへのセキュア読出し/書込みアクセスは可

ビット30 INVSECSTATE：SRAMxクロックのセキュリティ状態

このビットはRCCの内蔵SRAMのクロック制御がセキュアかどうかを定義するために使用されます。

0：MPCBBにセキュア領域が存在する場合、SRAMのクロックはセキュアです。セキュア領域がない場合は非セキュアです。

1：SRAMのクロックは、MPCBBにセキュア領域が存在しても非セキュアで、MPCBBにセキュアブロックが設定されていない場合でもセキュアです。

ビット29:1 予約済みであり、リセット値に保持する必要があります。

ビット0 GLOCK：MPCBBの制御レジスタを次のリセットまでロック

このビットはデフォルトではクリアされており、一旦、設定すると、システムリセットまでリセットできません。

0：制御レジスタロック解除

1：制御レジスタロック

5.8.2 GTZC1 SRAMz MPCBB設定ロックレジスタ1（GTZC1_MPCBBz_CFGLOCKR1）（z = 1、2、3、5、6）

アドレスオフセット：0x010

リセット値：0x0000 0000

セキュア特権アクセスのみ。

注： **一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPLCK 31	SPLCK 30	SPLCK 29	SPLCK 28	SPLCK 27	SPLCK 26	SPLCK 25	SPLCK 24	SPLCK 23	SPLCK 22	SPLCK 21	SPLCK 20	SPLCK 19	SPLCK 18	SPLCK 17	SPLCK 16
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPLCK 15	SPLCK 14	SPLCK 13	SPLCK 12	SPLCK 11	SPLCK 10	SPLCK 9	SPLCK 8	SPLCK 7	SPLCK 6	SPLCK 5	SPLCK 4	SPLCK 3	SPLCK 2	SPLCK 1	SPLCK 0
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット31:0 **SPLCKy**：スーパーブロックのセキュリティ /特権設定ロック（y = 31から0）
このビットはソフトウェアでセットされ、システムリセットでのみクリアが可能です。
0：GTZC1_MPCBBz_SECCFGRyおよびGTZC1_MPCBBz_PRIVCFGRyは書込み可能です。
1：GTZC1_MPCBBz_SECCFGRyおよびGTZC1_MPCBBz_PRIVCFGRyへの書込みは無視されます。

5.8.3 GTZC1 SRAMz MPCBB設定ロックレジスタ2（GTZC1_MPCBBz_CFGLOCKR2）（z = 1、2、3、5、6）

アドレスオフセット：0x014

リセット値：0x0000 0000

セキュア特権アクセスのみ。

注： **一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPLCK 51	SPLCK 50	SPLCK 49	SPLCK 48
												rs	rs	rs	rs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPLCK 47	SPLCK 46	SPLCK 45	SPLCK 44	SPLCK 43	SPLCK 42	SPLCK 41	SPLCK 40	SPLCK 39	SPLCK 38	SPLCK 37	SPLCK 36	SPLCK 35	SPLCK 34	SPLCK 33	SPLCK 32
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **SPLCKy**：スーパーブロックのセキュリティ /特権設定ロック（y = 51から32）
このビットはソフトウェアでセットされ、システムリセットでのみクリアが可能です。
0：GTZC1_MPCBBz_SECCFGRyおよびGTZC1_MPCBBz_PRIVCFGRyは書込み可能です。
1：GTZC1_MPCBBz_SECCFGRyおよびGTZC1_MPCBBz_PRIVCFGRyへの書込みは無視されます。

5.8.4 スーパーブロックxレジスタのGTZC1 SRAMz MPCBBセキュリティ設定（GTZC1_MPCBBz_SECCFGRx）（z = 1、2、3、5、6）

アドレスオフセット：0x100 + 0x04 * x（x = 0から51）

リセット値：0xFFFF FFFF

指定のリセット値はTZEN = 1のときに有効です。TZEN = 0のときのリセット値は0x0000 0000です。

このレジスタへの書込みアクセスはセキュア·アクセスのみです。あらゆる読込みが許可されています。

注： **一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEC31	SEC30	SEC29	SEC28	SEC27	SEC26	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SECy**：ブロックy（y = 31から0）のセキュリティ設定

0：スーパーブロックxに属するブロックyのみへの非セキュアアクセス。GTZC1_MPCBBz_CRでSRWILADISビットが設定されている場合、セキュアアクセスも許可されます。

1：スーパーブロックxに属するブロックyのみへのセキュアアクセス。

GTZC1_MPCBBz_PRIVCFGRxでPRIVyビットが設定されている場合、このビットへの非特権書込みは無視されます。

GTZC1_MPCBBz_CFGLOCKR1/2のSPLCKxビットが設定されている場合、書込みは無視されます。

5.8.5 スーパーブロックxレジスタのGTZC1 SRAMz MPCBB特権設定（GTZC1_MPCBBz_PRIVCFGRx）（z = 1、2、3、5、6）

アドレスオフセット：0x200 + 0x04 * x、（x = 0から51）

リセット値：0xFFFF FFFF

指定のリセット値はTZEN = 1のときに有効です。TZEN = 0のときのリセット値は0x0000 0000です。

このレジスタへの書込みアクセスは特権アクセスのみです。あらゆる読込みが許されています。

注： **一部のレジスタは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するメモリ領域を使用できるかどうかについては、デバイスのデータシートを参照してください。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRIV31	PRIV30	PRIV29	PRIV28	PRIV27	PRIV26	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRIVy**：スーパーブロックxに属するブロックyの特権設定（y = 31から0）。

0：スーパーブロックxに属するブロックyへの特権および非特権アクセス

1：スーパーブロックxに属するブロックyへの特権アクセスのみ

GTZC1_MPCBBz_SECCFGRxのSECyビットが設定されている場合、このビットへの非セキュア書込みは無視されます。

GTZC1_MPCBBz_CFGLOCKR1/2のSPLCKxビットが設定されている場合、書込みは無視されます。

5.8.6 GTZC1 MPCBBzレジスタマップ（z = 1、2、3、5、6）

表41.GTZC1 MPCBBzレジスタマップおよびリセット値（z = 1、2、3、5、6）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC1_ MPCBBz_CR	SRWILADIS	INVSECSTATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLOCK
	リセット値	0	0																														0
0x004 ~0x00C	予約済み	予約済み																															
0x010	GTZC1_MPCBBz_CFGLOCKR1	SPLCK31	SPLCK30	SPLCK29	SPLCK28	SPLCK27	SPLCK26	SPLCK25	SPLCK24	SPLCK23	SPLCK22	SPLCK21	SPLCK20	SPLCK19	SPLCK18	SPLCK17	SPLCK16	SPLCK15	SPLCK14	SPLCK13	SPLCK12	SPLCK11	SPLCK10	SPLCK9	SPLCK8	SPLCK7	SPLCK6	SPLCK5	SPLCK4	SPLCK3	SPLCK2	SPLCK1	SPLCK0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	GTZC1_MPCBBz_CFGLOCKR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPLCK51	SPLCK50	SPLCK49	SPLCK48	SPLCK47	SPLCK46	SPLCK45	SPLCK44	SPLCK43	SPLCK42	SPLCK41	SPLCK40	SPLCK39	SPLCK3/38	SPLCK37	SPLCK36	SPLCK35	SPLCK34	SPLCK33	SPLCK32
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018 ~0x0FC	予約済み	予約済み																															
0x100 + 0x04 *x （x = 0 ~ 51）	GTZC1_MPCBBz_SECCFGRx	SEC31	SEC30	SEC29	SEC28	SEC27	SEC26	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16	SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x180 ~0x1FC	予約済み	予約済み																															
0x200 + 0x04 *x （x = 0 ~ 51）	GTZC1_MPCBBz_PRIVCFGRx	PRIV31	PRIV30	PRIV29	PRIV28	PRIV27	PRIV26	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16	PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 29：GTZC1サブブロックのアドレスオフセットを参照してください。

5.9 GTZC2 TZSCレジスタ

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.9.1 GTZC2 TZSC制御レジスタ（GTZC2_TZSC_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCK
															rs

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 LCK : GTZC2_TZSC_SECCFGRxおよびGTZC2_TZSC_PRIVCFGRxレジスタの設定を次のリセットまでロック

このビットはデフォルトではクリアされており、一旦、設定すると、システムリセットまでリセットできません。

0：ロックされていないすべてのGTZC2_TZSC_SECCCFGRxおよびすべてのGTZC2_TZSC_PRIVCFGRxレジスタの設定

1：ロックされているすべてのGTZC2_TZSC_SECCCFGRxおよびすべてのGTZC2_TZSC_PRIVCFGRxレジスタの設定

5.9.2 GTZC2 TZSCセキュア設定レジスタ1（GTZC2_TZSC_SECCFGR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

書込みセキュアアクセスのみ。

このレジスタへの書込みは、対応するGTZC2_TZSC_PRIVCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のPRIVビットが設定されていない場合、相当するSECビットはセキュア非特権トランザクションによって書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ADF1SEC	DAC1SEC	Res.	VREFBUFSEC	ADC4SEC	COMPSEC	OPAMPSEC	LPTIM4SEC	LPTIM3SEC	LPTIM1SEC	I2C3SEC	LPUART1SEC	SPI3SEC
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ADF1SEC**：ADF1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット11 **DAC1SEC**：DAC1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **VREFBUFSEC**：VREFBUFのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット8 **ADC4SEC**：ADC4のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット7 **COMPSEC**：COMPのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット6 **OPAMPSEC**：OPAMPのセキュアアクセスモード
0：非セキュア
1：セキュア

ビット5 **LPTIM4SEC**：LPTIM4のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット4 **LPTIM3SEC**：LPTIM3のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット3 **LPTIM1SEC**：LPTIM1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット2 **I2C3SEC**：I2C3のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット1 **LPUART1SEC**：LPUART1のセキュアアクセスモード
0：非セキュア
1：セキュア

ビット0 **SPI3SEC**：SPI3のセキュアアクセスモード
0：非セキュア
1：セキュア

5.9.3 GTZC2 TZSC特権設定レジスタ1（GTZC2_TZSC_PRIVCFGR1）

アドレスオフセット：0x020

リセット値：0x0000 0000

書込み特権アクセスのみ。

このレジスタへの読出しまたは書込みは、対応するGTZC2_TZSC_SECCFGRレジスタの信号が1にセットされている場合に、セキュア特権トランザクションによってのみ可能です。所定のSECビットが設定されていない場合、相当するPRIVビットは非セキュア特権トランザクションによって読出しや書込みができます。

読出しアクセスは、セキュア/非セキュア、特権/非特権のどのタイプのトランザクションにも許可されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ADF1PRIV	DAC1PRIV	Res.	VREFBUFPRIV	ADC4PRIV	COMPPRIV	OPAMPPRIV	LPTIM4PRIV	LPTIM3PRIV	LPTIM1PRIV	I2C3PRIV	LPUART1PRIV	SPI3PRIV
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ADF1PRIV**：ADF1の特権アクセスモード
0：非特権
1：特権

ビット11 **DAC1PRIV**：DAC1の特権アクセスモード
0：非特権
1：特権

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **VREFBUFPRIV**：VREFBUFの特権アクセスモード
0：非特権
1：特権

ビット8 **ADC4PRIV**：ADC4の特権アクセスモード
0：非特権
1：特権

ビット7 **COMPPRIV**：COMPの特権アクセスモード
0：非特権
1：特権

ビット6 **OPAMPPRIV**：OPAMPの特権アクセスモード
0：非特権
1：特権

ビット5 **LPTIM4PRIV**：LPTIM4の特権アクセスモード
0：非特権
1：特権

ビット4 **LPTIM3PRIV**：LPTIM3の特権アクセスモード
0：非特権
1：特権

ビット3 **LPTIM1PRIV**：LPTIM1の特権アクセスモード
0：非特権
1：特権

ビット2 **I2C3PRIV**：I2C3の特権アクセスモード
0：非特権
1：特権

ビット1 **LPUART1PRIV**：LPUART1の特権アクセスモード
0：非特権
1：特権

ビット0 **SPI3PRIV**：SPI3の特権アクセスモード
0：非特権
1：特権

5.9.4 GTZC2 TZSCレジスタマップ

表42.GTZC2 TZSCレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC2_TZSC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCK
	リセット値																																0
0x004 - 0x00C	予約済み	予約済み																															
0x010	GTZC2_TZSC_SECCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1SEC	DAC1SEC	Res.	VREFBUFSEC	ADC4SEC	COMPSEC	OPAMPSEC	LPTIM4SEC	LPTIM3SEC	LPTIM1SEC	I2C3SEC	LPUART1SEC	SPI3SEC
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x014 ~0x01C	予約済み	予約済み																															
0x020	GTZC2_TZSC_PRIVCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1PRIV	DAC1PRIV	Res.	VREFBUFPRIV	ADC4PRIV	COMPPRIV	OPAMPPRIV	LPTIM4PRIV	LPTIM3PRIV	LPTIM1PRIV	I2C3PRIV	LPUART1PRIV	SPI3PRIV
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0

表 30：GTZC2サブブロックのアドレスオフセットを参照してください。

5.10 GTZC2 TZICレジスタ

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.10.1 GTZC2 TZIC割込み有効化レジスタ1（GTZC2_TZIC_IER1）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

このレジスタは、不正アクセスの割込みを有効にするために使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ADF1IE	DAC1IE	Res.	VREFBUFIE	ADC4IE	COMPIE	OPAMPIE	LPTIM4IE	LPTIM3IE	LPTIM1IE	I2C3IE	LPUART1IE	SPI3IE
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ADF1IE**：ADF1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット11 **DAC1IE**：DAC1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **VREFBUFIE**：VREFBUFの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット8 **ADC4IE**：ADC4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット7 **COMPIE**：COMPの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット6 **OPAMPIE**：OPAMPの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット5 **LPTIM4IE**：LPTIM4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット4 **LPTIM3IE**：LPTIM3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット3 **LPTIM1IE**：LPTIM1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット2 **I2C3IE**：I2C3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット1 **LPUART1IE**：LPUART1の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット0 **SPI3IE**：SPI3の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

5.10.2 GTZC2 TZIC割込み有効レジスタ2（GTZC2_TZIC_IER2）

アドレスオフセット：0x004

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MPCBB4_REGIE	SRAM4IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TZIC2IE	TZSC2IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTIIE	LPDMA1IE	RCCIE	PWRIE	TAMPIE	RTCIE	SYSCFGIE
rw	rw								rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **MPCBB4_REGIE**：MPCBB4レジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット24 **SRAM4IE**：SRAM4の不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **TZIC2IE**：GTZC2 TZICレジスタの不正アクセス割込みの有効化
0：割込みは無効です。
1：割込みは有効です。

ビット14 **TZSC2IE**：GTZC2 TZSCレジスタの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット13:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **EXTIIE**：EXTIの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット5 **LPDMA1IE**：LPDMAの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット4 **RCCIE**：RCCの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット3 **PWRIE**：PWRの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット2 **TAMPIE**：TAMPの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット1 **RTCIE**：RTCの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

ビット0 **SYSCFGIE**：SYSCFGの不正アクセス割込み有効化
0：割込みは無効です。
1：割込みは有効です。

5.10.3 GTZC2 TZICステータス・レジスタ1（GTZC2_TZIC_SR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ADF1F	DAC1F	Res.	VREFBUFF	ADC4F	COMPF	OPAMPF	LPTIM4F	LPTIM3F	LPTIM1F	I2C3F	LPUART1F	SPI3F
			r	r		r	r	r	r	r	r	r	r	r	r

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ADF1F**：ADF1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット11 **DAC1F**：DAC1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **VREFBUFF**：VREFBUFの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット8 **ADC4F**：ADC4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット7 **COMPF**：COMPの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット6 **OPAMPF**：OPAMPの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット5 **LPTIM4F**：LPTIM4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット4 **LPTIM3F**：LPTIM3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット3 **LPTIM1F**：LPTIM1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **I2C3F**：I2C3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **LPUART1F**：LPUART1の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **SPI3F**：SPI3の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.10.4 GTZC2 TZICステータスレジスタ2（GTZC2_TZIC_SR2）

アドレスオフセット：0x014

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MPCBB4_REGF	SRAM4F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						r	r								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TZIC2F	TZSC2F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTIF	LPDMA1F	RCCF	PWRF	TAMPF	RTCF	SYSCFGF
r	r								r	r	r	r	r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **MPCBB4_REGF**：MPCBB4レジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット24 **SRAM4F**：SRAM4の不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **TZIC2F**：GTZC2 TZICレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット14 **TZSC2F**：GTZC2 TZSCレジスタの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット13:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **EXTIF**：EXTIの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット5 **LPDMA1F**：LPDMAの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット4 **RCCF**：RCCの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット3 **PWRF**：PWRの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット2 **TAMPF**：TAMPの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット1 **RTCF**：RTCの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

ビット0 **SYSCFGF**：SYSCFGの不正アクセスフラグ
0：不正アクセスイベントなし
1：不正アクセスイベント

5.10.5 GTZC2 TZICフラグ･クリア･レジスタ1（GTZC2_TZIC_FCR1）

アドレスオフセット：0x020

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CADF1F	CDAC1F	Res.	CVREFBUFF	CADC4F	CCOMPF	COPAMPF	CLPTIM4F	CLPTIM3F	CLPTIM1F	CI2C3F	CLPUART1F	CSPI3F
			w	w		w	w	w	w	w	w	w	w	w	w

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CADF1F**：ADF1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット11 **CDAC1F**：DAC1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CVREFBUFF**：VREFBUFの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット8 **CADC4F**：ADC4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット7 **CCOMPF**：COMPの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット6 **COPAMPF**：OPAMPの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット5 **CLPTIM4F**：LPTIM4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット4 **CLPTIM3F**：LPTIM3の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット3 **CLPTIM1F**：LPTIM1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット2 **CI2C3F**：I2C3の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット1 **CLPUART1F**：LPUART1の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット0 **CSPI3F**：SPI3の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

5.10.6 GTZC2 TZICフラグクリアレジスタ2（GTZC2_TZIC_FCR2）

アドレスオフセット：0x024

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	CMPCBB4_REGF	CSRAM4F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						w	w								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTZIC2F	CTZSC2F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEXTIF	CLPDMA1F	CRCCF	CPWRF	CTAMPF	CRTCF	CSYSCFGF
w	w								w	w	w	w	w	w	w

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **CMPCBB4_REGF**：MPCBB4レジスタの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット24 **CSRAM4F**：SRAM4の不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **CTZIC2F**：GTZC2 TZICレジスタの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット14 **CTZSC2F**：GTZC2 TZSCレジスタの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット13:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CEXTIF**：EXTIの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット5 **CLPDMA1F**：LPDMAの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット4 **CRCCF**：RCCの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット3 **CPWRF**：PWRの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット2 **CTAMPF**：TAMPの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット1 **CRTCF**：RTCの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

ビット0 **CSYSCFGF**：SYSCFGの不正アクセスフラグクリア
0：効果なし
1：ステータスフラグのクリア

5.10.7 GTZC2 TZICレジスタマップ

表43.GTZC2 TZICレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC2_TZIC_IER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1IE	DAC1IE	Res.	VREFBUFIE	ADC4IE	COMPIE	OPAMPIE	LPTIM4IE	LPTIM3IE	LPTIM1IE	I2C3IE	LPUART1IE	SPI3IE
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x004	GTZC2_TZIC_IER2	Res.	Res.	Res.	Res.	Res.	Res.	MPCBB4_REGIE	SRAM4IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TZIC2IE	TZSC2IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTIIE	LPDMA1IE	RCCIE	PWRIE	TAMPIE	RTCIE	SYSCFGIE
	リセット値							0	0									0	0								0	0	0	0	0	0	0
0x008 ~0x00C	予約済み	予約済み																															
0x010	GTZC2_TZIC_SR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1F	DAC1F	Res.	VREFBUFF	ADC4F	COMPF	OPAMPF	LPTIM4F	LPTIM3F	LPTIM1F	I2C3F	LPUART1F	SPI3F
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x014	GTZC2_TZIC_SR2	Res.	Res.	Res.	Res.	Res.	Res.	MPCBB4_REGF	SRAM4F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TZIC2F	TZSC2F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTIF	LPDMA1F	RCCF	PWRF	TAMPF	RTCF	SYSCFGF
	リセット値							0	0									0	0								0	0	0	0	0	0	0
0x018 ~0x01C	予約済み	予約済み																															
0x020	GTZC2_TZIC_FCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CADF1F	CDAC1F	Res.	CVREFBUFF	CADC4F	CCOMPF	COPAMPF	CLPTIM4F	CLPTIM3F	CLPTIM1F	CI2C3F	CLPUART1F	CSPI3F
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x024	GTZC2_TZIC_FCR2	Res.	Res.	Res.	Res.	Res.	Res.	CMPCBB4_REGF	CSRAM4F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTZIC2F	CTZSC2F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEXTIF	CLPDMA1F	CRCCF	CPWRF	CTAMPF	CRTCF	CSYSCFGF
	リセット値							0	0									0	0								0	0	0	0	0	0	0

表 30：GTZC2サブブロックのアドレスオフセットを参照してください。

5.11 GTZC2 MPCBB4レジスタ

すべてのレジスタは、必ずワード（32bit）単位でアクセスされます。

5.11.1 GTZC2 SRAM4 MPCBB制御レジスタ（GTZC2_MPCBB4_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRWILADIS	INVSECSTATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLOCK
															rs

ビット31 SRWILADIS：セキュア読出し/書込み不正アクセス無効化

このビットは、セキュア読出し/書込みトランザクションがブロックベースSRAMの非セキュアブロックにアクセスするとき、不正アクセスの検出を無効にします（非セキュアブロックにおけるセキュアフェッチは常に不正とみなされます）。

0：有効、非セキュアSRAMブロックへのセキュア読出し/書込みアクセスは不可

1：無効、非セキュアSRAMブロックへのセキュア読出し/書込みアクセスは可

ビット30 INVSECSTATE：SRAMxクロックのセキュリティ状態

このビットはRCCの内蔵SRAMのクロック制御がセキュアかどうかを定義するために使用されます。

0：MPCBBにセキュア領域が存在する場合、SRAMのクロックはセキュアです。セキュア領域がない場合は非セキュアです。

1：SRAMのクロックは、MPCBBにセキュア領域が存在しても非セキュアで、MPCBBにセキュアブロックが設定されていない場合でもセキュアです。

ビット29:1 予約済みであり、リセット値に保持する必要があります。

ビット0 GLOCK：MPCBBの制御レジスタを次のリセットまでロック

このビットはデフォルトではクリアされており、一旦、設定すると、システムリセットまでリセットできません。

0：制御レジスタロック解除

1：制御レジスタロック

5.11.2 GTZC2 SRAM4 MPCBB設定ロックレジスタ1（GTZC2_MPCBB4_CFGLOCKR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

セキュア特権アクセスのみ。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPLCK0
															rs

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SPLCK0**：スーパーブロック0のセキュリティ /特権設定ロック
このビットはソフトウェアでセットされ、システムリセットでのみクリアが可能です。
0：GTZC2_MPCBB4_SECCFGR0およびGTZC2_MPCBB4_PRIVCFGR0は書込み可能です。
1：GTZC2_MPCBB4_SECCFGR0およびGTZC1_MPCBB4_PRIVCFGR0への書込みは無視されます。

5.11.3 スーパーブロック0のGTZC2 SRAM4 MPCBBセキュリティ設定レジスタ（GTZC2_MPCBB4_SECCFGR0）

アドレスオフセット：0x100

リセット値：0xFFFF FFFF

指定のリセット値はTZEN = 1のときに有効です。TZEN = 0のときのリセット値は0x0000 0000です。

このレジスタへの書込みアクセスはセキュア·アクセスのみです。あらゆる読込みが許可されています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEC31	SEC30	SEC29	SEC28	SEC27	SEC26	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SECy**：スーパーブロック0に属するブロックyのセキュリティ設定（y = 31から0）
0：スーパーブロック0に属するブロックyのみへの非セキュアアクセス。セキュアアクセスは、GTZC2_MPCBB4_CRにSRWILADISビットがセットされている場合も許可されます。
1：スーパーブロック0に属するブロックyのみへのセキュアアクセス。
GTZC2_MPCBB4_PRIVCFGR0のPRIVyビットが設定されている場合、このビットへの非特権書込みは無視されます。
GTZC2_MPCBB4_CFGLOCKR1のSPLCK0ビットが設定されている場合、書込みは無視されます。

5.11.4 スーパーブロック0のGTZC2 SRAM4 MPCBB特権設定レジスタ（GTZC2_MPCBB4_PRIVCFGR0）

アドレスオフセット：0x200

リセット値：0xFFFF FFFF

指定のリセット値はTZEN = 1のときに有効です。TZEN = 0のときのリセット値は0x0000 0000です。

このレジスタへの書込みアクセスは特権アクセスのみです。あらゆる読込みが許されています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRIV31	PRIV30	PRIV29	PRIV28	PRIV27	PRIV26	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRIVy**：スーパーブロック0に属するブロックyの特権設定（y = 31から0）。
0：スーパーブロック0に属するブロックyへの特権および非特権アクセス
1：スーパーブロック0に属するブロックyのみへの特権アクセス。
GTZC2_MPBCC4_SECCFGR0のSECyビットが設定されている場合、このビットへの非セキュア書込みは無視されます。
GTZC2_MPCBB4_CFGLOCKR1のSPLCK0ビットが設定されている場合、書込みは無視されます。

5.11.5 GTZC2 MPCBB4レジスタマップ

表44.GTZC2 MPCBB4レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GTZC2_MPCBB4_CR	SRWILADIS	INVSECSTATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLOCK
	リセット値	0	0																														0
0x004 - 0x00C	予約済み	予約済み																															
0x010	GTZC2_MPCBB4_CFGLOCKR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPLCK0
	リセット値																																0
0x014 - 0x0FC	予約済み	予約済み																															
0x100	GTZC2_MPCBB4_SECCFGR0	SEC31	SEC30	SEC29	SEC28	SEC27	SEC26	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16	SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x104 - 0x1FC	予約済み	予約済み																															
0x200	GTZC2_MPCBB4_PRIVCFGR0	PRIV31	PRIV30	PRIV29	PRIV28	PRIV27	PRIV26	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16	PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 30：GTZC2サブブロックのアドレスオフセットを参照してください。

6 RAM設定コントローラ（RAMCFG）

6.1 RAMCFGの概要

RAMCFGは、内部SRAM（SRAM1/2/3/4/5/6およびBKPSRAM）の機能を設定します。

6.2 RAMCFGの主な機能

内部SRAMは、RAMCFGで設定される下記の機能の一部をサポートします。

- エラーコード訂正（ECC）：
  - 割込み生成に対応のシングルエラー検出と訂正
  - 割込みまたはNMI生成に対応のダブルエラー検出
  - 失敗アドレスでのステータス
- 書込み保護（1KB単位）
- 電圧スケーリング範囲4のプログラム可能なウェイトステート
- SRAMソフトウェア消去

6.3 RAMCFGの機能詳細

6.3.1 内部SRAM機能

最大7つのSRAMがデバイスに内蔵されており、それぞれに特有の機能があります。

- SRAM1/2/3/5/6はメインのSRAMです。SRAM4は、STOP2モードのペリフェラル低消費電力バックグラウンド自律モード（LPBAM）に使用されるSRAMに含まれます。

表45.SRAMの構造

SRAM	STM32U535/545	STM32U575/585	STM32U59x/5Ax	STM32U5Fx/5Gx
SRAM1	192KB（64KBのブロック3個）		768KB（64KBのブロック12個）	
SRAM2	64KB（8KBと56KBのブロックで、STANDBYモードで保持可能）			
SRAM3	N/A	512KB （64KBのブロック8個）	832KB（64KBのブロック13個）	
SRAM4	16KB			
SRAM5	N/A	N/A	832KB（64KBのブロック13個）	
SRAM6	N/A	N/A	N/A	512KB （64KBのブロック8個）
BKPSRAM	2KB			

バックアップSRAM（BKPSRAM）は、すべての低電力モードとV_{BAT}モードでV_{DD}がオフのときに保持できます（詳細については、セクション 10：電源制御（PWR）を参照）。

- 内部SRAMは、レベル0.5またはレベル0への読出し保護（RDP）レベル回帰の場合、ハードウェアによって消去されます。詳細については、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。

- Flash メモリユーザオプションバイトで SRAM2_RST オプションビットが選択されている場合、SRAM2はシステムリセットが発生したときに消去されます。Flashメモリユーザオプションバイトで SRAM_RSTオプションビットが選択されている場合、SRAM1/3/4/5/6はシステムリセットが発生したときに消去されます。詳細については、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。
- SRAM2と、オプションでBKPSRAMは、タンパ検出回路によって保護され、タンパを検出した場合にハードウェアによって消去されます。BKPSRAMもバックアップドメインのリセット時にハードウェアによって消去されます。詳細については、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照してください。
- RAMCFGには、内部SRAM ECC、書込み保護、ウェイトステート設定、およびソフトウェア消去に関するレジスタが内蔵されています。

次の表に、各内部SRAMによってサポートされる機能の概要を示します。

表46.内部SRAM機能

SRAM機能	SRAM1	SRAM2	SRAM3	SRAM4	SRAM5	SRAM6	BKPSRAM
STOP0/1モードでのDMAのアクセス性	X	X	X	X	X	X	X
STOP2モードでのDMAのアクセス性	-	-	-	X	-	-	-
STANDBYモードでの保持（オプション）	-	X	-	-	-	-	X
V_{BAT}モードでの保持（オプション）	-	-	-	-	-	-	X
RDP回帰による消去	X	X	X	X	X	X	X
タンパ検出による消去またはブロック	-	X	-	-	-	-	X(1)
オプションでシステムリセットにより消去	X	X	X	X	X	X	-
ソフトウェア消去	X	X	X	X	X	X	X
ECC	-	X	X	-	-	-	X
書込み保護機能	-	X	-	-	-	-	-
ウェイトステート	X	X	X	X	X	X	X

1. オプション：BKPSRAMは、タンパ検出時に消去するかどうかを設定できます。

6.3.2 エラーコード訂正（SRAM2、SRAM3、BKPSRAM）

ECCは、SRAM2_ECC、SRAM3_ECC、およびBKPRAM_ECCユーザオプションビットで有効なとき、SRAM2/3およびBKPSRAMによってサポートされます。詳細については、セクション 7：内蔵Flashメモリ（FLASH）を参照してください。

7つのECCビットがSRAMの32bitごとに追加されることで、メモリ読出しアクセス時の2bitのエラー検出と1bitのエラー訂正が可能になります。

ECCは32bitワードについて計算され、チェックされるため、バイトおよびハーフワード書込みアクセスはSRAMインタフェースによって、ワード全体を読み出してから、ワードを新しいバイト/ハーフワード値で再び書き込むことによって管理されます。ECCダブルエラーも、このようなバイトまたはハードワードAHB書込みアクセス中に検出されます（インタフェースによる読出し/変更/書込み）。バイトまたはハーフワード書込みアクセス遅延のクロックサイクルはWSC[2:0] + 2 AHBです（セクション6.3.4を参照）。

注意： **ECC付きSRAMへのバイトまたはハーフワード書込みの場合、読出し/変更/書込み操作はバッファで行われます。SRAM AHBがリリースされてから2AHBクロックサイクル後（SRAMがアクセスされなくなってから）、バッファの内容がSRAMに書き込まれます。**

シングルおよびダブルECCエラー

シングルエラーが検出されると、自動的に訂正され、SEDC/CSEDCビットがそれぞれRAMCFG_MxISRおよびRAMCFG_MxICRでセットされます。RAMCFG_MxIERのSEIEによって有効にされている場合は、割込みが生成されます。RAMCFG_MxCRのALEビットがセットされている場合、失敗したアドレスはRAMCFG_MxSEARに格納されます。

注意： **SEDCビットがセットされているときには、シングルエラーを検出できません。**

ダブルエラーが検出されると、DED と CDED ビットがそれぞれ RAMCFG_MxISR および RAMCFG_MxICRにセットされます。RAMCFG_MxIERのDEIEまたはECCNMIによって有効にされている場合は、割込みまたはNMIが生成されます。RAMCFG_MxCRのALEビットがセットされている場合、失敗したアドレスはRAMCFG_MxDEARに格納されます。

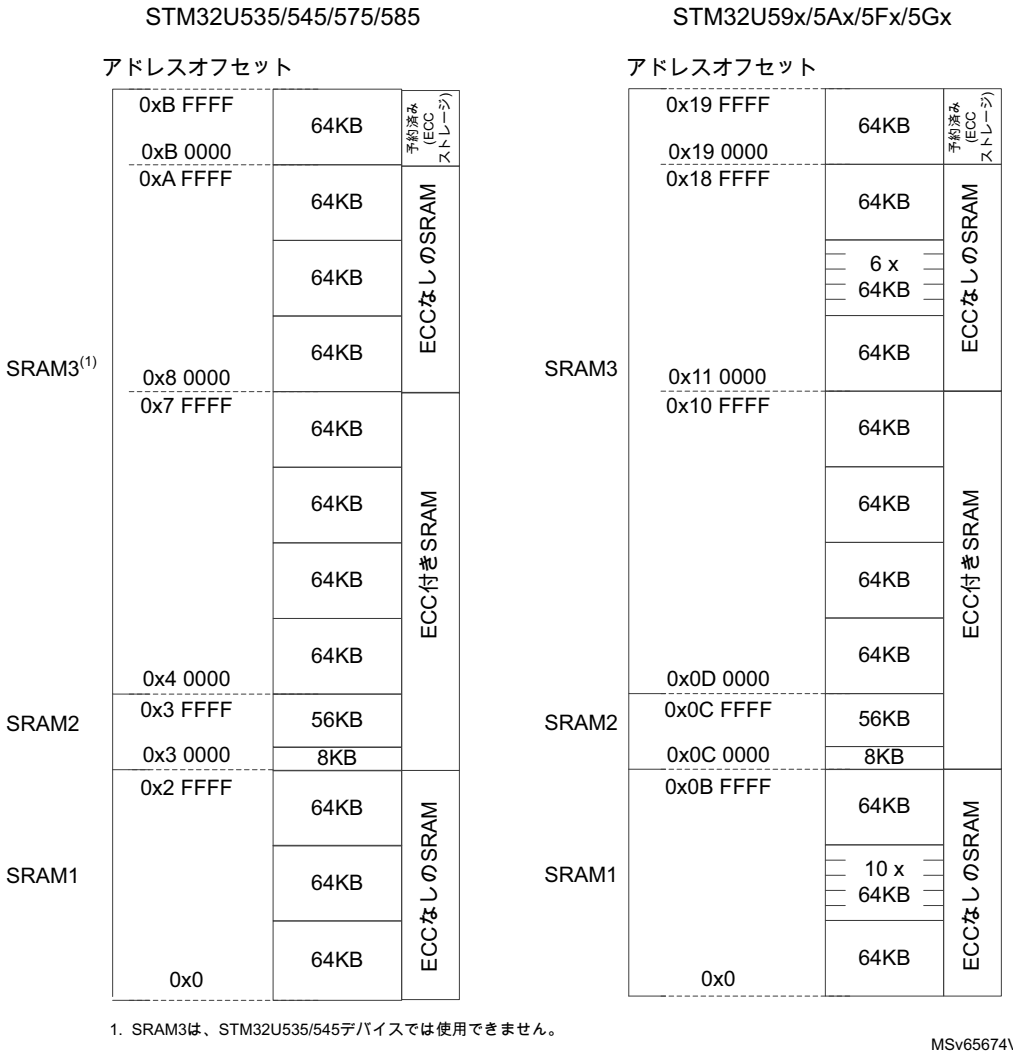
注意： **DEDビットがセットされているとき、ダブルエラーを検出できません。**

SRAM3 ECC固有の管理

SRAM3のECCが有効な場合、ECC付きはSRAM3の最初の256KBのみです。次のSTM32U575/585の192KBまたはSTM32U59x/5Ax/5Fx/5Gxの512KBはECCなしで、最後のブロックはECCの格納に使用されるため、アプリケーションには使用できません。

下の図は、SRAM2とSRAM3のECCが有効なときのSRAM領域を示しています。

図22. SRAM1、ECC付きのSRAM2、およびECC付きのSRAM3メモリマップ



ユーザオプションビットによって ECC が有効化されると、システムリセット後、関連するRAMCFG_MxCRでECCEビットが自動的にセットされます。

ECCは、以下のソフトウェアシーケンスを実行することで無効化できます。

1. RAMCFG_MxECCKEYRに0xAEを書き込みます。
2. RAMCFG_MxECCKEYRに0x75を書き込みます。
3. RAMCFG_MxCRのECCEビットに0を書き込みます。

ECCが無効な場合（ECCE = 0）、SRAM3 ECCストレージ領域（オフセット0xB0000からオフセット0xBFFFF）は他のSRAM3領域と同様に読み書きできます。ECCのメカニズムをテストするには、SRAM3の最初の256KBのみを1または2bitのワード単位で変更できます（それぞれシングルエラーまたはダブルエラーのテスト用）。

ECCの確認手順は次のとおりです。

1. 消去されたメモリに、ECC付きのデータを書き込みます。
2. ECCを無効にします。
3. 同じデータに1または2bitの変更を加えて書き込みます（それぞれシングルエラーまたはダブルエラーのテスト用）。
4. ECCを有効化します。
5. RAMCFG_M3CRのECCEビットが1として読み出されるまで待ちます。
6. リードデータ。シングルエラーまたはダブルエラーにより、有効化された割込みが生成されます。

手順4および5の命令は、ECC領域のあるSRAM3に配置しないでください。他のマスタによるECC領域のあるSRAM3へのアクセスは、手順4および5の実行が完了するまで禁止されています。

警告： ECCフォールト･インジェクション･テストは、SYSCFG_CFGR2のSPLビットがセットされている場合、TIM1/8/15/16/17でシステムブレークイベントをトリガします。
これは、タイマのPWM出力がアイドル状態の間にテストを行う必要があることを意味しています。

6.3.3 書込み保護（SRAM2）

SRAM2は、64の1KBページで構成されます。1KBページは、RAMCFG_M2WPR1およびRAMCFG_M2WPR2の対応するPxWP（x = 0 ～ 63）ビットをセットすることによって、それぞれ書込み保護できます。

6.3.4 読出しアクセスの遅延

データをSRAMから正しく読み出すには、次の表に示すように、AHBクロック周波数（HCLK）と電圧スケーリング範囲に従って、RAMCFG_MxCRのWSC[2:0]フィールドにウェイトステートの数を正しくプログラムする必要があります。

表47.HCLK周波数と電圧スケーリング範囲に対するウェイトステートの数

ウェイトステート（WS）（遅延）	HCLK (MHz)			
	V_{CORE}レンジ1	V_{CORE}レンジ2	V_{CORE}レンジ3	V_{CORE}レンジ4および STOP0/1/2モード(1)
0WS（1AHBサイクル）	≤ 160	≤ 110	≤ 55	≤ 16
1WS（2AHBサイクル）	-	-	-	≤ 25

1. SRAMへのDMA転送を実行するために、システムクロックをSTOP0/1/2モードでリクエストできます。

6.3.5 ソフトウェア消去

SRAM消去は、次のソフトウェアシーケンスを実行することでリクエストできます。

1. RAMCFG_MxERKEYRに0xCAを書き込みます。
2. RAMCFG_MxERKEYRに0x53を書き込みます。
3. RAMCFG_MxCRに1を書き込みます。

消去が進行中の間だけ、関連するSRAM割込みステータスレジスタのSRAMBUSYフラグがセットされます。

各SRAM消去の合計時間は、N AHBクロックサイクルであり、NはSRAMのサイズ（32bitワード数）です。

消去中にSRAMが読出しまたは書込みが行われた場合、消去操作の終了まで、AHBにウェイトステートが挿入されます。

6.4 RAMCFG低電力モード

表48.低消費電力モードがRAMCFGに与える影響

モード	説明
SLEEP	影響はありません。RAMCFG割込みによって、デバイスはSLEEPモードから復帰します。
STOP	RAMCFGレジスタの内容が保持されます。ECCが機能しており、デバイスはECCエラー割込みまたはNMIによってSTOP0およびSTOP1モードから復帰します。
STANDBY	RAMCFGペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

6.5 RAMCFG割込み

次の表に、RAMCFG割込みリクエストの一覧を示します。

表49.RAMCFG割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
RAMCFG	ECCシングルエラー検出および訂正	SEDC	SEIE	CSEDCに1を書き込む	あり	あり(1)	なし
	ECCダブルエラー検出	DED	DEIE = 1かつECCNMI = 0	CDEDに1を書き込む	あり	あり（1）	なし
NMI	ECCダブルエラー検出	DED	ECCNMI	CDEDに1を書き込む	あり	あり（1）	なし

1. STOP0およびSTOP1モードのみ。

6.6 RAMCFGレジスタ

下記のレジスタでは、xは以下を指します。

- x = 1/2/3/4のとき、それぞれSRAM1/2/3/4
- x = 5のとき、BKPSRAM
- x = 6のとき、SRAM5
- x = 7のとき、SRAM6

6.6.1 RAMCFGメモリx制御レジスタ(RAMCFG_MxCR)

アドレスオフセット：0x040 *（x - 1）、（x = 1 ~ 7）

リセット値：0x0000 000X

ECCEリセット値はECCイネーブルユーザオプションビットに依存します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
							rs				rw				rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **WSC[2:0]**：ウェイトステート設定

このフィールドは、アクセス時間に従って、SRAMの読出し時にAHBに挿入されたウェイトステートの数をプログラムするために使用されます。

000：0ウェイトステート

001：1ウェイトステート

...

111：7ウェイトステート（不要）

ビット15:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SRAMER**：SRAMの消去

このビットは、RAMCFG_MxERKEYRレジスタのERASEKEYフィールドにアンロックシーケンスを書き込んだ後でのみ、ソフトウェアによってセットできます。このビットをセットすると、SRAM消去が開始されます。このビットは、消去操作の終了時にハードウェアによって自動的にクリアされます。

0：進行中の消去操作はありません。

1：進行中の消去操作があります。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ALE**：アドレスラッチイネーブル

0：失敗したアドレスはSRAMxECCシングル/ダブルエラーアドレスレジスタに保存されません。

1：失敗したアドレスはSRAMxECCシングル/ダブルエラーアドレスレジスタに保存されます。

注： このビットは予約済みであり、SRAM1/4/5/6制御レジスタでリセット値に保持する必要があります。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **ECCE**：ECCイネーブル

このビットのリセット値は、ユーザオプションビット設定によって定義されます。セットされているときには、RAMCFG_MxECCKEYRレジスタにアンロックシーケンスを書き込んだ後でのみ、ソフトウェアによってクリアできます。

0：ECCは無効です。

1：ECCは有効です。

注： このビットは予約済みであり、SRAM1/4/5/6制御レジスタでリセット値に保持する必要があります。

6.6.2 RAMCFGメモリx割込みイネーブルレジスタ（RAMCFG_MxIER）

アドレスオフセット：0x004 + 0x40 * (x - 1), (x = 2, 3, 5)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCNMI	Res.	DEIE	SEIE
												rs		rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **ECCNMI**：ダブルエラー NMI

このビットは、ソフトウェアによってセットされ、グローバルRAMCFGリセットによってのみクリアされます。

0：ECCダブルエラー発生時、NMIは生成されません。

1：ECCダブルエラー発生時、NMIが生成されます。

注： ECCNMIがセットされた場合、DEIEビット値に関係なく、RAMCFGマスク可能割込みは生成されません。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DEIE**：ECCダブルエラー割込みイネーブル

0：ダブルエラー割込みは無効です。

1：ダブルエラー割込みは有効です。

ビット0 **SEIE**：ECCシングルエラー割込みイネーブル

0：シングルエラー割込みは無効です。

1：シングルエラー割込みは有効です。

6.6.3 RAMCFGメモリ割込みステータスレジスタ(RAMCFG_MxISR)

アドレスオフセット：0x008 + 0x40 * (x-1)(x = 1から7)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAM BUSY	Res.	Res.	Res.	Res.	Res.	Res.	DED	SEDC
							r							r	r

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SRAMBUSY**：消去操作によるSRAMビジー

0：進行中の消去操作はありません。

1：進行中の消去操作があります。

注： SRAMに応じて、消去操作は、ソフトウェアリクエスト、オプションビットが有効な場合はシステムリセット、タンパ検出、または読出し保護回帰によって行うことができます（表46を参照）。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DED**：ECCダブルエラー検出

0：ダブルエラーはありません。

1：ダブルエラーが検出されています。

注： このビットは予約済みであり、SRAM1/4/5/6 割込みステータスレジスタでリセット値に保持する必要があります。

ビット0 **SEDC**：ECCシングルエラー検出および訂正

0：シングルエラーはありません。

1：シングルエラーは検出も訂正もされていません。

注： このビットは予約済みであり、SRAM1/4/5/6 割込みステータスレジスタでリセット値に保持する必要があります。

6.6.4 RAMCFGメモリxECCシングルエラーアドレスレジスタ（RAMCFG_MxSEAR）

アドレスオフセット：0x00C + 0x40 * (x - 1), (x = 2, 3, 5)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ESEA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ESEA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **ESEA[31:0]**：ECCシングルエラーアドレス

RAMCFG_MxCRレジスタのALEビットがセットされたとき、このフィールドはECCシングルエラーに対応するアドレスで更新されます。

6.6.5 RAMCFGメモリxECCダブルエラーアドレスレジスタ（RAMCFG_MxDEAR）

アドレスオフセット：0x010 + 0x40 * (x - 1), (x = 2, 3, 5)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EDEA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EDEA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **EDEA[31:0]**：ECCダブルエラーアドレス

RAMCFG_MxCRレジスタのALEビットがセットされているとき、このフィールドはECCダブルエラーに対応するアドレスで更新されます。

6.6.6 RAMCFGメモリx割込みクリアレジスタx(RAMCFG_MxICR)

アドレスオフセット：0x014 + 0x40 * (x - 1), (x = 2, 3, 5)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDED	CSEDC
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CDED**：検出されたECCダブルエラーのクリア

このフラグに1を書き込むことで、RAMCFG_MxISRレジスタのDEDビットがクリアされます。このフラグを読み出すと、DED値が返されます。

ビット0 **CSEDC**：検出および訂正されたECCシングルエラーのクリア

このフラグに1を書き込むと、RAMCFG_MxISRレジスタのSEDCビットがクリアされます。このフラグを読み出すと、SEDC値が返されます。

6.6.7 RAMCFGメモリ2書込み保護レジスタ1（RAMCFG_M2WPR1）

アドレスオフセット：0x058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット31:0 **PyWP**：SRAM2 1KBページy書込み保護（y = 31 ~ 0）

これらのビットは、ソフトウェアによってセットされ、グローバルRAMCFGリセットによってのみクリアされます。

0：SRAM2 1KBページyの書込み保護は無効です。

1：SRAM2 1KBページyの書込み保護は有効です。

6.6.8 RAMCFGメモリ2書込み保護レジスタ2（RAMCFG_M2WPR2）

アドレスオフセット：0x05C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P63WP	P62WP	P61WP	P60WP	P59WP	P58WP	P57WP	P56WP	P55WP	P54WP	P53WP	P52WP	P51WP	P50WP	P49WP	P48WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P47WP	P46WP	P45WP	P44WP	P43WP	P42WP	P41WP	P40WP	P39WP	P38WP	P37WP	P36WP	P35WP	P34WP	P33WP	P32WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット31:0 **PyWP**：SRAM2 1Kbページy書込み保護（y（y = 63 ~ 32）

これらのビットは、ソフトウェアによってセットされ、グローバルRAMCFGリセットによってのみクリアされます。

0：SRAM2 1KBページyの書込み保護は無効です。

1：SRAM2 1KBページyの書込み保護は有効です。

6.6.9 RAMCFGメモリxECCキーレジスタ（RAMCFG_MxECCKEYR）

アドレスオフセット：0x024 + 0x40 * (x - 1), (x = 2, 3, 5)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCKEY[7:0]							
								w	w	w	w	w	w	w	w

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **ECCKEY[7:0]**：ECC書込み保護キー

RAMCFG_MxCRのECCEの書込み保護をアンロックするには、以下の手順が必要です。

1) ECCKEY[7:0]に0xAEを書き込みます。

2) ECCKEY[7:0]に0x75を書き込みます。

注： 誤ったキーを書き込むと、書込み保護が再度アクティブになります。

6.6.10 RAMCFGメモリx消去キーレジスタ（RAMCFG_MxERKEYR）

アドレスオフセット：0x028 + 0x40 * (x-1)(x = 1から7)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
								w	w	w	w	w	w	w	w

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **ERASEKEY[7:0]**：ECC書込み保護キーの消去

RAMCFG_MxCRのSRAMERの書込み保護をアンロックするには、以下の手順が必要です。

1) ERASEKEY[7:0]に0xCAを書き込みます。

2) ERASEKEY[7:0]に0x53を書き込みます。

注： 誤ったキーを書き込むと、書込み保護が再度アクティブになります。

6.6.11 RAMCFGレジスタマップ

表50.RAMCFGレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RAMCFG_M1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x04	予約済み	予約済み																															
0x08	RAMCFG_M1ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								0								
0x0Cから0x24	予約済み	予約済み																															
0x28	RAMCFG_M1ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x2Cから0x3C	予約済み	予約済み																															
0x40	RAMCFG_M2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x44	RAMCFG_M2IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCNMI	Res.	DEIE	SEIE
	リセット値																													0		0	0
0x048	RAMCFG_M2ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	DED	SEDC
	リセット値																								0							0	0
0x04C	RAMCFG_M2SEAR	ESEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	RAMCFG_M2DEAR	EDEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	RAMCFG_M2ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDED	CSEDC
	リセット値																															0	0
0x058	RAMCFG_M2WPR1	P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP	P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	RAMCFG_M2WPR2	P63WP	P62WP	P61WP	P60WP	P59WP	P58WP	P57WP	P56WP	P55WP	P54WP	P53WP	P52WP	P51WP	P50WP	P49WP	P48WP	P47WP	P46WP	P45WP	P44WP	P43WP	P42WP	P41WP	P40WP	P39WP	P38WP	P37WP	P36WP	P35WP	P34WP	P33WP	P32WP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	予約済み	予約済み																															
0x064	RAMCFG_M2ECCKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x068	RAMCFG_M2ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x06Cから0x07C	予約済み	予約済み																															

表50.RAMCFGレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080	RAMCFG_M3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				0
0x084	RAMCFG_M3IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCNMI	Res.	DEIE	SEIE
	リセット値																													0		0	0
0x088	RAMCFG_M3ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	DED	SEDC
	リセット値																								0							0	0
0x08C	RAMCFG_M3SEAR	ESEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x090	RAMCFG_M3DEAR	EDEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094	RAMCFG_M3ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDED	CSEDC
	リセット値																															0	0
0x098 ~0x0A0	予約済み	予約済み																															
0x0A4	RAMCFG_ M3ECCKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0A8	RAMCFG_ M3ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0AC - 0x0BC	予約済み	予約済み																															
0x0C0	RAMCFG_M4CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x0C4	予約済み	予約済み																															
0x0C8	RAMCFG_M4ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								0								
0x0CC から 0x0E4	予約済み	予約済み																															
0x0E8	RAMCFG_ M4ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[2:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0EC - 0x0FC	予約済み	予約済み																															
0x100	RAMCFG_M5CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x104	RAMCFG_M5IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCNMI	Res.	DEIE	SEIE
	リセット値																													0		0	0

表50.RAMCFGレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x108	RAMCFG_M5ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	DED	SEDC
	リセット値																								0							0	0
0x10C	RAMCFG_M5SEAR	ESEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x110	RAMCFG_M5DEAR	EDEA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x114	RAMCFG_M5ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDED	CSEDC
	リセット値																															0	0
0x118 - 0x120	予約済み	予約済み																															
0x124	RAMCFG_M5ECCKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x128	RAMCFG_M5ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x12C から 0x13C	予約済み	予約済み																															
0x140	RAMCFG_M6CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x144	予約済み	予約済み																															
0x148	RAMCFG_M6ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								0								
0x14C から 0x164	予約済み	予約済み																															
0x168	RAMCFG_M6ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x180	RAMCFG_M7CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WSC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMER	Res.	Res.	Res.	ALE	Res.	Res.	Res.	ECCE
	リセット値														0	0	0								0				0				x
0x184	予約済み	予約済み																															
0x188	RAMCFG_M7ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAMBUSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								0								
0x18C から 0x1A4	予約済み	予約済み																															
0x1A8	RAMCFG_M7ERKEYR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERASEKEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

7 内蔵Flashメモリ（FLASH）

7.1 FLASHの概要

Flashメモリインタフェースは、CPU、命令キャッシュ、およびDMAへのスループットを最大化して、Flashメモリへのアクセスを管理します。Flashメモリの消去操作とプログラム操作、読出しおよび書込み保護メカニズムを実装します。また、セキュリティと特権のアクセス制御機能も実装します。MCUが低電力モードの場合に、専用のモードによって消費電力を最適化します。

7.2 Flashの主な機能

- 書込み中読出し（RWW）機能に対応する最大容量4MBのFlashメモリ。
- メモリ構成
  - デュアルバンクアーキテクチャ（バンク1およびバンク2）
  - メインメモリ：バンク当たり最大2MB
  - 情報ブロック：バンク1で64.5KB
- プリフェッチによる128bit幅のデータ読出し
- 標準およびバースト･プログラミング･モード
- すべての電圧範囲における読出し、プログラム、および消去操作
- すべてのFlashメモリにおける10Kサイクルの耐性。バンク当たり最大256KBで100Kサイクル
- ページ消去、バンク消去、全体消去（両バンク）を実装
- バンクスワッピング：各バンクのユーザFlashメモリのアドレスマッピングをスワップ可能。
- TrustZoneのオプションビット（TZEN）によって有効化される製品セキュリティ
- 読出し保護（RDP）のオプションバイトによって管理されるデバイスのライフサイクル
- 4つの書込み保護領域（バンク当たり2つ）
- TrustZoneサポート：
  - 2つのセキュア領域（バンク当たり1つ）
  - セキュア領域の2つのセキュアHDP（非表示保護）領域（バンク当たり1つ）
- Flashページ単位の非特権アクセスに対する設定可能な保護
- エラーコード訂正：2bitのエラー検出と1bitのエラー訂正を可能にする、128bitのクワッドワードごとに9bitのECC
- オプションバイトローダ
- 高度な低電力モード（低電力読出しモード、バンク･パワーダウン･モード）

7.3 Flashの機能説明

7.3.1 Flashメモリの構成

Flashメモリは、以下のような主な特徴を備えています。

- 最大容量4MB
- デュアルバンクモード：
  - メインメモリでは、バンク当たり最大2MB
  - ページサイズ8KB
  - 137Bit幅のデータ読出しおよび書込み（128有効bit + ECC9bit）
  - ページ、バンク、および全体消去
  - 書込み中読出し機能に対応
  - 消費電力の削減のためのバンクのシャットダウン機能に対応

Flashメモリは、次のように構成されています。

- それぞれが8KBの最大256ページを含む、最大2MBの2つのバンクとして構成されるメインメモリブロック
- 情報ブロックには次のものが含まれます。
  - システムメモリ用の32KB。この領域は不変で、STMicroelectronics用に予約されています。USB（DFU）などのユーザ通信インタフェースの1つを通じてFlashメモリの再プログラムに使用される、ブートローダが含まれます。システムメモリは、デバイス製造時にSTMicroelectronicsによってプログラムされています。詳細については、アプリケーションノート「STM32 microcontroller system memory boot mode (AN2606)」を参照してください。
  - STMicroelectronicsによって開発されたルートセキュリティサービス（RSSおよびRSSライブラリ）を含む32KBの不変セキュア領域
  - ユーザデータ用の512バイトOTP（1度だけプログラムが可能な）（32クワッドワード）。OTPデータは消去不可で、1度だけ書込み可能です。
  - ユーザ設定のオプションバイト。ユーザFlashメモリやシステムメモリとは異なり、特定のメモリアドレスにマッピングされているわけではなく、Flashレジスタインタフェースを通じてのみアクセスすることができます。

メモリ構成は、次の表に示される主な領域と情報ブロックに基づきます。

表51.STM32U535/545向けのFlashモジュール512KBデュアルバンク構成(1)

Flash領域		Flashメモリアドレス	サイズ	名前
メインメモリ	バンク1	0x0800 0000 - 0x0800 1FFF	8KB	0ページ
		0x0800 2000 - 0x0800 3FFF	8KB	ページ1
		...	...	...
		0x0803 C000 - 0x803 DFFF	8KB	ページ30
		0x0803 E000 - 0x803 FFFF	8KB	ページ31
	バンク2	0x0804 0000 - 0x0804 3FFF	8KB	0ページ
		0x00804 2000 - 0x0804 3FFF	8KB	ページ1
		...	...	...
		0x0807 C000 - 0x0807 DFFF	8KB	ページ30
		0x0807 E000 - 0x0807 FFFF	8KB	ページ31

表51.STM32U535/545向けのFlashモジュール512KBデュアルバンク構成(1)

Flash領域	Flashメモリアドレス	サイズ	名前
非セキュア情報ブロック	0x0BF9 0000 - 0x0BF9 7FFF	32KB	システムメモリ
	0x0BFA 0000 - 0x0BFA 01FF	512バイト	OTPエリア
セキュア情報ブロック	0x0FF8 0000 - 0x0FF8 5FFF	24KB	RSS
	0x0FF8 6000 - 0x0FF8 7FFF	8KB	RSSライブラリ
	0x0FFA 0000 - 0x0FFA 01FF	512バイト	OTPエリアのエイリアス

1. デュアルバンクSTM32U535/545デバイスに対して、オプションバイトでDUALBANK = 1のとき、バンク2のベースアドレスは256KBFlashの場合0x0802 0000で、128KBFlashの場合、0x0801 0000です。

表52.STM32U575/585向けのFlashモジュール2MBデュアルバンク構成(1)

Flash領域		Flashメモリアドレス	サイズ	名前
メインメモリ	バンク1	0x0800 0000 - 0x0800 1FFF	8KB	0ページ
		0x0800 2000 - 0x0800 3FFF	8KB	ページ1
		...	...	...
		0x080F E000 - 0x080F FFFF	8KB	ページ127
	バンク2	0x0810 0000 - 0x0810 1FFF	8KB	0ページ
		0x0810 2000 - 0x0810 3FFF	8KB	ページ1
		...	...	...
		0x081F E000 - 0x081F FFFF	8KB	ページ127
非セキュア情報ブロック		0x0BF9 0000 - 0x0BF9 7FFF	32KB	システムメモリ
		0x0BFA 0000 - 0x0BFA 01FF	512バイト	OTPエリア
セキュア情報ブロック		0x0FF8 0000 - 0x0FF8 5FFF	24KB	RSS
		0x0FF8 6000 - 0x0FF8 7FFF	8KB	RSSライブラリ
		0x0FFA 0000 - 0x0FFA 01FF	512バイト	OTPエリアのエイリアス

1. 1MBのデュアルバンクSTM32U575/585デバイスに対して、オプションバイトでDUALBANK = 1のとき、バンク2のベースアドレスは0x0808 0000です。

表53.STM32U59x/5Ax/5Fx/5Gx向けのFlashモジュール4MBデュアルバンク構成(1)

Flash領域		Flashメモリアドレス	サイズ	名前
メインメモリ	バンク1	0x0800 0000 - 0x0800 1FFF	8KB	0ページ
		0x0800 2000 - 0x0800 3FFF	8KB	ページ1
		...	...	...
		0x081F E000 - 0x081F FFFF	8KB	ページ255
	バンク2	0x0820 0000 - 0x0820 1FFF	8KB	0ページ
		0x0820 2000 - 0x0820 3FFF	8KB	ページ1
		...	...	...
		0x083F E000 - 0x083F FFFF	8KB	ページ255

表53.STM32U59x/5Ax/5Fx/5Gx向けのFlashモジュール4MBデュアルバンク構成(1)（続き）

Flash領域	Flashメモリアドレス	サイズ	名前
非セキュア情報ブロック	0x0BF9 0000 - 0x0BF9 7FFF	32KB	システムメモリ
	0x0BFA 0000 - 0x0BFA 01FF	512バイト	OTPエリア
セキュア情報ブロック	0x0FF8 0000 - 0x0FF8 5FFF	24KB	RSS
	0x0FF8 6000 - 0x0FF8 7FFF	8KB	RSSライブラリ
	0x0FFA 0000 - 0x0FFA 01FF	512バイト	OTPエリアのエイリアス

1. 2MBのデュアルバンクSTM32U59x/5Ax/5Fx/5Gxデバイスに対して、オプションバイトでDUALBANK = 1のとき、バンク2のベースアドレスは0x0810 0000です。

注： **セキュア情報ブロックは、TrustZoneがアクセス可能な場合にのみ使用できます。**

7.3.2 エラーコード訂正（ECC）

Flashメモリのデータは137bitワードです。クワッドワード（128bit）単位で9bit追加されます。ECCメカニズムは次のものをサポートしています。

- 1つのエラー検出および訂正
- 2つのエラー検出

1つのエラーを検出して訂正する場合、ECCCフラグ（ECC訂正）がFLASH_ECCRにセットされます。ECCCIEビットがセットされた場合、割込みが生成されます。

2つのエラーが検出されたときに、FLASH_ECCRにECCDフラグ（ECC検出）がセットされます。この場合、NMIが生成されます。

ECCエラーが検出されると、失敗したクワッドワードのアドレスおよびその関連バンクがFLASH_ECCRのADDR_ECC[19:0]とBK_ECCに保存されます。ADDR_ECC[3:0]は常にクリアされます。

ECCCまたはECCDがセットされていると、新しいECCエラーが発生した場合にADDR_ECCおよびBK_ECCが更新されません。ECCフラグがクリアされた場合のみFlash_ECCRが更新されます。

注意： **ECCCフラグがセットされたときに、さらなる2つのエラー検出による、タイマへのNMIまたはブレーク信号を生成することはできません。そのため、訂正が行われたら、ECCエラー検出機能を保持するために、直ちにECCCフラグをクリアすることを推奨します。ダブルECCエラー検出（ECCDフラグがセットされ、NMIがトリガされている）の場合、ソフトウェアはNMIハンドラのキャッシュをクリアする必要があります。安全規格への準拠に及ぼす影響の詳しい説明については、「STM32U5 Series safety manual (UM2875)」を参照してください。**

注： **消去済みのFlashラインについては、1つのエラーを検出して訂正しますが、2つのエラー検出はサポートされません。ECCエラーが報告されると、データが現在のバッファにまだ存在する場合、ECCCおよびECCDがクリアされていても、失敗したアドレスで新しい読出しがECCエラーを生成しない場合があります。**

システムFlashメモリの以下のアドレスは、ECCエラーなどのワードを格納するために使用されます。これは、ECC訂正検出機能でソフトウェアによるランタイム･テストを可能にするためです。

- 0x0BFA1F00（1bitエラーのあるワードを内蔵）
- 0x0BFA1F80（2bitエラーのあるワードを内蔵）

2番目のアドレスが、たとえばデバッガ･メモリ･ビューアによって読み出される場合、NMIが生成されます。

7.3.3 読出しアクセスの遅延

データをFlashメモリから正しく読み出すには、CPUクロック（HCLK）の周波数およびデバイスの内部電圧範囲COREに従って、FLASH_ACRでウェイトステート（遅延）の数を正しくプログラムする必要があります。セクション 10.5.4：ダイナミック電圧スケーリングの管理を参照してください。

次の表には、ウェイトステートとCPUクロック周波数の対応を示します。

表54.CPUクロック（HCLK）周波数によるウェイトステート数（LPM=0）

ウェイトステート（WS）（遅延）	HCLK (MHz)			
	V_{CORE}レンジ1	V_{CORE}レンジ2	V_{CORE}レンジ3	V_{CORE}レンジ4
0WS（1CPUサイクル）	≤ 32	≤ 30	≤ 24	≤ 12
1WS（2CPUサイクル）	≤ 64	≤ 60	≤ 48	≤ 25
2WS（3CPUサイクル）	≤ 96	≤ 90	≤ 55	-
3WS（4CPUサイクル）	≤ 128	≤ 110	-	-
4WS（5CPUサイクル）	≤ 160	-	-	-

Flashメモリは、FLASH_ACRのLPMを設定する際に、低電力読出しモードをサポートしています。次の表に、LPMビットをセットしたときのウェイトステートとCPUクロック周波数の対応を示します。

表55.CPUクロック（HCLK）周波数によるウェイトステート数（LPM = 1）

ウェイトステート（WS）（遅延）	HCLK (MHz)	
	V_{CORE}レンジ1/2/3	V_{CORE}レンジ4
0WS（1CPUサイクル）	WS ≥ HCLK（MHz）/10-1 最大HCLK周波数は、表54から得られます。	≤ 8
1WS（2CPUサイクル）		≤ 16
2WS（3CPUサイクル）		≤ 25
3WS（4CPUサイクル）		-
...		-
15WS（16CPUサイクル）		-

リセット後、CPUクロック周波数は4MHzで、FLASH_ACRで0ウェイトステート（WS）が設定され、通常の読出しモードが選択されます（LPM = 0）。

命令プリフェッチ

Cortex-M33は、Cバスを使って、また有効になっている場合は命令キャッシュを経由して、命令とリテラル·プール（定数/データ）をフェッチします。プリフェッチブロックは、命令キャッシュが有効になっているときに、キャッシュリフィル遅延を減らすことでCバス·アクセスの効率を向上することを目的としています。プリフェッチは、連続コードの場合に効率的です。Flashメモリのプリフェッチにより、現在の命令ラインが命令キャッシュで満たされ、CPUによって実行されている間に、次の連続命令ラインをFlashメモリから読み出せます。

プリフェッチは、FLASH_ACRのPRFTENをセットすると有効になります。PRFTENは、Flashメモリのアクセスに1つ以上のウェイトステートが必要な場合にセットする必要があります。

注： **プリフェッチにより、追加のFlashメモリアクセス時に、コード処理性能が向上する傾向があります。低電力アプリケーションでは慎重に使用する必要があります。**

CPU周波数の変更時、Flashメモリにアクセスするために必要なウェイトステート数を調整するため、以下に示すソフトウェアシーケンスを適用する必要があります。

CPU周波数を大きく

1. FLASH_ACRレジスタのLATENCYビットに新しいウェイトステート数をプログラムします。
2. FLASH_ACRを読み戻してFlashメモリへのアクセスに新しいウェイトステート数が反映されていることを確認します。
3. RCC_CFGR1のWビットを書き込んでCPUクロックソースを変更します。
4. 必要に応じて、RCC_CFGR2のHPREビットを書き込んでCPUクロックプリスケーラを変更します。
5. RCC_CFGR1 および RCC_CFGR2 でクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPREビット）を読み出して新しいCPUクロックソースや新しいCPUクロックプリスケーラの値がそれぞれ考慮されていることを確認します。

CPU周波数を小さく

1. RCC_CFGR1のSWビットを書き込んでCPUクロックソースを変更します。
2. 必要に応じて、RCC_CFGR2のHPREビットを書き込んでCPUクロックプリスケーラを変更します。
3. RCC_CFGR1 および RCC_CFGR2 でクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPREビット）を読み出して新しいCPUクロックソースや新しいCPUクロックプリスケーラの値がそれぞれ考慮されていることを確認します。
4. FLASH_ACRレジスタのLATENCYビットに新しいウェイトステート数をプログラムします。
5. FLASH_ACRを読み戻してFlashメモリへのアクセスに新しいウェイトステート数が使用されていることを確認します。

読出しモードを変更するには、以下に示すソフトウェアシーケンスを適用する必要があります。

通常の読出しモードから低電力読出しモードまで

1. FLASH_ACRのLPMをセットします。
2. 低電力読出しモードは、FLASH_ACRの読出しによって有効化されることを確認してください。

低電力読出しモードから通常の読出しモードまで

1. FLASH_ACRのLPMビットをリセットします。
2. 通常の読出しモードは、FLASH_ACRの読出しによって有効化されることを確認してください。

7.3.4 バンク･パワーダウン･モード

リセット後、バンクは両方とも通常モードになります。消費電力を削減するには、FLASH_ACRのPDREQxをセットすることによって、各バンクを個別にパワーダウンモードにすることができます。

バンクxへのパワーダウンモードでのエントリのリクエスト

- バンク x がパワーダウンモードでないこと、またパワーダウンモードにするリクエストがペンディング状態でないことを確認します（FLASH_NSSRのPDxおよびFLASH_ACRのPDREQxをリセットする必要があります）。
- 正しいキーでPDKEYxRをアンロックします（FLASH_PDKEY1RまたはFLASH_PDKEY2Rを参照）。
- FLASH_ACRにPDREQxをセットします。
- LFASH_NSSR の PDx がセットされ、FLASH_ACR の PDREQx が自動的にリセットされ、PDKEYxRがロックされていることを確認します。

注： **バンクxが現在処理中の場合、パワーダウンのリクエストはアクセスが完了するまで遅延します。**

すでにパワーダウンモードのバンクにパワーダウン·エントリをリクエストしても意味がありません。FLASH_ACRのPDREQxは自動的にリセットされ、PDKEYxRはロックされます。

通常モードに戻る

パワーダウンモードでのバンクへの任意のアクセスにより、バンクは自動的にウェイクアップします。バンクをウェイクアップするには、最低5 µsのペナルティが課せられます。

バンク1（または2）のウェイクアップは、以下のいずれかの場合に行われます。

- バンク1（またはバンク2）への有効な読出しアクセス時
- バンク1（またはバンク2）への有効な書込みアクセス時
- バンク1（またはバンク2）の有効なバンク消去時

バンク1およびバンク2の両方のウェイクアップは、以下のいずれかの場合に行われます。

- 有効な全体消去時
- オプションバイトの変更時
- オプションバイトのロード時
- システムリセット時

注： **ソフトウェアは、バンクをウェイクアップする前にHSI16を有効にすることで、Flashバンクのウェイクアップ時間を短縮できます。**

7.3.5 Flashメモリプログラムおよび消去操作

内蔵Flashメモリは、インサーキットプログラミング（ICP）またはアプリケーション内プログラミング（IPA）を使用してプログラム可能です。

ICP方法は、マイクロコントローラにユーザアプリケーションをロードするために、JTAG、SWDプロトコル、またはブートローダを使用して、Flashメモリの内容全体を更新するために使用されます。ICPは、迅速かつ効率的な設計時の繰り返し操作を提供し、不要なパッケージ処理やデバイスのソケット処理を排除できます。

IAPは、メモリにプログラミングデータをダウンロードするために、マイクロコントローラ（I/O、USB、CAN、UART、I2C、またはSPI）がサポートする任意の通信インタフェースを使用できます。IAPを使用して、ユーザはアプリケーション実行中にFlashメモリを再プログラムできます。アプリケーションの一部は、ICPを使用して事前にFlashメモリにプログラムされていなければなりません。

進行中のFlashメモリ操作は、CPUが同じFlashメモリバンクにアクセスしない限り、CPUをブロックしません。コードまたはデータのフェッチは、1つのバンクで書込み/消去操作を実施している間にもう1つのバンクで可能です（セクション7.3.10を参照してください）。

逆に、Flashメモリへのプログラム/消去操作中に同じFlashメモリバンクを読み出そうとすると、バスがストールします。読出し操作は、プログラム/消去操作が完了すると、正しく処理されます。

MCUは、Flashメモリのセキュアおよび非セキュア領域を定義するTrustZoneをサポートします。すべてのプログラムおよび消去操作をセキュアレジスタによってセキュアモードで、または非セキュアレジスタによって非セキュアモードで行うことができます。詳細については、セクション7.5を参照してください。

セキュア/非セキュアなFLASH制御レジスタのアンロック

リセット後は、考えられる不要な操作（たとえば、電気妨害などによる）からFlashメモリを保護するため、FLASH_SECCRおよびFLASH_NSCRには書き込めません。

これらのレジスタのアンロックには、次のシーケンスを使用します。

1. FLASH_SECKEYRまたはFLASH_NSKEYRにKEY1=0x45670123を書き込みます。
2. FLASH_SECKEYRまたはFLASH_NSKEYRにKEY2 = 0xCDEF89ABを書き込みます。

シーケンスを誤ると、次のシステムリセットまでFLASH_SECCRまたはFLASH_NSCRがロックされます。キーシーケンスを誤ると、バスエラーが検出され、ハードフォールト割込みが発生します。

FLASH_NSCR（またはFLASH_SECCR）は、FLASH_NSCR（またはFLASH_SECCR）のLOCKをセットすることで、ソフトウェアで再びロックできます。

注： **FLASH_NSCR および FLASH_SECCR は、BSY ビットがセットされている場合は書き込めません。BSYビットがセットされている状態で書き込もうとすると、BSYビットがクリアされるまでAHBバスはストールします。**

書込みデータ待ちフラグ（WDW）

FLASH_NSSRおよびFLASH_SECSRのWDWフラグは、書込みバッファでセキュアまたは非セキュア書込みアクセスが行われると両方ともセットされます。これらはBSYフラグがセットされるか（つまり、書込みバッファが開放され、Flashメモリでプログラム操作が実際に開始される）、エラーが発生した場合にクリアされます。

ソフトウェアで、同じクワッドワードの4ワードがすべて書き込まれていることを確認する必要があります。

Flashのセキュア/非セキュアビジーフラグ

FLASH_NSSRおよびFLASH_SECSRのBSYフラグは、セキュアおよび非セキュアFlash操作が開始されたときに、両方ともセットされます。

- 消去操作：FLASH_NSCRまたはFLASH_SECCRのSTRTをセットします。
- 書込み操作：FLASH_NSCRまたはFLASH_SECCRのPGを設定し、Flashメモリのクワッドワードを書き込みます。
- オプションバイトのプログラム操作：FLASH_NSCRのOPTSTRTをセットします。

7.3.6 Flashメインメモリの消去シーケンス

Flashメモリの消去操作は、ページ単位、バンク単位、またはFlashメモリ全体（全体消去）に対して実行できます。全体消去は、情報ブロック（システムFlash、OTP、およびオプションバイト）には影響しません。消去操作はセキュアまたは非セキュアです。

ページ消去

ページを消去するには、次の手順に従います。

1. FLASH_NSSRまたはFLASH_SECSRのBSYを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERRがセットされます。
3. FLASH_NSCRまたはFLASH_SECCRの関連バンク（BKER）でPERビットをセットし、消去するページ（PNB）を選択します。
4. FLASH_NSCRまたはFLASH_SECCRのSTRTをセットします。
5. FLASH_NSSRまたはFLASH_SECSRのBSYがクリアされるのを待ちます。

バンク1またはバンク2の全体消去

バンクの全体消去を実行するには、次の手順に従います。

1. FLASH_NSSRまたはFLASH_SECSRのBSYを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERRがセットされます。
3. FLASH_NSCRまたはFLASH_SECCRのMER1またはMER2ビット（バンクによる）をセットします。どちらのバンクも同じ操作で選択できますが、その場合は全体消去に相当します。
4. FLASH_NSCRまたはFLASH_SECCRのSTRTをセットします。
5. FLASH_NSSRまたはFLASH_SECSRのBSYビットがクリアされるのを待ちます。
6. バンクの消去がそれ以上リクエストされなければ、MER1またはMER2ビットをクリアできます。

全体消去

全体消去を実行するには、次の手順に従います。

1. FLASH_NSSRまたはFLASH_SECSRのBSYを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによる非セキュアなエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERRビットがセットされます。
3. FLASH_NSCRまたはFLASH_SECCRのMER1ビットおよびMER2ビットをセットします。
4. FLASH_NSCRまたはFLASH_SECCRのSTRTをセットします。
5. FLASH_NSSRまたはFLASH_SECSRのBSYビットがクリアされるのを待ちます。
6. 全体消去がそれ以上リクエストされなければ、MER1およびMER2ビットをクリアできます。

注： **内部オシレータHSI16（16MHz）は、STRTビットがセットされた時点で自動的に有効になり、STRTビットがクリアされた時点で自動的に無効になります。ただし、HSI16がRCC_CRのHSIONで事前に有効化される場合は例外とします。**

ページ、バンクを消去する、または全体消去を実行するには、ソフトウェアに十分な特権がなければなりません（表73および表74を参照）。

7.3.7 Flashメインメモリのプログラミングシーケンス

Flashメモリは1回137bit（128bit + 9bit ECC）でプログラミングされます。

書き込むデータがすべてゼロである場合を除き、以前にプログラムされたアドレスではプログラミングできません。これを試みると、FLASH_NSSRまたはFLASH_SECSRにPROGERRフラグがセットされます。

クワッドワード（4 x 32bitデータ）のプログラムのみ可能です。

- バイトまたはハーフワード書込みを試みると、FLASH_NSSRまたはFLASH_SECSRにSIZERRフラグがセットされます。
- クワッドワードアドレスに合っていないクワッドワードの書込みを試みると、FLASH_NSSRまたはFLASH_SECSRにPGAERRフラグがセットされます。

Flashのプログラミング

Flashメモリのプログラミングシーケンスは、次のようになっています。

1. FLASH_NSSRまたはFLASH_SECSRのBSYを確認し、進行中のFlashメインメモリ操作がないことを確認します。

2. FLASH_NSSRまたはFLASH_SECSRのWDWを確認し、書込みバッファが空であることを確認します。
3. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERRがセットされます。
4. FLASH_NSCRまたはFLASH_SECCRのPGビットをセットします。
5. 目的のFlashメモリアドレス、またはOTPエリア内でのデータ書込み操作を実施します。クワッドワードのみをプログラムできます。OTPは非セキュアアクセスのときのみプログラムできます。
   – クワッドワードアドレスに合ったアドレスで最初のワードを書き込みます。FLASH_NSSRおよびFLASH_SECSRのWDWビットは、書込みバッファにより多くのデータを書き込めることを示すためにセットされます。
   – 2番目、3番目、4番目のワードを、同じクワッドワードに書き込みます。
6. BSYビットがセットされます。WDWは自動的にリセットされます。
7. FLASH_NSSR または FLASH_SECSR の BSY がクリアされるまで待ちます。ソフトウェアは、BSYがクリアされるのを待つ前に、BSYをセットするか、WDWをクリアする必要があります。
8. FLASH_NSSRまたはFLASH_SECSRのEOPフラグがセットされている場合（プログラム操作が成功し、EOPIEビットがセットされていることを意味する）、ソフトウェアでクリアする必要があります。
9. プログラミングリクエストがなくなったら、FLASH_NSCRまたはFLASH_SECCRのPGをクリアします。

注： **Flash メモリインタフェースが適切なシーケンス（クワッドワード）を受け取った場合、プログラミングが自動的に起動し、BSYビットがセットされます。内部オシレータHSI16（16MHz）は、PGビットがセットされた時点で自動的に有効になり、PGビットがクリアされた時点で自動的に無効になります。ただし、HSI16がRCC_CRのHSIONで事前に有効化される場合は例外とします。**

WDWビットがセットされている場合、オプションバイトの変更も消去リクエストも許可されません。

特権およびセキュリティ属性が考慮されている場合のみ、プログラミングが可能です（セクション7.7を参照）。

1ワードのみプログラムする必要がある場合にプログラミングを自動で起動するには、消去値0xFFFFFFFFでクワッドワードを埋める必要があります。

ECCは、プログラムするクワッドワードから計算されます。

Flashバースト·プログラミング（8クワッドワード）

Flashメモリのバースト·プログラミング·シーケンスは、次のようになっています。

1. FLASH_NSSRまたはFLASH_SECSRのBSYビットを確認し、進行中のFlashメインメモリ操作がないことを確認します。
2. FLASH_NSSRまたはFLASH_SECSRのWDWを確認し、書込みバッファが空であることを確認します。
3. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERRがセットされます。
4. FLASH_NSCRまたはFLASH_SECCRのBWRおよびPGビットをセットします。
5. 目的のFlashメモリアドレス、またはOTPエリア内でのデータ書込み操作を実施します。8クワッドワードのみプログラム可能です。
   – 8クワッドワードアドレスに合ったアドレスで最初の32bitワードを書き込みます（0x80の倍数）。FLASH_NSSRおよびFLASH_SECSRのWDWビットは、書込みバッファにより多くのデータを書き込めることを示すためにセットされます。
   – 31個のその他の32bitワードを連続して書き込みます。
6. BSYビットがセットされます。WDWは自動的にリセットされます。

7. FLASH_NSSR または FLASH_SECSR の BSY がクリアされるまで待ちます。ソフトウェアは、BSYがクリアされるのを待つ前に、BSYをセットするか、WDWをクリアする必要があります。
8. FLASH_NSSRまたはFLASH_SECSRのEOPフラグがセットされている場合（プログラム操作が成功し、EOPIEがセットされていることを意味する）、ソフトウェアでクリアする必要があります。
9. プログラミングリクエストがなくなったら、FLASH_NSCRまたはFLASH_SECCRのBWRおよびPGビットをクリアします。

注： **Flash メモリインタフェースが適切なシーケンスを受け取った場合、プログラミングが自動的に起動し、BSYビットがセットされます。内部オシレータHSI16（16MHz）は、PGビットがセットされた時点で自動的に有効になり、PGビットがクリアされた時点で自動的に無効になります。ただし、HSI16がRCC_CRレジスタのHSIONで事前に有効化される場合は例外とします。**

WDWビットがセットされている場合、オプションバイトの変更も消去リクエストも許可されません。

特権およびセキュリティ属性が考慮されている場合のみ、プログラミングが可能です（セクション7.7を参照）。

7.3.8 Flashメモリの書換え耐性

各Flashメモリ･ページは、10000回または100000回書込みおよび消去できます。バンクあたり最大32ページ(256KB)で、100kcyclesの耐久性が向上します。この強化された耐久性は、通常、コードストレージよりも集中的なサイクリング機能を必要とするデータストレージに使用できます。

任意のフラッシュページを選択して、最大100kcyclesまでサイクルすることができます。ページが10kcyclesを超えると、そのページは高サイクルページと見なされます(まだ100kcyclesに達していない場合でも)。アプリケーションは、32ページのデータ書換えが10000回を超えないように注意する必要があります。

STM32U535/545の場合、バンク当たり最大32ページ（256KB）が実装されるため、Flashメモリ全体で100Kサイクルに対応できます。

7.3.9 Flashメモリ･エラー･フラグ

Flashプログラミングエラー

セキュアおよび非セキュア操作中には、数種類のエラーが検出される可能性があります。エラーが発生した場合、Flashメモリ操作（プログラミングまたは消去）はアボートされます。

セキュアエラーフラグはセキュア操作中にセットされ、非セキュアフラグは非セキュアアクセス時にのみセットされます。

- **PROGERR**：セキュア/非セキュアプログラミングエラー

  これらのフラグは、プログラムするワードが次のようなアドレスを指しているときにセットされます。

  – 以前に消去されていないアドレス
  – 完全に0にプログラムされているアドレス
  – 部分的にプログラムされるアドレスで（0および1を含む）、新たに設定する値は全ゼロではない
  – OTPプログラミングの場合、アドレスが部分的にプログラムされている（0および1を含む）場合にセットされます。

- **SIZERR**：セキュア/非セキュアサイズプログラミングエラー

  32bitデータのみを書き込めます。バイトまたはハーフワードが書き込まれる場合、SIZERRフラグがセットされます。

- **PGAERR**：セキュア/非セキュア配置プログラミングエラー

  これらのフラグは、プログラムする最初のワードがクワッドワードアドレスと合っていないか、2番目、3番目、または4番目のワードが同じクワッドワードアドレスに属していない場合にセットされます。

  バースト･プログラミングについては、プログラムする最初のワードが8つのクワッドワードアドレスで整列されていない場合や、以下のワード書込みが連続32bitアドレスで行われていない場合にセットされます。

- **PGSERR**：プログラミングシーケンスエラー

  PGSERRは、消去またはプログラム操作中に、次のいずれかの条件が発生した場合にセットされます。

  – PGがクリアされたときにデータが書き込まれた場合
  – MER1、MER2、またはPER がセットされた状態でPGがセットされ、消去中にプログラム操作がリクエストされた場合
  – 消去シーケンスで、すでにSTRTがセットされているときに、PGがセットされた場合
  – 消去シーケンスで、MER1、MER2、およびPERがクリアされているときにSTRTが設定されている場合
  – MER1またはMER2がセットされた状態でSTRTおよびPERがセットされ、ページ消去と全体消去が同時にリクエストされた場合
  – WDWが既にセットされた状態で、STRTもしくはOPTSTRTがセットされ、書込みバッファが次のデータを待っている最中に別の操作が開始された場合
  – STRTおよびOPTSTRTが同時にセットされた場合。
  – 非セキュア STRT ビットがセキュアアクセスによって設定されているときに、非セキュアPGSERRが設定された場合
  – 以前のプログラミングエラーによりPROGERR、SIZERR、PGAERR、WRPERR、またはPGSERRがすでに設定されているときに、セキュアPGSERRが設定された場合
  – 以前のプログラミングエラーにより、PROGERR、SIZERR、PGAERR、WRPERR、PGSERR、またはOPTWERRがすでにセットされているときに、非セキュアなPGSERRがセットされた場合

- **WRPERR**：書込み保護エラー

  – WRPERRフラグセットのすべての条件は、表68から表71を参照してください。

- **OPTWERR**：オプションバイト書込みエラー

  OPTWERRは、ユーザオプションバイトが無効な設定で変更された場合にセットされます。これは、以下のことを試みた場合にセットされます。

  – 無効なウォーターマークベースセキュア領域をプログラム。表59を参照してください。
  – RDP が正しいレベルにない場合に、TZEN オプションビットをセットまたはクリア（特定のオプションバイトを変更する際の規則を参照）
  – RDPが正しいレベルにない場合に、BOOT_LOCKオプションビットをクリア（特定のオプションバイトを変更する際の規則を参照）
  – BOOT_LOCKおよびTZENのセット中にSWAP_BANKオプションビットを変更
  – BOOT_LOCKのセット中にSECBOOTADD0オプションビットを変更
  – BOOT_LOCKおよびTZENのセット中にDUALBANKオプションビットを変更
  – HDP1_ACCDIS（またはHDP2_ACCDIS）のセット中にSECWM1Rx（またはSECWM2Rx）を変更
  – RDP がレベル 2 にセットされている場合に、SWAP_BANK オプションビット以外のオプションバイトを変更
  – RDPレベル0.5からRDPレベル0に解除

- RDPレベルが0.5または1の間にOEM1KEYRxを変更し、OEM1LOCKビットをセット
- RDPレベルが1の間にOEM2KEYRxを変更し、OEM2LOCKビットをセット
- OEM1LOCKビットがセットされ、誤ったOEM1KEYがJTAGまたはSWDを通じてシフトしている間にRDPレベル1からRDPレベル0に解除
- OEM2LOCKビットがセットされ、誤ったOEM2KEYがJTAGまたはSWDを通じてシフトしている間にRDPレベル1からRDPレベル0.5に解除
- UNLOCKビットがクリアされている間にWRPxyRを変更
- RDPが正しいレベルにない場合に、WRPxyRのUNLOCKビットをセット（特定のオプションバイトを変更する際の規則を参照）

セキュアまたは非セキュアなプログラムまたは消去操作中にエラーが発生した場合、次のいずれかのプログラミングエラーフラグがセットされます。

- 非セキュアプログラミングエラーフラグ：FLASH_NSSR の PROGERR、SIZERR、PGAERR、PGSERR、OPTWRERR、またはWRPERRがセットされます。

  FLASH_NSCRのERRIEがセットされている場合、割込みが生成され、FLASH_NSSRレジスタの操作エラーフラグOPERRがセットされます。
- セキュアプログラミングエラーフラグ：FLASH_SECSR の PROGERR、SIZERR、PGAERR、PGSERR、またはWRPERRがセットされます。

  FLASH_SECCRのERRIEがセットされている場合、割込みが生成され、FLASH_SECSRの操作エラーフラグOPERRがセットされます。

注： **複数のエラー（FlashメモリへのDMA転送の場合など）が連続して検出されると、エラーフラグは連続した書込みリクエストが終了するまでクリアできません。任意のプログラミングエラーが書込みバッファを一掃します。**

7.3.10 書込み中読出し（RWW）

Flashメモリが2つのバンクに分割されますので、書込み中読出し操作が可能です。この機能により、1つのバンクで消去操作やプログラム操作を実施している間にもう1つのバンクから読出し操作を実施できます。

注： **書込み中書込み操作はできません。例に示すように、1 つのバンクでプログラミングをしている間にもう1つのバンクで消去操作を実施することはできません。**

バンク2ページ消去中のバンク1での読出し（またはその逆）

バンク1でプログラムコードを実行している間、バンク2でページ消去操作を実行することができます（その逆も可）。

バンク2全体消去中のバンク1での読出し（またはその逆）

バンク1でプログラムコードを実行している間、バンク2で全体消去操作を実行することができます（その逆も可）。

バンク2プログラミング中のバンク1での読出し（またはその逆）

バンク1でプログラムコードを実行している間、バンク2でプログラム操作を実行できます（その逆も可）。

注： **Cortex-M33にはCバスが搭載されているため、ユーザソフトウェアでは複数回の連続書込みによりCバスがストールしないようにする必要があります。したがって、BSYフラグがクリアされるのを待ってから、次のクワッドワードをプログラムすることを推奨します。**

7.3.11 FLASHのプログラム操作または消去操作中のパワーダウン

Flashメモリのプログラム操作または消去操作中にパワーダウンが発生すると、現在アクセスしているFlashメモリの内容は保証されません。

7.3.12 FLASHのプログラム操作または消去操作中のリセット

Flashメモリのプログラム操作または消去操作中にリセットが発生すると、現在アクセスしているFlashメモリの内容は保証されません。Flashメモリのプログラム操作または消去操作中にシステムリセットが発生した場合、FlashメモリのステータスはFLASH_OPSRから復元できます。

ソフトウェアは、Flashメモリのステータスを確認し、是正措置を取る必要があります。これは、各システムリセット後、他のプログラム操作または消去操作が行われる前に行う必要があります。

次の表に、FLASH_OPSRのCODE_OPフィールドによって提供されるステータスに応じて取るべき是正措置について説明します。

表56.システムリセットによるFlash操作の中断

CODE_OP	中断した操作	アドレス	バンク	システムFLASH	是正措置
0x0	何もしない	予約済み			なし
0x1	1回の書込み	ADDR_OP	BK_OP	SYSF_OP	ページ消去および同じ場所での1回の書込み
0x2	バースト書込み	ADDR_OP	BK_OP	SYSF_OP	ページ消去および同じ場所でのバースト書込み
0x3	ページ消去	ADDR_OP	BK_OP	予約済み	同じページの消去
0x4	バンク消去	予約済み	BK_OP	予約済み	同じバンクの消去
0x5	全体消去	予約済み			全体消去
0x6	オプション変更	予約済み			オプション変更
0x7	予約済み				

注： **1回の書込みおよびバースト書込みについては、現在のFlashメモリ位置が書込み可能ではなくなる場合があるため、必ずページ消去を行ってください。そのため、残りのページ内容はページを消去する前に保存して、後で復元する必要があります。**

OTP書込みについては、ページ消去を実施することはできません。OTPクワッドワードは失われます。

バースト書込みについては、ADDR_OPがバーストの最初のアドレスを示します。ユーザは同じバースト動作を再開する必要があります。

ページ消去については、ADDR_OPは消去したページの最初のアドレスを示します。

7.4 Flashオプションバイト

7.4.1 オプションバイトの説明

オプションバイトは、アプリケーション要件によってエンドユーザが設定します。設定の例として、ウォッチドッグをハードウェアまたはソフトウェアモードに選択できます（セクション7.4.2を参照してください）。ユーザオプションバイトは、Flashメモリ・レジスタからアクセスできます。

表57では、Flashメモリ・レジスタで使用可能なすべてのユーザオプションバイトの構成について説明しています。

表57.ユーザオプションバイト構成マッピング

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	レジスタ・マップ
TZEN	IO_VDDIO2_HSLV	IO_VDD_HSLV	PA15_PUPEN	NBOOT0	NSWBOOT0	SRAM2_RST	SRAM2_ECC	SRAM3_ECC	BKPRAM_ECC	DUALBANK	SWAP_BANK	WWDG_SW	IWDG_STDBY	IWDG_STOP	IWDG_SW	SRAM_RST	NRST_SHDW	NRST_STDBY	NRST_STOP	Res.	BOR_LEV[2:0]	RDP										セクション7.9.13
NSBOOTADD0 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	セクション7.9.14
NSBOOTADD1 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	セクション7.9.15
SECBOOTADD0 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK	セクション7.9.16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM1_PEND [6:0]															SECWM1_PSTRT [6:0]							セクション7.9.17
HDP1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP1_PEND [6:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	セクション7.9.18
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_PEND [6:0]															WRP1A_PSTRT [6:0]							セクション7.9.19
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_PEND [6:0]															WRP1B_PSTRT [6:0]							セクション7.9.20
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM2_PEND [6:0]															SECWM2_PSTRT [6:0]							セクション7.9.21
HDP2EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP2_PEND [6:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	セクション7.9.22
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_PEND [6:0]															WRP2A_PSTRT [6:0]							セクション7.9.23
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_PEND [6:0]															WRP2B_PSTRT [6:0]							セクション7.9.24
OEM1KEY [31:0]																															セクション7.9.25
OEM1KEY [63:32]																															セクション7.9.26
OEM2KEY [31:0]																															セクション7.9.27
OEM2KEY [63:32]																															セクション7.9.28

7.4.2 オプションバイトのプログラミング

リセット後、次のオプションレジスタ（FLASH_OPTR、FLASH_NSBOOTADD0R、FLASH_NSBOOTADD1R、FLASH_SECBOOTADD0R、FLASH_SECWM1R1、FLASH_SECWM1R2、FLASH_WRP1AR、FLASH_WRP1BR、FLASH_SECWM2R1、FLASH_SECWM2R2、FLASH_WRP2AR、FLASH_WRP2BR）に加え、FLASH_NSCRのOPT_STRTおよびOBL_LAUNCHビットも書込み保護されます。任意のオプションバイト操作を実行するには、FLASH_NSCRのOPTLOCKをクリアする必要があります。

このレジスタのアンロックには、次のシーケンスを使用します。

1. LOCKクリアシーケンスでFLASH_NSCRレジスタをアンロックします（セキュア/非セキュアなFLASH制御レジスタのアンロックを参照）。
2. FLASH_OPTKEYRにOPTKEY1 = 0x08192A3Bを書き込みます。
3. FLASH_OPTKEYRにOPTKEY2 = 0x4C5D6E7Fを書き込みます。

ソフトウェアによってOPTLOCKビットをセットすると、ユーザオプションを不要な消去/プログラム操作から保護することができます。

注： **FLASH_NSCRのLOCKがソフトウェアによってセットされると、OPTLOCKも自動的にセットされます。**

オプション･バイトの変更シーケンス

ユーザオプション値を変更するには、次の手順に従います。

1. FLASH_NSSRのBSYを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 上記のクリアシーケンスでOPTLOCKをクリアします。
3. 指定したオプション値をオプションレジスタに書き込みます。
4. FLASH_NSCRのOPTSTRTをセットします。
5. BSYビットがクリアされるのを待ちます。
6. オプションバイトローディングを開始するために、OBL_LAUNCHオプションビットをセットします。

注： **OPTWERRまたはPGSERRエラービットがセットされている場合は、古いオプションバイト値が保持されます。**

オプションバイトローディング（OBL）

BSYがクリアされると、すべての新しいオプションがFlashメモリに更新されますが、システムには適用されません。これらはロードされると、システムで有効になります。OBLは次の2つの場合に実行されます。

- FLASH_NSCRのOBL_LAUNCHがセットされた場合
- 電源リセット後（BORリセットまたはSTANDBY/SHUTDOWNモードの終了）

システムリセットの立ち上がりでは、内部オプションレジスタがオプションレジスタにコピーされます。これらのレジスタはオプションバイトの変更にも使用されます。これらのレジスタがユーザによって変更されない場合は、システムのオプション状態が反映されます。

特定のオプションバイトを変更する際の規則

一部のオプションバイトフィールドは、新しい値で更新する前に、特定の規則に従う必要があります。これらのオプションバイトと関連する制約について、以下で説明します。

- TZENオプションビット
  - TZENはRDPレベル0でのみセットできます。

  - TZENの無効化は、RDPがレベル1から0に変化するときにのみ可能です。
- BOOT_LOCKオプションビット
  - BOOT_LOCKは、TZENがセットされているときにのみ効果があります。
  - BOOT_LOCKは何の制約もなくセットできます。
  - BOOT_LOCKの無効化は、RDPがレベル0のときにのみ可能です。
- SWAP_BANKオプションビット
  - BOOT_LOCKおよびTZENオプションビットがセットされている場合は、変更できません。
- SECBOOTADD0オプションバイト
  - BOOT_LOCKオプションビットがセットされている場合は、変更できません。
- DUALBANKオプションビット
  - BOOT_LOCKおよびTZENオプションビットがセットされている場合は、変更できません。
- SECWMxRyオプションビット
  - バンクxのセキュアオプションビット（SECWMx_PSTRT[6:0]およびSECWMx_PEND[6:0]）およびHDPオプションビット（HDPx_PEND[6:0]およびHDPxEN）領域は、HDPx_ACCDISビットがセットされている場合は変更できません。
- RDPオプションビット
  - 読出し保護（RDP）遷移によって管理されるデバイスのライフサイクルを参照してください。
- WRPxyRオプションビット
  - これらのビットは、UNLOCKビットがクリアされるまで変更できません。
- UNLOCKオプションビット
  - これらのビットは、RDPのレベル1からレベル0に解除する場合にのみセットできます。

ユーザオプション変更において、関連規則に従わずにこれらのオプションバイトのいずれかをセットまたは変更しようとすると、オプションバイトの変更は破棄され、OPTWERRエラーフラグがセットされます。

7.5 FLASH TrustZoneのセキュリティと特権の保護

7.5.1 TrustZoneのセキュリティ保護

グローバルTrustZoneセキュリティシステムは、FLASH_OPTRのTZENをセットすることによって有効になります。

TrustZoneが有効（TZEN = 1）な場合、次のような追加のセキュリティ機能が使用可能です。

- セキュア領域およびHDP領域を定義するウォーターマークベースのセキュアなユーザオプションバイト
- リセット後にオンザフライで設定可能なセキュアまたは非セキュアブロックベース領域（揮発性セキュア領域）
- 追加のRDP保護：RDPレベル0.5
- 関連の設定ビットを使用して、セキュアまたは非セキュアモードで実行可能な消去またはプログラム操作

TrustZoneが無効（TZEN = 0）な場合、上記の機能は無効で、すべてのセキュアレジスタはRAZ/WIです。

TrustZoneセキュリティの有効化

TrustZoneを有効化（TZENを0から1に変更）する場合、ウォーターマークベースのセキュアなユーザオプションバイトはデフォルトのセキュア状態にセットされています。表58に示すように、すべてのFlashメモリはセキュアで、HDP領域はありません。

表58.TZEN有効化後のデフォルトのセキュアオプションバイト

OBL後、TZENが有効化された場合のウォーターマークベースのセキュアオプションバイトの値	セキュリティ属性
SECWMx_PSTRT = 0およびSECWMx_PEND = 0x7F	すべてのFlashメモリはセキュア
HDPxEN = 0およびHDPx_PEND = 0	セキュアHDP領域なし

不正アクセスの生成

セキュアFlashメモリ領域への非セキュアアクセスはRAZ/WIで、不正アクセスイベントを生成します。TZIC_IER2のFLASHIE不正アクセス割込みを有効にすると、不正アクセス割込みが生成されます。

FLASHのセキュアレジスタへの非セキュアアクセスにより、不正アクセスイベントが生成されます。TZIC_IER2のFLASH_REGIE不正アクセス割込みを有効にすると、不正アクセス割込みが生成されます。

TrustZoneセキュリティの無効化

TZENの無効化（1から0）は、RDPがレベル1からレベル0に変化するときにのみ可能です。

OBL後にTrustZoneを無効化（TZENを1から0に変更）する場合、次のセキュリティ機能が無効化されます。

- ウォーターマークベースのセキュア領域（セクション7.5.2を参照してください）
- ブロックベースのセキュア領域（セクション7.5.4を参照してください）
- RDPレベル0.5（セクション7.6.2を参照してください）
- セキュア割込み（セクション7.8を参照してください）

すべてのセキュアレジスタはRAZ/WI

7.5.2 ウォーターマークベースのセキュアFlashメモリの領域保護

TrustZoneセキュリティが有効（TZEN = 1）な場合、Flashメモリの一部を、非セキュア読出しおよび書込みアクセスから保護することができます。最大2つの異なる不揮発性セキュア領域はオプションバイトで定義でき、セキュアアクセスのみで読出しまたは書込みができます。バンク当たり1つの領域をページ単位で選択可能です。

セキュア領域は、SECWMx_PSTRTおよびSECWMx_PEND（領域1および領域2でx = 1、2）オプションバイトを使用して、開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットはFLASH_SECWM1R1およびFLASH_SECWM2R1で定義されます。

SECWMx_PSTRTおよびSECWMx_PENDオプションバイトは、HDPx_ACCDISビットがリセットされているときに、セキュアファームウェアによってのみ変更できます。このビットがセットされると、SECWMx_PSTRTおよびSECWMx_PENDは次のシステムリセットまで変更できません。

表59.ウォーターマークベースセキュア領域

ウォーターマークベースのセキュアオプションバイトの値（x = 1、2）	ウォーターマークベースセキュア保護領域
SECWMx_PSTRT > SECWMx_PEND	セキュア領域なし。
SECWMx_PSTRT = SECWMx_PEND	SECWMx_PSTRTで定義された1ページはウォーターマークベースのセキュア保護ページ。
SECWMx_PSTRT < SECWMx_PEND	SECWMx_PSTRTとSECWMx_PENDの間はウォーターマークベースのセキュア保護領域。

注意： **Flashメモリ領域のセキュアから非セキュアへの切替えでは、その内容は消去されません。あるセキュア領域を非セキュア属性に切り替える前に必要に応じてその領域を消去するために、セキュアユーザソフトウェアによって必要な操作を実行する必要があります。また、命令キャッシュを一掃することも推奨します。**

7.5.3 セキュア非表示保護（HDP）

セキュアHDP領域は、Flashメモリのウォーターマークベースのセキュア領域の一部です。非表示保護領域へのアクセスは、FLASH_SECHDPCRのHDPx_ACCDISをセットすることで拒否できます。

HDPx_ACCDISがセットされると、HDP領域での命令フェッチ、データの読出し、書込み、および消去操作は拒否されます。たとえば、セキュアなFlash非表示保護領域にあるソフトウェアコードは一度だけ実行でき、それ以上のこの領域へのアクセスは次のシステムリセットまでは拒否されます。HDPx_ACCDISは、システムリセットによってのみクリアできます。

注： **ソフトウェアは、HDPxENビットをリセットする前に、HDP領域の消去や命令キャッシュの一掃などの任意の適切な処置を取って、HDPコードを保護する必要があります。**

バンク当たり1つの不揮発性セキュアHDP領域は、ページ単位で定義できます。

セキュアHDP領域は、HDPxEN（領域1および2でそれぞれx =1、2）によって有効になります。HDPxENがリセットされているとき、HDP領域はありません。HDPxENビットは、HDPx_ACCDISビットがリセットされている場合、セキュアファームウェアによってオンザフライでセットまたはリセットできます。HDPx_ACCDISがセットされている場合、HDPxENおよびセキュアなウォーターマーク設定は、次のシステムリセットまで変更できません。

セキュアHDP領域のサイズは、終了ページオフセットにより、HDPx_PENDオプションバイトを使用して定義されますが、開始ページオフセットはSECWMx_PSTRTオプションバイトにより定義済みです。これらのオフセットは、セキュアウォーターマークレジスタのアドレスレジスタ、FLASH_SECWM1R1、FLASH_SECWM1R2、FLASH_SECWM2R1、およびFLASH_SECWM2R2で定義されます。

たとえば、アドレス0x0C00 4000（含む）からアドレス0x0C00 5FFF（含む）までをHDPで保護するには、次のようにします。

- バンクがスワップされていない場合、オプションバイトのレジスタは次のようにプログラムする必要があります。
  - SECWM1_PSTRT = 0x2
  - HDP1_PEND = 0x3
- 2つのバンクをスワップする場合、保護をバンク2に適用し、オプションバイトレジスタは次のようにプログラムする必要があります。
  - SECWM2_PSTRT = 0x2
  - HDP2_PEND = 0x3

注： **バンク･スワッピング･メカニズムの詳細については、セクション7.5.8を参照してください。**

無効なセキュアHDP領域が次の表に示すように定義される場合、OPTWERRフラグエラーがセットされ、オプションバイトの変更は破棄されます。

表60.セキュア非表示保護

HDPxウォーターマーク･オプションバイトの値（x =1、2）		HDPエリア
HDPxEN = 0	-	セキュアHDP領域なし
HDPxEN = 1	SECWMx_PSTRT ≤ HDPx_PEND ≤ SECWMx_PEND	SECWMx_PSTRTとHDPx_PENDの間はセキュアHDP保護領域。
	その他	無効なセキュア領域。HDP領域はセキュア領域外に定義される。

次の表に、実現可能なセキュアおよびHPD保護領域の設定の概要を示します。

表61.セキュアおよびHDP保護

セキュアおよびHDPウォーターマーク･オプションバイトの値		保護領域
HDPxEN	オプションバイト	
x	SECWMx_PSTRT > SECWMx_PEND	セキュア領域なし。
0	SECWMx_PSTRT ≤ SECWMx_PEND	セキュアHDP領域なし SECWMx_PSTRTおよびSECWMx_PEND間でセキュア – SECWMx_PSTRT = SECWMx_PENDの場合、SECWMx_PSTRTに定義されている1ページはセキュア保護ページ。
1	SECWMx_PSTRT ≤ HDPx_PEND≤SECWMx_PEND	SECWMx_PSTRTとHDPx_PENDの間はセキュアHDP保護領域。 – SECWMx_PSTRT = HDPx_PENDの場合、HDPx_PENDに定義されている1ページはセキュアHDP保護ページ。
	その他	無効なセキュア領域。HDP領域はセキュア領域外に定義される。

7.5.4 ブロックベースのセキュアFlashメモリの領域保護

すべてのページは、ブロックベースの設定レジスタを使用して、セキュアまたは非セキュアとしてオンザフライで設定可能です。FLASH_SECBB1Rx（またはFLASH_SECBB2Rx）は、バンク1（またはバンク2）のページにセキュリティ属性を設定するために使用します。

ページのセキュリティ属性ビット、SECyBBRxのビットiがセットされている場合、セキュリティ属性はウォーターマークベースのセキュア領域のものと同じです。セキュアページはセキュアアクセスによってのみアクセスできます。

SECyBBiビットがすでにウォーターマークベースのセキュア領域に含まれているページにセットまたはリセットされている場合、ページはウォーターマークベースの保護セキュリティ属性を保持します。

ブロックベースのページセキュリティ属性を変更するには、次の処置を推奨します。

- 関連ページ上で進行中のFlashメモリ操作がないことを確認します。
- SECyBBRxのページのセキュリティ属性ビットiを変更した後で、ISB命令を追加します。

注意： **ページまたはメモリブロックのセキュアから非セキュアへの切替えでは、関連するブロックの内容は消去されません。あるブロックを非セキュア属性に切り替える前に、セキュアユーザソフトウェアによって次の必要な操作を実行する必要があります。**

- ページの内容の消去
- 命令キャッシュの無効化

注： **SECyBBRxビットiのアクセス制御に関しては、表75を参照してください。**

7.5.5 Flashのセキュリティ属性状態

1つ以上のセキュア領域がウォーターマークベースのオプションバイトまたはブロックベースのセキュリティレジスタによって定義されている場合、Flashメモリはセキュアな状態です。

FLASH_SECCRレジスタのINVビットを使用して、Flashのセキュリティ状態を上書きすることができます。

RCCのFLASHENおよびFLASHSMENビットのセキュリティ属性は、Flashメモリのセキュリティ属性に従います。FLASH_SECCRレジスタのINVビットを使用して、RCCのFlashメモリのセキュリティ属性を上書きすることができます。Flashメモリがパワーダウンモードの場合、またはMCUが低電力モードに移行した場合、このINVビットをセットするセキュア·ファームウェアによって、非セキュアファームウェアはFlashメモリクロックを無効にすることができます。

表62.Flashのセキュリティ状態

セキュア領域	INVビット	Flashのセキュリティ状態
なし	0	非セキュア
	1	セキュア
あり	0	セキュア
	1	非セキュア

7.5.6 ブロックベースの特権Flashメモリの領域保護

すべてのページは、ブロックベースの設定レジスタを使用して、特権または非特権としてオンザフライで設定可能です。FLASH_PRIVBB1Rx（またはFLASH_PRIVBB2Rx）レジスタは、バンク1（またはバンク2）のページに特権属性を設定するために使用します。

ページ特権属性（PRIVyBBRxのビットi）がセットされている場合、ページは、特権アクセスによってのみアクセス可能です。非特権ページは、特権アクセスまたは非特権アクセスでアクセス可能です。

ブロックベースの特権属性を変更するには、次の処置を推奨します。

- 関連ページ上で進行中のFlash操作がないことを確認します。
- PRIVyBBRxのページのセキュリティ属性ビットiを変更した後で、ISB命令を追加します。

注意： **ページまたはメモリブロックの特権から非特権への切替えでは、関連するブロックの内容は消去されません。**

注： **PRIVyBBRxビットiのアクセス制御に関しては、表76と表77を参照してください。**

7.5.7 Flashメモリ・レジスタの特権および非特権モード

Flashメモリ・レジスタは、次の規則に従って、FLASH_PRIVCFGRのSPRIVおよびNSPRIVビットに応じて特権および非特権アクセスで読み書きができます。

- SPRIVビット（またはNSPRIV）がリセットされているとき、すべてのセキュア（またはノンセキュア）Flashに保存レジスタは特権と非特権のどちらのアクセスでも読み書きができます。
- SPRIV（または NSPRIV）ビットがセットされているとき、すべてのセキュア（またはノンセキュア）Flashに保存レジスタは特権アクセスのみで読み書きができます。特権レジスタへの非特権アクセスはRAZ/WIです。

表72にFlashメモリ・レジスタのアクセス制御の概要を示します。

7.5.8 バンクスワップの際のFlashメモリバンク属性

SWAP_BANKオプション・ビットがメモリ・マップの各バンクのアドレスを変更します。SWAP_BANKをリセットすると、Flashメモリのバンク1がより小さなアドレス範囲にマッピングされます。SWAP_BANKがセットされると、Flashメモリのバンク1がより大きなアドレス範囲にマッピングされます。

Flashメモリバンク属性は、そのバンクに従うため、バンクをスワップする際に次のレジスタを変更不要です。

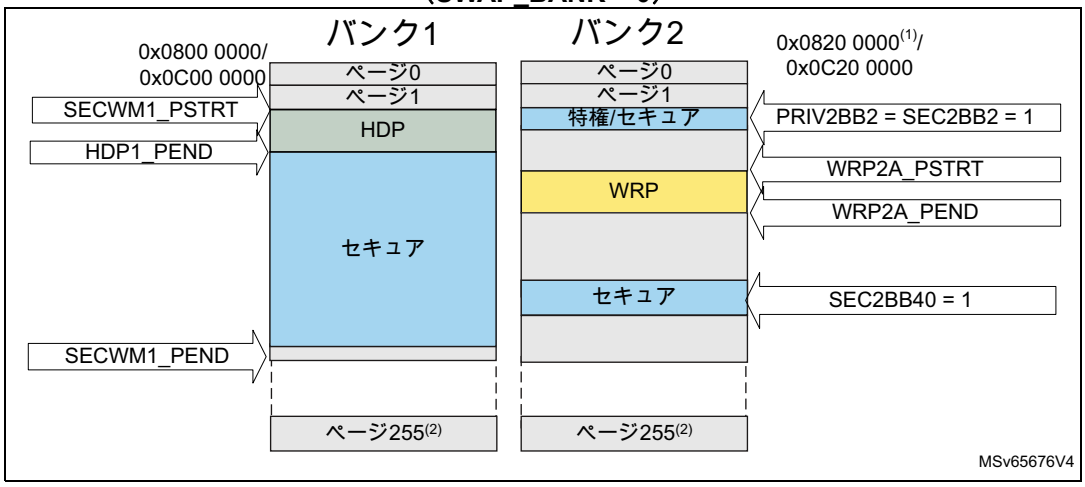
- FLASHのセキュアウォーターマークyレジスタx FLASH_SECWMyRx
- FLASHの書込み保護x領域yFLASH_WRPxyR（セクション7.6.1を参照）
- ブロックベースのFLASHセキュアバンクyレジスタx FLASH_SECyBBRx
- ブロックベースのFLASH特権バンクyレジスタx FLASH_PRIVyBBRx
- FLASH_ACRのPDREQxビット
- FLASH_NSSRのPDxビット

注： **FLASH_ECCRのBK_ECCビットが低い（または高い）場合は、SWAP_BANKの値に関わらず、必ずバンク1（またはバンク2）を参照します。**

FLASH_OPSRのBK_OPビットが低い（または高い）場合は、SWAP_BANKの値に関わらず、必ずバンク1（またはバンク2）を参照します。

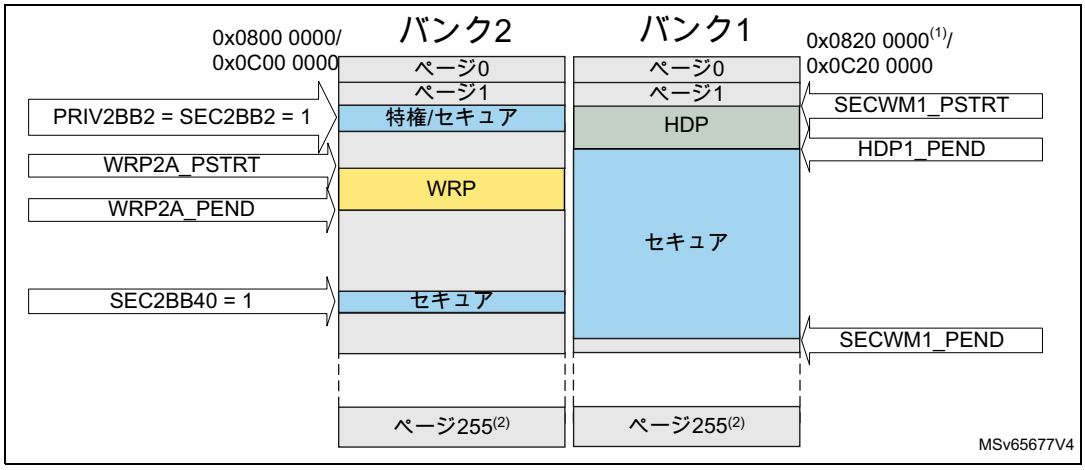
下の図は、バンクスワップ時のセキュリティ属性と保護の動作を示しています。

図23. バンクスワップがない場合のFlashメモリのセキュリティ属性および保護（SWAP_BANK = 0）



1. STM32U59x/5Ax/5Fx/5Gxで有効です。バンク2のベースアドレスは、STM32U575/585の場合は0x0810 0000/0x0C10 0000、STM32U535/545の場合は0x0804 0000/0x0C04 0000です。
2. 各デバイスの最後のページ番号については、表51から表53を参照してください。

図24. バンクスワップがある場合のFlashメモリのセキュリティ属性および保護（SWAP_BANK = 1）



1. STM32U59x/5Ax/5Fx/5Gxで有効です。バンク1のベースアドレスは、STM32U575/585の場合は0x0810 0000/0x0C10 0000、STM32U535/545の場合は0x0804 0000/0x0C04 0000です。
2. 各デバイスの最後のページ番号については、表51から表53を参照してください。

7.6 Flashメモリの保護

Flashメモリインタフェースには、以下の保護メカニズムが実装されています。

- 書込み保護（WRP）
- 読出し保護（RDP）
- TrustZoneが有効な場合の追加のセキュア保護（セクション7.5を参照）
  - 最大2つのウォーターマークベース不揮発性セキュア領域
  - 最大2つのセキュア非表示保護領域
  - ページ単位のブロックベース揮発性セキュア領域
- ページ単位のブロックベース揮発性特権領域（セクション7.5.6を参照）

7.6.1 書込み保護（WRP）

Flashメモリのユーザエリアを、不要な書込み操作から保護できます。2つの書込み保護（WRP）領域が、バンクごとにページ単位で定義できます。

各領域は、物理的なFlashバンクベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットはWRPアドレスレジスタのFLASH_WRP1AR、FLASH_WRP1BR、FLASH_WRP2AR、およびFLASH_WRP2BRで定義されます。

バンク「x」WRP「y」領域（x = 1、2およびy = A、B）は、以下のように定義されます。

- アドレス：バンク「x」ベースアドレス + [WRPxy_PSTRT x 0x2000]（含む）から
- アドレス：バンク「x」ベースアドレス + [(WRPxy_PEND+1) x 0x2000]（含まない）まで

たとえば、アドレス0x0806 2000（含む）からアドレス0x0807 3FFF（含む）までをWRPで保護するには、次のようにします。

- バンクがスワップされていない場合、FLASH_WRP1AR レジスタは次のようにプログラムする必要があります。
  - WRP1A_PSTRT = 0x31
  - WRP1A_PEND = 0x39

  Flash_WRP1BRのWRP1B_PSTRTとWRP1B_PENDが代わりに使用できます（バンク1の領域「B」）。
- 2つのバンクをスワップする場合、保護をバンク2に適用し、Flash_WRP2ARレジスタは次のようにプログラムする必要があります。
  - WRP2A_PSTRT = 0x31
  - WRP2A_PEND = 0x39

  Flash_WRP2BRのWRP2B_PSTRTとWRP2B_PENDが代わりに使用できます（バンク2の領域「B」）。

注： **バンク･スワッピング･メカニズムの詳細については、セクション7.5.8を参照してください。**

WRPがアクティブの場合、保護されたFlashメモリページの消去やプログラムはできません。結果として、領域が1つ書込み保護されているとソフトウェアの全体消去が実施できません。

Flashメモリの書込み保護されている領域に消去/プログラム操作を試みると、FLASH_NSSRまたはFLASH_SECSRレジスタでセキュアまたは非セキュア書込み保護エラーフラグ（WRPERR）がセットされます。このフラグは、次の領域へのすべての書込みアクセスにセットされます。

- システムFlashメモリ
- OTPエリア

注： **メモリ読出し保護レベル1が選択されると（RDPレベル = 1）、CPUデバッグ機能が接続されている場合（JTAGや単線）やブートコードがRAMまたはシステムFlashメモリから実行されている場合には、WRPが無効でもFlashメモリ（セキュアまたは非セキュア）をプログラムしたり消去したりすることはできません。**

メモリ読出し保護レベル0.5が選択されると（RDPレベル = 0.5）、CPUデバッグ機能が接続されている場合（JTAGや単線）には、WRPが無効でもFlashセキュアメモリをプログラムしたり消去したりすることはできません。

注： **WRPオプションを確認するには、Flash制御レジスタのOBL_LAUNCHビットを通じてオプションバイトを再ロードする必要があります。**

表63.WRP保護

WRPレジスタの値（x = 1/2y =A/B）	WRP領域
WRPxy_PSTRT = WRPxy_PEND	ページWRPxyは保護されます。
WRPxy_PSTRT > WRPxy_PEND	WRP領域なし
WRPxy_PSTRT < WRPxy_PEND	WRPxy_PSTRTからWRPxy_PENDまでは保護されたページ

書込み保護ロック

各WRP領域は、FLASH_WRP1AR、FLASH_WRP1BR、FLASH_WRP2AR、またはFLASH_WRP2BRのUNLOCKに0を書き込むことで、個々にロックできます。WRP領域は一度ロックされると、設定を変更できなくなります。WRP領域をアンロックするには、RDPレベル0への解除を始める必要があります。

WRP領域を不変にしてROMとして機能するようにするには、以下の操作が必要です。

- RDPレベルが0、0.5、または1の場合、キーを知らないユーザがRDPレベル0に解除しないように、OEM1KEYをプロビジョニングします。
- RDP レベルが 2 の場合は、OEM1KEY（最初の箇条書きを参照）をプロビジョニングするか、OEM2KEY（レベル2からレベル1への解除を防ぐ）をプロビジョニングしないでください。

RDP解除の詳細については、読出し保護（RDP）遷移によって管理されるデバイスのライフサイクルを参照してください。

7.6.2 読出し保護（RDP）

読出し保護は、Flashメインメモリ、オプションバイト、バックアップレジスタ、バックアップRAMおよびSRAMを保護します。最適な保護レベルに達するには、TrustZoneを有効化し、またパスワード認証解除を有効にしたRDPレベル2をセットします（TrustZoneが有効な場合の読出し保護レベルを参照）。

TrustZoneが無効な場合の読出し保護レベル

読出し保護には、保護なし（レベル0）から最大保護またはデバッグなし（レベル2）までの3つのレベルがあります。

Flashメモリは、次の表に示すRDPオプションバイト値に従って保護されます。

表64.Flashメモリ読出し保護ステータス（TZEN = 0）

RDPバイト値	読出し保護レベル
0xAA	レベル0
0xAAまたは0xCCを除く任意の値	レベル1
0xCC	レベル2

- レベル0：保護なし

  Flashメモリのメイン·メモリ領域に、読出し、プログラム、および消去操作を実行できます。すべての操作でオプションバイト、SRAM、バックアップレジスタにもアクセスできます。

- レベル1：読出し保護

  読出し保護レベル1をセットすると次のようになります。

  – **ユーザモード**：ユーザモードで実行されるコード（**BootFlash**）により、すべての操作（読出し、消去、プログラム）でFlashメインメモリ、オプションバイト、SRAM、バックアップレジスタにアクセスできます。

  – **デバッグ、ブートRAM、およびブートローダモード**：デバッグモードで、またはMCUがRAMまたはシステムメモリからブートするとき、Flashメインメモリ、バックアップレジスタ、バックアップRAM、およびSRAM2には、完全にアクセスできません。Flashメインメモリへの任意の読出しまたは書込みアクセスにより、バスエラーおよびハードフォールト割込みが発生します。オンザフライ復号領域（OCTOSPIのOTFDEC）が0として読み出されます。

- レベル2：デバッグなし

  読出し保護レベル2をセットすると次のようになります。

  – 読出し保護レベル1が保証されます。

  – すべてのデバッグ機能が無効になります。

    OEM2キーが提供されていない場合、JTAGとSWDは完全に無効化されます。

    OEM2キーが低いRDP保護で提供された場合、JTAGとSWDはリセット後も有効なままです。これは、デバイスIDを取得するためにDBGMCU_SR、DBGMCU_DBG_AUTH_HOST、およびDBGMCU_DBG_AUTH_DEVICEレジスタとインタフェースし、RDP解除をリクエストするためにOEM2キーを提供するためです。

  – SRAMからのブート（ブートRAMモード）およびシステムメモリからのブート（ブートローダモード）は使用できません。

  – メインFlashメモリからのブートのみ可能です。Flashメインメモリのすべての操作が許可されます。ユーザコードからのFlashメモリおよびSRAMへの読出し、消去、プログラムアクセスが可能です。

  – SWAP_BANK ビットを除くすべてのオプションバイトはプログラムも消去もできません。そのため、レベル2は解除する方法がありません。これは、OEM2キーがプロビジョニングされていない限り、不可逆操作です（OEM2 RDPロック機構を参照）。

注： **リセット後、デバッグ機能は無効化されます。**

STMicroelectronicsは、レベル2保護がセットされている不良部品には解析を実施できません。解析のために部品を返す前に、RDPレベル1に解除してください（OEM2 RDPロック機構を参照）。

表65.アクセス状態に対する保護レベルと実行モード（TZEN = 0の場合）

領域	RDP レベル	ユーザ実行（Flashからのブート）			RAM/ブートローダからのデバッグ/ブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
Flashメインメモリ	1	あり	あり	あり	なし	なし	なし (4)
	2	あり	あり	あり	N/A	N/A	N/A
システムメモリ (2)	1	あり	なし	なし	あり	なし	なし
	2	あり	なし	なし	N/A	N/A	N/A
オプションバイト(3)	1	あり	あり(4)	N/A	あり	あり (4)	N/A
	2	あり	なし(5)	N/A	N/A	N/A	N/A
OTP	1	あり	あり(6)	N/A	あり	あり (6)	N/A
	2	あり	あり (6)	N/A	N/A	N/A	N/A
バックアップレジスタ	1	あり	あり	N/A	なし	なし	N/A(7)
	2	あり	あり	N/A	N/A	N/A	N/A
SRAM2/バックアップRAM	1	あり	あり	N/A	なし	なし	N/A(8)
	2	あり	あり	N/A	N/A	N/A	N/A
OTFDEC領域（OCTOSPI）	1	あり	あり	あり	なし	あり	あり(9)
	2	あり	あり	あり	N/A	N/A	N/A

1. 保護レベル2が有効な場合、デバッグポート、RAMからのブート、およびシステムメモリからのブートは無効化されます。
2. システムメモリは、保護レベル（0、1、または2）および実行モードにかかわらず、読出しアクセスのみ可能です。
3. オプションバイトは、FlashレジスタのインタフェースとOPTSTRTビットからのみアクセス可能です。
4. メインFlashメモリは、RDPオプションバイトがレベル1からレベル0に変更されると消去されます。
5. SWAP_BANKオプションビットを変更できます。
6. OTPの書込みは1回のみです。
7. バックアップ·レジスタは、RDPがレベル1からレベル0に変更された場合に消去されます。
8. すべてのSRAMは、RDPがレベル1から0に変更された場合に消去されます。
9. OTFDECキーは、RDPオプションバイトがレベル1からレベル0に変更されると消去されます。

TrustZoneが有効な場合の読出し保護レベル

読出し保護には、保護なし（レベル0）から最大保護またはデバッグなし（レベル2）までの4つのレベルがあります。Flashメモリは、次の表に示すRDPオプションバイト値に従って保護されます。

表66.Flashメモリ読出し保護ステータス（TZEN = 1）

RDPバイト値	読出し保護レベル
0xAA	レベル0
0x55	レベル0.5
0xAA、0x55、0xCC以外のすべての値	レベル1
0xCC	レベル2

- レベル0：保護なし

  Flashメモリのメイン･メモリ領域に、読出し、プログラム、および消去操作を実行できます。すべての操作でオプションバイト、SRAM、バックアップレジスタにもアクセスできます。

  – **RSSモード**：RSSからブートする場合、デバッグアクセスはRSSコードの実行中は無効になります。

- レベル0.5：非セキュアデバッグのみ

  非セキュアFlashメモリに対するすべての読出し/書込み操作が可能です（書込み保護がセットされていない場合）。セキュア領域へのデバッグアクセスは禁止されます。非セキュア領域へのデバッグアクセスは可能なままです。

  – **ユーザモード**：ユーザモードで実行されるコード（**BootFlash**）により、すべての操作（読出し、消去、プログラム）でFlashメインメモリ、オプションバイト、SRAM、バックアップレジスタにアクセスできます。

  – **非セキュアデバッグモード**：非セキュアデバッグはCPUが非セキュア状態の場合に可能です。セキュアFlashメモリ、セキュアバックアップレジスタ、およびSRAM領域はアクセスできません。非セキュアFlashメモリ、非セキュアバックアップレジスタ、および非セキュアSRAM領域は、デバッグ目的でのみアクセス可能な状態を維持します。

  – **RSSモード**：RSSからブートする場合、デバッグアクセスはRSSコードの実行中は無効になります。

  – **ブートRAMモード**：SRAMからのブートはできません。

- レベル1：読出し保護

  読出し保護レベル1をセットすると次のようになります。

  – **ユーザモード**：ユーザモードで実行されるコード（**BootFlash**）により、すべての操作（読出し、消去、プログラム）でFlashメインメモリ、オプションバイト、SRAM、バックアップレジスタにアクセスできます。

  – **非セキュアデバッグモード**：非セキュアデバッグはCPUが非セキュア状態の場合に可能です。ただし、デバッグアクセスの際には侵入が検出されます。Flashメインメモリ、バックアップレジスタ、バックアップRAM、およびSRAM2には完全にアクセスできません。Flashメインメモリへの任意の読出しまたは書込みアクセスにより、バスエラーおよびハードフォールト割込みが発生します。オンザフライ復号領域（OCTOSPIのOTFDEC）が0として読み出されます。

  – **RSSモード**：RSSからブートする場合、デバッグアクセスはRSSコードの実行中は無効になります。

  – **ブートRAMモード**：SRAMからのブートはできません。

- レベル2：デバッグなし

  読出し保護レベル2をセットすると次のようになります。

  – 読出し保護レベル1が保証されます。

  – すべてのデバッグ機能が無効になります。

    OEM2キーが提供されていない場合、JTAGとSWDは完全に無効化されます。

    OEM2キーが低いRDP保護で提供された場合、JTAGとSWDはリセット後も有効なままです。これは、デバイスIDを取得するためにDBGMCU_SR、DBGMCU_DBG_AUTH_HOST、およびDBGMCU_DBG_AUTH_DEVICEレジスタとインタフェースし、RDP解除をリクエストするためにOEM2キーを提供するためです。

  – SRAMからのブート（ブートRAMモード）およびシステムメモリからのブート（ブートローダモード）は使用できません。

  – RSSからのブートが可能です。

  – メインFlashまたはRSSからブートする場合は、Flashメインメモリのすべての操作が許可されます。ユーザコードからのFlashメモリおよびSRAMへの読出し、消去、プログラムアクセスが可能です。

  – SWAP_BANK ビットを除くすべてのオプションバイトはプログラムも消去もできません。そのため、レベル2は解除する方法がありません。これは、OEM2キーがプロビジョニングされていない限り、不可逆操作です（OEM2 RDPロック機構を参照）。

注： **リセット後、デバッグ機能は無効化されます。**

STMicroelectronicsは、レベル2保護がセットされている不良部品には解析を実施できません。解析のために部品を返す前に、RDPレベル1に解除させます（OEM2 RDPロック機構を参照）。

表67.アクセス状態に対する保護レベルと実行モード（TZEN = 1の場合）

領域	RDPレベル	ユーザ実行（Flashからのブート）			デバッグ/ブートローダ(1)		
		読出し	書込み	消去	読出し	書込み	消去
Flashメインメモリ	0.5	あり	あり	あり	あり(2)	あり (2)	あり (2)
	1	あり	あり	あり	なし	なし	不可 (6)
	2	あり	あり	あり	N/A	N/A	N/A
システムメモリ (3)	0.5	あり	なし	なし	あり	なし	なし
	1	あり	なし	なし	あり	なし	なし
	2	あり	なし	なし	N/A	N/A	N/A
オプションバイト(4)	0.5	あり	あり (6)	N/A	あり	あり(5) (6)	N/A
	1	あり	あり(6)	N/A	あり	あり (5) (6)	N/A
	2	あり	なし(7)	N/A	N/A	N/A	N/A
OTP	0.5	あり	あり (8)	N/A	あり	あり (8)	N/A
	1	あり	あり(8)	N/A	あり	あり (8)	N/A
	2	あり	あり (8)	N/A	N/A	N/A	N/A

表67.アクセス状態に対する保護レベルと実行モード（TZEN = 1の場合）（続き）

領域	RDP レベル	ユーザ実行（Flashからのブート）			デバッグ/ブートローダ(1)		
		読出し	書込み	消去	読出し	書込み	消去
バックアップレジスタ	0.5	あり	あり	N/A	あり（2）	あり（2）	N/A（9）
	1	あり	あり	N/A	なし	なし	N/A(9)
	2	あり	あり	N/A	N/A	N/A	N/A
SRAM2/バックアップRAM	0.5	あり	あり	N/A	あり（2）	あり（2）	N/A（10）
	1	あり	あり	N/A	なし	なし	N/A(10)
	2	あり	あり	N/A	N/A	N/A	N/A
OTFDEC領域（OCTOSPI）	0.5	あり	あり	あり	なし	あり	あり(11)
	1	あり	あり	あり	なし	あり	あり（11）
	2	あり	あり	あり	N/A	N/A	N/A

1. 保護レベル2が有効な場合、デバッグポートおよびブートローダモードは無効化されます。
2. TrustZoneセキュリティのアクセス権限によって異なります。
3. システムメモリは、保護レベル（0、1、または2）および実行モードにかかわらず、読出しアクセスのみ可能です。
4. オプションバイトは、Flashレジスタのインタフェースと OPTSTRT ビットからのみアクセス可能です。
5. ブートローダは非セキュア·オプション·バイトのみを変更できます。
6. メインFlashメモリは、RDPオプションバイトがレベル1からレベル0に解除すると消去されます。
7. SWAP_BANKオプションビットを変更できます。
8. OTPの書込みは1回のみです。
9. バックアップレジスタは、RDPがレベル1からレベル0に変更された場合、およびRDPがレベル1からレベル0.5に変更された場合に消去されます。
10. すべてのSRAMは、RDPがレベル1からレベル0に変更された場合、およびRDPがレベル1からレベル0.5に変更された場合に消去されます。
11. OTFDECキーは、RDPオプションバイトがレベル1からレベル0に変更された場合、およびRDPがレベル1からレベル0.5に変更された場合に消去されます。

読出し保護（RDP）遷移によって管理されるデバイスのライフサイクル

レベル0からレベル0.5または1への移動は、RDPバイトの値を任意の値（0xCCを除く）に変更することで、簡単に行うことができます。RDPバイトに0xCC値をプログラミングすることで、直接レベル0、0.5、または1からレベル2に移動することができます。レベル2にすると、OEM2キーがプロビジョニングされない限り、読出し保護レベルを変更できなくなります（OEM2 RDPロック機構を参照）。

レベル1からレベル0に移行するためにRDPが値0xAAに再プログラムされると、Flashメインメモリとすべての SRAM で全体消去が実行されます。バックアップレジスタ、OTFDEC キー、ICACHE、DCACHE、およびPKASRAMも消去されます。OTP領域は消去されません。

RDP レベル0.5では、RDP レベル0をリクエストすることはできません。代わりに、RDPのレベル1に高めた後、レベル0へのRDP解除が必要です。

レベル1からレベル0.5に移行するためにRDPが値0x55にプログラムされると、Flashメインメモリの部分消去が実行されます。ウォーターマークベースの非セキュア領域のみが消去されます（ブロックベースでセキュアと定義されている場合でも）。バックアップレジスタ、OTFDECキー、ICACHE、DCACHE、PKASRAM、およびすべてのSRAMが全体消去されます。OTP領域は消去されません。RDP レベル0.5および非セキュアの部分消去は、TrustZoneが有効な場合にのみ使用できます。

注： 完全全体消去は、レベル1がアクティブでレベル0がリクエストされたときにのみ実施されます。保護レベルが引き上げられた場合には（0から0.5、0から1、0.5から1、1から2、0から2、0.5から2）、全体消去は実施されません。

読出し保護レベルの変更を確認するには、FLASH非セキュア制御レジスタ（FLASH_NSCR）のOBL_LAUNCHビットを通じてオプションバイトを再ロードする必要があります。

RDP解除を起動する前に、ソフトウェアはICACHEを無効化して、BUSYFビットがクリアされるのを待つ必要があります。

図25. TrustZoneが無効（TZEN = 0）な場合のRDPレベル遷移図

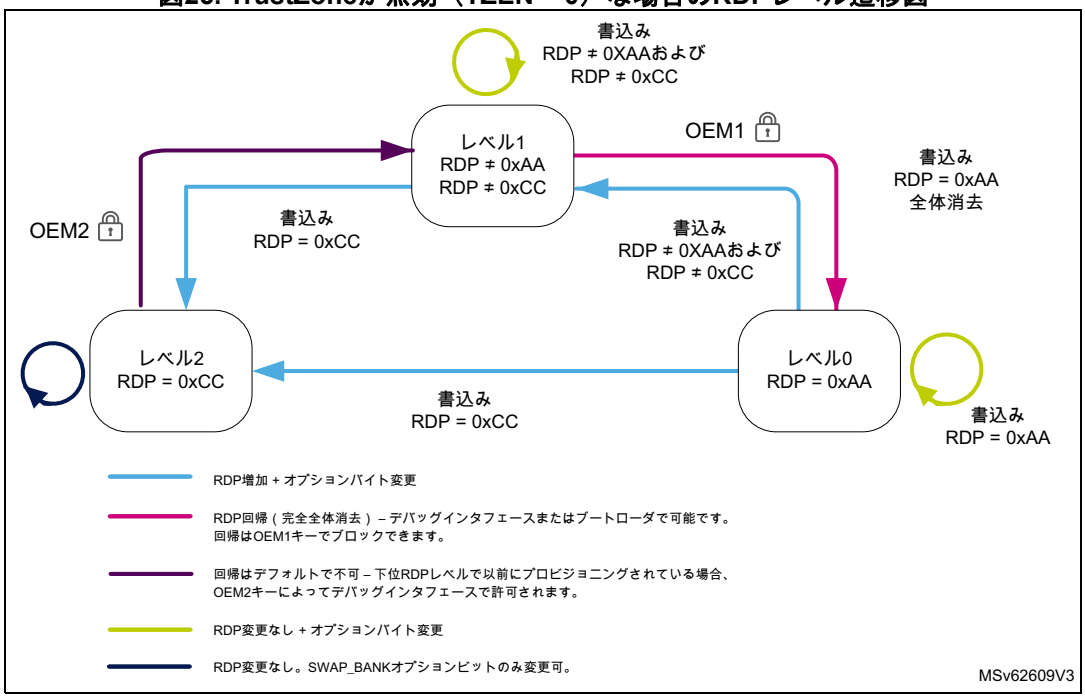
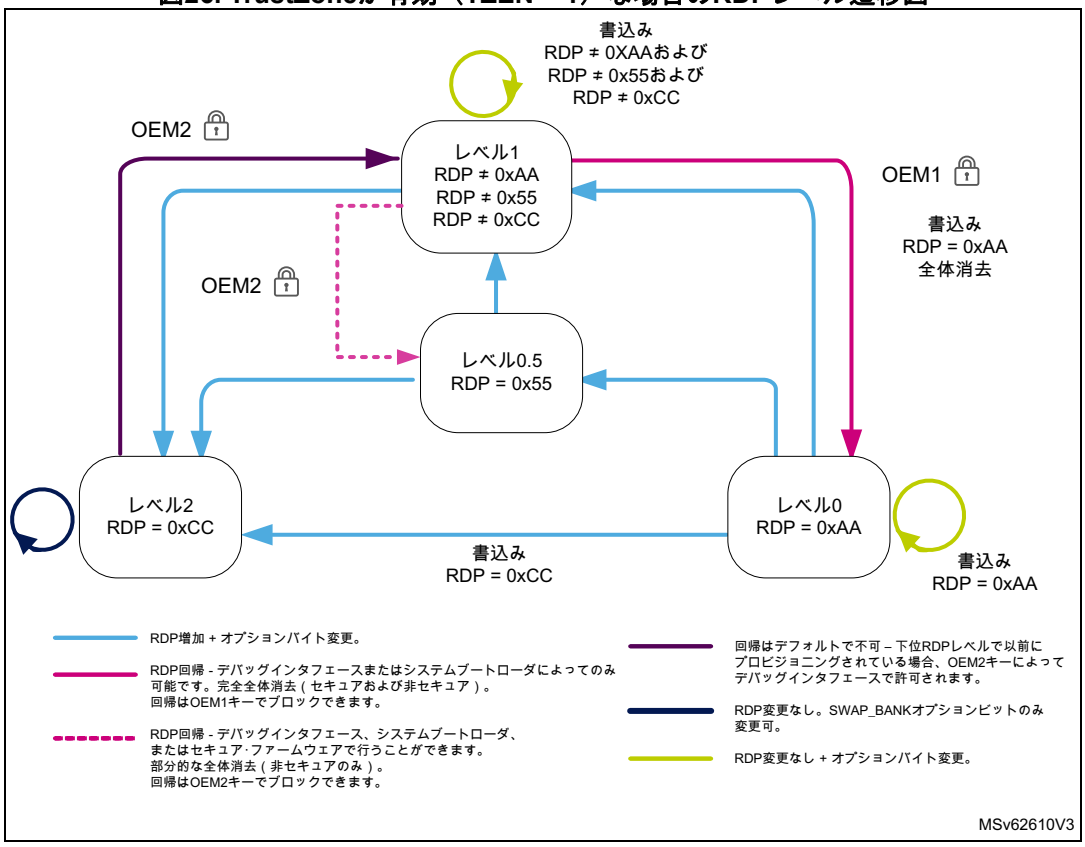


図26. TrustZoneが有効（TZEN = 1）な場合のRDPレベル遷移図



OEM1/OEM2ロックの有効化

2つの64bitのキー（OEM1KEYおよびOEM2KEY）は、RDP解除をロックするために定義できます。各64bitのキーは、FLASH_OEM1KEYR1（またはFLASH_OEM2KEYR1）およびFLASH_OEM1KEYR2FLASH_OEM2KEYR2）の2つのレジスタでコード化されますOEM1KEYおよびOEM2KEYは、これらのレジスタを通じて読み出すことができません。これらは、0として読み出されます。

OEM1KEYは、以下の場合に変更可能です。

- 読出し保護レベル0
- 読出し保護レベル0.5または1（FLASH_NSSRでOEM1LOCK = 0の場合）

OEM2KEYは、以下の場合に変更可能です。

- 読出し保護レベル0または0.5
- 読出し保護レベル1（FLASH_NSSRでOEM2LOCK = 0の場合）

これらの規則に従わずにFLASH_OEM1KEYR1、FLASH_OEM1KEYR2（またはFLASH_OEM2KEYR1、FLASH_OEM2KEYR2）を変更しようとすると、ユーザオプションの変更は行われず、OPTWERRビットが変更されます。

OEM1ロック機構を有効化するには、次の手順が必要です。

- OEM1LOCKビットがセットされていないか、読出し保護がレベル0であることを確認します。
- FLASH_OEM1KEYR1およびFLASH_OEM1KEYR2の64bitのキーを書き込みます。
- FLASH_NSCRのOPTSTRTビットをセットして、オプション変更を起動します。
- BSYビットがクリアされるのを待ってOPTWERRがセットされていないことを確認します。

- オプションバイトローディングを開始するか、パワーオンリセットを行うために、OBL_LAUNCHオプションビットをセットします。
- OEM1LOCKがセットされていることを確認します。

OEM2ロック機構を有効化するには、次の手順が必要です。

- OEM2LOCKビットがセットされていないか、読出し保護がレベル0または0.5であることを確認します。
- FLASH_OEM2KEYR1およびFLASH_OEM2KEYR2の64bitのキーを書き込みます。
- FLASH_NSCRのOPTSTRTビットをセットして、オプション変更を起動します。
- BSYビットがクリアされるのを待ってOPTWERRがセットされていないことを確認します。
- オプションバイトローディングを開始するか、パワーオンリセットを行うために、OBL_LAUNCHオプションビットをセットします。
- OEM2LOCKがセットされていることを確認します。

注： **OEM1KEYおよびOEM2KEYに1または0のみが含まれていてはなりません。**

OEM1 RDPロック機構

OEM1 RDPロック機構は、OEM1LOCKビットがセットされたときにアクティブになります。RDPレベル1からRDPレベル0への解除をブロックします。

RDPレベル1からRDPレベル0に解除するには、以下のアンロックシーケンスを適用する必要があります。

- OEM1KEY[31:0] と OEM1KEY[63:32] を DBGMCU_DBG_AUTH_HOST レジスタの JTAG または SWDを通じてシフトします。
- このキーがOEM1KEY値と一致する場合は、OPTSTRTビットをセットしてRDP解除を始めることができます。
- キーがOEM1KEY値と一致しない場合は、RDP解除とFlashメモリへのすべてアクセスが次のパワーオンリセットまでブロックされます。

このシーケンスに従わずにRDP レベル1からRDP レベル0に解除しようとすると、OPTWERRオプションビットがセットされ、オプションバイトは変更されません。

ロック機構がアクティブ化されていない場合（OEM1LOCK = 0）、RDPレベル1からRDPレベル0への解除が常に許可されます。

OEM2 RDPロック機構

OEM2 RDP機構は、OEM2LOCKビットがセットされたときにアクティブになります。以下の操作が可能になります。

- RDPレベル1からRDPレベル0.5への解除をブロックします。
- RDPレベル2からRDPレベル1への解除を許可します。

RDPレベル1からRDPレベル0.5に解除するには、以下のアンロックシーケンスを適用する必要があります。

- リセット状態で、OEM2KEY[31:0]とOEM2KEY[63:32]をDBGMCU_DBG_AUTH_HOSTレジスタのJTAGまたはSWDを通じてシフトします。
- このキーがOEM2KEY値と一致する場合は、OPTSTRTビットをセットしてRDP解除を始めることができます。
- キーがOEM2KEY値と一致しない場合は、RDP解除とFlashメモリへのすべてアクセスが次のパワーオンリセットまでブロックされます。

RDPレベル2からRDPレベル1に解除するには、以下のアンロックシーケンスを適用する必要があります。

- リセット状態で、OEM2KEY[31:0]とOEM2KEY[63:32]をDBGMCU_DBG_AUTH_HOSTレジスタのJTAGまたはSWDを通じてシフトします。
- このキーがOEM2KEY値と一致する場合：
  - ハードウェアによるRDP解除が始まります（キーが一致している場合は、命令を実行できません）。
  - パワーオンリセットを適用します（サイクルV_{DD}電源のオン/オフ）。
- キーがOEM2KEY値と一致しない場合は、RDP解除とFlashメモリへのすべてアクセスが次のパワーオンリセットまでブロックされます。

これらのシーケンスに従わずにRDPレベル2からRDPレベル1に解除しようとすると、オプションバイトは変更されないままです。

これらのシーケンスに従わずにRDPレベル1からRDPレベル0.5に解除しようとすると、OPTWERRオプションビットがセットされ、オプションバイトは変更されません。

ロック機構がアクティブ化されていない場合（OEM2LOCK = 0）、以下が発生します。

- RDPレベル1からRDPレベル0.5への解除が常に許可されます。
- RDPレベル2からRDPレベル1への解除が常に許可されません。オプションバイトを変更しようとすると、FLASH_NSSRレジスタの保護エラーフラグOPTWERRがセットされ、割込みを生成できます。

7.7 FlashメモリとFLASHレジスタのアクセス制御

次の表に、すべてのFlashメモリとレジスタのアクセス状態に対するRDPレベル、WRP、およびHDP保護の概要を示します。

表68.TrustZoneが有効（TZEN = 1）な場合のFlashメモリアクセスに対するRDPレベル

<table>
<tr><th colspan="2" rowspan="2">アクセスタイプ</th><th colspan="3">RDPレベル0、レベル0.5、レベル1侵入なし(1) またはレベル2</th><th>RDPレベル1侵入あり(2)</th></tr>
<tr><th rowspan="2">非セキュアページ</th><th colspan="2">セキュアページ</th><th rowspan="2">非セキュアまたはセキュアページ</th></tr>
<tr><th>HDP領域（HDPxEN = 1 およびACCDIS = 1）</th><th>その他(3)</th></tr>
<tr><td rowspan="4">セキュア</td><td>フェッチ</td><td>バスエラー</td><td rowspan="2">RAZ</td><td rowspan="2">OK</td><td rowspan="3">バスエラー</td></tr>
<tr><td>読出し</td><td>RAZ、Flashの不正アクセスイベント</td></tr>
<tr><td>書込み</td><td rowspan="2">WI、セキュアWRPERRフラグセット、FLASHの不正アクセスイベント</td><td rowspan="2">WI、セキュアWRPERRフラグセット</td><td rowspan="2">WRPなし：OK
WRPページ：WIおよびセキュアWRPERRフラグセット</td></tr>
<tr><td>ページ消去</td><td>WI、セキュアWRPERRフラグセット</td></tr>
<tr><td rowspan="4">非セキュア</td><td>フェッチ</td><td rowspan="2">OK</td><td colspan="2">バスエラー</td><td rowspan="2">バスエラー</td></tr>
<tr><td>読出し</td><td colspan="2">RAZ、Flashの不正アクセスイベント</td></tr>
<tr><td>書込み</td><td>WRPなし：OK
WRPページ：WIおよび非セキュアWRPERRフラグセット</td><td colspan="2" rowspan="2">WI、非セキュアWRPERRフラグセット、FLASHの不正アクセスイベント</td><td rowspan="2">WI、非セキュアWRPERRフラグセット</td></tr>
<tr><td>ページ消去</td></tr>
</table>

1. RDPレベル1侵入なし = ユーザFlashメモリからのブートおよびデバッグアクセスなしの場合。

2. RDPレベル1侵入あり=デバッグアクセスが検出ありの場合。
3. その他とは、HDP保護に関する設定以外のFlashメモリのセキュア設定を指します。
例：セキュアなFlashメモリと有効なHDP領域、ただしACCDIS = 0。

表69.TrustZoneが無効（TZEN = 0）な場合のFlashメモリアクセスに対するRDPレベル

アクセスタイプ	RDPレベル0、レベル1侵入なし(1)、またはレベル2	RDPレベル1侵入あり(2)
フェッチ	OK	バスエラー
読出し		
書込み	WRPなし：OK WRPページ：WIおよび非セキュアWRPERRフラグセット	
消去		WIおよび非セキュアWRPERRフラグセット

1. RDPレベル1侵入なし = ユーザFlashメモリからのブートおよびデバッグアクセスなしの場合。
2. RDPレベル1侵入あり = RAMまたはシステムメモリからのブート、またはデバッグアクセスが検出ありの場合。

表70.TrustZoneが有効（TZEN = 1）な場合のFlashメモリ全体消去に対するRDPレベル

アクセスタイプ		RDPレベル0、レベル0.5、レベル1侵入なし(1)、またはレベル2				RDPレベル1侵入あり(2)
		非セキュアFlashメモリ	セキュアFlashメモリ		非セキュアおよびセキュアFlashメモリの組み合わせ	非セキュアおよびセキュアFlashメモリ
			HDP領域（HDPxEN = 1およびACCDIS = 1）	その他(3)		
セキュア	バンクまたは全体消去	WI、セキュアWRPERRフラグセット、Flashメモリの不正アクセスイベント	WI、セキュアWRPERRフラグセット	WRPなし：OK WRPページ：WIおよびセキュアWRPERRフラグセット	WI、セキュアWRPERRフラグセット、Flashメモリの不正アクセスイベント	WI、セキュアWRPERRフラグセット
非セキュア	バンクまたは全体消去	WRPなし：OK WRPページ：WIおよび非セキュアWRPERRフラグセット	WI、非セキュアWRPERRフラグセット、Flashメモリの不正アクセスイベント			WI、非セキュアWRPERRフラグセット

1. RDPレベル1侵入なし = ユーザFlashメモリからのブートおよびデバッグアクセスなしの場合。
2. RDPレベル1侵入あり = デバッグアクセスが検出ありの場合。
3. その他とは、HDP保護に関する設定以外のFlashメモリのセキュア設定を指します。
例：セキュアなFlashメモリと有効なHDP領域、ただしACCDIS = 0。

表71.FlashシステムメモリのOTPおよびRSSアクセス(1)

アクセスタイプ		システムメモリ（ブートローダ）	OTP	RSS
セキュア（TZEN=1）	フェッチ	バスエラー		RAZ
	読出し	RAZ、Flashメモリ･レジスタの不正アクセスイベント	OK	
	書込み	WI、セキュアWRPERRフラグセット、Flashメモリの不正アクセスイベント		WI セキュアWRPERRフラグセット

表71.FlashシステムメモリのOTPおよびRSSアクセス(1)（続き）

アクセスタイプ		システムメモリ（ブートローダ）	OTP	RSS
非セキュア（TZEN = 0 または TZEN = 1）	フェッチ	OK	バスエラー	バスエラー
	読出し		OK	RAZ(2)
	書込み	WI 非セキュアWRPERR フラグセット	OK 未使用ではない場合： WI、非セキュアPROGERR フラグセット	WI、 非セキュアWRPERRフラグセット

1. すべてのRDPレベルで有効です。
2. Flashメモリの不正アクセスイベントはTZEN = 1のときに生成されます。

表72.Flashレジスタへのアクセス(1)

アクセスタイプ			非セキュアレジスタ		セキュアレジスタ	
			NSPRIV = 1	NSPRIV = 0	SPRIV = 1	SPRIV = 0
フェッチ	セキュア/非セキュア	特権/非特権	バスエラー			
読出し/書込み	セキュア(2)	特権	OK			
		非特権	RAZ/WI	OK	RAZ/WI	OK
	非セキュア(3)	特権	OK		RAZ/WIおよびFlashメモリ・レジスタの不正アクセスイベント(4)	
		非特権	RAZ/WI	OK		

1. SECyBBRx、PRIVyBBRx、およびPRIVCFGRレジスタ以外。
2. セキュアアクセスはTrustZoneが有効（TZEN = 1）な場合のみ有効です。
3. 非セキュアアクセスはTrustZoneが有効でも無効でも有効です。
4. Flashレジスタの不正アクセスイベントはTZEN = 1のときのみ生成されます。

表73.Flashページアクセスに対する特権モード(1)

アクセスタイプ		非特権ページ	特権ページ
フェッチ、読出し/書込み、ページ消去	特権	OK	
フェッチ、読出し	非特権	OK	RAZ
書込み、ページ消去	非特権		WI、セキュアまたは非セキュアWRPERRフラグセット

1. TZEN = 1の場合、アクセスは、特権を考慮する前にセキュリティファイアウォールが許可する必要があります。

表74.Flash全体消去に対する特権モード(1)

アクセスタイプ		非特権Flashメモリ	特権Flashメモリ	非特権および特権Flashメモリの組み合わせ
全体消去	特権	OK		
全体消去	非特権	OK	WI、セキュアまたは非セキュアWRPERRフラグセット	

1. TZEN = 1の場合、アクセスは、特権を考慮する前にセキュリティファイアウォールが許可する必要があります。

表75.TrustZoneが有効（TZEN = 1）な場合のSECyBBRxレジスタへのアクセス

アクセスタイプ			PRIVyBBRxのビットi	SECyBBRxのビットi
フェッチ	セキュア/非セキュア	特権/非特権	-	バスエラー
読出し	セキュア/非セキュア	特権/非特権	-	OK
書込み	セキュア	特権	-	OK
		非特権	0	ビットiについてOK
		非特権	1	ビットiについてWI
	非セキュア	特権/非特権	-	WIおよびFlashメモリ･レジスタの不正アクセスイベント

表76.TrustZoneが有効（TZEN = 1）な場合のPRIVyBBRxレジスタへのアクセス

アクセスタイプ			ページのセキュア状態（ウォーターマークベースまたはブロックベース）	PRIVyBBRxのビットi
フェッチ	特権/非特権	セキュア/非セキュア	-	バスエラー
読出し	特権/非特権	セキュア/非セキュア	-	すべてのビットについてOK
書込み	特権	セキュア	-	すべてのビットについてOK
		非セキュア	非セキュア	ビットiについてOK
		非セキュア	セキュア	ビットiについてWI
	非特権	セキュア/非セキュア	-	すべてのビットがWI

表77.TrustZoneが無効（TZEN = 0）な場合のPRIVyBBRxレジスタへのアクセス

アクセスタイプ		PRIVyBBRx
フェッチ	特権/非特権	バスエラー
読出し	特権/非特権	OK
書込み	特権	OK
	非特権	WI

7.8 FLASH割込み

表78.Flash割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イベントフラグ/割込みのクリア方法	割込み有効制御ビット	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
Flash_S	セキュア操作終了	セキュアEOP(1)	セキュアEOP = 1を書き込む	セキュアEOPIE	あり	なし
	セキュア操作エラー	セキュアOPERR(2)	セキュアOPERR = 1を書き込む	セキュアERRIE	あり	なし
FLASH	非セキュア操作終了	非セキュアEOP (1)	非セキュアEOP = 1を書き込む	非セキュアEOPIE	あり	なし
	非セキュア操作エラー	非セキュアOPERR (2)	非セキュアOPERR = 1を書き込む	非セキュアERRIE	あり	なし
	ECC訂正	ECCC	ECCC = 1を書き込む	ECCCIE	あり	なし

1. セキュアEOP（または非セキュアEOP）は、セキュアEOPIE（または非セキュア EOPIE）がセットされている場合のみセットされます。
2. セキュアOPERR（または非セキュアOPERR）は、セキュアERRIE（または非セキュアERRIE）がセットされている場合のみセットされます。

7.9 Flashレジスタ

7.9.1 Flashアクセス制御レジスタ（Flash_ACR）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：ノーウェイトステート（進行中のFlashメモリ読出しがない場合）、ワード、ハーフワード、バイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	PDREQ2	PDREQ1	LPM	Res.	Res.	PRFTEN	Res.	Res.	Res.	Res.	LATENCY[3:0]			
	rw	rs	rs	rw			rw					rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **SLEEP_PD**：SLEEPモード時のFlashメモリをパワーダウンモードにする

このビットは、デバイスがSLEEPモードである場合に、Flashメモリをパワーダウンモードにするか、アイドルモードにするかを決定します。

0：SLEEPモード時のFlashメモリをアイドルモードにする

1：SLEEPモード時のFlashメモリをパワーダウンモードにする

注意： プログラムまたは消去操作の進行中にFlashメモリをパワーダウンモードにしないでください。

ビット13 **PDREQ2**：バンク2パワーダウンモードリクエスト

このビットはFLASH_PDKEY2Rで書込み保護されています。このビットでは、パワーダウンモードに移行するためにバンク2を要求します。バンク2がパワーダウンモードに移行すると、このビットはハードウェアによってクリアされ、PDKEY2Rはロックされます。

0：バンク2に対するパワーダウンモードへの移行要求なし

1：バンク2に対してパワーダウンモードへの移行を要求する

ビット12 **PDREQ1**：バンク1パワーダウンモードリクエスト

このビットはFLASH_PDKEY1Rで書込み保護されています。このビットでは、パワーダウンモードに移行するためにバンク1を要求します。バンク1がパワーダウンモードに移行すると、このビットはハードウェアによってクリアされ、PDKEY1Rはロックされます。

0：バンク1に対するパワーダウンモードへの移行要求なし

1：バンク1に対してパワーダウンモードへの移行を要求する

ビット11 **LPM**：低電力読出しモード

このビットにより、Flashメモリは低電力読出しモードになります。

0：Flashメモリを低電力読出しモードにしない

1：Flashメモリを低電力読出しモードにする

ビット10:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PRFTEN**：プリフェッチ有効化

このビットは、内蔵Flashメモリのプリフェッチバッファを有効にします。

0：プリフェッチ無効

1：プリフェッチ有効

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **LATENCY[3:0]**：遅延

これらのビットは、HCLK（AHBクロック）周期とFlashメモリアクセス時間との間の割合を表します。

0000：0ウェイトステート

0001：1ウェイトステート

0010：2ウェイトステート

...

1111：15ウェイトステート

7.9.2 FLASH非セキュアキーレジスタ（FLASH_NSKEYR）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：1ウェイトステート、ワードアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSKEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NSKEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **NSKEY[31:0]**：非セキュアなFlashメモリキー

FLASH_NSCRレジスタをアンロックし、Flashメモリの非セキュアプログラミング/消去操作を許可するには、次の値を連続して書き込む必要があります。

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

7.9.3 FLASHセキュアキーレジスタ（FLASH_SECKEYR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：1ウェイトステート、ワードアクセス

このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。

このレジスタは、FLASH_PRIVCFGRレジスタのSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SECKEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SECKEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **SECKEY[31:0]**：セキュアなFlashメモリキー

FLASH_SECCRレジスタをアンロックし、Flashメモリのセキュアプログラミング/消去操作を許可するには、次の値を連続して書き込む必要があります。

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

7.9.4 Flashオプションキーレジスタ（Flash_OPTKEYR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：1ウェイトステート、ワードアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OPTKEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTKEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **OPTKEY[31:0]**：オプションバイトキー

FLASH_OPTRレジスタをアンロックし、オプションバイトのプログラミングや消去操作を許可するには、次の値を連続して書き込む必要があります。

KEY1：0x0819 2A3B

KEY2：0x4C5D 6E7F

7.9.5 FLASHバンク1パワーダウンキーレジスタ（FLASH_PDKEY1R）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PDKEY1 [31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDKEY1 [15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **PDKEY1[31:0]**：バンク1パワーダウンキー

FLASH_ACRのPDREQ1ビットをアンロックするには、次の値を連続して書き込む必要があります。

PDKEY1_1：0x0415 2637

PDKEY1_2：0xFAFB FCFD

7.9.6 FLASHバンク2パワーダウンキーレジスタ（FLASH_PDKEY2R）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PDKEY2 [31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDKEY2 [15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **PDKEY2[31:0]**：バンク2パワーダウンキー

FLASH_ACRのPDREQ2ビットをアンロックするには、次の値を連続して書き込む必要があります。
PDKEY2_1：0x4051 6273
PDKEY2_2：0xAFBF CFDF

7.9.7 FLASH非セキュア･ステータス･レジスタ（FLASH_NSSR）

アドレスオフセット：0x20

リセット値：0x000X 0000

アクセス：ノーウェイトステート、ワード、ハーフワード;およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD2	PD1	OEM2LOCK	OEM1LOCK	WDW	BSY
										r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OPTWERR	Res.	Res.	Res.	Res.	Res.	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
		rc_w1						rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **PD2**：パワーダウンモードのバンク2

このビットは、Flashメモリバンク2がパワーダウン状態にあることを示します。バンク2が通常モードに移行するか呼び起こされると、リセットされます。

ビット20 **PD1**：パワーダウンモードのバンク1

このビットは、Flashメモリバンク1がパワーダウン状態にあることを示します。バンク1が通常モードに移行するか呼び起こされると、リセットされます。

ビット19 **OEM2LOCK**：OEM2ロック

このビットは、OBL中のOEM2 RDPキー読出しが未使用ではないことを示します。このビットがセットされると、OEM2 RDPロック機構がアクティブになります。

ビット18 **OEM1LOCK**：OEM1ロック

このビットは、OBL中のOEM1 RDPキー読出しが未使用ではないことを示します。このビットがセットされると、OEM1 RDPロック機構がアクティブになります。

ビット17 **WDW**：非セキュア書込みデータ待ち

このビットは、Flashメモリの書込みバッファがセキュアまたは非セキュア操作で書き込まれていることを示します。これは、最初のデータがバッファに格納されたときにセットされ、書込みがFlashメモリで行われると、クリアされます。

ビット16 **BSY**：非セキュアビジー

これは、セキュアまたは非セキュアなFlash操作が進行中であることを示します。このビットはFlash操作の開始時にセットされ、操作が終了するかエラーが発生するとリセットされます。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **OPTWERR**：オプション書込みエラー

このビットは、オプションバイトが無効な設定で書き込まれた場合にハードウェアによってセットされます。1を書き込むことによってクリアされます。

エラーフラグセットのすべての条件は、セクション7.3.9を参照してください。

ビット12:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **PGSERR**：非セキュアプログラミングシーケンスエラー

このビットは、プログラミングシーケンスが正しくない場合に、ハードウェアによってセットされます。1を書き込むことによってクリアされます。エラーフラグセットのすべての条件は、セクション7.3.9を参照してください。

ビット6 **SIZERR**：非セキュアサイズエラー

このビットは、非セキュアプログラムシーケンス中のアクセスのサイズがバイトまたはハーフワードである場合にハードウェアによってセットされます。クワッドワードプログラミングのみが連続ワードアクセスによって許可されます。このビットは、1を書き込むことによってクリアされます。

ビット5 **PGAERR**：非セキュアプログラミング配置エラー

このビットは、プログラムする最初のワードがクワッドワードアドレスと合っていないか、2番目、3番目、または4番目のワードが同じクワッドワードアドレスに属していない場合に、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。

ビット4 **WRPERR**：非セキュア書込み保護エラー

このビットは、消去/プログラムする非セキュア·アドレスが（WRP、HDP、またはRDPレベル1によって）Flashメモリの書込み保護された部分のアドレスである場合、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。エラーフラグセットのすべての条件は、セクション7.3.9を参照してください。

ビット3 **PROGERR**：非セキュアプログラミングエラー

このビットは、書き込むデータがすべて0である場合を除き、プログラムする前に、プログラム対象の非セキュアクワッドワードアドレスにすべて1以外の値が含まれる場合、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OPERR**：非セキュア操作エラー

このビットは、非セキュアなFlashメモリ操作（プログラム/消去）が異常終了すると、ハードウェアによってセットされます。このビットは、非セキュアなエラー割込みが有効になっている場合（NSERRIE = 1）にのみセットされます。このビットは、1を書き込むことによってクリアされます。

ビット0 **EOP**：非セキュア操作終了

このビットは、1つ以上の非セキュアなFlashメモリ操作（プログラム/消去）が正常に完了すると、ハードウェアによってセットされます。このビットは、非セキュア操作終了割込みが有効になっている場合（FLASH_NSCRのEOPIE = 1）にのみセットされます。このビットは、1を書き込むことによってクリアされます。

7.9.8 FLASHセキュア･ステータス･レジスタ（Flash_SECSR）

アドレスオフセット：0x24

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード;およびバイトアクセス

このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRレジスタのSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDW	BSY
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
								rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **WDW**：セキュア書込みデータ待ち

このビットは、Flashメモリの書込みバッファがセキュアまたは非セキュア操作で書き込まれていることを示します。これは、最初のデータがバッファに格納されたときにセットされ、書込みがFlashメモリで行われると、クリアされます。

ビット16 **BSY**：セキュアビジー

このビットは、セキュアまたは非セキュアなFlash操作が進行中であることを示します。Flash操作の開始時にセットされ、操作が終了するかエラーが発生するとリセットされます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **PGSERR**：セキュアプログラミングシーケンスエラー

このビットは、プログラミングシーケンスが正しくない場合に、ハードウェアによってセットされます。1を書き込むことによってクリアされます。エラーフラグセットのすべての条件は、セクション7.3.9を参照してください。

ビット6 **SIZERR**：セキュアサイズエラー

このビットは、セキュアプログラムシーケンス中のアクセスのサイズがバイトまたはハーフワードである場合にハードウェアによってセットされます。クワッドワードプログラミングのみが連続ワードアクセスによって許可されます。このビットは、1を書き込むことによってクリアされます。

ビット5 **PGAERR**：セキュアプログラミング配置エラー

このビットは、プログラムする最初のワードがクワッドワードアドレスと合っていないか、2番目、3番目、または4番目のワードが同じクワッドワードアドレスに属していない場合に、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。

ビット4 **WRPERR**：セキュア書込み保護エラー

このビットは、消去/プログラムするセキュア·アドレスが（WRP、HDP、またはRDPレベル1によって）Flashメモリの書込み保護された部分のアドレスである場合、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。エラーフラグセットのすべての条件は、セクション7.3.9を参照してください。

ビット3 **PROGERR**：セキュアプログラミングエラー

このビットは、書き込むデータがすべて0である場合を除き、プログラムする前に、プログラム対象のセキュアクワッドワードアドレスにすべて1以外の値が含まれる場合、ハードウェアによってセットされます。このビットは、1を書き込むことによってクリアされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OPERR**：セキュア操作エラー

このビットは、セキュアなFlashメモリ操作（プログラム/消去）が異常終了すると、ハードウェアによってセットされます。このビットは、セキュアエラー割込みが有効になっている場合（SECERRIE = 1）にのみセットされます。このビットは、1を書き込むことによってクリアされます。

ビット0 **EOP**：セキュア操作終了

このビットは、1つ以上のセキュアなFlashメモリ操作（プログラム/消去）が正常に完了すると、ハードウェアによってセットされます。このビットは、セキュア操作終了割込みが有効になっている場合（FLASH_SECCRのEOPIE = 1）にのみセットされます。このビットは、1を書き込むことによってクリアされます。

7.9.9 FLASH非セキュア制御レジスタ（FLASH_NSCR）

アドレスオフセット：0x28

リセット値：0xC000 0000

アクセス：ノーウェイトステート（進行中のFlashメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

このレジスタは、BSYまたはOBL_LAUNCHがリセットされたときにのみ書き込むことができます。それ以外の場合、書込みアクセスはBSYビットがリセットされるまでストールします。

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPTLOCK	Res.	Res.	OBL_LAUNCH	Res.	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	Res.	OPTSTRT	STRT
rs	rs			rc_w1		rw	rw							rs	rs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MER2	BWR	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **LOCK**：非セキュアロック

このビットはセット専用です。このビットがセットされると、Flash_NSCRレジスタはロックされます。Flash_NSKEYRレジスタのアンロックシーケンスが検出されると、ハードウェアによってクリアされます。アンロック操作が成功しない場合には、このビットは次のシステムリセットまでセットされたままとなります。

ビット30 **OPTLOCK**：オプションロック

このビットはセット専用です。このビットがセットされると、FLASH_NSCRレジスタのユーザオプションに関するすべてのビットがロックされます。このビットは、アンロックシーケンスが検出されると、ハードウェアによってクリアされます。FLASH_NSCRのLOCKビットは、OPTLOCKビットに対してアンロック･シーケンスを行う前にクリアする必要があります。

アンロック操作が成功しない場合には、このビットは次のリセットまでセットされたままとなります。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **OBL_LAUNCH**：オプションバイトローディングの強制

1にセットすると、このビットではオプションバイトの再ロードを強制的に行います。このビットは、オプションバイトローディングが完了したときにのみクリアされます。OPTLOCKがセットされた場合に書き込めません。

0：オプションバイトローディングが完了している

1：オプションバイトローディングがリクエストされている

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25 **ERRIE**：非セキュアエラー割込みイネーブル

このビットは、FLASH_NSSRのOPERR = 1の場合の割込み生成を有効にします。

0：非セキュアOPERRエラー割込み無効

1：非セキュアOPERRエラー割込み有効

ビット24 **EOPIE**：非セキュア操作終了割込み有効化

このビットは、FLASH_NSSRのEOP = 1の場合の割込み生成を有効にします。

0：非セキュアEOP割込み無効

1：非セキュアEOP割込み有効

ビット23:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **OPTSTRT**：オプション変更開始

このビットがセットされると、オプション操作がトリガされます。OPTLOCKビットがセットされている場合は書き込むことができません。このビットはソフトウェアによってのみセットされ、Flash_NSSRのBSYビットのクリア時にクリアされます。

ビット16 **STRT**：非セキュア開始

このビットがセットされると、非セキュアな消去操作がトリガされます。MER1、MER2、およびPERビットがリセットされ、STRTビットがセットされると、FLASH_NSSRのPGSERRがセットされます（この状態は禁じられています）。

このビットは、ソフトウェアによってのみセットされ、BSYがFLASH_NSSRでクリアされるとクリアされます。

ビット15 **MER2**：非セキュアバンク2全体消去

このビットがセットされると、バンク2の非セキュア全体消去（バンク2の全ユーザページ）がトリガされます。

ビット14 **BWR**：非セキュアバースト書込みプログラミングモード

このビットがセットされると、バースト書込みプログラミングモードが選択されます。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **BKER**：ページ消去のための非セキュアバンク選択

0：非セキュアページ消去のためにバンク1を選択する

1：非セキュアページ消去のためにバンク2を選択する

ビット10:3 **PNB[7:0]**：非セキュアページ番号選択

これらのビットは、消去するページを選択します。

00000000：ページ0

00000001：ページ1

...

00011111：ページ31（STM32U535/545の場合は上位ページ）

...

01111111：ページ127（STM32U575/585の場合は上位ページ）

...

11111111：ページ255（STM32U59x/5Ax/5Fx/5Gxの場合は上位ページ）

ビット2 **MER1**：非セキュアバンク1全体消去

このビットがセットされると、バンク1の非セキュア全体消去（バンク1の全ユーザページ）がトリガされます。

ビット1 **PER**：非セキュアページ消去

0：非セキュアページ消去無効

1：非セキュアページ消去有効

ビット0 **PG**：非セキュアプログラミング

0：非セキュアFLASHプログラミング無効

1：非セキュアFLASHプログラミング有効

7.9.10 Flashセキュア制御レジスタ（Flash_SECCR）

アドレスオフセット：0x2C

リセット値：0x8000 0000

アクセス：ノーウェイトステート（進行中のFlashメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

このレジスタは、BSYまたはOBL_LAUNCHがリセットされたときにのみ書き込むことができます。それ以外の場合、書込みアクセスはBSYビットがリセットされるまでストールします。

このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRレジスタのSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	INV	Res.	Res.	Res.	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STRT
rs		rw				rw	rw								rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MER2	BWR	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **LOCK**：セキュアロック

このビットはセット専用です。このビットがセットされると、このレジスタはロックされます。Flash_SECKEYRレジスタのアンロックシーケンスが検出されると、ハードウェアによってクリアされます。

アンロック操作が成功しない場合には、このビットは次のシステムリセットまでセットされたままとなります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 **INV：**Flashメモリのセキュリティ状態の反転

このビットは、Flashメモリのセキュリティ状態を反転します。

ビット28:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **ERRIE**：セキュアエラー割込みイネーブル

このビットは、FLASH_SECSRのOPERR = 1の場合の割込み生成を有効にします。

0：セキュアOPERRエラー割込み無効

1：セキュアOPERRエラー割込み有効

ビット24 **EOPIE**：セキュア操作終了割込み有効化

このビットは、FLASH_SECSRのEOP = 1の場合の割込み生成を有効にします。

0：セキュアEOP割込み無効

1：セキュアEOP割込み有効

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **STRT**：セキュア開始

このビットがセットされると、セキュアな消去操作がトリガされます。MER1、MER2、およびPERビットがリセットされ、STRTビットがセットされると、FLASH_SECSRのPGSERRがセットされます（この状態は禁じられています）。

このビットは、ソフトウェアによってのみセットされ、BSYがFLASH_SECSRでクリアされるとクリアされます。

ビット15 **MER2**：セキュアバンク2全体消去

このビットがセットされると、バンク2のセキュア全体消去（バンク2の全ユーザページ）がトリガされます。

ビット14 **BWR**：セキュアバースト書込みプログラミングモード

このビットがセットされると、バースト書込みプログラミングモードが選択されます。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **BKER**：ページ消去のためのセキュアバンク選択

0：セキュアページ消去のためにバンク1を選択する

1：セキュアページ消去のためにバンク2を選択する

ビット10:3 **PNB[7:0]**：セキュアページ番号選択

これらのビットは、消去するページを選択します。

00000000：ページ0

00000001：ページ1

...

00011111：ページ31（STM32U535/545の場合は上位ページ）

...

01111111：ページ127（STM32U575/585の場合は上位ページ）

...

11111111：ページ255（STM32U59x/5Ax/5Fx/5Gxの場合は上位ページ）

ビット2 **MER1**：セキュアバンク1全体消去

このビットがセットされると、バンク1のセキュア全体消去（バンク1の全ユーザページ）がトリガされます。

ビット1 **PER**：セキュアページ消去

0：セキュアページ消去無効

1：セキュアページ消去有効

ビット0 **PG**：セキュアプログラミング

0：セキュアFLASHプログラミング無効

1：セキュアFLASHプログラミング有効

7.9.11 FlashECCレジスタ（Flash_ECCR）

アドレスオフセット：0x30

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCIE	Res.	SYSF_ECC	BK_ECC	ADDR_ECC[20:16]				
rc_w1	rc_w1						rw		r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_ECC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31 **ECCD**：ECC検出

このビットは、2つのECCエラーが検出されたときに、ハードウェアによってセットされます（ECCCおよびECCDが以前にクリアされている場合のみ）。このビットがセットされた場合、NMIが生成されます。このビットは、1を書き込むことによってクリアされます。

ビット30 **ECCC**：ECCエラー訂正

このビットは、1つのECCエラーが検出および訂正されたときに、ハードウェアによってセットされます（ECCCおよびECCDが以前にクリアされている場合のみ）。ECCIEがセットされると、割込みが生成されます。このビットは、1を書き込むことによってクリアされます。

ビット29:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ECCIE**：ECC訂正割込み有効化

このビットは、FLASH_ECCRレジスタのECCCビットがセットされている場合に割込み生成を有効にします。

0：ECCC割込み無効

1：ECCC割込み有効

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **SYSF_ECC**：システムFlashメモリECC失敗

このビットは、ECCエラー訂正またはダブルECCエラー検出がシステムFlashメモリで発生したことを示します。

ビット21 **BK_ECC**：ECC失敗バンク

このビットは、ECCエラー訂正またはダブルECCエラー検出が関連しているバンクを示します。

0：バンク1

1：バンク2

ビット20:0 **ADDR_ECC[20:0]**：ECC失敗アドレス

このフィールドは、ECCエラー訂正またはダブルECCエラー検出が関連しているアドレスを示します。アドレスは、アドレス0x0 0000から次のアドレスに、バンクによって与えられます。

0x7FFF0：STM32U535/545の上位アドレス

0xFFFF0：STM32U575/585の上位アドレス

0x1FFFF0：STM32U59x/5Ax/5Fx/5Gxの上位アドレス

7.9.12 FLASH操作ステータスレジスタ（FLASH_OPSR）

アドレスオフセット：0x34

リセット値：0xX0XX XXXX

（システムリセット後0xX0XX XXXX、およびパワーオンリセット後0x0000 0000）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CODE_OP[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	SYSF_OP	BK_OP	ADDR_OP[20:16]				
r	r	r							r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_OP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:29 **CODE_OP[2:0]**：Flashメモリ操作コード

このフィールドは、システムリセットによって中断されたFlashメモリ操作を示します。

000：前回のリセットによって中断されたFlash操作なし

001：1回の書込み操作の中断

010：バースト書込み操作の中断

011：ページ消去操作の中断

100：バンク消去操作の中断

101：全体消去操作の中断

110：オプション変更操作の中断

111：予約済み

ビット28:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **SYSF_OP**：システムFlashメモリ操作の中断

このビットは、システムFlashメモリの操作中にリセットが発生したことを示します。

ビット21 **BK_OP**：中断された操作バンク

このビットは、リセット発生時にアクセスしたFlashメモリバンクを示します。

0：バンク1

1：バンク2

ビット20:0 **ADDR_OP[20:0]**：中断された操作アドレス

このフィールドは、リセット発生時にアクセスしたFlashメモリのアドレスを示します。アドレスは、アドレス0x0 0000から次のアドレスに、バンクによって与えられます。

0x7FFF0：STM32U535/545の上位アドレス

0xFFFF0：STM32U575/585の上位アドレス

0x1FFFF0：STM32U59x/5Ax/5Fx/5Gxの上位アドレス

7.9.13 Flashオプションレジスタ（Flash_OPTR）

アドレスオフセット：0x40

リセット値：0xXXXX XXXX（ビット0~31には、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値：0x1FEF：F8AA

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス。

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TZEN	IO_VDDIO2_HSLV	IO_VDD_HSLV	PA15_PUPEN	NBOOT0	NSWBOOT0	SRAM2_RST	SRAM2_ECC	SRAM3_ECC	BKPRAM_ECC	DUALBANK	SWAP_BANK	WWDG_SW	IWDG_STDBY	IWDG_STOP	IWDG_SW
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SRAM_RST	NRST_SHDW	NRST_STDBY	NRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **TZEN**：TrustZoneグローバルセキュリティ有効化
0：TrustZoneグローバルセキュリティ無効
1：TrustZoneグローバルセキュリティ有効

ビット30 **IO_VDDIO2_HSLV**：低いV_{DDIO2}電圧設定ビットでのハイスピードIO。
このビットは、V_{DDIO2}が2.5V未満の場合にのみセットできます。
0：低いV_{DDIO2}電圧でのハイスピードIO機能は無効（V_{DDIO2}は2.5Vを超える場合がある）
1：低いV_{DDIO2}電圧でのハイスピードIO機能は有効（V_{DDIO2}は2.5V未満で保持される）

ビット29 **IO_VDD_HSLV**：低いV_{DD}電圧設定ビットでのハイスピードIO
このビットは、V_{DD}が2.5V未満の場合にのみセットできます。
0：低いV_{DD}電圧でのハイスピードIO機能は無効（V_{DD}は2.5Vを超える場合がある）
1：低いV_{DD}電圧でのハイスピードIO機能は有効（V_{DD}は2.5V未満で保持される）

ビット28 **PA15_PUPEN**：PA15プルアップ有効化
0：USB電源供給デッドバッテリは有効、TDIプルアップは無効
1：USB電源供給デッドバッテリは無効、TDIプルアップは有効

ビット27 **NBOOT0**：NBOOT0オプションビット
0：NBOOT0 = 0
1：NBOOT0 = 1

ビット26 **NSWBOOT0**：ソフトウェアBOOT0
0：オプションビットNBOOT0から取得したBOOT0
1：PH3/BOOT0ピンから取得したBOOT0

ビット25 **SRAM2_RST**：システムリセット時のSRAM2消去
0：システムリセットが発生した際にSRAM2を消去する
1：システムリセットが発生した際にSRAM2を消去しない

ビット24 **SRAM2_ECC**：SRAM2ECC検出および訂正有効化
0：SRAM2ECCチェック有効
1：SRAM2ECCチェック無効

ビット23 **SRAM3_ECC**：SRAM3ECC検出および訂正有効化
0：SRAM3ECCチェック有効
1：SRAM3ECCチェック無効
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。

ビット22 **BKPRAM_ECC**：バックアップRAM ECC検出および訂正有効化
0：バックアップRAM ECCチェック有効
1：バックアップRAM ECCチェック無効

ビット21 **DUALBANK**：デュアルバンク設定
– STM32U59x/5Ax/5Fx/5Gxの場合、2MBのFlashメモリデバイス
– STM32U575/585の場合、1MBのFlashメモリデバイス
– STM32U535/545の場合、256KBおよび128KBのFlashメモリデバイス
0：バンク1に連続アドレスがあるシングルバンクFlashメモリ
1：連続アドレスがあるデュアルバンクFlashメモリ

ビット20 **SWAP_BANK**：バンクのスワップ
0：バンク1とバンク2のアドレスをスワップしない
1：バンク1とバンク2のアドレスをスワップする

ビット19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択
0：ハードウェアによるウィンドウ型ウォッチドッグを選択する
1：ソフトウェアによるウィンドウ型ウォッチドッグを選択する

ビット18 **IWDG_STDBY**：STANDBYモードでの独立型ウォッチドッグカウンタの停止
0：STANDBYモードで独立型ウォッチドッグカウンタを停止する
1：STANDBYモードで独立型ウォッチドッグカウンタを動作する

ビット17 **IWDG_STOP**：STOPモードでの独立型ウォッチドッグカウンタの停止
0：STOPモードで独立型ウォッチドッグカウンタを停止する
1：STOPモードで独立型ウォッチドッグカウンタを動作する

ビット16 **IWDG_SW**：独立型ウォッチドッグの選択
0：ハードウェアによる独立型ウォッチドッグを選択する
1：ソフトウェアによる独立型ウォッチドッグを選択する

ビット15 **SRAM_RST**：システムリセット時のすべてのSRAM（SRAM2およびBKPSRAMを除く）の消去
0：システムリセットが発生した際に、すべてのSRAM（SRAM2およびBKPSRAMを除く）を消去する
1：システムリセットが発生した際に、すべてのSRAM（SRAM2およびBKPSRAMを除く）を消去しない

ビット14 **NRST_SHDW**：SHUTDOWNモードでのリセット生成
0：SHUTDOWNモードに移行するときにリセットを生成する
1：SHUTDOWNモードに移行するときにリセットを生成しない

ビット13 **NRST_STDBY**：STANDBYモードでのリセット生成
0：STANDBYモードに移行するときにリセットを生成する
1：STANDBYモードに移行するときにリセットを生成しない

ビット12 **NRST_STOP**：STOPモードでのリセット生成
0：STOPモードに移行するときにリセットを生成する
1：STOPモードに移行するときにリセットを生成しない

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **BOR_LEV[2:0]**：BORリセットレベル

これらのビットには、リセットをアクティブにしたりリセットを解除したりするV_{DD}供給レベル閾値が含まれています。

000：BORレベル0（約1.7Vのリセットレベル閾値）

001：BORレベル1（約2.0Vのリセットレベル閾値）

010：BORレベル2（約2.2Vのリセットレベル閾値）

011：BORレベル3（約2.5Vのリセットレベル閾値）

100：BORレベル4（約2.8Vのリセットレベル閾値）

ビット7:0 **RDP[7:0]**：読出し保護レベル

0xAA：レベル0（読出し保護は非アクティブ）

0x55：レベル0.5（読出し保護は非アクティブ、非セキュアデバッグアクセスのみ可能）。TrustZoneが有効（TZEN = 1）な場合のみ使用可能

0xCC：レベル2（チップ読出し保護はアクティブ）

その他：レベル1（メモリ読出し保護はアクティブ）

注： 詳細については、セクション7.6.2を参照してください。

7.9.14 FLASH非セキュアブートアドレス0レジスタ（FLASH_NSBOOTADD0R）

アドレスオフセット：0x44

リセット値：0xXXXX XXXX

（リセット解除時にFlashメモリからの値が入力されたオプションバイト）

ST出荷時の値：0x0800 007F

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス。

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSBOOTADD0 [24:9]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NSBOOTADD0 [8:0]									Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw							

ビット31:7 **NSBOOTADD0[24:0]**：非セキュアブートベースアドレス0

非セキュアブートメモリアドレスは、有効アドレス範囲内の任意のアドレスに対して128バイト単位で設定できます。これらのビットは、アドレス[31:7]に対応します。NSBOOTADD0オプションバイトは、BOOT0ピンまたはNSWBOOT0の状態に応じて選択されます。

例：

NSBOOTADD0[24:0] = 0x0100000：非セキュアFlashメモリからのブート（0x0800 0000）

NSBOOTADD0[24:0] = 0x017F200：システムメモリブートローダからのブート（0x0BF9 0000）

NSBOOTADD0[24:0] = 0x0400000：Sバスの非セキュアSRAM1からのブート（0x2000 0000）

ビット6:0 予約済みであり、リセット値に保持する必要があります。

7.9.15 FLASH非セキュアブートアドレス1レジスタ（FLASH_NSBOOTADD1R）

アドレスオフセット：0x48
リセット値：0xXXXX XXXX
（リセット解除時にFlashメモリからの値が入力されたオプションバイト）

ST出荷時の値：0x0BF9 007F

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス。

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSBOOTADD1 [24:9]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NSBOOTADD1 [8:0]									Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw							

ビット31:7 **NSBOOTADD1[24:0]**：非セキュアブートアドレス1

非セキュアブートメモリアドレスは、有効アドレス範囲内の任意のアドレスに対して128バイト単位で設定できます。これらのビットは、アドレス[31:7]に対応します。NSBOOTADD1オプションバイトは、BOOT0ピンまたはNSWBOOT0の状態に応じて選択されます。

例：

NSBOOTADD1[24:0] = 0x0100000：非セキュアFlashメモリからのブート（0x0800 0000）

NSBOOTADD1[24:0] = 0x017F200：システムメモリからのブートローダ（0x0BF9 0000）

NSBOOTADD1[24:0] = 0x0400000：Sバスの非セキュアSRAM1からのブート（0x2000 0000）

ビット6:0 予約済みであり、リセット値に保持する必要があります。

7.9.16 FLASHセキュアブートアドレス0レジスタ（FLASH_SECBOOTADD0R）

アドレスオフセット：0x4C

リセット値：0xXXXX XXXX

ST出荷時の値：0x0C00：007C

（リセット解除時にFlashメモリからの値が入力されたオプションバイト）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス。

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRのSPRIV = 1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SECBOOTADD0 [24:9]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SECBOOTADD0 [8:0]									Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK
rw	rw	rw	rw	rw	rw	rw	rw	rw							rw

ビット31:7 **SECBOOTADD0[24:0]**：セキュアブートベースアドレス0

セキュアなブートメモリアドレスは、有効アドレス範囲内の任意のアドレスに対して128バイト単位で設定できます。このビットは、アドレス[31:7]に対応します。SECBOOTADD0オプションバイトは、BOOT0ピンまたはNSWBOOT0の状態に応じて選択されます。

例：

SECBOOTADD0[24:0] = 0x0180000：セキュアFlashメモリからのブート（0x0C00 0000）

SECBOOTADD0[24:0] = 0x01FF000：RSSからのブート（0x0FF8 0000）

SECBOOTADD0[24:0] = 0x0600000：SバスのセキュアSRAM1からのブート（0x3000 0000）

ビット6:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **BOOT_LOCK**：ブートロック

このビットがセットされると、ブートは、ブート選択オプションが何であろうとも常にSECBOOTADD0[24:0]オプションバイトに設定されたベースアドレス値に強制的に設定されます。このビットは、RDPがレベル0である場合にのみクリアできます。

7.9.17 FLASHのセキュアウォーターマーク1レジスタ1（FLASH_SECWM1R1）

アドレスオフセット：0x50

リセット値：0xXXXX XXXX

（ビットには、OBL時にFlashメモリからの値が入力されます。予約済みビットは1として読み出されます。）

ST出荷時の値： 0xFFFF FF80（STM32U535/545/575/585の場合）

0xFFFF FF00（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス。

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRのSPRIV = 1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM1_PEND [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM1_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **SECWM1_PEND[7:0]**：1番目のセキュア領域の最終ページ
このフィールドは、バンク1のセキュア領域の最終ページを含んでいます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SECWM1_PSTRT[7:0]**：1番目のセキュア領域の開始ページ
このフィールドは、バンク1のセキュア領域の最初のページを含んでいます。

7.9.18 FLASHのセキュアウォーターマーク1レジスタ2（FLASH_SECWM1R2）

アドレスオフセット：0x54

リセット値：0xXXXX XXXX

（OBL時にFlashメモリからの値が入力されたビット）

ST出荷時の値： 0x7FE07FE0（STM32U535/545の場合）

0x7F807F80（STM32U575/585の場合）

0x7F007F00（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRのSPRIV = 1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HDP1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP1_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31 **HDP1EN**：1番目の非表示保護領域有効化

0：1番目のHDP領域なし

1：1番目のHDP領域有効

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **HDP1_PEND[7:0]**：1番目の非表示保護領域の最終ページ

このフィールドは、バンク1のHDP領域の最終ページを含んでいます。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

7.9.19 FLASHのWRP1領域Aのアドレスレジスタ（FLASH_WRP1AR）

アドレスオフセット：0x58

リセット値：0xXXXX XXXX

（ビットには、OBL時にFlashメモリからの値が入力されます。予約済みビットは1として読み出されます。）

ST出荷時の値： 0xFFE0 FFFF（STM32U535/545の場合）

0xFF80 FFFF（STM32U575/585の場合）

0xFF00 FFFF（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UNLOCK**：バンク1の1番目の WRP領域Aのアンロック
0：WRP1Aの開始ページと終了ページをロックする
1：WRP1Aの開始ページと終了ページをアンロックする

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **WRP1A_PEND[7:0]**：バンク1の1番目のWRP領域Aの終了ページ
このフィールドは、バンク1の1番目のWRP領域の最終ページを含みます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **WRP1A_PSTRT[7:0]**：バンク1の1番目のWRP領域Aの開始ページ
このフィールドは、バンク1の1番目のWRP領域の最初のページを含みます。

7.9.20 FLASHのWRP1領域Bのアドレスレジスタ（FLASH_WRP1BR）

アドレスオフセット：0x5C

リセット値：0xXXXX XXXX（ビットには、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値： 0xFFE0 FFFF（STM32U535/545の場合）

0xFF80 FFFF（STM32U575/585の場合）

0xFF00 FFFF（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UNLOCK**：バンク1の2番目のWRP領域Bのアンロック
0：WRP1Bの開始ページと終了ページをロックする
1：WRP1Bの開始ページと終了ページをアンロックする

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **WRP1B_PEND[7:0]**：バンク1の2番目のWRP領域Bの終了ページ
このフィールドは、バンク1の2番目のWRP領域の最終ページを含みます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **WRP1B_PSTRT[7:0]**：バンク1の2番目のWRP領域Bの開始ページ
このフィールドは、バンク1の2番目のWRP領域の最初のページを含みます。

7.9.21 FLASHのセキュアウォーターマーク2レジスタ1（FLASH_SECWM2R1）

アドレスオフセット：0x60

リセット値：0xXXXX XXXX（ビットには、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値： 0xFFFF FF80（STM32U535/545/575/585の場合）

0xFFFF FF00（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRのSPRIV = 1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM2_PEND [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM2_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **SECWM2_PEND[7:0]**：2番目のセキュア領域の最終ページ
このフィールドは、バンク2のセキュア領域の最終ページを含んでいます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SECWM2_PSTRT[7:0]**：2番目のセキュア領域の開始ページ
このフィールドは、バンク2のセキュア領域の最初のページを含んでいます。

7.9.22 FLASHのセキュアウォーターマーク2レジスタ2（FLASH_SECWM2R2）

アドレスオフセット：0x64

リセット値：0xXXXX XXXX（ビットには、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値： 0x7FE07FE0（STM32U535/545の場合）

0x7F807F80（STM32U575/585の場合）

0x7F007F00（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRのSPRIV = 1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HDP2EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP2_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31 **HDP2EN**：2番目の非表示保護領域有効化
0：2番目のHDP領域なし
1：2番目のHDP領域有効

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **HDP2_PEND[7:0]**：2番目の非表示保護領域の最終ページ
HDP2_PENDはバンク2のHDP領域の最終ページを含んでいます。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

7.9.23 FLASHのWRP2領域Aのアドレスレジスタ（FLASH_WRP2AR）

アドレスオフセット：0x68

リセット値：0xXXXX XXXX（ビットには、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値： 0xFFE0 FFFF（STM32U535/545の場合）

0xFF80 FFFF（STM32U575/585の場合）

0xFF00 FFFF（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UNLOCK**：バンク2の1番目の WRP領域Aのアンロック
0：WRP2Aの開始ページと終了ページをロックする
1：WRP2Aの開始ページと終了ページをアンロックする

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **WRP2A_PEND[7:0]**：バンク2の1番目のWRP領域Aの終了ページ
このフィールドは、バンク2の1番目のWRP領域の最終ページを含みます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **WRP2A_PSTRT[7:0]**：バンク2の1番目のWRP領域Aの開始ページ
このフィールドは、バンク2の1番目のWRP領域の最初のページを含みます。

7.9.24 FLASHのWPR2領域Bのアドレスレジスタ（FLASH_WRP2BR）

アドレスオフセット：0x6C

リセット値：0xXXXX XXXX（ビットには、OBL時にFlashメモリからの値が入力されます。）

ST出荷時の値： 0xFFE0 FFFF（STM32U535/545の場合）

0xFF80 FFFF（STM32U575/585の場合）

0xFF00 FFFF（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは、OPTLOCKビットがセットされている場合は書き込むことができません。このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも読み書きができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_PEND [7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_PSTRT [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UNLOCK**：バンク2の2番目のWRP領域Bのアンロック
0：WRP2Bの開始ページと終了ページをロックする
1：WRP2Bの開始ページと終了ページをアンロックする

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **WRP2B_PEND[7:0]**：バンク2の2番目のWPR領域Bの終了ページ
このフィールドは、バンク2の2番目のWRP領域の最終ページを含みます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **WRP2B_PSTRT[7:0]**：バンク2の2番目のWPR領域Bの開始ページ
このフィールドは、バンク2の2番目のWRP領域の最初のページを含みます。

7.9.25 FLASHOEM1キーレジスタ1（FLASH_OEM1KEYR1）

アドレスオフセット：0x70

リセット値：0x0000 0000

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも書込みができます。このレジスタは0として読み出されます。FLASH_PRIVCFGRレジスタのNSPRIVが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEM1KEY [31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OEM1KEY [15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **OEM1KEY[31:0]**：OEM1キーの最下位バイト

7.9.26 FLASHOEM1キーレジスタ2（FLASH_OEM1KEYR2）

アドレスオフセット：0x74

リセット値：0x0000 0000

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも書込みができます。このレジスタは0として読み出されます。FLASH_PRIVCFGRレジスタのNSPRIVが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEM1KEY [63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OEM1KEY [47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **OEM1KEY[63:32]**：OEM1キーの最上位バイト

7.9.27 FLASHOEM2キーレジスタ1（FLASH_OEM2KEYR1）

アドレスオフセット：0x78

リセット値：0x0000 0000

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも書込みができます。このレジスタは0として読み出されます。FLASH_PRIVCFGRレジスタのNSPRIVが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEM2KEY [31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OEM2KEY [15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **OEM2KEY[31:0]**：OEM2キーの最下位バイト

7.9.28 FLASHOEM2キーレジスタ2（FLASH_OEM2KEYR2）

アドレスオフセット：0x7C

リセット値：0x0000 0000

アクセス：ウェイトステートなし（進行中のオプションバイト変更がない場合）、ワード、ハーフワード、およびバイトアクセス

このレジスタは非セキュアです。セキュアおよび非セキュアアクセスのどちらでも書込みができます。このレジスタは、FLASH_PRIVCFGRレジスタのNSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEM2KEY [63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OEM2KEY [47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **OEM2KEY[63:32]**：OEM2キーの最上位バイト

7.9.29 ブロックベースのFLASHセキュアバンク1レジスタx（FLASH_SECBB1Rx）

アドレスオフセット：0x80 + 0x4 * (x-1)(x = 1から8)

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタはセキュアです。セキュアアクセスによってのみ書込み可能です。このレジスタは、非特権アクセスから保護することができます（表75を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEC1BB31	SEC1BB30	SEC1BB29	SEC1BB28	SEC1BB27	SEC1BB26	SEC1BB25	SEC1BB24	SEC1BB23	SEC1BB22	SEC1BB21	SEC1BB20	SEC1BB19	SEC1BB18	SEC1BB17	SEC1BB16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC1BB15	SEC1BB14	SEC1BB13	SEC1BB12	SEC1BB11	SEC1BB10	SEC1BB9	SEC1BB8	SEC1BB7	SEC1BB6	SEC1BB5	SEC1BB4	SEC1BB3	SEC1BB2	SEC1BB1	SEC1BB0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SEC1BBi**：ページのセキュア/非セキュア属性（i = 31 ~ 0）

各ビットは、バンク1にページセキュリティ属性を1つ設定するために使用します。

0：ブロックベースセキュアではないバンク1のページ（32 *（x - 1）+ i）

1：ブロックベースセキュアであるバンク1のページ（32 *（x - 1）+ i）

7.9.30 ブロックベースのFLASHセキュアバンク2レジスタx（FLASH_SECBB2Rx）

アドレスオフセット：0xA0 + 0x4 * (x-1)(x = 1から8)

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタはセキュアです。セキュアアクセスによってのみ書込み可能です。このレジスタは、非特権アクセスから保護することができます（表75を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEC2BB31	SEC2BB30	SEC2BB29	SEC2BB28	SEC2BB27	SEC2BB26	SEC2BB25	SEC2BB24	SEC2BB23	SEC2BB22	SEC2BB21	SEC2BB20	SEC2BB19	SEC2BB18	SEC2BB17	SEC2BB16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC2BB15	SEC2BB14	SEC2BB13	SEC2BB12	SEC2BB11	SEC2BB10	SEC2BB9	SEC2BB8	SEC2BB7	SEC2BB6	SEC2BB5	SEC2BB4	SEC2BB3	SEC2BB2	SEC2BB1	SEC2BB0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SEC2BBi**：ページのセキュア/非セキュア属性（i = 31 ~ 0）

各ビットは、バンク2にページセキュリティ属性を1つ設定するために使用します。

0：ブロックベースセキュアではないバンク2のページ（32 *（x - 1）+ i）

1：ブロックベースセキュアであるバンク2のページ（32 *（x - 1）+ i）

7.9.31 FLASHセキュアHDP制御レジスタ（FLASH_SECHDPCR）

アドレスオフセット：0xC0

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタはセキュアです。セキュアアクセスによってのみ読み書きができます。非セキュア読出し/書込みアクセスはRAZ/WIです。このレジスタは、FLASH_PRIVCFGRレジスタのSPRIVビットが1の場合に非特権アクセスから保護することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP2_ACCDIS	HDP1_ACCDIS
														rs	rs

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **HDP2_ACCDIS**：HDP2領域へのアクセス無効化

このビットがセットされると、システムリセットによってのみクリアされます。

0：HDP2領域へのアクセスを許可

1：HDP2領域へのアクセスを拒否（SECWM2Ryオプションバイトの変更をブロック、特定のオプションバイトを変更する際の規則を参照）

ビット0 **HDP1_ACCDIS**：HDP1領域へのアクセス無効化

このビットがセットされると、システムリセットによってのみクリアされます。

0：HDP1領域へのアクセスを許可

1：HDP1領域へのアクセスを許可（SECWM1Ryオプションバイトの変更をブロック、特定のオプションバイトを変更する際の規則を参照）

7.9.32 FLASHの特権設定レジスタ（FLASH_PRIVCFGR）

アドレスオフセット：0xC4.

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、特権および非特権アクセスのどちらでも読み出すことができます。NSPRIVは非セキュア･ビットです。SPRIVはセキュア･ビットです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **NSPRIV**：非セキュアレジスタの特権保護

このビットは、特権または非特権、セキュアおよび非セキュアアクセスの両方で読み出すことができます。

0：非セキュアFlashレジスタは、特権または非特権アクセスで読み書きできます。

1：非セキュアFlashレジスタは、特権アクセスでのみ読み書きできます。

NSPRIVビットは、セキュアまたは非セキュア特権アクセスで書き込むことができます。NSPRIVビットへのセキュアまたは非セキュアな非特権書込みアクセスは無視されます。

ビット0 **SPRIV**：セキュアレジスタの特権保護

このビットはTrustZoneが有効（TZEN = 1）な場合のみアクセスできます。このビットは、特権または非特権、セキュアおよび非セキュアアクセスの両方で読み出すことができます。

0：セキュアFlashレジスタは、特権または非特権アクセスで読み書きできます。

1：セキュアFlashレジスタは、特権アクセスでのみ読み書きできます。

SPRIVビットはセキュア特権アクセスによってのみ書き込むことができます。SPRIVビットへの非セキュア書込みアクセスは無視されます。SPRIVビットへのセキュアな非特権書込みアクセスは無視されます。

7.9.33 ブロックベースのFLASH特権バンク1レジスタx（FLASH_PRIVBB1Rx）

アドレスオフセット：0xD0 + 0x4 * (x-1)(x = 1から8)

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは特権です。これは特権アクセスによってのみ読出し、書込みが可能です。このレジスタは、非セキュアアクセスから保護することができます（表76を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRIV1 BB31	PRIV1 BB30	PRIV1 BB29	PRIV1 BB28	PRIV1 BB27	PRIV1 BB26	PRIV1 BB25	PRIV1 BB24	PRIV1 BB23	PRIV1 BB22	PRIV1 BB21	PRIV1 BB20	PRIV1 BB19	PRIV1 BB18	PRIV1 BB17	PRIV1 BB16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV1 BB15	PRIV1 BB14	PRIV1 BB13	PRIV1 BB12	PRIV1 BB11	PRIV1 BB10	PRIV1 BB9	PRIV1 BB8	PRIV1 BB7	PRIV1 BB6	PRIV1 BB5	PRIV1 BB4	PRIV1 BB3	PRIV1 BB2	PRIV1 BB1	PRIV1 BB0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRIV1BBi**：ページ特権/非特権属性（i = 31 ~ 0）

各ビットは、バンク1にページ特権属性を1つ設定するために使用します。

0：非特権アクセスでアクセス可能なバンク1のページ（32 *（x - 1）+ i）

1：特権アクセスでのみアクセス可能なバンク1のページ（32 *（x - 1）+ i）

7.9.34 ブロックベースのFLASH特権バンク2レジスタx（FLASH_PRIVBB2Rx）

アドレスオフセット：0xF0 + 0x4 * (x-1)(x = 1から8)

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは特権です。これは特権アクセスによってのみ読出し、書込みが可能です。このレジスタは、非セキュアアクセスから保護することができます（表76を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRIV2BB31	PRIV2BB30	PRIV2BB29	PRIV2BB28	PRIV2BB27	PRIV2BB26	PRIV2BB25	PRIV2BB24	PRIV2BB23	PRIV2BB22	PRIV2BB21	PRIV2BB20	PRIV2BB19	PRIV2BB18	PRIV2BB17	PRIV2BB16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV2BB15	PRIV2BB14	PRIV2BB13	PRIV2BB12	PRIV2BB11	PRIV2BB10	PRIV2BB9	PRIV2BB8	PRIV2BB7	PRIV2BB6	PRIV2BB5	PRIV2BB4	PRIV2BB3	PRIV2BB2	PRIV2BB1	PRIV2BB0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRIV2BBi**：ページ特権/非特権属性（i = 31 ~ 0）

各ビットは、バンク2にページセキュリティ属性を1つ設定するために使用します。

0：非特権アクセスでアクセス可能なバンク2のページ（32 *（x - 1）+ i）

1：特権アクセスでのみアクセス可能なバンク2のページ（32 *（x - 1）+ i）

7.9.35 Flashレジスタマップ

表79.Flashレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEP_PD	PDREQ2	PDREQ1	LPM			PRFTEN					LATENCY [3:0]			
	リセット値																		0	0	0	0			0					0	0	0	0
0x04	予約済み	予約済み																															
0x08	Flash_NSKEYR	NSKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	Flash_SECKEYR	SECKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FLASH_OPTKEYR	OPTKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	予約済み	予約済み																															
0x18	FLASH_PDKEY1R	PDKEY1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	FLASH_PDKEY2R	PDKEY2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	Flash_NSSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD2	PD1	OEM2LOCK	OEM1LOCK	WDW	BSY	Res.	Res.	OPTWERR	Res.	Res.	Res.	Res.	Res.	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値											0	0	X	X	X	X			0						0	0	0	0	0		0	0

表79.Flashレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	Flash_SECSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDW	BSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値															0	0									0	0	0	0	0		0	0
0x28	Flash_NSCR	LOCK	OPTLOCK	Res.	Res.	OBL_LAUNCH	Res.	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	Res.	OPTSTRT	STRT	MER2	BWR	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
	リセット値	1	1			0		0	0							0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0
0x2C	Flash_SECCR	LOCK	Res.	INV	Res.	Res.	Res.	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STRT	MER2	BWR	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
	リセット値	1		0				0	0								0	0	0			0		0	0	0	0	0	0	0	0	0
0x30	FLASH_ECCR	ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCCIE	Res.	SYSF_ECC	BK_ECC	ADDR_ECC[20:0]																					
	リセット値	0	0						0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	FLASH_OPSR	CODE_OP[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	SYSF_OP	BK_OP	ADDR_OP[20:0]																					
	リセット値	X	X	X							X	X	X	X	x	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x38 ~0x3C	予約済み	予約済み																															
0x40	FLASH_OPTR	TZEN	IO_VDDIO2_HSLV	IO_VDD_HSLV	PA15_PUPEN	NBOOT0	NSWBOOT0	SRAM2_RST	SRAM2_ECC	SRAM3_ECC	BKPRAM_ECC	DUALBANK	SWAP_BANK	WWDG_SW	IWDG_STDBY	IWDG_STOP	IWDG_SW	SRAM_RST	NRST_SHDW	NRST_STDBY	NRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X
0x44	FLASH_ NSBOOTADD0R	NSBOOTADD0 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X							
0x48	FLASH_ NSBOOTADD1R	NSBOOTADD1 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X							
0x4C	FLASH_ SECBOOTADD0R	SECBOOTADD0 [24:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X							X
0x50	FLASH_ SECWM1R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM1_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM1_PSTRT [7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x54	FLASH_ SECWM1R2	HDP1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP1_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	X								X	X	X	X	X	X	X	X																
0x58	FLASH_WRP1AR	UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_PSTRT [7:0]							
	リセット値	X								X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X

表79.Flashレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	FLASH_WRP1BR	UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_PSTRT [7:0]							
	リセット値	X								X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x60	FLASH_ SECWM2R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM2_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECWM2_PSTRT [7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x64	FLASH_ SECWM2R2	HDP2EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP2_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	X								X	X	X	X	X	X	X	X																
0x68	FLASH_WRP2AR	UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_PSTRT [7:0]							
	リセット値	X								X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x6C	FLASH_WRP2BR	UNLOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_PEND [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_PSTRT [7:0]							
	リセット値	X								X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x70	FLASH_ OEM1KEYR1	OEM1KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	FLASH_ OEM1KEYR2	OEM1KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	FLASH_ OEM2KEYR1	OEM2KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	FLASH_ OEM2KEYR2	OEM2KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x80 + 0x4 * (x - 1)、(x = 1 ~ 8) 最後のアドレス：0x9C	FLASH_ SECB1BRx	SEC1BB31	SEC1BB30	SEC1BB29	SEC1BB28	SEC1BB27	SEC1BB26	SEC1BB25	SEC1BB24	SEC1BB23	SEC1BB22	SEC1BB21	SEC1BB20	SEC1BB19	SEC1BB18	SEC1BB17	SEC1BB16	SEC1BB15	SEC1BB14	SEC1BB13	SEC1BB12	SEC1BB11	SEC1BB10	SEC1BB9	SEC1BB8	SEC1BB7	SEC1BB6	SEC1BB5	SEC1BB4	SEC1BB3	SEC1BB2	SEC1BB1	SEC1BB0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA0 + 0x4 * (x - 1)、(x = 1 ~ 8) 最後のアドレス：0xBC	FLASH_ SECBB2Rx	SEC2BB31	SEC2BB30	SEC2BB29	SEC2BB28	SEC2BB27	SEC2BB26	SEC2BB25	SEC2BB24	SEC2BB23	SEC2BB22	SEC2BB21	SEC2BB20	SEC2BB19	SEC2BB18	SEC2BB17	SEC2BB16	SEC2BB15	SEC2BB14	SEC2BB13	SEC2BB12	SEC2BB11	SEC2BB10	SEC2BB9	SEC2BB8	SEC2BB7	SEC2BB6	SEC2BB5	SEC2BB4	SEC2BB3	SEC2BB2	SEC2BB1	SEC2BB0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC0	FLASH_ SECHDPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDP2_ACCDIS	HDP1_ACCDIS
	リセット値																															0	0
0xC4	FLASH_ PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
	リセット値																															0	0
0xC8	予約済み	予約済み																															

表79.Flashレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xD0 + 0x4 *（x - 1）、（x = 1 ~ 8）最後のアドレス 1：0xEC	FLASH_PRIVBB1Rx	PRIV1BB31	PRIV1BB30	PRIV1BB29	PRIV1BB28	PRIV1BB27	PRIV1BB26	PRIV1BB25	PRIV1BB24	PRIV1BB23	PRIV1BB22	PRIV1BB21	PRIV1BB20	PRIV1BB19	PRIV1BB18	PRIV1BB17	PRIV1BB16	PRIV1BB15	PRIV1BB14	PRIV1BB13	PRIV1BB12	PRIV1BB11	PRIV1BB10	PRIV1BB9	PRIV1BB8	PRIV1BB7	PRIV1BB6	PRIV1BB5	PRIV1BB4	PRIV1BB3	PRIV1BB2	PRIV1BB1	PRIV1BB0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xF0 + 0x4 *（x - 1）、（x = 1 ~ 8）最後のアドレス：0x10C	FLASH_PRIVBB2Rx	PRIV2BB31	PRIV2BB30	PRIV2BB29	PRIV2BB28	PRIV2BB27	PRIV2BB26	PRIV2BB25	PRIV2BB24	PRIV2BB23	PRIV2BB22	PRIV2BB21	PRIV2BB20	PRIV2BB19	PRIV2BB18	PRIV2BB17	PRIV2BB16	PRIV2BB15	PRIV2BB14	PRIV2BB13	PRIV2BB12	PRIV2BB11	PRIV2BB10	PRIV2BB9	PRIV2BB8	PRIV2BB7	PRIV2BB6	PRIV2BB5	PRIV2BB4	PRIV2BB3	PRIV2BB2	PRIV2BB1	PRIV2BB0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

8 命令キャッシュ (ICACHE)

8.1 ICACHEの概要

命令キャッシュ (ICACHE)は、内部および外部メモリから命令およびデータをフェッチするときのパフォーマンスを高めるためにCortex-M33プロセッサのC-AHBコードバスに導入されました。

デュアル･マスタ･ポート、ヒットアンダーミス、クリティカルワードファースト･リフィル･ポリシーなど、いくつかの特定の機能により、ほとんどのユースケースにおいてゼロに近いウェイトステート性能が実現可能です。

8.2 ICACHEの主な機能

ICACHEの主な機能は次の通りです。

- バスインタフェース
  - 1つの32bit AHBスレーブポート、実行ポート（Cortex-M33 C-AHBコードインタフェースからの入力）
  - 2つのAHBマスタポート : マスタ1およびマスタ2ポート（それぞれメインAHBバスマトリックスの高速バスと低速バスへの出力）
  - 制御用の32bit AHBスレーブポートx1（AHBペリフェラル相互接続からの入力、ICACHEレジスタへのアクセス用）
- キャッシュアクセス
  - ヒット時のゼロウェイトステートを実現します。
  - ヒットアンダーミス機能：以前のキャッシュミスによるラインリフィル中にプロセッサリクエスト（キャッシュデータへのアクセス）を処理する機能です。
  - デュアルマスタアクセス：対象メモリに応じてトラフィックを切り離すために使用する機能です。たとえば、ICACHEは、高速トラフィックのAHBマスタ1ポートへの割当て（FlashおよびSRAMメモリのアドレス指定）、低速トラフィックのAHBマスタ2ポートへの割当て（外部メモリのアドレス指定）により、外部メモリからのラインリフィル時にプロセッサがストールするのを防ぎます。この機能より、内部Flashメモリにおける ISR（割込みサービスルーチン）フェッチを、外部メモリからのキャッシュラインリフィルと同時に行うことができます。
  - 2つのマスタがあることで、割込み遅延への影響が最小限に抑えられます。
  - キャッシュラインのサイズの WRAPw バースト（キャッシュラインのサイズに合わせた32bitのワードサイズw）による最適なキャッシュラインのリフィル
  - 1 ウェイに設定される可能性がある n ウェイセットアソシアティブのデフォルト設定とは、超低電力消費プロファイルを必要とする用途向けのダイレクトマップ方式のキャッシュを意味します。
- メモリアドレス再配置
  - 入力アドレスを最大4つのメモリ領域に再配置することができます（外部メモリにあるエイリアスコードをコード領域に再配置して、C-AHBコードインタフェースから実行するために使用されます）。
- 書換えとリフィル
  - pLRU-t書換え方式（疑似最低使用頻度、バイナリツリーベース）、複雑性と性能のベストバランスを提供するアルゴリズムを実装しています。
  - リフィルのクリティカルワードファースト方式がプロセッサのストールを最小化します。

  - 再配置した領域に対して AHB メモリトランザクションのバーストタイプ、INCRw または WRAPwを設定することができます（サイズwはキャッシュラインのサイズに合わせたもの）。
- 性能カウンタ

  ICACHEは次の2つの性能カウンタを実装しています。
  - ヒットモニタカウンタ（32bit）
  - ミスモニタカウンタ（16bit）
- エラー管理
  - 予期しないキャッシュ格納可能な書込みアクセスの検出、エラーフラグのセット、および割込み生成（オプション）が可能です。
- TrustZoneセキュリティをサポート
- メンテナンス操作
  - キャッシュの無効化：キャッシュの完全無効化、高速コマンド、割込み不可

8.3 ICACHEの実装

表80.STM32U535/545/575/585のICACHE機能

機能	ICACHE
ウェイ数	2
キャッシュサイズ	8KB
キャッシュライン幅	16バイト
再配置されるメモリ領域の範囲精度	2MB
再配置する領域数	4
AHBスレーブインタフェースのデータサイズ	32bit
AHB高速マスタ1インタフェースのデータサイズ	128bit
AHB低速マスタ2インタフェースのデータサイズ	32bit

表81.STM32U59x/5Ax/5Fx/5GxのICACHE機能

機能	ICACHE
ウェイ数	2
キャッシュサイズ	32KB
キャッシュライン幅	16バイト
再配置されるメモリ領域の範囲精度	2MB
再配置する領域数	4
AHBスレーブインタフェースのデータサイズ	32bit
AHB高速マスタ1インタフェースのデータサイズ	128bit
AHB低速マスタ2インタフェースのデータサイズ	32bit

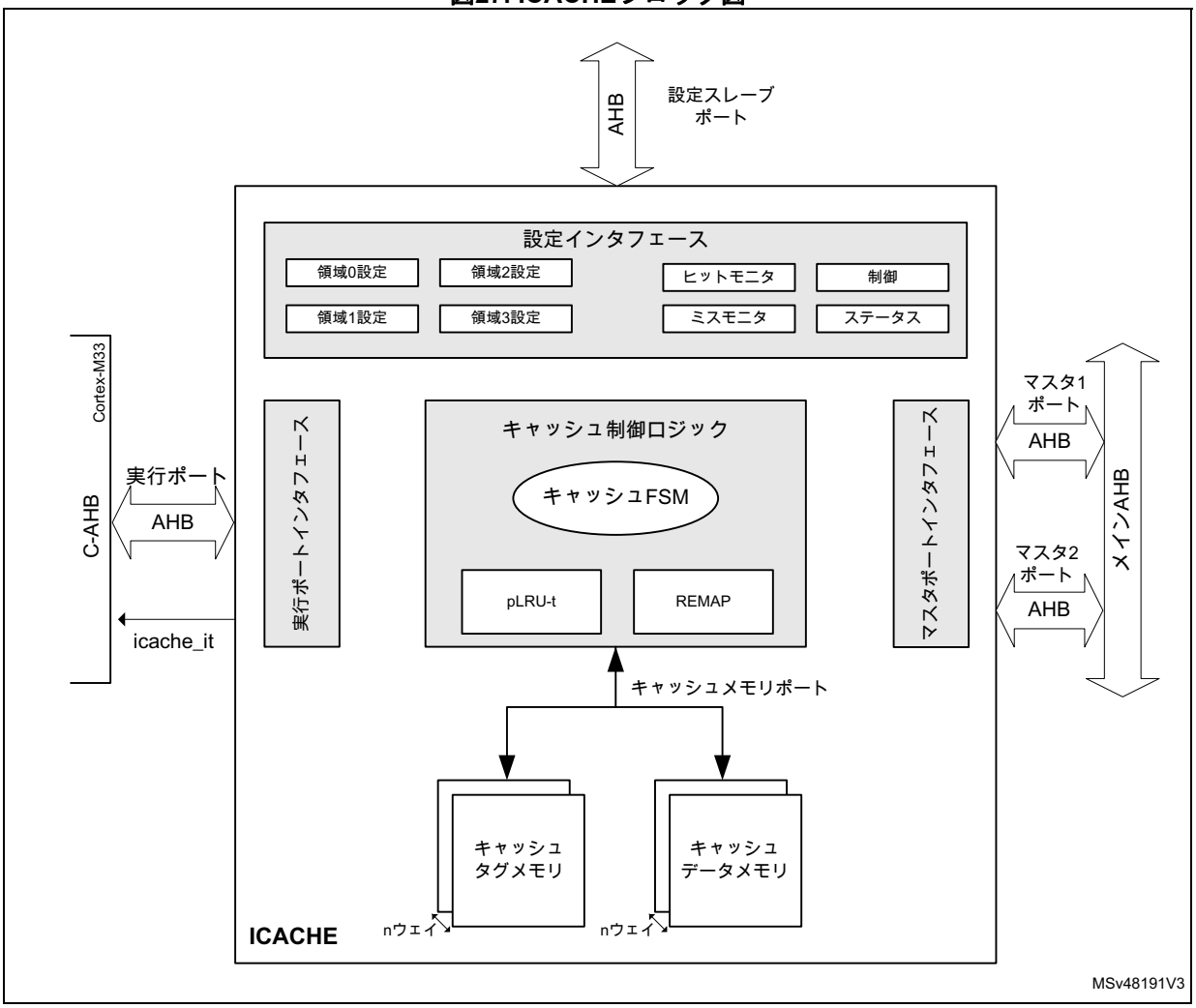
8.4 ICACHEの機能説明

命令キャッシュの目的は、プロセッサからの命令フェッチや命令メモリロードをキャッシュすることです。そのため、ICACHEは読出しトランザクションのみを管理し、書込みトランザクションは管理しません。

エラー管理の目的で、キャシュ可能な書込みトランザクションを実行する場合（ソフトウェアプログラミングが正しくない場合のみ）、ICACHEはエラーフラグをセットし、有効な場合はプロセッサに割込みを生成します。

8.4.1 ICACHEブロック図

図27. ICACHEブロック図



8.4.2 ICACHEのリセットおよびクロック

ICACHEは、Cortex-M33 C-AHBバスクロックでクロック供給されます。

ICACHEリセット信号が解除されると、キャッシュ無効化手順が自動的に起動し、ICACHEをビジーにします（ICACHE_SR = 0x0000 0001）。

この手順の終了後

- ICACHE の無効化：「コールドキャッシュ」と呼ばれる、すべてのキャッシュラインの有効ビットがゼロの状態になります（ICACHEは完全にフィルされる必要があります）。
- ICACHE_SR = 0x0000 0002（キャッシュがビジーでなくなったことを反映しています）。
- ICACHEは無効：ICACHE_CRのENビットはリセット状態（= 0）を保持します。

注： **無効な場合、機能継続中の再配置メカニズムを除き、ICACHEはバイパスされ、スレーブ入力リクエスト（再配置される/されない）はマスタポートに転送されるのみです。**

8.4.3 ICACHE TAGメモリ

ICACHEのタグメモリは以下を含みます。

- どのデータがキャッシュデータメモリに入っているかを示すアドレスタグ
- 有効性ビット

キャッシュラインごと（ウェイごと）に1つの有効ビットがあります。

有効ビットは、キャッシュラインのリフィル時（ミス後）にセットされます。

有効ビットは次のタイミングでリセットされます。

- ICACHEのリセットが解除されたあと
- ICACHE_CRのEN = 0をセットすることによって（ソフトウェアによって）、キャッシュが無効になったとき
- ICACHE_CRのCACHEINV = 1をセットすることによって（ソフトウェアによって）、ICACHE無効化コマンドを実行するとき

キャッシュ格納可能なトランザクションが実行入力ポートで受信されると、そのAHBアドレス（HADDR_in）は次のフィールドに分割されます（BおよびW定義の表82を参照）。

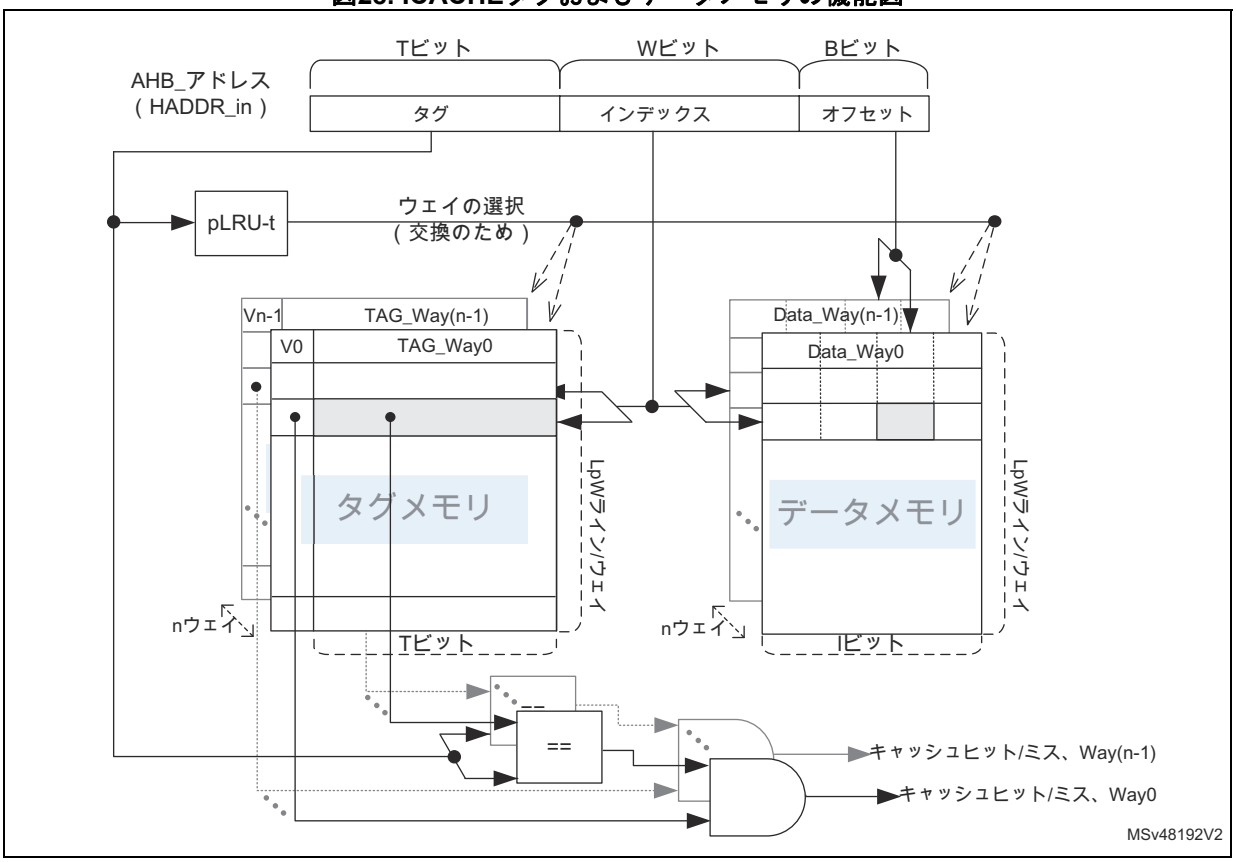
- HADDR_in[B-1:0]：アドレスバイトオフセット。キャッシュライン内で選択するバイトを示します。
- HADDR_in[B+W-1:B]：アドレスウェイインデックス。各ウェイ内で選択するキャッシュラインを示します。
- HADDR_in[31:B+W]：タグアドレス。リクエストされたデータがすでに ICACHE 内で使用可能（有効）かどうかを確認するために、TAGメモリアドレスと比較されます。

下の表に、TAGメモリのディメンショニングに関するICACHEの主なパラメータの概要を示します。
図28は、nウェイセットアソシアティブICACHEのTAGおよびデータメモリの機能図を示しています。

表82.nウェイセットアソシアティブ動作モード（デフォルト）のTAGメモリディメンショニングパラメータ

パラメータ	値	例
キャッシュサイズ	SKB = sバイト(s = 1024 x S)	8KB = 8192バイト
キャッシュのウェイ数	n	2
キャッシュラインサイズ	Lバイト = lビット(l = 8 x L)	16バイト = 128bit
キャッシュライン数(ウェイあたり)	LpW = s / (n x L) ライン / ウェイ	256ライン / ウェイ
アドレスバイトのオフセットサイズ	B = log2(L)ビット	4bit
アドレスウェイのインデックスサイズ	W = log2(LpW)ビット	8bit
タグアドレスサイズ	T = (32 - W - B)ビット	20bit

図28. ICACHEタグおよびデータメモリの機能図



8.4.4 ダイレクトマップICACHE(1ウェイキャッシュ)

デフォルト設定（リセット時）はnウェイセットアソシアティブキャッシュ（ICACHE_CRのWAYSEL = 1）ですが、ユーザはWAYSEL = 0を書き込むことによってICACHEをダイレクトマップとして設定できます（キャッシュが無効、つまりICACHE_CRのEN = 0のときのみ可能）。

下の表に、ダイレクトマップキャッシュ動作モードが選択されたときのTAGメモリに関するICACHEの主なパラメータの概要を示します。

表83.ダイレクトマップキャッシュモードのTAGメモリディメンショニングパラメータ

パラメータ	値	例
キャッシュサイズ	SKB = sバイト(s = 1024 x S)	8KB = 8192バイト
キャッシュのウェイ数	1	1
キャッシュラインサイズ	Lバイト = lビット(l = 8 x L)	16バイト = 128bit
キャッシュライン数	LpW = s / Lライン	512本のライン
アドレスバイトのオフセットサイズ	B = log2(L)ビット	4bit
アドレスウェイのインデックスサイズ	W = log2(LpW)ビット	9bit
タグアドレスサイズ	T = (32 - W - B)ビット	19bit

すべてのキャッシュ操作（読出し、リフィル、再配置、無効化）は、ダイレクトマップ設定でも同じままです。唯一の違いは、データのリフィルに使用できるウェイは1つのみ（固有のもの）なので、ラインエビクション（セクション8.4.8で説明）が発生した場合の置換アルゴリズムがないことです。

8.4.5 ICACHE有効化

ICACHEを有効にするには、ICACHE_CRのENビットを1にセットする必要があります。

無効な場合、ICACHEはバイパスされ、すべてのトランザクションはスレーブポートから同じクロックサイクルでマスタポートにコピーされます。

ICACHEが無効なときに、メインメモリの内容（後でキャッシュされる領域）を初期化または変更し、この領域が変更されないときにのみ、ICACHEを有効にすることをお勧めします（有効なICACHEはキャッシュ可能な書込みトランザクションをエラーとして検出します）。

パフォーマンスの確定性を確実にするためには、潜在的なキャッシュ無効化手順の終了を待ってから、ICACHEを有効にすることをお勧めします。無効化手順は、ハードウェアリセット信号が解除されたとき、CACHEINVがセットされたとき、またはICACHE_CRでENがクリアされたときに発生します。この手順中、ICACHE_SRのBUSYFがセットされ、手順が終了すると、BUSYFがクリアされ、同じレジスタのBSYENDFがセットされます（そのようなビジー終了条件では、有効な場合、ICACHE割込みが生成されます）。

ソフトウェアは、ICACHEを有効にする前に、BUSYFおよびBSYENDFの値をテストする必要があります。無効化手順が終了する前にICACHEが有効化された場合、キャッシュアクセス（BUSYF = 1のとき）はキャッシュ不可として扱われ、パフォーマンスはメインメモリのアクセス時間に依存することになります。

入力トランザクションアドレスが、ICACHE_CRRxで定義され、有効化されたメモリ領域に属する場合、ICACHEが有効かどうかに関係なく、アドレス再配置が行われます（図29を参照）。

ICACHEはデフォルトではブート時に無効化されます。

8.4.6 キャッシュ格納可および不可のトラフィック

ICACHEは、Cortex-M33コア用に開発されています。C-AHBバス上に配置されているため、コードメモリ領域（メモリマップの0x0000 0000から0x1FFF FFFFまで）をキャッシュします。

その他のメモリ領域をキャッシュ可能にするために、ICACHEはメモリ領域再配置機能をサポートしています。これは最大4つの外部メモリ領域を定義するために使用され、これらのアドレスはコード領域でエイリアスを持ちます。これらの外部メモリ領域をコードエイリアスアドレスを通じてアドレス指定することで、メモリリクエストをC-AHBバスにルーティングし、ICACHEによって管理できます。

[0x6000 0000:0xAFFF FFFF]の範囲にある任意のアドレスに物理的に配置されている外部メモリ空間は、[0x0000 0000:0x07FF FFFF]または[0x1000 0000:0x1FFF FFFF]の範囲にあるアドレスでエイリアスを作成することができます。

コード領域にある特定のメモリリクエストに対して、ICACHEはまずアドレス再配置機能を実装します。エイリアスが作成されると、それは次にキャッシュされるアドレスに再配置され、必要な場合は、メインAHBバスバスマトリックスのアドレスを指定するためにマスタポートに提供されます。デスティネーション物理アドレスは、AHBバスに対するそれ以上の操作を必要としません。

再配置機能は、キャッシュ不可トラフィックや、キャッシュが無効なときにも使用できます。

アドレス再配置の詳細については、セクション8.4.7を参照してください。

ICACHEへのメモリリクエストは、そのAHBトランザクションメモリルックアップ属性に応じて（表84を参照）、キャッシュ可能と定義されます。このAHB属性は、アドレス指定された領域のMPU（メモリ保護ユニット）プログラミングに依存します。

表84.AHBトランザクションのICACHEのキャッシュ可能性

AHBルックアップ属性	キャッシュ可能性
1	キャッシュ可能
0	キャッシュ不可

キャッシュ不可アクセスの場合、ICACHEはバイパスされます。すなわち、AHBトランザクションが変更されずにそのままマスタ出力ポートに伝搬されます。ただし、アドレス再配置機能（セクション8.4.7を参照）によって変更される可能性があるトランザクションアドレスを除きます。

バイパス、およびその結果としての再配置ロジックにより、対象メモリへのアクセス遅延時間が長くなることはありません。

キャッシュ可能アクセスの場合、ICACHEはセクション8.4.8で説明されているように動作します。

キャッシュ可能メモリ領域は、ユーザによってMPUで定義され、プログラムされます。MPUは特定の領域をアドレス指定するトランザクションについて、AHB属性信号の生成を行います。

次の表に、さまざまなメモリのプログラム可能な設定の概要を示します。

表85.メモリ設定

メモリ	キャッシュ可能 （MPUプログラミング）	ICACHEで再配置 (ICACHE_CRRxプログラミング)
Flashメモリ	はい/いいえ	不要
SRAM	推奨せず	不要
外部メモリ	あり	必要
	なし	ユーザが外部メモリのコードをC-AHBバスでフェッチしたい場合は必須（そうでない場合はS-AHBバスでフェッチ）

8.4.7 アドレス再配置

ICACHEにより、最大4つの外部メモリ領域について、コード領域でエイリアスアドレスを定義できます。

アドレス再配置はコードエイリアスアドレスに適用されて、デスティネーション外部物理アドレスに変換されます。

再配置操作は、ICACHE_CRRx（再配置領域数x = 0～3）をプログラムすることで、完全にソフトウェアによって設定可能です。このプログラミングは、ICACHEが無効なときのみ行うことができます。

各領域xは、ICACHE_CRRxのRENで個別に有効化できます。有効化されると、ICACHEが無効な場合や、トランザクションがキャッシュ不可の場合でも、再配置操作が発生します。

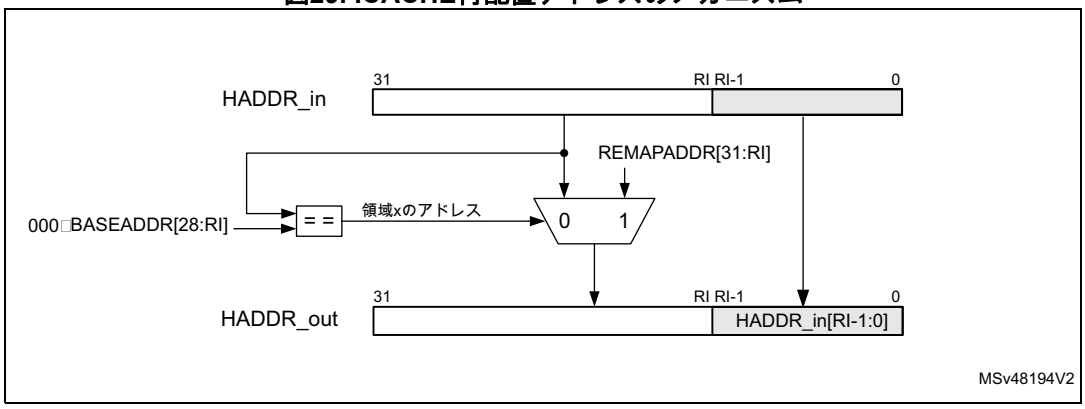
再配置領域のサイズは同じでなくても構いません。各領域のサイズはそのICACHE_CRRxのRSIZEでプログラムできます。各領域のサイズは範囲の最小単位（2MB）の2の累乗値であり、最小領域サイズは2MB、最大領域サイズは128MBです。

アドレス再配置のメカニズムは、受信AHBアドレス（HADDR_in）と特定のコードサブ領域のベースアドレスが一致することと、以下に示すように、このアドレスを（再配置された）外部物理アドレスに変更することに基づきます。

- HADDR_in[31:RI] = 000:BASEADDR[28:RI]の場合、HADDR_inは領域xに属します。ここで、
  - 000:BASEADDRは、ICACHE_CRRxのBASEADDRでプログラムされたコードサブ領域のベースアドレスです。
  - RIには考慮すべき上位ビット数が定義されます。RI = log2（領域サイズ）であり、最小値は21（2MB領域用）、最大値は27（128MB領域用）です。
- 領域xが有効な場合、マスタポート出力AHBのアドレス（HADDR_out）は次の2つの部分を連結することによって構成されます。
  - MSBとして、ICACHE_CRRxのREMAPADDR[31:RI]フィールド
  - LSBとして、HADDR_in[RI-1:0]

下の図は、一致および出力アドレス生成について示しています。

図29. ICACHE再配置アドレスのメカニズム



次の表に、RSIZEに応じて、ICACHE_CRRxのBASEADDRおよびREMAPADDR（上位MSBの数）の可能なすべての設定を示します。

表86.ICACHE再配置領域のサイズ、ベースアドレス、および再配置アドレス

領域サイズ（MB）	ベースアドレスサイズ（MSB）	再配置アドレス（MSB）
2	8	11
4	7	10
8	6	9
16	5	8
32	4	7
64	3	6
128	2	5

ICACHE_CRRxのBASEADDRおよびREMAPADDRをプログラムするときには注意が必要です。プログラムされた値が予想値より大きい場合（MSB数については表86を参照）、不要かつ余分なLSBは無視されます。

代表的な再配置の例：外部アドレス範囲[0x6800 0000:0x6FFF FFFF]に物理的に配置されている128MBのFMC領域（NOR/SRAM）を、コードセクション範囲[0x1000 0000:0x17FF FFFF]に再配置します。

- REMAPADDR[31:21] = 0x340
- BASEADDR[28:21] = 0x80
- HADDR_in[31:27]は000:BASEADDR[28:27],と比較され、HADDR_in/BASEADDR[26:21]は比較では無視されます。

比較して一致する場合

- HADDR_out[31:27]はREMAPADDR[31:27]になります（HADDR_in[31:27]の代わりに）。
- HADDR_out[26:0]はHADDR_in[26:0]になります。

ソフトウェアは、ICACHE_CRRxのHBURSTをセットすることによって、バスマトリックスのICACHEマスタポートによって生成されるAHBバースト（キャッシュラインのリフィルのため）の種類を次のようにプログラムできます。

- WRAP：バーストモードをサポートできるインタフェースを通じてアクセスされる再配置された外部メモリについてはWRAP。これにより、クリティカルワードファースト機能のパフォーマンス上のメリットが得られます。
  - ラップバーストサイズ = キャッシュラインサイズ
  - ラップバースト開始アドレス = コアによってリクエストされる最初のデータのワードアドレス
- INCR：WRAPバーストモードをサポートしていないインタフェースを通じてアクセスされる外部メモリについては、INCRバーストモード（クリティカルワードファースト機能のメリットは失われます）。
  - INCRバーストサイサイズ = キャッシュラインサイズ
  - INCRバースト開始アドレス = リクエストされたワードを含むキャッシュラインの境界に整列されたアドレス

注： **外部領域とそれらのエイリアスコードサブ領域の両方の SAU（セキュア属性ユニット）および MPU（メモリ保護ユニット）属性をプログラムするときには、一貫性が必要です。**

8.4.8 キャッシュ可能アクセス

ICACHEがキャッシュ可能なトランザクションをCortex-M33から受信すると、ICACHEはリクエストされたアドレスがTAGメモリにあるかどうか、また、対応するキャッシュラインが有効であるかどうかを確認します。

次の3通りの方法があります。

- アドレスがTAGメモリ内にあり、キャッシュラインが有効な場合 : **キャッシュヒット**。データはキャッシュから読み出され、同じサイクル内でプロセッサに提供されます。
- アドレスがTAGメモリにない場合 : **キャッシュミス**。データはメインメモリから読み出されてプロセッサに提供され、キャッシュラインのリフィルが行われます。

  クリティカルワードファーストポリシーによってプロセッサの最小ウェイトサイクルが保証されます。キャッシュがキャッシュラインリフィルを実行している間に読出しデータを提供できるためです（関連する遅延は、メインメモリから1ワードをフェッチする際の遅延です）。

  アドレスの再配置が発生しない場合、ICACHEマスタバスで生成されるバーストはWRAPwです（wは、ワード数によるキャッシュライン幅です）。アドレスの再配置が発生した場合、バーストの種類は、対応するICACHE_CRRxでプログラムされたHBURSTに依存します。

  AHB トランザクション属性も、ラインリフィルのために選択されたマスタポートのメインAHBバスマトリックスに伝搬されます。
- アドレスはタグメモリにないが、現在行われているメインメモリからのリフィルバーストに属する場合 : **キャッシュヒット**（ヒットアンダーミス機能）

  これは、キャッシュラインのリフィル中に発生します。ICACHEは、リクエストされたデータを、それがマスタインタフェースで使用可能になるとすぐに提供できるため、ミス（メインメモリからのデータのフェッチ）を回避できます。

（キャッシュミスにより）キャッシュリフィルが発生した場合、ICACHEはリフィルデータを書き込むキャッシュラインを次のように選択します。

- ダイレクトマップ（1 ウェイ）モードでは、1 ラインのみを使用してリフィルデータ、すなわち入力アドレスのインデックスで指し示されたラインを格納することができます。
- nウェイセットアソシアティブモードでは、nライン中の1ライン（n個のウェイのそれぞれにおいてアドレスインデックスで指し示されたライン）を使用できます。ウェイの選択は、pLRU-t置換アルゴリズムに基づき、これは、各インデックスについて、次のリフィルのためのウェイ候補を指し示します。

リフィルデータを書き込む必要があるキャッシュラインがすでに有効な場合でも、ターゲットのキャッシュラインを最初に無効化する必要があります。ダイレクトマップモードの場合でも、nウェイセットアソシアティブキャッシュモードの場合でも同様です。

8.4.9 デュアルマスタキャッシュ

ICACHEは、メインAHBバスマトリックス上にデュアルポートAHBマスタ、すなわちマスタ1およびマスタ2ポートを実装できます。これは、別々のデスティネーションメモリへのトラフィックを分割するために使用されます。

再配置されないトラフィックは自動的にマスタ1ポートになります。外部メモリに再配置されたトラフィックは、（領域単位で）ICACHE_CRRxのMSTSELをプログラムすることによって、マスタ2ポートにルーティングする必要があります。

コードは一般に、次のようにフェッチできます。

- マスタ1ポート（高速バス）の内部Flashメモリおよび内部SRAM
- マスタ2ポート（低速バス）の外部Flashメモリ/RAM

外部メモリを実装していないシステムの場合、内部Flashメモリへのトラフィックを内部SRAMへのトラフィックから切り離すことができます（ICACHEによって再配置された場合）。この機能は、ミス発生時のさらなるプロセッサのストールを防ぐために使用されます。

ヒットアンダーミス機能とならび、このデュアルマスタ機能により、プロセッサは異なるメモリからのフェッチ時に代替パスを使用できます。

8.4.10 ICACHEのセキュリティ

ICACHEは、Armv8-M TrustZoneを実装します。

ICACHEの設定レジスタは、システムレベルで保護されます。

8.4.11 ICACHEのメンテナンス

ソフトウェアは、ICACHE_CRレジスタのCACHEINVをプログラムすることによって、ICACHEの全内容を無効化できます。

CACHEINV = 1のとき、ICACHE制御ロジックはICACHE_SRのBUSYFフラグをセットし、キャッシュ無効化操作を開始して、各TAGの有効ビットを0にリセットします（キャッシュラインごとに1つの有効ビット）。CACHEINVは自動的にクリアされます。

無効化操作が終了すると、ICACHE はBUSYF を自動的にクリアして、ICACHE_SR レジスタのBSYENDFをセットします。

このフラグ条件で有効化された場合（ICACHE_IERのBSYENDIE = 1）、ICACHE割込みが生成されます。すると、（空の）キャッシュを再び使用できるようになります。

8.4.12 ICACHE性能の監視

ICACHEは、パフォーマンス分析のために以下のモニタを備えています。

- 32bitのヒットモニタは、ICACHEの内容にヒットしたスレーブキャッシュポート上のキャッシュ可能AHBトランザクションをカウントします。

  また、このモニタは、アドレスがTAGメモリまたはリフィルバッファにある（以前のミスのため、そのデータがキャッシュマスタポートから受信されるか、まもなく受信される）すべてのアクセスも考慮します（セクション8.4.8を参照）。
- 16bitのミスモニタは、ICACHEの内容にミスしたスレーブキャッシュポート上のキャッシュ可能AHBトランザクションをカウントします。

  また、このモニタは、アドレスがTAGメモリにもリフィルバッファにも存在しないすべてのアクセスも考慮します。

最大値に達しても、これらのモニタは最小値に戻りません。

ヒットおよびミスモニタは、特定のコードの分析が可能なソフトウェアによって有効化およびリセットすることができます。

ソフトウェアは次のタスクを実行できます。

- ヒットモニタの有効化/停止は、ICACHE_CRのHITMENで行います。
- ヒットモニタをリセットするには、ICACHE_CRのHITMRSTをセットします。
- ミスモニタの有効化/停止は、ICACHE_CRのMISSMENで行います。
- ミスモニタをリセットするには、ICACHE_CRのMISSMRSTをセットします。

消費電力を低減するために、これらのモニタはデフォルトでは無効化（停止）されています。

8.4.13 ICACHEブート

ICACHEはブート時に無効化されます（ICACHE_CRのEN = 0）。

Cortex-M33は0x0以外のブート開始アドレス定義を可能にするVTOR（ベクタテーブル）を実装しているため、ブート時にコードを再配置する必要はありません。

ブートが終了すると、ICACHEを有効にできます（ソフトウェアでICACHE_CRのEN = 1をセット）。

8.5 ICACHE低電力モード

デバイスレベルでは、ICACHEの使用により、たいていの場合、サイズが大きく、そのために消費電力も多いメインメモリからではなく、内部ICACHEから命令をフェッチすることによって、消費電力が低減されます。キャッシュされるメインメモリが外部メモリの場合、低減量はさらに大きくなります。

より低性能プロファイル（ヒット率の点で）のアプリケーションや、厳しい低消費電力制約を持つアプリケーションの場合、ダイレクトマップとして設定されたICACHEの低電力消費が有効かもしれません。シングルウェイキャッシュ設定は、ICACHE_CRのWAYSEL = 0をプログラムすることで取得されます（図28を参照）。消費電力は、リクエストのたびに、タグおよびデータメモリの必要な一部分だけにアクセスすることによって低減されます。その間も、キャッシュ効果によってフェッチパフォーマンスは向上します。それでも、ほとんどのコード実行の場合、nウェイセットアソシアティブキャッシュモードよりは、効率がやや低くなります。

8.6 ICACHEのエラー管理と割込み

サポートされていないキャッシュ可能な書込みリクエストが検出された場合（機能エラー）、ICACHEはICACHE_SRのERRFフラグをセットすることによってエラーを生成します。対応する割込みイネーブルビットがセットされている場合（ICACHE_IERのERRIE = 1）、割込みが生成されます。

その他にキャッシュ無効化操作の終わりにも、割込みが生成される可能性があります。キャッシュビジー状態が終了すると、ICACHEはICACHE_SRのBSYENDFフラグをセットします。対応する割込みイネーブルビットがセットされている場合（ICACHE_IERのBSYENDIE = 1）、割込みが生成されます。

すべてのICACHE割込みソースは、同一かつ一意な割込み信号icache_itを生成し、同じ割込みベクタを使用します。

表87.ICACHE割込み

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法
ICACHE	機能エラー	ICACHE_SRのERRF	ICACHE_IERのERRIE	ICACHE_FCRのCERRFを1にセット
	ビジー状態終了（無効化終了）	ICACHE_SRのBSYENDF	ICACHE_IERのBSYENDIE	ICACHE_FCRのCBSYENDFを1にセット

また、ICACHEは、すべてのAHBバスエラー（セキュリティ問題、アドレスデコード問題など）をマスタ1またはマスタ2ポートから実行ポートに再伝播します。

8.7 ICACHEレジスタ

8.7.1 ICACHE制御レジスタ（ICACHE_CR）

アドレスオフセット：0x000

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MISSMRST	HITMRST	MISSMEN	HITMEN
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WAYSEL	CACHEINV	EN
													rw	w	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **MISSMRST**：ミスモニタのリセット

0：影響なし。

1：キャッシュミスモニタをリセットします。

ビット18 **HITMRST**：ヒットモニタのリセット

0：影響なし。

1：キャッシュヒットモニタをリセットします。

ビット17 **MISSMEN**：ミスモニタの有効化

0：キャッシュミスモニタのスイッチを切ります。モニタは停止してもリセットされません。

1：キャッシュミスモニタを有効にします。

ビット16 **HITMEN**：ヒットモニタの有効化

0：キャッシュヒットモニタのスイッチを切ります。モニタは停止してもリセットされません。

1：キャッシュヒットモニタを有効にします。

ビット15:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **WAYSEL**：アソシアティブキャッシュモード選択

このビットによって、ICACHEセットアソシアティブを選択することができます。キャッシュが無効（EN = 0）の場合のみ、ソフトウェアで書き込むことができます。

0：ダイレクトマップキャッシュ（1-ウェイキャッシュ）

1：n-ウェイセットアソシアティブキャッシュ（リセット値）

ビット1 **CACHEINV**：キャッシュ無効化（インバリデート）

BUSYFフラグがセットされたときに（キャッシュのメンテナンス操作中）、ソフトウェアによってセットされ、ハードウェアによってクリアされます。0を書き込んでも、ビットの値は変化しません。

0：影響なし。

1：キャッシュ全体を無効化（インバリデート）します（すべてのキャッシュラインの有効ビット = 0）。

ビット0 **EN**：有効化（イネーブル）

0：キャッシュは無効です。（ディセーブル）

1：キャッシュは有効です。（イネーブル）

8.7.2 ICACHEステータスレジスタ（ICACHE_SR）

アドレスオフセット：0x004

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRF	BSYENDF	BUSYF
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ERRF**：キャッシュエラーフラグ

0：エラーはありません。

1：操作（キャッシュ可能な書込み）中にエラーが発生しました。

ビット1 **BSYENDF**：ビジー終了フラグ

0：キャッシュはビジーです。

1：完全無効化CACHEINV操作が終了しました。

ビット0 **BUSYF**：ビジーフラグ

0：キャッシュはCACHEINV操作でビジーではありません。

1：キャッシュは完全無効化CACHEINV操作を実行中です。

8.7.3 ICACHE割込み有効化レジスタ（ICACHE_IER）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRIE	BSYENDIE	Res.
													rw	rw	

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ERRIE**：キャッシュエラー時の割込みイネーブル

キャッシュ機能エラー（キャッシュ可能な書込みアクセス）発生時に割込み生成を有効にするために、ソフトウェアによってセットされます。

0：エラー時の割込みは無効です。

1：エラー時の割込みは有効です。

ビット1 **BSYENDIE**：ビジー終了時の割込みイネーブル

キャッシュ無効化（インバリデート）操作の終了時に割込み生成を有効にするために、ソフトウェアによってセットされます。

0：ビジー終了時の割込みは無効です。

1：ビジー終了時の割込みは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

8.7.4 ICACHEフラグクリアレジスタ（ICACHE_FCR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERRF	CBSYENDF	Res.
													w	w	

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CERRF**：キャッシュエラーフラグのクリア
ソフトウェアによってセットされます。
0：影響なし。
1：ICACHE_SRのERRFフラグをクリアします。

ビット1 **CBSYENDF**：ビジー終了フラグのクリア
ソフトウェアによってセットされます。
0：影響なし。
1：ICACHE_SRのBSYENDFフラグをクリアします。

ビット0 予約済みであり、リセット値に保持する必要があります。

8.7.5 ICACHEヒットモニタレジスタ（ICACHE_HMONR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HITMON[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HITMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **HITMON[31:0]**：キャッシュヒットモニタのカウンタ

8.7.6 ICACHEミスモニタレジスタ（ICACHE_MMONR）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISSMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **MISSMON[15:0]**：キャッシュミスモニタのカウンタ

8.7.7 ICACHE領域x設定レジスタ（ICACHE_CRRx）

アドレスオフセット：0x020 + 0x4 * x（x = 0から3）

リセット値：0x0000 0200

他の領域のコード領域でエイリアスアドレスを定義して、キャッシュ可能にします。

ICACHE_CRのEN = 1のとき、BASEDADDRおよびREMAPADDRフィールドは書込みロックされています（読出し専用）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HBURST	Res.	Res.	MSTSEL	Res.	REMAPADDR[31:21]										
rw			rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REN	Res.	Res.	Res.	RSIZE[2:0]			Res.	BASEADDR[28:21]							
rw				rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **HBURST**：領域xの出力バーストタイプ

0：WRAP

1：INCR

ビット30:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **MSTSEL**：領域xのAHBキャッシュマスタ選択

0：選択なし（デフォルトでマスタ1に選択されています）。

1：マスタ2を選択します。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26:16 **REMAPADDR[31:21]**：領域xの再配置アドレス

このフィールドには、BASEADDRフィールドで定義したエイリアスアドレスが入ります。

唯一の有用なビットは[31:RI]です。RIは21 ≤ RI ≤ 27の範囲にある、RSIZEのビット数です（セクション8.4.7を参照）。プログラムされた値にLSBの方が多い場合、無用なビットは無視されます。

ビット15 **REN**：領域xの有効化

0：無効

1：有効

ビット14:12 予約済みであり、リセット値に保持する必要があります。

ビット11:9 **RSIZE[2:0]**：領域xのサイズ

000：予約済み
001：2MB
010：4MB
011：8MB
100：16MB
101：32MB
110：64MB
111：128MB

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **BASEADDR[28:21]**：領域xのベースアドレス

このエイリアスアドレスはREMAPADDRフィールドによって置き換えられます。
唯一の有用なビットは[28:RI]です。RIは21 ≤ RI ≤ 27の範囲にある、RSIZEのビット数です（セクション8.4.7を参照）。プログラムされた値にLSBの方が多い場合、無用なビットは無視されます。

8.7.8 ICACHEレジスタマップ

表88.ICACHEレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	ICACHE_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MISSMRST	HITMRST	MISSMEN	HITMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WAYSEL	CACHEINV	EN
	リセット値													0	0	0	0														1	0	0
0x004	ICACHE_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRF	BSYENDF	BUSYF
	リセット値																														0	0	1
0x008	ICACHE_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRIE	BSYENDIE	Res.
	リセット値																														0	0	
0x00C	ICACHE_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERRF	CBSYENDF	Res.
	リセット値																														0	0	
0x010	ICACHE_HMONR	HITMON[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	ICACHE_MMONR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MISSMON[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018 - 0x01C	予約済み	予約済み																															
0x020	ICACHE_CRR0	HBURST	Res.	Res.	MSTSEL	Res.	Res.	REMAPADDR[31:21]										REN	Res.	Res.	Res.	RSIZE [2:0]			Res.	Res.	BASEADDR[28:21]						
	リセット値	0			0			0	0	0	0	0	0	0	0	0	0	0				0	0	1			0	0	0	0	0	0	0
0x024	ICACHE_CRR1	HBURST	Res.	Res.	MSTSEL	Res.	Res.	REMAPADDR[31:21]										REN	Res.	Res.	Res.	RSIZE [2:0]			Res.	Res.	BASEADDR[28:21]						
	リセット値	0			0			0	0	0	0	0	0	0	0	0	0	0				0	0	1			0	0	0	0	0	0	0
0x028	ICACHE_CRR2	HBURST	Res.	Res.	MSTSEL	Res.	Res.	REMAPADDR[31:21]										REN	Res.	Res.	Res.	RSIZE [2:0]			Res.	Res.	BASEADDR[28:21]						
	リセット値	0			0			0	0	0	0	0	0	0	0	0	0	0				0	0	1			0	0	0	0	0	0	0

表88.ICACHEレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	ICACHE_CRR3	HBURST	Res.	Res.	MSTSEL	Res.	Res.	REMAPADDR[31:21]										REN	Res.	Res.	Res.	RSIZE [2:0]			Res.	Res.	BASEADDR[28:21]						
	リセット値	0			0			0	0	0	0	0	0	0	0	0	0	0				0	0	1			0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

9 データキャッシュ (DCACHE)

9.1 DCACHEの概要

データキャッシュ（DCACHE）は、外部メモリとの間のデータトラフィックの性能を向上するために、Cortex-M33プロセッサのS-AHBシステムバス、またはマスタペリフェラルによって駆動されているAHBバスに導入されています。

ヒットアンダーミスやクリティカルワードファーストリフィルポリシーなど、いくつかの特定の機能により、外部メモリデータアクセスで最適なパフォーマンスが得られます。

9.2 DCACHEの主な機能

DCACHEの主な機能は次の通りです。

- バスインタフェース
  - 1つの32bit AHBスレーブポート、システムポート（Cortex-M33 S-AHBシステムインタフェースからの入力、またはGPU2DのポートM0によって駆動されるAHBバスからの入力）
  - 1つの32bit AHBマスタポート（メインAHBバスマトリックスへの出力）
  - 制御用の1つの32bit AHBスレーブポート（DCACHEレジスタにアクセスするためのAHBペリフェラル相互接続からの入力）
- キャッシュアクセス
  - ヒット時のゼロウェイトステートを実現します。
  - ヒットアンダーミス機能：以前のキャッシュミスによるラインリフィル中にプロセッサリクエスト（キャッシュデータへのアクセス）を処理する機能です。
  - キャッシュラインのサイズの WRAP バーストにより、最適化されたキャッシュラインリフィル（128bitのキャッシュラインのWRAP4など）
  - 2ウェイセットアソシアティブ
  - ライトバックとライトスルーの両方のポリシーをサポートします（AHBバッファ可能属性で選択可能）
  - 読出しおよびライトバックは常に割り当て
  - ライトスルーは常に非割り当て（write-around）
  - バイト、ハーフワード、およびワードの書込みをサポート
- 書換えとリフィル
  - pLRU-t書換え方式（疑似最低使用頻度、バイナリツリーベース）、複雑性と性能のベストバランスを提供するアルゴリズムを実装しています。
  - 読出しトランザクションのクリティカルワードファーストリフィルポリシーにより、プロセッサのストールを最小化します。
  - AHBメモリトランザクションのバーストタイプ、INCRwまたはWRAPwを設定できます（サイズwはキャッシュラインのサイズに合わせたもの）。
- 性能カウンタ

  DCACHEは次の4つのパフォーマンスカウンタを実装しています。
  - 2つのヒットモニタカウンタ（32bit）：読出しヒット数、書込みヒット数
  - 2つのミスモニタカウンタ（16bit）：読出しミス数、書込みミス数

- エラー管理
  - このエラーをフラグしたり、オプションで割込みを生成するために、DCACHE自体（エビクションまたはクリーン操作により、メインメモリに書き戻されたキャッシュライン）によって開始されたマスタポートリクエストのエラーを検出できます。
- TrustZoneセキュリティをサポート
- メンテナンス操作
  - キャッシュの無効化：キャッシュの完全無効化、高速コマンド、割込み不可
  - キャッシュ無効化範囲：指定されたアドレス範囲に属するキャッシュラインを無効化します（リセット有効ビット = 0）。バックグラウンドタスク、割込み可能。
  - キャッシュクリーン範囲：指定されたアドレス範囲に属するキャッシュラインをクリーンします（ダーティビット = 1の場合、ラインを書き戻し、ダーティビットをクリア）。バックグラウンドタスク、割込み可能。
  - キャッシュクリーンおよび無効化範囲：指定されたアドレス範囲に属するキャッシュラインをクリーンして無効化します（ダーティビット = 1の場合、ラインを書き戻し、有効ビットをクリア）。バックグラウンドタスク、割込み可能。

9.3 DCACHEの実装

DCACHE1はCortex-M33 S-AHBバスにあり、メモリマップのアドレス範囲[0x6000 0000:0xAFFF FFFF]で外部RAMメモリ領域（OCTOSPI、HSPI、、およびFMC）のみをキャッシュします。

実際、バスマトリックスでデマルチプレキシングノードをDCACHE1の前に置くことによって、SRAM領域またはペリフェラル領域をアドレス指定したS-AHBバスのメモリリクエスト（それぞれ[0x2000 0000:0x3FFF FFFF]と[0x4000 0000:0x5FFF FFFF]の範囲）は、メインAHBバスマトリックスに直接ルーティングされ、DCACHE1はバイパスされます。

STM32U59x/5Ax/5Fx/5Gxで、DCACHE2はGPU2DのポートM0によって駆動されるAHBバスに配置され、アクセスされるすべてのメモリ領域をキャッシュします。

システム設定コントローラでプログラムされた関連設定に応じてキャッシュ可能またはキャッシュ不可にすることができる内部SRAMへのトランザクションを除き、GPU2Dトランザクションはすべてキャッシュ可能です。

GPU2Dトラフィックは非セキュアであるため、DCACHE2はTrustZoneをサポートしません。

表89.STM32U535/545/575/585のDCACHE機能

機能	DCACHE1
ウェイ数	2
キャッシュサイズ	4KB
キャッシュライン幅	16バイト
AHBマスタインタフェースのデータサイズ	32bit

表90.STM32U59x/5Ax/5Fx/5GxのDCACHE機能

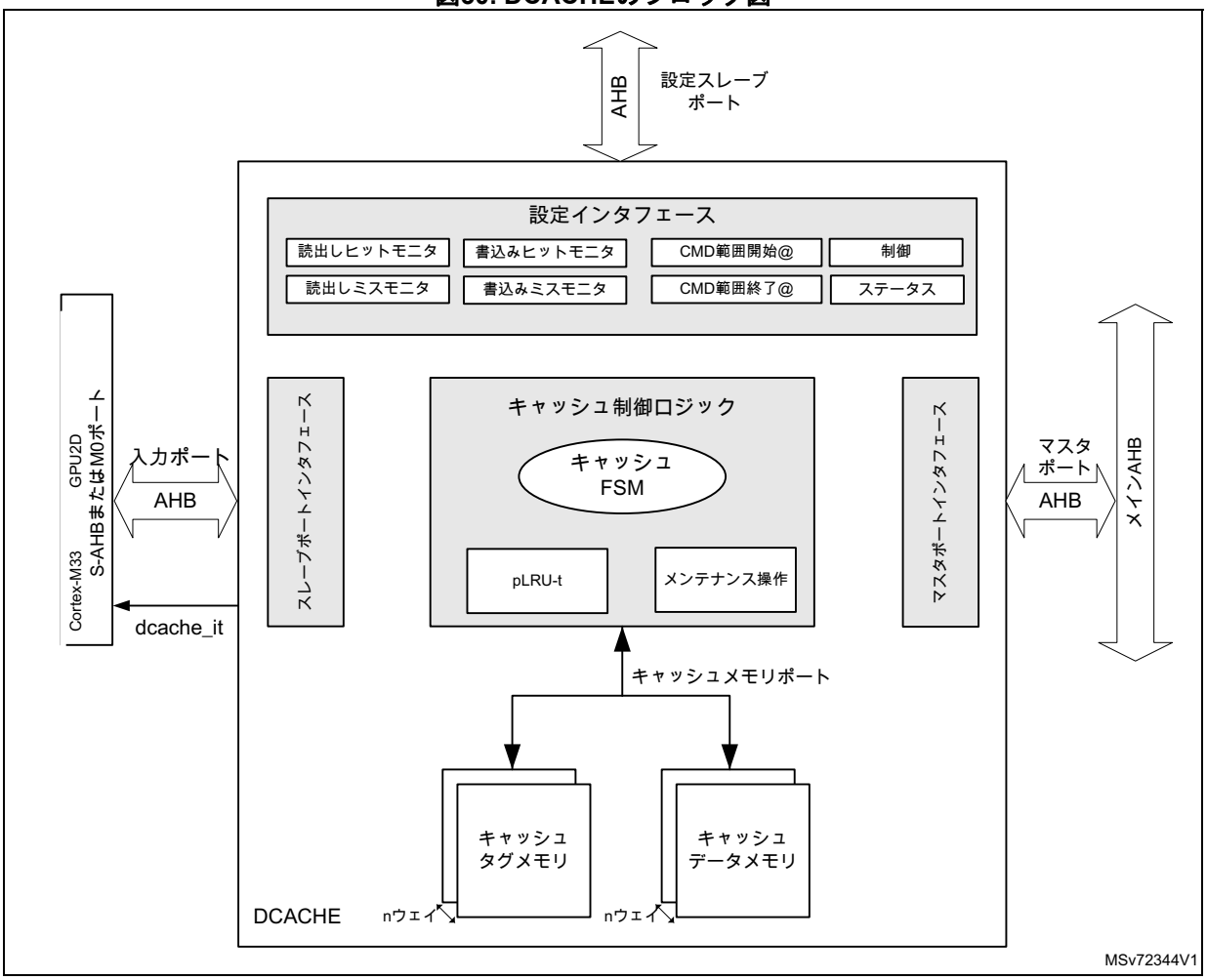
機能	DCACHE1/2
ウェイ数	4
キャッシュサイズ	16KB
キャッシュライン幅	32バイト
AHBマスタインタフェースのデータサイズ	32bit

9.4 DCACHEの機能説明

データキャッシュの目的は、外部メモリデータのプロセッサまたは別のバスマスタペリフェラルからのロードや格納をキャッシュすることです。これらのアクセスには、外部メモリアドレスで発生する可能性のある命令フェッチが含まれます。DCACHEは、読出しと書込みの両方のトランザクションを管理します。

9.4.1 DCACHEのブロック図

図30. DCACHEのブロック図



9.4.2 DCACHEのリセットおよびクロック

DCACHEは、Cortex-M33 S-AHBバスクロックでクロック供給されます。

DCACHEリセット信号が解除されると、キャッシュ無効化手順が自動的に開始され、DCACHEはビジーになります（DCACHE_SR = 0x0000 0001）。

この手順の終了後

- DCACHEの無効化：「コールドキャッシュ」と呼ばれる、すべてのキャッシュラインの有効、ダーティ、および特権ビットがゼロの状態になります（DCACHEは完全にフィルされる必要があります）。
- DCACHE_SR = 0x0000 0002（キャッシュがビジーでなくなったことを反映しています）。
- DCACHEは無効：DCACHE_CRのENビットはリセット状態（= 0）を保持します。

注： **無効なとき、DCACHEはバイパスされます。スレーブ入力リクエストはマスタポートに転送されるだけです。**

9.4.3 DCACHE TAGメモリ

DCACHEのタグメモリは以下を含みます。

- どのデータがキャッシュデータメモリに入っているかを示すアドレスタグ
- 有効性ビット
- ダーティビット
- 特権ビット

キャッシュラインごとに（ウェイごとに）1つの有効ビット、1つのダーティビット、および1つの特権ビットがあります。

有効ビットは、データキャッシュラインへのアクセスを有効/無効にします。ラインが有効でない場合、データアクセス（読出しまたは書込み）はメインメモリで行われます。

有効ビットは、キャッシュラインが書き込まれるとき（読出しミスまたはライトバックミスのいずれかによってリフィルされるとき）にセットされます。

有効ビットは次のタイミングでリセットされます。

- DCACHEのリセットが解除されたあと
- DCACHE_CRのEN = 0をセットすることによって（ソフトウェアによって）、キャッシュが無効になったとき
- ソフトウェアによってDCACHE_CRのCACHEINV = 0、またはCACHECMD = 0b010あるいは0b0011にセットすることによって、DCACHE無効化コマンドの1つを実行するとき（セクション9.4.8を参照）

ダーティビットは、キャッシュラインがメインメモリの内容に関して最新の値を持っていることを示します（キャッシュに最新の正しい値があり、メインメモリは最新ではない）。

ダーティビットは、キャッシュラインがスレーブポート書込みトランザクションによって書き込まれたときにセットされます（ライトバック属性を持つアクセスの場合のみ）。

ダーティビットは、以下のいずれかの場合にリセットされます。

- DCACHEのリセットが解除されたあと
- 読出しミス時にラインリフィルが実行されたとき（ライトバックミス時、リフィルされるキャッシュラインは書込みデータによって変更され、ダーティビット = 1）
- キャッシュ無効化が実行されたとき
- DCACHEクリーン操作の1つが実行され（キャッシュラインがメインメモリに書き戻される）、ソフトウェアによってDCACHE_CRのCACHECMD = 0b001または0b011にセットされたとき（セクション9.4.8を参照）。

特権ビットは、データが特権エンティティによって管理されるかどうかを示します。このラインへの最初のアクセスについて、入力スレーブポートでAHB特権属性の値に従って割り当てられます（ラインリフィル、読出しミス、またはライトバックミス時にのみ書き込まれます）。

特権ビットは、特権アクセスの場合は1、非特権アクセスの場合は0の、特権属性と同じ極性を持ちます。

特権ビットは、キャッシュが無効化されたとき、また、DCACHEリセットが解除された後にリセットされます。

キャッシュ可能トランザクションが入力スレーブポートで受信されたとき、そのAHBアドレス（HADDR_in）は以下のフィールドに分割されます（BおよびWの値については、次の表を参照）。

- HADDR_in[B-1:0] : アドレスバイトオフセット。キャッシュライン内で選択するバイトを示します。
- HADDR_in[B+W-1:B] : アドレスウェイインデックス。各ウェイ内で選択するキャッシュラインを示します。
- HADDR_in[31:B+W] : タグアドレス。リクエストされたデータがすでにDCACHE内で使用可能（有効）かどうかを確認するために、TAGメモリアドレスに比較されます。

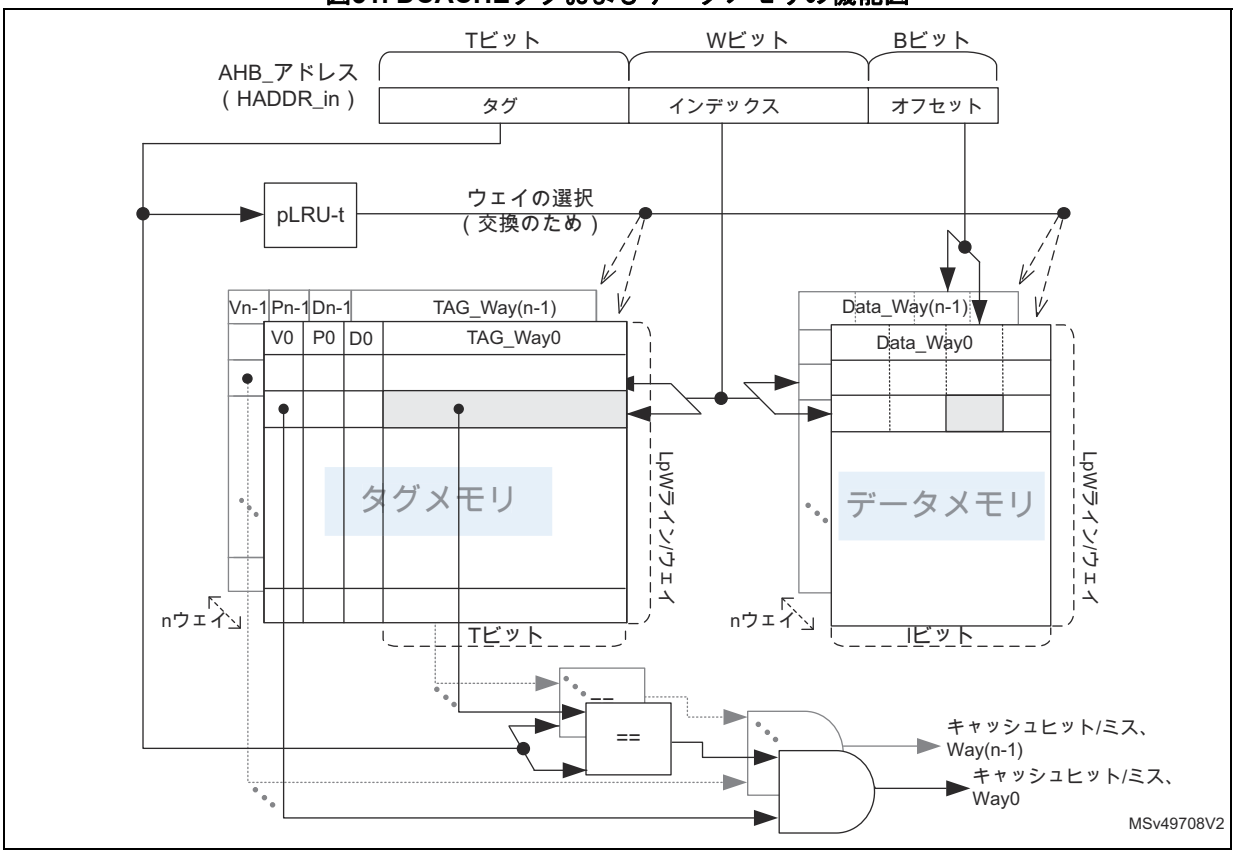
次の表に、TAGメモリのディメンショニングに関するDCACHEの主なパラメータを示します。

図31は、nウェイセットアソシアティブDCACHEについて、TAGおよびデータメモリの機能図を示しています。

表91.TAGメモリディメンショニングパラメータ

パラメータ	値	例
キャッシュサイズ	SKB = sバイト(s = 1024 x S)	4KB = 4096バイト
キャッシュのウェイ数	n	2
キャッシュラインサイズ	Lバイト = lビット(l = 8 x L)	16バイト = 128bit
キャッシュライン数(ウェイあたり)	LpW = s / (n x L)ライン/ウェイ	128ライン/ウェイ
アドレスバイトのオフセットサイズ	B = log2(L)ビット	4bit
アドレスウェイのインデックスサイズ	W = log2(LpW)ビット	7bit
タグアドレスサイズ	T = (32 - W - B)ビット	21bit

図31. DCACHEタグおよびデータメモリの機能図



9.4.4 DCACHE有効化

DCACHEの機能を有効にするには、DCACHE_CR制御レジスタのENビットをセットする必要があります。

無効な場合、DCACHEはバイパスされ、すべてのトランザクションはスレーブポートから同じクロックサイクルのマスタポートにコピーされ、TAGアドレスとの比較は行われません。

DCACHEはデフォルトではブート時に無効化されます。

9.4.5 キャッシュ格納可および不可のトラフィック

DCACHEはCortex-M33コアまたは別のバスマスタペリフェラル用に開発され、これに接続されたAHBバスによってアドレス指定されたメモリ領域をキャッシュします。

さらに、メモリ領域へのAHBバストラフィックは、キャッシュ可能な場合とキャッシュ不可の場合があります。DCACHEに着信するメモリリクエストは、AHBトランザクションメモリルックアップ属性に応じて、キャッシュ可能として定義されます。

書込みトランザクションの場合、DCACHE書込みポリシーは、AHBトランザクションメモリバッファ可能属性に従って、ライトスルーまたはライトバックとして定義されます（次の表を参照）。

これらのAHB属性は、アドレス領域のメモリ保護ユニット（MPU）のプログラミングに依存します。

表92.AHBトランザクションのDCACHEのキャッシュ可能性

AHBルックアップ属性	AHBバッファ可能属性	キャッシュ可能性
0	x	読出しおよび書込み：キャッシュ不可
1	0	読出し：キャッシュ可能 書込み：（キャッシュ可能）ライトスルー
1	1	読出し：キャッシュ可能 書込み：（キャッシュ可能）ライトバック

キャッシュ不可アクセスの場合、DCACHEはバイパスされます。すなわち、AHBトランザクションはマスタ出力ポートにそのまま伝播されます。

バイパスによって対象メモリへのアクセス遅延時間が長くなることはありません。

キャッシュ格納可アクセスの場合、DCACHEはセクション9.4.6で説明されているように動作します。

キャッシュ可能メモリ領域は、ユーザによってMPUで定義され、プログラムされます。MPUは、特定の領域をアドレス指定するトランザクションについて、AHB属性信号の生成を行います。

9.4.6 キャッシュ可能アクセス

DCACHEがスレーブポートでキャッシュ可能なトランザクションをCortex-M33または別のバスマスタペリフェラルから受信する場合、DCACHEはリクエストされたアドレスがそのタグメモリにあり、対応するキャッシュラインが有効であるかを確認します。

読出しトランザクションの場合、次の3通りの方法があります。

- アドレスがタグメモリ内にあり、キャッシュラインが有効な場合：**キャッシュ読出しヒット**。データがキャッシュから読み出され、同じサイクル内でプロセッサに提供されます。
- アドレスがタグメモリ内にない場合：**キャッシュ読出しミス**。データはメインメモリから読み出されてプロセッサに提供され、キャッシュラインのリフィルが行われます。

  クリティカルワードファーストポリシーにより、プロセッサの最小ウェイトサイクルが保証されます。キャッシュがまだキャッシュラインのリフィルを実行している間に読出しデータを提供することができるためです（関連する遅延は、メインメモリから1ワードをフェッチする際の遅延です）。

  DCACHEマスタバスで生成されるバーストの種類は、DCACHE_CRのHBURSTビットによって決まり、INCRwまたはWRAPwです（wはワード単位のキャッシュライン幅です）。

  AHBトランザクション属性も、スレーブ入力（ミスした）リクエストからマスタ出力リフィルリクエストに伝播されます。
- アドレスがタグメモリ内にないが、現在行われているメインメモリからのリフィルバーストに属している場合：同様に**キャッシュ読出しヒット**（ヒットアンダーミス機能）。

  ラインリフィルが読出しと書込みのいずれの（ミスした）トランザクションであっても、DCACHEは、データがマスタインタフェースで使用可能になるとすぐに、リクエストされた読出しデータを提供できるため、ミス（メインメモリからのデータフェッチ）を回避できます。

ライトバック·トランザクションの場合も（ライトバックバッファ可能属性を持つ書込みトランザクション）、3通りの方法があります。

- アドレスがタグメモリ内にあり、キャッシュラインが有効な場合：**キャッシュ·ライトバック·ヒット**。データがキャッシュに書き込まれます。
- アドレスがタグメモリ内にない（またはキャッシュラインが有効でない）場合：**キャッシュ·ライトバック·ミス**。

  まず、キャッシュライン全体のデータをメインメモリから読み出すことによって、ライン割り当てが行われます。このラインリフィルのためにDCACHEマスタバスで生成されるバーストの種類は、DCACHE_CRのHBURSTビットによって決まり、INCRwまたはWRAPwであり（wはワード単位のキャッシュライン幅）、AHBトランザクション属性がスレーブポートの最初のリクエストから伝播されます。

  リフィルされたラインがDCACHEに書き込まれると、スレーブポートで提供された最初のデータが、このDCACHEラインに書き込まれます（直前にリフィルされたキャッシュラインのデータ部分を上書きします）。
- アドレスがタグメモリ内にないが､現在行われているメインメモリからのリフィルバーストに属している場合：同様に**キャッシュ·ライトバック·ヒット**（ヒットアンダーミス機能）。

  ラインリフィルが読出しと書込みのいずれの（ミスした）トランザクションによる場合でも、DCACHEは、着信したデータをリフィルされたラインに直接書き込んで、ミス（メインメモリからのリフィル）を回避できます。

ライトスルートランザクション（ライトスルーバッファ可能属性を持つ書込みトランザクション）の場合、2通りの方法があります。

- アドレスがタグメモリ内にあり、キャッシュラインが有効な場合：**キャッシュ·ライトスルー·ヒット**。データがキャッシュとメインメモリの両方に書き込まれます（マスタポートを経由）。
- アドレスがタグメモリ内にない（またはキャッシュラインが有効でない）場合：**キャッシュ·ライトスルー·ミス**。スレーブポートで受信したデータは、メインメモリにのみ書き込まれます（ライトバックミスと違って、ライン割当てはなく、データがキャッシュに書き込まれることもありません）。

（キャッシュミスによる）キャッシュリフィルの場合、DCACHEは、リフィルデータを書き込むキャッシュラインを選択します。2ウェイセットアソシアティブキャッシュとして、2つのラインのうちの1つを使用できます（2ウェイのそれぞれのアドレスインデックスで指し示されたライン）。ウェイ選択は、pLRU-t置換アルゴリズムに基づき、これは、各インデックスについて、次のリフィルのためのウェイ候補を指し示します。

リフィルデータを書き込む必要があるキャッシュラインがすでに有効な場合でも、ターゲットのキャッシュラインを最初に無効化する必要があります。

- このラインのダーティタグが0に等しい場合（クリーンデータ）、ラインは単に無効化されます。
- 1に等しい場合（ダーティデータ）、ラインをメインメモリに書き戻す必要があります。

  DCACHEはマスタポートにバースト書込みトランザクションを生成します。これは、バーストタイプがINCRw（wはワード単位のキャッシュライン幅）に設定され、AHBメモリトランザクション属性信号が次のように設定されています。
  - データ（命令ではない）
  - 特権 = TAG特権ビット
  - ライトバック（関係ない場合でも）
  - 通常メモリ
  - キャッシュ可能
  - 割り当て（関係ない場合でも）
  - 共有不可

これらのAHB属性をスレーブポートから伝播することはできません（DCACHEによって送信される他のすべてのトランザクションの場合と違って）。無効化トランザクションは最初のミストランザクションとは関係がないためです。AHB属性の設定は固定されますが、特権ビットは無効化されたラインのTAG特権ビットからコピーされます。

9.4.7 DCACHEのセキュリティ

DCACHEは、Armv8-M TrustZoneを実装します。

DCACHEの設定レジスタは、システムレベルで保護されます。

9.4.8 DCACHEのメンテナンス

DCACHEはいくつかのメンテナンス操作を備え、ソフトウェアがDCACHE_CR制御レジスタでプログラムできます。

- **完全検証**：キャッシュ全体を検証します。割込み不可タスクです。

  ソフトウェアは、DCACHE_CRのCACHEINVをプログラムすることによって、DCACHEの内容全体を検証できます。

  CACHEINV = 1のとき、DCACHE制御ロジックはDCACHE_SRステータスレジスタのBUSYFフラグをセットし、キャッシュ無効化操作を実行して、各TAG有効ビットを0にリセットします（キャッシュラインごとに1つの有効ビット）。次のキャッシュライン検証時の未知の値を避けるために、キャッシュ無効化時には、各ダーティおよび特権ビットもリセットされます。CACHEINVは自動的にクリアされます。

  完全無効化操作が終了すると、DCACHEはBUSYFフラグを自動的にクリアして、DCACHE_SRのBSYENDFをセットします。

  このフラグ条件で有効化された場合（DCACHE_IERのBSYENDIE = 1）、DCACHE割込みがレイズされます。すると、（空の）キャッシュを再び使用できるようになります。

  この完全無効化操作は割込み不可です。つまり、BUSYF = 1の間は、キャッシュはキャッシュ可能リクエストを扱いません。ただし、キャッシュ不可トラフィックは扱われ（リクエストアドレスがTAGアドレスと比較されないため）、DCACHEは同じクロックサイクルでバイパスされます（DCACHEが無効なときと同じ動作）。

- **範囲無効化**：キャッシュ内の特定の範囲のアドレスを無効化します。バックグラウンドタスク（割込み可能）。

  ソフトウェアは、アドレス範囲がDCACHE_CMDRSADDRR（範囲開始アドレス）とDCACHE_CMDREADDRR（範囲終了アドレス）にプログラムされた後、DCACHE_CR のSTARTCMD = 1およびCACHECMD = 0b010をプログラムすることによって、DCACHE内の特定のデータ領域を無効化できます。

  その場合、DCACHE制御ロジックはTAGメモリ全体を解析します。読出しラインアドレス（タグアドレス + ラインインデックス）が、プログラムされたアドレス範囲に該当する場合（DCACHE_CMDRSADDRR ≤ Line Addr ≤ DCACHE_CMDREADDRR）、対応するキャッシュラインは無効化されます（ラインのタグビットがクリアされ、有効ビット = ダーティビット = 特権ビット = 0になります）。

  STARTCMDがセットされると、DCACHE制御ロジックはDCACHE_SRのBUSYCMDFをセットし、範囲無効化操作を開始します。STARTCMDも自動的にクリアされます。

  操作が完了すると（すべてのTAGメモリが解析されると）、DCACHEはDCACHE_SRのBUSYCMDFを自動的にクリアし、CMDENDFをセットします。

  このフラグ条件で有効化されると（DCACHE_IERのCMDENDIE = 1）、DCACHE割込みがレイズされます。

  範囲無効化操作中、DCACHEは割込み可能です。つまり、無効化プロセスより優先度の高い新しい着信リクエストを受け入れることができます。TAGメモリは、外部キャッシュリクエストに

よってすでにアクセスされていない場合のみ、範囲無効化操作のためにアクセスされます。このため、範囲無効化の実行は、通常、一気に行われるのではなく、割り込まれることがあります。

- **範囲クリーン**: キャッシュ内の特定の範囲のアドレスをクリーンします。バックグラウンドタスク（割込み可能）。

  キャッシュラインのクリーンとは、メインメモリの内容が最新であり、キャッシュで変更された可能性のあるデータも含んでいることを確認することを意味します。クリーン操作では、「ダーティ」としてタグ付けされているキャッシュライン（TAGダーティビットがセットされているもの）がメインメモリにライトバックされます。

  ソフトウェアは、アドレス範囲がDCACHE_CMDRSADDRR（範囲開始アドレス）とDCACHE_CMDREADDRR（範囲終了アドレス）にプログラムされた後、DCACHE_CR のSTARTCMD = 1およびCACHECMD = 0b001をプログラムすることによって、DCACHE内の特定のデータ領域をクリーンできます。

  その場合、DCACHE制御ロジックはTAGメモリ全体を解析します。読出しラインアドレス（タグアドレス + ラインインデックス）が、プログラムされたアドレス範囲に該当する場合（DCACHE_CMDRSADDRR ≤ Line Addr ≤ DCACHE_CMDREADDRR）で、対応するラインがダーティの場合、このラインはクリーンになります。つまり、キャッシュライン全体がDCACHEマスタポートを経由してメモリに書き戻され、タグのダーティビットがクリアされます。

  STARTCMDがセットされると、DCACHE制御ロジックはDCACHE_SRのBUSYCMDFをセットし、範囲クリーン操作を開始します。STARTCMDも自動的にクリアされます。

  操作が完了すると（すべてのTAG メモリが解析されると）、DCACHE はDCACHE_SR のBUSYCMDFを自動的にクリアし、CMDENDFをセットします。

  このフラグ条件で有効化されると（DCACHE_IERのCMDENDIE = 1）、DCACHE割込みがレイズされます。

  範囲クリーン操作中、DCACHEは割込み可能です。つまり、クリーニングプロセスより優先度の高い新しい着信リクエストを受け入れることができます。TAGメモリは、外部キャッシュリクエストによってすでにアクセスされていない場合のみ、範囲クリーン操作のためにアクセスされます。このため、範囲クリーンの実行は、通常、一気に行われるのではなく、割り込まれることがあります。

  範囲クリーンが完了するまで、クリーンされる領域の内容をバスイニシエータが変更を試みないようにするのは、ソフトウェアの責任です。そのため、ソフトウェアは、DCACHE_SRのBUSYCMDFフラグを利用でき、このフラグにポーリングして、クリーンされる領域への疑わしいアクセスを防ぐことができます。

  または、コマンド終了フラグ（CMDENDF）あるいはDCACHE割込みを利用して、範囲クリーンの実行終了を検出することもできます。

- **範囲のクリーンおよび無効化**: キャッシュ内の特定の範囲のアドレスをクリーンして無効化します。バックグラウンドタスク（割込み可能）。

  この操作は、操作アドレス範囲（範囲クリーン操作と同じ）に属する「ダーティ」なキャッシュラインをクリーンして、このアドレス範囲に属するすべての（有効な）キャッシュラインを（ダーティかどうかに関係なく）無効化します。

  ソフトウェアは、アドレス範囲がDCACHE_CMDRSADDRR（範囲開始アドレス）とDCACHE_CMDREADDRR（範囲終了アドレス）にプログラムされた後、DCACHE_CR のSTARTCMD = 1およびCACHECMD = 0b011をプログラムすることによって、この範囲のクリーンおよび無効化操作を開始できます。

  これは、範囲無効化操作または範囲クリーン操作と同じフラグ、および潜在的に同じ割込みをセットおよびクリアします。

9.4.9 DCACHE性能の監視

DCACHEは、パフォーマンス分析のために以下のモニタを備えています。

- 2つの32bitの読出しヒットおよび書込みヒットモニタは、DCACHE入力（スレーブポート）で、DCACHE出力（マスタポート）でトランザクションを生成しないAHBトランザクションをカウントします。

  これらのモニタは、アドレスがTAGメモリ、またはリフィルバッファにある（以前のミスのため、そのデータがキャッシュマスタポートから受信されるか、まもなく受信される）すべてのアクセスも考慮します（セクション9.4.6を参照）。
- 2 つの 16bit の読出しミスおよび書込みミスモニタは、DCACHE 入力（スレーブポート）で、DCACHE出力（マスタポート）でトランザクションを生成するAHBトランザクションをカウントします。

  これらのモニタは、アドレスがTAGメモリにもリフィルバッファにも存在しないすべてのアクセスも考慮します。

最大値に達しても、これらのモニタは最小値に戻りません。

ソフトウェアは次のタスクを実行できます。

- DCACHE_CRのR(W)HITMENを通じて、読出し（書込み）ヒットモニタを有効化/停止します。
- DCACHE_CRのR(W)HITMRSTを通じて、読出し（書込み）ヒットモニタをリセットします。
- DCACHE_CRのR(W)MISSMENを通じて、読出し（書込み）ミスモニタを有効化/停止します。
- DCACHE_CRのR(W)MISSMRSTを通じて、読出し（書込み）ミスモニタをリセットします。

消費電力を低減するために、これらのモニタはデフォルトでは無効化（停止）されています。

9.4.10 DCACHEブート

DCACHEはブート時に無効化されます（DCACHE_CRのEN = 0）。

ブートが終了すると、DCACHEを有効にできます（ソフトウェアでDCACHE_CRのEN = 1をセット）。

9.5 DCACHE低電力モード

製品レベルでは、DCACHEの使用により、たいていの場合、サイズが大きく、そのために消費電力も多いメインメモリからではなく、内部DCACHEとの間でデータのロード/格納を行うことによって、消費電力が低減されます。キャッシュされるメインメモリが外部メモリの場合、低減量はさらに大きくなります。

9.6 DCACHEのエラー管理と割込み

DCACHEマスタポートで開始されたトランザクションがエラーを返すことがあります（読出し専用メモリに書込みを試みた場合など）。マスタポートリクエストがスレーブポートリクエストに伝播された場合、エラーはスレーブポートに再伝播されます。マスタポートリクエストがDCACHE自体によって開始された場合でも（エビクションまたはクリーン操作により、キャッシュラインがメインメモリに書き戻される）、DCACHEはこの機能エラーを受信し、DCACHE_SRのERRFフラグをセットすることによって、内部でフラグを立てます。

そのような場合、対応する割込みイネーブルビットがセットされていた場合（DCACHE_IERのERRIE = 1）、割込みが生成されます。.

割込みが生成される可能性のある、もう1つのケースは、完全無効化操作の終了時です。つまり、キャッシュビジー状態が終了し、DCACHEがDCACHE_SRのBSYENDFフラグをセットしたときです。

対応する割込みイネーブルビットがセットされていた場合（DCACHE_IERのBSYENDIE = 1）、割込みが生成されます。

最後のケースは、範囲無効化またはクリーン操作の終了時です。つまり、コマンドビジー状態が終了し、DCACHEがDCACHE_SRのCMDENDFフラグをセットしたときです。

対応する割込みイネーブルビットがセットされていた場合（DCACHE_IERのCMDENDIE = 1）、割込みが生成されます。

すべてのDCACHE割込みソースは、同一かつ一意な割込み信号dcache_itをレイズし、同じ割込みベクタを使用します。

表93.DCACHE割込み

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法
DCACHE	機能エラー	DCACHE_SRのERRF	DCACHE_IERのERRIE	DCACHE_FCRのCERRFを1にセット
	ビジー状態の終了（完全無効化の終了）	DCACHE_SRのBSYENDF	DCACHE_IERのBSYENDIE	DCACHE_FCRのCBSYENDFを1にセット
	キャッシュ操作の終了（アドレス範囲ベース）	DCACHE_SRのCMDENDF	DCACHE_IERのCMDENDIE	DCACHE_FCRのCCMDENDFを1にセット

また、DCACHEは、すべてのAHBバスエラー（セキュリティ問題、アドレスデコード問題など）をマスタポートからS-AHBスレーブポートに再伝播します。

9.7 DCACHEレジスタ

9.7.1 DCACHE制御レジスタ(DCACHE_CR)

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HBURST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WMISSMRST	WHITMRST	WMISSMEN	WHITMEN	RMISSMRST	RHITMRST	RMISSMEN	RHITMEN
rw								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	STARTCMD	CACHECMD[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	CACHEINV	EN
				w	rw	rw	rw							w	rw

ビット31 **HBURST**：キャッシュマスタポート読出しアクセスの出力バーストタイプ
EN = 0のときのみ、ソフトウェアによってセットできます。
マスタポート書込みアクセスは、常にINCRバーストタイプで行われます。
0：WRAP
1：INCR

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **WMISSMRST**：書込みミスモニタのリセット
0：影響なし。
1：キャッシュ書込みミスモニタをリセットします。

ビット22 **WHITMRST**：書込みヒットモニタのリセット
0：影響なし。
1：キャッシュ書込みヒットモニタをリセットします。

ビット21 **WMISSMEN**：書込みミスモニタイネーブル
0：キャッシュ書込みミスモニタのスイッチを切ります。モニタは停止してもリセットされません。
1：キャッシュ書込みミスモニタを有効にします。

ビット20 **WHITMEN**：書込みヒットモニタイネーブル
0：キャッシュ書込みヒットモニタのスイッチを切ります。モニタは停止してもリセットされません。
1：キャッシュ書込みヒットモニタを有効にします。

ビット19 **RMISSMRST**：読出しミスモニタのリセット
0：影響なし。
1：キャッシュ読出しミスモニタをリセットします。

ビット18 **RHITMRST**：読出しヒットモニタのリセット
0：影響なし。
1：キャッシュ読出しヒットモニタをリセットします。

ビット17 **RMISSMEN**：読出しミスモニタイネーブル
0：キャッシュ読出しミスモニタのスイッチを切ります。モニタは停止してもリセットされません。
1：キャッシュ読出しミスモニタを有効にします。

ビット16 **RHITMEN**：読出しヒットモニタイネーブル
0：キャッシュ読出しヒットモニタのスイッチを切ります。モニタは停止してもリセットされません。
1：キャッシュ読出しヒットモニタを有効にします。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **STARTCMD**：メンテナンスコマンドを開始します（CACHECMDで定義されたメンテナンス操作）。
EN = 1、BUSYCMDF = 0、BUSYF = 0、およびCACHECMD = 0b001、0b010、または0b011のときのみ、ソフトウェアによって設定できます。
BUSYCMDFフラグがセットされたときに（キャッシュメンテナンス操作中）、ハードウェアによってクリアされます。0を書き込んでも、ビットの値は変化しません。
0：コマンド操作（キャッシュメンテナンス）終了
1：メンテナンスコマンド（キャッシュメンテナンス）開始

ビット10:8 **CACHECMD[2:0]**：キャッシュコマンドメンテナンス操作（アドレス範囲をクリーンまたは無効化）
メンテナンスコマンドが進行中でないときのみ（BUSYCMDF = 0）、ソフトウェアによってセットおよびクリアできます。
000：操作なし
001：範囲クリーン
010：範囲無効化
011：範囲クリーンおよび無効化
その他：予約済み

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CACHEINV**：完全キャッシュ無効化
EN = 1のときのみ、ソフトウェアによってセットできます。
BUSYFフラグがセットされたとき（完全キャッシュ無効化操作中）、ハードウェアによってクリアされます。0を書き込んでも、ビットの値は変化しません。
0：影響なし。
1：キャッシュ全体を無効化（インバリデート）します（すべてのキャッシュラインの有効ビット = 0）。

ビット0 **EN**：イネーブル

0：キャッシュは無効です。（ディセーブル）

1：キャッシュは有効です。（イネーブル）

9.7.2 DCACHEステータスレジスタ(DCACHE_SR)

アドレスオフセット：0x004

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDENDF	BUSYCMDF	ERRF	BSYENDF	BUSYF
											r	r	r	r	r

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CMDENDF**：コマンド終了フラグ

DCACHE_FCR.CCMDENDF = 1を書き込むことによってクリアされます。

0：キャッシュはビジーまたはアイドルです。

1：CACHECMDコマンド終了

ビット3 **BUSYCMDF**：コマンドビジーフラグ

0：キャッシュはCACHECMDコマンドでビジーではありません。

1：キャッシュはCACHECMDコマンド（アドレス範囲のクリーンまたは無効化）でビジーです。

ビット2 **ERRF**：キャッシュエラーフラグ

DCACHE_FCR.CERRF = 1を書き込むことによってクリアされます。

0：エラーはありません。

1：操作中にエラーが発生しました（エビクションまたはクリーン操作ライトバックエラー）。

ビット1 **BSYENDF**：完全無効化ビジー終了フラグ

DCACHE_FCR.CBSYENDF = 1を書き込むことによってクリアされます。

0：キャッシュはビジーまたはアイドルです。

1：完全無効化CACHEINV操作が終了しました。

ビット0 **BUSYF**：完全無効化ビジーフラグ

0：キャッシュはCACHEINV操作でビジーではありません。

1：キャッシュは完全無効化CACHEINV操作を実行中です。

9.7.3 DCACHE割込みイネーブルレジスタ(DCACHE_IER)

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDENDIE	Res.	ERRIE	BSYENDIE	Res.
											rw		rw	rw	

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CMDENDIE**：コマンド終了時の割込みイネーブル

キャッシュコマンド（アドレス範囲のクリーンまたは無効化）の終了時に割込み生成を有効にするために、ソフトウェアによってセットされます。

0：コマンド終了時の割込みは無効です。

1：コマンド終了時の割込みは有効です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ERRIE**：キャッシュエラー時の割込みイネーブル

キャッシュ機能エラー（エビクションまたはクリーン操作ライトバックエラー）発生時に割込み生成を有効にするために、ソフトウェアによってセットされます。

0：エラー時の割込みは無効です。

1：エラー時の割込みは有効です。

ビット1 **BSYENDIE**：ビジー終了時の割込みイネーブル

キャッシュの完全無効化操作の終了時に割込み生成を有効にするために、ソフトウェアによってセットされます。

0：ビジー終了時の割込みは無効です。

1：ビジー終了時の割込みは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

9.7.4 DCACHEフラグクリアレジスタ(DCACHE_FCR)

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCMDENDF	Res.	CERRF	CBSYENDF	Res.
											w		w	w	

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CCMDENDF**：コマンド終了フラグのクリア
ソフトウェアによってセットされます。
0：影響なし。
1：DCACHE_SRのCMDENDFフラグをクリアします。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CERRF**：キャッシュエラーフラグのクリア
ソフトウェアによってセットされます。
0：影響なし。
1：DCACHE_SRのERRFフラグをクリアします。

ビット1 **CBSYENDF**：完全無効化ビジー終了フラグのクリア
ソフトウェアによってセットされます。
0：影響なし。
1：DCACHE_SRのBSYENDFフラグをクリアします。

ビット0 予約済みであり、リセット値に保持する必要があります。

9.7.5 DCACHE読出しヒットモニタレジスタ(DCACHE_RHMONR)

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RHITMON[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RHITMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RHITMON[31:0]**：キャッシュ読出しヒットモニタカウンタ

9.7.6 DCACHE読出しミスモニタレジスタ(DCACHE_RMMONR)

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RMISSMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RMISSMON[15:0]**：キャッシュ読出しミスモニタカウンタ

9.7.7 DCACHE書込みヒットモニタレジスタ(DCACHE_WHMONR)

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WHITMON[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WHITMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **WHITMON[31:0]**：キャッシュ書込みヒットモニタカウンタ

9.7.8 DCACHE書込みミスモニタレジスタ(DCACHE_WMMONR)

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WMISSMON[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **WMISSMON[15:0]**：キャッシュ書込みミスモニタカウンタ

9.7.9 DCACHEコマンド範囲開始アドレスレジスタ（DCACHE_CMDRSADDRR）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMDSTARTADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMDSTARTADDR[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:4 **CMDSTARTADDR[31:4]**：DCACHE_CR.CACHECMDフィールドで指定されたキャッシュメンテナンスコマンドが適用される範囲の開始アドレス

このレジスタは、DCACHE_CR.CACHECMDが書き込まれる前にセットする必要があります。

注：ビット4はSTM32U59x/5Ax/5Fx/5Gxで予約済みです。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

9.7.10 DCACHEコマンド範囲終了アドレスレジスタ（DCACHE_CMDREADDRR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMDENDADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMDENDADDR[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:4 **CMDENDADDR[31:4]**：DCACHE_CR.CACHECMDフィールドで指定されたキャッシュメンテナンスコマンドが適用される範囲の終了アドレス

このレジスタは、DCACHE_CR.CACHECMDが書き込まれる前にセットする必要があります。

注：ビット4はSTM32U59x/5Ax/5Fx/5Gxで予約済みです。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

9.7.11 DCACHEレジスタマップ

表94.DCACHEレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DCACHE_CR	HBURST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WMISSMRST	WHITMRST	WMISSMEN	WHITMEN	RMISSMRST	RHITMRST	RMISSMEN	RHITMEN	Res.	Res.	Res.	Res.	STARTCMD	CACHECMD[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	CACHEINV	EN
	リセット値	0								0	0	0	0	0	0	0	0					0	0	0	0							0	0
0x004	DCACHE_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDENDF	BUSYCMDF	ERRF	BSYENDF	BUSYF
	リセット値																												0	0	0	0	1
0x008	DCACHE_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDENDIE	Res.	ERRIE	BSYENDIE	Res.
	リセット値																												0		0	0	
0x00C	DCACHE_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCMDENDF	Res.	CERRF	CBSYENDF	Res.
	リセット値																												0		0	0	
0x010	DCACHE_RHMONR	RHITMON[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	DCACHE_RMMONR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RMISSMON[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	DCACHE_WHMONR	WHITMON[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	DCACHE_WMMONR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WMISSMON[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表94.DCACHEレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x028	DCACHE_CMDRSADDRR	CMDSTARTADDR[31:4]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x02C	DCACHE_CMDREADDRR	CMDENDADDR[31:4]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				

レジスタ境界アドレスについては、セクション2.3を参照してください。

10 電源制御（PWR）

10.1 PWRの概要

電源コントローラは、すべてのデバイスの電源と電力モードの遷移を管理します。

10.2 PWRの主な機能

電源コントローラ（PWR）の主な機能を以下に示します。

- 電源供給と電源ドメイン
  - コアドメイン(V_{CORE})
  - V_{DD}ドメイン
  - バックアップドメイン
  - アナログドメイン（V_{DDA}）
  - SMPSパワーステージ用の電源（SMPSパッケージで使用可能）
  - ポートPG[15:2]上のV_{DDIO2}ドメイン
  - USBトランシーバ用のV_{DDUSB}およびオプションの$V_{DD11USB}$
  - DSIトランシーバ用のV_{DDDSI}および$V_{DD11DSI}$（STM32U59x/5Ax/5Fx/5Gxの場合のみ）
- システム電源の電圧レギュレーション
  - SMPSステップダウンコンバータ
  - リニア電圧レギュレータ(LDO)
- 電源供給の監視
  - BORモニタ
  - PVDモニタ
  - PVMモニタ（V_{DDA}、V_{DDUSB}、V_{DDIO2}）
  - 機能範囲外温度モニタ
  - 機能範囲外バックアップドメイン電圧モニタ
- 電源管理
  - 動作モード
  - 電圧スケーリングの制御
  - 低電力モード
- V_{BAT}バッテリの充電
- TrustZoneセキュリティおよび特権保護

10.3 PWRピンおよび内部信号

表95.PWRの入出力ピン

ピン名	信号タイプ	説明
VDD	電源	メイン電源
GND	電源	メイングラウンド
VDDA	電源	アナログペリフェラル電源
VSSA	電源	アナログペリフェラルグラウンド
VDDIO2	電源	独立I/O電源
VDDUSB	電源	USB/OTG_FS/OTG_HS電源
VDD11USB(1)	電源	OTG_HSトランシーバ電源（オプション）
VDD11（パッケージにSMPSを含む）/VCAP（パッケージにSMPSを含まない）	電源	ロジック電源(V_{CORE})
VBAT	電源	バックアップドメイン電源
VDDDSI（1）	電源	DSI電源
VDD11DSI（1）	電源	DSIトランシーバ電源
VDDSMPS	電源	SMPS電源
VSSSMPS	電源	SMPSグラウンド
VLXSMPS	電源	SMPS出力
VREF+	電源	ADC/DAC高基準電圧
VREF-	電源	ADC/DAC低基準電圧
WKUPx（x = 1？8）	入力	ウェイクアップピン
CSLEEP	出力	SLEEPモードのMCU
CDSTOP	出力	STOPモードでのCPUドメイン
SRDSTOP	出力	STOPモードでのSmartRunドメイン

1. STM32U59x/5Ax/5Fx/5Gxデバイスでのみ使用できます。

表96.PWR内部入力/出力信号

内部信号名	信号タイプ	説明
WKUPx_y（x = 1 ~ 8、y = 1 ~ 4）	入力	ウェイクアップイベントソース選択

8つのウェイクアップイベント（WKUPx）は、PWR_WUCR3のWUSELx[1:0]（x = 1 ~ 8）によって選択された、それぞれ4つのピンまたは内部イベントから生成できます。

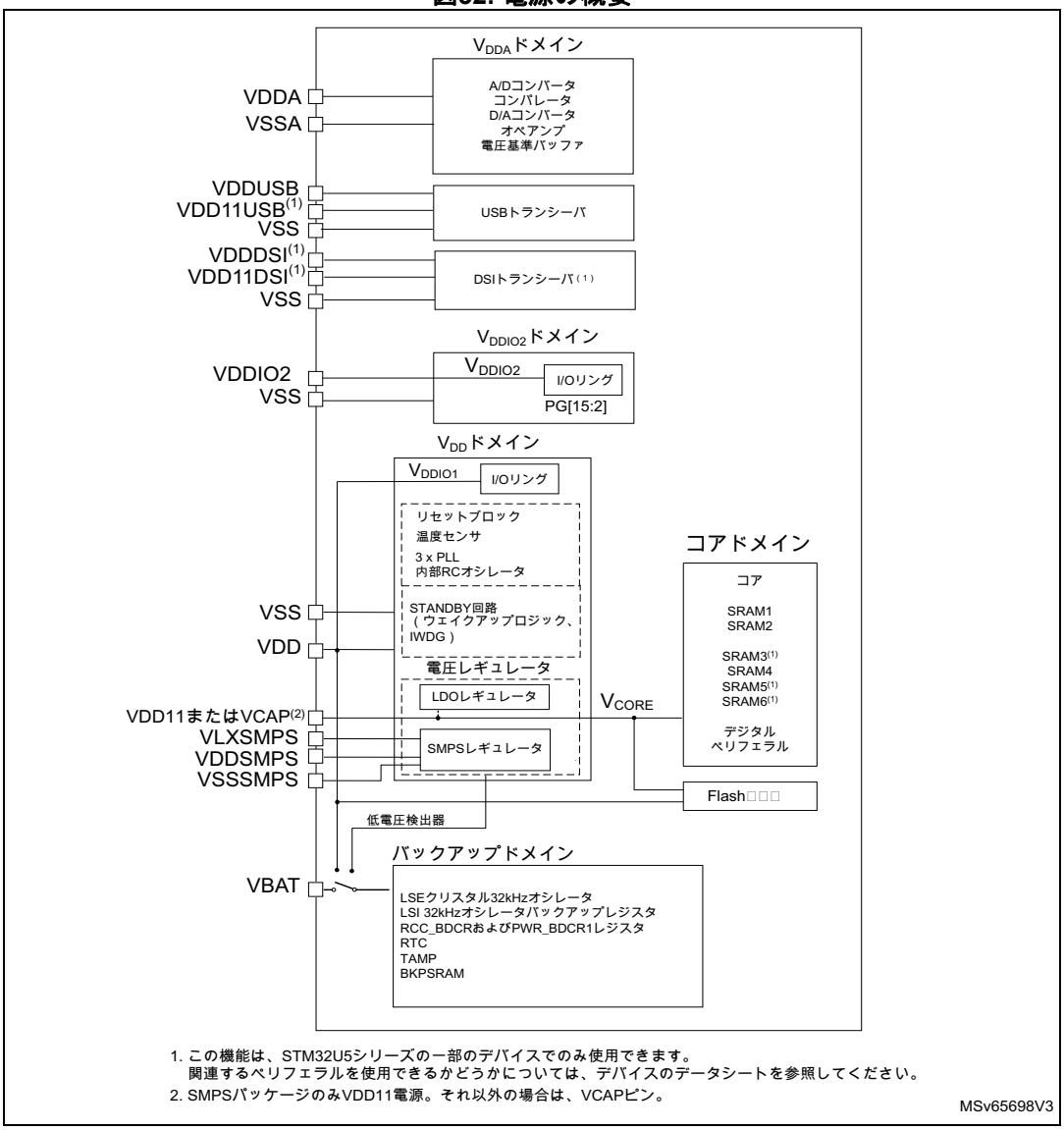
表97.PWRウェイクアップソース選択

ウェイクアップイベント	内部信号ソース（x = 1 ~ 8）			
	WKUPx_0（WUSELx = 00）	WKUPx_1（WUSELx = 01）	WKUPx_2（WUSELx = 10）	WKUPx_3（WUSELx = 11）
WKUP1	PA0	PB2	PE4	予約済み
WKUP2	PA4	PC13	PE5	予約済み
WKUP3	PE6	PA1	PB6	予約済み
WKUP4	PA2	PB1	PB7	予約済み
WKUP5	PC5	PA3	PB8	アーリー IWDG割込み(1)
WKUP6	PB5	PA5	PE7	RTC_ALRA_S、RTC_ALRB_S、RTC_WUT_S、またはRTC_TS_S
WKUP7	PB15	PA6	PE8	RTC_ALRA、RTC_ALRB、RTC_WUT、またはRTC_TS
WKUP8	PF2	PA7	PB10	TAMP

1. STM32U575/585を除くすべてのSTM32U5シリーズで使用可能な相互接続。

10.4 PWR電源供給と電源ドメイン

図32. 電源の概要



1. この機能は、STM32U5シリーズの一部のデバイスでのみ使用できます。
   関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。
2. SMPSパッケージのみVDD11電源。それ以外の場合は、VCAPピン。

MSv65698V3

10.4.1 外部電源

デバイスには、1.71V？3.6VV_{DD}動作電圧が必要です。特定のペリフェラルには、いくつかの独立した電源を提供できます。これらの電源は、VDDピン上の有効な動作電源なしに供給することはできません。

- V_{DD} = 1.71V ~ 3.6V（V_{BOR}最小値まで機能を保証）

  V_{DD}は、I/O、内部レギュレータ、およびリセット、電源管理、ならびに内部クロックなどのシステムアナログの外部電源です。VDDピンから外部より供給されます。

- V_{DDA} = 1.58V（COMP）/ 1.6V（DAC、OPAMP）/ 1.62V（ADC）/ 1.8V（VREFBUF）~ 3.6V

  V_{DDA}は、A/Dコンバータ、D/Aコンバータ、電圧リファレンスバッファ、オペアンプおよびコンパレータの外部アナログ電源です。V_{DDA}電圧レベルはV_{DD}電圧から独立しており、これらのペリフェラルを使用しない場合は、VDDまたはVSS（VDDを推奨）に接続する必要があります。

- V_{DDSMPS} = 1.71？3.6V

  V_{DDSMPS}は、SMPSステップダウンコンバータ用の外部電源です。VDDSMPS電源ピンから外部より供給され、SMPSがアプリケーションで使用されたときに、VDDピンと同じ電源に接続する必要があります。SMPSを使用しないときは、V_{DDSMPS}とV_{LXSMPS}の両方をGNDに接続することを推奨します。

- V_{LXSMPS}は、SMPSステップダウンコンバータの切替え後の出力です。

注： **SMPS電源ピンは、SMPSステップダウンコンバータオプション搭載の特定パッケージでのみ使用できます。**

- V_{DDUSB} = 3.0 ~ 3.6V

  V_{DDUSB}は、USB/OTG_FS/OTG_HSトランシーバ用の独立した外部電源です。V_{DDUSB}の電圧レベルは、V_{DD}電圧から独立しており、USBが使用されないときは、V_{DD}に接続することを推奨します。

- $V_{DD11USB}$ = 1.0V ~ 1.26V（STM32U59x/5Ax/5Fx/5Gxデバイスでのみ使用可能）

  $V_{DD11USB}$はOTG_HSトランシーバの外部電源です。この電源は、特定パッケージでのみ使用でき、VDD11に接続する必要があります。

- V_{DDIO2} = 1.08 ~ 3.6V

  V_{DDIO2}は14個のI/O（ポートG[15:2]）の外部電源です。V_{DDIO2}の電圧レベルは、V_{DD}電圧から独立しており、PG[15:2]を使用しない場合は、V_{DD}に接続することを推奨します。

- V_{BAT} = 1.65V ~ 3.6V（$_{BOR_VBAT}$最小値までの機能を保証、製品データシートを参照）

  V_{BAT}は、RTC、TAMP、外部/内部クロックの32kHzオシレータ、バックアップレジスタ、およびオプションのバックアップSRAMのV_{DD}が存在しない場合の電源です（電源スイッチ経由）。

- V_{DDDSI} = 1.71V ~ 3.6V（STM32U59x/5Ax/5Fx/5Gxデバイスでのみ使用可能）

  V_{DDDSI}はDSIコントローラの外部電源です。VDDDSI電源ピンから外部より供給され、VDDピンと同じ電源に接続する必要があります。

- $V_{DD11DSI}$ = 1.0V ~ 1.26V（STM32U59x/5Ax/5Fx/5Gxデバイスでのみ使用可能）

  $V_{DD11DSI}$は、DSIトランシーバの外部電源で、VDD11ピンに接続する必要があります。

- VREF-、VREF+

  V_{REF+}はADCおよびDACの入力リファレンス電圧です。有効時、内部電圧リファレンスバッファの出力にもなります。

  ADCおよびDACがアクティブでない場合、V_{REF}をアース接続できます。

  内部電圧基準バッファは次の4つの出力電圧をサポートしており、VREFBUF_CSRレジスタのVRSビットで設定します。

  – V_{REF+}約1.5V。これには、V_{DDA}≥1.8Vが必要です。

  – V_{REF+}約1.8V。これには、V_{DDA}≥2.1Vが必要です。

– V_{REF+}約2.048V。これには、V_{DDA}≥2.4Vが必要です。
– V_{REF+}約2.5V。これには、V_{DDA}≥2.8Vが必要です。

VREF-ピンおよびVREF+ピンはすべてのパッケージで使用できるわけではありません。使用可能でないとき、VSSAとVDDAにそれぞれ結合されます。

VREF+がパッケージのVDDAと二重結合されている場合、内部電圧リファレンスバッファは使用できず、無効にしておく必要があります。

V_{REF-}は、常にV_{SSA}と同じである必要があります。

10.4.2 内部レギュレータ

デバイスには2つのレギュレータが内蔵されています。デジタルペリフェラル、SRAM（BKPSRAMを除く）、および内蔵FlashメモリにV_{CORE}電源を提供する、並列の1つのLDOと1つのSMPSです。SMPSは、この電圧をVDD11（2つまたは3つのピン）上で合計4.7µF（標準）の外部キャパシタによって生成し、2.2µH（標準）の外部コイルを必要とします。

LDOはこの電圧を、外部コンデンサが合計4.7µF（標準）のVCAP（パッケージに応じて1つまたは2つのピン）で生成します。

両方のレギュレータで4つの異なる電圧を供給でき（電圧スケーリング）、STOPモードで動作できます。

SMPSからLDOへ、LDOからSMPSへのオンザフライでの切替えが可能です。

10.4.3 パワーアップおよびパワーダウン電源シーケンス

パワーアップおよびパワーダウンフェーズでは、以下の電源シーケンス要件を満たす必要があります。

- V_{DD}が1V未満のときには、他の電源(V_{DDA}、V_{DDIO2}、V_{DDUSB})はV_{DD} + 300mV未満に保つ必要があります。
- V_{DD}が1Vを上回るときには、すべての電源は独立となります。

パワーダウン·フェーズの間、マイクロコントローラに供給されるエネルギーが1mJ未満に留まっている場合に限り、一時的にDDが他の電源よりも低い電圧となっても構いません。これによって、パワーダウン過渡フェーズの間に、外部デカップリングコンデンサを異なる時定数で放電させることが可能となります。

10.4.4 独立したアナログペリフェラルの電源

ADCおよびDAC変換の精度を高め、電源の柔軟性を拡張するために、アナログペリフェラルには独立した電源が供給されます。この電源は、PCBのノイズを防ぐために、個別にフィルタしシールドすることができます。

- アナログペリフェラル用の電源入力は、独立したVDDAピンから供給されます。
- 電源のグラウンドは、独立したVSSAピンとして用意されています。

V_{DDA}電源電圧はV_{DD}とは異なる場合があります。V_{DDA}の有無は、V_{DDA} によって電源供給されるアナログペリフェラル（A/Dコンバータ、D/Aコンバータ、コンパレータ、オペアンプ、電圧基準バッファ）のいずれかを有効にする前に確認する必要があります。

リセット後、V_{DDA} によって電源供給されるADCおよびアナログスイッチ制御は、論理的および電気的に独立しているため、使用できません。V_{DDA}電源が供給されると、アナログペリフェラルを使用する前に、PWR_SVMCRレジスタのASVビットをセットして電源の分離を解除する必要があります。

V_{DDA}電源はアナログ電圧モニタ（AVM）で監視され、2つの閾値（AVM1では1.6VまたはAVM2では1.8V）と比較できます。詳細については、セクション10.6.3を参照してください。

単電源を使用している場合、ノイズフリーのV_{DDA}リファレンス電圧を確保するために、外部フィルタ回路を通してV_{DDA}をV_{DD}に外部的に接続することができます。

ADCおよびDACの基準電圧

低電圧入力および出力時の精度を確保するため、DDAより低い独立した基準電圧をVREF+に接続できます。V_{REF+} は、アナログ入力（ADC）または出力（DAC）信号のフルスケール値に相当する最高電圧です。

V_{REF+}は、外部リファレンスまたは内部バッファ付き電圧リファレンス（VREFBUF）のいずれかによって供給されます。内部電圧基準は、設定可能な電圧（1.5V、1.8V、2.048V、または2.4V）を出力できます。内部電圧基準は、VREF+ピンから外部コンポーネントに電圧を供給することもできます。詳細については、製品データシートおよびセクション 36 : 電圧基準バッファ（VREFBUF）を参照してください。

10.4.5 独立I/O電源レール

ポートG（PG[15:2]）からのI/Oの一部は、独立した電源レールから供給されます。このレールの電源は1.08～3.6Vの範囲で供給され、VDDIO2ンを通して外部から供給されます。V_{DDIO2}の電圧レベルは、V_{DD}またはV_{DDA}から完全に独立しています。VDDIO2ピンは、一部のパッケージでのみ使用できます。I/Oリストについては、関連するデバイスデータシートのピン配置図または表を参照してください。

リセット後、V_{DDIO2}によって供給されるI/Oは、論理的および電気的に独立しているため、使用できません。PG[15:2]からのI/Oを使用する前に、PWR_SVMCRレジスタのIO2SVビットをセットして分離を解除する必要があります（V_{DDIO2}電源が存在する場合）。

V_{DDIO2}電源はIO2電圧モニタ（IO2VM）で監視され、内部基準電圧（3/4V_{REFINT}、約0.9V）と比較されます。詳細については、セクション10.6.3を参照してください。

10.4.6 独立したUSBトランシーバの電源

USBトランシーバは、個別のVDDUSB電源ピンから供給されます。V_{DDUSB}の範囲は、3.0 ~ 3.6Vで、$_{VDD}$またはV_{DDA}から完全に独立しています。

リセット後、V_{DDUSB}によって供給されるUSB機能は、論理的および電気的に独立しているため、使用できません。V_{DDUSB}電源が供給されると、USB/OTG_FS/OTG_HSを使用する前に、PWR_SVMCRのUSVをセットして電源の分離を解除する必要があります。

V_{DDUSB}電源はUSB電圧モニタ（UVM）で監視され、内部基準電圧（V_{REFINT}、約1.2V）と比較されます。詳細については、セクション10.6.3を参照してください。

内部OTG_HSトランシーバ電源（STM32U59x/5Ax/5Fx/5Gxのみ）

OTG_HS高速トランシーバは、電圧スケーリングのレンジ1およびレンジ2で機能します。USB EPOD（内蔵パワー･ディストリビューション）ブースタは、OTG_HSを使用する前に有効かつレディ状態にする必要があります。PWR_VOSRのUSBPWRENおよびUSBBOOSTENビットは1にセットする必要があります（詳細については、セクション10.6.3およびセクション10.7.12を参照）。

10.4.7 バッテリバックアップドメイン

V_{DD}がオフになった場合に、バックアップレジスタの内容を保持し、RTCおよびTAMPの機能への電源供給を維持するために、VBATピンをバッテリやその他の電源から供給されるオプションのバックアップ電源に接続することができます。

バックアップドメインの電源はSWで、これはV_{DD}とV_{BAT}の間の電源スイッチの出力です。V_{DD}および$_{BAT}$電源間のスイッチは、ブラウンアウトリセット回路によって自動的に制御されます。

V_{DD}がブラウンアウトリセットの最低閾値を下回ると（V_{BOR0}）、VBATピンはRTCおよびTAMPペリフェラル、LSI および LSE オシレータに電源を供給します。バックアップ SRAM の電源は、PWR_BDCR1でBRENがセットされている場合、オプションでVBATピンから供給されます。

以下のピン機能にもVBATピンから電源が供給されます。

- RTC、TAMP、またはLSEによって設定可能なPC13、PC14、およびPC15（セクション 63.3 : RTCの機能説明を参照）
- TAMP でタンパピンとして設定されている場合は PE3、PE4、PE5、PE6、PC13、PA0、PA1、およびPC5
- RTCでRTC_OUT2出力として設定されている場合はPB2

V_{DD}がV_{BOR0}よりも高い場合、VDDピンはこれまでのすべての機能に電源を供給します。

注： **アナログ電源スイッチは限られた電流（3mA）しか流すことできないため、出力モードでの GPIO PC13 ~ PC15の使用には制限があります。最大負荷30pFで最大速度2MHzに制限する必要があり、これらのI/Oを電流ソースとして使用してはなりません（たとえば、LEDを駆動するなど）。**

警告： **$t_{RSTTEMPO}$（V_{DD}起動時の過渡期間）中やPDRの検出後、V_{BAT}とV_{DD}の間の電源スイッチはV_{BAT}に接続されたままになります。**
起動フェーズ中、V_{DD}が$t_{RSTTEMPO}$以内に規定値に達し（$t_{RSTTEMPO}$の値については、データシートを参照）、かつ$V_{DD} > V_{BAT} + 0.6V$である場合、電流はV_{BAT}までV_{DD}と電源スイッチ（V_{BAT}）の間に接続された内部ダイオードを通して注入されます。
VBATピンに接続された電源/バッテリがこの電流注入に対応できない場合は、この電源とVBATピンの間に外部低電圧降下ダイオードを接続することを強く推奨します。

外部バッテリを使用しないアプリケーションでは、V_{BAT}をV_{DD}に外部接続し、デカップリング用の100nFの外部セラミックコンデンサを接続することを推奨します。

バックアップドメインアクセス

システムリセット後、バックアップドメイン（RCCバックアップドメイン制御レジスタRCC_BDCR、RTCレジスタ、TAMPレジスタ、バックアップレジスタ、およびバックアップSRAM）は、望ましくない書込みアクセスから保護されます。

バックアップドメインへのアクセスを有効にするには、次の手順に従います。

1. RCC_AHB3ENRのPWRENビットをセットして、電源インタフェースクロックを有効にします。
2. PWR_DBPRのDBPビットをセットして、バックアップドメインへのアクセスを有効にします。

V_{BAT}バッテリの充電

V_{DD}が存在する場合、内部抵抗を通してVBATの外部バッテリを充電できます。

V_{BAT}充電は、PWR_BDCR2のVBRS値に応じて、5kΩレジスタまたは1.5kΩレジスタを通して行われます。

バッテリ充電を有効にするには、PWR_BDCR2のVBEをセットします。V_{BAT}モードで自動的に無効になります。

10.5 PWRシステム電源電圧レギュレーション

10.5.1 SMPSおよびLDO内蔵レギュレータ

デバイスには2つの内部レギュレータが内蔵されています。アプリケーションの要件に応じてアプリケーション実行時に選択できます。

- SMPSステップダウンコンバータ
- リニア電圧レギュレータ（LDO）

SMPSでは消費電力を低減できますが、一部のアプリケーションはSMPSによって生成されるノイズの干渉を受ける場合があります。これにより、アプリケーションはLDOへの切替えが必要になる場合があります。

LDOおよびSMPSレギュレータには、メインレギュレータモード（性能を重視する場合に使用）と低電力レギュレータモードの2つのモードがあります。LDOまたはSMPSは、すべての電圧スケーリングレンジと、すべてのSTOPモードで使用できます。

10.5.2 LDOおよびSMPSに対するリセット、電圧スケーリング、および低電力モード

リセット後、レギュレータはLDOで、レンジは4です。SMPSに切り替えることで、特にV_{DD}電圧が高いときに、消費電力を抑えられます。REGSELビットを設定することで、LDOからSMPS、またはSMPSからLDOに、任意のレンジで切り替えることができます。

STOPまたはSTANDBYモードの終了時のレギュレータは、低電力モードに移行するときと同じです。電圧レンジはレンジ4です。

警告： **STM32U59x/5Ax/5Fx/5Gxデバイスに限り、拡張温度範囲（85 ~ 125°C）のVOS範囲1でデバイスを使用する場合、LDOからSMPS、またはその逆に切り替える前に、システム周波数を110MHz未満に下げる必要があります。以下のように進める必要があります。**
1. システム周波数を110MHz未満に下げます。
2. 必要なレギュレータを選択するためにREGSELビットを設定します。
3. 8µs待ちます。
4. システム周波数を必要な周波数まで上げます。

10.5.3 LDOおよびSMPSステップダウンコンバータの高速起動

BORリセット後、LDOおよびSMPSレギュレータは低起動モードで起動します。この起動機能は、パワーオンリセット後に突入電流を制限するために選択します。これにより、STOPまたはSTANDBYモード終了時のウェイクアップ時間が増えます。

ただし、動作中により高速な起動を設定することができ、その設定がシステムリセットや低電力モード（SHUTDOWNおよびV_{BAT}モードを除く）からのウェイクアップ後の次の起動から使用されます。高速起動を選択するには、PWR_CR3のFSTENをセットします。

10.5.4 ダイナミック電圧スケーリングの管理

ダイナミック電圧スケーリングは、アプリケーションパフォーマンスおよび消費電力ニーズに応じて、デジタルペリフェラル（V_{CORE}）で使用する電圧の増減を行う電源管理技術です。

V_{CORE}を増加するダイナミック電圧スケーリングは、オーバーボルティングと呼ばれています。デバイスの性能を向上させます。

V_{CORE}を減少するダイナミック電圧スケーリングは、アンダーボルティングと呼ばれています。特に電源がバッテリから供給されるために制限される、ノートPCやその他のモバイルデバイスなどの節電のために実行されます。

レギュレータは次のレンジで動作します。

- レンジ1：高性能

  メインレギュレータは標準電圧1.2Vで出力します。システムクロック周波数は最大160MHzまで可能です。
- レンジ2：中 - 高性能

  メインレギュレータは標準電圧1.1Vで出力します。システムクロック周波数は最大110MHzまで可能です。
- レンジ3：中 - 低電力レンジ

  メインレギュレータは標準電圧1.0Vで出力します。システムクロック周波数は最大55MHzまで可能です。
- レンジ4：低電力レンジ

  メインレギュレータは標準電圧0.9Vで出力します。システムクロック周波数は最大25MHzまで可能です。

電圧スケーリングは、PWR_VOSRのVOS[1:0]から選択されます。EPOD（内蔵パワー･ディストリビューション）ブースタは、レンジ1とレンジ2でシステムクロック周波数を55MHzより高くする前に、有効かつレディ状態にする必要があります。

電圧スケーリングをレンジL（低電力）からレンジP（より高性能）に切り替えるためのシーケンスは次のとおりです（このときL > P）。

1. ターゲットのSYSCLKが55MHzを超える場合：
   a) 4 から 16MHz の間にブースタクロック周波数を生成するために、RCC_PLL1CFGR の PLL1MBOOST[3:0]を設定します。
   b) PLL1オシレータのクロックソースのスイッチをオンにします。
   c) PLL1クロックソース（RCC_PLL1CFGRのPLL1SRC[1:0]）を選択します。
2. PWR_VOSRのVOS[1:0]をレンジPにプログラムします。
3. PWR_VOSRのVOSRDYフラグがセットされるまで待ちます。
4. ターゲットのSYSCLKが55MHzを超える場合：
   a) PWR_VOSRのBOOSTENをセットします。このステップは、VOSプログラミング時に一緒に行うことができます。
   b) PWR_VOSRのBOOSTRDYフラグがセットされるまで待ちます。
5. レンジPの新しい周波数ターゲット（FLASH_ACRのLATENCYビットおよびRAMCFG_MxCRのWSCビット）に応じてウェイトステートの数を調整します。
6. 必要に応じてPLLを設定し、有効にします。
7. 新しいシステム周波数への設定および切替えを行います。

電圧スケーリングをレンジP（より高性能）からレンジL（低電力）に切り替えるためのシーケンスは次のとおりです（このときL > P）。

1. システム周波数をレンジLの最大周波数未満の値に下げます。
2. 新しい周波数ターゲット（FLASH_ACRのLATENCYビットおよびRAMCFG_MxCRのWSCビット）に応じてウェイトステートの数を調整します。
3. 新しいSYSCLKが55MHz以下の場合で、PWR_VOSRのBOOSTENがセットされている場合は、これをクリアします。
4. PWR_VOSRのレンジLにVOSビットをプログラムします。このステップは、BOOSTENのクリアと一緒に行うことができます。

STM32U59x/5Ax/5Fx/5Gxデバイスでのシステム周波数のステップ

STM32U59x/5Ax/5Fx/5Gxデバイスに限り、VOSレンジ1で増減する最大システム周波数は80MHzです。

VOSレンジ1の周波数を80MHzより上げるためのシーケンスは次のとおりです。

1. AHBプリスケーラを使用してシステムクロックを2分周します（RCC_CFGR2のHPRE = 0b1000）。
2. 必要に応じてPLL1を設定し、有効にします。
3. システムクロックソースとしてPLL1を選択します（RCC_CFGR1のSW = 0b11）。
4. 5µs待ちます。
5. AHBプリスケーラを1にセットします（RCC_CFGR2のHPRE = 0b0000）。

VOSレンジ1で80MHzより高い周波数で実行している場合、周波数を下げるシーケンスは次のとおりです。

1. AHBプリスケーラを使用してシステムクロックを2分周します（RCC_CFGR2のHPRE = 0b1000）。
2. 5µs待ちます。
3. 低い方のスピードクロックをシステムクロックソースとして定義します。
4. AHBプリスケーラを1にセットし直します（RCC_CFGR2のHPRE = 0b0000）。

他のVOSレンジでは、システム周波数の増減中に制限はありません。

10.6 PWR電源供給の監視

10.6.1 ブラウンアウトリセット（BOR）

このデバイスには、ブラウンアウトリセット（BOR）回路が組み込まれています。BORは、SHUTDOWNモードを除くすべての電力モードでアクティブで無効にできません。

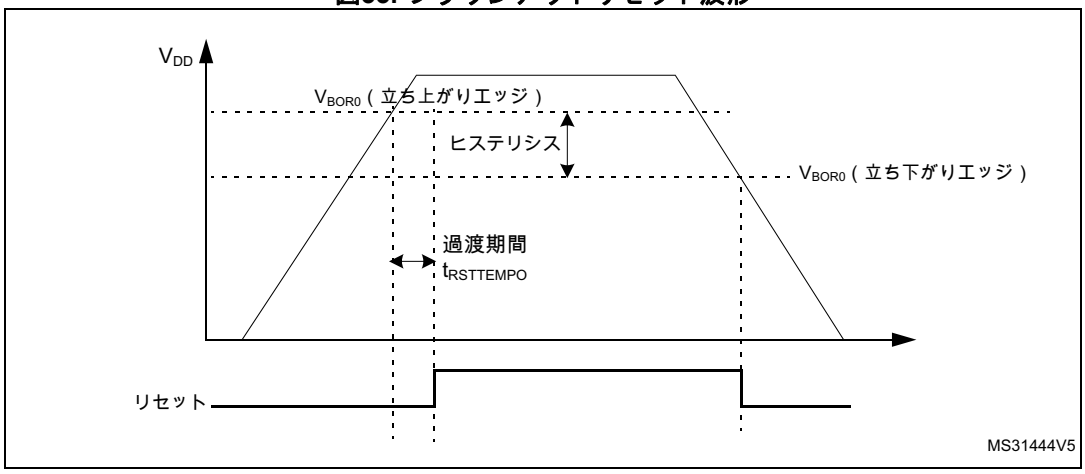
BORの閾値はオプションバイトで5種類から選択できます。BOR0は、その他の上位BORレベル選択から独立した、常に有効なパワーオン/パワーダウン機能を提供します。

パワーオン時、電源電圧V_{DD}が規定のV_{BORx}の閾値に達するまで、BORによってデバイスはリセット状態に保持されます。V_{DD}が、選択された閾値を下回ると、デバイスリセットが生成されます。V_{DD}がV_{BORx}の上限を上回る場合、デバイスリセットは解放され、システムを開始できます。

ブラウンアウトリセットの閾値の詳細については、データシートの電気的特性セクションの項を参照してください。

STANDBYモード中、BORレベル0が選択されている場合は、PWR_CR1のULPMENをセットすることで、消費電流をさらに低減するためにBORを超低電力モードにセットできます。

図33. ブラウンアウトリセット波形



1. リセットの過渡期間$t_{RSTTEMPO}$は、BOR最低閾値（V_{BOR0}）にのみ存在します。

10.6.2 プログラム可能な電圧検出器（PVD）

PVDを使用して、PWR_SVMCRのPVDLS[2:0]で選択した閾値と比較することで、V_{DD}電源を監視できます。

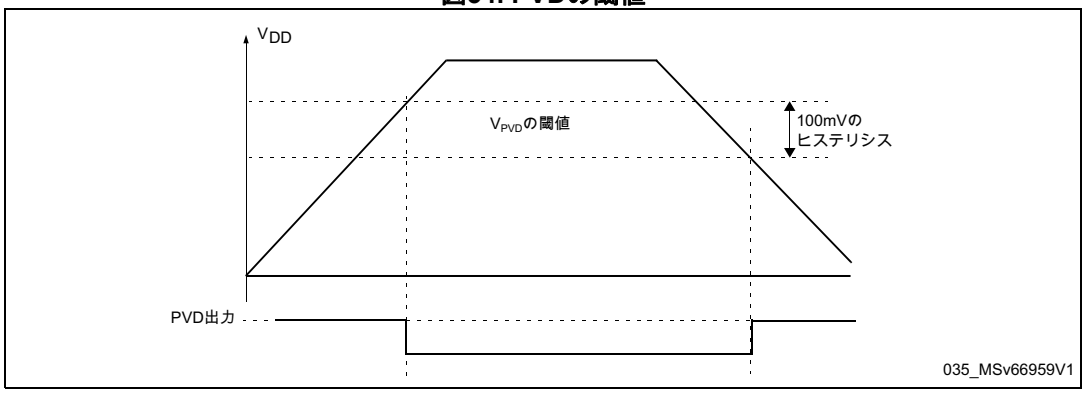
PVDはPVDEビットをセットすることで有効になります。

PWR_SVMCRのPVDOフラグを使用して、V_{DD}がPVDの閾値よりも高いか低いかを示すことができます。このイベントはEXTIに内部接続され、EXTIレジスタで有効化されている場合は、割込みリクエストを生成させることができます（表110を参照）。

EXTIラインの立ち上がり/立ち下がりエッジ感度は、PVD出力動作に応じて設定する必要があります。たとえば、EXTIラインが立ち上がりエッジ感度に設定されている場合、V_{DD}がPVD閾値を下回ったとき、割込みが生成されます。たとえば、サービスルーチンで、緊急停止処理を実行することなどが可能です。

各PVDは、STOP0、STOP1、STOP2モードでアクティブな状態のままにでき、PVM割込みでSTOPモードからウェイクアップできます。PVDはSTOP3モードでは機能しません。

図34. PVDの閾値



10.6.3 ペリフェラル電圧モニタ（PVM）

V_{DD}のみすべてのシステム関連機能に必要な唯一の電源であるため、デフォルトで監視されます。その他の電源（V_{DDA}、V_{DDIO2}、およびV_{DDUSB}）はV_{DD}から独立しており、4つのペリフェラル電圧モニタ（PVM）で監視できます。

- UVMは、USB電源V_{DDUSB}を監視します。VDDUSBRDYは、独立したV_{DDUSB} 電源がV_{UVM}閾値よりも高いか低いかを示します。
- IO2VMは、PG[15:2]電源V_{DDIO2}を監視します。VDDIO2RDYは、独立したV_{DDIO2}電源がV_{IO2VM}閾値よりも高いか低いかを示します。
- AVM1は、アナログ電源V_{DDA}を監視します。VDDA1RDYは、独立したV_{DDA} 電源がV_{AVM1}閾値よりも高いか低いかを示します。
- AVM2は、アナログ電源V_{DDA}を監視します。VDDA2RDYは、独立したV_{DDA} 電源がV_{AVM2}閾値よりも高いか低いかを示します。

各PVM出力はEXTIラインに接続され、EXTIレジスタで有効な場合は割込みを生成させることができます。独立した電源がPVM閾値を下回るか上回ったとき、あるいはその両方で、EXTIラインの立ち上がり/立ち下がりエッジの設定に応じて、PVMx出力割込みが生成されます（表110を参照）。

各PVMは、STOP0、STOP1、およびSTOP2モードでアクティブな状態のままにでき、PVM割込みでSTOPモードからウェイクアップできます。PVMはSTOP3モードでは機能しません。

表98.PVMの機能

PVM	電源	PVMの閾値
UVM	V_{DDUSB}	V_{UVM}（約1.2V）
IO2VM	V_{DDIO2}	V_{IO2VM}（約0.9V）
AVM1	V_{DDA}	V_{AVM1}（約1.6V）
AVM2	V_{DDA}	V_{AVM2}（約1.8V）

独立した電源（V_{DDA}、V_{DDIO2}、およびV_{DDUSB}）はデフォルトでは存在するとはみなされず、論理的および電気的分離を適用して、これらの専用電源が供給するペリフェラルからの情報を無視します。

- これらの電源が外部でV_{DD}に短絡された場合、アプリケーションはペリフェラル電圧モニタを有効にすることなく、それらを使用できるものと想定します。
- これらの電源がV_{DD}から独立している場合、ペリフェラル電圧モニタ（PVM）を有効にして電源が存在するかどうか確認できます。

次のシーケンスをUSB/OTG_FS/OTG_HSを使用する前に実行する必要があります。

1. V_{DDUSB}がV_{DD}から独立している場合：
   a) PWR_SVMCRのUVMENビットをセットすることによって、UVMを有効にします。
   b) UVMウェイクアップ時間を待ちます。
   c) PWR_SVMSRのVDDUSBRDYがセットされるまで待ちます。
   d) 消費電力削減のためにUVMを無効にします（オプション）。
2. PWR_SVMCRのUSVをセットして、V_{DDUSB}の電源アイソレーションを解除します。
3. STM32U59x/5Ax/5Fx/5Gxデバイスのみです。
   a) 電圧スケーリングがレンジ 1 またはレンジ 2 であることを確認します（PWR_VOSR の VOS[1:0]を使用）。
   b) EPODのブースタクロックが有効であることを確認します（RCC_PLL1CFGRの PLL1MBOOST[3:0]を使用）。

c) PWR_VOSRのUSBPWRENおよびUSBBOOSTENをセットして、USB内部電力を有効にします。

d) PWR_VOSRのUSBBOOSTRDYがセットされるまで待ちます。

次のシーケンスをPG[15:2]からI/Oを使用する前に実行する必要があります。

1. V_{DDIO2}がV_{DD}から独立している場合：
   a) PWR_SVMCRのIO2VMをセットすることによって、IO2VMを有効にします。
   b) IO2CVMウェイクアップ時間を待ちます。
   c) PWR_SVMSRのVDDIO2RDYがセットされるまで待ちます。
   d) 消費電力削減のためにIO2VMを無効にします（オプション）。
2. PWR_SVMCRのIO2SVをセットして、V_{DDIO2}の電源アイソレーションを解除します。

これらのアナログペリフェラル（A/Dコンバータ、D/Aコンバータ、コンパレータ、オペアンプ、電圧基準バッファ）のいずれかを使用する前に次のシーケンスを実行する必要があります。

1. V_{DDA}がV_{DD}から独立している場合：
   a) PWR_SVMCRのAVM1ENまたはAVM2ENをセットして、AVM1またはAVM2を有効にします。
   b) AVMウェイクアップ時間を待ちます。
   c) PWR_SVMCRのVDDA1RDYまたはVDDA2RDYがセットされるまで待ちます。
   d) 消費電力削減のためにAVMを無効にします（オプション）。
2. PWR_SVMCRのASVをセットして、V_{DDA}の電源アイソレーションを解除します。

10.6.4 バックアップドメイン電圧および温度監視

バックアップドメイン電圧および温度監視が有効な場合（PWR_DBPRのMONEN = 1）、バックアップドメイン電圧と温度が監視されます。

バックアップドメイン電圧監視内部タンパがTAMPペリフェラルで有効な場合（TAMP_CR1のITAMP1E = 1）、バックアップドメイン電圧が機能範囲を上回ると、タンパイベントが生成されます。バックアップドメイン電圧が機能範囲を下回ると、ブラウンアウトリセットが生成され、バックアップドメイン電を含むすべてのデバイスが消去されます。

注： **バックアップドメイン電圧は、存在するときにはV_{DD}であり、そうでない場合はV_{BAT}です。**

温度監視内部タンパがTAMPペリフェラルで有効な場合（TAMP_CR1のITAMP2E = 1）、温度が機能範囲を上回ったり下回ったりすると、タンパイベントが生成されます。

10.7 PWR電源管理

10.7.1 電源モード

デフォルトでは、マイクロコントローラは、システムまたは電源のリセット後はRUNモードです。RUNモードで消費電力を低減するには、アプリケーションパフォーマンスのニーズに応じて、電圧スケーリングを設定します。詳細については、セクション10.5.4を参照してください。未使用のRAMは、PWR_CR1のSRAMxPDビットで電源をオフにできます。消費電力は、SYSCLK、HCLK、およびPCLKのクロック速度、または未使用ペリフェラルクロックのゲーティングでも低減されます。詳細については、セクション 11：リセットおよびクロック制御（RCC）を参照してください。

外部イベント待ちなど、CPUの連続実行が不要なときの節電のために、いくつかの低電力モードが用意されています。消費電力の節減、スタートアップ時間の短縮、使用可能なウェイクアップソースを考慮した最適なモード選択はユーザに委ねられています。

このデバイスは、次の低電力モードを備えています。

- SLEEPモード：

  CPUクロックオフ。NVIC、SysTickなどのCortex-M33コアを含むすべてのペリフェラルが実行でき、割込みやイベントが発生したときにCPUをウェイクアップできます。セクション10.7.5を参照してください。

- STOP0、STOP1、STOP2、およびSTOP3モード：

  ストップモードはSRAMとレジスタの内容を保持したまま電力消費を最小限に抑えることができます。コアドメインのすべてのクロックが停止します。PLL、MSI（MSISおよびMSIK）RC、HSI16RC、およびHSE水晶発振器が無効になります。LSEやLSIは動作を継続します。

  RTCをアクティブなままにできます（RTCありのSTOPモード、RTCなしのSTOPモード）。

  一部のペリフェラルは自律しており、DMA（ペリフェラルや電力モードに応じてGPDMA1またはLPDMA1）でデータを転送するために、必要に応じてカーネルクロックとバス（APBまたはAHB）をリクエストすることで、STOPモードで動作できます。

  STOP2およびSTOP3モードでは、ほとんどのコアドメインは低リーク電流モードに置かれます。STOP0およびSTOP1は、アクティブなペリフェラルおよびウェイクアップソースの最大数と、短いウェイクアップ時間を提供しますが、STOP2より消費電力は大きくなります。

  STOP0モードでは、レギュレータはメインレギュレータモードのままです。これにより、ウェイクアップ時間は非常に速くなりますが、消費電力はさらに大きくなります。

  STOP3はSRAMとレジスタの内容が完全に保持された一番低い電力モードですが、機能しているウェイクアップのペリフェラルおよびソースは、STANDBYモード以外と同様に低減されます。

  システムクロックは、STOPモードを終了する際に、ソフトウェア設定に応じて最大24MHzのMSISまたはHSI16のいずれかにすることができます。

  セクション10.7.6、セクション10.7.7、セクション10.7.8、およびセクション10.7.9を参照してください。

- STANDBYモード：

  STANDBYモードでは、BORで電力消費を最小限に抑えることができます。内部レギュレータがオフになるため、コアドメインはパワーオフになります。PLL、MSI（MSISおよびMSIK）RC、HSI16RC、およびHSE水晶発振器もオフになります。

  RTCをアクティブなままにできます（RTCありのSTANDBYモード、RTCなしのSTANDBYモード）。

  ブラウンアウトリセット（BOR）は、STANDBYモードでは常にアクティブです。

  STANDBYモード中の各I/Oの状態は、ソフトウェアによって内部プルアップ、内部プルダウン、またはフローティングから選択できます。

  STANDBYモードへの移行後、バックアップドメインおよびSTANDBY回路のレジスタとバックアップSRAMを除き、SRAMとレジスタの内容は失われます。オプションで、低電力レギュレー

タによって電源供給されたSRAM2全体または8KBまたは56KBをSTANDBYモードで保持できます（SRAM2保持モードでのSTANDBY）。

BORは、STANDBYモード中および最低閾値選択時の消費電流をさらに低減するために、超低電力モードで設定できます（V_{BOR0}）。

デバイスは、外部リセット（NRSTピン）、IWDGリセット、WKUPピンイベント（設定可能な立ち上がりまたは立ち下がりエッジ）、RTCイベントの発生（アラーム、周期的ウェイクアップ、タイムスタンプ）、またはタンパ検出時に、STANDBYモードを終了します。タンパ検出は、外部ピンによって、または内部エラー検出によってレイズされます。

ウェイクアップ後のシステムクロックは、最大4MHzのMSISです。

セクション10.7.10を参照してください。

- SHUTDOWNモード：

  SHUTDOWNモードでは、消費電力を最小限に抑えることができます。内部レギュレータがオフになるため、コアドメインはパワーオフになります。PLL、HSI16、MSI（MSISおよびMSIK）、LSI、およびHSEオシレータもオフになります。

  RTCをアクティブなままにできます（RTCありのSHUTDOWNモード、RTCなしのSHUTDOWNモード）。
  BORはSHUTDOWNモードで使用できません。このモードで電源電圧監視はできないため、バックアップドメインへの切替えはサポートされません。

  バックアップドメインのレジスタを除いて、SRAMとレジスタの内容は失われます。

  デバイスは、外部リセット（NRSTピン）、WKUPピンイベント（設定可能な立ち上がりまたは立ち下がりエッジ）、RTCイベントの発生（アラーム、周期的ウェイクアップ、タイムスタンプ）、またはタンパ検出時に、SHUTDOWNモードを終了します。

  ウェイクアップ後のシステムクロックは、4MHzのMSISです。セクション10.7.11を参照してください。

  次の表に、電力モードの概要を示します。

表99.低消費電力モードの概要

モード名	エントリ	ウェイクアップソース(1)	ウェイクアップシステムクロック	クロックへの影響	電圧レギュレータ
SLEEP（Sleep-nowまたはSleep-on-exit）	WFIまたはISRからのリターン	レンジ4のOTG_FS、USB、およびUCPDを除いた任意の割込み – レンジ3および4のOTG_HS	SLEEPモードへの移行前と同様	CPUクロックオフ 他のクロックまたはアナログクロックソースには影響なし	範囲1、2、3、4
	WFE	ウェイクアップイベント			

表99.低消費電力モードの概要 （続き）

モード名	エントリ	ウェイクアップソース(1)	ウェイクアップシステムクロック	クロックへの影響	電圧レギュレータ
STOP 0	LPMS = 000 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE	任意のEXTIライン（EXTIレジスタで設定） 特定のペリフェラルイベント/割込み(2)	HSI16（RCC_CFGR1のSTOPWUCK = 1のとき） STOPモードに移行する前の24MHzに制限された周波数でのMSIS（STOPWUCK = 0のとき）	LSIおよびLSEを除くすべてのクロックオフ MSIK、MSIS、またはHSI16は、自律型ペリフェラルによってリクエストされた場合、または有効な状態で保持するよう強制されている場合は、一時的に有効にできます。	範囲1、2、3、4
STOP 1	LPMS = 001 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE				低電力レギュレータ（SMPSまたはLDO）
STOP 2	LPMS = 010 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE				
STOP 3	LPMS = 011 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE	WKUPピンのエッジ、RTC/TAMPイベント/割込み (2)、NRSTピンでの外部リセット、IWDGリセット		LSIおよびLSEを除くすべてのクロックオフ	
SRAM2_8KBでのSTANDBY	LPMS = 10x+ RRS1 = 1 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE		MSIS1 ~ 4MHzからのMSIS		
SRAM2_Fullでの STANDBY	LPMS = 10x+ RRS1 = RRS2 = 1+ SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE				
STANDBY	LPMS = 10x + RRS1 = RRS2 = 0 + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE				オフ
SHUTDOWN	LPMS = 11x + SLEEPDEEPビット + WFIまたはISRからのリターンまたはWFE	WKUPピンのエッジ、RTC/TAMPイベント/割込み (2)、NRSTピンでの外部リセット	MSIS 4MHz	LSEを除くすべてのクロックオフ	オフ

1. 表100を参照してください。
2. ウェイクアップイベントは、ペリフェラル割込み信号で生成できます。セクション ：低消費電力モードの終了を参照してください。

表100.動作モードに応じた機能(1)

ペリフェラル	RUN	SLEEP	STOP 0/1		STOP 2		STOP 3		STANDBY		SHUTDOW		VBAT
			-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
CPU	Y	-	-	-	-	-	-	-	-	-	-	-	-
Flashメモリ	O(2)	O (2)	-	- (5)	-	-	-	-	-	-	-	-	-
SRAM1	Y(3)	Y(4)	O (8)	-	O (8)	-	O (8)	-	-	-	-	-	-
SRAM2	Y (3)	Y (4)	O (8)	O(5)	O (8)	-	O (8)	-	O(6)	-	-	-	-
SRAM3(7)	Y (3)	Y (4)	O(8)	O (5)	O (8)	-	O (8)	-	-	-	-	-	-
SRAM4	Y (3)	Y (4)	O (8)	-	O (8)	-	O (8)	-	-	-	-	-	-
SRAM5 (7)	Y (3)	Y (4)	O (8)	-	O (8)	-	O (8)	-	-	-	-	-	-
SRAM6 (7)	Y (3)	Y (4)	O (8)	-	O (8)	-	O (8)	-	-	-	-	-	-
BKPSRAM	O	O	O	O (5)	O		O		O				O
FSMC	O	O	-	-	-	-	-	-	-	-	-	-	-
OCTOSPIx (7) (x = 1、2)	O	O	-	-	-	-	-	-	-	-	-	-	-
HSPI1 (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
バックアップレジスタ	Y	Y	Y	-	Y	-	Y	-	Y	-	Y	-	Y
ブラウンアウトリセット（BOR）	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	-	-	-
プログラム可能な電圧検出器（PVD）	O	O	O	O	O	O	-	-	-	-	-	-	-
ペリフェラル電圧モニタ	O	O	O	O	O	O	-	-	-	-	-	-	-
GPDMA	O	O	O	O(9)	-	-	-	-	-	-	-	-	-
LPDMA	O	O	O	O(10)	O	O (10)	-	-	-	-	-	-	-
DMA2D (7)	O	O											
ハイスピード内部（HSI16）	O	O	(11)	-	(11)	-	-	-	-	-	-	-	-
オシレータHSI48	O	O	-	-	-	-	-	-	-	-	-	-	-
高速外部（HSE）	O	O	-	-	-	-	-	-	-	-	-	-	-
低速内部（LSI）	O	O	O	-	O	-	O	-	O	-	-	-	O

表100.動作モードに応じた機能(1)（続き）

ペリフェラル	RUN	SLEEP	STOP 0/1		STOP 2		STOP 3		STANDBY		SHUTDOW		VBAT
			-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
低速外部（LSE）	O	O	O	-	O	-	O	-	O	-	O	-	O
マルチスピード内部（MSISおよびMSIK）	O	O	(11)	-	(11)	-	-	-	-	-	-	-	-
クロックセキュリティシステム（CSS）	O	O	-	-	-	-	-	-	-	-	-	-	-
LSE のクロックセキュリティシステム	O	O	O	O	O	O	O	O	O	O	O	O	O
バックアップドメイン電圧監視、温度監視	O	O	O	O	O	O	O	O	O	O	O	O	O
RTC/TAMP	O	O	O	O	O	O	O	O	O	O	O	O	O
TAMPタンパピンの数	8	8	8	O	8	O	8	O	8	O	8	O	8
OTG_FS (7)、OTG_HS (7)、USB (7)、UCPD (7)	O(12)	O (12)	-	O	-	-	-	-	-	-	-	-	-
USARTx（x = 1、2 (7)、3、4、5、6 (7)）	O	O	O(13)	O (13)	-	-	-	-	-	-	-	-	-
低電力UART（LPUART）	O	O	O (13)	O (13)	O (13)	O (13)	-	-	-	-	-	-	-
I2Cx（x = 1、2、4、5 (7)、6 (7)）	O	O	O(14)	O (14)	-	-	-	-	-	-	-	-	-
I2C3	O	O	O (14)	O (14)	O (14)	O (14)	-	-	-	-	-	-	-
SPIx（x = 1、2）	O	O	O(15)	O (15)	-	-	-	-	-	-	-	-	-
SPI3	O	O	O (15)	O (15)	O (15)	O (15)							
FDCAN1	O	O	-	-	-	-	-	-	-	-	-	-	-
SDMMCx（x = 1、2 (7)）	O	O	-	-	-	-	-	-	-	-	-	-	-
SAIx（x = 1、2 (7)）	O	O	-	-	-	-	-	-	-	-	-	-	-
ADCx（x = 1、2 (7)）	O	O	-	-	-	-	-	-	-	-	-	-	-
ADC4	O	O	O(16)	O (16)	O (16)	O (16)	-	-	-	-	-	-	-
DAC1（2つのコンバータ）	O	O	O	-	O	-	-	-	-	-	-	-	-
VREFBUF	O	O	O	-	O	-	-	-	-	-	-	-	-
OPAMPx（x = 1、2）	O	O	O	-	O	-	-	-	-	-	-	-	-

表100.動作モードに応じた機能(1)（続き）

ペリフェラル	RUN	SLEEP	STOP 0/1		STOP 2		STOP 3		STANDBY		SHUTDOW		VBAT
			-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
COMPx（x = 1、2）	O	O	O	O	O	O	-	-	-	-	-	-	-
温度センサ	O	O	O	-	O	-	-	-	-	-	-	-	-
タイマ（TIMx）	O	O	-	-	-	-	-	-	-	-	-	-	-
GFXTIM (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
LPTIMx（x = 1、3、4）	O	O	O(17)	O (17)	O (17)	O (17)	-	-	-	-	-	-	-
LPTIM2	O	O	O (17)	O (17)	-	-	-	-	-	-	-	-	-
独立型ウォッチドッグ（IWDG）	O	O	O	O	O	O	O	O	O	O	-	-	-
ウィンドウ型ウォッチドッグ（WWDG）	O	O	-	-	-	-	-	-	-	-	-	-	-
SysTickタイマ	O	O	-	-	-	-	-	-	-	-	-	-	-
多機能デジタルフィルタ（MDF）	O	O	O(18)	O (18)	-	-	-	-	-	-	-	-	-
オーディオ･デジタルフィルタ（ADF）	O	O	O (18)	O (18)	O (18)	O (18)	-	-	-	-	-	-	-
LTDC (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
DSI (7)	O(19)	O (19)	-	-	-	-	-	-	-	-	-	-	-
GFXMMU (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
GPU2D (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
JPEG (7)	O	O	-	-	-	-	-	-	-	-	-	-	-
デジタルカメラインタフェース（DCMI）	O	O	-	-	-	-	-	-	-	-	-	-	-
パラレル同期スレーブインタフェース（PSSI）	O	O	-	-	-	-	-	-	-	-	-	-	-
CORDICコプロセッサ（CORDIC）	O	O	-	-	-	-	-	-	-	-	-	-	-
フィルタ演算アクセラレータ（FMAC）	O	O	-	-	-	-	-	-	-	-	-	-	-
タッチセンシングコントローラ（TSC）	O	O	-	-	-	-	-	-	-	-	-	-	-
乱数発生器（RNG）	O	O	-	-	-	-	-	-	-	-	-	-	-

表100.動作モードに応じた機能(1)（続き）

ペリフェラル	RUN	SLEEP	STOP 0/1		STOP 2		STOP 3		STANDBY		SHUTDOW		VBAT
			-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
AESおよびセキュアAES（7）	O	O	-	-	-	-	-	-	-	-	-	-	-
公開鍵アクセラレータ（PKA）（7）	O	O	-	-	-	-	-	-	-	-	-	-	-
オンザフライ復号（OTFDEC）（7）	O	O	-	-	-	-	-	-	-	-	-	-	-
HASHアクセラレータ	O	O	-	-	-	-	-	-	-	-	-	-	-
CRC計算ユニット	O	O	-	-	-	-	-	-	-	-	-	-	-
GPIO	O	O	O	O	O	O	(20)	ピン：24本	(20)	ピン：24本	(21)	ピン：24本	-

1. Y = 使用可（有効）O = オプション（デフォルトでは無効、ソフトウェアによって有効にできる）- = 使用不可
各モードのウェイクアップ機能のセルは灰色でハイライトされています。
2. Flashメモリはパワーダウンモードに設定できます。デフォルトは、パワーダウンモードではありません。
3. SRAMは、個別にパワーオンまたはパワーオフできます。
4. SRAMクロックは、個別にゲートオンまたはゲートオフできます。
5. ECCエラー割込みまたはNMIはこのSTOPモードからウェイクアップします。
6. 8KB、56KB、またはSRAM2全体の内容を保存できます。
7. この機能は一部の STM32U5 シリーズデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。
8. SRAM1、SRAM3、SRAM5、SRAM6のサブブロックまたはブロック全体、SRAM2およびSRAM4のブロック全体は、消費電力を抑えるために電源をオフにできます。SRAM1、SRAM2、SRAM3、SRAM4、SRAM5、およびSRAM6は、STOP0およびSTOP1モードのGPDMAでアクセスできます。SRAM4は、STOP0、STOP1、およびSTOP2モードのLPDMAでアクセスできます。
9. GPDMA転送はSTOPモードでは有効かつ自律的で、転送イベントではウェイクアップ割込みを生成します。
10. LPDMA転送はSTOPモードでは有効かつ自律的で、転送イベントではウェイクアップ割込みを生成します。
11. 自律モードでSTOP機能からウェイクアップする一部のペリフェラルでは、HSI16、MSIS、またはMSIKを有効にすることをリクエストできます。この場合、オシレータはペリフェラルによってウェイクアップします。どのペリフェラルもそれを必要としない場合は、自動的にオフになります。
12. OTG_FSとUSBは、電圧スケーリングレンジ1、2、および3で機能します。OTG_HSは、電圧スケーリングレンジ1および2で機能します。
13. USARTおよびLPUARTの受信および送信は、STOPモード、非同期モード、およびSPIマスタモードでは有効かつ自律的で、転送イベントではウェイクアップ割込みを生成します。
14. I2Cの受信および送信は、STOPモードでは有効かつ自律的で、転送イベントではウェイクアップ割込みを生成します。
15. SPIの受信および送信は、STOPモードでは有効かつ自律的で、転送イベントではウェイクアップ割込みを生成します。
16. ADC変換はSTOPモードでは有効かつ自律的で、変換イベントではウェイクアップ割込みを生成します。
17. LPTIMはSTOPモードでは有効かつ自律的で、イベントではウェイクアップ割込みを生成します。
18. MDFおよびADFはSTOPモードでは有効かつ自律的で、イベントではウェイクアップ割込みを生成します。
19. DSIは、電圧スケーリングレンジ1および2で機能します。
20. I/Oは、STANDBYモードで内部プルアップ、プルダウンまたはフローティングで設定できます。
21. I/Oは、SHUTDOWNモードで内部プルアップ、プルダウンまたはフローティングで設定できますが、SHUTDOWNモードを終了すると設定が失われます。

さらに、次の手段のいずれかによってRUNモードの消費電力を節減できます。

- システムクロックの低速化と電圧スケーリングの低電力レンジへの設定
- 使用しないAPBおよびAHBペリフェラルへのクロックのゲーティング
- 未使用RAMの電源オフ

SRAMの電源がオフになったら、次の手順で再度電源をオンにできます。

1. PWR_CR1のSRAMxPDをリセットします。
2. 1.6µs待ちます。
3. RCC_AHBxENRのSRAMxENをセットします。

デバッグモード

デフォルトでは、デバッグ機能が使用されているときにアプリケーションがMCUをSTOP0、STOP1、STOP2、STOP3、STANDBY、またはSHUTDOWNモードにすると、デバッグ接続は失われます。これは、Cortex-M33がクロック供給されなくなるためです。

ただし、DBGMCU制御レジスタのいくつかの設定ビットをセットすることで、低消費電力モードを多用しているときでも、ソフトウェアのデバッグを行うことができます。詳細については、セクション 75.2.5：デバッグおよび低電力モードを参照してください。

10.7.2 自律型ペリフェラルと低消費電力バックグラウンド自律モード（LPBAM）

いくつかのペリフェラルでは自律モードをサポートしており、これによって機能的にSTOP0、STOP1、およびSTOP2モードのDMA転送を実行できます。加えて、低消費電力バックグラウンド自律モード（LPBAM）がSTOP2モードでサポートされており、これによって自律型ペリフェラルを用いたより複雑なユースケースを、DMA転送によってCPUウェイクアップなしで構築できます。

STOP0およびSTOP1モード

STOP0およびSTOP1モードでは、自律型ペリフェラルはADC4、DAC1、LPTIMx（x = 1 ~ 4）、USARTx（x = 1 ~ 6）、LPUART1、SPIx（x = 1 ~ 3）、I2Cx（x = 1 ~ 6）、MDF1、ADF1、GPDMA1、およびLPDMA1です。

- ADC4、DAC1、LPTIM1、LPTIM3、LPUART1、SPI3、I2C3、およびADF1は、LPDMA1およびSRAM4でのみ自律しています。
- LPTIM2、USARTx（x = 1 ~ 6）、SPI1、SPI2、I2C1、I2C2、I2C4、およびMDF1は、GPDMA1およびSRAM1からSRAM5でのみ自律しています。

STOP2モード

STOP2モードで、自律型ペリフェラルはADC4、DAC1、LPTIM1、LPTIM3、LPUART1、SPI3、I2C3、ADF1、およびLPDMA1です。このモードで、SRAM4はLPDMA1でアクセスできます。

自律型ペリフェラルおよびLPBAM機能

これらの自律型ペリフェラルは、次の機能をサポートしています。

- STOPモードの機能は、独自の独立したクロック（カーネルクロックと呼ばれる）のリクエスト機能によるものです。ペリフェラルカーネルクロックは、ペリフェラルからのリクエストがあれば自動的にオンになり、ペリフェラルからのリクエストがなければ自動的にオフになります。
- DMA転送は、システムクロックのリクエスト機能によりSTOPモードでサポートされています。システムクロック（MSISまたはHSI16）は、ペリフェラルからのリクエストがあれば自動的にオンになり、ペリフェラルからのリクエストがなければ自動的にオフになります。システムクロックが自律型ペリフェラルからリクエストされた場合、システムクロックはウェイクアップし、RCCで有効化されたすべてのペリフェラルに分配されます。これにより、DMAは有効化さ

れたSRAMと、任意の有効化されたペリフェラルレジスタ（たとえばGPIOまたはLPGPIOレジスタ）にアクセスできます。

- ハードウェアの同期または非同期トリガによるペリフェラルの自動起動（I/O のエッジ検出や低電力タイマイベントなど）
- ペリフェラル割込みによるSTOPモードからのウェイクアップ

GPDMA1およびLPDMA1はSTOPモードで完全に機能しており、リンクリストも更新されています。これにより、CPUウェイクアップなしで、さまざまなDMA転送をリンクできます。これは、さまざまなペリフェラルの転送をつなげたり、ペリフェラルのレジスタを書き込んでSTOPモードのままで設定を変更したりするために使用できます。LPDMA1リンクリスト転送によってSTOP2モードでこれらのペリフェラルのシナリオを作成するために、LPBAMアプリケーションドライバおよびツールはSTM32CubeMXで提供されます。

DMAのメモリ間転送は、ハードウェアの同期または非同期トリガによって起動できます。DMAのペリフェラルとメモリの間の転送も、これらのトリガによってゲートされます。

以下は、STOPモードのままで実行可能なユースケースです。

- 低電力タイマ（またはその他のトリガ）によってトリガされたADCまたはDAC変換
  - ADC変換結果がプログラムされた閾値から外れている場合のアナログウォッチドッグでのSTOPモードからのウェイクアップ
  - DMAバッファイベントでのSTOPモードからのウェイクアップ
- SRAMへのオーディオ·デジタル·フィルタのデータ転送
  - 音声アクティビティ検出でのSTOPからのウェイクアップ
- I^2Cスレーブ受信または送信、SPI受信、UART/LPUART受信
  - ペリフェラル転送の最後またはDMAバッファイベントでのウェイクアップ
- 低電力タイマ（またはその他のトリガ）によってトリガされたI^2Cマスタ転送、SPI送信、UART/LPUART送信
  - 例：周期的なセンサの読出し
  - ペリフェラル転送の最後またはDMAバッファイベントでのウェイクアップ
- ペリフェラル間のブリッジ
  - 例：通信ペリフェラルによって転送されたADC変換データ
- 次の目的で行われるGPIO/LPGPIOとSRAMの間のデータ転送：
  - 外部コンポーネントの制御
  - データ送受信プロトコルの実装
- SRAMから別のSRAMへのデータ転送

10.7.3 RUNモード

システムクロックの低速化

RUNモードでは、プリスケーラレジスタをプログラミングすることによって、システムクロック（SYSCLK、HCLK、PCLK）の速度を下げることができます。SLEEPモードに移行する前にペリフェラルの速度を下げるため、これらのプリスケーラを使用することもできます。

詳細については、セクション 11：リセットおよびクロック制御（RCC）を参照してください。

ペリフェラルクロックゲーティング

RUNモードでは、消費電力を低減するため、個々のペリフェラルとメモリへのHCLKおよびPCLKをいつでも停止することができます。

SLEEPモードで消費電力をさらに低減するため、WFIまたはWFE命令を実行する前に、ペリフェラルクロックを停止することができます。

ペリフェラルクロックゲーティングは、RCC_AHBxENRおよびRCC_APBxENRレジスタで制御されます。

RCC_AHBxSMENRレジスタとRCC_APBxSMENRレジスタの対応ビットをリセットすることで、SLEEPモード時のペリフェラルクロックを自動的に停止させることができます。このビットは、自律的なDMA転送のため、またはウェイクアップ割込みを生成するために、STOPモードでクロックをリクエストするペリフェラルのためにセットする必要があります。

RCC_AHB3AMENRとRCC_APB3AMENRの対応ビットをリセットすることで、STOP2モードのペリフェラルの自律的なクロックを自動的に停止できます。

10.7.4 低電力モード

低消費電力モードへの移行

MCUは、WFI（Wait For Interrupt）またはWFE（Wait For Event）命令の実行時、またはCortex-M33システム制御レジスタのSLEEPONEXITビットが割込みサービスルーチン（ISR）からの復帰でセットされたときに、低電力モードに移行します。

WFIまたはWFEによる低消費電力モードへの移行は、ペンディング状態の割込みやイベントがない場合にのみ実行されます。

注意： **自律型モードのペリフェラルは、内部イベントに応じてAHBまたはAPBのクロックリクエストを生成できます。WFIまたはWFEの実行時にクロックリクエストがある場合は、そのクロックリクエストがリリースされてから低電力モードに移行します。**

低消費電力モードの終了

MCUがSLEEPまたはSTOPモードを終了する方法は、低電力モードへの移行方法によって異なります。

- WFI命令または割込みサービスルーチン（ISR）終了によって低電力モードに移行した場合、NVICによって認識されたペリフェラル割込みであればどれでもデバイスをウェイクアップすることができます。
- WFE命令によって低電力モードに移行した場合、MCUはイベントの発生直後に低電力モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。
  - NVICIRQ/ペリフェラル割込み：

    - Cortex-M33システム制御レジスタのSEVONPEND = 0のとき

    ペリフェラル制御レジスタおよびNVICにて割込みを有効にすることで行います。MCUがWFEからリスタートするときには、ペリフェラル割込みペンディングビットとNVICペリ

フェラルIRQチャネルペンディングビット（NVIC割込みクリアペンディングレジスタ）をクリアする必要があります。十分高い優先度のNVIC割込みのみがMCUをウェイクアップし、割込みを生成します。

- Cortex-M33システム制御レジスタのSEVONPEND = 1のとき

ペリフェラル制御レジスタおよびNVIC（オプション）にて割込みを有効にすることで行います。MCUがWFEからリスタートするときには、ペリフェラル割込みペンディングビットとNVICペリフェラルIRQチャネルペンディングビット（NVIC割込みクリアペンディングレジスタ）をクリアする必要があります。すべてのNVIC割込み（無効化されているものも含む）によってMCUをウェイクアップします。十分高い優先度の有効化されたNVIC割込みのみがMCUをウェイクアップし、割込みを生成します。

– イベント：

EXTIラインをイベントモードに設定します。CPUがWFEからリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、EXITペリフェラル割込みペンディングビットやNVIC IRQチャネルペンディングビットをクリアする必要はありません。ペリフェラルの割込みフラグをクリアする必要があるかもしれません。

MCUは、外部リセット（NRSTピン）、IWDGリセット、有効なWKUPxピンのうちの1つの立ち上がりエッジ、またはRTC/TAMPイベントが発生すると、STOP3、STANDBY、またはSHUTDOWNモードを終了します（図 766：RTCのブロック図を参照）。

STANDBYまたはSHUTDOWNモードからのウェイクアップ後、プログラム実行はリセット後と同様にリスタートされます（ブートピン信号のサンプリング、オプションバイトローディング、リセットベクタのフェッチなど）。

注意： **デバイスがSTOPモードのとき、ペリフェラル割込みにより、内部オシレータがパワーオンされます。割込みによってデバイスがSTOPモードから復帰できるように、対応するNVIC割込みチャネルを有効にする必要があります。ペリフェラル割込みを有効にしたままNVICチャネルのみを無効にして、ペリフェラル割込みを無効にすることはできません。これは、クロックがオンの状態で、デバイスがSTOPモードを維持することがあるためです。**
自律型モードのペリフェラルは、内部イベントに応じて、デバイスがSTOPモードのときに、AHBまたはAPBのクロックリクエストを生成できます。ソフトウェアは、RCC、自律型ペリフェラル、DMAチャネル、およびNVICを適切かつ一貫した方法で設定することで、DMA転送または割込みのどちらかが処理されていることを確認する必要があります。自律型ペリフェラルがSTOPモードでバスクロックをリクエストするときに、AHBおよびAPBクロックがすべての有効化されたペリフェラルに分配される点に注意してください（STOP2モードのSmartRunドメインのペリフェラルのみ）。結果として、有効化されたペリフェラルは、自律モードの機能がなくても一時的にクロック供給され、この間に割込みを生成することもできます。,これらのペリフェラル割込みは、デバイスをSTOPモードからウェイクアップします。

10.7.5 SLEEPモード

SLEEPモードにおけるI/Oの状態

SLEEPモードで、すべてのI/OピンはRUNモードと同じ状態を保持します。

SLEEPモードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがクリアされたとき、低消費電力モードへの移行で説明されているように、SLEEPモードに移行します（SLEEPモードへの移行方法の詳細については、表101を参照）。

SLEEPモードの終了

MCUは、低消費電力モードの終了で説明されているようにSLEEPモードを終了します（SLEEPモードの終了方法の詳細については、表101を参照）。

表101.SLEEPモード

SLEEPモード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – SLEEPDEEP = 0 – ペンディング中の割込み（WFI）やイベント（WFE）なし Cortex-M33システム制御レジスタを参照。
	次の条件下での割込みサービスルーチン（ISR）終了時 – SLEEPDEEP = 0および – SLEEPONEXIT = 1 – ペンディング中の割込みなし Cortex-M33システム制御レジスタを参照。
モードの終了	WFIまたはISRからの復帰を使用して移行した場合： 割込み（表 184：STM32U5シリーズベクタテーブルを参照） WFE命令を使用して移行し、SEVONPEND = 0の場合： ウェイクアップイベント（セクション 23.3：EXTIの機能説明を参照） WFE命令を使用して移行し、SEVONPEND = 1の場合： NVIC（表 184：STM32U5シリーズベクタテーブルを参照）、またはウェイクアップイベント（セクション 23.3：EXTIの機能説明を参照）で無効化されている場合の割込みイベント
ウェイクアップ遅延時間	なし

10.7.6 STOP0モード

STOP0モードは、ペリフェラルクロックゲーティングと組み合わされたCortex-M33のディープスリープ（deepsleep）モードに基づきます。電圧レギュレータは、メインレギュレータモードで設定されます。STOP0モードでは、コアドメインのすべてのクロックが停止されます。PLL、MSIS、MSIK、HSI16、およびHSEオシレータは無効になります。

自律モード機能を有する一部のペリフェラルは、データ転送のためHSI16またはMSISまたはMSIKをオンにできます（詳細については、セクション10.7.2を参照）。

すべてのSRAMとレジスタの内容は保持されますが、消費電力をさらに削減するために、SRAMは完全に、または部分的にオフになります。

BORは常にSTOP0モードで使用できます。

STOP0モードにおけるI/Oの状態

STOP0モードでは、すべてのI/OピンはRUNモードと同じ状態を保持します。

STOP0モードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがセットされたとき、低消費電力モードへの移行で説明されているように、STOP0モードに移行します（STOP0モードへの移行方法の詳細については、表102を参照）。

Flashメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP0モードに移行します。

APBドメインにアクセス中の場合、APBアクセスが終了してから、STOP0モードに移行します。

STOP0モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 61.4：IWDGの機能説明を参照）。
- RCC_BDCRのRTCENによって有効化されたリアルタイムクロック（RTC）とタンパ（TAMP）カーネルクロック。
- 128分周された内部RCオシレータのLSIクロックまたはLSIクロックは、RCC_BDCRのLSIONおよびLSIPREDIVビットによって設定されます。
- 外部32.768kHzオシレータ（LSE）は、RCC_BDCRのLSEONビットによって設定されます。

いくつかのペリフェラルは、STOP0モードで自律的である場合があり、有効化されている場合は消費電力を追加できます（詳細については、セクション10.7.2を参照）。

STOP0モードでは、OPAMP、COMP、PVM、およびPVDを使用できます。これらが必要ではない場合、消費電力を低減するためにソフトウェアで無効にする必要があります。

ADCx（x = 1、4）、DAC1（2チャネル）、温度センサ、およびVREFBUFは、STOP0モードに移行する前に無効にしない限り、STOP0モード中に電力を消費します。

STOP0モードの終了

MCUは、低消費電力モードの終了で説明されているようにSTOP0モードを終了します（STOP0モードの終了方法の詳細については、表102を参照）。

割込みまたはウェイクアップイベントの発行によってSTOP0モードを終了する場合、RCC_CFGR1のSTOPWUCKがセットされると、システムクロックとしてHSI16が選択されます。STOPWUCKがクリアされると、システムクロックとしてMSISオシレータが選択されます。MSISを選択すると、より高い周波数（最大24MHz）でウェイクアップできます。
STOPモードではいくつかのペリフェラルが自律的で、STOP機能からのウェイクアップで割込みを生成できます。STOP割込みからウェイクアップできるようにするには、すべてのペリフェラルクロックを有効にする必要があります（ペリフェラルクロックゲーティングを参照）。

STOP0モードを終了するとき、MCUはレンジ4のRUNモードになります。

表102.STOP0モード

STOP0モード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_CR1のLPMS = 000
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXIT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 000
	注： STOP0モードに移行するには、すべてのEXTIラインのペンディングビット（EXTI_RPR2）とウェイクアップ割込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP0モード移行手順が無視され、プログラムが実行され続けます。

表102.STOP0モード（続き）

STOP0モード	説明
モードの終了	WFIまたはReturn from ISRを使用して移行した場合 - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで有効になっている必要があります）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み（ペリフェラルベクタをNVICで有効化する必要がある） WFE命令を使用して移行し、SEVONPEND = 0の場合： - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照）。 WFE命令を使用して移行し、SEVONPEND = 1の場合： - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで無効になっている場合も含む）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照） - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み
	注： このペリフェラルがSTOP割込みからウェイクアップを生成するためには、すべてのペリフェラルクロックを有効にする必要があります（RCCに[PERIPH]EN、[PERIPH]SMEN、および[PERIPH]AMENビットをセットする必要があり、機能的な独立したクロックを選択する必要があります）。
ウェイクアップ遅延時間	最長のウェイクアップ時間：MSISまたはHSI16ウェイクアップ時間とSTOP0モードのFLASHウェイクアップ時間の間。

10.7.7 STOP1モード

STOP1モードは、レギュレータが低電力モードであることを除き、STOP0モードと同じです（STOP1モードへの移行および終了方法の詳細については、次の表を参照）。

表103.STOP1モード

STOP1モード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_CR1のLPMS = 001
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXIT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 001
	注： STOP1 モードに移行するには、すべての EXTI ラインのペンディングビット（EXTI_RPR1）とウェイクアップ割込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP1モード移行手順が無視され、プログラムが実行され続けます。

表103.STOP1モード（続き）

STOP1モード	説明
モードの終了	WFIまたはISRからの復帰を使用して移行した場合： - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで有効になっている必要があります）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み（ペリフェラルベクタをNVICで有効化する必要がある） WFE命令を使用して移行し、SEVONPEND = 0の場合： - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照） WFE命令を使用して移行し、SEVONPEND = 1の場合： - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで無効になっている場合も含む）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照） - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み
	注：　このペリフェラルがSTOP割込みからウェイクアップを生成するためには、すべてのペリフェラルクロックを有効にする必要があります（RCCに[PERIPH]EN、[PERIPH]SMEN、および[PERIPH]AMENビットをセットする必要があり、機能的な独立したクロックを選択する必要があります）。
ウェイクアップ遅延時間	最長のウェイクアップ時間：MSISまたはHSI16ウェイクアップ時間と、レギュレータの低電力モードからのウェイクアップ時間 + STOP1モードのFLASHウェイクアップ時間の間。

10.7.8 STOP2モード

STOP2モードは、ほとんどのコアドメインが低リーク電流モードに置かれていることを除き、STOP1と同じです。AHB3およびAPB3ペリフェラルを内蔵するコアドメインの一部のみが完全に電源供給されており、これらのペリフェラルはこれによって機能できます。

自律モード機能を有する一部のAHB3およびAPB3ペリフェラルは、データ転送のためHSI16またはMSISまたはMSIKをオンにできます（詳細については、セクション10.7.2を参照）。

すべてのSRAMとレジスタの内容は保持されますが、消費電力をさらに削減するために、SRAMは完全に、または部分的にオフになります。

BORは常にSTOP2モードで使用できます。

STOP2モードにおけるI/Oの状態

STOP2モードでは、すべてのI/OピンはRUNモードと同じ状態を保持します。

STOP2モードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがセットされたとき、低消費電力モードへの移行で説明されているようにSTOP2モードに移行します（STOP2モードへの移行方法の詳細については、表104を参照）。

Flashメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP2モードに移行します。

APBドメインにアクセス中の場合、APBアクセスが終了してから、STOP2モードに移行します。

STOP2モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 61.4：IWDGの機能説明を参照）。
- RCC_BDCR の RTCEN ビットによって有効化されたリアルタイムクロック（RTC）とタンパ（TAMP）カーネルクロック。
- 128分周された内部RCオシレータのLSIクロックまたはLSIクロックは、RCC_BDCRのLSIONおよびLSIPREDIVビットによって設定されます。
- 外部32.768kHzオシレータ（LSE）は、RCC_BDCRのLSEONによって設定されます。

いくつかのペリフェラルは、STOP2モードで自律的である場合があり、有効化されている場合は消費電力を追加できます（詳細については、セクション10.7.2を参照）。

STOP2モードでは、OPAMP、COMP、PVM、およびPVDを使用できます。これらが必要ではない場合、消費電力を低減するためにソフトウェアで無効にする必要があります。

ADCx（x = 1、2、4）、DAC1（2チャネル）、温度センサ、およびVREFBUFは、STOP2モードに移行する前に無効にしない限り、STOP2モード中に電力を消費します。

注意： **STOP2モードで有効にできないすべてのペリフェラルは、ペリフェラル自体の有効ビットをクリアして無効にするか、RCCレジスタを設定してリセット状態にする必要があります。**

STOP2モードの終了

MCUは、低消費電力モードの終了で定義されているようにSTOP2モードを終了します（STOP2モードの終了方法の詳細については、表104を参照）。

割込みまたはウェイクアップイベントの発行によってSTOP2モードを終了する場合、RCC_CFGR1のSTOPWUCKビットがセットされると、システムクロックとしてHSI16が選択されます。STOPWUCKがクリアされた場合、システムクロックとしてMSISが選択されます。MSIを選択すると、より高い周波数（最大24MHz）でウェイクアップできます。

STOPモードではいくつかのペリフェラルが自律的で、STOP機能からのウェイクアップで割込みを生成できます。STOP割込みからウェイクアップできるようにするには、すべてのペリフェラルクロックを有効にする必要があります（ペリフェラルクロックゲーティングを参照）。

STOP2モードを終了するとき、MCUはレンジ4のRUNモードになります。

表104.STOP2モード

STOP2モード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_CR1のLPMS = 010
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXIT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 010
	注： **STOP2モードに移行するには、すべてのEXTIラインのペンディングビット（EXTI_RPR2）とウェイクアップ割込みを生成するペリフェラルフラグをクリアする必要があります。そうでない場合、STOPモード移行手順が無視され、プログラムが実行され続けます。**

表104.STOP2モード

STOP2モード	説明
モードの終了	WFIまたはReturn from ISRを使用して移行した場合 - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで有効になっている必要があります）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み（ペリフェラルベクタをNVICで有効化する必要がある） WFE命令を使用して移行し、SEVONPEND = 0の場合： - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照）。 WFE命令を使用して移行し、SEVONPEND = 1の場合： - 割込みモードに設定されている任意のEXTIライン（対応するEXTI割込みベクタがNVICで無効になっている場合も含む）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルです（表 184：STM32U5シリーズベクタテーブルを参照）。 - イベントモードに設定されている任意のEXTIライン（セクション 23.3：EXTIの機能説明を参照）。 - 自律型ペリフェラルのクロックリクエストにより、AHB/APBクロックが存在するときに発生するRTC、TAMP、IWDG割込み、またはその他のペリフェラル割込み。
	注： このペリフェラルがSTOP割込みからウェイクアップを生成するためには、すべてのペリフェラルクロックを有効にする必要があります（RCCに[PERIPH]EN、[PERIPH]SMEN、および[PERIPH]AMENビットをセットする必要があり、機能的な独立したクロックを選択する必要があります）。
ウェイクアップ遅延時間	最長のウェイクアップ時間：MSISまたはHSI16ウェイクアップ時間と、レギュレータの低電力モードからのウェイクアップ時間 + STOP2モードのFLASHウェイクアップ時間の間。

10.7.9 STOP3モード

STOP3モードは、ペリフェラルクロックゲーティングと組み合わされたCortex-M33のディープスリープ（deepsleep）モードに基づきます。STOP3モードでは、コアドメインのすべてのクロックが停止されます。PLL、MSIS、MSIK、HSI16、およびHSEオシレータは無効になります。

すべてのSRAMとレジスタの内容は保持されますが、消費電力をさらに低減するために、SRAMは完全に、または部分的にオフになります。

BORは常にSTOP3モードで使用できます。

すべてのペリフェラルは、ペリフェラル自体の有効ビットをクリアして無効にするか、RCCレジスタを設定してリセット状態にする必要があります。

STOP3モードにおけるI/Oの状態

STOP3モードでは、I/Oはデフォルトでフローティング状態にあります。PWR_APCRレジスタのAPCビットがセットされている場合、I/Oをプルアップ（PWR_PUCRxレジスタを参照）またはプルダウン（PWR_PDCRxレジスタを参照）に設定するか、またはPWR_PUCRxとPWR_PDCRxのどちらのレジスタも設定されていない場合はアナログ状態のまま保持することができます。同じI/Oに対してPWR_PUCRxおよびPWR_PDCRxがセットされている場合、プルダウン設定がプルアップ設定より優先されます。

一部のI/O（セクション 13：汎用I/O（GPIO）に記載）はJTAG/SWデバッグに使用され、STOP3モード中にPWR_PUCRxまたはPWR_PDCRxレジスタの各ビットを1にセットすることによってリセットプルアップまたはプルダウン状態にのみ設定できます。あるいは、ビットが0に保持されている場合はフローティング状態に設定されます。

PC13およびPB2のRTC出力はSTOP3モードで機能します。LSEに使用されるPC14とPC15も機能します。8つのイベント（WKUPx、x = 1 ~ 8）で多重化された24個のウェイクアップピンで、8つのRTCタンパピンも使用できます。

STOP3モードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがセットされたとき、低消費電力モードへの移行で説明されているように、STOP3モードに移行します（STOP3モードへの移行方法の詳細については、表105を参照）。

Flashメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP3モードに移行します。

APBドメインにアクセス中の場合、APBアクセスが終了してから、STOP3モードに移行します。

STOP3モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 61.4：IWDGの機能説明を参照）。
- RCC_BDCR の RTCEN ビットによって有効化されたリアルタイムクロック（RTC）とタンパ（TAMP）カーネルクロック。
- 128分周された内部RCオシレータのLSIクロックまたはLSIクロックは、RCC_BDCRのLSIONおよびLSIPREDIVビットによって設定されます。
- 外部32.768kHzオシレータ（LSE）は、RCC_BDCRのLSEONビットによって設定されます。

STOP3モードの終了

MCUは、低消費電力モードの終了で説明されているようにSTOP3モードを終了します（STOP3モードの終了方法の詳細については、表105を参照）。

割込みまたはウェイクアップイベントの発行によってSTOP3モードを終了する場合、RCC_CFGR1のSTOPWUCKビットがセットされると、システムクロックとしてHSI16が選択されます。STOPWUCKがクリアされた場合、システムクロックとしてMSISが選択されます。MSISを選択すると、より高い周波数（最大24MHz）でウェイクアップできます。

STOP3モードを終了するとき、MCUはレンジ4のRUNモードになります。

表105.STOP3モード

STOP3モード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_CR1のLPMS = 011 – PWR_WUSRのWUFxビットはクリアされている
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXIT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 011 – PWR_WUSRのWUFxビットはクリアされている – 選択されたウェイクアップソースに対応するRTC/TAMPフラグ、クリア
	注： STOP3モードに移行するには、すべてのWUFxと、ウェイクアップ割込みを生成するRTC/TAMPフラグをクリアする必要があります。そうしないと、STOP3モードへの移行手順は完了しますが、STOP3モードは移行後直ちに終了します。
モードの終了	WKUPxピンのエッジ、RTC/TAMPイベント/割込み、NRSTピンでの外部リセット、IWDGリセット、BORリセット
ウェイクアップ遅延時間	最長のウェイクアップ時間：MSISまたはHSI16ウェイクアップ時間と、レギュレータの低電力モードからのウェイクアップ時間 + STOP3モードのFLASHウェイクアップ時間の間。

10.7.10 STANDBYモード

BORがアクティブなとき、最も消費電力が少ないモードは、STANDBYモードです。このモードは、電圧レギュレータを無効にした状態（SRAM2の内容が保持されている場合を除く）のCortex-M33のディープスリープ（deepsleep）モードに基づきます。PLL、HSI16、MSIS、MSIK、およびHSEオシレータもオフになります。

バックアップドメインとSTANDBY回路のレジスタを除いて、SRAMとレジスタの内容は失われます（図32を参照）。SRAM2の内容は、PWR_CR1のRRSB1とRRSB2ビットの設定に応じて部分的または完全に保持できます。この場合、低電力レギュレータがオンになり、SRAM2に対してのみ電源供給されます。

BORは常にSTANDBYモードで使用できます。PWR_CR1のULPMENは、消費電力を最小限に抑えるために、1に設定する必要があります。これには、超低電力モードを強制的にBORにします（BORレベル0選択時のみ使用可能）。

STANDBYモードにおけるI/Oの状態

STANDBYモードでは、I/Oはデフォルトでフローティング状態です。PWR_APCRのAPCビットがセットされている場合、I/Oをプルアップ（PWR_PUCRxを参照）またはプルダウン（PWR_PDCRxを参照）に設定するか、またはPWR_PUCRxとPWR_PDCRxのどちらのレジスタも設定されていない場合はアナログ状態のまま保持することができます。同じ I/O に対して PWR_PUCRx および PWR_PDCRxがセットされている場合、プルダウン設定がプルアップ設定より優先されます。

一部のI/O（セクション 13：汎用I/O（GPIO）に記載）はJTAG/SWデバッグに使用され、STANDBYモード中にPWR_PUCRxまたはPWR_PDCRxレジスタの各ビットを1にセットすることによってリセットプルアップまたはプルダウン状態にのみ設定できます。あるいは、ビットが0に保持されている場合はフローティング状態に設定されます。

PC13およびPB2のRTC出力はSTANDBYモードで機能します。LSEに使用されるPC14とPC15も機能します。8つのイベント（WKUPx、x = 1 ~ 8）で多重化された24個のウェイクアップピンで、8つのRTCタンパピンも使用できます。

STANDBYモードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがセットされたとき、低消費電力モードへの移行で説明されているようにSTANDBYモードに移行します（STANDBYモードへの移行方法の詳細については、表106を参照）。

STANDBYモードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 61.4：IWDGの機能説明を参照）。
- RCC_BDCR の RTCEN ビットによって有効化されたリアルタイムクロック（RTC）とタンパ（TAMP）カーネルクロック。
- 128分周された内部RCオシレータのLSIクロックまたはLSIクロックは、RCC_BDCRのLSIONおよびLSIPREビットによって設定されます。
- 外部32.768kHzオシレータ（LSE）は、RCC_BDCRのLSEONビットによって設定されます。

STANDBYモードの終了

MCUは、低消費電力モードの終了で説明されているようにSTANDBYモードを終了します。PWRステータスレジスタ（PWR_SR）のSBFステータスフラグはMCUがSTANDBYモードにあったことを示します。PWR制御レジスタ3（PWR_CR3）を除くすべてのレジスタは、STANDBYからのウェイクアップ後にリセットされます（STANDBYモードの終了方法の詳細については、表106を参照）。

STANDBYモードを終了するとき、STANDBYモード中にPWR_PUCRx/PWR_PDCRxによってプルアップ/プルダウンに設定されていたI/Oは、STANDBYモード終了からPWR_CR3のAPCビットのクリアまではこの設定を維持します。APCがクリアされると、I/Oは、リセット値に設定されるか、GPIOx_PUPDRレジスタに従ってプルアップ/プルダウン状態に設定されます。PWR_PUCRxまたはPWR_PDCRxレジスタの内容は失われず、次のSTANDBYモード移行に再使用できます。

一部のI/O（セクション 13：汎用I/O（GPIO）に記載）はJTAG/SWデバッグに使用され、リセット後に内部プルアップまたはプルダウンを有効にします。そのため、STANDBYモード終了時と同じリセット値に設定されます。

リセット後に事前に定義したプルアップまたはプルダウン状態になるI/O（一部のJTAG/SWのI/O）、またはSTANDBY モード終了後に GPIOx_PUPDR をプログラムする I/O については、その設定がSTANDBYモード中にPWR_PUCRxまたはPWR_PDCRxに設定された値と異なる場合は、APCがクリアされ、PWR_PUCRxまたはPWR_PDCRxの設定値が解除されるまでプルアップとプルダウンの両方が適用されます。

表106.STANDBYモード

STANDBYモード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_CR1のLPMS = 10x (1) – PWR_WUSRのWUFxビットはクリアされている
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXIT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 10x (1) – PWR_WUSRのWUFxビットはクリアされている – 選択されたウェイクアップソースに対応するRTC/TAMPフラグ、クリア
モードの終了	WKUPxピンのエッジ、RTC/TAMPイベント/割込み、NRSTピンでの外部リセット、IWDGリセット、BORリセット
ウェイクアップ遅延時間	リセットフェーズ

1. STANDBYモードは、PWR_BDCR1のBREN = 1でPWR_CR1のLPMS = 11Xの場合にも移行します。

10.7.11 SHUTDOWNモード

SHUTDOWNモードでは、消費電力を最小限に抑えることができます。このモードは、電圧レギュレータを無効にした状態のディープスリープ（deepsleep）モードに基づきます。結果として、コアドメインの電源がオフになります。PLL、HSI16、MSIS、MSIK、およびHSEオシレータもオフになります。

バックアップドメインのレジスタを除いて、SRAM とレジスタの内容は失われます。BOR は SHUTDOWNモードで使用できません。このモードで電源電圧監視はできないため、バックアップドメインへの切替えはサポートされません。

SHUTDOWNモードにおけるI/Oの状態

SHUTDOWNモードでは、I/Oはデフォルトでフローティング状態にあります。PWR_APCRレジスタのAPCビットがセットされている場合、I/Oをプルアップ（PWR_PUCRxレジスタを参照（x = A ~ J））またはプルダウン（PWR_PDCRxレジスタを参照（x = A ~ J））に設定するか、またはPWR_PUCRxとPWR_PDCRxのどちらのレジスタも設定されていない場合はアナログ状態のまま保持することができます。同じI/Oに対してPWR_PUCRxおよびPWR_PDCRxがセットされている場合、プルダウン設定がプルアップ設定より優先されます。ただし、パワーオンリセットによりSHUTDOWNモードが終了すると、この設定は失われます。

一部のI/O（セクション 13 : 汎用I/O（GPIO）に記載）はJTAG/SWデバッグに使用され、SHUTDOWNモード中にPWR_PUCRxまたはPWR_PDCRxレジスタの各ビットを1にセットすることによってリセットプルアップまたはプルダウン状態にのみ設定できます。あるいは、ビットが0に保持されている場合はフローティング状態に設定されます。

PC13およびPB2のRTC出力はSHUTDOWNモードで機能します。LSEに使用されるPC14とPC15も機能します。8つのイベント（WKUPx、x = 1 ~ 8）で多重化された24個のウェイクアップピンで、8つのRTCタンパピンも使用できます。

SHUTDOWNモードへの移行

MCUは、Cortex-M33システム制御レジスタのSLEEPDEEPビットがクリアされたとき、低消費電力モードへの移行で説明されているように、SHUTDOWNモードに移行します（SHUTDOWNモードへの移行方法の詳細については、表107を参照）。

SHUTDOWNモードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- リアルタイムクロック（RTC）およびタンパ（TAMP）カーネルクロックは、バックアップドメイン制御レジスタ（RCC_BDCR）のRTCENビットで有効化されます。注意：V_{DD}がパワーダウンすると、RTCの内容は失われます。
- 外部32.768kHzオシレータ（LSE）は、バックアップドメイン制御レジスタ（RCC_BDCR）のLSEONビットで設定されます。

注意： **PWRバックアップドメイン制御レジスタ1（PWR_BDCR1）にBRENビットがセットされている場合は、SHUTDOWNモードに移行できません。BREN = 1の場合は、SHUTDOWNモードではなくSTANDBYモードに移行します。**

SHUTDOWNモードの終了

MCUは、低消費電力モードの終了で説明されているように、SHUTDOWNモードを終了します。SHUTDOWN　モードを終了すると、パワーオンリセットが発生します。レジスタはすべて、SHUTDOWNからのウェイクアップ後にリセットされます（バックアップドメイン内にあるものを除く）（SHUTDOWNモードの終了方法の詳細については、表107を参照）。

SHUTDOWNモードを終了するとき、SHUTDOWN中にPWR_PUCRx/PWR_PDCRxレジスタによってプルアップ/プルダウンに設定されていたI/Oは、その設定内容を失い、フローティング状態またはプルアップ/プルダウンのリセット値に設定されます（セクション 13：汎用I/O（GPIO）に記載の一部のI/O）。

表107.SHUTDOWNモード

SHUTDOWNモード	説明
モードへの移行	次の条件下でのWFI（Wait for Interrupt）またはWFE（Wait for Event） – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – ペンディング中の割込み（WFI）やイベント（WFE）なし – PWR_BDCR1のBREN = 0でPWR_CR1のLPMS = 11X – PWR_WUSRのWUFxビットはクリアされている
	次の条件下での割込みサービスルーチン（ISR）終了時 – Cortex-M33システム制御レジスタのSLEEPDEEPビットをセット – SLEEPONEXT = 1 – ペンディング中の割込みなし – PWR_CR1のLPMS = 11X – PWR_WUSRのWUFxビットはクリアされている – 選択されたウェイクアップソースに対応するRTC/TAMPフラグ、クリア
モードの終了	WKUPxピンのエッジ、RTC/TAMPイベント/割込み、NRSTピンでの外部リセット
ウェイクアップ遅延時間	リセットフェーズ

10.7.12 低電力モードでのUSBパワー・マネージメント（STM32U59x/5Ax/5Fx/5Gxのみ）

STOP0およびSTOP1モードでは、USBPWRENビットをセットしたままにすることで、OTG_HS設定を保持できます。これにより、OTG_HSはSTOPモードからMCUをウェイクアップできます。ただし、消費電力を低減するために、STOP0およびSTOP1モードに移行する前に、OTG_HSをシャットオフすることを推奨します。

STOP2およびSTOP3モードでは、OTG_HS設定を保持できません。STOP2およびSTOP3モードに移行する前に、OTG_HSをシャットオフする必要があります。

STOPモードに移行する前にOTG_HSをシャットオフするには、次の手順が必要です。

1. PWR_VOSRのUSBPWRENおよびUSBBOOSTENビットをクリアします。
2. STOPモードで移行をリクエストします。

STOPモードからウェイクアップする際、またOTG_HSを設定する前に、以下を行います。

1. 電圧スケーリングがレンジ1またはレンジ2であることを確認します（PWR_VOSRのVOS[1:0]を使用）。
2. EPODのブースタクロックが有効であることを確認します（RCC_PLL1CFGRのPLL1MBOOST[3:0]を使用）。
3. PWR_VOSRのUSBPWRENおよびUSBBOOSTENビットをセットして、USB内部電力を有効にします。
4. PWR_VOSRのUSBBOOSTRDYがセットされるのを待ちます。

PA11およびPA12 GPIOの使用

PA11およびPA12がOTG_HS_DMおよびOTG_HS_DPの追加機能として使用される場合、GPIOはアナログモードで設定する必要があります（デフォルト設定）。

PA11およびPA12が標準GPIOとして使用されている場合、PWR_VOSRのUSBPWRENおよびVDD11USBDISビットは、アナログ以外のモードでGPIOを設定する前にセットする必要があります。

STOP2、STOP3、およびSTANDBYモードでは、PA11およびPA12を標準GPIOまたはオルタネート機能として使用できます。ただし、STOP2、STOP3、およびSTANDBY低電力モードに移行する場合、OTG_HS PHY電力はハードウェアによってオフになり、PA11（OTG_HS_DMによる駆動）とPA12（OTG_HS_DPによる駆動）は強くプルダウンされます。PWR_CR1のFORCE_USBPWRビットをセットすることで、OTG_HS PHY電源を維持し、PA11およびPA12をGPIOとして使用できます。

注： **FORCE_USBPWRビットをセットすることで、STOP2、STOP3、およびSTANDBYモードの余分な消費電力が誘導されます（VDD11USBDISがセットされている場合は通常50µA）。これが許容できない場合は、FORCE_USBPWRビットをクリアしたままにする必要があり、この場合、STOP2、STOP3、およびSTANDBYモード中、PA11およびPA12はローレベルで維持する必要があります。**

10.7.13 電力モード出力ピン

デバッグを容易にするために、3つの信号をデバイスピンオルタネート機能として使用できます。

- **CSLEEP**

  セットされたとき、CSLEEPはCPUがSLEEPモードであることを示します。WFIまたはWFEが実行されています。クリアされたとき、CSLEEPはCPUがRUNモードであることを示します。

- **CDSTOP**

  セットされたとき、CDSTOPは、CPUドメイン（CD）がCStopモードであることを示します。つまり、以下の条件が満たされていることを意味します。

  – CPU SLEEPDEEP = 1でWFIまたはWFEが実行されました。

  – CPUドメインで動作中のAHB/APBクロックはありません。

  クリアすると、CDSTOPはCPUドメインがCStopモードではないことを示します。AHB/APBクロックはCPUドメインで動作します。

- **SRDSTOP**

  セットされたとき、SRDSTOPは、SmartRunドメイン（SRD）がDStopモードであることを示します。つまり、以下の条件が満たされていることを意味します。

  – CPU SLEEPDEEP = 1でWFIまたはWFEが実行されました。

  – SRDドメインで動作中のAHB/APBクロックはありません。

  クリアすると、SRDSTOPはSmartRunドメインがDStopモードではないことを示します。AHB/APBクロックはSRDドメインで動作します。

注： **自律型ペリフェラルがSTOPモードのバスクロックをリクエストした場合、WFIまたはWFEが実行された後にAHB/APBクロックが動作します。ペリフェラルのバスクロックリクエストは、デバイスの低電力モードへの移行を遅延させるか防止することができます（セクション10.7.2およびセクション10.7.4を参照）。**

次の表に、これらの信号状態に応じたMCU電力モードを示します。

表108.電力モード出力状態対MCU電力モード

CSLEEP	CDSTOP	SRDSTOP	MCU電力モード(1)
0	0	0	RUNモード
1	0	0	SLEEPモードまたは AHB/APBクロックがCPUドメイン（CD）およびSmartRunドメイン（SRD）で動作するSTOP0またはSTOP1モード
1	1	0	AHB/APBクロックがSmartRunドメイン（SRD）で動作するSTOP0、STOP1、またはSTOP2モード
1	1	1	STOP0、STOP1、またはSTOP2モード

1. CSLEEP、CDSTOP、およびSRDSTOPはコアドメインで生成されるため、STOP3、STANDBY、およびSHUTDOWNモードでは駆動されません。

10.8 PWRセキュリティおよび特権保護

10.8.1 PWRセキュリティ保護

TrustZoneセキュリティがFlash_OPTRのTZENで有効化される場合、PWRレジスタのフィールドの一部を非セキュアアクセスから保護することができます。

PWR TrustZoneセキュリティにより、PWR_SECCFGRを通じて以下の機能を保護できます。

- 低電力モード
- ウェイクアップ（WKUP）ピン
- 電圧の検出と監視
- V_{BAT}モード
- I/Oプルアップ/プルダウン設定

その他のPWR設定ビットは次の場合にセキュアです。

- システムクロックの選択がRCCでセキュア：電圧スケーリング（VOS）設定とレギュレータブースタ（BOOSTEN）がセキュアになります。
- GPIOがセキュアとして設定される：STANDBYモードにおいてプルアップ/プルダウン設定の対応ビットがセキュアになります。
- GTZCのUCPD1はセキュアです。すなわち、PWR_UCPDRレジスタはセキュアです。

表109は、PWR_SECCFGRのセキュリティ設定ビットに対するPWRセキュアビットの一覧です。1つ以上の機能がセキュアに設定されると、PWRクロック制御はRCCでセキュアになります。

セキュア保護されたレジスタビットへの非セキュアアクセスは拒否されます。

- セキュアビットは、非セキュア書込みアクセスで書き込まれません（WI）。
- セキュアビットは、非セキュア読出しアクセスで0として読み出されます（RAZ）。

PWR_SECCFGRへの非セキュア書込みアクセスはWIで、GTZCで有効化されたときに不正アクセスイベントおよび割込みが生成されます。これは、非セキュア読出しアクセスで読み出すことができます。

TrustZoneセキュリティが無効な場合（TZEN = 0）、PWR_SECCFGRはRAZ/WIであり、他のすべてのレジスタは非セキュアです。

表109.PWRセキュリティ設定一覧

セキュア設定レジスタ	セキュリティ設定ビット	レジスタ名	セキュアビット	セキュアビットでの非セキュアアクセス
PWR_SECCFGR	適用外(1)	PWR_SECCFGR	すべてのビット	読出しOK。WIおよび不正アクセスイベント
PWR_SECCFGR	最低でも1ビットをセット	PWR_PRIVCFGR	SPRIV	読出しOK。WI
PWR_SECCFGR	LPMSEC	PWR_CR1	すべてのビット	RAZ/WI
		PWR_CR2	すべてのビット	
		PWR_SR	CSSF	WI
PWR_SECCFGR	VDMSEC	PWR_CR3	すべてのビット	RAZ/WI
		PWR_SVMCR	すべてのビット	RAZ/WI

表109.PWRセキュリティ設定一覧 （続き）

セキュア設定レジスタ	セキュリティ設定ビット	レジスタ名	セキュアビット	セキュアビットでの非セキュアアクセス
PWR_SECCFGR	VBSEC	PWR_BDCR1	すべてのビット	RAZ/WI
		PWR_BDCR2	すべてのビット	RAZ/WI
		PWR_DBPR	すべてのビット	RAZ/WI
PWR_SECCFGR	APCSEC	PWR_APCR	すべてのビット	RAZ/WI
PWR_SECCFGR	WUPxSEC（x = 1 ~ 8）	PWR_WUCR1	WUPENx	RAZ/WI
		PWR_WUCR2	WUPPx	RAZ/WI
		PWR_WUCR3	WUSELx	RAZ/WI
		PWR_WUSCR	CWUFx	WI
GTZC_TZSC_SECCFGR	UCPD1SEC	PWR_UCPDR	すべてのビット	RAZ/WI
RCC_SECCFGR	SYSCLKSEC	PWR_VOSR	VOS[1:0]、BOOSTEN、USBPWREN、USBBOOSTEN(2)	RAZ/WI
GPIOx_SECCFGR（x = A、B ~ J）	SECy（y = 0 ~ 15）	PWR_PUCRx（x = A ~ J）	PUy（y = 0 ~ 15）	RAZ/WI
		PWR_PDCRx（x = A ~ J）	PDy（y = 0 ~ 15）	RAZ/WI

1. PWR_SECCFGRは常にセキュアです。
2. USBPWRENおよびUSBBOOSTENはSTM32U59x/5Ax/5Fx/5Gxでのみ使用できます。

10.8.2 WR特権保護

デフォルトでは、リセット後、すべてのPWRレジスタは、特権アクセスと非特権アクセスのどちらでも読み書き可能ですが、PWR_PRIVCFGRへの書込みは特権アクセスでのみ可能です。PWR_PRIVCFGRは、セキュアおよび非セキュア、特権および非特権アクセスによって読み出すことができます。

PWR_PRIVCFGRのSPRIVは、セキュア特権アクセスでのみ書き込むことができます。このビットは、（表109に示されているように、PWR_SECCFGR、GTZC、RCC、またはGPIOによって定義される）すべてのPWRセキュア機能の特権アクセスを設定します。
PWR_PRIVCFGRでSPRIVがセットされている場合：

- PWR セキュアビットは、PWR_SECCFGR も含めて、特権アクセスでのみ書き込むことができます。
- PWR セキュアビットは、特権アクセスでのみ読み出すことができますが、PWR_SECCFGR と PWR_PRIVCFGRは、特権または非特権アクセスによって読み出すことができます。
- 特権PWRビットまたはレジスタへの非特権アクセスは破棄されます。すなわち、ビットはゼロとして読み出され、これらのビットへの書込みは無視されます（RAZ/WI）。

PWR_PRIVCFGRのNSPRIVは、セキュアまたは非セキュア特権アクセスでのみ書き込むことができます。このビットは、（表109に示されているように、PWR_SECCFGR、GTZC、RCC、またはGPIOによって定義される）非セキュアとして設定されているすべてのPWRセキュア機能の特権アクセスを設定します。
PWR_PRIVCFGRでNSPRIVがセットされている場合：

- 非セキュアとして設定されたPWRセキュリティ保護可能ビットは、特権アクセスでのみ書き込むことができます。

- 非セキュアとして設定されたPWRセキュリティ保護可能ビットは、特権または非特権アクセスによって読み出すことができるPWR_PRIVCFGRを除いて、特権アクセスでのみ読み出すことができます。
- PWR_VOSR、PWR_SR、PWR_SVMSR、PWR_BDSR、およびPWR_WUSRのVOSRDYおよびBOOSTRDYビットは、特権または非特権アクセスで読み出すことができます。
- 特権PWRビットまたはレジスタへの非特権アクセスは破棄されます。すなわち、ビットはゼロとして読み出され、これらのビットへの書込みは無視されます（RAZ/WI）。

10.9 PWR割込み

次の表に割込みソースの概要と、その制御方法について示します。

表110.PWR割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP、STOP0、1、2モードの終了	STOP3、STANDBY、SHUTDOWNモードの終了
PWR_S3WU(1)	ウェイクアップ割込みフラグ	WUFx（x = 1 ~ 8）	WUPENx（x = 1 ~ 8）	CWUFx = 1を書き込む（x = 1 ~ 8）	なし	あり(2)
PVD_PVM	EXTI ライン16 を通じてプログラム可能な電圧検出器	PVDO	EXTIライン16有効	EXTI PIF16 = 1を書き込む	あり	なし
	EXTIライン19経由のUSB電源電圧モニタ	VDDUSBRDY	EXTIライン19有効	EXTI PIF19 = 1を書き込む	あり	なし
	EXTIライン20経由のV_{DDIO2}電源電圧モニタ	VDDIO2RDY	EXTIライン20有効	EXTI PIF20 = 1を書き込む		
	EXTIライン21経由のアナログ電源電圧モニタ1	VDDA1RDY	EXTIライン21有効	EXTI PIF21 = 1を書き込む		
	EXTIライン22経由のアナログ電源電圧モニタ2	VDDA2RDY	EXTIライン22有効	EXTI PIF22 = 1を書き込む		

1. PWR_S3WU割込みはデバイスがSTOP3モードの場合にのみ生成されます（RUN、SLEEP、STOP0、STOP1、およびSTOP2モードには適用されません）。
2. 割込みのみがSTOP3モードからウェイクアップできます（イベント時は不可）。

10.10 PWRレジスタ

10.10.1 PWR制御レジスタ1（PWR_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

このレジスタは、PWR_SECCFGRのLPMSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのLPMSEC = 1かつSPRIV = 1、またはLPMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FORCE_USBPWR	Res.	SRAM6PD	SRAM5PD	SRAM4PD	SRAM3PD	SRAM2PD	SRAM1PD	ULPMEN	RRSB2	RRSB1	Res.	Res.	LPMS[2:0]		
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **FORCE_USBPWR**：STOP2、STOP3、およびSTANDBY低電力モード中はOTG_HS PHY電力が維持されます。

0：低電力モード中にOTG_HS PHY電力が維持されません。

1：低電力モード中にOTG_HS PHY電力が維持されます。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **SRAM6PD**：SRAM6パワーダウン

このビットは、SRAM6の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM6電源オン

1：SRAM6電源オフ

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット12 **SRAM5PD**：SRAM5パワーダウン

このビットは、SRAM5の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM5電源オン

1：SRAM5電源オフ

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット11 **SRAM4PD**：SRAM4パワーダウン

このビットは、SRAM4の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM4電源オン

1：SRAM4電源オフ

ビット10 **SRAM3PD**：SRAM3パワーダウン

このビットは、SRAM3の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM3電源オン

1：SRAM3電源オフ

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット9 **SRAM2PD**：SRAM2パワーダウン

このビットは、SRAM2の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM2電源オン

1：SRAM2電源オフ

ビット8 **SRAM1PD**：SRAM1パワーダウン

このビットは、SRAM1の電源をオフにして、消費電力を低減するために使用されます。

0：SRAM1電源オン

1：SRAM1電源オフ

ビット7 **ULPMEN**：BOR0超低電力モード

このビットは、BORを不連続モードで設定して、消費電力を低減するために使用されます。このビットは、BORレベル0が選択されているとき、かつデバイスがSTANDBYモードのときにのみ有効になります。

0：STANDBYモードの連続（通常）モードで動作するBORレベル0

1：STANDBYモードの不連続（超低電力）モードで動作するBORレベル0

注意： このビットは、SHUTDOWNモードで消費電力を最小限に抑えるためにセットする必要があります。

ビット6 **RRSB2**：STOP3およびSTANDBYモードでのSRAM2ページ2の保持

このビットは、STOP3およびSTANDBYモードでSRAM2ページ2の内容を保持するために使用されます。SRAM2ページ2はSRAM2の最後の56KBに対応します（SRAM2のベースアドレス + 0x2000からSRAM2のベースアドレス + 0xFFFF）。

0：STOP3およびSTANDBYモードで、SRAM2ページ2の内容を保持しない

1：STOP3およびSTANDBYモードで、SRAM2ページ2の内容を保持する

注： このビットによるSHUTDOWNモードへの影響はありません。
このビットがセットされると、バックアップSRAMも保持されます。

ビット5 **RRSB1**：STOP3およびSTANDBYモードでのSRAM2ページ1の保持

このビットは、STOP3およびSTANDBYモードでSRAM2ページ1の内容を保持するために使用されます。SRAM2ページ1はSRAM2の最初の8KBに対応します（SRAM2のベースアドレスからSRAM2のベースアドレス + 0x1FFF）。

0：STOP3およびSTANDBYモードで、SRAM2ページ1の内容を保持しない

1：STOP3およびSTANDBYモードで、SRAM2ページ1の内容を保持する

注： このビットによるSHUTDOWNモードへの影響はありません。
このビットがセットされると、バックアップSRAMも保持されます。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **LPMS[2:0]**：低電力モード選択

これらのビットは、CPUがディープスリープ（deepsleep）モードへ移行する時に、移行する低電力モードを選択します。

000：STOP0モード

001：STOP1モード

010：STOP2モード

011：STOP3モード

10x：STANDBYモード（STANDBYモードには、PWR_BDCR1のBREN = 1でPWR_CR1のLPMS = 11Xの場合にも移行する）

11x：PWR_BDCR1のBREN = 0の場合はSHUTDOWNモード

10.10.2 PWR制御レジスタ2（PWR_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

このレジスタは、PWR_SECCFGRのLPMSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのLPMSEC = 1かつSPRIV = 1、またはLPMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRDRUN	Res.	Res.	Res.	Res.	JPEGRAMPDS	DSIRAMPDS	GPRAMPDS	SRAM3PDS8	SRAM3PDS7	SRAM3PDS6	SRAM3PDS5	SRAM3PDS4	SRAM3PDS3	SRAM3PDS2	SRAM3PDS1
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FLASHFWU	SRAM4FWU	PKARAMPDS	PRAMPDS	DMA2DRAMPDS	DC1RAMPDS	ICRAMPDS	DC2RAMPDS	SRAM4PDS	SRAM2PDS2	SRAM2PDS1	Res.	SRAM1PDS3	SRAM1PDS2	SRAM1PDS1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット31 **SRDRUN**：RUNモードのSmartRunドメイン

0：STOP0/1/2モードで、SmartRunドメインのAHB3およびAPB3クロックをデフォルトで無効化

1：STOP0/1/2モードで、SmartRunドメインのAHB3およびAPB3クロックを有効な状態で保持

ビット30:27 予約済みであり、リセット値に保持する必要があります。

ビット26 **JPEGRAMPDS**：STOP0/1モードでJPEG SRAMのパワーダウン

JPEG SRAMの内容は、STOP2およびSTOP3モードで常に失われます。

0：STOP0およびSTOP1モードで、JPEG SRAMの内容を保持する

1：STOP0およびSTOP1モードで、JPEG SRAMの内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット25 **DSIRAMPDS**：STOP0/1モードでDSI SRAMのパワーダウン

DSI SRAMの内容は、STOP2およびSTOP3モードで常に失われます。

0：STOP0およびSTOP1モードで、DSI SRAMの内容を保持する

1：STOP0およびSTOP1モードで、DSI SRAMの内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット24 **GPRAMPDS**：すべてのSTOPモードでのグラフィックペリフェラル（LTDC、GFXMMU）SRAMのパワーダウン

0：STOPモードで、グラフィックペリフェラルSRAMの内容を保持する

1：STOPモードで、グラフィックペリフェラルSRAMの内容を失う

注： LTDC SRAMの内容は、STOP2およびSTOP3モードで常に失われます。STOP0およびSTOP1モードでのみ保持できます。
このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット23 **SRAM3PDS8**：すべてのSTOPモードでのSRAM3ページ8（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ8の内容を保持する

1：STOPモードで、SRAM3ページ8の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット22 **SRAM3PDS7**：すべてのSTOPモードでのSRAM3ページ7（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ7の内容を保持する

1：STOPモードで、SRAM3ページ7の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット21 **SRAM3PDS6**：すべてのSTOPモードでのSRAM3ページ6（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ6の内容を保持する

1：STOPモードで、SRAM3ページ6の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット20 **SRAM3PDS5**：すべてのSTOPモードでのSRAM3ページ5（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ5の内容を保持する

1：STOPモードで、SRAM3ページ5の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット19 **SRAM3PDS4**：すべてのSTOPモードでのSRAM3ページ4（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ4の内容を保持する

1：STOPモードで、SRAM3ページ4の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット18 **SRAM3PDS3**：すべてのSTOPモードでのSRAM3ページ3（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ3の内容を保持する

1：STOPモードで、SRAM3ページ3の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット17 **SRAM3PDS2**：すべてのSTOPモードでのSRAM3ページ2（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ2の内容を保持する

1：STOPモードで、SRAM3ページ2の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット16 **SRAM3PDS1**：すべてのSTOPモードでのSRAM3ページ1（64KB）のパワーダウン

0：STOPモードで、SRAM3ページ1の内容を保持する

1：STOPモードで、SRAM3ページ1の内容を失う

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **FLASHFWU**：FlashメモリのSTOP0およびSTOP1モードからの高速ウェイクアップ
このビットは、STOP0またはSTOP1モードの終了時に低い消費電力とウェイクアップ時間の間で最高のトレードオフを実現するために使用されます。
このビットがセットされた場合、FlashメモリはSTOP0およびSTOP1モードで通常モードのまま保持されます。より消費電力の高い、高速起動時間を提供します。
0：STOP0/1モードで、Flashメモリを低電力モードに移行します（低消費電力）。
1：FlashメモリはSTOP0/1モードで通常モードのまま保持されます（より速いウェイクアップ時間）。

ビット13 **SRAM4FWU**：STOP0/1/2モードからのSRAM4高速ウェイクアップ
このビットは、低い消費電力とウェイクアップ時間の間で最高のトレードオフを実現するために使用されます。SRAM4ウェイクアップ時間は、STOP0/1/2モード終了時のウェイクアップ時間を増やし、またSTOPモード中のSRAM4へのLPDMAアクセス時間を増やします。
0：STOP0/1/2モードで、SRAM4を低電力モードに移行します（低消費電力のソースバイアス）。
1：SRAM4はSTOP0/1/2モードで通常モードのまま保持されます（消費電力は高くなりますが、SRAM4のウェイクアップ時間はなくなります）。

ビット12 **PKARAMPDS**：すべてのSTOPモード（STOP0/1/2/3）でのPKA SRAMのパワーダウン
0：STOPモードで、PKA SRAMの内容を保持する
1：STOPモードで、PKA SRAMの内容を失う
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット11 **PRAMPDS**：すべてのSTOPモード（STOP0/1/2/3）でのFMAC、FDCAN、およびUSB/OTG_FS/OTG_HSSRAMのパワーダウン
0：STOPモードで、FMAC、FDCAN、およびUSB/OTG_FS/OTG_HSSRAMの内容を保持する
1：STOPモードで、FMAC、FDCAN、およびUSB/OTG_FS/OTG_HSSRAMの内容を失う

ビット10 **DMA2DRAMPDS**：すべてのSTOPモードでのDMA2D SRAMのパワーダウン
0：STOPモードで、DMA2D SRAMの内容を保持する
1：STOPモードで、DMA2D SRAMの内容を失う
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット9 **DC1RAMPDS**：すべてのSTOPモードでのDCACHE1 SRAMのパワーダウン
0：STOPモードで、DCACHE1 SRAMの内容を保持する
1：STOPモードで、DCACHE1 SRAMの内容を失う

ビット8 **ICRAMPDS**：すべてのSTOPモードでのICACHE SRAMのパワーダウン
0：STOPモードで、ICACHE SRAMの内容を保持する
1：STOPモードで、ICACHE SRAMの内容を失う

ビット7 **DC2RAMPDS**：すべてのSTOPモードでのDCACHE2 SRAMのパワーダウン
0：STOPモードで、DCACHE2 SRAMの内容を保持する
1：STOPモードで、DCACHE2 SRAMの内容を失う
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット6 **SRAM4PDS**：すべてのSTOPモードでのSRAM4のパワーダウン
0：STOPモードで、SRAM4の内容を保持する
1：STOPモードで、SRAM4の内容を失う

ビット5 **SRAM2PDS2**：STOP0/1/2モードでのSRAM2ページ2（56KB）のパワーダウン
0：STOPモードで、SRAM2ページ2の内容を保持する
1：STOPモードで、SRAM2ページ2の内容を失う
注： STOP3でのSRAM2ページ2の保持は、PWR_CR1のRRSB2ビットで制御されます。

ビット4 **SRAM2PDS1**：STOP0/1/2モードでのSRAM2ページ1（8KB）のパワーダウン
0：STOPモードで、SRAM2ページ1の内容を保持する
1：STOPモードで、SRAM2ページ1の内容を失う
注： STOP3でのSRAM2ページ1の保持は、PWR_CR1のRRSB1ビットで制御されます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SRAM1PDS3**：すべてのSTOPモードでのSRAM1ページ3（64KB）のパワーダウン
0：STOPモードで、SRAM1ページ3の内容を保持する
1：STOPモードで、SRAM1ページ3の内容を失う

ビット1 **SRAM1PDS2**：すべてのSTOPモードでのSRAM1ページ2（64KB）のパワーダウン
0：STOPモードで、SRAM1ページ2の内容を保持する
1：STOPモードで、SRAM1ページ2の内容を失う

ビット0 **SRAM1PDS1**：すべてのSTOPモードでのSRAM1ページ1（64KB）のパワーダウン
0：STOPモードで、SRAM1ページ1の内容を保持する
1：STOPモードで、SRAM1ページ1の内容を失う

10.10.3 PWR制御レジスタ3（PWR_CR3）

アドレスオフセット：0x08

パワーオンリセット値：0x0000 0000

STANDBYモードの終了：影響を受けない
システムリセット：0にクリアされるREGSELを除いて影響されません。

このレジスタは、PWR_SECCFGRのVDMSEC = 1のとき、非セキュアアクセスから保護されます。
このレジスタは、PWR_PRIVCFGRのVDMSEC = 1かつSPRIV = 1、またはVDMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSTEN	REGSEL	Res.
													rw	rw	

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FSTEN**：高速ソフトウェア起動
0：LDO/SMPS高速起動の無効化（制限された突入電流）
1：LDO/SMPS高速起動は有効です。

ビット1 **REGSEL**：レギュレータ選択
0：LDOの選択
1：SMPSの選択
注： REGSELは予約済みで、SMPSなしのパッケージではリセット値に保持する必要があります。

ビット0 予約済みであり、リセット値に保持する必要があります。

10.10.4 PWR電圧スケーリングレジスタ（PWR_VOSR）

アドレスオフセット：0x0C

リセット値：0x0000 8000

一部のレジスタ･フィールドは、RCC_SECCFGRによって非セキュアアクセスから保護されます。これらのフィールドは、PWR_PRIVCFGRによって非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VDD11USBDIS	USBBOOSTEN	USBPWREN	BOOSTEN	VOS[1:0]	
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VOSRDY	BOOSTRDY	USBBOOSTRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r													

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **VDD11USBDIS**：OTG_HS VDD11USBの無効化

このビットは、RCC_SECCFGRのSYSCLKSEC = 1のとき、非セキュアアクセスから保護されます。このビットは、RCC_SECCFGRのSYSCLKSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSYSCLKSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

0：VDD11USBは有効です。

1：VDD11USBは無効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット20 **USBBOOSTEN**：OTG_HS EPODブースタの有効化

このビットは、RCC_SECCFGRのSYSCLKSEC = 1のとき、非セキュアアクセスから保護されます。このビットは、RCC_SECCFGRのSYSCLKSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSYSCLKSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

このビットは、OTG_HSを有効にする前に、レンジ1およびレンジ2でセットする必要があります。

このビットは、すべてのSTOPモードで実行中にリセットされます。

0：OTG_HSブースタは無効です。

1：OTG_HSブースタは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット19 **USBPWREN**：OTG_HSパワーの有効化

このビットは、RCC_SECCFGRのSYSCLKSEC = 1のとき、非セキュアアクセスから保護されます。このビットは、RCC_SECCFGRのSYSCLKSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSYSCLKSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

0：OTG_HSパワーは無効です。

1：OTG_HSパワーは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット18 **BOOSTEN**：EPODブースタの有効化

このビットは、RCC_SECCFGRのSYSCLKSEC = 1のとき、非セキュアアクセスから保護されます。このビットは、RCC_SECCFGRのSYSCLKSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSYSCLKSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

このビットは、システムクロック周波数を55MHzより高くする前に、レンジ1とレンジ2でセットする必要があります。このビットは、すべてのSTOPモードで実行中にリセットされます。

0：ブースタは無効です。

1：ブースタは有効です。

ビット17:16 **VOS[1:0]**：電圧スケーリングレンジ選択

このフィールドは、RCC_SECCFGRのSYSCLKSEC = 1のとき、非セキュアアクセスから保護されます。このフィールドは、RCC_SECCFGRのSYSCLKSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSYSCLKSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

00：レンジ4（最小電力）

01：レンジ3

10：レンジ2

11：レンジ1（最大周波数）

ビット15 **VOSRDY**：V_{CORE}電圧スケーリング出力選択のVOSRDYレディビット

0：動作可能状態ではありません。電圧レベルが選択されているVOSのレベルを下回っています。

1：動作可能状態です。電圧レベルが選択されているVOSのレベル以上です。

ビット14 **BOOSTRDY**：EPODブースタは動作可能状態です。

このビットは、パワーブースタの起動時間になったときに、ハードウェアによって1とセットされます。システムクロック周波数は、このビットがセットされた後でのみ55MHzより高い周波数に切り替えることができます。

0：電力ブースタが動作可能状態ではありません。

1：電力ブースタは動作可能状態です。

ビット13 **USBBOOSTRDY**：OTG_HS EPODブースタは動作可能状態です。

このビットは、パワーブースタの起動時間になったときに、ハードウェアによって1とセットされます。OTG_HSクロックは、このビットをセットした後でのみ提供できます。

0：OTG_HS電力ブースタが動作可能状態ではありません。

1：OTG_HS電力ブースタが動作可能状態です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット12:0 予約済みであり、リセット値に保持する必要があります。

10.10.5 PWR電源電圧モニタの制御レジスタ（PWR_SVMCR）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタは、PWR_SECCFGRのVDMSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのVDMSEC = 1かつSPRIV = 1、またはVDMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	ASV	IO2SV	USV	AVM2EN	AVM1EN	IO2VMEN	UVMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDLS[2:0]			PVDE	Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **ASV**：V_{DDA}独立したアナログ電源有効

このビットは、電気的および論理的アイソレーションの目的でV_{DDA}電源を検証するために使用されます。このビットをセットすると、アナログペリフェラルの使用が必須になります。V_{DDA}が常にアプリケーションに存在しない場合、V_{DDA}電圧モニタを使用してこの電源が準備状態かどうかを判断できます。

0：V_{DDA}なし。論理的および電気的アイソレーションが適用され、この電源は無視されます。

1：有効なV_{DDA}

ビット29 **IO2SV**：V_{DDIO2}独立したI/O電源有効

このビットは、電気的および論理的アイソレーションの目的でV_{DDIO2}電源を検証するために使用されます。PG[15:2]を使用するには、このビットのセットは必須です。V_{DDIO2}が常にアプリケーションに存在しない場合、V_{DDIO2}電圧モニタを使用してこの電源が準備状態かどうかを判断できます。

0：V_{DDIO2}なし。論理的および電気的アイソレーションが適用され、この電源は無視されます。

1：有効なV_{DDIO2}

ビット28 **USV**：V_{DDUSB}独立したUSB電源有効

このビットは、電気的および論理的アイソレーションの目的でV_{DDUSB}電源を検証するために使用されます。

USB/OTG_FS/OTG_HSを使用するには、このビットのセットは必須です。V_{DDUSB}が常にアプリケーションに存在しない場合、V_{DDUSB}電圧モニタを使用してこの電源が準備状態かどうかを判断できます。

0：V_{DDUSB}なし。論理的および電気的アイソレーションが適用され、この電源は無視されます。

1：有効なV_{DDUSB}

ビット27 **AVM2EN**：V_{DDA}独立型アナログ電源電圧モニタ2の有効化（1.8V閾値）

0:V_{DDA}電圧モニタ2は無効です。

1:V_{DDA}電圧モニタ2は有効です。

ビット26 **AVM1EN**：V_{DDA}独立型アナログ電源電圧モニタ1の有効化（1.6V閾値）

0:V_{DDA}電圧モニタ1無効

1:V_{DDA}電圧モニタ1は有効です。

ビット25 **IO2VMEN**：V_{DDIO2}独立型I/O電圧モニタの有効化

0:V_{DDIO2}電圧モニタは無効です。

1:V_{DDIO2}電圧モニタは有効です。

ビット24 **UVMEN**：V_{DDUSB}独立型USB電圧モニタの有効化

0:V_{DDUSB}電圧モニタは無効です。

1:V_{DDUSB}電圧モニタは有効です。

ビット23:8 予約済みであり、リセット値に保持する必要があります。

ビット7:5 **PVDLS[2:0]**：プログラム可能な電圧検出器（PVD）レベル選択

これらのビットは、PVDによって検出される電圧閾値を選択します。

000：約2.0V のV_{PVD0}

001：約2.2V のV_{PVD1}

010：約2.4V のV_{PVD2}

011：約2.5V のV_{PVD3}

100：約2.6V のV_{PVD4}

101：約2.8V のV_{PVD5}

110：約2.9V のV_{PVD6}

111：外部入力アナログ電圧PVD_IN（内部的にVREFINTと比較されます）

ビット4 **PVDE**：プログラム可能な電圧検出器の有効化

0：PVDは無効です。

1：PVDは有効です。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

10.10.6 PWRウェイクアップ制御レジスタ1（PWR_WUCR1）

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

アドレスオフセット：0x14

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各WUPENx（x = 1 ~ 8）は、PWR_SECCFGRのWUPxSEC = 1のとき、非セキュアアクセスから保護されます。各WUPENxは、PWR_SECCFGRのWUPxSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはWUPxSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPEN8	WUPEN7	WUPEN6	WUPEN5	WUPEN4	WUPEN3	WUPEN2	WUPEN1
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WUPEN8**：ウェイクアップピンWKUP8の有効化

0：WKUP8は無効です。

1：WKUP8は有効です。

ビット6 **WUPEN7**：ウェイクアップピンWKUP7の有効化

0：WKUP7は無効です。

1：WKUP7は有効です。

ビット5 **WUPEN6**：ウェイクアップピンWKUP6の有効化

0：WKUP6は無効です。

1：WKUP6は有効です。

ビット4 **WUPEN5**：ウェイクアップピンWKUP5の有効化

0：WKUP5は無効です。

1：WKUP5は有効です。

ビット3 **WUPEN4**：ウェイクアップピンWKUP4の有効化
0：WKUP4は無効です。
1：WKUP4は有効です。

ビット2 **WUPEN3**：ウェイクアップピンWKUP3の有効化
0：WKUP3は無効です。
1：WKUP3は有効です。

ビット1 **WUPEN2**：ウェイクアップピンWKUP2の有効化
0：WKUP2は無効です。
1：WKUP2は有効です。

ビット0 **WUPEN1**：ウェイクアップピンWKUP1の有効化
0：WKUP1は無効です。
1：WKUP1は有効です。

10.10.7 PWRウェイクアップ制御レジスタ2（PWR_WUCR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各WUPPx（x = 1 ~ 8）は、PWR_SECCFGRのWUPxSEC = 1のとき、非セキュアアクセスから保護されます。各WUPPxは、PWR_SECCFGRのWUPxSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはWUPxSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPP8	WUPP7	WUPP6	WUPP5	WUPP4	WUPP3	WUPP2	WUPP1
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WUPP8**：ウェイクアップピンWKUP8の極性
WUPEN8 = 0の場合、このビットを設定する必要があります。
0：ハイ･レベルで検出します（立ち上がりエッジ）。
1：ロー･レベルで検出します（立ち下がりエッジ）。

ビット6 **WUPP7**：ウェイクアップピンWKUP7の極性
WUPEN7 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット5 **WUPP6**：ウェイクアップピンWKUP6の極性
WUPEN6 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット4 **WUPP5**：ウェイクアップピンWKUP5の極性
WUPEN5 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット3 **WUPP4**：ウェイクアップピンWKUP4の極性
WUPEN4 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット2 **WUPP3**：ウェイクアップピンWKUP3の極性
WUPEN3 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット1 **WUPP2**：ウェイクアップピンWKUP2の極性
WUPEN2 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット0 **WUPP1**：ウェイクアップピンWKUP1の極性。
WUPEN1 = 0の場合、このビットを設定する必要があります。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

10.10.8 PWRウェイクアップ制御レジスタ3（PWR_WUCR3）

アドレスオフセット：0x1C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各WUSELx（x = 1 ~ 8）は、PWR_SECCFGRのWUPxSEC = 1のとき、非セキュアアクセスから保護されます。各WUSELxは、PWR_SECCFGRのWUPxSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはWUPxSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUSEL8 [1:0]		WUSEL7 [1:0]		WUSEL6 [1:0]		WUSEL5 [1:0]		WUSEL4 [1:0]		WUSEL3 [1:0]		WUSEL2 [1:0]		WUSEL1 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:14 **WUSEL8[1:0]**：ウェイクアップピンWKUP8選択
WUPEN8 = 0の場合、このフィールドを設定する必要があります。
00：WKUP8_0
01：WKUP8_1
10：WKUP8_2
11：WKUP8_3

ビット13:12 **WUSEL7[1:0]**：ウェイクアップピンWKUP7選択
WUPEN7 = 0の場合、このフィールドを設定する必要があります。
00：WKUP7_0
01：WKUP7_1
10：WKUP7_2
11：WKUP7_3

ビット11:10 **WUSEL6[1:0]**：ウェイクアップピンWKUP6選択
WUPEN6 = 0の場合、このフィールドを設定する必要があります。
00：WKUP6_0
01：WKUP6_1
10：WKUP6_2
11：WKUP6_3

ビット9:8 **WUSEL5[1:0]**：ウェイクアップピンWKUP5選択
WUPEN5 = 0の場合、このフィールドを設定する必要があります。
00：WKUP5_0
01：WKUP5_1
10：WKUP5_2
11：WKUP5_3

ビット7:6 **WUSEL4[1:0]**：ウェイクアップピンWKUP4選択
WUPEN4 = 0の場合、このフィールドを設定する必要があります。
00：WKUP4_0
01：WKUP4_1
10：WKUP4_2
11：WKUP4_3

ビット5:4 **WUSEL3[1:0]**：ウェイクアップピンWKUP3選択
WUPEN3 = 0の場合、このフィールドを設定する必要があります。
00：WKUP3_0
01：WKUP3_1
10：WKUP3_2
11：WKUP3_3

ビット3:2 **WUSEL2[1:0]**：ウェイクアップピンWKUP2選択
WUPEN2 = 0の場合、このフィールドを設定する必要があります。
00：WKUP2_0
01：WKUP2_1
10：WKUP2_2
11：WKUP2_3

ビット1:0 **WUSEL1[1:0]**：ウェイクアップピンWKUP1選択
WUPEN1 = 0の場合、このフィールドを設定する必要があります。
00：WKUP0_0
01：WKUP0_1
10：WKUP0_2
11：WKUP0_3

10.10.9 PWRバックアップドメイン制御レジスタ1（PWR_BDCR1）

アドレスオフセット：0x20

バックアップドメインリセット値：0x0000 0000

パワーオンリセット：影響なし

（STANDBYモードの終了の影響を受けないリセット値）

システムリセット：影響なし

このレジスタは、PWR_DBPRのDBPがクリアされているとき、書込み保護されます。このレジスタは、PWR_SECCFGRのVBSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのVBSEC = 1かつSPRIV = 1、またはVBSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
											rw				rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **MONEN**：バックアップドメイン電圧および温度監視の有効化

0：バックアップドメイン電圧および温度監視は無効です。

1：バックアップドメイン電圧および温度監視は有効です。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **BREN**：STANDBYおよびV_{BAT}モードでのバックアップRAMの保持

このビットがセットされた場合、バックアップRAMの内容はSTANDBY(1)およびV_{BAT}モードに保持されます。

BRENがリセットされると、RUN、SLEEP、およびSTOPモードではバックアップRAMをそのまま使用できます。ただし、STANDBY、SHUTDOWN、およびV_{BAT}モードではその内容は失われることになります。このビットは、レギュレータがLDOのときのみ書き込むことができます。これは、SMPSに切り替える前に設定する必要があります。

0：STANDBY (1) およびV_{BAT}モードで、バックアップRAMの内容を失う

1：STANDBYおよびV_{BAT}モードで、バックアップRAMの内容が保存される

注： バックアップRAMはSHUTDOWNモードでは保存できません。

1. SRAM2が保持されていない場合、バックアップSRAMの内容はSTANDBYモードで失われます。RRSB1またはRRSB2ビットのどちらかがSTANDBYモードでセットされると、バックアップSRAMも保持されます。

10.10.10 PWRバックアップドメイン制御レジスタ2（PWR_BDCR2）

アドレスオフセット：0x24

パワーオンリセット値: 0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

システムリセット：影響なし

このレジスタは、PWR_SECCFGRのVBSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのVBSEC = 1かつSPRIV = 1、またはVBSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **VBRS：V_{BAT}充電抵抗選択**

0：5kΩの抵抗でV_{BAT}を充電します。

1：1.5kΩの抵抗でV_{BAT}を充電します。

ビット0 **VBE：V_{BAT}充電の有効化**

0：V_{BAT}バッテリ充電は無効です。

1：V_{BAT}バッテリ充電は有効です。

10.10.11 PWRディセーブルバックアップドメインレジスタ（PWR_DBPR）

アドレスオフセット：0x28

リセット値：0x0000 0000

このレジスタは、PWR_SECCFGRのVBSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのVBSEC = 1かつSPRIV = 1、またはVBSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBP
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **DBP**：バックアップドメイン書込み保護の無効化

リセット状態で、バックアップドメインのすべてのレジスタとSRAMは不要な書込みアクセスから保護されます。これらのレジスタへの書込みアクセスを可能にするには、このビットをセットする必要があります。

0：バックアップドメインへの書込みアクセスは無効です。

1：バックアップドメインへの書込みアクセスは有効です。

10.10.12 PWR UCPDレジスタ（PWR_UCPDR）

アドレスオフセット：0x2C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

このレジスタは、TZSC_SECCFGRのUCPD1SEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのUCPD1SEC = 1かつSPRIV = 1、またはUCPD1SEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD_STBY	UCPD_DBDIS
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **UCPD_STBY**：UCPD STOP3およびSTANDBYモード

このビットがセットされると、STOP3およびSTANDBYモードにおけるUCPDの設定を記憶するために使用されます。

UCPDを使用する場合、STOP3またはSTANDBYモードに入る直前にこのビットに1を書き込む必要があります。STOP3またはSTANDBYモード終了後およびUCPDレジスタへの書込みの前には0を書き込んでください。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

ビット0 **UCPD_DBDIS**：UCPDのデッドバッテリの無効化

リセット終了後、USBタイプCの「デッドバッテリ」動作が有効になり、CC1およびCC2ピンがプルダウンの影響を受ける可能性があります。あらゆる場合にこのビットを無効にして、プルダウンを停止するか、またはUCPDに制御を引き渡すことを推奨します（無効にする前にUCPDを初期化する必要があります）。

0：UCPDx_CC1およびUCPDx_CC2ピンにおけるUCPDのデッドバッテリプルダウン動作を有効化

1：UCPDx_CC1およびUCPDx_CC2ピンにおけるUCPDのデッドバッテリプルダウン動作を無効化

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みとみなし、リセット値に保持します。

10.10.13 PWRセキュリティ設定レジスタ（PWR_SECCFGR）

アドレスオフセット：0x30

リセット値：0x0000 0000

このレジスタは、アクセスがセキュアなときのみ書き込むことができます。セキュアまたは非セキュアアクセスによって読み出すことができます。このレジスタは、PWR_PRIVCFGRのSPRIV = 1のとき、非特権書込みアクセスから書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
APCSEC	VBSEC	VDMSEC	LPMSEC	Res.	Res.	Res.	Res.	WUP8SEC	WUP7SEC	WUP6SEC	WUP5SEC	WUP4SEC	WUP3SEC	WUP2SEC	WUP1SEC
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **APCSEC**：プルアップ/プルダウンセキュア保護

0：PWR_APCRは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_APCRは、セキュアアクセスによってのみ読み書きができます。

ビット14 **VBSEC**：バックアップドメインのセキュア保護

0：PWR_BDCR1、PWR_BDCR2、およびPWR_DBPRは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_BDCR1、PWR_BDCR2、およびPWR_DBPRは、セキュアアクセスによってのみ読み書きができます。

ビット13 **VDMSEC**：電圧検出と監視のセキュア保護

0：PWR_SVMCRおよびPWR_CR3は、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_SVMCRおよびPWR_CR3は、セキュアアクセスによってのみ読み書きができます。

ビット12 **LPMSEC**：低電力モードのセキュア保護

0：PWR_SRのPWR_CR1、PWR_CR2、およびCSSFは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_SRのPWR_CR1、PWR_CR2、およびCSSFは、セキュアアクセスによってのみ読み書きができます。

ビット11:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WUP8SEC**：WUP8セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP8ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP8ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット6 **WUP7SEC**：WUP7セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP7ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP7ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット5 **WUP6SEC**：WUP6セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP6ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP6ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット4 **WUP5SEC**：WUP5セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP5ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP5ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット3 **WUP4SEC**：WUP4セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP4ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP4ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット2 **WUP3SEC**：WUP3セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP3ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP3ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット1 **WUP2SEC**：WUP2セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP2ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP2ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

ビット0 **WUP1SEC**：WUP1セキュア保護

0：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP1ピンに関連するビットは、セキュアまたは非セキュアアクセスによる読み書きができます。

1：PWR_WUCR1、PWR_WUCR2、PWR_WUCR3、およびPWR_WUSCRのWKUP1ピンに関連するビットは、セキュアアクセスによってのみ読み書きができます。

10.10.14 PWR特権制御レジスタ（PWR_PRIVCFGR）

アドレスオフセット：0x34

リセット値：0x0000 0000

このレジスタは、アクセスが特権であるときのみ、書き込むことができます。特権または非特権アクセスによって読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **NSPRIV**：PWR非セキュア機能の特権設定

このビットは、ソフトウェアによってセット/リセットされます。セキュアまたは非セキュアの特権アクセスによってのみ書込み可能です。

0：PWR非セキュア機能の読み書きは、特権または非特権アクセスによって行うことができます。

1：PWR非セキュア機能の読み書きは、特権アクセスによってのみ行うことができます。

ビット0 **SPRIV**：PWRセキュア機能の特権設定

このビットは、ソフトウェアによってセット/リセットされます。セキュア特権アクセスによってのみ書込み可能です。

0：PWRセキュア機能の読み書きは、特権または非特権アクセスによって行うことができます。

1：PWRセキュア機能の読み書きは、特権アクセスによってのみ行うことができます。

10.10.15 PWRステータスレジスタ（PWR_SR）

アドレスオフセット：0x38

リセット値：0x0000 0000

一部のレジスタ･フィールドは、PWR_SECCFGRによって非セキュアアクセスから保護されます。一部のレジスタ･フィールドは、PWR_PRIVCFGRによって非特権アクセスから保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SBF	STOPF	CSSF
													r	r	w

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SBF**：STANDBYフラグ

このビットは、デバイスがSTANDBYモードに移行したときにハードウェアによってセットされ、CSSFビットに1を書き込むか、パワーオンリセットによってクリアされます。システムリセットによってクリアされません。

0：デバイスがSTANDBYモードに移行しませんでした。

1：デバイスがSTANDBYモードに移行しました。

ビット1 **STOPF**：STOPフラグ

このビットは、デバイスがSTOPモードに移行したときにハードウェアによってセットされ、CSSFビットに1を書き込むことでソフトウェアによってクリアされます。

0：デバイスがSTOPモードに移行しませんでした。

1：デバイスがSTOPモードに移行しました。

ビット0 **CSSF**：STOPおよびSTANDBYフラグのクリア

このビットは、PWR_SECCFGRのLPMSEC = 1のとき、非セキュアアクセスから保護されます。このビットは、PWR_PRIVCFGRのLPMSEC = 1かつSPRIV = 1、またはLPMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

このビットに1を書き込むと、STOPFおよびSBFフラグがクリアされます。

10.10.16 PWR電源電圧モニタのステータスレジスタ（PWR_SVMSR）

アドレスオフセット：0x3C

リセット値：0x0000 8000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	VDDA2RDY	VDDA1RDY	VDDIO2RDY	VDDUSBRDY	Res.	Res.	Res.	Res.	Res.	Res.	ACTVOS[1:0]	
				r	r	r	r							r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACTVOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDO	Res.	Res.	REGS	Res.
r											r			r	

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **VDDA2RDY**：V_{DDA}レディに対する1.8V電圧モニタ

0：V_{DDA}はV_{DDA}電圧モニタ2の閾値（約1.8V）を下回ります。

1：V_{DDA}はV_{DDA}電圧モニタ2の閾値（約1.8V）以上です。

ビット26 **VDDA1RDY**：V_{DDA}レディに対する1.6V電圧モニタ

0：V_{DDA}はV_{DDA}電圧モニタ1の閾値（約1.6V）を下回ります。

1：V_{DDA}はV_{DDA}電圧モニタ1の閾値（約1.6V）以上です。

ビット25 **VDDIO2RDY**：V_{DDIO2}

0：V_{DDIO2}はV_{DDIO2}電圧モニタの閾値を下回ります。

1：V_{DDIO2}はV_{DDIO2}電圧モニタの閾値以上です。

ビット24 **VDDUSBRDY**：V_{DDUSB}

0：V_{DDUSB}はV_{DDUSB}電圧モニタの閾値を下回ります。

1：V_{DDUSB}はV_{DDUSB}電圧モニタの閾値以上です。

ビット23:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **ACTVOS[1:0]**：V_{CORE}に現在適用されているVOS（最後のVOS値）

00：レンジ4（最小電力）

01：レンジ3

10：レンジ2

11：レンジ1（最大周波数）

ビット15 **ACTVOSRDY**：現在使用されているVOSの電圧レベルが動作可能状態です。

0：V_{CORE}は、ACTVOS[1:0]によって提供される現在の電圧スケーリングを上回るか下回ります。

1：V_{CORE}は、ACTVOS[1:0]によって提供される現在の電圧スケーリングと等しくなります。

ビット14:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **PVDO**：プログラム可能な電圧検出器の出力

0：V_{DD}は、PVDLS[2:0]によって選択されたPVD閾値以上です。

1：V_{DD}は、PVDLS[2:0]によって選択されたPVD閾値を下回ります。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **REGS**：レギュレータ選択

0：LDOの選択

1：SMPSの選択

ビット0 予約済みであり、リセット値に保持する必要があります。

10.10.17 PWRバックアップドメインステータスレジスタ（PWR_BDSR）

アドレスオフセット：0x40

バックアップドメインリセット値：0x0000 0000

パワーオンリセット：影響なし

（STANDBYモードの終了の影響を受けないリセット値）

システムリセット：影響なし

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	VBATH	Res.
												r	r	r	

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **TEMPH**：温度レベルの監視と高閾値

0：温度 < 高閾値

1：温度 ≥ 高閾値

ビット2 **TEMPL**：温度レベルの監視と低閾値

0：温度 > 低閾値

1：温度 ≤ 低閾値

ビット1 **VBATH**：バックアップドメイン電圧レベル監視と高閾値

0：バックアップドメイン電圧レベル < 高閾値

1：バックアップドメイン電圧レベル ≥ 高閾値

ビット0 予約済みであり、リセット値に保持する必要があります。

10.10.18 PWRウェイクアップステータスレジスタ（PWR_WUSR）

アドレスオフセット：0x44

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUF8	WUF7	WUF6	WUF5	WUF4	WUF3	WUF2	WUF1
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WUF8**：ウェイクアップ･フラグ8

このビットは、WKUP8ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、WUSEL8 ≠ 11のときにPWR_WUSCRのCWUF8ビットに1を書き込むか、WUPEN8 = 0のときにハードウェアによってクリアされます。

WUSEL8 = 11の場合は、すべての内部ウェイクアップソースがクリアされると、このビットはハードウェアによってクリアされます。

ビット6 **WUF7**：ウェイクアップ･フラグ7

このビットは、WKUP7ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、WUSEL7 ≠ 11のときにPWR_WUSCRのCWUF7ビットに1を書き込むか、WUPEN7 = 0のときにハードウェアによってクリアされます。

WUSEL7 = 11の場合は、すべての内部ウェイクアップソースがクリアされると、このビットはハードウェアによってクリアされます。

ビット5 **WUF6**：ウェイクアップ･フラグ6

このビットは、WKUP6ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、WUSEL6 ≠ 11のときにPWR_WUSCRのCWUF6ビットに1を書き込むか、WUPEN6 = 0のときにハードウェアによってクリアされます。

WUSEL6 = 11の場合は、すべての内部ウェイクアップソースがクリアされると、このビットはハードウェアによってクリアされます。

ビット4 **WUF5**：ウェイクアップ･フラグ5

このビットは、WKUP5ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、PWR_WUSCRのCWUF5ビットに1を書き込むか、WUPEN5 = 0のときにハードウェアによってクリアされます。

ビット3 **WUF4**：ウェイクアップ･フラグ4

このビットは、WKUP4ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、PWR_WUSCRのCWUF4ビットに1を書き込むか、WUPEN4 = 0のときにハードウェアによってクリアされます。

ビット2 **WUF3**：ウェイクアップ･フラグ3

このビットは、WKUP3ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、PWR_WUSCRのCWUF3ビットに1を書き込むか、WUPEN3 = 0のときにハードウェアによってクリアされます。

ビット1 **WUF2**：ウェイクアップ･フラグ2

このビットは、WKUP2ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、PWR_WUSCRのCWUF2ビットに1を書き込むか、WUPEN2 = 0のときにハードウェアによってクリアされます。

ビット0 **WUF1**：ウェイクアップ･フラグ1

このビットは、WKUP1ピンでウェイクアップイベントが検出されたときにセットされます。このビットは、PWR_WUSCRのCWUF1ビットに1を書き込むか、WUPEN1 = 0のときにハードウェアによってクリアされます。

10.10.19 PWRウェイクアップステータスクリアレジスタ(PWR_WUSCR)

アドレスオフセット：0x48

リセット値：0x0000 0000

各CWUFx（x = 1 ~ 8）は、PWR_SECCFGRのWUPxSEC = 1のとき、非セキュアアクセスから保護されます。各CWUFxは、PWR_SECCFGRのWUPxSEC = 1かつPWR_PRIVCFGRのSPRIV = 1、またはWUPxSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CWUF8	CWUF7	CWUF6	CWUF5	CWUF4	CWUF3	CWUF2	CWUF1
								w	w	w	w	w	w	w	w

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **CWUF8**：ウェイクアップ･フラグ8
このビットに1を書き込むと、PWR_WUSRのWUF8フラグがクリアされます。

ビット6 **CWUF7**：ウェイクアップ･フラグ7
このビットに1を書き込むと、PWR_WUSRのWUF7フラグがクリアされます。

ビット5 **CWUF6**：ウェイクアップ･フラグ6
このビットに1を書き込むと、PWR_WUSRのWUF6フラグがクリアされます。

ビット4 **CWUF5**：ウェイクアップ･フラグ5
このビットに1を書き込むと、PWR_WUSRのWUF5フラグがクリアされます。

ビット3 **CWUF4**：ウェイクアップ･フラグ4
このビットに1を書き込むと、PWR_WUSRのWUF4フラグがクリアされます。

ビット2 **CWUF3**：ウェイクアップ･フラグ3
このビットに1を書き込むと、PWR_WUSRのWUF3フラグがクリアされます。

ビット1 **CWUF2**：ウェイクアップ･フラグ2
このビットに1を書き込むと、PWR_WUSRのWUF2フラグがクリアされます。

ビット0 **CWUF1**：ウェイクアップ･フラグ1
このビットに1を書き込むと、PWR_WUSRのWUF1フラグがクリアされます。

10.10.20 PWRで適用されるプル設定レジスタ（PWR_APCR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

このレジスタは、PWR_SECCFGRのAPCSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのAPCSEC = 1かつSPRIV = 1、またはAPCSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APC
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **APC**：プルアップおよびプルダウン設定の適用

1：PWR_PUCRxおよびPWR_PDCRxで定義されたI/Oプルアップおよびプルダウン設定が適用されます。

0：PWR_PUCRxおよびPWR_PDCRxはI/Oに適用されません。

10.10.21 PWRポートAプルアップ制御レジスタ（PWR_PUCRA）

アドレスオフセット：0x50

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOA_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOA_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **PU15**：ポートAプルアップビット15

このビットをセットすると、APCがPWR_APCRでセットされたときにPA15でプルアップが有効になります。対応するPD15ビットもセットされている場合、プルアップは有効になりません。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **PUy**：ポートAプルアップビットy（y = 13 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPAyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.22 PWRポートAプルダウン制御レジスタ（PWR_PDCRA）

アドレスオフセット：0x54

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOA_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOA_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **PD14**：ポートAプルダウンビット14

このビットをセットすると、APCビットがPWR_APCRでセットされたときにPA14でプルダウンが有効になります。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **PDy**：ポートAプルダウンビットy（y = 12 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPAyでプルダウンが有効になります。

10.10.23 PWRポートBプルアップ制御レジスタ（PWR_PUCRB）

アドレスオフセット：0x58

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOB_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOB_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートBプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPByでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.24 PWRポートBプルダウン制御レジスタ（PWR_PDCRB）

アドレスオフセット：0x5C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOB_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOB_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:5 **PDy**：ポートBプルダウンビットy（y = 15 ~ 5）

各ビットをセットすると、APCビットがPWR_APCRでセットされたときにPByでプルダウンが有効になります。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PDy**：ポートBプルダウンビットy（y = 3 ~ 0）

各ビットをセットすると、APCビットがPWR_APCRでセットされたときにPByでプルダウンが有効になります。

10.10.25 PWRポートCプルアップ制御レジスタ（PWR_PUCRC）

アドレスオフセット：0x60

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各レジスタビットPUyはGPIOC_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。

各レジスタビットPUyは、GPIOC_SECCFGRのSECy = 1およびPWR_PRIVCFGRのSPRIV = 1のとき、またはSECy = 0およびNSPRIV = 1.のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートCプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPCyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.26 PWRポートCプルダウン制御レジスタ（PWR_PDCRC）

アドレスオフセット：0x64

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOC_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOC_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートCプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPCyでプルダウンが有効になります。

10.10.27 PWRポートDプルアップ制御レジスタ（PWR_PUCRD）

アドレスオフセット：0x68

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOD_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOD_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートDプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPDyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.28 PWRポートDプルダウン制御レジスタ（PWR_PDCRD）

アドレスオフセット：0x6C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOD_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOD_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートDプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPDyでプルダウンが有効になります。

10.10.29 PWRポートEプルアップ制御レジスタ（PWR_PUCRE）

アドレスオフセット：0x70

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOE_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOE_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートEプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCビットがPWR_APCRでセットされたときにPEyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.30 PWRポートEプルダウン制御レジスタ（PWR_PDCRE）

アドレスオフセット：0x74

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOE_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOE_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートEプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPEyでプルダウンが有効になります。

10.10.31 PWRポートFプルアップ制御レジスタ（PWR_PUCRF）

アドレスオフセット：0x78

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOF_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOF_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートFプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPFyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.32 PWRポートFプルダウン制御レジスタ（PWR_PDCRF）

アドレスオフセット：0x7C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOF_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOF_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注：　一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16　予約済みであり、リセット値に保持する必要があります。

ビット15:0　**PDy**：ポートFプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPFyでプルダウンが有効になります。

10.10.33 PWRポートGプルアップ制御レジスタ（PWR_PUCRG）

アドレスオフセット：0x80

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOG_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOG_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16　予約済みであり、リセット値に保持する必要があります。

ビット15:0　**PUy**：ポートGプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPGyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.34 PWRポートGプルダウン制御レジスタ（PWR_PDCRG）

アドレスオフセット：0x84

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOG_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOG_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートGプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPGyでプルダウンが有効になります。

10.10.35 PWRポートHプルアップ制御レジスタ（PWR_PUCRH）

アドレスオフセット：0x88

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOH_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOH_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートHプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPHyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.36 PWRポートHプルダウン制御レジスタ（PWR_PDCRH）

アドレスオフセット：0x8C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOH_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOH_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートHプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPHyでプルダウンが有効になります。

10.10.37 PWRポートIプルアップ制御レジスタ（PWR_PUCRI）

アドレスオフセット：0x90

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOI_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOI_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PUy**：ポートIプルアップビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPIyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.38 PWRポートIプルダウン制御レジスタ（PWR_PDCRI）

アドレスオフセット：0x94

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOI_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOI_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PDy**：ポートIプルダウンビットy（y = 15 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPIyでプルダウンが有効になります。

10.10.39 PWRポートJプルアップ制御レジスタ（PWR_PUCRJ）

アドレスオフセット：0x98

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PUyは、GPIOJ_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PUyは、GPIOJ_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **PUy**：ポートJプルアップビットy（y = 11 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPJyでプルアップが有効になります。対応するPDyビットもセットされている場合、プルアップは有効になりません。

10.10.40 PWRポートJプルダウン制御レジスタ（PWR_PDCRJ）

アドレスオフセット：0x9C

リセット値：0x0000 0000

（STANDBYモードの終了の影響を受けないリセット値）

各PDyは、GPIOJ_SECCFGRのSECy = 1のとき、非セキュアアクセスから保護されます。各PDyは、GPIOJ_SECCFGRのSECy = 1かつPWR_PRIVCFGRのSPRIV = 1、またはSECy = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

アクセス：通常のAHBアクセスに対して14AHBクロックサイクルを追加

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **PDy**：ポートJプルダウンビットy（y = 11 ~ 0）

各ビットをセットすると、APCがPWR_APCRでセットされたときにPJyでプルダウンが有効になります。

10.10.41 PWR制御レジスタ4（PWR_CR4）

アドレスオフセット：0xA8

リセット値：0x0000 0000

このレジスタは、PWR_SECCFGRのLPMSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのLPMSEC = 1かつSPRIV = 1、またはLPMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SRAM5 PDS13	SRAM5 PDS12	SRAM5 PDS11	SRAM5 PDS10	SRAM5 PDS9	SRAM5 PDS8	SRAM5 PDS7	SRAM5 PDS6	SRAM5 PDS5	SRAM5 PDS4	SRAM5 PDS3	SRAM5 PDS2	SRAM5 PDS1
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SRAM3 PDS13	SRAM3 PDS12	SRAM3 PDS11	SRAM3 PDS10	SRAM3 PDS9	Res.	SRAM1 PDS12	SRAM1 PDS11	SRAM1 PDS10	SRAM1 PDS9	SRAM1 PDS8	SRAM1 PDS7	SRAM1 PDS6	SRAM1 PDS5	SRAM1 PDS4
	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:16 **SRAM5PDSx**：すべてのSTOPモードでのSRAM5 64KBページx（x = 13 ~ 1）のパワーダウン
0：STOPモードで、SRAM5ページxの内容を保持する
1：STOPモードで、SRAM5ページxの内容を失う

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:10 **SRAM3PDSx**：すべてのSTOPモードでのSRAM3 64KBページx（x = 13 ~ 9）のパワーダウン
0：STOPモードで、SRAM3ページxの内容を保持する
1：STOPモードで、SRAM3ページxの内容を失う

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **SRAM1PDSx**：すべてのSTOPモードでのSRAM1 64KBページx（x = 12 ~ 4）のパワーダウン
0：STOPモードで、SRAM1ページxの内容を保持する
1：STOPモードで、SRAM1ページxの内容を失う

10.10.42 PWR制御レジスタ5（PWR_CR5）

アドレスオフセット：0xAC

リセット値：0x0000 0000

このレジスタは、PWR_SECCFGRのLPMSEC = 1のとき、非セキュアアクセスから保護されます。このレジスタは、PWR_PRIVCFGRのLPMSEC = 1かつSPRIV = 1、またはLPMSEC = 0かつNSPRIV = 1のとき、非特権アクセスから保護されます。

注： **一部のビットは、STM32U5 シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルを使用できるかどうかについては、デバイスのデータシートを参照してください。ない場合、このビットは予約済みで、リセット値に保持されているものとみなします。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAM6 PDS8	SRAM6 PDS7	SRAM6 PDS6	SRAM6 PDS5	SRAM6 PDS4	SRAM6 PDS3	SRAM6 PDS2	SRAM6 PDS1
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SRAM6PDSx**：すべてのSTOPモードでのSRAM6 64KBページx（x = 8 ~ 1）のパワーダウン

0：STOPモードで、SRAM6ページxの内容を保持する

1：STOPモードで、SRAM6ページxの内容を失う

10.10.43 PWRレジスタマップ

表111.PWRレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	PWR_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FORCE_USBPWR	Res.	SRAM6PD	SRAM5PD	SRAM4PD	SRAM3PD	SRAM2PD	SRAM1PD	ULPMEN	RRSB2	RRSB1	Res.	Res.	LPMS [2:0]		
	リセット値																	0		0	0	0	0	0	0	0	0	0			0	0	0
0x04	PWR_CR2	SRDRUN	Res.	Res.	Res.	Res.	JPEGRAMPDS	DSIRAMPDS	GPRAMPDS	SRAM3PDS8	SRAM3PDS7	SRAM3PDS6	SRAM3PDS5	SRAM3PDS4	SRAM3PDS3	SRAM3PDS2	SRAM3PDS1	Res.	FLASHFWU	SRAM4FWU	PKARAMPDS	PRAMPDS	DMA2DRAMPDS	DC1RAMPDS	ICRAMPDS	DC2RAMPDS	SRAM4PDS	SRAM2PDS2	SRAM2PDS1	Res.	SRAM1PDS3	SRAM1PDS2	SRAM1PDS1
	リセット値	0					0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x08	PWR_CR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FSTEN	REGSEL	Res
	リセット値																														0	0	
0x0C	PWR_VOSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VDD11USBDIS	USBBOOSTEN	USBPWREN	BOOSTEN	VOS[1:0]		VOSRDY	BOOSTRDY	USBBOOSTRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	1	0	0													

表111.PWRレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x10	PWR_SVMCR	Res.	ASV	IO2SV	USV	AVM2EN	AVM1EN	IO2VMEN	UVMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDLS [2:0]			PVDE	Res.	Res.	Res.	Res.
	リセット値		0	0	0	0	0	0	0																	0	0	0	0				
0x14	PWR_WUCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPEN8	WUPEN7	WUPEN6	WUPEN5	WUPEN4	WUPEN3	WUPEN2	WUPEN1
	リセット値																									0	0	0	0	0	0	0	0
0x18	PWR_WUCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPP8	WUPP7	WUPP6	WUPP5	WUPP4	WUPP3	WUPP2	WUPP1
	リセット値																									0	0	0	0	0	0	0	0
0x1C	PWR_WUCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUSEL8 [1:0]		WUSEL7 [1:0]		WUSEL6 [1:0]		WUSEL5 [1:0]		WUSEL4 [1:0]		WUSEL3 [1:0]		WUSEL2 [1:0]		WUSEL1 [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	PWR_BDCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
	リセット値																												0				0
0x24	PWR_BDCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE
	リセット値																															0	0
0x28	PWR_DBPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBP
	リセット値																																0
0x2C	PWR_UCPDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD_STBY	UCPD_DBDIS
	リセット値																															0	0
0x30	PWR_SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APCSEC	VBSEC	VDMSEC	LPMSEC	Res.	Res.	Res.	Res.	WUP8SEC	WUP7SEC	WUP6SEC	WUP5SEC	WUP4SEC	WUP3SEC	WUP2SEC	WUP1SEC
	リセット値																	0	0	0	0					0	0	0	0	0	0	0	0
0x34	PWR_PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
	リセット値																															0	0
0x38	PWR_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SBF	STOPF	CSSF
	リセット値																														0	0	0
0x3C	PWR_SVMSR	Res.	Res.	Res.	Res.	VDDA2RDY	VDDA1RDY	VDDIO2RDY	VDDUSBRDY.	Res.	Res.	Res.	Res.	Res.	Res.	ACTVOS [1:0]		ACTVOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDO	Res.	Res.	REGS	Res.
	リセット値					0	0	0	0							0	0	1											0			0	
0x40	PWR_BDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	VBATH	Res.
	リセット値																													0	0	0	
0x44	PWR_WUSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUF8	WUF7	WUF6	WUF5	WUF4	WUF3	WUF2	WUF1
	リセット値																									0	0	0	0	0	0	0	0
0x48	PWR_WUSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CWUF8	CWUF7	CWUF6	CWUF5	CWUF4	CWUF3	CWUF2	CWUF1
	リセット値																									0	0	0	0	0	0	0	0

表111.PWRレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4C	PWR_APCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APC
	リセット値																																0
0x50	PWR_PUCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	PWR_PDCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																		0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x58	PWR_PUCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	PWR_PDCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0
0x60	PWR_PUCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	PWR_PDCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	PWR_PUCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	PWR_PDCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	PWR_PUCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	PWR_PDCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	PWR_PUCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	PWR_PDCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x80	PWR_PUCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	PWR_PDCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	PWR_PUCRH	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	PWR_PDCRH	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	PWR_PUCRI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94	PWR_PDCRI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x98	PWR_PUCRJ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0

表111.PWRレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x9C	PWR_PDCRJ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0xA0 ~0xA4	予約済み	予約済み																															
0xA8	PWR_CR4	Res.	Res.	Res.	SRAM5PDS13	SRAM5PDS12	SRAM5PDS11	SRAM5PDS10	SRAM5PDS9	SRAM5PDS8	SRAM5PDS7	SRAM5PDS6	SRAM5PDS5	SRAM5PDS4	SRAM5PDS3	SRAM5PDS2	SRAM5PDS1	Res.	SRAM3PDS13	SRAM3PDS12	SRAM3PDS11	SRAM3PDS10	SRAM3PDS9	Res.	SRAM1PDS12	SRAM1PDS11	SRAM1PDS10	SRAM1PDS9	SRAM1PDS8	SRAM1PDS7	SRAM1PDS6	SRAM1PDS5	SRAM1PDS4
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0
0xAC	PWR_CR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAM6PDS8	SRAM6PDS7	SRAM6PDS6	SRAM6PDS5	SRAM6PDS4	SRAM6PDS3	SRAM6PDS2	SRAM6PDS1
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

11 リセットおよびクロック制御（RCC）

11.1 RCCの概要

リセットおよびクロック制御（RCC）では、各種のリセットを管理し、バスおよびペリフェラル用のすべてのクロックを生成します。

11.2 RCCピンおよび内部信号

下の表に、パッケージのピンまたはボールに接続されたRCC入出力信号を示します。

表112.パッケージのピンまたはボールに接続されたRCCスレーブ入力／出力信号

信号名	信号タイプ	説明
NRST	I/O	システムリセット、外部デバイスにリセットを提供するために使用可能
OSC32_IN	I	32kHzオシレータ入力
OSC32_OUT	O	32kHzオシレータ出力
OSC_IN	I	システムオシレータ入力
OSC_OUT	O	システムオシレータ出力
MCO	O	外部デバイスの出力クロック
LSCO	O	外部デバイスの低速出力クロック
AUDIOCLK	I	SAI1、SAI2、MDF1、およびADF1の外部カーネルクロック入力

11.3 RCCリセット機能の説明

リセットには次の3種類があります。

- システムリセット
- 電源リセット
- バックアップドメインリセット

11.3.1 電源リセット

電源リセットは、次のいずれかのイベントが発生したときに生成されます。

- ブラウンアウトリセット（BOR）
- STANDBYモードの終了時
- SHUTDOWNモードを終了するとき

BORは、バックアップドメインを除くすべてのレジスタをリセット値に設定します。

STANDBYモードを終了すると、コアドメインのすべてのレジスタがリセット値にセットされます。コアドメイン以外のレジスタ（RTC、TAMP、WKUP、IWDGおよびSTANDBY/SHUTDOWNモード制御）は影響を受けません。

SHUTDOWNモードを終了すると、ブラウンアウトリセットが生成され、バックアップドメイン以外のすべてのレジスタがリセットされます。

11.3.2 システムリセット

システムリセットは、RCC_CSRのリセットフラグとバックアップドメインのレジスタを除き、すべてのレジスタをリセット値にセットします。

システムリセットは、次のイベントのいずれかの発生時に生成されます。

- NRSTピンのローレベル（外部リセット）
- ウィンドウ型ウォッチドッグイベント（WWDGリセット）
- 独立型ウォッチドッグイベント（IWDGリセット）
- ソフトウェアリセット（SWリセット）（ソフトウェアリセットを参照）
- 低消費電力モードセキュリティリセット（低消費電力モードセキュリティリセットを参照）
- オプションバイトローダリセット（オプションバイトローダリセットを参照）
- ブラウンアウトリセット

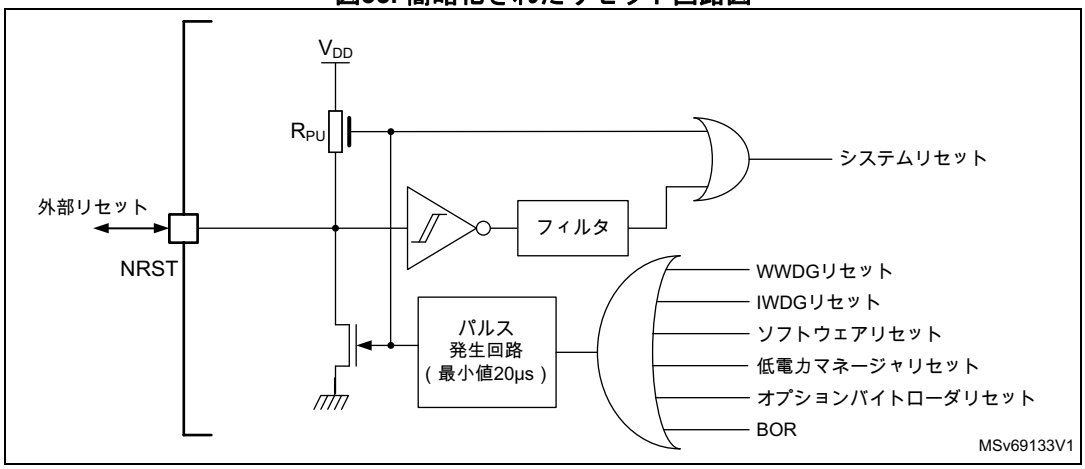
リセットソースは、RCC_CSRのリセットフラグを確認することによって識別できます。

これらのソースはNRSTピンに作用し、遅延フェーズの間中このピンをローに保ちます。リセットサービスルーチンベクタは、ブートオプションバイトを介して選択されます。

デバイスに与えられるシステムリセット信号は、NRSTピンに出力されます。パルス発生回路は、各内部リセットソースについて20µsの最小リセットパルス期間を保証します。外部リセットの場合、リセットパルスはNRSTピンがローにアサートされているときに生成されます。

内部リセットの場合、プルアップレジスタで消費電力を削減するために、内部プルアップR_{PU}が無効化されます。

図35. 簡略化されたリセット回路図



ソフトウェアリセット

デバイス上でソフトウェアリセットを実行するには、Cortex-M33のアプリケーション割込みおよびリセット制御レジスタのSYSRESETREQビットをセットする必要があります。

低消費電力モードセキュリティリセット

誤って重要なアプリケーションが低電力モードに移行しないように、以下の低電力モードセキュリティリセットが使用できます。オプションバイトで有効になっている場合、以下のいずれかの条件下でリセットを生成します。

- STANDBYモードへの移行：このリセットは、ユーザオプションバイトのNRST_STDBYビットをリセットすることによって有効になります。この場合、STANDBYモードへの遷移シーケンスが正常に実行されるたびに、STANDBYモードに移行する代わりにデバイスがリセットされます。
- STOP モードへの移行：このリセットは、ユーザオプションバイトの NRST_STOP ビットをリセットすることによって有効になります。この場合、STOPモードへの遷移シーケンスが正常に実行されるたびに、STOPモードに移行する代わりにデバイスがリセットされます。
- SHUTDOWNモードへの移行：このリセットは、ユーザオプションバイトのNRST_SHDWビットをリセットすることによって有効になります。この場合、SHUTDOWNモードへの遷移シーケンスが正常に実行されるたびに、SHUTDOWNモードに移行する代わりにデバイスがリセットされます。

ユーザオプションバイトの詳細については、セクション 7.4.1：オプションバイトの説明を参照してください。

オプションバイトローダリセット

FLASH_NSCRレジスタのOBL_LAUNCHビットがセットされると、オプションバイトローダリセットが生成されます。このビットは、ソフトウェアによるオプション･バイト･ローディングを起動するために使用されます。

11.3.3 バックアップドメインリセット

バックアップドメインには2つの固有のリセットがあります。

バックアップドメインのリセットは、次のいずれかのイベントが発生したときに生成されます。

- RCC_BDCRのBDRSTビットをセットすることでトリガされるソフトウェアリセット。
- V_{DD}またはV_{BAT}パワーオン。ただし、両方の電源供給がともにオフ状態であった場合。

バックアップドメインリセットは、LSE　オシレータ、RTC、TAMP、バックアップレジスタ、RCC_BDCR、およびPWR_BDCR1に影響します。PWR_BDCR1のリセットは、バックアップSRAMに影響します。

11.4 RCCクロックの機能説明

次の4つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます：

- HSI16：ハイスピード内部16MHzRCオシレータクロック
- MSIS：マルチスピード内部RCオシレータクロック
- HSE：高速外部クリスタルまたはクロック（4 ~ 50mHz）
- PLL1クロック

MSISは、4MHzで設定され、リセットからの起動後、システムクロックソースとして使用されます。

デバイスには、次の追加のクロックソースがあります。

- MSIK: ペリフェラルのカーネルクロックに使用されるマルチスピード内部RCオシレータクロック
- LSI : 32kHz/250Hzロースピード内部RC。独立型ウォッチドッグを駆動し、オプションで、STOPモードからSTANDBYモードへの自動ウェイクアップに使用されるRTCを駆動します。
- LSE：32.768kHzのロースピード外部クリスタルまたはクロック。オプションで、リアルタイムクロック（rtc_ck）を駆動します。
- HSI48：内部48MHzRCで、OTG_FS、USB、SDMMC、およびRNGを駆動できます。
- SHSI：セキュアなハイスピード内部48MHzRCで、SAESを駆動します。
- PLL2およびPLL3クロック

それぞれのクロックソースは、使用しないときに個別にオン／オフを切り替えて、電力消費を最適化可能です。

複数のプリスケーラを使用してAHB周波数、APB1およびAPB2ドメインを設定できます。AHBおよびAPBドメインの最大周波数は、160MHzです。

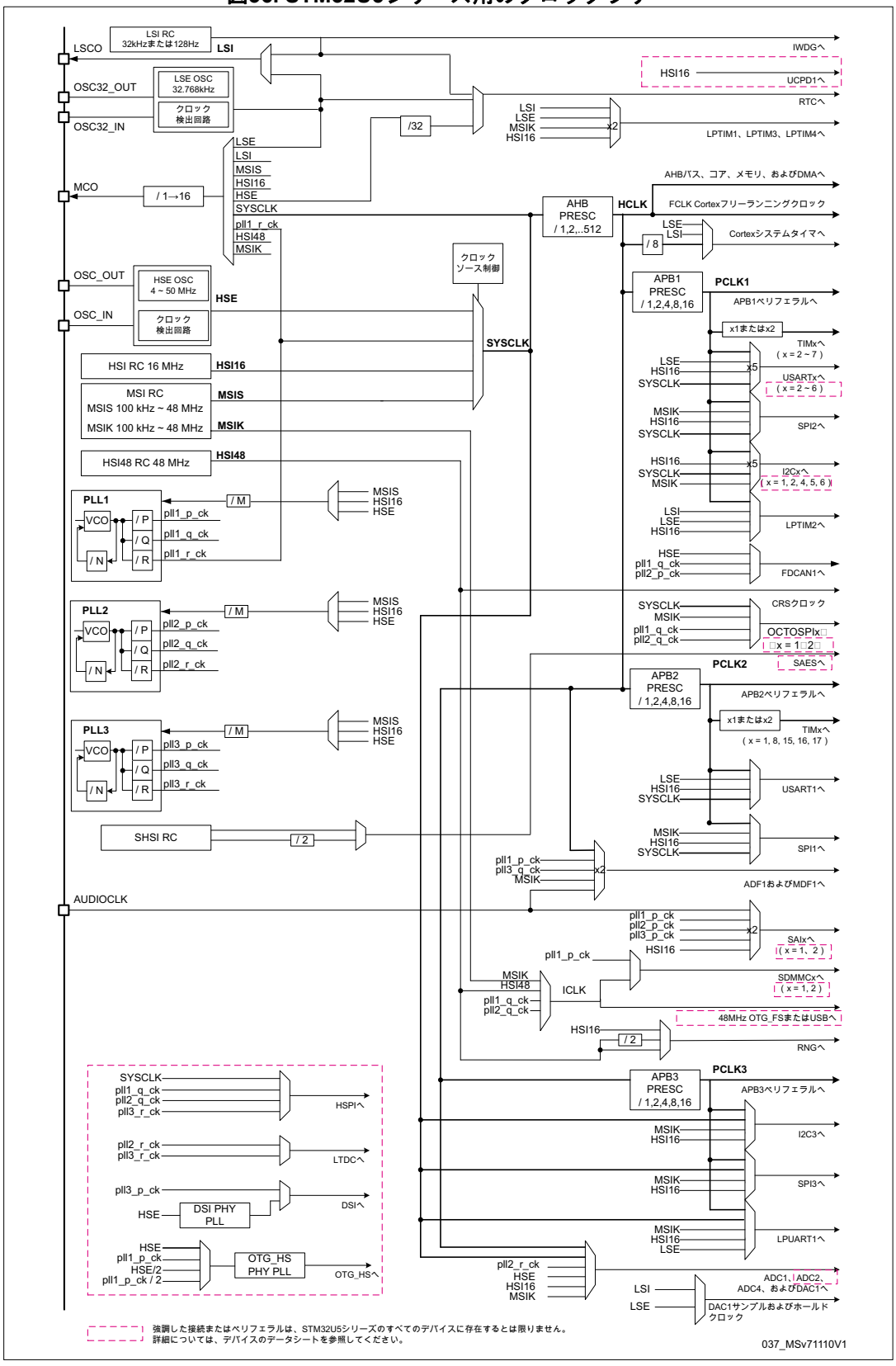
すべてのペリフェラルクロックは、バスクロック（HCLK、PCLK1、PCLK2、またはPCLK3）から生成されますが、以下のものは、独立したカーネルクロックを受信します。このカーネルクロックは、RCC_CCIPRxレジスタ（x = 1,2,3）を使用して、いくつかのソースの間でソフトウェアによって選択できます。例えば、OTG_FS、USB、またはOTG_HS、SDMMCx（x = 1,2）、RNG、ADCx（x = 1,2,4）、DAC1、U(S)ARTx（x = 1~6）、LPUART1、I2Cx（x = 1~6）、SPIx（x = 1~3）、OCTOSPIx（x = 1,2）、SAIx（x = 1,2）、MDF1、ADF1、FDCAN1、LPTIMx（x = 1~4）、SAES、DSI、LTDC、HSPI1が選択できます。

また、RTCカーネルクロックはソフトウェアにより、RCC_BDCRで選択されます。IWDGクロックは常にLSI 32kHzクロックとなります。

RCCはCortexシステムタイマ（SysTick）の外部クロックに、8分周したAHBクロック（HCLK）、あるいはLSEまたはLSIを供給します。SysTickは、この分周されたクロックか直接Cortexクロック（HCLK）で動作でき、SysTick制御およびステータスレジスタで設定可能です。

FCLKは、Cortex-M33のフリーランニングクロックとして動作します。

図36. STM32U5シリーズ用のクロックツリー



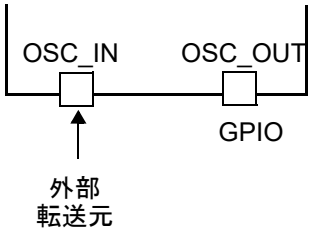
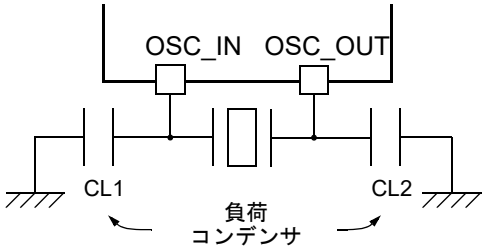
11.4.1 HSEクロック

ハイスピード外部クロック信号（HSE）は、次のどちらかのクロックソースから生成できます。

- HSE外部クリスタル／セラミック発振子
- HSEユーザ外部クロック

波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したオシレータに応じて調整する必要があります。

図37. HSE/LSEクロックソース

クロックソース	ハードウェア構成
外部クロック	
クリスタル／ セラミック発振子	

外部クリスタル／セラミック発振子（HSEクリスタル）

4から50MHzの外部オシレータには、メインクロックの周波数を非常に高い精度で生成できるという利点があります。

関連するハードウェア構成を図37に示します。詳細については、データシートの電気的特性のセクションを参照してください。

RCC_CRのHSERDYは、HSEオシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。RCC_CIERで有効になっていれば、割込みを生成することができます。

HSEクリスタルは、RCC_CRのHSEONを使用してオン／オフできます。

外部ソース（HSEバイパス）

このモードでは、外部クロックソースが必要です。最大50MHzまでの周波数を使用できます。このモードを選択するには、RCC_CRでHSEBYPおよびHSEONをセットします。周波数（データシートを参照）に応じて40 ~ 60%のデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）でOSC_INピンを駆動する必要があり、その間、OSC_OUTピンはGPIOとして使用できます（図37を参照）。

11.4.2 HSI16クロック

HSI16クロック信号は、内部16MHzRCオシレータから生成されます。

HSI16RCオシレータには、低コスト（外付部品なし）でクロックソースを供給できるという利点があります。同時に、HSEクリスタルオシレータよりも起動時間を短縮できまが、較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

STOPモードからのウェイクアップ後に、HSI16クロックをシステムクロックとして選択できます。セクション11.8.6を参照してください。これは、HSEクリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。セクション11.4.11を参照してください。

較正

RCオシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、T_A = 25°Cで1%の精度を確保するよう、STマイクロエレクトロニクス（ST）は製造時にデバイスごとの較正を行っています。

リセット後、工場較正値がRCC_ICSCR3のHSICAL[7:0]にロードされます。

アプリケーションが電圧または温度の変動を受ける場合、これがRCオシレータの速度に影響を与えることがあります。RCC_ICSCR3のHSITRIM[6:0]を使用して、アプリケーションでHSI16周波数をトリミングできます。

HSI16の周波数変動の測定方法の詳細については、セクション11.4.23を参照してください。

RCC_CRのHSIRDYは、HSI16RCオシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI16RC出力クロックはリリースされません。

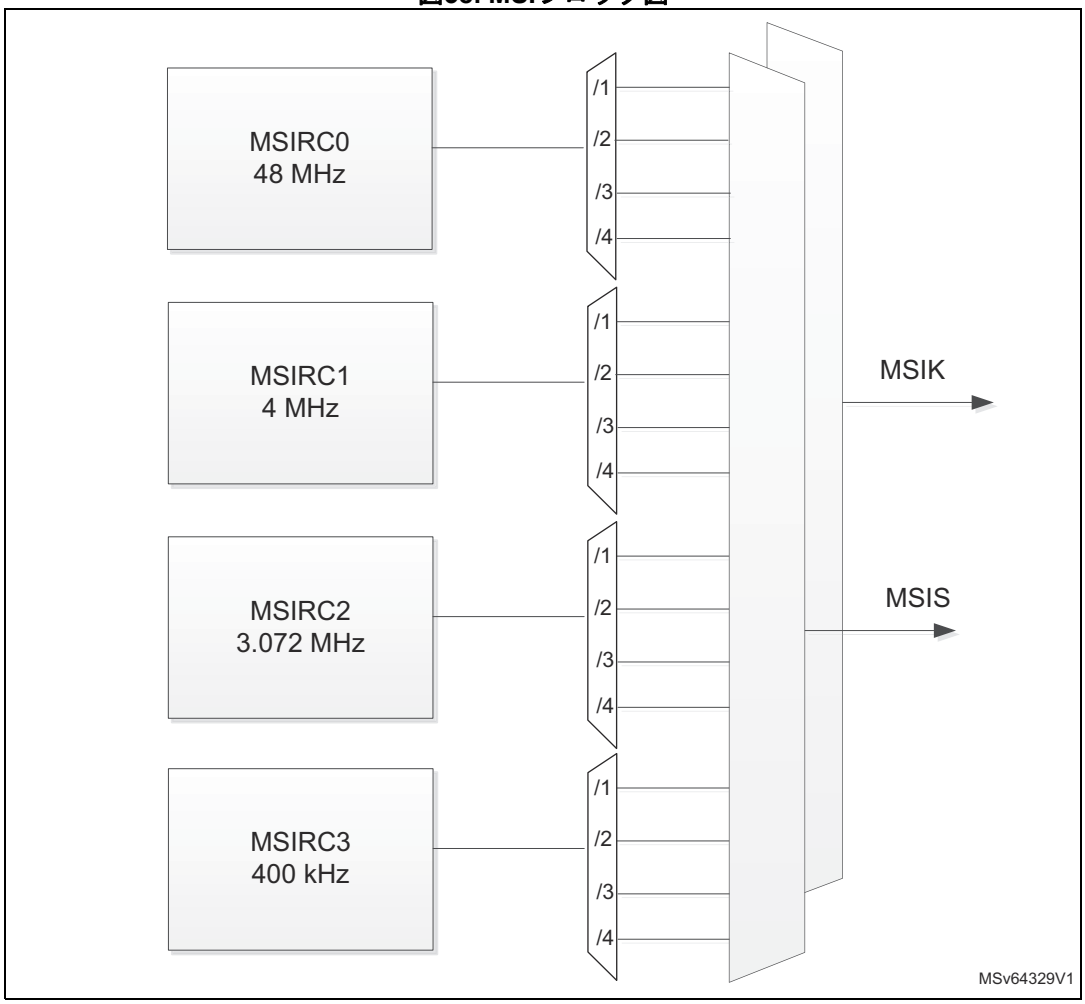
HSI16RCはRCC_CRのHSIONを使用してオン／オフの切り替えができます。

HSI16の信号は、HSEクリスタルオシレータに障害が起きた場合のバックアップソース（補助クロック）としても使用できます。セクション11.4.11を参照してください。

11.4.3 MSI（MSISおよびMSIK）クロック

MSIは4つの内部RCオシレータ、つまり48MHzのMSIRC0、4MHzのMSIRC1、3.072MHzのMSIRC2、および400kHzのMSIRC3で構成されます。各オシレータのプリスケーラは、分周比1、2、3、または4に対応しています。これらの分周されたオシレータから2つの出力クロックが生成されます。MSISはシステムクロックとして選択でき、MSIKは一部のペリフェラルによってカーネルクロックとして選択できます。

図38. MSIブロック図



MSISおよびMSIKの周波数範囲は、MSIRGSEL=1として、それぞれRCC_ICSCR1のMSISRANGE[3:0]およびMSIKRANGE[3:0]を使用してソフトウェアによって調整できます。16の周波数範囲が利用でき、以下の表に示すように、4つの内部RCから生成されます。

表113.内部MSIRCごとのMSISおよびMSIKレンジ

MSIRC0	MSIRC1	MSIRC2	MSIRC3
レンジ0：48MHz	レンジ4：4MHz	レンジ8：3.072MHz	レンジ12：400kHz
レンジ1：24MHz	レンジ5：2MHz	レンジ9：1.536MHz	レンジ13：200kHz
レンジ2：16MHz	レンジ6：1.33MHz	レンジ10：1.024MHz	レンジ14：133kHz
レンジ3：12MHz	レンジ7：1MHz	レンジ11：0.768MHz	レンジ15：100kHz

MSISクロックは、リセットからのリスタート後、およびSTANDBYおよびSHUTDOWN低電力モードからのウェイクアップ後に、システムクロックとして使用されます。リセットからのリスタート後、またはSHUTDOWNモードの終了時に、MSISおよびMSIKの周波数はデフォルト値の4MHzにセットされます。STANDBYモードからのウェイクアップ時の周波数範囲は、MSIRGSEL = 0として、それぞれMSISSRANGE[3:0]およびMSIKSRANGE[3:0]を使用してソフトウェアによって調整できます（RCC_CSRを参照）。

MSISクロックは、RCC_CRのSTOPWUCKに応じて、STOPモード（STOP0、STOP1、STOP2、またはSTOP3）からのウェイクアップ後にシステムクロックとして選択できます。これは、HSEクリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。セクション11.4.11を参照してください。

MSIの利点は、低コスト（外付部品なし）で低電力クロックソースを供給できることです。さらに、PLLモードでLSEと使用する場合、MSIは非常に高精度のクロックソースを提供します。このクロックソースは、OTG_FSまたはUSBで使用可能で、PLLに供給できます。

RCC_CRのMSISRDYおよびMSIKRDYは、MSISとMSIKRCが安定しているかどうかを示します。起動時、MSISおよびMSIKRC出力クロックは、それぞれのビットがハードウェアによってセットされるまで、出力されません。MSISおよびMSIKRCはRCC_CRのMSISONおよびMSIKONを使用してオン／オフの切り替えができます。

LSEによるハードウェア自動較正（PLLモード）

32.768kHzの外部オシレータがアプリケーションに存在する場合、PLLモードでMSISまたはMSIKを設定できます。このモードを以下のために有効化できます。

- MSIS用に有効化するには、RCC_CRでMSIPLLSEL = 1としてMSIPLLENをセットします。
- MSIK用に有効化するには、MSIPLLSEL = 0としてMSIPLLENをセットします。

MSISおよびMSIKレンジが同じMSIRCソースから生成される場合、PLLモードはMSISとMSIKの両方に適用されます。PLLモード設定時、MSISまたはMSIKはLSEによって自ら自動較正を行います。このモードは、すべてのMSI周波数範囲に対して使用可能です。48MHzでは、PLLモードのMSIKをOTG_FSまたはUSBに使用でき、外部ハイスピードクリスタルは不要となります。

LSEクロックパルスが停止すると、MSIPLLモードは自動的にアンロックされるため、MSIの精度は低下します。STM32U575/585 rev.Xを除くすべてのSTM32U5デバイスで、MSIPLLモードのアンロックイベントはEXTIラインに接続されます。これは、STOP0、STOP1、またはSTOP2モードからのウェイクアップに対応するイベントまたは割込みを生成するために使用されます（表 118：割込みソースとその制御および表 187：EXTIラインの接続を参照）。

MSIPLLモードの安定化までの時間

MSIPLLEN = 1のとき、（MSISON=1またはMSIKON = 1を書き込むか、STOPモードでペリフェラルクロックリクエストに続いて）MSIを有効にした後に最終的な精度が得られるのは、MSIPLLFAST = 0の場合に、安定化までの時間t_{STAB}(MSI)の経過後です。この安定化までの時間は、LSEが有効化されている状態でも必要です。t_{STAB}(MSI)の値については、データシートを参照してください。

MSIPLLFAST = 1でMSIPLLEN = 1の場合、スイッチオフのリクエスト受信時（MSISON = 0またはMSIKON = 0を書き込むことにより）、MSIオシレータはオンのままとなります（STOPモードではペリフェラルはこのクロックをリクエストしないため）。この場合、MSIがオフになる前に安定化までの時間t_{STAB}(MSI)に達した場合、MSIが再度オンになるとMSIPLLモードの精度が維持されます。このモードはSTOPモードでの精度が求められる自律型ペリフェラルに使用でき、オシレータはディスエーブル時に電源がオンのままでゲートオフされるため、消費電力は増えます。

ソフトウェア較正

MSIRCx（x = 0～3）のオシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、周辺温度T_A = 25°Cで1%の精度を確保するよう、STマイクロエレクトロニクス（ST）は製造時にデバイスごとの較正を行っています。リセット後、工場較正値がRCC_ICSCR1のMSICAL[4:0]（x = 0~3）にロードされます。アプリケーションが電圧または温度の変動を受ける場合、これがRCオシレータの速度に影響を与えることがあります。RCC_ICSCRのHSITRIM[4:0]（x = 0~3）を使用して、アプリケーションでMSIRCx周波数をトリミングできます。

注： **較正値の適用後、最終的な精度に達するのは、安定化までの時間の経過後です。この安定化までの時間は、STANDBYまたはSHUTDOWNモードの終了のリセット後に必要となります。PLLモードから通常モードへの切り替え時にも必要となります。**

LSEによるハードウェアの自動較正は、ソフトウェア較正と一緒に行わないでください。

MSI周波数変動の測定方法の詳細については、セクション11.4.23を参照してください。

11.4.4 HSI48クロック

HSI48クロック信号は、内部48MHzRCオシレータから生成され、USB/OTG_FS、RNG、およびSDMMCにも直接使用できます。

内部48MHzRCオシレータを使用する主な目的は、特殊なクロックリカバリシステム（CRS）回路によってOTG_FSおよびUSBに高精度のクロックを提供することです。CRSは、USB SOF信号（STM32U535/545/575/585でのみ）、LSEまたは外部信号を使用して、オシレータの周波数を動作中に自動的かつ迅速に調整できます。これは、システムがSTOPまたはSTANDBYモードに移行した直後に無効化されます。CRSを使用しない場合、HSI48RCオシレータは製造工程でのばらつきの影響を受けるデフォルトの周波数で実行されます。

CRSペリフェラルの設定および使用方法の詳細については、セクション 12：クロックリカバリシステム（CRS）を参照してください。

RCC_CRレジスタのHSI48RDYフラグは、HSI48RCオシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 RCオシレータ出力クロックはリリースされません。

HSI48は、RCC_CRレジスタのHSI48ONビットを使用してオン/オフできます。

11.4.5 SHSIクロック

SHSIは、SAESにクロックを供給するためのセキュリティ保護可能な内部RCオシレータです。RCC_CRのSHSIRDYフラグは、SHSIRCが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、SHSI RC出力クロックは出力されません。

SHSIRCは、RCC_CRのSHSIRCを使用してオン／オフの切り替えができます。

11.4.6 PLL

RCCには、次の3つのPLLが搭載されています。

- 通常CPUおよび一部のペリフェラルにクロックを供給するために使用するメインPLL（PLL1）
- ペリフェラルのカーネルクロックを生成するために使用する2つの専用のPLL2およびPLL3

RCCに統合されたPLLは、完全に独立しています。これらのPLLには以下の機能があります。

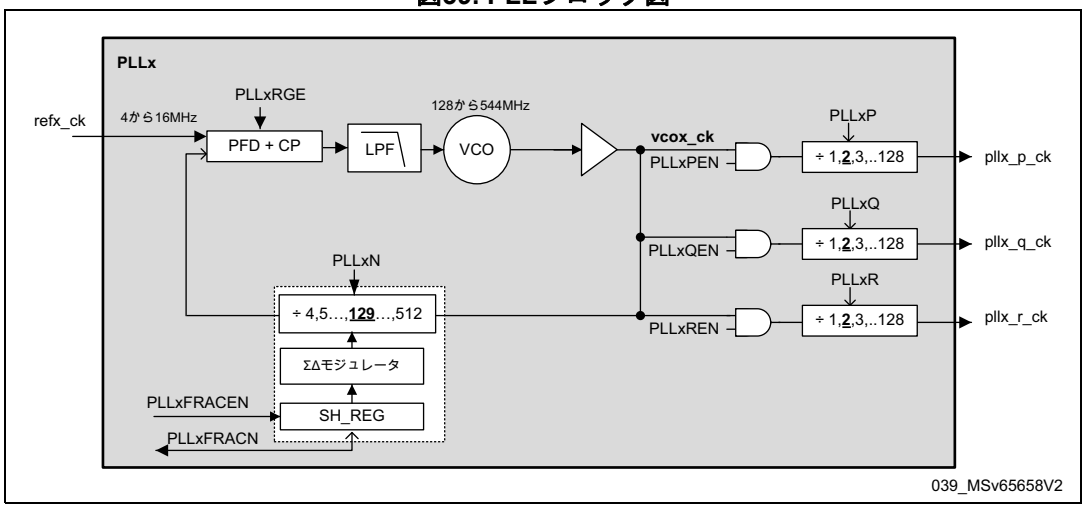
- 入力周波数範囲：4~16MHz
- 整数モードまたは分数モードで動作可能
- 11~0.3ppmのステップでVCO周波数を微調整可能な13bitのシグマデルタ（ΣΔ）モジュレータ
- ΣΔ モジュレータは、PLL 出力上で周波数のオーバーシュートを生成することなく、動作中に更新できます。
- 各PLLでは、ポスト分周器で3つの出力を使用できます。

PLLは、RCC_PLLxDIVR、RCC_PLLxFRACR、RCC_PLLxCFGR、およびRCC_CR（x = 1,2,3）によって制御されます。

PLL（refx_ck）に供給されるリファレンスクロックの周波数範囲は、4から16MHzの間でなければなりません。この条件に適合するように、ユーザーアプリケーションにより、RCC_PLL1CFGR、RCC_PLL2CFGR、およびRCC_PLL3CFGRのPLLxM（x = 1, 2, 3）分周器を適切にプログラムする必要があります。また、PLLが最適性能で動作するように、リファレンス入力周波数に合わせてPLLxRGEをセットする必要があります。

消費電力を低減するには、VCOx出力を最低周波数に設定することをお勧めします。

図39. PLLブロック図



VCO出力で期待される周波数を得るには、PLLxNループ分周器をプログラムする必要があります。また、VCO出力範囲を考慮する必要があります。

PLLxFRACENが0の場合、PLLxは整数モードで動作し、PLLはPLLxONにより有効化されます。分数モードは、PLL1FRACNを必要な値にセットし、PLLxFRACENを0から1に変更することにより、いつでも有効化できます。ΣΔモジュレータは、ごく小さなステップでの周波数の調整を可能にしながら、ジッタの影響を最小化するように設計されています。分数値を更新するには、PLLxFRACENを0にセットしてPLLxFRACN値を更新してから、PLLxFRACENを1にセットします。更新前のPLLxFRACN値は、PLLxFRACENを0から1に変更して新しい値が有効化されるまで使用されます。PLLxFRACNは、PLLxFRACENが0の場合にソフトウェアによってのみ更新される必要があります。

PLLsを有効化するには、RCC_CRでPLLxON = 1をセットします。PLLxRDYビットは、PLLがレディ状態（ロック状態）であることを示します。

注： **PLLを有効にする前に、PLLに供給されるリファレンス周波数（refx_ck）が安定していることを確認してください。PLLxが有効になった状態では、PLLxパラメータPLLxSRC、PLLxN、PLLxRGE、PLLxP、PLLxQ、およびPLLxRは変更できません。**

ハードウェアは、PLL1を使用してシステムクロックを供給している場合は、PLL1ONへの0の書込みを阻止します。

PLLが有効になった状態では、PLLパラメータPLLxN、PLLxRGE、PLLxP、PLLxQ、およびPLLxRは変更できません。

いずれかのポスト分周器（PLLxP、PLLxQ、またはPLLxR）を使用しない場合にPLLの最適な動作が保証されるように、アプリケーションにより、イネーブルビット（PLLxPEN、PLLxQEN、PLLxREN）をクリアし、対応するポスト分周器を最小値（PLLxR = 0、PLLxP = 0、またはPLLxQ = 0）に設定する必要があります。

上記のルールに従わない場合、PLL出力周波数の動作は保証されません。

出力周波数計算

PLLが整数モード（SH_REG = 0）で動作している場合、VCO周波数（F_{VCO}）は以下の式で求めることができます（x = 1,2,3）。

$$F_{VCOx} = F_{refx_ck} \times PLLxN$$

PLLが分数モード（SH_REG≠0）で動作している場合、PLLを有効にする前にPLLxN分周器を初期化する必要があります。ただし、PLL出力を損なうことなくPLLxFRACNの値を動作中に変更することができます。

この機能は、精度の良い任意のクリスタル値から特定の周波数を生成するため、または動作中に周波数を微調整するために使用できます。

各PLLのVCO周波数は、次の式で与えられます。

$$F_{VCOx} = F_{refx_ck} \times \left(PLLxN + \frac{PLLxFRACN}{2^{(13)}} \right)$$

整数モードと分数モードの両方で、以下の式でPLL1出力周波数を求めることができます。

$$F_{pllx_y_ck} = (F_{VCOx} / (PLLxy + 1)) \text{ with y = P, Q or R}$$

PLLは、以下の場合にハードウェアによって無効にされます。

- システムがSTOPまたはSTANDBYモードに移行したとき
- HSEまたはHSEでクロックされたPLLをシステムクロックとして使用して、HSEで障害が発生したとき

PLLを無効にすると、PLLによって使用された分数情報はリセットされます。

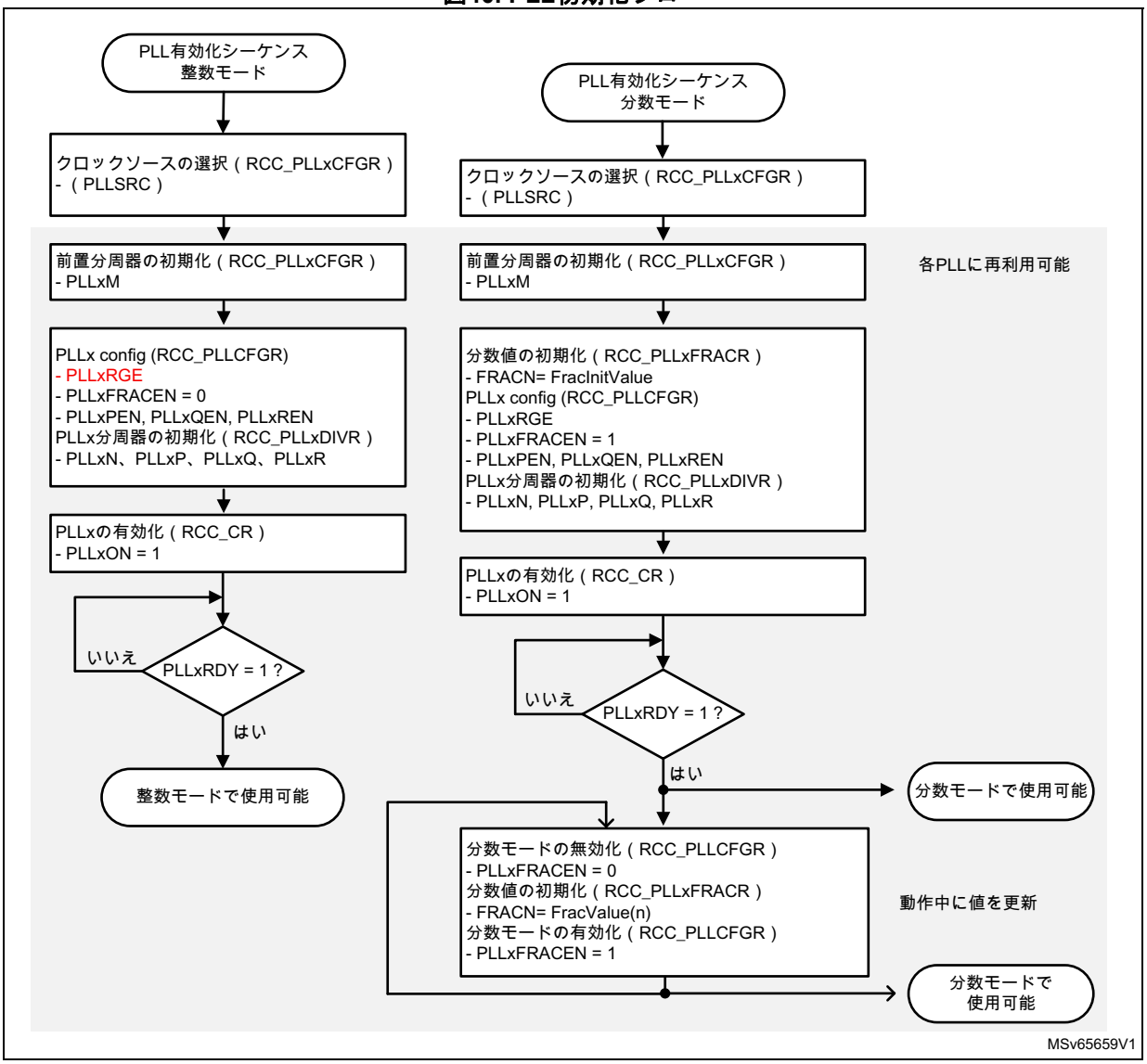
PLL初期化フェーズ

整数モードと分数モードでは、次のPLL初期化シーケンスを推奨します。PLLxは、初期化シーケンスの始めに無効にします。

1. 必要な周波数に応じてPLLレジスタを初期化します。
   - 整数モードの場合は、RCC_PLL1CFGR、RCC_PLL2CFGR、および RCC_PLL3CFGR で PLLxFRACENを0にセットします。
   - 分数モードの場合は、PLLxFRACN を必要な初期値（FracInitValue）にセットし、その後 PLLxFRACEN=1をセットします。
2. PLLxON =1に設定されたら、アプリケーションはPLLxRDY = 1になるまで待つ必要があります。PLLxRDY = 0の間は、PLLxFRACENを変更してはいけません。

3. PLLxRDY = 1になると、PLLxを使用できるようになります。
4. アプリケーションでPLLx周波数を動作中に調整する場合には、以下の条件があります。
   a) PLLxFRACENを0にセットし、PLLの動作を継続しながらPLLxFRACN値を更新する必要があります。
   b) PLLxFRACN（FracValue(n)）に新しい値をアップロードできます。
   c) PLLxFRACENを1にセットして、PLLxFRACNで新しくプログラムされ、PLLで考慮される値を有効化する必要があります。

図40. PLL初期化フロー



注： **PLLxRDYが1になると、PLLx出力周波数とターゲット値の差は±2%未満になります。**

11.4.7 LSEクロック

LSEクリスタルは、32.768kHzのロースピード外部クリスタルまたはセラミック発振子です。時計／カレンダ、その他のタイミング機能のためのリアルタイムクロックペリフェラル（RTC）に、低電力ながら高精度のクロックソースを供給できるという利点があります。

LSEクリスタルのオン／オフの切り替えは、RCC_BDCRのLSEONを使用して行います。LSEがRTC、TAMP、LSECSS以外のペリフェラル機能で使用される場合、RCC_BDCRでLSESYSENビットもセットする必要があります（RTC/TAMP、RCC機能以外のペリフェラルで使用する場合はLSEを参照）。

安定性と短い起動時間、および低い電力消費の最良のバランスが得られるように、クリスタルオシレータの駆動能力は、クリスタル仕様に従い、LSEDRV[1:0]ビットを使用して設定できます。LSEドライブは、LSEを有効にする前にプログラムする必要があります。

RCC_BDCRのLSERDYは、LSEクリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSEクリスタル出力クロック信号はリリースされません。RCC_CIERで有効になっていれば、割込みを生成することができます。

外部ソース（LSEバイパス）

このモードでは、外部クロックソースが必要です。このモードを選択するには、RCC_BDCRでLSEBYPおよびLSEONをセットします。最大50%のデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）でOSC32_INピンを駆動する必要があり、その間、OSC32_OUTピンはGPIOとして使用できます（図37を参照）。

RTC/TAMP、RCC機能以外のペリフェラルで使用する場合はLSE

デフォルトでは、有効にした場合、LSEはRTCとTAMPにのみ送信されます（RTCSEL=01の場合）。

LSEが他のペリフェラル（ペリフェラルクロックやトリガソースなど）のために必要な場合、またはLSEがRCC機能（LSCO、MCO、MSIPLLモードなど）でのみ使用される場合、以下のシーケンスを実行する必要があります。

1. RCC_BDCRのLSEONをセットし、RCC_BDCRでLSERD = 1になるのを待ちます。
2. RCC_BDCRでLSESYSEN = 1をセットします。
3. RCC_BDCRでLSESYSRDY = 1になるのを待ちます。

LSESYSEN = 1になると、LSEの消費量が増えます。

11.4.8 LSIクロック

LSIRCは、独立型ウォッチドッグ（IWDG）およびRTCのSTOPモードおよびSTANDBYモードでの動作を可能にする低電力のクロックソースとして動作します。クロック周波数は、RCC_BDCRのLSIPREDIVに応じて32kHzまたは250Hzになります。LSIPREDIVをセットすると、消費電力を低減できます（詳細は、データシートの電気的特性のセクションを参照）。

IWDGが有効である場合、あるいはRTCまたはTAMPがLSIによってクロック供給される場合、LSIPREDIVの変更はできません。

LSIRCは、RCC_BDCRのLSIONを使用してオン／オフの切り替えができます。

RCC_BDCRのLSIRDYは、LSIオシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。RCC_CIERで有効になっていれば、割込みを生成することができます。

11.4.9 システムクロック（SYSCLK）の選択

次の4つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます：

- MSISオシレータ
- HSI16オシレータ
- HSEオシレータ
- PLL

システムクロックの最大周波数は160MHzです。システムリセット後は、MSISオシレータが4MHzでシステムクロックとして選択されています。クロックソースが直接、またはPLLを経由してシステムクロックとして使用されているときに、このクロック信号を停止することはできません。

あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができている場合（起動遅延時間を経てクロックが安定している状態、または、PLLがロックされている状態）にのみ行われます。準備ができていないクロックソースが選択された場合は、クロックソースの準備ができたときに切り替えが行われます。RCC_CRのステータスビットは、どのクロックがレディ状態であるか、またどのクロックがシステムクロックとして使用されているかを示します。

次の表に、製品電圧レンジに応じて異なるバスの周波数を示します。

表114.バスの最大周波数

製品電圧レンジ	AHB1/AHB2/AHB3/APB1/APB2/APB3
レンジ1	160MHz
レンジ2	110MHz
レンジ3	55MHz
レンジ4	25MHz

11.4.10 クロックソースの周波数と電圧スケーリング

次の表に、製品電圧レンジに応じて異なるクロックソースの周波数を示します。

表115.クロックソースの最大周波数

電圧範囲	クロック周波数					
	MSIS、MSIK	HSI16	HSI48	SHSI	HSE	PLL出力（VCOの最小値~最大値）
レンジ1	全範囲	許可	許可	許可	50MHz	208MHz(1)（128~544MHz）
レンジ2	全範囲	許可	許可	許可	50MHz	110MHz（128~544MHz）
レンジ3	全範囲	許可	許可	許可	50MHz	55MHz（128~330MHz）
レンジ4	最大24MHz	許可	許可（2分周）	許可（2分周）	25MHz	設定禁止

1. 最大周波数は、PLL出力に接続されたペリフェラルによって決まります。

11.4.11 クロックセキュリティシステム（CSS）

CSSはソフトウェアで有効にできます。この場合、HSEオシレータのウェイクアップ時間の後にクロック検出回路が有効になり、このオシレータが停止すると検出回路も無効になります。

HSEクロックで障害が検出された場合、HSEオシレータは自動的に無効になります。クロック障害イベントが一部のタイマブレーク入力に送信され、障害についてソフトウェアに通知するために割込みが生成されます（クロックセキュリティシステム割込み（CSSI））。これにより、MCUで対応処理の実行が可能になります。CSSIIは、Cortex-M33NMI（ノンマスカブル割込み）例外ベクタにリンクされています。

注： **CSSを有効にすると、HSEクロックに障害が発生した場合にCSSIが発生し、NMIが自動的に生成されます。NMIは、CSSIビットがクリアされない限り、無限に実行されます。そのため、RCC_CICRのCSSCをセットすることにより、NMIISRでCSSIをクリアする必要があります。**

HSEオシレータが直接的または間接的にシステムクロックとして使用されている場合（間接的とは、PLL入力クロックとして使用され、そのPLLクロックがシステムクロックとして使用されることを意味する）、障害が検出されると、システムクロックがRCC_CRのSTOPWUCKの設定に応じてMSISまたはHSI16オシレータに切り替えられ、HSEオシレータは無効になります。障害発生時にHSEクロック（分周されているかいないかにかかわらず）がシステムクロックとして使用されているPLLのクロック入力であった場合には、PLLも無効になります。

11.4.12 LSEのクロックセキュリティシステム

LSEのクロックセキュリティシステムは、RCC_BDCRのLSECSSONを書き込むことによって、ソフトウェアで有効にできます。このビットは、ハードウェアリセットまたはRTCソフトウェアリセットにより、あるいはLSEでの障害検出後にのみ無効にできます。LSECSSONは、LSEを有効（LSEONを有効化）かつレディ状態（ハードウェアでLSERDYをセット）にし、RTCSELでRTCクロックを選択した後で書き込む必要があります。

LSEのCSSは、V_{BAT}モードを含め、すべてのモードで動作します。システムリセット（パワーオンリセットを除く）でも動作します。

LSEのCSSは、LSEが消えたことや過周波数を検出します。また、LSEのグリッチをフィルタするには、LSEGFONを設定します。LSEを無効にする場合、LSEGFONへの書込みが必要です（LSEON = 0かつLSERDY = 0）。

外部32kHzオシレータで障害が検出された場合、LSEクロックはRTCに供給しなくなりますが、レジスタに対するハードウェアアクションは行われません。MSIがPLLモードだった場合、このモードは無効になります。

LSEのCSS検出イベントは、TAMPの内部タンパ3に接続されます。

- 低消費電力モードからウェイクアップするには、STM32U575/585 rev.Xデバイスで内部タンパ3が有効（TAMP_CR1でITAMP3E = 1）であり、関連する割込みが有効（TAMP_IERのITAMP3IE）である必要があります。これにより、TAMPバックアップレジスタとバックアップSRAMも消去されますが、TAMP_CR3のITAMP3NOER = 1の場合を除きます（詳細については、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照）。
- 他のすべてのSTM32U575/585のリビジョンと、他のSTM32U5デバイスで、LSEのCSS検出イベントはEXTIラインにも接続され、タンパ検出を有効にしなくてもSTOP0、STOP1、またはSTOP2モードからのウェイクアップに対応するイベントまたは割込みを生成できます（表 118：割込みソースとその制御および表 187：EXTIラインの接続を参照）。

LSEのCSS検出イベントの場合（RCC_BDCRのLSECSSD = 1）、ソフトウェアは、LSECSSONビットを無効化し、欠陥のある32kHzオシレータを停止し（LSEONを無効化）、RTCクロックソースを変更するか（RTCSELで、クロックなし、またはLSIあるいはHSE）、アプリケーションをセキュリティ保護するアクションをとる必要があります。

LSEのCSSの電気特性については、データシートを参照してください。

11.4.13 ADCおよびDACクロック

ADCおよびDACカーネルクロックソースは、RCC_CCIPR3のADCDACSEL[2:0]を使用して選択されます。ADCクロック比は約50%とする必要があります。このため、AHBクロックはADCクロックとして選択する場合、HPREプリスケーラで分周しないようにする必要があります。ADCクロックソースとしてpll2_r_ckを選択した場合、PLL2R分周比を偶数にする必要があります（例えば、2分周または4分周）。

ADCまたはDACがTIMxタイマによって確実かつ精密にトリガされることがアプリケーションで求められる場合、ADCおよびDACカーネルクロックソースとしてHCLKを選択する必要があります。それ以外のクロックソースはTIMxタイマに対して非同期であるため、2つのクロックドメイン間の再同期により、トリガの瞬間の不確実性が高まります。LPTIMxタイマも非同期です。

DACでは、STOPモードで利用可能なサンプルおよびホールドモードで動作する追加の低電力クロック（LSIまたはLSE）が必要です。このクロックを選択するには、RCC_CCIPR3のDAC1SELを使用します。

11.4.14 RTCおよびTAMPクロック

RTCCLKクロックソースは、RTCおよびTAMPによって使用され、HSE/32、LSE、またはLSIクロックのいずれかです。選択するには、RCC_BDCRでRTCSEL[1:0]をプログラムします。この選択は、バックアップドメインをリセットしない限り変更できません。RTCを正常に動作させるには、PCLK周波数が常にRTCCLK周波数以上になるようにシステムを設定する必要があります。タンパがエッジ検出モードで、バックアップレジスタのみを使用する場合、TAMPにはカーネルクロックは不要です。その他すべてのタンパ検出モードでは、カーネルクロックが必要です（詳細については、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照）。

LSEおよびLSIクロックはバックアップドメインですが、HSEクロックはそうではありません。そのため、次のようになります。

- LSEまたはLSIがRTCおよびTAMPクロックとして選択された場合、これらのペリフェラルはV_{DD}供給がオフになっても動作し続けますが、V_{BAT}供給が維持されている必要があります。
- プリスケーラで分周されたHSEクロックがRTCまたはTAMPクロックとして使用されている場合、V_{DD}供給がオフになるか、内部電圧レギュレータがオフになった場合（コアドメインから電源が除去される）、RTCの状態は保証されません。TAMP設定に応じて、カーネルクロックを必要としないモードで使用された場合、これは機能し続けることができます。

RTCおよびTAMPクロックがLSEまたはLSIの場合、RTCは、システムリセット中でもクロック供給され、機能を保ちます。

LSEがRTCまたはTAMPのためだけに必要な場合、電力消費量を抑えるため、LSESYSENをリセット値に保持する必要があります。

11.4.15 タイマクロック

タイマクロック周波数は、ハードウェアによって自動で設定されます。

これには、2つのケースがあります。

- APBプリスケーラが1の場合、タイマクロック周波数はAPBドメインの周波数に設定されます。
- そうでない場合は、APBドメインの周波数の2倍に設定されます。

11.4.16 ウォッチドッグクロック

独立型ウォッチドッグ（IWDG）がハードウェアオプションまたはソフトウェアアクセスによって開始された場合、LSIオシレータは強制的にオンになり、オフにすることはできません。LSIオシレータの過渡期間が終わると、LSI 32kHzクロックがIWDGに供給されます。

11.4.17 OCTOSPIクロック

OCTOSPIxカーネルクロックはOCTOSPIxSEL[1:0]によって選択され、pll1_q_ckまたはpll2_q_ckを使用する場合は最大200MHzとすることができます。

11.4.18 HSPI1クロック

HSPI1カーネルクロックはHSPI1SEL[1:0]によって選択され、pll1_q_ck、pll2_q_ckまたはpll3_r_ckを使用する場合、最大200MHzとすることができます。

11.4.19 OTG_HSクロック

OTG_HSカーネルクロックはOTG_HSPHYによって生成されます。このPHYは次の周波数のリスト（16、19.2、20、24、26、または32MHz）にのみ対応し、精度は±400ppmです。これらの周波数は、HSE、HSE/2、PLL1_P、またはPLL1_P/2を使用して得られ、OTGHSSEL[1:0]マルチプレクサによって選択されます。

11.4.20 DSIクロック

DSIインタフェースクロックは、内部DSIPHYPLLから、またはpll3_p_ckクロックによって生成でき、DSISELマルチプレクサによって選択できます。

11.4.21 LTDCクロック

LTDCインタフェースクロックは、pll2_r_ckまたはpll3_r_ckクロックから生成でき、LTDCSELマルチプレクサによって選択できます。

11.4.22 クロック信号出力

- **MCO**

  マイクロコントローラ·クロック出力（MCO）機能では、外部MCOピンにクロックを出力できます。以下のクロックシグナルのうちの1つをMCOクロックとして選択できます。
  - LSI
  - LSE
  - SYSCLK
  - HSI16
  - HSI48
  - HSE
  - PLLCLK
  - MSIS
  - MSIK

  選択は、RCC_CRのMCOSEL[3:0]によって制御されます。選択されたクロックは、RCC_CRのMCOPRE[2:0]で分周できます。
- **LSCO**

  別の出力（LSCO）では、外部LSCOピンに以下のロースピードクロックの1つを出力することができます。
  - LSI
  - LSE

この出力は、すべてのSTOPモード、STANDBYモード、およびSHUTDOWNモードで引き続き使用できます。この出力は、V_{BAT}モードでは使用できません。選択はLSCOSELビットによって制御され、RCC_BDCRのLSCOENで有効化されます。

MCOクロック出力には、MCOピンで対応するオルタネート機能を選択する必要があります。

11.4.23 TIM15/TIM16/TIM17を使用した内部／外部クロックの測定

TIM15、TIM16、またはTIM17チャネル1の入力キャプチャと、LPTIM1またはLPTIM2チャネル2の入力キャプチャを使用することにより、ボード上のすべてのクロックソースの周波数を間接的に測定することができます。

LSEを使用したHSI16およびMSIの較正

LSEをTIM15、TIM16、およびTIM17のチャネル1の入力キャプチャと、LPTIM2のチャネル2の入力キャプチャに接続する主な目的は、HSI16システムクロックとMSIシステムクロックを正確に測定できるようにすることです（これには、HSI16またはMSISのどちらかがシステムクロックソースとして使用されている必要があります）。LSE信号の連続したエッジ間のHSI16（またはMSIS）クロックのカウント数により、内部クロックの周期が測定されます。LSEクリスタルの高い精度（通常は数十ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造、工程、温度、電圧に関連した周波数偏差に対してソースをトリミングし、補正することができます。

MSIおよびHSI16の4つのオシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります。

この基本概念は、相対測定（HSI16/LSE比など）ができることにあります。そのため、精度は2つのクロックソース間の比に密接にかかわっています。この比が大きければ大きいほど、測定精度は向上します。

注： **LSEが使用できる場合、PLLモードを使用して、MSIをLSEによって自動的にトリミングできます。**

HSEを使用したHSI16およびMSIの較正

HSEが使用できる場合、システムクロックとして使用でき、タイマ入力キャプチャをMSI（1024または4分周）またはHSI/256に接続する必要があります。TIM16およびTIM17チャネル1の入力キャプチャとLPTIM2の入力キャプチャ2は、RCC_CCIPR1でTIMICSEL[2:0]≠0の場合のみ、分周されたオシレータに接続されます。

タイマカウンタが16bitであることと、HSEと入力キャプチャ信号の比率ができるだけ高くなければならないことを考慮して、MSIRC0、MSIRC1、またはMSIRC2の測定時は1024での分周を選択し、MSIRC4の測定時は4での分周を選択する必要があります。

LSI較正

LSIの較正は同じ原則に従って行いますが、リファレンスクロックを変更します。LSIクロックは、TIM16またはTIM17のチャネル1の入力キャプチャ、またはLPTIM1のチャネル2の入力キャプチャに接続する必要があります。次に、HSEをシステムクロックソースとして定義し、LSI信号の連続したエッジ間のクロックカウント数により、内部ロースピードクロック周期が測定されます。

この基本概念は、相対測定（HSE/LSI比など）ができることにあります。そのため、精度は2つのクロックソース間の比に密接にかかわっています。この比が大きければ大きいほど、測定精度は向上します。

11.4.24 ペリフェラルクロックゲーティングおよび自律モード

RUNモードでのペリフェラルクロックゲーティング

各ペリフェラルクロックは、RCC_AHBxENRおよびRCC_APBxENRレジスタの対応するENビットによって有効にできます。

ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

有効化ビットには、グリッチのないペリフェラルのクロックを生成する同期メカニズムが含まれています。イネーブルビットがセットされた後、ペリフェラルバスクロックの2サイクル後にクロックはアクティブになります。

注意： **ペリフェラルのクロックを有効にした直後、ソフトウェアは、2クロックサイクルだけ待ってから、ペリフェラルレジスタにアクセスする必要があります。**

SLEEPおよびSTOPモードでのペリフェラルクロックゲーティング

ペリフェラルが有効な場合、デバイスがSLEEPモードの状態では、RCC_AHBxSMENRおよびRCC_APBxSMENRでペリフェラルのSMENビットをクリアすることにより、そのクロックを自動的にゲートオフすることができます。SLEEPモードでクロックを常にオンにするには、ペリフェラルのENとSMENの両方をセットする必要があります。

ペリフェラルのSMENビットは、STOP0およびSTOP1でペリフェラルリクエスト時にペリフェラルのクロック供給が行われるようにするためにも使用されます。

ペリフェラルによってクロックがリクエストされた場合、このクロックは有効なすべてのペリフェラルに分配されます。そのため、ペリフェラルをSTOPモードで使用しない場合は、STOPモードを開始する前にSMENビットをクリアする必要があります。

注意： **デバイスをSTOPモードからウェイクアップさせる割込みを生成できるようにするには、ペリフェラルのSMENビットをセットする必要があります。ペリフェラルのウェイクアップ割込みがEXTIを通して生成される場合、このビットのセットは不要です。**

STOP0/1/2モードでのペリフェラルクロックゲーティングおよび自律モード

一部のペリフェラルは自律モードに対応しています（表116を参照）。これらのペリフェラルでは、STOPモードでステータスレジスタを動作させ、更新することができるように、カーネルクロックリクエストとAHB/APBバスクロックリクエストを生成できます。ペリフェラルの設定に応じて、DMAリクエストまたは割込みをペリフェラルイベントに関連付けることができます。

自律型ペリフェラルからのAHBまたはAPBバスクロックリクエスト時に、RCC_CFGR1のSTOPWUCKによって選択されたオシレータに応じて、MSIまたはHSI16オシレータがウェイクアップします。

自律型ペリフェラルの設定時にDMAリクエストが有効化されている場合、AHB/APBクロックによりデータ転送が行われます。別のペリフェラルからのリクエストがなければ、転送が完了するとすぐに、バスクロックとオシレータ（HSI16またはMSI）は自動的にオフになります。デバイスは自動的にSTOPモードに戻ります。

自律型ペリフェラルの設定時に割込みが有効化されている場合、割込みによりデバイスがウェイクアップしてRUNモードになります。

AHB3またはAPB3にマッピングされた自律型ペリフェラルはSmartRunドメインに属し、LPDMA1およびSRAM4を使用してSTOP0、STOP1、およびSTOP2モードで自律状態となります。

AHB1、AHB2、APB1、およびAPB2にマッピングされた自律型ペリフェラルはCPUドメインに属し、GPDMA1およびSRAM1/2/3/4/5/6のみを使用してSTOP0およびSTOP1モードで自律状態となります。

以下の表に、自律モード機能を備えたペリフェラルのリストを示します。

表116.自律型ペリフェラル

ドメイン	ペリフェラル	STOP0、1モードで自律状態	STOP2モードで自律状態	関連するDMA	関連するSRAM
CPUドメイン（CD）	U(S)ARTx（x = 1~6）	あり(1)	なし	GPDMA1	SRAM1 SRAM2 SRAM3 SRAM4(2) SRAM5 SRAM6
	SPIx（x = 1,2）				
	I2Cx（x = 1,2,4,5,6）				
	LPTIM2				
	MDF1				
	GPDMA1			-	
SmartRun ドメイン（SRD）	LPUART1	あり (3)	あり (3)	LPDMA1	SRAM4
	SPI3				
	I2C3				
	LPTIMx（x = 1,3,4）				
	ADF1				
	DAC1				
	ADC4				
	LPDMA1			-	

1. ペリフェラルのxxENビットとxxSMENビットの両方がセットされている場合に有効になります（xx = インスタンス名）。
2. SRAM4はSmartRunドメイン（SRD）に属しますが、STOP0およびSTOP1モードでGPDMA1によりアドレス指定できます。
3. ペリフェラルのxxENビット、xxSMENビット、xxAMENビットがすべてセットされている場合に有効になります（xx=インスタンス名）。

CPUドメインのペリフェラルでは、ペリフェラルのxxENビットとxxSMENビットの両方がセットされている場合、STOP0およびSTOP1モードで自律モードが有効になります。

SmartRunドメインのペリフェラルでは、ペリフェラルのxxENビットとxxSMENビットの両方がセットされていて、RCC_SRDAMRでペリフェラルのxxAMENビットがセットされている場合、STOP0、STOP1、およびSTOP2モードで自律モードが有効になります。

自律型ペリフェラルでSTOP0、STOP1、またはSTOP2モードでカーネルクロックが必要な場合、内部オシレータ（HSI16またはMSI）がオフになっていればウェイクアップし、カーネルクロックはそれが必要なペリフェラルにのみ伝播されます。ペリフェラルからカーネルクロックリクエストが発行された場合、HSI16またはMSIは他のペリフェラルでリクエストされていなければオフになります。

CPUドメインに属している自律型ペリフェラルがSTOP0またはSTOP1モードでバスクロック（AHB1、AHB2、APB1、またはAPB2クロック）をリクエストした場合、内部オシレータ（RCC_CFGR1のSTOPWUCKに応じて、HSI16またはMSI）がオフになっていればウェイクアップし、システムクロックはxxENとxxSMENの両方が1に設定されたすべてのペリフェラルに伝播されます。

SmartRunドメインに属している自律型ペリフェラルがSTOP0、STOP1、またはSTOP2モードでバスクロック（AHB3またはAPB3クロック）をリクエストした場合、内部オシレータ（RCC_CFGR1のSTOPWUCK値に応じて、HSI16またはMSI）がオフになっていればウェイクアップし、HCLK3/PCLK3クロックはxxEN、xxSMEN、xxAMENがすべて1に設定されたSmartRunドメインのすべてのペリフェラルに伝播されます。

注意： **デバイスをSTOPモードからウェイクアップさせる割込みを生成できるようにするには、ペリフェラルのAMENビットをセットする必要があります。ペリフェラルのウェイクアップ割込みがEXTIを通して生成される場合、このビットのセットは不要です。**

注： **STOP0、STOP1、またはSTOP2モードでMSIKまたはHSI16を強制的にオンの状態に維持するには、RCC_CRでMSIKERONまたはHSIKERONを設定します。この場合、オシレータは、このオシレータをカーネルクロックとして選択した自律型ペリフェラルのカーネルクロックにのみ伝播されます。これにより、ペリフェラルからカーネルクロックがリクエストされたときにオシレータのウェイクアップ時間を待つ必要がなくなるため、ペリフェラルのボーレートまたは変換速度が高まります。**

ペリフェラルのカーネルクロックとして選択されたLSEまたはLSIは、STOPモードで常にオンの状態となります。

AHB3およびAPB3クロックを強制的にオンの状態に維持するには、PWR_CR2でSRDRUNをセットします。これにより、ペリフェラルからバスクロックがリクエストされたときにオシレータのウェイクアップ時間を待つ必要がなくなるため、LPDMA1の遅延が改善されます。

11.5 RCCセキュリティおよび特権機能の説明

11.5.1 RCC TrustZoneセキュリティ保護モード

TrustZoneセキュリティがFlash_OPTRのTZENオプションビットで有効化される場合、RCCはRCCの設定ビットおよびステータスビットが非セキュアアクセスによって変更されないように保護することができます。

これは、RCC_SECCFGRを通じて設定され、非セキュアアクセスによって以下の機能が読み出されたり変更されたりするのを防ぎます。

- HSE、HSE-CSS、HSI、MSI、LSI、LSE、LSE-CSS、LSCO、HSI48の設定およびステータスビット
- PLL1、PLL2、PLL3、AHB、およびAPBプリスケーラ設定およびステータスビット
- システムクロック（SYSCLK）およびICLKソースクロック選択およびステータスビット
- MCOクロック出力設定、STOPWUCKおよびSTOPKERWUCKビット
- リセットフラグ削除RMVF設定

RCC_PRIVCFGRでSPRIV = 1の場合、RCC_SECCFGRレジスタへの書込みは、セキュアおよび特権アクセスによってのみできます。RCC_PRIVCFGRでSPRIV = 0の場合、RCC_SECCFGRへの書込みは、特権または非特権のセキュアアクセスによってのみできます。

RCC_SECCFGRの読出しは、セキュア、非セキュア、特権、および非特権アクセスによってできます。

ペリフェラルがセキュアとして設定されている場合、そのクロック、リセット、クロックソース選択、および低電力モード時のクロック有効化に関連する制御ビットもセキュアです（RCC_AHBxENR、RCC_APBxENR、RCC_AHBxSMEN、RCC_APBxSMEN、RCC_SRDAMEN、RCC_CCIPR1、RCC_CCIPR2、RCC_CCIPR3、およびRCC_BDCRレジスタ）。

SAESがセキュアとして設定されている場合、SHSI設定およびステータスビットもセキュアです。

RTCまたはTAMPで少なくとも1つの機能がセキュアである場合、RCC_BDCRのBDRSTはセキュアです。

以下の場合にペリフェラルはセキュアになります。

- TZSC（TrustZone セキュリティコントローラ）によってセキュリティ保護可能なペリフェラルについては、このペリフェラルに対応したSECセキュリティビットはGTZCTZSCセキュア設定レジスタでセットされます。
- TrustZone　を認識するペリフェラルについては、このペリフェラルのセキュリティ機能が専用ビットを使って有効化されている場合

表117に、RCC_SECCFGRレジスタのセキュリティ設定ビットとRCCセキュアビットを示します。

1つのセキュリティ設定ビットがセットされると、複数の設定ビットおよびステータスビットがセキュアになります。RCCレジスタに含まれるセキュアおよび非セキュアビットは以下のとおりです。

- セキュアビット：読出しおよび書込み操作は、セキュアアクセスによってのみ可能です。非セキュア読出しは0を返し、書込みアクセスは無視されます。不正アクセスイベントは生成されません。
- 非セキュアビット：制約なしです。読出しおよび書込み操作は、セキュアおよび非セキュアアクセスのどちらでも可能です。
- RCC_SECCFGRへの非セキュア書込みアクセスは無視され、不正アクセスイベントが生成されます。GTZCTZICレジスタでRCC不正アクセス割込みが有効な場合、不正アクセス割込みが生成されます。RCC_SECCFGRは、セキュアまたは非セキュアアクセスによって読み出すことができます。

TrustZoneセキュリティが無効な場合（Flash_OPTRのTZEN = 0）、すべてのレジスタは非セキュアです。RCC_SECCFGR書込みアクセスは無視されます。

表117.RCCセキュリティ設定一覧

RCC_SECCFGRの設定ビット	セキュアビット	対応するレジスタ
HSISEC	HSION、HSIKERON、HSIRDY	RCC_CR
	HSICAL[7:0]、HSITRIM[6:0]	RCC_ICSCR3
	HSIRDYIE	RCC_CIER
	HSIRDYIF	RCC_CIFR
	HSIRDYC	RCC_CICR
HSESEC	HSEON、HSERDY、HSEBYP、HSECSSON、HSEEXT	RCC_CR
	HSERDYIE、HSECSSF	RCC_CIER
	HSERDYIF、HSECSSF	RCC_CIFR
	HSERDYC、HSECSSC	RCC_CICR
MSISEC	MSISON、MSIKERON、MSISRDY、MSIPLLEN、MSIKON、MSIKRDY、MSIPLLSEL、MSIPLLFAST	RCC_CR
	MISIRANGE[3:0]、MISIKRANGE[3:0]、MSIRGSEL、MSIBIAS、MSICAL0[4:0]、MSICAL1[4:0]、MSICAL2[4:0]、MSICAL3[4:0]	RCC_ICSCR1
	MSITRIM0[4:0]、MSITRIM1[4:0]、MSITRIM2[4:0]、MSITRIM3[4:0]	RCC_ICSCR2
	MSISRDYIE、MSIKRDYIE	RCC_CIER
	MSISRDYIF、MSIKRDYIF	RCC_CIFR
	MSISRDYIC、MSIKRDYIC	RCC_CICR
	MSISSRANGE[3:0]、MSIKSRANGE[3:0]	RCC_CSR
LSISEC	LSION、LSIRDY、LSIPREDIV、LSCOSEL、LSCOEN	RCC_BDCR
	LSIRDYIE	RCC_CIER
	LSIRDYIF	RCC_CIFR
	LSIRDYC	RCC_CICR

表117.RCCセキュリティ設定一覧 （続き）

RCC_SECCFGRの設定ビット	セキュアビット	対応するレジスタ
LSESEC	LSECSSON、LSECSSD、LSEDRV[1:0]、LSEBYP、LSERDY、LSEON、LSEGFON、LSESYSRDY、LSESYSEN、LSCOSEL、LSCOEN	RCC_BDCR
	LSERDYIE	RCC_CIER
	LSERDYF	RCC_CIFR
	LSERDYC	RCC_CICR
SYSCLKSEC	SW[1:0]、SWS[1:0]、STOPWUCK、STOPKERWUCK、MCOSEL[3:0]、MCOPRE[2:0]	RCC_CFGR1
	SYSTICKSEL[1:0]	RCC_CCIPR1
	VOS[1:0]	PWR_VOSR
PRESCSEC	HPRE[3:0]、PPRE1[2:0]、PPRE2[2:0]	RCC_CFGR2
	PPRE3 [2:0]	RCC_CFGR3
PLL1SEC	PLL1SRC[1:0]、PLL1RGE[1:0]、PLL1FRACEN、PLL1M[3:0]、PLL1MBOOST、PLL1PEN、PLL1QEN、PLL1REN	RCC_PLL1CFGR
	PLL1N[8:0]、PLL1P[6:0]、PLL1Q[6:0]、PLL1R[6:0]	RCC_PLL1DIVR
	PLL1FRACN [12:0]	RCC_PLL1FRACR
	PLL1RDY, PLL1ON	RCC_CR
	PLL1RDYIE	RCC_CIER
	PLL1RDYF	RCC_CIFR
	PLL1RDYC	RCC_CICR
PLL2SEC	PLL2SRC[1:0]、PLL2RGE[1:0]、PLL2FRACEN、PLL2M[3:0]、PLL2PEN、PLL2QEN、PLL2REN	RCC_PLL2CFGR
	PLL2N[8:0]、PLL2P[6:0]、PLL2Q[6:0]、PLL2R[6:0]	RCC_PLL2DIVR
	PLL2FRACN [12:0]	RCC_PLL2FRACR
	PLL2RDY, PLL2ON	RCC_CR
	PLL2RDYIE	RCC_CIER
	PLL2RDYF	RCC_CIFR
	PLL2RDYC	RCC_CICR
PLL3SEC	PLL3SRC[1:0]、PLL3RGE[1:0]、PLL3FRACEN、PLL3M[3:0]、PLL3PEN、PLL3QEN、PLL3REN	RCC_PLL3CFGR
	PLL3N[8:0]、PLL3P[6:0]、PLL3Q[6:0]、PLL3R[6:0]	RCC_PLL3DIVR
	PLL3FRACN [12:0]	RCC_PLL3FRACR
	PLL3RDY, PLL3ON	RCC_CR
	PLL3RDYIE	RCC_CIER
	PLL3RDYF	RCC_CIFR
	PLL3RDYC	RCC_CICR

表117.RCCセキュリティ設定一覧 （続き）

RCC_SECCFGRの設定ビット	セキュアビット	対応するレジスタ
HSI48SEC(1)	HSI48ON, HSI48RDY	RCC_CR
	HSI48CAL [8:0]	RCC_CRRCR
	HSI48RDYE	RCC_CIER
	HSI48RDYF	RCC_CIFR
	HSI48RDYC	RCC_CICR
ICLKSEL	ICLKSEL[1:0]	RCC_CCIPR1
RMVFSEC	RMVF	RCC_CSR

1. HSI48のTRIMフィールドは、CRSペリフェラルにあります。GTZC1 TZSCセキュア設定レジスタ1のCRSSECビットを使用して、セキュリティ保護してください。

11.5.2 RCC特権保護モード

デフォルトでは、リセット後、すべてのRCCレジスタは、特権アクセスと非特権アクセスのどちらでも読み書き可能ですが、RCC_PRIVCFGRへの書込みは特権アクセスでのみ可能です。RCC_PRIVCFGRは、セキュアおよび非セキュア、特権および非特権アクセスによって読み出すことができます。

RCC_PRIVCFGRのSPRIVは、セキュア特権アクセスでのみ書き込むことができます。このビットは、すべてのRCCセキュア機能の特権アクセスを設定します（RCC_SECCFGRによって、またはセキュリティ保護可能なペリフェラルの場合はGTZCによって、またはTrustZone対応ペリフェラルの場合はペリフェラル自体によって定義されます）。

RCC_PRIVCFGRでSPRIV = 1の場合：

- RCCセキュアビットの書込みは、RCC_SECCFGRも含め、特権アクセスでのみ可能です。
- RCCセキュアビットは、特権アクセスでのみ読み出すことができますが、RCC_SECCFGRおよびRCC_PRIVCFGRは、特権または非特権アクセスによって読み出すことができます。
- 特権RCCビットまたはレジスタへの非特権アクセスは破棄されます。すなわち、ビットはゼロとして読み出され、これらのビットへの書込みは無視されます（RAZ/WI）。

RCC_PRIVCFGRのNSPRIVは、セキュア機能または非セキュア機能の特権アクセスでのみ書き込むことができます（RCC_SECCFGRによって、またはセキュリティ保護可能なペリフェラルの場合はGTZCによって、またはTrustZone対応ペリフェラルの場合はペリフェラル自体によって定義されます）。

RCC_PRIVCFGRでNSPRIV = 1の場合：

- RCC非セキュアビットの書込みは、特権アクセスでのみ可能です。
- RCC 非セキュアビットは、特権アクセスでのみ読み出すことができますが、RCC_PRIVCFGRは、特権または非特権アクセスによって読み出すことができます。
- 特権RCCビットまたはレジスタへの非特権アクセスは破棄されます。すなわち、ビットはゼロとして読み出され、これらのビットへの書込みは無視されます（RAZ/WI）。

11.6 RCC低電力モード

- DMAクロックを含むAHBおよびAPBペリフェラルクロックは、ソフトウェアで無効にできます。
- SLEEPモードではCPUクロックを停止します。メモリインタフェースクロック（Flashメモリ、キャッシュ、およびすべてのSRAMインタフェース）は、ソフトウェアによってSLEEPモード中に停止できます。AHB-APBブリッジのクロックは、SLEEPモード中にこれらに接続されたペリフェラルのクロックがすべて無効になった場合に、ハードウェアによって無効化されます。
- STOP モード（STOP0、STOP1、STOP2、および STOP3）では、コアドメインのすべてのクロックを停止し、PLL、HSI16、HSI48、SHSI、MSI、およびHSEオシレータを無効にします。ただし、自律モード用にペリフェラルからリクエストされた場合や、ウェイクアップ割込みを生成するために、HSI16またはMSIをオンにすることができます（詳細は、セクション11.4.24を参照）。LSIおよびLSEは、STOPモードでもアクティブなままです。
- STANDBY モードおよびSHUTDOWN モードでは、コアドメインのすべてのクロックを停止し、PLL、HSI16、HSI48、SHSI、MSI、およびHSEオシレータを無効にします。

CPUのディープスリープモードは、DBGMCU_CRレジスタのDBG_STOPまたはDBG_STANDBYビットをセットすることによって、デバッグのためにオーバライドすることができます。

STOPモード（STOP0、STOP1、STOP2、またはSTOP3）の終了時、RCC_CFGR1のSTOPWUCKに応じて、システムクロックはMSISまたはHSI16になります。MSISまたはMSIKオシレータの周波数（レンジおよびユーザトリミング）は、STOPモードに移行する前に設定されたものです。ただし、24MHzを超える場合は例外です。この場合、MSISまたはMSIKレンジは24MHzレンジとなります。HSI16のユーザトリミングは保たれます。MSIがMSIPLLFAST = 0でSTOPモードに移行する前にPLLモードだった場合、LSEがSTOPモード中オンのままであった場合も、ウェイクアップ後PLLモード安定時間を待つ必要があります。MSIPLLFAST = 1の場合、STOP0、STOP1、またはSTOP2モードからのウェイクアップ後、安定化までの時間なしでPLLモードの精度が維持されます。STOP3モードの終了時、MSIPLLFASTビットは影響を受けません。

システムクロックによって使用されるものに加えて、他の内部オシレータを自動的にウェイクアップできます。これにより、デバイスがRUNモードに復帰したときに、他のオシレータのウェイクアップ時間の間待つ必要がなくなります。これは、RCC_CFGR1のSTOPKERWUCKを使用して行われます。

STANDBYおよびSHUTDOWNモード終了時、システムクロックはMSISです。STANDBYモードからのウェイクアップ時のMSISおよびMSIK周波数は、RCC_CSRのMSISSRANGEおよびMSIKSRANGEで1~4MHzに設定されます。SHUTDOWNモードからのウェイクアップ時のMSI周波数は4MHzです。ユーザトリミングは失われます。

Flashメモリプログラム操作が実行中の場合、Flashメモリインタフェースのアクセスが終了してから、STOPモード、STANDBYモード、またはSHUTDOWNモードに移行します。APBドメインにアクセス中の場合、APBアクセスが終了してから、STOPモード、STANDBYモード、またはSHUTDOWNモードに移行します。自律型ペリフェラルによりシステムクロックリクエストが生成される場合、システムクロックリクエストが発行されてから、STOPモード、STANDBYモード、SHUTDOWNモードに移行します。

11.7 RCC割込み

以下の表に、割込みソースと、その制御方法を示します。

表118.割込みソースとその制御

割込みベクタ	割込みイベントフラグ	説明	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOP、STANDBY、SHUTDOWNモードの終了
RCC	LSIRDYF	LSIレディ	LSIRDYIEおよびLSISEC = 0	LSIRDYCを「1」にセット	あり	なし
	LSERDYF	LSEレディ	LSERDYIEおよびLSESEC = 0	LSERDYCを「1」にセット	あり	なし
	HSIDRYF	HSIレディ	HSIDRYIEおよびHSISEC = 0	HSIRDYCを「1」にセット	あり	なし
	HSERDYF	HSEレディ	HSERDYIEおよびHSESEC = 0	HSERDYCを「1」にセット	あり	なし
	MSISRDYF	MSISレディ	MSISRDYIEおよびMSISEC = 0	MSISRDYCを1にセット	あり	なし
	MSIKRDYF	MSIKレディ	MSISRDYIEおよびMSISEC = 0	MSIKRDYCを1にセット	あり	なし
	SHSIRDYF	SHSIレディ	SHSIRDYIEおよびSAESSEC = 0（GTZC）	SHSIRDYCを1にセット	あり	なし
	HSI48RDYF	HSI48レディ	HSI48RDYIEおよびHSI48SEC = 0	HSI48RDYCを「1」にセット	あり	なし
	PLL1RDYF	PLL1レディ	PLL1RDYIEおよびPLL1SEC = 0	PLL1RDYCを「1」にセット	あり	なし
	PLL2RDYF	PLL2レディ	PLL2RDYIEおよびPLL2SEC = 0	PLL2RDYCを「1」にセット	あり	なし
	PLL3RDYF	PLL3レディ	PLL3RDYIEおよびPLL3SEC = 0	PLL3DYCを「1」にセット	あり	なし
RCC_S(1)	LSIRDYF	LSIレディ	LSIRDYIEおよびLSISEC = 1	LSIRDYCを「1」にセット	あり	なし
	LSERDYF	LSEレディ	LSERDYIEおよびLSESEC = 1	LSERDYCを「1」にセット	あり	なし
	HSIDRYF	HSIレディ	HSIDRYIEおよびHSISEC = 1	HSIRDYCを「1」にセット	あり	なし
	HSERDYF	HSEレディ	HSERDYIEおよびHSESEC = 1	HSERDYCを「1」にセット	あり	なし
	MSISRDYF	MSISレディ	MSISRDYIEおよびMSISEC = 1	MSISRDYCを1にセット	あり	なし
	MSIKRDYF	MSIKレディ	MSISRDYIEおよびMSISEC = 1	MSIKRDYCを1にセット	あり	なし

表118.割込みソースとその制御 （続き）

割込みベクタ	割込みイベントフラグ	説明	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOP、STANDBY、SHUTDOWNモードの終了
RCC_S (1)	SHSIRDYF	SHSIレディ	SHSIRDYIEおよびSAESSEC(2) = 1	SHSIRDYCを1にセット	あり	なし
	HSI48RDYF	HSI48レディ	HSI48RDYIEおよびHSI48SEC = 1	HSI48RDYCを「1」にセット	あり	なし
	PLL1RDYF	PLL1レディ	PLL1RDYIEおよびPLL1SEC = 1	PLL1RDYCを「1」にセット	あり	なし
	PLL2RDYF	PLL2レディ	PLL2RDYIEおよびPLL2SEC = 1	PLL2RDYCを「1」にセット	あり	なし
	PLL3RDYF	PLL3レディ	PLL3RDYIEおよびPLL3SEC = 1	PLL3RDYCを「1」にセット	あり	なし
TAMP	ITAMP3F (3)	LSEのCSS障害	LSECSSONおよびITAMP3E (3) およびITAMP3IE(3)	CITAMP3F (3) を1にセット	あり	あり
NMI	HSECSSF	HSEのCSS障害	-(4)	HSECSSCを「1」にセット	あり	なし
LSECSS(5)	EXTI経由	LSEのCSS障害	EXTI経由	EXTI経由	あり	あり(6) ／なし
MSI_PLL_UNLOCK (5)	EXTI経由	MSI PLLモードのアンロック(7)	EXTI経由	EXTI経由	あり	あり (6) ／なし

1. RCCセキュア割込みベクタは、TrustZoneが有効なときのみ使用されます。
2. SAESSECビットはGTZCペリフェラル内にあります。
3. LSE CSS障害イベント（LSECSSD）は、TAMP内部タンパ3に接続されます。割込みをこのイベントに関連付けるには、内部タンパ3が有効でなければならず、内部タンパ3割込みが有効でなければなりません。ITAMP3F、ITAMP3E、ITAMP3IE、およびCITAMP3Fビットは、TAMPペリフェラルにあります。そのため、LSECSSタンパ割込みにより、表 637：TAMP相互接続に示すデバイス機密情報が消去またはブロックされます。
4. セキュリティシステムの機能が有効である場合（HSECSSON = 「1」）、この割込みをマスクすることはできません。
5. STM32U575/585リビジョンXデバイスでは使用できません。他のすべてのSTM32U575/585リビジョン、および他のSTM32U5シリーズデバイスで使用できます。
6. この割込みは、STOP0、STOP1、およびSTOP2モードからのみウェイクアップ可能です。
7. この割込みは、LSEパルスの欠落のため、MSIがPLLモードを終了したことを示します。その結果、MSI周波数の精度が低下します。

11.8 RCCレジスタ

11.8.1 RCCクロック制御レジスタ（RCC_CR）

アドレスオフセット：0x000

リセット値：0x0000 0035

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

HSEBYPおよびHSEEXTはパワーオン·リセット時にクリアされます。それ以外のリセットでは影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	HSEEXT	CSSON	HSEBYP	HSERDY	HSEON
		r	rw	r	rw	r	rw				rw	rs	rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SHSIRDY	SHSION	HSI48RDY	HSI48ON	Res.	HSIRDY	HSIKERON	HSION	MSIPLLFAST	MSIPLLSEL	MSIKRDY	MSIKON	MSIPLLEN	MSISRDY	MSIKERON	MSISON
r	rw	r	rw		r	rw	rw	rw	rw	r	rw	rw	r	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **PLL3RDY**：PLL3クロックレディフラグ

このビットは、PLL3がロック状態であることを示すために、ハードウェアによってセットされます。

0：PLL3がアンロック状態

1：PLL3がロック状態

ビット28 **PLL3ON**：PLL3イネーブル

このビットは、PLL3を有効にするために、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、ハードウェアによってクリアされます。

0：PLL3 OFF

1：PLL3 ON

ビット27 **PLL2RDY**：PLL2クロックレディフラグ

このビットは、PLL2がロック状態であることを示すために、ハードウェアによってセットされます。

0：PLL2がアンロック状態

1：PLL2がロック状態

ビット26 **PLL2ON**：PLL2イネーブル

このビットは、PLL2を有効にするために、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、ハードウェアによってクリアされます。

0：PLL2 OFF

1：PLL2 ON

ビット25 **PLL1RDY**：PLL1クロックレディフラグ

このビットは、PLL1がロック状態であることを示すために、ハードウェアによってセットされます。

0：PLL1がアンロック状態

1：PLL1がロック状態

ビット24 **PLL1ON**：PLL1イネーブル

このビットは、メインPLLを有効にするために、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、ハードウェアによってクリアされます。PLL1クロックがシステムクロックとして使用されている場合、このビットをリセットできません。

0：PLL1 OFF

1：PLL1 ON

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **HSEEXT**：HSE外部クロックバイパスモード

このビットは、バイパスモードで外部クロックモードを選択するために、ソフトウェアによってセット／リセットされます。デバイスでHSEONビットを使用できるよう、外部クロックモードを設定する必要あります。このビットに書き込めるのは、HSEオシレータが無効なときだけです。このビットは、HSEバイパスモードが有効になっている場合のみアクティブとなります。

0：外部HSEクロックがアナログモード

1：外部HSEクロックがデジタルモード（I/Oシュミットトリガを通じて）

ビット19 **CSSON**：クロックセキュリティシステム有効化

このビットは、クロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。CSSONがセットされているとき、HSEオシレータがレディになるとクロック検出回路がハードウェアによって有効にされ、HSEクロックの障害が検出された場合ハードウェアによって無効にされます。このビットはセット専用で、リセットによってクリアされます。

0：クロックセキュリティシステムオフ（クロック検出回路オフ）

1：クロックセキュリティシステムオン（HSEオシレータが安定していればクロック検出回路オン、そうでなければオフ）

ビット18 **HSEBYP**：HSEクリスタルオシレータバイパス

このビットは、外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。デバイスで使用するには、HSEONビットをセットして外部クロックを有効にする必要があります。このビットに書き込めるのは、HSEオシレータが無効なときだけです。

0：HSEクリスタルオシレータはバイパスされません。

1：HSEクリスタルオシレータが外部クロックによってバイパスされます。

ビット17 **HSERDY**：HSEクロックレディフラグ

このビットは、HSEオシレータが安定状態であることを示すために、ハードウェアによってセットされます。

0：HSEオシレータはレディ状態ではありません。

1：HSEオシレータはレディ状態です。

注： HSEONビットがクリアされると、HSERDYは、HSEクロックの6サイクル後にローになります。

ビット16 **HSEON**：HSEクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、HSEオシレータを停止するためにハードウェアによってクリアされます。HSEオシレータが直接的または間接的にシステムクロックとして使用されている場合は、このビットをリセットできません。

0：HSEオシレータはオフです。

1：HSEオシレータはオンです。

ビット15 **SHSIRDY**：SHSIクロックレディフラグ

このビットは、SHSIオシレータが安定状態であることを示すために、ハードウェアによってセットされます。（SHSIONをセットすることで）ソフトウェアによってSHSIが有効になった場合のみセットされます。

0：SHSIオシレータはレディ状態ではありません。

1：SHSIオシレータはレディ状態です。

注： SHSIONビットがクリアされると、SHSIRDYは、SHSIクロックの6サイクル後にローになります。

ビット14 **SHSION**：SHSIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、SHSIを停止するためにハードウェアによってクリアされます。

0：SHSIオシレータはオフです。

1：SHSIオシレータはオンです。

ビット13 **HSI48RDY**：HSI48クロックレディフラグ

このビットは、HSI48オシレータが安定状態であることを示すために、ハードウェアによってセットされます。このビットは、HSI48が有効になった場合のみソフトウェアによって（HSI48ONをセットすることで）セットされます。

0：HSI48オシレータはレディ状態ではありません。

1：HSI48オシレータはレディ状態です。

ビット12 **HSI48ON**：HSI48クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、HSI48を停止するためにハードウェアによってクリアされます。

0：HSI48オシレータはオフです。

1：HSI48オシレータはオンです。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **HSIRDY**：HSI16クロックレディフラグ

このビットは、HSI16オシレータが安定状態であることを示すために、ハードウェアによってセットされます。このビットは、（HSIONをセットすることで）ソフトウェアによってHSI16が有効になった場合のみセットされます。

0：HSI16オシレータはレディ状態ではありません。

1：HSI16オシレータはレディ状態です。

注： HSIONビットがクリアされると、HSIRDYは、HSI16クロックの6サイクル後にローになります。

ビット9 **HSIKERON**：HSI16を一部のペリフェラルカーネルのために有効化

このビットは、STOPモードでもHSI16を強制的にオンにするために、ソフトウェアによってセット／クリアされます。STOPモードでHSI16をオンのままにしておくと、HSI16の起動時間による通信速度の低下を回避できます。このビットは、HSIONの値への影響はありません。詳細については、セクション11.4.24を参照してください。

このビットは、STOP3モードに移行する前に、0に設定する必要があります。

0：HSI16オシレータへの影響はありません。

1：HSI16オシレータは、強制的にオンになります（STOPモードの場合も含む）。

ビット8 **HSION**：HSI16クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、HSI16オシレータを停止するためにハードウェアによってクリアされます。このビットは、STOPモードを終了するときにSTOPWUCK = 1の場合、またはHSEクリスタルオシレータに障害が発生した場合、HSI16オシレータを強制的にオンにするために、ハードウェアによってセットされます。このビットは、HSI16が直接または間接的にシステムクロックとして使用されている場合に、ハードウェアによってセットされます。

0：HSI16オシレータはオフです。

1：HSI16オシレータはオンです。

ビット7 **MSIPLLFAST**：MSIのPLLモードの高速起動

このビットは、MSIクロックソースのPLLモードの高速起動を有効／無効にするために、ソフトウェアによってセット／リセットされます。このビットは、PLLモードが選択されている場合のみ使用されます（MSIPLLEN = 1）。

注意： 高速起動機能は、PLLモードの初回選択時にはアクティブになりません。高速起動がアクティブになるのは、MSIのPLLモードがオフ状態から復帰したときです。

0：MSIのPLLの通常起動

1：MSIのPLLの高速起動

ビット6 **MSIPLLSEL**：PLLモードを使用するMSIクロックの選択

このビットは、PLLモードを使用するMSI出力クロックを選択するために、ソフトウェアによってセット／クリアされます。これらに書き込めるのは、MSIPLLモードが無効なときだけです（MSIPLLEN = 0）。

0：PLLモードがMSIK（MSIカーネル）クロック出力に適用されます。

1：PLLモードがMSIS（MSIシステム）クロック出力に適用されます。

注： MSIカーネルクロック出力でMSIシステムクロック出力と同じオシレータソースを使用する場合、両方のクロック出力にPLLモードが適用されます。

ビット5 **MSIKRDY**：MSIKクロックレディフラグ

このビットは、MSIKが安定状態であることを示すために、ハードウェアによってセットされます。このビットはMSIKONをセットすることでソフトウェアによってMSIカーネルオシレータが有効化された場合のみセットされます。

0：MSIK（MSIカーネル）オシレータはレディ状態ではありません。

1：MSIK（MSIカーネル）オシレータはレディ状態です。

注： MSIKONビットがクリアされると、MSIKRDYは、MSIKオシレータクロックの6サイクル後にローになります。

ビット4 **MSIKON**：MSIKクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、MSIKを停止するためにハードウェアによってクリアされます。このビットは、STANDBYまたはSHUTDOWNモードを終了するときに、MSIKオシレータを強制的にオンにするためにハードウェアによってセットされます。STOPモードを終了するときにSTOPWUCK = 0またはSTOPKERWUCK = 0の場合、またはHSEオシレータに障害が発生した場合、MSIKオシレータを強制的にオンにするために、ハードウェアによってセットされます。

0：MSIK（MSIカーネル）オシレータは無効です。

1：MSIK（MSIカーネル）オシレータは有効です。

ビット3 **MSIPLLEN**：MSIクロックのPLLモードの有効化

このビットは、MSIクロックソースのPLL部分を有効／無効にするために、ソフトウェアによってセット／クリアされます。

MSIPLLENは、LSEを有効（LSEONを有効化）かつレディ状態（ハードウェアでLSERDYをセット）にした後で、有効にする必要があります。ハードウェアを保護することで、LSEがレディ状態になっていない状態でMSIPLLENが有効化されるのを防止することができます。このビットは、LSEが無効（LSEON = 0）になったときや、LSEのCSSがLSE障害を検出したときに、ハードウェアによってクリアされます（RCC_CSRレジスタを参照）。

0：MSIのPLLモードがオフ

1：MSIのPLLモードがオン

ビット2 **MSISRDY**：MSISクロックレディフラグ

このビットは、MSISオシレータが安定状態であることを示すために、ハードウェアによってセットされます。（MSISONをセットすることで）ソフトウェアによってMSISが有効になった場合のみセットされます。

0：MSIS（MSIシステム）オシレータはレディ状態ではありません。

1：MSIS（MSIシステム）オシレータはレディ状態です。

注： MSISONビットがクリアされると、MSISRDYは、MSISクロックの6サイクル後にローになります。

ビット1 **MSIKERON**：MSIを一部のペリフェラルカーネルのために有効化

このビットは、STOPモードでもMSIを強制的にオンにするために、ソフトウェアによってセット／クリアされます。STOPモードでMSIをオンのままにしておくと、MSIの起動時間による通信速度の低下を回避できます。このビットはMSISONおよびMSIKON値には影響しません（詳細は、セクション11.4.24を参照）。このビットは、STOP3モードに移行する前に、0に設定する必要があります。

0：MSIオシレータへの影響はありません。

1：MSIオシレータは、強制的にオンになります（STOPモードの場合も含む）。

ビット0 **MSISON**：MSISクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。STOP、STANDBY、またはSHUTDOWNモードに移行するときに、MSISオシレータを停止するためにハードウェアによってクリアされます。このビットは、STANDBYまたはSHUTDOWNモードを終了するときに、MSISオシレータを強制的にオンにするためにハードウェアによってセットされます。STOPモードを終了するときにSTOPWUCK = 0の場合、またはHSEオシレータに障害が発生した場合、MSISオシレータを強制的にオンにするために、ハードウェアによってセットされます。

直接または間接的にシステムクロックとして使用されている場合に、ハードウェアによってセットされます。

0：MSIS（MSIシステム）オシレータはオフです。

1：MSIS（MSIシステム）オシレータはオンです。

11.8.2 RCC内部クロックソース較正レジスタ1（RCC_ICSCR1）

アドレスオフセット：0x008

リセット値：0x440X XXXX

Xは出荷時にプログラムされます。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSISRANGE[3:0]				MSIKRANGE[3:0]				MSIRGSEL	MSIBIAS	Res.	Res.	MSICAL0 [4:1]			
rw	rw	rw	rw	rw	rw	rw	rw	rs	rw			r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSICAL0[0]	MSICAL1 [4:0]					MSICAL2 [4:0]					MSICAL3 [4:0]				
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 **MSISRANGE[3:0]**：MSISクロック範囲

これらのビットは、MSIRGSELがセットされているときに、MSISオシレータの周波数範囲を選択するためにソフトウェアによって設定されます。以下の16種類の周波数範囲が使用できます。

0000：レンジ0（約48MHz）
0001：レンジ1（約24MHz）
0010：レンジ2（約16MHz）
0011：レンジ3（約12MHz）
0100：レンジ4（約4MHz、リセット値）
0101：レンジ5（約2MHz）
0110：レンジ6（約1.33MHz）
0111：レンジ7（約1MHz）
1000：レンジ8（約3.072MHz）
1001：レンジ9（約1.536MHz）
1010：レンジ10（約1.024MHz）
1011：レンジ11（約768kHz）
1100：レンジ12（約400kHz）
1101：レンジ13（約200kHz）
1110：レンジ14（約133kHz）
1111：レンジ15（約100kHz）

注： **MSISがオフ（MSISON = 0）のとき、またはレディ状態（MSIRDY = 1）のときMSISRANGEを変更できます。MSISがオンで、レディ状態ではない（MSION = 1かつMSISRDY = 0）とき、MSISRANGEは変更しないでください。**

STOPモードからのウェイクアップ時、MSISRANGEは保持されます。ただし、MSISレンジが24MHzを超える場合は例外です。この場合、MSISRANGEはハードウェアによってレンジ2（24MHz）に変更されます。

ビット27:24 **MSISRANGE[3:0]**：MSISクロック範囲

これらのビットは、MSIRGSELがセットされているときに、MSIKオシレータの周波数範囲を選択するためにソフトウェアによって設定されます。以下の16種類の周波数範囲が使用できます。

0000：レンジ0（約48MHz）
0001：レンジ1（約24MHz）
0010：レンジ2（約16MHz）
0011：レンジ3（約12MHz）
0100：レンジ4（約4MHz、リセット値）
0101：レンジ5（約2MHz）
0110：レンジ6（約1.33MHz）
0111：レンジ7（約1MHz）
1000：レンジ8（約3.072MHz）
1001：レンジ9（約1.536MHz）
1010：レンジ10（約1.024MHz）
1011：レンジ11（約768kHz）
1100：レンジ12（約400kHz）
1101：レンジ13（約200kHz）
1110：レンジ14（約133kHz）
1111：レンジ15（約100kHz）

注： MSIKがオフ（MSISON = 0）のとき、またはレディ状態（MSIKRDY = 1）のとき、MSIKRANGEを変更できます。MSIKがオンで、レディ状態ではない（MSIKON = 1かつMSIKRDY = 0）とき、MSIKRANGEは変更しないでください。

STOPモードからのウェイクアップ時、MSIKRANGEは保持されます。ただし、MSIKレンジが24MHzを超える場合は例外です。この場合、MSIKRANGEはハードウェアによってレンジ2（24MHz）に変更されます。

ビット23 **MSIRGSEL**：MSIクロック範囲選択

このビットは、MSISRANGE[3:0]およびMSIKRANGE[3:0]を使用してMSISおよびMSIKクロック範囲を選択するために、ソフトウェアによってセットされます。0を書き込んでも、ビットの値は変化しません。STANDBYまたはSHUTDOWNモードの終了後、またはリセット後、このビットは0にセットされ、MSISおよびMSIKレンジはRCC_CSRのMSISSRANGE[3:0]およびMSIKSRANGE[3:0]によって規定されます。

0：MSIS/MSIKレンジはRCC_CSRのMSISSRANGE[3:0]およびMSIKSRANGE[3:0]によって規定されます。

1：MSIS/MSIKレンジはRCC_ICSCR1のMSISRANGE[3:0]およびMSIKSRANGE[3:0]によって規定されます。

ビット22 **MSIBIAS**：MSIバイアスモード選択

このビットは、MSIバイアスモードを選択するために、ソフトウェアによってセットされます。デフォルトでは、出力クロックの精度を維持するために、MSIバイアスは連続モードとなっています。このビットをセットすると、レギュレータがレンジ4にある場合、またはデバイスがSTOP1またはSTOP2モードの場合、MSIの消費量が低減できますが、MSIの精度は低下します。

0：MSIバイアス連続モード（クロック精度の高速整定時間）

1：レギュレータがレンジ4にある場合、またはデバイスがSTOP1またはSTOP2モードの場合のMSIバイアスサンプリングモード（超低電力モード）

ビット21:20 予約済みであり、リセット値に保持する必要があります。

ビット19:15 **MSICAL0[4:0]**：MSIレンジ0~3でのMSIRC0クロック較正

これらのビットは起動時に、出荷時にプログラムされたレンジ0~3のMSIRC0較正トリミング値で初期化されます。MSITRIM0の書込み時に、MSICAL0はMSITRIM0[4:0]と出荷時にプログラムされた較正トリミング値MSIRC0[4:0]の合計値で更新されます。

注意： MSITRIM ビットフィールドと出荷時にプログラムされたこの較正値用のビットフィールドの加算によるオーバーフローを制限するためのハードウェア保護はありません。ユーザレベルでソフトウェアによりコントロールする必要があります。

ビット14:10 **MSICAL1[4:0]**：MSIレンジ4~7でのMSIRC1クロック較正

これらのビットは起動時に、出荷時にプログラムされたレンジ4~7のMSIRC1較正トリミング値で初期化されます。MSITRIM1 の書込み時に、MSICAL1 はMSITRIM1[4:0] と出荷時の較正トリミング値MSIRC1[4:0]の合計値で更新されます。

注意： MSITRIM ビットフィールドと出荷時にプログラムされたこの較正値用のビットフィールドの加算によるオーバーフローを制限するためのハードウェア保護はありません。ユーザレベルでソフトウェアによりコントロールする必要があります。

ビット9:5 **MSICAL2[4:0]**：MSIレンジ8~11でのMSIRC2クロック較正

これらのビットは起動時に、出荷時にプログラムされたレンジ8~11のMSIRC2較正トリミング値で初期化されます。MSITRIM2 の書込み時に、MSICAL2 はMSITRIM2[4:0] と出荷時の較正トリミング値MSIRC2[4:0]の合計値で更新されます。

注意： MSITRIM ビットフィールドと出荷時にプログラムされたこの較正値用のビットフィールドの加算によるオーバーフローを制限するためのハードウェア保護はありません。ユーザレベルでソフトウェアによりコントロールする必要があります。

ビット4:0 **MSICAL3[4:0]**：MSIレンジ12~15でのMSIRC3クロック較正

これらのビットは起動時に、出荷時にプログラムされたレンジ12~15のMSIRC3較正トリミング値で初期化されます。MSITRIM3の書込み時に、MSICAL3はMSITRIM3[4:0]と出荷時の較正トリミング値MSIRC2[4:0]の合計値で更新されます。

注意： MSITRIM ビットフィールドと出荷時にプログラムされたこの較正値用のビットフィールドの加算によるオーバーフローを制限するためのハードウェア保護はありません。ユーザレベルでソフトウェアによりコントロールする必要があります。

11.8.3 RCC内部クロックソース較正レジスタ2（RCC_ICSCR2）

アドレスオフセット：0x00C

リセット値：0x0008 4210

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSITRIM0 [4:1]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSITRIM0[0]	MSITRIM1 [4:0]					MSITRIM2 [4:0]					MSITRIM3 [4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:15 **MSITRIM0[4:0]**：レンジ0~3でのMSIクロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は出荷時にプログラムされた較正トリミング値MSIRC0[4:0]ビットに加算されます。MSIの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

ビット14:10 **MSITRIM1[4:0]**：レンジ4~7でのMSIクロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は出荷時にプログラムされた較正トリミング値MSIRC1[4:0]ビットに加算されます。MSIの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

ビット9:5 **MSITRIM2[4:0]**：レンジ8~11でのMSIクロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は出荷時にプログラムされた較正トリミング値MSIRC2[4:0]ビットに加算されます。MSIの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

ビット4:0 **MSITRIM3[4:0]**：レンジ12~15でのMSIクロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は出荷時にプログラムされた較正トリミング値MSIRC3[4:0]ビットに加算されます。MSIの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

注： LSEによるハードウェアの自動較正は、ソフトウェア較正と一緒に行わないでください。

11.8.4 RCC内部クロックソース較正レジスタ3（RCC_ICSCR3）

アドレスオフセット：0x010

リセット値：0x0010：0XXX

Xは出荷時にプログラムされます。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSITRIM[4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HSICAL[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **HSITRIM[4:0]**：HSIクロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用できます。HSIの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **HSICAL[11:0]**：HSIクロック較正

これらのビットは起動時に、出荷時にプログラムされたHSI較正トリミング値で初期化されます。

11.8.5 RCCクロックリカバリRCレジスタ（RCC_CRRCR）

アドレスオフセット：0x014

リセット値：0x0000 0XXX

Xは出荷時にプログラムされます。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL [8:0]								
							r	r	r	r	r	r	r	r	r

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **HSI48CAL[8:0]**：HSI48クロック較正

これらのビットは起動時に、出荷時にプログラムされたHSI48較正トリミング値で初期化されます。

11.8.6 RCCクロック設定レジスタ1(RCC_CFGR1)

アドレスオフセット：0x01C

リセット値：0x0000 0000

アクセス：0 ≤ ウェイトステート ≤ 2、ワード、ハーフワード、およびバイトアクセス

クロックソースの切替え中にアクセスが発生した場合に限り、1または2個のウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCOPRE[2:0]			MCOSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOPKERWUCK	STOPWUCK	SWS[1:0]		SW[1:0]	
										rw	rw	r	r	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:28 **MCOPRE[2:0]**：マイクロコントローラクロック出力プリスケーラ

このビットフィールドは、ソフトウェアによってセット／クリアされます。MCO出力を有効にする前に、このプリスケーラを変更することを強く推奨します。

000：MCOは1分周されます。

001：MCOは2分周されます。

010：MCOは4分周されます。

011：MCOは8分周されます。

100：MCOは16分周されます。

その他：設定禁止

ビット27:24 **MCOSEL[3:0]**：マイクロコントローラクロック出力

このビットフィールドは、ソフトウェアによってセット／クリアされます。

0000：MCO出力無効、MCOにクロックなし

0001：SYSCLKシステムクロックの選択

0010：MSISクロックの選択

0011：HSI16クロックの選択

0100：HSEクロックの選択

0101：メインPLLクロックpll1_r_ckの選択

0110：LSIクロックの選択

0111：LSEクロックの選択

1000：内部HSI48クロックの選択

1001：MSIKクロックの選択

その他：予約済み

注： このクロック出力では、起動時またはMCOクロックソースの切り替え時に、切り捨てのサイクルがある場合があります。

ビット23:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **STOPKERWUCK**：STOPカーネルクロックからのウェイクアップの自動有効化の選択

このビットは、STOPモードの終了時に自動的に別のオシレータを有効にするために、ソフトウェアによってセット／クリアされます。このオシレータは、ペリフェラルによって独立したカーネルクロックとして使用できます。

0：STOPモードの終了時、またはHSEのCSSイベントの発生時に、MSIKオシレータが自動的に有効になります。

1：STOPモードの終了時、またはHSEのCSSイベントの発生時に、HSI16オシレータが自動的に有効になります。

ビット4 **STOPWUCK**：STOPモードからのウェイクアップおよびCSSバックアップクロック選択

このビットは、STOPモード終了時に使用されるシステムクロックを選択するために、ソフトウェアによってセット／クリアされます。選択されたクロックは、HSEのクロックセキュリティシステムの非常時クロックにも使用されます。

注： このビットが CSS バックアップクロックの選択に使用される場合、CSS イベント後に別のオシレータが起動されるのを回避するために、STOPKERWUCKビット値をSTOPWUCKと同じ値でプログラムする必要があります。

注意： CSSがRCC_CRのHSECSSONビットによって有効化され、システムクロックがHSE（SWS = 10）であるかHSEのスイッチ（SW = 10）を要求している場合、STOPWUCKを変更しないでください。

0：STOPからのウェイクアップクアップクロックおよびCSSバックアップクロックとしてMSISオシレータが選択されます。

1：STOPからのウェイクアップクアップクロックおよびCSSバックアップクロックとしてHSI16オシレータが選択されます。

ビット3:2 **SWS[1:0]**：システムクロックスイッチステータス

このビットは、どのクロックソースがシステムクロックとして使用されるかを示すために、ハードウェアによってセット／クリアされます。

00：MSISオシレータがシステムクロックとして使用されます。

01：HSI16オシレータがシステムクロックとして使用されます。

10：HSEがシステムクロックとして使用されます。

11：PLL pll1_r_ckがシステムクロックとして使用されます。

ビット1:0 **SW[1:0]**：システムクロックスイッチ

このビットフィールドは、システムクロックソース（SYSCLK）を選択するために、ソフトウェアによってセット／クリアされます。STANDBYまたはSHUTDOWNモードを終了するときに、MSISオシレータを強制的に選択するためにハードウェアによって設定されます。このビットフィールドは、STOPモードを終了するとき、またはHSEオシレータに障害が発生した場合、STOPWUCKの値に応じてMSISまたはHSI16オシレータを強制的に選択するためにハードウェアによってセットされます。

00：MSISがシステムクロックとして選択されます。

01：HSI16がシステムクロックとして選択されます。

10：HSEがシステムクロックとして選択されます。

11：PLL pll1_r_ckがシステムクロックとして選択されます。

11.8.7 RCCクロック設定レジスタ2(RCC_CFGR2)

アドレスオフセット：0x020

リセット値：0x0000 0000（STM32U535/545/575/585の場合）

リセット値：0x0000 6000（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ワード、ハーフワード、およびバイトアクセス

APBまたはAHBプリスケーラ値の更新中にアクセスが発生した場合、0～15個のウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APB2DIS	APB1DIS	AHB2DIS2	AHB2DIS1	AHB1DIS
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DPRE[2:0]			Res.	PPRE2 [2:0]			Res.	PPRE1 [2:0]			HPRE[3:0]			
	rw	rw	rw		rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **APB2DIS**：APB2クロックの無効化

このビットは、APB2ペリフェラルを使用しておらず、そのクロックがRCC_APB2ENRで無効にされている場合に、消費電力をさらに低減するためにセットできます。このビットをセットすると、すべてのAPB2ペリフェラルクロックがオフになります。

0 : APB2クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：APB2クロックは無効です。

ビット19 **APB1DIS**：APB1クロックの無効化

このビットは、APB1ペリフェラル（IWDG以外）を使用しておらず、そのクロックがRCC_APB1ENRで無効にされている場合に、消費電力をさらに低減するためにセットできます。このビットをセットすると、IWDG以外のすべてのAPB1ペリフェラルクロックがオフになります。

0 : APB1クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：APB1クロックは無効です。

ビット18 **AHB2DIS2**：AHB2_2クロックの無効化

このビットは、RCC_AHB2ENR2のいずれのAHB2ペリフェラルも使用されておらず、そのクロックがRCC_AHB2ENR2で無効になっている場合に、消費電力をさらに削減するためにセットできます。このビットをセットすると、RCC_AHB2ENR2のすべてのAHB2ペリフェラルクロックがオフになります。

0：AHB2_2クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：AHB2_2クロックは無効です。

ビット17 **AHB2DIS1**：AHB2_1クロックの無効化

このビットは、RCC_AHB2ENR1の（SRAM2およびSRAM3以外の）いずれのAHB2ペリフェラルも使用されておらず、そのクロックがRCC_AHB2ENR1で無効になっている場合に、消費電力をさらに削減するためにセットできます。このビットをセットすると、SRAM2 および SRAM3 以外のRCC_AHB2ENR1のすべてのAHB2ペリフェラルクロックがオフになります。

0：AHB2_1クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：AHB2_1クロックは無効です。

ビット16 **AHB1DIS**：AHB1クロックの無効化

このビットは、AHB1ペリフェラル（以下に示すもの以外）を使用しておらず、そのクロックがRCC_AHB1ENRで無効にされている場合に、消費電力をさらに低減するためにセットできます。このビットをセットすると、FLASH、BKPSRAM、ICACHE、DCACHE1、SRAM1以外のすべてのAHB1ペリフェラルクロックがオフになります。

0：AHB1クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：AHB1クロックは無効です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **DPRE[2:0]**：DSIPHYプリスケーラ

このビットフィールドは、DSIPHYバスクロック（DCLK）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：DCLKは分周されません。

100：DCLKは2分周されます。

101：DCLKは4分周されます。

110：DCLKは8分周されます。

111：DCLKは16分周されます。

注： このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **PPRE2[2:0]**：APB2プリスケーラ

このビットフィールドは、APB2クロック（PCLK2）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：PCLK2は分周されません。

100：PCLK2は2分周されます。

101：PCLK2は4分周されます。

110：PCLK2は8分周されます。

111：PCLK2は16分周されます。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **PPRE1[2:0]**：APB1プリスケーラ

このビットフィールドは、APB1クロック（PCLK1）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：PCLK1は分周されません。

100：PCLK1は2分周されます。

101：PCLK1は4分周されます。

110：PCLK1は8分周されます。

111：PCLK1は16分周されます。

ビット3:0 **HPRE[3:0]**：AHBプリスケーラ

このビットフィールドは、AHBクロック（HCLK）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

注意： デバイスの電圧レンジに応じて、ソフトウェアではこれらのビットを正しく設定し、システム周波数が最大許容周波数を超えないようにします（詳細については、表114を参照）。これらのビットの書込み動作後、電圧レンジを下げる前に、このレジスタを読み出して新しい値が正しく反映されていることを確認する必要があります。

0xxx：SYSCLKは分周されません。
1000：SYSCLKは2分周されます。
1001：SYSCLKは4分周されます。
1010：SYSCLKは8分周されます。
1011：SYSCLKは16分周されます。
1100：SYSCLKは64分周されます。
1101：SYSCLKは128分周されます。
1110：SYSCLKは256分周されます。
1111：SYSCLKは512分周されます。

11.8.8 RCCクロック設定レジスタ3(RCC_CFGR3)

アドレスオフセット：0x024

リセット値：0x0000 0000

アクセス：ワード、ハーフワード、およびバイトアクセス

APBまたはAHBプリスケーラ値の更新中にアクセスが発生した場合、0～15個のウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APB3DIS	AHB3DIS
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPRE3 [2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **APB3DIS**：APB3クロックの無効化

このビットは、RCC_APB3ENRのいずれのAPB3ペリフェラルも使用されておらず、そのクロックがRCC_APB3ENRで無効になっている場合に、消費電力をさらに削減するためにセットできます。このビットをセットすると、すべてのAPB3ペリフェラルクロックがオフになります。

0 : APB3クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：APB3クロックは無効です。

ビット16 **AHB3DIS**：AHB3クロックの無効化

このビットは、(SRAM4以外の)AHB3ペリフェラルを使用しておらず、そのクロックがRCC_AHB3ENRで無効にされている場合に、消費電力をさらに低減するためにセットできます。このビットをセットすると、SRAM4以外のすべてのAHB3ペリフェラルクロックがオフになります。

0：AHB3クロックは有効であり、専用クロックイネーブル制御ビットに従ってペリフェラルに配分されます。

1：AHB3クロックは無効です。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **PPRE3[2:0]**：APB3プリスケーラ

このビットフィールドは、APB3クロック（PCLK3）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：HCLKは分周されません。

100：HCLKは2分周されます。

101：HCLKは4分周されます。

110：HCLKは8分周されます。

111：HCLKは16分周されます。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

11.8.9 RCCPLL1設定レジスタ（RCC_PLL1CFGR）

アドレスオフセット：0x028

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL1REN	PLL1QEN	PLL1PEN
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL1MBOOST [3:0]				PLL1M [3:0]				Res.	Res.	Res.	PLL1FRACEN	PLL1RGE [1:0]		PLL1SRC [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **PLL1REN**：PLL1のDIVR分周器出力の有効化

このビットは、PLL1のpll1_r_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll1_r_ckが使用されないときには、PLL1RENPLL2RENおよびPLL1Rビットを0にセットする必要があります。このビットをクリアできるのは、PLL1がSYSCLKとして使用されていないときだけです。

0：pll1_r_ck出力は無効です。

1：pll1_r_ck出力は有効です。

ビット17 **PLL1QEN**：PLL1のDIVQ分周器出力の有効化

このビットは、PLL1のpll1_q_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll1_q_ckが使用されないときには、PLL1QENおよびPLL1Qビットを0にセットする必要があります。

0：pll1_q_ck出力は無効です。

1：pll1_q_ck出力は有効です。

ビット16 **PLL1PEN**：PLL1のDIVP分周器出力の有効化

このビットは、PLL1のpll1_p_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll1_p_ckが使用されないときには、PLL1PENおよびPLL1Pビットを0にセットする必要があります。

0：pll1_p_ck出力は無効です。

1：pll1_p_ck出力は有効です。

ビット15:12 **PLL1MBOOST[3:0]**：EPODブースタの入力クロック用プリスケーラ

このビットフィールドは、EPODブースタに使用されるPLL1のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。EPODブースタの入力周波数は、PLL1入力クロック周波数/PLL1MBOOSTです。

このビットに書き込めるのは、PLL1が無効で（PLL1ON = 0かつPLL1RDY = 0）、EPODboostモードが無効なときだけです（セクション 10：電源制御（PWR）を参照）。

0000：分周比1（バイパス）

0001：分周比2

0010：分周比4

0011：分周比6

0100：分周比8

0101：分周比10

0110：分周比12

0111：分周比14

1000：分周比16

その他：予約済み

ビット11:8 **PLL1M[3:0]**：PLL1のプリスケーラ

このビットフィールドは、PLL1のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。VCO1の入力周波数は、PLL1入力クロック周波数/PLL1Mです。

このビットに書き込めるのは、PLL1が無効なときだけです（PLL1ON=0およびPLL1RDY=0）。

0000：分周比1（バイパス）

0001：分周比2

0010：分周比3

...

1111：分周比16

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **PLL1FRACEN**：PLL1分数ラッチイネーブル

このビットは、PLL1FRACNの内容をΣΔモジュレータにラッチするために、ソフトウェアによってセット／リセットされますΣΔモジュレータにPLL1FRACNの値をラッチするには、PLL1FRACENを「0」に設定し、次に「1」にセットする必要があります。0から1への遷移により、モジュレータにPLL1FRACNの内容が転送されます（詳細は、PLL初期化フェーズを参照）。

ビット3:2 **PLL1RGE[1:0]**：PLL1入力周波数範囲

このビットは、PLL1で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。PLL1を有効にする前に書き込む必要があります。

00-01-10：PLL1入力（ref1_ck）クロック範囲の周波数は、4から8MHzです。

11：PLL1入力（ref1_ck）クロック範囲の周波数は、8から16MHzです。

ビット1:0 **PLL1SRC[1:0]**：PLL1入力クロックソース

このビットフィールドは、PLL1クロックソースを選択するために、ソフトウェアによってセット／クリアされます。ここに書き込めるのは、PLL1が無効なときだけです。電力を削減するために、PLL1を使用しない場合は、このビットフィールドの値をゼロにする必要があります。

00：PLL1にクロックが送信されません。

01：MSISクロックがPLL1クロック入力として選択されます。

10：HSI16クロックがPLL1クロック入力として選択されます。

11：HSEクロックがPLL1クロック入力として選択されます。

11.8.10 RCCPLL2設定レジスタ（RCC_PLL2CFGR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL2REN	PLL2QEN	PLL2PEN
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PLL2M [3:0]				Res.	Res.	Res.	PLL2FRACEN	PLL2RGE [1:0]		PLL2SRC [1:0]	
				rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **PLL2REN**：PLL2のDIVR分周器出力の有効化

このビットは、PLL2のpll2_r_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll2_r_ckが使用されないときには、PLL2RENおよびPLL2Rビットを0にセットする必要があります。

0：pll2_r_ck出力は無効です。

1：pll2_r_ck出力は有効です。

ビット17 **PLL2QEN**：PLL2のDIVQ分周器出力の有効化

このビットは、PLL2のpll2_q_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll2_q_ckが使用されないときには、PLL2QENおよびPLL2Qビットを0にセットする必要があります。

0：pll2_q_ck出力は無効です。

1：pll2_q_ck出力は有効です。

ビット16 **PLL2PEN**：PLL2のDIVP分周器出力の有効化

このビットは、PLL2のpll2_p_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll2_p_ckが使用されないときには、PLL2PENおよびPLL2Pビットを0にセットする必要があります。

0：pll2_p_ck出力は無効です。

1：pll2_p_ck出力は有効です。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **PLL2M[3:0]**：PLL2のプリスケーラ

このビットフィールドは、PLL2のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。VCO2の入力周波数は、PLL2入力クロック周波数/PLL2Mです。

このビットに書き込めるのは、PLL2が無効なときだけです（PLL2ON = 0およびPLL2RDY = 0）。

0000：分周比1（バイパス）

0001：分周比2

0010：分周比3

...

1111：分周比16

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **PLL2FRACEN**：PLL2分数ラッチイネーブル

このビットは、PLL2FRACNの内容をΣΔモジュレータにラッチするために、ソフトウェアによってセット／リセットされますΣΔモジュレータにPLL2FRACNの値をラッチするには、PLL2FRACENを「0」に設定し、次に「1」にセットする必要があります。0から1への遷移により、モジュレータにPLL2FRACNの内容が転送されます（詳細は、PLL初期化フェーズを参照）。

ビット3:2 **PLL2RGE[1:0]**：PLL2入力周波数範囲

このビットフィールドは、PLL2で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。PLL2を有効にする前に書き込む必要があります。

00-01-10：PLL2入力（ref2_ck）クロック範囲の周波数は、4から8MHzです。

11：PLL2入力（ref2_ck）クロック範囲の周波数は、8から16MHzです。

ビット1:0 **PLL2SRC[1:0]**：PLL2入力クロックソース

このビットフィールドは、PLL2クロックソースを選択するために、ソフトウェアによってセット／クリアされます。ここに書き込めるのは、PLL2が無効なときだけです。電力を削減するために、PLL2を使用しない場合は、このビットフィールドの値をゼロにする必要があります。

00：PLL2にクロックが送信されません。

01：MSISクロックがPLL2クロック入力として選択されます。

10：HSI16クロックがPLL2クロック入力として選択されます。

11：HSEクロックがPLL2クロック入力として選択されます。

11.8.11 RCC PLL3設定レジスタ（RCC_PLL3CFGR）

アドレスオフセット：0x030

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL3REN	PLL3QEN	PLL3PEN
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PLL3M [3:0]				Res.	Res.	Res.	PLL3FRACEN	PLL3RGE [1:0]		PLL3SRC [1:0]	
				rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **PLL3REN**：PLL3のDIVR分周器出力の有効化

このビットは、PLL3のpll3_r_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll3_r_ckが使用されないときには、PLL3RENおよびPLL3Rビットを0にセットする必要があります。

0：pll3_r_ck出力は無効です。

1：pll3_r_ck出力は有効です。

ビット17 **PLL3QEN**：PLL3のDIVQ分周器出力の有効化

このビットは、PLL3のpll3_q_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll3_q_ckが使用されないときには、PLL3QENおよびPLL3Qビットを0にセットする必要があります。

0：pll3_q_ck出力は無効です。

1：pll3_q_ck出力は有効です。

ビット16 **PLL3PEN**：PLL3のDIVP分周器出力の有効化

このビットは、PLL3のpll3_p_ck出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、pll3_p_ckが使用されないときには、PLL3PENおよびPLL3Pビットを0にセットする必要があります。

0：pll3_p_ck出力は無効です。

1：pll3_p_ck出力は有効です。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **PLL3M[3:0]**：PLL3のプリスケーラ

このビットフィールドは、PLL3のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。VCO3の入力周波数は、PLL3入力クロック周波数/PLL3Mです。このビットフィールドに書き込めるのは、PLL3が無効のときだけです（PLL3ON = 0かつPLL3RDY = 0）。

0000：分周比1（バイパス）

0001：分周比2

0010：分周比3

...

1111：分周比16

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **PLL3FRACEN**：PLL3分数ラッチイネーブル

このビットは、PLL3FRACNの内容をΣΔモジュレータにラッチするために、ソフトウェアによってセット／リセットされますΣΔモジュレータにPLL3FRACNの値をラッチするには、PLL3FRACENを「0」に設定し、次に「1」にセットする必要があります。0から1への遷移により、モジュレータにPLL3FRACNの内容が転送されます（詳細は、PLL初期化フェーズを参照）。

ビット3:2 **PLL3RGE[1:0]**：PLL3入力周波数範囲

このビットは、PLL3で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。PLL3を有効にする前に書き込む必要があります。

00-01-10：PLL3入力（ref3_ck）クロック範囲の周波数は、4から8MHzです。

11：PLL3入力（ref3_ck）クロック範囲の周波数は、8から16MHzです。

ビット1:0 **PLL3SRC[1:0]**：PLL3入力クロックソース

このビットフィールドは、PLL3クロックソースを選択するために、ソフトウェアによってセット／クリアされます。ここに書き込めるのは、PLL3が無効なときだけです。電力を削減するために、PLL3を使用しない場合は、このビットフィールドの値をゼロにする必要があります。

00：PLL3にクロックが送信されません。

01：MSISクロックがPLL3クロック入力として選択されます。

10：HSI16クロックがPLL3クロック入力として選択されます。

11：HSEクロックがPLL3クロック入力として選択されます。

11.8.12 RCCPLL1分周器レジスタ（RCC_PLL1DIVR）

アドレスオフセット：0x034

リセット値：0x0101 0280

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	PLL1R [6:0]							Res.	PLL1Q [6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL1P [6:0]								PLL1N [8:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:24 **PLL1R[6:0]**：PLL1DIVRの分周比

このビットフィールドは、pll1_r_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL1が無効なときだけです（PLL1ON = 0かつPLL1RDY = 0）。1の分周と偶数の分周比のみが使用できます。

0000000：pll1_r_ck = vco1_ck

0000001：pll1_r_ck = vco1_ck/2（リセット後のデフォルト）

0000010：予約済み

0000011：pll1_r_ck = vco1_ck/4

...

1111111：pll1_r_ck = vco1_ck/128

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **PLL1Q[6:0]**：PLL1DIVQの分周比

このビットフィールドは、pll1_q_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL1が無効なときだけです（PLL1ON = 0かつPLL1RDY = 0）。

0000000: pll1_q_ck = vco1_ck

0000001：pll1_r_ck =vco1_ck/2（リセット後のデフォルト）

0000010：pll1_q_ck =vco1_ck/3

0000011：pll1_q_ck = vco1_ck/4

...

1111111：pll1_q_ck = vco1_ck/128

ビット15:9 **PLL1P[6:0]**：PLL1 DIVPの分周比

このビットフィールドは、pll1_p_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL1が無効なときだけです（PLL1ON = 0かつPLL1RDY = 0）。

0000000：pll1_p_ck =vco1_ck

0000001：pll1_r_ck = vco1_ck/2（リセット後のデフォルト）

0000010：pll1_p_ck = vco1_ck/3

0000011：pll1_p_ck = vco1_ck/4

...

1111111：pll1_p_ck = vco1_ck/128

ビット8:0 **PLL1N[8:0]**：PLL1VCOの逓倍数

このビットフィールドは、VCOの逓倍数を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドに書き込めるのは、PLLが無効なときだけです（PLL1ON = 0かつPLL1RDY = 0）。

0x003：PLL1N =4

0x004：PLL1N = 5

0x005：PLL1N = 6

...

0x080：PLL1N = 129（リセット後のデフォルト）

...

0x1FF：PLL1N = 512

その他：予約済み

分数値0が次の条件でPLL1FRACNにロードされた場合、VCO出力周波数 = F_{ref1_ck}xPLL1Nです。

– 4から512の間のPLL1N

– 4から16MHzの間の入力周波数F_{ref1_ck}

11.8.13 RCC PLL1小数分周器レジスタ（RCC_PLL1FRACR）

アドレスオフセット：0x038

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードおよびハーフワードアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL1FRACN [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:3 **PLL1FRACN[12:0]**：PLL1 VCOの逓倍数の小数部

このビットフィールドは、VCO逓倍数の少数部を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドはいつでも書き込めるため、PLL1 VCOを動的に微調整することができます。VCO出力周波数 = F_{ref1_ck}x$\left(\text{PLL1N} + (\text{PLL1FRACN} / 2^{13})\right)$で、次の要件を満たすものとします。

– PLL1Nが4から512の間であること

– PLL1FRACNが0から2^{13}-1の間であること

– 入力周波数F_{ref1_ck}が4から16MHzの間であること

PLLが有効である場合でも、動作中にFRACNの値を変更するには、アプリケーションで次の手順に従う必要があります。

– PLL1FRACEN = 0にセットします。

– PLL1FRACNに新しい分数値を書き込みます。

– PLL1FRACEN = 1にセットします。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

11.8.14 RCCPLL2分周器設定レジスタ（RCC_PLL2DIVR）

アドレスオフセット：0x03C

リセット値：0x0101 0280

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	PLL2R [6:0]							Res.	PLL2Q [6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL2P [6:0]								PLL2N [8:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:24 **PLL2R[6:0]**：PLL2 DIVRの分周比

このビットフィールドは、pll2_r_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL2が無効なときだけです（PLL2ON = 0かつPLL2RDY = 0）。

0000000：pll2_r_ck = vco2_ck

0000001：pll2_r_ck = vco2_ck/2（リセット後のデフォルト）

0000010：pll2_r_ck = vco2_ck/3

0000011：pll2_r_ck = vco2_ck/4

...

1111111：pll2_r_ck = vco2_ck/128

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **PLL2Q[6:0]**：PLL2 DIVQの分周比

このビットフィールドは、pll2_q_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL2が無効なときだけです（PLL2ON = 0かつPLL2RDY = 0）。

0000000：pll2_q_ck = vco2_ck

0000001：pll2_q_ck = vco2_ck/2（リセット後のデフォルト）

0000010：pll2_q_ck = vco2_ck/3

0000011：pll2_q_ck = vco2_ck/4

...

1111111：pll2_q_ck = vco2_ck/128

ビット15:9 **PLL2P[6:0]**：PLL2DIVPの分周比

このビットフィールドは、pll2_p_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL2が無効なときだけです（PLL2ON = 0かつPLL2RDY = 0）。

0000000：pll2_p_ck = vco2_ck

0000001：pll2_p_ck = vco2_ck/2（リセット後のデフォルト）

0000010：pll2_p_ck = vco2_ck/3

0000011：pll2_p_ck = vco2_ck/4

...

1111111：pll2_p_ck = vco2_ck/128

ビット8:0 **PLL2N[8:0]**：PLL2 VCOの逓倍数

このビットフィールドは、VCOの逓倍数を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドに書き込めるのは、PLLが無効なときだけです（PLL2ON = 0かつPLL2RDY = 0）。

0x003：PLL2N =4

0x004：PLL2N = 5

0x005：PLL2N = 6

...

0x080：PLL2N = 129（リセット後のデフォルト）

...

0x1FF：PLL2N = 512

その他：予約済み

分数値0が次の条件でPLL2FRACNにロードされた場合、VCO出力周波数 = F_{ref2_ck}xPLL2Nです。

- 4から512の間のPLL2N
- 1から16MHzの間の入力周波数F_{ref2_ck}

11.8.15 RCCPLL2小数分周器レジスタ（RCC_PLL2FRACR）

アドレスオフセット：0x040

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードおよびハーフワードアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL2FRACN [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:3 **PLL2FRACN[12:0]**：PLL2 VCOの逓倍数の小数部

このビットフィールドは、VCO逓倍数の少数部を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドはいつでも書き込めるため、PLL2 VCOを動的に微調整することができます。

VCO出力周波数 = F_{ref2_ck}x$\left(\text{PLL2N} + (\text{PLL2FRACN} / 2^{13})\right)$で、次の要件を満たすものとします。

- PLL2Nが4から512の間であること
- PLL2FRACNが0から2^{13}-1の間であること
- 入力周波数F_{ref2_ck}が4から16MHzの間であること

PLLが有効である場合でも、動作中にFRACNの値を変更するには、アプリケーションで次の手順に従う必要があります。

- PLL2FRACENビットを0にセットします。
- PLL2FRACNに新しい分数値を書き込みます。
- PLL2FRACENビットを1にセットします。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

11.8.16 RCCPLL3分周器設定レジスタ（RCC_PLL3DIVR）

アドレスオフセット：0x044

リセット値：0x0101 0280

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	PLL3R [6:0]							Res.	PLL3Q [6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL3P [6:0]								PLL3N [8:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:24 **PLL3R[6:0]**：PLL3DIVRの分周比

このビットフィールドは、pll3_r_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL3が無効なときだけです（PLL3ON = 0かつPLL3RDY = 0）。

0000000：pll3_r_ck = vco3_ck

0000001：pll3_r_ck = vco3_ck/2（リセット後のデフォルト）

0000010：pll3_r_ck = vco3_ck/3

0000011：pll3_r_ck = vco3_ck/4

...

1111111：pll3_r_ck = vco3_ck/128

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **PLL3Q[6:0]**：PLL3DIVQの分周比

このビットフィールドは、pll3_q_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL3が無効なときだけです（PLL3ON = 0かつPLL3RDY = 0）。

0000000：pll3_q_ck = vco3_ck

0000001：pll3_q_ck = vco3_ck/2（リセット後のデフォルト）

0000010：pll3_q_ck = vco3_ck/3

0000011：pll3_q_ck = vco3_ck/4

...

1111111：pll3_q_ck = vco3_ck/128

ビット15:9 **PLL3P[6:0]**：PLL3 DIVPの分周比

このビットフィールドは、pll3_p_ckクロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。このビットに書き込めるのは、PLL3が無効なときだけです（PLL3ON = 0かつPLL3RDY = 0）。

0000000：pll3_p_ck = vco3_ck

0000001：pll3_p_ck = vco3_ck/2（リセット後のデフォルト）

0000010：pll3_p_ck = vco3_ck/3

0000011：pll3_p_ck = vco3_ck/4

...

1111111：pll3_p_ck = vco3_ck/128

ビット8:0 **PLL3N[8:0]**：PLL3VCOの逓倍数

このビットフィールドは、VCOの逓倍数を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドに書き込めるのは、PLLが無効なときだけです（PLL3ON = 0かつPLL3RDY = 0）。

0x003：PLL3N =4

0x004：PLL3N = 5

0x005：PLL3N = 6

...

0x080：PLL3N = 129（リセット後のデフォルト）

...

0x1FF：PLL3N = 512

その他：予約済み

分数値0が次の条件でPLL3FRACNにロードされた場合、VCO出力周波数 = F_{ref3_ck}xPLL3Nです。

- 4から512の間のPLL3N
- 4から16MHzの間の入力周波数F_{ref3_ck}

11.8.17 RCC PLL3小数分周器レジスタ（RCC_PLL3FRACR）

アドレスオフセット：0x048

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードおよびハーフワードアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLL3FRACN [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:3 **PLL3FRACN[12:0]**：PLL3 VCOの逓倍数の小数部

このビットフィールドは、VCO逓倍数の少数部を制御するために、ソフトウェアによってセット／リセットされます。このビットフィールドはいつでも書き込めるため、PLL3 VCOを動的に微調整することができます。

VCO出力周波数 = F_{ref3_ck}x$\left(\text{PLL3N} + (\text{PLL3FRACN} / 2^{13})\right)$で、次の要件を満たすものとします。

- PLL3Nが4から512の間であること
- PLL3FRACNが0から2^{13}-1の間であること
- 入力周波数F_{ref3_ck}が4から16MHzの間であること

PLLが有効である場合でも、動作中にFRACNの値を変更するには、アプリケーションで次の手順に従う必要があります。

- PLL3FRACENビットを0にセットします。
- PLL3FRACNに新しい分数値を書き込みます。
- PLL3FRACENビットを1にセットします。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

11.8.18 RCCクロック割込み有効化レジスタ（RCC_CIER）

アドレスオフセット：0x050

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	SHSIRDYIE	MSIKRDYIE	Res.	Res.	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	HSERDYIE	HSIRDYIE	MSISRDYIE	LSERDYIE	LSIRDYIE
			rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **SHSIRDYIE**：SHSIレディ割込みの有効化

このビットは、SHSIオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：SHSIレディ割込みは無効です。

1：SHSIレディ割込みは有効です。

ビット11 **MSIKRDYIE**：MSIKレディ割込みの有効化

このビットは、MSIKオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：MSIKレディ割込みは無効です。

1：MSIKレディ割込みは有効です。

ビット10:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PLL3RDYIE**：PLL3レディ割込みの有効化

このビットは、PLL3ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLL3ロック割込みは無効です。

1：PLL3ロック割込みは有効です。

ビット7 **PLL2RDYIE**：PLL2レディ割込みの有効化

このビットは、PLL2ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLL2ロック割込みが無効です。

1：PLL2ロック割込みは有効です。

ビット6 **PLL1RDYIE**：PLLレディ割込みの有効化

このビットは、PLL1ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLL1ロック割込みは無効です。

1：PLL1ロック割込みは有効です。

ビット5 **HSI48RDYIE**：HSI48レディ割込みの有効化

このビットは、HSI48オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI48レディ割込みは無効です。

1：HSI48レディ割込みは有効です。

ビット4 **HSERDYIE**HSEレディ割込みの有効化

このビットは、HSEオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSEレディ割込みは無効です。

1：HSEレディ割込みは有効です。

ビット3 **HSIRDYIE**：HSI16レディ割込みの有効化

このビットは、HSI16オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI16レディ割込みは無効です。

1：HSI16レディ割込みは有効です。

ビット2 **MSISRDYIE**：MSISレディ割込みの有効化

このビットは、MSISオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：MSISレディ割込みは無効です。

1：MSISレディ割込みは有効です。

ビット1 **LSERDYIE**：LSEレディ割込みの有効化

このビットは、LSEオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSEレディ割込みは無効です。

1：LSEレディ割込みは有効です。

ビット0 **LSIRDYIE**LSIレディ割込みの有効化

このビットは、LSIオシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSIレディ割込みは無効です。

1：LSIレディ割込みは有効です。

11.8.19 RCCクロック割込みフラグレジスタ（RCC_CIFR）

アドレスオフセット：0x054

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	SHSIRDYF	MSIKRDYF	CSSF	Res.	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	HSERDYF	HSIRDYF	MSISRDYF	LSERDYF	LSIRDYF
			r	r	r		r	r	r	r	r	r	r	r	r

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **SHSIRDYF**：SHSIレディ割込みフラグ

このビットは、SHSIクロックが安定し、SHSIRDYIEビットがセットされているときに、ハードウェアによってセットされます。SHSIRDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：SHSIオシレータによるクロックレディ割込みは発生していません。

1：SHSIオシレータによるクロックレディ割込みが発生しました。

ビット11 **MSIKRDYF**：MSIKレディ割込みフラグ

このビットは、MSIKクロックが安定し、MSIKRDYIEビットがセットされているときに、ハードウェアによってセットされます。MSIKRDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：MSIKオシレータによるクロックレディ割込みは発生していません。

1：MSIKオシレータによるクロックレディ割込みが発生しました。

ビット10 **CSSF**：クロックセキュリティシステム割込みフラグ

このビットは、HSEオシレータで障害が検出されたときに、ハードウェアによってセットされます。CSSCビットをセットすることによって、ソフトウェアによってクリアされます。

0：HSEクロック障害によるクロックセキュリティ割込みは発生していません。

1：HSEクロック障害によるクロックセキュリティ割込みが発生しました。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 PLL3RDYF：PLL3レディ割込みフラグ

このビットは、PLL3がロックされ、PLL3RDYIEがセットされているときに、ハードウェアによってセットされます。PLL3RDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：PLL3ロックによるクロックレディ割込みは発生していません。

1：PLL3ロックによるクロックレディ割込みが発生しました。

ビット7 PLL2RDYF：PLL2レディ割込みフラグ

このビットは、PLL2がロックされ、PLL2RDYIEがセットされているときに、ハードウェアによってセットされます。PLL2RDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：PLL2ロックによるクロックレディ割込みは発生していません。

1：PLL2ロックによるクロックレディ割込みが発生しました。

ビット6 PLL1RDYF：PLL1レディ割込みフラグ

このビットは、PLL1がロックされ、PLL1RDYIEがセットされているときに、ハードウェアによってセットされます。PLL1RDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：PLL1ロックによるクロックレディ割込みは発生していません。

1：PLL1ロックによるクロックレディ割込みが発生しました。

ビット5 **HSI48RDYF**：HSI48レディ割込みフラグ

このビットは、HSI48クロックが安定し、HSI48RDYIEビットがセットされているときに、ハードウェアによってセットされます。HSI48RDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：HSI48オシレータによるクロックレディ割込みは発生していません。

1：HSI48オシレータによるクロックレディ割込みが発生しました。

ビット4 **HSERDYF**：HSEレディ割込みフラグ

このビットは、HSEクロックが安定し、HSERDYIEビットがセットされているときに、ハードウェアによってセットされます。HSERDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：HSEオシレータによるクロックレディ割込みは発生していません。

1：HSEオシレータによるクロックレディ割込みが発生しました。

ビット3 **HSIRDYF**：HSI16レディ割込みフラグ

このビットは、HSI16クロックが安定し、HSIONの設定に応じてHSIRDYIE = 1がセットされたときに、ハードウェアによってセットされます（RCC_CRを参照）。HSION = 0で、ただしクロックリクエストを通じてペリフェラルによってHSI16オシレータが有効化された場合、このビットはセットされず、割込みも生成されません。このビットは、HSIRDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：HSI16オシレータによるクロックレディ割込みは発生していません。

1：HSI16オシレータによるクロックレディ割込みが発生しました。

ビット2 **MSISRDYF**：MSISレディ割込みフラグ

このビットは、MSISクロックが安定し、MSISRDYIEビットがセットされているときに、ハードウェアによってセットされます。MSISRDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：MSISオシレータによるクロックレディ割込みは発生していません。

1：MSISオシレータによるクロックレディ割込みが発生しました。

ビット1 **LSERDYF**：LSEレディ割込みフラグ

このビットは、LSEクロックが安定し、LSERDYIEビットがセットされているときに、ハードウェアによってセットされます。LSERDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：LSEオシレータによるクロックレディ割込みは発生していません。

1：LSEオシレータによるクロックレディ割込みが発生しました。

ビット0 **LSIRDYF**：LSIレディ割込みフラグ

このビットは、LSIクロックが安定し、LSIRDYIEビットがセットされているときに、ハードウェアによってセットされます。LSIRDYCビットをセットすることによって、ソフトウェアによってクリアされます。

0：LSIオシレータによるクロックレディ割込みは発生していません。

1：LSIオシレータによるクロックレディ割込みが発生しました。

11.8.20 RCCクロック割込みクリアレジスタ（RCC_CICR）

アドレスオフセット：0x058

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	SHSIRDYC	MSIKRDYC	CSSC	Res.	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	HSERDYC	HSIRDYC	MSISRDYC	LSERDYC	LSIRDYC
			w	w	w		w	w	w	w	w	w	w	w	w

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **SHSIRDYC**：SHSIオシレータレディ割込みのクリア

このビットに1を書き込むと、SHSIRDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット11 **MSIKRDYC**：MSIKオシレータレディ割込みのクリア

このビットに1を書き込むと、MSIKRDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット10 **CSSC**：クロックセキュリティシステム割込みのクリア

このビットに1を書き込むと、CSSFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PLL3RDYC**：PLL3レディ割込みのクリア

このビットに1を書き込むと、PLL3RDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット7 **PLL2RDYC**：PLL2レディ割込みのクリア

このビットに1を書き込むと、PLL2RDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット6 **PLL1RDYC**：PLL1レディ割込みのクリア

このビットに1を書き込むと、PLL1RDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット5 **HSI48RDYC**：HSI48レディ割込みのクリア

このビットに1を書き込むと、HSI48RDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット4 **HSERDYC**：HSEレディ割込みのクリア

このビットに1を書き込むと、HSERDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット3 **HSIRDYC**：HSI16レディ割込みのクリア

このビットに1を書き込むと、HSIRDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット2 **MSISRDYC**：MSISレディ割込みのクリア

このビットに1を書き込むと、MSISRDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット1 **LSERDYC**：LSEレディ割込みのクリア

このビットに1を書き込むと、LSERDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

ビット0 **LSIRDYC**：LSIレディ割込みのクリア

このビットに1を書き込むと、LSIRDYFフラグがクリアされます。0を書き込んでも、ビットの値は変化しません。

11.8.21 RCCAHB1ペリフェラルリセットレジスタ（RCC_AHB1RSTR）

アドレスオフセット：0x060

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPU2DRST	GFXMMURST	DMA2DRST	RAMCFGRST	TSCRST
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JPEGRST	Res.	Res.	CRCRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MDF1RST	FMACRST	CORDICRST	GPDMA1RST
rw			rw									rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **GPU2DRST**：GPU2Dのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：GPU2Dをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **GFXMMURST**：GFXMMUのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：GFXMMUをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **DMA2DRST**：DMA2Dのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：DMA2Dをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット17 **RAMCFGRST**：RAMCFGのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：RAMCFGをリセットします。

ビット16 **TSCRST**：TSCのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TSCをリセットします。

ビット15 **JPEGRST**：JPEGのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：JPEGをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CRCRST**：CRCのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：CRCをリセットします。

ビット11:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **MDF1RST**：MDF1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：MDF1をリセットします。

ビット2 **FMACRST**：FMACのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：FMACをリセットします。

ビット1 **CORDICRST**：CORDICのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：CORDICをリセットします。

ビット0 **GPDMA1RST**：GPDMA1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：GPDMA1をリセットします。

11.8.22 RCC AHB2ペリフェラルリセットレジスタ1（RCC_AHB2RSTR1）

アドレスオフセット：0x064

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SDMMC2RST	SDMMC1RST	Res.	Res.	OTFDEC2RST	OTFDEC1RST	Res.	OCTOSPIMRST	SAESRST	PKARST	RNGRST	HASHRST	AESRST
			rw	rw			rw	rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OTGRST	Res.	DCMI_PSSIRST	Res.	ADC12RST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	rw		rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **SDMMC2RST**：SDMMC2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SDMMC2をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット27 **SDMMC1RST**：SDMMC1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SDMMC1をリセットします。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **OTFDEC2RST**：OTFDEC2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：OTFDEC2をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 **OTFDEC1RST**：OTFDEC1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：OTFDEC1をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OCTOSPIMRST**：OCTOSPIMのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：OCTOSPIMをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット20 **SAESRST**：SAESハードウェアアクセラレータのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SAESをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **PKARST**：PKAのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：PKAをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **RNGRST**：RNGのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：RNGをリセットします。

ビット17 **HASHRST**：HASHのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：HASHをリセットします。

ビット16 **AESRST**：AESハードウェアアクセラレータのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：AESをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **OTGRST**：OTG_FSまたはOTG_HSのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：OTG_FSまたはOTG_HSをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DCMI_PSSIRST**：DCMIおよびPSSIのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：DCMIおよびPSSIをリセットします。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADC12RST**：ADC1およびADC2のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：ADC1およびADC2をリセットします。
注： このビットは、STM32U535/545/575/585のADC1と、STM32U59x/5Ax/5Fx/5GxのADC1/ADC2に影響します。

ビット9 **GPIOJRST**：I/OポートJのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートJをリセットします。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット8 **GPIOIRST**：I/OポートIのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートIをリセットします。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7 **GPIOHRST**：I/OポートHのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートHをリセットします。

ビット6 **GPIOGRST**：I/OポートGのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートGをリセットします。

ビット5 **GPIOFRST**：I/OポートFのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートFをリセットします。
注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。
これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット4 **GPIOERST**：I/OポートEのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートEをリセットします。

ビット3 **GPIODRST**：I/OポートDのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートDをリセットします。

ビット2 **GPIOCRST**：I/OポートCのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートCをリセットします。

ビット1 **GPIOBRST**：I/OポートBのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートBをリセットします。

ビット0 **GPIOARST**：I/OポートAのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I/OポートAをリセットします。

11.8.23 RCC AHB2ペリフェラルリセットレジスタ2（RCC_AHB2RSTR2）

アドレスオフセット：0x068

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	HSPI1RST	Res.	Res.	Res.	OCTOSPI2RST	Res.	Res.	Res.	OCTOSPI1RST	Res.	Res.	Res.	FSMCRST
			rw				rw				rw				rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **HSPI1RST**：HSPI1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：HSPI1をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **OCTOSPI2RST**：OCTOSPI2のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：OCTOSPI2をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OCTOSPI1RST**：OCTOSPI1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：OCTOSPI1をリセットします。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **FSMCRST**：フレキシブルメモリコントローラのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：FSMCをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

11.8.24 RCC AHB3ペリフェラルリセットレジスタ（RCC_AHB3RSTR）

アドレスオフセット：0x06C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	ADF1RST	LPDMA1RST	Res.	Res.	DAC1RST	ADC4RST	Res.	Res.	Res.	Res.	LPGPIO1RST
					rw	rw			rw	rw					rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADF1RST**：ADF1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：ADF1をリセットします。

ビット9 **LPDMA1RST**：LPDMA1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：LPDMA1をリセットします。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **DAC1RST**：DAC1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：DAC1をリセットします。

ビット5 **ADC4RST**：ADC4のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：ADC4インタフェースをリセットします。

ビット4:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **LPGPIO1RST**：LPGPIO1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：LPGPIO1をリセットします。

11.8.25 RCCAPB1ペリフェラルリセットレジスタ1（RCC_APB1RSTR1）

アドレスオフセット：0x074

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	USART6RST	CRSRST	Res.	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	Res.
						rw	rw		rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI2RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
	rw									rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **USART6RST**：USART6のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：USART6をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **CRSRST**：CRSのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：CRSをリセットします。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **I2C2RST**：I2C2のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I2C2をリセットします。

ビット21 **I2C1RST**：I2C1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：I2C1をリセットします。

ビット20 **UART5RST**：UART5のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：UART5をリセットします。

ビット19 **UART4RST**：UART4のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：UART4をリセットします。

ビット18 **USART3RST**：USART3のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：USART3をリセットします。

ビット17 **USART2RST**：USART2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：USART2をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット16:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **SPI2RST**：SPI2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SPI2をリセットします。

ビット13:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **TIM7RST**：TIM7のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM7をリセットします。

ビット4 **TIM6RST**：TIM6のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM6をリセットします。

ビット3 **TIM5RST**：TIM5のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM5をリセットします。

ビット2 **TIM4RST**：TIM4のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM4をリセットします。

ビット1 **TIM3RST**：TIM3のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM3をリセットします。

ビット0 **TIM2RST**：TIM2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM2をリセットします。

11.8.26 RCC APB1ペリフェラルリセットレジスタ2（RCC_APB1RSTR2）

アドレスオフセット：0x078

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.
								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1RST	Res.	I2C6RST	I2C5RST	LPTIM2RST	Res.	Res.	Res.	I2C4RST	Res.
						rw		rw	rw	rw				rw	

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **UCPD1RST**：UCPD1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：UCPD1をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **FDCAN1RST**：FDCAN1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：FDCAN1をリセットします。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C6RST**：I2C6のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし

1：I2C6をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット6 **I2C5RST**：I2C5のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし

1：I2C5をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット5 **LPTIM2RST**：LPTIM2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：LPTIM2をリセットします。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **I2C4RST**：I2C4のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし

1：I2C4をリセットします。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.27 RCCAPB2ペリフェラルリセットレジスタ（RCC_APB2RSTR）

アドレスオフセット：0x07C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DSIRST	LTDCRST	GFXTIMRST	USBRST	Res.	SAI2RST	SAI1RST	Res.	Res.	TIM17RST	TIM16RST	TIM15RST
				rw	rw	rw	rw		rw	rw			rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1RST	TIM8RST	SPI1RST	TIM1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw											

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DSIRST**：DSIのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：DSIをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット26 **LTDCRST**：LTDCのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：LTDCをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット25 **GFXTIMRST**：GFXTIMのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：GFXTIMをリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **USBRST**：USBのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：USBをリセットします。

注： このビットはSTM32U535/545デバイスでのみ使用できます。これは、STM32U5シリーズの他のデバイスで予約済みです。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **SAI2RST**：SAI2のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SAI2をリセットします。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット21 **SAI1RST**：SAI1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SAI1をリセットします。

ビット20:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **TIM17RST**：TIM17のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM17をリセットします。

ビット17 **TIM16RST**：TIM16のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM16をリセットします。

ビット16 **TIM15RST**：TIM15のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM15をリセットします。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **USART1RST**：USART1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：USART1をリセットします。

ビット13 **TIM8RST**：TIM8のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：TIM8をリセットします。

ビット12 **SPI1RST**：SPI1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SPI1をリセットします。

ビット11 **TIM1RST**：TIM1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：TIM1をリセットします。

ビット10:0 予約済みであり、リセット値に保持する必要があります。

11.8.28 RCC APB3ペリフェラルリセットレジスタ（RCC_APB3RSTR）

アドレスオフセット：0x080

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VREFRST	Res.	Res.	Res.	Res.
											rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMPRST	OPAMPRST	LPTIM4RST	LPTIM3RST	LPTIM1RST	Res.	Res.	Res.	I2C3RST	LPUART1RST	SPI3RST	Res.	Res.	Res.	SYSCFGRST	Res.
rw	rw	rw	rw	rw				rw	rw	rw				rw	

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **VREFRST**：VREFBUFのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：VREFBUFをリセットします。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **COMPRST**：COMPのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：COMPをリセットします。

ビット14 **OPAMPRST**：OPAMPのリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：OPAMPをリセットします。

ビット13 **LPTIM4RST**：LPTIM4のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：LPTIM4をリセットします。

ビット12 **LPTIM3RST**：LPTIM3のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：LPTIM3をリセットします。

ビット11 **LPTIM1RST**：LPTIM1のリセット
このビットは、ソフトウェアによってセット／クリアされます。
0：影響なし。
1：LPTIM1をリセットします。

ビット10:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C3RST**：I2C3のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：I2C3をリセットします。

ビット6 **LPUART1RST**：LPUART1のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：LPUART1をリセットします。

ビット5 **SPI3RST**：SPI3のリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SPI3をリセットします。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SYSCFGRST**：SYSCFGのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：影響なし。

1：SYSCFGをリセットします。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.29 RCC AHB1ペリフェラルクロック有効化レジスタ（RCC_AHB1ENR）

アドレスオフセット：0x088

リセット値：0xD000 0100（STM32U535/545/575/585の場合）

リセット値：0xD020 0100（STM32U59x/5Ax/5Fx/5Gxの場合）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM1EN	DCACHE1EN	Res.	BKPSRAMEN	Res.	Res.	Res.	GTZC1EN	Res.	Res.	DCACHE2EN	GPU2DEN	GFXMMUEN	DMA2DEN	RAMCFGEN	TSCEN
rw	rw		rw				rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JPEGEN	Res.	Res.	CRCEN	Res.	Res.	Res.	FlashEN	Res.	Res.	Res.	Res.	MDF1EN	FMACEN	CORDICEN	GPDMA1EN
rw			rw				rw					rw	rw	rw	rw

ビット31 **SRAM1EN**：SRAM1クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM1クロックは無効です。

1：SRAM1クロックは有効です。

ビット30 **DCACHE1EN**：DCACHE1クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：DCACHE1クロックは無効です。

1：DCACHE1クロックは有効です。

注： OCTOSPI1、OCTOSPI2、HSPI1、またはFSMCを通じて外部メモリにアクセスする場合、DCACHE1がバイパスされる場合でも、DCACHE1クロックを有効にする必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **BKPSRAMEN**：BKPSRAMクロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：BKPSRAMクロックは無効です。

1：BKPSRAMクロックは有効です。

ビット27:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **GTZC1EN**：GTZC1クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：GTZC1クロックは無効です。

1：GTZC1クロックは有効です。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **DCACHE2EN**：DCACHE2クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：DCACHE2クロックは無効です。

1：DCACHE2クロックは有効です。

注： DCACHE2がバイパスされる場合でも、DCACHE2クロックのメモリアクセスを有効にする必要があります。

このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット20 **GPU2DEN**：GPU2Dクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：GPU2Dクロックは無効です。

1：GPU2Dクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **GFXMMUEN**：GFXMMUクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：GFXMMUクロックは無効です。

1：GFXMMUクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **DMA2DEN**：DMA2Dクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA2Dクロックは無効です。

1：DMA2Dクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット17 **RAMCFGEN**：RAMCFGクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：RAMCFGクロックは無効です。
1：RAMCFGクロックは有効です。

ビット16 **TSCEN**：タッチセンシングコントローラクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TSCクロックは無効です。
1：TSCクロックは有効です。

ビット15 **JPEGEN**：JPEGクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：JPEGクロックは無効です。
1：JPEGクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CRCEN**：CRCクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：CRCクロックは無効です。
1：CRCクロックは有効です。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **FLASHEN**：FLASHクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。このビットは、Flashメモリがパワーダウンモードの場合にのみ無効になります。
0：FLASHクロックは無効です。
1：FLASHクロックは有効です。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **MDF1EN**：MDF1クロックの有効化
このビットは、ソフトウェアによってセット／リセットされます。
0：MDF1クロックは無効です。
1：MDF1クロックは有効です。

ビット2 **FMACEN**：FMACクロックの有効化
このビットは、ソフトウェアによってセット／リセットされます。
0：FMACクロックは無効です。
1：FMACクロックは有効です。

ビット1 **CORDICEN**：CORDICクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：CORDICクロックは無効です。
1：CORDICクロックは有効です。

ビット0 **GPDMA1EN**：GPDMA1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：GPDMA1クロックは無効です。
1：GPDMA1クロックは有効です。

11.8.30 RCCAHB2ペリフェラルクロック有効化レジスタ1（RCC_AHB2ENR1）

アドレスオフセット：0x08C

リセット値：0x4000 0000（STM32U535/545の場合）

リセット値：0xC000 0000（STM32U575/585/59x/5Ax/5Fx/5Gxの場合）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注：　ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM3EN	SRAM2EN	Res.	SDMMC2EN	SDMMC1EN	Res.	Res.	OTFDEC2EN	OTFDEC1EN	Res.	OCTOSPIMEN	SAESEN	PKAEN	RNGEN	HASHEN	AESEN
rw	rw		rw	rw			rw	rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OTGHSPHYEN	OTGEN	Res.	DCMI_PSSIEN	Res.	ADC12EN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
rw	rw		rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **SRAM3EN**：SRAM3クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM3クロックは無効です。

1：SRAM3クロックは有効です。

注：　このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット30 **SRAM2EN**：SRAM2クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM2クロックは無効です。

1：SRAM2クロックは有効です。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **SDMMC2EN**：SDMMC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SDMMC2クロックは無効です。

1：SDMMC2クロックは有効です。

注：　このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット27 **SDMMC1EN**：SDMMC1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SDMMC1クロックは無効です。

1：SDMMC1クロックは有効です。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **OTFDEC2EN**：OTFDEC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OTFDEC2クロックは無効です。

1：OTFDEC2クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 **OTFDEC1EN**：OTFDEC1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OTFDEC1クロックは無効です。

1：OTFDEC1クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OCTOSPIMEN**：OCTOSPIMクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OCTOSPIMクロックは無効です。

1：OCTOSPIMクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット20 **SAESEN**：SAESクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SAESクロックは無効です。

1：SAESクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **PKAEN**：PKAクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：PKAクロックは無効です。

1：PKAクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **RNGEN**：RNGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：RNGクロックは無効です。

1：RNGクロックは有効です。

ビット17 **HASHEN**：HASHクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：HASHクロックは無効です。

1：HASHクロックは有効です。

ビット16 **AESEN**：AESクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：AESクロックは無効です。

1：AESクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット15 **OTGHSPHYEN**：OTG_HSPHYclockの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OTG_HS PHYクロックは無効です。

1：OTG_HS PHYクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14 **OTGEN**：OTG_FSまたはOTG_HSクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OTG_FSまたはOTG_HSクロックは無効です。

1：OTG_FSまたはOTG_HSクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DCMI_PSSIEN**：DCMIおよびPSSIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：DCMIおよびPSSIクロックは無効です。

1：DCMIおよびPSSIクロックは有効です。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADC12EN**：ADC1およびADC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：ADC1およびADC2クロックは無効です。

1：ADC1およびADC2クロックは有効です。

注： このビットは、STM32U535/545/575/585のADC1と、STM32U59x/5Ax/5Fx/5GxのADC1/ADC2に影響します。

ビット9 **GPIOJEN**：I/OポートJクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I/OポートJクロックは無効です。

1：I/OポートJクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット8 **GPIOIEN**：I/OポートIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I/OポートIクロックは無効です。

1：I/OポートIクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7 **GPIOHEN**：I/OポートHクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートHクロックは無効です。
1：I/OポートHクロックは有効です。

ビット6 **GPIOGEN**：I/OポートGクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートGクロックは無効です。
1：I/OポートGクロックは有効です。

ビット5 **GPIOFEN**：I/OポートFクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートFクロックは無効です。
1：I/OポートFクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット4 **GPIOEEN**：I/OポートEクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートEクロックは無効です。
1：I/OポートEクロックは有効です。

ビット3 **GPIODEN**：I/OポートDクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートDクロックは無効です。
1：I/OポートDクロックは有効です。

ビット2 **GPIOCEN**：I/OポートCクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートCクロックは無効です。
1：I/OポートCクロックは有効です。

ビット1 **GPIOBEN**：I/OポートBクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートBクロックは無効です。
1：I/OポートBクロックは有効です。

ビット0 **GPIOAEN**：I/OポートAクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I/OポートAクロックは無効です。
1：I/OポートAクロックは有効です。

11.8.31 RCCAHB2ペリフェラルクロック有効化レジスタ2（RCC_AHB2ENR2）

アドレスオフセット：0x090

リセット値：0x0000 0000（STM32U535/545/575/585の場合）

リセット値：0x8000 0000（STM32U59x/5Axの場合）

リセット値：0xC000 0000（STM32U5Fx/5Gxの場合）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注：　ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM5EN	SRAM6EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	HSPI1EN	Res.	Res.	Res.	OCTOSPI2EN	Res.	Res.	Res.	OCTOSPI1EN	Res.	Res.	Res.	FSMCEN
			rw				rw				rw				rw

ビット31 **SRAM5EN**：SRAM5クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM5クロックは無効です。

1：SRAM5クロックは有効です。

注：　このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット30 **SRAM6EN**：SRAM6クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM6クロックは無効です。

1：SRAM6クロックは有効です。

注：　このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット29:13　予約済みであり、リセット値に保持する必要があります。

ビット12 **HSPI1EN**：HSPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：HSPI1クロックは無効です。

1：HSPI1クロックは有効です。

注：　このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット11:9　予約済みであり、リセット値に保持する必要があります。

ビット8 **OCTOSPI2EN**：OCTOSPI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OCTOSPI2クロックは無効です。

1：OCTOSPI2クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OCTOSPI1EN**：OCTOSPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OCTOSPI1クロックは無効です。

1：OCTOSPI1クロックは有効です。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **FSMCEN**：FSMCクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：FSMCクロックは無効です。

1：FSMCクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

11.8.32 RCCAHB3ペリフェラルクロック有効化レジスタ（RCC_AHB3ENR）

アドレスオフセット：0x094

リセット値：0x8000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM4EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	GTZC2EN	Res.	ADF1EN	LPDMA1EN	Res.	Res.	DAC1EN	ADC4EN	Res.	Res.	PWREN	Res.	LPGPIO1EN
			rw		rw	rw			rw	rw			rw		rw

ビット31 **SRAM4EN**：SRAM4クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。

0：SRAM4クロックは無効です。

1：SRAM4クロックは有効です。

ビット30:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **GTZC2EN**：GTZC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：GTZC2クロックは無効です。

1：GTZC2クロックは有効です。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADF1EN**：ADF1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：ADF1クロックは無効です。

1：ADF1クロックは有効です。

ビット9 **LPDMA1EN**：LPDMA1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：LPDMA1クロックは無効です。

1：LPDMA1クロックは有効です。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **DAC1EN**：DAC1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC1クロックは無効です。

1：DAC1クロックは有効です。

ビット5 **ADC4EN**：ADC4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：ADC4クロックは無効です。

1：ADC4クロックは有効です。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **PWREN**：PWRクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：PWRクロックは無効です。

1：PWRクロックは有効です。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **LPGPIO1EN**：LPGPIO1イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：LPGPIO1クロックは無効です。

1：LPGPIO1クロックは有効です。

11.8.33 RCC APB1ペリフェラルクロック有効化レジスタ1（RCC_APB1ENR1）

アドレスオフセット：0x09C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	USART6EN	CRSEN	Res.	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	Res.
						rw	rw		rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI2EN	Res.	Res.	WWDGEN	Res.	Res.	Res.	Res.	Res.	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	rw			rs						rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **USART6EN**：USART6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：USART6クロックは無効です。

1：USART6クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **CRSEN**：CRSクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CRSクロックは無効です。

1：CRSクロックは有効です。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **I2C2EN**：I2C2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I2C2クロックは無効です。

1：I2C2クロックは有効です。

ビット21 **I2C1EN**：I2C1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I2C1クロックは無効です。

1：I2C1クロックは有効です。

ビット20 **UART5EN**：UART5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：UART5クロックは無効です。

1：UART5クロックは有効です。

ビット19 **UART4EN**：UART4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：UART4クロックは無効です。

1：UART4クロックは有効です。

ビット18 **USART3EN**：USART3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：USART3クロックは無効です。

1：USART3クロックは有効です。

ビット17 **USART2EN**：USART2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：USART2クロックは無効です。

1：USART2クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット16:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **SPI2EN**：SPI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SPI2クロックは無効です。

1：SPI2クロックは有効です。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **WWDGEN**：WWDGクロックの有効化

このビットは、ウィンドウ型ウォッチドッグクロックを有効にするために、ソフトウェアによってセットされます。ハードウェアシステムリセットによってリセットされます。このビットは、WWDG_SWオプションビットがリセットされたときに、ハードウェアでもセットできます。

0：WWDGクロックは無効です。

1：WWDGクロックは有効です。

ビット10:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **TIM7EN**：TIM7クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM7クロックは無効です。

1：TIM7クロックは有効です。

ビット4 **TIM6EN**：TIM6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM6クロックは無効です。

1：TIM6クロックは有効です。

ビット3 **TIM5EN**：TIM5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM5クロックは無効です。

1：TIM5クロックは有効です。

ビット2 **TIM4EN**：TIM4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM4クロックは無効です。

1：TIM4クロックは有効です。

ビット1 **TIM3EN**：TIM3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM3クロックは無効です。

1：TIM3クロックは有効です。

ビット0 **TIM2EN**：TIM2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM2クロックは無効です。

1：TIM2クロックは有効です。

11.8.34 RCCAPB1ペリフェラルクロック有効化レジスタ2（RCC_APB1ENR2）

アドレスオフセット：0x0A0

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1EN	Res.	I2C6EN	I2C5EN	LPTIM2EN	Res.	Res.	Res.	I2C4EN	Res.
						rw		rw	rw	rw				rw	

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **UCPD1EN**：UCPD1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：UCPD1クロックは無効です。

1：UCPD1クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **FDCAN1EN**：FDCAN1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：FDCAN1クロックは無効です。

1：FDCAN1クロックは有効です。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C6EN**：I2C6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I2C6クロックは無効です。

1：I2C6クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット6 **I2C5EN**：I2C5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：I2C5クロックは無効です。

1：I2C5クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット5 **LPTIM2EN**：LPTIM2クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：LPTIM2クロックは無効です。
1：LPTIM2クロックは有効です。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **I2C4EN**：I2C4クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I2C4クロックは無効です。
1：I2C4クロックは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.35 RCCAPB2ペリフェラルクロック有効化レジスタ（RCC_APB2ENR）

アドレスオフセット：0x0A4

リセット値：0x0000 0000

アクセス：ワード、ハーフワード、およびバイトアクセス

注： **ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタへの読出し／書込みアクセスはサポートされません。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DSIEN	LTDCEN	GFXTIMEN	USBEN	Res.	SAI2EN	SAI1EN	Res.	Res.	TIM17EN	TIM16EN	TIM15EN
				rw	rw	rw	rw		rw	rw			rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1EN	TIM8EN	SPI1EN	TIM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw											

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DSIEN**：DSIクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：DSIクロックは無効です。
1：DSIクロックは有効です。
注： **このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。**

ビット26 **LTDCEN**：LTDCクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：LTDCクロックは無効です。
1：LTDCクロックは有効です。
注： **このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。**

ビット25 **GFXTIMEN**：GFXTIMクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：GFXTIMクロックは無効です。

1：GFXTIMクロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **USBEN**：USBクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：USBクロックは無効です。

1：USBクロックは有効です。

注： このビットはSTM32U535/545デバイスでのみ使用できます。これは、STM32U5シリーズの他のデバイスで予約済みです。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **SAI2EN**：SAI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI2クロックは無効です。

1：SAI2クロックは有効です。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。

ビット21 **SAI1EN**：SAI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI1クロックは無効です。

1：SAI1クロックは有効です。

ビット20:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **TIM17EN**：TIM17クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM17クロックは無効です。

1：TIM17クロックは有効です。

ビット17 **TIM16EN**：TIM16クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM16クロックは無効です。

1：TIM16クロックは有効です。

ビット16 **TIM15EN**：TIM15クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM15クロックは無効です。

1：TIM15クロックは有効です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **USART1EN**：USART1clockイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：USART1クロックは無効です。

1：USART1クロックは有効です。

ビット13 T**IM8EN**：TIM8クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM8クロックは無効です。

1：TIM8クロックは有効です。

ビット12 **SPI1EN**：SPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SPI1クロックは無効です。

1：SPI1クロックは有効です。

ビット11 **TIM1EN**：TIM1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TIM1クロックは無効です。

1：TIM1クロックは有効です。

ビット10:0 予約済みであり、リセット値に保持する必要があります。

11.8.36 RCC APB3ペリフェラルクロック有効化レジスタ（RCC_APB3ENR）

アドレスオフセット：0x0A8

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBEN	VREFEN	Res.	Res.	Res.	Res.
										rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMPEN	OPAMPEN	LPTIM4EN	LPTIM3EN	LPTIM1EN	Res.	Res.	Res.	I2C3EN	LPUART1EN	SPI3EN	Res.	Res.	Res.	SYSCFGEN	Res.
rw	rw	rw	rw	rw				rw	rw	rw				rw	

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RTCAPBEN**：RTCおよびTAMPAPBクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：RTCおよびTAMP APBクロックは無効です。

1：RTCおよびTAMP APBクロックは有効です。

ビット20 **VREFEN**：VREFBUFクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：VREFBUFクロックは無効です。

1：VREFBUFクロックは有効です。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **COMPEN**：COMPクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：COMPクロックは無効です。

1：COMPクロックは有効です。

ビット14 **OPAMPEN**：OPAMPクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OPAMPクロックは無効です。

1：OPAMPクロックは有効です。

ビット13 **LPTIM4EN**：LPTIM4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：LPTIM4クロックは無効です。

1：LPTIM4クロックは有効です。

ビット12 **LPTIM3EN**：LPTIM3クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：LPTIM3クロックは無効です。
1：LPTIM3クロックは有効です。

ビット11 **LPTIM1EN**：LPTIM1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：LPTIM1クロックは無効です。
1：LPTIM1クロックは有効です。

ビット10:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C3EN**：I2C3クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：I2C3クロックは無効です。
1：I2C3クロックは有効です。

ビット6 **LPUART1EN**：LPUART1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：LPUART1クロックは無効です。
1：LPUART1クロックは有効です。

ビット5 **SPI3EN**：SPI3クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SPI3クロックは無効です。
1：SPI3クロックは有効です。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SYSCFGEN**：SYSCFGクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SYSCFGクロックは無効です。
1：SYSCFGクロックは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.37 SLEEPモードおよびSTOPモードにおけるRCC AHB1ペリフェラルクロックの有効化レジスタ（RCC_AHB1SMENR）

アドレスオフセット：0x0B0

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM1SMEN	DCACHE1SMEN	ICACHESMEN	BKPSRAMSMEN	Res.	Res.	Res.	GTZC1SMEN	Res.	Res.	DCACHE2SMEN	GPU2DSMEN	GFXMMUSMEN	DMA2DSMEN	RAMCFGSMEN	TSCSMEN
rw	rw	rw	rw				rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JPEGSMEN	Res.	Res.	CRCSMEN	Res.	Res.	Res.	FlashSMEN	Res.	Res.	Res.	Res.	MDF1SMEN	FMACSMEN	CORDICSMEN	GPDMA1SMEN
rw			rw				rw					rw	rw	rw	rw

ビット31 **SRAM1SMEN**：SLEEPモードおよびSTOPモード時SRAM1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、SRAM1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、SRAM1クロックはクロックゲーティングにより有効になります。

ビット30 **DCACHE1SMEN**：SLEEPモードおよびSTOPモード時DCACHE1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、DCACHE1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、DCACHE1クロックはクロックゲーティングにより有効になります。

ビット29 **ICACHESMEN**：SLEEPモードおよびSTOPモード時ICACHEクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、ICACHEクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、ICACHEクロックはクロックゲーティングにより有効になります。

ビット28 **BKPSRAMSMEN**：SLEEPモードおよびSTOPモード時BKPSRAMクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、BKPSRAMクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、BKPSRAMクロックはクロックゲーティングにより有効になります。

ビット27:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **GTZC1SMEN**：SLEEPモードおよびSTOPモード時GTZC1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、GTZC1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、GTZC1クロックはクロックゲーティングにより有効になります。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **DCACHE2SMEN**：SLEEPモードおよびSTOPモード時DCACHE2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、DCACHE2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、DCACHE2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット20 **GPU2DSMEN**：SLEEPモードおよびSTOPモード時GPU2Dクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、GPU2Dクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、GPU2Dクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **GFXMMUSMEN**：SLEEPモードおよびSTOPモード時GFXMMUクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、GFXMMUクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、GFXMMUクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **DMA2DSMEN**：SLEEPモードおよびSTOPモード時DMA2Dクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、DMA2Dクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、DMA2Dクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット17 **RAMCFGSMEN**：SLEEPモードおよびSTOPモード時RAMCFGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、RAMCFGクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、RAMCFGクロックはクロックゲーティングにより有効になります。

ビット16 **TSCSMEN**：SLEEPモードおよびSTOPモード時TSCクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TSCクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TSCクロックはクロックゲーティングにより有効になります。

ビット15 **JPEGSMEN**：SLEEPモードおよびSTOPモード時JPEGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、JPEGクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、JPEGクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CRCSMEN**：SLEEPモードおよびSTOPモード時CRCクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、CRCクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、CRCクロックはクロックゲーティングにより有効になります。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **FLASHSMEN**：SLEEPモードおよびSTOPモード時FLASHクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、FLASHクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、FLASHクロックはクロックゲーティングにより有効になります。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **MDF1SMEN**：SLEEPモードおよびSTOPモード時MDF1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、MDF1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、MDF1クロックはクロックゲーティングにより有効になります。
注： ペリフェラルをSTOPモードからウェイクアップさせるには、このビットをセットする必要があります。

ビット2 **FMACSMEN**：SLEEPモードおよびSTOPモード時FMACクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、FMACクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、FMACクロックはクロックゲーティングにより有効になります。

ビット1 **CORDICSMEN**：SLEEPモードおよびSTOPモード時CORDICクロックの有効化
このビットは、SLEEPモード中にソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、CORDICクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、CORDICクロックはクロックゲーティングにより有効になります。

ビット0 **GPDMA1SMEN**：SLEEPモードおよびSTOPモード時GPDMA1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、GPDMA1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、GPDMA1クロックはクロックゲーティングにより有効になります。
注： ペリフェラルをSTOPモードからウェイクアップさせるには、このビットをセットする必要があります。

11.8.38 SLEEPモードおよびSTOPモードにおけるRCC AHB2ペリフェラルクロック有効レジスタ1（RCC_AHB2SMENR1）

アドレスオフセット：0x0B4

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM3SMEN	SRAM2SMEN	Res.	SDMMC2SMEN	SDMMC1SMEN	Res.	Res.	OTFDEC2SMEN	OTFDEC1SMEN	Res.	OCTOSPIMSMEN	SAESSMEN	PKASMEN	RNGSMEN	HASHSMEN	AESSMEN
rw	rw		rw	rw			rw	rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OTGHSPHYSMEN	OTGSMEN	Res.	DCMI_PSSISMEN	Res.	ADC12SMEN	GPIOJSMEN	GPIOISMEN	GPIOHSMEN	GPIOGSMEN	GPIOFSMEN	GPIOESMEN	GPIODSMEN	GPIOCSMEN	GPIOBSMEN	GPIOASMEN
rw	rw		rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **SRAM3SMEN**：SLEEPモードおよびSTOPモード時SRAM3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SRAM3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SRAM3クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット30 **SRAM2SMEN**：SLEEPモードおよびSTOPモード時SRAM2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SRAM2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SRAM2クロックはクロックゲーティングにより有効になります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **SDMMC2SMEN**：SLEEPモードおよびSTOPモード時SDMMC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SDMMC2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SDMMC2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット27 **SDMMC1SMEN**：SLEEPモードおよびSTOPモード時SDMMC1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SDMMC1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SDMMC1クロックはクロックゲーティングにより有効になります。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **OTFDEC2SMEN**：SLEEPモードおよびSTOPモード時OTFDEC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OTFDEC2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OTFDEC2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 **OTFDEC1SMEN**：SLEEPモードおよびSTOPモード時OTFDEC1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OTFDEC1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OTFDEC1クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OCTOSPIMSMEN**：SLEEPモードおよびSTOPモード時OCTOSPIMクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OCTOSPIMクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OCTOSPIMクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット20 **SAESSMEN**：SLEEPモードおよびSTOPモード時SAESクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SAESクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SAESクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット19 **PKASMEN**：SLEEPモードおよびSTOPモード時PKAクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、PKAクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、PKAクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット18 **RNGSMEN**：SLEEPモードおよびSTOPモード時RNGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、RNGクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、RNGクロックはクロックゲーティングにより有効になります。

ビット17 **HASHSMEN**：SLEEPモードおよびSTOPモード時HASHクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0:SLEEPモードおよびSTOPモード時､HASHクロックはクロックゲーティングにより無効になります。

1:SLEEPモードおよびSTOPモード時､HASHクロックはクロックゲーティングにより有効になります。

ビット16 **AESSMEN**：SLEEPモードおよびSTOPモード時AESクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、AESクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、AESクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット15 **OTGHSPHYSMEN**：SLEEPモードおよびSTOPモード時OTG_HSPHYクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OTG_HS PHYクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OTG_HS PHYクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14 **OTGSMEN**：SLEEPモードおよびSTOPモード時OTG_FSおよびOTG_HSクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OTG_FSおよびOTG_HSクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OTG_FSおよびOTG_HSクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DCMI_PSSISMEN**：SLEEPモードおよびSTOPモード時DCMIおよびPSSIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、DCMIおよびPSSIクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、DCMIおよびPSSIクロックはクロックゲーティングにより有効になります。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADC12SMEN**：SLEEPモードおよびSTOPモード時ADC1およびADC2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、ADC1およびADC2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、ADC1およびADC2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U535/545/575/585のADC1と、STM32U59x/5Ax/5Fx/5GxのADC1/ADC2に影響します。

ビット9 **GPIOJSMEN**：SLEEPモードおよびSTOPモード時I/OポートJクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I/OポートJクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I/OポートJクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット8 **GPIOISMEN**：SLEEPモードおよびSTOPモード時I/OポートIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I/OポートIクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I/OポートIクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7 **GPIOHSMEN**：SLEEPモードおよびSTOPモード時I/OポートHクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、IOポートHクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、IOポートHクロックはクロックゲーティングにより有効になります。

ビット6 **GPIOGSMEN**：SLEEPモードおよびSTOPモード時I/OポートGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、IOポートGクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、IOポートGクロックはクロックゲーティングにより有効になります。

ビット5 **GPIOFSMEN**：SLEEPモードおよびSTOPモード時I/OポートFクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、IOポートFクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、IOポートFクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット4 **GPIOESMEN**：SLEEPモードおよびSTOPモード時I/OポートEクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、IOポートEクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、IOポートEクロックはクロックゲーティングにより有効になります。

ビット3 **GPIODSMEN**：SLEEPモードおよびSTOPモード時I/OポートDクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、IOポートDクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、IOポートDクロックはクロックゲーティングにより有効になります。

ビット2 **GPIOCSMEN**：SLEEPモードおよびSTOPモード時I/OポートCクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、IOポートCクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、IOポートCクロックはクロックゲーティングにより有効になります。

ビット1 **GPIOBSMEN**：SLEEPモードおよびSTOPモード時I/OポートBクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、IOポートBクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、IOポートBクロックはクロックゲーティングにより有効になります。

ビット0 **GPIOASMEN**：SLEEPモードおよびSTOPモード時I/OポートAクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、IOポートAクロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、IOポートAクロックはクロックゲーティングにより有効になります。

11.8.39 SLEEPモードおよびSTOPモードにおけるRCC AHB2ペリフェラルクロック有効レジスタ2（RCC_AHB2SMENR2）

アドレスオフセット：0x0B8

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード;およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。

STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM5SMEN	SRAM6SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	HSPI1SMEN	Res.	Res.	Res.	OCTOSPI2SMEN	Res.	Res.	Res.	OCTOSPI1SMEN	Res.	Res.	Res.	FSMCSMEN
			rw				rw				rw				rw

ビット31 **SRAM5SMEN**：SLEEPモードおよびSTOPモード時SRAM5クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、SRAM5クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、SRAM5クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット30 **SRAM6SMEN**：SLEEPモードおよびSTOPモード時SRAM6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SRAM6クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SRAM6クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット29:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **HSPI1SMEN**：SLEEPモードおよびSTOPモード時HSPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、HSPI1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、HSPI1クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **OCTOSPI2SMEN**：SLEEPモードおよびSTOPモード時OCTOSPI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OCTOSPI2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OCTOSPI2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OCTOSPI1SMEN**：SLEEPモードおよびSTOPモード時OCTOSPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OCTOSPI1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OCTOSPI1クロックはクロックゲーティングにより有効になります。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **FSMCSMEN**：SLEEPモードおよびSTOPモード時FSMCクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、FSMCクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、FSMCクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

11.8.40 SLEEPモードおよびSTOPモードにおけるRCC AHB3ペリフェラルクロックの有効化レジスタ（RCC_AHB3SMENR）

アドレスオフセット：0x0BC

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM4SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	GTZC2SMEN	Res.	ADF1SMEN	LPDMA1SMEN	Res.	Res.	DAC1SMEN	ADC4SMEN	Res.	Res.	PWRSMEN	Res.	LPGPIO1SMEN
			rw		rw	rw			rw	rw			rw		rw

ビット31 **SRAM4SMEN**：SLEEPモードおよびSTOPモード時SRAM4クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、SRAM4クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、SRAM4クロックはクロックゲーティングにより有効になります。

ビット30:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **GTZC2SMEN**：SLEEPモードおよびSTOPモード時GTZC2クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、GTZC2クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、GTZC2クロックはクロックゲーティングにより有効になります。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ADF1SMEN**：SLEEPモードおよびSTOPモード時ADF1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、ADF1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、ADF1クロックはクロックゲーティングにより有効になります。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット9 **LPDMA1SMEN**：SLEEPモードおよびSTOPモード時LPDMA1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、LPDMA1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、LPDMA1クロックはクロックゲーティングにより有効になります。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **DAC1SMEN**：SLEEPモードおよびSTOPモード時DAC1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0:SLEEPモードおよびSTOPモード時、DAC1クロックはクロックゲーティングにより無効になります。
1:SLEEPモードおよびSTOPモード時、DAC1クロックはクロックゲーティングにより有効になります。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット5 **ADC4SMEN**：SLEEPモードおよびSTOPモード時ADC4クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0:SLEEPモードおよびSTOPモード時、ADC4クロックはクロックゲーティングにより無効になります。
1:SLEEPモードおよびSTOPモード時、ADC4クロックはクロックゲーティングにより有効になります。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **PWRSMEN**：SLEEPモードおよびSTOPモード時PWRクロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0:SLEEPモードおよびSTOPモード時、PWRクロックはクロックゲーティングにより無効になります。
1:SLEEPモードおよびSTOPモード時、PWRクロックはクロックゲーティングにより有効になります。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **LPGPIO1SMEN**：SLEEPモードおよびSTOPモード時LPGPIO1の有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、LPGPIO1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、LPGPIO1クロックはクロックゲーティングにより有効になります。

11.8.41 SLEEPモードおよびSTOPモードにおけるRCCAPB1ペリフェラルクロック有効レジスタ1（RCC_APB1SMENR1）

アドレスオフセット：0x0C4

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	USART6SMEN	CRSSMEN	Res.	I2C2SMEN	I2C1SMEN	UART5SMEN	UART4SMEN	USART3SMEN	USART2SMEN	Res.
						rw	rw		rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI2SMEN	Res.	Res.	WWDGSMEN	Res.	Res.	Res.	Res.	Res.	TIM7SMEN	TIM6SMEN	TIM5SMEN	TIM4SMEN	TIM3SMEN	TIM2SMEN
	rw			rw						rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **USART6SMEN**：SLEEPモードおよびSTOPモード時USART6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、USART6クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、USART6クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **CRSSMEN**：SLEEPモードおよびSTOPモード時CRSクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、CRSクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、CRSクロックはクロックゲーティングにより有効になります。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **I2C2SMEN**：SLEEPモードおよびSTOPモード時I2C2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C2クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット21 **I2C1SMEN**：SLEEPモードおよびSTOPモード時I2C1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C1クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット20 **UART5SMEN**：SLEEPモードおよびSTOPモード時UART5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、UART5クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、UART5クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット19 **UART4SMEN**：SLEEPモードおよびSTOPモード時UART4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、UART4クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、UART4クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット18 **USART3SMEN**：SLEEPモードおよびSTOPモード時USART3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、USART3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、USART3クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット17 **USART2SMEN**：SLEEPモードおよびSTOPモード時USART2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、USART2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、USART2クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット16:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **SPI2SMEN**：SLEEPモードおよびSTOPモード時SPI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SPI2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SPI2クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **WWDGSMEN**：SLEEPモードおよびSTOPモード時ウィンドウ型ウォッチドッグクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。このビットは、ハードウェアWWDGオプションが有効である場合、ハードウェアで強制的に「1」になります。

0：SLEEPモードおよびSTOPモード時、ウィンドウ型ウォッチドッグクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、ウィンドウ型ウォッチドッグクロックはクロックゲーティングにより有効になります。

ビット10:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **TIM7SMEN**：SLEEPモードおよびSTOPモード時TIM7クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM7クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM7クロックはクロックゲーティングにより有効になります。

ビット4 **TIM6SMEN**：SLEEPモードおよびSTOPモード時TIM6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM6クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM6クロックはクロックゲーティングにより有効になります。

ビット3 **TIM5SMEN**：SLEEPモードおよびSTOPモード時TIM5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM5クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM5クロックはクロックゲーティングにより有効になります。

ビット2 **TIM4SMEN**：SLEEPモードおよびSTOPモード時TIM4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM4クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM4クロックはクロックゲーティングにより有効になります。

ビット1 **TIM3SMEN**：SLEEPモードおよびSTOPモード時TIM3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM3クロックはクロックゲーティングにより有効になります。

ビット0 **TIM2SMEN**：SLEEPモードおよびSTOPモード時TIM2クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、TIM2クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、TIM2クロックはクロックゲーティングにより有効になります。

11.8.42 SLEEPモードおよびSTOPモードにおけるRCCAPB1ペリフェラルクロック有効レジスタ2（RCC_APB1SMENR2）

アドレスオフセット：0x0C8
リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1SMEN	Res.	I2C6SMEN	I2C5SMEN	LPTIM2SMEN	Res.	Res.	Res.	I2C4SMEN	Res.
						rw		rw	rw	rw				rw	

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **UCPD1SMEN**：SLEEPモードおよびSTOPモード時UCPD1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、UCPD1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、UCPD1クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット22:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **FDCAN1SMEN**：SLEEPモードおよびSTOPモード時FDCAN1クロックの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SLEEPモードおよびSTOPモード時、FDCAN1クロックはクロックゲーティングにより無効になります。
1：SLEEPモードおよびSTOPモード時、FDCAN1クロックはクロックゲーティングにより有効になります。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C6SMEN**：SLEEPモードおよびSTOPモード時I2C6クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C6クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C6クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット6 **I2C5SMEN**：SLEEPモードおよびSTOPモード時I2C5クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C5クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C5クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット5 **LPTIM2SMEN**：SLEEPモードおよびSTOPモード時LPTIM2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LPTIM2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LPTIM2クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **I2C4SMEN**：SLEEPモードおよびSTOPモード時I2C4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C4クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C4クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.43 SLEEPモードおよびSTOPモードにおけるRCC APB2ペリフェラルクロック有効レジスタ（RCC_APB2SMENR）

アドレスオフセット：0x0CC

リセット値：0xFFFF FFFF

アクセス：ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DSISMEN	LTDCSMEN	GFXTIMSMEN	USBSMEN	Res.	SAI2SMEN	SAI1SMEN	Res.	Res.	TIM17SMEN	TIM16SMEN	TIM15SMEN
				rw	rw	rw	rw		rw	rw			rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1SMEN	TIM8SMEN	SPI1SMEN	TIM1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw											

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DSISMEN**：SLEEPモードおよびSTOPモード時DSIクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、DSIクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、DSIクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット26 **LTDCSMEN**：SLEEPモードおよびSTOPモード時LTDCクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LTDCクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LTDCクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット25 **GFXTIMSMEN**：SLEEPモードおよびSTOPモード時GFXTIMクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、GFXTIMクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、GFXTIMクロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット24 **USBSMEN**：SLEEPモードおよびSTOPモード時USBクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、USBクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、USBクロックはクロックゲーティングにより有効になります。

注： このビットはSTM32U535/545デバイスでのみ使用できます。これは、STM32U5シリーズの他のデバイスで予約済みです。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22 **SAI2SMEN**：SLEEPモードおよびSTOPモード時SAI2クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SAI2クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SAI2クロックはクロックゲーティングにより有効になります。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット21 **SAI1SMEN**：SLEEPモードおよびSTOPモード時SAI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SAI1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SAI1クロックはクロックゲーティングにより有効になります。

ビット20:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **TIM17SMEN**：SLEEPモードおよびSTOPモード時TIM17クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM17クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM17クロックはクロックゲーティングにより有効になります。

ビット17 **TIM16SMEN**：SLEEPモードおよびSTOPモード時TIM16クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM16クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM16クロックはクロックゲーティングにより有効になります。

ビット16 **TIM15SMEN**：SLEEPモードおよびSTOPモード時TIM15クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM15クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM15クロックはクロックゲーティングにより有効になります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **USART1SMEN**：SLEEPモードおよびSTOPモード時USART1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、USART1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、USART1クロックはクロックゲーティングにより有効になります。

注： ペリフェラルをSTOPモードからウェイクアップさせるには、このビットをセットする必要があります。

ビット13 T**IM8SMEN**：SLEEPモードおよびSTOPモード時TIM8クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM8クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM8クロックはクロックゲーティングにより有効になります。

ビット12 **SPI1SMEN**：SLEEPモードおよびSTOPモード時SPI1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SPI1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SPI1クロックはクロックゲーティングにより有効になります。

注： ペリフェラルをSTOPモードからウェイクアップさせるには、このビットをセットする必要があります。

ビット11 **TIM1SMEN**：SLEEPモードおよびSTOPモード時TIM1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、TIM1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、TIM1クロックはクロックゲーティングにより有効になります。

ビット10:0 予約済みであり、リセット値に保持する必要があります。

11.8.44 SLEEPモードおよびSTOPモードにおけるRCC APB3ペリフェラルクロックの有効化レジスタ（RCC_APB3SMENR）

アドレスオフセット：0x0D0

リセット値：0xFFFF FFFF

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。STOPモードでビットがセットされた場合、ペリフェラル（このペリフェラルまたは他のペリフェラル）によりAHBまたはAPBクロックがリクエストされた場合のみ、対応するペリフェラルクロックが有効になります（セクション11.4.24を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBSMEN	VREFSMEN	Res.	Res.	Res.	Res.
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMPSMEN	OPAMPSMEN	LPTIM4SMEN	LPTIM3SMEN	LPTIM1SMEN	Res.	Res.	Res.	I2C3SMEN	LPUART1SMEN	SPI3SMEN	Res.	Res.	Res.	SYSCFGSMEN	Res.
rw	rw	rw	rw	rw				rw	rw	rw				rw	

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RTCAPBSMEN**：SLEEPモードおよびSTOPモード時RTCおよびTAMPAPBクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、RTCおよびTAMP APBクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、RTCおよびTAMP APBクロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット20 **VREFSMEN**：SLEEPモードおよびSTOPモード時VREFBUFクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、VREFBUFクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、VREFBUFクロックはクロックゲーティングにより有効になります。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **COMPSMEN**：SLEEPモードおよびSTOPモード時COMPクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、COMPクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、COMPクロックはクロックゲーティングにより有効になります。

ビット14 **OPAMPSMEN**：SLEEPモードおよびSTOPモード時OPAMPクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、OPAMPクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、OPAMPクロックはクロックゲーティングにより有効になります。

ビット13 **LPTIM4SMEN**：SLEEPモードおよびSTOPモード時LPTIM4クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LPTIM4クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LPTIM4クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット12 **LPTIM3SMEN**：SLEEPモードおよびSTOPモード時LPTIM3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LPTIM3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LPTIM3クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット11 **LPTIM1SMEN**：SLEEPモードおよびSTOPモード時LPTIM1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LPTIM1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LPTIM1クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット10:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C3SMEN**：SLEEPモードおよびSTOPモード時I2C3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、I2C3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、I2C3クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット6 **LPUART1SMEN**：SLEEPモードおよびSTOPモード時LPUART1クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、LPUART1クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、LPUART1クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット5 **SPI3SMEN**：SLEEPモードおよびSTOPモード時SPI3クロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SPI3クロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SPI3クロックはクロックゲーティングにより有効になります。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SYSCFGSMEN**：SLEEPモードおよびSTOPモード時SYSCFGクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：SLEEPモードおよびSTOPモード時、SYSCFGクロックはクロックゲーティングにより無効になります。

1：SLEEPモードおよびSTOPモード時、SYSCFGクロックはクロックゲーティングにより有効になります。

ビット0 予約済みであり、リセット値に保持する必要があります。

11.8.45 RCC SmartRunドメインペリフェラル自律モードレジスタ（RCC_SRDAMR）

アドレスオフセット：0x0D8

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM4AMEN	Res.	ADF1AMEN	LPDMA1AMEN	DAC1AMEN	LPGPIO1AMEN	ADC4AMEN	Res.	Res.	Res.	RTCAPBAMEN	VREFAMEN	Res.	Res.	Res.	Res.
rw		rw	rw	rw	rw	rw				rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMPAMEN	OPAMPAMEN	LPTIM4AMEN	LPTIM3AMEN	LPTIM1AMEN	Res.	Res.	Res.	I2C3AMEN	LPUART1AMEN	SPI3AMEN	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw				rw	rw	rw					

ビット31 **SRAM4AMEN**：STOP0/1/2モードにおけるSRAM4自律モードの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：STOP0/1/2モード時、SRAM4自律モードが無効です。

1：STOP0/1/2モード時、SRAM4自律モードが有効です。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 **ADF1AMEN**：STOP0/1/2モードにおけるADF1自律モードの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：STOP0/1/2モード時、ADF1自律モードが無効です。

1：STOP0/1/2モード時、ADF1自律モードが有効です。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット28 **LPDMA1AMEN**：STOP0/1/2モードにおけるLPDMA1自律モードの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：STOP0/1/2モード時、LPDMA1自律モードが無効です。

1：STOP0/1/2モード時、LPDMA1自律モードが有効です。

注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット27 **DAC1AMEN**：STOP0/1/2モードにおけるDAC1自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、DAC1自律モードが無効です。
1：STOP0/1/2モード時、DAC1自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット26 **LPGPIO1AMEN**：STOP0/1/2モードにおけるLPGPIO1自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、LPGPIO1自律モードが無効です。
1：STOP0/1/2モード時、LPGPIO1自律モードが有効です。

ビット25 **ADC4AMEN**：STOP0/1/2モードにおけるADC4自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、ADC4自律モードが無効です。
1：STOP0/1/2モード時、ADC4自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット24:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RTCAPBAMEN**：STOP0/1/2モードにおけるRTCおよびTAMP自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、RTCおよびTAMP自律モードが無効です。
1：STOP0/1/2モード時、RTCおよびTAMP自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット20 **VREFAMEN**：STOP0/1/2モードにおけるVREFBUF自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、VREFBUF自律モードが無効です。
1：STOP0/1/2モード時、VREFBUF自律モードが有効です。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **COMPAMEN**：STOP0/1/2モードにおけるCOMP自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、COMP自律モードが無効です。
1：STOP0/1/2モード時、COMP自律モードが有効です。

ビット14 **OPAMPAMEN**：STOP0/1/2モードにおけるOPAMP自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、OPAMP自律モードが無効です。
1：STOP0/1/2モード時、OPAMP自律モードが有効です。

ビット13 **LPTIM4AMEN**：STOP0/1/2モードにおけるLPTIM4自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、LPTIM4自律モードが無効です。
1：STOP0/1/2モード時、LPTIM4自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット12 **LPTIM3AMEN**：STOP0/1/2モードにおけるLPTIM3自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、LPTIM3自律モードが無効です。
1：STOP0/1/2モード時、LPTIM3自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット11 **LPTIM1AMEN**：STOP0/1/2モードにおけるLPTIM1自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、LPTIM1自律モードが無効です。
1：STOP0/1/2モード時、LPTIM1自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット10:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **I2C3AMEN**：STOP0/1/2モードにおけるI2C3自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、I2C3自律モードが無効です。
1：STOP0/1/2モード時、I2C3自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット6 **LPUART1AMEN**：STOP0/1/2モードにおけるLPUART1自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、LPUART1自律モードが無効です。
1：STOP0/1/2モード時、LPUART1自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット5 **SPI3AMEN**：STOP0/1/2モードにおけるSPI3自律モードの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：STOP0/1/2モード時、SPI3自律モードが無効です。
1：STOP0/1/2モード時、SPI3自律モードが有効です。
注： ペリフェラルを STOP モードからウェイクアップさせるには、このビットをセットする必要があります。

ビット4:0 予約済みであり、リセット値に保持する必要があります。

11.8.46 RCCペリフェラル独立クロック設定レジスタ1（RCC_CCIPR1）

アドレスオフセット：0x0E0

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMICSEL[2:0]			Res.	ICLKSEL[1:0]		FDCAN1SEL [1:0]		SYSTICKSEL[1:0]		SPI1SEL [1:0]		LPTIM2SEL [1:0]		SPI2SEL [1:0]	
rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C4SEL [1:0]		I2C2SEL [1:0]		I2C1SEL [1:0]		UART5SEL [1:0]		UART4SEL [1:0]		USART3SEL [1:0]		USART2SEL [1:0]		USART1SEL [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 **TIMICSEL[2:0]**：TIM16、TIM17、およびLPTIM2内部入力キャプチャ用のクロックソース

TIMICSEL2をセットした場合、TIM16、TIM17、およびLPTIM2内部入力キャプチャをHSI/256、MSI/4、またはMSI/1024に接続できます。TIMICSEL[1:0]の値に応じて、MSIはMSIKまたはMSISとなります。

TIMICSEL2をクリアした場合、HSI、MSIK、およびMSISクロックソースをTIM16、TIM17、またはLPTIM2内部入力キャプチャとして選択することはできません。

0xx：HSI、MSIK、およびMSIS分周器は無効です。

100：HSI/256、MSIS/1024、およびMSIS/4が生成され、TIM16、TIM17、およびLPTIM2により内部入力キャプチャとして選択できます。

101：HSI/256、MSIS/1024、およびMSIK/4が生成され、TIM16、TIM17、およびLPTIM2により内部入力キャプチャとして選択できます。

110：HSI/256、MSIK/1024、およびMSIS/4が生成され、TIM16、TIM17、およびLPTIM2により内部入力キャプチャとして選択できます。

111：HSI/256、MSIK/1024、およびMSIK/4が生成され、TIM16、TIM17、およびLPTIM2により内部入力キャプチャとして選択できます。

注： クロックソースの分周を選択または変更する前に、クロック分周を無効にする必要があります（TIMICSELを0xxに設定）。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27:26 **ICLKSEL[1:0]**：中間クロックソース選択

これらのビットは、OTG_FS、USB、およびSDMMCのクロックソースを選択するために使用されます。

00：HSI48クロックの選択

01：PLL2“Q”（pll2_q_ck）の選択

10：PLL1“Q”（pll1_q_ck）の選択

11：MSIKクロックの選択

ビット25:24 **FDCAN1SEL[1:0]**：FDCAN1カーネルクロックソースの選択

これらのビットは、FDCAN1カーネルクロックソースを選択するために使用されます。

00：HSEクロックの選択

01：PLL1“Q”（pll1_q_ck）の選択

10：PLL2“P”（pll2_p_ck）の選択

11：予約済み

ビット23:22 **SYSTICKSEL[1:0]**：SysTickクロックソース選択

これらのビットは、SysTickクロックソースを選択するために使用されます。

00：HCLK/8の選択

01：LSIの選択

10：LSEの選択

11：予約済み

注： LSEまたはLSIを選択する場合、AHB周波数をLSIまたはLSE周波数の少なくとも4倍にする必要があります。また、SysTick回路でHCLKを使用したLSEまたはLSIサンプリングにより、最大1サイクルのHCLKのジッタが適用されます。

ビット21:20 **SPI1SEL[1:0]**：SPI1カーネルクロックソースの選択

これらのビットは、SPI1カーネルクロックソースを選択するために使用されます。

00：PCLK2の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： SPI1は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

ビット19:18 **LPTIM2SEL[1:0]**：低電力タイマ2カーネルクロックソースの選択

これらのビットは、LPTIM2カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：LSIの選択

10：HSI16の選択

11：LSEの選択

注： LPTIM2は、HSIKERON = 1の状態でカーネルクロックがLSI、LSE、またはHSI16である場合のみ、STOP0およびSTOP1モードで機能します。

ビット17:16 **SPI2SEL[1:0]**：SPI2カーネルクロックソースの選択

これらのビットは、SPI2カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： SPI2は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

ビット15:14 **I2C4SEL[1:0]**：I2C4カーネルクロックソースの選択

これらのビットは、I2C4カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C4は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

ビット13:12 **I2C2SEL[1:0]**：I2C2カーネルクロックソースの選択

これらのビットは、I2C2カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C2は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

ビット11:10 **I2C1SEL[1:0]**：I2C1カーネルクロックソースの選択

これらのビットは、I2C1カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C1は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

ビット9:8 **UART5SEL[1:0]**：UART5カーネルクロックソースの選択

これらのビットは、UART5カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： UART5は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。

ビット7:6 **UART4SEL[1:0]**：UART4カーネルクロックソースの選択

これらのビットは、UART4カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： UART4は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。

ビット5:4 **USART3SEL[1:0]**：USART3カーネルクロックソースの選択

これらのビットは、USART3カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： USART3は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。

ビット3:2 **USART2SEL[1:0]**：USART2カーネルクロックソースの選択

これらのビットは、USART2カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： USART2は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット1:0 **USART1SEL[1:0]**：USART1カーネルクロックソースの選択

これらのビットは、USART1カーネルクロックソースを選択するために使用されます。

00：PCLK2の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： USART1は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。

11.8.47 RCCペリフェラル独立クロック設定レジスタ2（RCC_CCIPR2）

アドレスオフセット：0x0E4

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OTGHSSEL[1:0]		Res.	Res.	I2C6SEL [1:0]		I2C5SEL [1:0]		HSPI1SEL [1:0]		OCTOSPISEL[1:0]		Res.	LTDCSEL	USART6SEL [1:0]	
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DSISEL	SDMMCSEL	RNGSEL[1:0]		SAESSEL	SAI2SEL [2:0]			SAI1SEL [2:0]			Res.	Res.	MDF1SEL [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw

ビット31:30 **OTGHSSEL[1:0]**：OTG_HSPHYカーネルクロックソースの選択

これらのビットは、OTG_HSPHYカーネルクロックソースを選択するために使用されます。

00：HSEの選択

01：PLL1“P”（pll1_q_ck）の選択

10：HSE/2の選択

11：PLL1“P”は2分周されます（pll1_p_ck/2）

注： このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27:26 **I2C6SEL[1:0]**：I2C6カーネルクロックソースの選択

これらのビットは、I2C6カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C6は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。

このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット25:24 **I2C5SEL[1:0]**：I2C5カーネルクロックソースの選択

これらのビットは、I2C5カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C5は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0およびSTOP1モードで機能します。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット23:22 **HSPI1SEL[1:0]**：HSPI1カーネルクロックソースの選択

これらのビットは、HSPI1カーネルクロックソースを選択するために使用されます。

00：SYSCLKの選択

01：PLL1“Q”(pll1_q_ck)の選択、最大200MHzが可能

10：PLL2“Q”(pll2_q_ck)の選択、最大200MHzが可能

11：PLL3“R”(pll3_r_ck)の選択、最大200MHzが可能

注： このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット21:20 **OCTOSPISEL[1:0]**：OCTOSPI1およびOCTOSPI2カーネルクロックソースの選択

これらのビットは、OCTOSPI1およびOCTOSPI2カーネルクロックソースを選択するために使用されます。

00：SYSCLKの選択

01：MSIKの選択

10：PLL1“Q”(pll1_q_ck)の選択、最大200MHzが可能

11：PLL2“Q”(pll2_q_ck)の選択、最大200MHzが可能

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **LTDCSEL**：LTDCカーネルクロックソースの選択

このビットは、LTDCカーネルクロックソースを選択するために使用されます。

0：PLL3“R”（pll3_r_ck）の選択

1：PLL2“R”（pll2_r_ck）の選択

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット17:16 **USART6SEL[1:0]**：USART6カーネルクロックソースの選択

これらのビットは、USART6カーネルクロックソースを選択するために使用されます。

00：PCLK1の選択

01：SYSCLKの選択

10：HSI16の選択

11：LSEの選択

注： USART6は、カーネルクロックがHSI16またはLSEである場合のみ、STOP0およびSTOP1モードで機能します。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット15 **DSISEL**：DSIカーネルクロックソースの選択

このビットは、DSIカーネルクロックソースを選択するために使用されます。

0：PLL3“P”（pll3_p_ck）の選択

1：DSI PHY PLL出力の選択

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。
関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。
これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット14 **SDMMCSEL**：SDMMC1およびSDMMC2カーネルクロックソースの選択

このビットは、SDMMCカーネルクロックソースを選択するために使用されます。このビットは、リセット後およびSDMMCを有効にする前に変更することを推奨します。

0：ICLKクロックの選択

1：PLL1“P”（pll1_p_ck）の選択、48MHzを超える周波数が必要な場合（SDR50モード用）

ビット13:12 **RNGSEL[1:0]**：RNGカーネルクロックソースの選択

これらのビットは、RNGカーネルクロックソースを選択するために使用されます。

00：HSI48の選択

01：HSI48/2の選択（レンジ4で使用可能）

10：HSI16の選択

11：予約済み

ビット11 **SAESSEL**：SAESカーネルクロックソースの選択

このビットは、SAESカーネルクロックソースを選択するために使用されます。

0：SHSIの選択

1：SHSI/2の選択（レンジ4で使用可能）

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット10:8 **SAI2SEL[2:0]**：SAI2カーネルクロックソースの選択

これらのビットは、SAI2カーネルクロックソースを選択するために使用されます。

000：PLL2“P”（pll2_p_ck）の選択

001：PLL3“P”（pll3_p_ck）の選択

010：PLL1“P”（pll1_p_ck）の選択

011：入力ピンAUDIOCLKの選択

100：HSI16クロックの選択

その他：予約済み

注： 選択されたクロックが外部クロックで、このクロックが停止した場合、別のクロックへの切替えはできません。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット7:5 **SAI1SEL[2:0]**：SAI1カーネルクロックソースの選択

これらのビットは、SAI1カーネルクロックソースを選択するために使用されます。

000：PLL2“P”（pll2_p_ck）の選択

001：PLL3“P”（pll3_p_ck）の選択

010：PLL1“P”（pll1_p_ck）の選択

011：入力ピンAUDIOCLKの選択

100：HSI16クロックの選択

その他：予約済み

注： 選択されたクロックが外部クロックで、このクロックが停止した場合、別のクロックへの切替えはできません。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **MDF1SEL[2:0]**：MDF1カーネルクロックソースの選択

これらのビットは、MDF1カーネルクロックソースを選択するために使用されます。

000：HCLKの選択

001：PLL1“P”（pll1_p_ck）の選択

010：PLL3“Q”（pll3_q_ck）の選択

011：入力ピンAUDIOCLKの選択

100：MSIKクロックの選択

その他：予約済み

11.8.48 RCCペリフェラル独立クロック設定レジスタ3（RCC_CCIPR3）

アドレスオフセット：0x0E8

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1SEL [2:0]		
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DAC1SEL	ADCDACSEL[2:0]			LPTIM1SEL [1:0]		LPTIM34SEL [1:0]		I2C3SEL [1:0]		Res.	SPI3SEL [1:0]		LPUART1SEL [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **ADF1SEL[2:0]**：ADF1カーネルクロックソースの選択

これらのビットは、ADF1カーネルクロックソースを選択するために使用されます。

000：HCLKの選択

001：PLL1“P”（pll1_p_ck）の選択

010：PLL3“Q”（pll3_q_ck）の選択

011：入力ピンAUDIOCLKの選択

100：MSIKクロックの選択

その他：予約済み

注： ADF1は、カーネルクロックがAUDIOCLKまたはMSIKである場合のみ、STOP0、およびSTOP2モードで機能します。

ビット15 **DAC1SEL**：DAC1サンプルおよびホールドクロックソースの選択

このビットは、DAC1サンプルおよびホールドクロックソースを選択するために使用されます。

0：LSEの選択

1：LSIの選択

ビット14:12 **ADCDACSEL[2:0]**：ADC1、ADC2、ADC4、およびDAC1カーネルクロックソースの選択

これらのビットは、ADC1、ADC2、ADC4、およびDAC1カーネルクロックソースを選択するために使用されます。

000：HCLKクロックの選択

001：SYSCLKの選択

010：PLL2“R”（pll2_r_ck）の選択

011：HSEクロックの選択

100：HSI16クロックの選択

101：MSIKクロックの選択

その他：予約済み

注： ADC1、ADC2、ADC4、および DAC1 は、カーネルクロックが HSI16 または MSIK の場合のみ、STOP0、STOP1、およびSTOP2モードで機能します（STOP2モードで機能するのは、ADC4およびDAC1のみです）。

ビット11:10 **LPTIM1SEL[1:0]**：LPTIM1カーネルクロックソースの選択

これらのビットは、LPTIM1カーネルクロックソースを選択するために使用されます。

00：MSIKクロックの選択

01：LSIの選択

10：HSI16の選択

11：LSEの選択

注： LPTIM1は、HSIKERON = 1の状態でカーネルクロックがLSI、LSE、またはHSI16である場合、あるいはMSIKERON = 1の状態でカーネルクロックがMSIKである場合のみ、STOP0、STOP1、およびSTOP2モードで機能します。

ビット9:8 **LPTIM34SEL[1:0]**：LPTIM3およびLPTIM4カーネルクロックソースの選択

これらのビットは、LPTIM3およびLPTIM4カーネルクロックソースを選択するために使用されます。

00：MSIKクロックの選択

01：LSIの選択

10：HSIの選択

11：LSEの選択

注： LPTIM3およびLPTIM4は、HSIKERON = 1の状態でカーネルクロックがLSI、LSE、またはHSI16である場合、あるいはMSIKERON = 1の状態でカーネルクロックがMSIKである場合のみ、STOP0、STOP1、およびSTOP2モードで機能します。

ビット7:6 **I2C3SEL[1:0]**：I2C3カーネルクロックソースの選択

これらのビットは、I2C3カーネルクロックソースを選択するために使用されます。

00：PCLK3の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： I2C3は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0、STOP1、およびSTOP2モードで機能します。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:3 **SPI3SEL[1:0]**：SPI3カーネルクロックソースの選択

これらのビットは、SPI3カーネルクロックソースを選択するために使用されます。

00：PCLK3の選択

01：SYSCLKの選択

10：HSI16の選択

11：MSIKの選択

注： SPI3は、カーネルクロックがHSI16またはMSIKである場合のみ、STOP0、STOP1、およびSTOP2モードで機能します。

ビット2:0 **LPUART1SEL[2:0]**：LPUART1カーネルクロックソースの選択

これらのビットは、LPUART1カーネルクロックソースを選択するために使用されます。

000：PCLK3の選択

001：SYSCLKの選択

010：HSI16の選択

011：LSEの選択

100：MSIKの選択

その他：予約済み

注： LPUART1は、カーネルクロックがHSI16、LSE、またはMSIKである場合のみ、STOP0、STOP1、およびSTOP2モードで機能します。

11.8.49 RCCバックアップドメイン制御レジスタ（RCC_BDCR）

アドレスオフセット：0x00F0

バックアップドメインリセット値：0x0000 0000（STM32U575/585の場合）

バックアップドメインリセット値：0x0000 X000（他のSTM32U5シリーズデバイスの場合）

バックアップドメインのパワーオンリセットでのみリセットされるLSCOSEL、LSCOEN、およびBDRSTを除き、バックアップドメインリセットでリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

注： これらのレジスタビットは、コアドメイン外にあります。リセット後、これらのビットは書込み保護されるため、修正するにはPWR_BDCR1にDBPをセットする必要があります。（詳細は、セクション 10：電源制御（PWR）を参照）。これらのビット（LSCOSEL、LSCOEN、およびBDRSTを除く）は、バックアップドメインリセット後にのみリセットされます（セクション11.3.3を参照）。その他の内部または外部リセットは、これらのビットに影響しません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	LSIPREDIV	LSIRDY	LSION	LSCOSEL	LSCOEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDRST
			rw	rw	rw	rw	rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.	Res.	LSEGFON	LSESYSRDY	Res.	RTCSEL[1:0]		LSESYSEN	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
rw			rw	r		rw	rw	rw	r	rw	rw	rw	rw	r	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **LSIPREDIV**：ロースピードクロック分周器の設定

このビットは、LSI分周を有効にするために、ソフトウェアによってセット／クリアされます。このビットに書き込めるのは、LSIが無効なときだけです（LSION = 0かつLSIRDY = 0）。前にLSIが有効になっていた場合、LSIPREDIVに書き込む前に、LSIONビットをクリアしてから少なくとも60µs待つ必要があります（LSIを無効にする準備ができるまでの同期化時間）。LSIがIWDGやRTCによって使用されている場合、LSIPREDIVの変更はできません。

0：LSIは分周されません。

1：LSIは128分周されます。

ビット27 **LSIRDY**：LSIオシレータレディ状態

このビットは、LSIオシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSIONがクリアされた後、LSIRDYは、内部ロースピードオシレータクロックの3サイクル後にローになります。このビットは、LSION = 0の場合でも、LSIがIWDGまたはRTCによって使用されたときにセットされます。

0：LSIオシレータはレディ状態ではありません。

1：LSIオシレータはレディ状態です。

ビット26 **LSION**：LSIオシレータの有効化

このビットは、ソフトウェアによってセット／クリアされます。LSIオシレータは、LSIONビットをクリアした後、最大で60µs間無効になります。

0：LSIオシレータはオフです。

1：LSIオシレータはオンです。

ビット25 **LSCOSEL**：ロースピードクロック出力の選択

このビットは、ソフトウェアによってセット／クリアされます。

0：LSIクロックの選択

1：LSEクロックの選択

ビット24 **LSCOEN**：ロースピードクロック出力（LSCO）の有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：LSCOは無効です。

1：LSCOは有効です。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **BDRST**：バックアップドメインソフトウェアのリセット

このビットは、ソフトウェアによってセット／クリアされます。

0：リセットは有効ではありません。

1：バックアップドメイン全体をリセットします。

ビット15 **RTCEN**：RTCおよびTAMPクロックの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：RTCおよびTAMPクロックは無効です。

1：RTCおよびTAMPクロックは有効です。

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LSEGFON**：LSEクロックグリッチフィルタの有効化

このビットは、LSEグリッチフィルタを有効にするために、ハードウェアによってセット／クリアされます。このビットに書き込めるのは、LSEが無効なときだけです（LSEON = 0かつLSERDY = 0）。

0：LSEグリッチフィルタは無効です。

1：LSEグリッチフィルタは有効です。

ビット11 **LSESYSRDY**：LSEシステムクロック（LSESYS）レディ

このビットは、LSEシステムクロックが安定していることを示すために、ハードウェアによってセット／クリアされます。LSESYSENがセットされている場合、2LSEクロックサイクル後にこのLSESYSRDYフラグがセットされます。

LSEクロックはすでに有効かつ安定した状態（LSEONおよびLSERDYがセットされている）でなければなりません。

LSEONビットがクリアされた後、LSERDYは、外部ロースピードオシレータクロックの6サイクル後にローになります。

0：LSESYSクロックはレディ状態ではありません。

1：LSESYSクロックはレディ状態です。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **RTCSEL[1:0]**：RTCおよびTAMPクロックソースの選択

このビットは、RTCおよびTAMPのクロックソースを選択するために、ソフトウェアによってセットされます。選択が完了したRTCおよびTAMPクロックソースは、バックアップドメインがリセットされるか、LSEで障害が検出される（LSECSSDがセットされる）ことがない限り、変更できません。BDRSTビットを使用して、リセットできます。

00：クロックの選択なし

01：LSEオシレータクロックの選択

10：LSIオシレータクロックの選択

11：32分周のHSEオシレータクロックの選択

ビット7 **LSESYSEN**：LSEシステムクロック（LSESYS）の有効化

このビットは、RCCによって生成されるLSEシステムクロックを常に有効にするために、ソフトウェアによってセットされます。これは、ペリフェラルのソースクロックがLSEの場合、またはシステムレベルでLSCOSEL、MCO、MSIPLLモードのいずれかが必要なときに、ペリフェラルによって使用することができます。

0：LSEはRTC、TAMP、およびLSEのCSSでのみ使用できます。

1：LSEはその他の任意のペリフェラルまたは機能で使用できます。

ビット6 **LSECSSD**：LSEのCSSによる障害検出

このビットは、外部32kHzオシレータ（LSE）のCSSによって障害が検出されたことを示すために、ハードウェアによってセットされます。

0：LSEで障害は検出されていません。

1：LSEで障害が検出されました。

ビット5 **LSECSSON**：LSEのCSSの有効化

このビットは、LSEのCSSを有効にするために、ソフトウェアによってセットされます。このビットは、LSEオシレータを有効（LSEONビットを有効化）かつレディ状態（ハードウェアでLSERDYフラグをセット）にし、RTCSELビットを選択した後で、有効にする必要があります。

このビットは、一度有効にすると無効にすることはできません。ただし、LSEの障害検出（LSECSSD = 1）の後は例外です。この場合、LSECSSONビットをソフトウェアで無効にする必要があります。

0：LSEのCSSはオフです。

1：LSEのCSSはオンです。

ビット4:3 **LSEDRV[1:0]**：LSEオシレータの駆動能力

このビットフィールドは、LSEオシレータの駆動能力を調整するために、ソフトウェアによってセットされます。このビットに書き込めるのは、外部32kHzオシレータが無効なときだけです（LSEON = 0かつLSERDY = 0）。

00：「Xtalモード」低駆動

01：「Xtalモード」中低駆動

10：「Xtalモード」中高駆動

11：「Xtalモード」高駆動

注： オシレータは、バイパスモードでない場合、「Xtalモード」です。

ビット2 **LSEBYP**：LSEオシレータバイパス

このビットは、デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。このビットに書き込めるのは、外部32kHzオシレータが無効なときだけです（LSEON = 0かつLSERDY = 0）。

0：LSEオシレータはバイパスされません。

1：LSEオシレータはバイパスされます。

ビット1 **LSERDY**：LSEオシレータレディ状態

このビットは、外部32kHzオシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSEONがクリアされた後、LSERDYは、外部ロースピードオシレータクロックの6サイクル後にローになります。

0：LSEオシレータはレディ状態ではありません。

1：LSEオシレータはレディ状態です。

ビット0 **LSEON**：LSEオシレータの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：LSEオシレータはオフです。

1：LSEオシレータはオンです。

11.8.50 RCC制御／ステータスレジスタ（RCC_CSR）

アドレスオフセット：0x0F4

リセット値：0x0C00 4400

システムリセットによってリセットされます。ただし、リセットフラグは電源リセット時のみリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	BORRSTF	PINRSTF	OBLRSTF	Res.	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r		rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSISSRANGE[3:0]				MSIKSRANGE[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw								

ビット31 **LPWRRSTF**：低電力リセットフラグ

このビットは、STOP、STANDBY、またはSHUTDOWNモードへの移行によりリセットが発生したときにハードウェアによってセットされますが、これに対応するNRST_STOP、NRST_STBY、またはNRST_SHDWオプションビットはクリアされます。このビットは、RMVFビットに書き込むことによってクリアされます。

0：不正な低消費電力モードリセットは発生していません。

1：不正な低消費電力モードリセットが発生しました。

ビット30 **WWDGRSTF**：ウィンドウ型ウォッチドッグリセットフラグ

このビットは、ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。RMVFビットに書き込むことによってクリアされます。

0：ウィンドウ型ウォッチドッグリセットは発生していません。

1：ウィンドウ型ウォッチドッグリセットが発生しました。

ビット29 **IWDGRSTF**：独立型ウォッチドッグリセットフラグ

このビットは、独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。RMVFビットに書き込むことによってクリアされます。

0：独立型ウォッチドッグリセットは発生していません。

1：独立型ウォッチドッグリセットが発生しました。

ビット28 **SFTRSTF**：ソフトウェアリセットフラグ

このビットは、ソフトウェアリセットが発生したときに、ハードウェアによってセットされます。RMVFに書き込むことによってクリアされます。

0：ソフトウェアリセットは発生していません。

1：ソフトウェアリセットが発生しました。

ビット27 **BORRSTF**：ブラウンアウト･リセット（SHUTDOWNモードの終了リセット）フラグ

このビットは、ブラウンアウト･リセット（SHUTDOWNモードの終了リセット）が発生したときに、ハードウェアによってセットされます。RMVFビットに書き込むことによってクリアされます。

0：BOR/SHUTDOWNモードの終了リセットは発生していません。

1：BOR/SHUTDOWNモードの終了リセットが発生しました。

ビット26 **PINRSTF**：NRSTピンリセットフラグ

このビットは、NRSTピンからリセットが発生したときに、ハードウェアによってセットされます。RMVFビットに書き込むことによってクリアされます。

0：NRSTピンからのリセットは発生していません。

1：NRSTピンからのリセットが発生しました。

ビット25 **OBLRSTF**：オプションバイトローダリセットフラグ

このビットは、オプションバイトローディングからのリセットが発生したときに、ハードウェアによってセットされます。RMVFビットに書き込むことによってクリアされます。

0：オプションバイトローディングからのリセットは発生していません。

1：オプションバイトローディングからのリセットが発生しました。

ビット24 予約済みであり、リセット値に保持する必要があります。

ビット23 **RMVF**：削除リセットフラグ

このビットは、リセットフラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：リセットフラグをクリアします。

ビット22:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **MSISSRANGE[3:0]**：STANDBYモード後のMSISレンジ

このビットフィールドは、起動時にMSIS周波数を選択するために、ソフトウェアによってセットされます。このレンジは、STANDBYモード終了後MSIRGSELがセットされるまで使用されます。NRSTピンまたはパワーオンリセット後、あるいはSHUTDOWNモードの終了時、レンジは常に4MHzとなります。MSISSRANGEは、MSIRGSEL = 1のときのみ書き込むことができます。

0100：レンジ4（約4MHz、リセット値）

0101：レンジ5（約2MHz）

0110：レンジ6（約1.33MHz）

0111：レンジ7（約1MHz）

1000：レンジ8（約3.072MHz）

その他：予約済み

注： このビットフィールド変更によって、現在のMSIS周波数は変わりません。

ビット11:8 **MSIKSRANGE[3:0]**：STANDBYモード後のMSIKレンジ

このビットは、起動時にMSIK周波数を選択するために、ソフトウェアによってセットされます。このレンジは、STANDBYモード終了後MSIRGSELがセットされるまで使用されます。NRSTピンまたはパワーオンリセット後、あるいはSHUTDOWN モードの終了時、レンジは常に4MHz となります。MSIKSRANGEは、MSIRGSEL = 1のときのみ書き込むことができます。

0100：レンジ4（約4MHz、リセット値）

0101：レンジ5（約2MHz）

0110：レンジ6（約1.33MHz）

0111：レンジ7（約1MHz）

1000：レンジ8（約3.072MHz）

その他：予約済み

注： このビットフィールド変更によって、現在のMSIK周波数は変わりません。

ビット7:0 予約済みであり、リセット値に保持する必要があります。

11.8.51 RCCセキュア設定レジスタ（RCC_SECCFGR）

アドレスオフセット：0x110

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

システムがセキュアなとき（TZEN = 1）、このレジスタは、SPRIV = 1の場合はセキュア特権アクセスによってのみ書き込むことができ、SPRIV = 0の場合はセキュア特権または非特権アクセスによって書き込むことができます。非セキュア書込みアクセスは、不正アクセスイベントを生成し、データは書き込まれません。このレジスタは、セキュアまたは非セキュア、特権または非特権アクセスによって読み出すことができます。システムがセキュアでないとき（TZEN = 0）、このレジスタは0として読み出され、レジスタ書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RMVFSEC	HSI48SEC	ICLKSEC	PLL3SEC	PLL2SEC	PLL1SEC	PRESCSEC	SYSCLKSEC	LSESEC	LSISEC	MSISEC	HSESEC	HSISEC
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **RMVFSEC**：リセットフラグ解除のセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット11 **HSI48SEC**：HSI48クロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット10 **ICLKSEC**：中間クロックソース選択のセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット9 **PLL3SEC**：PLL3クロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット8 **PLL2SEC**：PLL2クロック設定ビットとステータスビットのセキュリティ
ソフトウェアでセット／クリアされます。
0：非セキュア
1：セキュア

ビット7 **PLL1SEC**：PLL1クロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット6 **PRESCSEC**：AHBx/APBxプリスケーラ設定ビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット5 **SYSCLKSEC**：SYSCLKクロック選択、STOPWUCKビット、MCOクロック出力設定のセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット4 **LSESEC**：LSEクロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット3 **LSISEC**：LSIクロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット2 **MSISEC**：MSIクロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット1 **HSESEC**：HSEクロック設定ビットとステータスビット、およびHSE_CSSのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

ビット0 **HSISEC**：HSIクロック設定ビットとステータスビットのセキュリティ
このビットは、ソフトウェアによってセット／リセットされます。
0：非セキュア
1：セキュア

11.8.52 RCC特権設定レジスタ(RCC_PRIVCFGR)

アドレスオフセット：0x114

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、特権アクセスによってのみ書き込むことができます。特権または非特権アクセスによって読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **NSPRIV**：RCC非セキュア機能の特権設定

このビットは、ソフトウェアによってセット／リセットされます。書込みは、セキュアまたは非セキュアの特権アクセスによってのみできます。

0：RCC非セキュア機能の読み書きは、特権または非特権アクセスによって行うことができます。

1：RCC非セキュア機能の読み書きは、特権アクセスによってのみ行うことができます。

ビット0 **SPRIV**：RCCセキュア機能の特権設定

このビットは、ソフトウェアによってセット／リセットされます。これはセキュア特権アクセスによってのみ書込みが可能です。

0：RCCセキュア機能の読み書きは、特権または非特権アクセスによって行うことができます。

1：RCCセキュア機能の読み書きは、特権アクセスによってのみ行うことができます。

11.8.53 RCCレジスタ･マップ

表119.RCCレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RCC_CR	Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	HSEEXT	CSSON	HSEBYP	HSERDY	HSEON	SHSIRDY	SHSION	HSI48RDY	HSI48ON	Res.	HSIRDY	HSIKERON	HSION	MSIPLLFAST	MSIPLLSEL	MSIKRDY	MSIKON	MSIPLLEN	MSISRDY	MSIKERON	MSISON
	リセット値			0	0	0	0	0	0				0	0	0	0	0	0	0	0	0		0	0	0	0	0	1	1	0	1	0	1
0x004	予約済み	予約済み																															
0x008	RCC_ICSCR1	MSISRANGE [3:0]				MSIKRANGE [3:0]				MSIRGSEL	MSIBIAS	Res.	Res.	MSICAL0 [4:0]					MSICAL1 [4:0]					MSICAL2 [4:0]					MSICAL3 [4:0]				
	リセット値	0	1	0	0	0	1	0	0	0	0			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x00C	RCC_ICSCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSITRIM0 [4:0]					MSITRIM1 [4:0]					MSITRIM2 [4:0]					MSITRIM3 [4:0]				
	リセット値													1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0
0x010	RCC_ICSCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSITRIM[4:0]					Res.	Res.	Res.	Res.	HSICAL[11:0]											
	リセット値												1	0	0	0	0					X	X	X	X	X	X	X	X	X	X	X	X
0x014	RCC_CRRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL [8:0]								
	リセット値																								X	X	X	X	X	X	X	X	X
0x018	予約済み	予約済み																															
0x01C	RCC_CFGR1	Res.	MCOPRE [2:0]			MCOSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOPKERWUCK	STOPWUCK	SWS[1:0]		SW[1:0]	
	リセット値		0	0	0	0	0	0	0																			0	0	0	0	0	0
0x020	RCC_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APB2DIS	APB1DIS	AHB2DIS2	AHB2DIS1	AHB1DIS	Res.	DPRE[2:0]			Res.	PPRE2 [2:0]			Res.	PPRE1 [2:0]			HPRE[3:0].			
	リセット値												0	0	0	0	0		1	1	0		0	0	0		0	0	0	0	0	0	0
0x024	RCC_CFGR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APB3DIS	AHB3DIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPRE3 [2:0]			Res.	Res.	Res.	Res.
	リセット値															0	0										0	0	0				
0x028	RCC_ PLL1CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL1REN	PLL1QEN	PLL1PEN	PLL1MBOOST [3:0]				PLL1M[3:0]				Res.	Res.	Res.	PLL1FRACEN	PLL1RGE [1:0]		PLL1SRC [1:0]	
	リセット値														0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x02C	RCC_ PLL2CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL2REN	PLL2QEN	PLL2PEN	Res.	Res.	Res.	Res.	PLL2M [3:0]				Res.	Res.	Res.	PLL2FRACEN	PLL2RGE [1:0]		PLL2SRC [1:0]	
	リセット値														0	0	0					0	0	0	0				0	0	0	0	0

表119.RCCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x030	RCC_PLL3CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL3REN	PLL3QEN	PLL3PEN	Res.	Res.	Res.	Res.	PLL3M [3:0]				Res.	Res.	Res.	PLL3FRACEN	PLL3RGE [1:0]		PLL3SRC [1:0]	
	リセット値														0	0	0					0	0	0	0				0	0	0	0	0
0x034	RCC_PLL1DIVR	Res.	PLL1R [6:0]							Res.	PLL1Q [6:0]							PLL1P [6:0]							PLL1N [8:0]								
	リセット値		0	0	1	0	0	0	1		0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x038	RCC_PLL1FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL1FRACN [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x03C	RCC_PLL2DIVR	Res.	PLL2R [6:0]							Res.	PLL2Q [6:0]							PLL2P [6:0]							PLL2N [8:0]								
	リセット値		0	0	1	0	0	0	1		0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x040	RCC_PLL2FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL2FRACN [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x044	RCC_PLL3DIVR	Res.	PLL3R [6:0]							Res.	PLL3Q [6:0]							PLL3P [6:0]							PLL3N [8:0]								
	リセット値		0	0	1	0	0	0	1		0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x048	RCC_PLL3FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLL3FRACN [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x04C	予約済み	予約済み																															
0x050	RCC_CIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SHSIRDYIE	MSIKRDYIE	Res.	Res.	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	HSERDYIE	HSIRDYIE	MSISRDYIE	LSERDYIE	LSIRDYIE
	リセット値																				0	0			0	0	0	0	0	0	0	0	0
0x054	RCC_CIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SHSIRDYF	MSIKRDYF	CSSF	Res.	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	HSERDYF	HSIRDYF	MSISRDYF	LSERDYF	LSIRDYF
	リセット値																				0	0	0		0	0	0	0	0	0	0	0	0
0x058	RCC_CICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SHSIRDYC	MSIKRDYC	CSSC	Res.	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	HSERDYC	HSIRDYC	MSISRDYC	LSERDYC	LSIRDYC
	リセット値																				0	0	0		0	0	0	0	0	0	0	0	0
0x05C	予約済み	予約済み																															
0x060	RCC_AHB1RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPU2DRST	GFXMMURST	DMA2DRST	RAMCFGRST	TSCRST	JPEGRST	Res.	Res.	CRCRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MDF1RST	FMACRST	CORDICRST	GPDMA1RST
	リセット値												0	0	0	0	0	0			0									0	0	0	0
0x064	RCC_AHB2RSTR1	Res.	Res.	Res.	SDMMC2RST	SDMMC1RST	Res.	Res.	OTFDEC2RST	OTFDEC1RST	Res.	OCTOSPIMRST	SAESRST	PKARST	RNGRST	HASHRST	AESRST	Res.	OTGRST	Res.	DCMI_PSSIRST	Res.	ADC12RST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	リセット値				0	0			0	0		0	0	0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0

表119.RCCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x068	RCC_AHB2RSTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPI1RST	Res.	Res.	Res.	OCTOSPI2RST	Res.	Res.	Res.	OCTOSPI1RST	Res.	Res.	Res.	FSMCRST
	リセット値																				0				0				0				0
0x06C	RCC_AHB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1RST	LPDMA1RST	Res.	Res.	DAC1RST	ADC4RST	Res.	Res.	Res.	Res.	LPGPIO1RST
	リセット値																						0	0			0	0					0
0x070	予約済み	予約済み																															
0x074	RCC_APB1RSTR1	Res.	Res.	Res.	Res.	Res.	Res.	USART6RST	CRSRST	Res.	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	Res.	Res.	SPI2RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
	リセット値							0	0		0	0	0	0	0	0			0									0	0	0	0	0	0
0x078	RCC_APB1RSTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1RST	Res.	I2C6RST	I2C5RST	LPTIM2RST	Res.	Res.	Res.	I2C4RST	Res.
	リセット値									0														0		0	0	0				0	
0x07C	RCC_APB2RSTR	Res.	Res.	Res.	Res.	DSIRST	LTDCRST	GFXTIMRST	USBRST	Res.	SAI2RST	SAI1RST	Res.	Res.	TIM17RST	TIM16RST	TIM15RST	Res.	USART1RST	TIM8RST	SPI1RST	TIM1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0	0	0		0	0			0	0	0		0	0	0	0											
0x080	RCC_APB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VREFRST	Res.	Res.	Res.	Res.	COMPRST	OPAMPRST	LPTIM4RST	LPTIM3RST	LPTIM1RST	Res.	Res.	Res.	I2C3RST	LPUART1RST	SPI3RST	Res.	Res.	Res.	SYSCFGRST	Res.
	リセット値												0						0	0	0	0				0	0	0				0	
0x084	予約済み	予約済み																															
0x088	RCC_AHB1ENR	SRAM1EN	DCACHE1EN	Res.	BKPSRAMEN	Res.	Res.	Res.	GTZC1EN	Res.	Res.	DCACHE2EN	GPU2DEN	GFXMMUEN	DMA2DEN	RAMCFGEN	TSCEN	JPEGEN	Res.	Res.	CRCEN	Res.	Res.	Res.	FLASHEN	Res.	Res.	Res.	Res.	MDF1EN	FMACEN	CORDICEN	GPDMA1EN
	リセット値	1	1		1				0			1	0	0	0	0	0	0			0				1				0	0	0	0
0x08C	RCC_AHB2ENR1	SRAM3EN	SRAM2EN	Res.	SDMMC2EN	SDMMC1EN	Res.	Res.	OTFDEC2EN	OTFDEC1EN	Res.	OCTOSPIMEN	SAESEN	PKAEN	RNGEN	HASHEN	AESEN	OTGHSPHYEN	OTGEN	Res.	DCMI_PSSIEN	Res.	ADC12EN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値	1	1		0	0			0	0		0	0	0	0	0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0
0x090	RCC_AHB2ENR2	SRAM5EN	SRAM6EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPI1EN	Res.	Res.	Res.	OCTOSPI2EN	Res.	Res.	Res.	OCTOSPI1EN	Res.	Res.	Res.	FSMCEN
	リセット値	1	1																		0				0				0				0
0x094	RCC_AHB3ENR	SRAM4EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GTZC2EN	Res.	ADF1EN	LPDMA1EN	Res.	Res.	DAC1EN	ADC4EN	Res.	Res.	PWREN	Res.	LPGPIO1EN
	リセット値	1																			0		0	0			0	0			0		0
0x098	予約済み	予約済み																															

表119.RCCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x09C	RCC_APB1ENR1	Res.	Res.	Res.	Res.	Res.	Res.	USART6EN	CRSEN	Res.	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	Res.	Res.	SPI2EN	Res.	Res.	WWDGEN	Res.	Res.	Res.	Res.	Res.	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	リセット値							0	0		0	0	0	0	0	0			0			0						0	0	0	0	0	0
0x0A0	RCC_APB1ENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1EN	Res.	I2C6EN	I2C5EN	LPTIM2EN	Res.	Res.	Res.	I2C4EN	Res.
	リセット値									0														0		0	0	0				0	
0x0A4	RCC_APB2ENR	Res.	Res.	Res.	Res.	DSIEN	LTDCEN	GFXTIMEN	USBEN	Res.	SAI2EN	SAI1EN	Res.	Res.	TIM17EN	TIM16EN	TIM15EN	Res.	USART1EN	TIM8EN	SPI1EN	TIM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0	0	0		0	0			0	0	0		0	0	0	0											
0x0A8	RCC_APB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBEN	VREFEN	Res.	Res.	Res.	Res.	COMPEN	OPAMPEN	LPTIM4EN	LPTIM3EN	LPTIM1EN	Res.	Res.	Res.	I2C3EN	LPUART1EN	SPI3EN	Res.	Res.	Res.	SYSCFGEN	Res.
	リセット値											0	0						0	0	0	0				0	0	0				0	
0x0AC	予約済み	予約済み																															
0x0B0	RCC_AHB1SMENR	SRAM1SMEN	DCACHE1SMEN	ICACHESMEN	BKPSRAMSMEN	Res.	Res.	Res.	GTZC1SMEN	Res.	Res.	DCACHE2SMEN	GPU2DSMEN	GFXMMUSMEN	DMA2DSMEN	RAMCFGSMEN	TSCSMEN	JPEGSMEN	Res.	Res.	CRCSMEN	Res.	Res.	Res.	Res.	FlashSMEN	Res.	Res.	Res.	MDF1SMEN	FMACSMEN	CORDICSMEN	GPDMA1SMEN
	リセット値	1	1	1	1				1			1	1	1	1	1	1	1			1				1				1	1	1	1
0x0B4	RCC_AHB2SMENR1	SRAM3SMEN	SRAM2SMEN	Res.	SDMMC2SMEN	SDMMC1SMEN	Res.	Res.	OTFDEC2SMEN	OTFDEC1SMEN	Res.	OCTOSPIMSMEN	SAESSMEN	PKASMEN	RNGSMEN	HASHSMEN	AESSMEN	OTGHSPHYSMEN	OTGSMEN	Res.	DCMI_PSSISMEN	Res.	ADC12SMEN	GPIOJSMEN	GPIOISMEN	GPIOHSMEN	GPIOGSMEN	GPIOFSMEN	GPIOESMEN	GPIODSMEN	GPIOCSMEN	GPIOBSMEN	GPIOASMEN
	リセット値	1	1		1	1			1	1		1	1	1	1	1	1	1	1		1		1	1	1	1	1	1	1	1	1	1	1
0x0B8	RCC_AHB2SMENR2	SRAM5SMEN	SRAM6SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPI1SMEN	Res.	Res.	Res.	OCTOSPI2SMEN	Res.	Res.	Res.	OCTOSPI1SMEN	Res.	Res.	Res.	FSMCSMEN
	リセット値	1	1																		1				1				1				1
0x0BC	RCC_AHB3SMENR	SRAM4SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GTZC2SMEN	Res.	ADF1SMEN	LPDMA1SMEN	Res.	Res.	DAC1SMEN	ADC4SMEN	Res.	Res.	PWRSMEN	Res.	LPGPIO1SMEN
	リセット値	1																			1		1	1			1	1					1
0x0C0	予約済み	予約済み																															
0x0C4	RCC_APB1SMENR1	Res.	Res.	Res.	Res.	Res.	Res.	USART6SMEN	CRSSMEN	Res.	I2C2SMEN	I2C1SMEN	UART5SMEN	UART4SMEN	USART3SMEN	USART2SMEN	Res.	Res.	SPI2SMEN	Res.	Res.	WWDGSMEN	Res.	Res.	Res.	Res.	Res.	TIM7SMEN	TIM6SMEN	TIM5SMEN	TIM4SMEN	TIM3SMEN	TIM2SMEN
	リセット値							1	1		1	1	1	1	1	1			1			1						1	1	1	1	1	1

表119.RCCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C8	RCC_APB1SMENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCAN1SMEN	Res.	I2C6SMEN	I2C5SMEN	LPTIM2SMEN	Res.	Res.	Res.	I2C4SMEN	Res.
	リセット値									1														1		1	1	1				1	
0x0CC	RCC_APB2SMENR	Res.	Res.	Res.	Res.	DSISMEN	LTDCSMEN	GFXTIMSMEN	USBSMEN	Res.	SAI2SMEN	SAI1SMEN	Res.	Res.	TIM17SMEN	TIM16SMEN	TIM15SMEN	Res.	USART1SMEN	TIM8SMEN	SPI1SMEN	TIM1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値					1	1	1	1		1	1			1	1	1		1	1	1	1												
0x0D0	RCC_APB3SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBSMEN	VREFSMEN	Res.	Res.	Res.	Res.	COMPSMEN	OPAMPSMEN	LPTIM4SMEN	LPTIM3SMEN	LPTIM1SMEN	Res.	Res.	Res.	I2C3SMEN	LPUART1SMEN	SPI3SMEN	Res.	Res.	Res.	SYSCFGSMEN	Res.
	リセット値											1	1					1	1	1	1	1				1	1	1				1	
0x0D4	予約済み	予約済み																															
0x0D8	RCC_SRDAMR	SRAM4AMEN	Res.	ADF1AMEN	LPDMA1AMEN	DAC1AMEN	LPGPIO1AMEN	ADC4AMEN	Res.	Res.	Res.	RTCAPBAMEN	VREFAMEN	Res.	Res.	Res.	Res.	COMPAMEN	OPAMPAMEN	LPTIM4AMEN	LPTIM3AMEN	LPTIM1AMEN	Res.	Res.	Res.	I2C3AMEN	LPUART1AMEN	SPI3AMEN	Res.	Res.	Res.	Res.	Res.
	リセット値	0		0	0	0	0	0				0	0					0	0	0	0	0				0	0	0					
0x0E0	RCC_CCIPR1	TIMICSEL [2:0]			Res.	ICLKSEL [1:0]		FDCANSEL [1:0]		SYSTICKSEL [1:0]		SPI1SEL [1:0]		LPTIM2SEL[1:0]		SPI2SEL [1:0]		I2C4SEL [1:0]		I2C2SEL [1:0]		I2C1SEL [1:0]		UART5SEL [1:0]		UART4SEL [1:0]		USART3SEL [1:0]		USART2SEL [1:0]		USART1SEL [1:0]	
	リセット値	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0E4	RCC_CCIPR2	OTGHSSEL [1:0]			Res.	I2C6SEL [1:0]		I2C5SEL [1:0]		HSPI1SEL [1:0]		OCTOSPISEL[1:0]		Res.	LTDCSEL	USART6SEL [1:0]		DSISEL	SDMMCSEL	RNGSEL [1:0]		SAESSEL	SAI2SEL [2:0]		SAI1SEL [2:0]		Res.	Res.	MDF1SEL [2:0]					
	リセット値	0	0			0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0E8	RCC_CCIPR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADF1SEL [2:0]			DAC1SEL	ADCDACSEL [2:0]			LPTIM1SEL [1:0]		LPTIM34SEL[1:0]		I2C3SEL [1:0]		Res.	SPI3SEL1:0]		LPUART1SEL [2:0]		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x0F0	RCC_BDCR	Res.	Res.	Res.	LSIPREDIV	LSIRDY	LSION	LSCOSEL	LSCOEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDRST	RTCEN	Res.	Res.	LSEGFON	LSESYSRDY	Res.	RTCSEL[1:0]		LSESYSEN	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
	リセット値				0	0	0	0	0								0	0			0	0		0	0	0	0	0	0	0	0	0	0
0x0F4	RCC_CSR	LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	BORRSTF	PINRSTF	OBLRSTF	Res.	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSISSRANGE [3:0]				MSIKSRANGE [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	1	1	0		0								0	1	0	0	0	1	0	0								
0x0F8 - 0x10C	予約済み	予約済み																															

表119.RCCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x110	RCC_SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RMVFSEC	HSI48SEC	ICLKSEC	PLL3SEC	PLL2SEC	PLL1SEC	PRESCSEC	SYSCLKSEC	LSESEC	LSISEC	MSISEC	HSESEC	HSISEC
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x114	RCC_PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSPRIV	SPRIV
	リセット値																															0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

12 クロックリカバリシステム（CRS）

12.1 概要

クロックリカバリシステム（CRS）は、高精度トリミング可能な内蔵RCオシレータHSI48に作用する、高度なデジタルコントローラです。CRSは、選択可能な同期信号との比較に基づいてオシレータ出力周波数を評価するための強力な手段を提供しています。CRSは、測定した周波数の誤差値に基づいて自動トリミング調整をおこなうことができ、手動トリミングも可能です。

CRSは、USBペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USBホストによって1ms間隔で送信されるUSBバス上のフレーム開始（SOF）パケット信号から同期信号を取得できます。

同期信号は、LSEオシレータ出力から生成することができ、ユーザソフトウェアによって生成することもできます。

12.2 CRSの主な機能

- 次のプログラム可能なプリスケーラと極性を持つ、選択可能な同期ソース：
  - LSEオシレータ出力
  - USB SOFパケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU不要の自動オシレータトリミング機能
- より迅速なスタートアップ収束のための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16bitの周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割込み／イベント：
  - 期待される同期（ESYNC）
  - 同期OK（SYNCOK）
  - 同期警告（SYNCWARN）
  - 同期またはトリミングエラー（ERR）

12.3 CRSの実装

表120.CRSの機能

機能	CRS1
TRIM幅	7bit

表121.STM32U535/545/575/585用の CRS内部入力／出力信号

内部信号名	信号タイプ	説明
crs_sync_in_1	入力	00：GPIO AFをSYNC信号ソースとして選択します。
crs_sync_in_2	入力	01：LSEをSYNC信号ソースとして選択します。
crs_sync_in_3	入力	10：USB SOFをSYNC信号ソースとして選択します（デフォルト）。
crs_sync_in_4	入力	11：予約済み

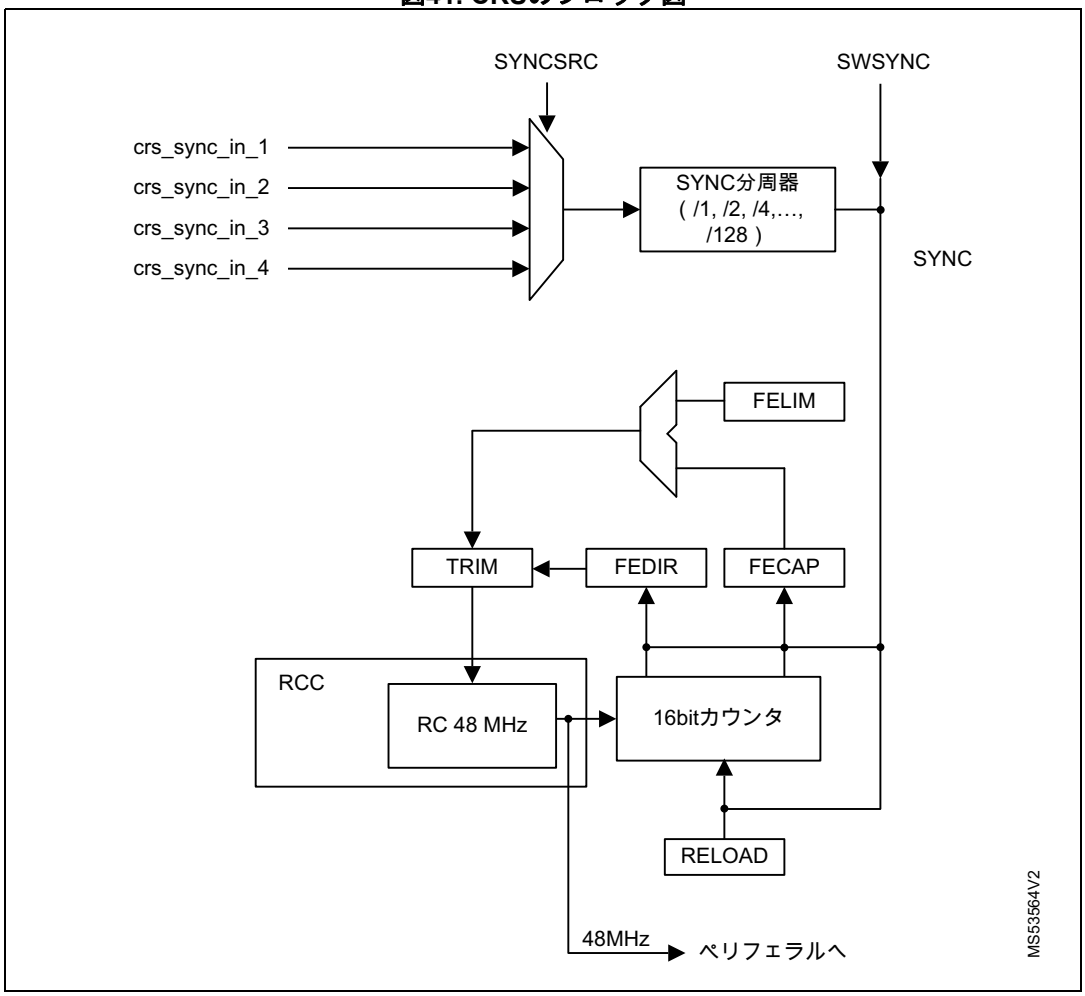
表122.STM32U59x/5Ax/5Fx/5Gx用のCRS内部入力／出力信号

内部信号名	信号タイプ	説明
crs_sync_in_1	入力	00：GPIO AFをSYNC信号ソースとして選択します。
crs_sync_in_2	入力	01：LSEをSYNC信号ソースとして選択します。
crs_sync_in_3	入力	01：予約済み（デフォルト）
crs_sync_in_4	入力	11：予約済み

12.4 CRSの機能詳細

12.4.1 CRSブロック図

図41. CRSのブロック図



12.4.2 同期入力

CRS_CFGRレジスタから選択可能なCRS同期（SYNC）ソースは、LSEクロックからの信号またはUSB SOF信号です。より安定したSYNC入力を得るために、グリッチをフィルタするために、シンプルなデジタルフィルタ（3対2の多数決方式、HSI48クロックによるサンプリング方法）が実装されています。このソース信号には設定可能な極性があり、プログラム可能なバイナリプリスケーラによって分周して、適切な周波数範囲（通常1kHz前後）の同期信号を得ることができます。

CRS同期ソース設定の詳細については、セクション12.7.2を参照してください。

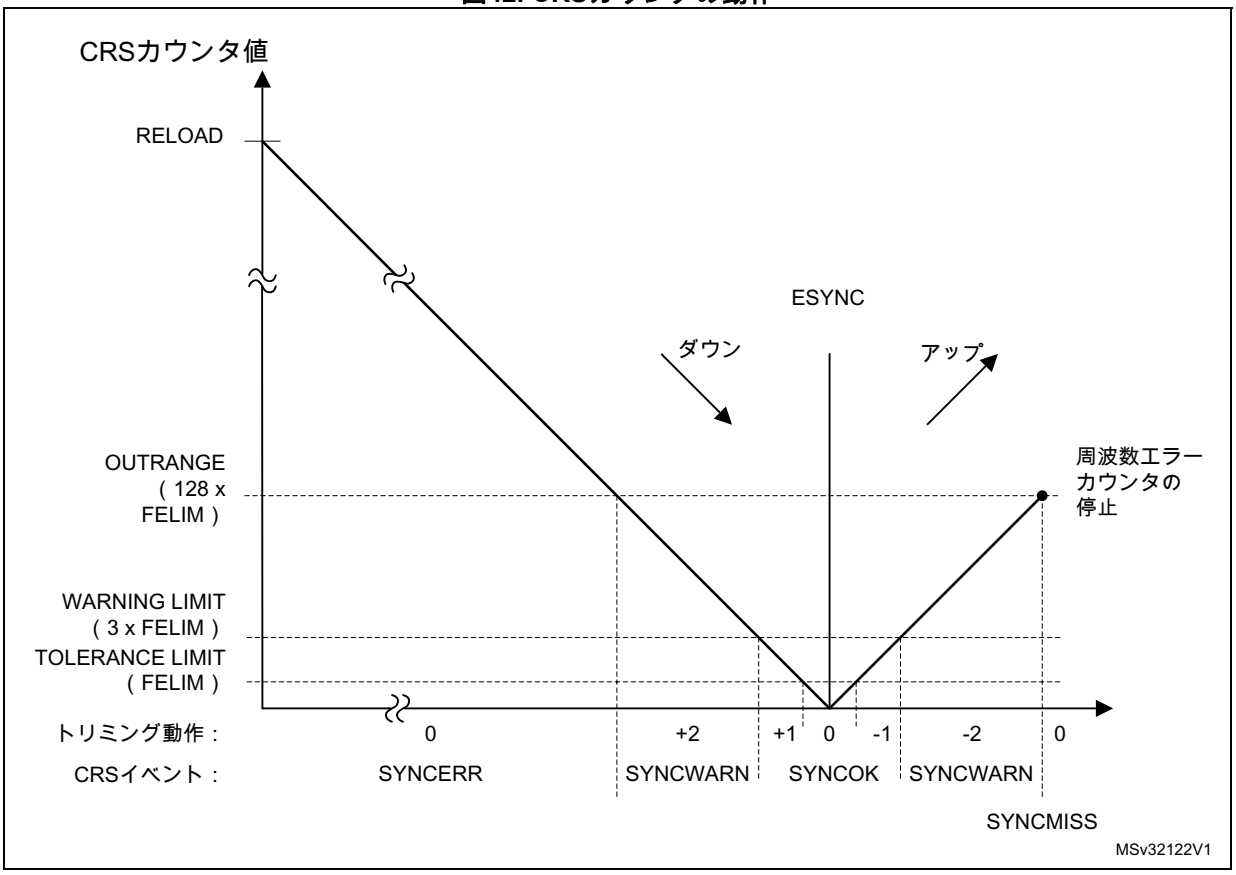
CRS_CRレジスタのSWSYNCビットをセットして、ソフトウェアで同期イベントを生成することもできます。

12.4.3 周波数誤差測定

この周波数誤差カウンタは、SYNCイベントごとにRELOAD値が再ロードされる16bitのダウン／アップカウンタです。0値に達するまでカウントダウンし、0値に達すると、ESYNC（期待される同期）イベントが生成されます。次に、OUTRANGE制限までカウントアップして、最終的に停止し（SYNCイベントが受信されなかった場合）、SYNCMISSイベントを生成します。OUTRANGE制限は、周波数誤差制限（CRS_CFGRレジスタのFELIMフィールド）に128を乗算した値として定義されます。

SYNCイベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向がFECAP（周波数誤差キャプチャ）フィールドとCRS_ISRレジスタのFEDIR（周波数誤差方向）ビットに保管されます。カウントダウンフェーズ中に（0値に達する前に）SYNCイベントが検出されると、実際の周波数がターゲットより低いことを意味します（TRIM値を増やす必要があります）。カウントアップフェーズ中に検出されると、実際の周波数が高いことを意味します（TRIM値を減らす必要があります）。

図42. CRSカウンタの動作



12.4.4 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGRレジスタのFELIMフィールドに直接与えられるTOLERANCE LIMIT
- WARNING LIMIT。3 * FELIM値として定義されます。
- OUTRANGE（エラー制限）。128 * FELIM値として定義されます。

この比較の結果は、ステータスの表示を生成し、CRS_CRレジスタのAUTOTRIMENビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
  - SYNCOKステータスが示されます。
  - TRIM値は、AUTOTRIMモードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1回のトリミングステップによる調整で、十分適切なTRIM値に達することができます。
  - SYNCOKステータスが示されます。
  - AUTOTRIMモードで、1回のトリミングステップによってTRIM値が調整されます。
- 周波数誤差が警告制限を上回るか同じであり、エラー制限を下回る場合は、より強力なトリミング動作が必要であり、次の周期までに適切なTRIM値に達しないリスクがあります。
  - SYNCWARNステータスが示されます。
  - AUTOTRIMモードで、2回のトリミングステップによってTRIM値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合、周波数はトリミング範囲外です。これは、SYNC入力がクリーンでないとき、または、一部のSYNCパルスが欠落しているときにも発生する可能性があります（たとえば、1つのUSB SOFが破損しているとき）。
  - SYNCERRまたはSYNCMISSステータスが示されます。
  - TRIM値は、AUTOTRIMモードでは変更されません。

注： **TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM値は制限に設定され、TRIMOVFステータスが示されます。**

AUTOTRIMモード（CRS_CRレジスタのAUTOTRIMENビットがセット）では、CRS_CRのTRIMフィールドがハードウェアによって調整され、読出し専用になります。

12.4.5 CRSの初期化と設定

RELOAD値

再ロード値は、ターゲット周波数とプリスケーリング後の同期ソースの周波数間の比率に従って選択する必要があります。この値は1だけ減分され、0値で期待される同期に達します。計算式は、次のとおりです。

$$RELOAD = (f_{TARGET} / f_{SYNC}) - 1$$

RELOADフィールドのリセット値は、48MHzのターゲット周波数と1kHzの同期信号周波数（USBからのSOF信号）に対応します。

FELIM値

FELIM値の選択は、HSI48オシレータ特性とその標準的なトリミングステップサイズに密接に関係しています。最適値は、オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

$$FELIM = (f_{TARGET} / f_{SYNC}) * STEP[\%] / 100\% / 2$$

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられる必要があります。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM値を少し上げることでヒステリシスを増やすことができます。

FELIMフィールドのリセット値は、$(f_{TARGET} / f_{SYNC}) = 48000$と、標準的なトリミングステップサイズ（0.14%）に対応しています。

注： **トリミングステップサイズは、製品によって異なります。正確な設定については、データシートを確認してください。**

注意： **RELOADおよびFELIMフィールドの誤った設定に対するハードウェアの保護はなく、不規則なトリム応答を引き起こす可能性があります。期待される動作モードでは、RELOAD値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM値よりも大きい値です（OUTRANGE制限）。**

12.5 CRS低電力モード

表123.低電力モードがCRSに与える影響

モード	説明
SLEEP	影響はありません。CRS割込みによって、デバイスはSLEEPモードから復帰します。
STOP	CRSレジスタを停止します。CRSは、STOPモードが終了し、HSI48オシレータがリスタートするまで、動作を停止します。
STANDBY	CRSペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

12.6 CRS割込み

表124.割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット	フラッグビットのクリア
期待される同期	ESYNCF	ESYNCIE	ESYNCC
同期OK	SYNCOKF	SYNCOKIE	SYNCOKC
同期警告	SYNCWARNF	SYNCWARNIE	SYNCWARNC
同期またはトリミングエラー（TRIMOVF、SYNCMISS、SYNCERR）	ERRF	ERRIE	ERRC

12.7 CRSレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位のみでアクセスすることができます。

12.7.1 CRS制御レジスタ（CRS_CR）

アドレスオフセット：0x00

リセット値：0x0000 2000

リセット値：0x0000 4000（7bit TRIM幅をサポートする製品）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TRIM[6:0]							SW SYNC	AUTO TRIMEN	CEN	Res.	ESYNC IE	ERR IE	SYNC WARNIE	SYNC OKIE
	rw	rw	rw	rw	rw	rw	rw	rt_w1	rw	rw		rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **TRIM[6:0]**：HSI48オシレータ高精度トリミング

HSI48オシレータ高精度トリミングのデフォルト値は64となり、この値はトリミングインターバルの中間に相当します。

ビット7 **SWSYNC**：ソフトウェアSYNCイベントの生成

このビットは、ソフトウェアSYNCイベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。

0：動作なし。

1：ソフトウェアSYNCイベントが生成されます。

ビット6 **AUTOTRIMEN**：自動トリミングの有効化

このビットは、2つのSYNCイベント間で測定した周波数誤差に応じて、TRIMビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIMビットが読出し専用になります。TRIM値は、測定した周波数誤差値に応じて、ハードウェアで1度に1または2ステップずつ調整できます。詳細については、セクション12.4.4を参照してください。

0：自動トリミングは無効です。ユーザは、TRIMビットを調整できます。

1：自動トリミングは有効です。TRIMビットは読出し専用で、ハードウェアによって制御されます。

ビット5 **CEN**：周波数誤差カウンタの有効化

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。

0：周波数誤差カウンタは無効です。

1：周波数誤差カウンタは有効です。

このビットをセットすると、CRS_CFGRレジスタは書込み保護され、変更できません。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 **ESYNCIE**：期待されるSYNC割込みの有効化

0：期待されるSYNC（ESYNCF）割込みは無効です。

1：期待されるSYNC（ESYNCF）割込みは有効です。

ビット2 **ERRIE**：同期またはトリミングエラー割込みの有効化

0：同期またはトリミングエラー（ERRF）割込みは無効です。

1：同期またはトリミングエラー（ERRF）割込みは有効です。

ビット1 **SYNCWARNIE**：SYNC警告割込みの有効化

0：SYNC警告（SYNCWARNF）割込みは無効です。

1：SYNC警告（SYNCWARNF）割込みは有効です。

ビット0 **SYNCOKIE**：SYNCイベントOK割込みの有効化

0：SYNCイベントOK（SYNCOKF）割込みは無効です。

1：SYNCイベントOK（SYNCOKF）割込みは有効です。

12.7.2 CRS設定レジスタ（CRS_CFGR）

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです（CENビットはCRS_CRでクリアされる）。このカウンタを有効にすると、このレジスタは書込み保護されます。

アドレスオフセット：0x04

リセット値：0x2022 BB7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SYNCPOL	Res.	SYNCSRC[1:0]		Res.	SYNCDIV[2:0]			FELIM[7:0]							
rw		rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **SYNCPOL**SYNC極性の選択

このビットは、SYNC信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。

0：SYNCは立上がりエッジでアクティブです（デフォルト）。

1：SYNCは立下がりエッジでアクティブです。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29:28 **SYNCSRC[1:0]**：SYNC信号ソースの選択

これらのビットは、SYNC信号ソースを選択するために、ソフトウェアによってセット／クリアされます（表121および表122を参照）。

00：:crs_sync_in_1がSYNC信号ソースとして選択されます。

01：crs_sync_in_2をSYNC信号ソースとして選択します。

10：crs_sync_in_3をSYNC信号ソースとして選択します。

11：crs_sync_in_4をSYNC信号ソースとして選択します。

注： USB LPM（Link Power Management、リンク電源管理）の使用時にデバイスがSLEEPモードである場合、周期的USB SOFはホストから生成されません。そのため、実行中にHSI48オシレータを較正するためにSYNC信号がCRSに供給されることはありません。SLEEPモードからのウェイクアップ後に必要なクロック精度を確保するために、GPIOのLSEまたはリファレンスクロックをSYNC信号として使用する必要があります。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **SYNCDIV[2:0]**：SYNC分周器

これらのビットは、SYNC信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。

000：SYNCは分周されません（デフォルト）。

001：SYNCは2分周されます。

010：SYNCは4分周されます。

011：SYNCは8分周されます。

100：SYNCは16分周されます。

101：SYNCは32分周されます。

110：SYNCは64分周されます。

111：SYNCは128分周されます。

ビット23:16 **FELIM[7:0]**：周波数誤差制限

FELIMには、CRS_ISRレジスタのFECAP[15:0]ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP評価の詳細については、セクション12.4.4を参照してください。

ビット15:0 **RELOAD[15:0]**：カウンタ再ロード値

RELOADは、SYNCイベントごとに周波数誤差カウンタにロードされる値です。

カウンタの動作の詳細については、セクション12.4.3を参照してください。

12.7.3 CRS割込みおよびステータスレジスタ（CRS_ISR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FECAP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEDIR	Res.	Res.	Res.	Res.	TRIM OVF	SYNC MISS	SYNC ERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNC WARNF	SYNC OKF
r					r	r	r					r	r	r	r

ビット31:16 **FECAP[15:0]**：周波数誤差のキャプチャ

FECAPは、最後のSYNCイベント時にラッチされた、周波数誤差カウンタ値です。

FECAPの使用の詳細については、セクション12.4.4を参照してください。

ビット15 **FEDIR**：周波数誤差の方向

FEDIRは、最後のSYNCイベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。

0：アップカウント方向。実際の周波数はターゲットを上回ります。

1：ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **TRIMOVF**：トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによってTRIM値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CRレジスタのERRIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのERRCビットをセットすることで、ソフトウェアによってクリアされます。

0：トリミングエラーは通知されていません。

1:トリミングエラーが通知されました。

ビット9 **SYNCMISS**：SYNCの欠落

このフラグは、周波数誤差カウンタが値FELIM * 128に達し、SYNCが検出されない場合にハードウェアによってセットされます。これは、SYNCパルスが欠落しているか、周波数誤差が大きすぎて（内部周波数が高すぎる）、TRIM値を調整しても補正できないため、他の操作を行う必要があることを意味します。この時点で、周波数誤差カウンタは停止し（次のSYNCを待つ）、CRS_CRレジスタのERRIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのERRCビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC欠落エラーは通知されていません。

1：SYNC欠落エラーが通知されました。

ビット8 **SYNCERR**：SYNCエラー

このフラグは、SYNCパルスがESYNCイベントより前に発生し、測定した周波数誤差がFELIM * 128以上である場合に、ハードウェアによってセットされます。これは、TRIM値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CRレジスタのERRIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのERRCビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNCエラーは通知されていません。

1：SYNCエラーが通知されました。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **ESYNCF**：期待されるSYNCフラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CRレジスタのESYNCIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのESYNCCビットをセットすることで、ソフトウェアによってクリアされます。

0：期待されるSYNCは通知されていません。

1：期待されるSYNCが通知されました。

ビット2 **ERRF**：エラーフラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、およびSYNCERRビットの論理和です。CRS_CRレジスタのERRIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのERRCビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、およびSYNCERRビットをクリアします。

0：同期またはトリミングエラーは通知されていません。

1：同期またはトリミングエラーが通知されました。

ビット1 **SYNCWARNF**：SYNC警告フラグ

このフラグは、測定した周波数誤差がFELIM * 3以上でFELIM * 128未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM値を複数のステップで調整する必要があります。CRS_CRレジスタのSYNCWARNIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのSYNCWARNCビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC警告は通知されていません。

1：SYNC警告が通知されました。

ビット0 **SYNCOKF**：SYNCイベントOKフラグ

このフラグは、測定した周波数誤差がFELIM * 3未満である場合に、ハードウェアによってセットされます。これは、TRIM値の補正が必要であること、または周波数誤差の補正は1つのトリミングステップで十分であることを意味します。CRS_CRレジスタのSYNCOKIEビットがセットされている場合、割込みが生成されます。CRS_ICRレジスタのSYNCOKCビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNCイベントOKは通知されていません。

1：SYNCイベントOKが通知されました。

12.7.4 CRS割込みフラグクリアレジスタ（CRS_ICR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNC WARNC	SYNC OKC
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **ESYNCC**：期待SYNCクリアフラグ

このビットに1を書き込むと、CRS_ISRレジスタのESYNCFフラグをクリアします。

ビット2 **ERRC**：エラークリアフラグ

このビットに1を書き込むと、TRIMOVF、SYNCMISS、およびSYNCERRビットがクリアされ、結果としてCRS_ISRレジスタのERRFフラグもクリアされます。

ビット1 **SYNCWARNC**：SYNC警告クリアフラグ

このビットに1を書き込むと、CRS_ISRレジスタのSYNCWARNFフラグをクリアします。

ビット0 **SYNCOKC**：SYNCイベントフラグ（OK）のクリア

このビットに1を書き込むと、CRS_ISRレジスタのSYNCOKFフラグをクリアします。

12.7.5 CRSレジスタマップ

表125.CRSレジスタマップとリセット値

オフ	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[6]	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
	リセット値																		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	CRS_CFGR	SYNCPOL	Res.	SYNC SRC [1:0]		Res.	SYNC DIV [2:0]			FELIM[7:0]								RELOAD[15:0]															
	リセット値	0		1	0		0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1
0x08	CRS_ISR	FECAP[15:0]																FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0					0	0	0	0
0x0C	CRS_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

13 汎用I/O（GPIO）

13.1 はじめに

各汎用 I/O ポートにはそれぞれ 4 つの 32bit 設定レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）、2つの32bitデータレジスタ（GPIOx_IDR、GPIOx_ODR）、1 つの 16bit リセットレジスタ（GPIOx_BRR）、および 1 つの 32bit セット／リセットレジスタ（GPIOx_BSRR）があります。

さらに、すべてのGPIOには、それぞれ32bitロックレジスタ（GPIOx_LCKR）、2つの32bitオルタネート機能選択レジスタ（GPIOx_AFRH、GPIOx_AFRL）、1つのセキュア設定レジスタ（GPIOx_SECCFGR）、および1つの低電圧のハイスピードレジスタ（GPIOx_HSLVR）があります。

13.2 GPIOの主な機能

- 出力状態：プッシュプルまたはオープンドレイン + プルアップ／プルダウン
- 出力データレジスタ（GPIOx_ODR）またはペリフェラル（オルタネート機能出力）からの出力データ
- 各I/Oのスピード選択
- 入力状態：フローティング、プルアップ／プルダウン、アナログ
- 入力データレジスタ（GPIOx_IDR）またはペリフェラル（オルタネート機能入力）への入力データ
- GPIOx_ODRへのビット単位の書込みアクセス用のビットセット／リセットレジスタ（GPIOx_BSRR）
- I/Oポート設定を固定するロック機構（GPIOx_LCKR）
- アナログ機能
- オルタネート機能選択レジスタ
- 2クロックサイクルで変化可能な高速トグル
- 柔軟性の高いピンの多重化により、I/OピンをGPIOまたはいくつかのペリフェラル機能の1つとして使用可能
- TrustZoneセキュリティをサポート

13.3 GPIOの機能説明

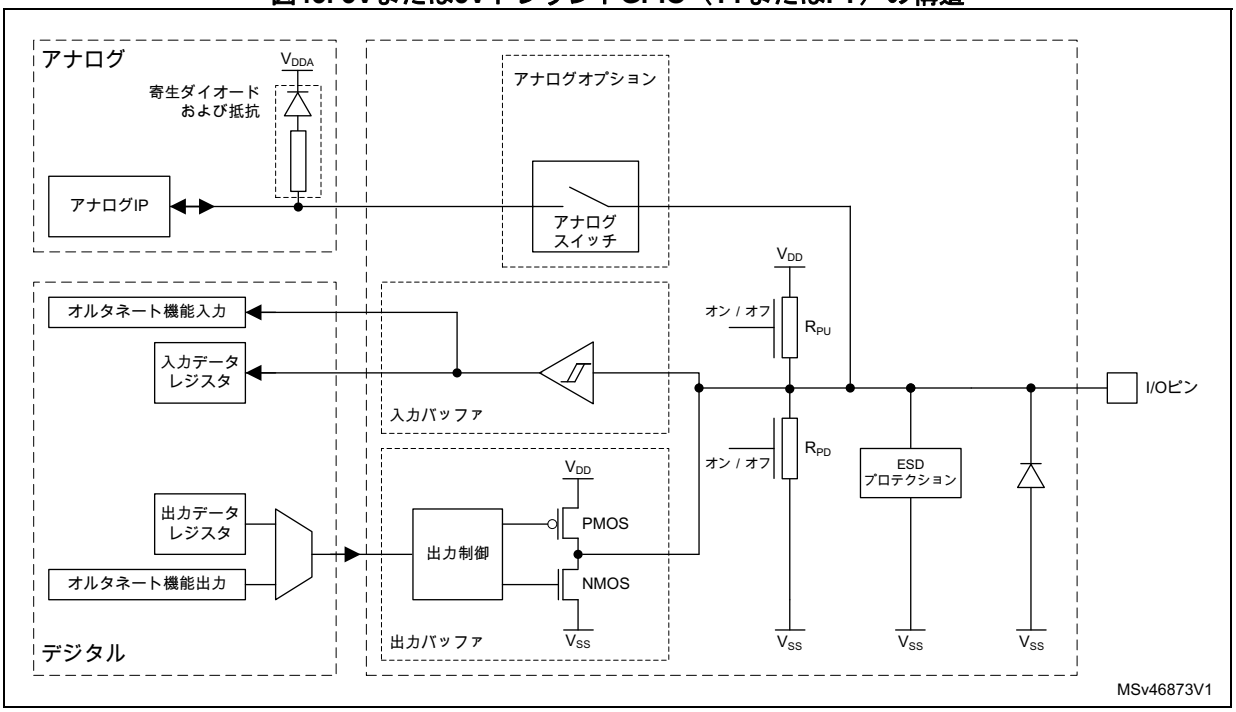
各I/Oポートの特定のハードウェア特性については、データシートに記載されています。汎用I/O（GPIO）ポートの各ポートビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各I/Oポートビットは自由にプログラム可能ですが、I/Oポートレジスタには32bitワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRRおよびGPIOx_BRRレジスタでは、任意のGPIOx_ODRレジスタに不可分な読出し／変更アクセスを行うことができます。これにより、読出しと変更アクセスの間にIRQが発生するリスクを回避できます。

次の図に、3Vまたは5VトレラントGPIO（TTまたはFT）の基本構造を示します。表126には、実行可能なポートビットの設定を示します。

図43. 3Vまたは5VトレラントGPIO（TTまたはFT）の構造



注： **TT GPIOにはアナログスイッチは存在せず、直接接続に置き換えられます。アナログブロックの寄生回路では、5Vは許容されません。**

表126.ポートビット設定(1)

MODE(i)[1:0]	OTYPE(i)	OSPEED(i)[1:0]	PUPD(i)[1:0]		I/O設定	
01	0	SPEED[1:0]	0	0	GP出力	PP
	0		0	1	GP出力	PP + PU
	0		1	0	GP出力	PP + PD
	0		1	1	予約済みです。	
	1		0	0	GP出力	OD
	1		0	1	GP出力	OD + PU
	1		1	0	GP出力	OD + PD
	1		1	1	予約済み（GP出力OD）	

表126.ポートビット設定(1) （続き）

MODE(i)[1:0]	OTYPE(i)	OSPEED(i)[1:0]		PUPD(i)[1:0]		I/O設定	
10	0	SPEED[1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	予約済みです。	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	予約済みです。	
00	x	x	x	0	0	入力	フローティング
	x	x	x	0	1	入力	PU
	x	x	x	1	0	入力	PD
	x	x	x	1	1	予約済み（入力フローティング）	
11	x	x	x	0	0	入力／出力	アナログ
	x	x	x	0	1	予約済みです。	
	x	x	x	1	0		
	x	x	x	1	1		

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

13.3.1 汎用I/O（GPIO）

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどのI/Oポートはアナログモードに設定されています。

リセット後、デバッグピンはオルタネート機能のプルアップ／プルダウンに設定されています。

- PA15：JTDIプルアップ
- PA14：JTCK/SWCLKプルダウン
- PA13：JTMS/SWDIOプルアップ
- PB4：NJTRSTプルアップ
- PB3：JTDO/TRACESWOフローティング状態、プルアップ／プルダウンなし

PH3/BOOT0は、少なくともオプションバイトのローディングが終了するまでリセット中は、入力モードになります（セクション13.3.15を参照）。

ピンが出力として設定されている場合、出力データレジスタ（GPIOx_ODR）に書き込まれた値がI/Oピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード（ローレベルのみが駆動され、ハイレベルはハイインピーダンス）で使用することができます。

入力データレジスタ（GPIOx_IDR）は、AHBクロックサイクルごとに、I/Oピン上のデータをキャプチャします。

すべてのGPIOピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDRレジスタの値によってこれらを有効化／無効化できます。

13.3.2 I/Oピンオルタネート機能マルチプレクサと配置

デバイスのI/Oピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に1つのペリフェラルオルタネート機能（AF）のみが1つのI/Oピンに接続可能となっています。この方法により、同じI/Oピンを共有するペリフェラル間の競合はなくなります。

各I/Oピンは、最大16のオルタネート機能入力（AF0～AF15）を持つマルチプレクサを内蔵しており、これらの入力は、GPIOx_AFRL（ピン0～7）とGPIOx_AFRH（ピン8～15）レジスタを介して設定することができます。

- リセット後、マルチプレクサの選択はオルタネート機能0（AF0）です。I/Oは、GPIOx_MODERレジスタを通してオルタネート機能モードで設定されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。

この柔軟性の高いI/O多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれのI/Oピンに配置されており、さらに小型のパッケージで使用できるペリフェラルの数を最適化します。

I/Oを任意の設定で使用するには、次の手順に従う必要があります。

- **デバッグ機能**：各デバイスのリセット後、これらのピンはデバッガホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **GPIO**：必要とするI/Oを、GPIOx_MODERで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**：
  - I/OをGPIOx_AFRLまたはGPIOx_AFRHのどちらかで必要とするAFxに接続します。
  - タイプ、プルアップ／プルダウン、出力スピードをそれぞれGPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDRで選択します。
  - 必要とするI/Oを、GPIOx_MODERでオルタネート機能として設定します。
- **追加機能**：
  - ADC、DAC、OPAMP、およびCOMPについては、GPIOx_MODERのアナログモードで必要とするI/Oを設定し、ADC、DAC、OPAMP、およびCOMPレジスタで必要な機能を設定します。
  - RTC、WKUPx、オシレータなどの追加機能については、関連するRTC、PWR、およびRCCレジスタで必要な機能を設定します。これらの機能は、標準のGPIOレジスタの設定よりも優先されます。

オルタネート機能I/Oピンの配置に関する詳細は、デバイスデータシートの「オルタネート機能配置」表を参照してください。

13.3.3 I/Oポート制御レジスタ

各GPIOポートには32bitメモリマップド制御レジスタが4つ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）あり、最大16個のI/Oを設定します。

- GPIOx_MODERはI/Oモード（入力、出力、AF、アナログ）を選択するために使用されます。
- GPIOx_OTYPERおよびGPIOx_OSPEEDRは、出力タイプ（プッシュプルまたはオープンドレイン）およびスピードを選択するために使用されます。
- I/Oの方向によらず、GPIOx_PUPDRはプルアップ／プルダウンを選択するために使用されます。

13.3.4 I/Oポートデータレジスタ

各GPIOには、16bitメモリマップドデータレジスタが2つあります。入力データレジスタ（GPIOx_IDR、x = A ~ J）と出力データレジスタ（GPIOx_ODR、x = A ~ J）です。

GPIOx_ODRは出力されるデータを格納し、読出し/書込みアクセスが可能です。I/Oから入力されるデータは読出し専用レジスタ（GPIOx_IDR）に格納されます。

13.3.5 I/Oデータのビット単位の操作

ビットセット／リセットレジスタ（GPIOx_BSRR）は32bitレジスタで、アプリケーションによる出力データレジスタ（GPIOx_ODR）のビット単位のセット／リセットを可能にします。ビットセットリセットレジスタはGPIOx_ODRの2倍のサイズです。

GPIOx_ODRの各ビットにはGPIOx_BSRRの2つの制御ビットBS(i)とBR(i)が対応します。BS(i)に1を書き込むと、対応するODR(i)ビットがセットされます。BR(i)に1を書き込むと、対応するODR(i)ビットがリセットされます。

GPIOx_BSRRのいずれかのビットに0を書き込んでも、GPIOx_ODRの対応するビットには影響しません。仮に、GPIOx_BSRRのビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRRレジスタを使用したGPIOx_ODR内の個々のビットの変更には、1回限りの効果しかなく、GPIOx_ODRビットを固定するものではありません。GPIOx_ODRのビットは常に直接アクセスすることができます。GPIOx_BSRRレジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODRをビットレベルでプログラムする場合は、ソフトウェアで割込みを無効にする必要はありません。1回の不可分なAHB書込みアクセスで1bitまたは複数ビットを変更できます。

13.3.6 GPIOロック機構

GPIOx_LCKRレジスタへの特定の書込みシーケンスを行うことにより、GPIO制御レジスタを停止できます。停止されるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH、およびGPIOx_HSLVRです。

GPIOx_LCKRに書き込むには、特定の書込み／読出しシーケンスを行う必要があります。このレジスタのビット16に適切なLOCKシーケンスを行う場合、LCKR[15:0]の値を使用してI/Oの設定を固定します（この書込みシーケンス中、LCKR[15:0]の値は同じである必要があります）。あるポートビットにロックシーケンスが行われると、次のMCUリセットまたはペリフェラルリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH）の対応するビットが停止されます。

GPIOx_LCKRビット16を[15:0]ビットと同時に設定する必要があるため、このロックシーケンスは、GPIOx_LCKRへのワード（32bit長）アクセスを使用してのみ実行できます。

13.3.7 I/Oオルタネート機能の入力／出力

各I/Oが使用できるオルタネート機能入力／出力の1つを選択するため、2つのレジスタが用意されています。これらのレジスタを使用し、必要に応じて、アプリケーションでオルタネート機能を他のピンに接続することができます。

つまり、GPIOx_AFRLおよびGPIOx_AFRHオルタネート機能レジスタを使用していくつかの使用可能なペリフェラル機能が、各GPIOに多重化されることになります。こうして各I/Oに使用可能な機能のどれか1つをアプリケーションによって選択できます。AF選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、任意のI/Oが持つオルタネート機能入力／出力に対し1つのチャネルが選択されます。

どの機能が各GPIOピンに多重化されているかについてはデバイスデータシートを参照してください。

13.3.8 外部割込み／ウェイクアップライン

すべてのポートに外部割込み機能があります。外部割込みラインを使用するには、ポートを入力、出力、またはオルタネート機能モードに設定します（アナログモードには設定しないでください）。セクション 23：拡張割込み／イベントコントローラ（EXTI）を参照してください。

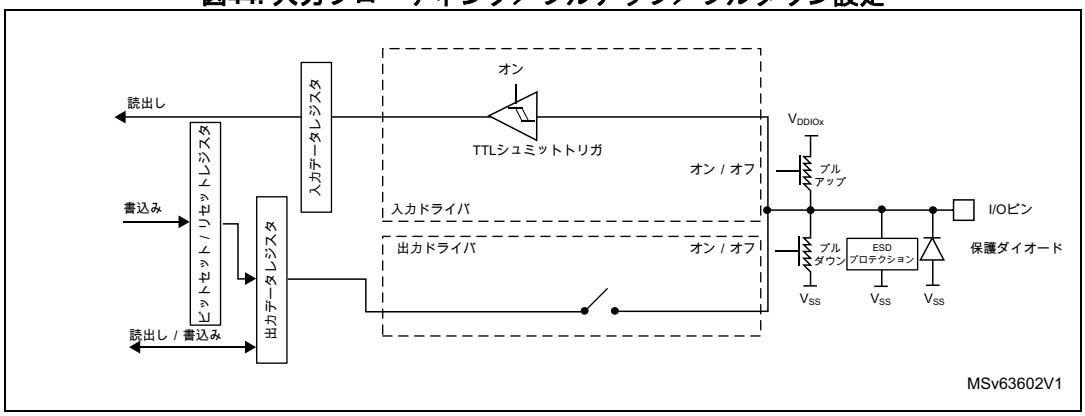
13.3.9 入力設定

I/Oポートが入力としてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDRの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/Oピン上のデータは、AHBクロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O状態が得られます。

次の図は、I/Oポートビットの入力設定を示しています。

図44. 入力フローティング／プルアップ／プルダウン設定



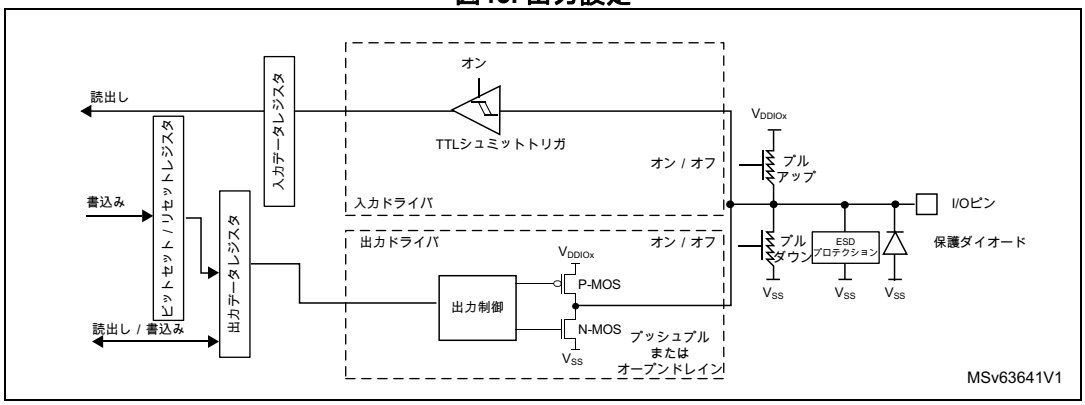
13.3.10 出力設定

I/Oポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
  - オープンドレインモード : 出力レジスタが0のときにはN-MOSが有効になり、1のときにはポートはハイインピーダンス状態のままです（P-MOSが有効になることはありません）。
  - プッシュプルモード : 出力レジスタが0のときN-MOSが有効になり、1のときにはPMOSが有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDRレジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/Oピン上のデータは、AHBクロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O状態が得られます。
- 出力データレジスタの読出しアクセスによって、最後に書き込まれたデータが得られます。

次の図は、I/Oポートビットの出力設定を示しています。

図45. 出力設定



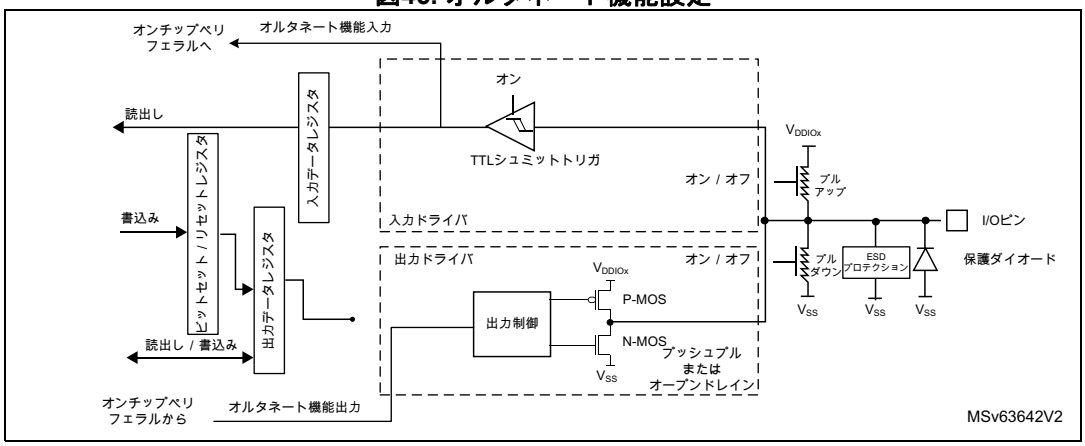
13.3.11 オルタネート機能設定

I/Oポートがオルタネート機能としてプログラムされた場合 :

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル（トランスミッタイネーブルおよびデータ）からの信号で駆動されます。
- シュミットトリガ入力が有効になります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/Oピン上のデータは、AHBクロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O状態が得られます。

次の図は、I/Oポートビットのオルタネート機能設定を示しています。

図46. オルタネート機能設定



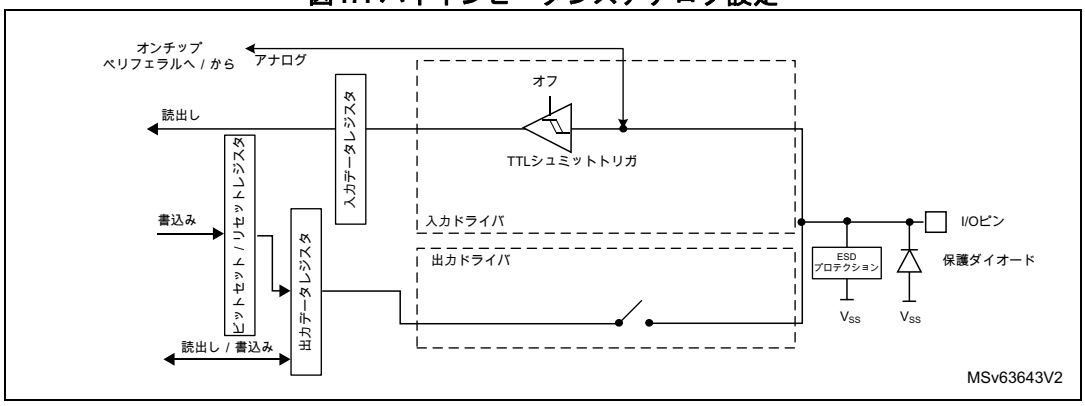
13.3.12 アナログ設定

I/Oポートがアナログとしてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップ抵抗およびプルダウン抵抗はハードウェアによって無効にされます。
- 入力データレジスタの読出しアクセスを行うと、値0が得られます。

次の図は、I/Oポートビットのハイインピーダンスアナログ入力設定を示しています。

図47. ハイインピーダンスアナログ設定



13.3.13 HSEまたはLSEオシレータのピンをGPIOとして使用

HSEまたはLSEオシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータピンを通常のGPIOとして使用することができます。

HSEまたはLSEオシレータがスイッチオンされた場合（RCC_CSRレジスタのHSEONまたはLSEONビットを設定することで）、オシレータは関連ピンを制御しますが、これらのピンのGPIO設定は無効です。

オシレータがユーザ外部クロックモードに設定されている場合、ピンのみがクロック入力のために確保されますが、OSC_OUTまたはOSC32_OUTピンは通常のGPIOとして使用することができます。

13.3.14 GPIOピンをRTC供給ドメインで使用

コア供給ドメインの電源がオフになったとき（デバイスがSTANDBYモードに移行したとき）、PC13/PC14/PC15のGPIOが機能しなくなります。この場合、それらのGPIO設定がRTC設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

RTCによるI/O制御の詳細は、セクション 63.3 : RTCの機能説明を参照してください。

13.3.15 PH3をGPIOとして使用

PH3はブートピン（BOOT0）またはGPIOとして使用できます。PH3は、ユーザオプションバイトのnSWBOOT0ビットに応じて、入力モードからアナログ入力モードに切り替わります。

- nSWBOOT0 = 1の場合、オプションバイトローディング後
- nSWBOOT0 = 0の場合、リセット後

13.3.16 PA11およびPA12をGPIOとして使用（STM32U59x/5Ax/5Fx/5Gxのみ）

PA11およびPA12はOTG_HSの追加機能を備えています。PA11およびPA12を標準GPIOまたはオルタネート機能として使用するには、制約があります。セクション 10.7.12 : 低電力モードでのUSBパワー・マネージメント （STM32U59x/5Ax/5Fx/5Gxのみ）を参照してください。

13.3.17 OPAMPx_VINM専用ピン

OPAMPx_VINM専用ピンは3Vトレラントで、V_{DDA}により電力供給されます。これらのピンは図43に示すような完全なTT構造を備えていませんが、OPAMPxに直接接続できます。OPAMPx_VINM専用ピンは特定のパッケージでのみ使用できます（これらのピンが使用できるかどうかについては、デバイスのデータシートを参照してください）。

13.3.18 TrustZoneセキュリティ

TrustZoneセキュリティは、FLASH_OPTRのTZENオプションビットによって有効化されます。TrustZoneが有効な場合（TZEN = 1）、GPIOポートの各I/OピンをGPIOx_SECCFGRレジスタによりそれぞれセキュアとして設定することができます。

選択したI/Oピンがセキュアとして設定されている場合、その対応するオルタネート機能、モード選択、I/Oデータの設定ビットは、非セキュアアクセスに対してセキュアです。非セキュアアクセスの場合、これらのビットはRAZ/WIです。RCCのGPIOクロックおよびリセット制御ビットは、GPIOの少なくとも1つのI/Oがセキュアになるとすぐに、自動的にセキュアとして設定されます。

ペリフェラル機能を持つI/Oは、ペリフェラル･セキュリティ設定によっても調整されます（詳細については、セクション 5：グローバルTrustZoneコントローラ（GTZC）を参照）。

- I/O ピン選択がオルタネート機能レジスタで行われるペリフェラル：ペリフェラルがセキュアに設定されている場合、非セキュアI/Oピンへの接続はできません。これが順守されない場合、セキュアなペリフェラルへの入力データは強制的に0になり（I/O入力ピンの値は無視される）、出力ピンの値は強制的に0になります。これにより、非セキュアI/Oからのセキュアな情報の漏洩を回避することができます。
- アナログスイッチを備え、ペリフェラル（ADCなど）により直接制御されるI/O：I/Oがセキュアな場合、I/Oアナログスイッチは非セキュアペリフェラルによって制御できません。これが順守されない場合、スイッチはオープンのままになります。これにより、アナログ･パスを経由した非セキュアなペリフェラル、またはI/Oへのセキュア･データのリダイレクトを防ぎます。詳細については、セクション 3：システムセキュリティを参照してください。
- I/O の「追加機能」とペリフェラルの間のパスの一部は、I/O がセキュアでペリフェラルが非セキュアである場合はブロックされません。そのため、アプリケーションで使用しない場合でも、これらのペリフェラルをセキュアに設定することが推奨されます。当該ペリフェラルの一覧については、セクション 3：システムセキュリティを参照してください。パスにセキュリティ制御がかかっている場合は、オルタネート機能によるI/O選択と同じ規則に従います。

ペリフェラルオルタネート機能や追加機能の配置の詳細については、データシートのデバイス･ピンの定義の表を参照してください。

リセット後、すべてのGPIOポートはセキュアです。

表127は、GPIO_SECCFGRレジスタのセキュリティ設定ビットに対するI/Oポートのセキュアビットの一覧です。I/Oビットポートがセキュアとして設定される場合

- セキュアビット：読出しおよび書込み操作は、セキュアアクセスによってのみ可能です。セキュアビットへの非セキュア読出し／書込みアクセスはRAZ/WIです。不正アクセスイベントは生成されません。
- 非セキュアビット：制約なしです。読出しおよび書込み操作は、セキュアおよび非セキュアアクセスのどちらでも可能です。

TrustZoneセキュリティが無効な場合（Flash_OPTRレジスタでTZEN = 0）、すべてのレジスタビットは非セキュアです。GPIOx_SECCFGRレジスタはRAZ/WIです。

表127.GPIOセキュアビット

セキュア設定ビット	セキュアビット	レジスタ名	セキュアビットへの非セキュアアクセス
GPIOx_SECCFGRにおいてSECy = 1(1)	MODEy[1:0]	GPIOx_MODER	RAZ/WI
	OTy	GPIOx_OTYPER	
	OSPEEDy[1:0]	GPIOx_OSPEEDR	
	PUPDy[1:0]	GPIOx_PUPDR	
	IDy	GPIOx_IDR	
	ODy	GPIOx_ODR	
	BSyおよびBRy	GPIOx_BSRR	
	LCKy	GPIOx_LCKR	
	BRy	GPIOx_BRR	
	AFSELy[3:0]	GPIOx_AFRH	
		GPIOx_AFRL	
	HSLVy	GPIOx_HSLVR	

1. GPIOxポートの数は、STM32U5シリーズデバイスごとに異なります。特定のポートが使用できるかどうかについては、製品のデータシートを参照してください。これらのポートがない場合は、関連のビットを予約済みと考え、リセット値のままにしてください。

13.3.19 特権および非特権モード

すべてのGPIOレジスタは、セキュリティ状態（セキュアまたは非セキュア）に関係なく、特権および非特権アクセスで読み書きできます。

13.3.20 ハイスピード低電圧モード（HSLV）

一部のI/Oは、HSLVモードに設定することで、低電圧で最大速度を上げる機能があります。I/O HSLVビットは、I/Oの出力スピードが3.3V（デフォルト設定）または1.8V（HSLV = 1）で動作するように最適化されるかどうかを制御します。

注意： **I/OのHSLV設定ビットは、I/O供給電圧（V_{DD}またはV_{DDIO2}）が2.7Vを超える場合はセットできません。電圧が2.7Vよりも高い場合にセットすると、デバイスを傷つける恐れがあります。I/OのHSLVビットは、対応するオプションビットが有効な場合にのみセットできます（I/O供給電圧に応じてIO_VDD_HSLVまたはIO_VDDIO2_HSLV、セクション 7.4 : Flashオプションバイトを参照）。この機能に関連付けられたハードウェア保護はありません。固定されたI/O供給電圧向けのスタティック・コンフィギュレーションとしてのみ使用することを推奨します。**

注意： **STM32U59x/5Ax/5Fx/5Gxデバイスで、GPIOx_HLSVRのHSLVyビットは、PF4/PF5、PI3/PI4、またはPI6/PI7のペアの値と同じ値でプログラムする必要があります。これらの各ペアはOCTOSPIまたはHSPIの差動クロックとして使用できますが、この注意はこれらのペアが別の目的で使用される場合でも適用されます。**

13.3.21 I/O補正セル

I/O転流スルーレート（t_{fall}/t_{rise}）は、電源のI/Oノイズを低減するために、プロセス、電圧、および温度の条件に応じて、ソフトウェアで対応可能です。詳細については、セクション 15 : システム設定コントローラ（SYSCFG）を参照してください。

13.4 GPIOレジスタ

このセクションには、GPIOレジスタの詳細な説明が記載されています。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

GPIOxポートの数は、STM32U5シリーズデバイスごとに異なります。特定のポートが使用できるかどうかについては、製品のデータシートを参照してください。これらのデバイスがない場合は、関連のビットを予約済みと考え、リセット値のままにしてください。

13.4.1 GPIOポートモードレジスタ（GPIOx_MODER）（x = A ~ J）

アドレスオフセット：0x00

リセット値：0xABFF FFFF（ポートAの場合）

リセット値：0xFFFF FEBF（ポートBの場合）

リセット値：0xFFFF FFFF（ポートC ~ Eの場合）

リセット値：0xFFFF FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートFの場合）

リセット値：0xFFFF FFF0（STM32U535/545のポートGの場合）

リセット値：0xFFFF FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートGの場合）

リセット値：0x0000 00CF（STM32U535/545のポートHの場合）

リセット値：0xFFFF FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートHの場合）

リセット値：0x0000 FFFF（STM32U575/585のポートIの場合）

リセット値：0xFFFF FFFF（STM32U59x/5Ax/5Fx/5GxのポートIの場合）

リセット値：0x00FF FFFF（STM32U59x/5Ax/5Fx/5GxのポートJの場合）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODE15[1:0]		MODE14 [1:0]		MODE13 [1:0]		MODE12 [1:0]		MODE11 [1:0]		MODE10 [1:0]		MODE9 [1:0]		MODE8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODE7 [1:0]		MODE6 [1:0]		MODE5 [1:0]		MODE4 [1:0]		MODE3 [1:0]		MODE2 [1:0]		MODE1 [1:0]		MODE0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MODEy[1:0]**：ポートx設定用I/Oピンy（y = 15 ~ 0）

これらのビットは、I/Oモードを設定するために、ソフトウェアによって書き込まれます。

00：入力モード

01：汎用出力モード

10：オルタネート機能モード

11：アナログモード（リセット状態）

注： このフィールドは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.2 GPIOポート出力タイプレジスタ（GPIOx_OTYPER）（x = A ~ J）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **OTy**：ポートx設定用I/OピンY（y = 15 ~ 0）

これらのビットは、I/O出力タイプを設定するために、ソフトウェアによって書き込まれます。

0：出力プッシュプル（リセット状態）

1：出力オープンドレイン

注： このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.3 GPIOポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ~ J）

アドレスオフセット：0x08

リセット値：0x0C00 0000（ポートAの場合）

リセット値：0x0000 00C0（ポートBの場合）

リセット値：0x0000 0000（その他のポートの場合）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEED15[1:0]		OSPEED14 [1:0]		OSPEED13 [1:0]		OSPEED12 [1:0]		OSPEED11 [1:0]		OSPEED10 [1:0]		OSPEED9 [1:0]		OSPEED8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSPEED7 [1:0]		OSPEED6 [1:0]		OSPEED5 [1:0]		OSPEED4 [1:0]		OSPEED3 [1:0]		OSPEED2 [1:0]		OSPEED1 [1:0]		OSPEED0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **OSPEEDy[1:0]**：ポートx設定用I/OピンY（y = 15 ~ 0）

これらのビットは、I/Oの出力スピードを設定するために、ソフトウェアによって書き込まれます。

00：ロースピード

01：ミディアムスピード

10：ハイスピード

11：超ハイスピード

注： 周波数仕様、およびスピード別の電源や負荷条件については、デバイスデータシートを参照してください。

このフィールドは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.4 GPIOポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A ~ J）

アドレスオフセット：0x0C

リセット値：0x6400 0000（ポートAの場合）

リセット値：0x0000 0100（ポートBの場合）

リセット値：0x0000 0000（その他のポートの場合）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPD15[1:0]		PUPD14 [1:0]		PUPD13 [1:0]		PUPD12 [1:0]		PUPD11 [1:0]		PUPD10 [1:0]		PUPD9 [1:0]		PUPD8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPD7 [1:0]		PUPD6 [1:0]		PUPD5 [1:0]		PUPD4 [1:0]		PUPD3 [1:0]		PUPD2 [1:0]		PUPD1 [1:0]		PUPD0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PUPDy[1:0]**：ポートx設定用I/Oピンy（y = 15 ~ 0）

これらのビットは、I/Oのプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

00：プルアップ／プルダウンなし

01：プルアップ

10：プルダウン

11：予約済み

注： このフィールドは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.5 GPIOポート入力データレジスタ（GPIOx_IDR）（x = A ~ J）

アドレスオフセット：0x10

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 IDy：ポートx入力データ用I/Oピンy（y = 15~0）

これらのビットは読出し専用です。これらのビットには、対応するI/Oポートの入力値が格納されています。

注： このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.6 GPIOポート出力データレジスタ（GPIOx_ODR）（x = A~J）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **ODy**：ポート出力データI/OピンY（y = 15~0）

これらのビットは、ソフトウェアによって読出し／書込みができます。

注： 不可分なビット単位のセット／リセットのために、これらのビットを、GPIOx_BSRR または GPIOx_BRR（x = A~J）への書込みによって、個々にセット／リセットすることができます。

このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.7 GPIOポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A~J）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 **BRy**：ポートxリセット用I/OピンY（y = 15~0）

これらのビットは書込み専用です。これらのビットを読み出すと、値0x0000が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをリセットします。

注： BSyビットとBRyビットの両方がセットされた場合、BSyビットが優先されます。

このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

ビット15:0 **BSy**：ポートxセット用I/OピンY（y = 15 ~ 0）

これらのビットは書込み専用です。これらのビットを読み出すと、値0x0000が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをセットします。

注： このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.8 GPIOポート設定ロックレジスタ（GPIOx_LCKR）（x = A ~ J）

アドレスオフセット：0x1C

リセット値：0x0000 0000

このレジスタは、ビット16（LCKK）に正しい書込みシーケンスが行われたときに、ポートビットの設定をロックするために使用されます。ビット[15:0]の値は、GPIOの設定をロックするために使用され、書込みシーケンスの間はLCKR[15:0]の値は同じでなければなりません。GPIOx_LCKRへの書込みには特定の書込みシーケンスが使われます。あるポートビットにLOCKシーケンスが適用されると、次のMCUリセットまたはペリフェラルリセットまで、このポートビットの値を変更できなくなります。

注： **GPIOx_LCKRへの書込みには特定の書込みシーケンスが使われます。このロックシーケンスではワードアクセス（32bit長）のみ可能です。**

各ロックビットによって、特定の設定レジスタ（制御レジスタおよびオルタネート機能レジスタ）が固定されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **LCKK**：ロックキー

このビットは常時読出しができます。ロックキー書込みシーケンスを使用しなければ変更できません。

0：ポート設定ロックキーは無効です。

1：ポート設定ロックキーは有効です。GPIOx_LCKRレジスタは、次のMCUリセットまたはペリフェラルリセットまでロックされます。

- LOCKキー書込みシーケンス：

書込みLCKR[16] = 1 + LCKR[15:0]

書込みLCKR[16] = 0 + LCKR[15:0]

書込みLCKR[16] = 1 + LCKR[15:0]

- LOCKキー読出し

読出しLCKR[16] = 1（この読出し操作は任意だが、ロックが有効であることを確認できる）

注： **ロックキー書込みシーケンスの間は、LCK[15:0]の値を変更することはできません。**

ロックシーケンス中にエラーが発生すると、LOCKは中止されます。

ポートの任意のビットの最初のロックシーケンスの後、次のMCUリセットまたはペリフェラルリセットまでは、LCKKビットのいかなる読出しアクセスに対しても、1が返されます。

ビット15:0 **LCKy**：ポートxロック用I/OピンY（y = 15~0）

これらのビットは読出し／書込みができますが、書き込めるのはLCKKビットが0のときだけです。

0：ポート設定はロックされません。

1：ポート設定はロックされます。

注： **このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。**

13.4.9 GPIOオルタネート機能下位レジスタ（GPIOx_AFRL）（x = A~J）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL7 [3:0]				AFSEL6[3:0]				AFSEL5 [3:0]				AFSEL4 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL3 [3:0]				AFSEL2 [3:0]				AFSEL1 [3:0]				AFSEL0 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 AFSEL**y[3:0]**：ポートxI/Oピンy（y = 7~0）に対するオルタネート機能の選択

これらのビットは、オルタネート機能I/Oを設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

注： このフィールドは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.10 GPIOオルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ~ J）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL15 [3:0]				AFSEL14 [3:0]				AFSEL13 [3:0]				AFSEL12 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL11 [3:0]				AFSEL10 [3:0]				AFSEL9 [3:0]				AFSEL8 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 AFSELy**[3:0]**：ポートxI/Oピンy（y = 15～8）に対するオルタネート機能の選択

これらのビットは、オルタネート機能I/Oを設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

注： このフィールドは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.11 GPIOポートビットリセットレジスタ（GPIOx_BRR）（x = A ~ J）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BRy**：ポートxリセット用I/OピンY（y = 15~0）

これらのビットは書込み専用です。これらのビットを読み出すと、値0x0000が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをリセットします。

注： このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.12 GPIOハイスピード低電圧レジスタ（GPIOx_HSLVR）（x = A~J）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSLV 15	HSLV 14	HSLV 13	HSLV 12	HSLV 11	HSLV 10	HSLV9	HSLV8	HSLV7	HSLV6	HSLV5	HSLV4	HSLV3	HSLV2	HSLV1	HSLV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HSLVy**：ポートxハイスピード低電圧設定（y = 15~0）

これらのビットは、I/O供給電圧が低い場合に、I/Oスピードを最適化するために、ソフトウェアによって書き込まれます。

各ビットは、対応するIO_VDD_HSLV/IO_VDDIO2_HSLVユーザオプションビットがセットされたときのみ有効です。これは、I/O供給電圧が2.7V未満の場合にのみ使用する必要があります。

I/O供給電圧（VDDまたはVDDIO2）が2.7Vよりも高い場合にこれらのビットをセットすると、デバイスにダメージを与えます。

0：I/Oスピードの最適化は無効です。

1：I/Oスピードの最適化は有効です。

注： すべてのI/OがHSLVモードをサポートしているわけではありません。この機能をサポートするI/Oのリストについては、対応するデータシートのI/O構造を参照してください。他のI/OのHSLV設定は、リセット値に保持する必要があります。

このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.13 GPIOセキュア設定レジスタ（GPIOx_SECCFGR）（x = A~J）

アドレスオフセット：0x30

リセット値：0x0000 FFFF（ポートA~Eの場合）

リセット値：0x0000 FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートFの場合）

リセット値：0x0000 FFFC（STM32U535/545のポートGの場合）

リセット値：0x0000 FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートGの場合）

リセット値：0x0000 000B（STM32U535/545のポートHの場合）

リセット値：0x0000 FFFF（STM32U575/585/59x/5Ax/5Fx/5GxのポートHの場合）

リセット値：0x0000 00FF（STM32U575/585のポートIの場合）

リセット値：0x0000 FFFF（STM32U59x/5Ax/5Fx/5GxのポートIの場合）

リセット値：0x0000 0FFF（STM32U59x/5Ax/5Fx/5GxのポートJの場合）

システムがセキュアな場合（TZEN = 1）、このレジスタは書込みアクセスのセキュリティを提供し、セキュアアクセスによってのみ書込みできます。選択したI/Oピンをセキュアとして設定するために使用します。このレジスタへの非セキュア書込みアクセスは破棄されます。

システムが非セキュアな場合（TZEN = 0）、このレジスタはRAZ/WIです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **SECy**：ポートxセキュアビット有効化用I/OピンY（y = 15 ~ 0）

これらのビットは、I/Oポートピンのセキュリティを有効／無効にするために、ソフトウェアによって書き込まれます。

0：I/Oピンは非セキュアです。

1：I/Oピンはセキュアです。対応するすべてのセキュアビットについては、表127を参照してください。

注： このビットは予約済みであり、選択したパッケージに対応するI/Oがない場合にはリセット値に保持する必要があります。

13.4.14 GPIOレジスタマップ

表128.GPIOレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	GPIOx_MODER（x = A～J）	MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]		MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
	ポートAのリセット値	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	ポートBのリセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
	ポートC～Iのリセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	ポートJのリセット値	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	GPIOx_OTYPER（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOx_OSPEEDR（x = A～J）	OSPEED15[1:0]		OSPEED14[1:0]		OSPEED13[1:0]		OSPEED12[1:0]		OSPEED11[1:0]		OSPEED10[1:0]		OSPEED9[1:0]		OSPEED8[1:0]		OSPEED7[1:0]		OSPEED6[1:0]		OSPEED5[1:0]		OSPEED4[1:0]		OSPEED3[1:0]		OSPEED2[1:0]		OSPEED1[1:0]		OSPEED0[1:0]	
	ポートAのリセット値	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	ポートBのリセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
	ポートC ~ Jのリセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOx_PUPDR（x = A～J）	PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]		PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
	ポートAのリセット値	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	ポートBのリセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	ポートC ~ Jのリセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPIOx_IDR（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x14	GPIOx_ODR（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	GPIOx_BSRR（x = A～J）	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	GPIOx_LCKR（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK	LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	GPIOx_AFRL（x = A～J）	AFSEL7[3:0]				AFSEL6[3:0]				AFSEL5 [3:0]				AFSEL4 [3:0]				AFSEL3 [3:0]				AFSEL2 [3:0]				AFSEL1 [3:0]				AFSEL0 [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	GPIOx_AFRH（x = A～J）	AFSEL15[3:0]				AFSEL14[3:0]				AFSEL13[3:0]				AFSEL12[3:0]				AFSEL11[3:0]				AFSEL10[3:0]				AFSEL9[3:0]				AFSEL8 [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	GPIOx_BRR（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	GPIOx_HSLVR（x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSLV15	HSLV14	HSLV13	HSLV12	HSLV11	HSLV10	HSLV9	HSLV8	HSLV7	HSLV6	HSLV5	HSLV4	HSLV3	HSLV2	HSLV1	HSLV0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表128.GPIOレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	GPIOx_ SECCFGR （x = A～J）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	A ~ Iのリセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	ポートJのリセット値																	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、セクション2.3を参照してください。

14 低消費電力汎用I/O（LPGPIO）

14.1 はじめに

低消費電力汎用入出力（LPGPIO）では、メモリ間転送モードでDMAを使用して、STOPモード（STOP0～2モード）でのI/O制御ができます。LPGPIOはGPIOと一緒に使用できるよう設計されています。

14.2 LPGPIOの主な機能

- STOP2モードまで動作する16個のI/O制御
- セキュアクロックおよびリセット管理
- 出力データレジスタ（LPGPIO_ODR）からの出力データ
- 入力データレジスタ（LPGPIO_IDR）への入力データ
- LPGPIO_ODRへのビット単位の書込みアクセス用のビットセット／リセットレジスタ（LPGPIO_BSRR）
- TrustZoneセキュリティをサポート

14.3 LPGPIOの機能詳細

14.3.1 LPGPIOおよびGPIOの設定

LPGPIOでは、LPGPIO_Py（y = 0 ~ 15）ピンのオルタネート機能を使用して、最大16個のI/Oの入力または出力を制御できます。この制御は、DMA転送により、STOP2モードでも機能します。LPGPIO機能を備えたI/Oは、GPIOx_MODERでLPGPIOオルタネート機能として設定する必要があります。そして、I/O値はLPGPIO入力データレジスタで読み出すことができ、I/OはLPGPIO出力データレジスタまたはLPGPIO_BSRR/LPGPIO_BRRレジスタにより駆動できます。

14.3.2 LPGPIO制御レジスタ

LPGPIOによって制御される16の各I/Oは、LPGPIO_MODERにより、入力または出力として設定できます。

14.3.3 LPGPIO I/Oデータレジスタ

LPGPIOには、次の16bitデータレジスタが含まれます。

- 出力されるデータを格納するためのLPGPIO_ODR（読出し／書込みアクセス可能）
- I/Oを通じて入力されるデータを格納するためのLPGPIO_IDR（読出し専用）

14.3.4 LPGPIO I/Oデータのビット単位の操作

32bit LPGPIO_BSRRは、LPGPIO_ODRでのビット単位のセット／リセットを可能にするために実装されます。

LPGPIO_ODRビットはそれぞれ、次のようなLPGPIO_BSRRの制御ビットを備えています。

- LPGPIO_BSRR(i) :「1」を書き込むと、このビットによりLPGPIO_ODR(i)ビットがセットされます。
- LPGPIO_BSRR(i + 16) :「1」を書き込むと、このビットによりLPGPIO_ODR(i)ビットがリセットされます。

注： **これらのビットに「0」を書き込んでも、LPGPIO_ODRの対応するビットには影響しません。**

同じインデックスのビットをセット／リセットしようとした場合、セット動作が優先されます。

LPGPIO_BSRRレジスタに書き込んでもLPGPIO_ODRビットはロックされず、直接アクセスできます。LPGPIO_BSRRによって、ビット単位でアトミック（最小単位実行）操作を行うことができます。

16bitのLPGPIO_BRRでは、個々のビットのリセットができます。LPGPIO_BSRRと同じですが、以下の機能を備えています。

- LPGPIO_BRR(i) :「1」を書き込むと、このビットによりLPGPIO_ODR(i)ビットがリセットされます。

14.3.5 セキュリティ保護

LPGPIOは、I/O設定とデータレジスタへのアクセスを許可／ロックするためのセキュリティメカニズムを備えています。このシステムを使用して、データ破損やデータ観測に対してI/Oを保護することができます。

LPGPIO内のセキュリティメカニズムは、GPIOx_SECCFGRから直接発行されます。そのため、追加の設定は不要です。

LPGPIOのセキュリティには、次のような意味があります。

- 実行されるコードがセキュアな場合、全てのビットにアクセスできます。
- 実行されるコードが非セキュアな場合、非セキュアなI/Oに関するビットのみにアクセスできます。

セキュアなI/Oレジスタビットへの非セキュアなアクセスは、サイレントフェイルです。

- セキュアなI/Oレジスタビットへの非セキュアな書込みは、無視されます（WI）。
- セキュアなI/Oレジスタビットへの非セキュアな読出しでは、0が返されます（RAZ）。
- バスエラーは生成されません。

14.3.6 セキュアクロックおよびリセット管理

RCCのLPGPIOクロックおよびリセット制御ビットは、LPGPIOオルタネート機能を備えたI/Oの少なくとも1つがセキュアになるとすぐに、自動的にセキュアとして設定されます（GPIO_SECCFGRレジスタの対応するI/Oを参照）。

14.4 LPGPIOレジスタ

このセクションには、LPGPIOレジスタの詳細な説明が記載されています。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

14.4.1 LPGPIOポートモードレジスタ（LPGPIO_MODER）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODE 15	MODE 14	MODE 13	MODE 12	MODE 11	MODE 10	MODE 9	MODE 8	MODE 7	MODE 6	MODE 5	MODE 4	MODE 3	MODE 2	MODE 1	MODE 0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **MODEy**：設定用I/Oピンy（y = 15 ~ 0）

これらのビットは、I/Oモードを設定するために、ソフトウェアによって書き込まれます。

0：入力モード

1：出力モード

14.4.2 LPGPIOポート入力データレジスタ（LPGPIO_IDR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **IDy**：入力データ用I/Oピンy（y = 15 ~ 0）

これらのビットは読出し専用です。これらのビットには、対応するI/Oポートの入力値が格納されています。

14.4.3 LPGPIOポート出力データレジスタ（LPGPIO_ODR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **ODy**：出力データ用I/OピンY（y = 15 ~ 0）

これらのビットは、ソフトウェアによって読出し／書込みができます。

注： アトミック（最小単位実行）ビットのセット／リセットでは、これらのODビットを、LPGPIO_BSRRまたはLPGPIO_BRRレジスタへの書込むことにより、個々にセット／リセットすることができます。

14.4.4 LPGPIOポートビットセット／リセットレジスタ（LPGPIO_BSRR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 **BRy**：リセット用I/OピンY（y = 15 ~ 0）

これらのビットは書込み専用です。これらのビットへの読出しでは、0が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをリセットします。

注： BSyとBRyの両方がセットされた場合、BSyが優先されます。

ビット15:0 **BSy**：ポートxセット用I/OピンY（y = 15 ~ 0）

これらのビットは書込み専用です。これらのビットへの読出しでは、0が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをセットします。

14.4.5 LPGPIOポートビットリセットレジスタ（LPGPIO_BRR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BRy**：リセット用I/OピンY（y = 15 ~ 0）

これらのビットは書込み専用です。これらのビットへの読出しでは、0が返されます。

0：対応するODyビットの値は変化しません。

1：対応するODyビットをリセットします。

14.4.6 LPGPIOレジスタマップ

表129.LPGPIOレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPGPIO_MODER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE15	MODE14	MODE13	MODE12	MODE11	MODE10	MODE9	MODE8	MODE7	MODE6	MODE5	MODE4	MODE3	MODE2	MODE1	MODE0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04 - 0x08	予約済み	予約済み																															
0x10	LPGPIO_IDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	LPGPIO_ODR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	LPGPIO_BSRR	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20 - 0x24	予約済み	予約済み																															
0x28	LPGPIO_BRR		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

15 システム設定コントローラ（SYSCFG）

15.1 SYSCFGの主な機能

STM32U5シリーズデバイスは設定レジスタ群を持っています。システム設定コントローラの主な目的は次の通りです。

- 堅牢性の管理機能
- FPU割込みの設定
- I/Oアナログスイッチの一部のI/Oと電圧ブースタのFMP高駆動モードの有効化／無効化
- I/O補正セルの管理
- レジスタセキュリティアクセスの設定
- OTG_HS PHYの設定（STM32U59x/5Ax/5Fx/5Gxの場合のみ）
- HSPI供給容量の調整（STM32U59x/5Ax/5Fx/5Gxの場合のみ）
- DCACHE2による内部SRAMキャッシュ機能の無効化（STM32U59x/5Ax/5Fx/5Gxの場合のみ）

15.2 SYSCFGの機能説明

15.2.1 I/O補正セルの管理

I/O補正セルは、I/Oバッファ向けに8bitの値を生成します（N-MOS用の4bitとP-MOS用の4bit）。これは、PVT動作条件（プロセス、電圧、温度）によって決まります。これらのビットは、I/Oバッファの電流のスルーレートと出力インピーダンスを制御するために使用されます。3つの補正セルが内蔵されています。V_{DD}によって供給されるI/O用のものと、V_{DDIO2}によって供給されるI/O用のものと、HSPIオルタネート機能（AF）を備えたGPIO専用のものです。

デフォルトでは、補正セルは無効であり、すべてのI/Oに固定コードが適用されます。HSIは補正セルによって使用され、SYSCFG_CCCSRで補正セルを有効にする前に有効にする必要があります。

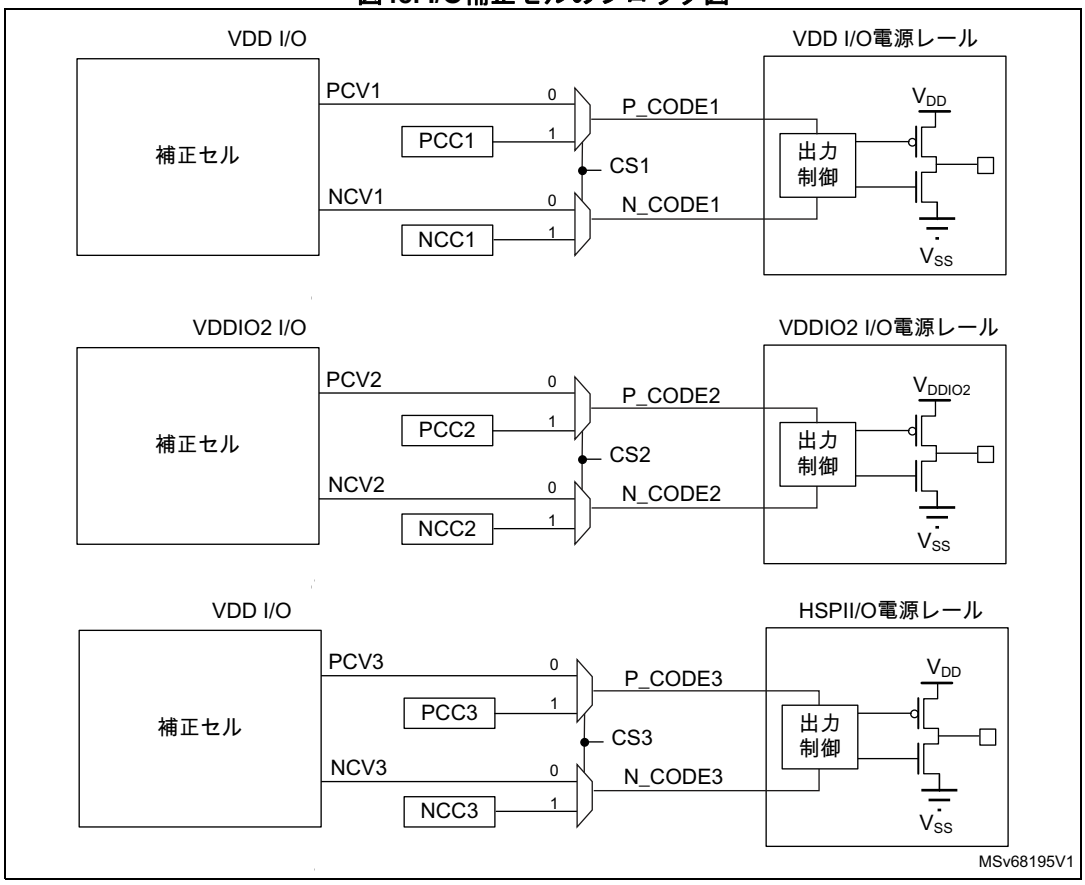
有効にすると、補正セルはPVTを追跡し、RDYxがセットされると、8bitコードPCVxおよびNCVx（HSPI AF機能を除いてV_{DD}によって供給されるI/Oの場合はx = 1、V_{DDIO2}によって供給されるI/Oの場合はx = 2、V_{DD}によって供給されるHSPIAF機能を備えたI/Oの場合はx = 3）がSYSCFG_CCVRで使用可能になります。CSxビットをクリアした場合、I/Oxは、SYSCFG_CCVRから補正セルで生成されたコードを受信します。

3セットのビットを使用できます。V_{DD}電源レールにはPCC1/NCC1およびPCC3/NCC3、V_{DDIO2}電源レールにはPCC2/NCC2が使用できます。これらは、SYSCFG_CCCSRのCS1、CS2、およびCS3ビットを通じて独立して選択できます（図48を参照）。

消費電力を低減するために、コードをSYSCFG_CCVRからSYSCFG_CCCRにコピーすることを推奨します。結果の準備ができたら、CSxビットをセットして、補正セルを無効にします。

注： **補正セルは、$1.6V \le V_{DDIOx} \le 3.6V$の場合のみ、使用できます。**

図48. I/O補正セルのブロック図



15.2.2 SYSCFG TrustZoneのセキュリティと特権

SYSCFG TrustZoneのセキュリティ

TrustZoneセキュリティが有効な場合、SYSCFGはレジスタが非セキュアアクセスによって変更されないように保護することができます。

TrustZoneセキュリティはFlash_OPTRレジスタのTZENオプションビットで有効化されます。

セキュアなレジスタへの非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。SYSCFG不正アクセスイベントがGTZCで有効な場合、不正アクセス割込みが生成されます。

特権／非特権モード

SYSCFGレジスタは、CPU設定用のSYSCFGレジスタ、つまりSYSCFG_CSLCKR、SYSCFG_FPUIMR、SYSCFG_CNSLCKR、およびSYSCFG_SECCFGRのFPUSECビットを除き特権および非特権アクセスにより読み書きできます。

特権レジスタへの非特権アクセスはRAZ/WIです。

以下の表に、レジスタのセキュリティ概要を示します。

表130.TrustZoneセキュリテと特権レジスタアクセス

SYSCFGレジスタ名	読出し／書込みアクセス		特権／非特権アクセス
TrustZone設定(1)	TZEN = 1	TZEN = 0	適用外
SYSCFG_SECCFGR	読出し：制約なし 書込み：セキュアアクセスのみ 非セキュア書込みはWIで、不正アクセスイベントが生成されます。	RAZ/WI	読出し：制限なし、FPUSEC特権書込み専用 その他のビットの書込み：制限なし
SYSCFG_CSLCKR	読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます。	RAZ/WI	特権のみ 非特権：RAZ/WI
SYSCFG_FPUIMR	– FPUSEC = 1の場合： 読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます。 – FPUSEC = 0の場合：読出し／書込み：制限なし	制約なし	特権のみ 非特権：RAZ/WI
SYSCFG_CNSLCKR	読出し／書込み：制約なし	制約なし	特権のみ 非特権：RAZ/WI
SYSCFG_CFGR1	読出し／書込み：GTZCのペリフェラルセキュリティビットに応じて、セキュアビットに対してのみセキュアアクセス 非セキュアビットに対してのみ非セキュアアクセス、それ以外はRAZ/WI	制約なし	制約なし
SYSCFG_CFGR2	– CLASSBSEC = 1の場合： 読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます。 – CLASSBSEC = 0の場合：読出し／書込み：制限なし	制約なし	制約なし
SYSCFG_MESR	– SYSCFGSEC = 1の場合： 読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます。 – SYSCFGSEC = 0の場合：読出し／書込み：制限なし	制約なし	制約なし
SYSCFG_CCCSR SYSCFG_CCVR SYSCFG_CCCR	– SYSCFGSEC = 1の場合： 読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます – SYSCFGSEC = 0の場合：読出し／書込み：制限なし	制約なし	制約なし
SYSCFG_RSSCMDR	レジスタアクセスが許可されていない場合、RAZ/WI(2)	RAZ/WI	制約なし
SYSCFG_OTGHSPHYCR SYSCFG_OTGHSPHYTUNER2	– OTGSEC = 1の場合： 読出し／書込み：セキュアアクセスのみ 非セキュアアクセスはRAZ/WIで、不正アクセスイベントが生成されます。 – OTGSEC = 0の場合：読出し／書込み：制限なし	制約なし	

1. TrustZoneセキュリティはFlash_OPTRレジスタのTZENオプションビットで有効化されます。
2. レジスタアクセスについては、レジスタの説明を参照してください。

15.2.3 OTG_HS PHYの設定（STM32U59x/5Ax/5Fx/5Gxの場合のみ）

OTG_HS PHYを使用するにはに、OTG_HSを設定する前に、次の設定手順を実行する必要があります。

1. SYSCFG、OTG_HS、OTG_HS PHY用のRCCクロックゲーティングレジスタでクロックを有効にします。
2. SYSCFG_OTGHSPHYCRのCLKSELビットフィールドを使用して、OTG_HS PHY用の必要なクロック設定項目を設定します。
3. COMPDISTUNEビットフィールドに0b010を書き込んで切断のしきい値を調整し、SYSCFG_OTGHSPHYTUNER2のSQRXTUNEビットフィールドに0b000を書き込んでスケルチのしきい値を調整します。
4. SYSCFG_OTGHSPHYCRでENをセットして、OTG_HS PHYを有効にします。

15.2.4 HSPI供給容量の調整（STM32U59x/5Ax/5Fx/5Gxの場合のみ）

HSPI供給容量は、SYSCFG_CFGR1のENDCAP[1:0]を使用して調整できます。HSPIオルタネート機能を使用しない場合、ENDCAP[1:0]をリセット値のままにする必要があります。

15.2.5 DCACHE2による内部SRAMキャッシュ機能（STM32U59x/5Ax/5Fx/5Gxの場合のみ）

DCACHE2はGPU2D M0ポートによってのみアドレス指定され、M1ポート上のデータはベクタグラフィックアルゴリズムによって操作できるため、キャッシュ·コヒーレンシの問題を回避するため、GPU2Dを有効にする前に、SYSCFG_CFGR1のSRAMCACHEDをクリアすることを推奨します。また、内部SRAMにはバスマトリックスを通じて0ウェイトステートでアクセスできるため、性能の低下は発生しないと考えられます。

15.3 SYSCFGレジスタ

15.3.1 SYSCFGセキュア設定レジスタ（SYSCFG_SECCFGR）

アドレスオフセット：0x00

リセット値：0x0000 0000

システムがセキュアな場合（TZEN = 1）、このレジスタは書込みアクセスのセキュリティを提供し、アクセスがセキュアな場合のみ書き込むことができます。全体的に書込み保護することができ、このレジスタの各ビットは、個別に書込み保護することもできます。非セキュア書込みアクセスはWIで、不正アクセスイベントが生成されます。読出しに関する制約はありません。

システムが非セキュアな場合（TZEN = 0）、このレジスタはRAZ/WIです。

このレジスタは特権および非特権アクセスにより読み書きできます。ただし、FPUSECは特権アクセスによってのみ書込みができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPUSEC	Res.	CLASSBSEC	SYSCFGSEC
												rw		rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **FPUSEC**：FPUのセキュリティ

0：SYSCFG_FPUIMRはセキュアおよび非セキュアアクセスによって読み書きできます。

1：SYSCFG_FPUIMRはセキュアアクセスによってのみ、読み書きできます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CLASSBSEC**：クラスBのセキュリティ

0：SYSCFG_CFGR2はセキュアおよび非セキュアアクセスによって読み書きできます。

1：SYSCFG_CFGR2はセキュアアクセスによってのみ、読み書きできます。

ビット0 **SYSCFGSEC**：SYSCFGクロック制御、メモリ消去ステータス、および補正セルレジスタのセキュリティ

0：RCCレジスタのSYSCFG_MESR、SYSCFG_CCCSR、SYSCFG_CCVR、およびSYSCFG_CCCRのSYSCFG設定クロックは、セキュアおよび非セキュアアクセスによって読み書きできます。

1：RCCレジスタのSYSCFG_MESR、SYSCFG_CCCSR、SYSCFG_CCVR、およびSYSCFG_CCCRのSYSCFG設定クロックは、セキュアアクセスによってのみ、読み書きできます。

15.3.2 SYSCFG設定レジスタ1（SYSCFG_CFGR1）

アドレスオフセット：0x04

リセット値：0x0000 0000（STM32U535/545/575/585の場合）

リセット値：0x1000 0000（STM32U59x/5Ax/5Fx/5Gxの場合）

システムがセキュアな場合（TZEN = 1）、このレジスタはGTZCペリフェラルのADCセキュリティ設定ビット、およびGPIOセキュリティビットに応じて、セキュアビットと非セキュアビットが入り混じっています。セキュアビットにおける非セキュア読出し／書込みアクセスはRAZ/WIです。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SRAMCACHED	Res.	Res.	ENDCAP[1:0]		Res.	Res.	Res.	Res.	PB9_FMP	PB8_FMP	PB7_FMP	PB6_FMP
			rw			rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	ANASWVDD	BOOSTEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **SRAMCACHED**：DCACHE2による内部SRAMキャッシュ機能の有効化

0：SRAMxはDCACHE2によりキャッシュされません。

1：SRAMxはDCACHE2によりキャッシュされます。

注： このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット27:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **ENDCAP[1:0]**：HSPI供給に対するデカップリング容量の有効化

00：容量オフ

01：1/3の容量

10：2/3の容量

11：フルの容量

注： このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PB9_FMP**：PB9の高速モードプラス駆動機能の有効化

このビットは、PB9がGPIOBでセキュアな場合に、セキュアアクセスによってのみ読み書きできます。このビットにより、PB9がI2Cペリフェラルによって使用されない場合に、PB9のFm+駆動モードが有効になります。たとえば、これを使用してLEDを駆動できます。

0：PB9ピンは標準モードで動作します。

1：Fm+モードはPB9ピンで有効にされ、速度制御はバイパスされます。

ビット18 **PB8_FMP**：PB8の高速モードプラス駆動機能の有効化

このビットは、PB8がGPIOBでセキュアな場合に、セキュアアクセスによってのみ読み書きできます。このビットにより、PB8がI2Cペリフェラルによって使用されない場合に、PB8のFm+駆動モードが有効になります。たとえば、これを使用してLEDを駆動できます。

0：PB8ピンは標準モードで動作します。

1：Fm+モードはPB8ピンで有効にされ、速度制御はバイパスされます。

ビット17 **PB7_FMP**：PB7の高速モードプラス駆動機能の有効化

このビットは、PB7がGPIOBでセキュアな場合に、セキュアアクセスによってのみ読み書きできます。このビットにより、PB7がI2Cペリフェラルによって使用されない場合に、PB7のFm+駆動モードが有効になります。たとえば、これを使用してLEDを駆動できます。

0：PB7ピンは標準モードで動作します。

1：Fm+モードはPB7ピンで有効にされ、速度制御はバイパスされます。

ビット16 **PB6_FMP**：PB6の高速モードプラス駆動機能の有効化

このビットは、PB6がGPIOBでセキュアな場合に、セキュアアクセスによってのみ読み書きできます。このビットにより、PB6がI2Cペリフェラルによって使用されない場合に、PB6のFm+駆動モードが有効になります。たとえば、これを使用してLEDを駆動できます。

0：PB6ピンは標準モードで動作します。

1：Fm+モードはPB6ピンで有効にされ、速度制御はバイパスされます。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **ANASWVDD**：GPIOアナログスイッチ制御電圧の選択

このビットは、GTZCでADC12またはADC4がセキュアな場合に、セキュアアクセスによってのみ読み書きできます。

0：I/OアナログスイッチはV_{DDA}によって、またはブースタがオンのときはブースタによって供給されます。

1：I/OアナログスイッチはV_{DD}によって供給されます。

注： ビット9の設定については、表131を参照してください。

ビット8 **BOOSTEN**：I/Oアナログスイッチ電圧ブースタの有効化

このビットは、GTZCでADC12またはADC4がセキュアな場合に、セキュアアクセスによってのみ読み書きできます。

0：I/OアナログスイッチはV_{DDA}電圧によって供給されます。

1：I/Oアナログスイッチは専用の電圧ブースタ（V_{DD}によって供給）によって供給されます。

注： ビット8の設定については、表131を参照してください。

ビット7:0 予約済みであり、リセット値に保持する必要があります。

下表は、電圧設定に応じたビット8（BOOSTEN）およびビット9（ANASWVDD）の設定条件を示します。

表131.BOOSTENおよびANASWVDDのセット／リセット

V_{DD}	V_{DDA}	BOOSTEN	ANASWVDD
-	> 2.4V	0	0
> 2.4V	< 2.4V		1
< 2.4V		1	0

15.3.3 SYSCFG FPU割込みマスクレジスタ（SYSCFG_FPUIMR）

アドレスオフセット：0x08

リセット値：0x0000 001F

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのFPUSECビットを設定することによって非セキュアアクセスから保護することができます。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権アクセスのみで読み書きできます。非特権アクセスはRAZ/WIです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPU_IE[5:0]					
										rw	rw	rw	rw	rw	rw

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5:0 **FPU_IE[5:0]**：浮動小数点演算装置割込み有効化ビット

FPU_IE[5]：不正確な割込みの有効化（リセット時割込み無効）

FPU_IE[4]：入力異常割込みの有効化

FPU_IE[3]：オーバーフロー割込みの有効化

FPU_IE[2]：アンダーフロー割込みの有効化

FPU_IE[1]：0による除算割込みの有効化

FPU_IE[0]：無効操作割込みの有効化

15.3.4 SYSCFG CPU非セキュアロックレジスタ（SYSCFG_CNSLCKR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このレジスタを使用して、非セキュアレジスタMPUおよびVTOR_NSの設定をロックします。このレジスタは特権アクセスのみで読み書きできます。非特権アクセスはRAZ/WIです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKNSMPU	LOCKNSVTOR
														rs	rs

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **LOCKNSMPU**：非セキュアMPUレジスタのロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。セットされているとき、このビットは非セキュアMPU_CTRL_NS、MPU_RNR_NS、およびMPU_RBAR_NSレジスタへの書込みアクセスを無効にします。

0：非セキュアMPUレジスタへの書込みは有効です。

1：非セキュアMPUレジスタへの書込みは無効です。

ビット0 **LOCKNSVTOR**：VTOR_NSレジスタのロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。

0：VTOR_NSレジスタへの書込みは有効です。

1：VTOR_NSレジスタへの書込みは無効です。

15.3.5 SYSCFG CPUセキュアロックレジスタ（SYSCFG_CSLCKR）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタを使用して、AIRCRレジスタのPRISおよびBFHFNMINSの設定、SAU、セキュアMPU、およびVTOR_Sレジスタの設定をロックします。

システムがセキュアな場合（TZEN = 1）、このレジスタはアクセスがセキュアな場合のみ書き込むことができます。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、このレジスタはRAZ/WIです。このレジスタは、特権アクセスでのみ読み書きできます。非特権アクセスはRAZ/WIです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKSAU	LOCKSMPU	LOCKSVTAIRCR
													rs	rs	rs

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **LOCKSAU**：SAUレジスタのロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。セットされているとき、このビットは、SAU_CTRL、SAU_RNR、SAU_RBAR、およびSAU_RLARレジスタへの書込みアクセスを無効にします。

0：SAUレジスタへの書込みは有効です。

1：SAUレジスタへの書込みは無効です。

ビット1 **LOCKSMPU**：セキュアMPUレジスタのロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。セットされているとき、このビットは、セキュアMPU_CTRL、MPU_RNR、およびMPU_RBARレジスタへの書込みアクセスを無効にします。

0：セキュアMPUレジスタへの書込みは有効です。

1：セキュアMPUレジスタへの書込みは無効です。

ビット0 **LOCKSVTAIRCR**：VTOR_SレジスタおよびAIRCRレジスタのビットロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。セットされているとき、このビットは、VTOR_Sレジスタ、AIRCRレジスタのPRISおよびBFHFNMINSビットへの書込みアクセスを無効にします。

0：VTOR_Sレジスタ、AIRCRレジスタのPRISおよびBFHFNMINSビットへの書込みは有効です。

1：VTOR_Sレジスタ、AIRCRレジスタのPRISおよびBFHFNMINSビットへの書込みは無効です。

15.3.6 SYSCFG設定レジスタ2（SYSCFG_CFGR2）

アドレスオフセット：0x14

リセット値：0x0000 0000

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのCLASSBSECビットを設定することによって非セキュアアクセスから保護することができます。CLASSBSECビットがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCL	PVDL	SPL	CLL
												rs	rs	rs	rs

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **ECCL**：ECCロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、TIM1/8/15/16/17のブレーク入力へのFlash ECCダブルエラー信号接続を有効にし、ロックするために使用できます。

0：ECCダブルエラーは、TIM1/8/15/16/17のブレーク入力に接続されていません。

1：ECCダブルエラーは、TIM1/8/15/16/17のブレーク入力に接続されています。

ビット2 **PVDL**：PVDロック有効化ビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、TIM1/8/15/16/17のブレーク入力へのPVD接続と、PWRレジスタのPVDEとPVDLS[2:0]を有効にし、ロックするために使用できます。

0：PVD割込みは、TIM1/8/15/16/17のブレーク入力に接続されていません。PVDEおよびPVDLS[2:0]ビットは、アプリケーションでプログラムできます。

1：PVD割込みはTIM1/8/15/16/17のブレーク入力に接続されています。PVDEおよびPVDLS[2:0]ビットは、読出し専用です。

ビット1 **SPL**：SRAM ECCのロックビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、TIM1/8/15/16/17のブレーク入力へのSRAM ECCダブルエラー信号接続を有効にし、ロックするために使用できます。

0：SRAMダブルエラーはTIM1/8/15/16/17のブレーク入力に接続されていません。

1：SRAMダブルエラーはTIM1/8/15/16/17のブレーク入力に接続されています。

ビット0 **CLL**：Cortex-M33 LOCKUP（ハードフォルト）出力の有効化

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、TIM1/8/15/16/17のブレーク入力へのCortex-M33 LOCKUP（ハードフォルト）出力の接続を有効にし、ロックするために使用できます。

0：Cortex-M33 LOCKUP出力はTIM1/8/15/16/17のブレーク入力に接続されていません。

1：Cortex-M33 LOCKUP出力はTIM1/8/15/16/17のブレーク入力に接続されています。

15.3.7 SYSCFGメモリ消去ステータスレジスタ（SYSCFG_MESR）

アドレスオフセット：0x18

パワーオンリセット値：0x0000 0000

システムリセット値：0x0000 000X（ビット0はシステムリセットの影響を受けません）

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのSYSCFGSECビットを設定することによって非セキュアアクセスから保護することができます。SYSCFGSECビットがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPMEE
															rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCLR
															rc_w1

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **IPMEE**：ICACHEおよびPKA SRAMの消去ステータス

このビットは、タンパ検出の可能性があった後、ICACHEおよびPKA RAMの消去が完了したときにハードウェアによってセットされます（詳細については、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照）。このビットは、ソフトウェアによって1を書き込むことによってクリアされます。

0：ICACHEおよびPKA SRAMがソフトウェアによってまだクリアされていない場合、消去中です。

1：ICACHEおよびPKA SRAMの消去が完了しました。

ビット15:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **MCLR**：デバイスメモリの消去ステータス

このビットは、パワーオンリセットまたはタンパ検出の後に、SRAM2、BKPSRAM、ICACHE、DCACHE1/2、PKA SRAMの消去が完了したときにハードウェアによってセットされます（詳細については、セクション 64：タンパおよびバックアップレジスタ（TAMP）を参照）。このビットは、システムリセットによってリセットされず、ソフトウェアによって1を書き込むことでクリアされます。

0：ソフトウェアによってまだクリアされていない場合、メモリ消去中

1：メモリの消去が完了しました。

15.3.8 SYSCFG補正セル制御／ステータスレジスタ（SYSCFG_CCCSR）

アドレスオフセット：0x1C

リセット値：0X0000 000A（STM32U535/545/575/585デバイスの場合）

リセット値：0X0000 002A（STM32U59x/5Ax/5Fx/5Gxデバイスの場合）

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのSYSCFGSECビットを設定することによって非セキュアアクセスから保護することができます。SYSCFGSECビットがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	RDY3	RDY2	RDY1	Res.	Res.	CS3	EN3	CS2	EN2	CS1	EN1
					r	r	r			rw	rw	rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **RDY3**：HSPI I/O補正セルレディフラグ

このビットは、V_{DD}によって供給されるHSPI I/Oの補正セルステータスを示します。

0：HSPI I/O補正セルはレディ状態ではありません。

1：HSPI I/O補正セルはレディ状態です。

注： 補正セルが正しく機能するには、HSIクロックが必要です。HSIクロックが有効でない場合、補正セルのレディビット（RDY3）はセットされません。この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット9 **RDY2**：VDDIO2 I/O補正セルレディフラグ

このビットは、V_{DDIO2}によって供給されるI/Oの補正セルステータスを示します。

0：VDDIO2 I/O補正セルはレディ状態ではありません。

1：VDDIO2 I/O補正セルはレディ状態です。

注： 補正セルが正しく機能するには、HSIクロックが必要です。HSIクロックが有効でない場合、補正セルのレディビット（RDY2）はセットされません。

ビット8 **RDY1**：VDD I/O補正セルレディフラグ

このビットは、V_{DD}によって供給されるI/Oの補正セルステータスを示します。

0：VDD I/O補正セルはレディ状態ではありません。

1：VDD I/O補正セルはレディ状態です。

注： 補正セルが正しく機能するには、HSIクロックが必要です。HSIクロックが有効でない場合、補正セルのレディビット（RDY1）はセットされません。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CS3**：HSPI I/Oコードの選択

このビットでは、V_{DD}によって供給されるHSPI I/Oの補正セルに適用するコードを選択します。

0：セルからのHSPI I/Oコード（SYSCFG_CCVRで使用可能）

1：SYSCFG補正セルコードレジスタ（SYSCFG_CCCR）からのHSPI I/Oコード

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット4 **EN3**：HSPI I/O補正セルの有効化

このビットでは、V_{DD}によって供給されるHSPI I/Oの補正セルを有効にします。

0：HSPI I/O補正セルは無効です。

1：HSPI I/O補正セルは有効です。

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットを予約済みと考え、リセット値のままにしてください。

ビット3 **CS2**：VDDIO2 I/Oコードの選択

このビットでは、V_{DDIO2}によって供給されるI/Oの補正セルに適用するコードを選択します。

0：セルからのVDDIO2 I/Oコード（SYSCFG_CCVRで使用可能）

1：SYSCFG_CCCRからのVDDIO2 I/Oコード

ビット2 **EN2**：VDDIO2 I/O補正セルの有効化

このビットでは、V_{DDIO2}によって供給されるI/Oの補正セルを有効にします。

0：VDDIO2 I/O補正セルは無効です。

1：VDDIO2 I/O補正セルは有効です。

ビット1 **CS1**：VDD IDコードの選択

このビットでは、V_{DD}によって供給されるI/Oの補正セルに適用するコードを選択します。

0：セルからのVDD I/Oコード（SYSCFG_CCVRで使用可能）

1：SYSCFG補正セルコードレジスタ（SYSCFG_CCCR）からのVDD I/Oコード

ビット0 **EN1**：VDD I/O補正セルの有効化

このビットでは、V_{DD}によって供給されるI/Oの補正セルを有効にします。

0：VDD I/O補正セルは無効です。

1：VDD I/O補正セルは有効です。

15.3.9 SYSCFG補正セル値レジスタ（SYSCFG_CCVR）

アドレスオフセット：0x20

リセット値：0x0000 0000

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのSYSCFGSECビットを設定することによって非セキュアアクセスから保護することができます。SYSCFGSECビットがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCV3 [3:0]				NCV3 [3:0]			
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCV2 [3:0]				NCV2 [3:0]				PCV1 [3:0]				NCV1 [3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:20 **PCV3[3:0]**：V_{DD}によって供給されるHSPI I/OのPMOS補正値

この値は、セルによって提供され、PMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS3 = 0の場合にI/O補正セルに適用されます。

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット19:16 **NCV3[3:0]**：V_{DD}によって供給されるHSPI I/OのNMOS補正値

この値は、セルによって提供され、NMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS3 = 0の場合にI/O補正セルに適用されます。

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット15:12 **PCV2[3:0]**：V_{DDIO2}によって供給されるI/OのPMOS補正値

この値は、セルによって提供され、PMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS2 = 0の場合にI/O補正セルに適用されます。

ビット11:8 **NCV2[3:0]**：V_{DDIO2}によって供給されるI/OのNMOS補正値

この値は、セルによって提供され、NMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS2 = 0の場合にI/O補正セルに適用されます。

ビット7:4 **PCV1[3:0]**：V_{DD}によって供給されるI/OのPMOS補正値

この値は、セルによって提供され、PMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS1 = 0の場合にI/O補正セルに適用されます。

ビット3:0 **NCV1[3:0]**：V_{DD}によって供給されるI/OのNMOS補正値

この値は、セルによって提供され、NMOSトランジスタのI/O補正セルコードを計算するためにCPUで使用できます。このコードは、SYSCFG_CCCSRでCS1 = 0の場合にI/O補正セルに適用されます。

15.3.10 SYSCFG補正セルコードレジスタ（SYSCFG_CCCR）

アドレスオフセット：0x24

リセット値：0x0000 7878（STM32U535/545/575/585の場合）

リセット値：0x0078 7878（STM32U59x/5Ax/5Fx/5Gxの場合）

システムがセキュアな場合（TZEN = 1）、このレジスタは、SYSCFG_SECCFGRレジスタのSYSCFGSECビットを設定することによって非セキュアアクセスから保護することができます。SYSCFGSECビットがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCC3 [3:0]				NCC3 [3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCC2 [3:0]				NCC2 [3:0]				PCC1 [3:0]				NCC1 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:20 **PCC3[3:0]**：V_{DD}によって供給されるHSPI I/OのPMOS補正コード

これらのビットは、PMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS3 = 1の場合にI/O補正セルに適用されます。

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット19:16 **NCC3[3:0]**：V_{DD}によって供給されるHSPI I/OのNMOS補正コード

これらのビットは、NMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS3 = 1の場合にI/O補正セルに適用されます。

注： この補正セルはHSPIオルタネート機能を備えたGPIOに作用しますが、このモードやAF設定からは独立して機能します。補正セル1はこれらのGPIOには影響を及ぼしません。
このビットフィールドは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。これらのデバイスがない場合は、このビットフィールドを予約済みと考え、リセット値のままにしてください。

ビット15:12 **PCC2[3:0]**：V_{DDIO2}によって供給されるI/OのPMOS補正コード

これらのビットは、PMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS2 = 1の場合にI/O補正セルに適用されます。

ビット11:8 **NCC2[3:0]**：V_{DDIO2}によって供給されるI/OのNMOS補正コード

これらのビットは、NMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS2 = 1の場合にI/O補正セルに適用されます。

ビット7:4 **PCC1[3:0]**：V_{DD}によって供給されるI/OのPMOS補正コード

これらのビットは、PMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS1 = 1の場合にI/O補正セルに適用されます。

ビット3:0 **NCC1[3:0]**：V_{DD}によって供給されるI/OのNMOS補正コード

これらのビットは、NMOSトランジスタのI/O補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSRでCS1 = 1の場合にI/O補正セルに適用されます。

15.3.11 SYSCFG RSSコマンドレジスタ（SYSCFG_RSSCMDR）

アドレスオフセット：0x2C

パワーオンリセット値：0x0000 0000

システムリセット：影響なし

システムがセキュアな場合（TZEN = 1）、このレジスタはAPBアクセスがセキュアな場合のみ読み書きできます。そうでない場合、RAZ/WIです。

システムが非セキュアな場合（TZEN = 0）、このレジスタはRAZ/WIです。このレジスタは、特権および非特権アクセスによって読み書きできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSSCMD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RSSCMD[15:0]**：RSSコマンド

このフィールドでは、RSSによって実行されるコマンドを定義します。

15.3.12 SYSCFG OTG_HS PHYレジスタ（SYSCFG_OTGHSPHYCR）

アドレスオフセット：0x74

リセット値：0x0000 0000

このレジスタは、GTZC1_TZSC_SECCFGR3のOTGSECを設定することによって非セキュアアクセスから保護することができます。OTGSECがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

このレジスタは、STM32U5シリーズの一部のデバイスでのみ使用できます。関連するペリフェラルが使用できるかどうかについては、デバイスのデータシートを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLKSEL[3:0]				PDCTRL	EN
										rw	rw	rw	rw	rw	rw

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5:2 **CLKSEL[3:0]**：OTG_HS PHY参照クロック周波数の選択

0011：16MHz
1000：19.2MHz
1001：20MHz
1010：24MHz
1110：26MHz
1011：32MHz
その他：予約済み

ビット1 **PDCTRL**：OTG_HS PHY共通ブロックのパワーダウン制御

0：SUSPENDで、PHYステートマシン、バイアス、およびOTG PHY PLLが電力供給されている状態です。
1：SUSPENDで、PHYステートマシン、バイアス、およびOTG PHY PLLがパワーダウンされています。

ビット0 **EN**：OTG_HS PHYの有効化

0：PHYはリセット中です。
1：PHYは有効です。

15.3.13 SYSCFG OTG_HS PHY調整レジスタ2（SYSCFG_OTGHSPHYTUNER2）

アドレスオフセット：0x7C

リセット値：0x81CD 06B1（STM32U59x/5Axの場合）

リセット値：0x81CD 06B2（STM32U5Fx/5Gxの場合）

このレジスタは、GTZC1_TZSC_SECCFGR3のOTGSECを設定することによって非セキュアアクセスから保護することができます。OTGSECがセットされている場合、セキュアアクセスのみ可能です。非セキュア読出し／書込みアクセスはRAZ/WIで、不正アクセスイベントが生成されます。

システムが非セキュアな場合（TZEN = 0）、アクセスに関する制約はありません。このレジスタは特権および非特権アクセスで読み書きできます。

このレジスタはSTM32U59x/5Ax/5Fx/5Gxデバイスのみ使用可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TXPREEMPAMPTUNE[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	SQRXTUNE[2:0]			Res.	COMPDISTUNE[2:0]		
	rw	rw							rw	rw	rw		rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:13 **TXPREEMPAMPTUNE[1:0]**：ハイスピード（HS）トランスミッタのプリエンファシス電流制御

11：HSトランスミッタのプリエンファシス回路ソースはプリエンファシス電流の3倍です。
10：HSトランスミッタのプリエンファシス回路ソースはプリエンファシス電流の2倍です。
01：HSトランスミッタのプリエンファシス回路ソースはプリエンファシス電流の1倍です。
00：HSトランスミッタのプリエンファシス回路は無効です。

ビット12:7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **SQRXTUNE[2:0]**：スケルチのしきい値の調整

このビットフィールドでは、有効なハイスピードデータを検出するためのしきい値の電圧レベルを調整します。

000：+15%（推奨値）

011：0%（デフォルト値）

その他：予約済み

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **COMPDISTUNE[2:0]**：切断のしきい値の調整

このビットフィールドでは、ホストでの切断イベントを検出するためのしきい値の電圧レベルを調整します。

010：+5.9%（推奨値）

001：0%（デフォルト値）

その他：予約済み

15.3.14 SYSCFGレジスタマップ

表132.SYSCFGレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SYSCFG_ SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPUSECC	Res.	CLASSBSEC	SYSCFGSEC
	リセット値																													0		0	0
0x04	SYSCFG_CFGR1	Res.	Res.	Res.	SRAMCACHED	Res.	Res.	ENDCAP [1:0]		Res.	Res.	Res.	Res.	PB9_FMP	PB8_FMP	PB7_FMP	PB6_FMP	Res.	Res.	Res.	Res.	Res.	Res.	ANASWVDD	BOOSTEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値				1			0	0					0	0	0	0							0	0								
0x08	SYSCFG_FPUIMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPU_IE[5:0]					
	リセット値																											0	1	1	1	1	1
0x0C	SYSCFG_ CNSLCKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKNSMPU	LOCKNSVTOR
	リセット値																															0	0
0x10	SYSCFG_ CSLCKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKSAU	LOCKSMPU	LOCKSVTAIRCR
	リセット値																														0	0	0
0x14	SYSCFG_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCL	PVDL	SPL	CLL.
	リセット値																													0	0	0	0
0x18	SYSCFG_MESR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPMEE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCLR
	リセット値																0																0
0x1C	SYSCFG_CCCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDY3	RDY2	RDY1	Res.	Res.	CS3	EN3	CS2	EN2	CS1	EN1
	リセット値																						0	0	0			1	0	1	0	1	0

表132.SYSCFGレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x20	SYSCFG_CCVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCV3 [3:0]				NCV3 [3:0]				PCV2 [3:0]				NCV2 [3:0]				PCV1 [3:0]				NCV1 [3:0]			
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SYSCFG_CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCC3 [3:0]				NCC3 [3:0]				PCC2 [3:0]				NCC2 [3:0]				PCC1 [3:0]				NCC1 [3:0]			
	リセット値									0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0
0x28	予約済み	予約済み																															
0x2C	SYSCFG_RSSCMDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSSCMD[15:0]															
	パワーオンリセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30 ~0x70	予約済み	予約済み																															
0x74	SYSCFG_OTGHSPHYCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLKSEL[3:0]				PDCTRL	EN
	リセット値																											0	0	0	0	0	0
0x78	予約済み	予約済み																															
0x7C	SYSCFG_OTGHSPHYTUNER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXPREEMPAMPTUNE[1:0]			Res.	Res.	Res.	Res.	Res.	Res.	SQRXTUNE[2:0]			Res.	COMPDISTUNE[2:0]		
	リセット値																		0	0							0	1	1		0	X	X

レジスタ境界アドレスについては、セクション2.3 を参照してください。

16 ペリフェラル相互接続マトリックス

16.1 相互接続マトリックスの概要

いくつかのペリフェラルには相互に直接接続できるものがあります。これにより、ペリフェラル間での自律的な通信や同期が可能になり、CPUリソースと消費電力を低減できます。

さらに、これらのハードウェア接続によってソフトウェアの遅延をなくし、予測可能なシステム設計が可能になります。

ペリフェラルに応じて、これらの相互接続は各種電力モード、つまりRUN、SLEEP、STOP0、STOP1およびSTOP2のモードで動作可能です。

16.2 接続の一覧

表133.ペリフェラル相互接続マトリックス(1)(2)

転送元	転送先																												
	TIM1	TIM8	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM15	TIM16	TIM17	LPTIM1/2/3	LPTIM4	ADC1/2	ADC4	MDF1	ADF1	OPAMP1/2	DAC1/2	COMP1/2	GP/LPDMA1	IRTIM	U(S)ARTs	LPUART1	I2Cs	SPIs	TAMP	RTC	AES/SAES
TIM1	-	1	1	1	1	1	-	-	1	-	-	-	-	20	20	5	-	-	4	9	-	-	-	-	-	-	-	-	-
TIM8	1	-	1	1	1	1	-	-	1	-	-	-	-	20	-	5	-	-	4	9	-	-	-	-	-	-	-	-	-
TIM2	1	1	-	1	1	1	-	-	1	-	-	-	-	20	20	-	-	-	4	9	16	-	-	-	-	-	-	-	-
TIM3	1	1	1	-	1	1	-	-	1	-	-	-	-	20	-	5	-	-	-	9	16	-	-	-	-	-	-	-	-
TIM4	1	1	1	1	-	1	-	-	1	-	-	-	-	20	-	5	-	-	4	-	16	-	-	-	-	-	-	-	-
TIM5	1	1	1	1	1	-	-	-	1	-	-	-	-	-	-	-	-	-	4	-	16	-	-	-	-	-	-	-	-
TIM6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	20	5	-	-	4	-	-	-	-	-	-	-	-	-	-
TIM7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	-	-	4	-	-	-	-	-	-	-	-	-	-
TIM15	1	1	1	1	1	-	-	-	-	-	-	-	-	20	20	-	-	-	4	9	16	-	-	-	-	-	-	-	-
TIM16	1	1	1	1	1	-	-	-	1	-	-	-	-	-	-	5	-	-	-	-	-	14	-	-	-	-	-	-	-
TIM17	1	1	1	1	1	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	14	-	-	-	-	-	-	-
LPTIM1	-	-	-	-	-	-	-	-	-	-	-	-	-	20	20	5	-	-	4	-	16	-	15	15	15	15	-	-	-
LPTIM2	-	-	-	-	-	-	-	-	-	-	-	-	-	20	-	-	-	-	-	-	16	-	15	-	15	15	-	-	-
LPTIM3	-	-	-	-	-	-	-	-	-	-	-	-	-	20	20	-	-	-	4	-	16	-	-	15	15	15	-	-	-
LPTIM4	-	-	-	-	-	-	-	-	-	-	-	-	-	20	-	-	-	-	-	-	16	-	-	-	-	-	-	-	-
ADC1/2	3	3	-	3	3	-	-	-	-	-	-	-	-	-	-	18	-	-	-	-	16	-	-	-	-	-	-	-	-
ADC4	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	20	-	-
MDF1	6	6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	-	-	-	-	-	-	-	-	-	-	-	-
ADF1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	-	-	-	-	-	-	-	-	-	-	-	-	-
DAC1/2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	17	-	17	-	-	-	-	-	-	-	-	-
COMP1/2	12	12	12	12	12	12	-	-	12	12	12	8	8	-	-	-	-	-	-	-	16	-	15	15	15	15	-	-	-
GPDMA1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	15	-	15	15	-	-	-
LPDMA1	-	-	-	-	-	-	-	-	-	-	-	8	8	-	-	-	-	-	-	-	16	-	-	15	15	15	-	-	-
EXTI	-	-	-	-	-	-	-	-	-	-	-	8	8	20	20	5	5	-	4	-	16	-	15	15	15	15	-	-	-
RTCウェイクアップ	-	-	-	-	-	10	-	-	-	10	10	-	-	-	-	-	-	-	-	-	16	-	15	15	15	15	-	-	-
RTCアラーム	-	-	-	-	-	-	-	-	-	-	-	8	8	-	-	-	-	-	-	-	16	-	15	15	15	15	-	-	-
TAMP	-	-	-	-	-	-	-	-	-	-	-	8	8	-	-	-	-	-	-	-	16	-	-	-	-	-	-	21	22

表133.ペリフェラル相互接続マトリックス[1][2] （続き）

転送元	転送先																												
	TIM1	TIM8	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM15	TIM16	TIM17	LPTIM1/2/3	LPTIM4	ADC1/2	ADC4	MDF1	ADF1	OPAMP1/2	DAC1/2	COMP1/2	GP/LPDMA1	IRTIM	U(S)ARTs	LPUART1	I2Cs	SPIs	TAMP	RTC	AES/SAES
HSE	-	-	-	-	-	-	-	-	-	7	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LSE	-	-	7	-	-	7	-	-	7	7	7	7	-	-	-	-	-	-	19	-	-	-	-	-	-	-	-	-	-
LSEのCSS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	20	-	-
MSIS/MSIK	7	7	7	7	7	7	-	-	-	7	7	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
HSI	7	7	7	7	7	7	-	-	-	7	7	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LSI	-	-	-	-	-	7	-	-	-	7	7	7	-	-	-	-	-	-	19	-	-	-	-	-	-	-	-	-	-
MCO	-	-	-	-	-	-	-	-	-	7	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VCORE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VREFINT	-	-	-	-	-	-	-	-	-	-	-	-	-	17	17	-	-	-	-	17	-	-	-	-	-	-	-	-	-
Tセンサ	-	-	-	-	-	-	-	-	-	-	-	-	-	17	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VBAT	-	-	-	-	-	-	-	-	-	-	-	-	-	17	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VBATおよび温度モニタ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	20	-	-
OPAMP1/2	-	-	-	-	-	-	-	-	-	-	-	-	-	17	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-
システムエラー	13	13	-	-	-	-	-	-	13	13	13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OTG_FS OTG_HS	-	-	11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
システムFlashメモリ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22
AES/SAES	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22
LTDC	-	-	20	20	20	20	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-	-
DSI	-	-	20	20	20	20	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-	-
DMA2D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-	-
GPU2D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-	-
DCMI	-	-	20	20	20	20	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

1. この表の数字は、セクション16.3の対応するサブセクションにリンクしています。
2. 「-」は、相互接続がないことを意味します。

16.3 相互接続の詳細

16.3.1 タイマのマスタ/スレーブ相互接続

タイマ（TIM1/2/3/4/5/8/15/16/17）からタイマ（TIM1/2/3/4/5/8/15）へ

目的

タイマの同期や連携した動作のために、いくつかのTIMxタイマを内部で相互リンクすることができます。

マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

機能の説明については、セクション 55.4.23：タイマの同期を参照してください。

同期モードの詳細については、以下を参照してください。

- セクション54.3.30（高機能制御タイマTIM1/TIM8）
- セクション55.4.22（汎用タイマTIM2/TIM3/TIM4/TIM5）
- セクション56.4.23（汎用タイマTIM15）

トリガ信号

出力（マスタから）は、設定可能なタイマイベントに応じた信号TIMx_TRGO（およびTIM1/8ではTIMx_TRGO2）で行われます。TIM16/17の場合は、信号tim16_oc1およびtim17_oc1でも行われます。入力（スレーブへの）は、信号TIMx_ITR0/1/2/3で行われます。

使用可能なマスタ／スレーブ接続を以下に示します。

- 表532（高機能制御タイマTIM1/8）
- 表557（汎用タイマTIM2/3/4/5）
- 表574（汎用タイマTIM15）

アクティブな電力モード

タイマは、オプションでRUNおよびSLEEPモードでアクティブです。低消費電力モードがTIMxに与える影響については、以下を参照してください。

- 表 546：低消費電力モードが TIM1/TIM8 に与える影響
- 表 566：低消費電力モードがTIM2/TIM3/TIM4/TIM5 に与える影響
- 表 581：低消費電力モードが TIM15/TIM16/TIM17 に与える影響

16.3.2 ADCへのトリガ

EXTI、タイマ（TIM1/2/3/4/5/8/15/16/17）、およびLPタイマ（LPTIM1/2/3/4）からADC1/ADC2へ

EXTI、タイマ（TIM1/2/6/15）、およびLPタイマ（LPTIM1/3）からADC4へ

目的

変換（または一連の変換）は、ソフトウェアによって、または外部イベント（タイマキャプチャまたは入力ピンなど）によってトリガされます。ADC12の場合、EXTEN[1:0]制御ビット（レギュラ変換の場合）またはJEXTEN[1:0]ビット（インジェクト変換の場合）が0b00以外の場合、外部イベントは選択された極性で変換をトリガできます。

詳細については、以下を参照してください。

- セクション 33.4.19：外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN[1:0]、JEXTSEL、JEXTEN[1:0]）

- EXTEN[1:0]は、ADC設定レジスタ（ADC_CFGR1）で定義されています。
- JEXTEN[1:0]は、ADCインジェクトシーケンスレジスタ（ADC_JSQR）で定義されています。

汎用タイマ（TIM2/3/4/5）、基本タイマ（TIM6）、高機能制御タイマ（TIM1/8）、および汎用タイマ（TIM15/16/17）を使用して、タイマ出力tim_ocおよびtim_trgoを通じてADCトリガイベントを生成できます。

チャネル11および15のEXTIに加えて、低電力タイマ（LPTIM1/2/3/4）を使用して、LPTIMチャネル（TIM1/8については、セクション 54.3.31 : ADCトリガに記載されたTIMx同期を参照）を通じて、ADCトリガイベントを生成できます。

ADC4にはインジェクトチャネルはありません。汎用タイマ（TIM2/15）、基本タイマ（TIM6）、および高機能制御タイマ（TIM1）を使用して、タイマ出力tim_ocおよびtim_trgoを通じてADCトリガイベントを生成できます。チャネル11および15のEXTIに加えて、低電力タイマ（LPTIM1/3）を使用して、LPTIMチャネルを通じて、ADCトリガイベントを生成できます。

トリガ信号

ADC1/ADC2の場合、入力トリガ信号と、ADC1/ADC2とタイマとの相互接続の説明については、以下を参照してください。

- adc_ext_trgy：表 303：ADC1/ADC12 のレギュラチャネル用外部トリガ
- adc_jext_trgy：表 304：ADC1/ADC12のインジェクトチャネルの外部トリガ
- セクション 33.4.19 : 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN[1:0]、JEXTSEL、JEXTEN[1:0]）
- セクション 33.4.25：タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

ADC4の場合、入力トリガ信号と、ADC4とタイマとの相互接続の説明については、以下を参照してください。

- 表 325：ADC相互接続
- セクション 34.4.16：外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN）
- セクション 34.4.21：タイミング図の例（シングル／連続モード、ハードウェア／ソフトウェアトリガ）

アクティブな電力モード

この相互接続は、すべてのADCに対してRUNおよびSLEEPモードでアクティブであり、ADC4に対しては、トリガイベントライン（LPTIMなど）がアクティブであれば、STOP0、STOP1、およびSTOP2モードでもアクティブです。タイマは、RUNおよびSLEEPモードでのみアクティブです。低消費電力モードが与える影響については、以下を参照してください。

- 表 546：低消費電力モードが TIM1/TIM8 に与える影響
- 表 566：低消費電力モードがTIM2/TIM3/TIM4/TIM5 に与える影響
- 表 581：低消費電力モードが TIM15/TIM16/TIM17 に与える影響
- セクション 34.5：ADC低電力モード（ADC4）
- 表 593：STM32U5シリーズ LPTIM機能
- 表 608：低消費電力モードがLPTIMに与える影響

16.3.3 タイマのトリガとしてのADCアナログウォッチドッグ

ADC1/ADC4からSTM32U535/345/575/585のTIM1/3/8へ

ADC1/ADC2/ADC4からSTM32U59x/5Ax/5Fx/5GxのTIM1/3/4/8へ

目的

ADC1/ADC2/ADC4から着信する内部アナログウォッチドッグ出力信号は、オンチップタイマに接続されます。ADC1/ADC2/ADC4 は、アナログウォッチドッグ信号を通じて、高機能制御タイマ（TIM1/3/4/8）にトリガイベントを供給し、カウンタのリセット、開始、停止、またはクロック供給を行うことができます。

ADCアナログウォッチドッグとタイマトリガの設定の説明については、以下を参照してください。

- セクション 54.3.6：外部トリガ入力（TIM1/8）
- 表533（TIM1/8（tim_etr）入力マルチプレクサに接続された内部ADC1/ADC2/ADC4ソース）
- 表559（TIM3/4（tim_etr）入力マルチプレクサに接続された内部ADC1/ADC2ソース）
- セクション33.4.30（ADC1/ADC2/ADC_AWDy_OUT信号出力生成）
- セクション34.4.25（ADC4/ADC_AWDy_OUT信号出力生成）

トリガ信号

出力（ADCからの）は、信号ADCn_AWDx_OUTで行われます。nはADCインスタンスであり、x = 1、2、3です（1つのADCにつき3つのウォッチドッグ）。入力（タイマへの）は、信号TIMx_ETR（外部トリガ）で行われます。

アクティブな電力モード

ADC1/ADC2およびADC4は、RUNおよびSLEEPモードでアクティブです。

ADC4変換はSTOP0、STOP1、STOP2モードで自律的に機能します。この変換では、タイマでのウェイクアップ割込みと希望のトリガアクションが生成されます。

16.3.4 DACへのトリガ

タイマ（TIM1/2/4/5/6/7/8/15）、低電力タイマ（LPTIM1/3）、およびEXTIからDAC（DAC1/2）へ

目的

汎用タイマ（TIM2/4/5/15）、基本タイマ（TIM6/7）、高機能制御タイマ（TIM1/8）、LPタイマ（LPTIM1/3）出力チャネル（lptim1_ch1およびlptim3_ch1）、およびEXTIをトリガイベントとして使用して、DAC変換を開始できます。

トリガ信号

TIMx_TRGO信号での出力（タイマからの）とLPタイマからの出力は、対応するDAC入力に直接接続されます。

DACでの入力トリガの選択については、以下を参照してください。

- 表 338：DACの相互接続
- セクション 35.4.8：DACトリガ選択

アクティブな電力モード

この相互接続は、RUN、SLEEP、STOP0、STOP1、およびSTOP2モードでアクティブです。

16.3.5 MDF1またはADF1のトリガ

EXTI、ADF1、タイマ（TIM1/3/4/6/7/8/16）、およびLPタイマ（LPTIM1）からMDF1へ

EXTIおよびMDF1からADF1へ

目的

汎用タイマ（TIM3/4/16）、基本タイマ（TIM6/7）、高機能制御タイマ（TIM1/8）、低電力タイマ（LPTIM1）、EXTI（EXTI11/15）およびADF1を使用して、MDF1モジュールでトリガイベントを生成し、A/C変換を開始することができます。また、EXTI15およびMDF1でADF1をトリガできます。

詳細については、以下を参照してください。

- セクション 39.4.2：MDFピンおよび内部信号
- セクション 39.4.11：開始シーケンスの例

トリガ信号

mdf_trgi[13:0]トリガ入力はトリガ入力信号です。MDFおよびADFトリガ入力接続については、以下を参照してください。

- 表 366：MDFトリガ接続
- 表 387：ADFトリガ接続

アクティブな電力モード

この相互接続は、MDF1に対してはSTOP0およびSTOP1モードまでアクティブのままとなり、ADF1に対しては、トリガソースがアクティブのままであれば、STOP0、STOP1、およびSTOP2モードでアクティブのままとなります。

16.3.6 MDF1からのタイマブレーク

MDF1から高機能制御タイマ（TIM1/8）へ

目的

MDFはアウトオフリミット検出回路（OLD）と短絡検出回路（SCD）機能を備えています。短絡またはオープン回路エラー（過電流、過電圧など）が検出されると、割込みイベントやブレーク信号をTIM1/8に対して生成できます。この動作ついては、以下を参照してください。

- セクション 39.4.7：短絡検出回路（SCD）
- セクション 39.4.9：限界外検出回路（OLD）
- セクション 54.3.18：ブレーク機能の使用
- セクション 54.3.19：双方向ブレーク入力

トリガ信号

mdf1_break[0:3]出力信号はTIM1/8のbreak1およびbreak2入力信号に接続されます。以下の表に、ブレーク信号の割り当てを示します。

- 表 367：MDFブレーク接続
- 表 535：タイマブレーク相互接続
- 表 536：タイマbreak2相互接続

アクティブな電力モード

この相互接続は、RUNおよびSLEEPモードでアクティブです。以下を参照してください。

- セクション 39.4.14：自律モード
- 表 378：低消費電力モードがMDFに与える影響

16.3.7 タイマへのクロックソース

HSE、LSE、LSI、MSIK、HSI、およびMCOからタイマ（TIM1/2/3/4/5/8/15/16/17）およびLPタイマ（LPTIM1/2/3）へ

目的

タイマ入力またはタイマカウンタは、異なるクロックソースから受信でき、たとえば、リファレンスクロックでの内部オシレータの較正に使用できます。

外部クロック（HSE、LSE）、内部クロック（LSI、MSI、HSI）、およびマイクロコントローラ·クロック出力（MCO）は、タイマへの入力として使用できます。

- MSIK/HSIは、外部トリガ信号入力（tim_etr3/tim_etr4）として高機能制御タイマTIM1/8に割り当てられます。MSIK/HSIは、モード2（外部トリガ入力tim_etr_in）の外部クロックソースによって供給されるカウンタクロックとして選択できます。入力割り当てとクロック選択の説明については、以下を参照してください。
  - セクション 54.3.7：クロック選択（TIM1/8）
  - 表 533：STM32U535/545/575/585向けの tim_etr入力マルチプレクサへの相互接続（TIM1/8）
- MSIK、HSI、および LSI は、外部入力信号として汎用タイマ TIM2/3/4/5 に割り当てられます。MSIK/HSI/LSIは、モード1（tim_ti1_in）およびモード2（外部トリガ入力tim_etr_in）の外部クロックソースによって供給されるカウンタクロックとして選択できます。入力割り当てとクロック選択の説明については、以下を参照してください。
  - セクション 55.4.5：クロック選択（TIM2/3/4/5）
  - 外部クロックモード1：表 553：tim_ti1入力マルチプレクサへの相互接続（TIM5、tim_ti1_in1（LSI）、およびtim_ti1_in2（LSE））
  - 外部クロックモード2：表559、tim_etr3（MSIK）、tim_etr4（HSI）、およびtim_etr5（MSIS）（TIM2/TIM3/TIM4/TIM5）
- LSE、LSI、MSI、およびHSIは、外部入力信号として汎用タイマTIM15/16/17に割り当てられます。LSE/LSI/MSI/HSIは、モード1の外部クロックソース（tim_ti1またはtim_ti2信号）によって供給されるカウンタクロックとして選択できます。入力割り当てとクロック選択の説明については、以下を参照してください。
  - 表 56.4.6：クロック選択（TIM15/16/17）.外部クロックモード1：外部入力ピン（使用可能な場合、tim_ti1またはtim_ti2）
  - 表 572：tim_ti1入力マルチプレクサへの相互接続、tim_ti1_in1（LSE-TIM15）、tim_ti1_in5（LSE-TIM16/17）、tim_ti1_in6（LSI-TIM16/17）、tim_ti1_in7/_8（MSI-TIM16/17）、およびtim_ti1_in9（HSI-TIM16/17）
- マイクロコントローラ出力クロック（MCO）は外部入力として汎用タイマTIM16/17に接続されます。これにより、HSI16/MSIシステムクロック（TIM15/16およびLSEで）またはLSI（TIM16およびHSEで）を較正できます。また、LSI（TIM16およびHSI16で）またはMSI（TIM17およびHSI16で）オシレータの周波数を正確に測定するためにも使用されます。ロースピード外部（LSE）オシレータが使用されている場合、追加のハードウェア接続は必要ありません。この機能については、以下を参照してください。
  - セクション 11.4.23：TIM15/TIM16/TIM17を使用した内部／外部クロックの測定

– 表 572：tim_ti1入力マルチプレクサへの相互接続（TIM15/TIM16/TIM17）

- LSIおよびLSEは、表 602：LPTIM1入力キャプチャ 2接続で説明されているように、LPTIM1への入力キャプチャ 2として選択できます。
- HSI/256は、表 603：LPTIM2入力キャプチャ 2接続で説明されているように、LPTIM2への入力キャプチャ 2として選択できます。
- MSI/1024 および MSI/4 は、表 604：LPTIM3 入力キャプチャ 2 接続で説明されているように、LPTIM3への入力キャプチャ 2として選択できます。

トリガ信号

lptim_ic2_mux1 LPTIM入力キャプチャ選択は、LPTIM設定レジスタ2（LPTIM_CFGR2）で設定できます。タイマの場合、内部クロック信号は、モード1（tim_ti1_in）およびモード2（外部トリガ入力tim_etr_in）の外部クロックソースとして選択できます。.

アクティブな電力モード

この機能は、RUNおよびSLEEPモードで使用できます。

16.3.8 低電力タイマへのトリガ

コンパレータ（COMP1/2）、EXTI、TAMP、およびRTCアラームからLPタイマ（LPTIM1/2/3/4）へ

目的

LPTIM1/2/3/4カウンタは、ソフトウェアによって、または8つのトリガ入力の1つでのアクティブエッジの検出後に開始されます（セクション 58.4.7：トリガマルチプレクサを参照）。

GPIOは、LPTIM設定レジスタ2（LPTIM_CFGR2）に従って、LPTIM入力キャプチャ選択またはLPTIM入力選択としても選択できます。

トリガ信号

このトリガ機能については、セクション 58.4.7：トリガマルチプレクサおよび以下のセクション）で説明されています。入力選択については、表 598：LPTIM1/2/3/4外部トリガ接続で説明されています。

アクティブな電力モード

この相互接続は、STOP2モードまでアクティブのままです。

16.3.9 コンパレータへのブランキングソース

タイマ（TIM1/2/3/4/5/8/15/16/17）からコンパレータ（COMP1/2）へ

目的

高機能制御タイマ（TIM1/8）および汎用タイマ（TIM2/3/4/5/15/16/17）は、COMP1/2へのブランキングウィンドウ入力として使用できます。

ブランキング機能については、セクション 37.4.6：コンパレータの出力のブランキング機能で説明されています。

ブランキングソースを次に示します。

- COMP1制御およびステータスレジスタ（COMP1_CSR）、BLANKSEL[4:0]
- COMP2制御およびステータスレジスタ（COMP2_CSR）、BLANKSEL[4:0]

トリガ信号

タイマ出力信号TIMx_Ocxは、COMP1/2のブランキングソースへの入力です。

アクティブな電力モード

この機能は、RUNおよびSLEEPモードで使用できます。

16.3.10 タイマへの入力としてのRTCウェイクアップ

RTCからタイマ（TIM5/16/17）へ

目的

RTCウェイクアップ割込みは、汎用タイマ（TIM5/16/17）チャネル1への入力として使用できます。

トリガ信号

RTCウェイクアップ信号は、TIM5の場合、表 553 : tim_ti1入力マルチプレクサへの相互接続で説明されているように、tim_ti1_in3信号に接続されます。

RTCウェイクアップ信号は、TIM16/TIM17の場合、表 572 : tim_ti1入力マルチプレクサへの相互接続で説明されているように、tim_ti1_in4信号に接続されます。

アクティブな電力モード

この相互接続は、STOP3モードまでアクティブです。タイマはアクティブではありませんが、カウントはウェイクアップ時に実行されます。

16.3.11 タイマへのトリガとしてのOTG_FS/OTG_HS SOF

OTG_FS/OTG_HS SOFからTIM2へ

目的

OTG_FS/OTG_HS SOF（フレーム開始）は、汎用タイマTIM2へのトリガを生成できます。TIM2へのOTG_FS/OTG_HS接続については、表 557 : TIMx内部トリガ接続を参照してください。

トリガ信号

tim_itr11内部信号はOTG_FS/OTG_HS SOFによって生成されます。

アクティブな電力モード

この相互接続は、RUNおよびSLEEPモードでアクティブです。

16.3.12 タイマへの入力、トリガ、またはブレーク信号としてのコンパレータ

コンパレータからタイマ（TIM1/2/3/4/5/8/15/16/17）へ

目的

コンパレータ（COMP1/2）出力値は、タイマ（TIM1/2/3/4/5/8/15/16/17）入力キャプチャ、TIMx_ETR、またはタイマブレーク信号に接続できます。ETRへの接続は、セクション 54.3.6 : 外部トリガ入力を参照してください。

コンパレータ（COMP1/2）出力値は、タイマ（TIM1、TIM8など）用のブレーク入力信号も生成できます。ブレーク（tim_brk）チャネル用のソースは、次のいずれかです。

- 外部：（AFIOコントローラでの選択に従って）TIMx_BKINピンの1つに接続され、極性選択およびオプションのデジタルフィルタリングあり
- 内部：コンパレータtim_brk_cmpx入力から着信するもの（製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照）

トリガ信号

(COMP1/2から着信し)TIM1/8に接続されたtim_etrおよびtim_brk信号については、以下を参照してください。

- tim_etr（表 533：STM32U535/545/575/585向けの tim_etr入力マルチプレクサへの相互接続）：外部トリガ内部入力バス

  これらの入力は、トリガ、外部クロックとして、またはハードウェアのサイクルごとのパルス幅制御に使用できます。
- tim_brk（表 535：タイマブレーク相互接続および表 536：タイマbreak2相互接続）
- セクション 54.3.6：外部トリガ入力
- セクション 54.3.18：ブレーク機能の使用

TIM2/3/4/5の場合、コンパレータとその他の一部のペリフェラルから着信してtim_ti[1:4]入力マルチプレクサに接続されたソースについては、以下を参照してください。

- 表 553：tim_ti1入力マルチプレクサへの相互接続
- 表 554：tim_ti2入力マルチプレクサへの相互接続
- 表 556：tim_ti4入力マルチプレクサへの相互接続
- 表 533：STM32U535/545/575/585向けの tim_etr入力マルチプレクサへの相互接続
- 表 534：STM2U59x/5Ax/5Fx/5Gx向けのtim_etr入力マルチプレクサへの相互接続
- 表 560：tim_ocref_clr入力マルチプレクサへの相互接続
- セクション 55.4.22：タイマと外部トリガの同期
- TIMxタイマ入力選択レジスタ（TIMx_TISEL）（x = 2～5）

TIM15/16/17の場合、コンパレータとその他のペリフェラルから着信してタイマに接続されたソースについては、以下を参照してください。

- 表 573：tim_ti2入力マルチプレクサへの相互接続
- 表 575：タイマブレーク相互接続
- 表 577：ocref_clr入力マルチプレクサへの相互接続
- セクション 56.4.15：ブレーク機能の使用
- セクション 56.4.23：外部トリガ同期（TIM15のみ）

アクティブな電力モード

RUN、SLEEP、および、STOP0、STOP1、およびSTOP2モードでのウェイクアップ機能（トリガソースとして）。入力とブレークは、タイマのアクティビティと同じ低電力モードでは、RUNおよびSLEEPモードでアクティブのままです。

16.3.13 タイマへのブレーク信号としてのシステムエラー

システムエラーからタイマ（TIM1/8/15/16/17）へ

目的

CSS、CPUロックアップ、SRAM2/3 ECCダブルエラー、FLASH ECCダブルエラー検出、およびPVDは、タイマ（TIM1/8/15/16/17）へのタイマブレークの形式でシステムエラーを生成できます。

ブレーク機能の目的は、タイマによって生成されるPWM信号によって駆動する電源スイッチを保護することです。

トリガ信号

可能なブレークのソースについては、以下を参照してください。

- セクション 54.3.18：ブレーク機能の使用（TIM1/8）
- セクション 56.4.15：ブレーク機能の使用（TIM15/16/17）
- 表 537：システムブレーク相互接続（TIM1/8）
- 表 576：システムブレーク相互接続（TIM15/16/17）

アクティブな電力モード

タイマは、オプションでRUNおよびSLEEPモードでアクティブです。低消費電力モードがTIMxに与える影響については、以下を参照してください。

- 表 546：低消費電力モードが TIM1/TIM8 に与える影響
- 表 566：低消費電力モードがTIM2/TIM3/TIM4/TIM5 に与える影響
- 表 581：低消費電力モードが TIM15/TIM16/TIM17 に与える影響

16.3.14 IRTIM信号を生成するタイマ

タイマ（TIM16/17）からIRTIMへ

汎用タイマ（TIM16/17）の出力チャネルTIMx_OC1は、赤外線信号出力の波形を生成するために使用されます。

この機能については、セクション 60：赤外線インタフェース（IRTIM）で説明されています。

16.3.15 通信ペリフェラルのためのトリガ

LPタイマ（LPTIM1/2/3）、コンパレータ（COMP1/2）、GPDMA1転送完了、LPDMA1転送完了、EXTI、RTCアラーム、およびRTCウェイクアップからUSART1/2/3/6、UART4/5、LPUART1、I2C1/2/3/4/5/6、およびSPI1/2/3へ

目的

LPタイマ（LPTIM1/2/3）出力チャネル（lptim1_ch1、lptim1_ch2、およびlptim3_ch1）、コンパレータ（COMP1/2）出力チャネル（comp1_out、およびcomp2_out）、EXTI、RTCアラーム、およびRTCウェイクアップは、選択したUSART、UART、LPUART、I2C、またはSPIペリフェラル上で通信を開始するためのトリガとして使用できます。

GPDMA1転送完了は、GPDMA1レギュラまたはリンクリストの新規転送と選択した通信ペリフェラル上の通信の両方をトリガできます。

LPDMA1転送完了は、LPDMA1新規転送と選択したペリフェラル上の通信の両方をトリガできます。

トリガ信号

トリガからの出力はペリフェラルのトリガ入力に直接接続されます。

入力トリガの選択については、以下を参照してください。

- 表 670：USARTの相互接続（USART1/2/3/6およびUART4/5）
- 表 682：LPUARTの相互接続（LPUART1）
- 表 650：I2C1、I2C2、I2C4、I2C5、I2C6相互接続および表 651：I2C3相互接続

出力（タイマからの）は、信号spi_itrx（x = 6, 7）のSPI1/2/3入力に直接接続されます。SPIでの入力トリガの選択については、以下を参照してください。

- 表 693：SPIの相互作用（SPI1およびSPI2）
- 表 694：SPIの相互接続（SPI3）

アクティブな電力モード

これらの相互接続は、ソースおよび通信ラインの両方がモードにおいて自律状態である場合、RUN、SLEEP、STOPモードでアクティブのままとなります。以下を参照してください。

- セクション 66.6：低消費電力モードのUSART
- セクション 65.5：I^2C低消費電力モード
- セクション 68.6：SPI低電力モード

16.3.16 GPDMA/LPDMAへのトリガ

EXTI、RTC（アラーム/ウェイクアップ）、TAMP（TAMP1/2/3）、タイマ（TIM2/15）、LPタイマ（LPTIM1/3/4）、コンパレータ（COMP1/2）、LP/GPDMA1転送完了（gpdma1_chx_tc/lpdma1_chx_tc）、ADC1/4アナログウォッチドッグ、およびLTDC、DSI、DMA2D、GPU2D、JPEGからLPDMA1、およびGPDMA1へ

目的

LP/GPDMA トリガは、LP/GPDMA チャネルxに割り当てることができます。プログラムされたLP/GPDMA転送は、選択した入力トリガイベントの立上がり／立下がりエッジによってトリガできます。トリガモードも、LLIリンク転送に応じてプログラムできます。詳細については、以下のセクションを参照してください。

- セクション 18.3.5：LPDMAトリガおよびセクション 17.3.5：GPDMAトリガ
- セクション 18.4.12：LPDMAトリガ転送およびセクション 17.4.12：GPDMAトリガ転送
- 以下の詳細については、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）とGPDMAチャネルx転送レジスタ2（GPDMA_CxTR2）を参照してください。
  - トリガ選択TRIGSEL[5:0]フィールド
  - TRIGM[1:0]によって定義されたトリガモード（LLI）
  - TRIGPOL[1:0]によって定義されたトリガ極性

トリガ信号

LPDMAトリガマッピングは、表 148：プログラムされたLPDMA1トリガで指定され、LPDMA_CxTR2.TRIGSEL[5:0]に従います。

GPDMAトリガマッピングは、表 138：プログラムされたGPDMA1トリガで指定され、GPDMA_CxTR2.TRIGSEL[5:0]に従います。

アクティブな電力モード

ソースがSTOPモードまでアクティブであれば、この相互接続はGPDMAの場合はSTOP0、およびSTOP1モードで機能し続け、LPDMAの場合はSTOP0、STOP1、およびSTOP2モードで機能し続けます。

以下を参照してください。

- セクション 18.6：低消費電力モードでのLPDMA
- セクション 17.6：低消費電力モードでのGPDMA

16.3.17 アナログペリフェラルへの内部アナログ信号

内部アナログソースからADC（ADC1/2/4）、コンパレータ（COMP1/2）、およびOPAMP（OPAMP1/2）へ

目的

内部基準電圧（V_{REFINT}）、内部温度センサ（V_{TS}）、およびV_{BAT}監視チャネルは、ADC（ADC1/2/4）入力チャネルに接続されます。また、内部デジタルコア電圧（V_{CORE}）はADC1/2/4入力チャネルに接続されます。

DACチャネル（DAC1_OUT/DAC2_OUT）およびVREFINTはコンパレータ（COMP1/2）に接続されます。

OPAMP1/2出力はGPIOを通じてADC1またはADC4入力チャネルに接続できます。DAC1_OUT1はOPAMP1_VINPに接続できます。DAC1_OUT2はOPAMP2_VINPに接続できます。

以下については、表 337：DAC内部入力／出力信号を参照してください。

- dac_out1アナログ出力DACチャネル1、出力（オンチップペリフェラル用）
- dac_out2アナログ出力DACチャネル2、出力（オンチップペリフェラル用）

これは、以下に従います。

- セクション33.2およびセクション 34.2：ADCの主な機能
- セクション 33.4.11：チャネルの選択（SQRx、JSQRx）
- 表 359：オペアンプの可能な接続
- セクション 37.4.2：COMPピンおよび内部信号

アクティブな電力モード

これらの相互接続は、選択したペリフェラルがアクティブのままであれば、STOPモードで維持されます。以下を参照してください。

- セクション 34.5：ADC低電力モード
- セクション 37.5：COMP低消費電力モード
- セクション 38.4：OPAMP低消費電力モード

16.3.18 MDF1によるADCデータのフィルタリング

ADC1/ADC2からMDF1へ

目的

MDF1では、最大2つのADCをフィルタパスに接続できます。各フィルタについて、MDFデジタルフィルタ設定レジスタx（MDF_DFLTxCICR）のDATSRC[1:0]フィールドを使用して、以下でADCからいずれかのデータが選択されます。

- セクション 33.4.4：ADC接続性

- 表 368：MDF ADCデータ接続
  - mdf_adcitf1_dat[15:0]からadc1_dat
  - mdf_adcitf2_dat[15:0]からadc2_dat
- 表 302：ADC1/ADC12相互接続
  - STM32U535/545/575/585の場合、adcx_dat[15:0]（x = 1）からmdf1_adcx_dat[15:0]
  - STM32U59x/5Ax/5Fx/5Gxの場合、adcx_dat[15:0]（x = 1, 2）からmdf1_adcx_dat[15:0]
- セクション 39.5：MDF低電力モード
  - STOP0およびSTOP1モード

アクティブな電力モード

この機能は、SLEEPモードまで使用可能です。

16.3.19 DACサンプルおよびホールドモードのクロックソース

LSI/LSEからDAC1/2へ

目的

DAC1/2は、STOPモードで動作できます。サンプルおよびホールドブロックとそれに関連するレジスタは、STOPモードでLSIまたはLSEクロックソース（dac_hold_ck）を使用します。

表 337：DAC内部入力／出力信号：

サンプルおよびホールドモードで使用されるdac_hold_ck、入力、DAC低電力クロック

アクティブな電力モード

この機能は、STOP2モードまで使用可能です。

16.3.20 グラフィックインタフェースからタイマへのトリガ

DCMI、LTDC、DSIからタイマ（TIM2/TIM3/TIM4/TIM5）（STM32U59x/5Ax/5Fx/5Gxの場合）およびGFXTIM（STM32U5Fx/5Gxの場合）へ

目的

DCMI、LTDC、およびDSI同期信号は、タイマを開始するためのトリガイベントとして使用できます。

トリガ信号

入力（DCMI、LTDC、およびDSIへ）はタイマ（TIM2/3/4/5）に直接接続されます。外部トリガについては、表 559：STM32U59x/5Ax/5Fx/5Gx向けのtim_etr入力マルチプレクサへの相互接続を参照してください。

GFXTIM入力トリガについては、表 613：GFXTIMトリガの相互接続を参照してください。

アクティブな電力モード

この機能は、SLEEPモードまで使用可能です。セクション 68.6：SPI低電力モードを参照してください。

16.3.21 内部タンパソース

内部ペリフェラル、クロック、または監視からタンパへ

目的

デバイスの破損を引き起こす異常な活動や試みを検出するため、タンパが導入され、システムにそのような望ましくないイベントを警告します。結果的に別のアクションを取ることができます。

タンパソースのリストは、表 637：TAMP相互接続にあります。

アクティブな電力モード

この相互接続は、タンパソースがアクティブな場合、すべての電力モードでアクティブです。

16.3.22 タンパからRTCへの出力

TAMPからRTCへ

目的

RTCでは、このような検出の時間の履歴を取得できるように、タンパイベントのタイムスタンプを記録できます。また、RTCは、GPIOを制御し、MCUの外部のタンパまたはアラームステータスに基づいて信号を設定できます。

詳細については、セクションセクション 63.3.3 : RTCおよびTAMPによって制御されるGPIOを参照してください。

アクティブな電力モード

この相互接続は、すべての電力モードでアクティブなままです。

16.3.23 AES/SAESへの暗号キー

TAMPバックアップレジスタ、システムFlashメモリからSAESとAES、およびその中間へ

目的

暗号メカニズムにはハードウェアキーが必要であり、保護された不揮発性メモリに保存される必要があります。読出し不能な方法でロードするため、別のアプローチが実装されます。タンパバックアップレジスタまたはシステムFlashメモリを使用して、それぞれBHKまたはRHUKを保存し、SAESに渡す専用バスを実装できます。

詳細については、セクション 50.4.9：SAESラップキーを使った の動作を参照してください。

AES暗号メカニズム（SAESより高速）は、SAESの共有キーのメリットを享受できます。詳細については、セクション 49.4.13：共有キーを使った AES の動作 を参照してください。

アクティブな電力モード

AESおよびSAESは、RUNおよびSLEEPモードで動作します。

17 汎用ダイレクト･メモリ･アクセス･コントローラ（GPDMA）

17.1 GPDMAの概要

汎用ダイレクト･メモリ･アクセス（GPDMA）コントローラは、バス･マスタとシステム･ペリフェラルで構成されます。

GPDMAを使用して、CPUを介さず、リンクリストを介してメモリマップペリフェラルやメモリの間でプログラム可能なデータ転送を実行します。

17.2 GPDMAの主な機能

- デュアル双方向AHBマスタ
- メモリマップされたデータの転送元から転送先への転送。
  - ペリフェラルからメモリ
  - メモリからペリフェラル
  - メモリ間転送
  - ペリフェラル間
- 低消費電力モード時の自律（autonomous）データ転送（セクション17.3.2参照）

  チャネルレベルで4グレードでプログラムされた優先順位に基づく転送アービトレーション：
  - タイミング重視のチャネルでの、優先順位が高いトラフィック･クラス（キュー 3）。
  - タイミング重視でないチャネルでの加重ラウンドロビンが割り当てられた､3つの優先順位の低いトラフィック･クラス（キュー 0、1、2）
- 次のいずれかのイベントに対する、チャネルごとのイベント生成：転送完了、1/2転送完了、データ転送エラー、ユーザ設定エラー、リンク転送エラー、中断完了、トリガ･オーバーラン
- イベントごとに別々にプログラムされた割込みイネーブルでの、チャネルごとの割込み生成
- 16個の同時GPDMAチャネル。
  - 転送元／転送先の転送のキューのためのチャネルごとのFIFO（セクション17.3.1参照）
  - 2つの実行モード、つまりrun-to-completionモードおよびリンク･ステップ･モードに対応する、プログラム可能なリンクリストを介してメモリに連結するチャネル内GPDMA転送
  - プログラム可能な GPDMA インプットを経由するチャネル内およびチャネル間 GPDMA 転送チェーンは、GPDMAタスク完了イベントへの接続をトリガします。
- チャネル内の各リンクリスト･アイテムごとに次の操作を行います。
  - 別々にプログラムされた転送元／転送先の転送
  - 転送元／転送先間でのプログラム可能データの処理：バイトベースの順序変更、パッキングまたはアンパッキング、パディングまたは切り詰め、符号拡張、および左詰め／右詰め
  - ブロックレベルを定義する、転送元から転送するプログラム可能なデータバイトの数
  - リニアな転送元および転送先のアドレス指定：固定または連続インクリメント･アドレス指定、ブロックレベルで、連続したバースト転送間でプログラム
  - 2Dの転送元および転送先のアドレス指定：チャネル･セットを減らすための、2番目の2D／繰返しブロック･レベルでの、連続したブロック間のプログラム可能な符号付きアドレス･オフセットと組合せた､連続したバースト転送間のプログラム可能な符号付きオフセット（ブロック内での非連続アドレス指定）（セクション17.3.1参照）

  - 2Dアドレス指定による、スキャッタ／ギャザー（マルチ･バッファ転送）、データ･インターリーブ、デインターリーブのサポート
  - プログラム可能なGPDMAリクエストとトリガ選択
  - プログラム可能なGPDMA1/2転送および転送完了イベントの生成
  - GPDMA リンクリスト制御レジスタの自動更新が行われる、メモリ内での次のリンクリスト･アイテムとそのデータ構造へのポインタ
- DEBUG：
  - チャネル･サスペンドおよび再開のサポート
  - FIFOレベル、イベントフラグを含む、チャネルステータスレポート
- TrustZoneサポート：
  - 最初のチャネルレベルや転送元／転送先、リンク･サブレベルに関係なく、セキュア／非セキュアGPDMA転送をサポートします。
  - それぞれいずれかのセキュア／非セキュアなチャネルから発行される、セキュア／非セキュアな割込みレポート
  - TrustZoneを認識するAHBスレーブ･ポートが、すべてのGPDMAセキュア･リソース（レジスタ、レジスタ･フィールド）を非セキュア･アクセスから保護します。
- 特権／非特権をサポート
  - チャネル･レベルで独立に、特権および非特権GPDMA転送をサポート
  - 特権を認識するAHBスレーブポート。

17.3 GPDMAの実装

17.3.1 GPDMAチャネル

特定のGPDMAチャネルxは、次の機能および使用目的により実装されます。GPDMA性能を最大限に利用するため、ある実装FIFOサイズやリクエストされたGPDMA転送に対して、ユーザがチャネルを選択し割り当てられるよう、下の表にいくつかの一般的な推奨事項を列記します。

表134.GPDMA1チャネル実装

チャネルx	ハードウェア･パラメータ		機能
	dma_fifo_size[x]	dma_addressing[x]	
x = 0 ~ 11	2	0	チャネルx（x = 0~11）を次のとおり実装します。 – 8バイトのFIFO、2ワード – 固定／連続インクリメント･アドレス指定 これらのチャネルは通常、APBまたはAHBペリフェラルとSRAM間のGPDMA転送に割り当てる必要があります。
x = 12 ~ 15	4	1	チャネルx（x = 12~15）を次のとおり実装します。 – 32バイトのFIFO、8ワード – 2Dアドレス指定 これらのチャネルは要求の厳しいAHBペリフェラルとSRAMの間のGPDMA転送、または外部メモリとの間の相互転送に使用することもできます。

17.3.2 低消費電力モードにおけるGPDMA自律モード

GPDMA自律モードとウェイクアップ機能が、下の表に示すようにデバイスの低電力モードで実装されています。

表135.低電力モードでのGPDMA1自律モードとウェイクアップ

機能	低電力モード
自律モードとウェイクアップ	SLEEP、STOP0、およびSTOP1モードのGPDMA1

17.3.3 GPDMAリクエスト

ペリフェラルからのGPDMAリクエストは、SWREQ=0の場合、GPDMA_CxTR2のREQSEL[6:0]でGPDMAチャネルxに割り当てることが可能です。

GPDMAリクエスト･マッピングを下の表に示します。

表136.プログラムされたGPDMA1リクエスト

GPDMA_CxTR2.REQSEL[6:0]	選択されるGPDMAリクエスト
0	adc1_dma
1	adc4_dma
2	dac1_ch1_dma
3	dac1_ch2_dma
4	tim6_upd_dma
5	tim7_upd_dma
6	spi1_rx_dma
7	spi1_tx_dma
8	spi2_rx_dma
9	spi2_tx_dma
10	spi3_rx_dma
11	spi3_tx_dma
12	i2c1_rx_dma
13	i2c1_tx_dma
14	i2c1_evc_dma
15	i2c2_rx_dma
16	i2c2_tx_dma
17	i2c2_evc_dma
18	i2c3_rx_dma
19	i2c3_tx_dma
20	i2c3_evc_dma
21	i2c4_rx_dma
22	i2c4_tx_dma
23	i2c4_evc_dma

表136.プログラムされたGPDMA1リクエスト（続き）

GPDMA_CxTR2.REQSEL[6:0]	選択されるGPDMAリクエスト
24	usart1_rx_dma
25	usart1_tx_dma
26	usart2_rx_dma
27	usart2_tx_dma
28	usart3_rx_dma
29	usart3_tx_dma
30	uart4_rx_dma
31	uart4_tx_dma
32	uart5_rx_dma
33	uart5_tx_dma
34	lpuart1_rx_dma
35	lpuart1_tx_dma
36	sai1_a_dma
37	sai1_b_dma
38	sai2_a_dma
39	sai2_b_dma
40	octospi1_dma
41	octospi2_dma
42	tim1_cc1_dma
43	tim1_cc2_dma
44	tim1_cc3_dma
45	tim1_cc4_dma
46	tim1_upd_dma
47	tim1_trg_dma
48	tim1_com_dma
49	tim8_cc1_dma
50	tim8_cc2_dma
51	tim8_cc3_dma
52	tim8_cc4_dma
53	tim8_upd_dma
54	tim8_trg_dma
55	tim8_com_dma
56	tim2_cc1_dma
57	tim2_cc2_dma
58	tim2_cc3_dma
59	tim2_cc4_dma
60	tim2_upd_dma

表136.プログラムされたGPDMA1リクエスト （続き）

GPDMA_CxTR2.REQSEL[6:0]	選択されるGPDMAリクエスト
61	tim3_cc1_dma
62	tim3_cc2_dma
63	tim3_cc3_dma
64	tim3_cc4_dma
65	tim3_upd_dma
66	tim3_trg_dma
67	tim4_cc1_dma
68	tim4_cc2_dma
69	tim4_cc3_dma
70	tim4_cc4_dma
71	tim4_upd_dma
72	tim5_cc1_dma
73	tim5_cc2_dma
74	tim5_cc3_dma
75	tim5_cc4_dma
76	tim5_upd_dma
77	tim5_trg_dma
78	tim15_cc1_dma
79	tim15_upd_dma
80	tim15_trg_dma
81	tim15_com_dma
82	tim16_cc1_dma
83	tim16_upd_dma
84	tim17_cc1_dma
85	tim17_upd_dma
86	dcmi_dmaまたはpssi_dma(1)
87	aes_in_dma
88	aes_out_dma
89	hash_in_dma
90	ucpd1_tx_dma
91	ucpd1_rx_dma
92	mdf1_flt0_dma
93	mdf1_flt1_dma
94	mdf1_flt2_dma
95	mdf1_flt3_dma
96	mdf1_flt4_dma
97	mdf1_flt5_dma

表136.プログラムされたGPDMA1リクエスト（続き）

GPDMA_CxTR2.REQSEL[6:0]	選択されるGPDMAリクエスト
98	adf1_flt0_dma
99	fmac_read_dma
100	fmac_write_dma
101	cordic_read_dma
102	cordic_write_dma
103	saes_in_dma
104	saes_out_dma
105	lptim1_ic1_dma
106	lptim1_ic2_dma
107	lptim1_ue_dma
108	lptim2_ic1_dma
109	lptim2_ic2_dma
110	lptim2_ue_dma
111	lptim3_ic1_dma
112	lptim3_ic2_dma
113	lptim3_ue_dma
114	hspi1_dma
115	i2c5_rx_dma
116	i2c5_tx_dma
117	i2c5_evc_dma
118	i2c6_rx_dma
119	i2c6_tx_dma
120	i2c6_evc_dma
121	usart6_rx_dma
122	usart6_tx_dma
123	adc2_dma
124	jpeg_rx_dma
125	jpeg_tx_dma

1. どの専用ファンクションを使用するかによる。

17.3.4 GPDMAブロック･リクエスト

GPDMAリクエストによっては、バースト･リクエストではなくブロック･リクエストとしてプログラムする必要があるものがあります。その後、ハードウェア･レベルでの、リクエストされたペリフェラル転送の正しいGPDMAの実行のために、GPDMA_CxTR2のBREQを設定する必要があります。

表137.ブロック･リクエストとしてプログラムされたGPDMA1リクエスト

GPDMAブロック･リクエスト
lptim1_ue_dma
lptim3_ue_dma

17.3.5 GPDMAトリガ

選択したトリガの立上がりエッジまたは立下りエッジがTRIGPOL[1:0]で定義されている場合（TRIGPOL[1:0] = 01またはTRIGPOL[1:0] = 10）、GPDMA_CxTR2のTRIGSEL[6:0]を介してGPDMAチャネルxにGPDMAトリガを割り当てることができます。

表 138.プログラムされたGPDMA1トリガ

GPDMA_CxTR2.TRIGSEL[6:0]	選択されるGPDMAトリガ
0	exti0
1	exti1
2	exti2
3	exti3
4	exti4
5	exti5
6	exti6
7	exti7
8	tamp_trg1
9	tamp_trg2
10	tamp_trg3
11	lptim1_ch1
12	lptim1_ch2
13	lptim2_ch1
14	lptim2_ch2
15	lptim4_out
16	comp1_out
17	comp2_out
18	rtc_alra_trg
19	rtc_alrb_trg
20	rtc_wut_trg
21	予約済み
22	gpdma1_ch0_tc

表 138.プログラムされたGPDMA1トリガ （続き）

GPDMA_CxTR2.TRIGSEL[6:0]	選択されるGPDMAトリガ
23	gpdma1_ch1_tc
24	gpdma1_ch2_tc
25	gpdma1_ch3_tc
26	gpdma1_ch4_tc
27	gpdma1_ch5_tc
28	gpdma1_ch6_tc
29	gpdma1_ch7_tc
30	gpdma1_ch8_tc
31	gpdma1_ch9_tc
32	gpdma1_ch10_tc
33	gpdma1_ch11_tc
34	gpdma1_ch12_tc
35	gpdma1_ch13_tc
36	gpdma1_ch14_tc
37	gpdma1_ch15_tc
38	lpdma1_ch0_tc
39	lpdma1_ch1_tc
40	lpdma1_ch2_tc
41	lpdma1_ch3_tc
42	tim2_trgo
43	tim15_trgo
44	tim3_trgo(1)
45	tim4_trgo (1)
46	tim5_trgo (1)
47	ltdc_li
48	dsi_te
49	dsi_er
50	dma2d_tc
51	dma2d_ctc
52	dma2d_tw
53	gpu2d_flag[0]
54	gpu2d_flag[1]
55	gpu2d_flag[2]
56	gpu2d_flag[3]
57	adc4_awd1
58	adc1_awd1
59	gfxtim_ev4

表 138.プログラムされたGPDMA1トリガ （続き）

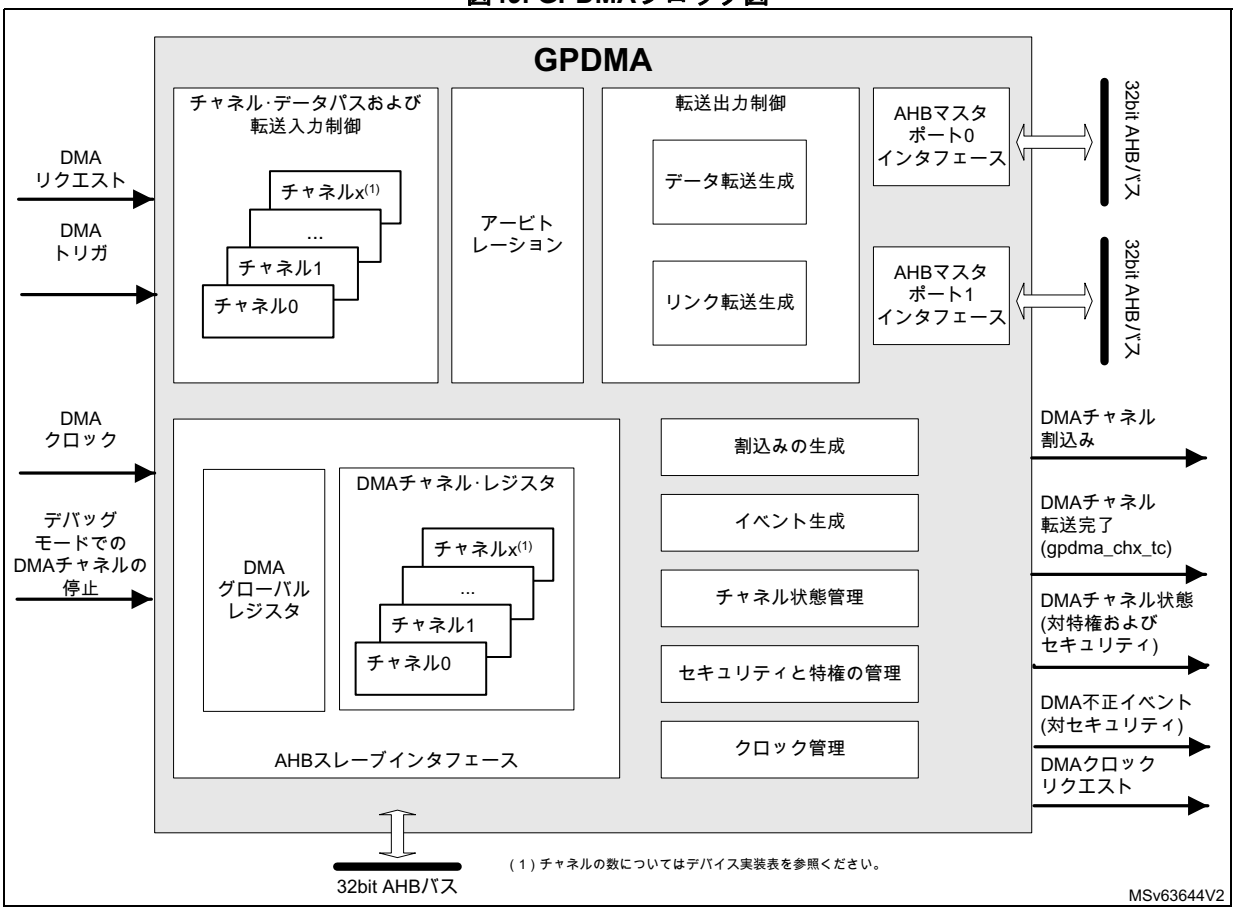
GPDMA_CxTR2.TRIGSEL[6:0]	選択されるGPDMA トリガ
60	gfxtim_ev3
61	gfxtim_ev2
62	gfxtim_ev1
63	jpeg_eoc_trg
64	jpeg_ifnf_trg
65	jpeg_ift_trg
66	jpeg_ofne_trg
67	jpeg_oft_trg

1. 接続が存在するのはSTM32U59x/5AxおよびSTM32U5Fx/5Gxのみです。

17.4 GPDMAの機能説明

17.4.1 GPDMAブロック図

図49. GPDMAブロック図



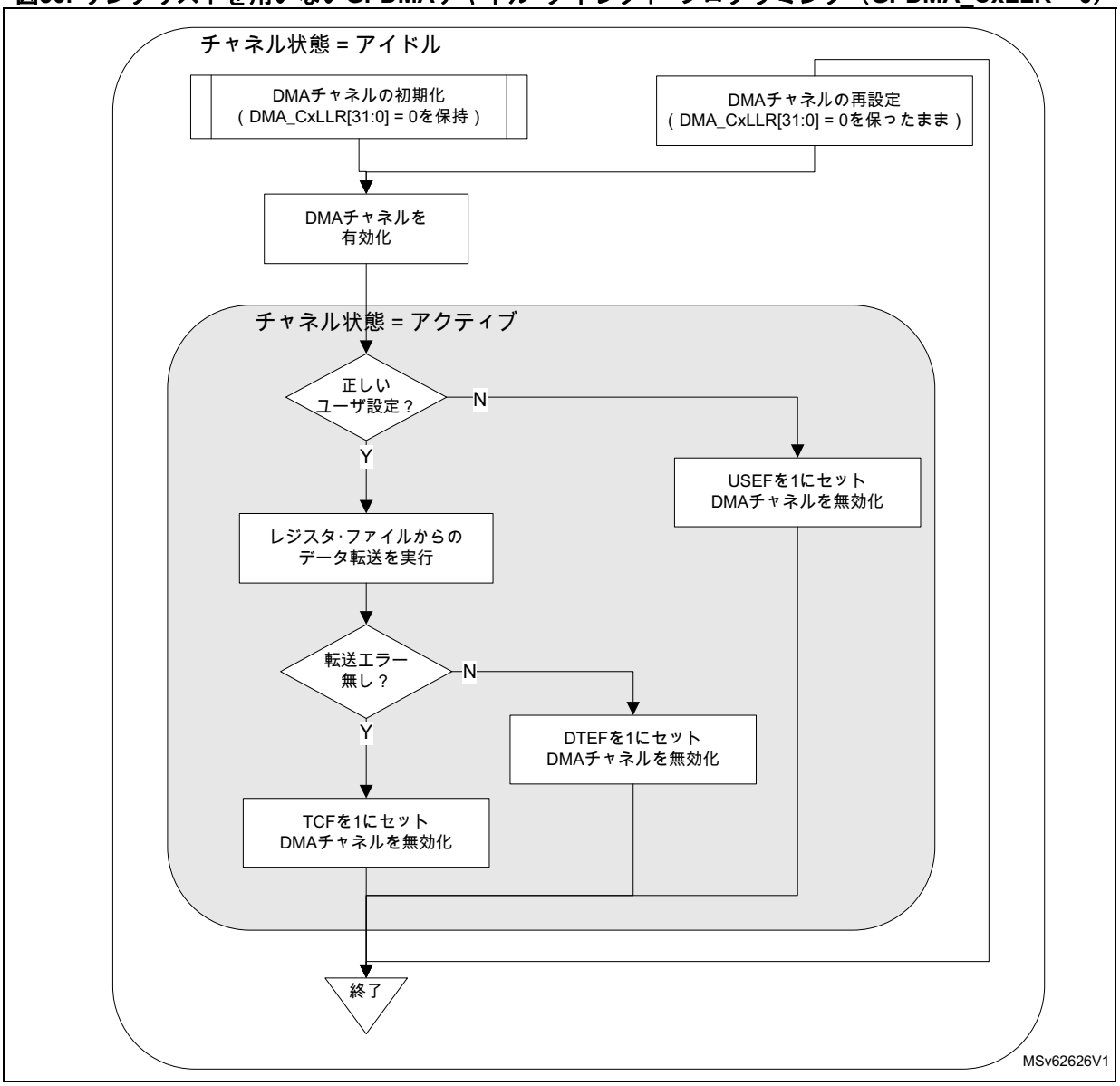
17.4.2 GPDMAチャネル状態とリンクリストなしの直接プログラミング

GPDMAリセット後、GPDMAチャネルxはアイドル状態となります。GPDMA_CxCR.ENにソフトウェアが1を書き込むと、そのチャネルは他のチャネルの設定レジスタ（GPDMA_CxXXX）の値を考慮して有効／非アイドル状態に切り替わり、対応するリクエストされたデータの転送の実行を開始します。

GPDMA_CxCR.ENに1を書き込むことでGPDMAチャネル転送を有効化／開始した後、GPDMAチャネルがアイドル状態に戻り（その後、ENがハードウェアによりデアサートされ）、再設定と再有効化の準備が整っていることを、転送完了時にGPDMAチャネル割込みがソフトウェアに通知します。

以下の図に、リンクリストなしのこのGPDMAの直接プログラミングを示します（GPDMA_CxLLR = 0）。

図50. リンクリストを用いないGPDMAチャネル･ダイレクト･プログラミング（GPDMA_CxLLR = 0）



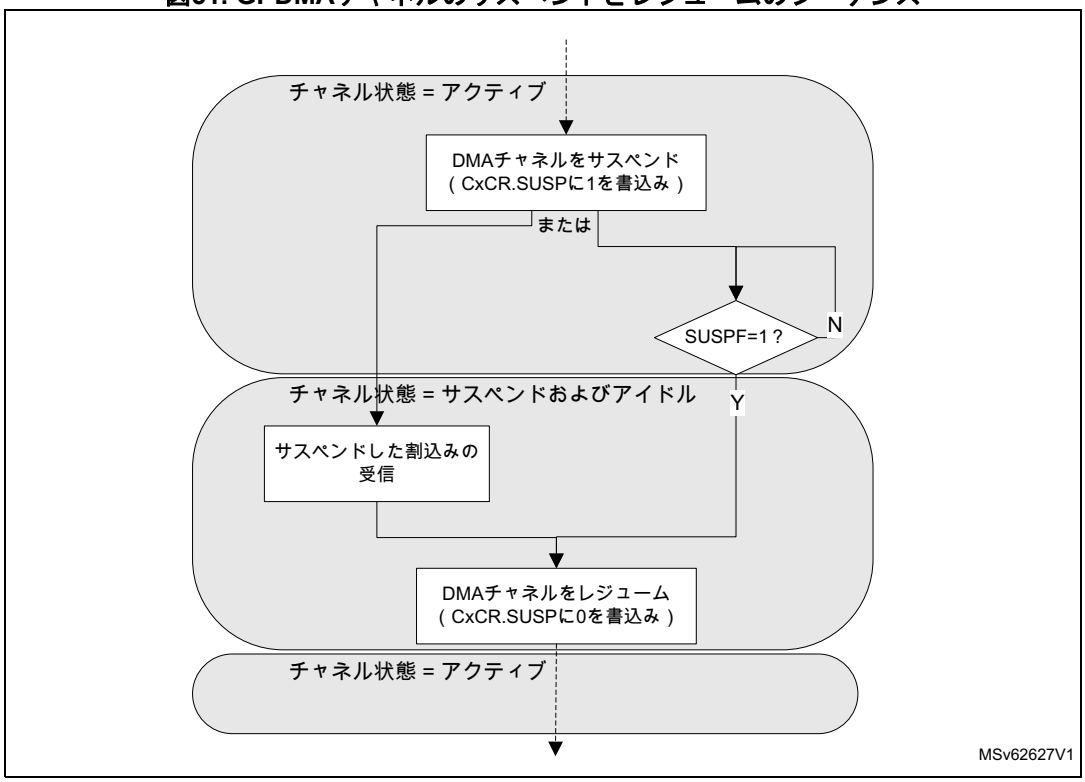
17.4.3 GPDMAチャネルのサスペンドとレジューム

次の手順により、有効になっているチャネルをソフトウェアのみでサスペンドすることが可能です。

1. ソフトウェアでGPDMA_CxCR.SUSPビットに1を書き込みます。
2. ソフトウェアは、SUSPF = 1になるまで、サスペンドしたフラグGPDMA_CxSR.SUSPFをポーリングするか、GPDMA_CxCR.SUSPIEに1を書き込むことにより前に有効化された割込みを待ちます。チャネルが効果的にサスペンド状態になるまで待つということは、マスタポート上で実行中のGPDMA転送の完了を待つということです。その後ソフトウェアは定常状態で、あらゆる読出しレジスタ、またはハードウェアによる変更が可能なレジスタ･フィールドを観察することが可能になります。

   進行中のGPDMA転送は、データ転送（転送元／転送先間のバースト転送）あるいは、次のリンクリスト･アイテムからのリンクリスト･レジスタ･ファイルの内部更新のためのリンク転送の可能性があることに注意してください。
3. ソフトウェアは、GPDMA_CxCR.SUSPに0を書き込むことでサスペンドしたチャネルを安全にレジュームします。

図51. GPDMAチャネルのサスペンドとレジュームのシーケンス



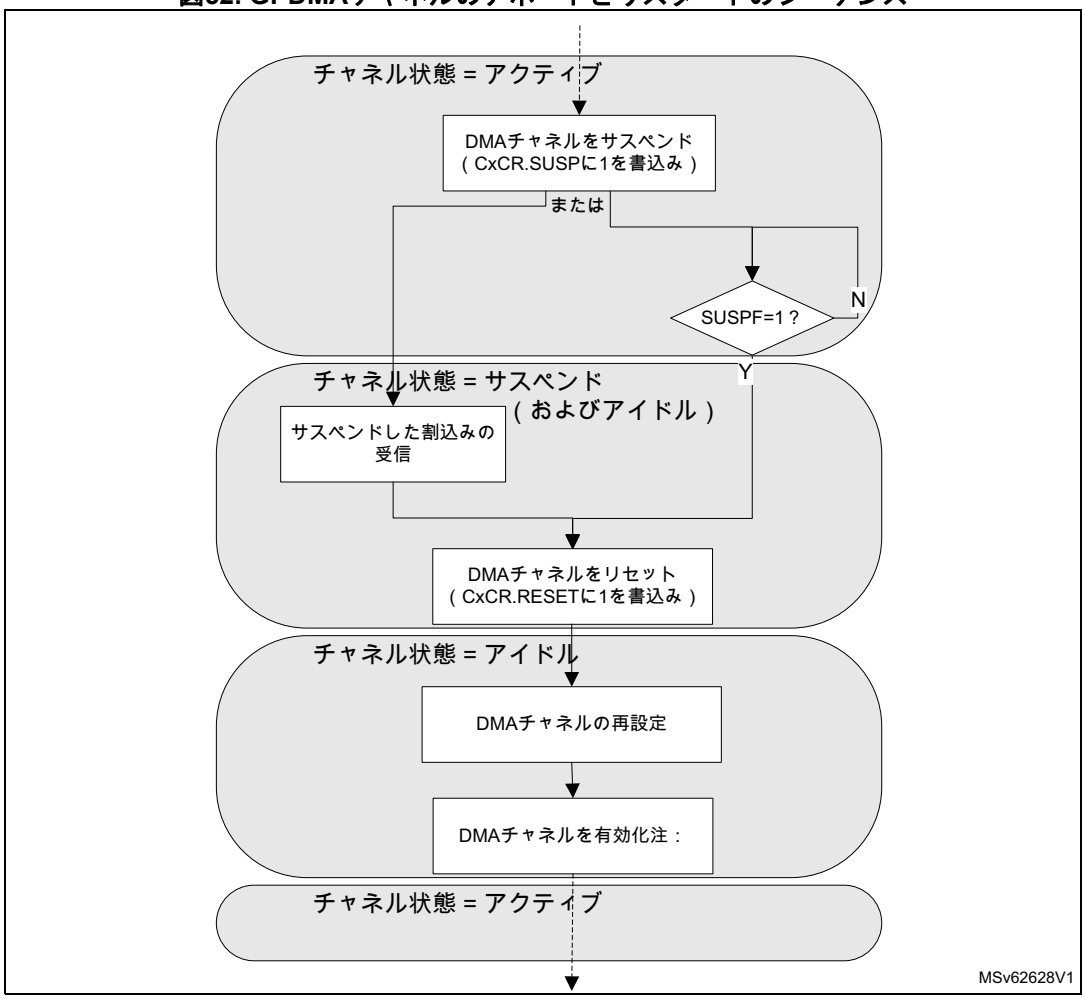
注： **サスペンドとレジュームのシーケンスはGPDMA_CxCR.ENビットには影響しません。チャネル（転送）のサスペンドが、開始したトリガ検出をサスペンドすることはありません。**

17.4.4 GPDMAチャネルのアボートとリスタート

サーキュラ･バッファまたはダブル･バッファによる連続GPDMA転送のアボート同様、以下に示すシーケンスにより、いまだに有効なチャネルをソフトウェアのみでアボートすることも可能です。

1. ソフトウェアでGPDMA_CxCR.SUSPビットに1を書き込みます。
2. ソフトウェアは、SUSPF = 1になるまで、サスペンドしたフラグGPDMA_CxSR.SUSPFをポーリングするか、GPDMA_CxCR.SUSPIEに1を書き込むことにより前に有効化された割込みを待ちます。チャネルが効果的にサスペンド状態になるまで待つということは、マスタポート上で実行中のGPDMA転送の完了を待つということです。
3. ソフトウェアはGPDMA_CxCR.RESETに1を書き込むことにより、チャネルをリセットします。これにより、FIFOがリセットされ、チャネルの内部状態がリセットされ、GPDMA_CxCR.ENビットとGPDMA_CxCR.SUSPビットがリセットされます。
4. ソフトウェアがチャネルを安全に再設定します。ソフトウェアは、ハードウェアにより変更されたGPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDARレジスタを再プログラムする必要があります。
5. ここでアボートしてから再プログラムされたチャネルをリスタートするために、ソフトウェアがGPDMA_CxCR.ENビットに1を書き込むことでこれを再び可能にします。

図52. GPDMAチャネルのアボートとリスタートのシーケンス



17.4.5 GPDMAのリンクリスト･データ構造

直接プログラミング･モードの他、チャネルは、リンクリスト･アイテム（LLI）のリストとして知られている転送リストによりプログラムすることが可能です。各LLIはそのデータ構造により定義されます。

チャネルxの次のLLI_{n+1}のデータ構造を持つメモリのベース･アドレスは以下の合計です。

- （GPDMA_CxLBARでの）チャネルxのリンク･ベース･アドレス
- リンク･アドレス･オフセット（GPDMA_CxLLRのLA[15:2]フィールド）。リンクリスト･レジスタGPDMA_CxLLRは、チャネルxの前のLLInのデータ構造から更新された結果です。

各LLIのデータ構造は、それぞれ異なる場合があります。

リンクリストのデータ構造は、GPDMA_CxLLRのUT1、UT2、UB1、USA、UDA、ULLビットの値に加え、UB2とUT3の値に従ってアドレス指定されます。

リンクリスト･モードでは、リンクリスト･レジスタ（GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDARまたはGPDMA_CxLLRに加えGPDMA_CxTR3またはGPDMA_CxBR2）は、条件付きで自動的に次のリンクリスト･データのメモリ構造から、GPDMA_CxLLRレジスタの現在の値に従って更新されます。このGPDMA_CxLLRレジスタの現在の値は、前のLLIのリンクリスト･データ構造から条件付きで更新されたものです。

静的リンクリスト･データ構造

たとえば、GPDMA_CxLLRの更新ビット（UT1、UT2、UB1、USA、UDA、ULLに加え、UB2およびUT3）がすべてアサートされている場合、メモリ内のリンクリストデータ構造は最大で次のようになります。

- GPDMA_CxTR1、GPGPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、およびGPDMA_CxLLR（図53を参照）と、メモリの最初のリンクリスト･レジスタ･ファイル（LLI_0）および次のLLI（LLI_1、LLI_2など）を含む、チャネルx（x = 0 to 11）の連続32bit位置。
- GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、GPDMA_CxLLRに加え、GPDMA_CxTR3、GPDMA_CxBR2（図54を参照）と、メモリの最初のリンクリスト･レジスタ･ファイル（LLI_0）および次のLLI（LLI_1、LLI_2など）を含む、チャネルx（x = 12~15）の連続32bit位置。

図53. チャネルxのリニア･アドレス指定の静的リンクリスト･データ構造（すべてUxx = 1）

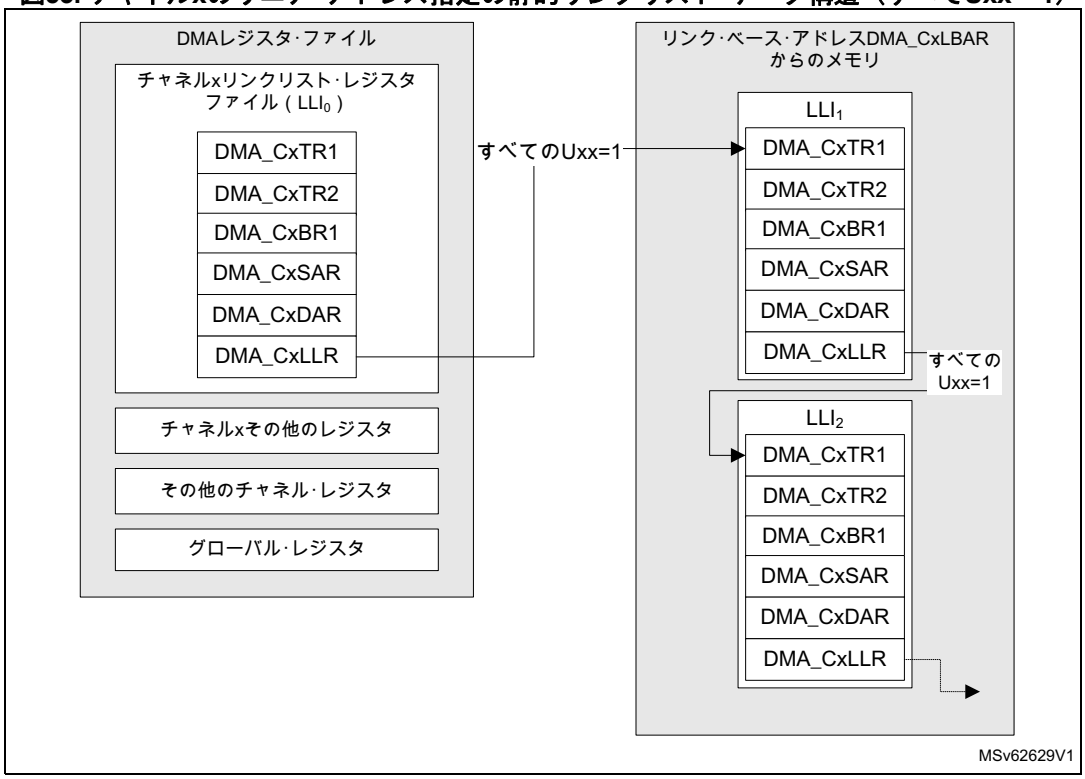
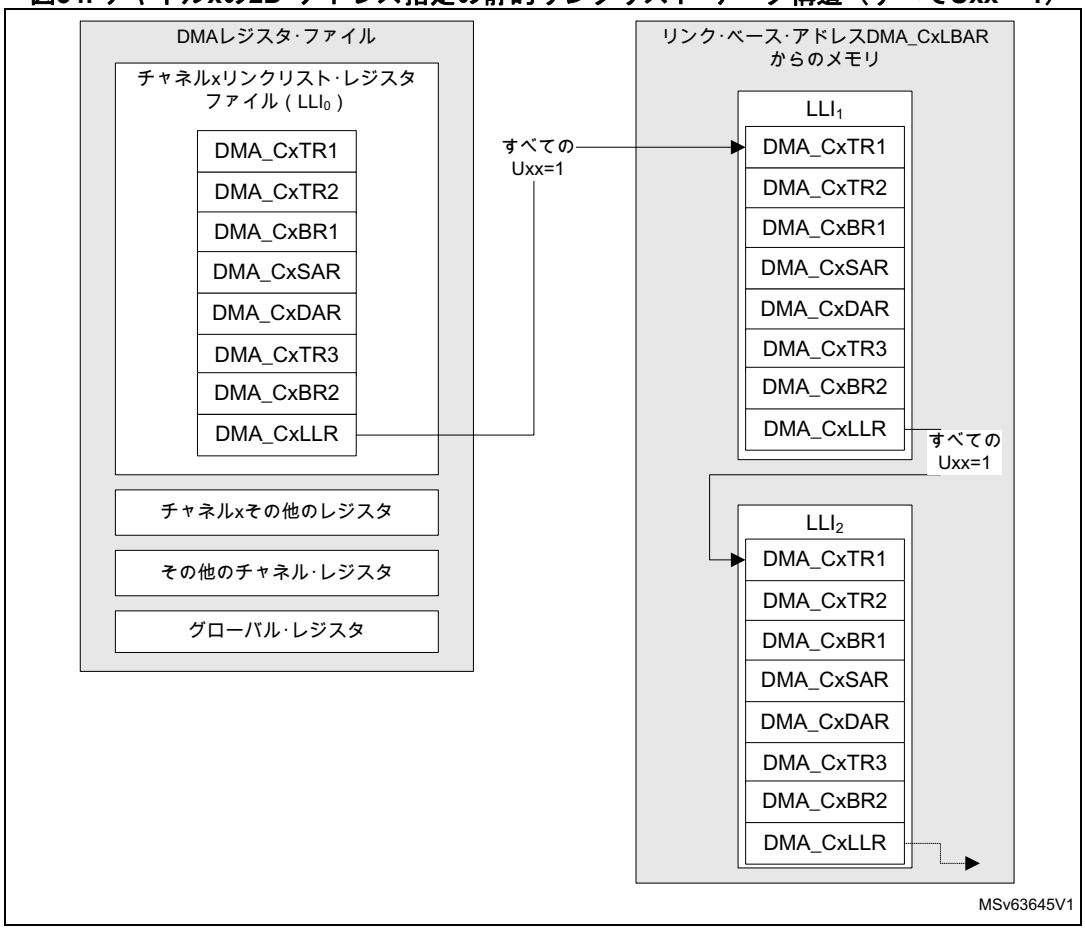


図54. チャネルxの2D・アドレス指定の静的リンクリスト・データ構造（すべてUxx = 1）



動的リンクリスト・データ構造

LLIの全リストのメモリ構成は、各LLI特有のデータ構造でコンパクト化が可能です。

UT1 = 0かつUT2 = 1の場合、レジスタGPDMA_CxLLRのリンク・アドレス・オフセットは、変更しないGPDMA_CxTR1の代わりに、GPDMA_CxTR2の更新された値を指します（図55参照）。

例 : チャネルxが2Dアドレス指定されUT1 = UB1 = USA = 0かつUT3 = UB2 = 0の場合、およびUT2 = UDA = ULL = 1の場合、次のLLIは、GPDMA_CxTR1、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxTR3、GPDMA_CxBR2 の（更新された）値をいずれも含みません。図 56 に示すように、次の LLI は GPDMA_CxTR2、GPDMA_CxDAR、GPDMA_CxLLRの更新された値を含みます。

図55. チャネルxのリニア・アドレス指定のGPDMA動的リンクリスト・データ構造

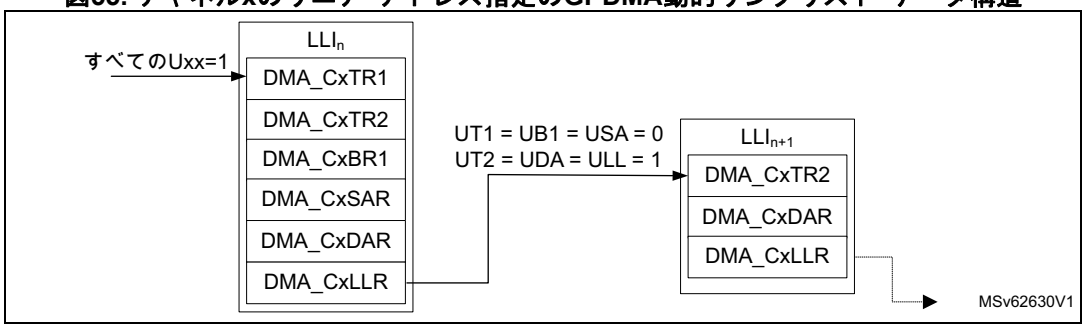
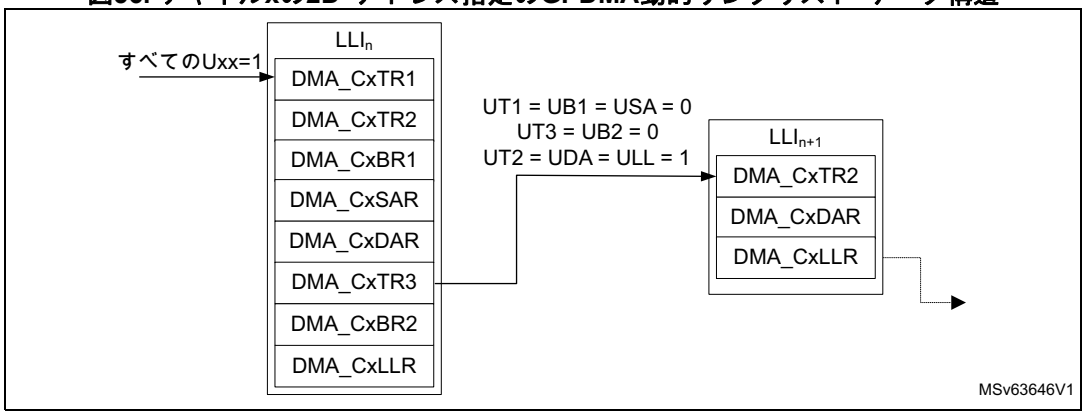


図56. チャネルxの2D･アドレス指定のGPDMA動的リンクリスト･データ構造



ユーザは各LLI_nのGPDMA_CxLLRが32bitに整列し、GPDMA_CxLBARが指す64KBのアドレス空間を超えないようにプログラムする必要があります。

17.4.6 リンクリスト･アイテムの転送実行

LLI_n転送は次のシーケンスです。

1. データ転送 : GPDMAは、GPDMA内部レジスタ･ファイルによって記述されたとおりにデータ転送を実行します（このデータ転送は、LLI_0ではvoid/nullにすることができます）
2. 条件付きリンク転送 : LLI_nのGPDMA_CxLLR値で定義されるように、次のLLI_{n+1}のデータ構造により内部レジスタ･ファイルをGPDMAが条件付きで自動的に更新。

注 : **内部レジスタ･ファイル（LLI_0）により定義される最初のデータ転送は、GPDMA_CxLLRの条件付き更新ビットUB1が設定されている（つまり、メモリの中の次のLLI_1により記述されるnullではないデータ転送が実行されることになっている）場合、null（GPDMA_CxBR1.BNDT[15:0] = 0）にすることができます。**

次のセクションで説明するとおり、意図するGPDMAの利用方法に従い、GPDMAチャネルxはフル･リンクリストの記述に従って実行する（run-to-completionモード、GPDMA_CxCR.LSM = 0）か、あるいは、LLIのシングル実行（リンク･ステップ･モード、GPDMA_CxCR.LSM=1）をプログラムすることが可能です。

17.4.7 GPDMAチャネル状態とrun-to-completionモードでのリンクリスト･プログラミング

GPDMA_CxCR.LSM = 0（フル･リスト実行モードでは、LLIのフル･シーケンスの実行はrun-to-completionモードと呼ばれます）の場合、GPDMAチャネルxは、GPDMA_CxCR.ENに1を書き込むことで開始し、チャネル･レベルで完了した後、最初にプログラムされます。チャネル転送は次のとおりです。

- 少なくとも次の設定をします。
  - 最初の LLI_0、内部リンクリストレジスタ･ファイル : GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、GPDMA_CxLLRに加え、GPDMA_CxTR3およびGPDMA_CxBR2
  - 最後のLLI_N、最後の前のLLI_{N-1}を反映したGPDMA_CxLLRで定義された、メモリ内のリンクリストデータ構造に記述されたもの
- 最後のLLI_{N-1}転送の終わりに、GPDMA_CxLLR[31:0] = 0、GPDMA_CxBR1.BRC[10:0] = 0、GPDMA_CxBR1.BNDT[15:0] = 0となったときに完了

GPDMA_CxLLR[31:0] = 0は、リンクリスト･ベースのチャネル完了の条件で、次のことを意味しています。

- 次のリンク･アドレスの下位16bit GPDMA_CxLLR.LA[15:0]がnull。
- すべての更新されたビットGPDMA_CxLLR.Uxxがnull（UT1、UT2、UB1、USA、UDA、ULLに加え、UB2およびUT3）。

GPDMA_CxLLR.LSM = 0の場合、チャネルは決して完了しません。

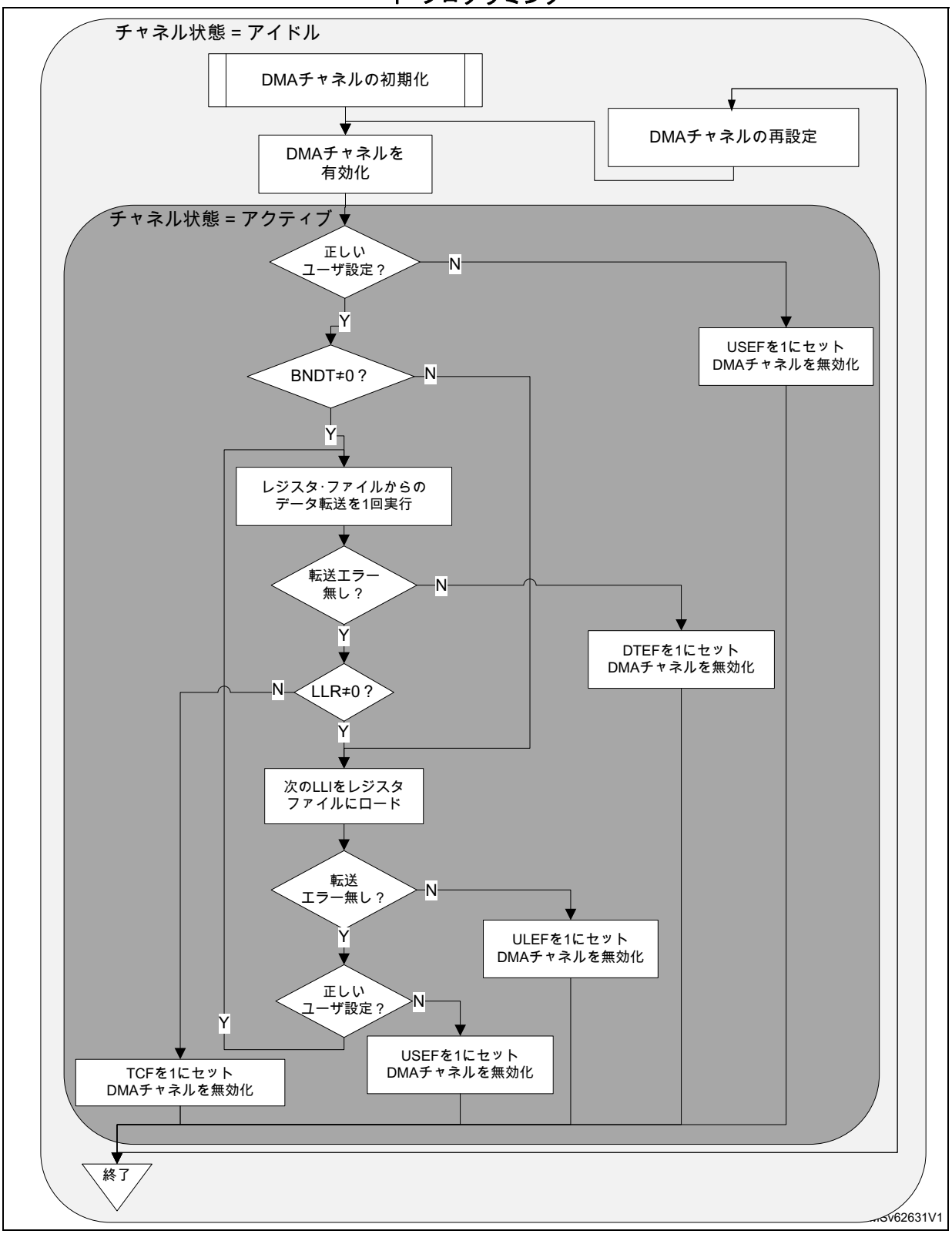
- 最後のLLI_Nが再帰的で、次のLLIとしてそれ自身を指す場合：
  - GPDMA_CxLLR.ULL = 1またはGPDMA_CxLLR.LA[15:2]のどちらかが同じ値で更新されます。
  - あるいはGPDMA_CxLLR.ULL = 0です。
- LLI_Nが前のLLIを指す場合。

通常のrun-to-completionモードでは、GPDMAチャネルの割当は、詳細プログラミングを含め、GPDMAの初期化の際に1度行われます。実行中に予約済みのデータ通信リンクとGPDMAサービスを確保するため、転送が連続的に反復される場合（ペリフェラルとメモリ間、またはメモリ間転送）、この予約済みのデータ通信リンクは1つのチャネルで構成することができ、チャネルを共有することもできます。反復される転送は、LLIのシーケンスで構成されます。

run-to-completionモードでのGPDMAチャネル実行と、そのレジスタのプログラミングを図57に示します。

注： **図57は、GPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、またはチャネルの完了時のみに立てることができます。run-to-completionモードでは、チャネル完了時のTCEM[1:0]の値に関わらず、ハードウェアは常にTCEF =1を設定し、そのチャネルを無効にします。**

図57. run-to-completionモード(GPDMA_CxCR.LSM = 0)でのGPDMAチャネル実行およびリンクリスト･プログラミング



run-to-completionモードでの補助チャネル経由のLLI$_n$の実行時挿入

LLI$_{n-1}$のリンク転送の開始（LLI$_n$ロードの開始）は、LLI$_{n-1}$のデータ構造のGPDMA_CxTR2の以下のフィールドをプログラミングする場合、トリガの発生を条件とすることができます。

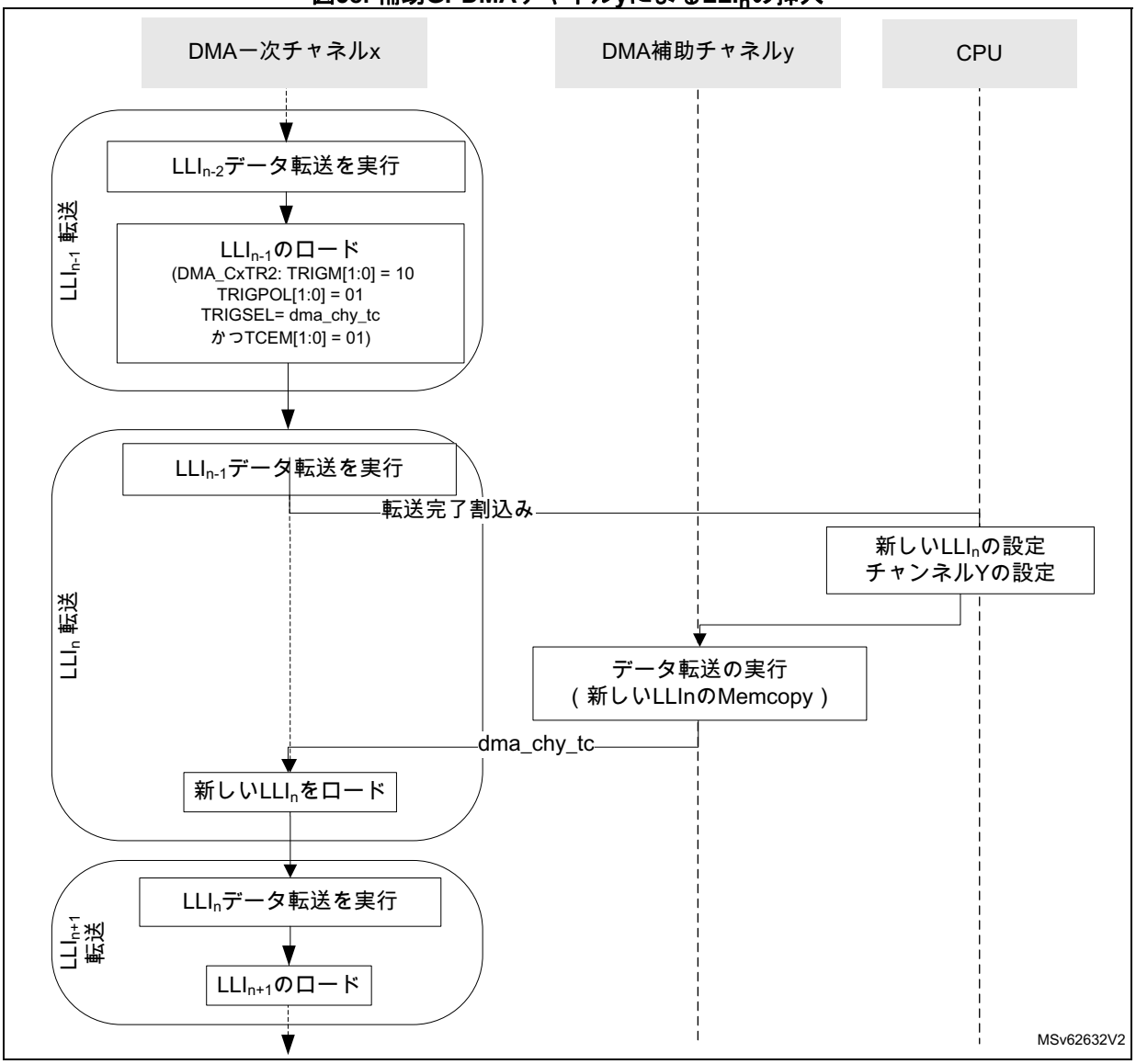
- TRIGM[1:0] = 10（リンク転送トリガ･モード）
- TRIGPOL[1:0] = 01または10（立上がり／立下りエッジ）
- TRIGSEL[6:0]（トリガ選択の詳細についてはセクション17.3.5を参照）

別の補助チャネル y は、チャネル xLLI$_n$ をメモリに格納するため、および転送完了イベントgpdma_chy_tcの生成に使用することができます。このイベントをチャネルxのLLI$_{n-1}$のリンク転送の入力トリガとして選択することで、ソフトウェアはそのLLI$_{n-1}$データ転送後、実際にLLI$_n$が実際に書き込まれるまで、一次チャネルxを休止することが可能です。

下図に、別の補助チャネルy経由での一次チャネルxのリンクリストの動的処理を示します。

注意： **この使用事例はトリガを必要としないLLI$_{n-1}$データ転送のアプリケーションに限定されています。このLLI$_{n-1}$のトリガ･モードは、次のLLI$_n$のロードに使用されます。**

図58. 補助GPDMAチャネルyによるLLI$_n$の挿入



17.4.8 リンク･ステップ･モードでのGPDMAチャネル状態およびリンクリスト･プログラミング

GPDMA_CxCR.LSM = 1（リンク･ステップ実行モードでの1つのLLIのシングル実行）の場合、LLIの各シングル実行後に、その（条件付き）データ転送と（条件付き）リンク転送を含む、チャネル転送を実行して完了します。

GPDMAチャネル転送はLLIレベルでプログラムし、GPDMA_CxCR.ENに1を書き込んで開始することが可能で、LLIレベルで完了した後、次のようになります。

- 現在のLLI$_n$転送は次のように記述されます。
  - GPDMA_CxTR1が転送元／転送先の基本シングル／バースト転送を定義。
  - GPDMA_CxBR1が、ブロック･レベルでのバイト数（BNDT[15:0]）、チャネルx(x = 12~15)での、2D／繰返しブロック･レベル（BRC[10:0]+1）のブロック数およびアドレス･オフセットのインクリメント／デクリメント･モードを定義します。

  - GPDMA_CxTR2が転送の入力制御（リクエスト、トリガ）および出力制御（転送完了イベント）を定義します。
  - GPDMA_CxSAR/GPDMA_CxDARが転送元／転送先の転送開始アドレスを定義します。
  - チャネルx（x = 12~15）のGPDMA_CxTR3が、バースト転送間の転送元／転送先の追加アドレス･オフセットを定義します。
  - チャネルx（x = 12~15）のGPDMA_CxBR2が、2D／繰返しブロック･レベルでのブロック間の転送元／転送先の追加アドレス･オフセットを定義します。
  - GPDMA_CxLLRが、データ構造とメモリの中の次のLLI_{n+1}アドレス･オフセットを定義します。
- 現在のLLI_nのシングル実行後に現在のLLI_n転送が完了します。
  - （条件付き）データ転送の完了後（GPDMA_CxBR1.BRC[10:0] = 0およびGPDMA_CxBR1.BNDT[15:0] = 0の場合）
  - メモリの中の次のLLI_{n+1}のデータ構造からのGPDMAリンク･レジスタ･ファイルの（条件付き）更新後

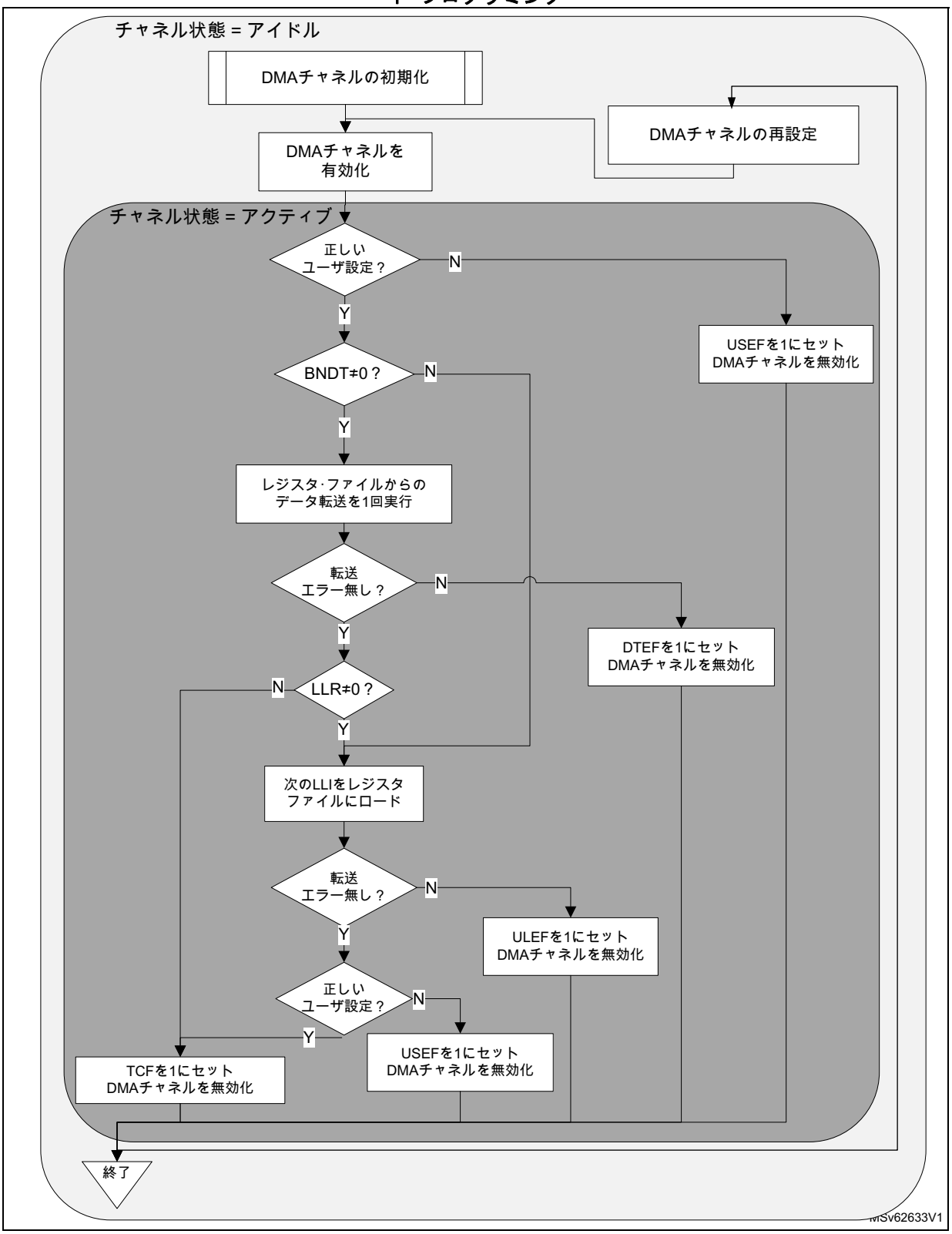
注： **LLIが再帰的である場合（次のLLIとしてそれ自身を指し、GPDMA_CxLLR.ULL = 1かつGPDMA_CxLLR.LA[15:2]が同じ値で更新されるか、GPDMA_CxLLR.ULL = 0の場合）、リンク･ステップ･モードのチャネルは、 このLLIのシングル実行の各繰返しの後に完了します。**

リンク･ステップ･モードを使用すると、実行中にメモリ内でLLIを動的に処理できます。あらゆるLLI_nに静的データ構造を使用することにより、ソフトウェアを支援することができます（GPDMA_CxLLRのすべての更新ビットが静的な値LLI_n.LLR.LA = LLI_{n-1}.LLR.LA + 定数を持ちます）。

図59にGPDMAチャネル実行モードと、そのリンク･ステップ･モードでのプログラミングを示します。

注： **図59は、GPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、または最後のLLIデータ転送の完了時のみに立てることができます。リンク･ステップ･モードでは、LLIの各シングル実行後にチャネルが無効になり、TCEM[1:0]の値によりTCEFを立てるかどうかが変わります。**

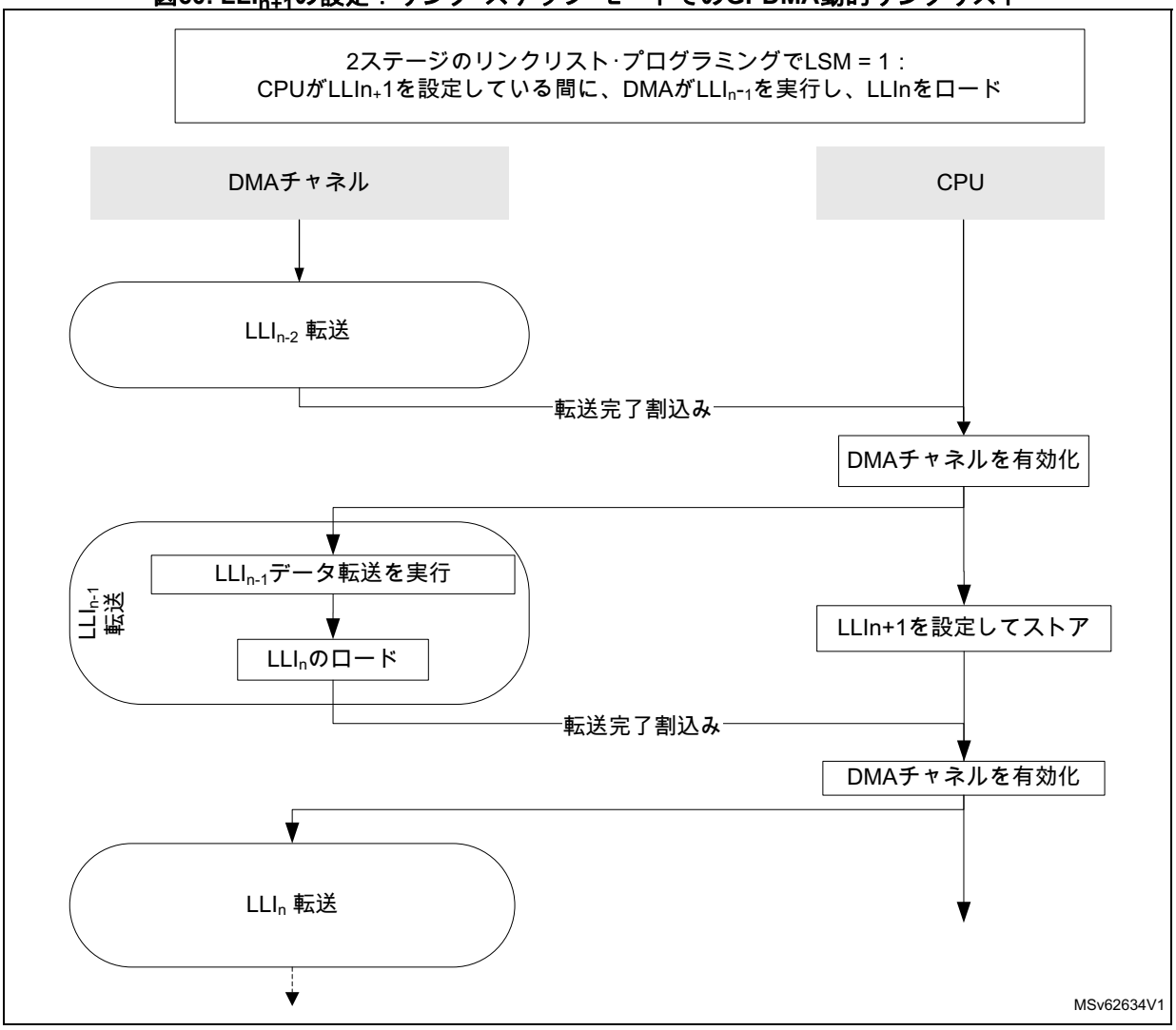
図59. リンク･ステップ･モード(GPDMA_CxCR.LSM = 1)でのGPDMAチャネル実行およびリンクリスト･プログラミング



リンク･ステップ･モードでのLLI$_{n+1}$の実行時追加

実行中に、ソフトウェアは下図に示すように、GPDMAがLLI$_{n-1}$からの転送を実行し、LLI$_n$をメモリからロードするまで、またはその後まで、LLI$_{n+1}$（および次のLLI）の処理を保留することができます。

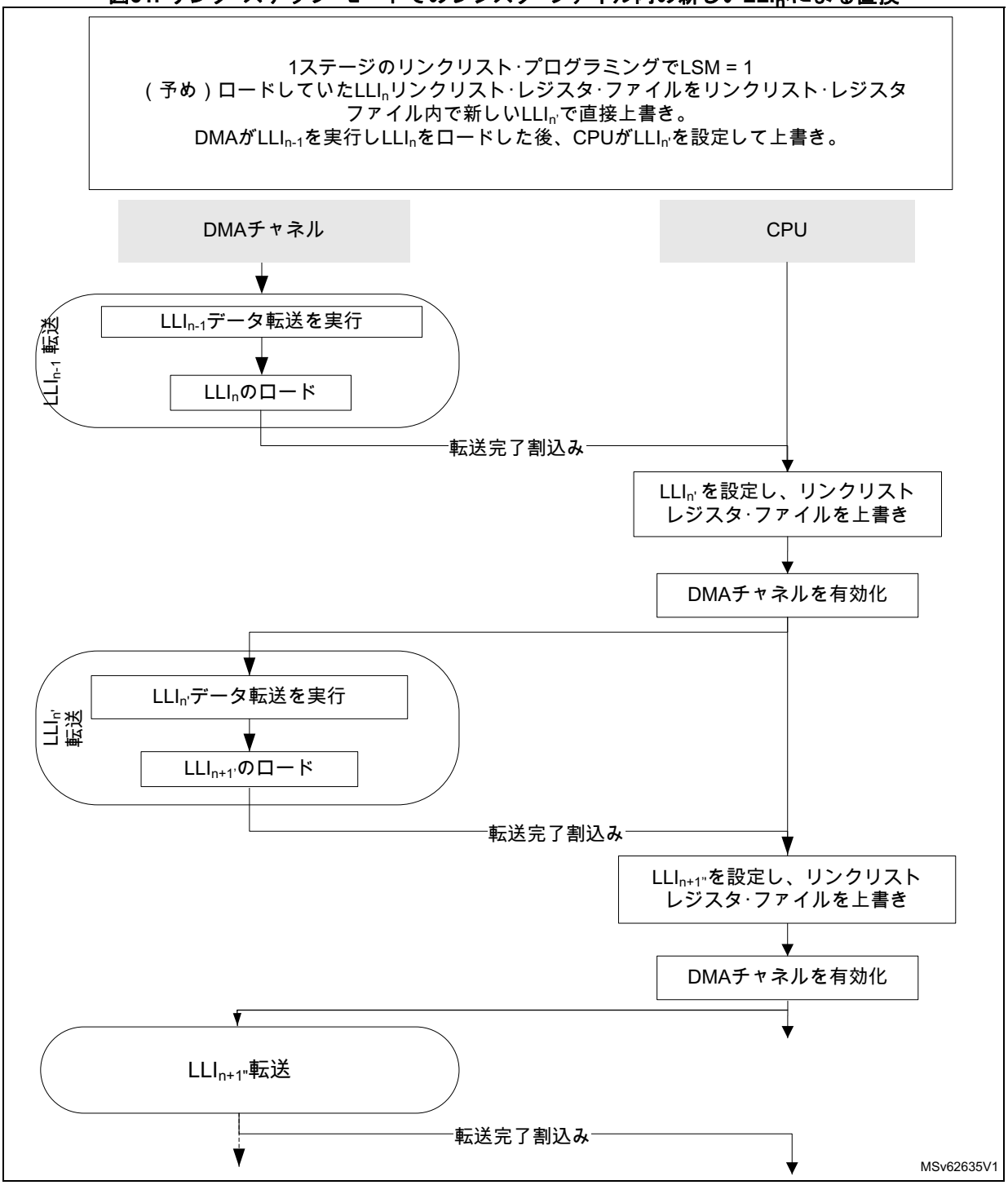
図60. LLI$_{n+1}$の設定：リンク･ステップ･モードでのGPDMA動的リンクリスト



リンク･ステップ･モードでのLLI$_n$と新しいLLI$_{n'}$との実行時置換（リンクリスト･レジスタ･ファイル内）

このリンク･ステップ･モードで、実行中に、下図に示すように、リンクリスト･レジスタ･ファイルを新しいLLI$_{n'}$で直接上書きすることによってGPDMAがLLI$_{n-1}$からの転送を実行し、前に処理されたLLI$_n$をメモリからロードした後に、ソフトウェアは新しいLLI$_{n'}$を設定し、挿入することができます。

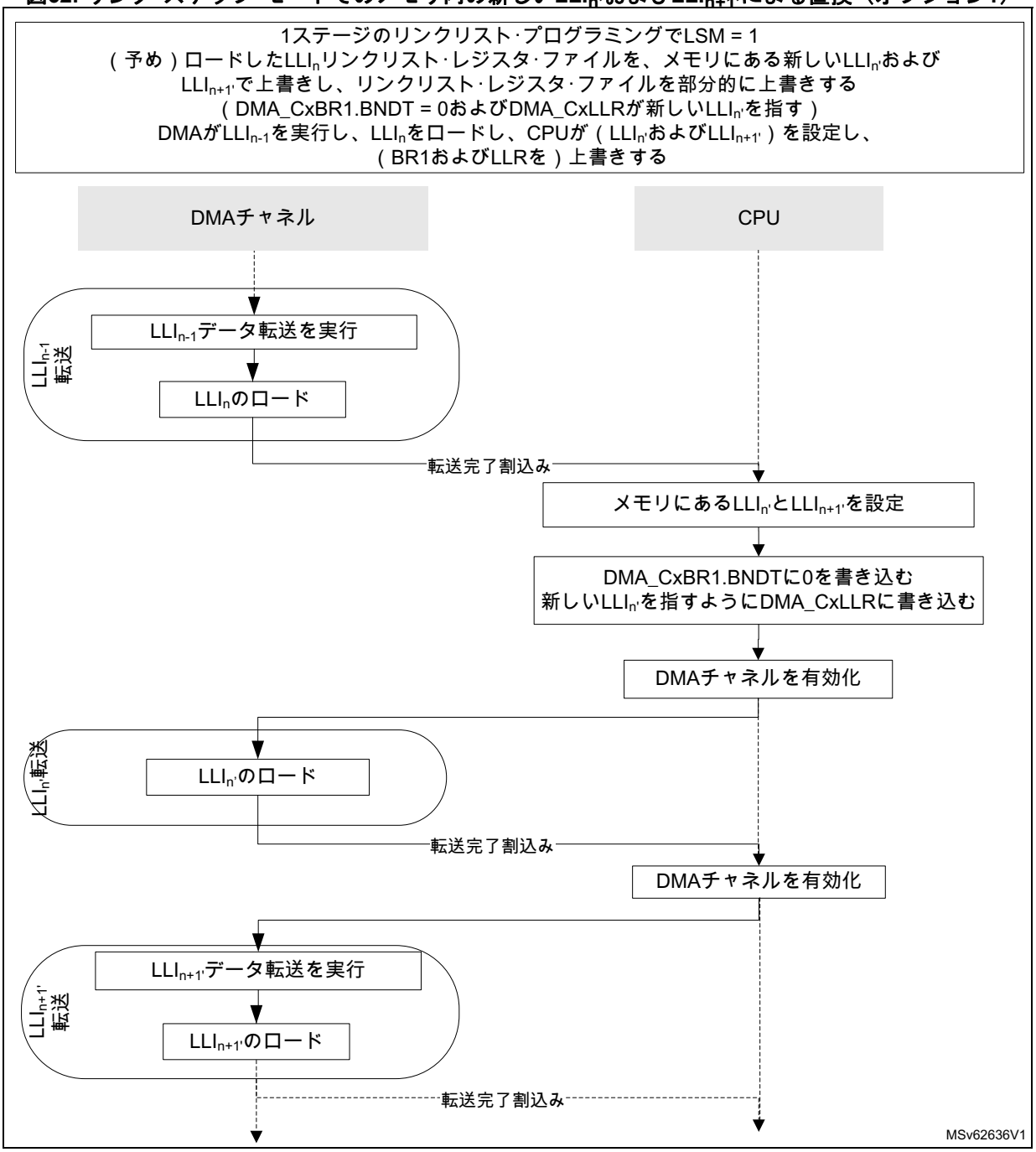
図61. リンク･ステップ･モードでのレジスタ･ファイル内の新しいLLI$_{n'}$による置換



リンク･ステップ･モードでのLLI$_n$と新しいLLI$_{n'}$との実行時置換（メモリ内）

ソフトウェアは、下図に示すように、リンクリスト･レジスタ･ファイルを部分的に上書きする（GPDMA_CxBR1.BNDT[15:0]をnullにし、GPDMA_CxLLRが新しいLLI$_{n'}$を指すようにする）ことにより、GPDMAがLLI$_{n-1}$から転送を実行し、前に処理したLLI$_n$をメモリからロードした後、メモリに新しいLLI$_{n'}$およびLLI$_{n+1'}$を設定し、挿入できます。

図62. リンク･ステップ･モードでのメモリ内の新しいLLI$_{n'}$およびLLI$_{n+1'}$による置換（オプション1）



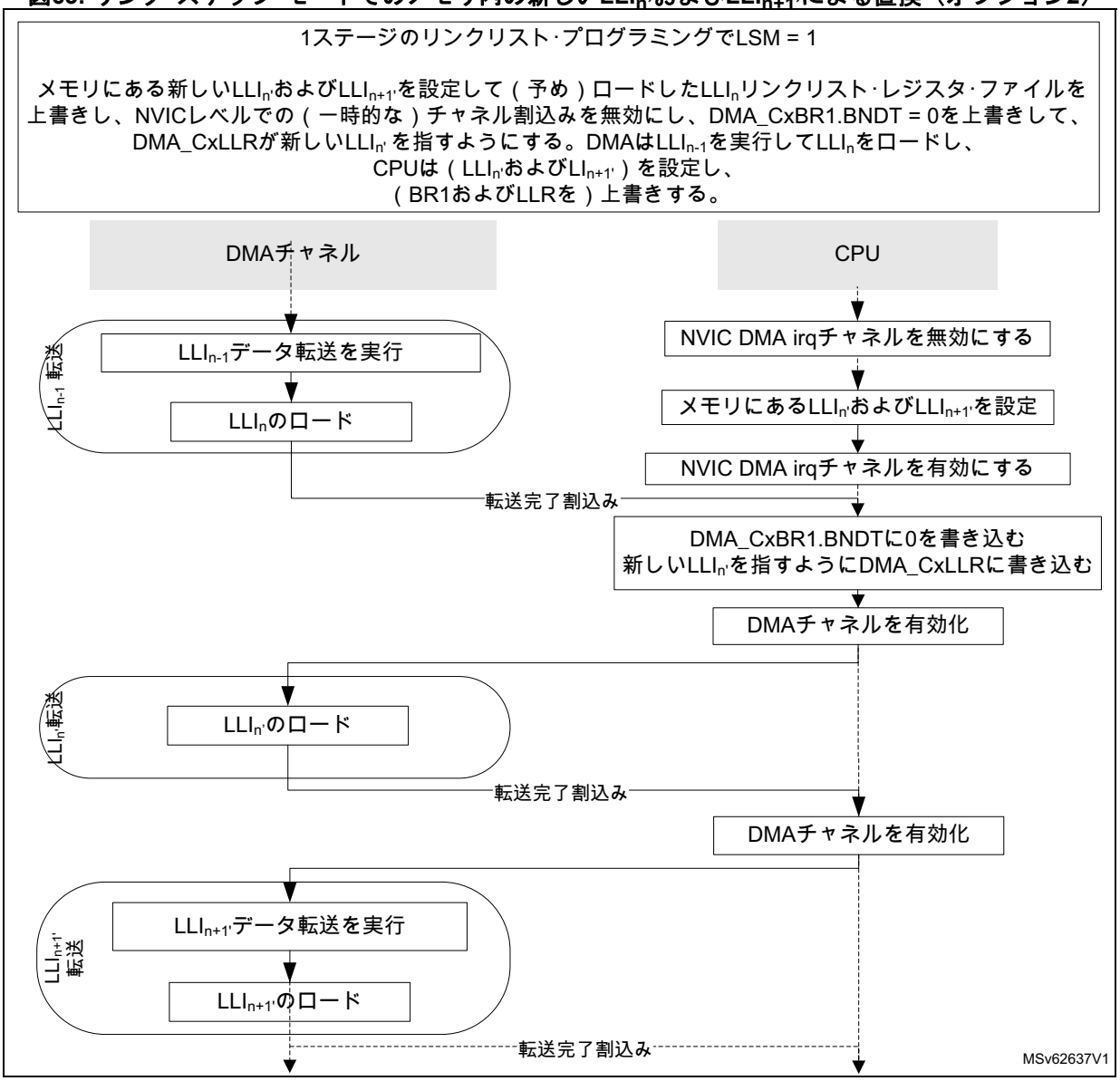
リンク･ステップ･モード（実行時）でのLLI$_n$を新しいLLI$_{n'}$への置き換え

他のソフトウェア実装が存在しますが、一方でGPDMAはLLI$_{n-1}$からの転送を実行して、前に（あるいはさらに前に）作成したLLI$_n$をメモリからロードします。ソフトウェアは次の項目を実行することができます。

1. 割込み操作による中断を防ぐためにNVICを無効にする。
2. 新しいLLI$_{n'}$とLLI$_{n+1'}$を設定する。
3. チャネル割込み（転送完了）通知のためにNVICを再び有効にする。

LLI$_{n-1}$割込みハンドラのソフトウェアはその後、GPDMA_CxBR1.BNDT[15:0]をnullに上書きすることと、GPDMA_CxLLRが下の図に示すように新しいLLI$_{n'}$を指すようにすることに限定されます。

図63. リンク･ステップ･モードでのメモリ内の新しいLLI$_{n'}$およびLLI$_{n+1'}$による置換（オプション2）



17.4.9 GPDMAチャネル状態およびリンクリスト･プログラミング

ソフトウェアは、チャネルが無効になっている時に（GPDMA_CxCR.EN = 0）チャネルを再設定し、実行モード（GPDMA_CxCR.LSM）を更新してrun-to-completionモードとリンク･ステップ･モードとを相互に切り替えることができます。

あらゆる実行モードにおいてソフトウェアは次の項目を行うことができます。

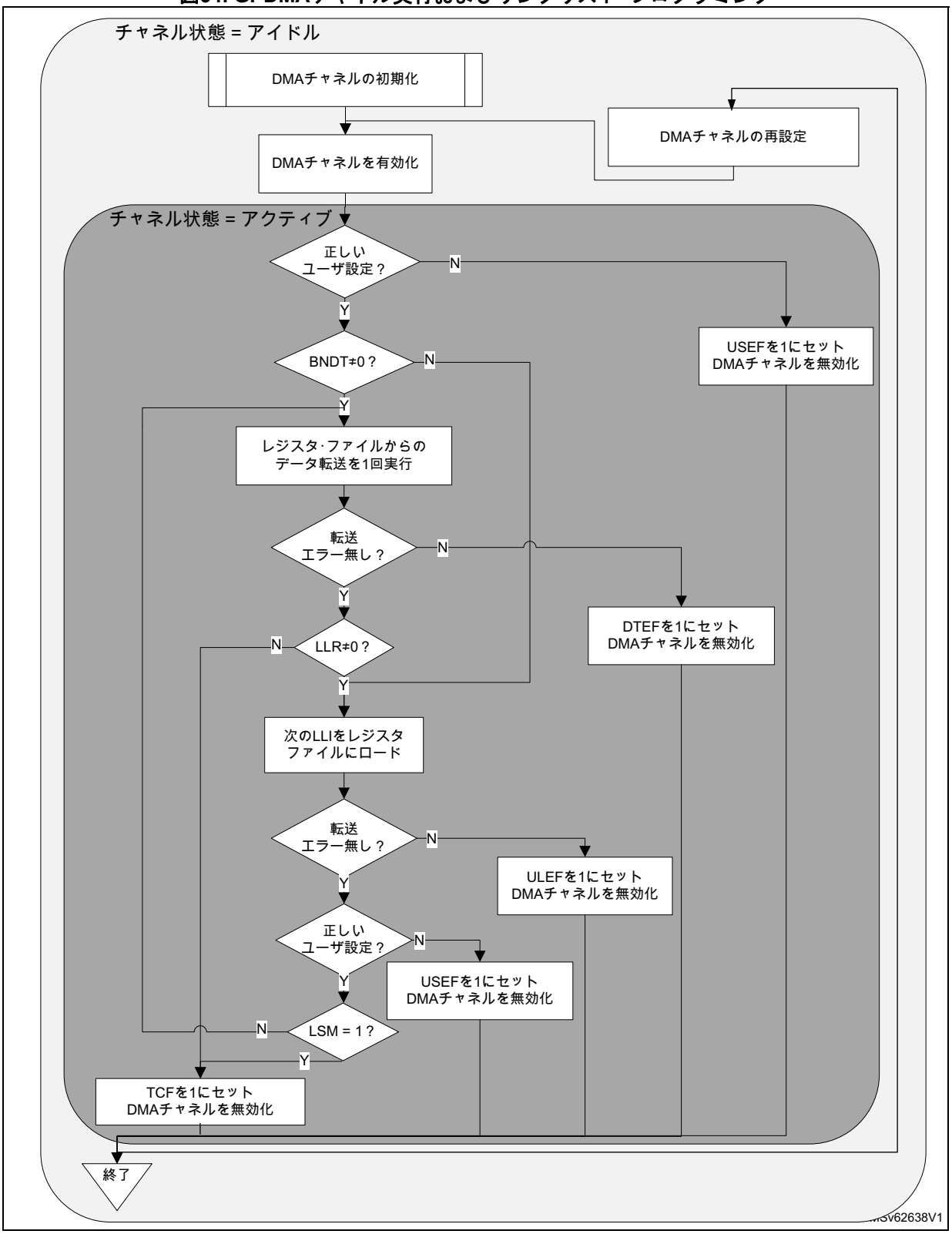
- このLLI$_{n+1}$がGPDMAチャネルによってロード／使用される前に、メモリ内でLLI$_{n+1}$を再プログラムして、このLLI$_{n+1}$により最終的にチャネルを完了（このLLI$_{n+1}$のGPDMA_CxLLRをクリア）します。
- チャネルをLSM更新によりアボートおよび再設定（セクション17.4.4を参照）

リンク･ステップ･モードでは、ソフトウェアがLLI$_{n-1}$の間、あらゆるLLIの各シングル実行後にLSMをクリアすることができます。

図64に実行モードに関係のない、全体の統合したGPDMAリンクリスト･プログラミングを示します。

注： **図64は、GPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、または最後のLLIデータ転送の完了時のみに立てることができます。run-to-completionモードではチャネル完了時のTCEM[1:0]の値に関わらず、ハードウェアは常にTCEF=1を設定し、そのチャネルを無効にします。リンク･ステップ･モードでは、LLIの各シングル実行後にチャネルが無効になり、TCEM[1:0]の値によりTCEFを立てるかどうかが変わります。**

図64. GPDMAチャネル実行およびリンクリスト･プログラミング



17.4.10 GPDMA FIFOを使用した転送

これはFIFOモードという1つの転送操作モードの説明です。FIFOを使用したFIFOベースの転送は複数存在します。あらゆるチャネルxは専用のFIFOにより実装され、そのサイズはdma_fifo_size[x]により定義されます（詳細はセクション17.3.1を参照）。

GPDMAバースト

もっとも単純なプログラム転送はGPDMAバーストです。

GPDMAバーストは転送元から受け取ったデータ･バースト、または転送先へ送信されたデータ･バーストです。転送元（転送先）バーストはGPDMA_CxTR1レジスタのSBL_1[5:0]（DBL_1[5:0]）フィールドにより定義されたバースト長、SDW_LOG2[1:0]（DDW_LOG2[1:0]）フィールドにより定義されたデータ幅でプログラムされます。

GPDMAバースト後の各データのアドレス指定モード（Beat（ビート）と呼ばれる）はGPDMA_CxTR1のSINCおよびDINCによりそれぞれ転送元、転送先に対して、固定アドレス指定、あるいは隣接データに対するインクリメントアドレス指定のどちらかが定義されます。

GPDMAソース／デスティネーション･バーストの開始および次のアドレス（GPDMA_CxSARおよびGPDMA_CxDARで定義される）は、それぞれのデータ幅に合っていなくてはなりません。

下の表にGPDMAバーストの主な特徴を列記します。

表139.プログラムされたGPDMAソース／デスティネーション･バースト

SDW_LOG2[1:0] DDW_LOG2[1:0]	データ幅（バイト）	SINC/DINC	SBL_1[5:0] DBL_1[5:0]	バースト長（データ／ビート）	次のデータ／ビートアドレス	次のバーストアドレス	バーストアドレス整合
00	1	0（固定）	n = 0から63(1)	n+1	+ 0	+ 0	1
01	2						2
10	4						4
00	1	1（連続的なインクリメント）			+ 1	+ (n + 1)	1
01	2				+ 2	+2×(n+1)	2
10	4				+ 4	+4×(n+1)	4
11	ユーザ設定禁止。USEFが生成されバーストが発行されません。						

1. S/DBL_1[5:0] = 0の場合、バースト長は1。この時、バーストはシングルとも呼ばれます。

上の表の次のバースト･アドレスは、プログラムされたソース／デスティネーション･バーストが一旦、完了したときに、GPDMA_CxSARまたはGPDMA_CxDARにより指定された次の転送元／転送先のデフォルト･アドレスです。このデフォルト値は固定／連続的インクリメント･アドレスを参照します。

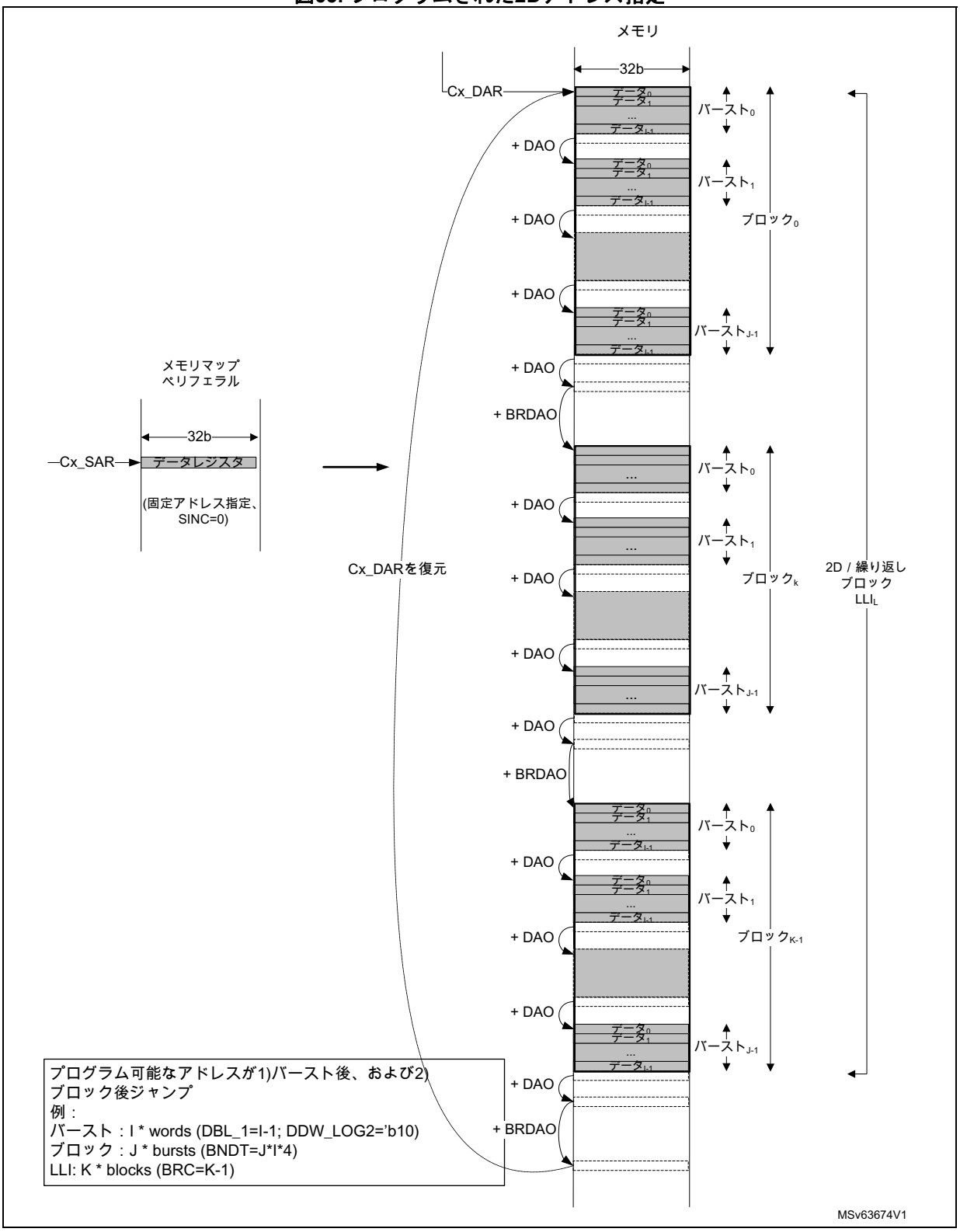
2Dアドレス指定ありのGPDMAバースト（チャネルx = 12~15）

チャネルに追加の2Dアドレス指定機能がある場合、このデフォルト値は2個のプログラムされたインクリメントまたはデクリメント･オフセットを考慮しない値を参照します。これら2つの追加オフセット（デフォルト値はnull）は次の場合に適用されます。

- それぞれ GPDMA_CxTR2.SAO[12:0]/DAO[12:0] および GPDMA_CxBR1.SDEC/DDEC で定義される、完了した各ソース／デスティネーション･バーストの後
- それぞれGPDMA_CxBR2.BRSAO[15:0]/BRDAO[15:0]および GPDMA_CxBR1.BRSDEC/BRDDECで定義される、完了した各ブロックの後

下に示す2D転送先バッファの図に示したとおり、各バースト完了後、最初にプログラムされたアドレス･ジャンプと各ブロックの後の2番目にプログラムされたアドレス･ジャンプで2D／繰返しブロックをアドレス指定することが可能です。

図65. プログラムされた2Dアドレス指定



GPDMA FIFOを使用した転送

FIFOモードの転送は一般的に2つのパイプライン化され分離したバースト転送からなります。

- 転送元からFIFOへのバーストの1つは、GPDMA_CxTR1SAPで定義されるとおり、割り当てられた転送元マスタ·ポートを経由します
- FIFOから転送先への1つのバーストは、GPDMA_CxTR1.DAPで定義されるとおり、割り当てられた転送先マスタ·ポートを経由します

GPDMAソース·バースト

リクエストされたFIFOへのソース·バースト転送は、割り当てられたポートを経由して、プログラムされたバースト·サイズに対する現在のFIFOレベルにより、できるだけ早くスケジュールすることが可能です（FIFOが転送元から新しいバーストを取得する準備ができたとき）。

FIFOレベル ≤ $2^{dma_fifo_size[x]}$ - (SBL_1[5:0] + 1) × $2^{SDW_LOG2[1:0]}$のとき。

ここで、

- FIFOレベルは現在のFIFOのフィル·レベル（バイト）です。
- $2^{dma_fifo_size[x]}$はチャネルxのFIFOサイズの半分（バイト）です（実装の詳細とdma_fifo_size[x]値はセクション17.3.1参照）。
- (SBL_1[5:0] + 1) × $2^{SDW_LOG2[1:0]}$はプログラムされたソース·バースト転送のサイズ（バイト）です。

チャネルの優先順位（GPDMA_CxCR.PRIO[1:0]）に基づき、この準備されたFIFOベースのソース転送は、内部で他のリクエストされ、アクティブなチャネルに対してアービトレーションされます。

GPDMAデスティネーション·バースト

リクエストされたFIFOからのデスティネーション·バースト転送は、割り当てられたポートを経由して、プログラムされたバースト·サイズに対する現在のFIFOレベルにより、できるだけ早くスケジュールすることが可能です（FIFOが転送先に1つの新しいバーストをプッシュする準備ができたとき）。

FIFOレベル ≥ (DBL_1[5:0] + 1) × $2^{DDW_LOG2[1:0]}$のとき。

ここで、

- FIFOレベルは現在のFIFOのフィル·レベル（バイト）です。
- (DBL_1[5:0] + 1) × $2^{DDW_LOG2[1:0]}$はプログラムされたデスティネーション·バースト転送のサイズ（バイト）です。

チャネルの優先順位に基づき、この準備されたFIFOベースのデスティネーション転送は、内部で他のリクエストされ、アクティブなチャネルに対してアービトレーションされます。

GPDMAバーストと転送元ブロックサイズ、1KBアドレス境界およびFIFOサイズ

プログラムされたソース／デスティネーションGPDMAバーストは、次のいずれかの条件を満足しない場合には、AHBバーストとしてそのまま実装されます。

- チャネルxのFIFOサイズの半分がプログラムされたソース／デスティネーション·バースト·サイズより小さい場合、プログラムされたソース／デスティネーションGPDMAバーストが、ユーザの一切の制約なく、各転送がFIFOサイズの半分以下の小さいサイズの一連のシングルまたはバースト転送で実装されます。
- 転送元のブロック·サイズ（GPDMA_CxBR1.BNDT[15:0]）が転送元のバースト·サイズの倍数ではなく、ソース·バーストのデータ幅（GPDMA_CxTR1.SDW_LOG2[1:0]）の倍数の場合、GPDMAは、ユーザの一切の制約なく、シングルまたは、より短いバーストへと変更、短縮を行い、転送元ブロックの厳密なサイズを転送します。

- ソース／デスティネーション･バースト転送が AHB 転送の 1KB のアドレス境界を越えた場合、GPDMAはプログラムされたバーストをシングルまたは、より短いバーストへと変更、短縮を行い、一切のユーザによる制約なく、AHBプロトコルに準拠するようにします。
- AHB転送におけるソース／デスティネーション･バースト長が16を超えた場合、GPDMAはプログラムされたバーストをより短いシングルまたはバーストに変更、短縮を行い、一切のユーザの制約なく、AHBプロトコルに準拠するようにします。

いずれの場合もGPDMAは転送元／転送先データ（とアドレス）の整合性を一切のユーザの制約なく確保します。（GPDMA_CxSRでソフトウェア読取り可能な）現在のFIFOレベルは、実効的な転送サイズと比較して更新され、変更された可能性のある、AHBの各シングルまたはバースト転送間のアービトレーションをGPDMAが再度行います。

各シングル、またはプログラムされたバーストより小さいバースト･サイズのバースト転送は、他のリクエストされた有効なチャネルに対し、チャネルの優先順位に基づいて内部でアービトレーションされます。

注： **リンクリスト･モードでのリンクリスト･パラメータの更新に関係したメモリから内部 GPDMA レジスタへのGPDMA読取り転送は、GPDMA_CxCR.LAPによるプログラムに従って、リンクが割り当てられたポートでスケジュールされます。**

GPDMAデータ･ハンドリング：バイト･ベースの再整列、パッキング／アンパッキング、パディング／切捨て、符号拡張、左／右詰め

データ･ハンドリングはGPDMA_CxTR1により制御されます。プログラムされたバーストの転送元／転送先データ幅は、SDW_LOG2[21:0]およびDDW_LOG2[1:0]フィールドに従って、バイト、ハーフワードまたはワードです（表140を参照）。

ユーザは、転送データの転送元と転送先の間でのデータ･ハンドリングを設定することができます。具体的には、プログラムされたデータ･ハンドリングは次のように整然と実行されます。

1. バイトベースの転送元、再整列
   - SBX = 1で転送元のデータ幅がワードの場合、各転送元データ･ワードの中央のアンアライン ド･ハーフワードの2バイトを交換します。
2. 転送先データ幅が転送元データ幅と異なる場合、PAM[1:0] に従った、パッキングやアンパッキング、パディング、または切捨てによるデータ幅の変換。
   - 転送先データ幅>転送元データ幅の場合、ポストSBX転送元データは、右寄せで0sでパディングするか、転送先データ幅まで符号拡張するか、あるいは、FIFOキューに入れて転送先データ幅までパッキングします。
   - 転送先のデータ幅<転送元のデータ幅の場合、ポストSBXデータは、右詰めにし、転送先のデータ幅に合わせて左を切り捨てるか、FIFOをキューイングし、アンパックし、転送先のデータ幅に合わせてストリームダウンします。
3. バイト･ベースの転送先の再整列。
   - DBX = 1で転送先データ幅がバイトではない場合、整列したポストPAM[1:0]ハーフワード内の2バイトを交換します。
   - DHX = 1で転送先データ幅がバイトでもハーフワードでもない場合、整列したポストPAM[1:0]ワード内の2つの整列したハーフワードを交換します。

注： **0sのパディングによる左詰めは、0sのパディングによる右詰めと、転送先のバイト･ベースの再整列の両方をプログラミングすることで行うことができます。**

下の表は転送元から転送先への考えられえるデータ･ハンドリングを列記したものです。

表140.プログラムされたデータ･ハンドリング

SDW_LOG2 [1:0]	転送元データ	転送元データストリーム(1)	SBX	DDW_LOG2 [1:0]	転送先データ	PAM[1:0].(2)	DBX	DHX	転送先データ･ストリーム(1)
00	バイト	B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0	x	00	バイト	xx	x	x	B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0
				01	ハーフワード	00 (RA, 0P)	0		$0B_3$、$0B_2$、$0B_1$、$0B_0$
							1		B_30、B_20、B_10、B_00
						01 (RA, SE)	0		SB_3、SB_2、SB_1、SB_0
							1		B_3S、B_2S、B_1S、B_0S
						1x (PACK)	0		B_7B_6、B_5B_4、B_3B_2、B_1B_0
							1		B_6B_7、B_4B_5、B_2B_3、B_0B_1
				10	ワード	00 (RA, 0P)	0	0	$000B_1$、$000B_0$
							1		$00B_10$、$00B_00$
							0	1	$0B_100$、$0B_000$
							1		B_1000、B_0000
						01 (RA, SE)	0	0	$SSSB_1$、$SSSB_0$
							1		SSB_1S、SSB_0S
							0	1	SB_1SS、SB_0SS
							1		B_1SSS、B_0SSS
						1x (PACK)	0	0	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$
							1		$B_6B_7B_4B_5$、$B_2B_3B_0B_1$
							0	1	$B_5B_4B_7B_6$、$B_1B_0B_3B_2$
							1		$B_4B_5B_6B_7$、$B_0B_1B_2B_3$
01	ハーフワード	B_7B_6、B_5B_4、B_3B_2、B_1B_0		00	バイト	00 (RA, LT)	x	x	B_6、B_4、B_2、B_0
						01 (LA, RT)			B_7、B_5、B_3、B_1
						1x (UNPACK)			B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0

表140.プログラムされたデータ・ハンドリング （続き）

SDW_LOG2 [1:0]	転送元データ	転送元データ・ストリーム(1)	SBX	DDW_LOG2 [1:0]	転送先データ	PAM[1:0].(2)	DBX	DHX	転送先データ・ストリーム(1)
01	ハーフワード	B_7B_6、B_5B_4、B_3B_2、B_1B_0	x	01	ハーフワード	xx	0	x	B_7B_6、B_5B_4、B_3B_2、B_1B_0
							1		B_6B_7、B_4B_5、B_2B_3、B_0B_1
				10	ワード	00 (RA, 0P)	0	0	$00B_3B_2$、$00B_1B_0$
							1		$00B_2B_3$、$00B_0B_1$
							0	1	B_3B_200、B_1B_000
							1		B_2B_300、B_0B_100
						01 (RA, SE)	0	0	SSB_3B_2、SSB_1B_0
							1		SSB_2B_3、SSB_0B_1
							0	1	B_3B_2SS、B_1B_0SS
							1		B_2B_3SS、B_0B_1SS
						1x (PACK)	0	0	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$
							1		$B_6B_7B_4B_5$、$B_2B_3B_0B_1$
							0	1	$B_5B_4B_7B_6$、$B_1B_0B_3B_2$
							1		$B_4B_5B_6B_7$、$B_0B_1B_2B_3$
10	ワード	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$	0	00	バイト	00 (RA, LT)	x	x	B_{12}、B_8、B_4、B_0
						01 (LA, RT)			B_{15}、B_{11}、B_7、B_3
						10 (UNPACK)			B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0
				01	ハーフワード	00 (RA, LT)	0		B_5B_4、B_1B_0
							1		B_4B_5、B_0B_1
						01 (LA, RT)	0		B_7B_6、B_3B_2
							1		B_6B_7、B_2B_3
						1x (UNPACK)	0		B_7B_6、B_5B_4、B_3B_2、B_1B_0
							1		B_6B_7、B_4B_5、B_2B_3、B_0B_1

表140.プログラムされたデータ･ハンドリング （続き）

SDW_LOG2 [1:0]	転送元データ	転送元データストリーム(1)	SBX	DDW_LOG2 [1:0]	転送先データ	PAM[1:0].(2)	DBX	DHX	転送先データ･ストリーム(1)
10	ワード	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$	0	10	ワード	xx	0	0	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$
							1		$B_6B_7B_4B_5$、$B_2B_3B_0B_1$
							0	1	$B_5B_4B_7B_6$、$B_1B_0B_3B_2$
							1		$B_4B_5B_6B_7$、$B_0B_1B_2B_3$
			1	00	バイト	00 (RA, LT)	x	x	B_{12}、B_8、B_4、B_0
						01 (LA, RT)			B_{15}、B_{11}、B_7、B_3
						1x (UNPACK)			B_7、B_5、B_6、B_4、B_3、B_1、B_2、B_0
				01	ハーフワード	00 (RA, LT)	0		B_6B_4、B_2B_0
							1		B_4B_6、B_0B_2
						01 (LA, RT)	0		B_7B_5、B_3B_1
							1		B_5B_7、B_1B_3
						1x (UNPACK)	0		B_7B_5、B_6B_4、B_3B_1、B_2B_0
							1		B_5B_7、B_4B_6、B_1B_3、B_0B_2
				10	ワード	xx	0	0	$B_7B_5B_6B_4$、$B_3B_1B_2B_0$
							1		$B_5B_7B_4B_6$、$B_1B_3B_0B_2$
							0	1	$B_6B_4B_7B_5$、$B_2B_0B_3B_1$
							1		$B_4B_6B_5B_7$、$B_0B_2B_1B_3$

1. データ･ストリームを最低インデックス（B_0）のバイトから始めて、タイミングで整列します。
2. RA = 右詰め、LA = 左詰め、RT = 右切捨て、LT = 左切捨て、0P = 転送先データ幅までゼロ･ビット･パッディング、SE=転送先データ幅まで符号ビット拡張。

17.4.11 GPDMA転送リクエストとアービトレーション

GPDMA転送リクエスト

GPDMA_CxTR2の定義に従って、プログラムされたGPDMAデータ転送を次のいずれかの方法でリクエストします。

- 制御ビットSWREQ = 1の場合はソフトウェア･リクエスト：一般的に、メモリマップされたアドレスから別のメモリマップされたアドレスへのデータ転送のためにCPUによって使用されます（メモリ間、GPIOとメモリ間）
- SWREQ = 0の場合はペリフェラルからの入力ハードウェア･リクエスト：GPDMAハードウェアペリフェラル･リクエストの選択は、REQSEL[6:0]フィールドを使用して行われます（セクション17.3.3を参照）。選択したハードウェア･リクエストは次のいずれかです。
  - GPDMAモードで設定されたペリフェラルからのハードウェア･リクエスト（ペリフェラル･データ･レジスタとメモリの相互転送）
  - メモリから制御レジスタを更新するための、ペリフェラルからのハードウェア･リクエスト
  - メモリに転送されたステータス･レジスタの読出しのための、ペリフェラルからのハードウェア･リクエスト

注意： **所定の時間にこのリクエストがペリフェラルによってアサートされ、各チャネルでこのリクエストされたデータ転送が可能な状態である場合、ユーザーは同じ入力ハードウェアペリフェラルGPDMAリクエストをGPDMA_CxTR.REQSEL[6:0]を介して2つの異なるチャネルに割り当てることはできません。ユーザ設定エラーのレポートはありません。**

アービトレーションのためのGPDMA転送リクエスト

割り当てられたマスタ･ポート（GPDMA_CxTR1.SAP）を使ってスケジュールされる、準備の整った（転送元アドレスからFIFOへの）FIFOベースのGPDMAソース･シングル／バースト転送は、他の同時にリクエストされた、同じマスタ･ポートへの GPDMA 転送に対して、チャネルの優先順位（GPDMA_CxCR.PRIO[1:0]）に基づいてアービトレーションされます。

割り当てられたマスタ･ポート（GPDMA_CxTR1.DAP）を使ってスケジュールされる、準備の整った（FIFOから転送先アドレスへの）FIFOベースのGPDMAデスティネーション･シングル／バースト転送は、他の同時にリクエストされた、同じマスタ･ポートへのGPDMA転送に対して、チャネルの優先順位（GPDMA_CxCR.PRIO[1:0]）に基づいてアービトレーションされます。

GPDMAにリクエストされ、アービトレーションされたリンク転送は、メモリのリンクリスト･データ構造からリンクリスト･レジスタ（GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDARまたはGPDMA_CxLLRに加えGPDMA_CxTR3、GPDMA_CxBR2）の1つへの、1回の32bit読出しにより構成されます。メモリからの各32bit読出しは、データ転送と同じチャネルの優先順位でアービトレーションされ、割り当てられたマスタ･ポート（GPDMA_CxCR.LAP）でスケジューリングされます。

メモリ間転送のためにソフトウェア･リクエストでプログラムされたデータ転送（GPDMA_CxTR2.SWREQ = 1）が何であれ、あるいはメモリとペリフェラル間のハードウェアがリクエストした転送（GPDMA_CxTR2.SWREQ = 0）であれ、または、そのハードウェア･リクエストの種類が何であれ、許可された各シングル／バースト転送後に再アービトレーションが行われます。

転送先のペリフェラルでハードウェア･リクエストがプログラムされると（GPDMA_CxTR2.SWREQ = 0およびGPDMA_CxTR2.DREQ = 1）、最初のメモリ読出し（おそらく2D／繰返し）ブロック（最初のレディになっているFIFOベースのソース･バースト･リクエスト）は、対応する選択されたハードウェア･リクエストの発生によりゲートされます。この最初のメモリへの読出しリクエストは、早い段階では（ブロック転送が有効になって実行可能になって直ぐには）アービタでは考慮されません。

GPDMAアービトレーション

GPDMAアービトレーションは4グレードが割り当てられたチャネルの優先順位（GPDMA_CxCR.PRIO[1:0]）から行われます。図66に図示したアービトレーション･ポリシーは次のように定義されます。

- 優先順位の高い1つのトラフィック･クラス（キュー3）。優先順位3が割り当てられたタイミングが重要なチャネル専用。

  このトラフィック･クラスは、他のあらゆる低優先順位のトラフィック･クラスに対して、固定優先順位アービトレーションで許可されます。このクラス内では、リクエストされたシングル／バースト転送はラウンド･ロビン（持ち回り）でアービトレーションされます。

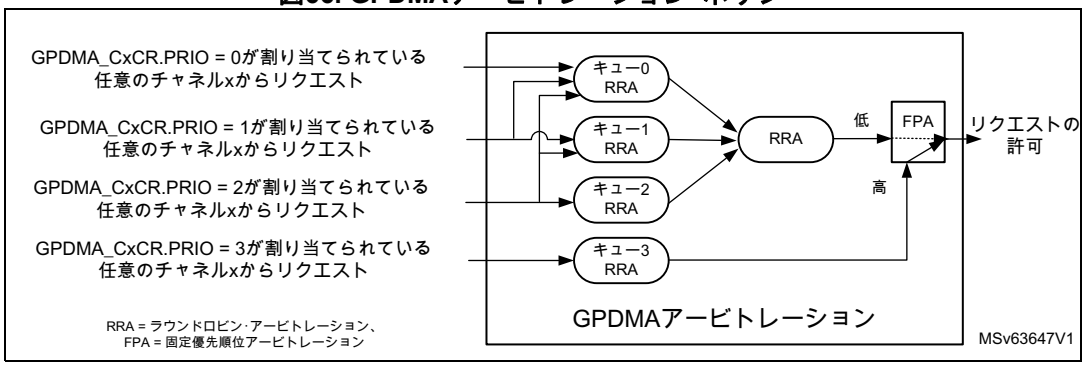
- 優先順位の低い3つのトラフィック･クラス（キュー 0、1、2）。タイミングの制約が緩い、優先順位0、1、2のチャネル。

  このクラスの、リクエストされたそれぞれのシングル／バースト転送は、プログラムされた優先順位で単純な重み付けを行い、ラウンド･ロビン（持ち回り）でアービトレーションされます：

  – 優先順位0のリクエストはキュー 0に割り当てられます。
  – 優先順位1のリクエストはキュー 0と1に割り当て、複製されます。
  – 優先順位2のリクエストはキュー 0、1、2に割り当て、複製されます。

– キュー0、1、2は、同時リクエストがある場合、その有効な入力リクエストをラウンド･ロビン（持ち回り）方式で平等に許可します。
– さらに、低優先順位のトラフィックの第2ステージがあり、そこにはキュー0、1、2から同時に選択されたリクエスト間で公平にオルタネートするラウンド･ロビン（持ち回り）のアービタがあります。

図66. GPDMAアービトレーション･ポリシー



GPDMAアービトレーションとバンド幅

このアービトレーション･ポリシーでは、次の事項が保証されます。

- 同じポリシーのリクエストの最大バンド幅は等しい
- タイミングが重要なリクエスト（優先順位3）の予約済みのバンド幅（B_{Q3}と表記）
- 様々な、低い優先順位のリクエスト（優先順位0、1、2の間）に対する残りの重み付けバンド幅。

次の2つの例では、重み付けラウンド･ロビン（持ち回り）アービトレーションがプログラムされた優先順位で実施されていることがよく分かります。

- **例1**：2つのタイミングの制約が緩いGPDMAリクエスト、req0とreq1の基本アプリケーション。次のようなプログラミングの可能性があります。
  – 同じ優先順位が割り当てられている場合、アービタによりreq0に割り当てられるバンド幅（B_{req0}）はreq1に割り当てられるバンド幅（B_{req1}）と**同じ**です。
    $B_{req0} = B_{req1} = 1/2 \times (1 - B_{Q3})$
  – req0が優先順位0に、req1が優先順位1に割り当てられると、req0に割り当てられるバンド幅（B_{P0}）はreq1に割り当てられるバンド幅（B_{P1}）の**3分の1**になります。
    $B_{req0} = B_{P0} = 1/2 \times 1/2 \times (1 - B_{Q3}) = 1/4 \times (1 - B_{Q3})$
    $B_{req1} = B_{P1} = (1/2 + 1) \times 1/3 \times (1 - B_{Q3}) = 3/4 \times (1 - B_{Q3})$
  – req0が優先順位0に、req1が優先順位2に割り当てられると、req0に割り当てられるバンド幅（B_{P0}）はreq1に割り当てられるバンド幅（B_{P2}）の**5分の1**になります。
    $B_{req0} = B_{P0} = 1/2 \times 1/3 \times (1 - B_{Q3}) = 1/6 \times (1 - B_{Q3})$
    $B_{req1} = B_{P2} = (1/2 + 1 + 1) \times 1/3 \times (1 - B_{Q3}) = 5/6 \times (1 - B_{Q3})$

  上で計算したバンド幅の計算は理論的な入力リクエストに基づいており、あらゆるGPDMAクロック･サイクルで常に有効です。アービタから計算されたこのバンド幅は、アプリケーションにより得られたリクエストの頻度により重み付けする必要があり、これは、常に有効とは限らず、あるGPDMAクライアント（例えば400kHzでI2C）と別のクライアント（1kHzのPWM）とでは、上で示した3倍や5倍とは大きく異なる場合があります。

- **例2**：ユーザが同じnullではないGPDMAリクエストのN数を、すべてのタイミングの制約が緩い優先順位0、1、2に配布するアプリケーション。このときのバンド幅の計算は次のとおりです。
  - 優先順位0の一連のリクエストに対して割り当てたバンド幅（B_{P0}）は、
    $B_{P0} = 1/3 \times 1/3 \times (1 - B_{Q3}) = 1/9 \times (1 - B_{Q3})$
  - 優先順位1の一連のリクエストに対して割り当てたバンド幅（B_{P1}）は、
    $B_{P1} = (1/3 + 1/2) \times 1/3 \times (1 - B_{Q3}) = 5/18 \times (1 - B_{Q3})$
  - 優先順位2の一連のリクエストに対して割り当てたバンド幅（B_{P2}）は、
    $B_{P2} = (1/3 + 1/2 + 1) \times 1/3 \times (1 - B_{Q3}) = 11/18 \times (1 - B_{Q3})$
  - その優先順位Pi（i = 0から2）のN個のリクエストの中のnリクエストに割り当てるバンド幅（B_n）は$B_n = 1/N \times B_{Pi}$
  - 優先順位0iのリクエストnに割り当てたバンド幅（$B_{n, Pi}$）は、
    $B_{n, P0} = 1/N \times 1/9 \times (1 - B_{Q3})$
    $B_{n, P1} = 1/N \times 5/18 \times (1 - B_{Q3})$
    $B_{n、P2} = 1/N \times 11/18 \times (1 - B_{Q3})$

  この例では、マスタ･ポート･バスのバンド幅がタイミングが重要なキュー3により完全には消費されていない場合、残りのバンド幅は、優先順位0のリクエストに割り当てられているバンド幅が優先順位1のリクエストに割り当てられているバンド幅の2.5分の1、優先順位2のリクエストに割り当てられているバンド幅に対しては5.5分の1になるように割り当てられます。

より一般的には、次のリクエストが存在することを想定します。

- I個のリクエスト（I ≥ 0）を優先順位0に割り当て。
  I > 0の場合、これらのリクエストはi = 0からI - 1で記録されます。
- J個のリクエスト（J ≥ 0）を優先順位1に割り当て。
  J > 0の場合、これらのリクエストはj = 0からJ - 1で記録されます。
- K個のリクエスト（K ≥ 0）を優先順位2に割り当て。
  これらのリクエストはk = 0からK - 1で記録されます。
- L個のリクエスト（L ≥ 0）を優先順位3に割り当て。
  L > 0場合、これらのリクエストはI = 0からL - 1で記録されます。

B_{Q3}はタイミングが重要なリクエストのために予約されているバンド幅なので、優先順位3の各リクエストLのバンド幅は、

- L > 0の場合、$B_I = B_{Q3}/ L$（その他の場合は、$B_I = 0$）

タイミングが重要ではないキューのそれぞれのバンド幅は次のとおりです。

- $B_{Q0} = 1/3 \times (1 - B_{Q3})$
- $B_{Q1} = 1/3 \times (1 - B_{Q3})$
- $B_{Q2} = 1/3 \times (1 - B_{Q3})$

優先順位0の一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P0} = I / (I + J + K) \times B_{Q0}$

優先順位0の各リクエストiに対するバンド幅は次のとおりです。

- L > 0の場合、$B_i = B_{P0} / I$（その他の場合は$B_{P0} = 0$）

優先順位1でキュー 0にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P1,Q0} = J / (I + J + K) \times B_{Q0}$

優先順位1でキュー 1にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P1,Q1} = J / (J + K) \times B_{Q1}$

優先順位1の一連のリクエストに対するバンド幅の合計は次のとおりです。

- $B_{P1} = B_{P1,Q0} + B_{P1,Q1}$

優先順位1のリクエストjに対するバンド幅は次のとおりです。

- J > 0の場合、$B_j = B_{P1} / J$（その他の場合は$B_j = 0$）

優先順位2でキュー 0にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q0} = K / (I + J + K) \times B_{Q0}$

優先順位2でキュー 1にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q1} = K / (J + K) \times B_{Q1}$

優先順位2でキュー 2にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q2} = B_{Q2}$

優先順位2の一連のリクエストに対するバンド幅の合計は次のとおりです。

- $B_{P2} = B_{P2,Q0} + B_{P2,Q1} + B_{P2,Q2}$

優先順位2のリクエストkに対するバンド幅は次のとおりです。

- $B_k = B_{P2} / K$（一般的にはK > 0）

そのため、タイミングが重要ではないリクエストi、j、kに割り当てられる残留バンド幅の最大値は最終的に次のようになります。

- 一般的な場合（優先順位2のリクエストkが最低1つある場合（K > 0））。
  - $B_i = 1/I \times 1/3 \times I/(I + J + K) \times (1 - B_{Q3})$
  - $B_j = 1/J \times 1/3 \times [J/(I + J + K) + J/(J + K)] \times (1 - B_{Q3})$
  - $B_k = 1/K \times 1/3 \times [K/(I + J + K) + K/(J + K) + 1] \times (1 - B_{Q3})$
- 特別な場合（優先順位2のリクエストkがない場合（K = 0））。
  - $B_i = 1/I \times 1/2 \times I/(I + J) \times (1 - B_{Q3})$
  - $B_j = 1/J \times 1/2 \times [J/(I + J) + 1] \times (1 - B_{Q3})$

結果としてGPDMAアービタは、各キューや、より一般的には各リクエスト／チャネルに対するプログラム可能な重み付けバンド幅のリミッターとして使用することができます。重み付けの差はプログラムされたチャネルの優先順位から単純に与えられることになるのです。

17.4.12 GPDMAトリガ転送

プログラムされたGPDMA転送は、GPDMA_CxTR2.TRIGPOL[1:0]およびGPDMA_CxTR2.TRIGSEL[6:0]での定義に従い、選択した入力トリガイベントの立上がり／立下がりエッジによってトリガできます（トリガの選択については、セクション17.3.5を参照）。

GPDMA_CxTR2.TRIGM[1:0]のトリガ･モードで定義されたトリガ転送は、LLIのデータ転送レベルとすることができ、ブロックの最初のバースト読出し、チャネルx（x = 12 ~ 15）の2D／繰返しブロックの最初のバースト読出し、あるいはプログラムされた各シングル読出しの条件付けを行います。トリガ･モードはLLIリンク転送の条件付けをプログラムをすることも可能です（詳細はGPDMA_CxTR2のTRIGM[1:0]参照）。

トリガ･ヒットのメモリ書込みとトリガ･オーバーラン･フラグの生成

チャネルxに対するトリガのGPDMA監視が開始されるのは、新しいアクティブなトリガ設定、つまり選択したトリガの立上がり／立下がりエッジ（それぞれ、TRIGPOL[1:0] = 01またはTRIGPOL[1:0] = 10）によりそのチャネルが有効化／ロードされるときです。

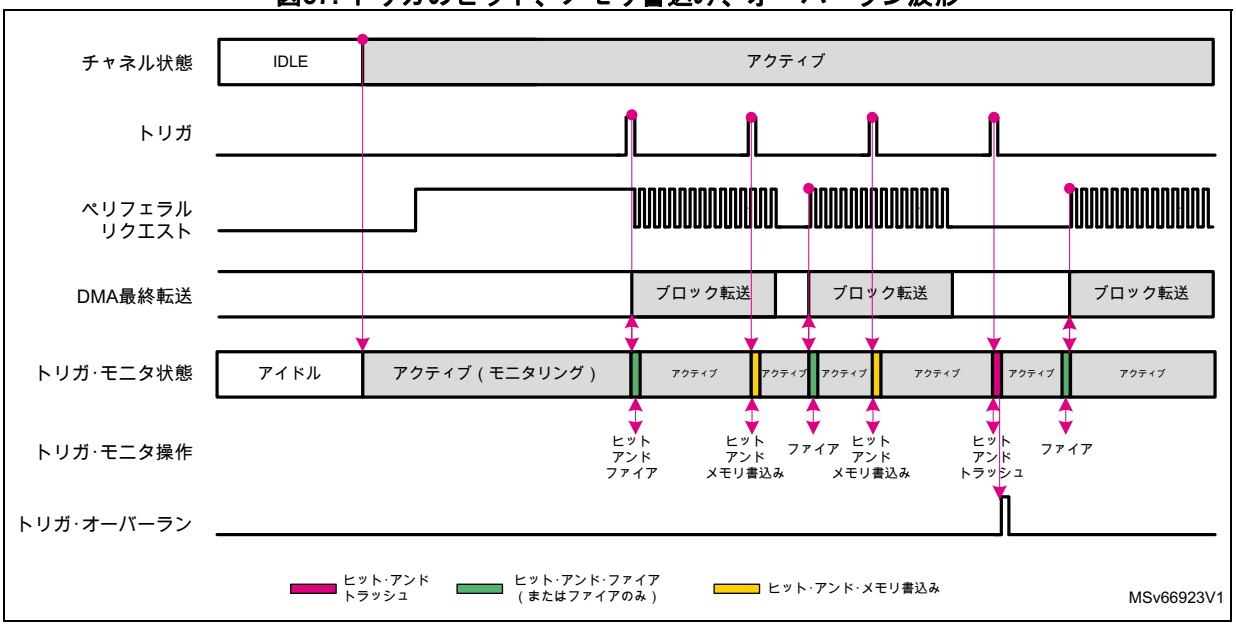
このトリガのモニタリングは、トリガされた完了していない（データまたはリンクの）転送を行っている間、アクティブなままとなります。新しいトリガが検出された場合、定義された立上がり／立下がりエッジとTRIGSEL[6:0]が変更されず、そのチャネルが有効である限り、このヒットは次の転送を許可するために内部的にメモリに書き込まれます。

TRIGSEL[6:0]またはいずれかのTRIGPOL[1:0]の新しい値でGPDMA_CxTR2を更新する次のLLI_{n+1}の転送が、モニタリングをリセットし、前に定義したLLI_nトリガのメモリ書込みされたヒットを破棄します。

注意： **最初の新しいトリガhit_{n+1}をメモリに書き込んだ後、別のトリガhit_{n+2}を検出し、hit_nにトリガされた転送が完了していない場合、hit_{n+2}は失われメモリに書き込まれません。トリガーオーバーランフラグが報告され(GPDMA_CxSR.TOF = 1)、有効な場合は割込みが生成されます(ifGPDMA_CxCR.TOIE = 1)トリガのオーバーランでハードウェアにより､チャネルが自動的に無効になることはありません。**

下図に、ブロックレベルのトリガモードと立上がりエッジのトリガの極性の設定例でのトリガのヒット、メモリ書込み、およびオーバーランを示します。

図67. トリガのヒット、メモリ書込み、オーバーラン波形



注： **ユーザは、異なるチャネルに同じ入力トリガ･イベントを割り当てることができます。これは、ブロードキャスト･トリガ･イベントに異なるチャネルをトリガするために使用可能です。**

17.4.13 リンクリスト･プログラミングでのGPDMAサーキュラ･バッファリング

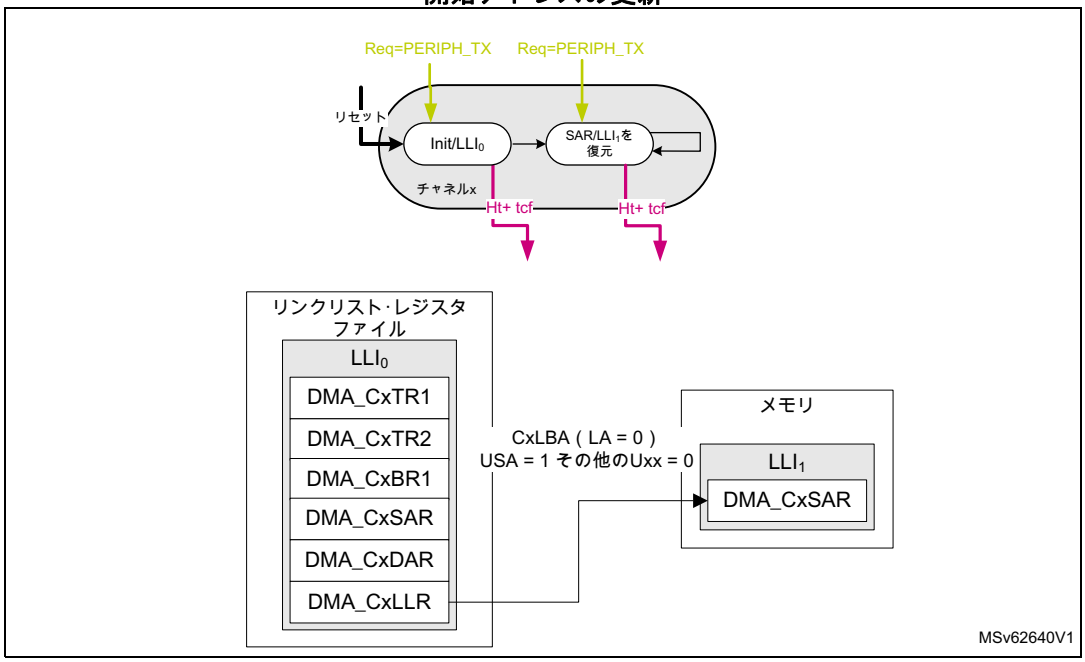
リニア･アドレス指定チャネルによる、メモリとペリフェラル間の相互転送のGPDMAサーキュラ･バッファリング

サーキュラ･バッファで、連続したメモリからペリフェラル（またはペリフェラルからメモリ）への転送を行うには、ソフトウェアが1/2転送と転送完了イベント／割込み生成（GPDMA_CxCR.HTIE = 1およびGPDMA_CxCR.TCIE = 1）でチャネルを設定して、バッファの同時ソフトウェア処理を可能にする必要があります。

LLI_0を、リニア･アドレス指定チャネルで最初のブロック転送に設定します。メモリからペリフェラルへの転送（ペリフェラルからメモリへの転送）で、メモリの転送元（転送先）の開始アドレスを復元するには、連続実行のLLI_1が必要です。GPDMAは、ブロック転送が完了し、GPDMA_CxBR1の復元が必要無くなると、最初にプログラムされたGPDMA_CxBR1.BNDT[15:0]を自動的にリロードします。

リニア･アドレス指定のGPDMAチャネルと転送元のサーキュラ･バッファによる、このプログラミングを下の図に示します。

図68. GPDMAサーキュラ･バッファ･プログラミング：リニア･アドレス指定によるメモリ開始アドレスの更新

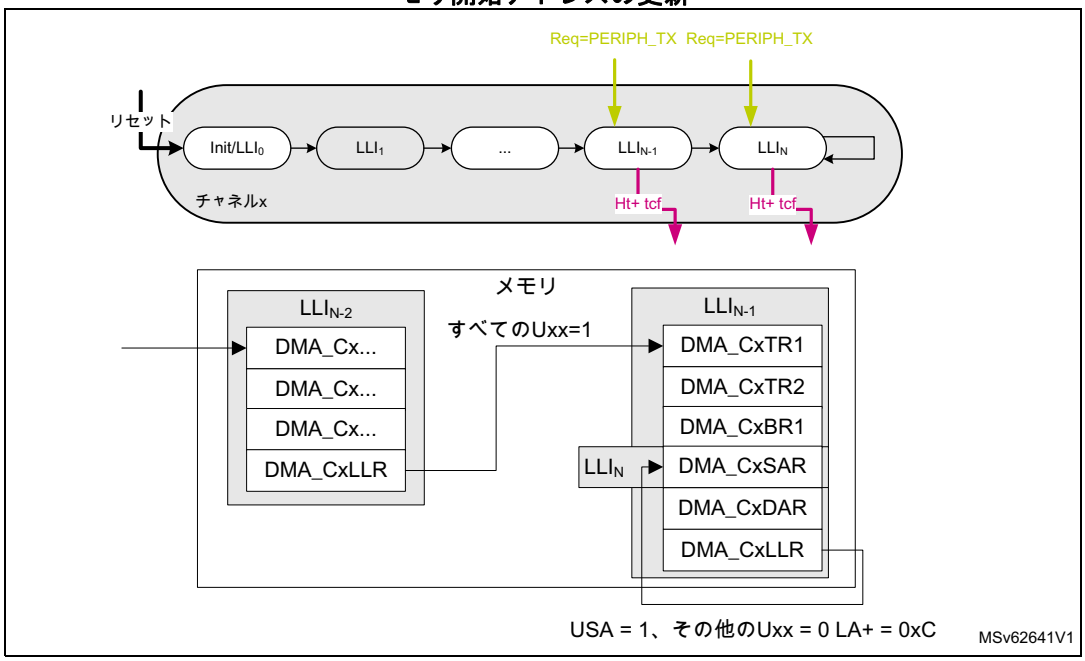


注： **2Dアドレス指定チャネルでは、ユーザがGPDMA_CxBR1.BRC[10:0] = 1でシングルLLIを使用し、GDMA_CxBR2とGDMA_CxBR1で負のメモリ･ブロック･アドレス･オフセットをプログラムして、メモリ転送元か転送先の開始アドレスへジャンプすることができます。**

共有GPDMAのチャネルxを経由した、他の転送後にサーキュラ･バッファリングを実行しなくてはならない場合、最初のブロック転送の設定にはメモリにある直前のLLI_{N-1}が必要です。最後のLLI_Nで、メモリからペリフェラルへの転送（ペリフェラルからメモリへの転送）で、メモリの転送元（転送先）の開始アドレスを復元します。

リニア･アドレス指定の共有GPDMAチャネル、と転送元のサーキュラ･バッファによる、このプログラミングを下の図に示します。

図69. サーキュラ･バッファのある共有GPDMAチャネル：リニア･アドレス指定によるメモリ開始アドレスの更新



17.4.14 GPDMAセキュア／非セキュア･チャネル

GPDMAコントローラは、チャネル･レベルでTrustZoneハードウェア･アーキテクチャに準拠しています。このアーキテクチャでは、コントローラの全リソースが、常にセキュア･ワールドおよび非セキュア･ワールドのどちらか一方に存在できるようにパーティション分割されています。

あらゆるチャネルxがセキュアまたは非セキュアのハードウェア･リソースで、GPDMA_SECCFGR.SECxにより設定されます。

チャネルxがセキュア特権エージェントによりセキュア状態に設定されている場合、以下のアクセス制御規則が適用されます。

- このチャネルのレジスタ･フィールドへの非セキュア読出しアクセスは、非セキュア･エージェントで読み出すことができるGPDMA_SECCFGR、GPDMA_PRIVCFGR、GPDMA_RCFGLOCKRを除いて、強制的に0を返します。
- このチャネルのレジスタ･フィールドへの非セキュア書込みアクセスによる影響はありません。

チャネルxがセキュア状態に設定されている場合、セキュア･エージェントは、転送元からのGPDMAデータ転送（GPDMA_CxTR1.SSEC）と、転送先へのGPDMAデータ転送（GPDMA_CxTR1.DSEC）を個別にセキュアまたは非セキュアに設定することができます。

チャネルxがセキュア状態のリンクリスト･モードに設定されている場合、GPDMAメモリから、そのレジスタ･ファイルへの次のリンクリスト･データ構造のロードは、マスタ･ポートに配置されたGPDMA_CxCR.LAPを経由したセキュア転送により自動的に実行されます。

GPDMAはGPDMA_SECCFGRを反映したセキュア･バスを生成して、他のペリフェラルに各GPDMAチャネルxのセキュア／非セキュア状態を常に通知します。

また、GPDMAは、セキュアGPDMAレジスタに対する不正な非セキュア･アクセスに関し、セキュリティの不正アクセス･パルス信号を生成します。この信号はTrustZone割込みコントローラに送られます。

セキュア･ソフトウェアがチャネルをセキュア状態から非セキュア状態に切り替える必要がある場合、セキュア･ソフトウェアはそのチャネルをアボートするか、セキュア･チャネルが完了するのを待ってから切り替えなくてはなりません。これは、チャネルを次の非セキュア転送に動的に再割り当てするために必要です。非セキュア･ソフトウェアは、これを行うことが許可されておらず、次の転送に向けGPDMA_CxCRを再プログラムできるようになる前にGPDMA_CxCR.EN = 0である必要があります。セキュア･ソフトウェアは、チャネルx（GPDMA_CxCR.RESET=1）だけではなく、フルチャネルxレジスタ･ファイルもリセット値にリセットする場合があります。

17.4.15 GPDMA特権／非特権チャネル

あらゆるチャネルxは、GPDMA_PRIVCFGR.PRIVxによる特権エージェントの設定に従って、特権または非特権ハードウェア･リソースとなります。

チャネルxが特権エージェントにより特権状態に設定されている場合、次のアクセス制御規則が適用されます。

- このチャネルのレジスタ･フィールドへの非特権読出しアクセスは、非特権･エージェントで読み出すことができるGPDMA_PRIVCFGR、GPDMA_SECCFGR、GPDMA_RCFGLOCKRを除いて、強制的に0を返します。
- このチャネルのレジスタフィールドへの非特権書込みアクセスによる影響はありません。

チャネルが特権（非特権）状態に設定されると、ソースおよびデスティネーション･データ転送は、AHBマスタ･ポート経由の特権（非特権）転送です。

チャネルが特権（非特権）状態のリンクリスト･モードに設定されている場合、GPDMAメモリからのそのレジスタ･ファイルへの次のリンクリスト･データ構造のロードは、マスタ･ポートに割り当てられたGPDMA_CxCR.LAPを経由した特権（非特権）転送で自動的に実行されます。

GPDMAはGPDMA_PRIVCFGRを反映した特権バスを生成して、他のペリフェラルに各GPDMAチャネルxの特権／非特権状態を常に通知します。

特権ソフトウェアがチャネルを特権状態から非特権状態に切り替える必要がある場合、特権ソフトウェアはそのチャネルをアボートするか、特権チャネルが完了するのを待ってから切り替えなくてはなりません。これは、チャネルを次の非特権転送に動的に再割り当てを行うために必要です。非特権ソフトウェアは、これを行うことが許可されておらず、次の転送に向け、GPDMA_CxCRを再プログラムできるようになるにはGPDMA_CxCR.EN=0でなくてはなりません。特権ソフトウェアは、チャネルx（GPDMA_CxCR.RESET = 1）だけではなく、xレジスタ･ファイルの全チャネルもリセット値にリセットする場合があります。

17.4.16 GPDMAエラー管理

GPDMAは次に示すように、根本原因に応じて転送エラーを管理しユーザに報告することができます。

データ転送エラー

転送元または転送先への（AHBのシングルまたはバースト転送としての）バス･アクセスに関して。

- 転送元または転送先のターゲットはAHBエラーを報告します。
- プログラムしたチャネル転送が停止します（GPDMA_CxCR.ENがGPDMAハードウェアでクリア）。チャネル状態レジスタはアイドル状態（GPDMA_CxSR.IDLEF = 1）とデータ･エラー（GPDMA_CxSR.DTEF = 1）を報告します。
- GPDMAデータ転送エラーの後、ユーザは、保護属性を含む、製品が定義した転送元と転送先のメモリ･マッピングを考慮して、デバッグ･セッションを実行しなくてはなりません。

- GPDMAデータ転送エラーの後、ユーザはチャネルのリセットを発行して（GPDMA_CxCR.RESETをセット）、ハードウェアの･GPDMAチャネル･データ経路とFIFOの内容をリセットしてから、そのチャネルを次の転送に対して有効にしなくてはなりません。

リンク転送エラー

メモリにあるプログラムされたLLIからのGPDMAチャネル･レジスタの暫定的な更新に関して。

- リンクリスト･メモリはAHBエラーを報告します。
- プログラムされたチャネル転送は停止し（GPDMA_CxCR.ENがGPDMAハードウェアでクリアされる）、チャネル状態レジスタがアイドル状態（GPDMA_CxSR.IDLEF = 1）と、リンク･エラー（GPDMA_CxSR.ULEF = 1）を報告します。
- GPDMA のリンクエラーの後、ユーザは、保護属性を含む、製品が定義したリンクリスト･データ構造（GPDMA_CxLBARおよびGPDMA_CxLLR）のメモリ･マッピングを考慮して、デバッグ･セッションを実行しなくてはなりません。
- GPDMA　リンク･エラーの後、ユーザはリンクリスト･レジスタ･ファイル（GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、GPDMA_CxLLR、に加え、GPDMA_CxTR3およびGPDMA_CxBR2）を明示的に書き出してから、そのチャネルを次の転送のために有効にしなくてはなりません。

ユーザ設定エラー

非認証ユーザ設定でのGPDMA転送の暫定実行に関して。

- プログラムされたチャネル転送を無効にし（GPDMAハードウェアによりGPDMA_CxCR.ENを強制しクリアし）、次の非認証プログラム･データ転送が実行されることを回避します。チャネル状態レジスタはアイドル状態（GPDMA_CxSR.IDLEF = 1）とユーザ設定エラー（GPDMA_CxSR.USEF = 1）を報告します。
- GPDMAユーザ設定エラーの後、ユーザはGPDMAチャネル･プログラミングを考慮に入れて、デバッグ･セッションを実行しなくてはなりません。ユーザ設定エラーは次のどれかが原因となっている可能性があります。
  - 次の LLI_1 によるプログラムされたこの値の更新が無い、プログラムされた null の転送元ブロック･サイズ（GPDMA_CxBR1.BNDT[15:0] = 0およびGPDMA_CxLLR.UB1 = 0）
  - プログラムされた null ではない転送元ブロック･サイズが、ソース･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxBR1.BNDT[2:0]に対してGPDMA_CxTR1.SDW_LOG2[1:0]）
  - パッキング／アンパッキング･モード（PAM[1] = 1）の場合、プログラムされたnullではない転送元ブロック･サイズが、デスティネーション･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxBR1.BNDT[2:0]に対してGPDMA_CxTR1.DDW_LOG2[1:0]）
  - プログラムされたアンアラインド転送元開始アドレスが、ソース･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxSAR[2:0]に対してGPDMA_CxTR1.SDW_LOG2[1:0]）
  - チャネルx（x = 12~15）: プログラムされたアンアラインド転送元アドレス･オフセットがソース･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxTR3.SAO[2:0]に対してGPDMA_CxTR1.SDW_LOG2[1:0]）
  - チャネルx（x = 12~15）: プログラムされたアンアラインド･ブロック繰返し転送元アドレス･オフセットがソース･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxBR2.BRSAO[2:0]に対してGPDMA_CxTR1.SDW_LOG2[1:0]）
  - プログラムされたアンアラインド転送先開始アドレスが、デスティネーション･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxDAR[2:0]に対してGPDMA_CxTR1.DDW_LOG2[1:0]）

- チャネルx（x = 12~15）：プログラムされたアンアラインド転送先アドレス･オフセットがデスティネーション･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxTR3.DAO[2:0]に対してGPDMA_CxTR1.DDW_LOG2[1:0]）
- チャネルx（x = 12~15）：プログラムされたアンアラインド･ブロック繰返し転送先アドレス･オフセットが、デスティネーション･バースト転送のプログラムされたデータ幅の倍数ではない（GPDMA_CxBR2.BRDAO[2:0]に対してGPDMA_CxTR1.DDW_LOG2[1:0]）
- プログラムされたダブルワードの転送元データ幅（GPDMA_CxTR1.SDW_LOG2[1:0] = 11）
- プログラムされたダブルワードの転送先データ幅（GPDMA_CxTR1.DDW_LOG2[1:0] = 11）
- nullデータ転送による、プログラムされたリンクリスト･アイテムLLI_{n+1}（GPDMA_CxLLR.UB1 = 1およびGPDMA_CxBR1BNDT = 0）

17.4.17 GPDMA自律モード

プログラムされたリンクリスト転送をGPDMAが実行する際の動的電力消費を削減するため、RCCが対応するGPDMAイネーブル制御ビットでプログラムされている場合、GPDMAのハードウェアはそのクロック･ゲーティングを自動で管理し、デバイスがRUNまたは低消費電力モードのときには必ずRCCにクロック･リクエスト出力信号を生成します。

RCCプログラミングの詳細についてはリファレンス･マニュアルのRCCセクションを参照してください。

GPDMA自律機能に対する低消費電力モードの可用性の詳細に関してはセクション17.3.2を参照してください。

次に示すように、ユーザはGPDMA_CxTR2により、特定のGPDMA転送の実行をGPDMAチャネルxのLLI_nレベルでプログラムしてスケジューリングすることができます。

- ソフトウェアが、入力トリガとリクエストのためのTRIGM[1:0]、TRIGPOL[1:0]、TRIGSEL[6:0]、SWREQ、REQSEL[6:0]で転送の入力を制御し、条件付けします。
- ソフトウェアが、TCEM[1:0]で転送出力の制御および信号出力を行い、転送完了イベントを生成するかどうか、および関連する1/2データ転送イベントを生成するかどうかを決めています。

詳細は、GPDMAチャネルx転送レジスタ2（GPDMA_CxTR2）を参照してください。

低消費電力モードで使用する場合、この機能は、有効になったGPDMA転送完了割込み（GPDMA_CxCR.TCIE = 1）により、特定の転送の完了でCPUウェイクアップを有効にし、および／または、同じチャネルでの別のLLI_{n+1}転送を操作するために自律GPDMAの継続を有効にします。

出力チャネルx転送完了イベントgpdma_chx_tcは、このイベントがループバックし、GPDMAレベルで接続されている場合（セクション17.3.5を参照）、チャネル用の選択した入力トリガとしてプログラムすることが可能で、クリアされた転送完了フラグ（TCF）なしで、自律的に細かいGPDMAチャネル間転送スケジューリングを可能にします。

任意のGPDMAチャネルxは、次のいずれかの条件でそのクロック･リクエストをアサートします。

- 実行する予定の次の転送がトリガ（GPDMA_CxTR2.TRIGPOL[1:0]およびGPDMA_CxTR2.TRIGM[1:0]）により条件付けされるようにプログラムされている場合、そのトリガのヒットが発生している時のみ。
- 実行する予定の次の転送がトリガにより条件付けされない場合は次のとおりです。
  - GPDMA_CxTR2.SWREQ = 0の場合、選択されたペリフェラルによりハードウェア･リクエストがアサートされたときのみ。
  - GPDMA_CxTR2.SWREQ = 1（メモリ間、GPIOとメモリ相互の間）の場合、GPDMAが有効になると直ぐ。

次の条件をすべて満足すると直ぐにGPDMAチャネルxがそのクロック･リクエストをリリースします。

- 実行予定の転送が完了している。
- GPDMAチャネルxはすぐには準備が整わず、次の転送の実行がリクエストされていない。
- チャネルx割込みが立っている場合、この割込みを発生させる可能性のある状態レジスタのすべてのフラグがソフトウェア･エージェントによりクリアされます。

1つのチャネルによってそのクロック･リクエストがアサートされる場合、GPDMAによってそのクロック･リクエストがRCCにアサートされます。どのチャネルからもクロック･リクエストがアサートされない場合、GPDMAによってRCCへのクロック･リクエストが解除されます。

17.5 デバッグ･モードでのGPDMA

マイクロコントローラがデバッグ･モードに入ると（コアが停止）、あらゆるチャネルxを、DBGMCUモジュールのプログラマブル制御ビットに応じて、個別に継続（デフォルト）またはサスペンドにすることが可能です。

注： **デバッグ･モードでは、GPDMA_CxSR.SUSPFはDBGMCUモジュールのプログラマブル制御ビットによるサスペンドの影響は受けません。この場合、GPDMA_CxSR.IDLEFをチェックしてチャネル･サスペンドの完了状態を知ることができます。**

17.6 低消費電力モードでのGPDMA

表141.低消費電力モードがGPDMAに与える影響

モード	説明
SLEEP	影響はありません。GPDMA割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	STOPモードに入っても、GPDMAレジスタの内容は保持されます。GPDMAレジスタの内容はメモリから次のリンクリスト･アイテムにより自律的に更新され、自律的なデータ転送を実行することが可能です。デバイスは、GPDMA割込みによってSTOPモード(1)から復帰することが可能です。
STANDBY	GPDMAはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. STOPモードが何かサポートされるかについては、セクション17.3.2を参照してください。

17.7 GPDMA割込み

各チャネルと各CPUには個別に1つのGPDMA割込みラインがあります（デバイスにいくつかある場合）。

表142.GPDMA割込みリクエスト

項目（割込みの略称）	割込みイベント	割込みイネーブル	イベントフラグ	イベントのクリア方法
GPDMA_CHx	転送完了	GPDMA_CxCR.TCIE	GPDMA_CxSR.TCF	GPDMA_CxFCR.TCFに1を書き込む
	1/2転送	GPDMA_CxCR.HTIE	GPDMA_CxSR.HTF	GPDMA_CxFCR.HTFに1を書き込む
	データ転送エラー	GPDMA_CxCR.DTEIE	GPDMA_CxSR.DTEF	GPDMA_CxFCR.DTEFに1を書き込む
	更新リンク･エラー	GPDMA_CxCR.ULEIE	GPDMA_CxSR.ULEF	GPDMA_CxFCR.ULEFに1を書き込む
	ユーザ設定エラー	GPDMA_CxCR.USEIE	GPDMA_CxSR.USEF	GPDMA_CxFCR.USEFに1を書き込む
	サスペンド状態	GPDMA_CxCR.SUSPIE	GPDMA_CxSR.SUSPF	GPDMA_CxFCR.SUSPFに1を書き込む
	トリガ･オーバーラン	GPDMA_CxCR.TOFIE	GPDMA_CxSR.TOF	GPDMA_CxFCR.TOFに1を書き込む

GPDMAチャネルxイベントは次のいずれかの場合があります。

- 転送完了
- 1/2転送完了
- 転送エラー、その原因は次のいずれか。
  - データ転送エラー
  - 更新リンク･エラー
  - ユーザ設定エラー、完了サスペンション
- トリガ･オーバーラン

注： **チャネルx転送完了イベントが発生すると、出力信号gpdma_chx_tcが1クロック･サイクルのハイ･パルスとして生成されます。**

次の両方の条件を満足すると、あらゆるxxイベントに続いて割込みが生成されます。

- 対応する割込みイベントxxが有効である（GPDMA_CxCR.xxIE=1）。
- 対応するイベント･フラグがクリアされている（GPDMA_CxSR.xxF=0）。これは、前の同じxxイベント発生の後に、ソフトウェア･エージェントが対応するxxフラグのクリア制御ビットに1を書き込んだはずだということを意味します（GPDMA_CxFCR.xxFに1を書き込む）。

TCF（転送完了）とHTF（1/2転送）イベントの生成は次のようにGPDMA_CxTR2.TCEM[1:0]によって制御されます。

- 転送完了イベントは、転送完了イベント･モード（GPDMA_CxTR2.TCEM[1:0]）に応じて、ブロック転送完了、2D／繰返しブロック転送完了、または、次のLLI（がある場合）のアップロードを含むLLI転送完了、あるいはフル･リンクリスト完了のいずれかです。
- 1/2転送イベントは、転送完了イベントモードGPDMA_CxTR2.TCEM[1:0]により、1/2ブロック転送または、2D／繰返しブロックの1/2転送です。

  転送元ブロックのバイト･サイズの半分（GPDMA_CxBR1.BNDT[15:0]/2を整数値に切り上げ）が転送先に転送されたときに1/2ブロック転送となります。

  繰返しブロックの半分((GPDMA_CxBR1.BRC[10:0]+1)/2を整数値に切り上げ）が転送先に転送されたときに、2D／繰返し1/2ブロック転送となります。

詳細は、GPDMAチャネルx転送レジスタ2（GPDMA_CxTR2）を参照してください。

次のいずれかの状況で転送エラーが立ちます。

- 転送元から、または転送先へのシングル／バースト･データ転送時（DTEF）
- メモリにあるプログラムされたLLIによるGPDMAチャネル･レジスタの更新時（ULEF）
- 非認証設定でのGPDMAチャネルの暫定実行時（USEF）

  ユーザはGPDMAチャネル･プログラミングをUSEFの根本原因リストに対して修正するデバッグ･セッションを行わなくてはなりません（セクション17.4.16参照）。

トリガ･オーバーランはトリガ･ヒットのメモリ書込みとトリガ･オーバーラン･フラグの生成に記載します。

17.8 GPDMAレジスタ

GPDMAレジスタは32bitワード･データ･でのアラインド･アクセスでアクセスしなくてはなりません。

17.8.1 GPDMAセキュア設定レジスタ（GPDMA_SECCFGR）

アドレスオフセット：0x00

リセット値：0x0000 0000

このレジスタへの書込みアクセスはセキュアおよび特権でなくてはなりません。読出しアクセスはセキュアまたは非セキュア、特権または非特権です。

対応するチャネルxがロックされている場合（GPDMA_RCFGLOCKR.LOCKx=1）、書込みアクセスはビット･レベルで無視されます。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

このレジスタは、GPDMAチャネルの初期化／クローズ時（GPDMA_CxCR.EN = 0のとき）にビット･レベルでプログラムし、あらゆるチャネルxを個別にセキュアまたは非セキュア･ワールドに安全に割り当てる必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **SECx**：チャネルx（x = 15 ~ 0）のセキュア状態

0：非セキュア

1：セキュア

17.8.2 GPDMA特権設定レジスタ（GPDMA_PRIVCFGR）

アドレスオフセット：0x04

リセット値：0x0000 0000

このレジスタへの書込みアクセスは特権でなくてはなりません。読出しアクセスは特権または非特権、セキュアまたは非セキュアで行うことができます。

このレジスタにはセキュアな情報と非セキュアな情報を混在させることができます。チャネルxがセキュアに設定された場合（GPDMA_SECCFGR.SECx = 1）、PRIVxビットはセキュア･（および特権）エージェントによってのみ書き込むことが可能です。

対応するチャネルxがロックされている場合（GPDMA_RCFGLOCKR.LOCKx = 1）、書込みアクセスはビット･レベルで無視されます。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

このレジスタは、GPDMAチャネルの初期化／閉鎖時（GPDMA_CxCR.EN = 0）にビット･レベルでプログラムし、あらゆるチャネルxを特権または非特権ワールドに割り当てる必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PRIVx**：チャネルx（x = 15~0）の特権状態

0：非特権

1：特権

17.8.3 GPDMA設定ロック･レジスタ（GPDMA_RCFGLOCKR）

アドレスオフセット：0x08

リセット値：0x0000 0000

このレジスタは、例えばブート時などに、あらゆるGPDMAチャネル／リソースのセキュア特権属性を個別にロックするために（例えばブート時に、あらゆるチャネルxのGPDMA_CxSECCFGRおよびGPDMA_CxPRIVCFGRの設定をロックするために）、セキュア特権属性のソフトウェア･エージェントにより書き込むことが可能です。

読出しアクセスは特権または非特権、セキュアまたは非セキュアで行うことができます。

注： **TZEN = 0の場合、このレジスタは書込みできません。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOCK15	LOCK14	LOCK13	LOCK12	LOCK11	LOCK10	LOCK9	LOCK8	LOCK7	LOCK6	LOCK5	LOCK4	LOCK3	LOCK2	LOCK1	LOCK0
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LOCKx**：グローバルGPDMAリセットされるまで、GPDMA_SECCFGR.SECxおよびGPDMA_PRIVCFGR.PRIVxの設定をロックします（x = 15 ~ 0）。

このビットはリセット後にクリアされ、一旦セットされると、グローバルGPDMAリセットされるまでリセットすることはできません。

0：チャネルxのセキュア特権設定が書込み可能です。

1：チャネルxのセキュア特権設定が書込み不可です。

17.8.4 GPDMA非セキュア･マスク割込みステータス･レジスタ（GPDMA_MISR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

レジスタは読出しレジスタです。

これは非セキュア･レジスタで、各非セキュア･チャネルxのマスク割込みステータス･ビットMISxを含みます（GPDMA_SECCFGR.SECx = 0のチャネルx設定）。これはGPDMA_CxSRのすべてのフラグの論理和で、各転送元フラグは対応するGPDMA_CxCRの割込みイネーブル･ビットにより有効になっています。

すべてのビットは、GPDMA_CxFCRの対応するフラグ･クリアビットに1を書き込むと、ハードウェアによりデアサートされます。

チャネルxがセキュア状態の場合（GPDMA_SECCFGR.SECx = 1）、このチャネルxのマスク割込みステータス･ビットMISxへの読出しアクセスは0を返します。

このレジスタには、各チャネルの特権状態GPDMA_PRIVCFGR.PRIVxによっては、特権情報と非特権情報が混在する可能性があります。特権ソフトウェアは非セキュア割込みステータスをすべて読み

出すことができます。非特権ソフトウェアに許可されるのは非特権（および非セキュア）チャネルのステータス読出しのみで、その他の特権ビット･フィールドはゼロが返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MIS15	MIS14	MIS13	MIS12	MIS11	MIS10	MIS9	MIS8	MIS7	MIS6	MIS5	MIS4	MIS3	MIS2	MIS1	MIS0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **MISx**：チャネルxのマスク割込みステータス（x = 15 ~ 0）

0：チャネルxで割込みは発生していません

1：チャネルxで割込みが発生しました

17.8.5 GPDMAセキュア･マスク割込みステータス･レジスタ（GPDMA_SMISR）

アドレスオフセット：0x10

リセット値：0x0000 0000

これはセキュア読出しレジスタで、各セキュア･チャネルxのマスク割込みステータス･ビットMISxを含みます（GPDMA_SECCFGR.SECx = 1）。これはGPDMA_CxSRフラグすべての論理和で、各転送元フラグは、対応するGPDMA_CxCR割込みイネーブル･ビットにより有効になっています。

各ビットは、対応するGPDMA_CxFCRフラグ･クリア･ビットに1をセキュアに書き込むと、ハードウェアによりデアサートされます。

このレジスタは非セキュア･チャネルに関する情報は何も含みません。

このレジスタには、各チャネルの特権状態GPDMA_PRIVCFGR.PRIVxによっては、特権情報と非特権情報が混在する可能性があります。特権ソフトウェアはセキュア割込みステータスをすべて読み出すことができます。非特権ソフトウェアに許可されるのは非特権およびセキュア･チャネルのステータス読出しのみで、その他の特権ビット･フィールドはゼロが返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MIS15	MIS14	MIS13	MIS12	MIS11	MIS10	MIS9	MIS8	MIS7	MIS6	MIS5	MIS4	MIS3	MIS2	MIS1	MIS0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **MISx**：セキュア･チャネルxのマスク割込みステータス（x = 15 ~ 0）

0：セキュア･チャネルxで割込みは発生していません

1：セキュア･チャネルxで割込みが発生しました

17.8.6 GPDMAチャネルxのリンクリスト･ベース･アドレス･レジスタ（GPDMA_CxLBAR）

アドレスオフセット：0x50 + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは特権ソフトウェアによって書き込まれる必要があります。これは、チャネルx GPDMA_PRIVCFGR.PRIVxの特権ステータスにより、特権読取りが可能かどうかが異なります。

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）により、セキュアまたは非セキュアとなります。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルをベースとするレジスタは、あるチャネルxに対して、メモリ領域のリンクリスト･ベースのアドレスで、ここからGPDMA転送のプログラムされたシーケンスを記述するLLIが、条件付きで自動的に更新されます。

64KBに合わせたチャネルxリンクリストのベース･アドレスは、各LLIのワード整列アドレス･オフセットを定義する、16bit GPDMA_CxLLRレジスタによりオフセットされています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LBA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:16 **LBA[31:16]**：GPDMAチャネルxのリンクリスト･ベース･アドレス

ビット15:0 予約済みであり、リセット値に保持する必要があります。

17.8.7 GPDMAチャネルxフラグ･クリア･レジスタ（GPDMA_CxFCR）

アドレスオフセット：0x5C + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

これは書込みレジスタで、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	w	w	w	w	w	w	w								

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOF**：トリガ･オーバーラン･フラグのクリア
0：影響なし。
1：対応するTOFフラグがクリアされました

ビット13 **SUSPF**：完了したサスペンド･フラグのクリア
0：影響なし。
1：対応するSUSPFフラグがクリアされました

ビット12 **USEF**：ユーザ設定エラー･フラグのクリア
0：影響なし。
1：対応するUSEFフラグがクリアされました

ビット11 **ULEF**：更新リンク転送エラー･フラグのクリア
0：影響なし。
1：対応するULEFフラグがクリアされました

ビット10 **DTEF**：データ転送エラー･フラグのクリア
0：影響なし。
1：対応するDTEFフラグがクリアされました

ビット9 **HTF**：1/2転送フラグのクリア
0：影響なし。
1：対応するHTFフラグがクリアされました

ビット8 **TCF**：転送完了フラグのクリア
0：影響なし。
1：対応するTCFフラグがクリアされました

ビット7:0 予約済みであり、リセット値に保持する必要があります。

17.8.8 GPDMAチャネルxステータス･レジスタ（GPDMA_CxSR）

アドレスオフセット：0x60 + 0x80 * x（x = 0から15）

リセット値：0x0000 0001

これはチャネル･ステータスを報告する読出しレジスタです。

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFOL[7:0]							
								r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDLEF
	r	r	r	r	r	r	r								r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **FIFOL[7:0]**：モニタされたFIFOレベル

FIFOで利用可能な書込みビートの数。単位はプログラムされた転送先データ幅です（バイト、ハーフワード、ワードの単位についてはGPDMA_CxTR1.DDW_LOG2[1:0]参照）。

注： 有効な転送をサスペンドした後、転送先にどれぐらいのデータが転送されたかを知るには、GPDMA_CxBR1.BDNT[15:0]とGPDMA_CxBR1.BRC[10:0]に加え、FIFOL[7:0]を読み出すことが必要な場合があります。読み出す前に転送のサスペンド（GPDMA_CxSR.SUSPF = 1）を待たなくてはならない場合があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOF**：トリガ･オーバーラン･フラグ

0：トリガ･オーバーラン･イベントはありません

1：トリガ･オーバーラン･イベントが発生しました

ビット13 **SUSPF**：完了したサスペンド･フラグ

0：完了したサスペンド･イベントはありません

1：サスペンド･イベントの完了が発生しました

ビット12 **USEF**：ユーザ設定エラー･フラグ

0：ユーザ設定エラー･イベントはありません

1：ユーザ設定エラー･イベントが発生しました

ビット11 **ULEF**：更新リンク転送エラー･フラグ

0：更新リンク転送エラー･イベントはありません

1：メモリからリンクリスト･レジスタを更新しているときに、マスタ･バス･エラー･イベントが発生しました

ビット10 **DTEF**：データ転送エラー･フラグ

0：データ転送エラー･イベントはありません

1：データ転送でマスタ･バス･エラー･イベントが発生しました

ビット9 **HTF**：1/2転送フラグ

0：1/2転送イベントは発生していません

1：1/2転送イベントが発生しました。

1/2転送イベントは、転送完了イベントモード(GPDMA_CxTR2.TCEM[1:0])に応じて、1/2ブロック転送または、2D／繰返しブロックの1/2転送です。

転送元ブロック･サイズの半分のバイト数（GPDMA_CxBR1.BNDT[15:0]/2を整数値に切り上げ）が転送先に転送されたときに1/2ブロック転送となります。

繰返しブロックの半分（(GPDMA_CxBR1.BRC[10:0] + 1)/2)を整数値に切り上げ）が転送先に転送されたときに、2D／繰返し1/2ブロック転送となります。

ビット8 **TCF**：転送完了フラグ

0：転送完了イベントは発生していません

1：転送完了イベントが発生しました

転送完了イベントは、転送完了イベント･モード（GPDMA_CxTR2.TCEM[1:0]）により、ブロック転送完了、2D／繰返しブロック転送完了、または、次のLLI（がある場合）のアップロードを含むLLI転送完了、あるいはフル･リンクリスト完了のいずれかです。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **IDLEF**：アイドル･フラグ

0：チャネルはアイドル状態ではありません

1：チャネルはアイドル状態です

このアイドル･フラグは、チャネルが有効なチャネル設定（USEFがすぐには報告されない）でイネーブルになっていると（GPDMA_CxCR.EN = 1）、ハードウェアによってデアサートされます。

このアイドル･フラグは、チャネルが（サスペンドまたはディスエーブル状態で）アイドル状態に戻った時に、ハードウェアによりアサートされるか、ハード･リセット後にアサートされます。

17.8.9 GPDMAチャネルx制御レジスタ（GPDMA_CxCR）

アドレスオフセット：0x64 + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタはチャネルの制御（アクティベート、サスペンド、アボート、ディスエーブル）に使われます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIO[1:0]		Res.	Res.	Res.	Res.	LAP	LSM
								rw	rw					rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOIE	SUSPIE	USEIE	ULEIE	DTEIE	HTIE	TCIE	Res.	Res.	Res.	Res.	Res.	SUSP	リセット	EN
	rw	rw	rw	rw	rw	rw	rw						rw	w	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:22 **PRIO[1:0]**：チャネルxGPDMA転送の、他に対する優先レベル

00：優先度低、重み低

01：優先度低、重み中

10：優先度低、重み高

11：優先度高

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット21:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **LAP**：リンクリスト割り当てポート

このビットは、メモリからのGPDMAのリンクリスト･レジスタの更新にマスタ･ポートを割り当てるために使用します。

0：ポート0（AHB）割り当て済み

1：ポート1（AHB）割り当て済み

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット16 **LSM**：リンク･ステップ･モード

0：フル･リンクリストで実行し、最後のLLIの終わりで完了したチャネル（GPDMA_CxLLR = 0）。このリンク･アドレスの下位16bitはnull（LA[15:0] = 0）で、すべての更新ビットもnull（UT1 = UB1 = UT2 = USA = UDA = ULL = 0およびUT3 = UB2 = 0）です。この時、GPDMA_CxBR1.BNDT[15:0] = 0およびGPDMA_CxBR1.BRC[10:0] = 0となります。

1：現在のLLIで**1回**実行されたチャネル

最初に、（おそらく、1D／繰返し）ブロック転送が、現在の内部レジスタ･ファイルの定義により、GPDMA_CxBR1.BNDT[15:0] = 0およびGPDMA_CxBR1.BRC[10:0] = 0まで実行されます。その後、次のリンクリスト･データ構造がGPDMA_CxLLRの定義に従い、条件付きでメモリからアップロードされます。それでチャネルの実行は完了です。

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOIE**:トリガ･オーバーラン割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット13 **SUSPIE**：完了サスペンド割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット12 **USEIE**：ユーザ設定エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット11 **ULEIE**：更新リンク転送エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット10 **DTEIE**：データ転送エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット9 **HTIE**：1/2転送完了割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット8 **TCIE**：転送完了割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SUSP**：サスペンド

RESET（ビット1）フィールドへ1を書き込むと、ビット2への書込みに関係なく、ハードウェアはこのビットをデアサートします。その他の場合：

ソフトウェアはアクティブなチャネル（マスタ･ポートを経由したGPDMA転送が進行中のチャネル）をサスペンドするために1を書き込まなくてはなりません。

ソフトウェアはサスペンドしたチャネルをレジュームするために、図51で詳細を説明しているプログラミング･シーケンスに従って0を書き込まなくてはなりません。

0：書込み：チャネルのレジューム、読出し：チャネルのサスペンドなし

1：書込み：チャネルをサスペンド、読出し：サスペンドしたチャネル

ビット1 **RESET**：リセット

このビットは書込み専用です。0を書き込んでも影響はありません。1の書込みは次の項目のリセットを意味します：FIFO、チャネルの内部状態、SUSPおよびENビット（それぞれ、ビット2およびビット0への書込み内容には関係ありません）。

リセットは、チャネルが定常状態のときに有効で、これは次のどれかを意味してます。

- サスペンドした状態のアクティブなチャネル（GPDMA_CxSR.SUSPF = 1および GPDMA_CxSR.IDLEF = GPDMA_CxCR.EN = 1）

- 無効状態のチャネル（GPDMA_CxSR.IDLEF = 1およびGPDMA_CxCR.EN = 0）

RESETを書き込んだ後、このチャネルを継続して使用するためには、ハードウェアにより変更される設定レジスタ（GPDMA_CxBR1,GPDMA_CxSARおよびGPDMA_CxDAR）など、チャネルを明示的に再設定してから、チャネルを再度イネーブルにすることが必要です（図52のプログラミング･シーケンス参照）。

0：チャネルのリセットなし

1：チャネルのリセット

ビット0 **EN**：イネーブル

RESET（ビット1）フィールドへ1を書き込むと、ビット0への書込み内容に関係なく、ハードウェアはこのビットをデアサートします。その他の場合：

このビットは、転送エラー（マスタ･バス･エラーまたはユーザ設定エラー）またはチャネル転送完了（例えば、LLIのシングル実行の最後にLSM = 1になっている場合など、チャネルの設定準備が完了している）時にハードウェアによりデアサートされます。

その他の場合、このビットは、ソフトウェアでアサートできます。

このENビットへ0を書き込んでも無視されます。

0：書込み：無視、読出し：チャネル無効化

1：書込み：チャネルの有効化、読出し：有効になったチャネル

17.8.10 GPDMAチャネルx転送レジスタ1（GPDMA_CxTR1）

アドレスオフセット：0x90 + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは、セキュアDSECとSSECを除き、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）に従って、セキュアまたは非セキュアとなり、GPDMA_PRIVCFGR.PRIVxのチャネルxの特権ステートに従って、特権または非特権となります。

このレジスタは、チャネルxの転送を制御します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了したら、このレジスタに書き込まなくてはなりません。その後、ハードウェアはGPDMA_CxCR.ENをデアサートします。チャネル転送は、ブロック、2D／繰返しブロック、LLI、あるいはフル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、GPDMA_CxLLR.UT1 = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DSEC	DAP	Res.	Res.	DHX	DBX	DBL_1 [5:0]						DINC	Res.	DDW_LOG2 [1:0]	
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSEC	SAP	SBX	PAM[1:0]		Res.	SBL_1 [5:0]						SINC	Res.	SDW_LOG2 [1:0]	
rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット31 **DSEC**：転送先へのGPDMA転送のセキュリティ属性

GPDMA_SECCFGR.SECx = 1で、アクセスがセキュアな場合。

0：GPDMA転送は非セキュアです。

1：GPDMA転送はセキュアです。

これはセキュア･レジスタ･ビットです。このビットはセキュア･ソフトウェアからのみ読み出すことができます。GPDMA_SECCFGR.SECx = 1のとき、このビットはセキュア･ソフトウェアにより書き込まれなくてはなりません。GPDMA_SECCFGR.SECx = 0のとき、セキュア書込みは無視されます。

GPDMA_SECCFGR.SECxがデアサートされると、このDSECビットも（チャネルのセキュア再設定で非セキュアとして）ハードウェアによりデアサートされ、転送先へのGPDMA転送は非セキュアになります。

ビット30 **DAP**：転送先割り当てポート

このビットはデスティネーション転送のマスタ･ポートの割り当てに使用されます。

0：ポート0（AHB）割り当て済み

1：ポート1（AHB）割り当て済み

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DHX**：転送先ハーフワード交換

転送先データ･サイズがワードより短い場合、このビットは無視されます。

転送先データ･サイズがワードの場合。

0：ワード内でのハーフワードベースの交換は無し

1：2つの連続した（PAM以降）ハーフワードが、各転送先ワードで交換されます。

ビット26 **DBX**：転送先バイト交換

転送先データ･サイズがバイト単位の場合、このビットは無視されます。

転送先データ･サイズがバイトではない場合。

0：ハーフワード内でのバイトベースの交換なし

1：2つの連続した（PAM以降）バイトが、各転送先ハーフワードで交換されます。

ビット25:20 **DBL_1[5:0]**：転送先のバースト長から1を引いた0から63の値。

バースト長の単位は、ビートと呼ばれるバーストのデータ1つ分です。DBL_1[5:0]=0の場合、バーストはシングルと呼ばれることがあります。各データ／ビートの幅は転送先データ幅DDW_LOG2[1:0]で定義されます。

注： バースト転送がAHB転送の1KBのアドレス境界を越えた場合、GPDMAはAHBプロトコルに準拠するよう、プログラムされたバーストをシングルまたは、より短いバーストへと変更、短縮を行います。

バースト転送の長さがチャネルxのFIFOサイズより大きい場合、GPDMAはFIFOサイズに準拠するよう、プログラムされたバーストをシングルまたは、より短いバーストへと変更、短縮を行います。GPDMAの再アービトレーションが有効長とより短いシングル／バースト長の間で行われるため、転送効率は低いですが、データの整合性は保証されます。

ビット19 **DINC**：転送:先インクリメント･バースト

0：固定バースト

1：隣接インクリメント･バースト

GPDMA_CxDARが指す転送先アドレスは、バースト･ビート／シングル転送後も保持されるか、隣接データに対応するオフセット分だけ転送後にインクリメントされます。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **DDW_LOG2[1:0]**：バーストの転送先データ幅の二進対数（バイト単位）

00：バイト

01：ハーフワード（2バイト）

10：ワード（4バイト）

11：ユーザ設定エラーが報告され、転送は行われませんでした

注： 8バイト･データ幅を設定すると、ユーザ設定エラーが報告され、転送は行われません。

デスティネーション･バースト転送は、そのデータ幅に合ったアドレスでなくてはなりません（開始アドレス GPDMA_CxDAR[2:0] で、DDW_LOG2[1:0] に対するアドレス･オフセット GPDMA_CxTR3.DAO[2:0]）。そうしないとユーザ設定エラーが報告され、転送は行われません。

ビット15 **SSEC**：転送元からのGPDMA転送のセキュリティ属性

GPDMA_SECCFGR.SECx = 1で、アクセスがセキュアな場合。

0：GPDMA転送は非セキュアです。

1：GPDMA転送はセキュアです。

これはセキュア･レジスタ･ビットです。このビットはセキュア･ソフトウェアからのみ読み出すことができます。GPDMA_SECCFGR.SECx = 1のとき、このビットはセキュア･ソフトウェアにより書き込まれなくてはなりません。GPDMA_SECCFGR.SECx = 0のとき、セキュア書込みは無視されます。

GPDMA_SECCFGR.SECxがデアサートされると、このSSECビットもハードウェアによりデアサートされ（チャネルのセキュア再設定で非セキュアとして）、転送元からのGPDMA転送は非セキュアになります。

ビット14 **SAP**：転送元割り当てポート

このビットはソース転送のマスタ･ポートの割り当てに使用されます。

0：ポート0（AHB）割り当て済み

1：ポート1（AHB）割り当て済み

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット13 **SBX**：各転送元のワードのアンアラインド･ハーフワード内での転送元のバイトを交換

転送元データ幅がワードより短い場合、このビットは無視されます。

転送元データ幅がワードの場合。

0：各転送元ワードのアンアラインド･ハーフワード内でのバイト･ベースの交換なし

1：各転送元ワードのアンアラインド･ハーフワード内の2つの連続したバイトが交換されます

ビット12:11 **PAM[1:0]**：パディング／配置モード

DDW_LOG2[1:0] = SDW_LOG2[1:0]の場合：バースト･デスティネーション転送とバースト･ソース転送のデータ幅が同じときには、このビットは無視されます。

それ以外の場合、次のエニュメレーションした値になり、条件PAM_1は転送先データ幅が転送元データ幅より大きい場合、条件PAM_2は転送元データ幅が転送先データ幅より大きい場合です。

条件：PAM_1

00：転送元データは、右寄せで、転送先データ幅まで0sでパディングされた値で転送されます

01：転送元データは、右寄せで、転送先データ幅まで符号拡張された値で転送されます

10-11：連続した転送元データをFIFOキューに入れ、左（LSB）から右（MSB）の順で転送先データ幅でパックしてから（リトル･エンディアンと呼ばれます）、デスティネーション転送します

条件：PAM_2

00：転送元データを、右寄せし、転送先データ幅に左側で切捨てをして転送します

01：転送元データを、左寄せし、転送先データ幅に右側で切捨てをして転送します

注： 10-11：転送元データを FIFO キューに入れ、転送先データ幅にアンパックし、左（LSB）から右（MSB）の順で転送します（リトル･エンディアンと呼ばれます）

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:4 **SBL_1[5:0]**：転送元のバースト長から1を引いた0から63の値。

バースト長の単位は、ビートと呼ばれるバーストのデータ1つ分です。SBL_1[5:0] = 0の場合、バーストはシングルと呼ばれることがあります。各データ／ビートの幅は転送先データ幅SDW_LOG2[1:0]で定義されます。

注： バースト転送がAHB転送の1KBのアドレス境界を越えた場合、GPDMAはAHBプロトコルに準拠するよう、プログラムされたバーストをシングルまたは、より短いバーストへと変更、短縮を行います。

バースト転送の長さがチャネルxのFIFOサイズより大きい場合、GPDMAはFIFOサイズに準拠するよう、プログラムされたバーストをシングルまたは、より短いバーストへと変更、短縮を行います。GPDMAの再アービトレーションが有効長とより短いシングル／バースト長の間で行われるため、転送効率は低いですが、データの整合性は保証されます。

ビット3 **SINC**：転送元インクリメント･バースト

0：固定バースト

1：隣接インクリメント･バースト

GPDMA_CxSARが指す転送元アドレスは、バースト･ビート／シングル転送後も保持されるか、隣接データに対応するオフセット分だけ転送後にインクリメントされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **SDW_LOG2[1:0]**：バーストの転送元データ幅の二進対数（バイト単位）

00：バイト

01：ハーフワード（2バイト）

10：ワード（4バイト）

11：ユーザ設定エラーが報告され、転送は行われませんでした

注： 8バイト･データ幅を設定すると、ユーザ設定エラーが報告され、転送は行われません。
転送元のブロック･サイズは転送元のデータ幅の倍数でなくてはなりません（GPDMA_CxBR1.BNDT[2:0]に対しSDW_LOG2[1:0]）。そうしないとユーザ設定エラーが報告され、転送は行われません。
ソース･バースト転送は、そのデータ幅による整列アドレスでなくてはなりません（開始アドレスGPDMA_CxSAR[2:0]に対してSDW_LOG2[1:0]）。そうしないとユーザ設定エラーが報告され、転送は行われまん。

17.8.11 GPDMAチャネルx転送レジスタ2（GPDMA_CxTR2）

アドレスオフセット：0x94 + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送を制御します。

このレジスタは、LGPDMA_CxCR.EN = 0の時に書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了したら（ハードウェアがGPDMA_CxCR.ENをデアサートしたら）、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.UT2 = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TCEM[1:0]		Res.	Res.	Res.	Res.	TRIGPOL[1:0]		Res.	TRIGSEL[6:0]						
rw	rw					rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGM[1:0]		Res.	Res.	BREQ	DREQ	SWREQ	Res.	Res.	REQSEL[6:0]						
rw	rw			rw	rw	rw			rw	rw	rw	rw	rw	rw	rw

ビット31:30 **TCEM[1:0]**：転送完了イベント･モード

このビットは転送完了と1/2転送完了イベント生成の粒度を規定します。

00：ブロック･レベル（GPDMA_CxBR1.BNDT[15:0] = 0のとき）：完全な転送（1/2転送）イベントがブロックの終わり（1/2の終わり）に生成されます。

注： 最初のLLI_0データ転送がnull／voidの（GPDMA_CxBR1.BNDT[15:0] = 0で内部のレジスタ･ファイルにより直接プログラムされる）場合、完全な転送イベントも1/2転送イベントも生成されません。

01：チャネルx（x = 0~11）、00同様チャネルx（x = 12 ~ 15）で、2D／繰返しブロック･レベル（GPDMA_CxBR1.BRC[10:0] = 0およびGPDMA_CxBR1.BNDT[15:0] = 0の場合）。完全な転送（および1/2転送）イベントが2D／繰返しブロックの最後（および1/2ブロックの最後）に生成されます。

注： 最初のLLI_0データ転送がnull／voidの（GPDMA_CxBR1.BNDT[15:0] = 0で内部のレジスタ･ファイルにより直接プログラムされる）場合、完全な転送イベントも1/2転送イベントも生成されません。

10：LLIレベル：LLIの更新（がある場合は）を含む、完全な転送イベントがLLI転送の最後に生成されます。LLIデータ転送の半分で1/2転送イベントが生成されます。データ転送がある場合、LLIデータ転送はチャネルx（x = 12~15）のブロック転送または2D／繰返しブロック転送です。

注： 最初のLLI_0データ転送がnull／voidの（内部のレジスタ･ファイルにより、GPDMA_CxBR1.BNDT[15:0] = 0で直接プログラムされている）場合、1/2転送イベントは生成されず、LLI_1のロードが完了したときに、転送完了イベントが生成されます。

11：チャネル･レベル：最後のLLI転送の終わりに、完全な転送イベントが生成されます。最後のLLIのデータ転送の半分で 1/2 転送イベントが生成されます。最後の LLI ではリンク･アドレスGPDMA_CxLLR.LA[15:2]が0に更新され、すべてのGPDMA_CxLLR更新ビットがクリアされます（UT1、UT2、UB1、USA、UDA、ULLに加え、UT3およびUB2）。チャネル転送が連続／無限の場合、イベントは生成されません。

ビット29:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **TRIGPOL[1:0]**：トリガ･イベント極性

このビットは、TRIGSEL[6:0]で定義される、選択したトリガ･イベント入力の極性を定義します。

00：トリガなし（マスク･トリガ･イベント）

01：立上がりエッジでのトリガ

10：立下がりエッジでのトリガ

11：00と同じ

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **TRIGSEL[6:0]**：トリガ･イベント入力の選択

このビットは、TRIGPOL[1:0]≠00の場合、有効なトリガ･イベントでGPDMA転送のトリガ･イベント入力を（セクション17.3.5に従って）選択します。

ビット15:14 **TRIGM[1:0]**：トリガ･モード

このビットはトリガにより、転送粒度の条件付けを定義します。

チャネルxがTRIGPOL[1:0] = 00または11で有効になっている（GPDMA_CxCR.ENがアサートされている）場合、TRIGM[1:0]ビットは無視されます。

その他の場合、GPDMA転送が最低1つのトリガ･ヒットにより条件付けされます。

00：ブロックレベル：最初の各ブロック転送のバースト読出しは、1つのヒット･トリガ（2D／繰返しブロックがGPDMA_CxBR1.BRC[10:0]≠0に設定された場合、各ブロックに対し、チャネルx = 12から15）により条件付けされます。

01：チャネルx（x = 0~11）は00と同じ。チャネルx（x = 12 ~ 15）は2D／繰返しブロック･レベル。2D／繰返しブロック転送の最初のバースト読出しは、1つのヒット･トリガで条件付けされます。

10：リンク･レベル：LLIリンク転送は1つのヒット･トリガにより条件付けされます。LLIデータ転送（がある場合）は条件付けされません。

11：プログラムされたバースト･レベルでSWREQ = 1の場合、プログラムされた各バースト読出しは1つのヒット･トリガにより条件付けされます。SWREQ = 0の場合、選択されたペリフェラルによりリクエストされたプログラムされたバーストは、それぞれ、1つのヒット･トリガにより条件付けされます。

-ペリフェラルがLLIデータ転送の転送元としてプログラムされている場合（DREQ = 0）、プログラムされたそれぞれのバースト読出しが条件付けされます。

-ペリフェラルがLLIデータ転送の転送先としてプログラムされている場合（DREQ = 1）、プログラムされたそれぞれのバースト書込みが条件付けされます。最初の準備されたFIFOベースのソース･バーストとも呼ばれる、（おそらく2D／繰返し）ブロックの最初のメモリ･バースト読出しは、ハードウェア･リクエストと最初のトリガ･ヒットの両方の発生によりゲートされます。

チャネルxに対するトリガのGPDMA監視が開始されるのは、新しいアクティブなトリガ設定、つまり選択したトリガの立上がり／立下がりエッジ（それぞれ、TRIGPOL[1:0] = 01またはTRIGPOL[1:0] = 10）によりそのチャネルが有効化／ロードされるときです。

トリガされ（データまたはリンクが）転送中で未完了の場合、このトリガの監視はアクティブのままとなり、新しいトリガが検出された場合、定義された立上がり／立下がりエッジとTRIGSEL[6:0]が変更されず、そのチャネルが有効である限り、このヒットは次の転送を許可するために内部的にメモリに書き込まれます。

TRIGSEL[6:0]またはTRIGPOL[1:0]の新しい値でGPDMA_CxTR2を更新する次のLLI_{n+1}の転送が、モニタリングをリセットし、前に定義したLLI_nトリガのメモリ書込みされたヒットを破棄します。

最初の新しいトリガhit_{n+1}を記憶した後、別の2つめのトリガhit_{n+2}を検出し、hit_nにトリガされた転送が完了していない場合、hit_{n+2}は失われ記憶されません。トリガ･オーバーラン･フラグが報告され（GPDMA_CxSR.TOF = 1）、イネーブル（GPDMA_CxCR.TOIE = 1）になると割込みが生成されます。トリガのオーバーランでハードウェアにより、チャネルが自動的に無効になることはありません。

注： **転送元ブロック･サイズがソース･バースト･サイズの倍数ではなく、転送元データ幅の倍数である場合、プログラムされた最後のソース･バーストは完了せず、ブロック･サイズに合うように内部で短縮されます。この場合、TRIGM[1:0] = 11でかつ(SWREQ = 1または(SWREQ = 0かつDREQ = 0))の場合、（シングルおよび／または、より短いバーストによる）短縮されたバースト転送は、トリガにより一旦、条件付けされます。**

プログラムされたデスティネーション･バーストは、シングルおよび／または、（FIFOサイズ、ブロックサイズ、1KBのアドレスの境界のはみ出しに対し）より短いバーストにより内部で短縮されます。トリガがプログラムされたデスティネーション･バーストを条件付けする（TRIGM[1:0] = 11かつSWREQ = 0かつDREQ = 1）場合、この短縮されたデスティネーション･バースト転送は、一旦トリガにより条件付けされます。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **BREQ**：ブロック･ハードウェア･リクエスト

チャネルxがSWREQ = 1（メモリ間転送のソフトウェア･リクエスト）で有効になっている場合（GPDMA_CxCR.ENがアサートされている場合）、このビットは無視されます。その他の場合：

0：選択されたハードウェア･リクエストが、バースト･レベルでのハードウェア･リクエスト／確認プロトコルでペリフェラルによりドライブされます。

1：選択されたハードウェア･リクエストが、ブロック･レベルでのハードウェア･リクエスト／確認プロトコルでペリフェラルによりドライブされます（セクション17.3.3を参照）。

ビット10 **DREQ**：転送先ハードウェア･リクエスト

チャネルxがSWREQ = 1（メモリ間転送のソフトウェア･リクエスト）で有効になっている場合（GPDMA_CxCR.ENがアサートされている場合）このビットは無視されます。その他の場合：

0：転送元のペリフェラルでドライブされる選択されたハードウェア･リクエスト（リクエスト信号が転送元／読出しポート経由でGPDMA転送スケジューラにより考慮されます）

1：転送先のペリフェラルでドライブされる選択されたハードウェア･リクエスト（リクエスト信号が転送先／書き出しポート経由でGPDMA転送スケジューラにより考慮されます）

注：

ビット9 **SWREQ**：ソフトウェア･リクエスト

このビットはGPDMA_CxCR.ENアサートされた場合、内部的に考慮されます。

0：ソフトウェア･リクエストは無し。選択されたハードウェア･リクエスト、REQSEL[6:0]、は考慮されます。

1：メモリ間転送のソフトウェア･リクエスト。REQSEL[6:0]に従ったデフォルトで選択されたハードウェア･リクエストは無視されます。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **REQSEL[6:0]**：GPDMAハードウェア･リクエストの選択

チャネルxがSWREQ = 1（メモリ間転送のソフトウェア･リクエスト）で有効になっている場合（GPDMA_CxCR.ENがアサートされている場合）、これらのビットは無視されます。その他の場合、選択されたハードウェア･リクエストはセクション17.3.3に従って内部的に考慮されます。

注意： **ユーザは同じ入力ハードウェア･リクエスト（同じREQSEL[6:0]値）を異なる有効なGPDMAチャネルに割り当ててはなりません（これらのチャネルにはGPDMA_CxCR.EN = 1およびGPDMA_CxTR2.SWREQ = 0を割り当てます）。GPDMAでは、同時に有効になっているチャネルが同じハードウェア･ペリフェラル･リクエスト信号により誤って設定されているケースを、ハードウェアがサポートすることは意図していません。また、ユーザ設定エラーは報告されません。**

17.8.12 GPDMAチャネルxブロック･レジスタ1（GPDMA_CxBR1）

アドレスオフセット：0x98 + 0x80 * x（x = 0から11）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送をブロック･レベルで制御します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルxが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時。

- GPDMA_CxLLR.UB1 = 1の場合、このレジスタは、メモリにある次のLLIからGPDMAにより、自動的に更新されます。
- GPDMA_CxLLR.UB1 = 0で、少なくとも1つのリンクリスト･レジスタがメモリの中の次のLLIから更新されることになっている場合、このレジスタは自動的、内部的にフィールドBNDT[15:0]のプログラムされた値で復元されます。
- すべての更新ビットGPDMA_CxLLR.Uxxがnullで、GPDMA_CxLLR.LA[15:0]≠0の場合、現在のLLIは最後の1つで、連続的に実行されます。このレジスタは、この最後の各LLIの実行後に、BNDT[15:0]にプログラムされた値で自動的、内部的に復元されます。
- GPDMA_CxLLR = 0の場合、このレジスタとBNDT[15:0]はnullのままとなり、チャネルxは完了します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BNDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BNDT[15:0]**：転送元から転送するデータ･バイトのブロック数

転送元から転送されてきたブロックのサイズ。チャネルが有効な場合、このフィールドは読出し専用となりデクリメントされます。これは現在の転送元ブロックにある転送する残りのデータ項目の数を示すものです。BNDT[15:0]はバイト数でプログラムされ、最大の転送元ブロック･サイズは64KB-1です。

一旦、最後のデータ転送が完了したとき（BNDT[15:0] = 0）。

- GPDMA_CxLLR.UB1 = 1の場合、このフィールドはメモリのLLIにより更新されます。
- GPDMA_CxLLR.UB1 = 0で、nullではないUxx更新ビットが最低1つある場合、このフィールドはプログラムされた値に内部的に復元されます。
- すべてのGPDMA_CxLLR.Uxxが0で、GPDMA_CxLLR.LA[15:0] = 0の場合、このフィールドはプログラムされた値（無限／連続の最後のLLI）に内部的に復元されます。
- GPDMA_CxLLR = 0の場合、このフィールドは最後のLLIデータ転送後に0に維持されます。

注： **nullではない転送元のブロック･サイズは、転送元のデータ幅の倍数でなくてはなりません（BNDT[2:0]に対しGPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。**

パッキング･モードに設定されている（GPDMA_CxTR1.PAM[1] = 1で転送先データ幅が転送元データ幅とは異なる）場合、（nullではない）転送元ブロック･サイズは、転送先データ幅の倍数でなくてはなりません（BNDT[2:0]に対しGPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

17.8.13 GPDMAチャネルxオルタネート･ブロック･レジスタ1（GPDMA_CxBR1）

アドレスオフセット：0x98 + 0x80 * x （x = 12から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送をブロック･レベルで制御します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルxが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時。

- GPDMA_CxLLR.UB1 = 1の場合、このレジスタは、メモリにある次のLLIからGPDMAにより、自動的に更新されます。
- GPDMA_CxLLR.UB1 = 0で、少なくとも1つのリンクリスト･レジスタがメモリの中の次のLLIから更新されることになっている場合、このレジスタは自動的、内部的にフィールドBNDT[15:0]およびBRC[10:0]のプログラムされた値で復元されます。
- すべての更新ビットGPDMA_CxLLR.Uxxがnullで、GPDMA_CxLLR.LA[15:0]≠0の場合、現在のLLIは最後の1つで、連続的に実行されます。このレジスタは、この最後のLLIの実行後に、フィールドBNDT[15:0]およびBRC[10:0]にプログラムされた値で自動的、内部的に復元されます。
- GPDMA_CxLLR = 0の場合、BNDT[15:0]およびBRC[10:0]はnullのままとなり、チャネルxは完了します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRDDEC	BRSDEC	DDEC	SDEC	Res.	BRC[10:0]										
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BNDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **BRDDEC**：ブロック繰返し転送先アドレスのデクリメント

0：ブロック転送の最後に、GPDMA_CxDARレジスタは現在のGPDMA_CxDAR値（現在の転送先アドレス）に、プログラムされたオフセットGPDMA_CxBR2.BRDAOを加えることで更新されます。

1：ブロック転送の最後に、GPDMA_CxDARレジスタは現在のGPDMA_CxDAR値（現在の転送先アドレス）からプログラムされたオフセットGPDMA_CxBR2.BRDAOを減算することで更新されます。

注： （BRDDEC に従った）このインクリメント／デクリメントに加え、GPDMA_CxDAR は同時にGPDMA_CxTR3.DAO値のインクリメント／デクリメント（DDECによる）によっても更新されます。これは通常、各プログラムされたバースト転送の最後に行われます。

ビット30 **BRSDEC**：ブロック繰返し転送元アドレスのデクリメント

0：ブロック転送の最後に、GPDMA_CxSARレジスタは現在のGPDMA_CxSAR値（現在の転送元アドレス）に、プログラムされたオフセットGPDMA_CxBR2.BRSAOを加えることで更新されます。

1：ブロック転送の最後に、GPDMA_CxSARレジスタは現在のGPDMA_CxSAR値（現在の転送元アドレス）から、プログラムされたオフセットGPDMA_CxBR2.BRSAOを減算することで更新されます。

注： （BRSDEC に従った）このインクリメント／デクリメントに加え、GPDMA_CxSAR は同時にGPDMA_CxTR3.SAO値のインクリメント／デクリメント（SDECによる）によっても更新されます。これは、あらゆるプログラムされたバースト転送の後に行われます。

ビット29 **DDEC**：転送先アドレス･デクリメント

0：転送先へのプログラムされたバースト転送の最後に、GPDMA_CxDARレジスタは現在のGPDMA_CxDAR値（現在の転送先アドレス）に、プログラムされたオフセットGPDMA_CxTR3.DAOを加算することで更新されます。

1：転送先へのプログラムされたバースト転送の最後に、GPDMA_CxDARレジスタは現在のGPDMA_CxDAR値（現在の転送先アドレス）から、プログラムされたオフセットGPDMA_CxTR3.DAOを減算することで更新されます。

ビット28 **SDEC**：転送元アドレス･デクリメント

0：転送元からのプログラムされたバースト転送の最後に、GPDMA_CxSAR レジスタは現在のGPDMA_CxSAR値（現在の転送元アドレス）に、プログラムされたオフセットGPDMA_CxTR3.SAOを加算することで更新されます。

1：転送元からのプログラムされたバースト転送の最後に、GPDMA_CxSAR レジスタは現在のGPDMA_CxSAR値（現在の転送元アドレス）から、プログラムされたオフセットGPDMA_CxTR3.SAOを減算することで更新されます。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26:16 **BRC[10:0]**:ブロック繰返しカウンタ

このフィールドには、現在のブロックの繰返し数が含まれます（0から2047）。

チャネルが有効になっている場合、このフィールドは読出し専用になります。デクリメント後、このフィールドは現在のブロックを除く、残りのブロックの数を示します。このカウンタはそれぞれの完了したブロック転送に対してハードウェアによりデクリメントされます。

一旦、最後のブロック転送が完了した場合（BRC[10:0]およびBNDT[15:0] = 0）。

– GPDMA_CxLLR.UB1 = 1の場合、全てのGPDMA_CxBR1フィールドはメモリの中の次のLLIにより更新されます。
– GPDMA_CxLLR.UB1 = 0で、nullではないUxx更新ビットが最低1つある場合、このフィールドはプログラムされた値に内部的に復元されます。
– すべてのGPDMA_CxLLR.Uxxが0で、GPDMA_CxLLR.LA[15:0]≠0の場合、このフィールドはプログラムされた値（無限／連続の最後のLLI）に内部的に復元されます。
– GPDMA_CxLLR = 0の場合、このフィールドは最後のLLIおよびデータ転送後に0に維持されます。

ビット15:0 **BNDT[15:0]**：転送元から転送するデータ･バイトのブロック数

転送元から転送されてきたブロックのサイズ。チャネルが有効な場合、このフィールドは読出し専用となりデクリメントされます。これは現在の転送元ブロックにある転送する残りのデータ項目の数を示すものです。BNDT[15:0]はバイト数でプログラムされ、最大の転送元ブロック･サイズは64KB-1です。

一旦、最後のデータ転送が完了したとき（BNDT[15:0] = 0）。

- GPDMA_CxLLR.UB1 = 1の場合、このフィールドはメモリのLLIにより更新されます。
- GPDMA_CxLLR.UB1 = 0で、nullではないUxx更新ビットが最低1つある場合、このフィールドはプログラムされた値に内部的に復元されます。
- すべてのGPDMA_CxLLR.Uxxが0で、GPDMA_CxLLR.LA[15:0]≠0の場合、このフィールドはプログラムされた値（無限／連続の最後のLLI）に内部的に復元されます。
- GPDMA_CxLLR = 0の場合、このフィールドは最後のLLIデータ転送後に0に維持されます。

注： nullではない転送元のブロック･サイズは、転送元のデータ幅の倍数でなくてはなりません（BNDT[2:0]に対しGPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

パッキング･モードに設定されている（GPDMA_CxTR1.PAM[1] = 1で転送先データ幅が転送元データ幅とは異なる）場合、（nullではない）転送元ブロック･サイズは、転送先データ幅の倍数でなくてはなりません（BNDT[2:0]に対しGPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

17.8.14 GPDMAチャネルx転送元アドレスレジスタ（GPDMA_CxSAR）

アドレスオフセット：0x9C + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは転送の転送元開始アドレスを設定します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

このレジスタはGPDMA_CxCR.EN = 1のときには読出し専用で、転送元からの次のバースト転送のアドレスを反映するように、ハードウェアにより継続的に更新されます。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、2D／繰返しブロック、LLI、あるいはフル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.USA = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SA[31:0]**：転送元アドレス

このフィールドは次のデータを読み込むアドレスを示すものです。

チャネルの操作に際し、このフィールドは、転送元のアドレス指定モード（GPDMA_CxTR1.SINC）に従い、各バースト転送の転送元データの後ろに固定、またはデータ幅（GPDMA_CxTR1.SDW_LOG2[1:0]）によりインクリメントし、データを読み出す次のアドレスを反映するようになっています。

チャネルの操作に際して、このアドレスは完了した各ソース･バーストの後に更新され、結果的に次のようになります。

– 固定アドレス指定モードまたは隣接データインクリメント･モードのプログラムされたソース･バースト。連続インクリメント（GPDMA_CxTR1.SINC = 1）の場合、追加のアドレス･オフセット値はプログラムされたバースト･サイズで、GPDMA_CxTR1.SBL_1[5:0]およびGPDMA_CxTR1.SDW_LOG2[21:0]で定義されます。

– GPDMA_CxBR1.SDECおよびGPDMA_CxTR3.SAO[12:0]でプログラムされた、追加の転送元のインクリメント／デクリメント･オフセット値。

– 転送元ブロック転送が完了した場合、2Dアドレス指定機能のあるチャネルx（x=12~15）では、GPDMA_CxBR1.BRSDECおよびGPDMA_CxBR2.BRSAO[15:0]でプログラムされた、追加のブロック繰返し転送元のインクリメント／デクリメント･オフセット値。

リンクリスト･モードでは、LLIがGPDMA_CxLLR.USA=1に設定されている場合、LLIデータ転送が完了した後、このレジスタがGPDMAによりメモリから自動的に更新されます。

注： 転送元アドレスは、プログラムされたソース･バーストのデータ幅に合わせなくてはなりません（SA[2:0]に対しGPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

転送元ブロック･サイズがソース･バースト･サイズの倍数ではなく、転送元データ幅の倍数である場合、最後のプログラムされたソース･バーストは完了せず、ブロック･サイズに合うように内部で短縮されます。この場合、追加のGPDMA_CxTR3.SAO[12:0]は適用されません。

17.8.15 GPDMAチャネルx転送先アドレス･レジスタ（GPDMA_CxDAR）

アドレスオフセット：0xA0 + 0x80 * x（x = 0から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは転送の転送先開始アドレスを設定します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

このレジスタはGPDMA_CxCR.EN = 1のときには読出し専用で、転送先への次のバースト転送のアドレスを反映するように、ハードウェアにより継続的に更新されます。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、2D／繰返しブロック、LLI、あるいはフル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.UDA = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DA[31:0]**：転送先アドレス

このフィールドは次のデータを書き出すアドレスを示すものです。

チャネルの操作に際し、このフィールドは転送先のアドレス指定モード（GPDMA_CxTR1.DINC）に従い､各バースト転送の転送先データの後ろに固定､またはデータ幅（GPDMA_CxTR1.DDW_LOG2[21:0]）によりインクリメントし、データを書き出す次のアドレスを反映するようになっています。

チャネルの操作に際して､このアドレスは完了した各デスティネーション･バーストの後に更新され､結果的に次のようになります。

– 固定アドレス指定モードまたは隣接データインクリメント･モードのプログラムされたデスティネーション･バースト。連続インクリメント（GPDMA_CxTR1.DINC = 1）の場合、追加のアドレス･オフセット値はプログラムされたバースト･サイズで､GPDMA_CxTR1.DBL_1[5:0]およびGPDMA_CxTR1.DDW_LOG2[1:0]で定義されます。

– GPDMA_CxBR1.DDECおよびGPDMA_CxTR3.DAO[12:0]でプログラムされた、追加の転送先のインクリメント／デクリメント･オフセット値。

– 転送先のブロック転送が完了した場合、2Dアドレス指定機能のあるチャネルx（x = 12~15）では、GPDMA_CxBR1.BRDDECおよびGPDMA_CxBR2.BRDAO[15:0]でプログラムされた追加のブロック繰返し転送先のインクリメント／デクリメント･オフセット値。

リンクリスト･モードでは、LLIがGPDMA_CxLLR.UDA = 1に設定されている場合、LLIデータ転送が完了した後、このレジスタがGPDMAによりメモリから自動的に更新されます。

注： 転送先アドレスは、プログラムされたデスティネーション･バーストのデータ幅に合わせなくてはなりません（DA[2:0]に対しGPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

17.8.16 GPDMAチャネルx転送レジスタ3（GPDMA_CxTR3）

アドレスオフセット：0xA4 + 0x80 * x （x = 12から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送を制御します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.UT3 = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	DAO[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	SAO[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:16 **DAO[12:0]**：転送先アドレス･オフセット･インクリメント

それぞれのプログラムされたデスティネーション･バーストのGPDMA_CxDARが指す転送先アドレスは、このオフセットDAO[12:0]により、（GPDMA_CxBR1.DDECに従って）インクリメントまたはデクリメントします。このオフセットは、完了したバーストがインクリメント･モード（GPDMA_CxTR1.DINC = 1）でアドレス指定される際に、プログラムされたバースト･サイズは含んでおらず、これに追加されるものです。

注： 転送先アドレス･オフセットは、プログラムされたデスティネーション･バーストのデータ幅に合わせなくてはなりません（DAO[2:0]に対しGPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **SAO[12:0]**：転送元アドレス･オフセット･インクリメント

それぞれのプログラムされたソース･バーストのGPDMA_CxSARが指す転送元アドレスは、このオフセットSAO[12:0]により、（GPDMA_CxBR1.SDECに従って）インクリメントまたはデクリメントします。このオフセットは、完了したバーストがインクリメント･モード（GPDMA_CxTR1.SINC = 1）でアドレス指定される際に、プログラムされたバースト･サイズは含んでおらず、これに追加されるものです。

注： 転送元アドレス･オフセットは、プログラムされたソース･バーストのデータ幅に合わせなくてはなりません（SAO[2:0]に対しGPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

転送元ブロック･サイズがデスティネーション･バースト･サイズの倍数ではなく、転送元データ幅の倍数である場合、最後のプログラムされたソース･バーストは完了せず、ブロック･サイズに合うように内部で短縮されます。この場合、追加のGPDMA_CxTR3.SAO[12:0]は適用されません。

17.8.17 GPDMAチャネルxブロック･レジスタ2（GPDMA_CxBR2）

アドレスオフセット：0xA8 + 0x80 * x （x = 12から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送を2D／繰返しブロック･レベルで制御します。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、2D／繰返しブロック、LLI、あるいはフル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.UB2 = 1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRDAO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRSAO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **BRDAO[15:0]**：ブロック繰返し転送先アドレスのオフセット

2Dアドレス指定機能があるチャネルのこのフィールドは、ブロック転送の最後に現在の転送先アドレス（GPDMA_CxDAR）を（GPDMA_CxBR1.BRDDECに従って加算または減算することで）更新するために使用されます。

注： ブロックが繰り返されている転送先アドレス･オフセットは、プログラムされたデスティネーション･バーストのデータ幅に合わせなくてはなりません（BRDAO[2:0]に対しGPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

ビット15:0 **BRSAO[15:0]**：ブロック繰返し転送元アドレスのオフセット

2Dアドレス指定機能があるチャネルのこのフィールドは、ブロック転送の最後に現在の転送元アドレス（GPDMA_CxSAR）を（GPDMA_CxBR1.BRSDECに従って加算または減算することで）更新するために使用されます。

注： ブロック繰返し転送元アドレス･オフセットは、プログラムされたソース･バーストのデータ幅に合わせなくてはなりません（BRSAO[2:0]に対しGPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

17.8.18 GPDMAチャネルxのリンクリスト･アドレス･レジスタ（GPDMA_CxLLR）

アドレスオフセット：0xCC + 0x80 * x（x = 0から11）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、メモリの次のLLIのデータ構造と、そのアドレス･ポインタの設定を行います。このレジスタがnullの場合、チャネル転送は完了です。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.ULL=1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UT1	UT2	UB1	USA	UDA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL
rw	rw	rw	rw	rw											rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LA[15:2]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット31 **UT1**：メモリからGPDMA_CxTR1を更新

このビットはリンク転送時にメモリからのGPDMA_CxTR1の更新を制御します。

0：GPDMA_CxTR1の更新はなし

1：GPDMA_CxTR1を更新

ビット30 **UT2**：メモリからGPDMA_CxTR2を更新

このビットはリンク転送時にメモリからのGPDMA_CxTR2の更新を制御します。

0：GPDMA_CxTR2の更新はなし

1：GPDMA_CxTR2を更新

ビット29 **UB1**：メモリからGPDMA_CxBR1を更新

このビットはリンク転送時にメモリからのGPDMA_CxBR1の更新を制御します。

UB1 = 0でGPDMA_CxLLR ≠ 0の場合、リンクリストは完了していません。GPDMA_CxBR1.BNDT[15:0]は、データ転送の完了後、リンク転送の前にプログラムされた値に復元されます。

0：メモリからのGPDMA_CxBR1の更新は無し（リンク転送が何かあった場合は、GPDMA_CxBR1.BNDT[15:0]が復元されます）

1：GPDMA_CxBR1を更新

ビット28 **USA**：メモリからGPDMA_CxSARを更新

このビットはリンク転送時にメモリからのGPDMA_CxSARの更新を制御します。

0：GPDMA_CxSAR更新はなし

1：GPDMA_CxSAR更新

ビット27 **UDA**：メモリからGPDMA_CxDARレジスタを更新

このビットはリンク転送時にメモリからのGPDMA_CxDARの更新の制御に使用されます。

0：GPDMA_CxDARの更新はなし

1：GPDMA_CxDARを更新

ビット26:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **ULL**：メモリからGPDMA_CxLLRレジスタを更新

このビットはリンク転送時にメモリからのGPDMA_CxLLRの更新の制御に使用されます。

0：GPDMA_CxLLR更新はなし

1：GPDMA_CxLLR更新

ビット15:2 **LA[15:2]**：次のリンクリスト･データ構造へのポインタ（16bitの下位アドレス）

UT1 = UT2 = UB1 = USA = UDA = ULL = 0でLA[15:20] = 0の場合、現在のLLIが最後です。チャネル転送が、リンクリストGPDMAレジスタ･ファイルの更新が一切なく完了します。

それ以外の場合、このフィールドは、データ転送が一旦完了すると、リンクリストGPDMA内部レジスタ･ファイル（GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、GPDMA_CxLLR）を条件付きで更新するため、次のリンクリスト･データ構造を自動的にフェッチする、メモリ･アドレス･オフセットへのポインタです。

注： ユーザはこのポインタを32bitに合わせてプログラムしなくてはなりません。下位2bitの書込みは無視されます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

17.8.19 GPDMAチャネルxオルタネート･リンクリスト･アドレス･レジスタ（GPDMA_CxLLR）

アドレスオフセット：0xCC + 0x80 * x （x = 12から15）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア･ステート（GPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権ステート（GPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、メモリの次のLLIのデータ構造と、そのアドレス･ポインタの設定を行います。このレジスタがnullの場合、チャネル転送は完了です。

GPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

GPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了し（その後、ハードウェアがGPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードでは、リンク転送時、GPDMA_CxLLR.ULL=1の場合、このレジスタはGPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UT1	UT2	UB1	USA	UDA	UT3	UB2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL
rw	rw	rw	rw	rw	rw	rw									rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LA[15:2]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット31 **UT1**：メモリからGPDMA_CxTR1を更新
このビットはリンク転送時にメモリからのGPDMA_CxTR1の更新を制御します。
0：GPDMA_CxTR1の更新はなし
1：GPDMA_CxTR1を更新

ビット30 **UT2**：メモリからGPDMA_CxTR2を更新
このビットはリンク転送時にメモリからのGPDMA_CxTR2の更新を制御します。
0：GPDMA_CxTR2の更新はなし
1：GPDMA_CxTR2を更新

ビット29 **UB1**：メモリからGPDMA_CxBR1を更新
このビットはリンク転送時にメモリからのGPDMA_CxBR1の更新を制御します。
UB1 = 0でGPDMA_CxLLR ≠ 0の場合､リンクリストは完了していません｡GPDMA_CxBR1.BNDT[15:0]は、データ転送の完了後、リンク転送の前にプログラムされた値に復元されます。
0：メモリからのGPDMA_CxBR1の更新は無し（リンク転送が何かあった場合は、GPDMA_CxBR1.BNDT[15:0]が復元されます）
1：GPDMA_CxBR1を更新

ビット28 **USA**：メモリからGPDMA_CxSARを更新
このビットはリンク転送時にメモリからのGPDMA_CxSARの更新を制御します。
0：GPDMA_CxSAR更新はなし
1：GPDMA_CxSAR更新

ビット27 **UDA**：メモリからGPDMA_CxDARレジスタを更新
このビットはリンク転送時にメモリからのGPDMA_CxDARの更新の制御に使用されます。
0：GPDMA_CxDARの更新はなし
1：GPDMA_CxDARを更新

ビット26 **UT3**：メモリからGPDMA_CxTR3を更新
このビットはリンク転送時にメモリからのGPDMA_CxTR3の更新を制御します。
0：GPDMA_CxTR3の更新はなし
1：GPDMA_CxTR3を更新

ビット25 **UB2**：メモリからGPDMA_CxBR2を更新
このビットはリンク転送時にメモリからのGPDMA_CxBR2の更新を制御します。
0：GPDMA_CxBR2の更新はなし
1：GPDMA_CxBR2を更新

ビット24:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **ULL**：メモリからGPDMA_CxLLRレジスタを更新
このビットはリンク転送時にメモリからのGPDMA_CxLLRの更新の制御に使用されます。
0：GPDMA_CxLLR更新はなし
1：GPDMA_CxLLR更新

ビット15:2 **LA[15:2]**：次のリンクリスト･データ構造へのポインタ（16bitの下位アドレス）
UT1 = UT2 = UB1 = USA = UDA = ULL = 0でLA[15:20] = 0の場合、現在のLLIが最後です。チャネル転送が、リンクリストGPDMAレジスタ･ファイルの更新が一切なく完了します。
それ以外の場合、このフィールドは、データ転送が一旦完了すると、リンクリストGPDMA内部レジスタ･ファイル（GPDMA_CxTR1、GPDMA_CxTR2、GPDMA_CxBR1、GPDMA_CxSAR、GPDMA_CxDAR、GPDMA_CxLLR）を条件付きで更新するため、次のリンクリスト･データ構造を自動的にフェッチする、メモリ･アドレス･オフセットへのポインタです。

注： ユーザはこのポインタを32bitに合わせてプログラムしなくてはなりません。下位2bitの書込みは無視されます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

17.8.20 GPDMAレジスタマップ

表143.GPDMAレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	GPDMA_ SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	GPDMA_ PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPDMA_ RCFGLOCKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK15	LOCK14	LOCK13	LOCK12	LOCK11	LOCK10	LOCK9	LOCK8	LOCK7	LOCK6	LOCK5	LOCK4	LOCK3	LOCK2	LOCK1	LOCK0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPDMA_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS15	MIS14	MIS13	MIS12	MIS11	MIS10	MIS9	MIS8	MIS7	MIS6	MIS5	MIS4	MIS3	MIS2	MIS1	MIS0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPDMA_SMISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS15	MIS14	MIS13	MIS12	MIS11	MIS10	MIS9	MIS8	MIS7	MIS6	MIS5	MIS4	MIS3	MIS2	MIS1	MIS0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14 - 0x4C	予約済み	予約済み																															
0x50 + 0x80 * x（x = 0 ~ 15）	GPDMA_ CxLBAR	LBA[31:16]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
0x5C + 0x80 * x（x = 0 ~ 15）	GPDMA_CxFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																		0	0	0	0	0	0	0								
0x60 + 0x80 * x（x = 0 ~ 15）	GPDMA_CxSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFOL[7:0]								Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDLEF
	リセット値									0	0	0	0	0	0	0	0		0	0	0	0	0	0	0								1
0x64 + 0x80 * x（x = 0 ~ 15）	GPDMA_CxCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIO [1:0]		Res.	Res.	Res.	Res.	LAP	LSM	Res.	TOIE	SUSPIE	USEIE	ULEIE	DTEIE	HTIE	TCIE	Res.	Res.	Res.	Res.	Res.	SUSP	リセット	EN
	リセット値									0	0					0	0		0	0	0	0	0	0	0						0	0	0
0x90 + 0x80 * x（x = 0 ~ 15）	GPDMA_CxTR1	DSEC	DAP	Res.	Res.	DHX	DBX	DBL_1 [5:0]						DINC	Res.	DDW_LOG2 [1:0]		SSEC	SAP	SBX	PAM[1:0]		Res.	SBL_1 [5:0]						SINC	Res.	SDW_LOG2 [1:0]	
	リセット値	0	0			0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0
0x94 + 0x80 * x（x = 0 ~ 15）	GPDMA_CxTR2	TCEM[1:0]		Res.	Res.	Res.	Res.	TRIGPOL [1:0]		Res.	TRIGSEL[6:0]							TRIGM [1:0]		Res.	Res.	BREQ	DREQ	SWREQ	REQSEL[7:0]								
	リセット値	0	0					0	0		0	0	0	0	0	0	0	0	0			0	0	0	0	0	0		0	0	0	0	0
0x98 + 0x80 * x（x = 0 ~ 11）	GPDMA_CxBR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BNDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x98+0x80 * x（x = 12 ~ 15）	GPDMA_CxBR1	BRDDEC	BRSDEC	DDEC	SDEC	Res.	BRC[10:0]											BNDT[15:0]															
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x9C + 0x80 * x（x = 0 ~ 15）	GPDMA_CxSAR	SA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA0 + 0x80 * x（x = 0 ~ 15）	GPDMA_CxDAR	DA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表143.GPDMAレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xA4 + 0x80 * x（x = 12 ~ 15）	GPDMA_CxTR3	Res.	Res.	Res.	DAO[12:0]													Res.	Res.	Res.	SAO[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0xA8 + 0x80 * x（x = 12 ~ 15）	GPDMA_CxBR2	BRDAO[15:0]																BRSAO[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xCC + 0x80 * x（x = 0 ~ 11）	GPDMA_CxLLR	UT1	UT2	UB1	USA	UDA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL	LA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0xCC + 0x80 * x（x = 12 ~ 15）	GPDMA_CxLLR	UT1	UT2	UB1	USA	UDA	UT3	UB2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL	LA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

レジスタ境界アドレスについては、セクション2.3を参照してください。

18 低電力ダイレクトメモリアクセスコントローラ（LPDMA）

18.1 LPDMAの概要

低電力ダイレクト·メモリ·アクセス（LPDMA）コントローラは、バス·マスタとシステム·ペリフェラルで構成されます。

LPDMAを使用して、CPUを介さず、リンクリストを介してメモリマップペリフェラルやメモリの間でプログラム可能なデータ転送を実行します。

18.2 LPDMAの主な機能

- シングル双方向AHBマスタ
- メモリマップされたデータの転送元から転送先への転送。
  - ペリフェラルからメモリ
  - メモリからペリフェラル
  - メモリ間転送
  - ペリフェラル間
- SLEEPおよびSTOPモード中の自律データ転送
- チャネルレベルで4グレードでプログラムされた優先順位に基づく転送アービトレーション：
  - タイミング重視のチャネルでの、優先順位が高いトラフィック·クラス（キュー 3）。
  - タイミング重視でないチャネルでの加重ラウンドロビンが割り当てられた、3つの優先順位の低いトラフィッククラス（キュー 0、1、2）
- 次のいずれかのイベントに対する、チャネルごとのイベント生成：転送完了、1/2転送完了、データ転送エラー、ユーザ設定エラー、リンク転送エラー、中断完了、トリガ·オーバーラン
- イベントごとに別々にプログラムされた割込みイネーブルでの、チャネルごとの割込み生成
- 4個の並行LPDMAチャネル。
  - 2つの実行モード、つまりrun-to-completionモードおよびリンク·ステップ·モードに対応する、プログラム可能なリンクリストを介してメモリに連結するチャネル内LPDMA転送
  - プログラム可能なLPDMA入力を経由するチャネル内およびチャネル間LPDMA転送チェーンは、LPDMAタスク完了イベントへの接続をトリガします。
- チャネル内の各リンクリスト·アイテムごとに次の操作を行います。
  - 別々にプログラムされた転送元／転送先の転送
  - 転送元／転送先間でのプログラム可能データの処理：バイトベースのパディングまたは切り詰め、符号拡張、および左詰め／右詰め
  - ブロックレベルを定義する、転送元から転送するプログラム可能なデータバイトの数
  - リニアな転送元および転送先のアドレス指定：固定または連続インクリメント·アドレス指定、ブロックレベルで、連続したシングル転送間でプログラム
  - プログラム可能なLPDMAリクエストとトリガ選択
  - プログラム可能なLPDMA1/2転送および転送完了イベントの生成
  - LPDMAリンクリスト制御レジスタの自動更新が行われる、メモリ内での次のリンクリスト·アイテムとそのデータ構造へのポインタ

- DEBUG：
  - チャネル･サスペンドおよび再開のサポート
  - チャネル状態のレポートとイベント･フラグ
- TrustZoneサポート
  - 最初のチャネルレベルや転送元／転送先、リンク･サブレベルに関係なく、セキュア／非セキュアLPDMA転送をサポートします。
  - それぞれいずれかのセキュア／非セキュアなチャネルから発行される、セキュア／非セキュアな割込みレポート
  - TrustZoneを認識するAHBスレーブ･ポートが、すべてのLPDMAセキュア･リソース（レジスタ、レジスタ･フィールド）を非セキュア･アクセスから保護します。
- 特権／非特権をサポート
  - チャネル･レベルで独立に、特権および非特権LPDMA転送をサポート
  - 特権を認識するAHBスレーブポート。

18.3 LPDMAの実装

18.3.1 LPDMAチャネル

特定のLPDMAチャネルxは、以下の表に示すハードウェア･パラメータにより実装されます。

表 144.LPDMA1チャネル実装

チャネルx	ハードウェア･パラメータ		機能
	dma_fifo_size[x]	dma_addressing[x]	
x = 0 ~ 3	0	0	チャネルx（x = 0から3）を次のとおり実装します。 – FIFOなし。内部で登録されているのは、単一ソースの転送セルのみです。 – 固定／連続インクリメント･アドレス指定

18.3.2 低電力モードでのLPDMA自律モード

LPDMA自律モードとウェイクアップ機能が、下の表に示すようにデバイスの低電力モードで実装されています。

表145.低電力モードでのLPDMA1自律モードとウェイクアップ

機能	低電力モード
自律モードとウェイクアップ	SLEEP、STOP0、STOP1、およびSTOP2モードのLPDMA1

18.3.3 LPDMAリクエスト

ペリフェラルからのLPDMAリクエストは、SWREQ = 0の場合、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）のREQSEL[4:0]でLPDMAチャネルxに割り当てることが可能です。

LPDMAリクエスト･マッピングを下の表に示します。

表146.プログラムされたLPDMA1リクエスト

LPDMA_CxTR2.REQSEL[4:0]	選択されるLPDMAリクエスト
0	lpuart1_rx_dma
1	lpuart1_tx_dma
2	spi3_rx_dma
3	spi3_tx_dma
4	i2c3_rx_dma
5	i2c3_tx_dma
6	i2c3_evc_dma
7	adc4_dma
8	dac1_ch1_dma
9	dac1_ch2_dma
10	adf1_flt0_dma
11	lptim1_ic1_dma
12	lptim1_ic2_dma
13	lptim1_ue_dma
14	lptim3_ic1_dma
15	lptim3_ic2_dma
16	lptim3_ue_dma

18.3.4 LPDMAブロック･リクエスト

LPDMAリクエストによっては、シングル･リクエストではなくブロック･リクエストとしてプログラムする必要があるものがあります。その後、ハードウェア･レベルでの、リクエストされたペリフェラル転送の正しいLPDMAの実行のために、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）のBREQビットを設定する必要があります。

LPDMAブロック･リクエストを下の表に示します。

表147.ブロック･リクエストとしてプログラムされたLPDMA1リクエスト

LPDMAブロック･リクエスト
lptim1_ue_dma
lptim3_ue_dma

18.3.5 LPDMAトリガ

選択したトリガの立上がりエッジまたは立下りエッジが TRIGPOL[1:0] で定義されている場合（TRIGPOL[1:0] = 01またはTRIGPOL[1:0] = 10）、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）のTRIGSEL[4:0]フィールドを介してLPDMAチャネルxにLPDMAトリガを割り当てることができます。

LPDMAトリガ･マッピングを下の表に示します。

表148.プログラムされたLPDMA1トリガ

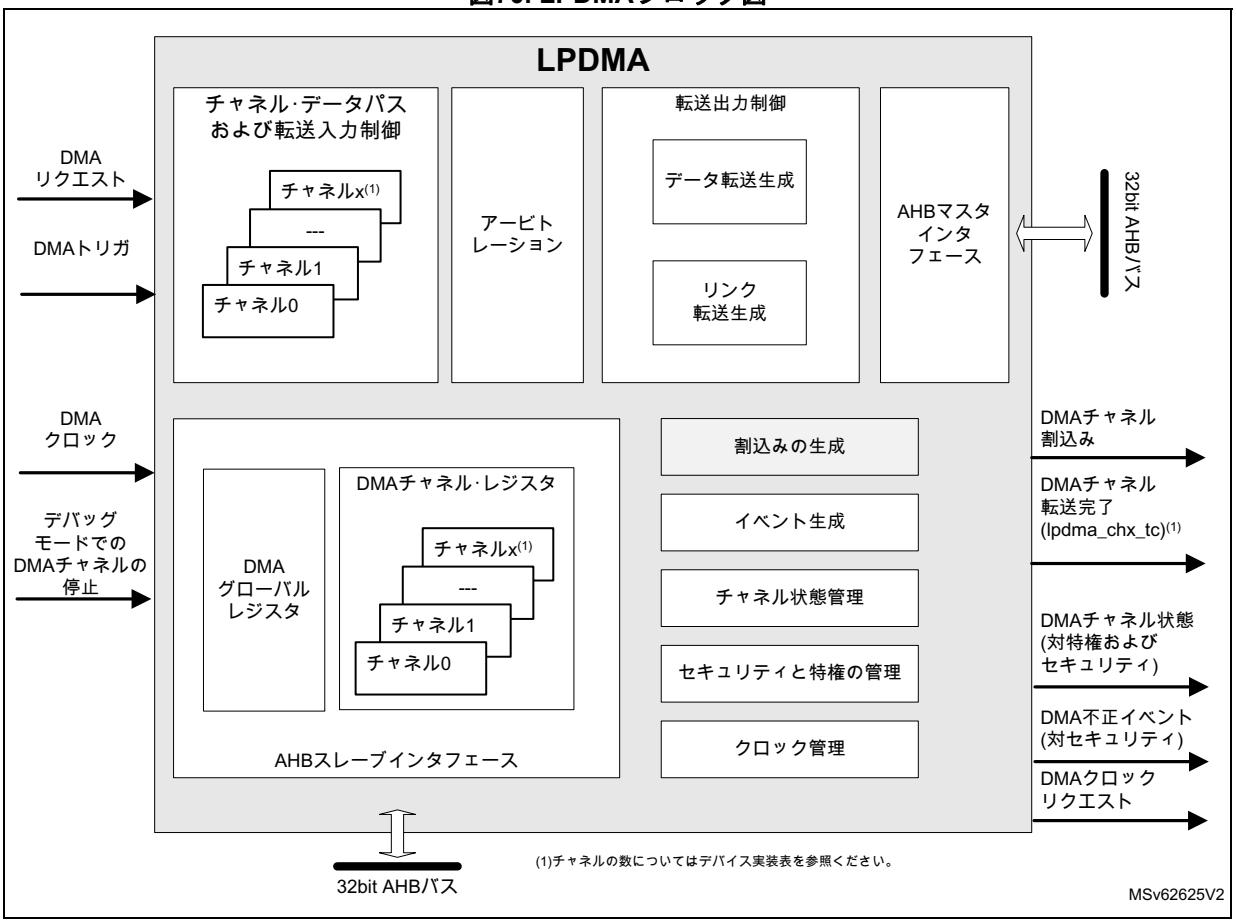
LPDMA_CxTR2.TRIGSEL[4:0]	選択されるLPDMAトリガ
0	exti0
1	exti1
2	exti2
3	exti3
4	exti4
5	tamp_trg1
6	tamp_trg2
7	tamp_trg3
8	lptim1_ch1
9	lptim1_ch2
10	lptim3_ch1
11	lptim4_out
12	comp1_out
13	comp2_out
14	rtc_alra_trg
15	rtc_alrb_trg
16	rtc_wut_trg
17	adc4_awd1
18	lpdma1_ch0_tc
19	lpdma1_ch1_tc
20	lpdma1_ch2_tc
21	lpdma1_ch3_tc
22	gpdma1_ch0_tc
23	gpdma1_ch1_tc
24	gpdma1_ch4_tc
25	gpdma1_ch5_tc
26	gpdma1_ch6_tc
27	gpdma1_ch7_tc
28	gpdma1_ch12_tc
29	gpdma1_ch13_tc
30	tim2_trgo
31	tim15_trgo

18.4 LPDMAの機能説明

18.4.1 LPDMAブロック図

以下にLPDMAブロック図を示します。

図70. LPDMAブロック図



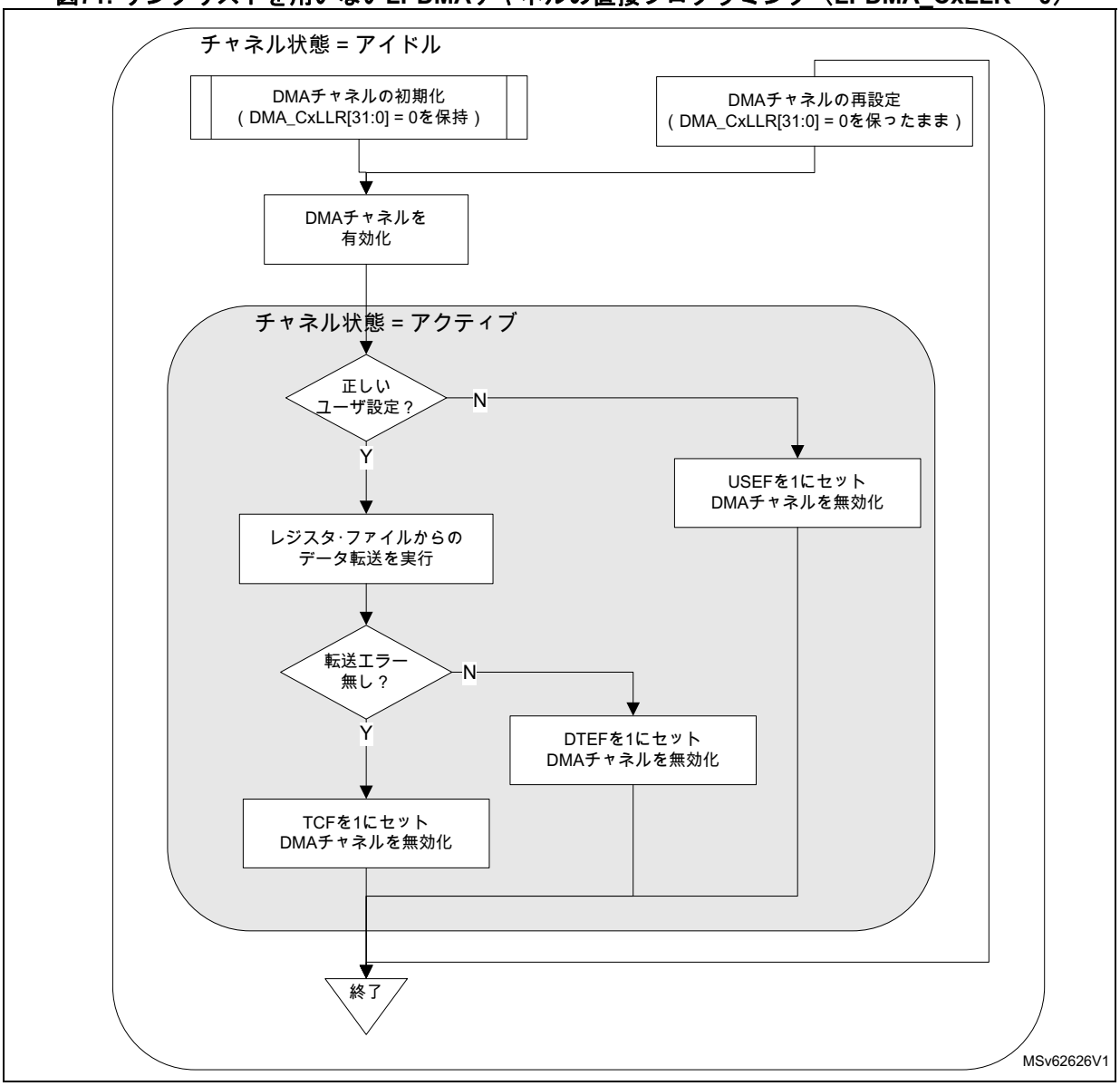
18.4.2 LPDMAチャネル状態とリンクリストなしの直接プログラミング

LPDMAリセット後、LPDMAチャネルxはアイドル状態となります。ソフトウェアがLPDMA_CxCR.ENに1を書き込むと、制御ビットが有効になり、チャネルで別のチャネル設定レジスタ（LPDMA_CxXXX）の値が考慮され、アクティブ／非アイドル状態に切り替わり、リクエストされた対応するデータ転送の実行が開始されます。

LPDMA_CxCR.ENビットに1を書き込むことでLPDMAチャネル転送を有効化／開始した後、転送完了時のLPDMAチャネル割込みにより、LPDMAチャネルがアイドル状態に戻り（その後、ENがハードウェアによりデアサートされ）、再設定と再有効化の準備が整っていることをソフトウェアに通知します。

下図に、リンクリストなしのこのLPDMAの直接プログラミングを示します（LPDMA_CxLLR = 0）。

図71. リンクリストを用いないLPDMAチャネルの直接プログラミング（LPDMA_CxLLR = 0）



18.4.3 LPDMAチャネルのサスペンドとレジューム

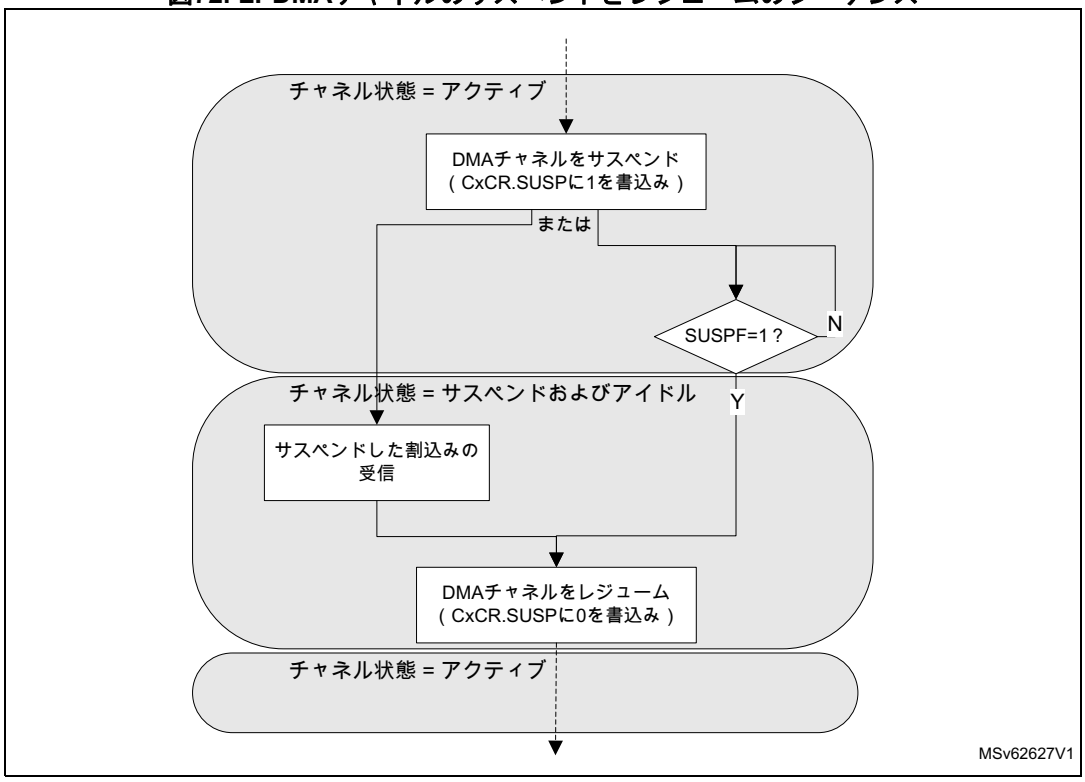
次の手順により、有効になっているチャネルをソフトウェアのみでサスペンドすることが可能です。

1. ソフトウェアでLPDMA_CxCR.SUSPビットに1を書き込みます。
2. ソフトウェアは、SUSPF = 1になるまで、サスペンドしたフラグLPDMA_CxSR.SUSPFにポーリングするか、LPDMA_CxCR.SUSPIEに1を書き込むことにより前に有効化された割込みを待ちます。チャネルが効果的にサスペンド状態になるまで待つということは、マスタポート上で実行中のLPDMA転送の完了を待つということです。その後ソフトウェアは定常状態で、あらゆる読出しレジスタ、またはハードウェアによる変更が可能なレジスタ･フィールドを観察することが可能になります。

   実行中のLPDMA転送は、データ転送（読出しとそれに続く書込みのシングル転送）あるいは、次のリンクリスト･アイテムからのリンクリスト･レジスタ･ファイルの内部更新のためのリンク転送である可能性があることに注意してください。
3. ソフトウェアは、LPDMA_CxCR.SUSPに0を書き込むことで、サスペンドしたチャネルを安全にレジュームします。

下図に、サスペンドとレジュームのシーケンスを示します。

図72. LPDMAチャネルのサスペンドとレジュームのシーケンス



注： **サスペンドとレジュームのシーケンスはLPDMA_CxCR.EN ビットには影響しません。チャネル（転送）のサスペンドが、開始したトリガ検出をサスペンドすることはありません。**

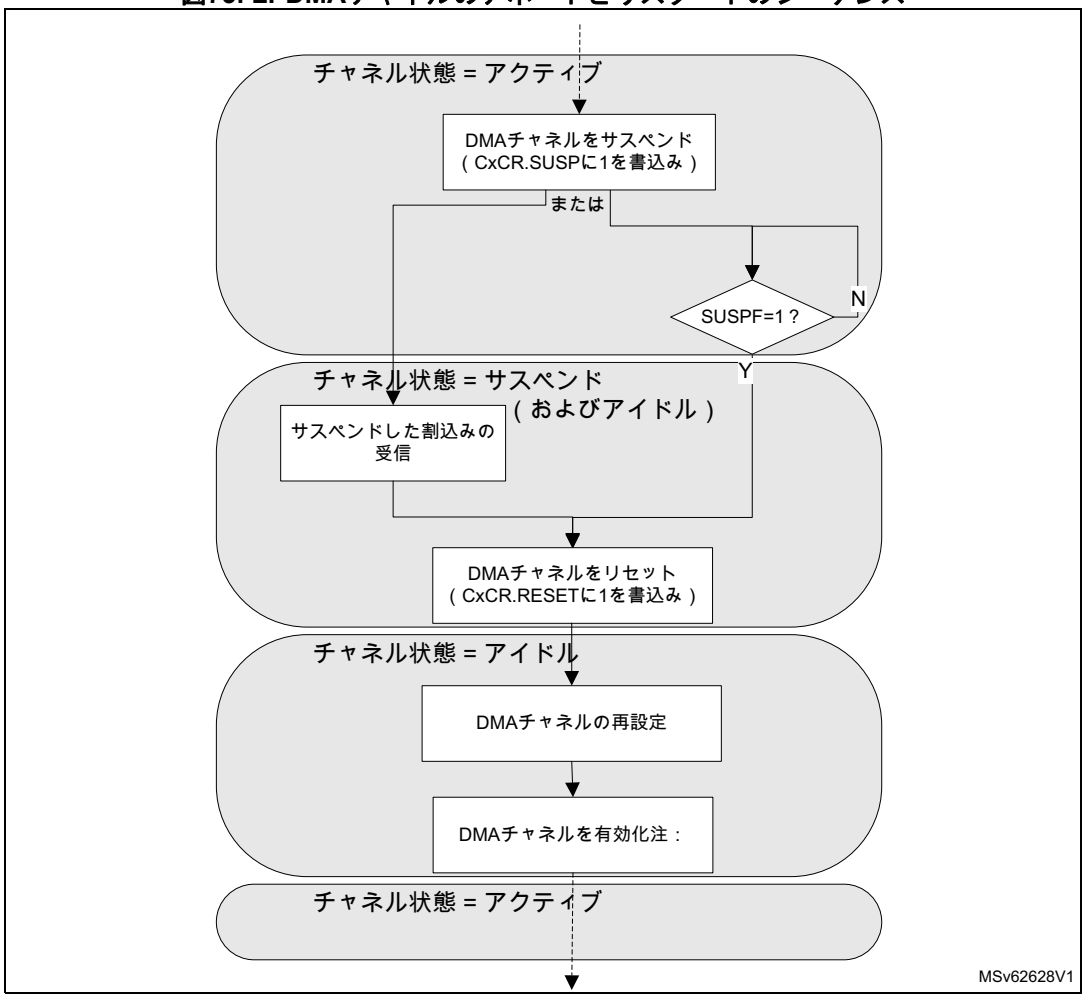
18.4.4 LPDMAチャネルのアボートとリスタート

サーキュラ·バッファまたはダブル·バッファによる連続LPDMA転送のアボート同様、以下に示すシーケンスにより、現在も有効なチャネルをソフトウェアのみでアボートすることも可能です。

1. ソフトウェアでLPDMA_CxCR.SUSPビットに1を書き込みます。
2. ソフトウェアは、SUSPF = 1になるまで、サスペンドしたフラグLPDMA_CxSR.SUSPFにポーリングするか、LPDMA_CxCR.SUSPIEに1を書き込むことにより前に有効化された割込みを待ちます。チャネルが効果的にサスペンド状態になるまで待つということは、マスタポート上で実行中のLPDMA転送の完了を待つということです。
3. ソフトウェアはLPDMA_CxCR.RESETに1を書き込むことにより、チャネルをリセットします。これにより、チャネルの内部状態がリセットされ、LPDMA_CxCR.ENビットとLPDMA_CxCR.SUSPビットがリセットされます。
4. ソフトウェアがチャネルを安全に再設定します。ソフトウェアは、ハードウェアにより変更されたLPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDARレジスタを再プログラムする必要があります。
5. ここでアボートしてから再プログラムされたチャネルをリスタートするために、ソフトウェアでLPDMA_CxCR.ENビットに1を書き込んで有効にします。

下図にアボートとリスタートシーケンスを示します。

図73. LPDMAチャネルのアボートとリスタートのシーケンス



18.4.5 LPDMAのリンクリスト･データ構造

直接プログラミング･モードの他、チャネルは、リンクリスト･アイテム（LLI）のリストとして知られている転送リストによりプログラムすることが可能です。各LLIはそのデータ構造により定義されます。

チャネルxの次のLLI_{n+1}のデータ構造を持つメモリのベース･アドレスは以下の合計です。

- （LPDMA_CxLBARでの）チャネルxのリンク･ベース･アドレス
- リンク･アドレス･オフセット（LPDMA_CxLLRのLA[15:2]フィールド）。

各LLIのデータ構造は、それぞれ異なる場合があります。

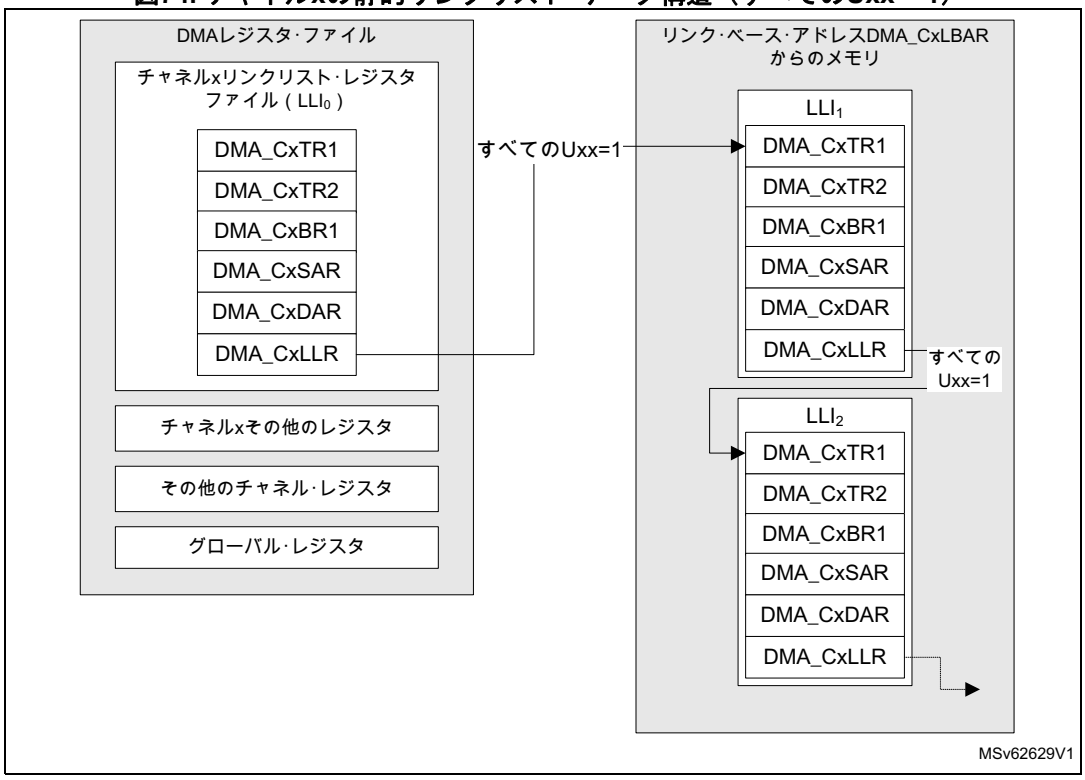
リンクリストのデータ構造は、LPDMA_CxLLRレジスタのUT1、UT2、UB1、USA、UDA、ULLビットの値に従ってアドレス指定されます。

リンクリスト･モードでは、LPDMAリンクリスト･レジスタ（LPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDARまたはLPDMA_CxLLR）は、条件付きで自動的にメモリ内の次のリンクリスト･データから、LPDMA_CxLLRレジスタの現在の値に従って更新されます。このLPDMA_CxLLRレジスタの現在の値は、前のLLIのリンクリスト･データ構造から条件付きで更新されたものです。

静的リンクリスト･データ構造

たとえば、LPDMA_CxLLRレジスタの更新ビット（UT1、UT2、UB1、USA、UDA、およびULL）がすべてアサートされている場合、メモリ内のリンクリスト･データ構造は最大で6個の連続32bit位置となります。ここにはLPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDAR、LPDMA_CxLLRが含まれ（下図を参照）、メモリ内の最初のリンクリスト･レジスタ･ファイル（LLI_0）とそれ以降のLLI（LLI_1、LLI_2、...）が含まれます。

図74. チャネルxの静的リンクリスト･データ構造（すべてのUxx = 1）



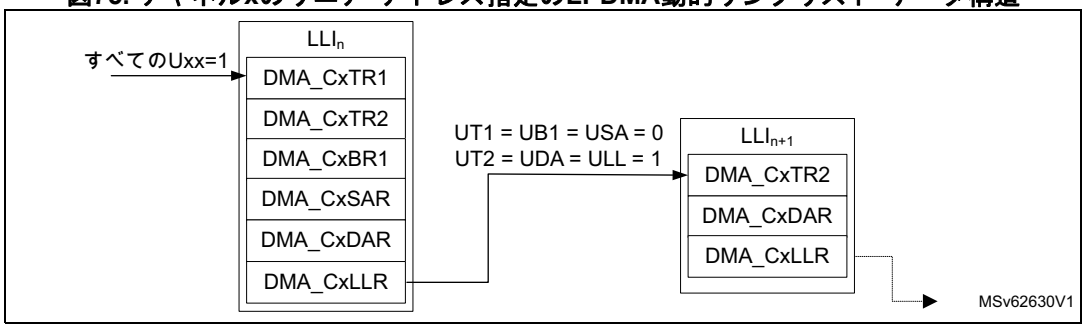
動的リンクリスト･データ構造

LLIの全リストのメモリ構成は、各LLI特有のデータ構造でコンパクト化が可能です。

UT1 = 0かつUT2 = 1の場合、レジスタLPDMA_CxLLRのリンク･アドレス･オフセットは、変更されないLPDMA_CxTR1の代わりに、LPDMA_CxTR2の更新された値を指します。

例:UT1 = UB1 = USA = 0かつUT2 = UDA = ULL = 1の場合､次のLLIにはLPDMA_CxTR1､LPDMA_CxBR1､LPDMA_CxSARのいずれの更新された値も含まれません。下図に示すように、その次のLLIには、LPDMA_CxTR2、LPDMA_CxDAR、LPDMA_CxLLRの更新された値が含まれます。

図75. チャネルxのリニア･アドレス指定のLPDMA動的リンクリスト･データ構造



ユーザは各LLI_nのLPDMA_CxLLRが32bitに整列し､LPDMA_CxLBARが指す64KBのアドレス空間を超えないようにプログラムする必要があります。

18.4.6 リンクリスト･アイテムの転送実行

LLI_n転送は次のシーケンスです。

1. データ転送：LPDMAは、LPDMA内部レジスタ･ファイルによって記述されたとおりにデータ転送を実行します（このデータ転送は、LLI_0ではvoid/nullにすることができます）
2. 条件付きリンク転送：LLI_nのLPDMA_CxLLR値で定義されるように、次のLLI_{n+1}のデータ構造により内部レジスタ･ファイルをLPDMAが条件付きで自動的に更新。

注： **内部レジスタ･ファイル（LLI_0）により定義される最初のデータ転送は、LPDMA_CxLLRの条件付き更新ビットUB1が設定されている（つまり､メモリの中の次のLLI_1により記述されるnullではないデータ転送が実行されることになっている）場合、null（LPDMA_CxBR1.BNDT[15:0] = 0）にすることができます。**

次のセクションで説明するとおり、意図するLPDMAの利用方法に従い、LPDMAチャネルxはフル･リンクリストの記述に従って実行する（run-to-completionモード、LPDMA_CxCR.LSM = 0）か、あるいは、LLIのシングル実行（リンク･ステップ･モード、LPDMA_CxCR.LSM=1）をプログラムすることが可能です。

18.4.7 LPDMAチャネル状態とrun-to-completionモードでのリンクリスト・プログラミング

LPDMA_CxCR.LSM = 0の場合、LPDMAチャネルxは内部的にプログラムされ、LPDMA_CxCR.ENに1を書き込むことで開始され、（可能であれば）チャネル・レベルで完了した後、次のようになります。チャネル転送は次のとおりです。

- 少なくとも次の設定をします。
  - 最初のLLI_0、内部リンクリストレジスタ・ファイル：LPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDAR、およびLPDMA_CxLLR
  - 最後のLLI_N、前の最後のLLI_{N-1}を反映したLPDMA_CxLLRで定義された、メモリ内のリンクリスト・データ構造に記述されたもの
- 最後のLLI_{N-1}転送の終わりに、LPDMA_CxLLR[31:0] = 0かつLPDMA_CxBR1.BNDT[15:0] = 0となったときに完了

LPDMA_CxLLR[31:0] = 0は、リンクリスト・ベースのチャネル完了の条件で、次のことを意味しています。

- 次のリンク・アドレスの下位16bit LPDMA_CxLLR.LA[15:0]がnull。
- LPDMA_CxLLRのすべての更新ビットUxx（UT1、UT2、UB1、USA、UDA、ULL）がnull。

LPDMA_CxLLR.LSM = 0の場合、チャネルは決して完了しません。

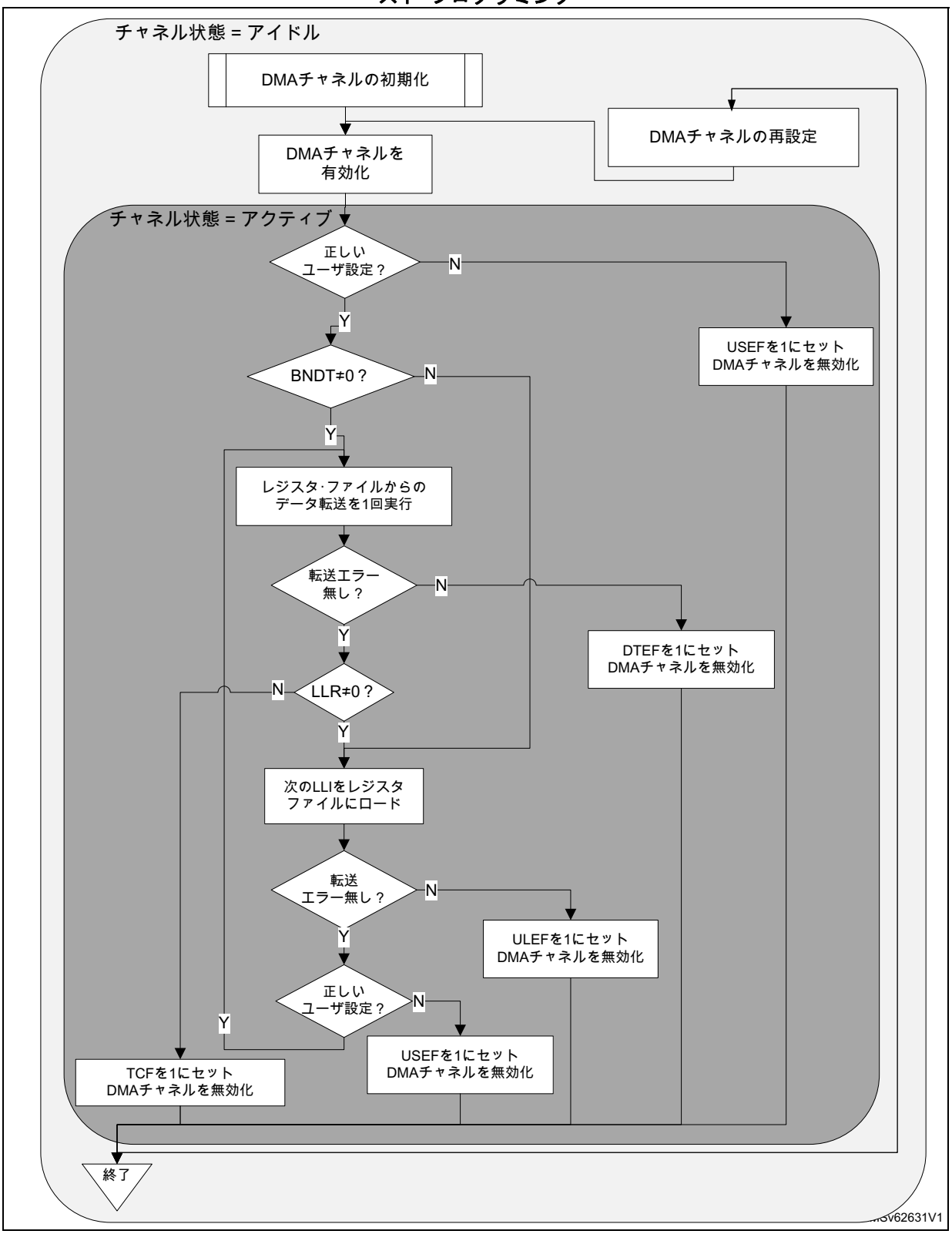
- 最後のLLI_Nが再帰的で、次のLLIとしてそれ自身を指す場合：
  - LPDMA_CxLLR.ULL = 1またはLPDMA_CxLLR.LA[15:2]のどちらかが同じ値で更新されます。
  - あるいはLPDMA_CxLLR.ULL = 0です。
- LLI_Nが前のLLIを指す場合。

通常のrun-to-completionモードでは、LPDMAチャネルの割当は、詳細プログラミングを含め、LPDMAの初期化の際に1度行われます。実行中に予約済みのデータ通信リンクとLPDMAサービスを確保するため、転送が連続的に反復される場合（ペリフェラルとメモリ間、またはメモリ間転送）、この予約済みのデータ通信リンクは1つのチャネルで構成することができ、チャネルを共有することもできます。反復される転送は、LLIのシーケンスで構成されます。

run-to-completionモードでのLPDMAチャネル実行と、そのレジスタのプログラミングを図76に示します。

注： **図76は、LPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、またはチャネルの完了時のみに立てることができます。run-to-completionモードでは、チャネル完了時のTCEM[1:0]の値に関わらず、ハードウェアは常にTCEF = 1を設定し、そのチャネルを無効にします。**

図76. run-to-completionモード（LPDMA_CxCR.LSM = 0）でのLPDMAチャネル実行およびリンクリスト･プログラミング



run-to-completionモードでの補助チャネル経由のLLI$_n$の実行時挿入

LLI$_{n-1}$のリンク転送の開始（LLI$_n$ロードの開始）は、LLI$_{n-1}$のデータ構造のLPDMA_CxTR2の以下のフィールドをプログラミングする場合、トリガの発生を条件とすることができます。

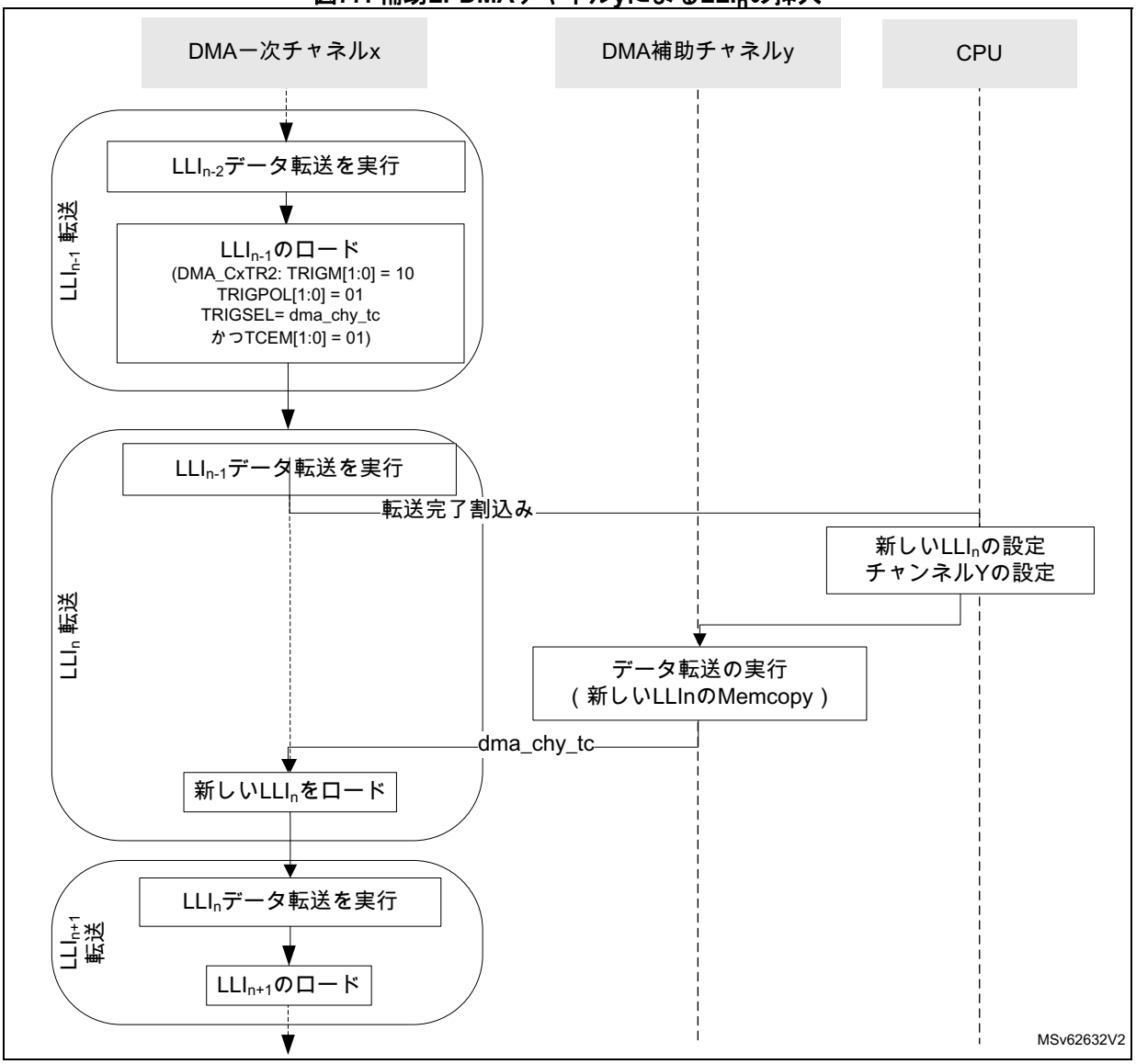
- TRIGM[1:0] = 10（リンク転送トリガ･モード）
- TRIGPOL[1:0] = 01または10（立上がり／立下りエッジ）
- TRIGSEL[4:0]（トリガ選択の詳細についてはセクション18.3.5を参照）

別の補助チャネル y は、チャネル xLLI$_n$ をメモリに格納するため、および転送完了イベントlpdma_chy_tcの生成に使用することができます。このイベントをチャネルxのLLI$_{n-1}$のリンク転送の入力トリガとして選択することで、ソフトウェアはそのLLI$_{n-1}$データ転送後、実際にLLI$_n$が実際に書き込まれるまで、一次チャネルxを休止することが可能です。

図77に、別の補助チャネルy経由での一次チャネルxのリンクリストの動的処理を示します。

注意： **この使用事例はトリガを必要としないLLI$_{n-1}$データ転送のアプリケーションに限定されています。このLLI$_{n-1}$のトリガ･モードは、次のLLI$_n$のロードに使用されます。**

図77. 補助LPDMAチャネルyによるLLI$_n$の挿入



18.4.8 リンク･ステップ･モードでのLPDMAチャネル状態およびリンクリスト･プログラミング

LPDMA_CxCR.LSM = 1の場合、LLIのシングル実行のたびに、（条件付きの）データ転送とその（条件付きの）リンク転送を含めてチャネル転送が実行され、完了します。

LPDMAチャネル転送はLLIレベルでプログラムし、LPDMA_CxCR.ENに1を書き込んで開始することが可能で、LLIレベルで完了した後、次のようになります。

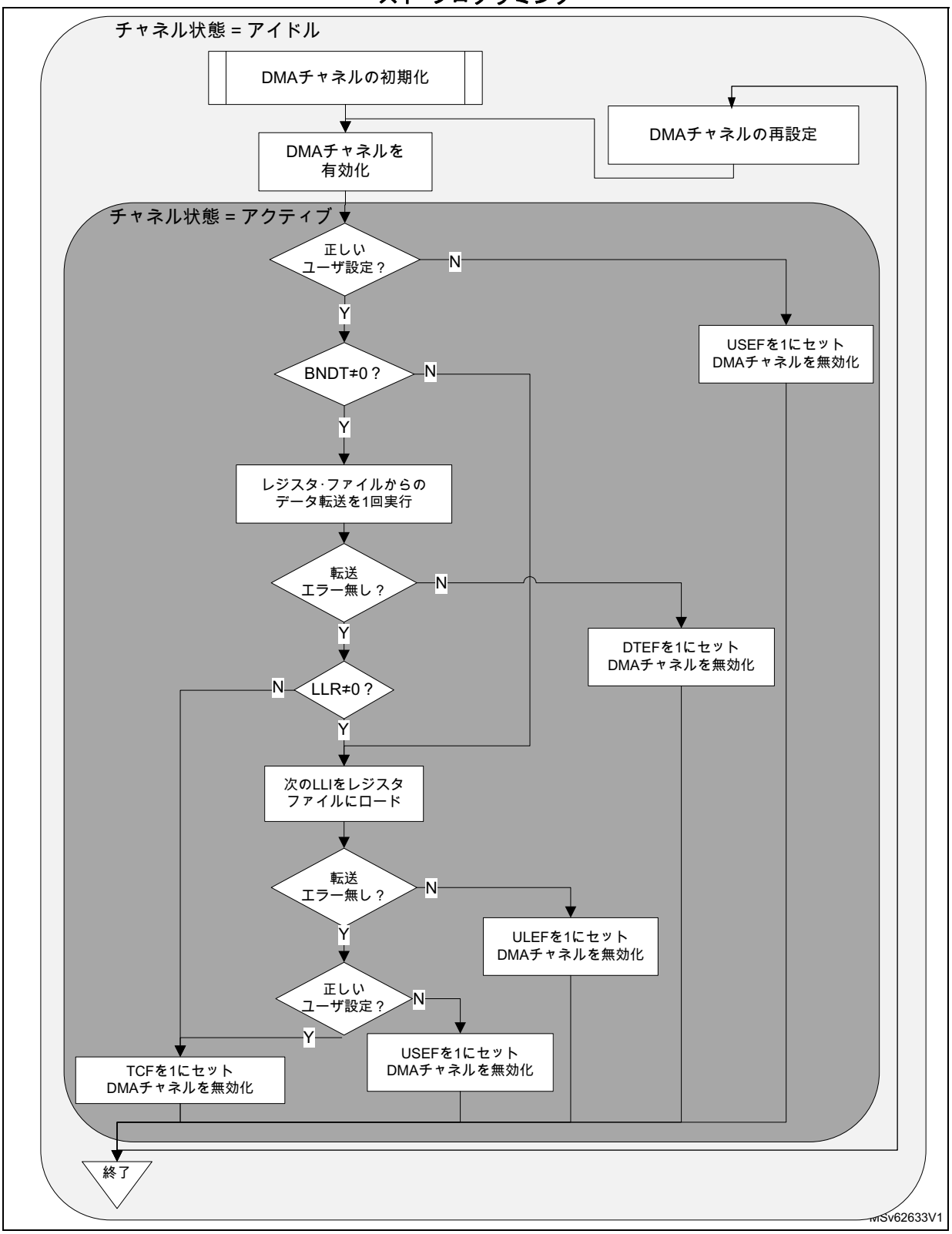
- 現在のLLI_n転送は次のように記述されます。
  - LPDMA_CxTR1が転送元／転送先の基本シングル転送を定義。
  - LPDMA_CxBR1がブロックレベルでバイト数を定義（BNDT[15:0]）。
  - LPDMA_CxTR2が転送の入力制御（リクエスト、トリガ）および出力制御（転送完了イベント）を定義します。
  - LPDMA_CxSAR/LPDMA_CxDARが転送元／転送先の転送開始アドレスを定義します。
  - LPDMA_CxLLRが、データ構造とメモリの中の次のLLI_{n+1}アドレス･オフセットを定義します。
- 現在のLLI_nのシングル実行後に現在のLLI_n転送が完了します。
  - （条件付きの）データ転送完了後（LPDMA_CxBR1.BNDT[15:0] = 0の場合）
  - メモリの中の次のLLI_{n+1}のデータ構造からのLPDMAリンク･レジスタ･ファイルの（条件付き）更新後

注： **LLIが再帰的である場合（次のLLIとしてそれ自身を指し、LPDMA_CxLLR.ULL = 1かつLPDMA_CxLLR.LA[15:2]が同じ値で更新されるか、LPDMA_CxLLR.ULL = 0の場合）、リンク･ステップ･モードのチャネルは、このLLIのシングル実行の各繰り返しの後に完了します。**

図78にLPDMAチャネル実行モードと、そのリンク･ステップ･モードでのプログラミングを示します。

注： **図78は、LPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、または最後のLLIデータ転送の完了時のみに立てることができます。リンク･ステップ･モードでは、LLIの各シングル実行後にチャネルが無効になり、TCEM[1:0]の値によりTCEFを立てるかどうかが変わります。**

図78. リンク･ステップ･モード（LPDMA_CxCR.LSM = 1）でのLPDMAチャネル実行およびリンクリスト･プログラミング

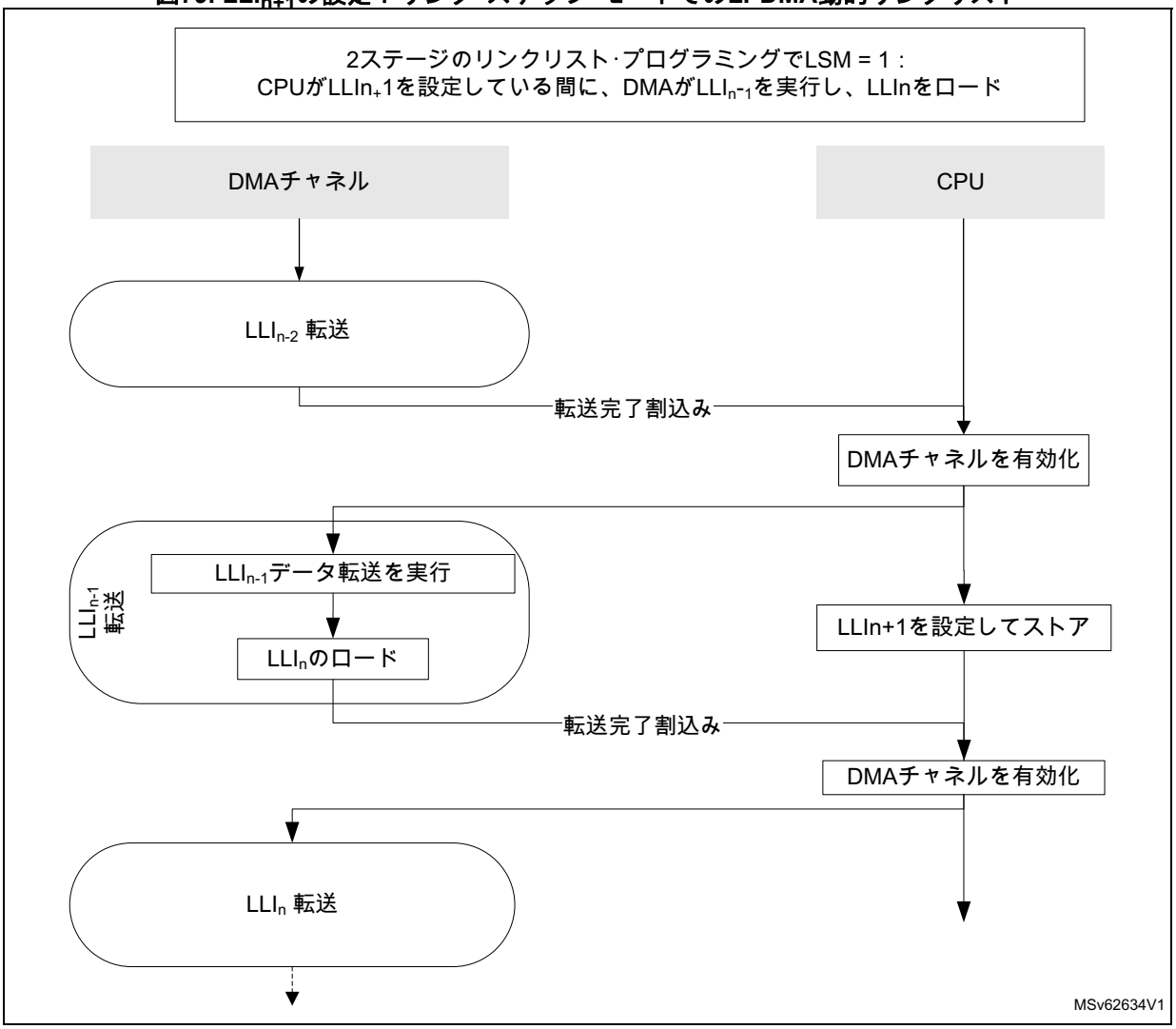


リンク･ステップ･モードを使用すると、実行中にメモリ内でLLIを動的に処理できます。あらゆるLLI_nに静的データ構造を使用することにより、ソフトウェアを支援することができます（LPDMA_CxLLRのすべての更新ビットが静的な値LLI_n.LLR.LA = LLI_{n-1}.LLR.LA+定数を持ちます）。

リンク･ステップ･モードでのLLI_{n+1}の実行時追加

実行中に、ソフトウェアは下図に示すように、LPDMAがLLI_{n-1}からの転送を実行し、LLI_nをメモリからロードするまで、またはその後まで、LLI_{n+1}（および次のLLI）の処理を保留することができます。

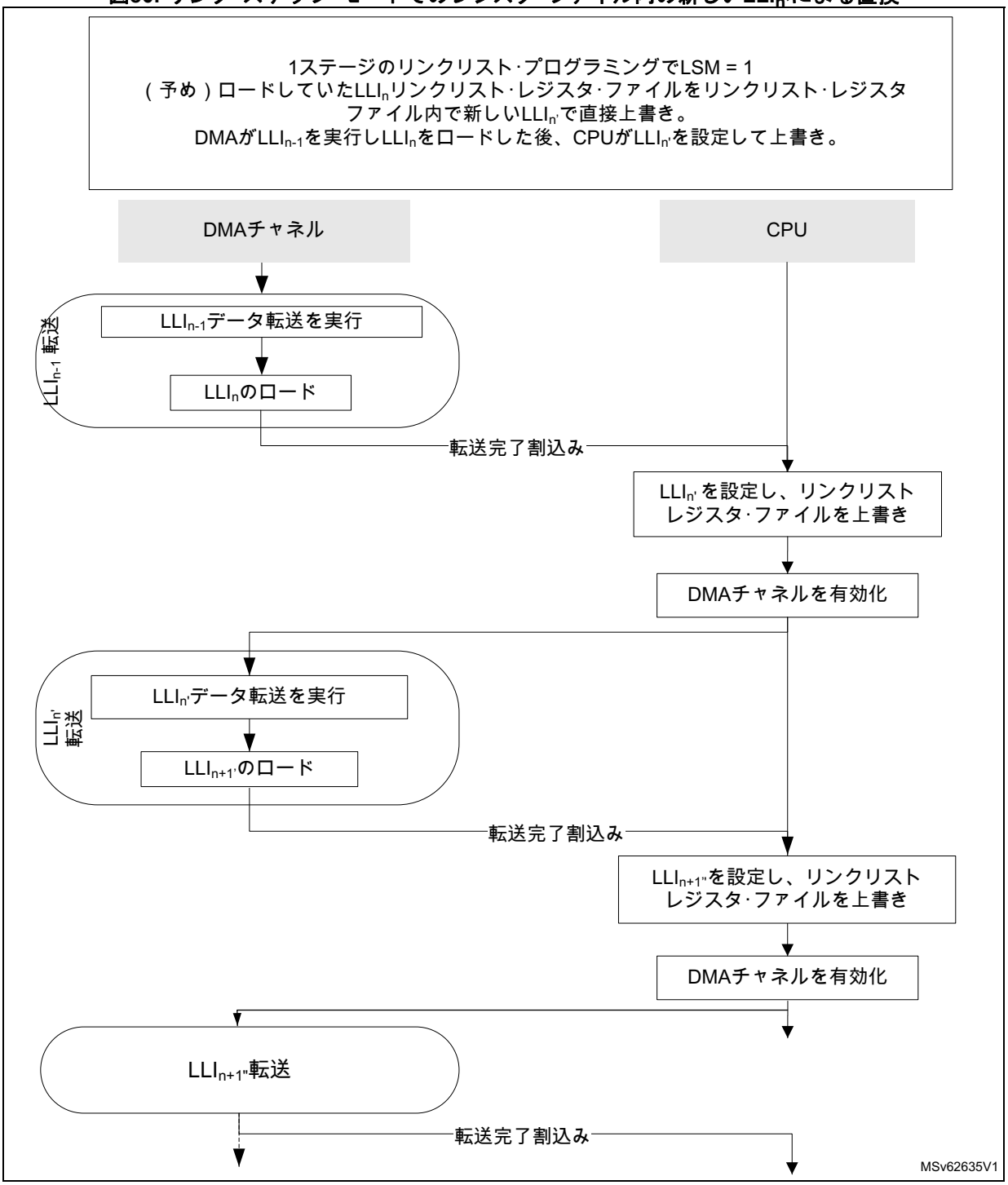
図79. LLI_{n+1}の設定：リンク･ステップ･モードでのLPDMA動的リンクリスト



リンク･ステップ･モードでのLLI_nと新しい$LLI_{n'}$との実行時置換（リンクリスト･レジスタ･ファイル内）

このリンク･ステップ･モードで、実行中に、図80に示すように、リンクリスト･レジスタ･ファイルを新しい$LLI_{n'}$で直接上書きすることによってLPDMAがLLI_{n-1}からの転送を実行し、前に処理されたLLI_nをメモリからロードした後に、ソフトウェアは新しい$LLI_{n'}$を設定し、挿入することができます。

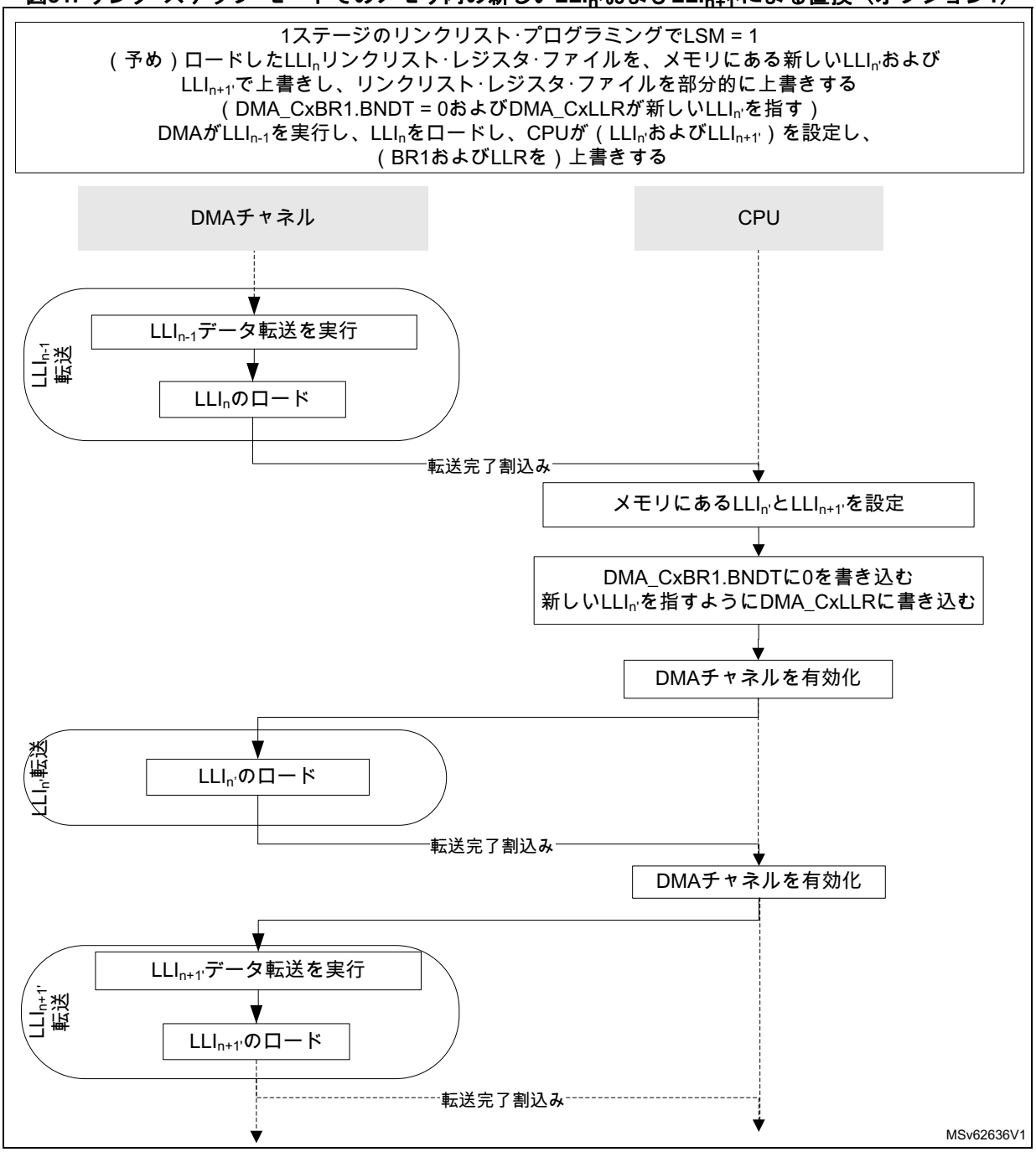
図80. リンク･ステップ･モードでのレジスタ･ファイル内の新しいLLI$_{n'}$による置換



リンク･ステップ･モードでのLLI$_n$と新しいLLI$_{n'}$との実行時置換（メモリ内）

図81に示すように、リンクリスト･レジスタ･ファイルを部分的に上書きする（LPDMA_CxBR1.BNDT[15:0]をnullにし、LPDMA_CxLLRが新しいLLI$_{n'}$を指すようにする）ことにより、LPDMAがLLI$_{n-1}$から転送を実行し、前に処理したLLI$_n$をメモリからロードした後に、ソフトウェアはメモリに新しいLLI$_{n'}$およびLLI$_{n+1'}$を設定し、挿入できます。

図81. リンク･ステップ･モードでのメモリ内の新しいLLI$_{n'}$およびLLI$_{n+1'}$による置換（オプション1）



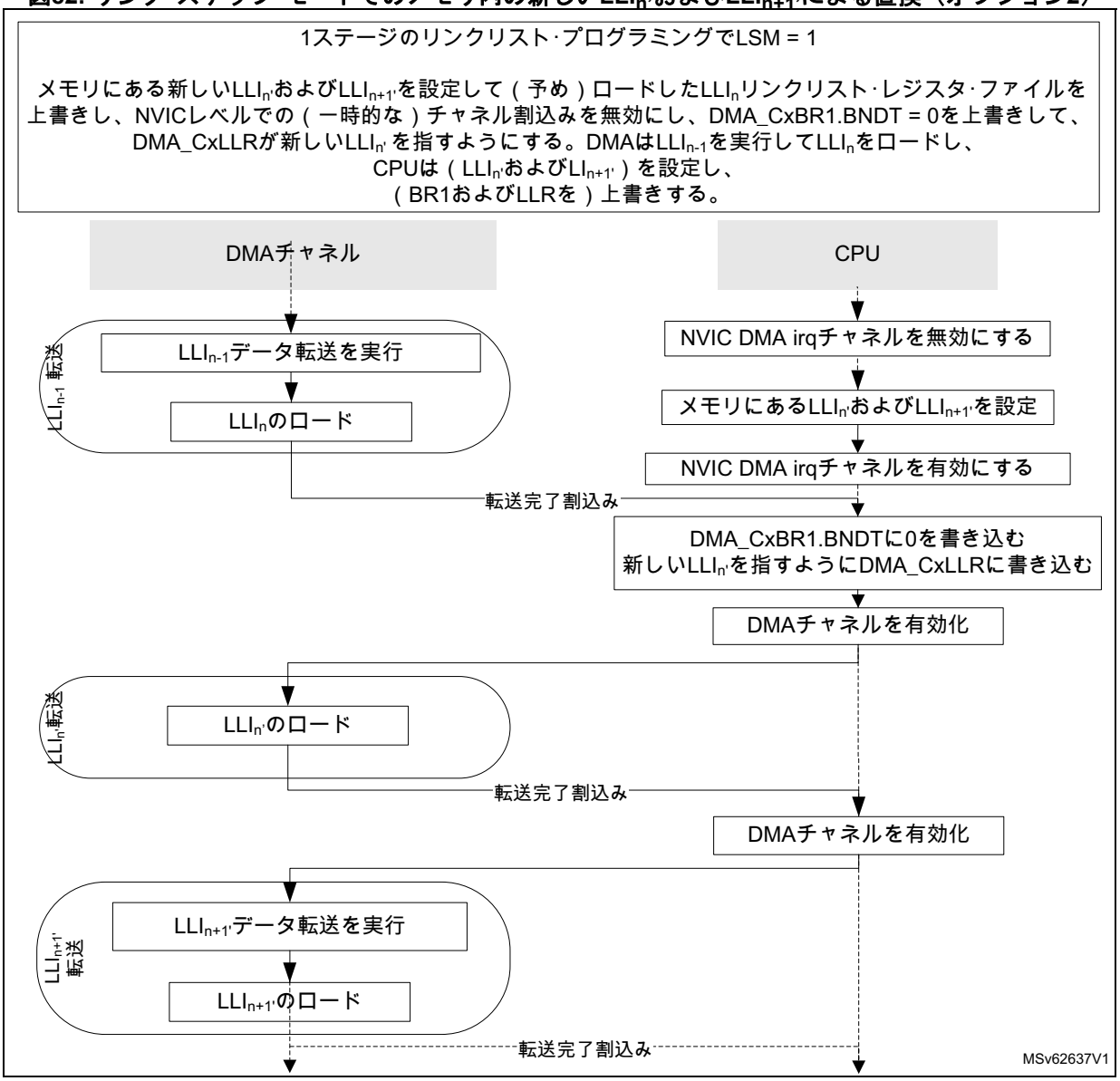
リンク･ステップ･モード（実行時）でのLLI$_n$を新しいLLI$_{n'}$への置き換え

他のソフトウェア実装が存在しますが、一方でLPDMAはLLI$_{n-1}$からの転送を実行して、前に（あるいはさらに前に）作成したLLI$_n$をメモリからロードします。ソフトウェアは次の項目を実行することができます。

1. 割込み操作による中断を防ぐためにNVICを無効にする。
2. 新しいLLI$_{n'}$とLLI$_{n+1'}$を設定する。
3. チャネル割込み（転送完了）通知のためにNVICを再び有効にする。

LLI$_{n-1}$割込みハンドラのソフトウェアはその後、図82に示すように、LPDMA_CxBR1.BNDT[15:0]をnullに上書きすることと、LPDMA_CxLLRが新しいLLI$_{n'}$を指すようにすることに限定されます。

図82. リンク･ステップ･モードでのメモリ内の新しいLLI$_{n'}$およびLLI$_{n+1'}$による置換（オプション2）



18.4.9 LPDMAチャネル状態およびリンクリスト･プログラミング

ソフトウェアは、チャネルが無効になっている時に（LPDMA_CxCR.EN = 0）チャネルを再設定し、実行モード（LPDMA_CxCR.LSM）を更新してrun-to-completionモードとリンク･ステップ･モードとを相互に切り替えることができます。

あらゆる実行モードにおいてソフトウェアは次の項目を行うことができます。

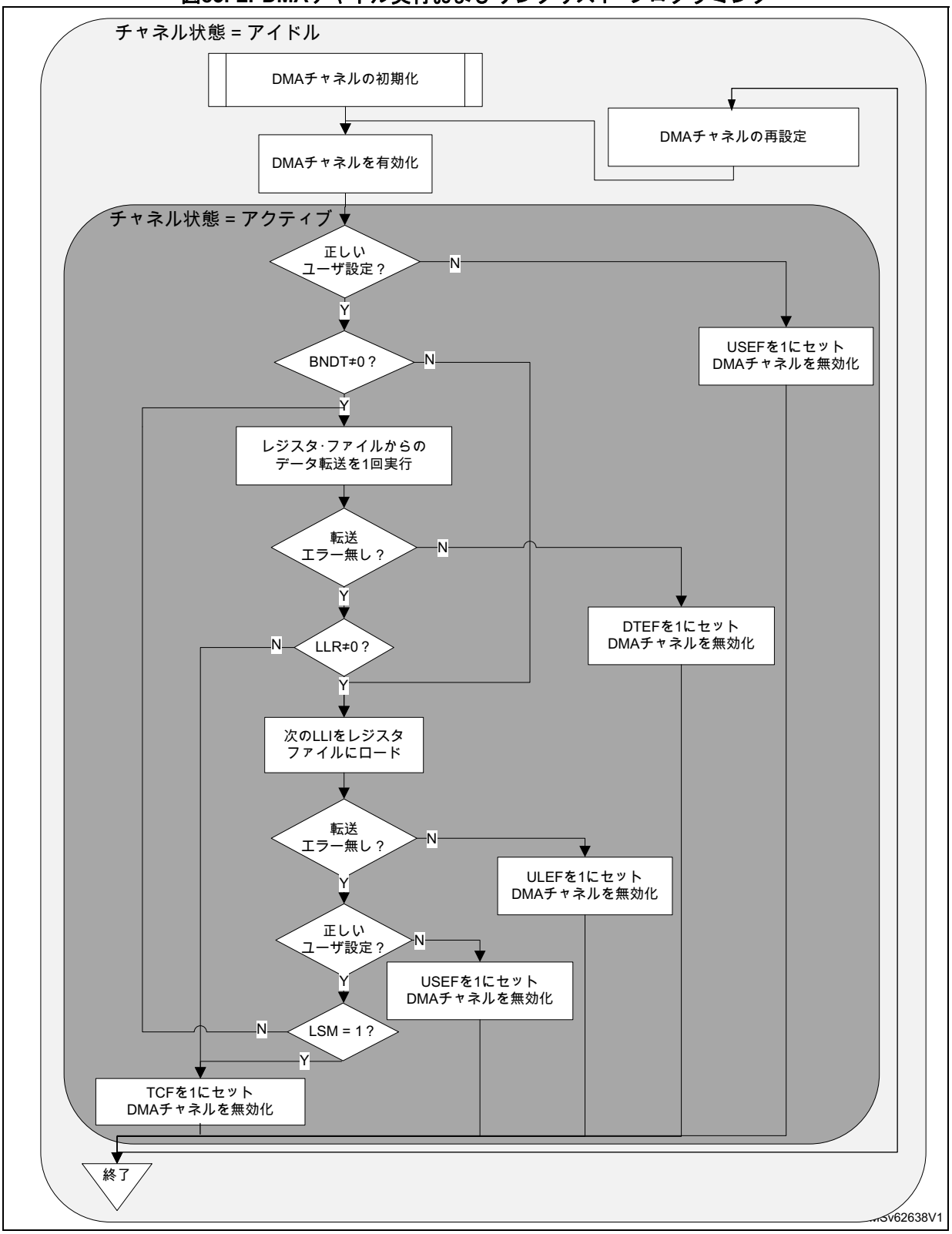
- このLLI$_{n+1}$がLPDMAチャネルによってロード／使用される前に、メモリ内でLLI$_{n+1}$を再プログラムして、このLLI$_{n+1}$（このLLI$_{n+1}$のLPDMA_CxLLRをクリア）により最終的にチャネルを完了します。
- チャネルをLSM更新によりアボートおよび再設定（セクション18.4.4を参照）

リンク･ステップ･モードでは、ソフトウェアがLLI$_{n-1}$の間、あらゆるLLIの各シングル実行後にLSMをクリアすることができます。

図83に実行モードに関係のない、全体の統合したLPDMAリンクリスト･プログラミングを示します。

注： **図83は、LPDMA_CxTR2でTCEM[1:0]のプログラム値に応じて、TCEFフラグを立てることができる頻度を示すものではありません。このフラグは、（各）ブロック完了時、（各）2Dブロック完了時、（各）LLI完了時、または最後のLLIデータ転送の完了時のみに立てることができます。run-to-completionモードではチャネル完了時のTCEM[1:0]の値に関わらず、ハードウェアは常にTCEF = 1を設定し、そのチャネルを無効にします。リンク･ステップ･モードでは、LLIの各シングル実行後にチャネルが無効になり、TCEM[1:0]の値によりTCEFを立てるかどうかが変わります。**

図83. LPDMAチャネル実行およびリンクリスト･プログラミング



18.4.10 LPDMA直接転送

直接モードというシングル転送操作モードがあります。LPDMAチャネルはどれも、直接モードで使用できます。どのチャネルもFIFOなしで実装されます（すべてのチャネルxで、dma_fifo_size[x] = 0）。

LPDMAシングル

最下位レベルでプログラムされる転送は、LPDMAシングルです。

LPDMAシングルデータ幅は、LPDMA_CxTR1のSDW_LOG2[1:0]およびDDW_LOG2[1:0]フィールド（それぞれ、転送元／転送先）での定義に従い、1、2、または4バイトです。

注： **8バイトのデータ幅（SDW_LOG2[1:0]=0b11またはDDW_LOG2[1:0] = 0b11）を割り当てないでください。割り当てた場合、ユーザ設定がレポートされ、転送は行われません。**

LPDMAシングルの各データがLPDMA_CxTR1のSINCおよびDINCビット（それぞれ、転送元／転送先用）によって定義された後のアドレス指定モード：固定アドレス指定または隣接データでのインクリメントアドレス指定。

LPDMAソース／デスティネーション･シングルの開始アドレスおよび次のアドレス（LPDMA_CxSARおよびLPDMA_CxDARで定義される）は、それぞれのデータ幅に合っていなくてはなりません。

下の表にLPDMAシングルの主な特徴を列記します。

上の表の次のシングル･アドレスは、プログラムされたソース／デスティネーション･シングルが一旦、完了したときに、LPDMA_CxSARまたはLPDMA_CxDARにより指定された次の転送元／転送先のアドレスです。

表149.プログラムされたLPDMAソース／デスティネーション･シングル

プログラムされたLPDMAソース／デスティネーション･シングル	SDW_LOG2[1:0] DDW_LOG2[1:0]	データ幅（バイト）	SINC/ DINC	アドレス指定モード	SAR/ DARnext シングル アドレス	アドレスのアライメント
固定バイト･シングル	00	1	0	固定	+0	1
固定ハーフワード･シングル	01	2				2
固定ワード･シングル	10	4				4
インクリメント･バイト･シングル	00	1	1	インクリメント	+1	1
インクリメント･ハーフワード･シングル	01	2			+2	2
インクリメント･ワード･シングル	10	4			+4	4
設定禁止	11	USEFが生成され、シングルは行われません。				

ダイレクト･モードでは、LPDMAシングルはAHBシングル転送となります。

LPDMAデータ･ハンドリング：バイト･ベースのパディング／切り詰め、符号拡張、左／右詰め

ユーザは、転送元から転送されたデータと転送先に転送するデータとの間でのデータ操作を設定できます。データ･ハンドリングはLPDMA_CxTR1レジスタによって次のように制御されます。

- 転送先のデータ幅=転送元のデータ幅の場合（DDW_LOG2[1:0] = SDW_LOG2[1:0]）、転送元のデータがそのままコピーされ、転送先に転送されます。
- それ以外の場合、PAMに応じて：
  - 転送先のデータ幅>転送元のデータ幅の場合、転送元のデータを右詰めにして0でパディングするか、転送先のデータ幅まで符号拡張することができます。
  - 転送先のデータ幅 < 転送元のデータ幅の場合、転送元のデータは、右詰めにし、転送先のデータ幅に合わせて左を切り捨てるか、左詰めにし、転送先のデータ幅に合わせて右を切り捨てることができます。

転送先への転送の生成前には、転送元からの2つの異なる転送データの間でデータ操作は行われません。

下の表は転送元から転送先への考えられえるデータ･ハンドリングを列記したものです。

表150.プログラムされたデータ･ハンドリング

SDW_LOG2 [1:0]	転送元データ	転送元データ･ストリーム(1)	DDW_LOG2 [1:0]	転送先データ	PAM[0]	転送先データ･ストリーム(1)
00	バイト	B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0	00	バイト	x	B_7、B_6、B_5、B_4、B_3、B_2、B_1、B_0
			01	ハーフワード	0 (RA, 0P)(2)(3)	$0B_3$、$0B_2$、$0B_1$、$0B_0$
					1 (RA, SE) (2)(4)	SB_3、SB_2、SB_1、SB_0
			10	ワード	0 (RA, 0P)	$000B_1$、$000B_0$
					1 (RA, SE)	$SSSB_1$、$SSSB_0$
01	ハーフワード	B_7B_6、B_5B_4、B_3B_2、B_1B_0	00	バイト	0 (RA, LT) (2)	B_6、B_4、B_2、B_0
					1 (LA, RT) (2)	B_7、B_5、B_3、B_1
			01	ハーフワード	xx	B_7B_6、B_5B_4、B_3B_2、B_1B_0
			10	ワード	0 (RA, 0P)	$00B_3B_2$、$00B_1B_0$
					1 (RA, SE)	SSB_3B_2、SSB_1B_0
10	ワード	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$	00	バイト	0 (RA, LT)	B_{12}、B_8、B_4、B_0
					1 (LA, RT)	B_{15}、B_{11}、B_7、B_3
			01	ハーフワード	0 (RA, LT)	B_5B_4、B_1B_0
				ワード	1 (LA, RT)	B_7B_6、B_3B_2
			10		xx	$B_7B_6B_5B_4$、$B_3B_2B_1B_0$

1. データ･ストリームを最低インデックス（B_0）のバイトから始めて、タイミングで整列します。
2. RA = 右詰め。LA = 左詰め。RT = 右を切り捨て。LT = 左を切り捨て。
3. 0P = 転送先のデータ幅まで0のビットをパディング。
4. SE = 転送先のデータ幅まで符号ビットで拡張。

18.4.11 LPDMA転送リクエストとアービトレーション

LPDMA転送リクエスト

LPDMA_CxTR2の定義に従って、プログラムされたLPDMAデータ転送を次のいずれかの方法でリクエストします。

- 制御ビットSWREQ = 1の場合はソフトウェア･リクエスト：一般的に、メモリマップされたアドレスから別のメモリマップされたアドレスへのデータ転送のためにCPUによって使用されます（メモリ間、GPIOとメモリ間）
- SWREQ = 0の場合はペリフェラルからの入力ハードウェア･リクエスト：LPDMAハードウェアペリフェラル･リクエストの選択は、REQSEL[4:0]フィールドを使用して行われます（セクション18.3.4を参照）。選択したハードウェア･リクエストは次のいずれかです。
  - LPDMA モードで設定されたペリフェラルからのハードウェア･リクエスト（ペリフェラル･データ･レジスタとメモリの相互転送）
  - メモリから制御レジスタを更新するための、ペリフェラルからのハードウェア･リクエスト
  - メモリに転送されたステータス･レジスタの読出しのための、ペリフェラルからのハードウェア･リクエスト

注意： **所定の時間にこのリクエストがペリフェラルによってアサートされ、各チャネルでこのリクエストされたデータ転送が可能な状態である場合、ユーザーは同じ入力ハードウェア･ペリフェラルLPDMAリクエストをLPDMA_CxTR.REQSEL[4:0]を介して2つの異なるチャネルに割り当てることはできません。ユーザ設定エラーのレポートはありません。**

アービトレーションのためのLPDMA転送リクエスト

特定のチャネルについて、LPDMAでリクエストされた転送元アドレスから転送先アドレスへのデータ転送は、他のチャネルから同時にリクエストされたLPDMA転送に対してアービトレーションされ、AHBマスタ･ポート上でスケジュールされます。LPDMAデータ転送はLPDMAアービトレーションの最小単位です。これはAHB読出しシングルと、それに続くAHB書込みシングルで構成されます。これはLPDMA_CxCR.PRIO[1:0]で定義されたチャネルの優先順位に基づいて、両方のAHB転送に対して1回、アービタによって許可されます。

LPDMAにリクエストされ、アービトレーションされたリンク転送は、メモリのリンクリスト･データ構造からリンクリスト･レジスタ（LPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDAR、またはLPDMA_CxLLR）の1つへの、1回の32bit読出しにより構成されます。メモリからの各32bit読出しは、データ転送と同じチャネルの優先順位でアービトレーションされ、マスタ･ポートでスケジューリングされます。

アービトレーションは、シングル転送が許可されるたびに次のように行われます。

- リクエストされたデータ転送のプログラム方法に関係なく：
  - メモリ間転送の場合はソフトウェア･リクエストにより（LPDMA_CxTR2.SWREQ = 1）
  - メモリとペリフェラルの間の相互転送の場合はハードウェアリクエストにより（LPDMA_CxTR2.SWREQ = 0）
- ハードウェア･リクエストのタイプに関係なく

リクエストされたデータ転送が、ペリフェラルからのハードウェア･リクエストでプログラムされている場合（LPDMA_CxTR2.SWREQ = 0）、ブロックの最初のメモリ読出しは、ペリフェラルが転送の転送元か転送先かに関係なく、選択された対応するハードウェア･リクエストが行われることにより許可されます。この最初のメモリへの読出しリクエストは、早い段階では（ブロック転送が有効になって実行可能になってすぐには）アービタでは考慮されません。

LPDMAアービトレーション

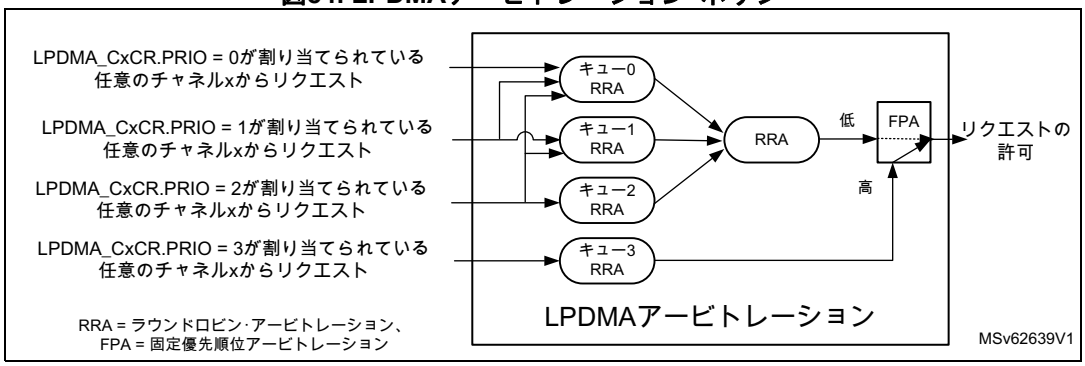
LPDMAアービトレーションは4グレードが割り当てられたチャネルの優先順位（LPDMA_CxCR.PRIO[1:0]）から行われます。図84に図示したアービトレーション･ポリシーは次のように定義されます。

- 優先順位の高い1つのトラフィック･クラス（キュー3）。優先順位3が割り当てられたタイミングが重要なチャネル専用。

  このトラフィック･クラスは、他のあらゆる低優先順位のトラフィック･クラスに対して、固定優先順位アービトレーションで許可されます。このクラス内では、リクエストされたシングル転送はラウンドロビン方式でアービトレーションされます。
- 優先順位の低い3つのトラフィック･クラス（キュー 0、1、2）。タイミングの制約が緩い、優先順位0、1、2のチャネル。

  このクラスの、リクエストされたそれぞれのシングル転送は、プログラムされた優先順位で単純な重み付けを行い、ラウンドロビン方式でアービトレーションされます：
  - 優先順位0のリクエストはキュー 0に割り当てられます。
  - 優先順位1のリクエストはキュー 0と1に割り当て、複製されます。
  - 優先順位2のリクエストはキュー 0、1、2に割り当て、複製されます。
  - キュー 0、1、2は、同時リクエストがある場合、その有効な入力リクエストをラウンド･ロビン（持ち回り）方式で平等に許可します。
  - さらに、低優先順位のトラフィックの第2ステージがあり、そこにはキュー 0、1、2から同時に選択されたリクエスト間で公平にオルタネートするラウンド･ロビン（持ち回り）のアービタがあります。

図84. LPDMAアービトレーション･ポリシー



1. RRA：ラウンドロビン方式のアービトレーション
2. FPA：固定優先順位のアービトレーション

LPDMAアービトレーションとバンド幅

このアービトレーション･ポリシーでは、次の事項が保証されます。

- 同じポリシーのリクエストの最大バンド幅は等しい
- タイミングが重要なリクエスト（優先順位3）の予約済みのバンド幅（B_{Q3}と表記）
- 様々な、低い優先順位のリクエスト（優先順位0、1、2の間）に対する残りの重み付けバンド幅。

次の2つの例では、重み付けラウンド･ロビン（持ち回り）アービトレーションがプログラムされた優先順位で実施されていることがよく分かります。

- **例1**：タイミングの制約が緩い2つのLPDMAリクエスト、req0とreq1の基本アプリケーション。次のようなプログラミングの可能性があります。
  - 同じ優先順位が割り当てられている場合、アービタによりreq0に割り当てられるバンド幅（B_{req0}）はreq1に割り当てられるバンド幅（B_{req1}）と**同じ**です。

    $B_{req0} = B_{req1} = 1/2 \times (1 - B_{Q3})$
  - req0が優先順位0に、req1が優先順位1に割り当てられると、req0に割り当てられるバンド幅（B_{P0}）はreq1に割り当てられるバンド幅（B_{P1}）の**3分の1**になります。

    $B_{req0} = B_{P0} = 1/2 \times 1/2 \times (1 - B_{Q3}) = 1/4 \times (1 - B_{Q3})$

    $B_{req1} = B_{P1} = (1/2 + 1) \times 1/3 \times (1 - B_{Q3}) = 3/4 \times (1 - B_{Q3})$
  - req0が優先順位0に、req1が優先順位2に割り当てられると、req0に割り当てられるバンド幅（B_{P0}）はreq1に割り当てられるバンド幅（B_{P2}）の**5分の1**になります。

    $B_{req0} = B_{P0} = 1/2 \times 1/3 \times (1 - B_{Q3}) = 1/6 \times (1 - B_{Q3})$

    $B_{req1} = B_{P2} = (1/2 + 1 + 1) \times 1/3 \times (1 - B_{Q3}) = 5/6 \times (1 - B_{Q3})$

  上で計算したバンド幅の計算は理論的な入力リクエストに基づいており、あらゆるLPDMAクロック･サイクルで常に有効です。アービタから計算されたこのバンド幅は、アプリケーションにより得られたリクエストの頻度により重み付けする必要があり、これは、常に有効とは限らず、あるLPDMAクライアント（たとえば400kHzのI2C）と別のクライアント（1kHzのPWM）とでは、上で示した3倍や5倍とは大きく異なる場合があります。
- **例2**：nullではない同じN個のLPDMAリクエストを、すべてのタイミングの制約が緩い優先順位0、1、2にユーザが配布するアプリケーション。このときのバンド幅の計算は次のとおりです。
  - 優先順位0の一連のリクエストに対して割り当てたバンド幅（B_{P0}）は、

    $B_{P0} = 1/3 \times 1/3 \times (1 - B_{Q3}) = 1/9 \times (1 - B_{Q3})$
  - 優先順位1の一連のリクエストに対して割り当てたバンド幅（B_{P1}）は、

    $B_{P1} = (1/3 + 1/2) \times 1/3 \times (1 - B_{Q3}) = 5/18 \times (1 - B_{Q3})$
  - 優先順位2の一連のリクエストに対して割り当てたバンド幅（B_{P2}）は、

    $B_{P2} = (1/3 + 1/2 + 1) \times 1/3 \times (1 - B_{Q3}) = 11/18 \times (1 - B_{Q3})$
  - その優先順位Pi（i = 0から2）のN個のリクエストの中のnリクエストに割り当てるバンド幅（B_n）は$B_n = 1/N \times B_{Pi}$
  - 優先順位0iのリクエストnに割り当てたバンド幅（$B_{n,\ Pi}$）は、

    $B_{n,\ P0} = 1/N \times 1/9 \times (1 - B_{Q3})$

    $B_{n,\ P1} = 1/N \times 5/18 \times (1 - B_{Q3})$

    $B_{n、P2} = 1/N \times 11/18 \times (1 - B_{Q3})$

  この例では、マスタ･ポート･バスのバンド幅がタイミングが重要なキュー3により完全には消費されていない場合、残りのバンド幅は、優先順位0のリクエストに割り当てられているバンド幅が優先順位1のリクエストに割り当てられているバンド幅の2.5分の1、優先順位2のリクエストに割り当てられているバンド幅に対しては5.5分の1になるように割り当てられます。

より一般的には、次のリクエストが存在することを想定します。

- I個のリクエスト（I ≥ 0）を優先順位0に割り当て。
  I > 0の場合、これらのリクエストはi = 0からI-1で記録されます。
- J個のリクエスト（J ≥ 0）を優先順位1に割り当て。
  J > 0の場合、これらのリクエストはj = 0からJ-1で記録されます。
- K個のリクエスト（K ≥ 0）を優先順位2に割り当て。
  これらのリクエストはk = 0からK-1で記録されます。
- L個のリクエスト（L ≥ 0）を優先順位3に割り当て。
  L > 0の場合、これらのリクエストはI = 0からL-1で記録されます。

B_{Q3}はタイミングが重要なリクエストのために予約されているバンド幅なので、優先順位3の各リクエストLのバンド幅は、

- L > 0の場合、$B_I = B_{Q3}/ L$（その他の場合は、$B_I = 0$）

タイミングが重要ではないキューのそれぞれのバンド幅は次のとおりです。

- $B_{Q0} = 1/3 \times (1 - B_{Q3})$
- $B_{Q1} = 1/3 \times (1 - B_{Q3})$
- $B_{Q2} = 1/3 \times (1 - B_{Q3})$

優先順位0の一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P0} = I / (I + J + K) \times B_{Q0}$

優先順位0の各リクエストiに対するバンド幅は次のとおりです。

- L > 0の場合、$B_i = B_{P0} / I$（その他の場合は$B_{P0} = 0$）

優先順位1で　キュー 0にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P1,Q0} = J / (I + J + K) \times B_{Q0}$

優先順位1でキュー 1にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P1,Q1} = J / (J + K) \times B_{Q1}$

優先順位1の一連のリクエストに対するバンド幅の合計は次のとおりです。

- $B_{P1} = B_{P1,Q0} + B_{P1,Q1}$

優先順位1のリクエストjに対するバンド幅は次のとおりです。

- J > 0の場合、$B_j = B_{P1} / J$（その他の場合は$B_j = 0$）

優先順位2でキュー 0にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q0} = K / (I + J + K) \times B_{Q0}$

優先順位2でキュー 1にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q1} = K / (J + K) \times B_{Q1}$

優先順位2でキュー 2にルーティングされる一連のリクエストに対するバンド幅は次のとおりです。

- $B_{P2,Q2} = B_{Q2}$

優先順位2の一連のリクエストに対するバンド幅の合計は次のとおりです。

- $B_{P2} = B_{P2,Q0} + B_{P2,Q1} + B_{P2,Q2}$

優先順位2のリクエストkに対するバンド幅は次のとおりです。

- $B_k = B_{P2} / K$（一般的にはK > 0）

そのため、タイミングが重要ではないリクエストi、j、kに割り当てられる残留バンド幅の最大値は最終的に次のようになります。

- 一般的な場合（優先順位2のリクエストkが最低1つある場合（K > 0））。
  - $B_i = 1/I \times 1/3 \times I/(I + J + K) \times (1 - B_{Q3})$
  - $B_j = 1/J \times 1/3 \times [J/(I + J + K) + J/(J + K)] \times (1 - B_{Q3})$
  - $B_k = 1/K \times 1/3 \times [K/(I + J + K) + K/(J + K) + 1] \times (1 - B_{Q3})$
- 特別な場合（優先順位2のリクエストkがない場合（K = 0））。
  - $B_i = 1/I \times 1/2 \times I/(I + J) \times (1 - B_{Q3})$
  - $B_j = 1/J \times 1/2 \times [J/(I + J) + 1] \times (1 - B_{Q3})$

結果としてLPDMAアービタは、各キューや、より一般的には各リクエスト／チャネルに対するプログラム可能な重み付けバンド幅のリミッターとして使用することができます。重み付けの差はプログラムされたチャネルの優先順位から単純に与えられることになるのです。

18.4.12 LPDMAトリガ転送

プログラムされたLPDMA転送は、LPDMA_CxTR2.TRIGPOL[1:0]およびLPDMA_CxTR2.TRIGSEL[4:0]での定義に従い、選択した入力トリガイベントの立上がり／立下がりエッジによってトリガできます（トリガの選択については、セクション18.3.5を参照）。

LPDMA_CxTR2.TRIGM[1:0]のトリガ･モードで定義されたトリガ転送は、LLIのデータ転送レベルでブロックの最初のシングル読出し、またはプログラムされた各シングル読出しの条件付けができます。トリガ･モードはLLIリンク転送の条件付けをプログラムをすることも可能です（詳細はLPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）のTRIGM[1:0]を参照）。

トリガ･ヒットのメモリ書込みとトリガ･オーバーラン･フラグの生成

チャネルxに対するトリガのLPDMA監視が開始されるのは、新しいアクティブなトリガ設定、つまり選択したトリガの立上がり／立下がりエッジ（それぞれ、TRIGPOL[1:0] = 01またはTRIGPOL[1:0] = 10）によりそのチャネルが有効化／ロードされるときです。

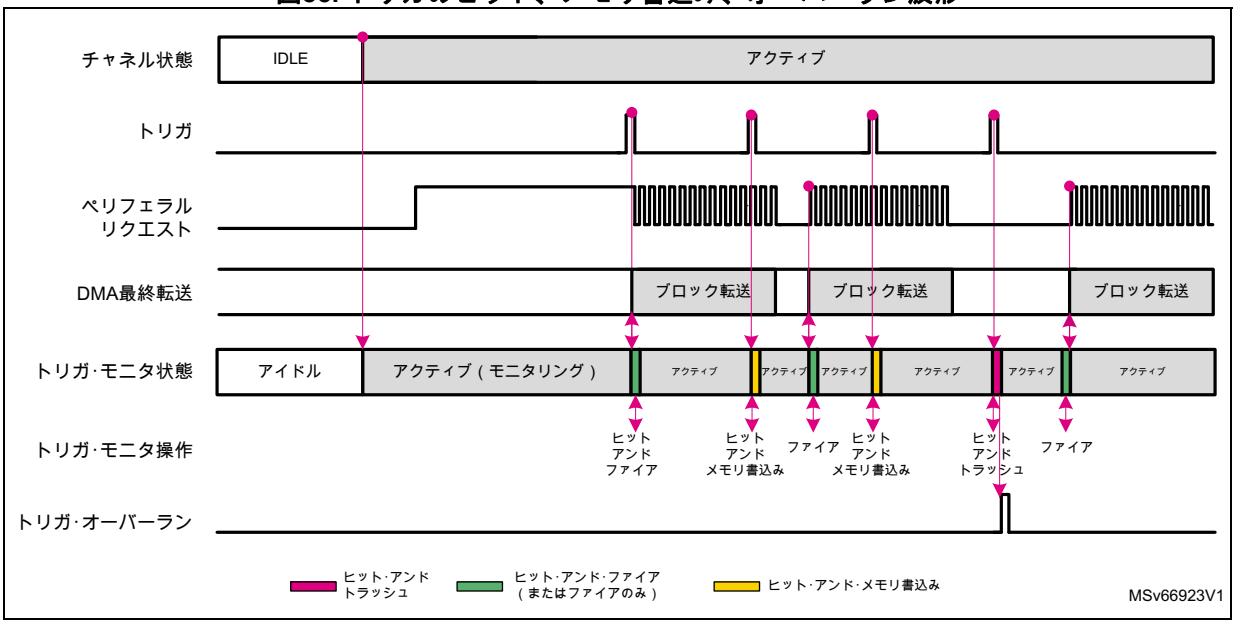
このトリガのモニタリングは、トリガされた完了していない（データまたはリンクの）転送を行っている間、アクティブなままとなります。新しいトリガが検出された場合、定義された立上がり／立下がりエッジとTRIGSEL[4:0]が変更されず、そのチャネルが有効である限り、このヒットは次の転送を許可するために内部的にメモリに書き込まれます。

TRIGSEL[4:0]またはTRIGPOL[1:0]の新しい値でLPDMA_CxTR2を更新する次のLLI_{n+1}の転送が、前に定義したLLI_nトリガのメモリ書込みされたヒットのモニタリング、トラッシングをリセットします。

注意： **最初の新しいトリガhit_{n+1}をメモリに書き込んだ後、別の2つめのトリガhit_{n+2}を検出し、hit_nにトリガされた転送が完了していない場合、hit_{n+2}は失われメモリに書き込まれません。トリガ･オーバーラン･フラグが報告され(LPDMA_CxSR.TOF = 1)、有効な場合は割込みが生成されます(LPDMA_CxCR.TOIE = 1の場合)トリガのオーバーランでハードウェアにより、チャネルが自動的に無効になることはありません。**

下図に、ブロックレベルのトリガモードと立上がりエッジのトリガの極性を記載した設定例でのトリガのヒット、メモリ書込み、オーバーランを示します。

図85. トリガのヒット、メモリ書込み、オーバーラン波形



注： **ユーザは、異なるチャネルに同じ入力トリガ·イベントを割り当てることができます。これは、ブロードキャスト·トリガ·イベントに異なるチャネルをトリガするために使用可能です。**

18.4.13 リンクリスト·プログラミングでのLPDMAサーキュラ·バッファリング

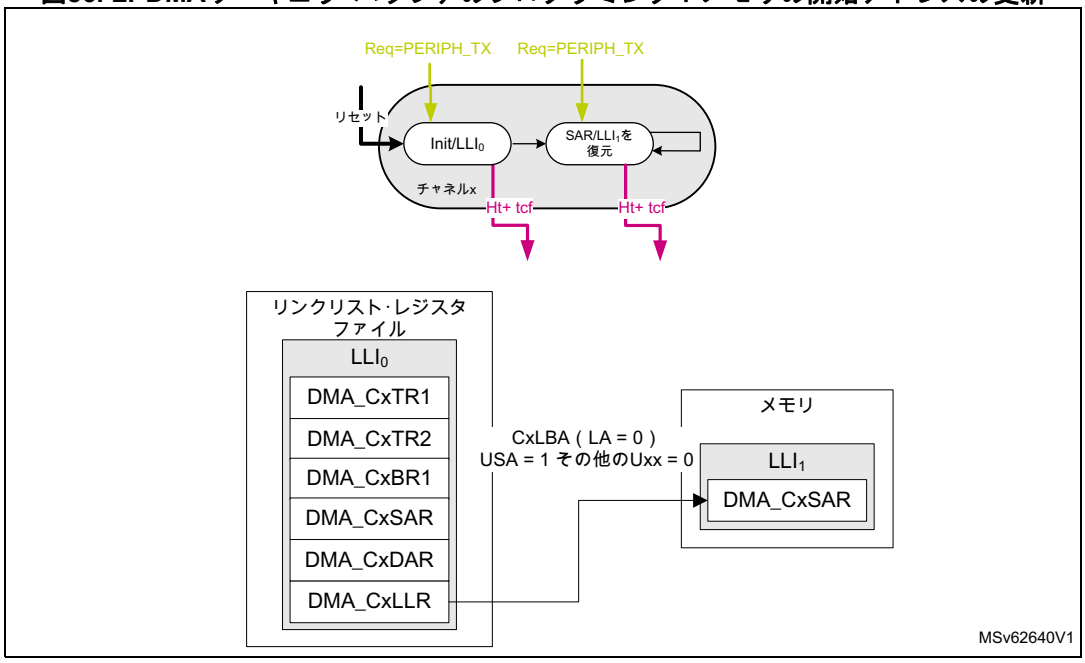
メモリとペリフェラル間の相互転送のLPDMAサーキュラ·バッファリング

サーキュラ·バッファで、連続したメモリからペリフェラル（またはペリフェラルからメモリ）への転送を行うには、ソフトウェアが1/2転送と完全な転送イベント／割込み生成（LPDMA_CxCR.HTIE = 1およびLPDMA_CxCR.TCIE = 1）でチャネルを設定して、同時バッファ·ソフトウェア·プロセシングを可能にする必要があります。

LLI_0は最初のブロック転送用に設定されます。メモリからペリフェラルへの転送（ペリフェラルからメモリへの転送）で、メモリの転送元（転送先）の開始アドレスを復元するには、連続実行のLLI_1が必要です。LPDMAは、ブロック転送が完了し、LPDMA_CxBR1の復元が必要無くなると、最初にプログラムされたLPDMA_CxBR1.BNDT[15:0]を自動的にリロードします。

LPDMAチャネルと転送元のサーキュラ·バッファによる、このプログラミングを下図に示します。

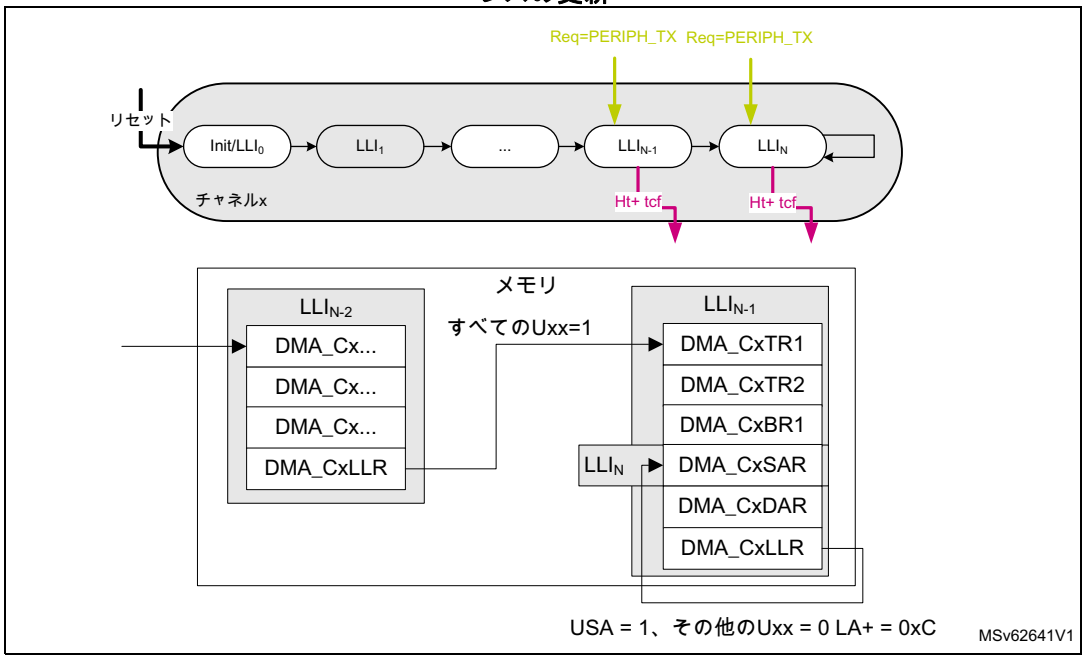
図86. LPDMAサーキュラ·バッファのプログラミング：メモリの開始アドレスの更新



共有LPDMAのチャネルxを経由した、他の転送後にサーキュラ·バッファリングを実行しなくてはならない場合、最初のブロック転送の設定にはメモリにある直前のLLI_{N-1}が必要です。最後のLLI_Nで、メモリからペリフェラルへの転送（ペリフェラルからメモリへの転送）で、メモリの転送元（転送先）の開始アドレスを復元します。

共有LPDMAチャネルと転送元のサーキュラ·バッファによる、このプログラミングを下図に示します。

図87. サーキュラ·バッファリングが実行される共有LPDMAチャネル：メモリの開始アドレスの更新



18.4.14 LPDMAセキュア／非セキュア･チャネル

LPDMAコントローラは、チャネル･レベルでTrustZoneハードウェア･アーキテクチャに準拠しています。このアーキテクチャでは、コントローラの全リソースが、常にセキュア･ワールドおよび非セキュア･ワールドのどちらか一方に存在できるようにパーティション分割されています。

あらゆるチャネルxがセキュアまたは非セキュアのハードウェア･リソースで、LPDMA_SECCFGR.SECxにより設定されます。

チャネルxがセキュア特権エージェントによりセキュア状態に設定されている場合、以下のアクセス制御規則が適用されます。

- このチャネルのレジスタ･フィールドへの非セキュア読出しアクセスは、非セキュア･エージェントで読み出すことができるLPDMA_SECCFGR、LPDMA_PRIVCFGR、LPDMA_RCFGLOCKRを除いて、強制的に0を返します。
- このチャネルのレジスタ･フィールドへの非セキュア書込みアクセスによる影響はありません。

チャネルxがセキュア状態に設定されている場合、セキュア･エージェントは、転送元からのLPDMAデータ転送（LPDMA_CxTR1.SSEC）と、転送先へのLPDMAデータ転送（LPDMA_CxTR1.DSEC）を個別にセキュアまたは非セキュアに設定することができます。

チャネルxがセキュア状態のリンクリスト･モードに設定されている場合、LPDMAメモリから、そのレジスタ･ファイルへの次のリンクリスト･データ構造のロードは、マスタ･ポートを経由したセキュア転送により自動的に実行されます。

LPDMAは、他のペリフェラルに各LPDMAチャネルxのセキュア／非セキュア状態が常に通知されるようにするため、LPDMA_SECCFGRを反映してセキュリティに対するDMAチャネルの状態を生成します。

また、LPDMAは、セキュアLPDMAレジスタに対する不正な非セキュア･アクセスに関し、セキュリティの不正アクセス･パルス信号を生成します。この信号はTrustZone割込みコントローラに送られます。

セキュア･ソフトウェアがチャネルをセキュア状態から非セキュア状態に切り替える必要がある場合、セキュア･ソフトウェアはそのチャネルをアボートするか、セキュア･チャネルが完了するのを待ってから切り替えなくてはなりません。これは、チャネルを次の非セキュア転送に動的に再割り当てするために必要です。非セキュア･ソフトウェアは、これを行うことが許可されておらず、次の転送に向けLPDMA_CxCRを再プログラムできるようになるにはLPDMA_CxCR.EN = 0でなくてはならないためです。セキュア･ソフトウェアは、チャネルx（LPDMA_CxCR.RESET = 1）だけではなく、フルチャネルxレジスタ･ファイルもリセット値にリセットすることができます。

18.4.15 LPDMA特権／非特権チャネル

あらゆるチャネルxは、LPDMA_PRIVCFGR.PRIVxによる特権エージェントの設定に従って、特権または非特権ハードウェア･リソースとなります。

チャネルxが特権エージェントにより特権状態に設定されている場合、次のアクセス制御規則が適用されます。

- このチャネルのレジスタ･フィールドへの非特権読出しアクセスは、非特権エージェントで読み出すことができるLPDMA_PRIVCFGR、LPDMA_SECCFGR、を除いて、強制的に0を返します。
- このチャネルのレジスタフィールドへの非特権書込みアクセスによる影響はありません。

チャネルが特権（非特権）状態に設定されると、ソースおよびデスティネーション･データ転送は、AHBマスタ･ポート経由の特権（非特権）転送です。

チャネルが特権（または非特権）状態のリンクリスト･モードに設定されている場合、LPDMAメモリからのそのレジスタ･ファイルへの次のリンクリスト･データ構造のロードは、マスタ･ポートを経由した特権（または非特権）転送で自動的に実行されます。

LPDMAは、他のペリフェラルに各DMAチャネルxの特権／非特権状態が常に通知されるようにするため、LPDMA_PRIVCFGRを反映して特権に対するDMAチャネルの状態を生成します。

さらに、LPDMAは、特権LPDMAレジスタに対する不正な非特権アクセスに関し、特権の不正アクセス·パルス信号を生成します。この信号を使用するかどうかは、製品の種類によって異なります（詳細については、システム·セキュリティのセクションを参照）。

特権ソフトウェアがチャネルを特権状態から非特権状態に切り替える必要がある場合、特権ソフトウェアはそのチャネルをアボートするか、特権チャネルが完了するのを待ってから切り替えなくてはなりません。これは、チャネルを次の非特権転送に動的に再割り当てするために必要です。非特権ソフトウェアは、これを行うことが許可されておらず、次の転送に向け、LPDMA_CxCRを再プログラムできるようになるにはLPDMA_CxCR.EN=0でなくてはならないためです。特権ソフトウェアは、チャネルx（LPDMA_CxCR.RESET = 1）だけではなく、フルチャネルxレジスタ·ファイルもリセット値にリセットすることができます。

18.4.16 LPDMAエラー管理

LPDMAは次に示すように、根本原因に応じて転送エラーを管理しユーザに報告することができます。

データ転送エラー

転送元または転送先への（AHBのシングルとしての）バス·アクセスに関して。

- 転送元または転送先のターゲットはAHBエラーを報告します。
- プログラムされたチャネル転送が停止します（LPDMA_CxCR.ENがLPDMAハードウェアでクリア）。チャネル·ステータス·レジスタはアイドル状態（LPDMA_CxSR.IDLEF = 1）とデータ·エラー（LPDMA_CxSR.DTEF = 1）を報告します。
- LPDMAデータ転送エラーの後、ユーザは、保護属性を含む、製品が定義した転送元と転送先のメモリ·マッピングを考慮して、デバッグ·セッションを実行しなくてはなりません。
- LPDMA データ転送エラーの後、ユーザはチャネルのリセットを発行して（LPDMA_CxCR.RESETをセット）、ハードウェアのLPDMAチャネル·データ·パスをリセットしてから、そのチャネルを次の転送に対して有効にしなくてはなりません。

リンク転送エラー

メモリにあるプログラムされたLLIからのLPDMAチャネル·レジスタの暫定的な更新に関して。

- リンクリスト·メモリはAHBエラーを報告します。
- プログラムされたチャネル転送が停止し（LPDMA_CxCR.ENがLPDMAハードウェアでクリアされる）、チャネル·ステータス·レジスタがアイドル状態（LPDMA_CxSR.IDLEF = 1）と、リンク·エラー（LPDMA_CxSR.ULEF = 1）を報告します。
- LPDMAのリンク·エラーの後、ユーザは、保護属性を含む、製品が定義したリンクリスト·データ構造（LPDMA_CxLBARおよびLPDMA_CxLLR）のメモリ·マッピングを考慮して、デバッグ·セッションを実行しなくてはなりません。
- LPDMAリンク·エラーの後、ユーザはリンクリスト·レジスタ·ファイル（LPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDAR、およびLPDMA_CxLLR）を明示的に書き込んでから、そのチャネルを次の転送のために有効にしなくてはなりません。

ユーザ設定エラー

非認証ユーザ設定でのLPDMA転送の暫定実行に関して。

- プログラムされたチャネル転送を無効にし（LPDMAハードウェアによりLPDMA_CxCR.ENを強制しクリアし）、次の非認証プログラム·データ転送が実行されることを回避します。チャネル·ステータス·レジスタがアイドル状態（LPDMA_CxSR.IDLEF=1）とユーザ設定エラー（LPDMA_CxSR.USEF = 1）を報告します。

- LPDMAユーザ設定エラーの後、ユーザはLPDMAチャネル·プログラミングを考慮に入れて、デバッグ·セッションを実行しなくてはなりません。ユーザ設定エラーは次のどれかが原因となっている可能性があります。
  - 次の LLI_1 によるプログラムされたこの値の更新が無い、プログラムされた null の転送元ブロック·サイズ（LPDMA_CxBR1.BNDT[15:0] = 0およびLPDMA_CxLLR.UB1 = 0）
  - プログラムされた null ではない転送元ブロック·サイズが、ソース·シングル転送のプログラムされたデータ幅の倍数ではない（LPDMA_CxBR1.BNDT[2:0]に対してLPDMA_CxTR1.SDW_LOG2[1:0]）
  - プログラムされたアンアラインド転送元開始アドレスが、ソース·シングル転送のプログラムされたデータ幅の倍数ではない（LPDMA_CxSAR[2:0]に対してLPDMA_CxTR1.SDW_LOG2[1:0]）
  - プログラムされたアンアラインド転送先開始アドレスが、デスティネーション·シングル転送のプログラムされたデータ幅の倍数ではない（LPDMA_CxDAR[2:0]に対してLPDMA_CxTR1.DDW_LOG2[1:0]）
  - プログラムされたダブルワードの転送元データ幅（LPDMA_CxTR1.SDW_LOG2[1:0] = 0b11）
  - プログラムされたダブルワードの転送先データ幅（LPDMA_CxTR1.DDW_LOG2[1:0] = 0b11）
  - nullデータ転送による、プログラムされたリンクリスト·アイテムLLI_{n+1}（LPDMA_CxLLR.UB1 = 1およびLPDMA_CxBR1.BNDT = 0）

18.4.17 LPDMA自律モード

プログラムされたリンクリスト転送をLPDMAが実行する際の動的電力消費を削減するため、RCCが対応するLPDMAイネーブル制御ビットでプログラムされている場合、LPDMAハードウェアはそのクロック·ゲーティングを自動で管理し、デバイスがRUN、SLEEP、またはSTOPモードのときには必ずRCCにクロック·リクエスト出力信号を生成します。

RCCプログラミングの詳細についてはリファレンス·マニュアルのRCCセクションを参照してください。

LPDMA自律機能に対するデバイスの低消費電力モードの可用性の詳細に関しては、セクション18.3.2を参照してください。

次に示すように、ユーザはLPDMA_CxTR2により、特定のLPDMA転送の実行をLPDMAチャネルxのLLI_nレベルでプログラムしてスケジューリングすることができます。

- ソフトウェアが、入力トリガとリクエストのためのTRIGM[1:0]、TRIGPOL[1:0]、TRIGSEL[4:0]、SWREQ、およびREQSEL[4:0]で転送の入力を制御し、条件付けします。
- ソフトウェアが、TCEM[1:0]で転送出力の制御および信号出力を行い、転送完了イベントを生成するかどうか、および関連する1/2データ転送イベントを生成するかどうかを決めています。

詳細は、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）を参照してください。

出力チャネルx転送完了イベントlpdma_chx_tcは、このイベントがループバックし、LPDMAレベルで接続されている場合（セクション18.3.5を参照）、チャネルの選択した入力トリガとしてプログラムすることが可能で、クリアされた転送完了フラグ（TCF）なしで、自律的に細かいDMAチャネル間転送スケジューリングを可能にします。

任意のLPDMAチャネルxは、次のいずれかの条件でそのクロック･リクエストをアサートします。

- 実行する予定の次の転送がトリガ（LPDMA_CxTR2.TRIGPOL[1:0]およびLPDMA_CxTR2.TRIGM[1:0]）により条件付けされるようにプログラムされている場合、そのトリガのヒットが発生している時のみ。
- 実行する予定の次の転送がトリガにより条件付けされない場合は次のとおりです。
  - LPDMA_CxTR2.SWREQ = 0の場合、選択されたペリフェラルによりハードウェア･リクエストがアサートされたときのみ。
  - LPDMA_CxTR2.SWREQ = 1（メモリ間、GPIOとメモリ相互の間）の場合、GPDMAが有効になるとすぐ。

次の条件がすべて満たされるとすぐに、LPDMAチャネルxがそのクロック･リクエストをリリースします。

- 実行予定の転送が完了している。
- LPDMAチャネルxはすぐには準備が整わず、次の転送の実行がリクエストされていない。
- チャネルx割込みが立っている場合、この割込みを発生させる可能性のある状態レジスタのすべてのフラグがソフトウェア･エージェントによりクリアされます。

1つのチャネルによってそのクロック･リクエストがアサートされる場合、LPDMAによってそのクロック･リクエストがRCCにアサートされます。チャネルによってクロック･リクエストがアサートされない場合、LPDMAによってRCCへのクロック･リクエストが解除されます。

18.5 デバッグ･モードでのLPDMA

マイクロコントローラがデバッグ･モードに入ると（コアが停止）、あらゆるチャネルxを、DBGMCUモジュールのプログラマブル制御ビットに応じて、個別に継続（デフォルト）またはサスペンドにすることが可能です。

注： **デバッグ･モードでは、LPDMA_CxSR.SUSPF はDBGMCU モジュールのプログラマブル制御ビットによるサスペンドの影響は受けません。この場合、LPDMA_CxSR.IDLEF をチェックしてチャネル･サスペンドの完了ステータスを知ることができます。**

18.6 低消費電力モードでのLPDMA

表151.低消費電力モードがLPDMAに与える影響

モード	説明
SLEEP	影響はありません。LPDMA割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	STOPモードに入っても、LPDMAレジスタの内容は保持されます。LPDMAレジスタの内容はメモリから次のリンクリスト･アイテムにより自律的に更新され、自律的なデータ転送を実行することが可能です。デバイスは、LPDMA割込みによってSTOPモードから復帰することが可能です。
STANDBY	LPDMAはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. どのSTOPモードがサポートされるかについては、セクション18.3.2を参照してください。

18.7 LPDMA割込み

各チャネルと各CPUには個別に1つのLPDMA割込みラインがあります（デバイスにいくつかある場合）。

表152.LPDMA割込みリクエスト

項目（割込みの略称）	割込みイベント	割込みイネーブル	イベントフラグ	イベントのクリア方法
LPDMA_CHx	転送完了	LPDMA_CxCR.TCIE	LPDMA_CxSR.TCF	LPDMA_CxFCR.TCFに1を書き込む
	1/2転送	LPDMA_CxCR.HTIE	LPDMA_CxSR.HTF	LPDMA_CxFCR.HTFに1を書き込む
	データ転送エラー	LPDMA_CxCR.DTEIE	LPDMA_CxSR.DTEF	LPDMA_CxFCR.DTEFに1を書き込む
	更新リンク·エラー	LPDMA_CxCR.ULEIE	LPDMA_CxSR.ULEF	LPDMA_CxFCR.ULEFに1を書き込む
	ユーザ設定エラー	LPDMA_CxCR.USEIE	LPDMA_CxSR.USEF	LPDMA_CxFCR.USEFに1を書き込む
	サスペンド状態	LPDMA_CxCR.SUSPIE	LPDMA_CxSR.SUSPF	LPDMA_CxFCR.SUSPFに1を書き込む
	トリガ·オーバーラン	LPDMA_CxCR.TOFIE	LPDMA_CxSR.TOF	LPDMA_CxFCR.TOFに1を書き込む

LPDMAチャネルxイベントは次のいずれかの場合があります。

- 転送完了
- 1/2転送完了
- 転送エラー、その原因は次のいずれか。
  - データ転送エラー
  - 更新リンク·エラー
  - ユーザ設定エラー、完了サスペンション
- トリガ·オーバーラン

注： **チャネルx転送完了イベントが発生すると、出力信号lpdma_chx_tcが1クロック·サイクルのハイ·パルスとして生成されます。**

次の両方の条件を満足すると、あらゆるxxイベントに続いて割込みが生成されます。

- 対応する割込みイベントxxが有効である（LPDMA_CxCR.xxIE = 1）。
- 対応するイベント·フラグがクリアされている（LPDMA_CxSR.xxF = 0）。これは、前の同じxxイベント発生の後に、ソフトウェア·エージェントが対応するxxフラグのクリア制御ビットに1を書き込んだはずだということを意味します（LPDMA_CxFCR.xxFに1を書き込む）。

TCF（転送完了）とHTF（1/2転送）イベントの生成は次のようにLPDMA_CxTR2.TCEM[1:0]によって制御されます。

- 転送完了イベントは、転送完了イベント·モード（LPDMA_CxTR2.TCEM[1:0]）に応じて、次のLLI（がある場合）のアップロードを含むLLI転送完了またはブロック転送完了、あるいはフル·リンクリスト完了のいずれかです。
- 1/2転送イベントは1/2ブロック転送です。転送元ブロックのバイト·サイズの半分（LPDMA_CxBR1.BNDT[15:0]/2を整数値に切り上げ）が転送先に転送されたときに1/2ブロック転送となります。

詳細は、LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）を参照してください。

次のいずれかの状況で転送エラーが立ちます。

- 転送元から、または転送先へのシングル･データ転送時（DTEF）
- メモリにあるプログラムされたLLIによるLPDMAチャネル･レジスタの更新時（ULEF）
- 非認証設定でのLPDMAチャネルの暫定実行時（USEF）

  ユーザはLPDMAチャネル･プログラミングをUSEFの根本原因リストと対比して修正するデバッグ･セッションを行わなくてはなりません（セクション18.4.16を参照）。

トリガ･オーバーランはトリガ･ヒットのメモリ書込みとトリガ･オーバーラン･フラグの生成に記載します。

18.8 LPDMAレジスタ

LPDMAレジスタは32bitワード･データでのアラインド･アクセスでアクセスしなくてはなりません。

18.8.1 LPDMAセキュア設定レジスタ（LPDMA_SECCFGR）

アドレスオフセット：0x000

リセット値：0x0000 0000

対応するチャネルxがロックされている場合（LPDMA_RCFGLOCKR.LOCKx = 1）、書込みアクセスはビット･レベルで無視されます。

このレジスタへの書込みアクセスはセキュアおよび特権でなくてはなりません。読出しアクセスはセキュアまたは非セキュア、特権または非特権です。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

このレジスタは、LPDMAチャネルの初期化／クローズ時（LPDMA_CxCR.EN = 0のとき）にビット･レベルでプログラムし、あらゆるチャネルxを個別にセキュアまたは非セキュア･ワールドに安全に割り当てる必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC3	SEC2	SEC1	SEC0
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **SECx**：チャネルx（x = 3 ~ 0）のセキュア状態

0：非セキュア

1：セキュア

18.8.2 LPDMA特権設定レジスタ（LPDMA_PRIVCFGR）

アドレスオフセット：0x004

リセット値：0x0000 0000

このレジスタへの書込みアクセスは特権でなくてはなりません。読出しアクセスは特権または非特権、セキュアまたは非セキュアで行うことができます。

対応するチャネルxがロックされている場合（GPDMA_RCFGLOCKR.LOCKx=1）、書込みアクセスはビット･レベルで無視されます。

このレジスタにはセキュアな情報と非セキュアな情報を混在させることができます。チャネルxがセキュアに設定された場合（LPDMA_SECCFGR.SECx = 1）、PRIVxビットはセキュア（および特権）エージェントによってのみ書き込むことが可能です。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

このレジスタは、LPDMAチャネルの初期化／クローズ時（LPDMA_CxCR.EN = 0）にビット･レベルでプログラムし、あらゆるチャネルxを特権または非特権ワールドに割り当てる必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV3	PRIV2	PRIV1	PRIV0
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PRIVx**：チャネルx（x = 3~0）の特権状態

0：非特権

1：特権

18.8.3 LPDMA設定ロック･レジスタ（LPDMA_RCFGLOCKR）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタは、たとえばブート時などに、あらゆるDMAチャネル／リソースのセキュア特権属性を個別にロックするために（たとえばブート時に、あらゆるチャネルxのLPDMA_SECCFGRおよびLPDMA_PRIVCFGRの設定をロックするために）、セキュア特権属性のソフトウェア･エージェントにより書き込むことが可能です。

読出しアクセスは特権または非特権、セキュアまたは非セキュアで行うことができます。

注： TZEN = 0の場合、このレジスタは書込みできません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK3	LOCK2	LOCK1	LOCK0
												rs	rs	rs	rs

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **LOCKx**：グローバルDMAがリセットされるまで、LPDMA_SECCFGR.SECxおよびLPDMA_PRIVCFGR.PRIVxの設定をロックします（x = 3 ~ 0）。

このビットはリセット後にクリアされ、一旦セットされると、グローバルDMAがリセットされるまでリセットすることはできません。

0：チャネルxのセキュア特権設定が書込み可能です。

1：チャネルxのセキュア特権設定が書込み不可です。

18.8.4 LPDMA非セキュアマスク割込みステータス･レジスタ（LPDMA_MISR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

レジスタは読出しレジスタです。

これは非セキュア･レジスタで、各非セキュア･チャネルxのマスク割込みステータス･ビットMISxを含みます（LPDMA_SECCFGR.SECx = 0のチャネルx設定）。これはLPDMA_CxSRのすべてのフラグの論理和で、各転送元フラグは対応するLPDMA_CxCRの割込みイネーブル･ビットにより有効になっています。

すべてのビットは、LPDMA_CxFCRの対応するフラグ･クリアビットに1を書き込むと、ハードウェアによりデアサートされます。

チャネルxがセキュア状態の場合（LPDMA_SECCFGR.SECx = 1）、このチャネルxのマスク割込みステータス･ビットMISxへの読出しアクセスは0を返します。

このレジスタには、各チャネルの特権状態LPDMA_PRIVCFGR.PRIVxによっては、特権情報と非特権情報が混在する可能性があります。特権ソフトウェアは非セキュア割込みステータスをすべて読み出すことができます。非特権ソフトウェアに許可されるのは非特権（かつ非セキュア）チャネルのステータス読出しのみで、その他の特権ビットフィールドはゼロが返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS3	MIS2	MIS1	MIS0
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **MISx**：チャネルxのマスク割込みステータス（x = 3 ~ 0）

0：非セキュア･チャネルxで割込みは発生していません

1：非セキュア･チャネルxで割込みが発生しました

18.8.5 LPDMAセキュア･マスク割込みステータス･レジスタ（LPDMA_SMISR）

アドレスオフセット：0x010

リセット値：0x0000 0000

これはセキュア読出しレジスタで、各セキュア･チャネルxのマスク割込みステータス･ビットMISxを含みます（LPDMA_SECCFGR.SECx = 1）。これはLPDMA_CxSRフラグすべての論理和で、各転送元フラグは、対応するLPDMA_CxCR割込みイネーブル･ビットにより有効になっています。

各ビットは、対応するLPDMA_CxFCRフラグ･クリア･ビットに1をセキュアに書き込むと、ハードウェアによりデアサートされます。

このレジスタは非セキュア･チャネルに関する情報は何も含みません。

このレジスタには、各チャネルの特権状態LPDMA_PRIVCFGR.PRIVxによっては、特権情報と非特権情報が混在する可能性があります。特権ソフトウェアはセキュア割込みステータスをすべて読み出すことができます。非特権ソフトウェアに許可されるのは非特権およびセキュア･チャネルのステータス読出しのみで、その他の特権ビット･フィールドはゼロが返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS3	MIS2	MIS1	MIS0
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **MISx**：セキュア･チャネルxのマスク割込みステータス（x = 3 ~ 0）

0：セキュア･チャネルxで割込みは発生していません

1：セキュア･チャネルxで割込みが発生しました

18.8.6 LPDMAチャネルxのリンクリスト･ベースのアドレス･レジスタ（LPDMA_CxLBAR）

アドレスオフセット：0x050 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは特権ソフトウェアによって書き込まれる必要があります。これは、チャネルx LPDMA_PRIVCFGR.PRIVxの特権状態により、特権読取りが可能かどうかが異なります。

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）により、セキュアまたは非セキュアとなります。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルをベースとするこのレジスタは、あるチャネルxに対して、メモリ領域のリンクリスト･ベースのアドレスで、ここからLPDMA転送のプログラムされたシーケンスを記述するLLIが、条件付きで自動的に更新されます。

64KBに合わせたチャネルxリンクリストのベース･アドレスは、各LLIのワード整列アドレス･オフセットを定義する、16bit LPDMA_CxLLRレジスタによりオフセットされています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LBA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:16 **LBA[31:16]**：LPDMAチャネルxのリンクリスト･ベース･アドレス

ビット15:0 予約済みであり、リセット値に保持する必要があります。

18.8.7 LPDMAチャネルxフラグ･クリア･レジスタ（LPDMA_CxFCR）

アドレスオフセット：0x05C + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

これは書込みレジスタで、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	w	w	w	w	w	w	w								

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOF**：トリガ･オーバーラン･フラグのクリア
0：影響なし。
1：対応するTOFフラグをクリアします

ビット13 **SUSPF**：完了したサスペンド･フラグのクリア
0：影響なし。
1：対応するSUSPFフラグがクリアされました

ビット12 **USEF**：ユーザ設定エラー･フラグのクリア
0：影響なし。
1：対応するUSEFフラグがクリアされました

ビット11 **ULEF**：更新リンク転送エラー･フラグのクリア
0：影響なし。
1：対応するULEFフラグがクリアされました

ビット10 **DTEF**：データ転送エラー･フラグのクリア
0：影響なし。
1：対応するDTEFフラグがクリアされました

ビット9 **HTF**：1/2転送フラグのクリア
0：影響なし。
1：対応するHTFフラグがクリアされました

ビット8 **TCF**：転送完了フラグのクリア
0：影響なし。
1：対応するTCFフラグがクリアされました

ビット7:0 予約済みであり、リセット値に保持する必要があります。

18.8.8 LPDMAチャネルxステータス･レジスタ（LPDMA_CxSR）

アドレスオフセット：0x060 + 0x80 * x（x = 0 ~ 3）

リセット値：0x0000 0001

これはチャネル･ステータスを報告する読出しレジスタです。

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDLEF
	r	r	r	r	r	r	r								r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOF**：トリガ･オーバーラン･フラグのクリア
0：影響なし。
1：対応するTOFフラグをクリアします

ビット13 **SUSPF**：完了したサスペンド･フラグ
0：完了したサスペンド･イベントはありません
1：サスペンド･イベントの完了が発生しました

ビット12 **USEF**：ユーザ設定エラー･フラグ

0：ユーザ設定エラー･イベントはありません

1：ユーザ設定エラー･イベントが発生しました

ビット11 **ULEF**：更新リンク転送エラー･フラグ

0：更新リンク転送エラー･イベントはありません

1：メモリからリンクリスト･レジスタを更新しているときに、マスタ･バス･エラー･イベントが発生しました

ビット10 **DTEF**：データ転送エラー･フラグ

0：データ転送エラー･イベントはありません

1：データ転送でマスタ･バス･エラー･イベントが発生しました

ビット9 **HTF**：1/2転送フラグ

0：1/2転送イベントは発生していません

1：1/2転送イベントが発生しました。

1/2転送イベントは、転送元ブロックのバイト･サイズの半分（LPDMA_CxBR1.BNDT[15:0]/2を整数値に切り上げ）が転送先に転送されたときに発生する1/2ブロック転送です。

ビット8 **TCF**：転送完了フラグ

0：転送完了イベントは発生していません

1：転送完了イベントが発生しました

転送完了イベントは、転送完了イベント･モード（LPDMA_CxTR2.TCEM[1:0]）に応じて、次のLLI（がある場合）のアップロードを含むLLI転送完了またはブロック転送完了、あるいはフル･リンクリスト完了のいずれかです。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **IDLEF**：アイドル･フラグ

0：チャネルはアイドル状態ではありません

1：チャネルはアイドル状態です

このアイドル･フラグは、チャネルが有効なチャネル設定でイネーブルになっていると（LPDMA_CxCR.EN = 1）、ハードウェアによってデアサートされます（USEFがすぐには報告されない）。このアイドル･フラグは、チャネルが（サスペンドまたはディスエーブル状態で）アイドル状態に戻った時に、ハードウェアによりアサートされるか、ハード･リセット後にアサートされます。

18.8.9 LPDMAチャネルx制御レジスタ（LPDMA_CxCR）

アドレスオフセット：0x64 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタはチャネルの制御（アクティベート、サスペンド、アボート、ディスエーブル）に使われます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIO[1:0]		Res.	Res.	Res.	Res.	Res.	LSM
								rw	rw						rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TOIE	SUSPIE	USEIE	ULEIE	DTEIE	HTIE	TCIE	Res.	Res.	Res.	Res.	Res.	SUSP	リセット	EN
	rw	rw	rw	rw	rw	rw	rw						rw	w	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:22 **PRIO[1:0]**：チャネルx LPDMA転送の、他に対する優先レベル

00：優先度低、重み低

01：優先度低、重み中

10：優先度低、重み高

11：優先度高

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット21:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **LSM**：リンク･ステップ･モード

0：フル･リンクリストで実行し、最後のLLIの終わりで完了したチャネル（LPDMA_CxLLR = 0）。このリンク･アドレスの下位16bitはnull（LA[15:0] = 0）で、すべての更新ビットもnull（UT1 = UB1 = UT2 = USA = UDA = ULL = 0）です。そして、LPDMA_CxBR1.BNDT[15:0] = 0になります。

1：現在のLLIで**1回**実行されたチャネル

最初に、LPDMA_CxBR1.BNDT[15:0] = 0になるまで、現在の内部レジスタ･ファイルの定義に従い、ブロック転送が実行されます。その後、次のリンクリスト･データ構造がLPDMA_CxLLRの定義に従い、条件付きでメモリからアップロードされます。それでチャネルの実行は完了です。

注： このビットはEN = 0のときには書き込まなくてはなりません。このビットは、EN = 1のときは読出し専用です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TOIE**:トリガ･オーバーラン割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット13 **SUSPIE**：完了サスペンド割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット12 **USEIE**：ユーザ設定エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット11 **ULEIE**：更新リンク転送エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット10 **DTEIE**：データ転送エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット9 **HTIE**：1/2転送完了割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット8 **TCIE**：転送完了割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SUSP**：サスペンド

RESET（ビット1）フィールドへ1を書き込むと、ビット2への書込みに関係なく、ハードウェアはこのビットをデアサートします。その他の場合：

ソフトウェアはアクティブなチャネル（マスタ·ポートを経由したDMA転送が進行中のチャネル）をサスペンドするために1を書き込まなくてはなりません。

ソフトウェアはサスペンドしたチャネルをレジュームするために、図72で詳細を説明しているプログラミング·シーケンスに従って0を書き込まなくてはなりません。

0：書込み：チャネルのレジューム、読出し：チャネルのサスペンドなし

1：書込み：チャネルをサスペンド、読出し：サスペンドしたチャネル

ビット1 **RESET**：リセット

このビットは書込み専用です。0を書き込んでも影響はありません。1の書込みは次の項目のリセットを意味します：チャネルの内部状態、SUSPおよびENビット（それぞれ、ビット2およびビット0への書込み内容には関係ありません）。

リセットは、チャネルが定常状態のときに有効で、これは次のどれかを意味してます。

-サスペンドした状態のアクティブなチャネル（LPDMA_CxSR.SUSPF = 1およびLPDMA_CxSR.IDLEF = LPDMA_CxCR.EN = 1）

-無効状態のチャネル（LPDMA_CxSR.IDLEF = 1およびLPDMA_CxCR.EN = 0）

RESETを書き込んだ後、このチャネルを継続して使用するためには、ハードウェアにより変更される設定レジスタ（LPDMA_CxBR1, LPDMA_CxSARおよびLPDMA_CxDAR）など、チャネルを明示的に再設定してから、チャネルを再度イネーブルにすることが必要です（図73のプログラミング·シーケンス参照）。

0：チャネルのリセットなし

1：チャネルのリセット

ビット0 **EN**：イネーブル

RESET（ビット1）フィールドへ1を書き込むと、ビット0への書込みに関係なく、ハードウェアはこのビットをデアサートします。その他の場合：

このビットは、転送エラー（マスタ·バス·エラーまたはユーザ設定エラー）またはチャネル転送完了（たとえば、LLIのシングル実行の最後にLSM = 1になっている場合など、チャネルの設定準備が完了している）時にハードウェアによりデアサートされます。

その他の場合、このビットは、ソフトウェアでアサートできます。

このENビットへ0を書き込んでも無視されます。

0：書込み：無視、読出し：チャネル無効化

1：書込み：チャネルの有効化、読出し：有効になったチャネル

18.8.10 LPDMAチャネルx転送レジスタ1（LPDMA_CxTR1）

アドレスオフセット：0x090 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、セキュアDSECとSSECを除き、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）に従って、セキュアまたは非セキュアとなり、LPDMA_PRIVCFGR.PRIVxのチャネルxの特権状態に従って、特権または非特権となります。

このレジスタは、チャネルxの転送を制御します。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了したら、このレジスタに書き込まなくてはなりません。ハードウェアによりLPDMA_CxCR.ENがデアサートされます。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、LPDMA_CxLLR.UT1 = 1の場合、このレジスタはLPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DSEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DINC	Res.	DDW_LOG2 [1:0]	
rw												rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSEC	Res.	Res.	Res.	PAM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SINC	Res.	SDW_LOG2 [1:0]	
rw				rw								rw		rw	rw

ビット31 **DSEC**：転送先へのLPDMA転送のセキュリティ属性

LPDMA_SECCFGR.SECx = 1で、アクセスがセキュアな場合。

0：LPDMA転送は非セキュアです。

1：LPDMA転送はセキュアです。

これはセキュア･レジスタ･ビットです。このビットはセキュア･ソフトウェアからのみ読み出すことができます。LPDMA_SECCFGR.SECx = 1のとき、このビットはセキュア･ソフトウェアにより書き込まれなくてはなりません。LPDMA_SECCFGR.SECx = 0のとき、セキュア書込みは無視されます。

LPDMA_SECCFGR.SECxがデアサートされると、このDSECビットも（チャネルのセキュア再設定で非セキュアとして）ハードウェアによりデアサートされ、転送先へのLPDMA転送は非セキュアになります。

ビット30:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **DINC**：転送先のシングルのインクリメント

0：シングルを固定

1：シングルを継続的にインクリメント

LPDMA_CxDARが指す転送先アドレスは、シングル転送後もそのまま保持されるか、シングル転送後に隣接データに対応するオフセット分だけインクリメントされます。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **DDW_LOG2[1:0]**：シングルの転送先データ幅の二進対数（バイト単位）

00：バイト

01：ハーフワード（2バイト）

10：ワード（4バイト）

11：ユーザ設定エラーが報告され、転送は行われませんでした

注： 8バイト･データ幅を設定すると、ユーザ設定エラーが報告され、転送は行われません。デスティネーション･シングル転送は、そのデータ幅による整列アドレスでなくてはなりません（開始アドレスLPDMA_CxDAR[2:0]に対してDDW_LOG2[1:0]）。そうしないとユーザ設定エラーが報告され、転送は行われまん。

ビット15 **SSEC**：転送元からのLPDMA転送のセキュリティ属性

LPDMA_SECCFGR.SECx = 1で、アクセスがセキュアな場合。

0：LPDMA転送は非セキュアです。

1：LPDMA転送はセキュアです。

これはセキュア･レジスタ･ビットです。このビットはセキュア･ソフトウェアからのみ読み出すことができます。LPDMA_SECCFGR.SECx = 1のとき、このビットはセキュア･ソフトウェアにより書き込まれなくてはなりません。LPDMA_SECCFGR.SECx = 0のとき、セキュア書込みは無視されます。

LPDMA_SECCFGR.SECxがデアサートされると、このSSECビットもハードウェアによりデアサートされ（チャネルのセキュア再設定で非セキュアとして）、転送元からのLPDMA転送は非セキュアになります。

ビット14:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **PAM**：パディング／配置モード

DDW_LOG2[1:0] = SDW_LOG2[1:0]の場合：シングル･デスティネーション転送とシングル･ソース転送のデータ幅が同じときには、このビットは無視されます。

それ以外の場合、次のエニュメレーションした値になり、条件PAM_1は転送先データ幅が転送元データ幅より大きい場合、条件PAM_2は転送先データ幅が転送元データ幅より大きい場合です。

条件：PAM_1

0：転送元データは、右詰めで、転送先データ幅まで0でパディングされた値で転送されます。

1：転送元データは、右詰めで、転送先データ幅まで符号拡張された値で転送されます。

条件：PAM_2

0：転送元データは、右詰めで、転送先データ幅に合わせて左側が切り捨てられた値で転送されます。

1：転送元データは、左詰めで、転送先データ幅に合わせて右側が切り捨てられた値で転送されます。

ビット10:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **SINC**：転送元のシングルのインクリメント

0：シングルを固定

1：シングルを継続的にインクリメント

LPDMA_CxSARが指す転送元アドレスは、シングル転送後もそのまま保持されるか、シングル転送後に隣接データに対応するオフセット分だけインクリメントされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **SDW_LOG2[1:0]**：シングルの転送元データ幅の二進対数（バイト単位）

00：バイト

01：ハーフワード（2バイト）

10：ワード（4バイト）

11：ユーザ設定エラーが報告され、転送は行われませんでした

注： 8バイト･データ幅を設定すると、ユーザ設定エラーが報告され、転送は行われません。転送元のブロック･サイズは転送元のデータ幅の倍数でなくてはなりません（LPDMA_CxBR1.BNDT[2:0]に対しSDW_LOG2[1:0]）。そうしないとユーザ設定エラーが報告され、転送は行われません。ソース･シングル転送は、そのデータ幅による整列アドレスでなくてはなりません（開始アドレスLPDMA_CxSAR[2:0]に対してSDW_LOG2[1:0]）。そうしないとユーザ設定エラーが報告され、転送は行われまん。

18.8.11 LPDMAチャネルx転送レジスタ2（LPDMA_CxTR2）

アドレスオフセット：0x094 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送を制御します。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了したら（ハードウェアがLPDMA_CxCR.ENをデアサートしたら）、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、LPDMA_CxLLR.UT2 = 1の場合、このレジスタはLPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TCEM[1:0]		Res.	Res.	Res.	Res.	TRIGPOL[1:0]		Res.	Res.	Res.	TRIGSEL[4:0]				
rw	rw					rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGM[1:0]		Res.	Res.	BREQ	Res.	SWREQ	Res.	Res.	Res.	Res.	REQSEL[4:0]				
rw	rw			rw		rw					rw	rw	rw	rw	rw

ビット31:30 **TCEM[1:0]**：転送完了イベント･モード

このビットは転送完了と1/2転送完了イベント生成の粒度を規定します。

00：ブロック･レベル（LPDMA_CxBR1.BNDT[15:0] = 0のとき）：完全な転送（1/2転送）イベントがブロックの終わり（1/2の終わり）に生成されます。

注： 最初のLLI_0データ転送がnull／voidの（LPDMA_CxBR1.BNDT[15:0] = 0で内部のレジスタ･ファイルにより直接プログラムされる）場合、完全な転送イベントも1/2転送イベントも生成されません。

01：00と同じです。

10：LLIレベル：LLIの更新（がある場合は）を含む、完全な転送イベントがLLI転送の最後に生成されます。データ転送がある場合、LLIデータ転送の半分で1/2転送イベントが生成されます（LLIデータ転送はブロック転送）。

注： 最初のLLI_0データ転送がnull／voidの（内部のレジスタ･ファイルにより、LPDMA_CxBR1.BNDT[15:0] = 0で直接プログラムされている）場合、1/2転送イベントは生成されず、LLI_1のロードが完了したときに、転送完了イベントが生成されます。

11：チャネル･レベル：最後のLLI転送の終わりに、完全な転送イベントが生成されます。最後のLLIのデータ転送の半分で 1/2 転送イベントが生成されます。最後の LLI ではリンク･アドレスLPDMA_CxLLR.LA[15:2]が0に更新され、すべてのLPDMA_CxLLR更新ビットがクリアされます（UT1、UT2、UB1、USA、UDA、およびULL）。チャネル転送が連続／無限の場合、イベントは生成されません。

ビット29:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **TRIGPOL[1:0]**：トリガ･イベント極性

このビットは、TRIGSEL[4:0]で定義される、選択したトリガ･イベント入力の極性を定義します。

00：トリガなし（マスクトリガ･イベント）

01：立上がりエッジでのトリガ

10：立下がりエッジでのトリガ

11：00と同じ

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **TRIGSEL[4:0]**：トリガ･イベント入力の選択

このビットは、TRIGPOL[1:0] = 00の場合、有効なトリガ･イベントでLPDMA転送のトリガ･イベント入力を（セクション18.3.5に従って）選択します。

ビット15:14 **TRIGM[1:0]**：トリガ･モード

このビットはトリガにより、転送粒度の条件付けを定義します。

チャネルxがTRIGPOL[1:0] = 0b00または0b11で有効になっている（LPDMA_CxCR.ENがアサートされている）場合、TRIGM[1:0]ビットは無視されます。

その他の場合、DMA転送が最低1つのトリガ･ヒットにより条件付けされます。

00：ブロック･レベルで、各ブロック転送の最初のシングル読出しが1つのヒット･トリガによって条件付けされます。

01：00と同じです。

10：リンク･レベル：LLIリンク転送は1つのヒット･トリガにより条件付けされます。LLIデータ転送（がある場合）は条件付けされません。

11：プログラムされたシングル･レベルで：プログラムされた各シングル読出しは1つのヒット･トリガにより条件付けされます。

チャネルxに対するトリガのLPDMA監視が開始されるのは、新しいアクティブなトリガ設定、つまり選択したトリガの立上がり／立下がりエッジ（それぞれ、TRIGPOL[1:0] = 0b01またはTRIGPOL[1:0] = 0b10）によりそのチャネルが有効化／ロードされるときです。

トリガされた未完了の（データまたはリンクの）転送中、このトリガの監視はアクティブのままとなり、新しいトリガが検出された場合、定義された立上がり／立下がりエッジとTRIGSEL[4:0]が変更されず、そのチャネルが有効である限り、このヒットは次の転送を許可するために内部的にメモリに書き込まれます。

TRIGSEL[4:0]またはTRIGPOL[1:0]の新しい値でLPDMA_CxTR2を更新する次のLLI_{n+1}の転送が、前に定義したLLI_nトリガのメモリ書込みされたヒットのモニタリング、トラッシングをリセットします。

最初の新しいトリガhit_{n+1}をメモリに書き込んだ後、別の2つめのトリガhit_{n+2}を検出し、hit_nにトリガされた転送が完了していない場合、hit_{n+2}は失われメモリに書き込まれません。そして、トリガ･オーバーラン･フラグが報告され（LPDMA_CxSR.TOF = 1）、有効であれば（LPDMA_CxCR.TOIE = 1）割込みが生成されます。トリガのオーバーランでハードウェアにより、チャネルが自動的に無効になることはありません。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 BREQ：**ブロック･ハードウェア･リクエスト**

チャネルxがSWREQ = 1（メモリ間転送のソフトウェア･リクエスト）で有効になっている場合（LPDMA_CxCR.ENがアサートされている場合）、このビットは無視されます。その他の場合：

0：選択されたハードウェア･リクエストが、シングル･レベルでのハードウェア･リクエスト／確認プロトコルでペリフェラルによりドライブされます。

1：選択されたハードウェア･リクエストが、ブロック･レベルでのハードウェア･リクエスト／確認プロトコルでペリフェラルによりドライブされます（セクション18.3.4を参照）。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **SWREQ**：ソフトウェア･リクエスト

このビットはLPDMA_CxCR.ENアサートされた場合、内部的に考慮されます。

0：ソフトウェア･リクエストは無し。選択されたハードウェア･リクエスト、REQSEL[4:0]、は考慮されます。

1：メモリ間転送のソフトウェア･リクエスト。REQSEL[4:0]に従ったデフォルトで選択されたハードウェア･リクエストは無視されます。

ビット8:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **REQSEL[4:0]**：DMAハードウェア･リクエストの選択

チャネルxがSWREQ = 1（メモリ間転送のソフトウェア･リクエスト）で有効になっている場合（LPDMA_CxCR.ENがアサートされている場合）、これらのビットは無視されます。その他の場合、選択されたハードウェア･リクエストはセクション18.3.3に従って内部的に考慮されます。

注意： ユーザは同じ入力ハードウェア･リクエスト（同じREQSEL[4:0]値）を異なる有効なDMAチャネルに割り当ててはなりません（これらのチャネルにはLPDMA_CxCR.EN = 1およびLPDMA_CxTR2.SWREQ = 0を割り当てます）。DMAでは、同時に有効になっているチャネルが同じハードウェア･ペリフェラル･リクエスト信号により誤って設定されているケースを、ハードウェアがサポートすることは意図していません。また、ユーザ設定エラーは報告されません。

18.8.12 LPDMAチャネルxブロック･レジスタ1（LPDMA_CxBR1）

アドレスオフセット：0x098 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、チャネルxの転送をブロック･レベルで制御します。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルxが完了し（その後、ハードウェアがLPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時。

- LPDMA_CxLLR.UB1 = 1の場合、このレジスタは、メモリにある次のLLIからDMAにより、自動的に更新されます。
- LPDMA_CxLLR.UB1 = 0で、少なくとも1つのリンクリスト･レジスタがメモリの中の次のLLIから更新されることになっている場合、このレジスタは自動的、内部的にフィールドBNDT[15:0]のプログラムされた値で復元されます。
- すべての更新ビットLPDMA_CxLLR.Uxxがnullで、LPDMA_CxLLR.LA[15:0] ≠ 0の場合、現在のLLIは最後の1つで、連続的に実行されます。このレジスタは、この最後の各LLIの実行後に、BNDT[15:0]にプログラムされた値で自動的、内部的に復元されます。
- LPDMA_CxLLR = 0の場合、このレジスタとBNDT[15:0]はnullのままとなり、チャネルxは完了します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BNDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BNDT[15:0]**：転送元から転送するデータ･バイトのブロック数

転送元から転送されてきたブロックのサイズ。チャネルが有効な場合、このフィールドは読出し専用となりデクリメントされます。これは現在の転送元ブロックにある転送する残りのデータ項目の数を示すものです。BNDT[15:0]はバイト数でプログラムされ、最大の転送元ブロック･サイズは64KB - 1です。

一旦、最後のデータ転送が完了したとき（BNDT[15:0] = 0）:

- LPDMA_CxLLR.UB1 = 1の場合、このフィールドはメモリのLLIにより更新されます。
- LPDMA_CxLLR.UB1 = 0で、nullではないUxx更新ビットが最低1つある場合、このフィールドはプログラムされた値に内部的に復元されます。
- すべてのLPDMA_CxLLR.Uxxが0で、LPDMA_CxLLR.LA[15:0] = 0の場合、このフィールドはプログラムされた値（無限／連続の最後のLLI）に内部的に復元されます。
- LPDMA_CxLLR = 0の場合、このフィールドは最後のLLIデータ転送後に0に維持されます。

注： nullではない転送元のブロック･サイズは、転送元のデータ幅の倍数でなくてはなりません（BNDT[2:0]に対しLPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

18.8.13 LPDMAチャネルx転送元アドレス･レジスタ（LPDMA_CxSAR）

アドレスオフセット：0x09C + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは転送の転送元開始アドレスを設定します。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

このレジスタはLPDMA_CxCR.EN = 1のときには読出し専用で、転送元からの次のシングル転送のアドレスを反映するように、ハードウェアにより継続的に更新されます。

チャネルが完了し（その後、ハードウェアがLPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、LPDMA_CxLLR.USA = 1の場合、このレジスタはLPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SA[31:0]**：転送元アドレス

このフィールドは次のデータを読み込むアドレスを示すものです。

チャネルの操作に際し、転送元のアドレス指定モード（LPDMA_CxTR1.SINC）に従い、このフィールドは、各シングル転送の転送元データの後ろで固定のままとなるか、データ幅（LPDMA_CxTR1.SDW_LOG2[1:0]）によりインクリメントされ、データを読み出す次のアドレスを反映するようになっています。

リンクリスト･モードでは、LLIがLPDMA_CxLLR.USA = 1に設定されている場合、LLIデータ転送が完了した後、このレジスタがLPDMAによりメモリから自動的に更新されます。

注： 転送元アドレスは、プログラムされたソース･シングルのデータ幅に合わせなくてはなりません（SA[32:0]に対しLPDMA_CxTR1.SDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

18.8.14 LPDMAチャネルx転送先アドレス･レジスタ（LPDMA_CxDAR）

アドレスオフセット：0x0A0 + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは転送の転送先開始アドレスを設定します。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

このレジスタはLPDMA_CxCR.EN = 1のときには読出し専用で、転送先への次のシングル転送のアドレスを反映するように、ハードウェアにより継続的に更新されます。

チャネルが完了し（その後、ハードウェアがLPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、LPDMA_CxLLR.UDA = 1の場合、このレジスタはLPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DA[31:0]**：転送先アドレス

このフィールドは次のデータを書き出すアドレスを示すものです。

チャネルの操作に際し、転送元のアドレス指定モード（LPDMA_CxTR1.DINC）に従い、このフィールドは、各シングル転送の転送先データの後ろで固定のままとなるか、データ幅（LPDMA_CxTR1.DDW_LOG2[21:0]）によりインクリメントされ、データを書き込む次のアドレスを反映するようになっています。

リンクリスト･モードでは、LLIがLPDMA_CxLLR.UDA = 1に設定されている場合、LLIデータ転送が完了した後、このレジスタがDMAによりメモリから自動的に更新されます。

注： 転送先アドレスは、プログラムされたデスティネーション･シングルのデータ幅に合わせなくてはなりません（DA[2:0]に対しLPDMA_CxTR1.DDW_LOG2[1:0]）。それ以外の場合、ユーザ設定エラーが報告され、転送は行われません。

18.8.15 LPDMAチャネルxのリンクリスト･アドレス･レジスタ（LPDMA_CxLLR）

アドレスオフセット：0x0CC + 0x80 * x（x = 0から3）

リセット値：0x0000 0000

このレジスタは、チャネルxのセキュア状態（LPDMA_SECCFGR.SECx）によって、セキュアまたは非セキュアとなり、チャネルxの特権状態（LPDMA_PRIVCFGR.PRIVx）によって、特権または非特権となります。

このレジスタは、メモリの次のLLIのデータ構造と、そのアドレス･ポインタの設定を行います。このレジスタがnullの場合、チャネル転送は完了です。

LPDMA_CxCR.EN = 0の場合、このレジスタに書き込まなくてはなりません。

LPDMA_CxCR.EN = 1のとき、このレジスタは読出し専用です。

チャネルが完了し（その後、ハードウェアがLPDMA_CxCR.ENをデアサートし）たら、このレジスタに書き込まなくてはなりません。チャネル転送は、ブロック、LLI、フル･リンクリストと、様々なレベルで完了したり、プログラムされたりする場合があります。

リンクリスト･モードのリンク転送時に、LPDMA_CxLLR.ULL = 1の場合、このレジスタはLPDMAによりメモリから自動的に更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UT1	UT2	UB1	USA	UDA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL
rw	rw	rw	rw	rw											rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LA[15:2]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット31 **UT1**：メモリからLPDMA_CxTR1を更新

このビットはリンク転送時にメモリからのLPDMA_CxTR1レジスタの更新を制御します。

0：LPDMA_CxTR1の更新はなし

1：LPDMA_CxTR1を更新

ビット30 **UT2**：メモリからLPDMA_CxTR2を更新

このビットはリンク転送時にメモリからのLPDMA_CxTR2レジスタの更新を制御します。

0：LPDMA_CxTR2の更新はなし

1：LPDMA_CxTR2を更新

ビット29 **UB1**：メモリからLPDMA_CxBR1を更新

このビットはリンク転送時にメモリからのLPDMA_CxBR1レジスタの更新を制御します。

0：メモリからのLPDMA_CxBR1更新は無しで、前にプログラムされている値に内部的に復元

1：LPDMA_CxBR1を更新

ビット28 **USA**：メモリからLPDMA_CxSARを更新

このビットはリンク転送時にメモリからのLPDMA_CxSARレジスタの更新を制御します。

0：LPDMA_CxSARの更新はなし

1：LPDMA_CxSAR更新

ビット27 **UDA**：メモリからLPDMA_CxDARレジスタを更新

このビットはリンク転送時にメモリからのLPDMA_CxDARレジスタの更新の制御に使用されます。

0：LPDMA_CxDARの更新はなし

1：LPDMA_CxDARを更新

ビット26:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **ULL**：メモリからLPDMA_CxLLRレジスタを更新

このビットはリンク転送時にメモリからのLPDMA_CxLLRレジスタの更新の制御に使用されます。

0：LPDMA_CxLLRの更新はなし

1：LPDMA_CxLLR更新

ビット15:2 **LA[15:2]**：次のリンクリスト･データ構造へのポインタ（16bitの下位アドレス）

UT1 = UT2 = UB1 = USA = UDA = ULL = 0でLA[15:20] = 0の場合、現在のLLIが最後です。チャネル転送が、リンクリストDMAレジスタ･ファイルの更新が一切なく完了します。

それ以外の場合、このフィールドは、、次のリンクリスト･データ構造を自動的にフェッチする、メモリ･アドレス･オフセットへのポインタで、データ転送が一旦完了すると、リンクリストDMA内部レジスタ･ファイル（LPDMA_CxTR1、LPDMA_CxTR2、LPDMA_CxBR1、LPDMA_CxSAR、LPDMA_CxDAR、LPDMA_CxLLR）は条件付きで更新されます。

注： ユーザはこのポインタを32bitに合わせてプログラムしなくてはなりません。下位2bitの書込みは無視されます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

18.8.16 LPDMAレジスタマップ

表153.LPDMAレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	LPDMA_SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC3	SEC2	SEC1	SEC0
	リセット値																													0	0	0	0
0x004	LPDMA_PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV3	PRIV2	PRIV1	PRIV0
	リセット値																													0	0	0	0
0x008	LPDMA_ RCFGLOCKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK3	LOCK2	LOCK1	LOCK0
	リセット値																													0	0	0	0
0x00C	LPDMA_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS3	MIS2	MIS1	MIS0
	リセット値																													0	0	0	0
0x010	LPDMA_SMISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIS3	MIS2	MIS1	MIS0
	リセット値																													0	0	0	0
0x014 - 0x04C	予約済み	Res.																															
0x050 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxLBAR	LBA[31:16]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
0x05C + 0x080 * x （x = 0 ~ 3）	LPDMA_CxFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																		0	0	0	0	0	0	0								
0x060 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOF	SUSPF	USEF	ULEF	DTEF	HTF	TCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDLEF
	リセット値																		0	0	0	0	0	0	0								1
0x064 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIO[1:0]		Res.	Res.	Res.	Res.	Res.	LSM	Res.	TOIE	SUSPIE	USEIE	ULEIE	DTEIE	HTIE	TCIE	Res.	Res.	Res.	Res.	Res.	SUSP	リセット	EN
	リセット値									0	0						0		0	0	0	0	0	0	0						0	0	0

表153.LPDMAレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x090 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxTR1	DSEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DINC	Res.	DDW_LOG2 [1:0]		SSEC	Res.	Res.	Res.	PAM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SINC	Res.	SDW_LOG2 [1:0]	
	リセット値	0												0		0	0	0				0								0		0	0
0x094 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxTR2	TCEM [1:0]		Res.	Res.	Res.	Res.	TRIGPOL [1:0]		Res.	Res.	Res.	TRIGSEL [4:0]				Res.	TRIGM [1:0]		Res.	Res.	Res.	BREQ	SWREQ	Res.	Res.	Res.	Res.	REQSEL [4:0]				
	リセット値	0	0					0	0				0	0	0	0		0	0				0	0					0	0	0	0	0
0x098 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxBR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BNDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x09C + 0x080 * x （x = 0 ~ 3）	LPDMA_CxSAR	SA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0 + 0x080 * x （x = 0 ~ 3）	LPDMA_CxDAR	DA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0CC + 0x080 * x （x = 0 ~ 3）	LPDMA_CxLLR	UT1	UT2	UB1	USA	UDA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ULL	LA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

レジスタ境界アドレスについては、セクション2.3を参照してください。

19 Chrom-Art Acceleratorコントローラ（DMA2D）

このセクションは、STM32U575/585/59x/5Axデバイスにのみ適用されます。

19.1 DMA2Dの概要

Chrom-ARTアクセラレータ（DMA2D）はイメージ操作専用の特殊なDMAです。以下の操作を実行できます。

- 転送先イメージの一部または全部を特定の色で塗りつぶす
- 転送元イメージの一部または全部を、転送先イメージの一部または全部にコピーする
- 転送元イメージの一部または全部を、ピクセルフォーマット変換を行って転送先イメージの一部または全部にコピーする
- 異なるピクセルフォーマットを持つ2つの転送元イメージの全部または一部をブレンドし、その結果を別のカラーフォーマットで転送先イメージの一部または全体にコピーする

すべての標準的なピクセルフォーマットがサポートされており、4bit/pixelから32bit/pixelのインデックスカラーモードまたはダイレクトカラーモードに対応しています。DMA2DはCLUT（color look-up tables：カラー･ルックアップ･テーブル）用に専用メモリを搭載しています。

19.2 DMA2Dの主な機能

DMA2Dの主な機能は次のとおりです。

- シングルAHBマスタ･バス･アーキテクチャ
- 8/16/32bit アクセスをサポートする AHB スレーブ･プログラミング･インタフェース（32bit の CLUTアクセスを除く）
- ユーザが任意に設定可能なワーク領域サイズ
- ピクセルまたはバイトで表現された転送元および転送先領域用の、ユーザーがプログラム可能なオフセット
- 全メモリ空間でユーザが任意に設定可能な転送元および転送先のアドレス
- 最大2つの転送元のブレンド操作
- アルファ値を調整可能（ソース値、固定値、または変調値）
- ユーザが任意に設定可能な転送元および転送先のカラーフォーマット
- インダイレクトカラーまたはダイレクトカラーで4bit/pixelから32bit/pixelまで、最大11のカラーフォーマットをサポート。
- インダイレクトカラーモードにおけるCLUT格納用に2つの内部メモリを搭載
- 自動CLUTローディングまたはCPUを使用したCLUTプログラミング
- ユーザが任意にプログラム可能なCLUTサイズ
- AHB帯域幅を制御する内部タイマ
- 6つの動作モード：レジスタからメモリへの転送、メモリ間転送、ピクセルフォーマット変換を伴うメモリ間転送、ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送、ピクセルフォーマット変換、ブレンディングおよび固定カラーフォアグラウンドを伴うメモリ間転送、ピクセルフォーマット変換、ブレンディングおよび固定カラーバックグラウンドを伴うメモリ間転送。
- 固定カラーでの矩形領域塗りつぶし
- 領域間のコピー

- 転送元イメージと転送先イメージ間のピクセルフォーマット変換を伴うコピー
- ピクセルフォーマットの異なる2つの転送元のコピー、およびブレンディング
- パラレルインタフェースを通してディスプレイのリフレッシュをサポートするための出力バッファバイトスワッピング
- DMA2D動作のアボートおよびサスペンド
- ユーザが任意に設定可能な転送先ラインにおけるウォーターマーク割込み
- バスエラーまたはアクセス競合時の割込み生成
- プロセス完了時の割込み生成

19.3 DMA2Dの機能詳細

19.3.1 DMA2Dブロック図

DMA2Dコントローラはダイレクトメモリ転送を行います。AHBマスタとして、AHBバスマトリックスを制御してAHBトランザクションを開始することができます。

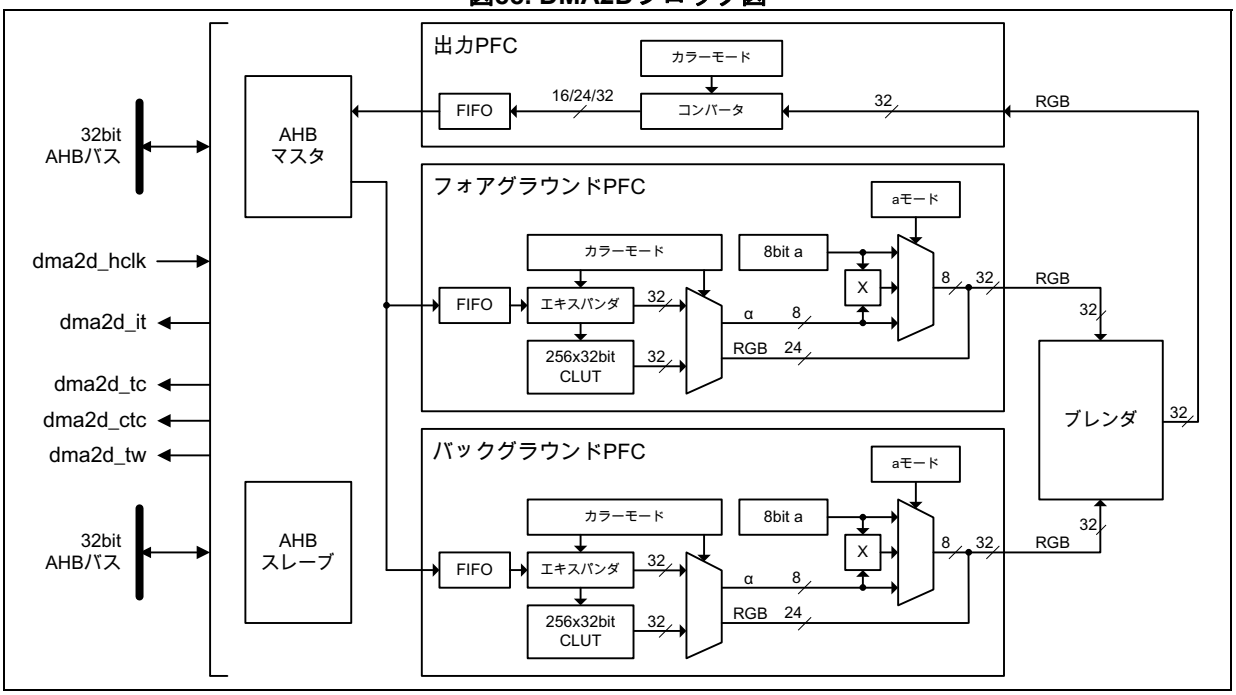
DMA2Dは、次のモードで動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- ピクセルフォーマット変換を伴うメモリ間転送
- ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送
- ピクセルフォーマット変換、ブレンディングおよび固定カラーフォアグラウンドを伴うメモリ間転送
- ピクセルフォーマット変換、ブレンディングおよび固定カラーバックグラウンドを伴うメモリ間転送

AHBスレーブポートは、DMA2Dコントローラをプログラムするために使用されます。

DMA2Dのブロック図を次の図に示します。

図88. DMA2Dブロック図



19.3.2 DMA2D内部信号

DMA2Dの内部信号を次の表に示します。

表154.DMA2D内部信号

名前	信号タイプ	説明
dma2d_hclk	入力	DMA2D AHBクロック
dma2d_it	出力	DMA2Dグローバル割込みリクエスト
dma2d_tc	出力	DMA2D転送完了トリガ
dma2d_ctc	出力	DMA2D CLUT転送完了トリガ
dma2d_tw	出力	DMA2D転送ウォーターマークトリガ

次の表に、DMA2Dのフラグの接続方法を示します。

表155.DMA2Dトリガ相互接続(1)

トリガ名	方向	トリガ転送元／転送先
dma2d_tc	出力	gpdma_trigsel[50]
dma2d_ctc	出力	gpdma_trigsel[51]
dma2d_tw	出力	gpdma_trigsel[52]

1. STM32U59x/5Ax製品でのみ使用できます。

19.3.3 DMA2D制御

DMA2Dコントローラは、DMA2D制御レジスタ（DMA2D_CR）を使用して設定します。

ユーザアプリケーションで次の操作を実行できます。

- 動作モードの選択
- DMA2D割込みの有効化／無効化
- 進行中のデータ転送の開始／サスペンド／アボート

19.3.4 DMA2DフォアグラウンドおよびバックグラウンドFIFO

DMA2Dフォアグラウンド（FG）FIFOおよびバックグラウンド（BG）FIFOはコピーなどの処理をする入力データをフェッチします。

これらのFIFOは、それぞれのピクセルフォーマットコンバータ(PFC)に定義されているカラーフォーマットに従ってピクセルデータをフェッチします。これらは次の制御レジスタにより設定可能です。

- DMA2Dフォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）
- DMA2Dフォアグラウンドオフセットレジスタ（DMA2D_FGOR）
- DMA2Dバックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）
- DMA2Dバックグラウンドオフセットレジスタ（DMA2D_BGOR）
- DMA2Dライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2Dがレジスタからメモリへの転送モードで動作する場合、どちらのFIFOもアクティブになりません。

DMA2Dがメモリ間転送モードで動作する（ピクセルフォーマット変換、ブレンディング処理のいずれも行わない）場合は、FG FIFOのみアクティブになりバッファとして機能します。

DMA2Dがピクセルフォーマット変換を伴うメモリ間転送モードで動作する（ブレンディング処理を行わない）場合は、BG FIFOはアクティブになりません。

19.3.5 DMA2Dフォアグラウンドおよびバックグラウンドのピクセルフォーマットコンバータ（PFC）

DMA2Dフォアグラウンドピクセルフォーマットコンバータ（PFC）およびバックグラウンドピクセルフォーマットコンバータはピクセルフォーマット変換を行って32bit/pixelのデータを生成します。PFCはアルファ値に調整を加えることもできます。

コンバータの一段目ではピクセルフォーマットを変換します。フォアグラウンドピクセルとバックグラウンドピクセルの元のピクセルフォーマットは、それぞれDMA2D_FGPFCCRおよびDMA2D_BGPFCCRのCM[3:0]ビットによって設定します。

サポートされている入力フォーマットを次の表に示します。

表156.入力でサポートされているカラーモード

CM[3:0]	ピクセルフォーマット
0000	ARGB8888
0001	RGB888
0010	RGB565
0011	ARGB1555
0100	ARGB4444

表156.入力でサポートされているカラーモード （続き）

CM[3:0]	ピクセルフォーマット
0101	L8
0110	AL44
0111	AL88
1000	L4
1001	A8
1010	A4

カラーフォーマットは次のとおりコード化されています。

- アルファ値フィールド：透明度
  0xFF値は不透明ピクセルに対応し0x00は透明ピクセルに対応します。
- Rフィールドは赤
- Gフィールドは緑
- Bフィールドは青
- Lフィールド：輝度
  このフィールドはRGBから3つの、ARGBから4つの成分を読み出すためのCLUTのインデックスです。

元のフォーマットがダイレクトカラーモード（ARGB/RGB）の場合、MSBをLSBへコピーしてチャネルあたり8bitへの拡張が行われます。これにより変換の線形性が完全に保証されます。

元のフォーマットがインダイレクトカラーモード（L/AL）の場合、CLUTが必要となり、各ピクセルフォーマットコンバータは、256エントリの32bit CLUTと連動します。

元のフォーマットにアルファ値が無い場合、アルファ値は自動的に0xFF（不透明）にセットされます。

特定のアルファモードA4およびA8では、カラー情報の格納、インデックスの利用のいずれも行われません。イメージ生成に使用されるカラーは固定となっており、フォアグラウンドピクセル用のカラーはDMA2D_FGCOLRレジスタで、バックグラウンドピクセル用のカラーはDMA2D_BGCOLRレジスタで定義されます。

システムメモリ内のフィールドの順序は次の表で定義しています。

表157.メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]
L8	L_3[7:0]	L_2[7:0]	L_1[7:0]	L_0[7:0]
AL44	A_3[3:0]L_3[3:0]	A_2[3:0]L_2[3:0]	A_1[3:0]L_1[3:0]	A_0[3:0]L_0[3:0]
AL88	A_1[7:0]	L_1[7:0]	A_0[7:0]	L_0[7:0]

表157.メモリ内のデータ順序 （続き）

L4	L_7[3:0]L_6[3:0]	L_5[3:0]L_4[3:0]	L_3[3:0]L_2[3:0]	L_1[3:0]L_0[3:0]
A8	A_3[7:0]	A_2[7:0]	A_1[7:0]	A_0[7:0]
A4	A_7[3:0]A_6[3:0]	A_5[3:0]A_4[3:0]	A_3[3:0]A_2[3:0]	A_1[3:0]A_0[3:0]

32bitに調整された24bit RGB888はARGB8888モードでサポートされます。

32bit値が生成されると、次の表に示すようにDMA2D_FGPFCCR/DMA2D_BGPFCCRレジスタのAM[1:0]に従ってアルファチャネルを調整することができます。

この時アルファチャネルは以下が可能です。

- そのままを維持（調整なし）
- DMA2D_FGPFCCR/DMA2D_BGPFCCRのALPHA[7:0]値で置換
- DMA2D_FGPFCCR/DMA2D_BGPFCCRのALPHA[7:0]値を255で除算した後、元のアルファ値に乗算した値で置換

表158.アルファモード設定

AM[1:0]	アルファモード
00	調整なし
01	DMA2D_xxPFCCRの値で置換
10	DMA2D_xxPFCCRを255で除算した値を元の値に乗算して置換
11	予約済みです。

注： **別のフォーマットをサポートするために、DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AI をセットして、入力されるアルファ値を反転できます。これは、DMA2D_FGPFCCR/DMA2D_BGPFCCRおよびCLUTに格納されるアルファ値にも適用されます。**

DMA2D_FGPFCCR/DMA2D_BGPFCCRレジスタのRBSをセットして、RフィールドとBフィールドを入れ替えることもできます。これは、CLUTおよびDMA2D_FGCOLR/DMA2D_BGCOLRレジスタで使用されるRGB順序にも適用されます。

19.3.6 DMA2DフォアグラウンドおよびバックグラウンドCLUTインタフェース

CLUTメモリアクセスおよびCLUTの自動ローディングはCLUTインタフェースによって管理されます。

次の3つのアクセスタイプが使用できます。

- ピクセルフォーマット変換操作中のPFCによるCLUTの読出し
- CPUがCLUTに対してデータの読出しまたは書込みを行っているときのAHBスレーブポートを経由したCLUTへのアクセス
- CLUTの自動ローディングが行われているときのAHBマスタポートを経由したCLUTへの書込み

CLUTメモリローディングは次の2つの方法で行うことができます。

- オートローディング

  CLUTのローディングは、次の順序で行う必要があります。

  a) DMA2D_FGCMAR（フォアグラウンドCLUT）またはDMA2D_BGCMAR（バックグラウンドCLUT）にCLUTアドレスを設定します。

  b) DMA2D_FGPFCCR（フォアグラウンドCLUT）またはDMA2D_BGPFCCR（バックグラウンドCLUT）のCS[7:0]にCLUTサイズを設定します。

  c) DMA2D_FGPFCCR（フォアグラウンドCLUT）またはDMA2D_BGPFCCR（バックグラウンドCLUT）のSTARTビットをセットして転送を開始します。この自動ロード処理中、CPUはCLUTにアクセスできません。競合が発生した場合、CLUTアクセスエラー割込みが発生します。ただし、DMA2D_CRでCAEIE = 1に設定されている必要があります。

- マニュアルローディング

  ローカルCLUTメモリがマップされているDMA2D AHBスレーブポートを使用して、アプリケーションがマニュアルでCLUTを設定する必要があります。フォアグラウンドCLUT（FGCLUT）はアドレスオフセット0x0400に配置され、バックグラウンドCLUT（BGCLUT）はアドレスオフセット0x0800に配置されます。

CLUTフォーマットは、24bitまたは32bitです。次の表に示すようにDMA2D_FGPFCCR（フォアグラウンドCLUT）またはDMA2D_BGPFCCR（バックグラウンドCLUT）のCCMビットを介してCLUTフォーマットを設定します。

表159.サポートされているCLUTカラーモード

CCM	CLUTカラーモード
0	32bit ARGB8888
1	24bit RGB888

システムメモリ内のCLUTデータの構成方法は次の表に規定しています。

表160.システムメモリ内のCLUTデータ順序

CLUTカラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]

19.3.7 DMA2Dブレンダ

DMA2Dブレンダは、転送元ピクセルをペアでブレンドし、出来上がりのピクセルを計算します。

ブレンドは次の式に従って行われます。

$$次の場合:\alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}}$$ ここでC = RまたはGまたはB

除算では、最も近い整数に切り捨てられます。

ブレンダは設定レジスタを必要としません。ブレンダの使用方法はDMA2D_CRのMODE[2:0]で定義されるDMA2D動作モードによって異なります。

19.3.8 DMA2D出力PFC

出力PFCは、32bitからDMA2D_OPFCCRのCM[2:0]で定義された出力フォーマットへのピクセルフォーマット変換を行います。

サポートされている出力フォーマットを次の表に示します。

表161.出力でサポートされているカラーモード

CM[2:0]	ピクセルフォーマット
000	ARGB8888
001	RGB888
010	RGB565
011	ARGB1555
100	ARGB4444

注： **別のフォーマットをサポートするために、DMA2D_OPFCCRのAIビットをセットして、計算されたアルファ値を反転します。これは、DMA2D_OCOLRで使用されるアルファ値にも適用されます。**

DMA2D_OPFCCRのRBSをセットして、RフィールドとBフィールドを入れ替えることもできます。これは、DMA2D_OCOLRで使用されるRGB順序にも適用されます。

19.3.9 DMA2D出力FIFO

出力FIFOは、出力PFCに定義されているカラーフォーマットに従ってピクセルをプログラムします。

転送先領域は、次の制御レジスタによって定義します。

- DMA2D出力メモリアドレスレジスタ（DMA2D_OMAR）
- DMA2D出力オフセットレジスタ（DMA2D_OOR）
- DMA2Dライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2Dがレジスタからメモリモードで動作する場合、設定された出力矩形領域が、32bit、24bitまたは16bitの固定値が格納されたDMA2D_OCOLRで指定された色で塗りつぶされます。このフォーマットは、DMA2D_OPFCCRのCM[2:0]によって選択されます。

データは、次の表で定義されている順番でメモリに格納されます。

表162.メモリ内のデータ順序

ピクセルフォーマット	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]

32bitに調整されたRGB888はARGB8888モードでサポートされます。

19.3.10 DMA2D出力FIFOバイトの順序変更

出力FIFOバイトは、DMA2Dから直接パラレルインタフェース（F（S）MC）を介してディスプレイフレームバッファの更新をサポートするように並べ替えられます。

バイトの順序変更は以下のように行われます。

- 赤と青のコンポーネントをスワップするRBSビット
- 出力FIFOで2 x 2でバイトスワップするSBビット

バイトスワッピングがアクティブになっている場合（DMA2D_OPFCCRのSB =1）、1行あたりのピクセル数（DMA2D_NLRのPLフィールド）は偶数、出力メモリアドレス（DMA2D_OMARのMAフィールド）は偶数である必要があります。また、バイト単位で計算された出力ラインオフセット（DMA2D_CRのLOMフィールドとDMA2D_OORのLOフィールドの値による結果）も偶数である必要があります。そうでない場合、設定エラーが検出されます。

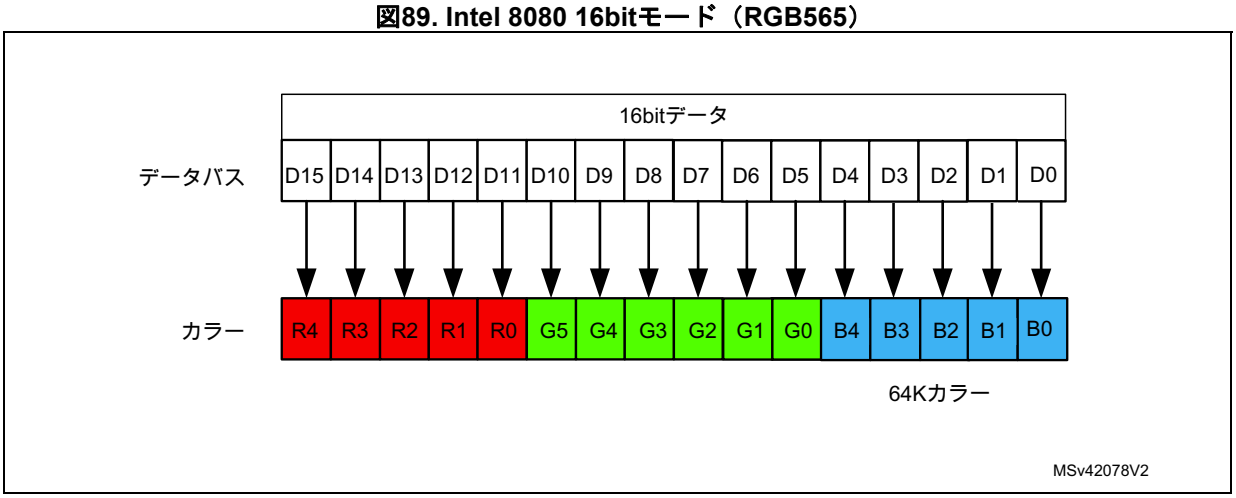
表163.メモリ内の標準データ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]

16bitモード（RGB565）

このモードは、DMA2Dによるバイトの並べ替えなしでサポートされます。

図89. Intel 8080 16bitモード（RGB565）



18/24bitモード（RGB888）

このモードは、データの並び替えが必要です。

1. 赤と青をスワップしなければなりません（RBSビットのセット）
2. ハーフワードのMSBおよびLSBバイトはスワップしなければなりません（SBビットのセット）。

図90. Intel 8080 18/24bitモード（RGB888）

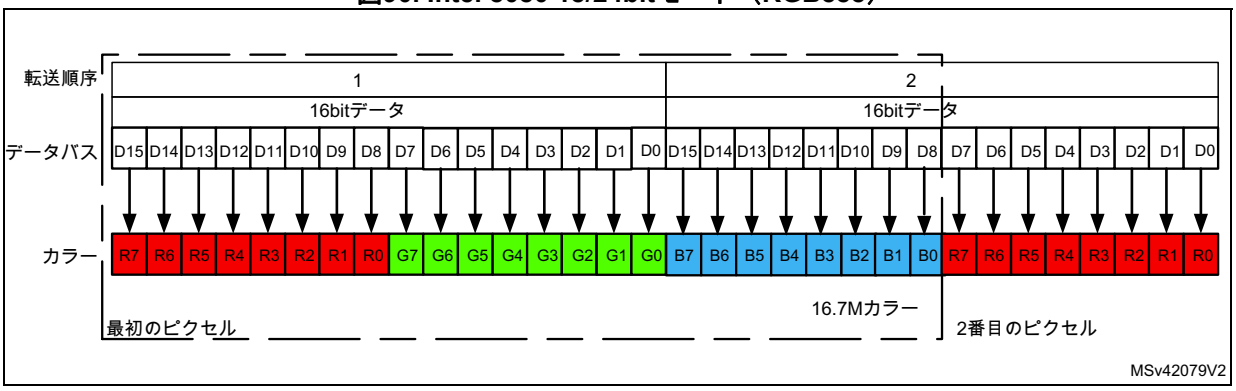


表164.出力FIFOバイトの順序変更ステップ

ステップ	@ + 3	@ + 2	@ + 1	@ + 0
元のデータ順序	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RBSビットのセット				
赤と青のスワップ後のデータ順序（RBSセット）	R_1[7:0]	B_0[7:0]	G_0[7:0]	R_0[7:0]
	G_2[7:0]	R_2[7:0]	B_1[7:0]	G_1[7:0]
	B_3[7:0]	G_3[7:0]	R_3[7:0]	B_2[7:0]
SBビットのセット				
バイトスワップ後のデータ順序（SBセット）	B_0[7:0]	R_1[7:0]	R_0[7:0]	G_0[7:0]
	R_2[7:0]	G_2[7:0]	G_1[7:0]	B_1[7:0]
	G_3[7:0]	B_3[7:0]	B_2[7:0]	R_3[7:0]

19.3.11 DMA2D AHBマスタポートタイマ

AHBマスタポートには8bitのタイマが組込まれており、クロスバーの帯域幅を任意に制限することができます。

このタイマはAHBクロックによって駆動され、2つの連続アクセス間のデッドタイムをカウントします。これによって帯域幅の使用を制限することができます。

タイマの有効化とデッドタイム値はAHBマスタポートタイマ設定レジスタ（DMA2D_AMPTCR）によって設定します。

19.3.12 DMA2Dトランザクション

DMA2Dトランザクションは、一定数のデータ転送のシーケンスで構成されています。データの数と幅はソフトウェアによりプログラムすることができます。

各DMA2Dデータ転送は、最大4つのステップで構成されています。

1. DMA2D_FGMAR で示されるメモリ位置からのデータローディングおよび DMA2D_FGCR で定義されるピクセルフォーマット変換。
2. DMA2D_BGMAR で示されるメモリ位置からのデータローディングおよび DMA2D_BGCR で定義されるピクセルフォーマット変換。
3. アルファ値に関する PFC 操作の結果として生じるアルファチャネルに従ったすべての読出しピクセルのブレンディング。
4. DMA2D_OCR に従ってできたピクセルのピクセルフォーマット変換および DMA2D_OMAR によってアドレスされたメモリ位置へのデータのプログラミング。

19.3.13 DMA2D設定

データ転送元およびデータ転送先は両方ともアドレス0x0000 0000と0xFFFF FFFFの間の全4Gバイトメモリ領域内のペリフェラルおよびメモリを対象にすることができます。

DMA2Dは、DMA2D_CRのMODE[2:0]を使用して選択された次のいずれのモードでも動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- PFCを伴うメモリ間転送
- PFCとブレンディングを伴うメモリ間転送
- PFC、ブレンディングおよび固定FGカラーを伴うメモリ間転送
- PFC、ブレンディングおよび固定BGカラーを伴うメモリ間転送

レジスタからメモリへの転送

“レジスタからメモリ”モードは、ユーザ定義領域を事前定義された色で塗りつぶすために使用します。

カラーフォーマットはDMA2D_OPFCCRでセットします。

DMA2Dはどのソースからもデータのフェッチを行いません。DMA2D_OCOLRで定義された色を、DMA2D_OMARで示されるアドレスの、DMA2D_NLRおよびDMA2D_OORで定義される領域に書き込むだけです。

メモリ間転送

メモリ間モードでは、DMA2Dはグラフィカルなデータの変換は何も行いません。フォアグラウンド入力FIFOはバッファの役目を果たし、データは、DMA2D_FGMARで定義される転送元のメモリ位置からDMA2D_OMARが示す転送先のメモリ位置へ転送されます。

DMA2D_FGPFCCRのCM[3:0]でプログラムされているカラーモードによって、入力および出力の両方のピクセルあたりのビット数を定義します。

転送する領域のサイズは、転送元に対してはDMA2D_NLRおよびDMA2D_FGORで定義され、転送先に対してはDMA2D_NLRおよびDMA2D_OORで定義されます。

PFCを伴うメモリ間転送

このモードではDMA2Dは転送元データのピクセルフォーマット変換を行いそれを転送先メモリ位置に格納します。

転送する領域のサイズは、転送元に対してはDMA2D_NLRおよびDMA2D_FGORで定義され、転送先に対してはDMA2D_NLRおよびDMA2D_OORで定義されます。

データはDMA2D_FGMARで定義された場所からフェッチされ、フォアグラウンドPFCによって処理されます。元のピクセルフォーマットはDMA2D_FGPFCCRを使用して設定されます。

元のピクセルフォーマットがダイレクトカラーモードである場合、カラーチャネルはすべて8bitに拡張されます。

ピクセルフォーマットがインダイレクトカラーモードである場合、関連するCLUTをCLUTメモリにロードする必要があります。

CLUTのローディングは以下の流れに従って自動的に行うことができます。

1. CLUTアドレスをDMA2D_FGCMARにセットします。
2. DMA2D_FGPFCCRのCS[7:0]ビットによりCLUTサイズをセットします。
3. DMA2D_FGPFCCRのCCMによりCLUTフォーマット（24bitまたは32bit）をセットします。
4. DMA2D_FGPFCCRにSTARTをセットして、CLUTローディングを開始します。

CLUTローディングが完了すると、DMA2D_IFRのCTCIFフラグが立てられ、DMA2D_CRでCTCIE = 1であれば割込みが生成されます。自動CLUTローディング処理は標準的なDMA2D転送と並行して動作することはできません。

CLUTはCPUから書き込むこともできれば、APBポート経由で他のマスタによって書き込むこともできます。DMA2D転送が進行中でかつCLUTを使用しているとき（インダイレクトカラーフォーマット）、CLUTにはアクセスできません。

カラー変換処理と並行して、DMA2D_FGPFCCRにプログラムされている値によっては、アルファ値を追加または変更することができます。元のイメージにアルファチャネルが無い場合、デフォルトのアルファ値 0xFF が自動的に追加されて完全に不透明なピクセルになります。アルファ値はDMA2D_FGPFCCRのAM[1:0]に従って調整されます。

- 変更しない設定ができます。
- DMA2D_FGPFCCRのALPHA[7:0]で定義された値で置き換えることができます。
- 元の値にALPHA[7:0]を掛けて255で割った値で置き換えることができます。

結果の32bitデータはOUT PFCによって、DMA2D_OPFCCRのCM[2:0]で指定されたフォーマットにエンコードされます。CLUT生成処理がサポートされていないため、出力ピクセルフォーマットをインダイレクトモードにすることはできません。

処理されたデータはDMA2D_OMARが指す転送先メモリ位置に書き込まれます。

PFCとブレンディングを伴うメモリ間転送

このモードでは、DMA2D_FGMARおよびDMA2D_BGMARで定義されるメモリ位置からフォアグラウンドFIFOおよびバックグラウンドFIFOに2つの転送元データがフェッチされます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換された後、それらは次の式に従ってブレンドされます。

$$\text{次の場合}: \alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}} \quad \text{ここで C = R または G または B}$$

除算では、小数点以下は切り捨てられます。

結果の32bitピクセル値は出力PFCにより指定出力フォーマットに従ってエンコードされ、データはDMA2D_OMARが指す転送先のメモリ位置に書き込まれます。

PFC、ブレンディングおよび固定カラー FGを伴うメモリ間転送

このモードでは、DMA2D_BGMARで定義されるメモリ位置からバックグラウンドFIFOに1つの転送元データのみがフェッチされます。

フォアグラウンドカラーの値はDMA2D_FGCOLRによって指定され、アルファ値は0xFF（不透明）にセットされます。

アルファ値は、DMA2D_FGPFCRのAM[1:0]およびALPHA[7:0]に従って置き換え、または調整することができます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換されると、すべてのピクセルは一緒にブレンドされ、結果の32bitピクセル値は出力PFCにより、指定出力フォーマットに従ってエンコードされます。データはDMA2D_OMARが指す転送先メモリ位置に書き込まれます。

PFC、ブレンディングおよび固定カラー BGを伴うメモリ間転送

このモードでは、DMA2D_FGMARで定義されるメモリ位置からフォアグラウンドFIFOに1つの転送元データのみがフェッチされます。

バックグラウンドカラーの値はDMA2D_BGCOLRによって指定され、アルファ値は0xFF（不透明）に設定されます。

アルファ値は、DMA2D_BGPFCRのAM[1:0]およびALPHA[7:0]に従って置き換え、または調整することができます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換されると、すべてのピクセルは一緒にブレンドされ、結果の32bitピクセル値は出力PFCにより、指定出力フォーマットに従ってエンコードされ、データはDMA2D_OMARが指す転送先のメモリ位置に書き込まれます。

設定エラー検出

DMA2Dは転送を行う前に設定が正しいことを確認します。新規の転送／自動ローディングが開始されるとき、誤った設定が検出されると、ハードウェアによって設定エラー割込みフラグがセットされます。DMA2D_CRでCEIE = 1である場合、割込みが生成されます。

検出することができる誤った設定は次のとおりです。

- フォアグラウンドCLUT自動ローディング：DMA2D_FGCMARのMAビットがDMA2D_FGPFCCRのCCMと整合が取れていません。
- バックグラウンドCLUT自動ローディング：DMA2D_BGCMARのMAビットがDMA2D_BGPFCCRのCCMと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGMARのMAビットがDMA2D_FGPFCCRのCMと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGPFCCRのCMビットが無効です。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_NLRのPLビットが奇数で、DMA2D_FGPFCCRのCMがA4またはL4です。

- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGORのLOビットが奇数で、DMA2D_FGPFCCRのCMがA4またはL4で、DMA2D_CRのLOMビットがピクセルモードです。
- メモリ転送（"ブレンディング"モードのみ、また"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_BGMARのMAビットがDMA2D_BGPFCCRのCMと整合が取れていません。
- メモリ転送（"ブレンディングモード"、および、"固定カラーFGを伴うブレンディング"モードのみ）：DMA2D_BGPFCCRのCMビットが無効です。
- メモリ転送：（"ブレンディング"モード、および、"固定カラーFGを伴うブレンディング"モードのみ）：DMA2D_NLRのPLビットが奇数で、DMA2D_BGPFCCRのCMがA4またはL4です。
- メモリ転送（"ブレンディング"モード、および、"固定カラーFGを伴うブレンディング"モードのみ）：DMA2D_BGORのLOビットが奇数で、DMA2D_BGPFCCRのCMがA4またはL4で、DMA2D_CRのLOMビットがピクセルモードです。
- メモリ転送（"メモリ間"モード、および、"ブレンディングおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_OMARのMAビットがDMA2D_OPFCCRのCMと整合が取れていません。
- メモリ転送（"メモリ間"モードを除く）：DMA2D_OPFCCRのCMビットが無効です。
- バイトスワッピングを伴うメモリ転送：DMA2D_NLRのPLビットが奇数またはDMA2D_OMARのMAビットが奇数、または（DMA2D_CRのLOMとDMA2D_OORのLOビットから生じる）LOのバイトが奇数で、DMA2D_OPFCCRでSB = 1です。
- メモリ転送：NL = 0x0（DMA2D_NLR）
- メモリ転送：PL = 0x0（DMA2D_NLR）
- メモリ転送：DMA2D_CRのMODEビットが無効です。

19.3.14 DMA2D転送制御（開始、サスペンド、アボート、完了）

DMA2Dが設定されると、DMA2D_CRのSTARTをセットすることにより転送を開始できます。転送が完了すると、START は自動的にリセットされ、DMA2D_ISR の TCIF フラグが立てられます。DMA2D_CRでTCIEがセットされている場合は、割込みを生成することができます。

DMA2D_CRのSUSPをセットすることにより、ユーザアプリケーションからいつでもDMA2Dをサスペンドすることができます。その後、DMA2D_CRのABORTをセットすることにより、トランザクションをアボートすることができます。また、DMA2D_CRのSUSPをリセットすることによりリスタートすることができます。

DMA2D_CRのABORTをセットすることにより、ユーザアプリケーションからいつでも進行中のトランザクションをアボートすることができます。この場合、TCIFフラグは立ちません。

DMA2D_CRのABORTまたはSUSPを使用することによって、自動CLUT転送もアボートしたりサスペンドしたりすることができます。

19.3.15 ウォーターマーク

ウォーターマークをプログラムして所定のラインの最後のピクセルが転送先のメモリ領域に書き込まれたときに割込みを生成することができます。

ライン番号は、DMA2D_LWRのLW[15:0]で定義します。

ラインの最後のピクセルが転送されると、DMA2D_CRのTWIEがセットされていれば、DMA2D_ISRのTWIFフラグが立ち、割込みが生成されます。

19.3.16 エラー管理

次の2種類のエラーをトリガすることができます。

- DMA2D_ISRのTEIFによって通知されるAHBマスタポートエラー
- CLUTアクセス（CLUTローディング中またはDMA2D転送中にCPUがCLUTにアクセスしようとする）によって引き起こされる競合。DMA2D_ISRのCAEIFによって通知されます。

両方のフラグはそれぞれDMA2D_CRの割込みイネーブルフラグに関連付けられており、必要に応じて割込みを生成します（TEIEとCAEIE）。

19.3.17 AHBデッドタイム

AHB帯域幅の使用を制限するために、2つの連続したAHBアクセス間のデッドタイムをプログラムすることができます。

この機能は、DMA2D_AMTCRのENをセットすることによって有効にできます。

デッドタイム値は、DMA2D_AMTCRのDT[7:0]に格納されます。この値は、AHBバス上の2つの連続するトランザクション間の最小保証サイクル数を表します。

DMA2D実行中のデッドタイム値の更新は、次のAHB転送に反映されます。

19.4 DMA2D割込み

割込みは次のイベントによって生成できます。

- 設定エラー
- CLUT転送完了
- CLUTアクセスエラー
- 転送ウォーターマークに到達
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表165.DMA2D割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット
設定エラー	CEIF	CEIE
CLUT転送完了	CTCIF	CTCIE
CLUTアクセスエラー	CAEIF	CAEIE
転送ウォーターマーク	TWF	TWIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

19.5 DMA2Dレジスタ

19.5.1 DMA2D制御レジスタ（DMA2D_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
		rw	rw	rw	rw	rw	rw		rw				rs	rw	rs

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **MODE[2:0]**：DMA2Dモード

これらのビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

000：メモリ間転送（FGフェッチのみ）

001：PFCを伴うメモリ間転送（FGP FCアクティブでFGフェッチのみ）

010：ブレンディングを伴うメモリ間転送（PFCおよびブレンディングを伴うFGフェッチおよびBGフェッチ）

011：レジスタからメモリ転送（FGおよびBGなし、出力ステージのみアクティブ）

100：ブレンディングおよび固定カラー FGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）

101：ブレンディングおよび固定カラー BGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）

その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CEIE**：設定エラー割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CE割込みは無効です。

1：CE割込みは有効です。

ビット12 **CTCIE**：CLUT転送完了割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CTC割込みは無効です。

1：CTC割込みは有効です。

ビット11 **CAEIE**：CLUTアクセスエラー割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CAE割込みは無効です。

1：CAE割込みは有効です。

ビット10 **TWIE**：転送ウォーターマーク割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TW割込みは無効です。

1：TW割込みは有効です。

ビット9 **TCIE**：転送完了割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC割込みは無効です。

1：TC割込みは有効です。

ビット8 **TEIE**：転送エラー割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE割込みは無効です。

1：TE割込みは有効です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LOM**：ラインオフセットモード

このビットは、フォアグラウンド、バックグラウンドおよび出力のラインオフセットの表現方法（ピクセルまたはバイト）を設定します。

このビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

0：ラインオフセットはピクセル単位で表現されます。

1：ラインオフセットはバイト単位で表現されます。

ビット5:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ABORT**：強制終了

このビットを使用して現在の転送を中止できます。このビットはソフトウェアによってセットされ、STARTビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送アボートはリクエストされていません。

1：転送アボートがリクエストされています。

ビット1 **SUSP**：一時停止

このビットを使用して現在の転送をサスペンドできます。このビットは、ソフトウェアによってセット／リセットされます。このビットはSTARTットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送はサスペンドしません。

1：転送はサスペンドします。

ビット0 **START**：開始

このビットを使用し、各種設定レジスタにロードしたパラメータに従ってDMA2Dを開始することができます。このビットは以下のイベントにより自動的にリセットされます。

– 転送終了時

– ユーザアプリケーションによりデータ転送がアボートされたとき（DMA2D_CRのABORTをセットして）

– データ転送エラーが発生したとき

– 設定エラーまたはすでに別の転送動作を行っているためデータ転送が開始されないとき（自動CLUTローディング）

19.5.2 DMA2D割込みステータスレジスタ（DMA2D_ISR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
										r	r	r	r	r	r

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CEIF**：設定エラー割込みフラグ

このビットはDMA2D_CR、DMA2DFGPFCCR、またはDMA2D_BGPFCCRのSTARTビットがセットされ、誤った設定がプログラムされているときにセットされます。

ビット4 **CTCIF**：CLUT転送完了割込みフラグ

このビットは、システムメモリ領域から内部DMA2Dメモリへの CLUTコピーが完了したときセットされます。

ビット3 **CAEIF**：CLUTアクセスエラー割込みフラグ

このビットは、CLUTがシステムメモリから内部DMA2Dへ自動的にコピーされている間にCPUがCLUTにアクセスした場合にセットされます。

ビット2 **TWIF**：転送ウォーターマーク割込みフラグ

このビットはウォーターマークを付けたラインの最後のピクセルが転送されたときセットされます。

ビット1 **TCIF**：転送完了割込みフラグ

このビットはDMA2D転送動作が完了したときセットされます（データ転送のみ）。

ビット0 **TEIF**：転送エラー割込みフラグ

このビットはDMA転送中（データ転送または自動CLUTローディング）にエラーが発生したときセットされます。

19.5.3 DMA2D割込みフラグクリアレジスタ（DMA2D_IFCR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CCEIF**：設定エラー割込みフラグのクリア

このビットを1にプログラムすると、DMA2D_ISRのCEIFがクリアされます。

ビット4 **CCTCIF**：CLUT転送完了割込みフラグのクリア
このビットを1にプログラムすると、DMA2D_ISRのCTCIFがクリアされます。

ビット3 **CAECIF**：CLUTアクセスエラー割込みフラグのクリア
このビットを1にプログラムすると、DMA2D_ISRのCAEIFがクリアされます。

ビット2 **CTWIF**：転送ウォーターマーク割込みフラグのクリア
このビットを1にプログラムすると、DMA2D_ISRのTWIFがクリアされます。

ビット1 **CTCIF**：転送完了割込みフラグのクリア
このビットを1にプログラムすると、DMA2D_ISRのTCIFがクリアされます。

ビット0 **CTEIF**：転送エラー割込みフラグのクリア
このビットを1にプログラムすると、DMA2D_ISRのTEIFがクリアされます。

19.5.4 DMA2Dフォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス
このフィールドには、フォアグラウンドイメージに使用されるデータのアドレスが格納されます。このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこのレジスタは読出し専用になります。
アドレス割当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32bit／ピクセルフォーマットは32bitに調整する必要があり、16bit／ピクセルフォーマットは16bitに調整する必要があり、4bit／ピクセルフォーマットは8bitに調整する必要があります。

19.5.5 DMA2Dフォアグラウンドオフセットレジスタ（DMA2D_FGOR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

ピクセル（LOMビットがリセットされた場合）およびバイト（LOMビットがセットされた場合）で表現されるフォアグラウンドイメージに使用するラインオフセットです。

ピクセルで表現される場合、LO[13:0]のみ考慮され、LO[15:14]は無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

イメージフォーマットが4bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

19.5.6 DMA2Dバックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

このフィールドには、バックグラウンドイメージに使用されるデータのアドレスが格納されます。このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこのレジスタは読出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32bit／ピクセルフォーマットは32bitに調整する必要があり、16bit／ピクセルフォーマットは16bitに調整する必要があり、4bit／ピクセルフォーマットは8bitに調整する必要があります。

19.5.7 DMA2Dバックグラウンドオフセットレジスタ（DMA2D_BGOR）

アドレスオフセット：0x018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

このフィールドには、ピクセル単位（LOMビットがリセットされた場合）およびバイト単位（LOMビットがセットされた場合）で表現されるバックグラウンドイメージに使用するラインオフセットが格納されます。

ピクセルで表現される場合、LO[13:0]のみ考慮され、LO[15:14]は無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

イメージフォーマットが4bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

19.5.8 DMA2Dフォアグラウンド PFC制御レジスタ（DMA2D_FGPFCCR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rs	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、またはAM[1:0]ビットによって選択されたアルファモードに従って元のアルファ値を乗算することができます。

これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するためにRとBをスワップできます。転送が開始されると、このビットは読出し専用になります。

0：レギュラモード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **AM[1:0]**：アルファモード

これらのビットによって、フォアグラウンドイメージに使用するアルファチャネル値を選択します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

00：フォアグラウンドイメージアルファチャネル値は修正されません。

01：元のフォアグラウンドイメージアルファチャネル値をALPHA[7:0]によって置き換えます。

10：元のフォアグラウンドイメージアルファチャネル値を、ALPHA[7:0]に元のアルファチャネル値を掛けた値で置き換えます。

その他：予約済み

ビット15:8 **CS[7:0]**：CLUTサイズ

これらのビットによって、フォアグラウンドイメージに使用するCLUTのサイズを定義します。CLUT転送が開始されるとこのフィールドは読出し専用になります。

CLUTエントリの数はCS[7:0] + 1に等しい値となります。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **START**：開始

このビットをセットしてCLUTの自動ロードを開始できます。リセットは次のときに自動的に行われます。

– 転送終了時

– DMA2D_CRのABORTをセットすることにより、ユーザアプリケーションにより転送がアボートされたとき

– 転送エラーが発生したとき

– 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンドCLUT転送）を行っているため転送が開始されないとき

ビット4 **CCM**：CLUTカラーモード

このビットによって、CLUTのカラーフォーマットを定義します。これは転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのビットは読出し専用になります。

0：ARGB8888

1：RGB888

その他：予約済み

ビット3:0 **CM[3:0]**：カラーモード

これらのビットによって、フォアグラウンドイメージのカラーフォーマットを定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

その他：予約済み

19.5.9 DMA2Dフォアグラウンドカラーレジスタ（DMA2D_FGCOLR）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **RED[7:0]**：赤の値

これらのビットによって、フォアグラウンドイメージのA4またはA8モードに対する赤の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラーFGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラーFGにも使用できます。

ビット15:8 **GREEN[7:0]**：緑の値

これらのビットによって、フォアグラウンドイメージのA4またはA8モードに対する緑の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラーFGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラーFGにも使用できます。

ビット7:0 **BLUE[7:0]**：青の値

これらのビットによって、フォアグラウンドイメージのA4またはA8モードに対する青の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラーFGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラーFGにも使用できます。

19.5.10 DMA2Dバックグラウンド PFC制御レジスタ（DMA2D_BGPFCCR）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rs	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、またはAM[1:0]によって選択されたアルファモードに従って元のアルファ値を乗算することができます。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するためにRとBをスワップできます。転送が開始されると、このビットは読出し専用になります。

0：レギュラモード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **AM[1:0]**：アルファモード

これらのビットによって、バックグラウンドイメージに使用するアルファチャネル値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

00：フォアグラウンドイメージアルファチャネル値は修正されません。

01：元のバックグラウンドイメージアルファチャネル値をALPHA[7:0]によって置き換えます。

10:元のバックグラウンドイメージアルファチャネル値をALPHA[7:0]に元のアルファチャネル値を掛けた値で置き換えます。

その他：予約済み

ビット15:8 **CS[7:0]**：CLUTサイズ

これらのビットによって、BGに使用するCLUTのサイズを定義します。CLUT転送が開始されるとこのフィールドは読出し専用になります。

CLUTエントリの数はCS[7:0] + 1に等しい値となります。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **START**：開始

このビットをセットしてCLUTの自動ロードを開始します。このビットは以下のときに自動的にリセットされます。

– 転送終了時

– DMA2D_CRのABORTをセットすることにより、ユーザアプリケーションにより転送がアボートされたとき

– 転送エラーが発生したとき

– 設定エラーまたはすでに別の転送動作（データ転送または自動フォアグラウンドCLUT転送）を行っているため転送が開始されないとき。

ビット4 **CCM**：CLUTカラーモード

これらのビットによって、CLUTのカラーフォーマットを定義します。このレジスタは、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのビットは読出し専用になります。

0：ARGB8888

1：RGB888

その他：予約済み

ビット3:0 **CM[3:0]**：カラーモード

これらのビットによって、バックグラウンドイメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

その他：予約済み

19.5.11 DMA2Dバックグラウンドカラーレジスタ（DMA2D_BGCOLR）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **RED[7:0]**：赤の値

これらのビットによって、バックグラウンドのA4またはA8モードに対する赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラー FGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラー FGにも使用されます。

ビット15:8 **GREEN[7:0]**：緑の値

これらのビットによって、バックグラウンドイメージのA4またはA8モードに対する緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラー FGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラー FGにも使用されます。

ビット7:0 **BLUE[7:0]**：青の値

これらのビットによって、バックグラウンドイメージのA4またはA8モードに対する青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

これらのビットは、ブレンディングおよび固定カラー FGを伴うメモリ間転送（FG PFCおよびBG PFCアクティブでBGフェッチのみ）モードでの固定カラー FGにも使用されます。

19.5.12 DMA2Dフォアグラウンド CLUTメモリアドレスレジスタ（DMA2D_FGCMAR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

このフィールドには、フォアグラウンドイメージ専用のCLUTアドレスに使用されるデータのアドレスが格納されます。このレジスタには、転送が行われていないときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

フォアグラウンドCLUTフォーマットが32bitの場合、アドレスは32bit境界に整列する必要があります。

19.5.13 DMA2Dバックグラウンド CLUTメモリアドレスレジスタ（DMA2D_BGCMAR）

アドレスオフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

このフィールドには、バックグラウンドイメージ専用のCLUTアドレスに使用されるデータのアドレスが格納されます。このレジスタには、転送が行われていないときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

バックグラウンドCLUTフォーマットが32bitの場合、アドレスは32bitに調整する必要があります。

19.5.14 DMA2D出力PFC制御レジスタ（DMA2D_OPFCCR）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
							rw						rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するためにRとBをスワップできます。転送が開始されると、このビットは読出し専用になります。

0：レギュラモード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット19:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SB**：バイトのスワップ

このビットがセットされている場合、出力FIFOのバイトは2 × 2でスワップされます。ラインごとのピクセル数（PL）が偶数、出力メモリアドレス（OMAR）が偶数である必要があります。

このレジスタは、転送が無効のときにのみ書込みできます。転送が開始されると、このレジスタは読出し専用になります。

0：バイトは出力FIFOでレギュラ順序になります。

1：バイトは出力FIFOで2 x 2でスワップされます。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **CM[2:0]**：カラーモード

これらのビットによって、出力イメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

その他：予約済み

19.5.15 DMA2D出力カラーレジスタ（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB8888またはRGB888カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：ARGB8888モードの出力カラーのアルファチャネル値（別途予約済み）

ビット23:16 **RED[7:0]**：ARGB8888またはRGB888モードでの出力イメージの赤の値

ビット15:8 **GREEN[7:0]**：ARGB8888またはRGB888での出力イメージの緑の値

ビット7:0 **BLUE[7:0]**：ARGB8888またはRGB888での出力イメージの青の値

19.5.16 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

RGB565カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RED[4:0]					GREEN[5:0]						BLUE[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:11 **RED[4:0]**：RGB565モードでの出力イメージの赤の値

ビット10:5 **GREEN[5:0]**：RGB565モードでの出力イメージの緑の値

ビット4:0 **BLUE[4:0]**：RGB565モードでの出力イメージの青の値

19.5.17 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB1555カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	RED[4:0]					GREEN[4:0]					BLUE[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **A**：ARGB1555モードでの出力カラーのアルファチャネル値

ビット14:10 **RED[4:0]**：ARGB1555モードでの出力イメージの赤の値

ビット9:5 **GREEN[4:0]**：ARGB1555モードでの出力イメージの緑の値

ビット4:0 **BLUE[4:0]**：ARGB1555モードでの出力イメージの青の値

19.5.18 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB4444カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **ALPHA[3:0]**：ARGB4444での出力カラー値のアルファチャネル

ビット11:8 **RED[3:0]**：ARGB4444モードでの出力イメージの赤の値

ビット7:4 **GREEN[3:0]**：ARGB4444モードでの出力イメージの緑の値

ビット3:0 **BLUE[3:0]**：ARGB4444モードでの出力イメージの青の値

19.5.19 DMA2D出力メモリアドレスレジスタ（DMA2D_OMAR）

アドレスオフセット：0x003C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

このフィールドには、出力FIFOに使用されるデータのアドレスが格納されます。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。例えば、32bit／ピクセルフォーマットは32bitに調整する必要があり、4bit／ピクセルフォーマットは8bitに調整する必要があります。

19.5.20 DMA2D出力オフセットレジスタ（DMA2D_OOR）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

このフィールドには、ピクセル単位（LOMビットがリセットされた場合）およびバイト単位（LOMビットがセットされた場合）で表現される出力に使用するラインオフセットが格納されます。

ピクセルで表現される場合、LO[13:0]のみ考慮され、LO[15:14]は無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

19.5.21 DMA2Dライン数レジスタ（DMA2D_NLR）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PL[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:16 **PL[13:0]**：ライン当たりのピクセル

転送する領域のラインあたりのピクセル数です。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

入力イメージフォーマットのいずれかがピクセルあたり4bitの場合、ラインあたりのピクセル数は偶数である必要があります。

ビット15:0 **NL[15:0]**：ライン数

転送する領域のライン数です。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

19.5.22 DMA2Dラインウォーターマークレジスタ（DMA2D_LWR）

アドレスオフセット：0x048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LW[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LW[15:0]**：ラインウォーターマーク

これらのビットによって、割込み生成のラインウォーターマークを設定することができます。

ウォーターマークを付けたラインの最後のピクセルが転送されたとき割込みが発生します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

19.5.23 DMA2D AHBマスタタイマ設定レジスタ（DMA2D_AMTCR）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **DT[7:0]**：デッドタイム

AHBマスタポートに対する2つの連続アクセス間に挿入されるAHBクロックサイクルのデッドタイム値です。これらのビットは2つの連続AHBアクセス間の最小保証サイクル数を表します。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：有効化

デットタイム機能を有効にします。

19.5.24 DMA2Dフォアグラウンド CLUT（DMA2D_FGCLUTx）

アドレスオフセット：0x400 + 0x4 * x（x = 0から255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ

フォアグラウンドのインデックス{x}のアルファ値

ビット23:16 **RED[7:0]**：赤

フォアグラウンドのインデックス{x}の赤の値。

ビット15:8 **GREEN[7:0]**：緑

フォアグラウンドのインデックス{x}の緑の値。

ビット7:0 **BLUE[7:0]**：青

フォアグラウンドのインデックス{x}の青の値。

19.5.25 DMA2Dバックグラウンド CLUT（DMA2D_BGCLUTx）

アドレスオフセット：0x800 + 0x4 * x（x = 0から255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ
バックグラウンドのインデックス{x}のアルファ値

ビット23:16 **RED[7:0]**：赤
バックグラウンドのインデックス{x}の赤の値

ビット15:8 **GREEN[7:0]**：緑
バックグラウンドのインデックス{x}の緑の値

ビット7:0 **BLUE[7:0]**：青
バックグラウンドのインデックス{x}の青の値

19.5.26 DMA2Dレジスタマップ

表166.DMA2Dレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMA2D_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]			Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
	リセット値														0	0	0			0	0	0	0	0	0		0				0	0	0
0x004	DMA2D_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
	リセット値																											0	0	0	0	0	0
0x008	DMA2D_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
	リセット値																											0	0	0	0	0	0
0x00C	DMA2D_FGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	DMA2D_FGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	DMA2D_BGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	DMA2D_BGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	DMA2D_FGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0].								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0

表166.DMA2Dレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x020	DMA2D_FGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	DMA2D_BGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM [1:0]		CS[7:0].								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x028	DMA2D_BGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	DMA2D_FGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	DMA2D_BGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	DMA2D_OPFCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
	リセット値											0	0												0						0	0	0
0x038	DMA2D_OCOLR ARGB8888またはRGB888カラーモード	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA2D_OCOLR RGB565カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[4:0]					GREEN[6:0]						BLUE[4:0]				
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA2D_OCOLR ARGB1555カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	A	RED[4:0]				GREEN[4:0]						BLUE[4:0]				
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA2D_OCOLR ARGB4444カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	DMA2D_OMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	DMA2D_OOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	DMA2D_NLR	Res.	Res.	PL[13:0]														NL[15:0]															
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	DMA2D_LWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LW[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	DMA2D_AMTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																	0	0	0	0	0	0	0	0								0
0x050 ~ 0x3FC	予約済み	予約済み																															
0x400 + 0x4 * x、（x = 0 ~ 255）	DMA2D_FGCLUTx	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x800 + 0x4 * x、（x = 0 ~ 255）	DMA2D_BGCLUTx	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、セクション2.3 を参照してください。

20 Chrom-Art Acceleratorコントローラ（DMA2D）

このセクションは、STM32U5Fx/5Gxデバイスにのみ適用されます。

20.1 DMA2Dの概要

Chrom-ARTアクセラレータ（DMA2D）はイメージ操作専用の特殊なDMAです。以下の操作を実行できます。

- 転送先イメージの一部または全部を特定の色で塗りつぶす
- 転送元イメージの一部または全部を、転送先イメージの一部または全部にコピーする
- 転送元イメージの一部または全部を、ピクセルフォーマット変換を行って転送先イメージの一部または全部にコピーする
- 異なるピクセルフォーマットを持つ転送元イメージの全部または一部をブレンドし、その結果を別のピクセルフォーマットで転送先イメージの一部または全体にコピーする

JPEGデコーダ出力を処理するためのブロックベースのYCbCrを含む、すべての標準的なピクセルフォーマットがサポートされており、4bit/pixelから32bit/pixelのインデックスカラーモードまたはダイレクトカラーモードに対応しています。DMA2DはCLUT（color look-up tables：カラールックアップテーブル）用に専用メモリを搭載しています。

20.2 DMA2Dの主な機能

DMA2Dの主な機能は次のとおりです。

- シングルAHBマスタバスアーキテクチャ
- 8/16/32bitアクセスをサポートするAHBスレーブプログラミングインタフェース（32bitのCLUTアクセスを除く）
- ユーザがプログラム可能なワーク領域サイズ
- ピクセルまたはバイトで表現された転送元および転送先領域用の、ユーザーがプログラム可能なオフセット
- 全メモリ空間でユーザがプログラム可能な転送元および転送先のアドレス
- 最大2つの転送元のブレンド操作
- 調整可能なアルファ値（ソース値、固定値、または調整値）
- ユーザが任意に設定可能な転送元および転送先のカラーフォーマット
- インダイレクトカラーまたはダイレクトカラーで4bit/pixelから32bit/pixelまで、最大12のカラーフォーマットをサポート。
- 4:4:4、4:2:2、および4:2:0のクロマサブサンプリング係数のブロックベース（8x8）のYCbCrをサポート
- インダイレクトカラーモードにおけるCLUT格納用に2つの内部メモリを搭載
- 自動CLUTロードまたはCPUを使用したCLUTプログラミング
- ユーザが任意にプログラム可能なCLUTサイズ
- AHB帯域幅を制御する内部タイマ

- 6つの動作モード：レジスタからメモリへの転送、メモリ間転送、ピクセルフォーマット変換を伴うメモリ間転送、ピクセルフォーマット変換およびブレンドを伴うメモリ間転送、ピクセルフォーマット変換、ブレンドおよび固定カラーフォアグラウンドを伴うメモリ間転送、ピクセルフォーマット変換、ブレンドおよび固定カラーバックグラウンドを伴うメモリ間転送。
- 固定カラーでの矩形領域塗りつぶし
- 領域間のコピー
- 転送元イメージと転送先イメージ間のピクセルフォーマット変換を伴うコピー
- ピクセルフォーマットの異なる2つの転送元のコピー、およびブレンド
- パラレルインタフェースを通してディスプレイのリフレッシュをサポートするための出力バッファバイトスワッピング
- DMA2D動作の強制終了および一時停止
- ユーザが任意に設定可能な転送先ラインにおけるウォーターマーク割込み
- バスエラーまたはアクセス競合時の割込み生成
- プロセス完了時の割込み生成

20.3 DMA2Dの機能詳細

20.3.1 概要

DMA2Dコントローラはダイレクトメモリ転送を行います。AHBマスタとして、AHBバスマトリックスを制御してAHBトランザクションを開始することができます。

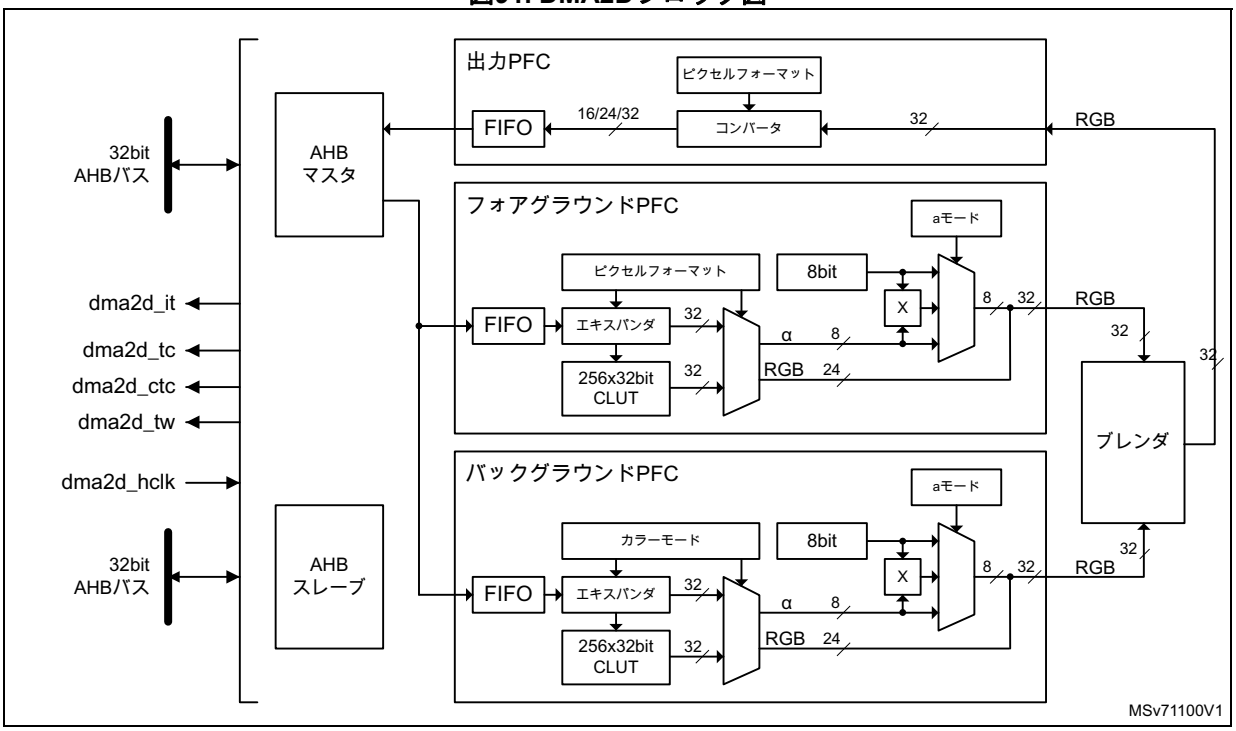
DMA2Dは、次のモードで動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- ピクセルフォーマット変換を伴うメモリ間転送
- ピクセルフォーマット変換およびブレンドを伴うメモリ間転送
- ピクセルフォーマット変換、ブレンドおよび固定カラーフォアグラウンドを伴うメモリ間転送
- ピクセルフォーマット変換、ブレンドおよび固定カラーバックグラウンドを伴うメモリ間転送

AHBスレーブポートは、DMA2Dコントローラをプログラムするために使用されます。

DMA2Dインタフェースのブロック図を図91に示します。

図91. DMA2Dブロック図



20.3.2 DMA2D内部信号

DMA2Dの内部信号を次の図に示します。

表167.DMA2D内部信号

名前	信号タイプ	説明
dma2d_hclk	入力	DMA2D AHBクロック
dma2d_it	出力	DMA2Dグローバル割込みリクエスト
dma2d_ctc	出力	DMA2D CLUT転送完了トリガ
dma2d_tc	出力	DMA2D転送完了トリガ
dma2d_tw	出力	DMA2D転送ウォーターマークトリガ

表168.DMA2Dトリガ相互接続

トリガ名	方向	トリガ送信元／送信先
dma2d_tc	出力	gpdma_trigsel[50]
dma2d_ctc	出力	gpdma_trigsel[51]
dma2d_tw	出力	gpdma_trigsel[52]

20.3.3 DMA2D制御

DMA2Dコントローラは、DMA2D_CRを使用して設定します。ユーザアプリケーションで次の操作を実行できます。

- 動作モードの選択
- DMA2D割込みの有効化／無効化
- 進行中のデータ転送の開始／一時停止／強制終了

20.3.4 DMA2DフォアグラウンドおよびバックグラウンドFIFO

DMA2Dフォアグラウンド（FG）FIFOおよびバックグラウンド（BG）FIFOはコピーなどの処理をする入力データをフェッチします。

これらのFIFOは、それぞれのピクセルフォーマットコンバータ（PFC）に定義されているピクセルフォーマットに従ってピクセルデータをフェッチします。

これらは、次の制御レジスタにより設定可能です。

- DMA2Dフォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）
- DMA2Dフォアグラウンドオフセットレジスタ（DMA2D_FGOR）
- DMA2Dバックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）
- DMA2Dバックグラウンドオフセットレジスタ（DMA2D_BGOR）
- DMA2Dライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2Dがレジスタからメモリへの転送モードで動作する場合、どちらのFIFOもアクティブになりません。

DMA2Dがメモリ間転送モードで動作する（ピクセルフォーマット変換、ブレンド処理のいずれも行わない）場合は、FG FIFOのみアクティブになりバッファとして機能します。

DMA2Dがピクセルフォーマット変換を伴うメモリ間転送モードで動作する（ブレンド処理を行わない）場合は、BG FIFOはアクティブになりません。

20.3.5 DMA2DフォアグラウンドおよびバックグラウンドPFC

DMA2Dフォアグラウンドピクセルフォーマットコンバータ（PFC）およびバックグラウンドピクセルフォーマットコンバータはピクセルフォーマット変換を行って32bit/pixelのデータを生成します。PFCはアルファ値に調整を加えることもできます。

PFCの一段目ではカラーフォーマットを変換します。フォアグラウンドピクセルとバックグラウンドピクセルの元のカラーフォーマットは、それぞれDMA2D_FGPFCCRおよびDMA2D_BGPFCCRのCM[3:0]によって設定します。

サポートされている入力フォーマットを次の表に示します。

表169.入力でサポートされているカラーモード

CM[3:0]	ピクセルフォーマット
0000	ARGB8888
0001	RGB888
0010	RGB565
0011	ARGB1555
0100	ARGB4444

表169.入力でサポートされているカラーモード （続き）

CM[3:0]	ピクセルフォーマット
0101	L8
0110	AL44
0111	AL88
1000	L4
1001	A8
1010	A4
1011	YCbCr（フォアグラウンドの場合のみ）

ピクセルフォーマットは次のとおりコード化されています。

- アルファ値フィールド：透明度
  0xFF値は不透明ピクセルに対応し0x00は透明ピクセルに対応します。
- Rフィールドは赤
- Gフィールドは緑
- Bフィールドは青
- Lフィールド：輝度
  このフィールドはRGBから3つの、ARGBから4つの成分を読み出すためのCLUTのインデックスです。

元のフォーマットがダイレクトカラーモード（ARGB/RGB）の場合、MSBをLSBへコピーしてチャネルあたり8bitへの拡張が行われます。これにより変換の線形性が完全に保証されます。

元のフォーマットにアルファ値が無い場合、アルファ値は自動的に0xFF（不透明）にセットされます。

元のフォーマットがインダイレクトカラーモード（L/AL）の場合、CLUTが必要となり、各ピクセルフォーマットコンバータは、256エントリの32bit CLUTと連動します。

特定のアルファモードA4およびA8では、カラー情報の格納、インデックスの利用のいずれも行われません。イメージ生成に使用されるカラーは固定となっており、フォアグラウンドピクセル用のカラーはDMA2D_FGCOLRで、バックグラウンドピクセル用のカラーはDMA2D_BGCOLRで定義されます。

システムメモリ内のフィールドの順序は次の表で定義しています。

表170.メモリ内のデータ順序

ピクセル フォーマット	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]
L8	L_3[7:0]	L_2[7:0]	L_1[7:0]	L_0[7:0]
AL44	A_3[3:0]L_3[3:0]	A_2[3:0]L_2[3:0]	A_1[3:0]L_1[3:0]	A_0[3:0]L_0[3:0]
AL88	A_1[7:0]	L_1[7:0]	A_0[7:0]	L_0[7:0]
L4	L_7[3:0]L_6[3:0]	L_5[3:0]L_4[3:0]	L_3[3:0]L_2[3:0]	L_1[3:0]L_0[3:0]
A8	A_3[7:0]	A_2[7:0]	A_1[7:0]	A_0[7:0]
A4	A_7[3:0]A_6[3:0]	A_5[3:0]A_4[3:0]	A_3[3:0]A_2[3:0]	A_1[3:0]A_0[3:0]

32bitに調整された24bit RGB888はARGB8888モードでサポートされます。

32bit値が生成されると、表171に示すようにDMA2D_FGPFCCRまたはDMA2D_BGPFCCRのAM[1:0]に従ってアルファチャネルを調整することができます。

アルファチャネルは次のいずれかの状態になります。

- そのままが維持されます（調整なし）。
- DMA2D_FGPFCCR/DMA2D_BGPFCCRでALPHA[7:0]の値によって置き換えられます。
- DMA2D_FGPFCCR/DMA2D_BGPFCCRのALPHA[7:0]値を255で除算した後、元のアルファ値に乗算した値で置き換えられます。

表171.アルファモード設定

AM[1:0]	アルファモード
00	調整なし
01	DMA2D_xxPFCCRの値で置換
10	DMA2D_xxPFCCRを255で除算した値を元の値に乗算して置換
11	予約済みです。

注： **別のフォーマットをサポートするために、DMA2D_FGPFCCR/DMA2D_BGPFCCR の AI をセットして、アルファ値を反転できます。これは、DMA2D_FGPFCCR/DMA2D_BGPFCCRおよびCLUTに格納されるアルファ値にも適用されます。**

DMA2D_FGPFCCRまたはDMA2D_BGPFCCRのRBSをセットして、RフィールドとBフィールドを入れ替えることもできます。これは、CLUTや、DMA2D_FGCOLRまたはDMA2D_BGCOLRで使用されるRGB順序にも適用されます。

20.3.6 DMA2DフォアグラウンドおよびバックグラウンドCLUTインタフェース

CLUTメモリアクセスおよびCLUTの自動ロードはCLUTインタフェースによって管理されます。

次の3種類のアクセスが可能です。

- ピクセルフォーマット変換操作中のPFCによるCLUTの読出し
- CPUがCLUTに対してデータの読出しまたは書込みを行っているときのAHBスレーブポートを経由したCLUTへのアクセス
- CLUTの自動ロードが行われているときのAHBマスタポートを経由したCLUTへの書込み

CLUTメモリロードは次の2つの方法で行うことができます。

- 自動ロード

  CLUTのロードは、次の順序で行う必要があります。

  a) DMA2D_FGCMAR（フォアグラウンドCLUT）またはDMA2D_BGCMAR（バックグラウンドCLUT）にCLUTアドレスを設定します。

  b) DMA2D_FGPFCCR（フォアグラウンドCLUT）、またはDMA2D_BGPFCCR（バックグラウンドCLUT）のCS[7:0]フィールドにCLUTサイズを設定します。

  c) DMA2D_FGPFCCR（フォアグラウンドCLUT）またはDMA2D_BGPFCCR（バックグラウンドCLUT）のSTARTをセットして転送を開始します。この自動ロード処理中、CPUはCLUTにアクセスできません。競合が発生した場合、CLUTアクセスエラー割込みが発生します。ただし、DMA2D_CRでCAEIE = 1に設定されている必要があります。

- 手動ロード

  ローカルCLUTメモリがマップされているDMA2D AHBスレーブポートを使用して、アプリケーションが手動でCLUTを設定する必要があります。フォアグラウンドCLUT（FGCLUT）はアドレスオフセット0x0400に配置され、バックグラウンドCLUT（BGCLUT）はアドレスオフセット0x0800に配置されます。

CLUTフォーマットは、24bitまたは32bitが可能です。次の表に示すようにDMA2D_FGPFCCR（フォアグラウンドCLUT）またはDMA2D_BGPFCCR（バックグラウンドCLUT）のCCMを介してCLUTフォーマットを設定します。

表172.サポートされているCLUTカラーモード

CCM	CLUTカラーモード
0	32bit ARGB8888
1	24bit RGB888

システムメモリ内のCLUTデータの構成方法は次の表に規定しています。

表173.メモリ内のCLUTデータ順序

CLUTカラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]

20.3.7 DMA2Dブレンダ

DMA2Dブレンダは、転送元ピクセルをペアでブレンドし、出来上がりのピクセルを計算します。

ブレンドは次の式に従って行われます。

$$\text{次の場合:}\alpha\text{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha OUT = \alpha FG + \alpha BG - \alpha Mult$$

$$COUT = \frac{CFG.\alpha FG + CBG.\alpha BG - CBG.\alpha Mult}{\alpha OUT}$$ ここでC = RまたはGまたはB

除算では、最も近い整数に切り捨てられます。

ブレンダは設定レジスタを必要としません。ブレンダの使用方法はDMA2D_CRのMODE[2:0]で定義されるDMA2D動作モードによって異なります。

20.3.8 DMA2D出力PFC

出力PFCは、32bitからDMA2D_OPFCCRのCM[2:0]で定義された出力フォーマットへのピクセルフォーマット変換を行います。

サポートされている出力フォーマットを次の表に示します

表174.出力でサポートされているカラーモード

CM[2:0]	ピクセルフォーマット
000	ARGB8888
001	RGB888
010	RGB565
011	ARGB1555
100	ARGB4444

注： **別のフォーマットをサポートするために、DMA2D_OPFCCRのAIビットをセットして、計算されたアルファ値を反転できます。これは、DMA2D_OCOLRで使用されるアルファ値にも適用されます。**

DMA2D_OPFCCRのRBSをセットして、Rフィールドと Bフィールドを入れ替えることもできます。これは、DMA2D_OCOLRで使用されるRGB順序にも適用されます。

20.3.9 DMA2D出力FIFO

出力FIFOは、出力PFCに定義されているカラーフォーマットに従ってピクセルをプログラムします。

転送先領域は、次のレジスタによって定義します。

- DMA2D出力メモリアドレスレジスタ（DMA2D_OMAR）
- DMA2D出力オフセットレジスタ（DMA2D_OOR）
- DMA2Dライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2Dがレジスタからメモリモードで動作する場合、設定された出力矩形領域が、32bit、24bitまたは16bitの固定値が格納されたDMA2D_OCOLRで指定された色で塗りつぶされます。このフォーマットは、DMA2D_OPFCCRのCM[2:0]によって選択されます。

データは、次の表で定義されている順番でメモリに格納されます。

表175.メモリ内のデータ順序

ピクセルフォーマット	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]

32bitに調整されたRGB888はARGB8888モードでサポートされます。

20.3.10 DMA2D出力FIFOバイトの順序変更

出力FIFOバイトは、DMA2Dから直接パラレルインタフェース（F（S）MC）を介してディスプレイフレームバッファの更新をサポートするように並べ替えることができます。

バイトの順序変更は以下のように行われます。

- 赤と青のコンポーネントをスワップするRBSビット
- 出力FIFOで2 x 2でスワップするSBビット

バイトのスワッピングがアクティブになっている場合（DMA2D_OPFCRのSB = 1）、1行あたりのピクセル数（DMA2D_NLRのPLフィールド）は偶数、出力メモリアドレス（DMA2D_OMARのMAフィールド）は偶数である必要があります。バイト単位で計算された出力ラインオフセット（DMA2D_CRのLOMフィールドとDMA2D_OOR値のLOフィールドの結果）も偶数である必要があります。そうでない場合、設定エラーが検出されます。

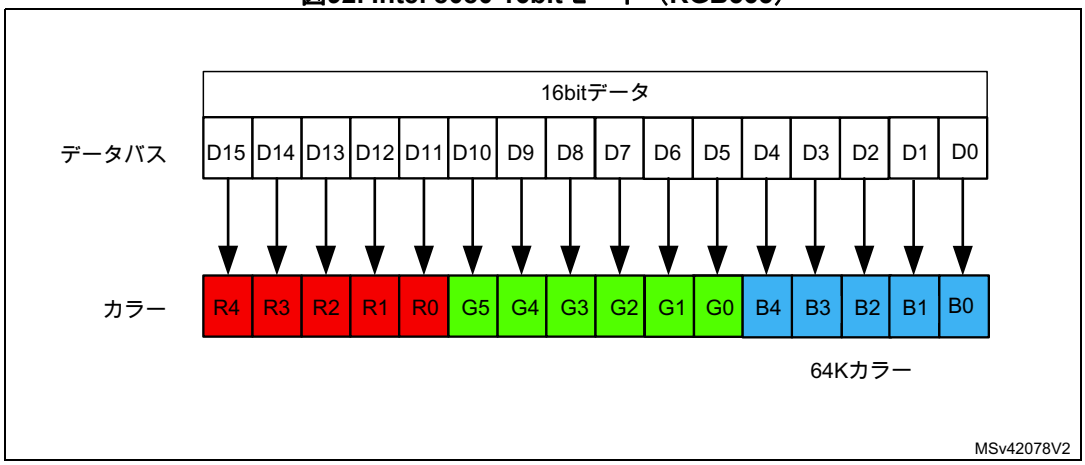
表176.メモリ内の標準データ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]

16bitモード（RGB565）

このモードは、DMA2Dによるバイトの並べ替えなしでサポートされます。

図92. Intel 8080 16bitモード（RGB565）



18/24bitモード（RGB888）

このモードは、データの並び替えが必要です。

1. 赤と青をスワップしなければなりません（RBSビットのセット）
2. ハーフワードのMSBおよびLSBバイトはスワップしなければなりません（SBビットのセット）。

図93. Intel 8080 18/24bitモード（RGB888）

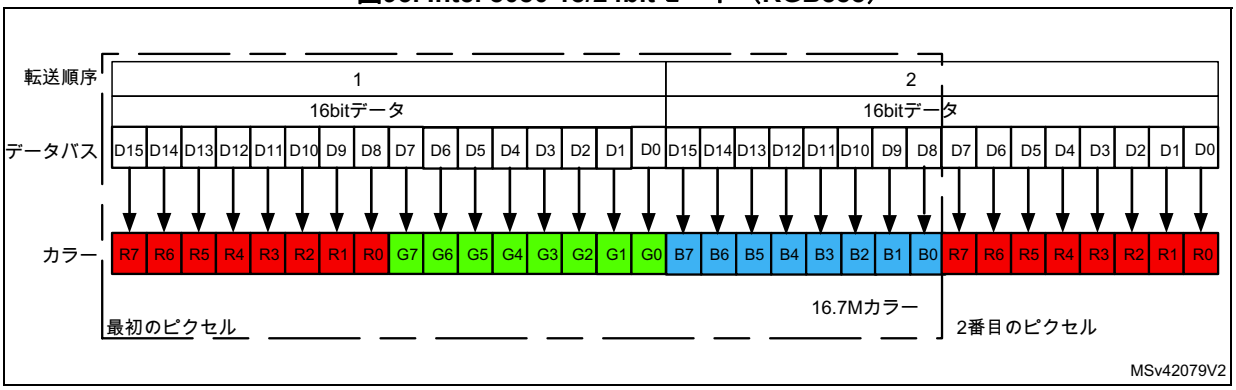


表177.出力FIFOバイトの順序変更ステップ

ステップ	@ + 3	@ + 2	@ + 1	@ + 0
元のデータ順序	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RBSビットのセット				
赤と青のスワップ後のデータ順序（RBSセット）	R_1[7:0]	B_0[7:0]	G_0[7:0]	R_0[7:0]
	G_2[7:0]	R_2[7:0]	B_1[7:0]	G_1[7:0]
	B_3[7:0]	G_3[7:0]	R_3[7:0]	B_2[7:0]
SBビットのセット				

表177.出力FIFOバイトの順序変更ステップ （続き）

ステップ	@ + 3	@ + 2	@ + 1	@ + 0
バイトスワップ後のデータ順序（SBセット）	B_0[7:0]	R_1[7:0]	R_0[7:0]	G_0[7:0]
	R_2[7:0]	G_2[7:0]	G_1[7:0]	B_1[7:0]
	G_3[7:0]	B_3[7:0]	B_2[7:0]	R_3[7:0]

20.3.11 DMA2DAHBマスタポートタイマ

AHBマスタポートには8bitのタイマが組込まれており、クロスバーの帯域幅を任意に制限することができます。このタイマはAHBクロックによって駆動され、2つの連続アクセス間のデッドタイムをカウントします。これによって帯域幅の使用を制限することができます。

タイマの有効化とデッドタイム値はDMA2D_AMPTCRによって設定します。

20.3.12 DMA2Dトランザクション

DMA2D トランザクションは、一定数のデータ転送のシーケンスで構成されています。データの数と幅はソフトウェアによりプログラムすることができます。

各DMA2Dデータ転送は、最大4つのステップで構成されています。

1. DMA2D_FGMAR で示されるメモリ位置からのデータロードおよび DMA2D_FGCR で定義されるピクセルフォーマット変換。
2. DMA2D_BGMAR で示されるメモリ位置からのデータロードおよび DMA2D_BGCR で定義されるピクセルフォーマット変換。
3. アルファ値に関する PFC 操作の結果として生じるアルファチャネルに従ったすべての読出しピクセルのブレンド。
4. 出力するピクセルをDMA2D_OCRに従ってピクセルフォーマット変換およびDMA2D_OMARによってアドレス指定したメモリ位置へのデータの書込み。

20.3.13 DMA2D設定

データ転送元およびデータ転送先は両方ともアドレス0x0000 0000と0xFFFF FFFFの間の全4Gバイトメモリ領域内のペリフェラルおよびメモリを対象にすることができます。

DMA2Dは、DMA2D_CRのMODE[2:0]を使用して選択された次のいずれのモードでも動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- PFCを伴うメモリ間転送
- PFCとブレンドを伴うメモリ間転送
- PFC、ブレンドおよび固定FGカラーを伴うメモリ間転送
- PFC、ブレンドおよび固定BGカラーを伴うメモリ間転送

レジスタからメモリへの転送

“レジスタからメモリ”モードは、ユーザ定義領域を事前定義された色で塗りつぶすために使用します。カラーフォーマットはDMA2D_OPFCCRでセットします。

DMA2Dはどのソースからもデータのフェッチを行いません。DMA2D_OCOLRで定義された色を、DMA2D_OMARでアドレス指定し、DMA2D_NLRおよびDMA2D_OORで定義した領域に書き込むだけです。

メモリ間転送

メモリ間モードでは、DMA2Dはグラフィカルなデータの変換は何も行いません。フォアグラウンド入力FIFOはバッファの役目を果たし、データは、DMA2D_FGMARで定義される転送元のメモリ位置からDMA2D_OMARが示す転送先のメモリ位置へ転送されます。

DMA2D_FGPFCCRのCM[3:0]でプログラムされているカラーモードによって、入力および出力の両方のピクセルあたりビット数を定義します。

転送する領域のサイズは、転送元に対してはDMA2D_NLRおよびDMA2D_FGORで定義され、転送先に対してはDMA2D_NLRおよびDMA2D_OORで定義されます。

PFCを伴うメモリ間転送

このモードではDMA2Dは転送元データのピクセルフォーマット変換を行い、それを転送先メモリ位置に格納します。

転送する領域のサイズは、転送元に対してはDMA2D_NLRおよびDMA2D_FGORで定義され、転送先に対してはDMA2D_NLRおよびDMA2D_OORで定義されます。

データはDMA2D_FGMARで定義された場所からフェッチされ、フォアグラウンドPFCによって処理されます。元のピクセルフォーマットはDMA2D_FGPFCCRを使用して設定されます。

元のピクセルフォーマットがダイレクトカラーモードである場合、カラーチャネルはすべて8bitに拡張されます。

ピクセルフォーマットがインダイレクトカラーモードである場合、関連するCLUTをCLUTメモリにロードする必要があります。

CLUTのロードは以下の流れに従って自動的に行うことができます。

1. CLUTアドレスをDMA2D_FGCMARにセットします。
2. DMA2D_FGPFCCRのCS[7:0]によりCLUTサイズをセットします。
3. DMA2D_FGPFCCRのCCMによりCLUTフォーマット（24bitまたは32bit）をセットします。
4. DMA2D_FGPFCCRにSTARTをセットして、CLUTロードを開始します。

CLUTロードが完了すると、DMA2D_IFRのCTCIFフラグが立てたれ、DMA2D_CRでCTCIEがセットされていれば割込みが生成されます。自動CLUTロード処理は標準的なDMA2D転送と並行して動作することはできません。

CLUTはCPUから書き込むこともできれば、AHBポート経由で他のマスタによって書き込むこともできます。DMA2D転送が進行中でかつCLUTを使用しているとき（インダイレクトカラーフォーマット）、CLUTにはアクセスできません。

カラー変換処理と並行して、DMA2D_FGPFCCRにプログラムされている値によっては、アルファ値を追加または変更することができます。元のイメージにアルファチャネルが無い場合、デフォルトのアルファ値0xFFが自動的に追加されて完全に不透明なピクセルになります。

アルファ値はDMA2D_FGPFCCRのAM[1:0]に従って調整されます。

- 変更なし。

- DMA2D_FGPFCCRのALPHA[7:0]の値によって置き換えられます。
- DMA2D_FGPFCCRのALPHA[7:0]値を255で除算した後、元の値に乗算した値で置き換えられます。

出力される32bitデータは出力PFCによって、DMA2D_OPFCCRのCM[2:0]で指定されたフォーマットにエンコードされます。CLUT生成処理がサポートされていないため、出力ピクセルフォーマットをインダイレクトモードにすることはできません。

処理されたデータはDMA2D_OMARが指す転送先メモリ位置に書き込まれます。

PFCとブレンドを伴うメモリ間転送

このモードでは、DMA2D_FGMARおよびDMA2D_BGMARで定義されるメモリ位置からフォアグラウンドFIFOおよびバックグラウンドFIFOに2つの転送元データがフェッチされます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換された後、それらは次の式に従ってブレンドされます。

$$\text{次の場合:}\ \alpha Mult = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha OUT = \alpha FG + \alpha BG - \alpha Mult$$

$$COUT = \frac{CFG.\alpha FG + CBG.\alpha BG - CBG.\alpha Mult}{\alpha OUT} \quad \text{ここで C = R または G または B}$$

除算では、小数点以下は切り捨てられます。

出力される32bitピクセル値は出力PFCにより指定出力フォーマットに従ってエンコードされ、データはDMA2D_OMARが指す転送先のメモリ位置に書き込まれます。

PFC、ブレンドおよび固定カラー FGを伴うメモリ間転送

このモードでは、DMA2D_BGMARで定義されるメモリ位置からバックグラウンドFIFOに1つの転送元データのみがフェッチされます。

フォアグラウンドカラーの値はDMA2D_FGCOLRによって指定され、アルファ値は0xFF（不透明）にセットされます。

アルファ値は、DMA2D_FGPFCCRのAM[1:0]およびALPHA[7:0]に従って置き換え、または調整することができます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換された後、すべてのピクセルが一緒にブレンドされます。出力される32bitピクセル値は出力PFCにより指定出力フォーマットに従ってエンコードされます。データはDMA2D_OMARが指す転送先メモリ位置に書き込まれます。

PFC、ブレンドおよび固定カラー BGを伴うメモリ間転送

このモードでは、DMA2D_FGMARで定義されるメモリ位置からフォアグラウンドFIFOに1つの転送元データのみがフェッチされます。

バックグラウンドカラーの値はDMA2D_BGCOLRによって指定され、アルファ値は0xFF（不透明）に設定されます。

アルファ値は、DMA2D_BGPFCCRのAM[1:0]およびALPHA[7:0]に従って置き換え、または調整することができます。

メモリ間モードで説明しているように、2つのピクセルフォーマットコンバータを設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自のCLUTメモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれのPFCによって32bitに変換された後、全てのピクセルが一緒にブレンドされます。出力される32bitピクセル値は出力PFCにより指定出力フォーマットに従ってエンコードされます。データはDMA2D_OMARが指す転送先メモリ位置に書き込まれます。

設定エラー検出

DMA2Dは転送を行う前に設定が正しいことを確認します。新規の転送／自動ロードが開始されるとき、誤った設定が検出されると、ハードウェアによって設定エラー割込みフラグがセットされます。DMA2D_CRでCEIE = 1である場合、割込みが生成されます。

検出することができる誤った設定は次のとおりです。

- フォアグラウンドCLUT自動ロード：DMA2D_FGCMARのMAビットがDMA2D_FGPFCCRのCCMと整合が取れていません。
- バックグラウンドCLUT自動ロード：DMA2D_BGCMARのMAビットがDMA2D_BGPFCCRのCCMと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンドおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGMARのMAビットがDMA2D_FGPFCCRのCMと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンドおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGPFCCRのCMビットが無効です。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンドおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_NLRのPLビットが奇数で、DMA2D_FGPFCCRのCMがA4またはL4です。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンドおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGORのLOビットが奇数で、DMA2D_FGPFCCRのCMがA4またはL4で、DMA2D_CRのLOMがピクセルモードです。
- メモリ転送（"ブレンド"モードのみ、および、"ブレンドおよび固定カラーFGを伴うメモリ間"モードを除く）：DMA2D_FGPFCCRのCMビットが無効です。
- メモリ転送：（" ブレンド " モード、および、" 固定カラーFG 伴うブレンド " モードのみ）：DMA2D_BGPFCCRのCMビットが無効です。
- メモリ転送：（" ブレンド " モード、および、" 固定カラーFG を伴うブレンド " モードのみ）：DMA2D_NLRのPLビットが奇数で、DMA2D_BGPFCCRのCMがA4またはL4です。
- メモリ転送（" ブレンド " モード、および、" 固定カラーFG を伴うブレンド " モードのみ）：DMA2D_BGORのLOビットが奇数で、DMA2D_BGPFCCRのCMがA4またはL4で、DMA2D_CRのLOMがピクセルモードです。
- メモリ転送（メモリ間を除く）：DMA2D_OMARのMAビットがDMA2D_OPFCCRのCMと整合が取れていません。
- メモリ転送（メモリ間を除く）：DMA2D_OPFCCRのCMビットが無効です。

- バイトスワッピングを伴うメモリ転送：DMA2D_NLRのPLビットが奇数またはDMA2D_OMARのMAビットが奇数、または（DMA2D_CRのLOMとDMA2D_OORのLOから生じる）バイトのLOが奇数で、DMA2D_OPFCCRでSB = 1です。
- メモリ転送：DMA2D_NLRのNLビットがすべてゼロです。
- メモリ転送：DMA2D_NLRのPLビットがすべてゼロです。
- メモリ転送：DMA2D_CRのMODEビットが無効です。
- YCbCrフォーマット：DMA2D_FGPFCCRのSTARTをセットすることによりCLUTロードが開始された場合
- YCbCrフォーマット：メモリ間モードが選択されている場合
- YCbCrフォーマット：YCbCR4:4:4が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が8ピクセルの倍数でない場合
- YCbCrフォーマット：YCbCr4:2:2またはYCbCr4:2:0が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が16ピクセルの倍数でない場合

20.3.14 YCbCrサポート

DMA2Dフォアグラウンド面では、8x8のブロックベースのYCbCrを、異なるクロマサブサンプリング係数を持つJPEGデコーダからの出力のようにサポートできます。

メモリ構成は標準JFIFルールに従います。

- 3つの色のコンポーネントをそれぞれ8bitでコード化する必要があります。
- 各コンポーネントは、MCUと呼ばれる8x8（64バイト）のブロックで整列する必要があります。

クロマサブサンプリング係数によっては、次の表を参照してMCUをメモリに配置する必要があります。

表178.メモリ内のMCU順序

サブサンプリング	@	@ + 64	@ + 128	@+192	@+256	@ + 320
4:4:4	Y_1	Cb_1	Cr_1	Y_2	Cb_2	Cr_2
4:2:2	Y_1	Y_2	Cb_{12}	Cr_{12}	Y_3	Y_4
4:2:0	Y_1	Y_2	Y_3	Y_4	Cb_{1234}	Cr_{1234}

クロマサブサンプリング係数は、DMA2D_FGPFCCRのCSSを介して設定されます。

YCbCrカラーモードで設定されたフォアグラウンドでDMA2Dが開始されると、最初の2つのクロマMCUがフォアグラウンドのCLUTにロードされます。クロマMCUがロードされると、従来のカラーモードと同様に、DMA2DがY MCUのロードを実行します。

20.3.15 DMA2D転送制御（開始、一時停止、強制終了、完了）

DMA2Dが設定されると、DMA2D_CRのSTARTをセットすることにより転送を開始できます。転送が完了すると、START は自動的にリセットされ、DMA2D_ISR の TCIF フラグが立てられます。DMA2D_CRでTCIEがセットされている場合は、割込みを生成することができます。

DMA2D_CRのSUSPをセットすることにより、ユーザアプリケーションからいつでもDMA2Dを一時停止することができます。その後、DMA2D_CRのABORTをセットすることにより、トランザクションを強制終了することができます。また、DMA2D_CRのSUSPをリセットすることにより再開することができます。

DMA2D_CRのABORTをセットすることにより、ユーザアプリケーションからいつでも進行中のトランザクションを強制終了することができます。この場合、TCIFフラグは立ちません。

DMA2D_FGPFCCRおよびDMA2D_BGPFCCRのそれぞれのSTARTビットを使用することによって、自動CLUT転送も強制終了したり一時停止したりすることができます。

20.3.16 ウォーターマーク

ウォーターマークをプログラムして所定のラインの最後のピクセルが転送先のメモリ領域に書き込まれたときに割込みを生成することができます。

ライン番号は、DMA2D_LWRのLW[15:0]で定義します。

このラインの最後のピクセルが転送されると、DMA2D_CR で TWIE がセットされていれば、DMA2D_ISRのTWIFフラグが立ち、割込みが生成されます。

20.3.17 エラー管理

次の2種類のエラーをトリガすることができます。

- AHBDMA2D_ISRのTEIFフラグによって通知されるマスタポートエラー
- CLUTアクセス（CLUTロード中またはDMA2D転送中にCPUがCLUTにアクセスしようとする）によって引き起こされる競合。DMA2D_ISRレジスタのCAEIFフラグによって通知されます。

両方のフラグはそれぞれDMA2D_CRの割込みイネーブルフラグに関連付けられており、必要に応じて割込みを生成します（TEIEとCAEIE）。

20.3.18 AHBデッドタイム

AHB帯域幅の使用を制限するために、2つの連続したAHBアクセス間のデッドタイムをプログラムすることができます。この機能は、DMA2D_AMTCRのENをセットすることによって有効にできます。

デッドタイム値は、DMA2D_AMTCRのDT[7:0]に格納されます。この値は、AHBバス上の2つの連続するトランザクション間の最小保証サイクル数を表します。

DMA2Dが実行中のデッドタイム値の更新は、次のAHB転送に反映されます。

20.4 DMA2D割込み

割込みは次のイベントによって生成できます。

- 設定エラー
- CLUT転送完了
- CLUTアクセスエラー
- 転送ウォーターマークに到達
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表179.DMA2D割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット
設定エラー	CEIF	CEIE
CLUT転送完了	CTCIF	CTCIE
CLUTアクセスエラー	CAEIF	CAEIE
転送ウォーターマーク	TWF	TWIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

20.5 DMA2Dレジスタ

20.5.1 DMA2D制御レジスタ（DMA2D_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
		rw	rw	rw	rw	rw	rw		rw				rs	rw	rs

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **MODE[2:0]**：DMA2Dモード

このビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

000：メモリ間転送（FGフェッチのみ）

001：PFCを伴うメモリ間転送（FG PFCアクティブでFGフェッチのみ）

010：ブレンドを伴うメモリ間転送（PFCおよびブレンドを伴うFGフェッチおよびBGフェッチ）

011：レジスタからメモリ転送（FGおよびBGなし、出力ステージのみアクティブ）

100:ブレンドおよび固定カラーFGを伴うメモリ間転送(FG PFCおよびBG PFCアクティブでBGフェッチのみ）

101:ブレンドおよび固定カラーBGを伴うメモリ間転送(FG PFCおよびBG PFCアクティブでFGフェッチのみ）

その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CEIE**：設定エラー（CE）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CE割込みは無効です。

1：CE割込みは有効です。

ビット12 **CTCIE**：CLUT転送完了（CTC）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CTC割込みは無効です。

1：CTC割込みは有効です。

ビット11 **CAEIE**：CLUTアクセスエラー（CAE）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：CAE割込みは無効です。

1：CAE割込みは有効です。

ビット10 **TWIE**：転送ウォーターマーク（TW）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TW割込みは無効です。

1：TW割込みは有効です。

ビット9 **TCIE**：転送完了（TC）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC割込みは無効です。

1：TC割込みは有効です。

ビット8 **TEIE**：転送エラー（TE）割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE割込みは無効です。

1：TE割込みは有効です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LOM**：ラインオフセットモード

このビットは、フォアグラウンド、バックグラウンドおよび出力のラインオフセットの表現方法（ピクセルまたはバイト）を設定します。

このビットは、ソフトウェアによってセット／クリアされます。転送中は変更できません。

0：ラインオフセットはピクセル単位で表現されます。

1：ラインオフセットはバイト単位で表現されます。

ビット5:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ABORT**：強制終了

このビットを使用して現在の転送を中止できます。このビットはソフトウェアによってセットされ、START = 0の場合ににハードウェアによって自動的にリセットされます。

0：転送強制終了はリクエストされていません。

1：転送強制終了がリクエストされています。

ビット1 **SUSP**：一時停止

このビットを使用して現在の転送を一時停止できます。このビットは、ソフトウェアによってセット／リセットされます。このビットはSTART = 0の場合にハードウェアによって自動的にリセットされます。

0：転送は一時停止しません。

1：転送は一時停止します。

ビット0 **START**：開始

このビットを使用し、各種設定レジスタにロードしたパラメータに従ってDMA2Dを開始することができます。このビットは以下のイベントにより自動的にリセットされます。

- 転送終了時
- ユーザがこのレジスタでABORTをセットすることにより、データ転送を強制終了したとき
- データ転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作を行っているためデータ転送が開始されないとき（自動CLUTロード）。

20.5.2 DMA2D割込みステータスレジスタ（DMA2D_ISR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
										r	r	r	r	r	r

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CEIF**：設定エラー割込みフラグ

このビットはDMA2D_CR、DMA2DFGPFCCR、またはDMA2D_BGPFCCRでSTARTがセットされ、誤った設定がプログラムされているときにセットされます。

ビット4 **CTCIF**：CLUT転送完了割込みフラグ

このビットは、システムメモリ領域から内部DMA2Dメモリへの CLUTコピーが完了したときセットされます。

ビット3 **CAEIF**：CLUTアクセスエラー割込みフラグ

このビットは、CLUTがシステムメモリから内部DMA2Dへ自動的にコピーされている間にCPUがCLUTにアクセスした場合にセットされます。

ビット2 **TWIF**：転送ウォーターマーク割込みフラグ

このビットはウォーターマークを付けたラインの最後のピクセルが転送されたときセットされます。

ビット1 **TCIF**：転送完了割込みフラグ

このビットはDMA2D転送動作が完了したときセットされます（データ転送のみ）。

ビット0 **TEIF**：転送エラー割込みフラグ

このビットはDMA転送中（データ転送または自動CLUTロード）にエラーが発生したときセットされます。

20.5.3 DMA2D割込みフラグクリアレジスタ（DMA2D_IFCR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CCEIF**：設定エラー割込みフラグのクリア

このビットを1にプログラムするとDMA2D_ISRのCEIFフラグがクリアされます。

ビット4 **CCTCIF**：CLUT転送完了割込みフラグのクリア

このビットを1にプログラムするとDMA2D_ISRのCTCIFフラグがクリアされます。

ビット3 **CAECIF**：CLUTアクセスエラー割込みフラグのクリア
このビットを1にプログラムするとDMA2D_ISRのCAEIFフラグがクリアされます。

ビット2 **CTWIF**：転送ウォーターマーク割込みフラグのクリア
このビットを1にプログラムするとDMA2D_ISRのTWIFフラグがクリアされます。

ビット1 **CTCIF**：転送完了割込みフラグのクリア
このビットを1にプログラムするとDMA2D_ISRのTCIFフラグがクリアされます。

ビット0 **CTEIF**：転送エラー割込みフラグのクリア
このビットを1にプログラムするとDMA2D_ISRのTEIFフラグがクリアされます。

20.5.4 DMA2Dフォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス、つまりフォアグラウンドイメージに使用されるデータのアドレスです。
アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32bitピクセルフォーマットは32bit境界に整列する必要があり、16bitピクセルフォーマットは16bit境界に整列する必要があり、4bitピクセルフォーマットは8bit境界に整列する必要があります。

20.5.5 DMA2Dフォアグラウンドオフセットレジスタ（DMA2D_FGOR）

アドレスオフセット：0x010

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

このフィールドには、以下で表現されるフォアグラウンドイメージに使用するラインオフセットが格納されます。

– DMA2D_CRでLOM = 0の場合、ピクセル単位。LO[13:0]ビットのみ考慮され、LO[15:14]ビットは無視されます。

– LOM = 1の場合、バイト単位。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

イメージフォーマットが4bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

20.5.6 DMA2Dバックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）

アドレスオフセット：0x014

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス、つまりバックグラウンドイメージに使用されるデータのアドレスです。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32bitピクセルフォーマットは32bit境界に整列する必要があり、16bitピクセルフォーマットは16bit境界に整列する必要があり、4bitピクセルフォーマットは8bit境界に整列する必要があります。

20.5.7 DMA2Dバックグラウンドオフセットレジスタ（DMA2D_BGOR）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

このフィールドには、以下で表現されるバックグラウンドイメージに使用するラインオフセットが格納されます。

– DMA2D_CRでLOM = 0の場合、ピクセル単位。LO[13:0]ビットのみ考慮され、LO[15:14]ビットは無視されます。

– LOM = 1の場合、バイト単位。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

イメージフォーマットが4bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

20.5.8 DMA2Dフォアグラウンド PFC制御レジスタ（DMA2D_FGPFCCR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、またはこのレジスタのAM[1:0]によって選択されたアルファモードに従って、元のアルファ値を乗算することができます。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤/青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するために赤と青をスワップできます。

0：通常モード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。

0：通常のアルファ値

1：アルファ反転

ビット19:18 **CSS[1:0]**：クロマサブサンプリング

これらのビットによって、YCbCrカラーモードのクロマサブサンプリングモードを定義します。

00：4:4:4（クロマサブサンプリングなし）

01：4:2:2

10：4:2:0

その他：予約済み

ビット17:16 **AM[1:0]**：アルファモード

これらのビットによって、フォアグラウンドイメージに使用するアルファチャネル値を選択します。

00：フォアグラウンドイメージのアルファチャネル値は修正されません。

01：元のフォアグラウンドイメージのアルファチャネル値をALPHA[7:0]によって置き換えます。

10：元のフォアグラウンドイメージのアルファチャネル値を、ALPHA[7:0]に元のアルファチャネル値を掛けた値で置き換えます。

その他：予約済み

ビット15:8 **CS[7:0]**：CLUTサイズ

これらのビットによって、フォアグラウンドイメージに使用するCLUTのサイズを定義します。

CLUTエントリの数はCS[7:0] + 1に等しい値となります。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **START**：開始

このビットをセットしてCLUTの自動ロードを開始できます。リセットは次のときに自動的に行われます。

- 転送終了時
- DMA2D_CRのABORTをセットすることにより、ユーザにより転送が強制終了されたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット4 **CCM**：CLUTカラーモード

このビットによって、CLUTのカラーフォーマットを定義します。

0：ARGB8888

1：RGB888

ビット3:0 **CM[3:0]**：カラーモード

これらのビットによって、フォアグラウンドイメージのカラーフォーマットを定義します。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

1011：YCbCr

その他：予約済み

20.5.9 DMA2Dフォアグラウンドカラーレジスタ（DMA2D_FGCOLR）

アドレスオフセット：0x020

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **RED[7:0]**：フォアグラウンドイメージのA4またはA8モードに対する赤の値。
ブレンドを伴うメモリ間転送での固定カラーFGおよび固定カラーFG（BGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

ビット15:8 **GREEN[7:0]**：フォアグラウンドイメージのA4またはA8モードに対する緑の値。
ブレンドを伴うメモリ間転送での固定カラーFGおよび固定カラーFG（BGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

ビット7:0 **BLUE[7:0]**：フォアグラウンドイメージのA4またはA8モードに対する青の値。
ブレンドを伴うメモリ間転送での固定カラーFGおよび固定カラーFG（BGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

20.5.10 DMA2Dバックグラウンドカラーレジスタ（DMA2D_BGPFCCR）

アドレスオフセット：0x024

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ値
これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、またはAM[1:0]によって選択されたアルファモードに従って元のアルファ値を乗算することができます。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤/青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するために赤と青をスワップできます。

0：通常モード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。

0：通常のアルファ値

1：アルファ反転

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **AM[1:0]**：アルファモード

これらのビットによって、バックグラウンドイメージに使用するアルファチャネル値を定義します。

00：フォアグラウンドイメージのアルファチャネル値は変更されません。

01：元のバックグラウンドイメージのアルファチャネル値をALPHA[7:0]によって置き換えます。

10:元のバックグラウンドイメージのアルファチャネル値をALPHA[7:0]に元のアルファチャネル値を掛けた値で置き換えます。

その他：予約済み

ビット15:8 **CS[7:0]**：CLUTサイズ

これらのビットによって、BGに使用するCLUTのサイズを定義します。

CLUTエントリの数はCS[7:0] + 1に等しい値となります。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **START**：開始

このビットをセットしてCLUTの自動ロードを開始します。このビットは以下のときに自動的にリセットされます。

- 転送終了時
- DMA2D_CRのABORTビットをセットすることにより、ユーザにより転送が強制終了されたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット4 **CCM**：CLUTカラーモード

これらのビットによって、CLUTのカラーフォーマットを定義します。

0：ARGB8888

1：RGB888

ビット3:0 **CM[3:0]**：カラーモード

これらのビットによって、バックグラウンドイメージのカラーフォーマットを定義します。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

その他：予約済み

20.5.11 DMA2Dバックグラウンドカラーレジスタ（DMA2D_BGCOLR）

アドレスオフセット：0x028

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **RED[7:0]**：バックグラウンドのA4またはA8モードに対する赤の値。
ブレンドを伴うメモリ間転送での固定カラー BGと固定BG（FGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

ビット15:8 **GREEN[7:0]**：バックグラウンドのA4またはA8モードに対する緑の値。
ブレンドを伴うメモリ間転送での固定カラー BGと固定BG（FGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

ビット7:0 **BLUE[7:0]**：バックグラウンドのA4またはA8モードに対する青の値。
ブレンドを伴うメモリ間転送での固定カラー BGと固定BG（FGフェッチはFGおよびBG PFCが有効の時のみ）に使用されます。

20.5.12 DMA2DフォアグラウンドCLUTメモリアドレスレジスタ（DMA2D_FGCMAR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されると、このレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス
フォアグラウンドイメージ専用のCLUTアドレスに使用されるデータのアドレスです。
フォアグラウンドCLUTフォーマットが32bitの場合、アドレスは32bit境界に整列する必要があります。

20.5.13 DMA2Dバックグラウンド CLUTメモリアドレスレジスタ（DMA2D_BGCMAR）

アドレスオフセット：0x030

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

バックグラウンドイメージ専用のCLUTアドレスに使用されるデータのアドレスです。

バックグラウンドCLUTフォーマットが32bitの場合、アドレスは32bitに調整する必要があります。

20.5.14 DMA2D出力PFC制御レジスタ（DMA2D_OPFCCR）

アドレスオフセット：0x034

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.
										rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
							rw						rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **RBS**：赤/青スワップ

このビットを使って、BGRまたはABGRのカラーフォーマットに対応するために赤と青をスワップできます。

0：通常モード（RGBまたはARGB）

1：スワップモード（BGRまたはABGR）

ビット20 **AI**：アルファ値反転

このビットはアルファ値を反転します。

0：レギュラアルファ

1：アルファ反転

ビット19:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SB**：バイトのスワップ

このビットがセットされると、出力FIFOのバイトが2 x 2でスワップします。ラインごとのピクセル数（PL）が偶数、出力メモリアドレス（OMAR）が偶数である必要があります。

0：バイトは出力FIFOでレギュラ順序になります。

1：バイトは出力FIFOで2 x 2でスワップされます。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **CM[2:0]**：カラーモード

これらのビットによって、出力イメージのカラーフォーマットを定義します。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

その他：予約済み

20.5.15 DMA2D出力カラーレジスタ（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB8888またはRGB888カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：ARGB8888モードの出力カラーのアルファチャネル値（別途予約済み）

ビット23:16 **RED[7:0]**：ARGB8888またはRGB888モードでの出力イメージの赤の値

ビット15:8 **GREEN[7:0]**：ARGB8888またはRGB888での出力イメージの緑の値

ビット7:0 **BLUE[7:0]**：ARGB8888またはRGB888での出力イメージの青の値

20.5.16 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

RGB565カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RED[4:0]					GREEN[5:0]						BLUE[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:11 **RED[4:0]**：RGB565モードでの出力イメージの赤の値

ビット10:5 **GREEN[5:0]**：RGB565モードでの出力イメージの緑の値

ビット4:0 **BLUE[4:0]**：RGB565モードでの出力イメージの青の値

20.5.17 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB1555カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	RED[4:0]					GREEN[4:0]					BLUE[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **A**：ARGB1555モードでの出力カラーのアルファチャネル値

ビット14:10 **RED[4:0]**：ARGB1555モードでの出力イメージの赤の値

ビット9:5 **GREEN[4:0]**：ARGB1555モードでの出力イメージの緑の値

ビット4:0 **BLUE[4:0]**：ARGB1555モードでの出力イメージの青の値

20.5.18 DMA2D出力カラーレジスタ[オルタネート]（DMA2D_OCOLR）

アドレスオフセット：0x038

リセット値：0x0000 0000

同じレジスタがカラー値を示すために使用されますが、カラーモードによってフォーマットは異なります。

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

ARGB4444カラーモード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **ALPHA[3:0]**：ARGB4444での出力カラー値のアルファチャネル値

ビット11:8 **RED[3:0]**：ARGB4444モードでの出力イメージの赤の値

ビット7:4 **GREEN[3:0]**：ARGB4444モードでの出力イメージの緑の値

ビット3:0 **BLUE[3:0]**：ARGB4444モードでの出力イメージの青の値

20.5.19 DMA2D出力メモリアドレスレジスタ（DMA2D_OMAR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MA[31:0]**：メモリアドレス

出力FIFOに使用されるデータのアドレスです。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32bitピクセルフォーマットは32bit境界に整列する必要があり、16bitピクセルフォーマットは16bit境界に整列する必要があります。

20.5.20 DMA2D出力オフセットレジスタ（DMA2D_OOR）

アドレスオフセット：0x040

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LO[15:0]**：ラインオフセット

このフィールドは以下の出力単位のラインオフセットを提供します。

– DMA2D_CRでLOM = 0の場合、ピクセル単位。LO[13:0]ビットのみ考慮され、LO[15:14]ビットは無視されます。

– LOM = 1の場合、バイト単位。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

20.5.21 DMA2Dライン数レジスタ（DMA2D_NLR）

アドレスオフセット：0x044

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PL[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:16 **PL[13:0]**：転送する領域のラインあたりのピクセル数

入力イメージフォーマットがピクセルあたり4bitの場合、ラインあたりのピクセル数は偶数である必要があります。

ビット15:0 **NL[15:0]**：転送する領域のライン数

20.5.22 DMA2Dラインウォーターマークレジスタ（DMA2D_LWR）

アドレスオフセット：0x048

リセット値：0x0000 0000

このレジスタには、転送が無効のときにのみ書込みできます。CLUT転送が開始されるとこのレジスタは読出し専用になります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LW[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LW[15:0]**：割込み生成のためのラインウォーターマーク
ウォーターマークを付けたラインの最後のピクセルが転送されたとき割込みが発生します。

20.5.23 DMA2DAHBマスタタイマ設定レジスタ（DMA2D_AMTCR）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **DT[7:0]**：デッドタイム
AHBマスタポート上の2つの連続アクセス間に挿入されるAHBクロックサイクルのデッドタイム値です。これらのビットは2つの連続AHBアクセス間の最小保証サイクル数を表します。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：デッドタイム機能の有効化

20.5.24 DMA2Dフォアグラウンド CLUT（DMA2D_FGCLUTx）

アドレスオフセット：0x400 + 0x4 * x（x = 0から255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ
フォアグラウンドのインデックス{x}のアルファ値

ビット23:16 **RED[7:0]**：赤
フォアグラウンドのインデックス{x}の赤の値。

ビット15:8 **GREEN[7:0]**：緑
フォアグラウンドのインデックス{x}の緑の値。

ビット7:0 **BLUE[7:0]**：青
フォアグラウンドのインデックス{x}の青の値。

20.5.25 DMA2Dバックグラウンド CLUT（DMA2D_BGCLUTx）

アドレスオフセット：0x800 + 0x4 * x（x = 0から255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ALPHA[7:0]**：アルファ
バックグラウンドのインデックス{x}のアルファ値

ビット23:16 **RED[7:0]**：赤
バックグラウンドのインデックス{x}の赤の値

ビット15:8 **GREEN[7:0]**：緑
バックグラウンドのインデックス{x}の緑の値

ビット7:0 **BLUE[7:0]**：青
バックグラウンドのインデックス{x}の青の値

20.5.26 DMA2Dレジスタマップ

表180.DMA2Dレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMA2D_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]			Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
	リセット値														0	0	0			0	0	0	0	0	0		0				0	0	0
0x004	DMA2D_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
	リセット値																											0	0	0	0	0	0
0x008	DMA2D_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
	リセット値																											0	0	0	0	0	0
0x00C	DMA2D_FGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	DMA2D_FGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	DMA2D_BGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	DMA2D_BGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	DMA2D_FGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x020	DMA2D_FGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	DMA2D_BGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x028	DMA2D_BGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	DMA2D_FGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	DMA2D_BGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	DMA2D_OPFCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
	リセット値											0	0												0						0	0	0
0x038	DMA2D_OCOLR ARGB8888またはRGB888カラーモード	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA2D_OCOLR RGB565カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[4:0]					GREEN[6:0]							BLUE[4:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA2D_OCOLR ARGB1555カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	A	RED[4:0]					GREEN[4:0]						BLUE[4:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表180.DMA2Dレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x038	DMA2D_OCOLR ARGB4444 カラーモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	DMA2D_OMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	DMA2D_OOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	DMA2D_NLR	Res.	Res.	PL[13:0]														NL[15:0]															
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	DMA2D_LWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LW[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	DMA2D_AMTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																	0	0	0	0	0	0	0	0								0
0x050 ~ 0x3FC	予約済み	予約済み																															
0x400 + 0x4 * x （x = 0 ~ 255）	DMA2D_FGCLUTx	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x800 + 0x4 * x （x = 0 ~ 255）	DMA2D_BGCLUTx	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、セクション2.3 を参照してください。

21 Chrom-GRC（GFXMMU）

このセクションは、STM32U599/5A9およびSTM32U5Fx/5Gxデバイスにのみ適用されます。

21.1 はじめに

グラフィックMMUは、以下を目的としたグラフィカル指向のメモリ管理ユニットです。

- ディスプレイ形状に応じたメモリの使用の最適化
- フレームバッファへのリニアアクセスのキャッシュ
- データのプリフェッチ

21.2 GFXMMUの主な機能

- 見えるピクセルのみを物理的に保管するための完全にプログラム可能なディスプレイ形状
- 最大4つの仮想バッファ
- 各仮想バッファのバイト数はラインごとに3072バイトまたは4096バイトで、1024本のラインを持つ
- 各仮想バッファは任意のシステムメモリに物理的に配置可能
- リニアアクセス用のオプションのキャッシュ
- キャッシュは仮想バッファにロック可能
- リニアアクセスを想定したプリフェッチ機構のキャッシュ
- バッファオーバーフロー時の割込み（バッファごとに1つ）
- メモリ転送エラー時の割込み

21.3 GFXMMUの実装

表181.GFXMMUの実装

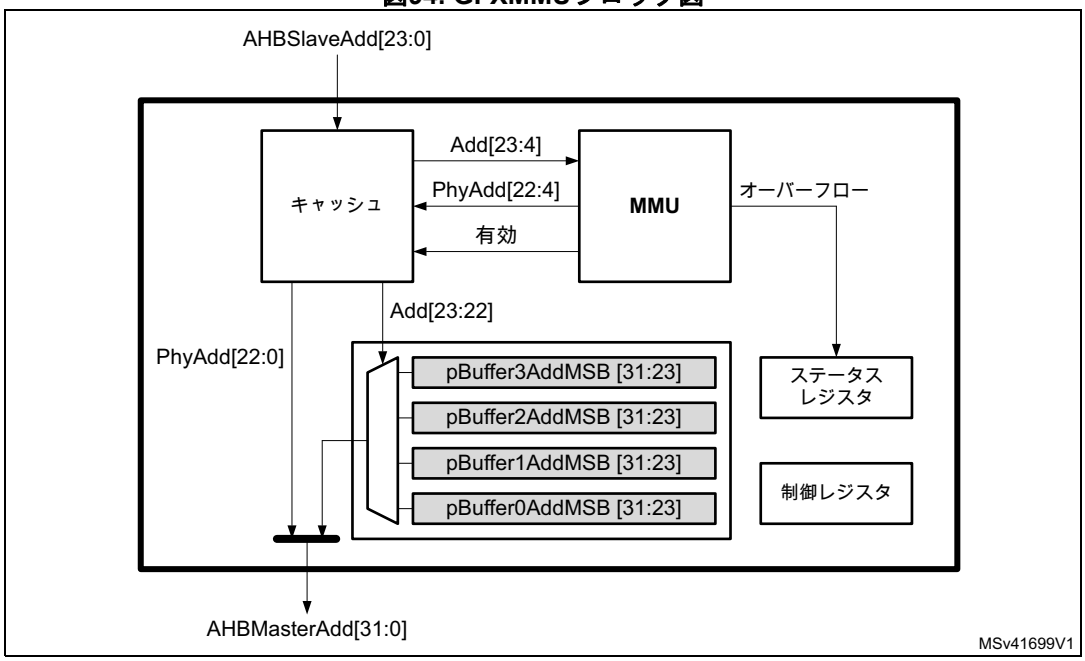
GFXMMUの機能	STM32U599/5A9	STM32U5Fx/5Gx
キャッシュとプリフェッチ機構	X	-
アドレスのキャッシュ	-	X

21.4 GFXMMUの機能およびアーキテクチャに関する説明

GFXMMUは次の2つのメインブロックで構成されます。

- アドレス解決用のMMU
- データ／アドレスのキャッシュ、プリフェッチ、およびパッキング／アンパッキング用のキャッシュ

図94. GFXMMUブロック図



21.4.1 仮想メモリ

GFXMMUは、システムマスタに表示される仮想メモリ空間を提供します[(a)]。仮想メモリ空間は、4つの仮想バッファに分けられています。

仮想バッファ

仮想バッファは、すべてのシステムマスタに対して、1024ラインの仮想フレームバッファを表す連続したメモリ空間として表示されます。
ラインはそれぞれ192個または256個の16バイトブロックに分けられ、これはGFXMMU設定レジスタ（GFXMMU_CR）の192BMビットに応じて異なります。

ディスプレイ形状とサイズによって、必要なブロックのみが物理メモリ位置に配置されます。この配置は、ラインごとにLUT入力をプログラミングして実行します。

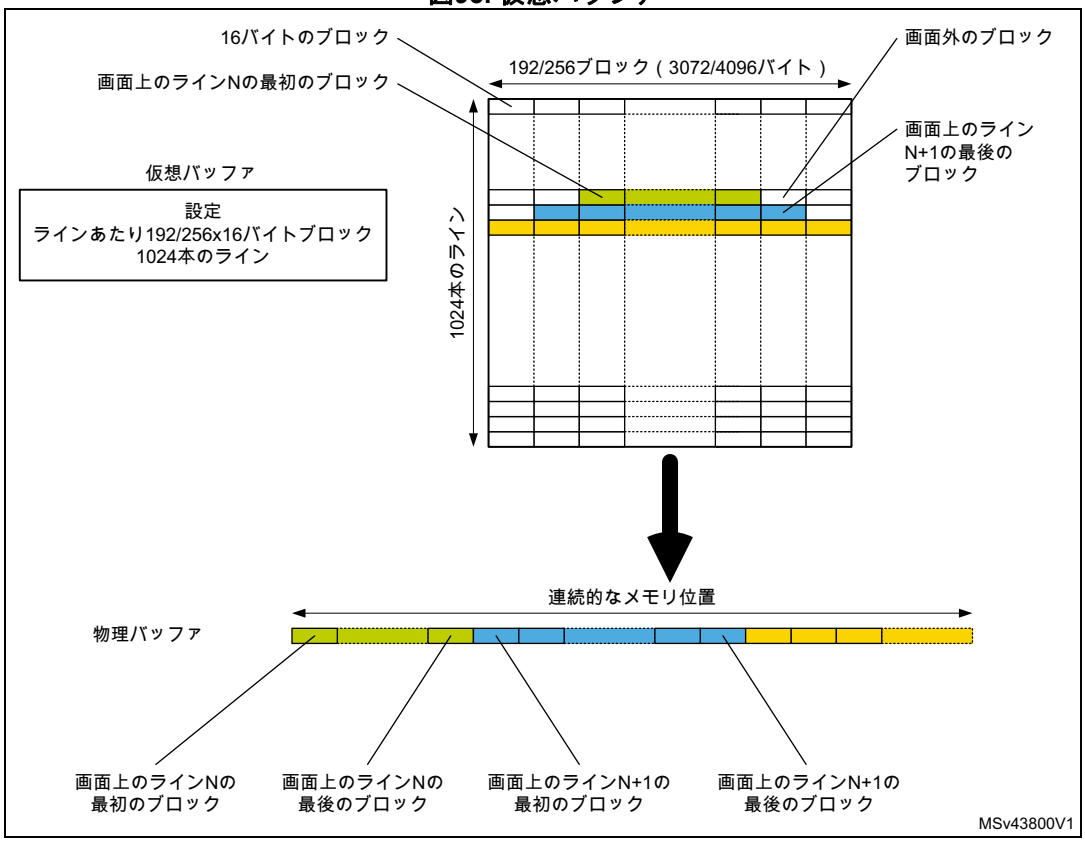
- ラインの有効化
- 最初に「見える」ブロックの数
- 最後に「見える」ブロックの数
- 物理バッファ内のラインのアドレスオフセット

a. GFXMMUにアクセスするシステムマスタについては、セクション「システム·アーキテクチャ」を参照してください。

「見える」ブロックは、各ラインのアドレスオフセットをプログラミングすることで、物理バッファ内に連続的に配置することができます。

LUTはすべてのバッファに共通です。つまり、バッファの「形状」はすべて同じです。

図95. 仮想バッファ



仮想バッファの概要

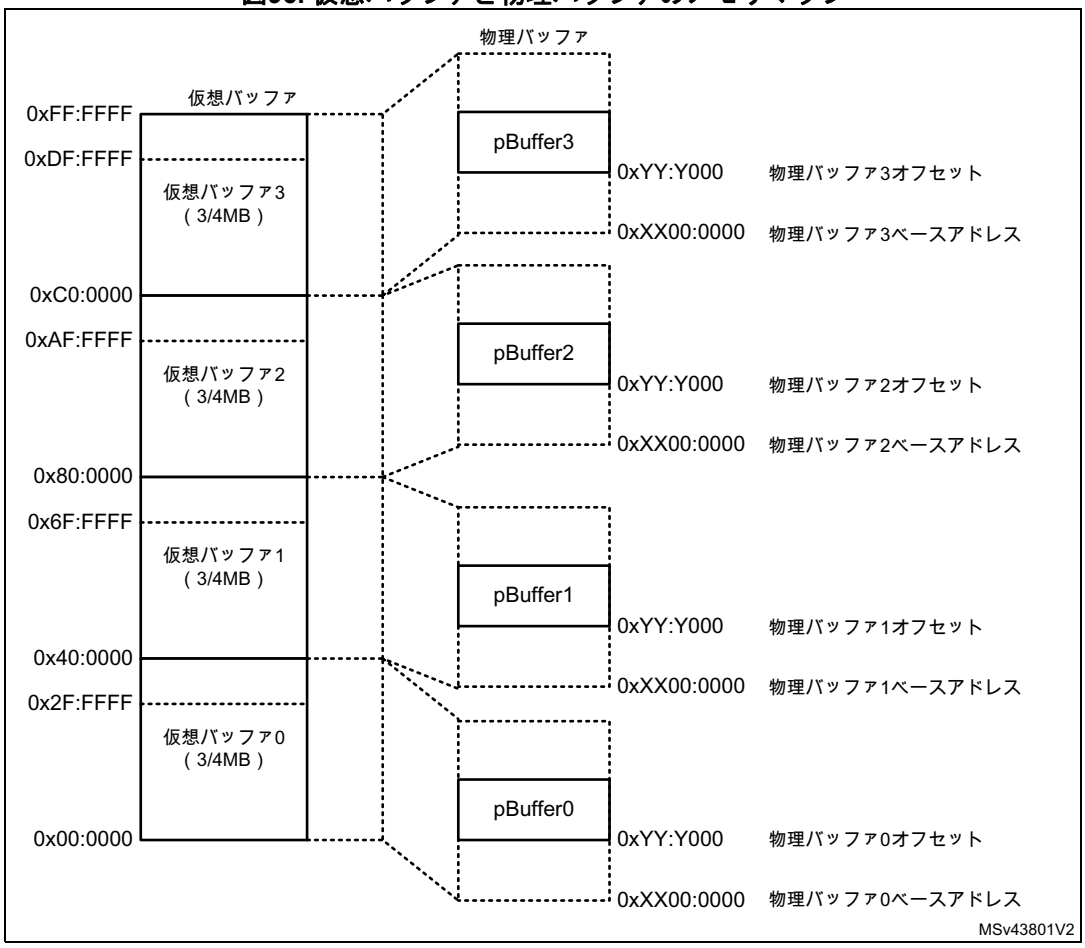
32bppまたは16bppでコード化されたフレームバッファの場合、仮想バッファには192個または256個のブロックを設定できます。これにより、仮想フレームバッファは32bppの場合は768x1024ピクセルまたは1024x1024ピクセル、16bppの場合は1536x1024ピクセルまたは2048x1024ピクセルになります。

24bppでコード化されたフレームバッファの場合、ラインごとのピクセル数を整数にするために、仮想バッファには192個のブロックを設定する必要があります。これにより、仮想フレームバッファは24bppの場合は1024x1024ピクセルになります。

次の理由により、各バッファは物理メモリ内の任意の場所に物理的に配置することができます。

- GFXMMUバッファの物理バッファベースアドレス（PBBA）フィールドx設定レジスタ（GFXMMU_BxCR）。これにより、バッファが配置された8MB領域の物理位置が設定されます。
- 物理バッファベースアドレスに対応する物理バッファの位置は、GFXMMUバッファの物理バッファオフセット（PBO）フィールドx設定レジスタ（GFXMMU_BxCR）によって定義されます。

図96. 仮想バッファと物理バッファのメモリマップ



バッファはそのベースアドレスで定義されたゾーンの8MBの境界をオーバーフローすることはできません。オーバーフローする場合、GFXMMU設定レジスタ（GFXMMU_CR）のバッファ xオーバーフロー割込みイネーブル（BxOIE）ビットがセットされている場合は、GFXMMUステータスレジスタ（GFXMMU_SR）のバッファ xオーバーフローフラグ（BxOF）がセットされ、割込みが生成されます。

仮想バッファアプリケーションの使用事例

物理位置は個別に設定できるため、4つの仮想バッファは連続しない場所にも物理的に配置することができます。これにより、たとえば4つのバッファを4つの異なるSDRAMバンクに配置することで、SDRAMアクセス時に追加のプリチャージサイクルが発生するのを避けることができます。

つまり、1つのバッファをLTDCで使用する場合、もう1つのバッファはCPU/ChromARTによってフレームバッファの計算に使用される必要があります。

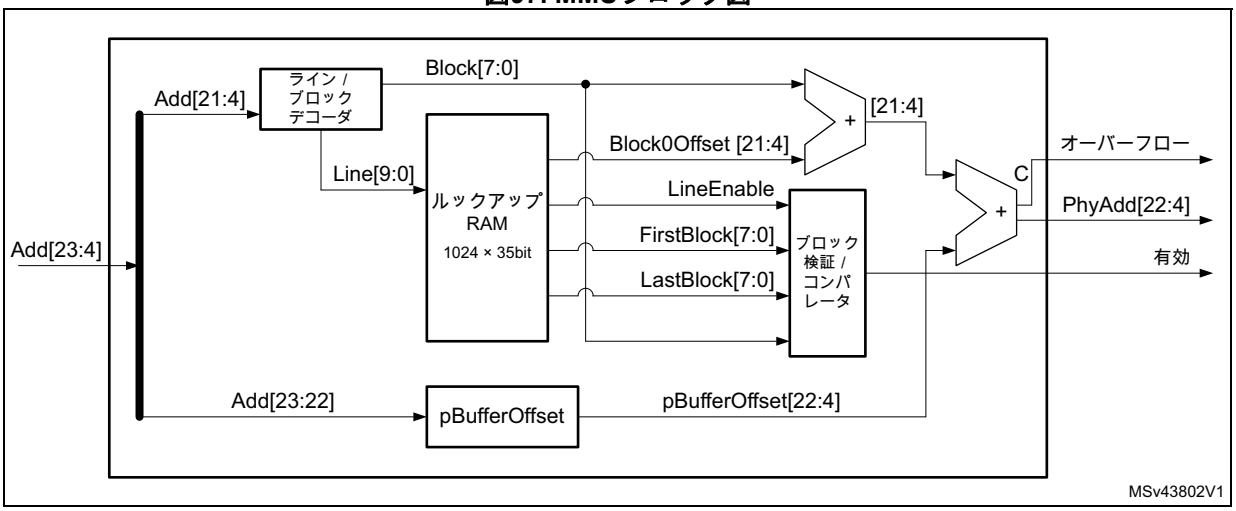
残りの2つのバッファは、追加の描画バッファを必要とするグラフィカルライブラリとして使用できます。

21.4.2 MMUアーキテクチャ

MMUブロックはアドレス解決用のブロックです。24bitのアドレスを受信して23bitの物理アドレスと、そのアドレスが物理的に配置されているかどうかを示す有効信号を返します。MMUでは、領域の境界のオーバーフローも確認します。

MMUのLUTは、1024x35bitのRAMとして実装されています。

図97. MMUブロック図



ライン･ブロック･デコーダ

ライン･ブロック･デコーダでは、ブロック番号とライン番号をアドレスに従って生成します。

ルックアップRAM

ルックアップRAMは、次のフィールドを持つ1024 x 35bit RAMです。

- 1bitのラインイネーブル
- 8bitの最初の有効なブロック
- 8bitの最後の有効なブロック
- ラインオフセット用の18bit

RAMはワードよりも大きいため、各エントリはメモリマップ上で2つのワードに分割されます。書込みアクセスは次の2つのステップで行われます。

1. GFXMMU_LUTxL メモリ位置の有効なブロック／最初の有効なブロック／最後の有効なブロックの最初のワードを書き込みます（内部バッファ付き）。
2. GFXMMU_LUTxH メモリ位置のラインオフセットの 2 番目のワードを書き込みます（内部バッファ付きの値によるメモリへの有効な書込み）。

LUTへの書込みはいつでも発生する可能性がありますが、同時にマスタがMMUを使用している場合は矛盾が生じる可能性があります。LUTのプログラミング中はCPUが優先されるため、MMUの計算速度が低下することがあります。

読出し操作中に制限はありませんが、LUTアクセスではMMUが優先されるため、CPUの速度が低下することがあります。

ブロックの検証／コンパレータ

このブロックの検証では、ブロックが有効であるかどうかを確認します。

ブロックは次の場合に有効（物理的に配置されている）とみなされます。

- ラインが有効である場合。
- ブロック数が最初の有効なブロック数以上である場合。
- ブロック数が最後の有効なブロック数以下である場合。

ブロックが有効な場合、生成される物理アドレスは正確であるとみなされます。

GFXMMU評価の結果が有効でない場合は、書込み操作が無視され、読出し操作ではGFXMMUデフォルト値レジスタ（GFXMMU_DVR）のデフォルト値（DV）フィールドに格納された、デフォルトの32bit値が返されます。

バッファ内のブロック･オフセット･アドレスの計算

ブロック数は、物理バッファ内のブロックのオフセットを取得するために、ラインオフセットに追加されます。

つまり、LUTに格納されたラインオフセットは、次の計算式によって与えられます。

ラインオフセット = [（すでに使用されている、見えるブロックの数）- （最初に見えるブロック）] x ブロックサイズ

ここで：

- ラインオフセットの最大値は､すべてのラインのすべてのブロックが使用されているときの値です。つまり、最後のラインのラインオフセットが最大値：1023 x 256 x 16 = 0x3F:F00xである可能性があります。
- ラインオフセットの最小値は､最初のラインの最後のブロックが最初の有効なブロックであるときの値：-255*16 = - 0xFFxすなわち0x3F:F01xです。

つまり、LUTのラインオフセット入力の全範囲が使用されます。

キャリーは、負のオフセット計算を実行できるよう、この段階では考慮されません（0x3F:F01xから0x3F:FFFxの値）。

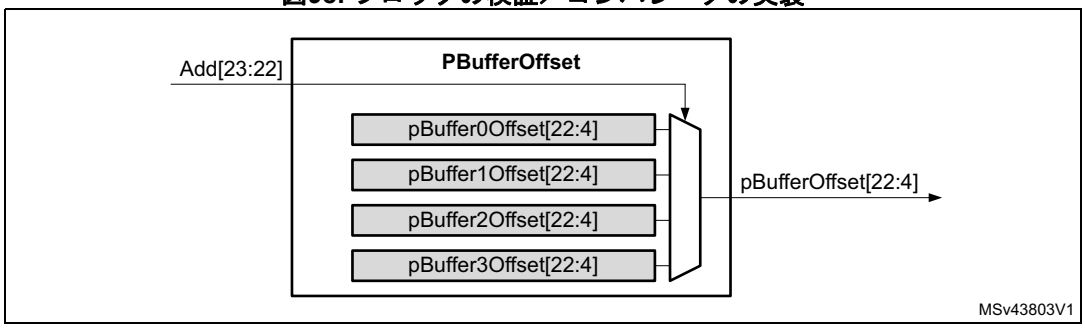
ブロックオフセットが4MBバッファ内にあるため、生成されるアドレスは22bit幅になります。

ブロック･オフセット･アドレスの計算

バッファ内のブロックオフセットが計算されると、この値は物理バッファベースアドレスに対応するブロックのオフセットに追加されます。

ブロックのオフセットは、図98に示すようにレジスタに定義されます。

図98. ブロックの検証／コンパレータの実装



結果のアドレスとバッファのオフセットアドレスは23bitになる必要があります。

キャリーは、アドレスのオーバーフローをトリガするために考慮されます。キャリーは、バッファ x オーバーフローフラグ（BxOF）をセットするためにGFXMMUステータスレジスタ（GFXMMU_SR）に伝播されます。

計算例

ここでは、仮想バッファ 0の次の設定を検証します。

- ライン0で最初に見えるブロック：ブロック7
- ライン0で見えるブロックの数：10
- ライン1で最初に見えるブロック：ブロック6
- ライン1で見えるブロックの数：12
- 物理バッファのアドレス：0xC020:0000

次のように設定される必要があります。

- 物理バッファ 0のベースアドレス：0xC000:0000
- バッファ 0のオフセット:0x20:0000
- ライン0で最初に見えるブロック：ブロック7
- ライン0で最後に見えるブロック：ブロック16
- ライン0のブロック0のオフセット：(0 - 7) x 0x10 = -0x70 = 0x3F:FF90
- ライン1で最初に見えるブロック：ブロック6
- ライン1で最後に見えるブロック：ブロック17
- ライン1のブロック0のオフセット：(10 - 6) x 0x10 = (0xA - 0x6) x 0x10 = 0x40

その結果、以下のようになります。

- ライン0のブロック7の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x3F:FF90 + 0x70、キャリーなし) = 0xC020:0000

- ライン0のブロック16の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x3F:FF90 + 0x100、キャリーなし) = 0xC020:0090

- ライン1のブロック6の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x40 + 0x60、キャリーなし) = 0xC020:00A0

- ライン1のブロック17の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x40 + 0x110、キャリーなし) = 0xC020:0150

21.4.3 キャッシュとプリフェッチ機構

グラフィック･フレーム･バッファを格納するGFXMMUの統合されたキャッシュ･ターゲットである内部または外部RAMデバイス。

GFXMMUにアクセスするマスタ

いくつかのマスタがGFXMMUを介してメモリにアクセスすると想定されています。

- CPU がフレームバッファにアクセスするのは、単一のピクセルの読出し／変更／書込みをフレームバッファで行っているためです。つまり、R/M/Wの最小単位はワード以下で、データキャッシュが必要です。フレームバッファは線形にスキャンされるため、現実的にはCPUが要求するバッファ内の次のデータが次のピクセルであると言えます。プリフェッチ機構では、これを効率的に予想できます。

- ChromARTまたはLTDCがフレームバッファにアクセスする際に、最大128バイトのロング･アクセスが生成されます。この場合、続けて1回の操作しか行われないため、キャッシュは不要になる場合があります。

そのため、キャッシュはCPU向けであり、DMA2DやLTDCと一緒に使用してはなりません。キャッシュはCPUでのみ使用し、コヒーレンシはソフトウェアで管理し、CPUの動作が終了したときにキャッシュをフラッシュすることを推奨します。

STM32U599/5A9では、フレームバッファがRGB888モードで設定されている場合、GPUでキャッシュを使用することも推奨されます。

キャッシュの有効化

キャッシュは、GFXMMU制御レジスタ（GFXMMU_CR）のCE（キャッシュ有効化）ビットをセットして有効化します。

キャッシュのロック機構

GFXMMU設定レジスタ（GFXMMU_CR）のキャッシュ･ロック･ビット（CL）をセットし、キャッシュ･ロック･バッファ（CLB）フィールドを設定することで、キャッシュをロックすることができます。

キャッシュが所定のバッファ上でロックされると、その他のバッファはキャッシュできなくなります。

ロックされたバッファの操作は、マスタの要求属性に従ってキャッシュできます。また、GFXMMU設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットをセットすることで、いつでもキャッシュできるようになります。強制キャッシュ機構は、キャッシュがバッファにロックされている場合にのみ使用できます。GFXMMU設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットは、GFXMMU設定レジスタ（GFXMMU_CR）のキャッシュ･ロック（CL）ビットがリセットされると、自動的にリセットされます。

キャッシュラインサイズ

キャッシュラインサイズは16バイトです。

キャッシュライン数

ライン数は、CPUがリニアアクセスを行うことを考慮して、可能な限り削減されます。

そのため、3本のラインが必要になります。

- 現在のアクセス用の1ライン
- 前回アクセス時の1ライン（ピクセルを2つの16バイトのブロックに分割できるため）
- プリフェッチされたアクセス用の1ライン

プリフェッチ機構

CPUは、ほとんどの場合フレームバッファにリニアアクセスするため、メモリから次のキャッシュラインを自動的に取得するためにプリフェッチ機構が用意されています。

プリフェッチ機構は、GFXMMU設定レジスタ（GFXMMU_CR）のプリフェッチ無効化（PD）ビットをセットすることで無効化できます。

プリフェッチ機構を無効化すると、プリフェッチ専用のラインのTAGだけは更新されますが、データはメモリから回収されます。

キャッシュのメンテナンス操作

キャッシュが非強制キャッシュモードで動作している場合、つまり GFXMMU 設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットがリセットされている場合、次の場合にラインエビクションが実行されます。

- キャッシュ格納不可アクセスが、データがキャッシュされたバッファ上で実行されるたび
- ミスが発生するたび

このため、キャッシュのメンテナンス操作は必然的に自動で行うことができます。

ただし、GFXMMUキャッシュ制御レジスタ（GFXMMU_CCR）のフラッシュの強制（FF）ビットがキャッシュにセットされている場合は、フラッシュを強制することができます。キャッシュをフラッシュする場合、ダーティなエントリはすべて書込みバッファに送信され、TAGのダーティ·ビットはすべてリセットされます。このとき、エントリは無効にはなりません。

GFXMMUキャッシュ制御レジスタ（GFXMMU_CCR）の無効化の強制（FI）ビットをセットすることで、キャッシュ·エントリを無効化することもできます。これにより、ダーティなエントリが書込みバッファに送信されることはありません（キャッシュにおける変更は失われます）。TAGのIDフィールドがリセット（未使用状態）されるだけです。

GFXMMUキャッシュ制御レジスタ（GFXMMU_CCR）のフラッシュの強制（FF）ビットと無効化の強制（FI）ビットの2つをセットすることで、次の操作が実行されます。

- キャッシュがフラッシュされ、ダーティなエントリがすべて書込みバッファに送信されます。
- TAGのダーティ·ビットをリセットします。
- TAGのIDフィールドをリセットします。

書込みバッファですでにFlashのトランザクションが行われているときに、この3つの操作のいずれかが実行されると、書込みバッファでは動作を継続します。

すべての動作が終了すると（書込みバッファエンプティ）、フラッシュの強制（FF）と無効化の強制（FI）は自動的にリセットされ、同期バリアが作成されます。

21.4.4 アドレスのキャッシュ

評価プロセスによる待ち時間を短縮するために、特定のキャッシュをアクティブにして最新のMMUルックアップ結果を保持できます。このアドレスのキャッシュは、選択された仮想バッファ上でのみアクティブになります。

このルックアップのキャッシュは、GFXMMU制御レジスタ（GFXMMU_CR）のアドレス·キャッシュ有効化（ACE）ビットでアクティブになります。

キャッシュされた仮想バッファは、GFXMMU制御レジスタ（GFXMMU_CR）のアドレス·キャッシュ·ロック·バッファ（ACLB）ビットフィールドで選択されます。

21.5 GFXMMU割込み

割込みは次のイベントによって生成できます。

- バッファ 0オーバーフロー
- バッファ 1オーバーフロー
- バッファ 2オーバーフロー
- バッファ 3オーバーフロー
- AHBマスタエラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表182.GFXMMU割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット
バッファ 0オーバーフロー	B0OF	B0OIE
バッファ 1オーバーフロー	B1OF	B1OIE
バッファ 2オーバーフロー	B2OF	B2OIE
バッファ 3オーバーフロー	B3OF	B3OIE
AHBマスタエラー	AMEF	AMEIE

21.6 GFXMMUレジスタ

21.6.1 GFXMMU設定レジスタ（GFXMMU_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACLB[1:0]		ACE	Res.	Res.	OB	OC
									rw	rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PD	FC	CLB[1:0]		CL	CE	192BM	Res.	AMEIE	B3OIE	B2OIE	B1OIE	B0OIE
			rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22:21 **ACLB[1:0]**：アドレス･キャッシュ･ロック･バッファ

このフィールドは、アドレス･キャッシュがアクティブな仮想バッファを選択します。

00：バッファ 0にアドレス･キャッシュをロックします。

01：バッファ 1にアドレス･キャッシュをロックします。

10：バッファ 2にアドレス･キャッシュをロックします。

11：バッファ 3にアドレス･キャッシュをロックします。

ビット20 **ACE**：アドレス･キャッシュ有効化

このビットは、GFXMMUアドレス･キャッシュを有効にします。

0：アドレス･キャッシュは無効です。

1：アドレス･キャッシュは有効です。

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **OB**：外部バッファ可能性

このビットは、GFXMMUキャッシュによって生成されるアクセスのバッファ可能性を設定します。

0：バッファ可能性なし

1：バッファ可能性あり

ビット16 **OC**：外部キャッシュ可能性

このビットは、GFXMMUキャッシュによって生成されるアクセスのキャッシュ可能性を設定します。

0：キャッシュ可能性なし

1：キャッシュ可能性あり

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **PD**：プリフェッチ無効化

このビットは、キャッシュのプリフェッチを無効にします。

0：プリフェッチは有効です。

1：プリフェッチは無効です。

ビット11 **FC**：強制キャッシュ

このビットは、MPU属性に関係なく、キャッシュへのキャッシュを強制します。

キャッシュは有効である必要があります（CEビットをセット）。

0：キャッシュを強制しない

1：キャッシュを強制する

ビット10:9 **CLB[1:0]**：キャッシュ･ロック･バッファ

このフィールドは、キャッシュがロックされたバッファを選択します。

00：バッファ 0にキャッシュをロックします。

01：バッファ 1にキャッシュをロックします。

10：バッファ 2にキャッシュをロックします。

11：バッファ 3にキャッシュをロックします。

ビット8 **CL**：キャッシュ･ロック

このビットは、CLBフィールドで定義されたバッファにキャッシュをロックします。

0：キャッシュはロックされません。

1：バッファにキャッシュをロックします。

ビット7 **CE**：キャッシュ有効化

このビットは、キャッシュ･ユニットを有効にします。

0：キャッシュは無効です。

1：キャッシュは有効です。

ビット6 **192BM**：192ブロックモード

このビットは、ラインあたりのブロック数を定義します

0：ラインあたり256ブロック

1：ラインあたり192ブロック

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **AMEIE**：AHBマスタエラー割込みイネーブル

このビットは、AHBマスタエラー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット3 **B3OIE**：バッファ 3オーバーフロー割込みイネーブル

このビットは、バッファ 3オーバーフロー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット2 **B2OIE**：バッファ 2オーバーフロー割込みイネーブル

このビットは、バッファ 2オーバーフロー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット1 **B1OIE**：バッファ 1オーバーフロー割込みイネーブル

このビットは、バッファ 1オーバーフロー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット0 **B0OIE**：バッファ 0オーバーフロー割込みイネーブル

このビットは、バッファ 0オーバーフロー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

21.6.2 GFXMMUのステータスレジスタ（GFXMMU_SR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AMEF	B3OF	B2OF	B1OF	B0OF
											r	r	r	r	r

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **AMEF**：AHBマスタエラーフラグ

このビットは、トランザクション中にAHBエラーが発生したときにセットされます。CAMEFに1を書き込むことによってクリアされます。

ビット3 **B3OF**：バッファ 3オーバーフローフラグ

このビットは、バッファ 3のオフセット計算中にオーバーフローが発生したときにセットされます。CB3OFに1を書き込むことによってクリアされます。

ビット2 **B2OF**：バッファ 2オーバーフローフラグ

このビットは、バッファ 2のオフセット計算中にオーバーフローが発生したときにセットされます。CB2OFに1を書き込むことによってクリアされます。

ビット1 **B1OF**：バッファ 1オーバーフローフラグ

このビットは、バッファ 1のオフセット計算中にオーバーフローが発生したときにセットされます。CB1OFに1を書き込むことによってクリアされます。

ビット0 **B0OF**：バッファ 0オーバーフローフラグ

このビットは、バッファ 0のオフセット計算中にオーバーフローが発生したときにセットされます。CB0OFに1を書き込むことによってクリアされます。

21.6.3 GFXMMUフラグクリアレジスタ（GFXMMU_FCR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CAMEF	CB3OF	CB2OF	CB1OF	CB0OF
											rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CAMEF**：AHBマスタエラーフラグのクリア

このビットに1を書き込むと、GFXMMU_SRレジスタのAHBマスタエラーフラグがクリアされます。

ビット3 **CB3OF**：バッファ 3オーバーフローフラグのクリア

このビットに1を書き込むと、GFXMMU_SRレジスタのバッファ 3オーバーフローフラグがクリアされます。

ビット2 **CB2OF**：バッファ 2オーバーフローフラグのクリア

このビットに1を書き込むと、GFXMMU_SRレジスタのバッファ 2オーバーフローフラグがクリアされます。

ビット1 **CB1OF**：バッファ 1オーバーフローフラグのクリア

このビットに1を書き込むと、GFXMMU_SRレジスタのバッファ 1オーバーフローフラグがクリアされます。

ビット0 **CB0OF**：バッファ 0オーバーフローフラグのクリア

このビットに1を書き込むと、GFXMMU_SRレジスタのバッファ 0オーバーフローフラグがクリアされます。

21.6.4 GFXMMUキャッシュ制御レジスタ（GFXMMU_CCR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FI	FF
														rs	rs

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **FI**：無効化の強制

セットすると、キャッシュ･エントリが無効化されます。このビットは、無効化が完了したときに、ハードウェアによってリセットされます。0を書き込んでも、ビットの値は変化しません。

0：無効化処理完了

1：無効化／進行中の無効化処理の強制

ビット0 **FF**：フラッシュの強制

セットすると、キャッシュ･エントリがフラッシュされます。このビットは、フラッシュが完了したときに、ハードウェアによってリセットされます。0を書き込んでも、ビットの値は変化しません。

0：フラッシュ処理完了

1：フラッシュ／進行中のフラッシュ処理の強制

21.6.5 GFXMMUデフォルト値レジスタ（GFXMMU_DVR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DV[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DV[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DV[31:0]**：デフォルトの値

このフィールドは、マスタが物理的に配置されていない仮想メモリの位置にアクセスしたときに返される、デフォルトの32bit値を示します。

21.6.6 GFXMMUバッファ 0設定レジスタ（GFXMMU_B0CR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PBBA[31:23]									PBO[22:16]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PBO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:23 **PBBA[31:23]**：物理バッファベースアドレス

物理バッファのベースアドレスのMSB。

ビット22:4 **PBO[22:4]**：物理バッファオフセット

物理バッファのオフセット。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

21.6.7 GFXMMUバッファ 1設定レジスタ（GFXMMU_B1CR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PBBA[31:23]									PBO[22:16]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PBO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスのMSB。

ビット22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

21.6.8 GFXMMUバッファ 2設定レジスタ（GFXMMU_B2CR）

アドレスオフセット：0x0028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PBBA[31:23]									PBO[22:16]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PBO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスのMSB。

ビット22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

21.6.9 GFXMMUバッファ 3設定レジスタ（GFXMMU_B3CR）

アドレスオフセット：0x002C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PBBA[31:23]									PBO[22:16]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PBO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスのMSB。

ビット22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

21.6.10 GFXMMULUT入力x、ロー（GFXMMU_LUTxL）

アドレスオフセット：0x1000 + 0x8 * x（x = 0から1023）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **LVB[7:0]**：最後の有効なブロック
ライン番号Xの最後の有効なブロックの番号。

ビット15:8 **FVB[7:0]**：最初の有効なブロック
ライン番号xの最初の有効なブロックの番号。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：有効化
ラインの有効化。
0：ラインを無効化（MMU評価の実施なし）
1：ラインを有効化（MMU評価の実施あり）

21.6.11 GFXMMU LUT入力x、ハイ（GFXMMU_LUTxH）

アドレスオフセット：0x1004 + 0x8 * x（x = 0から1023）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:16]					
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21:4 **LO[21:4]**：ラインオフセット

ライン番号xのラインオフセット（つまり、ラインxのブロック0のオフセット）

ビット3:0 予約済みであり、リセット値に保持する必要があります。

21.6.12 GFXMMUレジスタマップ

次の表にグラフィックMMUレジスタの一覧を示します。グラフィックMMUレジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表183.GFXMMUレジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	GFXMMU_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACLB [1:0]		ACE	Res.	Res.	OB	OC	Res.	Res.	Res.	PD	FC	CLB [1:0]		CL	CE	192BM	Res.	AMEIE	B3OIE	B2OIE	B1OIE	B0OIE
	リセット値										0	0	0			0	0				0	0	0	0	0	0	0		0	0	0	0	0
0x0004	GFXMMU_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AMEF	B3OF	B2OF	B1OF	B0OF
	リセット値																												0	0	0	0	0
0x0008	GFXMMU_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CAMEF	CB3OF	CB2OF	CB1OF	CB0OF
	リセット値																												0	0	0	0	0
0x000C	GFXMMU_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FI	FF
	リセット値																															0	0
0x0010	GFXMMU_DVR	DV[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014-0x001C	予約済み	予約済み																															
0x0020	GFXMMU_B0CR	PBBA[31:23]									PBO[22:4]																		Res.	Res.	Res.	Res.	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0024	GFXMMU_B1CR	PBBA[31:23]									PBO[22:4]																		Res.	Res.	Res.	Res.	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0028	GFXMMU_B2CR	PBBA[31:23]									PBO[22:4]																		Res.	Res.	Res.	Res.	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x002C	GFXMMU_B3CR	PBBA[31:23]									PBO[22:4]																		Res.	Res.	Res.	Res.	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0030 ~ 0x0FFC	予約済み	予約済み																															
0x1000	GFXMMU_LUT0L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]								FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0
0x1004	GFXMMU_LUT0H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:4]																		Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
...																																	
0x2FF8	GFXMMU_LUT1023L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]								FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0
0x2FFC	GFXMMU_LUT1023H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:4]																		Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

22 ネスト化されたベクタ割込みコントローラ（NVIC）

22.1 NVICの主な機能

- 最大140本のマスク可能な割込みチャネル（FPUを搭載したCortex-M33の16本の割込みラインは数に含まれていない）
- 16のプログラム可能な優先レベル（4bitの割込み優先順位を使用）
- 遅延時間の少ない例外および割込み処理
- 電源管理制御
- システム制御レジスタの実装

NVICとプロセッサ·コアのインタフェースを密に結合することで、割込み処理の遅延を低減し、後着割込みを効率的に処理することができます。

NVICレジスタは、セキュアステートと非セキュアステートにまたがってバンク分けされています。

コア例外を含むすべての割込みは、NVICによって管理されます。

22.2 SysTick較正値レジスタ

TrustZoneメインライン·セキュリティ拡張機能を備えたCortex-M33には、2つのSysTickタイマが組み込まれています。

TrustZoneがアクティブなときは、次に示すSysTickタイマが使用可能です。

- SysTickセキュアインスタンス
- SysTick非セキュアインスタンス

TrustZoneが無効なときは、1つのSysTickタイマだけが使用可能です。

SysTickタイマの較正値（STCALIB）は0x3E8です。1MHzのSysTickクロック周波数に基づいて、1msの基準タイムベースが得られます。特定の周波数で動作しているアプリケーションの1msタイムベースを一致させるには、SYSTick再ロード値をSYST_RVRレジスタに次のようにプログラムする必要があります。

- SysTickクロックソースがCPUクロックHCLKの場合、再ロード値 = （HCLK x STCALIB）- 1
- SysTickクロックソースが外部クロック（HCLK/8）の場合、再ロード値 = （HCLK/8 x STCALIB）- 1

HCLKはMHz単位のAHB周波数値を表します。

例：SysTickクロックソースが100MHzのCPUクロックHCLKの場合、1msのタイムベースを一致させるには：

SysTick再ロード値 = （100 × STCALIB）- 1 = 0x1869F

22.3 割込みベクタと例外ベクタ

次の表の灰色の行は、特定の位置を持たないベクタを示しています。

表184.STM32U5シリーズベクタテーブル(1)

位置	優先順位	優先種別	項目（略称）	説明	アドレス
-	-	-	-	予約済み	0x0000 0000
-	-4	固定	リセット	リセット	0x0000 0004
-	-2	固定	NMI	ノンマスカブル割込み。RCCクロックセキュリティシステム（CSS）はNMIベクタにリンクされます。	0x0000 0008
-	-3 または -1	固定	Secure HardFault	Secure HardFault	0x0000：000C
-	-1	固定	Nonsecure HardFault	非セキュア･ハードフォールト、あらゆる種類の異常	0x0000：000C
-	0	設定可能	MemManage	メモリ管理	0x0000 0010
-	1	設定可能	BusFault	プリフェッチ異常、メモリアクセス異常	0x0000 0014
-	2	設定可能	UsageFault	未定義命令または無効状態	0x0000 0018
-	3	設定可能	SecureFault	セキュアフォールト	0x0000：001C
-	-	-	-	予約済み	0x0000 0020～ 0x0000 0028
-	4	-	SVC	SWI命令によるシステムサービスコール	0x0000：002C
-	5	-	デバッグモニタ	デバッグモニタ	0x0000 0030
-	-	-	-	予約済み	0x0000 0034
-	6	設定可能	PendSV	ペンディング可能なシステムサービスリクエスト	0x0000 0038
-	7	設定可能	SysTick	システムティックタイマ	0x0000：003C
0	8	設定可能	WWDG	ウィンドウ型ウォッチドッグ割込み	0x0000 0040
1	9	設定可能	PVD_PVM	プログラム可能な電圧検出器／ペリフェラル電圧モニタ	0x0000 0044
2	10	設定可能	RTC	RTCグローバル非セキュア割込み	0x0000 0048
3	11	設定可能	RTC_S	RTCセキュア･グローバル割込み	0x0000：004C
4	12	設定可能	TAMP	タンパ･グローバル割込み	0x0000 0050
5	13	設定可能	RAMCFG	RAM設定グローバル割込み	0x0000 0054
6	14	設定可能	FLASH	Flashメモリ非セキュア･グローバル割込み	0x0000 0058
7	15	設定可能	Flash_S	Flashメモリセキュアグローバル割込み	0x0000：005C
8	16	設定可能	GTZC	GTZC1/GTZC2グローバル割込み	0x0000 0060
9	17	設定可能	RCC	RCC非セキュアグローバル割込み	0x0000 0064
10	18	設定可能	RCC_S	RCCセキュアグローバル割込み	0x0000 0068
11	19	設定可能	EXTI0	EXTIライン0割込み	0x0000：006C
12	20	設定可能	EXTI1	EXTIライン1割込み	0x0000 0070
13	21	設定可能	EXTI2	EXTIライン2割込み	0x0000 0074
14	22	設定可能	EXTI3	EXTIライン3割込み	0x0000 0078
15	23	設定可能	EXTI4	EXTIライン4割込み	0x0000：007C

表184.STM32U5シリーズベクタテーブル(1)（続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
16	24	設定可能	EXTI5	EXTIライン5割込み	0x0000 0080
17	25	設定可能	EXTI6	EXTIライン6割込み	0x0000 0084
18	26	設定可能	EXTI7	EXTIライン7割込み	0x0000 0088
19	27	設定可能	EXTI8	EXTIライン8割込み	0x0000：008C
20	28	設定可能	EXTI9	EXTIライン9割込み	0x0000 0090
21	29	設定可能	EXTI10	EXTIライン10割込み	0x0000 0094
22	30	設定可能	EXTI11	EXTIライン11割込み	0x0000 0098
23	31	設定可能	EXTI12	EXTIライン12割込み	0x0000：009C
24	32	設定可能	EXTI13	EXTIライン13割込み	0x0000：00A0
25	33	設定可能	EXTI14	EXTIライン14割込み	0x0000：00A4
26	34	設定可能	EXTI15	EXTIライン15割込み	0x0000：00A8
27	35	設定可能	IWDG	独立型ウォッチドッグ割込み	0x0000：00AC
28	36	設定可能	SAES	セキュアAES	0x0000：00B0
29	37	設定可能	GPDMA1_CH0	GPDMA1チャネル0グローバル割込み	0x0000：00B4
30	38	設定可能	GPDMA1_CH1	GPDMA1チャネル1グローバル割込み	0x0000：00B8
31	39	設定可能	GPDMA1_CH2	GPDMA1チャネル2グローバル割込み	0x0000：00BC
32	40	設定可能	GPDMA1_CH3	GPDMA1チャネル3グローバル割込み	0x0000：00C0
33	41	設定可能	GPDMA1_CH4	GPDMA1チャネル4グローバル割込み	0x0000：00C4
34	42	設定可能	GPDMA1_CH5	GPDMA1チャネル5グローバル割込み	0x0000：00C8
35	43	設定可能	GPDMA1_CH6	GPDMA1チャネル6グローバル割込み	0x0000：00CC
36	44	設定可能	GPDMA1_CH7	GPDMA1チャネル7グローバル割込み	0x0000：00D0
37	45	設定可能	ADC12	ADC12（14bit）グローバル割込み	0x0000：00D4
38	46	設定可能	DAC1	DAC1グローバル割込み	0x0000：00D8
39	47	設定可能	FDCAN1_IT0	FDCAN1割込み0	0x0000：00DC
40	48	設定可能	FDCAN1_IT1	FDCAN1割込み1	0x0000：00E0
41	49	設定可能	TIM1_BRK TIM1_TERR TIM1_IERR	TIM1ブレーク TIM1遷移エラー TIM1インデックスエラー	0x0000：00E4
42	50	設定可能	TIM1_UP	TIM1更新	0x0000：00E8
43	51	設定可能	TIM1_TRG_COM TIM1_DIR TIM1_IDX	TIM1トリガおよび転流 TIM1方向変更割込み TIM1インデックス	0x0000：00EC
44	52	設定可能	TIM1_CC	TIM1キャプチャ／比較割込み	0x0000：00F0
45	53	設定可能	TIM2	TIM2グローバル割込み	0x0000：00F4
46	54	設定可能	TIM3	TIM3グローバル割込み	0x0000：00F8
47	55	設定可能	TIM4	TIM4グローバル割込み	0x0000：00FC
48	56	設定可能	TIM5	TIM5グローバル割込み	0x0000 0100
49	57	設定可能	TIM6	TIM6グローバル割込み	0x0000 0104

表184.STM32U5シリーズベクタテーブル(1) （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
50	58	設定可能	TIM7	TIM7グローバル割込み	0x0000 0108
51	59	設定可能	TIM8_BRK TIM8_TERR TIM8_IERR	TIM8ブレーク割込み TIM8遷移エラー TIM8インデックスエラー	0x0000：010C
52	60	設定可能	TIM8_UP	TIM8更新割込み	0x0000 0110
53	61	設定可能	TIM8_TRG_COM TIM8_DIR TIM8_IDX	TIM8トリガおよび転流割込み TIM8方向変更割込み TIM8インデックス	0x0000 0114
54	62	設定可能	TIM8_CC	TIM8キャプチャ／比較割込み	0x0000 0118
55	63	設定可能	I2C1_EV	I2C1イベント割込み	0x0000：011C
56	64	設定可能	I2C1_ER	I2C1エラー割込み	0x0000 0120
57	65	設定可能	I2C2_EV	I2C2イベント割込み	0x0000 0124
58	66	設定可能	I2C2_ER	I2C2エラー割込み	0x0000 0128
59	67	設定可能	SPI1	SPI1グローバル割込み	0x0000：012C
60	68	設定可能	SPI2	SPI2グローバル割込み	0x0000 0130
61	69	設定可能	USART1	USART1グローバル割込み	0x0000 0134
62	70	設定可能	USART2	USART2グローバル割込み	0x0000 0138
63	71	設定可能	USART3	USART3グローバル割込み	0x0000：013C
64	72	設定可能	UART4	UART4グローバル割込み	0x0000 0140
65	73	設定可能	UART5	UART5グローバル割込み	0x0000 0144
66	74	設定可能	LPUART1	LPUART1グローバル割込み	0x0000 0148
67	75	設定可能	LPTIM1	LPTIM1グローバル割込み	0x0000：014C
68	76	設定可能	LPTIM2	LPTIM2グローバル割込み	0x0000 0150
69	77	設定可能	TIM15	TIM15グローバル割込み	0x0000 0154
70	78	設定可能	TIM16	TIM16グローバル割込み	0x0000 0158
71	79	設定可能	TIM17	TIM16グローバル割込み	0x0000：015C
72	80	設定可能	COMP	COMP1/COMP2	0x0000 0160
73	81	設定可能	USB/OTG_FS/ OTG_HS	USB/OTG_FS/OTG_HSグローバル割込み	0x0000 0164
74	82	設定可能	CRS	クロックリカバリシステムグローバル割込み	0x0000 0168
75	83	設定可能	FMC	FSMCグローバル割込み	0x0000：016C
76	84	設定可能	OCTOSPI1	OCTOSPI1グローバル割込み	0x0000 0170
77	85	設定可能	PWR_S3WU	STOP3からのPWRウェイクアップ割込み	0x0000 0174
78	86	設定可能	SDMMC1	SDMMC1グローバル割込み	0x0000 0178
79	87	設定可能	SDMMC2	SDMMC2グローバル割込み	0x0000：017C
80	88	設定可能	GPDMA1_CH8	GPDMA1チャネル8割込み	0x0000 0180
81	89	設定可能	GPDMA1_CH9	GPDMA1チャネル9割込み	0x0000 0184
82	90	設定可能	GPDMA1_CH10	GPDMA1チャネル10割込み	0x0000 0188
83	91	設定可能	GPDMA1_CH11	GPDMA1チャネル11割込み	0x0000：018C

表184.STM32U5シリーズベクタテーブル(1) （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
84	92	設定可能	GPDMA1_CH12	GPDMA1チャネル12割込み	0x0000 0190
85	93	設定可能	GPDMA1_CH13	GPDMA1チャネル13割込み	0x0000 0194
86	94	設定可能	GPDMA1_CH14	GPDMA1チャネル14割込み	0x0000 0198
87	95	設定可能	GPDMA1_CH15	GPDMA1チャネル15割込み	0x0000：019C
88	96	設定可能	I2C3_EV	I2C3イベント割込み	0x0000：01A0
89	97	設定可能	I2C3_ER	I2C3エラー割込み	0x0000：01A4
90	98	設定可能	SAI1	SAI1グローバル割込み	0x0000：01A8
91	99	設定可能	SAI2	SAI2グローバル割込み	0x0000：01AC
92	100	設定可能	TSC	TSCグローバル割込み	0x0000：01B0
93	101	設定可能	AES	AESグローバル割込み	0x0000：01B4
94	102	設定可能	RNG	RNGグローバル割込み	0x0000：01B8
95	103	設定可能	FPU	浮動小数点割込み	0x0000：01BC
96	104	設定可能	HASH	HASH割込み	0x0000：01C0
97	105	設定可能	PKA	PKAグローバル割込み	0x0000：01C4
98	106	設定可能	LPTIM3	LPTIM3グローバル割込み	0x0000：01C8
99	107	設定可能	SPI3	SPI3グローバル割込み	0x0000：01CC
100	108	設定可能	I2C4_ER	I2C4エラー割込み	0x0000：01D0
101	109	設定可能	I2C4_EV	I2C4イベント割込み	0x0000：01D4
102	110	設定可能	MDF1_FLT0	MDF1フィルタ0グローバル割込み	0x0000：01D8
103	111	設定可能	MDF1_FLT1	MDF1フィルタ1グローバル割込み	0x0000：01DC
104	112	設定可能	MDF1_FLT2	MDF1フィルタ2グローバル割込み	0x0000：01E0
105	113	設定可能	MDF1_FLT3	MDF1フィルタ3グローバル割込み	0x0000：01E4
106	114	設定可能	UCPD1	UCPD1グローバル割込み	0x0000：01E8
107	115	設定可能	ICACHE	命令キャッシュグローバル割込み	0x0000：01EC
108	116	設定可能	OTFDEC1	OTFDEC1セキュアグローバル割込み	0x0000：01F0
109	117	設定可能	OTFDEC2	OTFDEC2セキュアグローバル割込み	0x0000：01F4
110	118	設定可能	LPTIM4	LPTIM4グローバル割込み	0x0000：01F8
111	119	設定可能	DCACHE1	データ･キャッシュ･グローバル割込み	0x0000：01FC
112	120	設定可能	ADF1_FLT0	ADF1フィルタ0グローバル割込み	0x0000 0200
113	121	設定可能	ADC4	ADC4（12bit）グローバル割込み	0x0000 0204
114	122	設定可能	LPDMA1_CH0	LPDMA1 SmartRunチャネル0グローバル割込み	0x0000 0208
115	123	設定可能	LPDMA1_CH1	LPDMA1 SmartRunチャネル1グローバル割込み	0x0000：020C
116	124	設定可能	LPDMA1_CH2	LPDMA1 SmartRunチャネル2グローバル割込み	0x0000 0210
117	125	設定可能	LPDMA1_CH3	LPDMA1 SmartRunチャネル3グローバル割込み	0x0000 0214
118	126	設定可能	DMA2D	DMA2Dグローバル割込み	0x0000 0218
119	127	設定可能	DCMI_PSSI	DCMI/PSSIグローバル割込み	0x0000：021C

表184.STM32U5シリーズベクタテーブル(1) （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
120	128	設定可能	OCTOSPI2	OCTOSPI2グローバル割込み	0x0000 0220
121	129	設定可能	MDF1_FLT4	MDF1フィルタ4グローバル割込み	0x0000 0224
122	130	設定可能	MDF1_FLT5	MDF1フィルタ5グローバル割込み	0x0000 0228
123	131	設定可能	CORDIC	CORDIC割込み	0x0000：022C
124	132	設定可能	FMAC	FMAC割込み	0x0000 0230
125	133	設定可能	LSECSS MSI_PLL_UNLOCK	LSECSS割込み(2) MSI PLLアンロック割込み (2)	0x0000 0234
126	134	設定可能	USART6	USART6グローバル割込み	0x0000 0238
127	135	設定可能	I2C5_ER	I2C5エラー割込み	0x0000：023C
128	136	設定可能	I2C5_EV	I2C5イベント割込み	0x0000 0240
129	137	設定可能	I2C6_ER	I2C6エラー割込み	0x0000 0244
130	138	設定可能	I2C6_EV	I2C6イベント割込み	0x0000 0248
131	139	設定可能	HSPI1	Hexadeca-SPI1グローバル割込み	0x0000：024C
132	140	設定可能	GPU2D_IRQ	GPU2D割込み	0x0000 0250
133	141	設定可能	GPU2D_IRQSYS	GPU2Dシステム割込み	0x0000 0254
134	142	設定可能	GFXMMU	GFXMMUグローバルエラー割込み	0x0000 0258
135	143	設定可能	LCD_TFT	LTDCグローバル割込み	0x0000：025C
136	144	設定可能	LCD_TFT_ERR	LTDCグローバルエラー割込み	0x0000 0260
137	145	設定可能	DSIHOST	DSIグローバル割込み	0x0000 0264
138	146	設定可能	DCACHE2	DCACHE2グローバル割込み	0x0000 0268
139	147	設定可能	GFXTIM	GFXTIMグローバル割込み	0x0000：026C
140	148	設定可能	JPEG	JPEG同期割込み	0x0000 0270

1. いくつかの割込みラインは、一部のSTM32U5シリーズのデバイスでのみ使用可能です。関連するペリフェラルの有無については、デバイスのデータシートを参照してください。無い場合は、この割込みラインは予約済みと考えてください。

2. STM32U575/585のリビジョンXデバイスでは予約済み。LSECSSおよびMSI_PLL_UNLOCK割込みラインは、STM32U575/585の他のすべてのリビジョン、および他のすべてのSTM32U5シリーズ·デバイスで利用可能です。

23 拡張割込み／イベントコントローラ（EXTI）

拡張割込み／イベントコントローラ（EXTI）は、設定可能なイベントの入力によって個々のCPUおよびシステムのウェイクアップを管理します。電源制御にウェイクアップリクエストを供給し、CPUのNVICに割込みリクエストを生成し、CPUイベント入力へのイベントを生成します。CPUイベント信号を生成するには、CPUは追加のイベント生成ブロック（EVG）が必要になります。

EXTIウェイクアップリクエストにより、システムはSTOPモードからウェイクアップできます。

RUNモードでも、割込みリクエストとイベントリクエストの生成を使用できます。

EXTIIにはEXTIマルチプレクサのIOポートの選択も含まれています。

23.1 EXTIの主な機能

EXTIの主な機能は以下のとおりです。

- 26の入力イベントをサポート
- すべてのイベント入力はシステムをウェイクアップする効果があります。
- ペリフェラルにウェイクアップフラグがないイベントは、EXTIIにフラグを持ち、EXTIからCPUへの割込みを生成します。
- CPUウェイクアップイベントを生成するためにイベントを使用できます。

設定可能なイベントには、以下の特徴があります。

- 選択可能なアクティブトリガエッジ
- 立上がりおよび立下がりエッジに依存しない割込みペンディング･ステータス･レジスタ･ビット
- CPU のウェイクアップ、割込み、およびイベント生成を調整するために使用される個々の割込みおよびイベント生成マスク
- ソフトウェア･トリガの可能
- セキュアイベント : セキュア入力イベントの制御および設定ビットへのアクセスは、セキュアおよび／または特権で行うことができます。
- EXTIのIOポートの選択

23.2 EXTIブロック図

図99に示すように、EXTIはAHBインタフェースを介してアクセスするレジスタブロック、イベント入力トリガブロック、マスクブロック、およびEXTIマルチプレクスから成ります。

レジスタブロックは、すべてのEXTIレジスタを含んでいます。

イベント入力トリガブロックは、イベント入力エッジのトリガロジックを供給します。

マスクブロックは、さまざまなウェイクアップ、割込み、およびイベント出力に、イベント入力とそれらのマスクを供給します。

EXTIマルチプレクサは、EXTIイベント信号がIOポートを選択できるようにします。

図99. EXTIブロック図

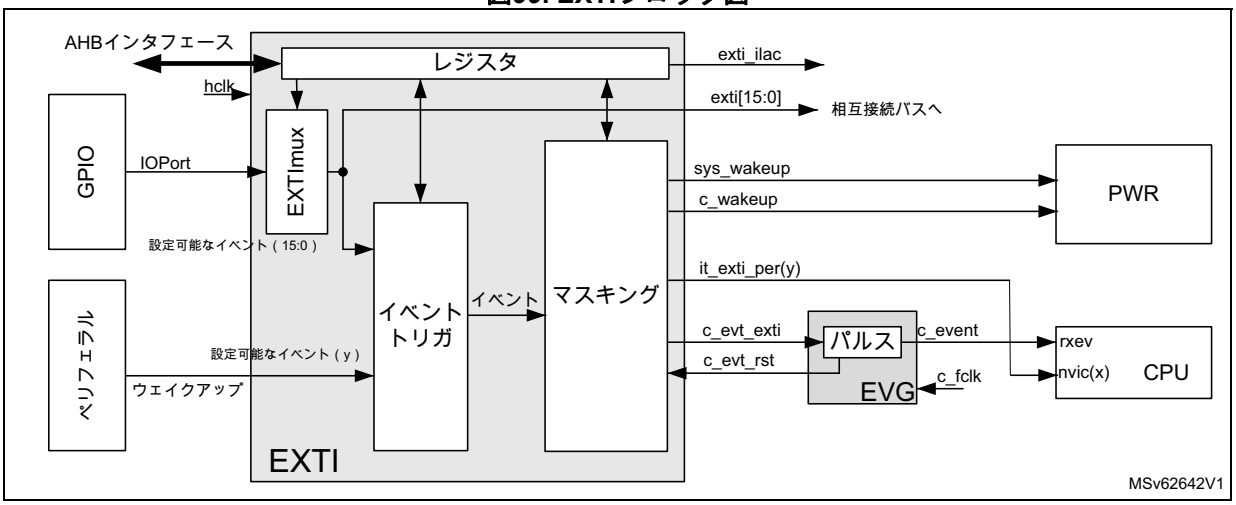


表185.EXTI信号

名前	I/O	説明
AHBインタフェース	I/O	EXTIレジスタのバスインタフェース。1つのイベントがセキュリティを有効にするように設定されている場合、AHBインタフェースはセキュアなアクセスをサポートします。
hclk	I	AHBバスクロックおよびEXTIシステムクロック
設定可能なイベント（y）	I	ペリフェラル割込みおよびフラグを持たないペリフェラルからの非同期ウェイクアップイベント
exti_ilac	O	不正アクセスイベント
IOPort（n）	I	GPIOのブロックIOポート[15:0]
exti[15:0]	O	他のペリフェラルをトリガするためのEXTI GPIO出力ポート
it_exti_per（y）	O	設定可能なイベント(y)に伴うCPUへの割込み
c_evt_exti	O	CPUへの、ハイレベルに反応するイベント出力、hclkと同期
c_evt_rst	I	c_evt_extiをクリアする非同期リセット入力
sys_wakeup	O	ck_sysおよびhclk用のPWRへの非同期システムウェイクアップリクエスト
c_wakeup	O	hclkに同期した、CPU用のPWRへのウェイクアップリクエスト

表186.EVG信号

名前	I/O	説明
c_fclk	I	CPUフリー･ランニング･クロック
c_evt_in	I	CPUクロックに非同期の、EXTIからのハイレベル対応イベント入力
c_event	O	CPUクロックに同期したイベントパルス
c_evt_rst	O	CPUクロックに同期したイベントリセット信号

23.2.1 ペリフェラルとCPU間のEXTI接続

システムのSTOPモード時にウェイクアップまたは割込みイベントを生成できるペリフェラルをEXTIに接続します。

- パルスを生成する、またはペリフェラルに割込みステータスビットがないペリフェラルのウェイクアップ信号は、EXTIの設定可能なイベント入力に接続されます。これらのイベントに、EXTIはクリアする必要があるステータスペンディングビットを供給します。CPUに割り込むのはステータス・ビットに関連したEXTI割込みです。
- すべてのGPIOポートはEXTIマルチプレクサに入力されるため、ポートピンを選択して設定可能なイベントによってシステムをウェイクアップすることができます。

EXTIの設定可能なイベント割込みは、NVICに接続されています。

専用のEXTI/EVG CPUイベントを、CPU rxev入力に接続します。

EXTIのCPUのウェイクアップ信号をPWRに接続し、システムおよびCPUサブシステム・バスのクロックをウェイクアップするために使用します。

23.2.2 EXTI割込み／イベントのマッピング

EXTIラインは下の表に示すように接続されています。

表187.EXTIラインの接続

EXTIのライン	ラインソース	ラインタイプ
0～15	GPIO	設定可能
16	PVD出力	設定可能
17	COMP1出力	設定可能
18(1)	COMP2出力	設定可能
19	V_{DDUSB}電圧モニタ	設定可能
20	V_{DDIO2}電圧モニタ	設定可能
21	V_{DDA}電圧モニタ1	設定可能
22	V_{DDA}電圧モニタ2	設定可能
23(2)	LSECSSまたはMSI_PLL_UNLOCK	設定可能
23(3)	MSI_PLL_UNLOCK	設定可能
24 (3)	LSECSS	設定可能
25 (3)	IWDG早期割込み	設定可能

1. STM32U535/545デバイスでは使用できません。
2. STM32U535/545/59x/5Ax/5Fx/5Gx デバイス、および STM32U575/585 のリビジョン X デバイスでは使用できません。STM32U575/585のその他のすべてのリビジョンで使用できます。
3. STM32U575/585デバイスを除くすべてのSTM32U5シリーズで使用できます。

23.3 EXTIの機能説明

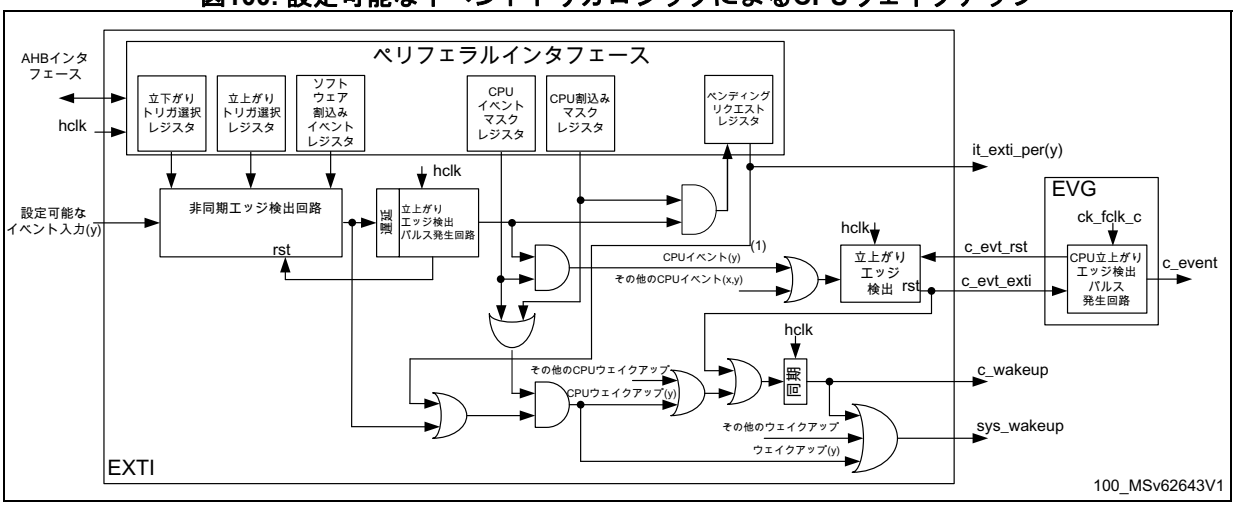
イベントの機能は、次のとおりレジスタ･ビットで制御されます。

- 以下によって有効になるアクティブなトリガエッジ
  - EXTI_RTSR1の立上がりエッジの選択
  - EXTI_FTSR1の立下がりエッジの選択
- EXTI_SWIER1のソフトウェアトリガ
- EXTI_RPR1およびEXTI_FPR1の割込み保留中フラグ
- EXTI_IMR1のCPUウェイクアップクアップおよび割込み有効化
- EXTI_IMR1のCPUウェイクアップクアップおよびイベント有効化

23.3.1 EXTIの設定可能なイベント入力によるウェイクアップ

下の図は、CPUサブシステム･バス･クロックをウェイクアップさせ、EXTIペンディング･フラグを生成し、またCPUやCPUウェイクアップ･イベントに割込みをかける、設定可能なイベント入力に関連したロジックの詳細を表しています。

図100. 設定可能なイベントトリガロジックによるCPUウェイクアップ



1. CPUのrxevを生成するc_eventをサポートしている入力イベントに対してのみ。

対応するレジスタ･ビットを書き込むことで、エッジ選択の設定に関係なく、ソフトウェア割込みイベント･レジスタにより、設定可能なイベントをソフトウェアでトリガできます。

設定可能なイベント･アクティブ･トリガ･エッジ（あるいは両方のエッジ）を選択して、立上がり／立下がりエッジ選択レジスタで有効にします。

CPUは専用のウェイクアップ（割込み）マスクレジスタおよび専用のイベントマスクレジスタを備えています。イベントが有効化されると、CPUにイベントが発生します。CPUに向けたすべてのイベントは論理和がとられて1つのCPUイベント信号に集約されます。CPUのイベント･ペンディング･レジスタ（EXTI_RPRおよびEXTI_FPR）は、マスクされていないCPUイベントにはセットされません。

設定可能なイベントには、固有の割込みペンディング･リクエスト･レジスタが割り当てられます。ペンディングレジスタは、マスクされていない割込みについてのみセットされます。設定可能な各イベントは、CPUに共通の割込みを提供します。設定可能なイベント割込みは、EXTI_RPRおよび／またはEXTI_FPRレジスタでソフトウェアにより確認する必要があります。

CPUウェイクアップ（割込み）またはCPUイベントが有効化されると、非同期エッジ検出回路は、クロック遅延および立上がりエッジ検出パルス発生回路によってリセットされます。これにより、EXTI hclkクロックは、非同期エッジ検出回路がリセットされる前にウェイクアップされることが保証されます。

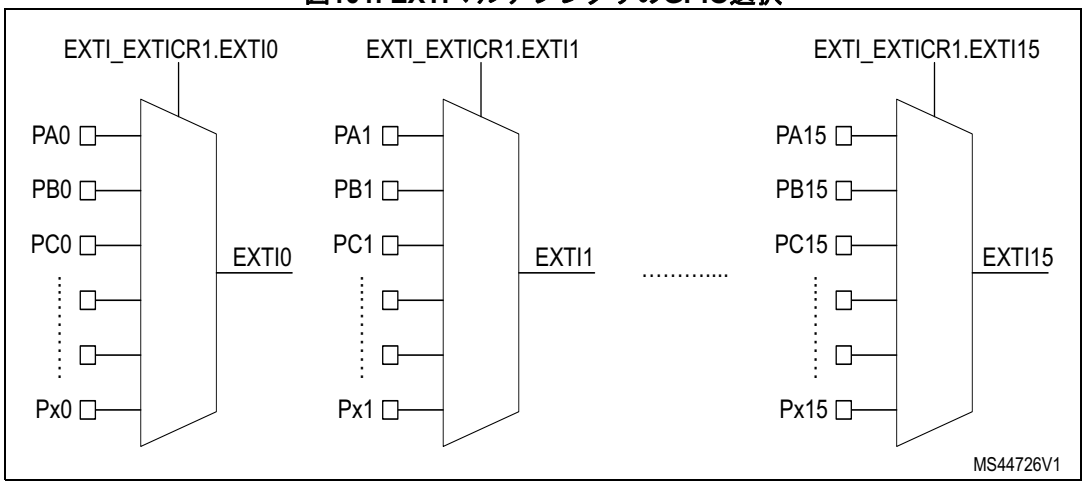
注： **検出された設定可能なイベント割込みペンディングリクエストは正しいアクセス許可を有する CPUによってクリアされることがあります。割込みペンディングリクエストがアクティブである限り、システムは低消費電力モードに入ることができません。**

23.3.2 EXTIマルチプレクサの選択

EXTIマルチプレクサにより、割込みおよびウェイクアップとしてGPIOの選択ができます。GPIOは、16EXTIマルチプレクサ·ラインを介し設定可能なイベントとして、最初の16EXTIイベントに接続されます。

EXTIマルチプレクサ出力としてのGPIOポートの選択は、EXTI_EXTICRmで制御されます。

図101. EXTIマルチプレクサのGPIO選択



EXTIマルチプレクサ出力は、EXTIからの出力信号として、EXTI_IMRやEXTI_EMRレジスタのマスクに関係なく、他のペリフェラルをトリガするために利用可能です。

23.4 EXTIの機能動作

設定可能なイベントは、少なくとも1つのトリガエッジを有効にすることで有効になります。

イベント入力が有効になると、CPUウェイクアップの生成はCPU割込みマスクおよびCPUイベント･マスクの条件により決定されます。

表188.マスクの機能

CPU割込み有効（EXTI_IMR.IMnにおいて）	CPUイベント有効（EXTI_EMR.EMnにおいて）	設定可能なイベント入力（EXTI_RPR.RPIFnおよびEXTI_FPR.FPIFnにおいて）	Exti（n）割込み(1)	CPUイベント	CPU ウェイクアップ
0	0	なし	マスク	マスク	マスク
	1	なし	マスク	あり	あり
1	0	ステータスラッチ	あり	マスク	あり(2)
	1	ステータスラッチ	あり	あり	あり

1. 単一のexti(n)割込みがCPUに送信されます。CPU(m)に割込みが必要ない場合、exti(n)割込みはCPU NVIC内でマスクしなければなりません。
2. CPU割込みがEXTI_IMR.IMnで有効化されている場合のみです。

設定可能なイベント入力の場合、イベント入力時にイネーブル･エッジが発生すると、イベント･リクエストが生成されます。関連するCPU割込みがマスクされていない場合、EXTI_RPRの対応するペンディングビットRPIFnおよび／またはEXTI_FPRの対応するペンディングビットFPIFnがセットされ、CPUサブシステムがウェイクアップされ、CPU割込み信号がアクティブになります。RPIFnおよび／またはFPIFnは、ソフトウェアで1を書き込むことによってクリアする必要があります。これによってCPU割込みもクリアされます。

設定可能なイベント入力の場合、ソフトウェア割込み／イベント･レジスタEXTI_SWIERに1を書き込むと（これによってイベント時に立上がりエッジが生成されます）、ソフトウェアによってイベント･リクエストを生成することができます。立上がりエッジ･イベント･ペンディング･ビットは、EXTI_RTSRの設定にかかわりなく、EXTI_RPRにセットされます。

23.5 EXTIイベント保護

EXTIは、イベントレジスタビットが非セキュアおよび非特権のアクセスによって変更されないように保護することができます。保護は、EXTI_SECCFGRおよびEXTI_PRIVCFGRのレジスタビットによって、入力イベントごとに個別にアクティブにされます。EXTIレベルでは、保護は以下の不正な書込みアクセスを防ぐことにあります。

- セキュアおよび／または特権の設定可能なイベントの設定の変更。
- セキュアおよび／または特権の入力イベントのマスクの変更。
- セキュアおよび／または特権の入力イベントのペンディング･ステータスのクリア。

表189.レジスタ保護の概要

レジスタ名	アクセスタイプ	保護[1][2]
EXTI_RTSR	RW	セキュリティおよび特権は、EXTI_SECCFGRおよびEXTI_PRIVCFGRでビット単位に有効にできます。
EXTI_FTSR	RW	
EXTI_SWIER	RW	
EXTI_RPR	RW	
EXTI_FPR	RW	
EXTI_SECCFGR	RW	常にセキュア。特権はEXTI_PRIVCFGRでビット単位で有効にできます。
EXTI_PRIVCFGR	RW	常に特権。セキュリティはEXTI_SECCFGRでビット単位で有効にできます。
EXTI_EXTICRn	RW	セキュリティおよび特権は、EXTI_SECCFGRおよびEXTI_PRIVCFGRでビット単位に有効にできます。
EXTI_LOCKR	RW	常にセキュア。
EXTI_IM	RW	セキュリティおよび特権は、EXTI_SECCFGRおよびEXTI_PRIVCFGRでビット単位に有効にできます。
EXTI_EMR	RW	

1. セキュリティは個々の入力イベントで有効になります（EXTI_SECCFGRレジスタ）。
2. 特権は個々の入力イベントで有効になります（EXTI_PRIVCFGRレジスタ）。

23.5.1 EXTIセキュリティ保護

入力イベントに対してセキュリティが有効になっている場合、関連する入力イベントの設定ビットおよび制御ビットはセキュア･アクセスによってのみ変更および読出しが可能です。非セキュア書込みアクセスは破棄され、読出しには0が返されます。

入力イベントが非セキュアな場合、セキュリティは無効になります。関連する入力イベントの設定ビットおよび制御ビットはセキュアアクセスおよび非セキュアアクセスによって変更および読出しができます。

レジスタEXTI_SECCFGRのセキュリティ設定は、EXTI_LOCKRのLOCKによってリセットした後にグローバルにロックすることができます。

23.5.2 EXTI特権保護

入力イベントに対して特権が有効になっている場合、関連する入力イベントの設定ビットおよび制御ビットは特権アクセスによってのみ変更および読出しが可能です。非特権書込みアクセスは破棄され、読出しアクセスには0が返されます。

入力イベントが非特権の場合、特権は無効になります。関連する入力イベントの設定ビットおよび制御ビットは特権アクセスおよび非特権アクセスによって変更および読出しができます。

EXTI_PRIVCFGRレジスタの特権設定は、EXTI_LOCKRのLOCKによってリセットした後にグローバルにロックすることができます。

23.6 EXTIレジスタ

EXTIレジスタマップは、以下のセクションに分けられています。

表190.EXTIレジスタマップセクション

アドレスオフセット	説明
0x000 - 0x01C	一般の設定可能なイベント[22:0]の設定
0x060 - 0x06C	EXTI IOポートのマルチプレクサ選択
0x070	EXTI保護ロック設定
0x080 - 0x0BC	CPU入力イベント設定

すべてのレジスタには、ワード（32bit）、ハーフワード（16bit）、およびバイト（8bit）単位でアクセスできます。

23.6.1 EXTI立上がりトリガ選択レジスタ（EXTI_RTSR1）

アドレスオフセット：0x000

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RT25	RT24	RT23	RT22	RT21	RT20	RT19	RT18	RT17	RT16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **RTx：**設定可能なイベント入力x[(1)]（x = 25〜0）の立上がりトリガ･イベント設定ビット

EXTI_SECCFGR.SECxが無効のとき、RTxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、RTxはセキュア･アクセスによってのみアクセスできます。このビットxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、RTxは非特権および特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、RTxは特権アクセスによってのみアクセスできます。このビットxへの非特権書込みは破棄され、非特権読出しには0が返ります。

0：入力ラインの（イベントと割込みについて）立上がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立上がりトリガは有効です。

注： RT25、RT24、およびRT23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

1. 設定可能なイベント入力はエッジトリガされています。これらの入力上でグリッチが生成されないようにする必要があります。レジスタへの書込み中に設定可能なイベント入力で立上がりエッジが発生した場合、関連するペンディングビットはセットされません。同じ設定可能なイベント入力に対して、立上がりエッジトリガと立下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

23.6.2 EXTI立下がりトリガ選択レジスタ（EXTI_FTSR1）

アドレスオフセット：0x004

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FT25	FT24	FT23	FT22	FT21	FT20	FT19	FT18	FT17	FT16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **FTx：**設定可能なイベント入力x[(1)]（x = 25～0）の立下がりトリガ･イベント設定ビット

EXTI_SECCFGR.SECxが無効のとき､FTxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、FTxはセキュア･アクセスによってのみアクセスできます。このFTxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、FTxは非特権および特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、FTxは特権アクセスによってのみアクセスできます。このFTxへの非特権書込みは破棄され、非特権読出しには0が返ります。

0：入力ラインの（イベントと割込みについて）立下がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立下がりトリガは有効です。

注： FT25､FT24､およびFT23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については､EXTIラインの接続表を参照してください｡存在しない場合､このビットは予約済みとみなし、リセット値に保持してください。

1. 設定可能なイベント入力はエッジトリガされています。これらの入力上でグリッチが生成されないようにする必要があります。レジスタへの書込み中に設定可能なイベント入力で立下がりエッジが発生した場合、関連するペンディングビットはセットされません。同じ設定可能なイベント入力に対して、立上がりエッジトリガと立下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

23.6.3 EXTIソフトウェア割込みイベントレジスタ（EXTI_SWIER1）

アドレスオフセット：0x008

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SWI25	SWI24	SWI23	SWI22	SWI21	SWI20	SWI19	SWI18	SWI17	SWI16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWI15	SWI14	SWI13	SWI12	SWI11	SWI10	SWI9	SWI8	SWI7	SWI6	SWI5	SWI4	SWI3	SWI2	SWI1	SWI0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **SWIx：**イベントx（x = 25～0）によるソフトウェア割込み

EXTI_SECCFGR.SECxが無効のとき、SWIxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、SWIxはセキュア･アクセスによってのみアクセスできます。このSWIxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、SWIxは非特権および特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、SWIxは特権アクセスによってのみアクセスできます。このSWIxへの非特権書込みは破棄され、非特権読出しには0が返ります。

ソフトウェア割込みは、EXTI_RTSRおよびEXTI_FTSRの設定とは無関係に生成されます。読出し時には必ず0が返ります。

0：0を書き込んでも、ビットの値は変化しません。

1：1を書き込むと、イベントxで立上がりエッジ･イベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

注： SW25、SW24、およびSW23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.4 EXTI立上がりエッジペンディングレジスタ（EXTI_RPR1）

アドレスオフセット：0x00C

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RPIF25	RPIF24	RPIF23	RPIF22	RPIF21	RPIF20	RPIF19	RPIF18	RPIF17	RPIF16
						rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RPIF15	RPIF14	RPIF13	RPIF12	RPIF11	RPIF10	RPIF9	RPIF8	RPIF7	RPIF6	RPIF5	RPIF4	RPIF3	RPIF2	RPIF1	RPIF0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **RPIFx**：設定可能なイベント入力xの立上がりエッジ･ペンディング･ビット（x = 25〜0）

EXTI_SECCFGR.SECxが無効のとき、RPIFxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、RPIFxはセキュア･アクセスによってのみアクセスできます。このRPIFxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、RPIFxは非特権および特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、RPIFxは特権アクセスによってのみアクセスできます。このRPIFxへの非特権書込みは破棄され、非特権読出しには0が返ります。

0：立上がりエッジトリガリクエストは発生していません。

1：立上がりエッジトリガリクエストが発生しました。

このビットは、立上がりエッジイベントまたはEXTI_SWIERソフトウェアトリガが設定可能なイベントラインに到着したときにセットされます。このビットは1を書き込むことによってクリアされます。

注： RPIF25、RPIF24、およびRPIF23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.5 EXTI立下がりエッジペンディングレジスタ（EXTI_FPR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FPIF25	FPIF24	FPIF23	FPIF22	FPIF21	FPIF20	FPIF19	FPIF18	FPIF17	FPIF16
						rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FPIF15	FPIF14	FPIF13	FPIF12	FPIF11	FPIF10	FPIF9	FPIF8	FPIF7	FPIF6	FPIF5	FPIF4	FPIF3	FPIF2	FPIF1	FPIF0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **FPIFx**：設定可能なイベント入力x立下がりエッジペンディングビット（x = 25〜0）

EXTI_SECCFGR.SECxが無効のとき、FPIFxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、FPIFxはセキュア･アクセスによってのみアクセスできます。このFPIFxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、FPIFxは非特権および特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、FPIFxは特権アクセスによってのみアクセスできます。このFPIFxへの非特権書込みは破棄され、非特権読出しには0が返ります。

0：立下がりエッジトリガリクエストは発生していません。

1：立下がりエッジトリガリクエストが発生しました。

このビットは、立下がりエッジイベントが設定可能なイベントラインに到着したときにセットされます。このビットは1を書き込むことによってクリアされます。

注： FPIF25、FPIF24、およびFPIF23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.6 EXTIセキュリティ設定レジスタ（EXTI_SECCFGR1）

アドレスオフセット：0x014

リセット値：0x0000 0000

このレジスタは書込みアクセスセキュリティを提供します。非セキュア書込みアクセスは無視され、不正アクセスイベントが生成されます。非セキュア読出しにはレジスタのデータが返ります。本レジスタにはセキュリティ対応入力イベント用のビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **SECx：**イベント入力x（x = 25～0）のセキュリティ有効化

EXTI_PRIVCFGR.PRIVxが無効のとき、SECxは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、SECxは特権アクセスによってのみ書込みができます。このSECxへの非特権書込みは破棄されます。

0：イベントのセキュリティは無効（非セキュア）です。

1：イベントのセキュリティは有効（セキュア）です。

注： SEC25、SEC24、およびSEC23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.7 EXTI特権設定レジスタ（EXTI_PRIVCFGR1）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタは特権書込みアクセス保護を提供します。非特権読出しにはレジスタのデータが返ります。本レジスタにはセキュリティ対応入力イベント用のビットのみが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **PRIVx**：イベント入力x（x = 25〜0）のセキュリティ有効化

EXTI_SECCFGR.SECxが無効のとき、PRIVxはセキュアおよび非セキュア·アクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、PRIVxはセキュア·アクセスによってのみ書込みができます。このPRIVxへの非セキュア書込みは破棄されます。

0：イベントの特権は無効です（非特権）。

1：イベントの特権は有効です（特権）。

注： PRIV25、PRIV24、およびPRIV23 ビットはSTM32U5 シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.8 EXTI外部割込み選択レジスタ（EXTI_EXTICRm）

アドレスオフセット：0x060 + 0x4 * (m - 1) （m = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXTI{4*(m-1)+3}[7:0]								EXTI{4*(m-1)+2}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI{4*(m-1)+1}[7:0]								EXTI{4*(m-1)}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **EXTI{4*(m-1)+3}[7:0]**：EXTI{4*(m-1)+3}のGPIOポートの選択

これらのビットは、EXTI{4*(m-1)+3}外部割込みのソース入力を選択するために、ソフトウェアで書き込みます。

EXTI_SECCFGR.SEC{4*(m-1)+3}が無効のとき、このフィールドは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SEC{4*(m-1)+3}が有効なとき、このフィールドはセキュアアクセスによってのみアクセスできます。非セキュア書込みは破棄され、非セキュア読出しには0が返されます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+3}が無効のとき、このフィールドは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+3}が有効なとき、このフィールドは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0x00：PA[{4*(m-1)+3}]ピン

0x01：PB[{4*(m-1)+3}]ピン

0x02：PC[{4*(m-1)+3}]ピン

0x03：PD[{4*(m-1)+3}]ピン

0x04：PE[{4*(m-1)+3}]ピン

0x05：PF[{4*(m-1)+3}]ピン

0x06：PG[{4*(m-1)+3}]ピン

0x07：PH[{4*(m-1)+3}]ピン

0x08：PI[{4*(m-1)+3}]ピン

0x09：PJ[{4*(m-1)+3}]ピン

その他：予約済み

ビット23:16 **EXTI{4*(m-1)+2}[7:0]**：EXTI{4*(m-1)+2}のGPIOポートの選択

これらのビットは、EXTI{4*(m-1)+2}外部割込みのソース入力を選択するために、ソフトウェアで書き込みます。

EXTI_SECCFGR.SEC{4*(m-1)+2}が無効のとき、このフィールドは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SEC{4*(m-1)+2}が有効なとき、このフィールドはセキュアアクセスによってのみアクセスできます。非セキュア書込みは破棄され、非セキュア読出しには0が返されます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+2}が無効のとき、このフィールドは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+2}が有効なとき、このフィールドは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0x00：PA[{4*(m-1)+2}]ピン
0x01：PB[{4*(m-1)+2}]ピン
0x02：PC[{4*(m-1)+2}]ピン
0x03：PD[{4*(m-1)+2}]ピン
0x04：PE[{4*(m-1)+2}]ピン
0x05：PF[{4*(m-1)+2}]ピン
0x06：PG[{4*(m-1)+2}]ピン
0x07：PH[{4*(m-1)+2}]ピン
0x08：PI[{4*(m-1)+2}]ピン
0x09：PJ[{4*(m-1)+2}]ピン
その他：予約済み

ビット15:8 **EXTI{4*(m-1)+1}[7:0]**：EXTI{4*(m-1)+1}のGPIOポートの選択

これらのビットは、EXTI{4*(m-1)+1}外部割込みのソース入力を選択するために、ソフトウェアで書き込みます。

EXTI_SECCFGR.SEC{4*(m-1)+1}が無効のとき、このフィールドは非セキュアおよびセキュア·アクセスによってアクセスできます。

EXTI_SECCFGR.SEC{4*(m-1)+1}が有効なとき、このフィールドはセキュアアクセスによってのみアクセスできます。非セキュア書込みは破棄され、非セキュア読出しには0が返されます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+1}が無効のとき、このフィールドは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIV{4*(m-1)+1}が有効なとき、このフィールドは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0x00：PA[{4*(m-1)+1]ピン
0x01：PB[{4*(m-1)+1]ピン
0x02：PC[{4*(m-1)+1]ピン
0x03：PD[{4*(m-1)+1]ピン
0x04：PE[{4*(m-1)+1]ピン
0x05：PF[{4*(m-1)+1]ピン
0x06：PG[{4*(m-1)+1]ピン
0x07：PH[{4*(m-1)+1]ピン
0x08：PI[{4*(m-1)+1]ピン
0x09：PJ[{4*(m-1)+1]ピン
その他：予約済み

ビット7:0 **EXTI{4*(m-1)}[7:0]**：EXTI{4*(m-1)}のGPIOポートの選択

これらのビットは、EXTI{4*(m-1)}外部割込みのソース入力を選択するために、ソフトウェアで書き込みます。

EXTI_SECCFGR.SEC{4*(m-1)}が無効のとき、このフィールドは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SEC{4*(m-1)}が有効なとき、このフィールドはセキュアアクセスによってのみアクセスできます。非セキュア書込みは破棄され、非セキュア読出しには0が返されます。

EXTI_PRIVCFGR.PRIV{4*(m-1)}が無効のとき、このフィールドは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIV{4*(m-1)}が有効なとき、このフィールドは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0x00：PA[{4*(m-1)}]ピン
0x01：PB[{4*(m-1)}]ピン
0x02：PC[{4*(m-1)}]ピン
0x03：PD[{4*(m-1)}]ピン
0x04：PE[{4*(m-1)}]ピン
0x05：PF[{4*(m-1)}]ピン
0x06：PG[{4*(m-1)}]ピン
0x07：PH[{4*(m-1)}]ピン
0x08：PI[{4*(m-1)}]ピン
0x09：PJ[{4*(m-1)}]ピン
その他：予約済み

23.6.9 EXTIロックレジスタ（EXTI_LOCKR）

アドレスオフセット：0x070

リセット値：0x0000 0000

このレジスタは書込みアクセス･セキュリティを提供します。非セキュア書込みアクセスは無視され、読出しアクセスにはゼロデータが返されます。そして両者は不正アクセスイベントを生成します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK
															rs

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **LOCK**：グローバル･セキュリティおよび特権設定レジスタ（EXTI_SECCFGRおよびEXTI_PRIVCFGR）のロック

このビットは、リセット後1度のみ書き込まれます。

0：セキュリティおよび特権の設定はオープンで、変更可能です。

1：セキュリティおよび特権の設定はロックされており、変更できません。

23.6.10 割込みマスクによるEXTI CPUウェイクアップレジスタ（EXTI_IMR1）

アドレスオフセット：0x080

リセット値：0x0000 0000

本レジスタには設定可能なイベントに対するビットが含まれています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **IMx**：イベント入力x[1] の割込みマスクによるCPUウェイクアップ（x = 25～0）

EXTI_SECCFGR.SECxが無効のとき､IMxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、IMxはセキュア･アクセスによってのみアクセスできます。このビットへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、IMxは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、IMxは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0：入力イベントxからの割込みリクエストによるウェイクアップはマスクされます。

1：入力イベントxからの割込みリクエストによるウェイクアップはマスクされません。

注： IM25、IM24、およびIM23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

1. デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は0に設定されます。

23.6.11 イベントマスクによるEXTI CPUウェイクアップレジスタ（EXTI_EMR1）

アドレスオフセット：0x084

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **EMx**：イベント入力xのイベント生成マスクによるCPUウェイクアップ（x = 25～0）

EXTI_SECCFGR.SECxが無効のとき、EMxは非セキュアおよびセキュアアクセスによってアクセスできます。

EXTI_SECCFGR.SECxが有効なとき、EMxはセキュア･アクセスによってのみアクセスできます。このビットxへの非セキュア書込みは破棄され、非セキュア読出しには0が返ります。

EXTI_PRIVCFGR.PRIVxが無効のとき、EMxは特権および非特権アクセスによってアクセスできます。

EXTI_PRIVCFGR.PRIVxが有効なとき、EMxは特権アクセスによってのみアクセスできます。このビットへの非特権書込みは破棄されます。

0：ラインxからのイベント生成によるウェイクアップはマスクされます。

1：ラインxからのイベント生成によるウェイクアップはマスクされません。

注： EM25、EM24、およびEM23ビットはSTM32U5シリーズのいくつかのデバイスでのみ利用できます。その利用可能性については、EXTIラインの接続表を参照してください。存在しない場合、このビットは予約済みとみなし、リセット値に保持してください。

23.6.12 EXTIレジスタ･マップ

表191.EXTIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	EXTI_RTSR1	Res.	Res.	Res.	Res.	Res.	Res.	RT25	RT24	RT23	RT22	RT21	RT20	RT19	RT18	RT17	RT16	RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	EXTI_FTSR1	Res.	Res.	Res.	Res.	Res.	Res.	FT25	FT24	FT23	FT22	FT21	FT20	FT19	FT18	FT17	FT16	FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	EXTI_SWIER1	Res.	Res.	Res.	Res.	Res.	Res.	SWI25	SWI24	SWI23	SWI22	SWI21	SWI20	SWI19	SWI18	SWI17	SWI16	SWI15	SWI14	SWI13	SWI12	SWI11	SWI10	SWI9	SWI8	SWI7	SWI6	SWI5	SWI4	SWI3	SWI2	SWI1	SWI0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	EXTI_RPR1	Res.	Res.	Res.	Res.	Res.	Res.	RPIF25	RPIF24	RPIF23	RPIF22	RPIF21	RPIF20	RPIF19	RPIF18	RPIF17	RPIF16	RPIF15	RPIF14	RPIF13	RPIF12	RPIF11	RPIF10	RPIF9	RPIF8	RPIF7	RPIF6	RPIF5	RPIF4	RPIF3	RPIF2	RPIF1	RPIF0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	EXTI_FPR1	Res.	Res.	Res.	Res.	Res.	Res.	FPIF25	FPIF24	FPIF23	FPIF22	FPIF21	FPIF20	FPIF19	FPIF18	FPIF17	FPIF16	FPIF15	FPIF14	FPIF13	FPIF12	FPIF11	FPIF10	FPIF9	FPIF8	FPIF7	FPIF6	FPIF5	FPIF4	FPIF3	FPIF2	FPIF1	FPIF0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	EXTI_ SECCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16	SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	EXTI_ PRIVCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16	PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020 - 0x05C	予約済み	予約済み																															
0x060	EXTI_EXTICR1	EXTI3 [7:0]								EXTI2 [7:0]								EXTI1 [7:0]								EXTI0 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x064	EXTI_EXTICR2	EXTI7 [7:0]								EXTI6 [7:0]								EXTI5 [7:0]								EXTI4 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x068	EXTI_EXTICR3	EXTI11 [7:0]								EXTI10 [7:0]								EXTI9 [7:0]								EXTI8 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表191.EXTIレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x06C	EXTI_EXTICR4	EXTI15 [7:0]								EXTI14 [7:0]								EXTI13 [7:0]								EXTI2 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x070	EXTI_LOCKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK
	リセット値																																0
0x074 - 0x07C	予約済み	予約済み																															
0x080	EXTI_IMR1	Res.	Res.	Res.	Res.	Res.	Res.	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16	IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	EXTI_EMR1	Res.	Res.	Res.	Res.	Res.	Res.	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16	EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

24 巡回冗長検査計算ユニット（CRC）

24.1 概要

CRC（Cyclic Redundancy Check：巡回冗長検査）計算ユニットは、8、16、または32bitデータワードと、ある生成多項式から、CRCコードを得るために使用されます。

他のアプリケーションの中でも、CRCベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがFlashメモリの整合性を確認するひとつの手段となっています。CRC計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたリファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

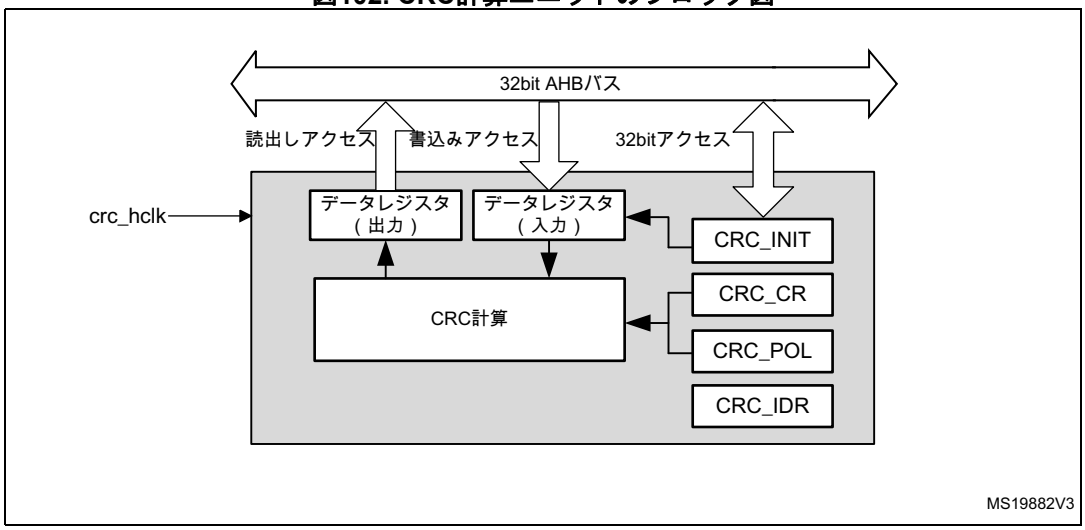
24.2 CRCの主な機能

- CRC-32（Ethernet）多項式：0x4C11DB7を使用
  $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- あるいは、完全にプログラム可能な多項式をプログラム可能なサイズ（7、8、16、32bit）で使用します。
- サイズが8、16、32bitのデータを取り扱います。
- プログラム可能なCRCの初期値
- シングル入力／出力32bitデータレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが32bitの場合、CRCの計算はAHBクロック4サイクル（HCLK）以内に行われます。
- 汎用8bitレジスタ（一時ストレージとして使用可能）
- I/Oデータの可逆性のオプション
- ワード、右詰めハーフワード、右詰めバイトでアクセスできるCRC_DRレジスタを除き、AHBスレーブ･ペリフェラル経由で32bitワードのみでアクセスします。

24.3 CRCの機能説明

24.3.1 CRCブロック図

図102. CRC計算ユニットのブロック図



24.3.2 CRC内部信号

表192.CRC内部入力／出力信号

信号名	信号タイプ	説明
crc_hclk	デジタル入力	AHBクロック

24.3.3 CRC操作

CRC計算ユニットは、1つの32bit読出し／書込みデータレジスタ（CRC_DR）を持っています。このレジスタを使用して、新しいデータを入力し（書込みアクセス）、前回のCRC計算結果を保持します（読出しアクセス）。

データレジスタへの書込み操作のたびに、前回のCRC値（CRC_DRに格納）と新しい値の組み合わせが作成されます。CRC計算は、書き込まれるデータのフォーマットに応じ、32bitデータワード全体に対して、またはバイト単位で行われます。

CRC_DRレジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32bitアクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32bitの場合、4AHBクロックサイクル
- 16bitの場合、2AHBクロックサイクル
- 8bitの場合、1AHBクロックサイクル

入力バッファを使うと、前回のCRC計算によるウェイトステートを待つことなく、すぐに第2のデータを書き込むことができます。

与えられたバイト数に対する書込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5バイトのCRCは、1ワードの書込みと、それに続く1バイトの書込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CRレジスタのREV_IN[1:0]ビットに応じて、8、16、および32bitで行うことができます。

たとえば、入力データ0x1A2B3C4Dは、CRC計算では以下のように使用されます。

- 0x58D43CB2（バイト単位でビットが逆転）
- 0xD458B23C（ハーフワード単位でビットが逆転）
- 0xB23CD458（フルワードでビットが逆転）

また、出力データも、CRC_CRレジスタのREV_OUTビットをセットすることによって逆にすることができます。

この操作はビット･レベルで行われます。例えば、出力データ0x11223344は0x22CC4488に変換されます。

CRC計算機は、CRC_CRレジスタのRESET制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は0xFFFFFFFF）。

CRCの初期値は、CRC_INITレジスタでプログラムすることができます。CRC_DRレジスタは、CRC_INITレジスタの書込みアクセス時に自動的に初期化されます。

CRC_IDRレジスタを使用して、CRC計算に関する一時的な値を保持することができます。このレジスタは、CRC_CRレジスタのRESETビットによる影響を受けません。

多項式のプログラミング可能性

多項式係数はCRC_POLレジスタを介して完全にプログラム可能であり、多項式のサイズはCRC_CRレジスタのPOLYSIZE[1:0]ビットをプログラムすることにより、7、8、16、または32bitに設定できます。偶多項式はサポートされていません。

注： **偶数多項式の種類は$X+X^2+..+X^n$で、奇数多項式の種類は$1+X+X^2+..+X^n$です。**

CRCデータが32bit未満であれば、その値はCRC_DRレジスタの下位ビットから読み出すことができます。

信頼できるCRC計算結果を得るために､CRC計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC計算が進行中である場合、アプリケーションは計算をリセットするか、またはCRC_DRの読出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32（イーサネット）多項式：0x4C11DB7です。

24.4 CRCレジスタ

CRC_DRレジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32bitアクセスのみ可能です。

24.4.1 CRCデータレジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DR[31:0]**：データレジスタビット

このレジスタを使用して、CRC計算機に新しいデータを書き込みます。
読出し時には、前回のCRC計算結果を保持します。
データサイズが32bit未満であれば、下位ビットを使用して正しい値の書込み／読出しを行います。

24.4.2 CRC独立型データレジスタ(CRC_IDR)

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IDR[31:0]**：汎用32bitデータレジスタビット

これらのビットは、4バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CRレジスタのRESETビットによって生成されるCRCリセットの影響を受けません。

24.4.3 CRC制御レジスタ（CRC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	リセット
								rw	rw	rw	rw	rw			rs

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **REV_OUT**：出力データ反転

このビットは、出力データビット順序の反転を制御します。

0：ビットの順序は変わりません。

1：ビットを反転した出力フォーマット。

ビット6:5 **REV_IN[1:0]**：入力データ反転

このビットフィールドは、入力データ･ビット順序の反転を制御します

00：ビットの順序は変わりません。

01：バイト単位でビットを反転します。

10：ハーフワード単位でビットを反転します。

11：ワード単位でビットを反転します。

ビット4:3 **POLYSIZE[1:0]**：多項式サイズ

これらのビットは、多項式のサイズを制御します。

00：32bitの多項式

01：16bitの多項式

10：8bitの多項式

11：7bitの多項式

ビット2:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **RESET**：RESETビット

このビットは、CRC計算ユニットをリセットし、CRC_INITレジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

24.4.4 CRCの初期値（CRC_INIT）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRC_INIT[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_INIT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CRC_INIT[31:0]** *:* プログラム可能なCRCの初期値

このレジスタを使用して、CRCの初期値を書き込みます。

24.4.5 CRC多項式（CRC_POL）

アドレスオフセット：0x14

リセット値：0x04C1 1DB7

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
POL[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **POL[31:0]** *:* プログラム可能な多項式

このレジスタを使用して、CRC計算に使用される多項式の係数を書き込みます。

多項式のサイズが32bit未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

24.4.6 CRCレジスタマップ

表193.CRCレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRC_DR	DR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	CRC_IDR	IDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	CRC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	リセット
	リセット値																									0	0	0	0	0			0
0x10	CRC_INIT	CRC_INIT[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	CRC_POL	POL[31:0]																															
	リセット値	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1	1	1

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

25 CORDICコプロセッサ（CORDIC）

25.1 CORDICの概要

CORDICコプロセッサは、モータ制御、計量、信号処理、その他多くのアプリケーションに共通して使用される演算関数（主として三角関数）のハードウェア高速化を提供します。

ソフトウェア実装に比べてこれらの関数の計算が高速化され、低い動作周波数での使用が可能となったり、他のタスクを実行するためにプロセッササイクルが開放されたりします。

25.2 CORDICの主な機能

- 24bit CORDICローテーションエンジン
- 円モードと双曲線モード
- 回転モードとベクタリングモード
- 関数：sine、cosine、sinh、cosh、atan、atan2、atanh、modulus、平方根、自然対数
- プログラム可能な精度
- 低遅延AHBスレーブインタフェース
- 結果はポーリングや割込みを行わずに準備が出来次第読出し可能
- DMA読出し／書込みチャネル
- DMAによる複数レジスタの読出し／書込み

25.3 CORDICの機能詳細

25.3.1 概要

CORDICは、三角関数と双曲線関数の値を求めるための低コストな逐似近似アルゴリズムです。

三角関数（円）モードでは、ある角度θの正弦と余弦は、回転角の累積和が入力角θに等しくなるまで単位ベクトル[1, 0]を回転させることで求められます。そこで、回転されたベクトルのデカルト座標成分xとyは、θの余弦と正弦にそれぞれ対応しています。反対に、逆正接（y/x）に対応するベクトル[x, y]の角度は、単位ベクトル[1, 0]を得るために角度を連続的に減少させて、[x, y]を回転させることにより求められます。回転角の累積和から元ベクトルの角度が得られます。

CORDICアルゴリズムは、連続した円回転を双曲線に沿ったステップに置き換えることにより、双曲線関数（sinh、cosh、atanh）の計算にも使用できます。

その他の関数は、上記の基本関数から導出可能です。

25.3.2 CORDIC関数

コプロセッサを使用する際の最初のステップは、CORDIC_CRレジスタのFUNCフィールドをしかるべくプログラムすることにより、必要な関数を選択することです。

CORDICコプロセッサでサポートしている関数を表194に列記します。

表194.CORDIC関数

機能	第1引数（ARG1）	第2引数（ARG2）	第1結果（RES1）	第2結果（RES2）
余弦	角度θ	モジュラスm	$m\cdot\cos\theta$	$m\cdot\sin\theta$
正弦	角度θ	モジュラスm	$m\cdot\sin\theta$	$m\cdot\cos\theta$
位相	x	y	atan2(y,x)	$\sqrt{x^2+y^2}$
モジュラス	x	y	$\sqrt{x^2+y^2}$	atan2(y,x)
逆正接	x	なし	$\tan^{-1} x$	なし
双曲線余弦	x	なし	cosh x	sinh x
双曲線正弦	x	なし	sinh x	cosh x
双曲線逆正接	x	なし	$\tanh^{-1} x$	なし
自然対数	x	なし	ln x	なし
平方根	x	なし	$\sqrt{x}$	なし

いくつかの関数は2つの入力引数（ARG1とARG2）をとりますし、2つの結果（RES1とRES2）を同時に生成するものもあります。これはアルゴリズムの副作用であり、2つの値を得るために必要な演算は1回のみであることを意味します。これは、たとえば極座標を直交座標に変換する場合です。sin θによってcos θも生成され、cos θによってsin θも生成されます。直交座標の極座標変換（phase(x,y)、modulus(x,y)）や、双曲線関数（cosh θ、sinh θ）でも同様です。

注： **指数関数exp xは、sinh xとcosh xの和として求められます。また、Nを底とする対数$\log_N$ xは、ln xに定数K（ここでK = 1/ln N）を乗じることにより導出できます。**

特定の関数（atan、log、sqrt）では、スケーリングファクタ（セクション25.3.4参照）を適用することにより、q1.31固定小数点が対応している最大値[-1, 1]を超えて関数の範囲を拡大できます。スケーリングファクタは、その他すべての円関数では0に、双曲線関数では1にセットする必要があります。

余弦

表195.余弦パラメータ

パラメータ	説明	範囲
ARG1	ラジアン単位の角度θをπで割ったもの	[-1, 1]
ARG2	モジュラスm	[0, 1]
RES1	$m \cdot \cos\theta$	[-1, 1]

表195.余弦パラメータ （続き）

パラメータ	説明	範囲
RES2	m · sin θ	[-1, 1]
SCALE	適用外	0

この関数は、-πからπの範囲の角度の余弦を計算します。極座標から直交座標への変換にも使用できます。

第1引数は、ラジアン単位の角度θです。ARG1をプログラムする前に、πで割る必要があります。

第2引数はモジュラスmです。mが1よりも大きい場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG2とする必要があります。

第1結果RES1は、角度の余弦にモジュラスを掛けた値です。

第2結果RES2は、角度の正弦にモジュラスを掛けた値です。

正弦

表196.正弦パラメータ

パラメータ	説明	範囲
ARG1	ラジアン単位の角度θをπで割ったもの	[-1, 1]
ARG2	モジュラスm	[0, 1]
RES1	m · sin q	[-1, 1]
RES2	m · cos θ	[-1, 1]
SCALE	適用外	0

この関数は、-πからπの範囲の角度の正弦を計算します。極座標から直交座標への変換にも使用できます。

第1引数は、ラジアン単位の角度θです。ARG1をプログラムする前に、πで割る必要があります。

第2引数はモジュラスmです。mが1よりも大きい場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG2とする必要があります。

第1結果RES1は、角度の正弦にモジュラスを掛けた値です。

第2結果RES2は、角度の余弦にモジュラスを掛けた値です。

位相

表197.位相パラメータ

パラメータ	説明	範囲
ARG1	x座標	[-1, 1]
ARG2	y座標	[-1, 1]
RES1	ラジアン単位の位相角θをπで割ったもの	[-1, 1]
RES2	モジュラスm	[0, 1]
SCALE	適用外	0

この関数は、ベクトルv = [x y]の-πからπの範囲にある位相角を計算します（atan2(y,x)とも言います）。直交座標から極座標への変換にも使用できます。

第1引数はx座標、すなわち、ベクトルのx軸方向の大きさです。|x| > 1の場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG1とする必要があります。

第2引数はy座標、すなわち、ベクトルのy軸方向の大きさです。|y| > 1の場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG2とする必要があります。

第1結果RES1は、ベクトル**v**の位相角 θです。ラジアン単位で角度を得るには、RES1にπを乗算する必要があります。πに近い値は、位相角の円の性質のために-πに折り返す場合があることに注意してください。

第2結果RES2は、$|v| = \sqrt{x^2 + y^2}$ の式によって与えられるモジュラスです。|**v**| > 1の場合には、RES2の結果は1に飽和しています。

モジュラス

表198.モジュラス·パラメータ

パラメータ	説明	範囲
ARG1	x座標	[-1, 1]
ARG2	y座標	[-1, 1]
RES1	モジュラスm	[0, 1]
RES2	位相角θ	[-1, 1]
SCALE	適用外	0

この関数は、ベクトル**v** = [x y]の大きさ（モジュラス）を計算します。直交座標から極座標への変換にも使用できます。

第1引数はx座標、すなわち、ベクトルのx軸方向の大きさです。|x| > 1の場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG1とする必要があります。

第2引数はy座標、すなわち、ベクトルのy軸方向の大きさです。|y| > 1の場合には、ソフトウェアでスケーリングを適用してq1.31範囲のARG2とする必要があります。

第1結果RES1は、$|v| = \sqrt{x^2 + y^2}$ の式によって与えられるモジュラスです。|v| > 1の場合には、RES1の結果は1.に飽和しています。

第2結果RES2は、ベクトル**v**の位相角 θです。ラジアン単位で角度を得るには、RES2にπを乗算する必要があります。πに近い値は、位相角の円の性質のために-πに折り返す場合があることに注意してください。

逆正接

表199.逆正接パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-1, 1]
ARG2	適用外	-
RES1	ラジアン単位の$2^{-n} \cdot \tan^{-1} x$を^{p}で割ったもの	[-1, 1]
RES2	適用外	-
SCALE	n	[0 7]

この関数は、入力引数xの逆正接を計算します。

第1引数ARG1は、入力値x = tan θです。|x| > 1の場合には、-1 < x · 2-n < 1となるようにソフトウェアでスケーリングファクタ2^{-n}を適用する必要があります。スケーリングされた値x · 2-nはARG1に、スケールファクタnをSCALEパラメータにプログラムする必要があります。

許容される最大入力値はθ = 128であり、この値は角度θ = 89.55度に相当することに注意してください。|x| > 128の場合には、$\tan^{-1}$ xを計算するためにソフトウェア手法を用いる必要があります。

第2引数ARG2は使用しません。

第1結果RES1は、θ = $\tan^{-1}$ xの角度です。ラジアン単位で角度を得るために、RES1に$2^{n} \cdot \pi$を掛ける必要があります。

第2結果RES2は使用しません。

双曲線余弦

表200.双曲線余弦パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.559 0.559]
ARG2	適用外	-
RES1	$2^{-n} \cdot \cosh x$	[0.5 0.846]
RES2	$2^{-n} \cdot \sinh x$	[-0.683 0.683]
SCALE	n	1

この関数は、双曲角xの双曲線余弦を計算します。指数関数e^{x} = cosh x + sinh xとe^{-x} = cosh x - sinh xを計算するためにも使用できます。

第1引数は、双曲角xです。-1.118～+1.118の範囲のxの値のみに対応しています。cosh xの最小値はq1.31フォーマットの範囲を超える1であるため、ソフトウェアでスケーリングファクタ2^{-n}を適用する必要があります。SCALEパラメータにファクタn = 1をプログラムする必要があります。

第2引数は使用しません。

第1結果RES1は、双曲線余弦cosh xです。正しい結果を得るために、RES1に2を掛ける必要があります。

第2結果RES2は、双曲線正弦sinh xです。正しい結果を得るために、RES2に2を掛ける必要があります。

双曲線正弦

表201.双曲線正弦パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.559, 0.559]
ARG2	適用外	-
RES1	$2^{-n} \cdot \sinh x$	[-0.683, 0.683]
RES2	$2^{-n} \cdot \cosh x$	[0.5, 0.846]
SCALE	n	1

この関数は、双曲角xの双曲線正弦を計算します。指数関数$e^x = \cosh x + \sinh x$と$e^{-x} = \cosh x - \sinh x$を計算するためにも使用できます。

第1引数は、双曲角xです。-1.118～+1.118の範囲のxの値のみに対応しています。すべての入力値に対して、ソフトウェアでスケーリングファクタ2^{-n}（n = 1）を適用する必要があります。スケーリングされた値x · 0.5をARG1に、ファクタn = 1をSCALEパラメータにプログラムする必要があります。

第2引数は使用しません。

第1結果RES1は、双曲線正弦sinh xです。正しい結果を得るために、RES1に2を掛ける必要があります。

第2結果RES2は、双曲線余弦cosh xです。正しい結果を得るために、RES2に2を掛ける必要があります。

双曲線逆正接

表202.双曲線逆正接パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.403 0.403]
ARG2	適用外	-
RES1	$2^{-n} \cdot \text{atanh } x$	[-0.559 0.559]
RES2	適用外	-
SCALE	n	1

この関数は、入力引数xの双曲線逆正接を計算します。

第1引数ARG1は、入力値xです。-0.806～+0.806の範囲のxの値にのみ対応しています。値xは、ファクタ2^{-n}（n = 1）でスケーリングする必要があります。スケーリングされた値x · 0.5をARG1に、ファクタn = 1をSCALEパラメータにプログラムする必要があります。

第2引数は使用しません。

第1結果RES1は、双曲線逆正接atanh xです。正しい値を得るために、RES1に2を掛ける必要があります。

第2結果は使用しません。

自然対数

表203.自然対数パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[0.054 0.875]
ARG2	適用外	-
RES1	$2^{-(n+1)}.\ln x$	[-0.279 0.137]
RES2	適用外	-
SCALE	n	[1 4]

この関数は、入力引数xの自然対数を計算します。

第1引数ARG1は、入力値xです。0.107～9.35の範囲のxの値のみに対応しています。値xは、$x \cdot 2^{-n} < 1-2^{-n}$となるように、ファクタ$2^{-n}$でスケーリングする必要があります。スケーリングされた値$x \cdot 2^{-n}$はARG1に、ファクタnをSCALEパラメータにプログラムする必要があります。

有効なスケーリングファクタnと、それに対応する角度xとARG1の範囲を表204に示します。

表204.自然対数のスケーリングファクタと対応範囲

n	xの範囲	ARG1の範囲
1	$0.107 \le x < 1$	$0.0535 \le ARG1 < 0.5$
2	$1 \le x < 3$	0.25 £ ARG1 < 0.75
3	$3 \le x < 7$	$0.375 \le ARG1 < 0.875$
4	$7 \le x \le 9.35$	0.4375 £ ARG1 < 0.584

第2引数は使用しません。

第1結果は自然対数ln xです。正しい値を得るために、RES1に$2^{(n+1)}$を掛ける必要があります。

第2結果は使用しません。

平方根

表205.平方根パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[0.027 0.875]
ARG2	適用外	-
RES1	$2^{-n}\sqrt{x}$	[0.04 1]
RES2	適用外	-
SCALE	n	[0 2]

この関数は、入力引数xの平方根を計算します。

第1引数ARG1は、入力値xです。0.027～2.34の範囲のxの値のみに対応しています。値xは、$x \cdot 2^{-n} < (1 - 2^{(-n-2)})$となるように、ファクタ$2^{-n}$でスケーリングする必要があります。

スケーリングされた値$x \cdot 2^{-n}$はARG1に、ファクタnをSCALEパラメータにプログラムする必要があります。

有効なスケーリングファクタnと、それに対応する角度xとARG1の範囲を表206に示します。

表206.平方根のスケーリング･ファクタと対応範囲

n	xの範囲	ARG1の範囲
0	0.027 ≤ x < 0.75	0.027 ≤ ARG1 <0.75
1	0.75 ≤ x <1.75	0.375 ≤ ARG1 <0.875
2	1.75 ≤ x ≤ 2.341	0.4375 ≤ ARG1 <0.585

第2引数は使用しません。

第1結果はxの平方根です。正しい値を得るために、RES1に2^nを掛ける必要があります。

第2結果は使用しません。

25.3.3 固定小数点表現

CORDICは符号付き固定小数点整数フォーマットで動作します。入出力値はq1.31とq1.15のいずれかです。

q1.31フォーマットでは、数字は1bitの符合部分と31bitの小数部分（バイナリ小数位）によって表現されます。したがって、数値範囲は-1（0x80000000）から$1 - 2^{-31}$（0x7FFFFFFF）となります。

q1.15フォーマットでは、数値範囲は1（0x8000）から$1 - 2^{-15}$（0x7FFF）となります。このフォーマットには、2個の入力引数が単一の32bit書込みに格納可能で、2個の結果が1回の32bit読出しでフェッチ可能であるという利点があります。

25.3.4 スケーリングファクタ

セクション25.3.2に示した関数のいくつかには、スケーリングファクタSCALEが指定されています。これによって、入力、出力、内部レジスタを飽和させることなく、CORDICがサポートしている値の範囲全体が対象となるように、関数の入力範囲を拡張することが可能になります。スケーリングファクタが必要な場合には、ソフトウェアで計算して、CORDIC_CSRレジスタのSCALEフィールドにプログラムする必要があります。スケーリングされた値をCORDIC_WDATAレジスタにプログラムする前に、入力引数をそれに応じてスケーリングする必要があります。また、CORDIC_RDATAレジスタから読み出された結果に対して、スケーリングを元に戻す必要があります。

注： **スケーリングされた値の切り捨てのため、必然的にスケーリングファクタには精度の低下が伴います。**

25.3.5 Precision

結果の精度は、CORDICの繰返し回数に依存します。このアルゴリズムは、三角関数（正弦、余弦、位相、モジュラス）の繰返し1回あたり1バイナリデジットの一定速度で収束します（図103参照）。

双曲線関数（双曲線正弦、双曲線余弦、自然対数）では、CORDICアルゴリズムの特殊性のために収束速度の一定度は劣ります（図104参照）。平方根関数は、双曲線関数のおよそ2倍の速度で収束します（図105参照）。

図103. 三角関数に対するCORDICの収束

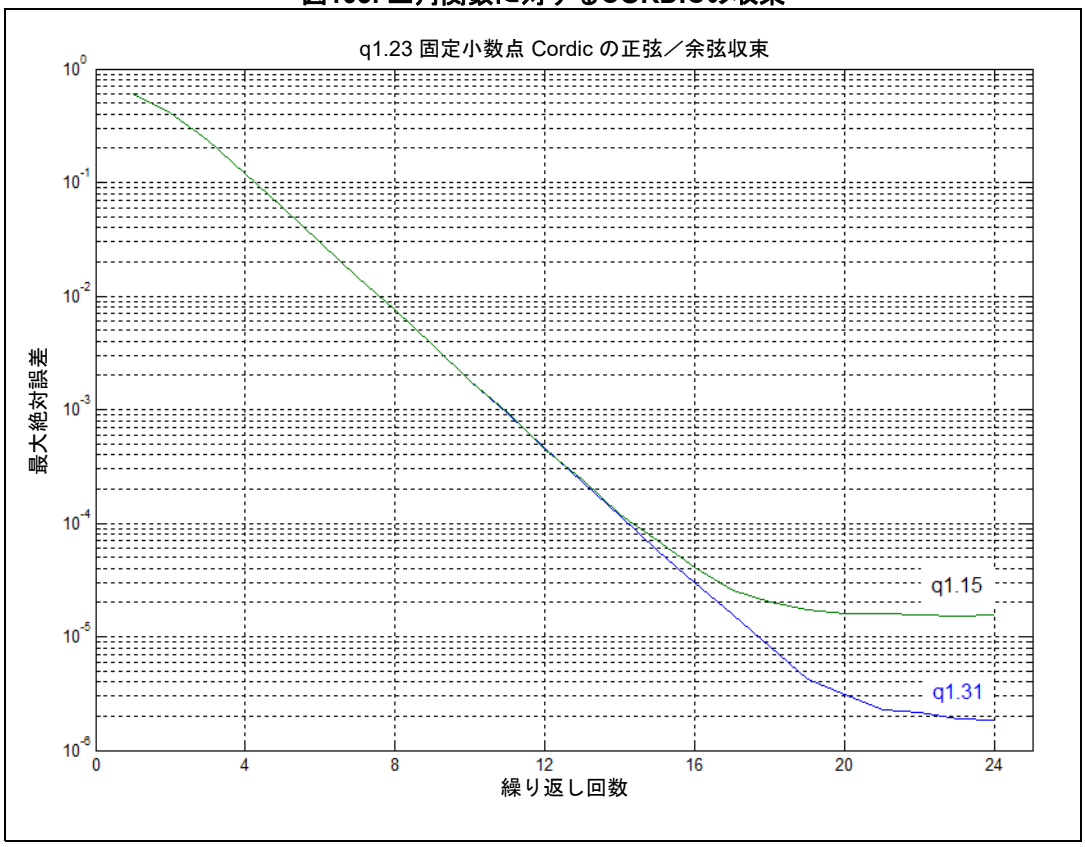


図104. 双曲線関数に対するCORDICの収束

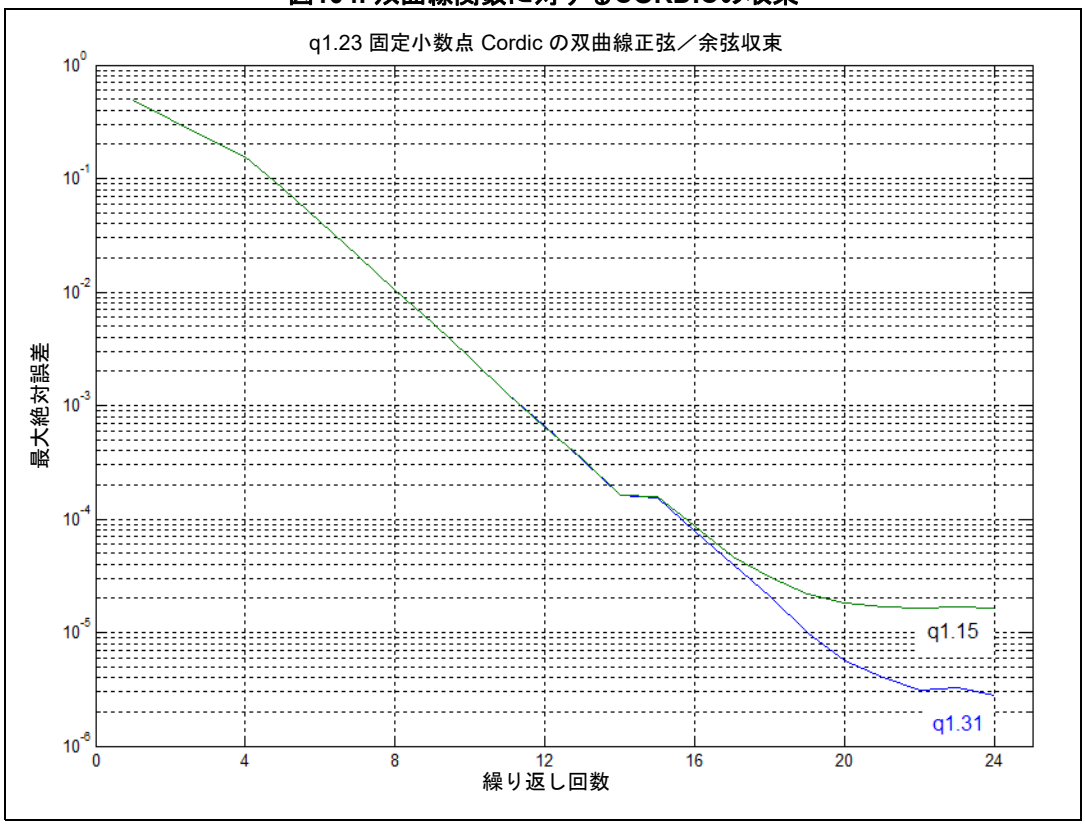
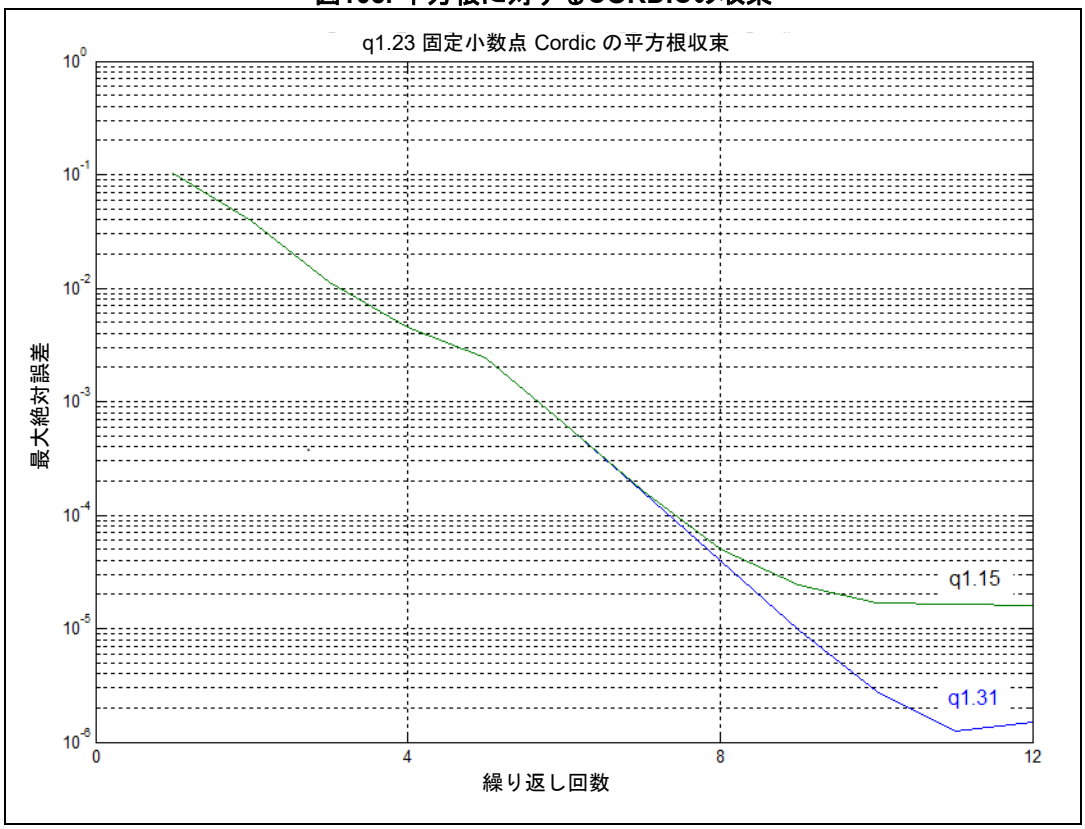


図105. 平方根に対するCORDICの収束



注： **収束速度は、量子化誤差が大きくなり始めるにつれて減少します。**

CORDICは、1クロックサイクルあたり4回の繰返しを実行可能です。それぞれの関数に対して、繰返しを4回ずつ実行した後に残留している最大誤差を、その精度に到達するために必要なクロックサイクル数とともに表207に示します。この表から、必要なサイクル数を決定して、CORDIC_CRレジスタのPRECISIONフィールドにプログラムすることができます。プログラムされた回数の繰返しが完了するや否やコプロセッサが停止し、結果は直ちに読出し可能となります。

表207.精度対繰返し回数

機能	繰返し回数	サイクル数	最大残留誤差(1)	
			q1.31フォーマット	q1.15フォーマット
Sin、Cos、Phase(2)、Mod、Atan (4)	4	1	2^{-3}	2^{-3}
	8	2	2^{-7}	2^{-7}
	12	3	2^{-11}	2^{-11}
	16	4	2^{-15}	2^{-15}
	20	5	2^{-18}	2^{-16}
	24	6	2^{-19}	2^{-16}

表207.精度対繰返し回数 （続き）

機能	繰返し回数	サイクル数	最大残留誤差(1)	
			q1.31フォーマット	q1.15フォーマット
Sinh、Cosh、Atanh、Ln(3)	4	1	2^{-2}	2^{-2}
	8	2	2^{-6}	2^{-6}
	12	3	2^{-10}	2^{-10}
	16	4	2^{-13}	2^{-13}
	20	5	2^{-17}	2^{-15}
	24	6	2^{-18}	2^{-15}
Sqrt(4)	4	1	2^{-7}	2^{-7}
	8	2	2^{-14}	2^{-14}
	12	3	2^{-19}	2^{-15}

1. 最大残留誤差とは、倍精度浮動小数点で実行された同じ計算と比較して、所定の繰返し回数後に残っている最大誤差のことです。q15フォーマットでは最大で2^{-16}、q31フォーマットでは最大で2^{-20}の追加の丸め誤差が生じることがあります。
2. モジュラスが0.5を上回る場合。到達可能な精度は、量子化誤差が大きくなるため、モジュラスの大きさに比例して減少します。
3. SCALE = 1。より大きなスケーリングファクタを用いると、それに比例して到達可能な精度が減少します。
4. SCALE = 0。より大きなスケーリングファクタを用いると、それに比例して到達可能な精度が減少します。

25.3.6 ゼロオーバーヘッドモード

コプロセッサを使用する最速の方法は、実行する関数（FUNC）、必要なクロックサイクル数（PRECISION）、入出力値のサイズ（ARGSIZE、RESSIZE）、入力引数の個数（NARGS）や結果の個数（NRES）、さらに、該当する場合にはスケーリングファクタ（SCALE）をCORDIC_CSRレジスタに前もってプログラムしておくことです。

その後、入力引数をCORDIC_WDATAレジスタに書き込むことにより、計算がトリガされます。正しい個数の入力引数が書き込まれると（かつ進行中の計算がすべて完了すると）、これらの入力引数ならびにその時点のCORDIC_CSR設定値を用いて新しい計算が直ちに起動されます。変更がないのであれば、CORDIC_CSRレジスタを再プログラムする必要はありません。

32bit入力引数が2個必要な場合（ARGSIZE = 0、NARGS = 1）、最初に第1入力引数ARG1を書き込み、その後に第2引数ARG2を書き込む必要があります。一連の計算において第2引数が不変である場合には、最初の計算が始まったところで引数の個数を1（NARGS　= 0）に再プログラムすることにより、2番目の書込みを行わないことができます。関数が変更されない限り、第2引数はプログラムされた値を保持します。

注： **リセット後、ARG2は+1（0x7FFFFFFF）にセットされます。**

16bit引数が2個用いられる場合（ARGSIZE = 1）、ARG1を最下位ハーフワード、ARG2を最上位ハーフワードとして32bitワード1個に格納する必要があります。すると、格納された32bitワードはCORDIC_WDATAレジスタに書き込まれます。この場合（NARGS = 0）に必要な書込み回数は1回のみです。

入力引数が1個のみ（ARG1）の関数では、NARGS = 0とすることをお勧めします。NARGS = 1とした場合、計算をトリガするにはCORDIC_WDATAに対して2回目の書込みを行う必要があります。この場合、ARG2データは使用されません。

ひとたび計算が始まってからCORDIC_RDATAレジスタの読出しを試みると、結果を返す前に、計算が完了するまでバスにウェイト･ステートが挿入されます。したがって、ソフトウェアが入力を書き込んだ直後に、有効かどうか確認するためのポーリングを行わずに結果を読み出すことができます。

もしくは、プロセッサは、適切なクロックサイクル数だけ待ってから結果を読み出すこともできます。この時間は、次の計算のためにCORDIC_CSRレジスタのプログラムを行い、必要に応じて次の入力データを準備することに使用可能です。CORDIC_CSRレジスタは計算の実行中に再プログラム可能であり、その計算結果に影響を与えることはありません。同様に、ひとたび前回の引数が反映されれば、次の引数でCORDIC_WDATAレジスタを更新できます。次の引数と設定値は、前回の計算が完了するまで保留されます。

計算が完了すると、その結果はCORDIC_RDATAレジスタから読み出すことができます。32bitの結果が2個期待されている場合（NRES = 1、RESSIZE = 0）、最初に第1結果（RES1）を読み出し、その後に第2結果（RES2）を読み出す必要があります。32bitの結果が1個だけ期待されている場合（NRES = 0、RESSIZE = 0）、RES1が1回目の読出しで出力されます。

16bitの結果が期待されている場合（RESSIZE = 1）、CORDIC_RDATAを1回読み出すと、32bitワードに格納された両方の結果がフェッチされます。RES1は下位ハーフワード、RES2は上位ハーフワードに格納されています。この場合には、NRES = 0とプログラムすることをお勧めします。NRES = 1とした場合、次の操作のためにCORDICを開放する目的でCORDIC_RDATAの2回目の読出しを行う必要があります。この2回目の読出しによるデータは破棄する必要があります。

次の計算は、期待される個数の引数が書込み済みであることを前提として、期待される個数の結果が読み出されたときに始まります。このことは、常に、実行中または結果読出し待ちの計算が1つ、保留中の演算が1つ存在し得ることを意味しています。演算の保留中にCORDIC_WDATAにさらにアクセスすると、保留中の演算はキャンセルされ、データが上書きされます。

以下のシーケンスは、ゼロオーバーヘッドモードにおけるCORDIC_IPの使用方法の要約です。

1. CORDIC_CSRレジスタに適切な設定値をプログラムします。
2. CORDIC_WDATAレジスタに最初の計算の引数をプログラムします。これによって最初の計算が起動されます。
3. 必要に応じて、次の計算のための設定値でCORDIC_CSRレジスタを更新します。
4. CORDIC_WDATAレジスタに次の計算の引数をプログラムします。
5. CORDIC_RDATAレジスタから結果を読み出します。これによって次の計算がトリガされます。
6. ステップ3に戻ります。

25.3.7 ポーリングモード

新しい結果がCORDIC_RDATAレジスタに有効になると、CORDIC_CSRレジスタにRRDYフラグがセットされます。このフラグは、そのレジスタを読み出すことによってポーリング可能です。CORDIC_RDATAレジスタを（CORDIC_CSRレジスタのNRESフィールドに応じて1回か2回）読み出すと、このフラグはリセットされます。

有効になり次第結果が読み出されるのではないため、RRDYフラグのポーリングは、CORDIC_RDATAレジスタを直接読み出すよりも多少時間がかかります。しかしながら、CORDIC_CSRレジスタの読出しの間にプロセッサとバスインタフェースが停止しないことから、プロセッサの停止が許容されない場合（低遅延割込みが必要な場合など）に、このモードが考慮の対象となる可能性があります。

25.3.8 割込みモード

CORDIC_CSRレジスタの割込みイネーブル（IE）ビットをセットすることにより、RRDYフラグがセットされるたびに割込みが生成されます。割込みは、このフラグがリセットされたときにクリアされます。

このモードによって、割込みサービスルーチンの中で計算結果を読み出すことが可能となり、他のタスクよりも高い優先度が与えられます。ただし、割込み処理の遅延のため、結果の直接読出しやフラグのポーリングよりも時間がかかります。

25.3.9 DMAモード

CORDIC_CSRレジスタにDMA書込みイネーブル（DMAWEN）ビットがセットされ、保留となっている演算がない場合、DMA書込みチャネルリクエストが生成されます。DMAコントローラは、メモリからCORDIC_WDATAレジスタへと第1入力引数（ARG1）を転送可能となります。レジスタに書き込むと、DMAリクエストはデアサートされます。CORDIC_CSRレジスタがNARGS = 1である場合、2番目のDMA書込みチャネルリクエストが生成され、第2入力引数（ARG2）がCORDIC_WDATAレジスタに転送されます。入力引数がすべて書き込まれ、（結果の読出しによって）実行中の計算がすべて完了すると、新しい計算が始まり、別のDMA書込みチャネルリクエストが生成されます。

CORDIC_CSRレジスタにDMA読出しイネーブル（DMAREN）ビットがセットされると、RRDYフラグがアクティブになり、DMA読出しチャネル·リクエストが生成されます。すると、DMAコントローラは、CORDIC_RDATAレジスタからメモリへと第1結果（RES1）を転送可能となります。レジスタを読み出すと、DMAリクエストはデアサートされます。CORDIC_CSRレジスタがNRES = 1である場合、2番目のDMAリクエストが生成され、第2結果（RES2）が読み出されます。結果がすべて読み出されると、RRDYフラグがデアサートされます。

DMA読出しチャネルと書込みチャネルは、別々に有効化できます。両方のチャネルが有効化された場合、CORDICは、プロセッサの介入なしに計算をデータバッファ上で自律的に繰り返すことができます。これによってプロセッサは他のタスクを実行できます。DMAコントローラは、書込みチャネルでは“メモリからペリフェラル”モード、読出しチャネルでは“ペリフェラルからメモリ”モードで動作しています。プロセッサがDMAWENフラグをセットするとシーケンスが開始されることに注意してください。それ以降、CORDICがデータを処理可能な速度でDMA読出しリクエストとDMA書込みリクエストが生成されます。

場合によっては、入力データはメモリに格納され、出力は一定の時間間隔でD/Aコンバータなどの別のペリフェラルへと転送されることがあります。この場合、転送先のペリフェラルは、新しいデータを必要とするたびに DMA リクエストを生成します。DMA コントローラは、次のサンプルをCORDIC_RDATAレジスタから直接フェッチすることができます（この場合、DMAコントローラは、転送元がペリフェラルレジスタであっても“メモリからペリフェラル”モードで操作しています）。結果を読み出す行為によって、CORDICは新しい計算を開始できるようになりますが、その結果としてDMA書込みチャネルリクエストが生成され、DMAコントローラが次の入力値をCORDIC_WDATAレジスタに転送します。DMA書込みチャネルは有効化（DMAWEN = 1）されますが、読出しチャネルは有効化しないでください。

同様に、ADCなど別ペリフェラルからのデータは、（“ペリフェラルからメモリ”モードで）CORDIC_WDATAレジスタに直接転送可能です。DMA書込みチャネルは有効化しないでください。CORDICは入力データを処理し、DMAREN = 1の場合、それが完了するとDMA読出しリクエストを生成します。すると、DMAコントローラは、CORDIC_RDATAレジスタからメモリへと結果を転送します（“ペリフェラルからメモリ”モード）。

注： **CORDIC_CSRレジスタのプログラムのためのDMAリクエストは生成されません。したがって、DMAモードが有用なのは、同じ関数を同じ設定で繰返し実行する場合に限られます。スケーリング·ファクタは一連のDMA転送中に変更することはできません。**

注： **CORDIC_WDATAレジスタやCORDIC_RDATAレジスタにアクセスするDMAの結果として、それぞれのDMAリクエストに確認応答を行う必要があります。これよりも前に当該レジスタへの無関係なアクセスが発生すると、確認応答が途中でアサートされ、DMAチャネルがブロックされることがあります。したがって、DMA読出しチャネルが有効化されたときには、CPUによるCORDIC_RDATAレジスタへのアクセスは避ける必要があります。同様に、DMA書込みチャネルが有効化されたときには、プロセッサはCORDIC_WDATAレジスタへのアクセスを避ける必要があります。**

25.4 CORDICレジスタ

CORDICレジスタには32bitワード･フォーマットでのみアクセスできます。

25.4.1 CORDIC制御／ステータスレジスタ（CORDIC_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0050

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARG SIZE	RES SIZE	NARGS	NRES	DMA WEN	DMA REN	IEN
r									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SCALE[2:0]			PRECISION[3:0]				FUNC[3:0]			
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **RRDY**：結果レディフラグ

出力レジスタに新しい結果がありません。

CORDIC_RDATAレジスタに新しいデータが入っています。

このビットは､CORDICの演算が完了したときに､ハードウェアによってセットされます｡CORDIC_RDATAレジスタが（NRES+1）回読み込まれると、ハードウェアによってリセットされます。

このビットがセットされたときに、IENビットもセットされていると、CORDIC割込みがアサートされます。DMARENビットがセットされていると、DMA読出しチャネルリクエストが生成されます。このビットがセットされている間は、新しい計算は開始されません。

ビット30:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **ARGSIZE**: 入力データの幅

0：32bit

1：16bit

ARGSIZEによって入力データの表現に用いられるビット数が選択されます。

32bitデータが選択されると、CORDIC_WDATAレジスタはq1.31フォーマットの引数を期待します。

16bitデータが選択されると、CORDIC_WDATAレジスタはq1.15フォーマットの引数を期待します。最下位ハーフワードに第1引数（ARG1）、最上位ハーフワードに第2引数（ARG2）が書き込まれます。

ビット21 **RESSIZE**: 出力データの幅

0：32bit

1：16bit

RESSIZEによって出力データの表現に用いられるビット数が選択されます。

32bitデータが選択されると、CORDIC_RDATAレジスタにはq1.31フォーマットの結果が含まれます。

16bitデータが選択されると、CORDIC_RDATAの最下位ハーフワードには第1結果（RES1）がq1.15フォーマットで含まれ、最上位ハーフワードには第2結果（RES2）がこちらもq1.15フォーマットで含まれます。

ビット20

NARGS: CORDIC_WDATAレジスタが期待する引数の個数

次の計算には1回の32bit書込みのみ（もしくはARGSIZE = 1の場合に16bit値2個）が必要です。

次の計算のトリガには、32bit値を2個CORDIC_WDATAレジスタに書き込む必要があります。

読み出すと、ビットの現在の状態が返されます。

ビット19 **NRES**: CORDIC_RDATAレジスタ内の結果の個数

0: 次の計算が完了すると、32bit値1個のみ（もしくはRESSIZE = 1の場合に16bit値2個）がCORDIC_RDATAレジスタに転送されます。CORDIC_RDATAから1回読み出すと、RRDYフラグがリセットされます。

次の計算が完了すると、32bit値が2個CORDIC_RDATAレジスタに転送されます。RRDYフラグのリセットには、CORDIC_RDATAから2回読み出す必要があります。

読み出すと、ビットの現在の状態が返されます。

ビット18 **DMAWEN**: DMA書込みチャネルの有効化

0：無効DMA書込みリクエストは生成されません。

1：有効保留中の演算が存在しない場合には、DMA書込みチャネルにリクエストが常に生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット17 **DMAREN**: DMA読出しチャネルの有効化

0：無効DMA読出しリクエストは生成されません。

1：有効RRDYフラグがセットされるたびに、DMA読出しチャネルにリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット16 **IEN**：割込みの有効化

0：無効割込みリクエストは生成されません。

1：有効RRDYフラグがセットされるたびに、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **SCALE[2:0]**：スケーリングファクタ

このフィールドの値は、引数や結果に適用されたスケーリングファクタを示します。値nは、引数に係数2^{-n}が掛けられているか、結果に2^{n}を掛ける必要のあることを意味します。スケーリングファクタの各関数に対する適用可能性と適用範囲については、セクション25.3.2を参照してください。

ビット7:4 **PRECISION[3:0]**：必要精度（繰り返し回数）

0：予約済み

1から15：（繰り返し回数）／4

所定の精度に対して必要となる繰返し回数を求めるには表207を参照してください。

多くの関数では、このフィールドの推奨範囲は3～6であることに注意してください。

ビット3:0 **FUNC[3:0]**：機能

0：余弦

1：正弦

2：位相

3：モジュラス

4：逆正接

5：双曲線余弦

6：双曲線正弦

7：双曲線逆正接

8：自然対数

9：平方根

10 ～ 15：予約済み

25.4.2 CORDIC引数レジスタ（CORDIC_WDATA）

アドレスオフセット：0x04

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARG[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARG[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **ARG[31:0]**：関数の入力引数

このレジスタには、CORDIC_CSRレジスタのFUNCフィールドで選択された関数に対する入力引数をプログラムします。

32bitフォーマットが選択され（CORDIC_CSR.ARGSIZE = 0）、入力引数が2個必要である場合（CORDIC_CSR.NARGS = 1）、このレジスタに2回の連続した書込みが必要となります。1回目には第1引数（ARG1）を書き込み、2回目には第2引数（ARG2）を書き込みます。

32bitフォーマットが選択され、必要な入力引数は1個のみである場合（NARGS = 0）、このレジスタに必要な書込みは第1引数（ARG1）が含まれた1回のみとなります。

16bitフォーマットが選択された場合（CORDIC_CSR.ARGSIZE = 1）、このレジスタに対する1回の書込みには両方の引数が含まれます。下位ハーフのARG[15:0]に第1引数（ARG1）、上位ハーフのARG[31:16]に第2引数（ARG2）が含まれます。この場合、NARGSは0にセットする必要があります。

各関数に必要な引数とその許容範囲については、セクション25.3.2を参照してください。

必要個数の引数が書き込まれると、前回の計算が完了していれば、CORDICは、与えられた入力引数を用いてCORDIC_CSR.FUNCによって指定された関数を評価します。計算が実行中である場合、その計算が完了して結果が読み出されるまで、ARG1とARG2の値は保留されます。この間にレジスタへの書込みがあると、保留中の演算はキャンセルされ、引数データが上書きされます。

25.4.3 CORDIC結果レジスタ（CORDIC_RDATA）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RES[31:0]**：関数の結果

32bitフォーマットが選択され（CORDIC_CSR.RESSIZE = 0）、出力値が2個期待されている場合（CORDIC_CSR.NRES = 1）、このレジスタはRRDYがセットされたときに2回読み出される必要があります。1回目の読出しによって第1結果（RES1）がフェッチされます。2回目の読出しによって第2結果（RES2）がフェッチされ、RRDYがリセットされます。

32bitフォーマットが選択され、期待されている出力値は1個のみである場合（NRES = 0）、第1結果（RES1）のフェッチに必要となるこのレジスタの読出しは1回のみであり、RRDYがリセットされます。

16bitフォーマットが選択された場合（CORDIC_CSR.RESSIZE = 1）、このレジスタの下位ハーフのRES[15:0]に第1結果（RES1）、上位ハーフのRES[31:16]に第2結果（RES2）が含まれます。この場合、NRESは0にセットする必要があり、実行される読出しは1回のみとなります。

このレジスタを読み出すと、CORDIC_CSRレジスタのRRDYフラグがリセットされます。

25.4.4 CORDICレジスタマップ

表208.CORDICレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CORDIC_CSR	RRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARGSIZE	RESSIZE	NARGS	NRES	DMAWEN	DMAREN	IEN	Res.	Res.	Res.	Res.	SCALE [2:0]				PRECISION [3:0]				FUNC [3:0]			
	リセット値	0									0	0	0	0	0	0	0					0	0	0	0	0	1	0	1	0	0	0	0
0x04	CORDIC_WDATA	ARG[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x08	CORDIC_RDATA	RES[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

26 フィルタ演算アクセラレータ（FMAC）

26.1 FMACの概要

フィルタ演算アクセラレータユニットは、ベクトルの算術演算を実行します。積和演算器（MAC）ユニットから構成されており、ローカルメモリに保持されているベクトル要素のインデックス付けを可能とするアドレス生成ロジックを備えています。

このユニットには、デジタルフィルタの実装を可能とする、入出力上のサーキュラバッファサポートが含まれています。有限インパルス応答フィルタと無限インパルス応答フィルタのどちらも実現可能です。

このユニットによって、高頻度であったり時間がかかったりするフィルタリング動作をCPUからオフロードして、プロセッサを他のタスクに開放することができます。多くの場合には、このような計算をソフトウェアよりも高速に実行可能ですので、速度重視のタスクが高速化されます。

26.2 FMACの主な機能

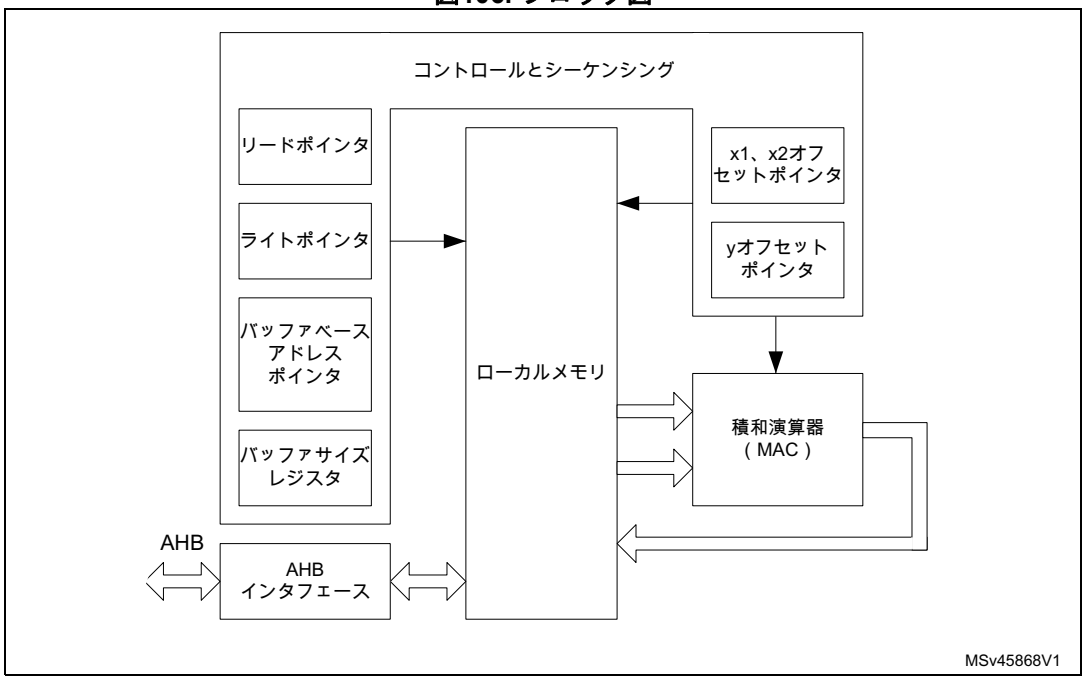
- 16 x 16bit乗算器
- 加減算対応24 + 2bitアキュムレータ
- 16bit入出力データ
- 256 x 16bitローカルメモリ
- メモリの最大3領域（2領域は入力、1領域は出力）をデータバッファとして定義可能（プログラム可能なベースアドレスポインタおよび付随するサイズレジスタによって定義）
- 入出力バッファはサーキュラバッファ対応可能
- フィルタ機能：FIR、IIR（ダイレクトフォーム（直接型）1）
- ベクトル関数：ドット積、畳み込み、相関
- AHBスレーブインタフェース
- DMA読出し／ライトデータチャネル

26.3 FMACの機能詳細

26.3.1 概要

FMACを図106に示します。

図106. ブロック図



このユニットは、固定小数点積和演算器（MAC）の周辺に構築されています。MACは、メモリから16bit符号付き入力の値を2個とり、互いに掛け合わせてアキュムレータの内容に加算します。メモリの入力値アドレスは、ポインタへセットすることにより決定されます。これらのポインタは、内部ハードウェアによってロード、インクリメント、デクリメント、リセットが行えます。ポインタとMAC演算は、要求された演算を実行するために、内蔵シーケンサによって制御されます。

ドット積を計算するために、プロセッサまたはDMAによって2本の入力ベクトルがローカルメモリにロードされ、要求された演算を選択して開始します。入力ベクトル要素の各ペアは、メモリからフェッチされ、互いに掛け合わされてから累積されます。ベクトル要素の処理がすべて完了するとアキュムレータの内容がローカルメモリに格納され、そこからプロセッサやDMAによる読出しが行えます。

有限インパルス応答（FIR）フィルタ演算（畳み込みとも呼ばれます）では、係数ベクトルと入力サンプル·ベクトルのドット積が繰返し計算されます。繰返しのたびに、後者を1サンプル·ディレイでシフトし、最も古いサンプルを破棄して新しいサンプルを追加します。

無限インパルス応答（IIR）フィルタ演算では、フィードバック係数と前回の出力サンプルとの畳み込みがFIR畳み込みの結果に加算されます。

フィルタ演算の詳細は、セクション 26.3.6：フィルタ機能に記載されています。

26.3.2 ローカルメモリとバッファ

このユニットには、ローカルストレージとして用いられる256 x 16bitリード／ライトメモリが含まれています。

- 入力値（入力ベクトルの要素）は、バッファ X1とX2に格納されます。
- 出力値（演算結果）は、別のバッファ Yに格納されます。
- バッファのサイズと位置は、次のように指定されています。
  - x1_base：X1バッファのベースアドレス
  - x2_base：X2バッファのベースアドレス
  - y_base：Yバッファのベースアドレス
  - x1_buf_size：X1バッファに割り当てられたデータ数（16bitワード単位）
  - x2_buf_size：X2バッファに割り当てられたデータ数（16bitワード単位）
  - y_buf_size：Yバッファに割り当てられたデータ数（16bitワード単位）

これらのパラメータは、ユニットの設定時に、対応するレジスタにプログラムされます。

CPU（またはDMAコントローラ）は、初期化機能（セクション 26.3.5：初期化機能）を用いてライト・データ・レジスタに書き込むことにより、各バッファの内容を初期化できます。データは、ライトポインタによって示されるターゲットバッファ内の位置に転送されます。新たな書込みが終わるたびに、ライトポインタはインクリメントされます。ライトポインタが割り当てられたバッファ空間の最後に達すると、ベースアドレスに戻って折り返します。この機能は、演算の前にベクトルの要素をロードしたり、フィルタを初期化してフィルタ係数をロードしたりするために用いられます。

バッファ設定

X1、X2およびYバッファ設定レジスタでバッファサイズとベースアドレスオフセットを設定する必要があります。各機能に必要なバッファサイズは、セクション 26.3.6：フィルタ機能の機能詳細に規定されています。ベースアドレスは内部メモリのどこにでも選択可能ですが、すべてのバッファが内部メモリのアドレス範囲（0x00～0xFF）に収まることが条件となっていますので、ベースアドレス + バッファサイズは256未満である必要があります。

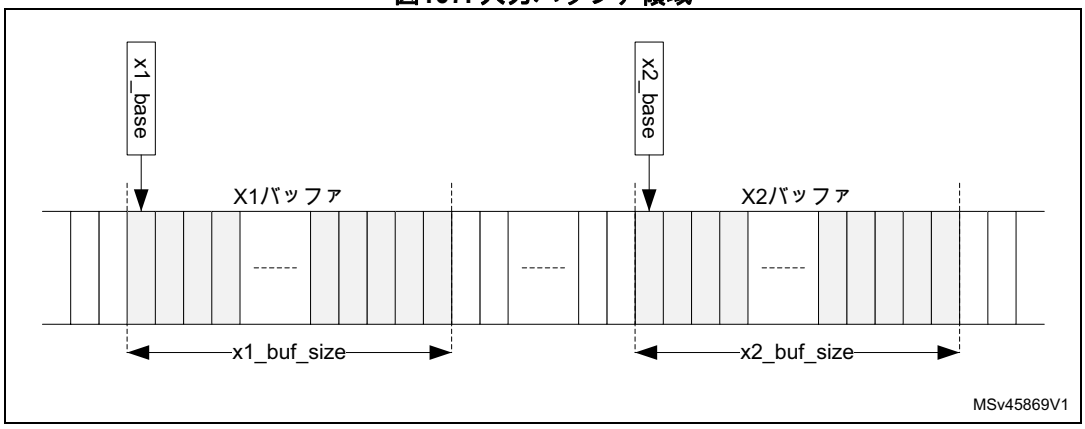
バッファのサイズと位置に関する制約はありません（オーバーラップしていても、完全に一致していても構いません）。フィルタ機能では、こうすると不正な動作を招くおそれがあるため、バッファをオーバーラップさせないことをお勧めします。

サーキュラバッファ演算が必要である場合には、オプションの「ヘッドルーム」dをバッファサイズに加算しても構いません。さらに、ウォーターマークレベルを設定してCPUやDMAの動作を調整することもできます。dの値とウォーターマークレベルは、アプリケーション性能要件に応じて選択する必要があります。スループットを最大とするためには、入力バッファは決して空となってはなりませんので、dはウォーターマーク・レベルよりも多少大きな値とし、割込みやDMAの遅延を可能とするべきです。その一方で、ユニットの処理可能速度に見合った速さで入力データを供給できない場合には、次のデータが書き込まれるのを待つためにバッファを空にすることができるので、（入力にオーバーフローが発生しないことを保証するために）dはウォーターマークレベルと等しくとも構いません。

26.3.3 入力バッファ

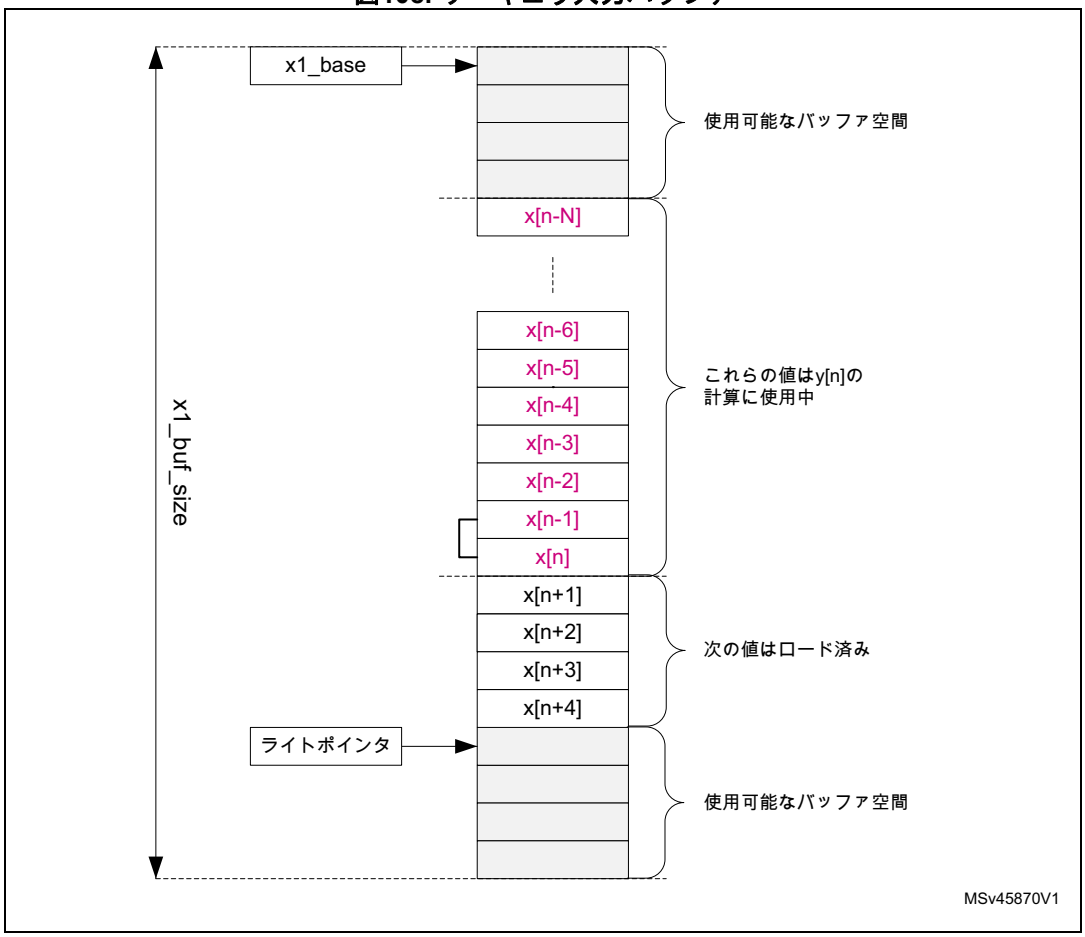
X1バッファとX2バッファは、MACに入力するデータの格納に用いられます。乗算のたびにX1バッファからの値とX2バッファからの値が互いに掛け合わされます。制御ユニットのポインタによって、それぞれの値に対する（バッファベースアドレスを基準とした）リードアドレスオフセットが生成されます。ポインタは、現在の機能に応じてハードウェアによって管理されます。

図107. 入力バッファ領域



X1バッファはサーキュラバッファとして使用できますが、この場合、バッファに空きがあるときには必ず入力バッファに新しいデータが連続転送されます。デジタルフィルタでは、このバッファのプリロードは任意となります。それは、演算が開始されたときにバッファに入力サンプルが何も書き込まれていない場合にエンプティフラグが立ち、このフラグが、演算を開始するのに十分な量となるまでCPUまたはDMAが新しいサンプルをロードするトリガとなるためです。とはいえ、入力データがすでにシステムメモリの中にあってサーキュラ演算が必要ではないベクトル演算の場合は、プリロードは有用です。

図108. サーキュラ入力バッファ



X2バッファは（サーキュラモードではなく）ベクトルモードでのみ使用可能であり、バッファの内容が1つの演算から次の演算まで変化しない場合を除いて、プリロードする必要があります。フィルタ機能に対しては、X2バッファはフィルタ係数の格納に用いられます。

サーキュラバッファとしての動作時には、バッファ内に新しい値が常に存在しているようにするために、バッファに割り当てられたサイズ（x1_buf_size）は、現在の計算に使用されている要素数よりも大きくしなければならないのが一般的です。図108にフィルタ動作のためのバッファレイアウトが図示されています。出力サンプルy[n]を計算している間、このユニットは、x[n-N]からx[n]までN+1個の入力サンプルのセットを使用します。計算が完了すると、このユニットは、x[n-N+1]からx[n+1]までの入力サンプルのセットを使用してy[n+1]の計算を開始します。最も古い入力サンプルx[n-N]が入力セットから抜け、新しいサンプルx[n+1]が加わります。

プロセッサやDMAコントローラは、必要時に新しいサンプルx[n+1]がバッファの中で利用可能であることを保証する必要があります。そうならない場合、バッファにはエンプティフラグが立ち、新しいサンプルが追加されるまでユニットの実行は停止します。アンダーフロー状態はX1バッファにシグナル送信されません。

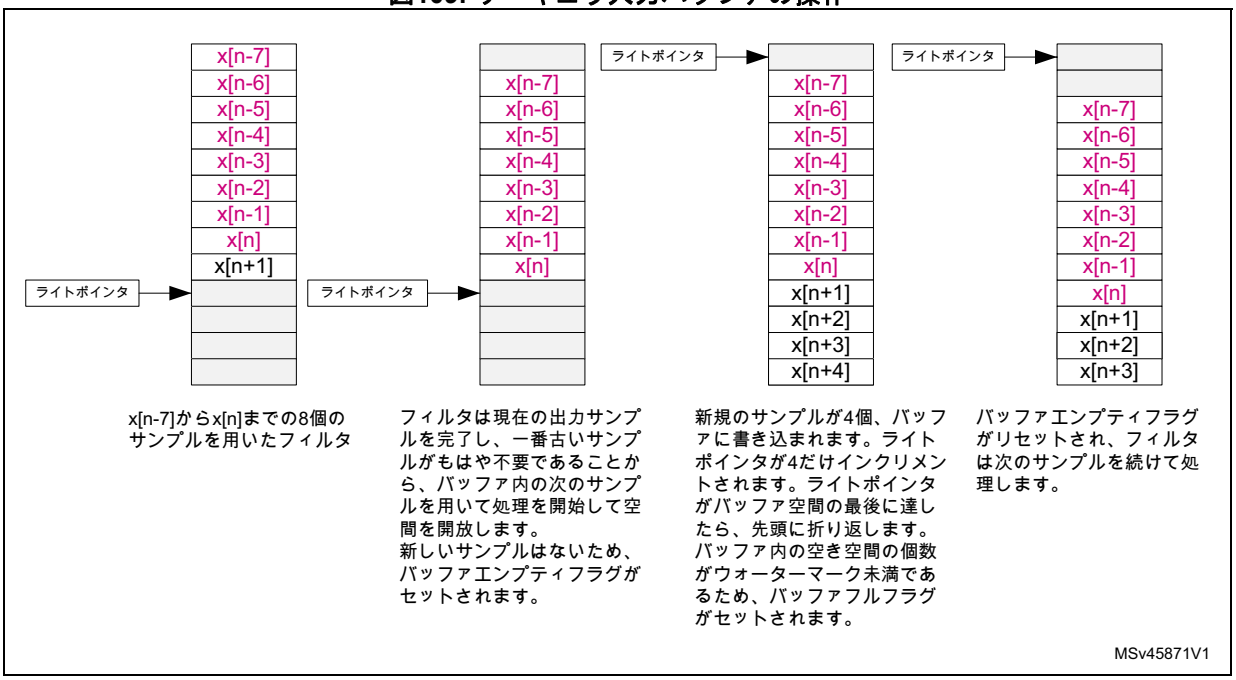
注： **タイマまたはADCなどその他のペリフェラルによってサンプルのフローが制御されている場合、転送元が供給可能な速度よりも、フィルタがそれぞれの新しいサンプルを処理する速度が速いため、バッファは定期的に空となります。これは、フィルタ演算の基本的な特徴です。**

FMAC_X1BUFCFGレジスタのFULL_WMビットフィールドにプログラムされたウォーターマーク閾値よりもバッファ内の空きスペースの個数の方が少ない場合、バッファにはフルフラグが立ちます。フル·フラグがセットされていない限り、割込みが生成されて（有効となっている場合）バッファに

追加データを要求します。ウォーターマークによって、オーバーフローの危険を伴わずに、1回の割込みで複数のデータが転送可能となります。それにもかかわらずオーバーフローが発生した場合には、OVFLエラーフラグがセットされて、ライトデータは無視されます。オーバーフローが発生した場合には、ライト·ポインタがインクリメントされません。

フィルタ演算中のX1バッファの操作を図109に図示します。この例には、ウォーターマークが4にセットされた8タップFIRフィルタが示されています。

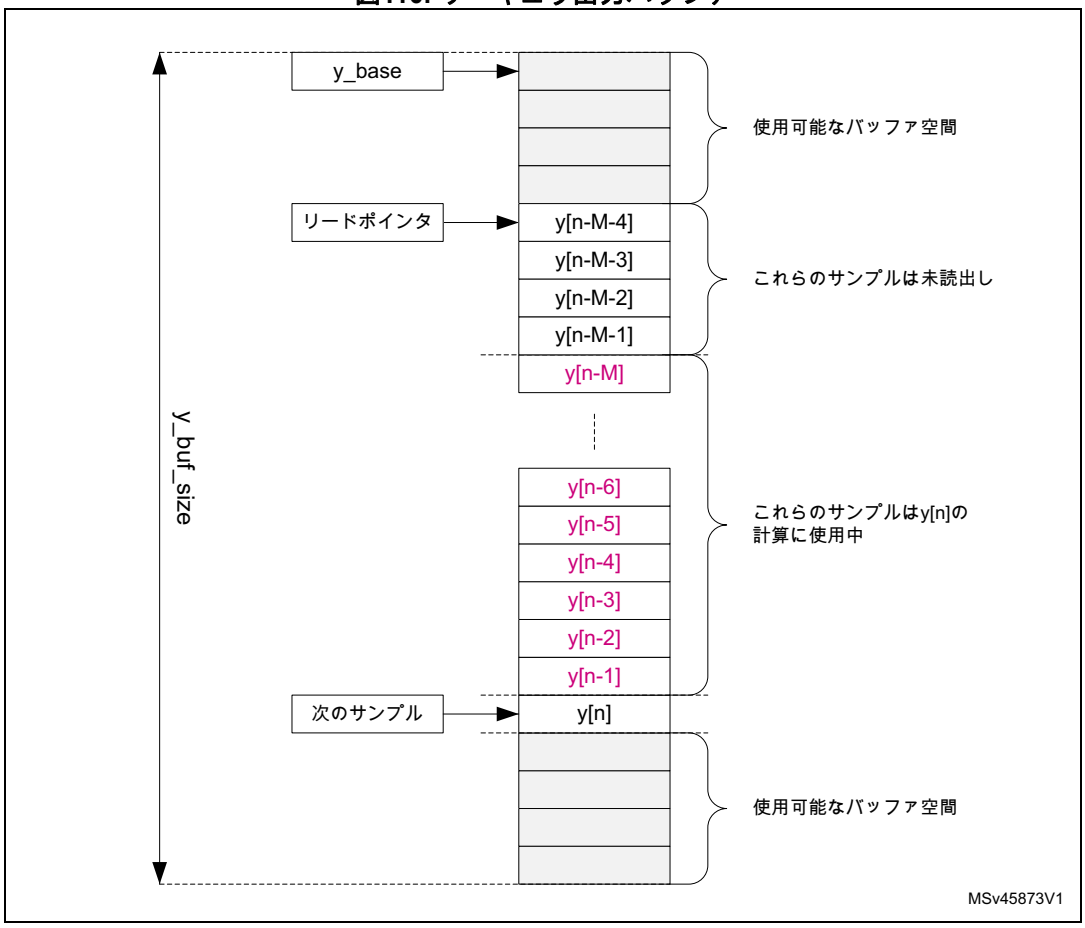
図109. サーキュラ入力バッファの操作



26.3.4 出力バッファ

Y（出力）バッファは、演算結果の格納に用いられます。それぞれの新しい出力値は、プロセッサかDMAコントローラによって読み出されるまでバッファの中に格納されています。リードデータレジスタに対してリードアクセスが行われるたびに、リードポインタで示されているアドレスからリードデータがフェッチされます。毎回の読出しの後にこのポインタはインクリメントされ、割り当てられたYバッファ空間の最後に到達すると、ベースアドレスに戻って折り返されます。

図110. サーキュラ出力バッファ



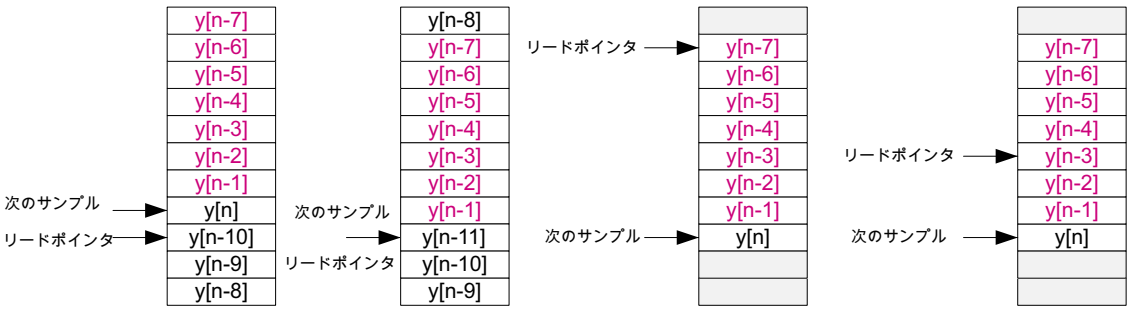
Yバッファはサーキュラバッファとしても使用可能です。次の出力値のアドレスがリード･ポインタで示されているものと同一である場合（未読出しのサンプル）には、バッファにはフルフラグが立ち、サンプルが読み出されるまで実行は停止します。

IIRフィルタの場合、Yバッファは、次の出力サンプルy[n]の計算に使うy[n-M]からy[n-1]までM個の過去の出力サンプルセットの格納に用いられます。新しいサンプルがセットに追加されるたびに、最も古いサンプルであるy[n-M]が脱落します。

FMAC_YBUFCFGレジスタのEMPTY_WMビットフィールドにプログラムされたウォーターマーク閾値よりもバッファ内の未読出しデータの個数の方が少ない場合、バッファにはエンプティフラグが立ちます。エンプティフラグがセットされていない限り、割込みまたはDMAリクエストが生成されて（有効となっている場合）バッファからの読出しを要求します。ウォーターマークによって、アンダーフローの危険を伴わずに、1回の割込みで複数のデータが転送可能となります。それにもかかわらずアンダーフローが発生した場合には、UNFLエラーフラグがセットされます。この場合、リードポインタはインクリメントされずに、リードポインタドレスにあるメモリの内容が読出し操作によって返されます。

サーキュラモードにおけるYバッファの操作を図111に図示します。この例には、ウォーターマークが4にセットされた7タップIIRフィルタが示されています。

図111. サーキュラ出力バッファの操作



y[n-7]からy[n-1]までの7個のサンプルを用いたフィルタ y[n-7] to y[n-1]

フィルタは現在の出力サンプルを完了。バッファに空き空間がないため、フルフラグがセットされて実行が停止します。

サンプルが4個読み出されます。読出しポインタが4だけインクリメントされ、バッファ空間の先頭に折り返します。バッファに空き空間があるできたため、実行が再開します。

サンプルがさらに4個読み出されます。読出しポインタが4だけインクリメントされます。バッファ内の未読出しサンプルの個数がウォーターマーク未満であるため、エンプティフラグがセットされます。

MSv45874V1

26.3.5 初期化機能

以下の機能によってFMACユニットが初期化されます。これらの機能は、STARTビットをセットして、FMAC_PARAMレジスタのFUNCビットフィールドに適切な値を書き込むことによってトリガされます。以下に示すように、PビットフィールドとQビットフィールドにも各機能に適したパラメータ値が含まれている必要があります。Rビットフィールドは使用しません。この機能が完了すると、STARTビットはハードウェアによって自動的にリセットされます。

初期化している間は、DMAリクエストと割込みを無効化していくことをお勧めします。フロー制御は不要であるため、FMACメモリへのデータ転送は、ソフトウェアでもメモリ間DMA転送でも行うことができます。

X1バッファのロード

この機能を使用すると、X1_BASEにあるアドレスから始まるN個の値がX1バッファにプリロードされます。FMAC_WDATAレジスタへの連続書込みを行うと、X1バッファにライトデータがロードされて、ライトアドレスがインクリメントされます。この機能が完了すると、ライト･ポインタはアドレスX1_BASE + Nを指します。

この機能は、あるベクトルの要素をバッファにプリロードしたり、あるフィルタの入力ストレージ要素を初期化したりするのに使用できます。

パラメータ

- パラメータPには、X1バッファにロードされる値の個数Nが含まれています。
- パラメータQとパラメータRは使用しません。

この機能は、FMAC_WDATAレジスタにN回の書込みが行われると完了します。

X2バッファのロード

この機能を使用すると、X2_BASEにあるアドレスから始まるN + M個の値がX2バッファにプリロードされます。FMAC_WDATAレジスタへの連続書込みを行うと、X2バッファにライトデータがロードされて、ライトアドレスがインクリメントされます。

この機能は、ベクトルの要素やフィルタの係数をバッファにプリロードするのに使用できます。IIRの場合には、N個のフィードフォワード係数とM個のフィードバック係数が連結され、一緒にX2バッファにロードされます。係数の総数は、N + Mに等しくなります。FIRの場合には、フィードバック係数は存在しませんので、M = 0となります。

パラメータ

- パラメータPには、アドレスX2_BASEから始まってX2バッファにロードされる値の個数Nが含まれています。
- パラメータQには、アドレスX2_BASE + Nから始まってX2バッファにロードされる値の個数Mが含まれています。
- パラメータRは使用しません。

この機能は、FMAC_WDATAレジスタにN + M回の書込みが行われると完了します。

Yバッファのロード

この機能を使用すると、Y_BASEにあるアドレスから始まるN個の値がYバッファにプリロードされます。FMAC_WDATAレジスタへの連続書込みを行うと、Yバッファにライトデータがロードされて、ライトアドレスがインクリメントされます。この機能が完了すると、リード･ポインタはアドレスY_BASE + Nを指します。

この機能は、IIRフィルタのフィードバックストレージ要素をプリロードするのに使用できます。

パラメータ

- パラメータPには、Yバッファにロードされる値の個数が含まれています。
- パラメータQとパラメータRは使用しません。

この機能は、FMAC_WDATAレジスタにN回の書込みが行われると完了します。

26.3.6 フィルタ機能

以下のフィルタ機能がFMACユニットによってサポートされています。これらの機能は、STARTビットをセットして、FMAC_PARAMレジスタのFUNCビットフィールドに該当する値を書き込むことによってトリガされます。以下に示すように、P、Q、Rビットフィールドにも各機能に適したパラメータ値が含まれている必要があります。フィルタ機能は、ソフトウェアによってSTARTビットがリセットされるまで継続します。

畳み込み（FIRフィルタ）

$\underline{\mathbf{Y}} = \underline{\mathbf{B}} * \underline{\mathbf{X}}$

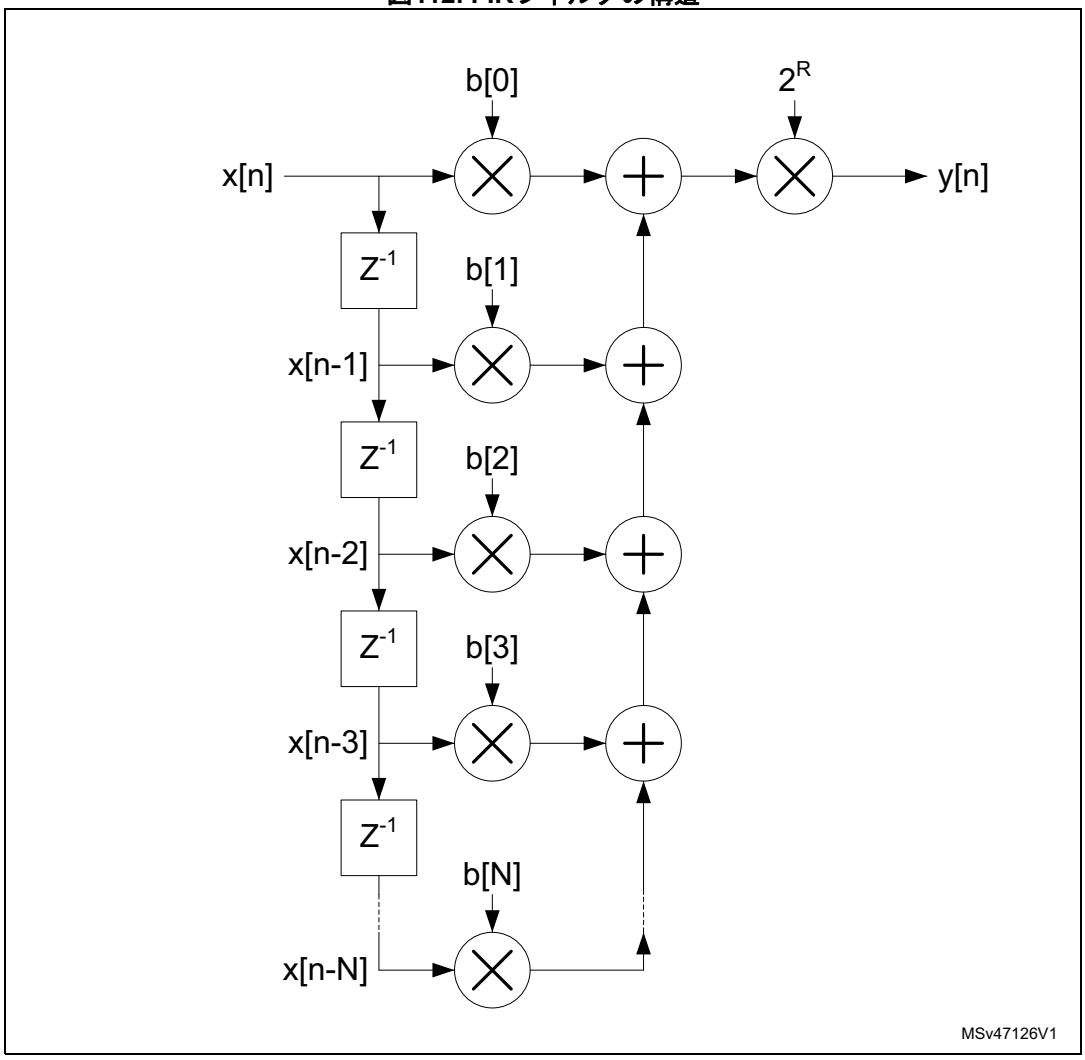
$$y_n = 2^R \cdot \sum_{k=0}^{N} b_k x_{n-k}$$

この機能は、長さがN+1のベクトル$\underline{\mathbf{B}}$と不定長のベクトル$\underline{\mathbf{X}}$の畳み込みを行います。インクリメントする値nに対する$\underline{\mathbf{Y}}$の要素は、ドット積$y_n = \underline{B}.\underline{X}_n$として計算されます。ここで$\underline{X}_n = [x_{n-N},...,x_n]$は、インデックスがn - Nからnまでの N+1個の$\underline{\mathbf{X}}$の要素から構成されます。

この機能は、有限インパルス応答（FIR）フィルタに相当するものであり、ベクトル**B**にはフィルタ係数が、ベクトル**X**にはサンプルデータが含まれています。

フィルタの構造（直接型）を図112に示します。

図112. FIRフィルタの構造



係数ベクトル**B**の次元を反対にすることにより、相互相関ベクトルの計算が可能であることに注意してください。

入力：

- X1バッファにはベクトル**X**の要素が含まれています。長さがN + 1 + dのサーキュラバッファとなります。
- X2バッファにはベクトル**B**の要素が含まれています。長さがN + 1の固定バッファとなります。

出力：

- Yバッファには出力値y_nが含まれています。長さがdのサーキュラバッファとなります。

パラメータ：

- パラメータPIには、[2:127]の範囲の係数ベクトル**B**の長さN+1が含まれています。
- パラメータRIには、アキュムレータ出力に適用されるゲインが含まれています。Yバッファに出力される値には2^Rが掛けられます。ここで、RIは[0:7]の範囲です。
- パラメータQIは使用しません。

この機能は、FMAC_PARAMレジスタのSTARTビットがソフトウェアによってリセットされると完了します。

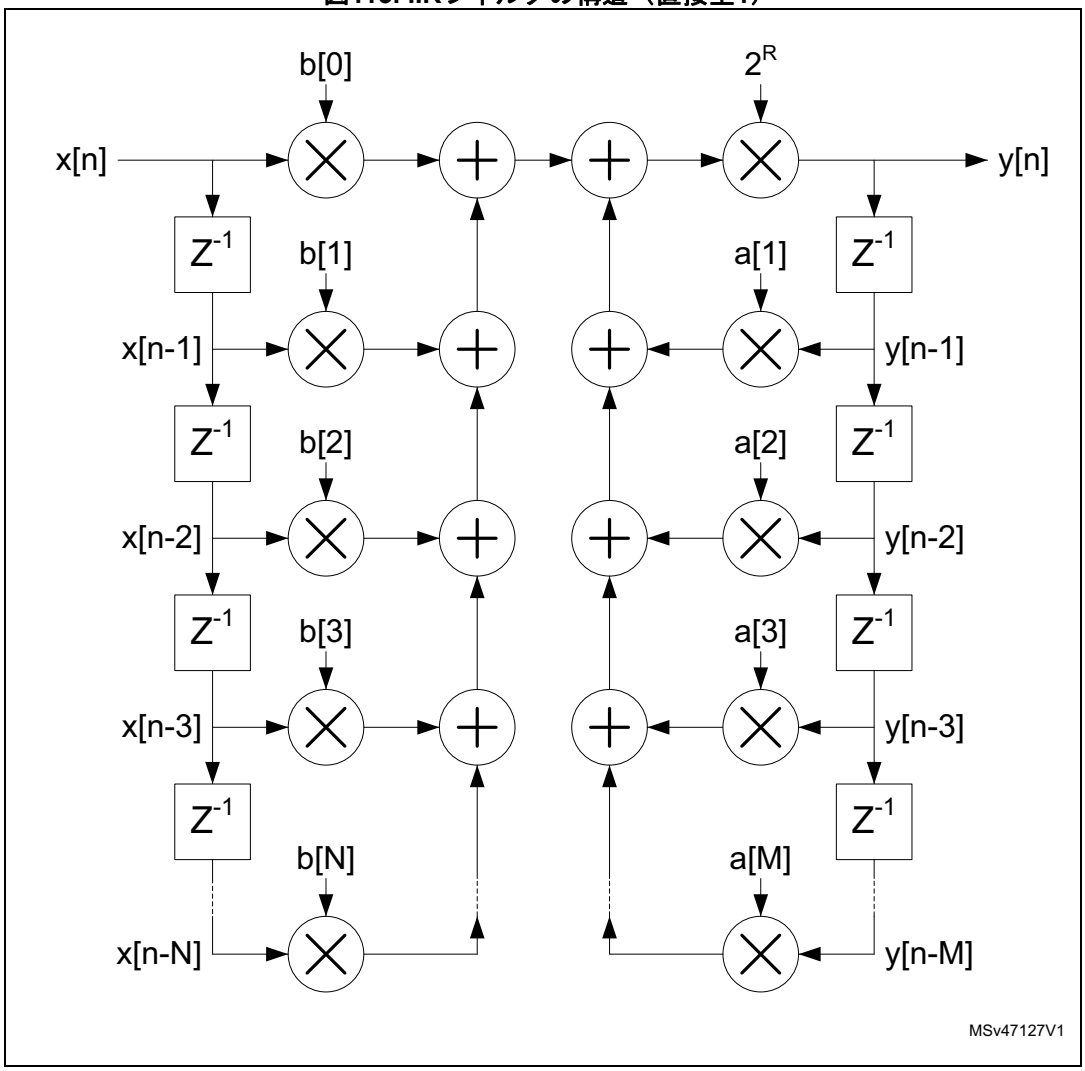
IIRフィルタ

Y = **B*****X**+**A*****Y'**

$$y_n = 2^R \cdot \left(\sum_{k=0}^{N} b_k x_{n-k} + \sum_{k=1}^{M} a_k y_{n-k} \right)$$

この機能は、無限インパルス応答（IIR）フィルタを実行します。フィルタ出力ベクトル**Y**は、長さがN+1の係数ベクトル**B**と不定長のベクトル**X**との畳み込みと、長さがMの2番目の係数ベクトル**A**と遅延出力ベクトルY'との畳み込みの和です。インクリメントする値nに対する**Y**の要素は、$y_n = \underline{B}.\underline{X}_n + \mathbf{A}.\underline{Y}_{n-1}$として計算されます。ここで$\underline{X}_n = [x_{n-N},...,x_n]$は、インデックスがn - Nからnまでの**X**のN+1個の要素から構成され、$\underline{Y}_{n-1} = [y_{n-M},...,y_{n-1}]$はインデックスがn - Mからn - 1までのM個の**Y**の要素から構成されます。フィルタの構造（直接型1）を図113に示します。

図113. IIRフィルタの構造（直接型1）



入力：

- X1バッファにはベクトル**X**の要素が含まれています。長さがN + 1 + dのサーキュラバッファとなります。
- X2バッファには係数ベクトル**B**と**A**が連結された要素（b_0, b_1, b_2..., b_N, a_1, a_2, ..., a_M）が含まれます。長さがM + N + 1の固定バッファとなります。

出力：

- Yバッファには出力値y_nが含まれています。長さがM + dのサーキュラバッファとなります。

パラメータ

- パラメータPIには、[2:64]の範囲の係数ベクトル**B**の長さN+1が含まれています。
- パラメータQには、[1:63]の範囲の係数ベクトル**A**の長さMが含まれています。
- パラメータRには、アキュムレータ出力に適用されるゲインが含まれています。Yバッファに出力される値には2^Rが掛けられます。ここで、Rは[0:7]の範囲です。

この機能は、FMAC_PARAMレジスタのSTARTビットがソフトウェアによってリセットされると完了します。

26.3.7 固定小数点表現

FMACは符号付き固定小数点整数フォーマットで動作します。入出力値はq1.15です。

q1.15フォーマットでは、数字は1bitの符合部分と15bitの小数部分（バイナリ小数位）によって表現されます。したがって、数値範囲は-1（0x8000）から$1 - 2^{-15}$（0x7FFF）となります。

アキュムレータは26bitであり、そのうち22bitが小数部分、4bitが整数／符合部分です（q4.22）。これによって、-8（0x2000000）から+7.99999976（0x1FFFFFF）までの範囲の部分累積和に対応可能となっています。6dBステップで0dBから42dBまでプログラム可能なゲインをアキュムレータの出力に適用できます。

数値範囲を超過してもアキュムレータの内容は飽和しないことに注意してください。値が+7.99999976を上回るか-8を下回る部分和は折り返されますが、それ以降の累積で折り返しが取り消されない限り、これは無害です。ただし、折り返しが発生するとFMAC_SRレジスタのSATフラグがセットされ、FMAC_CRレジスタのSATIENビットがセットされている場合には割込みが発生します。これはフィルタのデバッグに役立ちます。

オプションとして、FMAC_CRレジスタのCLIPENビットをセットすることにより、プログラム可能なゲインを適用した後に、アキュムレータによるデータ出力を飽和させることも可能です。このビットがセットされると、q1.15の数値範囲を超えるあらゆる値は、符合に応じて$1 - 2^{-15}$または-1に設定されます。クリッピングが有効化されていない場合、ゲインを適用した後に使われていないアキュムレータのビットは単純に切り詰められます。

26.3.8 FMACを用いたFIRフィルタの実装

FMACは長さがNのFIRフィルタに対応しています。ここで、Nはタップまたは係数の個数です。長さがNのFIRフィルタに対するローカルメモリの最小要件は2N + 1です。

- 係数N個
- 入力サンプルN個
- 出力サンプル1個

ローカルメモリのサイズは256ですので、Nの最大値は127となります。

最大のスループットが必要な場合には、新しい入力サンプル待ちや出力サンプルの読出し待ちによりフィルタが停止することが絶対にないことを確実なものとするために、入力と出力のサンプルバッファそれぞれに対して、少量の追加スペースd1とd2を割り当てる必要があるかもしれません。この場合、ローカルメモリ要件は2N + d1 + d2となります。

バッファは次のように設定する必要があります。

- X1_BUF_SIZE = N + d1
- X2_BUF_SIZE = N
- Y_BUF_SIZE = d2（追加スペースが不要な場合は1）

バッファ･ベース･アドレスはどこにでも割り当て可能ですが、X2バッファは他のものとオーバーラップさせないでください。そうしないと、係数が上書きされます。次に設定例を示します。

- X2_BASE = 0
- X1_BASE = N
- Y_BASE = 2N + d1

ただし、メモリ空間が限られている場合には、各出力サンプルがもはや必要ではなくなった最も古い入力サンプルの場所を使用できるように、X1バッファ領域とYバッファ領域をオーバーラップさせることができます。

- X2_BASE = 0
- X1_BASE = N
- Y_BASE = N

この場合にはY_BUF_SIZE = X1_BUF_SIZE = N + d1であり、バッファは同期したままとなります。

注： **X1バッファ設定レジスタのFULL_WMビットフィールドは、log_2(d1)以下の値でプログラムする必要があります。そうしないと、入力サンプルがN個書き込まれる前にバッファにフルフラグが立ってしまい、それ以上のサンプルがリクエストされません。同様に、Yバッファ設定レジスタのEMPTY_WMビットフィールドは、log_2(d2)以下とする必要があります。**

フィルタ係数は、ロードX2バッファ機能を用いてX2バッファにプリロードする**必要があります**。オプションとして、最大N個までの任意の個数のサンプルをX1バッファにプリロードできます。FIRフィルタにはフィードバックパスが存在しないため、Yバッファのプリロードは無意味です。

バッファの設定と初期化の後、FMACメモリにデータを読み書きする方法に従ってFMAC_CRレジスタをプログラムする必要があります。

以下の3種類の方法に対応しています。

- ポーリング：DMAリクエストや割込みリクエストは生成されません。ソフトウェアは、WDATAに書き込む前にX1_FULLフラグがローであることを、RDATAから読み出す前にY_EMPTYフラグがローであることを確認する必要があります。
- 割込み：割込みリクエストは、書込みではX1_FULLフラグがロー、読出しではY_EMPTYフラグがローであるときにアサートされます。
- DMA：DMAリクエストは、X1_FULLフラグがローである間にDMAライトチャネルに、Y_EMPTYフラグがローである間にリードチャネルにアサートされます。

読出しと書込みには別の方法を用いることもできます。ただし、同じ操作に対して割込みとDMAリクエストの両方を使用することは推奨されません[a]。有効な組合せを表209に示します。

表 209. 読出し手法と書込み手法の有効な組合せ

WIEN	RIEN	DMAWEN	DMAREN	書込み	読出し
0	0	0	0	ポーリング	ポーリング
0	1	0	0	ポーリング	割込み
1	0	0	0	割込み	ポーリング
1	1	0	0	割込み	割込み
0	0	0	1	ポーリング	DMA
0	0	1	0	DMA	ポーリング
0	0	1	1	DMA	DMA
0	1	1	0	DMA	割込み
1	0	0	1	割込み	DMA

a. 割込みとDMAリクエストの両方が有効化された場合には、DMAのみが転送を行うようにしなくてはなりません。

フィルタは、FMAC_PARAMレジスタに次のビットフィールド値を書き込むことによって開始されます。

- FUNC = 8（FIRフィルタ）
- P = N（係数の個数）
- Q = 「無視」
- R = ゲイン
- START = 1

X1バッファにプリロードされている値がN + d - 2^{FULL_WM}個未満である場合、X1FULLフラグはローのままとなります。FMAC_CRレジスタのWIENビットがセットされると、直ちに割込みリクエストがアサートされて、追加で2^{FULL_WM}個のサンプルをFMAC_WDATAレジスタ経由でバッファに書き込むようにプロセッサにリクエストします。FMAC_SRレジスタでX1FULLフラグがハイになるまで、アサートされたままとなります。割込みサービスルーチンは、FMAC_WDATAレジスタへの 2^{FULL_WM} 回の書込みのたびにX1FULLフラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。同様に、FMAC_CRレジスタのDMAWENビットがセットされた場合には、X1FULLフラグがハイになるまでDMA書込みチャネル·リクエストが生成されます。

（プリロードされたすべてのサンプルを含めて）少なくともN個のサンプルがX1バッファに書き込まれると、フィルタは最初の出力サンプルの計算を行います。

2^{EMPTY_WM}個の出力サンプルがYバッファに書き込まれると、FMAC_SRレジスタのYEMPTYフラグがローになります。FMAC_CRレジスタのRIENビットがセットされると、割込みリクエストがアサートされて、2^{EMPTY_WM}個のサンプルをFMAC_RDATAレジスタ経由でバッファから読み出すようにプロセッサにリクエストします。YEMPTYフラグがハイになるまで、アサートされたままとなります。割込みサービスルーチンは、FMAC_RDATAレジスタからの2^{EMPTY_WM}回の読出しのたびにYEMPTYフラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。FMAC_CRのDMARENビットがセットされた場合には、YEMPTYフラグがハイになるまでDMA読出しチャネル·リクエストが生成されます。

ソフトウェアがSTARTビットをリセットして停止されるまで、フィルタはこの方法での動作を継続します。

26.3.9 FMACを用いたIIRフィルタの実装

FMACは長さがNのIIRフィルタに対応しています。ここで、Nはフィードフォワードタップまたは係数の個数です。フィードバック係数の個数Mは1からN - 1までの任意の値とできます。直接型1の実装のみが実現可能ですので、それ以外の型のために設計されたフィルタは変換する必要があります。

フィードフォワード係数N個、フィードバック係数M個のIIRフィルタに対するメモリの最小要件は2N + 2Mです。

- 係数N + M個
- 入力サンプルN個
- 出力サンプルM個

M = N - 1である場合、実装可能な最大フィルタ長はN = 64となります。

FIRの場合、最大スループットのために、入力と出力のバッファサイズそれぞれに対して、少量の追加空間d1とd2を与えるので、合計のメモリ要件は2M + 2N + d1 + d2となります。

バッファは次のように設定する必要があります。

- X1_BUF_SIZE = N + d1
- X2_BUF_SIZE = N + M
- Y_BUF_SIZE = M + d2

バッファベースアドレスはどこにでも割り当て可能ですが、オーバーラップはさせないでください。次に設定例を示します。

- X2_BASE = 0
- X1_BASE = N + M
- Y_BASE = 2N + M + d1

注： **X1バッファ設定レジスタのFULL_WMビットフィールドは、log_2(d1)以下の値でプログラムする必要があります。そうしないと、入力サンプルがN個書き込まれる前にバッファにフルフラグが立ってしまい、それ以上のサンプルがリクエストされません。同様に、Yバッファ設定レジスタのEMPTY_WMビットフィールドは、log_2(d2)以下とする必要があります。**

フィルタ係数（フィードフォワード係数N個の後にフィードバック係数M個）は、ロードX2バッファ機能を用いてX2バッファにプリロードする**必要があります**。オプションとして、最大N個までの任意の個数のサンプルをX1バッファにプリロードできます。Yバッファには、オプションとして最大M個までの任意の個数の値をプリロードできます。これには、フィードバックディレイラインを初期化する効果があります。

バッファの設定の後、FIRフィルタと同じ方法でFMAC_CRレジスタをプログラムする必要があります（セクション 26.3.8：FMACを用いたFIRフィルタの実装参照）。

フィルタは、FMAC_PARAMレジスタに次のビットフィールド値を書き込むことによって開始されます。

- FUNC = 9（IIRフィルタ）
- P = N（フィードフォワード係数の個数）
- Q = M（フィードバック係数の個数）
- R = ゲイン
- START = 1

X1バッファにプリロードされている値がN + d - 2^{FULL_WM}個未満である場合、X1FULLフラグはローのままとなります。FMAC_CRレジスタのWIENビットがセットされると、直ちに割込みリクエストがアサートされて、追加で2^{FULL_WM}個のサンプルをFMAC_WDATAレジスタ経由でバッファに書き込むようにプロセッサにリクエストします。FMAC_SRレジスタでX1FULLフラグがハイになるまで、アサートされたままとなります。割込みサービスルーチンは、FMAC_WDATAレジスタへの 2^{FULL_WM} 回の書込みのたびにX1FULLフラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。同様に、FMAC_CRレジスタのDMAWENビットがセットされた場合には、X1FULLフラグがハイになるまでDMA書込みチャネル·リクエストが生成されます。

（プリロードされたすべてのサンプルを含めて）少なくともN個のサンプルがX1バッファに書き込まれると、フィルタは最初の出力サンプルの計算を行います。X1バッファの最初のN個のサンプルと、Yバッファの最初のM個のサンプルを使用して（プリロードされているか否かにはよりません）、最初のサンプルが計算されます。最初の出力サンプルは、Yバッファの中のY_BASE + Mに書き込まれます。

2^{EMPTY_WM}個の新たな出力サンプルがYバッファに書き込まれると、FMAC_SRレジスタのYEMPTYフラグがローになります。FMAC_CRレジスタのRIENビットがセットされると、割込みリクエストがアサートされて、2^{EMPTY_WM}個のサンプルをFMAC_RDATAレジスタ経由でバッファから読み出すようにプロセッサにリクエストします。YEMPTYフラグがハイになるまで、アサートされたままとなります。割込みサービスルーチンは、FMAC_RDATAレジスタからの2^{EMPTY_WM}回の読出しのたびにYEMPTYフラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。FMAC_CRのDMARENビットがセットされた場合には、YEMPTYフラグがハイになるまでDMA読出し·チャネル·リクエストが生成されます。

ソフトウェアがSTARTビットをリセットして停止されるまで、フィルタはこの方法での動作を継続します。

26.3.10 フィルタ初期化の例

図114. X1バッファの初期化

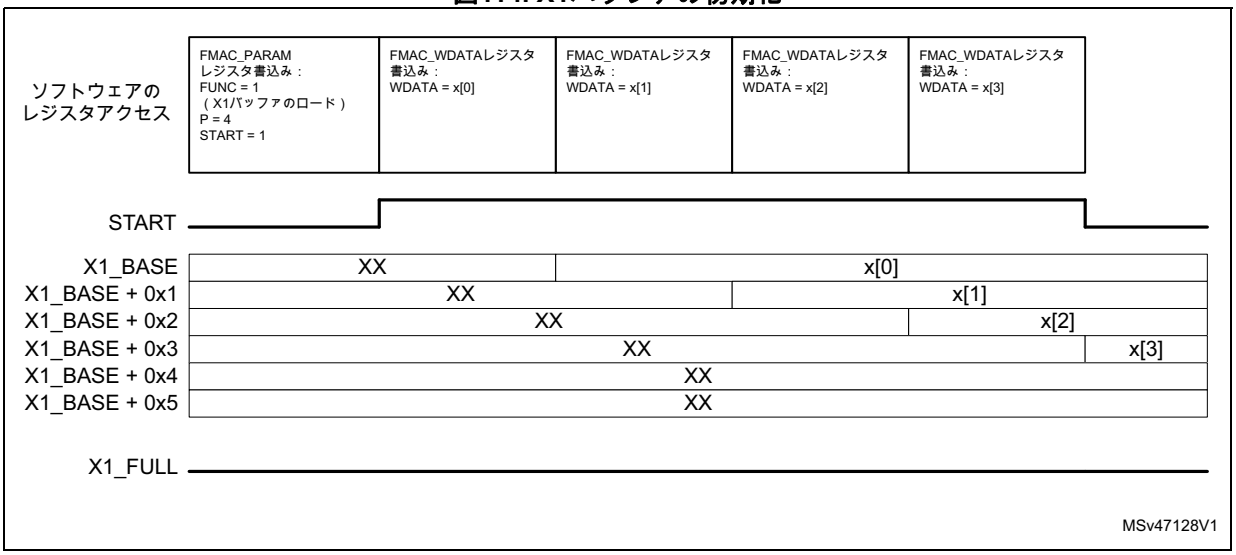


図114の例には、4個のサンプル（P = 4）を用いたX1バッファのプリロードが図示されています。バッファサイズは6です（X1_BUF_SIZE = 6）。初期化は、STARTビットをセットしてFMAC_PARAMレジスタをプログラムすることによって起動されます。すると、4個のサンプルはFMAC_WDATAに書き込まれ、X1_BASE以降のローカルメモリに転送されます。4番目のサンプルが書き込まれるとSTARTビットはリセットされます。この時点で、X1バッファには書込み順に4個のサンプルが格納されており、ライトポインタ（次の空き空間）はX1_BASE + 0x4となっています。

26.3.11 フィルタ演算の例

図115. フィルタ例1

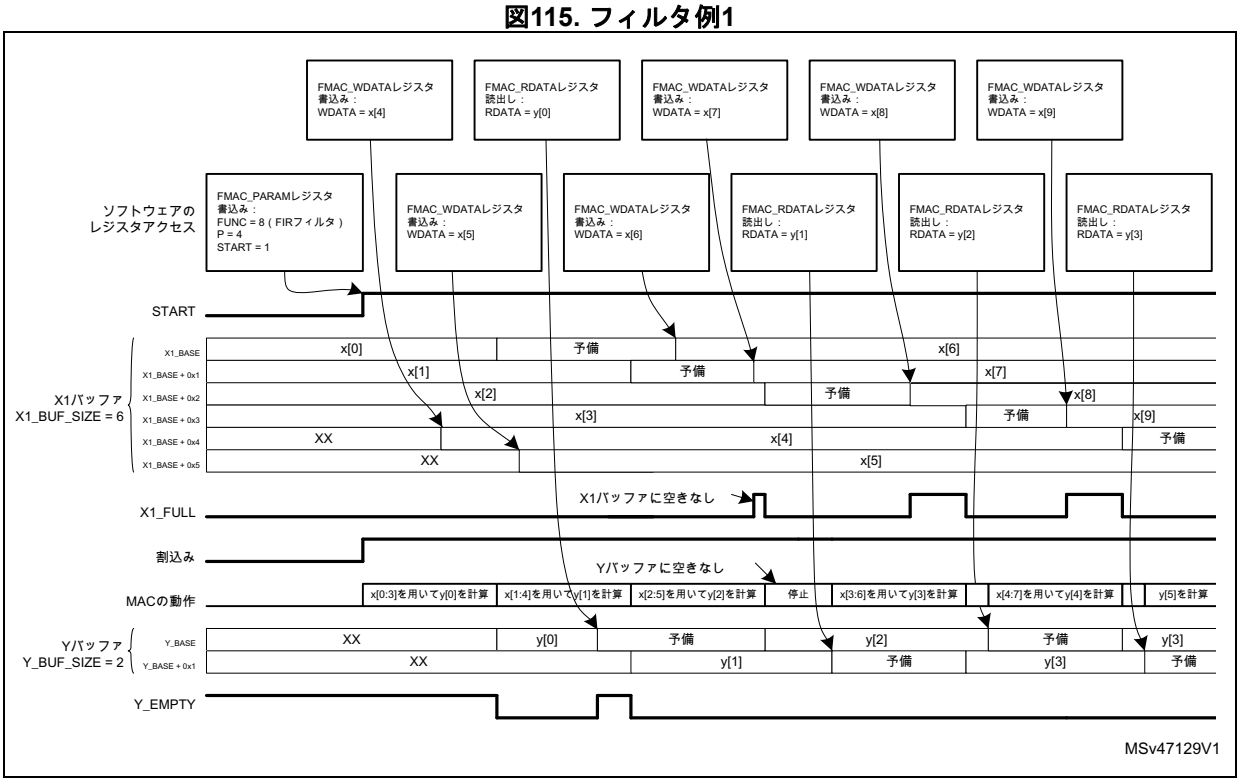


図115の例には、フィルタ演算の始まりが図示されています。フィルタは4タップです（P = 4）。X1バッファサイズは6、Yバッファサイズは2です。FULL_WMビットフィールドとEMPTY_WMビットフィールドはどちらも0にセットされています。図114に示すように、フィルタを開始する前に、X1バッファには4個のサンプルx[0:3]がプリロードされています。したがって、STARTビットがセットされると、フィルタは最初の出力サンプルy[0]の計算を直ちに開始します。X1FULLはセットされていません（X1バッファ内の2つの未初期化領域のため）ので、直ちに割込みがアサートされ、新しいデータをリクエストします。プロセッサが2個の新しいサンプルx[4]およびx[5]をFMAC_WDATAレジスタに書き込むと、それらのサンプルはX1バッファの空いている位置に転送されます。

そうしている間にFMACは最初の出力サンプルy[0]の計算を完了してYバッファの中に書き込むことにより、Y_EMPTYフラグがローとなります。同時に、x[0]サンプルはもはや必要ないため破棄され、メモリ内のその位置（X1_BASEの位置）を開放します。必要な入力サンプルx[1:5]はすべてX1バッファの中に存在するので、FMACは、2番目の出力サンプルy[1]に対する処理を直ちに開始可能となります。

Y_EMPTYフラグがローであるため、プロセッサがx[5]の書込みをし終わった後も割込みはアクティブのままとなります。プロセッサがFMAC_RDATAレジスタからy[0]を読み出し、Yバッファ内のその位置を開放します。依然としてy[1]の計算中であるため、出力バッファの中にサンプルがなくなり、Y_EMPTYフラグがハイになります。ただし、依然としてX1バッファの中に空き空間が存在するために割込みはアクティブのままであり、この空間は次にプロセッサがx[6]などを入れます。

注： **この例では、FMACが処理可能な速度以上で入力バッファにプロセッサがフィルできるため、X1_fullフラグは定期的にアクティブとなっています。ただし、Yバッファを十分な速度で読み出すことには苦労しており、Yバッファの空間開放待ちで定期的にFMACが停止しています。つまり、このフィルタは、最大スループットでは動作していません。その理由は、この例では、フィルタ長が短く、プロセッサが比較的低速であるためです。したがって、Yバッファサイズを増やしても効果はありません。**

図116. フィルタ例2

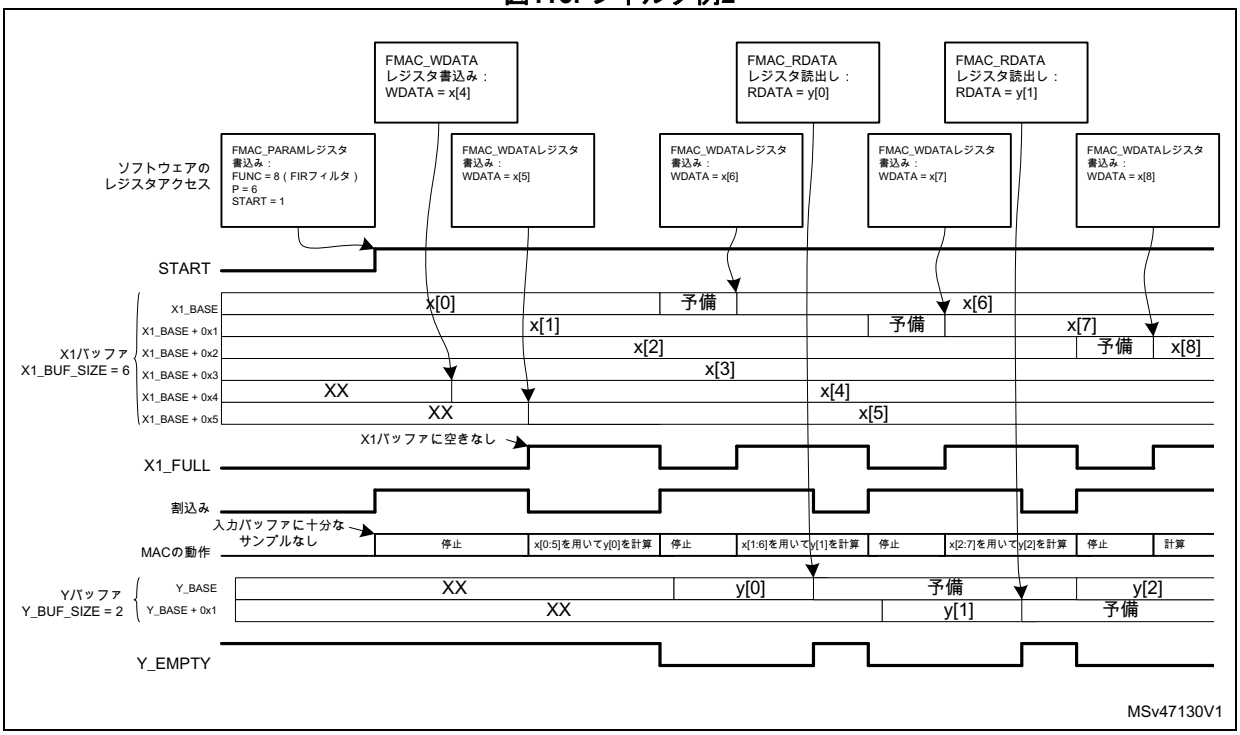


図116の例には、同じフィルタ演算の始まりが図示されていますが、今回のフィルタは6タップです（P = 6）。X1バッファサイズは6、Yバッファサイズは2です。FULL_WMビットフィールドとEMPTY_WMビットフィールドはどちらも0にセットされています。図114に示すように、フィルタを開始する前に、X1バッファには4個のサンプルx[0:3]がプリロードされています。入力バッファに十分な数のサンプルがないことから、X1FULLフラグはセットされませんので、直ちに割込みがアサートされ、新しいデータをリクエストします。FMACは停止します。

プロセッサが2個の新しいサンプルx[4]およびx[5]をFMAC_WDATAレジスタに書き込むと、それらのサンプルはX1バッファの空いている位置に転送されます。X1バッファに未使用のサンプルが6個存在すると、X1_FULLフラグが直ちにアクティブとなり（バッファサイズが6であるため）、割り込むが無効になります。FMACは最初の出力サンプルy[0]の計算を開始します。これには6個の入力サンプルがすべて必要であるため、X1バッファには空き空間が存在せず、X1_FULLフラグはアクティブのままとなります。FMACがy[0]の計算を完了してYバッファに書き込んで初めてx[0]の破棄が可能となり、X1バッファの空間が開放されて、X1_FULLがデアサートされます。同時にY_EMPTYフラグが非アクティブになります。これら両方のフラグの状態によって割込みがアサートされ、最初に新しい入力サンプルの書込みを、その次に計算されたばかりの出力サンプルの読出しを、プロセッサにリクエストします。新しい入力サンプルが書き込まれるまで、FMACは停止したままです。

この例では、プロセッサが新しい入力サンプルを書込み可能となる前に、FMACが現在の出力サンプルの計算を完了するのを待つ必要がありますので、X1バッファが定期的に空となり、FMACが停止します。入力バッファに追加空間をいくらか追加することにより、これは回避可能です。

26.3.12 フィルタ設計のヒント

FMACのアーキテクチャにより、デジタルフィルタの設計には下に詳細を示したような、いくつかの制約条件が課されます。

1. 直接型 2 と転置型の実装は効率的ではありません。このような型のために設計されたフィルタは、直接型1に変換しなくてはなりません。
2. カスケードフィルタ（縦続形）は、単一ステージに組合せるか、別のフィルタとして実装する必要があります。後者の場合には、複数セット（ステージあたり１セット）のフィルタ係数をメモリにプリロードしておき、X2_BASEアドレスレジスタを変更するだけで、どのセットを使用するか選択できます。複数ステージフィルタを実装する最も効率的な方法は、大きなX1バッファに入力サンプルをプリロードしておき、最初のステージの係数を使用してIIRフィルタ機能を実行し、出力サンプルをメモリに戻して格納するというものです。次に、X2_BASEポインタが2番目のステージの係数を指すように変更し、入力バッファに最初のステージの出力を再ロード（必要に応じてゲインを適用）してからIIR機能を再実行します。すべてのステージに対して、この手順を繰返します。最終ステージのサンプルがシステムメモリに転送されたら、入力バッファに次セットの入力サンプルをロードして、新しいラウンドの計算を開始できます。各ラウンドの間で連続性を維持するために、各ステージのNサンプル入力バッファには、最初に、前回ラウンドのN - 1個の最後の入力をすべてプリロードしてから、新しいサンプルを1個加える必要があることに注意してください。同様に、同じ理由により、各ステージの出力バッファには、前回ラウンドから最後のM個のサンプルをロードする必要があります。
3. IIR設計に直接型1を用いると、たとえば入力に大きなステップが生じたり、一部のフィルタ係数の絶対値が1を上回ったりするような場合に、アキュムレータの中に正または負の大きな部分和が生じることがあります。アキュムレータは26bitに制限されているため、折り返し（符合の変化）なしに処理できる最大値は、プラス側で0x1FFFFFF、マイナス側で0x2000000となります。q3.23固定小数点フォーマットでは、これらの値はそれぞれ3.99999988と-4に対応します。累積の最後まで折り返しが取り消されない限り、折り返しが問題となることはありません。ただし、フィルタが開始されて予期しない結果に至る可能性がある場合には、このことは必ずしも当てはまりません。これを避けるためには、出力バッファに適切な値をプリロードすることを検討してください。
4. IIRフィルタには、フィードフォワード（分子）係数[b_0, b_1, ..., b_{N-1}]とフィードバック（分母）係数[1, a_1, ..., a_M]があります。多くのIIRフィルタでは、周波数応答において急峻なロールオフを得るために、一部の分母係数が1を上回る絶対値であることが要求されます。固定小数点のq1.15フォーマットで係数がコード化されることから考えれば、これは不可能です。それにもかかわらず、2^{-R}.[1、a_1、...、a_M]がすべて1未満となるようにファクタ2^{-R}によって分母係数をスケーリングすることにより、このようなフィルタについても実装可能となります。ただし、スケーリングを打ち消すために、アキュムレータの出力に対して2^Rの逆ゲインを適用する必要があります。こうすることにより、信号対雑音比（SNR）に悪影響が生じます。

26.4 FMACレジスタ

26.4.1 FMAC X1バッファ設定レジスタ（FMAC_X1BUFCFG）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：ワード･アクセス

このレジスタは、FMAC_PARAMレジスタのSTART = 0の場合にのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FULL_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X1_BUF_SIZE [7:0]								X1_BASE [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **FULL_WM[1:0]**：バッファフルフラグに対するウォーターマーク

サーキュラモードでの動作時のX1バッファフルフラグの設定閾値を定義します。バッファ内の空き空間の個数が2^{FULL_WM}未満である場合にこのフラグがセットされます。

0：閾値 = 1

1：閾値 = 2

2：閾値 = 4

3：閾値 = 8

1よりも大きい閾値を設定すると、1回の割込みで複数のデータをバッファに転送可能となります。

DMA書込みリクエストが有効である場合（FMAC_CRレジスタのDMAWEN = 1）、閾値は1にセットしなくてはなりません。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **X1_BUF_SIZE[7:0]**：X1バッファに割り当てられたサイズ（16bitワード単位）

最小バッファサイズは、フィルタのフィードフォワードタップ数（+ ウォーターマーク閾値 - 1）です。

ビット7:0 **X1_BASE[7:0]**：X1バッファのベースアドレス

26.4.2 FMAC X2バッファ設定レジスタ（FMAC_X2BUFCFG）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：ワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X2_BUF_SIZE [7:0]								X2_BASE [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **X2_BUF_SIZE[7:0]**：X2バッファのサイズ（16bitワード単位）
機能の動作中（START = 1）はこのビットフィールドは変更できません。

ビット7:0 **X2_BASE[7:0]**：X2バッファのベースアドレス
係数値の変更などのために、START = 1の間にもX2バッファベースアドレスは変更可能です。計算中に係数を変更すると結果に影響が生じるため、これを行っている間はフィルタを停止する必要があります。

26.4.3 FMAC Yバッファ設定レジスタ（FMAC_YBUFCFG）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：ワード･アクセス

このレジスタは、FMAC_PARAMレジスタのSTART = 0の場合にのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	EMPTY_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Y_BUF_SIZE[7:0]								Y_BASE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **EMPTY_WM[1:0]**：バッファエンプティフラグに対するウォーターマーク
サーキュラモードでの動作時のYバッファエンプティフラグの設定閾値を定義します。バッファ内の未読値の個数が2^{EMPTY_WM}未満である場合にこのフラグがセットされます。
0：閾値 = 1
1：閾値 = 2
2：閾値 = 4
3：閾値 = 8
1よりも大きい閾値を設定すると、1回の割込みで複数のデータをバッファから転送可能となります。
DMA読出しリクエストが有効である場合（FMAC_CRレジスタのDMAREN = 1）、閾値は1にセットする必要があります。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **Y_BUF_SIZE[7:0]**：Yバッファのサイズ（16bitワード単位）
FIRフィルタでは、最小バッファサイズは1（+ ウォーターマーク閾値）です。IIRフィルタでは、最小バッファサイズはフィードバックタップ数（+ ウォーターマーク閾値）です。

ビット7:0 **Y_BASE[7:0]**：Yバッファのベースアドレス

26.4.4 FMACパラメータ･レジスタ（FMAC_PARAM）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：ワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
START	FUNC[6:0]							R[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Q[7:0]								P[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **START**：実行の有効化

0：実行を停止します。

1：実行を開始します。

このビットを設定すると、FUNCビットフィールドで選択された機能の実行がトリガされます。ソフトウェアでリセットすると、実行中のあらゆる機能が停止します。初期化機能では、このビットはハードウェアによってリセットされます。

ビット30:24 **FUNC[6:0]**：機能

0：予約済み

1：X1バッファのロード

2：X2バッファのロード

3：Yバッファのロード

4～7：予約済み

8：畳み込み（FIRフィルタ）

9：IIRフィルタ（直接型1）

10～127：予約済み

機能の動作中（START = 1）はこのビットフィールドは変更できません。

ビット23:16 **R[7:0]**：入力パラメータR

このパラメータの値は機能に依存します。

機能の動作中（START = 1）はこのビットフィールドは変更できません。

ビット15:8 **Q[7:0]**：入力パラメータQ

このパラメータの値は機能に依存します。

機能の動作中（START = 1）はこのビットフィールドは変更できません。

ビット7:0 **P[7:0]**：入力パラメータP

このパラメータの値は機能に依存します。

機能の動作中（START = 1）はこのビットフィールドは変更できません。

26.4.5 FMAC制御レジスタ（FMAC_CR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	リセット
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLIP EN	Res.	Res.	Res.	Res.	Res.	DMA WEN	DMA REN	Res.	Res.	Res.	SAT IEN	UNFL IEN	OVFL IEN	WIEN	RIEN
rw						rw	rw				rw	rw	rw	rw	rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RESET**：FMACユニットのリセット

ライト･ポインタとリード･ポインタ、内部制御ロジック、FMAC_SRレジスタ、FMAC_PARAMレジスタ（アクティブである場合にはSTARTビットを含む）がリセットされます。それ以外のレジスタ設定には影響しません。このビットは、ハードウェアによってリセットされます。

0：リセットが非アクティブです。

1：リセットがアクティブです。

ビット15 **CLIPEN**：クリッピングの有効化

0：クリッピングが無効です。q1.15の範囲を超過したアキュムレータの出力の値は折り返します。

1：クリッピングが有効です。q1.15の範囲を超過したアキュムレータの出力の値は、符合に応じて正または負の最大値（+1または-1）に飽和します。

ビット14:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **DMAWEN**：DMAライトチャネルリクエストの有効化

0：無効DMAリクエストは生成されません

1：有効X1バッファがフルではない間、DMAリクエストが生成されます。

このビットは、FMAC_PARAMレジスタのSTART = 0の場合にのみ修正できます。読み出すと、ビットの現在の状態が返されます。

ビット8 **DMAREN**：DMAリードチャネルリクエストの有効化

0：無効DMAリクエストは生成されません

1：有効Yバッファが空ではない間、DMAリクエストが生成されます。

このビットは、FMAC_PARAMレジスタのSTART = 0の場合にのみ修正できます。読み出すと、ビットの現在の状態が返されます。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **SATIEN**：飽和エラー割込みの有効化

0：無効飽和が検出されても割込みは生成されません。

1：有効SATフラグがセットされると、割込みリクエストが生成されます

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット3 **UNFLIEN**：アンダーフローエラー割込みの有効化

0：無効アンダーフローが検出されても割込みは生成されません。

1：有効UNFLフラグがセットされると、割込みリクエストが生成されます

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット2 **OVFLIEN**：オーバーフローエラー割込みの有効化

0：無効オーバーフローが検出されても割込みは生成されません。

1：有効OVFLフラグがセットされると、割込みリクエストが生成されます

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット1 **WIEN**：書込み割込みの有効化

0：無効書込み割込みリクエストは生成されません。

1：有効X1バッファ FULLフラグがセットされていない場合、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット0 **RIEN**：読出し割込みの有効化

0：無効読出し割込みリクエストは生成されません。

1：有効Yバッファ EMPTYフラグがセットされていない場合、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

26.4.6 FMACステータス･レジスタ（FMAC_SR）

アドレスオフセット：0x14

リセット値：0x0000 0001

アクセス：ワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SAT	UNFL	OVFL	Res.	Res.	Res.	Res.	Res.	Res.	X1 FULL	Y EMPTY
					r	r	r							r	r

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **SAT**：飽和エラーフラグ

飽和は、演算の結果がアキュムレータの数値範囲を超えたときに発生します。

0：飽和は検出されていません。

1：飽和が検出されています。SATIENビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット9 **UNFL**：アンダーフローエラーフラグ

Yバッファに有効なデータが存在しないときにFMAC_RDATAから読出しを行うとアンダーフローが発生します。

0：アンダーフローは検出されていません。

1：アンダーフローが検出されています。UNFLIENビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット8 **OVFL**：オーバーフローフローエラーフラグ

X1バッファに空き空間が存在しないときにFMAC_WDATAへの書込みを行うとオーバーフローが発生します。

0：オーバーフローは検出されていません。

1：オーバーフローが検出されています。OVFLIENビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **X1FULL**：X1バッファフルフラグ

利用可能な空間の数がFULL_WM閾値未満の場合に、バッファにフルフラグが立ちます。利用可能な空間の数は、ライトポインタと現在使用中のサンプルの中で最も古いものとの差となります。

0：X1バッファはフルではありません。WIENビットがセットされていると、フラグがセットされるまで割込みリクエストがアサートされます。DMAWENがセットされていると、フラグがセットされるまでDMA書込みチャネル·リクエストが生成されます。

1：X1バッファはフルです。

このフラグはハードウェアまたはリセットによってクリアされます。

注： X1バッファの中で利用可能な最後の空間がフィルされた後、X1FULLフラグがハイになるまでに、3クロックサイクルの遅延があります。オーバーフローのリスクを回避するために、X1バッファに書き込んでからFMAC_SRを読み出すまでの間にソフトウェアディレイを挿入することをお勧めします。その代わりに、FULL_WM閾値を2とすることもできます。

ビット0 **YEMPTY**：Yバッファエンプティフラグ

未読データの数がEMPTY_WM閾値未満の場合に、バッファにエンプティフラグが立ちます。未読データの数は、リードポインタと現在の出力転送先アドレスとの差となります。

0：Yバッファは空ではありません。RIENビットがセットされていると、フラグがセットされるまで割込みリクエストがアサートされます。DMARENがセットされていると、フラグがセットされるまでDMA読出しチャネル·リクエストが生成されます。

1：Yバッファは空です。

このフラグはハードウェアまたはリセットによってクリアされます。

注： Yバッファから最後のサンプルが読み出された後、YEMPTYフラグがハイになるまでに、3クロックサイクルの遅延があります。アンダーフローのリスクを回避するために、Yバッファから読み出してからFMAC_SRを読み出すまでの間にソフトウェアディレイを挿入することをお勧めします。その代わりに、EMPTY_WM閾値を2とすることもできます。

26.4.7 FMACライト･データ･レジスタ（FMAC_WDATA）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：ワード、およびハーフワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDATA[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **WDATA[15:0]**：書込みデータ

このレジスタへのライト･アクセスが発生すると、ライト･ポインタで示されているアドレス･オフセットにライト･データが転送されます。ポインタアドレスは、毎回のライトアクセスの後に自動的にインクリメントされます。

26.4.8 FMACリード･データ･レジスタ（FMAC_RDATA）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ワード、およびハーフワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RDATA[15:0]**：読出しデータ

このレジスタへのリードアクセスが発生すると、リードデータはREADポインタで示されているアドレスオフセットのY出力バッファの内容となります。ポインタアドレスは、毎回のリードアクセスの後に自動的にインクリメントされます。

26.4.9 FMACレジスタマップ

表210.FMACレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMAC_X1BUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	FULL_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	X1_BUF_SIZE [7:0]								X1_BASE [7:0]							
	リセット値							0	0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	FMAC_X2BUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	X2_BUF_SIZE [7:0]								X2_BASE [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FMAC_YBUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Y_BUF_SIZE[7:0]								Y_BASE[7:0]							
	リセット値							0	0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FMAC_PARAM	START	FUNC[6:0]							R[7:0]								Q[7:0]								P[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FMAC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	リセット	CLIPEN	Res.	Res.	Res.	Res.	Res.	DMAWEN	DMAREN	Res.	Res.	Res.	SATIEN	UNFLIEN	OVFLIEN	WIEN	RIEN
	リセット値																0	0						0	0				0	0	0	0	0
0x14	FMAC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAT	UNFL	OVFL	Res.	Res.	Res.	Res.	Res.	Res.	X1FULL	YEMPTY
	リセット値																						0	0	0							0	1
0x18	FMAC_WDATA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	FMAC_RDATA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

27 フレキシブル･スタティック･メモリ･コントローラ（FSMC）

このセクションは、STM32U575/585/59x/5Ax/5Fx/5Gxデバイスにのみ適用されます。

27.1 概要

フレキシブルスタティックメモリコントローラ（FSMC）には、次の2つのメモリコントローラが含まれています。

- NOR/PSRAMメモリコントローラ
- NANDメモリコントローラ

このメモリ･コントローラはフレキシブル･メモリ･コントローラ（FMC）とも呼ばれています。

27.2 FMC の主な特徴

FMC機能ブロックは、同期および非同期のスタティックメモリ、NAND Flashメモリとのインタフェースを構成します。主要目的は、

- AHBトランザクションを該当する外部デバイスプロトコルへ翻訳すること
- 外部メモリデバイスのアクセス時間条件を満たすようにすること

すべての外部メモリが、アドレス信号、データ信号、制御信号をコントローラと共用します。各外部デバイスは、一意なチップセレクトによってアクセスされます。FMCは外部デバイスへ一度に1回だけアクセスします。

FMCコントローラの主要機能は次の通りです。

- 次のスタティックメモリにマップされたデバイスとのインタフェース
  - スタティックランダムアクセスメモリ（SRAM）
  - NOR型Flashメモリ/OneNAND Flashメモリ
  - PSRAM（4メモリバンク）
  - 強誘電性RAM（FRAM）
  - 最大8KBのデータをチェックできるECCハードウェア付きNAND Flashメモリ
- Intel 8080およびモトローラ6800モードをサポートしている並列LCDモジュール用インタフェース
- NOR Flashメモリ、PSRAMなどの同期デバイスに対するアクセスを高速化するバーストモードのサポート
- 非同期アクセスと同期アクセス用のプログラマブルな連続クロック出力
- 8bit、16bit幅のデータバス
- メモリバンクごとに独立したチップセレクト制御
- メモリバンクごとに独立した設定
- PSRAM、SRAMデバイスで使用する書込みイネーブルとバイトレーン選択出力
- 外部非同期ウェイト制御
- 16x32bitの深さの書込みFIFO

書込みFIFOはすべてのメモリコントローラに共通であり、以下の構成となっています。

- 書込みデータFIFOは、メモリへ書き込むAHBデータ（最大32bit）と、AHB転送（バーストモードまたは非シーケンシャルモード）用の1bitを格納

- 書込みアドレスFIFOは、AHBアドレス（最大28bit）とAHBデータサイズ（最大2bit）を格納バーストモードで動作する場合、ページ境界（PSRAM）を跨ぐとき以外は、開始アドレスのみを格納この場合、AHBバーストは2つのFIFOエントリに分割されます。

開始時に、ユーザアプリケーションからFMCピンを設定する必要があります。アプリケーションで使用しないFMCI/Oピンは、他の用途に使用することができます。

外部デバイスタイプおよびその特性を定義するFMCレジスタは、通常、起動時に設定され、次のリセットまたは電源投入まで変更されません。

ただし、以下のいくつかのビットだけなら動作中に変更可能です。

- FMC_BCRxレジスタのMBKEN、FMCEN、WENビット
- FMC_PCRレジスタのECCENおよびPBKENビット
- FMC_SRレジスタのIFS、IRS、およびILSビット

以下のシーケンスに従って、FMCの有効時にパラメータを変更します。

1. 最初に、FMCコントローラを無効にすることで、レジスタの変更中はそれ以上メモリコントローラへアクセスできないようにします。
2. すべての必要な設定を更新します。
3. FMCコントローラを再び有効にします。

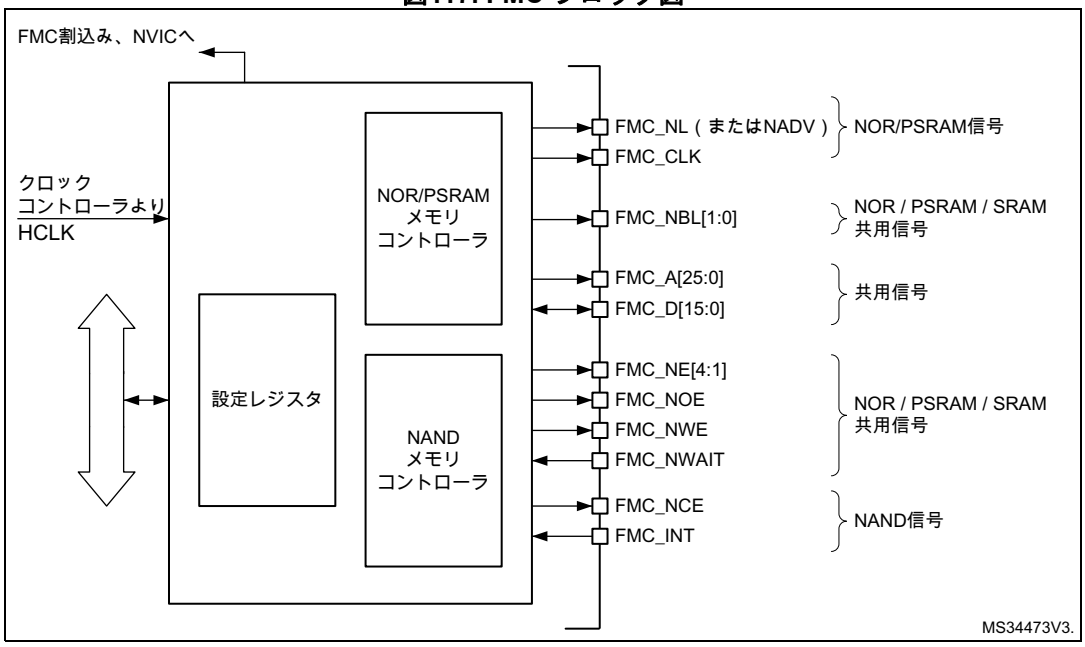
27.3 FMCのブロック図

FMCは次のメインブロックで構成されます。

- AHBインタフェース（FMC設定レジスタを含む）
- NOR Flash/PSRAM/SRAMコントローラ

以下にブロック図を示します。

図117. FMC ブロック図



27.4 AHBインタフェース

AHBスレーブインタフェースは、内部CPUおよび他のバスマスタペリフェラルから外部スタティックメモリへのアクセスを可能にします。

AHBトランザクションは、外部デバイスプロトコルに変換されます。特に、選択された外部メモリが16または8bit幅の場合、AHB上の32bit幅のトランザクションは、16または8bitの連続した複数のアクセスに分割されます。FMCチップセレクト（FMC_NEx）は、拡張モードが有効になっている際にアクセスモードDの場合を除いて、複数の連続アクセス間にトグルしません。

FMCは、以下の条件下でAHBエラーを生成します。

- 有効化してないFMCバンク（バンク1～4）の読出しまたは書込みを行ったとき。
- FMC_BCRxレジスタのFACCENビットがリセットされている状態でNOR Flashバンクの読出しまたは書込みを行ったとき。

このAHBエラーの影響は、読出し／書込みアクセスを試みたAHBマスタに依存します。

- アクセスがCortex®-M33CPUから行われた場合は、ハードフォールト割込みが発生します。
- アクセスが DMA コントローラから行われた場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効化されます。

AHBクロック（HCLK）は、FMCのリファレンスクロックです。

27.4.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

リクエストされるAHBトランザクションのデータサイズは、アクセスされる外部デバイスのデータ幅が固定であっても、8、16、または32bit幅です。このため、転送に矛盾が生じることがあります。

したがって、次のようないくつかの単純なトランザクション規則を守る必要があります。

- AHBトランザクションサイズとメモリデータサイズが等しい場合：

  この場合は問題ありません。
- AHBトランザクションサイズがメモリサイズより大きい場合：

  この場合、FMCは、AHBトランザクションを外部データ幅に合わせて、連続した小さなメモリアクセスに分割します。FMCチップセレクト（FMC_NEx）は、複数の連続アクセス間にトグルしません。バスターンアラウンドタイミングが0以外の値に設定されている場合は、FMCチップセレクト（FMC_NEx）は複数の連続アクセス間にトグルします。この機能は、FRAMメモリとインタフェースするときに必要です。
- AHBトランザクションサイズがメモリサイズより小さい場合：

  外部デバイスのタイプによっては、転送に一貫性がなくなる場合があります：
  - バイト選択機能を持つデバイス（SRAM、ROM、PSRAM）に対するアクセス

    この場合、FMCは、読出し／書込みトランザクションを許可して、バイト･レーンNBL[1:0]を通じて正しいデータにアクセスします。

    書込み対象バイトはNBL[1:0]によりアドレス指定されます。

    すべてのメモリバイトが読み出され（読出しトランザクション中NBL[1:0]はローに駆動されます）、不要なバイトは無視されます。
  - バイト選択機能を持たないデバイス（NORおよびNAND Flashメモリ）に対するアクセス

    この状況は、16bit幅のFlashメモリへのバイト･アクセスが要求されたときに発生します。バイト･モードでデバイスにアクセスできないため（Flashメモリに対しては16bitワードの読出し／書込みのみが可能）、書込みトランザクションと読出しトランザクションが可能です（コントローラは16bitメモリ･ワード全体を読み出して、必要なバイトのみを使用）。

NOR型Flash/PSRAMのラップサポート

同期メモリに対するラップバーストモードはサポートされていません。メモリは、不定長のリニアバーストモードに設定する必要があります。

設定レジスタ

FMCは、レジスタのセットとして設定することができます。NOR Flash／PSRAMコントローラ･レジスタの詳細については、セクション27.6.6を参照してください。NAND Flashレジスタの詳細については、セクション27.7.7を参照してください。

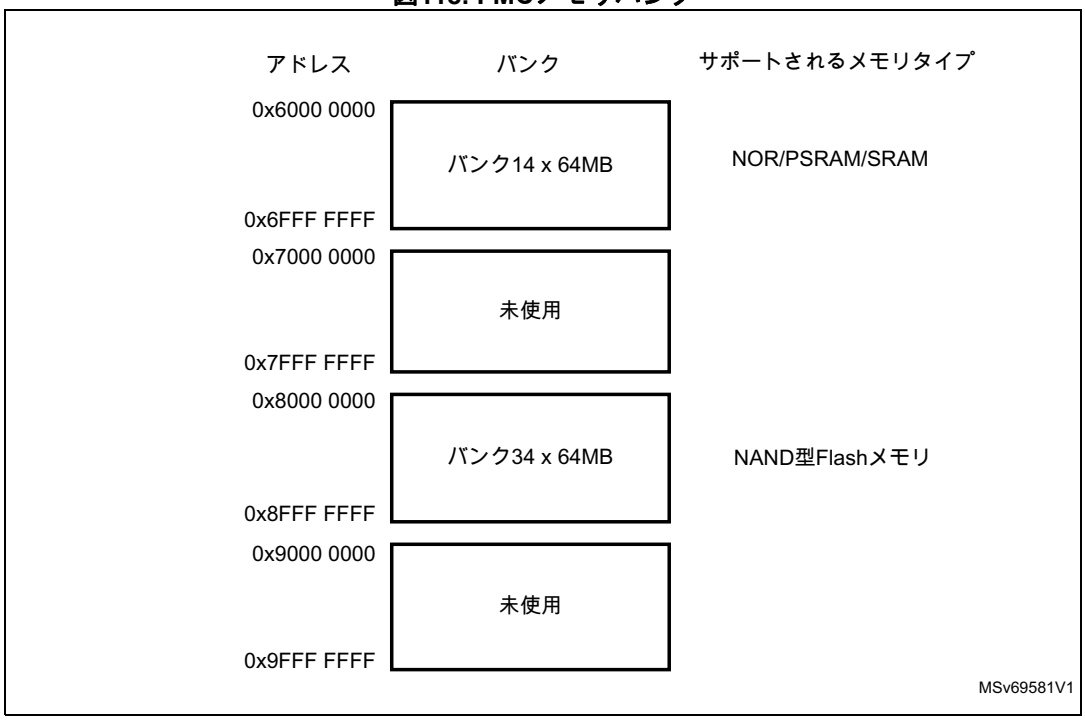
27.5 外部デバイスアドレスマッピング

FMCから見ると、外部メモリは、それぞれ256MBの固定サイズのバンクに分けられています（図118を参照）。

- バンク1は、4つまでのNOR FlashメモリまたはPSRAMメモリデバイスのアドレス指定に使用されます。このバンクは、次のように4つの専用チップセレクトを持つ4つのNOR/PSRAMサブバンクに分割されます。
  - バンク1 - NOR/PSRAM 1
  - バンク1 - NOR/PSRAM 2
  - バンク1 - NOR/PSRAM 3
  - バンク1 - NOR/PSRAM 4
- バンク3は、NAND Flashメモリデバイスのアドレス指定に使用されます。この空間のMPUメモリ属性をソフトウェアによってデバイスに再設定する必要があります。

各バンクで使用されるメモリのタイプは、ユーザアプリケーションから設定レジスタを使って設定することができます。

図118. FMCメモリバンク



27.5.1 NOR/PSRAMアドレスマッピング

HADDR[27:26]ビットは、表211に示される4つのメモリバンクの1つを選択するために使用されます。

表211.NOR/PSRAMバンク選択

HADDR[27:26](1)	選択されるバンク
00	バンク1 - NOR/PSRAM 1
01	バンク1 - NOR/PSRAM 2
10	バンク1 - NOR/PSRAM 3
11	バンク1 - NOR/PSRAM 4

1. HADDRは、外部メモリに変換される内部AHBアドレスラインです。

HADDR[25:0]bitビットは、外部メモリアドレスを含みます。HADDRbitはバイトアドレスですが、メモリはワードレベルでアクセスされるので、メモリに対して実際に発行されるアドレスは、次の表に示されるように、メモリのデータ幅に応じて変わります。

表212.NOR/PSRAMbit外部メモリアドレス

メモリ幅(1)	メモリに発行されるデータアドレス	最大メモリ容量（ビット）
8bit	HADDR[25:0]	64bit MB × 8 = 512 M bit
16bit	HADDR[25:1] >> 1	64bit MB/2 × 16 = 512 M bit

1. 外部メモリがbit16bit幅の場合、FMCbitは、内部でbitHADDR[25:1]bitを使用して、外部メモリbitFMC_A[24:0]bitに対するアドレスを生成します。
外部メモリの幅にかかわらず、FMC_A[0]bitは外部メモリアドレスbitA[0]bitに接続されなくてはなりません。

27.5.2 NAND Flashメモリ･アドレス･マッピング

NANDバンクは、表213に示されるメモリ領域に分割されます。

表213.NANDメモリマッピングおよびタイミングレジスタ

開始アドレス	終了アドレス	FMCバンク	メモリ空間	タイミングレジスタ
0x8800 0000	0x8BFF FFFF	バンク3 - NAND Flash	属性	FMC_PATT（0x8C）
0x8000 0000	0x83FF FFFF		共有	FMC_PMEM（0x88）

NAND型Flashメモリでは、共通および属性メモリ空間は、下位256KBに位置する3つのセクションに分割されます（表214を参照）。

- データセクション（共通/属性メモリ空間の最初の64KB）
- コマンドセクション（共通/属性メモリ空間の2番目の64KB）
- アドレスセクション（共通/属性メモリ空間の次の128KB）

表214.NANDバンク選択

セクション名	HADDR[17:16]	アドレス範囲
アドレスセクション	1X	0x020000-0x03FFFF
コマンドセクション	01	0x010000-0x01FFFF
データセクション	00	0x000000-0x0FFFF

アプリケーション･ソフトウェアは、3つのセクションを使用して、NAND型Flashメモリにアクセスします。

- **NAND Flashメモリにコマンドを送信するときは、**ソフトウェアからコマンドセクションの任意のメモリ位置にコマンドの値を書き込む必要があります。
- **読出しまたは書込みを行う必要があるNAND Flashのアドレスを指定するには、**ソフトウェアは、アドレス･セクションの任意のメモリ位置にアドレス値を書き込む必要があります。アドレスは4または5バイト長なので（実際のメモリサイズに依存）、アドレス全体を指定するには、アドレスセクションへのいくつかの連続した書込みが必要です。
- **データの読出しまたは書込みを行うときは、**ソフトウェアは、データセクションの任意のメモリ位置からデータを読み出すか、書き込む必要があります。

NAND型Flashメモリはアドレスを自動的にインクリメントするので、連続したメモリ位置にアクセスするには、データ･セクションのアドレスをインクリメントする必要はありません。

27.6 NOR型Flash／PSRAMコントローラ

FMCは、以下のタイプのメモリを駆動するのに適した信号タイミングを発生します。

- 非同期SRAM、FRAM、およびROM
  - 8bit
  - 16bit
- PSRAM（CellularRAM™）
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス
- NOR Flashメモリ
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス

FMCは、バンクごとに1つのチップセレクト信号NE[4:1]を出力します。その他の信号（アドレス、データ、および制御）はすべて、共有されます。

FMCは、次のようなプログラム可能なタイミングにより、広範囲なデバイスをサポートしています。

- プログラム可能なウェイトステート（15まで）
- プログラム可能なバスターンアラウンドサイクル（15まで）
- プログラム可能な出力イネーブルおよび書込みイネーブル遅延（15まで）
- 独立した読出しおよび書込みタイミングとプロトコルにより、広範囲なメモリおよびタイミングをサポート
- プログラム可能な連続クロック（FMC_CLK）出力

FMCクロック（FMC_CLK）はHCLKクロックの約数です。このクロックは、同期アクセス中にのみ、またはFMC_BCR1レジスタのCCKENビット設定に応じて非同期アクセスおよび同期アクセス中に、選択した外部デバイスへ供給することができます。

- CCLKENビットがリセットされている場合、FMCは同期アクセス（読出し／書込みトランザクション）中にのみクロック（CLK）を生成します。
- CCLKENビットがセットされている場合、FMCは非同期アクセス中および同期アクセス中に連続クロックを生成します。FMC_CLK連続クロックを生成するためには、バンク1を同期モードに設定する必要があります（セクション 27.6.6：NOR/PSRAMコントローラレジスタを参照）。

すべての同期メモリに対して同じクロックが使用されるため､連続出力クロックを生成して同期アクセスを実行する場合、AHBデータサイズはメモリ･データ幅（MWID）に一致する必要があります。そうしないと、AHBデータ･トランザクションに応じて、FMC_CLK周波数が変化します（FMC_CLK分周比の式についてはセクション 27.6.5：同期トランザクションを参照）。

各バンクのサイズは64MBに固定されます。各バンクは、専用レジスタによって設定されます（セクション 27.6.6：NOR/PSRAMコントローラレジスタを参照）。

プログラム可能なメモリ･パラメータは、アクセス･タイミング（表215を参照）を含み、ウェイト管理をサポートします（バーストモードでのPSRAMおよびNOR Flashへのアクセス）。

表215.プログラム可能なNOR/PSRAMのアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	AHBクロックサイクル（HCLK）	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクスI/O	AHBクロックサイクル（HCLK）	1	15
NBLセットアップ	バイトレーンセットアップフェーズ時間	非同期	AHBクロックサイクル（HCLK）	0	3
データセットアップ	データセットアップフェーズ時間	非同期	AHBクロックサイクル（HCLK）	1	256
データホールド	データホールドフェーズ時間	非同期	AHBクロックサイクル（HCLK）	0	3
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読出し／書込み	AHBクロックサイクル（HCLK）	0	15
クロック分周比	1つのメモリクロックサイクル（CLK）に必要なAHBクロックサイクル（HCLK）数	同期	AHBクロックサイクル（HCLK）	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル（CLK）	2	17

27.6.1 外部メモリインタフェース信号

表216、表217および表218に、NOR Flashメモリ、SRAM、およびPSRAMへのインタフェースで一般に使用される信号を示します。

注： **接頭辞「N」は、アクティブローの信号を表します。**

NOR Flashメモリ、非マルチプレクスI/O

表216.非マルチプレクスI/O NOR Flashメモリ

FMC信号名	I/O	機能
CLK	O	クロック（同期アクセス用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル

表216.非マルチプレクスI/O NOR Flashメモリ （続き）

FMC信号名	I/O	機能
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチ･イネーブル（この信号は、一部のNOR型Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMCへのNOR型Flashウェイト入力信号

最大容量は、512Mビット（26アドレスライン）です。

NOR Flashメモリ、16bitマルチプレクスI/O

表217.16bitマルチプレクスI/O NOR Flashメモリ

FMC信号名	I/O	機能
CLK	O	クロック（同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16bitマルチプレクス、双方向アドレス／データバス（16bitアドレスA[15:0]とデータD[15:0]がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチ･イネーブル（この信号は、一部のNOR型Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMCへのNOR型Flashウェイト入力信号

最大容量は512Mビットです。

PSRAM/FRAM/SRAM、非マルチプレクスI/O

表218.非マルチプレクスI/O PSRAM/SRAM

FMC 信号名	I/O	機能
CLK	O	クロック（PSRAM同期アクセス専用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4（PSRAM（CellularRAM™、すなわちCRAM）ではNCEと呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	PSRAM入力の場合のみアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMCへのPSRAMウェイト入力信号
NBL[1:0]	O	バイトレーン出力バイト0およびバイト1制御（上位および下位バイトイネーブル）

最大容量は512 Mbitです。

PSRAM、16bitマルチプレクスI/O

表219.16bitマルチプレクスI/O PSRAM

FMC信号名	I/O	機能
CLK	O	クロック（同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16bitマルチプレクス、双方向アドレス／データバス（16bitアドレスA[15:0]とデータD[15:0]がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4（PSRAM（CellularRAM™、すなわちCRAM）ではNCEと呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	PSRAM入力のアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMCへのPSRAMウェイト入力信号
NBL[1:0]	O	バイトレーン出力バイト0およびバイト1制御（上位および下位バイトイネーブル）

最大容量は、512 Mbit（26アドレスライン）です。

27.6.2 サポートされるメモリおよびトランザクション

以下の表220に、NOR Flashメモリ、PSRAM、SRAMに対してメモリデータバスが16bit幅の場合にサポートされているデバイス、アクセスモード、トランザクションの例を示します。FMCでは許容されない（すなわち非サポートの）トランザクションは、この例では灰色で表示されています。

表220.NOR Flash／PSRAM：サポートされているメモリおよびトランザクションの例

デバイス	モード	読出し／書込み	AHBデータサイズ	メモリデータサイズ	許可／禁止	コメント
NOR型Flash（マルチプレクスI/Oと非マルチプレクスI/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2つのFMCアクセスに分割
	非同期	W	32	16	Y	2つのFMCアクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-

表220.NOR Flash／PSRAM：サポートされているメモリおよびトランザクションの例 （続き）

デバイス	モード	読出し／書込み	AHBデータサイズ	メモリデータサイズ	許可／禁止	コメント
PSRAM（マルチプレクスI/Oおよび非マルチプレクスI/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	Y	バイトレーンNBL[1:0]の使用
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2つのFMCアクセスに分割
	非同期	W	32	16	Y	2つのFMCアクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-
	同期	W	8	16	Y	バイトレーンNBL[1:0]の使用
	同期	W	16/32	16	Y	-
SRAMとROM	非同期	R	8 / 16	16	Y	-
	非同期	W	8 / 16	16	Y	バイトレーンNBL[1:0]の使用
	非同期	R	32	16	Y	2つのFMCアクセスに分割
	非同期	W	32	16	Y	2つのFMCアクセスに分割、バイトレーンNBL[1:0]の使用

27.6.3 一般的なタイミング規則

信号の同期

- すべてのコントローラ出力信号は、内部クロック（HCLK）の立上がりエッジで変化します。
- 同期モード（読出しまたは書込み）では、すべての出力信号がHCLKの立上がりエッジで変化します。CLKDIVの値によらず、すべての出力は次のように変化します。
  - NOEL/NWEL/NEL/NADVL/NADVH/NBLL/アドレスの有効出力は、FMC_CLKクロックの立下がりエッジで変化します。
  - NOEH/NWEH/NEH/NOEH/NBLH/ アドレスの無効出力は、FMC_CLK クロックの立上がりエッジで変化します。

27.6.4 NOR Flash/PSRAMコントローラ非同期トランザクション

非同期スタティック･メモリ（NOR Flash、PSRAM、SRAM、FRAM）

- 信号は、内部クロックHCLKによって同期されます。このクロックはメモリには発行されません。
- FMCは、NOE信号をネゲートする前に、常にデータをサンプリングします。これにより、メモリのデータホールドタイミングに関する制約が満たされます（チップイネーブルハイからデータ遷移までの最小時間は通常0nsです）。
- 拡張モードを有効にした場合（FMC_BCRxレジスタのEXTMODビットをセット）、最大4つの拡張モード（A、B、C、D）が使用できます。書込み動作および読出し動作で、A、B、C、Dモードをミックスすることができます。例えば、読出し動作をモードAで実行し、書込み動作をモードBで実行することができます。
- 拡張モードを無効にした場合（FMC_BCRxレジスタでEXTMODビットをリセット）、FMCは次のようにモード1またはモード2で動作することができます。
  - SRAM/PSRAMメモリタイプを選択した場合（FMC_BCRxレジスタでMTYP = 0x0または0x01）、モード1がデフォルトモードです。
  - NORメモリタイプを選択した場合（FMC_BCRxレジスタでMTYP = 0x10）、モード2がデフォルトモードです。

モード1 - SRAM/FRAM/PSRAM（CRAM）

次の図に、サポートされているモードに対する読出しトランザクションと書込みトランザクションを示します。それに続いて、FMC_BCRxレジスタ、およびFMC_BTRx/FMC_BWTRxレジスタの必要とされる設定を示します。

図119. モード1読出しアクセス波形

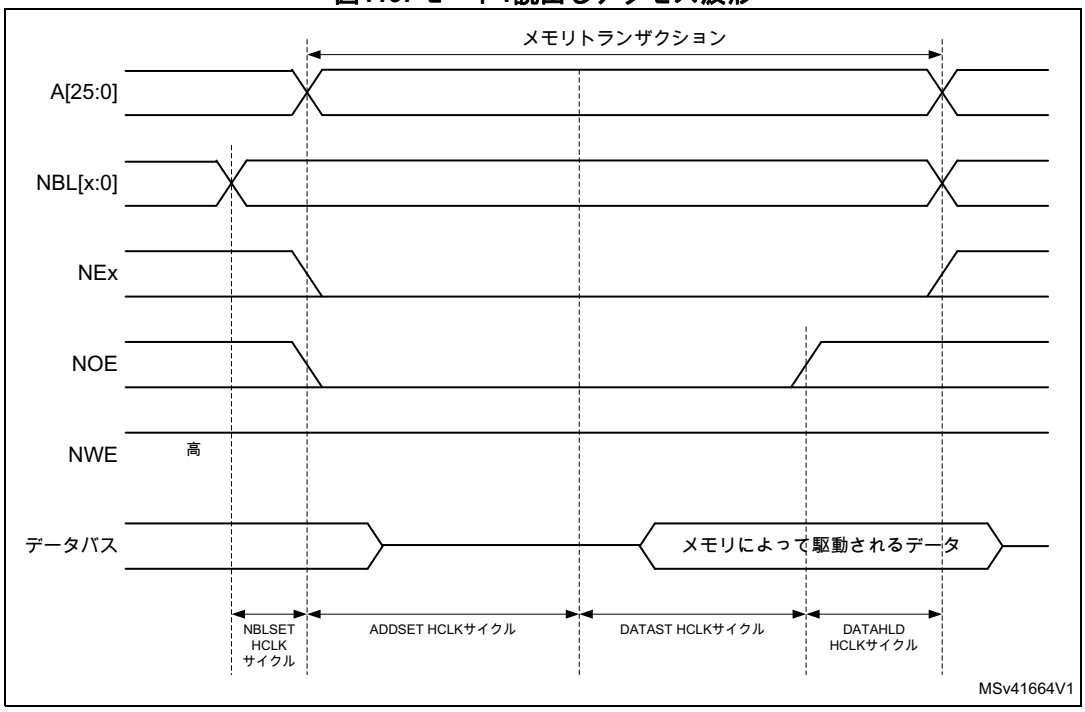
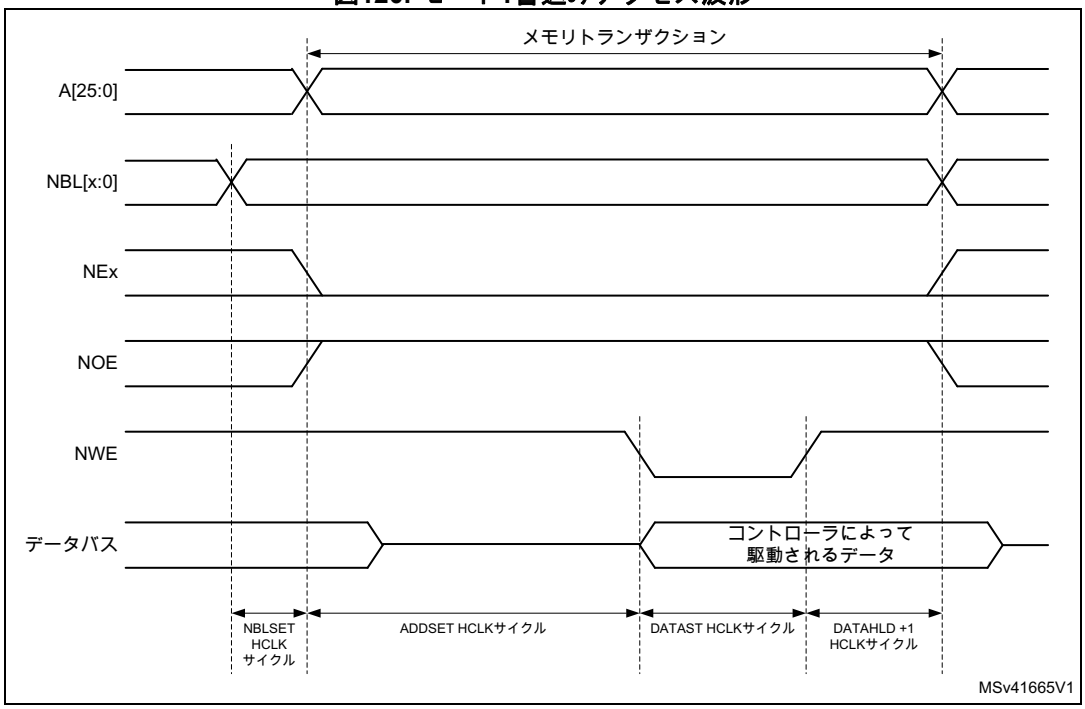


図120. モード1書込みアクセス波形



読出しおよび書込みトランザクションの最後のDATAHLD時間は、NOE/NWEの立上がりエッジ後のアドレスおよびデータホールド時間を保証します。DATASTの値はゼロより大きくなければなりません（DATAST ＞ 0）。

表221.FMC_BCRxビットフィールド（モード1）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。

表221.FMC_BCRxビットフィールド（モード1）（続き）

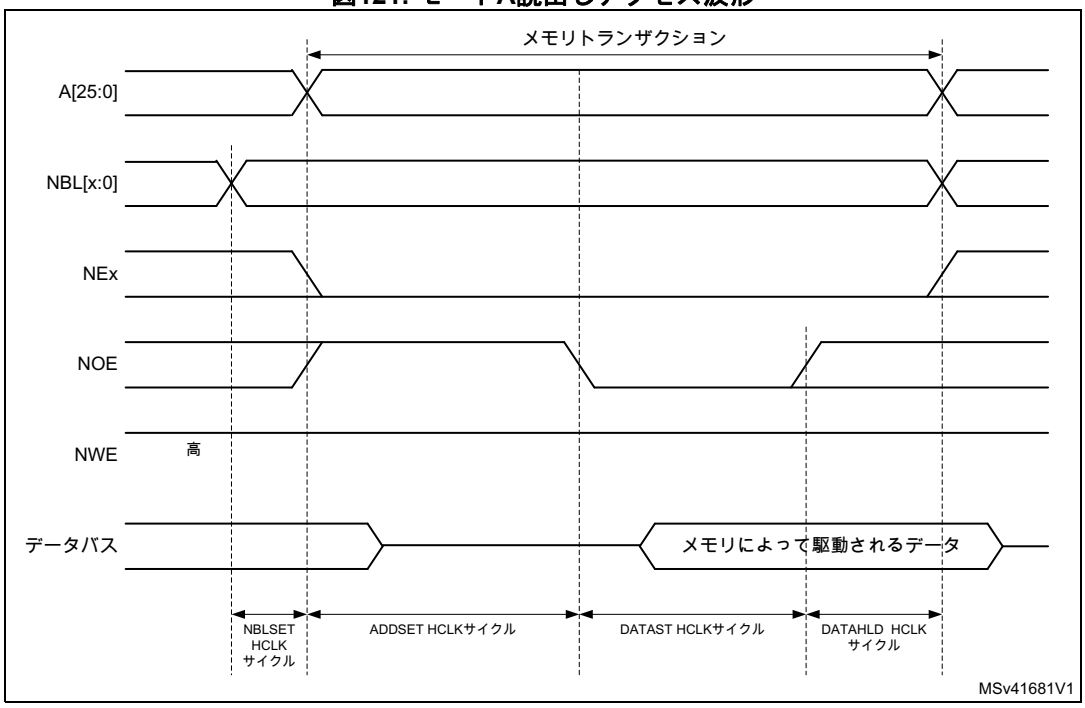
ビット番号	ビット名	設定値
3:2	MTYP	必要に応じて、0x2（NOR Flashメモリ）を除きます。
1	MUXE	0x0
0	MBKEN	0x1

表222.FMC_BTRxビットフィールド（モード1）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合はDATAHLD HCLKサイクル、書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	無視
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。 ADDSETの最小値は0。

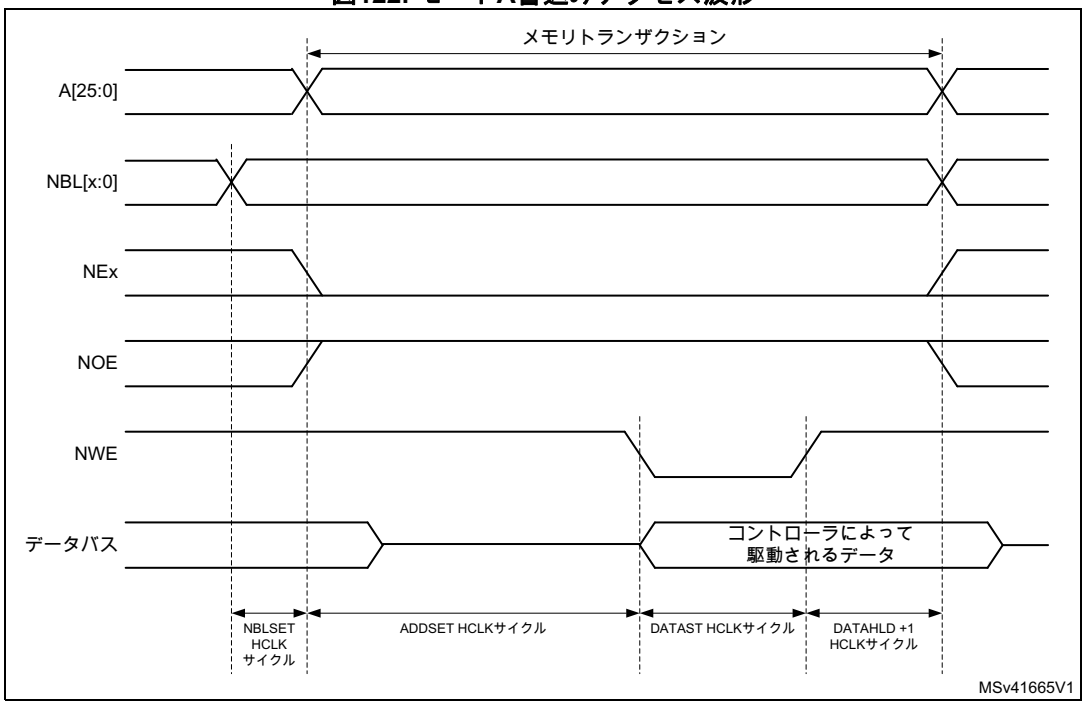
モードA - SRAM/FRAM/PSRAM（CRAM）OEトグル

図121. モードA読出しアクセス波形



1. NBL[1:0]は読出しアクセス中ローレベルに駆動されます。

図122. モードA書込みアクセス波形



モード1と比較した場合の相違点は、NOEのトグリングと、独立した読出しおよび書込みのタイミングです。

表223.FMC_BCRxビットフィールド（モードA）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。

表223.FMC_BCRxビットフィールド（モードA）（続き）

ビット番号	ビット名	設定値
3:2	MTYP	必要に応じて、0x2（NOR Flashメモリ）を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

表224.FMC_BTRxビットフィールド（モードA）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合はDATAHLD HCLKサイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。 ADDSETの最小値は0。

表225.FMC_BWTRxビットフィールド（モードA）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。 ADDSETの最小値は0。

モード2/B - NOR型Flash

図123. モード2およびモードB読出しアクセス波形

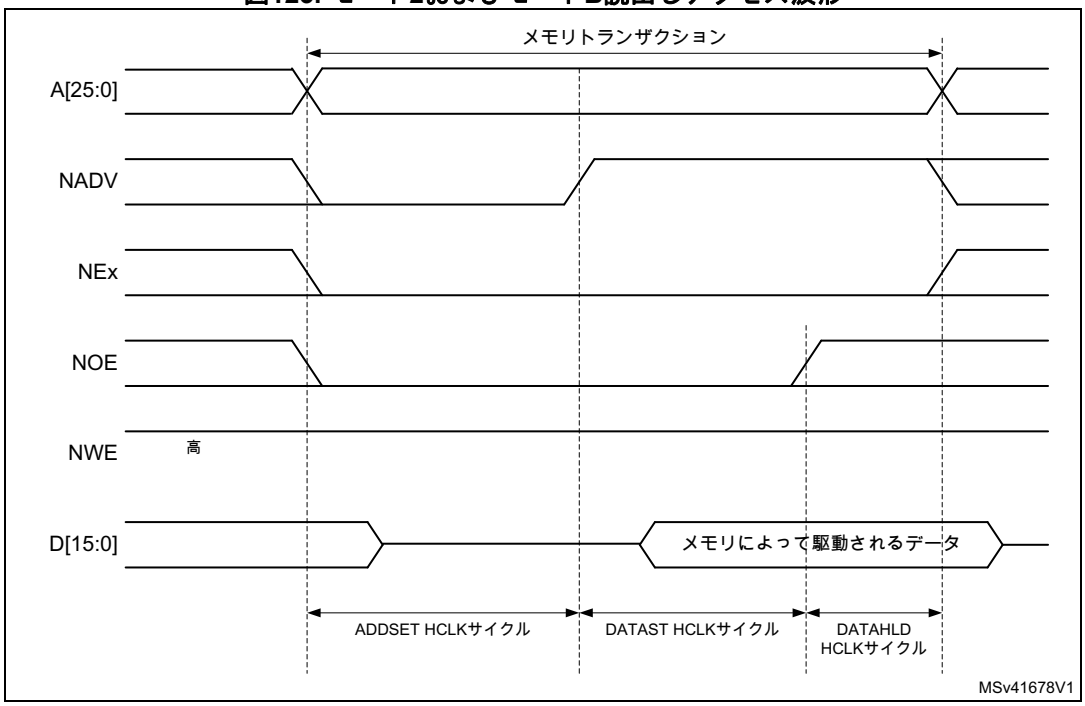


図124. モード2書込みアクセス波形

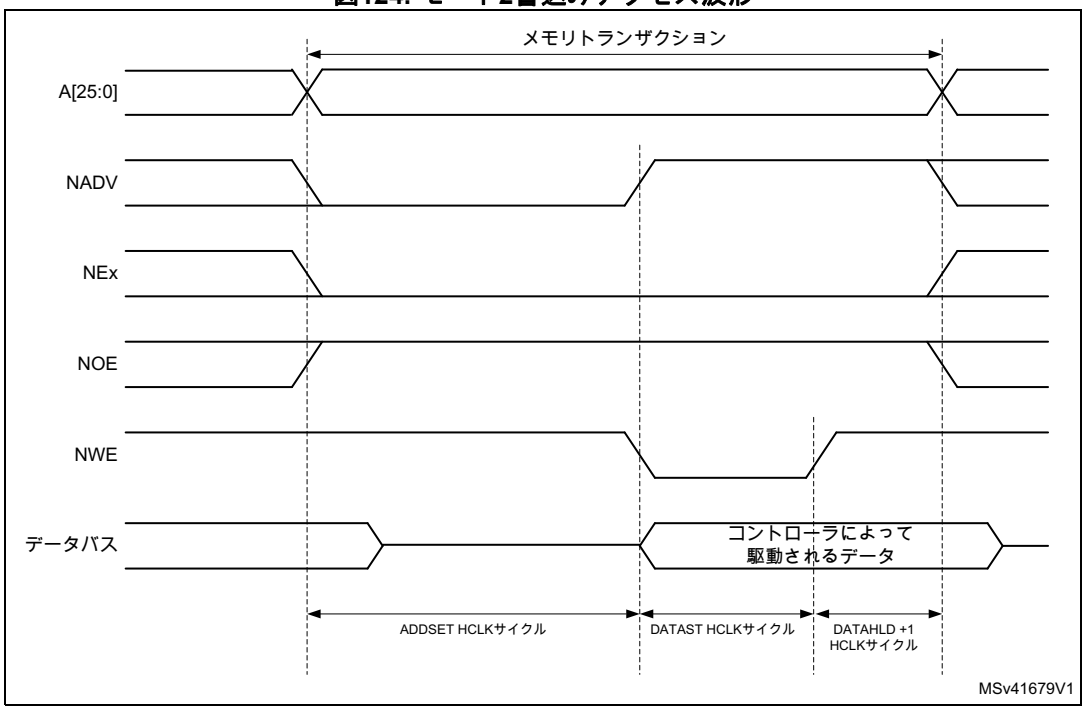
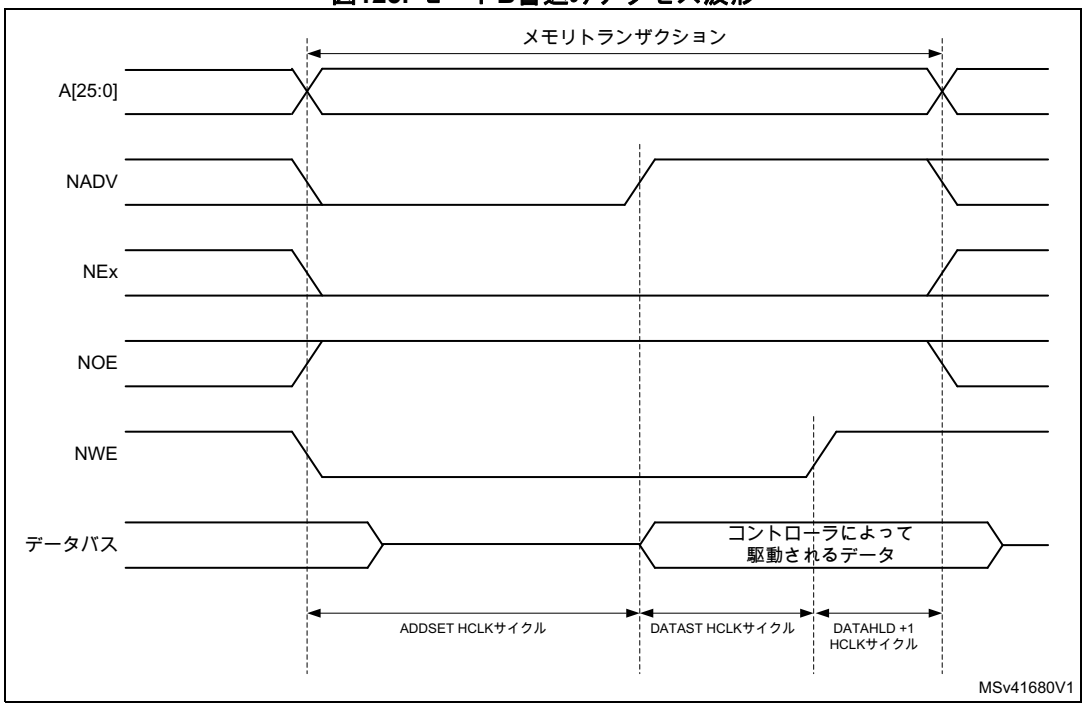


図125. モードB書込みアクセス波形



モード1と比較した場合の相違点は、NWEのトグリングと拡張モードがセットされたとき（モードB）の独立した読出しおよび書込みのタイミングです。

表226.FMC_BCRxビットフィールド（モード2/B）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	モードBでは0x1、モード2では0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。

表226.FMC_BCRxビットフィールド（モード2/B）（続き）

ビット番号	ビット名	設定値
3:2	MTYP	0x2（NOR型Flashメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表227.FMC_BTRxビットフィールド（モード2/B）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（拡張モードが無効のとき、読出しアクセスの場合はDATAHLD HCLKサイクル、書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	拡張モードがセットされている場合は0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しアクセスに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は0。

表228.FMC_BWTRxビットフィールド（モード2/B）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	拡張モードがセットされている場合は0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みアクセスに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は0。

注： **FMC_BWTRxレジスタは、拡張モードがセットされている場合（モードB）のみ有効です。それ以外の場合には、その内容は意味を持ちません。**

モードC - NOR Flash - OEトグリング

図126. モードC読出しアクセス波形

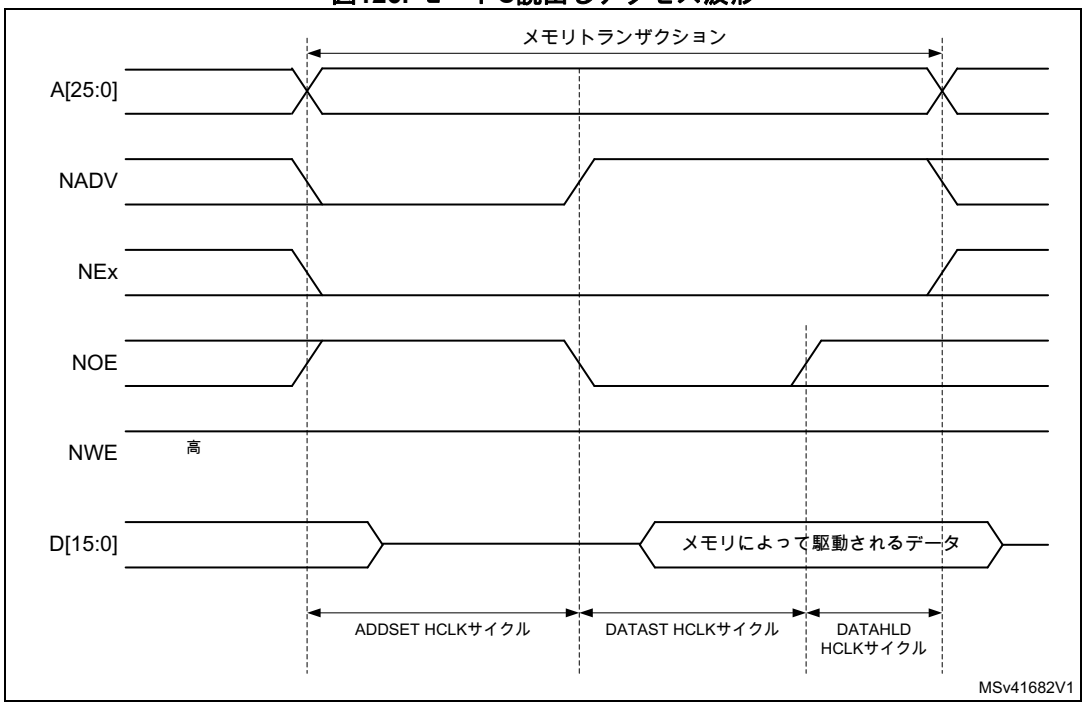
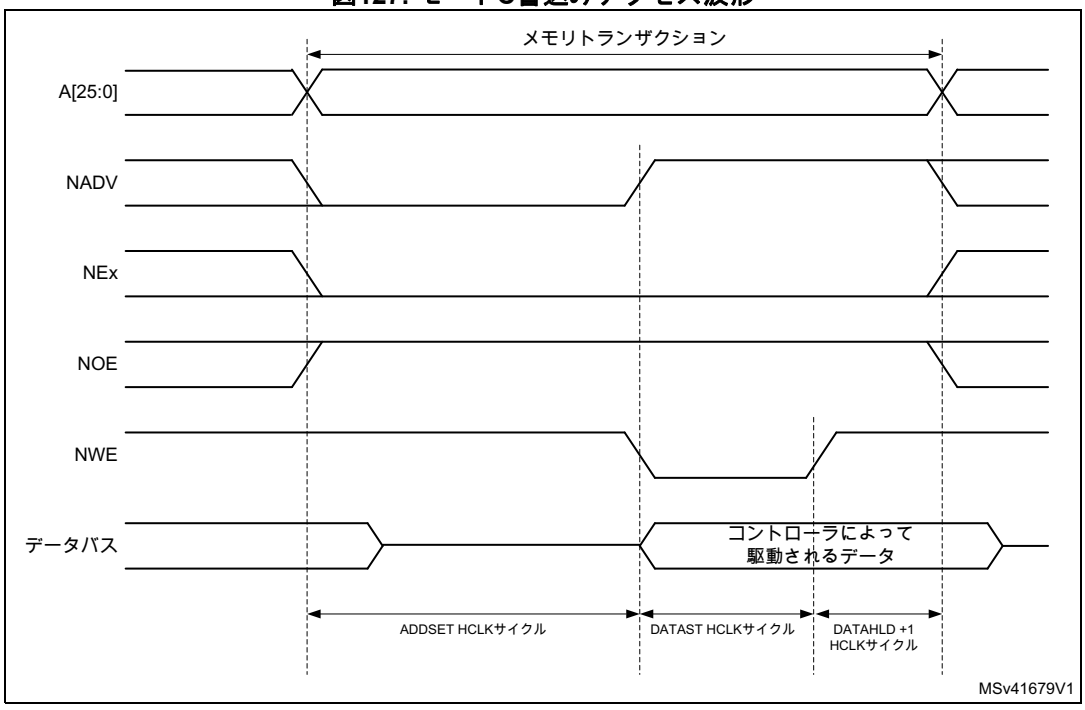


図127. モードC書込みアクセス波形



モード1と比較した場合の相違点は、NOEのトグリングと、独立した読出しおよび書込みのタイミングです。

表229.FMC_BCRxビットフィールド（モードC）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x02（NOR型Flashメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表230.FMC_BTRxビットフィールド（モードC）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合はDATAHLD HCLKサイクル）。
29:28	ACCMOD	0x2
27:24	DATLAT	0x0
23:20	CLKDIV	0x0
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は0。

表231.FMC_BWTRxビットフィールド（モードC）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	0x2
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は0。

モードD - 拡張アドレスによる非同期アクセス

図128. モードD読出しアクセス波形

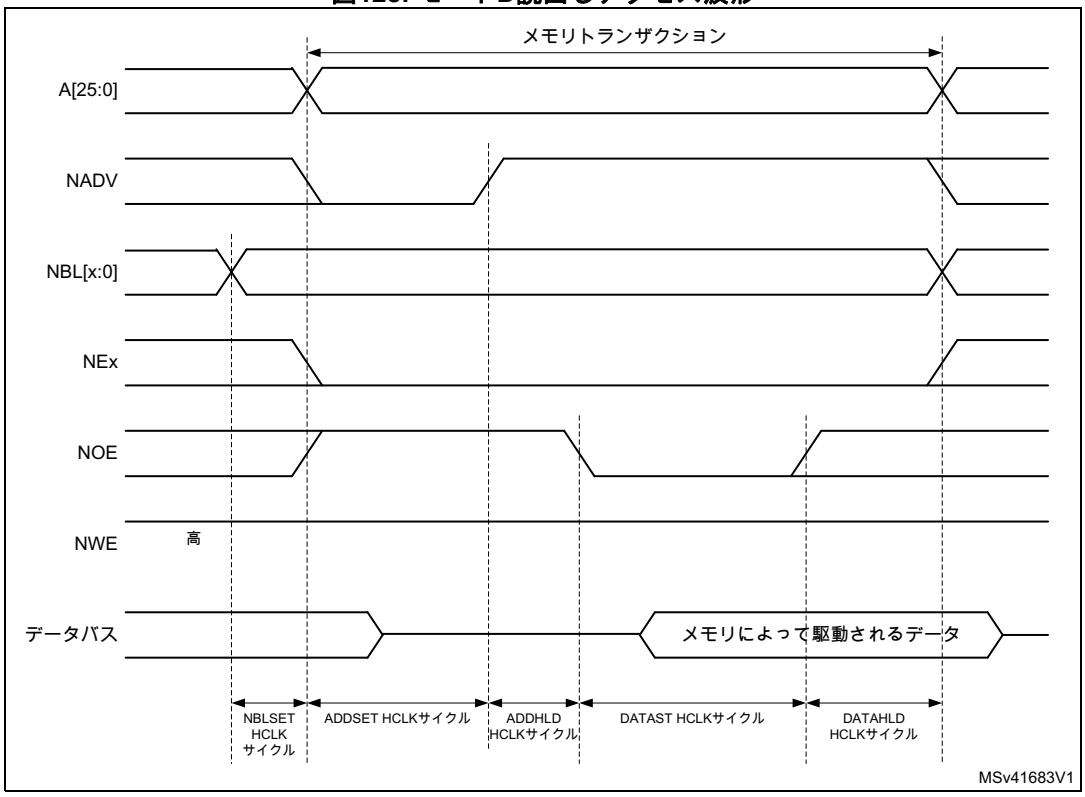
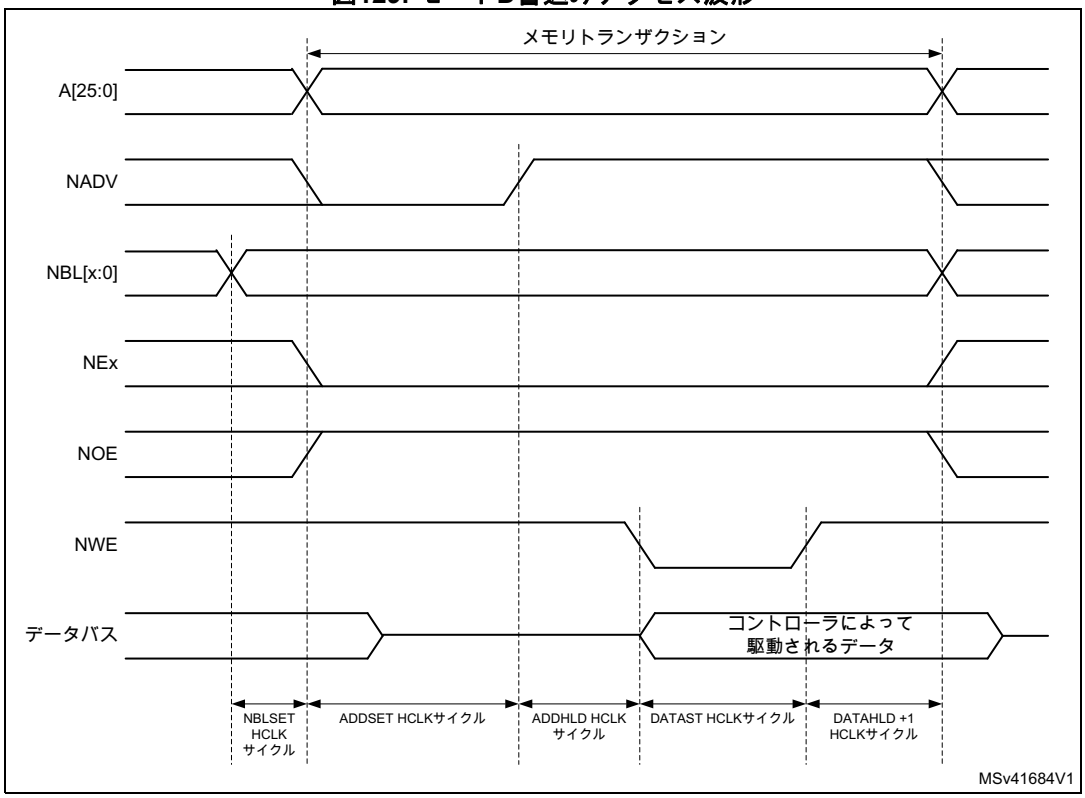


図129. モードD書込みアクセス波形



モード1と比較した場合の相違点は、NADVが変化した後にトグルを続けるNOEのトグルと、独立した読出し書込みタイミングです。

表232.FMC_BCRxビットフィールド（モードD）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1

表232.FMC_BCRxビットフィールド（モードD）（続き）

ビット番号	ビット名	設定値
6	FACCEN	メモリのサポート状況に応じて設定します。
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。
1	MUXEN	0x0
0	MBKEN	0x1

表233.FMC_BTRxビットフィールド（モードD）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合はDATAHLD HCLKサイクル）。
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しに対する2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	読出しにおける中間アクセスフェーズの時間（ADDHLD HCLKサイクル）
3:0	ADDSET	読出しに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は1。

表234.FMC_BWTRxビットフィールド（モードD）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	書込みアクセスにおける中間アクセスフェーズの時間（ADDHLD HCLKサイクル）
3:0	ADDSET	書込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は1。

マルチプレクス･モード - NOR型Flashメモリに対するマルチプレクス非同期アクセス

図130. マルチプレクス読出しアクセス波形

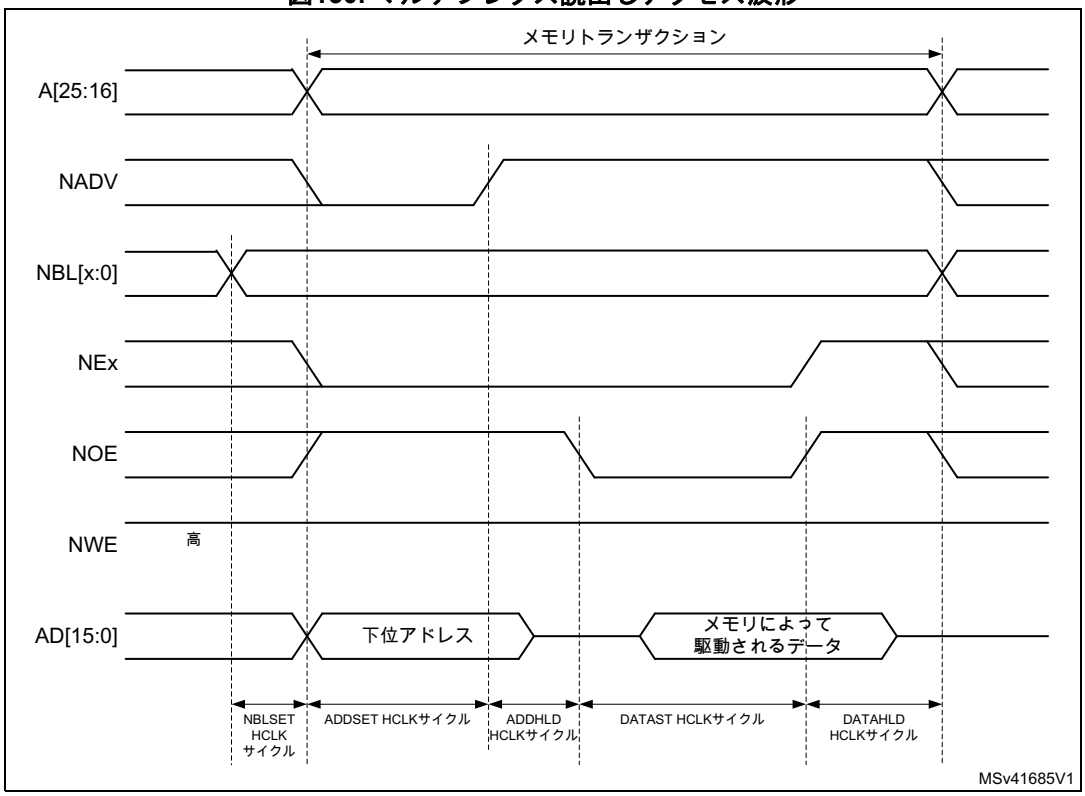
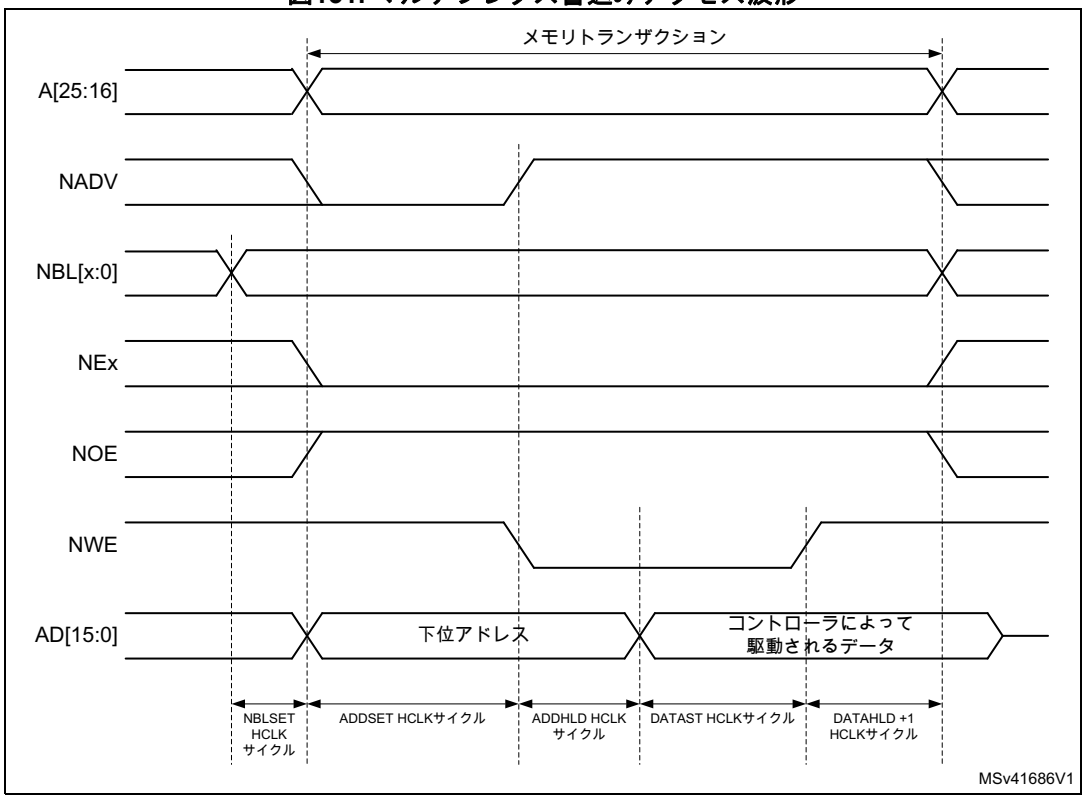


図131. マルチプレクス書込みアクセス波形



モードDとの相違点は、データバス上の下位アドレスバイトのドライブです。

表235.FMC_BCRxビットフィールド（マルチプレクス･モード）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は1にセットします。そうでない場合は、0のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット15が1の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1

表235.FMC_BCRxビットフィールド（マルチプレクス･モード）（続き）

ビット番号	ビット名	設定値
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2（NOR型Flashメモリ）または0x1（PSRAM）
1	MUXEN	0x1
0	MBKEN	0x1

表236.FMC_BTRxビットフィールド（マルチプレクス･モード）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合はDATAHLD HCLKサイクル、書込みアクセスの場合はDATAHLD+1 HCLKサイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2番目のアクセスフェーズの時間（DATAST HCLKサイクル）。
7:4	ADDHLD	書込みアクセスにおける中間アクセスフェーズの時間（ADDHLD HCLKサイクル）。
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET HCLKサイクル）。ADDSETの最小値は1。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを知らせるために、WAIT信号をアサートした場合、FMC_BCRxレジスタのASYNCWAITビットをセットする必要があります。

WAIT信号がアクティブな場合（WAITPOLビットに依存して、ハイまたはロー）、DATASTビットによってプログラムされる2番目のアクセスフェーズ（データセットアップフェーズ）は、WAITビットが非アクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSETおよびADDHLDビットによってプログラムされる最初のアクセスフェーズ（アドレスセットアップおよびアドレスホールドフェーズ）は、WAITに対応しないので、延長されません。

データセットアップフェーズは、メモリトランザクションが終了する4HCLKサイクル前にWAITが検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルするNOE/NWEに合わせたWAIT信号をアサートする場合。

$$DATAST \geq (4 \times HCLK) + max_wait_assertion_time$$

2. メモリが、NEx（またはトグルしないNOE/NWE）に合わせたWAIT信号をアサートする場合。
   次の条件が成立するならば、

$$max_wait_assertion_time > address_phase + hold_phase$$

   以下の通りです。

$$DATAST \geq (4 \times HCLK) + (max_wait_assertion_time - address_phase - hold_phase)$$

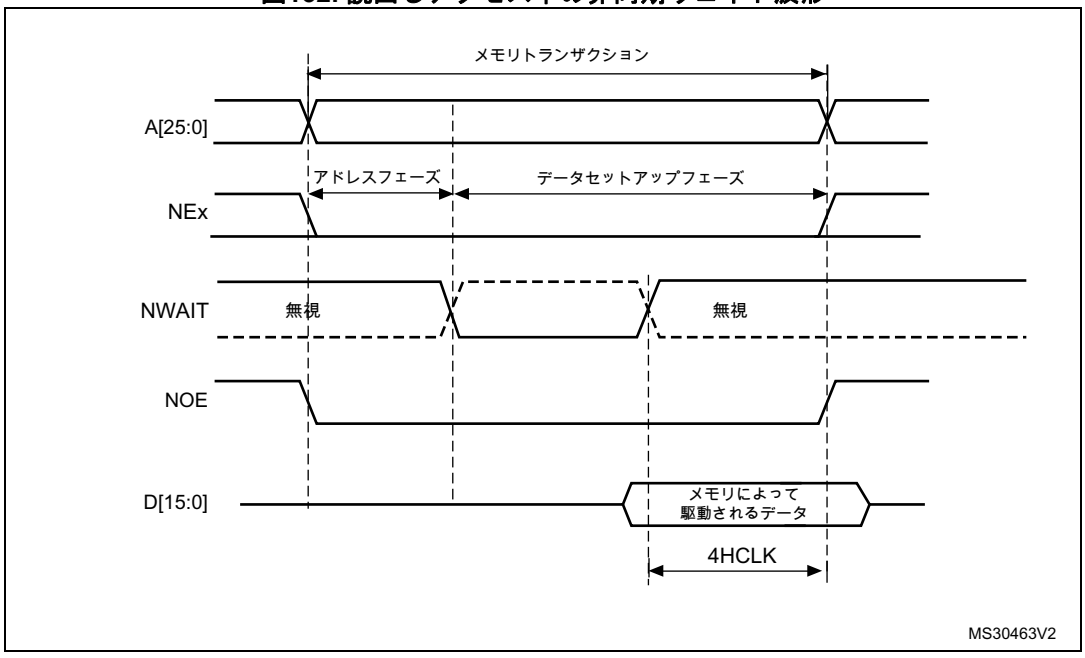
   それ以外の場合は、

$$DATAST \geq 4 \times HCLK$$

   ここで、max_wait_assertion_timeは、NEx/NOE/NWEがローになったときにメモリがWAIT信号をアサートするために必要な最大時間です。

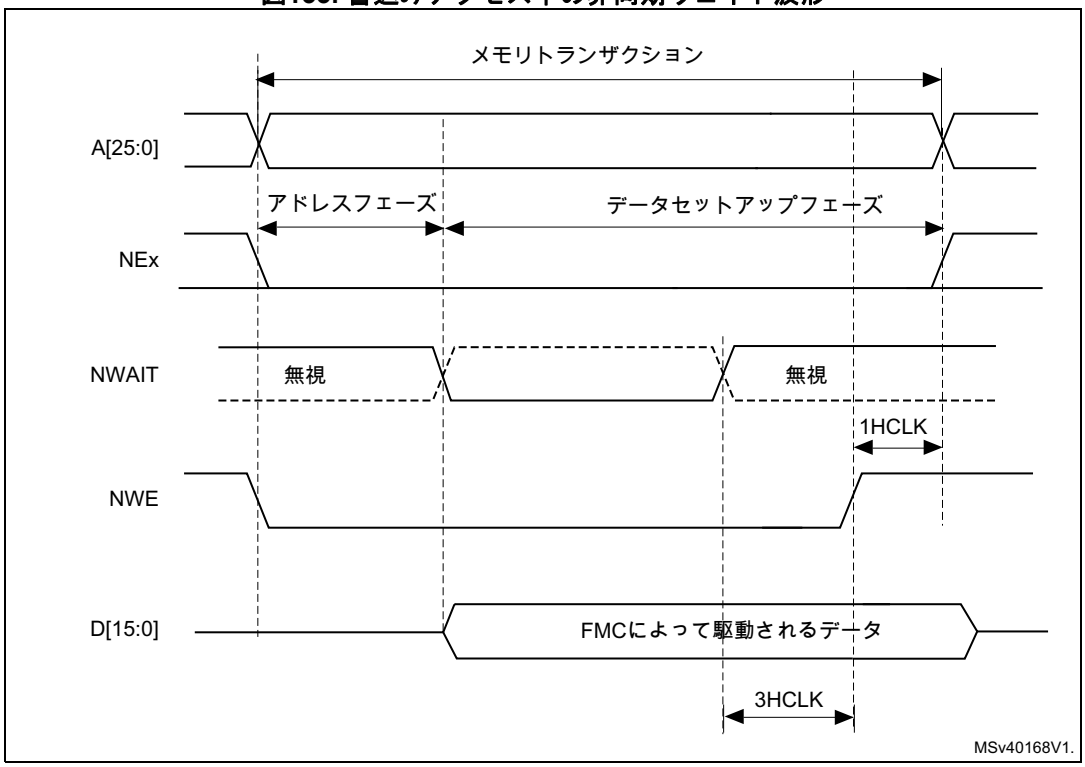
図132と図133に、非同期メモリによってWAITが解除された後、メモリ･アクセス･フェーズに追加されるHCLKクロックのサイクル数を示します（上記のケースに無関係）。

図132. 読出しアクセス中の非同期ウェイト波形



1. NWAIT極性は、FMC_BCRxレジスタのWAITPOLビット設定値に依存します。

図133. 書込みアクセス中の非同期ウェイト波形



1. NWAIT極性は、FMC_BCRxレジスタのWAITPOLビット設定値に依存します。

CellularRAM™（PSRAM）のリフレッシュ管理

CellularRAM™では、チップ･セレクト信号（NE）をメモリ･デバイスに指定されているt_{CEM}タイミングよりも長くローに維持することはできません。このタイミングはFMC_PCSCNTRレジスタでプログラムすることができます。そこには、非同期アクセスの場合はHCLKサイクルで、同期アクセスの場合はFMC_CLKサイクルで、NEのローパルスの最大期間を定義します。

27.6.5 同期トランザクション

メモリクロックFMC_CLKは、HCLKの約数です。この値はCLKDIV値とMWID/AHBデータサイズに依存し、次式で与えられます。

MWIDサイズが16bitであれ8bitであれ、FMC_CLKの分周比は常にCLKDIVのプログラムされた値で決まります。

例：

- CLKDIV = 1、MWID = 16bit、AHBデータサイズ = 8bitの場合、FMC_CLK = HCLK / 2

NOR型Flashメモリは、NADVのアサーションからCLKがハイになるまでの最小時間を指定します。この制約を満たすために、FMCは、同期アクセスの最初の内部クロックサイクルでは（NADVアサーションの前）、クロックをメモリに供給しません。これにより、メモリクロックの立上がりエッジは、NADVローパルスの中間で発生します。

データ遅延とNORメモリ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLATの値は、NOR型Flash設定レジスタで指定された遅延の値と一致しなければなりません。FMCは、NADVがローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： **一部のNOR Flashメモリは、データ遅延カウントにNADVローサイクルを含めるので、NOR Flashの遅延とFMCDATLATパラメータの厳密な関係は、次のいずれかです。**

- NOR型Flash遅延 = (DATLAT + 2) CLKクロック･サイクル、または
- NOR型Flash遅延 = (DATLAT + 3) CLKクロック･サイクル

最近のメモリの中には、遅延フェーズにおいてNWAITをアサートするものがあります。そのような場合には、DATLATを最小値に設定することができます。結果として、FMCはデータをサンプリングして、データが有効であるかどうか評価するのに十分な時間、ウェイトします。このように、FMCは、メモリが遅延から抜けたことを検出し、実際のデータが処理されます。

その他のメモリは、遅延中にはNWAITをアサートしません。この場合、FMCとメモリの両方について、遅延が正しく設定されなければなりません。そうしないと、正しいデータの代わりに無効なデータが取られたり、有効なデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードに設定されている場合、例えば、16bitメモリでAHBシングルバーストトランザクションがリクエストされると、FMCは長さ1（AHB転送が16bitの場合）または2（AHB転送が32bitの場合）のバーストトランザクションを実行して、最後のデータがストローブされると、チップセレクト信号をデアサートします。

このような転送はサイクル数の点で、非同期読出し動作に比較して最も効率的ではありません。しかし、時間がかかるメモリアクセスモードの再プログラムには、まず、ランダム非同期アクセスが必要です。

CellularRAM™ 1.5のページ境界またぎ

CellularRAM™ 1.5では、ページ境界をまたぐバーストアクセスは許可されていません。メモリ･ページ･サイズに従ってFMC_BCR1レジスタのCPSIZEビットを設定することで、メモリ･ページ･サイズに達した時点で自動的にバースト･アクセスを分割するためにFMCコントローラを使用します。

ウェイト管理

同期NOR Flashメモリでは、NWAITがプログラムされた遅延期間（（DATLAT+2）CLKサイクルに対応）後に評価されます。

NWAITがアクティブの場合（WAITPOL = 0ではローレベル、WAITPOL = 1ではハイレベル）、ウェイトステートはNWAITが非アクティブ（WAITPOL = 0ではハイレベル、WAITPOL = 1ではローレベル）になるまでが挿入されます。

NWAITが非アクティブのときには、データはただちに（ビットWAITCFG=1）または次のクロックエッジで（ビットWAITCFG=0）、有効とみなされます。

NWAIT信号によるウェイトステートの挿入時には、コントローラはメモリへのクロックパルスの送信を続け、チップセレクトと出力イネーブル信号を有効に維持します。データは有効と見なしません。

バースト･モードでのNOR FlashのNWAIT信号には、以下の2つのタイミング設定があります。

- Flash メモリは、ウェイト･ステートの 1 データサイクル前に NWAIT 信号をアサートします（リセット後のデフォルト）。
- Flashメモリは、ウェイト･ステートの間にNWAIT信号をアサートします。

FMCはFMC_BCRxレジスタ（x = 0..3）のWAITCFGビットを使って、各チップセレクトに対する両NOR Flashウェイト･ステート設定をサポートします。

図134. 待ち設定波形

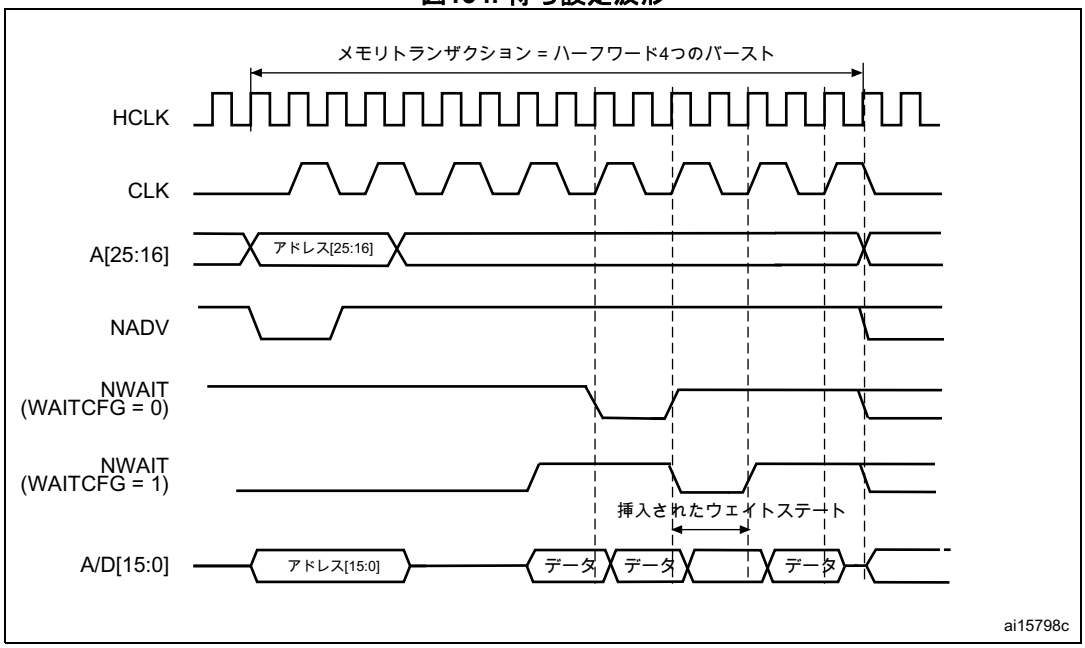
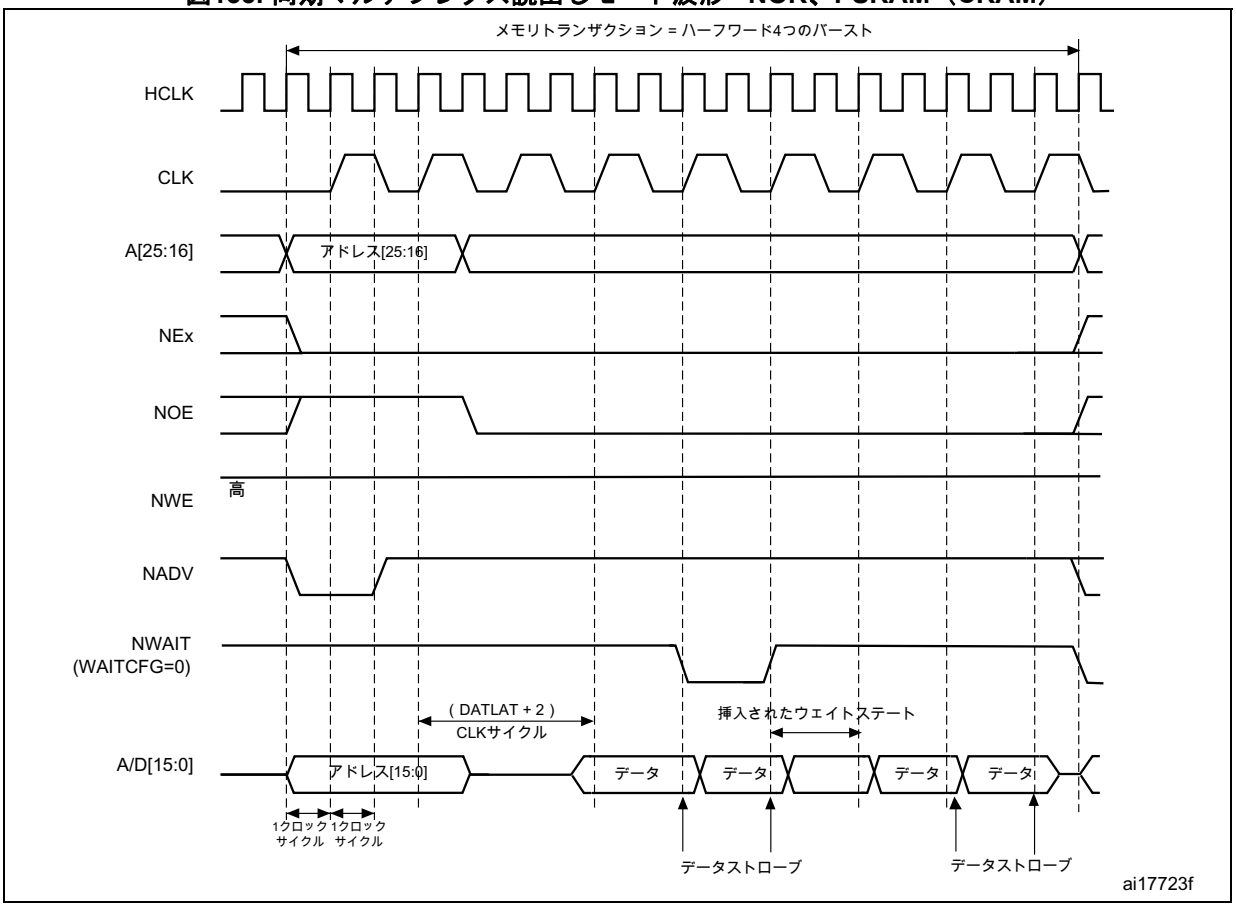


図135. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM）



1. バイトレーン出力（NBLは示されていません。NORアクセス時にはハイに保たれ、PSRAM（CRAM）アクセス時にはローに保たれます。）

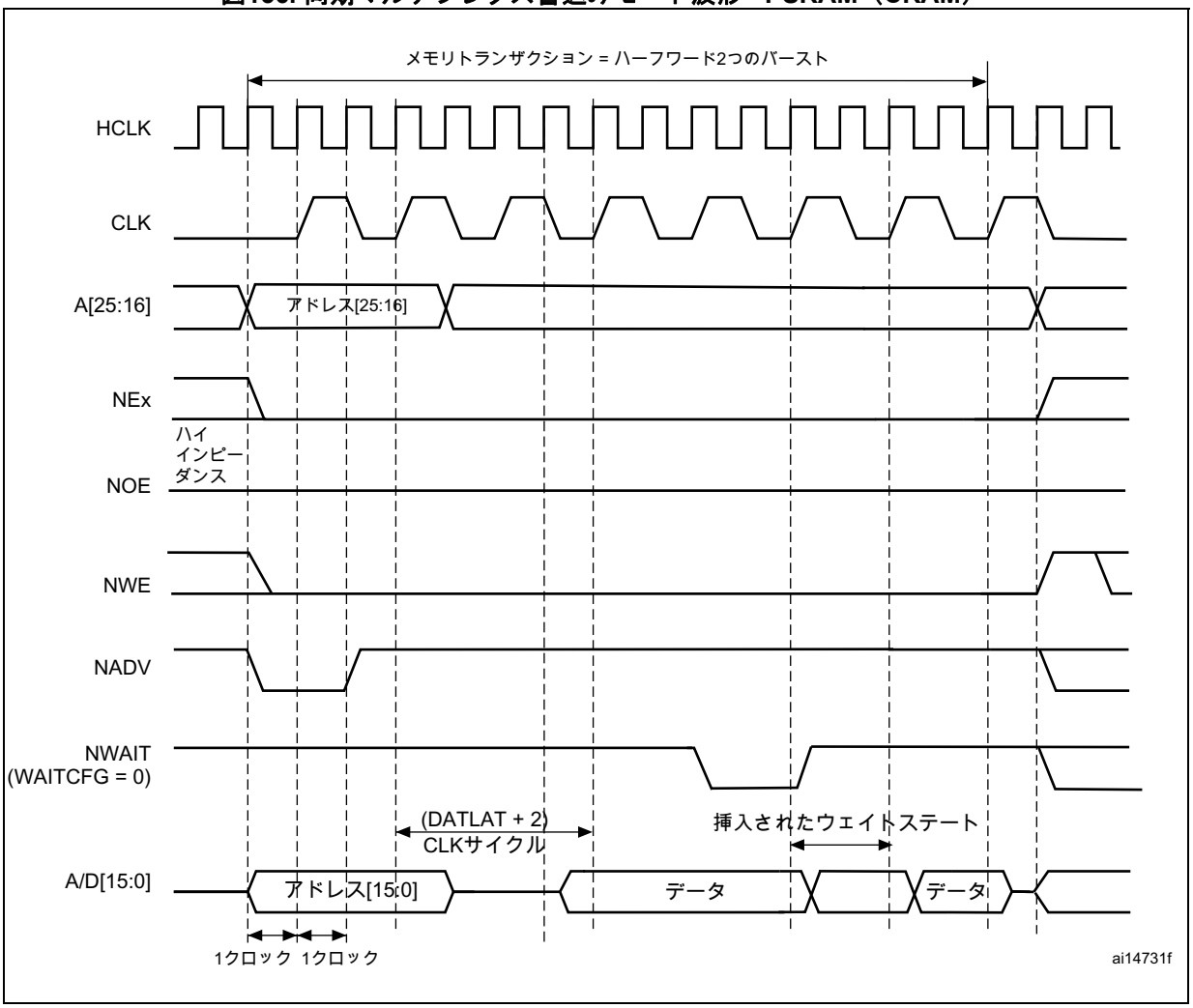
表237.FMC_BCRxビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読出し時は影響しません。
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は1にセット、そうでなければ0のままとします。
12	WREN	同期読出し時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	予約済みです。	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済みです。	0x1
6	FACCEN	メモリ･サポートに従って設定します（NOR型Flashメモリ）
5～4	MWID	必要に応じて設定します。
3～2	MTYP	0x1または0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表238.FMC_BTRxビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31:30	DATAHLD	無視
29:28	ACCMOD	0x0
27 ~ 24	DATLAT	データ遅延
27 ~ 24	DATLAT	データ遅延
23～20	CLKDIV	0x0の場合、CLK＝HCLKとなります。 0x1の場合、CLK＝2×HCLKとなります。 ..
19～16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15～8	DATAST	無視
7～4	ADDHLD	無視
3～0	ADDSET	無視

図136. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM）



1. メモリは、NWAIT信号を1サイクル前に発行しなければならないので、WAITCFGを0にプログラムする必要があります。
2. バイトレーン（NBL）出力は、図に示されておらず、NExがアクティブな間はローに保たれます。

表239.FMC_BCRxビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:24	予約済み	0x000
23:22	NBLSET[1:0].	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x1
18:16	CPSIZE	必要に応じて（CRAM1.5は0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は1にセット、そうでなければ0のままとします。

表239.FMC_BCRxビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
12	WREN	0x1
11	WAITCFG	0x0
10	予約済みです。	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期読出し時は影響しません。
7	予約済みです。	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5～4	MWID	必要に応じて設定します。
3～2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表240.FMC_BTRxビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31～30	DATAHLD	無視
29:28	ACCMOD	0x0
27 ~ 24	DATLAT	データ遅延
23～20	CLKDIV	0x0の場合、CLK＝HCLKとなります。 0x1の場合、CLK＝2×HCLKとなります。
19～16	BUSTURN	NExハイからNExローまでの時間（BUSTURN HCLK）。
15～8	DATAST	無視
7～4	ADDHLD	無視
3～0	ADDSET	無視

27.6.6 NOR/PSRAMコントローラレジスタ

SRAM/NOR型Flashのバンクxのチップセレクト制御レジスタ（FMC_BCRx）（x = 1 ~ 4）

アドレスオフセット：0x00 + 0x8 * (x-1)(x = 1から4)

リセット値：0x0000 30DB, 0x0000 30D2, 0x0000 30D2, 0x0000 30D2

このレジスタは、SRAM、PSRAM、FRAM、およびNOR型Flashメモリで使用される各メモリ･バンクの制御情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBLSET[1:0]		WFDIS	CCLK EN	CBURST RW	CPSIZE[2:0]		
rw								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ASYNCW AIT	EXT MOD	WAIT EN	WREN	WAIT CFG	Res.	WAIT POL	BURSTE N	Res.	FACC EN	MWID[1:0]		MTYP[1:0]		MUX EN	MBK EN
rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31 **FMCEN**：FMCコントローラ有効化

このビットは、FMCコントローラを有効または無効にします。

0：FMCコントローラを無効にします。

1：FMCコントローラを有効にします。

注： FMC_BCR2..4レジスタのFMCENビットは無視されます。このビットはFMC_BCR1レジスタからのみ有効化できます。

ビット30:24 予約済みであり、リセット値に保持する必要があります。

ビット23:22 **NBLSET[1:0]：**バイトレーン（NBL）セットアップ

これらのビットは、NBLxローからチップセレクトNExローまでのNBLセットアップ時間を設定します。

00：NBLセットアップ時間は、0AHBクロック･サイクルです。

01：NBLセットアップ時間は、1AHBクロック･サイクルです。

10：NBLセットアップ時間は、2AHBクロック･サイクルです。

11：NBLセットアップ時間は、3AHBクロック･サイクルです。

ビット21 **WFDIS：**書込みFIFO無効化

このビットは、FMCコントローラで使用される書込みFIFOを無効にします。

0：書込みFIFO有効（リセット後のデフォルト）

1：書込みFIFO無効

注： FMC_BCR2..4レジスタのWFDISビットは無視されます。このビットはFMC_BCR1レジスタからのみ有効化できます。

ビット20 **CCLKEN：連続クロックの有効化**

このビットは、外部メモリデバイスに対するFMC_CLKクロック出力を有効化します。

0：FMC_CLKは同期メモリ･アクセス（読出し／書込みトランザクション）時にのみ生成されます。FMC_CLKクロック比は、FMC_BCRxレジスタに設定したCLKDIV値により指定されます（リセット後のデフォルト）。

1：FMC_CLKは、非同期アクセスと同期アクセス時に連続的に生成されます。CCLKENがセットされると、FMC_CLKクロックがアクティブになります。

注： FMC_BCR2..4レジスタのCCLKENビットは無視されます。このビットはFMC_BCR1レジスタからのみ有効化できます。FMC_CLK連続クロックを発生するときは、バンク1を同期モードに設定する必要があります。

注： CCLKENビットがセットされている場合、FMC_CLKクロック比はFMC_BTR1レジスタのCLKDIV値で指定されます。FMC_BWTR1のCLKDIVは、無視されます。

注： 同期モードを使用し、かつCCLKENビットがセットされる場合、バンク1以外のバンクに接続される同期メモリはすべて、同じクロックで駆動されます（他のバンクに対する FMC_BTR2..4 レジスタと FMC_BWTR2..4レジスタのCLKDIV値は無効です）。

ビット19 **CBURSTRW：書込みバースト有効化**

バーストモードで動作するPSRAM（CRAM）の場合、このビットは書込み動作時に同期アクセスを有効にします。同期読出しアクセスのイネーブルビットは、FMC_BCRxレジスタのBURSTENビットです。

0：書込み動作は、常に非同期モードで実行されます。

1：書込み動作は、同期モードで実行されます。

ビット18:16 **CPSIZE[2:0]**：CRAMのページサイズ

これらのビットは、ページ間でアドレス境界をまたぐバーストアクセスができないCellularRAM™ 1.5に使用されます。これらのビットを設定すると、FMCコントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割します（ページサイズについてはメモリのデータシートを参照）。

000：ページ境界を跨ぐ場合、バーストは分割されません（リセット後のデフォルト）

001：128バイト

010：256バイト

011：512バイト

100：1024バイト

その他：予約済み

ビット15 **ASYNCWAIT**：非同期転送中のウエイト信号

このビットは、非同期プロトコル時でも、FMCがウェイト信号を有効化／無効化できるようにします。

0：非同期プロトコル動作中は、NWAIT信号は考慮されません（リセット後のデフォルト）。

1：非同期プロトコル動作中にNWAIT信号は考慮されます。

ビット14 **EXTMOD：拡張モード有効化**

このビットを使うと、FMCがFMC_BWTRレジスタで非マルチプレクス非同期アクセスの書込みタイミングをプログラムできるため、読出しと書込みで異なるタイミングが可能になります。

0：FMC_BWTRレジスタ内部の値は考慮されません（リセット後のデフォルト）。

1：FMC_BWTRレジスタ内部の値が考慮されます。

注： 拡張モードを無効化した場合、FMCは次のようにモード1またはモード2で動作することができます。

- **SRAM/PSRAMメモリ･タイプを選択した場合（MTYP = 0x0または0x01）、モード1はデフォルト･モードです。**
- **モード2は、NORメモリタイプを選択した場合（MTYP = 0x10）のデフォルトモードです。**

ビット13 **WAITEN：ウェイトイネーブルビット**

このビットは、同期モードでFlashメモリをアクセスするときNWAIT信号を使用したウェイトステートの挿入を有効化／無効化します。

0：NWAIT信号は無効です（レベルは考慮されず、プログラムされたFlash遅延時間後にウェイトステートは挿入されません）。

1：NWAIT信号は有効です（レベルが考慮され、アサートされた場合、プログラムされた遅延時間後にウェイトステートが挿入されます）（リセット後のデフォルト）。

ビット12 **WREN：ライタトイネーブルビット**

このビットは、FMCによるバンクへの書込み動作の有効／無効を示します。

0：FMCによるバンクへの書込み動作は無効にされています。AHBエラーが報告されます。

1：FMCによるバンクへの書込み動作は有効にされています（リセット後のデフォルト）。

ビット11 **WAITCFG：ウェイトタイミング設定**

NWAIT信号は、メモリからのデータが有効か否か、または同期モードでメモリをアクセスするときウェイトステートを挿入すべきかを表示します。この設定ビットは、ウェイトステートの1クロック前、またはウェイトステート中に、メモリによってNWAITがアサートされるかどうかを定義します。

0：NWAIT信号は、ウェイトステートの1データサイクル前にアクティブです（リセット後のデフォルト）。

1：NWAIT信号は、ウェイトステートの間アクティブです（PSRAMには不使用）。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **WAITPOL：ウェイト信号極性ビット**

同期モードまたは非同期モードで使用されるメモリからのウェイト信号の極性を定義します。

0：NWAITはアクティブローです（リセット後のデフォルト）。

1：NWAITはアクティブハイです。

ビット8 **BURSTEN：バーストイネーブルビット**

このビットは、読出し動作での同期アクセスを有効化/無効化します。これは、バースト･モードでの同期メモリ動作に対してのみ有効です。

0：バースト･モードは無効です（リセット後のデフォルト）。読出しアクセスは非同期モードで実行されます。

1：バースト･モードは有効です。読出しアクセスは同期モードで実行されます。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **FACCEN**：Flashアクセス有効化

NOR型Flashメモリ･アクセス動作を有効にします。

0：対応するNOR型Flashメモリ･アクセスは無効です。

1：対応するNOR型Flashメモリ･アクセスは有効です（リセット後のデフォルト）。

ビット5:4 **MWID[1:0]：メモリのデータバス幅**

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。

00：8bit

01：16bit（リセット後のデフォルト）

10：予約済み

11：予約済み

ビット3:2 **MTYP[1:0]：メモリタイプ**

対応するメモリ･バンクに接続される外部メモリのタイプを定義します。

00：SRAM/FRAM（バンク2...4のリセット後のデフォルト）

01：PSRAM（CRAM）/ FRAM

10：NOR型Flash/OneNAND Flash（バンク1のリセット後のデフォルト）

11：予約済み

ビット1 **MUXEN：アドレス／データマルチプレクシングイネーブルビット**

このビットがセットされているときには、データバス上でアドレスとデータがマルチプレクスされます。NORおよびPSRAMメモリでのみ有効です。

0：アドレス／データはマルチプレクスされません。

1：アドレス／データはデータバス上でマルチプレクスされます（リセット後のデフォルト）。

ビット0 **MBKEN：メモリバンクイネーブルビット**

メモリバンクを有効にします。リセット後は、バンク1は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AHBバス上でERRORが発生します。

0：対応するメモリバンクは無効です。

1：対応するメモリバンクは有効です。

SRAM/NOR型Flashのバンクxのチップセレクトタイミングレジスタ（FMC_BTRx）

アドレスオフセット：0x04 + 0x8 * (x-1)(x = 1から4)

リセット値：0x0FFF FFFF

このレジスタは、SRAM、PSRAM、およびNOR型Flashメモリで使用される各メモリ･バンクの制御情報を含みます。FMC_BCRxレジスタのEXTMODビットがセットされている場合、このレジスタは、書込みと読出しアクセスに分割されます。すなわち、読出しアクセスを設定するためのレジスタ（このレジスタ）と、書込みアクセスを設定するためのレジスタ（FMC_BWTR x レジスタ）の2つのレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 **DATAHLD[1:0]**：データホールドフェーズ時間

これらのビットは、データホールドフェーズの時間をHCLKサイクル数で定義するためにソフトウェアによって書き込まれ（図119～図131を参照）、非同期アクセスで使用されます。

読出しアクセスの場合

00：DATAHLDフェーズ時間 = 0 * HCLKクロック･サイクル（デフォルト）

01：DATAHLDフェーズ時間 = 1 * HCLKクロック･サイクル

10：DATAHLDフェーズ時間 = 2 * HCLKクロック･サイクル

11：DATAHLDフェーズ時間 = 3 * HCLKクロック･サイクル

書込みアクセスの場合

00：DATAHLDフェーズ時間 = 1 * HCLKクロック･サイクル（デフォルト）

01：DATAHLDフェーズ時間 = 2 * HCLKクロック･サイクル

10：DATAHLDフェーズ時間 = 3 * HCLKクロック･サイクル

11：DATAHLDフェーズ時間 = 4 * HCLKクロック･サイクル

ビット29:28 **ACCMOD[1:0]**：アクセスモード

タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRxレジスタのEXTMODビットが1にセットされている場合のみ考慮されます。

00：アクセスモードA

01：アクセスモードB

10：アクセスモードC

11：アクセスモードD

ビット27:24 **DATLAT[3:0]**：（下のビット説明参照）：同期メモリのデータ遅延

読出し／書込みバーストモードを有効にした同期アクセスの場合（BURSTEN/CBURSTRWビットをセット）、最初のデータの読み書きの前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、HCLK周期ではなく、FMC_CLK周期で表されます。

非同期アクセスでは、この値は無視されます。

0000：最初のバーストアクセスでは、2CLKクロックサイクルのデータ遅延

1111：最初のバーストアクセスでは、17CLKクロックサイクルのデータ遅延（リセット後のデフォルト値）

ビット23:20 **CLKDIV[3:0]**：クロック分周比（FMC_CLK信号）

FMC_CLKクロック出力信号の周期を定義します。HCLKサイクル数で表されます。

0000：FMC_CLK周期 = 1 × HCLK周期

0001：FMC_CLK周期 = 2 × HCLK周期

0010：FMC_CLK周期 = 3 × HCLK周期

1111：FMC_CLK周期 = 16 × HCLK周期（リセット後のデフォルト値）

非同期NOR型Flash、SRAM、またはPSRAMアクセスでは、この値は無視されます。

注： FMC_CLK分周比の式については、セクション 27.6.5: 同期トランザクションを参照してください。

ビット19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

これらのビットはソフトウェアから書き込まれて、同じバンクの次のトランザクションまでの現在の読出しまたは書込みトランザクションの終わりに遅延を追加します。

この遅延を使うと、連続トランザクション間の最小時間（NExハイからNExローまでのt_{EHEL}）と、読出しアクセス後にメモリがデータ･バスを開放するために要する最大時間（t_{EHQZ}、チップ･イネーブル･ハイから出力ハイインピーダンスまで）を次のように一致させることができます。モードDおよびマルチプレクスモードでは、この遅延が推奨されます。非マルチプレクスメモリでは、バスターンアラウンド遅延を最小値にセットできます。

(BUSTURN + 1) HCLK周期 ≥ max(t_{EHEL} min, t_{EHQZ} max)

FRAMメモリでは、バス･ターン･アラウンド遅延を最小tPC（プリチャージ時間）タイミングに合わせて設定する必要があります。バスターンアラウンド遅延は、tPCメモリタイミングと合わせて、同じバンクでの連続トランザクション間（読出し／読出し、書込み／書込み、読出し／書込み、および書込み／読出し）に挿入されます。チップセレクトは、連続アクセス間でトグルします。

(BUSTURN + 1) HCLK周期 ≥ tPC min

0000：BUSTURNフェーズ時間 = 1HCLKクロックサイクル追加

...

1111：BUSTURNフェーズ時間 = 16 × HCLKクロックサイクル追加（リセット後のデフォルト値）

ビット15:8 **DATAST[7:0]**：データフェーズ時間

これらのビットは、非同期アクセスで使用されるデータフェーズの時間を定義するためにソフトウェアにより書き込まれます（図119～図131を参照）。

0000 0000：予約済み

0000 0001：DATASTフェーズ時間 = 1 * HCLKクロックサイクル

0000 0010：DATASTフェーズ時間 = 2 * HCLKクロックサイクル

...

1111 1111：DATASTフェーズ時間 = 255 * HCLKクロックサイクル（リセット後のデフォルト値）

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図（図119～図131）を参照してください。

例：モード1、書込みアクセス、DATAST=1：データフェーズ時間 = DATAST+1 = 2HCLKクロックサイクル

注： 同期アクセスでは、この値は無視されます。

ビット7:4 **ADDHLD[3:0]**：アドレスホールドフェーズ時間

これらのビットは、アドレスホールドフェーズの時間を定義するためにソフトウェアから書き込まれ（図119～図131を参照）、モードDまたはマルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLDフェーズ時間 = 1 * HCLKクロックサイクル

0010：ADDHLDフェーズ時間 = 2 * HCLKクロックサイクル

...

1111：ADDHLDフェーズ時間 = 15 * HCLKクロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図（図119～図131）を参照してください。

注： 同期アクセスでは、この値は使用されず、アドレスホールドフェーズは、常に1メモリクロック周期の長さです。

ビット3:0 **ADDSET[3:0]**：アドレスセットアップフェーズ時間

これらのビットは、アドレス･セットアップ･フェーズの時間を定義するために、ソフトウェアによって書き込まれ（図119～図131を参照）、SRAM、ROM、非同期NOR Flash、およびPSRAMで使用されます。

0000：ADDSETフェーズ時間 = 0 * HCLKクロックサイクル

...

1111：ADDSETフェーズ時間 = 15 * HCLKクロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図（図119～図131）を参照してください。

注：　同期アクセスでは、この値は無視されます。

マルチプレクスモードとモードDでは、ADDSETの最小値は1です。

モード1とPSRAMメモリでは、ADDSETの最小値は1です。

注：　PSRAM（CRAM）の遅延は、内部更新のために可変です。したがって、これらのメモリは、必要に応じて遅延を延長するために、遅延フェーズ全体についてNWAIT信号を発行します。
PSRAM（CRAM）では、フィールドDATLATを0に設定しなければなりません。こうすると、FMCは、すぐにレイテンシフェーズを終了して、メモリからのNWAIT信号のサンプリングを開始し、メモリがレディ状態になると、読出しまたは書込みを開始します。
この方法は、古い世代のFlashメモリと異なり、NWAIT信号を発行する最新世代の同期Flashメモリでも使用できます（使用中の特定のFlashメモリのデータシートを参照してください）。

SRAM/NOR型Flash書込みタイミングレジスタx（FMC_BWTRx）

アドレスオフセット：0x104 + 0x8 * (x - 1)、（x = 1 ~ 4）

リセット値：0x0FFF FFFF

このレジスタには各メモリバンクの制御情報が格納されます。SRAM、PSRAM、NOR Flashメモリに使用されます。FMC_BCRxレジスタのEXTMODビットがセットされているときには、このレジスタは書込みアクセスについてアクティブになります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]			
rw	rw	rw	rw									rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 **DATAHLD[1:0]**：データホールドフェーズ時間

これらのビットは、データホールドフェーズの時間をHCLKサイクル数で定義するためにソフトウェアによって書き込まれ（図119～図131を参照）、非同期書込みアクセスで使用されます。

00：DATAHLDフェーズ時間 = 1 * HCLKクロック･サイクル（デフォルト）

01：DATAHLDフェーズ時間 = 2 * HCLKクロック･サイクル

10：DATAHLDフェーズ時間 = 3 * HCLKクロック･サイクル

11：DATAHLDフェーズ時間 = 4 * HCLKクロック･サイクル

ビット29:28 **ACCMOD[1:0]**：アクセスモード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRxレジスタのEXTMODビットが1にセットされている場合のみ考慮されます。

00：アクセスモードA

01：アクセスモードB

10：アクセスモードC

11：アクセスモードD

ビット27:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

これらのビットはソフトウェアから書き込まれて、同じバンクの次のトランザクションまでの現在の書込みトランザクションの終わりに遅延を追加します。

FRAMメモリでは、バス･ターン･アラウンド遅延を最小t_{PC}（プリチャージ時間）タイミングに合わせて設定する必要があります。バスターンアラウンド遅延は、同じバンクでの連続トランザクション間（読出し／読出し、書込み／書込み、読出し／書込み、および書込み／読出し）に挿入されます。チップセレクトは、連続アクセス間でトグルします。

(BUSTURN + 1) HCLK周期 ≥ tPC min

0000：BUSTURNフェーズ時間 = 1HCLKクロックサイクル追加

...

1111：BUSTURNフェーズ時間 = 16 × HCLKクロックサイクル追加（リセット後のデフォルト値）

ビット15:8 **DATAST[7:0]：**データフェーズ時間。

これらのビットは、非同期SRAM、PSRAM、NOR Flashメモリ･アクセスで使用されるデータ･フェーズの時間を定義するために、ソフトウェアによって書き込まれます（図119～図131を参照）。

0000 0000：予約済み

0000 0001：DATASTフェーズ時間 = 1 * HCLKクロックサイクル

0000 0010：DATASTフェーズ時間 = 2 * HCLKクロックサイクル

...

1111 1111：DATASTフェーズ時間 = 255 * HCLKクロックサイクル（リセット後のデフォルト値）

ビット7:4 **ADDHLD[3:0]：**アドレスホールドフェーズ時間。

これらのビットは、アドレスホールドフェーズの時間を定義するためにソフトウェアによって書き込まれ（図128～図131を参照）、非同期マルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLDフェーズ時間 = 1 * HCLKクロックサイクル

0010：ADDHLDフェーズ時間 = 2 * HCLKクロックサイクル

...

1111：ADDHLDフェーズ時間 = 15 * HCLKクロックサイクル（リセット後のデフォルト値）

注： 同期NOR型Flashアクセスでは、この値は使用されず、アドレス･ホールド･フェーズは常に1Flashクロック周期の長さです。

ビット3:0 **ADDSET[3:0]：**アドレスセットアップフェーズ時間。

これらのビットは、アドレスセットアップフェーズの時間をHCLKサイクル数で定義するためにソフトウェアによって書き込まれ（図119～図131を参照）、非同期アクセスで使用されます。

0000：ADDSETフェーズ時間 = 0 * HCLKクロックサイクル

...

1111：ADDSETフェーズ時間 = 15 * HCLKクロックサイクル（リセット後のデフォルト値）

注： 同期アクセスでは、この値は使用されず、アドレス･セットアップ･フェーズは常に 1Flash クロック周期の長さです。マルチプレクスモードでは、ADDSETの最小値は1です。

PSRAMチップセレクトカウンタレジスタ（FMC_PCSCNTR）

アドレスオフセット：0x20

リセット値：0x0000 0000

このレジスタは、同期および非同期モードのPSRAMチップセレクトカウンタの値を格納します。チップセレクトカウンタはすべてのバンクに共通で、バンクごとに個別に有効にできます。PSRAMの読出しまたは書込みアクセス時に、この値はタイマにロードされ、タイマはNE信号がローに保持されている間、デクリメントされます。タイマが0に達すると、PSRAMコントローラは現在のアクセスを分割し、NEをトグルしてPSRAMデバイスのリフレッシュを可能にし、そして新しいアクセスを再開します。プログラムされたカウンタ値によって、PSRAMデバイスに指定された最大NEパルス幅（t_{CEM}）が保証されます。カウンタは、NEのハイからローへの遷移によって新しいアクセスが開始されるたびに、リロードされてデクリメントが開始されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTB4EN	CNTB3EN	CNTB2EN	CNTB1EN
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSCOUNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CNTB4EN**：カウンタバンク4有効化

このビットは、PSRAM/NORバンク4に対するチップセレクトカウンタを有効にします。

0：バンク4に対するカウンタは無効です。

1：バンク4に対するカウンタは有効です。

ビット18 **CNTB3EN**：カウンタバンク3有効化

このビットは、PSRAM/NORバンク3に対するチップセレクトカウンタを有効にします。

0：バンク3に対するカウンタは無効です。

1：バンク3に対するカウンタは有効です。

ビット17 **CNTB2EN**：カウンタバンク2有効化

このビットは、PSRAM/NORバンク2に対するチップセレクトカウンタを有効にします。

0：バンク2に対するカウンタは無効です。

1：バンク2に対するカウンタは有効です。

ビット16 **CNTB1EN**：カウンタバンク1有効化

このビットは、PSRAM/NORバンク1に対するチップセレクトカウンタを有効にします。

0：バンク1に対するカウンタは無効です。

1：バンク1に対するカウンタは有効です。

ビット15:0 **CSCOUNT[15:0]**：チップセレクトカウンタ

このビットフィールドはチップがローを選択する期間の最大値を定義するために使用されます。これは次の式によって求められます。

CSCOUNT[15:0] × T_{AHB}、ここでT_{AHB}はAHBクロック周期です。

リフレッシュを考慮し、PSRAMチップはt_{CEM} = ~4µsを超えてローを選択してはなりません。

CSCOUNT[15:0]は非同期および同期モードの両方に適用されます。

CSCOUNT[15:0] = 0x0000の場合、この機能は無効です。

27.7 NAND型Flashコントローラ

FMCは、以下のタイプのデバイスを駆動するのに適した信号タイミングを発生します。

- 8bitと16bitのNAND Flashメモリ

NANDバンクは、専用レジスタによって設定されます（セクション27.7.7を参照）。プログラ可能なメモリパラメータは、アクセスタイミング（表241を参照）とECC構成を含みます。

表241.プログラム可能なNAND Flashのアクセス･パラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
メモリセットアップ時間	コマンドアサーション前にアドレスをセットアップするために必要なクロックサイクル（HCLK）数	読出し／書込み	AHBクロックサイクル（HCLK）	1	255
メモリウェイト	コマンドアサーションの最小時間（HCLKクロックサイクル数）	読出し／書込み	AHBクロックサイクル（HCLK）	2	255
メモリホールド	コマンドアサーション解除後に、アドレスを（書込みアクセス時はデータも）保持する必要があるクロックサイクル数（HCLK）。	読出し／書込み	AHBクロックサイクル（HCLK）	1	254
メモリデータバスハイインピーダンス	書込みアクセス開始後にデータバスがハイインピーダンス状態に保たれるクロックサイクル数（HCLK）	書込み	AHBクロックサイクル（HCLK）	1	255

27.7.1 外部メモリインタフェース信号

次の表に、通常、NAND型Flashメモリとのインタフェースに使用される信号を示します。

注： **接頭辞「N」は、アクティブローの信号を表します。**

8bit NAND Flashメモリ

表242.8bit NAND型Flash

FMC信号名	I/O	機能
A[17]	O	NAND型Flashアドレス･ラッチ･イネーブル（ALE）信号
A[16]	O	NAND型Flashコマンド･ラッチ･イネーブル（CLE）信号
D[7:0]	I/O	8bitマルチプレクス、双方向アドレス／データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMCへのNAND型Flashレディ／ビジー入力信号

FMCは必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16bit NAND型Flashメモリ

表243.16bit NAND型Flash

FMC信号名	I/O	機能
A[17]	O	NAND型Flashアドレス･ラッチ･イネーブル（ALE）信号
A[16]	O	NAND型Flashコマンド･ラッチ･イネーブル（CLE）信号
D[15:0]	I/O	16bitマルチプレクス、双方向アドレス／データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMCへのNAND型Flashレディ／ビジー入力信号

FMCは必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

27.7.2 NAND型Flashによってサポートされるメモリとトランザクション

表244に、サポートされるデバイス、アクセスモード、およびトランザクションを示します。NAND Flashコントローラによって許可されない（またはサポートされない）トランザクションは、グレイで示されています。

表244.サポートされるメモリおよびトランザクション

デバイス	モード	読出し／書込み	AHB データ サイズ	メモリ データ サイズ	許可／禁止	コメント
NAND 8bit	非同期	R	8	8	Y	-
	非同期	W	8	8	Y	-
	非同期	R	16	8	Y	2つのFMCアクセスに分割
	非同期	W	16	8	Y	2つのFMCアクセスに分割
	非同期	R	32	8	Y	4つのFMCアクセスに分割
	非同期	W	32	8	Y	4つのFMCアクセスに分割
NAND 16bit	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2つのFMCアクセスに分割
	非同期	W	32	16	Y	2つのFMCアクセスに分割

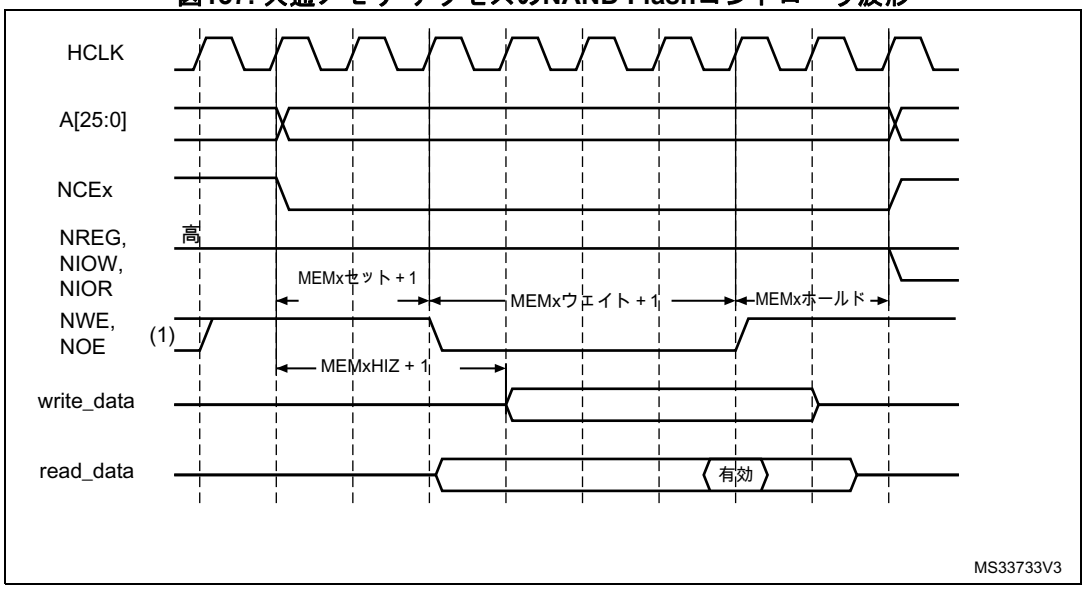
27.7.3 NAND Flashメモリのタイミング図

NAND型Flashメモリのバンクは、以下のレジスタを通じて管理されます。

- 制御レジスタFMC_PCR
- 割込みステータスレジスタ：FMC_SR
- ECCレジスタ：FMC_ECCR
- 共通メモリ空間用タイミングレジスタ：FMC_PMEM
- 属性メモリ空間用タイミングレジスタ：FMC_PATT

各タイミング設定レジスタは、NAND Flashアクセスの3つのフェーズのHCLKサイクル数を定義するための3つのパラメータと、書込みアクセス時にデータ･バスの駆動を開始するタイミングを定義する1つのパラメータを格納します。図137に共通メモリアクセス用のタイミングパラメータの定義を示します。属性メモリ空間のアクセスタイミングが同様であることがわかります。

図137. 共通メモリ･アクセスのNAND Flashコントローラ波形



1. 書込みアクセス時、NOEはハイ（非アクティブ）に保たれます。読出しアクセス時、NWEはハイ（非アクティブ）に保たれます。
2. 書込みアクセスの場合、ホールド･フェーズ遅延は (MEMHOLD)HCLK サイクルとなり、読出しアクセスの場合は (MEMHOLD + 2)HCLKサイクルとなります。

27.7.4 NAND Flash動作

NAND Flashデバイスのコマンド･ラッチ･イネーブル（CLE）およびアドレス･ラッチ･イネーブル（ALE）信号は、FMCコントローラのアドレス信号によって駆動されます。これは、NAND Flashメモリにコマンドまたはアドレスを送信するには、CPUは、CPUのメモリ空間内の特定のアドレスに書込みを行う必要があることを意味します。

NAND Flashデバイスからの一般的なページ読出し動作には、次のステップが必要です。

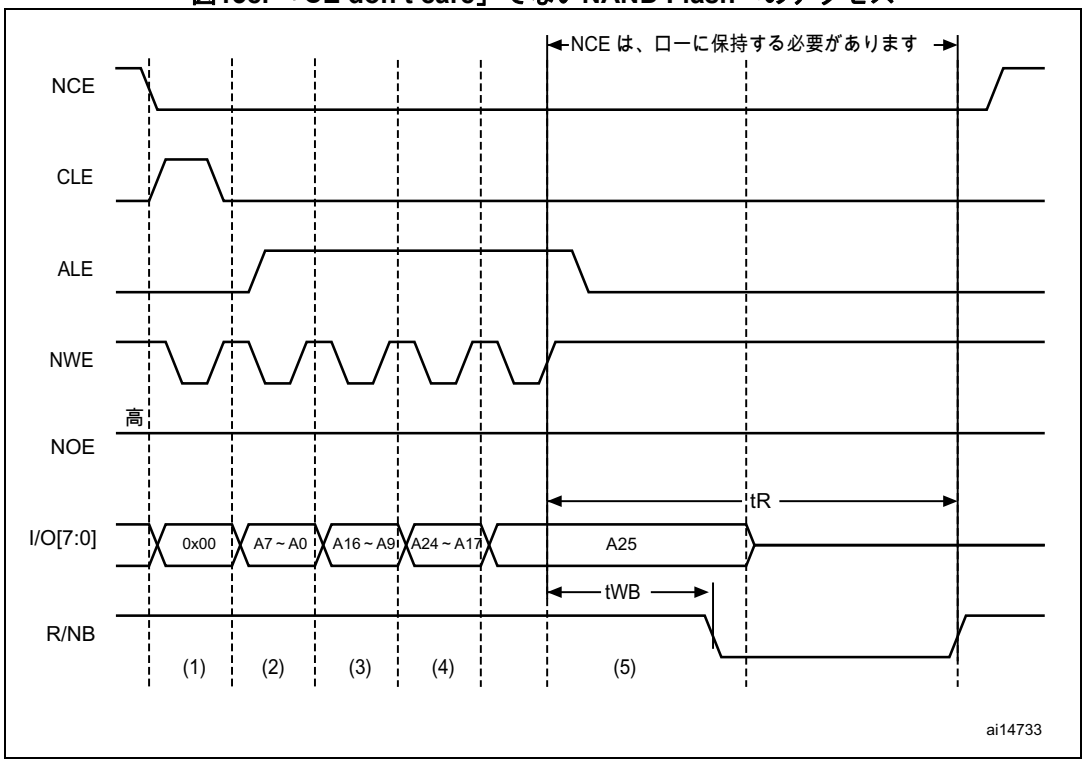
1. FMC_PCR レジスタと FMC_PMEM レジスタ（デバイスによっては、FMC_PATT も。セクション 27.7.5 : NAND Flashのプリウェイト機能を参照）を、NAND Flashメモリの特性（NAND Flashのデータバス幅に対するPWIDビット、PTYP = 1、必要に応じてPWAITEN = 0または1、タイミング設定についてはセクション 27.5.2 : NAND Flashメモリ･アドレス･マッピングを参照）に従って設定することで、対応するメモリバンクのプログラムおよび有効化を行います。

2. CPU は、1 つの Flash コマンド･バイトと等しいデータ･バイト（たとえば、Samsung の NAND Flashデバイスの場合は0x00）で、共通メモリ空間にバイト書込みを行います。NAND Flashメモリの LE入力は、書込みストローブ（NWE上のローパルス）の間はアクティブであり、書き込まれたバイトは、NAND Flashメモリによってコマンドとして認識されます。コマンドがメモリデバイスにラッチされたら、次のページ読出し動作のために書き込む必要はありません。
3. CPUは、4バイト（より小容量のデバイスの場合は3バイト）を共通メモリまたは属性空間に書き込むことによって、読出し操作の開始アドレス（STARTAD）を送信できます（STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、および64Mbit x 8bit NAND Flash メモリの場合は最後にSTARTAD[25]）。NAND型FlashデバイスのALE入力は、書込みストローブの間アクティブであり（NWE上のローパルス）、書き込まれたバイトは、読出し動作の開始アドレスとして解釈されます。属性メモリ空間を使用すると、FMCの異なるタイミング設定を使用でき、一部のNAND Flashメモリで必要とされるプリウェイト機能を実装できます（詳細については、セクション 27.7.5 : NAND Flashのプリウェイト機能を参照してください）。
4. コントローラは、NAND Flashメモリがレディ（R/NB信号がハイ）になるのを待ってから、同じメモリ･バンクまたは別のメモリ･バンクへの新しいアクセスを開始します。待っている間、コントローラはNCE信号をアクティブ（ロー）に保ちます。
5. CPUは、NAND Flashページ（データ･フィールド＋スペア･フィールド）をバイト単位で読み出すために、共通メモリ空間からバイト読出し動作を行います。
6. 次のNAND　Flashページは、CPUコマンドやアドレス書込み動作なしに読み出すことができます。これは次の3つの方法で行うことができます。
   - ステップ5に記述された動作を実行します。
   - ステップ3から動作をリスタートすることによって、新しいランダムアドレスにアクセスできます。
   - ステップ2からリスタートすることによって、NAND型Flashデバイスに新しいコマンドを送信できます。

27.7.5 NAND Flashのプリウェイト機能

一部のNAND Flashデバイスでは、アドレスの最後の部分を書き込んだ後、コントローラはR/NB信号を待つ必要があります。（図138を参照。）

図138. 「CE don’t care」でないNAND Flashへのアクセス



1. CPUはアドレス0x7001 0000にバイト0x00を書き込みました。
2. CPUはアドレス0x7002 0000にバイトA7～A0を書き込みました。
3. CPUはアドレス0x7002 0000にバイトA16～A9を書き込みました。
4. CPUはアドレス0x7002 0000にバイトA24～A17を書き込みました。
5. CPUはアドレス0x7802 0000にバイトA25を書き込みました：FMCはFMC_PATTのタイミング定義を使用して書込みアクセスを実行します。ここでATTHOLD ≥ 7（ただし、(7+1) * HCLK = 112 ns > t_{WB} max）。これにより、NCEはR/NBがローから再びハイになるまでローを維持します（NCEが無視されないNAND型Flashメモリの場合のみ必要）。

この機能が必要な場合、MEMHOLD値をプログラムすることによりt_{WB}タイミングを確実に満たすことができます。ただし、NAND Flashメモリに対するすべてのCPU読出しアクセスでは、（MEMHOLD + 2）HCLKサイクル、CPU書込みアクセスでは、（MEMHOLD）HCLKサイクルのホールド遅延がNWE信号の立上がりエッジと次のアクセスとの間に挿入されます。

このタイミング制約に対処するためには、t_{WB}タイミングに応じたATTHOLD値でタイミングレジスタをプログラムし、MEMHOLD値を最小値のままにしておくことで、属性メモリ空間を使用できます。この場合、CPUはNAND Flashのすべての読出しおよび書込みアクセスには共通メモリ空間を使用する必要があります。ただし、NAND Flashデバイスに最後のアドレス･バイトを書き込むときには、属性メモリ空間に書き込む必要があります。

27.7.6 NAND型Flashメモリ内のエラー訂正コード（ECC）の計算

FMCNANDカードコントローラには、メモリバンクごとに1つずつ、2つのエラー訂正コード計算ハードウェアブロックがあります。これらのハードウェアブロックは、ソフトウェアでECCを処理する場合、ホストCPUの負荷を削減します。

この2つのECCブロックは同一であり、それぞれバンク2とバンク3に接続されます。結果として、バンク4に接続されたメモリでは、ハードウェアECC計算を使用できません。

FMCに実装されたエラー訂正コード（ECC）のアルゴリズムでは、NAND Flashメモリに対する256、512、1024、2048、4096、または8192バイトの読出しまたは書込みごとに、1bitのエラー訂正と2bitのエラー検出を実行できます。ハミングコーディングアルゴリズムを採用し、行と列のパリティ計算を行います。

ECCモジュールは、NAND Flashメモリバンクがアクティブになるたびに、NAND Flashデータバスおよび読出し／書込み信号（NCEとNWE）を監視します。

ECCは次のように動作します。

- NAND Flashメモリのバンク2またはバンク3へアクセスするときは、D[15:0]バス上のデータがラッチされて、ECC計算に使用されます。
- NAND Flashメモリの他のアドレスをアクセスするときは、ECCロジックはアイドル状態となり、動作しません。このため、NAND Flashメモリに対するコマンドまたはアドレスを定義する書込み動作は、ECC計算で考慮されません。

目的のバイト数がホストCPUによってNAND Flashメモリから読み出されるか、NAND Flashメモリに書き込まれたら、FMC_ECCRレジスタを読み出して、計算値を取得する必要があります。読出し後は、ECCENビットを"0"にリセットすることによってクリアしなくてはなりません。新しいデータブロックを計算するには、FMC_PCRレジスタのECCENビットを1にセットする必要があります。

ECC計算を行うときは：

1. FMC_PCRレジスタでECCENビットを有効にします。
2. NAND型Flashメモリ･ページへデータを書き込みます。NANDページへ書き込む際、ECCブロックはECC値を計算します。
3. FMC_ECCRレジスタにあるECC値を読み出して、変数内に格納します。
4. ECCENビットをクリアし、FMC_PCRレジスタで有効にした後、書き込んだデータをNANDページから読み出します。NANDページを読み出す際、ECCブロックはECC値を計算します。
5. FMC_ECCRレジスタにある新しいECC値を読み出します。
6. 2つのECC値が同じ場合、訂正は不要です。その他の場合、ECCエラーが存在するためソフトウェア訂正ルーチンがエラーが訂正できるか否かの情報を返します。

27.7.7 NAND Flashコントローラ･レジスタ

NAND Flash制御レジスタ（FMC_PCR）

アドレスオフセット：0x80

リセット値：0x0000 0018

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS[2:0]			TAR3
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAR[2:0]			TCLR[3:0]				Res.	Res.	ECCEN	PWID[1:0]		PTYP	PBKEN	PWAITEN	Res.
rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:17 **ECCPS[2:0]**：ECCのページサイズ

外部ECCのページサイズを定義します：

000：256バイト

001：512バイト

010：1024バイト

011：2048バイト

100：4096バイト

101：8192バイト

ビット16:13 **TAR[3:0]**：ALEからRE間の遅延

ALEローからREローまでの時間をAHBクロックサイクル（HCLK）数で設定します。

時間は、t_ar =（TAR + SET +2）× THCLKです。THCLKはHCLKクロック周期です。

0000：1HCLKサイクル（デフォルト）

1111：16HCLKサイクル

注： SETは、アドレス指定される空間に応じて、MEMSETまたはATTSETです。

ビット12:9 **TCLR[3:0]**：CLEからRE間の遅延

CLEローからREローまでの時間をAHBクロックサイクル（HCLK）数で設定します。

時間は、t_clr = (TCLR + SET + 2) * THCLKです。THCLKはHCLKクロック周期です。

0000：1HCLKサイクル（デフォルト）

1111：16HCLKサイクル

注： SETは、アドレス指定される空間に応じて、MEMSETまたはATTSETです。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **ECCEN**：ECC計算ロジックイネーブルビット

0：ECCロジックは無効であり、リセットされています（リセット後のデフォルト）。

1：ECCロジックは有効です。

ビット5:4 **PWID[1:0]**：データバス幅

外部メモリデバイスの幅を定義します。

00：8bit

01：16bit（リセット後のデフォルト）

10：予約済み

11：予約済み

ビット3 **PTYP**：メモリタイプ

対応するメモリバンクに接続されたデバイスのタイプを定義します。

0：予約済み、リセット値に保持する必要があります。

1：NAND Flash（リセット後のデフォルト）

ビット2 **PBKEN**：NAND型Flashメモリバンクイネーブルビット

メモリバンクを有効にします。無効なメモリバンクにアクセスすると、AHBバスのERRORになります。

0：対応するメモリバンクは無効です（リセット後のデフォルト）。

1：対応するメモリバンクは有効です。

ビット1 **PWAITEN**：ウェイト機能イネーブルビット

NAND型Flashメモリ･バンクのウェイト機能を有効にします。

0：無効

1：有効

ビット0 予約済みであり、リセット値に保持する必要があります。

FIFOステータスおよび割込みレジスタ（FMC_SR）

アドレスオフセット：0x84

リセット値：0x0000 0040

このレジスタは、FIFOステータスと割込みに関する情報を格納します。FMCには、AHBの16ワードまでのデータをメモリに書き込んで格納するときに使用されるFIFOがあります。

これは、FMCがFIFOをメモリへ転送すると同時に、FIFOに迅速に書き込んで、FMC以外のペリフェラルへのトランザクションにAHBを解放するために使用されます。これらのレジスタビットの1つは、ECCのために、FIFOのステータスを示します。

ECCは、データのメモリへの書込み中に計算されます。正しいECCを読み出すため、ソフトウェアはFIFOがエンプティになるまで待つ必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
									r	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **FEMPT**：FIFOエンプティ

FIFOのステータスを表示する読み取り専用ビット

0：FIFOノットエンプティ

1：FIFOエンプティ

ビット5 **IFEN**：割込み立下がりエッジ検出イネーブルビット

0：割込み立下がりエッジ検出リクエストは無効

1：割込み立下がりエッジ検出リクエストは有効

ビット4 **ILEN**：割込みハイレベル検出イネーブルビット

0：割込みハイレベル検出リクエストは無効

1：割込みハイレベル検出リクエストは有効

ビット3 **IREN**：割込み立上がりエッジ検出イネーブルビット

0：割込み立上がりエッジ検出リクエストは無効

1：割込み立上がりエッジ検出リクエストは有効

ビット2 **IFS**：割込み立下がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立下がりエッジは発生しませんでした。

1：割込み立下がりエッジが発生しました。

注： ソフトウェアがこのビットに1を書き込むとセットされます。

ビット1 **ILS**：割込みハイレベルステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込みハイレベルは発生しませんでした。

1：割込みハイレベルが発生しました。

ビット0 **IRS**：割込み立上がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立上がりエッジは発生しませんでした。

1：割込み立上がりエッジが発生しました。

注： ソフトウェアがこのビットに1を書き込むとセットされます。

共通メモリ空間タイミングレジスタ（FMC_PMEM）

アドレスオフセット：アドレス：0x88

リセット値：0xFCFC FCFC

FMC_PMEM読出し／書込みレジスタは、NAND Flashメモリ･バンクのタイミング情報を格納します。この情報は、NAND Flashの共通メモリ空間に、コマンド、アドレスの書込みアクセス、およびデータ読出し／書込みアクセスするために使われます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MEMHIZ[7:0]								MEMHOLD[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMWAIT[7:0]								MEMSET[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **MEMHIZ[7:0]**：共通メモリxデータバスハイインピーダンス時間

ソケット上の共通メモリ空間へのNAND Flashによる書込みアクセスの開始後に、データ･バスをハイインピーダンスに保つHCLKクロック･サイクル数を定義します。これは、書込みトランザクションに対してのみ有効です。

0000 0000: 1HCLKサイクル

1111 1110: 255HCLKサイクル

1111 1111：予約済み

ビット23:16 **MEMHOLD[7:0]**：共通メモリホールド時間

ソケットx上の共通メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書込みアクセス時はデータも）保持する書込みアクセス時のHCLKクロックサイクル数と読出しアクセス時のHCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001: 書込みアクセスの場合1HCLKサイクル／読出しアクセスの場合3HCLKサイクル

1111 1110: 書込みアクセスの場合254HCLKサイクル／読出しアクセスの場合256HCLKサイクル

1111 1111：予約済み

ビット15:8 **MEMWAIT[7:0]**：共通メモリウェイト時間

ソケット上の共通メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンド（NWE、NOE）をアサートするためのHCLK（＋1）クロックサイクルの最小数を定義します。プログラムされたHCLK値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001: 2HCLKサイクル（＋ NWAITのデアサーションによって導入されるウェイトサイクル）

1111 1110: 255HCLKサイクル（＋ NWAITのデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット7:0 **MEMSET[7:0]**：共通メモリxセットアップ時間

ソケットx上の共通メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンド･アサーション（NWE、NOE）前にアドレスをセットアップするためのHCLK（+1）クロック･サイクル数を定義します。

0000 0000: 1HCLKサイクル

1111 1110: 255HCLKサイクル

1111 1111：予約済み

属性メモリ空間タイミングレジスタ（FMC_PATT）

アドレスオフセット：0x8C

リセット値：0xFCFC FCFC

FMC_PATT読出し／書込みレジスタは、NAND Flashメモリバンクのタイミング情報を格納します。この情報は、最後のアドレス書込みアクセスの際に、タイミングが以前のアクセスのタイミングと異なる必要がある場合に、NAND Flashの属性メモリ空間へ8bitアクセスするために使用されます（レディ／ビジー管理については、セクション 27.7.5：NAND Flashのプリウェイト機能を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ATTHIZ[7:0]								ATTHOLD[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATTWAIT[7:0]								ATTSET[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **ATTHIZ[7:0]**：属性メモリデータバスハイインピーダンス時間

ソケット上の属性メモリ空間へのNAND Flashによる書込みアクセスの開始後に、データ･バスをハイインピーダンスに保つHCLKクロック･サイクル数を定義します。これは、書込みトランザクションに限り有効です。

0000 0000: 0HCLKサイクル

1111 1110: 255HCLKサイクル

1111 1111：予約済み

ビット23:16 **ATTHOLD[7:0]**：属性メモリホールド時間

ソケット上の属性メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書込みアクセス時はデータも）保持する書込みアクセス時のHCLKクロックサイクル数と読出しアクセス時のHCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001: 書込みアクセスの場合1HCLKサイクル／読出しアクセスの場合3HCLKサイクル

1111 1110: 書込みアクセスの場合254HCLKサイクル／読出しアクセスの場合256HCLKサイクル

1111 1111：予約済み

ビット15:8 **ATTWAIT[7:0]**：属性メモリウェイト時間

ソケットx上の属性メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンド（NWE、NOE）をアサートするためのHCLK（＋1）クロック･サイクルの最小数を定義します。プログラムされたHCLK値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001: 2HCLKサイクル（＋ NWAITのデアサーションによって導入されるウェイトサイクル）

1111 1110: 255HCLKサイクル（＋ NWAITのデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット7:0 **ATTSET[7:0]**：属性メモリセットアップ時間

ソケット上の属性メモリ空間へのNAND Flashによる読出しまたは書込みアクセスの際に、コマンド･アサーション（NWE、NOE）前にアドレスをセットアップするためのHCLK（＋1）クロック･サイクル数を定義します。

0000 0000: 1HCLKサイクル

1111 1110: 255HCLKサイクル

1111 1111：予約済み

ECC結果レジスタ（FMC_ECCR）

アドレスオフセット：0x94

リセット値：0x0000 0000

このレジスタは、FMCNANDコントローラのECC計算モジュールによって計算された現在のエラー訂正コード値を格納します。CPUが、NAND Flashメモリ･ページ上の正しいアドレスからデータを読み出すときには（セクション 27.7.6：NAND型Flashメモリ内のエラー訂正コード（ECC）の計算を参照）、NAND Flashメモリに対して読み書きされたデータは、ECC計算モジュールによって自動的に処理されます。Xバイトを読み出した後（FMC_PCRレジスタのECCPSフィールドに従って）、CPUはFMC_ECCレジスタから計算されたECC値を読み出す必要があります。次に、これらの計算されたパリティデータがスペア領域に記録されているパリティ値と一致するか検証して、ページが有効か否か、訂正が必要か否かを確認します。FMC_ECCRレジスタは、読出し後、ECCENビットを0に設定することによってクリアする必要があります。新しいデータ･ブロックを計算するときは、ECCENビットを1にセットする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ECC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ECC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **ECC[31:0]**：ECCの結果

このフィールドは、ECC計算ロジックによって計算された値を格納します。表245に、これらのビットフィールドの値を示します。

表245.ECC結果関連ビット

ECCPS[2:0]	ページサイズ（バイト数）	ECCビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]

表245.ECC結果関連ビット （続き）

ECCPS[2:0]	ページサイズ（バイト数）	ECCビット
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]

27.7.8 FMC レジスタマップ

表246.FMCレジスタマップとリセット値

オフセット	レジスタ名 リセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMC_BCR1	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		WFDIS	CCLKEN	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値	0								0	0	0	0	0	0	0	0	0	0	1	1	0		0	0		1	0	1	1	0	1	1
0x08	FMC_BCR2	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値	0								0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x10	FMC_BCR3	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値	0								0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x18	FMC_BCR4	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値	0								0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x04	FMC_BTR1	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0C	FMC_BTR2	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	FMC_BTR3	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	FMC_BTR4	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x20	FMC_ PCSCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTB4EN	CNTB3EN	CNTB2EN	CNTB1EN	CSCOUNT[15:0]															
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表246.FMCレジスタマップとリセット値 （続き）

オフセット	レジスタ名 リセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x104	FMC_BWTR1	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x10C	FMC_BWTR2	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x114	FMC_BWTR3	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x11C	FMC_BWTR4	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0].								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x80	FMC_PCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS [2:0]				TAR[3:0]				TCLR[3:0]				Res.	Res.	ECCEN	PWID [1:0]		PTYP	PBKEN	PWAITEN	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0			0	0	1	1	0	0	
0x84	FMC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
	リセット値																										1	0	0	0	0	0	0
0x88	FMC_PMEM	MEMHIZx[7:0]								MEMHOLDx[7:0]								MEMWAITx[7:0]								MEMSETx[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x8C	FMC_PATT	ATTHIZ[7:0]								ATTHOLD[7:0]								ATTWAIT[7:0]								ATTSET[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x94	FMC_ECCR	ECCx[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

28 Octo-SPIインタフェース（OCTOSPI）

28.1 概要

OCTOSPIは、シリアルPSRAM、シリアルNANDおよびシリアルNOR Flashメモリ、HyperRAM™、HyperFlash™メモリなど、ほとんどの外部シリアルメモリを以下の機能モードでサポートしています。

- インダイレクトモード : すべての動作は、コマンド、アドレス、データ、および転送パラメータをプリセットするOCTOSPIレジスタを使用して実行されます。
- 自動ステータスポーリングモード : 外部メモリステータスレジスタが定期的に読み出され、フラグのセット時に割込みを生成することができます。この機能は、レギュラコマンドプロトコルでのみ使用できます。
- メモリマップドモード : 外部メモリはメモリ空間に配置され、システムからは内部メモリであるかのように見え、読出しと書込みの両方の動作がサポートされます。

OCTOSPIは、以下の関連するフレームフォーマットを備えたプロトコルをサポートしています。

- コマンド、アドレス、オルタネートバイト、ダミーサイクル、およびデータフェーズから成るレギュラコマンドフレームフォーマット
- HyperBus™フレームフォーマット

28.2 OCTOSPIの主な機能

- 機能モード : インダイレクト、自動ステータスポーリング、およびメモリマップドモード
- メモリマップドモードでの読出しおよび書込みのサポート
- 外部（P）SRAMメモリのサポート
- シングル、デュアル、クワッド、およびオクタル通信に対応
- 2つのクワッドメモリに並列にアクセスすることで8bitの同時送受信を可能にするデュアルメモリ構成
- SDR（シングルデータレート）およびDTR（ダブル転送レート）のサポート
- データストローブのサポート
- 完全にプログラム可能なOPコード
- 完全にプログラム可能なフレームフォーマット
- 読出し方向のメモリへのラップタイプのアクセスをサポート
- HyperBusのサポート
- 受信および送信用の内蔵FIFO
- 非同期バスクロックとカーネルクロックのサポート
- 8、16、32bitのデータアクセスを許可
- DMAのプロトコルに対応
- インダイレクトモードで動作するためのDMAチャネル
- FIFO閾値、タイムアウト、動作完了、アクセスエラーによる割込み生成
- トランザクションの受け入れが1つに制限されたAHBインタフェース : メモリ側で前の転送が完了した後でのみ、AHBバス上で次の転送を受け入れるインタフェース

28.3 OCTOSPIの実装

表247.STM32U5シリーズデバイス上のインスタンス

デバイス	OCTOSPI1	OCTOSPI2	OCTOSPIM	HSPI1
STM32U535/545	X	-	-	-
STM32U575/585	X	X	X	-
STM32U59x/5Ax	X	X	X	X
STM32U5Fx/5Gx	X	X	X	X

表248.OCTOSPI/HSPIの実装

OCTOSPIの機能	OCTOSPI1/2	HSPI1
HyperBus規格準拠	X	X
Xcella規格準拠	X	X
XSPI（JEDEC251ES）規格準拠	X	X
AMBA®AHB準拠のデータインタフェース	X	X
デュアルAHBインタフェース	X	X
非同期AHBクロックとカーネルクロック	X	X
機能モード：インダイレクト、自動ステータスポーリング、およびメモリマップドモード	X	X
メモリマップドモードでの読出しおよび書込みのサポート	X	X
デュアルクワッド構成	X	X
デュアルオクタル構成	-	X
SDR（シングルデータレート）およびDTR（ダブル転送レート）	X	X
データストローブ（DS、DQS）	X	X
完全にプログラム可能なOPコード	X	X
完全にプログラム可能なフレームフォーマット	X	X
受信および送信用の内蔵FIFO	X	X
8、16、32bitのデータアクセス	X	X
FIFO閾値、タイムアウト、動作完了、アクセスエラーによる割込み	X	X
デュアルOCTOSPIアービタ（通信調整）に準拠	X	-
拡張CSHTタイムアウト	X	X
メモリマップド書込み	X	X
リフレッシュカウンタ	X	X
GPDMAインタフェース	X	X
ハイスピードインタフェース	-	X
プリフェッチ無効化	-	-
プリフェッチ用ハードウェアおよびソフトウェア	-	-

28.4 OCTOSPIの機能詳細

28.4.1 OCTOSPIブロック図

図139. OCTOSPIブロック図（オクタル構成）
（STM32U535/545デバイスを除くSTM32U5シリーズ）

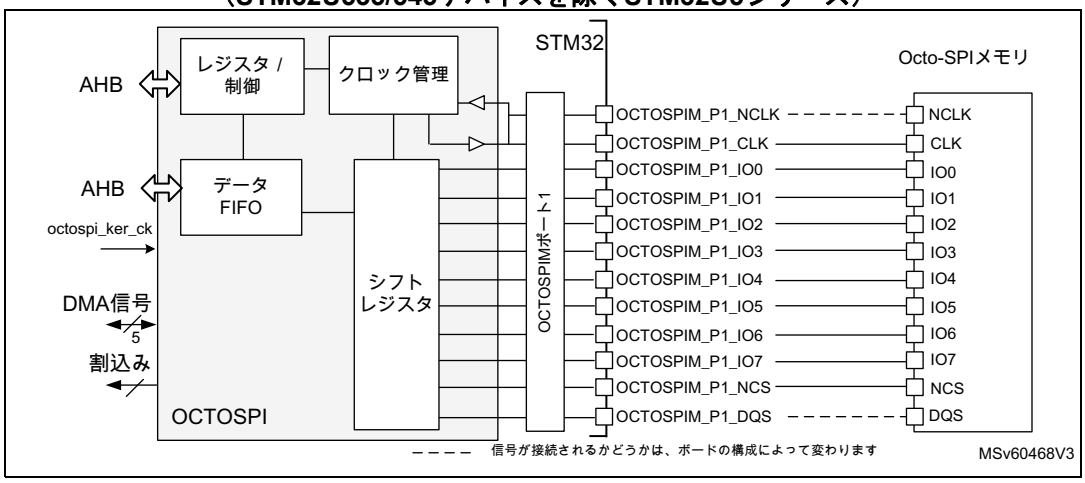


図140. OCTOSPIブロック図（クワッド構成）
（STM32U535/545デバイスを除くSTM32U5シリーズ）

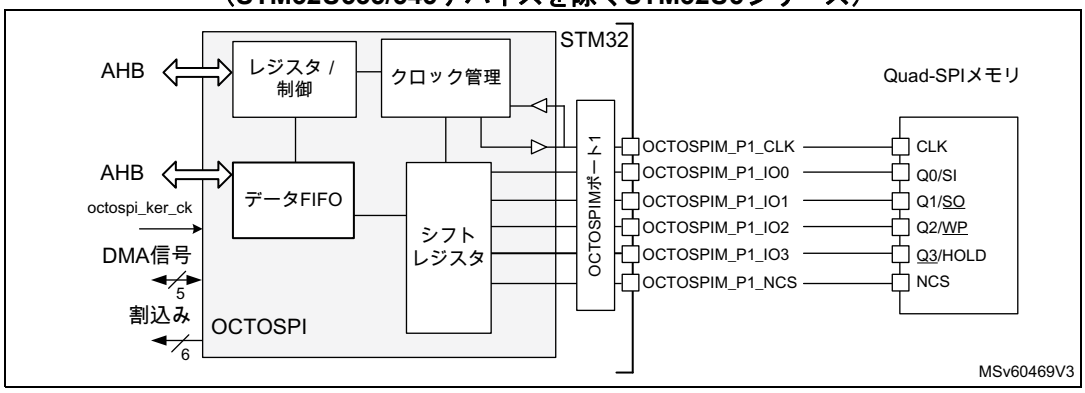


図141. OCTOSPIブロック図（デュアルクワッド構成）
（STM32U535/545デバイスを除くSTM32U5シリーズ）

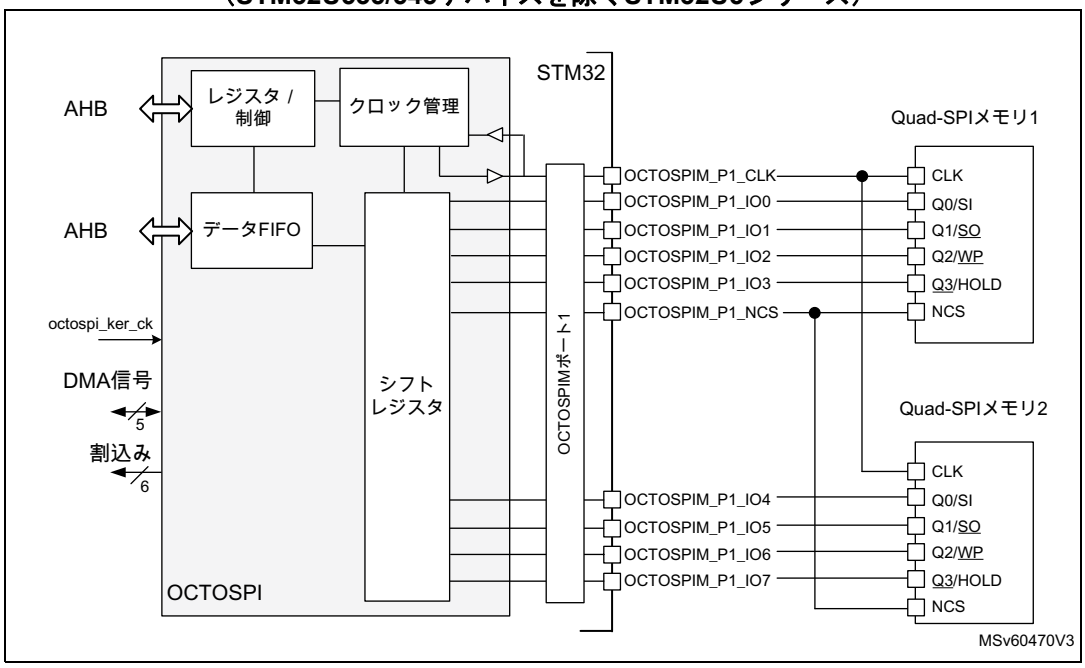


図142. OCTOSPIブロック図（オクタル構成）
（STM32U535/545デバイスのみ）

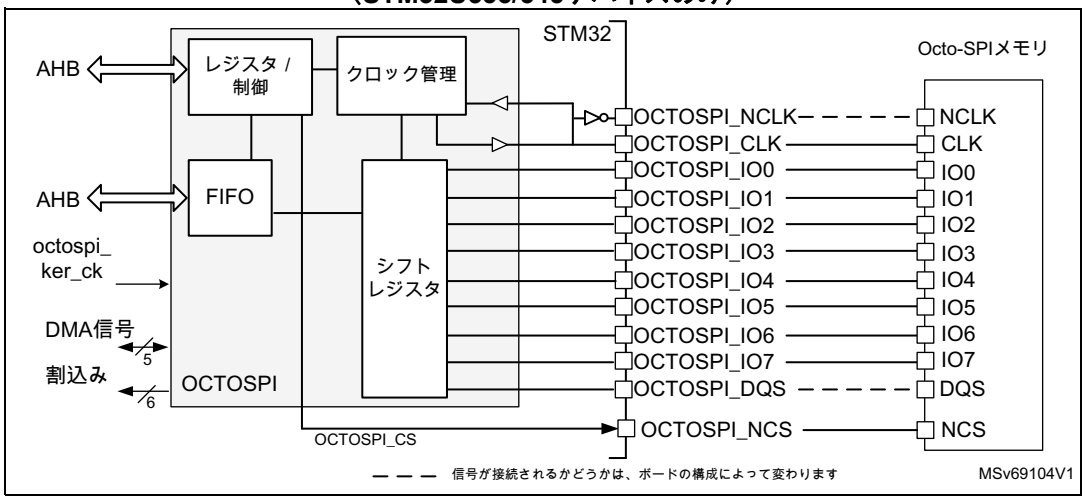


図143. OCTOSPIブロック図（クワッド構成）
（STM32U535/545デバイスのみ）

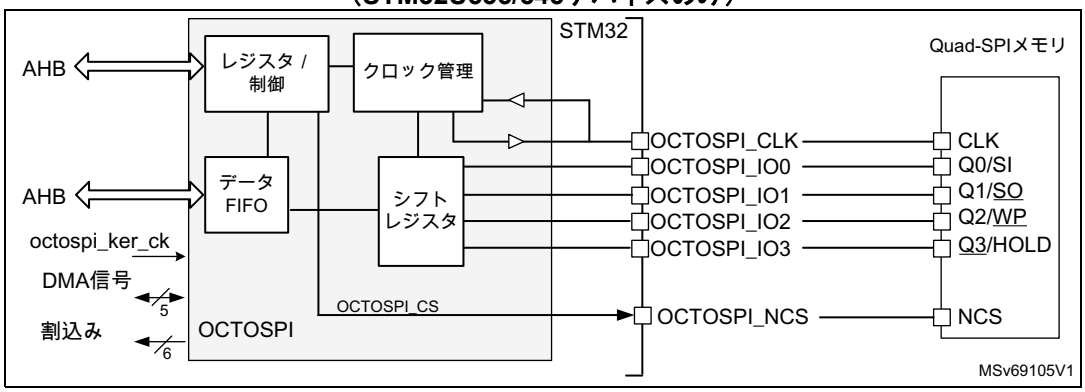
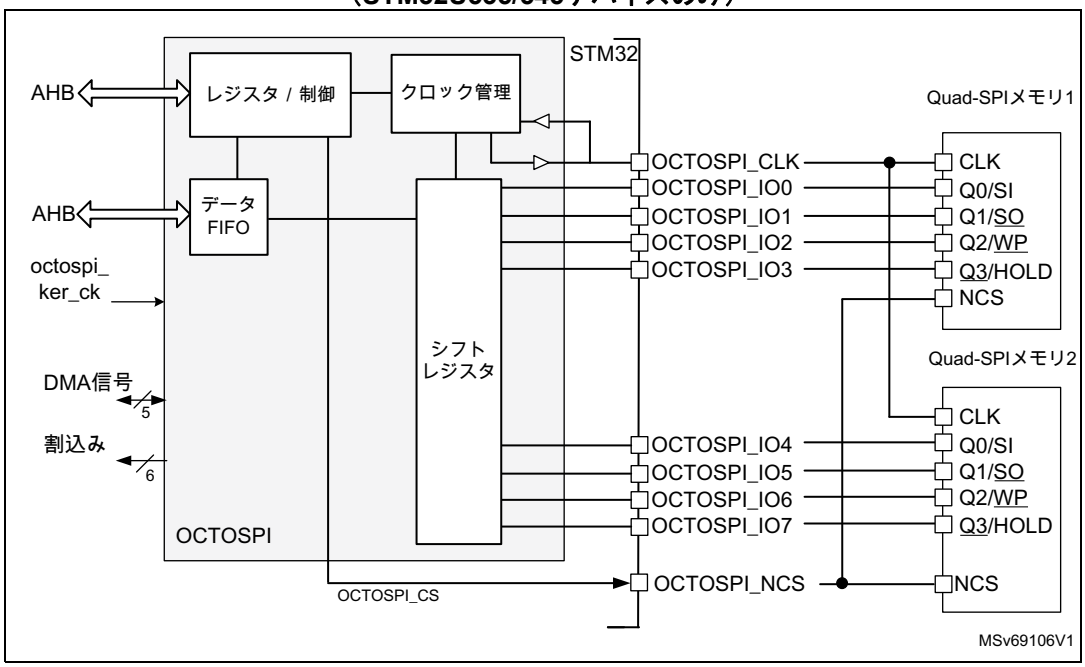


図144. OCTOSPIブロック図（デュアルクワッド構成）
（STM32U535/545デバイスのみ）



28.4.2 OCTOSPIピンおよび内部信号

表249.OCTOSPIの入出力ピン(1)（STM32U535/545デバイスを除くSTM32U5シリーズ）

ピン名	タイプ	説明
OCTOSPIM_Px_NCLK	出力	1.8V HyperBusプロトコルに対応したOCTOSPI反転クロック
OCTOSPIM_Px_CLK		OCTOSPIクロック
OCTOSPIM_Px_IOn（N = 0～7）	入力／出力	OCTOSPIデータピン
OCTOSPIM_Px_NCS	出力	メモリのチップセレクト
OCTOSPIM_Px_DQS	入力／出力	メモリとの間のデータストローブ／書込みマスク信号

1. x = 1～2

注意： データの破損を避けるために、すべてのOCTOSPI入出力ピンに同じ設定（出力速度、HSLV）を使用してください。

表250.OCTOSPI入出力ピン（STM32U535/545デバイスのみ）

ピン名	タイプ	説明
OCTOSPI_NCLK	出力	1.8V HyperBusプロトコルに対応したOCTOSPI反転クロック
OCTOSPI_CLK		OCTOSPIクロック
OCTOSPI_IOn（n = 0～7）	入力／出力	OCTOSPIデータピン
OCTOSPI_NCS	出力	メモリのチップセレクト
OCTOSPI_DQS	入力／出力	メモリとの間のデータストローブ／書込みマスク信号

注意： データの破損を避けるために、すべてのOCTOSPI入出力ピンに同じ設定（出力速度、HSLV）を使用してください。

表251.OCTOSPI内部信号

信号名	タイプ	説明
octospi_hclk	入力	OCTOSPI用AHBクロック
octospi_ker_ck	入力	OCTOSPIカーネルクロック
octospi_dma	N/A	DMAリクエスト信号
octospi_it	出力	グローバル割込みライン（割込みの複数ソースについては表254を参照）

28.4.3 OCTOSPIインタフェースを用いたメモリモード

OCTOSPIは次のプロトコルをサポートしています。

- レギュラコマンドプロトコル
- HyperBusプロトコル

OCTOSPIは機能モードに応じて、下記の6から12本の信号を使ってメモリとインタフェースを取ります。

- NCS：チップセレクト
- CLK：通信クロック
- NCLK：1.V HyperBusプロトコルでのみ使用される反転クロック
- DQS：レギュラコマンドプロトコルでのみ、入力としてのみ使用されるデータストローブ
- IO[3:0]：LSB用データバス
- IO[7:4]：
  - デュアルクワッド構成およびオクタル構成で使用されるMSB用データバス
  - データバスは、Quad-SPIモードのリマップとして使用できます

28.4.4 OCTOSPIレギュラコマンドプロトコル

レギュラコマンドプロトコルでは、OCTOSPIはコマンドを使って外部デバイスと通信します。各コマンドには以下の5つのフェーズを含めることができます。

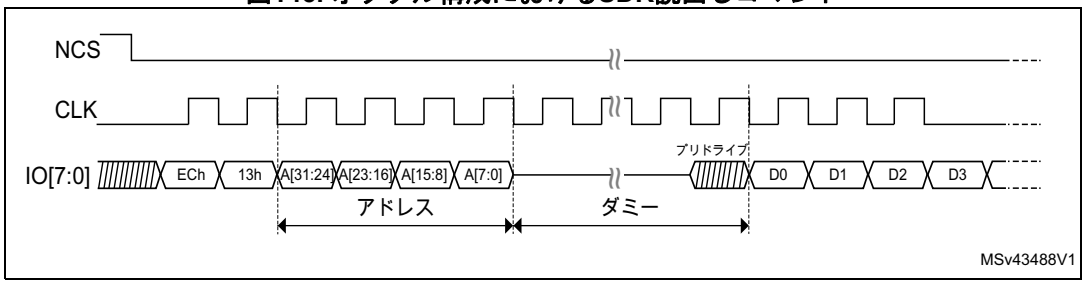
- 命令フェーズ
- アドレスフェーズ
- オルタネートバイトフェーズ
- ダミーサイクルフェーズ
- データフェーズ

これらのフェーズはいずれもスキップするように構成できますが、単一フェーズコマンドの場合、サポートされる使用例は命令フェーズのみです。

NCSは各コマンドの開始前に立下がり、各コマンドの終了後に再び立上ります。

メモリマップドモードでは、読出しと書込みの両方の動作がサポートされており、その結果、いくつかの設定レジスタは書込み動作を指定するために複製されます（読出し動作は通常のレジスタを使用して設定されます）。

図145. オクタル構成におけるSDR読出しコマンド



レギュラコマンドプロトコルの具体的な機能は0x0100～0x01FCのオフセット範囲のレジスタによって設定されます。

命令フェーズ

このフェーズでは、1～4バイトの命令が外部デバイスに送信され、実行すべき動作のタイプが指定されます。送信される命令のサイズはOCTOSPI_CCRのISIZE[1:0]で設定され、命令はOCTOSPI_IRのINSTRUCTION[31:0]にプログラムされます。

命令フェーズでは、オプションで以下を送信できます。

- 一度に1bit（Single-SPIモードでIO0、SO信号を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（Octal-SPIモードでIO0～IO7を経由して）。

これは、OCTOSPI_CCRのIMODE[2:0]を使用して設定できます。

命令は、OCTOSPI_CCRのIDTRをセットすることによって、DTRモードでクロックの立上がりおよび立下がりの各エッジで送信することができます。

OCTOSPI_CCRのIMODE[2:0] = 000の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

メモリマップドモードでは、書込み動作に使用される命令はOCTOSPI_WIRで指定され、命令フォーマットはOCTOSPI_WCCRで指定されます。読出し動作に使用される命令と命令フォーマットはOCTOSPI_IRとOCTOSPI_CCRで指定されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す1～4バイトが外部デバイスに送信されます。送信されるアドレスバイト数は、OCTOSPI_CCRのADSIZE[1:0]で設定されます。

インダイレクトモードと自動ステータスポーリングモードでは、送信されるアドレスバイトはQUADSPI_ARのADDRESS[31:0]によって指定されます。メモリマップドモードでは、アドレスは（システムのどのマスタからも）AHB経由で直接与えられます。

アドレスフェーズでは、以下のものを送信できます。

- 一度に1bit（Single-SPIモードでIO0、SO信号を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（Octal-SPIモードでIO0～IO7を経由して）

これは、OCTOSPI_CCRのADMODE[2:0]を使用して設定できます。

アドレスは、OCTOSPI_CCRのADDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送信することができます。

ADMODE[2:0] = 000の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

メモリマップドモードでは、書込み動作用のアドレスフォーマットはOCTOSPI_WCCRにて指定されます。読出し動作用のアドレスフォーマットはOCTOSPI_CCRにて指定されます。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作のモードを制御するために、1～4バイトが外部デバイスに送信されます。送信されるオルタネートバイト数はOCTOSPI_CCRのABSIZE[1:0]で設定されます。送信されるバイトはOCTOSPI_ABRで指定されます。

オルタネートバイトフェーズでは、以下のものを送信できます。

- 一度に1bit（Single-SPIモードでIO0、SO信号を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（Octal-SPIモードでIO0～IO7を経由して）

これは、OCTOSPI_CCRのABMODE[2:0]を使用して設定できます。

オルタネートバイトは、OCTOSPI_CCRのABDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送信することができます。

ABMODE[2:0] = 000の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

場合によっては、オルタネートバイトフェーズ中にフルバイトではなく2分の1バイト（4bit）だけを送信する必要があるかもしれません。たとえば、Dual-SPIモード使用時で、オルタネートバイトに2サイクルだけが使われる場合などです。

この場合、ファームウェアによってQuad-SPIモード（ABMODE[2:0] = 011）を使用し、ALTERNATE[31:0]のビット7および3を1（IO3ラインをハイに保持）に、OCTOSPI_IRのビット6および2を0（IO2ラインをローに保持）に設定して1バイトを送信できます。

送信される4bitの上位2bitはALTERNATE[31:0]のビット5:4にセットされ、下位2bitはビット1:0にセットされます。たとえば、4bitの値2（0010）がIO0/IO1で送信される場合、ALTERNATE[31:0]は0x8A（1000_1010）にセットされている必要があります。

メモリマップドモードでは、書込み動作用に使用されるオルタネートバイトはOCTOSPI_WABRで指定され、オルタネートバイト･フォーマットはOCTOSPI_WCCRで指定されます。読出し動作用に使用されるオルタネートバイトとオルタネートバイト･フォーマットはOCTOSPI_ABRとOCTOSPI_CCRで指定されます。

ダミーサイクルフェーズ（メモリ待ち時間）

ダミーサイクルフェーズでは、データを送受信せずに1～31回の空サイクルが設けられます。これは、より高いクロック周波数が使われている場合に、外部デバイスにデータフェーズの準備をする時間を与えるためです。このフェーズでのサイクル数は、OCTOSPI_TCRのDCYC[4:0]によって指定されます。SDRモードとDTRモードのどちらでも、この持続時間はフルCLKサイクル数として指定されます。

DCYC[4:0] = 00000の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

データ信号を出力モードから入力モードに変更するのに十分な「ターンアラウンド」時間を確保するために、Dual-SPI、Quad-SPI、またはOctal-SPIモードを使用して外部デバイスからデータを受信する場合、少なくとも1ダミーサイクルが必要です。

メモリマップドモードでは、書込み動作用のダミーサイクルはOCTOSPI_WTCRに指定されます。読出し動作用のダミーサイクルはOCTOSPI_TCRで指定されます。

データフェーズ

データフェーズでは、外部デバイスとの間で任意のバイト数を送受信できます。

インダイレクトモードでは、送受信するバイト数はOCTOSPI_DLRで指定されます。このモードでは、外部デバイスに送信するデータをOCTOSPI_DRに書き込む必要があります。一方、インダイレクト読出しモードでは、外部デバイスから受信するデータはOCTOSPI_DRを読み出して取得します。

自動ステータスポーリングモードでは、受信するバイト数はOCTOSPI_DLRにて指定され、外部デバイスから受信するデータはOCTOSPI_DRを読み出して取得できます。

メモリマップドモードでは、読み書きされるデータは、直接AHBを経由してCortexコアまたはDMAとの間で送受信されます。

データフェーズでは、以下のものを送受信できます。

- 一度に1bit（Single-SPIモードでIO0/IO1（SO/SIそれぞれ）を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（Octal-SPIモードでIO0～IO7を経由して）

これは、OCTOSPI_CCRのDMODE[2:0]を使用して設定できます。

データは、OCTOSPI_CCRのDDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送受信することができます。

DMODE[2:0] = 000の場合、データフェーズはスキップされるので、コマンドシーケンスはNCSを立ち上げることですぐに終了します。この設定はインダイレクト書込みモードでのみ使用してください。

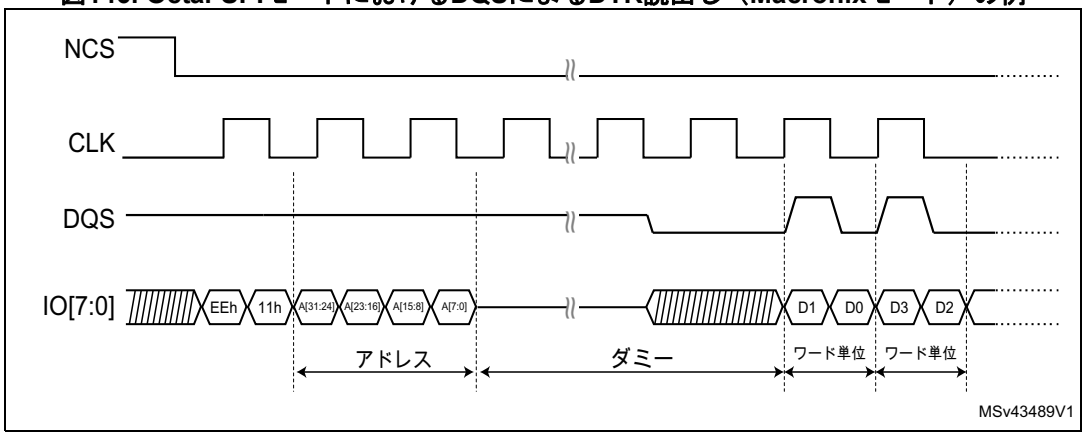
メモリマップドモードでは、書込み動作用のデータフォーマットはOCTOSPI_WCCRにて指定されます。読出し動作用のデータフォーマットはOCTOSPI_CCRにて指定されます。

DQSの使用

DQS信号は、デバイスがデータに合わせてDQSをトグルしているときの読出しトランザクション中のデータストローブに使用できます。

DQS管理はOCTOSPI_CCRのDQSEを設定することによって有効にできます。

図146. Octal-SPIモードにおけるDQSによるDTR読出し（Macronixモード）の例



28.4.5 OCTOSPIレギュラコマンドプロトコル信号インタフェース

Single-SPIモード

レガシー SPIモードでは、1bitの順次送受信のみが可能です。このモードでは、データはSO信号（そのI/OはIO0と共有される）を経由して外部デバイスに送信されます。外部デバイスから受信するデータは、SI（そのI/OはIO1と共有される）を経由して到着します。

OCTOSPI_CCRおよびOCTOSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ001にセットすることによって、このSingle-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをSingle-SPIモードに設定する場合：

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2は出力モードで、強制的に0に設定されます（書込み保護機能を無効にするため）。
- IO3は出力モードで、強制的に1に設定されます（ホールド機能を無効にするため）。
- IO4からIO7は出力モードで、強制的に0に設定されます。

これは、DMODE[2:0] = 001であれば、ダミーフェーズにも該当します。

Dual-SPIモード

Dual-SPIモードでは、IO0/IO1信号を経由して2bitが同時に送受信されます。

OCTOSPI_CCRおよびOCTOSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ010にセットすることによって、DUAL-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをDual-SPIモードに設定する場合：

- IO0/IO1は、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2は出力モードで、強制的に0に設定されます。
- IO3は出力モードで、強制的に1に設定されます。
- IO4からIO7は出力モードで、強制的に0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 010の場合、IO0/IO1は常にハイインピーダンスです。

Quad-SPIモード

Quad-SPIモードでは、IO0/IO1/IO2/IO3信号を経由して4bitが同時に送受信されます。

OCTOSPI_CCRおよびOCTOSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ011にセットすることによって、Quad-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをQuad-SPIモードに設定する場合：

- IO0～IO3はすべて、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO4からIO7は出力モードで、強制的に0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 011の場合、IO0～IO3はすべてハイインピーダンスです。

Octal-SPIモード

通常のOctal-SPIモードでは、IO[0:7]信号を経由して8bitが同時に送受信されます。

OCTOSPI_CCRおよびOCTOSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ100にセットすることによって、Octal-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

Octal-SPIモードに設定されている各フェーズでは、IO[0:7]はすべてデータフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE[2:0] = 100の場合、IO[0:7]はすべてハイインピーダンスです。

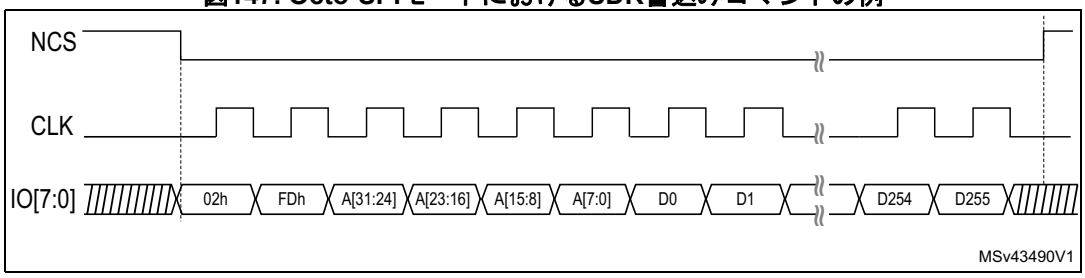
シングルデータレート（SDR）モード

すべてのフェーズは、デフォルトでSDRモードで動作します。

このモードでは、QUADSPIがIO0/SO、IO1～IO7信号を駆動している場合、これらの信号はCLKの立下がりエッジでのみ遷移します。

SDRモードでデータを受信する場合、OCTOSPIは外部デバイスもCLKの立下がりエッジを使用してデータを送信するとみなします。デフォルトでは（OCTOSPI_TCRにおいてSSHIFT = 0の場合）、信号は次のCLKの（立上がり）エッジを使用してサンプリングされます。

図147. Octo-SPIモードにおけるSDR書込みコマンドの例



注： **最後のデータが読み取られた後、内部同期のために、Octo-SPIインタフェースによって最大6つの追加のダミークロックサイクルが生成される場合があります。**

ダブル転送レート（DTR）モード

OCTOSPI_CCRのIDTR、ADDTR、ABDTR、およびDDTRをセットすることによって、命令、アドレス、オルタネートバイト、およびデータの各フェーズをDTRモードで動作するように設定できます。

メモリマップドモードでは、書込み動作の各フェーズのDTRモードはOCTOSPI_WCCRで指定されます。読出し動作の各フェーズのDTRモードはOCTOSPI_CCRで指定されます。

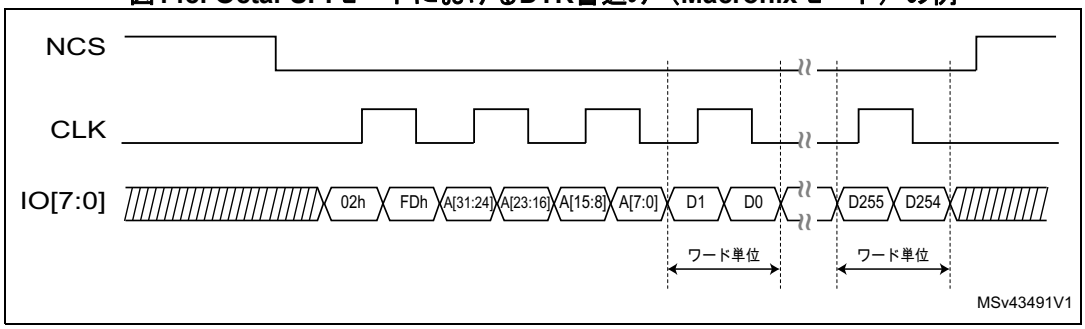
DTRモードでは、OCTOSPIが命令、アドレス、オルタネートバイトの各フェーズでIO0/SO、IO1～IO7信号を駆動している場合、CLKの立下がりエッジおよび立上がりエッジのそれぞれでビットを送受信します。

DTRモードでデータを受信する場合、OCTOSPIは外部デバイスもCLKの立上がりと立下がりの両方のエッジを使用してデータを送信するとみなします。OCTOSPI_CCRでDDTR = 1の場合、ソフトウェアはOCTOSPI_TCRのSSHIFTをクリアする必要があります。したがって、信号は1/2CLKサイクル後にサンプリングされます（下図の逆エッジで）。

DTRモードでは、OCTOSPI_TCRのDHQCをセットして、出力をクォータサイクルだけシフトし、メモリ側のホールド問題を避けることを推奨します。

注： **予期せぬ動作を引き起こす場合があるため、プリスケーラ値が0の場合はDHQCをセットしてはなりません。**

図148. Octal-SPIモードにおけるDTR書込み（Macronixモード）の例



注： **最後のデータが読み取られた後、内部同期のために、Octo-SPIインタフェースによって最大6つの追加のダミークロックサイクルが生成される場合があります。**

デュアルクワッド構成

OCTOSPI_CRにおいてDMM = 1の場合、OCTOSPIはデュアルメモリ構成です。すなわち、DMODE = 011の場合、2つの外部Quad-SPIデバイス（デバイスAとデバイスB）が使用され、サイクルごとに8bitの送受信（DTRモードでは16bit）を行うことで、スループットを効果的に2倍にします。

どちらのデバイス（AまたはB）も同じCLKとNCS信号を使用しますが、それぞれが個別のIO0～IO3信号を有しています。

デュアルクワッド構成は、SDRモードまたはDTRモードと組合せて使用できるだけでなく、Single-SPI、Dual-SPI、およびQuad-SPIモードとも組合せて使用できます。

OCTOSPI_DCR1のDEVSIZE[4:0]で指定されているデバイスのサイズは、外部デバイスの総容量（コンポーネント1個のサイズの2倍）を反映していなければなりません。

アドレスXが偶数の場合、OCTOSPIがアドレスXに対して与えるバイトはデバイスAのアドレスX/2にあるバイトであり、アドレスX+1に対して与えるバイトはデバイスBのアドレスX/2にあるバイトです。言い換えれば、偶数アドレスにあるバイトはすべてデバイスAに、奇数アドレスにあるバイトはすべてデバイスBに格納されます。

デュアルクワッド構成でデバイスのステータスレジスタを読み出す場合、レギュラコマンドプロトコルで読み出すバイト数の2倍のバイトを読み出す必要があります。各デバイスがステータスレジスタのフェッチ命令後に有効な8bitを提供する場合、OCTOSPIはデータ長2バイト（16bit）で設定する必要があり、それによりOCTOSPIは各デバイスから1バイトを受信します。

各デバイスが16bitのステータスを提供する場合、OCTOSPIは、4バイトを読み出して、デュアルクワッド構成の両方のデバイスのステータスビットをすべて取得できるように設定する必要があります。リザルト（データレジスタ内）の最下位バイトはデバイスAのステータスレジスタの最下位バイトです。次のバイトはデバイスBのステータスレジスタの最下位バイトです。さらに、データレジスタの3番目のバイトはデバイスAの2番目のバイトです。4番目のバイトはデバイスBの2番目のバイトになります（デバイスに16bitステータスレジスタがある場合）。

デュアルクワッド構成では、常に偶数バイトをアクセスする必要があります。このため、DMM = 1のとき、OCTOSPI_DLRのDL[31:0]のビット0は1のままとなります。

デュアルクワッド構成では、デバイスAのインタフェース信号の動作は基本的に通常モードの動作と同じです。デバイスBのインタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、デバイスAのものとまったく同じです。つまり、どちらのデバイスも常に同じ命令や同じアドレスを受信するということです。

データフェーズ中は、2つのバスAIOxとBIOxは並行してデータ転送を行いますが、デバイスAに対する送信（または受信）データはデバイスBのものとは異なります。

28.4.6 HyperBusプロトコル

OCTOSPIはHyperBusプロトコルを使って外部デバイスと通信することができます。

HyperBusは、動作電圧に応じて11から12ピンを使用します。

- 双方向データバスとしてIO[7:0]
- データ読み書きストローブおよび遅延挿入のためのRWDS（DQSピンに配置）
- NCS
- CLK
- 1.8V動作のNCLK（このモードをサポートするには、デバイスに1.8Vが供給されていなければなりません）

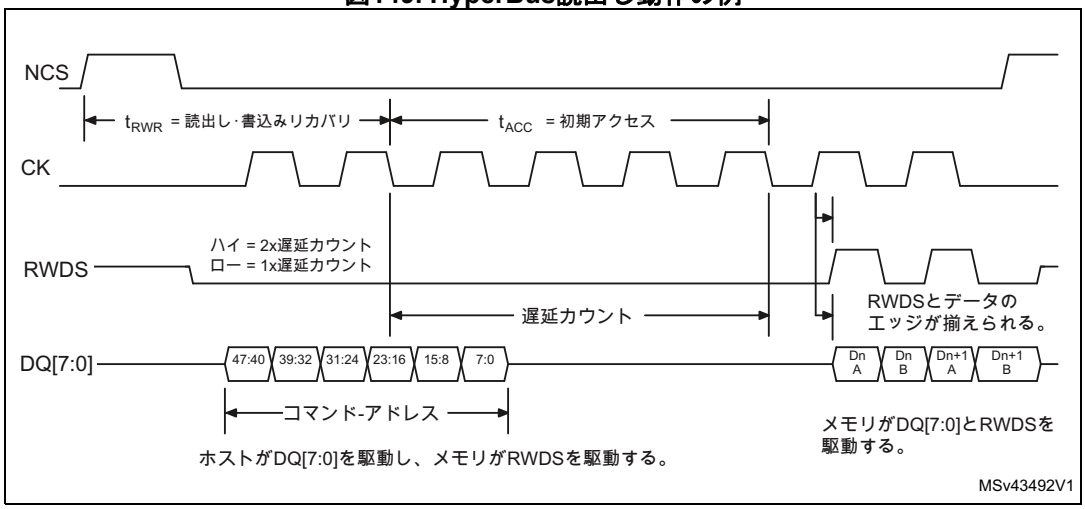
HyperBusでは、コマンド指定もオルタネートバイトも必要としません。結果として、トランザクションのタイミングを定義するために個別のレジスタセットが使用されます。

HyperBusのフレームは次の2つのフェーズで構成されます。

- コマンド／アドレスフェーズ
- データフェーズ

NCSはトランザクションの開始前に立下り、各トランザクションの終了後に再び立上ります。

図149. HyperBus読出し動作の例



注： **最後のデータが読み取られた後、内部同期のために、Octo-SPIインタフェースによって最大6つの追加のダミークロックサイクルが生成される場合があります。**

HyperBusの具体的な機能は0x0200～0x02FCのオフセット範囲のレジスタによって設定されます。

コマンド／アドレスフェーズ

この初期フェーズでは、OCTOSPIは外部デバイスとの間で実施する動作を指定するために、IO[7:0]を経由して48bitを送信します。

表252.コマンド／アドレスフェーズの説明

CAビット	ビット名	説明
47	R/W#	トランザクションが読出しか書込みかを識別します。
46	アドレス空間	トランザクションがメモリか、またはレジスタ空間にアクセスしているかを示します。
45	バーストタイプ	バーストがリニアか、またはラップトであるかを示します。
44 ~ 16	行アドレスおよび上位列アドレス	行アドレスおよび上位列アドレスを選択します。
15～3	予約済み	-
2～0	下位列アドレス	ハーフページ内の開始16bitワードを選択します。

アドレス空間はOCTOSPI_DCR1のメモリタイプを示すMTYP[2:0]に設定します。

デバイスの合計サイズはOCTOSPI_DCR1のDEVSIZE[4:0]に設定します。マルチチップ製品（MCP）の場合、デバイスサイズはすべてのMCPダイのすべてのサイズの合計です。

初期遅延を設けた読出し／書込み動作

HyperBusの読出しおよび書込み動作では、次の2つのタイミングを考慮する必要があります。

- t_{RWR}：デバイスの最小限の読出し／書込みリカバリ時間（OCTOSPI_HLCRのTRWR[7:0]で定義）
- t_{ACC}：メモリ待ち時間に応じたデバイスへのアクセス時間（OCTOSPI_HLCRのTACC[7:0]で定義）

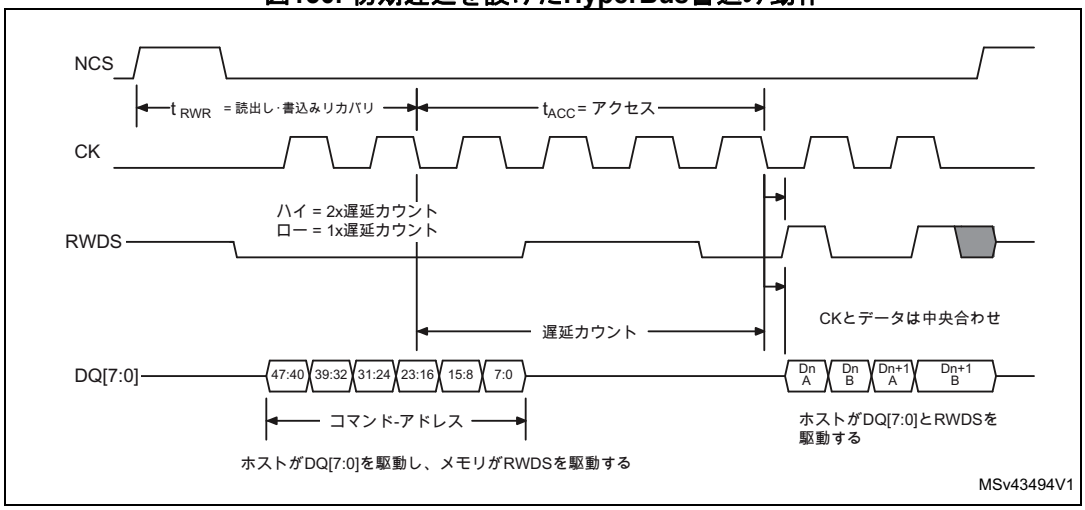
読出し動作中、デバイスはRWDSを次の2通りの方法で使用します（図149を参照）。

- コマンド／アドレスフェーズ時は、追加の遅延を要求するため
- データフェーズ時は、データストローブのため

書込み動作中、RWDSは次のように使用されます。

- デバイスによって、コマンド／アドレスフェーズ時に追加の遅延を要求するために。
- OCTOSPIによって、データフェーズ時にデータ書込みマスキングのために。

図150. 初期遅延を設けたHyperBus書込み動作



追加の遅延を設けた読出し／書込み動作

デバイスに追加の遅延が必要な場合（たとえばSDRAMのリフレッシュ期間中）、コマンド／アドレスフェーズ中にRWDS信号の1つの期間中にRWDSを1につなげる必要があります。

デバイスの要求を満たすために、OCTOSPIによって追加のt_{ACC}期間が追加されます。

図151. 追加の遅延を設けたHyperBus読出し動作

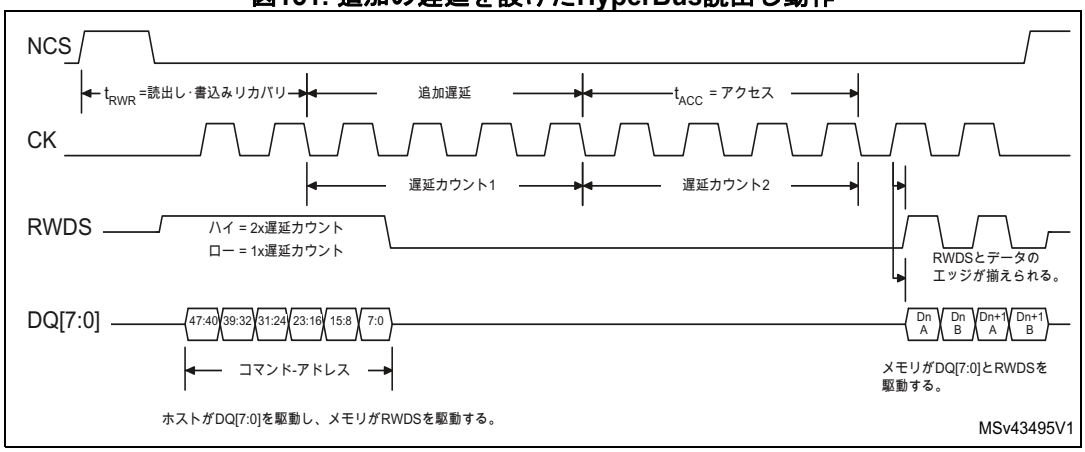
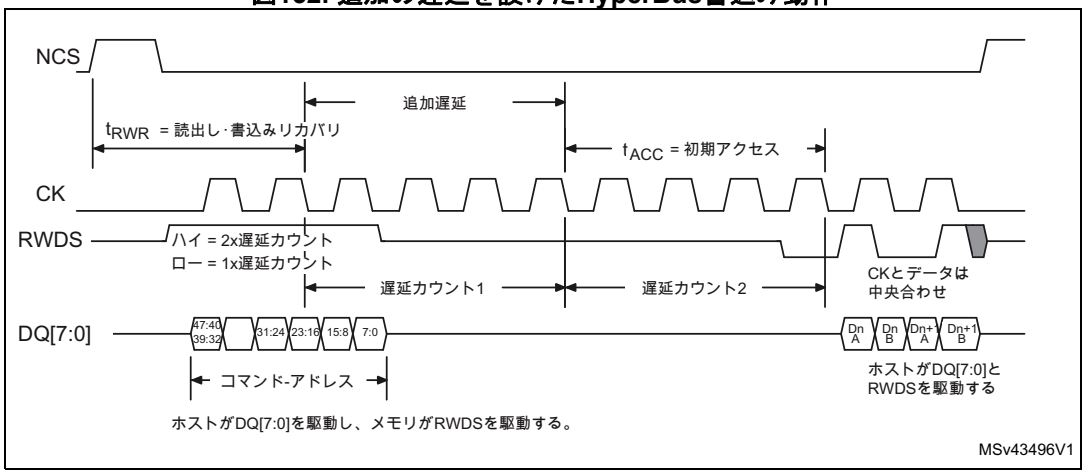


図152. 追加の遅延を設けたHyperBus書込み動作



固定-遅延モード

デバイスまたはアプリケーションによっては、上に述べたような可変の遅延を設けた動作を望まないものもあります。

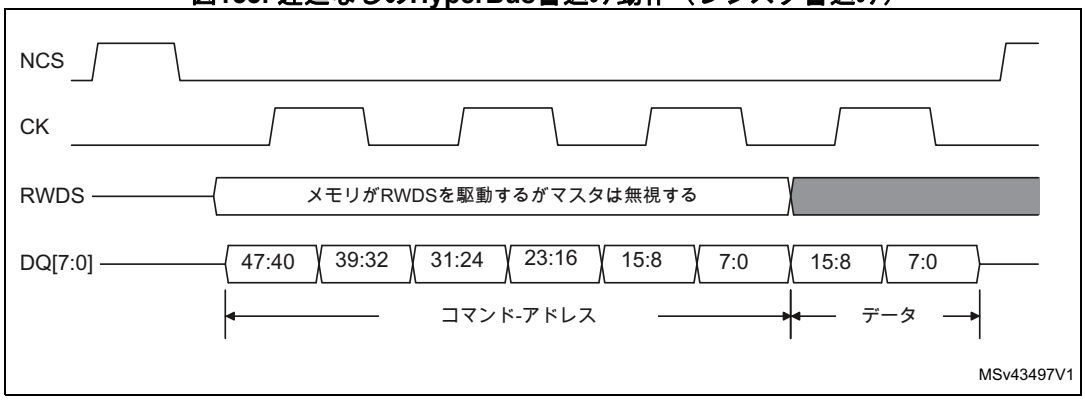
OCTOSPI_HLCRのLMをセットすることで、遅延を強制的に2 x t_{ACC}にすることができます。

このOCTOSPI遅延モードでは、RWDS信号の状態がOCTOSPIによって考慮されず、常に追加の遅延が追加されて、固定の2 x t_{ACC}の遅延時間が発生します。

遅延なしの書込み動作

また、デバイスによっては書込み動作にゼロ遅延を要求する可能性もあります。この書込みゼロ遅延は、OCTOSPI_HLCRのWZLを設定することによって強制することができます。

図153. 遅延なしのHyperBus書込み動作（レジスタ書込み）

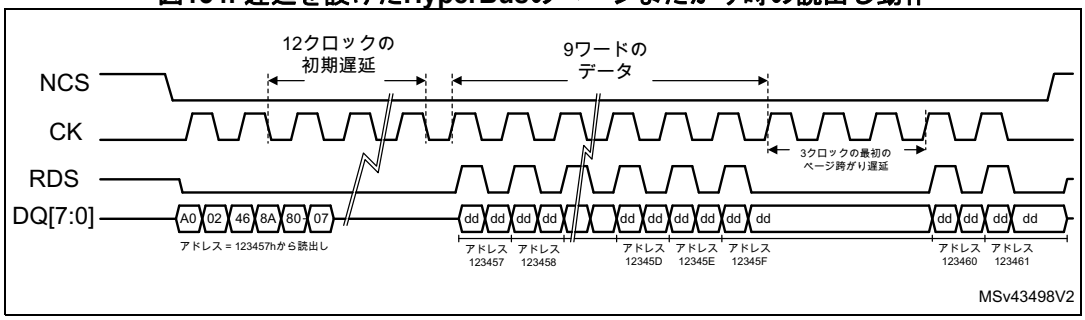


読出し動作中のページまたがり時の遅延

デバイスによっては、ページをまたがって読出し動作をする際に追加の遅延が必要になることがあります。

ページアクセスには初期遅延を考慮する必要があるので、その結果、最初のアクセスがページ境界に近いと、t_{ACC}時間を考慮するためにページまたがり時に自動的に遅延が追加されます。

図154. 遅延を設けたHyperBusのページまたがり時の読出し動作



28.4.7 特有の機能

OCTOSPIは次のようないくつかの特有の機能をサポートしています。

- ラップのサポート
- NCS境界とリフレッシュ
- 通信調整

ラップのサポート

OCTOSPIでは、HyperBusプロトコルで定義されているハイブリッドラップがサポートされています。ハイブリッドラップは、レギュラコマンドプロトコルでもサポートされます。

ハイブリッドラップでは、トランザクションは最初のラップの後もインクリメントアクセスで続行できます。

対象のメモリがサポートするラップサイズは、OCTOSPI_DCR2のWRAPSIZEで設定します。

ラップはメモリの読出し方向でのみサポートされます。また、データサイズ = 4バイトの場合のみサポートされます。ラップド読出しは、HyperBusとレギュラコマンドプロトコルの両方でサポートされます。ラップド読出しアクセスを有効にするには、ラップド読出しアクセス特性に応じた専用のOCTOSPI_WPxxxレジスタをプログラムする必要があります。これらのレジスタは、HyperBusとレギュラコマンドプロトコルの両方に適用されます。

対象のメモリがハイブリッドラップをサポートしていない場合は、WRAPSIZEは0にセットする必要があります。

注： **ラップ動作はリフレッシュによって中断することはできません。リフレッシュイベントはラップの完了後にのみ考慮されます。**

NCS境界とリフレッシュ

OCTOSPIトランザクションを調整するために次の2つのプロセスをアクティブにすることができます。

- NCS境界
- リフレッシュ

NCS境界機能によって、トランザクションは整列されたアドレスの境界に制限されます。整列させるアドレスのサイズは、OCTOSPI_DCR3のCSBOUND[4:0]で設定され、$2^{CSBOUND}$に等しくなります。

たとえば、CSBOUND[4:0] = 0x4の場合、境界は2^4 = 16バイトにセットされます。そうすると、LSBアドレスが0xFに等しくなるたびに、および新しいトランザクションが発行されて次のデータに取りかかるたびに、NCSが解除されます。

CSBOUND[4:0] = 0の場合、この機能は無効になります。最小値の3が推奨されます。

書込みモードでは、Flashメモリの他のページをプログラムするためにコマンドが必要なので、NCS境界機能は書込みモードのFlashメモリデバイスには使用できません。

リフレッシュ機能では、トランザクションの期間をOCTOSPI_DCR4のREFRESH[31:0]にプログラムされた値に制限します。期間はサイクル数で表されます。これにより、外部RAMは定期的に内部リフレッシュ動作を実行することができます。

リフレッシュ値は、コマンド／アドレス／オルタネート／ダミーフェーズを含む最小トランザクションサイズ（サイクル数の単位で）より大きくなければなりません。

NCS境界とリフレッシュが同時に有効にされた場合、NCSは最初の条件が合致した時に解除されます。

通信調整

通信調整機能では、トランザクションの最大期間を、OCTOSPI_DCR3のMAXTRAN[7:0]でプログラムされた値に制限します。

クロックサイクル数がMAXTRAN + 1の値に達し、また2番目のOCTOSPIがアクセスをリクエストした場合、NCSがリリースされ、次のデータに取りかかかるために新しトランザクションが発行されます。2番目のOCTOSPIがアクセスをリクエストしなかった場合は、トランザクションは停止せず、またNCSはリリースされません。

MAXTRAN[7:0] = 0の場合、制限は発生しません。

MAXTRAN[7:0]の値は、コマンド、アドレス、オルタネート、ダミーフェーズを含む最小トランザクションサイズより（サイクル数の単位で）大きくなければなりません。

注： **通信調整機能は、NCSが再びアクティブになった後に書込み動作を再有効化するための追加コマンドを必要とするFlashメモリデバイスの書込みモードでは使用できません。**

NCS境界、リフレッシュ、通信調整が同時に有効にされた場合、NCSは最初の条件が合致した時にリリースされます。

中断された転送後の再開始

読取りまたは書込み動作がタイムアウトまたは通信調整機能によって中断された場合、Octo-SPIインタフェースでは、ポートの所有権を取り戻した後、できるだけ早く、中断前に実際にアクセスした最後のアドレスに続くアドレスとともに最初のコマンドシーケンスを再発行します。最初に設定された転送は継目なく進行し、終了します。

28.4.8 OCTOSPIの動作モードの概要

OCTOSPIには、使用されるローレベルのプロトコル（レギュラコマンドまたはHyperBusのいずれか）に関係なく、次の動作モードがあります。

- インダイレクトモード（読出しまたは書込み）
- 自動ステータスポーリングモード（レギュラコマンドプロトコルでのみ）
- メモリマップドモード

28.4.9 OCTOSPIインダイレクトモード

インダイレクトモードでは、OCTOSPIレジスタに書き込むことでコマンドが開始され、データレジスタへの書込み／読出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

OCTOSPI_CRにおいてFMODE[1:0] =00の場合、OCTOSPIはインダイレクト書込みモードです。このモードでは、データフェーズ中に外部デバイスへバイトが送信されます。データは、OCTOSPI_DRへの書込みによって提供されます。

FMODE[1:0] = 01の場合、OCTOSPIはインダイレクト読出しモードです。このモードでは、データフェーズ中に外部デバイスからバイトが受信されます。データは、OCTOSPI_DRの読出しによって復元されます。

インダイレクトモードでは、DQSは無効の状態で、OCTOSPIが8つのレーンにわたってDTRモードに設定されている場合、指定する開始アドレスとデータ長は偶数でなければなりません。

注： **開始アドレスが前回のインダイレクトアクセスの開始アドレスと同じであっても、OCTOSPI_AR レジスタを更新する必要があります。**

読出し／書込みするバイト数はOCTOSPI_DLRで指定されます。

- DL[31:0] = 0xFFFF FFFFの場合、データ長は不定長とみなされ、OCTOSPIは単純に外部デバイスの最後（DEVSIZEによって定義される）に達するまでデータを転送し続けます。転送されるバイトが無い場合は、OCTOSPI_CCRのDMODE[2:0]を0にセットする必要があります。
- DL[31:0] = 0xFFFF FFFFでDEVSIZE[4:0] = 0x1F（最大値は4GBのデバイスを示す）の場合、転送は無期限に続けられ、アボートリクエスト後またはOCTOSPIの無効化後にやっと停止します。最後のメモリアドレスが（アドレス0xFFFF FFFFで）読み出された後も、アドレス = 0x0000 0000で読出しが続けられます。

プログラムされた送信または受信バイト数に達すると、OCTOSPI_SRにTCFビットがセットされ、OCTOSPI_CRにおいてTCIE = 1の場合に割込みが生成されます。データ数が定義されていない場合、外部SPIメモリの限界に達した時点で、OCTOSPI_DCR1に定義されているデバイスのサイズに応じてTCFがセットされます。

レギュラコマンドプロトコルでの転送開始のトリガ

レギュラコマンドプロトコルを使用している場合、インダイレクトモードで転送を開始する方法は、OCTOSPIの設定に応じて3通りあります。一般に、転送の開始はソフトウェアがコマンドに必要な最後の情報を提供するとすぐにトリガされます。もっと具体的に言えば、インダイレクトモードでは、以下のいずれかのイベントシーケンスが発生した場合に転送が開始されます。

- アドレスが不要（ADMODE[2:0] = 000）で、ソフトウェアからのデータ提供が不要（FMODE[1:0] = 01またはDMODE[2:0] = 000）な場合で、OCTOSPI_IRのINSTRUCTION[31:0]に書込みが実行された時点
- アドレスが必要（ADMODE[2:0] ≠ 000）で、ソフトウェアからのデータ提供が不要（FMODE[1:0] = 01またはDMODE[2:0] = 000）な場合で、OCTOSPI_ARのADDRESS[31:0]に書込みが実行された時点
- ソフトウェアからのデータ提供が必要（FMODE[1:0] = 00でDMODE[2:0] ≠ 000）な場合で、OCTOSPI_DRのDATA[31:0]に書込みが実行された時点

OCTOSPI_ABRへの書込みが通信開始をトリガすることはありません。オルタネートバイトが必要な場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSYビットが自動的にOCTOSPI_SRにセットされます。

HyperBusプロトコルでの転送開始のトリガ

OCTOSPIの設定に応じて、インダイレクトモードでコマンドを開始するさまざまな方法があります。一般に、ファームウェアが転送を始めるのに必要な最後の情報を提供するとすぐに転送は開始されます。もっと具体的に言えば、以下のレジスタ設定のいずれかが実行する最後のものであった場合、それによってインダイレクトモードでの通信が開始されます。

- インダイレクト読出しモード（FMODE[1:0] = 01）で、ADMODE[2:0] ≠ 000でADDRESS[31:0]（OCTOSPI_AR）への書込みを行ったとき。
- インダイレクト書込みモード（FMODE = 00の場合）で、DATA[31:0]（OCTOSPI_DR）への書込みを行ったとき。
- インダイレクト読出しモード（ADMODE[2:0] = 000かつFMODE = 01の状態）に対して、INSTRUCTION[31:0]（OCTOSPI_IR）への（ダミー）書込みを行ったとき。

転送が開始されたらすぐに、BUSYビット（OCTOSPI_SR[5]）が自動的にセットされます。

FIFOおよびデータの管理

インダイレクトモードでは、データはOCTOSPIに内蔵されている32バイトのFIFOを通ります。OCTOSPI_SRのFLEVELは、現在FIFOに保持されているバイト数を示します。

AHBバーストトランザクションがサポートされています。バーストのデータはOCTOSPI_DRに連続して書き込まれ、すぐに内蔵FIFOに転送されます。

インダイレクト書込みモード（FMODE[1:0] = 00）では、ソフトウェアはOCTOSPI_DRに書き込む時点でFIFOにデータを追加します。書込みによりFIFOに追加されるバイト数は、ワード書込みで4バイト、ハーフワード書込みで2バイト、バイト書込みで1バイトだけです。ソフトウェアがFIFOに追加するバイト数が多すぎる（DL[31:0]で示される数より多い）場合、余分なバイトは、書込み動作の終了時（TCFがセットされるとき）にFIFOから一掃されます。

OCTOSPI_DRへのバイト／ハーフワードアクセスは、32bitレジスタの最下位バイト／ハーフワードにのみ実行する必要があります。

FTHRESを使用してFIFO閾値を定義します。そのポイントに達するとFIFO閾値フラグFTFがセットされます。インダイレクト読出しモードでは、FIFOから読み出される有効バイト数が閾値を超えるとFTFがセットされます。また、FTHRESの設定にかかわらず、外部デバイスから最後のバイトが読み出された後にFIFO内にデータが残っている場合もFTFがセットされます。インダイレクト書込みモードでは、FIFO内の空のバイト数が閾値を超えるとFTFがセットされます。

FTIE = 1の場合、FTFがセットされた時点で割込みが発生します。DMAEN = 1の場合、FTFがセットされた時点でDMA転送が開始されます。閾値条件が真でなくなると（CPUまたはDMAによって十分なデータが転送された後）すぐに、ハードウェアによってFTFがクリアされます。

次のラインに対する要求がない限り、RX FIFOに最後に読み出されたデータは有効なままです。つまり、アプリケーションが同じ場所で連続して何回か読み出す場合、データはRX FIFOから提供され、離れているメモリから再度読み出されることはありません。

28.4.10 OCTOSPI自動ステータスポーリングモード

自動ステータスポーリングモードでは、OCTOSPIは定期的にコマンドを開始させて、定義された数のステータスバイト（最大4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割込みを生成することができます。自動ステータスポーリングモードはレギュラコマンドプロトコルでのみ使用する必要があります。HyperBusプロトコルの場合、HyperFlashメモリへの読出しステータスレジスタは2つのステップ（書込み操作とそれに続く読出し操作）で行う必要があるため、悪用することができません。

デバイスへのアクセスは、インダイレクト読出しモードと同じ方法で開始されます。この時点でOCTOSPI_SRのBUSYがハイになり、周期的なアクセスが実行される間もハイのままです。

OCTOSPI_PSMARのMASK[31:0]の内容は、自動ステータスポーリングモードで外部デバイスからのデータをマスクするために使用されます。

- MASK[n] = 0の場合、リザルトのビットnはマスクされ、考慮されません。
- MASK[n] = 1かつビット[n]の値がOCTOSPI_PSMARのMATCH[n]の値と同じであれば、ビットnでの一致が検出されます。

OCTOSPI_CRでPMM = 0の場合、AND一致モードがアクティブになります。つまり、SMFは、マスクされていないすべてのビットで一致が検出された場合にのみOCTOSPI_SRにセットされます。

OCTOSPI_CRでPMM = 1の場合、OR一致モードがアクティブになります。つまり、SMFは、マスクされていないいずれかのビットで一致が検出された場合にセットされます。

割込みは、SMIE = 1の場合にSMF = 1になるとき呼び出されます。

APMSがOCTOSPI_CRでセットされている場合、一致が検出されるとすぐに動作は停止し、BUSYが0になります。そうでない場合、BUSYは1のままで、アボートが検出されるまで、またはOCTOSPIが無効化される（EN = 0）まで周期的なアクセスが継続します。

OCTOSPI_DRには、最後に受信したステータスバイトが含まれます（FIFOは無効化されます）。レジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。OCTOSPI_SRのFTFは新たなステータスの読出しが完了するとすぐにセットされます。FTFは、データが読み出された直後にクリアされます。

自動ステータスポーリングモードでは、可変の遅延はサポートされていません。したがって、メモリは固定遅延に設定する必要があります。

28.4.11 OCTOSPIメモリマップドモード

メモリマップドモードに設定された場合、外部SPIデバイスは内部メモリとみなされます。

注： **外部デバイスの容量がもっと大きい場合でも、アドレス指定できるのは256MBまでです。**

DEVSIZE[4:0]で規定されている範囲外のアドレスにアクセスすると、256MBの範囲内であっても、AHBエラーが発生します。このエラーの影響は、アクセスを試みたAHBマスタに依存します。

- Cortex CPUの場合は、ハードフォールト割込みが発生します。
- DMAの場合は、DMA転送エラーが発生し、対応するDMAチャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。これにより、OCTOSPIは最も最近にアクセスしたアドレスに続くアドレスに継続してバイトをロードします。続くバイトに次のアクセスが連続している場合、それらの動作結果はプリフェッチされているので、その動作は迅速に完了します。

デフォルトでは、OCTOSPIはプリフェッチ動作を停止することはありません。外部デバイスへのアクセスが長時間なくても、NCSをローに保持したまま前の読出し動作をアクティブな状態に保つか、または新たな転送を再起動します。

NCSがローのときに外部デバイスはより多く電力を消費する傾向があるため、アプリケーションがタイムアウトカウンタの起動（OCTOSPI_CRにおいてTCEN = 1）を要求する場合があります。クロックが非アクティブになってからアクセスのない状態でxサイクルが経過すると、OCTOSPI_LPTRのTIMEOUT[15:0]で定義された期間の後、NCSが解除されます。

最初のメモリマップドアクセスが発生するとすぐに、BUSYはハイになります。プリフェッチ動作の影響で、BUSYの立下がりは、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、お勧めできません。インダイレクト書込みモードがこの動作を実現します。

28.4.12 OCTOSPIの設定の概要

OCTOSPIの設定は次の3つのステップで行われます。

1. OCTOSPIのシステム設定
2. OCTOSPIのデバイス設定
3. OCTOSPIのモード設定

28.4.13 OCTOSPIのシステム設定

OCTOSPIはOCTOSPI_CRを使用して設定します。ユーザは下記をプログラムする必要があります。

- FMODE[1:0]による機能モード
- 自動ステータスポーリングモードの動作（PMMおよびAPMSで必要な場合）
- FTHRESによるFIFOレベル
- DMAENによるDMAの使用
- TCENによるタイムアウトカウンタの使用
- デュアルメモリ構成（必要な場合、DMMによって）

割込みを使用する場合、それぞれのイネーブルビットもこのフェーズ中にセットすることができます。

タイムアウトカウンタを使用する場合、タイムアウト値をOCTOSPI_LPTRにプログラムします。

DMAチャネルはOCTOSPIの設定中に有効にしてはなりません。予期しないリクエストの生成を避けるため、動作を完全に設定した後にのみ有効にする必要があります。

DMAとOCTOSPIはデータ長に関して整合のとれた方法で設定しなければなりません。すなわち、FTHRES値はDMAバーストサイズを反映する必要があります。

28.4.14 OCTOSPIのデバイス設定

対象の外部デバイスに関連するパラメータはOCTOSPI_DCR1とOCTOSPI_DCR2によって設定されます。ユーザは下記をプログラムする必要があります。

- DEVSIZE[4:0]によるデバイスサイズ
- CSHT[5:0]によるチップセレクト最小ハイ時間
- FRCKとCKMODEによるクロックモード
- PRESCALER[7:0]によるデバイス周波数

MTYP[2:0]では、8ラインモードで使用されるメモリタイプを指定します。

- 8データビットモード（DMODE[2:0] = 100）でD0/D1順序のMicronモード
- 8データビットモード（DMODE[2:0] = 100）でD1/D0順序のMacronixモード。
  MTYP[2:0] = 001はOctaFlashメモリを対象とし、MTYP[2:0] = 011は特定のアドレスフェーズを持つOctaRAM™メモリを対象とします（アドレスはMacronix要件に合うように行と列で構築します）。
- HyperBusメモリモード：プロトコルはHyperBus仕様に準拠します。
- レジスタ空間をアドレス指定するHyperBusレジスタモード：このモードでのメモリマップドアクセスをキャッシュ不可能にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

DEVSIZE[4:0]は、次の式を使用して外部メモリのサイズを定義します。

$$\text{デバイス内のバイト数} = 2^{\text{DEVSIZE+1}}$$

ここで、DEVSIZE+1は外部デバイスのアドレスを指定するために必要なアドレスビット数です。外部デバイスの容量はインダイレクトモードでは最大4GB（32bitを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は256MBに制限されます。

DMM=1の場合、DEVSIZE[4:0]は2つのデバイスの総容量を示します。

OCTOSPIが2つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは2つのコマンドの間に1CLKサイクルの間だけハイレベルのチップセレクト信号（NCS）を立ち上げます。

外部デバイスがコマンド間により長い時間を必要とする場合、チップセレクトハイ時間を示すCSHT[5:0]を使用して、NCSがハイに保たれなければならない最小のCLKサイクル数を指定することができます。

CKMODEは、コマンド間でのCLKのレベルを示します（NCS=1のとき）。

HyperBusプロトコルでは、デバイスのタイミング（t_{ACC}とt_{RWR}）および遅延モードをOCTOSPI_HLCRに設定する必要があります。

28.4.15 OCTOSPIレギュラコマンドモードの設定

インダイレクトモードの設定

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0] = 01の場合、インダイレクト読出しモードが選択され、データを外部デバイスから読み出すことができます。

OCTOSPIがインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_DLRに読み出される／書き込まれるデータのバイト数を指定します。
2. OCTOSPI_TCRに、フレームのタイミングを指定します。
3. OCTOSPI_CCRに、フレームフォーマットを指定します。
4. OCTOSPI_IRに、命令を指定します。
5. OCTOSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
6. OCTOSPI_ARに、ターゲットアドレスを指定します。
7. 必要に応じてDMAチャネルを有効にします。
8. OCTOSPI_DRによりFIFOに対してデータの読出し／書込みを行います（DMAを使用しない場合）。

アドレスレジスタ（OCTOSPI_AR）とデータレジスタ（OCTOSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、OCTOSPI_IRが書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE[2:0]とDMODE[2:0]がどちらも000である場合、またはインダイレクト読出しモード（FMODE[1:0] = 01）でADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）で、データレジスタに書き込む必要がない場合（FMODE[1:0] = 01またはDMODE[2:0] = 000）、OCTOSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE[1:0] = 00かつDMODE[2:0] ≠ 000）、OCTOSPI_DRによるFIFOへの書込みによって通信の開始がトリガされます。

自動ステータスポーリングモードの設定

自動ステータスポーリングモードは、FMODE[1:0] = 10にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は4バイトです。OCTOSPI_DLRでもっと大量のデータが要求されても、そのリクエストは無視され、4バイトだけが読み出されます。OCTOSPI_PIRでは周期が指定されます。

ステータスデータが取得されると、以下を処理できます。

- SMFをセットする（有効になっていれば割込みを生成する）。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値はOCTOSPI_PSMKRに格納されている値でマスクし、OCTOSPI_PSMARに格納されている値で論理和または論理積をとることができます。

一致する場合、SMFがセットされ、有効になっていれば割込みが生成されます。AMPSがセットされている場合、OCTOSPIを自動的に停止することができます。いずれにしろ、一番最後に取得した値はOCTOSPI_DRで使用可能です。

OCTOSPIが自動ステータスポーリングモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_PSMKRに、入力マスクを指定します。
2. OCTOSPI_PSMARに、比較値を指定します。
3. OCTOSPI_PIRに、読出し周期を指定します。
4. OCTOSPI_DLRに、読み出されるデータのバイト数を指定します。
5. OCTOSPI_TCRに、フレームのタイミングを指定します。
6. OCTOSPI_CCRに、フレームフォーマットを指定します。
7. OCTOSPI_IRに、命令を指定します。
8. OCTOSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
9. OCTOSPI_ARに、オプションのターゲットアドレスを指定します。

アドレスレジスタ（OCTOSPI_AR）が特定のコマンドに関して更新する必要がない場合、OCTOSPI_CCRが書き込まれるとすぐにコマンドシーケンスが開始されます。これはADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）、OCTOSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

メモリマップドモードの設定

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、消去やプログラム·ステータスの内部フラグをポーリングする必要があるため、お勧めできません。インダイレクト書込みモードでは、場合によっては自動ステータスポーリングモードと組合せて、この動作を実現します。

メモリマップドモードに移行するには、OCTOSPI_CRにおいてFMODE[1:0] = 11にセットします。

プログラムされた命令およびフレームは、AHBマスタがメモリマップド空間にアクセスするときに送信されます。

FIFOは、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでのOCTOSPI_DRへのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（OCTOSPI_DLR）は意味を持ちません。

OCTOSPIがメモリマップドモードで使用される場合、フレームは次のように構成されます。

1. 読出し動作に対して、OCTOSPI_TCRにフレームのタイミングを指定します。
2. 読出し動作に対して、OCTOSPI_CCRにフレームフォーマットを指定します。
3. OCTOSPI_IRに、命令を指定します。
4. 読出し動作に対して、OCTOSPI_ABR にアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。
5. 書込み動作に対して、OCTOSPI_WTCRにフレームのタイミングを指定します。
6. 書込み動作に対して、OCTOSPI_WCCRにフレームフォーマットを指定します。
7. OCTOSPI_WIRに、命令を指定します。
8. 書込み動作に対して､OCTOSPI_WABRにアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。

すべての設定操作は、メモリ領域への最初のアクセスの前に完了する必要があります（BUSY = 0をチェックすることで保証されます）。BUSY = 1のときのレジスタ書込み操作は効果がなく、エラー応答は通知されません。最初のアクセスでOCTOSPIはビジーになり、それ以上の設定は許可されません。この場合、BUSYをローにする唯一の方法は、ENABLEビットをクリアするか、ABORTビットをセットして中止することです。

DQSを使用しない場合のOCTOSPI遅延データサンプリング

デフォルトでは、DQSを使用しない場合、外部デバイスに駆動されているデータをOCTOSPIがサンプリングするタイミングは、外部デバイスが信号を駆動してから1/2CLKサイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。OCTOSPI_TCRのSSHIFTを使用することで、データのサンプリングを1/2CLKサイクルだけシフトすることができます。

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ファームウェアはSSHIFTをクリアする必要があります。

DQSを使用する場合のOCTOSPI遅延データサンプリング

外部DQSをサンプリングクロックとして使用する場合、データ伝播遅延を補正するためにその時間をシフトすることができます。このシフトは、OCTOSPIの外部にある外部遅延ブロックによって行われます。この機能の制御は、デバイスの実装により異なります（詳細については製品リファレンスマニュアルを参照）。

遅延を補償する必要がない構成では、OCTOSPI_DCR1でDLYBYPをセットすることで外部遅延ブロックをバイパスできます。

1回限りの命令の送信（SIOO）

Flashメモリは、最初のコマンドシーケンスでのみ命令を送信し、その後のコマンドは直接アドレスから開始されるというモードを提供することができます。OCTOSPI_CCRのSIOOを使用して、このタイプの機能を活用することができます。

SIOOはメモリマップドモードでのみ有効です。このビットがセットされている場合､OCTOSPI_CCRへの書込み後の最初のコマンドに対してのみ命令が送信されます。

その後のコマンドシーケンスでは、OCTOSPI_CCRへの書込みがあるまで、命令フェーズがスキップされます。IMODE[1:0] = 00（命令なし）の場合、SIOOは効果がありません。

SIOOモードは、通信調整、NCS境界、またはリフレッシュのいずれかの機能が使用されている場合はサポートされません。

28.4.16 OCTOSPI HyperBusプロトコルの設定

インダイレクトモードの設定（HyperBus）

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0] = 01の場合、データを外部デバイスから読み出すことができるインダイレクト読出しモードが選択されます。ADMODEは000と異なる値（たとえば、ADMODE = 100など）で設定する必要があります。

OCTOSPIがインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_DLRに読み出される／書き込まれるデータのバイト数を指定します。
2. OCTOSPI_ARに、ターゲットアドレスを指定します。
3. 必要に応じてDMAチャネルを有効にします。
4. OCTOSPI_DRによりFIFOに対してデータの読出し／書込みを行います（DMAを使用しない場合）。

インダイレクト読出しモードでは、OCTOSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

インダイレクト書込みモードでは、OCTOSPI_DRによるFIFOへの書込みによって通信の開始がトリガされます。

メモリマップドモードの設定（HyperBus）

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、お勧めできません。インダイレクト書込みモードがこの動作を実現します。

メモリマップドモードに移行するには、FMODE[1:0] = 11にセットします。プログラムされた命令およびフレームは、AHBマスタがメモリマップド空間にアクセスするときに送信されます。

FIFOは、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでのOCTOSPI_DRへのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（OCTOSPI_DLR）は意味を持ちません。

メモリ領域へ最初にアクセスする前に、すべての設定操作を完了する必要があります。最初のアクセスでOCTOSPIはビジーになり、いかなる設定も許可されません。この場合、BUSYをローにする唯一の方法は、ENABLEビットをクリアするか、ABORTビットをセットして中止することです。

28.4.17 OCTOSPIエラー管理

次の場合に、エラーが生成される可能性があります。

- インダイレクトまたは自動ステータスポーリングモードでは、QUADSPI_ARに間違ったアドレスがプログラムされている場合（DEVSIZE[4:0]で定義されているデバイスのサイズに照らして）、TEFがセットされ、有効になっていれば割込みが生成されます。
- インダイレクトモードでは､アドレスにデータ長を加えた値がデバイスのサイズを超えている場合は、アクセスがトリガされるとすぐにTEFがセットされます。
- メモリマップドモードでは、AHB マスタによる範囲外のアクセスが行われた場合、誤った AHB リクエストへの応答としてAHBエラーが生成されます。
- メモリマップドモードが無効になっているときに、メモリマップド領域にアクセスすると、誤ったAHBリクエストへの応答としてAHBエラーが生成されます。

OCTOSPIは、次の状況でAHBスレーブエラーを発生します。

- メモリマップドモードが無効になっていて、AHB読出しリクエストが発生した場合。
- 読出しまたは書込みアドレスが、外部メモリのサイズを超過した場合。
- 読出しまたは書込みバーストの進行中に、アボートを受信した場合。
- 読出しまたは書込みバーストの進行中に、OCTOSPIが無効になった場合。
- 書込みラップバーストを受信した場合。
- オクタルDTRモードまたはデュアルメモリ構成のOCTOSPI_WCCRでDQSE = 0の間に書込みリクエストが受信された場合。
- MTYP[2:0]がHyperBusの場合を除き、DMODE[2:0] = 000（データフェーズなし）のときに書込みリクエストを受信した場合。
- ラップ読出しバースト時にアクセスサイズが不正である場合。これは、HSIZEが4バイトとは異なることを意味します（メモリマップドモードの場合のみ）。
- 4 バイトと異なるサイズの読出しラップバーストを受信した時のラップサイズが不正である場合（メモリマップドモードの場合のみ）。

28.4.18 OCTOSPIのBUSYおよびアボート

OCTOSPIが外部デバイスでの動作を開始すると、BUSYがOCTOSPI_SRに自動的にセットされます。

インダイレクトモードでは、OCTOSPIが要求されたコマンドシーケンスを完了し、FIFOが空になると、BUSYがリセットされます。

自動ステータスポーリングモードでは、一致（APMS = 1の場合）またはアボートの検出により、最後の周期的なアクセスが完了して初めてBUSYビットはローになります。

メモリマップドモードでの最初のアクセス後、アボートの検出でのみ、BUSYはローになります。

OCTOSPI_CRのABORTをセットすることによって、いかなる動作も中止することができます。アボートが完了すると、BUSYおよびABORTは自動的にリセットされ、FIFOが一掃されます。

ABORTを設定する前に、ソフトウェアは同期バリアを使用して現在のすべてのトランザクションが終了していることを確認する必要があります。OCTOSPI_DRでデータの読取りまたは書込み動作を処理するためにDMAが有効になっている場合は、OCTOSPIを中止する前にDMAチャネルを無効にすることをお勧めします。

注： **デバイスには、ステータスレジスタへの書込み動作が中止された場合に誤動作するものもあります。**

28.4.19 OCTOSPIの再設定または無効化

OCTOSPIの再設定の前に、ソフトウェアはすべてのトランザクションが完了していることを確認する必要があります。

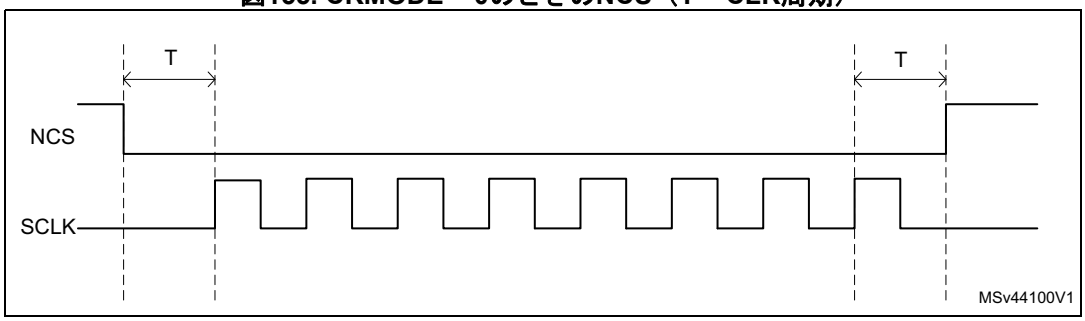
- メモリマップド書込みの後、ソフトウェアはダミーリードに続いて同期バリア、その後にABORTを実行する必要があります。
- メモリマップド読出しの後、ソフトウェアは同期バリア、その後にABORTを実行する必要があります。

28.4.20 NCSの動作

デフォルトではNCSハイで、外部デバイスは選択解除されています。NCSの立下がりは動作が始まる前に、立上がりは動作終了の直後に起こります。

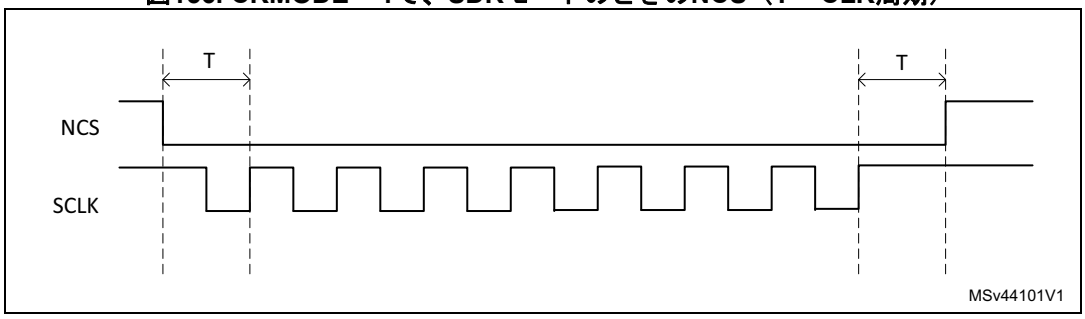
CKMODE = 0（クロックモード0 : 進行中の動作がない場合はCLKがローに保持される）のとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。

図155. CKMODE = 0のときのNCS（T = CLK周期）



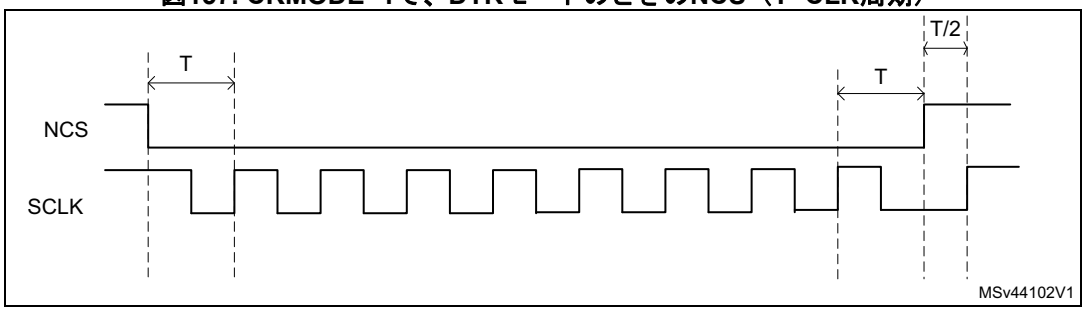
CKMODE = 1（クロックモード3 : 進行中の動作がない場合はCLKがハイになる）、かつSDRモードのとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。

図156. CKMODE = 1で、SDRモードのときのNCS（T = CLK周期）



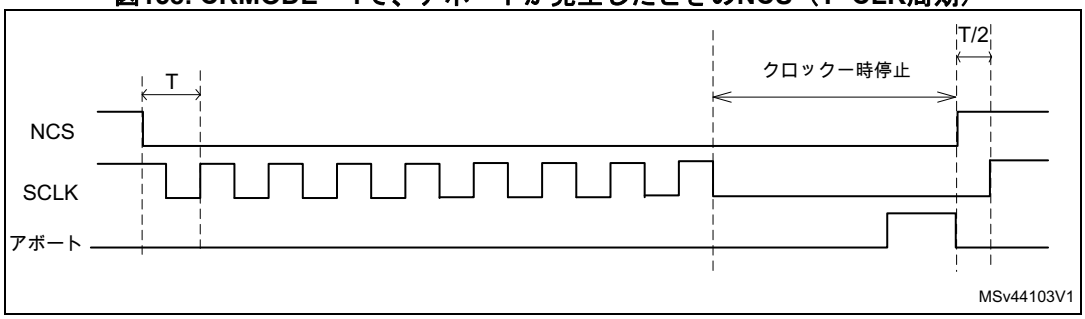
CKMODE = 1（クロックモード3）、かつDDTR = 1（データDTRモード）のとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のアクティブCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。DTR動作は立下がりエッジで終わらなければならないため、NCSが立ち上がるときCLKはローであり、その後CLKは1/2CLKサイクル後に立ちあがり直します。

図157. CKMODE=1で、DTRモードのときのNCS（T=CLK周期）



読出し動作中でFIFOがフルのまま、または書込み動作中でFIFOが空のままであれば、ソフトウェアがFIFOを処理するまでその動作は停止し、CLKはローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後にNCSが立ち上がり、CLKは1/2CLKサイクル後に立ち上がります（下図参照）。

図158. CKMODE = 1で、アボートが発生したときのNCS（T=CLK周期）



28.5 アドレスのアライメントとデータ数

以下の表は、使用例に応じたアドレスアラインメントとプログラムされたデータ数への影響をまとめたものです。

表253.アドレスアライメントのケース

メモリタイプ	トランザクションタイプ	アドレスの制約(1)	アドレスの制約不遵守の場合の影響	バイト数の制約 (1)	バイト数の制約不遵守の場合の影響
シングル、デュアル、クワッドFlashまたはSRAM（DMM = 0）	IND(2) 読出し	なし	なし	なし	なし
	MM(3) の読出し				
	IND書込み				
	MM書込み				
シングル、デュアル、クワッドFlashまたはSRAM（DMM = 1）	INDの読出し	偶数	ADDR[0] は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み	偶数	ADDR[0] は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MM書込み	偶数	スレーブエラー	偶数	最後のバイトが失われます。
SDRモードのオクタルFlash	INDの読出し	なし	なし	なし	なし
	MMの読出し				
	IND書込み				
	MM書込み				
DTRモードでWDMなしのオクタルメモリ(6)	INDの読出し	偶数	ADDR[0] は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み	偶数	ADDR[0] は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MM書込み	偶数	スレーブエラー	偶数	最後のバイトが失われます。

表253.アドレスアライメントのケース （続き）

メモリタイプ	トランザクションタイプ	アドレスの制約(1)	アドレスの制約不遵守の場合の影響	バイト数の制約 (1)	バイト数の制約不遵守の場合の影響
DTRモードでWDMありのオクタルFlashまたはRAM	INDの読出し	偶数	ADDR[0]は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				
HyperBus	INDの読出し	偶数	ADDR[0]は0にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				

1. ソフトウェアで遵守すること。
2. IND=インダイレクトモード
3. MM=メモリマップドモード
4. 転送開始時の追加データ
5. 転送終了時の追加データ
6. WDM = 書込みデータマスク

28.6 OCTOSPI割込み

割込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO閾値
- 転送完了
- 転送エラー

より高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表254.OCTOSPIの割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

28.7 OCTOSPIレジスタ

28.7.1 OCTOSPI制御レジスタ（OCTOSPI_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FMODE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
		rw	rw					rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FTHRES[4:0]					MSEL	DMM	Res.	Res.	TCEN	DMAEN	ABORT	EN
			rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:28 **FMODE[1:0]**：機能モード

このビットフィールドは、OCTOSPI機能の動作モードを定義します。

00：インダイレクト書込みモード

01：インダイレクト読出しモード

10：自動ステータスポーリングモード（レギュラコマンドプロトコルにのみ該当）

11：メモリマップドモード

すでにDMAEN = 1の場合、対応するチャネル用のDMAコントローラはFMODE[1:0]値を変更する前に無効にする必要があります。DMAEN=1のときにFMODE[1:0]とFTHRES[4:0]が誤って更新されると、DMAリクエスト信号は自動的に非アクティブ状態になります。

注： このビットフィールドは、BUSYが0のときにのみ修正できます。

ビット27:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **PMM**：ポーリング一致モード

このビットは、自動ステータスポーリングモードの時に「一致」を判断するために使用する必要のある方法を示します。

0：AND一致モード。SMFは、デバイスから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1：OR一致モード。SMFは、デバイスから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット22 **APMS**：自動ステータスポーリングモード停止

このビットは、一致後に自動ステータスポーリングモードを停止するかどうかを決定します。

0：自動ステータスポーリングモードは、アボートまたはOCTOSPIの無効化によってのみ停止されます。

1：自動ステータスポーリングモードは、一致が検出されるとすぐに停止されます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 **TOIE**：タイムアウト割込みイネーブル

このビットは、タイムアウト割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット19 **SMIE**：ステータス一致割込みイネーブル

このビットは、ステータス一致割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット18 **FTIE**：FIFO閾値割込みイネーブル

このビットは、FIFO閾値割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット17 **TCIE**：転送完了割込み有効化

このビットは、転送完了割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット16 **TEIE**：転送エラー割込みの有効化

このビットは、転送エラー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **FTHRES[4:0]**：FIFO閾値レベル

このビットフィールドでは、インダイレクトモードで、OCTOSPI_SRのFIFO閾値フラグを示すFTFのセットを引き起こすFIFO内のバイトの閾値数を定義します。

00000：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが1バイト以上ある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが1バイト以上ある場合にセットされます。

00001：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが2バイト以上ある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが2バイト以上ある場合にセットされます。

...

11111：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが32バイトある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが32バイトある場合にセットされます。

注： DMAEN=1の場合、対応するチャネル用のDMAコントローラはFTHRES[4:0]値を変更する前に無効にする必要があります。

ビット7 **MSEL**：外部メモリ選択

このビットは、シングルメモリ構成（DMM=0）においてシングル-、デュアル-、クワッド-SPIモードでアドレス指定する外部メモリを選択します。

0：外部メモリ1が選択されます（データはIO[3:0]を経由して交換される）。

1：外部メモリ2が選択されます（データはIO[7:4]を経由して交換される）。

このビットは、DMM = 1のとき、またはOctal-SPIモードが選択されたときは、無視されます。

ビット6 **DMM**：デュアルメモリ構成

このビットはデュアルメモリ構成をアクティブにします。このモードでは、2つの外部デバイスを同時に使用して、スループットおよび容量を2倍にします。

0：デュアルメモリ構成は無効です。

1：デュアルメモリ構成は有効です。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット5:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **TCEN**：タイムアウトカウンタ･イネーブル

このビットは、メモリマップドモードが選択されている（FMODE[1:0] = 11）場合にのみ有効です。このビットは、タイムアウトカウンタを有効にします。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（NCS）はメモリマップドモードにおける1回のアクセス後は無限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、メモリマップドモードで外部デバイスが非アクティブになってからTIMEOUT[15:0]サイクル後に解除されます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット2 **DMAEN**:DMAイネーブル

インダイレクトモードでは、DMAを使用して、OCTOSPI_DR経由でデータを入出力することができます。DMA転送は、FTFがセットされた時点で開始されます。

0：インダイレクトモードでDMAは無効です。

1：インダイレクトモードでDMAは有効です。

注： DMA 転送が進行中に DMAEN ビットをリセットすると、DMA とのハンドシェイクが切れます。DMAの動作中に、このビットを書き込まないでください。

ビット1 **ABORT**：アボートリクエスト

このビットは進行中のコマンドシーケンスを中止します。アボートが完了すると自動的にリセットされます。このビットは現在の転送を中止させます。

0：アボートは要求されていません。

1：アボートが要求されています。

注： このビットは常に0として読み出されます。

ビット0 **EN**：有効化

このビットは、OCTOSPIを有効にします。

0：OCTOSPIは無効です。

1：OCTOSPIは有効です。

注： 動作中にこのENビットがクリアされた場合、ACKを受信せずにDMAリクエストを中止できます。DMA転送中にこのビットが0にセットされた場合、DMAへのREQ信号は、DMAからのACK信号がアクティブになるのを待たずに非アクティブ状態に戻ります。

28.7.2 OCTOSPIデバイス設定レジスタ1（OCTOSPI_DCR1）

アドレスオフセット：0x0008

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	MTYP[2:0]			Res.	Res.	Res.	DEVSIZE[4:0]				
					rw	rw	rw				rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CSHT[5:0]						Res.	Res.	Res.	Res.	DLY BYP	Res.	FRCK	CKMODE
		rw	rw	rw	rw	rw	rw					rw		rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **MTYP[2:0]**：メモリタイプ

このビットフィールドは、サポートするメモリのタイプを示します。

000：DTR8データビットモードでD0/D1順序のMicronモード。single-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。

注： このモードでは、DQS 信号の極性がメモリクロック信号に対して反転されます。これはデフォルト値であり、Micron以外のメモリのMTYP[2:0]を変更する際は注意する必要があります。

001：DTR8データビットモードでD1/D0順序のMacronixモード。single-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。

010：標準モード

011：DTR8データビットモードでD1/D0順序のMacronix RAMモード。single-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。

100：HyperBusメモリモード。プロトコルはHyperBus仕様に準拠します。

101：HyperBusレジスタモード。レジスタ空間にアドレス指定。このモードでメモリマップドアクセスを行う場合、キャッシュ不可能にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

その他：予約済み

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **DEVSIZE[4:0]**：デバイスサイズ

このビットフィールドは、次の式を使用して外部デバイスのサイズを定義します。

デバイス内のバイト数 = $2^{[DEVSIZE+1]}$。

DEVSIZE+1は外部デバイスのアドレスを指定するために必要とされる有効なアドレスビット数です。デバイスの容量はインダイレクトモードでは最大4Gバイト（32bitを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は256Mバイトに制限されます。

レギュラコマンドプロトコルでは、DMM = 1の場合、DEVSIZE[4:0]は2つの外部デバイスのうちの2つの容量を示します。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:8 **CSHT[5:0]**：チップセレクトハイ時間

CSHT+1は、外部デバイスに対して発行されるコマンド間でチップセレクト（NCS）がハイに保たれなければならないCLKサイクルの最小数を定義します。

0x0：NCSは、外部デバイスコマンド間で少なくとも1サイクルの間はハイに保持されます。

0x1：NCSは、外部デバイスコマンド間で少なくとも2サイクルの間はハイに保持されます。

...

0x3F：NCSは、外部デバイスコマンド間で少なくとも64サイクルの間はハイに保持されます。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **DLYBYP**：遅延ブロックバイパス

0：内部サンプリングクロック（フィードバッククロック）またはDQS外部データストローブ信号は遅延ブロックによって遅延されます（このブロックはOCTOSPIペリフェラルの一部ではないため、詳細については、リファレンスマニュアルの該当するセクションを参照してください）。

1：遅延ブロックはバイパスされるので、内部サンプリングクロックまたはDQS外部データストローブ信号は遅延ブロックに影響されません。遅延は、遅延値を遅延ブロックで最小値に設定しても、遅延ブロックをバイパスしなかった場合より短くなります。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **FRCK**：フリー･ランニング･クロック

このビットによってフリー･ランニング･クロックを設定します。

0：CLKはフリーランニングではありません。

1：CLKはフリーランニングです（常に提供される）。

注： フリー･ランニング･クロック･モードは遅延の較正のみを目的としています。FRCKがセットされているとき、メモリやその他のデバイスへのアクセスはできません。

ビット0 **CKMODE**：クロックモード0/モード3

このビットは、コマンド間（NCS = 1のとき）にCLKが取るレベルを示します。

0：CLKはNCSがハイである（チップセレクトが解除されている）間はローに保持される必要があります。これは、クロックモード0と呼ばれます。

1：CLKはNCSがハイである（チップセレクトが解除されている）間はハイに保持される必要があります。これは、クロックモード3と呼ばれます。

28.7.3 OCTOSPIデバイス設定レジスタ2（OCTOSPI_DCR2）

アドレスオフセット：0x000C

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **WRAPSIZE[2:0]**：ラップサイズ

このビットフィールドは、メモリが設定される際のラップサイズを示します。ラップされた命令用の個別のコマンドを備えるメモリに対して、このビットフィールドは、OCTOSPI_WPIRレジスタに保持されたコマンドに関連付けられたラップサイズを示します。

000：ラップド読出しはこのメモリでサポートされていません。

010：外部メモリは16バイトのラップサイズをサポートしています。

011：外部メモリは32バイトのラップサイズをサポートしています。

100：外部メモリは64バイトのラップサイズをサポートしています。

101：外部メモリは128バイトのラップサイズをサポートしています。

その他：予約済み

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PRESCALER[7:0]**：クロックプリスケーラ

このビットフィールドは、カーネルクロックに基づいてCLKを生成するスケーラ係数（値 + 1）を定義します。

0：$F_{CLK}=F_{KERNEL}$、カーネルクロックが直接OCTOSPI CLK（プリスケーラはバイパス）として使用されます。この場合、DTRモードを使用する場合は、デューティサイクル50%のカーネルクロックをOCTOSPIIに提供することが必須です。

1：$F_{CLK} = F_{KERNEL}/2$

2：$F_{CLK} = F_{KERNEL}/3$

...

255：$F_{CLK} = F_{KERNEL}/256$

クロック分周比が奇数の場合、CLKのデューティサイクルは50%ではありません。クロック信号は、ハイに保持された時間より1サイクル分長く、ローに保持されます。

28.7.4 OCTOSPIデバイス設定レジスタ3（OCTOSPI_DCR3）

アドレスオフセット：0x0010

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBOUND[4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MAXTRAN[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **CSBOUND[4:0]**：NCS境界

このビットフィールドは、トランザクション境界機能を有効にします。有効な場合、最小値の3が推奨されます。$2^{CSBOUND}$バイトの各境界でNCSが解除されます。

0：NCS境界は無効です。

その他：NCS境界は、$2^{CSBOUND}$バイトに設定されます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **MAXTRAN[7:0]**：最大転送

このビットフィールドは、通信調整機能を有効にします。他のOCTOSPIがバスへのアクセスをリクエストするとき、NCSはMAXTRAN+1クロックサイクルごとにリリースされます。

0：最大通信は無効です。

その他：最大通信はMAXTRAN+1バイトに設定されます。

28.7.5 OCTOSPIデバイス設定レジスタ4（OCTOSPI_DCR4）

アドレスオフセット：0x0014

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REFRESH[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REFRESH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REFRESH[31:0]**：リフレッシュレート

このビットフィールドは、リフレッシュレート機能を有効にします。NCSは、書込みのREFRESH + 1クロックサイクルごと、および読出しのREFRESH + 4クロックサイクルごとに解除されます。Single-、Dual-、またはQuad-SPIモードで1つのバイト送信中にリフレッシュが発生した場合、そのバイト送信を完了させる必要があるため、これらの2つの値は数クロックサイクル拡張される可能性があります。

0：リフレッシュは無効です。

その他：最大通信長はREFRESH+1クロックサイクルに設定されます。

注： REFRESHカウントは分周されたクロック周期に基づいています。したがって、OCTOSPI_DCR2 PRESCALERビットフィールドが変更された場合、それに応じてREFRESHフィールドを更新する必要があります。

28.7.6 OCTOSPIステータスレジスタ（OCTOSPI_SR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FLEVEL[5:0]						Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
		r	r	r	r	r	r			r	r	r	r	r	r

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:8 **FLEVEL[5:0]**：FIFOレベル

このビットフィールドはFIFO内に保持されている有効バイト数を示します。FIFOが空の場合、FLEVEL = 0で、フルになっている場合、32です。

自動ステータスポーリングモードでは、FLEVELはゼロです。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。これは、外部デバイスの動作が終わり、FIFOが空になると自動的にクリアされます。

ビット4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOFに1を書き込むことによってクリアされます。

ビット3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（OCTOSPI_PSMAR）の対応するビットと一致している場合に、自動ステータスポーリングモードでセットされます。
CSMFに1を書き込むことによってクリアされます。

ビット2 **FTF**：FIFO閾値フラグ

インダイレクトモードでは、このビットはFIFO閾値に達した時点で、または外部デバイスからの読出し完了後にFIFO内にデータが残っている場合にセットされます。
閾値条件が真でなくなるとすぐに自動的にクリアされます。
自動ステータスポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット1 **TCF**：転送完了フラグ

このビットは、プログラムされたデータ数が転送された時点でインダイレクトモードでセットされるか、または転送が中止された時点でいずれかのモードでセットされます。また、CTCFに1を書き込むことによってクリアされます。

ビット0 **TEF**：転送エラーフラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときに、インダイレクトモードでセットされます。
CTEFに1を書き込むことによってクリアされます。

28.7.7 OCTOSPIフラグクリアレジスタ（OCTOSPI_FCR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
											w	w		w	w

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CTOF**：タイムアウトフラグのクリア

このビットに1を書き込むと、OCTOSPI_SRレジスタのTOFフラグがクリアされます。

ビット3 **CSMF**：ステータス一致フラグのクリア

このビットに1を書き込むと、OCTOSPI_SRレジスタのSMFフラグがクリアされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CTCF**：転送完了フラグのクリア

このビットに1を書き込むと、OCTOSPI_SRレジスタのTCFフラグがクリアされます。

ビット0 **CTEF**：転送エラーフラグのクリア

このビットに1を書き込むと、OCTOSPI_SRレジスタのTEFフラグがクリアされます。

28.7.8 OCTOSPIデータ長レジスタ（OCTOSPI_DLR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DL[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DL[31:0]**：データ長

インダイレクトモードおよび自動ステータスポーリングモードで取得されるデータ数（値+1）自動ステータスポーリングモードの場合、3より大きい値（4バイトを示す）を使用する必要があります。

インダイレクトモードでのすべて1は、不定長を意味します。この場合、DEVSIZEによって定義されているメモリの最後までOCTOSPIが継続します。

0x0000_0000：1バイトが転送されます。

0x0000_0001：2バイトが転送されます。

0x0000_0002：3バイトが転送されます。

0x0000_0003：4バイトが転送されます。

...

0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。

0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。

0xFFFF_FFFF：不定長--すべてのバイトが外部デバイスの終了（DEVSIZEによって定義）まで転送されます。DEVSIZE = 0x1Fの場合、読出しが無期限に続けられます。

デュアルメモリ構成（DMM = 1）では、DL[0]は、たとえこのビットに0が書き込まれても1のままとなります。したがって、アクセスのたびに必ず偶数バイトが転送されます。

このビットフィールドはメモリマップドモードでは効果がありません。

28.7.9 OCTOSPIアドレスレジスタ（OCTOSPI_AR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

このレジスタは、BUSY= 0、かつFMODE ≠ 11のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDRESS[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRESS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ADDRESS[31:0]**：アドレス

外部デバイスに送信されるアドレスHyperBusプロトコルでは、16bitワード指向なので、このフィールドは偶数でなければなりません。デュアルメモリ構成では、AR[0]は0に強制されます。

28.7.10 OCTOSPIデータレジスタ（OCTOSPI_DR）

アドレスオフセット：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATA[31:0]**：データ

外部SPIデバイスに対して送受信されるデータ

インダイレクト書込みモードでは、このレジスタに書き込まれたデータはいったんFIFOに格納されてから、データフェーズ時に外部デバイスに送信されます。FIFOが満杯の場合、書き込まれるデータ量を受け入れるだけの十分な空間がFIFOにできるまで、書込み動作は停止されます。

インダイレクト読出しモードでは、このレジスタを読み出すことで（FIFO経由）、外部デバイスから受信したデータを取得できます。FIFO内のバイト数が読出し動作で要求したバイト数ほど多くない場合で、BUSY =1のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読出し動作は停止されます。

自動ステータスポーリングモードでは、このレジスタは外部デバイスから読み出された最後のデータを格納しています（マスキングなし）。

このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書込みモードでの書込みによりFIFOに追加されるバイト数は、バイト書込みで1バイト、ハーフワード書込みで2バイト、ワード書込みで4バイトです。

同様に、インダイレクト読出しモードでの読出しによりFIFOから削除されるバイト数は、バイト読出しで1バイト、ハーフワード読出しで2バイト、ワード読出しで4バイトです。インダイレクトモードでのアクセスは、このレジスタのボトムに整列されなければなりません。つまり、バイト読出しではDATA[7:0]が、ハーフワード読出しではDATA[15:0]が読み出されなければなりません。

28.7.11 OCTOSPIポーリングステータスマスクレジスタ（OCTOSPI_PSMKR）

アドレスオフセット：0x0080

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MASK[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MASK[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MASK[31:0]**：ステータスマスク

自動ステータスポーリングモードで受信したステータスバイトに適用されるマスク
ビットnの場合
0：自動ステータスポーリングモードで受信したデータのビットnはマスクされ、その値は一致ロジックでは考慮されません。
1：自動ステータスポーリングモードで受信したデータのビットnはマスクされず、その値は一致ロジックで考慮されます。

28.7.12 OCTOSPIポーリングステータス一致レジスタ（OCTOSPI_PSMAR）

アドレスオフセット：0x0088

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MATCH[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MATCH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MATCH[31:0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

28.7.13 OCTOSPIポーリングインターバルレジスタ（OCTOSPI_PIR）

アドレスオフセット：0x0090

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTERVAL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INTERVAL[15:0]**：ポーリングインターバル

自動ステータスポーリングフェーズ時の読出し間のCLKサイクル数

28.7.14 OCTOSPI通信設定レジスタ（OCTOSPI_CCR）

アドレスオフセット：0x0100

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SIOO	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
rw		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		AD DTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31 **SIOO**：1回限りの命令の送信モード

このビットは、IMODE = 00のときには効果がありません（1回限りの命令の送信（SIOO）を参照）。

0：トランザクションごとに命令を送信します。

1：最初のコマンドに対してのみ命令を送信します。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル

このビットは、データストローブ管理を有効にします。

0：DQSは無効です。

1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート

このビットは、データフェーズのDTRモードをセットします。

0：データフェーズのDTRモードは無効です。

1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード
このビットフィールドは動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは1本のライン上にあります。
010：データは2本のライン上にあります。
011：データは4本のライン上にあります。
100：データは8本のライン上にあります。
その他：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ
このビットフィールドはオルタネートバイトのサイズを定義します。
00：8bitオルタネートバイト
01：16bitオルタネートバイト
10：24bitオルタネートバイト
11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
100：オルタネートバイトは8本のライン上にあります。
その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ

このビットフィールドは命令のサイズを定義します。

00：8bit命令

01：16bit命令

10：24bit命令

11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート

このビットは、命令フェーズのDTRモードをセットします。

0：命令フェーズのDTRモードは無効です。

1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード

このビットフィールドは動作の命令フェーズのモードを定義します。

000：命令はありません。

001：命令は1本のライン上にあります。

010：命令は2本のライン上にあります。

011：命令は4本のライン上にあります。

100：命令は8本のライン上にあります。

その他：予約済み

28.7.15 OCTOSPIタイミング設定レジスタ（OCTOSPI_TCR）

アドレスオフセット：0x0108

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SSHIFT	Res.	DHQC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw		rw												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **SSHIFT**：サンプリングシフト

デフォルトでは、OCTOSPIは外部デバイスによるデータ駆動の1/2CLKサイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2サイクルのシフト

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ソフトウェアはSSHIFT = 0であることを確認する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **DHQC**：遅延ホールドクォータサイクル

0：遅延ホールドなし

1：1/4サイクルのホールド

ビット27:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。
SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。
DQSをアクティブにしてメモリを使用する場合は、少なくとも6ダミーサイクルを置くことを推奨します。

28.7.16 OCTOSPI命令レジスタ（OCTOSPI_IR）

アドレスオフセット：0x0110

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令

外部SPIデバイスに送信される命令

28.7.17 OCTOSPIオルタネートバイトレジスタ（OCTOSPI_ABR）

アドレスオフセット：0x0120

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト

アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

28.7.18 OCTOSPI低電力タイムアウトレジスタ（OCTOSPI_LPTR）

アドレスオフセット：0x00130

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEOUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **TIMEOUT[15:0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、OCTOSPIはその後のバイトをプリフェッチし、それらをFIFOに保持します。

このビットフィールドは、クロックが非アクティブになってからNCSを立ち上げるまでにOCTOSPIが待つCLKサイクルの数を示し、外部デバイスを消費電力がより少ない状態にします。

28.7.19 OCTOSPIラップ通信設定レジスタ（OCTOSPI_WPCCR）

アドレスオフセット：0x0140

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		AD DTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル

このビットは、データストローブ管理を有効にします。

0：DQSは無効です。

1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート

このビットは、データフェーズのDTRモードをセットします。

0：データフェーズのDTRモードは無効です。

1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード
このビットフィールドは動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは1本のライン上にあります。
010：データは2本のライン上にあります。
011：データは4本のライン上にあります。
100：データは8本のライン上にあります。
その他：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ
このビットフィールドはオルタネートバイトのサイズを定義します。
00：8bitオルタネートバイト
01：16bitオルタネートバイト
10：24bitオルタネートバイト
11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトなし
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
100：オルタネートバイトは8本のライン上にあります。
その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ

このビットフィールドは命令のサイズを定義します。

00：8bit命令

01：16bit命令

10：24bit命令

11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート

このビットは、命令フェーズのDTRモードをセットします。

0：命令フェーズのDTRモードは無効です。

1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード

このビットフィールドは動作の命令フェーズのモードを定義します。

000：命令はありません。

001：命令は1本のライン上にあります。

010：命令は2本のライン上にあります。

011：命令は4本のライン上にあります。

100：命令は8本のライン上にあります。

その他：予約済み

28.7.20 OCTOSPIラップタイミング設定レジスタ（OCTOSPI_WPTCR）

アドレスオフセット：0x0148

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SSHIFT	Res.	DHQC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw		rw												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **SSHIFT**：サンプリングシフト

デフォルトでは、OCTOSPIは外部デバイスによるデータ駆動の1/2CLKサイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2サイクルのシフト

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ファームウェアはSSHIFT = 0であることを確実にする必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **DHQC**：遅延ホールドクォータサイクル

ホールド要件を満たすために、DTR通信の出力に1/4サイクルの遅延を追加します。

0：1/4サイクル遅延なし

1：1/4サイクルの遅延を挿入

ビット27:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。DQSをアクティブにしてメモリを使用する場合は、少なくとも5ダミーサイクルを置くことを推奨します。

28.7.21 OCTOSPIラップ命令レジスタ（OCTOSPI_WPIR）

アドレスオフセット：0x0150

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令

外部SPIデバイスに送信される命令

28.7.22 OCTOSPIラップオルタネートバイトレジスタ（OCTOSPI_WPABR）

アドレスオフセット：0x0160

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト

アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

28.7.23 OCTOSPI書込み通信設定レジスタ（OCTOSPI_WCCR）

アドレスオフセット：0x0180

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル
このビットは、データストローブ管理を有効にします。
0：DQSは無効です。
1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート
このビットは、データフェーズのDTRモードをセットします。
0：データフェーズのDTRモードは無効です。
1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード
このビットフィールドは動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは1本のライン上にあります。
010：データは2本のライン上にあります。
011：データは4本のライン上にあります。
100：データは8本のライン上にあります。
その他：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ
このビットフィールドはオルタネートバイトのサイズを定義します。
00：8bitオルタネートバイト
01：16bitオルタネートバイト
10：24bitオルタネートバイト
11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
100：オルタネートバイトは8本のライン上にあります。
その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ
このビットフィールドは命令のサイズを定義します。
00：8bit命令
01：16bit命令
10：24bit命令
11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート
このビットは、命令フェーズのDTRモードをセットします。
0：命令フェーズのDTRモードは無効です。
1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード
このビットフィールドは動作の命令フェーズのモードを定義します。
000：命令はありません。
001：命令は1本のライン上にあります。
010：命令は2本のライン上にあります。
011：命令は4本のライン上にあります。
100：命令は8本のライン上にあります。
その他：予約済み

28.7.24 OCTOSPI書込みタイミング設定レジスタ（OCTOSPI_WTCR）

アドレスオフセット：0x0188

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。
SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。DQSをアクティブにしてメモリを使用する場合は、少なくとも5ダミーサイクルを置くことを推奨します。

28.7.25 OCTOSPI書込み命令レジスタ（OCTOSPI_WIR）

アドレスオフセット：0x0190

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令

外部SPIデバイスに送信される命令

28.7.26 OCTOSPI書込みオルタネートバイトレジスタ（OCTOSPI_WABR）

アドレスオフセット：0x01A0

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト

アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

28.7.27 OCTOSPI HyperBus遅延設定レジスタ（OCTOSPI_HLCR）

アドレスオフセット：0x0200

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **TRWR[7:0]**：読出し･書込み最小リカバリ時間

通信クロックサイクル数で表されるデバイスの読出しから書込み／書込みから読出しへの最小リカバリ時間

ビット15:8 **TACC[7:0]**：アクセス時間

通信クロックサイクル数で表される、メモリ待ち時間に応じたデバイスアクセス時間

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **WZL**：書込みゼロ遅延

このビットは、書込み動作でのゼロ遅延を有効にします。

0：書込みアクセスでの遅延あり

1：書込みアクセスでの遅延なし

ビット0 **LM:**：遅延モード

このビットは、遅延モードを選択します。

0：可変の初期遅延

1：固定遅延

28.7.28 OCTOSPIレジスタマップ

表255.OCTOSPIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	OCTOSPI_CR	Res.	Res.	FMDOE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	Res.	FTHRES[4:0]					MSEL	DMM	Res.	Res.	TCEN	DMAEN	ABORT	EN
	リセット値			0	0					0	0		0	0	0	0	0				0	0	0	0	0	0	0			0	0	0	0
0x0004	予約済み	予約済み																															
0x0008	OCTOSPI_DCR1	Res.	Res.	Res.	Res.	Res.	MTYP[2:0]			Res.	Res.	Res.	DEVSIZE[4:0]					Res.	Res.	CSHT[5:0]						Res.	Res.	Res.	Res.	DLYBYP	Res.	FRCK	CKMODE
	リセット値						0	0	0				0	0	0	0	0			0	0	0	0	0	0					0		0	0
0x000C	OCTOSPI_DCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
	リセット値														0	0	0									0	0	0	0	0	0	0	0
0x0010	OCTOSPI_DCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBOUND[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MAXTRAN[7:0]							
	リセット値												0	0	0	0	0									0	0	0	0	0	0	0	0
0x0014	OCTOSPI_DCR4	REFRESH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018 - 0x001C	予約済み	予約済み																															
0x0020	OCTOSPI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[5:0]						Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																			0	0	0	0	0	0			0	0	0	0	0	0
0x0024	OCTOSPI_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x0028- 0x003C	予約済み	予約済み																															
0x0040	OCTOSPI_DLR	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	予約済み	予約済み																															
0x0048	OCTOSPI_AR	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	予約済み	予約済み																															
0x0050	OCTOSPI_DR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054 - 0x007C	予約済み	予約済み																															
0x0080	OCTOSPI_ PSMKR	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	予約済み	予約済み																															

表255.OCTOSPIレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0088	OCTOSPI_PSMAR	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008C	予約済み	予約済み																															
0x0090	OCTOSPI_PIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094 - 0x00FC	予約済み	予約済み																															
0x0100	OCTOSPI_CCR	SIOO	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x0104	予約済み	予約済み																															
0x0108	OCTOSPI_TCR	Res.	SSHIFT	Res.	DHQC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値		0		0																								0	0	0	0	0
0x010C	予約済み	予約済み																															
0x0110	OCTOSPI_IR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114 ~0x011C	予約済み	予約済み																															
0x0120	OCTOSPI_ABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0124 - 0x012C	予約済み	予約済み																															
0x0130	OCTOSPI_LPTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134 - 0x013C	予約済み	予約済み																															
0x0140	OCTOSPI_WPCCR	Res.	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE [1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x0144	予約済み	予約済み																															
0x0148	OCTOSPI_WPTCR	Res.	SSHIFT	Res.	DHQC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値		0		0																								0	0	0	0	0
0x014C	予約済み	予約済み																															
0x0150	OCTOSPI_WPIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0154 - 0x015C	予約済み	予約済み																															

表255.OCTOSPIレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0160	OCTOSPI_WPABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0164 - 0x017C	予約済み	予約済み																															
0x0180	OCTOSPI_WCCR	Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]			Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x0184	予約済み	予約済み																															
0x0188	OCTOSPI_WTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値																												0	0	0	0	0
0x018C	予約済み	予約済み																															
0x0190	OCTOSPI_WIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0194 - 0x019C	予約済み	予約済み																															
0x01A0	OCTOSPI_WABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01A4 - 0x01FC	予約済み	予約済み																															
0x0200	OCTOSPI_HLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]								TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

29 OCTOSPIのI/Oマネージャ（OCTOSPIM）

このセクションは、STM32U535/545デバイスには適用されません。

29.1 概要

OCTOSPI I/Oマネージャは、（オルタネート機能配置の前に）完全I/Oマトリックスを用いた効率的なOCTOSPIピンの割当てを可能にし、同じバス上でシングル／デュアル／クアッド／オクタルのSPIインタフェースの多重化を可能にする低レベルインタフェースです。

29.2 OCTOSPIMの主な機能

- シングル／デュアル／クワッド／オクタルSPIインタフェースを最大2個サポート
- ピン割当てのポートを最大2個サポート
- 機能（データ／制御／クロック）によるピン割当て用の完全にプログラム可能なI/Oマトリックス

29.3 OCTOSPIMの実装

次の表に、OCTOSPIMの実装の説明を示します。

表256.OCTOSPIMの実装

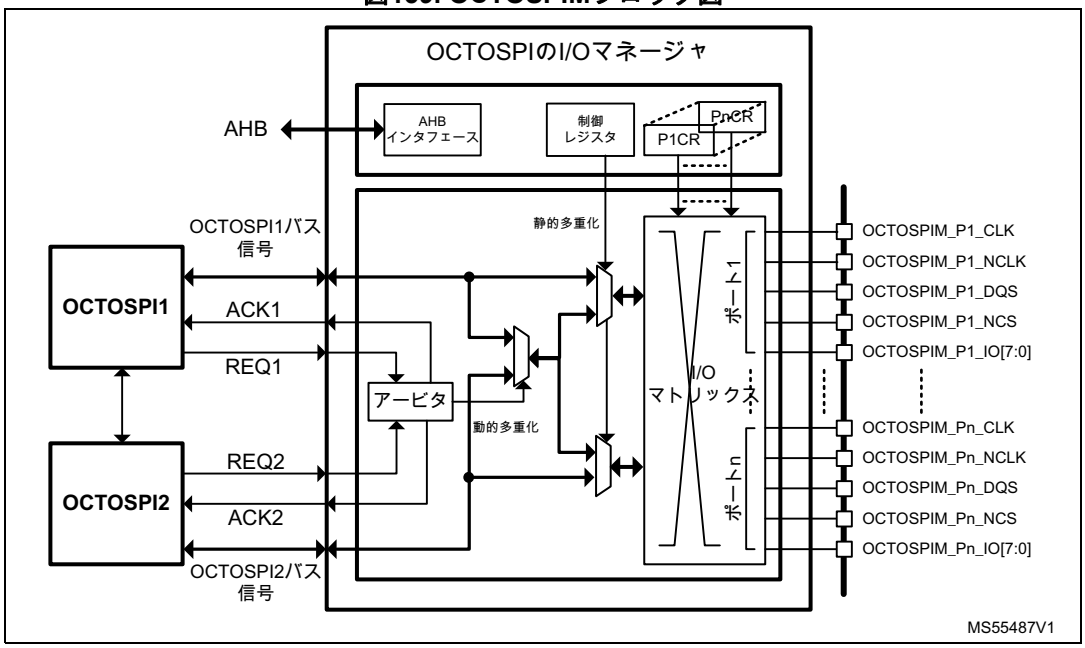
OCTOSPIの機能	デバイスで使用可能
シングル／デュアル／クワッド･インタフェースを最大2個サポート	X
完全I/O多重化機能	X
時分割マルチプレクス･モードのサポート	X
ハイスピードインタフェースのサポート	-
OCTOSPIがデュアル･チップセレクトを提供する場合のチップセレクトの選択	-
16bitデータインタフェースとデュアル･オクタルモードのサポート	-

29.4 OCTOSPIMの機能説明

29.4.1 OCTOSPIMブロック図

OCTOSPIのI/Oマネージャのブロック図を図159に示します。

図159. OCTOSPIMブロック図



1. ポート数（n）は2個です。
2. アービトレーションは、両方のI/Oマトリックス入力ポートで可能です。

29.4.2 OCTOSPIMの入出力ピン

表257.OCTOSPIMの入出力ピン

ピン名(1)	信号タイプ	説明
OCTOSPIM_Px_NCLK	出力	1.8V HyperBusプロトコルに対応したOCTOSPI反転クロック
OCTOSPIM_Px_CLK		OCTOSPIクロック
OCTOSPIM_Px_IOn（N = 0～7）	入力／出力	OCTOSPIデータピン
OCTOSPIM_Px_NCS	出力	メモリのチップセレクト
OCTOSPIM_Px_DQS	入力／出力	メモリとの間のデータストローブ／書込みマスク信号

1. x = 1～2

29.4.3 OCTOSPIMマトリックス

OCTOSPIのI/Oマネージャ･マトリックスにより、ユーザは完全にプログラム可能な機能の事前配置を設定できます。

- 任意のOCTOSPIM_Pn_CLK/OCTOSPIM_Pn_NCLKペアを個別にOCTOSPI1_CLK/OCTOSPI1_NCLKまたはOCTOSPI2_CLK/OCTOSPI2_NCLKに配置できます。
- 任意のOCTOSPIM_Pn_DQSを個別にOCTOSPI1_DQSまたはOCTOSPI2_DQSに配置できます。
- 任意のOCTOSPIM_Pn_NCSを個別にOCTOSPI1_NCSまたはOCTOSPI2_NCSに配置できます。
- 任意の OCTOSPIM_Pn_IO[3:0] および OCTOSPIM_Pn_IO[7:4] を個別に OCTOSPI1_IO[3:0]、OCTOSPI1_IO[7:4]、OCTOSPI2_IO1[3:0]、またはOCTOSPI2_IO[7:4]に配置できます。

OCTOSPIのI/Oマネージャポートごとに、個別の信号が有効化され、対応するOCTOSPIのI/Oマネージャのポートn設定レジスタ（OCTOSPIM_PnCR）により配置が設定されます。

いくつかのI/Oピンの設定が同じで、同時に有効化された場合、結果は予測できない可能性があります。

デフォルトのリセット解除設定では、OCTOSPI1とOCTOSPI2の信号は、それぞれポート1とポート2に配置されます。

OCTOSPIMの設定を変更できるのは、OCTOSPIがすべて無効なときだけです。

29.4.4 OCTOSPIMマルチプレクス･モード

このモードが設定されると、OCTOSPIは同じバス上で時分割多重化されます。両OCTOSPIはREQ/ACK信号によりリクエスト／確認応答プロトコルを通じてバスの所有権を（順番に）取得します。

時分割多重化は、設定レジスタOCTOSPIM_CRのMUXENビットをセットすることで有効になります。

各OCTOSPIの公平性カウンタ（MAXTRAN）は、特定のOCTOSPIがバスを取得する最大時間を正確に管理するために使用できます。この機能は、他方のOCTOSPIの最大バスアクセス遅延を保証します。バスが1つのOCTOSPIによって開放されると、ラウンドロビン方式のアービトレーション･フェーズが発生します。つまり、もう1つのOCTOSPIは、バスをリクエストすると取得できます。

マルチプレクス･モードが有効になっている場合、両方のOCTOSPIインタフェースの公平性カウンタまたはリフレッシュ･タイムアウト･カウンタのいずれかがアクティブになっている必要があります。

OCTOSPIn_NCS は多重化の対象ではありません。OCTOSPIn_IOs、OCTOSPIn_DQS、およびOCTOSPIn_CLK/OCTOSPIn_NCLKのみが多重化されます。

マルチプレクス･モードが使用されると、OCTOSPIではクロックモード0のみがサポートされます。

アービトレーションとバスの共有により、OCTOSPIの自動ポーリングインターバル時間は、使用時に増加する場合があります。

最短切り替え時間

OCTOSPIから別のOCTOSPIに切り替える際に必要な最小サイクル数を設定できます。

この内部タイマは、アクティブなOCTOSPIのREQ信号の立下がりエッジ（アクティブなOCTOSPIがバスを解放）とリクエストするOCTOSPIへのACK信号の立上がりエッジ（要求しているOCTOSPIにバスが許可される）間の遅延を保証します。

この時間は、設定レジスタOCTOSPIM_CRのREQ2ACK_TIMEフィールドで定義されます。

マルチプレクス·モードでのピン配置

マルチプレクス·モードでは、バスの配置は次のように行われます。

- OCTOSPI1_NCS と OCTOSPI2_NCS は同じように機能するので、それらは、非マルチプレクス·モードでそれぞれのOCTOSPIM_Pn_NCSに割り当てる必要があります。
- その他のすべての信号は、I/Oマトリックスによって OCTOSPI1と同様に表示されます。

29.5 OCTOSPIMレジスタ

29.5.1 OCTOSPIM制御レジスタ（OCTOSPIM_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REQ2ACK_TIME [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUXEN
															rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **REQ2ACK_TIME[7:0]**：REQからACKまでの時間

マルチプレクス･モード（MUXEN = 1）では、このフィールドは2つのトランザクション間の時間を定義します。

この値はOCTOSPIクロックサイクル数 - 1の値です。

ビット15:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **MUXEN**：マルチプレクス･モード有効化

このビットは、2つのOCTOSPIの多重化を有効にします。

0：多重化はしません。よってアービトレーションもしません。

1：OCTOSPI1およびOCTOSPI2は、同一バス上で多重化されます。

29.5.2 OCTOSPIMポートn設定レジスタ（OCTOSPIM_PnCR）

アドレスオフセット：0x0000 + 0x4 * n（n = 1～2）

リセット値：0x0301 0111, 0x0705 0333

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	IOHSRC[1:0]		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC[1:0]		IOLEN
					rw	rw	rw						rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
						rw	rw			rw	rw			rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:25 **IOHSRC[1:0]**：ポートnのIO[7:4]ソース

このビットは、ポートnのソースIO[7:4]を選択します。

00：非マルチプレクス･モードではOCTOSPI1_IO[3:0]／マルチプレクス･モードではmultiplexed_IO[3:0]

01：非マルチプレクス･モードではOCTOSPI1_IO[7:4]／マルチプレクス･モードではmultiplexed_IO[7:4]

10：非マルチプレクス･モードではOCTOSPI2_IO[3:0]／マルチプレクス･モードでは未使用

11：非マルチプレクス･モードではOCTOSPI2_IO[7:4]／マルチプレクス･モードでは未使用

ビット24 **IOHEN**：ポートnのIO[7:4]有効化
このビットは、ポートnのIO[7:4]を有効にします。
0：ポートnのIO[7:4]を無効化
1：ポートnのIO[7:4]を有効化

ビット23:19 予約済みであり、リセット値に保持する必要があります。

ビット18:17 **IOLSRC[1:0]**：ポートnのIO[3:0]ソース
このビットは、ポートnのソースIO[3:0]を選択します。
00：非マルチプレクス･モードではOCTOSPI1_IO[3:0]／マルチプレクス･モードではmultiplexed_IO[3:0]
01：非マルチプレクス･モードではOCTOSPI1_IO[7:4]／マルチプレクス･モードではmultiplexed_IO[7:4]
10：非マルチプレクス･モードではOCTOSPI2_IO[3:0]／マルチプレクス･モードでは未使用
11：非マルチプレクス･モードではOCTOSPI2_IO[7:4]／マルチプレクス･モードでは未使用

ビット16 **IOLEN**：ポートnのIO[3:0]有効化
このビットは、ポートnのIO[3:0]を有効にします。
0：ポートnのIO[3:0]を無効化
1：ポートnのIO[3:0]を有効化

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **NCSSRC**：ポートnのNCSソース
このビットは、ポートnのソースNCSを選択します。
0：OCTOSPI1_NCS
1：OCTOSPI2_NCS

ビット8 **NCSEN**：ポートnのNCS有効化
このビットは、ポートnのNCSを有効にします。
0：ポートnのNCSを無効化
1：ポートnのNCSを有効化

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **DQSSRC**：ポートnのDQSソース
このビットは、ポートnのソースDQSを選択します。
0：非マルチプレクス･モードではOCTOSPI1_DQS／マルチプレクス･モードではmultiplexed_DQS
1：非マルチプレクス･モードではOCTOSPI2_DQS／マルチプレクス･モードでは未使用

ビット4 **DQSEN**：ポートnのDQS有効化
このビットは、ポートnのDQSを有効にします。
0：ポートnのDQSを無効化
1：ポートnのDQSを有効化

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CLKSRC**：ポートnのCLK/NCLKソース
このビットは、ポートnのCLK/NCLKソースを選択します。
0：非マルチプレクス･モードではOCTOSPI1_CLK/NCLK/マルチプレクス･モードでは multiplexed_CLK/CLKn
1：非マルチプレクス･モードではOCTOSPI2_CLK/NCLK/マルチプレクス･モードでは未使用

ビット0 **CLKEN**：ポートnのCLK/NCLK有効化
このビットは、ポートnのCLK/NCLKを有効にします。
0：ポートnのCLK/NCLKを無効化
1：ポートnのCLK/NCLKを有効化

29.5.3 OCTOSPIMレジスタマップ

表258.OCTOSPIMレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	OCTOSPIM _CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REQ2ACK_TIME [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUXEN
	リセット値									0	0	0	0	0	0	0	0																0
0x0004	OCTOSPIM _P1CR	Res.	Res.	Res.	Res.	Res.	IOHSRC [1:0]		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC [1:0]		IOLEN	Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
	リセット値						0	1	1						0	0	1							0	1			0	1			0	1
0x0008	OCTOSPIM _P2CR	Res.	Res.	Res.	Res.	Res.	IOHSRC [1:0]		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC [1:0]		IOLEN	Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
	リセット値						1	1	1						1	0	1							1	1			1	1			1	1

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

30 Hexadeca-SPIインタフェース（HSPI）

30.1 概要

HSPIは、シリアルPSRAM、シリアルNANDおよびシリアルNOR Flashメモリ、HyperRAM™およびHyperFlash™メモリなど、ほとんどの外部シリアルメモリを以下の機能モードでサポートしています。

- インダイレクトモード : すべての動作は、コマンド、アドレス、データ、および転送パラメータをプリセットするHSPIレジスタを使用して実行されます。
- 自動ステータスポーリングモード : 外部メモリステータスレジスタが定期的に読み出され、フラグのセット時に割込みを生成することができます。この機能は、レギュラコマンドプロトコルでのみ使用できます。
- メモリマップドモード : 外部メモリはメモリ空間に配置され、システムからは内部メモリであるかのように見え、読出しと書込みの両方の動作がサポートされます。

HSPIは、以下の関連するフレームフォーマットを備えたプロトコルをサポートしています。

- コマンド、アドレス、オルタネートバイト、ダミーサイクル、およびデータフェーズから成るレギュラコマンドフレームフォーマット
- HyperBus™フレームフォーマット

30.2 HSPI の主な特徴

- 機能モード : インダイレクト、自動ステータスポーリング、およびメモリマップドモード
- メモリマップドモードでの読出しおよび書込みのサポート
- シングル、デュアル、クワッド、およびオクタル通信に対応
- 2つのクワッドメモリまたは2つのオクタルメモリに並列にアクセスすることで8bitの同時送受信を可能にするデュアルメモリ構成
- 単一の16bitメモリにアクセスするHSPIモード
- SDR（シングルデータレート）およびDTR（ダブル転送レート）のサポート
- データストローブのサポート
- 完全にプログラム可能なOPコード
- 完全にプログラム可能なフレームフォーマット
- 読出し方向のメモリへのラップタイプのアクセスをサポート
- HyperBusのサポート
- 受信および送信用の内蔵FIFO
- 8、16、32bitのデータアクセスを許可
- DMAのプロトコルに対応
- インダイレクトモードで動作するためのDMAチャネル
- FIFO閾値、タイムアウト、動作完了、アクセスエラーによる割込み生成
- トランザクションの受け入れが1つに制限されたAHBインタフェース : メモリ側で前の転送が完了した後でのみ、AHBバス上で次の転送を受け入れるインタフェース

30.3 HSPI の実装

表259.STM32U5シリーズデバイス上のインスタンス

デバイス	OCTOSPI1	OCTOSPI2	OCTOSPIM	HSPI1
STM32U535/545	X	-	-	-
STM32U575/585	X	X	X	-
STM32U59x/5Ax	X	X	X	X
STM32U5Fx/5Gx	X	X	X	X

表260.HSPIの実装

機能	HSPI1	OCTOSPI1/2
HyperBus規格準拠	X	X
Xcella規格準拠	X	X
HSPI（JEDEC251ES）規格準拠	X	X
AMBA®AHB準拠のデータインタフェース	X	X
非同期AHBクロックとカーネルクロック	X	X
デュアルAHBインタフェース	X	X
機能モード：インダイレクト、自動ステータスポーリング、およびメモリマップドモード	X	X
デュアル·チップセレクトのサポート（NCS1およびNCS2）	-	-
メモリマップドモードでの読出しおよび書込みのサポート	X	X
デュアルクワッド構成	X	X
デュアルオクタル構成	X	-
SDR（シングルデータレート）およびDTR（ダブル転送レート）	X	X
データストローブ（DS、DQS）	X	X
完全にプログラム可能なOPコード	X	X
完全にプログラム可能なフレームフォーマット	X	X
受信および送信用の内蔵FIFO	X	X
8、16、32bitのデータアクセス	X	X
FIFO閾値、タイムアウト、動作完了、アクセスエラーによる割込み	X	X
デュアルOCTOSPIアービタ（通信調整）に準拠	-	X
拡張CSHTタイムアウト	X	X
メモリマップド書込み	X	X
リフレッシュカウンタ	X	X
GPDMAインタフェース	X	X
ハイスピードインタフェース	X	-
プリフェッチ無効化	-	-
プリフェッチ用ハードウェアおよびソフトウェア	-	-

30.4 HSPI機能詳細

30.4.1 HSPI ブロック図

以下に示すブロック図は、I/Oポートに接続された1つのHSPIに対応しています。

図160. 16bit構成のHSPIブロック図

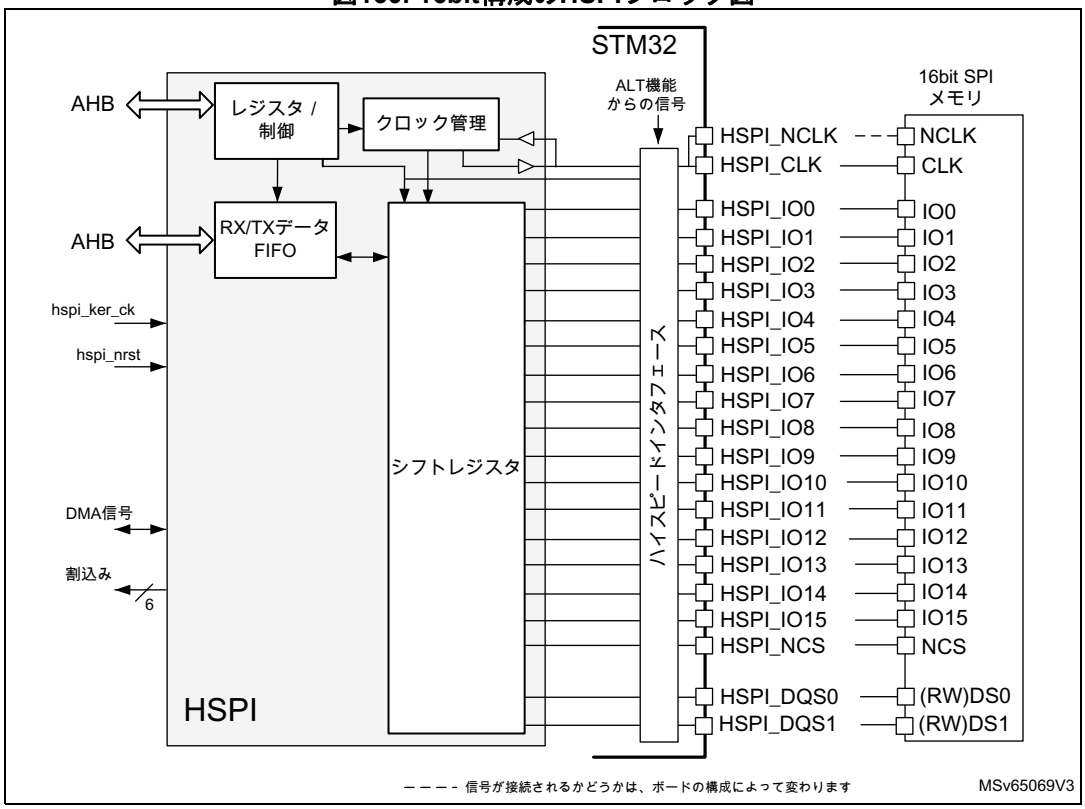


図161. デュアルオクタル構成のHSPIブロック図

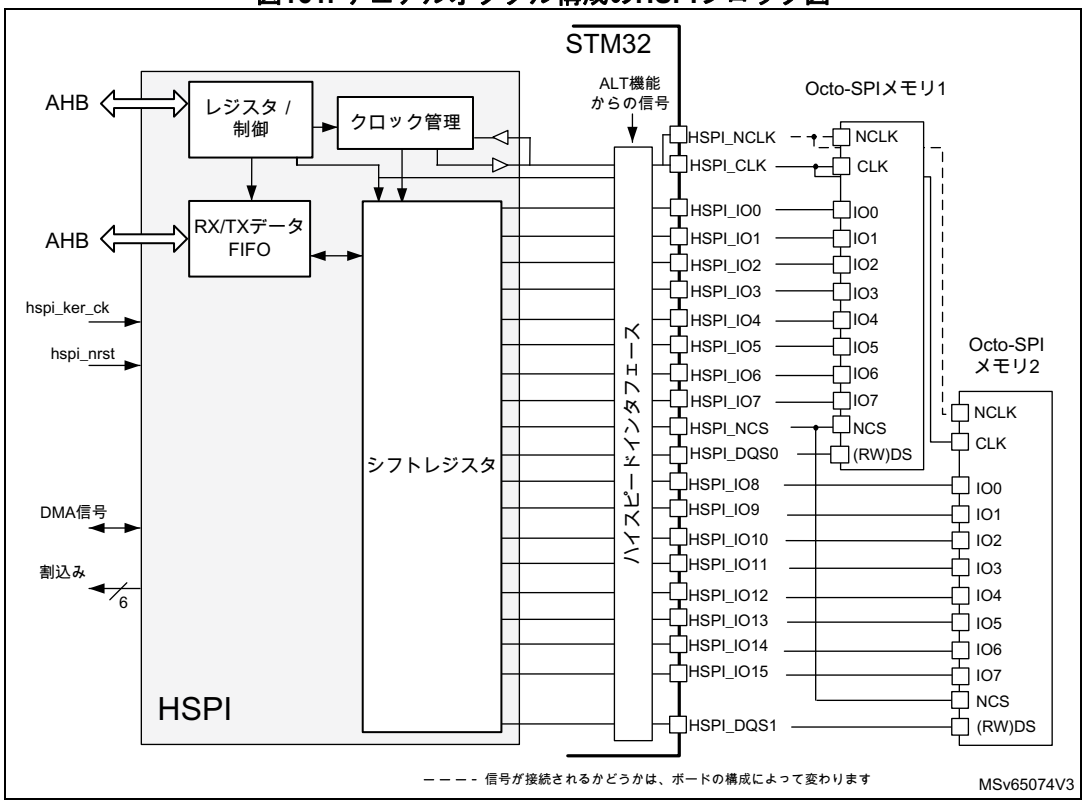


図162. オクタル構成のHSPIブロック図

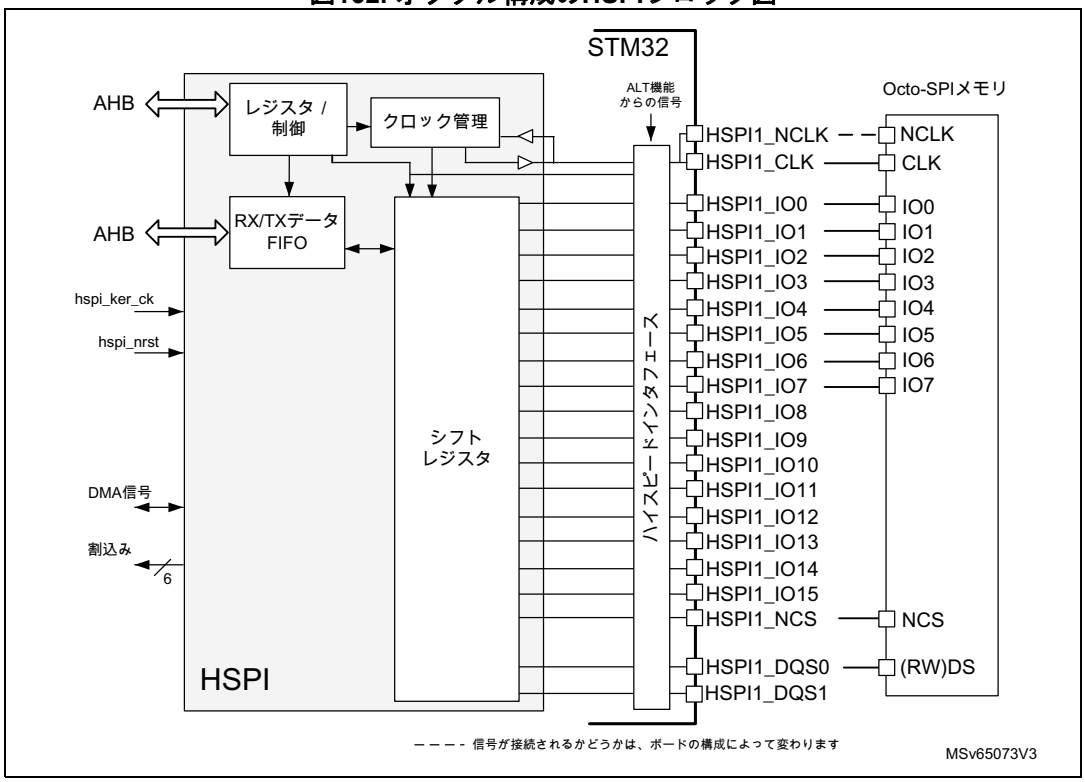


図163. クワッド構成のHSPIブロック図

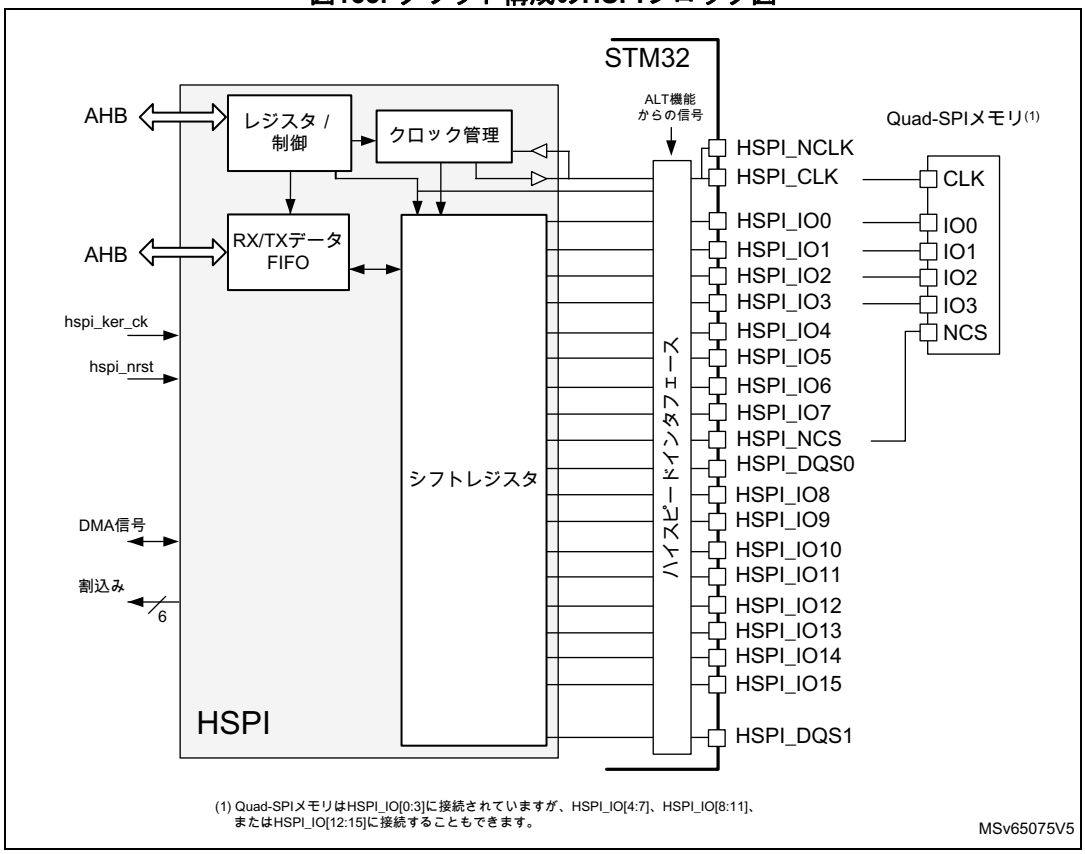
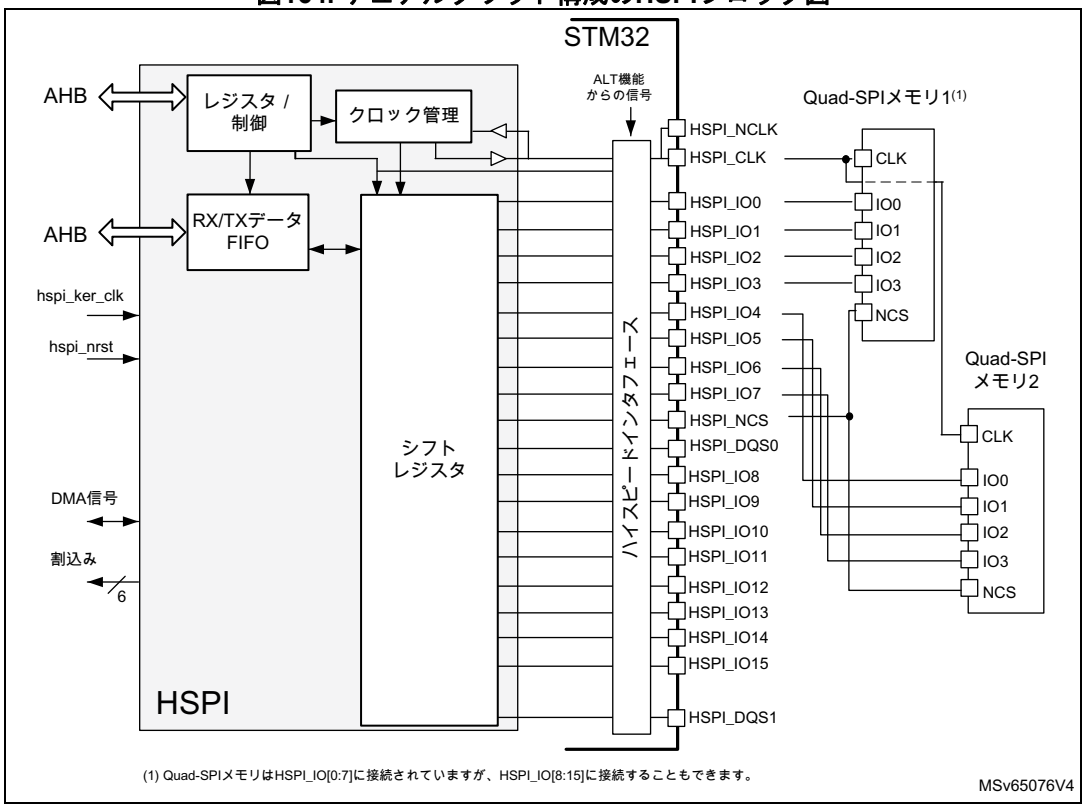


図164. デュアルクワッド構成のHSPIブロック図



30.4.2 HSPIピンおよび内部信号

表261.HSPIの入出力ピン

ピン名	タイプ	説明
HSPI_NCLK	出力	1.8V HyperBusプロトコルに対応したHSPI反転クロック
HSPI_CLK	出力	HSPIクロック
HSPI_IOn（n = 0～15）	入力／出力	HSPIデータピン
HSPI_NCS	出力	メモリのチップセレクト
HSPI_DQS0、1	入力／出力	メモリとの間のデータストローブ／書込みマスク信号

注意： **データの破損を避けるために、すべてのHSPI入出力ピンに同じ設定（出力速度、HSLV）を使用してください。**

表262.HSPI内部信号

信号名	タイプ	説明
hspi_hclk	入力	HSPI AHBクロック
hspi_ker_ck	入力	HSPIカーネルクロック
hspi_dma	NA	DMAリクエスト信号
hspi_it	出力	グローバル割込みライン（割込みの複数ソースについては表265を参照）

30.4.3 メモリに対するHSPIインタフェースのモード

HSPIは次のプロトコルをサポートしています。

- レギュラコマンドプロトコル
- HyperBusプロトコル

HSPIは機能モードに応じて、下記の6から21本の信号を使ってメモリとインタフェースを取ります。

- NCS：チップセレクト
- CLK：通信クロック
- NCLK：1.8V HyperBusプロトコルでのみ使用される反転クロック
- DQS0、DQS1：レギュラコマンドプロトコルでのみ使用されるデータストローブ
- IO[3:0]：LSB用データバス
- IO[7:4]：
  - デュアルクワッド構成およびオクタル構成で使用されるMSB用データバス
  - quad-SPIモードでの再割当てとして使用できるデータバス
- IO[15:8]：
  - デュアルクワッド、デュアルオクタル、および16bit構成で使用されるデータバスのMSB
  - octal-SPIモードでの再割当てとして使用できるデータバス
  - IO[15:12]およびIO[11:8]はquad-SPIモードでの再割当てとしても使用できます。

30.4.4 HSPIレギュラコマンドプロトコル

レギュラコマンドプロトコルでは、HSPIはコマンドを使って外部デバイスと通信します。各コマンドには以下の5つのフェーズを含めることができます。

- 命令フェーズ
- アドレスフェーズ
- オルタネートバイトフェーズ
- ダミーサイクルフェーズ
- データフェーズ

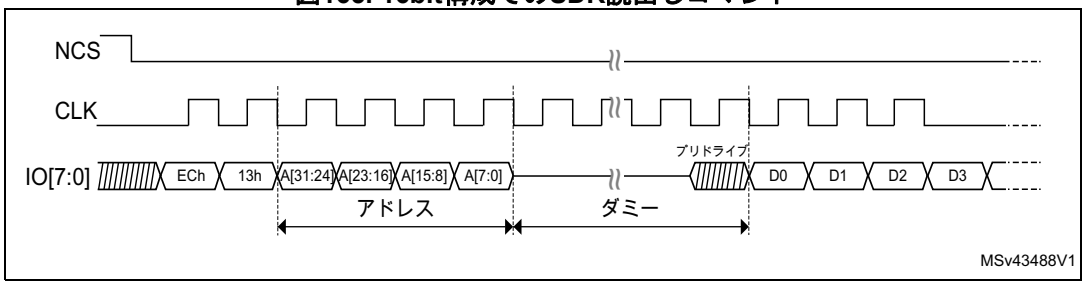
データフェーズのみ16bitを使用します。オクタル構成に関しては、命令、アドレス、オルタネートの各フェーズでは、バスのLSB 8bitのみが使用されます。

これらのフェーズのいずれもスキップするように設定できますが、サポートされるシングルフェーズコマンドは命令フェーズを備えたコマンドのみです。

NCSは各コマンドの開始前に立下がり、各コマンドの終了後に再び立上ります。

メモリマップドモードでは、読出しと書込みの両方の動作がサポートされており、その結果、いくつかの設定レジスタは書込み動作を指定するために複製されます（読出し動作は通常のレジスタを使用して設定されます）。

図165. 16bit構成でのSDR読出しコマンド



1. データ（D0、D1、D2など）は、IO[15:0]上で16bit構成モードで送信されます。オクタルモードに関しては、コマンドとアドレスのみがIO[7:0]上で送信されます。

レギュラコマンドプロトコルの具体的な機能は0x0100～0x01FCのオフセット範囲のレジスタによって設定されます。

命令フェーズ

このフェーズでは、1～4バイトの命令が外部デバイスに送信され、実行すべき動作のタイプが指定されます。送信される命令のサイズは HSPI_CCR の ISIZE[1:0] で設定され、命令は HSPI_IR の INSTRUCTION[31:0]にプログラムされます。

命令フェーズでは、オプションで以下を送信できます。

- 一度に1bit（single-SPIモードでIO0（SO信号）を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（octal-SPIモード、または16bit SPIモードでIO0～IO7を経由して）

これは、HSPI_CCRのIMODE[2:0]を使用して設定できます。

命令は、HSPI_CCRのIDTRを設定することで、DTR（ダブル転送レート）モードでクロックの立上がりおよび立下がりの各エッジで送信することができます。

HSPI_CCRのIMODE[2:0] = 000の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

メモリマップドモードでは、書込み動作に使用される命令はHSPI_WIRで指定され、命令フォーマットはHSPI_WCCRで指定されます。読出し動作に使用される命令と命令フォーマットはHSPI_IRとHSPI_CCRで指定されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す1～4バイトが外部デバイスに送信されます。送信されるアドレスバイト数は、HSPI_CCRレジスタのADSIZE[1:0]で設定されます。

インダイレクトモードと自動ステータスポーリングモードでは、送信されるアドレスバイトはHSPI_ARのADDRESS[31:0]に指定されます。メモリマップドモードでは、アドレスは（システムのどのマスタからも）AHB経由で直接与えられます。

アドレスフェーズでは、以下のものを送信できます。

- 一度に1bit（single-SPIモードでIO0（SO信号）を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（octal-SPIモード、または16bit SPIモードでIO0～IO7を経由して）

これは、HSPI_CCRのADMODE[2:0]を使用して設定できます。

アドレスは、HSPI_CCRのADDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送信することができます。

ADMODE[2:0] = 000の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

メモリマップドモードでは、書込み動作用のアドレスフォーマットはHSPI_WCCRにて指定されます。読出し動作用のアドレスフォーマットはHSPI_CCRにて指定されます。

警告： 一部のメモリ仕様では、各アドレスは16bitの値に対応すると見なされます。HSPIでは、各アドレスは8bitの値に対応すると見なされます。したがって、ソフトウェアはメモリレジスタにアクセスするときにアドレスを2倍する必要があります。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作のモードを制御するために、1～4バイトが外部デバイスに送信されます。送信されるオルタネートバイト数はHSPI_CCRのABSIZE[1:0]で設定されます。送信されるバイトはHSPI_ABRで指定されます。

オルタネートバイトフェーズでは、以下のものを送信できます。

- 一度に1bit（single-SPIモードでIO0（SO信号）を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（octal-SPIモード、または16bit SPIモードでIO0～IO7を経由して）

これは、HSPI_CCRのABMODE[2:0]を使用して設定できます。

オルタネートバイトは、HSPI_CCRのABDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送信することができます。

ABMODE[2:0] = 000の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

オルタネートバイトフェーズ中にフルバイトではなく2分の1バイト（4bit）だけを送信する必要があるかもしれません。たとえば、dual-SPIモード使用時で、オルタネートバイトに2サイクルだけが使われる場合などです。

この場合、ファームウェアによってquad-SPIモード（ABMODE[2:0] = 011）を使用し、ALTERNATE[31:0]のビット7および3を1（IO3ラインをハイに保持）に、HSPI_IRのビット6および2を0（IO2ラインをローに保持）に設定して1バイトを送信できます。

送信される4bitの上位2bitはALTERNATE[31:0]のビット5:4にセットされ、下位2bitはビット1:0にセットされます。たとえば、4bitの値2（0010）がIO0/IO1で送信される場合、ALTERNATE[31:0]は0x8A（1000_1010）にセットされている必要があります。

メモリマップドモードでは、書込み動作用に使用されるオルタネートバイトはHSPI_WABRで指定され、オルタネートバイト･フォーマットはHSPI_WCCRで指定されます。読出し動作用に使用されるオルタネートバイトとオルタネートバイト･フォーマットはHSPI_ABRとHSPI_CCRで指定されます。

ダミーサイクルフェーズ（メモリ待ち時間）

ダミーサイクルフェーズでは、データを送受信せずに1～31回の空サイクルが設けられます。これは、より高いクロック周波数が使われている場合に、外部デバイスにデータフェーズの準備をする時間を与えるためです。このフェーズでのサイクル数は、HSPI_TCRのDCYC[4:0]に指定されます。SDRモードとDTRモードのどちらでも、この持続時間はフルCLKサイクル数として指定されます。

DCYC[4:0] = 00000の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

データ信号を出力モードから入力モードに変更するのに十分な「ターンアラウンド」時間を確保するために、dual-SPI、quad-SPI、octal-PSI、または16bit SPIモードを使用して外部デバイスからデータを受信する場合、少なくとも1ダミーサイクルが必要です。

メモリマップドモードでは、書込み動作用のダミーサイクルはHSPI_WTCRに指定されます。読出し動作用のダミーサイクルはHSPI_TCRで指定されます。

データフェーズ

データフェーズでは、外部デバイスとの間で任意のバイト数を送受信できます。

インダイレクトモードでは、送受信するバイト数はHSPI_DLRで指定されます。このモードでは、外部デバイスに送信するデータをHSPI_DRに書き込む必要があります。インダイレクト読出しモードでは、外部デバイスから受信するデータはHSPI_DRを読み出して取得します。

自動ステータスポーリングモードでは、受信するバイト数はHSPI_DLRにて指定され、外部デバイスから受信するデータはHSPI_DRを読み出して取得できます。

メモリマップドモードでは、読み書きされるデータは、直接AHBを経由してCortexコアまたはDMAとの間で送受信されます。

データフェーズでは、以下のものを送受信できます。

- 一度に1bit（Single-SPIモードでIO0/IO1（SO/SIそれぞれ）を経由して）
- 一度に2bit（Dual-SPIモードでIO0/IO1を経由して）
- 一度に4bit（Quad-SPIモードでIO0～IO3を経由して）
- 一度に8bit（Octal-SPIモードでIO0～IO7を経由して）
- 一度に16bit（16bit SPIモードでIO0～IO15を経由して）

これは、HSPI_CCRのDMODE[2:0]を使用して設定できます。

データは、HSPI_CCRのDDTRをセットすることでDTRモードで（クロックの立上がりおよび立下がりの各エッジで）送受信することができます。

DMODE[2:0] = 000の場合、データフェーズはスキップされるので、コマンドシーケンスはNCSを立ち上げることですぐに終了します。この設定はインダイレクト書込みモードでのみ使用してください。

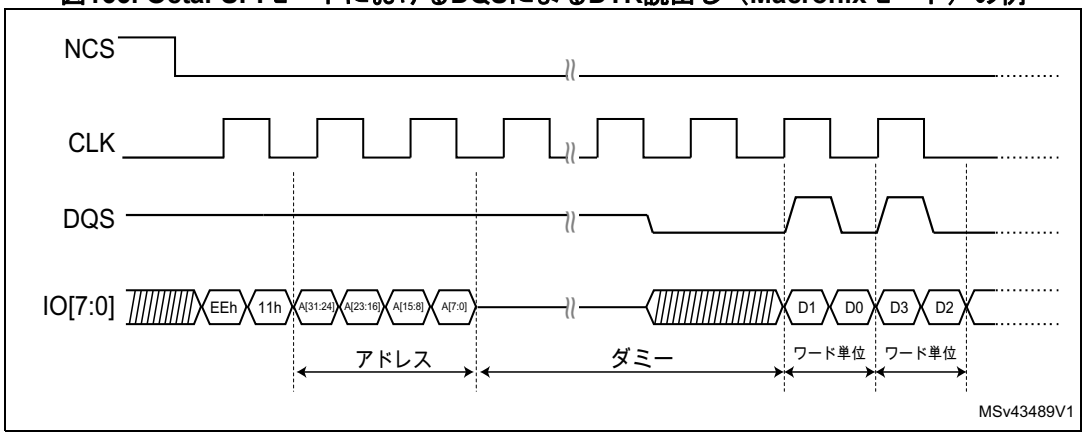
メモリマップドモードでは、書込み動作用のデータフォーマットはHSPI_WCCRにて指定されます。読出し動作用のデータフォーマットはHSPI_CCRにて指定されます。

DQSの使用

DQS信号は、デバイスがデータに合わせてDQSをトグルしているときの読出しトランザクション中のデータストローブに使用できます。

DQS管理はHSPI_CCRのDQSEを設定することによって有効にできます。

図166. Octal-SPIモードにおけるDQSによるDTR読出し（Macronixモード）の例



30.4.5 HSPIレギュラコマンドプロトコル信号インタフェース

Single-SPIモード

レガシー SPIモードでは、1bitの順次送受信のみが可能です。このモードでは、データはSO信号（そのI/OはIO0と共有される）を経由して外部デバイスに送信されます。外部デバイスから受信するデータは、SI（そのI/OはIO1と共有される）を経由して到着します。

HSPI_CCRおよびHSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ001にセットすることによって、このsingle-bitモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをSingle-SPIモードに設定する場合：

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2は出力モードで、強制的に0に設定されます（書込み保護機能を無効にするため）。
- IO3は出力モードで、強制的に1に設定されます（ホールド機能を無効にするため）。
- IO4からIO15は出力モードで、強制的に0に設定されます。

これは、DMODE[2:0] = 001であれば、ダミーフェーズにも該当します。

Dual-SPIモード

Dual-SPIモードでは、IO0/IO1信号を経由して2bitが同時に送受信されます。

HSPI_CCRおよびHSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ010にセットすることによって、dual-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをDual-SPIモードに設定する場合：

- IO0/IO1は、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2は出力モードで、強制的に0に設定されます。
- IO3は出力モードで、強制的に1に設定されます。
- IO4からIO15は出力モードで、強制的に0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 010の場合、IO0/IO1は常にハイインピーダンスです。

Quad-SPIモード

Quad-SPIモードでは、IO0/IO1/IO2/IO3信号を経由して4bitが同時に送受信されます。

HSPI_CCRおよびHSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ011にセットすることによって、quad-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

各フェーズをQuad-SPIモードに設定する場合：

- IO0～IO3はすべて、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO4からIO15は出力モードで、強制的に0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 011の場合、IO0～IO3はすべてハイインピーダンスです。

Octal-SPIモード

通常のOctal-SPIモードでは、IO[0:7]信号を経由して8bitが同時に送受信されます。

HSPI_CCRおよびHSPI_WCCRのIMODE、ADMODE、ABMODE、およびDMODEフィールドをそれぞれ100にセットすることによって、octal-SPIモードを使用できるように、フェーズごとに個別に設定することができます。

Octal-SPIモードに設定されている各フェーズでは、IO[0:7]はすべてデータフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE[2:0] = 100の場合、IO[0:7]はすべてハイインピーダンスです。

HSPIモード

HSPIモードでは、データフェーズ中にIO[0:15]信号を経由して16bitが同時に送受信されます。

HSPIモードを使用するには、次のフェーズを個別に設定する必要があります。

1. （HSPI_CCRおよびHSPI_WCCRで）IMODE、ADMODE、ABMODEフィールドをそれぞれ100にセットします。
2. （HSPI_CCRおよびHSPI_WCCRで）DMODEフィールドを101にセットします。

HSPIモードに設定されている各フェーズでは、IO[0:15]はすべてデータフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE[2:0] = 101の場合、IO[0:15]はすべてハイインピーダンスです。

IO[8:15]はHSPIモードでのみ使用されます。このモードを使用できるように設定されたフェーズが1つもない場合、IO[8:15]に対応するピンはHSPIがアクティブなときでも、他の機能に使用することができます。

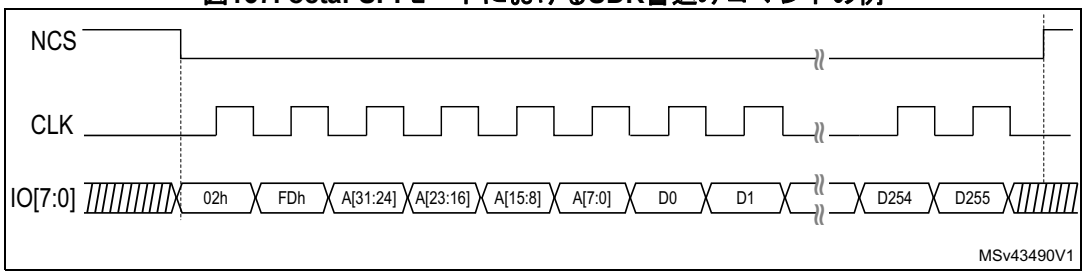
シングルデータレート（SDR）モード

すべてのフェーズは、デフォルトでシングルデータレート（SDR）モードで動作します。

SDRモードでは、HSPIがIO0/SOおよびIO1～IO15信号を駆動する場合、これらの信号はCLKの立下がりエッジでのみ遷移します。

SDRモードでデータを受信する場合、HSPIは外部デバイスもCLKの立下がりエッジを使用してデータを送信するとみなします。デフォルトでは（HSPI_TCRにおいてSSHIFT = 0の場合）、信号は次のCLKの（立上がり）エッジを使用してサンプリングされます。

図167. octal-SPIモードにおけるSDR書込みコマンドの例



ダブル転送レート（DTR）モード

HSPI_CCRのIDTR、ADDTR、ABDTR、およびDDTRをセットすることによって、命令、アドレス、オルタネートバイト、およびデータの各フェーズをDTRモードで動作するように設定できます。

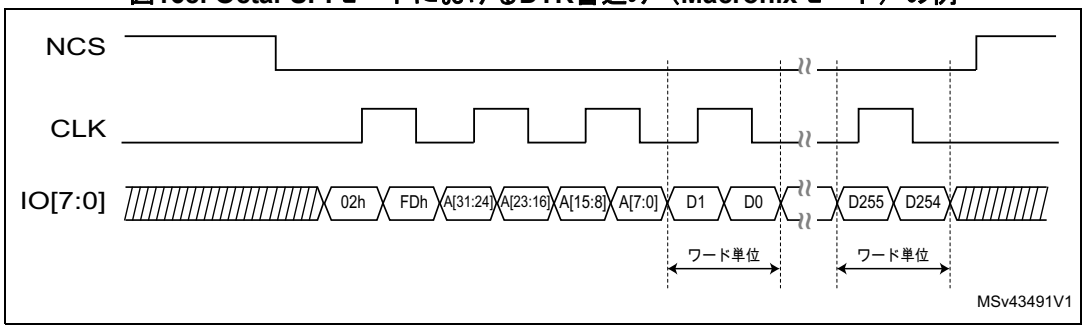
メモリマップドモードでは、書込み動作の各フェーズのDTRモードはHSPI_WCCRで指定されます。読出し動作の各フェーズのDTRモードはHSPI_CCRで指定されます。

DTRモードでは、HSPIが命令、アドレス、オルタネートバイトの各フェーズでIO0/SO、IO1～IO7信号を駆動している場合、CLKの立下がりエッジおよび立上がりエッジのそれぞれでビットを送受信します。

DTRモードでは、HSPIがデータフェーズでIO0～IO15信号を駆動している場合、CLKの立下がりエッジおよび立上がりエッジのそれぞれでビットを送信します。

DTRモードでデータを受信する場合、HSPIは外部デバイスもCLKの立上がりと立下がりの両方のエッジを使用してデータを送信するとみなします。HSPI_CCRでDDTR = 1の場合、ソフトウェアはHSPI_TCRのSSHIFTをクリアする必要があります。したがって、信号は（次の反対側のエッジの）1/2CLKサイクル後にサンプリングされます。

図168. Octal-SPIモードにおけるDTR書込み（Macronixモード）の例



デュアルクワッド構成

HSPI_CRにおいてDMM=1の場合、HSPIはデュアルメモリ構成です。すなわち、DMODE = 011の場合、2つの外部quad-SPIデバイス（デバイスAとデバイスB）が使用され、サイクルごとに8bitの送受信（DTRモードでは16bit）を行うことで、スループットを効果的に2倍にします。

どちらのデバイス（AまたはB）も同じCLKとNCS信号を使用しますが、それぞれが個別のIO0～IO3信号を有しています。

デュアルクワッド構成は、SDRモードまたはDTRモードと組合せて使用できるだけでなく、single-SPI、dual-SPI、およびquad-SPIモードとも組合せて使用できます。

HSPI_DCR1のDEVSIZE[4:0]で指定されているデバイスのサイズは、外部デバイスの総容量（コンポーネント1個のサイズの2倍）を反映していなければなりません。

アドレスXが偶数の場合、HSPIがアドレスXに対して与えるバイトはデバイスAのアドレスX/2にあるバイトであり、HSPI　がアドレスX+1に対して与えるバイトはデバイスBのアドレスX/2にあるバイトです。言い換えれば、偶数アドレスにあるバイトはすべてデバイスAに、奇数アドレスにあるバイトはすべてデバイスBに格納されます。

デュアルクワッド構成でデバイスのステータスレジスタを読み出す場合、レギュラコマンドプロトコルで読み出すバイト数の2倍のバイトを読み出す必要があります。各デバイスがステータスレジスタのフェッチ命令後に有効な8bitを提供する場合、HSPIはデータ長2バイト（16bit）で設定する必要があり、それによりHSPIは各デバイスから1バイトを受信します。

各デバイスが16bitのステータスを提供する場合、HSPIは、4バイトを読み出して、デュアルクワッド構成の両方のデバイスのステータスビットをすべて取得できるように設定する必要があります。リザルト（データレジスタ内）の最下位バイトはデバイスAのステータスレジスタの最下位バイトです。次のバイトはデバイスBのステータスレジスタの最下位バイトです。さらに、データレジスタの3番目のバイトはデバイスAの2番目のバイトです。4番目のバイトはデバイスBの2番目のバイトになります（デバイスに16bitのステータスレジスタがある場合）。

デュアルクワッド構成では、常に偶数バイトをアクセスする必要があります。このため、DMM = 1のとき、HSPI_DLRのDL[31:0]のビット0は1のままとなります。

デュアルクワッド構成では、デバイスAのインタフェース信号の動作は基本的に通常モードの動作と同じです。デバイスBのインタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、デバイスAのものとまったく同じです。つまり、どちらのデバイスも常に同じ命令や同じアドレスを受信するということです。

データフェーズ中は、2つのバスAIOxとBIOxは並行してデータ転送を行いますが、デバイスAに対する送信（または受信）データはデバイスBのものとは異なります。

デュアルオクタル構成

HSPI_CRにおいてDMM = 1の場合、HSPIはデュアルメモリ構成です。すなわち、DMODE = 100の場合、2つの外部octal-SPIデバイス（デバイスAとデバイスB）が使用され、サイクルごとにDTRモードで32bitを受信することで、スループットならびに容量が効果的に2倍になります。

どちらのデバイス（AまたはB）も同じCLKとNCS信号を使用しますが、それぞれが個別のIO0～IO7信号を有しています。

デュアルオクタル構成は、single-SPI、dual-SPI、quad-SPI、およびoctal-SPIの各モードと組み合わせてDTRモードで排他的に使用できます。

HSPI_DCR1のDEVSIZE[4:0]で指定されているデバイスのサイズは、外部デバイスの総容量（コンポーネント1個のサイズの2倍）を反映していなければなりません。

アドレスXが偶数の場合、HSPIがアドレスXに対して与えるバイトはデバイスAのアドレスX/2にあるバイトであり、HSPIがアドレスX+1に対して与えるバイトはデバイスBのアドレスX/2にあるバイトです。言い換えれば、偶数アドレスにあるバイトはすべてデバイスAに、奇数アドレスにあるバイトはすべてデバイスBに格納されます。

デュアルオクタルDTRモードでデバイスのステータスレジスタを読み出す場合、通常のDTRモードで読み出すバイト数の2倍のバイトを読み出す必要があります。各デバイスがステータスレジスタのフェッチ命令後に有効な8bitを2回提供する場合、HSPIはデータ長4バイトで設定する必要があります。そのLSBはデバイスAのLSBであり、3番目のバイトはデバイスBのLSBです。

各デバイスが16bitのステータスを提供する場合、HSPIは、4バイトを読み出して、デュアルオクタルDTRモードの両方のデバイスのステータスビットをすべて取得できるように設定する必要があります。この場合、取得されるステータスビットの順序は以下のとおりです。

- 最初のバイト：デバイスAのLSB
- 2番目のバイト：デバイスAの2番目のバイト

- 3番目のバイト：デバイスBのLSB
- 4番目のバイト：デバイスBの2番目のバイト

DTRモードを使用するインダイレクトモードでは、HSPIモードで常に4の倍数のバイト数をアクセスする必要があります。このため、DMODE[2:0] = 101のとき、HSPI_DLRのDL[31:0]ビットフィールドのビット0とビット1は0のままとなります。

デュアルオクタル構成では、デバイスAのインタフェース信号の動作は基本的に通常モードの動作と同じです。デバイスBのインタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、デバイスAのものとまったく同じです。つまり、どちらのデバイスも常に同じ命令や同じアドレスを受信するということです。

データフェーズ中は、2つのバスAIOxとBIOxは並行してデータ転送を行いますが、デバイスAに対する送信（または受信）データはデバイスBのものとは異なります。

注： **デュアルオクタル構成では、可変の遅延はサポートされていません。**

HSPIモード

HSPI_CCRのDMODE[2:0] = 0b101の場合、HSPIは単一の16bitメモリ構成となります。つまり、DMODE[2:0] = 0b101の場合、DMM値は無視されます。1つの外部HSPIデバイスを使用してサイクルごとに16bit（DTRモードでは32bit）を送受信することで、スループットを効果的に2倍にします。

デバイスは、2つの個別のDQS信号、つまり8つのLSB用のDQS0と8つのMSB用のDQS1を送受信します。

HSPIモードは、SDRモードまたはDTRモードと組合せて使用できるだけでなく、single-SPI、dual-SPI、quad-SPI、およびoctal-SPIモードとも組合せて使用できます。

HSPI_DCR1のDEVSIZE[4:0]で指定されているデバイスのサイズは、外部デバイスの総容量を反映していなければなりません。

SDRモードでは、HSPIモードで常に2の倍数のバイト数をアクセスする必要があります。このため、DMODE = 101のとき、HSPI_DLRのDL[31:0]のビット0は0のままとなります。

DTRモードでは、HSPIモードで常に4の倍数のバイト数をアクセスする必要があります。このため、DMODE = 101のとき、HSPI_DLRのDL[31:0]のビット0とビット1は0のままとなります。

30.4.6 HyperBusプロトコル

HSPIはHyperBusプロトコルを使って外部デバイスと通信することができます。

HyperBusは、動作電圧に応じて以下のように、8bitデータモードで11～12本のピン、または16bitデータモードで19～20本のピンを使用します。

- 8bitデータモードでは双方向データバスとしてのIO[7:0]、16bitデータモードでは双方向データバスとしてのIO[15:0]
- データ読み書きストローブおよび遅延挿入のためのRWDS（DQSピンに配置）
- NCS
- CLK
- 1.8V動作のNCLK（このモードをサポートするには、デバイスに1.8Vが供給されていなければなりません）

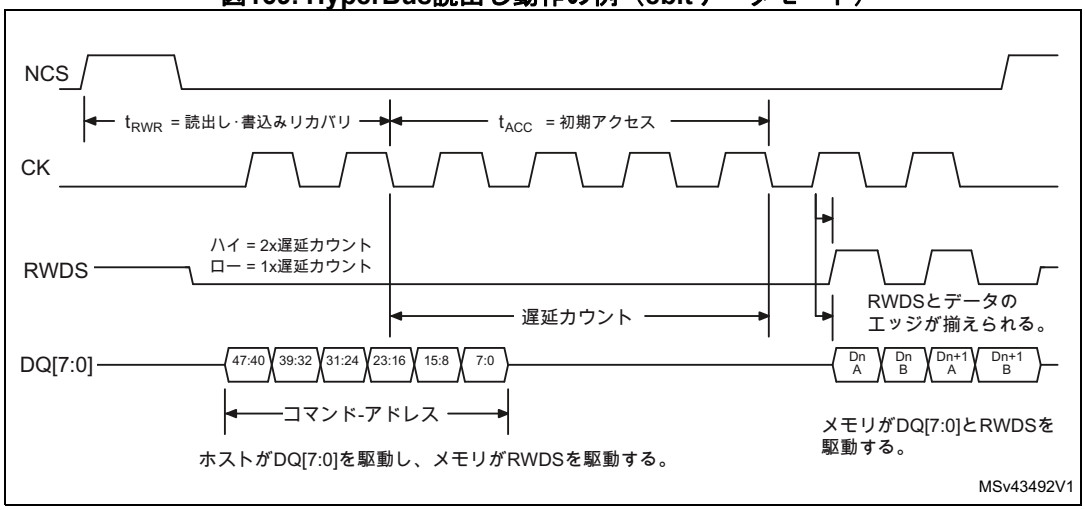
HyperBusでは、コマンド指定もオルタネートバイトも必要としません。結果として、トランザクションのタイミングを定義するために個別のレジスタセットが使用されます。

HyperBusのフレームは次の2つのフェーズで構成されます。

- コマンド／アドレスフェーズ
- データフェーズ

NCSはトランザクションの開始前に立下り、各トランザクションの終了後に再び立上ります。

図169. HyperBus読出し動作の例（8bitデータモード）



HyperBusの具体的な機能は0x0200～0x02FCのオフセット範囲のレジスタによって設定されます。

コマンド／アドレスフェーズ

この初期フェーズでは、HSPIは外部デバイスとの間で実施する動作を指定するために、IO[7:0]を経由して48bitを送信します。

表263.コマンド／アドレスフェーズの説明

CAビット	ビット名	説明
47	R/W#	トランザクションが読出しか書込みかを識別します。
46	アドレス空間	トランザクションがメモリか、レジスタ空間にアクセスしているかを示します。
45	バーストタイプ	バーストがリニアか、ラップトであるかを示します。
44 ~ 16	行アドレスおよび上位列アドレス	行アドレスおよび上位列アドレスを選択します。
15～3	予約済み	-
2～0	下位列アドレス	ハーフページ内の開始16bitワードを選択します。

アドレス空間はHSPI_DCR1のメモリタイプを示すMTYP[2:0]によって設定します。

デバイスの合計サイズはHSPI_DCR1のDEVSIZE[4:0]に設定します。マルチチップ製品（MCP）の場合、デバイスサイズはすべてのMCPダイのすべてのサイズの合計です。

警告： **一部のメモリ仕様では、各アドレスは16bitの値に対応すると見なされます。HSPIでは、各アドレスは8bitの値に対応すると見なされます。したがって、ソフトウェアはメモリレジスタにアクセスするときにアドレスを2倍する必要があります。**

初期遅延を設けた読出し／書込み動作

HyperBusの読出しおよび書込み動作では、次の2つのタイミングを考慮する必要があります。

- t_{RWR}：デバイスの最小限の読出し／書込みリカバリ時間（HSPI_HLCRのTRWR[7:0]で定義）
- t_{ACC}：メモリ遅延に応じたデバイスのアクセス時間（HSPI_HLCRのTACC[7:0]で定義）

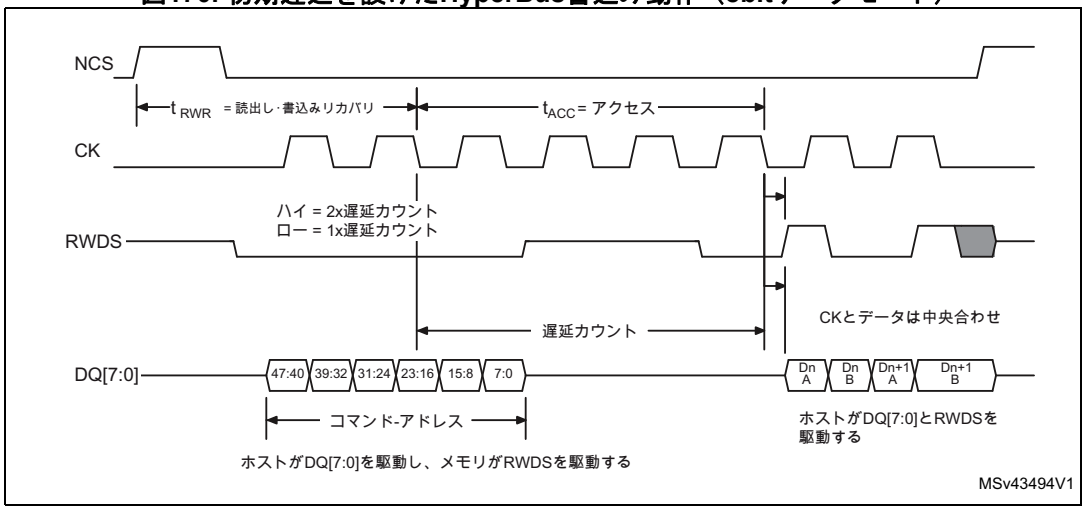
読出し動作中、デバイスはRWDSを次の2通りの方法で使用します（図169を参照）。

- コマンド／アドレスフェーズ時は、追加の遅延を要求するため
- データフェーズ時は、データストローブのため

書込み動作中、RWDSは次のように使用されます。

- デバイスによって、コマンド／アドレスフェーズ時に追加の遅延を要求するために。
- HSPIによって、データフェーズ時にデータ書込みマスキングのために。

図170. 初期遅延を設けたHyperBus書込み動作（8bitデータモード）



追加の遅延を設けた読出し／書込み動作

デバイスに追加の遅延が必要な場合（たとえばSDRAMのリフレッシュ期間中）、コマンド／アドレスフェーズ中にRWDS信号の1つの期間中にRWDSを1につなげる必要があります。

デバイスの要求を満たすために、HSPIによって追加のt_{ACC}期間が追加されます。

図171. 追加の遅延を設けたHyperBus読出し動作（8bitデータモード）

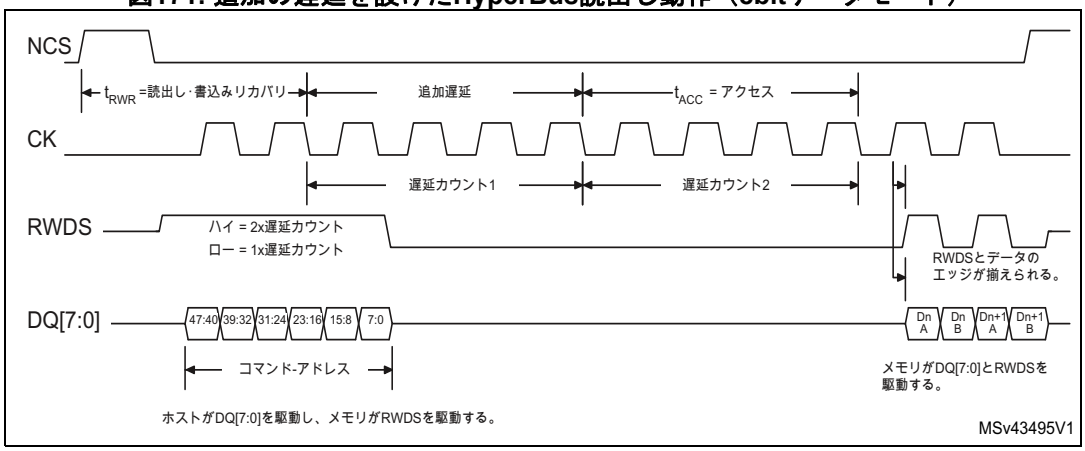
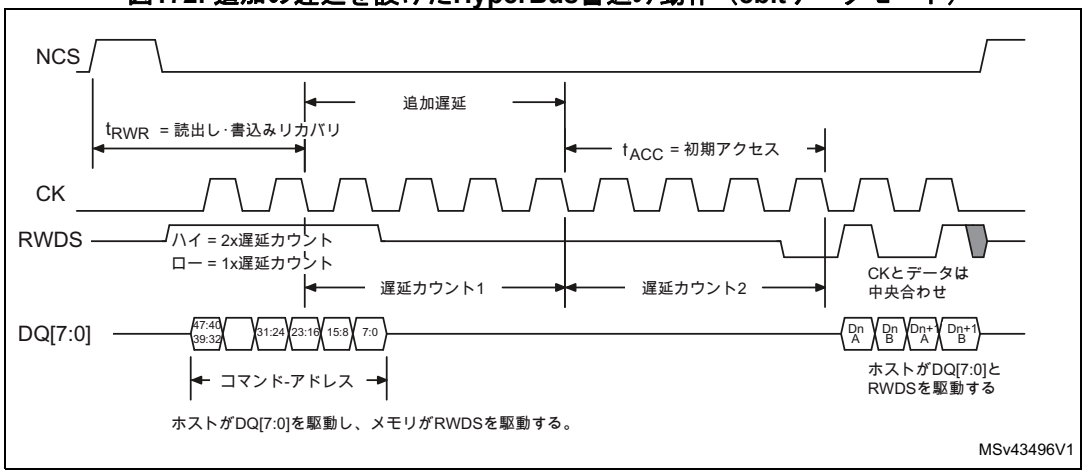


図172. 追加の遅延を設けたHyperBus書込み動作（8bitデータモード）



固定遅延モード

デバイスまたはアプリケーションによっては、上に述べたような可変の遅延を設けた動作を望まないものもあります。

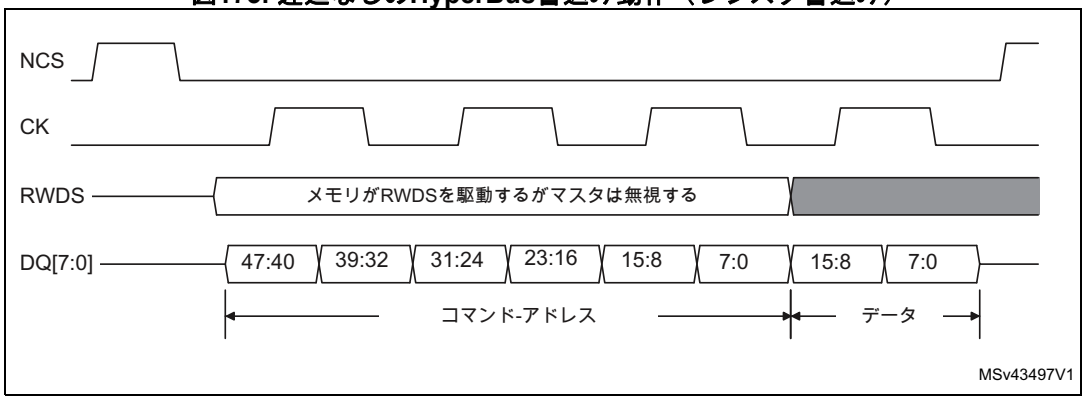
HSPI_HLCRのLMをセットすることで、遅延を強制的に2 x t_{ACC}にすることができます。

このHSPI遅延モードでは、RWDS信号の状態がHSPIによって考慮されず、常に追加の遅延が追加されて、固定の2xt_{ACC}の遅延時間が発生します。

遅延なしの書込み動作

また、デバイスによっては書込み動作にゼロ遅延を要求する可能性もあります。この書込みゼロ遅延は、HSPI_HLCRのWZLを設定することによって強制することができます。

図173. 遅延なしのHyperBus書込み動作（レジスタ書込み）

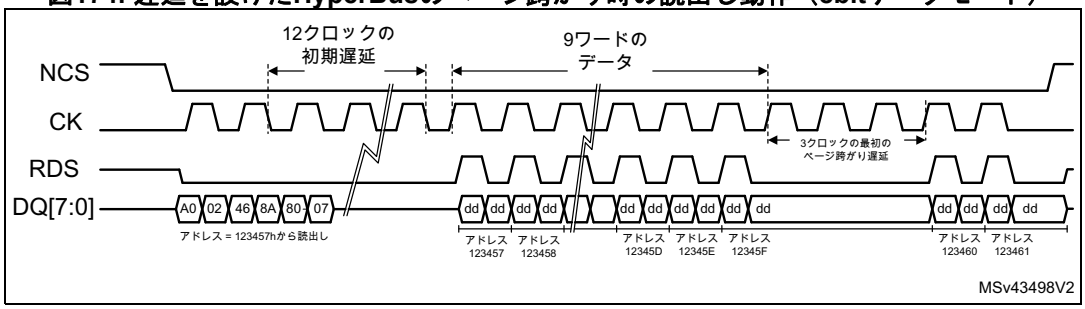


読出し動作中のページまたがり時の遅延

デバイスによっては、ページをまたがって読出し動作をする際に追加の遅延が必要になることがあります。

ページアクセスには初期遅延を考慮する必要があるので、その結果、最初のアクセスがページ境界に近いと、t_{ACC}時間を考慮するためにページまたがり時に自動的に遅延が追加されます。

図174. 遅延を設けたHyperBusのページ跨がり時の読出し動作（8bitデータモード）



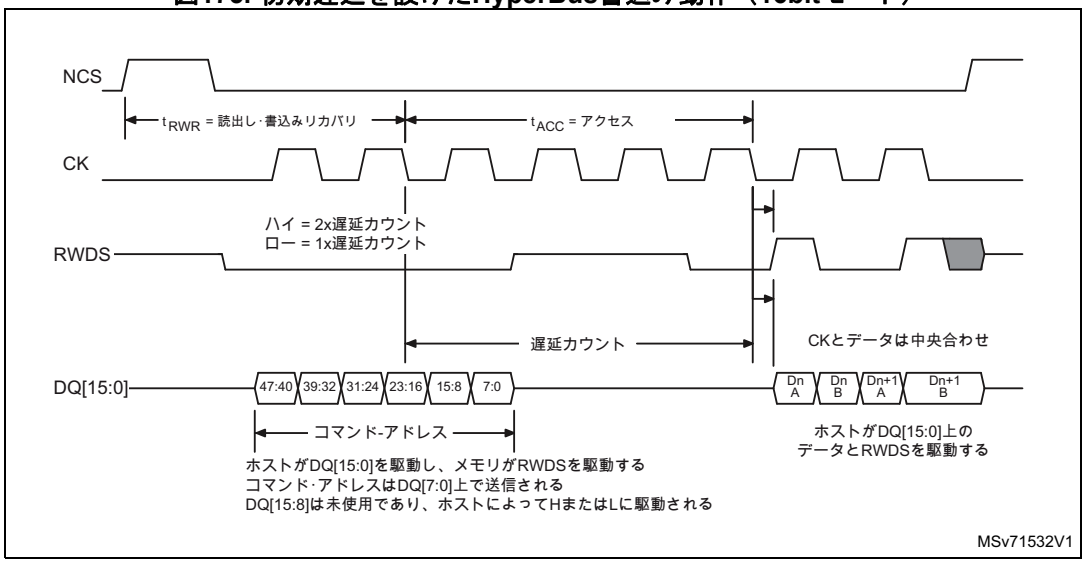
HyperBusを使用した16bitデータ転送

HyperBusプロトコルでは、HSPI_CRでDMM = 1のとき、HSPIはデュアルオクタル構成（16bitデータ転送）をサポートしています。1つのオクタルHyperBusメモリがIO0～IO7に接続され、もう1つがIO8～IO15に接続されます。これらのメモリは、専用信号であるRWDS以外のすべての信号を共有します。

16bitデータ転送では、DMODE[2:0]は101に等しくなければなりません。それ以外のDMODE[2:0]の値は8bitデータ転送に対応します（HyperBusプロトコルにはクワッドビット、デュアルビット、シングルビットのデータ転送は存在しません）。コマンド･アドレスフェーズは、HyperBusプロトコルでは常に8bit（IO8～IO15）を使用します。メモリ空間に（IO0～IO15から）アクセスする書込みまたは読出し動作では、たとえば図175に示すように、データのみが16bitになります。メモリレジスタ･アクセスでは、図173に示すようにデータは8bitです（IO0からIO7まで、IO8からIO15は使用されず、コントローラによって駆動されます）。

プロトコル･レベルでのインタフェースの動作は、デュアルオクタルのHyperBus構成で可変の遅延がサポートされていないことを除けば、前述のHyperBusオクタル構成と全く同じです。HSPI_HLCRのLMをセットする必要があります。

図175. 初期遅延を設けたHyperBus書込み動作（16bitモード）



30.4.7 特有の機能

HSPIは次のようないくつかの特有の機能をサポートしています。

- ラップのサポート
- NCS境界とリフレッシュ

ラップのサポート

HSPIでは、HyperBusプロトコルで定義されているハイブリッドラップがサポートされています。ハイブリッドラップは、レギュラコマンドプロトコルでもサポートされます。

ハイブリッドラップでは、トランザクションは最初のラップの後もインクリメントアクセスで続行できます。

対象のメモリでサポートされているラップのサイズは、HSPI_DCR2のWRAPSIZEで設定します。

ラップはメモリの読出し方向でのみサポートされます。また、データサイズ = 4バイトの場合のみサポートされます。ラップド読出しは、HyperBusとレギュラコマンドプロトコルの両方でサポートされます。ラップド読出しアクセスを有効にするには、ラップド読出しアクセス特性に応じた専用のHSPI_WPxxxレジスタをプログラムする必要があります。専用のHSPI_WPxxxレジスタは、HyperBusとレギュラコマンドモードの両方に適用されます。

対象のメモリがハイブリッドラップをサポートしていない場合は、WRAPSIZEは0にセットする必要があります。

注： **ハイブリッドラップでは、（ラップ·コマンドの後に開始される）正しいデータ·プリフェッチが満たされるように、ラップされないレジスタ（HSPI_CCR、HSPI_TCR、HSPI_IR）がメモリ構成に従って設定される必要があります。**

ラップ動作はリフレッシュによって中断することはできません。リフレッシュイベントはラップの完了後にのみ考慮されます。

NCS境界とリフレッシュ

HSPIトランザクションを調整するために次の2つのプロセスをアクティブにすることができます。

- NCS境界
- リフレッシュ

NCS境界機能によって、トランザクションは整列されたアドレスの境界に制限されます。整列させるアドレスのサイズは、HSPI_DCR3のCSBOUND[4:0]で設定され、$2^{CSBOUND}$に等しくなります。

たとえば、CSBOUND[4:0] = 0x4の場合、境界は2^4 = 16バイトにセットされます。その結果、LSBアドレスが0xFに等しくなるたびに、および新しいトランザクションが発行されて次のデータに取りかかるたびに、NCSがリリースされます。

CSBOUND[4:0] = 0の場合、この機能は無効になります。最小値の3が推奨されます。

書込みモードでは、Flashメモリの他のページをプログラムするためにコマンドが必要なので、NCS境界機能は書込みモードのFlashメモリデバイスには使用できません。

リフレッシュ機能では、トランザクションの期間をHSPI_DCR4のREFRESH[31:0]にプログラムされた値に制限します。期間はサイクル数で表されます。これにより、外部RAMは定期的に内部リフレッシュ動作を実行することができます。

リフレッシュ値は、コマンド、アドレス、オルタネート／ダミーのフェーズを含む最小トランザクションサイズ（サイクル数の単位で）より大きくなければなりません。

NCS境界とリフレッシュが同時に有効にされた場合、NCSは最初の条件が合致した時に解除されます。

中断された転送後の再開始

読取りまたは書込み動作がタイムアウトによって中断された場合、HSPIインタフェースでは、ポートの所有権を取り戻した後、できるだけ早く、中断前に実際にアクセスした最後のアドレスに続くアドレスとともに最初のコマンドシーケンスを再発行します。最初に設定された転送は継目なく進行し、終了します。

30.4.8 HSPI動作モードの概要

HSPIには、使用されるローレベルのプロトコル（レギュラコマンドまたはHyperBusのいずれか）に関係なく、次の動作モードがあります。

- インダイレクトモード（読出しまたは書込み）
- 自動ステータスポーリングモード（レギュラコマンドプロトコルでのみ）
- メモリマップドモード

30.4.9 HSPIインダイレクトモード

インダイレクトモードでは、HSPIレジスタに書き込むことでコマンドが開始され、データレジスタへの書込み／読出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

HSPI_CRにおいてFMODE[1:0] = 00の場合、HSPIはインダイレクト書込みモードです。このモードでは、データフェーズ中に外部デバイスへバイトが送信されます。データは、HSPI_DRへの書込みによって提供されます。

FMODE[1:0] = 01の場合、HSPIはインダイレクト読出しモードです。このモードでは、データフェーズ中に外部デバイスからバイトが受信されます。データは、HSPI_DRの読出しによって取得されます。

インダイレクトモードでは、DQSは無効の状態で、HSPIが8つのレーンにわたってDTRモードに設定されている場合、指定する開始アドレスとデータ長は偶数でなければなりません。

注： **開始アドレスが前回のインダイレクトアクセスの開始アドレスと同じであっても、HSPI_ARレジスタを更新する必要があります。**

読出し／書込みするバイト数はHSPI_DLRで指定されます。

- DL[31:0] = 0xFFFF FFFFの場合、データ長は不定長とみなされ、HSPIは単純に外部デバイスの最後（DEVSIZEによって定義される）に達するまでデータを転送し続けます。転送されるバイトが無い場合は、HSPI_CCRのDMODE[2:0]を0にセットする必要があります。
- DL[31:0] = 0xFFFFF FFFFでDEVSIZE[4:0] = 0x1F（最大値は4GBのデバイスを示す）の場合、転送は無期限に続けられ、アボートリクエスト後またはHSPIの無効化後にやっと停止します。最後のメモリアドレスが（アドレス0xFFFF FFFFで）読み出された後も、アドレス = 0x0000 0000で読出しが続けられます。

プログラムされた送信または受信バイト数に達すると、HSPI_SRにTCFビットがセットされ、HSPI_CRにおいてTCIE=1の場合に割込みが生成されます。データ数が定義されていない場合、外部SPIメモリの限界に達した時点で、HSPI_DCR1に定義されているデバイスのサイズに応じてTCFがセットされます。

レギュラコマンドプロトコルでの転送開始のトリガ

レギュラコマンドプロトコルを使用している場合、インダイレクトモードで転送を開始する方法は、HSPIの設定に応じて3通りあります。一般に、転送の開始はソフトウェアがコマンドに必要な最後の情報を提供するとすぐにトリガされます。もっと具体的に言えば、インダイレクトモードでは、以下のいずれかのイベントシーケンスが発生した場合に転送が開始されます。

- アドレスが不要（ADMODE[2:0] = 000）でソフトウェアからのデータ提供が不要（FMODE[1:0] = 01またはDMODE[2:0] = 000）な場合で、HSPI_IRのINSTRUCTION[31:0]に書込みが実行された時点
- アドレスが必要（ADMODE[2:0]≠000）でソフトウェアからのデータ提供が不要（FMODE[1:0] = 01またはDMODE[2:0] = 000）な場合で、HSPI_ARのADDRESS[31:0]に書込みが実行された時点
- ソフトウェアからのデータ提供が必要（FMODE[1:0]=00でDMODE[2:0]≠000）な場合で、HSPI_DRのDATA[31:0]に書込みが実行された時点

HSPI_ABRへの書込みが通信開始をトリガすることはありません。オルタネートバイトが必要な場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSYビットが自動的にHSPI_SRにセットされます。

HyperBusプロトコルでの転送開始のトリガ

HSPIの設定に応じて、インダイレクトモードでコマンドを開始するさまざまな方法があります。一般に、ファームウェアが転送を始めるのに必要な最後の情報を提供するとすぐに転送は開始されます。もっと具体的に言えば、以下のレジスタ設定のいずれかが実行する最後のものであった場合、それによってインダイレクトモードでの通信が開始されます。

- インダイレクト読出しモード（FMODE[1:0] = 01）で、ADMODE[2:0] ≠ 000でADDRESS[31:0]（HSPI_AR）への書込みを行ったとき。
- インダイレクト書込みモード（FMODE = 00の場合）で、DATA[31:0]（HSPI_DR）への書込みを行ったとき。
- インダイレクト読出しモード（ADMODE[2:0] = 000かつFMODE = 01の状態）に対して、INSTRUCTION[31:0]（HSPI_IR）への（ダミー）書込みを行ったとき。

転送が開始されたらすぐに、BUSYビット（HSPI_SR[5]）が自動的にセットされます。

FIFOおよびデータの管理

インダイレクトモードでは、データはHSPIに内蔵されている64バイトのFIFOを通ります。HSPI_SRのFLEVELは、現在FIFOに保持されているバイト数を示します。

インダイレクト書込みモード（FMODE[1:0] = 00）では、ソフトウェアはHSPI_DRに書き込む時点でFIFOにデータを追加します。書込みによりFIFOに追加されるバイト数は、ワード書込みで4バイト、ハーフワード書込みで2バイト、バイト書込みで1バイトだけです。ソフトウェアがFIFOに追加するバイト数が多すぎる（DL[31:0]で示される数より多い）場合、余分なバイトは、書込み動作の終了時（TCFがセットされるとき）にFIFOから一掃されます。

HSPI_DRへのバイト／ハーフワードアクセスは、32bitレジスタの最下位バイト／ハーフワードにのみ実行する必要があります。

FTHRESを使用してFIFO閾値を定義します。そのポイントに達するとFIFO閾値フラグFTFがセットされます。インダイレクト読出しモードでは、FIFOから読み出される有効バイト数が閾値を超えるとFTFがセットされます。また、FTHRESの設定にかかわらず、外部デバイスから最後のバイトが読み出された後にFIFO内にデータが残っている場合もFTFがセットされます。インダイレクト書込みモードでは、FIFO内の空のバイト数が閾値を超えるとFTFがセットされます。

FTIE = 1の場合、FTFがセットされた時点で割込みが発生します。DMAEN = 1の場合、FTFがセットされた時点でDMA転送が開始されます。閾値条件が真でなくなると（CPUまたはDMAによって十分なデータが転送された後）すぐに、ハードウェアによってFTFがクリアされます。

インダイレクト読出しモードでは、FIFOがフルになると、HSPIは一時的に外部デバイスからのバイトの読出しを停止して、オーバーランを回避します。外部デバイスの読出しは、FIFO内の4バイトが空になる（FLEVEL ≤ (FIFO_size-4)となるとき）まで再開されませんのでご注意ください。したがって、FTHRES ≥ (FIFO_size-3)の場合、HSPIが外部デバイスからのデータ取得を再開できるように、十分なバイトの読出しがアプリケーションによって確実に行われる必要があります。そうしないと、FLEVEL < FTHRESである限り、FTFフラグは0のままとなります。

次のラインに対する要求がない限り、RXFIFOに最後に読み出されたデータは有効なままです。つまり、アプリケーションが同じ場所で連続して何回か読み出す場合、データはRXFIFOから提供され、離れているメモリから再度読み出されることはありません。

30.4.10 HSPI自動ステータスポーリングモード

自動ステータスポーリングモードでは、HSPIは定期的にコマンドを開始させて、定義された数のステータスバイト（最大4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割込みを生成することができます。

自動ステータスポーリングモードはレギュラコマンドプロトコルでのみ使用する必要があります。HyperBusプロトコルの場合、HyperFlashメモリへの読出しステータスレジスタは2つのステップ（書込み操作とそれに続く読出し操作）で行う必要があるため、悪用することができません。

デバイスへのアクセスは、インダイレクト読出しモードと同じ方法で開始されます。この時点でHSPI_SRのBUSYがハイになり、周期的なアクセスが実行される間もハイのままです。

HSPI_PSMARのMASK[31:0]の内容は、自動ステータスポーリングモードで外部デバイスからのデータをマスクするために使用されます。

- MASK[n] = 0の場合、リザルトのビットnはマスクされ、考慮されません。
- MASK[n] = 1かつビット[n]の値がHSPI_PSMARのMATCH[n]の値と同じであれば、ビットnでの一致が検出されます。

HSPI_CRでPMM = 0の場合、AND一致モードがアクティブになります。つまり、SMFは、マスクされていないすべてのビットで一致が検出された場合にのみHSPI_SRにセットされます。

HSPI_CRでPMM = 1の場合、OR一致モードがアクティブになります。つまり、SMFは、マスクされていないいずれかのビットで一致が検出された場合にセットされます。

割込みは、SMIE = 1の場合にSMF = 1になるとき呼び出されます。

APMSがHSPI_CRでセットされている場合、一致が検出されるとすぐに動作は停止し、BUSYが0になります。そうでない場合、BUSYは1のままで、アボートが検出されるまで、またはHSPIが無効化される（EN = 0）まで周期的なアクセスが継続します。

HSPI_DRには、最後に受信したステータスバイトが含まれます（FIFOは無効化されます）。レジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。HSPI_SRのFTFは新たなステータスの読出しが完了するとすぐにセットされます。FTFは、データが読み出された直後にクリアされます。

自動ステータスポーリングモードでは、可変の遅延はサポートされていません。したがって、メモリは固定遅延で設定する必要があります。

30.4.11 HSPIメモリマップドモード

メモリマップドモードに設定された場合、外部SPIデバイスは内部メモリとみなされます。

注： **外部デバイスの容量がもっと大きい場合でも、アドレス指定できるのは256MBまでです。**

DEVSIZE[4:0]で規定されている範囲外のアドレスにアクセスすると、256MBの範囲内であっても、AHBエラーが発生します。このエラーの影響は、アクセスを試みたAHBマスタに依存します。

- Cortex CPUの場合は、ハードフォールト割込みが発生します。
- DMAの場合は、DMA転送エラーが発生し、対応するDMAチャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。これにより、HSPIは最も最近にアクセスしたアドレスに続くアドレスに継続してバイトをロードします。続くバイトに次のアクセスが連続している場合、それらの動作結果はプリフェッチされているので、その動作は迅速に完了します。

デフォルトでは、HSPIはプリフェッチ動作を停止することはなく、外部デバイスへのアクセスが長時間なくても、NCSをローに保持したまま前の読出し動作をアクティブな状態に保つか、または新たな転送を再起動します。

NCSがローのときに外部デバイスはより多く電力を消費する傾向があるため、アプリケーションがタイムアウトカウンタ（HSPI_CRにおいてTCEN = 1）の起動を要求する場合があります。クロックが非アクティブになってからアクセスのない状態で x サイクルが経過すると、HSPI_LPTR の TIMEOUT[15:0]で定義された期間の後、NCSがリリースされます。

最初のメモリマップドアクセスが発生するとすぐに、BUSYはハイになります。プリフェッチ動作の影響で、BUSYの立下がりは、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、お勧めできません。インダイレクト書込みモードがこの動作を実現します。

30.4.12 HSPIの設定の概要

HSPIの設定は次の3つのステップで行われます。

1. HSPIシステムの設定
2. HSPIデバイスの設定
3. HSPIモードの設定

30.4.13 HSPIシステムの設定

HSPIはHSPI_CRを使用して設定します。ユーザは下記をプログラムする必要があります。

- FMODE[1:0]による機能モード
- 自動ステータスポーリングモードの動作（PMMおよびAPMSで必要な場合）
- FTHRESによるFIFOレベル
- DMAENによるDMAの使用
- TCENによるタイムアウトカウンタの使用
- デュアルメモリ構成（必要な場合、DMMによって）

割込みを使用する場合、それぞれのイネーブルビットもこのフェーズ中にセットすることができます。

タイムアウトカウンタを使用する場合、タイムアウト値をHSPI_LPTRにプログラムします。

DMAチャネルはHSPIの設定中に有効にしてはなりません。予期しないリクエストの生成を避けるため、動作を完全に設定した後にのみ有効にする必要があります。

DMAとHSPIはデータ長に関して整合のとれた方法で設定しなければなりません。すなわち、FTHRES値はDMAバーストサイズを反映する必要があります。

30.4.14 HSPIデバイスの設定

対象の外部デバイスに関連するパラメータはHSPI_DCR1とHSPI_DCR2によって設定されます。ユーザは下記をプログラムする必要があります。

- DEVSIZE[4:0]によるデバイスサイズ
- CSHT[5:0]によるチップセレクト最小ハイ時間
- PRESCALER[7:0]によるデバイス周波数

MTYP[2:0]では、8ラインモードで使用されるメモリタイプを指定します。

- 8データビットモード（DMODE[2:0] = 100）でD0/D1順序のMicronモード
- 8データビットモード（DMODE[2:0] = 100）でD1/D0順序のMacronixモード。Macronixのメモリには2つの異なる構成があります。
  MTYP[2:0] = 001はOctaFlashメモリを対象とし、MTYP[2:0] = 011は特定のアドレスフェーズを持つOctaRAM™メモリを対象とします（アドレスはMacronix要件に合うように行と列で構築します）。
- HyperBusメモリモード：プロトコルはHyperBus仕様に準拠します。
- レジスタ空間をアドレス指定するHyperBusレジスタモード：このモードでのメモリマップドアクセスをキャッシュ不可能にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

APmemoryからのメモリについては、メモリ･プロバイダの要件（アドレス･ビット管理）に合うようにMTYP[2:0]=6を選択します。

DEVSIZE[4:0]は、次の式を使用して外部メモリのサイズを定義します。

$$\text{デバイス内のバイト数} = 2^{[\text{DEVSIZE}+1]}$$

ここで、DEVSIZE+1は外部デバイスのアドレスを指定するために必要なアドレスビット数です。外部デバイスの容量はインダイレクトモードでは最大4GB（32bitを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は256MBに制限されます。

DMM = 1の場合、DEVSIZE[4:0]は2つのデバイスの総容量を示します。

HSPIが2つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは2つのコマンドの間に少なくとも1CLKサイクルの間NCSをハイに立ち上げます。

外部デバイスがコマンド間により長い時間を必要とする場合、CSHT[5:0]を使用して、NCSがハイに保たれなければならない最小のCLKサイクル数（最大64）を指定することができます。

CKMODEは、コマンド間でのCLKのレベルを示します（NCS = 1のとき）。

HyperBusプロトコルでは、デバイスのタイミング（t_{ACC}とt_{RWR}）および遅延モードをHSPI_HLCRに設定する必要があります。

30.4.15 HSPIレギュラコマンドモードの設定

インダイレクトモードの設定

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0] = 01の場合、インダイレクト読出しモードが選択され、データを外部デバイスから読み出すことができます。

HSPIがインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. HSPI_DLRに読み出される／書き込まれるデータのバイト数を指定します。
2. HSPI_TCRに、フレームのタイミングを指定します。
3. HSPI_CCRに、フレームフォーマットを指定します。
4. HSPI_IRに、命令を指定します。
5. HSPI_ABRに、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
6. HSPI_ARに、ターゲットアドレスを指定します。
7. 必要に応じてDMAチャネルを有効にします。
8. HSPI_DRによりFIFOに対してデータの読出し／書込みを行います（DMAを使用しない場合）。

アドレスレジスタ（HSPI_AR）とデータレジスタ（HSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、HSPI_IRが書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE[2:0]とDMODE[2:0]がどちらも000である場合、またはインダイレクト読出しモード（FMODE[1:0] = 01）でADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）で、データレジスタに書き込む必要がない場合（FMODE[1:0] = 01またはDMODE[2:0] = 000）、HSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE[1:0] = 00かつDMODE[2:0] ≠ 000）、HSPI_DRによるFIFOへの書込みによって通信の開始がトリガされます。

自動ステータスポーリングモードの設定

自動ステータスポーリングモードは、FMODE[1:0] = 10にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は4バイトです。HSPI_DLRでもっと大量のデータが要求されても、そのリクエストは無視され、4バイトだけが読み出されます。HSPI_PIRでは周期が指定されます。

ステータスデータが取得されると、以下を処理できます。

- SMFをセットする（有効になっていれば割込みを生成する）。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値はHSPI_PSMKRに格納されている値でマスクし、HSPI_PSMARに格納されている値で論理和または論理積をとることができます。

一致する場合、SMFがセットされ、有効になっていれば割込みが生成されます。AMPSがセットされている場合、HSPIを自動的に停止することができます。いずれにしろ、一番最後に取得した値はHSPI_DRで使用可能です。

HSPIが自動ステータスポーリングモードで使用される場合、フレームは次のように構成されます。

1. HSPI_PSMKRに、入力マスクを指定します。
2. HSPI_PSMARに、比較値を指定します。
3. HSPI_PIRに、読出し周期を指定します。
4. HSPI_DLRに、読み出されるデータのバイト数を指定します。
5. HSPI_TCRに、フレームのタイミングを指定します。
6. HSPI_CCRに、フレームフォーマットを指定します。
7. HSPI_IRに、命令を指定します。
8. HSPI_ABRに、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
9. HSPI_ARに、オプションのターゲットアドレスを指定します。

アドレスレジスタ（HSPI_AR）が特定のコマンドに関して更新する必要がない場合、HSPI_CCRが書き込まれるとすぐにコマンドシーケンスが開始されます。これはADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）、HSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

メモリマップドモードの設定

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、消去やプログラム·ステータスの内部フラグをポーリングする必要があるため、お勧めできません。インダイレクト書込みモードでは、場合によっては自動ステータスポーリングモードと組合せて、この動作を実現します。

メモリマップドモードに移行するには、HSPI_CRにおいてFMODE[1:0] = 11にセットします。

プログラムされた命令およびフレームは、AHBマスタがメモリマップド空間にアクセスするときに送信されます。

FIFOは、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでのHSPI_DRへのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（HSPI_DLR）は意味を持ちません。

HSPIがメモリマップドモードで使用される場合、フレームは次のように構成されます。

1. 読出し動作に対して、HSPI_TCRにフレームのタイミングを指定します。
2. 読出し動作に対して、HSPI_CCRにフレームフォーマットを指定します。
3. HSPI_IRに、命令を指定します。
4. 読出し動作に対して、HSPI_ABR にアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。
5. 書込み動作に対して、HSPI_WTCRにフレームのタイミングを指定します。
6. 書込み動作に対して、HSPI_WCCRにフレームフォーマットを指定します。
7. HSPI_WIRに命令を指定します。
8. 書込み動作に対して、HSPI_WABRにアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。

すべての設定操作は、メモリ領域への最初のアクセスの前に完了する必要があります（BUSY = 0をチェックすることで保証されます）。BUSY = 1のときのレジスタ書込み操作は効果がなく、エラー応答は通知されません。最初のアクセスでHSPIはビジーになり、それ以上の設定は許可されません。この場合、BUSYをローにする唯一の方法は、ENABLEビットをクリアするか、ABORTビットをセットして中止することです。

DQSを使用しない場合のHSPI遅延データサンプリング

デフォルトでは、DQSを使用しない場合、外部デバイスに駆動されているデータをHSPIがサンプリングするタイミングは、外部デバイスが信号を駆動してから1/2CLKサイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。HSPI_TCRのSSHIFTを使用することで、データのサンプリングを1/2CLKサイクルだけシフトすることができます。

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ファームウェアはSSHIFTをクリアする必要があります。

DQSを使用する場合のHSPI遅延データサンプリング

外部DQSがサンプリング･クロックとして使用される場合、freq_minを超えるすべての周波数に対して、SPIクロック･サイクルの1/4だけ正確にシフトされ、製品が「ハイスピードインタフェース」を内蔵している場合のデータ伝搬遅延を補償します。

30.4.16 HSPIHyperBusプロトコルの設定

インダイレクトモードの設定（HyperBus）

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0] = 01の場合、データを外部デバイスから読み出すことができるインダイレクト読出しモードが選択されます。ADMODEは000と異なる値（たとえば、ADMODE = 100など）で設定する必要があります。

HSPIがインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. HSPI_DLRに読み出される／書き込まれるデータのバイト数を指定します。
2. HSPI_ARに、ターゲットアドレスを指定します。
3. 必要に応じてDMAチャネルを有効にします。
4. HSPI_DRによりFIFOに対してデータの読出し／書込みを行います（DMAを使用しない場合）。

インダイレクト読出しモードでは､HSPI_ARが書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

インダイレクト書込みモードでは、HSPI_DRによるFIFOへの書込みによって通信の開始がトリガされます。

メモリマップドモードの設定（HyperBus）

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップド書込みを使用してFlashメモリをプログラムすることは、お勧めできません。インダイレクト書込みモードがこの動作を実現します。

メモリマップドモードに移行するには、FMODE[1:0] = 11にセットします。プログラムされた命令およびフレームは、AHBマスタがメモリマップド空間にアクセスしているときに送信されます。

FIFOは、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでのHSPI_DRへのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（HSPI_DLR）は意味を持ちません。

メモリ領域へ最初にアクセスする前に、すべての設定操作を完了する必要があります。最初のアクセスでHSPIはビジーになり、いかなる設定も許可されません。この場合、BUSYをローにする唯一の方法は、ENABLEビットをクリアするか、ABORTビットをセットして中止することです。

30.4.17 HSPI エラー管理

次の場合に、エラーが生成される可能性があります。

- インダイレクトモードまたは自動ステータスポーリングモードで、HSPI_AR に間違ったアドレスがプログラムされている場合（DEVSIZE[4:0]で定義されているデバイスのサイズに応じて）。この場合、TEFがセットされ、有効になっていれば割込みが生成されます。
- インダイレクトモードでは､アドレスにデータ長を加えた値がデバイスのサイズを超えている場合。アクセスがトリガされるとすぐにTEFがセットされます。
- メモリマップドモードでは、AHB マスタによる範囲外のアクセスが行われた場合、誤った AHB リクエストへの応答としてAHBエラーが生成されます。
- メモリマップドモードが無効になっているときに、メモリマップド領域にアクセスすると、誤ったAHBリクエストへの応答としてAHBエラーが生成されます。

HSPIは、次の状況でAHBスレーブエラーを発生します。

- メモリマップドモードが無効になっていて、AHB読出しリクエストが発生した場合。
- 読出しまたは書込みアドレスが、外部メモリのサイズを超過した場合。
- 読出しまたは書込みバーストの進行中に、アボートを受信した場合。
- 読出しまたは書込みバーストの進行中に、HSPIが無効になった場合。
- 書込みラップバーストを受信した場合。
- オクタルDTRモードまたはデュアルメモリ構成のHSPI_WCCRでDQSE = 0の間に書込みリクエストが受信された場合。
- MTYP[2:0]がHyperBusの場合を除き、DMODE[2:0] = 000（データフェーズなし）のときに書込みリクエストを受信した場合。
- ラップ読出しバースト時にアクセスサイズが不正である場合。これは、HSIZEが4バイトとは異なることを意味します（メモリマップドモードの場合のみ）。
- 48 バイトと異なるサイズの読出しラップバーストを受信した時のラップサイズが不正である場合（メモリマップドモードの場合のみ）。

30.4.18 HSPIハイスピードインタフェースと較正

より高い周波数に到達するために、HSPI（または製品がI/Oマネージャを内蔵している場合はI/Oマネージャ）とI/Oパッドの間に専用のハイスピードインタフェースが挿入されています。

以下の説明は、すべてのデータバス･サイズ1、2、4、8、16bitについて有効です。

ハイスピードインタフェース･ブロックには、同じくハイスピードインタフェースに配置されているDLL（遅延ロックループ）から作られる遅延クロックによってクロック供給される再同期レジスタが組み込まれています。ハイスピードインタフェース機能は、HSPIにあるレジスタによって制御されます。

再同期の目的は、主にデータまたはデータストローブを、正しいタイミング精度で、オクタルバスのクロック周期の1/4だけシフトさせることです。DLLはこのクロック周期に対して較正される必要があります。

較正プロセスは、以下の3つの条件のいずれかが満たされると自動的に有効になります。

- HSPIがリセット状態を終了する。
- HSPI_DCR2のPRESCALER[7:0]に値が書き込まれる。
- HSPI_CCRに値が書き込まれる。

較正プロセスは、以下の2つの条件が**両方とも**満たされると開始されます。

- 較正が、上記3条件のいずれかによって有効になっている。
- BUSY = 1をセットする動作が実行される。たとえば、較正が有効になった後の最初のメモリへの転送。較正が完了すると、BUSYは0に戻ります。

定期的な再較正が必要な場合（たとえば、温度や電源の長期的な変化を考慮するため）、この再較正は、BUSY = 0のときにHSPI_DCR2のPRESCALER[7:0]に定期的に書き込むことによってトリガする必要があります。

較正が完了すると、単位遅延の数で表されるSPIバス･クロック周期の値が、HSPI_CALFCRのCOARSE[5:0]およびFINE[6:0]で、利用可能になります。

自動較正の後、HSPI_CALSORとHSPI_CALSIRには、1/4サイクルの遅延に対応する同じ値が自動的にロードされます。

メモリがDQSをサポートしていない場合（DQSE = 0）、受信時に自動較正は行われません。フィードバック･クロック（HSPI_CALMR）を遅延させるために、代わりにDLLマスタが使用されます。この遅延は、正しい読出し動作を保証するために、最適な遅延を決定するソフトウェア･シーケンスを使用して、アプリケーション自体で調整する必要があります。

DTR送信モードでクロックが分周される場合、DQS/データの1/4サイクル遅延はDLL自体によって挿入されるのではなく、内部フロップ設計スキームによって挿入されます。SDR送信モードでは、クロック･プリスケーラの値に関係なく、DLLは使用されません。

DTRモードでプリスケーラがバイパスされている場合（PRESCALER[7:0] = 0）、インタフェースに提供されるカーネルクロックのデューティ･サイクルは50%でなければなりません。

ハイスピードインタフェースを使用する場合、システムクロック（AHBクロック）は少なくともSPIクロックと同じ速度である必要があります。

30.4.19 HSPIのBUSYおよびABORT

HSPIが外部デバイスでの動作を開始すると、BUSYがHSPI_SRに自動的にセットされます。

インダイレクトモードでは、HSPIが要求されたコマンドシーケンスを完了し、FIFOが空になると、BUSYがリセットされます。

自動ステータスポーリングモードでは、一致（APMS = 1の場合）またはアボートの検出により、最後の周期的なアクセスが完了して初めてBUSYビットはローになります。

メモリマップドモードでの最初のアクセス後、アボートの検出でのみ、BUSYはローになります。

HSPI_CRのABORTをセットすることによって、いかなる動作も中止することができます。アボートが完了すると、BUSYおよびABORTは自動的にリセットされ、FIFOが一掃されます。

ABORTを設定する前に、ソフトウェアは同期バリアを使用して現在のすべてのトランザクションが終了していることを確認する必要があります。HSPI_DRでデータの読取りまたは書込み動作を処理するためにDMAが有効になっている場合は、HSPIを中止する前にDMAチャネルを無効にすることをお勧めします。

注： **デバイスには、ステータスレジスタへの書込み動作が中止された場合に誤動作するものもあります。**

30.4.20 HSPIの再設定または無効化

HSPIの再設定の前に、ソフトウェアはすべてのトランザクションが完了していることを確認する必要があります。

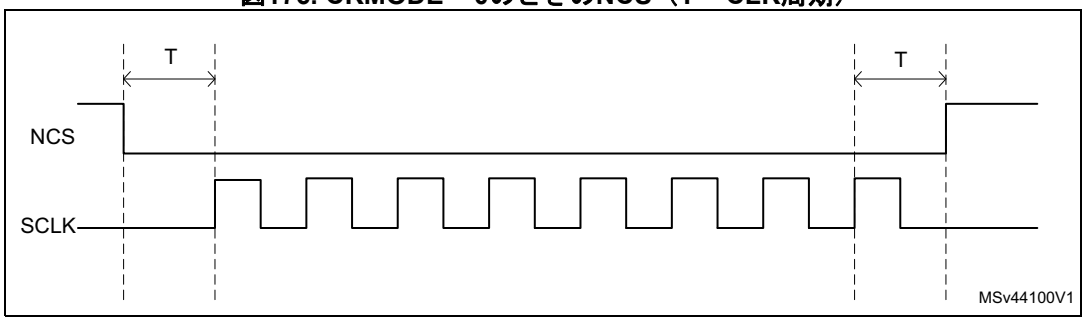
- メモリマップド書込みの後、ソフトウェアはダミーリードに続いて同期バリア、その後にABORTを実行する必要があります。
- メモリマップド読出しの後、ソフトウェアは同期バリア、その後にABORTを実行する必要があります。

30.4.21 NCSの動作

デフォルトではNCSハイで、外部デバイスは選択解除されています。NCSの立下がりは動作が始まる前に、立上がりは動作終了の直後に起こります。

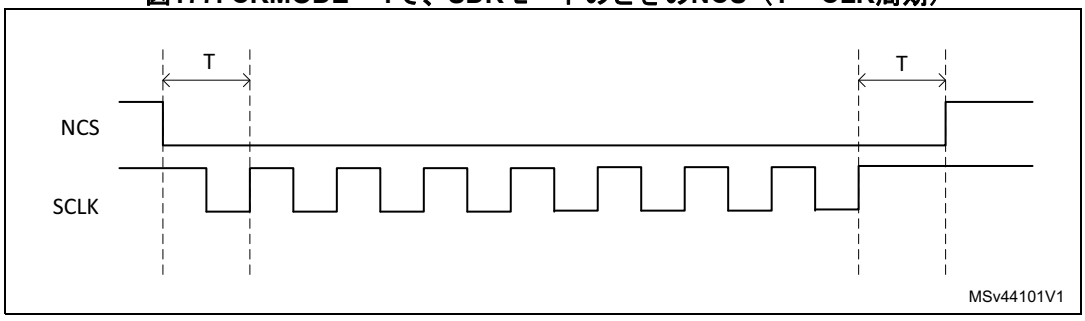
CKMODE = 0（クロックモード0 : 進行中の動作がない場合はCLKがローに保持される）のとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。

図176. CKMODE = 0のときのNCS（T = CLK周期）



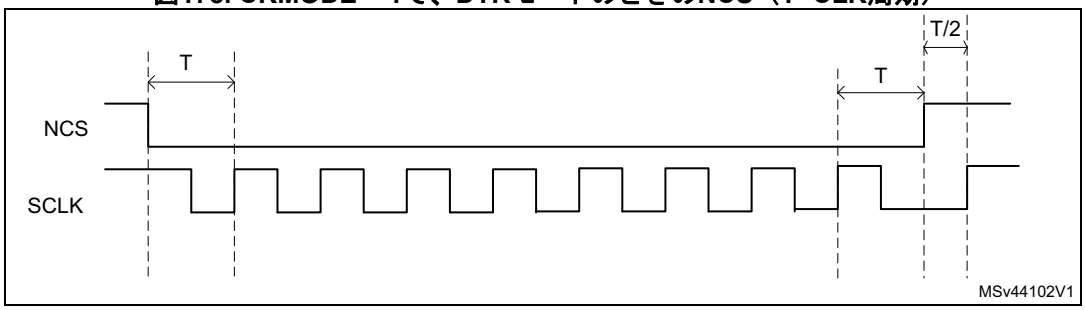
CKMODE = 1（クロックモード3 : 進行中の動作がない場合はCLKがハイになる）、かつSDRモードのとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。

図177. CKMODE = 1で、SDRモードのときのNCS（T = CLK周期）



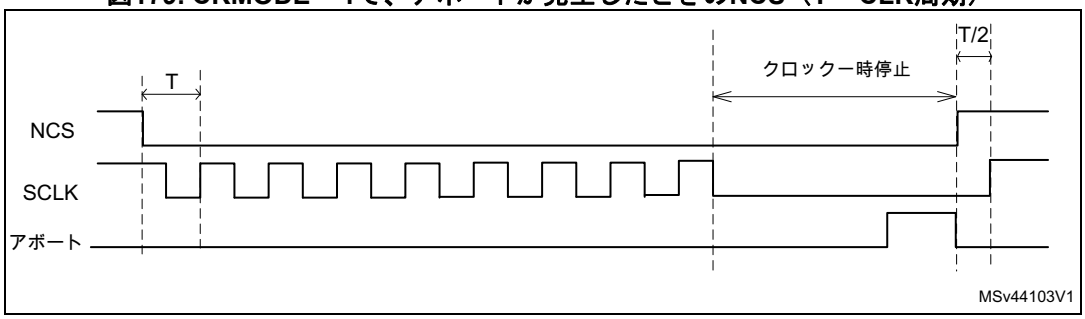
CKMODE = 1（クロックモード3）、かつDDTR = 1（データDTRモード）のとき、NCSの立下がりは動作の最初のCLKエッジ立上がりの1CLKサイクル前に、NCSの立上がりは動作の最後のアクティブCLKエッジ立上がりから1CLKサイクル後に起こります（下図参照）。DTR動作は立下がりエッジで終わらなければならないため、NCSが立ち上がるときCLKはローであり、その後CLKは1/2CLKサイクル後に立ち上がり直します。

図178. CKMODE = 1で、DTRモードのときのNCS（T=CLK周期）



読出し動作中でFIFOがフルのまま、または書込み動作中でFIFOが空のままであれば、ソフトウェアがFIFOを処理するまでその動作は停止し、CLKはローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後にNCSが立ち上がり、CLKは1/2CLKサイクル後に立ち上がります（下図参照）。

図179. CKMODE = 1で、アボートが発生したときのNCS（T = CLK周期）



30.5 アドレスのアライメントとデータ数

以下の表は、使用例に応じたアドレスアラインメントとプログラムされたデータ数への影響をまとめたものです。

表264.アドレスアライメントのケース

メモリタイプ	トランザクションタイプ	アドレスの制約(1)	アドレスの制約不遵守の場合の影響	バイト数の制約 (1)	バイト数の制約不遵守の場合の影響
シングル、デュアル、クワッドFlashまたはSRAM（DMM = 0）	IND(2) 読出し	なし	なし	なし	なし
	MM(3) の読出し				
	IND書込み				
	MM書込み				
シングル、デュアル、クワッドFlashまたはSRAM（DMM = 1）	INDの読出し	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MM書込み	偶数	スレーブエラー	偶数	最後のバイトが失われます。

表264.アドレスアライメントのケース （続き）

メモリタイプ	トランザクションタイプ	アドレスの制約(1)	アドレスの制約不遵守の場合の影響	バイト数の制約 (1)	バイト数の制約不遵守の場合の影響
SDRモードのオクタルFlash	INDの読出し	なし	なし	なし	なし
	MMの読出し				
	IND書込み				
	MM書込み				
WDM(6) なしのDTRモードのオクタルメモリ、またはWDMなしのSDRモードの16bitメモリ	INDの読出し	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MM書込み	偶数	スレーブエラー	偶数	最後のバイトが失われます。
WDMなしのDTRモードの16bitまたはデュアルオクタル･メモリ	IND読出し32bit	境界調整	ADDR[1:0]は00とみなされます。(4)	N×4	DLR[1:0] は11 とみなされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み	境界調整	ADDR[1:0]は00とみなされます。(4)	N×4	DLR[1:0] は11 とみなされます。(5)
	MM書込み	境界調整	スレーブエラー	N×4	最後のバイトが失われます。
DTRモードでWDMありのオクタルFlashまたはRAM	INDの読出し	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				
HyperBus	INDの読出し	偶数	ADDR[0] は0 にセットされます。(4)	偶数	DLR[0]は1にセットされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				
WDMありの16bitデータバス･メモリまたはデュアルオクタル･メモリ	IND読出し32bit	境界調整	ADDR[1:0]は00とみなされます。(4)	N*4	DLR[1:0] は11 とみなされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				
16bit HyperBus	IND読出し32bit	境界調整	ADDR[1:0]は00とみなされます。(4)	N*4	DLR[1:0] は11 とみなされます。(5)
	MMの読出し	なし	なし	なし	なし
	IND書込み				
	MM書込み				

1. ソフトウェアで遵守すること。
2. IND = インダイレクトモード
3. MM = メモリマップドモード
4. 転送開始時の追加データ
5. 転送終了時の追加データ
6. WDM = 書込みデータマスク

30.6 HSPI割込み

割込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO閾値
- 転送完了
- 転送エラー

より高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表265.HSPI割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

30.7 HSPI レジスタ

30.7.1 HSPI制御レジスタ（HSPI_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSEL[1:0]		FMODE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
rw	rw	rw	rw					rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FTHRES[5:0]						Res.	DMM	Res.	Res.	TCEN	DMAEN	ABORT	EN
		rw	rw	rw	rw	rw	rw		rw			rw	rw	rw	rw

ビット31:30 **MSEL[1:0]**：Flash選択

これらのビットは、シングルメモリ構成（DMM = 0）においてsingle-、dual-、quad-、またはoctal-SPIモードでアドレス指定するメモリを選択します。

- quad-SPIモードの場合：

00：データはIO[3:0]を経由して交換されます。

01：データはIO[7:4]を経由して交換されます。

10：データはIO[11:8]を経由して交換されます。

11：データはIO[15:12]を経由して交換されます。

- octal-SPIモードまたはデュアルクワッド構成の場合：

0x：データはIO[7:0]を経由して交換されます。

1x：データはIO[15:8]を経由して交換されます。

これらのビットは、デュアルオクタル構成（データは8bitでDMM = 1）または16bit構成（データはIO[15:0]経由で交換される）の場合には無視されます。

注： ビット30はビット7に反映されます。このビットフィールドは、BUSYが0のときにのみ修正できます。

ビット29:28 **FMODE[1:0]**：機能モード

このビットフィールドは、HSPI機能の動作モードを定義します。

00：インダイレクト書込みモード

01：インダイレクト読出しモード

10：自動ステータスポーリングモード（レギュラコマンドプロトコルにのみ該当）

11：メモリマップドモード

すでにDMAEN = 1の場合、対応するチャネル用のDMAコントローラはFMODE[1:0]値を変更する前に無効にする必要があります。DMAEN = 1のときにFMODE[1:0]とFTHRES[4:0]が誤って更新されると、DMAリクエスト信号は自動的に非アクティブ状態になります。

注： このビットフィールドは、BUSYが0のときにのみ修正できます。

ビット27:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **PMM**：ポーリング一致モード

このビットは、自動ステータスポーリングモードの時に「一致」を判断するために使用する必要のある方法を示します。

0：AND一致モード。SMFは、デバイスから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1：OR一致モード。SMFは、デバイスから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット22 **APMS**：自動ステータスポーリングモード停止

このビットは、一致後に自動ステータスポーリングモードを停止するかどうかを決定します。

0：自動ステータスポーリングモードは、アボートまたはHSPIの無効化によってのみ停止されます。

1：自動ステータスポーリングモードは、一致が検出されるとすぐに停止されます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 **TOIE**：タイムアウト割込みイネーブル

このビットは、タイムアウト割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット19 **SMIE**：ステータス一致割込みイネーブル

このビットは、ステータス一致割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット18 **FTIE**：FIFO閾値割込みイネーブル

このビットは、FIFO閾値割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット17 **TCIE**：転送完了割込み有効化

このビットは、転送完了割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット16 **TEIE**：転送エラー割込みの有効化

このビットは、転送エラー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:8 **FTHRES[5:0]**：FIFO閾値レベル

このビットフィールドでは、インダイレクトモードで、HSPI_SRのFIFO閾値フラグFTFのセットを引き起こすFIFO内のバイトの閾値数を定義します。

000000：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが1バイト以上ある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが1バイト以上ある場合にセットされます。

000001：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが2バイト以上ある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが2バイト以上ある場合にセットされます。

...

111111：FTFは、インダイレクト書込みモードで、FIFO内に書き込める空きバイトが64バイトある場合、またはインダイレクト読出しモードで、FIFOから読み出せる有効なバイトが64バイトある場合にセットされます。

注： DMAEN = 1の場合、対応するチャネル用のDMAコントローラはFTHRES[5:0]値を変更する前に無効にする必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **DMM**：デュアルメモリ構成

このビットはデュアルメモリ構成をアクティブにします。このモードでは、2つの外部デバイスを同時に使用して、スループットおよび容量を2倍にします。

0：デュアルメモリ構成は無効です。

1：デュアルメモリ構成は有効です。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット5:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **TCEN**：タイムアウトカウンタ･イネーブル

このビットは、メモリマップドモードが選択されている（FMODE[1:0] = 11）場合にのみ有効です。このビットは、タイムアウトカウンタを有効にします。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（NCS）はメモリマップドモードにおける1回のアクセス後は無限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、メモリマップドモードで外部デバイスが非アクティブになってからTIMEOUT[15:0]サイクル後に解除されます。

注： このビットは、BUSYが0のときにのみ修正できます。

ビット2 **DMAEN**:DMAイネーブル

インダイレクトモードでは、DMAを使用して、HSPI_DR経由でデータを入出力することができます。DMA転送は、FTFがセットされた時点で開始されます。

0：インダイレクトモードでDMAは無効です。

1：インダイレクトモードでDMAは有効です。

注： DMA 転送が進行中に DMAEN ビットをリセットすると、DMA とのハンドシェイクが切れます。DMAの動作中に、このビットを書き込まないでください。

ビット1 **ABORT**：アボートリクエスト

このビットは進行中のコマンドシーケンスを中止します。アボートが完了すると自動的にリセットされます。このビットは現在の転送を中止させます。

0：アボートは要求されていません。

1：アボートが要求されています。

注： このビットは常に0として読み出されます。

ビット0 **EN**：有効化

このビットは、HSPIを有効にします。

0：HSPI無効

1：HSPI有効

注： 動作中にこのENビットがクリアされた場合、ACKを受信せずにDMAリクエストを中止できます。DMA転送中にこのビットが0にセットされた場合、DMAへのREQ信号は、DMAからのACK信号がアクティブになるのを待たずに非アクティブ状態に戻ります。

30.7.2 HSPIデバイス設定レジスタ1（HSPI_DCR1）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	MTYP[2:0]			Res.	Res.	Res.	DEVSIZE[4:0]				
					rw	rw	rw				rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CSHT[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	FRCK	CKMODE
		rw	rw	rw	rw	rw	rw							rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **MTYP[2:0]**：メモリタイプ

このビットフィールドは、サポートするメモリのタイプを示します。

000：DTR8データビットモードでD0/D1順序のMicronモード。single-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。このモードでは、DQS信号極性がメモリクロック信号に対して反転されます。これはデフォルト値であり、Micron以外のメモリのMTYP[2:0]を変更する際は注意する必要があります。

001：DTR8データビットモードでD1/D0順序のMacronixモード。single-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。

010：標準モード

011：DTR8データビットモードでD1/D0順序のMacronixRAMモード。アドレスマッピング専用のsingle-、dual-、quad-、octal-SPIモードのレギュラコマンドプロトコル。

100：HyperBusメモリモード。プロトコルはHyperBus仕様に準拠します。

101：HyperBusレジスタモード。レジスタ空間にアドレス指定。このモードでメモリマップドアクセスを行う場合、キャッシュ不可能にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

110：APmemoryモード。DMODE = 101の場合、プロバイダの要件に適合するように、アドレスワードのビット10以上に対して特別なハードウェア操作が行われます（アドレス･ビット10から左にシフト操作し、この最後のビットを0に保ちます）。

その他：予約済み

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **DEVSIZE[4:0]**：デバイスサイズ

このビットフィールドは、次の式を使用して外部デバイスのサイズを定義します。

デバイス内のバイト数 = $2^{[DEVSIZE+1]}$。

DEVSIZE+1は外部デバイスのアドレスを指定するために必要とされる有効なアドレスビット数です。

デバイスの容量はインダイレクトモードでは最大4Gバイト（32bitを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は256Mバイトに制限されます。

レギュラコマンドプロトコルでは、DMM = 1の場合、DEVSIZE[4:0]は2つのデバイスの総容量を示します。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:8 **CSHT[5:0]**：チップセレクトハイ時間

CSHT + 1は、外部デバイスに対して発行されるコマンド間でチップセレクト（NCS）がハイに保たれなければならないCLKサイクルの最小数を定義します。

0x0：NCSは、外部デバイスコマンド間で少なくとも1サイクルの間はハイに保持されます。

0x1：NCSは、外部デバイスコマンド間で少なくとも2サイクルの間はハイに保持されます。

...

0x3F：NCSは、外部デバイスコマンド間で少なくとも64サイクルの間はハイに保持されます。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **FRCK**：フリー･ランニング･クロック

このビットによってフリー･ランニング･クロックを設定します。

0：CLKはフリーランニングではありません。

1：CLKはフリーランニングです（常に提供される）。

ビット0 **CKMODE**：クロックモード0/モード3

このビットは、コマンド間でCLKが取るレベルを示します（NCS = 1のとき）。

0：CLKはNCSがハイである（チップセレクトが解除されている）間はローに保持される必要があります。クロックモード0と呼ばれます。

1：CLKはNCSがハイである（チップセレクトが解除されている）間はハイに保持される必要があります。クロックモード3と呼ばれます。

30.7.3 HSPIデバイス設定レジスタ2（HSPI_DCR2）

アドレスオフセット：0x00C

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **WRAPSIZE[2:0]**：ラップサイズ

このビットフィールドは、メモリが設定される際のラップサイズを示します。ラップされた命令用の個別のコマンドを備えるメモリに対して、このビットフィールドは、HSPI_WPIRに保持されたコマンドに関連付けられたラップサイズを示します。

000：ラップド読出しはこのメモリでサポートされていません。

010：外部メモリは16バイトのラップサイズをサポートしています。

011：外部メモリは32バイトのラップサイズをサポートしています。

100：外部メモリは64バイトのラップサイズをサポートしています。

101：外部メモリは128バイトのラップサイズをサポートしています。

その他：予約済み

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PRESCALER[7:0]**：クロックプリスケーラ

このビットフィールドは、カーネルクロックに基づいてCLKを生成するスケーラ係数（値+1）を定義します。

0：$F_{CLK}=F_{KERNEL}$、カーネルクロックが直接HSPICLKとして使用されます（プリスケーラはバイパス）。この場合、DTRモードを使用する場合は、デューティサイクル50%のカーネルクロックをHSPIに提供することが必須です。

1：$F_{CLK} = F_{KERNEL}/2$

2：$F_{CLK} = F_{KERNEL}/3$

...

255：$F_{CLK} = F_{KERNEL}/256$

クロック分周比が奇数の場合、CLKのデューティサイクルは50%ではありません。クロック信号は、ハイに保持された時間より1サイクル分長く、ローに保持されます。

このビットフィールドに書き込むと、HSPI_CALOSRまたはHSPI_CALISRがその間に書き込まれた場合を除き、次の転送の開始時にハイスピードインタフェースDLLの新たな較正が自動的に開始されます。較正の実行中はずっと、BUSYはハイのままです。

30.7.4 HSPIデバイス設定レジスタ3（HSPI_DCR3）

アドレスオフセット：0x010

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBOUND[4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **CSBOUND[4:0]**：NCS境界

このビットフィールドは、トランザクション境界機能を有効にします。有効な場合、最小値の3が推奨されます。

$2^{CSBOUND}$バイトの各境界でNCSが解除されます。

0：NCS境界は無効です。

その他：NCS境界は、$2^{CSBOUND}$バイトに設定されます。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

30.7.5 HSPIデバイス設定レジスタ4（HSPI_DCR4）

アドレスオフセット：0x014

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REFRESH[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REFRESH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REFRESH[31:0]**：リフレッシュレート

このビットフィールドは、リフレッシュレート機能を有効にします。NCSは、書込みではREFRESH + 1クロックサイクルごと、および読出しではREFRESH + 4クロックサイクルごとに解除されます。single-、dual-、またはquad-SPIモードで1つのバイト送信中にリフレッシュが発生した場合、そのバイト送信を完了させる必要があるため、これらの2つの値は数クロックサイクル拡張される可能性があります。

0：リフレッシュは無効です。

その他：最大通信長はREFRESH + 1クロックサイクルに設定されます。

30.7.6 HSPIステータスレジスタ（HSPI_SR）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FLEVEL[6:0]							Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **FLEVEL[6:0]**：FIFOレベル

このビットフィールドはFIFO内に保持されている有効バイト数を示します。FIFOが空の場合、FLEVEL = 0で、フルになっている場合、64です。自動ステータスポーリングモードでは、FLEVELはゼロです。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。これは、外部デバイスの動作が終わり、FIFOが空になると自動的にクリアされます。

ビット4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOFに1を書き込むことによってクリアされます。

ビット3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（HSPI_PSMAR）の対応するビットと一致している場合に、自動ステータスポーリングモードでセットされます。
CSMFに1を書き込むことによってクリアされます。

ビット2 **FTF**：FIFO閾値フラグ

インダイレクトモードでは、このビットはFIFO閾値に達した時点で、または外部デバイスからの読出し完了後にFIFO内にデータが残っている場合にセットされます。
閾値条件が真でなくなるとすぐに自動的にクリアされます。
自動ステータスポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット1 **TCF**：転送完了フラグ

このビットは、プログラムされたデータ数が転送された時点でインダイレクトモードでセットされるか、または転送が中止された時点でいずれかのモードでセットされます。また、CTCFに1を書き込むことによってクリアされます。

ビット0 **TEF**：転送エラーフラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときに、インダイレクトモードでセットされます。
CTEFに1を書き込むことによってクリアされます。

30.7.7 HSPIフラグクリアレジスタ（HSPI_FCR）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
											w	w		w	w

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CTOF**：タイムアウトフラグのクリア

このビットに1を書き込むと、HSPI_SRレジスタのTOFフラグがクリアされます。

ビット3 **CSMF**：ステータス一致フラグのクリア

このビットに1を書き込むと、HSPI_SRレジスタのSMFフラグがクリアされます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CTCF**：転送完了フラグのクリア

このビットに1を書き込むと、HSPI_SRレジスタのTCFフラグがクリアされます。

ビット0 **CTEF**：転送エラーフラグのクリア

このビットに1を書き込むと、HSPI_SRレジスタのTEFフラグがクリアされます。

30.7.8 HSPIデータ長レジスタ（HSPI_DLR）

アドレスオフセット：0x040

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DL[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DL[31:0]**：データ長

インダイレクトモードおよび自動ステータスポーリングモードで取得されるデータ数（値+1）
自動ステータスポーリングモードの場合、3より大きい値（4バイトを示す）を使用する必要があります。
インダイレクトモードでのすべて1は、不定長を意味します。この場合、DEVSIZEによって定義されているメモリの最後までHSPIが継続します。
0x0000_0000：1バイトが転送されます。
0x0000_0001：2バイトが転送されます。
0x0000_0002：3バイトが転送されます。
0x0000_0003：4バイトが転送されます。
...
0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。
0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。
0xFFFF_FFFF：不定長--すべてのバイトが外部デバイスの終了（DEVSIZEによって定義）まで転送されます。DEVSIZE = 0x1Fの場合、読出しが無期限に続けられます。
デュアルメモリ構成（DMM = 1）では、DL[0]は、たとえこのビットに0が書き込まれても1のままとなります。したがって、アクセスのたびに必ず偶数バイトが転送されます。
このビットフィールドはメモリマップドモードでは効果がありません。

30.7.9 HSPIアドレスレジスタ（HSPI_AR）

アドレスオフセット：0x048

リセット値：0x0000 0000

このレジスタは、BUSY = 0、かつFMODE≠11のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDRESS[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRESS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ADDRESS[31:0]**：アドレス

外部デバイスに送信されるアドレスHyperBusプロトコルでは、同プロトコルは16bitワード指向なので、このビットフィールドは偶数でなければなりません。デュアルメモリ構成では、AR[0]は0に強制されます。

注意： 一部のメモリ仕様では、各アドレスは16bitの値に対応すると見なされます。HSPIでは、各アドレスは8bitの値に対応すると見なされます。したがって、ソフトウェアはメモリレジスタにアクセスするときにアドレスを2倍する必要があります。

30.7.10 HSPIデータレジスタ（HSPI_DR）

アドレスオフセット：0x050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATA[31:0]**：データ

外部SPIデバイスに対して送受信されるデータ

インダイレクト書込みモードでは、このレジスタに書き込まれたデータはいったんFIFOに格納されてから、データフェーズ時に外部デバイスに送信されます。FIFOが満杯の場合、書き込まれるデータ量を受け入れるだけの十分な空間がFIFOにできるまで、書込み動作は停止されます。

インダイレクト読出しモードでは、このレジスタを読み出すことで（FIFO経由）、外部デバイスから受信したデータを取得できます。FIFOのバイト数が読出し動作で要求したバイト数ほど多くない場合で、BUSY = 1のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読出し動作は停止されます。

自動ステータスポーリングモードでは、このレジスタは外部デバイスから読み出された最後のデータを格納しています（マスキングなし）。

このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書込みモードでの書込みによりFIFOに追加されるバイト数は、バイト書込みで1バイト、ハーフワード書込みで2バイト、ワード書込みで4バイトです。

同様に、インダイレクト読出しモードでの読出しによりFIFOから削除されるバイト数は、バイト読出しで1バイト、ハーフワード読出しで2バイト、ワード読出しで4バイトです。インダイレクトモードでのアクセスは、このレジスタのボトムに整列されなければなりません。つまり、バイト読出しではDATA[7:0]が、ハーフワード読出しではDATA[15:0]が読み出されなければなりません。

30.7.11 HSPIポーリングステータスマスクレジスタ（HSPI_PSMKR）

アドレスオフセット：0x080

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MASK[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MASK[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MASK[31:0]**：ステータスマスク

自動ステータスポーリングモードで受信したステータスバイトに適用されるマスク
ビットnの場合
0：自動ステータスポーリングモードで受信したデータのビットnはマスクされ、その値は一致ロジックでは考慮されません。
1：自動ステータスポーリングモードで受信したデータのビットnはマスクされず、その値は一致ロジックで考慮されます。

30.7.12 HSPIポーリングステータス一致レジスタ（HSPI_PSMAR）

アドレスオフセット：0x088

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MATCH[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MATCH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **MATCH[31:0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

30.7.13 HSPIポーリングインターバルレジスタ（HSPI_PIR）

アドレスオフセット：0x090

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTERVAL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INTERVAL[15:0]**：ポーリングインターバル

自動ステータスポーリングフェーズ時の読出し間のCLKサイクル数

30.7.14 HSPI通信設定レジスタ（HSPI_CCR）

アドレスオフセット：0x100

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル

このビットは、データストローブ管理を有効にします。

0：DQSは無効です。

1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート

このビットは、データフェーズのDTRモードをセットします。

0：データフェーズのDTRモードは無効です。

1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード
このビットフィールドは動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは1本のライン上にあります。
010：データは2本のライン上にあります。
011：データは4本のライン上にあります。
100：データは8本のライン上にあります。
101：データは16本のライン上にあります。
その他：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ
このビットフィールドはオルタネートバイトのサイズを定義します。
00：8bitオルタネートバイト
01：16bitオルタネートバイト
10：24bitオルタネートバイト
11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ

このビットフィールドは命令のサイズを定義します。

00：8bit命令

01：16bit命令

10：24bit命令

11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート

このビットは、命令フェーズのDTRモードをセットします。

0：命令フェーズのDTRモードは無効です。

1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード

このビットフィールドは動作の命令フェーズのモードを定義します。

000：命令はありません。

001：命令は1本のライン上にあります。

010：命令は2本のライン上にあります。

011：命令は4本のライン上にあります。

100：命令は8本のライン上にあります。

その他：予約済み

30.7.15 HSPIタイミング設定レジスタ（HSPI_TCR）

アドレスオフセット：0x108

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SSHIFT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **SSHIFT**：サンプリングシフト

デフォルトでは、HSPIは外部デバイスによるデータ駆動の1/2CLKサイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2サイクルのシフト

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ソフトウェアはSSHIFT = 0であることを確認する必要があります。

ビット29:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。

SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。

30.7.16 HSPI命令レジスタ（HSPI_IR）

アドレスオフセット：0x110

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令
外部SPIデバイスに送信される命令

30.7.17 HSPIオルタネートバイトレジスタ（HSPI_ABR）

アドレスオフセット：0x120

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト
アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

30.7.18 HSPI低電力タイムアウトレジスタ（HSPI_LPTR）

アドレスオフセット：0x130

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEOUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **TIMEOUT[15:0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、HSPIはその後のバイトをプリフェッチし、それらをFIFOに保持します。

このビットフィールドは、クロックが非アクティブになってからNCSを立ち上げるまでにHSPIが待つCLKサイクルの数を示し、外部デバイスを消費電力がより少ない状態にします。

30.7.19 HSPIラップ通信設定レジスタ（HSPI_WPCCR）

アドレスオフセット：0x140

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル

このビットは、データストローブ管理を有効にします。

0：DQSは無効です。

1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート

このビットは、データフェーズのDTRモードをセットします。

0：データフェーズのDTRモードは無効です。

1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード

このビットフィールドは動作のデータフェーズのモードを定義します。

000：データはありません。

001：データは1本のライン上にあります。

010：データは2本のライン上にあります。

011：データは4本のライン上にあります。

100：データは8本のライン上にあります。

101：データは16本のライン上にあります。

その他：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ

このビットフィールドはオルタネートバイトのサイズを定義します。

00：8bitオルタネートバイト

01：16bitオルタネートバイト

10：24bitオルタネートバイト

11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
100：オルタネートバイトは8本のライン上にあります。
101：オルタネートバイトは16本のライン上にあります。
111：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ
このビットフィールドは命令のサイズを定義します。
00：8bit命令
01：16bit命令
10：24bit命令
11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート
このビットは、命令フェーズのDTRモードをセットします。
0：命令フェーズのDTRモードは無効です。
1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード

このビットフィールドは動作の命令フェーズのモードを定義します。

000：命令はありません。

001：命令は1本のライン上にあります。

010：命令は2本のライン上にあります。

011：命令は4本のライン上にあります。

100：命令は8本のライン上にあります。

その他：予約済み

30.7.20 HSPIラップタイミング設定レジスタ（HSPI_WPTCR）

アドレスオフセット：0x148

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SSHIFT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **SSHIFT**：サンプリングシフト

デフォルトでは、HSPIは外部デバイスによるデータ駆動の1/2CLKサイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2サイクルのシフト

データフェーズがDTRモード（DDTR = 1）に設定されている場合、ソフトウェアはSSHIFT = 0であることを確認する必要があります。

ビット29:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。

SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。

30.7.21 HSPIラップ命令レジスタ（HSPI_WPIR）

アドレスオフセット：0x150

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令
外部SPIデバイスに送信される命令

30.7.22 HSPIラップオルタネートバイトレジスタ（HSPI_WPABR）

アドレスオフセット：0x160

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト
アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

30.7.23 HSPI書込み通信設定レジスタ（HSPI_WCCR）

アドレスオフセット：0x180

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DQSE**：DQSイネーブル
このビットは、データストローブ管理を有効にします。
0：DQSは無効です。
1：DQSは有効です。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 **DDTR**：データダブル転送レート
このビットは、データフェーズのDTRモードをセットします。
0：データフェーズのDTRモードは無効です。
1：データフェーズのDTRモードは有効です。

ビット26:24 **DMODE[2:0]**：データモード
このビットフィールドは動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは1本のライン上にあります。
010：データは2本のライン上にあります。
011：データは4本のライン上にあります。
100：データは8本のライン上にあります。
101：データは16本のライン上にあります。
111：予約済み

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **ABSIZE[1:0]**：オルタネートバイトのサイズ
このビットフィールドはオルタネートバイトのサイズを定義します。
00：8bitオルタネートバイト
01：16bitオルタネートバイト
10：24bitオルタネートバイト
11：32bitオルタネートバイト

ビット19 **ABDTR**：オルタネートバイトのダブル-転送レート
このビットは、オルタネートバイトフェーズのDTRモードを設定します。
0：オルタネートバイトフェーズのDTRモードは無効です。
1：オルタネートバイトフェーズのDTRモードは有効です。

ビット18:16 **ABMODE[2:0]**：オルタネートバイトのモード
このビットフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは1本のライン上にあります。
010：オルタネートバイトは2本のライン上にあります。
011：オルタネートバイトは4本のライン上にあります。
100：オルタネートバイトは8本のライン上にあります。
その他：予約済み

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ADSIZE[1:0]**：アドレスのサイズ
このビットフィールドはアドレスサイズを定義します。
00：8bitアドレス
01：16bitアドレス
10：24bitアドレス
11：32bitアドレス

ビット11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズのDTRモードをセットします。
0：アドレスフェーズのDTRモードは無効です。
1：アドレスフェーズのDTRモードは有効です。

ビット10:8 **ADMODE[2:0]**：アドレスモード
このビットフィールドは動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは1本のライン上にあります。
010：アドレスは2本のライン上にあります。
011：アドレスは4本のライン上にあります。
100：アドレスは8本のライン上にあります。
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ISIZE[1:0]**：命令サイズ
このビットフィールドは命令のサイズを定義します。
00：8bit命令
01：16bit命令
10：24bit命令
11：32bit命令

ビット3 **IDTR**：命令ダブル転送レート
このビットは、命令フェーズのDTRモードをセットします。
0：命令フェーズのDTRモードは無効です。
1：命令フェーズのDTRモードは有効です。

ビット2:0 **IMODE[2:0]**：命令モード
このビットフィールドは動作の命令フェーズのモードを定義します。
000：命令はありません。
001：命令は1本のライン上にあります。
010：命令は2本のライン上にあります。
011：命令は4本のライン上にあります。
100：命令は8本のライン上にあります。
その他：予約済み

30.7.24 HSPI書込みタイミング設定レジスタ（HSPI_WTCR）

アドレスオフセット：0x188

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DCYC[4:0]**：ダミーサイクルの数

このビットフィールドは、メモリ待ち時間に応じたダミーフェーズの時間の長さを定義します。SDRモードとDTRモードの両方で、CLKサイクルの数を指定します（0～31）。

30.7.25 HSPI書込み命令レジスタ（HSPI_WIR）

アドレスオフセット：0x190

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **INSTRUCTION[31:0]**：命令

外部SPIデバイスに送信される命令

30.7.26 HSPIライトオルタネートバイトレジスタ（HSPI_WABR）

アドレスオフセット：0x1A0

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。その内容は、メモリマップトモードで書込み動作をリクエストする場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ALTERNATE[31:0]**：オルタネートバイト

アドレスの直後に外部SPIデバイスに送信されるオプションのデータ。

30.7.27 HSPIHyperBus遅延設定レジスタ（HSPI_HLCR）

アドレスオフセット：0x200

リセット値：0x0000 0000

このレジスタは、BUSYが0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **TRWR[7:0]**：読出し･書込み最小リカバリ時間

通信クロックサイクル数で表されるデバイスの読出しから書込み／書込みから読出しへの最小リカバリ時間

ビット15:8 **TACC[7:0]**：アクセス時間

通信クロックサイクル数で表される、メモリ待ち時間に応じたデバイスアクセス時間

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **WZL**：書込みゼロ遅延

このビットは、書込み動作でのゼロ遅延を有効にします。

0：書込みアクセスでの遅延あり

1：書込みアクセスでの遅延なし

ビット0 **LM:**：遅延モード

このビットは、遅延モードを選択します。

0：可変の初期遅延

1：固定遅延

注： デュアルオクタルHyperBus構成を使用する場合に、このビットをセットする必要があります。

30.7.28 HSPIフルサイクル較正の設定（HSPI_CALFCR）

アドレスオフセット：0x210

リセット値：0x0000 0000

この読出し専用レジスタは、DLLマスタが必要とする較正コードを提供し、その遅延が完全なメモリ·クロック·サイクルに相当するようにします。このレジスタの値は、自動較正が終了するたびに更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALMAX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]				
r											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
									r	r	r	r	r	r	r

ビット31 **CALMAX**：最大値

このビットは、メモリクロック周期がDLLマスタの範囲外の場合にセットされ、その場合HSPI_CALFCRとHSPI_CALSRは最大遅延の値で更新されます。

ビット30:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **COARSE[4:0]**：簡易較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **FINE[6:0]**：微較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

30.7.29 HSPIDLLマスタ較正の設定（HSPI_CALMR）

アドレスオフセット：0x218

リセット値：0x0000 0000

DQSなしで読み出すときにフィードバック·クロックを遅延させるためにDLLマスタが使用されます。マスタDLLの遅延が、このレジスタの値によって決まります。

このレジスタは、いつでもソフトウェアで読み出すことができ、BUSY = 0のときにのみ修正できます。

このレジスタは、ハードウェアによって自動的に更新されることはありません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]				
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **COARSE[4:0]**：簡易較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **FINE[6:0]**：微較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

30.7.30 HSPIDLLスレーブ出力較正の設定 （HSPI_CALSOR）

アドレスオフセット：0x220

リセット値：0x0000 0000

DLL出力スレーブは、DDRモードでの書込み動作時に出力データを遅延させるために使用されます。出力スレーブDLLの遅延が、このレジスタの値によって決まります。

このレジスタは、較正の終了時にハードウェアによって自動的に更新されます（HSPI_CALFCRが更新されるのと同じ時点で）。

このレジスタがHSPI_DCR2への最後の書込みの後に書き込まれた場合、自動較正は次の転送では実行されません。

このレジスタは、いつでもソフトウェアで読み出すことができ、BUSY = 0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **COARSE[4:0]**：簡易較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **FINE[6:0]**：微較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

30.7.31 HSPIDLLスレーブ入力較正の設定（HSPI_CALSIR）

アドレスオフセット：0x228

リセット値：0x0000 0000

DLL入力スレーブは、読出し動作でDQSが有効になっているときに、データサンプリングのためにDQS入力を遅延させるために使用されます。入力スレーブDLLの遅延が、このレジスタの値によって決まります。

このレジスタは、較正の終了時にハードウェアによって自動的に更新されます（HSPI_CALFCRが更新されるのと同じ時点で）。

このレジスタがHSPI_DCR2への最後の書込みの後に書き込まれた場合、自動較正は次の転送では実行されません。

このレジスタは、いつでもソフトウェアで読み出すことができ、BUSY = 0のときにのみ修正できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]				
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **COARSE[4:0]**：簡易較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **FINE[6:0]**：微較正

このビットフィールドの遅延単位値は製品のテクノロジーによって異なります（データシートを参照）。

30.7.32 HSPI レジスタマップ

表266.HSPIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	HSPIHSPI_CR	MSEL[1:0]		FMDOE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	FTHRES[5:0]						Res.	DMM	Res.	Res.	TCEN	DMAEN	ABORT	EN
	リセット値	0	0	0	0					0	0		0	0	0	0	0			0	0	0	0	0	0		0			0	0	0	0
0x0004	予約済み	予約済み																															
0x008	HSPIHSPI_DCR1	Res.	Res.	Res.	Res.	Res.	MTYP [2:0]			Res.	Res.	Res.	DEVSIZE[4:0]					Res.	Res.	CSHT[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	FRCK	CKMODE
	リセット値						0	0	0				0	0	0	0	0			0	0	0	0	0	0							0	0
0x00C	HSPI_DCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
	リセット値														0	0	0									0	0	0	0	0	0	0	0
0x010	HSPIHSPI_DCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBOUND[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値												0	0	0	0	0																
0x014	HSPI_DCR4	REFRESH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018 - 0x01C	予約済み	予約済み																															
0x020	HSPI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[6:0]							Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																		0	0	0	0	0	0	0			0	0	0	0	0	0
0x024	HSPI_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x028- 0x03C	予約済み	予約済み																															
0x040	HSPI_DLR	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	予約済み	予約済み																															
0x048	HSPI_AR	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	予約済み	予約済み																															
0x050	HSPI_DR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054 - 0x07C	予約済み	予約済み																															
0x080	HSPI_PSMKR	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表266.HSPIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x084	予約済み	予約済み																															
0x088	HSPI_PSMAR	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08C	予約済み	予約済み																															
0x090	HSPI_PIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094 - 0x0FC	予約済み	予約済み																															
0x100	HSPI_CCR	Res.	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
	リセット値			0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x104	予約済み	予約済み																															
0x108	HSPI_TCR	Res.	SSHIFT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値		0																									0	0	0	0	0	
0x10C	予約済み	予約済み																															
0x110	HSPI_IR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x114 ~0x11C	予約済み	予約済み																															
0x120	HSPI_ABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x124 - 0x12C	予約済み	予約済み																															
0x130	HSPI_LPTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x134 - 0x13C	予約済み	予約済み																															
0x140	HSPI_WPCCR	Res.	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE [1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x144	予約済み	予約済み																															
0x148	HSPI_WPTCR	Res.	SSHIFT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値		0																									0	0	0	0	0	
0x14C	予約済み	予約済み																															
0x150	HSPI_WPIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x154 - 0x15C	予約済み	予約済み																															

表266.HSPIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x160	HSPI_WPABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x164 - 0x17C	予約済み	予約済み																															
0x180	HSPI_WCCR	Res.	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE [1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x184	予約済み	予約済み																															
0x188	HSPI_WTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値																												0	0	0	0	0
0x18C	予約済み	予約済み																															
0x190	HSPI_WIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x194 - 0x19C	予約済み	予約済み																															
0x1A0	HSPI_WABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1A4 - 0x1FC	予約済み	予約済み																															
0x200	HSPI_HLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]								TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0	0
0x204 - 0x20C	予約済み	予約済み																															
0x210	HSPI_CALFCR	CALMAX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
	リセット値	0											0	0	0	0	0										0	0	0	0	0	0	0
0x214	予約済み	予約済み																															
0x218	HSPI_CALMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
	リセット値												0	0	0	0	0										0	0	0	0	0	0	0
0x22C	予約済み	予約済み																															
0x220	HSPI_CALSOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
	リセット値												0	0	0	0	0										0	0	0	0	0	0	0
0x224	予約済み	予約済み																															
0x228	HSPI_CALSIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COARSE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINE[6:0]						
	リセット値												0	0	0	0	0										0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

31 セキュアデジタル入力／出力MultiMediaCardインタフェース（SDMMC）

31.1 SDMMCの主な機能

SD/SDIO、embedded MultiMediaCard（*e*•MMC）ホストインタフェース（SDMMC）は、AHBバスとSDメモリカード、SDIOカード、および*e*•MMCデバイス間のインタフェースを提供します。

MMCA技術委員会から発行されたMultiMediaCardシステム仕様書は、MultiMediaCard Associationのウェブサイトwww.jedec.orgから入手できます。

SDメモリカードとSD I/Oカードシステム仕様書は、SD card Associationのウェブサイト www.sdcard.orgから入手できます。

SDMMCには以下の機能があります。

- Embedded MultiMediaCard System Specification Version 5.1 に準拠。
  3種類の異なるデータバスモードのカードに対応：1bit（デフォルト）、4bitおよび8bit。
  （HS200 SDMMC_CK速度は最大許容I/O速度に制限されています）（HS400はサポートされていません）
- 以前のバージョンのMultiMediaCardとの完全互換（下位互換性）
- SD memory card specifications version 6.0 に完全準拠。
  （SDR104 SDMMC_CK速度は最大許容I/O速度に制限され、SPIモードおよびUHS-IIモードはサポートされません）
- SDIO card specification version 4.0 に完全準拠。
  2種類の異なるデータバスモードのカードに対応：1bit（デフォルト）および4bit。
  （SDR104 SDMMC_CK速度は最大許容I/O速度に制限され、SPIモードおよびUHS-IIモードはサポートされません）
- 8bitモードで最大 208MB/秒のデータ転送
  （最大許容I/O速度に基づきます）
- 外部双方向ドライバを制御するデータおよびコマンド出力有効信号
- IDMAリンクリストのサポート

MultiMediaCard/SDバスは、カードをホストに接続します。

現行バージョンのSDMMCは、1度に1枚のSD/SDIO/*e*•MMCカードと、*e*•MMCのスタックだけをサポートします。

31.2 SDMMCの実装

表267.SDMMCの機能

SDMMCのモード／機能(1)	SDMMC1	SDMMC2(2)
可変遅延（SDR104、HS200）	X	X
SDMMC_CKIN	X	-
SDMMC_CDIR、SDMMC_D0DIR	X	-
SDMMC_D123DIR	X	-

1. X：サポートされています。
2. STM32U535/545では使用できません。

31.3 SDMMCバストポロジ

バス上の通信は、コマンド／レスポンスとデータ転送に基づいて行われます。

SD/SDIO/*e*•MMCバス上の基本的なトランザクションは、コマンド／レスポンストランザクションです。これらの種類のバストランザクションでは、コマンドやレスポンスの決められた通信フォーマットでカードの情報を直接転送します。また、データトークンを持つ操作もあります。

データ転送は次の方法に従って行います。

- ブロックモード：Nの範囲0～14でブロックサイズ2^N バイトのデータブロック
- SDIOマルチバイトモード：ブロックサイズ範囲1～512バイトの単一データブロック
- *e*•MMCストリームモード：連続的なデータ･ストリーム

e•MMCカードに対するデータ転送は、データブロックまたはストリーム単位で行われます。

図180. SDMMC「レスポンス無し」と「データなし」操作

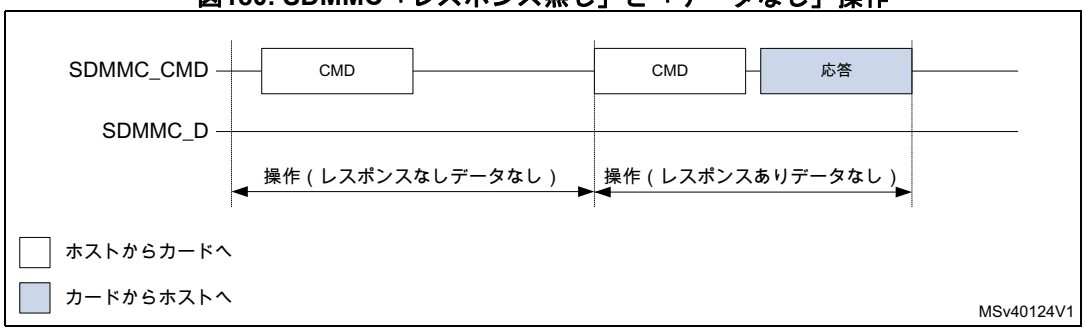
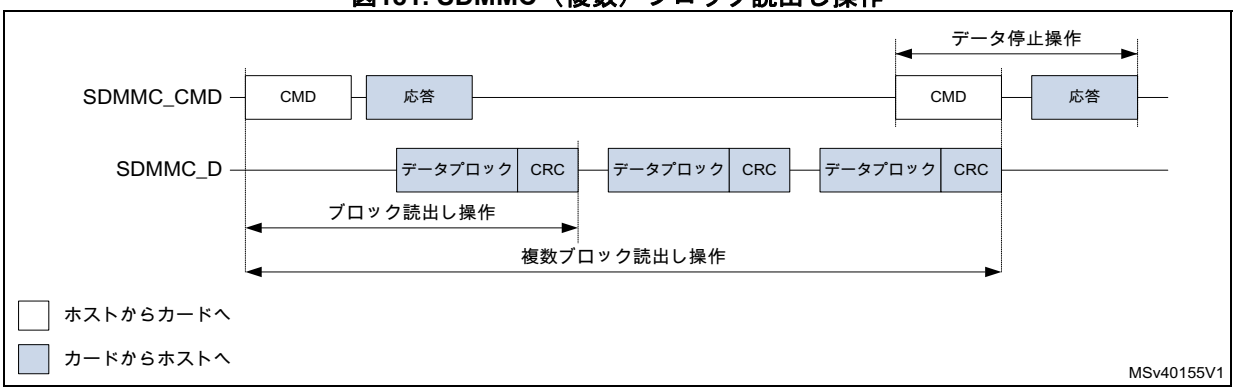
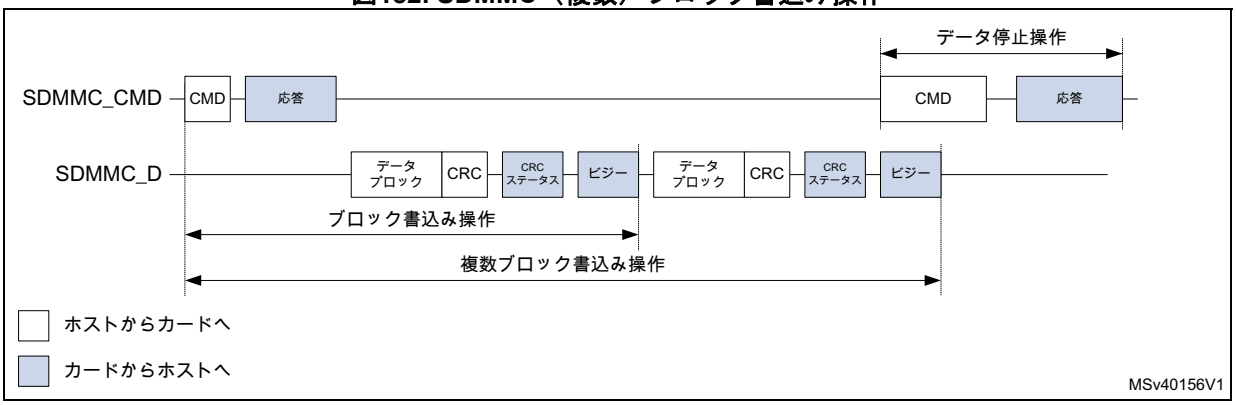


図181. SDMMC（複数）ブロック読出し操作



注： **事前定義されたブロック数でのe•MMC複数ブロック読出し終了時に、送信停止コマンドは必要ありません。**

図182. SDMMC（複数）ブロック書込み操作



注： **事前定義されたブロック数でのeMMC複数ブロック書込み終了時に、送信停止コマンドは必要ありません。**

注： **SDMMCは、ビジー信号がアサートされている限り（SDMMC_D0をローへ固定）、データを一切送信しません。**

図183. SDMMC（シーケンシャル）ストリーム読出し操作

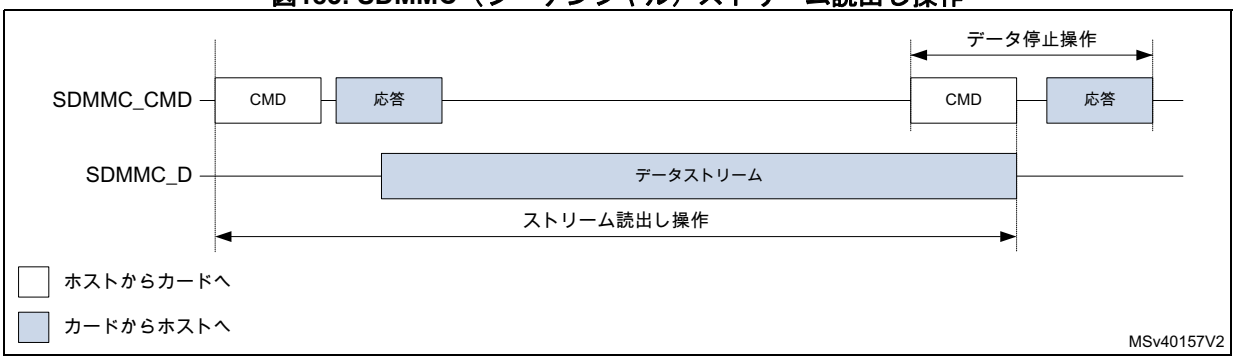
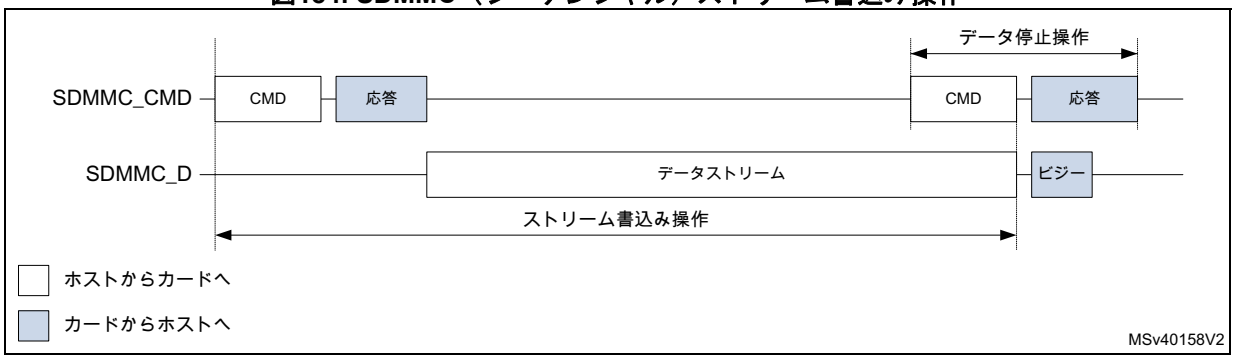


図184. SDMMC（シーケンシャル）ストリーム書込み操作



ストリームデータ転送は、シングルデータレートモード（DS、ハイスピード およびSDR）のSDMMC_D0で1bit幅のビットバス設定でのみ動作します。

31.4 SDMMC動作モード

表268.SDMMC動作モードSDおよびSDIO

SDIOバス速度モード(1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz](4)	信号電圧 [V]
DS（デフォルトスピード）	12.5	25	3.3
HS（ハイスピード）	25	50	3.3
SDR12	12.5	25	1.8
SDR25	25	50	1.8
DDR50	50	50	1.8
SDR50	50	100	1.8
SDR104	104	208	1.8

1. SDRシングルデータレートシグナリング
2. DDRダブルデータレートシグナリング（データが両方のSDMMC_CKクロックエッジでサンプリングされます）
3. 4bitバス幅でのSDIOバス速度
4. 最大許容I/O速度に基づく最大周波数

SDR104モードでは、サンプリングポイントチューニングを使用した可変遅延サポートが必要です。SDR50モードでは、可変遅延の使用はオプションです。

表269.SDMMC動作モード *e*•MMC

e•MMCバス速度モード (1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz](4)	信号電圧 [V](5)
レガシー互換	26	26	3/1.8/1.2V
ハイスピードSDR	52	52	3/1.8/1.2V
ハイスピードDDR	104	52	3/1.8/1.2V
ハイスピードHS200	200	200	1.8/1.2V

1. SDRシングルデータレートシグナリング
2. DDRダブルデータレートシグナリング（データが両方のSDMMC_CKクロックエッジでサンプリングされます）
3. 8bitバス幅での *e*•MMCバス速度
4. 最大許容I/O速度に基づく最大周波数
5. サポートされる信号電圧レベルはI/Oポートの特性によって異なります。デバイスのデータシートを参照してください。

HS200モードでは、サンプリングポイントチューニングを使用した可変遅延サポートが必要です。

31.5 SDMMCの機能説明

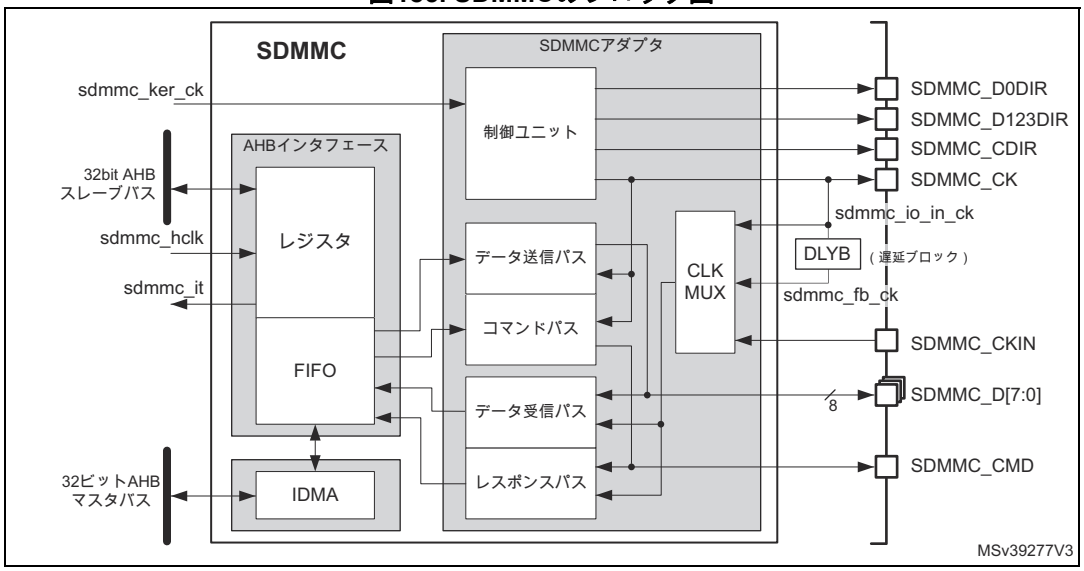
SDMMCは、4つの部分で構成されています。

- AHBスレーブインタフェースは、SDMMCアダプタレジスタにアクセスして、割込み信号およびIDMA制御信号を生成します。
- SDMMCアダプタブロックは、クロック生成ユニット、コマンドおよびデータ転送など、*e*•MMC/SD/SD I/Oカードに固有のすべての機能を提供します。
- 内部DMA（IDMA）ブロックはAHBマスタインタフェースを搭載しています。
- 遅延ブロック（DLYB）は、受信データのサンプル·クロック配置を処理します。遅延ブロックは、SDMMCの一部ではありません。遅延ブロックは、SDR104またはHS200をサポートする場合に必須です。

31.5.1 SDMMCブロック図

図185 にSDMMCブロック図を示します。

図185. SDMMCのブロック図



31.5.2 SDMMCピンおよび内部信号

表270 ではSDMMC内部入力／出力信号を、表271 ではSDMMCピン（オルタネート機能）を一覧化しています。

表270.SDMMC内部入力／出力信号

信号名	信号タイプ	説明
sdmmc_ker_ck	デジタル入力	SDMMCカーネル·クロック
sdmmc_hclk	デジタル入力	AHBクロック
sdmmc_it	デジタル出力	SDMMCグローバル割込み
sdmmc_io_in_ck	デジタル入力	SD/SDIO/*e*•MMCカードフィードバッククロック。この信号は、SDMMC_CKピンに内部接続されます（DSおよび ハイスピードモードの場合）。
sdmmc_fb_ck	デジタル入力	DLYB遅延ブロック後のSD/SDIO/*e*•MMCカード調整フィードバッククロック（SDR50、DDR50、SDR104、HS200の場合）

表271.SDMMCピン

信号名	信号タイプ	説明
SDMMC_CK	デジタル出力	SD/SDIO/*e*•MMCカードへのクロック
SDMMC_CKIN	デジタル入力	SD/SDIO/*e*•MMCカードの外部ドライバからのクロックフィードバック（SDR12、SDR25、SDR50、DDR50の場合）
SDMMC_CMD	デジタル入出力	SD/SDIO/*e*•MMCカードの双方向コマンド／レスポンス信号
SDMMC_CDIR	デジタル出力	SD/SDIO/*e*•MMCカードのSDMMC_CMD信号のI/O方向指示
SDMMC_D[7:0]	デジタル入出力	SD/SDIO/*e*•MMCカードの双方向データライン
SDMMC_D0DIR	デジタル出力	SD/SDIO/*e*•MMCカードのSDMMC_D0データラインのI/O方向指示
SDMMC_D123DIR	デジタル出力	SD/SDIO/*e*•MMCカードのデータラインSDMMC_D[3:1] のI/O方向指示

31.5.3 概要

SDMMC_D[7:0]ラインには異なる動作モードがあります。

- デフォルトでは、SDMMC_D0ラインはデータ転送に使用されます。初期化後、ホストはデータバス幅を変更することができます。
- *e*•MMCでは、1bit（SDMMC_D0）、4bit（SDMMC_D[3:0]）、または8bit（SDMMC_D[7:0]）のデータバス幅を使用できます。
- SDカードまたはSDIOカードでは、1bit（SDMMC_D0）または4bit（SDMMC_D[3:0]）を使用できます。すべてのデータラインはプッシュプルモードで動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、データラインでのデータフローの方向をI/O方向信号で示します。**SDMMC_D0DIR** 信号はSDMMC_D0データライン、**SDMMC_D123DIR** はSDMMC_D[3:1] データラインのI/O方向を示します。

SDMMC_CMDはプッシュプルモードでのみ動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、SDMMC_CMDラインでのデータフローの方向をI/O方向信号**SDMMC_CDIR**で示します。

カードへの**SDMMC_CK**クロックは**sdmmc_ker_ck**から生成されます。

- sdmmc_ker_ckクロックが50% デューティサイクルの場合、バイパスモードでも使用できます（CLKDIV = 0）。
- sdmmc_ker_ckクロックのデューティサイクルが50% ではない場合、2以上で分周するためにCLKDIVを使用する必要があります（CLKDIV > 0）。
- SDMMC_CMD / SDMMC_D[7:0] 出力とSDMMC_CKの間の位相関係は、NEGEDGEビットで選択できます。この位相関係は、CLKDIV、NEGEDGEおよびDDRの設定に依存します。図186を参照してください。

図186. SDMMCコマンドおよびデータの位相関係

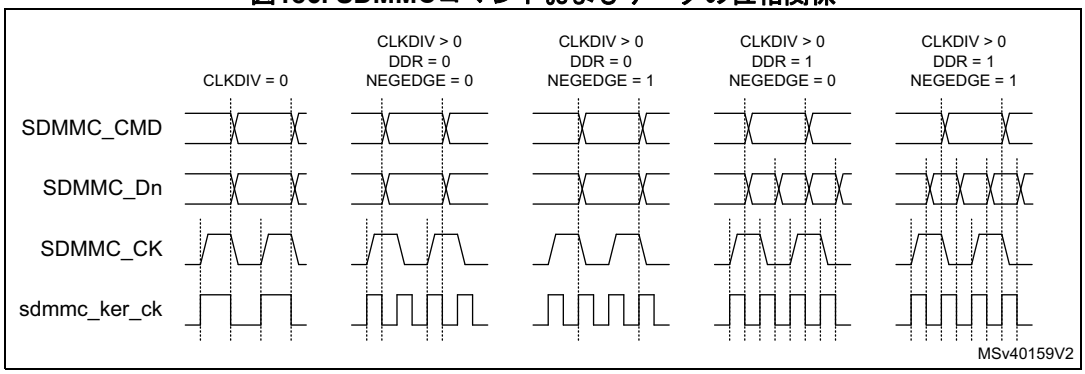


表272.SDMMCコマンドおよびデータの位相選択

CLKDIV	DDR	NEGEDGE	SDMMC_CK	コマンド出力	データ出力
0	x	x	= sdmmc_ker_ck	sdmmc_ker_ck立下がりエッジで生成	
>0	0	0	sdmmc_ker_ck 立上がり エッジで生成	SDMMC_CK立上がりエッジに続くsdmmc_ker_ck立下がりエッジで生成	
		1		SDMMC_CK立下がりエッジを生成する同じsdmmc_ker_ck立上がりエッジで生成	
	1	0		SDMMC_CK立上がりエッジに続くsdmmc_ker_ck 立下がりエッジで生成	SDMMC_CKエッジに続く sdmmc_ker_ck 立下がりエッジで生成
		1		SDMMC_CK立下がりエッジを生成する同じ sdmmc_ker_ck立上がりエッジで生成	

デフォルトでは、**sdmmc_io_in_ck** フィードバッククロック入力が、SDMMC受信パスで受信データをサンプリングするために選択されます。これはSDMMC_CKピンから生成されます。

サンプリングクロックの位相をチューニングして受信データタイミングを調整するために、デバイスで使用できるDLYB遅延ブロックを **sdmmc_io_in_ck** 信号（DLYB入力dlyb_in_ck）とSDMMCの **sdmmc_fb_ck** クロック入力（DLYB出力dlyb_out_ck）の間に接続できます。受信パスで **sdmmc_fb_ck** クロック入力を選択すると、受信データの位相調整サンプリングクロックを使用できるようになります。これは、SDMMCでSDR104動作モードおよびHS200動作モードをサポートするために必要であり、SDR50モードおよびDDR50モードではオプションです。

外部ドライバ（電圧スイッチトランシーバ）を使用する場合、SDMMC_CKINフィードバッククロック入力を選択して受信データをサンプリングできます。

SD/SDIO/*e*•MMCカードの場合、クロック周波数は0～208MHzの範囲（最大I/O速度により制限）で可変です。

選択したバスモード（SDRまたはDDR）に応じて、各クロックサイクルでSDMMC_D[7:0] ラインに1bitまたは2bitが転送されます。SDMMC_CMDラインは各クロックサイクルで1bitだけ転送します。

31.5.4 SDMMCアダプタ

SDMMCアダプタ（図 185：SDMMCのブロック図を参照）は、MultiMediaCardスタックやセキュアデジタルメモリカードとのインタフェースを提供するマルチメディア／セキュアデジタルメモリカードバスマスタです。これは次のサブユニットで構成されています。

- 制御ユニット
- データ送信パス
- コマンドパス
- データ受信パス
- レスポンスパス
- 受信データパスクロックマルチプレクサ
- SDMMCの外部の遅延ブロック（DLYB）
- アダプタレジスタブロック
- データFIFO
- 内部DMA（IDMA）

注： **アダプタレジスタとFIFOは、*AHB*クロック·ドメイン（sdmmc_hclk）を使用します。制御ユニット、コマンドパス、およびデータ送信パスは、SDMMCアダプタ·クロック·ドメイン（sdmmc_ker_ck）を使用します。レスポンスパスとデータ受信パスは、sdmmc_io_in_ck、SDMMC_CKIN、またはDLYBによって生成されたsdmmc_fb_ck からのSDMMCアダプタフィードバッククロックドメインを使用します。**

デバイスのDLYB遅延ブロックは、SDMMCアダプタと連携して、SDMMC受信モードでサンプリングクロックの位相を調整してデータを受信するために使用できます。これは、SDMMCでSDR104動作モードおよびHS200動作モードをサポートするために必要であり、SDR50モードおよびDDR50モードではオプションです。

アダプタレジスタブロック

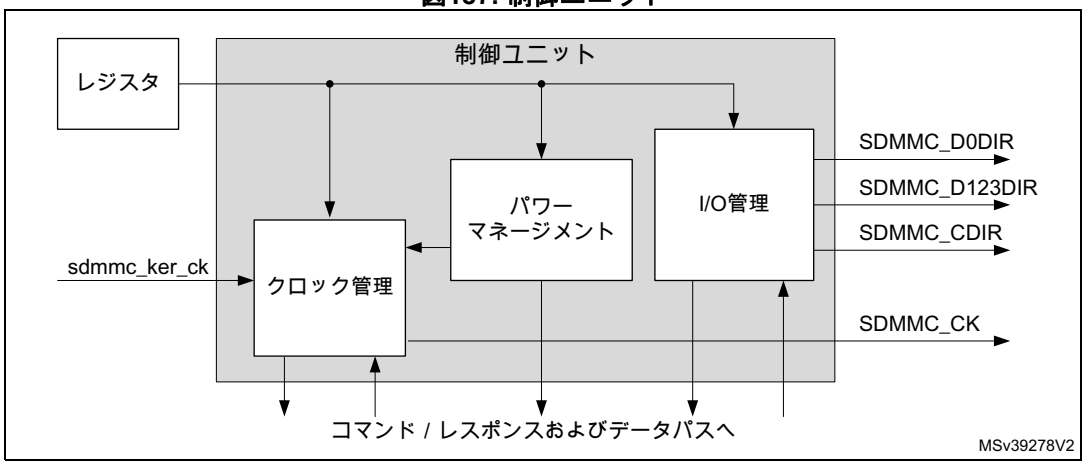
アダプタレジスタブロックには、すべてのシステム制御レジスタ、SDMMCコマンドおよびレスポンスレジスタ、データFIFOが含まれています。

このブロックは、SDMMCアダプタのスタティックフラグをクリアするSDMMCクリアレジスタの対応するビット位置からの信号も生成します。

制御ユニット

図187に示された制御ユニットには、電源管理機能、分周回路でのSDMMC_CKクロック管理、I/O方向管理が含まれています。

図187. 制御ユニット



電源管理サブユニットは、パワーオフおよびパワーアップフェーズの間、カードバス出力信号を無効にします。

3つの電源フェーズがあります。

- パワーオフ
- パワーアップ
- パワーオン

クロック管理サブユニットは、sdmmc_ker_ckを使用してSDMMC_CKを生成し、分周制御を提供します。また、フロー制御などでSDMMC_CKを停止することもできます。

クロック出力は、次の場合にインアクティブです。

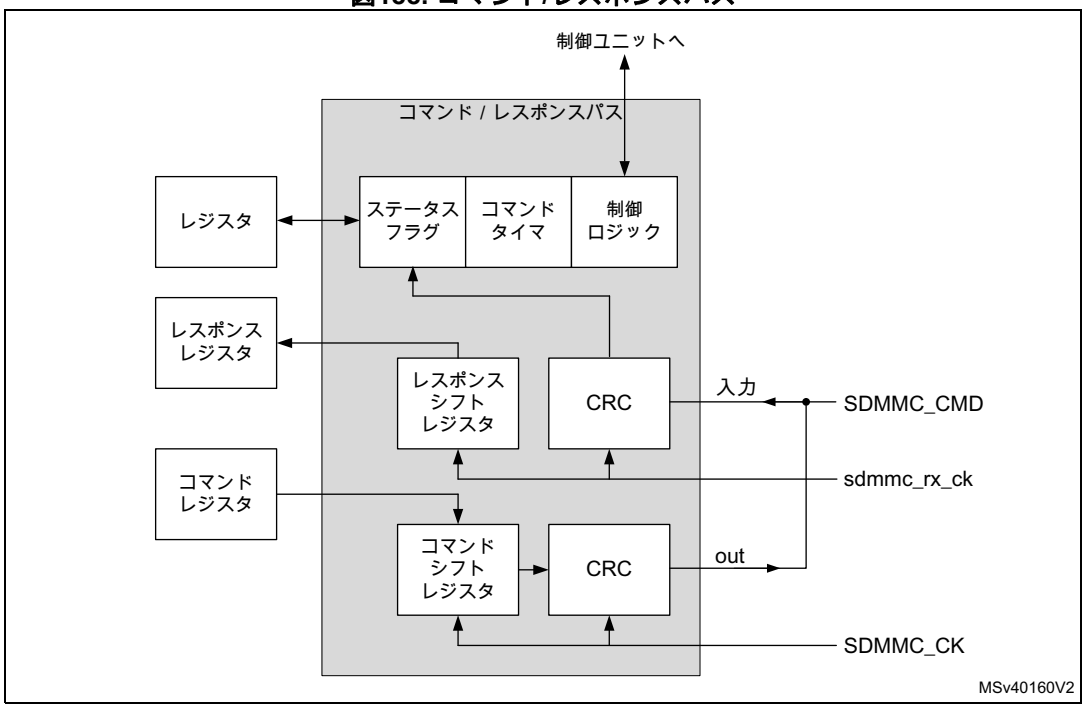
- リセット後
- パワーオフまたはパワーアップフェーズ中
- 節電モード（レジスタビットPWRSAV）が有効であり、カードバスが8クロック周期の間アイドル状態の場合。クロックは、コマンド/レスポンスCPSMおよびデータパスDPSMサブユニットの両方がアイドルフェーズに移行してから8サイクル後に停止します。クロックは、コマンド/レスポンスCPSMまたはデータパスDPSMがアクティブになった（有効になった）ときに再開します。

I/O管理サブユニットは、SDMMC_DnおよびSDMMC_CMDのI/O方向信号を管理し、外部電圧トランシーバを制御します。

コマンド/レスポンスパス

コマンド/レスポンスパスサブユニットは、SDMMC_CMDラインでコマンドとレスポンスを転送します。コマンドパスはSDMMC_CKに基づいてクロック供給され、コマンドをカードに送信します。レスポンスパスはsdmmc_rx_ckに基づいてクロック供給され、カードからレスポンスを受信します。

図188. コマンド/レスポンスパス

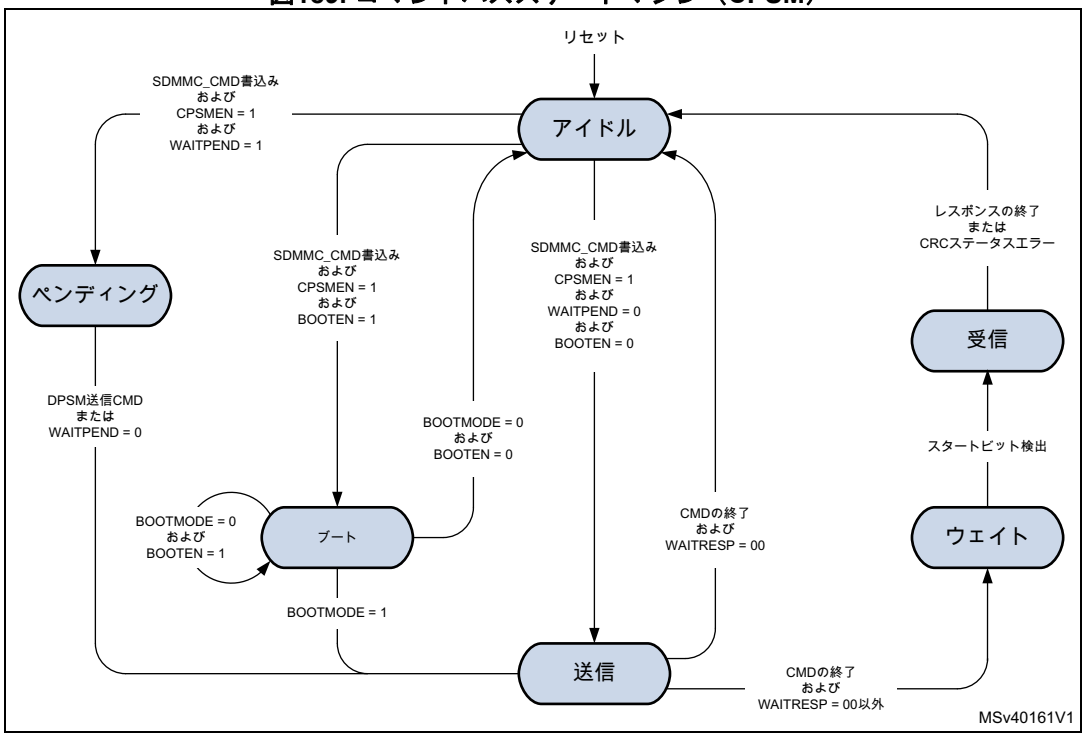


コマンド/レスポンスパスステートマシン（CPSM）

- コマンドレジスタが書き込まれて、有効ビットがセットされると、コマンド転送が開始します。コマンドが送信されると、CRCが付加され、コマンドパスステートマシン（CPSM）がステータスフラグをセットします。
  - レスポンスが必要ではない場合は、アイドル状態に入ります。
  - レスポンスが必要な場合は、レスポンスを待ちます。
- レスポンス受信時：
  - CRC付きのレスポンスの場合、受信したCRCコードと内部生成されたコードを比較し、結果に応じて適切なステータスフラグをセットします。
  - CRCなしのレスポンスの場合、CRCはチェックされず、適切なステータスフラグがセットされません。

アイドル状態ではない場合などCPSMがアクティブであるときは必ずCPSMACTビットがセットされます。

図189. コマンドパスステートマシン（CPSM）



- **アイドル**：コマンドパスがインアクティブです。コマンド制御レジスタが書き込まれ、有効ビット（CPSMEN）がセットされると、CPSMはSDMMC_CKクロック（節電PWRSAVビットのために停止している場合）を有効化して次の状態に移行します。
  - WAITPEND = 0かつBOOTEN = 0の場合は送信状態
  - WAITPEND = 1の場合はペンディング状態
  - BOOTEN = 1の場合はブート状態
- **送信**：コマンドが送信され、CRCが付加されます。
  - CMDTRANSビットがセットされている場合、またはBOOTENビットがセットされていて、かつBOOTMODEが代替ブートで、DTDIR = 受信の場合、コマンド終了時にCPSM DataEnable信号がDPSMに発行されます。
  - CMDTRANSビットがセットされていて、かつCMDSUSPENDビットが0の場合、コマンド終了時に割込み期間が終了します。
  - CMDSTOPビットがセットされている場合、コマンド終了時にCPSM Abort信号がDPSMに発行されます。
  - レスポンスが期待されていない（WAITRESP = 00）場合、CPSMはアイドル状態に移行し、CMDSENTフラグがセットされます。BOOTMODE = 1かつBOOTEN = 0の場合、CMDSENTフラグはコマンド終了ビットの56サイクル後に遅延され、それ以外の場合、コマンド終了ビット直後に CMDSENTフラグが生成されます。
    RESPCMDRレジスタおよびRESPxRレジスタは変更されません。
  - コマンドレスポンスが期待されている（WAITRESPが00ではない）場合、CPSMはウェイトステートに移行し、レスポンスタイムアウトを開始します。

- **待機**：コマンドパスがレスポンスを待ちます。
  - WAITINTビットが0の場合、コマンドタイマが動作を開始し、CPSMはスタートビットを待ちます。
  a) スタートビットがタイムアウトまでに検出された場合、CPSMは受信状態に移行します。
  b) CPSM がレスポンススタートビットを検出する前にタイムアウトに達した場合には、タイムアウトフラグ（CTIMEOUT）がセットされ、CPSMがアイドル状態に移行します。RESPCMDRレジスタおよびRESPxRレジスタは変更されません。
  - WAITINTビットが1の場合、タイマは無効になり、CPSMはカードの1枚からの割込みリクエスト（レスポンススタートビット）を待ちます。
  a) スタートビットが検出された場合、CPSMは受信状態に移行します。
  b) WAITINTに0を書き込む（割込みモードを中止する）と、ホストは自身でレスポンスを送信し、スタートビット検出時にCPSMは受信状態に移行します。
- **受信**：コマンドレスポンスが受信されます。コマンド制御レジスタのレスポンスモードビットWAITRESPに応じて、CRCの有無によりレスポンスを短くしたり長くしたりできます。受信したCRCコードがある場合、内部で生成されたCRCコードを使用して確認されます。
  - CMDSUSPENDビットがセットされていて、かつSDIOレスポンスビットBS = 0（レスポンスビット [39]）の場合、レスポンスの後に割込み期間が開始されます。CMDSUSPENDビットがクリアされている場合、またはCMDSUSPENDビットが1で、かつSDIOレスポンスビットBS = 1（レスポンスビット [39]）の場合、割込み期間は開始されません。
  - CMDTRANS ビットと CMDSUSPEND ビットがセットされていて、かつ SDIO レスポンスビットDF = 1（レスポンスビット [32]）の場合、レスポンスの後に割込み期間が終了します。
  - CRCステータスがパスするか、CRCが存在しない場合、CMDRENDフラグがセットされ、CPSMがアイドル状態に移行します。RESPCMDRレジスタおよびRESPxRレジスタは受信したレスポンスで更新されます。
    - BOOTMODE = 1かつBOOTEN = 0の場合、CMDRENDフラグはレスポンス終了ビットの56サイクル後に遅延され、それ以外の場合、レスポンス終了ビット直後にCMDRENDフラグが生成されます。
    - CMDTRANSビットがセットされていて、かつDTDIR = 送信の場合、コマンドレスポンス終了時にCPSM DataEnable信号がDPSMに発行されます。
  - CRCステータスが失敗となる場合、CCRCFAILフラグがセットされ、CPSMがアイドル状態に移行します。RESPCMDRレジスタおよびRESPxRレジスタは受信したレスポンスで更新されます。
- **ペンディング**：コマンドレジスタのペンディングWAITPENDビットに応じて、CPSMはペンディング状態に入ります。
  - DATALENGTH =< 5バイトの場合、CPSMは送信状態に入り、DataEnable信号を生成してCMD12送信停止コマンドで調整されたデータ転送を開始します。
  - DATALENGTH > 5バイトの場合、データ転送を開始するためにCPSM DataEnable信号がDPSMに発行されます。CPSMは DPSMからの送信CMD信号を待ってから、送信状態に入ります。これにより、CMD12送信停止コマンドなどをデータで調整して送信できます。
  - WAITPENDに0を書き込むと、CPSMは送信状態に移行します。
- **ブート**：コマンドレジスタのBOOTEN ビットがセットされていると、CPSM は次の場合にブート状態に入ります。
  - BOOTMODE = 0でSDMMC_CMDラインがローに駆動されている場合、かつCMDTRANSビットがセットされ、DTDIR = 受信の場合、CPSM DataEnable信号がDPSMに発行されます。これにより、通常ブート動作が可能になります。この状態からは、レジスタビットBOOTENをクリアすることでブート手順の終了時に離脱し、これによってSDMMC_CMDラインがハイに駆動され、CPSM Abort信号がDPSMに発行されてから、アイドル状態に移

行します。SDMMC_CMDラインがハイになってから56サイクル後にCMDSENTフラグが生成されます。

– BOOTMODE = 1の場合、送信状態へ移行します。これでCMD0（ブート）の送信が可能になります。BOOTENをクリアしても影響はありません。

注： CPSMは、N_{CC} およびN_{RC} タイミング制約を満たすために、少なくとも8 SDMMC_CK周期の間、アイドル状態を保ちます。N_{CC} は、2つのホストコマンド間の最小遅延であり、N_{RC} は、ホストコマンドとカードレスポンスの間の最小遅延です。

注： レスポンスタイムアウトは、64 SDMMC_CKクロック周期の固定値です。

コマンドは、動作を開始するトークンです。コマンドは、ホストから1枚のカード（アドレス付きコマンド）または接続されたすべてのカードに送信されます（ブロードキャストコマンドは*e*•MMC V3.31以前で使用可能）。コマンドは、SDMMC_CMDライン上で連続的に転送されます。すべてのコマンドは、固定長48bitです。SDメモリカード、SDIOカード、*e*•MMCカードのコマンドトークンの一般的なフォーマットを表273. に示します。

コマンドトークンデータは2つのレジスタから取得され、1つには32bitの引数が含まれ、他方には6bitのコマンドインデックス（カードに送信される6bit）が含まれます。

表273.コマンドトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	1	送信ビット
[45:40]	6	x	コマンドインデックス
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

コマンドデータの隣にはコマンドタイプ（WAITRESP）ビットがあり、コマンドパスステートマシン（CPSM）を制御します。これらのビットによって、コマンドがレスポンスを必要とするかどうかと、レスポンスが短い（48bit）か長い（136bit）か、CRCが存在するかどうかも決まります。

レスポンスは、以前に受信したコマンドへの応答として、アドレス指定されたカードまたは接続されたすべてのカードから同時にホストに送信されるトークンです。すべてのレスポンスはコマンドラインSDMMC_CMD経由で送信されます。レスポンスの送信は、常に、レスポンスコードワードに対応するビット文字列の左のビットから始まります。コード長は、レスポンスのタイプに依存します。レスポンストークンR1、R2、R3、R4、R5、R6には、コンテンツに応じてさまざまなコード体系があります。SDメモリカード、SDIOカード、*e*•MMCカードのレスポンストークンの一般的なフォーマットを表274、表275、表276に示します。

レスポンスは、常にスタートビット（常に0）から始まり、送信方向を示すビットが続きます（card=0）。下の表でxで示されている値は、可変エントリを示します。一部を除くほとんどのレスポンスは、CRCによって保護されます。すべてのコマンドコードワードは、終了ビット（常に1）で終わります。

レスポンストークンデータは5つのレジスタに格納され、4つは32bitのカードステータス、OCRレジスタ、引数または内部CRCを含んだ127bit CIDやCSDレジスタを含み、あと1つのレジスタは6bitのコマンドインデックスを含みます。

表274.CRCありのショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み111111）
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

表275.CRCなしのショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み111111）
[39:8]	32	x	引数
[7:1]	7	1111111	（予約済み1111111）
0	1	1	終了ビット

表276.CRCありのロングレスポンスのトークンフォーマット

ビット位置	幅	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	111111	予約済み
[127:1]	127:8	x	CIDまたはCSDスライス
	7:1	x	CRC7（CID、CSDレジスタに含まれる）
0	1	1	終了ビット

コマンド/レスポンスパスは、コマンドを送信したりレスポンスを受信したりできるように半二重モードで動作します。CPSMが送信状態でない場合、SDMMC_CMD出力はハイインピーダンス状態になります。SDMMC_CMDで送信されたデータは、NEGEDGEレジスタビットに応じてSDMMC_CKと同期します。図186を参照してください。

コマンドおよびCRCありのショートレスポンスでは、CRC生成回路は、CRCコードの前のすべての40bitについてCRCチェックサムを計算します。これには、スタートビット、送信ビット、コマンドインデックス、およびコマンド引数（またはカードステータス）が含まれます。

ロングレスポンスでは、R2 CIDまたはCSDの120bitを超える場合のみCRCチェックサムを計算します。スタートビット、送信ビット、および6つの予約ビットは、CRC計算では使用されませんので注意してください。

CRCチェックサムは7bitの値です。

CRC[6:0] = 剰余 [(M(x) * x^7) / G(x)]

G(x) = $x^7 + x^3 + 1$

M(x) =（最初のビット）* x^n +（2番目のビット）* x^{n-1}+... +（CRCの前の最後のビット）* x^0

n = 39または119の場合。

CPSMENがセットされている場合、CPSMは、さまざまな動作モードに対応するために数多くの固有コマンドを送信できます。表277 を参照してください。

表277.固有コマンドの概要

VSWITCH	BOOTEN	BOOTMODE	CMDTRANS	WAITPEND	CMDSTOP	WAITINT	説明
1	x	x	x	x	x	x	電圧スイッチシーケンス開始
0	1	x	x	x	x	x	通常ブート開始
0	1	1	x	x	x	x	代替ブート開始
0	0	1	x	x	x	x	代替ブート停止
0	0	0	1	x	x	x	関連付けしたデータ転送でコマンド送信
0	0	0	0	1	1	x	*e*•MMC ストリームデータ転送、データ転送終了までコマンド（STOP_TRANSMISSION）ペンディング
0	0	0	0	1	0	x	*e*•MMCストリームデータ転送、データ転送終了まで（STOP_TRANSMISSION）ペンディング以外のコマンド
0	0	0	0	0	1	x	コマンド送信（STOP_TRANSMISSION）、進行中のデータ送信を停止
0	0	0	0	0	0	1	*e*•MMCウェイト割込み（Wait-IRQ）モードへ移行
0	0	0	0	0	0	0	その他の固有コマンドなし

コマンド/レスポンスパスは、表278 に示すステータスフラグおよび関連付けられたクリアビットを実装しています。

表278.コマンドパスステータスフラグ

フラグ	説明
CMDSENT	レスポンスなしのコマンド終了時にセット。（CPSMは送信からアイドルに移行）
CMDREND	CRCがOKのときにコマンドレスポンス終了時にセット（CPSMが受信からアイドルに移行）。
CCRCFAIL	CRCがFAILのときにコマンドレスポンス終了時にセット（CPSMが受信からアイドルに移行）。
CTIMEOUT	タイムアウト前に受信したレスポンススタートビットがないときにコマンド後にセット。（CPSMはウェイトからアイドルに移行）
CKSTOP	CRCがOKかつSDMMC_CKが停止したときに電圧スイッチ（VSWITCHEN = 1）コマンドレスポンス後にセット。（CPSMに影響なし）
VSWEND	5ms + 1msの電圧スイッチ（VSWITCH = 1）タイムアウト後にセット。（CPSMに影響なし）
CPSMACT	コマンド転送中です。（CPSMがアイドル状態ではない）

コマンドパスのエラー処理を、表279 に示します。

表279.コマンドパスのエラー処理

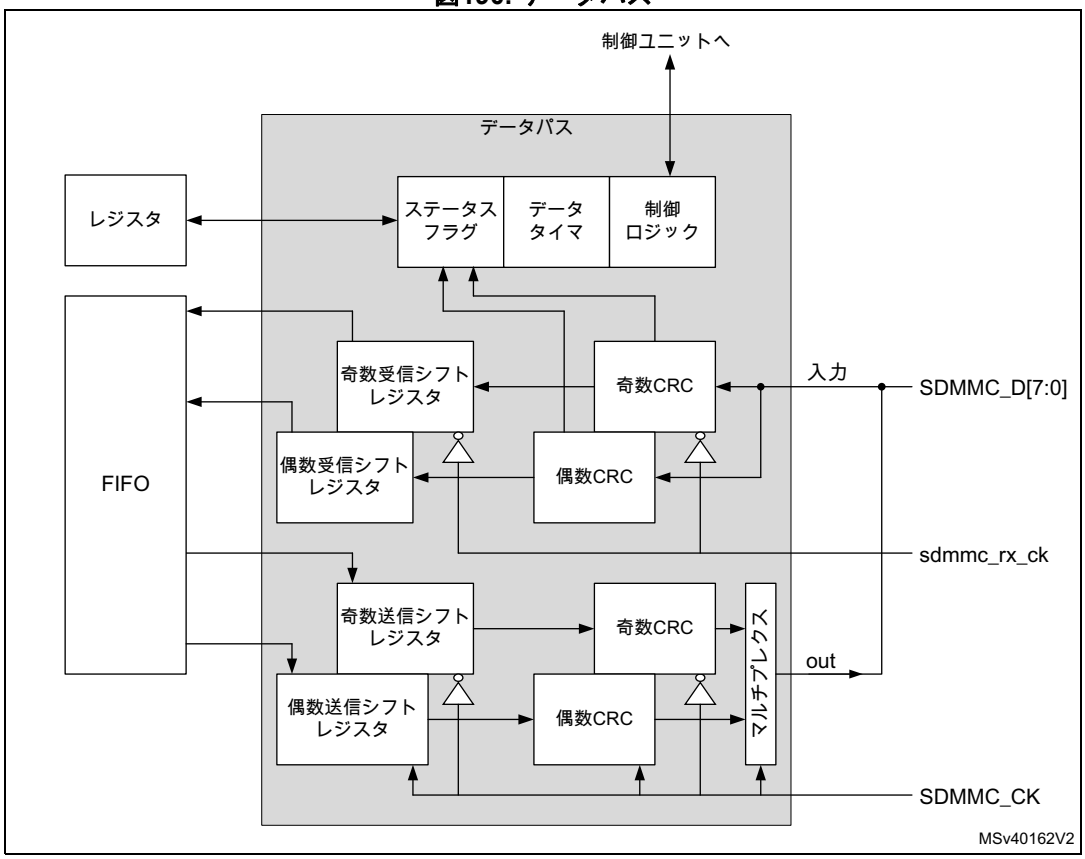
誤差	CPSMの状態	原因	カード動作	ホスト動作	CPSM動作
タイムアウト	ウェイト	時間内にスタートビットなし	不明	リセットまたはサイクルパワーカード(1)	アイドルに移行
CRCステータス	受信	否定ステータス	コマンド無視	コマンド再送信 (1)	アイドルに移行
		送信エラー	コマンド許可	コマンド再送信 (1)	

1. CMDTRANSをセットするとき、stop_transmissionコマンドも送信して、DPSMをアイドルに移行させる必要があります。

データパス

データパスサブユニットは、カードとの間でSDMMC_D[7:0] ラインでデータ転送を行います。データ送信パスはSDMMC_CKに基づいてクロック供給され、データをカードに送信します。データ受信パスはsdmmc_rx_ckに基づいてクロック供給され、カードからデータを受信します。図 190にデータパスブロック図を示します。

図190. データパス



カードデータバス幅は、クロック制御レジスタビットWIDBUSでプログラムできます。サポートされるデータバス幅モードは次のとおりです。

- ワイドバスモードが有効でない場合は、SDMMC_D0上で1bitだけが転送されます。
- 4bit幅バスモードが有効な場合、データはSDMMC_D[3:0] 上で、4bitで転送されます。
- 8bit幅バスモードが有効な場合、データはSDMMC_D[7:0] 上で、8bitで転送されます。

データバス幅の次に、データサンプリングモードはクロック制御レジスタビットDDRでプログラムできます。サポートされるデータサンプリングモードは次のとおりです。

- シングルデータレートシグナリング（SDR）では、データはクロックの立上がりエッジでクロック供給されます。
- ダブルデータレートシグナリング（DDR）では、データはクロックの両方のエッジでクロック供給されます。DDRモードはワイドバスモード（4bit幅および8bit幅）でのみサポートされます。

注： **データサンプリングモードは、SDMMC_D[7:0] ラインのみに適用されます。（SDMMC_CMDラインには適用できません）。**

DDRモードでは、次のルールに応じてSDMMC_CKの両方のエッジでデータがサンプリングされます。図 191 および図 192も参照してください。

- クロックの立上がりエッジで奇数バイトがサンプリングされます。
- クロックの立下がりエッジで偶数バイトがサンプリングされます。
- データのペイロードサイズは常に2バイトの倍数になります。
- 2つのCRC16がデータラインごとに計算されます。
  - クロックの立下がりエッジでクロック供給された奇数ビットCRC16
  - クロックの立上がりエッジでクロック供給された偶数ビットCRC16
- スタートビット、終了ビットおよびアイドル状態がフルサイクルになります。
- CRC ステータス／ブート確認応答およびビジーシグナリングがフルサイクルになり、クロックの立上がりエッジでのみサンプリングされます。

DDRモードでは、SDMMC_CKクロック分周は2以上でなければなりません。

図191. DDRモードのデータパケットクロック供給

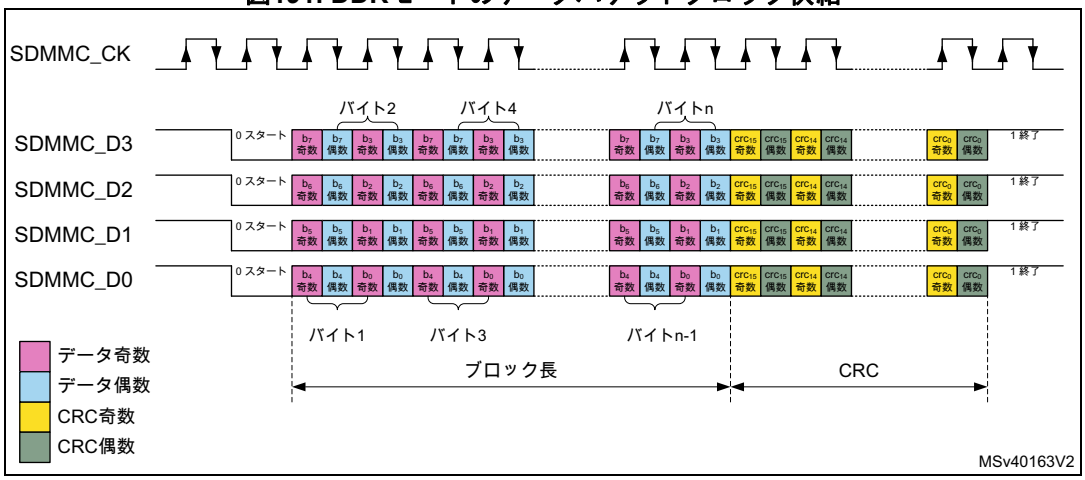
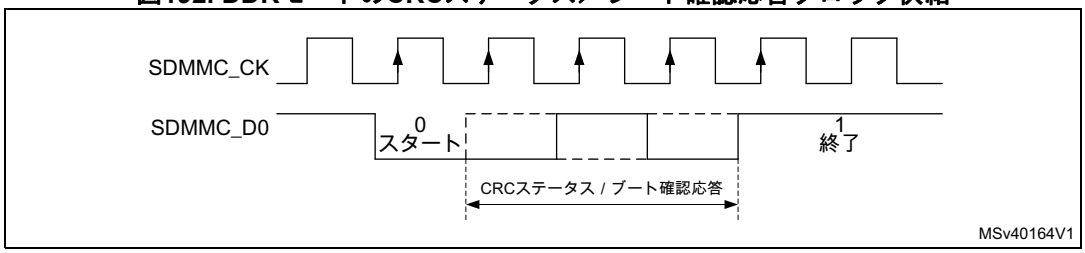


図192. DDRモードのCRCステータス／ブート確認応答クロック供給



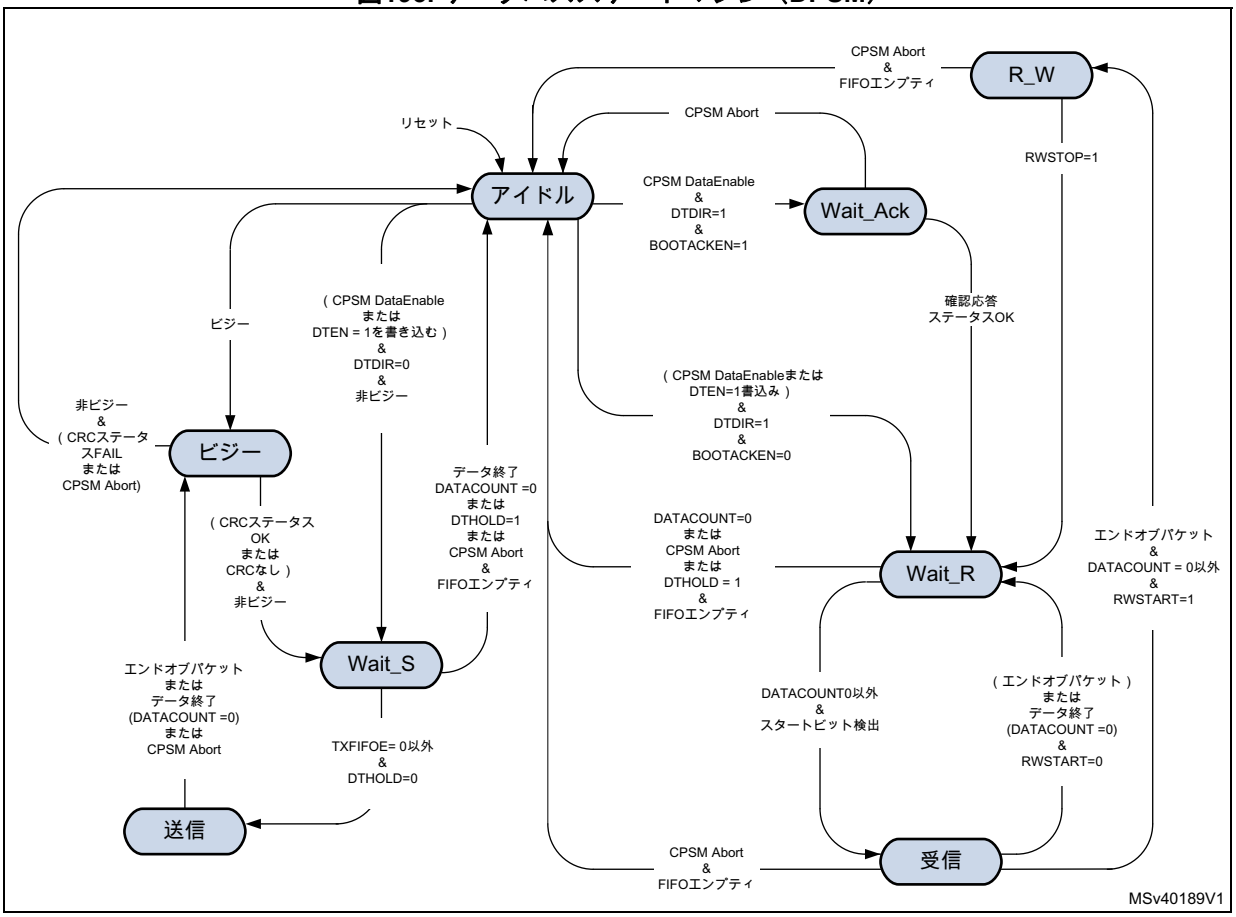
データパスステートマシン（DPSM）

転送の方向（送信または受信）によって、データパスステートマシン（DPSM）は、有効になったときにWait_SまたはWait_R状態へ移行します。

- 送信：DPSMはWait_S状態へ移行します。送信FIFOにデータがある場合、DPSMは送信状態へ移行し、データパスサブユニットはカードへのデータ転送を開始します。
- 受信：DPSMはWait_R状態へ移行して、スタートビットを待ちます。スタートビットを受信すると、DPSMは受信状態へ移行し、データパスサブユニットはカードからのデータ受信を開始します。
  確認応答ありのブート動作の場合、DPSMはWait_Ack状態に移行し、ブート確認応答を待ってからWait_R状態に移行します。

DPSMはSDMMC_CKで動作します。図 193に示すように、DPSMには、次の状態があります。アイドル状態ではない場合などDPSMがアクティブであるときは必ずDPSMACTビットがセットされます。

図193. データパスステートマシン（DPSM）



- **アイドル**状態：データパスはインアクティブであり、SDMMC_D[7:0] 出力はPWRCTRLの設定に従います。CMDTRANSビットをセットしてコマンドを送信するか、DTENビットをセットするか、またはSDMMC_D0でビジーを検出すること（R1bレスポンスありのコマンドの後）のどれかによってDPSMが有効化されます。

  ビジーではない場合、DPSMはSDMMC_CKクロック（節電PWRSAVビットのために停止している場合）を有効化し、新しい（DATALENGTH）の値でデータカウンタをロードします。そして、

  – データ方向ビット（DTDIR）が送信を示している場合、Wait_Sに移行します。

– データ方向ビット（DTDIR）が受信を示している場合、次のように移行します。
- BOOTACKENレジスタビットがクリアされている場合はWait_Rになります。
- BOOTACKENレジスタビットがセットされている場合は、Wait_Ackとなり、確認応答タイムアウトが開始します。

ビジーの場合、DPSMはSDMMC_CKクロックをアクティブなままとし、ビジー状態に移行します。

注： **DTENはSD、SDIOおよびeMMCカードでデータ転送の開始に使用してはなりません。**

- **Wait_Ack**状態：データパスがブート確認応答トークンを待ちます。
  - DPSMがタイムアウト前にエラーなし確認応答を受信した場合､Wait_R状態に移行します。
  - 確認応答とは異なるパターンを受信した場合、確認応答ステータスエラーが生成され、確認応答失敗ステータスフラグ（ACKFAIL）がセットされます。DPSMは、Wait_Ackのままとなります。
  - スタートビットを検出する前にタイムアウト（ACKTIME）に達した場合、タイムアウトステータスフラグ（ACKTIMEOUT）をセットします。DPSMは、Wait_Ackのままとなります。
  - CPSM Abort信号がセットされている場合、アイドル状態へ移行し、DABORTフラグをセットします。
- **Wait_R**状態：データカウンタがゼロでなく、データが保持されない場合、データパスはSDMMC_D[n:0] 上のスタートビットを待ちます。データカウンタがゼロで、データを保持している場合、FIFOが空になるのを待ちます。
  - ブロックモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DBLOCKSIZEでデータブロックカウンタをロードします。
  - SDIO　マルチバイトモードでは、タイムアウトまでにスタートビットを受信した場合、DPSMは受信状態に移行し、DATALENGTHでデータブロックカウンタをロードします。
  - ストリームモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DATALENGTHでデータカウンタをロードします。
  - データカウンタ（DATACOUNT）がゼロ（データの終了）に等しい場合、DPSMは受信FIFOが空になると、アイドル状態へ移行し、DATAENDフラグがセットされます。
  - スタートビットを検出する前にタイムアウト（DATATIME）に達した場合、タイムアウトステータスフラグ（DTIMEOUT）をセットし、DPSMはWait_R状態のままとなります。
  - CPSM Abort信号がセットされている場合：
    DATACOUNT > 0の場合、FIFOが空になったときとIDMAEN = 0がFIFORSTでリセットされたときに、DPSMはアイドル状態に移行し、DABORTフラグがセットされます。
    DATACOUNTがゼロの場合、通常動作を続け、転送が正常に完了するため、DABORTフラグはセットされません。
  - DTHOLDビットがセットされている場合：
    - DATACOUNT > 0の場合、受信FIFOが空になったときとIDMAEN = 0がFIFORSTでリセットされたときに、DPSMはアイドル状態に移行し、DHOLDフラグが発行されます。保留状態ではタイムアウトは無効になります。保留状態でCPSM Abort信号を受信すると、転送はアボートされます。
    - DATACOUNT = 0の場合、転送は正常に完了し、DHOLDフラグは発行されません。
  - DPSM が DTEN で開始した場合、エラー（DTIMEOUT）後、FIFO が空になったときと、IDMAEN = 0がFIFORSTでリセットされたときに、DPSMはアイドル状態に移行します。
- **R_W**状態：データパスがバスの読出しを待ちます。
  - 読出しウェイト停止ビット（RWSTOP）がセットされると、DPSM は Wait_R 状態に移行し、受信タイムアウトを開始します。
  - CPSM Abort信号がセットされ、FIFOが空になり、IDMAEN = 0の場合、FIFORSTでリセットされるのを待つ場合、アイドル状態に移行し、DABORTフラグがセットされます。

- **受信**状態 : データパスがカードからシリアルデータを受信します。データは数バイトにまとめられ、データFIFOに書き込まれます。データ制御レジスタで選択された転送モード（DTMODE）に応じて、データ転送はブロックモードまたはストリームモードのいずれかとなります。
  - ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を受信すると、DPSMはCRCコードを受信するまで待ちます。
  - SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を受信すると、DPSMはCRCコードを受信するまで待ちます。

  a) 受信したCRCコードが内部生成されたCRCコードと一致する場合、DPSMは次の状態へ移行します。
  - RWSTART = 1かつDATACOUNT > 0の場合、R_W状態となり、DBCKENDフラグがセットされます。
  - そうでない場合、Wait_R状態になります。

  b) 受信したCRCコードが内部生成されたCRCコードと一致しない場合、追加のデータ受信は阻止されます。
  - すべてのデータを受信していない場合（DATACOUNT > 0）、CRC失敗ステータスフラグ（DCRCFAIL）をセットして、DPSMは受信状態のままとなります。
  - すべてのデータを受信した場合（DATACOUNT = 0）、FIFOが空になるのを待ってから、CRC失敗ステータスフラグ（DCRCFAIL）をセットして、DPSMはアイドル状態に移行します。

  - ストリームモードでは、DPSMはデータカウンタDATACOUNT > 0の間、データを受信します。カウンタがゼロになると、シフトレジスタ内の残りのデータがデータFIFOに書き込まれ、DPSMはWait_R状態へ移行します。
  - FIFO オーバーランエラーが発生した場合、DPSM は FIFO オーバーランエラーフラグ（RXOVERR）をセットして、追加のデータ受信は阻止されます。DPSMは受信状態のままとなります。
  - CPSM Abort信号を受信した場合 :
    - DATACOUNT = 0でデータの最後の2bitまでにCPSM Abort信号を受信した場合、転送がアボートされます。シフトレジスタに残っているデータがデータFIFOに書き込まれ、FIFOが空になって、IDMAEN = 0の場合、FIFORSTでリセットされるのを待ってから、DPSMはアイドル状態に移行し、DABORTフラグがセットされます。
    - DATACOUNT = 0で転送の最後の2bit処理中または以降にCPSM Abort信号を受信した場合、転送は正常に完了します。DPSMは受信状態のままとなり、DABORTフラグは生成されません。
  - DPSMがDTENで開始した場合、エラー（DATACOUNT > 0時のDCRCFAILまたはRXOVERR）後、FIFOが空になったときと、IDMAEN = 0がFIFORSTでリセットされたときに、DPSMはアイドル状態に移行します。

- **Wait_S**状態 : データパスがFIFOからのデータが使用できるようになるのを待ちます。
  - データカウンタDATACOUNT > 0の場合、データFIFOエンプティフラグ（TXFIFOE）がネゲートされるまで待ってから、DTHOLDはセットされず、送信状態へ移行します。
  - データカウンタ（DATACOUNT）が0の場合、DPSMはアイドル状態に移行します。
    - DTHOLDが無効である場合、DATAENDフラグがセットされます。
    - DTHOLDが有効である場合、DHOLDフラグがセットされます。
  - DTHOLDがセットされ、DATACOUNT > 0の場合
    - IDMAが有効である場合、DBCKENDフラグがセットされ、続けてFIFOが一掃され、さらにDPSMはアイドル状態に移行し、DHOLDフラグがセットされます。
    - IDMAが無効である場合、DBCKENDフラグがセットされます。FIFOがソフトウェアによってFIFORSTでリセットされるのを待って、DPSMはアイドル状態に移行し、DHOLDフラグを発行します。
  - DTHOLDがセットされ、DATACOUNT = 0の場合、転送は正常に完了します。

– CPSM Abort信号を受信した場合
- CPSM Abort信号をDATACOUNT = 0でデータの最後の2bitまでに受信した場合、転送はアボートされ、FIFOが空になるまで、IDMAEN = 0の場合はFIFORSTでリセットされるまで待ってから、DPSMはアイドル状態に移行し、DABORTフラグがセットされます。
- DATACOUNT = 0で転送の最後の2bit処理中または以降にCPSM Abort信号を受信した場合、通常動作を続け、転送が正常に完了するため、DABORTフラグはセットされません。

注： **DPSMは、N_{WR} タイミングの要件を満たすため、少なくとも2クロック周期の間、Wait_S状態を保ちます。ここで、N_{WR} は、カードレスポンスの受信とホストからのデータ転送の開始との間のクロックサイクル数です。**

- **送信**状態：DPSM はカードへのデータの送信を開始します。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモード、SDIOマルチバイトモードまたはストリームモードのいずれかとなります。
  – ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を送信すると、DPSMは内部生成されたCRCコードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を送信すると、DPSMは内部生成されたCRCコードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – ストリームモードでは、DPSMはデータカウンタDATACOUNT > 0の間、カードにデータを送信します。データカウンタが0に達した場合、ビジー状態に移行し、送信タイムアウトを開始します。
  DATACOUNTに従って最後のストリームバイトを送信する前に、DPSMは 送信CMD信号にトリガを発行します。この信号は、CPSMが何らかのペンディングコマンドを送信するために使用します。（例：CMD12送信停止コマンド）
  – FIFO アンダーランエラーが発生した場合、DPSM は FIFO アンダーランエラーフラグ（TXUNDERR）をセットします。DPSMは送信状態のままとなります。
  – CPSM Abort信号を受信した場合
  - DATACOUNT = 0で転送の最後の2bitまでにCPSM Abort信号を受信した場合、転送はアボートされます。DPSMは終了ビットの前に最後のデータビットを送信します。FIFOは無効化/一掃され、DPSMはビジー状態に移行し、ビジーではなくなるのを待ってからDABORTフラグをセットします。
  - DATACOUNT = 0で転送の最後の2bit処理中または以降にCPSM Abort信号を受信した場合、転送は正常に完了し、DABORTフラグはセットされません。
- **ビジー**状態：DPSMは期待されている場合にCRCステータストークンを待って、ビジー以外の信号を待ちます。
  – CRCステータストークンが期待され、「エラーなし送信」を示す場合や期待されたCRCがない場合：
  – SDMMC_D0がローでない（カードがビジーでない）場合はWait_S状態へ移行します。
  - カードがビジーで、SDMMC_D0がローの場合、ビジー状態のままとなります。
  – CRCステータストークンが期待され、「エラーあり送信」を示す場合
  - すべてのデータが送信されていない場合（DATACOUNT > 0）DPSMは、ビジー以外になるまで待ってから、CRC失敗ステータスフラグ（DCRCFAIL）がセットされます。FIFOは無効化/一掃され、DPSMはビジー状態のままとなります。
  - すべてのデータが送信された場合（DATACOUNT = 0）DPSMは、ビジー以外になるまで待ってから、CRC失敗ステータスフラグ（DCRCFAIL）がセットされ、DPSMはアイドル状態に移行します。
  – DPSM がビジー状態の間に CRC ステータス（Ncrc）タイムアウトが発生した場合には、DPSMはデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。

- DPSM がビジー状態の間にビジータイムアウトが発生した場合には、DPSM はデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。
- ビジー状態でCPSM Abort信号を受信した場合：
  - DATACOUNT > 0でCRCレスポンスの最後の2bitまでにCPSM Abort信号を受信した場合、データ転送はアボートされます。DPSMは、ビジー以外になり、FIFOが無効化／一掃されるまで待ってから、アイドル状態に移行して、DABORTフラグがセットされます。
  - DATACOUNT = 0のとき、またはCRCが期待されていなくて、DATACOUNT = 0で、DTIMEOUTエラーがないときに、CRCレスポンスの最後の2bit処理中または以降にCPSM Abort信号を受信した場合、転送は正常に完了できるので、DPSMはビジー状態のままで、DABORTフラグは生成されません。
  - DTIMEOUTエラーが発生したときにCPSM Abort信号を受信した場合、DPSMは、ビジーではなくなるまで、そしてFIFOが無効化/一掃されるまで待ってから、アイドル状態に移行し、DABORTフラグがセットされます。
- 送信状態でのアボートによってビジー状態に移行したとき、DPSM は、ビジーではなくなるまで待ってから、アイドル状態に移行し、DABORTフラグがセットされます。
- DPSMがDTENで開始した場合、エラー（DATACOUNT > 0時のDCRCFAILまたはDTIMEOUT）後、FIFOがリセットされたときに、DPSMはアイドル状態に移行します。
- SDMMC_D0 でビジーになっているため、DPSM が開始した場合、ビジー以外になるまで待ってから、ビジー終了ステータスフラグ（BUSYD0END）がセットされ、DPSMはアイドル状態に移行します。

データタイマ（DATATIME）は、DPSMがWait_Rまたはビジー状態のときに、データブロック終了ビットまたはデータ読出しコマンド終了ビット、又はR1bレスポンス後の2サイクル有効になり、以下の場合にデータタイムアウトエラー（DTIMEOUT）を生成します。

- データ送信時にタイムアウトが発生します。
  - CRCステータスが期待されており、8 SDMMC_CKサイクル内で受信するスタートビットがない場合、DTIMEOUTフラグがセットされます。
  - ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUTフラグがセットされます。
- データ受信時にタイムアウトが発生します。
  - DATACOUNT > 0で受信するデータがまだあり、プログラムされたタイムアウト期間の前に受信するスタートビットがない場合、DTIMEOUTフラグがセットされます。
- R1bレスポンス後に、タイムアウトが発生します。
  - ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUTフラグがセットされます。

DATATIME = 0の場合

- 受信時、スタートビットはデータブロック終了ビットまたはデータ読出しコマンド終了ビットの2サイクル後に、存在する必要があります。
- 送信時、CRCトークン終了ビットまたはストリームデータ終了ビットの後2サイクルで、ビジーがタイムアウトします。
- R1bレスポンス後、レスポンス終了ビットの後2サイクルで、ビジーがタイムアウトします。

データは、カードからホスト（送信）へ、またはホストからカード（受信）へ転送できます。データは、SDMMC_Dnデータラインを介して転送され、FIFOに格納されます。

表280.データトークンフォーマット

説明	スタートビット	データ(1)	CRC 16	終了ビット	DTMODE
ブロックデータ	0	（DBLOCKSIZE、DATALENGTH）	あり	1	00
SDIOマルチバイト	0	（DATALENGTH）	あり	1	01
e•MMCストリーム	0	（DATALENGTH）	なし	1	10

1. 転送するデータの総数はDATALENGTHで指定します。ブロックデータの場合、各ブロックのデータの数はDBLOCKSIZEで指定します。

データトークンフォーマットは、適宜レジスタビットDTMODEで選択されます。

データパスは、表281に示すようにステータスフラグおよび関連付けられたクリアビットを実装します。

表281.データパスステータスフラグとクリアビット

フラグ		説明
DATAEND	TX	CRCがOKで、ビジーが終了し、DTHOLD = 0かつDATACOUNT = 0になったときにすべてのデータ転送終了時にセットされます。（DPSMはWait_Sからアイドルに移行）
	RX	CRC が OK で、すべてのデータを読み出したときにすべてのデータ転送終了時にセット。（DATACOUNT = 0かつFIFOが空）。（DPSMはWait_Rからアイドルに移行）
	ブート	
DCRCFAIL	TX	FAILで、ビジーが終了したときにCRC終了時にセット。（送信するデータがまだあり、CPSM Abortを待っている場合、DPSMはビジーのまま）（すべてのデータが送信された場合、またはDPSMがDTENで開始された場合、DPSMはビジーからアイドル状態に移行）
	RX	FAILで、FIFOが空のときにCRC終了時にセット。（受信するデータがまだあり、CPSM Abortを待っている場合、DPSMは受信のまま）（すべてのデータが受信された場合、またはDPSMがDTENで開始された場合、DPSMは受信からアイドル状態に移行）
	ブート	
ACKFAIL	ブート	失敗した場合に、ブート確認応答の終了時にセット。（DPSMはWait_Ackのままで、CPSM Abortを待機）
DTIMEOUT	CMD R1b	タイムアウト前に受信したビジー終了がないコマンドレスポンスの後にセット。（DPSMはビジーのままで、CPSM Abortを待機）
	TX	Ncrc内で受信したCRCトークンスタートビットがなかった場合、またはタイムアウト前に受信したビジー終了がなかった場合にセット。（DPSMはビジーのままで、CPSM Abortを待機）（DPSMがDTENで開始されたときは、アイドル状態に移行） 注：ビジータイムアウト前にCRCコマンド失敗時にDCRCFAILフラグもセットされる場合があります。
	RX	タイムアウト前に受信したスタートビットがないときにセット。（DPSMはWait_Rのままで、CPSM Abortを待機）（DPSMがDTENで開始されたときは、アイドル状態に移行）
	ブート	
ACKTIMEOUT	ブート	タイムアウト前に受信したスタートビットがないときにセット。（DPSMはWait_Ackのままで、CPSM Abortを待機）
DBCKEND	TX	DTHOLD = 1かつIDMAEN = 0の場合：CRCがOKで、ビジーが終了したとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSMはビジーからWait_Sに移行）
	RX	RWSTART = 1の場合：CRCがOKのとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSMは受信からR_Wに移行）
	ブート	

表281.データパスステータスフラグとクリアビット （続き）

フラグ		説明
DHOLD	TX	DTHOLD = 1の場合：CRCがOKで、ビジーが終了したときにデータブロック転送終了時にセット。（DPSMはWait_Sからアイドルに移行）
	RX	DTHOLD = 1の場合：CRCがOKで、すべてのデータが読み出された（FIFOが空になった）とき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSMはWait_Rからアイドルに移行）
DABORT	CMD R1b	CPSMによってCPSM Abortイベントが送信され、ビジーが終了したとき。（DPSMはビジーからアイドルに移行）
	TX	転送の最後の2bitまでにCPSMによってCPSM Abortイベントが送信されたとき。（DPSMは任意の状態からアイドルに移行）
	RX	
	ブート	
BUSYD0END	CMD R1b	タイムアウト前にビジー終了になったときのコマンドレスポンスの後にセット。（DPSMはビジーからアイドルに移行）
DPSMACT		データ転送中。（DPSMがアイドル状態ではない）

データパスエラー処理は、表282に示します。

表282.データパスエラー処理

誤差	DPSMの状態	原因	カード動作	ホスト動作	DPSM動作
タイムアウト	Wait_Ack	時間内に確認応答なし	不明	カードサイクルパワー	Wait_Ackのまま（RCC.SDMMCxRST レジスタビットでSDMMCをリセット）
	Wait_R	時間内にスタートビットなし	不明	データ受信の停止 送信停止コマンドの送信	CPSM Abortでアイドルに移行
			不明	ブート手順の停止	
	ビジー	長すぎるビジー（データ転送による）	不明	データ受信の停止 送信停止コマンドの送信	
		長すぎるビジー（R1bによる）	不明	リセットコマンドの送信	
CRC	受信	送信エラー	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM Abortでアイドルに移行
CRCステータス	ビジー	否定ステータス	追加データの無視	データ送信の停止 送信停止コマンドの送信	CPSM Abortでアイドルに移行
		送信エラー	追加データを待つ		
確認応答ステータス	Wait_Ack	送信エラー	ブートデータの送信	ブート手順の停止	CPSM Abortでアイドルに移行
オーバーラン	受信	FIFOフル	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM Abortでアイドルに移行
アンダーラン	送信	FIFOエンプティ	追加データの受信	データ送信の停止 送信停止コマンドの送信	CPSM Abortでアイドルに移行

データFIFO

データFIFO（ファーストインファーストアウト）サブユニットには、送受信データバッファが含まれます。シングルFIFOは、DTDIRビットで選択されたとおり送信または受信に使用されます。FIFOには、32bit幅、16 ワードの深さのデータバッファと、制御ロジックがあります。データFIFOはAHBクロック･ドメイン（sdmmc_hclk）で動作するため、SDMMCクロック･ドメイン（SDMMC_CK/sdmmc_rx_ck）のサブユニットからの信号はすべて再同期されます。

FIFOは次の状態のいずれかになり得ます。

– 送信FIFOは、カードにデータを送信すると、送信ロジックとデータバッファを参照します。（DTDIR = 0）

– 受信FIFOは、カードからデータを受信すると、受信ロジックとデータバッファを参照します。（DTDIR = 1）

正常に完了したFIFOからのSDMMCデータ転送の終了は、データパスサブユニットによって駆動されるDATAENDフラグで示されます。FIFOからの誤った（アボートされた）SDMMCデータ転送は、データパスサブユニットによって駆動されるエラーフラグ（DCRCFAIL、DTIMEOUT、DABORT）のいずれかか、FIFO制御によって駆動されるFIFOエラーフラグ（TXUNDERR、RXOVERR）のいずれかで示されます。

データFIFOには、次の方法でアクセスすることができます。表283を参照してください。

表283.データFIFOアクセス

データFIFOアクセス	IDMAEN
AHBスレーブインタフェース経由でファームウェアから	0
AHBマスタインタフェース経由でIDMAから	1

送信FIFO：

データは、DPSMが有効化されたとき（DPSMACT = 1）に、送信FIFOに書き込むことができます。

IDMAEN = 1の場合、FIFOはIDMAによって完全に処理されます。

IDMAEN = 0の場合、FIFOはAHBスレーブインタフェース経由でファームウェアによって制御されます。送信FIFOは、連続アドレスを介してアクセス可能です。送信FIFOには、読出しポインタが示すデータワードを保持するデータ出力レジスタが含まれています。データパスサブユニットがシフトレジスタへロードすると、読出しポインタがインクリメントされ、新しいデータが出力されます。送信FIFOは、次の方法で処理されます。

1. データ長はDATALENGTHに、ブロック長はDBLOCKSIZEに書き込みます。
   – ブロックデータ転送（DTMODE = 0）の場合、DATALENGTHはDBLOCKSIZEの整数倍でなければなりません。
2. 送信モードでSDMMCをセットします（DTDIR = 0）。
   – 送信モードでFIFOを設定します。
3. 次のいずれかでデータ転送を有効化します。
   – CMDTRANSビットをセットしてCPSMからコマンドを送信
   – DTENビットをセット
4. （DPSMACT = 1）の場合、データをFIFOに書き込みます。
   – DPSMは、FIFOがフル（TXFIFOF = 1）またはDATALENGTHで示された数字になるまでWait_S状態のままとなります。
   - SDMMCは、FIFOが空でない限りデータを送信し続けます。データ転送中のハードウェアフロー制御が、FIFOアンダーランを防ぐために使用されます。

5. データをFIFOに書き込みます。
   – FIFOをソフトウェアで処理する場合、FIFOがハーフエンプティ（TXFIFOHEフラグ）になるまで待って、FIFOがフル（TXFIFOF = 1）になるまで、または最後のデータが書き込まれるまでデータをFIFOに書き込みます。
   – FIFOをIDMAで処理する場合、IDMAがFIFOデータを転送します。
6. 最後のデータが書き込まれると、データの終了（DATAENDフラグ）を待ちます。
   – SDMMCがすべてのデータを完全に送信して、DPSMが無効になります（DPSMACT = 0）。

IDMAEN = 0のとき、データ転送エラーまたは 転送保留が発生した場合、ファームウェアはFIFOへの書込みを停止して、FIFORSTレジスタビットでFIFOを一掃してリセットする必要があります。

送信FIFOステータスフラグを表284に示します。

表284.送信FIFOステータスフラグ

フラグ	説明
TXFIFOF	すべての送信FIFOワードが有効なデータを含んでいるとき、ハイにセットされます。
TXFIFOE	送信FIFOが有効なデータを含んでいないとき、ハイにセットされます。
TXFIFOHE	半分以上の送信FIFOワードが空のとき、ハイにセットされます。
TXUNDERR	アンダーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMCクリアレジスタへの書込みによってクリアされます。

受信FIFO：

データは、DPSMが有効になっているとき（DPSMACT = 1）に、受信FIFOから読み出すことができます。

IDMAEN = 1の場合、FIFOはIDMAによって完全に処理されます。

IDMAEN = 0の場合、FIFOはAHBスレーブインタフェース経由で ファームウェア によって制御されます。データパスサブユニットがワードデータを受信すると、書込みデータバス上にデータが出力されます。書込み操作が完了すると、書込みポインタがインクリメントされます。読出し側では、読出しポインタの現在値が示すFIFOワードの内容が、読出しバス上に出力されます。受信FIFOは、連続アドレスを介してアクセス可能です。

受信FIFOは、次の方法で処理されます。

1. データ長はDATALENGTHに、ブロック長はDBLOCKSIZEに書き込みます。
   – ブロックデータ転送（DTMODE = 0）の場合、DATALENGTHはDBLOCKSIZEの整数倍でなければなりません。
2. 受信モードでSDMMCをセットします（DTDIR = 1）。
   – 受信モードでFIFOを設定します。
3. 次のいずれかでDPSM転送を有効化します。
   – CMDTRANSビットをセットしてCPSMからコマンドを送信
   – DTENビットをセット
4. DPSMACT = 1の場合、FIFOはデータを受信する準備ができています。
   – DPSMは受信したデータをFIFOに書き込みます。
   - SDMMCは、FIFOがフルでない限りデータを受信し続けます。データ転送中のハードウェアフロー制御が、FIFOオーバーランを防ぐために使用されます。

5. FIFOからデータを読出します。
   – FIFOをソフトウェアで処理する場合、FIFOがハーフフル（RXFIFOHFフラグ）になるまで待って、FIFOがエンプティ（RXFIFOE = 1）になるまでFIFOからデータを読み出します。-最後のデータを受信すると、FIFOがエンプティ（DATAEND = 1）になるまでFIFOからデータを読み出します。
   – FIFOをIDMAで処理する場合、IDMAがFIFOデータを転送します。
6. SDMMCがすべてのデータを完全に受信して、DPSMが無効になります（DPSMACT = 0）。

IDMAEN = 0のとき、データ転送保留が発生した場合、ファームウェアはFIFOがエンプティになるまで残りのデータを読み出し、FIFOをFIFORSTレジスタビットでリセットする必要があります。これにより、DPSMはアイドル状態（DPSMACT = 0）に移行します。

IDMAEN = 0のとき、データ転送エラーが発生した場合、ファームウェアはFIFOの読出しを停止し、FIFORSTレジスタビットでFIFOを一掃してリセットする必要があります。これにより、DPSMはアイドル状態（DPSMACT = 0）に移行します。

受信FIFOステータスフラグを表285に示します。

表285.受信FIFOステータスフラグ

フラグ	説明
RXFIFOF	すべての受信FIFOワードが有効なデータを含んでいるとき、ハイにセットされます。
RXFIFOE	受信FIFOが有効なデータを含んでいないとき、ハイにセットされます。
RXFIFOHF	半分以上の受信FIFOワードが有効なデータを含んでいるとき、ハイにセットされます。
RXOVERR	オーバーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMCクリアレジスタへの書込みによってクリアされます。

CLKMUXユニット

CLKMUXは、受信したデータおよびコマンドレスポンスと使用するクロックのソースsdmmc_rx_ckを選択します。受信データのクロックソースは、次のうちのクロック制御レジスタビットSELCLKRXで選択できます。

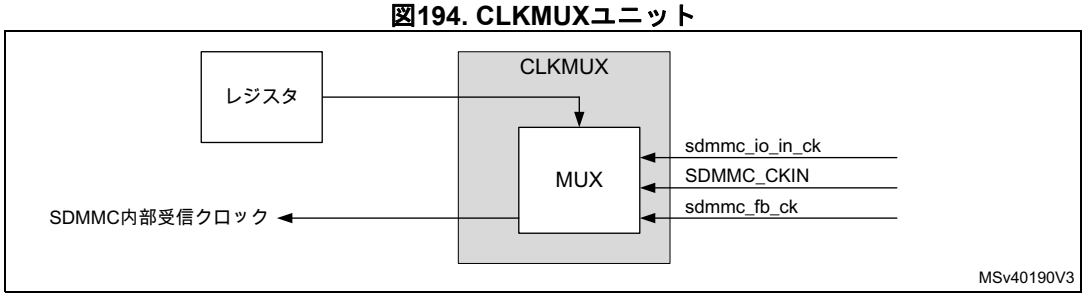
- sdmmc_io_in_ckバスマスタメインフィードバッククロック
- SDMMC_CKIN外部バスフィードバッククロック
- sdmmc_fb_ckバス調整フィードバッククロック

sdmmc_io_in_ckは、外部ドライバがない場合にDSおよびハイスピードで選択されます。

SDMMC_CKINは、外部ドライバがある場合にSDR12、SDR25、SDR50およびDDR50で選択されます。

デバイスのDLYBブロックがSDR104、HS200で使用され、オプションでSDR50モードおよびDDR50モードで使用されている場合、sdmmc_fb_ckクロック入力を選択しなければなりません。

図194. CLKMUXユニット



sdmmc_rx_ckソースの変更は、CPSMおよびDPSMがアイドル状態のときに行う必要があります。

31.5.5 SDMMC AHBスレーブインタフェース

AHBスレーブインタフェースは、割込みリクエストを生成し、SDMMCアダプタレジスタとデータFIFOにアクセスし、データパス、レジスタデコーダ、および割込みロジックで構成されています。

SDMMC FIFO

FIFOのアクセスは、ワードアクセスのみに制限されます。

- 送信FIFOモードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32bit）でFIFOに書き込まれます。DATALENGTHが4の整数倍ではない場合、最後に残ったデータ（1、2、または3バイト）は1ワード転送で書き込まれます。
- 受信FIFOモードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32bit）でFIFOから読み出されます。DATALENGTHが4の整数倍ではない場合、最後に残ったデータ（1、2、または3バイト）は0値バイトでパディングされた1ワード転送で書き込まれます。

ハーフワードまたはバイトアクセスでFIFOにアクセスすると、AHBバスフォールトが生成されます。

SDMMC割込み

割込みロジックは、マスクされていないステータスフラグの少なくとも1つがアクティブのときにアサートされる割込みリクエスト信号を生成します。割込みを生成する条件を選択できるように、マスクレジスタが提供されています。ステータスフラグは、対応するマスクフラグがセットされている場合に割込みリクエストを生成します。一部のステータスフラグは、クリアレジスタで間接的なクリアが必要です。

31.5.6 SDMMC AHBマスタインタフェース

AHBマスタインタフェースは、SDMMC IDMAを使用したメモリとFIFOの間でのデータ転送に使用されます。

SDMMC IDMA

DMA（Direct Memory Access）は、SDMMC FIFOとメモリの間でハイスピード転送を行うために使用します。AHBマスタはシステムバスのバンド幅を最適化します。SDMMC内部DMA（IDMA）には、送信または受信に使用するチャネルが1つ搭載されています。

IDMAは、IDMAENビットで有効になり、8 ビートのバースト転送をサポートします。

- 送信バースト転送モードの場合：
  - データは、DATALENGTHに応じたすべてのデータが転送されるまで、FIFOがバースト転送の数に対してエンプティであればいつでもメモリからバーストでフェッチされます。
    DATALENGTHがバーストサイズの整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。
    DATALENGTHが4の整数倍ではない場合、最後に残ったデータ（1、2、または3バイト）は1ワード転送でフェッチされます。
- 受信バースト転送モードの場合：
  - データは、DATALENGTHに応じたすべてのデータが転送されるまで、FIFOにバースト転送の数がある場合に限りメモリにバーストで格納されます。
    DATALENGTHがバースト転送の整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。
    DATALENGTHが4の整数倍ではない場合、最後に残ったデータ（1、2、または3バイト）はハーフワード転送またはバイト転送で格納されます。

さらに、IDMAには、IDMABMODEで選択される次のチャネル設定があります。

- シングルバッファチャネル
- リンクリストのチャネル

シングルバッファチャネル

シングルバッファ設定では、メモリ側のデータにベースアドレスIDMABASE から始まる直線的な方法でアクセスします。IDMAがすべてのデータの転送を終了し、DPSMが転送を完了したときに、DATAENDフラグがセットされます。

リンクリストのチャネル

リンクリスト設定では（IDMAMODE = 1）メモリ側のデータは、ベースアドレス（IDMABASE）にあるリンクバッファからの連続したアクセスになります。メモリバッファのサイズは、IDMABSIZEによって定義されます。バッファサイズは、バーストサイズの整数倍でなければなりません。ビットULAは、リンクリストの新しいバッファ設定を、リンクリストのテーブルからロードする必要があるかどうかを示すために使用されます。現在のリンクリスト項目のULAビットがセットされていると、リンクリストの新しい設定がロードされます。

最初のリンクリスト項目の設定は、ファームウェアでSDMMCレジスタに直接プログラムします。

IDMAでは、IDMABSIZEに従って1つのリンクリストバッファのすべてのデータの転送を終了し、リンクリスト項目のULAビットが設定されていると、リンクリストのテーブルから新たなリンクリスト項目をロードし、次のリンクリストバッファからデータの転送を継続します。IDMAがIDMABSIZEとULAに従ってすべてのデータの転送を終了し、DPSMがDATALENGHTに従って転送を完了したときに、DATAENDフラグがセットされます。

以下の場合、リンクリストによって、転送するデータより多くのバッファスペースが用意されており、現在のリンクリストバッファのデータが完全に転送されていないことになります。

- ULAビットがセットされていて、DATALENGTHに従ったすべてのSDMMCデータが転送されてしまっている（DATAENDフラグ）。
- 転送エラー（DATACOUNT > 0の場合のDCRCFAIL、RXOVERR、TXUNDERR）が発生している。
- 転送が保留されている（DTHOLD）

上記のすべての場合、IDMAのリンクリストは停止され、FIFOは一掃/リセットされます。新しいSDMMC転送を開始または再開する前に、ソフトウェアは正しいIDMABASEおよびIDMABSIZEを使用して新たなリンクリストを初期化する必要があります。

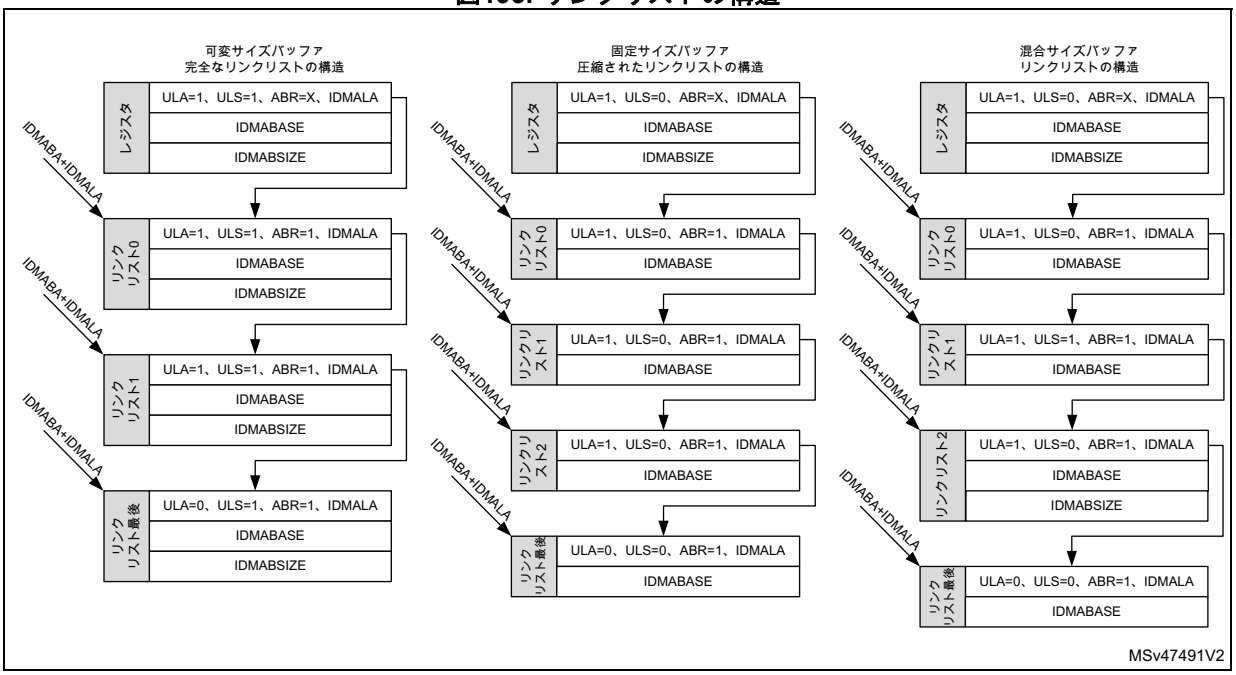
IDMA転送エラーが発生した場合（セクション ：IDMA転送エラー管理を参照）、またはリンクリストに十分なバッファスペースが用意されていない場合、

- リンクリストはULA = 0で終了し、最後のリンクリストバッファのデータはすべて転送されていますが、DATALENGTHに従ったすべてのSDMMCデータが転送されている訳ではありません。SDMMCの転送は停止し、IDMA転送エラーが生成されます（セクション ：IDMA転送エラー管理を参照）。

特定のリンクリスト項目のベースアドレスは、リンクリストのベースIDMABAレジスタ値とリンクリストのオフセットIDMALAレジスタ値を加算した値によって与えられます。

各リンクリスト項目の内容はULSビットで指定でき、これによってオプションでIDMABSIZEをロードできるようになり、3ワードのリンクリスト構造になります。IDMABSIZEがロードされない（つまり、固定サイズのバッファの）場合、IDMABASER値とIDMALAR値のみを含む、圧縮された縮小2ワードのリンクリスト構造を使用することができます。

図195. リンクリストの構造



両方のリンクリストの項目の構造を1つのリストに混在させることに制限はありません。これにより、必要な場合にのみIDMABSIZEを更新できます。

リンクリストのバッファが転送され、現在のバッファのULA = 1のときはいつでも、end-of-linked-list-buffer-transfer-complete（リンクリスト終端のバッファ転送完了）割込み（IDMABTC）が生成されることがあります（割込みが有効な場合）。

リンクリストの確認応答

SDMMC転送中に、ソフトウェアがリンクリストを動的に更新する場合、確認応答バッファレディ（ABR）ビットによって新たなリンクリストバッファの利用可能性を確認できます。

ABRにより新たなリンクリストのバッファの準備ができたことが確認応答されると、IDMAは新たなリンクリストのバッファからのデータの転送を続行します。

ABRにより新たなリンクリストのバッファの準備ができていないことが示されると、IDMA転送エラーが生成されます（セクション ：IDMA転送エラー管理 を参照）。いつIDMA転送エラーが発生するかによって、通常はTXUNDERRまたはRXOVERRエラーが生成されます。リンクリストのバッファが時間内に確認応答されない場合、SDMMC転送は停止します。

ソフトウェアでプログラムされたレジスタ情報からリンクリストを開始する場合、ABR情報は無視されます。最初のリンクリストのバッファは、SDMMC転送を開始する前に準備完了にしておく必要があります。

IDMA転送エラー管理

IDMA転送エラーは次の場合に発生します。

- 予約済みアドレス空間の読取りまたは書込み時（データまたはリンクリストの情報に関して）。
- 受信したSDMMCデータを保存するためのリンクリストのバッファ空間がなくなった場合。
- すべてのリンクリストのバッファのデータが転送されたが、さらにまだSDMMCデータを送信する必要がある場合。
- リンクリストのバッファの使用可能性が確認応答されない場合。

IDMA転送エラー時、その後のIDMA転送は無効になり、IDMATEフラグがセットされ、ハードウェアフロー制御は無効になります。いつIDMA転送エラーが発生するかによって、通常はTXUNDERRまたはRXOVERRエラーが生成されます。

IDMATEフラグの動作は、IDMA転送エラーがSDMMC転送中にいつ発生したかに依存します。

- IDMA転送エラーは、あらゆるSDMMC転送エラー（TXUNDERR、RXOVERR、DCRCFAIL、またはDTIMEOUT）の前に検出されます。
  - IDMATEフラグは、SDMMC転送エラーフラグと同時にセットされます。
  - TXUNDERR、RXOVERR、DCRCFAIL、またはDTIMEOUT割込みが生成されます。
- IDMA転送エラーは、STOP_TRANSNMISSIONコマンド中に検出されます。
  - IDMATEフラグは、DABORTフラグと同時にセットされます。
  - DABORT割込みが生成されます。
- IDMA転送エラーは、SDMMC転送（DHOLDまたはDATAEND）の終了後に検出されます。
  - IDMATEフラグは、SDMMC転送の終了時にセットされます。
  - SDMMC転送終了割込みが生成され、DHOLDまたはDATAENDフラグがセットされます。

IDMATEは、他のSDMMC転送割込み（TXUNDERR、RXOVERR、DCRCFAIL、DTIMEOUT、DABORT、DHOLD、またはDATAEND）で生成されます。

31.5.7 AHBとSDMMC_CKのクロック関係

AHBは、少なくともSDMMCバス帯域幅の3倍以上にする必要があります。すなわち、SDR50 4bitモード（50MB/秒）の場合、sdmmc_hclkの最低周波数は37.5MHz（150MB/秒）とします。

表286.AHBとSDMMC_CKのクロック周波数の関係

SDMMCバスモード	SDMMCバス幅	最大SDMMC_CK [MHz]	最小AHBクロック [MHz]
e•MMC DS	8	26	19.5
e•MMC HS	8	52	39
e•MMC DDR52	8	52	78
e•MMC HS200	8	200	150
SD DS / SDR12	4	25	9.4
SDハイスピード / SDR25	4	50	18.8
SD DDR50	4	50	37.5
SD SDR50	4	100	37.5
SD SDR104	4	208	78

31.6 カード機能詳細

31.6.1 SD I/Oモード

以下の機能は、SDMMC固有の操作です。

- SDIO割込み
- SDIOサスペンド／レジューム操作（書込みおよび読出しサスペンド）
- クロックの停止によるSDIO読出しウェイト操作
- SDMMC_D2信号によるSDIO読出しウェイト操作

表287.SDIO特殊動作制御

動作モード	SDIOEN	RWMOD	RWSTOP	RWSTART	DTDIR
割込み検出	1	X	X	X	X
サスペンド／レジューム操作	X	X	X	X	X
読出しウェイトSDMMC_CKクロック停止（START）	X	1	0	1	1
読出しウェイトSDMMC_CKクロック停止（STOP）	X	1	1	1	1
読出しウェイトSDMMC_D2信号（START）	X	0	0	1	1
読出しウェイトSDMMC_D2信号（STOP）	X	0	1	1	1

SD I/O割込み

SDインタフェースのピン8（4bitモードでSDMMC_D1と共有）には、SD I/Oカードからホストへの割込みを可能にする割込み機能が用意されています。割込みの使用は、各カードまたはカード内の各機能でオプションとなっています。SD I/O割込みはレベル対応であるため、割込みラインはホストによって認識され、割込みが処理されるか、割込み期間の終了によってネゲートされるまで、アクティブ（ロー）に保たれる必要があります。ホストが割込み処理を終えると、SD I/Oカードの内部レジスタの該当するビットへのI/O書込みによって割込みステータスビットがクリアされます。すべてのSD I/Oカードの割込み出力はアクティブローであり、アプリケーションはすべてのデータライン（SDMMC_D[3:0]）上に、外部プルアップ抵抗を提供する必要があります。

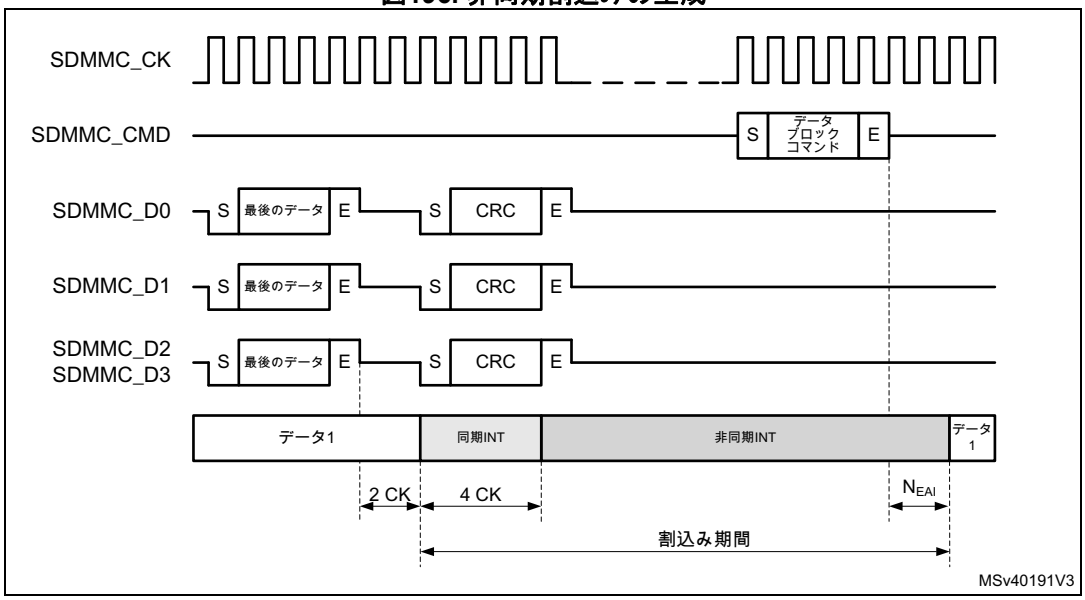
SD 1bitモードでは、ピン8が割込み機能（IRQ）専用となり、割込みに関するタイミングの制約がありません。

SD 4bitモードでは、ホストは、割込み期間に限って、割込み検出器にピン8（SDMMC_D1/IRQ）のレベルをサンプリングします。その他の状況では、ホスト割込みはこの値を無視します。割込み期間は、割込みがカードで有効化され、SDIOENビットがセットされた際に開始します。表287のレジスタ設定を参照してください。

4bitモードでは、カードはカードCCCRレジスタSAIおよびEAIビットで示されたとおりに同期割込みまたは非同期割込みを生成できます。

- 同期割込みは、SDMMC_CKをアクティブにする必要があります。
- 非同期割込みは、SDMMC_CKが最後のデータブロック後のカード割込み期間の開始後4サイクル停止したときに生成できます。

図196. 非同期割込みの生成



割込み期間のタイミングは、バス速度モードに依存します。

レジスタビットBUSSPEEDで選択するDS、ハイスピード、SDR12、SDR25のモードでは、割込み期間がSDクロックと同期します。

- 割込み期間は、データブロックを転送するコマンド（CMDTRANSビットをセットして送信されたコマンド）の終了ビット以降の次のクロック、またはDTENビットがセットされたときに終了します。
- 割込み期間は、データブロックの完了後に2 SDMMC_CKで再開します。
- データブロックの差で、割込み期間は2 SDMMC_CKサイクルに制限されます。

注： **DTENはSDおよびeMMCカードでデータ転送の開始に使用してはなりません。**

図197. 同期割込み期間のデータ読出し

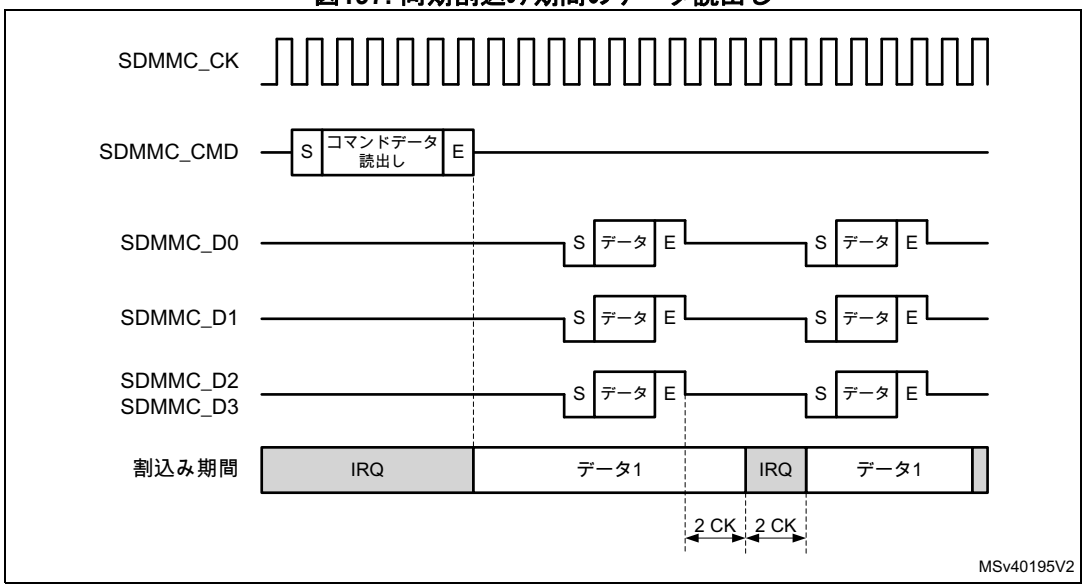
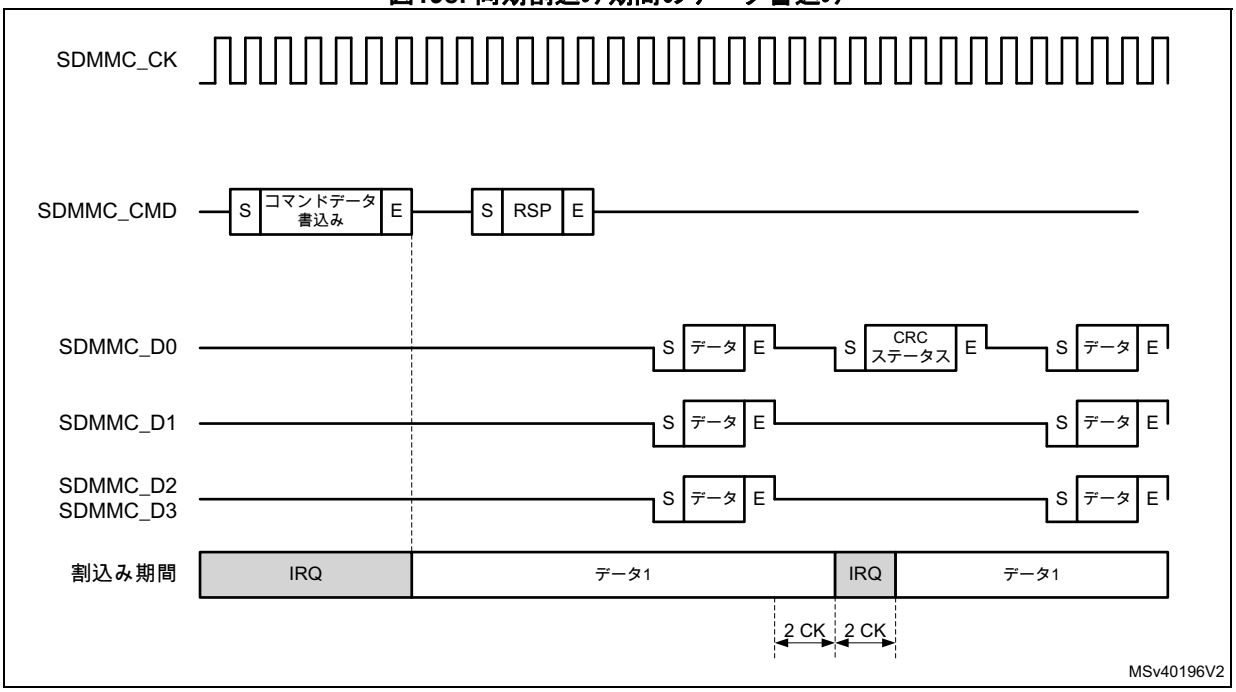


図198. 同期割込み期間のデータ書込み



レジスタビットBUSSPEEDで選択するSDR50、SDR104、およびDDR50では、カードからホストへの伝搬遅延によって、割込み期間は非同期になります。

- カードの割込み期間は、データブロックを転送するコマンド（CMDTRANSビットをセットして送信されたコマンド）の終了ビット以降の0～2 SDMMC_CKサイクル後、またはDTENビットがセットされたときに終了します。ホストでは、割込み期間はデータブロックを転送するコマンドの終了ビット以降に終了します。コマンド終了ビット後1～2サイクルで発行されたカードの割込みは、この割込み期間中ホストによって検出されません。
- カードの割込み期間は、最後のデータブロックの完了後に2～4 SDMMC_CKで再開します。ホストは、最後のデータブロック後に常に2サイクルで割込み期間を再開します。
- データブロックの間で割込み期間はありません。

注： **DTENはSDおよびeMMCカードでデータ転送の開始に使用してはなりません。**

図199. 非同期割込み期間のデータ読出し

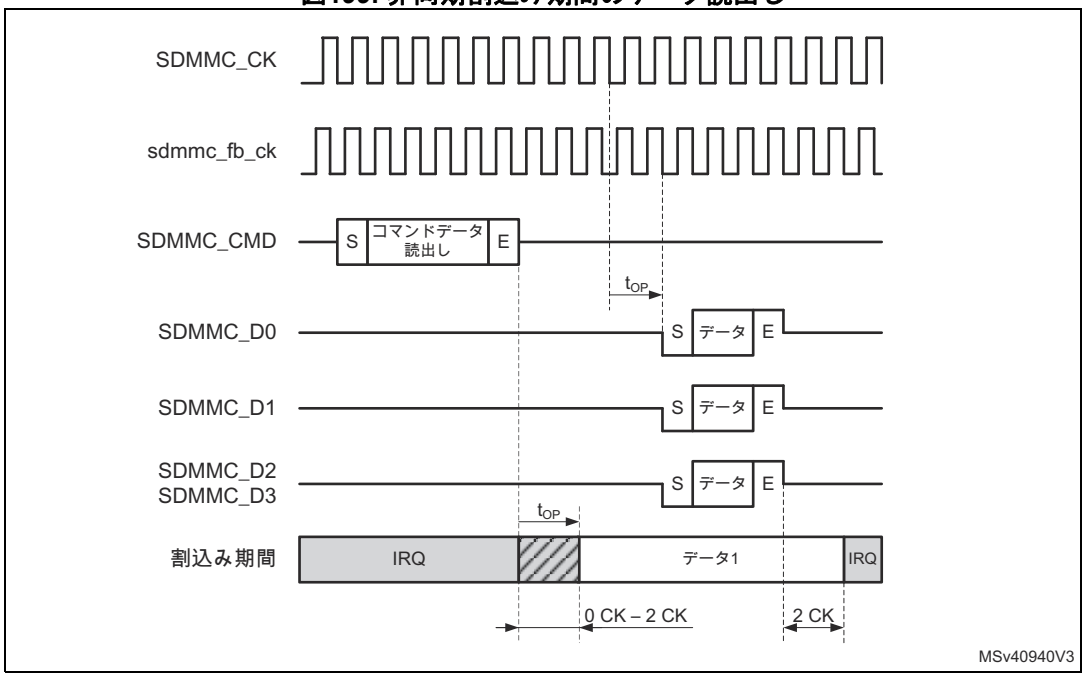
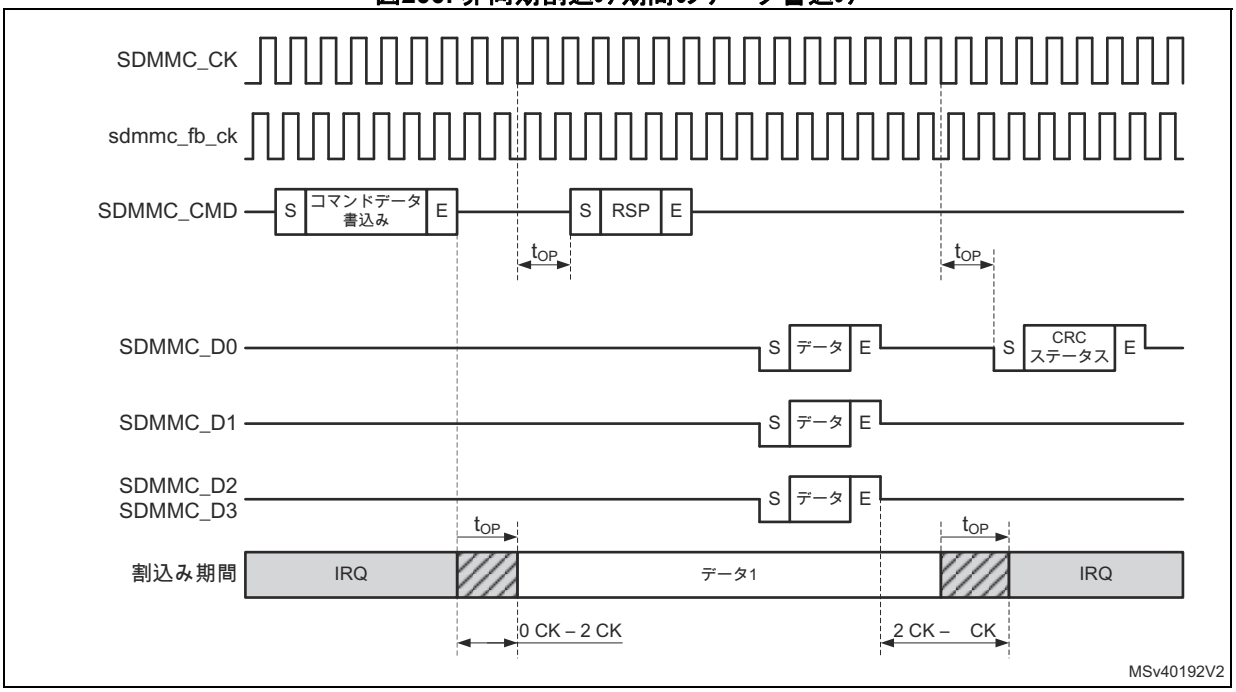


図200. 非同期割込み期間のデータ書込み



オープンエンド複数ブロックデータを転送し、DTMODE「STOP_TRANSMISSIONコマンドで終わるブロックデータ転送」を使用する場合、SDMMCは最後のデータブロックからCMD12 STOP_TRANSMISSIONコマンドの終了まで割込み期間をマスクします。

割込み期間は、メモリとI/Oの両方の操作に適用されます。

4bitモードでは、表288に従って他の信号と区別できます。

表288.4bitモードのスタート、割込み、CRCステータスの信号検出

SDMMCデータライン	開始	割込み	CRCステータス
SDMMC_D0	0	1またはCRCステータス	0
SDMMC_D1	0	0	X
SDMMC_D2	0	1または 読出しウェイト	X
SDMMC_D3	0	1	X

SD I/Oのサスペンドとレジューム

この機能は、SDIOバージョン4.00以降ではサポートされていません。

マルチファンクションSD I/OまたはI/Oとメモリの両方の機能を持つカードでは、*e*•MMC/SDバスへのアクセスを共有する複数のデバイス（I/Oとメモリ）があります。複数デバイス間でホストへのアクセスを共有するため、SD I/Oおよびコンボカードはオプションでサスペンド／レジュームの概念を実装します。カードがサスペンド／レジュームをサポートするときには、ホストは1つの機能またはメモリへのデータ転送を一時的に中止（サスペンド）して、別の機能またはメモリへの、より優先順位の高い転送用にバスを開放できます。この優先順位の高い転送が完了すると、本来の転送がアボートした所から再開（レジューム）されます。

バス上でサスペンド／レジューム操作を行うには、ホストは以下の手順を実行します。

1. SDMMC_D[3:0] ラインを現在使用している機能を特定します。
2. 優先順位が低いトランザクションや時間のかかるトランザクションをサスペンドするように要求します。
3. トランザクションのサスペンドの完了を待ちます。
4. 優先順位の高いトランザクションを開始します。
5. 優先順位の高いトランザクションの完了を待ちます。
6. サスペンドされたトランザクションをリストアします。

サスペンドコマンドを受信したカードは、現在のバスステータスで応答します。バスがカードによってサスペンドされていた場合のみ、バスステータスがサスペンドを完了したことを示します。

次のような個別のサスペンドケースの条件があります。

- データ転送開始前に許可したサスペンドリクエスト。
- （データが同時に転送されたため）許可されなかったサスペンドリクエスト。ホストは許可されるまでリクエストをチェックし続けます。（データ転送がサスペンドしました）。
- 書込みビジー中のサスペンドリクエスト。
- 複数書込み中のサスペンドリクエスト。
- 読出しウェイト中のサスペンドリクエスト。

ホストでバスが開放されたかどうか確認するためには、サスペンドが完了したというサスペンドリクエストのステータスを確認する必要があります。

サスペンドリクエストレスポンスのバスステータスがサスペンド完了を示すと、カードがバスを開放します。このとき、サスペンドされた操作の状態を保存しなければならず、その後は、他の操作を開始できます。

サスペンドコマンドはCMDSUSPENDビットをセットして送信する必要があります。これにより、バスがサスペンドされた（レスポンスビットBS = 0）ときに、サスペンドコマンドレスポンスの後に割込み期間を開始できます。

ハードウェアは、サスペンド操作を再開させるときに転送する残りのデータの数を保存しません。転送されたデータを判定して、正しい残りのデータバイト数で再開することはファームウェア次第となります。

カードからデータを受信している間、SDMMCは読出しデータブロック終了（Wait_RのDPSM）以後の読出し操作をサスペンドできます。カードからサスペンド確認応答レスポンスを受信した後、ファームウェアで次の手順を実行する必要があります。

1. 通常の受信プロセスは、DTHOLDビットをセットすることで停止する必要があります。
   a) FIFOの残りのデータバイト数を、受信FIFOがエンプティになる（RXFIFOEフラグがセットされる）まで読み出し、IDMAEN = 0の場合は、FIFOをFIFORSTでリセットする必要があります。
2. すべてのデータがFIFOから読み出されたことと、サスペンドが完了したことの確認は、DHOLDフラグによって示されます。
   a) 動作再開時に読み出されるべき残りのデータバイト数（データブロックの倍数）は、DATACOUNTによって示される残りのバイト数から判定する必要があります。

注： **サスペンド手順中にDTIMEOUTフラグが発生した場合、これは無視する必要があります。**

カードからデータの受信を再開するには、ファームウェアで次の手順を実行する必要があります。

1. 残りのデータバイト数（データブロックの倍数）をDATALENGTHにプログラムする必要があります。
2. DPSMを、DTDIRビットでデータを受信するように設定する必要があります。
3. CMDTRANSビットとCMDSUSPENDビットをセットして、レジュームコマンドをCPSMから送信する必要があります。これによって、データ転送が再開されたとき（レスポンスビットDF = 1）に割込み期間が終了し、DPSMが有効になり、その後、カードがデータ送信を再開します。

カードにデータを送信している間、SDMMCは書込みデータブロックCRCステータス終了（ビジーのDPSM）以後の書込み操作をサスペンドできます。カードにサスペンドコマンドを送信する前に、ファームウェアで次の手順を実行する必要があります。

1. DHOLDフラグ（IDMAEN = 0の場合はDBCKENDフラグも）を有効にします。
2. DTHOLDをセットすることで､DPSMが新しいデータブロックの送信を開始できないようにする必要があります。
3. IDMAEN = 0の場合：DBCKENDフラグを受信すると、データ転送が停止します。ファームウェアがFIFOへの入力を停止でき、その後FIFOはFIFORSTでリセットする必要があります。FIFOに残っているバイトは、操作再開時に再書込みする必要があります。
4. DHOLDフラグを受信すると、データ転送が停止します。再開時に書き込まれるべき残りのデータバイト数は、DATACOUNTによって示される残りのバイト数から判定する必要があります。
5. カードをサスペンドするために、CMDSUSPEND　ビットをセットしてサスペンドコマンドをCPSMから送信する必要があります。これにより、バスがサスペンドされた（レスポンスビットBS = 0）ときに、サスペンドコマンドレスポンスの後に割込み期間を開始できます。

カードへのデータ送信を再開するには、ファームウェアで次の手順を実行する必要があります。

1. 残りのデータバイト数をDATALENGTHにプログラムする必要があります。
2. DTDIR をセットしてDPSM を送信用に設定し、CMDTRANS ビットとCMDSUSPEND ビットをセットしてCPSMにレジュームコマンドを送信させることでDPSMを有効にする必要があります。これにより、割込み期間が終了し、データ転送が開始されます。DPSMは、SDMMC_D0がビジーの信号を発していない場合Wait_S状態に移行し、ビジーの信号を発している場合はビジー状態に移行します。
3. IDMAEN=1の場合：残りのバイトを転送するためにIDMAを再プログラムする必要があります。
4. IDMAEN=0の場合：ファームウェアが残りのデータをFIFOに入力し始めます。

SD I/O読出しウェイト

ブロックのギャップの間にデータ転送を一時停止する2つの方法があります。

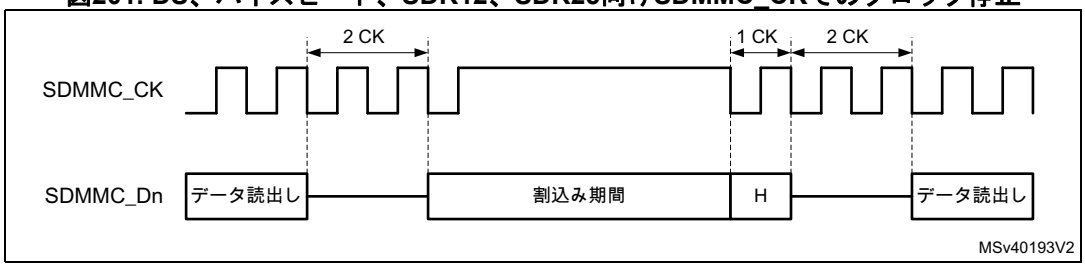
1. SDMMC_CKの停止
2. SDMMC_D2に読出しウェイト信号を使用

SDMMCは、表287 に従ったレジスタ設定で読出しウェイトを実行できます。

SDMMC動作モード（DS、ハイスピード、SDR12、SDR25）または（SDR50、SDR104、DDR）に応じて、各方法には異なる特性があります。

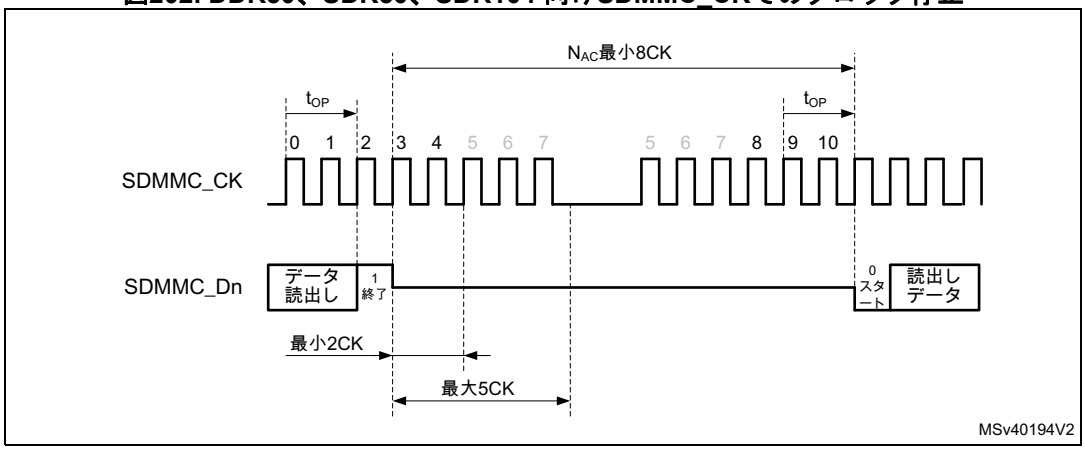
DS、ハイスピード、SDR12、SDR25の場合SDMMC_CKを停止することで読出し動作を一時停止するタイミングは、終了ビットの2 SDMMC_CKサイクル後に、SDMMC_CKを停止できます。準備が整うと、ホストはクロックを再開することで再開します。図 201を参照してください。

図201. DS、ハイスピード、SDR12、SDR25向けSDMMC_CKでのクロック停止



SDR50、SDR104、、DDR50の場合SDMMC_CKを停止することで読出し動作を一時停止するタイミングは、終了ビット後の最小2 SDMMC_CKサイクルおよび最大5 SDMMC_CKサイクルで、SDMMC_CKを停止できます。準備が整うと、ホストはクロックを再開することで再開します。図 202を参照してください。（DDR50モードでは、クロックラインがローの場合、SDMMC_CKは立下がりエッジの後でのみ停止する必要があります。）

図202. DDR50、SDR50、SDR104 向けSDMMC_CKでのクロック停止



ReadWait SDMMC_CKクロック停止では、RWSTARTがセットされている場合、DSPMは現在受信したデータブロックCRCの終了ビット後にクロックを停止します。RWSTOPビットに1が書き込まれた後、クロックは再開し、その後、DPSMはカードからのスタートビットを待ちます。

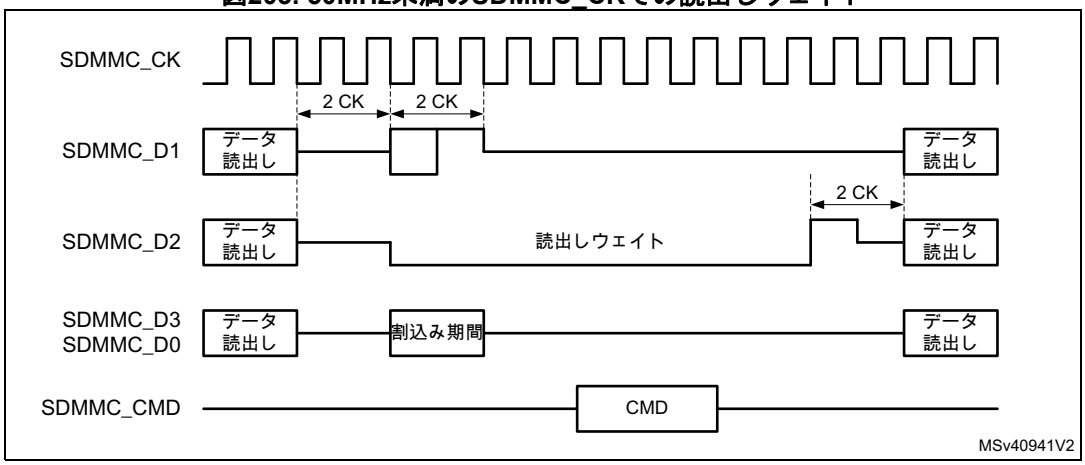
SDMMC_CKが停止されると、コマンドをカードに発行できません。読出しウェイトインターバル中、SDMMCはSDMMC_D1上のSDIO割込みを検出できます。

オプションのSDMMC_D2での読出しウェイト信号（RW）操作は、SDの1bitおよび4bitモードについてのみ定義されます。読出しウェイト操作によって、ホストは複数のレジスタ（IO_RW_EXTENDED、

CMD53）を読み出しているカードにデータ転送を一時的にストールするよう信号を送り、その間にホストはSD I/Oデバイス内の機能にコマンドを送信できます。カードが読出しウェイトプロトコルをサポートしているかどうかを判断するには、ホストは内部カードレジスタの機能ビットをテストする必要があります。

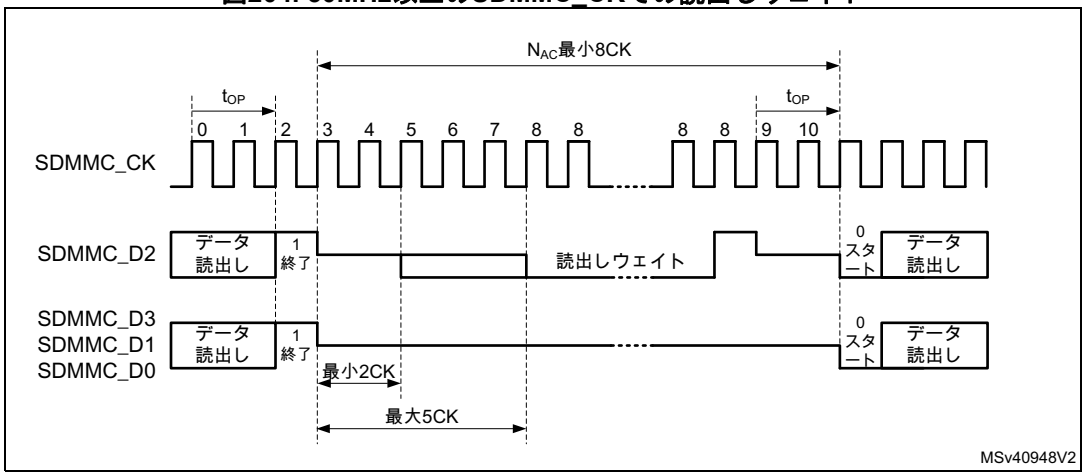
50MHz未満のSDMMC_CKでの読出しウェイトのタイミング（DS、ハイスピード、SDR12、SDR25）は、カードによってSDMMC_D1上に生成された割込み期間に基づきます。割込み期間中にSDMMC_D2をローでアサートすることで、ホストはカードに読出しウェイトに移行するようリクエストします。読出しウェイトを終了するには、ホストはSDMMC_D2をハイインピーダンスにする前の1 SDMMC_CKサイクル中にSDMMC_D2をハイにする必要があります。図 203を参照してください。

図203. 50MHz未満のSDMMC_CKでの読出しウェイト



50MHz以上のSDMMC_CKのSDR50、SDR104 およびDDR50の場合、カードはSDMMC_D2での読出しウェイトリクエストを非同期イベントとして処理します。最小2 SDMMC_CKサイクルおよび最大5 SDMMC_CKサイクルの後にSDMMC_D2をローでアサートすることで、ホストはカードに読出しウェイトに移行するようリクエストします。読出しウェイト を終了するには、ホストはSDMMC_D2をハイインピーダンスにする前の1 SDMMC_CKサイクル中にSDMMC_D2をハイにする必要があります。ホストはSDMMC_CKクロックに基づいてSDMMC_D2を立ち上げる必要があります（図 204 を参照してください）。

図204. 50MHz以上のSDMMC_CKでの読出しウェイト



ReadWait SDMMC_D2信号では、RWSTARTがセットされている場合、DSPMは現在受信したデータブロックCRCの終了ビット後にSDMMC_D2を駆動します。SDMMC_D2でのReadWait信号は、

RWSTOPビットに1が書き込まれると削除されます。DPSMは、さらに2 SDMMC_CKクロックサイクルの間、R_W状態にとどまり、1クロックサイクルでSDMMC_D2を1に（SDIO仕様に従って）駆動し、その後、DPSMはカードからのスタートビットを待ちます。

SDMMC_D2上で読出しウェイト信号がある間、コマンドをカードに発行できます。読出しウェイトインターバル中、SDMMCはSDMMC_D1上のSDIO割込みを検出できます。

31.6.2 CMD12送信タイミング

CMD12は、データ転送の停止／アボートに使用され、送信停止コマンドの終了ビット後2クロックサイクルでカードのデータ送信は終了します。

表289.CMD12の使用事例

データ操作	送信停止コマンドCMD12の説明
SDMMCストリーム書込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
SDMMCオープンエンド複数ブロックの書込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数でのSDMMCブロック書込み	この種類の複数ブロック書込み終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを受信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
SDMMCストリーム読出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
SDMMCオープンエンド複数ブロックの読出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数でのSDMMCブロック読出し	この種類の複数ブロック読出し終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを送信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） トランザクションは、送信停止コマンドを送信することでアボートできます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。

すべてのデータ書込みと読出しは、送信停止コマンドCMD12でいつでもアボートできます。データ転送進行中は、次のデータアボート手順が適用されます。

1. レジスタでCMD12送信停止コマンドをロードして、CMDSTOPビットをセットします。
   a) これにより、コマンドがDPSMに送信されると、CPSM Abort信号が生成されます。
2. 直ちにコマンドを送信するためにCPSMを設定します（WAITPENDをクリアします）。
   a) カードはデータを送信すると、送信停止コマンドの終了ビットの2サイクル後にデータ転送を停止します。
   データが送信されていない場合は、カードは新しいデータの送信を開始しません。
   b) ホストはデータを送信すると、送信停止コマンドの終了ビット後に、終了ビットが続く最後の1つのデータビットを送信します。
   データが送信されていない場合は、ホストは新しいデータの送信を開始しません。

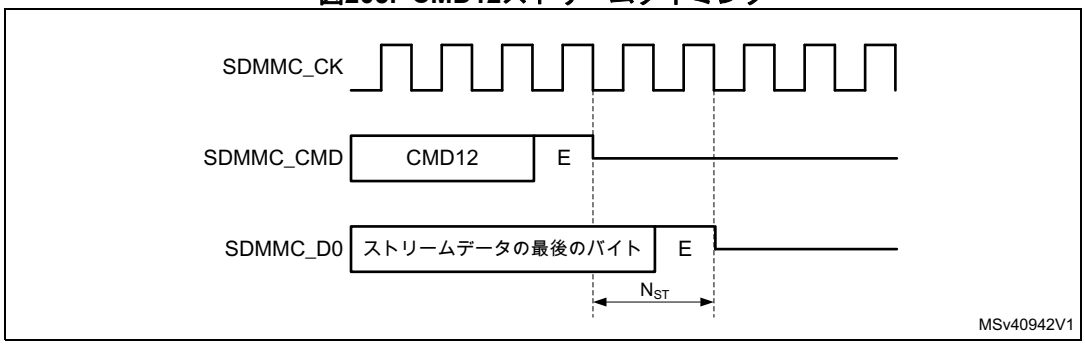
3. IDMAEN = 0の場合、FIFOはFIFORSTでリセットされる必要があります。
   a) カードにデータを書き込む場合。CMDRENDフラグでは、波長によりFIFOへのデータ書込みを停止する必要があります。その後、FIFOはFIFORSTでリセットする必要があり、これによりFIFOが一掃されます。
   b) カードからデータを読み出す場合。CMDREND　フラグでは、波長により残りのデータをFIFOから読み取る必要があります。その後、FIFOはFIFORSTでリセットする必要があります。
4. IDMAEN = 1の場合、ハードウェアがFIFOを管理します。
   a) カードにデータを書き込む場合。CPSM Abort信号で、ハードウェアはIDMAを停止し、その後、FIFOが一掃されます。
   b) カードからデータを読み出す場合。CPSM Abort信号で、ハードウェアはIDMAに、FIFOからRAMに残りのデータを転送するように命令します。
5. FIFOがエンプティになるか、リセットされると、DABORTフラグが生成されます。

ストリーム動作とCMD12

転送する最後のバイトの後にストリーム転送を停止するためには、CMD12終了ビットのタイミングを最後のバイトのデータ･ストリーム終了と合わせて送信する必要があります。次の書込みストリームデータ手順が適用されます。

1. DPSMで、DTMODE = MCCストリームデータ転送の場合、ストリームデータを初期化します。
2. CMDTRANS = 1でCPSMからWRITE_DATA_STREAMコマンドを送信します。
3. CMDSTOPビットをセットして、コマンドレジスタでCMD12をプリロードします。
4. CPSMを設定して、最後のデータ（DATALENGTHによる）のペンディング待機（WAITPEND = 1）終了後にのみコマンドを送信します。
5. CPSM を有効にして、STOP_TRANSMISSION コマンドを送信すると、ストリームデータ終了ビットとコマンド終了ビットが揃えられます。
   a) DATALENGTH > 5バイトの場合、データ転送終了ビットと合わせるためにCPSMでコマンドCMD12が待たされます。
   b) DATALENGTH < 5バイトの場合、コマンドCMD12が先に開始され、DPSMはデータ転送終了をCMD12の終了ビットと合わせるためにWait_S状態にとどまります。
6. 書込みストリームデータは、WAITPENDビットをクリアすることでいつでもアボートできます。これにより、プリロードされたCMD12が直ちに送信され、書込みデータ･ストリームが停止されます。

図205. CMD12ストリームタイミング



最後のバイトの後に読出しストリームの転送を停止するために、CMD12終了ビットのタイミングを最後のデータストリームバイトの後に発生させる必要があります。次の読出しストリームデータ手順が適用されます。

1. すべてのデータがDPSMによって受信され、FIFOから読み出されるのを待ちます（DATAENDフラグ）。
   a) カードがより多くのデータを送信しても、DPSMはDATALENGTHで示された以上のデータを受信しません。
2. CPSMでCMD12を送信します。
   a) CMD12によりカードのデータ送信が停止されます。

注： カードがデータを送信し続けても、DATACOUNT = 0の場合、SDMMCはカードからそれ以上のデータを受信しません。

ブロック動作とCMD12

データの終了時にブロック転送を停止するためには、最後のブロック終了ビットの後にCMD12終了ビットを送信する必要があります。

カードにデータを書き込んでいる場合、書込みデータブロックのCRCトークン終了ビットの後に、CMD12終了ビットを送信する必要があります。これには、データブロック送信タイミングと紐づいたCMD12送信が必要になります。オープンエンド複数ブロック書込みを停止するには、次の手順が適用されます。

1. データ転送開始前に、DTMODEを「STOP_TRANSMISSIONコマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータがDPSMによって送信され、CRCトークンが受信されるのを待ちます（DATAENDフラグ）。
   a) DPSMはDATALENGTHで示された以上のデータを送信しません。
3. CPSMでCMD12を送信します。
   a) CMD12によってカードはアイドルモードにセットされます。

カードからデータを読み出している場合、カード読出しデータブロックの最後のデータビットと同じタイミングで最も速くCMD12終了ビットを送信する必要があります。これには、データブロック受信タイミングと紐づいたCMD12送信が必要になります。以下のオープンエンド複数ブロックの読出しデータブロックの手順が適用されます。

1. データ転送開始前に、DTMODEを「STOP_TRANSMISSIONコマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータがDPSMによって受信され、FIFOから読み出されるのを待ちます（DATAENDフラグ）。
   a) カードがより多くのデータを送信しても、DPSMはDATALENGTHで示された以上のデータを受信しません。
3. CPSMでCMDSTOPビットをセットしてCMD12を送信します。
   a) CMD12によってカードの追加データ送信が停止され、カードがアイドルモードにセットされます。カードによって進行中のあらゆるブロック転送がアボートされます。

注： カードがデータを送信し続けても、DATACOUNT = 0の場合、SDMMCはカードからそれ以上のデータを受信しません。

31.6.3 SLEEP（CMD5）

e•MMCカードは、CMD5によってSLEEP状態とSTANDBY状態を切り替える場合があります。SLEEP状態では、カードの消費電力が最小限になり、Vcc電源がオフに切り替わります。

CMD5（SLEEP）は、STANDBY状態からSLEEP状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0をプルダウンします。カードがSDMMC_DOラインのプルダウンを停止すると、SLEEP状態に到達します。

カードをSLEEP状態にセットするには、次の手順が適用されます。

1. BUSYD0ENDの割込みを有効にします。
2. CMD5（SLEEP）を送信します。
3. BUSYD0END割込みで、カードがSLEEP状態になります。
4. Vcc電源をオフに切り替えられます。

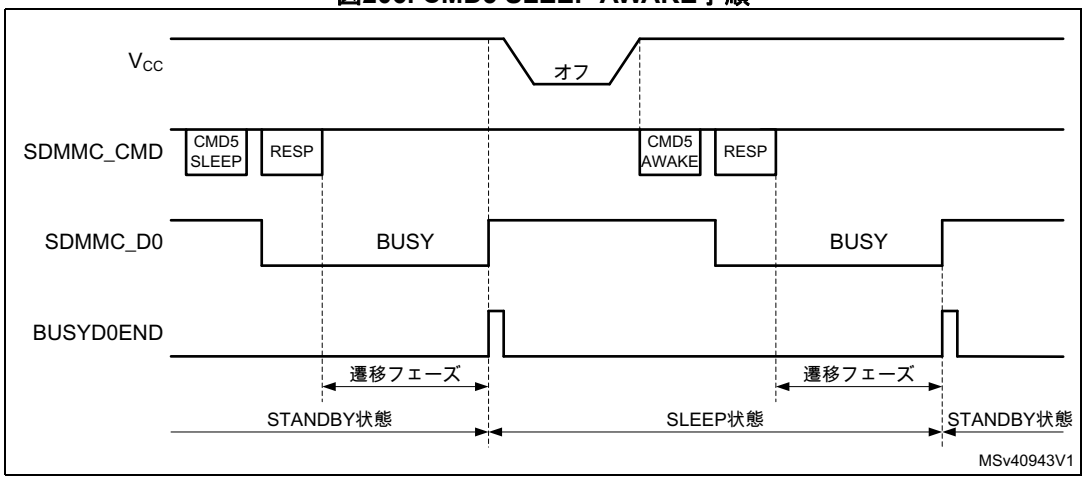
CMD5（AWAKE）は、SLEEP状態からSTANDBY状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0をプルダウンします。カードがSDMMC_DOラインのプルダウンを停止すると、STANDBY状態に到達します。

カードをSLEEP状態にセットするには、次の手順が適用されます。

1. Vcc電源をオンに切り替え、最低動作レベルに到達するまで待ちます。
2. BUSYD0ENDの割込みを有効にします。
3. CMD5（AWAKE）を送信します。
4. BUSYD0END割込みで、カードがSTANDBY状態になります。

SLEEP状態に到達した後でのみ、Vcc電源をオフに切り替えられます。Vcc電源は、CMD5（AWAKE）が送信される前に再起動する必要があります。

図206. CMD5 SLEEP AWAKE手順



31.6.4 割込みモード（Wait-IRQ）

ホストとカードは、同時に割込みモード（Wait-IRQ）の移行と終了を行います。割込みモードでは、データ転送はありません。許可されるメッセージは、カードまたはホストからの割込みサービスリクエストレスポンスだけです。割込みモードが正しく動作するために、オープンドレインモードで実現可能なSDMMC_CMDデータレート（これは静電容量負荷とプルアップ抵抗によって変わります）に従ってSDMMC_CK周波数をセットする必要があります。CLKDIVは >1にセットし、SETCLKRXではsdmmc_io_in_ckまたはSDMMC_CLKINソースを選択する必要があります。

ホストは、カードがSTANDBY状態であることをCMD40（GO_IRQ_STATE）を発行する前に確認する必要があります。割込みレスポンスを待っている間、SDMMC_CKクロック信号をアクティブに保つ必要があります。

割込みモード（IRQ状態）のカード：

- 内部カード割込みイベントを待っています。イベントが発生すると、カードは割込みサービスリクエストレスポンスの送信を開始します。レスポンスはオープンドレインモードで送信されます。
- 内部カード割込みイベントを待っている間、カードはSDMMC_CMDラインのスタートビットも監視します。スタートビットを検出すると、カードは割込みモードをアボートして、STANDBY状態に切り替わります。

割込みモードのホスト（割込みを待つCPSMウェイトステート）：

- カード割込みサービスリクエストレスポンス（スタートビット）を待っています。
- カード割込みサービスリクエストレスポンスを待っている間、ホストは（WAITINTレジスタビットをクリアすることで）割込みモードをアボートする場合があり、オープンドレインモードでRCA = 0x0000で割込みサービスリクエストレスポンスR5を送信します。

割込みサービスリクエストレスポンスを送信すると、送信者ビット単位がSDMMC_CMDビットストリームを監視します。割込みサービスリクエストレスポンスビットがSDMMC_CMDラインのビットに対応しない送信者は、送信を停止します。送信側が複数の場合、完全な割込みサービスリクエストレスポンスの送信に成功するのは1つだけです。ホストが同時に送信した場合、ホストは送信ビット後に送信ができなくなります。

割込みモードを処理するには、次の手順が適用されます。

1. オープンドレインモードで実現可能なSDMMC_CMDデータレートに従ってSDMMC_CK周波数をセットし、CLKDIVは >1にセットし、SETCLKRXはsdmmc_io_in_ckを選択する必要があります。
2. コマンドレジスタでCMD40（GO_IRQ_STATE）をロードします。
3. WAITINTレジスタビットをセットして、割込みの待機を有効にします。
4. 直ちにコマンドを送信するためにCPSMを設定します。
   a) これにより、CMD40が送信され、CPSMがウェイトステートで中止し、割込みサービスリクエストレスポンス待ちになります。
5. 割込みの待機状態（CPSMウェイトステート）を終了する方法：
   a) 割込みサービスリクエストレスポンスのスタートビットを検出すると、CPSM はレスポンスを受け取る受信状態に移行します。レスポンスの完全な受信は、CMDRENDまたはコマンドCRCエラーフラグによって示されます。
   b) 割込みモードをアボートするためには、ホストはWAITINTレジスタビットをクリアします。これにより、ホストは独自に割込みサービスリクエストレスポンスを送信します。これでCPSMは受信状態に移行します。レスポンスの完全な受信は、CMDRENDまたはコマンドCRCエラーフラグによって示されます。

注： **同時送信した割込みサービスリクエストレスポンスのスタートビットが衝突した場合、ホストは送信ビット後にバスアクセスを失います。**

31.6.5 ブート操作

ブート操作モードでは、ホストは2つのブート操作機能のうちいずれかでカードからブートデータを読み出せます。

1. 通常ブート（CMDラインがローのまま）
2. 代替ブート（CMD0を引数0xFFFFFFFAで送信）

ブートデータは、カードレジスタ設定に基づいて、次の設定オプションに応じて読み出すことができます。

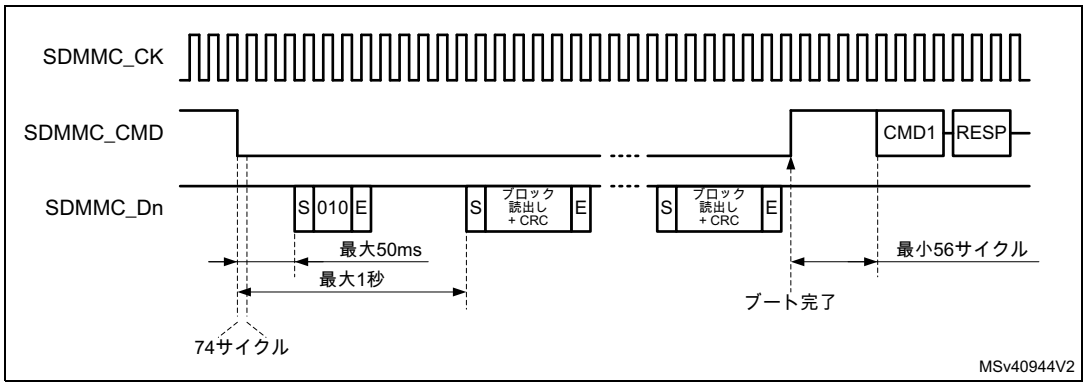
- ブートデータが読み出されるパーティション（EXT_CSD Byte[179]）
- ブートデータサイズ（EXT_CSD Byte[226]）
- ブート中のバス設定（EXT_CSD Byte[177]）
- カードからのブート確認応答の受信（EXT_CSD Byte[179]）

ブート確認応答が有効である場合、CMDラインがローに移行、またはCMD0を引数0xFFFFFFFAでブートモードがリクエストされた後の50ms以内に、カードがSDMMC_D0でパターン010を送信します。ブート確認応答タイムアウト（ACKTIMEOUT）と確認応答ステータス（ACKFAIL）が提供されます。

通常ブート動作

カードのパワーアップまたはリセットから最初のコマンドが発行されるまで最低74クロックサイクルの間、SDMMC_CMDラインがローで保たれる場合、カードはブートモードが開始されていると認識します。CMDラインがローになった後1秒以内に、カードはSDMMC_Dnラインで最初のブートコードデータの送信を開始します。ホストは、すべてのブートデータが読み出されるまでSDMMC_CMDラインをローのままにしておく必要があります。ホストは、SDMMC_CMDラインをハイにすることでブートモードを終了できます。

図207. 通常ブートモード動作



通常ブート手順を実行するには、次の手順が必要です。

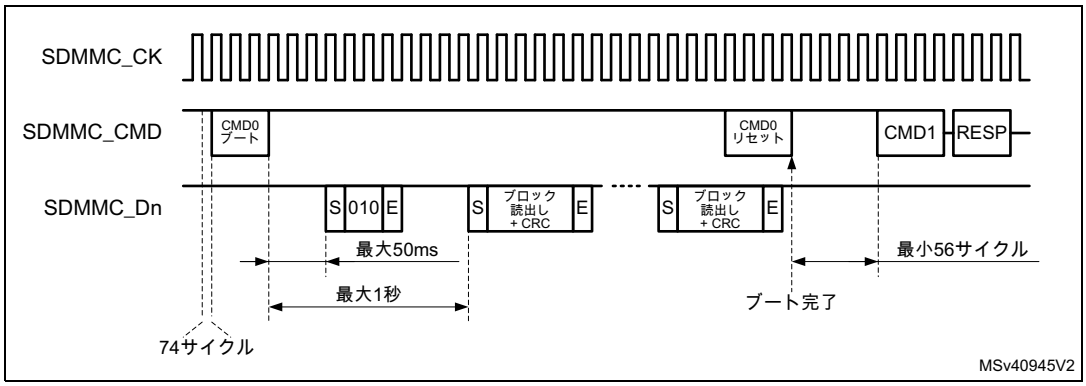
1. カードをリセットします。
2. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKFAILおよびACKTIMEOUT割込みを有効にします。
3. 受信モード（DTDIR）でDPSMとDATALENGTHで受信するデータバイト数をセットしてデータ受信を有効にします。
4. ブートコマンドの終了確認にDTIMEOUT、DATAEND、およびCMDSENT割込みを有効にします。

5. BOOTMODE で通常ブート動作モードを選択し、BOOTEN でブートを有効にします。ブート手順は、CPSMENでCPSMを有効にすることで開始されます。これにより次のことが発生します。
   – SDMMC_CMDがローに駆動されます。（BOOTMODE = 通常ブート）。
   – ACKタイムアウトが開始します。
   – DPSMが有効になります。
6. ブート確認応答の誤受信は、ACKFAILフラグまたはACKTIMEOUTフラグが有効時に検出できます。
   – 正しくないブート確認応答が受信された場合、ACKFAILフラグが発生します。
   – ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUTフラグが発生します。
7. すべてのブートデータが受信されると、DATAENDフラグが発生します。
   – データCRCが失敗した場合、DCRCFAILフラグも生成されます。
   – データタイムアウトが発生した場合、DTIMEOUTフラグも生成されます。
8. 最後のデータが受信されると、FIFOがエンプティになるまでFIFOからデータを読み出し、その後、データの終了DATAENDフラグが生成されます。
   – SDMMCがすべてのデータを完全に受信して、DPSMが無効になります。
9. ブート手順は、ファームウェアがBOOTENをクリアすることで終了し、これによりSDMMC_CMDラインがハイになります。CMDSENTフラグが56サイクル後に生成され、新しいコマンドを送信できることを示します。
   a) すべてのデータが受信される前にブート手順がファームウェアによってアボートされた場合、CPSM Abort信号によってデータ受信が停止され、DPSMが無効にされ、これにより、有効になっていればDABORTフラグがトリガされます。
10. CMDSENTフラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。

代替ブート動作

カードのパワーアップまたはリセットの後、ホストがCMD0が発行されるまでの74クロックサイクル後に引数0xFFFFFFFAでCMD0を送信する場合、カードはブートモードが開始されていると認識します。引数0xFFFFFFFAでCMD0が送信された後1秒以内に、カードはSDMMC_Dnラインで最初のブートコードデータの送信を開始します。マスタがCMD0（リセット）を送信してブート動作を終了します。

図208. 代替ブートモード動作



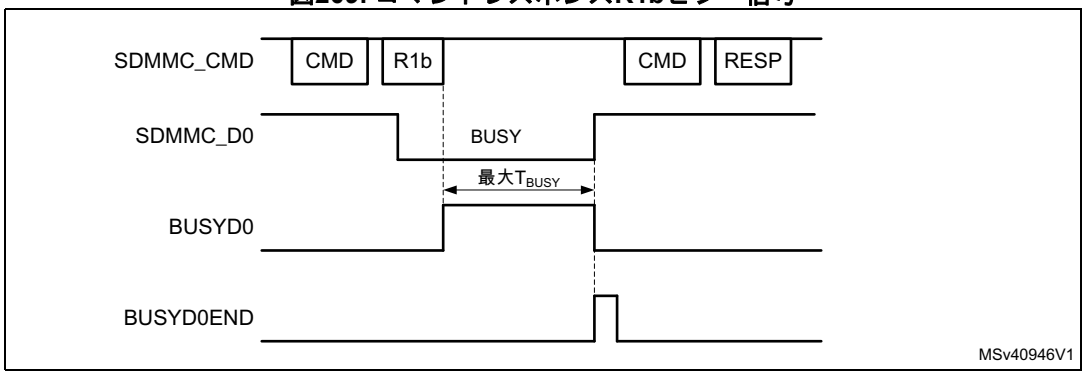
代替ブート手順を実行するには、次の手順が必要です。

1. SDMMCをパワーオフ状態に移行させ、カードをリセットします。
2. SDMMCをパワーオン状態に移行させます。これにより、どのコマンドの前にも74 SCDMMC_CKサイクルがクロック供給されることが保証されます。
3. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKTIMEOUTフラグを有効にします。
4. 受信モード（DTDIR）でDPSMとDATALENGTHで受信するデータ数をセットしてデータ受信を有効にします。DTIMEOUTとDATAENDフラグを有効にします。
5. BOOTMODE で代替ブート動作モードを選択し、コマンドレジスタで 0xFFFFFFFA の引数でCMD0 をロードします。ブートコマンドの終了確認用のCMDSENT フラグを有効にして、BOOTENでブートを有効にします。ブート手順は、CPSMENでCPSMを有効にすることで開始されます。これにより次のことが発生します。
   – ロードされたコマンドと引数が送信されます。（BOOTMODE = 代替ブート）。
   – ACKタイムアウトが開始します。
   – DPSMが有効になります。
6. コマンドが送信されると、CMDSENT フラグが生成され、その時にBOOTEN ビットをクリアする必要があります。
7. ブート確認応答の受信は、ACKFAILフラグが有効時に検出できます。
   – ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUTフラグが発生します。
8. すべてのブートデータが受信されると、DATAENDフラグが発生します。
   – データCRCが失敗した場合、DCRCFAILフラグも生成されます。
   – データタイムアウトが発生した場合、DTIMEOUTフラグも生成されます。
9. 最後のデータが受信されると、FIFOがエンプティになるまでFIFOからデータを読み出し、その後、データの終了DATAENDフラグが生成されます。
   – SDMMCがすべてのデータを完全に受信して、DPSMが無効になります。
10. BOOTMODE = 代替モードでCMD0（リセット）を送信してブート手順を終了する前にBOOTENビットをクリアする必要があります。これにより、CMDSENTフラグがコマンドの後56サイクルで発生します。
    – すべてのデータが受信される前にブート手順がファームウェアによってアボートされた場合、CPSM Abort信号によってデータ受信が停止され、DPSMが無効にされ、これにより、有効になっていればDABORTフラグがトリガされます。
11. CMDSENTフラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。RESETコマンドが正常に送信されると、ブート動作を終了するためにBOOTMODE制御ビットをクリアしなければなりません。

31.6.6 レスポンスR1bの処理

R1bレスポンスのあるコマンドを送信する場合、ビジー信号はBUSYD0レジスタビットに反映され、BUSYD0ENDフラグでビジーの開放を反映します。SDMMC_D0ラインは、R1bレスポンスの終了時にサンプリングされ、BUSYD0レジスタビットで信号送信されます。BUSYD0レジスタビットは、BUSYD0ENDフラグが生成されると同時に、SDMMC_D0ラインがビジーを開放したときにビジーでない状態にリセットされます。

図209. コマンドレスポンスR1bビジー信号



期待される最大ビジー時間を、コマンド送信前にDATATIMEレジスタでセットする必要があります。R1bレスポンスビジーがプログラムされた時間より長くアクティブになっていると、DTIMEOUTが有効になっている場合、そのフラグがセットされます。

R1bレスポンスでコマンドを送信したときにSDMMC_D0ビジー信号を検出するには、次の手順が適用されます。

- CMDRENDフラグを有効にします。
- CPSMでコマンドを送信します。
- CMDRENDフラグで、BUSYD0レジスタビットをチェックします。
  - BUSYD0がビジーではない場合、ビジー開放の信号をファームウェアに送信します。
  - BUSYD0がビジーである場合、BUSYD0ENDフラグを待ちます。
- BUSYD0ENDフラグで、ビジー開放の信号をファームウェアに送信します。
- DTIMEOUTフラグで、ビジーがプログラムされた時間以上にアクティブになります。

31.6.7 リセットとカードサイクルパワー

リセット

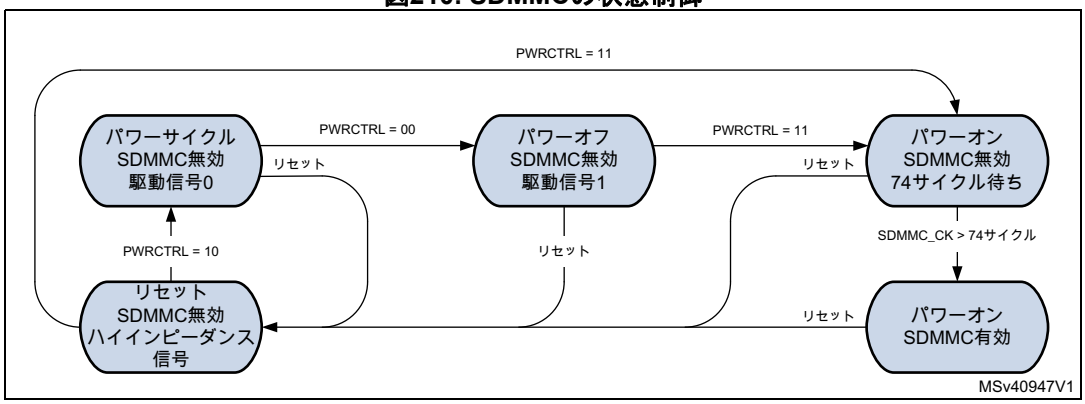
リセット後、SDMMCはリセット状態になります。この状態では、SDMMCが無効になり、コマンドやデータが転送できません。SDMMC_D[7:0] およびSDMMC_CMDはハイインピーダンスになり、SDMMC_CKはローに駆動されます。

パワーオン状態に移行する前に、SDMMCを設定する必要があります。

パワーオン状態では、SDMMC_CKクロックが実行しています。最初に74 SDMMC_CKサイクルがクロック供給され、その後にSDMMCが有効になり、コマンドとデータを転送できるようになります。

SDMMC状態は、図210.に従ったPWRCTLレジスタビットでファームウェアによって制御されます。

図210. SDMMCの状態制御

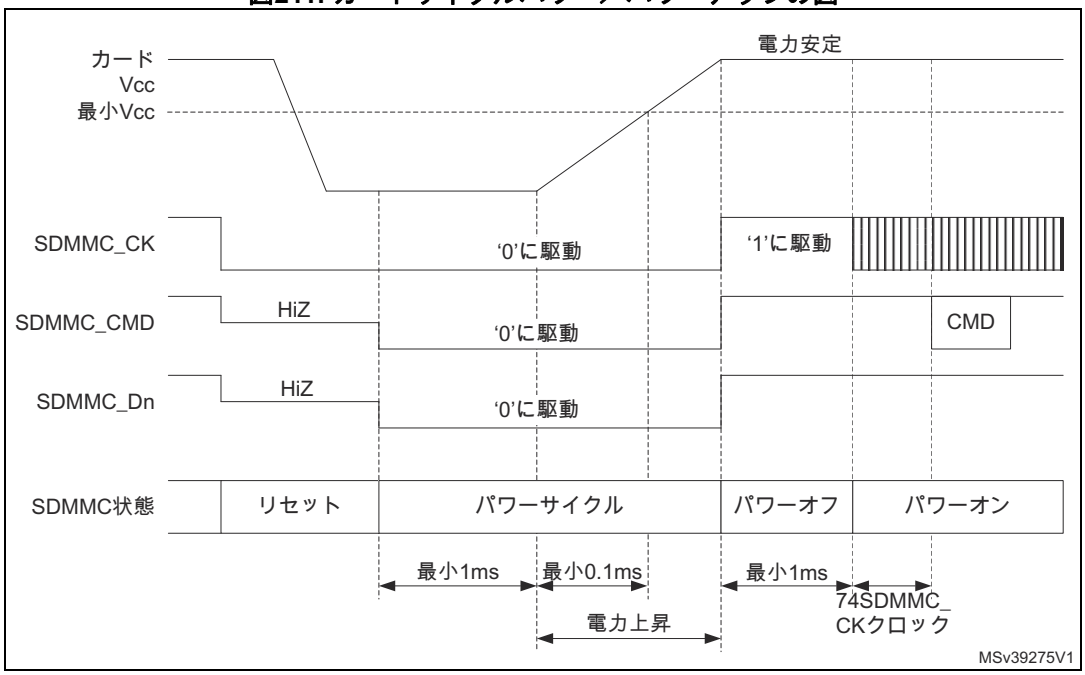


カードサイクルパワー

カードサイクルパワーを実行するには、次の手順が適用されます。

1. RCC.SDMMCxRSTレジスタビットでSDMMCをリセットします。これにより、SDMMCがリセット状態にリセットされ、CPSMとDPSMはアイドル状態へ移行します。
2. カードへのVcc電源を無効にします。
3. パワーサイクル状態でSDMMCをセットします。これにより、SDMMC_D[7:0]、SDMMC_CMDおよびSDMMC_CKがローに駆動され、カードが信号ラインから電源供給されることを防ぎます。
4. 最低1ms後に、カードへのVcc電源を有効にします。
5. パワーアップ後、最低 1ms の間 SDMMC をパワーオフ状態にセットします。SDMMC_D[7:0]、SDMMC_CMDおよびSDMMC_CKがセットされ「1」を駆動します。
6. 1msの遅延後、SDMMCをパワーオン状態にセットし、SDMMC_CKクロックを有効にします。
7. 74 SDMMC_CKサイクル後、最初のコマンドをカードに送信できるようになります。

図211. カードサイクルパワー／パワーアップの図



31.7 ハードウェアフロー制御

データ転送中のハードウェアフロー制御機能は、FIFOアンダーラン（TXモード）およびオーバーラン（RXモード）エラーを回避するために使用します。

この機能は、データ転送中にSDMMC_CKを停止し、SDMMCステートマシンをフリーズします。FIFOを使用してデータを送受信できない場合、データ転送がストールされます。送信FIFOがハーフフルになるか、DATALENGHTに応じてすべてのデータが格納されるか、受信FIFOがハーフエンプティになるまで、データ転送がストールされたままとなります。SDMMC_CKによってクロック供給されるステートマシンだけが停止し、AHBインタフェースは動作を続けます。したがって、フロー制御がアクティブであっても、FIFOがいっぱいになったり、空になることがあります。

IDMAのリンクリストの転送エラーが発生すると、ハードウェアフロー制御が無効になります。その結果、IDMAのリンクリストの転送エラーがいつ発生するかによって、アンダーランエラーまたはオーバーランエラーも発生することがあります（セクション ：IDMA転送エラー管理 を参照）。

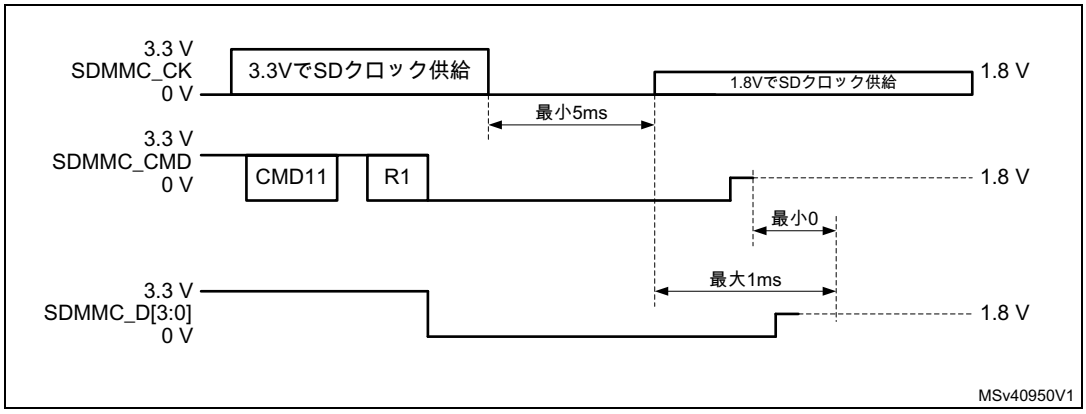
データ転送中にハードウェアフロー制御を有効にするには、HWFC_ENレジスタビットを1にセットする必要があります。リセット後、ハードウェアフロー制御は無効になります。

ハードウェアフロー制御は、SDMMC_DnデータがSDMMC_CKとサイクル調整されている場合のみ使用する必要があります。t_{OP}およびDt_{OP}の遅延が1サイクルを超えるSDR104モードの場合などDLYB遅延ブロックからのsdmmc_fb_ckが使用されている場合は、常にハードウェアフロー制御を使用できません。

31.8 ウルトラハイスピードフェーズI（UHS-I）の電圧スイッチ

UHS-Iモード（SDR12、SDR25、SDR50、SDR104、およびDDR50）は、1.8V信号のサポートが必須です。パワーアップ後、カードは3.3Vモードで開始します。CMD11が電圧スイッチシーケンスを呼び出し、1.8Vモードにします。電圧シーケンスが正常に完了すると、カードはデフォルトのSDR12でUHS-Iモードに移行し、カードの入力および出力タイミングが変わります。

図212. CMD11信号電圧スイッチシーケンス開始



信号電圧スイッチシーケンスを実行するには、次の手順が必要です。

1. 電圧スイッチ手順を開始する前に、SDMMC_CK周波数は、100～400kHzの範囲にセットする必要があります。
2. ホストは、CMD11を送信する前にVSWITCHENビットをセットして電圧スイッチを開始します。
3. カードがR1レスポンスを返します。
   - レスポンス CRC がパスして、電圧スイッチ手順を続行する場合、ホストは電圧スイッチシーケンスが完了するまでCMDおよびSDMMC_D[3:0] 信号を駆動しません。レスポンス後の数サイクルで、SDMMC_CKは停止し、CKSTOPフラグがセットされます。
   - レスポンスCRCが失敗した場合（CCRCFAILフラグ）、またはタイムアウトまでにレスポンスを受信しなかった場合（CTIMEOUTフラグ）、電圧スイッチ手順は停止します。
4. カードが、R1レスポンス後の次のクロックでCMDおよびSDMMC_D[3:0] をローに駆動します。
5. ホストは、R1レスポンスを受信した後、BUSYD0レジスタビットでSDMMC_D0ラインを監視します。SDMMC_D0ラインは、レスポンス後2 SDMMC_CKクロックサイクルでサンプリングされます。ファームウェアは、CKSTOPフラグの後にBUSYD0レジスタビットを読み出す場合があります。
   - BUSYD0がローで検出されると、ホストファームウェアは電圧レギュレータを1.8Vに切り替え、SDMMCにレジスタビットVSWITCHをセットして電圧スイッチシーケンスのタイミングが重要なセクションを開始するよう命令します。ハードウェアは、最低5msの間SDMMC_CKをローに保って停止させ続けます。
   - BUSYD0がハイで検出されると、ホストは電圧スイッチシーケンスをアボートして、カードをサイクルパワーします。
6. SDMMC_CKがローで検出された後、カードは信号電圧を1.8Vに切り替え始めます。
7. 最低5ms後に、ホストSDMMCハードウェアはSDMMC_CKを再開します。
8. SDMMC_CKの遷移を検出してから1ms以内に、カードは最低1 SDMMC_CKサイクルの間CMDとDAT[3:0] をハイに駆動してから、CMDとDAT[3:0] の駆動を停止させます。
9. ホストSDMMCハードウェアでは、SDMMC_CKが再開してから1msでSDMMC_D0がBUSYD0にサンプリングされ、VSWENDフラグがセットされます。

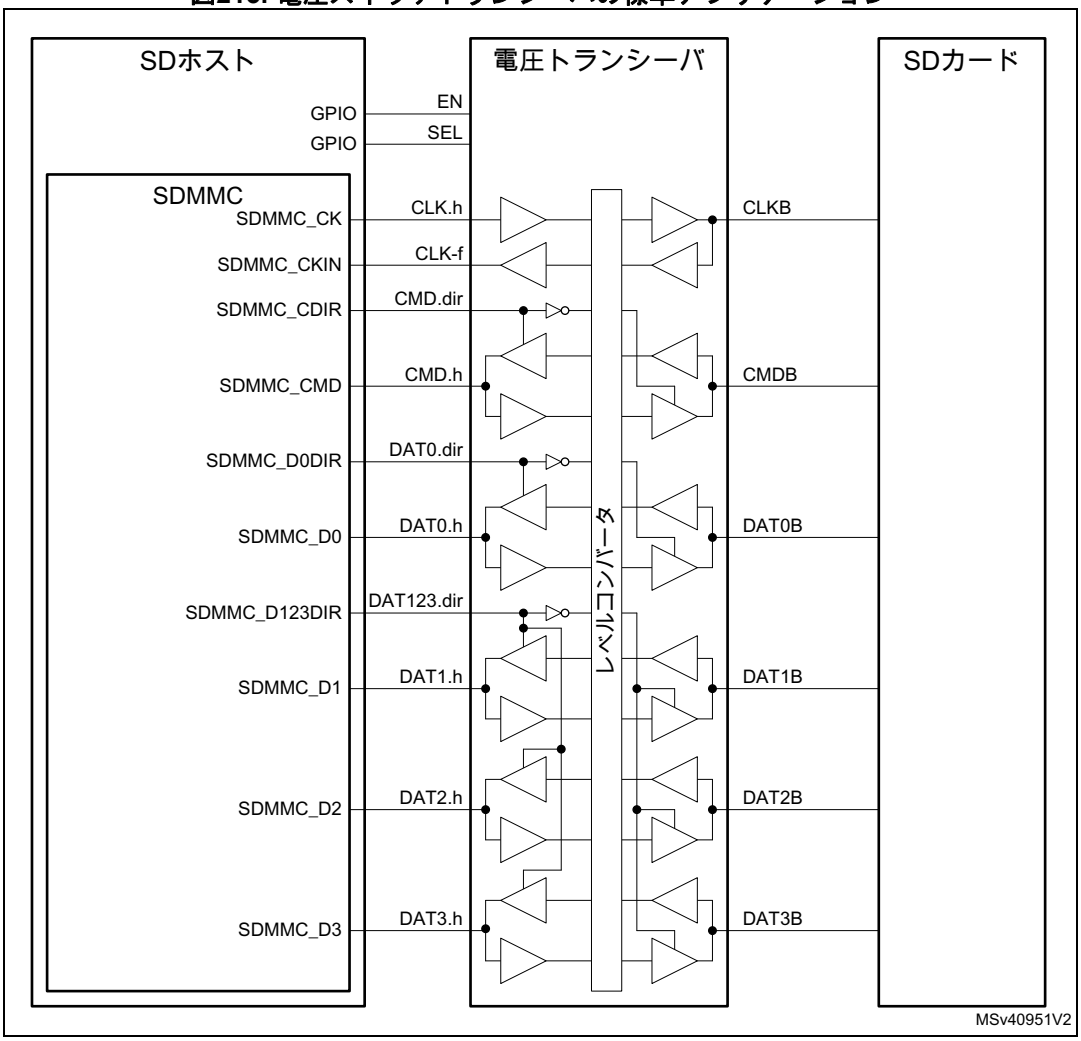
10. VSWENDフラグで、ホストはBUSYD0レジスタビットを使ってSDMMC_D0ラインをチェックし、電圧スイッチシーケンスの完了を確認します。
    - BUSYD0がハイで検出される場合、電圧スイッチが正常に完了しています。
    - BUSYD0がローで検出される場合、電圧スイッチは失敗であり、ホストはカード電源をサイクルパワーします。

SDMMC_CKを停止するための最低5msの時間は、クロック分周回路CLKDIVでセットされている、内部のゲートされていないSDMMC_CKクロックが使用され、最大周波数は25MHz（SDモード）です。5msを超える時間は、2^{12}サイクル（10.24ms @ 400kHz）でカウントされます。より低いSDMMC_CK周波数がクロック分周回路CLKDIVで選択されている場合、SDMMC_CKクロックが停止する時間は長くなります。

カードがSDMMC_DnとSDMMC_CMDをハイに駆動する最大1msの時間は、クロック分周回路CLKDIVでセットされている、内部のゲートされていないSDMMC_CKが使用され、最大周波数は25MHz（SDモード）です。SDMMCは、1msを超える時間以降、2^{9}サイクル（1.28ms @ 25MHz）でカウントしてラインをチェックします。より低いSDMMC_CK周波数がクロック分周回路CLKDIVで選択されている場合、ラインをチェックする時間は長くなります。

信号電圧レベルは、STマイクロエレクトロニクスのST6G3244MEのような外部電圧変換トランシーバを通じてサポートされます。

図213. 電圧スイッチトランシーバの標準アプリケーション



外部ドライバ（電圧スイッチトランシーバ）とインタフェース接続するために、標準信号とは別に、SDMMCは次の信号を使用します。

SDMMC_CKIN フィードバック入力クロック

SDMMC_CDIR CMD信号のI/O方向制御

SDMMC_D0DIR SDMMC_D0信号のI/O方向制御

SDMMC_D123DIR SDMMC_D1、SDMMC_D2、およびSDMMC_D3信号のI/O方向制御

電圧トランシーバ信号**EN**および**SEL** は、汎用IOを通じて処理されます。

SDMMC_CDIR、SDMMC_D0DIR、およびSDMMC_D123DIR信号の極性は、SDMMC_POWER.DIRPOL制御ビットで選択できます。

31.9 SDMMC割込み

表290.SDMMC割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了
SDMMC	コマンドレスポンスCRC失敗	CCRCFAIL	CCRCFAILIE	CCRCFAILC	あり
SDMMC	データブロックCRC失敗	DCRCFAIL	DCRCFAILIE	DCRCFAILC	あり
SDMMC	コマンドレスポンスタイムアウトです。	CTIMEOUT	CTIMEOUTIE	CTIMEOUTC	あり
SDMMC	データタイムアウトです。	DTIMEOUT	DTIMEOUTIE	DTIMEOUTC	あり
SDMMC	送信FIFOアンダーラン	TXUNDERR	TXUNDERRIE	TXUNDERRC	あり
SDMMC	受信FIFOオーバーラン	RXOVERR	RXOVERRIE	RXOVERRC	あり
SDMMC	コマンドレスポンス受信	CMDREND	CMDRENDIE	CMDRENDC	あり
SDMMC	コマンド送信	CMDSENT	CMDSENTIE	CMDSENTC	あり
SDMMC	データ転送終了	DATAEND	DATAENDIE	DATAENDC	あり
SDMMC	データ転送保留	DHOLD	DHOLDIE	DHOLDC	あり
SDMMC	データブロック送受信	DBCKEND	DBCKENDIE	DBCKENDC	あり
SDMMC	データ転送アボート	DABORT	DABORTIE	DABORTC	あり
SDMMC	送信FIFOハーフエンプティ	TXFIFOHE	TXFIFOHEIE	該当なし	あり
SDMMC	受信FIFOハーフフル	RXFIFOHF	RXFIFOHFIE	該当なし	あり
SDMMC	送信FIFOがいっぱいです。	TXFIFOF	該当なし	該当なし	あり
SDMMC	受信FIFOがいっぱいです。	RXFIFOF	RXFIFOFIE	該当なし	あり
SDMMC	送信FIFOエンプティ	TXFIFOE	TXFIFOEIE	該当なし	あり
SDMMC	受信FIFOエンプティ	RXFIFOE	該当なし	該当なし	あり
SDMMC	コマンドレスポンスのビジー終了	BUSYD0END	BUSYD0ENDIE	BUSYD0ENDC	あり
SDMMC	SDIO割込み	SDIOIT	SDIOITIE	SDIOITC	あり
SDMMC	ブート確認応答失敗	ACKFAIL	ACKFAILIE	ACKFAILC	あり
SDMMC	ブート確認応答タイムアウト	ACKTIMEOUT	ACKTIMEOUTIE	ACKTIMEOUTC	あり
SDMMC	電圧スイッチタイミング	VSWEND	VSWENDIE	VSWENDC	あり
SDMMC	SDMM_CKが電圧スイッチで停止	CKSTOP	CKSTOPIE	CKSTOPC	あり
SDMMC	IDMA転送エラー	IDMATE	IDMATEIE	IDMATEC	あり
SDMMC	IDMAバッファ転送完了	IDMABTC	IDMABTCIE	IDMABTCC	あり

31.10 SDMMCレジスタ

デバイスは、AHBスレーブインタフェース経由でアクセス可能な32bitの制御レジスタによってシステムと通信します。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスする必要があります。バイト（8-bit）およびハーフワード（16-bit）のアクセスは、AHBバスエラーをトリガします。

31.10.1 SDMMC電源制御レジスタ（SDMMC_POWER）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR POL	VSWI TCHEN	VSWI TCH	PWRCTRL[1:0]	
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **DIRPOL**: データおよびコマンドの方向信号の極性選択

このビットは、SDMMCがパワーオフ状態（PWRCTRL = 00）のときのみ書き込むことができます。

0：方向信号がローのときに電圧トランシーバのIOが出力として駆動されます。

1：方向信号がハイのときに電圧トランシーバのIOが出力として駆動されます。

ビット3 **VSWITCHEN**: 電圧スイッチ手順有効

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットは、電圧スイッチコマンドレスポンスの後SDMMC_CKを停止するために使用されます。

0：コマンドレスポンスを正常に受信した後、SDMMC_CKクロックに変化はありません。

1：コマンドレスポンスを正常に受信した後、SDMMC_CKクロックが停止します。

ビット2 VSWITCH：電圧スイッチシーケンス開始

このビットは、電圧スイッチシーケンスのタイミングの重要なセクションの開始に使用されます。

0：電圧スイッチシーケンスは開始されずアクティブではありません。

1：電圧スイッチシーケンスが開始されているかアクティブです。

ビット1:0 **PWRCTRL[1:0]**：SDMMCの状態制御ビット

これらのビットは、SDMMCがパワーオン状態ではないとき（PWRCTRL ≠ 11）のみ書き込むことができます。

これらのビットは、SDMMC信号の機能状態を定義するために使用されます。

00：リセット後、リセット：SDMMCは無効化され、カードへのクロックは停止され、SDMMC_D[7:0]およびSDMMC_CMDはハイインピーダンスになり、SDMMC_CKはローに駆動されます。
00が書き込まれた場合、パワーオフ：SDMMCは無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMDおよびSDMMC_CKはハイに駆動されます。

01：予約済み。（01が書き込まれた場合、PWRCTRL値は変化しません）

10：パワーサイクル、SDMMCは無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMDおよびSDMMC_CKはローに駆動されます。

11：パワーオン：カードはクロック供給されます。最初の74 SDMMC_CKサイクルはSDMMCが無効化されたままになります。74サイクル後、SDMMCは有効になり、SDMMC_D[7:0]、SDMMC_CMDおよびSDMMC_CKはSDMMCの動作に従って制御されます。

追加の書込みは無視され、PWRCTRL値は11のままとなります。

31.10.2 SDMMCクロック制御レジスタ（SDMMC_CLKCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

SDMMC_CLKCRレジスタは、SDMMC_CKの出力クロック、sdmmc_rx_ck受信クロック、バス幅を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUS SPEED	DDR	HWFC_EN	NEG EDGE
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WID BUS[1:0]		Res.	PWR SAV	Res.	Res.	CLKDIV[9:0].									
rw	rw		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 SELCLKRX[1:0]：受信クロックの選択

これらのビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

00：sdmmc_io_in_ckが受信クロックとして選択されます。

01：SDMMC_CKINフィードバッククロックが受信クロックとして選択されます。

10：sdmmc_fb_ck調整フィードバッククロックが受信クロックとして選択されます。

11：予約済み（sdmmc_io_in_ckを選択）

ビット19 **BUSSPEED**：SDMMC動作モードの選択のためのバス速度

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

0：DS、ハイスピード、SDR12、SDR25、レガシー互換、高速SDR、高速DDRのバス速度モードが選択されます。

1：SDR50、DDR50、SDR104、HS200 のバス速度モードが選択されます。

ビット18 DDR：データレートシグナリングの選択

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

DDR レートは、4bitまたは8bit幅のバスモードでのみ選択する必要があります。（WIDBUS > 00）。WIDBUS = 00（1bit幅バス）の場合、DDR = 1は効果がありません。

DDRレートは、クロック分周が1より大きい場合のみ選択する必要があります。（CLKDIV > 0）。

0：SDRシングルデータレートシグナリング

1：DDRダブルデータレートシグナリング

ビット17 HWFC_EN：ハードウェアフロー制御の有効化

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

0：ハードウェアフロー制御は無効です。

1：ハードウェアフロー制御は有効です。

ハードウェアフロー制御が有効なときのTXFIFOEおよびRXFIFOFフラグ変更の意味については、セクション31.10.11のSDMMCステータスレジスタの定義を参照してください。

ビット16 NEGEDGE：データとコマンドのSDMMC_CK位相反転選択ビット

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

クロック分周 = 1（CLKDIV = 0）の場合、このビットは意味を持ちません。コマンドとデータはSDMMC_CK立下がりエッジで変更されます。

0：クロック分周 >1（CLKDIV > 0）およびDDR = 0の場合：

– コマンドとデータはSDMMC_CK立上がりエッジに続くsdmmc_ker_ck立下がりエッジで変更されます。
– SDMMC_CKエッジはsdmmc_ker_ck立上がりエッジで発生します。

クロック分周 >1（CLKDIV > 0）およびDDR = 1の場合：

– コマンドはSDMMC_CK立上がりエッジに続くsdmmc_ker_ck立下がりエッジで変更されます。
– データはSDMMC_CKエッジに続くsdmmc_ker_ck立下がりエッジで変更されます。
– SDMMC_CKエッジはsdmmc_ker_ck立上がりエッジで発生します。

1：クロック分周 >1（CLKDIV > 0）およびDDR = 0の場合：

– コマンドとデータはSDMMC_CK立下がりエッジを生成する同じsdmmc_ker_ck立上がりエッジで変更されます。

クロック分周 >1（CLKDIV > 0）およびDDR = 1の場合：

– コマンドはSDMMC_CK立下がりエッジを生成する同じsdmmc_ker_ck立上がりエッジで変更されます。
– データはSDMMC_CKエッジに続くSDMMC_CK立下がりエッジで変更されます。
– SDMMC_CKエッジはsdmmc_ker_ck立上がりエッジで発生します。

ビット15:14 **WIDBUS[1:0]**：ワイドバスモード有効化ビット

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

00：デフォルトの1bit幅のバスモード：SDMMC_D0使用（DDRをサポートしません）

01：4bit幅のバスモード：SDMMC_D[3:0] 使用

10：8bit幅のバスモード：SDMMC_D[7:0] 使用

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 PWRSAV：節電構成ビット

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

節電のため、バスがアイドル状態のときには、PWRSAVをセットしてSDMMC_CKクロック出力を無効にできます。

0：SDMMC_CKクロックは常に有効です。

1：SDMMC_CKはバスがアクティブなときに限り有効です。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **CLKDIV[9:0]**：クロック分周ファクタ

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

このフィールドは、入力クロック（sdmmc_ker_ck）と出力クロック（SDMMC_CK）の間の分周ファクタを定義します。SDMMC_CK周波数 = sdmmc_ker_ck / [2 * CLKDIV]。

0x000：SDMMC_CK周波数 = sdmmc_ker_ck / 1（DDRをサポートしません）

0x001：SDMMC_CK周波数 = sdmmc_ker_ck / 2

0x002：SDMMC_CK周波数 = sdmmc_ker_ck / 4

0x0XX：同様に続く

0x080：SDMMC_CK周波数 = sdmmc_ker_ck / 256

0xXXX：同様に続く

0x3FF：SDMMC_CK周波数 = sdmmc_ker_ck / 2046

注： 1 SD/SDIOカードまたはeMMCが識別モードのときには、SDMMC_CK周波数は400kHz未満である必要があります。

2 クロック周波数は、相対カードアドレスがすべてのカードに割り当てられているときには、最大カードバス周波数に変更できます。

3 このレジスタへの各書込みアクセス間には、7 sdmmc_hclkクロック周期以上が必要です。SDMMC_CKは、SD I/Oカードの読出しウェイトインターバルの間も、停止することができます。この場合、SDMMC_CLKCRレジスタはSDMMC_CKを制御しません。

31.10.3 SDMMC引数レジスタ（SDMMC_ARGR）

アドレスオフセット：0x008

リセット値：0x00000

SDMMC_ARGRレジスタは、コマンドメッセージの一部としてカードに送信される32bitのコマンド引数を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMDARG[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMDARG[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CMDARG[31:0]**：コマンド引数

これらのビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。コマンドメッセージの一部としてカードに送られるコマンド引数です。コマンドが引数を含む場合、コマンドをコマンドレジスタに書き込む前に、このレジスタにロードする必要があります。

31.10.4 SDMMCコマンドレジスタ（SDMMC_CMDR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

SDMMC_CMDRレジスタは、コマンドインデックスおよびコマンドタイプビットを含みます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。コマンドタイプビットは、コマンドパスステートマシン（CPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMD SUS PEND
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT ピン EN	BOOT ピン MODE	DT HOLD	CPSM EN	WAITP END	WAIT INT	WAITRESP[1:0]		CMD STOP	CMD TRANS	CMDINDEX[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 CMDSUSPEND：CPSMがコマンドをサスペンドまたはレジュームコマンドとして処理し、割込み期間の開始／終了の信号を送信

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

CMDSUSPEND = 1およびCMDTRANS = 0のサスペンドコマンドで、レスポンスビットBS=0の場合、割込み期間を開始します。

CMDSUSPEND = 1およびCMDTRANS = 1のレジュームコマンドとデータで、レスポンスビットDF=1の場合、割込み期間を終了します。

ビット15 BOOTEN：ブートモード手順の有効化

0：ブートモード手順は無効です。

1：ブートモード手順は有効です。

ビット14 BOOTMODE：使用するブートモード手順の選択

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

0：通常ブートモード手順を選択します。

1：代替ブートモード手順を選択します。

ビット13 DTHOLD：DPSMでの新しいデータブロックの送受信の保留

このビットがセットされると、DPSMはWait_S状態から送信状態へ、またはWait_R状態から受信状態へ移行しません。

ビット12 CPSMEN：コマンドパスステートマシン（CPSM）有効化ビット

このビットは、CPSMがアイドル状態に移行すると、ファームウェアによって1が書き込まれ、ハードウェアによってクリアされます。

このビットがセットされると、CPSMが有効になります。

DTEN = 1の場合、コマンドは転送されず、ブート手順も開始されません。CPSMENは0にクリアされます。

SDMMC_CKを停止した読出しウェイト状態の間、コマンドは送信されず、CPSMENは0のままとなります。

ビット11 WAITPEND：CPSMのDPSMからのデータ転送終了待ち（CmdPend内部信号）

このビットがセットされると、CPSMは、データ転送の終了を待ってから、コマンドの送信を開始します。

DTMODE = *e*•MMCストリームデータ転送、WIDBUS = 1bit幅バスモード、DPSMACT = 1かつDTDIR = ホストからカードの場合のみ、WAITPENDは考慮されます。

ビット10 WAITINT：CPSMの割込みリクエスト待ち

このビットがセットされると、CPSMは、コマンドタイムアウトを無効にして、割込みリクエスト（レスポンス）を待ちます。

CPSMウェイトステートでこのビットがクリアされると、割込みモードのアボートが発生します。

ビット9:8 **WAITRESP[1:0]**：レスポンスビット待ち

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

これらは、CPSMがレスポンスを待つかどうか、および、待つ場合にはレスポンスの種類を設定するために使用されます。

00：レスポンスなし。CMDSENTフラグを待ちます。

01：ショートレスポンス。CMDRENDまたはCCRCFAILフラグを待ちます。

10：ショートレスポンス。CMDRENDフラグ（CRCなし）を待ちます。

11：ロングレスポンス。CMDRENDまたはCCRCFAILフラグを待ちます。

ビット7 CMDSTOP：CPSMがコマンドを送信停止コマンドとして処理し、DPSMにAbort信号を送信

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSMはコマンド送信時にAbort信号をDPSMに発行します。

ビット6 CMDTRANS：CPSMがコマンドをデータ転送コマンドとして処理し、割込み期間を停止し、DPSMにDataEnable信号を送信

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSMはコマンド送信時に割込み期間の終了とDataEnable信号をDPSMに発行します。

ビット5:0 **CMDINDEX[5:0]**：コマンドインデックス

このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。

注： 1 このレジスタへの各書込みアクセス間には、7 sdmmc_hclkクロック周期以上が必要です。

2 MultiMediaCardはショートレスポンス（48bit）またはロングレスポンス（136bit）の2種類のレスポンスを送信できます。SDカードとSD I/Oカードはショートレスポンスのみ送信でき、引数はレスポンスのタイプによって異なることがあります。ソフトウェアは、送信コマンドに応じてレスポンスのタイプを区別します。

31.10.5 SDMMCコマンドレスポンスレジスタ（SDMMC_RESPCMDR）

アドレスオフセット：0x010

リセット値：0x0000 0000

SDMMC_RESPCMDRレジスタは、最後に受信したコマンドレスポンスのコマンドインデックスフィールドを含みます。コマンドレスポンス送信にコマンドインデックスフィールドが含まれていない場合（ロングまたはOCRレスポンス）、RESPCMDフィールドは未知ですが、111111b（レスポンスの予約フィールドの値）が含まれている必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
										r	r	r	r	r	r

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5:0 **RESPCMD[5:0]**：レスポンスコマンドインデックス

読出し専用ビットフィールドです。最後に受信したコマンドレスポンスのコマンドインデックスを含みます。

31.10.6 SDMMCレスポンスxレジスタ（SDMMC_RESPxR）

アドレスオフセット：0x010 + 0x004 * x（x = 1から4）

リセット値：0x0000 0000

SDMMC_RESP1/2/3/4Rレジスタは、受信したレスポンスの一部であるカードのステータスを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CARDSTATUS[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CARDSTATUS[15:0] :															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **CARDSTATUS[31:0]**：以下の表に従ったカードステータス

表291を参照してください。

カードステータスサイズは、レスポンスのタイプによって32または128bitです。

表291.レスポンスタイプとSDMMC_RESPxRレジスタ

レジスタ(1)	ショートレスポンス	ロングレスポンス
SDMMC_RESP1R	カードステータス [31:0]	カードステータス [127:96]
SDMMC_RESP2R	すべて0	カードステータス [95:64]
SDMMC_RESP3R	すべて0	カードステータス [63:32]
SDMMC_RESP4R	すべて0	カードステータス [31:0](2)

1. カードステータスの最上位ビットから先に受信します。
2. SDMMC_RESP4RレジスタLSBは常に0です。

31.10.7 SDMMCデータタイマレジスタ（SDMMC_DTIMER）

アドレスオフセット：0x024

リセット値：0x0000 0000

SDMMC_DTIMERレジスタは、データタイムアウト期間（カードバスクロック周期単位）を含みます。

カウンタはSDMMC_DTIMERレジスタから値をロードして、データパスステートマシン（DPSM）がWait_Rまたはビジー状態に入ったときに、デクリメントを開始します。DPSMがこのどちらかの状態のときにタイマが0に達した場合、タイムアウトステータスフラグがセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATATIME[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATATIME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATATIME[31:0]**：データおよびR1bビジータイムアウト期間

このビットに書き込めるのは、CPSMおよびDPSMがアクティブではないときだけです（CPSMACT = 0およびDPSMACT = 0）。

カードバスクロック周期で表されたデータおよびR1bビジータイムアウト期間です。

注：　データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。

31.10.8 SDMMCデータ長レジスタ（SDMMC_DLENR）

アドレスオフセット：0x028

リセット値：0x0000 0000

SDMMC_DLENRレジスタは、転送されるデータバイト数を含みます。値は、データ転送が開始されたときに、データカウンタにロードされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATALENGTH[24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATALENGTH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 DATALENGTH[24:0]：データ長の値

このレジスタは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

転送されるデータバイト数です。

DDR = 1の場合、DATALENGTHは2の倍数に切り詰められます。（最後の奇数バイトは転送されません）

DATALENGTH = 0の場合、データは転送されず、CPSMENおよびCMDTRANS = 1でリクエストされたときは、コマンドも転送されません。DTENおよびCPSMENは0にクリアされます。

注： **ブロックデータ転送では、データ長レジスタの値はブロックサイズの倍数である必要があります（SDMMC_DCTRLを参照してください）。データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。**

SDMMCマルチバイト転送の場合、データ長レジスタ内の値は1から512までででなければなりません。

31.10.9 SDMMCデータ制御レジスタ（SDMMC_DCTRL）

アドレスオフセット：0x02C

リセット値：0x0000 0000

SDMMC_DCTRLレジスタは、データパスステートマシン（DPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FIFO RST	BOOT ピン ACK（確認応答）EN	SDIO EN	RW MOD	RW STOP	RW START	DBLOCKSIZE[3:0]				DTMODE[1:0]		DTDIR	DTEN
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **FIFORST**：FIFOリセットで残りのデータを一掃

このビットは、IDMAEN= 0およびDPSMがアクティブ（DPSMACT = 1）のときのみファームウェアで書き込むことができます。このビットは、転送エラーまたは転送保留が発生したときのみ有効になります。

0：FIFOに影響がありません。

1：残りのデータを一掃してFIFOポインタをリセットします。このビットは、DPSMがインアクティブ（DPSMACT = 0）になったとき、ハードウェアによって自動的に0にクリアされます。

ビット12 BOOTACKEN：ブート確認応答の受信の有効化

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ブート確認応答は無効になり、受信の予定はありません。

1：ブート確認応答は有効になり、受信の予定があります。

ビット11 SDIOEN：SD I/O割込みの有効化機能

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされると、DPSMはSD I/Oカード固有の割込み操作を可能にします。

ビット10 **RWMOD**：読出しウェイトモード

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：SDMMC_D2を使用する読出しウェイト制御です。

1：SDMMC_CKを停止する読出しウェイト制御です。

ビット9 **RWSTOP**：読出しウェイト停止

このビットは、ファームウェアによって書き込まれ、DPSMがR_W状態からWait_Rまたはアイドル状態に移行すると、ハードウェアによって自動クリアされます。

0：読出しウェイト停止はありません。

1：DPSMがR_W状態の場合、読出しウェイト停止が有効になります。

ビット8 **RWSTART**：読出しウェイト開始

このビットがセットされると、読出しウェイト動作が開始されます。

ビット7:4 DBLOCKSIZE[3:0]：データブロックサイズ

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ブロックデータ転送モードが選択されているときのデータブロック長を定義します。

0000：ブロック長 = 2^0 = 1バイト

0001：ブロック長 = 2^1 = 2バイト

0010：ブロック長 = 2^2 = 4バイト

0011：ブロック長 = 2^3 = 8バイト

0100：ブロック長 = 2^4 = 16バイト

0101：ブロック長 = 2^5 = 32バイト

0110：ブロック長 = 2^6 = 64バイト

0111：ブロック長 = 2^7 = 128バイト

1000：ブロック長 = 2^8 = 256バイト

1001：ブロック長 = 2^9 = 512バイト

1010：ブロック長 = 2^{10} = 1024バイト

1011：ブロック長 = 2^{11} = 2048バイト

1100：ブロック長 = 2^{12} = 4096バイト

1101：ブロック長 = 2^{13} = 8192バイト

1110：ブロック長 = 2^{14} = 16384バイト

1111：予約済み

DATALENGTHがDBLOCKSIZEの整数倍ではない場合、転送されたデータはDBLOCKSIZEの倍数に切り捨てられます（残りのデータは転送されません）。

DDR = 1の場合、DBLOCKSIZE = 0000は使用してはなりません。（データは転送されません。）

ビット3:2 **DTMODE[1:0]**：データ転送モードの選択

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

00：ブロック数で終了するブロックデータ転送。

01：SDIOマルチバイトデータ転送。

10：*e*•MMCストリームデータ転送（WIDBUSでは1bit幅のバスモードを選択する必要があります）

11：STOP_TRANSMISSIONコマンドで終了するブロックデータ転送（DTENで開始するデータ転送では使用しません）。

ビット1 DTDIR：データ転送方向の選択

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ホストからカードへ。

1：カードからホストへ。

ビット0 DTEN：データ転送有効化ビット

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。このビットは、データ転送が完了したとき、ハードウェアによってクリアされます。

このビットは、関連するデータ転送コマンドが使用されないときのみ、データを転送するために使用する必要があります。つまり、SDカードや *e*•MMCカードでは使用しないでください。

0：CPSMデータ転送コマンドなしでデータ転送を開始しません。

1：CPSMデータ転送コマンドなしでデータ転送を開始します。

31.10.10 SDMMCデータカウンタレジスタ（SDMMC_DCNTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

SDMMC_DCNTRレジスタは、DPSMがアイドル状態からWait_RまたはWait_S状態へ移行すると、データ長レジスタから値をロードします（SDMMC_DLENRを参照してください）。データが転送されると、カウンタは0に達するまで値をデクリメントします。その後、DPSMはアイドル状態へ移行し、エラーがなく、送信データ転送保留がない場合、データステータス終了フラグ（DATAEND）がセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT[24:16]								
							r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATACOUNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 DATACOUNT[24:0]：データカウント値

読み出すと、残りの転送バイト数が返されます。書込み操作は何の影響も与えません。

注： **このレジスタは、データ転送を完了または保留したときだけ読み出す必要があります。エラーイベントの後に読み出す場合、読出しデータカウント値は実際に転送されたデータバイト数と異なる場合があります。**

31.10.11 SDMMCステータスレジスタ（SDMMC_STAR）

アドレスオフセット：0x034

リセット値：0x0000 0000

SDMMC_STARレジスタは、読出し専用レジスタです。2種類のフラグを含みます。

- スタティックフラグ（ビット [28, 21, 11:0]）：これらのビットは、SDMMC割込みクリアレジスタ（SDMMC_ICRを参照）に書き込んでクリアされるまで、アサートされたままです。
- ダイナミックフラグ（ビット [20:12]）：これらのビットは、基礎回路の状態に応じて状態が変わります（たとえば、FIFOフルおよびエンプティフラグは、データがFIFOに書き込まれるとアサートおよびネゲートされます）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMABTC	IDMATE	CKSTOP	VSWEND	ACK（確認応答）TIMEOUT	ACK（確認応答）FAIL	SDIOIT	BUSYD0END	BUSYA0	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF
			r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFIFOHF	TXFIFOHE	CPSMACT	DPSMACT	DABORT	DBCKEND	DHOLD	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **IDMABTC**：IDMAバッファ転送完了
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット27 **IDMATE**：IDMA転送エラー
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット26 CKSTOP：SDMMC_CKが電圧スイッチ手順で停止
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット25 VSWEND：電圧スイッチの重要なタイミングセクションの完了
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット24 **ACKTIMEOUT**：ブート確認応答タイムアウト
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット23 ACKFAIL：ブート確認応答の受信（ブート確認応答チェック失敗）
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット22 **SDIOIT**：SDIO割込みの受信
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット21 **BUSYD0END**：CMDレスポンス検出後のSDMMC_D0ビジーの終了
これは、CMDレスポンス後のビジーの終了のみ示します。このビットは、データ転送によってビジー信号を送信しません。割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。
0：カードSDMMC_D0信号は、ビジーからビジーでない状態への変更を伝えません。
1：カードSDMMC_D0信号は、ビジーからビジーでない状態に変更されました。

ビット20 BUSYD0 : CMDレスポンスの終了時と、CMDレスポンス後の2 SDMMC_CKサイクルでサンプリングされたSDMMC_D0ラインの反転された値（ビジー）

このビットは、SDMMCD0ラインがビジーからビジーでない状態に変わる際に、ビジーでない状態にリセットされます。このビットは、データ転送によってビジー信号を送信しません。これはハードウェアステータスフラグのみで、割込みは生成されません。

0：カードがSDMMC_D0でビジーでない状態を伝えます。

1：カードがSDMMC_D0でビジー状態であることを伝えます。

ビット19 **RXFIFOE**：受信FIFOエンプティ

これはハードウェアステータスフラグのみで、割込みは生成されません。このビットは、1つのFIFOの位置がフルになったときにクリアされます。

ビット18 **TXFIFOE**：送信FIFOエンプティ

このビットは、1つのFIFOの位置がフルになったときにクリアされます。

ビット17 **RXFIFOF**：受信FIFOフル

このビットは、1つのFIFOの位置がエンプティになったときにクリアされます。

ビット16 **TXFIFOF**：送信FIFOフル

これはハードウェアステータスフラグのみで、割込みは生成されません。このビットは、1つのFIFOの位置がエンプティになったときにクリアされます。

ビット15 RXFIFOHF：受信FIFOハーフフル

FIFOには少なくともワード数の半分入っています。このビットは、FIFOがハーフ +1空になったときにクリアされます。

ビット14 TXFIFOHE：送信FIFOハーフエンプティ

FIFOに少なくともワード数の半分を書込み可能です。このビットは、FIFOがハーフ +1フルになったときにクリアされます。

ビット13 CPSMACT：コマンドパスステートマシンがアイドル状態などではなく、アクティブ

これはハードウェアステータスフラグのみで、割込みは生成されません。

ビット12 DPSMACT：データパスステートマシンがアイドル状態などではなく、アクティブ

これはハードウェアステータスフラグのみで、割込みは生成されません。

ビット11 DABORT：CMD12によるデータ転送のアボート

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット10 DBCKEND：データブロック送受信

次のような場合にDBCKENDが生成されます。

- CRC確認に成功し、DPSMがR_W状態へ移行

または

- IDMAEN = 0かつ送信データ転送が保留され、DATACOUNT >0であり、DPSMがWait_Sへ移行

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット9 **DHOLD**：データ転送保留

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット8 **DATAEND**：データ転送の正常終了

データカウンタのDATACOUNTが0で、発生したエラーがなく、送信データ転送保留がない場合にDATAENDがセットされます。

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット7 CMDSENT：コマンド送信（レスポンス不要）。

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット6 CMDREND：コマンドレスポンス受信（CRC確認成功、またはCRCなし）

割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット5 **RXOVERR**：受信FIFOオーバーランエラー（IDMAが有効な場合はハードウェアでマスクされます）
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット4 **TXUNDERR**：送信FIFOアンダーランエラー（IDMAが有効な場合はハードウェアでマスクされます）
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット3 **DTIMEOUT**：データタイムアウト
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット2 **CTIMEOUT**：コマンドレスポンスタイムアウト
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。
コマンドタイムアウト期間は、固定値の64 SDMMC_CKクロック周期です。

ビット1 DCRCFAIL：データブロック送受信（CRC確認失敗）
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

ビット0 CCRCFAIL：コマンドレスポンス受信（CRC確認失敗）
割込みフラグは、SDMMC_ICRで対応する割込みクリアビットに書き込むことでクリアされます。

注： FIFO割込みフラグは、IDMAモード使用時SDMMC_MASKRでマスクする必要があります。

31.10.12 SDMMC割込みクリアレジスタ（SDMMC_ICR）

アドレスオフセット：0x038

リセット値：0x0000 0000

SDMMC_ICRレジスタは、書込み専用レジスタです。ビットに1を書き込むと、SDMMC_STARステータスレジスタの対応するビットがクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMA BTCC	IDMA TEC	CK STOPC	VSW ENDC	ACK（確認応答）TIME OUTC	ACK（確認応答）FAILC	SDIO ITC	BUSY A0 ENDC	Res.	Res.	Res.	Res.	Res.
			rw	rw	rw	rw	rw	rw	rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	D ABORT C	DBCK ENDC	DHOLD C	DATA ENDC	CMD SENTC	CMDR ENDC	RX OVERR C	TX UNDER RC	D TIME OUTC	C TIME OUTC	DCRC FAILC	CCRC FAILC
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **IDMABTCC**：IDMAバッファ転送完了クリアビット
IDMABTCCフラグをクリアするために、ソフトウェアによってセットされます。
0：IDMABTCCはクリアされません。
1：IDMABTCCはクリアされます。

ビット27 **IDMATEC**：IDMA転送エラークリアビット
IDMATEフラグをクリアするために、ソフトウェアによってセットされます。
0：IDMATEはクリアされません。
1：IDMATEはクリアされます。

ビット26 CKSTOPC：CKSTOPフラグクリアビット
CKSTOPフラグをクリアするために、ソフトウェアによってセットされます。
0：CKSTOPはクリアされません。
1：CKSTOPはクリアされます。

ビット25 VSWENDC：VSWENDフラグクリアビット
VSWENDフラグをクリアするために、ソフトウェアによってセットされます。
0：VSWENDはクリアされません。
1：VSWENDはクリアされます。

ビット24 ACKTIMEOUTC：ACKTIMEOUTCフラグクリアビット
ACKTIMEOUTフラグをクリアするために、ソフトウェアによってセットされます。
0：ACKTIMEOUTはクリアされません。
1：ACKTIMEOUTはクリアされます。

ビット23 ACKFAILC：ACKFAILフラグクリアビット
ACKFAILフラグをクリアするために、ソフトウェアによってセットされます。
0：ACKFAILはクリアされません。
1：ACKFAILはクリアされます。

ビット22 SDIOITC：SDIOITフラグクリアビット
SDIOITフラグをクリアするために、ソフトウェアによってセットされます。
0：SDIOITはクリアされません。
1：SDIOITはクリアされます。

ビット21 BUSYD0ENDC：BUSYD0ENDフラグクリアビット
BUSYD0ENDフラグをクリアするために、ソフトウェアによってセットされます。
0：BUSYD0ENDはクリアされません。
1：BUSYD0ENDはクリアされます。

ビット20:12 予約済みであり、リセット値に保持する必要があります。

ビット11 DABORTC：DABORTフラグクリアビット
DABORTフラグをクリアするために、ソフトウェアによってセットされます。
0：DABORTはクリアされません。
1：DABORTはクリアされます。

ビット10 DBCKENDC：DBCKENDフラグクリアビット
DBCKENDフラグをクリアするために、ソフトウェアによってセットされます。
0：DBCKENDはクリアされません。
1：DBCKENDはクリアされます。

ビット9 DHOLDC：DHOLDフラグクリアビット
DHOLDフラグをクリアするために、ソフトウェアによってセットされます。
0：DHOLDはクリアされません。
1：DHOLDはクリアされます。

ビット8 DATAENDC：DATAENDフラグクリアビット
DATAENDフラグをクリアするために、ソフトウェアによってセットされます。
0：DATAENDはクリアされません。
1：DATAENDはクリアされます。

ビット7 CMDSENTC：CMDSENTフラグクリアビット
CMDSENTフラグをクリアするために、ソフトウェアによってセットされます。
0：CMDSENTはクリアされません。
1：CMDSENTはクリアされます。

ビット6 CMDRENDC：CMDRENDフラグクリアビット
CMDRENDフラグをクリアするために、ソフトウェアによってセットされます。
0：CMDRENDはクリアされません。
1：CMDRENDはクリアされます。

ビット5 RXOVERRC：RXOVERRフラグクリアビット
RXOVERRフラグをクリアするために、ソフトウェアによってセットされます。
0：RXOVERRはクリアされません。
1：RXOVERRはクリアされます。

ビット4 TXUNDERRC：TXUNDERRフラグクリアビット
TXUNDERRフラグをクリアするために、ソフトウェアによってセットされます。
0：TXUNDERRはクリアされません。
1：TXUNDERRはクリアされます。

ビット3 DTIMEOUTC：DTIMEOUTフラグクリアビット
DTIMEOUTフラグをクリアするために、ソフトウェアによってセットされます。
0：DTIMEOUTはクリアされません。
1：DTIMEOUTはクリアされます。

ビット2 CTIMEOUTC：CTIMEOUTフラグクリアビット
CTIMEOUTフラグをクリアするために、ソフトウェアによってセットされます。
0：CTIMEOUTはクリアされません。
1：CTIMEOUTはクリアされます。

ビット1 DCRCFAILC：DCRCFAILフラグクリアビット
DCRCFAILフラグをクリアするために、ソフトウェアによってセットされます。
0：DCRCFAILはクリアされません。
1：DCRCFAILはクリアされます。

ビット0 CCRCFAILC：CCRCFAILフラグクリアビット
CCRCFAILフラグをクリアするために、ソフトウェアによってセットされます。
0：CCRCFAILはクリアされません。
1：CCRCFAILはクリアされます。

31.10.13 SDMMCマスクレジスタ（SDMMC_MASKR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

割込みマスクレジスタは、対応するビットを1にセットすることによって、割込みリクエストを生成するステータスフラグを決めます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMABTCIE	Res.	CKSTOPIE	VSWENDIE	ACK（確認応答）TIMEOUTIE	ACK（確認応答）FAILIE	SDIOITIE	BUSYA0ENDIE	Res.	Res.	TXFIFOEIE	RXFIFOFIE	Res.
			rw		rw	rw	rw	rw	rw	rw			rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFIFOHFIE	TXFIFOHEIE	Res.	Res.	DABORTIE	DBCKENDIE	DHOLDIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 IDMABTCIE：IDMAバッファ転送完了割込みの有効化

IDMAがメモリバッファに属するすべてのデータを転送したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：IDMAバッファ転送完了割込みは無効です。

1：IDMAバッファ転送完了割込みは有効です。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 CKSTOPIE：電圧スイッチクロック停止割込みの有効化

電圧スイッチクロック停止による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチクロック停止割込みは無効です。

1：電圧スイッチクロック停止割込みは有効です。

ビット25 VSWENDIE：電圧スイッチの重要なタイミングセクション完了割込みの有効化

電圧スイッチの重要なタイミングセクションが完了したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチの重要なタイミングセクション完了割込みは無効です。

1：電圧スイッチの重要なタイミングセクション完了割込みは有効です。

ビット24 ACKTIMEOUTIE：確認応答タイムアウト割込みの有効化

確認応答タイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答タイムアウト割込みは無効です。

1：確認応答タイムアウト割込みは有効です。

ビット23 ACKFAILIE：確認応答失敗割込みの有効化

確認応答失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答失敗割込みは無効です。

1：確認応答失敗割込みは有効です。

ビット22 SDIOITIE：SDIOモード割込み受信割込みの有効化

SDIOモード割込みを受信したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：SDIOモード割込み受信割込みは無効です。

1：SDIOモード割込み受信割込みは有効です。

ビット21 BUSYD0ENDIE：BUSYD0ENDイベントOK割込みの有効化

SDMMC_D0信号がCMDレスポンスの後にビジーからビジーではない状態に変更したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：BUSYD0END割込みは無効です。

1：BUSYD0END割込みは有効です。

ビット20:19 予約済みであり、リセット値に保持する必要があります。

ビット18 TXFIFOEIE：Tx FIFOエンプティ割込みの有効化

Tx FIFOエンプティによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFOエンプティ割込みは無効です。

1：Tx FIFOエンプティ割込みは有効です。

ビット17 **RXFIFOFIE**：Rx FIFOフル割込みの有効化

Rx FIFOが満杯になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFOフル割込みは無効です。

1：Rx FIFOフル割込みは有効です。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **RXFIFOHFIE**：Rx FIFOハーフフル割込みの有効化

Rx FIFOがハーフフル状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFOハーフフル割込みは無効です。

1：Rx FIFOハーフフル割込みは有効です。

ビット14 **TXFIFOHEIE**：Tx FIFOハーフエンプティ割込みの有効化

Tx FIFOがハーフエンプティ状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFOハーフエンプティ割込みは無効です。

1：Tx FIFOハーフエンプティ割込みは有効です。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 DABORTIE：データ転送アボート割込みの有効化

アボートされているデータ転送による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ転送アボート割込みは無効です。

1：データ転送アボート割込みは有効です。

ビット10 DBCKENDIE：データブロックエンド割込みの有効化

データブロックエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データブロックエンド割込みは無効です。

1：データブロックエンド割込みは有効です。

ビット9 DHOLDIE：データホールド割込みの有効化

DPSMのWait_S状態で新しいデータの送信が保留されている場合に生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データホールド割込みは無効です。

1：データホールド割込みは有効です。

ビット8 **DATAENDIE**：データエンド割込みの有効化

データエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データエンド割込みは無効です。

1：データエンド割込みは有効です。

ビット7 CMDSENTIE：コマンド送信割込みの有効化

コマンド送信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド送信割込みは無効です。

1：コマンド送信割込みは有効です。

ビット6 CMDRENDIE：コマンドレスポンス受信割込みの有効化

コマンドレスポンスの受信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドレスポンス受信割込みは無効です。

1：コマンドレスポンス受信割込みは有効です。

ビット5 **RXOVERRIE**：Rx FIFOオーバーランエラー割込みの有効化

Rx FIFOオーバーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFOオーバーランエラー割込みは無効です。

1：Rx FIFOオーバーランエラー割込みは有効です。

ビット4 **TXUNDERRIE**：Tx FIFOアンダーランエラー割込みの有効化

Tx FIFOアンダーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFOアンダーランエラー割込みは無効です。

1：Tx FIFOアンダーランエラー割込みは有効です。

ビット3 DTIMEOUTIE：データタイムアウト割込みの有効化

データタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データタイムアウト割込みは無効です。

1：データタイムアウト割込みは有効です。

ビット2 CTIMEOUTIE：コマンドタイムアウト割込みの有効化

コマンドタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドタイムアウト割込みは無効です。

1：コマンドタイムアウト割込みは有効です。

ビット1 DCRCFAILIE：データCRC失敗割込みの有効化

データCRC失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データCRC失敗割込みは無効です。

1：データCRC失敗割込みは有効です。

ビット0 CCRCFAILIE：コマンドCRC失敗割込みの有効化

コマンドCRC失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドCRC失敗割込みは無効です。

1：コマンドCRC失敗割込みは有効です。

31.10.14 SDMMC確認応答タイマレジスタ（SDMMC_ACKTIMER）

アドレスオフセット：0x040

リセット値：0x0000 0000

SDMMC_ACKTIMERレジスタは、確認応答タイムアウト期間（SDMMC_CKバスクロック周期単位）を含みます。

カウンタはSDMMC_ACKTIMERレジスタから値をロードして、データパスステートマシン（DPSM）がWait_Ack状態に入ったときに、デクリメントを開始します。DPSMがこの状態のときにタイマが0に達した場合、確認応答タイムアウトステータスフラグがセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACKTIME[24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACKTIME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **ACKTIME[24:0]**：ブート確認応答タイムアウト期間
このビットは、CPSMが無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。
カードバスクロック周期で表されたブート確認応答タイムアウト期間です。

注： **データ転送は、データ制御レジスタに書き込まれる前に、確認応答タイマレジスタに書き込まれなければなりません。**

31.10.15 SDMMCデータFIFOレジスタx（SDMMC_FIFORx）

アドレスオフセット：0x080 + 0x004 * x（x = 0から15）

リセット値：0x0000 0000

受信および送信FIFOは、ワード（32bit）幅のレジスタとしてのみ読み書きできます。FIFOは、連続したアドレスに16個のエントリを含みます。このため、CPUは多重ロードおよびストアオペランドを使用してFIFOへの読み書きを行うことができます。FIFOレジスタインタフェースは、FIFO内でデータが正しく配置されるよう処理します。CPUが使用するFIFOレジスタアドレスは重要です。

ハーフワードまたはバイトアクセスでSDMMC_FIFORにアクセスすると、AHBバスフォールトが生成されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIFODATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIFODATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **FIFODATA[31:0]**：受信および送信FIFOデータ
このレジスタは、DPSMがアクティブ（DPSMACT = 1）のときのみファームウェアで読み書きできます。
FIFOデータは、32bitワードの16個のエントリを占有します。

31.10.16 SDMMC DMA制御レジスタ（SDMMC_IDMACTRLR）

アドレスオフセット：0x050

リセット値：0x0000 0000

受信および送信FIFOは、32bit幅のレジスタとして読み書きできます。FIFOは、32の連続したアドレスに32個のエントリを含みます。このため、CPUはそのロードとストアで複数のオペランドを使用して、FIFOから読み書きできるようになります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABMODE	IDMAEN
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **IDMABMODE**：バッファモードの選択

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：シングルバッファモード。

1：リンクリストモード。

ビット0 **IDMAEN**：IDMAの有効化

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：IDMAは無効です。

1：IDMAは有効です。

31.10.17 SDMMC IDMAバッファサイズレジスタ（SDMMC_IDMABSIZER）

アドレスオフセット：0x054

リセット値：0x0000 0000

リンクリストの設定では、SDMMC_IDMABSIZERレジスタにバッファサイズが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABNDT[11]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABNDT[10:0]											Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw					

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16:5 **IDMABNDT[11:0]**：バッファごとのバイト数

この12bit値は、32bitワードでのバッファサイズを得るには、8倍する必要があり、バイト単位でのバッファのサイズを得るには32倍する必要があります。

例：IDMABNDT = 0x001：バッファサイズ = 8ワード = 32バイト。

例：IDMABNDT = 0x800：バッファサイズ = 16384ワード = 64 KB。

これらのビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

31.10.18 SDMMC IDMAバッファベースアドレスレジスタ（SDMMC_IDMABASER）

アドレスオフセット：0x058

リセット値：0x0000 0000

SDMMC_IDMABASERレジスタには、シングルバッファ設定およびリンクリスト設定でメモリバッファベースアドレスが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABASE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABASE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r

ビット31:0 IDMABASE[31:0]：バッファメモリベースアドレスビット [31:2] は、ワード境界に合わせる必要があります（ビット [1:0] は常に0で読出し専用です）。

このレジスタは、DPSMがインアクティブ（DPSMACT = 0）のときファームウェアによって書き込むことができ、DPSMがアクティブ（DPSMACT = 1）のときはファームウェアによって動的に書き込むことができます。

31.10.19 SDMMC IDMAリンクリストのアドレスレジスタ（SDMMC_IDMALAR）

アドレスオフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ULA	ULS	ABR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMALA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット31 **ULA**：リンクリストモード（SDMMC_IDMACTRLR.IDMABMODEでリンクリストモードを選択）の場合、リンクリストからSDMMC_IDMALARを更新

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：SDMMC_IDMALARは更新されず、最後のリンクリスト項目です。

1：SDMMC_IDMALARは、リンクリストのテーブルから更新されます。

ビット30 **ULS**：リンクリストモードの場合、次のリンクリストからSDMMC_IDMABSIZEを更新（SDMMC_IDMACTRLR.IDMABMODEでリンクリストモードを選択し、ULA = 1）。

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：SDMMC_IDMABSIZERは、次のリンクリストのテーブルからは更新されません。

1：SDMMC_IDMABSIZERは、次のリンクリストのテーブルから更新されます。

ビット29 **ABR**：リンクリストバッファの準備完了確認応答

このビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ソフトウェアでプログラムされたレジスタ情報から最初のリンクリストバッファを開始する場合、このビットは考慮されません。ABRは、後続してロードされるリンクリストの項目においてのみ考慮されます。

0：ロード済みのリンクリストバッファの準備ができていません（これにより、リンクリストのIDMA転送エラーが生成されます）。

1：ロード済みのリンクリストバッファの準備完了確認応答。リンクリストバッファのデータはIDMAで転送されます。

ビット28:16 予約済みであり、リセット値に保持する必要があります。

ビット15:2 **IDMALA[13:0]**：ワード調整済みのリンクリスト項目のアドレスのオフセット

次のリンクリスト項目構造のベースへの、リンクリスト項目のオフセットポインタ。

リンクリスト項目のベースアドレスはIDMABA + IDMALAとなります。

これらのビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

31.10.20 SDMMC IDMAリンクリストメモリのベースレジスタ（SDMMC_IDMABAR）

アドレスオフセット：0x068

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABA[29:14]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット31:2 **IDMABA[29:0]**：ワード調整済みのリンクリストメモリのベースアドレス

リンクリストメモリのベースポインタ。

これらのビットは、DPSMがインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

31.10.21 SDMMCレジスタマップ

表292.SDMMCレジスタマップ

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	SDMMC_POWER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIRPOL	VSWITCHEN	VSWITCH	PWRCTRL[1:0]	
	リセット値																												0	0	0	0	0
0x004	SDMMC_CLKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUSSPEED	DDR	HWFC_EN	NEGEDGE	WIDBUS[1:0]		Res.	PWRSAV	Res.	Res.	CLKDIV[9:0].									
	リセット値											0	0	0	0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0
0x008	SDMMC_ARGR	CMDARG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	SDMMC_CMDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDSUSPEND	BOOTEN	BOOTMODE	DTHOLD	CPSMEN	WAITPEND	WAITINT	WAITRESP[1:0]		CMDSTOP	CMDTRANS	CMDINDEX[5:0]					
	リセット値																																
0x010	SDMMC_RESPCMDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
	リセット値																											0	0	0	0	0	0
0x014	SDMMC_RESP1R	CARDSTATUS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	SDMMC_RESP2R	CARDSTATUS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	SDMMC_RESP3R	CARDSTATUS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	SDMMC_RESP4R	CARDSTATUS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	SDMMC_DTIMER	DATATIME[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	SDMMC_DLENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATALENGTH[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	SDMMC_DCTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFORST	BOOTACKEN	SDIOEN	RWMOD	RWSTOP	RWSTART	DBLOCKSIZE[3:0]			DTMODE[1:0]		DTDIR	DTEN	
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表292.SDMMCレジスタマップ （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x030	SDMMC_ DCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	SDMMC_ STAR	Res.	Res.	Res.	IDMABTC	IDMATE	CKSTOP	VSWEND	ACKTIMEOUT	ACKFAIL	SDIOIT	BUSYD0END	BUSYD0	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF	RXFIFOHF	TXFIFOHE	CPSMACT	DPSMACT	DABORT	DBCKEND	DHOLD	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	SDMMC_ ICR	Res.	Res.	Res.	IDMABTCC	IDMATEC	CKSTOPC	VSWENDC	ACKTIMEOUTC	ACKFAILC	SDIOITC	BUSYD0ENDC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DABORTC	DBCKENDC	DHOLDC	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
	リセット値				0	0	0	0	0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0
0x03C	SDMMC_ MASKR	Res.	Res.	Res.	IDMABTCIE	Res.	CKSTOPIE	VSWENDIE	ACKTIMEOUTIE	ACKFAILIE	SDIOITIE	BUSYD0ENDIE	Res.	Res.	TXFIFOEIE	RXFIFOFIE	Res.	RXFIFOHFIE	TXFIFOHEIE	Res.	Res.	DABORTIE	DBCKENDIE	DHOLDIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
	リセット値				0		0	0	0	0	0	0			0	0		0	0			0	0	0	0	0	0	0	0	0	0	0	0
0x040	SDMMC_ ACKTIMER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACKTIME[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044 - 0x04C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x050	SDMMC_ IDMACTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABMODE	IDMAEN
	リセット値																															0	0
0x054	SDMMC_ IDMABSIZER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABNDT[11:0]												Res.	Res.	Res.	Res.	Res.
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0					
0x058	SDMMC_ IDMABASER	IDMABASE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C - 0x060	予約済み	Res.																															
0x064	SDMMC_ IDMALAR	ULA	ULS	ABR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMALA[13:0]														Res.	Res.
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x068	SDMMC_ IDMABAR	IDMABA[29:0]																													Res.	Res.	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x06C - 0x07C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表292.SDMMCレジスタマップ （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080 + 0x04 * x （x = 0 ～ 15）	SDMMC_FIFOR	FIFODATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

32 遅延ブロック（DLYB）

32.1 概要

遅延ブロック（DLYB）は、入力クロックからディフェーズ（位相をずら）した出力クロックを生成するために使用します。出力クロックのフェーズは、ユーザアプリケーションでプログラムする必要があります。次に、出力クロックを使用して、SDMMCまたはOcto-SPI インタフェースなどの別のペリフェラルで受信するデータにクロック供給します。

遅延は電圧や温度に依存するため、場合によってはアプリケーションで出力クロックの位相を再設定し、受信データと中央揃えを再度行う必要があるかもしれません。

32.2 DLYBの主な機能

遅延ブロックは、次のような機能を備えています。

- 入力クロック周波数は25MHzから通信インタフェースでサポートされる最大周波数までの範囲（データシートを参照）
- 最大12のオーバーサンプリングフェーズ

32.3 DLYBの実装

表293.STM32U5シリーズ機能

DLYB関連のペリフェラル	DLYBOS1	DLYBOS2	DLYBSD1	DLYBSD2
OCTOSPI1	X	-	-	-
OCTOSPI2	-	X	-	-
SDMMC1	-	-	X	
SDMMC2	-	-	-	X

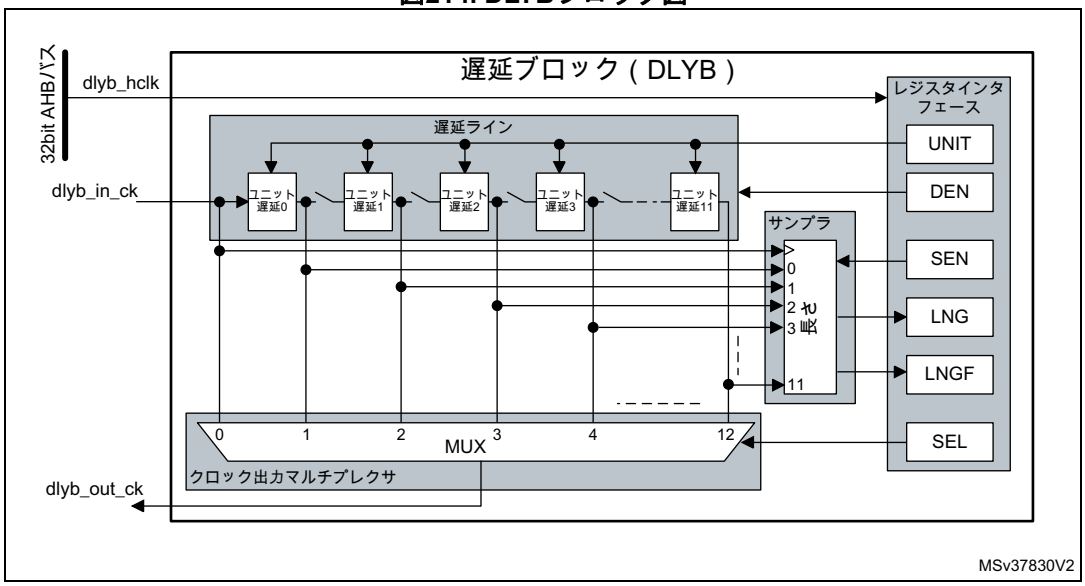
32.4 DLYBの機能説明

32.4.1 DLYB図

遅延ブロックは、次のようなサブブロックから成ります（次の図に示します）。

- DLYBレジスタへのAHBアクセスを可能にするレジスタインタフェースブロック
- ユニット遅延をサポートする遅延ライン
- 遅延ライン長さサンプラ
- 出力クロック選択マルチプレクサ

図214. DLYBブロック図



32.4.2 DLYBピンおよび内部信号

表294 にDLYB内部信号を示します。

表294.DLYB内部入力／出力信号

信号名	信号タイプ	説明
dlyb_hclk	デジタル入力	遅延ブロックレジスタのインタフェースクロック
dlyb_in_ck	デジタル入力	遅延ブロック入力クロック
dlyb_out_ck	デジタル出力	遅延ブロック出力クロック

32.4.3 概要

遅延ブロックは、DLYB制御レジスタ（DLYB_CR）のDENビットをセットすることによって有効にします。長さサンプラは、DLYB_CRレジスタのSENビットによって有効にします。

遅延ブロックが有効なとき、ユニット遅延によって追加される遅延は DLYB 設定レジスタ（DLYB_CFGR）のUNIT[6:0] フィールドで定義します。

注： **UNIT[6:0] は、出力クロックが無効（SEN = 1）のときのみプログラム可能です。**

遅延ブロックが有効なとき、出力クロックの位相はDLYB_CFGRレジスタのSEL[3:0] フィールドによって選択します。

注： **SELは出力クロックが無効（SEN = 1）のときのみプログラム可能です。**

遅延ラインの長さを1周期にまたがるように設定することにより、出力クロックは入力クロックの1周期にわたって位相をずらすことができます。遅延ラインの長さはSENビットによって長さサンプラを有効にすることで設定でき、それによりDLYB_CFGRの遅延ライン長さ（LNG[11:0] および長さ有効フラグ（LNGF）へのアクセスが可能になります。

入力クロックの1周期より短い出力クロック遅延が必要な場合は、遅延線の長さを短くすることができます。これにより、より小さなユニット遅延が可能になり、より高い分解能が得られます。

遅延ライン長さが設定されると、出力クロックマルチプレクサによって、位相がずらされた出力クロックを選択することができます。この選択はSEL[3:0] で行います。出力クロックはSENが0にセットされている場合にのみ、選択された位相で使用できます。

下の表に、遅延ブロック制御の概要を示します。

表295.遅延ブロック制御

DEN	SEN	UNIT	SEL	LNG	LNGF	出力クロック
0	0	無視	無視	無視	無視	有効（ = 入力クロック）
x	1	ユニット遅延	出力クロックフェーズ	長さ	長さフラグ	無効
1	0	ユニット遅延 (1)	出力クロックフェーズ (2)	無視	無視	有効（ = 選択されたフェーズ）

1. ユニット遅延はSEN = 1の場合にのみ変更できます。
2. 出力クロックの位相はSEN = 1の場合にのみ変更できます。

32.4.4 遅延ライン長さ設定手順

LNG[11:0] を使用して、入力クロック周期に対する遅延ライン長さを決定します。遅延ライン長さは、入力クロックの1周期全体をカバーできるような長さに設定しなければなりません。

遅延線には12個のユニット遅延要素がありますが、次の手順の説明では0～10の間の長さが返されることに注意してください。1入力クロックサイクル全体にわたって遅延が調整されることを保証するために、最上位の遅延出力値が使用されるからです。クロック周波数およびUNIT値によっては、ユニット遅延要素10もクロックサイクル長から切捨てられることがあります。

チューニング手順全体を通じて、クロック入力（フリーランニングクロック）が必要です。

入力クロック1周期に対する遅延ライン長さを設定するには、以下のシーケンスに従います。

1. DENビットを1に設定して遅延ブロックを有効にします。
2. SENビットを1に設定して長さサンプリングを有効にします。
3. SEL[3:0] を12に設定してすべての遅延セルを有効にします。
4. UNIT[6:0] = 0～127に対して（遅延ライン長さが設定されるまで、このステップを繰り返すこと）:
   a) UNIT[6:0] の値を更新し、長さフラグLNGFが1にセットされるのを待ちます。
   b) LNG[11:0] を読み出します。
   LNG[10:0] > 0、かつLNG[11] またはLNG[10] = 0の場合、遅延ライン長さは入力クロック1周期に対して設定されます。
5. 入力クロック1周期にわたるユニット遅延の数（N）を決定します。N = 0～10について、LNG[N] = 1の場合、入力クロック周期にわたるユニット遅延の数 = Nとなります。
6. SENビットを0クリアして長さサンプリングを無効にします。

入力クロックの1周期より短い出力クロック遅延が必要な場合は、遅延線の長さを入力クロックの1周期より短くすることができます。これにより、より小さなユニット遅延が可能になり、短い時間間隔にわたってより高い分解能が得られます。

32.4.5 出力クロック位相の設定手順

遅延ライン長さを入力クロック1周期に対して設定する場合、入力クロック1周期の期間中に発生するユニット遅延の間で出力クロックフェーズを選択することができます。

出力クロックフェーズを選択するには以下の手順に従います。

1. SENビットを1にセットすることで、出力クロックを無効にし、位相選択SEL[3:0] ビットへのアクセスを有効にします。
2. 希望する出力クロックの位相値でSEL[3:0] をプログラムします。
3. SENを0クリアすることにより、選択された位相で出力クロックを有効にします。

32.5 DLYBレジスタ

すべてのレジスタはワード、ハーフワード、バイトでアクセスすることができます。

32.5.1 DLYB制御レジスタ（DLYB_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEN	DEN
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SEN**：サンプラ長さ有効ビット

0：サンプラ長さと、UNIT[6:0] およびSEL[3:0] へのレジスタのアクセスは無効、出力クロックは有効です。

1：サンプラ長さと、UNIT[6:0] およびSEL[3:0] へのレジスタのアクセスは有効、出力クロックは無効です。

ビット0 **DEN**：遅延ブロック有効ビット

0：DLYBは無効です。

1：DLYBは有効です。

32.5.2 DLYB設定レジスタ（DLYB_CFGR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LNGF	Res.	Res.	Res.	LNG[11:0]											
r				r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	UNIT[6:0]							Res.	Res.	Res.	Res.	SEL[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

ビット31 **LNGF**：長さ有効フラグ

このフラグはUNIT[6:0] ビットの変更後、LNG[11:0] に含まれる遅延ライン長さの値が有効になった時を知らせます。

0：LNGの長さの値は無効です。

1：LNGの長さの値は有効です。

ビット30:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **LNG[11:0]**：遅延ライン長さの値

これらのビットには、入力クロックの立上がりエッジでサンプリングされた12個のユニット遅延の値が反映されます。

値はLNGF = 1のときにのみ有効です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **UNIT[6:0]**：ユニット遅延セルの遅延。

これらのビットは、SEN = 1のときにのみ書き込むことができます。

ユニット遅延 = 初期遅延 + UNIT[6:0] x遅延ステップ（刻み）

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **SEL[3:0]**：出力クロックの位相。

これらのビットは、SEN = 1のときにのみ書き込むことができます。

出力クロックの位相 = 入力クロック + SEL[3:0] xユニット遅延

32.5.3 DLYBレジスタマップ

表296.DLYBレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DLYB_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEN	DEN
	リセット値																															0	0
0x004	DLYB_CFGR	LNGF	Res.	Res.	Res.	LNG												Res.	UNIT							Res.	Res.	Res.	Res.	SEL			
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

33 アナログデジタルコンバータ（ADC12）

STM32U535/545/575/585デバイスには1台のアナログデジタルコンバータ（ADC1）が内蔵され、STM32U59x/5Ax/5Fx/5Gxデバイスには2台のアナログデジタルコンバータ（ADC1およびADC2）が内蔵されていて、これは単一のインタフェースADCによって制御されます。

33.1 概要

このセクションでは、14bitのADC逐次比較アナログデジタルコンバータの実装について説明します。

ADC1およびADC2は密接に結合されており、デュアルモードで動作できます（ADC1はマスタです）。

ADCには、最大20の多重化チャネルがあります。チャネルのA/D変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADCの結果は、左詰めまたは右詰めで32bitのデータレジスタに格納できます。

ADCはAHBバスに配置され、データを高速で処理できます。

また、アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

ADCに組み込まれたハードウェアのオーバーサンプリング回路により、CPUの計算負荷を軽減しながらアナログ性能を高めることができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードも実装されています。

33.2 ADCの主な機能

- ハイパフォーマンス機能
  - デュアルモード動作（セクション 33.3：A/Dコンバータの実装を参照）
  - 14、12、10、または8bitに設定可能な分解能
  - ADC変換時間はAHBバスクロック周波数から独立
  - 分解能を下げることで変換時間を短縮可
  - シングルエンドまたは差動入力（チャネルごとにプログラム可能）の管理
  - 高速データ処理のためのAHBスレーブバスインタフェース
  - 自己較正（オフセットと直線性とも）
  - チャネル単位でプログラム可能なサンプリング時間
  - 柔軟なサンプリング時間制御
  - 最大4つのインジェクトチャネル（レギュラまたはインジェクトチャネルへのアナログ入力の割当てを完全に設定可能）
  - インジェクトチャネルコンテキストを準備し、高速コンテキスト切り替えを可能にするのに役立つハードウェア
  - 組み込みのデータコヒーレンシによるデータ配置
  - FIFOを用いたレギュラチャネル変換のための、汎用DMAによるデータ管理
  - 後処理のためのMDFへのデータルーティング
  - 4つのインジェクトチャネル専用のデータレジスタ
- オーバーサンプリング回路
  - 32bitデータレジスタ

  - 2 ~ 1024倍の範囲でオーバーサンプリング比を調整可能
  - プログラム可能なデータの右左シフト
- データプレコンディショニング
  - ゲイン補正
  - オフセット補正
- 低電力機能
  - 低周波数での動作時にADC消費電流を低減するための速度適応型低電力モード
  - 最適なADC性能を維持しながら、低いバス周波数の適用をサポート
  - AHBバスクロック低周波数アプリケーション（自動遅延モード）でADCオーバーランを避けるための自動制御
- 専用のGPIOパッドに接続された最大 17 の外部アナログ入力チャネル
- 3 個の内部専用チャネル
  - 内部基準電圧用の1つのチャネル（V_{REFINT}）
  - 内部温度センサ用の1つのチャネル（V_{SENSE}）
  - V_{BAT} 監視チャネル用の1つのチャネル（$V_{BAT}/4$）
- 変換開始は、次のように開始できます。
  - レギュラ変換とインジェクト変換の両方について、ソフトウェアによって開始、または
  - レギュラ変換とインジェクト変換の両方について、極性が設定可能なハードウェアトリガによって開始（内部タイマイベントまたはGPIO入力イベント）
- 変換モード
  - シングルモード：ADC により 1 つのチャネルを変換します。変換は特別なイベントによってトリガされます。
  - スキャンモード：ADCにより一連のチャネルをスキャンし、変換します。
  - 連続モード：ADCにより、選択した入力を連続的に変換します。
  - 不連続モード：ADCにより変換シーケンスのサブセットを変換します。
- ADC のレディ時、サンプリング終了時、（レギュラまたはインジェクト）変換終了時、（レギュラまたはインジェクト）シーケンス変換終了時、アナログウォッチドッグ1、2、3、またはオーバーランイベント発生時に割込みを生成
- 3つのアナログウォッチドッグ

  ウォッチドッグはフィルタリングを行って、範囲外のデータを無視できます。
- ADC入力電圧範囲：$V_{REF_} \le V_{IN} \le V_{REF+}$

図215 に、1つのADCのブロック図を示します。

33.3 A/Dコンバータの実装

次の表に、STM32U5シリーズデバイスでのADC実装を示します。比較のため、表にはADC4も含めています。

表297.A/Dコンバータの機能(1)

A/Dコンバータのモード／機能	STM32U535/545/575/585	STM32U59x/5Ax/5Fx/5Gx		STM32U535/545/575/585/59x/5Ax/5Fx/5Gx
	ADC1	ADC1	ADC2	ADC4
分解能	14bit			12bit
14bit分解能用の最大サンプリング速度	2.5Msps			2.5Msps
ハードウェアオフセット較正	X			X
ハードウェア直線性較正	X			-
拡張較正モード	X(2)	X		-
シングルエンド入力	X			X
差動入力	X			-
インジェクトチャネル変換	X			-
オーバーサンプリング	最大x1024			最大x256
データレジスタ	32bit			16bit
DMAサポート	X			X
MDFへのパラレルデータ出力	X			-
デュアル･モード	-	X		-
自律モード	-			X
オフセット補正	X			-
ゲイン補正	X			-
アナログウォッチドッグの数	3			3
STOPモードからのウェイクアップ	-			X(3)

1. 注：「X」はサポート対象、「-」はサポート対象外です。
2. STM32U575/585の場合、デバイスリビジョン X では拡張較正モードはサポートされていません。
3. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

表298.温度センサ較正値のメモリ位置

名前	説明	メモリアドレス
TS_CAL1	30°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）でADC1によって取得された温度センサの14bit元データ	0x0BFA0710 ~ 0x0BFA0711
TS_CAL2	130°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）でADC1によって取得された温度センサの14bit元データ	0x0BFA0742 ~ 0x0BFA0743

表299.内部基準電圧センサの較正値のメモリ位置

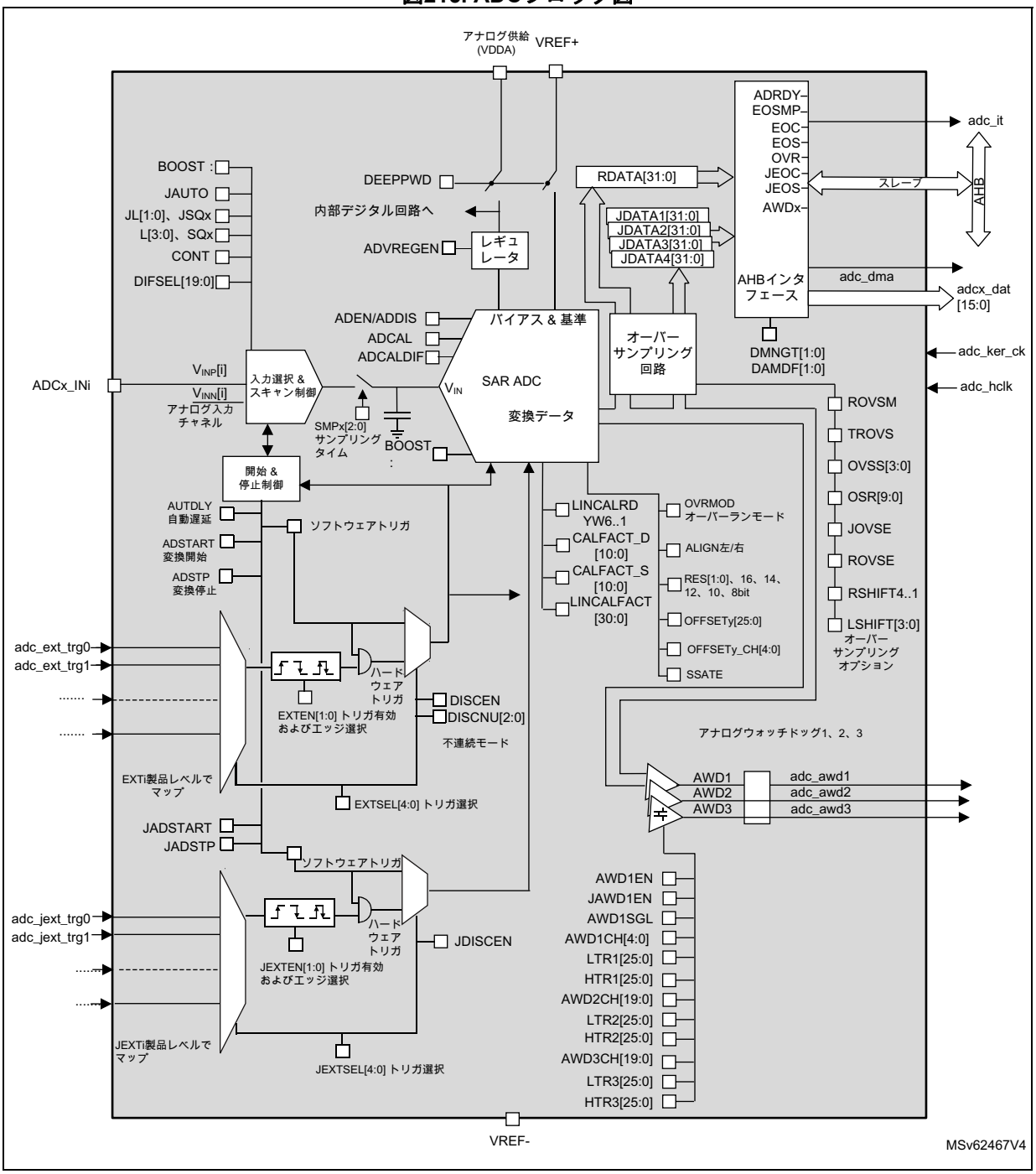
名前	説明	メモリアドレス
VREFINT_CAL	30°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）でADC1によって取得された14bit元データ	0x0BFA 07A5 ~ 0x0BFA 07A6

33.4 ADCの機能詳細

33.4.1 ADCのブロック図

図215 にADCブロック図を、表300 にADCピンの説明を示します。

図215. ADCブロック図



33.4.2 A/Dコンバータのピンおよび内部信号

表300.A/Dコンバータの各入出力ピン

名前	信号タイプ	説明
VREF+	入力、アナログ基準電圧正	ADCのハイレベル／正基準電圧
VDDA	入力、アナログ電源供給	V_{DDA} に等しいアナログ電源電圧
VREF-	入力、アナログ基準電圧負	ADCのローレベル／負基準電圧、VREF– = VSSA
VSSA	入力、アナログ供給グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド
ADCx_INy	外部アナログ入力信号	最大 17 の外部アナログ入力チャネル

表301.ADC内部入力／出力信号

内部信号名	信号タイプ	説明
V_{INP}[i]	アナログ入力	各ADCの正のアナログ入力チャネル
V_{INN}[i]	アナログ入力	各ADCの負のアナログ入力チャネル
adc_ext_trgy	入力	レギュラ変換用の外部トリガ入力（オンチップタイマに接続可能）。
adc_jext_trgy	入力	インジェクト変換用の外部トリガ入力（オンチップタイマに接続可能）。
adc_awd1 adc_awd2 adc_awd3	出力	オンチップタイマに接続された内部アナログウォッチドッグの出力信号
adc_it	出力	ADC割込み
adc_hclk	入力	AHBクロック
adc_ker_ck	入力	A/Dコンバータのカーネル·クロック
adc_dma	出力	ADC DMAリクエスト
adcx_dat[15:0]	出力	ADCデータ出力（レギュラデータレジスタ）

表302.ADC1/ADC12相互接続

信号名	転送元／転送先
ADCx V_{INP}[0]（x = 1、2）	V_{REFINT} バッファ付き電圧
ADCx V_{INP}[18]（x = 1, 2）	V_{BAT}/4
ADCx V_{INP}[19]（x = 1, 2）	V_{SENSE}
adcx_dat[15:0]（x = 1、2）	mdf1_adcx_dat [15:0]

表303.ADC1/ADC12 のレギュラチャネル用外部トリガ

名前	転送元
adc_ext_trg0	tim1_oc1
adc_ext_trg1	tim1_oc2
adc_ext_trg2	tim1_oc3
adc_ext_trg3	tim2_oc2
adc_ext_trg4	tim3_trgo

表303.ADC1/ADC12 のレギュラチャネル用外部トリガ （続き）

名前	転送元
adc_ext_trg5	tim4_oc4
adc_ext_trg6	exti11
adc_ext_trg7	tim8_trgo
adc_ext_trg8	tim8_trgo2
adc_ext_trg9	tim1_trgo
adc_ext_trg10	tim1_trgo2
adc_ext_trg11	tim2_trgo
adc_ext_trg12	tim4_trgo
adc_ext_trg13	tim6_trgo
adc_ext_trg14	tim15_trgo
adc_ext_trg15	tim3_oc4
adc_ext_trg16	exti15
adc_ext_trg18	lptim1_ch1
adc_ext_trg19	lptim2_ch1
adc_ext_trg20	lptim3_ch1
adc_ext_trg21	lptim4_out

表304.ADC1/ADC12のインジェクトチャネルの外部トリガ

名前	転送元
adc_jext_trg0	tim1_trgo
adc_jext_trg1	tim1_oc4
adc_jext_trg2	tim2_trgo
adc_jext_trg3	tim2_oc1
adc_jext_trg4	tim3_oc4
adc_jext_trg5	tim4_trgo
adc_jext_trg6	exti15
adc_jext_trg7	tim8_oc4
adc_jext_trg8	tim1_trgo2
adc_jext_trg9	tim8_trgo
adc_jext_trg10	tim8_trgo2
adc_jext_trg11	tim3_oc3
adc_jext_trg12	tim3_trgo
adc_jext_trg13	tim3_oc1
adc_jext_trg14	tim6_trgo
adc_jext_trg15	tim15_trgo
adc_jext_trg18	lptim1_ch2
adc_jext_trg19	lptim2_ch2

表304.ADC1/ADC12のインジェクトチャネルの外部トリガ （続き）

名前	転送元
adc_jext_trg20	lptim3_ch1
adc_jext_trg21	lptim4_out1

33.4.3 A/Dコンバータのクロック

デュアルクロックドメインアーキテクチャ

デュアルクロックドメインアーキテクチャでは、ADCカーネル·クロックがADCレジスタへのアクセスのためのAHBバスクロックから独立しています。

adc_ker_ck入力クロックは、次の2つの異なるクロックソースから選択できます（図 216 : ADCクロック構成を参照）。この選択は、RCCで行われます（詳細はRCCのセクションを参照）。

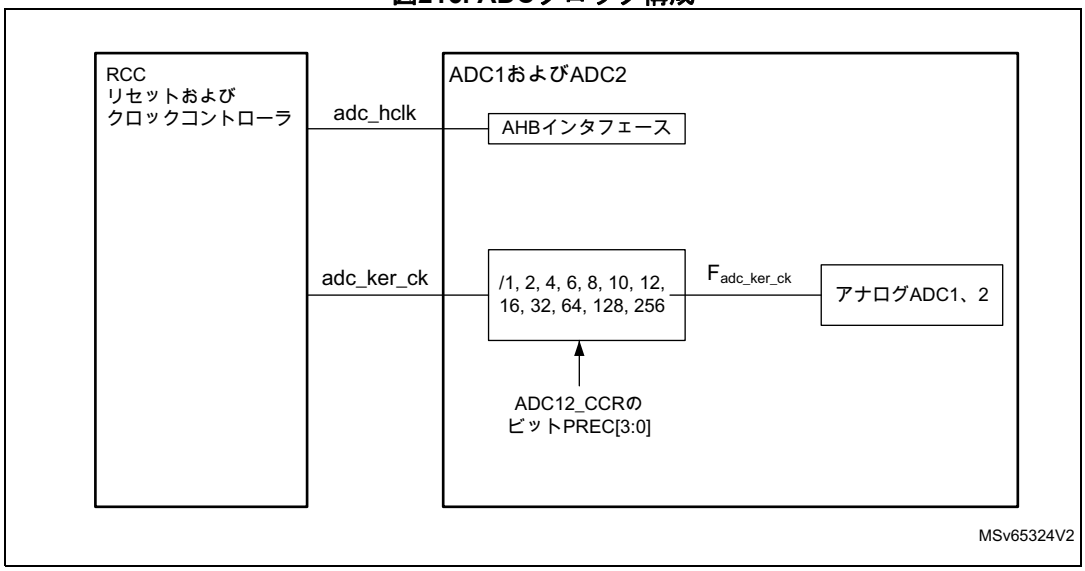
1. ADC クロックは、AHB クロックから独立し、これと同期される内部または外部クロックソースから供給できます。
2. ADCクロックはAHBから生成できます。

オプション **1**には、選択されたAHBクロック構成にかかわらず、最大のADCクロック周波数に到達できるという利点があります。ADCクロックは、最終的には1、2、4、6、8、10、12、16、32、64、128、または256の比率で分周できます。このとき、ADC12_CCRレジスタのPRESC[3:0] ビットで設定されたプリスケーラを使用します。

オプション **2** では、クロック·ドメインの再同期をバイパスできます。これは、ADCがタイマによってトリガされるときと、アプリケーションがADCの確実で精密なトリガを必要とする場合に便利です（そうしないと、トリガインスタンスの不確実性は、2つのクロック·ドメイン間の再同期によって高まります）。

クロックはRCCを通して設定されます。これは、デバイスデータシートに指定された動作周波数と一致している必要があります。

図216. ADCクロック構成



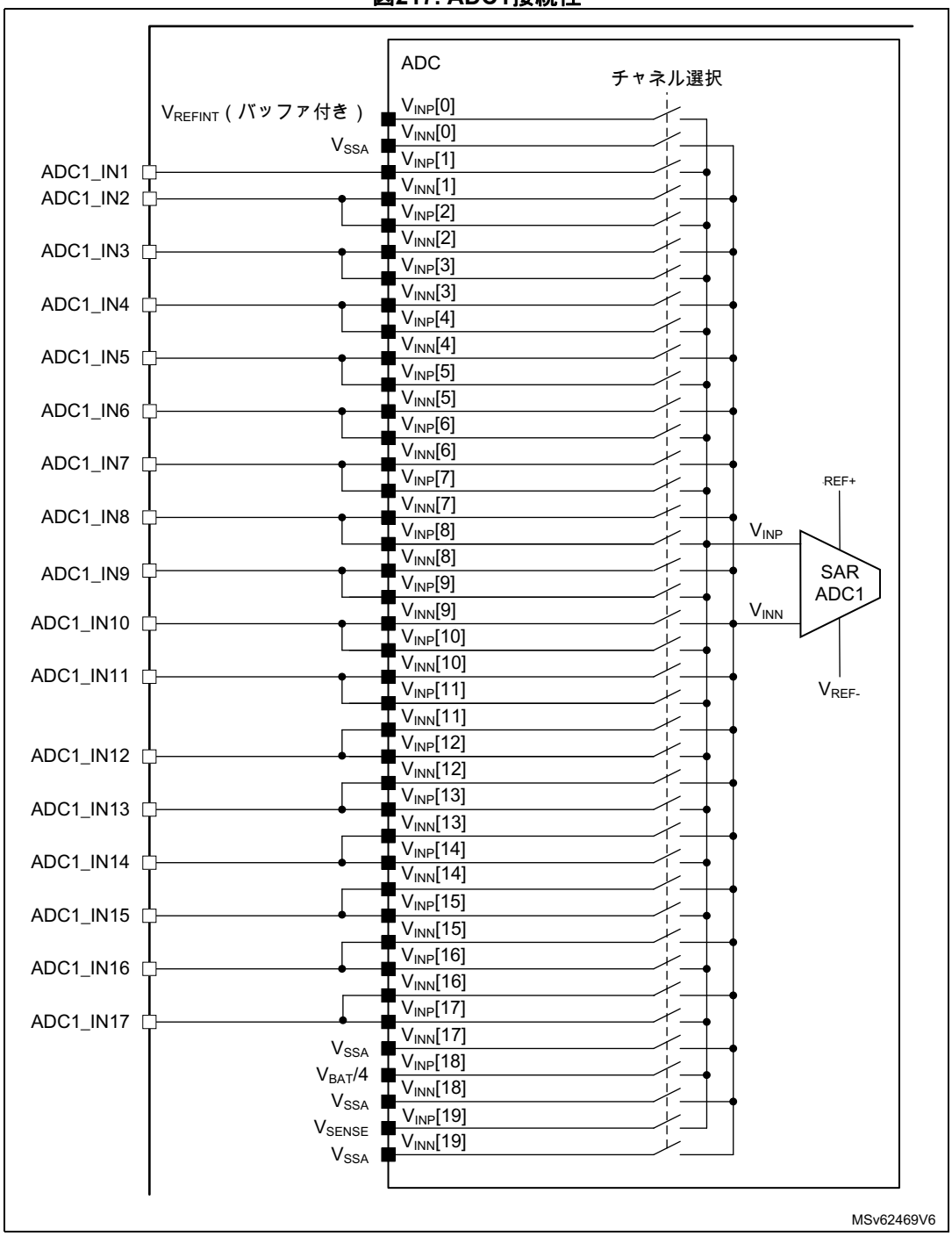
1. adc_hclkおよびadc_ker_ckの生成の詳細については、RCCのセクションを参照してください。

A/Dコンバータのクロックと AHBのクロックとの比率に対する制約

通常、ADCクロックとAHBクロックの間で注意すべき制約はありません。ただし、クロックAHBがADCクロックよりもはるかに低速である場合は、オーバーランを避けるために比率の選択を慎重に行う必要があります。

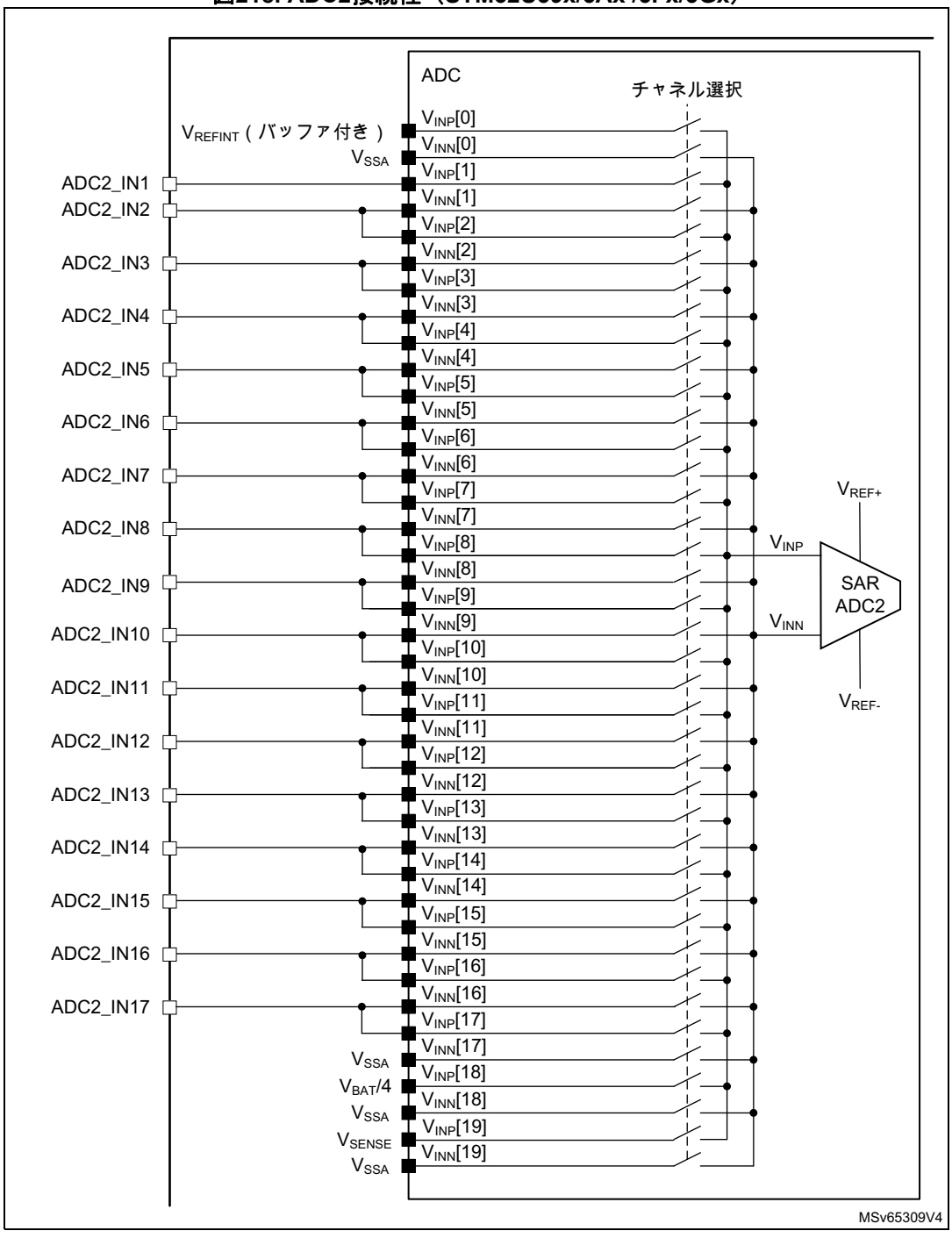
33.4.4 ADC接続性

図217. ADC1接続性



1. 対応するADC入力チャネルが差動モードとして設定された場合に限り、$V_{INN}[i]$ 信号を使用できます。

図218. ADC2接続性（STM32U59x/5Ax /5Fx/5Gx）



33.4.5 スレーブAHBインタフェース

ADCは、制御／ステータスレジスタおよびデータアクセスのためにAHBスレーブポートを実装しています。AHBインタフェースの機能を次に示します。

- ワード（32bit）アクセス
- シングルサイクルレスポンス
- レジスタへのすべての読出し／書込みアクセスに対してゼロウェイトステートの応答。

AHBスレーブインタフェースでは分割／再試行リクエストをサポートしておらず、またAHBエラーも発生しません。

33.4.6 ADCディープ･パワー･ダウン･モード（DEEPPWD）およびADC電圧レギュレータ（ADVREGEN）

デフォルトでは、リーク電流を低減するために電源電圧が内部的にオフになっている（ADC_CRレジスタのビットDEEPPWDのリセット状態が1である）場合、ADCはディープ･パワー･ダウン･モードになります。

A/Dコンバータの動作を開始するには、以下のシーケンスを実行します。

1. まず、DEEPPWDビットをクリアしてディープ･パワー･ダウン･モードを終了します。
2. そして、ADC_CRレジスタのADVREGENビットをセットして、ADC内部電圧レギュレータを有効にします。ソフトウェアでは、較正を開始したりADCを有効化したりする前に、ADC電圧レギュレータの起動時間（$T_{ADCVREG_STUP}$）を待つ必要があります。これを行うには、ソフトウェアにより、ADC_ISRレジスタのLDORDYビットをポーリングします。

ADCの動作が完了すると、ADCを無効化できます（ADEN = 0）。ADC電圧レギュレータを無効にすることでも節電できます。これにはADVREGENビットをクリアします。電力消費をさらに低減するには、漏れ電流を低減します。また、ADCディープ･パワー･ダウン･モードに戻すには、ADC_CRレジスタのDEEPPWDビットをセットします。これは特にSTOPモードに移行する前に有益です。

注： **DEEPPWDをセットすると、ADC電圧レギュレータは自動的に無効になり、ビットADVREGENが自動的にクリアされます。**

内部電圧レギュレータが無効な状態（ADVREGEN = 0）でも、内部アナログ較正係数は保持されます。

ADCディープ･パワー･ダウン･モード（DEEPPWD = 1）では、内部アナログ較正が失われ、較正を再起動するか、以前に保存した較正係数を再適用する必要があります（セクション 33.4.8：較正（ADCAL、ADCALLIN、ADC_CALFACT）を参照）。

33.4.7 シングルエンドおよび差動入力のチャネル

ADCチャネルは、シングルエンド入力または差動入力として設定できます。この設定を行うには、ADC_DIFSEL レジスタのDIFSEL[19:0] ビットに書き込みます。この設定は、ADCが無効化された（ADEN = 0）状態で行う必要があります。

シングルエンド入力モードで、チャネル「i」用に変換されるアナログ電圧は、外部電圧V_{INP}[i]（正の入力）とV_{REF-}（負の入力）の差です。

差動入力モードで、チャネル「i」用に変換されるアナログ電圧は、外部電圧V_{INP}[i]（正の入力）とV_{INN}[i]（負の入力）の差です。

差動モードの出力データは、符号なしデータです。

- V_{INP}[i] = V_{REF-} のとき：V_{INN}[i] = V_{REF+}および出力データ = 0x0000（14bit分解能モード）
- V_{INP}[i] = V_{REF+} のとき：V_{INN}[i] = V_{REF-} および出力データ = 0x3FFF

$$\text{Converted value} = \frac{\text{ADC_Full_Scale}}{2} \times \left[1 + \frac{V_{INP} - V_{INN}}{V_{VREF+REF+}}\right]$$

ADCが差動モードに設定されている場合、両方の入力をV_{REF+} / 2の電圧でバイアスする必要があります。

入力チャネルの接続方法の詳細については、セクション 33.4.4：ADC接続性を参照してください。

注意： **差動入力モードでチャネル「i」を設定する場合、負の入力電圧はV_{INN}[i-1] に接続されます。**

33.4.8 較正（ADCAL、ADCALLIN、ADC_CALFACT）

ADCには、ADCパワーオン／オフを含む較正シーケンス全体を制御する自動キャリブレーション手順が用意されています。この手順の間に、ADCによりシングルエンドおよび差動モード用のオフセット較正係数が計算されます。この係数には、次のADCパワーオフまでADCに内部的に適用される内部オフセットと直線性が含まれます。較正処理時、アプリケーションではADCを使用してはならず、較正が完了するまで待つ必要があります。

較正は、すべてのADCの動作の前提となります。較正によって、チップごとに異なる系統的エラーが除去され、オフセットと直線性の偏差を補償することができます。

オフセット較正は、シングルエンドチャネルと差動チャネルで共通です。

シングル構成と差動構成のいずれであっても、直線性の補正は一度だけとする必要があります。

- ADC_CRのADCALLINをセットしてから較正を起動します。この場合、直線性較正とオフセット較正が同時に、または
- ADC_CR の ADCALLIN をクリアしてから較正を起動します。この場合、直線性較正は実行されず、オフセット較正のみが実行されます。

較正は、ADCALビットをセットすることによって、ソフトウェアによって開始されます。較正は、ADCが無効のとき（ADEN = 0のとき）だけ開始できます。すべての較正シーケンス時、ADCALビットは1のままです。較正が完了するとすぐに、ハードウェアによってクリアされます。この時点で、関連する較正係数はアナログADCに内部的に格納されます。

ADCが無効の場合（ADEN = 0）、内部アナログ較正が保持されます。ただし、長時間にわたってADCが無効化されている場合、温度が変化する場合、または供給電圧を 10% 以上変える場合、ADCを再度有効化する前に、新しいオフセット較正サイクルを実行することを推奨します。

内部アナログ較正は、ADCの電源がオフになるたびに失われます（たとえば、デバイスがSTANDBYまたはVBATモードになった場合）。この場合、ADCの再較正に時間をかけるのを避けるために、前回の較正時に生成された較正係数が保存されていれば、再較正せずにADCアナログブロックに較正係数を再度書き込むことができます。

工場出荷時のテスト中に取得された較正係数が、特定のデバイスオプションバイトにプログラムされます。較正時間を短縮するため、この値をアナログADCにコピーすることができます。

ADCが有効であり、較正の実行中でない（ADEN = 1で、ADSTART = 0かつJADSTART = 0）場合、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログADCに自動的にインジェクトされます。この動作は透過的であり、変換開始のサイクル遅延は増加しません。V_{REF+} が 10% を超えて変化した場合、ADCオフセットを再較正することを推奨します。

オフセットおよび直線性較正の時間の要件については、デバイスのデータシートを参照してください。

ソフトウェアでのADCの較正手順

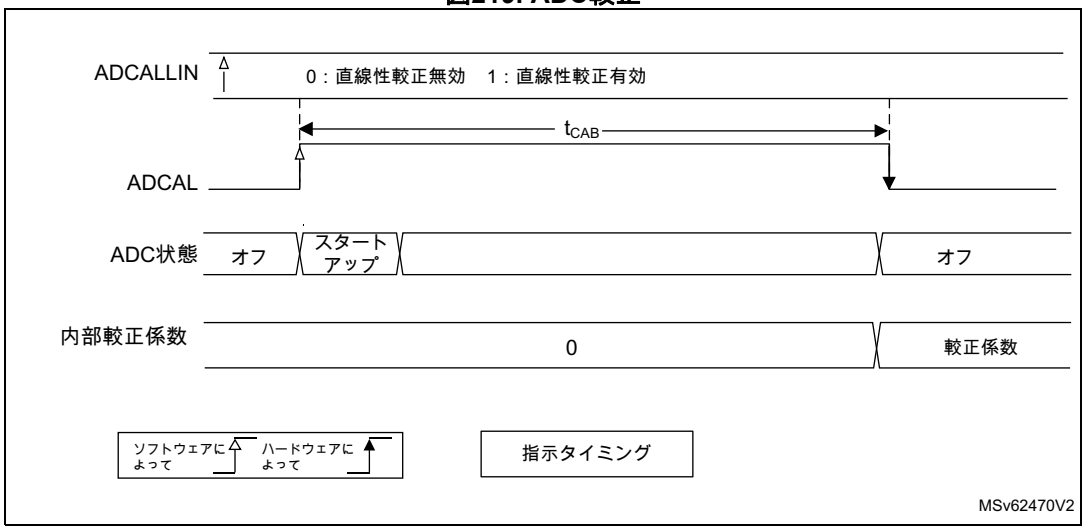
1. DEEPPWD = 0、ADVREGEN = 1であることを確認し、ADC電圧レギュレータの起動時間が経過している（LDORDY = 1）ことを確かめます。
2. ADEN = 0であることを確認します。
3. 直線性較正を有効化（ADCALLIN = 1）、または無効化（ADCALLIN = 0）します。
4. ADC_CALFACTのCAPTURE_COEFおよびLATCH_COEFがクリアされていることを確認します。
5. Set in the ADC_CRレジスタのADCALをセットし、ADCAL = 0になるのを待ちます。

図219 に、ADC較正のタイミング図を示します。

注： **ソフトウェアで較正を起動できるのは、ADEN = 0でLDORDY = 1（ADCが無効化され、LDOがレディ状態）の場合のみです。**

シングルエンドチャネルと差動チャネルを同時に較正することはできません。オフセット較正は同じシーケンス内で実行する必要があります。

図219. ADC較正



較正係数の読出し手順

較正が完了（ハードウェアによりADCALビットがクリアされる）すると、ADC_CALFACT2レジスタを使用して較正係数を読み出すことができます。この操作を行うには、9つの読出しアクセスが必要です。

1. DEEPPWD = 0、ADVREGEN = 1であることを確認し、ADC電圧レギュレータの起動時間が経過している（LDORDY = 1）ことを確かめます。
2. ADEN = 1であることを確認します。
3. ADC_CALFACレジスタでCAPTURE_COEFをセットし、LATCH_COEFをクリアします。
4. ADC制御レジスタ（ADC_CR）のCALLINDEX[0:3] をセットして、較正係数を選択します。
5. ADC_CALFACT2レジスタから較正係数を読み出します。
6. 必要なそれぞれの較正係数に対して、手順 **4**と**5**を繰り返します。
7. ADC_CALFACTレジスタのCAPTURE_COEFビットをクリアします。

注： **ソフトウェアから較正係数にアクセスできるのは、ADEN = 1で、ADSTART = 0かつJADSTART = 0である（ADCが有効化され、変換の実行中でない）ときだけです。**

ソフトウェアでのADCへの較正係数の再インジェクト手順

1. ADEN = 1、ADSTART = 0、かつJADSTART = 0である（ADCが有効化され、変換の実行中でない）ことを確認します。
2. ADC_CALFACTレジスタのCAPTURE_COEFビットとLATCH_COEFビットをクリアします。
3. CALINDEX[3:0] に更新する対象の較正インデックスをセットし、ADC_CALFACT2レジスタのCALFACT[31:0] に較正係数を書き込みます。
4. 必要な較正インデックスに対して、手順3を繰り返します。
5. ADC_CALFACTレジスタのLATCH_COEFビットをセットします。
6. ADC_CALFACTレジスタのLATCH_COEFビットをクリアします。

注： **ソフトウェアで較正係数を更新できるのは、ADEN = 1で、ADSTART = 0かつJADSTART = 0である（ADCが有効化され、変換の実行中でない）ときだけです。**

較正係数のインデックス

較正係数は、インデックスを使用してアナログブロックに格納されます。インデックス0b0000および0b1000にはオフセット較正係数が格納され、インデックス0b0001 ~ 0b0110には直線性係数が格納されます。インデックス0b0111の下位2バイトには、直線性較正係数が格納されます。内部オフセット較正係数は、インデックス0b0111の第2バイトにプログラムする必要があります。ただし、読出しはインデックス0b1000の第3バイトで行われます。較正係数のプログラミングまたは再インジェクションを行う場合、読出し操作と書込み操作には正しいインデックスを使用するようにしてください。

概要は表305 を参照してください。

表305.較正係数のインデックス

CALINDEX[3:0] の値	較正係数			
	バイト位置			
	バイト3	バイト2	バイト1	バイト0
0b0000	差動オフセット		シングルエンドオフセット	
0b0001	直線性係数1			
0b0010	直線性係数2			
0b0011	直線性係数3			
0b0100	直線性係数4			
0b0101	直線性係数5			
0b0110	直線性係数6			
0b0111（読出し）	予約済み	予約済み	直線性係数7	
0b0111（書込み）	予約済み	内部オフセット		
0b1000（読出し）	内部オフセット	予約済み	予約済み	予約済み
0b1001	予約済み	較正モード	予約済み	予約済み
0b1000（書込み）	予約済み	予約済み	予約済み	予約済み
その他の値	予約済み			

ソフトウェアによって直線性較正係数がプログラムされている状態でオフセット較正を実行するには、まず直線性較正係数をプログラムしてから、オフセット較正を実行します。先にオフセット較正を実行すると、LATCH_COEFビットをプログラムしたときに、オフセット較正係数が上書きされます。

オフセット較正係数をインデックス0b0111のバイト2にプログラムするときにバイト1およびバイト0（直線性係数7）が上書きされるため、オフセット較正を単独でプログラムすることはできません。これを避けるため、直線性較正係数を読み出しておき、読み出した値でインデックス0b0111をプログラムしてください。

拡張較正モード

ADCの性能を向上させるため、一部の製品バージョンには拡張較正モードが実装されています（セクション 33.3：A/Dコンバータの実装を参照）。

以下に、拡張較正モードを有効化する手順を示します。

1. DEEPPWD = 0、ADVREGEN = 1であることを確認し、ADC電圧レギュレータの起動時間が経過している（LDORDY = 1）ことを確かめます。
2. ADC_CRレジスタのADCCALLINをセットします。
3. ADC_CALFACTのCAPTURE_COEFおよびLATCH_COEFがクリアされていることを確認します。
4. ADENをセットし、ADRDYがセットされるまで待ちます。
5. CALINDEX[3:0] = 0b1001をセットし、ADC_CALFACT2レジスタのCALFACT[31:0] に0x0302 1100を書き込みます。
6. ADC_CALFACTレジスタのLATCH_COEFビットをセットします。
7. ADDISをセットし、ADENビットがクリアされるまで待ちます。
8. ADC_CRレジスタのADCALをセットし、ADCALビットがクリアされるまで待ちます。

注： **較正が完了すると、CALINDEX[3:0] = 0b1001のCALFACT[31:0] の値がリセットされます。**

33.4.9 ADCオン / オフ制御（ADEN、ADDIS、ADRDY）

最初に、セクション 33.4.6：ADCディープ･パワー･ダウン･モード（DEEPPWD）およびADC電圧レギュレータ（ADVREGEN）で説明している手順に従ってください。

DEEPPWDビットがクリアされ、ADVREGENビットがセットされると、ADCを有効化できます。正確な変換を開始するためには、その前に安定化時間t_{STAB}が必要です（図220 を参照）。ADCは、次の2つの制御ビットによって有効化または無効化されます。

- ADEN = 1の場合：ADCは有効です。ADCの動作準備ができると、ADRDYがセットされます。
- ADDIS = 1の場合：ADCは無効です。

アナログADCが実質的に無効になると、ADENおよびADDISビットがハードウェアによって自動的にクリアされます。

その後、レギュラ変換を開始できます。ADSTARTをセットすることによって（セクション 33.4.19：外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN[1:0]、JEXTSEL、JEXTEN[1:0]）を参照）、またはトリガが有効化された状態で外部トリガイベントが発生したときに、開始します。

インジェクト変換は、JADSTARTをセットすることによって、またはインジェクトトリガが有効化された状態で外部インジェクトトリガイベントが発生したときに、開始されます。

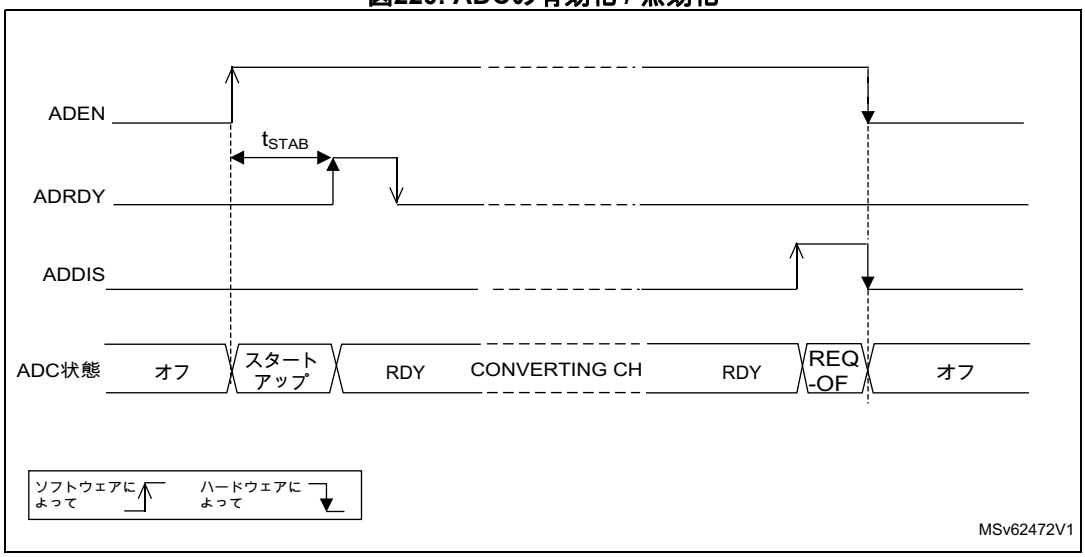
ソフトウェアでのADCの有効化手順

1. ADC_ISRレジスタのADRDYビットに「1」を書き込んでクリアします。
2. ADEN = 1にセットします。
3. ADRDY = 1になるまで待ちます（ADRDYはADC起動時間後にセットされます）。これは、対応する割込みを使用することで実行できます（ADRDYIE=1）。
4. ADC_ISRレジスタのADRDYビットに「1」を書き込んでクリアします（オプション）。

ソフトウェアでのADCの無効化手順

1. ADSTART = 0かつJADSTART = 0であることを確認し、変換が実行中でないことを確認します。必要な場合は、ADSTP = 1およびJADSTP= 1に設定して、実行中のレギュラ変換とインジェクト変換を停止したうえで、ADSTP = 0およびJADSTP = 0になるまで待ちます。
2. ADDIS = 1に設定します。
3. アプリケーションによって必要とされる場合、ADEN = 0になり、アナログADCが実質的に無効になるまで待ちます（ADEN = 0になると、ADDISは自動的にリセットされます）。

図220. ADCの有効化 / 無効化



33.4.10 ADC制御ビット書込み時の制約

ソフトウェアにより、ADCクロックの設定と有効化のためにRCC制御ビットをプログラムし（リセットとクロック制御のセクションを参照）、ADC_DIFSELレジスタの制御DIFSELビット、ADC12_CCRレジスタ、ADC_CRレジスタのADCALおよびADEN制御ビットをプログラムすることができます。ただし、そのためにはADCが無効化されている必要があります。

ADENがクリアされている場合、較正が起動されるようにソフトウェアによりADCALビットをプログラムすることができます。ADENがセットされていて、変換が実行中でなければ（ADSTARTとJADSTARTが両方ともクリアされていれば）、較正係数の読出しや更新ができます。

ADCが有効であり、ADCを無効にするための保留中のリクエストがない場合のみ（ADEN = 1かつADDIS = 0である必要があります）、ソフトウェアはADC_CRレジスタのADSTART、JADSTART、およびADDIS制御ビットを書き込むことができます。

次の制約は、ADC_CFGRx、ADC_SMPRy、ADC_LTRy、ADC_HTRy、ADC_SQRy、ADC_OFRyおよびADC_IERレジスタのそれ以外のすべての制御ビットに適用されます。

- レギュラ変換の設定に関連する制御ビット：ADCが有効（ADEN = 1）であり、実行中のレギュラ変換がない（ADSTART = 0）場合のみ、ソフトウェアで書き込むことができます。
- インジェクト変換の設定に関連する制御ビット：ADCが有効（ADEN = 1）であり、実行中のインジェクト変換がない（JADSTART = 0）場合のみ、ソフトウェアで書き込むことができます。
- アナログデジタル変換の実行中にADC_LTRy、ADC_HTRy レジスタの内容を変更できます（詳細については、セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照）。

ADCが有効であり、変換が実行中で、ADCを無効にするための保留中のリクエストがない場合のみ（ADSTART = 1またはJADSTART = 1、かつADDIS = 0）、ソフトウェアでADC_CRレジスタのADSTP制御ビットまたはJADSTP制御ビットを書き込むことができます。

注： **これらの禁止された書込みアクセスを防ぐハードウェア保護はなく、ADCの挙動は不明な状態になる場合があります。この状況から復帰するには、ADCを無効にする必要があります（ADENとADC_CRレジスタのすべてのビットをクリアします）。**

33.4.11 チャネルの選択（SQRx、JSQRx）

ADCごとに最大20の多重化チャネルがあります。

- GPIOパッドからの最大 17 のアナログ入力
- 各 ADCは次の3 つの内部アナログ入力に接続
  - 内部温度センサ（V_{SENSE}）
  - 内部基準電圧（V_{REFINT}）
  - V_{BAT} 監視チャネル（$V_{BAT}/4$）

上記の内部アナログ入力の、外部ADCピンまたは内部信号への接続については、セクション 33.4.2 : A/Dコンバータのピンおよび内部信号の ADC相互接続の表を参照してください。

変換は、レギュラとインジェクトの2つのグループに構成することができます。各グループは、任意のチャネルに対して任意の順序で行うことができる一連の変換で構成されます。たとえば、変換シーケンスの実装の順序を、ADC_IN3、ADC_IN8、ADC_IN2、ADC_IN2、ADC_IN0、ADC_IN2、ADC_IN2、ADC_IN15という順にすることができます。

- **レギュラグループ**は、最大 16 の変換で構成されます。レギュラチャネルと変換シーケンス内での順序は、ADC_SQRyレジスタで選択する必要があります。レギュラグループの合計変換数は、ADC_SQR1レジスタのL[3:0] ビットに書き込む必要があります。
- **インジェクトグループ**は、最大4つの変換で構成されます。インジェクトチャネルと変換シーケンス内での順序は、ADC_JSQRレジスタで選択する必要があります。インジェクトグループの合計変換数は、ADC_JSQRレジスタのL[1:0] ビットに書き込む必要があります。

レギュラ変換の実行中は、ADC_SQRyレジスタを変更しないでください。ADC_SQRyレジスタを変更するには、ADSTPをセットして、まずADCレギュレータ変換を手停止する必要があります（セクション 33.4.18 : 実行中の変換の停止（ADSTP、JADSTP）を参照）。

注： **内部アナログチャネルのうちの 1 つを変換するには、最初に ADC12_CCR レジスタの VBATEN、VSENSESEL、またはVREFENビットをプログラムして、対応するアナログソースを有効化する必要があります。**

33.4.12 チャネル事前選択レジスタ（ADC_PCSEL）

SQRxまたはJSQRxビットで選択された各チャネルについて、対応するADC_PCSELビットを事前に設定する必要があります。

このADC_PCSELビットは、I/Oレベルに組み込まれているアナログスイッチを制御します。ADC入力マルチプレクサは、SQRxおよびJSQRxの設定に従ってADC入力を超高速で選択しますが、I/Oに組み込まれているアナログスイッチは、ADCマルチプレクサほどの高速では反応できません。I/Oでのアナログ切替え制御による遅延を回避するには、SQRxおよびJSQRxで選択される入力チャネルを事前選択する必要があります。この選択は、各ADC V_{INP}[i] に基づいています。たとえば、ADCでADC_IN1を変換する場合、ADC_PCSELでPCSEL1bitもセットする必要があります。

33.4.13 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）

変換を開始する前に、ADCは測定する電圧ソースとADCの内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースが内蔵コンデンサを入力電圧レベルまでチャージするのに十分な長さが必要です。

各チャネルは、ADC_SMPR1およびADC_SMPR2レジスタのSMP[2:0] ビットをプログラムすることでな異なるサンプリング時間でサンプリングできます。したがって、次のサンプリング時間の値から選択することができます。

- SMP = 000：5 ADCクロックサイクル
- SMP = 001：6 ADCクロックサイクル
- SMP = 010：12 ADCクロックサイクル
- SMP = 011：20 ADCクロックサイクル
- SMP = 100：36 ADCクロックサイクル
- SMP = 101：68 ADCクロックサイクル
- SMP = 110：391 ADCクロックサイクル
- SMP = 111：814 ADCクロックサイクル

合計変換時間は、次のように計算されます。

$$T_{CONV} = \text{サンプリング時間} + \text{Conversion 時間}$$

例

1つのデータを変換する場合、サンプリング時間は5サイクルとなり、変換時間は14bitモードでは17サイクルとなります。$F_{adc_ker_ck}$ が55MHzの場合、次のようになります。

T_{CONV} = (5 + 17) ADCクロックサイクル = 22 ADCクロックサイクル = 0.40 µs

上記の結果は、R_{AIN} << 1KΩ を前提としています（外部抵抗に基づいて追加すべきサンプリング時間については、データシートを参照してください）。

ADCは、ステータスビットEOSMPをセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。

I/Oアナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/Oアナログスイッチの抵抗が増加します。サンプリング時間は、状況に応じて調整する必要があります（対応する電気的特性については、デバイスデータシートを参照）。この抵抗は、内部電圧ブースタを有効化することにより、低いV_{DDA} 電圧で最小化することができます（詳細は、SYSCFGセクションを参照）。

バルブサンプリングモード

ADC_CFGR2レジスタでBULBビットがセットされているとき、サンプリング期間は最後のADC変換の直後に開始します。ハードウェアまたはソフトウェアトリガは、サンプリング時間がADC_SMPR1レジスタでプログラムされた後、変換を開始します。最初のADC変換は、ADCが有効化された後、SMPビットでプログラムされたサンプリング時間で実行されます。バルブモードは、2番目の変換から開始するのに有効です。

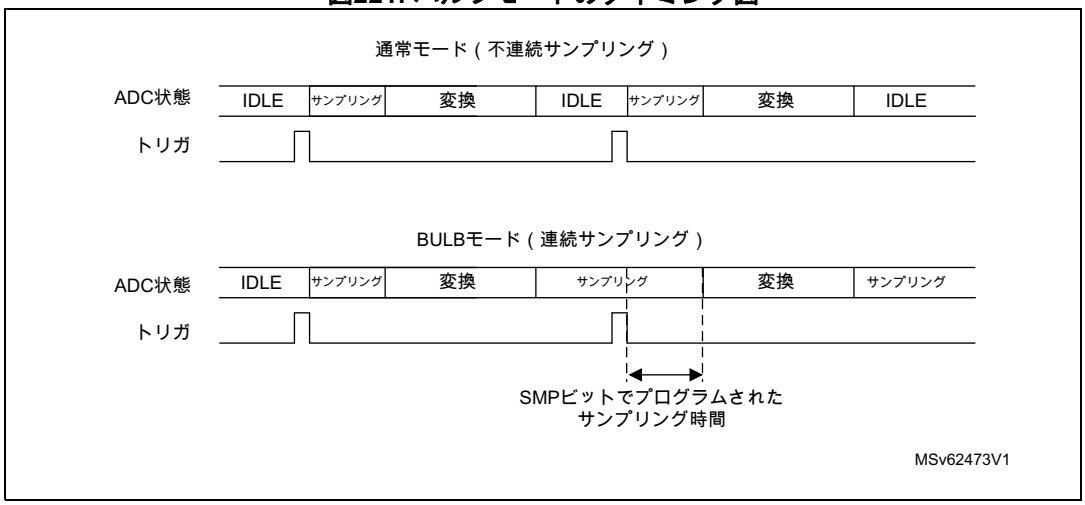
最大サンプリング時間は制限されます（データシートのADC特性のセクションを参照）。

バブルモードは、連続変換モードで、またはインジェクトチャネル変換と一緒に使用できないことがあります。

BULBビットがセットされているとき、ADC_CFGR2のSMPTRIGビットをセットすることはできません。

ADSTPをセットすることによりバルブモードでの変換を停止する場合、またはDMA転送の完了時、ADDISビットをセットしてADCを無効化する必要があります。

図221. バルブモードのタイミング図



サンプリング時間制御トリガモード

SMPTRIGビットがセットされているとき、SMPxビットによってプログラムされたサンプリング時間は適用されません。サンプリング時間は、トリガ信号エッジによって制御されます。

ハードウェアトリガが選択されたとき、トリガ信号の各立ち上がりエッジによってサンプリング期間は開始します。立ち下がりエッジによってサンプリング期間は終了し、変換が開始されます。

ソフトウェアトリガが選択されたとき、ソフトウェアトリガはADC_CRのADSTARTビットではなく、SWTRIGビットです。サンプリング期間を開始するには、SWTRIGビットをセットする必要があり、サンプリング期間を終了して変換を開始するには、SWTRIGビットをクリアする必要があります。

最大サンプリング時間は制限されます（データシートのADC特性のセクションを参照）。

このモードは、連続変換モードともインジェクトチャネル変換とも互換性がありません。

SMPTRIGビットがセットされているとき、BULBビットはセットできません。

33.4.14 シングル変換モード（CONT = 0）

シングル変換モードでは、ADCはすべてのチャネルをの変換を一度行います。CONTビットが0の時、次のいずれかによってこのモードが開始します。

- ADC_CRレジスタのADSTARTビットをセットします（レギュラチャネル、ソフトウェアトリガを選択）。
- ADC_CR レジスタの JADSTART ビットをセットします（インジェクトチャネル、ソフトウェアトリガを選択）。
- 外部ハードウェアトリガイベント（レギュラまたはインジェクトチャネル）

  外部イベントをトリガする前にADSTARTビットまたはJADSTARTビットをセットする必要があります。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは32bitのADC_DRレジスタに格納されます。
- EOC （レギュラ変換の完了）フラグがセットされます。
- EOCIEビットがセットされている場合、割込みが生成されます。

インジェクトシーケンス内で、各変換の完了後、

- 変換データは4つの32bit ADC_JDRyレジスタのうちの1つに格納されます。
- JEOC （インジェクト変換の完了）フラグがセットされます。
- JEOCIEビットがセットされている場合、割込みが生成されます。

レギュラシーケンスの完了後、

- EOS （レギュラシーケンスの完了）フラグがセットされます。
- EOSIEビットがセットされている場合、割込みが生成されます。

インジェクトシーケンスの完了後、

- JEOS （インジェクトシーケンスの完了）フラグがセットされます。
- JEOSIEビットがセットされている場合、割込みが生成されます。

次に、ADCは、新しい外部レギュラトリガまたは外部インジェクトトリガが発生するか、ADSTARTまたはJADSTARTビットが再びセットされるまで停止します。

注： **単一チャネルを変換するには、長さが1のシーケンスをプログラムします。**

33.4.15 連続変換モード（CONT = 1）

このモードはレギュラチャネルにのみ適用されます。

連続変換モードでは、ソフトウェアまたはハードウェアのレギュラトリガイベントが発生すると、ADCはチャネルのレギュラ変換を一度すべて実行し、自動的に再起動して、シーケンスの各変換を連続的に実行します。CONTビットが1の時、外部トリガまたはADC_CRレジスタのADSTARTビットをセットすることによって、このモードが開始されます。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは32bitのADC_DRレジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIEビットがセットされている場合、割込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIEビットがセットされている場合、割込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADCは変換シーケンスを連続的に繰り返します。

注： **単一チャネルを変換するには、長さが1のシーケンスをプログラムします。**

不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方を設定することは禁じられています。

インジェクトチャネルを連続して変換することはできません。唯一の例外は、連続モードに設定されたレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTOビットを使用）場合です。セクション ：自動インジェクションモードを参照してください。

33.4.16 変換の開始（ADSTART、JADSTART）

ソフトウェアによりADCレギュラ変換を開始するには、ADTARTをセットします。ADSTARTがセットされると、次のように変換が開始します。

- EXTEN[1:0] = 0x0（ソフトウェアトリガ）の場合、すぐに開始、または
- EXTEN[1:0] ≠ 0x0の場合、選択されたレギュラハードウェアトリガの次のアクティブエッジで開始します。

ソフトウェアによりADCインジェクト変換を開始するには、JADSTARTをセットします。JADSTARTがセットされると、次のように変換が開始します。

- JEXTEN[1:0] = 0x0（ソフトウェアトリガ）の場合、すぐに開始、または
- JEXTEN[1:0] ≠ 0x0の場合、選択されたインジェクトハードウェアトリガの次のアクティブエッジで開始します。

注： **自動インジェクションモード（JAUTO = 1）では、ADSTARTビットを使用してレギュラ変換を開始し、つづいて自動インジェクト変換を実行します（JADSTARTはクリア状態に維持する必要があります）。**

ADSTARTおよびJADSTARTでは、ADCの動作が実行中かどうかを示す情報も提供しています。ADCは、ADSTARTとJADSTARTの両方がクリアされた状態（ADCがアイドル）で再設定できます。

次のときに、ADSTARTはハードウェアによってクリアされます。

- ソフトウェアトリガによるシングルモードのとき（CONT = 0、EXTEN[1:0] = 0x0）：変換シーケンスの終了時（EOS = 1）
- ソフトウェアトリガによる不連続モードのとき（CONT = 0、DISCEN = 1、EXTEN[1:0] = 0x0）：変換の終了時（EOC = 1）
- それ以外の場合（CONT = x、EXTEN[1:0] = x）：ソフトウェアによるADSTPアサーション手順の実行後

注： **連続モード（CONT = 1）では、シーケンスは自動的に再起動されるので、EOSがアサートされても、ADSTARTはハードウェアによってクリアされません。**

シングルモードでハードウェアトリガが選択されたとき（CONT = 1かつEXTEN[1:0] ≠ 0x0）、次のハードウェアトリガイベントでADSTARTの再リセットを必要としないソフトウェアを支援するために、EOSがアサートされても、ADSTARTはハードウェアによってクリアされません。これにより、さらなるハードウェアトリガの見落としを防ぐことができます。

次のときに、JADSTARTはハードウェアによってクリアされます。

- ソフトウェアインジェクトトリガによるシングルモードのとき（JEXTEN[1:0] = 0x0）：インジェクト変換シーケンスの終了時（JEOSのアサート）、またはJDISCEN = 1の場合はサブグループ処理の終了時
- それ以外のとき（JEXTEN[1:0]=x）：ソフトウェアによりJADSTPアサーション手順の実行後

注： **ソフトウェアトリガが選択されているとき、EOCフラグがまだハイの場合にADSTARTビットをセットしてはいけません。**

33.4.17 タイミング

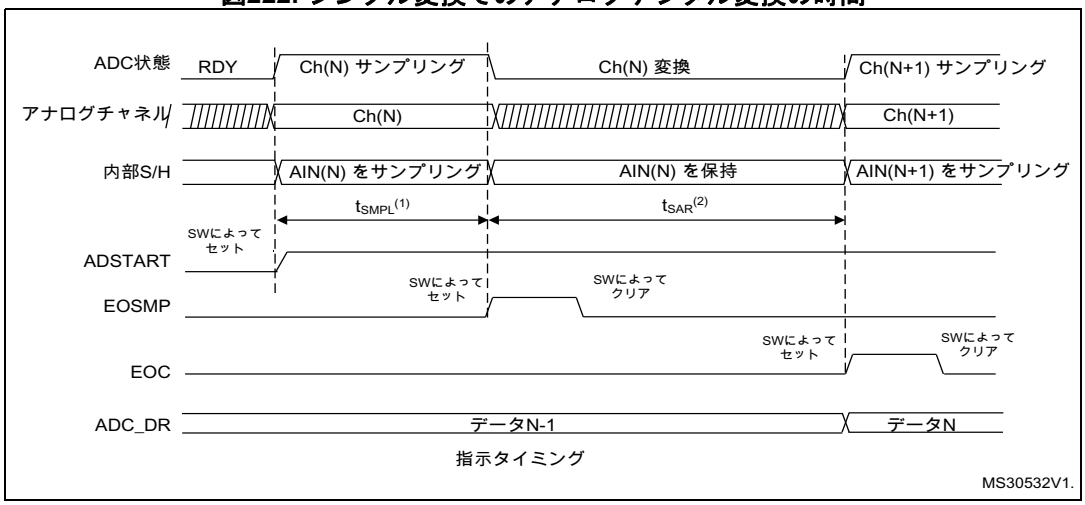
変換の開始から変換の終了までの経過時間は、シングル変換の場合は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計で、連続変換の場合は、設定されたさンプリング時間から、サンプリングと前のSARの間の重複した時間を差し引いた時間です。

単一変換モードでは次のようになります。

$$T_{CONV} = T_{SMPL} + T_{SAR} = [5_{min} + 17_{14bits}] \times T_{adc_ker_ck}$$

$$T_{CONV} = T_{SMPL} + T_{SAR} = 22 \times T_{adc_ker_ck\ min\ for\ 14bits} = 400.0\ ns\ (for\ F_{adc_ker_ck} = 55\ MHz)$$

図222. シングル変換でのアナログデジタル変換の時間



1. T_{SMPL} はSMP[2:0] に依存します。
2. T_{SAR} はRES[1:0] に依存します。

33.4.18 実行中の変換の停止（ADSTP、JADSTP）

ソフトウェアは、ADSTPをセットすることによって実行中のレギュラ変換を、またJADSTPをセットすることによって実行中のインジェクト変換を停止することができます。

変換を停止すると、実行中のADC動作がリセットされます。その後、（たとえば、チャネルの選択や鳥がを変更して）ADCを再設定できます。再設定すると、新しい動作が可能な状態になります。

レギュラ変換がまだ実行中のときにインジェクト変換を停止したり、インジェクト変換がまだ実行中のときにレギュラ変換を停止したりすることができます。これにより、たとえば、レギュラ変換がまだ実行中のときにインジェクト変換シーケンスやトリガを再設定することができます（その逆も可能）。

ADSTPビットがソフトウェアによってセットされると、実行中のレギュラ変換はアボートされ、部分的な結果は破棄されます（ADC_DRレジスタは現在の変換では更新されません）。

JADSTPビットがソフトウェアによってセットされると、実行中のインジェクト変換はアボートされ、部分的な結果は破棄されます（ADC_JDRyレジスタは現在の変換では更新されません）。スキャンシーケンスも中止され、リセットされます（ADCを再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTP/ADSTARTビット（レギュラ変換の場合）、またはJADSTP/JADSTARTビット（インジェクト変換の場合）がハードウェアによってクリアされます。ソフトウェアでADCが

完全に停止していることを確認するには、ADSTART（またはJADSTART）ビットがリセットされるまでこれらのビットに対してポーリングする必要があります。

注： **自動インジェクション･モード（JAUTO = 1）では、ADSTPビットをセットすると、レギュラ変換とインジェクト変換の両方が中止されます（JADSTPを使用しないでください）。**

図223. 実行中のレギュラ変換の停止

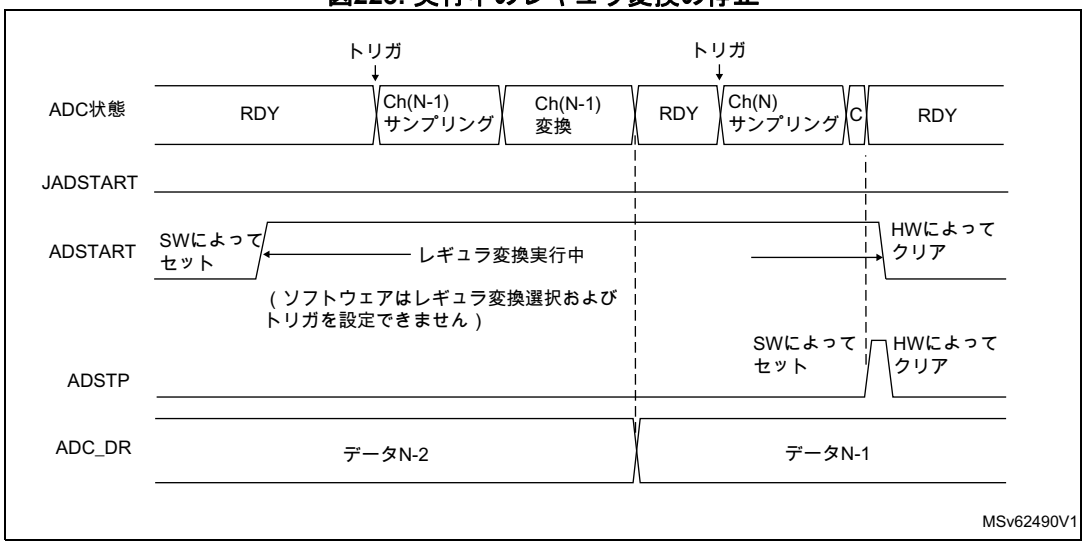
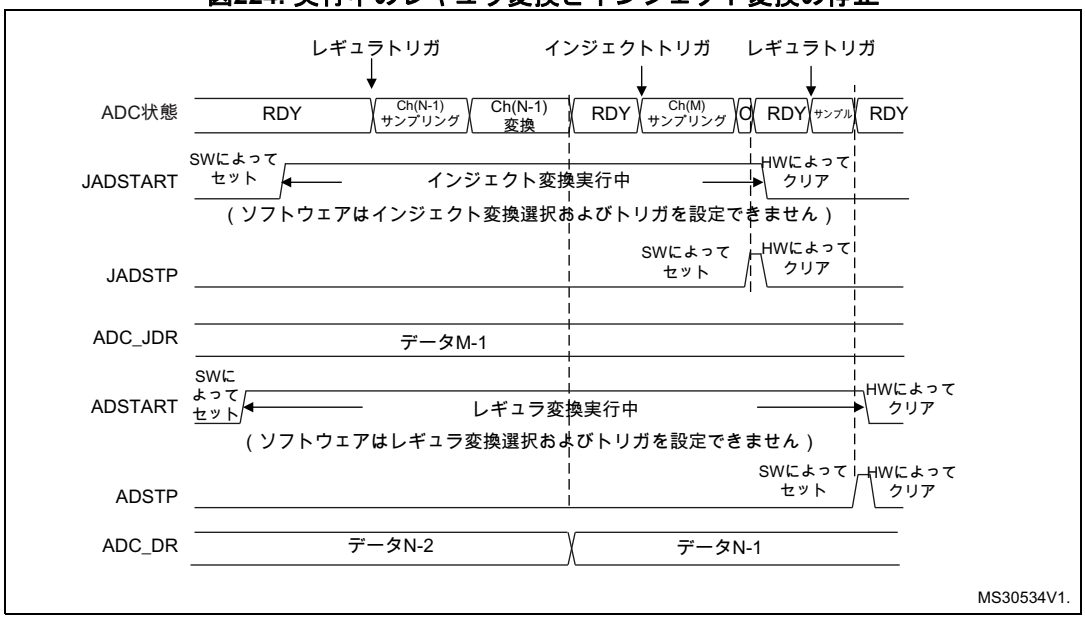


図224. 実行中のレギュラ変換とインジェクト変換の停止



33.4.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN[1:0]、JEXTSEL、JEXTEN[1:0]）

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャや入力ピンなど）によってトリガできます。EXTEN[1:0] 制御ビット（レギュラ変換の場合）またはJEXTEN[1:0] ビット（インジェクト変換の場合）が00以外の場合、外部イベントは選択した極性で変換をトリガできます。

レギュラトリガ選択は、一度ソフトウェアでビットADSTART = 1をセットすると有効になります。インジェクトトリガ選択は、一度ソフトウェアでビットJADSTART = 1をセットすると有効になります。

変換の実行中に発生したハードウェアトリガは無視されます。

- ADSTART = 0の場合、レギュラハードウェアトリガは無視されます。
- JADSTART = 0の場合、インジェクトハードウェアトリガは無視されます。

表306 にEXTEN[1:0] とJEXTEN[1:0] の値とトリガ極性の対応を示します。

表306.レギュラ外部トリガのトリガ極性の設定

EXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジでハードウェアトリガを検出します。
10	立ち下がりエッジでハードウェアトリガを検出します。
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： **レギュラトリガの極性は動作中に変更することはできません。**

表307.インジェクト外部トリガのトリガ極性の設定

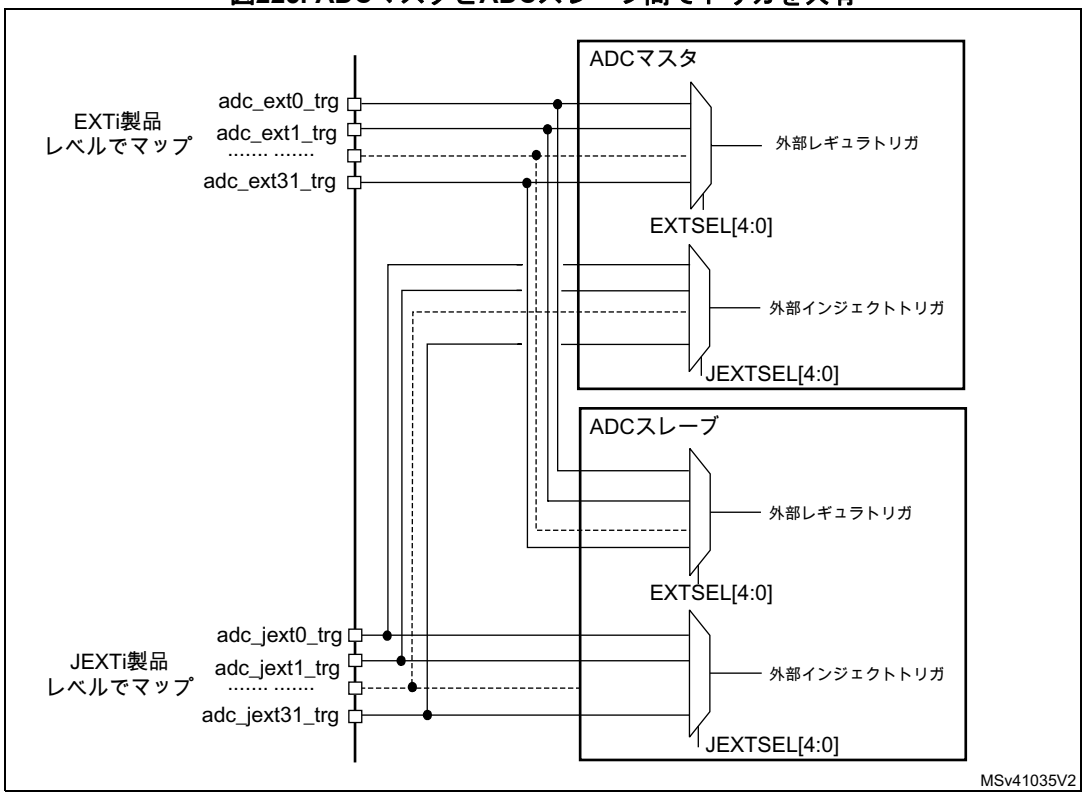
JEXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジでハードウェアトリガを検出します。
10	立ち下がりエッジでハードウェアトリガを検出します。
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

EXTSEL[4:0] およびJEXTSEL[4:0] 制御ビットでは、21の候補のうち、どのイベントでレギュラグループ変換とインジェクトグループ変換をトリガするかを選択します。

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： **トリガの選択は動作中に変更することはできません。**

図225. ADCマスタとADCスレーブ間でトリガを共有



上記の内部アナログ入力から外部ADCピンまたは内部信号への接続については、セクション 33.4.2 : A/Dコンバータのピンおよび内部信号の表「レギュラチャネル用のADC外部トリガ」と表「インジェクトチャネル用のADC外部トリガ」を参照してください。

33.4.20 インジェクトチャネルの管理

トリガインジェクションモード

トリガインジェクションを使用するには、ADC_CFGR1レジスタのJAUTOビットがクリアされていなければなりません。

1. 外部トリガによって、または、ADC_CRレジスタのADSTARTビットをセットすることによって、レギュラチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合、またはADC_CRレジスタのJADSTARTビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンススイッチが起動されます（すべてのインジェクトチャネルが一度変換されます）。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。
4. インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。図226 に対応するタイミング図を示します。

注： **トリガインジェクションを使用する場合は、トリガイベントの間隔がインジェクションシーケンスより長くなるようにしなければなりません。たとえば、シーケンスの長さが44 ADCクロックサイクルの場合（最小サンプリング時間を持つ2つの変換）、トリガの最小間隔は45 ADCクロックサイクルでなければなりません。**

自動インジェクションモード

ADC_CFGR1レジスタのJAUTOビットがセットされている場合、レギュラグループチャネルの変換後、インジェクトグループチャネルが自動的に変換されます。これを使用して、ADC_SQRyレジスタおよびADC_JSQRレジスタにプログラミングされた最大20回の変換シーケンスを実行することができます。

このモードでは、レギュラ変換を開始し、続いてインジェクト変換を開始するためにADC_CRレジスタのADSTARTビットをセットする必要があります（JADSTARTはクリアされたままでなければなりません）。ADSTPビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTPビットを使用することはできません）。

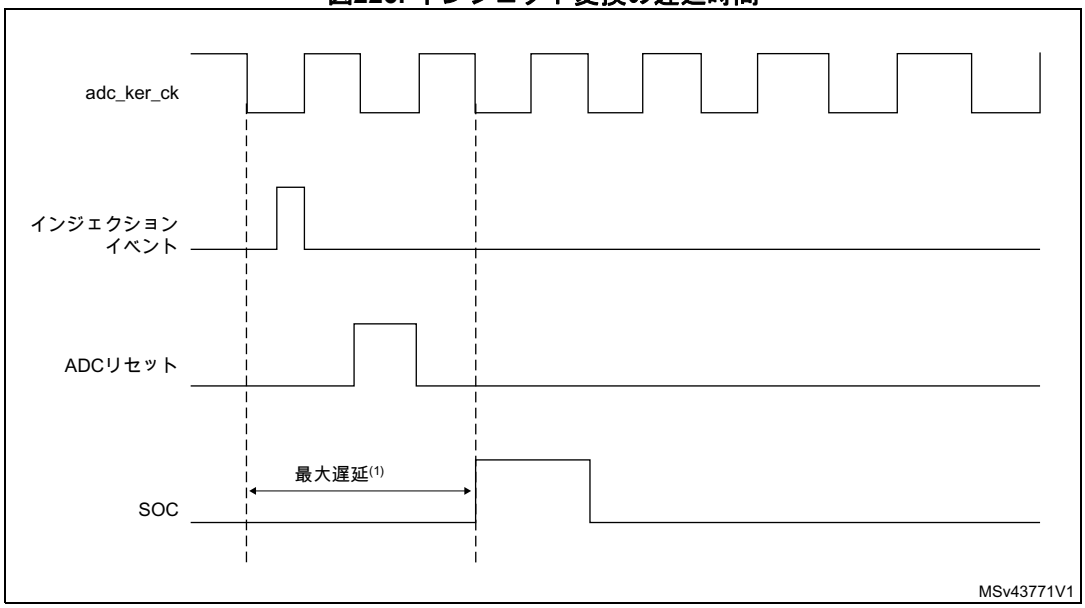
このモードでは、インジェクトチャネルの外部トリガは無効でなければなりません。

JAUTOビットに加えてCONTビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： **自動インジェクトモードと不連続モードを同時に使用することはできません。**

JAUTOモードで、レギュラシーケンスのデータをエクスポートする際にDMAを使用する場合は、サーキュラモードでプログラムする必要があります（DMA_CCRxレジスタのCIRCビットをセット）。CIRCビットがリセットされると（シングルショットモード）、DMA転送完了イベントによってJAUTOシーケンスが停止されます。

図226. インジェクト変換の遅延時間



1. 最大遅延時間の値は、デバイスのデータシートの電気特性に記載されています。

33.4.21 不連続モード（DISCEN、DISCNUM、JDISCEN）

レギュラグループモード

このモードを有効にするには、ADC_CFGR1レジスタのDISCENビットをセットします。

これを使用して、ADC_SQRyレジスタで選択された変換シーケンスの一部である短いn回（n ≤8）の変換シーケンス（サブグループ）を変換できます。nの値は、ADC_CFGR1レジスタのDISCNUM[2:0]ビットに書き込むことによって指定します。

外部トリガが発生すると、ADC_SQRxレジスタで選択された次のn個の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_SQR1レジスタのL[3:0]ビットによって定義されます。

例

- DISCEN=1、n=3、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1番目のトリガ : 変換されるチャネルは1、2、3です（各変換でEOCイベントが生成されます）。
  - 2番目のトリガ : 変換されるチャネルは6、7、8です（各変換でEOCイベントが生成されます）。
  - 3番目のトリガ : 変換されるチャネルは9、10、11です（各変換でEOCイベントが生成されます）。EOSイベントはチャネル11の変換後に生成されます。
  - 4番目のトリガ : 変換されるチャネルは1、2、3です（各変換でEOCイベントが生成されます）。
  - ...
- DISCEN = 0、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1番目のトリガ : シーケンス全体、すなわち、チャネル1、次いで2、3、6、7、8、9、10、および11が変換されます。各変換後にEOCイベントが生成され、最後の変換後にはEOSイベントも生成されます。
  - 後続のすべてのトリガイベントによって、シーケンス全体が再起動されます。

注 : **レギュラグループが不連続モードで変換されるときには、ロールオーバーは発生しません（シーケンスの最後のサブグループの変換はn回以下です）。**

すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。上の例では、4番目のトリガで最初のサブグループのチャネル1、2、および3が再変換されます。

不連続モードと連続モードの両方を有効にすることはできません。この場合（DISCEN = 1、CONT = 1の場合）、A/Dコンバータは連続モードが無効化されたかのように動作します。

インジェクトグループモード

このモードは、ADC_CFGR1 レジスタのJDISCEN ビットをセットすることで有効になります。ADC_JSQRレジスタで選択されたシーケンスを、外部インジェクトトリガイベント後に1チャネルずつ変換します。これは、「n」が1に固定されたレギュラチャネルの不連続モードに相当します。

外部トリガが発生すると、ADC_JSQRレジスタで選択された次のチャネル変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_JSQRレジスタのJL[1:0]ビットによって定義されます。

例

- JDISCEN = 1、変換されるチャネル = 1、2、3
  - 1番目のトリガ : チャネル1が変換されます（JEOCイベントが生成されます）。
  - 2番目のトリガ : チャネル2が変換されます（JEOCイベントが生成されます）。
  - 3番目のトリガ : チャネル3が変換され、JEOCイベントとJEOSイベントが生成されます。
  - ...

注： **すべてのインジェクトチャネルが変換されると、次のトリガでは、最初のインジェクトチャネルの変換が開始されます。上の例では、4番目のトリガで最初のインジェクトチャネル1が再変換されます。**

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTOのセット時には、ビットDISCENとJDISCENはクリアされたままでなければなりません。

33.4.22 プログラム可能な分解能（RES）- 高速変換モード

ADCの分解能を下げることによって、高速変換を行うことができます。

分解能は、制御ビットRES[1:0] をプログラムすることによって、14、12、10、または8bitに設定できます。図231、図232、図233、および図234 に、分解能とデータの整列に対する変換結果のフォーマットを示します（連続モードでは、入力の高い抵抗のための余分なサンプリングサイクルは前提としていません）。

分解能を下げることによって変換時間を高速にでき、高いデータ精度を必要としないアプリケーションに有効です。表308 のように、これによって逐次比較ステップの間に変換時間は短縮されます。

表308.T_{SAR} タイミングは分解能に依存

RES	T_{SAR}（ADCクロックサイクル）	T_{SAR}（ns）（$F_{adc_ker_ck}$ = 55 MHz時）	$T_{adc_ker_ck}$（ADCクロックサイクル）、サンプリング時間 =5 ADCクロックサイクル	$T_{adc_ker_ck}$（ns）（$F_{adc_ker_ck}$ = 55MHz時）
14	17 ADCクロックサイクル	309.1	22 ADCクロックサイクル	400.0
12	15 ADCクロックサイクル	272.7	20 ADCクロックサイクル	363.6
10	13 ADCクロックサイクル	236.4	18 ADCクロックサイクル	327.3
8	11 ADCクロックサイクル	200.0	16 ADCクロックサイクル	290.9

33.4.23 変換の終了とサンプリングフェーズの終了（EOC、JEOC、EOSMP）

ADCは各レギュラ変換の終了（EOC）イベントと各インジェクト変換の終了（JEOC）イベントをアプリケーションに通知します。

ADCは、新しいレギュラ変換データがADC_DRレジスタで使用可能になるとすぐにEOCフラグをセットします。EOCIEビットがセットされている場合は、割込みを生成することができます。EOCフラグは、ソフトウェアによって1を書き込むことによって、またはADC_DRを読み出すことによってクリアされます。

ADCは、新しいインジェクト変換データがADC_JDRyレジスタの1つで使用可能になるとすぐにJEOCフラグをセットします。JEOCIEビットがセットされている場合は、割込みを生成することができます。JEOCフラグは、ソフトウェアによって1を書き込むことによって、または対応するADC_JDRyレジスタを読み出すことによってクリアされます。

さらに、ADCは、ステータスビットEOSMPをセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。EOSMPフラグは、ソフトウェアによって1を書き込むことによってクリアされます。EOSMPIEビットがセットされている場合は、割込みを生成することができます。

33.4.24 変換シーケンスの終了（EOS、JEOS）

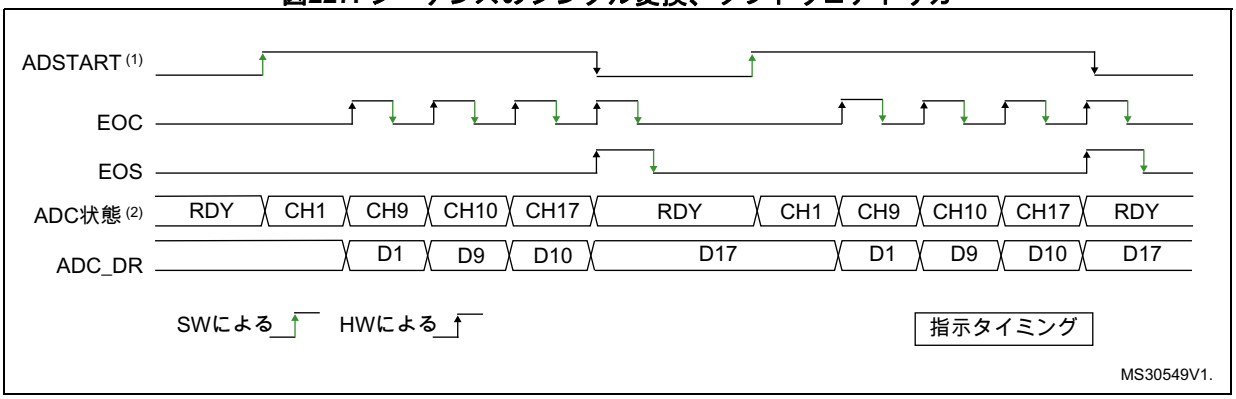
ADCは各レギュラシーケンスの終了（EOS）と各インジェクトシーケンスの終了（JEOS）イベントをアプリケーションに通知します。

ADCは、レギュラ変換シーケンスの最後のデータがADC_DRレジスタで使用可能になるとすぐに、EOSフラグをセットします。EOSIEビットがセットされている場合は、割込みを生成することができます。EOSフラグは、ソフトウェアによって1を書き込むことによってクリアされます。

ADCは、インジェクト変換シーケンスの最後のデータが完了するとすぐに、JEOSフラグをセットします。JEOSIEビットがセットされている場合は、割込みを生成することができます。JEOSフラグは、ソフトウェアによって1を書き込むことによってクリアされます。

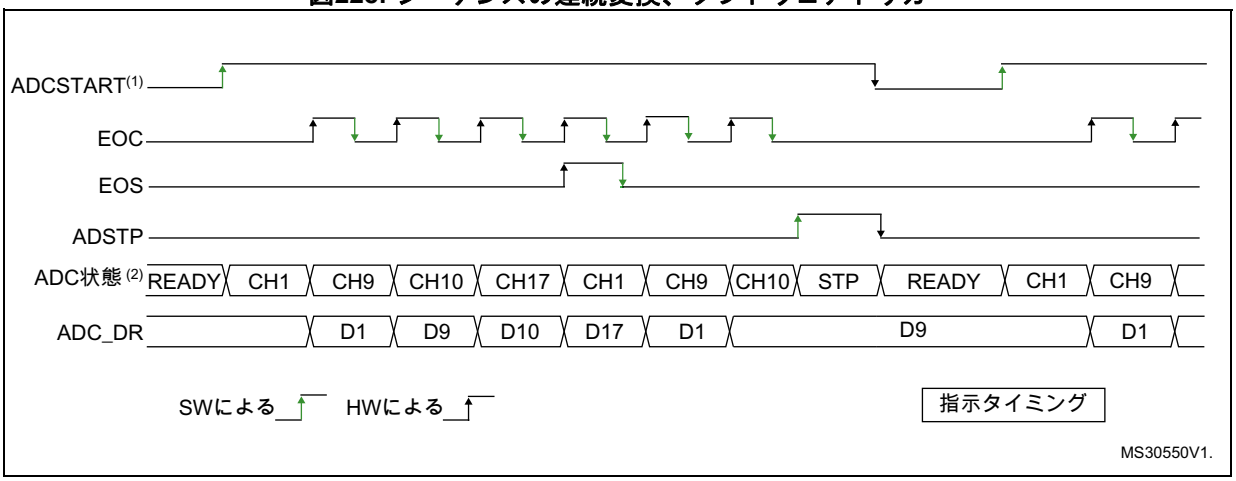
33.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

図227. シーケンスのシングル変換、ソフトウェアトリガ



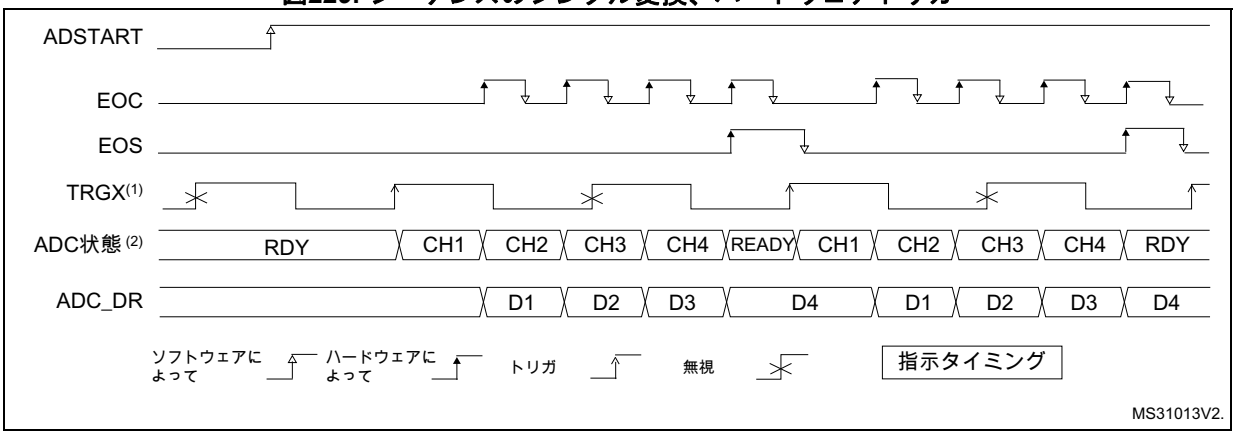
1. EXTEN[1:0] = 0x0、CONT = 0
2. 選択されたチャネル = 1、9、10、17；AUTDLY = 0。

図228. シーケンスの連続変換、ソフトウェアトリガ



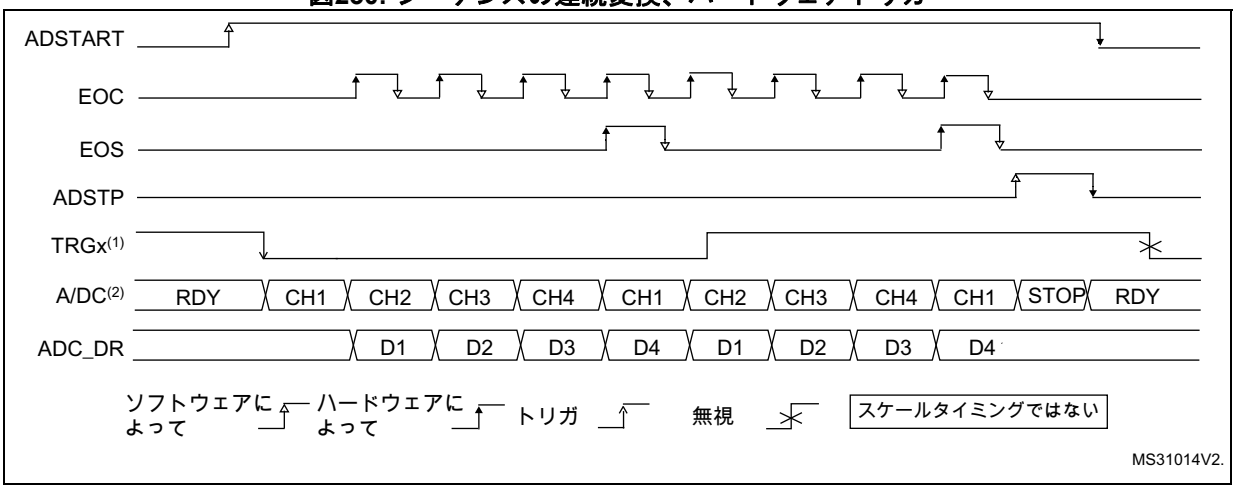
1. EXTEN[1:0] = 0x0、CONT = 1
2. 選択されたチャネル = 1、9、10、17；AUTDLY = 0。

図229. シーケンスのシングル変換、ハードウェアトリガ



1. TRGx（オーバー周波数）がトリガ･ソースとして選択されます（EXTEN[1:0] = 01、CONT = 0）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY = 0。

図230. シーケンスの連続変換、ハードウェアトリガ



1. TRGxがトリガ･ソースとして選択されます（EXTEN[1:0] = 10、CONT = 1）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY = 0。

33.4.26 低周波数トリガモード（LFTRIG）

ADCが有効になるかまたは最後のADC変換が完了したとき、ADCは次の変換を開始する準備が整います。ADCは定義した時間（t_{idle}）に開始する必要があります。そうしないと、トランジスタのリークにより変換データが破壊される可能性があります（t_{idle} の最大値についてはデータシートを参照してください）。

アプリケーションで、シングル変換モードでは1回のトリガと別のトリガの間で、またはADC有効化と最初の変換の間でT_{idle} の最大値より長い時間が必要な場合、ADCコンバータの入力マルチプレクサを再初期化する必要があります。この動作を選択するには、ADC_CFGR2のLFTRIGビットをセットします。

33.4.27 データ管理

データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、USAT、SSAT）

データおよび配置

各レギュラ変換チャネルの終了時（EOCイベントの発生時）、変換データの結果は32bit幅のADC_DRデータレジスタに格納されます。

各インジェクト変換チャネルの終了時（JEOCイベントの発生時）、変換データの結果は対応する32bit幅のADC_JDRyデータレジスタに格納されます。

ADC_CFGR2レジスタのOVSS[3:0] とLSHIFT[3:0] のビットフィールドは、変換後に格納されるデータの配置を選択します。デフォルトでは、データは右詰めになります。データの配置の例については、図231、図232、図233、および図234 を参照してください。

注： **データは、通常モードとオーバーサンプリングモードで再配置できます。**

オフセット

オフセットy（y = 1、2、3、4）は、ADC_OFRyレジスタのOFFSETy[23:0] ビットフィールドに0とは異なる値をプログラムすることでチャネルに適用されます。オフセットを適用するチャネルは、ADC_OFRyレジスタのOFFSETy_CH[4:0] ビットにプログラムします。オフセットは、POSOFFFビットの値に応じて、正または負の値にすることができます。POSOFFFがクリアされDLLRDYフラグを、変換された値から、OFFSETy[23:0] ビットに書き込まれたユーザ定義のオフセットが減算されます。結果は負の値とすることができます。そのため、読出しデータは符号付きで、SEXTビットは拡張符号値を表します。

オフセット値は最大変換値より小さくする必要があります（たとえば、14bitモードでは、最大オフセット値は 0x3FFFとなります）。

オフセットは、符号なしデータを符号付きデータに変換するために使用できます（たとえば、14bitモードでは、オフセット値は 0x2000となります）。

オフセット補正は、オーバーサンプリングモードでもサポートされます。オーバーサンプリングモードでは、OVSS右シフトが適用される前にオフセットが減算されます。

表309 に、アナログウォッチドッグ1、2、3に可能なすべての分解能での比較方法を示します。

表309.オフセット計算対データ分解能

分解能（ビットRES[1:0]）	元の変換データとオフセット間での減算／加算		結果	コメント
	元データ（変換済み、左詰め）	オフセット		
00：14bit	DATA[13:0].	OFFSET[13:0]	符号付きまたは符号なしの24bitデータ、[13:0]の右詰め	-
01：12bit	DATA[13:2],00	OFFSET[13:2]	符号付きまたは符号なしの24bitデータ、[11:0]の右詰め	ユーザはOFFSET[11:0] に0b0000 0000 0000を設定する必要があります。

表309.オフセット計算対データ分解能 （続き）

分解能（ビット RES[1:0]）	元の変換データとオフセット間での減算／加算		結果	コメント
	元データ（変換済み、左詰め）	オフセット		
10：10bit	DATA[13:4],0000	OFFSET[13:4]	符号付きまたは符号なしの24bitデータ、[9:0]の右詰め	ユーザはOFFSET[3:0] に0b0000を設定する必要があります。
11：8bit	DATA[13:6],000000	OFFSET[13:6]	符号付きまたは符号なしの24bitデータ、[7:0]の右詰め	ユーザはOFFSET[5:0] に0b000000を設定する必要があります。

図231、図232、図233、および図234に、符号付きおよび符号なしのデータの配置を、対応するOVSS値およびLSHIFT値とともに示します。

図231. 右詰め（オフセット無効、符号なしの値）

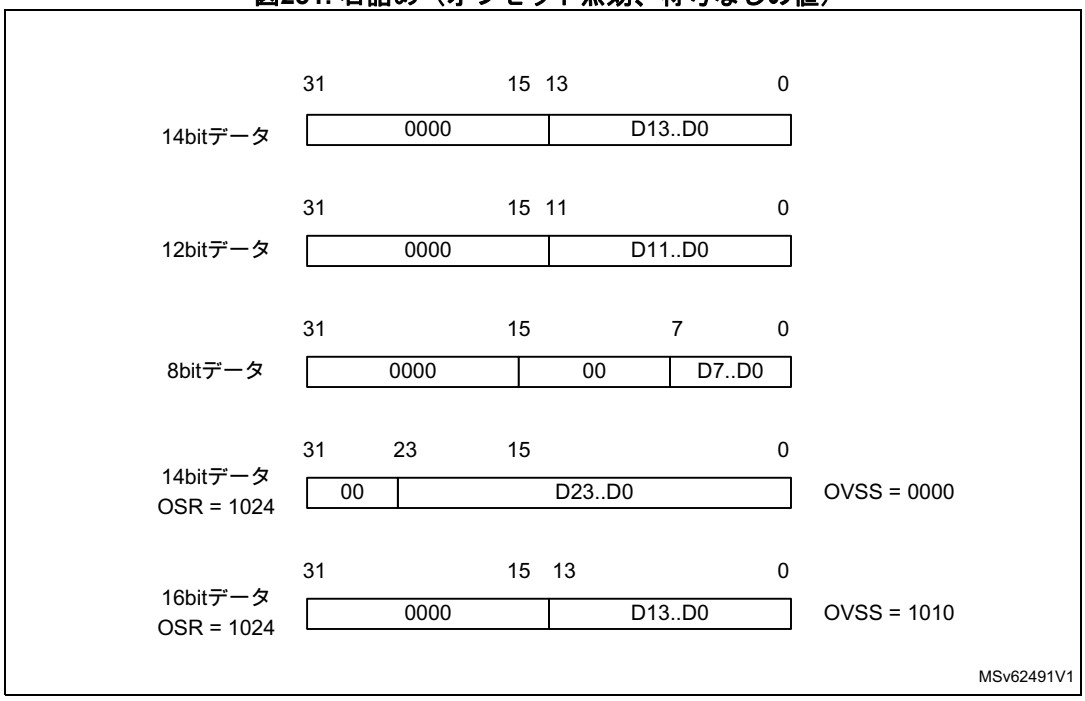


図232. 右詰め（オフセット有効、符号付きの値）

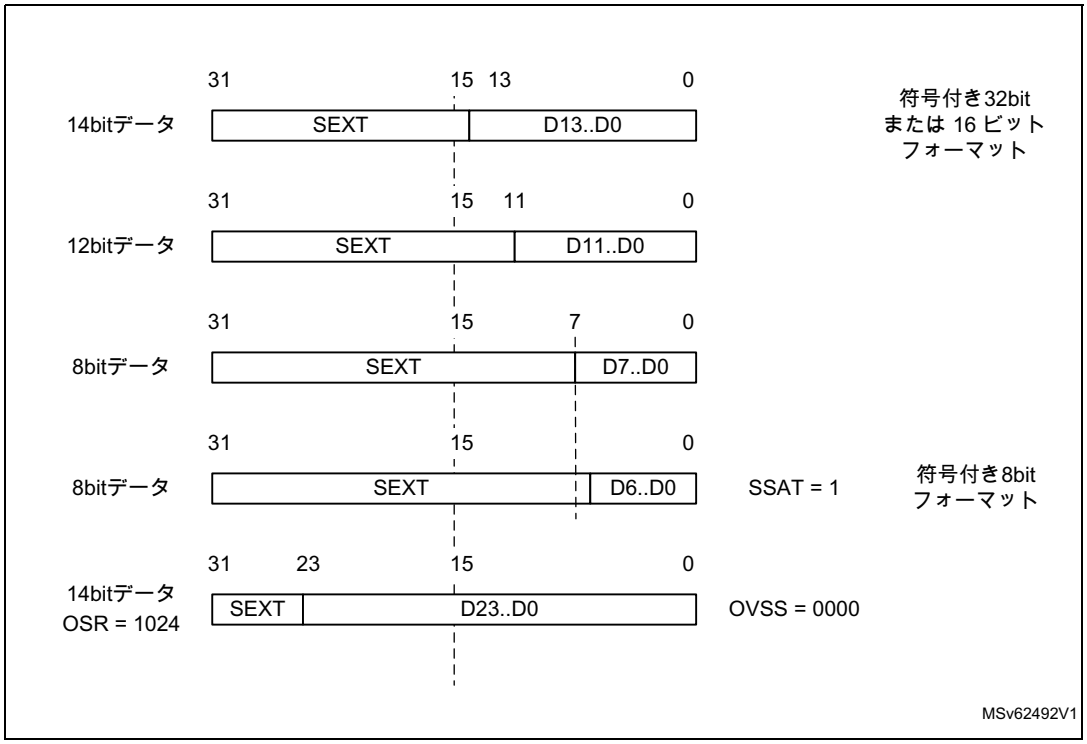


図233. 左詰め（オフセット無効、符号なしの値）

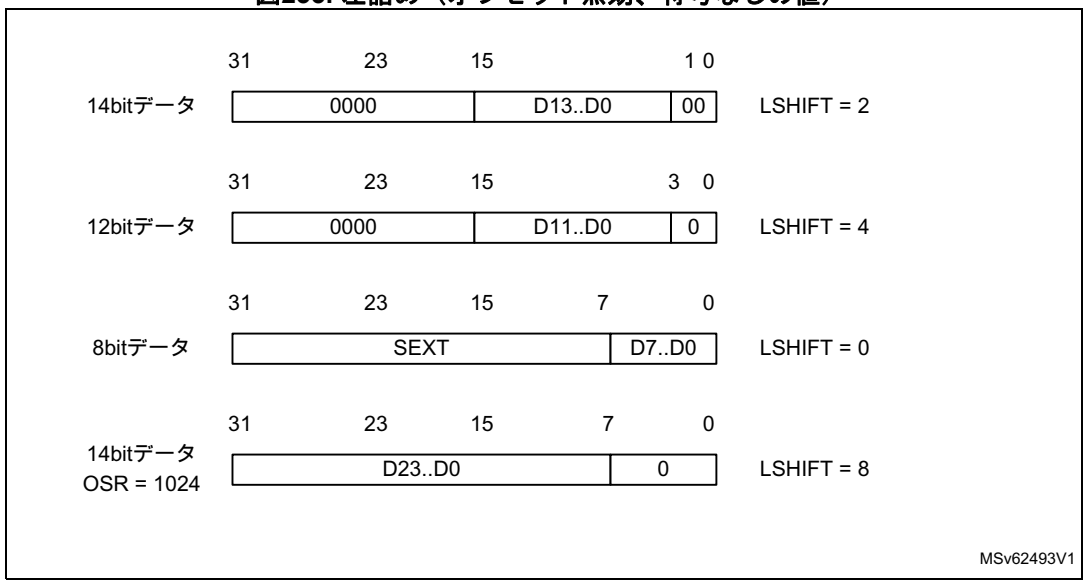
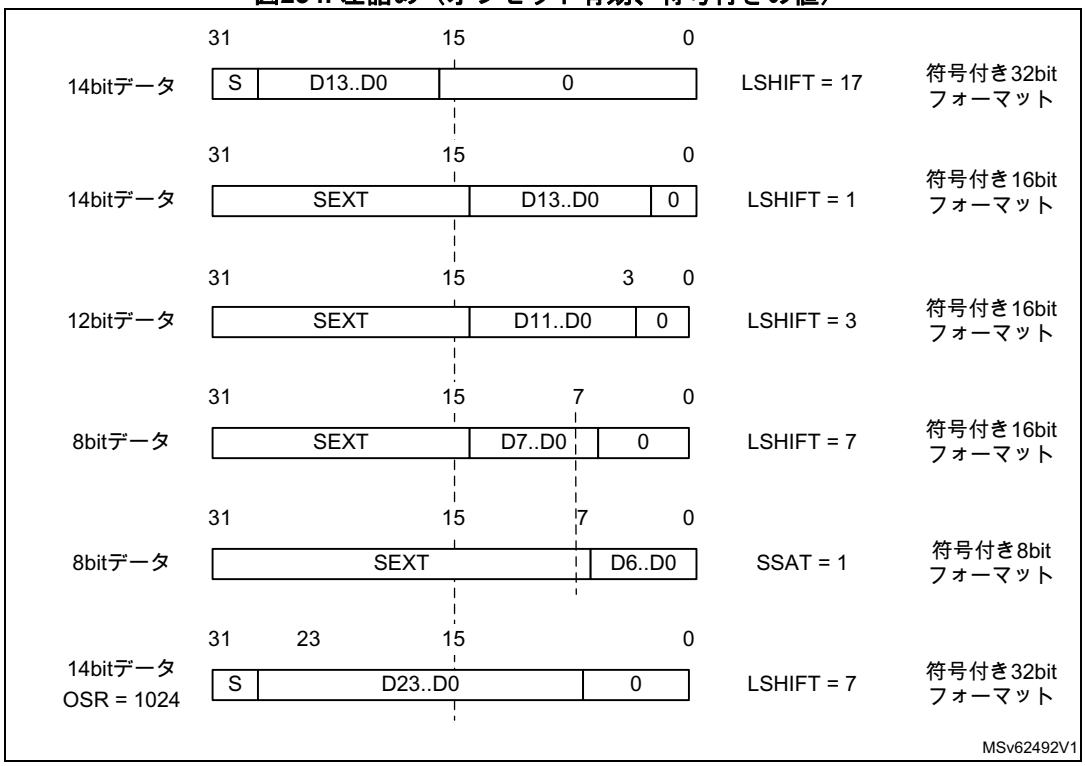


図234. 左詰め（オフセット有効、符号付きの値）



符号付きおよび符号なし飽和フォーマットの管理（SSAT、USAT）

オフセット補正により、データ幅を元のデータより広くすることができます。

元の幅を制限するために、ADC_OFRyレジスタのSSATおよびUSATビットでデータ飽和を有効化することができます。

0x2000以外のオフセット値を使用して、14bitの符号なしデータを15bitの符号付きデータに拡張できます。

元のデータ幅を保持するには、SSATビットをセットしてデータ幅を14bitに制限します。

符号なしデータを元のデータ幅に対して飽和させるには、USATビットにセットします。

表310 に、異なる分解能に対応する符号付き拡張データフォーマットを示します。

表310.14bitデータフォーマット

SSAT	USAT	フォーマット	データ範囲 （オフセット = 0x2000）
0	0	符号拡張された有効桁15bitのデータ SEXT[31:14] DATA[13:0]	0x0000 1FFF ~ 0xFFFF E000
1	0	符号拡張された有効桁14bitのデータ SEXT[31:13] DATA[12:0]	0x1FFF - 0xE000

表310.14bitデータフォーマット （続き）

SSAT	USAT	フォーマット	データ範囲 （オフセット = 0x2000）
0	1	符号なしの飽和の有効桁14bitのデータ DATA[13:0]	0x3FFF - 0x0000
1	1	予約済みです。	-

表311 に、3つの異なるオフセット値に対する数値の例を示します。

表311.16bitフォーマットでの数値の例

生の変換結果	オフセット値	結果 SSAT = 0 USAT = 0	結果 SSAT = 0 USAT = 1	結果 SSAT = 1 USAT = 0
0x3FFF	0x2000	0x0000：1FFF	1FFF	1FFF
0x2000		0x0000 0000	0000	0000
0x0000		0xFFFF：E000	E000	E000
0x3FFF	0x2020	0x0000：1FDF	1FDF	1FDF
0x2000		0xFFFF：FFE0	0000	FFE0
0x0000		0xFFFF：DFE0	0000	DFE0
0x3FFF	0x1FE0	0x0000：201F	201F	1FFF
0x2000		0x0000 0020	0020	0020
0x0000		0xFFFF：E020	0000	E020
0x3FFF	0x20	0x0000：3FDF	3FDF	3FDF
0x2000		0x0000：1FE0	1FE0	1FE0
0x0000		0xFFFF：FFE0	0000	FFE0

注意： **SSATはUSATと一緒に使用しないでください。この推奨事項が順守されるていることを確認するためのハードウェアチェックは行われません。**

ゲイン補正

ADC_CFGR2レジスタでGCOMPビットがセットされているとき、ゲイン補正はすべての変換データで有効です。各変換後、データは次の式で計算されます。

$$DATA = DATA(\text{adc result}) \times (GCOMPCOEFF) / 4096$$

GCOMPCOEFFは0から16383までプログラムできるため、実際のゲイン補正係数は0から3.999756までです。

結果のデータをRDATAまたはJDATAxレジスタに格納する前に、LSB−1値が評価され、データが丸められてエラーが最小化されます。

ゲイン補正は、オーバーサンプリング にも効果的です。ゲイン補正がオーバーサンプリング モードで使用されるとき、電力消費を最小化するために、ゲイン計算は累積および右シフト操作後に行われます（ゲイン計算は、変換ごとではなく、一度だけ行われます）。内部乗算器の幅は32bitで、ゲイン補正のための入力データ幅は18bit未満でなければなりません。インジェクト変換モードおよびレギュラ変換モードでオーバーサンプリングを使用する場合、保留中の変換を適切な値で再開するには、ADC_CFGR2レジスタのROVSMビットをセットする必要があります。

ADCのオーバーラン（OVRMOD）

オーバーランフラグ（OVR）は、ADC_DR FIFO（8ステージ）がオーバーフローする前に（CPUまたはDMAによって）レギュラ変換データが読み出されなかったときにバッファオーバーランイベントが発生したことを通知します。

OVRフラグは、ADC_CRレジスタのFIFOがフルの間に新しい変換が完了するとセットされます。OVRIEビットがセットされている場合、割込みが生成されます。

オーバーランイベントが発生したとき、ADCは動作し続け、ソフトウェアがADSTPビットをセットすることによってシーケンスの停止およびリセットを決めるまで、変換を続行できます。

OVRフラグは、ソフトウェアによって1を書き込むことによってクリアされます。

オーバーランイベントの発生時にデータを保持するか上書きするように設定するには、ADC_CFGR1レジスタのOVRMOD制御ビットをプログラムします。

- OVRMOD = 0

  オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータはADC_DR FIFO深さ（8データ）まで保持され、新しい変換は破棄され、失われます。OVRが1のままの場合、その後の変換は行われますが、結果データは同様に破棄されます。
- OVRMOD = 1

  データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。このモードでは、ADC_DR FIFOが無効になります。OVRが1のままの場合、その後の変換は通常どおり動作し、ADC_DRレジスタには常に最新の変換データが格納されます。

図235. オーバーラン（OVRMOD = 0）の例

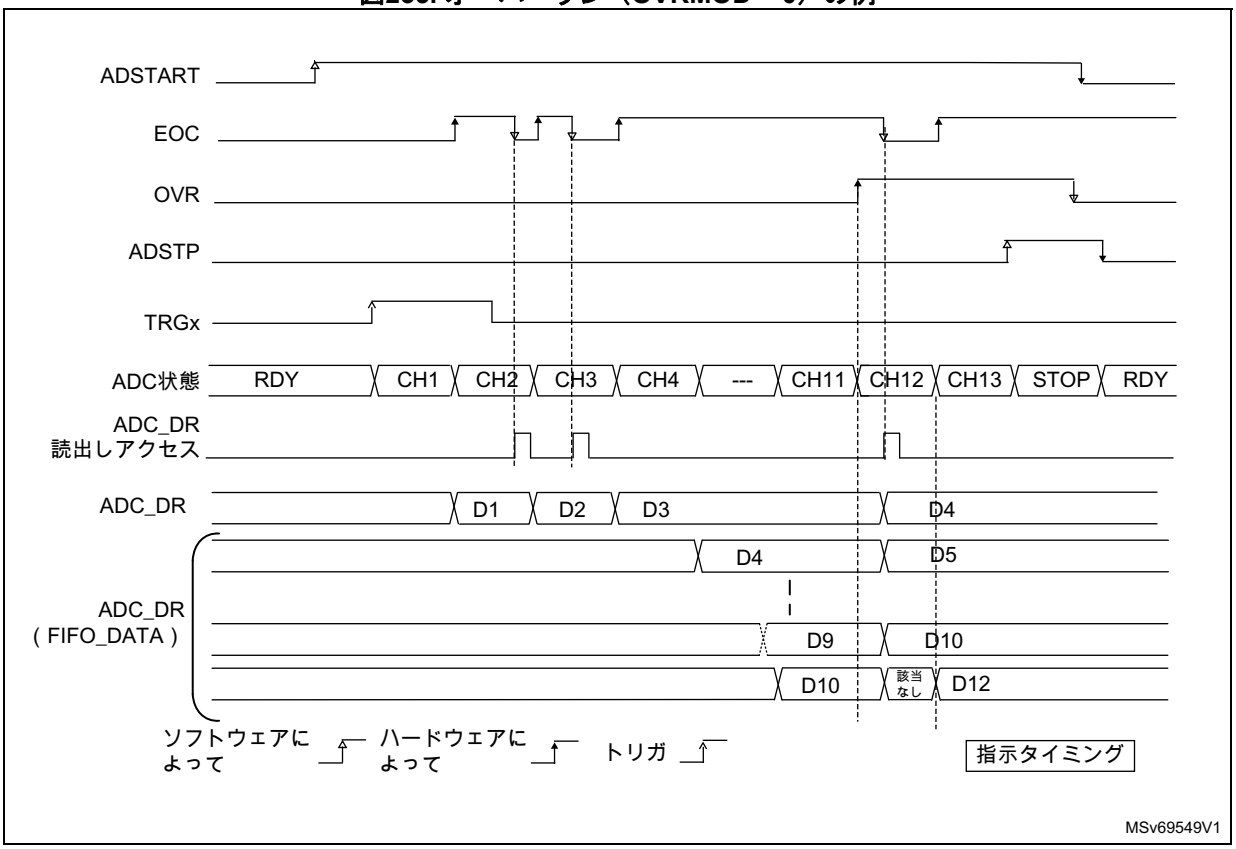
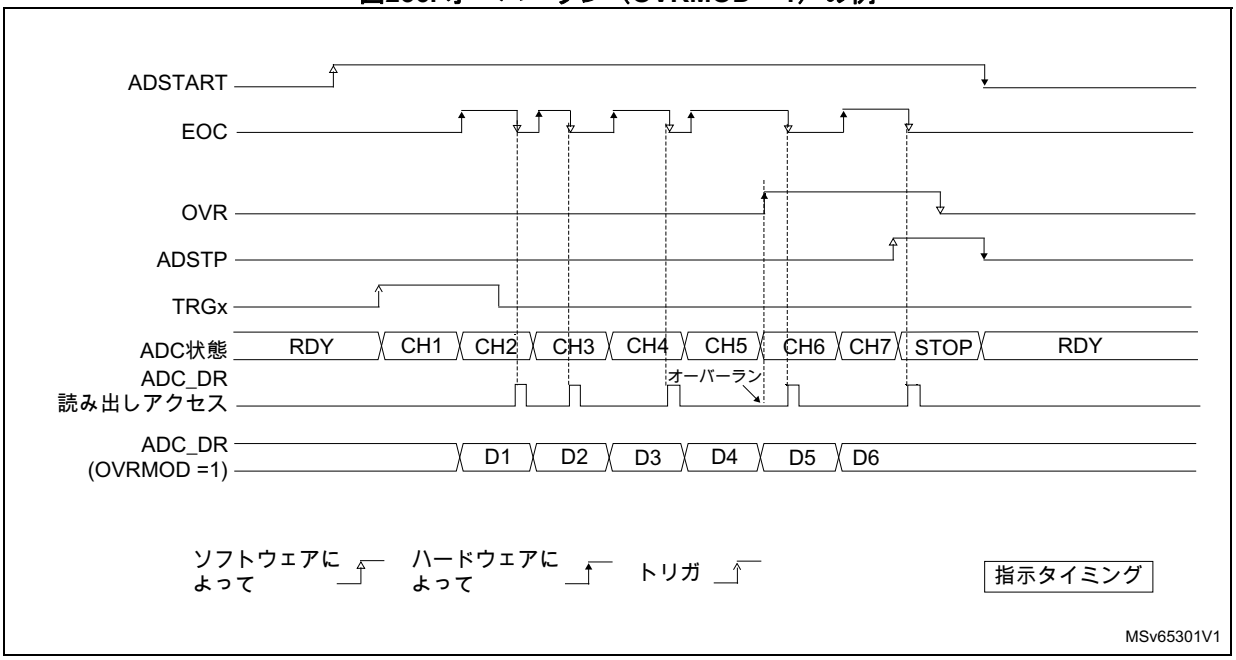


図236. オーバーラン（OVRMOD = 1）の例



注： **インジェクトチャネルでは、4 つのインジェクトチャネルにそれぞれ専用のデータレジスタがあるため、オーバーランを検出する機能はありません。**

DMAを使用しない変換シーケンスの管理

変換が十分遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアではEOCフラグと関連の割込みを使用して、各データを処理する必要があります。変換が終了するごとに、EOCはセットされ、ADC_DRレジスタを読み出すことができます。オーバーランイベントをエラーとして管理するには、OVRMODに0を設定する必要があります。

オーバーランなしでのDMAを使用しない変換の管理

毎回データの読出しをせずにADCに1つまたは複数のチャネルを変換させると便利な場合があります（たとえば、デバイスにアナログウォッチドッグがある場合）。この場合、OVRMODビットに1を設定する必要があり、ソフトウェアはOVRフラグを無視する必要があります。オーバーランイベントが発生してもADCは変換を続行し、ADC_DRレジスタには常に最新の変換が含まれます。

DMAを使用した変換の管理

変換されたチャネルの値は特定のデータレジスタに格納されるので、複数のチャネルの変換にはDMAの使用が便利です。これによって、ADC_DRレジスタにすでに格納されているデータの損失を防ぐことができます。

DMAモードが有効なとき（シングルADCモードでADC_CFGRレジスタでDMNGT[1:0] = 01または11にセットされている場合、またはデュアルADCモードでDAMDFが00以外に設定されている場合）、各チャネルの変換後、DMAリクエストが生成されます。これにより、変換されたデータをADC_DRレジスタからソフトウェアで選択した場所へ転送することができます。

これにもかかわらず、DMAがDMA転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR=1）、ADCはDMAリクエストの生成を停止し、新しい変換に対応するデータはDMAによって転送されません。つまり、RAMへ転送されたデータはすべて有効とみなすことができます。

OVRMODビットの設定に応じて、データは保持または上書きされます。

DMA転送リクエストは、ソフトウェアがOSVビットをクリアするまでブロックされます。

アプリケーションによって、2種類のDMAモードが提案されます。これらのモードは、シングルADCモードではADC_CFGR1レジスタのDMNGT[1:0] ビットフィールドを通して、デュアルADCモードではADC12_CCRレジスタのDAMDFビットを通して設定できます。

- DMAワンショットモード（DMNGT[1:0] = 01）
  このモードは、DMAが固定数のデータを転送するようにプログラムされた場合に適しています。
- DMAサーキュラモード（DMNGT[1:0] = 11）
  このモードは、DMAをサーキュラモードでプログラムする場合に適しています。

DMAワンショットモード（DMNGT[1:0] = 01）

このモードでは、ADCは新しい変換データが使用可能になるたびにDMA転送リクエストを生成し、変換が再び開始された場合でも、DMAが最後のDMA転送に達すると（転送完了割込みが発生すると）（DMAセクションを参照）、DMAリクエストの生成を停止します。

DMA転送完了後（DMAコントローラで設定されたすべての転送後）:

- ADCデータレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMAコントローラに対する新しいDMAリクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMAは停止します。

DMAサーキュラモード（DMNGT[1:0] = 11）

このモードでは、DMAが最後のDMA転送に達した場合でも、ADCは新しい変換データがデータレジスタで使用可能になるたびにDMA転送リクエストを生成します。これにより、DMAを連続的なアナログ入力データ·ストリームを処理するようにサーキュラモードに設定できます。

FIFO付きDMA

出力データレジスタは、8ステージのFIFOを備えています。2種類のDMAリクエストが並行して生成されます。1つのデータが利用できる場合、「SREQシングルリクエスト」が生成されます。4つのデータが利用できる場合、「BREQバーストリクエスト」が生成されます。このモードに応じて、DMAは、シングル転送モードとインクリメンタルバーストモード（4ビート）のどちらでもプログラム可能です。正しいリクエストラインがDMAによって選択されます。詳細については、DMAの章を参照してください。

33.4.28 MDFを使用した変換の管理

ADCの変換結果はMDFに直接転送することができます。

この場合、DMNGT[1:0] ビットは10にセットしてください。

ADCは、レギュレータデータレジスタの最下位16bitをadcx_dat[15:0] バスを通してMDFに転送し、転送が有効になるとEOCフラグをリセットします。

データフォーマットは、16bit符号付きフォーマットである必要があります。

ADC_DR[31:16] = 無視

ADC_DR[15] = 符号

ADC_DR[14:0] = データ

16bit符号付きフォーマットを超える値は切り捨てられます。

33.4.29 動的低電力機能

自動遅延変換モード（AUTDLY）

ADCは、AUTDLY設定ビットによって制御されている自動遅延変換モードを実装しています。自動遅延変換は、ソフトウェアを単純化する場合だけでなく、ADCオーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化する場合に便利です。

AUTDLY = 1の場合、新しい変換は同じグループの前のデータがすべて処理された場合にのみ開始できます。

- レギュラ変換の場合：ADC_DRレジスタの読出し後、またはEOCビットのクリア後（図237 を参照）
- インジェクト変換の場合：JEOSビットのクリア後（図238 を参照）

自動遅延変換モードでは、ADCの速度をデータを読み出すシステムの速度に自動的に適応させることができます。

遅延は、各レギュラ変換後（DISCEN = 0か1かにかかわらず）、およびインジェクト変換の各シーケンス後（JDISCEN = 0か1かにかかわらず）に挿入されます。

注： **遅延は、インジェクトシーケンスの各変換の間には挿入されません（最後の変換を除く）。**

変換時、この遅延中に発生した（同じ変換グループに対する）ハードウェアトリガイベントは無視されます。

注： **ソフトウェアトリガは、変換を再開するために、この遅延中に ADSTART ビットまたは JADSTART ビットをセットできるようにしている点で、ハードウェアトリガとは異なります。新しい変換を起動する前にデータを読み出すかどうかは、ソフトウェアに依存します。**

遅延は、異なるグループの変換（レギュラ変換に続いてインジェクト変換、またはその逆）の間には挿入されません。

- レギュラ変換の自動遅延中にインジェクトトリガが発生すると、すぐにインジェクト変換が開始されます（図238 を参照）。
- インジェクトシーケンスが完了すると、新しいレギュラ変換を開始する前に、ADCは前のレギュラ変換の遅延を待ちます（終了していない場合）（図240 を参照）。

自動インジェクトモード（JAUTO = 1）では挙動が少し異なり、前のインジェクト変換シーケンスの自動遅延が終了した場合（JEOSがクリアされた場合）にのみ新しいレギュラ変換を開始することができます。これは、新しいシーケンスを開始する前に、ソフトウェアが特定のシーケンスのデータをすべて読み出すことができるようにするためです（図241 を参照）。

自動遅延モード（JAUTO = 1、CONT = 1、およびAUTDLY = 1）と組み合わされた連続自動インジェクションモードの変換を停止するには、次の手順に従います。

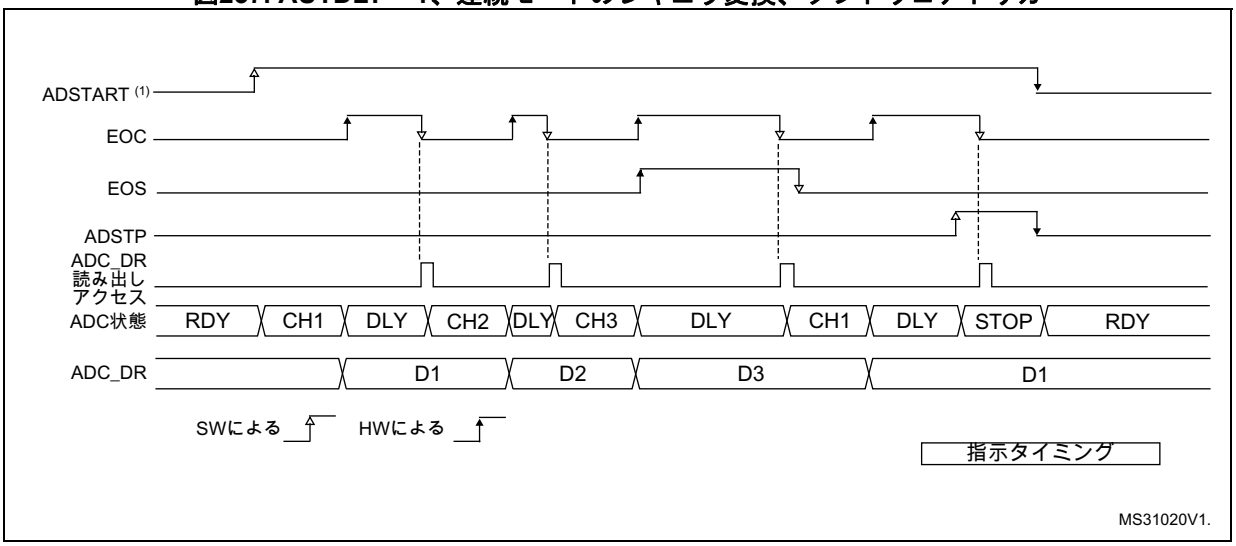
1. JEOS = 1になるまで（リスタートされる変換がなくなるまで）待ちます。
2. JEOSをクリアします。
3. ADSTP = 1にセットします。
4. レギュラデータを読み出します。

この手順に従わない場合、ADSTPがセットされた後でJEOSがクリアされると、新しいレギュラシーケンスが再開されます。

AUTDLYモードでは、すでに実行中のレギュラシーケンス中、またはシーケンスの最後のレギュラ変換後の遅延中に発生した場合、ハードウェアレギュラトリガイベントは無視されます。ただし、この遅延後に発生した場合は、後続の遅延のインジェクトシーケンス中に発生した場合でも、保留とみなされます。その後、インジェクトシーケンスの遅延終了時に変換が開始されます。

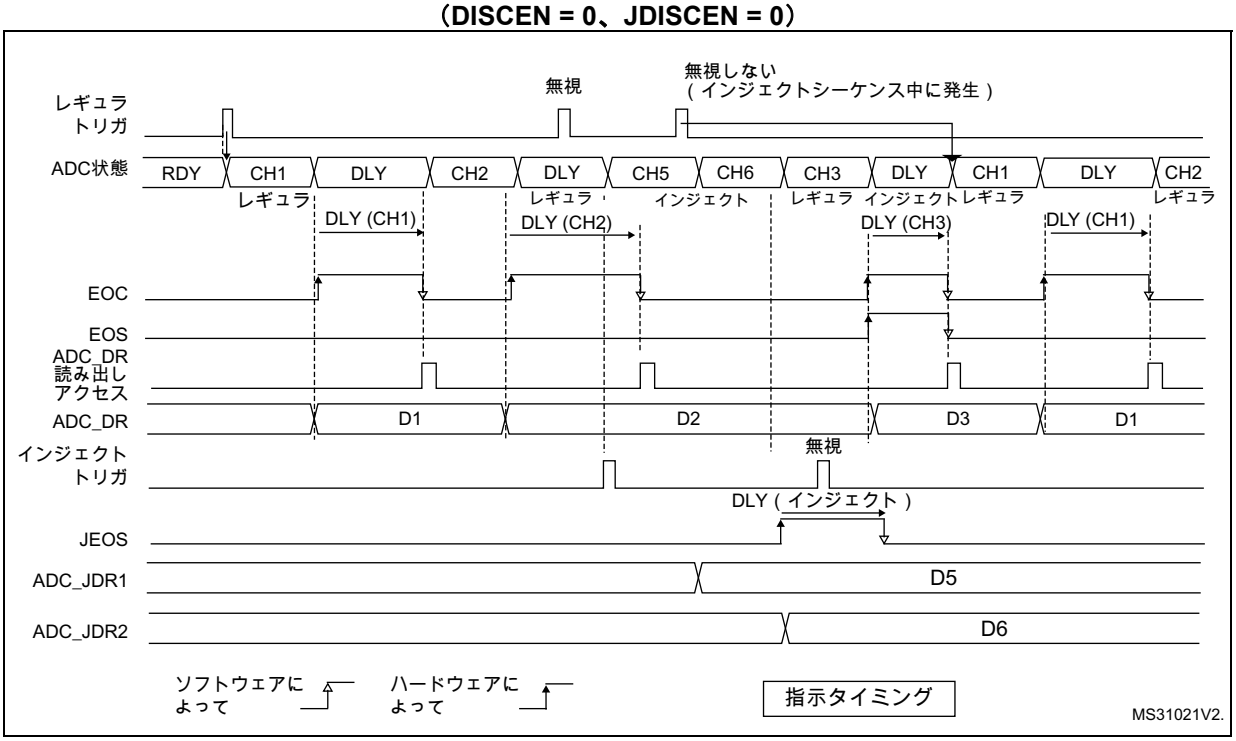
AUTDLYモードでは、すでに実行中のインジェクトシーケンス中、またはシーケンスの最後のインジェクト変換後の遅延中に発生した場合、ハードウェアインジェクトトリガイベントは無視されます。

図237. AUTDLY = 1、連続モードのレギュラ変換、ソフトウェアトリガ



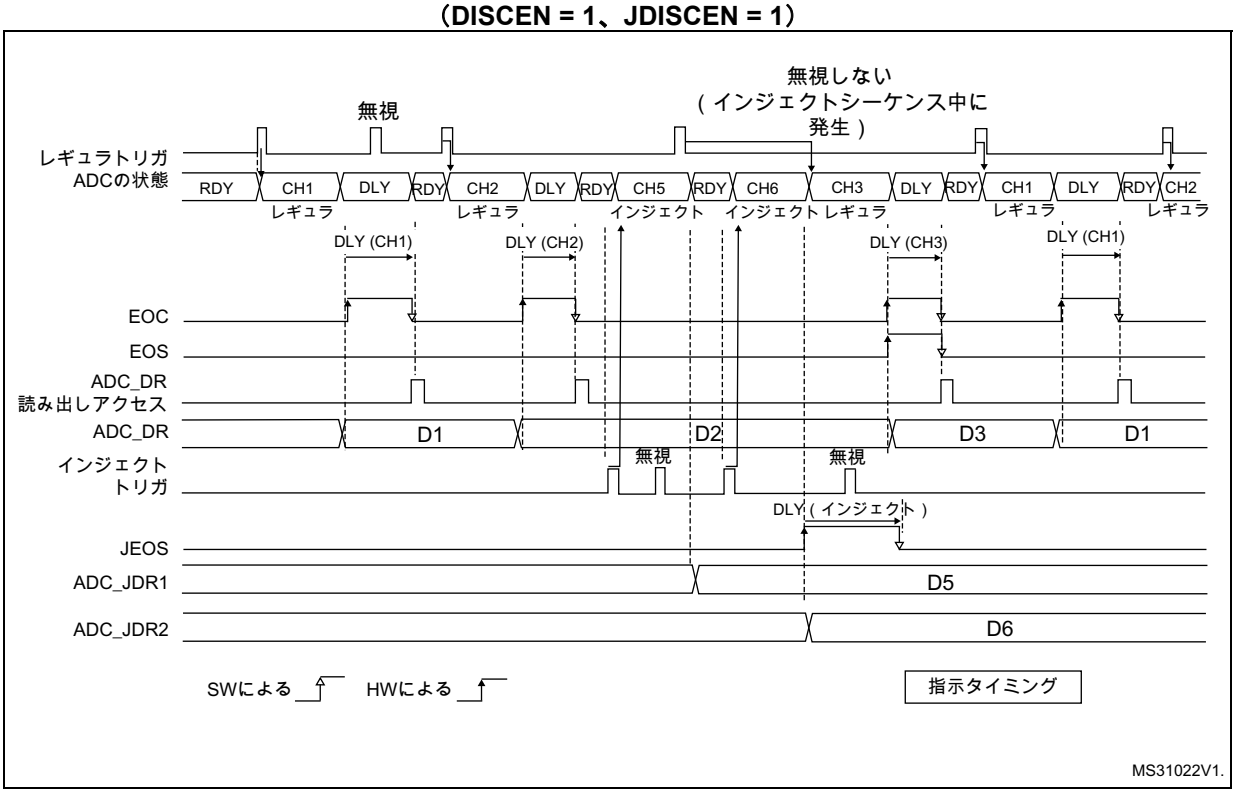
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 0x0（ソフトウェアトリガ）、CONT = 1、CHANNELS = 1、2、3。
3. インジェクト設定：無効

図238. AUTODLY = 1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN = 0、JDISCEN = 0）



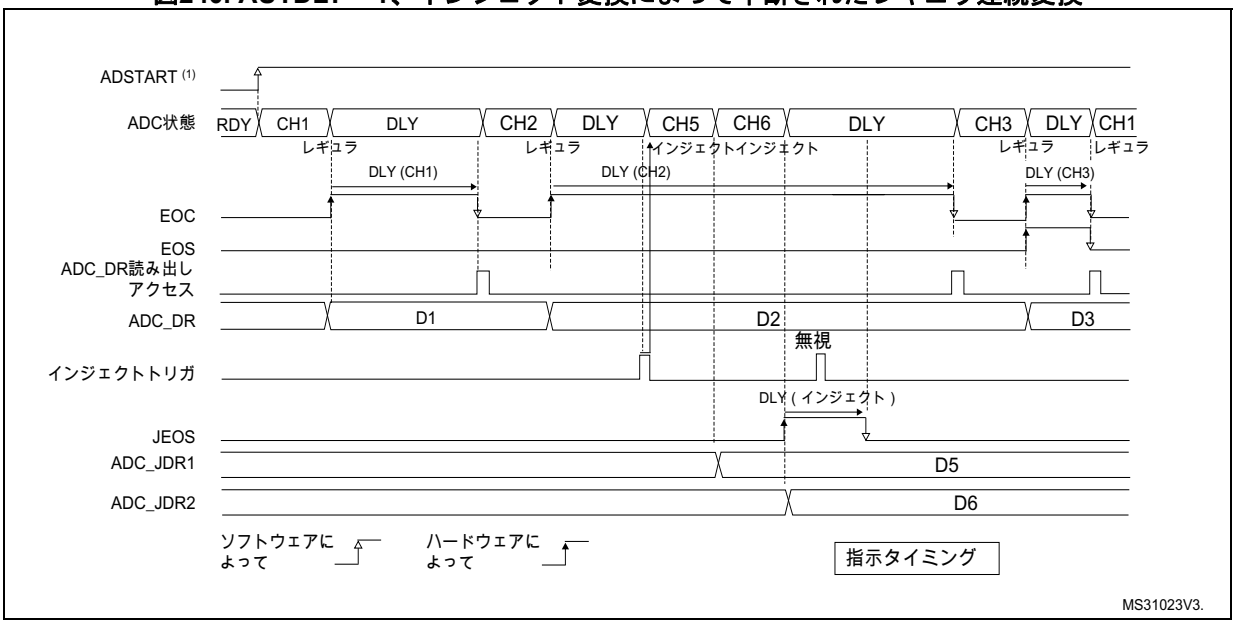
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 0x1（ハードウェアトリガ）、CONT = 0、DISCEN = 0、CHANNELS = 1、2、3。
3. インジェクト設定：JEXTEN[1:0] = 0x1（ハードウェアトリガ）、JDISCEN = 0、CHANNELS = 5、6。

図239. AUTODLY = 1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN = 1、JDISCEN = 1）



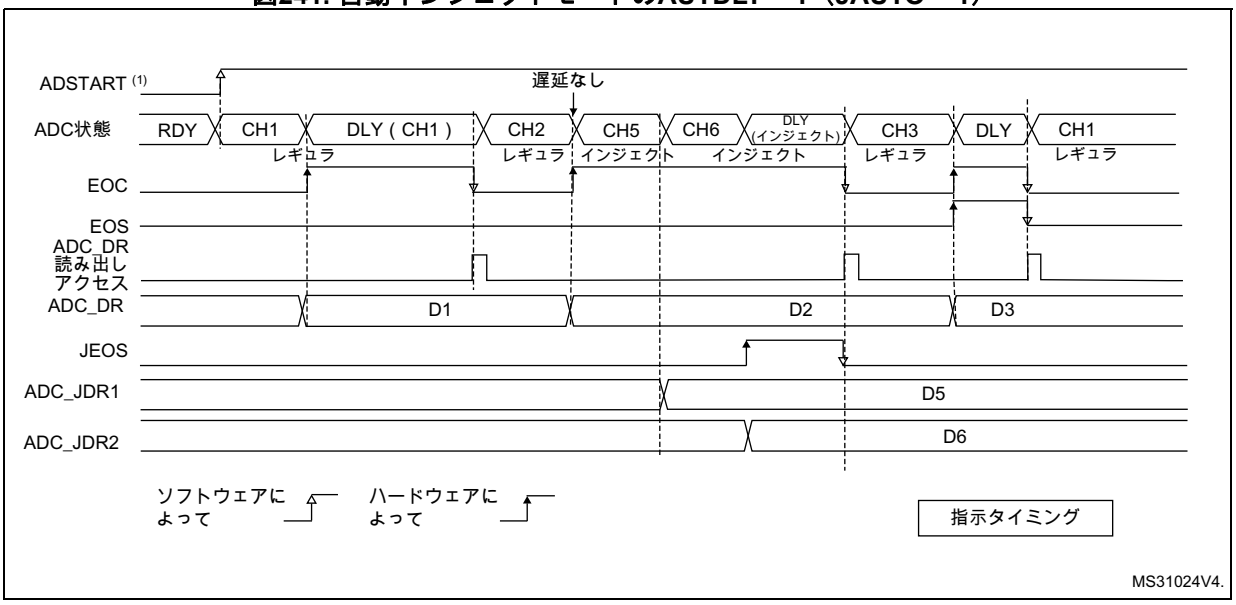
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 0x1（ハードウェアトリガ）、CONT = 0、DISCEN = 1、DISCNUM = 1、CHANNELS = 1、2、3。
3. インジェクト設定：JEXTEN[1:0] = 0x1（ハードウェアトリガ）、JDISCEN = 1、CHANNELS = 5、6。

図240. AUTDLY = 1、インジェクト変換によって中断されたレギュラ連続変換



1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 0x0（ソフトウェアトリガ）、CONT = 1、DISCEN = 0、CHANNELS = 1、2、3。
3. インジェクト設定：JEXTEN[1:0] = 0x1（ハードウェアトリガ）、JDISCEN = 0、CHANNELS = 5、6。

図241. 自動インジェクトモードのAUTDLY = 1（JAUTO = 1）

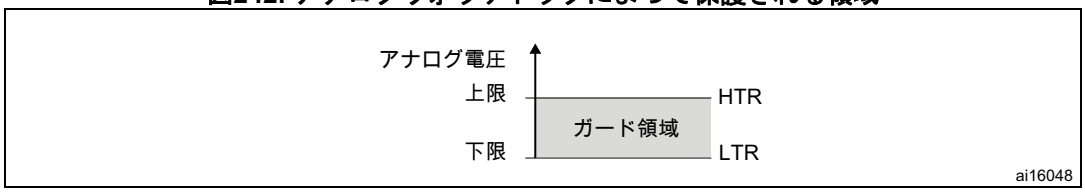


1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 0x0（ソフトウェアトリガ）、CONT = 1、DISCEN = 0、CHANNELS = 1、2。
3. インジェクト設定：JAUTO = 1、CHANNELS = 5、6。

33.4.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）

3つのAWDアナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

図242. アナログウォッチドッグによって保護される領域



AWDxフラグと割込み

ADC_IERレジスタのAWDyIE （y=1、2、3）をセットすることによって、3つのアナログウォッチドッグでそれぞれ割込みを有効にできます。

AWDy （y=1、2、3）フラグは、1を書き込むことにより、ソフトウェアによってクリアされます。

ADC変換結果は配置される前に低閾値および高閾値と比較されます。

アナログウォッチドッグ1の説明

AWDアナログウォッチドッグ1は、ADC_CFGR1レジスタのAWD1ENビットをセットすることによって有効になります。ウォッチドッグは、選択された1つのチャネルまたはすべての有効チャネルが設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視します。

表312に、1つまたは複数のチャネル上でアナログウォッチドッグを有効にするためにどのようにしてADC_CFGR1レジスタを設定するかを示します。

表312.アナログウォッチドッグチャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGLビット	AWD1ENビット	JAWD1ENビット
なし	x	0	0
すべてのインジェクトチャネル	0	0	1
すべてのレギュラチャネル	0	1	0
すべてのレギュラおよびインジェクトチャネル	0	1	1
1つの(1) インジェクトチャネル	1	0	1
1つの (1) レギュラチャネル	1	1	0
1つの (1) レギュラまたはインジェクトチャネル	1	1	1

1. AWDyCH[4:0] ビットによって選択チャネルは、適切なレギュラまたはインジェクトシーケンスで変換されるようにプログラムする必要があります。

アナログウォッチドッグ（AWD1）ステータスビットは、ADCによって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。

これらの閾値は、アナログウォッチドッグ1用に、ADC_HTR1レジスタのHTR1[24:0]ビットおよびADC_LTR1レジスタのLTR1[24:0]ビットでプログラムされます。

閾値は、最高25bit（オーバーサンプリング付き14bit分解能、OSR=256）まで可能です。

14bit未満の分解能でデータを変換するときには（RES[1:0] ビットに従って）、内部比較は常に14bit全体の変換データ（半ワード境界で左詰め）に対して実行されるため、プログラムされた閾値のLSBはクリアされたままである必要があります。

表313 に、アナログウォッチドッグ1、2、3に可能なすべての分解能での比較方法を示します。

表313.アナログウォッチドッグ1、2、3の比較

分解能（RES[1:0] ビット）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：14bit	DATA[13:0]	LTR1[24:0] および HTR1[24:0]	-
01：12bit	DATA[13:2],00	LTR1[24:0] および HTR1[24:0]	ユーザはLTR1[1:0] とHTR1[1:0] を00に設定する必要があります。
10：10bit	DATA[13:4],0000	LTR1[24:0] および HTR1[24:0]	ユーザはLTR1[3:0] とHTR1[3:0] を0000に設定する必要があります。
11：8bit	DATA[13:6],000000	LTR1[24:0] および HTR1[24:0]	ユーザはLTR1[5:0] とHTR1[5:0] を000000に設定する必要があります。

1. ウォッチドッグの比較は、オーバーサンプリング、ゲイン補正、オフセット補正が完了したときに行われます（符号付きまたは符号なしのデータを比較できます）。

ウォッチドッグ1のアナログウォッチドッグフィルタ

ADCが1つの入力チャネルだけで設定されているとき（スキャンモードで複数のチャネルを選択することはできない）、有効なADC変換データ範囲をADC_LTR1およびADC_HTR1レジスタで設定できます。

- 変換データがADC_LTR1およびADC_HTR1で定義された範囲に入ると、DMAリクエストが生成されます。
- そうでない場合、DMAリクエストは発行されません。RDATAレジスタは、変換ごとに更新されます。データが範囲外になった回数が、ADC_HTR1のAWDFILTビットで指定された値を上回った場合、AWDxフラグがセットされ、対応する割込みが発行されます。

アナログウォッチドッグ2および3の説明

2番目と3番目のアナログウォッチドッグはより柔軟性の高く、AWDCHy[19:0]（y=2、3）の対応するビットをプログラムすることにより、いくつかの選択されたチャネルをガードすることができます。

対応するウォッチドッグは、AWDCHy[19:0]（y=2、3）の任意のビットがセットされると有効になります。

閾値は、最高25ビット（オーバーサンプリング付き14bit分解能、OSR = 1024、および符号付きフォーマットのオフセット変換）とすることができ、ADC_HTR2、ADC_LTR2、ADC_LTR3、およびADC_HTR3レジスタでプログラムされます。

14bit未満の分解能でデータを変換するときには（RES[1:0] ビットに従って）、内部比較は常に14bit全体の変換データ（左詰め）に対して実行されるため、プログラムされた閾値のLSBはクリアされたままである必要があります。

ADC_AWDy_OUT信号出力の生成

各アナログウォッチドッグは、一部のオンチップタイマにETR入力（外部トリガ）が直接接続された内部ハードウェア信号ADCy_AWDx_OUT（yはウォッチドッグの数）と関連付けられています。ETRとしてADC_AWDy_OUT信号を選択する方法を理解するには、オンチップタイマのセクションを参照してください。

ADC_AWDy_OUTは、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADC_AWDy_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADC_AWDy_OUT は、次のガードされた変換がプログラムされた閾値内にある場合、終了後にリセットされます（次のガードされた変換がプログラムされた閾値の外にある場合は1のまま維持されます）。
- ADC_AWDy_OUTはADCを無効にする場合にもリセットされます（ADDIS = 1にセットする場合）。レギュラまたはインジェクト変換の停止（ADSTP = 1またはJADSTP = 1に設定）は、ADCy_AWDx_OUTの生成にまったく影響しない点に注意してください。

注： **AWDx フラグはハードウェアによりセットされ、ソフトウェアによりリセット：AWDx フラグは、ADCy_AWDx_OUTの生成にまったく影響しません（例：ソフトウェアによってAWDxフラグがクリアされず、このフラグが1のまま維持される間、ADCy_AWDx_OUTをトグルできます）。**

図243. ADCy_AWDx_OUT信号生成（すべてのレギュラチャネル）

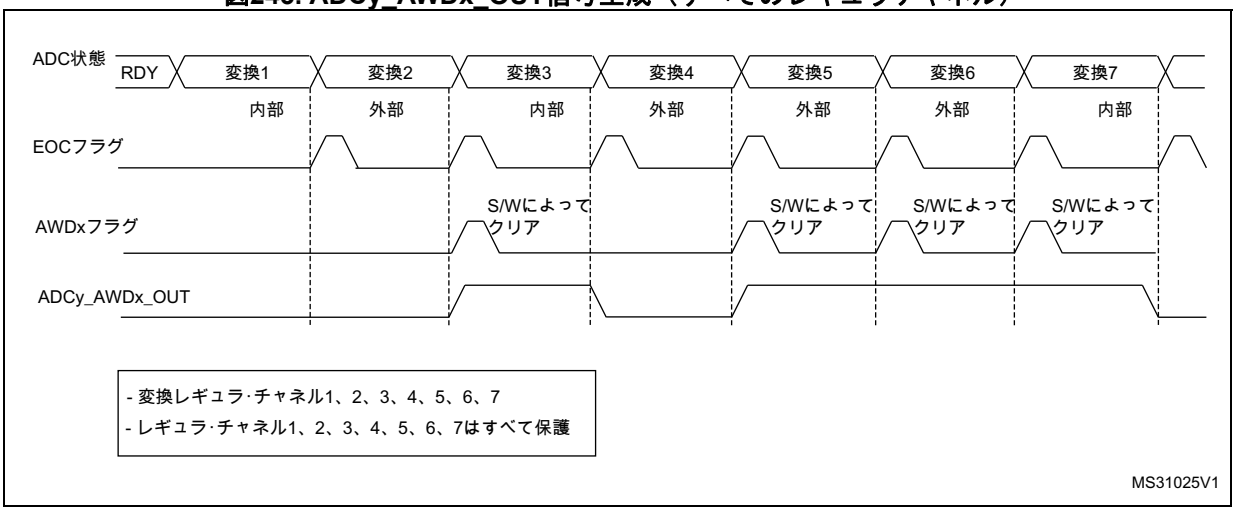


図244. ADC_AWDx_OUT信号生成（ソフトウェアによってAWDxフラグがクリアされない場合）

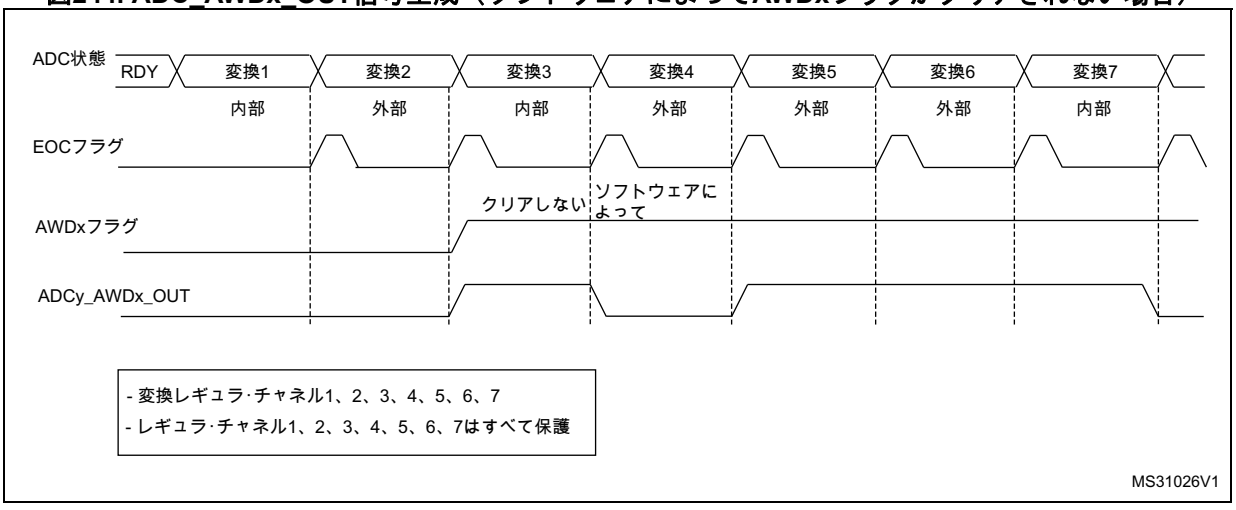


図245. ADC_AWDx_OUT信号の生成（1つのレギュラチャネル）

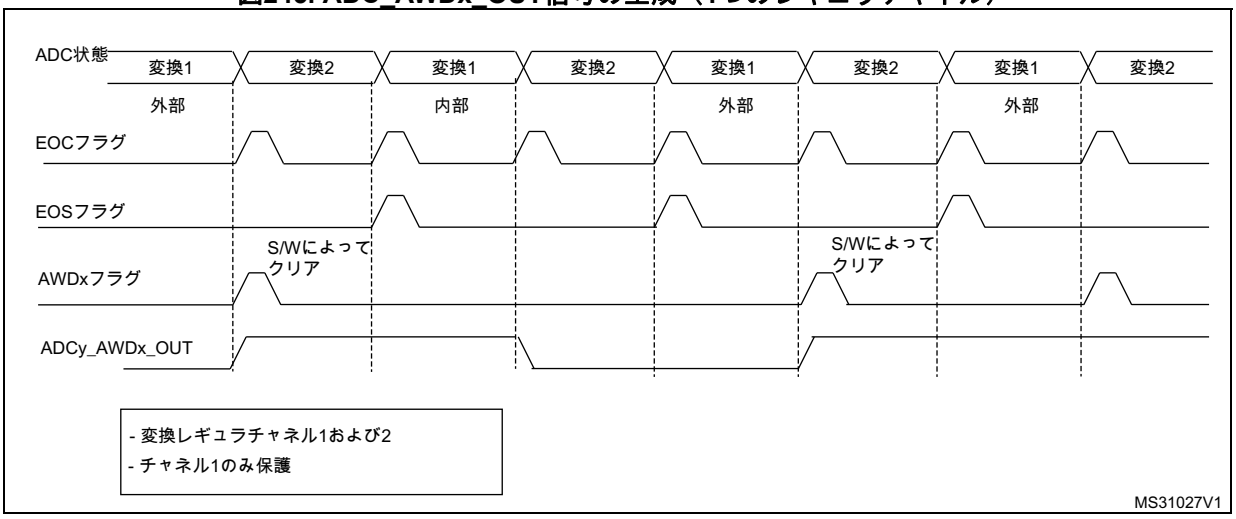
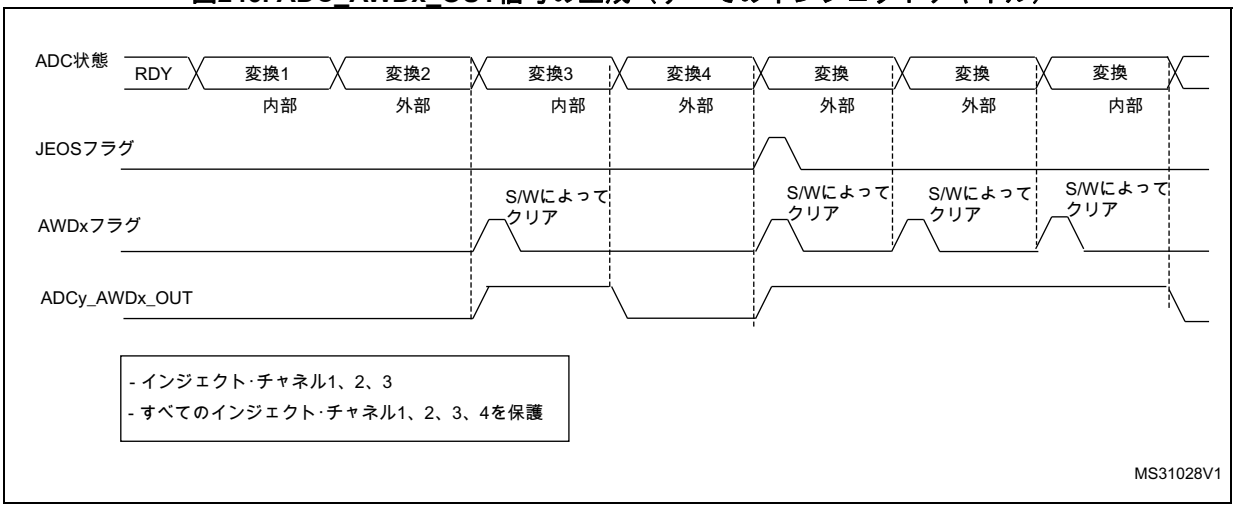


図246. ADC_AWDx_OUT信号の生成（すべてのインジェクトチャネル）



アナログウォッチドッグ閾値制御

LTRx[24:0] およびHTRx[24:0] は、アナログデジタル変換中（ADC内部状態の変換開始から変換終了後までの間）に変更できます。LTRx[24:0] およびHTRx[24:0] が、ADCガードチャネルのADC変換中に更新された場合、この変換については、ウォッチドッグ機能はマスクされます。このマスクは、次の変換の開始時に除去され、その結果、アナログウォッチドッグ閾値が次のADC変換から適用されます。アナログウォッチドッグ比較は、変換が終了するたびに行われます。現在のADCデータが新しい間隔から外れていた場合、割込みまたはADC_AWDx_OUT信号は発行されません。割込みおよびAWD生成は、閾値更新後に開始された変換の終了時にのみ行われます。AWDx_OUTがすでにアサートされていた場合、新しい閾値をプログラムしても、AWD_OUT信号はデアサートされません。

アナログウォッチドッグとゲインおよびオフセット補正

ゲインおよびオフセット補正が有効なとき、アナログウォッチドッグは閾値と補正後のデータを比較します。

33.4.31 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPUの負荷を軽減します。複数の変換を処理して、最大24bit（14bitの値とOSR = 1024）幅の単一データに平均化できます。

以下の形式で結果を提供します。NおよびMは調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

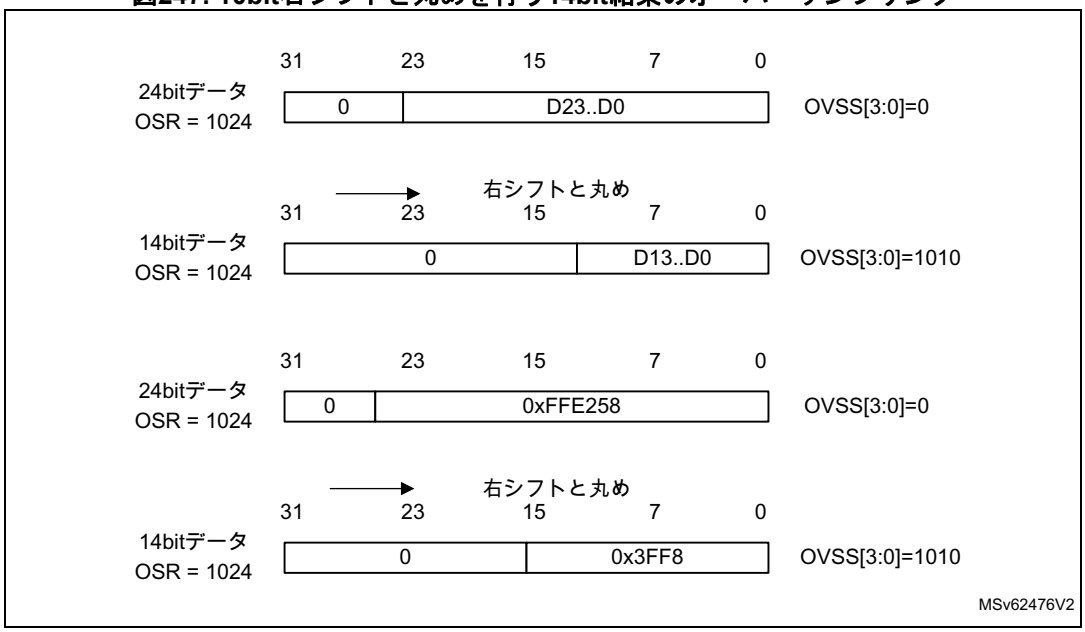
ハードウェアにより、平均化、データレートの削減、SNRの向上、基本的フィルタリングの機能を実行することができます。

オーバーサンプリング比Nは、ADC_CFGR2レジスタのOSR[9:0] ビットによって定義され、その範囲は2～1024倍です。分周係数Mは、最大10bitの右ビットシフトから成っており、ADC_CFGR2レジスタのOVSS[3:0] ビットを使用して定義します。

合計ユニットは最大24bit（1024 x 14bitの結果）の結果をもたらし、この値は左または右へシフトできます。右シフトが選ばれた場合には、シフトによって残った最下位側ビットを使用して最も近い値に丸められてからADC_DRデータレジスタに転送されます。

図247に、元の24bitの累積データから最終的な14bitの結果への処理の数値例を示します。

図247. 10bit右シフトと丸めを行う14bit結果のオーバーサンプリング



オーバーサンプリングモードでの変換のタイミングは変化なし：サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータはN回の変換ごとに提供され、同等遅延は$N \times T_{CONV} = N \times (t_{SMPL} + t_{SAR})$に等しくなります。フラグは、次のようにセットされます。

- サンプリングフェーズの終了（EOSMP）は、各サンプリングフェーズ後にセットされます。
- 変換の終了（EOC）は、N回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOS）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N xシーケンス長の変換合計後）。

オーバーサンプリング中にサポートされる動作モード

オーバーサンプリングモードでは、ほとんどのADC動作モードが維持されます。

- シングルまたは連続モードの変換
- ソフトウェアまたはトリガによるADC変換の開始
- 変換中のADCの停止（中止）
- オーバーラン検出時のCPUまたはDMA経由でのデータの読出し
- 低電力モード（AUTDLY）
- プログラム可能な分解能：この場合、削減された変換値（ADC_CFGR1レジスタのRES[1:0] ビットで設定されます）の累積、切詰め、丸め、およびシフトは、14bit変換と同様に行われます。

注： **オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADC_CFGR の ALIGN ビットは無視され、データは常に右詰めで提供されます。**

オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSEおよび／またはJOVSEビットがセットされている場合、ADC_OFRyレジスタのOFFSETy_ENビットの内容は無視されます（リセットとみなされます）。

アナログウォッチドッグ

アナログウォッチドッグ機能は保持されますが、次のような違いがあります。

- RES[1:0] ビットは無視されます。比較は常に25bit値HTRx[24:0] およびLTRx[24:0] を使用して行われます。
- 比較は、シフトする前にオーバーサンプリングされた合計値に対して行われます。

注： **高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなるからです。たとえば、オーバーサンプリングされた結果が4bitシフトされた場合、8bitの右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8bitに対してだけです。比較は、ADC_DR[11:4] と HT[7:0]/LT[7:0] との間で行われます。HT[11:8]/LT[11:8] はリセット状態にしておく必要があります。**

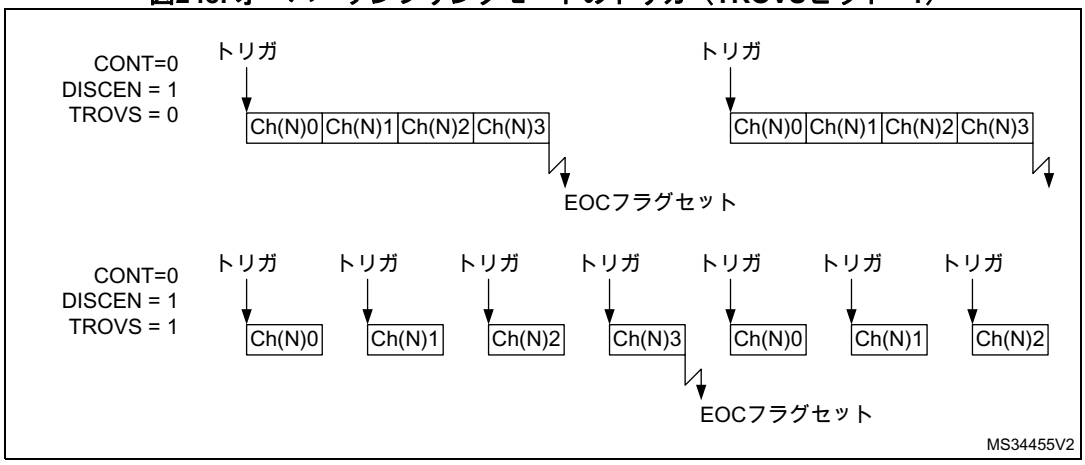
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり強力なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、DC_CFGR2のTROVSビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

下記の図248 に、不連続モード中にトリガに反応して変換を開始する方法を示します。

TROVSビットがセットされている場合、DISCENビットの内容は無視され、1とみなされます。

図248. オーバーサンプリングモードのトリガ（TROVSビット =1）



オーバーサンプリング時のインジェクトおよびレギュラシーケンサ管理

オーバーサンプリングモードでは、インジェクトシーケンスとレギュラシーケンスとで動作を変えることができます。シーケンサを同時に使用する必要がある場合、それら両方を制限付きでオーバーサンプリングを有効にできます（これは、独自の累算ユニットに関連します）。

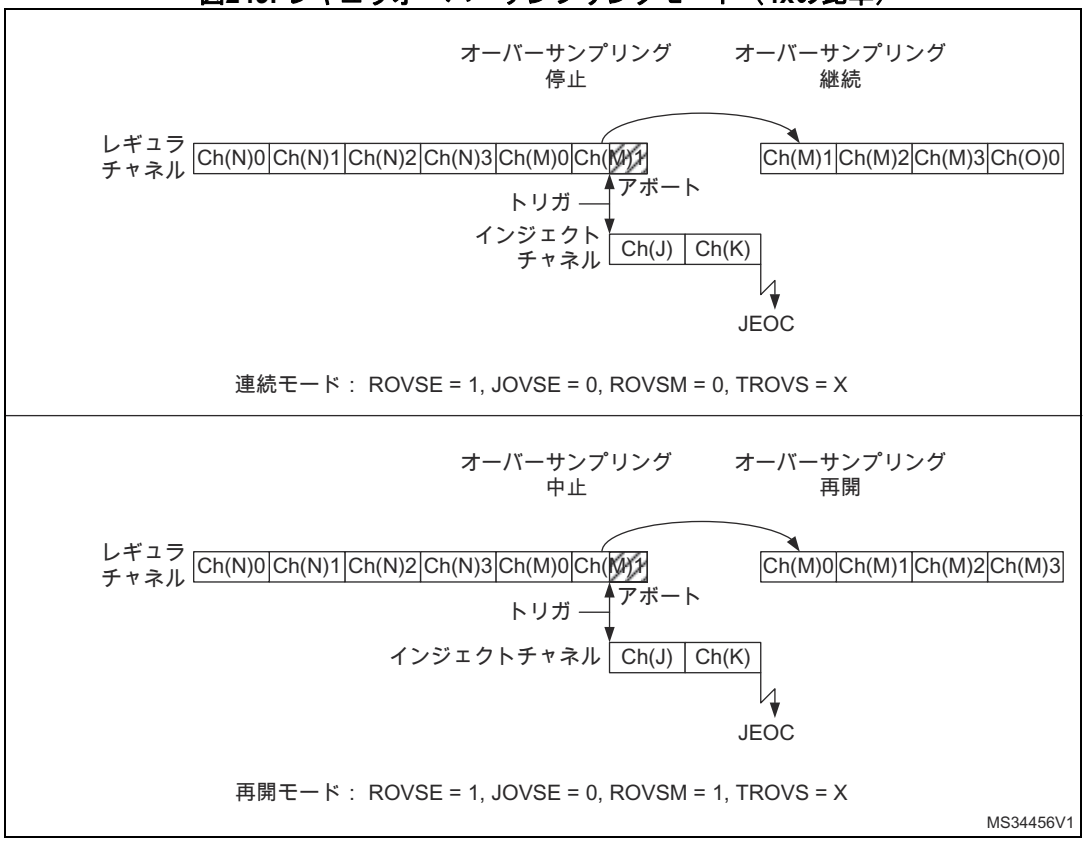
オーバーサンプリングレギュラチャネルのみ

レギュラオーバーサンプリングモードビットROVSMは、インジェクト変換によって割込みが発生した場合のレギュラサンプリングシーケンスの再開方法を定義します。

- 連続モードでは、最後の有効なデータ（インジェクトトリガによる変換中止リクエストの前）から累算を再開します。これにより、インジェクションの頻度に関係なく、オーバーサンプリングを完了できます（トリガ間で最低1回のレギュラ変換を完了できます）。
- 再開モードでは、累算が 0 から再開します（前の変換結果は無視されます）。このモードでは、オーバーサンプリングに使用されるすべてのデータが1つの時間スロット内で連続で変換されたことを保証できます。オーバーサンプリング時間を超えるインジェクショントリガ時間がかかることに注意が必要です。この条件が満たされない場合、オーバーサンプリングを完了できず、レギュラシーケンサはブロックされます。

図249 に、オーバーサンプリング比4xの例を示します。

図249. レギュラオーバーサンプリングモード（4xの比率）



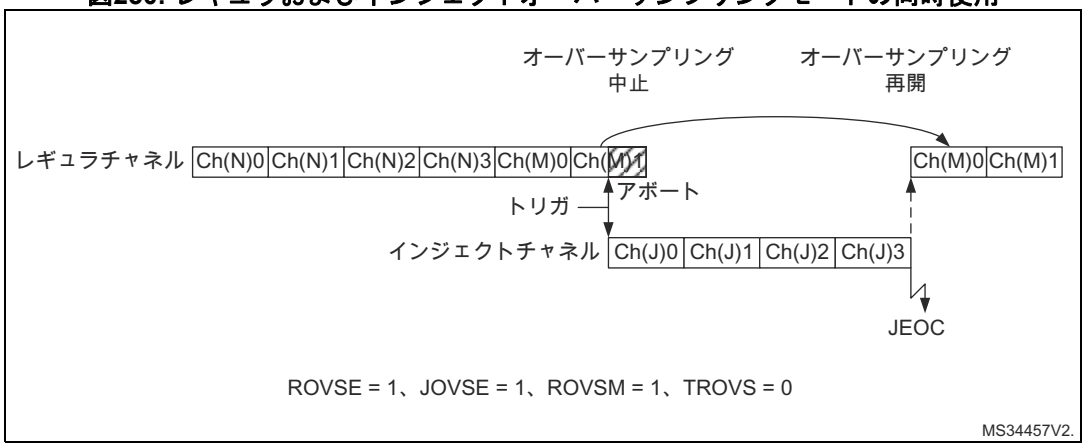
オーバーサンプリングインジェクトチャネルのみ

インジェクトオーバーサンプリングモードビットJOVSEは、インジェクトシーケンサでの変換専用のオーバーサンプリングを有効にします。

オーバーサンプリングレギュラおよびインジェクトチャネル

ROVSEビットとJOVSEビットを両方同時にセットできます。この場合、レギュラオーバーサンプリングモードは、図250に示すように、強制的に再開モードになります（ROVSMビットは無視されます）。

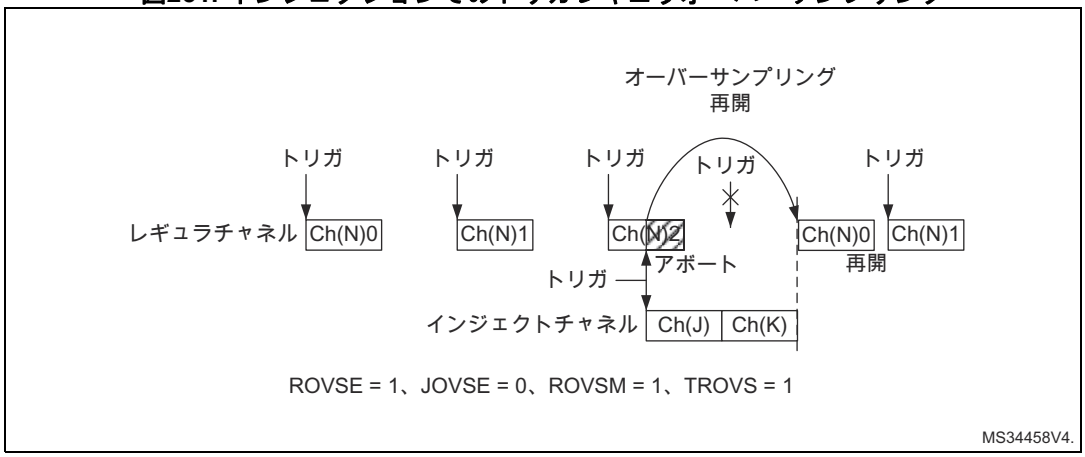
図250. レギュラおよびインジェクトオーバーサンプリングモードの同時使用



インジェクト変換でのトリガレギュラオーバーサンプリングモード

インジェクト変換は、トリガされたレギュラモードで実行できます。この場合、インジェクトモードオーバーサンプリングモードは無効になり、ROVSMビットは無視されます（再開モードが強制されます）。JOVSEビットをリセットする必要があります。動作を図251 に示します。

図251. インジェクションでのトリガレギュラオーバーサンプリング

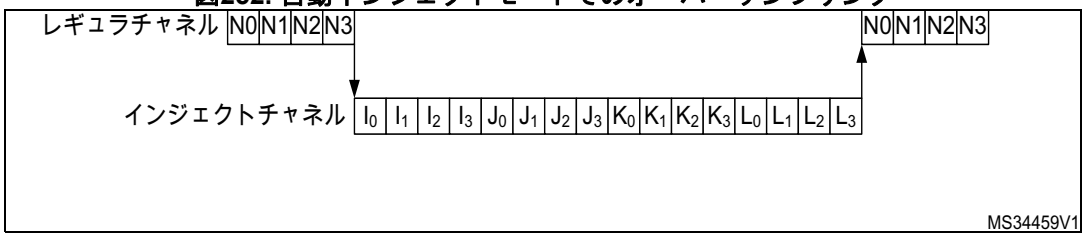


自動インジェクトモード

自動インジェクトシーケンスをオーバーサンプリングして、レジスタにすべての変換結果を格納できます。これにより、DMAのリソースを節約できます。このモードは、レギュラオーバーサンプリングとインジェクトオーバーサンプリングの両方がアクティブになっている、つまりJAUTO = 1、ROVSE = 1、かつJOVSE = 1の場合のみ利用できます。これ以外の組み合わせでは利用できません。ROVSMビットは自動インジェクトモードでは無視されます。

図252 に、変換シーケンスの方法を示します。

図252. 自動インジェクトモードでのオーバーサンプリング



TROVSビットを使用してトリガモードを有効にすることもできます。この場合、ADCの設定は、JAUTO = 1、DISCEN = 0、JDISCEN = 0、ROVSE = 1、JOVSE = 1、TROVSE = 1とする必要があります。

オーバーサンプリング時のデュアルADCモードのサポート

デュアルADC設定で動作時、インジェクト同時モードおよびレギュラ同時モードに対してオーバーサンプリングを有効にできます。この場合、2つのADCをまったく同じ設定（オーバーサンプリング含む）でプログラムする必要があります。

他のすべてのデュアルADCモードは、レギュラまたはインジェクトオーバーサンプリングが有効になっている場合（ROVSE = 1またはJOVSE = 1）サポートされません。

組み合わせモードの概要

表314 に、サポートされていないモードも含めて、すべてのモードの組み合わせの概要を示します。

表314.オーバーサンプリング回路の動作モードの概要

レギュラオーバーサンプリング：ROVSEビット	インジェクトオーバーサンプリング：JOVSEビット	オーバーサンプリング回路モード：ROVSMビット 0 = 連続 1 = 再開	トリガレギュラモード：TROVSビット	コメント
1	0	0（続き）	0	レギュラ連続モード
1	0	0（続き）	1	サポートされません
1	0	1（再開）	0	レギュラ再開モード
1	0	1（再開）	1	トリガレギュラ再開モード
1	1	0（続き）	X	サポートされません
1	1	1（再開）	0	インジェクトおよびレギュラ再開モード
1	1	1（再開）	1	サポートされません
0	1	X	X	インジェクトオーバーサンプリング

33.4.32 デュアルADCモード

2つ以上のADCを持つデバイスでは、デュアルADCモードを使用することができます（図33.4.33を参照）。

- ADC1およびADC2は、デュアルモードで一緒に使用できます（ADC1がマスタ）。

デュアルADCモードでは、変換の開始はADC12_CCRレジスタのDUAL[4:0] ビットによって選択されたモードに応じて、ADCxマスタにより交互または同時に、ADCスレーブに対してトリガされます。

次の4つのモードを備えています。

- インジェクト同時モード
- レギュラ同時モード
- インタリーブモード
- オルタネートトリガモード

これらのモードを次のように組み合わせて使用することも可能です。

- インジェクト同時モード + レギュラ同時モード
- レギュラ同時モード + オルタネートトリガモード
- インジェクト同時モード + インタリーブモード

デュアルADCモード で（ADC12_CCRレジスタのDUAL[4:0] ビットがゼロ以外の場合）、ADC_CFGRレジスタのCONT、AUTDLY、DISCEN、DISCNUM[2:0]、JDISCEN、JAUTOビット は、マスタおよびスレーブADCの間で共有されます。スレーブADCのビットは、常にマスタADCの対応するビットと等しくなります。

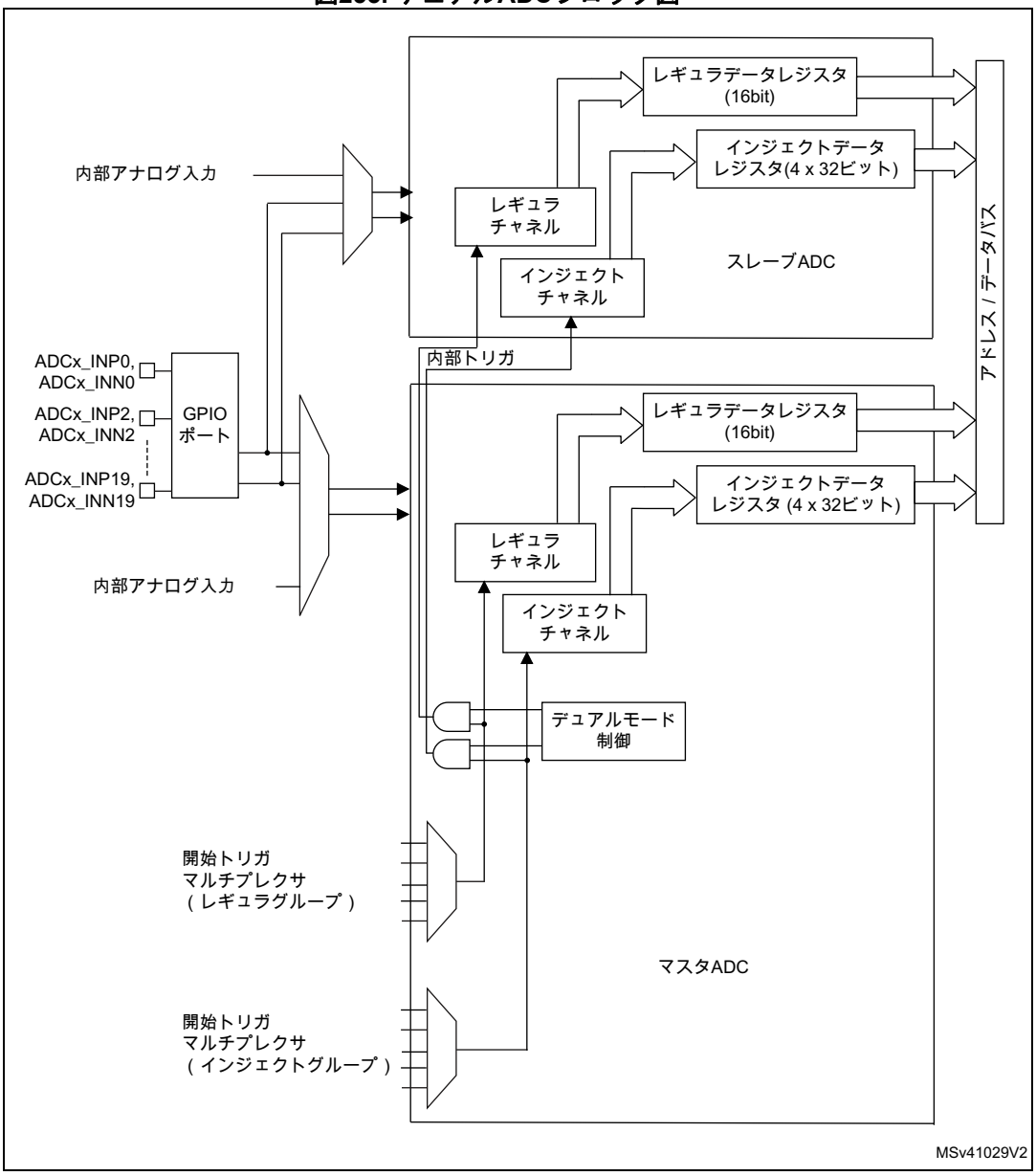
デュアル·モードで変換を開始するには、ソフトウェアまたはハードウェアのトリガやレギュラまたはインジェクトのトリガを設定するために、マスタA/Dコンバータ のEXTEN[1:0] 、EXTSEL、JEXTEN[1:0]、JEXTSELビットのみをプログラムする必要があります（スレーブADCのEXTEN[1:0] およびJEXTEN[1:0] ビットは無視されます）。

レギュラ同時モードまたはインタリーブモードでは、マスタADCのADSTARTまたはADSTPビットを一度セットすると、スレーブADCの対応するビットも自動的にセットされます。ただし、スレーブADCのADSTARTまたはADSTPビットは、必ずしもマスタADCビットと同時にクリアされません。

インジェクト同時モードまたはオルタネートトリガモードでは、マスタADCのJADSTARTまたはJADSTPビットを一度セットすると、スレーブADCの対応するビットも自動的にセットされます。ただし、スレーブADCのJADSTARTまたはJADSTPビット は、必ずしもマスタADCビットと同時にクリアされません。

デュアルADCモードでは、ADC共通データレジスタ（ADC12_CDR）を読み出すことで、マスタおよびスレーブADCの変換データを並列で読み出すことができます。ステータスビットも、デュアルモードステータスレジスタ（ADC12_CSR）を読み出すことで、並列で読み出すことができます。

図253. デュアルADCブロック図 (1)



1. スレーブADCにも外部トリガは存在しますが、この図には表示していません。
2. ADC共通データレジスタ（ADC12_CDR）にはマスタとスレーブの両方のADCレギュラ変換データが格納されます。

インジェクト同時モード

このモードは、ビットDUAL[4:0] = 00101をプログラムすることによって選択されます。

このモードは、インジェクトチャネルグループを変換します。外部トリガのソースは、マスタADCのインジェクトグループマルチプレクサ（ADC_JSQRレジスタのJEXTSEL[4:0] ビットによって選択）から供給されます。

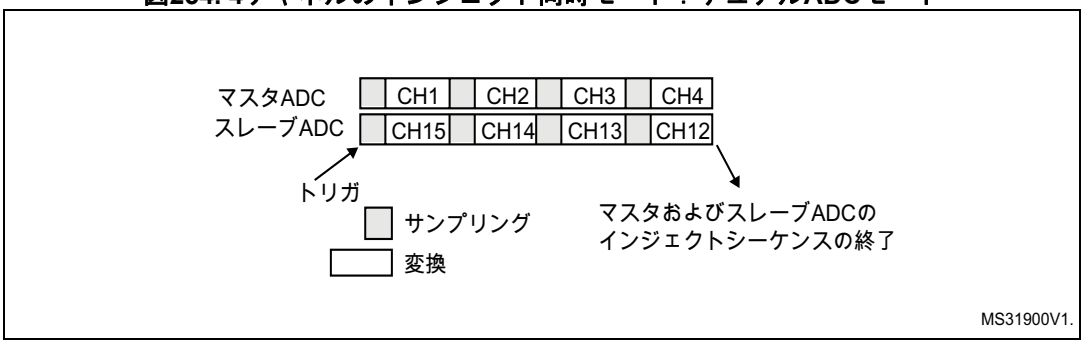
注： **2つのADCの同じチャネルを変換しないでください（同じチャネルを変換する場合、2つのADCのサンプリング時間が重なってはいけません）。**

同時モードでは、同じ長さのシーケンスを変換する必要があり、シーケンスの中では、マスタとスレーブのN番目の変換は同じサンプリング時間で設定されなければなりません。

レギュラ変換であれば、1つまたはすべてのADCで行うことができます。その場合、それらは互いに独立していてインジェクトイベントが発生すると中断されます。それら（レギュラ変換）はインジェクト変換グループ終了時に再開されます。

- マスタADCでの変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはマスタADC_JDRyレジスタに格納され、JEOS割込みが生成されます（有効な場合）。
- スレーブADCでの変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはスレーブADC_JDRyレジスタに格納され、JEOS割込みが生成されます（有効な場合）。
- マスタインジェクトシーケンスの長さがスレーブインジェクトシーケンスの長さと等しい場合（例：図254）、ソフトウェアでは2つのJEOS割込みのうちの1つのみを有効にすることができ（例：マスタJEOS）、（マスタADC_JDRyとスレーブADC_JDRyレジスタの）両方の変換データを読み出すことができます。

図254. 4チャネルのインジェクト同時モード：デュアルADCモード



JDISCEN = 1の場合、インジェクトシーケンスの各同時変換にはインジェクトトリガイベントが発生する必要があります。

このモードは、AUTDLYモードと組み合わせることができます。

- 変換の同時インジェクトシーケンスの終了後、マスタとスレーブADCの両方のJEOSビットがクリアされた場合（遅延フェーズ）にのみ、新しいインジェクトトリガイベントは受け入れられます。実行中のインジェクトシーケンスで発生したすべての新しいインジェクトトリガイベントと、これに関連する遅延フェーズは無視されます。
- マスタADCのレギュラ変換シーケンスが終了すると、マスタデータレジスタ（ADC_DR）が読み出された場合にのみ、マスタADCの新しいレギュラトリガイベントが受け入れられます。実行中のレギュラシーケンスでマスタADCに対して発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。
  スレーブADCで発生するレギュラシーケンスでも同じ挙動を示します。

独立インジェクト変換に対応したレギュラ同時モード

このモードは、ビットDUAL[4:0]=00110をプログラムすることによって選択されます。

このモードは、レギュラチャネルグループに対して行われます。外部トリガのソースは、マスタADCのレギュラグループマルチプレクサ（ADC_CFGRレジスタのEXTSEL[4:0] ビットによって選択）から供給されます。同時トリガは、スレーブADCに供給されます。

このモードでは、独立インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションリクエストにより、現在の同時変換がアボートされます。これは、インジェクト変換の完了後に再開されます。

注： **2つのADCの同じチャネルを変換しないでください（同じチャネルを変換する場合、2つのADCのサンプリング時間が重なってはいけません）。**

レギュラ同時モードでは、同じ長さのシーケンスを変換する必要があり、シーケンスの中では、マスタとスレーブのN番目の変換は同じサンプリング時間で設定されなければなりません。

データが読み出せるタイミングは、割込みによってソフトウェアに通知されます。

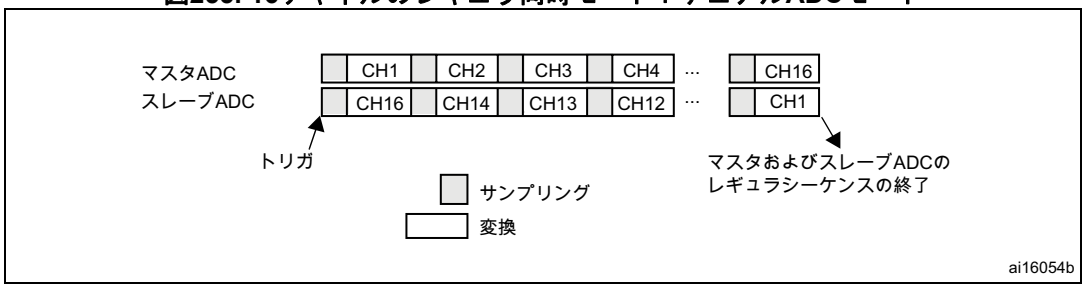
- マスタADCでの各変換イベント（EOC）の終了時、マスタEOC割込みが生成され（EOCIEが有効な場合）、ソフトウェアではマスタADCのADC_DRを読み出すことができます。
- スレーブADCでの各変換イベント（EOC）の終了時、スレーブEOC割込みが生成され（EOCIEが有効な場合）、ソフトウェアではスレーブADCのADC_DRを読み出すことができます。
- マスタレギュラシーケンスの長さがスレーブレギュラシーケンスの長さと等しい場合（例：図255）、ソフトウェアでは2つのEOC割込みのうちの1つのみを有効にすることができ（例：マスタEOC）、共通データレジスタ（ADC12_CDR）から両方の変換データを読み出すことができます。

DMAを使用してレギュラデータを読み出すことができます。次の2つの方法でできます。

- 2つのDMAチャネルを使用します（1つはマスタ用、もう1つはスレーブ用）。この場合、ビットDAMDF[1:0] をクリア状態に保つ必要があります。
  - マスタからADC_DRを読み出すために、DMAのマスタADCチャネルを設定します。DMAリクエストは、マスタADCの各EOCイベントで生成されます。
  - スレーブから ADC_DR を読み出すために、DMA のスレーブ ADC チャネルを設定します。DMAリクエストは、スレーブADCの各EOCイベントで生成されます。
- デュアルADCモードデータフォーマットのDAMDF[1:0] ビットを設定し、他の用途のために1つのDMAチャネルを残します。
  - DAMDF[1:0] = 10または11を設定します（分解能による）。
  - DMAチャネルを1つ使用します（マスタのうちの1つ）。共通ADC12レジスタ（ADC_CDR）を読み出すために、DMAのマスタADCチャネルを設定します。
  - マスタおよびスレーブの両方のEOCイベントが発生するたびにDMAリクエストが1つ生成されます。このとき、スレーブADCの変換データをADC12_CDR 32bitレジスタの上位ハーフワードで使用でき、またマスタADCの変換データをADC12_CDRレジスタの下位ハーフワードで使用できます。
  - EOCフラグは、DMAがADC12_CDRレジスタを読み出すと両方ともクリアされます。

注： **DAMDF[1:0] = 10または11の場合には、ユーザはマスタのシーケンスとスレーブのシーケンスに同じ変換数をプログラムする必要があります。そうしないと、残りの変換ではDMAリクエストは生成されません。**

図255. 16チャネルのレギュラ同時モード：デュアルADCモード



DISCEN = 1の場合、レギュラシーケンスの各「n」回の同時変換でレギュラトリガイベントが発生する必要があります（「n」はDISCNUMで定義される）。

このモードは、AUTDLYモードと組み合わせることができます。

- シーケンスの同時変換の終了後、シーケンスの次の変換は、共通データレジスタ ADC12_CDR（またはマスタADCのレギュラデータレジスタ）が読み出された場合（遅延フェーズ）にのみ開始されます。
- 変換の同時レギュラシーケンスの終了後、新しいレギュラトリガイベントは共通データレジスタ（ADC12_CDR）が読み出された場合（遅延フェーズ）にのみ受け入れられます。実行中のレギュラシーケンスで発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。

AUTDLYモードと組み合わせたレギュラ同時モードでデータを処理するために、DMAを使用することができます（このとき、マルチDMAモードが使われるものとします）。DAMDF[1:0] ビットには10または11をセットする必要があります。

レギュラ同時モードをAUTDLYモードと組み合わせる場合、ユーザは次の内容を確認する必要があります。

- マスタのシーケンスの変換数が、スレーブのシーケンス変換数と同じであること。
- シーケンスが同時変換されるたびに、スレーブADCの変換の長さがマスタADCの変換の長さよりも短いこと。シーケンスの長さは、変換するチャネルの数および各チャネルのサンプリング時間と分解能に依存する点に注意してください。

注： **このレギュラ同時モードとAUTDLYモードの組み合わせは、レギュラチャネルのみがプログラムされている場合に制限されます。この組み合わせモードでインジェクトチャネルをプログラムすることは禁止されています。**

独立インジェクト変換に対応したインタリーブモード

このモードは、ビットDUAL[4:0]=00111をプログラムすることによって選択されます。

このモードは、レギュラグループ（通常は1つのチャネル）でのみ開始できます。外部トリガソースは、マスタADCのレギュラチャネルマルチプレクスから供給されます。

外部トリガが発生した後、

- マスタADCはすぐに開始します。
- スレーブADCは、マスタADCのサンプリングフェーズが完了した後で、数ADCクロックサイクルの遅延後に開始します。

インタリーブモードにおける2つの変換間の最小遅延はADC12_CCRレジスタのDELAYビットで設定されます。遅延のカウントは、マスタ変換のサンプリングフェーズ終了後に開始します。これにより、

ADCは相補ADCが依然として入力をサンプリングしている場合、変換を開始できません（与えられた時間に1つのADCのみ入力信号をサンプリングすることができます）。

- DELAYに設定可能な最小値は1です。これにより、マスタADCサンプリングフェーズのアナログスイッチを開いてスレーブADCサンプリングフェーズのアナログスイッチを閉じるまでに、最低1つのサイクル時間を確保できます。
- 最大DELAY値は選択した分解能に対するサイクル数と同じです。ただし、ほかのADCが依然として入力をサンプリングしているときにADCが変換を開始しないよう、ユーザはこの遅延を適切に計算する必要があります。

マスタとスレーブの両方のADCでCONTビットがセットされている場合、両方のADCの選択されたレギュラチャネルが連続的に変換されます。

スレーブADCでの各変換の終了イベント（EOC）時に、データを読み出せるタイミングが、割込みによってソフトウェアに通知されます。スレーブおよびマスタEOC割込みが生成され（EOCIEが有効な場合）、ソフトウェアではスレーブ／マスタADCのADC_DRを読み出すことができます。

注： **16bitデータフォーマットでは、スレーブEOC割込みのみを有効化し、共通データレジスタ（ADC12_CDR）を読み出します。32bitデータフォーマットの場合、スレーブとマスタの両方のEOC割込みを有効化し、ADC12_CDR2レジスタを読み出します。ただしこの場合、ユーザは変換時間が適合しており、それによって新しいマスタ変換をリスタートする前に、シーケンス内では常にマスタ変換に続いてスレーブ変換が実行されることを確認する必要があります。MDMAモードを使用することを推奨します。**

レギュラデータをDMAによって転送することもできます。この場合、各ADCの個別のDMAリクエストを使用できず、次のようにDAMDFモードを使用することが必須になります。

- DAMDF[1:0] = 10または11を設定します（分解能による）。
- DMAチャネルを1つ使用します（マスタのうちの1つ）。共通ADCレジスタ（ADC12_CDR）を読み出すために、DMAのマスタADCチャネルを設定します。
- マスタおよびスレーブの両方のEOCイベントが発生するたびにDMAリクエストが1つ生成されます。このとき、スレーブADCの変換データをADC12_CDR 32bitレジスタの上位ハーフワードで使用でき、またマスタADCの変換データをADC12_CCRレジスタの下位ハーフワードで使用できます。
- EOCフラグは、DMAがADC12_CCRレジスタを読み出すと両方ともクリアされます。

図256. 連続変換モードにおける1チャネルのインタリーブモード：デュアルADCモード

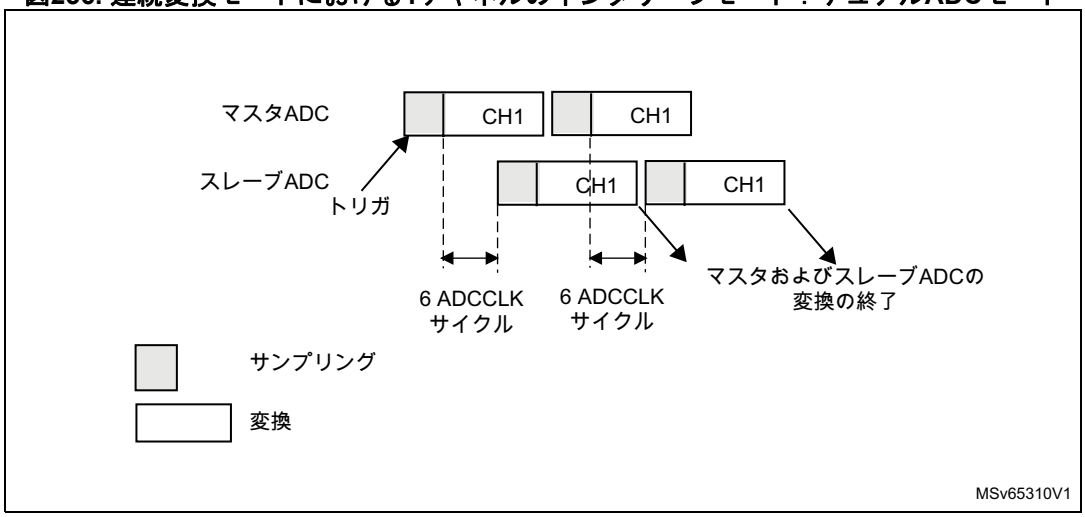
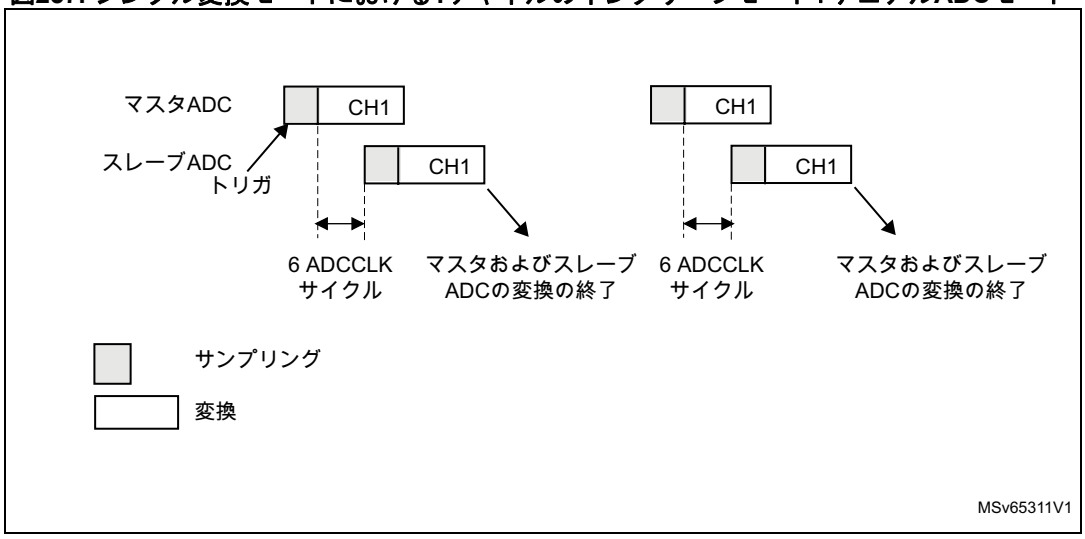


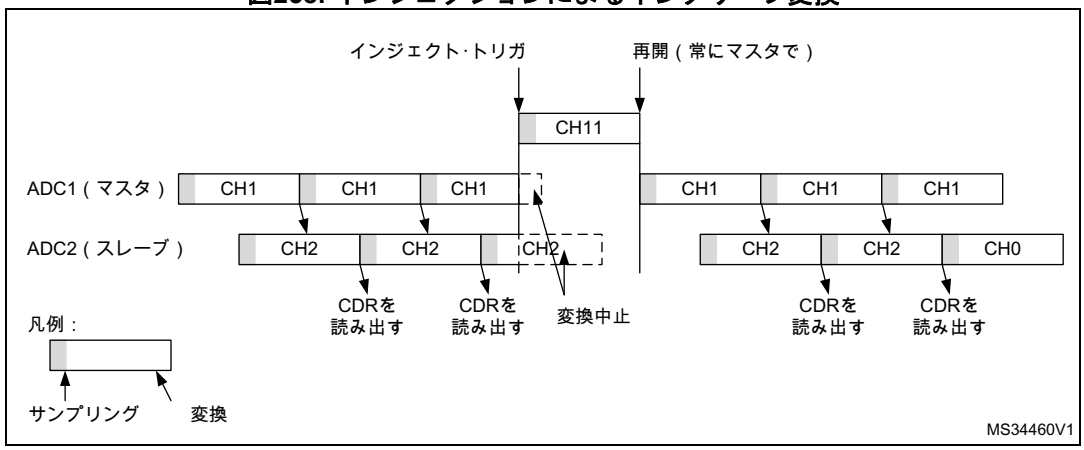
図257. シングル変換モードにおける1チャネルのインタリーブモード: デュアルADCモード



DISCEN = 1の場合、レギュラシーケンスの各「n」回の同時変換（「n」はDISCNUMで定義される）でレギュラトリガイベントが発生する必要があります。

このモードでは、インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションが完了すると、マスタとスレーブの両方のレギュラ変換がアボートされ、マスタからシーケンスが再開されます（以下の図258 を参照）。

図258. インジェクションによるインタリーブ変換



オルタネートトリガモード

このモードは、ビットDUAL[4:0]=01001をプログラムすることによって選択されます。

このモードは、インジェクトグループでのみ開始できます。外部トリガのソースは、マスタADCのインジェクトグループマルチプレクサです。

このモードは、ハードウェアトリガの選択時にのみ使用できます。JEXTEN[1:0] は0x0にしないようにする必要があります。

インジェクト不連続モードは無効です（両方のADCでJDISCEN = 0）。

1. 最初のトリガが発生すると、グループ内のすべてのインジェクトされたマスタ ADC チャネルが変換されます。
2. 2番目のトリガが発生すると、グループ内のすべてのインジェクトされたスレーブADCチャネルが変換されます。
3. 以下同様です。

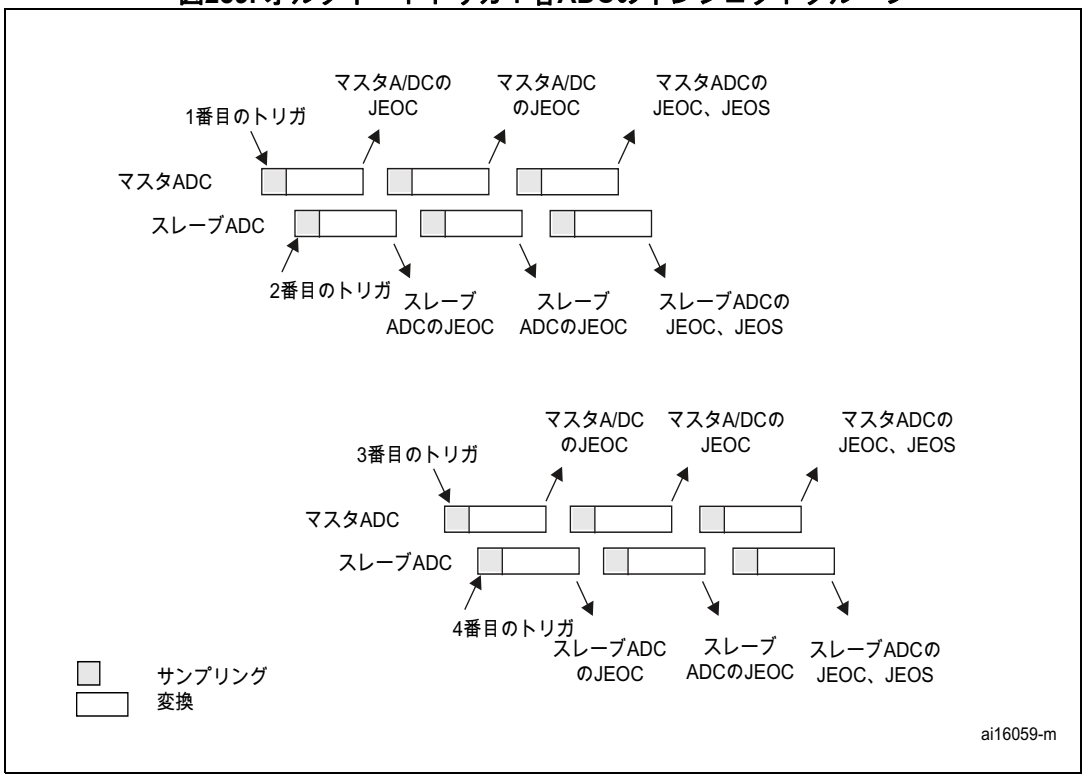
グループ内のマスタADCのすべてのインジェクトチャネルが変換されると、JEOS割込み（有効な場合）が生成されます。

グループ内のスレーブADCのすべてのインジェクトチャネルが変換されると、JEOS割込み（有効な場合）が生成されます。

JEOC割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内すべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、グループ内のマスタADCのインジェクトチャネルを変換することによって、オルタネートトリガプロセスがリスタートします。

図259. オルタネートトリガ：各ADCのインジェクトグループ



注： **レギュラ変換であれば1つまたはすべてのADCで有効にすることができます。この場合、レギュラ変換は互いに独立しています。ADCがインジェクト変換を行う必要があるとき、レギュラ変換は中断されます。インジェクト変換が終了すると(レギュラ変換が)再開されます。**

2つのトリガイベント間の間隔は1 ADCクロック期間と同等か長くする必要があります。同じADCで変換を開始する2つのトリガイベント間の最小間隔はシングルADCモードと同じです。

インジェクト不連続モードは有効です（両方のADCでJDISCEN = 1）。

マスタとスレーブの両方のADCに対してインジェクト不連続モードが有効な場合、

- 最初のトリガが発生すると、マスタADCの最初のインジェクトチャネルが変換されます。
- 2番目のトリガが発生すると、スレーブADCの最初のインジェクトチャネルが変換されます。
- 以下同様です。

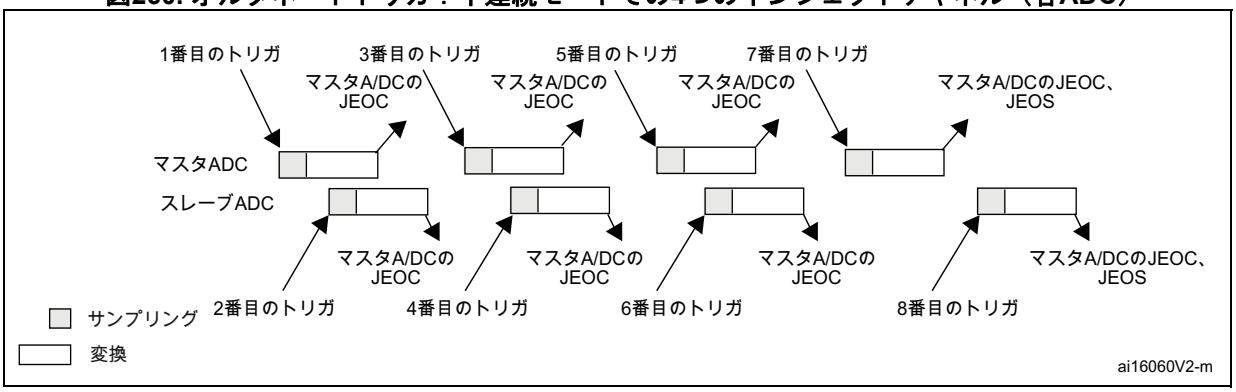
グループ内のマスタADCのすべてのインジェクトチャネルが変換されると、JEOS割込み（有効な場合）が生成されます。

グループ内のスレーブADCのすべてのインジェクトチャネルが変換されると、JEOS割込み（有効な場合）が生成されます。

JEOC割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内のすべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、オルタネートトリガプロセスがリスタートします。

図260. オルタネートトリガ：不連続モードでの4つのインジェクトチャネル（各ADC）



レギュラ／インジェクト同時モードの組み合わせ

このモードは、ビットDUAL[4:0] = 00001をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループの同時変換を開始することができます。

注： **シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が2つのシーケンスの長い方の変換時間より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスのADCが前の変換を完了する前に、短いシーケンスのADCがリスタートすることがあります。**

レギュラ同時 + オルタネートトリガモードの組み合わせ

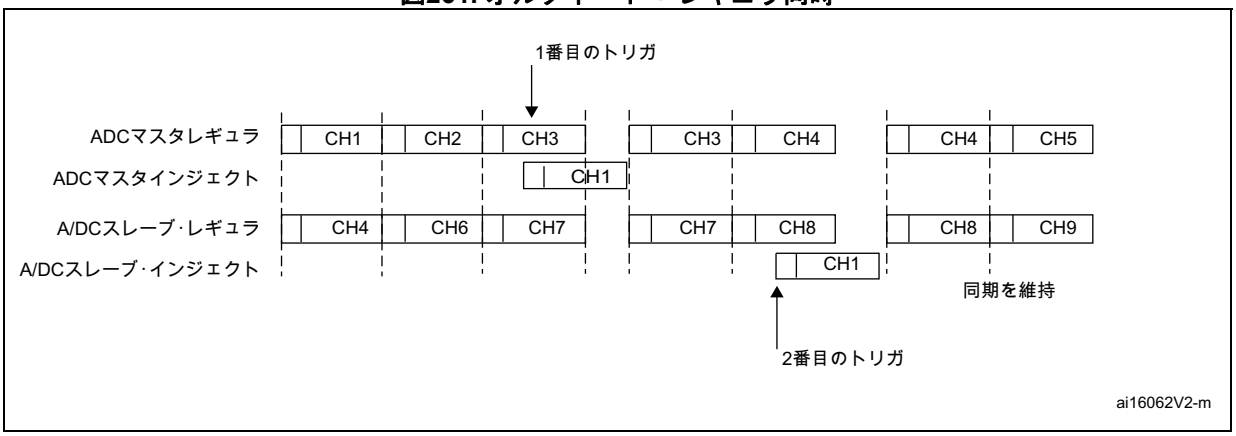
このモードは、ビットDUAL[4:0] = 00010をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループのオルタネートトリガ変換を開始することができます。図261は、同時レギュラ変換に割り込むオルタネートトリガの動作を示します。

インジェクトオルタネート変換は、インジェクトイベント後、ただちに開始されます。すでにレギュラ変換が実行中の場合、インジェクト変換後の同期を確保するために、すべての（マスタ／スレーブ）ADCのレギュラ変換は停止し、インジェクト変換の終了と同期して再開されます。

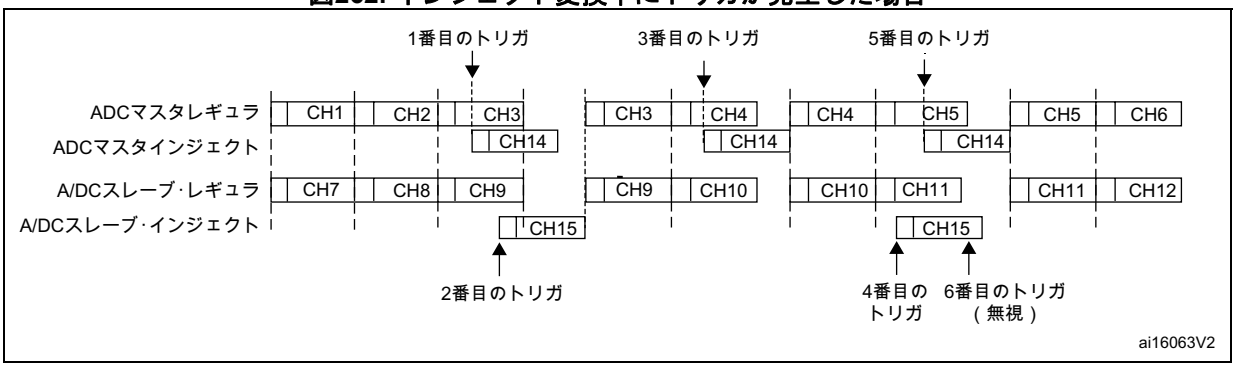
注： **シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が2つのシーケンスの長い方の変換時間より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスのADCが前の変換を完了する前に、短いシーケンスのADCがリスタートすることがあります。**

図261. オルタネート + レギュラ同時



レギュラ変換に割り込んだインジェクト変換の最中にトリガが発生した場合はオルタネートトリガが処理されます。図262に、この場合の挙動を示します（関連付けられたオルタネート変換は完了していないため、6番目のトリガは無視されます）。

図262. インジェクト変換中にトリガが発生した場合



インジェクト同時モードとインタリーブの組み合わせ

このモードは、ビットDUAL[4:0] = 00011としてプログラムすることによって選択されます。

インタリーブ変換に同時インジェクトイベントでの割込みが可能です。

この場合、インタリーブ変換は即座に割り込まれ、同時インジェクト変換が開始します。インジェクトシーケンス終了時に、インタリーブ変換が再開されます。インタリーブレギュラ変換が再開されるとき、実行される最初のレギュラ変換は常にマスタのものです。図263、図264 および図265 には、例を使用して動作を示しています。

注意： **このモードでは、1 回の読出しアクセスでレギュラデータを読み出すために共通データレジスタを使用する必要があります。逆に、マスタ - スレーブのデータコヒーレンシは保証されません。**

図263. インジェクトシーケンスCH11、CH12とインタリーブシングルチャネルCH 0

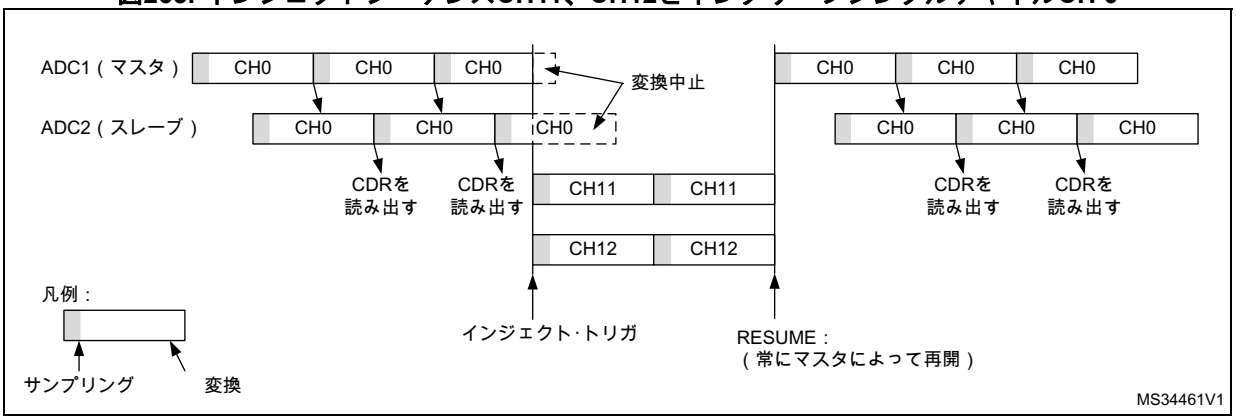


図264. インジェクトシーケンスCH11、CH12と2つのインタリーブチャネル（CH1、CH2）- ケース1：マスタ割込み優先

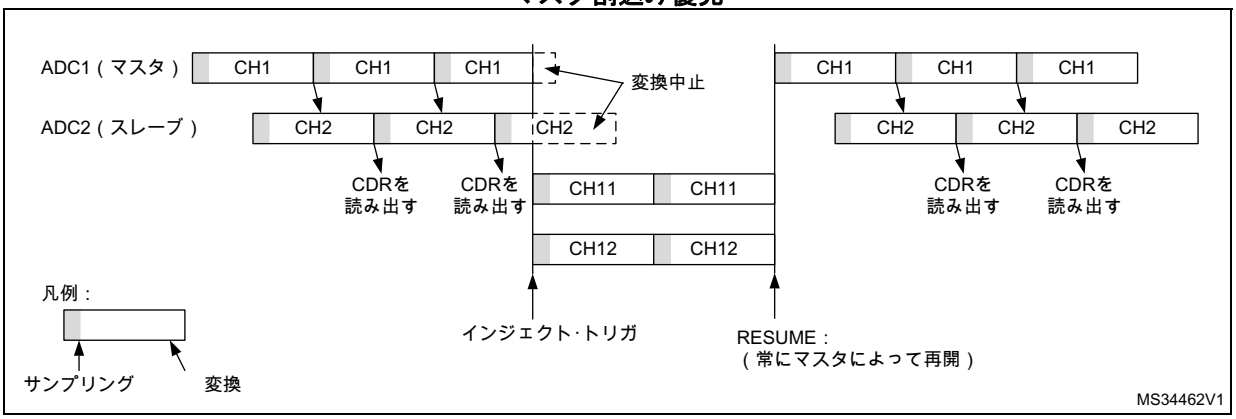
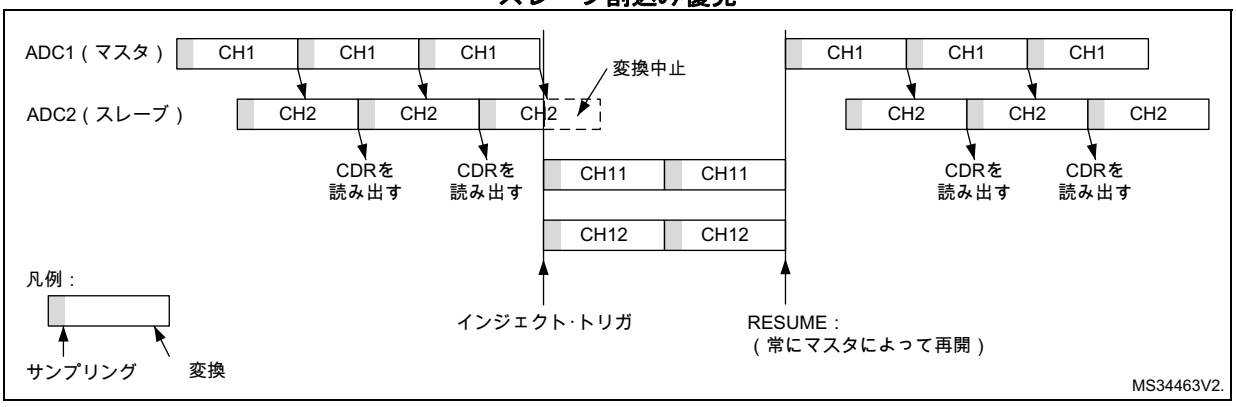


図265. インジェクトシーケンスCH11、CH12と2つのインタリーブチャネル（CH1、CH2）- ケース2：スレーブ割込み優先

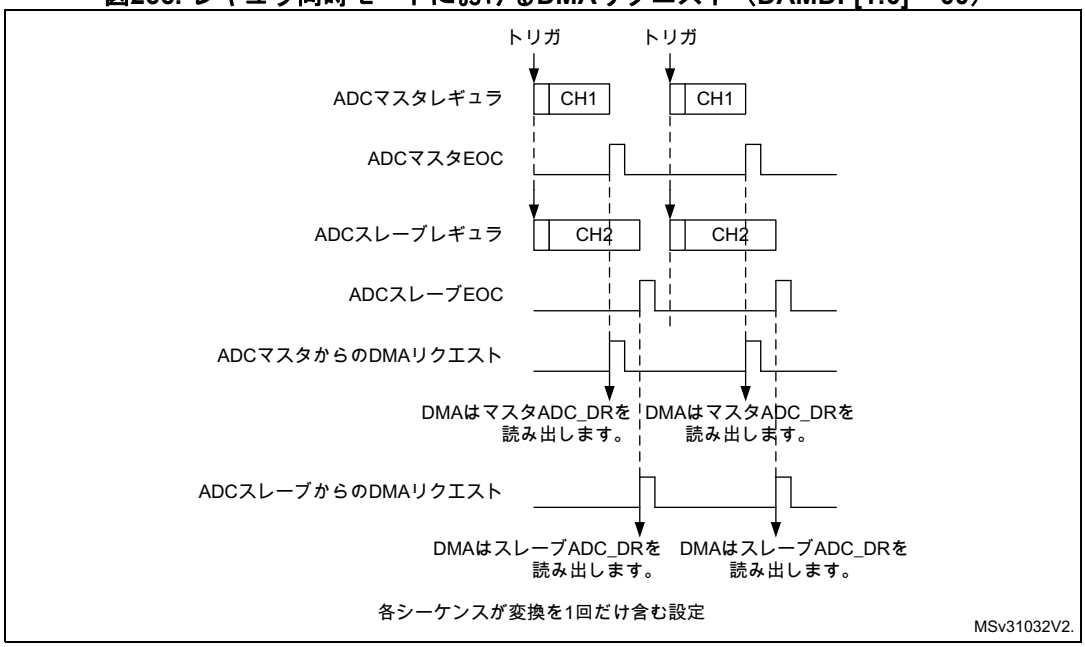


デュアルADCモードにおけるDMAリクエスト

すべてのデュアルADCモードでは、シングルモードの場合と同様、データを転送するために2つのDMAチャネル（マスタ用1つとスレーブ用1つ）を使用することができます（図 266：レギュラ同時

モードにおけるDMAリクエスト（DAMDF[1:0] = 00）を参照）。

図266. レギュラ同時モードにおけるDMAリクエスト（DAMDF[1:0] = 00）



レギュラ同時モードおよびインタリーブモードでは、1つのDMAチャネルを確保し、シングルDMAチャネルを使用して両方のデータを転送することもできます。そのためには、ADC12_CCRレジスタでDAMDF [1:0] ビットを設定する必要があります。

- **DAMDF[1:0] = 10、32bitフォーマット**：マスタとスレーブいずれかのEOCイベントが発生するたびに、それに応じてDMAリクエストが1つ生成されます。そのとき、データ項目は代わりにADC12_CDR2 32bitレジスタにあります。このモードは、分解能が16bitを超えている場合に、インタリーブモードおよびレギュラ同時モードで使用されます。**例**：

  インタリーブデュアルモード：新しい32bitデータが使用可能になるたびにDMAリクエストが1つ生成されます。

  最初のDMAリクエスト：ADC12_CDR2[31:0] = MST_ADC_DR[31:0]

  2番目のDMAリクエスト：ADC12_CDR2[31:0] = SLV_ADC_DR[31:0]

- **DAMDF[1:0] = 10、16bitフォーマット**：マスタおよびスレーブの両方のEOCイベントが発生するたびにDMAリクエストが1つ生成されます。このとき、2つのデータアイテムが使用可能です。また、32bitレジスタADC12_CDRには、2つの ADCで変換されたデータアイテムを表す2つのハーフワードが含まれます。スレーブADCデータは上位ハーフワードを使用し、マスタADCデータは下位ハーフワードを使用します。

  このモードは、分解能が10bitから16bitの範囲の場合に、インタリーブモードおよびレギュラ同時モードで使用されます。マスタコンバータまたはスレーブコンバータの16bitを超えるすべての値は、最下位16bitに切り捨てられます。

  例：

  インタリーブデュアルモード：2つのデータ項目が使用可能になるたびにDMAリクエストが1つ生成されます。

  最初のDMAリクエスト：ADC12_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

  2番目のDMAリクエスト：ADC12_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

図267. レギュラ同時モードにおけるDMAリクエスト（DAMDF[1:0] = 10）

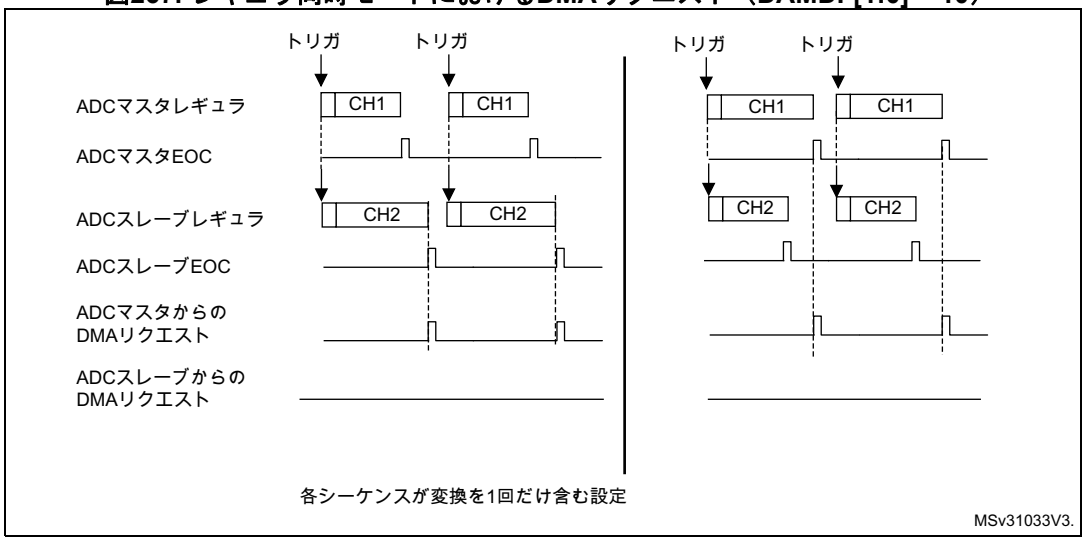
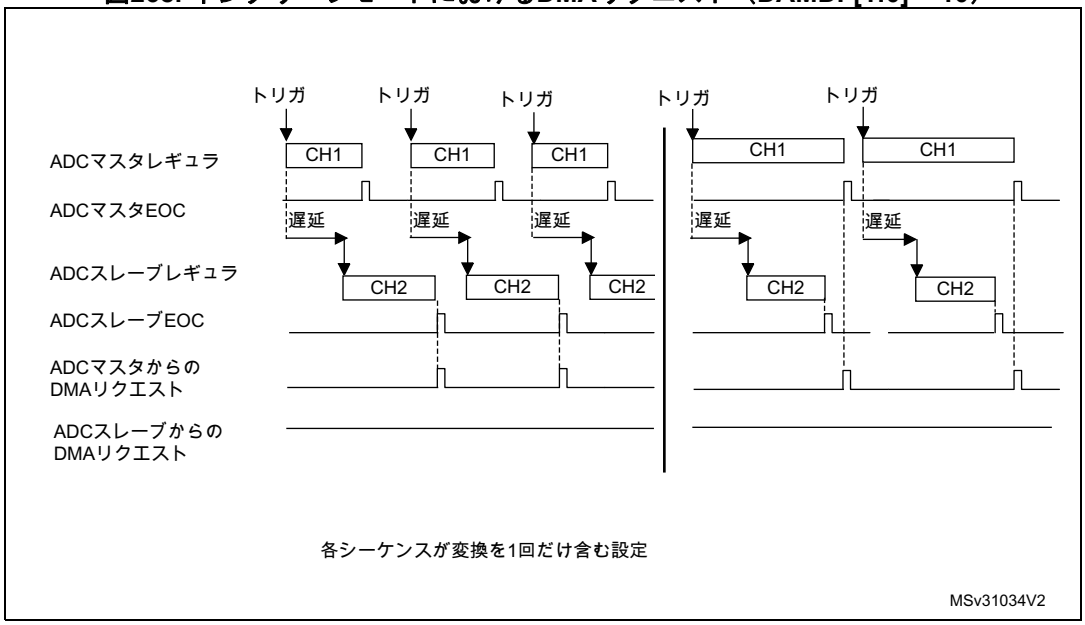


図268. インタリーブモードにおけるDMAリクエスト（DAMDF[1:0] = 10）



注： **マルチADCモードを使用する場合は、マスタおよびスレーブ変換時間を適切に設定するよう注意する必要があります。これにより、新しい変換が使用可能になる前にDMAリクエストを生成して両方のデータ（マスタ + スレーブ）の読出しを処理できます。**

- **DAMDF[1:0] = 11**：このモードはDAMDF[1:0] = 10と同様です。唯一の違いは、DMAリクエストごとに（データ項目は2つ使用可能）、2つのADC変換データを表す2バイトがハーフワードとして転送されます。

  このモードは、結果が8bitである場合に、インタリーブモードおよびレギュラ同時モードで使用されます。新しい8bit値が4つある場合に、DMAリクエストが1つ発行されます。

  例：

  インタリーブデュアルモード：2つのデータ項目が使用可能になるたびにDMAリクエストが1つ生成されます。

  DMAリクエスト：

  ADC12_CDR[7:0] = MST_ADC_DR[7:0]

  ADC12_CDR[15:8] = SLV_ADC_DR[7:0]

  ADC12_CDR[31:16] = 0x0

オーバーラン検出

デュアルADCモード（DUAL[4:0] が0b00000以外の場合）で、ADCのうちの1つにオーバーランが検出された場合、RAMに転送されたすべてのデータが有効であることを保証するためDMAリクエストの発行を停止します（この動作はDAMDFの設定にかかわらず発生します）。あるADCのデータレジスタに有効なデータが格納されていることによりそのADCに対応するEOCビットがセットされたままになることがあります。

マルチADCモード選択時のDMAワンショットモード／DMAサーキュラモード

DAMDFモードが選択されている場合（10または11）、で説明しているように、DMAワンショットモード／サーキュラモードのどちらかを選択するために、マスタADC ADC12_CCRレジスタのビットDMNGT[1:0] = 10も設定する必要がありますセクション ：DMAを使用した変換の管理。

デュアルADCモードにおける変換の停止

デュアルADCモードにおけるADCの変換を両方とも停止するには、マスタADCの制御ビットにADSTP/JADSTPをセットする必要があります。デュアルADCモードには、スレーブADCのその他のADSTP制御ビットの効果はありません。

両方のADCが実質的に無効になると、マスタおよびスレーブADCの両方のADSTART/JADSTARTビットがハードウェアによってクリアされます。

デュアルADCモードインタリーブモードにおけるMDFモード

デュアルADCインタリーブモードでは、ADC変換結果は、多機能デジタルフィルタ（MDF）に直接転送できます。

このモードは、マスタADCのADC_CFGRレジスタのビットDMNGT[1:0] = 10をセットすることによって有効になります。

ADCは、マスタおよびスレーブコンバータからMDFのシングルチャネルにレギュレータデータの最下位16bitを代わりに転送します。

データフォーマットは、16bit符号である必要があります。

ADC_DR[31:16] = 0x0000

ADC_DR[15] = 符号

ADC_DR[14:0] = データ

どのコンバータでも、16bit符号付きフォーマットを超えるあらゆるデータは切り捨てられます。

デュアルADC同時モードにおけるMDFモード

変換データは個別チャネルによって扱われるため、デュアルADC同時モードでMDFを使用する時には、DFSDMのデュアルモードは必要ありません。同じトリガソースでのシングルモードは、MDFインタフェースでの同時変換になります。

33.4.33 温度センサ

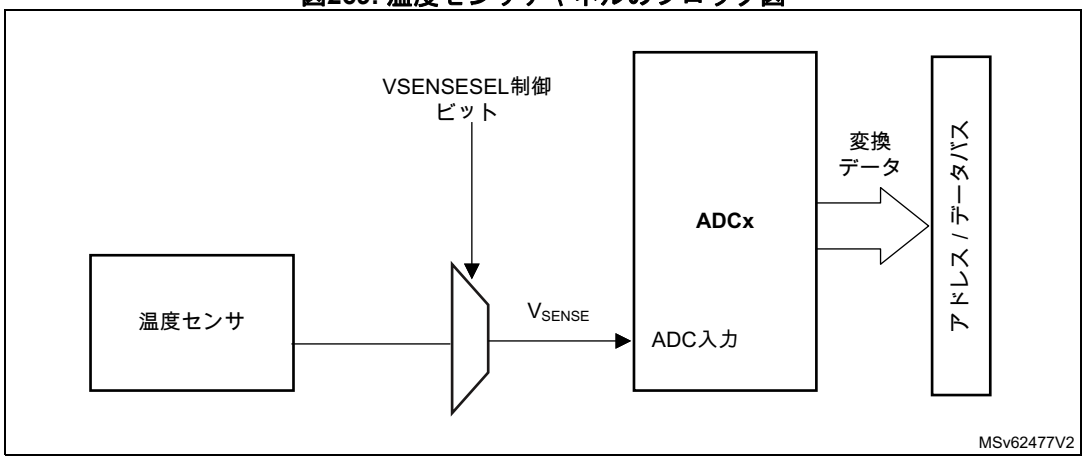
温度センサは、温度範囲が -40 ~ 125°Cのデバイスの接合温度（T_J）を測定できます。

温度センサは、センサの出力電圧をデジタル値に変換するためにADC入力チャネルに内部接続されます（詳細はセクション 33.4.4：ADC接続性を参照）。温度センサのアナログピンのサンプリング時間は、デバイスのデータシートに記載された整定時間より大きくなければなりません。

使用しないときには、センサをパワーダウンモードにすることができます。

図269 に、温度センサのブロック図を示します。

図269. 温度センサチャネルのブロック図



温度の読出し

温度センサを使用するには：

1. 温度センサを接続する入力チャネルを選択します（適切なサンプリング時間が必要）。
2. 適切なサンプリング時間をプログラムします（デバイスデータシートの電気的特性のセクションを参照）。
3. ADC12_CCRレジスタのVSENSESELビットをセットして、温度センサをパワーダウンモードからウェイクアップします。
4. ADC変換を開始します。
5. ADCデータレジスタに格納されたデータを読み出します。

6. 次の式を使用して、実際の温度を計算します。

$$\text{Temperature (in °C)} = \frac{\text{TS_CAL2_TEMP} - \text{TS_CAL1_TEMP}}{\text{TS_CAL2} - \text{TS_CAL1}} \times (\text{TS_DATA} - \text{TS_CAL1}) + \text{TS_CAL1_TEMP}$$

ここで、

– TS_CAL2は、TS_CAL2_TEMPで得られた温度センサの較正値です。

– TS_CAL1は、TS_CAL1_TEMPで得られた温度センサの較正値です。

– TS_DATAは、ADCによって変換された実際の温度センサの出力値です。

TS_CAL1およびTS_CAL2較正ポイントについての詳細は、セクション 33.3 : A/Dコンバータの実装を参照してください。

注： **センサがパワーダウンモードからウェイクアップして、正しいレベルで出力できるようになるまでには時間がかかります（スタートアップ時間）。ADCでは、パワーオン後にもスタートアップ時間が必要です。そのため、遅延を最小限に抑えるには、ADENビットとVSENSESELビットを同時にセットする必要があります。**

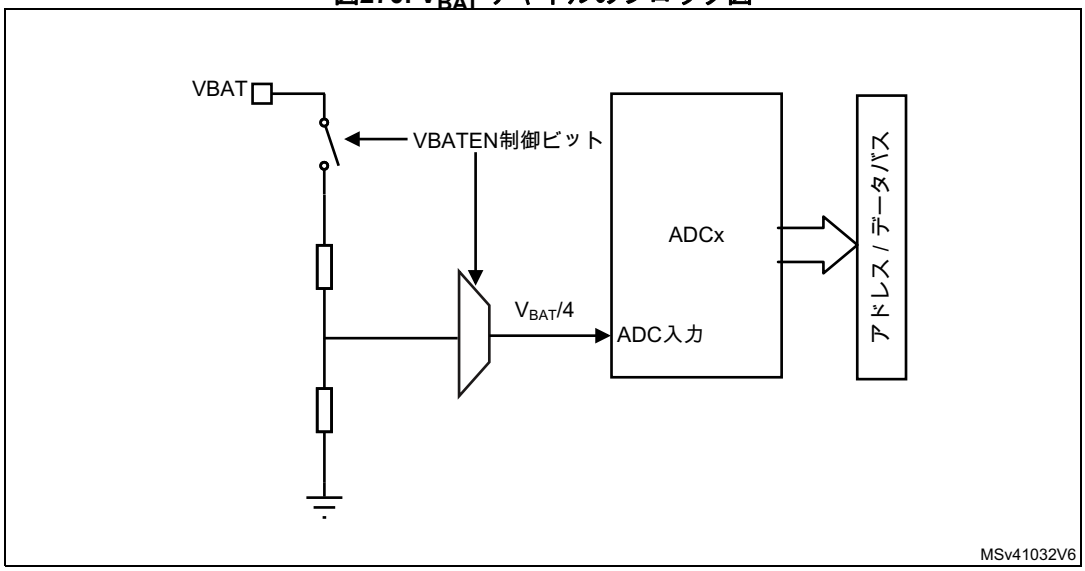
33.4.34 V_{BAT} 電源監視

ADC12_CCRレジスタのVBATENビットはバッテリ電圧に切り替えるために使用します。V_{BAT} 電圧は V_{DDA} より高くなることがあるので、ADCの正しい動作を確保するために、VBATピンはブリッジ4分圧回路に内部接続されています。このブリッジは、VBATENがセットされると自動的に有効になり、V_{BAT}/4を対応するADC入力チャネルに接続します（詳細はセクション 33.4.4 : ADC接続性を参照）。結果として、変換されたデジタル値はV_{BAT} 電圧の1/4です。不要なバッテリ消費を避けるには、ADC変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

V_{BAT} /4電圧の変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性を参照してください。

図270 に、V_{BAT} センシング機能のブロック図を示します。

図270. V_{BAT} チャネルのブロック図



注： **VBAT が接続される ADC 内部チャネルの変換を有効化するには、ADC12_CCR レジスタの VBATEN ビットをセットする必要があります（詳細はセクション 33.4.4：ADC接続性を参照）。**

33.4.35 内部電圧基準の監視

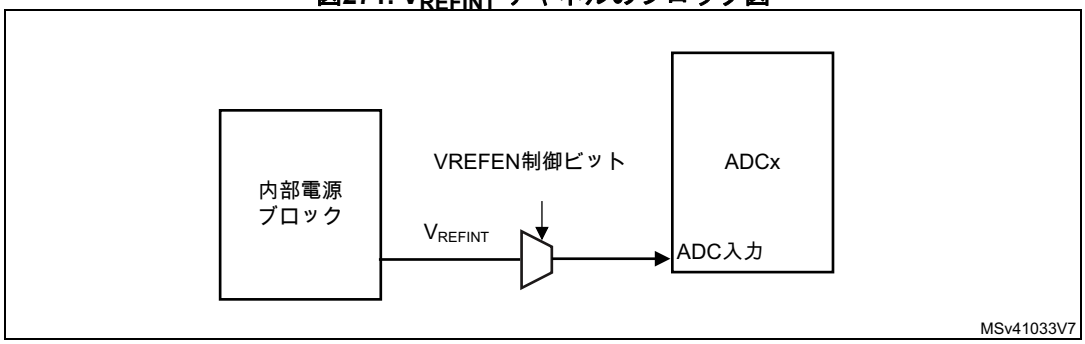
ADC V_{REF+} 電圧レベルを評価するための基準点を持つために、内部電圧基準を監視することができます。

内部電圧基準が内部的に接続されているADC入力チャネルの詳細については、セクション 33.4.4：ADC接続性を参照してください。

このチャネルのサンプリング時間は、デバイスのデータシートに記載された安定時間より大きくなければなりません。

図270 に、V_{REFINT} センシング機能のブロック図を示します。

図271. V_{REFINT} チャネルのブロック図



注： **VREFINT が接続される ADC 内部チャネルの変換を有効化するには、ADC12_CCR レジスタの VREFEN ビットをセットする必要があります（詳細はセクション 33.4.4：ADC接続性を参照）。**

内部基準電圧を使用した実際のVREF+ 電圧の計算

マイクロコントローラに印加されるV_{DDA} 電源電圧は、ばらつきがあり、または正確にはわからないことがあります。VREF+ ピンが可変V_{DDA} 電源に接続されている場合、内蔵の内部電圧基準（V_{REFINT}）と、製造プロセス時にV_{REF} = 3.0VでADCによって得られた較正データを使用して、実際のV_{REF+} 電圧レベルを評価することができます。

デバイスに印加される実際のV_{REF+} 電圧は、次の式で求められます。

$$V_{REF+} = 3.0\ V \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- VREFINT_CALはVREFINT較正値です（VREFINT_CALの値については、セクション 33.3 : A/Dコンバータの実装を参照）。
- VREFINT_DATAは、ADCによって変換された実際のVREFINTの出力値です。

電源に相対的なADC測定値から絶対電圧値への変換

ADCは、電圧基準V_{REF+}と、変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。ほとんどのアプリケーションの使用事例では、この比をV_{DDA}に依存しない電圧に変換する必要があります。V_{DDA} が既知であり、ADCによって変換された値が右詰めされるアプリケーションでは、次の式を使用して、この絶対値を求めることができます。

$$V_{CHANNELx} = \frac{V_{REF+}}{NUM_CODENUM_CODES} \times ADC_DATA$$

V_{DDA} 値が既知ではないアプリケーションの場合、内部電圧基準とV_{DDA} を、セクション ：内部基準電圧を使用した実際のVREF+ 電圧の計算に記載されている式で置き換えることができ、次の式が得られます。

$$V_{CHANNELx} = \frac{3.0\ V \times VREFINT_CAL \times ADC_DATA}{VREFINT_DATA \times NUM_CODES}$$

ここで、

VREFINT_CALはVREFINT較正値です（VREFINT_CALの値については、セクション 33.3 : A/Dコンバータの実装を参照）。

ADC_DATAは、チャネルxでADCによって測定された値です（右詰め）。

VREFINT_DATAは、ADCによって変換された実際のVREFINTの出力値です。

NUM_CODESはADC出力コードの数です。たとえば、14bit分解能では、2^{14} = 16384になり、8bit分解能では2^8 = 256になります。

注： **ADC測定が14bit右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。**

33.5 ADC割込み

各ADCに、割込みを生成することができます。

- ADCパワーアップ後、ADCの準備ができたとき（ADRDYフラグ）
- レギュラグループの任意の変換終了時（EOCフラグ）
- レギュラグループの任意の変換シーケンス終了時（EOSフラグ）
- インジェクトグループの任意の変換終了時（JEOCフラグ）
- インジェクトグループの任意の変換シーケンス終了時（JEOSフラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、およびAWD3フラグ）
- サンプリングフェーズ終了の発生時（EOSMPフラグ）
- データオーバーランの発生時（OVRフラグ）

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表315.ADC割込み

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOP 0、STOP 1、およびSTOP 2モードの終了	STOP 3およびSTANDBYモードの終了
ADC	ADCレディ	ADRDY	ADRDYIE	ハードウェアによってセットされ、ソフトウェアによってクリアされます。	あり	なし	なし
	レギュラグループの変換終了	EOC	EOCIE				
	レギュラグループの変換シーケンスの終了	EOS	EOSIE				
	インジェクトグループの変換終了	JEOC	JEOCIE				
	インジェクトグループの変換シーケンスの終了	JEOS	JEOSIE				
	アナログウォッチドッグ1のステータスビットのセット	AWD1	AWD1IE				
	アナログウォッチドッグ2のステータスビットのセット	AWD2	AWD2IE				
	アナログウォッチドッグ3のステータスビットのセット	AWD3	AWD3IE				
	サンプリングフェーズの終了	EOSMP	EOSMPIE				
	オーバーラン	OVR	OVRIE				

33.6 ADCレジスタ（ADCごと）

レジスタの説明で使用されている略語のリストについては、セクション1.2 を参照してください。

33.6.1 ADC割込みおよびステータスレジスタ（ADC_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LDO RDY	Res.	Res.	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
			r			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LDORDY**：ADC電圧レギュレータのレディ状態

このビットは、ハードウェアによってセットされます。ADC内部電源がレディ状態であることを示します。ADCは$t_{ADCVREG_SETUP}$ 時間の経過後に利用可能になります。

0：ADC電圧レギュレータは無効です。

1：ADC電圧レギュレータは有効です。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 AWD3：アナログウォッチドッグ3フラグ

このビットは、変換された電圧が、ADC_LTR3およびADC_HTR3レジスタのフィールドLT3[7:0] およびHT3[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：アナログウォッチドッグ3イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ3イベントが発生しました。

ビット8 **AWD2**：アナログウォッチドッグ2フラグ

このビットは、変換された電圧が、ADC_LTR2およびADC_HTR2レジスタのフィールドLT2[7:0] およびHT2[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：アナログウォッチドッグ2イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ2イベントが発生しました。

ビット7 **AWD1**：アナログウォッチドッグ1フラグ

このビットは、変換された電圧が、ADC_LTR1およびADC_HTR1レジスタのフィールドLT1[11:0] およびHT1[11:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。1を書き込むことにより、ソフトウェアでクリアされます。

0：アナログウォッチドッグ1イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ1イベントが発生しました。

ビット6 **JEOS**：インジェクトチャネルのシーケンス終了フラグ

このビットは、グループ内のすべてのインジェクトチャネル変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：インジェクト変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクト変換シーケンスが完了しました。

ビット5 **JEOC**：インジェクトチャネルの変換終了フラグ

このビットは、チャネルの各インジェクト変換の終了時に、新しいデータが対応するADC_JDRyレジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって1を書き込むことによって、または対応するADC_JDRyレジスタを読み出すことによってクリアされます。

0：インジェクトチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトチャネル変換が完了しました。

ビット4 **OVR**：ADCオーバーラン

このビットは、レギュラチャネルでオーバーランが発生したときにハードウェアによってセットされ、EOCフラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで1を書き込むことによってクリアされます。

0：オーバーランは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット3 **EOS**：レギュラシーケンス終了フラグ

このビットは、レギュラチャネルシーケンスの変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：レギュラ変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラ変換シーケンスが完了しました。

ビット2 **EOC**：変換終了フラグ

このビットは、チャネルの各レギュラ変換の終了時に、新しいデータがADC_DRレジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって1を書き込むことによって、またはADC_DRレジスタを読み出すことによってクリアされます。

0：レギュラチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラチャネル変換が完了しました。

ビット1 **EOSMP**：サンプリング終了フラグ

このビットは、任意のチャネル（レギュラチャネルのみ）の変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズの終了に達しました。

ビット0 **ADRDY**：ADCのレディ状態

このビットは、A/Dコンバータが有効にされた後（ビットADEN = 1）、A/Dコンバータが変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで1を書き込むことによってクリアされます。

0：ADCはまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：ADCは変換を開始する準備ができました。

33.6.2 ADC割込み有効レジスタ（ADC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **AWD3IE**：アナログウォッチドッグ3割込みの有効化

このビットは、アナログウォッチドッグ2割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ3割込みは無効です。

1：アナログウォッチドッグ3割込みは有効です。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット8 **AWD2IE**：アナログウォッチドッグ2割込みの有効化

このビットは、アナログウォッチドッグ2割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ2割込みは無効です。

1：アナログウォッチドッグ2割込みは有効です。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット7 **AWD1IE**：アナログウォッチドッグ1割込みの有効化

このビットは、アナログウォッチドッグ1割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ1割込みは無効です。

1：アナログウォッチドッグ1割込みは有効です。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット6 **JEOSIE**：インジェクト変換シーケンス終了割込みの有効化

このビットは、インジェクト変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOS割込みは無効です。

1：JEOS割込みは有効です。JEOSビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット5 **JEOCIE**：インジェクト変換終了割込みの有効化

このビットは、インジェクト変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOC割込みは無効です。

1：JEOC割込みは有効です。JEOCビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART = 0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット4 **OVRIE**：オーバーラン割込みの有効化

このビットは、レギュラ変換のオーバーラン割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割込みは無効です。

1：オーバーラン割込みは有効です。OVRビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット3 **EOSIE**：レギュラ変換シーケンス終了割込みの有効化

このビットは、レギュラ変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS割込みは無効です。

1：EOS割込みは有効です。EOSビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット2 **EOCIE**：レギュラ変換終了割込みの有効化

このビットは、レギュラ変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC割込みは無効です。

1：EOC割込みは有効です。EOCビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット1 **EOSMPIE**：レギュラ変換用のサンプリング終了フラグ割込みの有効化

このビットは、レギュラ変換のサンプリングフェーズ終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP割込みは無効です。

1：EOSMP割込みは有効です。EOSMPビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット0 **ADRDYIE**：ADCレディ割込みの有効化

このビットは、ADCレディ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY割込みは無効です。

1：ADRDY割込みは有効です。ADRDYビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

33.6.3 ADC制御レジスタ（ADC_CR）

アドレスオフセット：0x08

リセット値：0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	Res.	DEEPPWD	ADVREGEN	CALINDEX[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADCALLIN
rs		rw	rw	rw	rw	rw	rw								rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
										rs	rs	rs	rs	rs	rs

ビット31 **ADCAL**：ADCの較正

このビットは、ADCの較正を開始するために、ソフトウェアによってセットされます。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：1を書き込むと、ADCが較正されます。1が読み出された場合、較正の実行中であることを意味します。

注： ソフトウェアは、ADEN = 0のときだけADCALをセットして、較正を起動することができます。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 **DEEPPWD**：ディープパワーダウンの有効化

このビットは、ADCをディープ･パワー･ダウン･モードにするために、ソフトウェアによってセット／クリアされます。

0：ADCがディープパワーダウン中ではありません。

1：ADCがディープパワーダウン中です（デフォルトのリセット状態）。

注： ソフトウェアは、A/Dコンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、JADSTP = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットを書き込むことができます。

ビット28 **ADVREGEN**：ADC電圧レギュレータの有効化

これらのビットは、ADC電圧レギュレータを有効にするために、ソフトウェアによってセットされます。

較正の起動やADCの有効化などの操作を実行する前に、まずはADC電圧レギュレータを有効にし、ソフトウェアはレギュレータの起動時間まで待つ必要があります。

0：ADC電圧レギュレータは無効です。

1：ADC電圧レギュレータは有効です。

ADC電圧レギュレータの有効化および無効化シーケンスの詳細については、セクション 33.4.6：ADCディープ･パワー･ダウン･モード（DEEPPWD）およびADC電圧レギュレータ（ADVREGEN）を参照してください。

ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、ADEN = 0）、このビットフィールドをプログラムできます。

ビット27:24 **CALINDEX[3:0]**：較正係数

このビットフィールドは、読出し／書込みのための較正係数を制御します。

較正インデックス0はシングルエンドおよび差動オフセット専用で、較正インデックス1 ~ 7は直線性較正係数専用、較正インデックス8は内部オフセット専用です。

0000：オフセット較正係数

0001：直線較正係数1

0010：直線較正係数2

0011：直線較正係数3

0100：直線較正係数4

0101：直線較正係数5

0110：直線較正係数6

0111：直線性較正係数7および内部オフセット（書込みアクセス専用）

1000：内部オフセット（読出しアクセス専用）

1001：較正モードの選択

その他：予約済み。使用できません。

注： **ADC_CALFACT2[31:0] はCALINDEX[3:0] 較正係数データの位置に対応します（詳細はセクション 33.4.8：較正（ADCAL、ADCALLIN、ADC_CALFACT）を参照）。**

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **ADCALLIN**：直線性較正

このビットは、直線性較正を有効にするために、ソフトウェアによってセット／クリアされます。

0：ADCALを書き込むと、直線性較正なしで較正が起動されます。

1：ADCALを書き込むと、直線性較正ありで較正が起動されます。

注： **ソフトウェアは、A/Dコンバータが無効で較正中でないときだけ（ADCAL = 0、JADSTART = 0、JADSTP = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットを書き込むことができます。**

ビット15:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **JADSTP**：ADCのインジェクト変換停止コマンド

このビットは、実行中のインジェクト変換を停止および破棄するためにソフトウェアによってセットされます（JADSTPコマンド）。

変換が効果的に破棄され、ADCインジェクトシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADCは新しいインジェクト変換の開始（JADSTARTコマンド）を受け入れる準備ができます。

0：ADCインジェクト変換停止コマンドは実行中ではありません。

1：1を書き込むと、実行中のインジェクト変換が停止します。1として読み出されたときには、ADSTPコマンドが実行中であることを意味します。

注： **ソフトウェアは、JADSTART = 1かつADDIS = 0である（ADCが有効であり、最終的にインジェクト変換が変換され、ADCを無効にする保留中のリクエストがない）ときだけ、JADSTPをセットできます。**

自動インジェクションモード（JAUTO = 1）では、ADSTPビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTPは使用しないでください）。

ビット4 **ADSTP**：ADCのレギュラ変換停止コマンド

このビットは、実行中のレギュラ変換を停止および破棄するためにソフトウェアによってセットされます（ADSTPコマンド）。

変換が効果的に破棄され、ADCレギュラシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADCは新しいレギュラ変換の開始（ADSTARTコマンド）を受け入れる準備ができます。

0：ADCのレギュラ変換停止コマンドは実行中ではありません。

1：1を書き込むと、実行中のレギュラ変換が停止します。1として読み出されたときには、ADSTPコマンドが実行中であることを意味します。

注： ソフトウェアは、ADSTART = 1かつADDIS = 0である（A/Dコンバータが有効であり、最終的にレギュラ変換が実行され、A/Dコンバータを無効にする保留中のリクエストがない）ときだけ、ADSTPをセットできます。

自動インジェクション･モード（JAUTO = 1）で ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

ビット3 **JADSTART**：ADCのインジェクト変換開始

このビットは、インジェクトチャネルのADC変換を開始するためにソフトウェアによってセットされます。設定ビットJEXTEN[1:0] に応じて、変換はただちに開始されるか（ソフトウェア･トリガ設定）またはインジェクト･ハードウェア･トリガ･イベントが発生（ハードウェア･トリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モードでソフトウェアトリガが選択された場合（JEXTSEL = 0x0）：インジェクト変換シーケンス終了（JEOS）フラグのアサート時。
- すべての場合：JADSTPコマンドの実行後、JADSTPビットのハードウェアによるクリアと同時。

0：ADCインジェクト変換は実行中ではありません。

1：1を書き込むと、インジェクト変換が開始します。1として読み出されたときには、ADCが動作中であり、最終的にインジェクトチャネルが変換されることを意味します。

注： ソフトウェアは、ADEN = 1かつADDIS =0である（A/Dコンバータが有効であり、A/Dコンバータを無効にする保留中のリクエストがない）ときだけ、JADSTARTをセットできます。

自動インジェクション･モード（JAUTO = 1）では、ADSTARTビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTARTはクリアされたままでなければなりません）。

ビット2 **ADSTART**：ADCのレギュラ変換開始

このビットは、レギュラチャネルのADC変換を開始するためにソフトウェアによってセットされます。設定ビットEXTEN[1:0] に応じて、変換はただちに開始されるか（ソフトウェア･トリガ設定）またはレギュラ･ハードウェア･トリガ･イベントが発生（ハードウェア･トリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モード（CONT = 0、DISCEN = 0）でソフトウェアトリガが選択された場合（EXTEN[1:0] = 0x0）：レギュラ変換シーケンス終了（EOS）フラグのアサート時。
- 不連続変換モード（CONT = 0、DISCEN = 1）でソフトウェアトリガが選択された場合（EXTEN[1:0] = 0x0）：変換終了（EOS）フラグ時。
- これ以外のすべての場合：ADSTPコマンドの実行後、ADSTPビットがハードウェアによってクリアされると同時に。

0：ADCレギュラ変換は実行中ではありません。

1：1を書き込むと、レギュラ変換が開始します。1として読み出されたときには、ADCが動作中であり、最終的にレギュラチャネルが変換されることを意味します。

注： ソフトウェアは、ADEN = 1かつADDIS = 0である（A/Dコンバータが有効であり、A/Dコンバータを無効にする保留中のリクエストがない）ときだけ、ADSTARTをセットできます。

自動インジェクション･モード（JAUTO = 1）では、ADSTARTビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTARTはクリアされたままでなければなりません）。

ビット1 **ADDIS**：ADC無効化コマンド

このビットは、ADCを無効にして（ADDISコマンド）、パワーダウン状態（OFF状態）にするためにソフトウェアによってセットされます。

ADCが効果的に無効化されると、ハードウェアによってクリアされます（ADENもこの時点でハードウェアによってクリアされます）。

0：ADDISコマンドは実行中ではありません。

1：1を書き込むと、ADCが無効化されます。1が読み出された場合、ADDISコマンドが実行中であることを意味します。

注： ソフトウェアは、ADEN = 1でADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、ADDISをセットできます。

ビット0 **ADEN**：ADCの有効化の制御

このビットは、ADCを有効化するために、ソフトウェアによってセットされます。フラグADRDYがセットされたときは、ADCは動作する準備ができています。

ADDISコマンドの実行後、ADCが無効になるとハードウェアによってクリアされます。

0：ADCは無効です（OFF状態）。

1：1を書き込むと、ADCが有効になります。

注： ソフトウェアは、1に設定する必要があるADVREGENビットを除き、ADC_CRレジスタのすべてのビットが0のとき（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、ADEN = 0）だけ、ADENをセットできます（また、ソフトウェアは、電圧レギュレータの起動時間まで待っていなければなりません）。

33.6.4 ADC設定レジスタ（ADC_CFGR1）

アドレスオフセット：0x0C

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	Res.	JDISCEN	DISCNUM[2:0]			DISCEN
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL[4:0]					Res.	RES[1:0]		DMNGT[1:0]	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:26 **AWD1CH[4:0]**：アナログウォッチドッグ1チャネルの選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：AWD1によって監視されているADCアナログ入力チャネル0

00001：AWD1によって監視されているADCアナログ入力チャネル1

.....

10011：AWD1によって監視されているADCアナログ入力チャネル19

その他：予約済み。使用できません。

注： AWD1CHで選択したチャネルは、SQRiまたはJSQRiレジスタでも選択する必要があります。

ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット25 **JAUTO**：自動インジェクトグループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動インジェクトグループ変換は無効です。

1：自動インジェクトグループ変換は有効です。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。

ビット24 **JAWD1EN**：アナログウォッチドッグ1有効（インジェクトチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルのアナログウォッチドッグ1は無効です。

1：インジェクトチャネルのアナログウォッチドッグ1は有効です。

注： ソフトウェアは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット23 **AWD1EN**：アナログウォッチドッグ1有効（レギュラチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルのアナログウォッチドッグ1は無効です。

1：レギュラチャネルのアナログウォッチドッグ1は有効です。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット22 **AWD1SGL**：ウォッチドッグ1を単一チャネルまたはすべてのチャネルで有効化

このビットは、AWD1CH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ1は有効です。

1：単一チャネルでアナログウォッチドッグ1は有効です。

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 **JDISCEN**：インジェクトチャネルの不連続モード

このビットは、グループのインジェクトチャネルに対する不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルの不連続モードは無効です。

1：インジェクトチャネルの不連続モードは有効です。

注： ソフトウェアは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTOのセット時には、ビットDISCENとJDISCENはクリアされたままでなければなりません。

ビット19:17 **DISCNUM[2:0]**：不連続モードのチャネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるレギュラチャネルの数を定義するために、ソフトウェアによって書き込まれます。

000：1チャネル

001：2チャネル

...

111：8チャネル

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット16 **DISCEN**：レギュラチャネルの不連続モード

このビットは、レギュラチャネルの不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルの不連続モードは無効です。

1：レギュラチャネルの不連続モードは有効です。

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方を設定することは禁じられています。

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTOのセット時には、ビットDISCENとJDISCENはクリアされたままでなければなりません。

ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **AUTDLY**：遅延変換モード

このビットは、自動遅延変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動遅延変換モードはオフです。

1：自動遅延変換モードはオンです。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット13 **CONT**：レギュラ変換のシングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的にレギュラ変換が行われます。

0：シングル変換モード

1：連続変換モード

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方を設定することは禁じられています。

ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット12 **OVRMOD**：オーバーランモード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADC_DRレジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADC_DRレジスタは最新の変換結果で上書きされます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット11:10 **EXTEN[1:0]**：レギュラチャネルの外部トリガの有効化と極性の選択

これらのビットは、外部トリガ極性を選択しレギュラグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット9:5 **EXTSEL[4:0]**：レギュラグループの外部トリガの選択

これらのビットは、レギュラグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：adc_ext_trg0

00001：adc_ext_trg1

...

トリガマッピングの詳細については、セクション 33.4.2 : A/Dコンバータのピンおよび内部信号のレギュラチャネルのADC外部トリガを参照してください。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3:2 **RES[1:0]**：データの分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

00：14bit

01：12bit

10：10bit

11：8bits

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット1:0 **DMNGT[1:0]**：データ管理設定

このビットは、ADCインタフェース出力データを管理する方法を選択するために、ソフトウェアによってセット／クリアされます。

00：レギュラ変換データはDRでのみ保存されます。

01：DMAワンショットモードが選択されています。

10：MDFモードが選択されています。

11：DMAサーキュラモードが選択されています。

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

33.6.5 ADC設定レジスタ2（ADC_CFGR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LSHIFT[3:0]				LFTRIG	Res.	OSR[9:0]									
rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMPTRIG	SWTRIG	BULB	Res.	Res.	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE
rw	rw	rw			rw	rw	rw	rw	rw	rw				rw	rw

ビット31:28 **LSHIFT[3:0]**：左シフト係数

このビットフィールドは、オーバーサンプリングありとなしの最終結果に適用する左シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：左シフトなし
0001：左に1bitシフト
0010：左に2bitシフト
0011：左に3bitシフト
0100：左に4bitシフト
0101：左に5bitシフト
0110：左に6bitシフト
0111：左に7bitシフト
1000：左に8bitシフト
1001：左に9bitシフト
1010：左に10bitシフト
1011：左に11bitシフト
1100：左に112bitシフト
1101：左に13bitシフト
1110：左に14bitシフト
1111：左に15bitシフト

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット27 **LFTRIG**：低周波数トリガ

このビットは、ソフトウェアによってセット／クリアされます。

0：低周波数トリガモードは無効です。
1：低周波数トリガモードは有効です。

注： ソフトウェアは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **OSR[9:0]**：オーバーサンプリング比

このビットフィールドは、オーバーサンプリング比を定義するために、ソフトウェアによってセット／クリアされます。

0：1x（オーバーサンプリングなし）

1：2x

2：3x

...

1023：1024x

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット15 **SMPTRIG**：サンプリング時間制御トリガモード

このビットは、サンプリング時間制御トリガモードを有効にするために、ソフトウェアによってセット／クリアされます。

0：サンプリング時間制御トリガモードは無効です。

1：サンプリング時間制御トリガモードは有効です。

サンプリング時間はトリガの立ち上がりエッジで開始し、変換はトリガの立ち下がりエッジで開始します。

EXTEN[1:0] ビットに01をセットする必要があります。SMPTRIGビットがセットされている場合、BULBビットをセットすることはできません。

EXTEN[1:0] ビットに00セットがされているとき、サンプリングを開始するにはSWTRIGビットをセットし、変換を開始するにはSWTRIGビットをクリアします。

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット14 **SWTRIG**：サンプリング時間制御トリガモードのソフトウェアトリガビット

このビットは、バルブサンプリングモードを有効にするために、ソフトウェアによってセット／クリアされます。

0：ソフトウェアトリガはサンプリング時間制御トリガモードの変換を開始します。

1：ソフトウェアトリガはサンプリング時間制御トリガモードのサンプリングを開始します。

注： ソフトウェアは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット13 **BULB**：バルブサンプリングモード

このビットは、バルブサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：バルブサンプリングモードは無効です。

1：バルブサンプリングモードは有効です。サンプリング期間は前の変換終了直後に開始されます。

BULBビットがセットされている場合、SMPTRIGビットをセットすることはできません。

最初のADC変換はSMPxビットで指定されたサンプリング時間で実行されます。

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット12:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **ROVSM**：レギュラオーバーサンプリングモード

このビットは、レギュラオーバーサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：連続モード：インジェクト変換がトリガされると、オーバーサンプリングが一時的に停止され、インジェクションシーケンス後に続行されます（オーバーサンプリングバッファはインジェクションシーケンス中、維持されます）。

1：再開モード：インジェクト変換がトリガされると、現在のオーバーサンプリングが中止され、インジェクションシーケンス後に最初から再開されます（オーバーサンプリングバッファはインジェクションシーケンス開始によって0になります）。

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット9 **TROVS**：トリガレギュラオーバーサンプリング

このビットは、トリガオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとに新しいトリガが必要です。

注： ソフトウェアは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット8:5 **OVSS[3:0]**：オーバーサンプリングの右シフト

このビットフィールドは、そのままのオーバーサンプリング結果に適用する右シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：右シフトなし

0001：右に1bitシフト

0010：右に2bitシフト

0011：右に3bitシフト

0100：右に4bitシフト

0101：右に5bitシフト

0110：右に6bitシフト

0111：右に7bitシフト

1000：右に8bitシフト

1001：右に9bitシフト

1010：右に10bitシフト

1011：右に11bitシフト

その他：予約済み。使用できません。

注： ソフトウェアでは、ADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **JOVSE**：インジェクトオーバーサンプリングの有効化

このビットは、インジェクトオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトオーバーサンプリングは無効です。

1：インジェクトオーバーサンプリングは有効です。

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット0 **ROVSE**：レギュラオーバーサンプリングの有効化

このビットは、レギュラオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラオーバーサンプリングは無効です。

1：レギュラオーバーサンプリングは有効です。

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

33.6.6 ADCサンプル時間レジスタ1（ADC_SMPR1）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]	SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:0 SMPx[2:0]：チャネルxサンプリング時間の選択（x = 9 ~ 0）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000：5 ADCクロックサイクル
001：6 ADCクロックサイクル
010：12 ADCクロックサイクル
011：20 ADCクロックサイクル
100：36 ADCクロックサイクル
101：68 ADCクロックサイクル
110：391 ADCクロックサイクル
111：814 ADCクロックサイクル

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.7 ADCサンプル時間レジスタ2（ADC_SMPR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP19 [2:0]			SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:0 SMPx[2:0]：チャネルxサンプリング時間の選択（x = 19 ~ 10）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプリングサイクル中は、チャネル選択ビットを変更することはできません。

000：5 ADCクロックサイクル
001：6 ADCクロックサイクル
010：12 ADCクロックサイクル
011：20 ADCクロックサイクル
100：36 ADCクロックサイクル
101：68 ADCクロックサイクル
110：391 ADCクロックサイクル
111：814 ADCクロックサイクル

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.8 ADCチャネル事前選択レジスタ（ADC_PCSEL）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCSEL 15	PCSEL 14	PCSEL 13	PCSEL 12	PCSEL 11	PCSEL 10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **PCSEL[19:0]**：チャネルi（V_{INP}[i]）の事前選択

これらのビットは、変換するI/Oインスタンスの入力チャネルを事前選択するために、ソフトウェアによって書き込まれます。

0：入力チャネルi（V_{INP}[i]）はこの変換用に事前選択されず、このチャネルをADC変換すると誤った結果が示されます。

1：入力チャネルi（V_{INP}[i]）が変換用に事前選択されます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.9 ADCレギュラシーケンスレジスタ1（ADC_SQR1）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ4 [4:0]					Res.	SQ3 [4:0]					Res.	SQ2[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ2 [3:0]				Res.	SQ1 [4:0]					Res.	Res.	L[3:0]			
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:24 **SQ4[4:0]:** レギュラーシーケンス内の4番目の変換

これらのビットには、レギュラ変換シーケンス内の4番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:18 **SQ3[4:0]:** レギュラーシーケンス内の3番目の変換

これらのビットには、レギュラ変換シーケンス内の3番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット16:12 **SQ2[4:0]:** レギュラーシーケンス内の2番目の変換

これらのビットには、レギュラ変換シーケンス内の2番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **SQ1[4:0]:** レギュラーシーケンス内の1番目の変換

これらのビットには、レギュラ変換シーケンス内の1番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット5:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **L[3:0]：**レギュラチャネルシーケンス長

これらのビットは、レギュラチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

0000：1つの変換

0001：2つの変換

...

1111：16の変換

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.10 ADCレギュラシーケンスレジスタ2（ADC_SQR2）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ9 [4:0]					Res.	SQ8 [4:0]					Res.	SQ7[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ7 [3:0]				Res.	SQ6 [4:0]					Res.	SQ5 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:24 **SQ9[4:0]:** レギュラーシーケンス内の9番目の変換

これらのビットには、レギュラ変換シーケンス内の9番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:18 **SQ8[4:0]:** レギュラーシーケンス内の8番目の変換

これらのビットには、レギュラ変換シーケンス内の8番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット16:12 **SQ7[4:0]:** レギュラーシーケンス内の7番目の変換

これらのビットには、レギュラ変換シーケンス内の7番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **SQ6[4:0]:** レギュラーシーケンス内の6番目の変換

これらのビットには、レギュラ変換シーケンス内の6番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **SQ5[4:0]:** レギュラーシーケンス内の5番目の変換

これらのビットには、レギュラ変換シーケンス内の5番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.11 ADCレギュラシーケンスレジスタ3（ADC_SQR3）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ14 [4:0]					Res.	SQ13 [4:0]					Res.	SQ12[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ12 [3:0]				Res.	SQ11 [4:0]					Res.	SQ10 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:24 **SQ14[4:0]:** レギュラーシーケンス内の14番目の変換

これらのビットには、レギュラ変換シーケンス内の14番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:18 **SQ13[4:0]:** レギュラーシーケンス内の13番目の変換

これらのビットには、レギュラ変換シーケンス内の13番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット16:12 **SQ12[4:0]:** レギュラーシーケンス内の12番目の変換

これらのビットには、レギュラ変換シーケンス内の12番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **SQ11[4:0]:** レギュラーシーケンス内の11番目の変換

これらのビットには、レギュラ変換シーケンス内の11番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **SQ10[4:0]:** レギュラーシーケンス内の10番目の変換

これらのビットには、レギュラ変換シーケンス内の10番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.12 ADCレギュラシーケンスレジスタ4（ADC_SQR4）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
					rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **SQ16[4:0]:** レギュラーシーケンス内の16番目の変換

これらのビットには、レギュラ変換シーケンス内の16番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **SQ15[4:0]:** レギュラーシーケンス内の15番目の変換

これらのビットには、レギュラ変換シーケンス内の15番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.13 ADCレギュラデータレジスタ（ADC_DR）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RDATA[31:0]**：変換されたレギュラデータ

これらのビットは読出し専用です。これらは、最後の変換されたレギュラチャネルの変換結果を含んでいます。セクション 33.4.27：データ管理 に示すように、データは左詰めまたは右詰めされています。

33.6.14 ADCインジェクトシーケンスレジスタ（ADC_JSQR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JSQ4 [4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:1]			
rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ2[0]	Res.	JSQ1 [4:0]					JEXTEN[1:0]		JEXTSEL[4:0]					JL[1:0]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 **JSQ4[4:0]:** インジェクトシーケンス内の4番目の変換

これらのビットには、インジェクト変換シーケンス内の4番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25:21 **JSQ3[4:0]:** インジェクトシーケンス内の3番目の変換

これらのビットには、インジェクト変換シーケンス内の3番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19:15 **JSQ2[4:0]:** インジェクトシーケンス内の2番目の変換

これらのビットには、インジェクト変換シーケンス内の2番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:9 **JSQ1[4:0]:** インジェクトシーケンス内の1番目の変換

これらのビットには、インジェクト変換シーケンス内の1番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット8:7 **JEXTEN[1:0]**：インジェクトチャネルの外部トリガの有効化と極性の選択

これらのビットは、外部トリガ極性を選択しインジェクトグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット6:2 **JEXTSEL[4:0]**：インジェクトグループの外部トリガの選択

これらのビットは、インジェクトグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：adc_jext_trg0

00001：adc_jext_trg1

...

トリガマッピングの詳細については、セクション 33.4.2 : A/Dコンバータのピンおよび内部信号のインジェクトチャネルのADC外部トリガを参照してください。

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット1:0 **JL[1:0]**：インジェクトチャネルシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

00：1つの変換

01：2つの変換

10：3つの変換

11：4つの変換

注： ソフトウェアでは、JADSTART = 0である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.15 ADCオフセットyレジスタ（ADC_OFRy）

アドレスオフセット：0x60 + 0x04 * (y - 1)（y = 1 ~ 4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSET_CH[4:0]					SSAT	USAT	POSOFF	OFFSET[23:16]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 **OFFSET_CH[4:0]**：データオフセットyのチャネルの選択

このビットは、OFFSETy[25:0] ビットにプログラムされたオフセットが適用されるチャネルを定義するために、ソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

OFFSETy_ENビットがセットされている場合、同じチャネルを別のADC_OFRyレジスタで選択することはできません。

ビット26 **SSAT**：符号付き飽和の有効化

このビットは、符号付き飽和機能を有効または無効にするために、ソフトウェアによって書き込まれます。（詳細はセクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、USAT、SSAT）を参照）

0：オフセットが減算され、データの完全性を維持して変換データのサイズを拡張（9bitまたは15bit符号付きフォーマット）します。

1：オフセットが減算され、変換データのサイズを維持するために結果を飽和させます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット25 **USAT**：符号なし飽和の有効化

このビットは、符号なし飽和機能を有効または無効にするために、ソフトウェアによって書き込まれます。

0：オフセットが減算され、データの完全性を維持して変換データのサイズを維持します。

1：オフセットが減算され、変換データのサイズを維持するために結果を飽和させます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット24 **POSOFF**：オフセットの符号

このビットは、正のオフセットを有効にするために、ソフトウェアによってセット／クリアされます。

0：負のオフセット

1：正のオフセット

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット23:0 **OFFSET[23:0]**：OFFSETy_CH[4:0] ビットでプログラムされるチャネル用のデータオフセットy

これらのビットは、チャネル（レギュラまたはインジェクト）を変換するときに元の変換データから引かれるオフセットyを定義するために、ソフトウェアによって書き込まれます。データオフセットyを適用するチャネルは、ビットOFFSETy_CH[4:0] ビットにプログラムする必要があります。変換結果は、ADC_DRレジスタ（レギュラ変換）またはADC_JDRyiレジスタ（インジェクト変換）から読み出すことができます。

OFFSETy[21:0] ビットフィールドがリセットされると、オフセット補償は無効化されます。

注： ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

複数のオフセット（OFFSETy）が同じチャネルを指している場合、減算時には最小のyを持つオフセットのみが考慮されます。

例：OFFSET1_CH[4:0] = 4かつOFFSET2_CH[4:0] = 4である場合、チャネル4を変換するときに減算されるのはOFFSET1[25:0] です。

33.6.16 ADCゲイン補正レジスタ（ADC_GCOMP）

アドレスオフセット：0x70

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GCOMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	GCOMPCOEFF[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **GCOMP**：ゲイン補正モード

このビットは、ゲイン補正モードを有効にするために、ソフトウェアによってセット／クリアされます。

0：ADCのレギュラ動作モード

1：ゲイン補正は有効で、すべてのチャネルに適用されます。

注： ソフトウェアは、ADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット30:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **GCOMPCOEFF[13:0]**：ゲイン補正係数

これらのビットは、ゲイン補正係数をプログラムするために、ソフトウェアによってセット／クリアされます。

00 1000 0000 0000：ゲイン係数0.5

...

01 0000 0000 0000：ゲイン係数1

10 0000 0000 0000：ゲイン係数2

11 0000 0000 0000：ゲイン係数3

...

係数を4096で除算して、0から3.999756までのゲイン係数を求めます。

注： このゲイン補正は、ADCx_CFGR2レジスタのGCOMPビットが1にセットされているときのみ適用されます。

33.6.17 ADCインジェクトデータレジスタ（ADC_JDRy）

アドレスオフセット：0x80 + 0x04 * (y - 1)、（y = 1 ~ 4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **JDATA[31:0]**：インジェクトデータ

これらのビットは読出し専用です。これらのビットには、インジェクトチャネルyの変換結果が格納されます。セクション 33.4.27：データ管理に示すように、データは左詰めまたは右詰めされています。

33.6.18 A/Dコンバータアナログウォッチドッグ2設定レジスタ（ADC_AWD2CR）

アドレスオフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **AWD2CH[19:0]**：アナログウォッチドッグ2チャネルの選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ2によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADCアナログ入力チャネルiはAWD2によって監視されません。

AWD2CH[i] = 1：ADCアナログ入力チャネルiはAWD2によって監視されます。

AWD2CH[19:0] = 000..0である場合、アナログウォッチドッグ2は無効になります。

注： AWD2CHで選択したチャネルは、SQRiまたはJSQRiレジスタでも選択する必要があります。

ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.19 A/Dコンバータアナログウォッチドッグ3設定レジスタ（ADC_AWD3CR）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **AWD3CH[19:0]**：アナログウォッチドッグ3チャネルの選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ3によって保護される入力チャネルを有効化して選択します。

AWD3CH[i] = 0：ADCアナログ入力チャネルiはAWD3によって監視されません。

AWD3CH[i] = 1：ADCアナログ入力チャネルiはAWD3によって監視されます。

AWD3CH[19:0] = 000..0である場合、アナログウォッチドッグ3は無効になります。

注： AWD3CHで選択したチャネルは、SQRiまたはJSQRiレジスタでも選択する必要があります。

ソフトウェアでは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

33.6.20 ADCウォッチドッグ閾値レジスタ1（ADC_LTR1）

アドレスオフセット：0xA8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR1 [24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **LTR1[24:0]**：アナログウォッチドッグ1低閾値

これらのビットは、アナログウォッチドッグ1の低閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.21 ADCウォッチドッグ閾値レジスタ1（ADC_HTR1）

アドレスオフセット：0xAC

リセット値：0x01FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AWDFILT1 [2:0]			Res.	Res.	Res.	Res.	HTR1 [24:16]								
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 **AWDFILT1[2:0]**：アナログウォッチドッグフィルタリングパラメータ

このビットは、ソフトウェアによってセット／クリアされます。
000：フィルタリングなし
001：2個の連続した検出によってAWDxフラグまたは割込みが生成されます。
...
111：8個の連続した検出によってAWDxフラグまたは割込みが生成されます。

注： ソフトウェアは、ADSTART = 0かつJADSTART = 0である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット28:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **HTR1[24:0]**：アナログウォッチドッグ1高閾値

これらのビットは、アナログウォッチドッグ1の高閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.22 A/Dコンバータウォッチドッグ低閾値レジスタ2（ADC_LTR2）

アドレスオフセット：0xB0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR2 [24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **LTR2[24:0]**：アナログウォッチドッグ2低閾値

これらのビットは、アナログウォッチドッグ2の低閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.23 A/Dコンバータウォッチドッグ高閾値レジスタ2（ADC_HTR2）

アドレスオフセット：0xB4

リセット値：0x01FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HTR2 [24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **HTR2[24:0]**：アナログウォッチドッグ2高閾値

これらのビットは、アナログウォッチドッグ2の高閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.24 A/Dコンバータウォッチドッグ低閾値レジスタ3（ADC_LTR3）

アドレスオフセット：0xB8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR3 [24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **LTR3[24:0]**：アナログウォッチドッグ3低閾値

これらのビットは、アナログウォッチドッグ3の低閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.25 A/Dコンバータウォッチドッグ高閾値レジスタ3（ADC_HTR3）

アドレスオフセット：0xBC

リセット値：0x01FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HTR3 [24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:0 **HTR3[24:0]**：アナログウォッチドッグ3高閾値

これらのビットは、アナログウォッチドッグ3の高閾値を定義するために、ソフトウェアによって書き込まれます。セクション 33.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

33.6.26 A/Dコンバータ差動モード選択レジスタ（ADC_DIFSEL）

アドレスオフセット：0xC0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIFSEL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **DIFSEL[19:0]**：チャネル（19 ~ 0）の差動モード

これらのビットは、ソフトウェアによってセット／クリアされます。これにより、チャネルの設定をシングルエンドモードまたは差動モードから選択できます。

DIFSEL[i] = 0：ADCアナログ入力チャネルiはシングルエンドモードで設定されます。

DIFSEL[i] = 1：ADCアナログ入力チャネルiは差動モードで設定されます。

注： ソフトウェアは、A/Dコンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、JADSTP = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、これらのビットに書き込むことができます。

33.6.27 ADCクロック制御レジスタ（ADC_CALFACT）

アドレスオフセット：0xC4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	CAPTURE_COEF	LATCH_COEF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALIDITY
						rw	rw								r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I_APB_DATA[7:0]								I_APB_ADDR[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **CAPTURE_COEF**：較正係数キャプチャの有効化ビット

このビットでは、内部較正係数のキャプチャを有効化できます。

0：較正係数はキャプチャされません。

1：較正係数がCALFACT[31:0] ビットにセットされていて、較正係数のインデックスがCALINDEX[3:0] ビットにより定義されます。

ビット24 **LATCH_COEF**：較正係数ラッチの有効化ビット

このビットでは、CALFACT[31:0] ビット内で較正係数をラッチできます。

0：影響なし

1：較正係数が LATCH_COEFビットのアナログブロックでラッチされ、0から1に遷移します。較正係数のラッチの前に、CALFACT[31:0] ビットをCALINDEX[3:0] ビットの内容によりプログラムする必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **VALIDITY**：遅延書込みアクセスステータスビット

このビットは、ADCのデジタルブロックとアナログブロックの間の通信ステータスを示します。

0：動作が進行中です。

1：動作が完了しました。

ビット15:8 **I_APB_DATA[7:0]**：遅延書込みアクセスデータ

このビットフィールドには、遅延書込みアクセス中に書き込まれるデータが格納されます。

ビット7:0 **I_APB_ADDR[7:0]**：遅延書込みアクセスアドレス

このビットフィールドには、遅延書込みアクセス中に書き込まれるアドレスが格納されます。

33.6.28 ADC較正係数レジスタ（ADC_CALFACT2）

アドレスオフセット：0xC8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALFACT[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALFACT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CALFACT[31:0]**：直線性またはオフセット較正係数

これらのビットには、ハードウェアまたはソフトウェアによって書き込むことができます。

ここには、32bitのオフセットまたは直線性較正係数が格納されます。

CAPTURE_COEFがセットされている場合、アナログブロックの較正係数が読み戻されて CALFACT[31:0] に格納され、CALINDEX[3:0] ビットによってインデックスが付与されます。

LATCH_COEFがセットされている場合、アナログブロックの較正係数が CALFACT[31:0] でプログラムされた値により更新され、CALINDEX[3:0] ビットによってインデックスが付与されます。

すべての更新係数を読み出すには、ADC_CALFACT2レジスタに9回アクセスします。

すべての更新係数を書き込むには、ADC_CALFACT2レジスタに8回アクセスします。

注： ソフトウェアは、ADEN = 1、ADSTART = 0、およびJADSTART = 0のときのみ（A/Dコンバータが有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

33.7 ADC共通レジスタ

これらのレジスタでは、マスタおよびスレーブADCに共通の制御およびステータスレジスタを定義します。

33.7.1 ADC共通ステータスレジスタ（ADC12_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0000

このアドレスオフセットはマスタA/Dコンバータのベースアドレス + 0x300に対する相対値です。

このレジスタにより、別のADCのステータスビットの写像を得ることができます。ただし、これは読出し専用で、別のステータスビットをクリアすることはできません。その代わり、対応するADC_ISRレジスタで1を書き込むことによって各ステータスビットをクリアする必要があります。

ADC1およびADC2は、同じインタフェースによって制御されます。

このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3 : A/Dコンバータの実装を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	LDORDY_SLV	Res.	Res.	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV
			r			r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LDORDY_MST	Res.	Res.	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
			r			r	r	r	r	r	r	r	r	r	r

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **LDORDY_SLV**：スレーブADCのADC電圧レギュレータレディフラグ
このビットは対応するADCx+1_ISRレジスタのLDORDYビットのコピーです。

ビット27:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **AWD3_SLV**：スレーブADCのアナログウォッチドッグ3フラグ
このビットは対応するADCx+1_ISRレジスタのAWD3bitのコピーです。

ビット24 **AWD2_SLV**：スレーブADCのアナログウォッチドッグ2フラグ
このビットは対応するADCx+1_ISRレジスタのAWD2bitのコピーです。

ビット23 **AWD1_SLV**：スレーブADCのアナログウォッチドッグ1フラグ
このビットは対応するADCx+1_ISRレジスタのAWD1bitのコピーです。

ビット22 **JEOS_SLV**：スレーブADCのインジェクトシーケンスの終了フラグ
このビットは対応するADCx+1_ISRレジスタのJEOSビットのコピーです。

ビット21 **JEOC_SLV**：スレーブADCのインジェクト変換の終了フラグ
このビットは対応するADCx+1_ISRレジスタのJEOCビットのコピーです。

ビット20 **OVR_SLV**：スレーブADCのオーバーランフラグ
このビットは対応するADCx+1_ISRレジスタのOVRビットのコピーです。

ビット19 **EOS_SLV**：スレーブADCのレギュラシーケンスの終了フラグ
このビットは対応するADCx+1_ISRレジスタのEOSビットのコピーです。

ビット18 **EOC_SLV**：スレーブADCのレギュラ変換の終了フラグ
このビットは対応するADCx+1_ISRレジスタのEOCビットのコピーです。

ビット17 **EOSMP_SLV**：スレーブADCのサンプリングフェーズの終了フラグ
このビットは対応するADCx+1_ISRレジスタのEOSMP2bitのコピーです。

ビット16 **ADRDY_SLV**：スレーブADCのレディ状態
このビットは対応するADCx+1_ISRレジスタのADRDYビットのコピーです。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LDORDY_MST**：マスタADCのADC電圧レギュレータレディフラグ
このビットは対応するADC_ISRレジスタのLDORDYビットのコピーです。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **AWD3_MST**：マスタADCのアナログウォッチドッグ3フラグ
このビットは対応するADC_ISRレジスタのAWD3bitのコピーです。

ビット8 **AWD2_MST**：マスタADCのアナログウォッチドッグ2フラグ
このビットは対応するADC_ISRレジスタのAWD2bitのコピーです。

ビット7 **AWD1_MST**：マスタADCのアナログウォッチドッグ1フラグ
このビットは対応するADC_ISRレジスタのAWD1bitのコピーです。

ビット6 **JEOS_MST**：マスタADCのインジェクトシーケンスの終了フラグ
このビットは対応するADC_ISRレジスタのJEOSビットのコピーです。

ビット5 **JEOC_MST**：マスタADCのインジェクト変換の終了フラグ
このビットは対応するADC_ISRレジスタのJEOCビットのコピーです。

ビット4 **OVR_MST**：マスタADCのオーバーランフラグ
このビットは対応するADC_ISRレジスタのOVRビットのコピーです。

ビット3 **EOS_MST**：マスタADCのレギュラシーケンスの終了フラグ
このビットは対応するADC_ISRレジスタのEOSビットのコピーです。

ビット2 **EOC_MST**：マスタADCのレギュラ変換の終了フラグ
このビットは対応するADC_ISRレジスタのEOCビットのコピーです。

ビット1 **EOSMP_MST**：マスタADCのサンプリングフェーズの終了フラグ
このビットは対応するADC_ISRレジスタのEOSMPビットのコピーです。

ビット0 **ADRDY_MST**：マスタADCのレディ状態
このビットは対応するADC_ISRレジスタのADRDYビットのコピーです。

33.7.2 ADC共通制御レジスタ（ADC12_CCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

このアドレスオフセットはマスタA/Dコンバータのベースアドレス + 0x300に対する相対値です。

ADC12_CCRはADC1とADC2で共通です。ADC2は一部のデバイスでは利用できません（セクション 33.3：A/Dコンバータの実装を参照)。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	VSENSESEL	VREFEN	PRESC[3:0]				Res.	Res.
							rw	rw	rw	rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DAMDF[1:0]		Res.	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
rw	rw			rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **VBATEN**：VBATの有効化

このビットは、V_{BAT} チャネルを制御するために、ソフトウェアによってセット／クリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

注： ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットに書き込むことができます。

ビット23 **VSENSESEL**：温度センサの電圧の選択

このビットは、温度センサチャネルを制御するために、ソフトウェアによってセット／クリアされます。

0：温度センサチャネルは無効です。

1：温度センサチャネルは有効です。

注： ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットに書き込むことができます。

ビット22 **VREFEN**：V_{REFINT} の有効化

このビットは、VREFINT を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} チャネルは無効です。

1：V_{REFINT} チャネルは有効です。

注： ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットに書き込むことができます。

ビット21:18 **PRESC[3:0]**：ADCプリスケーラ

これらのビットは、ADCクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべてのADCで共通です。

0000：ADC入力クロックは分周されません。
0001：ADC入力クロックが2分周されます。
0010：ADC入力クロックが4分周されます。
0011：ADC入力クロックが6分周されます。
0100：ADC入力クロックが8分周されます。
0101：ADC入力クロックが10分周されます。
0110：ADC入力クロックが12分周されます。
0111：ADC入力クロックが16分周されます。
1000：ADC入力クロックが32分周されます。
1001：ADC入力クロックが64分周されます。
1010：ADC入力クロックが128分周されます。
1011：ADC入力クロックが256分周されます。
その他：予約済み。使用できません。

注： ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットに書き込むことができます。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15:14 **DAMDF[1:0]**：デュアルADCモードのデータフォーマット

このビットフィールドは、ソフトウェアによってセット／クリアされます。共通データレジスタADC12_CDRのデータフォーマットを指定します。

00：データパッキングなしデュアルADCモード（ADC12_CDRレジスタとADC12_CDR2レジスタは不使用）。
01：予約済み。
10：32bit分解能以下10bit分解能までのデータフォーマット設定モード
11：8bit分解能のデータフォーマット設定モード

注： このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3：A/Dコンバータの実装を参照）。

ソフトウェアは、A/Dコンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、これらのビットに書き込むことができます。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **DELAY[3:0]**：マスタADCサンプリングフェーズと終了から

スレーブADCサンプリングフェーズの開始までの遅延

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットはデュアルインタリーブモードで使用します。ADC分解能の値対DELAYビット値については、表316 を参照してください。

注： このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3：A/Dコンバータの実装を参照）。

ソフトウェアは、A/Dコンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、これらのビットに書き込むことができます。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DUAL[4:0]**：デュアルADCモードの選択

これらのビットは、動作モードを選択するために、ソフトウェアによって書き込まれます。

すべてのA/Dコンバータは単独で動作します。

00000：独立モード

00001から01001の設定は、次の動作モードに対応します。これらはデュアルモードで、マスタおよびスレーブADCは協調して動作します。

00001：レギュラ同時 + インジェクト同時モードの組み合わせ

00010：レギュラ同時 + オルタネートトリガモードの組み合わせ

00011：インタリーブモード + インジェクト同時モードの組み合わせ

00100：予約済み

00101：インジェクト同時モードのみ

00110：レギュラ同時モードのみ

00111：インタリーブモードのみ

01001：オルタネートトリガモードのみ

他のすべての組み合わせは予約済みであり、プログラムすることはできません。

注：　このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3：A/Dコンバータの実装を参照）。

ソフトウェアは、A/Dコンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、これらのビットに書き込むことができます。

表316.DELAYビット対ADC分解能

DELAYビット	14bit分解能	12bit分解能	10bit分解能	8bit分解能
0000	1 * $F_{adc_ker_ck}$	1 * $F_{adc_ker_ck}$	1 * $F_{adc_ker_ck}$	1 * $F_{adc_ker_ck}$
0001	2 * $F_{adc_ker_ck}$	2 * $F_{adc_ker_ck}$	2 * $F_{adc_ker_ck}$	2 * $F_{adc_ker_ck}$
0010	3 * $F_{adc_ker_ck}$	3 * $F_{adc_ker_ck}$	3 * $F_{adc_ker_ck}$	3 * $F_{adc_ker_ck}$
0011	4 * $F_{adc_ker_ck}$	4 * $F_{adc_ker_ck}$	4 * $F_{adc_ker_ck}$	4 * $F_{adc_ker_ck}$
0100	5 * $F_{adc_ker_ck}$	5 * $F_{adc_ker_ck}$	5 * $F_{adc_ker_ck}$	5 * $F_{adc_ker_ck}$
0101	6 * $F_{adc_ker_ck}$	6 * $F_{adc_ker_ck}$	6 * $F_{adc_ker_ck}$	6 * $F_{adc_ker_ck}$
0110	7 * $F_{adc_ker_ck}$	7 * $F_{adc_ker_ck}$	7 * $F_{adc_ker_ck}$	7 * $F_{adc_ker_ck}$
0111	8 * $F_{adc_ker_ck}$	8 * $F_{adc_ker_ck}$	8 * $F_{adc_ker_ck}$	8 * $F_{adc_ker_ck}$
1000	9 * $F_{adc_ker_ck}$	9 * $F_{adc_ker_ck}$	9 * $F_{adc_ker_ck}$	9 * $F_{adc_ker_ck}$
1001	10 * $F_{adc_ker_ck}$	10 * $F_{adc_ker_ck}$	10 * $F_{adc_ker_ck}$	10 * $F_{adc_ker_ck}$
1010	11 * $F_{adc_ker_ck}$	11 * $F_{adc_ker_ck}$	11 * $F_{adc_ker_ck}$	11 * $F_{adc_ker_ck}$
1011	12 * $F_{adc_ker_ck}$	12 * $F_{adc_ker_ck}$	12 * $F_{adc_ker_ck}$	12 * $F_{adc_ker_ck}$
1100	13 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$
1101	14 * $F_{adc_ker_ck}$	14 * $F_{adc_ker_ck}$	14 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$
1110	15 * $F_{adc_ker_ck}$	15 * $F_{adc_ker_ck}$	15 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$
1111	16 * $F_{adc_ker_ck}$	16 * $F_{adc_ker_ck}$	15 * $F_{adc_ker_ck}$	13 * $F_{adc_ker_ck}$

33.7.3 デュアルモード用ADC共通レギュラデータレジスタ（ADC12_CDR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このアドレスオフセットはマスタA/Dコンバータのベースアドレス + 0x300に対する相対値です。

ADC1およびADC2は、同じインタフェースによって制御されます。

このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3：A/Dコンバータの実装を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA_SLV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA_MST[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **RDATA_SLV[15:0]**：スレーブADCのレギュラデータ

デュアルモードでは、これらのビットにはスレーブADCのレギュラデータが格納されます。セクション 33.4.32：デュアルADCモードを参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、USAT、SSAT）に示すように適用されています。

ビット15:0 **RDATA_MST[15:0]**：マスタADCのレギュラデータ

デュアルモードでは、これらのビットにはマスタADCのレギュラデータが格納されます。セクション 33.4.32：デュアルADCモードを参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、USAT、SSAT）に示すように適用されています。

DAMDF[1:0] = 11モードでは、ビット15:8にはSLV_ADC_DR[7:0] が、ビット7:0にはMST_ADC_DR[7:0] が含まれています。

33.7.4 32bitデュアルモード用ADC共通レギュラデータレジスタ（ADC12_CDR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

このアドレスオフセットはマスタA/Dコンバータのベースアドレス + 0x300に対する相対値です。

ADC1およびADC2は、同じインタフェースによって制御されます。

このレジスタは、デュアルモードに対応しているデバイスでのみ利用できます（セクション 33.3：A/Dコンバータの実装を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA_ALT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA_ALT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RDATA_ALT[31:0]**：マスタ／スレーブオルタネートADCのレギュラデータ

デュアルモードでは、これらのビットにはマスタADCとスレーブADCのどちらかのレギュラ32bitデータが格納されます。セクション 33.4.32：デュアルADCモードを参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、USAT、SSAT）に示すように適用されています。

33.8 ADCレジスタマップ

表317.ADCグローバルレジスタマップ

オフセット	レジスタ
0x000 - 0x0D0	マスタADC1
0x0D4 - 0x0FC	予約済み
0x100 - 0x1D0	スレーブADC2
0x1D4 - 0x2FC	予約済み
0x300 - 0x310	マスタおよびスレーブADCに共通のレジスタ

表318.ADCレジスタマップと各ADCのリセット値（マスタADCのオフセットは0x000、スレーブADCのオフセットは0x100）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	ADC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDORDY	Res.	Res.	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																				0			0	0	0	0	0	0	0	0	0	0
0x04	ADC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x08	ADC_CR	ADCAL	Res.	DEEPPWD	ADVREGEN	CALINDEX[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADCALIN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
	リセット値	0		0	0	0	0	0	0								0											0	0	0	0	0	0
0x0C	ADC_CFGR1	Res.	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	Res.	JDISCEN	DISCNUM [2:0]			DISCEN	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL[4:0]					Res.	RES [1:0]		DMNGT [1:0]		
	リセット値		0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0
0x10	ADC_CFGR2	LSHIFT[3:0]				LSTRIG	Res.	OSR[9:0]										SMPTRIG	SWTRIG	BULB	Res.	Res.	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE	
	リセット値	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0				0	0
0x14	ADC_SMPR1	Res.	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:0]			SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	ADC_SMPR2	Res.	Res.	SMP19 [2:0]			SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:0]			SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	ADC_PCSEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16	PCSEL15	PCSEL14	PCSEL13	PCSEL12	PCSEL11	PCSEL10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20 ~0x2C	予約済み	Res.																															
0x30	ADC_SQR1	Res.	Res.	Res.	SQ4 [4:0]					Res.	SQ3 [4:0]						Res.	SQ2 [4:0]				Res.	SQ1 [4:0]					Res.	Res.	L[3:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0			0	0	0	0		0	0	0	0	0			0	0	0	0
0x34	ADC_SQR2	Res.	Res.	Res.	SQ9 [4:0]					Res.	SQ8 [4:0]						Res.	SQ7 [4:0]				Res.	SQ6 [4:0]					Res.	Res.	SQ5 [4:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0			0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x38	ADC_SQR3	Res.	Res.	Res.	SQ14 [4:0]					Res.	SQ13 [4:0]						Res.	SQ12 [4:0]				Res.	SQ11 [4:0]					Res.	SQ10 [4:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0			0	0	0	0		0	0	0	0	0		0	0	0	0	0

表318.ADCレジスタマップと各ADCのリセット値（マスタADCのオフセットは0x000、スレーブADCのオフセットは0x100）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3C	ADC_SQR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
	リセット値																						0	0	0	0	0		0	0	0	0	0
0x40	ADC_DR	RDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44 ~0x48	予約済み	Res.																															
0x4C	ADC_JSQR	JSQ4 [4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:0]					Res.	JSQ1 [4:0]			JEXTEN [1:0]		JEXTSEL [4:0]				JL[1:0].				
	リセット値	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50 ~ 0x5C	予約済み	Res.																															
0x60	ADC_OFR1	OFFSET1_CH[4:0]					SSAT	USAT	POSOFF	OFFSET[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	ADC_OFR2	OFFSET2_CH[4:0]					SSAT	USAT	POSOFF	OFFSET[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	ADC_OFR3	OFFSET3_CH[4:0]					SSAT	USAT	POSOFF	OFFSET[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	ADC_OFR4	OFFSET4_CH [4:0]					SSAT	USAT	POSOFF	OFFSET[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	ADC_GCOMPR	GCOMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GCOEFF[13:0]													
	リセット値	0																		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 ~ 0x7C	予約済み	Res.																															
0x80	ADC_JDR1	JDATA1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	ADC_JDR2	JDATA2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	ADC_JDR3	JDATA3 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	ADC_JDR4	JDATA4 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90 ~0x9C	予約済み	Res.																															
0xA0	ADC_AWD2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	ADC_AWD3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA8	ADC_LTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR1[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xAC	ADC_HTR1	AWDFILT1 [2:0]			Res.	Res.	Res.	Res.	HTR1 [24:0]																								
	リセット値	0	0	0					1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表318.ADCレジスタマップと各ADCのリセット値（マスタADCのオフセットは0x000、スレーブADCのオフセットは0x100）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB0	ADC_LTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR2 [24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	ADC_HTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HTR2 [24:0]																								
	リセット値								1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB8	ADC_LTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTR3 [24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xBC	ADC_HTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HTR3 [24:0]																								
	リセット値								1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC0	ADC_DIFSEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC4	ADC_CALFACT	Res.	Res.	Res.	Res.	Res.	Res.	CAPTURE_COEF	LATCH_COEF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALIDITY	I_APB_DATA[7:0]								I_APB_ADDR[7:0]							
	リセット値							0	0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC8	ADC_CALFACT2	CALFACT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表319.ADCレジスタマップとリセット値（マスタおよびスレーブADC共通レジスタ）オフセット = 0x300

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	ADC12_CSR	Res.	Res.	Res.	LDORDY_SLV	Res.	Res.	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV	Res.	Res.	Res.	LDORDY_MST	Res.	Res.	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
					スレーブADC2																マスタADC1												
	リセット値				0			0	0	0	0	0	0	0	0	0	0				0			0	0	0	0	0	0	0	0	0	0
0x04	予約済み	Res.																															
0x08	ADC12_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	VSENSESEL	VREFEN	PRESC[3:0]				Res.	Res.	DAMDF[1:0]		Res.	Res.	DELAY[3:0]			Res.	Res.	Res.	DUAL[4:0]					
	リセット値								0	0	0	0	0	0	0			0	0			0	0	0	0				0	0	0	0	0
0x0C	ADC12_CDR	RDATA_SLV[15:0]																RDATA_MST[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	ADC12_CDR2	RDATA_ALT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3 を参照してください。

34 アナログデジタルコンバータ（ADC4）

34.1 概要

この12bit ADCは、逐次比較型アナログデジタルコンバータです。最大 25 の多重化チャネルを持ち、最大19の外部ソースおよび 6 つの内部ソースからの信号を測定することができます。さまざまなチャネルのA/D変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADCの結果は、左詰めまたは右詰めで16bitのデータレジスタに格納されます。

アナログウォッチドッグ機能により、入力電圧がユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードが実装されています。

組み込みのオーバーサンプリング回路により、CPUの計算負荷を軽減しながらアナログ性能を高めることができます。

34.2 ADCの主な機能

- 高性能
  - 12bit、10bit、8bit、または6bitの設定可能な分解能
  - ADC変換時間：12bit分解能（2.5 Msps）で0.4 µs。分解能を下げることによって、より高速な変換時間を達成可能。
  - 自己較正
  - プログラム可能なサンプリング時間
  - 組み込みのデーター貫性によるデータ配置
  - DMAサポート
- 低消費電力
  - アプリケーションでは最適な ADC 性能を維持しながら、バスクロックの周波数を低減し、低電力動作を実現できます。たとえば、バスクロックの周波数にかかわらず、0.4 µsの変換時間が維持されます。
  - ウェイトモード：低周波数のバスクロックのアプリケーションでのADCオーバーランを防止します。
  - オートオフモード：アクティブ変換フェーズ中以外は、ADC の電源は自動的に切れます。これはADCの消費電力を大幅に低減します。
- 自律モード
  - ストップ･モードでは変換とDMA転送をサポート
  - ADC割込み時のSTOPからのウェイクアップ
  - ディープ･パワー･ダウン･モードの開始と終了の自動管理
- アナログ入力チャネル
  - 外部アナログ入力
  - 内部温度センサ用の1つのチャネル（V_{SENSE}）
  - 内部基準電圧用の $_1$ つのチャネル（VREFINT）
  - 内部デジタルコア電圧用の1つのチャネル（V_{CORE}）
  - 外部VBAT電源ピンを監視するための1つのチャネル
  - D/Aコンバータ内部チャネルへの接続

- 変換開始は、次のように開始できます。
  - ソフトウェアによって
  - 極性が設定可能なハードウェアトリガによって（タイマイベントまたはGPIO入力イベント）
- 変換モード
  - 単一チャネルを変換でき、または一連のチャネルをスキャンできます。
  - シングルモードは、選択された入力をトリガごとに1回変換します。
  - 連続モードは、選択された入力を連続的に変換します。
  - 不連続モード
- サンプリング終了時、変換終了時、シーケンス変換終了時、およびアナログウォッチドッグイベントまたはオーバーランイベント時に割込みを生成します。
- アナログウォッチドッグ
- オーバーサンプリング回路
  - 16bitデータレジスタ
  - 2 ～ 256倍までオーバーサンプリング比を調整可能
  - 最大8bitまでプログラム可能なデータシフト
- ADC入力電圧範囲：$V_{SSA} \leq V_{IN} \leq V_{REF+}$

34.3 A/Dコンバータの実装

表320.ADCの主な機能(1)

A/Dコンバータのモード／機能	STM32U535/545/575/585	STM32U59x/5Ax/5Fx/5Gx		STM32U535/545/575/585/59x/5Ax/5Fx/5Gx
	ADC1	ADC1	ADC2	ADC4
分解能	14bit			12bit
14bit分解能用の最大サンプリング速度	2.5Msps			2.5Msps
ハードウェアオフセット較正	X			X
ハードウェア直線性較正	X			-
シングルエンド入力	X			X
差動入力	X			-
インジェクトチャネル変換	X			-
オーバーサンプリング	最大x1024			最大x256
データレジスタ	32bit			16bit
DMAサポート	X			X
MDFへのパラレルデータ出力	X			-
デュアル･モード	-	X		-
自律モード	-			X
オフセット補正	X			-
ゲイン補正	X			-
アナログウォッチドッグの数	3			3
STOPモードからのウェイクアップ	-			X(2)

1. 注：「X」はサポート対象、「-」はサポート対象外です。
2. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

表321.温度センサ較正値のメモリ位置

名前	説明	メモリアドレス
TS_CAL1	30°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）で ADC1 によって取得された温度センサの14bit元データ	0x0BFA0710 ~ 0x0BFA0711
TS_CAL2	130°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）で ADC1 によって取得された温度センサの14bit元データ	0x0BFA0742 ~ 0x0BFA0743

表322.内部基準電圧センサの較正値のメモリ位置

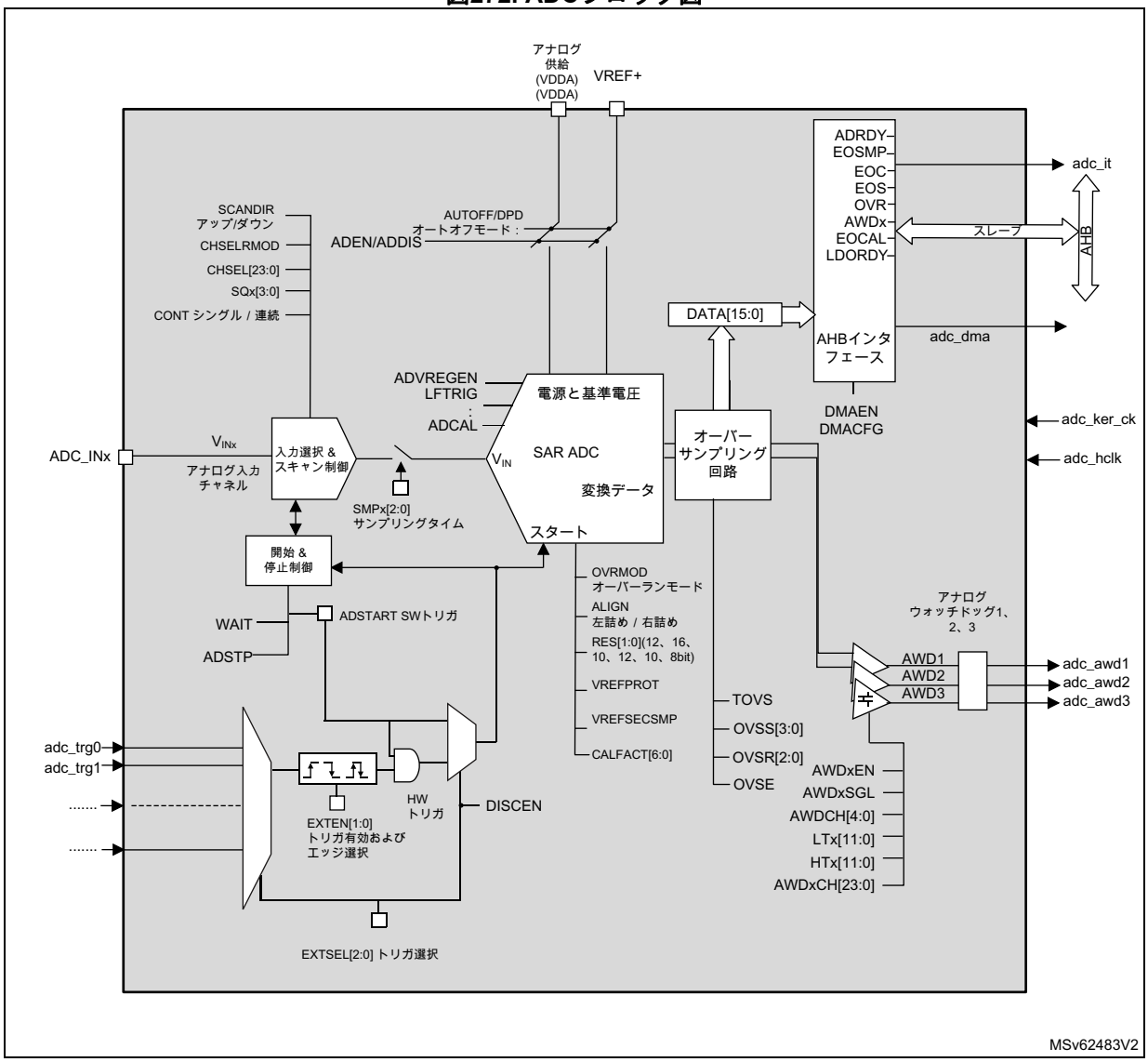
名前	説明	メモリアドレス
VREFINT_CAL	30°C（± 5°C）、V_{DDA} = V_{REF+} = 3.0V（±10mV）でADC1によって取得された14bit元データ	0x0BFA 07A5 ~ 0x0BFA 07A6

34.4 ADCの機能説明

34.4.1 ADCのブロック図

図272にADCブロック図を、表323にADCピンの説明を示します。

図272. ADCブロック図



34.4.2 A/Dコンバータのピンおよび内部信号

表323.A/Dコンバータの各入出力ピン

ピン名	信号タイプ	説明
VDDA	入力、アナログ電源	ADCのアナログ電源と正基準電圧、$V_{DDA} \geq V_{DD}$
VSSA	入力、アナログ供給グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド。

表323.A/Dコンバータの各入出力ピン （続き）

ピン名	信号タイプ	説明
VREF+	入力、基準電圧正	A/Dコンバータのハイレベル／正基準電圧。
ADC_INx	アナログ入力信号	最大19外部アナログ入力チャネル。

表324.ADC内部入力／出力信号

内部信号名	信号タイプ	説明
V_{IN}[x]	アナログ入力	内部チャネルまたはADC_INx 外部チャネルを制御するアナログ入力チャネル。
adc_trgx	入力	ADC変換トリガ
adc_awdx	出力	オンチップタイマに接続された内部アナログウォッチドッグの出力信号（x = アナログウォッチドッグ番号 = 1、2、3）
adc_it	出力	ADC割込み
adc_hclk	入力	AHBクロック
adc_ker_ck	入力	RCCブロックからのADCカーネル･クロック入力。
adc_dma	出力	ADC DMAリクエスト

表325.ADC相互接続

信号名	転送元／転送先
ADC4 V_{IN}[13]	V_{SENSE}VSENSE（内部温度センサ出力電圧）
ADC4 V_{IN}[0]	V_{REFINT} （内部基準電圧からのバッファ付き電圧）。
ADC4 V_{IN}[14]	V_{BAT}/4 （4で分圧されたVBATピン入力電圧）。
ADC4 V_{IN}[12]	V_{CORE} （内部ロジック電源電圧）。
ADC4 V_{IN}[21]	dac1_out1
	dac1_out2
adc_trg0	tim1_trgo2
adc_trg1	tim1_oc4
adc_trg2	tim2_trgo
adc_trg3	tim15_trgo
adc_trg4	tim6_trgo
adc_trg5	lptim1_ch1
adc_trg6	lptim3_ch2
adc_trg7	exti15

34.4.3 ADC電圧レギュレータ（ADVREGEN）

ADCには特定の内部電圧レギュレータがあり、ADCを使用する前に有効にして、安定させる必要があります。

ADC内部電圧レギュレータは、ADC_CRレジスタのADVREGENビットを1にすれば有効にできます。ソフトウェアは、較正を起動するかADCを有効化する前に、ADC電圧レギュレータの起動時間（$t_{ADCVREG_SETUP}$）だけ待つ必要があります。LDOステータスは、ADC_ISRレジスタのLDORDYビットを確認して検証できます。

ADCの動作が完了すると、ADCを無効化できます（ADEN = 0）。その後、ADC電圧レギュレータを無効にすることによって、さらに節電できます（セクション ：ADC電圧レギュレータの無効化シーケンスを参照してください）。

注： **内部電圧レギュレータを無効化した場合、内部アナログ較正係数がリセットされるため、較正を新しく実行する必要があります。**

ADC電圧レギュレータ有効化シーケンス

ADC電圧レギュレータを有効にするには、以下のシーケンスに従います。

1. ADC_ISRレジスタのLDORDYビットを1に設定してクリアします。
2. ADC_CRレジスタのADVREGENビットに1をセットします。
3. ADC_ISRレジスタのLDORDYが1になるまで待ちます（LDORDYはADC電圧レギュレータの起動時間後にセットされます）。これは、ADC_IERレジスタのLDORDYIEビットをセットすることによって割込みを有効にした場合、割込みによって処理できます。

ADC電圧レギュレータの無効化シーケンス

ADC電圧レギュレータを無効にするには、以下のシーケンスに従います。

1. ADCが無効であることを確認します（ADEN = 0）。
2. ADC_CRレジスタのADVREGENビットをクリアします。
3. ADC_ISRレジスタのLDORDYビットを1に設定してクリアします（オプション）。

34.4.4 較正（ADCAL）

ADCは較正機能を備えています。この手順時、ADCは、次のADCの電源オフまでADCに内部で適用される較正係数を計算します。アプリケーションは、較正中はADCを使用してはならず、較正が完了するまで待つ必要があります。

較正は、アナログデジタル変換を開始する前に実行する必要があります。較正を行うことで、プロセスのばらつきや、電源電圧、温度によりチップごとに異なることがあるオフセットエラーを除去できます。

ソフトウェアにより較正が開始されるようにするには、ADCALビットに1をセットします。次の条件がすべて満たされている場合のみ、較正を開始できます。

- ADC電圧レギュレータが有効化されている（ADVREGEN = 1かつLDORDY = 1）
- ADCが無効化されている（ADEN = 0）
- オートオフモードが無効化されている（AUTOFF = 0）

すべての較正シーケンス時、ADCALビットは1のままです。較正が完了すると、ハードウェアによってクリアされます。この後、較正係数をADC_DRレジスタ（ビット6 ～ 0）から読み出すことができます。

ADCが無効の場合（ADEN = 0）、内部アナログ較正が保持されます。ADC動作条件が変化したときには（V_{DDA} の変化がADCオフセットのばらつきの主因であり、温度変化はそれほど影響しません）、較正サイクルを再実行することが推奨されます。V_{REF+} 電圧の変化が10% を上回った場合、再較正することを推奨します。

以下の場合、較正係数は失われます。

- ADCの電源がオフ（たとえば、製品がSTANDBY またはV_{BAT} モードになった場合）。
- ADCペリフェラルがリセットされたとき。

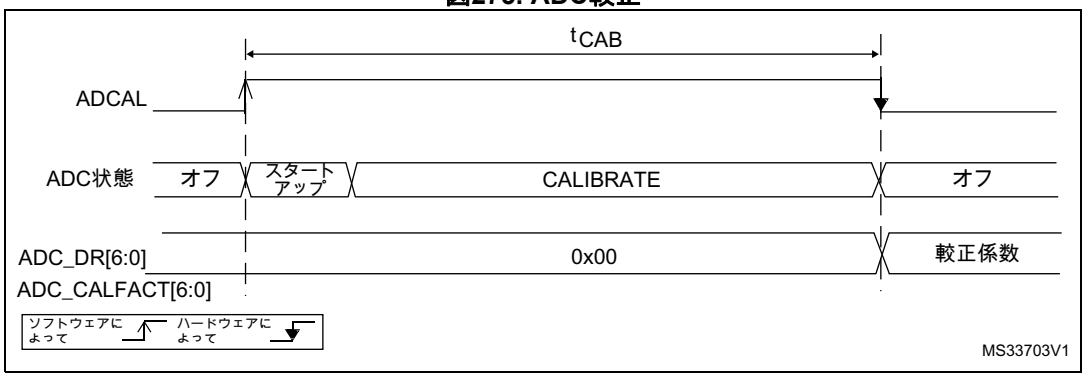
較正係数はADCの電源がオフになるたびに失われます（たとえば、製品がSTANDBY またはV_{BAT} モードになった場合）。しかし、ADCを再起動するときの時間を短縮するために、ソフトウェアによって較正係数を保存して復元することも可能です（ADCパワーダウン中の温度と電圧が安定している限り）。

ADCが有効であり、変換中でない場合（ADEN = 1かつADSTART = 0）、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログADCに自動的にインジェクトされます。この負荷は透過的であり、変換開始のサイクル遅延は増加しません。

ソフトウェア較正手順

1. ADEN = 0、ADVREGEN = 1、AUTOFF = 0、DPD = 0、かつDMAEN=0であることを確認します。
2. ADCAL = 1にセットします。
3. ADCAL= 0（またはEOCAL = 1）になるまで待ちます。これは、ADC_IERレジスタのEOCALIEビットをセットすることによって割込みを有効にした場合、割込みによって処理できます。
4. 較正係数をADC_DRまたはADC_CALFACTレジスタのビット6:0から読み出すことができます。

図273. ADC較正

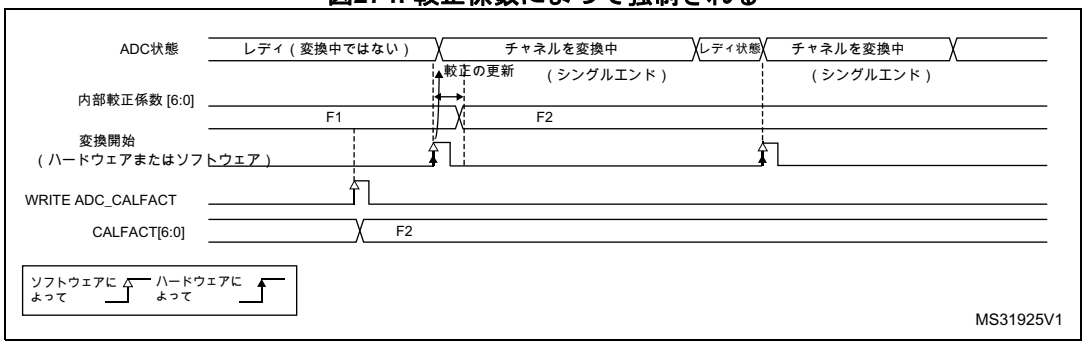


1. t_{CAB} の値については、デバイスのデータシートを参照してください。

較正係数によって強制されるソフトウェア手順

1. ADEN = 1かつADSTART = 0である（ADCが起動していて、変換の実行中でない）ことを確認します。
2. 保存済みの較正係数とともにADC_CALFACTを書き込みます。
3. 較正係数は、新しい変換が開始されるとすぐに使用されます。

図274. 較正係数によって強制される



34.4.5 ADCオン / オフ制御（ADEN、ADDIS、ADRDY）

パワーアップ時、ADCは無効になり、パワーダウンモードになります（ADEN = 0）。

図275 に示すように、ADCは正確な変換を開始する前に、安定時間t_{STAB} を必要とします。

ADCを有効化または無効化するには、2つの制御ビットが使用されます。

- ADCを有効にするには、ADEN = 1をセットします。ADCの動作準備ができると、ADRDYフラグがセットされます。
- ADCを無効にして、ADCをパワーダウンモードにするには、ADDIS = 1をセットします。ADCが完全に無効になると、ADENおよびADDISビットはハードウェアによって自動的にクリアされます。

その後、ADSTARTに1をセットすることによって（セクション 34.4.16： 1371ページの外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN）を参照）、またはトリガが有効な場合は外部トリガイベントが発生したときに、変換を開始できます。

ADCを有効にするには、次の手順を実行します。

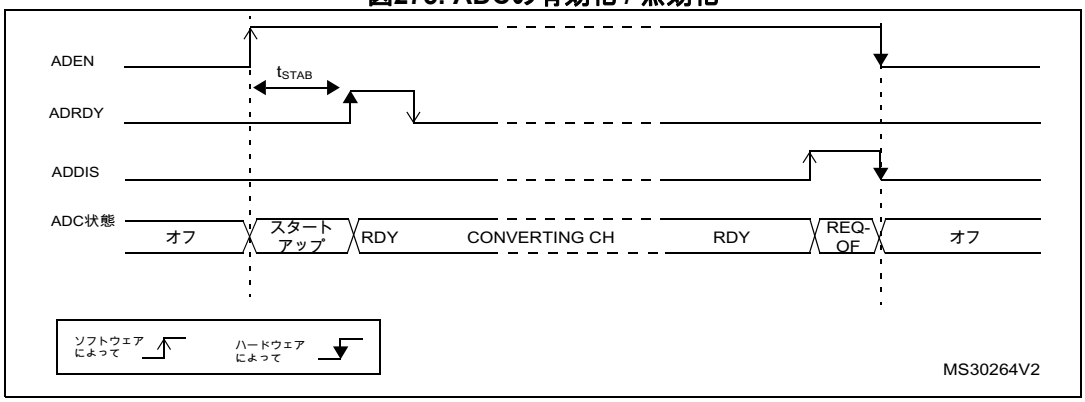
1. ADC_ISRレジスタのADRDYビットを1に設定してクリアします。
2. ADC_CRレジスタのADEN = 1をセットします。
3. ADC_ISRレジスタのADRDYが1になるまで待ちます（ADRDYはADC起動時間後にセットされます）。これは、ADC_IERレジスタのADRDYIEビットをセットすることによって割込みを有効にした場合、割込みによって処理できます。

ADCを無効にするには、次の手順を実行します。

1. ADC_CRレジスタでADSTART = 0になっているか確認して、変換が実行中でないことを確かめます。ソフトウェアトリガモードが使用されている場合、ソフトウェアトリガモードを停止します。ADC_CRレジスタのADSTPビットに1を書き込み、このビットで0が読み出されるまで待ってください。
2. ADC_CRレジスタのADDIS = 1をセットします。
3. アプリケーションによって必要とされる場合、ADC_CRレジスタのADENが0になり、ADCが完全に無効であることが示されるまで待ちます（ADEN = 0になると、ADDISは自動的にリセットされます）。

4. ADC_ISRレジスタのADRDYビットを1に設定してクリアします（オプション）。

図275. ADCの有効化 / 無効化

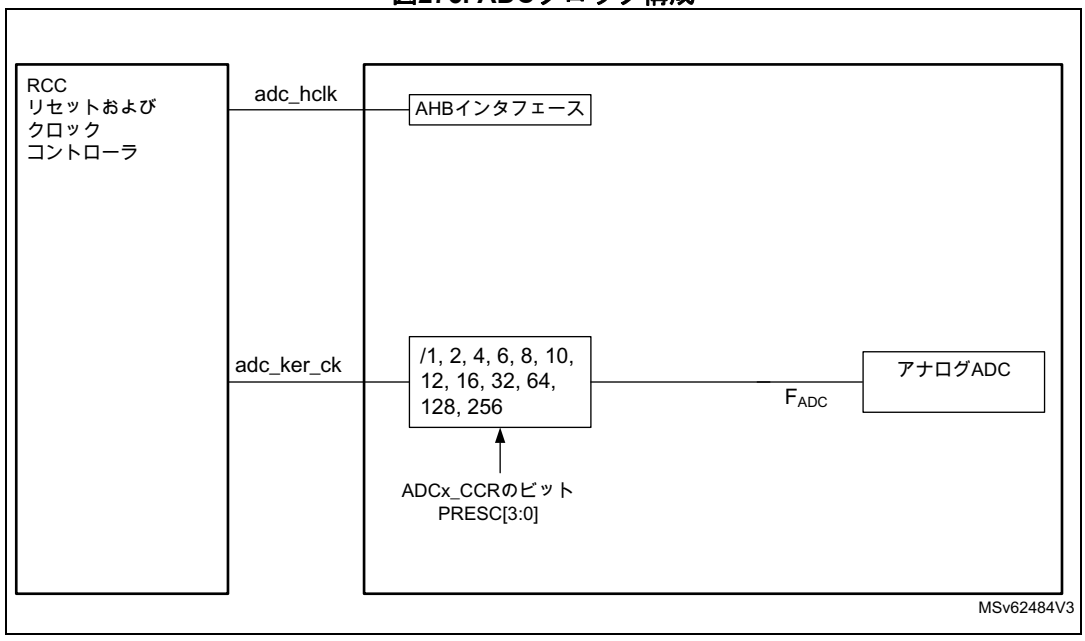


注： オートオフモード（AUTOFF = 1）の場合、パワーオン／オフフェーズはハードウェアによって自動的に実行され、ADRDYフラグはセットされません。

34.4.6 ADCクロック（PRESC[3:0]）

ADCはデュアルクロックドメインアーキテクチャを持つので、ADCにバスクロックから独立したクロック（ADC非同期クロック）を供給することができます。

図276. ADCクロック構成



1. バスクロックおよびADC非同期クロックを有効にする方法については、セクションリセットおよびクロック制御（RCC）を参照してください。

adc_ker_ck入力クロックは、次の2つの異なるクロックソースから選択できます（図 276 : ADCクロック構成を参照）。この選択は、RCCで行われます（詳細はRCCのセクションを参照）。

- ADC クロックは、バスクロックから独立し、これと同期される内部または外部クロックソースから供給できます。
- ADCクロックをバスクロックから取得するには、adc_ker_ckをバスクロックとして選択します。

オプションa）には、選択されたクロック構成にかかわらず、最大のADCクロック周波数に到達できるという利点があります。ADCクロックは、最終的には1、2、4、6、8、10、12、16、32、64、128、または256のプログラム可能な比率で分周できます。これは、ADCx_CCRレジスタのPRESC[3:0] ビットで設定します。

オプションb) には、クロック･ドメインの再同期を迂回するという利点があります。これは、ADCがタイマによってトリガされるときと、アプリケーションがADCの確実で精密なトリガを必要とする場合に便利です（そうしないと、トリガインスタンスの不確実性は、2つのクロック･ドメイン間の再同期によって高まります）。

表326.トリガから変換開始までの遅延(1)

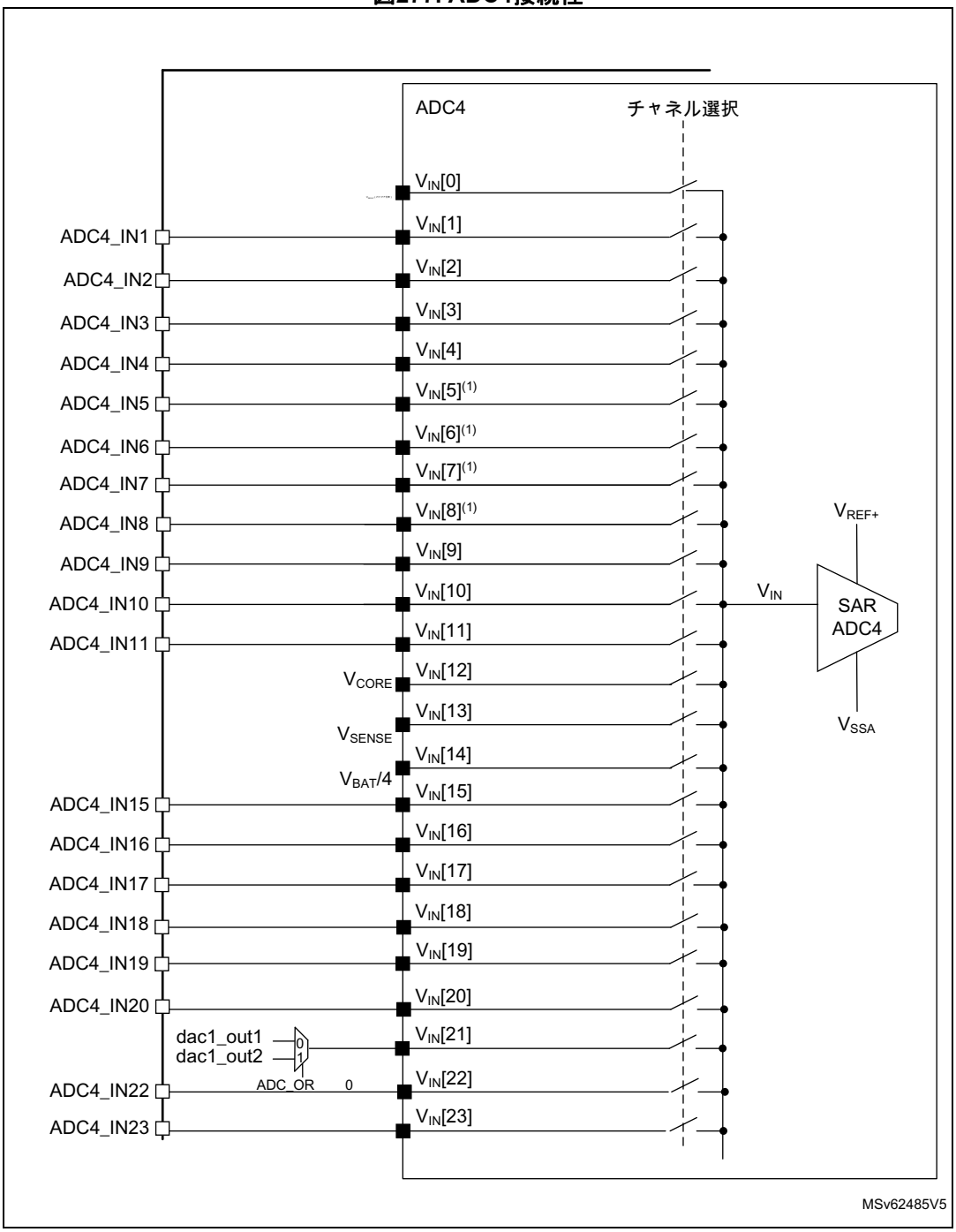
ADCクロックソース	トリガイベントから変換開始までの遅延
バスクロック以外のクロック	遅延は決定的ではありません（ジッタ）。
2分周されたバスクロック	遅延は決定的であり（ジッタではない）、4 ADCクロックサイクルに等しい。
4分周されたバスクロック	遅延は決定的であり（ジッタではない）、3.75 ADCクロックサイクルに等しい。
1分周されたバスクロック	遅延は決定的であり（ジッタではない）、4 ADCクロックサイクルに等しい。

1. 最大のF_{ADC} 周波数は、デバイスのデータシートを参照してください。

34.4.7 ADC接続性

ADC入力は、図277に示すように内部ソースに加えて外部チャネルにも接続されている。

図277. ADC4接続性



1. VIN[5] ~ VIN[8] はSTM32U535/545デバイスで予約済みです。

34.4.8 ADCの設定

ADCが無効化されている場合、ソフトウェアによりADC_CRおよびADC_PWRRレジスタのADCALおよびADENビットに書き込むことができます（ADENは0である必要があります）。

ADCが有効化されていて、ADCを無効化するための保留中のリクエストがない場合のみ（ADEN = 1かつADDIS = 0）、ソフトウェアによりADC_CRレジスタのADSTARTおよびADDISビットに書き込む必要があります。

ADC_IER ADC_CFGRi、ADC_SMPR、ADC_CHSELR、およびADC_CCRレジスタの他のすべての制御ビットについては、セクション 34.7：ADCレジスタに示す対応する制御ビットの記述内容を参照してください。ADCがソフトウェアトリガモードで動作している場合、ADC_CRレジスタのADSTPビットをセットし、ADSTPビットが0になるのを待ってから、上記のレジスタを再設定してください。

変換の実行中にADC_AWDTRiレジスタの内容を変更できます。

ADCが有効化されていて（おそらく変換中）、ADCを無効化するための保留中のリクエストがない場合のみ（ADSTART=1かつADDIS=0）、ソフトウェアによりADC_CRレジスタのADSTPビットに書き込む必要があります。

注： **上記のルールによって禁じられた書込み操作をソフトウェアが行わないようにするハードウェア保護はありません。そのような禁止された書込みアクセスが発生した場合、ADCは未定義の状態になることがあります。この場合に正しい動作を回復するには、ADCを無効にする必要があります（ADEN = 0とADC_CRレジスタのすべてのビットをクリアします）。**

34.4.9 チャネル選択 (CHSEL, SCANDIR, CHSELRMOD)

最大 25 の多重化チャネルがあります。

- GPIOピンからの 最大19 のアナログ入力（ADC_INx）
- 6 個の内部アナログ入力：温度センサ、内部基準電圧、V_{CORE}、V_{BAT} チャネル、DAC内部チャネル

単一チャネル、または一連のチャネルを変換できます。

変換されるチャネルのシーケンスは、ADC_CHSELRチャネル選択レジスタでプログラムできます。各アナログ入力チャネルに専用の選択ビットがあります（CHSELx）。

ADCスキャンシーケンサは2つのモードで使用ことができます。

- 完全には設定可能ではないシーケンサ

  チャネルがスキャンされる順序は、チャネル番号によって定義されます（CHSELRMODビットをADC_CFGR1レジスタでクリアする必要があります）。
  - シーケンスの長さはADC_CHSELRレジスタのCHSELxビットで設定します。
  - シーケンス方向：チャネルは、SCANDIRビットの値に応じて、前方（最小チャネル番号から最大チャネル番号へ）または後方（最大チャネル番号から最小チャネル番号へ）にスキャンされます（SCANDIR = 0：前方スキャン、SCANDIR = 1：後方スキャン）。
  - どのチャネルでもこれらのシーケンスに属すことができます。
- 完全に設定可能なシーケンサ

  CHSELRMODビットがADC_CFGR1レジスタにセットされます。
  - シーケンサの長さは最大で8チャネルです。
  - チャネルがスキャンされる順序はチャネル番号とは無関係です。どのような順序でもADC_CHSELRレジスタのSQ1[3:0] からSQ8[3:0] ビットで設定できます。
  - このシーケンスでは15のチャネルのみを選択できます（セクション 34.7.10：ADCチャネル選択レジスタ［オルタネート］（ADC_CHSELR）を参照）。
  - シーケンサがSQx[3:0] = 0b1111を検出すると、次のSQx[3:0] レジスタは無視されます。

– 0b1111がSQx[3:0] にプログラムされていない場合、シーケンサは8チャネル全部をスキャンします。

ソフトウェアでCHSEL、SCANDIR、CHSELRMODビットをプログラムできるのは、ADC_CRレジスタでADSTARTビットがクリアされている場合のみです。この場合は、変換が実行中でないことが保証されます。ADCがソフトウェアトリガモードで動作している場合、ADSTPビットをセットし、ADSTPビットが0になるのを待ってから、これらのレジスタを再設定してください。変換後にADSTARTビットが0にクリアされている場合でも、このシーケンスに従う必要があります。

温度センサ、DAC出力、V_{REFINT}、V_{BAT}、およびV_{CORE} 内部チャネル

温度センサ、内部DACチャネル、内部基準電圧（V_{REFINT}）、V_{BAT}、およびV_{CORE} はADC内部チャネルに接続されます。詳細については、セクション 34.4.2：A/Dコンバータのピンおよび内部信号のADC相互接続の表を参照してください。

34.4.10 プログラマブルサンプリング時間 (SMPx[2:0])

変換を開始する前に、ADCは測定する電圧ソースとADCの内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースがサンプルをチャージし、コンデンサを入力電圧レベルに保持できるだけの十分な長さが必要です。

プログラム可能なサンプリング時間によって、入力電圧ソースの入力抵抗に従って変換速度を微調整することができます。

ADCは、一定のADCクロックサイクル数だけ入力電圧をサンプリングしますが、この時間は、ADC_SMPRレジスタのSMP1[2:0] およびSMP2[2:0] ビットを使用して変更できます。

各チャネルは、ADC_SMPRレジスタのSMPSELxビットにより、SMP1[2:0] およびSMP2[2:0] ビットフィールドに設定された2つのサンプリング時間から1つ選択できます。

合計変換時間は、次のように計算されます。

t_{CONV} = サンプリング時間 + 12.5 x ADCクロックサイクル

例：

ADC_CLK = 16MHz、サンプリング時間 = 1.5 ADCクロックサイクル：

t_{CONV} = 1.5 + 12.5 = 14 ADCクロックサイクル = 0.875µs

ADCは、EOSMPフラグをセットすることによって、サンプリングフェーズの終了を示します。

I/Oアナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/Oアナログスイッチの抵抗が増加します。サンプリング時間は、状況に応じて調整する必要があります（対応する電気的特性については、デバイスのデータシートを参照）。この抵抗を低V_{DDA} 電圧で最小化するには、SYSCFG_CFGR1レジスタのBOOSTENビットで内部電圧ブースタを有効にするか、SYSCFG_CFGR1レジスタのANASWVDDビットでV_{DD} ブースタ電圧を選択します。

34.4.11 シングル変換モード（CONT = 0）

シングル変換モードでは、ADCは単一シーケンスの変換を実行して、すべてのチャネルを一度変換します。このモードは、ADC_CFGR1レジスタのCONTがクリアされたときに選択されます。変換は、次のいずれかによって開始されます。

- ADC_CRレジスタのADSTARTビットのセット
- ハードウェアトリガイベント

シーケンス内で、各変換の完了後、

- 変換されたデータは16bitのADC_DRレジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIEビットがセットされている場合、割込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIEビットがセットされている場合、割込みが生成されます。

次に、ADCは、新しい外部トリガイベントが発生するか、ADSTARTビットが再びセットされるまで停止します。

注： **単一チャネルを変換するには、長さが1のシーケンスをプログラムします。**

34.4.12 連続変換モード（CONT = 1）

連続変換モードでは、ソフトウェアまたはハードウェアトリガイベントが発生すると、ADCは一連の変換を実行して、すべてのチャネルを一度変換した後、自動的に再起動して、同じ変換シーケンスを連続的に実行します。このモードは、ADC_CFGR1レジスタのCONTに1がセットされたときに選択されます。変換は、次のいずれかによって開始されます。

- ADC_CRレジスタのADSTARTビットのセット
- ハードウェアトリガイベント

シーケンス内で、各変換の完了後、

- 変換されたデータは16bitのADC_DRレジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIEビットがセットされている場合、割込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIEビットがセットされている場合、割込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADCは変換シーケンスを連続的に繰り返します。

注： **単一チャネルを変換するには、長さが1のシーケンスをプログラムします。**

不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方をセットすることは禁じられています。

34.4.13 変換の開始（ADSTART）

ADSTART = 1にセットすることで、ソフトウェアによりADC変換が開始されます。

ADSTARTがセットされると、変換は、

- EXTEN = 00（ソフトウェアトリガ）の場合、すぐに開始します。
- EXTEN≠00の場合、選択されたハードウェアトリガの次のアクティブエッジで開始します。

ADSTARTビットは、ADCの動作が実行中かどうかを示すためにも使用されます。ADSTART = 0であり、ADCがアイドルであることを示しているときには、ADCを再設定できます。

ADSTARTビットは、ハードウェアによってクリアされます。

- ソフトウェアトリガによるシングルモードのとき（CONT = 0、EXTEN = 00）
  - 変換シーケンスの終了時に（EOS = 1）
- ソフトウェアトリガによる不連続モードのとき（CONT = 0、DISCEN = 1、EXTEN = 00）
  - 変換の終了時に（EOC = 1）
- すべての場合に（CONT = x、EXTEN = XX）
  - ソフトウェアによって起動されたADSTP手順の実行後（セクション 34.4.15：1371ページの実行中の変換の停止（ADSTP）を参照）

ADCが自律モードで動作している場合（DPDが1から0に遷移、セクション ：自律モード（AUTOFF、DPD）を参照）、ADSTARTビットはADCが起動されている状態でのみセットできます（LDORDY = 1かつADRDY = 1）。連続モード（CONT = 1）では、シーケンスは自動的に再起動されるので、EOSフラグがセットされたとき、ADSTARTビットはハードウェアによってクリアされません。

注：**シングルモードでハードウェアトリガが選択されたとき（CONT = 0かつEXTEN = 01）、ADSTARTは、EOSフラグのセット時にハードウェアによってクリアされません。このため、ハードウェアはADSTARTビットを再びセットする必要がなく、次のトリガイベントを見逃す恐れがありません。**

ソフトウェアによりADC変換モードがトリガされた後にADSTARTビットがクリアされる場合でも、ADSTPに1をセットし、ADSTPがクリアされるまで待ってから、ADCを再設定したり無効化したりする必要があります。

34.4.14 タイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計です。

$t_{CONV} = t_{SMPL} + t_{SAR} = [1.5 |min + 12.5 |12bit] \times t_{ADC_CLK}$

$t_{CONV} = t_{SMPL} + t_{SAR} = 42.9ns |min + 357.1ns |12bit = 0.400\mu s |min$（$f_{ADC_CLK}$ = 35MHz の場合）

図278. アナログ / デジタル変換時間

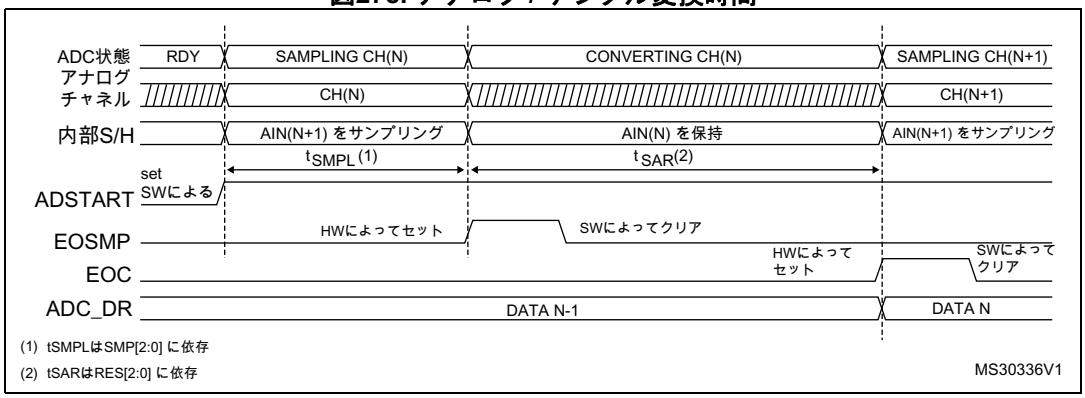
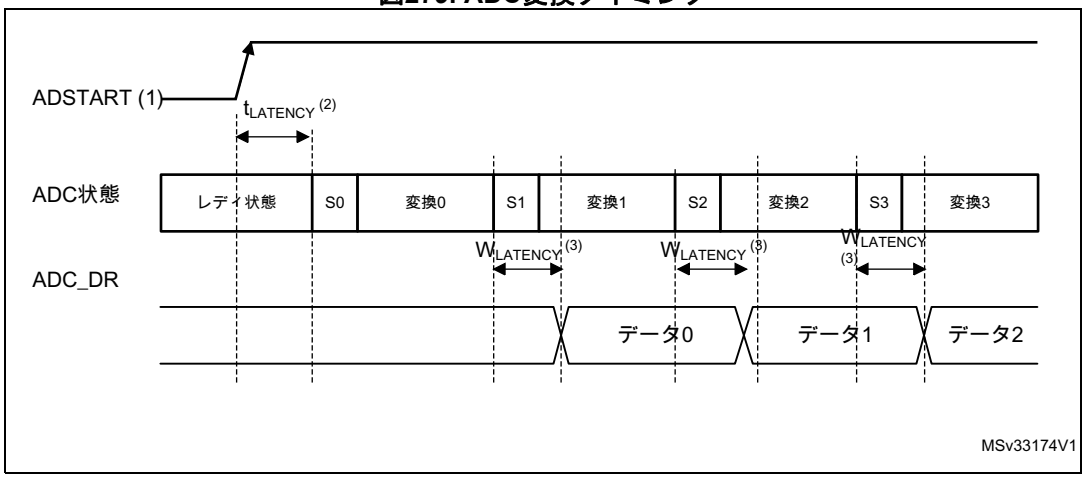


図279. ADC変換タイミング



1. EXTEN = 00またはEXTEN ≠ 00
2. トリガ遅延（詳細についてはデータシートを参照）
3. ADC_DRレジスタ書込み遅延（詳細についてはデータシートを参照）

34.4.15 実行中の変換の停止（ADSTP）

ソフトウェアは、ADC_CRレジスタのADSTPに1をセットすることによって、実行中の変換を停止することができます。

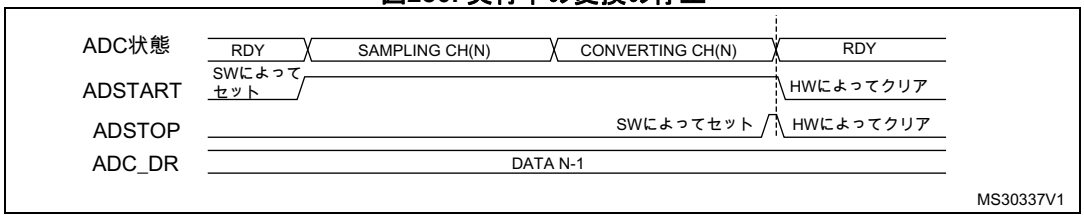
これによってADC動作がリセットされ、ADCはアイドルなり、新しい動作の準備ができます。

ADSTPビットがソフトウェアによってセットされると、実行中の変換は中止され、結果は破棄されます（ADC_DRレジスタは現在の変換で更新されません）。

スキャンシーケンスも中止され、リセットされます（ADCを再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTPおよびADSTARTビットの両方がハードウェアによってクリアされ、ソフトウェアは新しい変換を開始する前にADSTARTが0にクリアされるまで待つ必要があります。

図280. 実行中の変換の停止



34.4.16 外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN）

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャなど）によってトリガされます。EXTEN[1:0] 制御ビットが “0b00” に等しくない場合、外部イベントは選択された極性で変換をトリガできます。トリガ選択は、ソフトウェアによりビットADSTARTに1がセットされると有効になります。

変換中に発生したハードウェアトリガは無視されます。

ビットADSTARTがクリアされた場合、発生するハードウェアトリガは無視されます。

表327 にEXTEN[1:0] の値とトリガ極性の対応を示します。

表327. トリガ極性の設定

転送元	EXTEN[1:0]
トリガ検出は無効です。	00
立ち上がりエッジで検出します。	01
立ち下がりエッジで検出します。	10
立ち上がりエッジと立ち下がりエッジの両方で検出します。	11

注： **外部トリガの極性は、ADCが変換中でない（ADSTART = 0）ときだけ変更できます。**

EXTSEL[2:0] 制御ビットは、8つの可能なイベントのうち、変換をトリガするイベントを選択するために使用されます。

セクション 34.4.2 : A/Dコンバータのピンおよび内部信号レギュラ変換に使用可能なすべての外部トリガのリストについては、のADC相互接続の表を参照してください。

ソフトウェアソーストリガイベントは、ADC_CRレジスタのADSTARTビットをセットすることによって生成できます。

注： **トリガ選択は、ADCが変換中でない（ADSTART = 0）ときだけ変更できます。**

34.4.17 不連続モード（DISCEN）

このモードを有効にするには、ADC_CFGR1レジスタのDISCENビットをセットします。

このモード（DISCEN = 1）では、シーケンスで定義された各変換を開始するには、ハードウェアまたはソフトウェアトリガイベントが必要です。逆に、DISCENがクリアされている場合は、単一のハードウェアまたはソフトウェアトリガイベントにより、シーケンス内で定義されたすべての変換が連続的に開始されます。

例：

- DISCEN = 1で、変換されるチャネルが0、3、7、10
  - 最初のトリガ：チャネル0が変換され、EOCイベントが生成されます。
  - 2番目のトリガ：チャネル3が変換され、EOCイベントが生成されます。
  - 3番目のトリガ：チャネル7が変換され、EOCイベントが生成されます。
  - 4番目のトリガ：チャネル10が変換され、EOCイベントとEOSイベントの両方が生成されます。
  - 5番目のトリガ：チャネル0が変換され、EOCイベントが生成されます。
  - 6番目のトリガ：チャネル3が変換され、EOCイベントが生成されます。
  - ...
- DISCEN = 0で、変換されるチャネルが0、3、7、10
  - 最初のトリガ：シーケンス全体、すなわち、チャネル0、3、7、および10が変換されます。各変換後にEOCイベントが生成され、最後の変換後にはEOSイベントも生成されます。
  - 後続のトリガイベントがあると、シーケンス全体が再開されます。

注： **不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方をセットすることは禁じられています。**

34.4.18 プログラム可能な分解能（RES）- 高速変換モード

ADCの分解能を下げることによって、高速な変換時間（t_{SAR}）が可能になります。

分解能は、ADC_CFGR1レジスタのRES[1:0] ビットをプログラムすることによって、12、10、8、または6bitに設定できます。分解能を下げることによって、高いデータ精度を必要としないアプリケーションの変換時間を高速にできます。

注： **RES[1:0] ビットは、ADENビットがリセットされたときだけ変更する必要があります。**

変換結果は常に12bit幅であり、未使用のLSBビットはゼロとして読み出されます。

分解能を下げると、表328 に示すように、逐次比較ステップに必要な変換時間が短縮されます。

表328.t_{SAR} タイミングは分解能に依存

RES[1:0] ビット	t_{SAR} (ADC クロック サイクル)	t_{SAR} (ns) (f_{ADC} = 35MHz時)	$t_{SMPL\ (min)}$ (ADCクロック サイクル)	t_{CONV} (ADCクロック サイクル) (min. t_{SMPL})	t_{CONV} (ns) (f_{ADC} = 35MHz時)
12	12.5	357	1.5	14	400
10	10.5	300	1.5	12	343
8	8.5	243	1.5	10	286
6	6.5	186	1.5	8	229

34.4.19 変換の終了、サンプリングフェーズの終了（EOC、EOSMPフラグ）

ADCは、各変換の終了（EOC）イベントを示します。

ADCは、新しい変換データ結果がADC_DRレジスタで使用可能になるとすぐに、ADC_ISRレジスタのEOCフラグをセットします。ADC_IERレジスタのEOCIEビットがセットされている場合、割込みが生成されます。EOCフラグは、ソフトウェアによって1を書き込むことによって、またはADC_DRレジスタを読み出すことによってクリアされます。

ADCは、ADC_ISRレジスタのEOSMPフラグをセットすることによって、サンプリングフェーズの終了も示します。EOSMPフラグは、ソフトウェアによって1を書き込むことによってクリアされます。ADC_IERレジスタのEOSMPIEビットがセットされている場合、割込みが生成されます。

この割込みの目的は、処理を変換と同期するためです。一般に、アナログマルチプレクサには変換フェーズ中の隠された時間でアクセスできるため、マルチプレクサは次のサンプリング開始時に位置づけられます。

注： **サンプリングの終了から変換の終了までは非常に短時間しか残されていないため、割込みとWFI命令ではなく、ポーリングまたはWFE命令を使用することが推奨されます。**

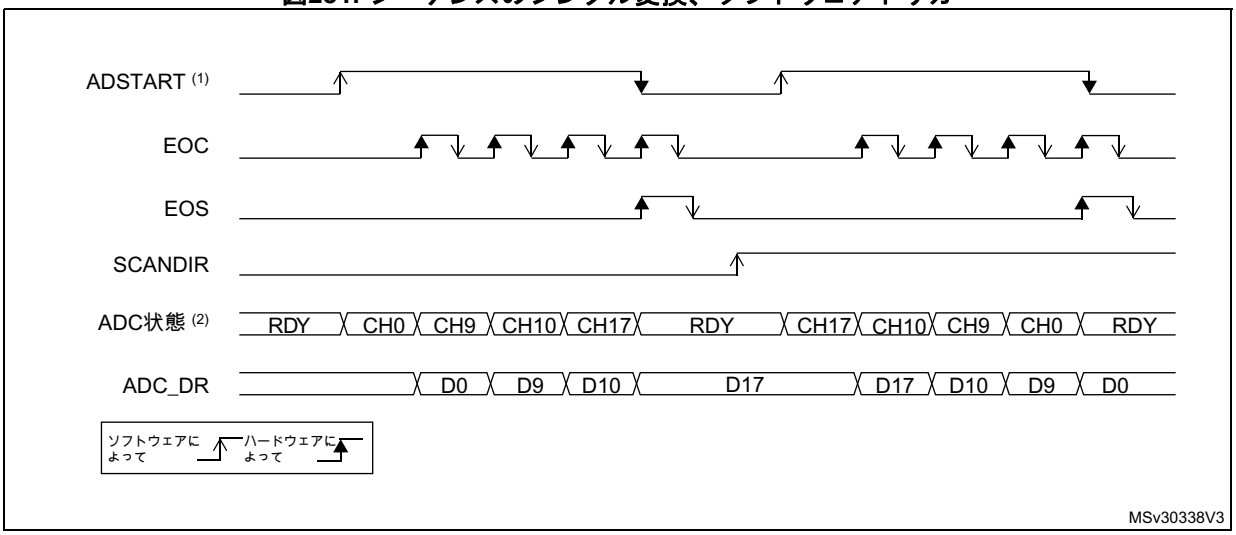
34.4.20 変換シーケンスの終了（EOSフラグ）

ADCは各シーケンスの終了（EOS）イベントをアプリケーションに通知します。

ADCは、変換の最後のデータ結果がADC_DRレジスタで使用可能になるとすぐに、ADC_ISRレジスタのEOSフラグをセットします。ADC_IERレジスタのEOSIEビットがセットされている場合、割込みが生成されます。EOSフラグは、ソフトウェアによって1を書き込むことによってクリアされます。

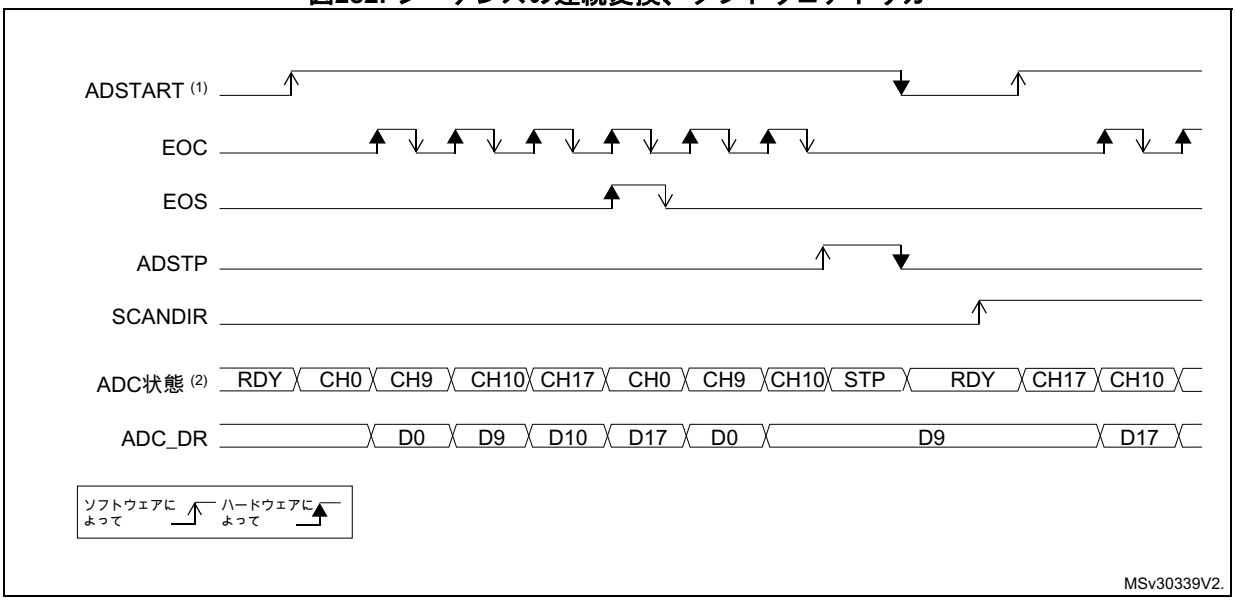
34.4.21 タイミング図の例（シングル／連続モード、ハードウェア／ソフトウェアトリガ）

図281. シーケンスのシングル変換、ソフトウェアトリガ



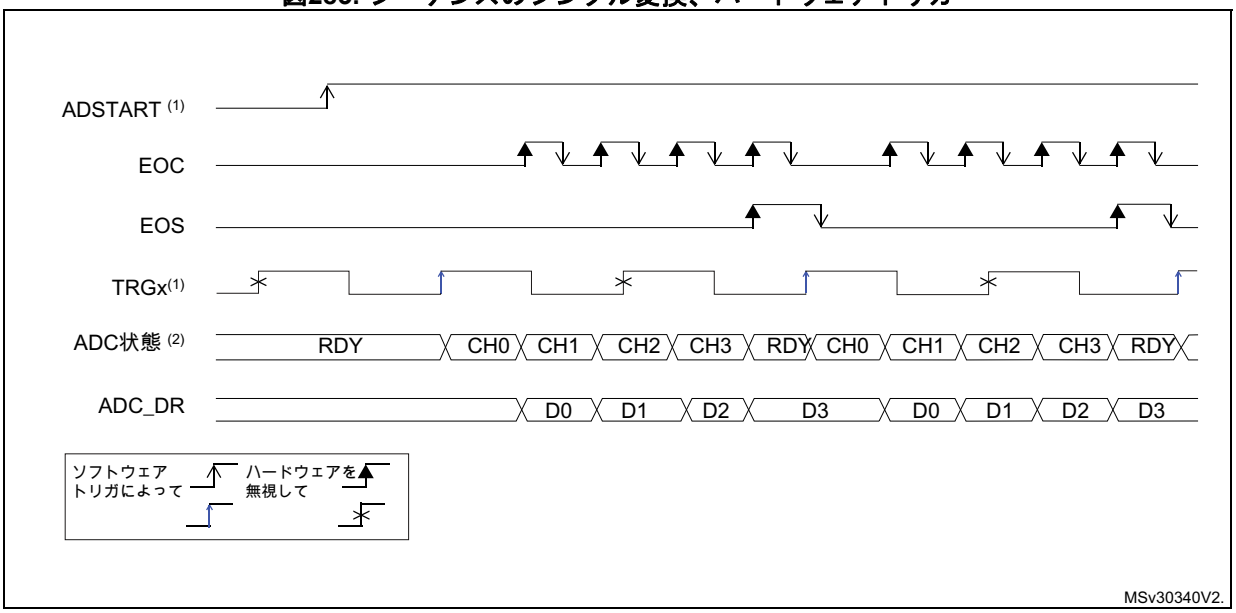
1. EXTEN = 00、CONT = 0
2. CHSEL = 0x20601、WAIT = 0、AUTOFF = 0。

図282. シーケンスの連続変換、ソフトウェアトリガ



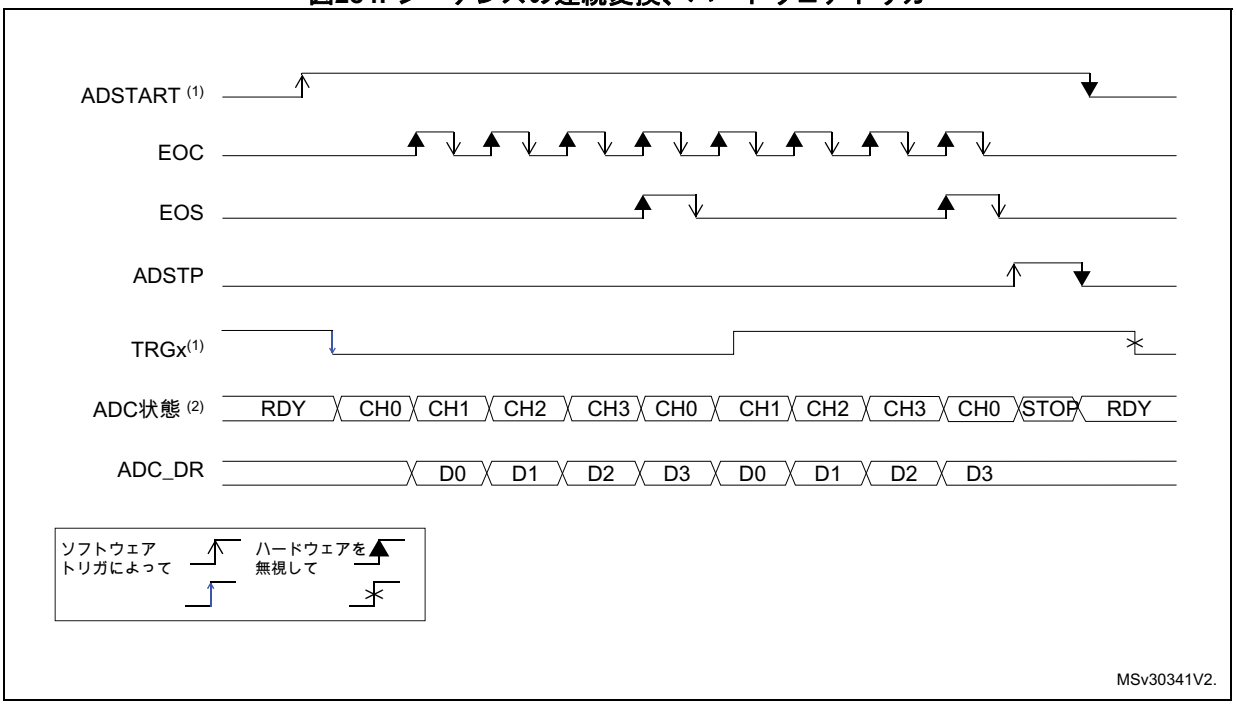
1. EXTEN = 00、CONT = 1。
2. CHSEL = 0x20601、WAIT = 0、AUTOFF = 0。

図283. シーケンスのシングル変換、ハードウェアトリガ



1. EXTSEL = TRGx（オーバー周波数）、EXTEN = 01（立ち上がりエッジ）、CONT = 0。
2. CHSEL = 0xF、SCANDIR = 0、WAIT = 0、AUTOFF = 0。

図284. シーケンスの連続変換、ハードウェアトリガ



1. EXTSEL = TRGx、EXTEN = 10（立ち下がりエッジ）、CONT = 1。
2. CHSEL = 0xF、SCANDIR = 0、WAIT = 0、AUTOFF = 0。

34.4.22 低周波数トリガモード

アプリケーションが最大のtIDLE 値（シングル変換モードにおける1つのトリガからもう1つのトリガまでの間、またはADCの有効化と最初のADC変換の間）より長い時間をサポートしなければならない場合、ADC内部状態の再設定が必要になります。このメカニズムは、ADC_CFGR2レジスタ のLFTRIGビットに1をセットすれば有効になります。このビットをセットすることにで、どのようなトリガでも（ソフトウェアまたはハードウェア）ADCにリアームコマンドを送ることができます。変換は、LFTRIGが0にセットされてから2 ADCクロックサイクル後に開始されます。

AUTOFFビットが1のときはこのモードを使う必要はありません。ウェイトモードの場合、内部リアームコマンドを生成できるのは最初のトリガだけです。

34.4.23 データ管理

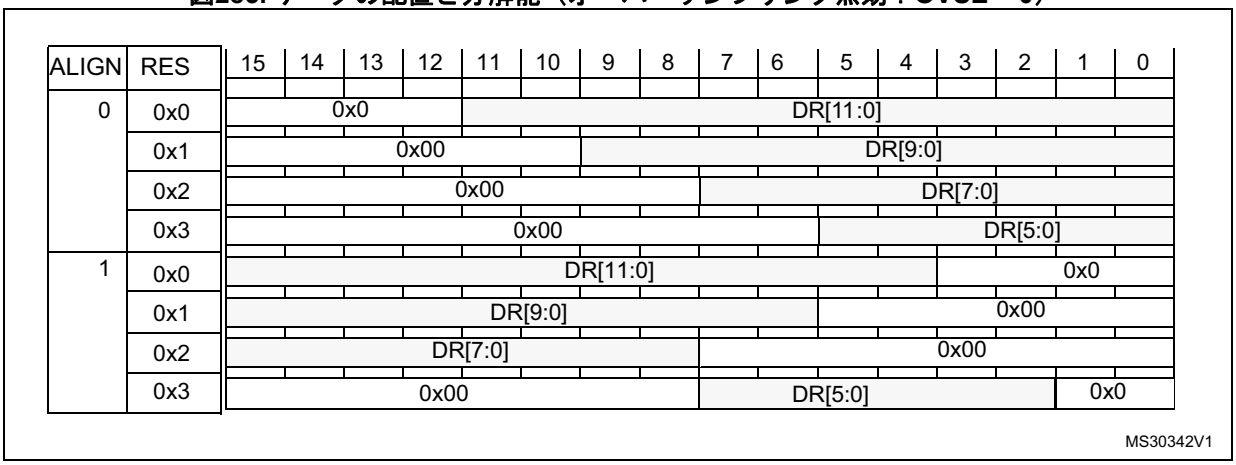
データレジスタおよびデータの配置（ADC_DR、ALIGN）

各変換の終了時（EOCイベントの発生時）、変換されたデータの結果は16bit幅のADC_DRデータレジスタに格納されます。

ADC_DRのフォーマットは、設定されたデータ配置と分解能に依存します。

ADC_CFGR1レジスタのALIGNビットは、変換後に格納されるデータの配置を選択します。図285に示すように、データは右詰め（ALIGN = 0）または左詰め（ALIGN = 1）にできます。

図285. データの配置と分解能（オーバーサンプリング無効：OVSE = 0）



ADCオーバーラン（OVR、OVRMOD）

オーバーランフラグ（OVR）は、新しい変換からのデータが使用可能になる前に、変換されたデータがCPUまたはDMAによって時間内に読み出されなかったときに、データオーバーランを示します。

OVRフラグは、新しい変換が完了した時点でEOCフラグが1のままであった場合に、ADC_ISRレジスタでセットされます。ADC_IERレジスタのOVRIEビットがセットされている場合、割込みが生成されます。

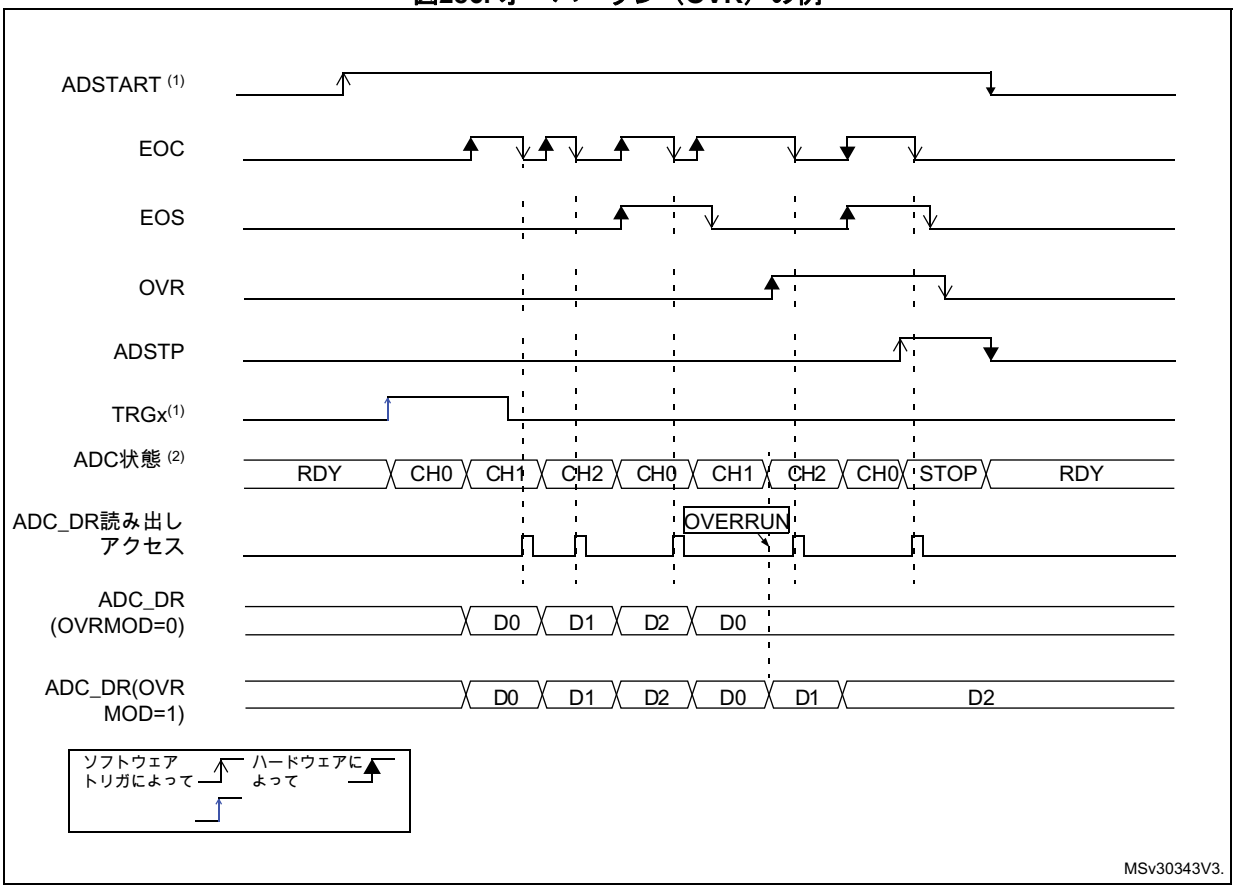
オーバーラン条件が発生すると、ADCは動作し続け、ソフトウェアがADC_CRレジスタのADSTPビットをセットすることによってシーケンスの停止とリセットを決めるまで、変換を続行できます。

OVRフラグは、ソフトウェアによって1を書き込むことによってクリアされます。

ADC_CFGR1レジスタのOVRMODビットをプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD = 0
  - オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータは保持され、新しい変換は破棄されます。OVRが1のままの場合、さらに変換を実行できますが、結果データは破棄されます。
- OVRMOD = 1
  - データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。OVRが1のままの場合、さらに変換を実行でき、ADC_DRレジスタは常に最新の変換からのデータを含みます。

図286. オーバーラン（OVR）の例



DMAを使用しない変換データシーケンスの管理

変換が十分に遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアはEOCフラグと関連の割込みを使用して、各データ結果を処理する必要があります。変換が完了するたびに、ADC_ISRレジスタのEOCビットがセットされ、ADC_DRレジスタを読み出すことができます。オーバーラインイベントをエラーとして管理するには、ADC_CFGR1レジスタのOVRMODビットに0を設定する必要があります。

オーバーランなしでのDMAを使用しない変換データの管理

変換のたびにデータの読出しをせずにADCに1つまたは複数のチャネルを変換させると便利な場合があります。この場合、OVRMODビットに1を設定する必要があり、ソフトウェアはOVRフラグを無視する必要があります。OVRMODに1がセットされているとき、オーバーランイベントが発生してもADCは変換を続行し、ADC_DRレジスタには常に最新の変換データが格納されます。

DMAを使用した変換データの管理

変換されたレギュラチャネルの値はすべて、単一のデータレジスタに格納されるので、複数のチャネルを変換するときにはDMAを使用すると効率的です。これによって、ADC_DRレジスタに格納されている変換データ結果が失われるのを避けることができます。

DMAモードが有効なとき（ADC_CFGR1レジスタのDMAENビットが1にセットされている）、各チャネルの変換後、DMAリクエストが生成されます。これにより、変換されたデータをADC_DRレジスタからソフトウェアで選択した場所へ転送することができます。

注： **ADC_CFGR1レジスタのDMAENビットは、ADC較正フェーズ後にセットする必要があります。**

これにもかかわらず、DMAがDMA転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR = 1）、ADCはDMAリクエストの生成を停止し、新しい変換に対応するデータはDMAによって転送されません。つまり、RAMへ転送されたデータはすべて有効とみなすことができます。

OVRMODビットの設定に応じて、データは保存または上書きされます（セクション ： 1377ページのADCオーバーラン（OVR、OVRMOD）を参照してください）。

DMA転送リクエストは、ソフトウェアがOVRビットをクリアするまでブロックされます。

アプリケーションの用途に応じて2つのDMAモードがあり、ADC_CFGR1レジスタのビットDMACFGで設定されます。

- DMAワンショットモード（DMACFG = 0）。
  DMAが固定数のデータワードを転送するようにプログラムされたときには、このモードを選択してください。
- DMAサーキュラモード（DMACFG = 1）
  DMAをサーキュラモードまたはダブルバッファモードでプログラムするときには、このモードを選択してください。

DMAワンショットモード（DMACFG = 0）

このモードでは、ADCは新しい変換データワードが使用可能になるたびにDMA転送リクエストを生成し、変換が再び開始された場合でも、DMAが最後のDMA転送に達すると（転送完了割込みが発生すると）（DMAセクションを参照）、DMAリクエストの生成を停止します。

DMA転送完了後（DMAコントローラで設定されたすべての転送後）：

- ADCデータレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMAコントローラに対する新しいDMAリクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMAは停止します。

DMAサーキュラモード（DMACFG = 1）

このモードでは、DMAが最後のDMA転送に達した場合でも、ADCは新しい変換データワードがデータレジスタで使用可能になるたびにDMA転送リクエストを生成します。これにより、連続的なアナログ入力データ·ストリームを処理するようにDMAをサーキュラモードに設定できます。

34.4.24 低電力機能

ウェイト変換モードは（WAIT）

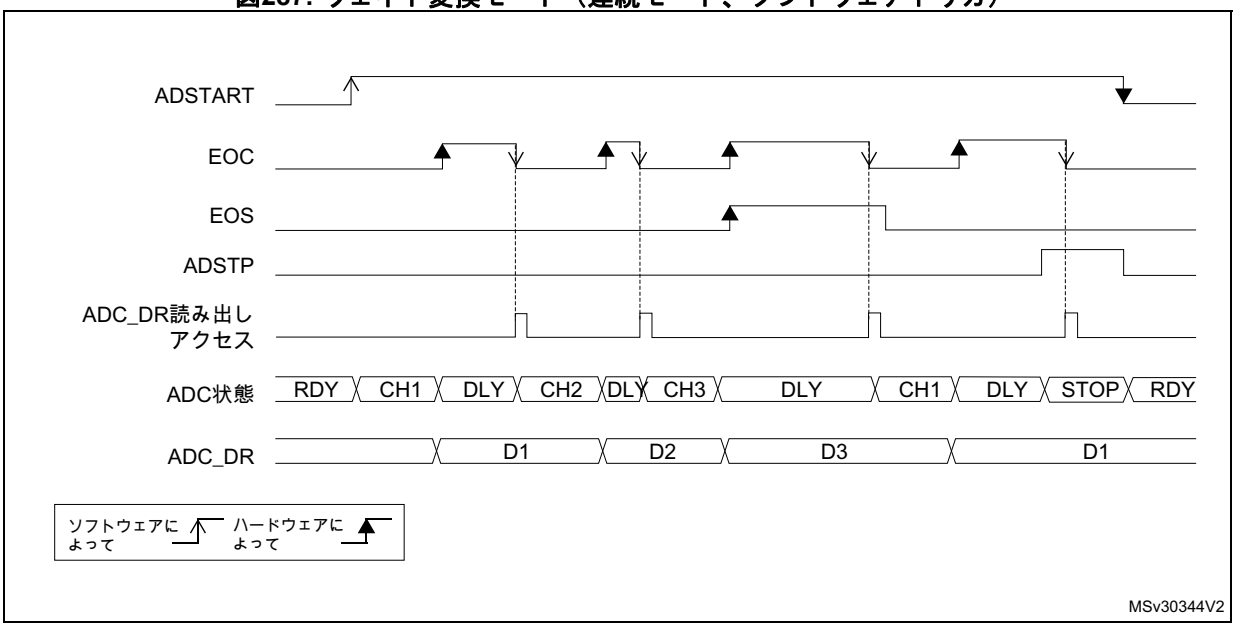
ウェイト変換モードを使用すると、ソフトウェアを単純化するだけでなく、ADCオーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化できます。

ADC_CFGR1レジスタのWAITビットが1にセットされているとき、新しい変換は、前のデータが処理された場合、ADC_DRレジスタが読み出された場合、またはEOCビットがクリアされた場合のみ開始できます。

これは、ADCの速度をデータを読み出すシステムの速度に自動的に適応させる方法です。

注： **変換中または読出しアクセス前のウェイト時間中に発生したハードウェアトリガは無視されます。**

図287. ウェイト変換モード（連続モード、ソフトウェアトリガ）



1. EXTEN = 00、CONT = 1
2. CHSEL= 0x3、SCANDIR = 0、WAIT = 1、AUTOFF = 0。

ADC省電力モード

ADCには、省電力モードとして、オートオフモードと自律モードの2つが組み込まれています。

オートオフモード（AUTOFF）

オートオフモードは、ADC_PWRRレジスタのAUTOFFビットに1をセットすることで有効になります。

以下に、オートオフモードの動作シーケンスを示します。

1. AUTOFFに1がセットされている場合、変換が実行中でなければ、ADCは常に電源がオフになります。
2. そして、ソフトウェアまたはハードウェアによって変換がトリガされると自動的にウェイクアップし、トリガイベントとADCサンプリング時間の間に起動時間が挿入されます。
3. 変換シーケンスが完了すると、ADCは自動的に無効になります。
4. ハードウェアまたはソフトウェアのトリガが連続して発生すると、ADCが自動的に有効化され、変換が処理されます。

オートオフモードの状態図については、図288 を参照してください。

オートオフモードを使用すると、限られた数の変換しか行われない、または変換リクエストがまれにしか発生しないため、（低周波数ハードウェアトリガなどにより）ADCのオン／オフを切り替えるための電力と時間を使う意味があるようなアプリケーションにおいて、電力消費を大幅に削減することができます。

低周波数のクロックで動作するアプリケーションの場合、オートオフモードとウェイトモード（WAIT = 1）を組み合わせることができます。この組み合わせでは、ADCがウェイトフェーズで自動的に電源オフされ、ADC_DRレジスタがアプリケーションによって読み出されるとすぐに再起動する場合に、大幅に電力が削減できます（図 289：WAIT = 0かつAUTOFF = 1でのADCの動作と図 290：WAIT = 1かつAUTOFF = 1でのADCの動作を参照してください）。

オートオフモードは、低消費電力バックグラウンド自律モード（LPBAM）に対応しています。

注： **専用の内部オシレータを管理する方法については、リセットおよびクロック制御（RCC）のセクションを参照してください。ADCインタフェースは、節電のために、これらの内部オシレータの オン／オフを自動的に切り替えることができます。**

図288. オートオフモードの状態図

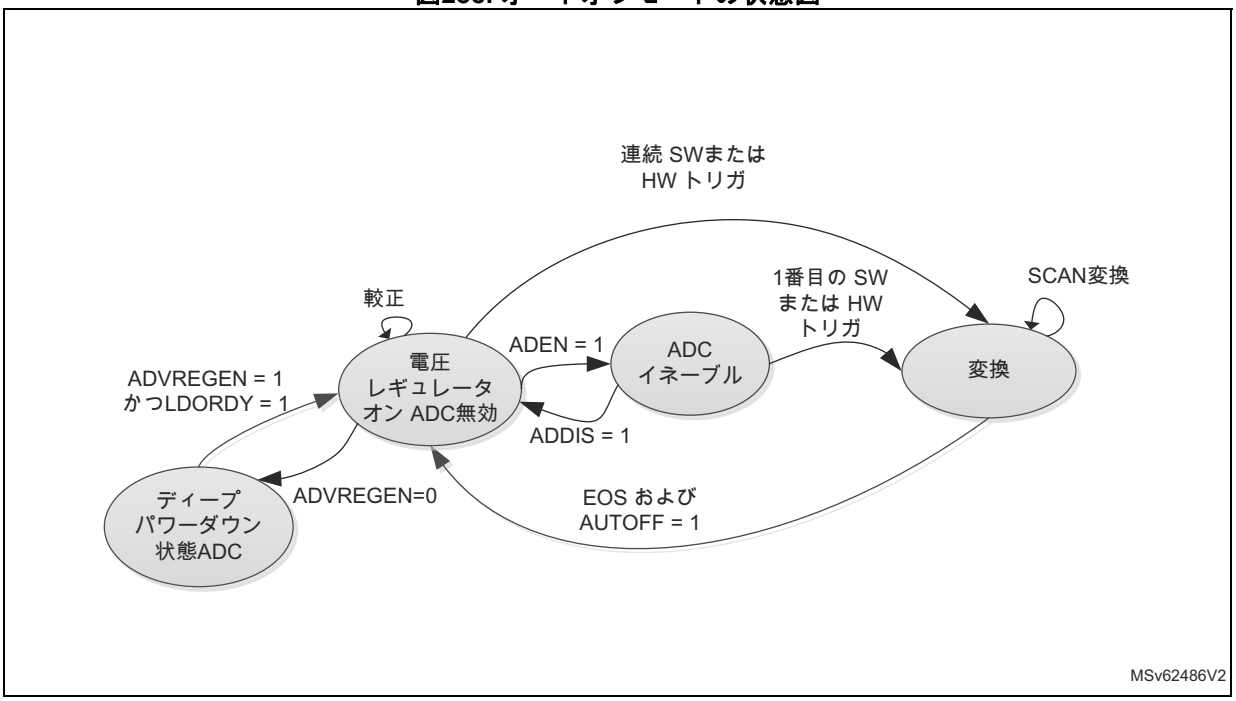
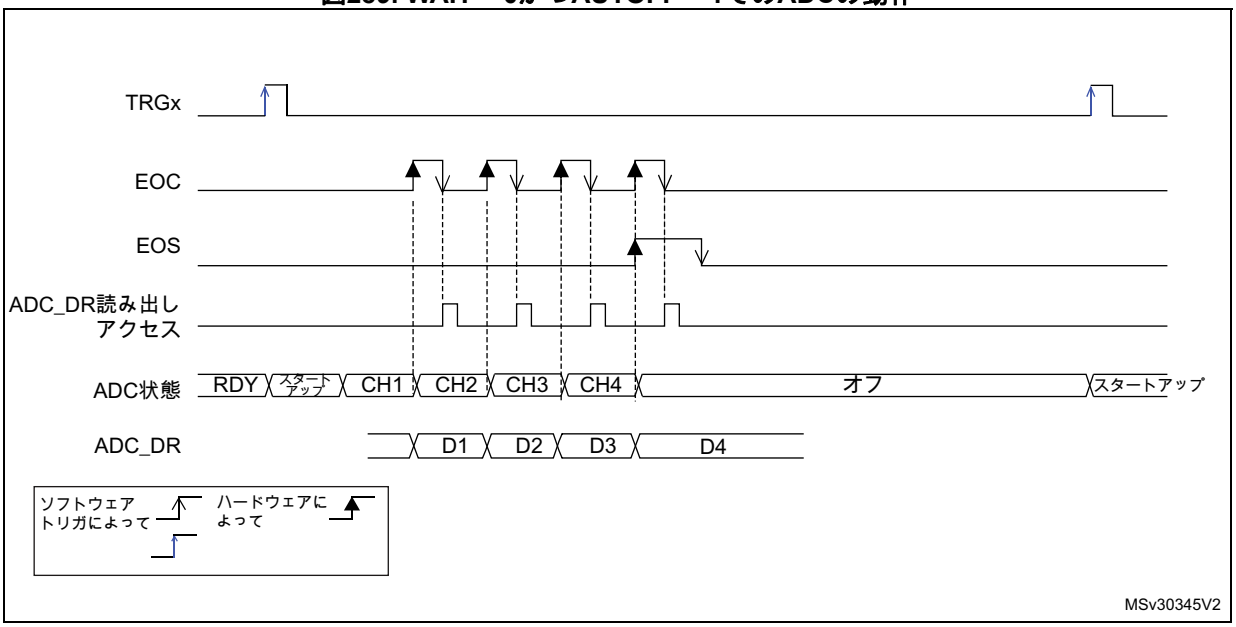
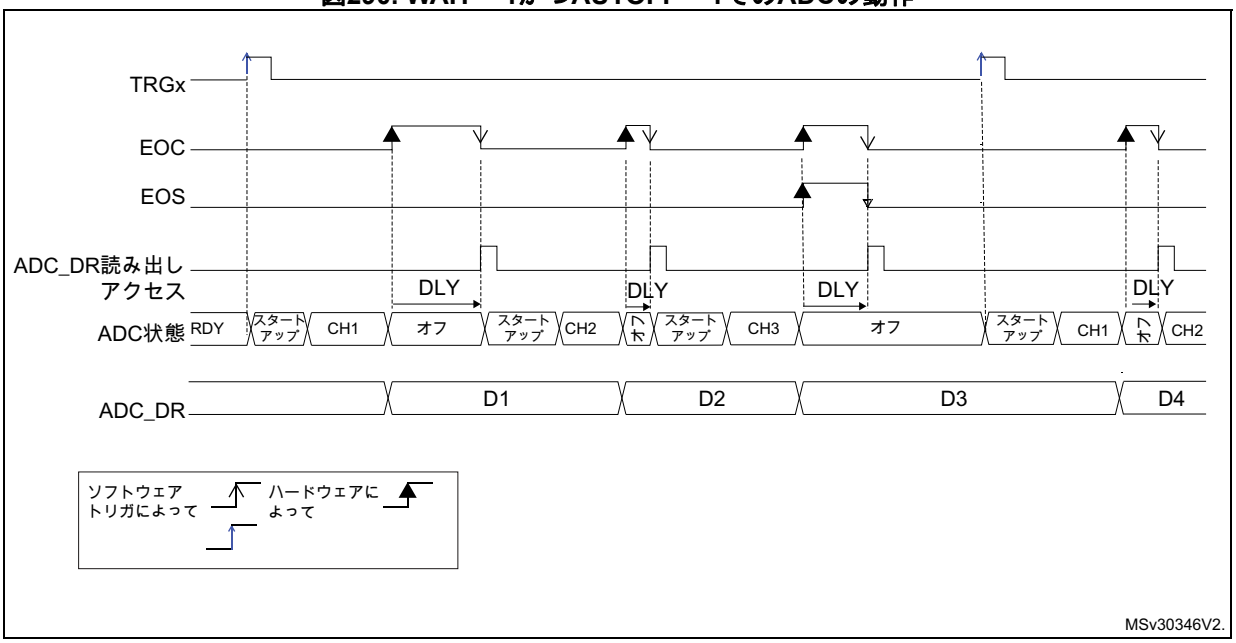


図289. WAIT = 0かつAUTOFF = 1でのADCの動作



1. EXTSEL = TRGx、EXTEN = 01（立ち上がりエッジ）、CONT = x、ADSTART = 1、CHSEL = 0xF、SCANDIR = 0、WAIT = 0、AUTOFF = 1。

図290. WAIT = 1かつAUTOFF = 1でのADCの動作



1. EXTSEL = TRGx、EXTEN = 01（立ち上がりエッジ）、CONT = x、ADSTART = 1、CHSEL = 0xF、SCANDIR = 0、WAIT = 1、AUTOFF = 1。

自律モード（AUTOFF、DPD）

自律モードを有効化するには、ADC_PWRRレジスタのAUTOFFビットとDPDビットの両方に1をセットします。また、RCCでも自律モードを有効化する必要があります。

以下に、自律モードの動作シーケンスを示します。

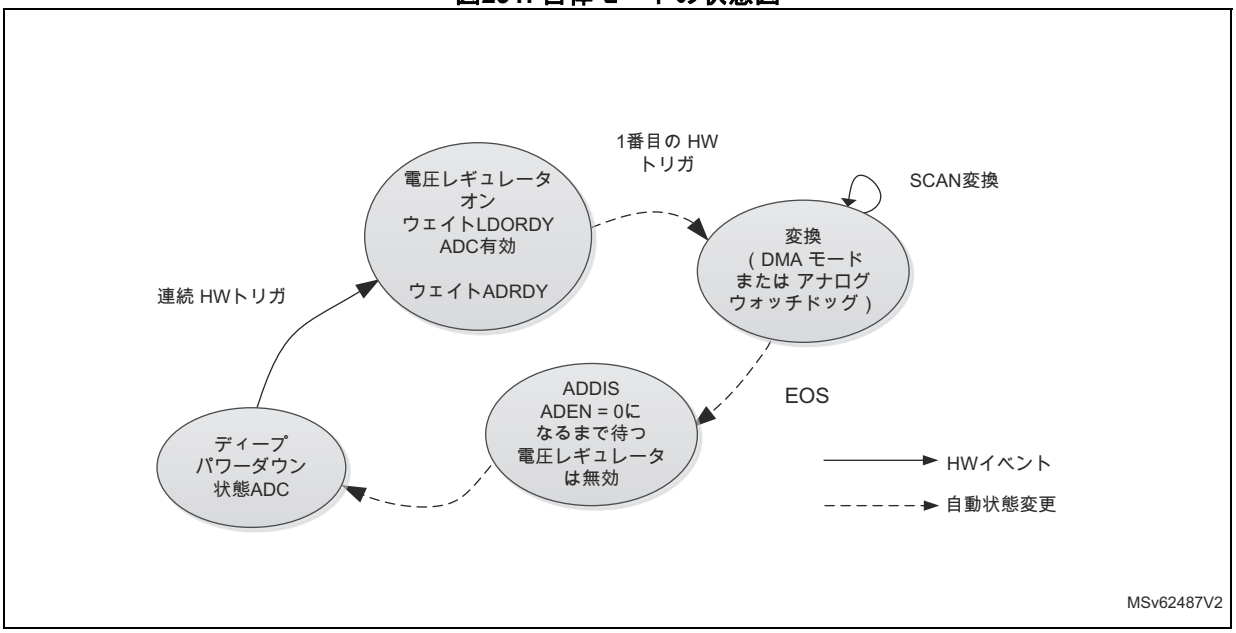
1. AUTOFFとDPDの両方に1がセットされている場合、変換が実行中でなければ、ADCは電源がオフになります。
2. ハードウェアトリガを受信すると、ADCからRCCにadc_ker_ckおよびadc_hclkクロックがリクエストされ、ADC電圧レギュレータが有効化され、較正係数がロードされ、ADCが有効化され、変換が開始します。
3. ADCの変換が完了すると、ペリフェラル設定に応じて、ADCによりAWDx割込みまたはDMAリクエストが生成されます。
   - DMAモードが有効化されている場合、ADCにより、メモリまたは他のペリフェラルにデータを転送するためのDMAリクエストが生成されます。
   - アナログウォッチドッグが有効化されている場合、ADCデータを転送する必要はありません。アナログウォッチドッグにより、データとプログラムされた閾値が比較され、データが閾値の下限を下回っているか、上限を上回っている場合にデバイスをウェイクアップするためのAWDx割込みが生成されます。
4. ADCの変換／シーケンスまたは変換が完了すると、ADCおよびADC電圧レギュレータ、V_{REFINT}バッファ、温度センサが自動的に無効化され、それ以降のクロックリクエストがネゲートされます。これにより、電流消費が最小限に抑えられます。
5. ハードウェアのトリガが連続して発生すると、ADCが自動的に有効化され、変換が処理されます。

自律モードの状態図については、図291を参照してください。

自律モードでは、デバイスがSTOPモードの状態でもADCペリフェラルの動作が可能です。ただし、RUNモードまたはSLEEPモードでも使用できます。

低消費電力バックグラウンド自律モード（LPBAM）に対応しています。

図291. 自律モードの状態図



34.4.25 アナログウィンドウウォッチドッグ

3つのAWDアナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

アナログウォッチドッグ1の説明

AWD1アナログウォッチドッグを有効にするには、ADC_CFGR1レジスタのAWD1ENビットをセットします。図292に示すように、選択された1つのチャネルまたはすべての有効チャネル （表 330：アナログウォッチドッグ1チャネル選択を参照）が設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視するために使用されます。

アナログウォッチドッグ（AWD1）ステータスビットは、ADCによって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。これらの閾値は、ADC_AWD1TRレジスタのHT1[11:0] ビットおよびLT1[11:0] ビットでプログラムされます。ADC_IERレジスタのAWD1IEビットをセットすることによって、割込みを有効にできます。

AWD1フラグは、ソフトウェアによって1を書き込むことによってクリアされます。

12bit未満の分解能でデータを変換するときには（DRES[1:0] ビットに従って）、内部比較は常に12bit全体の元の変換データ（左詰め）に対して実行されるため、プログラムされた閾値のLSBはクリアされたままである必要があります。

表329 に、可能なすべての分解能での比較方法を示します。

表329.アナログウォッチドッグ比較

分解能ビット RES[1:0]	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：12bit	DATA[11:0]	LTx[11:0] およびHTx[11:0]	-
01：10bit	DATA[11:2],00	LTx[11:0] およびHTx[11:0]	ユーザはLTx[1:0] とHTx[1:0] を “00” に設定する必要があります。
10：8bit	DATA[11:4],0000	LTx[11:0] およびHTx[11:0]	ユーザはLTx[3:0] とHTx[3:0] を “0000” に設定する必要があります。
11：6bit	DATA[11:6],000000	LTx[11:0] およびHTx[11:0]	ユーザはLTx[5:0] とHTx[5:0] を “000000” に設定する必要があります。

1. 配置計算の前に、元の変換データに対するウォッチドッグ比較が行われます。

表330に、ADC_CFGR1レジスタのAWD1SGLおよびAWD1ENビットを設定して、1つ以上のチャネルに対してアナログウォッチドッグを有効にする方法を示します。

図292. アナログウォッチドッグによって保護される領域

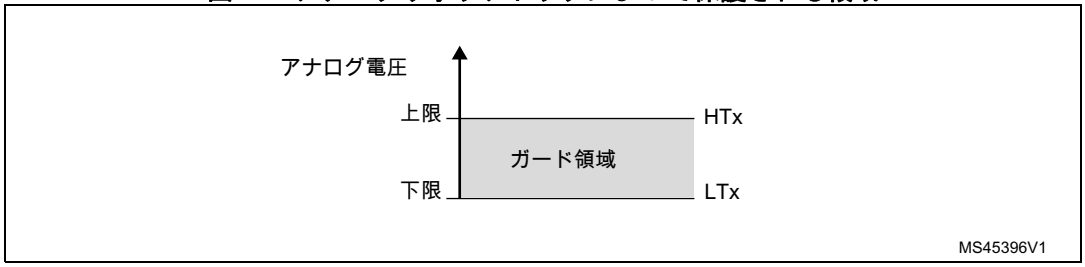


表330.アナログウォッチドッグ1チャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGLビット	AWD1ENビット
なし	x	0
すべてのチャネル	0	1
単一の(1) チャネル	1	1

1. AWD1CH[4:0] ビットによって選択されます。

アナログウォッチドッグ2および3の説明

2番目と3番目のアナログウォッチドッグはより柔軟性の高く、ADC_AWDxCR（x = 2、3）のAWDxCHyをプログラムすることにより、いくつかの選択されたチャネルをガードすることができます。

対応するウォッチドッグは、ADC_AWDxCRレジスタでAWDxCHy（x = 2、3）の任意のビットがセットされると有効になります。

12bit未満の分解能でデータを変換するときには（DRES[1:0] ビットに従って）、内部比較は常に12bit全体の元の変換データ（左詰め）に対して実行されるため、プログラムされた閾値のLSBはクリアされたままである必要があります。

表329 に、可能なすべての分解能での比較方法を示します。

AWD2/3アナログウォッチドッグステータスビットは、ADCにより変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。これらの閾値は、ADC_AWDxTRレジスタ（x = 2または3）の HTx[11:0] およびLTx[11:0] にプログラムされます。ADC_IERレジスタのAWDxIEビットをセットすることによって、割込みを有効にできます。

AWD2およびAWD3フラグは、ソフトウェアによって1を書き込むことによってクリアされます。

ADC_AWDx_OUT信号出力生成

各アナログウォッチドッグは、一部のオンチップタイマのETR入力（外部トリガ）に直接接続された内部ハードウェア信号 ADC_AWDx_OUT（x はウォッチドッグの数）？ と関連付けられています（ADC_AWDx_OUT信号をETRとして選択する方法の詳細はタイマのセクションを参照してください）。

ADC_AWDx_OUTは、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADC_AWDx_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADC_AWDx_OUT は、プログラムされた閾値内にある次のガードされた変換の終了後にリセットされます。それは、次のガードされた変換がまだプログラムされた閾値の外にある間は1であり続けます。
- ADC_AWDx_OUTは、ADCを無効化（ADDISを1にセット）するときにもリセットされます。なお、変換を中止すると（ADSTPを1にセット）ADC_AWDx_OUTの状態をクリアしてしまうことがあります。
- ADCがガードされていないチャネルを変換している間、ADC_AWDx_OUT の状態は変化しません（図295を参照）。

AWDxフラグはハードウェアによりセットされ、ソフトウェアによりリセットされます。AWDxフラグは、ADC_AWDx_OUTの生成にまったく影響しません（例として、ソフトウェアによってAWDxフラグがクリアされず、このフラグが1のまま維持される間、ADC_AWDx_OUTをトグルできます）。

ADC_AWDx_OUT信号はADC_CLKドメインで生成されます。この信号は、バスクロックが停止していても生成できます。

EOC比較はADCが終了するたびに実施されます。ADC_AWDx_OUTの立ち上がりおよび立ち下がりエッジは、比較完了後2 ADC_CLKサイクルが経過してから発生します。

ADC_AWDx_OUTはADC_CLKドメインにより、AWDフラグはバス･クロック･ドメインによりそれぞれ生成されるため、これらの信号の立ち上がりエッジは同期していません。

図293. ADC_AWDx_OUT信号生成

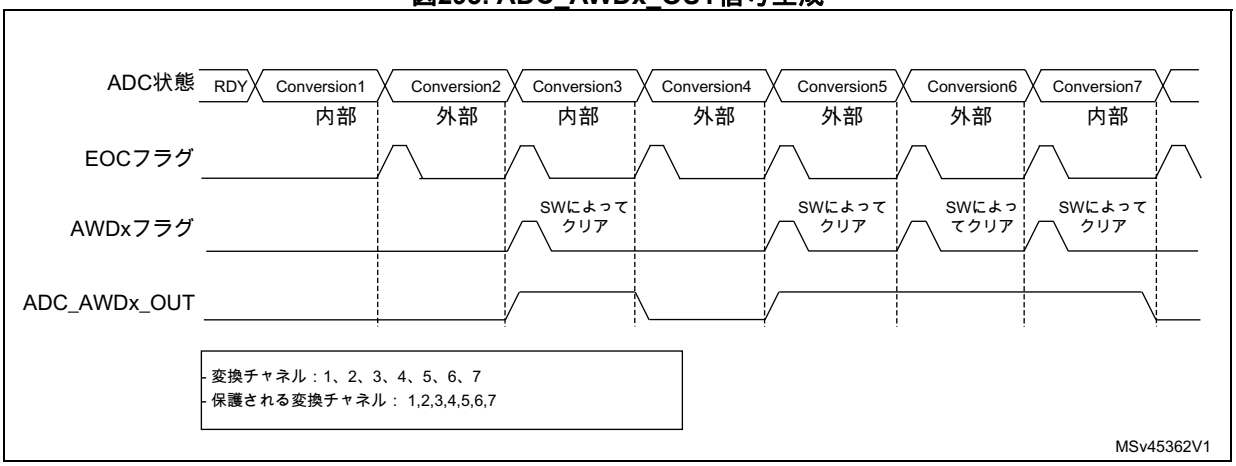


図294. ADC_AWDx_OUT信号生成（ソフトウェアによってAWDxフラグがクリアされない場合）

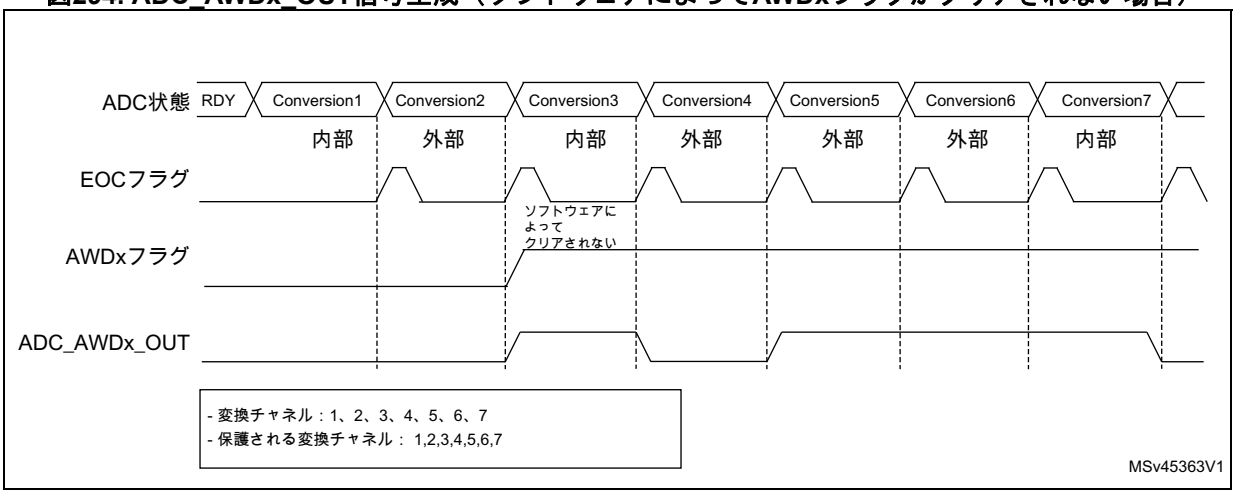
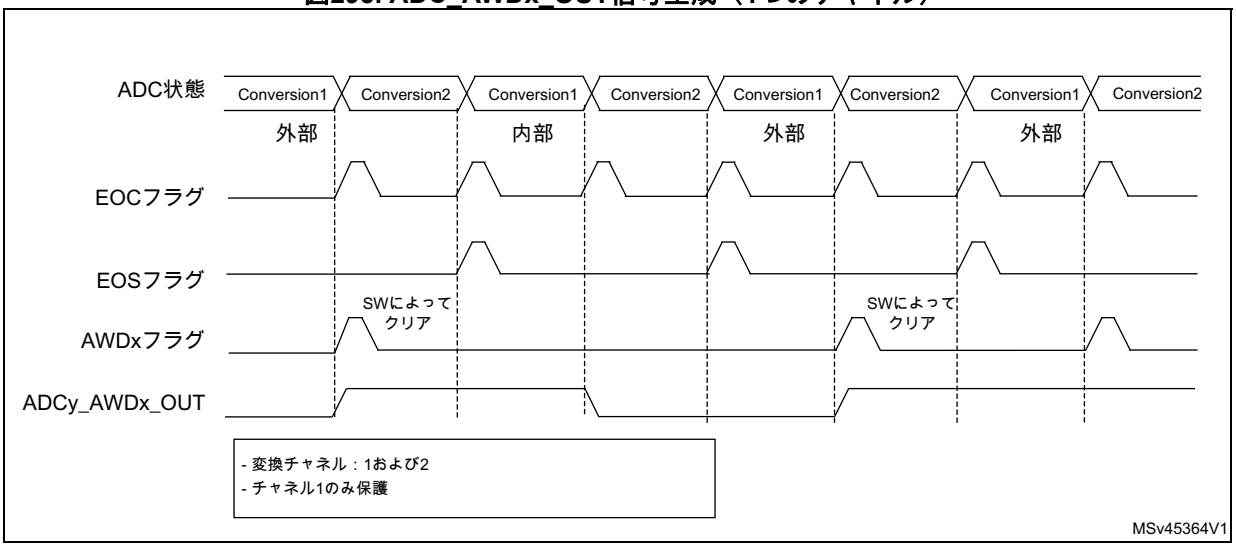


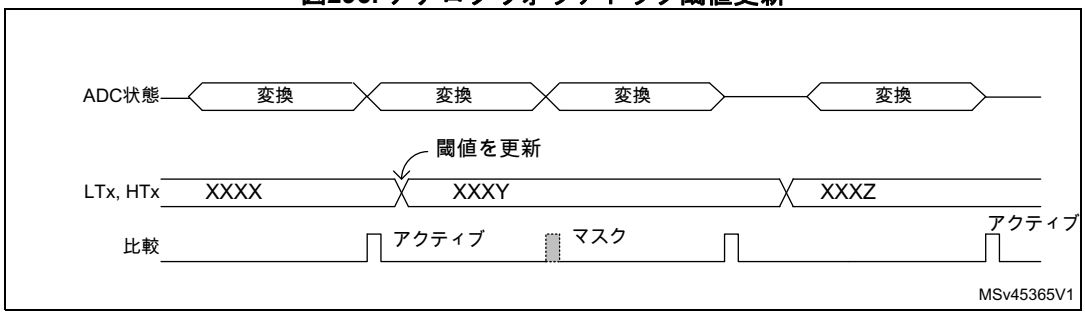
図295. ADC_AWDx_OUT信号生成（1つのチャネル）



アナログウォッチドッグ閾値制御

LTx[11:0] およびHTx[11:0] は、アナログ／デジタル変換中（ADC内部状態の変換開始から変換終了後）に変更できます。HTxおよびLTxのビットが、ADCのガードされたチャネルの変換中にプログラムされる場合、この変換のためにウォッチドッグ機能はマスクされます。このマスクは新規の変換が開始されるとクリアされ、その結果新しいAWD閾値が適用されて次のADC変換結果が開始します。AWD比較は変換が終了するたびに実施されます。現在のADCデータが新しい閾値間隔から外れていても、割込みまたはADC_AWDx_OUT信号は生成されません。割込みおよびADC_AWDx_OUTが生成されるのは、閾値更新の後に開始されたADC変換の終了時のみです。ADC_AWDx_OUTがすでにアサートされている場合、新しい閾値をプログラムしてもADC_AWDx_OUT信号はデアサートされません。

図296. アナログウォッチドッグ閾値更新



34.4.26 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPUの負荷を軽減します。複数の変換を処理して、最大16bitのデータ幅の単一データに平均化できます。

以下の形式で結果を提供します。NおよびMは調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

ハードウェアにより、平均化、データレートの削減、SNRの向上、基本的フィルタリングの機能を実行することができます。

オーバーサンプリング比Nは、ADC_CFGR2レジスタのOVSR[2:0] ビットによって定義されます。2xから256xまでの範囲にできます。分周係数Mは、最大8bitの右ビットシフトから成ります。これは、ADC_CFGR2レジスタのOVSS[3:0] ビットによって設定されます。

合計ユニットは最大20bit（256 x 12bit）の結果をもたらし、最初に右へシフトされます。結果の下位ビットは切り詰められ、最下位16bitのみが保持されて、シフトによって残った最下位ビットを使用して最も近い値に丸められてからADC_DRデータレジスタに転送されます。

注： **シフト後の中間結果が16bitを超える場合、結果の上位ビットは単純に切り詰められます。**

図297. 20bitから16bitへの結果の切詰め

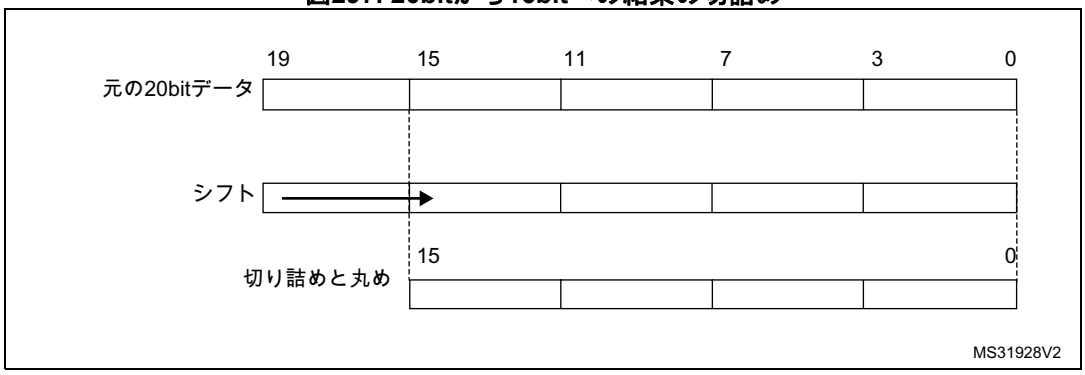
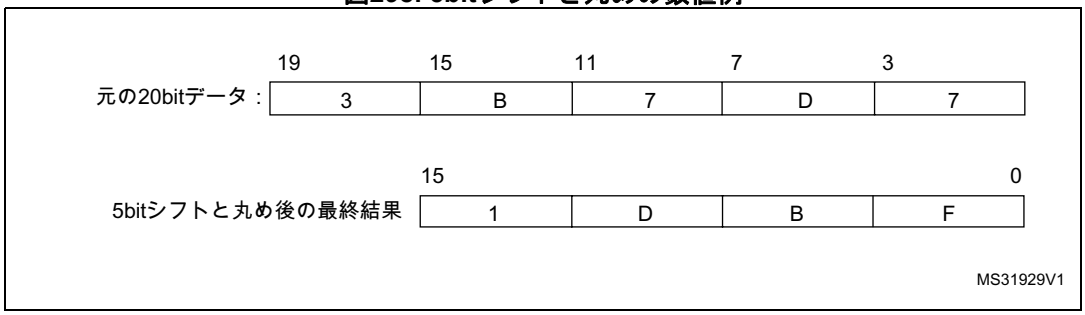


図298に、元の20bitの累積データから最終的な16bitの結果への処理の数値例を示します。

図298. 5bitシフトと丸めの数値例



下の 表331に、元の 変換データが0 x FFFの場合の、さまざまなNとMの組み合わせでのデータフォーマットを示します。

表331.最大出力結果対NとM。グレイの値は切り詰めを示す

オーバーサンプリング比	最大値 元のデータ	シフトなし OVSS = 0 000	1bit シフト OVSS = 0001	2bit シフト OVSS = 0010	3bit シフト OVSS = 0011	4bit シフト OVSS = 0100	5bit シフト OVSS = 0101	6bit シフト OVSS = 0110	7bit シフト OVSS = 0111	8bit シフト OVSS = 1000
2x	0x1FFE	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040	0x0020
4x	0x3FFC	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040
8x	0x7FF8	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080
16x	0xFFF0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100
32x	0x1FFE0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200
64x	0x3FFC0	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400
128x	0x7FF80	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800
256x	0xFFF00	0xFF00	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF

オーバーサンプリングモードでの変換タイミングは、標準変換モードと同じです。サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータはN回の変換ごとに提供され、同等遅延はN x t_{CONV} = N x（t_{SMPL} + t_{SAR}）に等しくなります。フラグ機能は、次のように実行されます。

- サンプリングフェーズの終了（EOSMP）は、各サンプリングフェーズ後にセットされます。
- 変換の終了（EOC）は、N回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOCSEQ）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N xシーケンス長の変換合計後）。

オーバーサンプリング時のADC動作モードのサポート

オーバーサンプリングモードでは、ほとんどのADC動作モードが使用可能です。

- シングルまたは連続モード変換、前方または後方スキャンシーケンス 、および最大8チャネルのプログラムされたシーケンス
- ソフトウェアまたはトリガによるADC変換の開始
- 変換中のADCの停止（中止）
- オーバーラン検出時のCPUまたはDMA経由でのデータの読出し
- 低電力モード（WAIT、AUTOFF）
- プログラム可能な分解能 : この場合、削減された変換値（ADC_CFGR1レジスタのRES[1:0] ビットに従って）の累積、切り詰め、丸め、およびシフトは、12bit変換と同様に行われます。

注： **オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADC_CFGR1のALIGNビットは無視され、データは常に右詰めで提供されます。**

アナログウォッチドッグ

アナログウォッチドッグ機能は利用できますが、次のような違いがあります。

- RES[1:0] ビットは無視され、比較は常に12bit値全部HT[11:0] およびLT[11:0] を使用して行われます。
- 比較は、16bitのオーバーサンプリングされた結果の上位12bit ADC_DR[15:4] に対して行われます。

注： **高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなります。たとえば、オーバーサンプリングされた結果が4bitシフトされた場合、12bitの右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8bitに対してだけです。比較はADC_DR[11:4] とHTx[7:0] / LTx[[7:0] の間で行われ、HTx[11:8] / LTx[11:8] はリセットしておく必要があります。.**

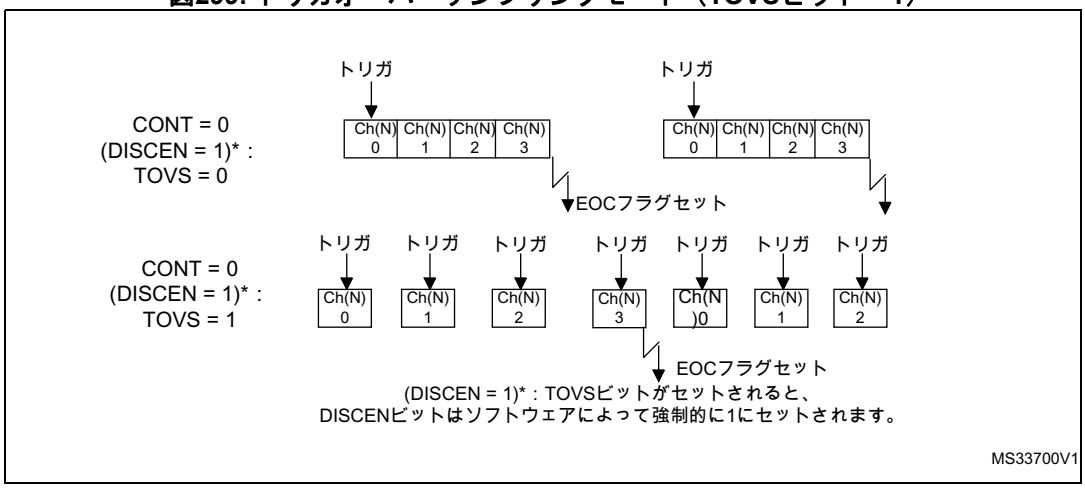
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり効率的なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、ADC_CFGR2のTOVSビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

図299 に、不連続モードでトリガに反応して変換を開始する方法を示します。

TOVSビットがセットされている場合、DISCENビットの内容は無視され、1とみなされます。

図299. トリガオーバーサンプリングモード（TOVSビット = 1）



34.4.27 温度センサと内部基準電圧

温度センサを使用して、デバイスの接合温度（T_J）を測定できます。温度センサは、センサの出力電圧をデジタル値に変換するADC内部入力チャネルに内部接続されています。温度センサのアナログピンのサンプリング時間は、最小のT_{S_temp} より大きい値である必要があります。使用しないときには、センサをパワーダウンモードにすることができます。

内部電圧基準（VREFINT）は、ADCとコンパレータに安定した（バンドギャップ）電圧出力を提供します。

上記の電圧が接続されるADC内部入力チャネルの詳細については、セクション 34.4.2 : A/Dコンバータのピンおよび内部信号のADC相互接続の表を参照してください。

図300に、温度センサ、内部電圧基準、およびADC間の接続のブロック図を示します。

VSENSE の変換を有効にするには、VSENSESELビットをセットする必要があり、V_{REFINT} の変換を有効にするには、VREFENビットをセットする必要があります。

ADCが自律モードで動作している場合、これらの信号は、電力消費を抑えるために自動的に制御されます（自律モードで電圧を測定するには、VSENSESELと VREFENをセットする必要があります）。

温度センサの出力電圧は、温度に比例して変化します。このラインのオフセットは、プロセスのばらつきにより、チップごとに異なります（チップ間で最大45°C）。

較正されていない内部温度センサは、温度の絶対値の代わりに温度変化を検出するアプリケーションに適しています。温度センサの測定精度を高めるために、生産時にSTマイクロエレクトロニクスによって各デバイスの較正値がシステムメモリに格納されています。

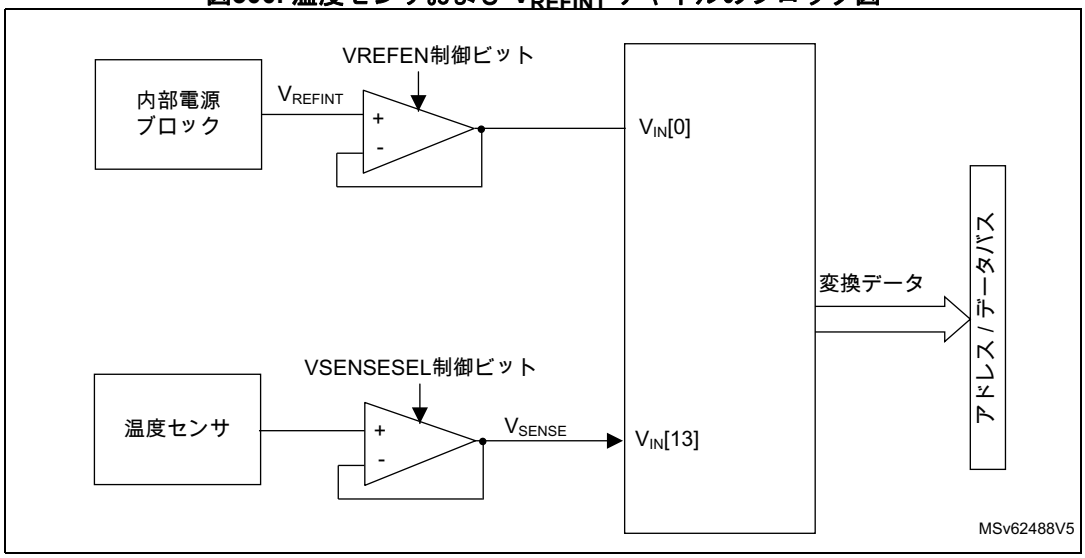
製造プロセス中に、温度センサと内部電圧基準の較正データがシステムメモリ領域に格納されます。ユーザアプリケーションはこれらを読み出して、温度センサまたは内部基準の精度の向上に使用できます。詳細については、データシートを参照してください。

製造プロセス中に、内部電圧基準の較正データがシステムメモリ領域に格納されます。ユーザアプリケーションはこれらを読み出して、内部基準の精度の向上のために使用できます。詳細については、デバイスのデータシートの電気的特性のセクションを参照してください。

主な特長

- 直線性：最大±2°C、精度は較正に依存

図300. 温度センサおよび V_{REFINT} チャネルのブロック図



温度の読出し

1. V_{SENSE} に接続された入力チャネルを選択します（セクション 34.4.2 : A/Dコンバータのピンおよび内部信号のADC相互接続の表を参照）。
2. デバイスのデータシートで指定されている適切なサンプリング時間（T_{S_temp}）を選択します。
3. ADC_CCRレジスタのVSENSESELビットをセットして、温度センサをパワーダウンモードからウェイクアップし、安定時間（t_{START}）だけ待ちます。
4. ADC_CR レジスタのADSTART ビットをセットすることによって（または外部トリガによって）ADC変換を開始します。
5. ADC_DRレジスタから結果のV_{SENSE} データを読み出します。
6. 次の式を使用して、温度を計算します。

$$\text{Temperature (in °C)} = \frac{\text{TS_CAL2_TEMP} - \text{TS_CAL1_TEMP}}{\text{TS_CAL2} - \text{TS_CAL1}} \times (\text{TS_DATA} - \text{TS_CAL1}) + \text{TS_CAL1_TEMP}$$

ここで、

- TS_CAL2は、TS_CAL2_TEMPで得られた温度センサの較正値です。
- TS_CAL1は、TS_CAL1_TEMPで得られた温度センサの較正値です。
- TS_DATAは、ADCによって変換された実際の温度センサの出力値です。

TS_CAL1およびTS_CAL2較正ポイントについての詳細は、セクション 34.3 : A/Dコンバータの実装を参照してください。

注： **センサがパワーダウンモードからウェイクアップして、正しいレベルでV_{SENSE} を出力できるようになるまでには時間がかかります（スタートアップ時間）。ADCにも起動後のスタートアップ時間があるので、遅延を最小にするには、ADENビットとVSENSESELビットを同時にセットしてください。**

内部基準電圧を使用した実際の V_{REF+} 電圧の計算

マイクロコントローラに印加されるV_{DDA} 電源電圧は、ばらつきがあり、または正確にはわからないことがあります。VREF+ ピンが可変V_{DDA} 電源に接続されている場合、内蔵の内部電圧基準（VREFINT）と、製造プロセス時に V_{REF+} = 3.0VでADCによって得られた較正データを使用して、実際の V_{REF+} 電圧レベルを評価することができます。

デバイスに印加される実際の V_{REF+} 電圧は、次の式で求められます。

$$V_{REF+} = 3.0\ V \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- VREFINT_CALは、VREFINTの較正値です。
- VREFINT_DATAは、ADCによって変換された実際のVREFINTの出力値です。

電源に相対的なADC測定値から絶対電圧値への変換

ADCは、電圧基準 V_{REF+} と、変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。ほとんどのアプリケーションの使用事例では、この比を V_{REF+} に依存しない電圧に変換する必要があります。V_{REF+} が既知であり、ADCによって変換された値が右詰めされるアプリケーションでは、次の式を使用して、この絶対値を求めることができます。

$$V_{CHANNELx} = \frac{V_{REF+}}{FULL_SCALE} \times ADC_DATA_x$$

V_{REF+} 値が既知ではないアプリケーションの場合、内部電圧基準を使用する必要があり、V_{REF+} をセクション ：内部基準電圧を使用した実際の V_{REF+} 電圧の計算に記載されている式で置き換えることができ、次の式が得られます。

$$V_{CHANNELx} = \frac{3.0\ V \times VREFINT_CAL \times ADC_DATA_x}{VREFINT_DATA \times FULL_SCALE}$$

ここで、

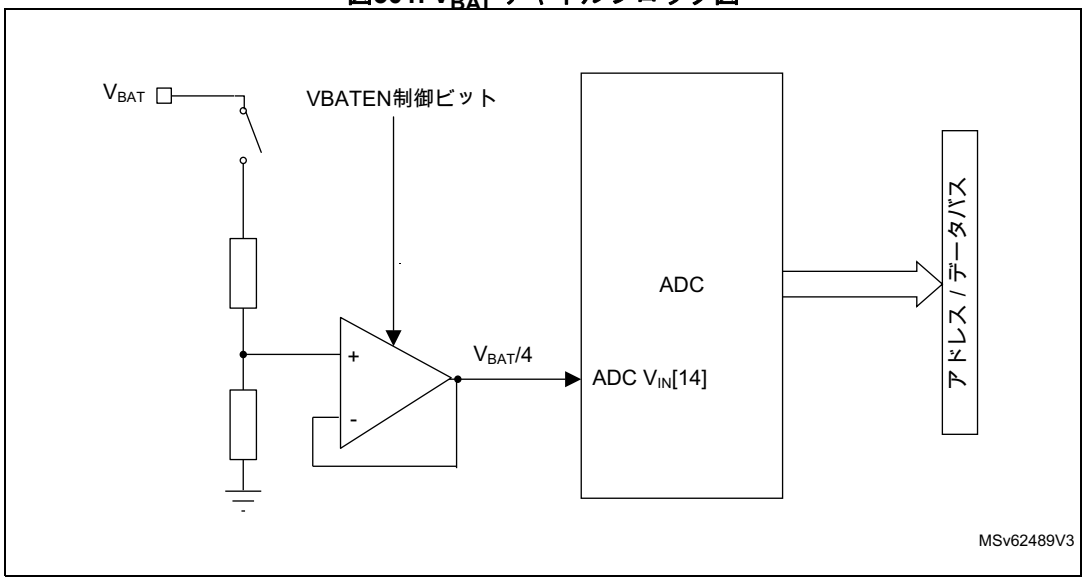
- VREFINT_CALはVREFINT較正値です（VREFINT_CALの値については、セクション 34.3 : A/Dコンバータの実装を参照）。
- ADC_DATA_x は、チャネルxでADCによって測定された値です（右詰め）。
- VREFINT_DATAは、ADCによって変換された実際のVREFINTの出力値です。
- FULL_SCALEは、ADC出力の最大デジタル値です。たとえば、12bit分解能では、2^{12} - 1 = 4095であり、8bit分解能では2^8 - 1 = 255です。

注： **ADC測定が12bit右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。**

34.4.28 バッテリ電圧監視

アプリケーションではADC_CCRレジスタのVBATENビットによって、VBATピンのバックアップバッテリ電圧を測定できます。V_{BAT} 電圧はV_{DDA} より高くなることがあるので、ADCの正しい動作を確保するために、V_{BAT} ピンはブリッジ 4 分圧回路に内部接続されています。このブリッジは、VBATENビットがセットされると自動的に有効になり、V_{BAT}/ 4を対応するADC入力チャネルに接続します（セクション 34.4.2 : A/Dコンバータのピンおよび内部信号のADC相互接続の表を参照）。結果として、変換されたデジタル値はV_{BAT} 電圧の半分です。望ましくないバッテリ消費を避けるには、ADC変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

図301. V_{BAT} チャネルブロック図



34.4.29 別のADCによる同時動作

ADC4 が別のADCと同時に使用される場合（ADCxとし、xは4以外）、ADCxの動作により、V_{REF+} 電圧にノイズが発生することがあります。V_{REF+} は ADC4 の基準電圧でもあるため、このために変換エラーが発生することがあります。この問題が発生しないようにするには、ADC_PWRRレジスタのVREFPROTENビットをセットします。ADCxのサンプリングフェーズが開始するとすぐに、ADC4 の1クロックサイクルの間、ADC4 は保留状態になるため、ADC4 の変換時間は1クロックサイクル分長くなります。

また、ADC4 の変換中、ADCxは2サンプリングフェーズになることがあります。これは、ADCxのインジェクト変換機能のためです。ADC_PWRRのVREFSECSMPに1をセットすることにより、変換フェーズの間、ADC4 の動作を2回にすることができます。VREFSECSMPビットがセットされている場合、ADC4 の変換時間は2クロックサイクル分長くなります。

VREFSECSMPビットとVREFPROTENビットを同時にセットする必要があります。

ADCxと ADC4 で同時動作機能を利用するには、両方で同じクロックソースを使用する必要があります。

34.5 ADC低電力モード

表332.低電力モードがA/Dコンバータに与える影響

モード	説明
SLEEP	影響なし。DMAリクエストは機能します。ADC割込みによって、デバイスはSLEEPモードを終了します。
STOP	ADCレジスタの内容は保持されます。 ADCは機能します。DMAリクエストは機能し、割込みにより、デバイスはSTOPモードを終了します。
STANDBY	ADCペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

34.6 ADC割込み

割込みは、次のイベントによって生成できます。

- 較正の終了（EOCALフラグ）
- ADCパワーアップ、ADCの準備ができたとき（ADRDYフラグ）
- 変換の終了（EOCフラグ）
- 変換シーケンスの終了（EOSフラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、AWD3フラグ）
- サンプリングフェーズの終了が発生したとき（EOSMPフラグ）
- データオーバーランの発生時（OVRフラグ）
- LDOレディ、LDO出力の安定時（LDORDYフラグ）

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表333.ADCのウェイクアップおよび割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了(1)	STANDBYモードの終了
ADC	LDOレディ	LDORDY	LDORDYIE	LDORDYを1にプログラム	あり	あり	なし
	較正の終了	EOCAL	EOCALIE	EOCALを1にプログラム		あり	
	ADCレディ	ADRDY	ADRDYIE	ADRDYを1にプログラム		あり	
	変換の終了	EOC	EOCIE	EOCを1にプログラムし、ADC_DRを読み出す		あり	
	変換シーケンスの終了	EOS	EOSIE	EOSを1にプログラム		あり	
	アナログウォッチドッグ1のステータスビットのセット	AWD1	AWD1IE	AWD1を1にプログラム		あり	
	アナログウォッチドッグ2のステータスビットのセット	AWD2	AWD2IE	AWD2を1にプログラム		あり	
	アナログウォッチドッグ3のステータスビットのセット	AWD3	AWD3IE	AWD2を1にプログラム		あり	
	サンプリングフェーズの終了	EOSMP	EOSMPIE	EOSMPを1にプログラム		なし	
	オーバーラン	OVR	OVRIE	OVRを1にプログラム		あり	

1. ADCは、ペリフェラルインスタンスがSTOPモードからのウェイクアップ機能をサポートしている場合にのみ、デバイスをSTOPモードからウェイクアップすることができます。サポートされているSTOPモードのリストについては、セクション 34.3 : A/Dコンバータの実装を参照してください。

34.7 ADCレジスタ

レジスタの説明で使用されている略語のリストについては、セクション1.2を参照してください。

34.7.1 ADC割込みおよびステータスレジスタ（ADC_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LDO RDY	EOCAL	Res.	AWD3	AWD2	AWD1	Res.	Res.	OVR	EOS	EOC	EOSMP	ADRDY
			rc_w1	rc_w1		rc_w1	rc_w1	rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LDORDY**：LDOのレディ状態

このビットは、ハードウェアによってセットされます。ADC内部LDO出力がレディ状態であることを示します。1を書き込むことにより、ソフトウェアによってクリアされます。

0：ADC電圧レギュレータは無効です。

1：ADC電圧レギュレータは有効で、安定しています。

ビット11 EOCAL：較正終了フラグ

このビットは、較正が完了したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：較正は完了していません。

1：較正は完了しています。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **AWD3**：アナログウォッチドッグ3フラグ

このビットは、変換された電圧がADC_AWD3TRおよびADC_AWD3TRレジスタでプログラミングされた値を逸脱したときに、ハードウェアによってセットされます。1を書き込むことにより、ソフトウェアによってクリアされます。

0：アナログウォッチドッグイベントは発生しませんでした（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグイベントが発生しました。

ビット8 **AWD2**：アナログウォッチドッグ2フラグ

このビットは、変換された電圧がADC_AWD2TRおよびADC_AWD2TRレジスタでプログラミングされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：アナログウォッチドッグイベントは発生しませんでした（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグイベントが発生しました。

ビット7 **AWD1**：アナログウォッチドッグ1フラグ

このビットは、変換された電圧がADC_TR1およびADC_HR1レジスタでプログラミングされた値を逸脱したときに、ハードウェアによってセットされます。1を書き込むことにより、ソフトウェアによってクリアされます。

0：アナログウォッチドッグイベントは発生しませんでした（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグイベントが発生しました。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OVR**：ADCオーバーラン

このビットは、オーバーランが発生したときにハードウェアによってセットされ、EOCフラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで1を書き込むことによってクリアされます。

0：オーバーランは発生しませんでした（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット3 **EOS**：シーケンス終了フラグ

このビットは、CHSELビットによって選択されたチャネルのシーケンスの変換終了時にハードウェアによってセットされます。ソフトウェアで1を書き込むことによってクリアされます。

0：変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：変換シーケンスが完了しました。

ビット2 **EOC**：変換終了フラグ

このビットは、チャネルの各変換の終了時に、新しいデータ結果がADC_DRレジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって1を書き込むことによって、またはADC_DRレジスタを読み出すことによってクリアされます。

0：チャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：チャネル変換が完了しました。

ビット1 **EOSMP**：サンプリング終了フラグ

このビットは、変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。1を書き込むことにより、ソフトウェアによってクリアされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズの終了に達しました。

ビット0 **ADRDY**：ADCのレディ状態

このビットは、ADCが有効にされた後（ADEN = 1）、ADCが変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで1を書き込むことによってクリアされます。

0：ADCはまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：ADCは変換を開始する準備ができました。

34.7.2 ADC割込み有効レジスタ（ADC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LDORDYIE	EOCALIE	Res.	AWD3IE	AWD2IE	AWD1IE	Res.	Res.	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
			rw	rw		rw	rw	rw			rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LDORDYIE**：LDOレディ割込みの有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、LDORDY割込みを有効化／無効化するために使用されます。

0：LDOレディ値割込みは無効です。

1：LDOレディ値割込みは有効です。LDO出力がレディ状態になると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット11 **EOCALIE**：較正終了割込みの有効化

このビットは、較正終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：較正終了割込みは無効です。

1：較正終了割込みは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **AWD3IE**：アナログウォッチドッグ3割込みの有効化

このビットは、アナログウォッチドッグ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ割込みは無効です。

1：アナログウォッチドッグ割込みは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット8 AWD2IE：アナログウォッチドッグ2割込みの有効化

このビットは、アナログウォッチドッグ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ割込みは無効です。

1：アナログウォッチドッグ割込みは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット7 **AWD1IE**：アナログウォッチドッグ1割込みの有効化

このビットは、アナログウォッチドッグ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ割込みは無効です。

1：アナログウォッチドッグ割込みは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OVRIE**：オーバーラン割込みの有効化

このビットは、オーバーラン割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割込みは無効です。

1：オーバーラン割込みは有効です。OVRビットがセットされると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット3 **EOSIE**：変換シーケンス終了割込みの有効化

このビットは、変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS割込みは無効です。

1：EOS割込みは有効です。EOSビットがセットされると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット2 **EOCIE**：変換終了割込みの有効化

このビットは、変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC割込みは無効です。

1：EOC割込みは有効です。EOCビットがセットされると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット1 **EOSMPIE**：サンプリング終了フラグ割込みの有効化

このビットは、サンプリングフェーズ終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP割込みは無効です。

1：EOSMP割込みは有効です。EOSMPビットがセットされると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット0 **ADRDYIE**：ADCレディ割込みの有効化

このビットは、ADCレディ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY割込みは無効です。

1：ADRDY割込みは有効です。ADRDYビットがセットされると、割込みが生成されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

34.7.3 ADC制御レジスタ（ADC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	Res.	Res.	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rs			rw												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADSTP	Res.	ADSTART	ADDIS	ADEN
											rs		rs	rs	rs

ビット31 **ADCAL**：ADCの較正

このビットは、ADCの較正を開始するためにソフトウェアによってセットされます。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：1を書き込むと、ADCが較正されます。1が読み出された場合、較正の実行中であることを意味します。

注： ソフトウェアは、ADCが無効なときだけ（ADCAL = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、AUTOFF = 0、およびADEN = 0）、このビットを書き込むことができます。

ソフトウェアは、ADENに1をセットし、ADSTPに1を書き込んでADSTATが0にクリアされたときのみ（ADCが有効であり、変換中でない）、ADC_CALFACTを書き込むことによって較正係数を更新できます。

ビット30:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ADVREGEN**：ADC電圧レギュレータの有効化

このビットは、ADC内部電圧レギュレータを有効にするために、ソフトウェアによってセットされます。$t_{ADCVREG_SETUP}$後、電圧レギュレータ出力は使用可能になります。

電圧レギュレータを無効にすることによって、ソフトウェアによってクリアされます。ADENを0にセットすることによってのみクリアされます。

0：ADC電圧レギュレータは無効です。

1：ADC電圧レギュレータは有効です。

注： ソフトウェアは、ADCが無効のときのみ（ADCAL = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、このビットフィールドをプログラムできます。

ビット27:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ADSTP**：ADC変換停止コマンド

このビットは、実行中の変換を停止および破棄するためにソフトウェアによってセットされます（ADSTPコマンド）。

変換が効果的に破棄され、ADCが新しい変換開始コマンドを受け入れる準備ができたときに、ハードウェアによってクリアされます。

0：ADC変換停止コマンドは実行中ではありません。

1：1を書き込むと、ADCが停止します。1が読み出された場合、ADSTPコマンドが実行中であることを意味します。

注： ADCの状態をクリアするには、ソフトウェアによりA/D変換がトリガされた後にADSTARTが0にクリアされていても、ADSTPに1をセットする必要があります。設定を変更する必要がある場合は、ADSTPに1をセットすることを推奨します。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ADSTART**：ADC変換開始コマンド

このビットは、ADC変換を開始するためにソフトウェアによってセットされます。EXTEN［1:0］設定ビットに応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはハードウェアトリガイベントが発生したときに（ハードウェアトリガ設定）開始されます。

次のときに、ハードウェアによってクリアされます。

– シングル変換モード（CONT = 0、DISCEN = 0）では、ソフトウェアトリガが選択されたとき（EXTEN = 00）：変換シーケンス終了（EOS）フラグのアサート時。

– 不連続変換モード（CONT = 0、DISCEN = 1）では、ソフトウェアトリガが選択されたとき（EXTEN = 00）：変換終了（EOS）フラグのアサート時。

– すべての場合：ADSTPコマンドの実行後、ADSTPビットがハードウェアによってクリアされると同時に。

0：ADC変換は実行中ではありません。

1：1を書き込むと、ADC開始されます。1が読み出された場合、ADCが動作中であり、変換中である可能性があることを意味します。

注： ソフトウェアは、ADEN = 1かつADDIS = 0である（ADCが有効であり、ADCを無効にする保留中のリクエストがない）ときだけ、ADSTARTをセットできます。

ビット1 **ADDIS**：ADC無効化コマンド

このビットは、ADCを無効にして（ADDISコマンド）、パワーダウン状態（OFF状態）にするためにソフトウェアによってセットされます。

ADCが効果的に無効化されると、ハードウェアによってクリアされます（ADENもこの時点でハードウェアによってクリアされます）。

0：ADDISコマンドは実行中ではありません。

1：1を書き込むと、ADCが無効化されます。1が読み出された場合、ADDISコマンドが実行中であることを意味します。

注： ADEN = 1かつADSTART = 0（変換が実行中でない）のときのみ、ADSTPに1をセットすることは有効です。

ビット0 **ADEN**：ADC有効化コマンド

このビットは、ADCを有効化するために、ソフトウェアによってセットされます。ADRDYフラグ がセットされたときは、ADCは動作する準備ができています。

ADDISコマンドの実行後、ADCが無効になるとハードウェアによってクリアされます。

0：ADCは無効です（OFF状態）。

1：1を書き込むと、ADCが有効になります。

注： ソフトウェアは、ADC_CRレジスタのすべてのビットが0のとき（ADCAL = 0、ADSTP = 0、ADSTART = 0、ADDIS = 0、およびADEN = 0）だけ、ADENをセットできます。

34.7.4 ADC設定レジスタ1（ADC_CFGR1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWD1CH[4:0]					Res.	Res.	AWD1 EN	AWD1 SGL	CHSELR MOD	Res.	Res.	Res.	Res.	DISCEN
	rw	rw	rw	rw	rw			rw	rw	rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WAIT	CONT	OVRMOD	EXTEN[1:0]		Res.	EXTSEL[2:0]			ALIGN	SCANDI R	RES[1:0]		DMACF G	DMAEN
	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:26 **AWD1CH[4:0]**：アナログウォッチドッグチャネルの選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：ADCアナログ入力チャネル0がAWDによって監視されます。

00001：ADCアナログ入力チャネル1がAWDによって監視されます。

.....

10001：ADCアナログ入力チャネル17がAWDによって監視されます。

10110：ADCアナログ入力チャネル22がAWDによって監視されます。

10111：ADCアナログ入力チャネル23がAWDによって監視されます。

その他：予約済み

注： AWDCH［4;0］ビットによって選択されたチャネルは、CHSELRレジスタにもセットされる必要があります。
ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット25:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **AWD1EN**：アナログウォッチドッグの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 1 は無効です。

1：アナログウォッチドッグ 1 は有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット22 **AWD1SGL**：単一チャネルまたはすべてのチャネルでのウォッチドッグの有効化

このビットは、AWDCH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ 1 は有効です。

1：単一チャネルでアナログウォッチドッグ 1 は有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット21 **CHSELRMOD**：ADC_CHSELRレジスタのモード選択

このビットは、ADC_CHSELR機能を制御するために、ソフトウェアによってセット／クリアされます。

0：ADC_CHSELRレジスタの各ビットにより入力が有効化されます。

1：ADC_CHSELRレジスタは最大で8チャネルのシーケンスが可能です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット20:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **DISCEN**：不連続モード

このビットは、不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：不連続モードは無効です。

1：不連続モードは有効です。

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方をセットすることは禁じられています。

ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **WAIT**：ウェイト変換モード

このビットは、ウェイト変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ウェイト変換モードはオフです。

1：ウェイト変換モードはオンです。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット13 **CONT**：シングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的に変換が行われます。

0：シングル変換モード

1：連続変換モード

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1とCONT = 1の両方をセットすることは禁じられています。

ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット12 **OVRMOD**：オーバーラン管理モード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADC_DRレジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADC_DRレジスタは最新の変換結果で上書きされます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット11:10 **EXTEN[1:0]**：外部トリガの有効化と極性の選択

これらのビットは、外部トリガ極性を選択し、トリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって開始できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8:6 **EXTSEL[2:0]**：外部トリガの選択

これらのビットは、変換の開始をトリガするために使用される外部イベントを選択します（詳細については、セクション 34.4.2：A/Dコンバータのピンおよび内部信号の ADC相互接続の表を参照）。

000：adc_trg0
001：adc_trg1
010：adc_trg2
011：adc_trg3
100：adc_trg4
101：adc_trg5
110：adc_trg6
111：adc_trg7

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット5 **ALIGN**：データの配置

このビットは、右詰めまたは左詰めを選択するために、ソフトウェアによってセット／クリアされます。図 285： 1377ページのデータの配置と分解能（オーバーサンプリング無効：OVSE = 0）を参照してください。

0 :右詰め
1：左詰め

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット4 **SCANDIR**：スキャンシーケンス方向

このビットは、シーケンス内のチャネルをスキャンする方向を選択するために、ソフトウェアによってセット／クリアされます。CHSELRMODビットが0にクリアされたときにのみ有効です。

0：前方スキャン（CHSEL0からCHSEL23）
1：後方スキャン（CHSEL23 からCHSEL0）

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット3:2 **RES[1:0]**：データの分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

00：12bit
01：10bit
10：8bit
11：6bit

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこれらのビットに書き込むことができます。

ビット1 **DMACFG**：ダイレクトメモリアクセス設定

このビットは、2つのDMA動作モードを選択するためにソフトウェアによってセット／クリアされ、DMAEN = 1のときのみ有効です。

0：DMAワンショットモードが選択されています。

1：DMAサーキュラモードが選択されています。

詳細については、セクション ： 1378ページのDMAを使用した変換データの管理を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット0 **DMAEN**：ダイレクトメモリアクセスの有効化

このビットは、DMAリクエストの生成を有効にするために、ソフトウェアによってセット／クリアされます。これにより、DMAコントローラによる変換データの自動管理が可能になります。詳細については、セクション ： 1378ページのDMAを使用した変換データの管理を参照してください。

0：DMAは無効です。

1：DMAは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

34.7.5 ADC設定レジスタ2（ADC_CFGR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	LFTRIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TOVS	OVSS[3:0]				OVSR[2:0]			Res.	OVSE
						rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **LFTRIG**：低周波数トリガモードの有効化

このビットは、ソフトウェアでセットする必要があります。

0：予約済み

1：低周波数トリガモードは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされているとき（変換が実行中でない）のみ、ソフトウェアでこのビットに書き込むことができます。

ビット 28:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **TOVS**：トリガ*オーバーサンプリング*

このビットは、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとにトリガが必要です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット8:5 **OVSS[3:0]**：オーバーサンプリングのシフト

このビットは、ソフトウェアによってセット／クリアされます。

0000：シフトなし

0001：1bitシフト

0010：2bitシフト

0011：3bitシフト

0100：4bitシフト

0101：5bitシフト

0110：6bitシフト

0111：7bitシフト

1000：8bitシフト

その他：予約済み

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット4:2 **OVSR[2:0]**：オーバーサンプリング比

このビットは、オーバーサンプリング比の数を定義します。

000：2x

001：4x

010：8x

011：16x

100：32x

101：64x

110：128x

111：256x

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **OVSE**：オーバーサンプリング回路の有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：オーバーサンプリング回路は無効です。

1：オーバーサンプリング回路は有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.6 ADCサンプリング時間レジスタ（ADC_SMPR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SMPSEL23	SMPSEL22	SMPSEL21	SMPSEL20	SMPSEL19	SMPSEL18	SMPSEL17	SMPSEL16	SMPSEL15	SMPSEL14	SMPSEL13	SMPSEL12	SMPSEL11	SMPSEL10	SMPSEL9	SMPSEL8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMPSEL7	SMPSEL6	SMPSEL5	SMPSEL4	SMPSEL3	SMPSEL2	SMPSEL1	SMPSEL0	Res.	SMP2 [2:0]			Res.	SMP1 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw		rw	rw	rw

ビット 31:8 **SMPSELx**：チャネルxサンプリング時間の選択（x = 23 ~ 0）

これらのビットは、使用するサンプリング時間を選択するため、ソフトウェアによって書き込まれます。

0：CHANNELxのサンプリング時間は、SMP1[2:0] レジスタの設定を使います。

1：CHANNELxのサンプリング時間は、SMP2[2:0] レジスタの設定を使います。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **SMP2[2:0]**：サンプリング時間の選択2

これらのビットは、使用するサンプリング時間を定義するため、ソフトウェアによって書き込まれます。

000：1.5 ADCクロックサイクル

001：3.5 ADCクロックサイクル

010：7.5 ADCクロックサイクル

011：12.5 ADCクロックサイクル

100：19.5 ADCクロックサイクル

101：39.5 ADCクロックサイクル

110：79.5 ADCクロックサイクル

111：814.5 ADCクロックサイクル

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 SMP1[2:0]：サンプリング時間の選択1

これらのビットは、すべてのチャネルに適用されるサンプリング時間を選択するために、ソフトウェアによって書き込まれます。

000：1.5 ADCクロックサイクル

001：3.5 ADCクロックサイクル

010：7.5 ADCクロックサイクル

011：12.5 ADCクロックサイクル

100：19.5 ADCクロックサイクル

101：39.5 ADCクロックサイクル

110：79.5 ADCクロックサイクル

111：814.5 ADCクロックサイクル

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.7 ADCウォッチドッグ閾値レジスタ（ADC_AWD1TR）

アドレスオフセット：0x20

リセット値：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT1 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LT1 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **HT1[11:0]**：アナログウォッチドッグ1高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LT1[11:0]**：アナログウォッチドッグ 1 低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

34.7.8 ADCウォッチドッグ閾値レジスタ（ADC_AWD2TR）

アドレスオフセット：0x24

リセット値：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT2 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LT2 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **HT2[11:0]**：アナログウォッチドッグ2高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LT2[11:0]**：アナログウォッチドッグ2低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

34.7.9 ADCチャネル選択レジスタ［オルタネート］（ADC_CHSELR）

アドレスオフセット：0x28

リセット値：0x0000 0000

同じレジスタは2つのモードで使用することができます。

– 各ADC_CHSELRビットにより入力が有効化されます（ADC_CFGR1でCHSELRMOD = 0）。このセクションを参照してください。

– ADC_CHSELRは最大8チャネルまで連続させることができます（ADC_CFGR1でCHSELRMOD = 1）。次のセクションを参照してください。

ADC_CFGR1でCHSELRMOD = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHSEL23	CHSEL22	CHSEL21	CHSEL20	CHSEL19	CHSEL18	CHSEL17	CHSEL16
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHSEL15	CHSEL14	CHSEL13	CHSEL12	CHSEL11	CHSEL10	CHSEL9	CHSEL8	CHSEL7	CHSEL6	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 :0 **CHSELx**：チャネルxの選択（x = 23 ~ 0）

これらのビットはソフトウェアによって書き込まれ、変換されるチャネルのシーケンスの一部になるチャネルを定義します。

0：入力チャネルxは変換対象として選択されません。

1：入力チャネルxは変換対象として選択されます。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.10 ADCチャネル選択レジスタ［オルタネート］（ADC_CHSELR）

アドレスオフセット：0x28

リセット値：0x0000 0000

同じレジスタは2つのモードで使用することができます。

– 各ADC_CHSELRビットにより入力が有効化されます（ADC_CFGR1でCHSELRMOD = 0）。この前のセクションを参照してください。

– ADC_CHSELRは最大8チャネルまで連続させることができます（ADC_CFGR1でCHSELRMOD = 1）。このセクションを参照してください。

ADC_CFGR1でCHSELRMOD = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SQ8 [3:0]				SQ7 [3:0]				SQ6 [3:0]				SQ5 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ4 [3:0]				SQ3 [3:0]				SQ2 [3:0]				SQ1 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 **SQ8[3:0]**：シーケンスの8番目の変換

これらのビットは、シーケンスの8番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

0000：CH0

0001：CH1

...

1010：CH10

1011：CH11

1100：CH12

1101：CH13

1110：CH14

1111：チャネル未選択（シーケンスの終了）

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット27:24 **SQ7[3:0]**：シーケンスの7番目の変換

これらのビットは、シーケンスの7番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット23:20 **SQ6[3:0]**：シーケンスの6番目の変換

これらのビットは、シーケンスの6番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット19:16 **SQ5[3:0]**：シーケンスの5番目の変換

これらのビットは、シーケンスの5番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット15:12 **SQ4[3:0]**：シーケンスの4番目の変換

これらのビットは、シーケンスの4番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット11:8 **SQ3[3:0]**：シーケンスの3番目の変換

これらのビットは、シーケンスの3番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット7:4 **SQ2[3:0]**：シーケンスの2番目の変換

これらのビットは、シーケンスの2番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット3:0 **SQ1[3:0]**：シーケンスの1番目の変換

これらのビットは、シーケンスの1番目の変換に割り当てられたチャネル番号を使いソフトウェアによってプログラムされます。0b1111はシーケンスの終了を示します。

0b1111（シーケンスの終了）が低いシーケンスチャネルにプログラムされると、これらのビットは無視されます。

チャネル選択の定義はSQ8[3:0] を参照してください。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.11 ADCウォッチドッグ閾値レジスタ（ADC_AWD3TR）

アドレスオフセット：0x2C

リセット値：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT3 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LT3 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **HT3[11:0]**：*アナログウォッチドッグ3高閾値*

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LT3[11:0]**：アナログウォッチドッグ *3* 低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。
セクション 34.4.25： 1384ページのアナログウィンドウウォッチドッグを参照してください。

34.7.12 ADCデータレジスタ（ADC_DR）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **DATA[15:0]**：変換されたデータ

これらのビットは読出し専用です。これらは、最後の変換チャネルの変換結果を含んでいます。図 285： 1377ページのデータの配置と分解能（オーバーサンプリング無効：OVSE = 0）に示すように、データは左詰めまたは右詰めされています。

較正の完了直後、DATA[6:0] は較正係数を含んでいます。

34.7.13 ADC電源制御レジスタ（ADC_PWRR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VREFSECSMP	VREFPROT	DPD	AUTOFF
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **VREFSECSMP**：V_{REF+} の2番目のサンプルビット

このビットは、ソフトウェアによってセット／クリアされます。複数のADCが同時に動作していて、1クロック分周が使用されている場合に、2番目のV_{REF+} の保護を有効化／無効化するために使用されます。

0：V_{REF+} の2番目のサンプルが無効です。

1：V_{REF+} の2番目のサンプルが有効です。

注： ソフトウェアは、ADENビットが0にクリアされている（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット2 **VREFPROT**：V_{REF+} の保護ビット

このビットは、ソフトウェアによってセット／クリアされます。複数のADCが同時に動作していて、クロック分周回路が使用されている場合に、V_{REF+} の保護を有効化／無効化するために使用されます。

0：V_{REF} の保護は無効です。

1：V_{REF} の保護は有効です。

注： ソフトウェアは、ADENビットが0にクリアされている（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット1 **DPD**：ディープ･パワー･ダウン･モード･ビット

このビットは、ソフトウェアによってセット／クリアされます。ADCが使用されていない場合に、自律モードでディープ･パワー･ダウン･モードを有効化／無効化するために使用されます。

0：ディープ･パワー･ダウン･モードは無効です。

1：ディープ･パワー･ダウン･モードは有効です。

注： ソフトウェアは、ADENビットが0にクリアされている（変換が実行中でない）ときのみ、このビットに書き込むことができます。

オートオフモードでDPDをセットすると、自動的にLDOが無効化されます。

ビット0 **AUTOFF**：オートオフモードビット

このビットは、オートオフモードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オートオフモードは無効です。

1：オートオフモードは有効です。

注： ソフトウェアは、ADENビットが0にクリアされている（変換が実行中でない）ときのみ、このビットに書き込むことができます。

34.7.14 ADCアナログウォッチドッグ2設定レジスタ（ADC_AWD2CR）

アドレスオフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH23	AWD2CH22	AWD2CH21	AWD2CH20	AWD2CH19	AWD2CH18	AWD2CH17	AWD2CH16
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH15	AWD2CH14	AWD2CH13	AWD2CH12	AWD2CH11	AWD2CH10	AWD2CH9	AWD2CH8	AWD2CH7	AWD2CH6	AWD2CH5	AWD2CH4	AWD2CH3	AWD2CH2	AWD2CH1	AWD2CH0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:0 **AWD2CHx**：アナログウォッチドッグチャネルの選択（x = 23 ~ 0）

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ2（AWD2）によって保護される入力チャネルを有効化して選択します。

0：ADCアナログチャネルxはAWD2によって監視されません。

1：ADCアナログチャネルxはAWD2によって監視されます。

注： ADC_AWD2CRで選択したチャネルは、ADC_CHSELRレジスタにも設定する必要があります。チャネル選択の定義はSQ8[3:0] を参照してください。ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.15 ADCアナログウォッチドッグ3設定レジスタ（ADC_AWD3CR）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH23	AWD3CH22	AWD3CH21	AWD3CH20	AWD3CH19	AWD3CH18	AWD3CH17	AWD3CH16
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH15	AWD3CH14	AWD3CH13	AWD3CH12	AWD3CH11	AWD3CH10	AWD3CH9	AWD3CH8	AWD3CH7	AWD3CH6	AWD3CH5	AWD3CH4	AWD3CH3	AWD3CH2	AWD3CH1	AWD3CH0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:0 **AWD3CHx**：アナログウォッチドッグチャネルの選択（x = 23 ~ 0）

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ3（AWD3）によって保護される入力チャネルを有効化して選択します。

0：ADCアナログチャネルxはAWD3によって監視されません。

1：ADCアナログチャネルxはAWD3によって監視されます。

注： ADC_AWD3CRで選択したチャネルは、ADC_CHSELRレジスタにも設定する必要があります。チャネル選択の定義はSQ8[3:0] を参照してください。ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

34.7.16 ADC較正係数（ADC_CALFACT）

アドレスオフセット：0xC4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **CALFACT[6:0]**：較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

– 較正が完了すると、ハードウェアによって較正係数で更新されます。

– ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログADCに格納されている現在のものと異なる場合は、新しい較正の起動時に適用されます。

– 較正の完了直後、DATA[6:0] は較正係数を含んでいます。

注： ソフトウェアは、ADEN = 1のときのみ（ADCが有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。

34.7.17 ADCオプションレジスタ（ADC_OR）

アドレスオフセット：0xD0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHN21SEL
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **CHN21SEL**：チャネル21選択ビット

このビットは、ソフトウェアによってセット／クリアされます。これは、ADC入力チャネル21に接続された内部ソースを選択するために使用されます。:

0：dac1_out1が選択されます。

1：dac1_out2が選択されます。

注： ソフトウェアは、ADSTPに1 = 0を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、このビットに書き込むことができます。

34.7.18 ADC共通設定レジスタADC_CCR）

アドレスオフセット：0x308

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	VSENSESEL	VREFEN	PRESC[3:0]				Res.	Res.
							rw	rw	rw	rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **VBATEN**：V_{BAT} の有効化

このビットは、V_{BAT} チャネルを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット23 **VSENSESEL**：温度センサの選択

このビットは、温度センサを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：温度センサは無効です。

1：温度センサは有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット22 **VREFEN**：V_{REFINT} の有効化

このビットは、VREFINT を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} は無効です。

1：V_{REFINT} は有効です。

注： ADSTPに1を書き込むことによってADSTARTビットが0にクリアされている（変換が実行中でない）ときのみ、ソフトウェアでこのビットに書き込むことができます。

ビット21:18 **PRESC[3:0]**：ADCプリスケーラ

ADCへのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。

0000：ADC入力クロックは分周されません。
0001：ADC入力クロックが2分周されます。
0010：ADC入力クロックが4分周されます。
0011：ADC入力クロックが6分周されます。
0100：ADC入力クロックが8分周されます。
0101：ADC入力クロックが10分周されます。
0110：ADC入力クロックが12分周されます。
0111：ADC入力クロックが16分周されます。
1000：ADC入力クロックが32分周されます。
1001：ADC入力クロックが64分周されます。
1010：ADC入力クロックが128分周されます。
1011：ADC入力クロックが256分周されます。
その他：予約済み

注： ADCが無効のときだけ（ADCAL = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、およびADEN = 0）、ソフトウェアでこれらのビットに書き込むことができます。

ビット17:0 予約済みであり、リセット値に保持する必要があります。

34.8 ADCレジスタ·マップ

次の表はADCレジスタの一覧です。

表334.ADCレジスタマップとリセット値

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADC_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDORDY	EOCAL	Res.	AWD3	AWD2	AWD1	Res.	Res.	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																				0	0		0	0	0			0	0	0	0	0
0x04	**ADC_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDORDYIE	EOCALIE	Res.	AWD3IE	AWD2IE	AWD1IE	Res.	Res.	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																				0	0		0	0	0			0	0	0	0	0
0x08	**ADC_CR**	ADCAL	Res.	Res.	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADSTP	Res.	ADSTART	ADDIS	ADEN
	リセット値	0			0																								0		0	0	0
0x0C	**ADC_CFGR1**	Res.	AWDCH[4:0]					Res.	Res.	AWD1EN	AWD1SGL	CHSELRMOD	Res.	Res.	Res.	Res.	DISCEN	Res.	WAIT	CONT	OVRMOD	EXTEN[1:0]		Res.	EXTSEL[2:0]			ALIGN	SCANDIR	RES[1:0]		DMACFG	DMAEN
	リセット値		0	0	0	0	0			0	0	0					0		0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x10	**ADC_CFGR2**	Res.	Res.	LFTRIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOVS	OVSS[3:0]			OVSR [2:0]			Res.	OVSE	
	リセット値			0																				0	0	0	0	0	0	0	0		0
0x14	**ADC_SMPR**	SMPSEL23.	SMPSEL22	SMPSEL21.	SMPSEL20.	SMPSEL19.	SMPSEL18	SMPSEL17	SMPSEL16	SMPSEL15	SMPSEL14	SMPSEL13	SMPSEL12	SMPSEL11	SMPSEL10	SMPSEL9	SMPSEL8	SMPSEL7	SMPSEL6	SMPSEL5	SMPSEL4	SMPSEL3	SMPSEL2	SMPSEL1	SMPSEL0	Res.	SMP2 [2:0]			Res.	SMP1 [2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0		0	0	0
0x18 ~0x1C	予約済み	予約済み																															
0x20	**ADC_AWD1TR**	Res.	Res.	Res.	Res.	HT1 [11:0]												Res.	Res.	Res.	Res.	LT1 [11:0]											
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x24	**ADC_AWD2TR**	Res.	Res.	Res.	Res.	HT2 [11:0]												Res.	Res.	Res.	Res.	LT2 [11:0]											
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x28	**ADC_CHSELR**（CHSELRMOD = 0）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHSEL23	CHSEL22	CHSEL21	CHSEL20	CHSEL19	CHSEL18	CHSEL17	CHSEL16	CHSEL15	CHSEL14	CHSEL13	CHSEL12	CHSEL11	CHSEL10	CHSEL9	CHSEL8	CHSEL7	CHSEL6	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**ADC_CHSELR**（CHSELRMOD = 1）	SQ8 [3:0]				SQ7 [3:0]				SQ6 [3:0]				SQ5 [3:0]				SQ4 [3:0]				SQ3 [3:0]				SQ2 [3:0]				SQ1 [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**ADC_AWD3TR**	Res.	Res.	Res.	Res.	HT3 [11:0]												Res.	Res.	Res.	Res.	LT3 [11:0]											
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x30～ 0x3C	予約済み	予約済み																															
0x40	**ADC_DR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表334.ADCレジスタマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	**ADC_PWRR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VREFSECSMP	VREFPROT	DPD	AUTOFF
	リセット値																													0	0	0	0
0x48 - 0xBF	予約済み	予約済み																															
0xA0	**ADC_AWD2CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH23	AWD2CH22	AWD2CH21	AWD2CH20	AWD2CH19	AWD2CH18	AWD2CH17	AWD2CH16	AWD2CH15	AWD2CH14	AWD2CH13	AWD2CH12	AWD2CH11	AWD2CH10	AWD2CH9	AWD2CH8	AWD2CH7	AWD2CH6	AWD2CH5	AWD2CH4	AWD2CH3	AWD2CH2	AWD2CH1	AWD2CH0
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	**ADC_AWD3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH23	AWD3CH22	AWD3CH21	AWD3CH20	AWD3CH19	AWD3CH18	AWD3CH17	AWD3CH16	AWD3CH15	AWD3CH14	AWD3CH13	AWD3CH12	AWD3CH11	AWD3CH10	AWD3CH9	AWD3CH8	AWD3CH7	AWD3CH6	AWD3CH5	AWD3CH4	AWD3CH3	AWD3CH2	AWD3CH1	AWD3CH0
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4 - 0xC0	予約済み	予約済み																															
0xC4	**ADC_CALFACT**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT[6:0]						
	リセット値																										0	0	0	0	0	0	0
0xB8 - 0xCF	予約済み	予約済み																															
0xD0	**ADC_OR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHN21SEL
	リセット値																																0
0x308	**ADC_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	VSENSESEL	VREFEN	PRESC3	PRESC2	PRESC1	PRESC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0	0	0	0	0	0	0																		

レジスタ境界アドレスについては136ページのセクション 2.3を参照してください。

35 D/Aコンバータ（DAC）

35.1 概要

DACモジュールは、12bitの電圧出力デジタルアナログコンバータです。DACは、8または12bitモードで設定でき、DMAコントローラと組み合わせて使用することもできます。12bitモードでは、データを左詰めまたは右詰めにできます。DACには2つの出力チャネルがあり、それぞれがコンバータを搭載しています。デュアルDACチャネルモードでは、変換は独立して行うか、両方のチャネルが同期更新操作のためにグループ化されているときには同時に行うことができます。精度を高めるために、入力基準ピンV_{REF+}（その他のアナログペリフェラルと共用）を使用することができます。内部基準も、同じ入力でセットできます。電圧基準バッファ（VREFBUF）セクションを参照してください。

DAC出力が出力パッドから切断され、オンチップペリフェラルに接続されている場合、DACx_OUTyピンは汎用入力／出力（GPIO）として使用できます。駆動出力電流を上げるために、オプションでDAC出力バッファを有効にできます。個別の較正が各DAC出力チャネルに適用可能です。D/Aコンバータ出力チャネルは低消費電力モード（サンプルおよびホールドモード）をサポートしています。

35.2 DACの主な機能

DACの主な機能は以下のとおりです（図 302：デュアルチャネル DAC ブロック図を参照）。

- 1つの DACインタフェースで、最大2つの出力チャネル
- 12bitモードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波および三角波生成
- デュアルDACチャネルの独立または同時変換
- DMAアンダーランエラー検出を含む各チャネルの DMA機能
- バス活動を軽減するダブルデータDMA機能
- 変換外部トリガ
- DAC出力チャネルバッファ／バッファ無モード
- バッファオフセット較正
- 各DAC出力はDACx_OUTy出力ピンから切り離し可能
- オン-チップペリフェラルへのDAC出力接続
- STOPモードの低消費電力動作でのサンプルおよびホールドモード
- システムの電力消費を抑えるための自律モード
- V_{REF+} ピンまたは内部VREFBUF基準からの入力電圧基準

図302はDACチャネルのブロック図を、表336はピンの概要を示します。

35.3 DACの実装

表335.DACの機能

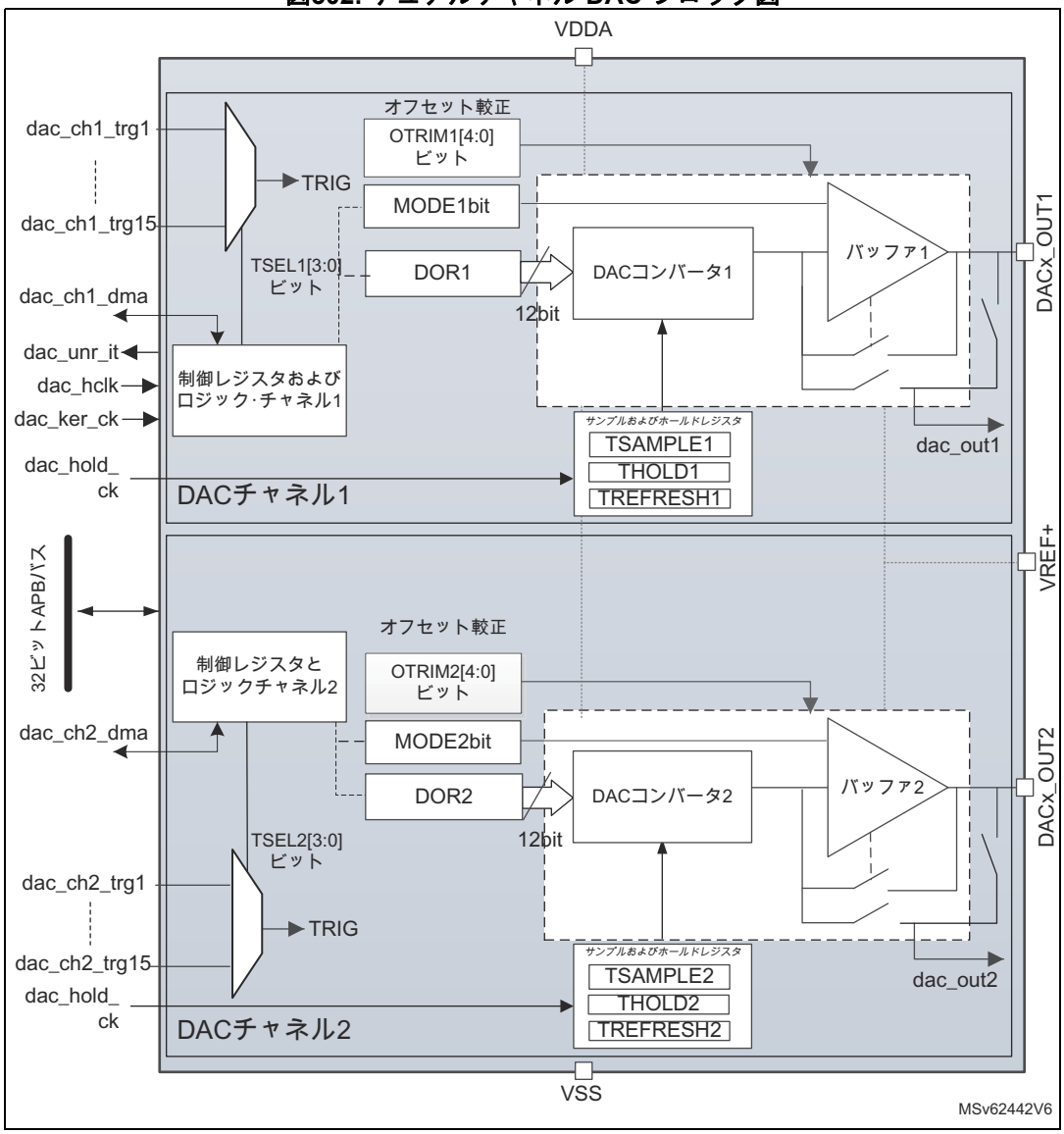
DACの機能	DAC1
デュアルチャネル	X
出力バッファ	X
I/O接続	PA4へDAC1_OUT1、PA5へDAC1_OUT2
最大サンプリング時間	1Msps
自律モード(1)	X
VREF+ ピン	X

1. 自律モードは、STOP 0、STOP 1、STOP 2モードでサポートされています。

35.4 DACの機能説明

35.4.1 DACブロック図

図302. デュアルチャネル DAC ブロック図



1. DAC_MCRのMODEx ビットによって、出力モードが制御され、バッファ有／バッファ無し設定の通常モードとサンプルおよびホールドモードを切り替えることができます。
2. チャネル2の有無については、セクション 35.3 : DACの実装を参照してください。

35.4.2 DACピンおよび内部信号

DACには次の要素が含まれます。

- 最大2つの出力チャネル
- DACx_OUTyを出力ピンから切り離して通常のGPIOとして使用可能
- dac_outx は、コンパレータ、オペアンプ、およびADC（利用可能な場合）などを含むオンチップペリフェラルへの内部ピン接続が使用可能
- DAC出力チャネルのバッファ有またはバッファ無
- スタティック変換のために LSI/LSE クロックソース（dac_hold_ck）を使用し、STOPモードで動作可能なサンプルおよびホールドブロックレジスタ。

DACには最大2つの個別出力チャネルが搭載されています。各出力チャネルは、コンパレータ、オペアンプ、およびADC（使用できる場合）などのオンチップペリフェラルに接続できます。この場合、DAC出力チャネルは DACx_OUTy 出力ピンから切断でき、対応するGPIOを別の目的に使用できます。

DAC出力はバッファの有無を設定できます。サンプルホールドブロックと、それに関連するレジスタは、LSI/LSE クロックソース（dac_hold_ck）を使用してSTOPモードで動作可能です。

表336.DACの入出力ピン

ピン名	信号タイプ	説明
VREF+	入力、正アナログ基準電圧	DAC、$V_{REF+} \le V_{DDAmax}$ 用のハイレベル／正基準電圧（データシートを参照）
VDDA	入力、アナログ電源	アナログ電源
VSSA	入力、アナログ電源グラウンド	アナログ電源のグラウンド
DACx_OUTy	アナログ出力信号	DACxチャネル y のアナログ出力

表337.DAC内部入力／出力信号

内部信号名	信号タイプ	説明
dac_ch1_dma	双方向	D/Aコンバータチャネル1 DMAリクエスト／確認応答
dac_ch2_dma	双方向	D/Aコンバータチャネル2 DMAリクエスト／確認応答
dac_ch1_trgx（x = 1 ~ 15）	入力	DAC channel1トリガ入力
dac_ch2_trgx（x = 1 ~ 15）	入力	DAC channel2トリガ入力
dac_unr_it	出力	DACアンダーラン割込み
dac_hclk	入力	DACペリフェラルクロック
dac_ker_ck	入力	DACカーネルクロック
dac_hold_ck	入力	サンプルおよびホールドモードで使用されるD/Aコンバータ低消費電力クロック
dac_out1	アナログ出力	オンチップペリフェラル用DAC channel1出力
dac_out2	アナログ出力	オンチップペリフェラル用DAC channel2出力

表338.DACの相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	ck_lsi or ck_lse	RCCで選択されたLSIまたはLSEクロック
dac_chx_trg1（x = 1、2）	tim1_trgo	オンチップタイマからの内部信号
dac_chx_trg2（x = 1、2）	tim2_trgo	オンチップタイマからの内部信号
dac_chx_trg3（x = 1、2）	tim4_trgo	オンチップタイマからの内部信号
dac_chx_trg4（x = 1、2）	tim5_trgo	オンチップタイマからの内部信号
dac_chx_trg5（x = 1、2）	tim6_trgo	オンチップタイマからの内部信号
dac_chx_trg6（x = 1、2）	tim7_trgo	オンチップタイマからの内部信号
dac_chx_trg7（x = 1、2）	tim8_trgo	オンチップタイマからの内部信号
dac_chx_trg8（x = 1、2）	tim15_trgo	オンチップタイマからの内部信号
dac_chx_trg11（x = 1、2）	lptim1_ch1	オンチップタイマからの内部信号
dac_chx_trg12（x = 1、2）	lptim3_ch1	オンチップタイマからの内部信号
dac_chx_trg13（x = 1、2）	exti9	外部ピン

35.4.3 DACクロック

DACの更新用に2つのクロックソースを使用できます。

- dac_hclk：DACペリフェラルクロック（AHBクロック）
- dac_ker_ck : DACカーネル･クロック：このクロックはDACとADCを同期させるために使用できます。
- dac_hold_ck：サンプルおよびホールドモードで使用される低消費電力クロック

DACクロックはRCCで選択されます。

35.4.4 DACチャネルイネーブル

各 DACチャネルは、DAC_CRレジスタの対応するENx ビットをセットすることによって起動できます。DACチャネルは、t_{WAKEUP} のスタートアップ時間後に有効になります。

DACインタフェースがデータを受信する準備ができているとき、DAC_SRレジスタのDAC x RDYビットがセットされます。DACxRDY信号がリセットされているときにENxビットがセットされている場合、新しいデータの書込みまたはトリガのアサートはできません。

注： **ENx ビットは、アナログDACチャネル x のみを有効にします。DACチャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。**

35.4.5 DACデータフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- シングルDACチャネル

  この場合、次の3つの設定が可能です。

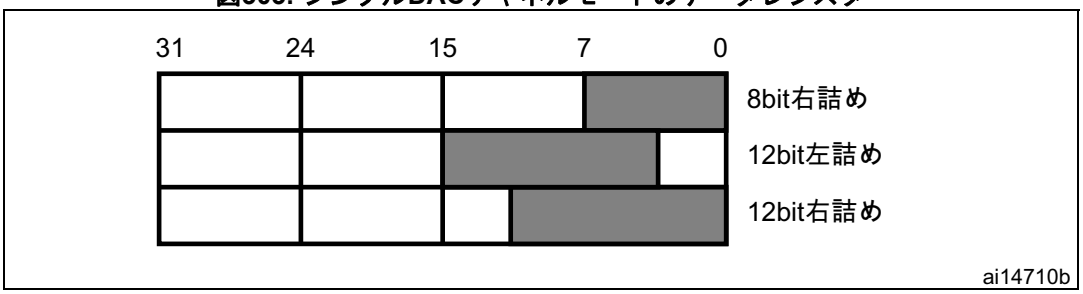
  – 8bit右詰め : ソフトウェアはデータをDAC_DHR8Rx[7:0] ビットにロードする必要があります（DHRx[11:4] ビットに格納）。

  – 12bit左詰め : ソフトウェアはデータをDAC_DHR12Lx [15:4] ビットにロードする必要があります（DHRx[11:0] ビットに格納）。

  – 12bit右詰め : ソフトウェアはデータをDAC_DHR12Rx [11:0] ビットにロードする必要があります（DHRx [11:0] ビットに格納）。

ユーザによって書き込まれたデータは、ロードされたDAC_DHRyyyxレジスタに応じて、シフトされてから、対応するDHRx（メモリマップされない内部レジスタであるデータ保持レジスタx）に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガまたは外部イベントトリガによって、DORx レジスタにロードされます。

図303. シングルDACチャネルモードのデータレジスタ



- デュアルDACチャネル（使用できる場合）

  この場合、次の3つの設定が可能です。

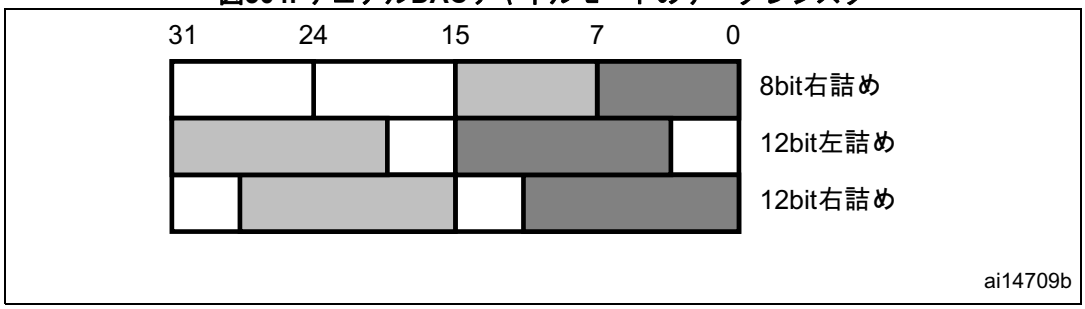
  – 8bit右詰め : DACチャネル1のデータは、DAC_DHR8RD [7:0] ビット（DHR1[11:4] ビットに格納）に、DACチャネル2のデータは、DAC_DHR8RD [15:8] ビット（DHR2[11:4] ビットに格納）にロードされます。

  – 12bit左詰め : DACチャネル1のデータは、DAC_DHR12LD [15:4] ビット（DHR1[11:0] ビットに格納）に、DACチャネル2のデータは、DAC_DHR12LD [31:20] ビット（DHR2[11:0] ビットに格納）にロードされます。

  – 12bit右詰め : DACチャネル1のデータは、DAC_DHR12RD [11:0] ビット（DHR1[11:0] ビットに格納）に、DACチャネル2のデータは、DAC_DHR12RD [27:16] ビット（DHR2[11:0] ビットに格納）にロードされます。

ロードされたDAC_DHRyyyDレジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1およびDHR2（Data Holding Register、メモリマップされない内部レジスタ）に格納されます。その後、DHR1およびDHR2レジスタは、自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、それぞれDAC_DOR1およびDOR2レジスタにロードされます。

図304. デュアルDACチャネルモードのデータレジスタ



符号付き/符号なしデータ

DAC入力データは符号なし:12bitモードでは、0x000は最小値に対応し、0xFFFは最大値に対応します。

DACは2の補数形式の符号付き入力データも扱うことができます。このためには、DAC_MCRレジスタのSINFORMATxビットをセットします。

SINFORMATxビットがセットされると、DHRxレジスタに書き込まれたデータのMSBビットは、DAC_DORxレジスタにコピーされるときに反転され、DACインタフェースは符号付データを受け入れることができます（Q1.15、Q1.11、またはQ1.7形式）。DAC_DHR12Lxレジスタを使用して、16bit符号付きデータをデータ保持レジスタに格納することができます。16bitデータの12 MSBは、DAC出力データに使用され、MSBビットは反転されます。4つのLSBは単純に無視されます。

表339.データフォーマット（12bitデータの場合）

SINFORMATxビット	DHRxレジスタに書き込まれるデータ	DORxレジスタに転送されるデータ
0	0x000	0x000
0	0xFFF	0xFFF
1	0x7FF	0xFFF
1	0x000	0x800
1	0xFFF	0x7FF
1	0x800	0x000

35.4.6 DAC変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって（DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx、DAC_DHR8RD、DAC_DHR12RD、またはDAC_DHR12LD への書込み操作）、DACチャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない（DAC_CRレジスタのTEN x ビットがリセットされている）場合に、1 dac_hclk クロックサイクル後にDAC_DORxレジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている（DAC_CRレジスタのTENx ビットがセットされている）ときには、トリガが発生すると、転送はトリガ信号の3 dac_hclk クロックサイクル後に行われます。

DAC_DORx にDAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される$t_{SETTLING}$ 時間後にアナログ出力電圧が使用可能になります。

DACとADCを同期させるために、両方のペリフェラルで同じクロックソースを使用することができます。そのためには、RCCでdac_hclkクロック（AHBクロック）の代わりにdac_ker_ckクロックを選択します。

dac_hclk またはdac_ker_ck クロック速度が80MHzを超える場合は、DAC_MCRのHFSELビットをセットする必要があります。それによって、DAC_DHRxレジスタからDAC_DORxレジスタへの転送に余分の遅延が追加されます。

HFSELビットとdac_hclkクロック周波数に応じたDAC_DORxの更新速度の制限については、以下の「HFSELの説明」の表を参照してください。

許可されない期間にデータが更新された場合、またはソフトウェア/ハードウェアトリガイベントが発生した場合、ペリフェラルの動作は予期できません。

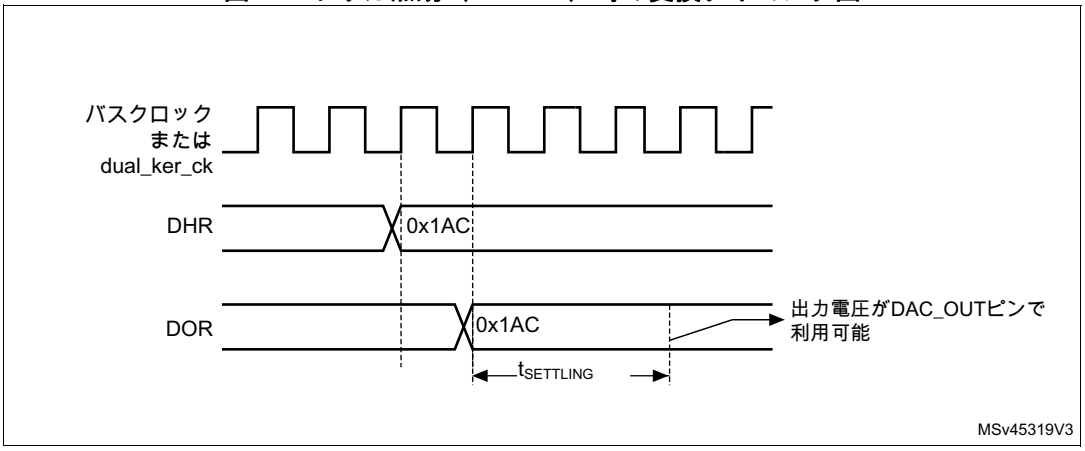
上記のタイミングは、DACインタフェースの制限にのみ関係します。製品データシートの$t_{SETTLING}$パラメータも参照してください。

表340.HFSELの説明

HFSEL [1:0]	AHB周波数	AHBクロック（dac_hclk）を使用した遅延	dac_ker_ck クロックを使用した遅延	機能
00	< 80MHz	3	4	DAC_DORの更新速度は最大3 AHB クロックサイクルまたは 4 dac_ker_ckサイクル。
01	≥ 80MHz(1)	5	5	DAC_DORの更新速度は最大5 AHB クロックサイクルまたは 5 dac_ker_ckサイクル。
10	≥ 160MHz	7	6	DAC_DORの更新速度は最大7 AHB クロックサイクルまたは 6 dac_ker_ckサイクル。
11	予約済みです。	-	-	-

1. 最大のAHB周波数の値については、デバイスデータシートを参照してください。

図305. トリガ無効（TEN = 0）時の変換タイミング図



35.4.7 DAC出力電圧

デジタル入力は、0からV_{REF+}までのリニア変換で出力電圧に変換されます。

各 DACチャネルピンのアナログ出力電圧は、次の式によって求められます。

$$\text{DACoutput} = V_{REF} \times \frac{DOR}{4096}$$

35.4.8 DACトリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割込みラインなど）によって変換をトリガできます。DAC_CR レジスタの TSELx[3:0] ビットに示すように、TSELx[3:0] 制御ビットは、変換をトリガするイベントを 16 のイベントから決定します。これらのイベントは、ソフトウェアトリガまたはハードウェアトリガのいずれかです。セクション 35.4.2 : DAC ピンおよび内部信号の相互接続の表を参照してください。

DACインタフェースが選択されたトリガソース（次の表を参照）で立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータがDAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから3 dac_hclk サイクル後に更新されます。

ソフトウェアトリガが選択されている場合、変換は、SWTRIGビットがセットされると開始されます。SWTRIGビットは、DAC_DORx レジスタの内容がDAC_DHRx レジスタにロードされると、ハードウェアによってリセットされます。

注： **ENx ビットがセットされているときには、TSELx[3:0] ビットを変更することはできません。**

ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタからDAC_DORx レジスタへの転送は、わずか1 dac_hclk クロックサイクルで行われます。

35.4.9 DMAリクエスト

各 DACチャネルは、DMA機能を備えています。DACチャネルのDMAリクエストは、2つのDMA チャネルを使用して処理されます。

DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガ以外）が発生した場合、DAC_DHRx レジスタの値はDAC_DORx レジスタに転送され、転送が終了すると、DMAリクエストが生成されます。

デュアルモードでは、両方のDMAENxビットがセットされている場合、2つのDMAリクエストが生成されます。1つのDMAリクエストしか必要ない場合には、対応するDMAENxビットだけをセットする必要があります。これにより、アプリケーションは1つのDMAリクエストと一意なDMAチャネルを使用して、両方のDACチャネルをデュアルモードで管理することがきます。

DAC_DHRx からDAC_DORx へのデータ転送はDMAリクエストの前に起きるので、最初のトリガイベントが発生する前に最初のデータをDAC_DHRx に書き込む必要があります。

DMAアンダーラン

DAC DMAリクエストはキューされないので、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に2つ目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタのDMAチャネル x アンダーランフラグDMAUDRx がセットされてエラー条件を報告します。DACチャネル x は、古いデータを変換し続けます。

ソフトウェアは、1を書き込むことによってDMAUDRx フラグをクリアし、使用されたDMAストリームのDMAENビットをクリアし、DMAとD/Aコンバータの両方のチャネル x を再初期化して転送を正しくリスタートさせる必要があります。また、D/Aコンバータトリガ変換周波数を変更するかDMAの負荷を軽減して、新しいDMAアンダーランを回避する必要があります。最後に、DMAデータ転送と変換トリガを有効にすることによってDAC変換を再開することができます。

各DACチャネルx では、DAC_CRレジスタの対応するDMAUDRIEx ビットが有効にされている場合、割込みも生成されます。

DMAダブルデータモード

DMAコントローラが通常モードで使用されるときには、DMAリクエストによって12bit（または8bit）データのみが転送されます。AHB幅は32bitなので、2つの12bitデータを同時に転送できます。このモードを使用するには、DAC_MCRレジスタのDMADOUBLExビットをセットします。

DAC DMAリクエストは、DMAENxビットがセットされているとき、外部トリガ（ソフトウェアトリガでなく）が2つ発生するたびに生成されます。

1. 最初のトリガが検出されると、DAC_DHRxおよびDAC_DHRB x レジスタの値がDAC_DORxおよびDAC_DORB x レジスタに転送されます。実際のDACデータがDAC_DORxレジスタにロードされます。その後、DMAリクエストが生成されます。DMAは、新しいデータをDAC_DHR x およびDAC_DHRB x データレジスタに書き込みます。
2. 次のトリガが検出されると、実際のDACデータがDAC_DHRBxレジスタにロードされます。この2つ目のトリガでは、DMAリクエストは生成されません。DORSTATxビットは、アナログDAC入力に実際にロードされるDORデータを示します。

DMAダブルデータモードでは、DMAアンダーラン機能もサポートされます。

DMAダブルモードでは、DMAリクエストは1つのDACチャネルのみを処理できます。DMAダブルモードで2つのチャネル出力を使用するには、各DMAチャネルを個別に設定する必要があります。

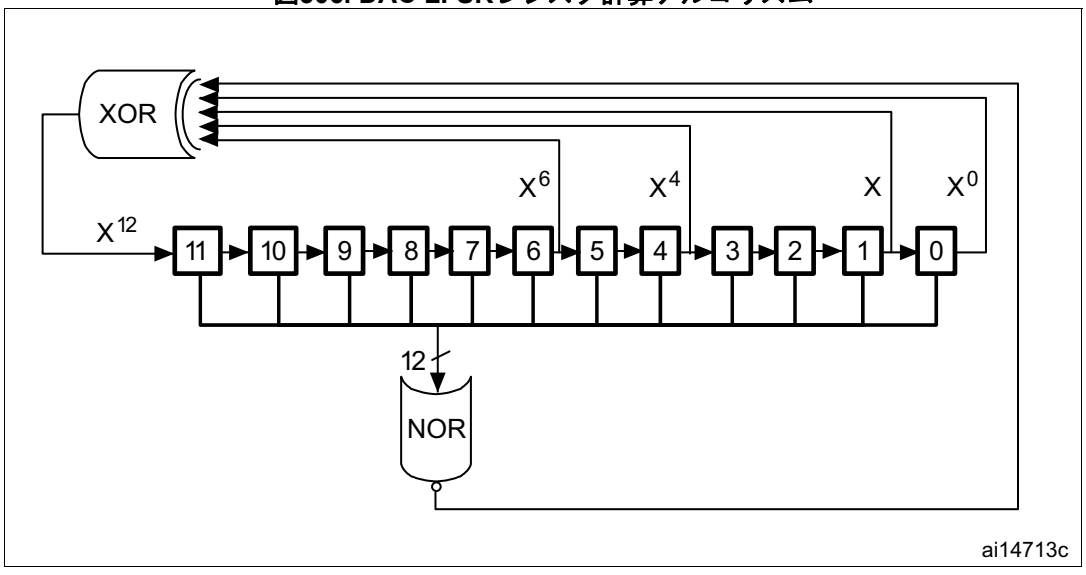
ダブルデータモードからシングルデータモードに、またはその逆に変更するには、以下の条件が満たされている必要があります。

- DACを無効にする必要があります。
- DMAENビットをクリアする必要があります（ENx = 0かつDMAEN = 0）。

35.4.10 ノイズ生成

リニアフィードバックシフトレジスタ（LFSR）を使用して、可変振幅の擬似ノイズを生成することができます。D/Aコンバータノイズ生成を選択するには、WAVEx[1:0] を01にセットします。LFSRにプリロードされる値は0xAAAです。このレジスタは、各トリガイベントの3 dac_hclk クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図306. DAC LFSRレジスタ計算アルゴリズム

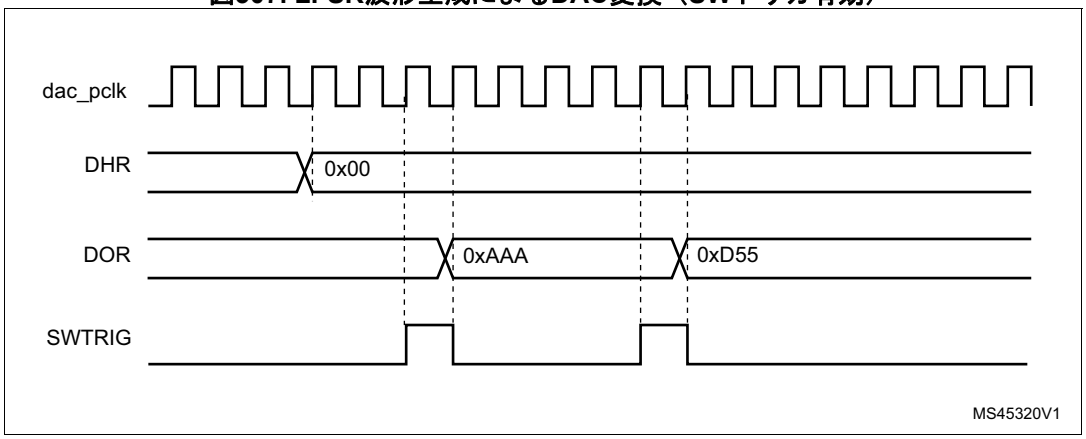


LFSR値は、DAC_CRレジスタのMAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしにDAC_DHRx の内容に加算され、DAC_DORx レジスタに転送されます。

LFSRが0x0000の場合、“1”がインジェクトされます（アンチロックアップメカニズム）。

WAVEx[1:0] ビットをリセットすることによって、LFSR波形生成をリセットできます。

図307. LFSR波形生成によるDAC変換（SWトリガ有効）



注： **ノイズ生成のためには、DAC_CRレジスタのTENx ビットをセットすることによって、DACトリガを有効にしなければなりません。**

35.4.11 三角波生成

DCまたは低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC三角波生成を選択するには、WAVEx[1:0] を10にセットします。振幅は、DAC_CRレジスタのMAMPx[3:0] ビットによって設定されます。内部三角波カウンタは、各トリガイベントの3 dac_hclk クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしにDAC_DHRx レジスタに加えられ、合計はDAC_DORx レジスタに転送されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは0にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波形生成をリセットできます。

図308. DAC三角波生成

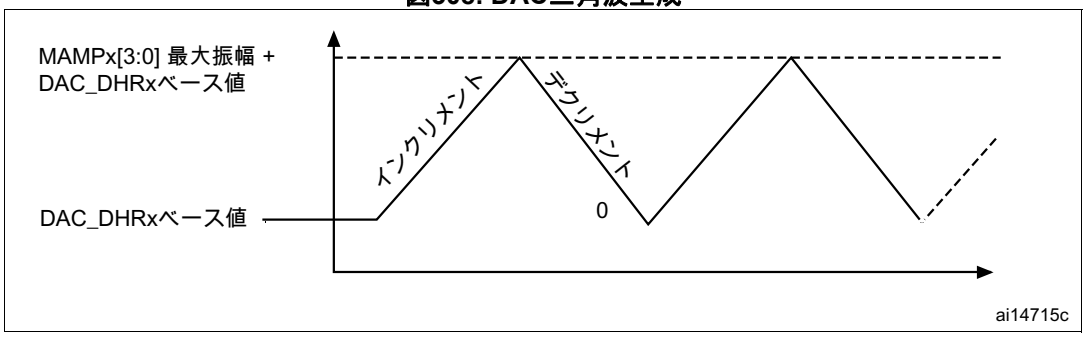
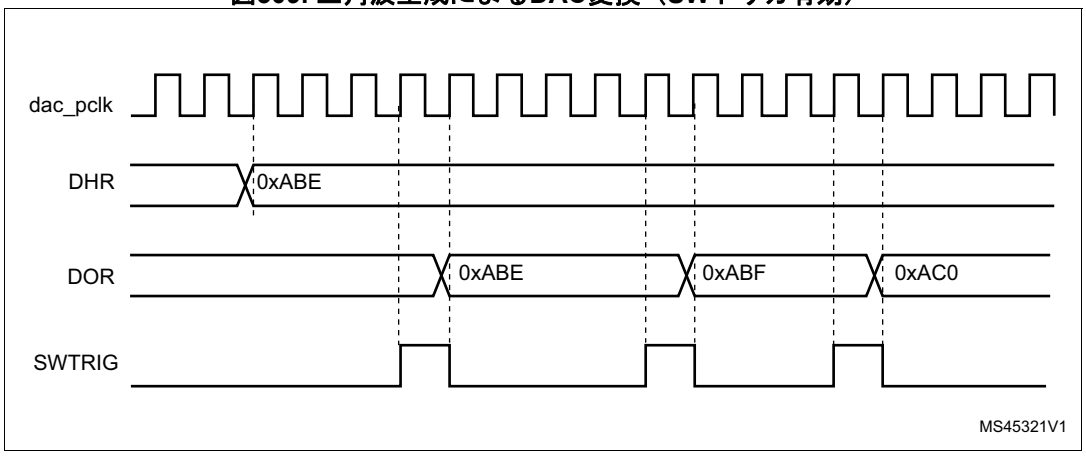


図309. 三角波生成によるDAC変換（SWトリガ有効）



注： **三角波生成のためには、DAC_CRレジスタのTENx ビットをセットすることによって、DACトリガを有効にしなければなりません。**

DACを有効にするには、その前にMAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

35.4.12 D/Aコンバータチャネルモード

各 D/Aコンバータチャネルは、通常モードまたはサンプルおよびホールドモードのいずれかに設定できます。高駆動機能を利用するために、出力バッファを有効にできます。出力バッファを有効にする前に、電圧オフセットを較正する必要があります。この較正は出荷時に実施され（リセット後にロードされ）、アプリケーション動作中にソフトウェアで調整できます。

通常モード

通常モードでは、バッファ状態の変更とDACx_OUTy ピン相互接続の変更による4つの組み合わせがあります。

出力バッファを有効にするには、DAC_MCRレジスタのMODEx[2:0] ビットを次のようにセットする必要があります。

- 000：DACを外部ピンに接続
- 001：DACを外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCRレジスタのMODEx[2:0] ビットを次のようにセットする必要があります。

- 010：DACを外部ピンに接続
- 011：DACをオンチップペリフェラルに接続

サンプルおよびホールドモード

サンプルおよびホールドモードでは、D/Aコンバータコアがトリガ変換でデータを変換してから、変換された電圧をコンデンサで保持します。変換しない場合、DACコアおよびバッファはサンプル間で完全にオフになり、DAC出力はトライステートになるため、全体の消費電力を低減します。新しい変換を行う前ごとに、安定時間（その値はバッファ状態に応じて変わります）が必要になります。

このモードでは、DACコアならびにすべての対応するロジックとレジスタは、dac_hclk クロックに加えて LSI/LSE 低速クロック（dac_hold_ck）によっても駆動されますので、STOPモードなどの超低消費電力モードでDACチャネルを使用できます。

LSI/LSE 低速クロック（dac_hold_ck）は、サンプルおよびホールドモードが有効な状態で停止しないでください。

サンプル／ホールドモードの動作は、3つのフェーズに分けられます。

1. サンプルフェーズ : サンプル／ホールド要素が対象の電圧までチャージされます。チャージ時間はコンデンサ値（ユーザによって選択された内部または外部）によって異なります。サンプリング時間は、DAC_SHSRx レジスタのTSAMPLEx[9:0] ビットで設定されます。TSAMPLEx[9:0] ビットの書込み中、両方のクロックドメイン（AHB および低速クロック）を同期するためにDAC_SRレジスタのBWSTx ビットが1にセットされ、D/Aコンバータチャネル動作中にソフトウェアによってサンプルフェーズの値を変更できるようになります。
2. ホールドフェーズ : DAC出力チャネルはトライステートになり、DACコアおよびバッファはオフになって消費電流を低減します。ホールド時間は、DAC_SHHRレジスタのTHOLDx[9:0] ビットで設定されます。
3. リフレッシュフェーズ : リフレッシュ時間は、DAC_SHRRレジスタのTREFRESHx[7:0] ビットで設定されます。

上記3つのフェーズのタイミングは、LSI/LSE クロック周期の単位で行われます。たとえば、350µsのサンプル時間、2msのホールド時間、100µsのリフレッシュ時間を設定するために、LSI/LSE ～32KHzを選択するとします。

12サイクルがサンプルフェーズで必要になります。TSAMPLEx[9:0] = 11、

62サイクルがホールドフェーズで必要になります。THOLDx[9:0] = 62、

そして、4サイクルがリフレッシュ時間で必要になります。TREFRESHx[7:0] = 4.

この例では、消費電力が通常モードに対してほぼ15分の1に低減されます。

適切なサンプルおよびリフレッシュタイミングを計算する公式は、以下の表に記載しており、ホールド時間はリーク電流によって異なります。

表341.サンプルおよびリフレッシュタイミング

バッファ状態	t_{SAMP} (1)(2)	$t_{REFRESH}$ (2)(3)
イネーブル	7 µs + (10*R_{BON}*C_{SH})	7 µs + (R_{BON}*C_{SH})*ln(2*N_{LSB})
無効化	3 µs + (10*R_{BOFF}*C_{SH})	3 µs + (R_{BOFF}*C_{SH})*ln(2*N_{LSB})

1. **上記の公式で、½ LSBエラー精度での目標のコード値への整定には、12bitの分解能の場合、10定数時間が必要です。8bit分解能の場合、整定時間は7定数時間です。**
2. **C_{SH} はサンプルおよびホールドモードでのコンデンサ容量です。**
3. **ホールドフェーズ中に許容される電圧低下「Vd」は、コンデンサが出力リーク電流で放電した後の LSBの数で表されます。½ LSBエラー精度で目標のコード値へ整定し直すには、DACのln(2*Nlsb) 定数時間が必要です。**

出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例

以下に使用される値は、例示のみを目的としています。製品データについては、製品データシートを参照してください。

C_{SH} = 100 nF

V_{REF+} = 3.0V

サンプリングフェーズ :

t_{SAMP} = 7 µs + (10 * 2000 * 100 * 10^{-9}) = 2.007ms

（ここで、R_{BON} = 2kΩ）

リフレッシュフェーズ：

$t_{REFRESH}$ = 7 µs + (2000 * 100 * 10^{-9}) * ln(2*10) = 606.1 µs

（ここで、N_{LSB} = 10（ホールドフェーズ中は10 LSB低下））

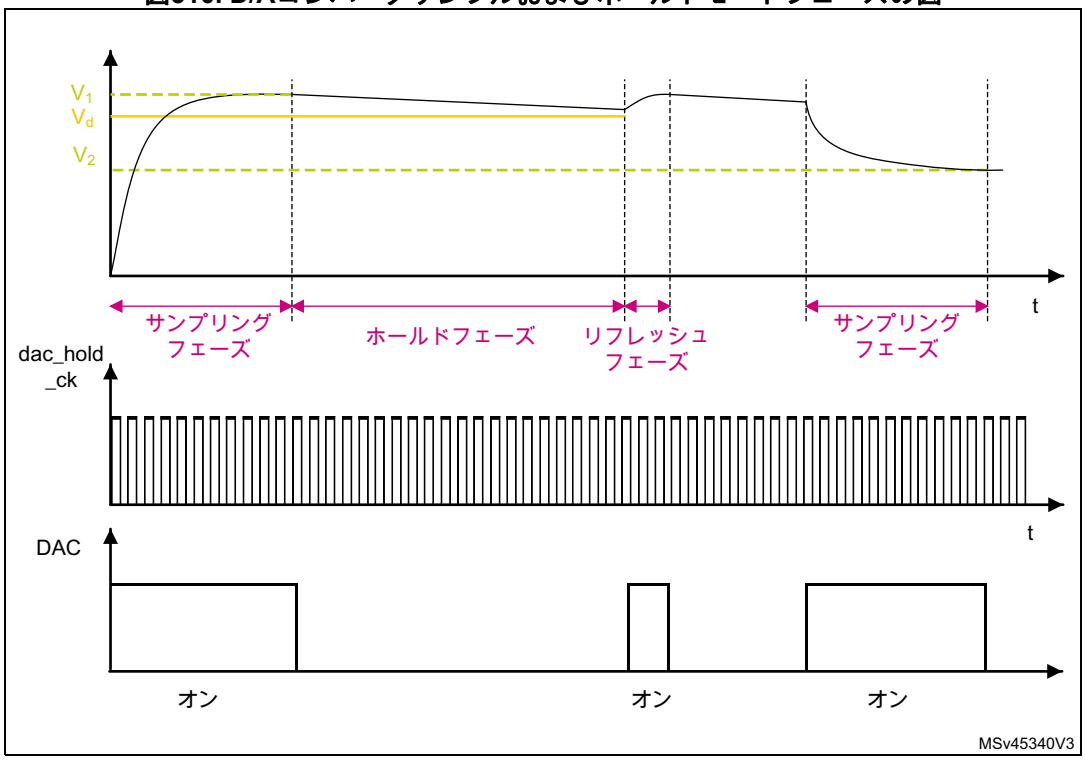
ホールドフェーズ：

$D_V = i_{leak} * t_{hold} / C_{SH}$ = 0.0073V（3Vで12bitの10 LSB）

i_{leak} = 150 nA（すべての温度範囲のIOリークで最悪のケース）

t_{hold} = 0.0073 * 100 * 10^{-9} / (150 * 10^{-9}) = 4.867ms

図310. D/Aコンバータサンプルおよびホールドモードフェーズの図



通常モードと同様に、サンプルおよびホールドモードには個別の設定があります。

出力バッファを有効にするには、DAC_MCRレジスタのMODEx[2:0] ビットを次のようにセットする必要があります。

- 100：DACを外部ピンに接続
- 101：DACを外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCRレジスタのMODEx[2:0] ビットを次のようにセットする必要があります。

- 110：DACを外部ピンとオンチップペリフェラルに接続
- 111：DACをオンチップペリフェラルに接続

  MODEx[2:0] ビットが111と等しい場合、内部コンデンサC_{Lint}は、D/Aコンバータコアの電圧出力を保持し、それによってオンチップペリフェラルを駆動します。

すべてのサンプルおよびホールドフェーズは割込み可能であり、DAC_DHRx のあらゆる変更は新しいサンプルフェーズを即座にトリガします。

表342.チャネル出力モードの概要

MODEx[2:0]			モード	バッファ	出力接続
0	0	0	通常モード	有効	外部ピンに接続
0	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
0	1	0		無効	外部ピンに接続
0	1	1			オンチップペリフェラルに接続（コンパレータなど）
1	0	0	サンプルおよびホールドモード	有効	外部ピンに接続
1	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	0		無効	外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	1			オンチップペリフェラルに接続（コンパレータなど）

35.4.13 DAC出力バッファ較正

Nビットのデジタルアナログコンバータ（DAC）の転送関数は次のとおりです。

$$V_{out} = ((D/2^N) \times G \times V_{ref}) + V_{OS}$$

ここで、VOUTはアナログ出力、Dはデジタル入力、Gはゲイン、Vrefは公称フルスケール電圧、VOSはオフセット電圧です。理想的なDACチャネルでは、G = 1、VOS = 0 となります。

出力バッファの特性によって、オフセット電圧が部分的に異なり、アナログ出力で絶対的なオフセットエラーをもたらす場合があります。VOS を補正するには、トリミング技法による較正が必要です。

この較正は、DACチャネル x がバッファ有効状態で動作している場合（MODEx[2:0] = 0b000、0b001、0b100、0b101のいずれか）のみ有効です。バッファがオフのときに他のモードで適用される場合は影響しません。較正中は次のようになります。

- バッファ出力がピンの内部／外部接続から切断され、トライステートモード（HiZ）になります。
- バッファがミドルコード値0x800を検出して、内部ブリッジによるVREF+/2信号と比較するためにコンパレータとして機能し、比較結果（CAL_FLAGx ビット）に応じて出力信号を0または1にトグルします。

次の2つの較正技法があります。

- 出荷時トリミング（デフォルト設定）
  DACバッファオフセットが出荷時にトリミングされます。DAC_CCR レジスタの OTRIMx[4:0] ビットのデフォルト値は出荷時トリミング値となり、DACデジタルインタフェースがリセットされるとロードされます。
- ユーザトリミング
  動作条件が出荷時トリミング条件と異なる場合、特に V_{DDA} 電圧、温度、VREF+ の値が変化する場合に、ユーザトリミングはソフトウェアによって適用時にいつでも実行できます。

注： **出荷時トリミング条件の詳細については、データシートを参照してください。**

また、VDDが削除される場合（デバイスがSTANDBYモードやVBATモードに移行する場合など）、較正が必要になります。

ユーザトリミング較正を実行する手順は次のとおりです。

1. DACチャネルがアクティブな場合、DAC_CRのENx ビットに0を書き込み、チャネルを無効にします。

2. DAC_MCRレジスタに MODEx[2:0] = 0b000、0b001、0b100、0b101のいずれかを書き込んで、バッファが有効なモードを選択します。
3. DAC_CRレジスタのCENx ビットを1にセットして、D/Aコンバータチャネル x の較正を開始します。
4. トリミングアルゴリズムを適用します。
   a) OTRIMx[4:0] ビットにコード（0b00000 で開始する）を書き込みます。
   b) tTRIM の遅延を待ちます。
   c) DAC_SRのCAL_FLAGx ビットが1にセットされているか確認します。
   d) CAL_FLAGx に1がセットされている場合、OTRIMx[4:0] トリミングコードが検索され、これを出力値を補正するデバイスの操作中に使用できます。それ以外は OTRIMx[4:0] をインクリメントし、（a）から（d）のサブステップを再び繰り返します。

ソフトウェアアルゴリズムによって、逐次比較や二分法を使用して、より早く OTRIMx[4:0] ビットの内容を計算してセットできます。

CAL_FLAGx ビットの転流／トグルは、オフセットが正しく補正されたことを示し、対応するトリミングコードをDAC_CCRレジスタの OTRIMx[4:0] ビットに保持しておく必要があります。

注： **tTRIM** 遅延が、OTRIMx[4:0] ビットへの書込みとDAC_SRレジスタの CAL_FLAGx ビットの読出しの間で守られなければ、正しい値を取得できません。このパラメータは、データシートの電気的特性のセクションで指定されています。

V_{DDA}、VREF+、および温度条件がデバイス動作中に変化せず、STANDBYおよびVBATモードに頻繁に移行する場合、ソフトウェアは最初のユーザ較正時に見つかった OTRIMx[4:0] ビットをFlashまたはバックアップレジスタに格納して、後でデバイスの電源が復帰したときに直接ロード／書き込むことができ、新しい較正時間を待たなくて済むようになります。

CENx ビットがセットされているとき、ENx ビットをセットすることはできません。

35.4.14 デュアルD/Aコンバータチャネル変換モード（デュアルチャネルを使用できる場合）

同時に2つのDACチャネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、およびDHR12LDの3つのデュアルレジスタが搭載されています。両方のDACチャネルを同時に駆動するには、一意なレジスタアクセスが必要です。波形の生成の場合は、DHRxxxDレジスタへのアクセスは必要ありません。したがって、2つの出力チャネルは独立に、または同時に使用できます。

2つのDACチャネルとこれらのデュアルレジスタを使用することで、15 の変換モードが使用可能です。すべての変換モードは、必要な場合には、別のDHRxレジスタを使っても利用できます。

すべてのモードについて、以下の節で説明します。

波形生成なしの独立トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

DACチャネル1トリガが発生すると、DHR1レジスタがDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。

DACチャネル2トリガが発生すると、DHR2レジスタがDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。

1つのLFSR生成による独立トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2つのDACチャネルWAVEx[1:0] ビットを01に設定し、MAMPx[3:0] ビットで同じLFSRマスク値を設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

DACチャネル1トリガが発生すると、同じマスクを持つLFSR1カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1カウンタが更新されます。

DACチャネル2トリガが発生すると、同じマスクを持つLFSR2カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2カウンタが更新されます。

異なるLFSR生成による独立トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2つのDACチャネルWAVEx[1:0] ビットを01に設定し、MAMP1[3:0] ビットとMAMP2[3:0] ビットで異なるLFSRマスク値を設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

DACチャネル1トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つLFSR1カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1カウンタが更新されます。

DACチャネル2トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つLFSR2カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2カウンタが更新されます。

1つの三角波生成による独立トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2つのDACチャネルのWAVEx[1:0] ビットを1xに設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

DACチャネル1トリガが発生すると、同じ三角波振幅を持つDACチャネル1の三角波カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル1の三角波カウンタが更新されます。

DACチャネル2トリガが発生すると、同じ三角波振幅を持つDACチャネル2の三角波カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル2の三角波カウンタが更新されます。

異なる三角波生成による独立トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2ビットに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2つのDACチャネルWAVEx[1:0] ビットを1xに、異なる最大振幅値をMAMP1[3:0] とMAMP2[3:0] ビットに設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

DACチャネル1トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つDACチャネル1の三角波カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル1の三角波カウンタが更新されます。

DACチャネル2トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つDACチャネル2の三角波カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル2の三角波カウンタが更新されます。

同時ソフトウェア開始

DACをこの変換モードに設定するには、次の手順が必要です。

- デュアルDACチャネルデータを目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にロードします。

この設定では、1 dac_hclk クロックサイクル後に、DHR1およびDHR2レジスタがDAC_DOR1とDAC_DOR2にそれぞれ転送されます。

波形生成なしの同時トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに同じ値をセットすることによって、両方のDACチャネルに同じトリガソースを設定します。
3. デュアルDACチャネルデータを目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にロードします。

トリガが発生すると、DHR1およびDHR2レジスタがDAC_DOR1とDAC_DOR2にそれぞれ転送されます（3 dac_hclk クロックサイクル後）。

1つのLFSR生成による同時トリガ

1. DACをこの変換モードに設定するには、次の手順が必要です。
2. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
3. TSEL1およびTSEL2bitフィールドに同じ値をセットすることによって、両方のDACチャネルに同じトリガソースを設定します。
4. 2つのDACチャネルWAVEx[1:0] ビットを01に設定し、MAMPx[3:0] ビットで同じLFSRマスク値を設定します。
5. デュアルDACチャネルデータを目的のDHRレジスタ（DHR12RD、DHR12LD、またはDHR8RD）にロードします。

トリガが発生すると、同じマスクを持つLFSR1カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1カウンタが更新されます。同時に、同じマスクを持つLFSR2カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2カウンタが更新されます。

異なるLFSR生成による同時トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに同じ値をセットすることによって、両方のDACチャネルに同じトリガソースを設定します。
3. 2つのDACチャネルのWAVEx[1:0] ビットを01に設定し、MAMP1[3:0] とMAMP2[3:0] ビットで異なるLFSRマスク値をセットします。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つLFSR1カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1カウンタが更新されます。
同時に、MAMP2[3:0] で設定されたマスクを持つLFSR2カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2カウンタが更新されます。

1つの三角波生成による同時トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに同じ値をセットすることによって、両方のDACチャネルに同じトリガソースを設定します。
3. 2つのDACチャネルのWAVEx[1:0] ビットを1xに設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

トリガが発生すると、同じ三角波振幅を持つDACチャネル1の三角波カウンタがDHR 1レジスタに加算され、合計がDAC_DOR1に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル1の三角波カウンタが更新されます。

同時に、同じ三角波振幅を持つDACチャネル2の三角波カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル2の三角波カウンタが更新されます。

異なる三角波生成による同時トリガ

DACをこの変換モードに設定するには、次の手順が必要です。

1. TEN1とTEN2の2つのDACチャネルトリガイネーブルビットをセットします。
2. TSEL1およびTSEL2bitフィールドに同じ値をセットすることによって、両方のDACチャネルに同じトリガソースを設定します。
3. 2つのDACチャネルWAVEx[1:0] ビットを1xに、異なる最大振幅値をMAMP1[3:0] とMAMP2[3:0] ビットに設定します。
4. 目的のDHRレジスタ（DAC_DHR12RD、DAC_DHR12LD、またはDAC_DHR8RD）にデュアルDACチャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つDACチャネル1の三角波カウンタがDHR1レジスタに加算され、合計がDAC_DOR1に転送されます（3 AHB クロックサイクル後）。その後、DACチャネル1の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つDACチャネル2の三角波カウンタがDHR2レジスタに加算され、合計がDAC_DOR2に転送されます（3 dac_hclk クロックサイクル後）。その後、DACチャネル2の三角波カウンタが更新されます。

35.4.15 DAC自律モード

自律モードを使用すると、STOPモードでDAC出力電圧を更新することができます。これにより、デバイスがRUN、SLEEP、またはSTOPモードで動作しているときにDMA転送を実行できます。自律モードでAUTOMODEビットがサポートされるのは、DACがサンプルおよびホールドモードの場合のみです。このモードを有効にするには、DAC_AUTOCRレジスタ内のこのビットをセットします。DMAリクエストも有効化する必要があります（DMAEN = 1）。

AUTOMODEビットをセットすると、各ハードウェアトリガ信号によりRCCへのAHBクロックリクエストが生成されます。ペリフェラルがAHBクロックを受信すると、DHRxレジスタの内容がDAC_DORレジスタにロードされ、DMAリクエストが生成されます。DMAトランザクションが完了すると、DACではAHBクロックリクエストをデアサートし、新しいトリガイベントを待ちます。

サンプルおよびホールドモードを選択した場合、RCCによってdac_ker_ck低消費電力クロックを有効化する必要があります。

自律モードでのDMA転送は、次のシーケンスに従って行われます。

1. DACがGPIOまたはオンチップペリフェラルで電圧を出力します（dac_ker_ckとdac_hclkは不要です）。
2. DACが別のペリフェラルからハードウェアトリガを受信します。
3. DACがクロックリクエストをアサートします。
4. DACがDHRxレジスタの内容をDAC_DORレジスタにロードします。
5. DACがDMAリクエストを生成します。
6. DMA新しいデータをメモリからDHRxレジスタに転送します。
7. DACがクロックリクエストをデアサートします。

STOPモードで自律的に動作するようDACを初期化するには、次のシーケンスに従います。

1. RCCでDACクロックの自律モードを有効化します。
2. RCCでDACカーネル·クロックを選択します。
3. RCCでDAC用にdac_hold_ckクロックを有効化します。
4. MODExビットを使用してサンプルおよびホールドモードを選択します。
5. TSAMPLEx[9:0]、THOLDx[9:0]、TREFRESHx[7:0] ビットをセットして、サンプルおよびホールドモードを設定します。
6. ターゲット値をDHRxレジスタにロードします。
7. DAC値を出力するためにTENxビットをセットします。
8. TSELx[3:0] ビットを使用してトリガを設定します。
9. DMAENxビットをセットして、DMAインタフェースを設定します。
10. AUTOMODEビットをセットします。

DHRxからDAC_DORへの転送のタイミングの不確実性

各トリガイベントの後に、DHRxレジスタに格納されたデータがDAC_DORレジスタに転送されます。この転送はAHBクロックに基づいて行われるため、転送のタイミングはクロックの使用状況によって異なります。

AHBクロックが停止した場合、再開までに最小時間が必要です。この時間は、クロックソースによって異なります。

35.5 低消費電力モードのDAC

表343.低消費電力モードがDACに与える影響

モード	説明
SLEEP	影響はありません。DACがDMAと使用されます。
STOP(1)	DACは機能が保持され、サンプルおよびホールドモードでのDMA転送を実行できます。
STANDBY	DACペリフェラルはパワーダウンされ、STANBYまたはSHUTDOWNモード終了後に再初期化する必要があります。
SHUTDOWN	

1. DACペリフェラルでサポートされているSTOPモードについての詳細はセクション 35.3：DACの実装を参照してください。

35.6 DAC割込み

表344.DAC割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
DAC	DMAアンダーラン	DMAUDRX	DMAUDRIEX	DMAUDRx = 1を書き込む	可能	不可	不可

35.7 DACレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスする必要があります。

35.7.1 DAC制御レジスタ（DAC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2 [3:0]				WAVE2 [1:0]		TSEL2[3]	TSEL2[2]	TSEL2[1]	TSEL2[0]	TEN2	EN2
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1 [3:0]				WAVE1 [1:0]		TSEL1[3]	TSEL1[2]	TSEL1[1]	TSEL1[0]	TEN1	EN1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **CEN2**：DACチャネル2較正イネーブル

このビットは、DACチャネル2の較正を有効化/無効化するためにソフトウェアによってセット/クリアされます。DAC_CRでEN2bitが0にセットされている場合のみ書込み可能で（DACチャネルが無効化された場合のみ較正モードに移行／終了できます）、それ以外の書込み操作は無視されます。

0：DACチャネル2は通常動作モードです。

1：DACチャネル2は較正モードです。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット29 **DMAUDRIE2**：DACチャネル2 DMAアンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DACチャネル2 DMAアンダーラン割込みは無効です。

1：DACチャネル2 DMAアンダーラン割込みは有効です。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット28 **DMAEN2**：DACチャネル2 DMAイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DACチャネル2 DMAモードは無効です。

1：DACチャネル2 DMAモードは有効です。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット27:24 **MAMP2[3:0]**：DACチャネル2マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット0は1に等しい。
0001：LFSR／三角波振幅のアンマスクビット [1:0] は3に等しい。
0010：LFSR／三角波振幅のアンマスクビット [2:0] は7に等しい。
0011：LFSR／三角波振幅のアンマスクビット [3:0] は15に等しい。
0100：LFSR／三角波振幅のアンマスクビット [4:0] は31に等しい。
0101：LFSR／三角波振幅のアンマスクビット [5:0] は63に等しい。
0110：LFSR／三角波振幅のアンマスクビット [6:0] は127に等しい。
0111：LFSR／三角波振幅のアンマスクビット [7:0] は255に等しい。
1000：LFSR／三角波振幅のアンマスクビット [8:0] は511に等しい。
1001：LFSR／三角波振幅のアンマスクビット [9:0] は1023に等しい。
1010：LFSR／三角波振幅のアンマスクビット [10:0] は2047に等しい。
≥ 1011: LFSR／三角波振幅のアンマスクビット [11:0] は4095に等しい。

注： これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット23:22 **WAVE2[1:0]**：DACチャネル2ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／リセットされます。

00：波形生成は無効です。
01：ノイズ波生成は有効です。
1x：三角波生成は有効です。

注： ビットTEN2 = 1（DACチャネル2トリガ有効）の場合のみ使用されます。
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット21:18 **TSEL2[3:0]**：DACチャネル2トリガ選択

これらのビットは、DACチャネル2をトリガするために使用される外部イベントを選択します。

0000：SWTRIG2
0001：dac_ch2_trg1
0010：dac_ch2_trg2
.....
1111：dac_ch2_trg15

トリガの設定とマッピングの詳細については、セクション 35.4.2：DACピンおよび内部信号のトリガ選択表を参照してください。

注： ビットTEN2 = 1（DACチャネル2トリガ有効）の場合のみ使用されます。
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット 17 **TEN2**：DACチャネル2トリガイネーブル

このビットは、DACチャネル2トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DACチャネル2トリガは無効であり、DAC_DHR2レジスタに書き込まれたデータは、1 dac_hclk クロックサイクル後にDAC_DOR2レジスタに転送されます。
1：DACチャネル2トリガは有効であり、DAC_DHR2レジスタからのデータは、3 dac_hclk クロックサイクル後にDAC_DOR2レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHR2からDAC_DOR2レジスタへの転送は、わずか1 dac_hclk クロックサイクルで行われます。
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット16 **EN2**：DACチャネル2イネーブル

このビットは、DACチャネル2を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DACチャネル2は無効です。

1：DACチャネル2は有効です。

注： これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **CEN1**：DACチャネル1較正イネーブル

このビットは、DACチャネル1の較正を有効化／無効化するためにソフトウェアによってセット/クリアされます。DAC_CRでビットEN1 = 0の場合のみDAC_CRに書込み可能で（DACチャネルが無効化された場合のみ較正モードに移行／終了できます）、それ以外の書込み操作は無視されます。

0：DACチャネル1は通常動作モードです。

1：DACチャネル1は較正モードです。

ビット13 **DMAUDRIE1**：DACチャネル1 DMAアンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DACチャネル1 DMAアンダーラン割込みは無効です。

1：DACチャネル1 DMAアンダーラン割込みは有効です。

ビット12 **DMAEN1**：DACチャネル1 DMAイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DACチャネル1 DMAモードは無効です。

1：DACチャネル1 DMAモードは有効です。

ビット11:8 **MAMP1[3:0]**：DACチャネル1マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット0は1に等しい。

0001：LFSR／三角波振幅のアンマスクビット [1:0] は3に等しい。

0010：LFSR／三角波振幅のアンマスクビット [2:0] は7に等しい。

0011：LFSR／三角波振幅のアンマスクビット [3:0] は15に等しい。

0100：LFSR／三角波振幅のアンマスクビット [4:0] は31に等しい。

0101：LFSR／三角波振幅のアンマスクビット [5:0] は63に等しい。

0110：LFSR／三角波振幅のアンマスクビット [6:0] は127に等しい。

0111：LFSR／三角波振幅のアンマスクビット [7:0] は255に等しい。

1000：LFSR／三角波振幅のアンマスクビット [8:0] は511に等しい。

1001：LFSR／三角波振幅のアンマスクビット [9:0] は1023に等しい。

1010：LFSR／三角波振幅のアンマスクビット [10:0] は2047に等しい。

≥ 1011: LFSR／三角波振幅のアンマスクビット [11:0] は4095に等しい。

ビット7:6 **WAVE1[1:0]**：DACチャネル1ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：波形生成は無効です。

01：ノイズ波生成は有効です。

1x：三角波生成は有効です。

ビットTEN1 = 1（DACチャネル1トリガ有効）の場合のみ使用されます。

ビット5:2 **TSEL1[3:0]**：DACチャネル1トリガ選択

これらのビットは、DACチャネル1をトリガするために使用される外部イベントを選択します。

0000：SWTRIG1

0001：dac_ch1_trg1

0010：dac_ch1_trg2

.....

1111：dac_ch1_trg15

トリガの設定とマッピングの詳細については、セクション 35.4.2：DACピンおよび内部信号のトリガ選択表を参照してください。

注： ビットTEN1 = 1（DACチャネル1トリガ有効）の場合のみ使用されます。

ビット 1 **TEN1**：DACチャネル1トリガイネーブル

このビットは、DACチャネル1トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DACチャネル1トリガは無効であり、DAC_DHR1レジスタに書き込まれたデータは、1 dac_hclk クロックサイクル後にDAC_DOR1レジスタに転送されます。

1：DACチャネル1トリガは有効であり、DAC_DHR1レジスタからのデータは、3 dac_hclk クロックサイクル後にDAC_DOR1レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHR1 レジスタからDAC_DOR1レジスタへの転送は、わずか1 dac_hclk クロックサイクルで行われます。

ビット0 **EN1**：DACチャネル1イネーブル

このビットは、DACチャネル1を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DACチャネル1は無効です。

1：DACチャネル1は有効です。

35.7.2 DACソフトウェアトリガレジスタ（DAC_SWTRGR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
														w	w

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **SWTRIG2**：DACチャネル2ソフトウェアトリガ

このビットは、ソフトウェアトリガモードでDACをトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR2レジスタの値がDAC_DOR2レジスタにロードされると、ハードウェアによってクリアされます（1 dac_hclk クロックサイクル後）。

このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

ビット0 **SWTRIG1**：DACチャネル1ソフトウェアトリガ

このビットは、ソフトウェアトリガモードでDACをトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR1レジスタの値がDAC_DOR1レジスタにロードされると、ハードウェアによってクリアされます（1 dac_hclk クロックサイクル後）。

35.7.3 DACチャネル1の12bit右詰めデータ保持レジスタ（DAC_DHR12R1）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC1DHRB [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **DACC1DHRB[11:0]**：DACチャネル1の12bit右詰めデータB

これらのビットは、ソフトウェアによって書き込まれます。DACがダブルデータモードで動作しているとき、これらにはDACチャネル1の12bitのデータを指定します。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DACC1DHR[11:0]**：DACチャネル1の12bit右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DACチャネル1の12bitデータを指定します。

35.7.4 DACチャネル1の12bit左詰めデータ保持レジスタ（DAC_DHR12L1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC1DHRB [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:20 **DACC1DHRB[11:0]**：DACチャネル1の12bit左詰めデータB

これらのビットは、ソフトウェアによって書き込まれます。DACがダブルデータモードで動作しているとき、これらにはDACチャネル1の12bitのデータを指定します。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット15:4 **DACC1DHR[11:0]**：DACチャネル1の12bit左詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。

これらには、DACチャネル1の12bitデータを指定します。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

35.7.5 DACチャネル1の8bit右詰めデータ保持レジスタ（DAC_DHR8R1）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHRB [7:0]								DACC1DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **DACC1DHRB[7:0]**：DACチャネル1の8bit右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DACがダブルデータモードで動作しているとき、これらにはDACチャネル1の8bitのデータを指定します。

ビット7:0 **DACC1DHR[7:0]**：DACチャネル1の8bit右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DACチャネル1の8bitデータを指定します。

35.7.6 DACチャネル2の12bit右詰めデータ保持レジスタ（DAC_DHR12R2）

このレジスタは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHRB [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **DACC2DHRB[11:0]**：DACチャネル2の12ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DACがDMAダブルデータモードで動作しているとき、これらにはDACチャネル2の12bitのデータを指定します。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DACC2DHR[11:0]**：DACチャネル2の12ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DACチャネル2の12bitデータを指定します。

35.7.7 DACチャネル2の12bit左詰めデータ保持レジスタ（DAC_DHR12L2）

このレジスタは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHRB [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:20 **DACC2DHRB[11:0]**：DACチャネル2の12bit左詰めデータB

これらのビットは、ソフトウェアによって書き込まれます。DACがダブルデータモードで動作しているとき、これらにはDACチャネル2の12bitのデータを指定します。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15:4 **DACC2DHR[11:0]**：DACチャネル2の12ビット左詰めデータ

これらのビットは、DACチャネル2の12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

35.7.8 DACチャネル2の8bit右詰めデータ保持レジスタ（DAC_DHR8R2）

このレジスタは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHRB [7:0]								DACC2DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **DACC2DHRB[7:0]**：DACチャネル2の8ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DACがダブルデータモードで動作しているとき、これらにはDACチャネル2の8bitのデータを指定します。

ビット7:0 **DACC2DHR[7:0]**：DACチャネル2の8ビット右詰めデータ

これらのビットは、DACチャネル2の8ビットデータを指定するために、ソフトウェアによって書き込まれます。

35.7.9 デュアルDAC 12bit右詰めデータ保持レジスタ（DAC_DHR12RD）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **DACC2DHR[11:0]**：DACチャネル2の12ビット右詰めデータ

これらのビットは、DACチャネル2の12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DACC1DHR[11:0]**：DACチャネル1の12bit右詰めデータ

これらのビットは、DACチャネル1の12bitデータを指定するために、ソフトウェアによって書き込まれます。

35.7.10 デュアルDAC 12bit左詰めデータ保持レジスタ（DAC_DHR12LD）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット31:20 **DACC2DHR[11:0]**：DACチャネル2の12ビット左詰めデータ

これらのビットは、DACチャネル2の12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15:4 **DACC1DHR[11:0]**：DACチャネル1の12bit左詰めデータ

これらのビットは、DACチャネル1の12bitデータを指定するために、ソフトウェアによって書き込まれます。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

35.7.11 デュアルDAC 8bit右詰めデータ保持レジスタ（DAC_DHR8RD）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [7:0]								DACC1DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **DACC2DHR[7:0]**：DACチャネル2の8ビット右詰めデータ

これらのビットは、DACチャネル2の8ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット7:0 **DACC1DHR[7:0]**：DACチャネル1の8bit右詰めデータ

これらのビットは、DACチャネル1の8bitデータを指定するために、ソフトウェアによって書き込まれます。

35.7.12 DACチャネル1データ出力レジスタ（DAC_DOR1）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC1DORB [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **DACC1DORB[11:0]**：DACチャネル1データ出力

これらのビットは読出し専用です。これらには、DACチャネル1 Bのデータ出力が含まれます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DACC1DOR[11:0]**：DACチャネル1データ出力

これらのビットは読出し専用であり、DACチャネル1のデータ出力を含みます。

35.7.13 DACチャネル2データ出力レジスタ（DAC_DOR2）

このレジスタは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DORB [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **DACC2DORB[11:0]**：DACチャネル2データ出力

これらのビットは読出し専用です。これらには、DACチャネル2 Bのデータ出力が含まれます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DACC2DOR[11:0]**：DACチャネル2データ出力

これらのビットは読出し専用であり、DACチャネル2のデータ出力を含みます。

35.7.14 DACステータスレジスタ（DAC_SR）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWST2	CAL_FLAG2	DMAUDR2	DORSTAT2	DAC2RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1	r	r											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BWST1	CAL_FLAG1	DMAUDR1	DORSTAT1	DAC1RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1	r	r											

ビット31 **BWST2**：DACチャネル2ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによってセットされます。このビットは、ソフトウェアがDAC_SHSR2レジスタに書き込むたびにセットされ、DAC_SHSR2の書込み操作が完了するとハードウェアによってクリアされます（同期に約3 LSI/LSE 周期かかります）。

0：進行中のDAC_SHSR2の書込み操作はありません。DAC_SHSR2は書込み可能です。

1：進行中のDAC_SHSR2の書込み操作があります。DAC_SHSR2は書込みできません。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装 を参照してください。

ビット30 **CAL_FLAG2**：DACチャネル2較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

ビット29 **DMAUDR2**：DACチャネル2 DMAアンダーランフラグ

このビットは、ハードウェアによってセットされ、（1を書き込むことによって）ソフトウェアによってクリアされます。

0：DACチャネル2にDMAアンダーランエラー条件は発生しませんでした。

1：DACチャネル2にDMAアンダーランエラー条件が発生しました（現在選択されているトリガは、DMAサービス機能のレートを上回る周波数でDACチャネル2変換を駆動しています）。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット28 **DORSTAT2**：DACチャネル2出力レジスタステータスビット

このビットは、ハードウェアによってセット／クリアされます。これは、DACがダブルデータモードで動作している場合のみ適用されます。

0：DOR[11:0] が実際のDAC出力に使用されます。

1：DORB[11:0] が実際のDAC出力に使用されます。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット27 **DAC2RDY**：DACチャネル2レディステータスビット

このビットは、ハードウェアによってセット／クリアされます。

0：DACチャネル2は、まだトリガも出力データも受け入れる準備ができていません。

1：DACチャネル2は、トリガまたは出力データを受け入れる準備ができています。

注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット26:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **BWST1**：DACチャネル1ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによって、およびソフトウェアがDAC_SHSR1レジスタに書き込むたびにセットされます。DAC_SHSR1の書込み操作が完了するとハードウェアによってクリアされます（同期に約3 LSI/LSE 周期かかります）。

0：進行中のDAC_SHSR1の書込み操作はありません。DAC_SHSR1は書込み可能です。

1：進行中のDAC_SHSR1の書込み操作があります。DAC_SHSR1は書込みできません。

ビット14 **CAL_FLAG1**：DACチャネル1較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット13 **DMAUDR1**：DACチャネル1のDMAアンダーランフラグ

このビットは、ハードウェアによってセットされ、（1を書き込むことによって）ソフトウェアによってクリアされます。

0：DACチャネル1にDMAアンダーランエラー条件は発生しませんでした。

1：DACチャネル1にDMAアンダーランエラー条件が発生しました（現在選択されているトリガは、DMAサービス機能のレートを上回る周波数でDACチャネル1変換を駆動しています）。

ビット12 **DORSTAT1**：DACチャネル1出力レジスタステータスビット

このビットは、ハードウェアによってセット／クリアされます。これは、DACがダブルデータモードで動作している場合のみ適用されます。

0：DOR[11:0] が実際のDAC出力に使用されます。

1：DORB[11:0] が実際のDAC出力に使用されます。

ビット11 **DAC1RDY**：DACチャネル1レディステータスビット

このビットは、ハードウェアによってセット／クリアされます。

0：DACチャネル1は、まだトリガも出力データも受け入れる準備ができていません。

1：DACチャネル1は、トリガまたは出力データを受け入れる準備ができています。

ビット10:0 予約済みであり、リセット値に保持する必要があります。

35.7.15 DAC較正制御レジスタ（DAC_CCR）

アドレスオフセット：0x38

リセット値：0x00XX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2 [4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1 [4:0]				
											rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **OTRIM2[4:0]**：DACチャネル2オフセットトリミング値
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

ビット15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **OTRIM1[4:0]**：DACチャネル1オフセットトリミング値

35.7.16 DACモード制御レジスタ（DAC_MCR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SINFORMAT2	DMADOUBLE2	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]		
						rw	rw						rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HFSEL[1]	HFSEL[0]	Res.	Res.	Res.	Res.	SINFORMAT1	DMADOUBLE1	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
rw	rw					rw	rw						rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **SINFORMAT2**：DACチャネル2の符号付きフォーマットの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：入力データは符号なしフォーマットです。
1：入力データは符号付きフォーマット（2の補数）です。MSBビットは、符号を表します。
注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

ビット24 **DMADOUBLE2**：DACチャネル2 DMAダブルデータモード
このビットは、ソフトウェアによってセット／クリアされます。
0：DMA通常モードが選択されています。
1：DMAダブルデータモードが選択されています。
注： このビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

ビット23:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **MODE2[2:0]**：DACチャネル2モード

これらのビットは、DACが無効になっており、較正モードでない場合（DAC_CRレジスタでビットEN2 = 0およびCEN2 = 0）のみ書込み可能です。EN2 = 1またはCEN2 = 1の場合、書込み操作は無視されます。
これらは、DACチャネル2モードを選択するために、ソフトウェアによってセット/クリアされます。

– DACチャネル2は通常動作モードです。

000：DACチャネル2はバッファ有効で外部ピンに接続されています。
001：DACチャネル2はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。
010：DACチャネル2はバッファ無効で外部ピンに接続されています。
011：DACチャネル2はバッファ無効でオンチップペリフェラルに接続されています。

– D/Aコンバータチャネル2はサンプルおよびホールドモードです。

100：DACチャネル2はバッファ有効で外部ピンに接続されています。
101：DACチャネル2はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。
110：DACチャネル2はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。
111：DACチャネル2はバッファ無効でオンチップペリフェラルに接続されています。

注： このレジスタは、EN2 = 0のときにのみ修正できます。
DACチャネル2の有無については、セクション 35.3：DACの実装を参照してください。

ビット15:14 **HFSEL[1:0]**：高周波数インタフェースモードの選択

00：高周波数インタフェースモードは無効です。
01：高周波数インタフェースモードは、AHBクロック周波数 > 80MHzに対して有効です。
10：高周波数インタフェースモードは、AHBクロック周波数 > 160MHzに対して有効です。
11：予約済みです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **SINFORMAT1**：DACチャネル1の符号付きフォーマットの有効化

このビットは、ソフトウェアによってセット／クリアされます。
0：入力データは符号なしフォーマットです。
1：入力データは符号付きフォーマット（2の補数）です。MSBビットは、符号を表します。

ビット8 **DMADOUBLE1**：DACチャネル1のDMAダブルデータモード

このビットは、ソフトウェアによってセット／クリアされます。
0：DMA通常モードが選択されています。
1：DMAダブルデータモードが選択されています。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **MODE1[2:0]**：DACチャネル1モード

これらのビットは、DACが無効になっており、較正モードでない場合（DAC_CRレジスタでビットEN1 = 0およびCEN1 = 0）のみ書込み可能です。EN1 = 1またはCEN1 = 1の場合、書込み操作は無視されます。
これらは、DACチャネル1モードを選択するために、ソフトウェアによってセット/クリアされます。

– DACチャネル1は通常動作モードです。

000：DACチャネル1はバッファ有効で外部ピンに接続されています。
001：DACチャネル1はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。
010：DACチャネル1はバッファ無効で外部ピンに接続されています。
011：DACチャネル1はバッファ無効でオンチップペリフェラルに接続されています。

– DACチャネル1はサンプルおよびホールドモードです。

100：DACチャネル1はバッファ有効で外部ピンに接続されています。
101：DACチャネル1はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。
110：DACチャネル1はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。
111：DACチャネル1はバッファ無効でオンチップペリフェラルに接続されています。

注： このレジスタは、EN1 = 0のときにのみ修正できます。

35.7.17 DACチャネル1サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR1）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **TSAMPLE1[9:0]**：D/Aコンバータチャネル1サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットはDACチャネル1が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SRレジスタのBWST1がローのときのみ書込み可能であり、BWST1 = 1の場合、書込み操作は無視されます。

注： **これは、サンプルフェーズを実行する LSI/LSE クロックの数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x LSI/LSE** クロック周期です。

35.7.18 DACチャネル2サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2）

このレジスタは、デュアルチャネルDACでのみ使用できます。セクション 35.3：DACの実装を参照してください。

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **TSAMPLE2[9:0]**：D/Aコンバータチャネル2サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットはDACチャネル2が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SRレジスタのBWST2がローのときのみ書込み可能であり、BWST2 = 1の場合、書込み操作は無視されます。

注： **これは、サンプルフェーズを実行する LSI/LSE クロックの数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x LSI/LSE** クロック周期です。

35.7.19 DACサンプルおよびホールド時間レジスタ（DAC_SHHR）

アドレスオフセット：0x48

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **THOLD2[9:0]**：D/Aコンバータチャネル2ホールド時間（サンプルおよびホールドモードでのみ有効）
ホールド時間 = (THOLD[9:0]) x LSI/LSE クロック周期

注： このレジスタは、EN2 = 0のときにのみ修正できます。
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **THOLD1[9:0]**：D/Aコンバータチャネル1ホールド時間（サンプルおよびホールドモードでのみ有効）
ホールド時間 = (THOLD[9:0]) x LSI/LSE クロック周期

注： このレジスタは、EN1 = 0のときにのみ修正できます。

注： これらのビットは、DACチャネルが無効になっており、通常動作モードの場合（DAC_CRレジスタのビットENx = 0およびビットCENx = 0）のみ書込み可能です。ENx = 1またはCENx = 1の場合、書込み操作は無視されます。

35.7.20 DACサンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）

アドレスオフセット：0x4C

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **TREFRESH2[7:0]**：D/Aコンバータチャネル2リフレッシュ時間（サンプルおよびホールドモードでのみ有効）
リフレッシュ時間 = (TREFRESH[7:0]) x LSI/LSE クロック周期

注： このレジスタは、EN2 = 0のときにのみ修正できます。
これらのビットは、デュアルチャネルDACでのみ使用できます。セクション 35.3 : DACの実装を参照してください。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **TREFRESH1[7:0]**：D/Aコンバータチャネル1リフレッシュ時間（サンプルおよびホールドモードでのみ有効）
リフレッシュ時間 = (TREFRESH[7:0]) x LSI/LSE クロック周期

注： このレジスタは、EN1 = 0のときにのみ修正できます。

注： これらのビットは、DACチャネルが無効になっており、通常動作モードの場合（DAC_CRレジスタのビットENx = 0およびビットCENx = 0）のみ書込み可能です。ENx = 1またはCENx = 1の場合、書込み操作は無視されます。

35.7.21 DAC自律モード制御レジスタ（DAC_AUTOCR）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AUTOMODE	Res.	Res.	Res.	Res.	Res.	Res.
									rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **AUTOMODE**：DAC自律モード
このビットは、ソフトウェアによってセット／クリアされます。
0：DAC自律モードは無効です。
1：DAC自律モードは有効です。

ビット21:0 予約済みであり、リセット値に保持する必要があります。

35.7.22 DACレジスタマップ

表345 にDACレジスタの要約を示します。

表345.DACレジスタのマップとリセット値

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DAC_CR	Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2 [3:0]				WAVE2 [2:0]		TSEL2 [3:1]			TSEL2[0]	TEN2	EN2	Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1 [3:0]				WAVE1 [1:0]		TSEL1 [3:1]			TSEL1[0]	TEN1	EN1
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	DAC_SWTRGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
	リセット値																															0	0
0x08	DAC_DHR12R1	Res.	Res.	Res.	Res.	DACC1DHRB [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DAC_DHR12L1	DACC1DHRB [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x10	DAC_DHR8R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHRB [7:0]				DACC1DHR [7:0]											
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	DAC_DHR12R2	Res.	Res.	Res.	Res.	DACC2DHRB [11:0]												Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x18	DAC_DHR12L2	DACC2DHRB [11:0]												Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	DAC_DHR8R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHRB [7:0]				DACC2DHR [7:0]											
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DAC_DHR12RD	Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x24	DAC_DHR12LD	DACC2DHR [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x28	DAC_DHR8RD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [7:0]				DACC1DHR [7:0]											
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	DAC_DOR1	Res.	Res.	Res.	Res.	DACC1DORB [11:0]												Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x30	DAC_DOR2	Res.	Res.	Res.	Res.	DACC2DORB [11:0]												Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x34	DAC_SR	BWST2	CAL_FLAG2	DMAUDR2	DORSTAT2	DAC2RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BWST1	CAL_FLAG1	DMAUDR1	DORSTAT1	DAC1RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0												0	0	0	0	0											

表345.DACレジスタのマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	DAC_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2[4]	OTRIM2[3]	OTRIM2[2]	OTRIM2[1]	OTRIM2[0]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1[4]	OTRIM1[3]	OTRIM1[2]	OTRIM1[1]	OTRIM1[0]
	リセット値												X	X	X	X	X												X	X	X	X	X
0x3C	DAC_MCR	Res.	Res.	Res.	Res.	Res.	Res.	SINFORMAT2	DMADOUBLE2	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]			HFSEL[1]	HFSEL[0]	Res.	Res.	Res.	Res.	SINFORMAT1	DMADOUBLE1	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
	リセット値							0	0						0	0	0	0	0					0	0						0	0	0
0x40	DAC_SHSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x44	DAC_SHSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x48	DAC_SHHR	Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]										Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	1							0	0	0	0	0	0	0	0	0	1
0x4C	DAC_SHRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
	リセット値									0	0	0	0	0	0	0	1									0	0	0	0	0	0	0	1
0x50	予約済みです。	Res.																															
0x54	DAC_AUTOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AUTOMODE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値										0																						

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

36 電圧基準バッファ（VREFBUF）

36.1 概要

デバイスには電圧基準バッファが搭載されており、ADC や DAC の電圧基準や、VREF+ ピンを介した外部コンポーネントの電圧基準としても使用できます。

36.2 VREFBUFの実装

以下の表にVREFBUF電圧の標準値を示します。

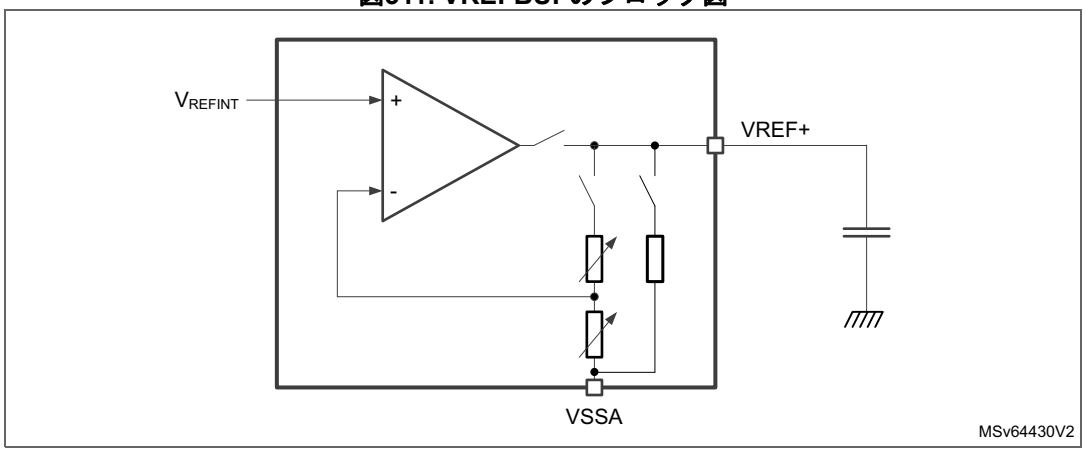
表346.VREFBUFの標準値

記号	値
VREFBUF0	1.5 V
VREFBUF1	1.8 V
VREFBUF2	2.048 V
VREFBUF3	2.5 V

注： **詳細については、製品データシートを参照してください。**

36.3 VREFBUFの機能説明

図311. VREFBUFのブロック図



内部基準電圧バッファは、ゲインをプログラムできるオペアンプです。増幅器入力は内部電圧基準 V_{REFINT} に接続されています。VREFBUFは次の4つの電圧をサポートしており[(a)]、VREFBUF_CSRレジスタのVRSビットで設定します。

- VRS = 000：VREFBUF0電圧が選択されます。
- VRS = 001：VREFBUF1電圧が選択されます。
- VRS = 010：VREFBUF2電圧が選択されます。
- VRS = 011：VREFBUF3電圧が選択されます。

内部電圧基準はENVRビットおよびHIZビットの設定に応じて4つの個別モードに設定できます。これらのモードを以下の表に示します。

表347.VREFバッファモード

ENVR	HIZ	VREFバッファ設定
0	0	VREFBUFバッファオフモード： – V_{REF+} ピンがV_{SSA} にプルダウン
0	1	外部基準電圧モード（デフォルト値）： – VREFBUFバッファオフ – V_{REF+} ピンは入力モード
1	0	内部基準電圧モード – VREFBUFバッファオン – V_{REF+} ピンはVREFBUFバッファ出力に接続
1	1	ホールドモード： – VREFは出力バッファなしで有効、VREF+ ピン電圧は外部コンデンサで保持 – VRRの検出は無効で、VRRビットは最後の状態を保持

VREFBUF_CSRレジスタのENVRビットをセットし､HIZビットをクリアすることでVREFBUFを有効化した後、ユーザは電圧基準出力が期待値に達したことを示すVRRビットがセットされるまで待つ必要があります。

36.4 VREFBUFのトリミング

VREFBUF出力電圧は、工場出荷時にSTによって構成されます。リセット時、またVRS設定を変更するたびに、較正データが自動的にTRIMレジスタにロードされます。

また、オプションでTRIMレジスタのビットを直接変更することによって、出力電圧を調整することができます。この場合、デバイスをリセットするまでは、VRSの設定がTRIMレジスタに影響することはありません。

a. V_{DDA} の最小電圧はVRSの設定に依存します。製品データシートを参照してください。

36.5 VREFBUFレジスタ

36.5.1 VREFBUF制御およびステータスレジスタ（VREFBUF_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRS[2:0]			VRR	Res.	HIZ	ENVR
									rw	rw	rw	r		rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **VRS[2:0]**：電圧基準スケール

これらのビットは、基準電圧バッファによって生成される値を選択します。

VRS = 000：VREFBUF0電圧が選択されます。

VRS = 001：VREFBUF1電圧が選択されます。

VRS = 010：VREFBUF2電圧が選択されます。

VRS = 011：VREFBUF3電圧が選択されます。

その他：予約済み

注： 各VREFBUFx電圧の設定値については、製品データシートを参照してください。ソフトウェアは、VREFBUFが無効な（ENVR=0）ときだけこのビットフィールドをプログラムできます。

ビット 3 VRR：電圧基準バッファのレディ状態

0：基準電圧バッファ出力がレディ状態ではありません。

1：基準電圧バッファ出力がリクエストされたレベルに達しました。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **HIZ**：ハイインピーダンスモード

このビットによって、V_{REF+} ピンを接続／切断するためのアナログスイッチを制御します。

0：V_{REF+} ピンが電圧基準バッファ出力に内部接続されます。

1：VREF+ ピンをハイインピーダンスにします。

ENVRビットの設定に応じたモード説明については、表 347：VREFバッファモードを参照してください。

ビット0 **ENVR**：電圧基準バッファモードの有効化

このビットは、基準電圧バッファモードの有効化に使用されます。

0：内部電圧基準モードは無効です（外部電圧基準モード）。

1：内部電圧基準モード（基準バッファ有効化またはホールドモード）は有効です。

36.5.2 VREFBUF較正制御レジスタ（VREFBUF_CCR）

アドレスオフセット：0x04

リセット値：0x0000：00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
										rw	rw	rw	rw	rw	rw

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5:0 TRIM[5:0]：トリミングコード

TRIMコードは、次に示す機構に従ってセットおよび更新された6ビットの符号なしデータ（最小000000、最大111111）です。

リセット：

TRIM[5:0] は、生産試験中にFlashメモリに格納されたトリミング値VRS = 0で自動的に初期化されます。

VRSの変化：

TRIM[5:0] は、生産試験中にFlashメモリに格納された（VRS設定に対応する）トリミング値で自動的に初期化されます。

TRIM[5:0] への書込み：

ユーザはTRIM[5:0] を任意の値で変更できます。これにより、VRSによるトリミング値の制御は恒久的に無効化されます（デバイスがリセットされるまで）。

注： ユーザアプリケーションがトリミングを行う場合、トリミングコードは、000000から111111まで昇順で開始する必要があります。

36.5.3 VREFBUFレジスタマップ

次の表に、VREFBUFレジスタマップとリセット値を示します。

表348.VREFBUFレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	VREFBUF_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		VRS[2:0]		VRR	Res.	HIZ	ENVR
	リセット値																										0	0	0	0		1	0
0x04	VREFBUF_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
	リセット値																											x	x	x	x	x	x

レジスタ境界アドレスについては、セクション2.3を参照してください。

37 コンパレータ（COMP）

37.1 概要

このデバイスには、2個の超低消費電力コンパレータCOMP1およびCOMP2が内蔵されています。

これらのコンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低消費電力モードからのウェイクアップ
- アナログ信号調節
- タイマからのPWM出力を組み合わせた場合のサイクルごとの電流制御ループ

37.2 COMPの主な機能

- 各コンパレータは、電圧を柔軟に選択できるように、次のような設定可能な正入力と負入力を備えています。
  - マルチプレクスI/Oピン
  - DACチャネル1およびチャネル2
  - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および3つの分圧比（1/4、1/2、3/4）
- プログラム可能なヒステリシス
- プログラム可能なスピード／消費電力
- ブレークイベントをトリガして高速PWMシャットダウンするための、I/Oまたはタイマ入力へのリダイレクトが可能な出力
- ブランキングソースとともにコンパレータ出力
- ウィンドウコンパレータとして組み合わせることができるコンパレータ
- SLEEPおよびSTOPモードからのウェイクアップ（EXTIコントローラ経由）を備えた各コンパレータ用の割込み生成機能

37.3 COMPの実装

次の表に、STM32U5シリーズデバイスのCOMP機能を示します。

表349.COMP機能(1)

COMPのモード／機能	STM32U535/545	その他のデバイス
COMP1	X	X
COMP1 INPyの最大入力	y = 6	y = 3
COMP1_INPSEL[2:0] の予約済みコード	≥ 101	≥ 011
COMP2	-	X
ウィンドウコンパレータ機能	-	X

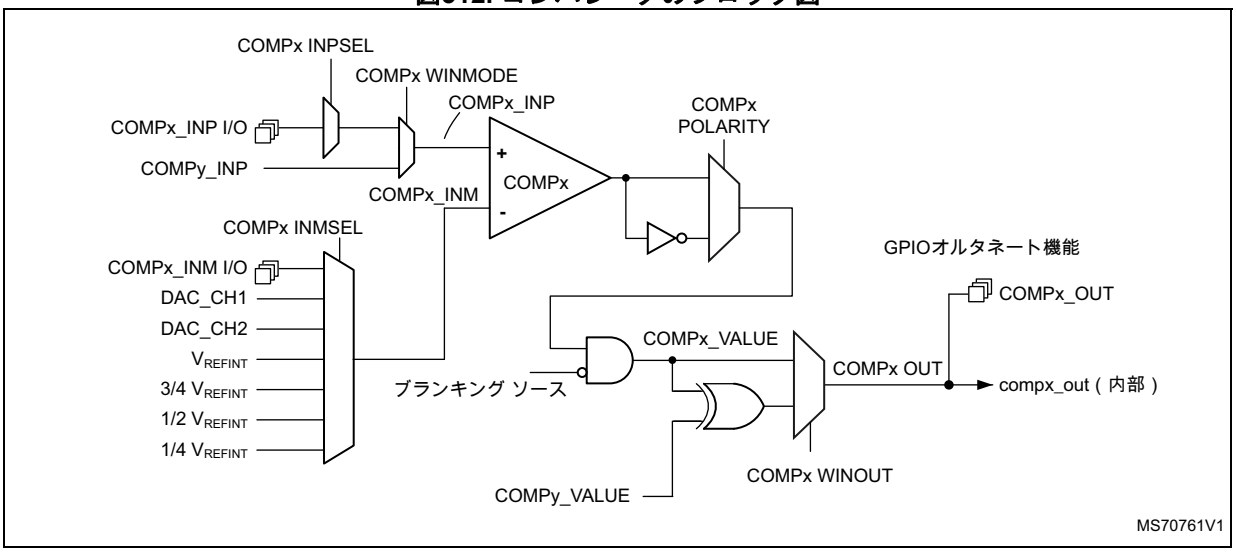
1. 「X」：サポート対象です
「-」：サポート対象外です

37.4 COMPの機能説明

37.4.1 COMPのブロック図

次の図に、コンパレータのブロック図を示します。

図312. コンパレータのブロック図



37.4.2 COMPピンおよび内部信号

コンパレータ入力として使用されるI/Oは、GPIOレジスタのアナログモードで設定する必要があります。

コンパレータの出力は、データシートの「オルタネート機能配置」表に示されているオルタネート機能チャネルを使用してI/Oに接続することができます。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- BKIN入力およびBKIN2入力を使用したPWM信号の緊急停止
- OCREF_CLR入力を使用したサイクル-バイ-サイクル電流制御
- タイミング測定のためのキャプチャ入力

コンパレータの出力は、内部と外部に同時にリダイレクトできます。

表350.COMP1非反転入力割当て

COMP1_INP	COMP1_INPSEL [2:0]
COMP1_INP1	000
COMP1_INP2	001
COMP1_INP3	010
COMP1_INP4	011
COMP1_INP5	100
COMP1_INP6	101

表351.COMP1反転入力割当て

COMP1_INM	COMP1_INMSEL [3:0]
¼ V_{REFINT}	0000
½ V_{REFINT}	0001
¾ V_{REFINT}	0010
V_{REFINT}	0011
DACチャネル1	0100
DACチャネル2	0101
COMP1_INM1	0110
COMP1_INM2	0111
予約済みです。	≥ 1000

表352.COMP2非反転入力割当て

COMP2_INP	COMP2_INPSEL [1:0]
COMP2_INP1	00
COMP2_INP2	01
予約済みです。	10
予約済みです。	11

表353.COMP2反転入力割当て

COMP2_INM	COMP2_INMSEL [3:0]
¼ V_{REFINT}	0000
½ V_{REFINT}	0001
¾ V_{REFINT}	0010
V_{REFINT}	0011
DACチャネル1	0100
DACチャネル2	0101
COMP2_INM1	0110
COMP2_INM2	0111
予約済みです。	≥ 1000

表354.COMP1出力ブランキングPWM割当て

PWM出力	COMP1_BLANKSEL [4:0]
なし（ブランキングなし）	00000
tim1_oc5	xxxx1
tim2_oc3	xxx1x
tim3_oc3	xx1xx
予約済み	その他

表355.COMP2出力ブランキングPWM割当て

PWM出力	COMP2_BLANKSEL [4:0]
なし（ブランキングなし）	00000
tim3_oc4	xxxx1
tim8_oc5	xxx1x
tim15_oc1	xx1xx
予約済み	その他

37.4.3 コンパレータのロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるアプリケーションの場合、万一誤ったレジスタへのアクセスやプログラムカウンタの破壊が起こった場合に、コンパレータのプログラミングが絶対に変更されないようにする必要があります。このような目的で、コンパレータの制御レジスタおよびステータスレジスタを書込み保護することができます（読出し専用）。

プログラミングが完了したら、COMPxLOCKビットを”1”にセットすることができます。これにより、COMPxLOCKビットを含むCOMPx_CSRレジスタ全体が読出し専用になります。

書込み保護はMCUリセットによってのみリセット可能です。

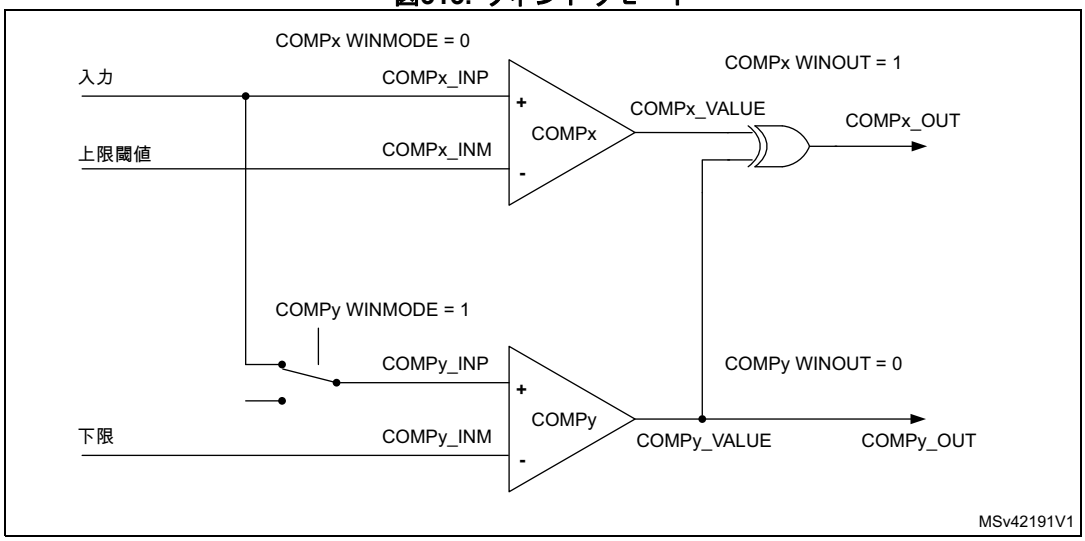
37.4.4 ウィンドウコンパレータ

ウィンドウコンパレータの目的は、アナログ電圧が下限および上限閾値によって定義された指定の電圧範囲内にあるかどうか監視することです。

2個の内蔵コンパレータは、ウィンドウコンパレータの作成に使用されます。監視されるアナログ電圧は、一緒に接続されるコンパレータの非反転（正）入力に接続されます。上限および下限閾値電圧は、コンパレータの反転（負）入力に接続されます。

2個の非反転入力は、WINMODEビットを有効化することで内部的に一緒に接続されるので、1つのIOを他の目的のために確保できます。

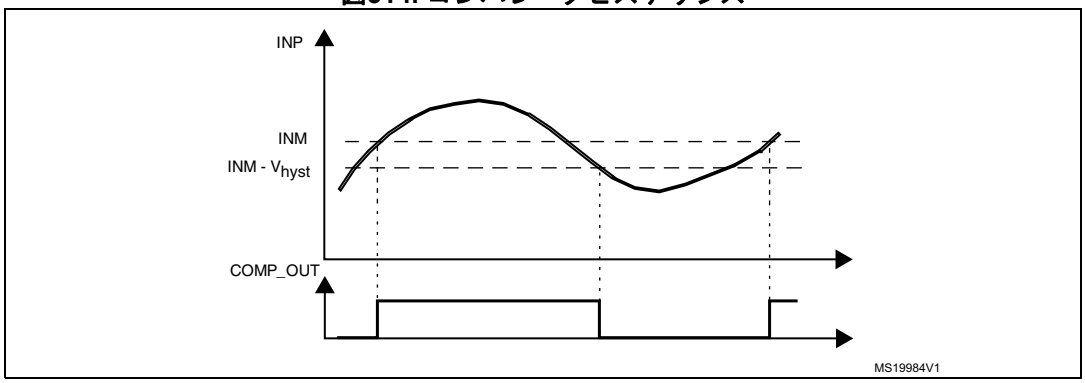
図313. ウィンドウモード



37.4.5 ヒステリシス

コンパレータには、ノイズの多い信号での疑似出力遷移を避けるために、プログラム可能なヒステリシスが含まれています。不要な場合は、ヒステリシスを無効にすることができます（低消費電力モードを終了する場合など）。これにより、外部コンポーネントを使用してヒステリシス値を強制できます。

図314. コンパレータヒステリシス

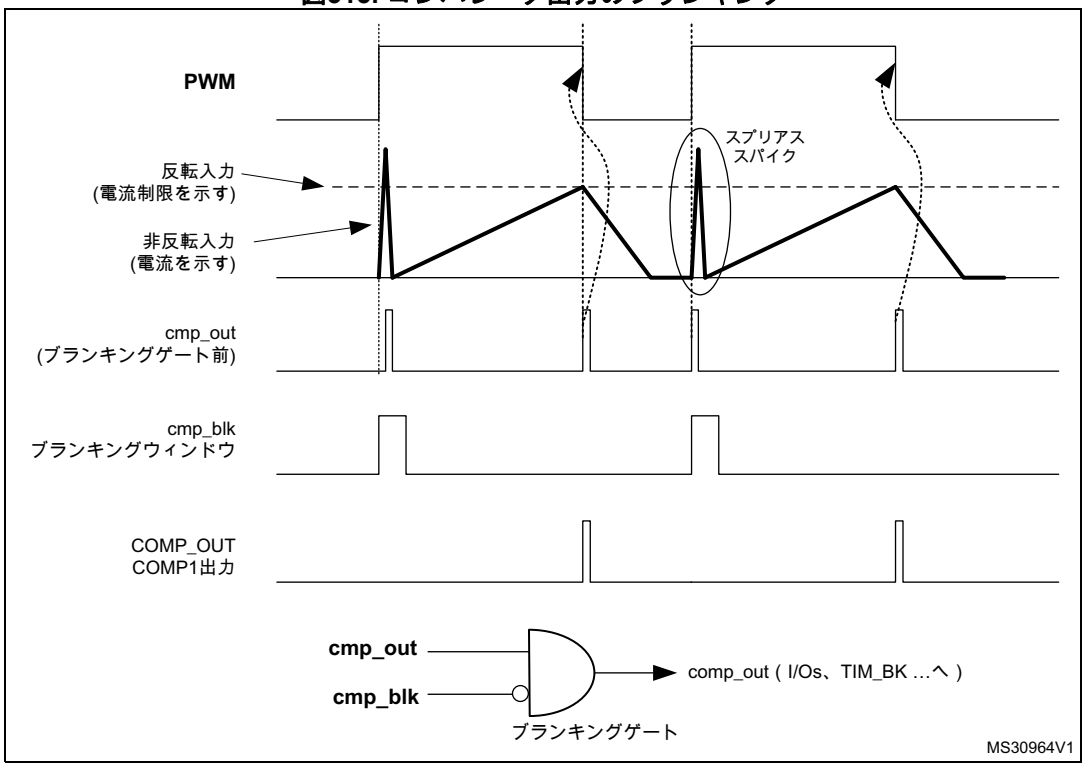


37.4.6 コンパレータの出力のブランキング機能

ブランキング機能は、PWM期間の開始時に短い電流スパイク (通常は電源スイッチのアンチパラレルダイオードのリカバリ電流) によって電流制御がトリップするのを防ぎます。このブランキング機能は、タイマ出力比較信号であるブランキングウィンドウの選択で成っています。選択はソフトウェアで行われます（考えられるブランキング信号については、コンパレータレジスタの説明を参照）。

ブランキング信号の補数は、望ましいコンパレータ出力を得るために、コンパレータ出力に対してAND演算されます（次の図の例を参照）。

図315. コンパレータ出力のブランキング



37.4.7 COMP電力とスピードモード

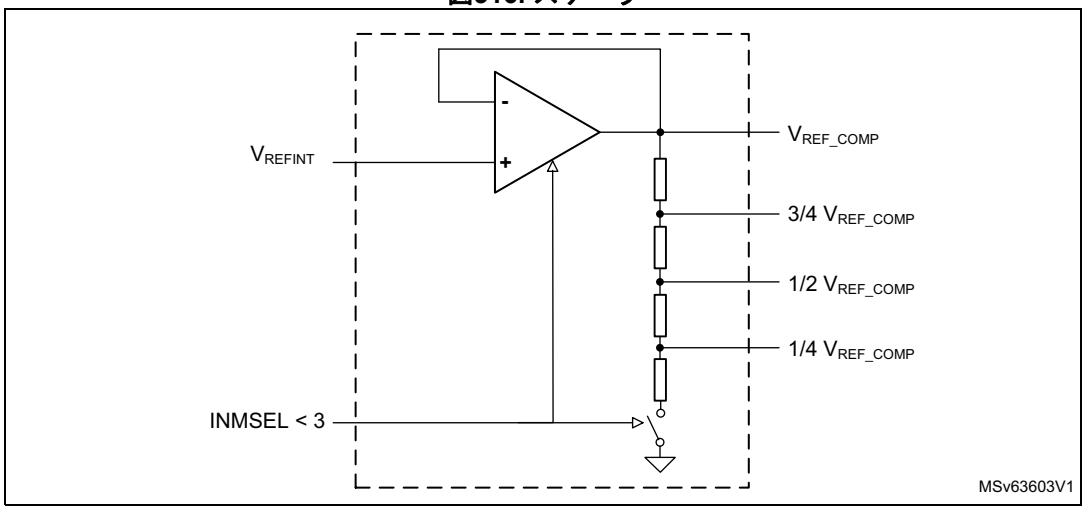
COMP1およびCOMP2の電力消費と伝搬遅延を調節して、特定のアプリケーションに最適なトレードオフを実現させることができます。

37.4.8 スケーラ機能

スケーラブロックは、異なる電圧基準レベルをコンパレータの入力に提供します。このブロックは抵抗器ブリッジを駆動する増幅器をベースにしています。増幅器入力は、内部電圧基準に接続されています。増幅器と抵抗器ブリッジは、対応する反転入力をスケーラ出力に接続するために、COMP_CFGRxレジスタのINMSEL値をセットすることにより有効化されます。

抵抗器で分圧された電圧を使用しないときには、消費電力削減のために抵抗器ブリッジと増幅器を無効化することができます。抵抗器ブリッジを切断すると、1/4 VREF_COMP、1/2 VREF_COMP、および3/4 VREF_COMPレベルはVREF_COMPと等しくなります。

図316. スケーラ



37.5 COMP低消費電力モード

表356.低消費電力モードでのコンパレータの動作

モード	説明
SLEEP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスはSLEEPモードから復帰します。
STOP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスはSTOPモードから復帰します。
STANDBY	COMPレジスタはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

37.6 COMP割込み

コンパレータの出力は拡張割込み／イベントコントローラ（EXTI）に内部的に接続されます。各コンパレータには専用のEXTIラインがあり、割込みまたはイベントを生成することができます。低消費電力モードを終了するときにも同じ方法が使用されます。

詳細については、セクション 23 : 拡張割込み／イベントコントローラ（EXTI）を参照してください。

COMPx割込みを有効にするには、次のシーケンスを実行します。

1. 割込みモードでCOMPx出力イベントに対応するEXTIラインを設定して有効にし、立ち上がり、立ち下がり、または両方のエッジ感度を選択します。
2. 対応するEXTIラインにマップされたNVIC IRQチャネルを設定し、有効にします。
3. COMPxを有効にします。

表357.割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
COMP1出力	COMP1_CSRのVALUE	EXTI経由	あり	あり	なし
COMP2出力	COMP2_CSRのVALUE	EXTI経由	あり	あり	なし

37.7 COMP レジスタ

37.7.1 COMP1制御およびステータスレジスタ（COMP1_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	BLANKSEL[4:0]					PWRMODE[1:0]		HYST[1:0]	
rw	r						rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POLARITY	WINOUT	Res.	Res.	WIN MODE	INPSEL[2:0]			INMSEL[3:0]				Res.	Res.	Res.	EN
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw				rw

ビット31 **LOCK**：COMP1_CSRレジスタロック

このビットは、ソフトウェアでセットされ、リセットでクリアされます。これは、COMP1_CSRの内容全体をロックします。

0：COMP1_CSR読出し／書込みビットをソフトウェアによって書き込むことができます。

1：COMP1_CSRビットをソフトウェアによって読み出すことができますが、書き込むことはできません。

ビット30 **VALUE**：COMP1出力ステータス

このビットは読出し専用です。このビットは、極性セレクタとブランキングの後のCOMP1出力のレベルを反映します（図315を参照）。

ビット29:25 予約済みであり、リセット値に保持する必要があります。

ビット24:20 **BLANKSEL[4:0]**：COMP1ブランキングソースセレクタ

このフィールドは、ソフトウェアによって制御され（ロックされていない場合）、コンパレータの出力ブランキング用のPWM信号を選択します（割当てについては表354を参照）。

ビット19:18 **PWRMODE[1:0]**：COMP1電力モードセレクタ

このビットフィールドは、ソフトウェアで 制御されます（ロックされていない場合）。このビットでは、電力消費とその結果としてCOMP1の速度を選択します。

00：ハイスピード

01 ~ 10：中程度の速度と電力

11：超低消費電力

ビット17:16 **HYST[1:0]**：COMP1ヒステリシスセレクタ

このビットフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP1ヒステリシスを選択します。

00：なし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット15 **POLARITY**：COMP1極性セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP1出力極性を選択します。

0：反転されません。

1：反転されます。

ビット14 **WINOUT**：COMP1出力セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP1出力を選択します。

このビットは、ウィンドウコンパレータ機能がサポートされていない場合は0にしておく必要があります。

0：COMP1_VALUE

1：COMP1_VALUE XOR COMP2_VALUE（ウィンドウモードで必要、図313を参照)

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **WINMODE**：ウィンドウモードのためのCOMP1非反転入力セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP1のCOMP1_INP入力のための信号を選択します。

このビットは、ウィンドウコンパレータ機能がサポートされていない場合は0にしておく必要があります。

0：INPSEL[1:0] で選択した信号

1：COMP2のCOMP2_INP信号（ウィンドウモードで必要、図313を参照）

ビット10:8 **INPSEL[2:0]**：非反転入力用のCOMP1 信号セレクタ

このフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、非反転入力COMP1_INP用の信号を選択します（割当てについては表350を参照）。

ビット7:4 **INMSEL[3:0]**：反転入力INM用の COMP1 信号セレクタ

このフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、反転入力COMP1_INM用の信号を選択します（割当てについては表351を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：COMP1の有効化

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP1を有効化します。

0：COMP1は無効です。

1：COMP1は有効です。

37.7.2 COMP2制御およびステータスレジスタ（COMP2_CSR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	BLANKSEL[4:0]					PWRMODE[1:0]		HYST[1:0]	
rw	r						rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POLARITY	WIN OUT	Res.	Res.	WIN MODE	Res.	INPSEL[1:0]		INMSEL[3:0]				Res.	Res.	Res.	EN
rw	rw			rw		rw	rw	rw	rw	rw	rw				rw

ビット31 **LOCK**：COMP2_CSRレジスタロック

このビットは、ソフトウェアでセットされ、リセットでクリアされます。これは、COMP2_CSRの内容全体をロックします。

0：COMP2_CSR読出し／書込みビットをソフトウェアによって書き込むことができます。

1：COMP2_CSRビットをソフトウェアによって読み出すことができますが、書き込むことはできません。

ビット30 **VALUE**：COMP2出力ステータス

このビットは読出し専用です。このビットは、極性セレクタとブランキングの後のCOMP2出力のレベルを反映します（図315を参照）。

ビット29:25 予約済みであり、リセット値に保持する必要があります。

ビット24:20 **BLANKSEL[4:0]**：COMP2ブランキングソースセレクタ

このフィールドは、ソフトウェアによって制御され（ロックされていない場合）、コンパレータの出力ブランキング用のPWM信号を選択します（割当てについては表355を参照）。

ビット19:18 **PWRMODE[1:0]**：COMP2電力モードセレクタ

このビットフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、電力消費とその結果としてCOMP2の速度を選択します。

00：ハイスピード

01 ~ 10：中程度の速度と電力

11：超低消費電力

ビット17:16 **HYST[1:0]**：COMP2ヒステリシスセレクタ

このビットフィールドは、ソフトウェアで 制御されます（ロックされていない場合）。このビットでは、COMP2ヒステリシスを選択します。

00：なし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット15 **POLARITY**：COMP2極性セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP2出力極性を選択します。

0：反転されません。

1：反転されます。

ビット14 **WINOUT**：COMP2出力セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP2出力を選択します。

0：COMP2_VALUE

1：COMP1_VALUE XOR COMP2_VALUE（ウィンドウモードで必要、図313を参照)

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **WINMODE**：ウィンドウモードのためのCOMP2非反転入力セレクタ

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP2のCOMP2_INP入力のための信号を選択します。

0：INPSEL[1:0] で選択した信号

1：COMP1のCOMP1_INP信号（ウィンドウモードで必要、図313を参照）

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **INPSEL[1:0]**：非反転入力用のCOMP2 信号セレクタ

このフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、非反転入力COMP2_INP用の信号を選択します（割当てについては表352を参照）。

ビット7:4 **INMSEL[3:0]**：反転入力INM用の COMP2 信号セレクタ

このフィールドは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、反転入力COMP2_INM用の信号を選択します（割当てについては表353を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：COMP2の有効化

このビットは、ソフトウェアで制御されます（ロックされていない場合）。このビットでは、COMP2を有効化します。

0：COMP2は無効です。

1：COMP2は有効です。

37.7.3 COMPレジスタマップ

表358.COMPレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	COMP1_CSR	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	BLANKSEL[4:0]					PWRMODE[1:0]		HYST[1:0]		POLARITY	WINOUT	Res.	Res.	WINMODE	INPSEL[2:0]			INPSEL[3:0].				Res.	Res.	Res.	EN
	リセット値	0	0						0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0				0
0x04	COMP2_CSR	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	BLANKSEL[4:0]					PWRMODE[1:0]		HYST[1:0]		POLARITY	WINOUT	Res.	Res.	WINMODE	Res.	INPSEL[1:0]		INPSEL[3:0].				Res.	Res.	Res.	EN
	リセット値	0	0						0	0	0	0	0	0	0	0	0	0	0			0		0	0	0	0	0	0				0

レジスタ境界アドレスについては、セクション2.3を参照してください。

38 オペアンプ（OPAMP）

38.1 OPAMPの概要

STM32U535/545デバイスは1台の操作可能な増幅器を内蔵、STM32U575/585/59x/5Ax/5Fx/5Gxデバイスは2台の操作可能な増幅器を内蔵、どちらも2つの入力と1つの出力を装備。3つのI/Oは外部ピンにつなげることができ、あらゆる種類の外部相互接続が可能です。オペアンプは、フォロアまたはアンプとして2～16の非反転ゲイン範囲で内部的に設定できます。

正入力は、内部DACに接続できます。

出力は、内部ADCに接続できます。

38.2 OPAMPの主な機能

- レールツーレール入力電圧範囲
- 低入力バイアス電流
- 低入力オフセット電圧
- 低消費電力モード
- スルーレートが改善されるハイスピードモード
- 高速ウェイクアップ時間
- 1.6MHzのゲイン周波数

38.3 OPAMPの機能説明

OPAMPにはいくつかのモードがあります。

各OPAMPは個別に有効化でき、無効になっているときの出力はハイインピーダンスです。

有効時は、較正モードにしてからOPAMPのすべての入出力を切断したり、機能モードにしたりできます。

低消費電力モード、ハイスピードモード、通常モードの3種類の機能モードがあります。機能モードでは、OPAMPの入出力はセクション38.3.3に示すように接続されます。

38.3.1 OPAMPのリセットおよびクロック

オペアンプのクロックはレジスタへのアクセスに必要です。アプリケーションがこれらのレジスタに対する読み書きアクセスを必要としない場合、ペリフェラルクロック有効化レジスタでクロックをオフに切り替えられます（RCC_APB3ENRのOPAMPENビットを参照）。

OPAMPx_CSRのOPAENビットは、OPAMPの動作を有効化または無効化します。OPAMPx_CSRのOPAENビットがセットされている場合、OPAMPレジスタの設定を変更することができます。ただし、スプリアス（ノイズ、グリッチ、オーバーシュートまたは飽和）が生じる可能性があります。設定を変更する場合、アプリケーションファームウェアによりこれらのスプリアスに対処する必要があります（OPAMP出力のADC結果を無視するなど）。

OPAMP出力が必要でなくなった場合、節電のためにOPAMPを無効化することができます。以前にセットしたすべての設定（較正を含む）は、OPAMPが無効化されても保持されます。

38.3.2 初期設定

OPAMPのデフォルト設定は、3つのIOが外部ピンに接続されている機能モードです。デフォルトモードでは、OPAMPは出荷時のトリミング値を使用します（出荷時のトリミング条件については、データシートの「電気的特性」セクションを参照してください。通常、温度は30°C、電圧は3Vです）。トリミング値は調整できます（セクション38.3.5を参照）。デフォルト設定で通常モードを使用する場合、標準のパフォーマンスが得られます。OPAMPを低消費電力モードに切り替えてパフォーマンスを抑えるように、OPAMPx_CSRのOPALPMビットをセットできます。通常モード、低消費電力モード、ハイスピードモードの特性は、データシートの「電気的特性」セクションで定義されています。利用する前に、OPAMP1_CSRのOPA_RANGEビットに1をセットする必要があります。

OPAMPx_CSRのOPAENビットがセットされるとすぐに、OPAMPは機能します。2個の入力ピンおよび出力ピンは、セクション38.3.3に定義されているように接続されており、デフォルト接続設定は変更できます。

注： **入力ピンおよび出力ピンは、対応するGPIOx_MODERレジスタでアナログモード（デフォルト状態）で設定する必要があります。**

38.3.3 信号接続

OPAMPのピン接続は、OPAMPx_CSRレジスタで決まります。

オペアンプ（OPAMP1とOPAMP2）の接続を次の表に記載しています。

表359.オペアンプの可能な接続(1)

信号	ピン	内部	コメント
OPAMP1_VINM	PA1または専用ピン(2)	OPAMP1_OUTまたはPGA	OPAMODEおよびVM_SELによって制御されます。
OPAMP1_VINP	PA0	dac1_out1	VP_SELビットによって制御されます。
OPAMP1_VOUT	PA3	ADC1_IN8 ADC2_IN8	OPAMPが有効になっているときにピンが接続されます。 ADC入力は、ADCによって制御されます。
OPAMP2_VINM	PA7または専用ピン (2)	OPAMP2_OUTまたはPGA	ビットOPAMODEおよびVM_SELによって制御されます。
OPAMP2_VINP	PA6	dac1_out2	VP_SELビットによって制御されます。
OPAMP2_VOUT	PB0	ADC1_IN15 ADC2_IN15 ADC4_IN18	OPAMPが有効になっているときにピンが接続されます。 ADC入力は、ADCによって制御されます。

1. STM32U535/545デバイスにはOPAMP1のみが内蔵されています。
2. 専用ピンは、BGA132/169/216パッケージでのみ使用できます。この設定で最低の入力バイアス電流を実現します（データシートを参照）。

38.3.4 OPAMPモード

すべてのOPAMPの入出力に、端子からアクセスできます。増幅器は次のような複数の設定環境で使用できます。

- スタンドアロンモード（外部ゲイン設定モード）
- フォロア設定モード
- PGAモード

出力インピーダンスを最小限に抑えるために、アンプ出力ピンは出力パッドに直接接続されています。アンプがPGAとして設定され、ADCチャネルにのみ接続されていても、汎用I/Oとして使用することはできません。

信号のインピーダンスを、（ソースの抵抗性電圧降下が原因で）重大な乱れをつくる入力リークが発生しないレベル以下に維持する必要があります。詳細については、データシートの「電気的特性」のセクションを参照してください。

スタンドアロンモード（外部ゲイン設定モード）

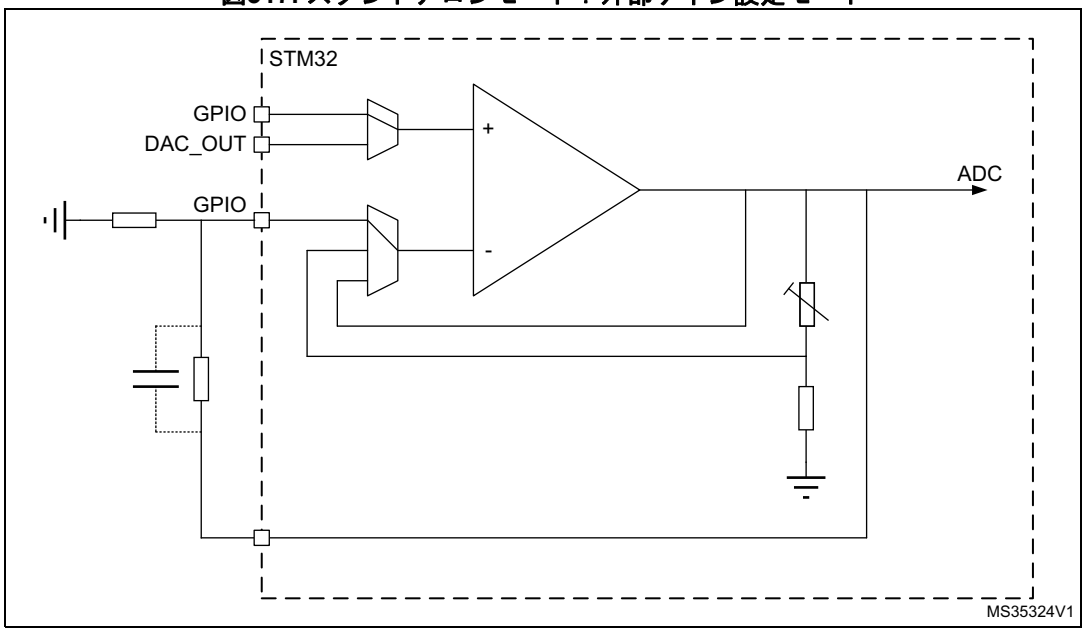
スタンドアロンモードでOPAMPを使用する手順を以下で説明します。

1. OPAMPx_CSRのデフォルト値とGPIOx_MODERのデフォルト状態を維持します。
2. OPAMPx_CSRのOPAENがセットされるとすぐに、2個の入力ピンと出力ピンがオペアンプに接続されます。

このデフォルト設定では、出荷時トリミング値を使用して、通常モード（標準のパフォーマンス）で動作します。OPAMPの動作は、OPAMPx_CSRの次のビットで変更できます。

- OPALPMに1がセットされている場合、OPAMPは節電のために低消費電力モードに切り替わります。
- OPAHSM に 1 がセットされている場合、OPAMP は高いスルーレートが得られるようにハイスピードモードに切り替わります。
- USERTRIMに1がセットされている場合、入力オフセット値をトリミングできます。

図317. スタンドアロンモード：外部ゲイン設定モード



フォロア設定モード

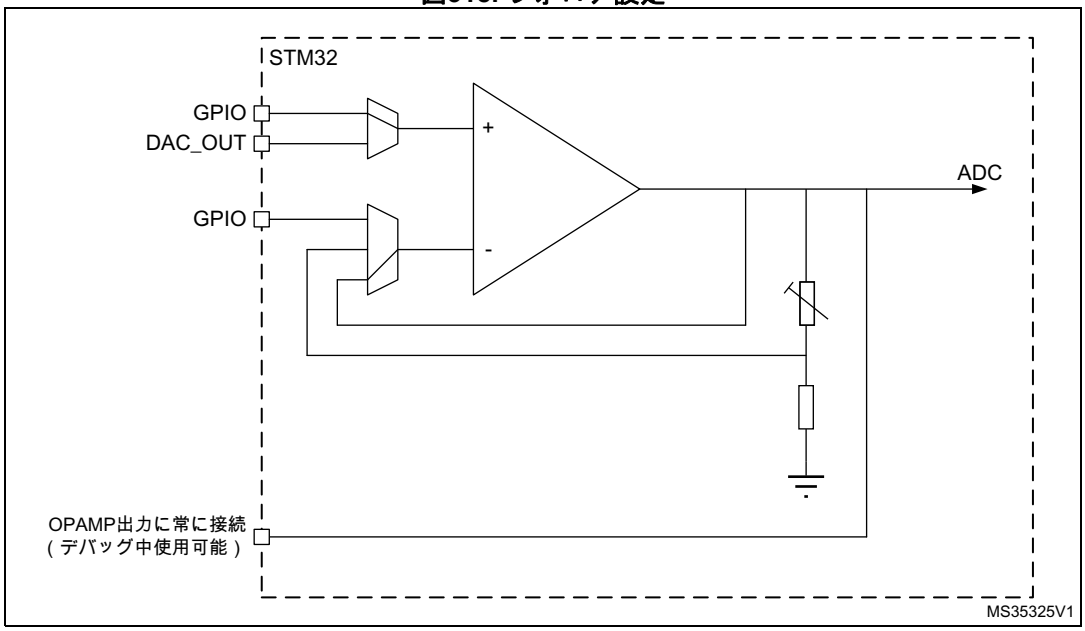
次のモードでOPAMPを使用する手順を以下で説明します（OPAMPx_CSRのすべてのビット）。

1. OPAMODE[1:0] = 11にセットします（内部フォロア）。
2. VP_SELを0にクリアします（OPAMPx_VINPに接続されたGPIOで、本書ではVINPという名前）。
3. OPAENに1がセットされるとすぐに、VINPの信号がOPAMP_VOUTピンにコピーされます。

注： **OPAMP_VINMに対応するピンは、他の用途のために解放されます。**

OPAMP出力の信号もADC入力とみなされます。結果として、フォロアモードに設定されたOPAMPを使用して、入力信号周波数がオペアンプのゲイン周波数の仕様と互換性があると想定して、入力信号をADC入力に供給する前にインピーダンス適合を実行できます。

図318. フォロア設定



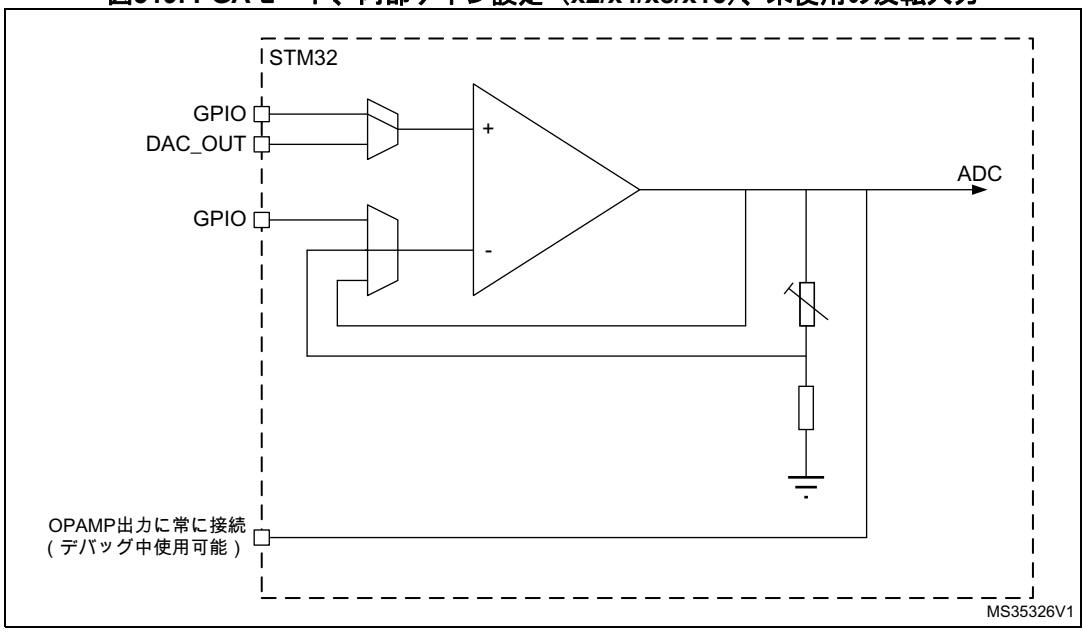
プログラマブルゲインアンプモード

入力信号を増幅するためにOPAMPを使用する手順をここで説明します。

1. OPAMODE[1:0] に10をセットします（内部PGAが有効）。
2. PGA_GAIN[1:0] に、選択したOPAMPx_CSRのPGAゲイン（2、4、8、または16）をセットします。
3. OPAMPx_CSRのVM_SEL[1:0] に1xをセットします（反転入力は外部接続しない）。
4. OPAMPx_CSRのVP_SELに0をセットします（GPIOをVINPに接続）。
5. OPAMPx_CSRのOPAENがセットされるとすぐに、OPAMP_VINPピンの信号が選択されたゲインで増幅され、OPAMP_VOUTピンで確認できます。

注： **飽和を避けるために、入力電圧は選択されたゲインで分圧されたV_{DDA}未満にしておく必要があります。**

図319. PGAモード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力



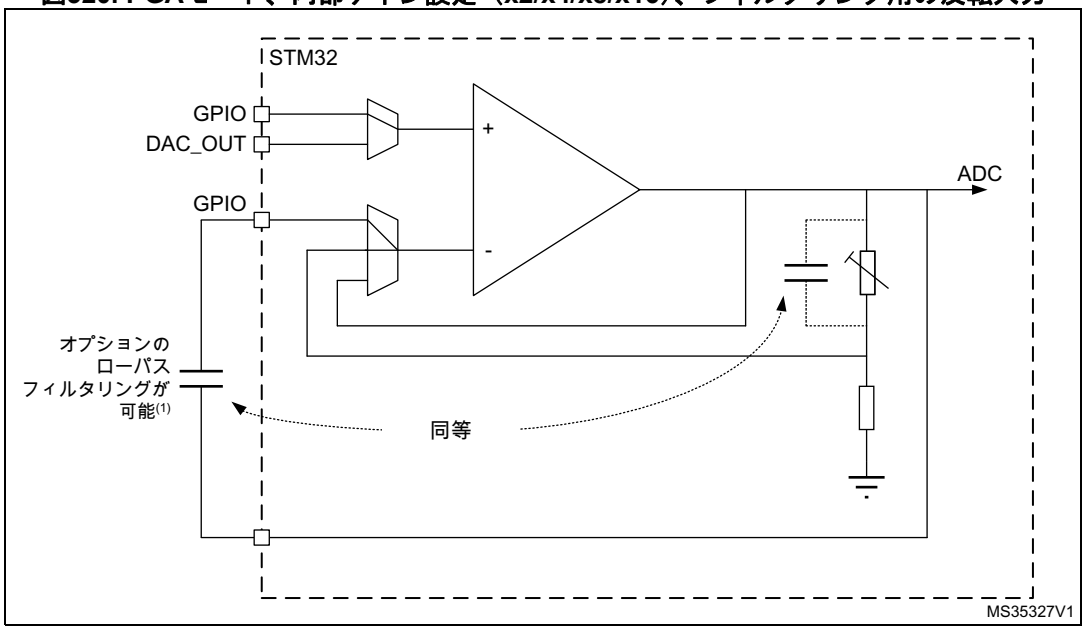
外部フィルタリングでのプログラム可能なゲインアンプモード

入力信号を増幅するためにOPAMPを使用する手順を以下で説明します（OPAMPx_CSRのすべてのビット）。

1. OPAMODE[1:0] に10を設定します（内部PGAが有効）。
2. PGA_GAIN[1:0] に、選択したPGAゲインをセットします（2、4、8、または16）。
3. VM_SEL[1:0] を00または01でクリアします（GPIOをVINMに接続）。
4. VP_SELを0でクリアします（GPIOをVINPに接続）。

VINPであらゆる外部接続が内部PGAと並列で使用できます。たとえば、フィルタリング目的でコンデンサをVOUTとVINMの間に接続できます（PGAレジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図320. PGAモード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力



1. ゲインはカットオフ周波数によって決まります。

38.3.5 較正

起動時、トリミング値はプリセットされた「出荷時」トリミング値に初期化されます。

各OPAMPオフセットを、ユーザがトリミングすることもできます。特定のレジスタでは、通常モードと低消費電力モードで異なるトリミング値を使用することができます。

較正の目的は、OPAMP入力オフセット電圧をできる限り相殺することです。較正回路を使えば、安定した電圧条件と温度条件の範囲で、入力オフセット電圧を +/-1.5mV未満まで低減できます。

各OPAMPおよび各モードに対して、N差分ペア用とP差分ペア用の2つのトリミング値をトリミングすることができます。

操作可能な各増幅器用にオフセットをトリミングするためのレジスタは次のとおりです。

- OPAMPx_OTR（通常モード用）
- OPAMPx_LPOTR（低消費電力モード用）

各レジスタは、P差分ペアトリミング用の5個のビット、N差分ペアトリミングの5個のビットで構成されています。これらは「ユーザ」による値です。

ユーザはOPAMPx_CSRのUSERTRIMビットにより、「出荷時」トリミング値から「ユーザ」トリミング値に切り替えることができます。このビットは起動時にリセットされ、「出荷時」値がデフォルトでOPAMPトリミングレジスタに適用されます。

トリミング値の変更は、較正モードまたは機能モードで行えます。

オフセットトリミングレジスタは、通常、較正動作の初期化後に設定されます（OPAMPx_CSRのCALONが1に設定されます）。CALON = 1の場合、OPAMP入力が機能環境から切断されます。

OPAMPx_CSRのCALSELに1をセットすると、P差分ペアのオフセット較正が初期化されます（低電圧基準を使用）。

CALSELを0にクリアすると、N差分ペアのオフセット較正が初期化されます（高電圧基準を使用）。

CALON = 1の場合、OPAMPx_CSRのCALOUTビットに、CALSELとOPALPMによって選択されたトリミング値の影響が反映されます。CALOUTの値が2つの連続したトリミング値で切り替わる場合、その2つの値が最適なトリミング値であることを示しますCALOUTフラグが安定するには、トリミング値の変更後、最大1 msが必要です（データシートの「電気的特性」のセクションにある$t_{OFFTRIM}$ 最大遅延仕様を参照）。

注： **トリミング値が最適なトリミング値に近いほど、安定化にかかる時間が長くなります（いずれの場合でも最大安定時間が1ms未満のままの状態）。**

較正動作が完了したら、OPAMPx_CSRのOPAHSMを0にクリアする必要があります。

表360.動作モードと較正

モード	制御ビット				出力	
	OPAEN	OPALPM	CALON	CALSEL	V_{OUT}	CALOUT
通常動作モード	1	0	0	X	アナログ	0
低消費電力モード	1	1	0	X	アナログ	0
パワーダウン	0	X	X	X	Z	0
通常モードのオフセット較正高	1	0	1	0	アナログ	X
通常モードのオフセット較正低	1	0	1	1	アナログ	X
低消費電力モードのオフセット較正高	1	1	1	0	アナログ	X
低消費電力モードのオフセット較正低	1	1	1	1	アナログ	X

較正手順

オペアンプのいずれか1つを完全に較正するための手順を示します。

1. OPAMPx_CSRのOPAENに1をセットしてOPAMPを有効化し、OPAMP1_CSRでOPA_RANGE = 1をセットします。
2. OPAMPx_CSRのOPAHSMを0にクリアします。
3. OPAMPx_CSRのCALONおよびUSERTRIMに1をセットします。
4. 較正モードを選択します（表360を参照）。手順4と5を4回繰り返します。初回は、OPAMPx_CSRでOPALPM = 0およびCALSEL = 0をセットした状態で、通常モード、オフセット較正高（N差分ペア）を選択します。
5. OPAMPx_CSRのCALOUTが1に変わるまで、OPAMPx_OTRのTRIMOFFSETN[4:0] を0から増やします。

   OPAMP_OTRレジスタの書込みとCALOUT値の読出しの間に、データシートの「電気的特性」のセクションで指定された$t_{OFFTRIM}$ 最大遅延を待って、正しいCALOUT値を取得します。

   この転流は、オフセットが正しく補正されることと、対応するトリミングコードをOPAMP_OTRに保存する必要があることを示します。
6. 次の場合、手順4～5を繰り返します。
   - 通常モードかつオフセット較正低
   - 低消費電力モードかつオフセット較正高
   - 低消費電力モードかつオフセット較正低

   モードを使用していない場合、対応する較正はスキップできます。

すべてのOPAMPを同時に較正できます。

注： **較正フェーズ全体において：**

- OPAMP出力の外部接続によって、500 µAを超えるプルアップまたはプルダウンはしないでください。

- OPAMODE[1:0] に00または01（PGAを無効化）、あるいは11（内部フォロア）をセットする必要があります。

38.4 OPAMP低消費電力モード

表361.低消費電力モードがOPAMPに与える影響

モード	説明
SLEEP	影響はありません。
Stop0/1/2	影響はありません。OPAMPレジスタの内容は保たれます。
Stop3	OPAMPは無効で、レジスタの内容は保たれます。
STANDBY	OPAMPレジスタはパワーダウンされ、STANBYモード終了後に再初期化する必要があります。

38.5 OPAMPレジスタ

これらのレジスタは、ワードによってのみアクセスできます（バイト、ハーフワードはサポートされていません）。

38.5.1 OPAMP1制御／ステータス･レジスタ（OPAMP1_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OPA_RANGE	OPAHSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL[1:0]		Res.	Res.	PGA_GAIN[1:0]		OPAMODE[1:0]		OPALPM	OPAEN
r	rw	rw	rw		rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31 **OPA_RANGE**：OPAMP範囲設定

このビットは、OPAMPを有効化する前にセットする必要があり、すべてのOPAMPインスタンスに影響します。

0：予約済み

1：OPAMPの範囲のセット

ビット30 **OPAHSM**：OPAMPハイスピードモード

このビットは、通常モードと低消費電力モードの両方に有効です。

0：通常モード（標準スルーレート）

1：スルーレートを向上させるために消費量が増加

ビット29:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **CALOUT**：OPAMP較正出力

較正モード中、この信号がトグルされると、オフセットがトリミングされます。

ビット14 **USERTRIM**：「出荷時」または「ユーザ」オフセットトリミング値の選択
このビットは、通常モードおよび低消費電力モードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット13 **CALSEL**：較正の選択
0：NMOS較正（200mVをOPAMP入力に適用）
1：PMOS較正（V_{DDA}-200mVをOPAMP入力に適用）

ビット12 **CALON**：較正モードの有効化
0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **VP_SEL**：非反転入力の選択
0：GPIOがVINPに接続されます。
1：DACがVINPに接続されます。

ビット9:8 **VM_SEL[1:0]**：反転入力の選択
これらのビットは、OPAMODE = 00、01または10の場合のみ使用されます。
00：GPIOがVINMに接続されます（フィルタリングのためのPGAモードでも有効）。
01：専用の低リーク入力がVINMに接続されます（フィルタリングのためのPGAモードでも有効）。
1x：外部接続せずに反転入力します。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **PGA_GAIN[1:0]**：OPAMPのプログラム可能アンプゲイン値
00：内部PGAゲイン2
01：内部PGAゲイン4
10：内部PGAゲイン8
11：内部PGAゲイン16

ビット3:2 **OPAMODE[1:0]**：OPAMP PGAモード
00および01：内部PGAは無効です。
10：内部PGAは有効で、ゲインがPGA_GAINでプログラムされています。
11内部フォロア

ビット1 **OPALPM**：OPAPM低消費電力モード
この設定を変更するには、OPAMPを無効化する必要があります
0：通常モード
1：低消費電力モード

ビット0 **OPAEN**：OPAMPの有効化
0：OPAMPは無効です。
1：OPAMPは有効です。

38.5.2 通常モードでのOPAMP1オフセットトリミングレジスタ（OPAMP1_OTR）

アドレスオフセット：0x04

リセット値：0x0000 XXXX

XXXXは出荷時のトリミング値です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMOFFSETP[4:0]					Res.	Res.	Res.	TRIMOFFSETN[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **TRIMOFFSETP[4:0]**：PMOS差分ペアのトリミング

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **TRIMOFFSETN[4:0]**：NMOS差分ペアのトリミング

38.5.3 低消費電力モードでのOPAMP1オフセットトリミングレジスタ（OPAMP1_LPOTR）

アドレスオフセット：0x08

リセット値：0x0000 XXXX

XXXXは出荷時のトリミング値です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMLPOFFSETP[4:0]					Res.	Res.	Res.	TRIMLPOFFSETN[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **TRIMLPOFFSETP[4:0]**：PMOS差分ペアの低消費電力モードトリミング

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **TRIMLPOFFSETN[4:0]**：NMOS差分ペアの低消費電力モードトリミング

38.5.4 OPAMP2制御／ステータスレジスタ（OPAMP2_CRS）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタの有無は、STM32U5シリーズデバイスで異なります。セクション38.1を参照してください。ない場合、予約済みと考え、リセット値のままにしてください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	OPAHSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL[1:0]		Res.	Res.	PGA_GAIN[1:0]		OPAMODE[1:0]		OPALPM	OPAEN
r	rw	rw	rw		rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **OPAHSM**：OPAMPハイスピードモード
このビットは、通常モードとハイスピードモードの両方に有効です。
0：通常モード（標準スルーレート）
1：スルーレートを向上させるために消費量が増加

ビット29:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **CALOUT**：OPAMP較正出力
較正モード中、この信号がトグルされるとオフセットがトリミングされます。

ビット14 **USERTRIM**：「出荷時」または「ユーザ」オフセットトリミング値の選択
このビットは、通常モードおよび低消費電力モードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット13 **CALSEL**：較正の選択
0：NMOS較正（200mVをOPAMP入力に適用）
1：PMOS較正（V_{DDA}-200mVをOPAMP入力に適用）

ビット12 **CALON**：較正モードの有効化
0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **VP_SEL**：非反転入力の選択
0：GPIOがVINPに接続されます。
1：DACがVINPに接続されます。

ビット9:8 **VM_SEL[1:0]**：反転入力の選択
これらのビットは、OPAMODE = 00、01または10の場合のみ使用されます。
00：GPIOがVINMに接続されます（フィルタリングのためのPGAモードでも有効）。
01：専用の低リーク入力がVINMに接続されます（
フィルタリングのためのPGAモードでも有効）
1x：外部接続せずに反転入力します。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **PGA_GAIN[1:0]**：OPAMPのプログラム可能アンプゲイン値
00：内部PGAゲイン2
01：内部PGAゲイン4
10：内部PGAゲイン8
11：内部PGAゲイン16

ビット3:2 **OPAMODE[1:0]**：OPAMP PGAモード
00および01：内部PGAは無効です。
10：内部PGAは有効で、ゲインがPGA_GAINでプログラムされています。
11内部フォロア

ビット1 **OPALPM**：OPAPM低消費電力モード
この設定を変更するには、OPAMPを無効化する必要があります
0：通常モード
1：低消費電力モード

ビット0 **OPAEN**：OPAMPの有効化
0：OPAMPは無効です。
1：OPAMPは有効です。

38.5.5 通常モードでのOPAMP2オフセットトリミングレジスタ（OPAMP2_OTR）

アドレスオフセット：0x14

リセット値：0x0000 XXXX

XXXXは出荷時のトリミング値です。

このレジスタの有無は、STM32U5シリーズデバイスで異なります。セクション38.1を参照してください。ない場合、予約済みと考え、リセット値のままにしてください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMOFFSETP[4:0]					Res.	Res.	Res.	TRIMOFFSETN[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **TRIMOFFSETP[4:0]**：PMOS差分ペアのトリミング

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **TRIMOFFSETN[4:0]**：NMOS差分ペアのトリミング

38.5.6 低消費電力モードでのOPAMP2オフセットトリミングレジスタ（OPAMP2_LPOTR）

アドレスオフセット：0x18

リセット値：0x0000 XXXX

XXXXは出荷時のトリミング値です。

このレジスタの有無は、STM32U5シリーズデバイスで異なります。セクション38.1を参照してください。ない場合、予約済みと考え、リセット値のままにしてください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMLPOFFSETP[4:0]					Res.	Res.	Res.	TRIMLPOFFSETN[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **TRIMLPOFFSETP[4:0]**：PMOS差分ペアの低消費電力モードトリミング

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **TRIMLPOFFSETN[4:0]**：NMOS差分ペアの低消費電力モードトリミング

38.5.7 OPAMPレジスタマップ

表362.OPAMPレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	OPAMP1_CSR	OPA_RANGE	OPAHSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL[1:0]		Res.	Res.	PGA_GAIN [1:0]		OPAMODE [1:0]		OPALPM	OPAEN
	リセット値	0	0															0	0	0	0		0	0	0			0	0	0	0	0	0
0x04	OPAMP1_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				X	X	X	X	X				X	X	X	X	X
0x08	OPAMP1_ LPOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMLP OFFSETP[4:0]					Res.	Res.	Res.	TRIMLP OFFSETN[4:0]				
	リセット値																				X	X	X	X	X				X	X	X	X	X
0x10	OPAMP2_CSR	Res.	OPAHSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL[1:0]		Res.	Res.	PGA_GAIN[1:0]		OPAMODE[1:0]		OPALPM	OPAEN
	リセット値		0															0	0	0	0		0	0	0			0	0	0	0	0	0
0x14	OPAMP2_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				X	X	X	X	X				X	X	X	X	X
0x18	OPAMP2_ LPOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMLP OFFSETP[4:0]					Res.	Res.	Res.	TRIMLP OFFSETN[4:0]				
	リセット値																				X	X	X	X	X				X	X	X	X	X

レジスタ境界アドレスについては、セクション2.3 を参照してください。

39 多機能デジタル･フィルタ（MDF）

39.1 はじめに

多機能デジタル･フィルタ（MDG）は、外部シグマデルタ（ΣΔ）モジュレータの接続専用の高性能モジュールです。主に、次の用途が対象となります。

- オーディオ･キャプチャ信号
- モータ制御
- スマート･メータ

MDFは、最大6デジタル･シリアル･インタフェース （SITFx）とデジタル･フィルタ（DFLTx）を備えていて、最大24bitの最終分解能に対応する柔軟なデジタル処理オプションが利用できます。

MDFのDFLTx には、オーディオ･デジタル･フィルタ（ADF）のフィルタも含まれます。

MDFはシリアル･インタフェースを通して、各デジタル･センサからのストリームを受信できます。

MDFは次の規格に対応していて、各種ΣΔ モジュレータ･センサの接続が可能です。

- SPIインタフェース
- マンチェスタ符号化1線インタフェース
- PDMインタフェース

柔軟なビットストリーム･マトリックス（BSMX）により、フィルタへの受信ビットストリームの接続が可能になります。

MDFは入力データ･ストリームをデシメートされたクリーンなデジタル･データ･ワードに変換します。この変換は、ローパス･デジタル･フィルタとデシメーション･ブロックにより行われます。また、ハイパス･フィルタまたはDCオフセット補正ブロックを挿入することもできます。

変換速度および分解能は、フィルタ･タイプ、フィルタ段数、デシメーション比、積分器の長さといったデジタル処理に関する設定可能なパラメータに合わせて調整できます。最大出力データの分解能は24bitです。変換モードには、シングル変換モードと連続モードの2種類があります。データは、DMAを介してシステムRAMバッファに自動的に保存され、ソフトウェアの負荷を軽減できます。

変換の開始を制御するには、柔軟なトリガ･インタフェースを使用できます。このタイミング制御では、同時変換をトリガすることも、変換の間にプログラム可能な遅延を挿入することもできます。

MDFは、限界外検出回路（OLD）機能を備えています。OLDはデジタル･フィルタ･チェーンごとに1つずつあります。独立したプログラム可能な閾値が各OLDで利用でき、これは過電流検出に最適です。

短絡検出回路（SCD）も、選択したすべてのビットストリームで利用できます。SCDでは、非常に短い遅延時間で短絡状態を検出できます。短絡状態を定義するために、独立したプログラム可能な閾値が用意されています。

デジタル処理は、カーネル･クロックのみを使用して行われます。データ転送が必要な場合、または特定のイベントによりシステム･プロセッサの応対処置がリクエストされた場合、MDFではバス･インタフェース･クロック（AHBクロック）のみがリクエストされます。

39.2 MDFの主な機能

- AHBインタフェース
- 最大6 シリアル･デジタル入力：
  - さまざまなデジタル･センサを接続するために設定可能なSPIインタフェース
  - 設定可能なマンチェスタ符号化インタフェースのサポート
  - PDMインタフェースとの互換性によりデジタル･マイクロフォンに対応
- ΣΔ モジュレータ用の2つの共通クロック入力／出力
- フィルタとデジタル入力を接続するための柔軟なマトリックス（ＢＳＭＸ）
- 内部ADCを接続するための2つの入力
- 次のような最大6柔軟なデジタル･フィルタ･パス
  - 設定可能なCICフィルタ：
    - 2つのCICフィルタ、つまり高分解能フィルタと限界外検出回路に分割可能
    - $Sinc^4$ フィルタで設定可能
    - $Sinc^5$ フィルタで設定可能
    - デシメーション比が調整可能
  - 帯域外除去とインバンド･リップルを向上させるための再形成フィルタ
  - ADCオフセットを解除するためのハイパス･フィルタ
  - オフセット･エラーの解除
  - 制御の獲得
  - 飽和ブロック
  - 限界外検出回路
- 短絡検出回路
- クロックなし検出回路
- 16または24bitの符号付き出力データ分解能
- 連続またはシングル変換
- 各ビットストリームを個別に遅延させることが可能
- 各種トリガが使用可能
- 限界外または短絡検出回路イベントでのブレーク生成
- STOPモードでの自律型機能
- DMAを使用した変換データの読出しが可能
- 割込みサービス

39.3 MDFの実装

デバイスには1つのMDFインスタンスと1つのADFインスタンスが内蔵されています。これらは共通の機能を備えたデジタル･フィルタです。

表363.ADF/MDFの機能(1)

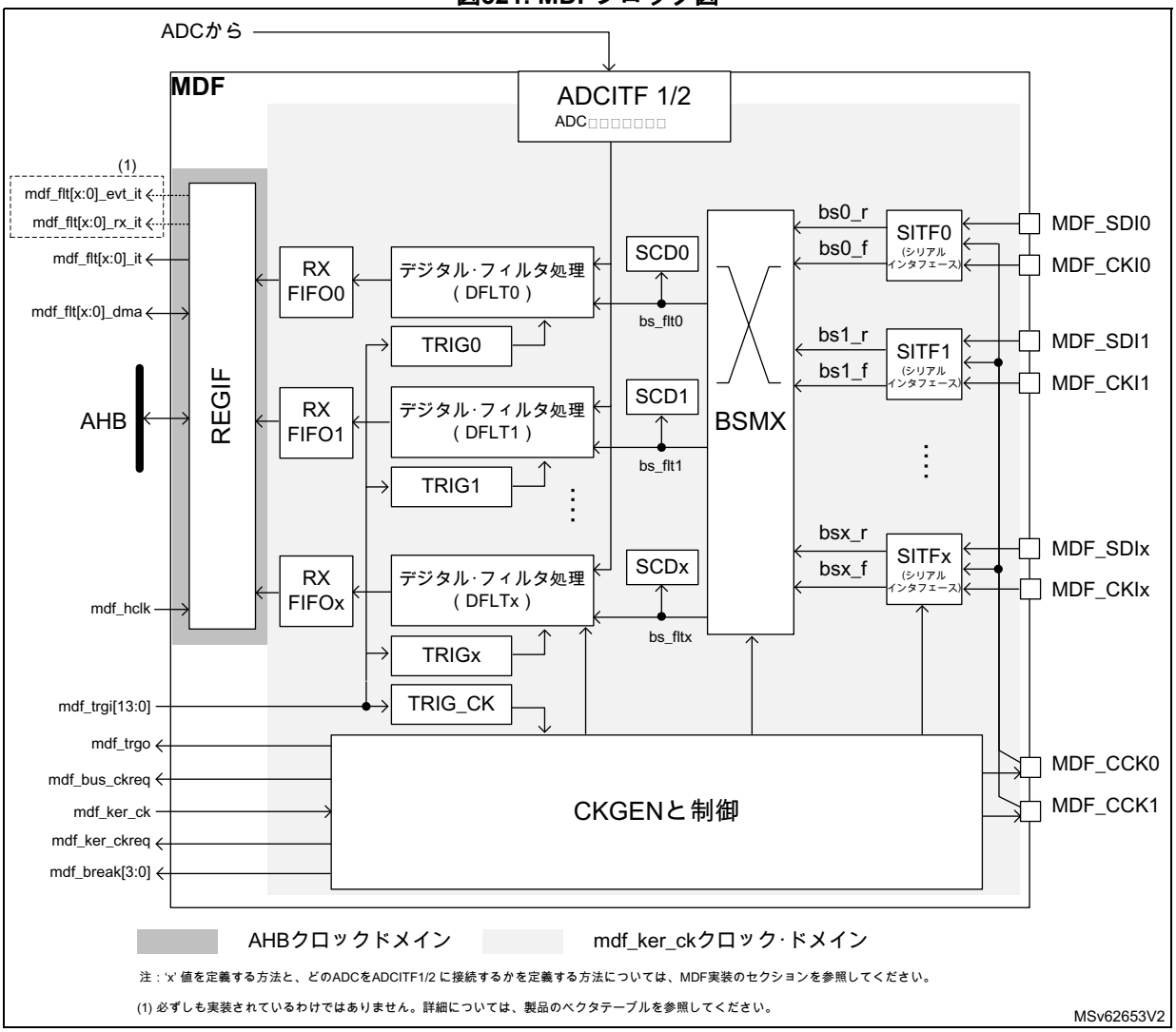
モードまたは機能	ADF1すべてのデバイス	MDF1 STM32U535/545	MDF1 STM32U575/585	MDF1 STM32U59x/5Ax 5Fx/5Gx
フィルタの数（DFLTx）とシリアル･インタフェース（SITFx）	1	2	6	
ピンに接続されたMDF_CKIy/ADF_CKI0	-	-	X	
サウンド･アクティビティ検出（SAD）	X	-		
RXFIFOの深さ（24bit･ワード数）	4	4		
ADCITF1に接続されたADC	-	ADC1		
ADCITF2に接続されたADC	-	-		ADC2
モータ専用機能（SCD、OLD、OEC、INT、スナップショット、ブレーク）	-	X		
CIC4、CIC5を備えたメイン･パス	X			
CIC1、2、3またはFastSincを備えたメイン･パス	-			
RSFLT、HPF、SAT、SCALE、DLY、破棄機能	X			
STOPモードでの自律動作	X(2)	X(3)		

1. “X” はサポートされています。“-” はサポートされていません。
2. STOP 0、STOP 1およびSTOP 2モードのみ。
3. STOP 0およびSTOP 1モードのみ。

39.4 MDFの機能説明

39.4.1 MDFブロック図

図321. MDFブロック図



39.4.2 MDFピンおよび内部信号

表364.MDF外部ピン

名前	信号タイプ	コメント
MDF_CKIy（y = 0 ~ 5）(1)	入力	外部センサからの専用クロック信号
MDF_CKIy（y = 0 ~ 5）(1)	入力	外部センサからのデータ信号
MDF_CCKy（y = 0、1）	入力／出力	外部センサ用のクロック出力、または外部センサからの共通クロック入力

1. 使用可能な入力の数は、フィルタの数によって決まります。詳細については、セクション39.3を参照してください。

表365.MDF内部信号

名前	信号タイプ	コメント
mdf_trgi[13:0]	入力	取得を制御するためのトリガ入力（詳細については次の表を参照）
mdf_trgo	出力	複数のMDFを同期させるためのトリガ出力
mdf_break[3:0]	出力	過電流検出回路または短絡検出回路からのブレーク信号イベント生成（詳細については次の表を参照）
mdf_flt[5:0]_dma(1)	入力／出力	各フィルタ処理チェーンのためのDMAリクエスト/確認応答信号
mdf_flt[5:0]_it (1)	出力	各MDFフィルタのためのグローバル割込み信号
mdf_flt[5:0]_rx_it (1)(2)	出力	各MDFフィルタのための受信割込み信号。
mdf_flt[5:0]_evt_it (1) (2)	出力	各MDFフィルタのためのイベント割込み信号。
mdf_bus_ckreq	出力	バス･インタフェース･クロック･リクエスト出力
mdf_ker_ckreq	出力	カーネル･クロック･リクエスト出力
mdf_ker_ck	入力	カーネル･クロック入力
mdf_hclk	入力	AHBバス･インタフェース･クロック入力
mdf_adcitf1_dat [15:0]	入力	ADCITF1データ入力
mdf_adcitf2_dat [15:0]	入力	ADCITF2データ入力

1. 使用可能な信号の数は、フィルタの数によって決まります。詳細については、セクション39.3を参照してください。
2. 常に接続されているわけではありません。詳細については、割込みベクタテーブルを参照してください。

次の表に、MDFのトリガ入力の接続方法を示します。

表366.MDFトリガ接続

トリガ名	トリガソース
mdf_trgi0	tim1_trgo
mdf_trgi1	tim1_trgo2
mdf_trgi2	tim8_trgo
mdf_trgi3	tim8_trgo2
mdf_trgi4	tim3_trgo
mdf_trgi5	tim4_trgo
mdf_trgi6	tim16_oc1
mdf_trgi7	tim6_trgo
mdf_trgi8	tim7_trgo
mdf_trgi9	adf1_sad_det （ADF1からのサウンド･アクティビティ検出信号）
mdf_trgi10	exti11
mdf_trgi11	exti15
mdf_trgi12	lptim1_ch1
mdf_trgi13	ADF1からのadf1_trgo信号

次の表に、MDFのブレーク出力の接続方法を示します。

表367.MDFブレーク接続

トリガ名	トリガソース
mdf_break0	tim1_brk_cmp7
mdf_break1	tim1_brk2_cmp7
mdf_break2	tim8_brk_cmp7
mdf_break3	tim8_brk2_cmp7

次の表に、ADCデータとMDFの接続方法を示します。

表368.MDF ADCデータ接続

ADCデータ･バス名	ADCソース
mdf_adcitf1_dat [15:0]	adc1_dat
mdf_adcitf2_dat [15:0]	adc2_dat(1)

1. STM32U59x/5Ax/5Fx/5Gxのみで使用できます。STM32U535/545/575/585では接続されません。

39.4.3 シリアル入力インタフェース（SITF）

SITFx 入力インタフェースを使用すると、ビットストリーム･マトリックス（BSMX）を通して、外部センサをデジタル･フィルタに接続することができます。SITFx シリアル･インタフェースは次のモードで設定できます。

- LF_MASTER SPIモード（低周波数）
- 通常SPIモード
- マンチェスタ･モード

SITFxインスタンスの量はフィルタの量と等しくなります。

各シリアル･インタフェースからのデータは、以下を行うために、任意のフィルタにルーティングすることができます。

- PDMからPCMへの変換
- 限界外検出
- 短絡検出

シリアル･インタフェースを有効化するには、対応する SITFENビットを1にセットします。インタフェースが有効化されると、外部 ΣΔ モジュレータからシリアル･データを受信します。

注： **シリアル･インタフェースを有効化する前に、mdf_proc_ck が有効化されていることを確認する必要があります（詳細は、セクション39.4.5 を参照）。**

SITFx の制御は、MDFシリアル･インタフェース制御レジスタx (MDF_SITFxCR)を通して行われます。

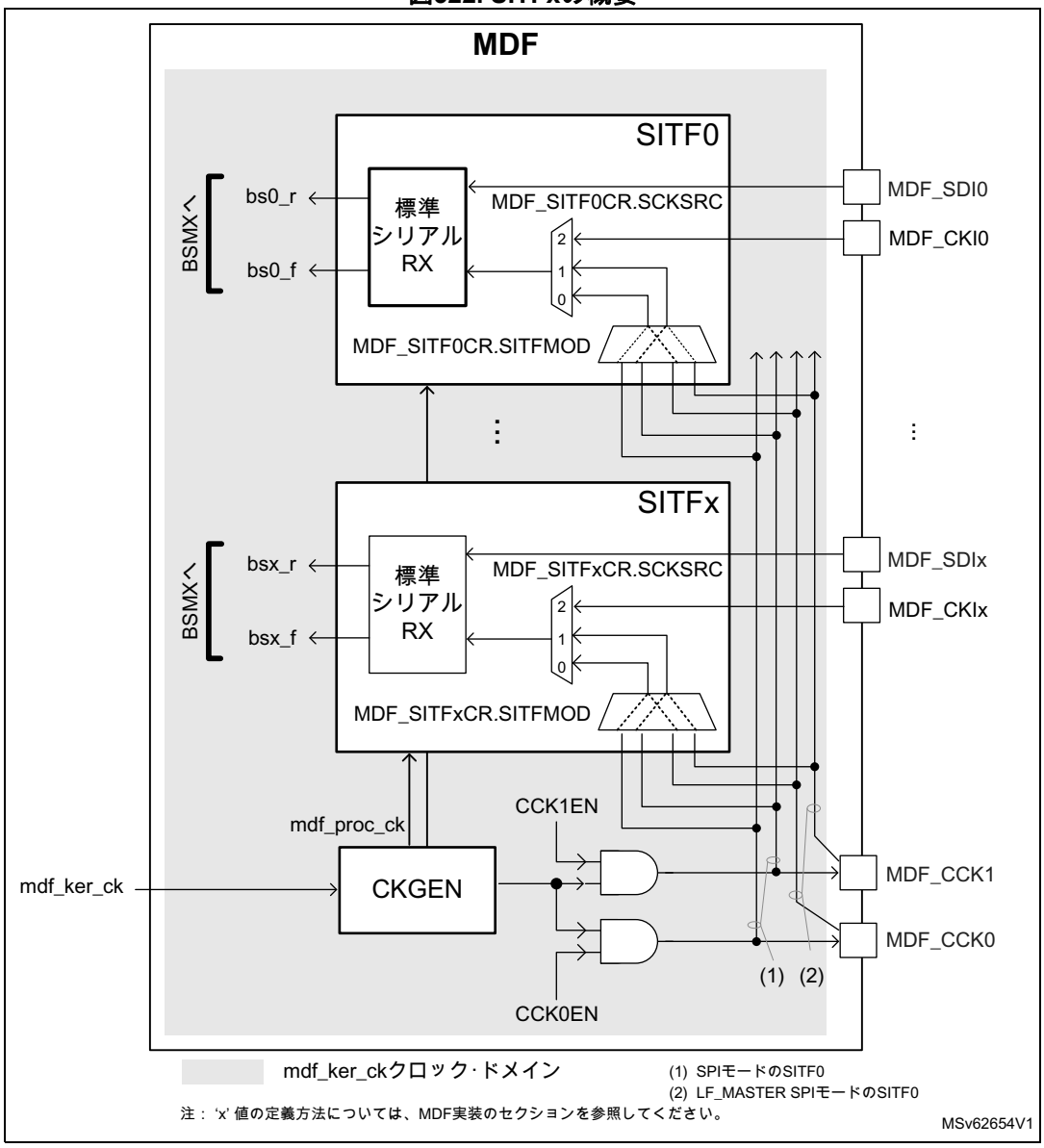
図322に示すように、各SITFについて、クロックの選択肢が多数用意されています。

- シリアル･インタフェースがSPIモードでプログラムされている場合、選択されるクロック･ソースはMDF_CCK0、MDF_CCK1、またはMDF_CKIxピンにあるクロックのコピーです（図322の2を参照）。
- シリアル･インタフェースがLF_MASTER SPIモードでプログラムされている場合、選択されるクロック･ソースはCCKDIVからMDF_CCK0またはMDF_CCK1ピンに直接供給されるクロックでなければなりません（図322 の1を参照）。この場合、MDF_CKIxを選択しないようにする必要があります。

詳細については、表369 を参照してください。

注： **共通クロック（MDF_CCK0またはMDF CCK1）を使用すると、複数のSITFx間で同じクロックを共有するのに有益です。**

図322. SITFxの概要



LF_MASTERおよび通常SPIモード

LF_MASTER SPIモードは、周波数がセンサ･クロックの2倍の高さであるmdf_proc_ckクロック周波数を使用できる特別なモードです。このモードは低消費電力ユースケース専用で、ロースピード･センサを使用します。

LF_MASTER SPIモードでは、MDFはMDF_CCK0およびMDF_CCK1ピンを通して外部センサにビットストリーム･クロックを供給し、シリアルデータ入力MDF_SDIx を通してビットストリーム･データを受信する必要があります。

各 SITFx について、最適なタイミングが得られるように、アプリケーションでは外部センサに供給されるのと同じクロック（MDF_CCK0 または MDF_CCK1）を選択する必要があります。選択は SCKSRC[1:0] を通して行います。

警告： MDF_CKIxピンはLF_MASTER SPIモードでは使用できません。

通常SPIインタフェースはLF_MASTER SPIよりも柔軟性の高いインタフェースですが、mdf_proc_ck周波数を少なくともセンサ･クロックの4倍にする必要があります。

アプリケーションでは、MDF_SDIx　ピンを通して受信したデータをキャプチャするために、MDF_CCK0、MDF_CCK1、またはMDF_CKIxクロックを選択することができます。

MDFでは、必要に応じて、MDF_CCK0またはMDF_CCK1を通してセンサへのクロックを生成できます。

すべてのSPIモードにおいて、リクエストされるI/Oの量を最適化するため、すべてのSITFで同じクロック入力（MDF_CCK0またはMDF_CCK1）を共有できます。

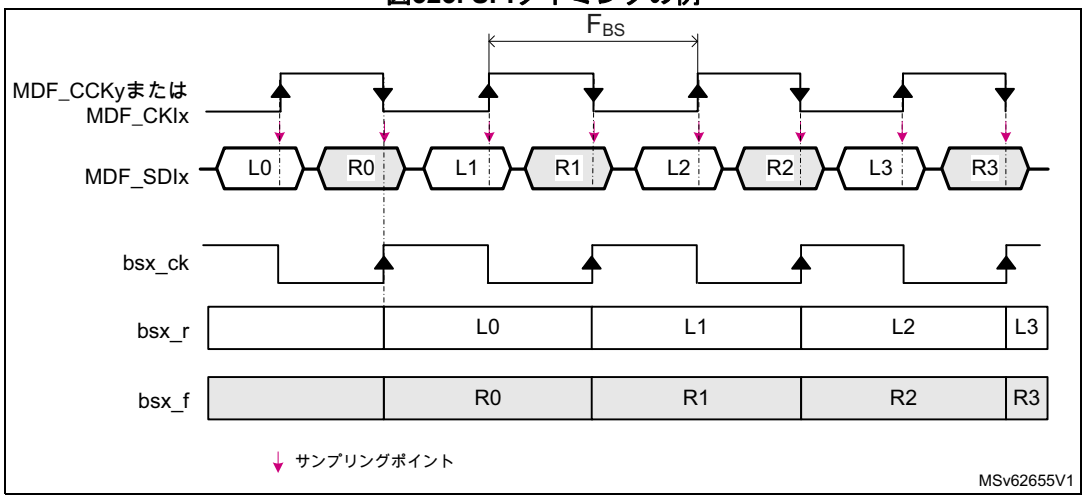
すべてのSPIモードにおいて、選択したクロックの立ち上がりエッジと立ち下がりエッジを使用して、シリアル･データがキャプチャされます。SITFx では常に次のビットストリームが提供されます。

- ビットストリーム･クロックの立ち下がりエッジを使用して受信したビットストリーム（bsx_f）
- ビットストリーム･クロックの立ち上がりエッジを使用して受信したビットストリーム（bsx_r）

接続されているセンサによっては、2つのビットストリームのいずれかが利用できないことがあります。

アプリケーションから、BSMXマトリックスを通して望ましいストリームを選択することができます。

図323. SPIタイミングの例



データ･ストリームを適切に同期／受信するには、表370 に示す制約に従って、mdf_proc_ckクロックの周波数を調整する必要があります。

クロック無検出

シリアル･インタフェースが通常SPIモードで動作している場合、クロック遷移なし期間が検出されることがあります。この機能は、SPIリンクでクロック障害を検出するために使用できます。

アプリケーションから、対応する SITFx のSTH[4:0] ビットフィールドを通してタイムアウト値をプログラムすることができます。STH[4:0] x $T_{mdf_proc_ck}$ の間、MDFによってクロック遷移が検出されない場合、CKABFフラグがセットされます。

CKABIEに1がセットされている場合は、割込みを生成することができます。STH[4:0] ビットフィールドは MDFシリアル･インタフェース制御レジスタx (MDF_SITFxCR)にあります。

シリアル･インタフェースが有効化されている場合、最初のクロック遷移が検出されるまで、CKABFフラグは1のままです。

疑似クロックなし検出を避けるため、次のシーケンスに従う必要があります。

1. シリアル･インタフェースを通常SPIモードで設定し、有効化します。
2. CKABFビットに1を書き込んでCKABFフラグをクリアします。

   シリアル･インタフェースでクロック遷移なしが検出された場合、ハードウェアによりただちにCKABFフラグに1をセットします。
3. CKABFフラグを読み出します。
   - CKABF = 1の場合、手順2に戻ります。
   - CKABF = 0の場合、クロックが検出されています。クロックなし検出時に割込みを発生させるには、CKABIEビットに1をセットします。

注： **クロックなし検出機能は、LF_MASTER SPIモードでは使用できません。**

マンチェスタ･モード

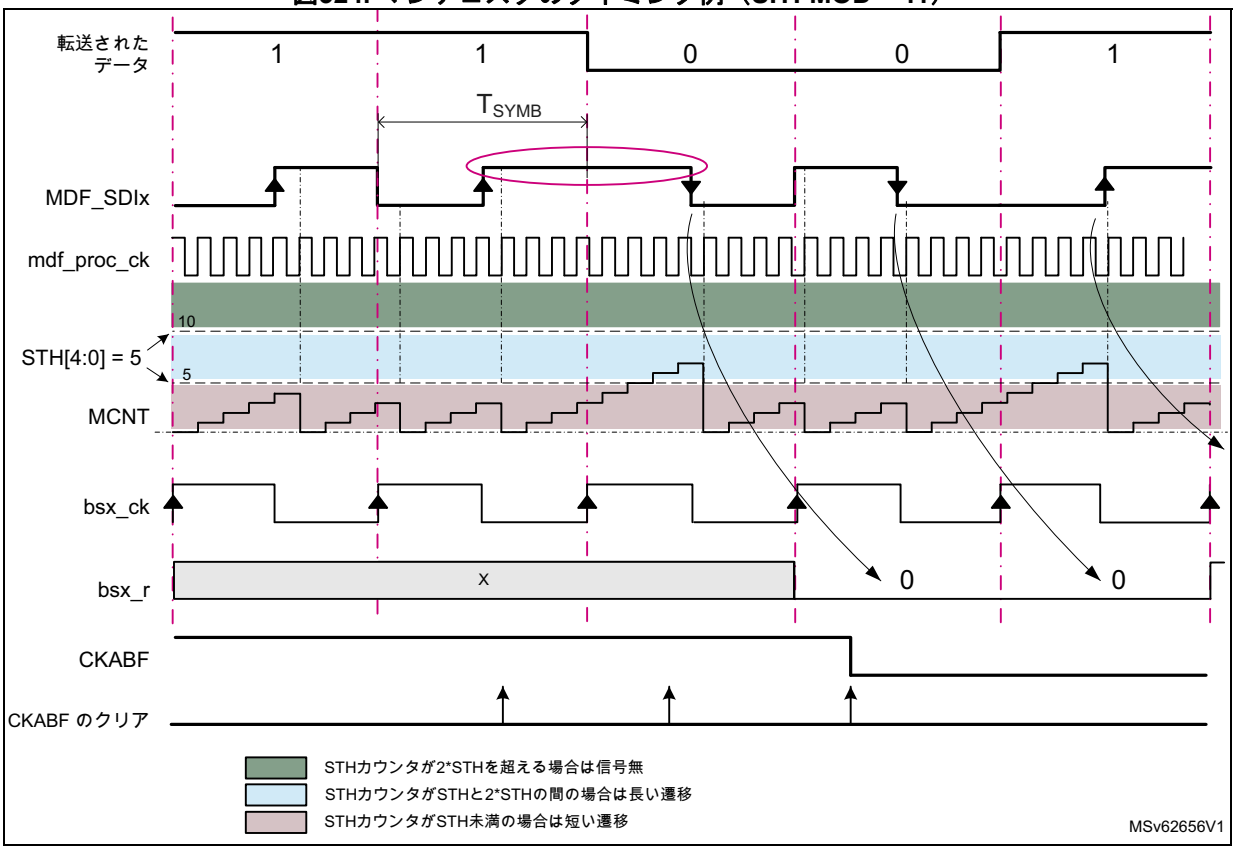
マンチェスタ符号化フォーマットでは、MDFはMDF_SDIx ピンを通してのみ、外部センサからデータ･ストリームを受信します。

このモードでは、MDF_CKIx ピンは不要です。

デコードされたデータおよびクロック信号は、マンチェスタ･デコードの後、シリアル･ストリームから復元されます。これらは、bsx_r上で利用できます。マンチェスタ符号化では、次の2通りの設定ができます。

- 信号立ち上がりエッジを0として符号化し、信号立ち下がりエッジを1として符号化
- 信号立ち上がりエッジを1として符号化し、信号立ち下がりエッジを0として符号化

図324. マンチェスタのタイミング例（SITFMOD = 11）



マンチェスタの受信ストリームをデコードするには、MDFシリアル･インタフェース制御レジスタx (MDF_SITFxCR)でSTH[4:0] ビットフィールドをプログラムする必要があります。マンチェスタ･シンボル長を推定し、クロックなし検出を行うため、STH[4:0] ビットフィールドがSITFx によって使用されます。内部カウンタ（MCNT）は、MDF_SDIx 入力で遷移が検出されるたびに再開されます。これは、短い遷移、長い遷移、クロックなしの検出に使用されます。長い遷移は、データ値が変化したことを示します。図324 には、OVRが高さの周辺でSTH[4:0] = 5であるケースを示します。

マンチェスタの推定シンボルレート（T_{SYMB}）は、次の式に従う必要があります。

$$(STH + 1)\times T_{mdf_proc_ck} < T_{SYMB} < (2 \times STH \times T_{mdf_proc_ck})$$

STHの計算は次のように行うことを推奨します。

$$STH[4:0] = round\left(\frac{(2 \times OVR) - 1}{3}\right)$$

ここで、OVRはmdf_proc_ck周波数と予測されるマンチェスタ･シンボル周波数との比率を示します。OVRは5より大きくする必要があり､mdf_proc_ckクロックは表370 に示す制約に従って調整する必要があります。

2 x STH[4:0] x $T_{mdf_proc_ck}$ より長い間、遷移が検出されない場合、またはSITFx がまだ受信マンチェスタ･ストリームに同期されていない場合、クロックなしフラグCKABFに1がセットされます。また、ビットCKABIEに1がセットされている場合、割込みを生成できます。

シリアル･インタフェースが有効化されている場合、まずMDFを受信マンチェスタス･トリームに同期する必要があります。0から1、または1から0へのデータの遷移（図324 のピンクの楕円）が検出されると、同期は終了します。

同期フェーズの終了は、次のソフトウェア･シーケンスに従って確認できます。

1. CKABFビットに1を書き込んで MDF DFLTx割込みステータス･レジスタx（MDF_DFLTxISR）のCKABFフラグをクリアします。シリアル･インタフェースがまだ同期されていない場合、ハードウェアによりすぐにCKABFフラグに1がセットされます。
2. CKABFフラグを読み出します。
   – CKABF = 1の場合、手順1に戻ります。
   – CKABF = 0の場合、マンチェスタ･インタフェースが同期され、有効なデータが提供されます。

プログラミングの例

次の例では、MDFカーネル･クロック周波数（$F_{mdf_ker_ck}$）は100MHzで、受信マンチェスタ･ストリームは約6MHz（F_{SYMB}）となります。

1. 有効なmdf_proc_ckをSITFx に供給します。

   mdf_proc_ck周波数はマンチェスタ･シンボル周波数の少なくとも6倍（つまり、少なくとも36MHz）にする必要があります。

   カーネル･クロックを2分周するには、PROCDIVを1にプログラムします。その場合、$F_{mdf_proc_ck}$ = 50MHzとなります（マンチェスタ･シンボル周波数の8.33倍）。
2. STHを計算します。

   OVRの計算の式は次のとおりです。$OVR = F_{mdf_proc_ck} / F_{SYMB}$ = 50MHz / 6MHz = 8.33

   そして、次の計算をします。$STH[4:0] = \text{round}\left(\frac{(2 \times 8.33) - 1}{3}\right) = 5$

   マンチェスタ･ストリームの最小許容周波数は、次のようになります。

   $1 / (2 \times STH \times T_{mdf_proc_ck}) = 1 / (10 \times 20ns) = 5MHz$

   マンチェスタ･ストリームの最大許容周波数は、次のようになります。

   $1 / (STH+1 \times T_{mdf_proc_ck} = 1 / (6 \times 20ns) = 8.33MHz$

39.4.4 ADCスレーブ･インタフェース（ADCITF）

ADCは必ずしもMDFに接続されているわけではありません。この製品の状況を確認するには、セクション39.3 を参照してください。

MDFではフィルタ･パスに最大2つのADCを接続できます。それぞれのフィルタについて、MDFデジタル･フィルタ設定レジスタx（MDF_DFLTxCICR）のDATSRC[1:0] ビットフィールドにより、ADCからいずれかのデータを選択することができます。

警告： MDFは、いずれかのADCITF入力からの受信インタリーブ･データには対応していません。

39.4.5 クロック･ジェネレータ（CKGEN）

RCC（リセットおよびクロック･コントローラ）はMDFに次のクロックを供給します。

- レジスタ･インタフェース用のAHBクロック（mdf_hclk）
- 主にCKGEN経由で回路のその他のすべての部分で使用されるカーネル･クロック（mdf_ker_ck）

これらのクロックは非同期クロックになりますので、両方のクロック･ドメインを跨ぐすべての信号は再同期されます。

クロック･ジェネレータ（CKGEN）では処理クロックの生成が行われ、クロックがMDF_CCK0およびMDF_CCK1ピンに供給されます。これらのすべてのクロックはmdf_ker_ckから生成されます。

処理クロック（mdf_proc_ck）は、すべての信号処理の実行と、受信シリアルまたはパラレル･ストリームの再サンプリングのために使用されます。

注： **サンプルを処理するには、再形成フィルタ（RSFLT）で最大24サイクルのmdf_proc_ckクロックが必要です。**

RCCによって供給されるカーネル･クロック周波数を適合させるために、次の分周回路を使用できます。

- PROCDIV[6:0]：カーネル･クロック周波数をパラレルおよびシリアル･インタフェースの制約、および処理ブロックに適合させるために使用します。
- CCKDIV[3:0]：MDF_CCK0およびMDF_CCK1クロックの周波数を適合させるために使用します。

PROCDIV[6:0] およびCCKDIV[3:0]：分周回路にクロックが供給されていない場合にプログラムする必要があります（CKGDEN = 0）。

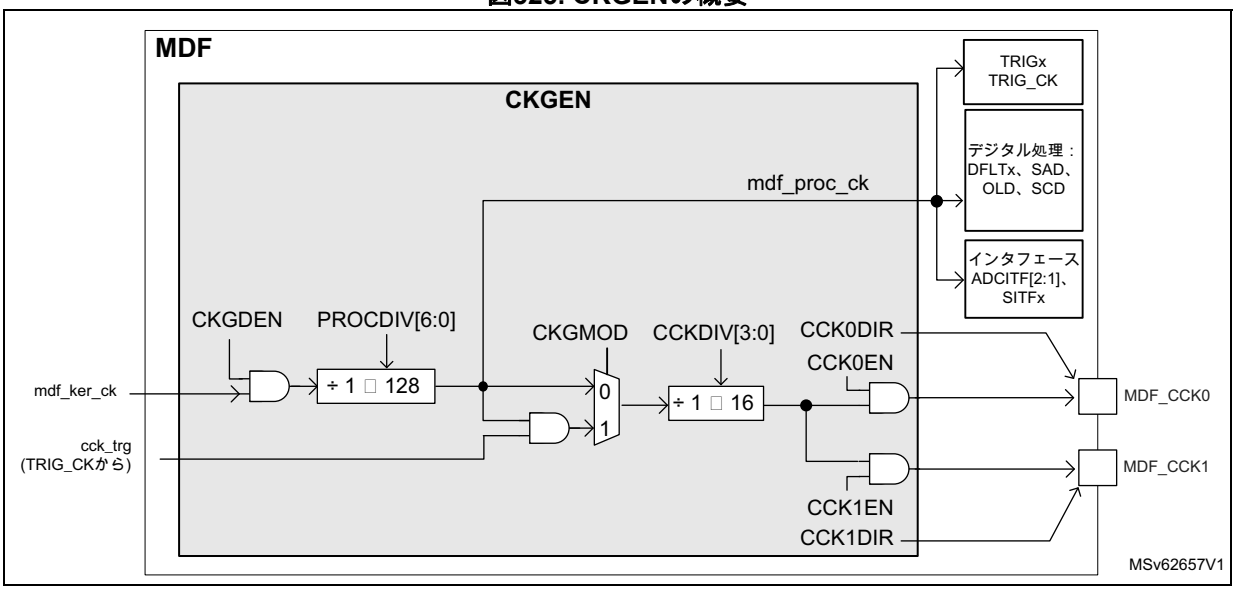
mdf_proc_ckの生成はCKGDENによって制御されます。

また、CKGMODビットにより、CCKDIV分周回路のトリガ方法を定義できます。

- CKGMOD = 0の場合、CCKDIV分周回路はCKGDENに1がセットされるとすぐに開始されます。
- CKGMOD = 1の場合、CCKDIV分周回路はCKGDENに1がセットされ、プログラムされたトリガ条件が発生すると開始されます。

CKGENを制御するすべてのビットとフィールドはMDF_CKGCRにあります。

図325. CKGENの概要



CKGENのトリガ･ロジックはTRG_CKブロックにより処理されます。図334 に示すように、CCKDIV分周回路は16個のいずれかのトリガ･ソースの立ち上がりエッジまたは立ち下がりエッジでトリガできます。適切なトリガ条件が発生すると、cck_trg信号が高くなり、CCKDIV分周回路が始動します。CKGDENに0がセットされると、TRG_CKロジックはリセットされます。

この機能は、いくつかのMDFインスタンスのMDF_CCKy（y = 0、1）クロックを同期したり、クロック生成をタイマ･イベントと同期したりする上で有益です。

アプリケーションにより、MDF_CCK0 または MDF_CCK1 ピンの CCK0EN/CCK1EN および CCK0DIR/CCK1DIRビットの有効化を制御できます。

- CCKyENは、CCKDIVを有効化し、外部センサ用のクロックを生成するために使用されます。
- CCKyDIRは、ADF_CCKyピンの方向（入力または出力）を制御するために使用されます。

表369.共通クロック生成の制御(1)

CCKyEN	CCKyDIR	説明
0	0	MDF_CCKyピンは入力にあります。シリアル･ストリームをデコードするために、外部クロックをMDF_CCKyピンに接続し、SITFx で使用することができます。
0	1	MDF_CCKyピンは出力にあります。クロックは生成されないため、MDF_CCKyピンはローに駆動されます。
1	1	MDF_CCKyピンは出力にあります。クロックはMDF_CCKyピンで生成されます。シリアル･ストリームをデコードするために、SITFx ではこのピンをクロック･ソースとして使用することができます。

1. CCKyEN = 1かつCCKyDIR = 0という設定は避ける必要があります（無意味です）。

CCKyDIR = 1の場合、CCKyENビットに1がセットされるとすぐに、スプールを生成することなく対応する出力にクロックが生成されます。

注： **他のブロック（SITFx、DFLTx など）を有効化する前に、mdf_proc_ckを（CKGDEN = 1により）有効化する必要があります。**

CKGENの有効化シーケンスの例

1. CKGDENを0にセットします。
2. CKGACTIVE = 0を待ちます。CKGDENが前に有効化されている場合このフェーズには2周期のmdf_hclkと2周期のmdf_proc_ckが必要となることがあります。
3. PROCDIV[6:0]、CKGMOD、CCKDIV[3:0]、TRGSRC[3:0]、およびTRGSENSをプログラムします。
4. CKGDENを1にセットします。
5. CCKxDIRに1をセットします（オプション）。
6. CCKxENに1をセットします（オプション）。

必要に応じて、クロック･ジェネレータを無効にすることなく、いつでもCCK0ENまたはCCK1ENの値を変更することができます。

クロック周波数の制約

次の表に、サンプルを適切に受信し、処理するための周波数の制約を示します。

表370.受信ストリームに関するクロックの制約(1)

SITFxモード	MDFクロックの制約	
	RSFLT無効時	RSFLT有効時
LF_MASTER SPI	F_{MDF_CCKy} 最大周波数は5Mhzに制限	
	$F_{mdf_proc_ck} > 2 * F_{MDF_CCKy}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$	$F_{mdf_proc_ck} > 24 * F_{MDF_CCKy} / (MCICD+1)$ および $F_{mdf_proc_ck} > 2 * F_{MDF_CCKy}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$
マスタSPI スレーブSPI	F_{MDF_CKx} 最大周波数は25Mhzに制限	
	$F_{mdf_proc_ck} > 4 * F_{MDF_CCKy}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$	$F_{mdf_proc_ck} > 24 * F_{MDF_CKKy} / (MCICD+1)$ および $F_{mdf_proc_ck} > 4 \times F_{MDF_CKKy}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$
マンチェスタ	F_{SYMB} 最大周波数は20Mhzに制限	
	$F_{mdf_proc_ck} > 6 * F_{SYMB}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$	$F_{mdf_proc_ck} > 24 * F_{MDF_CKKy} / (MCICD+1)$ および $F_{mdf_proc_ck} > 6 * F_{SYMB}$ および $F_{mdf_hclk} \geq F_{mdf_proc_ck}$

1. F_{MDF_CKKy}は、MDF_CKIxおよびMDF_CCKyを通して受信した、またはMDF_CCKyを通して生成されたクロックの周波数を示します。F_{SYMB}はマンチェスタ･モード用に受信したシンボル･レートの周波数を示します。

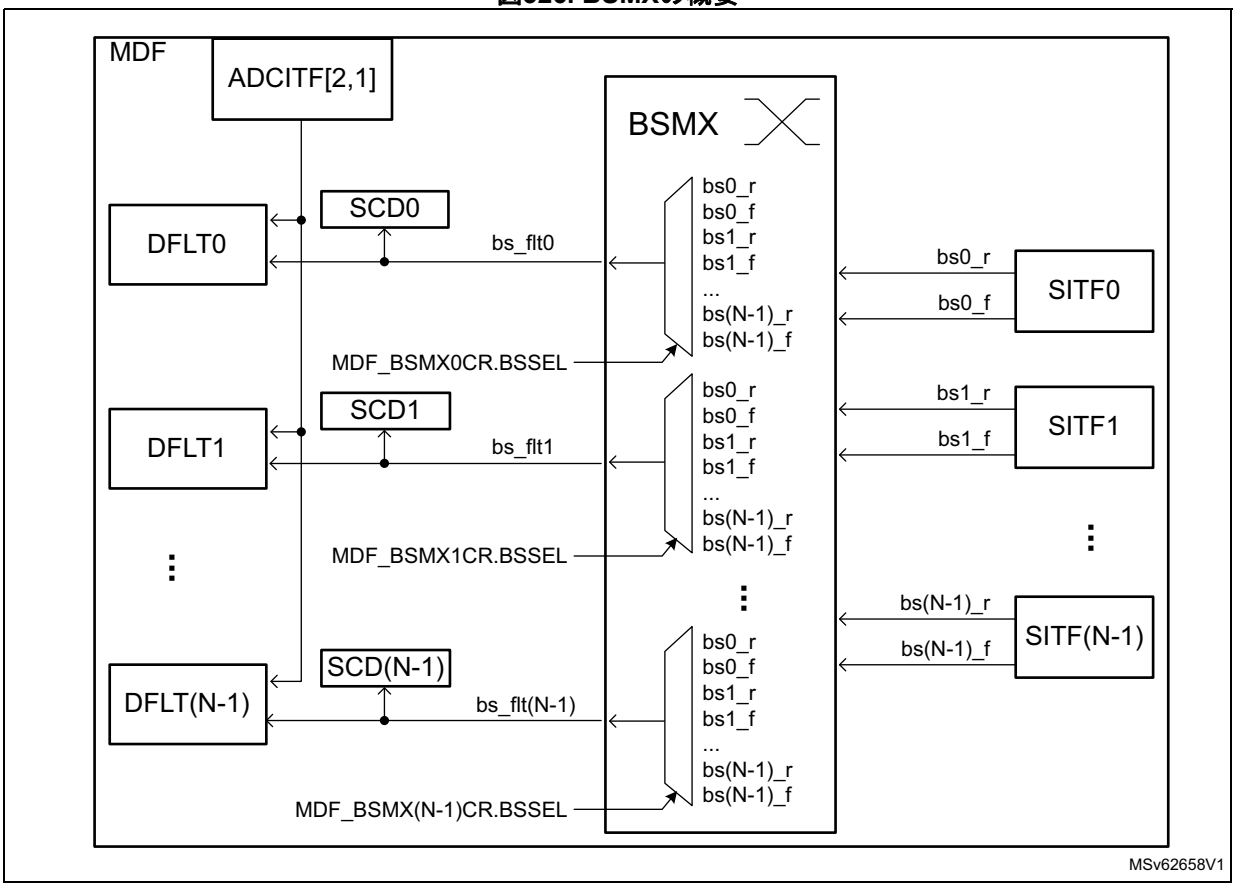
39.4.6 ビットストリーム･マトリックス（BSMX）

BSMXはすべての シリアル･インタフェース（SITFx）からビットストリームを受信し、選択された入力をデジタル･フィルタ（DFLTx）と短絡検出回路 (SCDx）に提供します。各フィルタ･パスについて、任意のビットマップ入力を選択できます。

図322 に示すように、各 SITFx ブロックはBSMXに2つのビットストリーム（bsx_rとbsx_f）を提供します。

アプリケーションは MDFビットストリーム･マトリックス制御レジスタx（MDF_BSMXxCR）を通して望ましいストリームを選択します。この選択は静的なものです。

図326. BSMXの概要



BSMXのプログラミングのシーケンス例

BSSEL[4:0] ビットフィールドは、対応する SCDx、OLDx、またはDFLTx が有効化されている場合は変更できません。フィルタ･パスxについて、BSMXの値を変更するには次の手順を実行する必要があります。

1. SCDx のSCDENに0をセットします。
2. DFLTx のDFLTENに0をセットします。
3. OLDx のOLDENに0をセットします。
4. BSMXACTIVE = 0を待ちます。
5. フィルタ･パスx のBSSEL[4:0] をプログラムします。
6. SCDx のSCDENに1をセットします（オプション）。

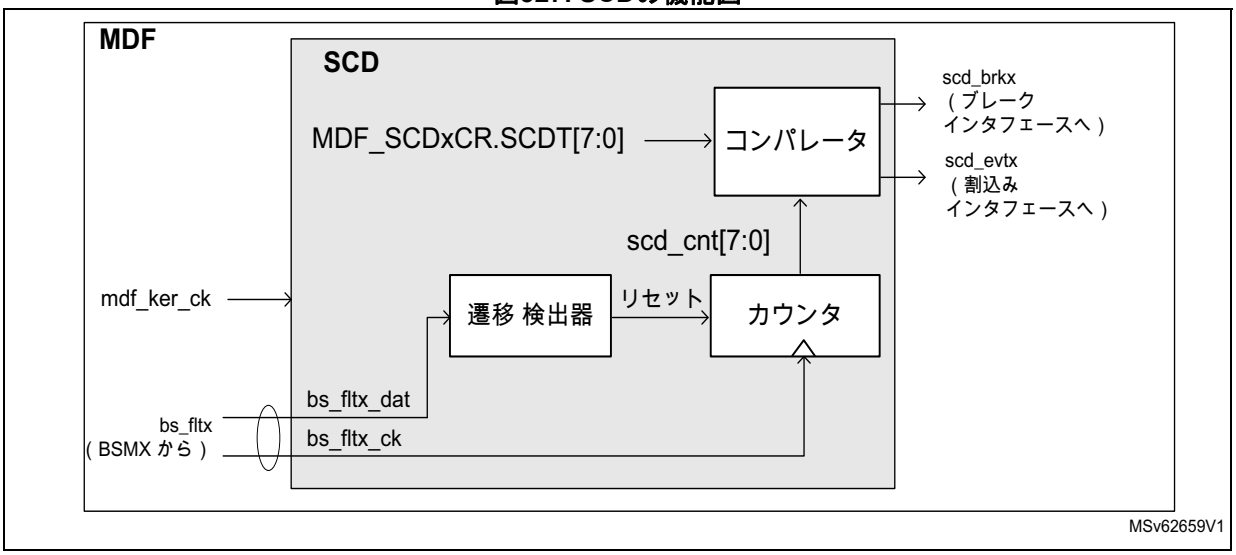
7. DFLTx のDFLTENに1をセットします（オプション）。
8. OLDx ブロックのOLDENに1をセットします（オプション）。

39.4.7 短絡検出回路（SCD）

SCDx は、アナログ信号が飽和した値（最大測定範囲外）に達してこの値が一定時間残ったかどうかを、非常に短い応答時間で検出します。

この動作により、短絡または開回路エラー（過電流、過電圧など）を検出できます。割込みイベントやブレーク信号を生成できます。

図327. SCDの機能図

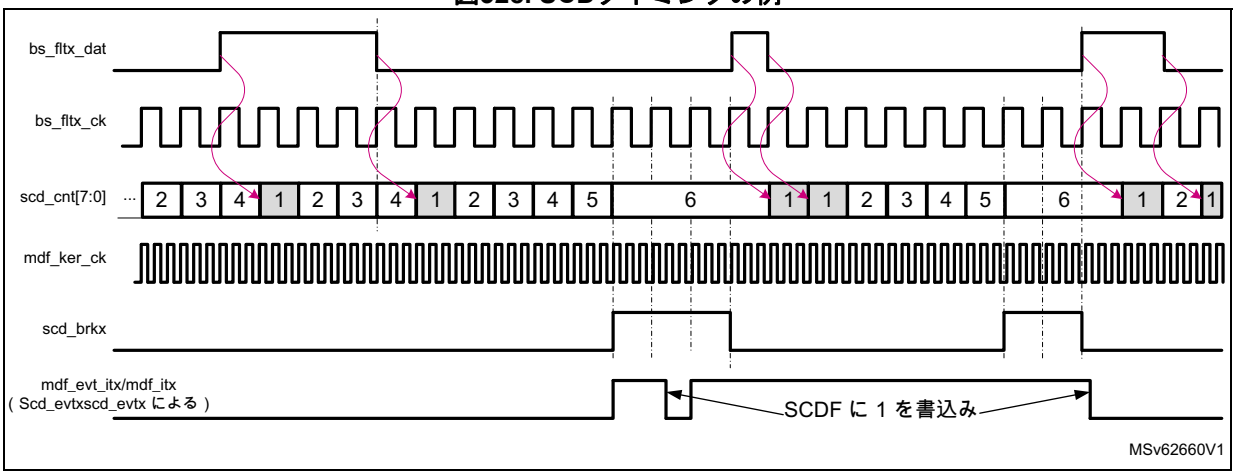


SCDx は、シリアル･インタフェース（SITFx）から受信した連続する0または連続する1をカウントしています。受信したデータ･ストリームが変化した場合、カウンタは再起動されます。このカウンタが短絡閾値（MDF短絡検出回路の制御レジスタx（MDF_SCDxCR）のSCDT[7:0]）に達すると、短絡イベントが呼び出されます。各BSMX出力には、独自の短絡検出回路があります。BSMX出力を継続的に監視するには、MDF_SCDxCR で対応する SCDENビットに1をセットします。各SCDには独自の閾値設定（SCDT）と独自のステータス･ビット（SCDF）があります。

次の図に、SCDT[7:0] に6がセットされている例を示します。短絡状態が存在する限り、ブレーク信号はハイのままです。

新しい短絡状態が検出された状態で割込みイベントが保留になっている場合、オーバーラン･イベントは生成されません。

図328. SCDタイミングの例



SCDx ブロックによって生成されたSCDx イベントは、ブレーク出力信号mdf_break[3:0] に割り当てることができます。特定の短絡検出回路イベントへのブレーク信号の割当ては、MDF_SCDxCRのBKSCD[3:0] によって行えます。ブレーク出力は過電流機能と共有されます。

注： **SCDは、ADCデータ･インタフェース（ADCITF）の監視には使用できません。**

SCDの有効化シーケンスの例

1. CKGENを有効化して設定し、mdf_proc_ckを生成します。
2. SCDENを0にセットします。
3. SCDACTIVE = 0を待ちます。SCDENが前に有効化されている場合、このフェーズには2周期のmdf_hclkと2周期のmdf_proc_ckが必要となることがあります。
4. BKSCD[3:0] とSCDT[7:0] をプログラムします。
5. SCDENを1にセットします。

注： **SCDACTIVE = 1の場合、BKSCD[3:0] とSCDT[7:0] は変更しないでください。**

39.4.8 デジタル･フィルタ処理（DFLT）

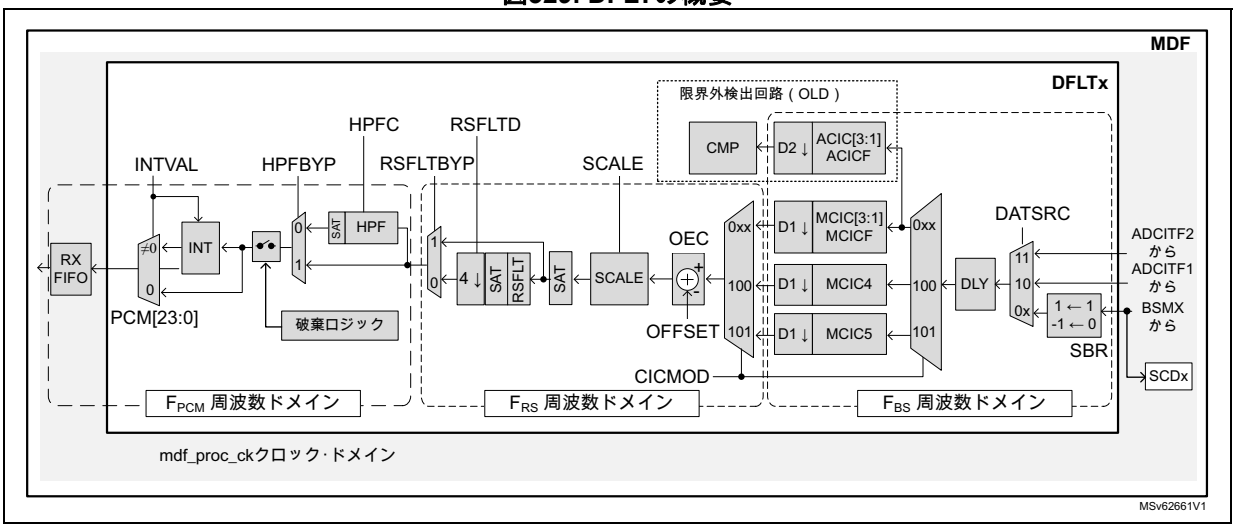
デジタル･フィルタ処理には、次のサブブロックが含まれます。

- シンボル･リマップ（SBR）
- ソース･セレクタ
- クロック･スキッパ遅延（DLY）
- 以下で設定可能なCICデシメーション･フィルタ：
  - 単一のSinc^4 またはSinc^5 の次数CIC（MCIC）
  - 2つのCICフィルタ：
    - 高分解能のメイン･フィルタ（MCIC）
    - 限界外検出回路（OLD）用の低遅延の補助フィルタ（ACIC）
    - MCICとACICの両方をSinc^1、Sinc^2、Sinc^3、またはFastSincとして設定可能
- ゲイン･コントロール（SCALE）
- 信号飽和（SAT）
- 再形成フィルタ（RSFLT）
- ハイパス･フィルタ（HPF）

- 積分器（INT）
- 受信RXFIFO

図329 に、CICMOD[2:0] に従ったフィルタ･パス設定を示します。アプリケーションのニーズに応じて、いくつかの設定ビットを利用してデジタル･フィルタを設定することができます。

図329. DFLTの概要



シンボル･リマップとソース選択

シンボル･リマップ（SBR）では、BSMXによって選択されたビットストリームをフィルタ･パスで使用可能なデータに変換します。具体的に言うと、次のようになります。

- 上位レベルは16bitの符号付き数字 + 1に変換されます。
- 下位レベルは16bitの符号付き数字 - 1に変換されます。

デジタル･フィルタの信号ソースは、以下に従って、2つの間でDATSRC[1:0] を通して選択できます。

- BSMXからのデータ
- ADCインタフェースのどちらか（ADCITF2または1）からのデータ

プログラム可能なマイクロ遅延制御（DLY）

デジタル･フィルタは、ビットストリーム･クロックの分解能により各ストリームのタイミング調整を行える遅延ラインを備えています。

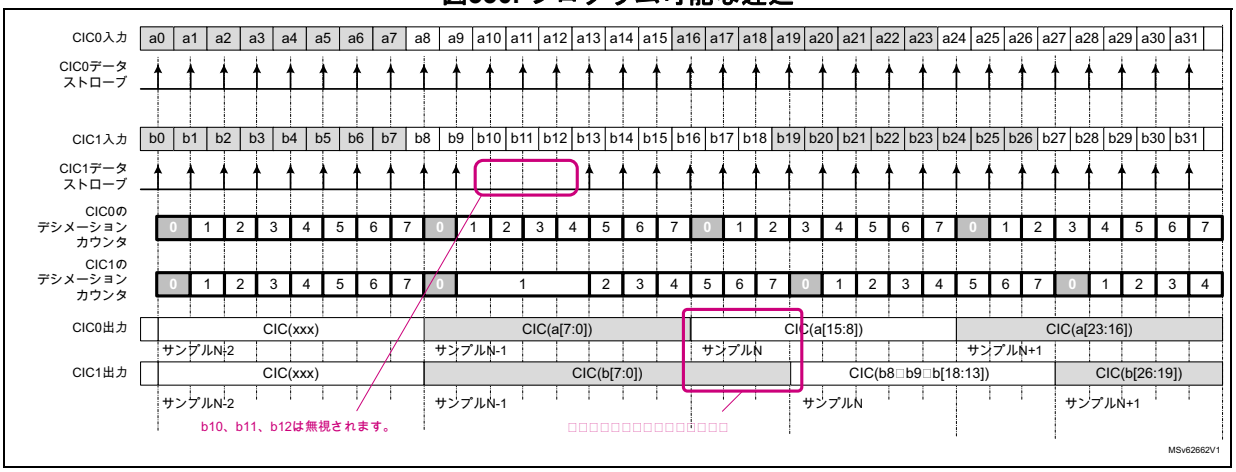
この機能は、マイクロフォンのビーム形成アプリケーションで、受信ストリームに最終的なサンプリング･レートよりも小さな遅延を適用する必要がある場合に特に有益です。この機能は、たとえば、MDFがビーム･フォーミング･アプリケーション用の他のADFまたはMDFブロックと同期される場合に有益です。

遅延は、サンプルがCICフィルタに入る前に、選択した入力ストリームから所定の数のサンプルを破棄することによって実行されます。このデータ破棄は、所定の数のデータ･ストローブをスキップし、CICフィルタでこれらのデータが処理されないようにすることで実行されます。

指定した量のデータ･ストローブがスキップされると、次の受信サンプルは正常にストローブ処理されます。

図330 に、受信ストリームに動的に小さな遅延を適用する方法の例を示します。簡単に言うと、この例では、CICフィルタにより高さによるデシメーションが実行されます。CIC1はADFに含まれるCICで、CIC0は別のADFまたはMDFインスタンスのフィルタです。

図330. プログラム可能な遅延



フィルタ1のCIC（CIC1）は、3つの受信サンプルをスキップするためのコマンドを受信します。そのため、入力サンプルb10、b11、およびb12はCIC1によって処理されません。その結果、CIC0によって生成された出力サンプルN+1は入力サンプルa[23:16] から構築され、CIC1のサンプルN+1は入力サンプルb[26:19] から構築されます。

最後に、スキップされないデータ･ストリームは、ビットストリームの3周期分遅延しているように見えます。

注： **入力データ･ストローブがスキップされるき、デシメーション･カウンタはフリーズしたままとなります。その結果、CIC1によって提供されるサンプルは少し遅延します。**

警告： 特にインタリーブ取得モードでMDFがプログラムされている場合、プログラムされたデシメーション比（CIC + RSFLTデシメーション）より大きな遅延を適用することは推奨できません。FIFO内でデータのミスアライメントが発生するリスクが高まります。

スキップするビットストリーム･クロックの周期の数をプログラムするには、次の手順を実行する必要があります。

1. SKPBFが0になるのを待ちます。
2. SKPDLY[6:0] に、スキップさせるビットストリーム･クロックの周期の数を書き込みます。SKPBFフラグがすぐに1になり、SKPDLY[6:0] に入力された遅延値が処理中であることが示されます。
   - 対応する DFLTx がまだ有効になっていない（DFLTEN = 0）場合、DLYロジックはDFLTEN = 1になるのを待ちます。アプリケーションによりDFLTENに1がセットされると、DLYロジックは必要な量のデータ･ストローブのスキップを開始します。
   - 対応するデジタル･フィルタがすでに有効になっている（DFLTEN = 1）場合、DLYロジックはすぐに必要な量のデータ･ストローブのスキップを開始します。

   MDFで必要な量のデータ･ストローブのスキップが行われると、SKPBFは0に戻ります。
3. アプリケーションでさらにデータ･ストローブのスキップが必要な場合、ステップ 1 から操作を繰り返す必要があります。

このメカニズムで実行された遅延の影響は、MDFが有効である限り、蓄積されます。所定のフィルタで、アプリケーションによりD1遅延が実行され、続いてD2遅延が実行された場合、それ以外のすべてのアクティブなフィルタはD1 + D2の時間だけ遅延します。

ADCから着信するデータについては、この機能も利用できます。

インタリーブ取得モードでは、アプリケーションにより、別のマイクロフォンに適用された遅延がRXFIFOバッファの深さに適合していることが保証されます。有効化されたフィルタ間の相対遅延がデシメーション比よりも小さい場合、1つ分のFIFO場所が使用されます。

インタリーブ取得モードを使用しない場合、遅延値はRXFIFOバッファリングに影響しません。

注： **SKPBF = 1の状態でSKPDLY[6:0] に書き込んだ場合、書込み操作は無視されます。**

Cascaded-integrator-comb（CIC）フィルタ

CICデジタル･フィルタはローパス･フィルタを効率的に実装したもので、デシメーションや内挿によく使用されます。CIC周波数応答は$Sinc^N$ 関数に等しいため、Sincフィルタとも呼ばれます。

MDFに組み込まれた$Sinc^N$ デジタル･フィルタは、対象となるアプリケーションに応じて設定できます。

- デジタル･マイクロフォンからのスピーチ･キャプチャのようなオーディオ･アプリケーションの場合、CICMODに100または101をセットすることにより、高分解能のローパス･デシメーション･フィルタを選択することができます。
- 対象となるアプリケーションがモータ制御または他のセンサ･キャプチャの場合、アプリケーションでCICを設定して、取得用のメイン･フィルタ･パス（MCIC）と高速イベント生成用の補助フィルタ（ACIC）を提供することができます。補助フィルタは、たとえば、過電流状態の検出に使用できます。このモードは、CICMOD[2:0] = 0xxをセットすることによって選択されます。

CICMOD[2:0] = 0xxの場合、次のCICフィルタが利用できます。

- メイン･フィルタ（MCIC）
- 補助CICフィルタ（ACIC）

どちらのフィルタもFastSinc、$Sinc^1$ ~ $Sinc^3$ で設定できます。

CICMOD[2:0] = 100の場合、CICは単一の$Sinc^4$ に設定され、CICMOD[2:0] = 101の場合、CICは単一の$Sinc^5$ フィルタに設定されます。

フィルタには次の転送機能（zドメインのインパルス応答）があります。

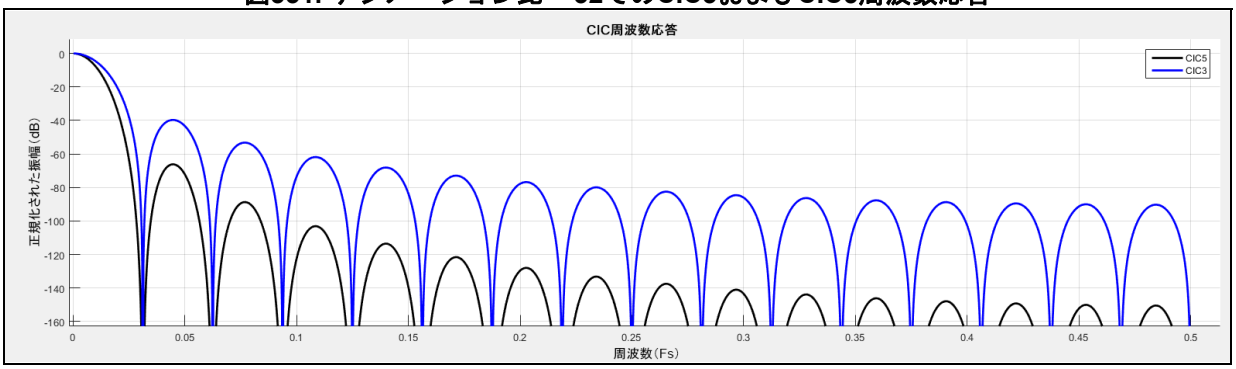
- $Sinc^N$ フィルタ･タイプ：
$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-1}}\right)^N$$

- FastSincフィルタタイプ：
$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot D)})$$

ここで、Nは1、2、3、4、または5とすることができ、Dはデシメーション比です。

DはMCICD+1またはACICD+1と等しくなります。

図331. デシメーション比 = 32でのCIC3およびCIC5周波数応答



CIC の出力データ･サイズ

CICによって提供されるサンプルのサイズ（DS_{CIC}）は、次のパラメータによって決まります。

- CICの次数（N）
- CICのデシメーション比（D）
- 入力ストリームのデータ･サイズ（DSIN）

データ･サイズが26bitのCICの容量を超えないことを保証するため、CICの次数とデシメーション比をプログラムする必要があります。

次の式では、上記のパラメータに従って出力データ･サイズが算出できます（DS_{CIC}）。

$$DS_{CIC} = \left(\frac{N \times Ln(D)}{Ln(2)}\right) + DS_{IN}$$

また、CICゲインは次の式で算出できます。

$$G_{CIC} = (D)^{N}$$

デシメーション比は、メインCICフィルタでは2 ~ 512の範囲で調整でき、補助CICフィルタでは1 ~ 32の範囲で調整できます。

次の表に、データ･ソースがシリアル･インタフェースまたは12bit ADCからの最大測定範囲の信号である場合の、一部のデシメーション値の一部のデータ出力サイズ（ビット単位）を示します。

注： **シリアル･ビットストリームではDS_{IN} = 1bitですが、ADCITFから受信する場合、最大16bitとすることができます。**

表371.CICの次数とCICのデシメーション値に従ったデータ･サイズ

デシメーション	DS_{IN} = 1bitの場合のデータ･サイズ（ビット）（SITFx からのデータ）						DS_{IN} = 12bitの場合のデータ･サイズ（ビット）（ADCITFからのデータ）					
	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
4	3	5	6	7	9	11	14	16	17	18	20	22
8	4	7	8	10	13	16	15	18	19	21	24	-
12	5	9	9	12	16	19	16	20	21	23	-	-
16	5	9	10	13	17	21	16	20	21	24	-	-
24	6	11	12	15	20	24	17	22	23	-	-	-
32	6	11	12	16	21	26	17	22	23	-	-	-
64	7	13	14	19	25	-	18	24	25	-	-	-

表371.CICの次数とCICのデシメーション値に従ったデータ･サイズ

デシメーション	DS_{IN} = 1bitの場合のデータ･サイズ（ビット）（SITFx からのデータ）						DS_{IN} = 12bitの場合のデータ･サイズ（ビット）（ADCITFからのデータ）					
	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
128	8	15	16	22	-	-	19	26	-	-	-	-
256	9	17	18	25	-	-	20	-	-	-	-	-
512	10	19	20	-	-	-	21	-	-	-	-	-

CICによって提供されるデータのLSB部分は、必ずしも重要ではありません。これはセンサのパフォーマンスと、帯域外のノイズを除去するCICの能力によって変わります。

CIC出力でのサンプル･サイズは、SCALEブロックを使用して調整できます。

次の表に、入力データ･サイズによって決まる、CICフィルタのための最大許容デシメーション比を示します。デシメーション比を大きくすると、強い入力信号に対して、CIC出力での信号のラップアラウンドが発生します。

注： **MDFではCICのラップアラウンドの検出や防止はできません。**

表372.最大デシメーション比と次数、入力データ･サイズ

フィルタ段数	DS_{IN} = 1bit時の最大デシメーション比（SITFx）	DS_{IN} = 12bit時の最大デシメーション比（ADCITF）	DS_{IN} = 16bit時の最大デシメーション比（ADCITF）
$Sinc^1$	512	512	512
$Sinc^2$	512	128	32
FastSinc	512	90	22
$Sinc^3$	322	25	10
$Sinc^4$	76	11	5
$Sinc^5$	32	6	4

スケーリング（SCALE）と飽和（SAT）

SCALEブロックを使用すると、CICによって提供される信号の振幅を、アプリケーションにより3dB（± 0.5dB）刻みで調整することができます。

信号の振幅は、最大で8bit（- 48.2dB）減らし、最大で12bit（+ 72.2dB）増やすことができます。

ゲインは、MDFデジタル･フィルタ設定レジスタx（MDF_DFLTxCICR）のSCALE[5:0] ビットフィールドにより調整されます。

SCALE[5:0] は、対応する DLFTx が有効化されている場合でも変更できます。ゲインの遷移中、フィルタによって提供される信号は中断されます。

内部的な再同期により、アプリケーションで新しいゲインを書き込んでから、ゲインが効果的にサンプルに適合されるまでの間に、mdf_proc_ckクロックの一部のサイクルで遅延が発生します。前の値がまだ適用されていないうちにアプリケーションで新しいゲイン値を書き込もうとした場合、この新しいゲイン値は無視されます。SCALE[5:0] を読み戻すと、アプリケーションに現在のゲイン値が通知されます。

次の表に、使用可能なゲイン値を示します。

表373.使用可能なゲイン値

SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）
0x20	- 48.2	0x2B	- 14.5	0x06	+ 18.1	0x11	+ 51.7
0x21	- 44.6	0x2C	- 12.0	0x07	+ 21.6	0x12	+ 54.2
0x22	- 42.1	0x2D	- 8.5	0x08	+ 24.1	0x13	+ 57.7
0x23	- 38.6	0x2E	- 6.0	0x09	+ 27.6	0x14	+ 60.2
0x24	- 36.1	0x2F	- 2.5	0x0A	+ 30.1	0x15	+ 63.7
0x25	- 32.6	0x00	0.0	0x0B	+ 33.6	0x16	+ 66.2
0x26	- 30.1	0x01	+ 3.5	0x0C	+ 36.1	0x17	+ 69.7
0x27	- 26.6	0x02	+ 6.0	0x0D	+ 39.6	0x18	+ 72.2
0x28	- 24.1	0x03	+ 9.5	0x0E	+ 42.1	-	-
0x29	- 20.6	0x04	+ 12.0	0x0F	+ 45.7	-	-
0x2A	- 18.1	0x05	+ 15.6	0x10	+ 48.2	-	-

SATブロックは、コードが最大値を下回るか、最小値を下回った場合に、バイナリ･コードのラップアラウンドが発生しないようにできます。

MDFは、次のレベルで飽和操作を実行します。

- SCALEブロックの後（SATブロックにより実行）。信号は24bitで飽和します。
- RSFLT内。フィルタの正常な動作を保証します。
- HPFの出力。出力信号が24bitを超えないことを保証します。

SATFビットはアプリケーションに、SCALEの後、RSFLTの内部、またはHPFの後に飽和が発生したことを通知します。また、SATIEに1がセットされている場合、割込みを生成できます。飽和が検出されるとすぐに、SATFフラグに1がセットされます。新しい飽和を検出できるようにするには、アプリケーションでこのフラグをクリアする必要があります。

これらのビットは MDF DFLTx割込み有効化レジスタx（MDF_DFLTxIER）と MDF DFLTx割込みステータス･レジスタx（MDF_DFLTxISR）にあります。

ゲイン調整ポリシー

MDFの最高のパフォーマンスを得るには、SCALE[5:0] を通してゲイン値を適切に調整することが重要です。

通常、ゲインを調整するには、最大入力信号については、最終的な信号振幅が24bitの最大測定範囲にできるだけ近くなるようなSCALE[5:0] の値を選択します。

最適なゲインを選択するための方法は、次のとおりです。

1. 予想される入力信号について、CICフィルタへのデータ･サイズが26bitを超えないことを確認してください。確認は、次の式でできます。

$$\frac{LN(SIN_{pp} \cdot D^N)}{LN(2)} < 26$$

ここで、NはCICの次数、Dはデシメーション比、SIN_{pp} は入力信号のピークツーピークの最大振幅です。

SIN_{pp} は次の値とすることができます。

– SITFから受信するサンプルの場合、ピークツーピークの最大振幅は2（± 1）

– 12bit ADCから受信するサンプルの場合、ピークツーピークの最大振幅は4095（+ 2047、- 2048）

例：最大入力信号が ± 0.35を超えない場合、$Sinc^4$ ではデシメーション比を96とすることができます。実際には次のようになります。

$$\frac{LN(0.7 \cdot 96^4)}{LN(2)} \sim 25.82 \text{ bits} < 26 \text{ ビット}$$

2. SCALE値を調整します。

適切なSCALE値を選択するには、RSFLTが使用されているかどうかを確認する必要があります。RSFLTが使用されている場合、SCALE出力でのデータ･サイズは22bitを超えないようにする必要があり、使用されていない場合は最大で24bitとすることができます。

次の式を使用して、dB単位のSCALE値を選択します。

$$SCALE_{dB} < 20 \cdot \log 10\left(\frac{2^{NB}}{SIN_{pp} \cdot D^N}\right)$$

ここで、RSFLTが有効化されている場合、NBは22になり、RSFLTがバイパスされている場合は24になります。$SCALE_{dB}$ は、SCALE[5:0] によって選択されたゲイン値です。

例：デシメーション比が96、SIN_{pp} が0.7の$Sinc^4$の場合、次のようになります。

– RSFLTがバイパスされている場合：

$$20 \cdot \log 10\left(\frac{2^{24}}{0.7 \cdot 96^4}\right) \sim -11 \text{ dB}$$

$SCALE_{dB}$ 値は - 11dB未満でなければなりません。それより小さくて最も近い値は - 12dB（SCALE[5:0] = 0x2C）です。

– RSFLTが有効な場合：

$$20 \cdot \log 10\left(\frac{2^{22}}{0.7 \cdot 96^4}\right) \sim -23 \text{ dB}$$

$SCALE_{dB}$ 値は - 23dB未満でなければなりません。それより小さくて最も近い値は - 24.1dB（SCALE[5:0] = 0x28）です。

SCALE[5:0] にセットされた値がそれより大きい場合、飽和が発生する可能性があります。イベント･フラグにより、飽和が発生しているかどうかが通知されます。

次の表に、データをSITFx から受信した場合の、CICの次数とCICのデシメーション比に従った、フィルタ設定ごとのゲイン値を示します。この表は包括的なものではなく、最大測定範囲の入力信号を考慮したものです（詳細はセクション39.7.5を参照）。

表374.推奨の最大ゲイン値とCICのデシメーション比

CICのデシメーション比	SITF + CICx + RSFLT（+ HPF）構成のゲイン設定（dB）					SITF + CICx（+ HPF）構成のゲイン設定（dB）				
	CIC5	CIC4	CIC3	CIC2	CIC1	CIC5	CIC4	CIC3	CIC2	CIC1
8	33.6	51.7	69.7	72.2	72.2	45.7	63.7	72.2	72.2	72.2
12	18.1	39.6	60.2			30.1	51.7			
16	3.5	27.6	51.7			15.6	39.6	63.7		
20	- 6.0	21.6	48.2			6.0	33.6	60.2		
24	- 12.0	15.6	42.1	69.7		0	27.6	54.2		
28	- 20.6	9.5	36.1	66.2		- 8.5	21.6	48.2		
32	-26.6	3.5	33.6	63.7		- 14.5	15.6	45.7		
48	-	- 8.5	24.1	57.7		-	3.5	31.6	69.7	
64	-	- 20.6	15.6	51.7		-	-8.5	27.6	63.7	
128	-	-	- 2.5	39.6		-	-	9.5	51.7	
256	-	-	- 20.6	27.6		-	-	-8.5	39.6	

再形成フィルタ（RSFLT）

MDFは、CICを備えているほか、主にオーディオアプリケーション用で他の用途にも使用できる再形成IIRフィルタも備えています。

RSFLTを使用する場合、入力でのサンプル･サイズは22bitを超えないようにする必要があります。

RSFLT出力でのサンプルは、MDF再形成フィルタ設定レジスタx（MDF_DFLTxRSFR）のRSFLTDビットに従って、4でデシメーションするかどうかを決定することができます。

RSFLTをバイパスするには、MDF_DFLTxRSFRでRSFBYPに1をセットします。

次の表に、一般的なオーディオ･ストリームを処理するためにRSFLTに必要なサンプリング･レートを示します。

RSFLT遮断周波数（F_C）は、入力でのサンプル･レート（F_{RS}）によって決まり、次の式で算出されます。

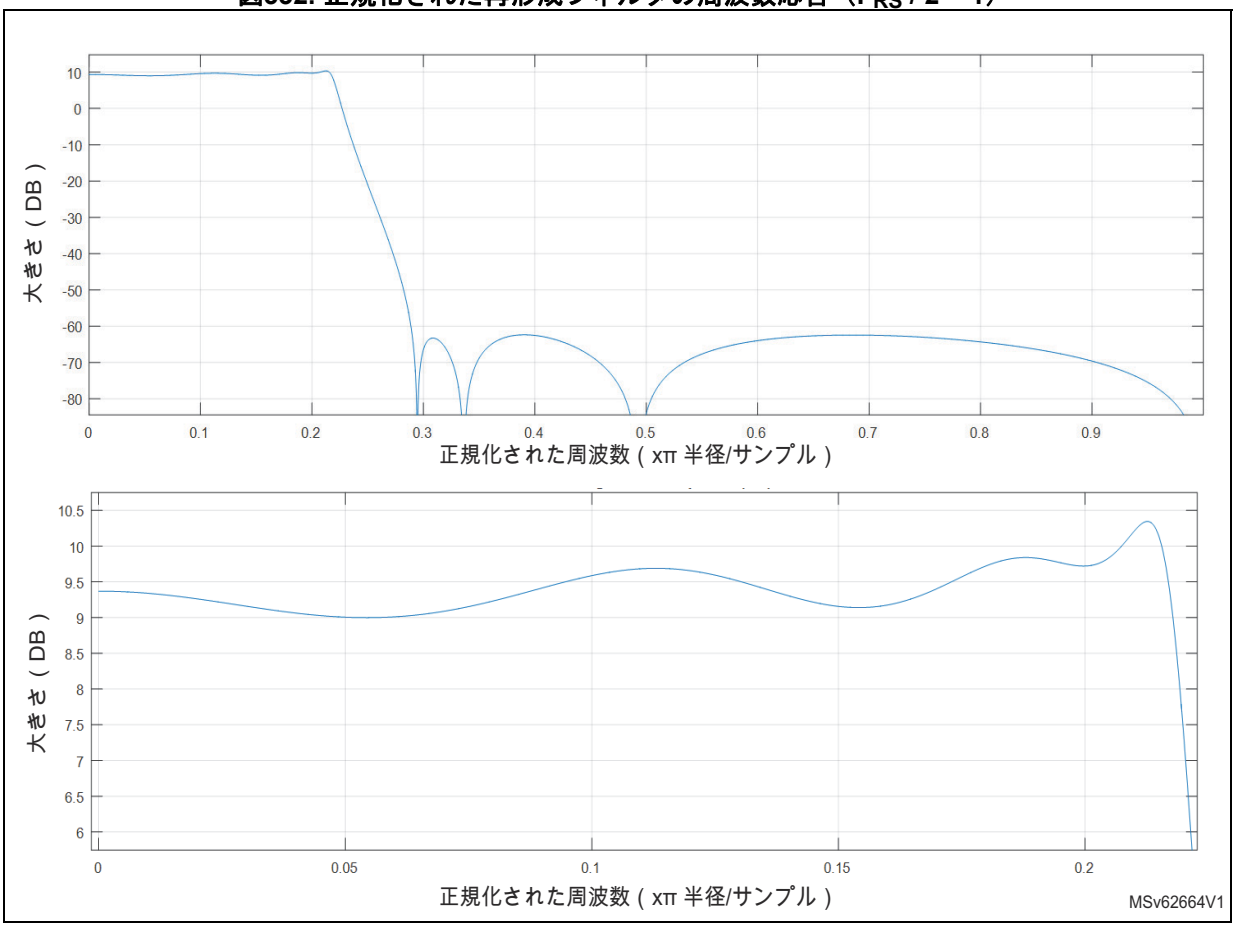
$$F_C = 0.111 \times F_{RS}$$

表375.一般的なマイクロフォンの設定

RSFLTでのサンプル･レート（kHz）（F_{RS}）	通過帯域（kHz）	D2	PCMサンプル･レート（kHz）
32	3.55	4	8
64	7.1	4	16
128	14.2	4	32
192	21.3	4	48

次の図に、再形成フィルタの周波数応答を示します。

図332. 正規化された再形成フィルタの周波数応答（F_{RS} / 2 = 1）



RSFLTゲインは約9.3dBであるため、出力データ･サイズは、22bit幅の入力信号では24bitよりやや小さくなります。

F_{RS} で1つのサンプルを処理するには、RSFLTでは24クロックサイクルのmdf_proc_ckクロックが必要です。RSFLTを有効化した場合、mdf_proc_ckの速度がF_{RS} の少なくとも24倍であることがアプリケーションによって確認される必要があります。

前のサンプルの処理中にRSFLTで新しいサンプルを受信した場合、RSFLTによりイベント（rfovr_evt）が生成され、RFOVRFフラグがセットされます。

RFOVRFがセットされた場合、RSFLTによって提供されたサンプルは無効です。アプリケーションによってデータ取得が停止され、高速のmdf_proc_ckクロックがRSFLTに提供される必要があります。

ハイパス･フィルタ（HPF）

ハイパス･フィルタは、連続変換モードの場合に、最終出力データ･ストリームの低周波数成分を抑制します。ハイパス･フィルタは、MDF再形成フィルタ設定レジスタx（MDF_DFLTxRSFR）のHPFBYPを通して有効化または無効化できます。

HPFは、入力データ･ソース中に寄生低周波数ノイズ（またはDC信号）があり、最終データから除去する必要がある場合に有益です。

HPFは一次IIRフィルタで、遮断周波数は MDF再形成フィルタ設定レジスタx（MDF_DFLTxRSFR）のHPFC[1:0] を通して次の値から選択することができます。

- 0.000625 x F_{PCM}
- 0.00125 x F_{PCM}
- 0.00250 x F_{PCM}
- 0.00950 x F_{PCM}

表376.HPF 3dB遮断周波数の例

HPFC	共通F_{PCM} 周波数（Hz）用の3dBの遮断周波数		
	F_{PCM} = 8kHz	F_{PCM} = 16kHz	F_{PCM} = 48kHz
0	5	10	30
1	10	20	60
2	20	40	120
3	76	152	456

HPF出力は24bitで飽和します。サンプルが飽和すると、SATFフラグがセットされます。

オフセット･エラー補正（OEC）

デジタル･フィルタはそれぞれ独自のOECを備えています。オフセット補正は、CICで提供される信号に対して、MDFオフセット･エラー補正制御レジスタx（MDF_OECxCR）のOFFSET[25:0] を減算して行います。

内部的な再同期により、アプリケーションで新しいオフセットを書き込んでから、新しいオフセット値が効果的にサンプルに適合されるまでの間に、mdf_proc_ckクロックの一部のサイクルで遅延が発生します。前の値がまだ適用されていないうちにアプリケーションで新しいオフセット値を書き込もうとした場合、この新しいオフセット値は無視されます。OFFSET[25:0] を読み戻すと、アプリケーションに現在のオフセット値が通知されます。

積分器（INT）

INTは、デジタル･フィルタからのデータにおける追加のデシメーションおよび分解能向上を実行します。INTは、単純にフィルタからの特定のデータサンプル数に対してデジタル･フィルタからのデータの合計を実行します。

INTを有効化するには、INTVAL[6:0] に0以外の値をセットします。

積算値の量は、2 ~ 128の範囲でINTVAL[6:0] によって定義できます。

積分器出力でデータ幅を制御するために、得られたデータを1、4、32、または128で分周することができます。この機能は、INTDIV[1:0] によって制御されます。

39.4.9 限界外検出回路（OLD）

OLDは、信号が所定の閾値の上限または下限に達するか、これを超えた場合に、イベントをトリガします。条件が満たされた場合、生成されたイベントにより割込みまたはブレーク信号（mdf_break[3:0]）を駆動できます。

OLDは、CICフィルタがモータおよび感知モード（CICMOD[2:0] = 0xx）で設定されている場合のみ使用できます。

OLDはメイン･フィルタで選択された入力信号を取得し、$Sinc^N$ またはFastSincフィルタ（ACIC）を使用して処理し、生成される信号をプログラムされた閾値と比較します。

OLDはMDF_OLD0CRのOLDENビットを通して有効化されます。有効化されると、入力データは継続的に監視されます。OLD機能を使用するためにDFLTx を有効化する必要はありません。

MDFは、特定のデータ値と比較するための閾値の上限と下限のレジスタを備えています。

信号がこれらの2つの閾値で定義された境界線の範囲内または範囲外の場合、アプリケーションによりイベントを生成できます。この動作は、MDF_OLD0CRのTHINBを通して制御されます。

入力信号がOLDTHHより大きい場合のみアプリケーションでイベントが生成されるようにしたい場合、THINBとOLDTHLの両方を0にクリアする必要があります。

入力信号がOLDTHHより小さい場合のみアプリケーションでイベントが生成されるようにしたい場合、THINBには1をセットし、OLDTHLは0にクリアする必要があります。

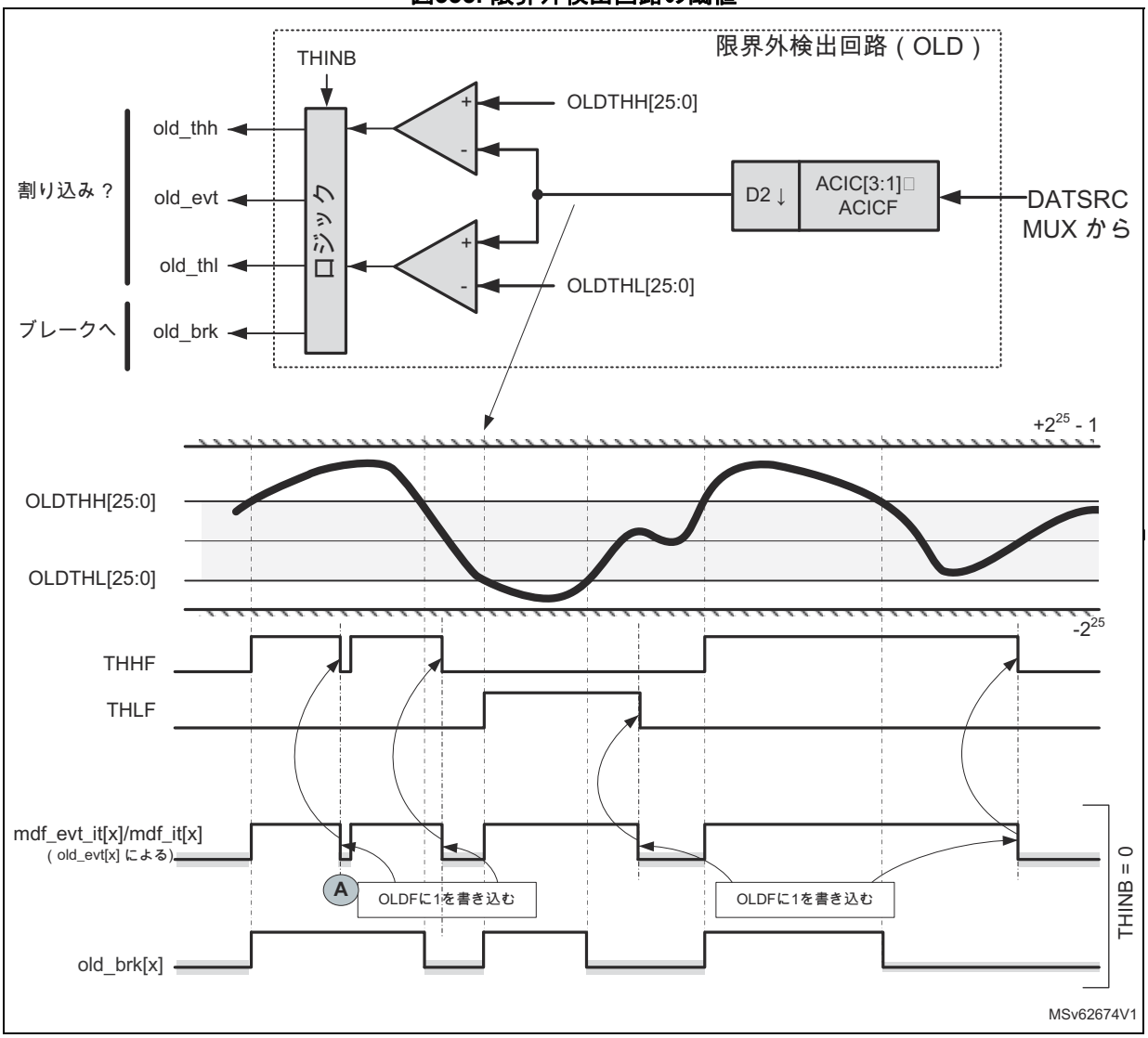
注： **OLDTHLにOLDTHHより大きな値をセットすることは推奨できません。**

OLDの応答時間は、いくつかのパラメータによって決まり、最も重要なものを次に示します。

- 外部センサによって使用されるサンプル･レート周波数
- ACICのデシメーション比
- ACICの次数

OLDは過電流の検出に使用できますが、old_brk信号を受信するタイマによってPWM信号が生成される場合、電流リミッタとしても使用できます。一般的に、過電流検出の応答時間を短くするには、ACICの次数とデシメーション比をできるだけ小さくすることを推奨します。FastSincも過電流の検出のために有益です。アプリケーションにより、フィルタの分解能と応答時間の妥協点を見出す必要があります。

図333. 限界外検出回路の閾値



ステータス･フラグを使用して、OLDイベントが検出されたことをアプリケーションに通知することができます。ラッチされたイベントをクリアするには、LOLDFビットに1を書き込みます。

図333 に示すように、THINB = 0の場合、割込み信号はグレーの領域外にある状態ではアクティブなままです。ポジションAでは、アプリケーションにより割込みがクリアされますが、信号がまだグレーの領域外にあるため、割込みは再アサートされます。アプリケーションによりOLDFがクリアされると、THHFおよびTHLFもクリアされます。

OLDイベントには、1つまたは複数のブレーク出力信号を割り当てることができます（mdf_break[3:0]）。特定の OLD イベントにブレーク信号を割り当てるには、MDF 限界外検出回路制御レジスタ x（MDF_OLDxCR）のBKOLD[3:0] を使用します。

注： **ブレーク信号の生成は、割込みの生成とは関係なく行われます。**

OLDの有効化シーケンスの例

1. CKGENを有効化し、設定します。
2. OLDENを0にセットします。
3. OLDACTIVE = 0を待ちます。OLDENが前に有効化されている場合このフェーズには2周期のmdf_hclkと2周期のmdf_proc_ckが必要となることがあります。
4. BKOLD[3:0]、ACICN[1:0]、ACICD[4:0]、THINB、OLDTHL[25:0]、およびOLDTHH[25:0] をプログラムします。
5. OLDENを1にセットします。

39.4.10 デジタル･フィルタ取得モード

MDFでは、データ･キャプチャ実行用の次のモードが用意されています。

- 非同期連続取得モード
- 非同期シングルショット取得モード
- 同期連続取得モード
- 同期シングルショット取得モード
- ウィンドウ連続取得モード
- 同期スナップショット取得モード

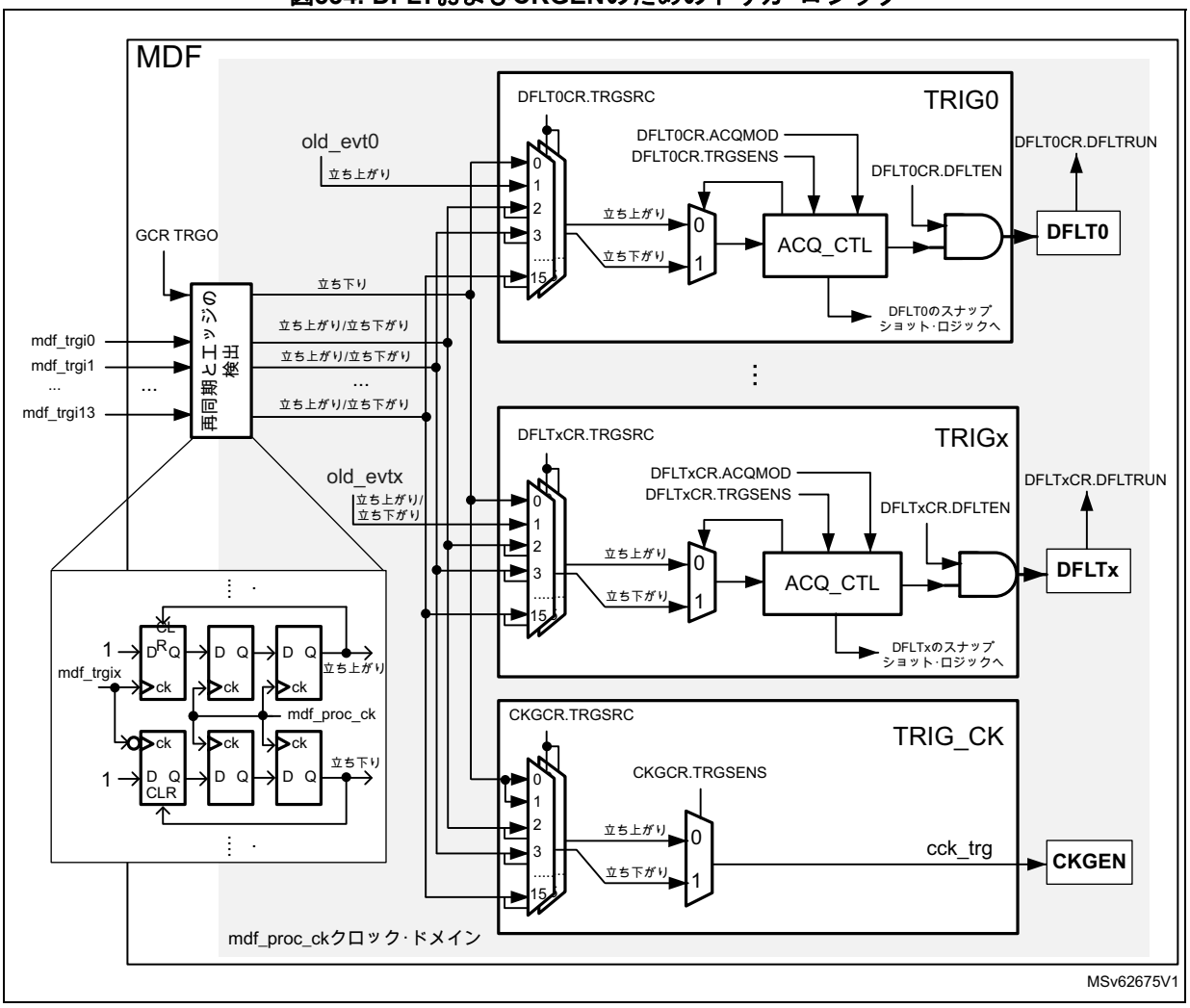
フィルタごとに、これらのいずれかのモードを個別に選択できます。

注： **データ･キャプチャを実行するには、フィルタ、データを提供するインタフェース（SITFx または ADCITF）、CKGENを有効化する必要があります。必要に応じて、MDF_CCK0またはMDF_CCK1も有効化する必要があります。**

DFLTENに0をセットした場合、フィルタをすぐに停止することができます。これにより、フィルタがリセットされ、対応するフィルタ･パスのRXFIFOが一掃されます。RXFIFOとフィルタがリセットされると、DFLTACTIVEフラグも0に戻ります。

次の図に、各フィルタとクロック･ジェネレータで利用できるトリガ･ロジックの概要を示します。

図334. DFLTおよびCKGENのためのトリガ･ロジック



すべてのTRIGブロックに共通のブロックにより、立ち上がりエッジおよび立ち下がりエッジの検出と、mdf_ker_ckクロック･ドメインへの入力トリガの再同期化が実行されます。この実装により、アプリケーションではmdf_ker_ck周期より小さなパルス幅を持つトリガを使用できるようになります。

同期モードでは、TRIGブロックにより次のいずれかのトリガ･ソースを選択できます。

- mdf_trgi[13:0] 信号（トリガ接続については、表366 を参照）
- MDF_GCRのTRGOビット
- 各フィルタ･パスのOLDイベント

TRGOイベントとOLDイベント以外では、エッジの検知方向も選択できます。

非同期連続取得モード

このモードでは、DFLTENビットに1を書き込むだけで、アプリケーションにより 1つまたは複数のフィルタでの連続取得を開始することができます。

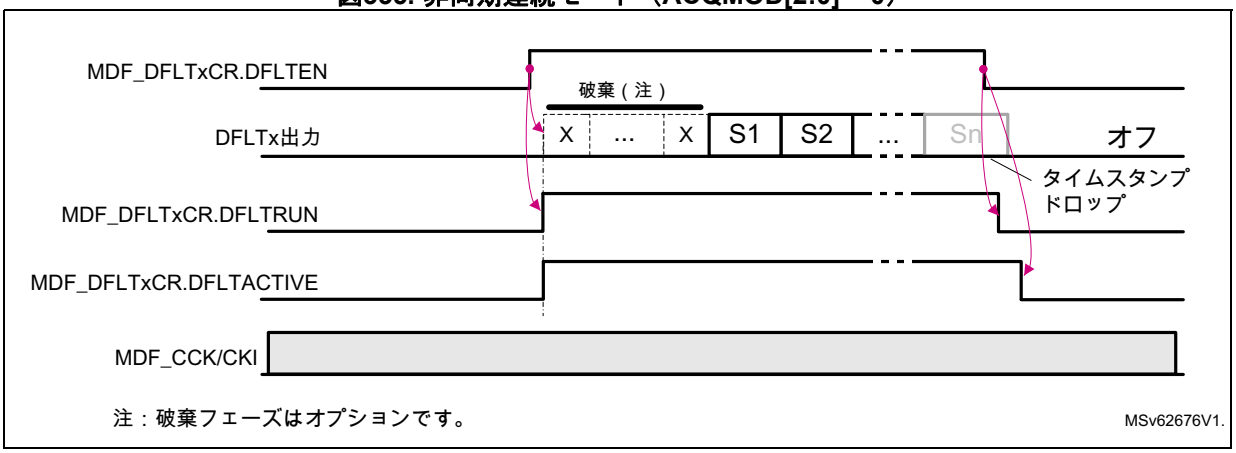
非同期連続取得モードは、ACQMOD[2:0] = 0のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTEN ビットに 0がセットされていることが前提です）。

1. mdf_proc_ck周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表380 の例を参照）。
2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に0をセットします。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
6. フィルタを有効化するためにDFLTENに1 がセットされると、取得シーケンスがすぐに開始されます。

次の図に、DFLTx によって生成されるサンプルの簡単な例を示します。

図335. 非同期連続モード（ACQMOD[2:0] = 0）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

非同期シングルショット取得モード

このモードでは、DFLTENビットに1を書き込むだけで、アプリケーションにより 1つまたは複数のフィルタでの1つのサンプルの取得を開始することができます。

非同期シングルショット取得モードは、ACQMOD[2:0] = 001のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTEN ビットが 0にクリアされていることが前提です）。

1. mdf_proc_ck周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表376 の例を参照）。

2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に001をセットします。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
6. フィルタを有効化するためにDFLTENに1 をセットすると、選択された各フィルタによりRXFIFOにデータが1つ提供され、取得が停止します。

新しい取得をトリガするには、各フィルタについて、アプリケーションで次のことを実行する必要があります。

1. DFLTRUN = 0になるのを待って、前の取得が完了したことを確認します。
2. DFLTENに再度1をセットします。

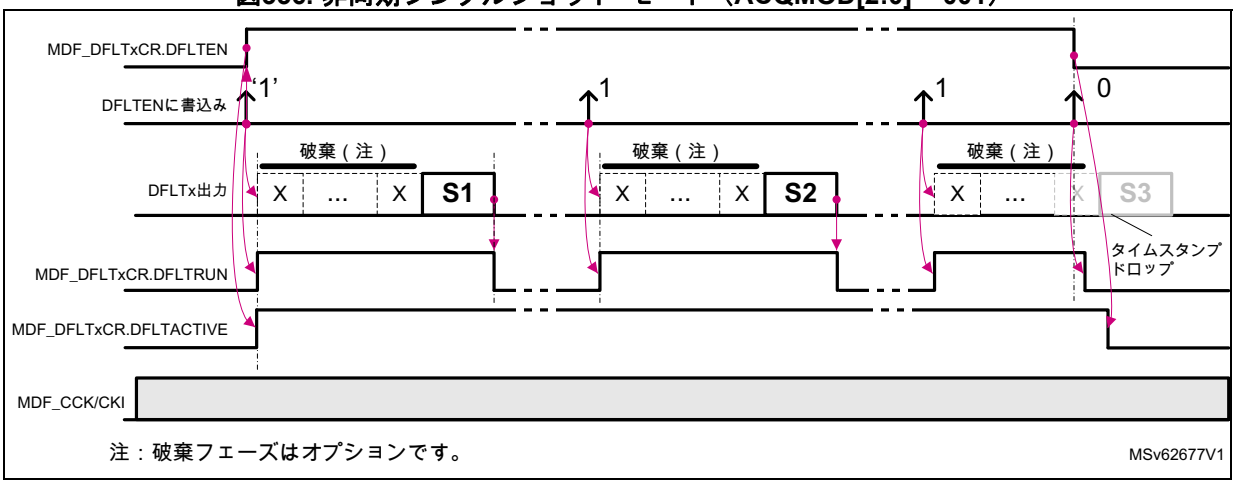
このシーケンスは、新しいデータの変換が必要になるたびに繰り返すことができます。

図336 に示すように、DFLTENに1がセットされるたびに、取得シーケンスがトリガされます。必要に応じて、フィルタによって提供される最初のサンプルを破棄することができます。各変換の最後に、デシメーション･カウンタとフィルタ･タップがリセットされ、フィルタは新しい変換の開始が可能になります。

取得の実行中にDFLTENに0がセットされた場合、実行中の変換は停止します（この例では、S3は失われます）。この状況を回避するには、次の手順を実行します。

1. DFLTRUN = 0を待ちます。
2. RXFIFOからサンプルを読み出します。
3. DFLTENを0にセットします。

図336. 非同期シングルショット･モード（ACQMOD[2:0] = 001）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

同期連続取得モード

このモードでは、次のトリガ･ソースを使用して、アプリケーションにより 1つまたは複数のフィルタでの連続取得を開始することができます。

- いずれかのmdf_trg[13:0] 信号
- 対応するフィルタのOLDイベント
- TRGOビット

同期連続取得モードは、ACQMOD[2:0] = 010のときに選択されます。

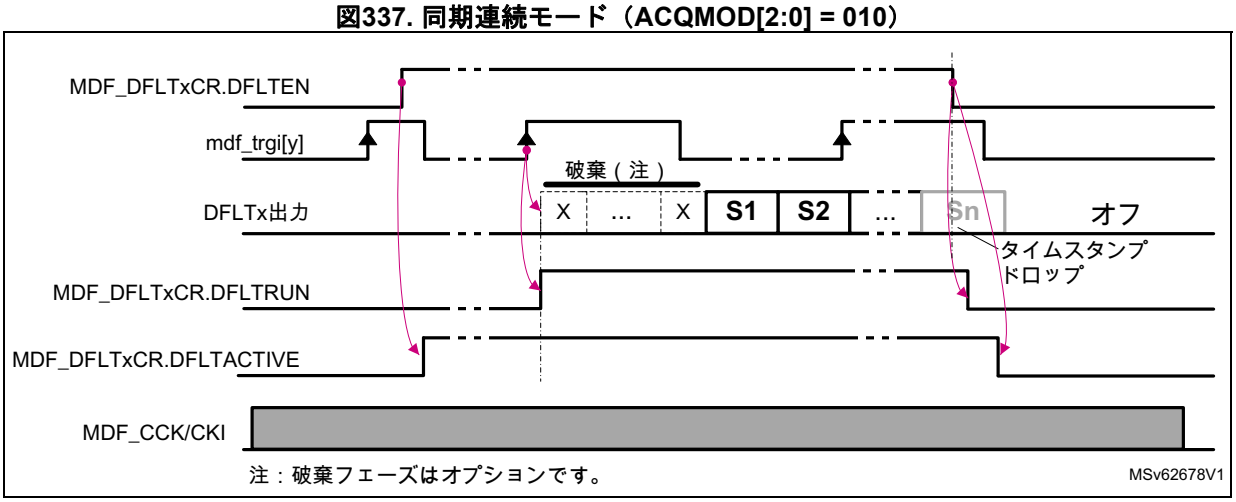
次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTENビットに0がセットされていることが前提です）。

1. mdf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表376 の例を参照）。
2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に010をセットします。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. 各フィルタに適したトリガ･ソースとエッジ検知方向を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
7. フィルタを有効化するためにDFLTENに1をセットします。
8. トリガの条件が満たされると、フィルタで取得が開始されます。

TRGSENSビットでは、トリガのエッジ（立ち上がりまたは立ち下がり）を選択できます。取得の実行中、またはDFLTENに0がセットされている場合、トリガは無視されます。

次の図に、トリガ･ロジックが立ち上がりエッジのトリガの影響を受ける（TRGSENS = 0）場合の簡単な例を示します。DFLTEN = 0であるため、トリガ信号の最初の立ち上がりエッジは無視されます。次の立ち上がりエッジは考慮され、取得が開始されます。その他すべての立ち上がりエッジは無視されます。DFLTRUNが0に戻ると、トリガ･ロジックは再初期化されます。

図337. 同期連続モード（ACQMOD[2:0] = 010）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

同期シングルショット取得モード

このモードでは、次のトリガ･ソースを使用して、アプリケーションにより 1つまたは複数のフィルタでのシングル取得を開始することができます。

- いずれかのmdf_trg[13:0] 信号
- 対応するフィルタのOLDイベント
- TRGOビット

同期シングルショット取得モードは、ACQMOD[2:0] = 011のときに選択されます。

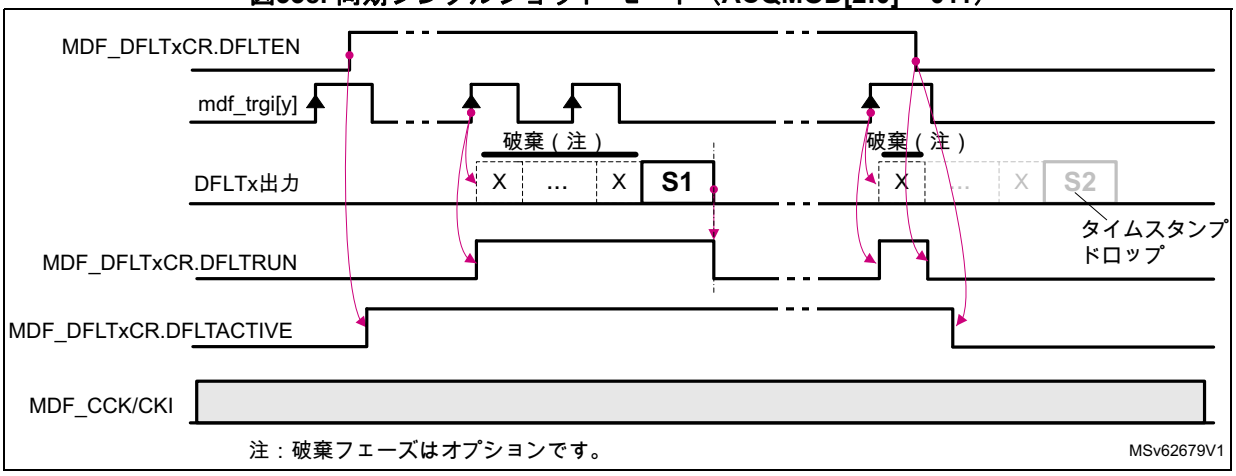
次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTEN ビットに 0がセットされていることが前提です）。

1. mdf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表376 の例を参照）。
2. CKGDENを有効化し、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. 必要なフィルタ設定をプログラムし、ACQMOD[2:0] に011をセットします。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. 各フィルタに適したトリガ･ソースとエッジ検知方向を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了していることを確認できます。
7. フィルタを有効化するためにDFLTENに1 をセットします。
8. トリガ条件が満たされると、フィルタで取得が開始され、RXFIFOにデータが1つ提供され、フィルタで新しいトリガを受け付けられる状態になります。

TRGSENSでは、トリガのエッジ（立ち上がりまたは立ち下がり）を選択できます。取得の実行中、またはDFLTENに0がセットされている場合、トリガは無視されます。

次の図に、トリガ･ロジックが立ち上がりエッジのトリガの影響を受ける（TRGSENS = 0）場合の簡単な例を示します。DFLTEN = 1でトリガの立ち上がりエッジが検出されるたびに、取得シーケンスがトリガされます。必要に応じて、フィルタによって提供される最初のサンプルを破棄することができます。各変換の終了時に、デシメーション･カウンタとフィルタ･タップがリセットされます。DFLTRUNに0がセットされ、フィルタは新しい変換の開始が可能になります。

図338. 同期シングルショット･モード（ACQMOD[2:0] = 011）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

図338 に、取得の実行中にDFLTENが0にクリアされたため、サンプルS2が失われるケースを示します。この状況を回避するには、次の手順を実行します。

1. DFLTRUN = 0を待ちます。
2. RXFIFOからサンプルを読み出します。
3. DFLTENを0にクリアします。

注： **実行中のDMA転送は適切に終了します。**

ウィンドウ連続取得モード

このモードでは、アプリケーションにより、次のいずれかのトリガ･ソースの連続エッジで制御される 1つまたは複数のフィルタでの連続取得を開始または停止することができます。

- いずれかのmdf_trg[13:0] 信号
- TRGOビット

ウィンドウ連続取得モードは、ACQMOD[2:0] = 100のときに選択されます。

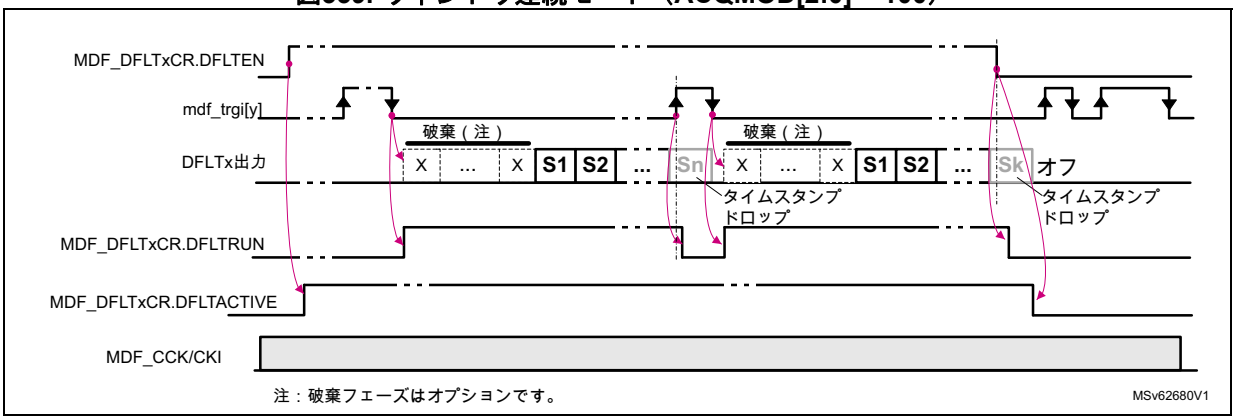
次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTEN ビットに 0がセットされていることが前提です）。

1. mdf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表376 の例を参照）。
2. CKGDENを有効化し、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. 必要なフィルタ設定をプログラムし、ACQMOD[2:0] に100をセットします。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. 各フィルタに適したトリガ･ソースとエッジ検知方向を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了していることを確認できます。
7. フィルタを有効化するためにDFLTENに1 をセットします。
8. TRGSENS = 0の場合、取得はトリガの立ち上がりエッジで開始され、トリガの立ち下がりエッジで停止します。TRGSENS = 1の場合、取得はトリガの立ち下がりエッジで開始され、トリガの立ち上がりエッジで停止します。

注： **トリガ条件が再び有効になると、取得が再開可能になります。**

図339 に、TRGSENS = 1でのウィンドウ連続取得モードの簡単な例を示します。DFLTENに1がセットされると、MDFは選択されたトリガ入力で立ち下がりエッジを待ちます。トリガ条件が満たされると、DFLTRUNが1に戻り、取得が開始されます。MDFにより、選択されたトリガ入力で立ち上がりエッジが検出されると、取得は停止します。DFLTENに1がセットされたままであれば、MDFは選択されたトリガ入力で再び立ち下がりエッジを待ちます。

図339. ウィンドウ連続モード（ACQMOD[2:0] = 100）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

同期スナップショット取得モード

前のセクションで説明した取得モードでは、アプリケーションは有効な新しいサンプルを取得する前に、フィルタの設定時間または変換の終了を待つ必要があります。遅延の影響が大きいアプリケーションでは、スナップショット取得モードを使用して、このサンプルの取得時期に関する情報と一緒に前回の有効なサンプルをすぐに取得することができます。

スナップショット･モードでは、DFLTxは継続的にサンプルを取得していますが、処理されたサンプルはRXFIFOに格納されません。トリガが発生すると、前回の有効なサンプル、MCICの現在のデシメーション値、INTのサンプル･カウンタがMDF_SNPS0DRに格納されます。

次のトリガ･ソースが利用できます。

- いずれかのmdf_trg[13:0] 信号
- 対応するフィルタのOLDイベント
- TRGOビット

同期スナップショット取得モードは、ACQMOD[2:0] = 101のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（フィルタのDFLTEN ビットに 0がセットされていることが前提です）。

1. mdf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表376 の例を参照）。
2. CKGDENを有効化し、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
3. 必要なフィルタ設定をプログラムし、ACQMOD[2:0] に101をセットします。RXFIFOが空でなくなるとすぐに割込みを受信できるように、FTHは通常、0にクリアされます。
4. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
5. 各フィルタに適したトリガ･ソースとエッジ検知方向を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了していることを確認できます。
7. フィルタを有効化するためにDFLTENに1 をセットします。

トリガ条件が満たされると、次のようになります。

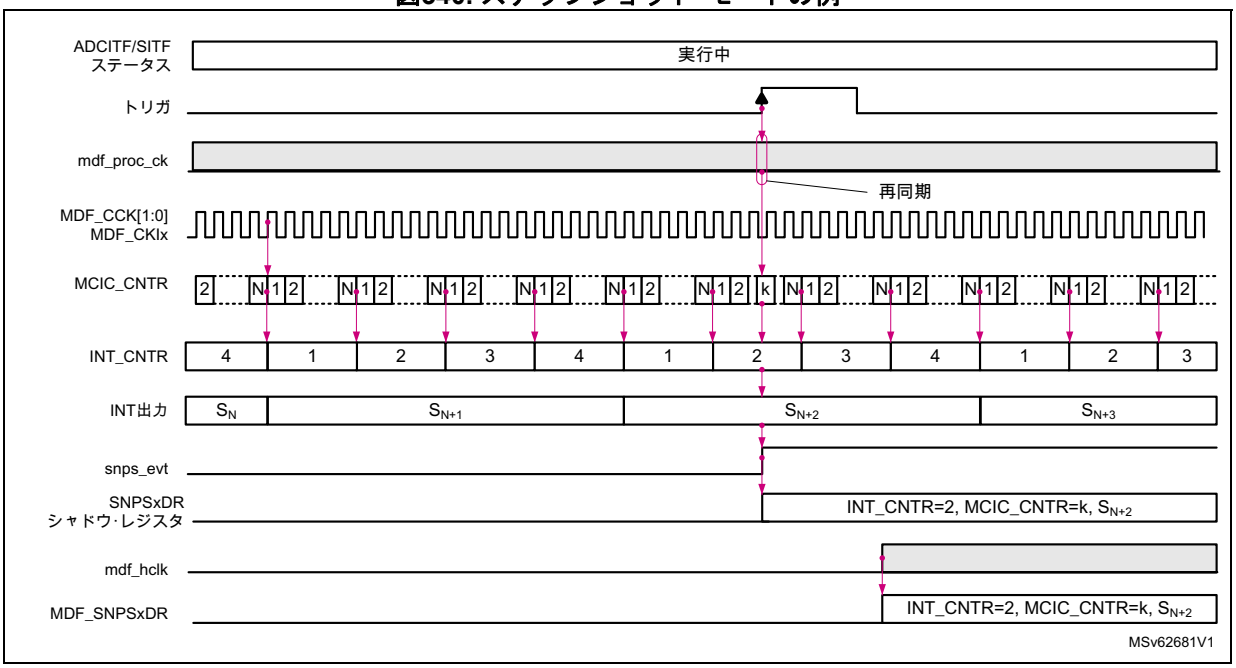
1. SNPSFMT によって選択されている場合、対応するフィルタによって提供された前回の有効なデータ、MCICデシメーション･カウンタの値、INTのカウンタ値がMDTによってシャドウ･レジスタに格納されます。
2. snps_evtが有効化され、データの準備が整っていることが示されます。SSDRIEに1がセットされている場合、MDFからAHBクロックがリクエストされます。
3. AHBクロックが利用可能な場合、シャドウ･レジスタが MDFスナップショット･データ･レジスタx（MDF_SNPSxDR）に格納され、割込みが生成されます。
4. 割込みサブルーチンで、アプリケーションにより次のことが行われる必要があります。
   a) MDF_SNPSxDRに格納されたデータの読出し
   b) SSDRFフラグに1を書き込んでクリア

図340 に示すように、スナップショット･モードでは、トリガ･イベントが発生すると、処理クロック（mdf_proc_ck）と同期され、MCICのデシメーション･カウンタ （MCIC_CNTR）、積分器カウンタ（INT_CNTR）、前回の有効なサンプル（S_{N+2}）がシャドウ･レジスタに格納され、snps_evtイベントが生成されます。AHB クロックが利用可能な場合、割込みが生成され、シャドウ･レジスタがMDF_SNPSxDRにコピーされます。アプリケーションでは、すぐにサンプルとタイミング情報を読み出すことができます。

SSOVRFオーバーラン･フラグも利用でき、オーバーラン状態が発生しているかどうかをアプリケーションで確認することができます。スナップショットのオーバーラン状態の検出は、新しいトリガ･イベントの発生時にSSDRFフラグに1がセットされている場合に行われます。

その場合、新しいトリガ･イベントは考慮されません。アプリケーションによりSSOVRFフラグをクリアする必要があります。

図340. スナップショット･モードの例



注： **有効化されたフィルタでデータ･ストローブが受信されない場合、フィルタはフリーズします。ストリームを提供するインタフェースが有効化されていない場合（SITFまたはADCITF）、フィルタはフリーズします。DFLT xが有効化されると、CIC、HPF、IIR、およびINTは再初期化されます。各フィルタについて、ステータス･ビットにより、フィルタが現在動作中かどうかが示されます。**

複数のフィルタの同期開始

複数のフィルタの取得を同期させて開始するには、次のシーケンスを実行する必要があります（DFLTENに0がセットされていることが前提です）。

1. CKGDENを有効化し、必要に応じて、MDF_CCK0およびMDF_CCK1クロックを有効化します。
2. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
3. 各フィルタについて、取得モードを「同期」にセットします（ACQMOD[2:0] = 01x）。
4. 各フィルタについて、TRGSRC[3:0] に0をセットします（TRGOを選択）。
5. 各フィルタについて、TRGSENSに0をセットします（立ち上がりエッジ）。
6. 各フィルタについて、DFLTENに1をセットします。
7. 0が読み出されるまで、TRGOビットを読み出します。
8. TRGOを1にセットします。すぐに、選択したすべてのフィルタの取得シーケンスが開始されます。

新しい取得をトリガするには（シングルショットの場合）、アプリケーションで次のことが行われる必要があります。

1. DFLTRUN = 0になるのを待って、前の取得が完了したことを確認します。
2. 0が読み出されるまで、TRGOを読み出します。
3. TRGOに再度1をセットします。

破棄するサンプル

MDFでは、次の目的で、再起動後に破棄するサンプルの量をプログラムすることができます。

- フィルタのインパルス応答の影響を受けたサンプルのキャプチャを回避するため
- 特定のサンプルの数だけフィルタの取得を遅延させるため

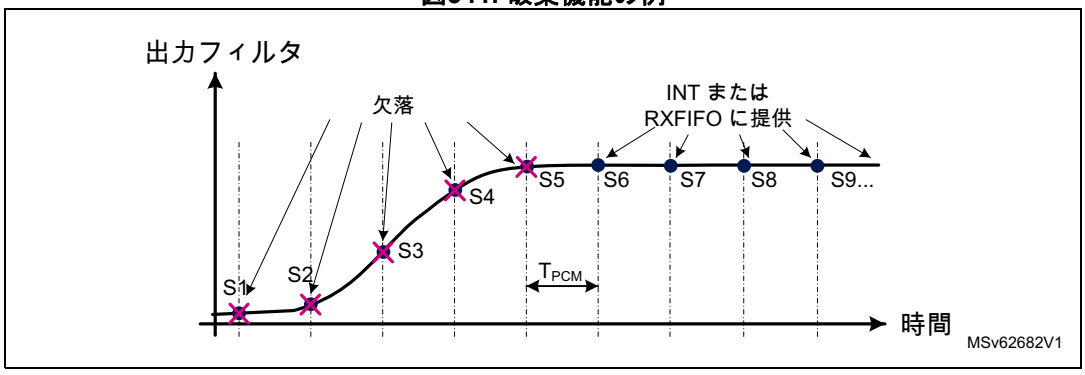
破棄機能は、次のようにNBDIS[7:0] を通して制御されます。

- NBDIS[7:0] = 0の場合、破棄機能は無効化されます。
- NBDIS[7:0] ≠ 0の場合、次のいずれかの条件で破棄機能は有効化されます。
  - DFLTENビットが1に変更されたとき
  - （非）同期シングルショット･モードで取得が開始されるたび

図335~ 図339、図342および 図343を参照してください。

次の図に示す例では、デジタル･フィルタによって提供される最初の5つのサンプル（S1 ~ S5）を破棄するために破棄機能が使用されています。RXFIFO（または、有効化されている場合はINTブロック）に最初に転送されるサンプルはS6です。

図341. 破棄機能の例



警告：　インタリーブDMAモードで動作するすべてのフィルタでは、同じNBDIS[7:0] 値を使用する必要があります。

データ･インタフェースの有効化

データ･インタフェースを有効化するには、対応するSITFENまたはADCITFENビットに1をセットします。デジタル･フィルタが有効化されると、外部 ΣΔ モジュレータまたはパラレル内部データ･ソース（ADC）からシリアル･データを受信します。

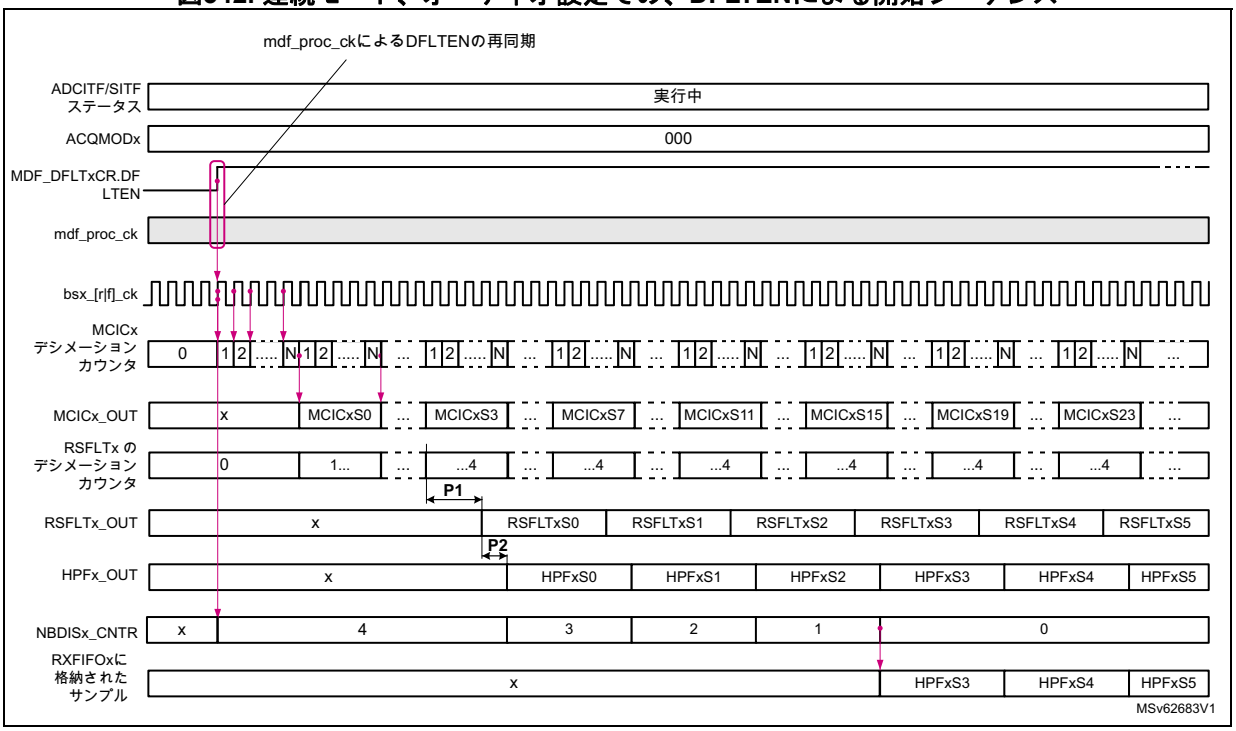
39.4.11 開始シーケンスの例

図342 に、NBDIS[7:0] = 3（取得の前に3つのサンプルを破棄）という設定で、DFLTEN（ACQMOD[2:0] = 0）によりトリガされるデジタル･フィルタの取得シーケンス開始の詳細を示します。

DFLTx はオーディオ･アプリケーション用に設定され、MCIC、RSFLT、およびHPFが有効化されます。データインタフェース（SITFx またはADCITF）はすでに有効化されていることが前提となります。

注：　図を簡素化するため、NBDIS[7:0] には意図的に低い値をセットします。

図342. 連続モード、オーディオ設定での、DFLTENによる開始シーケンス



DFLTENビットは、MDF処理クロック･ドメインに再サンプリングされます。DFLTENがハイとして検出されると、フィルタ･チェーンが有効化され、MCICフィルタのデシメーション･カウンタがビットストリーム･クロックのレートでインクリメントされます。

MCICデシメーション･カウンタがプログラムされた値Nに達すると、RSFLTでサンプルが使用可能になります。

RSFLTは、MCICによって提供されるすべてのサンプルを処理し、4つのサンプルを処理するたびにHPFにサンプルを送信します（4でデシメーション）。RSFLTではサンプル（P1）を送信する前に、最大24サイクルのmdf_proc_ckクロックが必要です。

HPFはRSFLTによって提供されたすべてのサンプルを処理しますが、NBDIS_CNTRが0に達していない間は、NBDIS機能によってRXFIFOへのデータの書込みが防止されます。このカウンタは、HPFによってサンプルが送信されるたびにデクリメントされます。

NBDIS_CNTRが0に達すると、HPFによって提供されるサンプルはRXFIFOに格納されます。

図343 に示す例は、モータ制御フィルタ設定に基づくものです。

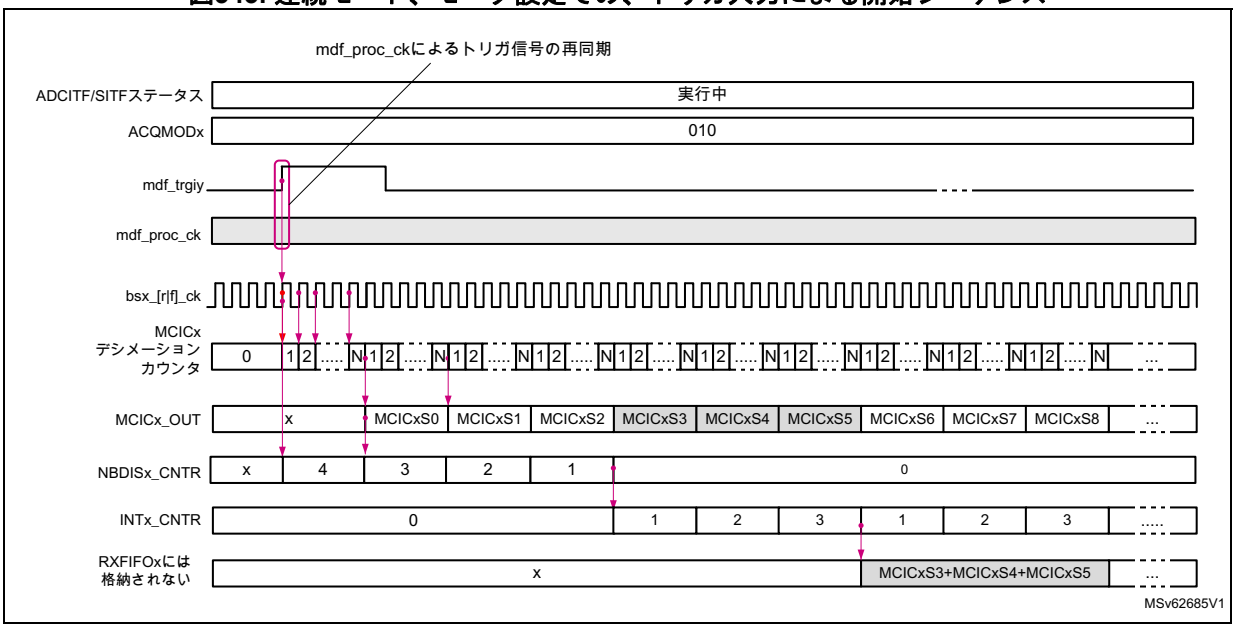
- MCICおよびINTは有効です。
- SFLTおよびHPFは無効です。
- ACQMOD[2:0] は1がセットされ、トリガとしてmdf_trgiy入力を使用します。
- NBDIS[7:0] は3がセットされます。
- INTVAL[6:0] は2がセットされます。

メタステーブルを回避するため、mdf_trgiy入力信号はMDF処理クロック･ドメインに再サンプリングされます。トリガ条件が満たされると（立ち上がりエッジ）、MCICフィルタのデシメーション･カウンタがビットストリーム･クロックのレートでインクリメントされます。

MCICデシメーション･カウンタがプログラムされた値Nに達すると、サンプルが使用可能になります。NBDIS_CNTRが0に達していない間は、NBDIS機能によってINTへのサンプルの書込みが防止されます。このカウンタは、MCICによってサンプルが送信されるたびにデクリメントされます。

NBDIS_CNTRが0に達すると、新しいサンプルがINTに提供され、ここで3つの連続するサンプルが統合され、RXFIFOに格納されます。

図343. 連続モード、モータ設定での、トリガ入力による開始シーケンス

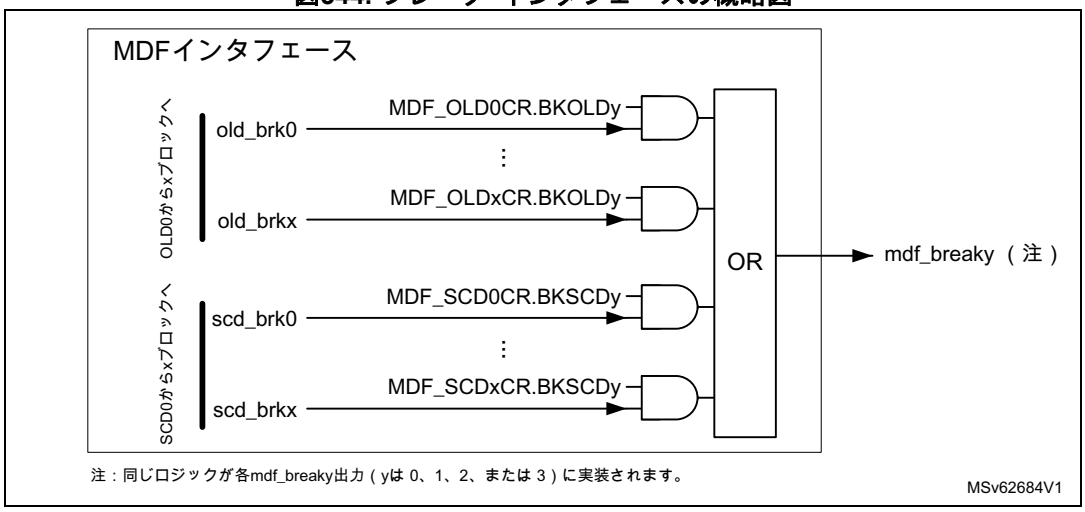


39.4.12 ブレーク･インタフェース

ブレーク･インタフェースは、OLDx およびSDCx ブロックから受信するブレーク･イベントを、製品の各ペリフェラルに接続された4つのブレーク信号に統合します（詳細は表366 を参照）。

図344 に示すように、複数のブロックで同じブレーク･ラインを共有できます。同じブロックで複数のブレーク･ラインを駆動することもできます。ブレーク･インタフェースはMDF_OLDxCRおよびMDF_SCDxCRを通して制御されます。

図344. ブレーク･インタフェースの概略図



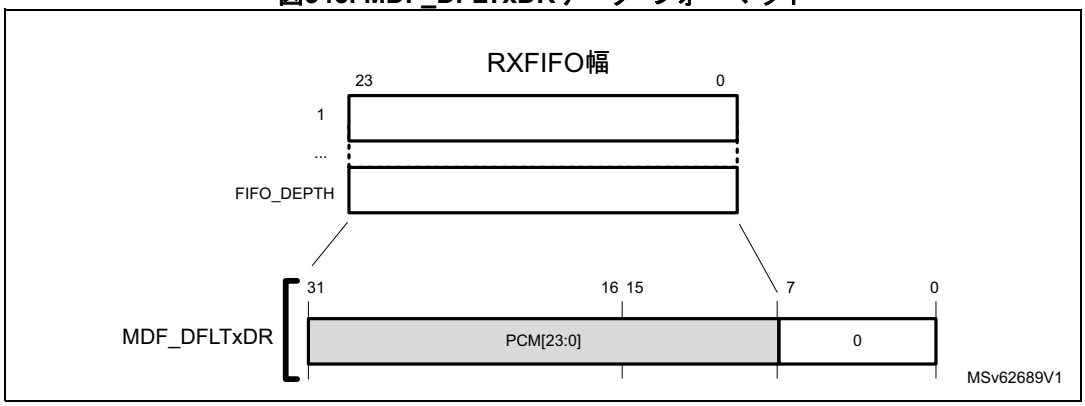
39.4.13 メモリへのデータ転送

データフォーマット

DFLTx によって処理されるサンプルは、RXFIFOに格納されます。アプリケーションでは、MDFデジタル･フィルタのデータ･レジスタx（MDF_DFLTxDR）を通して、FIFO に格納されたサンプルを読み出すことができます。このレジスタ内のサンプルは符号付きで、左詰めになっています。ビット31は常に符号を表します。

MDFは24bitの左詰めのデータを提供します。MDF_DFLTxDR への16bitのアクセスを実行すると、16の最上位ビットが取得できます。MDF_DFLTxDR への32bitのアクセスを実行すると、24bitのデータ･サイズが得られます。

図345. MDF_DFLTxDRデータ･フォーマット



データの再同期化

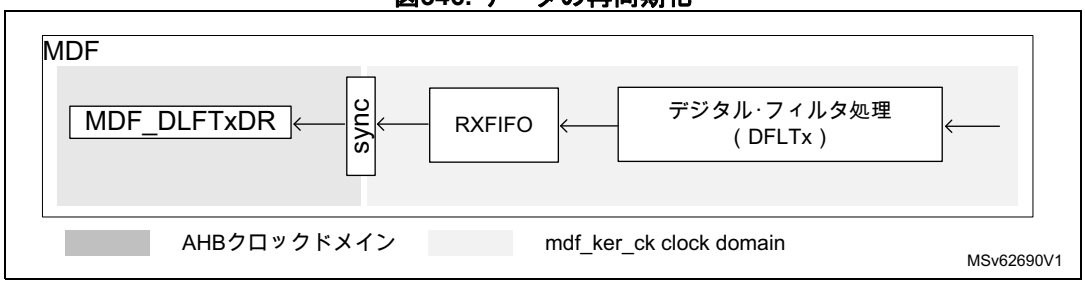
RXFIFO に格納されたサンプルは、DMAリクエストまたは割込み信号を使用して、メモリに転送することができます。

注： **RXFIFO は mdf_ker_ckクロック･ドメインに配置されますが、MDF_DFLTxDR は mdf_hclk（AHB）クロック･ドメインに配置されます。**

AHBクロックが利用可能な状態で、MDF_DFLTxDR が空でサンプルがRXFIFOにある場合、このサンプルは MDF_DFLTxDR に転送されます。

RXFIFOから MDF_DFLTxDR へのサンプルの転送には、2周期のAHBクロック（mdf_hclk）と2周期のmdf_ker_ckクロックが必要です。RXFIFOから MDF_DFLTxDR への新しいサンプルの転送が完了していない状態でアプリケーションにより MDF_DFLTxDR の読出し操作が行われると、MDFは自動的にウェイトステートを挿入します。

図346. データの再同期化



MDFでは、2つの転送タイプ、独立転送モードとインタリーブ転送モードを組み合わせることもできます。

独立転送モード

このモードでは、RXFIFOはそれぞれ独自のDMAチャネルと独自のFTHFフラグ･イベントを備えています。

このモードではシングルDMA転送とバーストDMA転送の両方がサポートされていますが、アプリケーションでは次の点に注意する必要があります。

- RXFIFOに少なくともバーストサイズと同じ量のサンプルが格納されている必要があります。
- 前のセクションで説明したデータの再同期化により､バーストモードを効率的に減らすことができます。

また、アプリケーションではRXFIFO閾値（FTHビット）を選択してデータ転送をトリガすることができます。データ転送は、RXFIFOが空でなくなった直後に、またはRXFIFOがハーフフル（半分の深さのサンプル）になったときにトリガできます。

DMA転送では、RXFIFOの1つが閾値レベルに達するとすぐに、データ転送を要求するために、対応するDMAリクエストがアサートされます。対応するRXFIFOが空になるまで、DMAリクエストが連続で行われます。

RXFIFOxのDMAモードは、MDFデジタル･フィルタ制御レジスタx（MDF_DFLTxCR）の対応するDMAENビットを通して有効化されます。

割込み信号については、次のケースを考慮する必要があります。

- FTH = 0の場合、MDF_DFLTxDRでデータが利用可能になるとすぐに、対応するFTHFがセットされ、割込みの生成が可能になります。FTHFはMDF_DFLTxDRが読み出されるとすぐに解放されます。

- FTH = 1の場合、RXFIFOの1つが閾値レベルに達し、データがMDF_DFLTxDRで利用可能になるとすぐに、対応するFTHFがセットされ、割込みの生成が可能になります。FTHFは、データが1つ読み出されるとすぐに解放されます。閾値の条件が再び満たされると、FTHFが再びセットされます。このモードでは、割込みが発生するたびに、FIFO_SIZE / 2のデータが読み出されます。

独立転送モードは、各フィルタ･パスによって提供されるサンプル･レートの同期が完全でない場合や、ストリームが独立している場合に使用する必要があります。この状況は、例えば次のような場合に発生します。

- SITF がスレーブ･モードで、外部センサがそれぞれ独自のサンプリング･クロックを供給している場合
- DFLTxのデシメーション比が同じでない場合
- すべてのストリームが同時に開始しているわけではない場合

インタリーブ転送モード

このモードでは、シングルDMAリクエストを複数のRXFIFOで共有することにより、DMAリクエスト･リソースを最適化します。

このモードはシングルDMA転送にのみ対応しています。

インタリーブ転送モードでは、FTHは使用できず、RXFIFO0のFTHFのみが割込み信号に使用できます。

DMA転送では、インタリーブ転送モードで動作するすべてのRXFIFOが空でない場合、MDF_DFLT0DRを通して各RXFIFOの1つのサンプルを順に読み出すために、一部のDMAリクエストが生成されます。

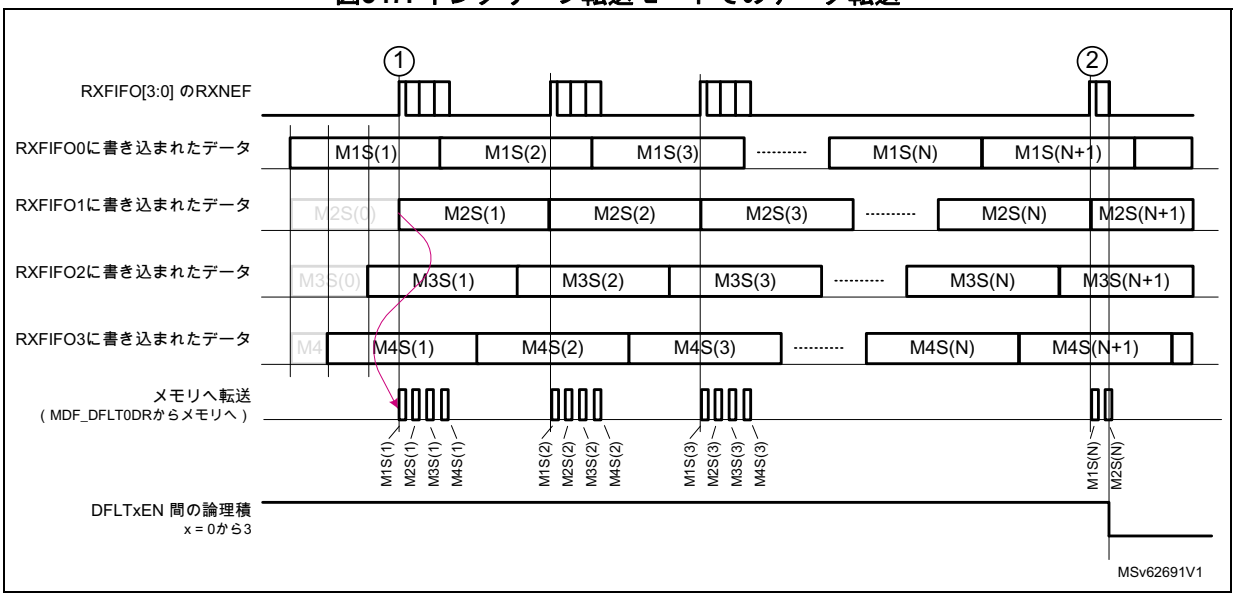
インタリーブ転送モードで動作するRXFIFOでは、DMAチャネルを有効化するために、MDF_DFLT0DRのDMAENに1をセットし、インタリーブ･モードで動作するRXFIFOの量をILVNB[3:0]を通して定義します。インタリーブ転送モードで動作する他のRXFIFOのためのDMAEN（RXFIFO 1 ~ ILVNB[3:0]）は考慮されません。

図347 に、インタリーブ転送モードで動作する4つのRXFIFOを示します。各RXFIFOには、DLYブロックのプログラミングによる遅延があります。

すべてのRXFIFOで少なくとも1つのサンプルが利用できる場合（この例では、サンプルを最後に受信するFIFOはRXFIFO1）、MDFによりメモリへのデータ転送がリクエストされます（図347 の1を参照）。この図では、DMAはメモリへのデータ転送に使用されることになっています。

取得を停止するには、インタリーブ転送モードで設定されたフィルタの1つに対して、DFTLENに0をセットします。この例では、DFLTENに0が設定されると（図347 の2を参照）、メモリへの転送はすぐに停止します。サンプルM2S（N）、M3S（N）、およびM4S（N）は失われることがあります。

図347. インタリーブ転送モードでのデータ転送



割込みモードでは、インタリーブ転送モードで動作するすべてのRXFIFOに準備できたデータがある場合、RXFIFO0のFTHFフラグがアサートされ、割込みの生成が可能になります。次に、アプリケーションにより、MDF_DFLT0DRを通して各RXFIFOの1つのサンプルが順に読み出されます。FTHFイベントは、最初のデータが読み出されると解放されます。

インタリーブ転送モードで動作するRXFIFOでは、割込みモードを有効化するために、MDF_DFLT0DRのFTHIEに1をセットし、インタリーブ転送モードで動作するRXFIFOの量をILVNB[3:0] を通して定義します。インタリーブ転送モードで動作する他のRXFIFOのためのFTHIE（RXFIFO1 ~ ILVNB[3:0]）は考慮されません。

インタリーブ転送モードを適用できるのは、RXFIFO0からRXFIFOkまでだけです。インタリーブ転送モードで動作するRXFIFOの数は、ILVNB[3:0] によって定義されます。

ILVNB[3:0] = 0の場合、RXFIFOはすべて独立転送モードで動作します。

インタリーブ転送モードが使用できるのは、相互に完全に同期してサンプルの送信ができるデジタル･フィルタの場合のみです。一般的にこれは、同じビットストリーム･クロックを共有するデジタル･マイクロホンのアレイを使用したサウンド･キャプチャの場合です。

注意： **インタリーブ転送モードでMDFを正しく動作させるには、次の規則に従う必要があります。**

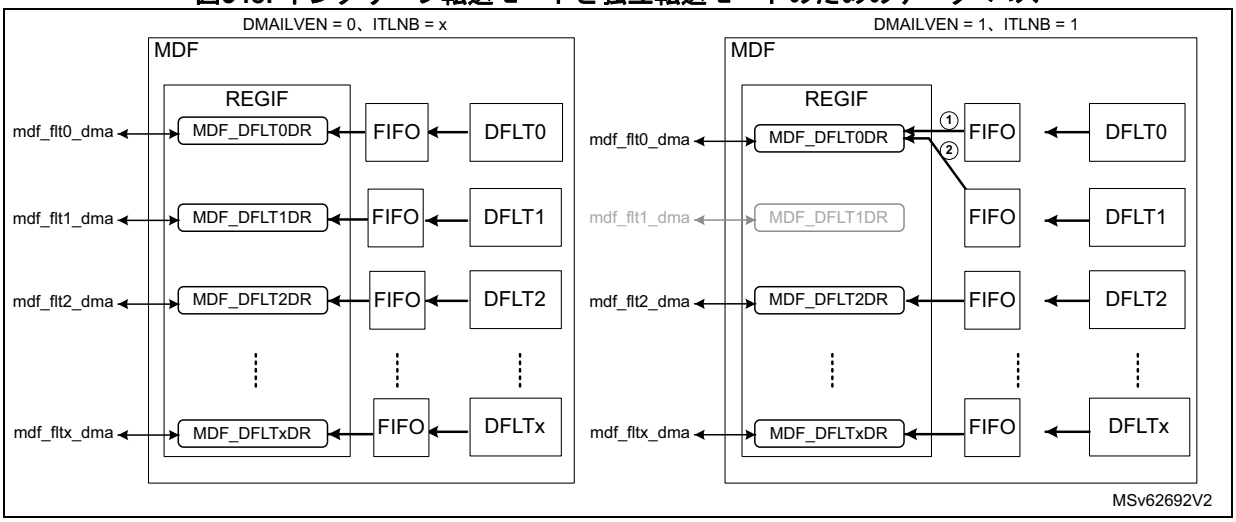
- インタリーブ転送モードで動作するすべてのフィルタが同期されていなければなりません。つまり、すべてのフィルタで同じACQMOD[2:0]（すべて010または011）、同じトリガ･ソース、同じトリガ検知方向が選択されている必要があります。
- インタリーブ転送 モードで動作するすべてのフィルタで同じ設定を使用する必要があります。
- インタリーブ転送モードで動作するフィルタにデータを提供するすべてのセンサでは、同じビットストリーム･クロック周波数を使用する必要があります。
- SPKDLY[6:0] により、インタリーブ転送モードで動作するフィルタに適用される遅延の最大差は、厳密にRXFIFOの深さよりも小さく維持する必要があります。通常、この遅延は1周期のT_{PCM}を超える必要はありません。

注： **独立転送モードとインタリーブ転送モードの両方を並行して動作させることができます。たとえば、最初のRXFIFO（RXFIFO0 ~ RXFIFOk）をインタリーブ動作モードで動作させ、それ以外（RXFIFO(k+1) ~ RXFIFO(N)）を独立転送モードで動作させることができます。**

データ転送を実行する必要がある状況でAHBクロックがない場合、MDFは最初にAHBクロックをリクエストします（詳細はセクション 39.4.5：クロック･ジェネレータ（CKGEN）を参照）。

次の図に、独立転送モードの設定のためのデータ･パス（左側の図）と、独立転送モードとインタリーブ転送モードが混在した設定のためのデータ･パス（右側の図）を示します。

図348. インタリーブ転送モードと独立転送モードのためのデータ･パス



右側の図では、DFLT0およびDFLT1フィルタはインタリーブ転送モードで設定されます。これらのフィルタでは、FTHはハードウェアによって考慮されません。DFLT0とDFLT1によって提供されるサンプルは、MDF_DFLT0DRから読み出されます。両方のRXFIFOが空でなくなると、MDFにより2つのDMAリクエストが生成されます。DFLT0からサンプルを読み出すためのリクエストと、DFLT1からサンプルを読み出すためのリクエストです。再び両方のRXFIFOが空でなくなると、同じシーケンスがトリガされます。

注： **インタリーブ転送モードで動作するフィルタの１つが無効になると、インタリーブ転送モードのすべてのフィルタでのメモリへのデータ転送もすぐに停止します。**

RXFIFOオーバーランです。

RXFIFOのオーバーラン状態は、RXFIFOがフルで、DFLTxからの新しいサンプルを書き込む必要がある場合に検出されます。

この場合、DOVRFがセットされ、新しいサンプルは破棄されます。RXFIFOで少なくとも1つの場所が利用できる場合、新しく受信するサンプルがRXFIFOに書き込まれます。

図349 に、RXFIFOの深さが4ワード、FTHに1がセットされ、RXFIFOがハーフフルのときにFTHFが１ になる例を示します。

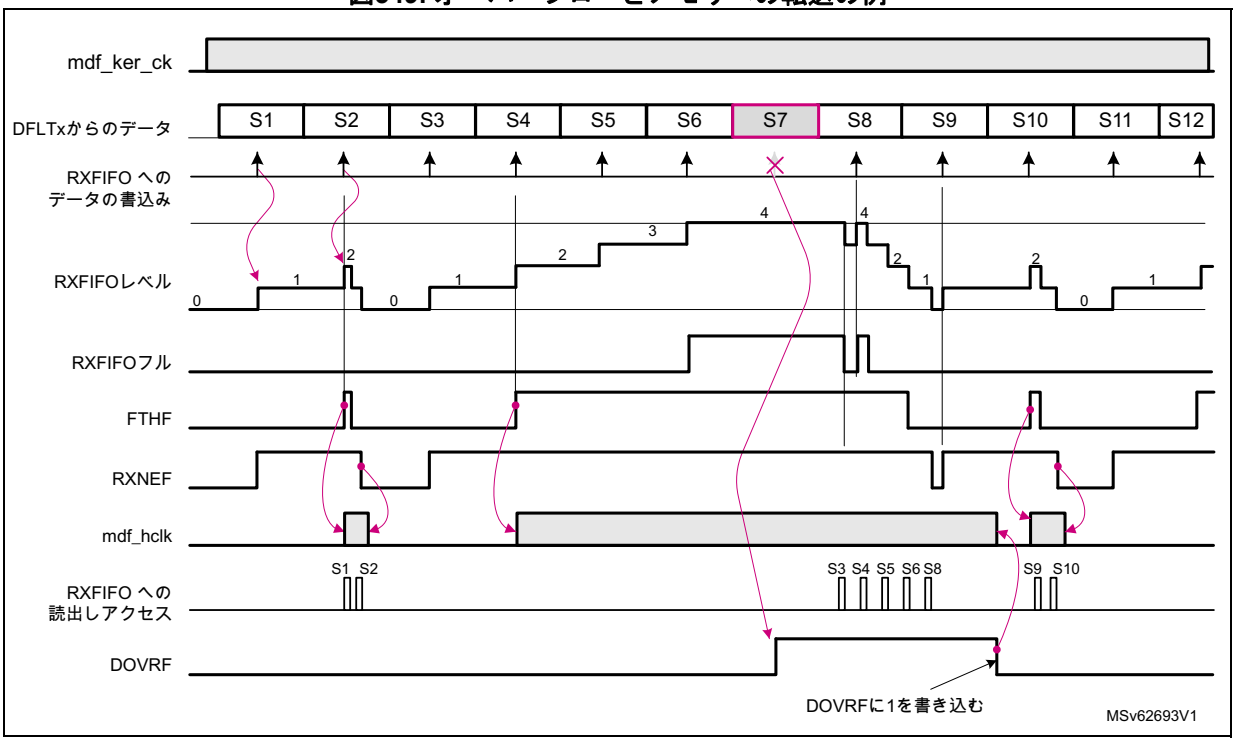
S7サンプルはオーバーランにより失われます。S7をRXFIFOに書き込む必要があるときにRXFIFOはフルです。転送S7書込み操作は行われません。書込み操作が予期された瞬間に、DOVRFに1がセットされます。アプリケーションによってクリアされない限り､オーバーフロー･イベントは1のままです。

この例では、オーバーラン状態が検出された場合に割込みが生成されるように、DOVRIEに1がセットされます。

S7サンプルの後に、アプリケーションによりRXFIFOからのデータの読出しが行われ、MDFではS8サンプル以降を書き込むことができます。その後、アプリケーションでDOVRがクリアされ、新しいオーバーラン状況の検出が可能になります。

mdf_hclkラインで、グレーのボックスはMDFによりAHBクロックがリクエストされたことを示します。次の図に、MDFによってリクエストされたときのみ利用可能なAHBクロックを示します。実際には、AHBクロックはMDFによってリクエストされなくても存在することがあります。

図349. オーバーフローとメモリへの転送の例



注： MDFがインタリーブ転送モードで動作している場合、アプリケーションではインタリーブ転送モードで動作するすべてのRXFIFOのオーバーラン･ステータスを確認する必要があります。

39.4.14 自律モード

MDFは、AHBバス･クロックが利用できない場合でも動作可能です（STOPモード）。MDFでは、レジスタ･インタフェース用にのみAHBクロックを使用します。処理部分はすべて、カーネル･クロックによりクロック供給されます。

STOPモードでは、次の条件が満たされた場合に、MDFはカーネル･クロックを受信します。

- MDF自律モードがRCCで有効化されている。
- 選択したカーネル･クロック･ソースが、STOPモードで利用可能なオシレータから取得される。

STOPモードで、次の条件が満たされた場合にMDFはAHBクロックを受信します。

- MDF自律モードがRCCで有効化されている。
- MDFが次の状況で、AHBクロックをリクエストする。
  - MDFがDMAを通してメモリにデータを転送する必要がある場合

    製品がSTOPモードのままで、DMAを使用してSRAMにデータが直接転送されます。AHBクロック･リクエストは、DMA転送が完了するまで維持されます。

– MDFで割込みを生成する必要がある場合

アプリケーションからのアクションが必要なため、一般的に割込みにより、デバイスはSTOPモードからウェイクアップします。AHBクロックが利用可能になると、割込みが生成されます。AHBクロック･リクエストは、有効化された割込みフラグがアクティブである間、保持されます。正確に言うと、割込みを生成する必要があるときに、MDFによりAHBクロックがリクエストされます。

39.4.15 レジスタの保護

MDFには、無効な状況を回避するためのハードウェア保護が組み込まれています。次の表に、書込み保護されているフィールドとされていないフィールドのリストを示します。

表377.レジスタ保護のまとめ

レジスタ	書込み保護されていないフィールド	書込み保護されているフィールド	書込み保護の条件
MDFグローバル制御レジスタ（MDF_GCR）	TRGO	ILVNB[3:0]	DFLTACTIVE0 = 1
MDFMDFクロック･ジェネレータ制御レジスタ（MDF_CKGCR）	CKGDEN CCK0EN CCK1EN	PROCDIV[6:0]、CCKDIV[3:0]、CKGMOD、TRGSRC[3:0]、TRGSENS、CCK[1:0]DIR	CKGACTIVE = 1
MDFシリアル･インタフェース制御レジスタx (MDF_SITFxCR)	SITFEN	STH[4:0]、SITFMOD[1:0]、SCKSRC[1:0]	SITFACTIVEx = 1
MDFビットストリーム･マトリックス制御レジスタx（MDF_BSMXxCR）	-	BSSEL[4:0]	DFLTACTIVEx = 1 または SCDACTIVEx = 1 または OLDACTIVEx = 1
MDFデジタル･フィルタ制御レジスタx（MDF_DFLTxCR）	DFLTEN	NBDIS[7:0]、TRGSRC[3:0]、TRGSENS、FTH、DMAEN、SNPSFMT、ACQMOD[2:0]	DFLTACTIVEx = 1
MDFデジタル･フィルタ設定レジスタx（MDF_DFLTxCICR）	SCALE[5:0]	MCICD[8:0]、CICMOD[2:0]、DATSRC[1:0]	
MDF再形成フィルタ設定レジスタx（MDF_DFLTxRSFR）	-	すべてのフィールド	
MDF積分器設定レジスタx（MDF_DFLTxINTR）	-	すべてのフィールド	
MDF限界外検出回路制御レジスタx（MDF_OLDxCR）	OLDEN	ACICD[4:0]、ACICN[1:0]、THINB BKOLD[3:0]	OLDACTIVEx = 1
MDF OLDxの閾値下限レジスタx（MDF_OLDxTHLR）	-	すべてのフィールド	
MDF OLDxの閾値上限レジスタx（MDF_OLDxTHHR）	-	すべてのフィールド	
MDF遅延制御レジスタx（MDF_DLYxCR）	-	SKPDLY[6:0]	SKPBF = 1
MDF短絡検出回路の制御レジスタx（MDF_SCDxCR）	SCDEN	BKSCD[3:0]、SCDT[7:0]	SCDACTIVEx = 1
MDF DFLTx割込み有効化レジスタx（MDF_DFLTxIER）	すべてのフィールド	-	-

表377.レジスタ保護のまとめ（続き）

レジスタ	書込み保護されていないフィールド	書込み保護されているフィールド	書込み保護の条件
MDF DFLTx割込みステータス･レジスタx（MDF_DFLTxISR）	すべてのフィールド	-	-
MDFオフセット･エラー補正制御レジスタx（MDF_OECxCR）	OFFSET[25:0]	-	-

MDF処理はすべて、mdf_proc_ckクロック･ドメインで行われます。そのため、AHBクロック･ドメインとmdf_proc_ckクロック･ドメインの間の再同期化のために、MDFサブブロックの有効化と無効化には多少時間がかかることがあります。XXXACTIVEフラグを使用することにより、アプリケーションでは2つのクロック･ドメイン間の同期が完了したかどうかを確認できます。

書込み保護されたビットフィールドを変更するには、アプリケーションは次のシーケンスに従います。

1. サブブロックの有効化ビットに0をセットします。
2. 対応するフラグXXXACTIVEが0になるのを待ちます。
3. 必要なフィールドを変更します。
4. サブブロックの有効化ビットに1をセットします。

詳細については、各サブブロックの説明を参照してください。

39.5 MDF低電力モード

表378.低消費電力モードがMDFに与える影響

モード	説明
SLEEP	影響はありません。MDF割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	MDFレジスタの内容は保たれます。 MDFがSTOPモードで利用可能な内部オシレータによりクロック供給されている場合、MDFはアクティブなままです。割込みにより、デバイスはSTOPモードを終了します。
STANDBY	MDFはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. MDFでサポートされているSTOPモードについての詳細はセクション 39.3：MDFの実装を参照してください。

39.6 MDF割込み

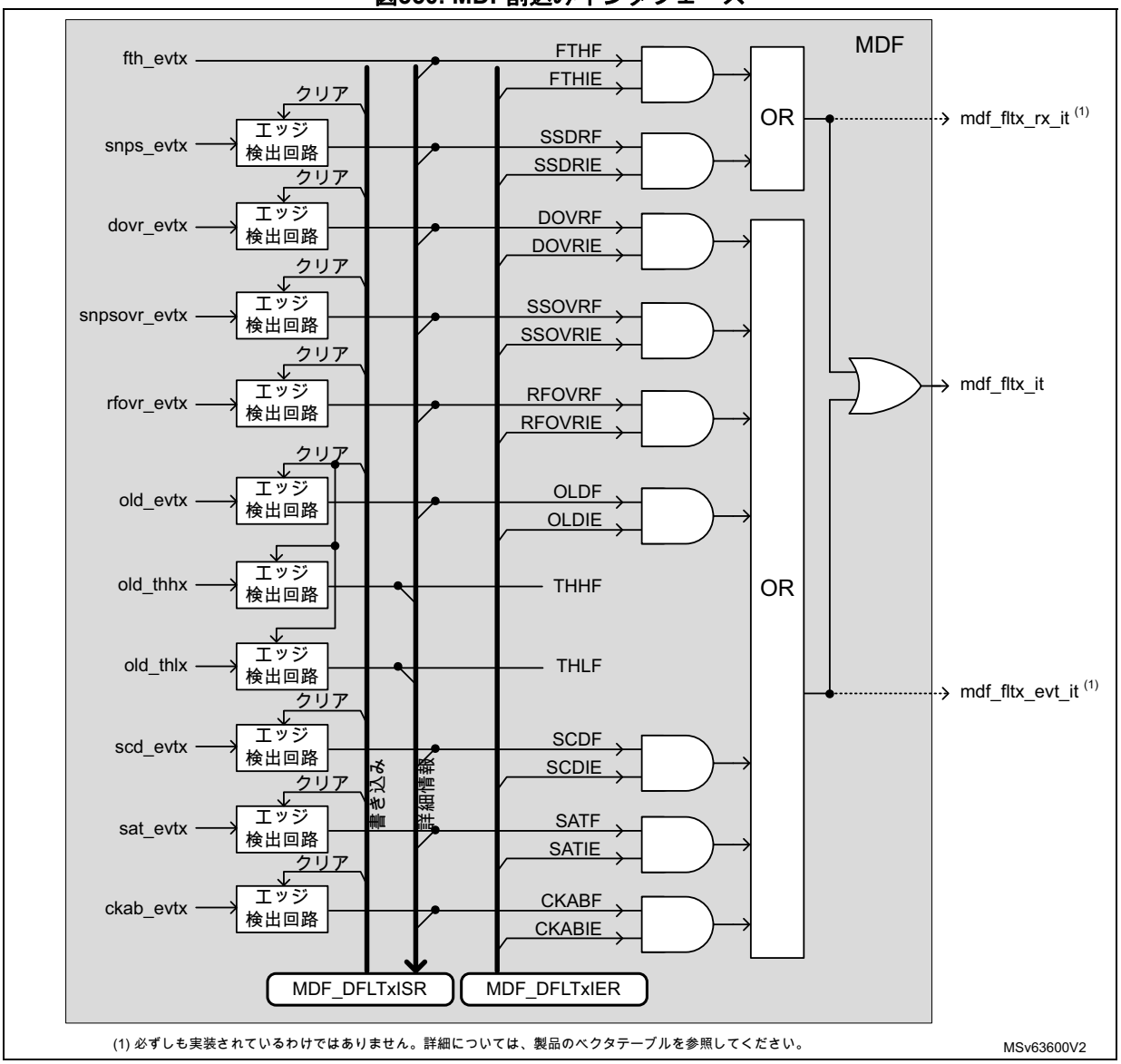
CPUの性能を高めるため、MDFにはデジタル･フィルタごとに次の割込みラインが用意されています。

- 受信割込みMDF_FLTx_RX（mdf_fltx_rx_it）
- イベント割込みMDF_FLTx_EVT（mdf_fltx_evt_it）
- 両方の割込みを組み合わせたものMDF_FLTx（mdf_fltx_it）

注： **割込みが常にデバイスに接続されるわけではありません（詳細はセクション 39.3：MDFの実装を参照）。ステータス･フラグは、対応する割込み有効化フラグが有効化されていない場合でも利用できます。**

割込みインタフェースは MDF DFLTx割込み有効化レジスタx（MDF_DFLTxIER）および MDF DFLTx割込みステータス･レジスタx（MDF_DFLTxISR）を通して制御されます。

図350. MDF割込みインタフェース



次の表に、どのイベントがどの割込みに影響するかと、イベント／割込みのクリアと有効化の方法を示します。

表379.MDF割込みリクエスト

割込みベクタ		割込みイベント	イベントフラグ	イベント／割込みのクリア方法	SLEEPモードの終了	STOPモードの終了(1)	STANDBYモードの終了
MDF_FLTx(2)	MDF_FLTx_RX(3)	RXFIFO閾値に達した	FTHF	RXFIFOレベルが閾値を下回るまでMDF_DFLTxDRを読み出します。	あり	あり	なし
		スナップショット･データ準備完了	SSDRF	SSDRFに1を書き込みます。			
	MDF_FLTx_EVT(4)	スナップショット･データのオーバーラン	SSOVRF	SSOVRFに1を書き込みます。			
		RXFIFOオーバーランです。	DOVRF	DOVRFに1を書き込みます。			
		RSFLTオーバーランです。	RFOVRF	RFOVRFに1を書き込みます。			
		短絡検出回路	SCDF	SCDFに1を書き込みます。			
		飽和の検出	SATF	SATFに1を書き込みます。			
		チャネル･クロックなし検出	CKABF	CKABFに1を書き込みます。			
		限界外検出回路	OLDF	OLDFに1を書き込みます。			
-	-		THHF	OLDFに1を書き込みます。	-	-	-
-	-		THLF	OLDFに1を書き込みます。	-	-	-

1. 詳細については、セクション 39.3：MDFの実装を参照してください。
2. MDF_FLTxベクタはmdf_fltx_it信号のアサートに相当します。
3. MDF_FLTxベクタはmdf_fltx_rx信号のアサートに相当します。
4. MDF_FLTxベクタはmdf_fltx_evt信号のアサートに相当します。

39.7 MDFアプリケーション情報

39.7.1 オーディオ･キャプチャ用のMDF設定の例

表380 に、16kHzおよび48kHzの出力データ･レートに焦点を当てた、デジタル･マイクロフォンのMDF設定の例を示します。これらの例では、次のことが前提となります。

- INTブロックがバイパスされていること（INTVAL = 0）
- オフセット･エラー補正が無効化されていること（OFFSET = 0）

設定 #1と #2は非常に消費電力の少ないユースケース用で、S/N比は低くなっています。ユーザは、選択したデジタルマイクロフォンが512kHzで正しく動作することも確認する必要があります。これらの設定はサウンド検出に使用できます。RSFLTは、カーネル･クロック（mdf_ker_ck）の周波数をできるだけ抑えるために使用するものではありません。

設定 #3、#4、#9、#10、#11は、理想的なマイクロフォンのモデルと997Hzの正弦波信号を使用した、約115dBでのS/N比を示しています。RSFLTを使用して、良好な帯域内リップルの制御と良好なイメージ周波数除去を行うことができます。

設定 #7、#8、#10は、理想的なマイクロフォンのモデルと997Hzの正弦波信号を使用した、約120dBでのS/N比を示しています。

表380.マイクロ･フォンキャプチャ用のMDF設定の例

設定	mdf_ker_ck（MHz）	PROCDIV + 1	CCKDIV + 1	CICの次数 (1)	MCICD + 1	SCALE	RSFLTBYP	RSFLTD	HPFBYP	-	mdf_proc_ck（MHz）	総デシメーション比	F_{RS}（kHz）	F_{MDF_CCKx}（MHz）	F_{PCM}（kHz）
1	1.024	1	2	4	64	0x2D（-8.5dB）	1	x	x	=>	1.024	64	-	0.512	8
2		1	2	5	32	0x2B（-14.5dB）						32	-		16
3		1	2	5	16	0x01（+3.5dB）	0	0				64	32		8
4	2.048	1	2	5	16	0x01（+3.5dB）	0	0			2.048	64	64	1.024	16
5	3.072	1	6	5	8	0x0B（33.6dB）	0	0			3.072	32	32	0.512	8
6		2	2	5	12	0x06（18.1dB）						48	64	0.768	16
7		1	2	5	24	0x2C（-12dB）						96		1.536	
8	4.096	1	2	5	32	0x27（-26.6dB）	0	0			4.096	128	64	2.048	16
9	6.144	3	2	5	16	0x02（6.0dB）	0	0			2.048	64	64	1.024	16
10		2	2	5	24	0x2C（-12dB）					3.072	96		1.536	
11		1	2	5	16	0x01（+3.5dB）					6.144	64	192	3.072	48
12	7.680	1	2	5	20	0x2E（-6.0dB）	0	0			7.680	80	192	3.840	48

1. CICの次数が4の場合、CICMOD = 100。CICの次数が5の場合、CICMOD = 101。

39.7.2 プログラミングの例

この例では、それぞれのマイクロフォンのペアで同じデータ･ラインが共有されていることを前提として、4本のマイクロフォンからサウンドをキャプチャする方法を説明します。MDF_SD0およびMDF_SD2データ･ラインが使用されます。マイクロフォンのクロックは、MDF_CCK0ピンを通してMDFによって供給されます。

表381.プログラミング･シーケンス

動作	コメント
RCCブロックを通してカーネル･クロック周波数を調整	12.288MHzのカーネル･クロック（mdf_ker_ck）が供給されるようにRCCがプログラムされていることが前提となります。
RCCブロックを通して適切なMDFカーネル･クロック･ソースを選択	製品のRCCを参照してください。
RCCブロックを通してMDFクロックを有効化	製品のRCCを参照してください。
RCCブロックを通してMDFをリセット	製品のRCCを参照してください。
AFMUXプログラミング	MDF_SD0、MDF_SD2、およびMDF_CCK0機能を選択するようAFMUXをプログラムします。
MDF処理クロックの有効化： MDF_CKGCR = 0x0103 0023	PROCDIV = 1（2で分周）: mdf_proc_ck周波数は6.144MHz。 CCKDIV = 3（4で分周）: MDF_CCK0クロック周波数は1.536MHz。 マイクロフォンが低消費電力モードを終了できるように、MDF_CCK0パッドは出力にセットされ、クロックを生成します。

表381.プログラミング･シーケンス （続き）

動作	コメント
シリアル･インタフェース設定： MDF_SITF0CR = 0x0000 1F01 MDF_SITF2CR = 0x0000 1F01	MDF_CCK0をシリアル･クロックとして選択するため、SCKSRC = 0。 LF_MASTERモードを選択するため、SIFTMOD = 0。 クロックなし機能はこのモードでは動作しません。 シリアル･インタフェースは有効です。
ビットストリーム･マトリックス設定： MDF_BSMX0CR = 0x0000 0000 MDF_BSMX1CR = 0x0000 0001 MDF_BSMX2CR = 0x0000 0004 MDF_BSMX3CR = 0x0000 0005	DFLT0フィルタはSITF0のビットストリームを取得(立ち上がりエッジでサンプリング)。 DFLT1フィルタはSITF0のビットストリームを取得(立ち下がりエッジでサンプリング)。 DFLT2フィルタはSITF2のビットストリームを取得(立ち上がりエッジでサンプリング)。 DFLT3フィルタはSITF2のビットストリームを取得(立ち下がりエッジでサンプリング)。
フィルタ設定（CIC）： MDF_DFLT0CICR = 0x02C0 1750 MDF_DFLT1CICR = 0x02C0 1750 MDF_DFLT2CICR = 0x02C0 1750 MDF_DFLT3CICR = 0x02C0 1750	飽和を回避するため、SCALE = 0x2C（- 12dB） MCICD = 0x17（24でデシメーション） $Sinc^5$ を選択するため、CICMOD = 5 BSMXから受信するデータを選択するため、DATSCR = 0
フィルタ設定（RSFLTおよびHPF）： MDF_DFLT0RSFR = 0x0000 0100 MDF_DFLT1RSFR = 0x0000 0100 MDF_DFLT2RSFR = 0x0000 0100 MDF_DFLT3RSFR = 0x0000 0100	HPFC = 1：16kHzの遮断周波数 * 0.00125 = 20Hz HPFBYP = 0：HPFをバイパスしない RSFLTD = 0：RSFLTを4でデシメーション RSFLTBYP = 0：RSFLTをバイパスしない
フィルタ設定（INT）： MDF_DFLT0INTR = 0x0000 0000 MDF_DFLT1INTR = 0x0000 0000 MDF_DFLT2INTR = 0x0000 0000 MDF_DFLT3INTR = 0x0000 0000	INTVAL = 0：INTフィルタを使用しない その他のパラメータは重要ではありません。
遅延の微調整： MDF_DLY0CR = 0x0000 0005 MDF_DLY0CR = 0x0000 0012 MDF_DLY0CR = 0x0000 0023 MDF_DLY0CR = 0x0000 0000	各マイクロフォンの遅延の初期微調整（値は一例）
オフセット･エラー訂正： MDF_OEC0CR = 0x0000 0000 MDF_OEC1CR = 0x0000 0000 MDF_OEC2CR = 0x0000 0000 MDF_OEC3CR = 0x0000 0000	訂正なし。DCオフセットはHPFによって除去されます。
短絡検出： MDF_SCD0CR = 0x0000 0000 MDF_SCD1CR = 0x0000 0000 MDF_SCD2CR = 0x0000 0000 MDF_SCD3CR = 0x0000 0000	各フィルタのSCD機能は無効です。

表381.プログラミング･シーケンス （続き）

動作	コメント
割込みイベントの有効化： MDF_DFLT0IER = 0x0000 0202 MDF_DFLT1IER = 0x0000 0202 MDF_DFLT2IER = 0x0000 0202 MDF_DFLT3IER = 0x0000 0202	アプリケーションで扱う割込みイベントを有効化します。 この例では、飽和またはデータのオーバーフローが発生した場合に割込みが生成されるように、ATIEおよびDOVRIEビットに1がセットされます。
デジタル･フィルタの制御： MDF_DFLT0CR = 0x0000 0027 MDF_DFLT1CR = 0x0000 0027 MDF_DFLT2CR = 0x0000 0027 MDF_DFLT3CR = 0x0000 0027 フィルタ用にDFLTACTIVE = 0を待つ	NBDIS = 0：サンプルの破棄なし TRGSRC = 0：TRGOをトリガ･ソースとして選択 TRGSENS = 0：立ち上がりエッジでトリガ ACQMOD = 2：同期連続取得モード DMAEN = 1：DMAインタフェースが有効 DFLTEN = 1：デジタル･フィルタが有効
ステータス･フラグのクリア： MDF_DFLT0ISR = 0x0000 0FFF MDF_DFLT1ISR = 0x0000 0FFF MDF_DFLT2ISR = 0x0000 0FFF MDF_DFLT3ISR = 0x0000 0FFF	フィルタの実行前にステータス･フラグをすべてクリアします。
DMAのプログラム DMAを有効にします。	DMAリクエストが生成されるたびにMDF_DFLT0DR内部でデータを読み出すには、DMAをプログラムする必要があります。 ただし、MDFがインタリーブ取得モードで動作している場合、フィルタ0、1、2、および3のデータはMDF_DFLT0DRレジスタを通して読み出されます。
取得の開始： MDF_DFLT0CICR = 0x0000 0031	ILVNB = 3：DFLT0、1、2、および3でのインタリーブ･モード TRGO立ち上がりエッジ･イベントを待っているすべてのフィルタの取得をトリガするには、TRGO = 1

39.7.3 接続の例

図351 に、MDFを外部センサに接続する簡単な例を示します。

- 図の左側：MDFに接続された4本のデジタル･マイクロフォン

  この接続では、DMIC1 は単体での使用が想定されています。そのため、専用のクロック（MDF_CCK0）を使用します。DMIC1は、DMIC1のみを使用し、DMIC2/3/4をパワーダウンするする場合に備えて、専用のデータ･ラインにも接続されます。DMIC2/3/4への電力供給を維持する場合、DMIC2とDMIC1で同じデータ･ラインを共有し、4つのI/Oへの接続を低減することができます。

  使用するマイクロフォンが1本で、電力消費が問題にならない場合、すべてのマイクロフォンを共通のクロック（MDF_CCK0）に接続し、各マイクロフォンのペアで同じデータ･リンクを共有することができます。この場合、必要なI/Oは3つだけです。

  この例では、メモリへのデータ転送は、インタリーブ転送モードまたは独立転送モードで行えます。

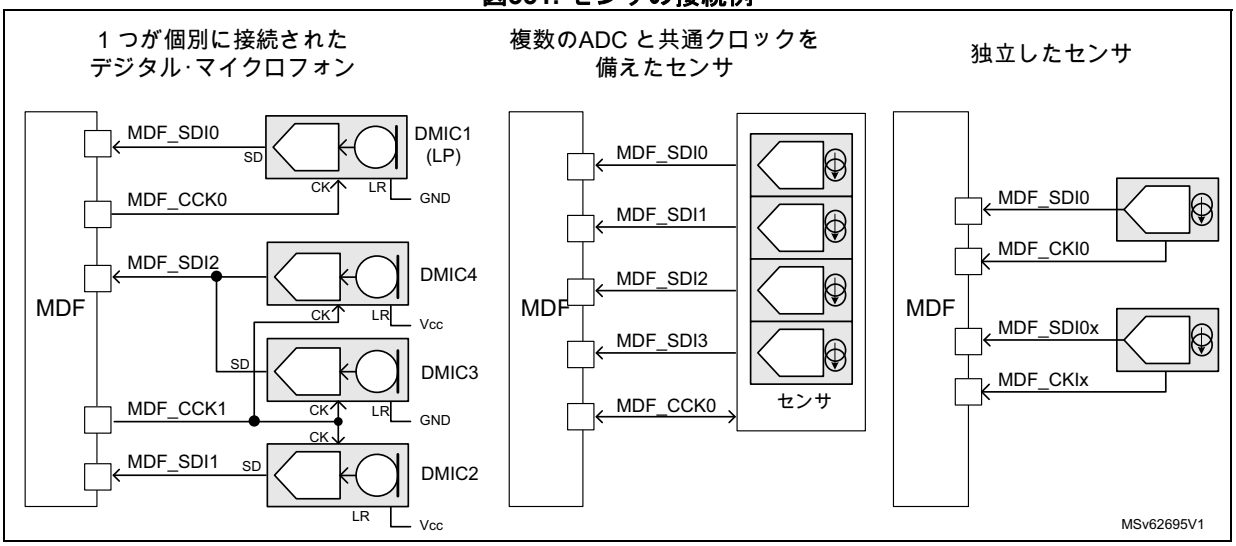
- 図の中央：共通クロックを使用して複数のデータ･リンクを提供するセンサ

  各データ･ラインでは異なるパラメータ（電流や電圧など）を表わすことができます。センサまたはMDFによって共通のクロックを供給することができます。データ･ラインは、センサで許可されていれば、2つのセンサで共有することもできます。次の図では、センサでデータ･ラインの共有は許可されていません。

  この例では、メモリへのデータ転送は、インタリーブ転送モードまたは独立転送モードで行えます。

- 図の右側：MDFに接続された2つの独立したセンサ
  それぞれが専用のクロックとデータ･ラインを備えています。この場合、メモリへのデータ転送は独立転送モードで行う必要があります。

図351. センサの接続例



39.7.4 グローバル周波数応答

図352 に、1.024Mhzで動作するデジタル･マイクロフォンを使用した16kHzのオーディオ信号のグローバル周波数応答を示します。フィルタ設定は、次のとおりです。

- CICの次数4または5、デシメーション比は16
- RSFLT有効、デシメーション比は4
- HPF有効、遮断周波数は40Hz

次の図に、CIC4とCIC5を使用した理論上の周波数応答を示します。

図352. グローバル周波数応答

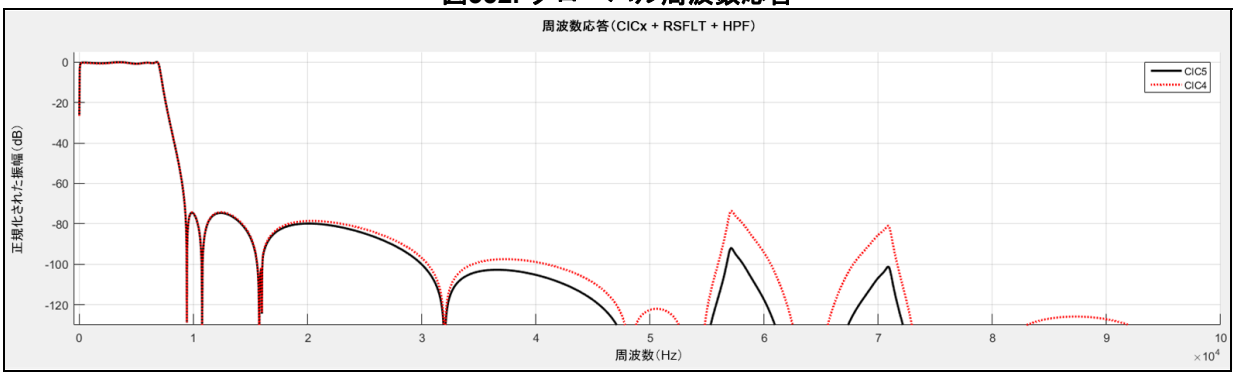


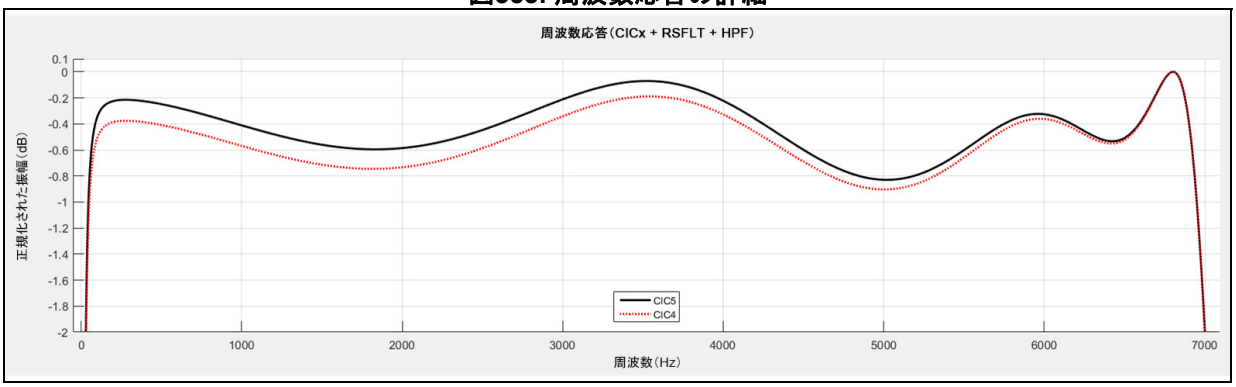
図353 に、1.024Mhzで動作するデジタル･マイクロフォンを使用した16kHzのオーディオ信号の帯域内リップルを示します。フィルタ設定は、次のとおりです。

- CICの次数4または5、デシメーション比は16
- RSFLT有効、デシメーション比は4
- HPF有効、遮断周波数は20Hz

発生する帯域内リップルは、CIC5では ± 0.41dB、CIC4では ± 0.45となります。

-3dBの遮断周波数は7061Hzです。

図353. 周波数応答の詳細



39.7.5 総MDFゲイン

このセクションでは、フィルタ設定に従って、MDFによって提供される信号のレベルを計算する方法を説明します。この式では、フィルタ転送機能は考慮しません。

信号のレベルは、dBFS単位（デシベルの最大測定範囲）で表現できます。0dBFSのレベルは、許容される最大のデジタル･レベルに割り当てられます。たとえば、最大レベルの50 % に達した信号は、- 6dBFSのレベル（最大測定範囲を6dB下回る）となります。

たとえば、最終データ幅が24bitのMDFの場合、振幅が2 * 10^6 LSBの信号のレベルは次のようになります。

$$20 \times \log_{10}\left(\frac{2 \times 10^6}{2^{(24-1)}}\right) = -12.45 \text{ dBFS}$$

また、LSB単位で表現される振幅（Amp）を持つ信号のデータ･サイズは、次の式で算出できます。

$$DS = \left(\frac{\ln(Amp)}{\ln(2)} + 1\right) \text{ビット}$$

負の値の場合、1bit追加する必要があります。

そのため、振幅が2 * 10^6 LSBの信号のデータ･サイズは21.9bitです。

CICゲイン

CICゲイン（G_{CIC} およびGdB_{CIC}）は、データ･サイズをビット単位として（DS_{CIC}）、次の式から推定することができます。

$$DS_{CIC} = (N \times \log_2(D1)) + DS_{IN}$$

また、ビット幅は次のようになります。

$$BG_{CIC} = (N \times \log_2(D1))$$

ここで、NはCICの次数（CICMOD[2:0] により選択）、D1はデシメーション比（MCICDにより指定）を表します。

DS_{IN} は入力信号のデータ･サイズ（ビット単位）を表します。

警告： DS_{CIC} はCICフィルタにとって非常に重要です。正しく動作するには、DS_{CIC} が26bitを超えないようにする必要があります。

$$G_{CIC} = 2^{(BG_{CIC})} = (D1)^N$$

デシベル単位では次のようになります。

$$GdB_{CIC} = 20 \times \log_{10}((D1)^N)$$

注： これらの式はACICに対しても有効です。

SCALE出力でのデータ･サイズ

SCALE出力でのピークツーピークの信号振幅が22bitより大きくなるとRSFLTが飽和し始めるため、SCALE出力でのデータ･サイズ（CICゲインを含めて）は重要な情報です。

RSFLTがバイパスされる場合、24bitのピークツーピークの信号振幅が受容されます。実際のデータ･サイズは次の式で算出できます。

$$DS_{SCALE} = N \times \log_2(D1) + \log_2\left(10^{\frac{GDB_{SCALE}}{20}}\right) + DS_{IN}$$

SCALE出力でのデータ･サイズ（DS_{SCALE}）はビット単位で表され、GdB_{SCALE} はSCALE[5:0] により選択されるdB単位のゲインを表します。

RSFLTゲイン

有益な帯域幅でのRSFLTゲインは通常は9.5dBですが、リップルのために、約 ± 0.5dBのマージンを考慮する必要があります。通常、RSFLTではビットサイズがBG_{RSFLT} ずつ増加します。

$$BG_{RSFLT} = 10^{\frac{9.5\ dB}{20}} = 2.98 = 1.6\ ビット$$

INTゲイン

INTブロックでは、スケール調整値が積算値と異なる場合もゲインを導入できます。

$$G_{INT} = \frac{IVAL}{IDIV}$$

また

$$GdB_{INT} = 20 \times \log_{10}\left(\frac{IVAL}{IDIV}\right)$$

INTのビットの増加は、次の式によって算出できます。

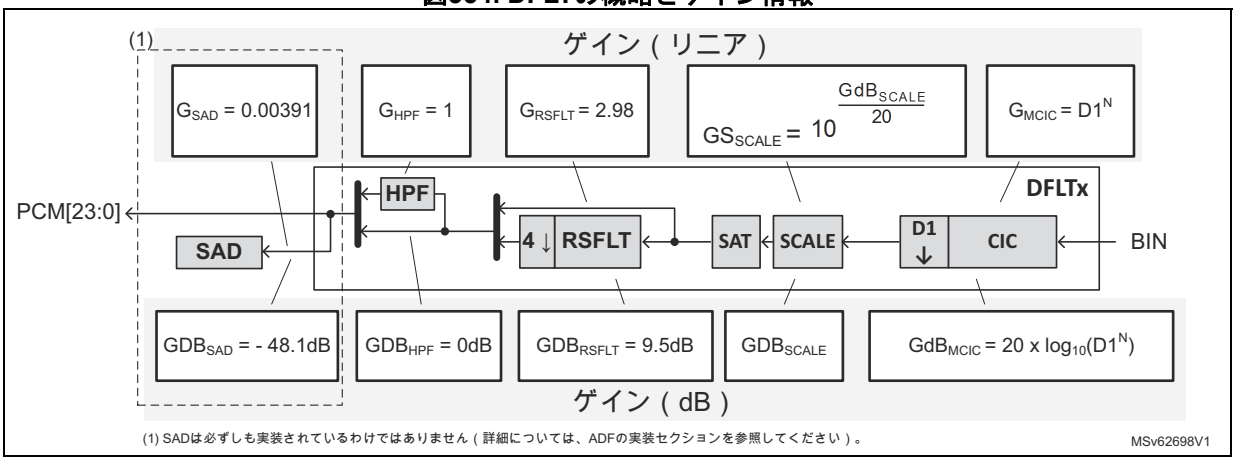
$$BG_{INT} = \log_2\left(\frac{IVAL}{IDIV}\right)$$

IVALはINTVAL[6:0] によって選択される積算値を表し、IDIVはINTDIV[1:0] によって選択される積分器の出力の分周比を表します。

注： **HPFフィルタのゲインは0dBです。**

次の図にフィルタ･パスの概略を示し、重要なコンポーネントについては、ビットの増加とゲインの式を示します。

図354. DFLTの概略とゲイン情報



次の表に、フィルタ設定ごとの最終的なデータ･サイズをまとめます。

表382.出力信号レベル

フィルタ設定	最終的な信号レベル（dBFS）	最終的な信号サイズ（ビット）
CIC + RSFLT（+ HPF）	$SdB_{OUT} = 20 \times \log_{10}\left(\frac{2^{DS_{OUT}}}{2^{24}}\right)$	$DS_{OUT} = DS_{SCALE} + 1.6$ ビット DS_{SCALE} は22bitより小さくする必要があります。
CIC + RSFLT（+ HPF）+ INT		$DS_{OUT} = DS_{SCALE} + BG_{INT} + 1.6$ ビット DS_{SCALE} は22bitより小さくする必要があります。
CIC（+ HPF）+ INT		$DS_{OUT} = DS_{SCALE} + BG_{INT}$ DS_{SCALE} は24bitより小さくする必要があります。

メイン･フィルタ･チェーンを使用する例

MDFフィルタが次のようにプログラムされている場合：

- 入力信号をシリアル･インタフェースから受信する（DS_{IN} = 1bit）。
- CICの次数 = 5（N）で、デシメーション値が24（D1）。
- SCALE[5:0] に -12dBをセット。
- RSFLTが有効で、4でのデシメーションが有効。
- HPFが有効。
- INTは無効です。

まず、CIC出力でのデータ･サイズを確認してください。

$$DS_{CIC} = (5 \times \log_2(24)) + 1 \text{ bit} = 23.92 \text{ ビット}$$

サイズが26bit未満であれば、CICは正しく動作します。

CIC出力のデータ･サイズは24bitに非常に近くなるため、RSFLTに最大22bitの信号を提供するには、SCALEを調整する必要があります。12dBの減衰が必要です。

そして、RSFLTに提供される信号のレベルは次のようになります。

$$DS_{SCALE} = DS_{CIC} + \log_2\left(10^{\frac{-12}{20}}\right) = 23.92 - 1.99 = 21.93 \text{ ビット}$$

最後に、最終的な信号振幅は次のようになります。

$$DS_{OUT} = DS_{SCALE} + 1.6 \text{ ビット} = (21.93 + 1.6) = 23.52 \text{ ビット}$$

または

$$SDB_{OUT} = 20 \times \log_{10}\left(\frac{2^{23.52}}{2^{24}}\right) = -2.84 \text{ dBFS}$$

RSFLTリップルでは、約 ± 0.41dBのマージン（この設定で）を考慮する必要があります。

OLDフィルタ･チェーンを使用する例

次の例では、シャント抵抗からの電圧が ± 200mVに達すると、アプリケーションによりOLDイベントがトリガされます。

前提：

- ± 350mVの最大測定範囲の信号に対応するよう ΣΔ センサを指定。
- ACICのデシメーション比（D2）を32に固定。
- ACICの次数（N）を3に固定。

ΣΔ センサは、± 350mV以上の信号に対して、最大測定範囲のデジタル信号（1bitの振幅）を提供します。入力信号が200mVの場合、センサによって提供されるデジタル信号の振幅は200 / 350 = 0.571となります。

ACICフィルタのゲインは次のようになります。

$G_{CIC} = (D2)^N = 32^3 = 32768$

センサ入力での振幅が200mVの信号の場合、ACIC出力での振幅は約32768 * 0.571 = 18725 LSBになります。

OLDTHHに18725をセットし、OLDTHLに -18725をセットする必要があります。

39.8 MDFレジスタ

MDF レジスタへのアクセスは、ワード･フォーマット（32bit）またはハーフワード･フォーマット（16bit）で行う必要があります。

レジスタは、最大数のフィルタを持つMDFインスタンスについて説明しています。フィルタに関連するレジスタの場合、レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

39.8.1 MDFグローバル制御レジスタ（MDF_GCR）

アドレスオフセット：0x000

リセット値：0x0000 0000

このレジスタは、すべてのデジタル･フィルタに共通の制御部のために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ILVNB[3:0]				Res.	Res.	Res.	TRGO
								rw	rw	rw	rw				rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **ILVNB[3:0]**：インタリーブ番号

このビットは、ソフトウェアによってセット／リセットされます。インタリーブ転送モードを有効化または無効化し、このモードで動作可能なデジタル･フィルタの数を定義します。

MDF_DFLT0CRでDFLTEN = 1の場合、このビットフィールドは変更できません。

0000：インタリーブ転送モードは無効です。

0001：DFLT0およびDFLT1からのデータがインタリーブされます。

0010：DFLT0、DFLT1およびDFLT2からのデータがインタリーブされます。

...

1111：DFLT0 ~ DFLT15からのデータがインタリーブされます。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **TRGO**：トリガ出力の制御

このビットは、ソフトウェアによってセットされ、ハードウェアによってリセットされます。これは、複数のフィルタの取得を同期させて開始するために使用されます。mdf_trgo信号を制御することによって複数のMDFを同期させることもできます。

0：0を書き込んでも、ビットの値は変化しません。0が読み出された場合、トリガに再び1をセットできることを意味します。

1：1を書き込むと、mdf_trgo信号で正のパルスが生成され、ACQMOD[2:0] = 0x1とし、TRGOをトリガとして選択して、有効なフィルタでの取得がトリガされます。1が読み出された場合、トリガ･パルスがアクティブなままであることを意味します。

39.8.2 MDFMDFクロック･ジェネレータ制御レジスタ（MDF_CKGCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

このレジスタは、クロック･ジェネレータを制御するために使用されます。他のMDF部分を有効化する前に、mdf_proc_ckクロックを有効化する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CKGACTIVE	PROCDIV[6:0]							Res.	Res.	Res.	Res.	CCKDIV[3:0]			
r	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRGSRC[3:0]				Res.	Res.	Res.	TRGSENS	Res.	CCK1DIR	CCK0DIR	CKGMOD	Res.	CCK1EN	CCK0EN	CKGDEN
rw	rw	rw	rw				rw		rw	rw	rw		rw	rw	rw

ビット31 **CKGACTIVE**：クロック･ジェネレータのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。このフラグは、クロック･ジェネレータが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用される必要があります。この機能の保護されたフィールドを更新できるのは、CKGACTIVE = 0のときのみです（詳細はセクション39.4.15を参照）。CKGDENでの遷移とCKGACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のmdf_proc_ckです。

0：クロック･ジェネレータはアクティブでなく、必要に応じて設定できます。

1：クロック･ジェネレータはアクティブで、保護されたフィールドは設定できません。

ビット30:24 **PROCDIV[6:0]**：シリアル･インタフェース･クロックを制御する分周回路

このビットは、ソフトウェアによってセット／リセットされます。これは、SITFに提供されるクロックの周波数を調整するために使用されます。

$$F_{mdf_itf_ck} = \frac{F_{mdf_ker_ck}}{(PROCDIV + 1)}$$

いずれかのフィルタが有効化されている（DFTEN = 1）場合、このビットフィールドは変更しないでください。

0：SITFに供給されるmdf_ker_ck

1：SITFに供給されるmdf_ker_ck/2

2：SITFに供給されるmdf_ker_ck/3

...

127：SITFに供給されるmdf_ker_ck/128

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **CCKDIV[3:0]**：MDF_CCKクロックを制御する分周回路

このビットは、ソフトウェアによってセット／リセットされます。これは、SMDF_CCKクロックの周波数を調整するために使用されます。この分周回路の入力 クロックはSITFに供給されるクロックです。包括的に言うと、MDF_CCKの周波数は、次の式で算出できます。

$$F_{MDF_CCK} = \frac{F_{mdf_ker_ck}}{(PROCDIV + 1) \times (CCKDIV + 1)}$$

いずれかのフィルタが有効化されている（DFTEN = 1）場合、このビットフィールドは変更しないでください。

0000：MDF_CCKクロックはmdf_proc_ckです。

0001：MDF_CCKクロックはmdf_proc_ck / 2です。

0010：MDF_CCKクロックはmdf_proc_ck / 3です。

...

1111：MDF_CCKクロックはmdf_proc_ck / 16です。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット15:12 **TRGSRC[3:0]**：デジタル･フィルタ･トリガ信号の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、どの外部信号により対応するフィルタをトリガするかを選択するのに使用されます。このビットフィールドは、CKGMOD = 0の場合は重要ではありません。

000x：TRGOが選択されています。

0010：mdf_trg[0] が選択されています。

0011：mdf_trg[1] が選択されています。

...

1111：mdf_trg[13] が選択されています。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **TRGSENS**：CKGENトリガの検知方向の選択

このビットは、ソフトウェアによってセット／クリアされます。これは、トリガ信号のトリガ感度を選択するのに使用されます。このビットは、CKGMOD = 0の場合は重要ではありません。

0：立ち上がりエッジ･イベントによりCKGEN分周回路の有効化がトリガされます。

1：立ち下がりエッジ･イベントによりCKGEN分周回路の有効化がトリガされます。

注： トリガ･ソースがTRGOの場合、TRGSENS値は考慮されません。TRGOを選択すると、検知方向は強制的に立ち下がりエッジになります。このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CCK1DIR**：MDF_CCK1の方向

このビットは、ソフトウェアによってセット／リセットされます。これは、MDF_CCK1ピンの方向を制御するために使用されます。

0：MDF_CCK1ピンの方向は入力です。

1：MDF_CCK1ピンの方向は出力です。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット5 **CCK0DIR**：MDF_CCK0の方向

このビットは、ソフトウェアによってセット／リセットされます。これは、MDF_CCK0の方向を制御するために使用されます。

0：MDF_CCK0ピンの方向は入力です。

1：MDF_CCK0ピンの方向は出力です。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット4 **CKGMOD**：クロック･ジェネレータのモード

このビットは、ソフトウェアによってセット／リセットされます。これは、クロック･ジェネレータの有効化方法を定義するために使用されます。いずれかのフィルタが有効化されている（DFTEN = 1）場合、このビットは変更しないでください。

0：カーネル･クロックは、CKGDENに1がセットされるとすぐに分周回路に供給されます。

1：カーネル･クロックは、CKGDENに1がセットされ、トリガ条件が満たされたときに、分周回路に供給されます。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CCK1EN**：MDF_CCK1クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。これは、MDF_CCK1ピンでのビットストリーム･クロックの生成を制御するために使用されます。

0：ビットストリーム･クロックが生成されません。

1：MDF_CCK1パッドでビットストリーム･クロックが生成されます。

ビット1 **CCK0EN**：MDF_CCK0クロックの有効化

このビットは、ソフトウェアによってセット／リセットされます。これは、MDF_CCK0ピンでのビットストリーム･クロックの生成を制御するために使用されます。

0：ビットストリーム･クロックが生成されません。

1：MDF_CCK0パッドでビットストリーム･クロックが生成されます。

ビット0 **CKGDEN**：CKGEN分周回路の有効化

このビットは、ソフトウェアによってセット／リセットされます。これは、CKGENのクロック分周回路、PROCDIVおよびCCKDIVを有効化／無効化するために使用されます。

0：CKGEN分周回路は無効です。

1：CKGEN分周回路は有効です。

39.8.3 MDFシリアル･インタフェース制御レジスタx (MDF_SITFxCR)

アドレスオフセット：0x080 + 0x80 * x（x = 0から5）

リセット値：0x0000：1F00

このレジスタは、シリアル･インタフェース（SITFx）を制御するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SITFACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	STH[4:0]					Res.	Res.	SITFMOD[1:0]		Res.	SCKSRC[1:0]		SITFEN
			rw	rw	rw	rw	rw			rw	rw		rw	rw	rw

ビット31 **SITFACTIVE**：シリアル･インタフェースのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、シリアル･インタフェースが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用されます。この機能の保護されたフィールドを更新できるのは、SITFACTIVEに0がセットされているときのみです（詳細はセクション39.4.15を参照）。

SITFENでの遷移とSITFACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のmdf_proc_ckです。

0：シリアル･インタフェースはアクティブでなく、必要に応じて設定できます。

1：シリアル･インタフェースはアクティブで、保護されたフィールドは設定できません。

ビット30:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **STH[4:0]**:：マンチェスタ･シンボル閾値／SPI閾値

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、想定されるシンボルの閾値レベルを定義するために、マンチェスタ･モードで使用されます（計算の詳細はマンチェスタ･モードを参照）。

また、このビットフィールドは、通常SPIモードでクロックなし検出のタイムアウト値を定義するために使用されます。4未満のSTH[4:0] 値は無効です。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **SITFMOD[1:0]**：シリアル･インタフェースのタイプ

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、シリアル･インタフェースのタイプを定義するために使用されます。

00：LF_MASTER SPIモード

01：通常SPIモード

10：マンチェスタ･モード：立ち上がりエッジ = ロジック0、立ち下がりエッジ = ロジック1

11：マンチェスタ･モード：立ち上がりエッジ = ロジック1、立ち下がりエッジ = ロジック0

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:1 **SCKSRC[1:0]**：クロック･ソースの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、シリアルインタフェースのクロックソースを選択するために使用されます。

00：シリアル･クロック･ソースはMDF_CCK0です。

01：シリアル･クロック･ソースはMDF_CCK1です。

1x：シリアル･クロック･ソースはMDF_CKIxです（LF_MASTER SPIモードでは使用できません）

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット0 **SITFEN**：シリアル･インタフェースの有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、シリアル･インタフェースを有効化／無効するために使用されます。

0：シリアル･インタフェースは無効です。

1：シリアル･インタフェースは有効です。

39.8.4 MDFビットストリーム･マトリックス制御レジスタx（MDF_BSMXxCR）

アドレスオフセット：0x084 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、対応するデジタル･フィルタとSCDに提供するビットストリームを選択するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BSMXACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSSEL[4:0]				
											rw	rw	rw	rw	rw

ビット31 **BSMXACTIVE**：BSMXアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、BSMXが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用されます。BSSEL[4:0] は、BSMXACTIVEに0がセットされている場合のみ更新されます。このBSMXACTIVEフラグは、OLDACTIVE、DFLTACTIVE、およびSCDACTIVEフラグの論理和です。BSSEL[4:0] ビットフィールドを更新するには、これらに0をセットする必要があります。

0：BSMXはアクティブでなく、必要に応じて設定できます。

1：BSMXはアクティブで、保護されたフィールドは設定できません。

ビット30:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **BSSEL[4:0]**：ビットストリームの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、DFLTxおよびSCDx用に処理されるビットストリームを選択するために使用されます。このビットフィールドのサイズは、インスタンス化されるDFLTxの数によって決まります。このビットフィールドでインスタンス化されていない入力が選択された場合、MDFではインデックス番号の大きい有効なストリームbsx_fが選択されます。

00000：bs0_rをDFLTxおよびSCDxに提供

00001：bs0_fをDFLTxおよびSCDxに提供

00010：bs1_rをDFLTxおよびSCDxに提供（インスタンス化されている場合）

00011：bs1_fをDFLTxおよびSCDxに提供（インスタンス化されている場合）

...

11110：bs15_rをDFLTxおよびSCDxに提供（インスタンス化されている場合）

11111：bs15_fをDFLTxおよびSCDxに提供（インスタンス化されている場合）

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.5 MDFデジタル･フィルタ制御レジスタx（MDF_DFLTxCR）

アドレスオフセット：0x088 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFLTACTIVE	DFLTRUN	Res.	Res.	NBDIS[7:0]								Res.	Res.	Res.	SNPSFMT
r	r			rw	rw	rw	rw	rw	rw	rw	rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRGSRC[3:0]				Res.	Res.	Res.	TRGSENS	Res.	ACQMOD[2:0]			Res.	FTH	DMAEN	DFLTEN
rw	rw	rw	rw				rw		rw	rw	rw		rw	rw	w

ビット31 **DFLTACTIVE**：デジタル･フィルタのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、デジタル･フィルタがアクティブかどうか、つまりイベントを実行できるか、待っているかを示します。

0：デジタル･フィルタはアクティブではありません（櫃余に応じて、DFLTENを通して再有効化できます）。

1：デジタル･フィルタはアクティブです。

ビット30 **DFLTRUN**：デジタル･フィルタの動作ステータス･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、デジタル･フィルタが動作中かどうかを示します。

0：デジタル･フィルタは動作中でなく、新しいトリガイベントを受け付けられる状態です。

1：デジタル･フィルタは動作中です。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27:20 **NBDIS[7:0]**：破棄されるサンプルの数

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、DFLTxが再開されるたびに破棄されるサンプルの数を定義するために使用されます。

0：サンプルの破棄なし

1：1つのサンプルが破棄されます

2：2つのサンプルが破棄されます

.....

255：255のサンプルが破棄されます

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **SNPSFMT**：スナップショット･データ･フォーマット

このビットは、ソフトウェアによってセット／クリアされます。これは、スナップショット･モード用のデータ･フォーマットを選択するために使用されます。

0：積分器カウンタ（INT_CNT）はMDF_SNPSxDRに挿入されず、データの分解能は23bitのままです。

1：積分器カウンタ（INT_CNT）はMDF_SNPSxDRの位置 [15:9] に挿入され、データの分解能は16bitのままです。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット15:12 **TRGSRC[3:0]**：デジタル･フィルタ･トリガ信号の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、どの外部信号により対応するフィルタがトリガされるかを選択するために使用されます。

0000：TRGOが選択されています。

0001：OLDxイベントが選択されています。

0010：mdf_trg[0] が選択されています。

...

1111：mdf_trg[13] が選択されています。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **TRGSENS**：デジタル･フィルタ･トリガのエッジ検知方向の選択

このビットは、ソフトウェアによってセット／クリアされます。これは、外部信号のトリガエッジ検知方向を選択するために使用されます。

0：立ち上がりエッジ･イベントにより取得がトリガされます。

1：立ち下がりエッジ･イベントにより取得がトリガされます。

注： トリガ･ソースがTRGOまたはOLDxイベントの場合、TRGSENS値は考慮されません。TRGOを選択するとエッジ検知方向は強制的に立ち下りエッジになり、OLDxイベントを選択するとエッジ検知方向は強制的に立ち上がりエッジになります。
このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **ACQMOD[2:0]**：デジタル･フィルタのトリガ･モード

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、フィルタのトリガ･モードを選択するために使用されます。

000：非同期連続取得モード

001：非同期シングルショット取得モード

010：同期連続取得モード

011：同期シングルショット取得モード

100：ウィンドウ連続取得モード

101：同期スナップショット･モード

その他：000と同じ

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FTH**：RXFIFO閾値の選択

このビットは、ソフトウェアによってセット／クリアされます。これは、RXFIFO閾値を選択するために使用されます。このビットは、インタリーブ転送モードで動作しているRXFIFOでは重要ではありません（詳細はインタリーブ転送モードを参照）。

0：RXFIFOが空でない場合にRXFIFO閾値イベントが生成されます。

1：RXFIFOがハーフフルの場合にRXFIFO閾値イベントが生成されます。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット1 **DMAEN**：DMAリクエストの有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、処理済みのサンプルをメモリに転送するDMAリクエストの生成を制御するために使用されます。

0：対応するデジタル･フィルタのためのDMAインタフェースは無効です。

1：対応するデジタル･フィルタのためのDMAインタフェースは有効です。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット0 **DFLTEN**：デジタル･フィルタの有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、対応するデジタル･フィルタ･パスの取得の開始を制御するために使用されます。このビットの動作は、ACQMOD[2:0] と外部イベントによって決まります。サンプルを提供するシリアル･インタフェースまたはパラレル･インタフェースも有効化する必要があります。

0：取得がすぐに停止します。

1：ACQMOD[2:0] = 00xまたは101の場合は、取得がすぐに開始されます。ACQMOD[2:0] = 01xまたは100の場合は、該当するトリガ･イベントが発生したときに取得が開始されます。

39.8.6 MDFデジタル･フィルタ設定レジスタx（MDF_DFLTxCICR）

アドレスオフセット：0x08C + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、メインCICフィルタを制御するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SCALE[5:0]						Res.	Res.	Res.	MCICD8
						rw	rw	rw	rw	rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCICD[7:0]								Res.	CICMOD[2:0]			Res.	Res.	DATSRC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:20 **SCALE[5:0]**：スケーリング･ファクタの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、CIC出力に適用されるゲインを選択するために使用されます（詳細は表373を参照）。前の値がまだ適用されていないうちにアプリケーションで新しいゲイン値を書き込もうとした場合、この新しいゲイン値は無視されます。このビットフィールドを読み戻すと、アプリケーションに現在のゲイン値が通知されます。

000000：0dB
000001：+ 3.5dB
000010：+ 6dBまたは左に1bitシフト
.....
011000：+ 72dBまたは左に12bitシフト
100000：- 48.2dBまたは右に8bitシフト（デフォルト値）
100001：- 44.6dB
100010：- 42.1dBまたは右に7bitシフト
100011：- 38.6dB
.....
101110：-6dBまたは右に1bitシフト
101111：- 2.5dB
その他：予約済み

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16:8 **MCICD[8:0]**：CICのデシメーション比の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、CICのデシメーション比を選択するために使用されます。2未満のデシメーション比は指定できません。デシメーション比は（CICDEC+1）により算出されます。

0：デシメーション比は2です。

1：デシメーション比は2です。

2：デシメーション比は3です。

3：デシメーション比は4です。

...

511：デシメーション比は512です。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **CICMOD[2:0]**：CICモードの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、MCICの設定と次数を選択するために使用されます。CICMOD[2:0] = 0xxの場合、CICはメインCIC（MCIC）と補助CIC（ACIC、限界外検出回路に使用）の2つのフィルタに分割されます。

000：CICは2つのフィルタに分割され、MCICがFastSincフィルタに設定されます。

001：CICは2つのフィルタに分割され、MCICが$Sinc^1$ フィルタに設定されます。

010：CICは2つのフィルタに分割され、MCICが$Sinc^2$ フィルタに設定されます。

011：CICは3つのフィルタに分割され、MCICが$Sinc^3$ フィルタに設定されます。

100：CICは単一の$Sinc^4$ フィルタに設定されます。

その他：CICは単一の$Sinc^5$ フィルタに設定されます。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **DATSRC[1:0]**：デジタル･フィルタのためのソース･データ

このビットフィールドは、ソフトウェアによってセット／クリアされます。

0x：BSMXからのストリームが選択されます。

10：ADCITF1からのストリームが選択されます。

11：ADCITF2からのストリームが選択されます。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.7 MDF再形成フィルタ設定レジスタx（MDF_DFLTxRSFR）

アドレスオフセット：0x090 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、再形成フィルタとHPFフィルタを制御するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HPFC[1:0]		HPFBYP	Res.	Res.	RSFLTD	Res.	Res.	Res.	RSFLTBYP
						rw	rw	rw			rw				rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **HPFC[1:0]**：ハイパス･フィルタの遮断周波数

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、ハイパス･フィルタの遮断周波数を選択するために使用されます。F_{PCM} はHPE入力でのサンプリング周波数を表します。

00：遮断周波数 = 0.000625 x F_{PCM}

01：遮断周波数 = 0.00125 x F_{PCM}

10：遮断周波数 = 0.00250 x F_{PCM}

11：遮断周波数 = 0.00950 x F_{PCM}

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット7 **HPFBYP**：ハイパス･フィルタのバイパス

このビットは、ソフトウェアによってセット／クリアされます。これは、ハイパス･フィルタをバイパスするために使用されます。

0：HPFはバイパスされません（デフォルト値）。

1：HPFはバイパスされます。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **RSFLTD**：再形成フィルタのデシメーション比

このビットは、ソフトウェアによってセット／クリアされます。これは、再形成フィルタ用のデシメーション比を選択するために使用されます。

0：デシメーション比は4です（デフォルト値）。

1：デシメーション比は1です。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **RSFLTBYP**：再形成フィルタのバイパス

このビットは、ソフトウェアによってセット／クリアされます。これは、再形成フィルタとそのデシメーション･ブロックをバイパスするために使用されます。

0：再形成フィルタはバイパスされません（デフォルト値）。

1：再形成フィルタはバイパスされます。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.8 MDF積分器設定レジスタx（MDF_DFLTxINTR）

アドレスオフセット：0x094 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、積分器（INT）の設定に使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	INTVAL[6:0]							Res.	Res.	INTDIV[1:0]	
					rw	rw	rw	rw	rw	rw	rw			rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10:4 **INTVAL[6:0]**：積算値の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、積算値を選択するために使用されます。

0：積算値は1、つまりバイパス･モードです（リセット後のデフォルト）。

1：積算値は2です。

2：積算値は3です。

...

127：積算値は128です。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **INTDIV[1:0]**：積分器の出力の分周

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、データ幅が24bit未満に維持されるように、積分器の出力での信号をスケール調整するために使用されます。

00：積分器のデータ出力は128で分周されます（デフォルト値）。

01：積分器のデータ出力は32で分周されます。

10：積分器のデータ出力は4で分周されます。

11：積分器のデータ出力は分周されません。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.9 MDF限界外検出回路制御レジスタx（MDF_OLDxCR）

アドレスオフセット：0x098 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、OLDxを設定するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OLDACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACICD[4:0]					Res.
r										rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ACICN[1:0]		Res.	Res.	Res.	Res.	BKOLD[3:0]				Res.	Res.	THINB	OLDEN
		rw	rw					rw	rw	rw	rw			rw	rw

ビット31 **OLDACTIVE**：OLDxのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、OLDxが効果的に有効化されている（アクティブになっている）かどうかを確認するために使用されます。この機能の保護されたフィールドとレジスタを更新できるのは、OLDACTIVEに0がセットされているときのみです（詳細はセクション39.4.15を参照）。

OLDENでの遷移とOLDACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のmdf_proc_ckです。

0：OLDxはアクティブでなく、必要に応じて設定できます。

1：OLDxはアクティブで、保護されたフィールドは設定できません。

ビット30:22 予約済みであり、リセット値に保持する必要があります。

ビット21:17 **ACICD[4:0]**：OLDx CICのデシメーション比の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、ACICのデシメーション比を選択するために使用されます。MDFによって考慮されるのは、CICMOD[2:0] = 0xxの場合のみです。デシメーション比は（ACICD + 1）により算出されます。

0：デシメーション比は1です。

1：デシメーション比は2です。

2：デシメーション比は3です。

3：デシメーション比は4です。

...

31：デシメーション比は32です。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット16:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **ACICN[1:0]**：OLDx CICの次数の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、ACICのタイプと次数を選択するために使用されます。MDFによって考慮されるのは、CICMOD[2:0] = 0xxの場合のみです。

00：FastSincフィルタ･タイプ

01：$Sinc^1$ フィルタ･タイプ

10：$Sinc^2$ フィルタ･タイプ

11：$Sinc^3$ フィルタ･タイプ

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット11:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **BKOLD[3:0]**：限界外検出回路のためのブレーク信号の割当て

このビットフィールドは、ソフトウェアによってセット／クリアされます。

BKOLD[i] = 0：ブレーク信号（mdf_break[i]）は閾値イベントに割り当てられません。

BKOLD[i] = 1：ブレーク信号（mdf_break[i]）は閾値イベントに割り当てられます。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **THINB**：帯域内の閾値

このビットは、ソフトウェアによってセット／クリアされます。

0：OLDxによるイベントの生成は、信号がOLDTHLより小さい、またはOLDTHH（デフォルト値）より大きい場合に行われます。

1：OLDxによるイベントの生成は、信号がOLDTHHより小さく、かつOLDTHLより大きい場合に行われます。

注： このビットは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット0 **OLDEN**：OLDxの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：OLDxは無効です（デフォルト値）。

1：OLDxは有効で、ACICフィルタは連続モードで動作しています。

39.8.10 MDF OLDxの閾値下限レジスタx（MDF_OLDxTHLR）

アドレスオフセット：0x09C + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、限界外閾値の下限を調整するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	OLDTHL[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OLDTHL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **OLDTHL[25:0]**：OLD閾値の下限値

このビットフィールドは、ソフトウェアによってセット／クリアされます。OLDTHLは26bitの符号付きの値です。フィルタによって提供される信号と比較される実際の閾値はOLDTHLです。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.11 MDF OLDxの閾値上限レジスタx（MDF_OLDxTHHR）

アドレスオフセット：0x0A0 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、OLDx閾値の上限を調整するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	OLDTHH[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OLDTHH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **OLDTHH[25:0]**：OLDx閾値の上限値

このビットフィールドは、ソフトウェアによってセット／クリアされます。OLDTHHは26bitの符号付きの値です。フィルタによって提供される信号と比較される実際の閾値はOLDTHHです。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

39.8.12 MDF遅延制御レジスタx（MDF_DLYxCR）

アドレスオフセット：0x0A4 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、ストリームの遅延を調整するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SKPBF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SKPDLY[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31 **SKPBF**：スキップ･ビジー･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、遅延シーケンスが完了しているかどうかを制御するために使用されます。

0：MDFで新しい値をSKPDLY[6:0] に受け入れることが可能です。

1：前回の有効なSKPDLY[6:0] がまだ処理中です。

ビット30:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **SKPDLY[6:0]**：ビットストリームに適用するための遅延

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、スキップする入力サンプルの数を定義します。SKPBF = 0で、対応するDFLTENが1の場合、このビットフィールドに書き込んだ直後にスキップが適用されます。SKPBF = 1の場合、レジスタに書き込まれた値は遅延ステートマシンによって無視されます。

0：入力サンプルはスキップされません。

1：1個の入力サンプルがスキップされます。

.....

127：127個の入力サンプルがスキップされます。

39.8.13 MDF短絡検出回路の制御レジスタx（MDF_SCDxCR）

アドレスオフセット：0x0A8 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、ストリームの遅延を調整するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCDACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCDT[7:4]			
r												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCDT[3:0]				Res.	Res.	Res.	Res.	BKSCD[3:0]				Res.	Res.	Res.	SCDEN
rw	rw	rw	rw					rw	rw	rw	rw				rw

ビット31 **SCDACTIVE**：SCDxのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、SCDxが効果的に有効化されている（アクティブになっている）かどうかを確認するために使用されます。この機能の保護されたフィールドを更新できるのは、SCDACTIVEに0がセットされているときのみです（詳細はセクション39.4.15を参照）。

SCDENでの遷移とSCDACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のmdf_proc_ckです。

0：SCDxはアクティブでなく、必要に応じて設定できます。

1：SCDxはアクティブで、保護されたフィールドは設定できません。

ビット30:20 予約済みであり、リセット値に保持する必要があります。

ビット19:12 **SCDT[7:0]**：SCDxの閾値

このビットフィールドは、ソフトウェアによってセット／クリアされます。これらのビットは、SCDxの閾値カウンタを定義するために、ソフトウェアによって書き込まれます。この値に達すると、特定の入力ストリームで短絡検出回路イベントが発生します。

0：2個の連続する1または0によりイベントが生成されます。

1：2個の連続する1または0によりイベントが生成されます。

2：3個の連続する1または0によりイベントが生成されます。

...

255：256個の連続する1または0によりイベントが生成されます。

注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット11:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **BKSCD[3:0]**：短絡検出回路のためのブレーク信号の割当て
このビットフィールドは、ソフトウェアによってセット／クリアされます。
BKSCD[i] = 0：ブレーク信号（mdf_break[i]）はこのSCDイベントに割り当てられません。
BKSCD[i] = 1：ブレーク信号（mdf_break[i]）はこのSCDイベントに割り当てられます。
注： このビットフィールドは書込み保護できます（詳細はセクション39.4.15を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SCDEN**：SCDxの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SCDxは無効です。
1：SCDxは有効です。

39.8.14 MDFMDF DFLT0割込み有効化レジスタ0 （MDF_DFLT0IER）

アドレスオフセット：0x0AC

リセット値：0x0000 0000

このレジスタは、割込みを生成するイベントを許可または禁止するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RFOVRIE	CKABIE	SATIE	SCDIE	SSOVRIE	Res.	Res.	OLDIE	Res.	SSDRIE	DOVRIE	FTHIE
				rw	rw	rw	rw	rw			rw		rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **RFOVRIE**：再形成フィルタ･オーバーラン割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：再形成フィルタ･オーバーラン割込みは無効です。
1：再形成フィルタ･オーバーラン割込みは有効です。

ビット10 **CKABIE**：クロックなし検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：クロックなし割込みは無効です。
1：クロックなし割込みは有効です。

ビット9 **SATIE**：飽和検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：飽和割込みは無効です。
1：飽和割込みは有効です。

ビット8 **SCDIE**：SCD0割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SCD0割込みは無効です。
1：SCD0割込みは有効です。

ビット7 **SSOVRIE**：スナップショット･オーバーラン割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：スナップショット･オーバーラン割込みは無効です。
1：スナップショット･オーバーラン割込みは有効です。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OLDIE**：OLD0割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：OLD0イベント割込みは無効です。
1：OLD0イベント割込みは有効です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SSDRIE**：スナップショット･データレディ割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：スナップショット･データレディ割込みは無効です。
1：スナップショット･データレディ割込みは有効です。

ビット1 **DOVRIE**：データ･オーバーフロー割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：データ･オーバーフロー割込みは無効です。
1：データ･オーバーフロー割込みは有効です。

ビット0 **FTHIE**：RXFIFO閾値割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：RXFIFO閾値割込みは無効です。
1：RXFIFO閾値割込みは有効です

39.8.15 MDF DFLTx割込み有効化レジスタx（MDF_DFLTxIER）

アドレスオフセット：0x12C + 0x80 * (x-1)(x = 1から5)

リセット値：0x0000 0000

このレジスタは、割込みを生成するイベントを許可または禁止するために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RFOVRIE	CKABIE	SATIE	SCDIE	SSOVRIE	Res.	Res.	OLDIE	Res.	SSDRIE	DOVRIE	FTHIE
				rw	rw	rw	rw	rw			rw		rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **RFOVRIE**：再形成フィルタ･オーバーラン割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：再形成フィルタ･オーバーラン割込みは無効です。
1：再形成フィルタ･オーバーラン割込みは有効です。

ビット10 **CKABIE**：クロックなし検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：クロックなし割込みは無効です。
1：クロックなし割込みは有効です。

ビット9 **SATIE**：飽和検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：飽和割込みは無効です。
1：飽和割込みは有効です。

ビット8 **SCDIE**：SCDx割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：SCDx割込みは無効です。
1：SCDx割込みは有効です。

ビット7 **SSOVRIE**：スナップショット･オーバーラン割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：スナップショット･オーバーラン割込みは無効です。
1：スナップショット･オーバーラン割込みは有効です。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **OLDIE**：OLDx割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：OLDx割込みは無効です。
1：OLDx割込みは有効です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SSDRIE**：スナップショット･データレディ割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：スナップショット･データレディ割込みは無効です。
1：スナップショット･データレディ割込みは有効です。

ビット1 **DOVRIE**：データ･オーバーフロー割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：データ･オーバーフロー割込みは無効です。
1：データ･オーバーフロー割込みは有効です。

ビット0 **FTHIE**：RXFIFO閾値割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：RXFIFO閾値割込みは無効です。
1：RXFIFO閾値割込みは有効です

39.8.16 MDF DFLT0割込みステータス･レジスタ0 （MDF_DFLT0ISR）

アドレスオフセット：0x0B0

リセット値：0x0000 0000

このレジスタには、各デジタル･フィルタ･パス用のステータス･フラグが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RFOVRF	CKABF	SATF	SCDF	SSOVRF	THHF	THLF	OLDF	RXNEF	SSDRF	DOVRF	FTHF
				rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	rc_w1	r	rc_w1	rc_w1	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **RFOVRF**：再形成フィルタ･オーバーラン検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、再形成フィルタ･オーバーランが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、再形成フィルタ･オーバーランが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット10 **CKABF**：クロックなし検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、クロックなしが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、クロックなしが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット9 **SATF**：飽和検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、飽和が検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、飽和が検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット8 **SCDF**：短絡検出回路フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、SCD0イベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、SCD0イベントが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット7 **SSOVRF**：スナップショット･オーバーラン･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、スナップショット･オーバーラン･イベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、スナップショット･オーバーラン･イベントが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット6 **THHF**：閾値上限ステータス･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。これは、前回OLD0イベントが発生したときの閾値上限コンパレータのステータスを示します。このビットでは、前回のOLD0イベントをトリガした条件に関する追加情報が示されます。クリアするには、OLDFフラグに1を書き込みます。

0：前回OLD0イベントが発生したとき、信号はOLDTHH未満でした。

1：前回OLD0イベントが発生したとき、信号はOLDTHHを上回っていました。

ビット5 **THLF**：閾値下限ステータス･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。これは、前回OLD0イベントが発生したときの閾値下限コンパレータのステータスを示します。このビットでは、前回のOLD0イベントをトリガした条件に関する追加情報が示されます。クリアするには、OLDFフラグに1を書き込みます。

0：前回OLD0イベントが発生したとき、信号はOLDTHLを上回っていました。

1：前回OLD0イベントが発生したとき、信号はOLDTHL未満でした。

ビット4 OLDF：OLD0フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、OLD0イベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、OLD0イベントが検出されることを意味します。1を書き込むと、THHF、THLF、およびOLDFがクリアされます。

ビット3 **RXNEF**：RXFIFOノットエンプティ･フラグ

このビットは、RXFIFOレベルに従って、ハードウェアによってセット／クリアされます。

0：RXFIFOエンプティ

1：RXFIFOノットエンプティ

ビット2 **SSDRF**：スナップショット･データレディ･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、データが利用可能でないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、新しいデータが利用可能であることを意味します。1を書き込むと、このフラグがクリアされます。

ビット1 **DOVRF**：データ･オーバーフロー･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、オーバーフローが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、オーバーフローが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット0 **FTHF**：RXFIFO閾値フラグ

このビットはハードウェアによってセットされ、RXFIFOレベルが閾値を下回るとハードウェアによってクリアされます。

0：RXFIFO閾値に達していません。

1：RXFIFO閾値に達しました。

39.8.17 MDF DFLTx割込みステータス･レジスタx（MDF_DFLTxISR）

アドレスオフセット：0x130 + 0x80 * (x-1)(x = 1から5)

リセット値：0x0000 0000

このレジスタには、各デジタル･フィルタ･パス用のステータス･フラグが格納されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RFOVRF	CKABF	SATF	SCDF	SSOVRF	THHF	THLF	OLDF	RXNEF	SSDRF	DOVRF	FTHF
				rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	rc_w1	r	rc_w1	rc_w1	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **RFOVRF**：再形成フィルタ･オーバーラン検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、再形成フィルタ･オーバーランが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、再形成フィルタ･オーバーランが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット10 **CKABF**：クロックなし検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、クロックなしが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、クロックなしが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット9 **SATF**：飽和検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、飽和が検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、飽和が検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット8 **SCDF**：短絡検出回路フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、SCDイベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、SCDイベントが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット7 **SSOVRF**：スナップショット･オーバーラン･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、スナップショット･オーバーラン･イベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、スナップショット･オーバーラン･イベントが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット6 **THHF**：閾値上限ステータス･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。これは、前回OLDxイベントが発生したときの閾値上限コンパレータのステータスを示します。このビットでは、前回のOLDxイベントをトリガした条件に関する追加情報が示されます。クリアするには、OLDFフラグに1を書き込みます。

0：前回OLDxイベントが発生したとき、信号はOLDTHH未満でした。

1：前回OLDxイベントが発生したとき、信号はOLDTHHを上回っていました。

ビット5 **THLF**：閾値下限ステータス･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。これは、前回OLDxイベントが発生したときの閾値下限コンパレータのステータスを示します。このビットでは、前回のOLDxイベントをトリガした条件に関する追加情報が示されます。クリアするには、OLDFフラグに1を書き込みます。

0：前回OLDxイベントが発生したとき、信号はOLDTHLを上回っていました。

1：前回OLDxイベントが発生したとき、信号はOLDTHL未満でした。

ビット4 OLDF OLDxフラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、OLDxイベントが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、OLDxイベントが検出されることを意味します。1を書き込むと、THHF、THLF、およびOLDFがクリアされます。

ビット3 **RXNEF**：RXFIFOノットエンプティ･フラグ

このビットは、RXFIFOレベルに従って、ハードウェアによってセット／クリアされます。

0：RXFIFOエンプティ

1：RXFIFOノットエンプティ

ビット2 **SSDRF**：スナップショット･データレディ･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、データが利用可能でないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、新しいデータが利用可能であることを意味します。1を書き込むと、このフラグがクリアされます。

ビット1 **DOVRF**：データ･オーバーフロー･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、オーバーフローが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、オーバーフローが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット0 **FTHF**：RXFIFO閾値フラグ

このビットはハードウェアによってセットされ、RXFIFOレベルが閾値を下回るとハードウェアによってクリアされます。

0：RXFIFO閾値に達していません。

1：RXFIFO閾値に達しました。

39.8.18 MDFオフセット･エラー補正制御レジスタx（MDF_OECxCR）

アドレスオフセット：0x0B4 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタには、オフセット補正値が格納されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	OFFSET[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **OFFSET[25:0]**：オフセット･エラー補正

このビットフィールドは、ソフトウェアによってセット／クリアされます。前の値がまだ適用されていないうちにアプリケーションで新しいオフセット値を書き込もうとした場合、この新しいオフセット値は無視されます。このビットフィールドを読み戻すと、アプリケーションに現在のオフセット値が通知されます。

このビットフィールドは、SCALEに進む前に信号から減算する値を示します。

39.8.19 MDFスナップショット･データ･レジスタx（MDF_SNPSxDR）

アドレスオフセット：0x0EC + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、スナップショット･モードで各デジタル･フィルタによって処理されたデータを読み出すために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SDR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTSDR[6:0]							MCICDC[8:0]								
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **SDR[15:0]**：デジタル･フィルタによって処理された前回の有効なデータの16MSBが格納されます。

ビット15:9 **EXTSDR[6:0]**：拡張データ･サイズ

SNPSFMT = 0の場合、このビットフィールドにはデジタル･フィルタによって処理された前回の有効なデータのビット7 ~ 1が格納されます。

SNPSFMT = 1の場合、このビットフィールドには、前回トリガ･イベントが発生したときのINTアキュムレータのカウント値（INT_CNT）が格納されます。

ビット8:0 **MCICDC[8:0]**：前回トリガ･イベントが発生したときのMCICデシメーションのカウント値（MCIC_CNT）が格納されます。

39.8.20 MDFデジタル･フィルタのデータ･レジスタx（MDF_DFLTxDR）

アドレスオフセット：0x0F0 + 0x80 * x（x = 0から5）

リセット値：0x0000 0000

このレジスタは、各デジタル･フィルタによって処理されたデータを読み出すために使用されます。レジスタの数はフィルタの量と等しくなります。詳細については、セクション39.3を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r								

ビット31:8 **DR[23:0]**：デジタル･フィルタによって処理されたデータ

ビット7:0 予約済みであり、リセット値に保持する必要があります。

39.8.21 MDFレジスタ･マップ

表383.MDFレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	MDF_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ILVNB [3:0]				Res.	Res.	Res.	TRGO
	リセット値																									0	0	0	0				0
0x004	MDF_CKGCR	CKGACTIVE	PROCDIV[6:0]							Res.	Res.	Res.	Res.	CCKDIV [3:0]			TRGSRC [3:0]					Res.	Res.	Res.	TRGSENS	Res.	CCK1DIR	CCK0DIR	CKGMOD	Res.	CCK1EN	CCK0EN	CKGDEN
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0				0		0	0	0		0	0	0
0x008 - 0x07C	予約済み	予約済み																															
0x080 + 0x80 * x (x = 0 ~ 5)	MDF_SITFxCR	SITFACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STH[4:0]					Res.	Res.	SITFMOD [1:0]		Res.	SCKSRC [1:0]		SITFEN
	リセット値	0																			1	1	1	1	1			0	0		0	0	0
0x084 + 0x80 * x (x = 0 ~ 5)	MDF_BSMXxCR	BSMXACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSSEL[4:0]				
	リセット値	0																											0	0	0	0	0
0x088 + 0x80 * x (x = 0 ~ 5)	MDF_DFLTxCR	DFLTACTIVE	DFLTRUN	Res.	Res.	NBDIS[7:0]								Res.	Res.	Res.	SNPSFMT	TRGSRC [3:0]				Res.	Res.	Res.	TRGSENS	Res.	ACQMOD [2:0]			Res.	FTH	DMAEN	DFLTEN
	リセット値	0	0			0	0	0	0	0	0	0	0				0	0	0	0	0				0		0	0	0		0	0	0
0x08C + 0x80 * x (x = 0 ~ 5)	MDF_DFLTxCICR	Res.	Res.	Res.	Res.	Res.	Res.	SCALE[5:0]						Res.	Res.	Res.	MCICD[8:0]									Res.	CICMOD [2:0]			Res.	Res.	DATSRC [1:0]	
	リセット値							0	0	0	0	0	0				0	0	0	0	0	0	0	0	0		0	0	0			0	0
0x090 + 0x80 * x (x = 0 ~ 5)	MDF_ DFLTxRSFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HPFC[1:0]		HPFBYP	Res.	Res.	RSFLTD	Res.	Res.	Res.	RSFLTBYP
	リセット値																							0	0	0			0				0
0x094 + 0x80 * x (x = 0 ~ 5)	MDF_DFLTxINTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTVAL[6:0]							Res.	Res.	INTDIV [1:0]	
	リセット値																						0	0	0	0	0	0	0			0	0
0x098 + 0x80 * x (x = 0 ~ 5)	MDF_OLDxCR	OLDACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACICD[4:0]					Res.	Res.	Res.	ACICN [1:0]		Res.	Res.	Res.	Res.	BKOLD [3:0]				Res.	Res.	THINB	OLDEN
	リセット値	0										0	0	0	0	0				0	0					0	0	0	0			0	0

表383.MDFレジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x09C + 0x80 * x (x = 0 ~ 5)	MDF_OLDxTHLR	Res.	Res.	Res.	Res.	Res.	Res.	OLDTHL[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0 + 0x80 * x (x = 0 ~ 5)	MDF_OLDxTHHR	Res.	Res.	Res.	Res.	Res.	Res.	OLDTHH[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A4 + 0x80 * x (x = 0 ~ 5)	MDF_DLYxCR	SKPBF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SKPDLY[6:0]						
	リセット値	0																									0	0	0	0	0	0	0
0x0A8 + 0x80 * x (x = 0 ~ 5)	MDF_SCDxCR	SCDACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCDT[7:0]								Res.	Res.	Res.	Res.	BKSCD [3:0]				Res.	Res.	Res.	SCDEN
	リセット値	0												0	0	0	0	0	0	0	0					0	0	0	0				0
0x0AC	MDF_DFLT0IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOVRIE	CKABIE	SATIE	SCDIE	SSOVRIE	Res.	Res.	OLDIE	Res.	SSDRIE	DOVRIE	FTHIE
	リセット値																					0	0	0	0	0			0		0	0	0
0x0B0	MDF_DFLT0ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOVRF	CKABF	SATF	SCDF	SSOVRF	THHF	THLF	OLDF	RXNEF	SSDRF	DOVRF	FTHF
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x0B4 + 0x80 * x (x = 0 ~ 5)	MDF_OECxCR	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B8 ~0x0E8	予約済み	予約済み																															
0x0EC + 0x80 * x (x = 0 ~ 5)	MDF_SNPSxDR	SDR[15:0]																EXTSDR[6:0]							MCICDC[8:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0F0 + 0x80 * x (x = 0 ~ 5)	MDF_DFLTxDR	DR[23:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x12C + 0x80 * (x-1) (x = 1 ~ 5)	MDF_DFLTxIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOVRIE	CKABIE	SATIE	SCDIE	SSOVRIE	Res.	Res.	OLDIE	Res.	SSDRIE	DOVRIE	FTHIE
	リセット値																					0	0	0	0	0			0		0	0	0
0x130 + 0x80 * (x-1) (x = 1 ~ 5)	MDF_DFLTxISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOVRF	CKABF	SATF	SCDF	SSOVRF	THHF	THLF	OLDF	RXNEF	SSDRF	DOVRF	FTHF
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

40 オーディオ･デジタル･フィルタ（ADF）

40.1 概要

オーディオ･デジタル･フィルタ（ADF）は、外部シグマデルタ（ΣΔ）モジュレータの接続専用の高性能モジュールです。主に、次の用途が対象となります。

- オーディオ･キャプチャ信号
- 計測

ADFは、1のデジタル･シリアル･インタフェース （SITF0）と1つのデジタル･フィルタ（DFLT0）を備えていて、最大24bitの最終分解能に対応する柔軟なデジタル処理オプションが利用できます。

ADFシリアル･インタフェースはいくつかの規格に対応していて、各種 ΣΔ モジュレータ･センサの接続が可能です。

- SPIインタフェース
- マンチェスタ符号化1線インタフェース
- PDMインタフェース

ADFは入力データ･ストリームをデシメートされたクリーンなデジタル･データ･ワードに変換します。この変換は、ローパス･デジタル･フィルタとデシメーション･ブロックにより行われます。また、ハイパス･フィルタを挿入することもできます。

変換速度および分解能は、フィルタのバイパス、フィルタ段数、デシメーション比といったデジタル処理に関する設定可能なパラメータに合わせて調整できます。最大出力データの分解能は24bitです。変換モードには、シングル変換モードと連続モードの2種類があります。データは、DMAを介してシステムRAMバッファに自動的に保存され、ソフトウェアの負荷を軽減できます。

サウンドまたは音声信号の検出のために、サウンド･アクティビティ検出回路（SAD）が利用できます。SADはDFLT0の出力に接続されます。サウンド環境に合わせてSADを適切に調整するため、いくつかのパラメータをプログラムすることができます。SADでは、処理される信号がプログラムされた基準に一致しないかぎり、システムメモリにサンプルを保存しないようにすることで、電力消費を大幅に抑えています。

デジタル処理は、カーネル･クロックのみを使用して行われます。データン転送が必要な場合、または特定のイベントによりシステム･プロセッサの応対処置がリクエストされた場合、ADFではバス･インタフェース･クロック（AHBクロック）のみがリクエストされます。

40.2 ADFの主な機能

- AHBインタフェース
- 1 シリアル･デジタル入力：
  - さまざまなデジタル･センサを接続するために設定可能なSPIインタフェース
  - 設定可能なマンチェスタ符号化インタフェースのサポート
  - PDMインタフェースとの互換性によりデジタル･マイクロフォンに対応
- ΣΔ モジュレータ用の2つの共通クロック入力／出力
- 次のような1柔軟なデジタル･フィルタ･パス
  - 調整可能なデシメーション比を備えた、$Sinc^4$ または$Sinc^5$ フィルタで設定可能なMCICフィルタ
  - 帯域外除去とインバンド･リップルを向上させるための再形成フィルタ
  - ADCオフセットを解除するためのハイパス･フィルタ

  - 制御の獲得
  - 飽和ブロック
- クロックなし検出回路
- サウンド･アクティビティ検出回路
- 24bitの符号付き出力データ分解能
- 連続またはシングル変換
- 選択したビットストリームを遅延させることが可能
- 1つのトリガ入力
- STOPモードでの自律型機能
- DMAを使用した変換データの読出しが可能
- 割込みサービス

40.3 ADFの実装

デバイスには1つのMDFインスタンスと1つのADFインスタンスが内蔵されています。これらは共通の機能を備えたデジタル･フィルタです。

表384.ADF/MDFの機能(1)

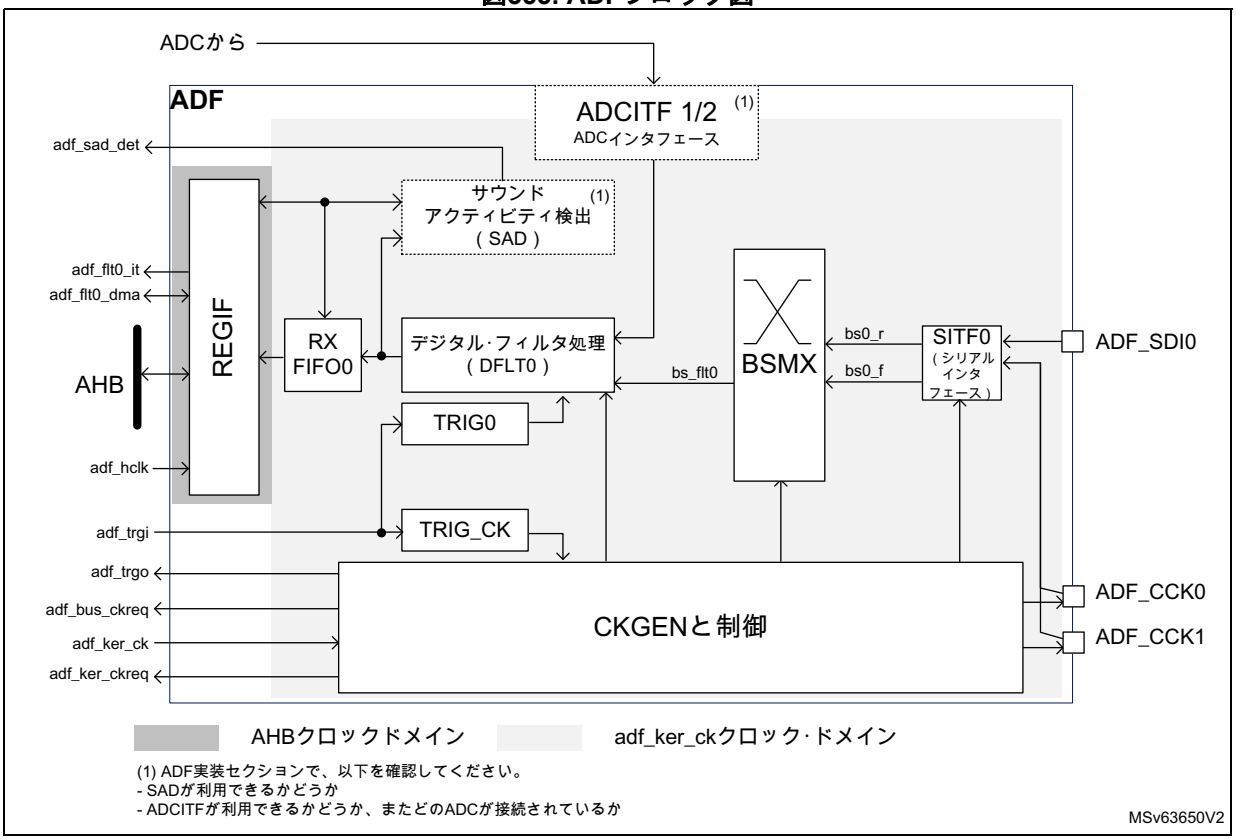
モードまたは機能	ADF1すべてのデバイス	MDF1 STM32U535/545	MDF1 STM32U575/585	MDF1 STM32U59x/5Ax 5Fx/5Gx
フィルタの数（DFLTx）と シリアル･インタフェース（SITFx）	1	2	6	
ピンに接続されたMDF_CKIy/ADF_CKI0	-	-	X	
サウンド･アクティビティ検出（SAD）	X	-		
RXFIFOの深さ（24bit･ワード数）	4	4		
ADCITF1に接続されたADC	-	ADC1		
ADCITF2に接続されたADC	-	-		ADC2
モータ専用機能（SCD、OLD、OEC、INT、スナップショット、ブレーク）	-	X		
CIC4、CIC5を備えたメイン･パス	X			
CIC1、2、3またはFastSincを備えたメイン･パス	-			
RSFLT、HPF、SAT、SCALE、DLY、破棄機能	X			
STOPモードでの自律型	X(2)	X(3)		

1. “X” はサポートされています。“-” はサポートされていません。
2. STOP 0、STOP 1およびSTOP 2モードのみ。
3. STOP 0およびSTOP 1モードのみ。

40.4 ADFの機能説明

40.4.1 ADFブロック図

図355. ADFブロック図



40.4.2 ADFピンおよび内部信号

表385.ADF外部ピン

名前	信号タイプ	説明
ADF_SDI0	入力	外部センサからのデータ信号。
ADF_CCKy （y = 0、1）	入力／出力	外部センサ用のクロック出力、または外部センサからの共通クロック入力

表386.ADF内部信号

名前	信号タイプ	説明
adf_trgi	入力	取得を制御するトリガ入力（詳細については次の表を参照）
adf_trgo	出力	他の MDFインスタンスとの同期化のためのトリガ出力
adf_flt0_dma	入力／出力	ADF処理チェーンのためのDMAリクエスト／確認応答信号。
adf_flt0_it	出力	グローバル割込み信号
adf_bus_ckreq	出力	バス･インタフェース･クロック･リクエスト出力

表386.ADF内部信号 （続き）

名前	信号タイプ	説明
adf_ker_ckreq	出力	カーネル･クロック･リクエスト出力
adf_ker_ck	入力	カーネル･クロック入力
adf_hclk	入力	AHBバス･インタフェース･クロック入力
adf_sad_det	出力	SADサウンド検出：1はサウンドの検出を意味します
adf_adcitf1_dat [15:0]	入力	ADCITF1データ入力
adf_adcitf2_dat [15:0]	入力	ADCITF2データ入力

表387.ADFトリガ接続

トリガ名	方向	トリガ転送元／転送先
adf_trgi	入力	exti15から
adf_trgo	出力	mdf_trgi13へ

40.4.3 シリアル入力インタフェース（SITF）

SITF0 入力インタフェースを使用すると、ビットストリーム･マトリックス（BSMX）を通して、外部センサをデジタル･フィルタに接続することができます。SITF0 は次のモードで設定できます。

- LF_MASTER SPIモード（低周波数）
- 通常SPIモード
- マンチェスタ･モード

シリアルインタフェースからのデータは、PDMからPCMへの変換とサウンド･アクティビティ検出を実行するために、フィルタにルーティングされます。

シリアル･インタフェースを有効化するには、対応するSITFENビットに1をセットします。インタフェースが有効化されると、外部 ΣΔ モジュレータからシリアル･データを受信します。

注： **シリアル･インタフェースを有効化する前に、adf_proc_ck が有効化されていることを確認する必要があります（詳細は、セクション 40.4.5：クロック･ジェネレータ（CKGEN）を参照）。**

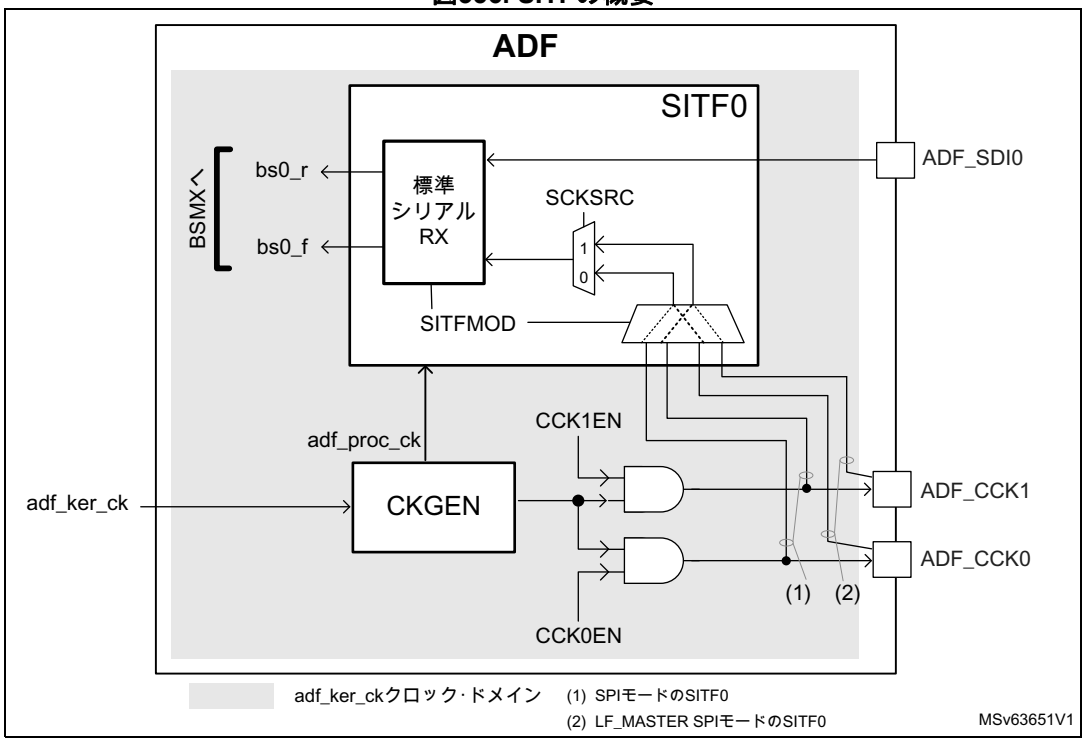
SITF0 は、ADFシリアル･インタフェース制御レジスタ0（ADF_SITF0CR）を通して制御されます。

図 355 に示すように、受信ビットストリームをサンプリングするために、ADF_CCK0 または ADF_CCK1をクロック･ソースとして選択できます。

- シリアル･インタフェースがSPIモードでプログラムされている場合、選択されるクロック･ソースはADF_CCK0またはADF_CCK1ピンにあるクロックのコピーです。
- シリアル･インタフェースがLF_MASTER SPIモードでプログラムされている場合、選択されるクロック･ソースはCCKDIVからADF_CCK0またはADF_CCK1ピンに直接供給されるクロックです。

詳細については、表386を参照してください。

図356. SITFの概要



LF_MASTERおよび通常SPIモード

LF_MASTER SPIモードは、周波数がセンサ･クロックの2倍の高さであるadf_proc_ckクロック周波数を使用できる特別なモードです。このモードは低消費電力ユースケース専用で、ロースピード･センサを使用します。

LF_MASTER SPIモードでは、ADFがADF_CCK0およびADF_CCK1ピンを通して外部センサにビットストリーム･クロックを供給する必要があります。ADFは、シリアル･データ入力ADF_SDI0 を通してビットストリーム･データを受信します。

SITF0 について、最適なタイミングが得られるように、アプリケーションでは外部センサに供給されるのと同じクロック（ADF_CCK0 または ADF_CCK1）を選択する必要があります。選択はSCKSRC[1:0] を通して行います。

通常SPIインタフェースはLF_MASTER SPIよりも柔軟性の高いインタフェースですが、adf_proc_ck周波数を少なくともセンサ･クロックの4倍にする必要があります。

アプリケーションでは、ADF_SDI0 ピンを通して受信したデータをキャプチャするために、ADF_CCK0またはADF_CCK1 クロックを選択することができます。

ADFでは、必要に応じて、ADF_CCK0またはADF_CCK1を通してセンサへのクロックを生成できます。

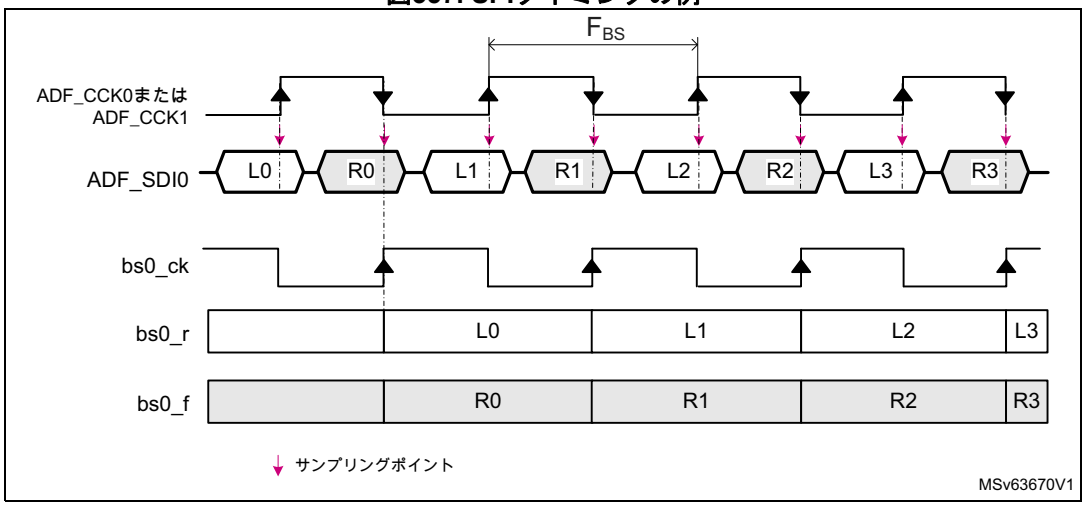
すべてのSPIモードにおいて、選択したクロックの立ち上がりエッジと立ち下がりエッジを使用して、シリアル･データがキャプチャされます。SITF0 では常に次のビットストリームが提供されます。

- ビットストリーム･クロックの立ち下がりエッジを使用して受信したビットストリーム（bs0_f）
- ビットストリーム･クロックの立ち上がりエッジを使用して受信したビットストリーム（bs0_r）

接続されているセンサによっては、2つのビットストリームのいずれかが利用できないことがあります。

アプリケーションから、BSMXマトリックスを通して望ましいストリームを選択することができます。

図357. SPIタイミングの例



データ･ストリームを適切に同期／受信するには、表387に示す制約に従って、adf_proc_ck周波数を調整する必要があります。

クロックなし検出

シリアル･インタフェースが通常SPIモードで動作している場合、クロック遷移なし期間が検出されることがあります。この機能は、SPIリンクでクロック障害を検出するために使用できます。

アプリケーションから、SITF0 のSTH[4:0] ビットフィールドを通してタイムアウト値をプログラムすることができます。STH[4:0] x $T_{adf_proc_ck}$ の間、ADFによってクロック遷移が検出されない場合、CKABFフラグがセットされます。

CKABIEに1がセットされている場合は、割込みを生成することができます。STH[4:0bitフィールドは、ADFシリアル･インタフェース制御レジスタ0（ADF_SITF0CR）にあります。

シリアル･インタフェースが有効化されている場合、最初のクロック遷移が検出されるまで、CKABFフラグは1のままです。

疑似クロックなし検出を避けるため、次のシーケンスに従う必要があります。

1. シリアル･インタフェースを通常SPIモードで設定し、有効化します。
2. CKABFビットに1を書き込んでCKABFフラグをクリアします。

   シリアル･インタフェースでクロック遷移なしが検出された場合、ハードウェアによりただちにCKABFフラグに1をセットします。
3. CKABFフラグを読み出します。
   - CKABF = 1の場合、手順2に戻ります。
   - CKABF = 0の場合、クロックが検出されています。クロックなし検出時に割込みを発生させるには、CKABIEビットに1をセットします。

注： **クロックなし検出機能は、LF_MASTER SPIモードでは使用できません。**

マンチェスタ･モード

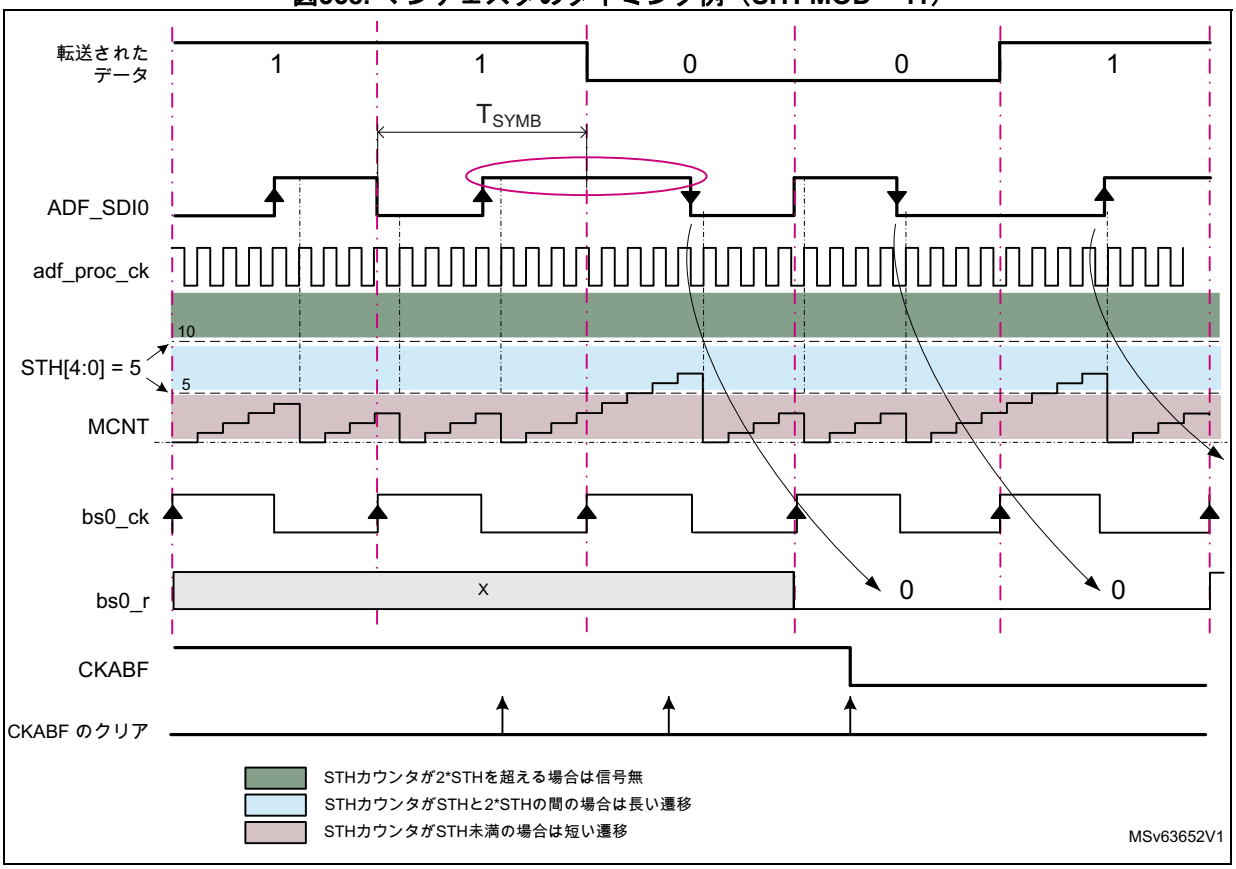
マンチェスタ符号化フォーマットでは、ADFはADF_SDI0 ピンを通してのみ、外部センサからデータ･ストリームを受信します。

このモードでは、ADF_CCK0ピンとADF_CCK1ピンは不要です。

デコードされたデータおよびクロック信号は、マンチェスタ･デコードの後、シリアル･ストリームから復元されます。これらは、bs0_r上で利用できます。マンチェスタ符号化では、次の2通りの設定ができます。

- 信号立ち上がりエッジを0として符号化し、信号立ち下がりエッジを1として符号化
- 信号立ち上がりエッジを1として符号化し、信号立ち下がりエッジを0として符号化

図358. マンチェスタのタイミング例（SITFMOD = 11）



受信マンチェスタ･ストリームをデコードするには、ADFシリアル･インタフェース制御レジスタ0（ADF_SITF0CR）でSTH[4:0] をプログラムする必要があります。マンチェスタ･シンボル長を推定し、クロックなし検出を行うため、STH[4:0] ビットフィールドがSITF0 によって使用されます。内部カウンタ（MCNT）は、ADF_SDI0 入力で遷移が検出されるたびに再開されます。これは、短い遷移、長い遷移、クロックなしの検出に使用されます。長い遷移は、データ値が変化したことを示します。図357には、OVRが高さの周辺でSTH[4:0] = 5であるケースを示します。

マンチェスタの推定シンボルレート（T_{SYMB}）は、次の式に従う必要があります。

$$(STH + 1)\times T_{adf_proc_ck} < T_{SYMB} < (2 \times STH \times T_{adf_proc_ck})$$

STHの計算は次のように行うことを推奨します。

$$STH[4:0] = round\left(\frac{(2 \times OVR) - 1}{3}\right)$$

ここで、OVRはadf_proc_ck周波数と予測されるマンチェスタ･シンボル周波数との比率を示します。OVRは5より大きくする必要があり、adf_proc_ckクロックは表387に示す制約に従って調整する必要があります。

2 x STH[4:0] x $T_{adf_proc_ck}$ より長い間、遷移が検出されない場合、またはSITF0 がまだ受信マンチェスタ･ストリームに同期されていない場合、クロックなしフラグCKABFに1がセットされます。また、ビットCKABIEに1がセットされている場合、割込みを生成できます。

シリアル･インタフェースが有効化されている場合、まずADFを受信マンチェスタス･トリームに同期する必要があります。0から1、または1から0へのデータの遷移（図357のピンクの楕円）が検出されると、同期は終了します。

同期フェーズの終了は、次のソフトウェア･シーケンスに従って確認できます。

1. CKABFビットに1を書き込んで、ADF DFLT0割込みステータス･レジスタ0 （ADF_DFLT0ISR）でCKABFフラグをクリアします。シリアル･インタフェースがまだ同期されていない場合、ハードウェアによりすぐにCKABFフラグに1がセットされます。
2. CKABFフラグを読み出します。
   - CKABF = 1の場合、手順1に戻ります。
   - CKABF = 0の場合、マンチェスタ･インタフェースが同期され、有効なデータが提供されます。

プログラミングの例

次の例では、ADFカーネル･クロック周波数（$F_{adf_ker_ck}$）は100MHzで、受信マンチェスタ･ストリームは約6MHz（F_{SYMB}）となります。

1. 有効なadf_proc_ckをSITF0 に供給します。

   adf_proc_ck周波数はマンチェスタ･シンボル周波数の少なくとも6倍（つまり、少なくとも36MHz）にする必要があります。

   カーネル･クロックを2分周するには、PROCDIVを1にプログラムします。その場合、$F_{adf_proc_ck}$ = 50MHzとなります（マンチェスタ･シンボル周波数の8.33倍）。
2. STHを計算します。

   OVRは次の式で算出できます。OVR = $F_{adf_proc_ck}$ / F_{SYMB} = 50MHz / 6MHz = 8.33。

   そして、次の計算をします。STH[4:0] = $\text{round}\left(\frac{(2 \times 8.33) - 1}{3}\right) = 5$

   マンチェスタ･ストリームの最小許容周波数は、次のようになります。

   1 / (2 × STH × $T_{adf_proc_ck}$) = 1 / (10 x 20ns) = 5MHz

   マンチェスタ･ストリームの最大許容周波数は、次のようになります。

   1 / (STH+1 × $T_{adf_proc_ck}$ = 1 / (6 x 20ns) = 8.33MHz

40.4.4 ADCスレーブ･インタフェース（ADCITF）

ADCは必ずしもADFに接続されているわけではありません。この製品の状況を確認するには、セクション40.3を参照してください。

ADFではフィルタ･パスに最大2つのADCを接続できます。この フィルタの場合、ADFデジタル･フィルタ設定レジスタ0（ADF_DFLT0CICR）のDATSRC[1:0] ビットフィールドを使用して、アプリケーションによりADCからデータを選択することができます。

警告： ADFは、いずれかのADCITF入力からの受信インタリーブ･データには対応していません。

40.4.5 クロック･ジェネレータ（CKGEN）

RCC（リセットおよびクロック･コントローラ）はADFに次のクロックを供給します。

- レジスタ･インタフェース用のAHBクロック（adf_hclk）
- 主にCKGEN経由で回路のその他のすべての部分で使用されるカーネル･クロック（adf_ker_ck）

これらのクロックは非同期クロックになりますので、両方のクロック･ドメインを跨ぐすべての信号は再同期されます。

クロック･ジェネレータ（CKGEN）では処理クロックの生成が行われ、クロックがADF_CCK0およびADF_CCK1ピンに供給されます。これらのすべてのクロックはadf_ker_ckから生成されます。

処理クロック（adf_proc_ck）は、すべての信号処理の実行と、受信シリアルまたはパラレル･ストリームの再サンプリングのために使用されます。

RCCによって供給されるカーネル･クロック周波数を適合させるために、次の分周回路を使用できます。

- PROCDIV[6:0]：カーネル･クロック周波数をパラレルおよびシリアル･インタフェースの制約、および処理ブロックに適合させるために使用します。
- CCKDIV[3:0]：ADF_CCK0およびADF_CCK1クロックの周波数を適合させるために使用します。

PROCDIV[6:0] およびCCKDIV[3:0]：分周回路にクロックが供給されていない場合にプログラムする必要があります（CKGDEN = 0）。

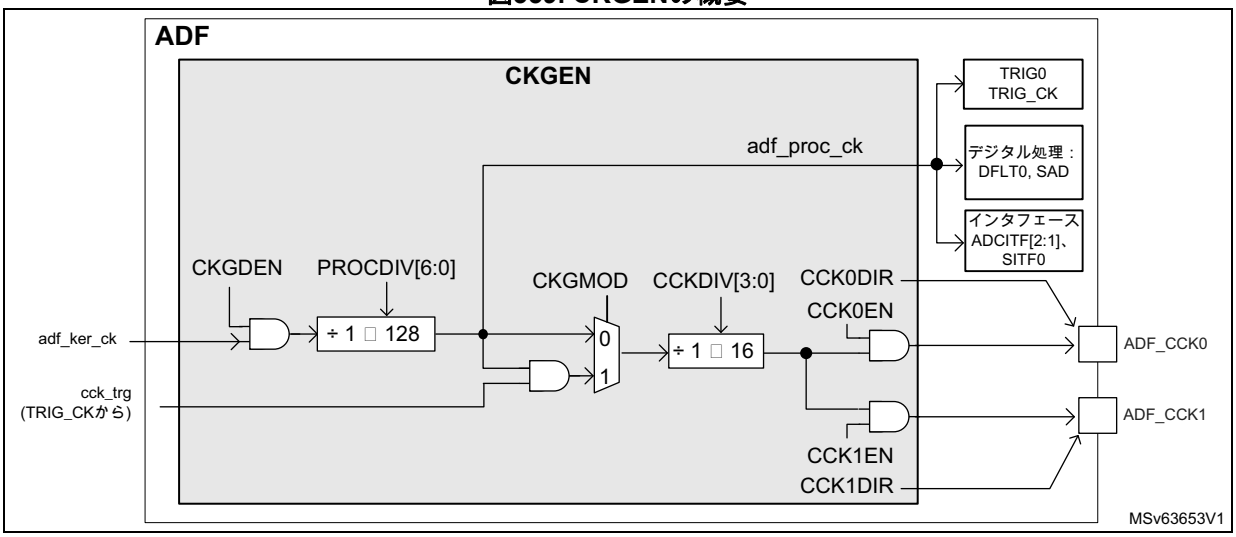
adf_proc_ckの生成はCKGDENによって制御されます。

また、CKGMODビットにより、CCKDIV分周回路のトリガ方法を定義できます。

- CKGMOD = 0の場合、CCKDIV分周回路はCKGDENに1がセットされるとすぐに開始されます。
- CKGMOD = 1の場合、CCKDIV分周回路はCKGDENに1がセットされ、プログラムされたトリガ条件が発生すると開始されます。

CKGEN を制御するビットとフィールドはすべて、ADF クロック･ジェネレータ制御レジスタ（ADF_CKGCR）にあります。

図359. CKGENの概要



CKGENのトリガ･ロジックは、TRG_CKブロックにより処理されます。図364に示すように、CCKDIV分周回路が外部トリガ･ソースの立ち上がりエッジまたは立ち下がりエッジでトリガできます。適切なトリガ条件が発生すると、cck_trg信号が高くなり、CCKDIV分周回路が始動します。CKGDENに0がセットされると、TRG_CKロジックはリセットされます。

この機能は、いくつかのADFインスタンスのADF_CCKy（y = 0、1）クロックを同期したり、クロック生成をタイマ･イベントと同期したりする上で有益です。

アプリケーションにより、CCK0EN/CCK1EN および CCK0DIR/CCK1DIR ビットを使用して、ADF_CCK0またはADF_CCK1ピンの有効化を制御できます。

- CCKyENは、CCKDIVを有効化し、外部センサ用のクロックを生成するために使用されます。
- CCKyDIRは、ADF_CCKyピンの方向（入力または出力）を制御するために使用されます。

表388.共通クロック生成の制御(1)

CCKyEN	CCKyDIR	説明
0	0	ADF_CCKyピンは入力にあります。シリアル･ストリームをデコードするために、外部クロックをADF_CCKyピンに接続し、SITF0 で使用することができます。
0	1	ADF_CCKyピンは出力にあります。クロックは生成されません。ADF_CCKyローに駆動されます。
1	1	ADF_CCKyピンは出力にあります。クロックはADF_CCKyピンで生成されます。シリアル･ストリームをデコードするために、SITF0 ではこのピンをクロック･ソースとして使用することができます。

1. CCKyEN = 1かつCCKyDIR = 0という設定は避ける必要があり、示していません（無意味です）。

注： 他のブロック（SITF0、DFLT0 など）を有効化する前に、adf_proc_ckを（CKGDEN = 1により）有効化する必要があります。

CKGENの有効化シーケンスの例

- CKGDENを0にセットします。
- CKGACTIVE = 0を待ちます。CKGDENが前に有効化されている場合、このフェーズは2周期のadf_hclkと2周期のadf_proc_ckとすることができます。
- PROCDIV[6:0]、CKGMOD、CCKDIV[3:0]、TRGSRC[3:0]、TRGSENS、CCK1EN、およびCCK0ENをプログラムします。
- CKGDENを1にセットします。

必要に応じて、クロック･ジェネレータを無効にすることなく、いつでもCCK[1:0]ENビットフィールド値を変更することができます。

クロック周波数の制約

次の表に、サンプルを適切に受信し、処理するための周波数の制約を示します。

注： **1つのサンプルを処理するには、再形成フィルタ（RSFLT）で最大24サイクルのadf_proc_ckクロックが必要です。**

表389.受信ストリームに関するクロックの制約(1)

SITF0 モード	ADFクロックの制約	
	RSFLT無効時	RSFLT有効時
LF_MASTER SPI	F_{ADF_CCKy} 最大周波数は5Mhzに制限	
	$F_{adf_proc_ck} > 2 \times F_{ADF_CCKy}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$	$F_{adf_proc_ck} > 24 \times F_{ADF_CCKy} / (MCICD+1)$ および $F_{adf_proc_ck} > 2 \times F_{ADF_CCKy}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$
マスタSPI スレーブSPI	F_{ADF_CKx} 最大周波数は25Mhzに制限	
	$F_{adf_proc_ck} > 4 \times F_{ADF_CCKy}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$	$F_{adf_proc_ck} > 24 \times F_{ADF_CCKy} / (MCICD+1)$ および $F_{adf_proc_ck} > 4 \times F_{ADF_CCKy}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$
マンチェスタ	F_{SYMB} 最大周波数は20Mhzに制限	
	$F_{adf_proc_ck} > 6 \times F_{SYMB}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$	$F_{adf_proc_ck} > 24 \times F_{ADF_CCKy} / (MCICD+1)$ および $F_{adf_proc_ck} > 6 \times F_{SYMB}$ および $F_{adf_hclk} \geq F_{adf_proc_ck}$

1. F_{ADF_CCKy} は、ADF_CCKyを通して受信したクロックの周波数、またはADF_CCKyを通して生成されたクロックの周波数を表します。F_{SYMB} はマンチェスタ･モード用の受信シンボル･レートの周波数を表します。

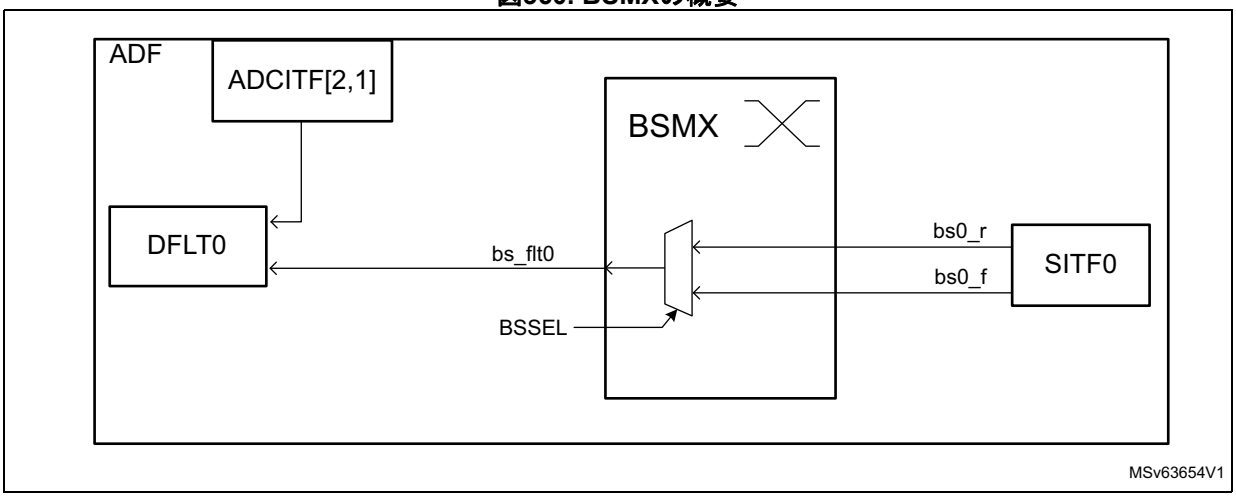
40.4.6 ビットストリーム･マトリックス（BSMX）

BSMXはリアル･インタフェースSITF0 からビットストリームを受信し、選択されたストリームをデジタル･フィルタDFLT0 に送信します。

図355に示すように、SITF0 は2つのビットストリーム（bs0_rおよびbs0_f）をBSMXに送信します。

アプリケーションにより、ADFビットストリーム･マトリックス制御レジスタ0（ADF_BSMX0CR）を通して該当するストリームが選択されます。この選択は静的なものです。

図360. BSMXの概要



BSMXのプログラミングのシーケンス例

DFLT0 が有効化されている場合、BSSEL[4:0] ビットフィールドは変更できません。BSMXの値を変更するには、次の手順を実行する必要があります。

- DFLT0 のDFLTENに0をセットします。
- BSMXACTIVE = 0を待ちます。
- BSSEL[4:0] をプログラムします。
- DFLT0 のDFLTENに1をセットします。

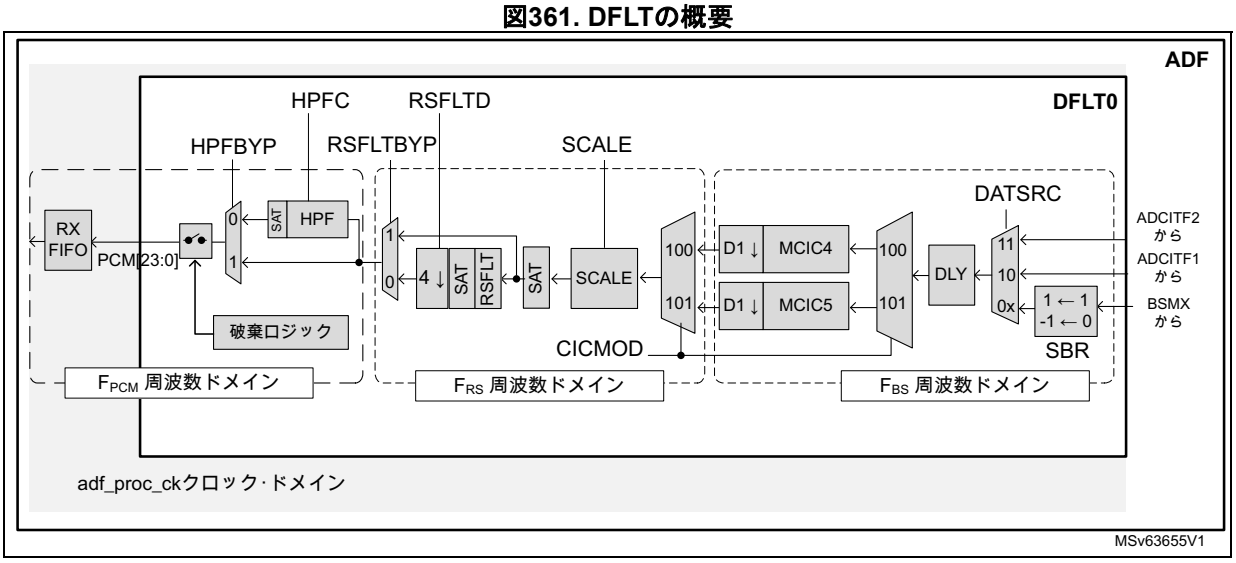
40.4.7 デジタル･フィルタ処理（DFLT）

デジタル･フィルタ処理には、次のサブブロックが含まれます。

- シンボル･リマップ（SBR）
- クロック･スキッパ遅延（DLY）
- $Sinc^4$ または$Sinc^5$ で設定可能なMCICデシメーション･フィルタ
- ゲイン･コントロール（SCALE）
- 信号飽和（SAT）
- 再形成フィルタ（RSFLT）
- ハイパス･フィルタ（HPF）
- 受信RXFIFO

次の図に、CICMOD[2:0] に従ったフィルタ･パス設定を示します。アプリケーションのニーズに応じて、いくつかの設定ビットを利用してデジタル･フィルタを設定することができます。

図361. DFLTの概要



シンボル･リマップとソース選択

シンボル･リマップ（SBR）では、BSMXによって選択されたビットストリームをフィルタ･パスで使用可能なデータに変換します。具体的に言うと、次のようになります。

- 上位レベルは16bitの符号付き数字 + 1に変換されます。
- 下位レベルは16bitの符号付き数字 - 1に変換されます。

デジタル･フィルタの信号ソースは、以下に従って、2つの間でDATSRC[1:0] を通して選択できます。

- BSMXからのデータ
- ADCインタフェースのどちらか（ADCITF2または1）からのデータ

プログラム可能なマイクロ遅延制御（DLY）

デジタル･フィルタは、ビットストリーム･クロックの分解能により各ストリームのタイミング調整を行える遅延ラインを備えています。

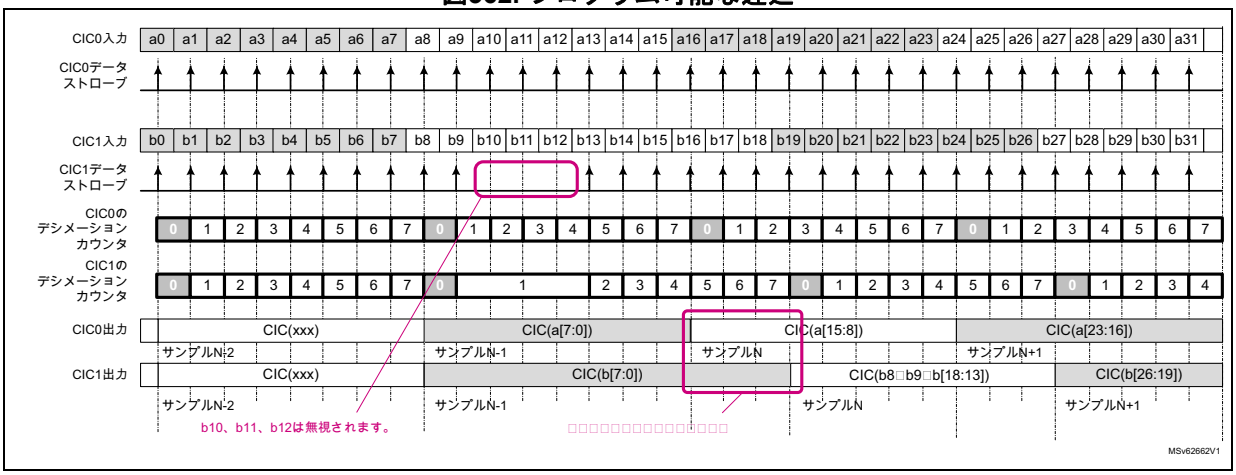
この機能は、マイクロフォンのビーム形成アプリケーションで、受信ストリームに最終的なサンプリング･レートよりも小さな遅延を適用する必要がある場合に特に有益です。この機能は、たとえば、ADFがビーム形成アプリケーション用の他の MDFインスタンス（製品内にある場合）と同期される場合に利用できます。

遅延は、サンプルがCICフィルタに入る前に、選択した入力ストリームから所定の数のサンプルを破棄することによって実行されます。このデータ破棄は、所定の数のデータ･ストローブをスキップし、CICフィルタでこれらのデータが処理されないようにすることで実行されます。

指定した量のデータ･ストローブがスキップされると、次の受信サンプルは正常にストローブ処理されます。

次の図に、受信ストリームに動的に小さな遅延を適用する方法の例を示します。簡単に言うと、この例では、CICフィルタにより高さによるデシメーションが実行されます。CIC1はADFに含まれるCICを表し、CIC0は別のMDFMDFインスタンスのフィルタ（製品内にある場合）を表します。

図362. プログラム可能な遅延



ADFのCIC（CIC1）は、3つの受信サンプルをスキップするためのコマンドを受信します。そのため、入力サンプルb10、b11、およびb12はCIC1によって処理されません。その結果、CIC0によって生成された出力サンプルN+1は入力サンプルa[23:16] から構築され、CIC1のサンプルN+1は入力サンプルb[26:19] から構築されます。

最後に、スキップされないデータ･ストリームは、ビットストリームの3周期分遅延しているように見えます。

注： **入力データ･ストローブがスキップされるとき、デシメーション･カウンタはフリーズしたままとなります。その結果、CIC1によって提供されるサンプルは少し遅延します。**

スキップするビットストリーム･クロックの周期の数をプログラムするには、次の手順を実行する必要があります。

1. SKPBFが0になるのを待ちます。
2. SKPDLY[6:0] に、スキップさせるビットストリーム･クロックの周期の数を書き込みます。SKPBFフラグがすぐに1になり、SKPDLY[6:0] に入力された遅延値が処理中であることが示されます。
   - DFLT0 がまだ有効になっていない（DFLTEN = 0）場合、DLYロジックはDFLTEN = 1になるのを待ちます。アプリケーションによりDFLTENに1がセットされると、DLYロジックは必要な量のデータ･ストローブのスキップを開始します。
   - DFLT0 がすでに有効になっている（DFLTEN = 1）場合、DLYロジックはすぐに必要な量のデータ･ストローブのスキップを開始します。

   ADFで必要な量のデータ･ストローブのスキップが行われると、SKPBFは0に戻ります。
3. アプリケーションでさらにデータ･ストローブのスキップが必要な場合、ステップ 1 から操作を繰り返す必要があります。

このメカニズムで実行された遅延の影響は、ADFが有効である限り、蓄積されます。アプリケーションによりD1遅延が実行され、続いてD2遅延が実行された場合、それ以外のすべてのアクティブなフィルタはD1 + D2の時間だけ遅延します。

注： **SKPBF = 1の状態でSKPDLY[6:0] に書き込んだ場合、書込み操作は無視されます。**

Cascaded-integrator-comb（CIC）フィルタ

CICデジタル･フィルタはローパス･フィルタを効率的に実装したもので、デシメーションや内挿によく使用されます。CIC周波数応答は$Sinc^N$ 関数に等しいため、Sincフィルタとも呼ばれます。

ADFに組み込まれた$Sinc^N$ デジタル･フィルタは、CICMODに従って、$Sinc^4$ または$Sinc^5$ で設定できます。

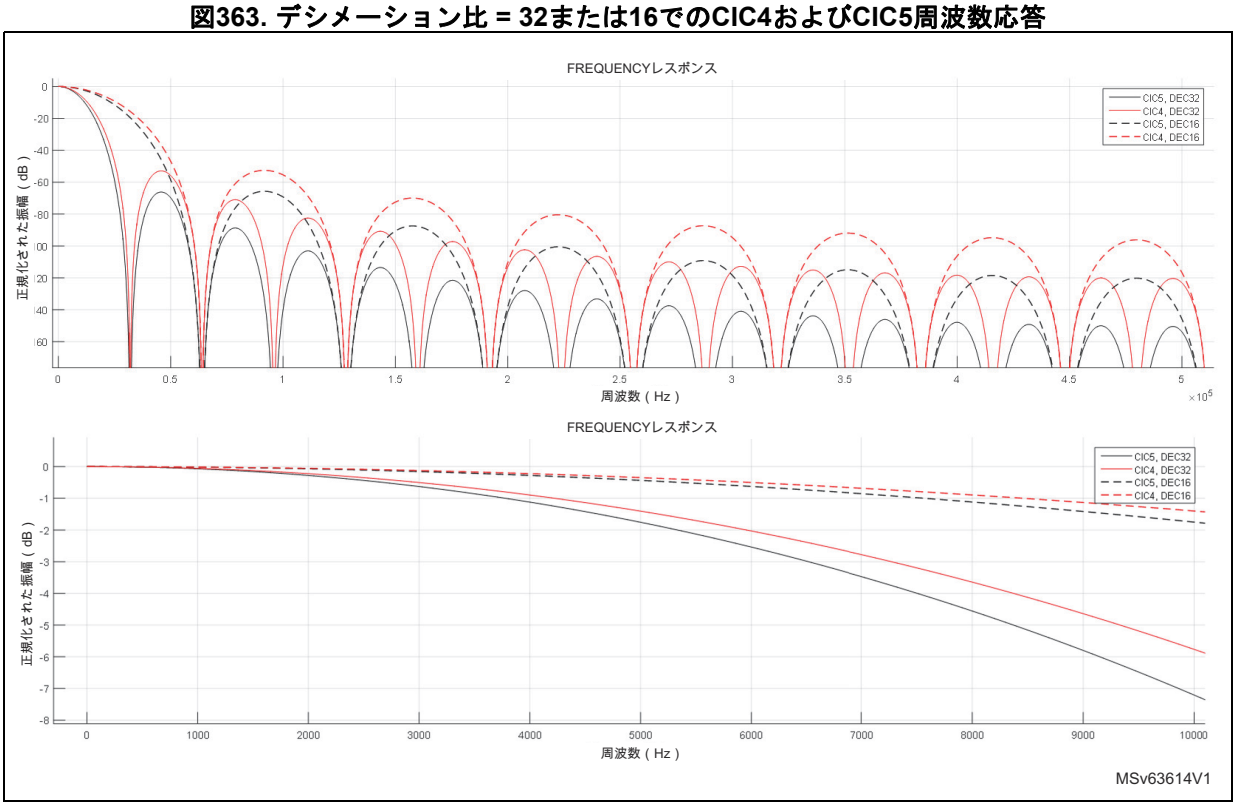
- CICMOD[2:0] = 4の場合、$Sinc^4$ が選択されます。
- CICMOD[2:0] = 5の場合、$Sinc^5$ が選択されます。

フィルタには、次の転送関数が用意されています。

$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-1}}\right)^N$$

ここで、Nは4、または5とすることができ、Dはデシメーション比です。DはMCICD+1と等しくなります。

図363. デシメーション比 = 32または16でのCIC4およびCIC5周波数応答



CIC の出力データ･サイズ

CICによって提供されるサンプルのサイズ（DS_{CIC}）は、次のパラメータによって決まります。

- CICの次数（N）
- CICのデシメーション比（D）
- 入力ストリームのデータ･サイズ（DSIN）

データ･サイズが26bitのCICの容量を超えないことを保証するため、CICの次数とデシメーション比をプログラムする必要があります。

次の式では、上記のパラメータに従って出力データ･サイズが算出できます（DS_{CIC}）。

$$DS_{CIC} = \left(\frac{N \times Ln(D)}{Ln(2)}\right) + DS_{IN}$$

また、CICゲインは次の式で算出できます。

$$G_{CIC} = (D)^{N}$$

デシメーション比は、CICフィルタに対して2 ~ 512の範囲で調整できます。

次の表に、データ･ソースがシリアル･インタフェースまたは12bit ADCからの最大測定範囲の信号である場合の、一部のデシメーション値の一部のデータ出力サイズ（ビット単位）を示します。

注： **シリアル･ビットストリームではDS_{IN} = 1bitですが、ADCITFから受信する場合、最大16bitとすることができます。**

表390.CICの次数とCICのデシメーション値に従ったデータ･サイズ

デシメーション	DS_{IN} = 1bitの場合のデータ･サイズ（ビット）（SITF0 からのデータ）		DS_{IN} = 12bitの場合のデータ･サイズ（ビット）（ADCITFからのデータ）	
	$Sinc^4$	$Sinc^5$	$Sinc^4$	$Sinc^5$
4	9	11	20	22
8	13	16	24	-
10	15	18	26	-
12	16	19	-	-
16	17	21	-	-
20	19	23	-	-
24	20	24	-	-
32	21	26	-	-
48	24	-	-	-
64	25	-	-	-
76	26	-	-	-

注： **最大測定範囲の入力信号では、Sinc^4 の場合はデシメーション比が76を超えないようにし、Sinc^5 の場合は32を超えないようにする必要があります。**

CICによって提供されるデータのLSB部分は、必ずしも重要ではありません。これはセンサのパフォーマンスと、帯域外のノイズを除去するCICの能力によって変わります。

CIC出力でのサンプル·サイズは、SCALEブロックを使用して調整できます。

スケーリング（SCALE）と飽和（SAT）

SCALEブロックを使用すると、CICによって提供される信号の振幅を、アプリケーションにより3dB（± 0.5dB）刻みで調整することができます。

信号の振幅は、最大で8bit（- 48.2dB）減らし、最大で12bit（+ 72.2dB）増やすことができます。

ゲインは、ADFデジタル·フィルタ設定レジスタ0（ADF_DFLT0CICR）のSCALE[5:0] ビットフィールドによって調整されます。

SCALE[5:0] は、DFLT0 が有効化されている場合でも変更できます。ゲインの遷移中、フィルタによって提供される信号は中断されます。

内部的な再同期により、アプリケーションで新しいゲインを書き込んでから、ゲインが効果的にサンプルに適合されるまでの間に、adf_proc_ckクロックの一部のサイクルで遅延が発生します。前の値がまだ適用されていないうちにアプリケーションで新しいゲイン値を書き込もうとした場合、この新しいゲイン値は無視されます。SCALE[5:0] を読み戻すと、アプリケーションに現在のゲイン値が通知されます。

次の表に、使用可能なゲイン値を示します。

表391.使用可能なゲイン値

SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）	SCALE[5:0]	ゲイン（dB）
0x20	- 48.2	0x2B	- 14.5	0x06	+ 18.1	0x11	+ 51.7
0x21	- 44.6	0x2C	- 12.0	0x07	+ 21.6	0x12	+ 54.2
0x22	- 42.1	0x2D	- 8.5	0x08	+ 24.1	0x13	+ 57.7
0x23	- 38.6	0x2E	- 6.0	0x09	+ 27.6	0x14	+ 60.2
0x24	- 36.1	0x2F	- 2.5	0x0A	+ 30.1	0x15	+ 63.7
0x25	- 32.6	0x00	0.0	0x0B	+ 33.6	0x16	+ 66.2
0x26	- 30.1	0x01	+ 3.5	0x0C	+ 36.1	0x17	+ 69.7
0x27	- 26.6	0x02	+ 6.0	0x0D	+ 39.6	0x18	+ 72.2
0x28	- 24.1	0x03	+ 9.5	0x0E	+ 42.1	-	-
0x29	- 20.6	0x04	+ 12.0	0x0F	+ 45.7	-	-
0x2A	- 18.1	0x05	+ 15.6	0x10	+ 48.2	-	-

SATブロックは、コードが最大値を下回るか、最小値を下回った場合に、バイナリ·コードのラップアラウンドが発生しないようにできます。

ADFは、次のレベルで飽和操作を実行します。

- SCALEブロックの後（SATブロックにより実行）：信号は24bitで飽和します。
- RSFLT内。フィルタの正常な動作を保証します。
- HPFの出力。出力信号が24bitを超えないことを保証します。

SATFビットはアプリケーションに、SCALEの後、RSFLTの内部、またはHPFの後に飽和が発生したことを通知します。また、SATIEに1がセットされている場合、割込みを生成できます。飽和が検出されるとすぐに、SATFフラグに1がセットされます。新しい飽和を検出できるようにするには、アプリケーションでこのフラグをクリアする必要があります。

これらのビットは、ADF DFLT0割込み有効化レジスタ（ADF_DFLT0IER）および ADF DFLT0割込みステータス･レジスタ0 （ADF_DFLT0ISR）内にあります。

ゲイン調整ポリシー

ADFの最高のパフォーマンスを得るには、SCALE[5:0] を通してゲイン値を適切に調整することが重要です。

通常、ゲインを調整するには、最大入力信号については、最終的な信号振幅が24bitの最大測定範囲にできるだけ近くなるようなSCALE[5:0] の値を選択します。

最適なゲインを選択するための方法は、次のとおりです。

1. 予想される入力信号について、CICフィルタへのデータ･サイズが26bitを超えないことを確認してください。確認は、次の式でできます。

$$\frac{LN(SIN_{pp} \cdot D^N)}{LN(2)} < 26$$

ここで、NはCICの次数、Dはデシメーション比、SIN_{pp} は入力信号のピークツーピークの最大振幅です。

SIN_{pp} は次の値とすることができます。

– SITF0 から受信するサンプルの場合、ピークツーピークの最大振幅は2（± 1）

– 12bit ADCから受信するサンプルの場合、ピークツーピークの最大振幅は4095（+ 2047、- 2048）

例 : 最大入力信号が ± 0.35を超えない場合、$Sinc^4$ ではデシメーション比を96とすることができます。実際には次のようになります。

$$\frac{LN(0.7 \cdot 96^4)}{LN(2)} \sim 25.82 \text{ bits} < 26 \text{ ビット}$$

2. SCALE値を調整します。

適切なSCALE値を選択するには、RSFLTが使用されているかどうかを確認する必要があります。RSFLTが使用されている場合、SCALE出力でのデータ･サイズは22bitを超えないようにする必要があり、使用されていない場合は最大で24bitとすることができます。

次の式を使用して、dB単位のSCALE値を選択する必要があります。

$$SCALE_{dB} < 20 \cdot \log 10\left(\frac{2^{NB}}{SIN_{pp} \cdot D^{N}}\right)$$

ここで、RSFLTが有効化されている場合、NBは22になり、RSFLTがバイパスされている場合は24になります。$SCALE_{dB}$ は、SCALE[5:0] によって選択されたゲイン値です。

例：デシメーション比が96、SIN_{pp} が0.7の$Sinc^4$ の場合、次のようになります。

– RSFLTがバイパスされている場合：

$$20 \cdot \log 10\left(\frac{2^{24}}{0.7 \cdot 96^{4}}\right) \sim -11\ dB$$

$SCALE_{dB}$ 値は - 11dB未満でなければなりません。それより小さくて最も近い値は - 12dB（SCALE[5:0] = 0x2C）です。

– RSFLTが有効な場合：

$$20 \cdot \log 10\left(\frac{2^{22}}{0.7 \cdot 96^{4}}\right) \sim -23\ dB$$

$SCALE_{dB}$ 値は - 23dB未満でなければなりません。それより小さくて最も近い値は - 24.1dB（SCALE[5:0] = 0x28）です。

SCALE[5:0] にセットされた値がそれより大きい場合、飽和が発生する可能性があります。イベント･フラグにより、飽和が発生しているかどうかが通知されます。

次の表に、データをSITF0 から受信した場合の、MCICの次数とMCICのデシメーション比に従った、フィルタ設定ごとのゲイン値を示します。この表は包括的なものではなく最大測定範囲の入力信号を考慮したものです（詳細は、「セクション 40.7.5：総ADFゲイン」を参照）。

表392.推奨の最大ゲイン値とCICのデシメーション比

CICのデシメーション比	SITF + CICx + RSFLT（+ HPF）構成のゲイン設定（dB）		SITF + CICx（+ HPF）構成のゲイン設定（dB）	
	CIC5	CIC4	CIC5	CIC4
8	33.6	51.7	45.7	63.7
12	18.1	39.6	30.1	51.7
16	3.5	27.6	15.6	39.6
20	- 6.0	21.6	6.0	33.6
24	- 12.0	15.6	0	27.6
28	- 20.6	9.5	- 8.5	21.6
32	- 26.6	3.5	- 4.5	15.6

表392.推奨の最大ゲイン値とCICのデシメーション比 （続き）

CICのデシメーション比	SITF + CICx + RSFLT（+ HPF）構成のゲイン設定（dB）		SITF + CICx（+ HPF）構成のゲイン設定（dB）	
	CIC5	CIC4	CIC5	CIC4
48	-	- 8.5	-	3.5
64	-	- 20.6	-	- 8.5

再形成フィルタ （RSFLT）

ADFは、CICを備えているほか、主にオーディオアプリケーション用で他の用途にも使用できる再形成IIRフィルタも備えています。

RSFLTを使用する場合、入力でのサンプル･サイズは22bitを超えないようにする必要があります。

RSFLT出力でのサンプルは、ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）のRSFLTDビットに従って、4でデシメーションするかどうかを決定することができます。

RSFLTをバイパスするには、ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）のRSFBYPに1をセットします。

次の表に、一般的なオーディオ･ストリームを処理するためにRSFLTに必要なサンプリング･レートを示します。

RSFLT遮断周波数（F_C）は、入力でのサンプル･レート（F_{RS}）によって決まり、次の式で算出されます。

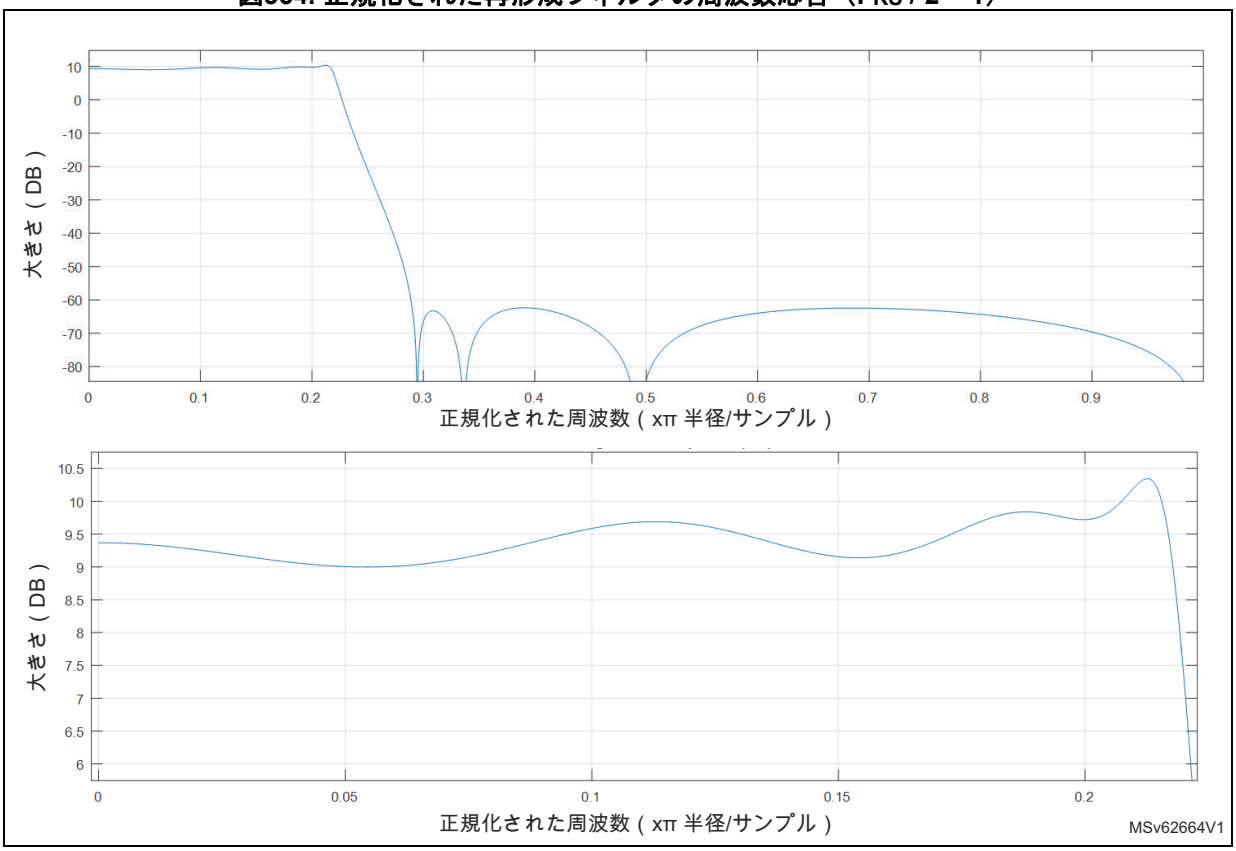
$$F_C = 0.111 \times F_{RS}$$

表393.一般的なマイクロフォンの設定

RSFLTでのサンプル･レート（kHz）（F_{RS}）	通過帯域（kHz）	D2	PCMサンプル･レート（kHz）
32	3.55	4	8
64	7.1	4	16
128	14.2	4	32
192	21.3	4	48

次の図に、再形成フィルタの周波数応答を示します。

図364. 正規化された再形成フィルタの周波数応答（FRS / 2 = 1）



RSFLTゲインは約9.3dBであるため、出力データ･サイズは、22bit幅の入力信号では24bitよりやや小さくなります。

F_{RS} で1つのサンプルを処理するには、RSFLTでは24クロックサイクルのadf_proc_ckクロックが必要です。RSFLTを有効化した場合、adf_proc_ckの速度がF_{RS} の少なくとも24倍であることがアプリケーションによって確認される必要があります。

前のサンプルの処理中にRSFLTで新しいサンプルを受信した場合、RSFLTによりイベント（rfovr_evt）が生成され、RFOVRFフラグがセットされます。

RFOVRFがセットされた場合、RSFLTによって提供されたサンプルは無効です。アプリケーションによってデータ取得が停止され、高速のadf_proc_ckクロックがRSFLTに提供される必要があります。

ハイパス･フィルタ（HPF）

ハイパス･フィルタは、連続変換モードの場合に、最終出力データ･ストリームの低周波数成分を抑制します。ハイパス･フィルタの有効化／無効化は、ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）のHPFBYPを通して行えます。

HPFは、入力データ･ソース中に寄生的な低周波数ノイズ（またはDC信号）があり、最終データから除去する必要がある場合に有益です。

HPFは一次IIRフィルタで、遮断周波数は ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）の HPFC[1:0] を通して、次の値から選択することができます。

- 0.000625 x F_{PCM}

- 0.00125 x F_{PCM}
- 0.00250 x F_{PCM}
- 0.00950 x F_{PCM}

表394.HPF 3dB遮断周波数の例

HPFC	共通F_{PCM}周波数（Hz）用の3dBの遮断周波数		
	F_{PCM} = 8kHz	F_{PCM} = 16kHz	F_{PCM} = 48kHz
0	5	10	30
1	10	20	60
2	20	40	120
3	76	152	456

HPF出力は24bitで飽和します。サンプルが飽和すると、SATFフラグがセットされます。

40.4.8 デジタル･フィルタ取得モード

ADFでは、データ･キャプチャ実行用の次のモードが用意されています。

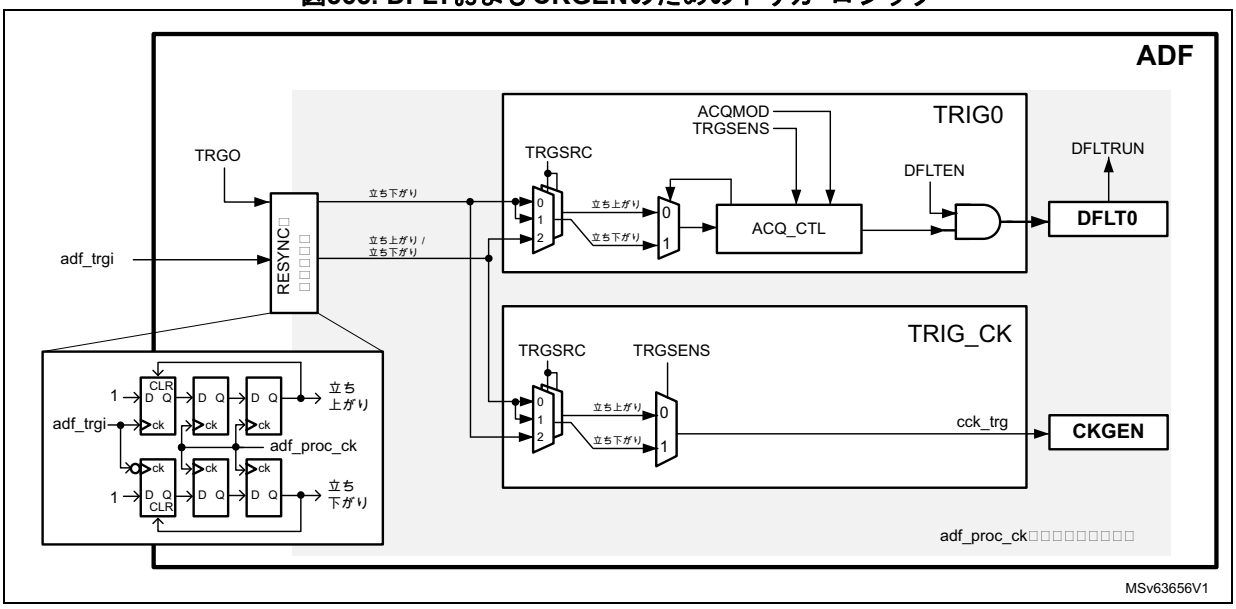
- 非同期連続取得モード
- 非同期シングルショット取得モード
- 同期連続取得モード
- 同期シングルショット取得モード
- ウィンドウ連続取得モード

注： **データ･キャプチャを実行するには、フィルタ、データを提供するインタフェース（SITF0　または ADCITF）、CKGENを有効化する必要があります。必要に応じて、ADF_CCK0またはADF_CCK1も有効化する必要があります。**

DFLTENに0をセットした場合、フィルタをすぐに停止することができます。このアクションにより、フィルタがリセットされ、RXFIFO が一掃されます。RXFIFOとフィルタがリセットされると、DFLTACTIVEフラグも0に戻ります。

次の図に、各フィルタとクロック･ジェネレータで利用できるトリガ･ロジックの概要を示します。

図365. DFLTおよびCKGENのためのトリガ･ロジック



すべてのTRIGブロックに共通のブロックにより、立ち上がりエッジおよび立ち下がりエッジの検出と、adf_ker_ckクロック･ドメインへの入力トリガの再同期化が実行されます。この実装により、アプリケーションではadf_ker_ck周期より小さなパルス幅を持つトリガを使用できるようになります。

同期モードでは、TRIGブロックを使用することで、トリガ･ソースとしてadf_trgiまたはTRGOビットを選択できます。TRGOビットは、ADFグローバル制御レジスタ（ADF_GCR）内にあります。

エッジの感度も選択できます。

非同期連続取得モード

このモードでは、DFLTENビットに1を書き込むだけで、アプリケーションにより連続取得を開始することができます。

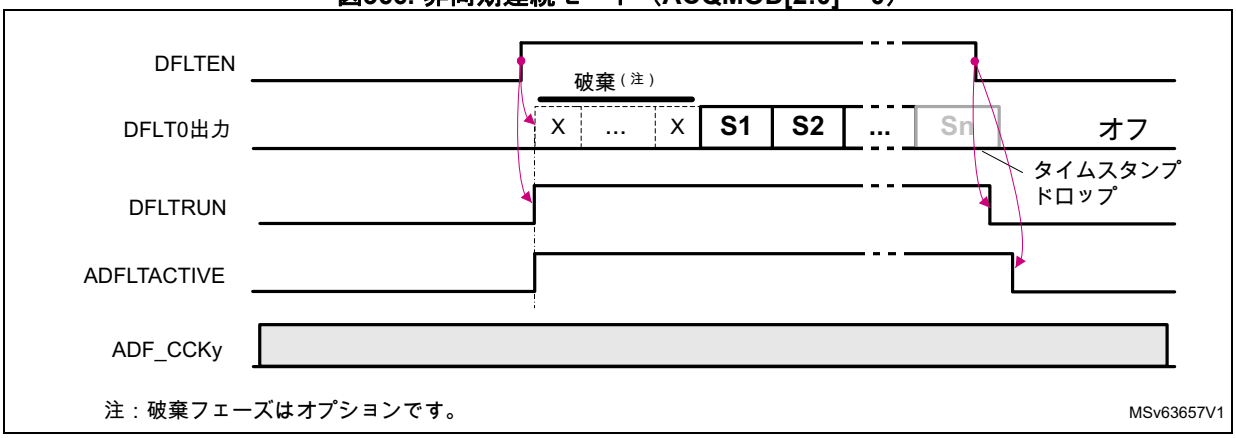
非同期連続取得モードは、ACQMOD[2:0] = 0のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（DFLTEN に0がセットされていることが前提です）。

1. adf_proc_ck周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表398の例を参照）。
2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に0をセットします。
4. シリアル･データ･インタフェースのSITFENビットに1をセットします。
5. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
6. DFLTENに1がセットされると、すぐに取得シーケンスが開始されます。

次の図に、DFLT0 によって生成されるサンプルの簡単な例を示します。

図366. 非同期連続モード（ACQMOD[2:0] = 0）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

非同期シングルショット取得モード

このモードでは、DFLTENビットに1を書き込むだけで、アプリケーションにより1つのサンプルの取得を開始することができます。

非同期シングルショット取得モードは、ACQMOD[2:0] = 001のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（DFLTEN に0がセットされていることが前提です）。

1. adf_proc_ck周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表398の例を参照）。
2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に001をセットします。
4. SITFENビットに1をセットします。
5. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
6. DFLTENに1をセットすると、フィルタによりRXFIFOにデータが1つ提供され、取得が停止します。

新しい取得をトリガするには、アプリケーションで次のことを実行する必要があります。

1. DFLTRUN = 0になるのを待って、前の取得が完了したことを確認します。
2. DFLTENに再度1をセットします。

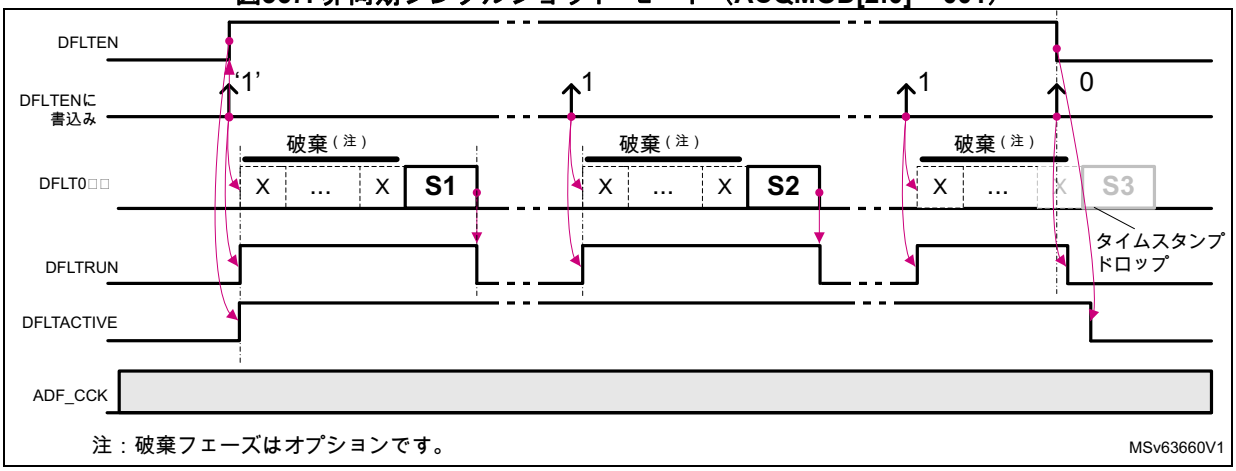
このシーケンスは、新しいデータの変換が必要になるたびに繰り返すことができます。

図366に示すように、DFLTENに1がセットされるたびに、取得シーケンスがトリガされます。必要に応じて、フィルタによって提供される最初のサンプルを破棄することができます。各変換の最後に、デシメーション･カウンタとフィルタ･タップがリセットされ、フィルタは新しい変換の開始が可能になります。

取得の実行中にDFLTENに0がセットされた場合、実行中の変換は停止します（この例では、S3は失われます）。この状況を回避するには、次の手順を実行します。

1. DFLTRUN = 0を待ちます。
2. RXFIFOからサンプルを読み出します。
3. DFLTENを0にセットします。

図367. 非同期シングルショット･モード（ACQMOD[2:0] = 001）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

同期連続取得モード

このモードでは、次のいずれかのトリガ･ソースを使用して、アプリケーションにより連続取得を開始することができます。

- adf_trgi信号
- TRGOビット

同期連続取得モードは、ACQMOD[2:0] = 010のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（DFLTEN に0がセットされていることが前提です）。

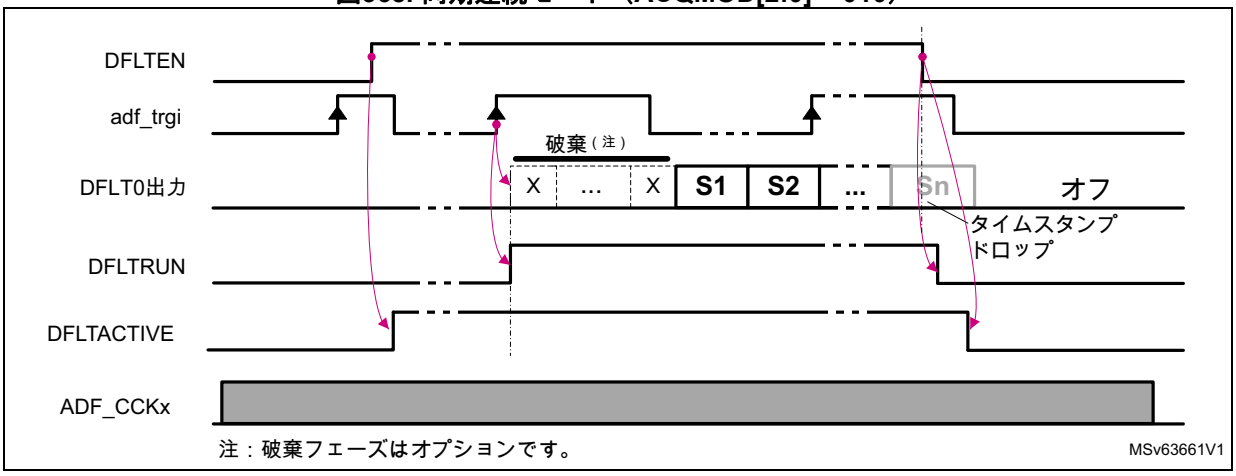
1. adf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表398の例を参照）。
2. CKGENを有効化し（CKGDEN = 1）、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に010をセットします。
4. SITFENビットに1をセットします。
5. 適切なトリガ･ソースと感度を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
7. DFLTENに1をセットします。
8. トリガの条件が満たされると、フィルタで取得が開始されます。

TRGSENSビットでは、トリガのエッジ（立ち上がりまたは立ち下がり）を選択できます。取得の実行中、またはDFLTENに0がセットされている場合、トリガは無視されます。

次の図に、トリガ･ロジックが立ち上がりエッジのトリガの影響を受ける（TRGSENS = 0）場合の簡単な例を示します。DFLTEN = 0であるため、トリガ信号の最初の立ち上がりエッジは無視されます。

次の立ち上がりエッジは考慮され、取得が開始されます。その他すべての立ち上がりエッジは無視されます。DFLTRUNが0に戻ると、トリガ･ロジックは再初期化されます。

図368. 同期連続モード（ACQMOD[2:0] = 010）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

同期シングルショット取得モード

このモードでは、次のいずれかのトリガ･ソースを使用して、アプリケーションによりシングル取得を開始することができます。

- adf_trgi信号
- TRGOビット

同期シングルショット取得モードは、ACQMOD[2:0] = 011のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（DFLTEN に0がセットされていることが前提です）。

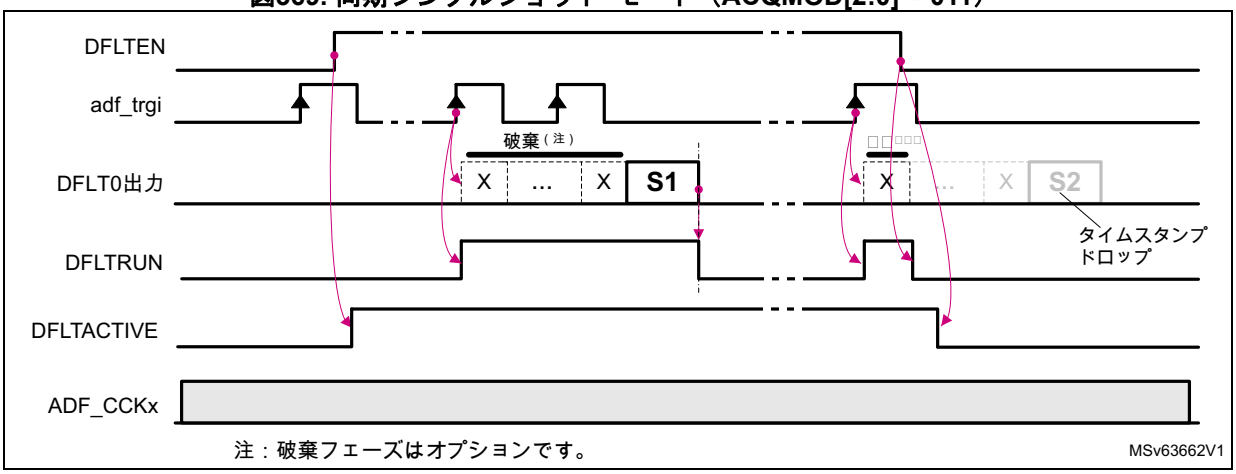
1. adf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表398の例を参照）。
2. CKGDENを有効化し、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に011をセットします。
4. SITFENビットに1をセットします。
5. 適切なトリガ･ソースと感度を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
7. DFLTENに1をセットします。
8. トリガ条件が満たされると、フィルタで取得が開始され、RXFIFOにデータが1つ提供され、フィルタで新しいトリガを受け付けられる状態になります。

TRGSENSでは、トリガのエッジ（立ち上がりまたは立ち下がり）を選択できます。取得の実行中、またはDFLTENに0がセットされている場合、トリガは無視されます。

次の図に、トリガ･ロジックが立ち上がりエッジのトリガの影響を受ける（TRGSENS = 0）場合の簡単な例を示します。DFLTEN = 1でトリガの立ち上がりエッジが検出されるたびに、取得シーケンスがトリガされます。必要に応じて、フィルタによって提供される最初のサンプルを破棄することがで

きます。各変換の終了時に、デシメーション･カウンタとフィルタ･タップがリセットされます。DFLTRUNに0がセットされ、フィルタは新しい変換の開始が可能になります。

図369. 同期シングルショット･モード（ACQMOD[2:0] = 011）



注： **取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。**

図368に、取得の実行中にDFLTENに0がセットされたため、サンプルS2が失われるケースを示します。この状況を回避するには、次の手順を実行します。

1. DFLTRUN = 0を待ちます。
2. RXFIFOからサンプルを読み出します。
3. DFLTENを0にクリアします。

ウィンドウ連続取得モード

このモードでは、アプリケーションにより、次のいずれかのトリガ･ソースの連続エッジで制御される連続取得を開始または停止することができます。

- adf_trgi信号
- TRGOビット

ウィンドウ連続取得モードは、ACQMOD[2:0] = 100のときに選択されます。

次のシーケンスに、最も重要なプログラム手順を示します（DFLTEN に0がセットされていることが前提です）。

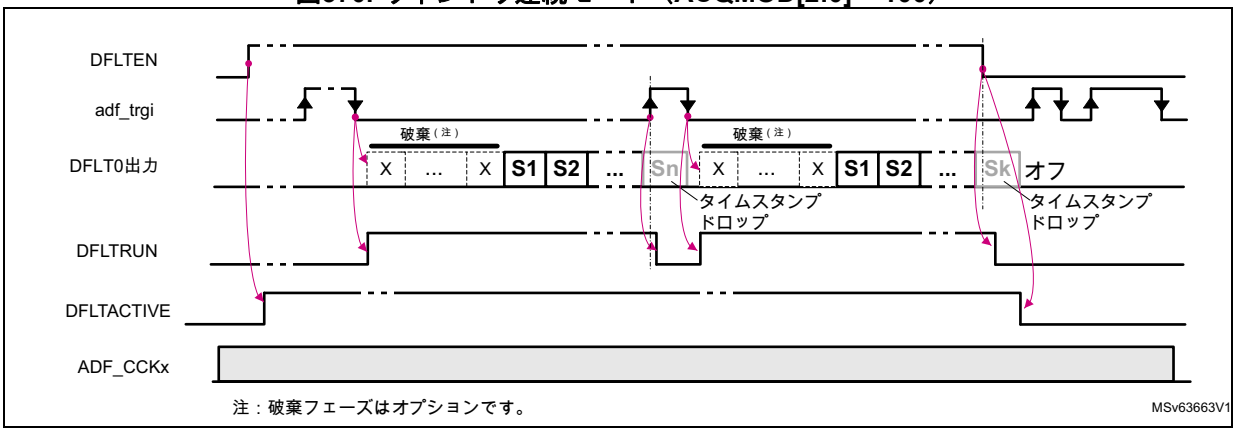
1. adf_proc_ck クロックの周波数が対象のアプリケーションに対応するように、クロック･ジェネレータ（CKGEN）を設定し、有効化します（表398の例を参照）。
2. CKGDENを有効化し、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
3. フィルタ設定をプログラムし、ACQMOD[2:0] に100をセットします。
4. SITFENビットに1をセットします。
5. 適切なトリガ･ソースと感度を選択します。
6. DFLTENに1をセットする前に、DFLTACTIVE = 0になるのを待ちます。これにより、前のフィルタ無効化シーケンスが適切に終了することを確認できます。
7. DFLTENに1をセットします。

8. TRGSENS = 0の場合、取得はトリガの立ち上がりエッジで開始され、トリガの立ち下がりエッジで停止します。TRGSENS = 1の場合、取得はトリガの立ち下がりエッジで開始され、トリガの立ち上がりエッジで停止します。

注：　トリガ条件が再び有効になると、取得が再開可能になります。

図369に、TRGSENS = 1でのウィンドウ連続取得モードの簡単な例を示します。DFLTENに1がセットされると、ADFは選択されたトリガ入力で立ち下がりエッジを待ちます。トリガ条件が満たされると、DFLTRUNが1に戻り、取得が開始されます。ADFにより、選択されたトリガ入力で立ち上がりエッジが検出されると、取得は停止します。DFLTENに1がセットされたままであれば、ADFは選択されたトリガ入力で再び立ち下がりエッジを待ちます。

図370. ウィンドウ連続モード（ACQMOD[2:0] = 100）



注：　取得を停止するには、DFLTENを0に戻します。これにより、フィルタがリセットされ、RXFIFOが一掃されるため、RXFIFOに格納されているサンプルは失われます。実行中のDMA転送は適切に終了します。フィルタ･チェーンがリセットされ、RXFIFOが一掃されると、DFLTACTIVEは0に戻ります。

複数のフィルタの同期開始

ADFをMDFインスタンス（製品内にある場合）と一緒に使用する場合、すべてのフィルタの取得を同時に開始することができます。この同期機能は、製品内でのトリガの接続方法によって決まります。一般的に、ADFのadf_trgo信号がMDFインスタンスへのトリガ入力として接続されている場合、ADFでMDFインスタンスをトリガすることができます（トリガ機能を確認するには、「セクション 40.4.2 : ADFピンおよび内部信号」を参照）。

次のプログラミング例では、1つのADFのadf_trgo信号がいくつかのMDFに接続されています。複数のフィルタの取得を同期させて開始するには、次のシーケンスを実行する必要があります（フィルタのDFLTENビットに0がセットされていることが前提です）。

adf_trgoトリガを受信するMDFで次のことを行います。

1. CKGENを有効化し（CKGDEN = 1）、必要に応じて、ADF_CCK0およびADF_CCK1クロックを有効化します。
2. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
3. 各フィルタについて、取得モードを「同期」にセットします（ACQMOD[2:0] = 01x）。
4. 各フィルタについて、adf_trgoトリガ入力を選択するために、TRGSRC[3:0] をセットします。
5. 各フィルタについて、TRGSENSに0をセットします（立ち上がりエッジ）。
6. 各フィルタについて、DFLTENに1をセットします。

adf_trgoトリガを生成するADFで次のことを行います。

1. CKGENを有効化し（CKGDEN = 1）、必要に応じて、ADF_CCK[1:0] クロックを有効化します。

2. リクエストされたデータ･インタフェースのSITFENビットに1をセットします。
3. 取得モードを「同期」にセットします（ACQMOD[2:0] = 01x）。
4. TRGSRC[3:0] に0をセットします（TRGOを選択）。
5. TRGSENSに0をセットします（立ち上がりエッジ）。
6. DFLTENを1にセットします。
7. 0が読み出されるまで、TRGOビットを読み出します。
8. TRGOを1にセットします。すぐに、選択したすべてのフィルタの取得シーケンスが開始されます。

新しい取得をトリガするには（シングルショットの場合）アプリケーションで次のことが行われる必要があります。

- DFLTRUN = 0になるのを待って、前の取得が完了したことを確認します。
- 0が読み出されるまで、TRGOを読み出します。
- 再度、TRGOビットに1をセットします。

破棄するサンプル

ADFでは、次の目的で、再起動後に破棄するサンプルの数をプログラムすることができます。

- フィルタのインパルス応答の影響を受けたサンプルのキャプチャを回避するため
- 特定のサンプルの数だけフィルタの取得を遅延させるため

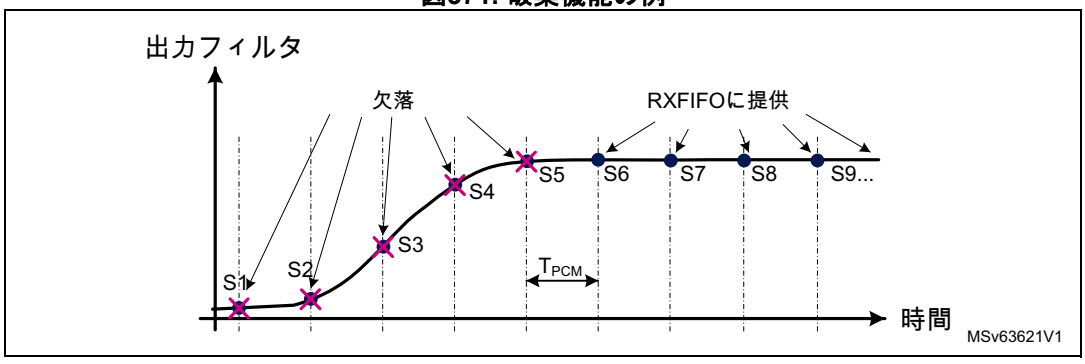
破棄機能は、次のようにNBDIS[7:0] を通して制御されます。

- NBDIS[7:0] = 0の場合、破棄機能は無効化されます。
- NBDIS[7:0] ≠ 0の場合、次のいずれかの条件で破棄機能は有効化されます。
  - DFLTENビットが1に変更されたとき
  - （非）同期シングルショット･モードで取得が開始されるたび

図365~図369と図371を参照してください。

次の図に示す例では、デジタル･フィルタによって提供される最初の5つのサンプル（S1 ~ S5）を破棄するために破棄機能が使用されています。RXFIFOに最初に転送されるサンプルはS6です。

図371. 破棄機能の例



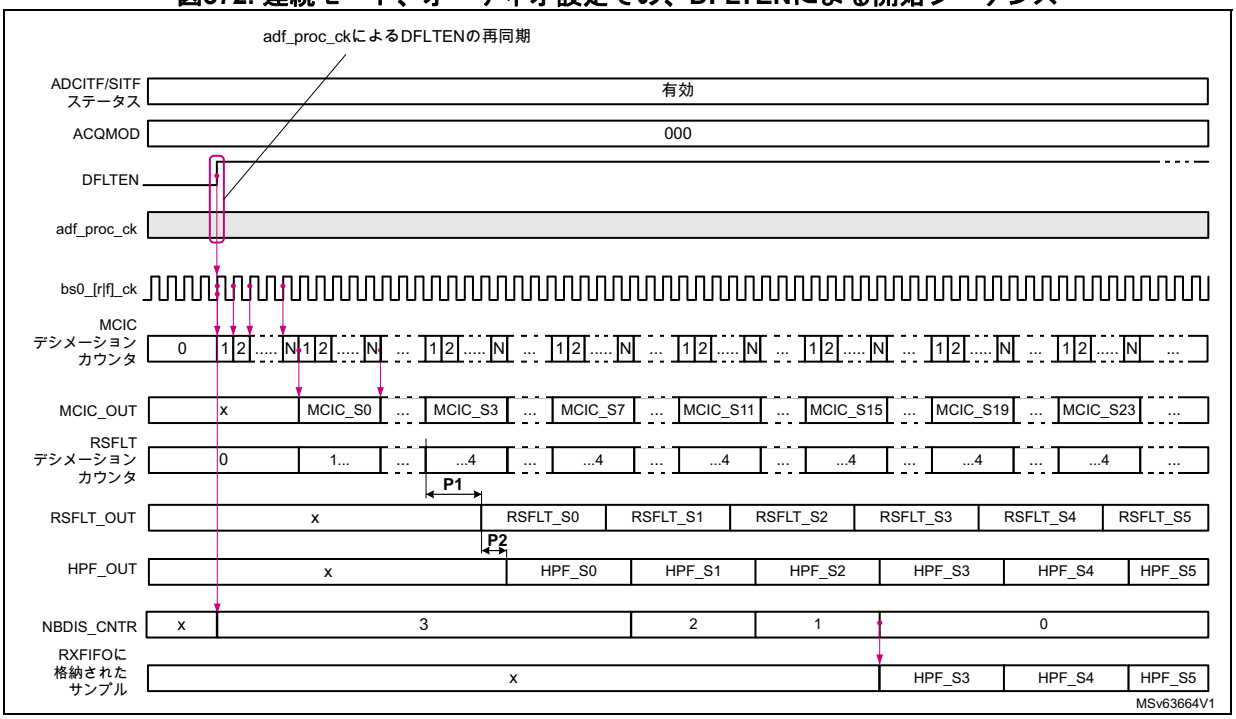
40.4.9 開始シーケンスの例

次の図に、NBDIS[7:0] = 3（取得の前に3つのサンプルを破棄）という設定で、DFLTEN（ACQMOD[2:0] = 0）によりトリガされるデジタル･フィルタの取得シーケンス開始の詳細を示します。

DFLT0 がオーディオ･アプリケーション用に設定され、MCIC、RSFLT、およびHPFが有効化されます。データ･インタフェース（SITF0 またはADCITF）はすでに有効化されていることが前提となります。

注： **図を簡素化するため、NBDIS[7:0] には意図的に低い値をセットします。**

図372. 連続モード、オーディオ設定での、DFLTENによる開始シーケンス



DFLTENビットは、ADF処理クロック･ドメインに再サンプリングされます。DFLTENがハイとして検出されると、フィルタ･チェーンが有効化され、MCICフィルタのデシメーション･カウンタがビットストリーム･クロックのレートでインクリメントされます。

MCICデシメーション･カウンタがプログラムされた値Nに達すると、RSFLTでサンプルが使用可能になります。

RSFLTは、MCICによって提供されるすべてのサンプルを処理し、4つのサンプルを処理するたびにHPFにサンプルを送信します（4でデシメーション）。RSFLTではサンプル（P1）を送信する前に、最大24サイクルのadf_proc_ckクロックが必要です。

HPFはRSFLTによって提供されたすべてのサンプルを処理しますが、NBDIS_CNTRが0に達していない間は、NBDIS機能によってRXFIFOへのデータの書込みが防止されます。

NBDIS_CNTRが0に達すると、HPFによって提供されるサンプルはRXFIFOに格納されます。

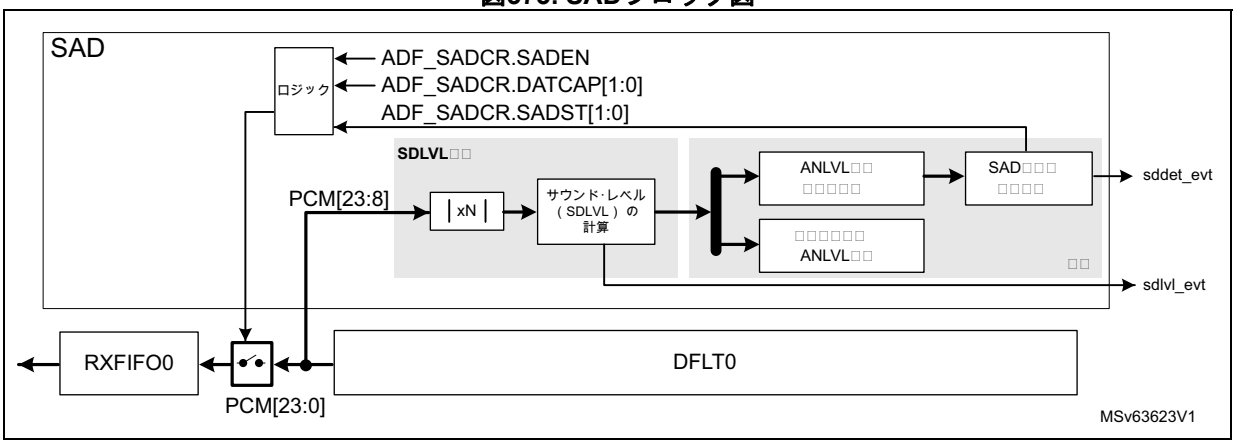
40.4.10 サウンド･アクティビティ検出（SAD）

SADは環境雑音レベル（ANLVL）と短期サウンド･レベル（SDLVL）の計算に基づいています。SADでは、次の場合にサウンドが検出されます。

- SDLVLが環境雑音レベルと呼ばれる閾値に達したとき
- SDLVLが固定の閾値に達したとき
- ANLVLが固定の閾値に達したとき

次の図に示すように、SADではDFLT0出量から得られる16個のMSBサンプルを使用します。

図373. SADブロック図



SADでは高度な設定が可能で、アプリケーションで複数のパラメータを調整できます。

- SAD検出動作（SADMOD）
- サウンド･レベルの計算に使用するサンプルの数（FRSIZE）
- 学習フェーズ中に環境雑音レベルの計算に使用するフレームの数（LFRNB）
- 環境雑音推定機能のスロープ（ANSLP）
- 最小予測環境騒音レベル（ANMIN）
- 閾値レベル（SNTHR）
- 閾値ヒステリシス（HYSTEN）
- DETECT 状態と MONITOR 状態の間の疑似遷移をフィルタリングするためのハングオーバー･ウィンドウ（HGOVR）
- データ･キャプチャ･モード（DATCAP）

SAD検出動作

SADでは、次のように、SADMOD[1:0] によって選択されたサウンドを検出することができます。

- SADMOD[1:0] = 0の場合、SADは音声アクティビティ検出のように機能します。このモードでは、SADは算出されたサウンド･レベル値に従って、環境雑音レベルを推定します。トリガの閾値は推定された環境雑音から導出されます。最後に、現在のサウンド･レベルがこの閾値と比較されます。最初の概算では、入力信号のピーク対平均値がSNTHR[3:0] で定義されたレベルに達した場合にSADがトリガされます。
- SADMOD[1:0] = 01の場合、SADでは、現在のサウンド･レベル（SDLVL）と、アプリケーションによってSNTHR[3:0] とANMIN[12:0] を通して定義された固定のトリガ値とを比較します。このモードでは、サウンドレベルの比較に使用されるサンプルの数はFRSIZE[2:0] を通して設定できるため、SADはすばやく反応できます。

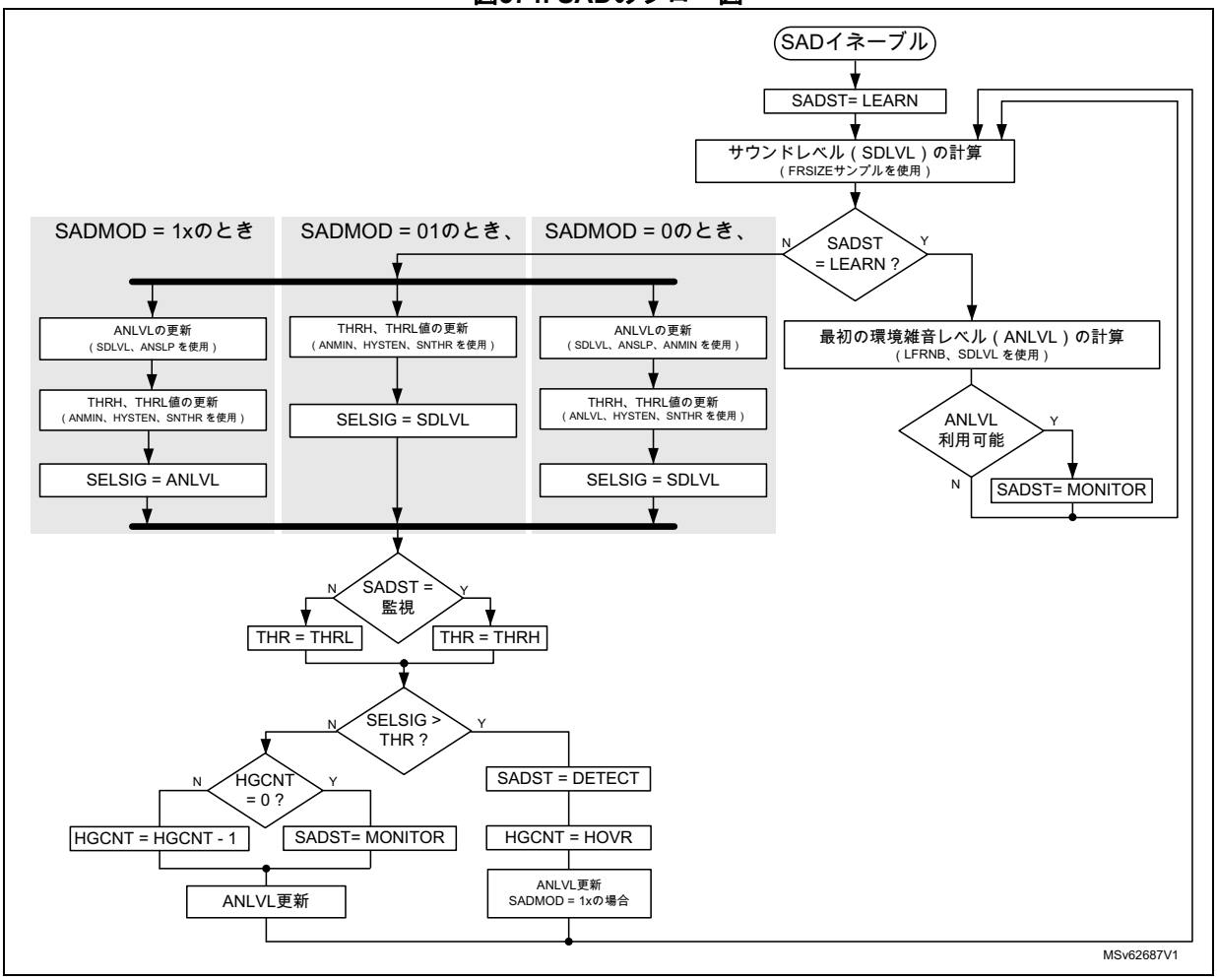
- SADMOD[1:0] = 1xの場合、SADでは、推定環境雑音レベル（ANLVL）と、アプリケーションによってSNTHR[3:0] とANMIN[12:0] を通して定義された固定のトリガ値とを比較します。このモードでは、ピーク･レベルによる望ましくないトリガを回避できますが、入力信号の変化に対するSADの反応は遅くなります。ただし、FRSIZE[2:0] とANSLP[2:0] を通して反応時間を調整することができます。

SADの状態

図373に示すとおり、SADは次のように機能します。

1. 有効化されている(SADEN = 1)場合、SADはまずLEARN状態で最初の環境雑音レベルの推定を行います。
2. SAD では、DFLT0 によって提供されるサンプルを使用して、継続的に短期サウンド･レベル（SDLVL）を計算します。サウンド･レベルの計算に使用するサンプルの数は、FRSIZE[2:0] から取得します。DFLT0によって処理されたサンプルをメモリに転送するかどうかは、DATCAP[1:0] の値によって決まります。
3. 連続するサウンド･レベル値を使用して、最初の環境雑音レベル（ANLVL）が算出されます。最初の環境雑音推定（LFRNB）の計算に使用するサウンド･レベル値の程度は、アプリケーションで定義できます。
4. 最初の環境雑音レベル（ANLVL）の計算中、SADはMONITOR状態に切り替わります。
5. 新しい短期サウンド･レベル値が利用可能になるたびに、選択した検出モードに従って、SADで環境雑音レベルと閾値が更新されます。
6. SADがトリガされると、次のことが行われます。
   - SADがDETECT状態に切り替わります。
   - sddet_evtイベントがアサートされます。
   - adf_sad_det信号がハイにセットされます。
7. HGOVR によって指定された時間について、ハングオーバー機能により、サウンド･レベルが閾値レベルを下回った場合でもDETECT状態が維持されます。

図374. SADのフロー図



サウンド･レベルの計算（SDLVL）

有効化すると、SADによりサウンド･レベル値が継続的に計算されます。サウンド･レベルとは、FRSIZE[2:0] によって指定されるPCMサンプル数の絶対値の平均のことです。

$$SDLVL = \frac{1}{N_{FRSIZE}} \times \sum_{n=1}^{N_{FRSIZE}} |PCM(n)|$$

ここで、N_{FRSIZE} はFRSIZE[2:0] によって指定されるPCMサンプルの数です。

環境雑音推定（ANLVL）

環境雑音レベル（ANLVL）は、SADMOD[1:0] が00または10の場合に算出されます。

以下に示すように、環境雑音レベルの計算方法はSADの状態によって異なります。

- LEARN状態でのANLVLの計算

  SADが有効化されるたびに、環境雑音レベルの最初の値を推定するため、学習フェーズが開始されます。このフェーズ中、SADはトリガできません。

  LEARNフェーズ中、環境雑音レベルは次のように算出されます。

$$ANLVL = \frac{1}{N_{LFRNB}} \times \sum_{n=1}^{N_{LFRNB}} |SDLVL(n)|$$

  ここで、N_{LFRNB} はLFRNB[2:0] ビットフィールドによって指定されるフレームの数です。

- MONITORまたはDETECT状態でのANLVLの計算

  学習フェーズが完了すると、SADでは次のように環境雑音レベルが更新されます。

  a) SADにより環境雑音レベル用の新しい値が算出されます。

  ANLVL_UP = ANLVL * (1+2^(ANSLP-12))

  ANLVL_DN = ANLVL * (1-2^(ANSLP-10))

  b) 現在のサウンド･レベルがANLVL_DNより低い場合はANLVLとしてANLVL=DN値が使用され、それ以外の場合はANLVL_UPの値が使用されます。

  現在のサウンド･レベルが閾値レベルより高い場合はANLVLは更新されません。ただし、SADMOD[1:0] = 10の場合を除きます。

  c) SADMOD[1:0] = 0のとき、新しいANLVL値がANMIN[12:0] より低い場合は、ANLVLがANMINに置き換えられます。

  最適な信号の検出が行えるように、雑音推定機能のスロープを調整することができます。このスロープは、ADF SAD設定レジスタ（ADF_SADCFGR）のANSLP[2:0] を通して調整されます。

次の図に、フレーム･サイズに従って許容される値と、SADによって処理されるデータのサンプリング･レートを示します。グレーで示したセルは、SADMOD[1:0] = 0のときの推奨値です。

表395.FRSIZEとサンプリング･レートに対するANSLP値

FRSIZE	Fs = 8kHzの場合のANSLP			Fs = 16kHzの場合のANSLP		
	低速(1)	標準(2)	高速(3)	低速(1)	標準(2)	高速(3)
0（8サンプル）	0	1	2	-	0	1
1（16サンプル）	1	2	3	0	1	2
2（32サンプル）	2	3	4	1	2	3
3（64サンプル）	3	4	5	2	3	4
4（128サンプル）	4	5	6	3	4	5
5（256サンプル）	5	6	7	4	5	6
6（512サンプル）	6	7	-	5	6	7

1. 低速スロープは、負のスロープでは - 8.5dB/s、正のスロープでは + 2.1dB/sとなります。
2. 標準スロープは、負のスロープでは - 17.1dB/s、正のスロープでは + 4.2dB/sとなります。
3. 高速スロープは、負のスロープでは - 34.2dB/s、正のスロープでは + 8.5dB/sとなります。

スロープは、次の式でも計算できます。

$$SLC_{UP} = 20 \times \frac{F_s}{F_{SIZE}} \times \log 10(1 + 2^{(ANSLP - 12)})$$

$$SLC_{DN} = 20 \times \frac{F_s}{F_{SIZE}} \times \log 10(1 - 2^{(ANSLP - 10)})$$

ここで、F_S はSADによって処理されたストリームのサンプリング･レート、F_{SIZE} はFRSIZE[2:0]によって定義されるフレーム･サイズです。

閾値の計算

閾値の計算方法は、SADMOD[1:0] によって決まります。

- SADMOD[1:0] = 0の場合、THRHを求めるには、現在のANLVL値と、SNTHR[3:0] で定義されたゲインを積算します。

  $$THR_H = ANLVL \times 10^{\frac{GdB_{SNTHR}}{20}}$$

  そして、閾値が現在のサウンド･レベル（SDLVL）と比較されます。
- SADMOD[1:0] = 01の場合、THRHを求めるには、現在のANMIN [12:0] と、SNTHR[3:0] によって定義されたゲインを積算します。

  $$THR_H = ANMIN \times 10^{\frac{GdB_{SNTHR}}{20}}$$

  そして、閾値が現在のサウンド･レベル（SDLVL）と比較されます。
- SADMOD[1:0] = 1xの場合、THRHを求めるには、現在のANMIN[12:0] を4倍します。

$$THR_H = ANMIN \times 4$$

この閾値が以下と比較されます。

$$ANLVL \times 10^{\frac{GdB_{SNTHR}}{20}}$$

ヒステリシス･モードを有効化すると、MONITOR状態とDETECT状態の間の疑似遷移を低減することができます。ヒステリシス･モード（HYSTEN = 1）では、次の閾値が使用されます。

- SADがMONITOR状態の場合はTHRH。
- SADがDETECT状態の場合はTHRL。

次の表に、SNTHRに従った閾値を示します。

表396.SNTHRに従った閾値(1)

SNTHR[3:0]	THR_H		THR_L		コメント
0	LVL + 3.5dB	LVL x 1.5	LVL + 3.5dB	LVL x 1.5	ヒステリシスなし
1	LVL + 6.0dB	LVL x 2	LVL + 3.5dB	LVL x 1.5	ヒステリシスが2.5dB
2	LVL + 9.5dB	LVL x 3	LVL + 6.0dB	LVL x 2	ヒステリシスが3.5dB
3	LVL + 12.0dB	LVL x 4	LVL + 9.5dB	LVL x 3	ヒステリシスが2.5dB
4	LVL + 15.6dB	LVL x 6	LVL + 12.0dB	LVL x 4	ヒステリシスが3.5dB
5	LVL + 18.1dB	LVL x 8	LVL + 15.6dB	LVL x 6	ヒステリシスが2.5dB
6	LVL + 21.6dB	LVL x 12	LVL + 18.1dB	LVL x 8	ヒステリシスが3.5dB
7	LVL + 24.1dB	LVL x 16	LVL + 21.6dB	LVL x 12	ヒステリシスが2.5dB
8	LVL + 27.6dB	LVL x 24	LVL + 24.1dB	LVL x 16	ヒステリシスが3.5dB
9	LVL + 30.1dB	LVL x 32	LVL + 27.6dB	LVL x 24	ヒステリシスが2.5dB

1. SADMOD[1:0] = 0の場合はLVLをANLVLに置き換え、その他のSADMOD[1:0] 値の場合はANMINに置き換える必要があります。

ヒステリシス機能が無効化されている場合、SADでは常にTHRHを使用します。

注： **SADMOD[1:0] = 1xの場合、ヒステリシス･モードを使用しないでください。**

トリガ･ロジック

この閾値と比較される信号も、SADMOD[1:0] によって決まります。

選択される信号（SELSIG）が閾値レベルを上回っている場合、トリガ条件に達します。

トリガ条件が満たされた場合、次のことが行われます。

- SADがDETECT状態に切り替わります。
- SADにより、HGOVRを使用してハングオーバー･カウンタが参照されます。
- SADがMONITOR状態からDETECT状態に遷移すると、sddet_evtイベントがアサートされます。
- adf_sad_det信号がハイにセットされます。

トリガ条件が満たされている間、またはハングオーバー･ダウンカウンタが0以外である間、SADはDETECT状態のままとなります。

sddet_evtイベントは、SADがDETECT状態になったかDETECT状態を終了したときに発行されます。このイベントは、サウンドが検出されたとき、またはサウンドが検出されなくなったときに割込みを生成するために使用されます。

- DETCFG = 0の場合、SADがDETECT状態になったときにのみ、アプリケーションでイベントを受信します。
- DETCFG = 1の場合、SADがDETECT状態になったとき、またはDETECT状態を終了したときに、アプリケーションでイベントを受信します。

SADがDETECT状態である間、adf_sad_det信号はハイのままです。

SADは、新しいサウンド･レベル値が利用可能であることを示すフラグ（SDLVLF）も備えています。前回算出したサウンド･レベル（SDLVL[14:0]）はADF SADサウンド･レベル･レジスタ（ADF_SADSDLVR）に格納され、前回算出した環境雑音レベル（ANLVL[14:0]）はADF SAD環境雑音レベル･レジスタ（ADF_SADANLVR）に格納されています。

注： **SADは、AHBクロックがなくても動作可能です。その場合、SADによるSDLVL[14:0] およびANLVL[14:0] の更新は行われません。**

有効な最新のSDLVL[14:0] およびANLVL[14:0] 値を取得するには、SDLVLFフラグがハイの状態でアプリケーションによりADF_SADSDLVRおよびADF_SADANLVRレジスタを読み出す必要があります。これは、以下の方法で行うことができます。

- SDLVLFフラグをポーリングする方法：
  a) SDLVLFに1を書き込んでSDLVLFフラグをクリアします。
  b) ADF_DFLTxISRを読み出して、SDLVLF = 1になるのを待ちます。
  c) ADF_SADSDLVRとADF_SADANLVRを読み出します。
  d) SDLVLFに1を書き込んでクリアします。
  e) 他の値を読み出す必要がある場合は、2番目の手順に戻ります。
- 割込みを生成する方法：
  a) ADF_DFLTxISRを読み出します。
  b) SDLVLF = 1の場合、ADF_SADSDLVRとADF_SADANLVRを読み出し、SDLVLFに1を書き込んでクリアします。
  c) 別のステータス･フラグを処理し、ISRを終了します。

メモリへのサンプルの転送

SADには、DFLT0からシステムメモリへのサンプルの転送を制御するための次のオプションが用意されています。

- DATCAP[1:0] = 1xの場合、DFLT0とSADが有効化されるとすぐに、サンプルがシステム･メモリに転送されます。転送はSADの状態に関係なく行われます。
- DATCAP[1:0] = 01の場合、DFLT0とSADが有効化されていれば、SADでサウンドが検出されたとき（SADがDETECT状態のとき）にサンプルがシステム･メモリに転送されます。
- DATCAP[1:0] = 0の場合、サンプルはメモリに転送されません。このモードが利用できるのは、アプリケーションでサンプルを参照したいだけで、それ以外処理のためにサンプルを必要としているわけではない場合です。

注： **DATCAP[1:0] は、SADEN = 1の場合のみ考慮されます。たとえば、SADの設定がDATCAP[1:0] = 0、SADEN = DFLTEN = 1で、アプリケーションによりここでSADENに0がセットされると、DFLT0によって提供されるサンプルはRXFIFOに転送されます。**

プログラミング上の推奨事項

SAD機能を正しく機能させるには、ADFを次のようにプログラムする必要があります。

1. 適切なカーネル･クロック（adf_ker_ck）をADFに供給します。
2. CKGENを設定し、有効化します。
3. SITFを設定し、有効化します（ただし、マイクロフォンには数ミリ秒の設定時間があります）。
4. DFLT0を設定します。標準の設定は次のとおりです。
   - CIC5のデシメーション比は12、16、または24
   - RSFLTのデシメーション比は4
   - HPFの設定はHPFC = 2または3

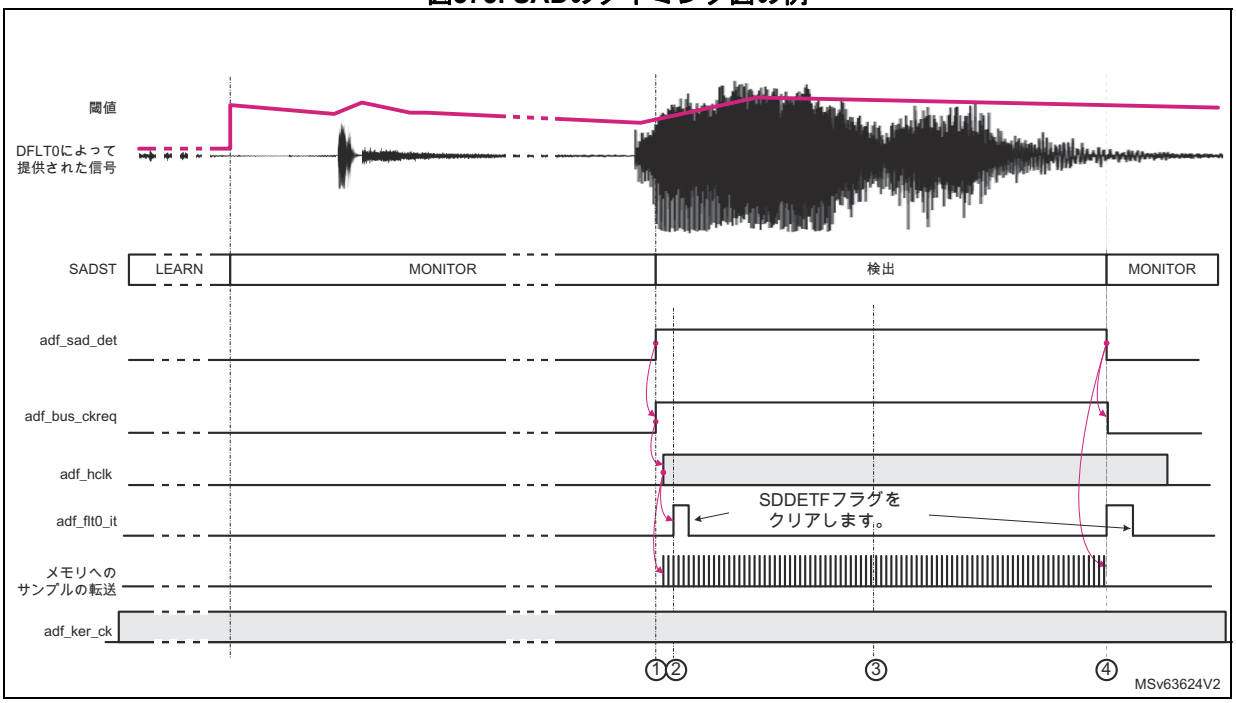
   超低消費電力実装の場合、RSFLTをバイパスすることができます。
5. SADENを0にセットします。
6. SADACTIVE = 0を待ちます。SADENが前に有効化されている場合、このフェーズは2周期のadf_hclkと2周期のadf_proc_ckとすることができます。
7. 次のようにSADを設定します。
   - DATCAP[1:0] に0をセット：アプリケーションによりサンプルをシステム･メモリに格納しない場合。
   - DATCAP[1:0] に01をセット：SADでサウンドが検出される場合のみ、アプリケーションによりサンプルをシステム･メモリに格納する場合。
   - DATCAP[1:0] に11をセット：アプリケーションにより継続的にサンプルをシステム･メモリに格納する場合。
8. DMAを設定します（オプション）。
9. SADを有効にします。
10. DFLT0を有効化します。

図374に、DATCAP[1:0] = 01の設定でSADが機能している場合のタイミングの概略図を示します。

カーネル･クロック（adf_ker_ck）により、SADはDFLT0によって提供されたオーディオ信号を継続的に監視します。閾値も、環境雑音レベルの推定に従って、継続的に更新されます。

- プログラムされた閾値（1）よりも高いサウンドがSADで検出されると、ADFによりバス･クロックがリクエストされます（adf_bus_ckreqがアサートされます）。
- バス･クロックが利用できる場合（図374の2を参照）、次のようになります。
  - メモリへのデータ転送がトリガされます。
  - イベント割込み（adf_evt_it）を生成できます。
- この例では、アプリケーションのウェイクアップのためにイベント割込み（adf_evt_it）が使用されます。割込みラインを解放するには、SDDETFに1を書き込んでクリアします。
- SADがDETECT状態である間、アプリケーションは十分な数のサンプルとコールが得られるのを待ちます。たとえば、キーワード識別アルゴリズムなどです（図374の3を参照）。
- 次の図に示すケースでは、SADの状態（AADST）はキーワードが識別される前にMONITORに戻ります。DETCFGに1がセットされている場合、SADがMONITOR状態に戻ったときに、イベントから信号送信されます。SADによりメモリへのサンプルの転送が停止され、アプリケーションでは次の検出のために受信バッファをクリーンアップすることができます（図374の4を参照）。

図375. SADのタイミング図の例



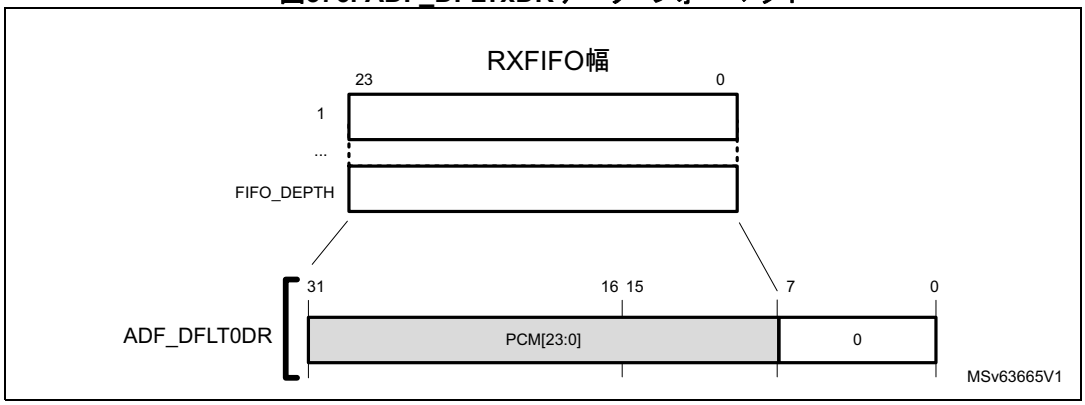
40.4.11 メモリへのデータ転送

データフォーマット

DFLT0 によって処理されるサンプルは、RXFIFOに格納されます。アプリケーションでは、ADFデジタル･フィルタ･データ･レジスタ0（ADF_DFLT0DR）を通して、これらのFIFOに格納されたサンプルを読み出すことができます。このレジスタ内のサンプルは符号付きで、左詰めになっています。ビット31は常に符号を表します。

ADFは24bitの左詰めのデータを提供します。ADF_DFLT0DR への16bitのアクセスを実行すると、16の最上位ビットが取得できます。ADF_DFLT0DR への32bitのアクセスを実行すると、24bitのデータ･サイズが得られます。

図376. ADF_DFLTxDRデータ･フォーマット



データの再同期化

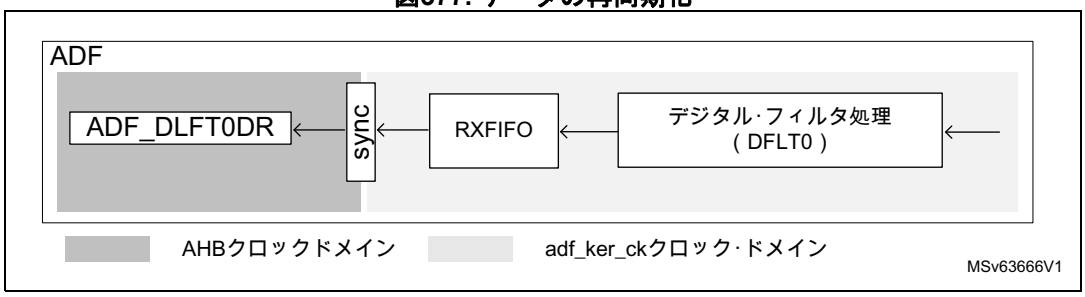
RXFIFO に格納されたサンプルは、DMAリクエストまたは割込み信号を使用して、メモリに転送することができます。

注： **RXFIFO は adf_ker_ck clockクロック･ドメインに配置されますが、ADF_DFLT0DR はadf_hclk（AHB）クロック･ドメインに配置されます。**

AHBクロックが利用可能な状態で、ADF_DFLT0DR が空でサンプルがRXFIFOにある場合、このサンプルは ADF_DFLT0DR に転送されます。

RXFIFOから ADF_DFLT0DR へのサンプルの転送には、2周期のAHBクロック（adf_hclk）と2周期のadf_ker_ckクロックが必要です。RXFIFOから ADF_DFLT0DR への新しいサンプルの転送が完了していない状態でアプリケーションにより ADF_DFLT0DR の読出し操作が行われると、ADFは自動的にウェイトステートを挿入します。

図377. データの再同期化



データ転送

RXFIFOの内容をメモリに転送するには、DMAチャネルまたは割込みサービスを使用します。

ADFではシングルDMA転送とバーストDMA転送の両方がサポートされていますが、アプリケーションで次の点に注意する必要があります。

- RXFIFOに少なくともバーストサイズと同じ量のサンプルが格納されている必要があります。
- 前のセクションで説明したデータの再同期化により､バーストモードを効率的に減らすことができます。

注： **バースト･モードはすべての製品で利用できるわけではありません（製品でサポートされているかどうかは、DMAのセクションを参照してください）。**

また、アプリケションではRXFIFO閾値（FTHビット）を選択してデータ転送をトリガすることができます。データ転送は、RXFIFOが空でなくなった直後に、またはRXFIFOがハーフフル（半分の深さのサンプル）になったときにトリガできます。

DMA転送では、RXFIFOの1つが閾値レベルに達するとすぐに、データ転送を要求するために、DMAリクエストがアサートされます。RXFIFOが空になるまで、DMAリクエストが連続で行われます。

RXFIFOのDMAモードは、ADF_DFLT0DRのDMAENビットを通して有効化されます。

割込み信号については、次のケースを考慮する必要があります。

- FTH = 0の場合、ADF_DFLT0DRでデータが利用可能になるとすぐに、FTHFがセットされ、割込みの生成が可能になります。FTHFはADF_DFLT0DRが読み出されるとすぐに解放されます。
- FTH = 1の場合、RXFIFOが閾値レベルに達し、データがADF_DFLT0DRで利用可能になるとすぐに、FTHFがセットされ、割込みの生成が可能になります。FTHFは、データが1つ読み出されるとすぐに解放されます。閾値の条件が再び満たされると、FTHFが再びセットされます。このモードでは、割込みが発生するたびに、FIFO_SIZE / 2のデータが読み出されます。

RXFIFOのオーバーラン

RXFIFOのオーバーラン状態は、RXFIFOがフルで、DFLT0 からの新しいサンプルを書き込む必要がある場合に検出されます。

この場合、DOVRFがセットされ、新しいサンプルは破棄されます。RXFIFOで少なくとも1つの場所が利用できる場合、新しく受信するサンプルがRXFIFOに書き込まれます。

図377に、RXFIFOの深さが4ワード、FTHに1がセットされ、RXFIFOがハーフフルのときにFTHFが１ になる例を示します。

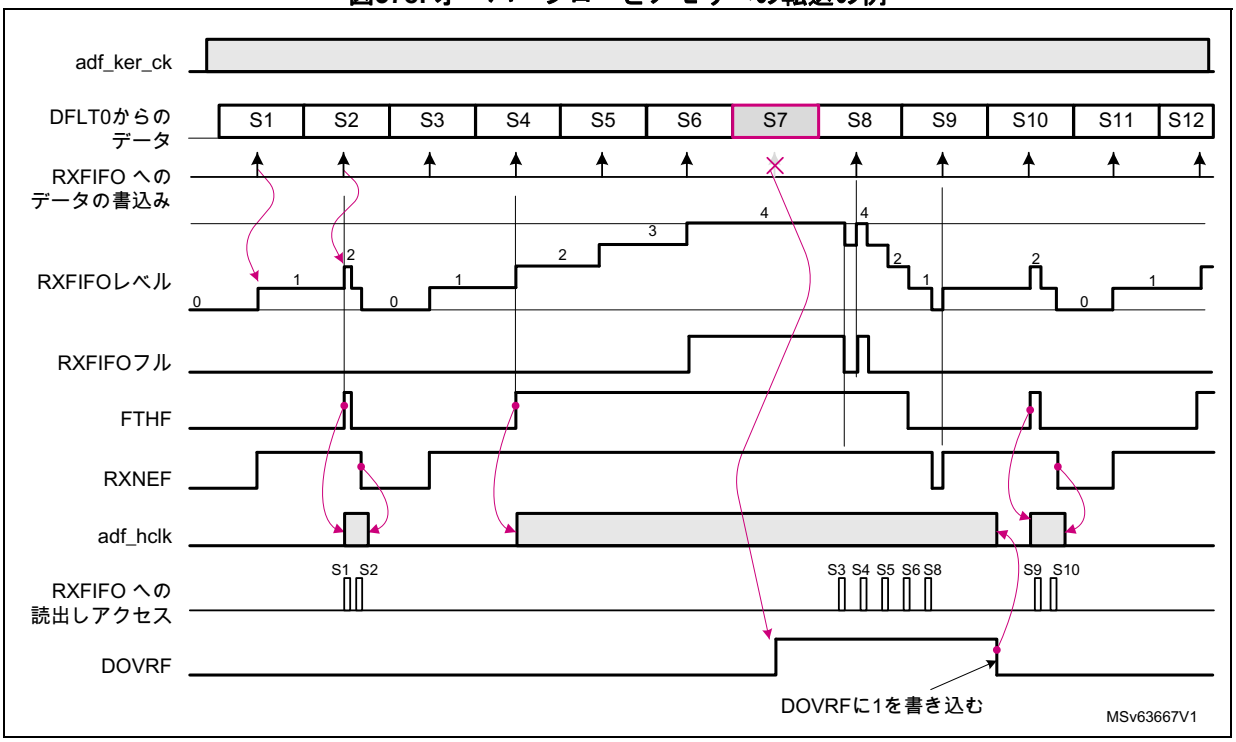
S7サンプルはオーバーランにより失われます。S7をRXFIFOに書き込む必要があるときにRXFIFOはフルです。転送S7書込み操作は行われません。書込み操作が予期された瞬間に、DOVRFに1がセットされます。アプリケーションによってクリアされない限り､オーバーフロー･イベントは1のままです。

この例では、オーバーラン状態が検出された場合に割込みが生成されるように、DOVRIEに1がセットされます。

S7サンプルの後に、アプリケーションによりRXFIFOからのデータの読出しが行われ、ADFではS8サンプル以降を書き込むことができます。その後、アプリケーションでDOVRがクリアされ、新しいオーバーラン状況の検出が可能になります。

adf_hclkラインで､グレーのボックスはADFによりAHBクロックがリクエストされたことを示します。次の図に、ADFによってリクエストされたときのみ利用可能なAHBクロックを示します。実際には、AHBクロックはADFによってリクエストされなくても存在することがあります。

図378. オーバーフローとメモリへの転送の例



40.4.12 自律モード

ADFは、AHBバス･クロックが利用できない場合でも動作可能です（STOPモード）。ADFでは、レジスタ･インタフェース用にのみAHBクロックを使用します。処理部品はすべて、カーネル･クロックによりクロック供給されます。

STOPモードでは、次の条件が満たされた場合に、ADFはカーネル･クロックを受信します。

- ADF自律モードがRCCで有効化されている。
- 選択したカーネル･クロック･ソースが、STOPモードで利用可能なオシレータから取得される。

STOPモードで、次の条件が満たされた場合にADFはAHBクロックを受信します。

- ADF自律モードがRCCで有効化されている。
- ADFが次の状況で、AHBクロックをリクエストする。
  - ADFがDMAを通してメモリにデータを転送する必要がある場合

    製品がSTOPモードのままで、DMAを使用してSRAMにデータが直接転送されます。AHBクロック･リクエストは、DMA転送が完了するまで維持されます。
  - ADFで割込みを生成する必要がある場合

    アプリケーションからのアクションが必要なため、一般的に割込みにより、デバイスはSTOPモードからウェイクアップします。AHBクロックが利用可能になると、割込みが生成されます。AHBクロック･リクエストは、有効化された割込みフラグがアクティブである間、保持されます。

40.4.13 レジスタの保護

ADFには、無効な状況を回避するためのハードウェア保護が組み込まれています。次の表に、書込み保護されているフィールドとされていないフィールドのリストを示します。

表397.レジスタ保護のまとめ

レジスタ	書込み保護されていないフィールド	書込み保護されているフィールド	書込み保護の条件
ADFグローバル制御レジスタ（ADF_GCR）	TRGO	-	DFLTACTIVE0 = 1
ADFクロック･ジェネレータ制御レジスタ（ADF_CKGCR）	CKGDEN CCK0EN CCK1EN	PROCDIV[6:0]、CCKDIV[3:0]、CKGMOD、TRGSRC[3:0]、TRGSENS、CCK[1:0]DIR	CKGACTIVE = 1
ADFシリアル･インタフェース制御レジスタ0（ADF_SITF0CR）	SITFEN	STH[4:0]、SITFMOD[1:0]、SCKSRC[1:0]	SITFACTIVEx = 1
ADFビットストリーム･マトリックス制御レジスタ0（ADF_BSMX0CR）	-	BSSEL[4:0]	DFLTACTIVEx = 1
ADFデジタル･フィルタ制御レジスタ0（ADF_DFLT0CR）	DFLTEN	NBDIS[7:0]、TRGSRC[3:0]、TRGSENS、FTH、DMAEN、SNPSFMT、ACQMOD[2:0]	DFLTACTIVEx = 1
ADFデジタル･フィルタ設定レジスタ0（ADF_DFLT0CICR）	SCALE[5:0]	MCICD[8:0]、CICMOD[2:0]、DATSRC[1:0]	
ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）	-	すべてのフィールド	
ADF遅延制御レジスタ0（ADF_DLY0CR）	-	SKPDLY[6:0]	SKPBF = 1
ADF DFLT0割込み有効化レジスタ（ADF_DFLT0IER）	すべてのフィールド	-	-

表397.レジスタ保護のまとめ （続き）

レジスタ	書込み保護されていないフィールド	書込み保護されているフィールド	書込み保護の条件
ADF DFLT0割込みステータス･レジスタ0（ADF_DFLT0ISR）	すべてのフィールド	-	-
ADF SAD制御レジスタ（ADF_SADCR）	SADEN	FRSIZE[2:0]、DETCFG、SADMOD[1:0]、HYSTEN、DATCAP[1:0]	SADACTIVE = 1
ADF SAD設定レジスタ（ADF_SADCFGR）	-	ANMIN[12:0]、SNTHR[3:0]、HGOVR[3:0]、LFRNB[2:0]、ANSLP[2:0]	SADACTIVE = 1

ADF処理はすべて、adf_proc_ckクロック･ドメインで行われます。そのため、AHBクロック･ドメインとadf_proc_ckクロック･ドメインの間の再同期化のために、ADFサブブロックの有効化と無効化には多少時間がかかることがあります。XXXACTIVEフラグを使用することにより、アプリケーションでは2つのクロック･ドメイン間の同期が完了したかどうかを確認できます。

書込み保護されたビットフィールドを変更するには、アプリケーションは次のシーケンスに従います。

1. サブブロックの有効化ビットに0をセットします。
2. 対応するフラグXXXACTIVEが0になるのを待ちます。
3. 必要なフィールドを変更します。
4. サブブロックの有効化ビットに1をセットします。

詳細については、各サブブロックの説明を参照してください。

40.5 ADF低電力モード

表398.低消費電力モードがADFに与える影響

モード	説明
SLEEP	影響はありません。ADF割込みによって、デバイスはSLEEPモードから復帰します。
STOP(1)	ADFレジスタの内容は保たれます。 RCCで自律モードが有効化されていて、ADFがSTOPモードで利用可能な内部オシレータによってクロック供給されている場合、ADFはアクティブなままとなります。DMAリクエストは機能し、これらのモードでの割込みにより、デバイスはSTOPモードを終了します。
STANDBY	ADFはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. Refer to セクション 40.8.9：ADF DFLT0割込み有効化レジスタ（ADF_DFLT0IER）for details about Stop modes supported by the ADF.

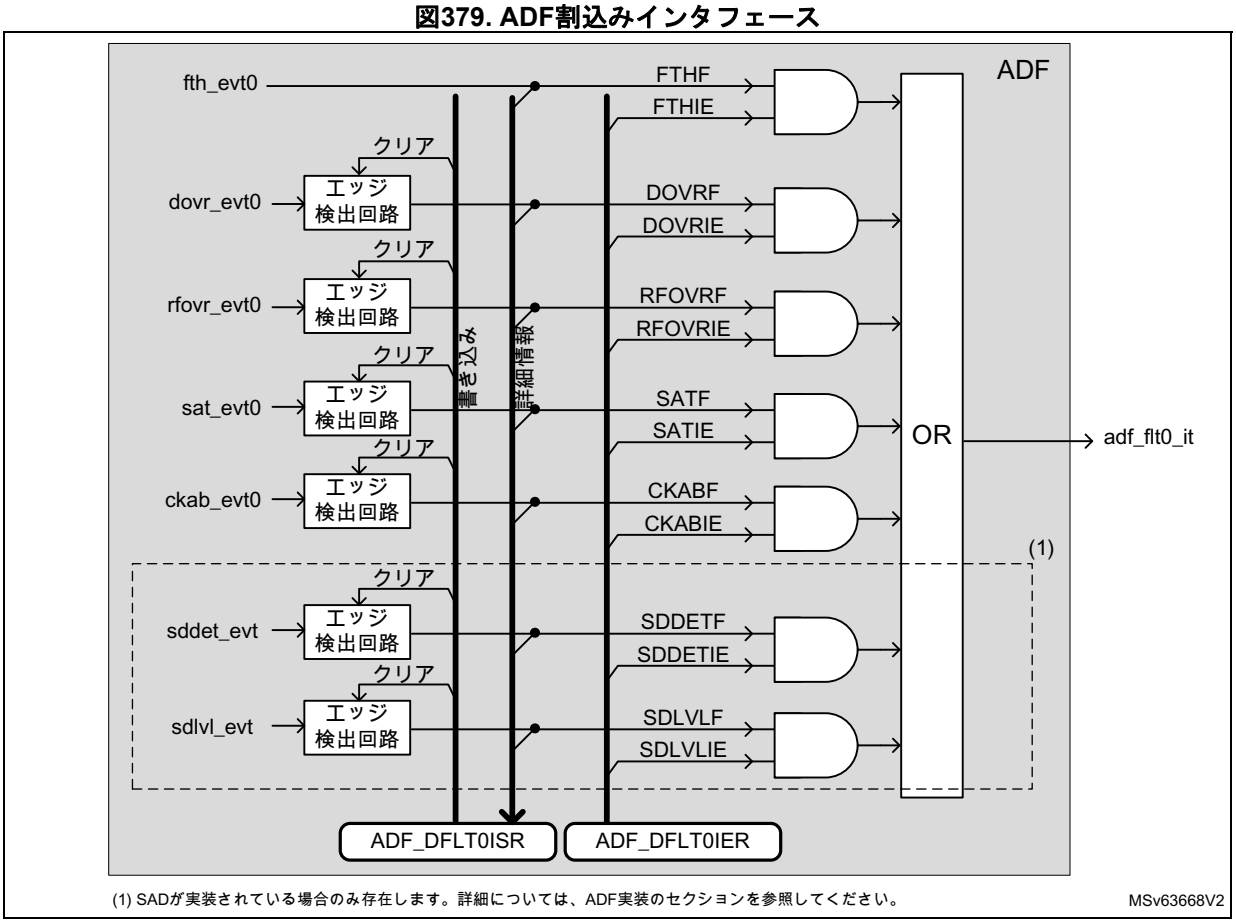
40.6 ADF割込み

CPUのパフォーマンスを向上させるため、ADFには複数のイベントに対応する割込みライン（adf_flt0_it）が用意されています。

注： **ステータス･フラグは、対応する割込み有効化フラグが有効化されていない場合でも利用できます。**

割込みインタフェースは、ADF DFLT0割込み有効化レジスタ（ADF_DFLT0IER）およびADF DFLT0割込みステータス･レジスタ0 （ADF_DFLT0ISR）を通して制御されます。

図379. ADF割込みインタフェース



次の表に、どのイベントがどの割込みに影響するかと、イベント／割込みのクリアと有効化の方法を示します。

表399.ADF割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イベント／割込みのクリア方法	SLEEPモードの終了	STOPモードの終了(1)	STANDBYモードの終了
ADF_FLT0(2)	RXFIFO閾値に達した	FTHF	RXFIFOレベルが閾値を下回るまでADF_DFLT0DRを読み出します。	可能	可能	不可
	RXFIFOオーバーランです。	DOVRF	DOVRFに1を書き込みます。			
	RSFLTオーバーランです。	RFOVRF	RFOVRFに1を書き込みます。			
	飽和の検出	SATF	SATFに1を書き込みます。			
	チャネル･クロックなし検出	CKABF	CKABFに1を書き込みます。			
	SAD：サウンド検出済み	SDDETF	SDDETFに1を書き込みます。			
	SAD：サウンドレベル値が利用可能	SDLVLF	SDLVLFに1を書き込みます。			

1. 詳細については、セクション40.3を参照してください。
2. ADF_FLT0ベクタはadf_flt0_it信号のアサートに相当します。

40.7 ADFアプリケーション情報

40.7.1 オーディオ･キャプチャ用のADF設定の例

表398に、16kHzおよび48kHzの出力データ･レートに焦点を当てた、デジタル･マイクロフォンのADF設定の例を示します。

設定 #1と #2は非常に消費電力の少ないユースケース用で、S/N比は低くなっています。ユーザは、選択したデジタルマイクロフォンが512kHzで正しく動作することも確認する必要があります。これらの設定はサウンド検出に使用できます。RSFLTは、カーネル･クロック（adf_ker_ck）の周波数をできるだけ抑えるために使用するものではありません。

設定 #3、#4、#9、#10、#11は、理想的なマイクロフォンのモデルと997Hzの正弦波信号を使用した、約115dBでのS/N比を示しています。RSFLTを使用して、高度なインバンド･リップルの制御と高度な画像除去を行うことができます。

設定 #7、#8、#10は、理想的なマイクロフォンのモデルと997Hzの正弦波信号を使用した、約120dBでのS/N比を示しています。

表400.マイクロ･フォンキャプチャ用のADF設定の例

設定	adf_ker_ck（MHz）	PROCDIV + 1	CCKDIV + 1	CICの次数 (1)	MCICD + 1	SCALE	RSFLTBYP	RSFLTD	HPFBYP	-	adf_proc_ck（MHz）	総デシメーション比	F_{RS}（kHz）	F_{ADF_CCKx}（MHz）	F_{PCM}（kHz）
1	1.024	1	2	4	64	0x2D（-8.5dB）	1	x	x	=>	1.024	64	-	0.512	8
2		1	2	5	32	0x2B（-14.5dB）						32	-		16
3		1	2	5	16	0x01（+3.5dB）	0	0				64	32		8
4	2.048	1	2	5	16	0x01（+3.5dB）	0	0			2.048	64	64	1.024	16
5	3.072	1	6	5	8	0x0B（+33.6dB）	0	0			3.072	32	32	0.512	8
6		2	2	5	12	0x06（+18.1dB）						48	64	0.768	16
7		1	2	5	24	0x2C（-12dB）						96		1.536	
8	4.096	1	2	5	32	0x27（-26.6dB）	0	0			4.096	128	64	2.048	16
9	6.144	3	2	5	16	0x02（+6.0dB）	0	0			2.048	64	64	1.024	16
10		2	2	5	24	0x2C（-12dB）					3.072	96		1.536	
11		1	2	5	16	0x01（+3.5dB）					6.144	64	192	3.072	48
12	7.680	1	2	5	20	0x2E（-6.0dB）	0	0			7.680	80	192	3.840	48

1. CICの次数が4の場合、CICMOD = 100。CICの次数が5の場合、CICMOD = 101。

40.7.2 プログラミングの例

例1

この例では、カーネル･クロックが1.536MHzと仮定して、デシメーションを48とし、CIC4のみを使用して、デジタル･マイクロフォンから受信する信号をキャプチャするためにADFをプログラミングする方法について説明します。通常、この設定は、SADを使用してサウンドを検出するために使用できます。

表401.プログラミング･シーケンス（CIC4）

操作	コメント
RCCを通してカーネル･クロック周波数を調整	RCオシレータから受信する1.536MHzのカーネル･クロック（adf_ker_ck）が供給されるようにRCCがプログラムされていることが前提となります。
RCCを通して適切なADFカーネル･クロック･ソースを選択	製品のRCCを参照してください。
RCCを通してADFクロックを有効化	製品のRCCを参照してください。
RCCを通してADFをリセット	製品のRCCを参照してください。
AFMUXプログラミング	ADF_SD0およびADF_CCK0機能を選択するようAFMUXをプログラムします。
ADF処理クロックの有効化： ADF_CKGCR = 0x0001 0023	PROCDIV = 0（バイパス）：adf_proc_ck周波数は1.536MHz。 CCKDIV = 1（2で分周）：ADF_CCK0クロック周波数は768kHzです。 マイクロフォンが低消費電力モードを終了できるように、ADF_CCK0ピンは出力にセットされ、クロックを生成します。

表401.プログラミング･シーケンス（CIC4）（続き）

操作	コメント
シリアル･インタフェース設定： ADF_SITF0CR = 0x0000 1F01	ADF_CCK0をシリアル･クロックとして選択するため、SCKSRC = 0。 LF_MASTER SPIモードを選択するため、SIFTMOD = 0。 クロックなし機能はこのモードでは動作しません。 シリアル･インタフェースは有効です。
ビットストリーム･マトリックス設定： ADF_BSMX0CR = 0x0000 0000	DFLT0フィルタはSITF0のビットストリームを取得（立ち上がりエッジでサンプリング）
フィルタ設定（CIC）： ADF_DFLT0CICR = 0x0040 2F40	SCALE = 0x04（12dB）：RSFLTは有効ではありません。ただし、ゲインは8.5dBで、SADの信号精度を高めるため、推奨より高くなっています。最大測定範囲よりもはるかに低い信号の検出が求められるだけであるため、このケースでは飽和は問題となりません。 MCICD = 0x2F（48でデシメーション） $Sinc^4$ を選択するため、CICMOD = 4 BSMXから受信するデータを選択するため、DATSCR = 0
フィルタ設定（RSFLTおよびHPF）： ADF_DFLT0RSFR = 0x0000 0301	HPFC = 3：16kHzの遮断周波数 * 0.0095 = 152Hz HPFBYP = 0：HPFはバイパスされていません。 RSFLTBYP = 1：RSFLTはバイパスされています。
遅延の微調整： ADF_DLY0CR = 0x0000 0000	この例では使用されません。
割込みイベントの有効化： ADF_DFLT0IER = 0x0000 1000	アプリケーションで扱う割込みイベントを有効化します。 この例では、サウンドが検出された場合に割込みが生成されるように、 SDDETIEに1がセットされます。
SADの設定： ADF_SADCR = 0x0000 1400	SADMOD = 1：サウンド･レベル値に基づいてトリガ FRSIZE = 4：サウンド･レベルは128のサンプルで算出 HYSTEN = 0：ヒステリシスなし DETCFG = 0：SADがDETECT状態になったときに割込みが生成されます。 DATCAP = 0：サンプルはメモリに転送されません。 SADEN = 0：SADはまだ有効になっていません。
SADの設定： ADF_SADCFGR = 0x0025 0001	ANMIN = 0x25：サウンドの閾値37 * 10^SNTHR/20 = 74 LSB(1) HGOVR = 0：4つのフレーム（4 x 128 = 512サンプル） LFRNB = 0：学習フェーズを2フレームにセット。このモードでは使用しません。 ANSLP = 0：このモードでは使用しません。 SNTHR = 1：閾値を6dBにセット（ANMINを参照）
ステータス･フラグのクリア： ADF_DFLT0ISR = 0x0000 0FFF	ADFの実行前にステータス･フラグをすべてクリアします。
SADの有効化： ADF_SADCR = 0x0000 1401 SADACTIVEフラグが1になるのを待つ	メモリへのデータ転送がSADによってブロックされるように、最初にSADを有効化します。
フィルタの有効化： ADF_DFLT0CR = 0x0000 0001	NBDIS = 0：サンプルの破棄なし ACQMOD = 0：非同期連続取得モード DMAEN = 0：DMAインタフェースは使用されていません。 DFLTEN = 1：フィルタは有効です。

1. SNTHRおよびANMINの値は、「SADMOD = 01 での閾値のプログラミング（63dBSPLを上回るサウンドの検出）」と同じアプローチで算出されます。

例2

この例では、CIC5とRSFLTを使用し、総デシメーションを64として、デジタル･マイクロフォンから受信する信号をキャプチャするためにADFをプログラミングする方法について説明します。

表402.プログラミング･シーケンス（CIC5）

操作	コメント
RCCを通してカーネル･クロック周波数を調整	6.144MHzのカーネルクロック（adf_ker_ck）が供給されるようにRCCがプログラムされていることが前提となります。
RCCを通して適切なADFカーネル･クロック･ソースを選択	製品のRCCを参照してください。
RCCを通してADFクロックを有効化	製品のRCCを参照してください。
RCCを通してADFをリセット	製品のRCCを参照してください。
AFMUXプログラミング	ADF_SD0およびADF_CCK0機能を選択するようAFMUXをプログラムします。
ADF処理クロックの有効化： ADF_CKGCR = 0x0201 0023	PROCDIV = 2（3で分周）：adf_proc_ck周波数は6.144MHz。 CCKDIV = 1（2で分周）：ADF_CCK0クロック周波数は1.024MHzです。 マイクロフォンが低消費電力モードを終了できるように、ADF_CCK0ピンは出力にセットされ、クロックを生成します。
シリアル･インタフェース設定： ADF_SITF0CR = 0x0000 1F01	ADF_CCK0をシリアル･クロックとして選択するため、SCKSRC = 0。 LF_MASTER SIPIモードを選択するため、SIFTMOD = 0。 クロックなし機能はこのモードでは動作しません。 シリアル･インタフェースは有効です。
ビットストリーム･マトリックス設定： ADF_BSMX0CR = 0x0000 0000	DFLT0フィルタはSITF0のビットストリームを取得（立ち上がりエッジでサンプリング）
デジタル･フィルタの制御： ADF_DFLT0CR = 0x0000 0006	NBDIS = 0：サンプルの破棄なし ACQMOD = 0：非同期連続取得モード DMAEN = 1：DMAインタフェースは使用されていません。 DFLTEN = 0：フィルタ無効 FTH = 1：RXFIFOハーフフル
フィルタ設定（CIC）： ADF_DFLT0CICR = 0x0010 0F50	SCALE = 0x01（+3.5dB）：出力をRSFLTにより22bitに制限。 MCICD = 0xF（16でデシメーション） $Sinc^5$ を選択するため、CICMOD = 5 BSMXから受信するデータを選択するため、DATSCR = 0
フィルタ設定（RSFLTおよびHPF）： ADF_DFLT0RSFR = 0x0000 0200	HPFC = 2：16kHzの遮断周波数 * 0.0025 = 40Hz HPFBYP = 0：HPFはバイパスされていません。 RSFLTBYP = 0：RSFLTはバイパスされていません。 RSFLTD = 0：4でデシメーション
遅延の微調整： ADF_DLY0CR = 0x0000 0000	この例では使用されません。
割込みイベントの有効化： ADF_DFLT0IER = x	アプリケーションで扱う割込みイベントを有効化します。
ステータス･フラグのクリア： ADF_DFLT0ISR = 0x0000 0FFF	ADFの実行前にステータス･フラグをすべてクリアします。

表402.プログラミング·シーケンス（CIC5）（続き）

操作	コメント
DMAのプログラム DMAを有効にします。	DMAリクエストが生成されるたびにADF_DLT0DRレジスタ内部でデータを読み出すには、DMAをプログラムする必要があります。
取得の開始： ADF_DFLT0CR = 0x0000 0007	ADFでデータのフィルタリングを開始します。

40.7.3 接続の例

図379に、ADFを外部センサに接続する簡単な例を示します。

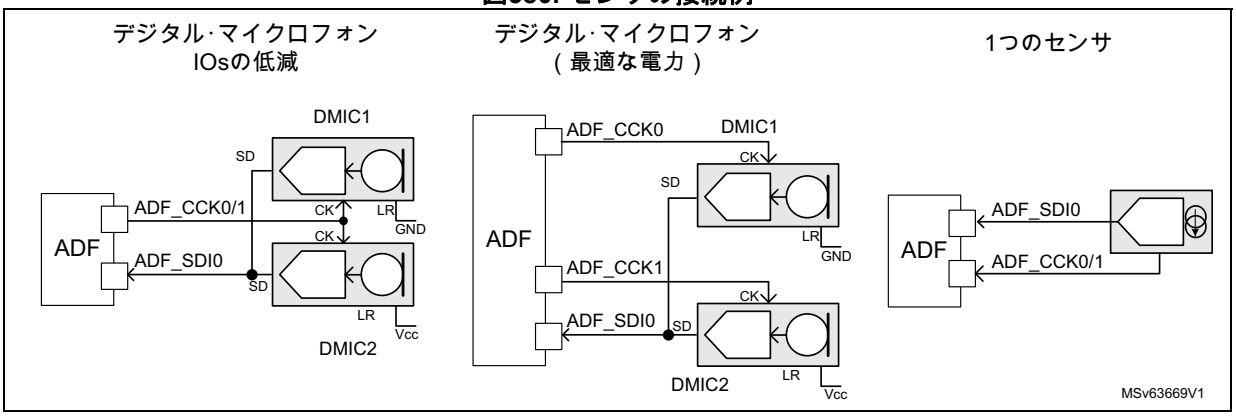
- 図の左側：ADFに接続された2本のデジタル·マイクロフォン

  この接続では、接続の数が最適化され、DMIC1とDMIC2が同じデータ·リンクと同じクロック·ラインを共有しています。アプリケーションは、BSMXにより、デジタルフィルタをDMIC1またはDMIC2に接続することができます。この設定では、2本のマイクロフォンが同じクロックを共有しているため、一方を使用するときに、もう一方も有効化されます。
- 図の中央：ADFに接続された2本のデジタル·マイクロフォン

  この設定では、DMIC1とDMIC2は同じデータ·ラインを共有していますが、専用のクロック·ラインを使用しています。アプリケーションは、BSMXにより、フィルタをDMIC1またはDMIC2に接続することができます。アプリケーションで一方のマイクロフォンを使用するとき、クロック·ラインを強制的に0（CCKyEN = 0、CCKyDIR = 1）にすることにより、もう一方を低消費電力モードのままにすることができます。
- 図の右側：ADFに接続された1つのセンサ

  CCK0ピンとCCK1ピンを入力に設定して、クロックを供給するセンサを接続することもできます。

図380. センサの接続例



40.7.4 グローバル周波数応答

図380に、1.024Mhzで動作するデジタル･マイクロフォンを使用した16kHzのオーディオ信号のグローバル周波数応答を示します。フィルタ設定は、次のとおりです。

- CICの次数4または5、デシメーション比は16
- RSFLT有効、デシメーション比は4
- HPF有効、遮断周波数は40Hz

次の図に、CIC4とCIC5を使用した理論上の周波数応答を示します。

図381. グローバル周波数応答

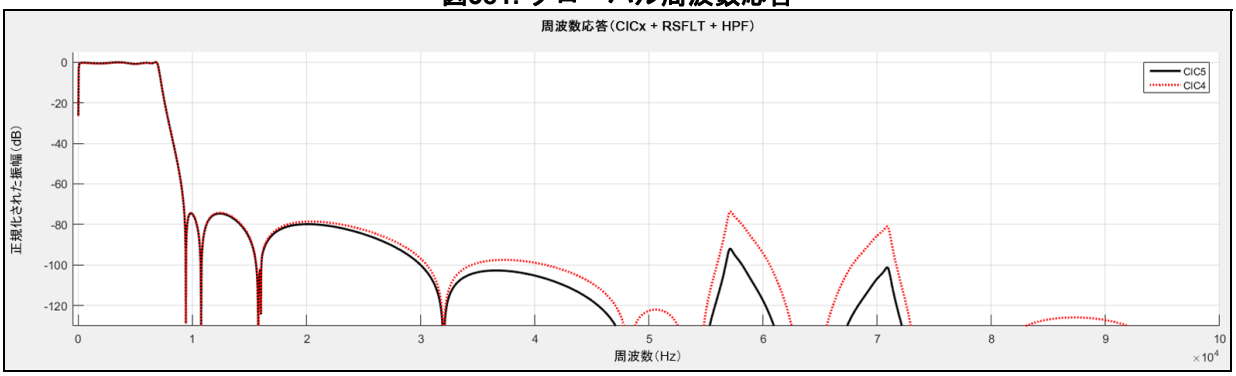


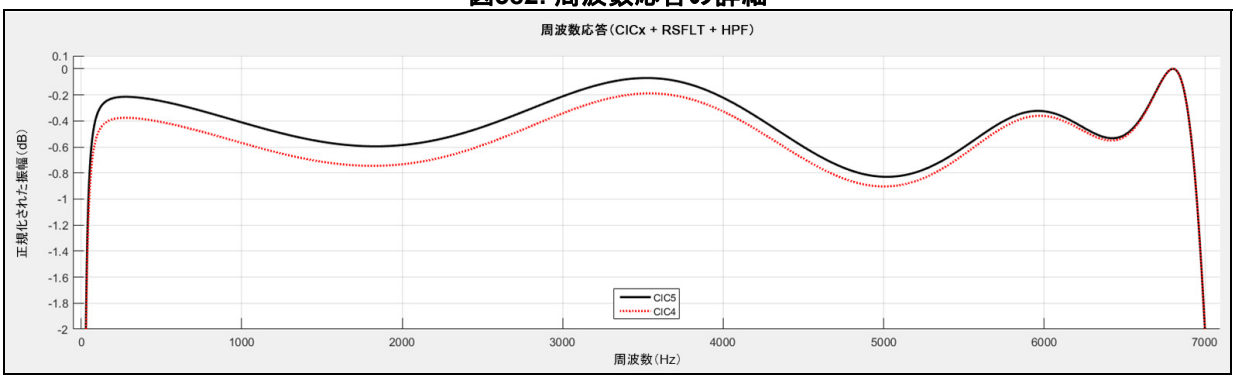
図381に、1.024Mhzで動作するデジタル･マイクロフォンを使用した16kHzのオーディオ信号のインバンド･リップルを示します。フィルタ設定は、次のとおりです。

- CICの次数4または5、デシメーション比は16
- RSFLT有効、デシメーション比は4
- HPF有効、遮断周波数は20Hz

発生するインバンド･リップルは、CIC5では ± 0.41dB、CIC4では ± 0.45dBとなります。

-3dBの遮断周波数は7061Hzです。

図382. 周波数応答の詳細



40.7.5 総ADFゲイン

このセクションでは、フィルタ設定に従って、ADFによって提供される信号のレベルを計算する方法を説明します。

信号のレベルは、dBFS単位（デシベルの最大測定範囲）で表現できます。0dBFSのレベルは、許容される最大のデジタル･レベルに割り当てられます。たとえば、最大レベルの50 % に達した信号は、- 6dBFSのレベル（最大測定範囲を6dB下回る）となります。

たとえば、最終データ幅が24bitのADFの場合、振幅が2 * 10^6 LSBの信号のレベルは次のようになります。

$$20 \times \log_{10}\left(\frac{2 \times 10^6}{2^{(24-1)}}\right) = -12.45 \text{ dBFS}$$

また、LSB単位で表現される振幅（Amp）を持つ信号のデータ･サイズは、次の式で算出できます。

$$DS = \left(\frac{\ln(Amp)}{\ln(2)} + 1\right) \text{ビット}$$

負の値の場合、1bit追加する必要があります。

そのため、振幅が2 * 10^6 LSBの信号のデータ･サイズは21.9bitです。

CICゲイン

CICゲイン（G_{CIC} およびGdB_{CIC}）は、データ･サイズをビット単位として（DS_{CIC}）、次の式から推定することができます。

$$DS_{CIC} = (N \times \log_2(D1)) + DSin$$

ここで、NはCICの次数（CICMOD[2:0] により選択）、D1はデシメーション比（MCICD[8:0] により指定）を表します。

DSinはCIC入力での信号のデータ･サイズ（ビット単位）を表します。

警告： DS_{CIC} はCICフィルタにとって非常に重要です。正しく動作するには、DS_{CIC} が26bitを超えないようにする必要があります。

CICのゲインG_{CIC} は次の式で算出できます。

$$G_{CIC} = (D1)^N$$

デシベル単位では次のようになります。

$$GdB_{CIC} = 20 \times \log_{10}((D1)^N)$$

SCALE出力でのデータ･サイズ

SCALE出力でのピークツーピークの信号振幅が22bitより大きくなるとRSFLTが飽和しはじめるため、SCALE出力でのデータ･サイズ（CICゲインを含めて）は重要な情報です。

RSFLTがバイパスされる場合、24bitのピークツーピークの信号振幅が受容されます。

SCALE出力での信号振幅は、次のようになります。

$$Asout_{SCALE} = D1^{N} \times 10^{\frac{GdB_{SCALE}}{20}} \times Asin_{DFLT}$$

GdB_{SCALE} は、SCALE[5:0] で選択されたdB単位のゲインを表します。

$Asout_{SCALE}$ はSCALE出力での信号振幅（LSB単位）、$Asin_{DFLT}$ はCIC入力での信号振幅（LSB単位）です。

$$DS_{SCALE} = \frac{\ln(Asout_{SCALE})}{\ln(2)} + 1$$

SCALE出力でのデータ･サイズ（DS_{SCALE}）はビット単位で表わされます。

RSFLTゲイン

有益な帯域幅でのRSFLTゲインは通常は9.5dBですが、リップルのために、約 ± 0.41dBのマージンを考慮する必要があります。

$$G_{RSFLT} = 10^{\frac{9.5\ dB}{20}} = 2.98 \text{ 標準}$$

注： **HPFフィルタのゲインは0dBです。**

SADゲイン

SADでは信号で16 MSBのみを使用しているため、SADの観点からは、24bitから16bitへの切詰めは減衰と考えられます。

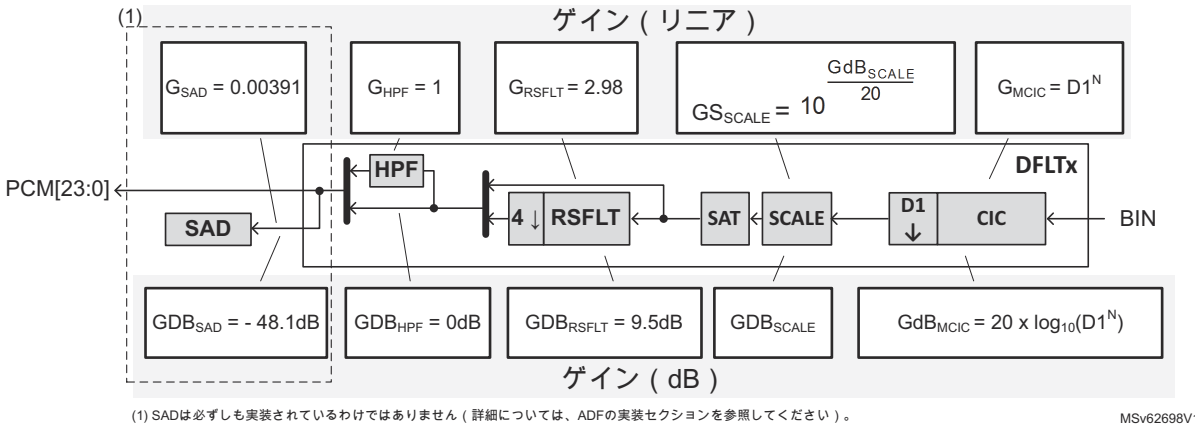
$GdB_{SAD} = -48.1dB$

および

$G_{SAD} = 0.003906$

次の図にフィルタ･パスの概略を示し、重要なコンポーネントについては、ビットの増加とゲインの式を示します。

図383. DFLTの概略とゲイン情報



次の表に、フィルタ設定ごとの最終的なデータ･サイズをまとめます。

表403.出力信号レベル

フィルタ設定	最終信号振幅（LSB）
CIC + RSFLT + HPF + SAD	RXFIFOに提供されるサンプル： $Asout_{RXFIFO} = D1^N \times 10^{\frac{GdB_{SCALE}}{20}} \times 10^{\frac{9.5}{20}} \times Asin_{DFLT}$ SADに提供されるサンプル： $Asout_{SAD} = D1^N \times 10^{\frac{GdB_{SCALE}}{20}} \times 10^{\frac{9.5}{20}} \times 0.003906 \times Asin_{DFLT}$ DS_{SCALE} は22bitより小さくする必要があります
CIC + RSFLT（+ HPF）	$ASout_{HPF} = D1^N \times 10^{\frac{GdB_{SCALE}}{20}} \times 10^{\frac{9.5}{20}} \times Asin_{DFLT}$ DS_{SCALE} は22bitより小さくする必要があります
CIC（+ HPF）	$ASout_{HPF} = D1^N \times 10^{\frac{GdB_{SCALE}}{20}} \times Asin_{DFLT}$ DS_{SCALE} は24bitより小さくする必要があります

メイン･フィルタ･チェーンを使用する例

ADFフィルタが次のようにプログラムされている場合：

- 入力信号をシリアル･インタフェースから受信する（$Dsin_{RSFLT}$ = 1bit）。
- CICの次数 = 5（N）で、デシメーション値が24（D1）。
- SCALE[5:0] に -12dBをセット。
- RSFLTが有効で、4でのデシメーションが有効。
- HPFが有効。

まず、CIC出力でのデータ･サイズを確認してください。

$$DS_{CIC} = (5 \times \log_2(24)) + 1 \text{ bit} = 23.92 \text{ ビット}$$

サイズが26bit未満であれば、CICは正しく動作します。

CIC出力のデータ･サイズは24bitに非常に近くなるため、RSFLTに最大22bitの信号を提供するには、SCALEを調整する必要があります。12dBの減衰が必要です。

そして、RSFLTに提供される信号のレベルは次のようになります。

$$Asout_{SCALE} = 24^5 \times 10^{\frac{-12}{20}} \times 1 = 2.10^6$$

$$DS_{SCALE} = \frac{\ln(2.10^6)}{\ln(2)} + 1 = 21.93 \text{ ビット}$$

これより大きなゲインを使用する場合、RSFLTで強い入力信号用に対して出力信号が飽和することがあります。

最後に、最終的な信号振幅は次のようになります。

$$Asout_{HPF} = 24^5 \times 10^{\frac{-12}{20}} \times 10^{\frac{9.5}{20}} \times 1 = 5.9711 \times 10^6$$

$$Dsout_{HPF} = \left(\frac{\ln(5.9711 \times 10^6)}{\ln(2)} + 1\right) = 23.51 \text{ ビット}$$

または

$$SDB_{OUT} = 20 \times \log_{10}\left(\frac{2^{23.51}}{2^{24}}\right) = -2.84 \text{ dBFS}$$

40.7.6 SAD閾値の計算方法

SADで計算されるのは、変換された信号のRMS値ではなく、絶対値の平均です。そのため、レベルの推定値は信号のRMS値とは異なります。

- RMS値が1の正弦信号の場合、SADで計算されるレベルは0.9となります。
- RMS値が1の白またはピンクの雑音信号の場合、SADで計算されるレベルは約0.8となります。

注： **FRSIZE[2:0] はレベルの推定の精度に大きな影響を及ぼし、FRSIZE[2:0] の値を大きくすると精度の高い結果が得られます。**

SADMOD = 01での閾値のプログラミング

キャプチャされるサウンドが63dBSPLより大きい場合にアプリケーションによりシステムをウェイクアップさせるようなサウンド･キャプチャのケースを考えてみましょう。

サウンド･ドキャプチャは、MP45DT02などのデジタル･マイクロフォンで行えます。

このマイクロフォンの感度は、94dBSPLの入力信号で通常 -26dBFSとなります。

63dBSPLの音響信号で生成されるデジタル信号は、
- 26dBFS - (94 - 63) = - 57dBFS程度となります。

- SCALE値の調整

  この例では、フィルタ設定は次のとおりです。

  – CIC5は16でデシメーション
  – RSFLTはデシメーション比4で有効化
  – HPFは有効です。

  この設定では、SCALE値として3.5dBが推奨されます。DFTL0ではサンプルがサウンド検出用に提供されるため（サンプルはアプリケーションには提供されません）、大きなゲイン値を適用することができ、これによってSADの精度が上がり、SADの動作への飽和の影響はありません（たとえば、SCALE値を15.6dBにします）。

- 入力信号の振幅

  入力信号は - 57dBFSで、次の振幅に対応します。

  $Asin = 10^{(-57/20)} = 0.00141$ LSB。

- SAD入力での信号レベル

  SADの総フィルタ･ゲインは次のとおりです。

$$G_{SAD} = 16^5 \times 10^{\frac{15.6}{20}} \times 10^{\frac{9.5}{20}} \times \frac{1}{256} = 73.68 \times 10^3 \text{ または97.3dB}$$

  SADで受信する信号の振幅は次のとおりです。

$$Ain_{SAD} = 0.00141 \times G_{SAD} \sim 104 \text{ LSB}$$

  考えられる振幅が小さすぎる場合、ゲインを増やすことができます。対象のアプリケーションでは、104 LSBが適切です。

  入力信号がサインと考えられる場合、信号振幅104 LSBのサウンド･レベルは次のとおりです。

$$SDLVL = \frac{Ain_{SAD} \times \sqrt{2}}{2} \times 0.9 \sim 66 \text{ LSB}$$

  ここで、0.9は、RMS値について適用すべき補正係数です。

- トリガ値のプログラム

  入力信号レベルが66 LSBに達したときにSADをトリガするには、ANMINとSNTHRをプログラムする必要があります。

  SADMOD[1:0] = 01の場合、閾値は次のようになります。

$$THRH = ANMIN \times 10^{\frac{GdB_{SNTHR}}{20}}$$

  ここで、GdB_{SNTHR} はSNTHR[3:0] によって選択されるデシベル値を表します。

  たとえば、SNTHR[3:0] = 6dBの場合、この式は次のようになります。

$$THRH = 2 \times ANMIN$$

  つまり、ANMIN = THRH/2 = 66 /2 = 33 LSBとなります。

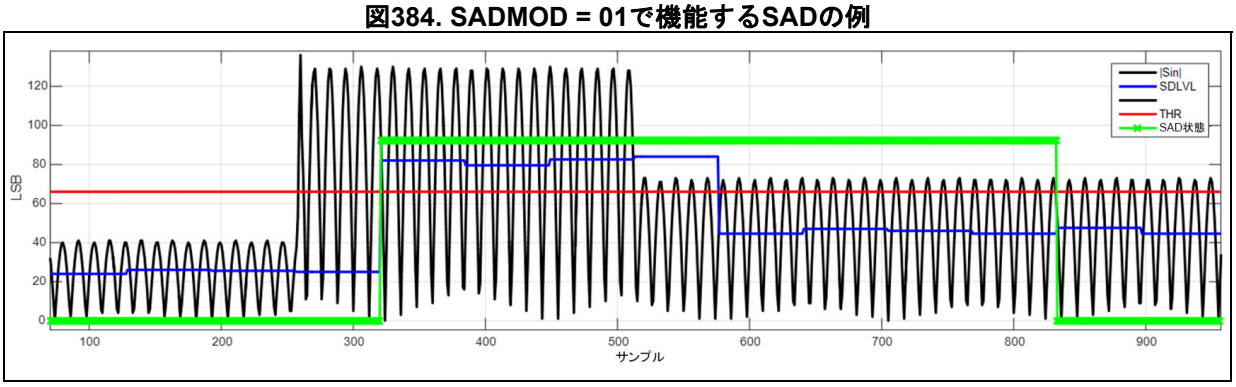
  図383では、トリガ値（赤色のTHRH）は66 LSBに固定されています。入力信号は256サンプル中は - 65dBFSで、256サンプルに対して値が - 55dBまで上がり、最終的に - 60dBFSまで下がります。

  青色の線は、時系列でのサウンド･レベル推定（SDLVL）を示しています。推定値の変動は、FRSIZEサンプルのウィンドウイイング効果により確認できます。

  ハングオーバー機能の値により、4つの追加フレームの間、SADのDETECT状態（緑色の信号がハイのとき）が維持されます。

  この例では、ANSLP = FRSIZE = 3（64サンプル）、LFRNB = 0（2フレーム）、HGOVR = 0（4フレーム）、SNTHR = 1（6dB）、ANMIN = 33 となります。

図384. SADMOD = 01で機能するSADの例



SADMOD = 1xでの閾値のプログラミング

キャプチャされるサウンドが57dBSPLより大きい場合にアプリケーションによりシステムをウェイクアップさせるようなサウンド･キャプチャのケースを考えてみましょう。

サウンド･ドキャプチャは、MP45DT02などのデジタル･マイクロフォンで行えます。このマイクロフォンの感度は、94dBSPLの入力信号で通常 -26dBFSとなります。

57dBSPLの音響信号で生成されるデジタル信号は、- 26dBFS - (94 - 63) = - 63dBFS程度となります。

- SCALE値の調整

  この例では、フィルタ設定は次のとおりです。

  – CIC4は48でデシメーション
  – RSFLTはバイパスされています。
  – HPFは有効です。

  この設定では、SCALE値として3.5dBが推奨されます。DFTL0ではサンプルがサウンド検出用に提供されるだけで、アプリケーションには提供されません。大きなゲイン値を適用することができ、これによってSADの精度が上がり、SADの動作への飽和の影響はありません（たとえば、SCALE値を24dBにします）。

- 入力信号の振幅

  入力信号は - 63dBFSで、次の振幅に対応します。:

  Asin = $10^{(-63/20)}$ = 0.000708 LSB。

- SAD入力での信号レベル

  SADの総フィルタ･ゲインは次のとおりです。

$$G_{SAD} = 48^{4} \times 10^{\frac{24}{20}} \times 0.003906 = 328.6 \times 10^{3} \text{ または} 110.3\text{dB}$$

  SADで受信する信号の振幅は次のとおりです。

$$Ain_{SAD} = 0.000708 \times G_{SAD} \sim 232 \text{ LSB}$$

  考えられる振幅が小さすぎる場合、ゲインを増やすことができます。

  入力信号がサインと考えられる場合、信号振幅232 LSBのサウンド･レベルは次のとおりです。

$$SDLVL = \frac{Ain_{SAD} \times \sqrt{2}}{2} \times 0.9 \sim 148 \text{ LSB}$$

  ここで、0.9は、RMS値について適用すべき補正係数です。

注： **ANLVLは、時間の経過につれてSDLVL値の平均に近づきます。**
そのため、57dBSPLの一定の入力信号ではSDLVL ~ ANLVL = 148 LSBとなります。

- トリガ値のプログラム

  SADMOD = 1の場合、SADでは、推定環境雑音にSNTHR[3:0] によって選択されたゲインを積算した値と、ANMIN[12:0] * 4が比較されます。

  簡単に言うと、SNTHR[3:0] に1をセットします（6dB）。つまり、ANLVLと2を積算します。

  2 * ANLVL > THRHの場合に、SADがトリガされます。

  このモードでは、次のようになります。

$$THRH = 4 \times ANMIN$$

  そのため、次の場合にSADがトリガされます。

$$ANLVL > 2 \times ANMIN$$

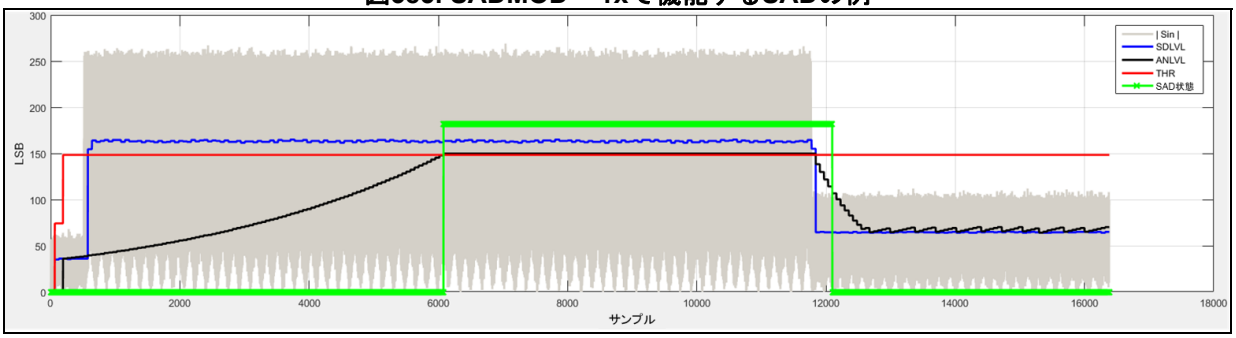
  つまり、ANMIN = 148 / 2 = 74 LSBとなります

  図384では、トリガ値（赤色のTHRH）は148 LSBに固定されています。入力信号は512サンプル中は - 75dBFSで、11000サンプルに対して値が - 62dBまで上がり、最終的に - 70dBFSまで下がります。

  青色の線は、時系列でのサウンド･レベル推定（SDLVL）を示しています。黒色の線は、時系列での環境雑音推定値を示し、対数的に増減します。学習フェーズ中にSDLVL値に達します。

  この例では、ANSLP = 6、FRSIZE = 3（64サンプル）、LFRNB = 0（2フレーム）、HGOVR = 0（4フレーム）、SNTHR = 1（6dB）、ANMIN = 74 となります。

図385. SADMOD = 1xで機能するSADの例



40.8 ADFレジスタ

ADF レジスタへのアクセスは、ワード･フォーマット（32bit）またはハーフワード･フォーマット（16bit）で行う必要があります。

40.8.1 ADFグローバル制御レジスタ（ADF_GCR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGO
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 TRGO：トリガ出力の制御

このビットは、ソフトウェアによってセットされ、ハードウェアによってリセットされます。これは、複数のフィルタの取得を同期させて開始するために使用されます。adf_trgo信号を制御することによって複数のADFを同期させることもできます。

0：0を書き込んでも、ビットの値は変化しません。0が読み出された場合、トリガに再び1をセットできることを意味します。

1：1を書き込むと、adf_trgo信号で正のパルスが生成され、ACQMOD[2:0] = 01xとし、TRGOをトリガとして選択して、有効なフィルタでの取得がトリガされます。1が読み出された場合、トリガ･パルスがアクティブなままであることを意味します。

40.8.2 ADFクロック･ジェネレータ制御レジスタ（ADF_CKGCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

このレジスタは、クロック･ジェネレータを制御するために使用されます。他のADF部分を有効化する前に、クロックadf_proc_ckを有効化する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CKGACTIVE	PROCDIV[6:0]							Res.	Res.	Res.	Res.	CCKDIV[3:0]			
r	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRGSRC[3:0]				Res.	Res.	Res.	TRGSENS	Res.	CCK1DIR	CCK0DIR	CKGMOD	Res.	CCK1EN	CCK0EN	CKGDEN
rw	rw	rw	rw				rw		rw	rw	rw		rw	rw	rw

ビット31 CKGACTIVE：クロック･ジェネレータのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、クロック･ジェネレータが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用されます。この機能の保護されたフィールドを更新できるのは、CKGACTIVE = 0のときのみです（詳細はセクション40.4.13を参照）。

CKGDENでの遷移とCKGACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のadf_proc_ckです。

0：クロック･ジェネレータはアクティブでなく、必要に応じて設定できます。

1：クロック･ジェネレータはアクティブで、保護されたフィールドは設定できません。

ビット30:24 **PROCDIV[6:0]**：シリアル･インタフェース･クロックを制御する分周回路

このビットは、ソフトウェアによってセット／リセットされます。これは、SITFに提供されるクロックの周波数を調整するために使用されます。

$$F_{adf_itf_ck} = \frac{F_{adf_ker_ck}}{(PROCDIV + 1)}$$

フィルタが有効化されている（DFTEN = 1）場合、このビットフィールドは変更しないでください。

0：SITFに供給されるadf_ker_ck

1：SITFに供給されるadf_ker_ck/2

2：SITFに供給されるadf_ker_ck/3

.....

127：SITFに供給されるadf_ker_ck/128

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **CCKDIV[3:0]**：ADF_CCKクロックを制御する分周回路

このビットは、ソフトウェアによってセット／リセットされます。これは、ADF_CCKクロックの周波数を調整するために使用されます。この分周回路の入力 クロックはSITFに供給されるクロックです。包括的に言うと、ADF_CCKの周波数は、次の式で算出できます。

$$F_{ADF_CCK} = \frac{F_{adf_ker_ck}}{(PROCDIV + 1) \times (CCKDIV + 1)}$$

フィルタが有効化されている（DFTEN = 1）場合、このビットフィールドは変更しないでください。

0000：ADF_CCKクロックはadf_proc_ckです。

0001：ADF_CCKクロックはadf_proc_ck/2です。

0010：ADF_CCKクロックはadf_proc_ck/3です。

.....

1111：ADF_CCKクロックはadf_proc_ck/16です。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット15:12 **TRGSRC[3:0]**：デジタル･フィルタ･トリガ信号の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、どの外部信号により対応するフィルタがトリガされるかを選択するために使用されます。このビットフィールドは、CKGMOD = 0の場合は重要ではありません。

000x：TRGOが選択されます。

0010：adf_trg1が選択されます。

その他：予約済みです。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 TRGSENS：CKGENトリガの感度の選択

このビットは、ソフトウェアによってセット／クリアされます。これは、トリガ信号のトリガ感度を選択するのに使用されます。このビットは、CKGMOD = 0の場合は重要ではありません。

0：立ち上がりエッジ･イベントによりCKGEN分周回路の有効化がトリガされます。

1：立ち下がりエッジ･イベントによりCKGEN分周回路の有効化がトリガされます。

注： トリガ･ソースがTRGOの場合、感度は強制的に立ち下がりエッジに設定されるため、TRGSENS値は考慮されません。このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CCK1DIR**：ADF_CCK1の方向

このビットは、ソフトウェアによってセット／リセットされます。これは、ADF_CCK1ピンの方向を制御するために使用されます。

0：ADF_CCK1ピンの方向は入力です。

1：ADF_CCK1ピンの方向は出力です。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット5 **CCK0DIR**：ADF_CCK0の方向

このビットは、ソフトウェアによってセット／リセットされます。これは、ADF_CCK0ピンの方向を制御するために使用されます。

0：ADF_CCK0ピンの方向は入力です。

1：ADF_CCK0ピンの方向は出力です。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット4 **CKGMOD**：クロック･ジェネレータのモード

このビットは、ソフトウェアによってセット／リセットされます。これは、クロック･ジェネレータの有効化方法を定義するために使用されます。フィルタが有効化されている（DFTEN = 1）場合、このビットは変更しないでください。

0：カーネル･クロックは、CKGDENに1がセットされるとすぐに分周回路に供給されます。

1：カーネル･クロックは、CKGDENに1がセットされ、トリガ条件が満たされたときに、分周回路に供給されます。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CCK1EN**：ADF_CCK1クロック有効化

このビットは、ソフトウェアによってセット／リセットされます。これは、ADF_CCK1ピンでのビットストリーム･クロックの生成を制御するために使用されます。

0：ビットストリーム･クロックが生成されません。

1：ADF_CCK1ピンでビットストリーム･クロックが生成されます。

ビット1 **CCK0EN**：ADF_CCK0クロック有効化

このビットは、ソフトウェアによってセット／リセットされます。これは、ADF_CCK0ピンでのビットストリーム･クロックの生成を制御するために使用されます。

0：ビットストリーム･クロックが生成されません。

1：ADF_CCK0ピンでビットストリーム･クロックが生成されます。

ビット0 **CKGDEN**：CKGEN分周回路は有効です。

このビットは、ソフトウェアによってセット／リセットされます。これは、CKGENのクロック分周回路、PROCDIVおよびCCKDIVを有効化／無効化するために使用されます。

0：CKGEN分周回路は無効です。

1：CKGEN分周回路は有効です。

40.8.3 ADFシリアル･インタフェース制御レジスタ0（ADF_SITF0CR）

アドレスオフセット：0x080

リセット値：0x0000：1F00

このレジスタは、シリアル･インタフェースSITF0を制御するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SITFACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	STH[4:0]					Res.	Res.	SITFMOD[1:0]		Res.	SCKSRC[1:0]		SITFEN
			rw	rw	rw	rw	rw			rw	rw		rw	rw	rw

ビット31 **SITFACTIVE**：シリアル･インタフェースのアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、シリアル･インタフェースが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用されます。この機能の保護されたフィールドを更新できるのは、SITFACTIVEに0がセットされているときのみです（詳細はセクション40.4.13を参照）。

SITFENでの遷移とSITFACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のadf_proc_ckです。

0：シリアル･インタフェースはアクティブでなく、必要に応じて設定できます。

1：シリアル･インタフェースはアクティブで、保護されたフィールドは設定できません。

ビット30:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **STH[4:0]**：マンチェスタ･シンボル閾値／SPI閾値

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、想定されるシンボルの閾値レベルを定義するために、マンチェスタ･モードで使用されます（計算の詳細は「マンチェスタ･モード」を参照）。

また、このビットフィールドは、通常SPIモードでクロックなし検出のタイムアウト値を定義するために使用されます。4未満のSTH[4:0] 値は無効です。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **SITFMOD[1:0]**：シリアル･インタフェースのタイプ

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、シリアル･インタフェースのタイプを定義するために使用されます。

00：LF_MASTER SPIモード

01：通常SPIモード

10：マンチェスタ･モード：立ち上がりエッジ = ロジック0、立ち下がりエッジ = ロジック1

11：マンチェスタ･モード：立ち上がりエッジ = ロジック1、立ち下がりエッジ = ロジック0

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:1 **SCKSRC[1:0]**：シリアル･クロック･ソース

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、シリアル･インタフェースのクロック･ソースを選択するために使用されます。

00：シリアル･クロック･ソースはADF_CCK0です。

01：シリアル･クロック･ソースはADF_CCK1です。

その他：予約済み

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット0 **SITFEN**：シリアル･インタフェースは有効です。

このビットは、ソフトウェアによってセット／クリアされます。これは、シリアル･インタフェースを有効化／無効するために使用されます。

0：シリアル･インタフェースは無効です。

1：シリアル･インタフェースは有効です。

40.8.4 ADFビットストリーム･マトリックス制御レジスタ0（ADF_BSMX0CR）

アドレスオフセット：0x084

リセット値：0x0000 0000

このレジスタは、DFLT0に提供するビットストリームを選択するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BSMXACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSSEL[4:0]				
											rw	rw	rw	rw	rw

ビット31 **BSMXACTIVE**：BSMXアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、BSMXが効果的に有効化されている（アクティブになっている）かどうかを確認するために、アプリケーションで使用されます。BSSEL[4:0] は、BSMXACTIVEに0がセットされている場合のみ更新されます。このBSMXACTIVEフラグは、DFLT0が有効化されている場合、0にできません。

0：BSMXはアクティブでなく、必要に応じて設定できます。

1：BSMXはアクティブで、保護されたフィールドは設定できません。

ビット30:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **BSSEL[4:0]**：ビットストリームの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、DFLT0用に処理されるビットストリームを選択するために使用されます。

00000：bs0_rがDFLT0に提供されます。

00001：bs0_fがDFLT0に提供されます。

その他：予約済み

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

40.8.5 ADFデジタル･フィルタ制御レジスタ0（ADF_DFLT0CR）

アドレスオフセット：0x088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFLTACTIVE	DFLTRUN	Res.	Res.	NBDIS[7:0]								Res.	Res.	Res.	Res.
r	r			rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRGSRC[3:0]				Res.	Res.	Res.	TRGSENS	Res.	ACQMOD[2:0]			Res.	FTH	DMAEN	DFLTEN
rw	rw	rw	rw				rw		rw	rw	rw		rw	rw	w

ビット31 **DFLTACTIVE**：DFLT0アクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、DFLT0がアクティブかどうか、つまりイベントを実行できるか、待っているかを示します。

0：DFLT0はアクティブではありません（必要に応じて、DFLTENを通して再有効化できます）。

1：DFLT0アクティブ

ビット30 **DFLTRUN**：DFLT0動作ステータス･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、DFLT0が動作中かどうかを示します。

0：DFLT0は動作中でなく、新しいトリガイベントを受け付けられる状態です。

1：DFLT0running

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27:20 **NBDIS[7:0]**：破棄されるサンプルの数

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、DFLT0が再開されるたびに破棄されるサンプルの数を定義するために使用されます。

0：サンプルの破棄なし

1：1つのサンプルが破棄されます

2：2つのサンプルが破棄されます

.....

255：255のサンプルが破棄されます

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **TRGSRC[3:0]**：DFLT0トリガ信号の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、どの外部信号によりDFLT0がトリガされるかを選択するために使用されます。

0000：TRGOが選択されます。

0010：adf_trgiが選択されます。

その他：予約済みです。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 TRGSENS：DFLT0トリガの感度の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、外部信号のトリガ感度を選択するために使用されます。

0：立ち上がりエッジ･イベントにより取得がトリガされます。

1：立ち下がりエッジ･イベントにより取得がトリガされます。

注： トリガ･ソースがTRGOの場合、TRGSENS値は考慮されません。TRGOを選択すると、感度は強制的に立ち下がりエッジになります。
このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **ACQMOD[2:0]**：DFLT0トリガ･モード

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、フィルタのトリガ･モードを選択するために使用されます。

000：非同期連続取得モード

001：非同期シングルショット取得モード

010：同期連続取得モード

011：同期シングルショット取得モード

100：ウィンドウ連続取得モード

その他：000と同様です。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FTH**：RXFIFO閾値選択

このビットは、ソフトウェアによってセット／クリアされます。これは、RXFIFO閾値を選択するために使用されます。

0：RXFIFOが空でない場合にRXFIFO閾値イベントが生成されます。

1：RXFIFOがハーフフルの場合にRXFIFO閾値イベントが生成されます。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット1 **DMAEN**：DMAリクエストイネーブル

このビットは、ソフトウェアによってセット／クリアされます。これは、処理済みのサンプルをメモリに転送するDMAリクエストの生成を制御するために使用されます。

0：対応するデジタル･フィルタのためのDMAインタフェースは無効です。

1：対応するデジタル･フィルタのためのDMAインタフェースは有効です。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット0 **DFLTEN**：DFLT0有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、DFLT0パスの取得の開始を制御するために使用されます。このビットの動作は、ACQMOD[2:0] と外部イベントによって決まります。サンプルを提供するシリアル･インタフェースまたはパラレル･インタフェースも有効化する必要があります。

0：取得がすぐに停止します。

1：ACQMOD[2:0] = 00xまたは101の場合は、取得がすぐに開始されます。ACQMOD[2:0] = 01xの場合は、該当するトリガ･イベントが発生したときに取得が開始されます。

40.8.6 ADFデジタル･フィルタ設定レジスタ0（ADF_DFLT0CICR）

アドレスオフセット：0x08C

リセット値：0x0000 0000

このレジスタは、メインCICフィルタを制御するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SCALE[5:0]						Res.	Res.	Res.	MCICD8
						rw	rw	rw	rw	rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCICD[7:0]								Res.	CICMOD[2:0]			Res.	Res.	DATSRC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:20 **SCALE[5:0]**：スケーリング･ファクタの選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、CIC出力に適用されるゲインを選択するために使用されます（詳細は表389を参照）。前の値がまだ適用されていないうちにアプリケーションで新しいゲイン値を書き込もうとした場合、この新しいゲイン値は無視されます。このビットフィールドを読み戻すと、アプリケーションに現在のゲイン値が通知されます。

000000：0dB
000001：+ 3.5dB
000010：+ 6dBまたは左に1bitシフト
.....
011000：+ 72dBまたは左に12bitシフト
100000：- 48.2dBまたは右に8bitシフト（デフォルト値）
100001：- 44.6dB
100010：- 42.1dBまたは右に7bitシフト
100011：- 38.6dB
.....
101110：-6dBまたは右に1bitシフト
101111：-2.5dB
その他：予約済みです。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16:8 **MCICD[8:0]**：CICのデシメーション比の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、CICのデシメーション比を選択するために使用されます。2未満のデシメーション比は指定できません。デシメーション比は（CICDEC+1）により算出されます。

0：デシメーション比は2です。
1：デシメーション比は2です。
2：デシメーション比は3です。
3：デシメーション比は4です。
.....
511：デシメーション比は512です。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **CICMOD[2:0]**：CICの次数の選択

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、MCICの次数を選択するために使用されます。

100：MCICは単一のSinc^4 フィルタに設定されます。
101：MCICは単一のSinc^5 フィルタに設定されます。
その他：予約済みです。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **DATSRC[1:0]**：デジタル･フィルタのためのソース･データ

このビットフィールドは、ソフトウェアによってセット／クリアされます。

0x：BSMXからのストリームが選択されます。
10：ADCITF1からのストリームが選択されます。
11：ADCITF2からのストリームが選択されます。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

40.8.7 ADF再形成フィルタ設定レジスタ0（ADF_DFLT0RSFR）

アドレスオフセット：0x090

リセット値：0x0000 0000

このレジスタは、再形成フィルタとHPFフィルタを制御するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HPFC[1:0]		HPFBYP	Res.	Res.	RSFLTD	Res.	Res.	Res.	RSFLTBYP
						rw	rw	rw			rw				rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **HPFC[1:0]**：ハイパス･フィルタの遮断周波数

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、ハイパス･フィルタの遮断周波数を選択するために使用されます。F_{PCM} はHPE入力でのサンプリング周波数を表します。

00：遮断周波数 = 0.000625 x FPCM

01：遮断周波数 = 0.00125 x FPCM

10：遮断周波数 = 0.00250 x FPCM

11：遮断周波数 = 0.00950 x FPCM

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 **HPFBYP**：ハイパス･フィルタのバイパス

このビットは、ソフトウェアによってセット／クリアされます。これは、ハイパス･フィルタをバイパスするために使用されます。

0：HPFはバイパスされません（デフォルト値）。

1：HPFはバイパスされています。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **RSFLTD**：再形成フィルタのデシメーション比

このビットは、ソフトウェアによってセット／クリアされます。これは、再形成フィルタ用のデシメーション比を選択するために使用されます。

0：デシメーション比は4です（デフォルト値）。

1：デシメーション比は1です。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **RSFLTBYP**：再形成フィルタはバイパスされます。

このビットは、ソフトウェアによってセット／クリアされます。これは、再形成フィルタとそのデシメーション･ブロックをバイパスするために使用されます。

0：再形成フィルタはバイパスされません（デフォルト値）。

1：再形成フィルタはバイパスされます。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

40.8.8 ADF遅延制御レジスタ0（ADF_DLY0CR）

アドレスオフセット：0x0A4

リセット値：0x0000 0000

このレジスタは、ストリームの遅延を調整するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SKPBF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SKPDLY[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31 **SKPBF**：スキップ･ビジー･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、遅延シーケンスが完了しているかどうかを制御するために使用されます。

0：ADFで新しい値をSKPDLY[6:0] に受け入れることが可能です。

1：前回の有効なSKPDLY[6:0] がまだ処理中です。

ビット30:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **SKPDLY[6:0]**：ビットストリームに適用するための遅延

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、スキップする入力サンプルの数を定義します。SKPBF = 0で、DFLTEN = 1の場合、このビットフィールドに書き込んだ直後にスキップが適用されます。SKPBF = 1の場合、レジスタに書き込まれた値は遅延状態のマシンによって無視されます。

0：サンプルはスキップされません。

1：1個の入力サンプルがスキップされます。

.....

127：127個の入力サンプルがスキップされます。

40.8.9 ADF DFLT0割込み有効化レジスタ（ADF_DFLT0IER）

アドレスオフセット：0x0AC

リセット値：0x0000 0000

このレジスタは、割込みを生成するイベントを許可または禁止するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SDLVLIE	SDDETIE	RFOVRIE	CKABIE	SATIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOVRIE	FTHIE
		rw	rw	rw	rw	rw								rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **SDLVLIE**：SADサウンドレベル値レディの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：サウンドレベルレディ割込みは無効です。

1：サウンドレベルレディ割込みは有効です。

ビット12 **SDDETIE**：サウンド･アクティビティ検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：サウンドトリガ割込みは無効です。
1：サウンドトリガ割込みは有効です。

ビット11 **RFOVRIE**：再形成フィルタ･オーバーラン割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：再形成フィルタ･オーバーラン割込みは無効です。
1：再形成フィルタ･オーバーラン割込みは有効です。

ビット10 **CKABIE**：クロックなし検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：クロックなし割込みは無効です。
1：クロックなし割込みは有効です。

ビット9 **SATIE**：飽和検出割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：飽和割込みは無効です。
1：飽和割込みは有効です。

ビット8:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DOVRIE**：データ･オーバーフロー割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：データ･オーバーフロー割込みは無効です。
1：データ･オーバーフロー割込みは有効です。

ビット0 **FTHIE**：RXFIFO閾値割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：RXFIFO閾値割込みは無効です。
1：RXFIFO閾値割込みは有効です。

40.8.10 ADF DFLT0割込みステータス･レジスタ0 （ADF_DFLT0ISR）

アドレスオフセット：0x0B0

リセット値：0x0000 0000

このレジスタには、デジタル･フィルタ･パス用のステータス･フィルタが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SDLVLF	SDDETF	RFOVRF	CKABF	SATF	Res.	Res.	Res.	Res.	Res.	RXNEF	Res.	DOVRF	FTHF
		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1						r		rc_w1	r

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **SDLVLF**：サウンド･レベル値のレディ･フラグ
このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。
0：0が読み出された場合、新しいサウンド･レベル値がレディ状態でないことを意味します。0を書き込んでも、ビットの値は変化しません。
1：1が読み出された場合、新しいサウンド･レベル値がレディ状態であることを意味します。1を書き込むと、このフラグがクリアされます。

ビット12 **SDDETF**：サウンド･アクティビティ検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、サウンド･アクティビティが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、サウンド･アクティビティが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット11 **RFOVRF**：再形成フィルタ･オーバーラン検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、再形成フィルタ･オーバーランが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、再形成フィルタ･オーバーランが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット10 **CKABF**：クロックなし検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、クロックなしが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、クロックなしが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット9 **SATF**：飽和検出フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、飽和が検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、飽和が検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット8:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RXNEF**：RXFIFOノットエンプティ･フラグ

このビットは、RXFIFOレベルに従って、ハードウェアによってセット／クリアされます。

0：RXFIFOエンプティ

1：RXFIFOは空ではありません。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DOVRF**：データ･オーバーフロー･フラグ

このビットは、ハードウェアによってセットされ、1を書き込むことでソフトウェアによってクリアされます。

0：0が読み出された場合、オーバーフローが検出されないことを意味します。0を書き込んでも、ビットの値は変化しません。

1：1が読み出された場合、オーバーフローが検出されることを意味します。1を書き込むと、このフラグがクリアされます。

ビット0 **FTHF**：RXFIFO閾値フラグ

このビットはハードウェアによってセットされ、RXFIFOレベルが閾値を下回るとハードウェアによってクリアされます。

0：RXFIFO閾値に達していない

1：RXFIFO閾値に達した

40.8.11 ADF SAD制御レジスタ（ADF_SADCR）

アドレスオフセット：0x0B8

リセット値：0x0000 0000

このレジスタは、サウンド･アクティビティ検出の設定と制御に使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SADACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SADMOD[1:0]		Res.	FRSIZE[2:0]			HYSTEN	Res.	SADST[1:0]		DETCFG	DATCAP[1:0]		SADEN
		rw	rw		rw	rw	rw	rw		r	r	rw	rw	rw	rw

ビット31 **SADACTIVE**：SADアクティブ･フラグ

このビットは、ハードウェアによってセット／クリアされます。これは、SADが効果的に有効化されている（アクティブになっている）かどうかを確認するために使用されます。この機能の保護されたフィールドとレジスタを更新できるのは、SADACTIVEに0がセットされているときのみです（詳細はセクション40.4.13を参照）。

SADENでの遷移とSADACTIVEでの遷移の間の遅延は、2周期のAHBクロックと2周期のadf_proc_ckです。

0：SADはアクティブでなく、必要に応じて設定できます

1：SADはアクティブで、保護されたフィールドは設定できません。

ビット30:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **SADMOD[1:0]**：SADの動作モード

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、SADの動作方法を定義するために使用されます。

00：閾値は、推定環境雑音に従って比較

定義された閾値よりもサウンド･レベル（SDLVL）が大きい場合に、SADがトリガされます。このモードでは、SADは音声アクティビティ検出回路のように動作します。

01：閾値は、SNTHR[3:0] によって選択されたゲインとANMIN[12:0] との積算に等しい

定義された閾値よりもサウンド･レベル（SDLVL）が大きい場合に、SADがトリガされます。このモードでは、SADはサウンド検出回路のように動作します。

1x：閾値は4 x ANMIN[12:0] により指定

SNTHR[3:0] によって選択されたゲインと推定環境雑音（ANLVL）との積算値が、定義された閾値よりも大きい場合に、SADがトリガされます。このモードでは、SADは環境雑音推定機能のように動作します。このモードでは、ヒステリシス機能は使用できません。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **FRSIZE[2:0]**：フレーム･サイズ

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、1フレームのサイズを定義するため、また短期信号レベルを計算するにあたって考慮するサンプルの数を定義するために使用されます。

000：8個のPCMサンプルを短期信号レベルの計算に使用

001：16個のPCMサンプルを短期信号レベルの計算に使用

010：32個のPCMサンプルを短期信号レベルの計算に使用

011：64個のPCMサンプルを短期信号レベルの計算に使用

100：128個のPCMサンプルを短期信号レベルの計算に使用

101：256個のPCMサンプルを短期信号レベルの計算に使用

11x：512個のPCMサンプルを短期信号レベルの計算に使用

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 **HYSTEN**：ヒステリシスの有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、ヒステリシス機能を有効化／無効化するために使用されます（詳細は*表389* を参照）。SADMOD[1:0] = 1xのとき、このビットは0にセットしておく必要があります。

0：ヒステリシス機能は無効です。常にTHR_H が使用されます。

1：ヒステリシス機能は有効です。MONITORからDETECTへの遷移ではTHR_H が使用され、DETECTからMONITORへの遷移ではTHR_L が使用されます。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **SADST[1:0]**：SADの状態

このビットフィールドは、ハードウェアによってセット／クリアされます。これは、SADの状態を示し、SADEN = 1の場合のみ意味があります。SADの状態には以下があります。

- SADが学習フェーズ、またはSDLVL計算モードの場合はLEARN
- SADが監視フェーズの場合はMONITOR
- SADでサウンドが検出される場合はDETECT

00：SADがLEARN状態

01：SADがMONITOR状態

11：SADがDETECT状態

ビット3 **DETCFG**：サウンド･トリガ･イベントの設定

このビットは、ソフトウェアによってセット／クリアされます。これは、sddet_evtイベントが生成されるのが、SADのMONITOR状態が開始した場合のみか、それともSADのDETECT状態が開始または終了したときかを定義するために使用されます。

0：SADのMONITOR状態が開始した場合に、sddet_evtが生成されます。

1：SADのDETECT状態が開始または終了した場合に、sddet_evtが生成されます。

注： このビットは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット2:1 **DATCAP[1:0]**：データ･キャプチャ･モード

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、DLFT0によって提供されたサンプルをメモリに格納する際の条件を定義するために使用されます。

00：DFLT0からのサンプルをメモリに転送しません。

01：SADがDETECT状態のときに、DFLT0からのサンプルをメモリに転送します。

1x：SADおよびDFLT0が有効化されているときに、DFL0からのサンプルをメモリに転送します。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット0 **SADEN**：サウンド･アクティビティ検出回路の有効化

このビットは、ソフトウェアによってセット／クリアされます。これは、SADを有効化／無効するために使用されます。

0：SADは無効で、SADの状態がリセットされます。

1：SADは有効です。

40.8.12 ADF SAD設定レジスタ（ADF_SADCFGR）

アドレスオフセット：0x0BC

リセット値：0x0000 0000

このレジスタは、サウンド･アクティビティ検出を設定するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ANMIN[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	HGOVR[2:0]			Res.	LFRNB[2:0]			Res.	ANSLP[2:0]			SNTHR[3:0]			
	rw	rw	rw		rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:16 **ANMIN[12:0]**：最小雑音レベル

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、最小雑音レベルを定義し、その後に感度を定義するために使用されます。正の数字で表します。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **HGOVR[2:0]**：ハングオーバー時間枠

このビットフィールドは、ソフトウェアによってセット／クリアされます。SAD状態がDETECTになると、このパラメータは、サウンドが閾値を下回っていることを許可され、SADがMONITOR状態に切り替わることはない時間の長さを定義するために使用されます。

000：サウンドが閾値を下回っている時間が4フレームに達すると、SADはMONITOR状態に戻ります。
001：サウンドが閾値を下回っている時間が8フレームに達すると、SADはMONITOR状態に戻ります。
010：サウンドが閾値を下回っている時間が16フレームに達すると、SADはMONITOR状態に戻ります。
011：サウンドが閾値を下回っている時間が32フレームに達すると、SADはMONITOR状態に戻ります。
100：サウンドが閾値を下回っている時間が64フレームに達すると、SADはMONITOR状態に戻ります。
101：サウンドが閾値を下回っている時間が128フレームに達すると、SADはMONITOR状態に戻ります。
110：サウンドが閾値を下回っている時間が256フレームに達すると、SADはMONITOR状態に戻ります。
111：サウンドが閾値を下回っている時間が512フレームに達すると、SADはMONITOR状態に戻ります。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **LFRNB[2:0]**：学習フレームの数

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、雑音レベルの最初の推定を行うための学習フレームの数を定義するために使用されます。

000：最初の雑音レベルを計算するために2フレームを使用します。
001：最初の雑音レベルを計算するために4フレームを使用します。
010：最初の雑音レベルを計算するために8フレームを使用します。
011：最初の雑音レベルを計算するために16フレームを使用します。
1xx：最初の雑音レベルを計算するために32フレームを使用します。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **ANSLP[2:0]**：環境雑音のスロープ制御

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、ANLVLの更新を行う雑音推定機能の正負のスロープを定義するために使用されます（このビットフィールドのプログラミングについては、「環境雑音推定（ANLVL）」を参照）。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

ビット3:0 **SNTHR[3:0]**：雑音閾値への信号

このビットフィールドは、ソフトウェアによってセット／クリアされます。これは、THR_H（ヒステリシスが有効化されている場合はTHR_L も）を定義するために使用されます。詳細については、表389を参照してください。

0000：THR_H をANLVLよりも3.5dB高くする
0001：THR_H をANLVLよりも6.0dB高くする
0010：THR_H をANLVLよりも9.5dB高くする
0011：THR_H をANLVLよりも12dB高くする
0100：THR_H をANLVLよりも15.6dB高くする
0101：THR_H をANLVLよりも18dB高くする
0110：THR_H をANLVLよりも21.6dB高くする
0111：THR_H をANLVLよりも24.1dB高くする
1000：THR_H をANLVLよりも27.6dB高くする
1001：THR_H をANLVLよりも30.1dB高くする
上記以外：予約済みです。

注： このビットフィールドは書込み保護できます（詳細はセクション40.4.13を参照）。

40.8.13 ADF SADサウンド･レベル･レジスタ（ADF_SADSDLVR）

アドレスオフセット：0x0C0

リセット値：0x0000 0000

このレジスタには、SADによって計算される短期サウンド･レベルが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SDLVL[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **SDLVL[14:0]**：短期サウンド･レベル

このビットフィールドはハードウェアによってセットされます。ここには、SADによって計算される最新のサウンド･レベルが格納されます。この値を一掃するには、SDLVLFをクリアする必要があります。

40.8.14 ADF SAD環境雑音レベル･レジスタ（ADF_SADANLVR）

アドレスオフセット：0x0C4

リセット値：0x0000 0000

このレジスタには、SADによって計算される環境雑音レベルが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ANLVL[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **ANLVL[14:0]**：環境雑音レベルの推定

このビットフィールドはハードウェアによってセットされます。ここには、SADによって計算される最新の環境雑音レベルが格納されます。このビットフィールドを一掃するには、SDLVLFフラグをクリアする必要があります。

40.8.15 ADFデジタル･フィルタ･データ･レジスタ0（ADF_DFLT0DR）

アドレスオフセット：0x0F0

リセット値：0x0000 0000

このレジスタは、デジタル･フィルタによって処理されたデータを読み出すために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r								

ビット31:8 **DR[23:0]**：DFT0によって処理されるデータ

ビット7:0 予約済みであり、リセット値に保持する必要があります。

40.8.16 ADFレジスタマップ

表404.ADFレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	ADF_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGO
	リセット値																																0
0x004	ADF_CKGCR	CKGACTIVE	PROCDIV[6:0]							Res.	Res.	Res.	Res.	CCKDIV [3:0]				TRGSRC [3:0]				Res.	Res.	Res.	TRGSENS	Res.	CCK1DIR	CCK0DIR	CKGMOD	Res.	CCK1EN	CCK0EN	CKGDEN
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0				0		0	0	0		0	0	0
0x008 - 0x07C	予約済みです。	予約済みです。																															
0x080	ADF_SITF0CR	SITFACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STH[4:0]				Res.	Res.	SITFMOD [1:0]		Res.	SCKSRC [1:0]		SITFEN
	リセット値	0																			1	1	1	1	1			0	0		0	0	0
0x084	ADF_BSMX0CR	BSMXACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSSEL[4:0]				
	リセット値	0																											0	0	0	0	0

表404.ADFレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x088	ADF_DFLT0CR	DFLTACTIVE	DFLTRUN	Res.	Res.	NBDIS[7:0]								Res.	Res.	Res.	Res.	TRGSRC [3:0]				Res.	Res.	Res.	TRGSENS	Res.	ACQMOD [2:0]			Res.	FTH	DMAEN	DFLTEN
	リセット値	0	0			0	0	0	0	0	0	0	0					0	0	0	0				0		0	0	0		0	0	0
0x08C	ADF_DFLT0CICR	Res.	Res.	Res.	Res.	Res.	Res.	SCALE[5:0]						Res.	Res.	Res.	MCICD[8:0]									Res.	CICMOD[2:0]			Res.	Res.	DATSRC[1:0]	
	リセット値							0	0	0	0	0	0				0	0	0	0	0	0	0	0	0		0	0	0			0	0
0x090	ADF_ DFLT0RSFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HPFC[1:0]		HPFBYP	Res.	Res.	RSFLTD	Res.	Res.	Res.	RSFLTBYP
	リセット値																							0	0	0			0				0
0x094 - 0x0A0	予約済みです。	予約済みです。																															
0x0A4	ADF_DLY0CR	SKPBF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SKPDLY[6:0]						
	リセット値	0																									0	0	0	0	0	0	0
0x0A8	予約済みです。	予約済みです。																															
0x0AC	ADF_DFLT0IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDLVLIE	SDDETIE	RFOVRIE	CKABIE	SATIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOVRIE	FTHIE
	リセット値																		0	0	0	0	0	0	0				0		0	0	0
0x0B0	ADF_DFLT0ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDLVLF	SDDETF	RFOVRF	CKABF	SATF	Res.	Res.	Res.	Res.	Res.	Res.	RXNEF	Res.	DOVRF	FTHF
	リセット値																		0	0	0	0	0							0		0	0
0x0B4	予約済みです。	予約済みです。																															
0x0B8	ADF_SADCR	SADACTIVE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SADMOD[1:0]		Res.	FRSIZE[2:0]			HYSTEN	Res.	SADST[1:0]		DETCFG	DATCAP[1:0]		SADEN
	リセット値	0																		0	0		0	0	0	0		0	0	0	0	0	0
0x0BC	ADF_SADCFGR	Res.	Res.	Res.	ANMIN[12:0]													Res.	HGOVR [2:0]			Res.	LFRNB [2:0]			Res.	ANSLP[2:0]			Res.	SNTHR [2:0]		
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0	0		0	0	0
0x0C0	ADF_SADSDLVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDLVL[14:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C4	ADF_SADANLVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ANLVL[14:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C8 - 0x0EC	予約済みです。	予約済みです。																															

表404.ADFレジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0F0	ADF_DFLT0DR	DR[23:0]																								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								

レジスタ境界アドレスについては、セクション2.3を参照してください。

41 デジタルカメラインタフェース（DCMI）

41.1 概要

このデジタルカメラインタフェースは、外部8、10、12、または14bit CMOSカメラモジュールから高速データフローを受信できる同期型パラレルインタフェースです。YCbCr4:2:2/RGB565プログレッシブビデオおよび圧縮データ（JPEG）など、さまざまなデータフォーマットをサポートします。

41.2 DCMIの主な機能

- 8、10、12、または14bitのパラレルインタフェース
- 埋め込み／外部ラインおよびフレーム同期
- 連続モードまたはスナップショットモード
- クロップ機能
- 以下のデータフォーマットに対応：
  - 8/10/12/14bitのプログレッシブビデオ：モノクロまたはRawベイヤー
  - YCbCr 4:2:2プログレッシブビデオ
  - RGB 565プログレッシブビデオ
  - 圧縮データ：JPEG

41.3 DCMIの機能説明

デジタルカメラインタフェースは、高速のデータフローを受信できる同期型パラレルインタフェースです。最大14本のデータライン（DCMI_D[13:0]）と1本のピクセルクロックライン（DCMI_PIXCLK）で構成されています。ピクセルクロックの極性はプログラム可能であるため、データはピクセルクロックの立ち上がりエッジまたは立ち下がりエッジのどちらでもキャプチャすることができます。

データは32bitのデータレジスタ（DCMI_DR）に格納され、汎用DMAチャネルを介して転送されます。画像バッファはカメラインタフェースではなくDMAが管理します。

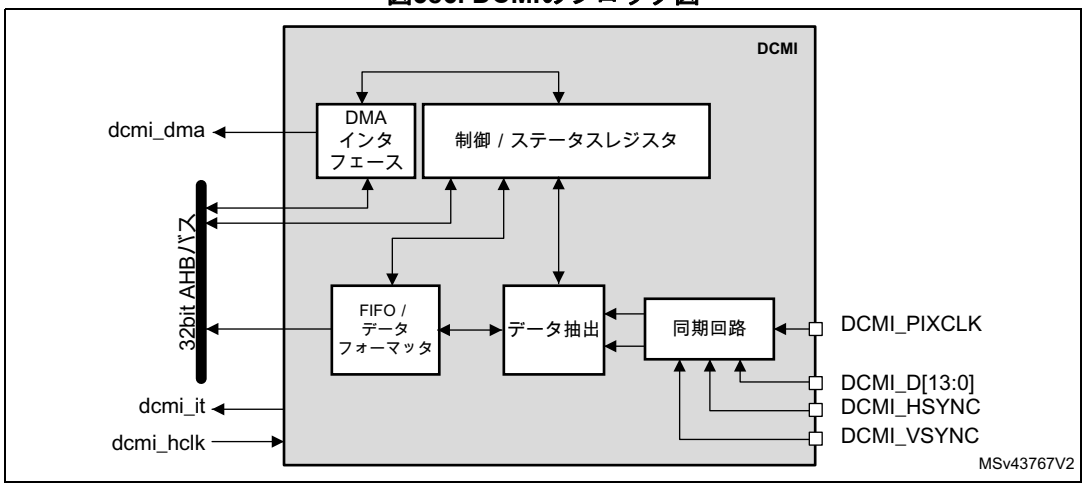
カメラから受信したデータは、ライン/フレーム（Raw YUV/RGB/Bayerモード）で構成したり、一連のJPEG画像にすることができます。JPEG画像を受信できるようにするには、JPEGビット（DCMI_CRレジスタのビット3）をセットする必要があります。

データフローは、オプションのDCMI_HSYNC（水平同期）信号およびDCMI_VSYNC（垂直同期）信号を用いてハードウェアで同期をとることもできれば、データフローに埋め込まれてた同期コードで同期をとることもできます。

41.3.1 DCMIのブロック図

図386 にDCMIブロック図を示します。

図386. DCMIのブロック図



41.3.2 DCMIピンおよび内部信号

次の表に、DCMIのピンを示します。

表405.DCMIの入出力ピン

モード	ピン名	信号タイプ	説明
8bit 10bit 12bit 14bit	DCMI_D[7:0] DCMI_D[9:0] DCMI_D[11:0] DCMI_D[13:0]	入力	DCMIデータ
DCMI_PIXCLK		入力	ピクセルクロック
DCMI_HSYNC		入力	水平同期/データ有効
DCMI_VSYNC		入力	垂直同期

次の表に、DCMIの内部信号を示します。

表406.DCMI内部入力／出力信号

内部信号名	信号タイプ	説明
dcmi_dma	出力	DCMI DMAリクエスト
dcmi_it	出力	DCMI割込みリクエスト
dcmi_hclk	入力	DCMIインタフェースクロック

41.3.3 DCMIクロック

デジタルカメラインタフェースは、DCMI_PIXCLKとHCLKの2つのクロックドメインを使用します。DCMI_PIXCLKとHCLKが安定した後、DCMI_PIXCLKで生成された信号は、HCLKの立ち上がりエッジでサンプリングされます。カメラからのデータが安定し、サンプリング可能状態であることを示すため、イネーブル信号がHCLKドメインで生成されます。最大DCMI_PIXCLK周期は、HCLK 2.5周期分よりも長い必要があります。

41.3.4 DCMI DMAインタフェース

DCMI_CRレジスタのCAPTUREビットがセットされているときには、DMAインタフェースがアクティブです。カメラインタフェースがレジスタにまとまった32bitデータブロックを受信するたびに、DMAリクエストが生成されます。

41.3.5 DCMI物理インタフェース

インタフェースは、11/13/15/17個の入力で構成されています。スレーブモードにのみ対応しています。

カメラインタフェースは、DCMI_CRレジスタのEDM[1:0] ビットに従って、8、10、12、または14ビットのデータをキャプチャすることができます。使用されるのが14bit未満である場合、使用されない入力ピンはグラウンドに接続する必要があります。

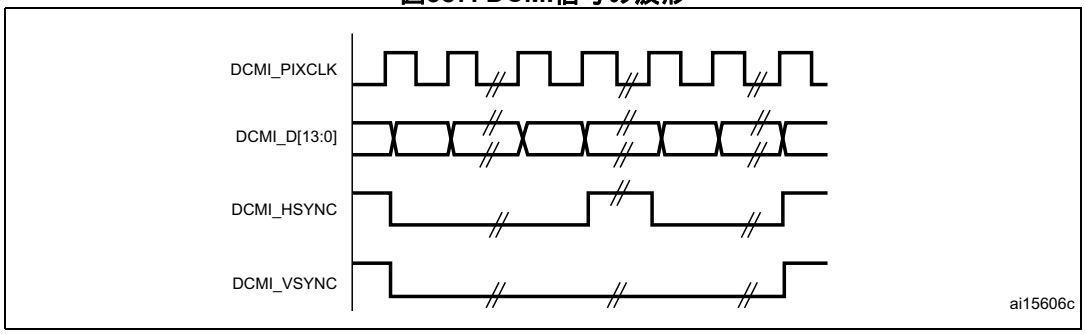
DCMIのピンを表405 に示します。

データはDCMI_PIXCLKと同期していますが、ピクセルクロックの立上がり/立下がりエッジにおけるデータは極性によって変わります。

DCMI_HSYNC信号は、ラインの開始/終了を示します。

DCMI_VSYNC信号は、フレームの開始/終了を示します。

図387. DCMI信号の波形



1. DCMI_PIXCLKのキャプチャエッジは立ち下がりエッジであり、DCMI_HSYNCとDCMI_VSYNCのアクティブ状態は1です。
2. DCMI_HSYNCとDCMI_VSYNCの状態は、同時に変化することがあります。

8bitデータ

DCMI_CRでEDM[1:0] = 00の場合、このインタフェースは入力（DCMI_D[7:0]）でLSBの8bitをキャプチャし、それを8bitデータとして記憶します。DCMI_D[13:8] 入力は無視されます。この場合、32bitワードをキャプチャするには、カメラインタフェースでピクセルクロック4サイクルが必要です。

最初にキャプチャされたデータバイトは32bitワードのLSB位置に置かれ、4番目にキャプチャされたデータバイトは32bitワードのMSB位置に置かれます。次の表に、2つの32bitワードでキャプチャされたデータバイトの配置例を示します。

表407.32bitワード（8bit幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

10bitデータ

DCMI_CRでEDM[1:0] = 01の場合、カメラインタフェースは入力（DCMI_D[9:0]）で10bitデータをキャプチャし、それを16bitワードのLSBの10bitとして記憶します。DCMI_DRレジスタの残りのMSB（ビット11から15）はゼロにクリアされます。したがって、この場合、32bitデータワードはピクセルクロック2サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは32bitワードのLSB位置に置かれ、2番目にキャプチャされたデータは32bitワードのMSB側に置かれます。

表408.32bitワード（10bit幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:26	25:16	15:10	9:0
0	0	D_{n+1}[9:0]	0	D_n[9:0]
4	0	D_{n+3}[9:0]	0	D_{n+2}[9:0]

12bitデータ

DCMI_CRでEDM[1:0] = 10の場合、カメラインタフェースは入力（DCMI_D[11:0]）で12bitデータをキャプチャし、それを16bitワードのLSBの12bitとして記憶します。残りのMSBはゼロにクリアされます。したがって、この場合、32bitデータワードはピクセルクロック2サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは32bitワードのLSB位置に置かれ、2番目にキャプチャされたデータは32bitワードのMSB側に置かれます。

表409.32bitワード（12bit幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:28	27:16	15:12	11:0
0	0	D_{n+1}[11:0]	0	D_n[11:0]
4	0	D_{n+3}[11:0]	0	D_{n+2}[11:0]

14bitデータ

DCMI_CRでEDM[1:0] = 11の場合、カメラインタフェースは入力（DCMI_D[13:0]）で14bitデータをキャプチャし、それを16bitワードのLSBの14bitとして記憶します。残りのMSBはゼロにクリアされます。したがって、この場合、32bitデータワードはピクセルクロック2サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは32bitワードのLSB位置に置かれ、2番目にキャプチャされたデータは32bitワードのMSB側に置かれます。

表410.32bitワード（14bit幅）でキャプチャされたデータバイトの配置

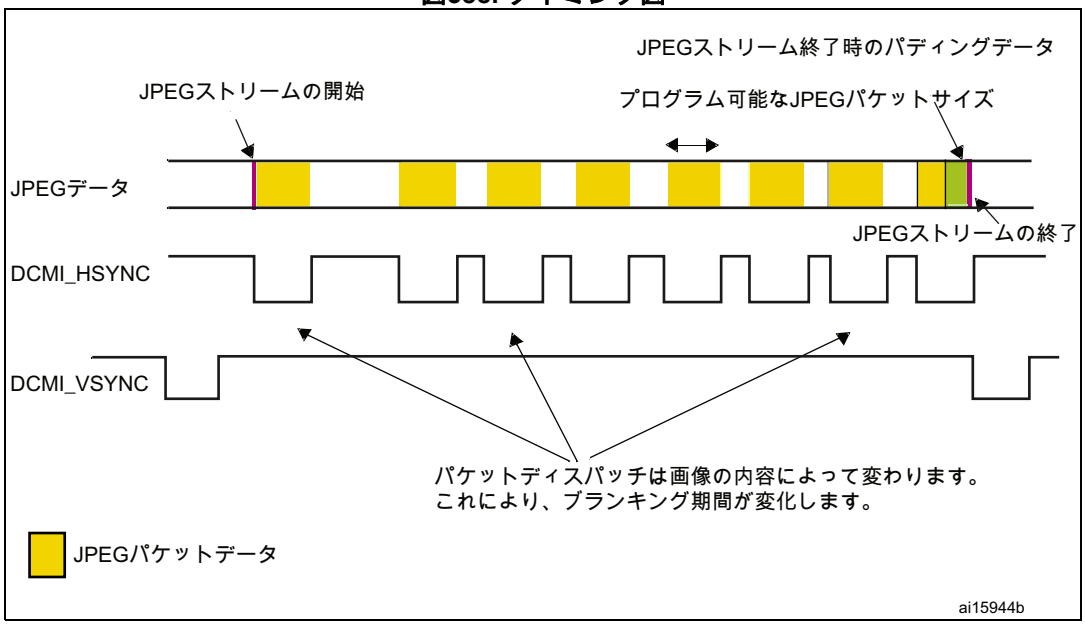
バイトアドレス	31:30	29:16	15:14	13:0
0	0	D_{n+1}[13:0]	0	D_n[13:0]
4	0	D_{n+3}[13:0]	0	D_{n+2}[13:0]

41.3.6 DCMIの同期

このデジタルカメラインタフェースは、データ同期またはハードウェア（DCMI_HSYNCおよびDCMI_VSYNC）同期に対応します。埋め込み同期を使用する場合、0x00および0xFFの値が（有効画像データではなく）同期のためにのみ使用されているのを確認するのはデジタルカメラモジュールの仕様に依存します。埋め込み同期コードは、8bitパラレルデータインタフェース幅（すなわちDCMI_CRレジスタのEDM[1:0] ビットをクリア）に対してのみサポートされています。

圧縮データに対しては、DCMIはハードウェア同期モードにのみ対応します。この場合、DCMI_VSYNCは画像の開始／終了として使用し、DCMI_HSYNCはデータ有効信号として使用されます。図388 に対応するタイミング図を示します。

図388. タイミング図



ハードウェア同期モード

ハードウェア同期モードでは、2つの同期信号（DCMI_HSYNC/DCMI_VSYNC）を使用します。

カメラモジュール／モードによっては、データが水平／垂直同期期間中に転送されることがあります。DCMI_HSYNC/DCMI_VSYNCのアクティブ期間に受信されるデータはすべて無視されるため、これらの信号はブランキング信号のように振る舞います。

また、画像を正しくDMA/RAMバッファに転送するため、データ転送はDCMI_VSYNC信号と同期します。ハードウェア同期モードを選択し、キャプチャを有効（DCMI_CRのCAPTUREビットをセットする）にすると、データ転送はDCMI_VSYNC信号の無効化（フレームの次の開始）と同期します。

その後、連続したバッファまたは同じ／サーキュラバッファへDMAによって転送された連続フレームにより、連続的な転送ができるようになります。連続したフレームのDMA管理を可能にするため、各フレームの終了時にはVSIF（垂直同期割込みフラグ）がアクティブになります。

データ同期モード

この同期モードでは、データフローに埋め込まれている32bitコードを使用してデータフローを同期させます。このコードには、有効画像データでは使用されていない 0x00/0xFF値を使用します。コードには4種類あり、いずれも0xFF0000XYフォーマットを用います。埋め込み同期コードは、8bitパラレルデータ幅キャプチャ（DCMI_CRレジスタのEDM[1:0] ビットをクリア）においてのみサポートされています。他のデータ幅に対しては、このモードは予期しない動作となるため、使用しないでください。

注： **カメラモジュールでは、このようなコードが8つ使用可能です（インタリーブモード）。このため、インタリーブモードはDCMIではサポートされません（ハーフフレームごとに捨てられてしまうため）。**

- モード2：

  4種類の同期コードにより、次のイベントが通知されます。

  – フレーム開始（FS）
  – フレーム終了（FE）
  – ライン開始（LS）
  – ライン終了（LE）

  4種類のコードの0xFF0000XYフォーマットにおけるXY値は、プログラム可能です（セクション 41.5.7：DCMI埋め込み同期コードレジスタ（DCMI_ESCR）を参照）。

  「フレーム終了」としてプログラムされた値“0xFF”は、使用されていないすべてのコードが有効なフレーム終了コードとして解釈されることを意味します。

  このモードでは、カメラインタフェースが有効になると、最初に発生したフレーム終了（FE）コードの後にフレーム開始（FS）コードが続くと、フレームのキャプチャを開始します。

- モード1：

  もう一方のデータ同期モードとして、カメラモード1が使用できます。このモードはITU656対応です。

  以下の同期コードにより次の各イベントが通知されます。

  – SAV（アクティブライン）- ライン開始
  – EAV（アクティブライン）- ライン終了
  – SAV（ブランキング）- インタフレームブランキング期間中のライン終了
  – EAV（ブランキング）- インタフレームブランキング期間中のライン終了

このモードは、次のコードをプログラムすることにより対応できます。

- FS ≤ 0xFF
- FE ≤ 0xFF
- LS ≤ SAV（アクティブ）
- LS ≤ EAV（アクティブ）

埋め込み型マスク解除コードもまた、フレーム／ラインの開始コードおよびラインの終了コード用に準備されています。これを使用すれば、プログラムされたコードに対して選択したマスク解除ビットのみを比較することができます。したがって、1つのビットを選択して埋め込みコード内で比較し、フレーム／ラインの開始や終了を検出することができます。つまり、マスク解除ビットの位置が同じでも、フレーム／ラインの開始や終了のコードは異なる可能性があるということです。

例

FS = 0xA5

FSのマスク解除コード = 0x10

この場合、フレーム開始コードがフレーム開始コードのビット4に埋め込まれています。

41.3.7 DCMIキャプチャモード

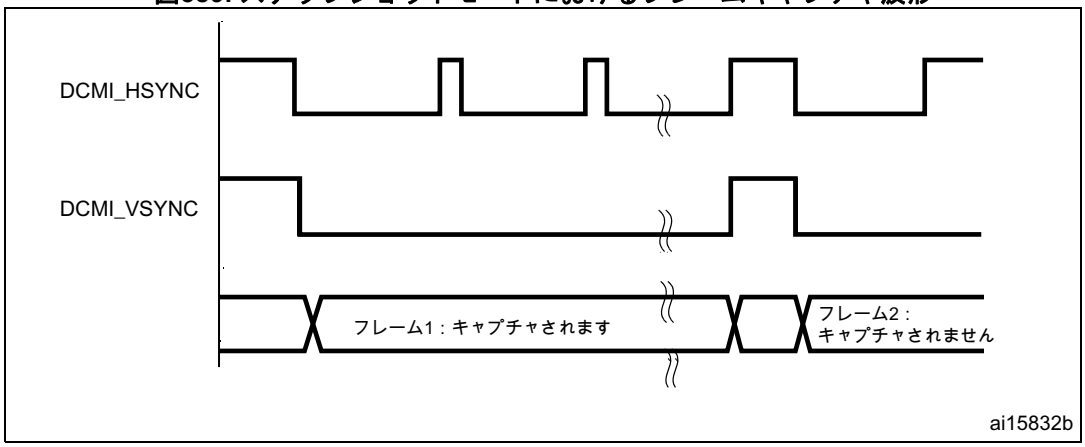
このインタフェースは、スナップショット（単一フレーム）、連続グラブの2種類のキャプチャをサポートしています。

スナップショットモード（単一フレーム）

このモードでは、1つのフレームをキャプチャします（DCMI_CRレジスタのCM = 1）。DCMI_CRでCAPTUREビットがセットされると、インタフェースはフレームの開始が検出されるのを待ってからデータをサンプリングします。最初の完全なフレームを受信した後、カメラインタフェースは自動的に無効になります（DCMI_CRのCAPTUREビットがクリアされます)。割込みが有効になっている場合、割込み（IT_FRAME）が生成されます。

オーバーランの場合、フレームが失われ、CAPTUREビットがクリアされます。

図389. スナップショットモードにおけるフレームキャプチャ波形

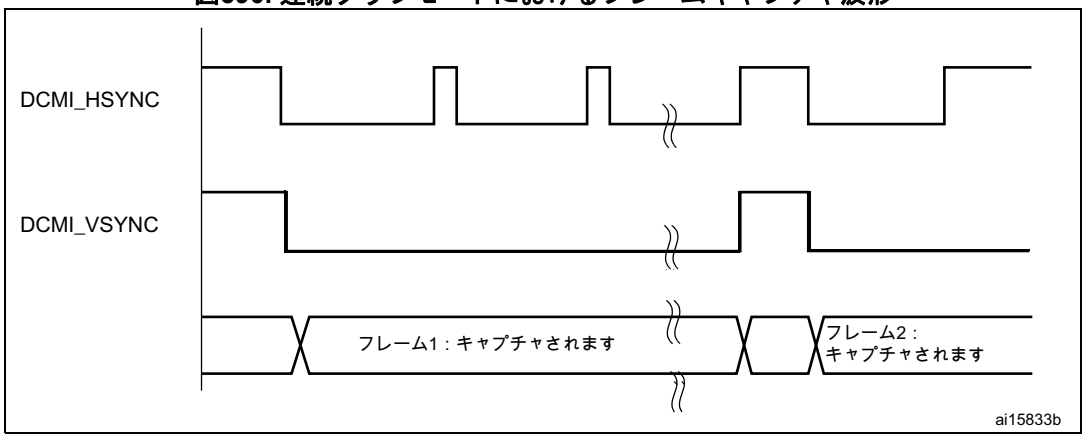


1. ここで、DCMI_HSYNCとDCMI_VSYNCのアクティブ状態は1です。
2. DCMI_HSYNCとDCMI_VSYNCの状態は、同時に変化することがあります。

連続グラブモード

このモード（DCMI_CRのCMビット = 0）では、DCMI_CRのCAPTUREビットがセットされると、モードによっては次のDCMI_VSYNCまたは埋め込みフレーム開始でグラブ処理が開始されます。この処理は、DCMI_CRのCAPTUREビットがクリアされるまで継続します。いったんCAPTUREビットがクリアされると、現在のフレームの終了までグラブ処理が続きます。

図390. 連続グラブモードにおけるフレームキャプチャ波形



1. ここで、DCMI_HSYNCとDCMI_VSYNCのアクティブ状態は1です。
2. DCMI_HSYNCとDCMI_VSYNCの状態は、同時に変化することがあります。

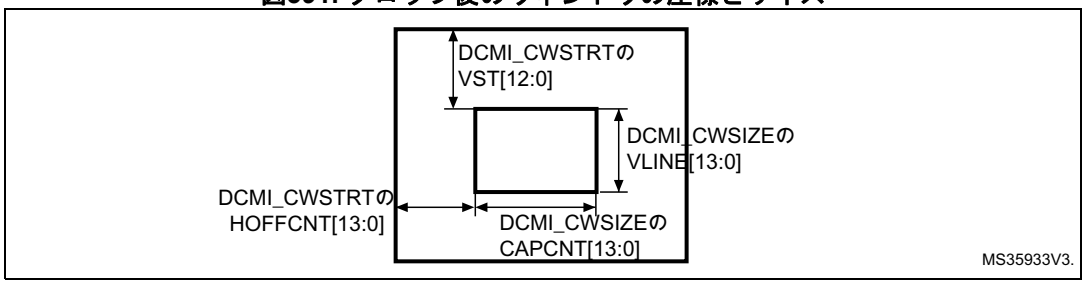
連続グラブモードでは、DCMI_CRのFCRC[1:0] ビットを設定して、すべての画像をグラブ、1枚おきにグラブ、または4枚おきにグラブしてフレームキャプチャ速度を下げることができます。

注： **ハードウェア同期モード（DCMI_CRのESS = 0）では、IT_VSYNC割込みが有効になっていればDCMI_CRでCAPTURE = 0になっていてもIT_VSYNC割込みが発生しますので、スナップショットモードと併せてIT_VSYNC割込みを使用し、2つのキャプチャ間でのフレーム数を数えることにより、さらにフレームキャプチャレートを下げることができます。これは埋め込みデータ同期モードでは行うことができません。**

41.3.8 DCMIクロップ機能

クロップ機能を用いて、カメラインタフェースが受信した画像から長方形のウィンドウを選択することができます。2つの32bitレジスタ（DCMI_CWSTRTおよびDCMI_CWSIZE）を使用して開始（左上の角）座標とサイズ（水平方向はピクセルクロック数、垂直方向はライン数）を指定します。ウィンドウのサイズは、ピクセルクロック数（水平方向）とライン数（垂直方向）で指定します。

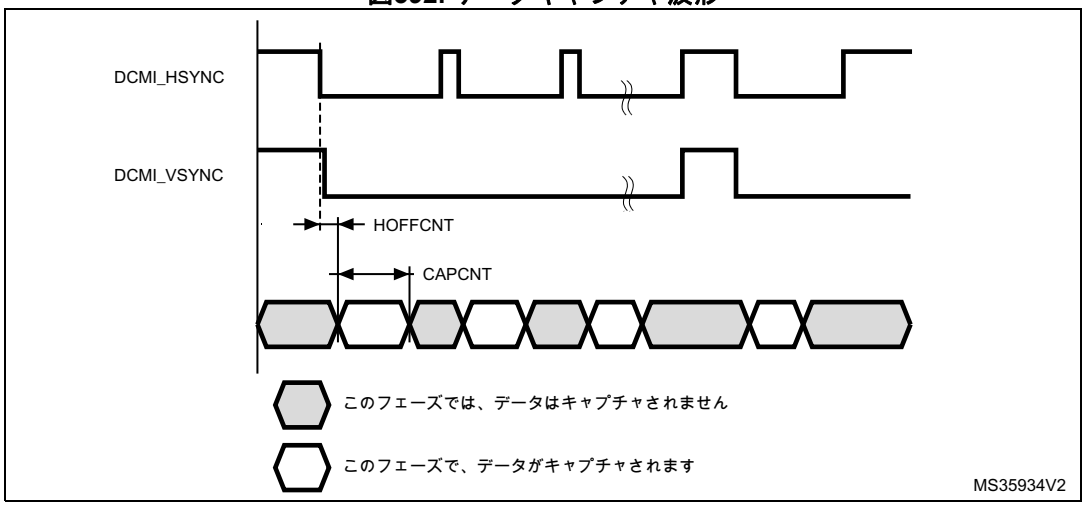
図391. クロップ後のウィンドウの座標とサイズ



これらのレジスタを用いて、キャプチャウィンドウの開始点の座標をライン番号（フレーム内、0から開始）とピクセルクロック数（ライン上、0から開始）で指定し、ウィンドウのサイズをライン番号とピクセルクロック数で指定します。CAPCNT値は、DMAを介してデータを正しく転送できるよう、必ず4の倍数となります（LSBの2bitは強制的に0）。

DCMI_CWSIZEレジスタでライン番号が指定される前にDCMI_VSYNC信号がアクティブになると、キャプチャが停止し、IT_FRAME割込みが有効になっていればIT_FRAME割込みが生成されます。

図392. データキャプチャ波形



1. ここで、DCMI_HSYNCとDCMI_VSYNCのアクティブ状態は1です。
2. DCMI_HSYNCとDCMI_VSYNCの状態は、同時に変化することがあります。

41.3.9 DCMI JPEGフォーマット

JPEG画像を受信できるようにするには、DCMI_CRレジスタのJPEGビットをセットする必要があります。JPEG画像はラインやフレームとして保存されないので、DCMI_HSYNCがデータイネーブル信号として使用されている間に、DCMI_VSYNC信号を使用してキャプチャを開始します。1ライン内のバイト数は4の倍数でないこともあります。キャプチャされたデータから完全な32bitワードが構成されるたびにDMAリクエストが生成されるため、この場合のデータ操作には注意が必要です。フレームの終了が検出され、転送される32bitワードが完全に受信されていない場合、残りのデータはゼロでパディングされ、DMAリクエストが生成されます。

クロップ機能および埋め込み同期コードは、JPEG形式では使用できません。

41.3.10 DCMIのFIFO

AHB上でのデータ転送レートを管理するため、8ワードのFIFOが搭載されています。このDCMIには、カメラインタフェースがAHBから読出しを行うたびにインクリメントされる読出しポインタと、FIFOに書込みを行うたびにインクリメントされる書込みポインタを持った単純なFIFOコントローラを備えています。AHBインタフェースがデータ転送速度を維持できない場合にデータの上書きを防ぐオーバーラン保護はありません。

オーバーランや同期信号でのエラーの場合、FIFOはリセットされ、DCMIインタフェースは新規フレームの開始を待ちます。

41.3.11 DCMIデータフォーマットの説明

データフォーマット

サポートされるデータは次の3種類です。

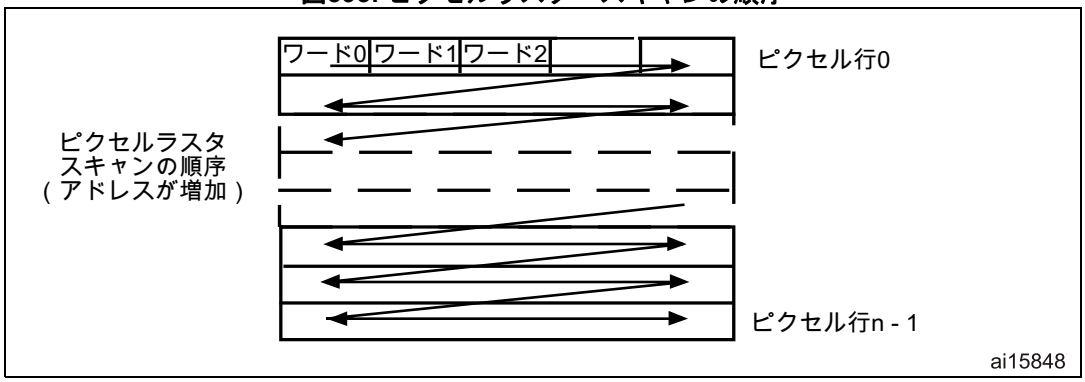
- 8/10/12/14bitのプログレッシブビデオ：モノクロまたはRawベイヤーフォーマット
- YCbCr 4:2:2プログレッシブビデオ
- RGB565プログレッシブビデオ。16bit（青が5bit、赤が5bit、緑が6bit）で符号化されたピクセルの転送には、クロック2サイクルが必要です。

圧縮データ：JPEG

モノクロ（白黒）、YCbCr、RGBデータの場合、最大入力サイズは2048 × 2048ピクセルです。JPEG圧縮モードでは制限はありません。

モノクロ、RGB、YCbCrの場合、フレームバッファはラスターモードで保存されます。32bitワードが使用されます。リトルエンディアン形式のみをサポートします。

図393. ピクセルラスタースキャンの順序



モノクロフォーマット

特徴：

- ラスタフォーマット
- ピクセル当たり8bit

次の表に、データ保存方法を示します。

表411.モノクロプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	n + 3	n + 2	n + 1	n
4	n + 7	n + 6	n + 5	n + 4

RGBフォーマット

特徴：

- ラスタフォーマット
- RGB
- インタリーブ：バッファ 1つ：R、G、Bをインタリーブ（BRGBRGBRGのように）
- ディスプレイ出力に対して最適

RGBプレーナフォーマットは、標準OSフレームバッファディスプレイフォーマットと互換性があります。

16 BPP（bits per pixel）：RGB565（32bitワード当たり2ピクセル）だけサポートされています。

24 BPP（パレットフォーマット）とグレースケールフォーマットは、サポートされません。ピクセルはラスタースキャン順、つまりピクセル行に対して上から下、ピクセル行内では左から右の順で保存されます。ピクセルコンポーネントはR（赤）、G（緑）、B（青）です。すべてのコンポーネントの空間分解能は同じ（4：4：4フォーマット）です。ピクセル単位でインタリーブされたコンポーネントと共に、フレームは単一パートで保存されます。

次の表に、データ保存方法を示します。

表412.RGBプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:27	26:21	20:16	15:11	10:5	4:0
0	赤n + 1	緑n + 1	青n + 1	赤n	緑n	青n
4	赤n + 4	緑n + 3	青n + 3	赤n + 2	緑n + 2	青n + 2

YCbCrフォーマット

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- インタリーブ：バッファ 1つ：Y、Cb、Crをインタリーブ（CbYCrYCbYCrのように）

ピクセルコンポーネントはY（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。各コンポーネントは8bitで符号化されます。輝度と色度は、次の表に示すように、（インタリーブして）一緒に保存されます。

表413.YCbCrプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 1	Cr n	Y n	Cb n
4	Y n + 3	Cr n + 2	Y n + 2	Cb n + 2

YCbCrフォーマット - Yのみ

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- バッファにはY情報のみが含まれる - モノクロ画像

ピクセルコンポーネントはY（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。このモードでは、色度情報がドロップされます。8bitでエンコードされた各ピクセルのルマコンポーネントのみが表414 に示すように保存されます。

結果として、元のYCbCrデータと同じ分解能を持つモノクロ画像が得られます。

表414.YCbCrプログレッシブビデオフォーマットでのデータ保存 - Y抽出モード

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 3	Y n + 2	Y n + 1	Y n
4	Y n + 7	Y n + 6	Y n + 5	Y n + 4

ハーフ解像度画像抽出

以前の受信モードを変更したもので、モノクロ、RGB、またはY抽出モードに適用可能です。

このモードは、ハーフ解像度の画像のみを保存するために使用します。OELSおよびLSM制御ビットを通して選択されます。

41.4 DCMI割込み

5種類の割込みが生成されます。すべての割込みは、ソフトウェアでマスク可能です。グローバル割込み（dcmi_it）は、個別割込みすべての論理和（OR）です。次の表に、すべての割込みの一覧を示します。

表415.DCMI割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
dcmi_it	ラインの終了	LINE_RIS	LINE_IE	LINE_ISCのセット	あり	なし
	フレームキャプチャ終了	FRAME_RIS	FRAME_IE	FRAME_ISCのセット	あり	なし
	データ受信のオーバーラン	OVR_RIS	OVR_IE	OVR_ISCのセット	あり	なし
	同期フレーム	VSYNC_RIS	VSYNC_IE	VSYNC_ISCのセット	あり	なし
	埋め込み同期フレーム検出におけるエラーの検出	ERR_RIS	ERR_IE	ERR_ISCのセット	あり	なし

41.5 DCMIレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。すべてのDCMIレジスタは、32bitワードとしてアクセスする必要があり、そうしないとバスエラーが発生します。

41.5.1 DCMI制御レジスタ（DCMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM[1:0]	
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ENABLE	Res.	Res.	EDM[1:0]		FCRC[1:0]		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **OELS**：奇数／偶数ライン選択（ライン選択開始）

このビットはLSMフィールドと連携して動作します（LSM = 1）。

0：インタフェースはフレーム開始後に最初のラインをキャプチャします（2番目のラインはドロップされる）。

1：インタフェースはフレーム開始から2番目のラインをキャプチャします（最初のラインはドロップされる）。

ビット19 **LSM**：ライン選択モード

0：インタフェースはすべての受信したラインをキャプチャします。

1：インタフェースは2つのラインのうちの1つをキャプチャします。

ビット18 **OEBS**：奇数／偶数バイト選択（バイト選択開始）

このビットはBSMフィールドと連携して動作します（BSM ≠ 00）。

0：インタフェースはフレーム／ライン開始から最初のデータ（バイトまたはダブルバイト）をキャプチャします（2番目のデータはドロップされる）。

1：インタフェースはフレーム／ライン開始から2番目のデータ（バイトまたはダブルバイト）をキャプチャします（最初のデータはドロップされる）。

ビット17:16 **BSM[1:0]**：バイト選択モード

00：インタフェースはすべての受信したデータをキャプチャします。

01：インタフェースは受信したデータから1つおきにバイトをキャプチャします。

10：インタフェースは4つのバイトのうちの1つをキャプチャします。

11：インタフェースは4つのバイトのうちの2つをキャプチャします。

注： このモードはEDM[1:0]=00でのみ動作します。その他すべてのEDM値に対し、このフィールドをリセット値にプログラムする必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **ENABLE**：DCMIイネーブル

0：DCMIIは無効です。

1：DCMIIは有効です。

注： このビットを有効にする前に、DCMI設定レジスタを正しくプログラムする必要があります。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはピクセルクロックごとに8bitデータをキャプチャします。

01：インタフェースはピクセルクロックごとに10bitデータをキャプチャします。

10：インタフェースはピクセルクロックごとに12bitデータをキャプチャします。

11：インタフェースはピクセルクロックごとに14bitデータをキャプチャします。

ビット9:8 **FCRC[1:0]**：フレームキャプチャレート制御

このビットによって、フレームキャプチャの頻度を規定します。これらのビットは、連続グラブモードでなければ意味を持ちません。スナップショットモードでは無視されます。

00：全フレームをキャプチャ。

01：フレームを1つおきにキャプチャ（帯域幅50 % 減少）

10：4フレームごとに1フレームをキャプチャ（帯域幅75 % 減少）

11：予約済み

ビット7 **VSPOL**：垂直同期極性

このビットは、データがパラレルインタフェース上で有効でないときのDCMI_VSYNCピン上でのレベルを示します。

0：DCMI_VSYNCはアクティブローです。

1：DCMI_VSYNCはアクティブハイです。

ビット6 **HSPOL**：水平同期極性

このビットは、データがパラレルインタフェース上で有効でないときのDCMI_HSYNCピン上でのレベルを示します。

0：DCMI_HSYNCはアクティブローです。

1：DCMI_HSYNCはアクティブハイです。

ビット5 **PCKPOL**：ピクセルクロック極性

このビットによってピクセルクロックのキャプチャエッジを設定します。

0：立ち下がりエッジでアクティブ

1：立ち上がりエッジでアクティブ

ビット4 **ESS**：埋め込み同期選択

0：ハードウェア同期。データキャプチャ（フレーム／ラインの開始／終了）はDCMI_HSYNC/DCMI_VSYNC信号と同期します。

1：埋め込み同期。データキャプチャはデータフローに埋め込まれた同期コードと同期します。

注： 8bitのパラレルデータに対してのみ有効です。ESSビットがセットされると、HSPOL/VSPOLは無視されます。

このビットは、JPEGモードでは無効になります。

ビット3 **JPEG**：JPEGフォーマット

0：非圧縮ビデオフォーマット

1：このビットはJPEGデータ転送に使用します。DCMI_HSYNC信号はデータイネーブルとして使用します。このモードでは、クロップおよび埋め込み同期機能（ESSビット）は使用できません。

ビット2 **CROP**：クロップ機能

0：画像全体がキャプチャされます。この場合、1枚の画像フレーム内の総バイト数は4の倍数でなければなりません。

1：クロップレジスタで指定されたウィンドウ内のデータのみがキャプチャされます。クロップウィンドウのサイズがピクチャサイズを超える場合は、そのピクチャサイズのみがキャプチャされます。

ビット1 **CM**：キャプチャモード

0：連続グラブモード - 受信したデータはDMAを通じて転送先メモリに転送されます。バッファの場所やモード（リニアバッファまたはサーキュラバッファ）は、システムDMAを通じて制御されます。

1：スナップショットモード（単一フレーム）- アクティブになると、インタフェースはフレーム開始を待ってからDMAを通じて単一フレームを転送します。フレーム終了時、CAPTUREビットは、自動的にリセットされます。

ビット0 **CAPTURE**：キャプチャイネーブル

0：キャプチャは無効です。

1：キャプチャは有効です。

カメラインタフェースは最初のフレーム開始を待ち、その後DMAリクエストが生成されて受信したデータが転送先メモリに転送されます。

スナップショットモードでは、受信した最初のフレームの終了時にCAPTUREビットが自動的にクリアされます。

連続グラブモードにおいて、キャプチャ進行中にソフトウエアがこのビットをクリアすると、そのビットは実質的にフレーム終了時にクリアされます。

注： DMAコントローラとすべてのDCMI設定レジスタを正しくプログラムしてから、このビットを有効にする必要があります。

41.5.2 DCMIステータスレジスタ（DCMI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FNE	VSYNC	HSYNC
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FNE**：FIFOノットエンプティ

このビットは、FIFOのステータスを示します。

1：FIFOに有効なデータあり。

0：FIFOエンプティ

ビット1 **VSYNC**：垂直同期

このビットは、極性が正しくプログラムされたDCMI_VSYNCピンの状態を示します。埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブフレーム

1：フレーム間同期

埋め込み同期の場合、このビットはDCMI_CRのCAPTUREビットがセットされている場合にのみ意味を持ちます。

ビット0 **HSYNC**：水平同期

このビットは、極性が正しくプログラムされたDCMI_HSYNCピンの状態を示します。埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブライン

1：ライン間同期

埋め込み同期の場合、このビットはDCMI_CRのCAPTUREビットがセットされている場合にのみ意味を持ちます。

41.5.3 DCMI raw割込みステータスレジスタ（DCMI_RIS）

DCMI_RISはraw割込みステータスを示し、読出し専用でアクセスできます。読出し時、このレジスタはDCMI_IERレジスタ値でマスクする前の関連する割込みのステータスを返します。

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
											r	r	r	r	r

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **LINE_RIS**：ラインraw割込みステータス

DCMI_HSYNC信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。ラインが有効でなくてもハイになります。

埋め込み同期の場合、このビットはDCMI_CRのCAPTUREビットがセットされている場合にのみセットされます。

DCMI_ICRレジスタのLINE_ISCビットをセットすることで、クリアされます。

ビット3 **VSYNC_RIS**：DCMI_VSYNC raw割込みステータス

DCMI_VSYNC信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。

埋め込み同期の場合、このビットはDCMI_CRのCAPTUREビットがセットされている場合にのみセットされます。

DCMI_ICRレジスタのVSYNC_ISCビットをセットすることで、クリアされます。

ビット2 **ERR_RIS**：同期エラー raw割込みステータス

0：同期エラーは検出されていません。

1：埋め込み同期文字が正しい順番で受信されません。

このビットは、埋め込み同期モードでのみ有効です。DCMI_ICRレジスタのERR_ISCビットをセットすることで、クリアされます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット1 **OVR_RIS**：オーバーランraw割込みステータス

0：データバッファオーバーランは発生していません。

1：データバッファオーバーランが発生し、データFIFOが不正です。

DCMI_ICRレジスタのOVR_ISCビットをセットすることで、このビットはクリアされます。

ビット0 **FRAME_RIS**：キャプチャ完了raw割込みステータス

0：新規キャプチャはありません。

1：フレームキャプチャ完了しています。

このビットは、フレームやウィンドウがキャプチャされるとセットされます。

クロップされたウィンドウの場合、このビットは、クロップの最後のラインの終了時にセットされます。キャプチャされたフレームが空でもセットされます（例：フレーム外でクロップされたウィンドウなど）。

DCMI_ICRレジスタのFRAME_ISCビットをセットすることで、このビットはクリアされます。

41.5.4 DCMI割込み有効レジスタ（DCMI_IER）

DCMI_IERレジスタは、割込みを有効にするために使用します。DCMI_IERビットの1つがセットされると、該当する割込みが有効になります。このレジスタは、読出しと書込みの両方でアクセスできます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_IE	VSYNC_IE	ERR_IE	OVR_IE	FRAME_IE
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **LINE_IE**：ライン割込みイネーブル

0：ライン受信時に割込みは生成されません。
1：ラインすべてを受信したときに割込みが生成されます。

ビット3 **VSYNC_IE**：DCMI_VSYNC割込みイネーブル

0：割込みは生成されません。
1：DCMI_VSYNCがインアクティブからアクティブ状態に遷移するたびに割込みが生成されます。
DCMI_VSYNC信号のアクティブ状態は、VSPOLビットによって定義されます。

ビット2 **ERR_IE**：同期エラー割込みイネーブル

0：割込みは生成されません。
1：埋め込み同期コードが正しい順番で受信されない場合に割込みが生成されます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット1 **OVR_IE**：オーバーラン割込みイネーブル

0：割込みは生成されません。
1：DMAが新しいデータ（32bit）を受信する前に最後のデータを転送できなかった場合に割込みが生成されます。

ビット0 **FRAME_IE**：キャプチャ完了割込みイネーブル

0：割込みは生成されません。
1:受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に割込みが生成されます。

41.5.5 DCMIマスク割込みステータスレジスタ（DCMI_MIS）

DCMI_MISレジスタは、読出し専用レジスタです。読出し時、該当する割込みの現在のマスクステータス値（DCMI_IERの値によって変わります）を返します。DCMI_IER内の該当するイネーブル･ビットがセットされ、DCMI_RIS内の該当ビットがセットされると、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
											r	r	r	r	r

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **LINE_MIS**：マスクされたライン割込みステータス

このビットは、マスクされたライン割込みのステータスを示します。

0：ライン受信時に割込みは生成されません。

1：ラインが完全に受信され、DCMI_IERのLINE_IEビットがセットされていると割込みが生成されます。

ビット3 **VSYNC_MIS**：マスクされたVSYNC割込みステータス

このビットは、マスクされたVSYNC割込みのステータスを示します。

0：DCMI_VSYNCの遷移で割込みは生成されません。

1：DCMI_VSYNCがインアクティブからアクティブ状態に遷移するたびに、DCMI_IERのVSYNC_IEビットがセットされていると割込みが生成されます。

DCMI_VSYNC信号のアクティブ状態は、VSPOLビットによって定義されます。

ビット2 **ERR_MIS**：マスクされた同期エラー割込みステータス

このビットは、マスクされた同期エラー割込みのステータスを示します。

0：同期エラー時、割込みは生成されません。

1：埋め込み同期コードが正しい順番で受信されず、DCMI_IERのERR_IEビットがセットされていると割込みが生成されます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット1 **OVR_MIS**：マスクされたオーバーラン割込みステータス

このビットは、マスクされたオーバ-フロー割込みのステータスを示します。

0：オーバーラン時、割込みは生成されません。

1：DMAが新しいデータ（32bit）を受信する前に最後のデータを転送できず、DCMI_IERのOVR_IEビットがセットされていると割込みが生成されます。

ビット0 **FRAME_MIS**：マスクされたキャプチャ完了割込みステータス

このビットは、マスクされたキャプチャ完了割込みのステータスを示します。

0：キャプチャ完了で割込みは生成されません。

1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時にDCMI_IERのFRAME_IEビットがセットされていると割込みが生成されます。

41.5.6 DCMI割込みクリアレジスタ（DCMI_ICR）

DCMI_ICRレジスタは、書込み専用です。このレジスタのビットをセットすることによってDCMI_RISおよびDCMI_MISレジスタの対応するフラグをクリアします。0を書き込んでも、ビットの値は変化しません。

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
											w	w	w	w	w

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **LINE_ISC**：ライン割込みステータスクリア
このビットをセットするとDCMI_RISレジスタのLINE_RISフラグがクリアされます。

ビット3 **VSYNC_ISC**：垂直同期割込みステータスクリア
このビットをセットするとDCMI_RISレジスタのVSYNC_RISフラグがクリアされます。

ビット2 **ERR_ISC**：同期エラー割込みステータスクリア
このビットをセットするとDCMI_RISレジスタのERR_RISフラグがクリアされます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット1 **OVR_ISC**：オーバーラン割込みステータスクリア
このビットをセットするとDCMI_RISレジスタのOVR_RISフラグがクリアされます。

ビット0 **FRAME_ISC**：キャプチャ完了割込みステータスクリア
このビットをセットするとDCMI_RISレジスタのFRAME_RISフラグがクリアされます。

41.5.7 DCMI埋め込み同期コードレジスタ（DCMI_ESCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FEC[7:0]								LEC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LSC[7:0]								FSC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **FEC[7:0]**：フレーム終了デリミタコード

このバイトによって、フレーム終了デリミタのコードを指定します。このコードは0xFF、0x00、0x00、FECの形式で4バイトで構成されています。

FECが0xFFにプログラムされている場合、使用されていないすべてのコード（0xFF0000XY）はフレーム終了デリミタとして解釈されます。

ビット23:16 **LEC[7:0]**：ライン終了デリミタコード

このバイトによって、ライン終了デリミタのコードを指定します。このコードは0xFF、0x00、0x00、LECの形式で4バイトで構成されています。

ビット15:8 **LSC[7:0]**：ライン開始デリミタコード

このバイトによって、ライン開始デリミタのコードを指定します。このコードは0xFF、0x00、0x00、LSCの形式で4バイトで構成されています。

ビット7:0 **FSC[7:0]**：フレーム開始デリミタコード

このバイトによって、フレーム開始デリミタのコードを指定します。このコードは0xFF、0x00、0x00、FSCの形式で4バイトで構成されています。

FSCが0xFFにプログラムされている場合、フレーム開始デリミタは検出されません。FECコードの後に現れる最初のLSCは、フレームの開始デリミタとして解釈されます。

41.5.8 DCMI埋め込み同期マスク解除レジスタ（DCMI_ESUR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FEU[7:0]								LEU[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LSU[7:0]								FSU[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **FEU[7:0]**：フレーム終了デリミタマスク解除

このバイトによって、フレーム終了デリミタのコードに適用するマスクを指定します。

0:受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCRのFECバイトの該当ビットがマスクされます。

1:受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCRのFECバイトの該当ビットが比較されます。

ビット23:16 **LEU[7:0]**：ライン終了デリミタマスク解除

このバイトによって、ライン終了デリミタのコードに適用するマスクを指定します。

0：受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCRのLECバイトの該当ビットがマスクされます。

1：受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCRのLECバイトの該当ビットが比較されます。

ビット15:8 **LSU[7:0]**：ライン開始デリミタマスク解除

このバイトによって、ライン開始デリミタのコードに適用するマスクを指定します。

0：受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCRのLSCバイトの該当ビットがマスクされます。

1：受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCRのLSCバイトの該当ビットが比較されます。

ビット7:0 **FSU[7:0]**：フレーム開始デリミタマスク解除

このバイトによって、フレーム開始デリミタのコードに適用するマスクを指定します。

0:受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCRのFSCバイトの該当ビットがマスクされます。

1:受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCRのFSCバイトの該当ビットが比較されます。

41.5.9 DCMIクロップウィンドウ開始（DCMI_CWSTRT）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	VST[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	HOFFCNT[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:16 **VST[12:0]**：垂直方向開始ラインカウント

このライン番号で画像キャプチャを開始します。これより前のラインデータは無視されます。

0x0000：ライン1

0x0001：ライン2

0x0002：ライン3

....

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **HOFFCNT[13:0]**：水平方向オフセットカウント

キャプチャ開始前にカウントするピクセルクロック数を示します。

41.5.10 DCMIクロップウィンドウサイズ（DCMI_CWSIZE）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	VLINE[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CAPCNT[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:16 **VLINE[13:0]**：垂直方向ラインカウント

開始点からのキャプチャライン数を示します。

0x0000：1ライン

0x0001：2ライン

0x0002：3ライン

....

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **CAPCNT[13:0]**：キャプチャカウント

同一ライン上でキャプチャする開始点からのピクセルクロック数を示します。この値は、パラレルインタフェースのさまざまなデータ幅に対してワードアライメントが施されたデータに対応する必要があります。

0x0000：1ピクセル

0x0001：2ピクセル

0x0002：3ピクセル

....

41.5.11 DCMIデータレジスタ（DCMI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0000

このデジタルカメラインタフェースは、DMA転送を要求する前に受信したデータをすべて32bit形式にまとめます。DMA転送に十分な時間を与え、DMAオーバーランを回避するため、深さ 8ワードのFIFOが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BYTE3 [7:0]								BYTE2 [7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYTE1 [7:0]								BYTE0 [7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 **BYTE3[7:0]**：データバイト3

ビット23:16 **BYTE2[7:0]**：データバイト2

ビット15:8 **BYTE1[7:0]**：データバイト1

ビット7:0 **BYTE0[7:0]**：データバイト0

41.5.12 DCMIレジスタマップ

表416.DCMIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DCMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM[1:0]		Res.	ENABLE	Res.	Res.	EDM[1:0]		FCRC[1:0]		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	リセット値												0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0	0	0
0x04	DCMI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FNE	VSYNC	HSYNC
	リセット値																														0	0	0
0x08	DCMI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
	リセット値																												0	0	0	0	0
0x0C	DCMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_IE	VSYNC_IE	ERR_IE	OVR_IE	FRAME_IE
	リセット値																												0	0	0	0	0
0x10	DCMI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
	リセット値																												0	0	0	0	0
0x14	DCMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
	リセット値																												0	0	0	0	0
0x18	DCMI_ESCR	FEC[7:0]								LEC[7:0]								LSC[7:0]								FSC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	DCMI_ESUR	FEU[7:0]								LEU[7:0]								LSU[7:0]								FSU[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DCMI_CWSTRT	Res.	Res.	Res.	VST[12:0]													Res.	Res.	HOFFCNT[13:0]													
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	DCMI_CWSIZE	Res.	Res.	VLINE[13:0]														Res.	Res.	CAPCNT[13:0]													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	DCMI_DR	BYTE3 [7:0]								BYTE2 [7:0]								BYTE1 [7:0]								BYTE0 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

42 パラレル同期スレーブインタフェース（PSSI）

PSSIペリフェラルとDCMI（デジタルカメラインタフェース）は同じ回路を使用します。その結果、これらの2つのペリフェラルを同時に使用することはできません。PSSIを使用する場合、DCMIレジスタにはアクセスできません。その逆も同様です。

さらに、PSSIとDCMIは同じオルタネート機能と割込みベクタを共有します（セクション 42.3.2 : PSSIピンおよび内部信号を参照）。

42.1 はじめに

PSSIは、汎用の同期型8/16ビットパラレルデータ入出力スレーブインタフェースです。これにより、データが有効になった時を示すデータ有効信号をトランスミッタが送信し、データをサンプリングする準備ができた時を示すフロー制御信号をレシーバが出力することができます。

42.2 PSSIの主な機能

PSSIペリフェラルの主な機能は以下のとおりです。

- スレーブモード動作
- 8ビットまたは16ビットのパラレルデータ入力または出力
- 8ワード（32バイト）FIFO
- データイネーブル（PSSI_DE）オルタネート機能入力およびレディ（PSSI_RDY）オルタネート機能出力

  これらを選択すると、これらの信号によって、トランスミッタはデータが有効であること、レシーバはデータをサンプリングする準備ができたことを示すことができます。あるいはそのどちらも可能です。

42.3 PSSIの機能説明

PSSIは、高速のデータフローを送受信できる同期型パラレルスレーブインタフェースです。最大16本のデータライン（PSSI_D[15:0]）と1本のクロックライン（PSSI_PDCK）で構成されています。クロックの極性は、データがクロックの立上がりエッジまたは立下がりエッジのどちらでもキャプチャまたは送信されるように設定できます。

通常、汎用DMAチャネルは、データレジスタ（PSSI_DR）を介して32ビットのパックデータを渡すために使用されます。

データフローは、連続、またはオプションのPSSI_DE（データイネーブル）およびPSSI_RDY（レディ）信号を使用してハードウェアによって同期されます。

図394にPSSIブロック図を示します。

42.3.1 PSSIブロック図

図394. PSSIブロック図

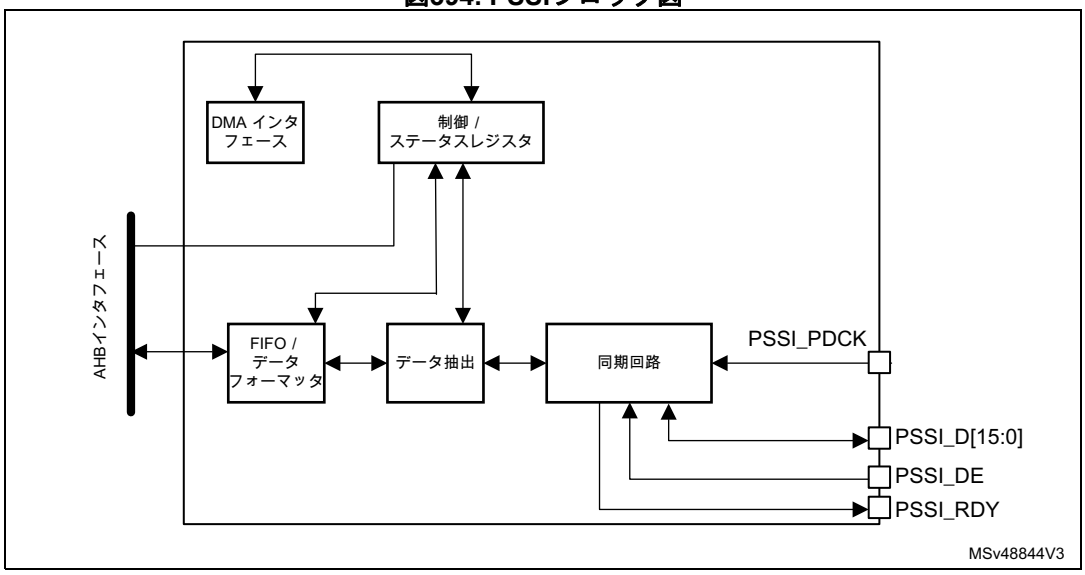
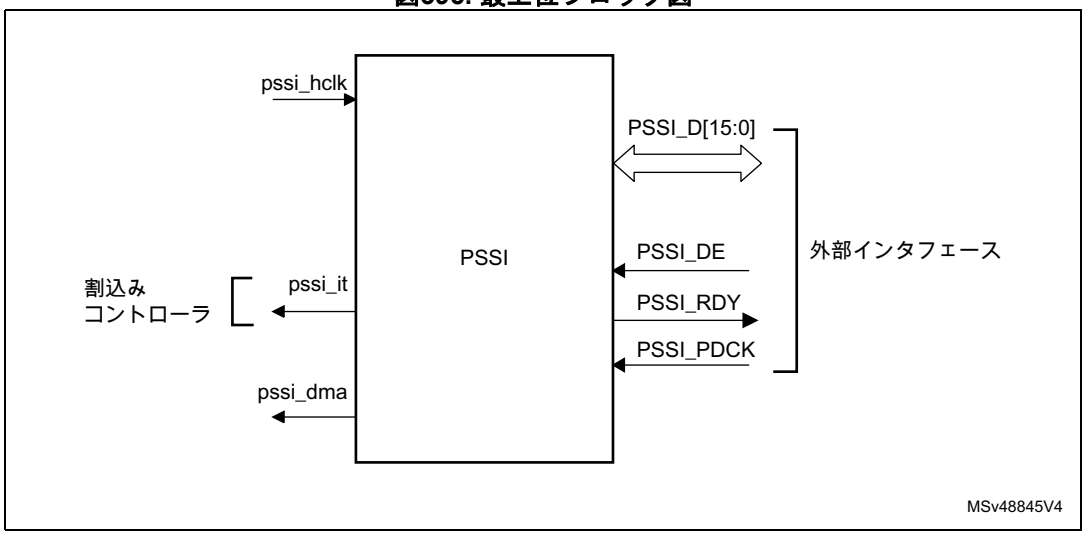


図395. 最上位ブロック図



42.3.2 PSSIピンおよび内部信号

PSSIインタフェースは19本のピンで構成されていますが、パラレルデータを転送するには9本の信号で十分です。表417にPSSIピンを示します。

PSSI ENABLEビット（PSSI_CRのビット14）が1にセットされていると、オルタネート機能と割込みベクタはPSSIに関連付けられます。そうでない場合、それらはDCMIに関連付けられます。DCMI ENABLEビット（DCMI_CRのビット15）とPSSI ENABLEビット（PSSI_CRのビット14）を同時に1にセットしないでください。たとえば、オルタネート機能PSSI_PDCK/DCMI_PIXCKを使用するようにGPIOが設定されている場合、PSSI_CR/ENABLEが1にセットされていれば、アクティブになるのはPSSI_PDCK機能です。

表417.PSSIの入出力ピン

PSSI信号名	共有されるDCMI信号	信号タイプ	説明
PSSI_PDCK	DCMI_PIXCK	入力	パラレルデータのクロック入力
PSSI_D[15:0]	DCMI_D[13:0]	入力／出力	送信時はデータ出力、受信時はデータ入力
PSSI_DE	DCMI_HSYNC	入力	データイネーブル信号：受信時のデータ有効信号または送信時のフロー制御信号
PSSI_RDY	DCMI_VSYNC	出力	レディ信号：受信時のフロー制御信号または送信時のデータ有効信号

表418 に、PSSIの内部入力／出力信号を示します。

表418.PSSI内部入力／出力信号

内部信号名	信号タイプ	説明
pssi_it	出力	割込み
pssi_dma	出力	DMAリクエスト
pssi_hclk	入力	AHBクロック

42.3.3 PSSI クロック

AHBクロック周波数はPSSI_PDCK周波数よりも少なくとも2.5倍以上高くなければなりません。周波数比が2.5未満の場合、データが転送中に破損または失われる可能性があります。

データ転送はPSSI_PDCKと同期しています。PSSI_PDCKの極性は、CKPOLビット（PSSI_CRのビット5）を介して、次のように設定できます。

- CKPOL = 0の場合
  - 入力ピンはPSSI_PDCKの立下がりエッジでサンプリングされます。
  - 出力ピンはPSSI_PDCKの立上がりエッジで駆動されます。
- CKPOL = 1の場合
  - 入力ピンはPSSI_PDCKの立上がりエッジでサンプリングされます。
  - 出力ピンはPSSI_PDCKの立下がりエッジで駆動されます。

42.3.4 PSSIデータ管理

データの方向

データ転送の方向は、OUTEN制御ビット（PSSI_CRのビット31）を介して次のように設定されます。

- OUTENを0にクリアすると（デフォルト設定）、PSSIは受信モードで動作し、データピンにデータが入力されます。
- OUTENを1にセットすると、ペリフェラル は送信モードで動作し、データピンにデータが出力されます。

OUTENは、ENABLEビットが0にクリアされている場合にのみ変更できます。

データレジスタおよびDMA

データは、PSSI_DRデータレジスタを使用してFIFOとの間で以下のように転送されます。

- 受信モードでは、PSSI_DRを読み取ることによってFIFOからデータを読み出す必要があります。
- 送信モードでは、PSSI_DRに書き込むことによってFIFOにデータを書き込む必要があります。

PSSI_DRへのワード（32ビット）アクセスおよびPSSI_DR[15:0] へのハーフワード（16ビット）アクセスは、すべてのモードで許可されます。PSSI_DR[7 : 0] へのバイト（8ビット）アクセスは、PSSIが一度に8ビットを転送するように設定されている場合（PSSI_CRレジスタのEDM = 00）にのみ許可されます。

CPUの負荷を軽減するには、DMAを使用してPSSI FIFOとの間でデータを転送することを推奨します。これを使用する場合、PSSI_DRレジスタを介してデータを転送するようにDMAを設定する必要があります。32ビット転送を使用すると、帯域幅が最適化され、バス負荷が軽減されます。ただし、8ビットおよび16ビットの転送も許可されています。

DMAを使用するには、PSSI DMAイネーブル·ビット（PSSI_CRのDMAEN）を1（デフォルト設定）にセットします。DMAENが1にセットされている場合、FIFOで32ビット転送の準備（受信モードで4つの有効バイト、または送信モードで4つの空バイト）ができると、DMA転送が開始されます。したがって、受信モードでは、DMAが8ビット転送を実行するように設定されている場合でも、FIFOに3バイト以下しかない場合、DMA転送は開始されません。

RTT4BおよびRTT1Bステータスビット（PSSI_SR）は、CPUがFIFOとの間で直接転送を実行するときに役立ちます。RTT4Bが1にセットされると、FIFOが4バイトを転送する準備ができていることになります。つまり、受信モードのFIFOで少なくとも4つの値は有効なバイト、または送信モードで少なくとも4つの値は空きバイトがある状態になります。RTT1Bが1にセットされると、FIFOが1バイトを転送する準備ができていることになります。つまり、受信モードのFIFOで少なくとも1つの有効なバイト、または送信モードで少なくとも1つの空きバイトがある状態になります。

8bitデータ

PSSIパラレルインタフェースは、EDM[1:0] 制御ビット（PSSI_CRのビット11:10）に応じて、8ビットのデータ（D[7:0] を使用）または16ビットのデータ（D[15:0] を使用）を転送できます。8ビットの設定を選択（EDM[1:0] を00にセット）した場合、使っていないD[15:0] ピンをGPIOまたは他の機能に使用できます。

PSSI_CRのEDM[1:0] が00にプログラムされている場合、インタフェースはD[7:0] ピンを使用して8ビットを転送します。この場合、D[15:8] は使用されず、32ビットワードを転送するには4サイクルのPSSI_PDCKが必要です。

最下位バイト（ビット7:0）は転送された最初のバイトに対応し、最上位バイト（ビット31:24）は転送された4番目のバイトに対応します。表419 に、2つの32ビットワード内のデータバイトの配置を示します。

表419.32bitワード（8bit幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

16bitデータ

PSSI_CRのEDM[1:0] が11にプログラムされている場合、インタフェースはD[15:0] ピンを使用して16ビットを転送します。この場合、32ビットワードを転送するには2サイクルのPSSI_PDCKが必要です。

最下位のハーフワード（ビット15:0）は転送された最初のハーフワードに対応し、最上位のハーフワード（ビット31:16）は転送された2番目のハーフワードに対応します。表420 に、2つの32ビットワード内のデータの配置を示します。

表420.32bitワード（16bit幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:16	15:0
0	D_{n+1}[15:0]	D_n[15:0]
4	D_{n+3}[15:0]	D_{n+2}[15:0]

FIFOデータバッファとエラー状態

8 ワードのFIFOは、パフォーマンスの向上に役立ち、オーバーランとアンダーランを回避します。

受信モードでレディ信号（PSSI_RDY）が無効になっている場合、FIFOがフルのときにクロックのアクティブエッジが発生すると、オーバーランエラーが生成されます。この場合、入力データは失われます。

送信モードでデータイネーブル信号（PSSI_DE）が無効になっている場合、FIFOが空のときにクロックのアクティブエッジが発生すると、アンダーランエラーが生成されます。この場合、予測不能なデータが出力されます。

OVR_RISステータスビットは、オーバーランまたはアンダーランが発生したことを示します。これらのイベントが発生した場合、割込みを生成することができます。

42.3.5 PSSIオプション制御信号

データイネーブル（PSSI_DE）オルタネート機能入力

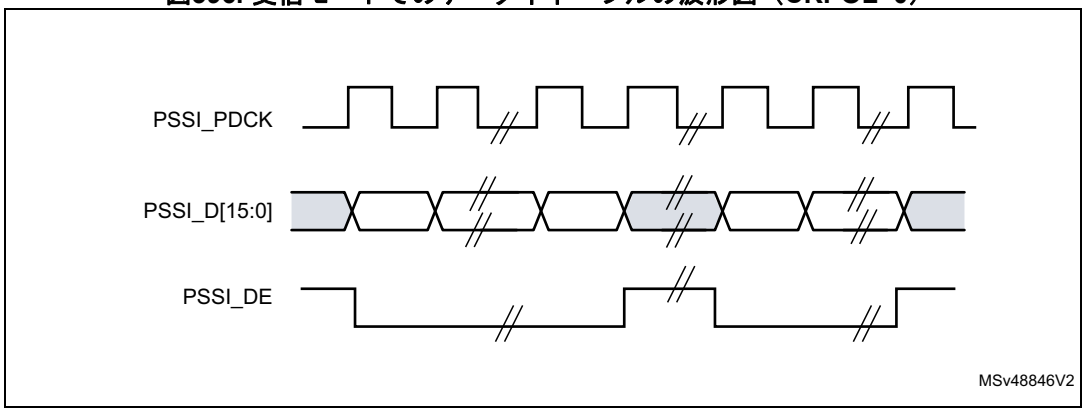
データイネーブル信号PSSI_DEはオプション信号です。それは、現在のサイクル中にデータの転送が有効になっていることを示すために、データ転送元／トランスミッタによって駆動されます。PSSI_DEが非アクティブの場合、受信側は次のクロックエッジでデータをサンプリングしてはならないことを意味します。

このオルタネート機能信号は、DERDYCFG（PSSI_CRのビット20:18）制御ビットを使用して有効にできます。PSSI_DEの極性は、DEPOL制御ビット（PSSI_CRのビット6）を介して設定されます。DEPOLを0にクリアするとPSSI_DEはアクティブローになり、DEPOLを1にセットするとアクティブハイになります。

PSSI_DE信号の方向は、OUTENの値によって定義されます。それはデータの方向と同じです。

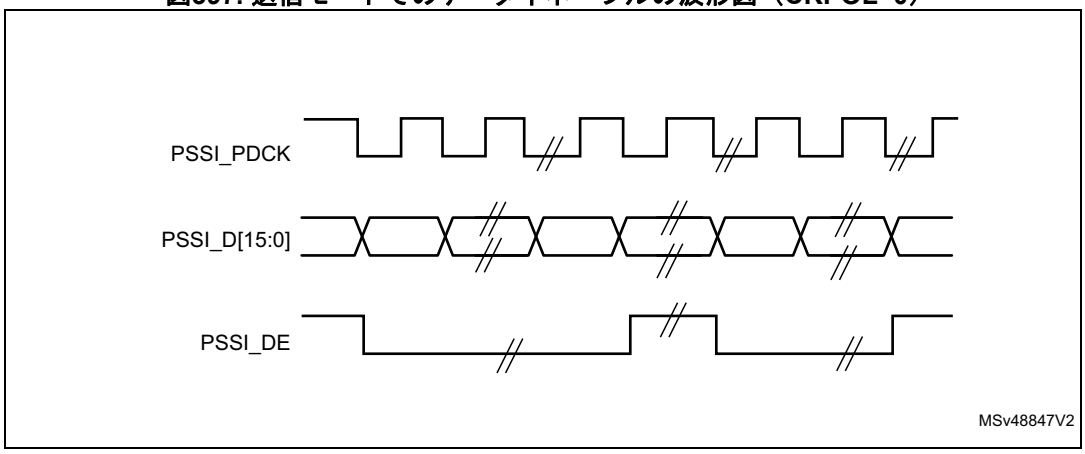
PSSI_DEオルタネート機能入力が（DERDYCFGを介して）受信モード（OUTENが0にクリアされている状態）で有効になっている場合、PSSIはデータ（D[15:0]）のサンプリングに使用されるものと同じPSSI_PDCKエッジでPSSI_DEをサンプリングします。PSSI_DEがアクティブな場合、サンプリングされたデータはFIFOに保存されます。そうでない場合、サンプリングされたデータは無効と見なされ、破棄されます。送信側デバイスは、PSSI_DEをデータ有効信号として使用し、現在のサイクルのデータが無効な場合にPSSI_DEを非アクティブにできます。このフロー制御機能により、アンダーランエラーを回避できます。

図396. 受信モードでのデータイネーブルの波形図（CKPOL=0）



PSSI_DEオルタネート出力機能が（DERDYCFGを介して）送信モード（OUTEN=1）で有効になっている場合、PSSIはデータ（D[15:0]）の駆動に使用されるものと同じPSSI_PDCKエッジでPSSI_DEを駆動します。新しい8または16ビットデータ（PSSI_CRのEDM[1:0] 制御ビットでプログラムされたもの）が内部FIFOでの送信に利用できる場合、このデータはデータ出力（D[15:0]）に出力され、そして、PSSI_DE出力は現在のPSSI_PDCKエッジでアクティブになります。そうでない場合（TX FIFOが空の場合）、D[15:0] 出力は次のクロックエッジで変更されないままで、PSSI_DE出力は非アクティブになります。

図397. 送信モードでのデータイネーブルの波形図（CKPOL=0）



レディ（PSSI_RDY）オルタネート機能出力

レディ信号PSSI_RDYはオプション信号です。それは、受信デバイスによって駆動され、現在のサイクル中にデータを受け入れられるかどうかを示します。PSSI_RDYが非アクティブの場合、受信側は次のクロックエッジでデータをサンプリングしてはならないことを意味します。

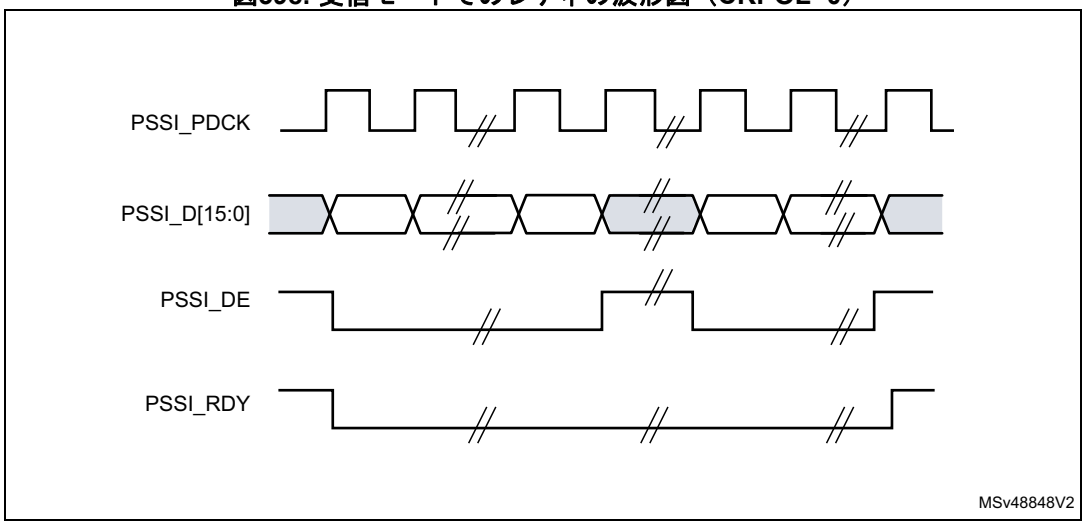
このオルタネート機能信号は、DERDYCFG制御ビット（PSSI_CRのビット20:18）を使用して有効にできます。PSSI_RDYの極性は、RDYPOL制御ビット（PSSI_CRのビット6）を介して設定されます。RDYPOLを0にクリアするとPSSI_RDYはアクティブローになり、RDYPOLを1にセットするとアクティブハイになります。

PSSI_RDY信号の方向は、OUTEN（PSSI_CRのビット31）によって定義されます。PSSI_DEおよびデータ信号とは逆の方向に設定されます。

PSSI_RDYオルタネート出力機能が（DERDYCFGを介して）受信モード（OUTEN=0）で有効になっている場合、PSSI はデータ（D[15:0]）をサンプリングした後、PSSI_PDCK の半サイクル後にPSSI_RDYを駆動します。FIFOにさらにデータを受信するのに十分な空きスペースがある場合、PSSI

はPSSI_RDY信号をアクティブにします。そうではなく、FIFOがフルでさらにデータを受信することができない場合、PSSIはPSSI_RDY信号を非アクティブにします。送信側デバイスは、PSSI_RDYが非アクティブであることを検出すると、次のサイクルで現在のデータを繰り返す必要があります。このフロー制御機能により、システムが（DMAを介して）データフローに対応できない場合に、PSSIはオーバーランエラーを回避できます。

図398. 受信モードでのレディの波形図（CKPOL=0）



PSSI_RDYオルタネート入力機能が（DERDYCFGを介して）送信モード（OUTEN=1）で有効になっている場合、PSSIはデータ（D[15:0]）が駆動されるのと逆のPSSI_PDCKのエッジでPSSI_RDY信号をサンプリングします。PSSI_RDY信号が非アクティブの場合、PSSIは同じデータ（D[15:0]）を保持し、次のPSSI_PDCKクロックサイクル中に有効なデータが利用可能であることがPSSI_DEによって通知されます。そうではなく、PSSI_RDY信号がアクティブとしてサンプリングされると、TX FIFO（利用可能な場合）からの次のデータがデータ出力（D[15:0]）に出力されます。TX FIFOに新しいデータがない場合、PSSIはデータ出力値を保持し、PSSI_DE信号を（有効な場合）非アクティブとして出力します。

受信デバイスはPSSI_RDYを使用してデータフローを制御し、システムが（DMAを介して）データフローに対応できない場合に、オーバーランエラーを回避できます。

双方向PSSI_DE/PSSI_RDY信号

PSSI_CRレジスタでDEPOLとRDYPOLの両方が1にセットされ、DERDYCFGが111または100にセットされている場合、1つのピンをデータイネーブル（PSSI_DE）機能とレディ（PSSI_RDY）機能の両方に使用できます。この場合、選択したオルタネート機能（DERDYCFG = 111の場合はPSSI_DE、またはDERDYCFG = 100の場合はPSSI_RDY）に対応するGPIOをオープンドレインとして設定する必要があります。他方のデバイスも、信号線をオープンドレインとして駆動するように設定する必要があり、弱いプルアップを信号線に適用する必要があります。

こうして、信号は双方向になります。送信側が信号線をローに駆動する（データが無効であることを示す）か、受信側が信号線をローに駆動する（現在のデータをサンプリングしていないことを示す）場合、両方のデバイスは現在のサイクルでデータが転送されていないことを知ります。

図399. 双方向PSSI_DE/PSSI_RDYの波形

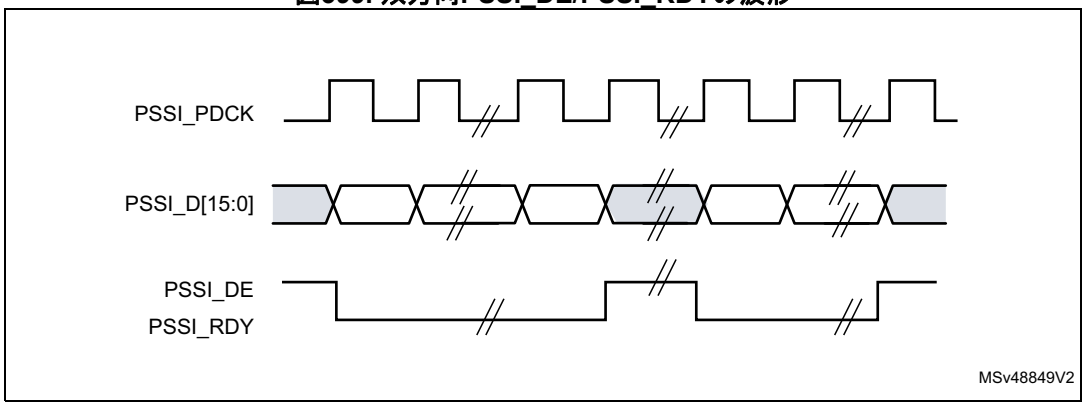
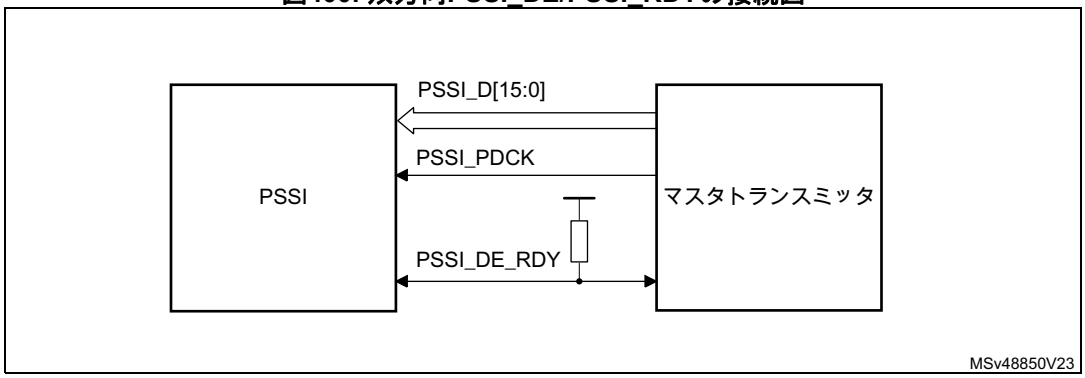


図400. 双方向PSSI_DE/PSSI_RDYの接続図



42.4 PSSI割込み

PSSIは割込みを1つだけ（IT_OVR）生成します。したがって、グローバル割込み（pssi_it）と同等です。割込みのリストについては、表421 を参照してください。

PSSIとDCMIは同じ割込みベクタを共有します。PSSI ENABLEビット（PSSI_CRのビット14）が1にセットされていると、PSSIによってこれらの割込みがトリガされます。そうでない場合、それらはDCMIによって制御されます。

DCMI ENABLEビット（DCMI_CRのビット14）とPSSI ENABLEビットを同時に1にセットしないでください。

表421.PSSI割込みリクエスト

項目（割込みの略称）	DCMIと共用	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	低消費電力モードの終了
IT_OVR	IT_OVR	受信モードでのオーバーランまたは送信モードでのアンダーランを示す	OVR_RIS	OVR_IE	OVR_ISC	NA

42.5 PSSIレジスタ

PSSI_DR以外のPSSIレジスタへ8bitの書込みまたは16bitの書込み操作を行うと、バスエラーになります。32ビットの読出しおよび書込み操作が許可されています。

42.5.1 PSSI制御レジスタ（PSSI_CR）

アドレスオフセット：0x00

リセット値：0x4000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OUTEN	DMAEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DERDYCFG[2:0]			Res.	Res.
rw	rw										rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ENABLE	Res.	Res.	EDM[1:0]		Res.	RDYPOL	Res.	DEPOL	CKPOL	Res.	Res.	Res.	Res.	Res.
	rw			rw	rw		rw		rw	rw					

ビット31 **OUTEN**：データ方向選択ビット

0：受信モード：データはPSSI_PDCKと同期して入力されます。

1：送信モード：データはPSSI_PDCKと同期して出力されます。

ビット30 **DMAEN**：DMAイネーブル·ビット

0：DMA転送は無効になります。DMA転送が無効になっているとき、ユーザアプリケーションはPSSI_DRレジスタに直接アクセスできます。

1：DMA転送は有効です（デフォルトの設定）。汎用DMAコントローラのDMAチャネルは、PSSI_DRとの間の転送を実行するように設定する必要があります。

ビット 29:21 予約済みであり、リセット値に保持する必要があります。

ビット20:18 **DERDYCFG[2:0]**：データイネーブルおよびレディの設定

000：PSSI_DEおよびPSSI_RDYの両方無効

001：PSSI_RDYのみ有効

010：PSSI_DEのみ有効

011：PSSI_RDYおよびPSSI_DEオルタネート機能の両方有効

100：PSSI_RDYおよびPSSI_DE機能の両方有効 - PSSI_RDYピンで双方向（1683ページの双方向PSSI_DE/PSSI_RDY信号を参照）

101：PSSI_RDY機能のみ有効ですが、PSSI_DEピンに配置されます。

110：PSSI_DE機能のみ有効ですが、PSSI_RDYピンに配置されます。

111：PSSI_RDYおよびPSSI_DE機能の両方有効 - PSSI_DEピンで双方向（1683ページの双方向PSSI_DE/PSSI_RDY信号を参照）

PSSI_RDY機能がPSSI_DEピンに配置されている場合でも（設定101または111）、極性を決定するのはRDYPOLビットです。同様に、PSSI_DE機能がPSSI_RDYピンに配置されている場合でも（設定110または111）、極性を決定するのはDEPOLビットです。

ビット17:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **ENABLE**：PSSIイネーブル

0：PSSIは無効です。

1：PSSIは有効です。

ENABLEが0にクリアされると、FIFOの内容は一掃されます。

注： ENABLE = 1のとき、ENABLEビット自体を除いて、PSSI_CRの内容を変更しないでください。ENABLEが0から1に変わるとすぐに、すべての設定ビットが変化する可能性があります。

DMAコントローラとすべてのPSSI設定レジスタを正しくプログラムしてから、ENABLEビットを1にセットする必要があります。

ENABLEビットとDCMI ENABLEビット（DCMI_CRのビット15）を同時に1にセットしないでください。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはパラレルデータクロックごとに8ビットデータをキャプチャします。

01：予約済み。選択できません。

10：予約済み。選択できません。

11：インタフェースはパラレルデータクロックごとに16ビットデータをキャプチャします。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **RDYPOL:**レディ（PSSI_RDY）の極性

このビットは、データがパラレルインタフェース上で有効でないときのPSSI_RDYピン上でのレベルを示します。

0：PSSI_RDYはアクティブローです（0は 受信側が受信準備完了していることを示します）。

1：PSSI_RDYはアクティブハイです（1は 受信側が受信準備完了していることを示します）。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **DEPOL**：データイネーブル（PSSI_DE）の極性

このビットは、データがパラレルインタフェース上で有効でないときのPSSI_DEピン上でのレベルを示します。

0：PSSI_DEはアクティブローです（0は データが有効であることを示します）。

1：PSSI_DEはアクティブハイです（1は データが有効であることを示します）。

ビット5 **CKPOL**：パラレルデータのクロック極性

このビットは、OUTENに応じて、パラレルクロックのキャプチャエッジまたは出力の駆動に使用されるエッジを設定します。

0：入力に対して立ち下がりエッジでアクティブ、または出力に対して立ち上がりエッジでアクティブ

1：入力に対して立ち上がりエッジでアクティブ、または出力に対して立ち下がりエッジでアクティブ

ビット4:0 予約済みであり、リセット値に保持する必要があります。

42.5.2 PSSIステータスレジスタ（PSSI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTT1B	RTT4B	Res.	Res.
												r	r		

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RTT1B：**FIFOは1バイトを転送する準備ができています

1：FIFOは1バイト（32ビット）の転送準備ができています。受信モードでは、少なくとも1バイトの有効データがFIFOにあることを意味します。送信モードでは、少なくとも1バイトの空きがFIFOにあることを意味します。

0：FIFOは1バイトの転送準備ができていません。

ビット2 **RTT4B：**FIFOは4バイトを転送する準備ができています

1：FIFOは4バイト（32ビット）の転送準備ができています。受信モードでは、少なくとも4バイトの有効データがFIFOにあることを意味します。送信モードでは、少なくとも4バイトの空きがFIFOにあることを意味します。

0：FIFOは4バイトの転送準備ができていません。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

42.5.3 PSSI raw割込みステータスレジスタ（PSSI_RIS）

アドレスオフセット：0x08

リセット値：0x0000 0000

PSSI_RISはraw割込みステータスを提供します。このレジスタは読出し専用です。読出し時、これはPSSI_IERレジスタ値でマスクする前の関連する割込みのステータスを返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_RIS	Res.
														r	

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OVR_RIS**：データバッファのオーバーラン／アンダーランraw割込みステータス

0：オーバーラン／アンダーランは発生しませんでした。

1：オーバーラン／アンダーランが発生しました：受信モードでのオーバーラン、送信モードでのアンダーラン。

このビットは、PSSI_ICRのOVR_ISCビットに1を書き込むことによってクリアされます。

ビット0 予約済みであり、リセット値に保持する必要があります。

42.5.4 PSSI割込み有効レジスタ（PSSI_IER）

アドレスオフセット：0x0C

リセット値：0x0000 0000

PSSI_IERレジスタは、割込みを有効にするために使用します。PSSI_IERビットの1つがセットされると、該当する割込みが有効になります。このレジスタは、読出しと書込みの両方のモードでアクセスできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_IE	Res.
														rw	

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OVR_IE**：データバッファのオーバーラン／アンダーラン割込みイネーブル

0：割込みは生成されません。

1：オーバーランエラーまたはアンダーランエラーのどちらかが発生した場合、割込みが生成されます。

ビット0 予約済みであり、リセット値に保持する必要があります。

42.5.5 PSSIマスク割込みステータスレジスタ（PSSI_MIS）

このPSSI_MISレジスタは読出し専用です。読出し時、これは、該当する割込みの現在のマスクされたステータス値（PSSI_IERの値によって変わります）を返します。PSSI_IER内の該当するイネーブル･ビットがセットされ、PSSI_RIS内の該当ビットがセットされている場合、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_MIS	Res.
														r	

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OVR_MIS**：データバッファのオーバーラン／アンダーランのマスクされた割込みステータス

このビットは、PSSI_IER/OVR_IEとPSSI_RIS/OVR_RISの両方とも1にセットされている場合のみ、1にセットされます。

0：オーバーラン／アンダーランエラーが発生したとき、割込みは生成されません。

1：オーバーランまたはアンダーランエラーがあり、PSSI_IERでOVR_IEビットがセットされている場合、割込みが生成されます。

ビット0 予約済みであり、リセット値に保持する必要があります。

42.5.6 PSSI割込みクリアレジスタ（PSSI_ICR）

アドレスオフセット：0x14

リセット値：0x0000 0000

PSSI_ICRレジスタは、書込み専用です。このレジスタのビットに1を書き込むことによってPSSI_RISおよびPSSI_MISレジスタの対応するビットをクリアします。0を書き込んでも、ビットの値は変化しません。このレジスタを読み出すと常に0が返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_ISC	Res.
														w	

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **OVR_ISC**：データバッファのオーバーラン／アンダーラン割込みステータスクリア
このビットに1を書き込むと、PSSI_RISのOVR_RISビットがクリアされます。

ビット0 予約済みであり、リセット値に保持する必要があります。

42.5.7 PSSIデータレジスタ（PSSI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0000

受信モード（OUTEN = 0）では、DMAコントローラはこのレジスタから受信データを読み取る必要があります。PSSI_DRへ書込み操作を行うと、エラー応答が発生します。FIFOで有効なバイト数より多くのバイトが読み出されると、無効なバイトはゼロを返します。

送信モード（OUTEN = 1）では、DMAコントローラはこのレジスタに送信データを書き込む必要があります。PSSI_DRへの読出し操作を行うと、エラー応答が発生します。

PSSI_DRに対して、32ビット、16ビット、および8ビットのアクセスはすべてサポートされています。たとえば、16ビットの読出し／書込み操作では、2バイトがFIFOから削除／FIFOへ追加されます。ただし、8ビットアクセスは、PSSIが一度に8ビットのデータを転送するように設定されている場合（PSSI_CRのEDM=00）にのみ許可されます。EDMが0にセットされていない場合にPSSI_DRへ8ビットアクセスを行うと、エラー応答が発生します。

すべてのアクセスはバイト0を含む必要があります。すなわち、8ビットアクセスはビット7～0に対して行い、16ビットアクセスはビット15～0に対して行う必要があります。バイト0を含まないアクセスを行うと、エラー応答が発生します。

PSSI_CRのENABLEビットが0にセットされているときにPSSI_DRにアクセスすると、エラー応答が発生します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BYTE3 [7:0]								BYTE2 [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYTE1 [7:0]								BYTE0 [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **BYTE3[7:0]**：データバイト3

ビット23:16 **BYTE2[7:0]**：データバイト2

ビット15:8 **BYTE1[7:0]**：データバイト1

ビット7:0 **BYTE0[7:0]**：データバイト0

42.5.8 PSSIレジスタマップ

表422.PSSIレジスタマップとリセット値

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	PSSI_CR	OUTEN	DMAEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DERDYCFG			Res.	Res.	Res.	ENABLE[2:0]			EDM		Res.	RDYPOL	Res.	DEPOL	CKPOL	Res.	Res.	Res.	Res.	Res.
	リセット値	0	1										0	0	0				0			0	0		0		0	0					
0x04	PSSI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTT1B	RTT4B	Res.	Res.
	リセット値																													0	0		
0x08	PSSI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_RIS	Res.
	リセット値																															0	
0x0C	PSSI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.am	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_IE	Res.
	リセット値																															0	
0x10	PSSI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_MIS	Res.
	リセット値																															0	
0x14	PSSI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_ISC	Res.
	リセット値																															0	
0x18 ~ 0x24	予約済み	Res.																															
0x28	PSSI_DR	BYTE3 [7:0]								BYTE2 [7:0]								BYTE1 [7:0]								BYTE0 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

43 LCD-TFTディスプレイコントローラ（LTDC）

このセクションは、STM32U599/5A9/5Fx/5Gxデバイスにのみ適用されます。

43.1 はじめに

このLCD-TFT（TFT液晶）ディスプレイコントローラは、さまざまなLCDおよびTFTパネルに直接接続するため、デジタルパラレルRGB（赤、緑、青）や、水平／垂直同期、ピクセルクロック、およびデータイネーブルの出力信号を提供します。

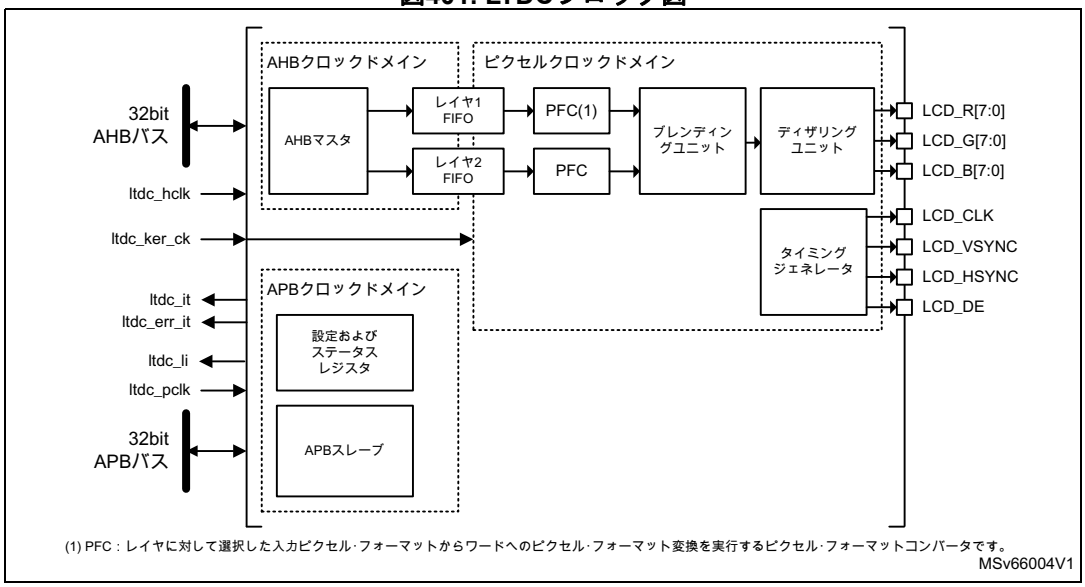
43.2 LTDCの主な機能

- 24bit RGBパラレルピクセル出力、8bit／ピクセル（RGB888）
- 専用FIFO（64x32bit）を備えた2つの表示レイヤ
- 各レイヤにつき最大256色（256x24bit）のカラールックアップテーブル（CLUT）
- さまざまなディスプレイパネルに対応するプログラム可能なタイミング
- プログラム可能なバックグラウンドカラー
- HSYNC、VSYNC、データイネーブルに対するプログラム可能な極性
- 各レイヤにつき最大8種類の入力カラーフォーマットが選択可能：
  - ARGB8888
  - RGB888
  - RGB565
  - ARGB1555
  - ARGB4444
  - L8（8bit輝度またはCLUT）
  - AL44（4bitアルファ＋4bit輝度）
  - AL88（8bitアルファ 8bit輝度）
- 各チャネルにつき、低ビットに対する疑似ランダムディザリング出力
  - 赤、緑、青に対するディザ幅2bit
- アルファ値（ピクセルごとまたは一定値）を用いた2つのレイヤ間の柔軟なブレンディング
- カラーキーイング（透明色）
- プログラム可能なウィンドウ位置とサイズ
- TFTカラー液晶ディスプレイをサポート
- 16 ワードのバースト付き AHB マスタインタフェース
- 最大4つのプログラム可能な割込みイベント

43.3 LTDCの機能詳細

43.3.1 LTDCのブロック図

図401. LTDCブロック図



43.3.2 LTDCピンおよび内部信号

下表に、LTDC信号インタフェースの一覧を示します。

表423.LTDC外部ピン

LCD-TFT信号	信号タイプ	説明
LCD_CLK	出力	クロック出力
LCD_HSYNC	出力	水平同期
LCD_VSYNC	出力	垂直同期
LCD_DE	出力	非データイネーブル
LCD_R[7:0]	出力	8bitの赤のデータ
LCD_G[7:0]	出力	8bitの緑のデータ
LCD_B[7:0]	出力	8bitの青のデータ

LTDCピンは、ユーザアプリケーションで設定する必要があります。使用されていないピンは、他の目的に使用できます。

24bitまでのLTDC出力（RGB888）に対して、16bit（RGB565）または18bit（RGB666）カラーのディスプレイに接続するため、8bpp未満の色深度を使用する場合、ディスプレイ側のRGBデータラインはLTDCのRGBデータラインのMSB側に接続する必要があります。

例えば、LTDCを16bit（RGB565）カラーのディスプレイと接続する場合、LTDCディスプレイ側のR[4:0]、G[5:0]、B[4:0] のデータラインのピンはLCD_R[7:3]、LCD_G[7:2]、LCD_B[7:3] ピンに接続する必要があります。

LTDCの内部信号を次の表に示します。

表424.LTDC内部信号

名前	信号タイプ	説明
ltdc_hclk	入力	LTDC AHBクロック
ltdc_pclk	入力	レジスタアクセス用のLTDC APBクロック
ltdc_ker_ck	入力	LCD_CLK（ピクセルクロック）生成に使用されるLTDCカーネルクロック
ltdc_it	出力	LTDCグローバル割込みリクエスト
ltdc_err_it	出力	LTDCグローバルエラー割込みリクエスト
ltdc_li	出力	LTDCライン割込みフラグ

次の表に、LTDCフラグの接続方法を示します。

表425.LTDCトリガの相互接続

トリガ名	方向	トリガ転送元／転送先
ltdc_li	出力	gpdma_trigsel[47]

43.3.3 LTDCのリセットとクロック

LTDCコントローラのペリフェラルでは次のクロックドメインを使用します。

- AHB クロックドメイン（ltdc_aclk）
  このドメインには、メモリからレイヤFIFOへのデータ転送のためのLTDC AHB マスタインタフェースとフレームバッファ設定レジスタが含まれています。
- APBクロックドメイン（ltdc_pclk）
  このドメインには、グローバル設定レジスタと割込みレジスタが含まれています。
- ピクセルクロックドメイン（LCD_CLK）
  このドメインには、ピクセルデータ生成、レイヤ設定レジスタとともにLTDCインタフェース信号生成器が含まれています。LCD_CLK出力は、パネル要件に従って設定する必要があります。LCD_CLKは、特定のPLL出力（リセットとクロック制御のセクションを参照）から生成されます。

次の表に、各レジスタのクロックドメインの概要を示します。

表426.各レジスタのクロックドメイン

LTDCレジスタ	クロックドメイン
LTDC_LxCR	ltdc_hclk
LTDC_LxCFBAR	
LTDC_LxCFBLR	
LTDC_LxCFBLNR	
LTDC_SRCR	ltdc_pclk
LTDC_IER	
LTDC_ISR	
LTDC_ICR	

表426.各レジスタのクロックドメイン （続き）

LTDCレジスタ	クロックドメイン
LTDC_SSCR	ピクセルクロック（LCD_CLK）
LTDC_BPCR	
LTDC_AWCR	
LTDC_TWCR	
LTDC_GCR	
LTDC_BCCR	
LTDC_LIPCR	
LTDC_CPSR	
LTDC_CDSR	
LTDC_LxWHPCR	
LTDC_LxWVPCR	
LTDC_LxCKCR	
LTDC_LxPFCR	
LTDC_LxCACR	
LTDC_LxDCCR	
LTDC_LxBFCR	
LTDC_LxCLUTWR	

LTDCレジスタにアクセスするとき、以下の時間、APBバスが停止するので、注意する必要があります。

- レジスタの書込みアクセスと更新の場合、ltdc_pclkの6周期 + LCD_CLKの5周期（AHBクロックドメインのレジスタではltdc_hclkの5周期）
- レジスタ読出しアクセスの場合、ltdc_pclkの7周期 + LCD_CLKの5周期（AHBクロックドメインのレジスタではltdc_hclkの5周期）

ltdc_pclkクロックドメインのレジスタでは、レジスタ書込みアクセス時にltdc_pclkの6周期の間、読出しアクセス時にltdc_pclkの7周期の間、APBバスは停止します。

LTDCコントローラは、RCCの該当するビットをセットすることによってリセット可能であり、そうすることで、3つのクロックドメインがリセットされます。

43.4 LTDCのプログラム可能なパラメータ

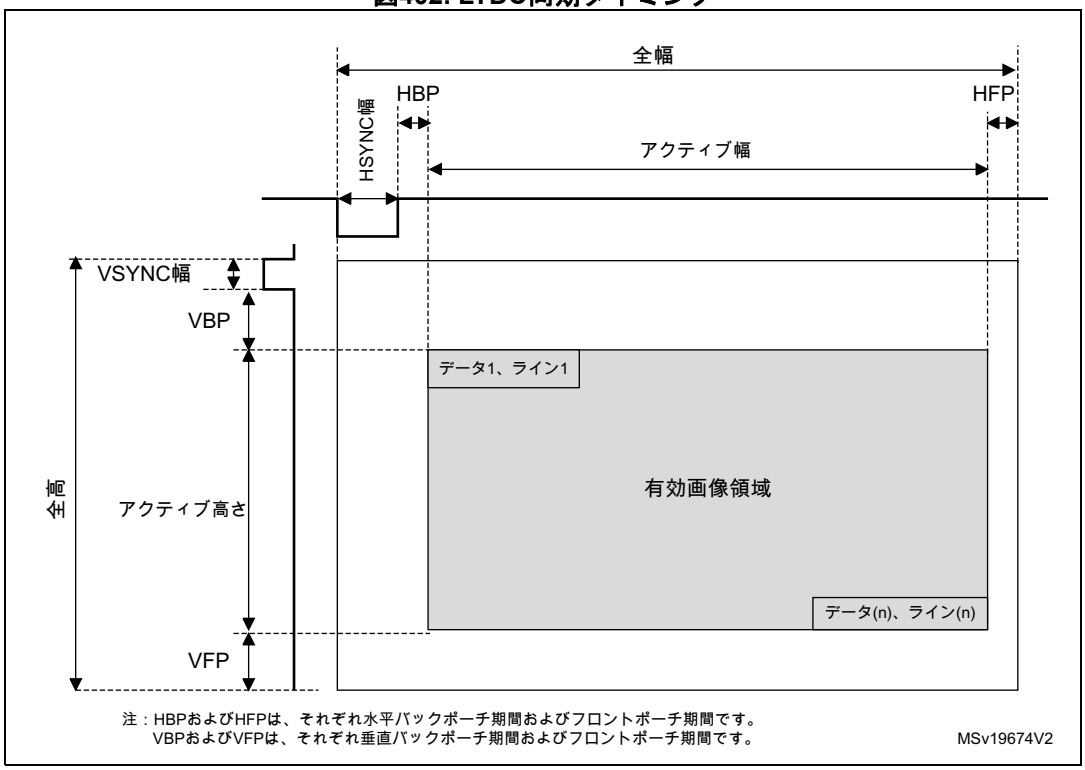
LTDCコントローラのパラメータは、フレキシブルに設定することができます。LTDC_GCRレジスタを介して有効または無効にすることができます。

43.4.1 LTDCグローバル設定パラメータ

同期タイミング

下の図は、図401のブロック図で示す同期タイミングジェネレータブロックで生成される設定可能なタイミングパラメータを示します。水平／垂直同期信号、ピクセルクロックおよびデータイネーブル信号が生成されます。

図402. LTDC同期タイミング



LTDCのプログラム可能な同期タイミングは次のとおりです。

- HSYNCおよびVSYNC幅：LTDC_SSCRレジスタにおけるHSYNC幅 - 1およびVSYNC幅 - 1の値をプログラムして、設定される水平および垂直同期信号幅です。
- HBPおよびVBP：LTDC_BPCRレジスタにおけるHSYNC幅 + HBP - 1の合計値とVSYNC幅 + VBP - 1の合計値をプログラムして、設定される水平および垂直同期バックポーチ幅です。
- アクティブ幅とアクティブ高さ：アクティブ幅とアクティブ高さは、LTDC_AWCR レジスタでHSYNC幅 + HBP + アクティブ幅 - 1の合計値とVSYNC幅 + VBP + アクティブ高さ - 1の合計値をプログラムして設定します。
- 全幅：全幅は、LTDC_TWCRレジスタでHSYNC幅 + HBP + アクティブ幅 + HFP - 1の合計値をプログラムして設定します。HFPは、水平フロントポーチ期間です。
- 全高：全高は、LTDC_TWCRレジスタでVSYNC高さ + VBP + アクティブ高さ + VFP - 1の合計値をプログラムして設定します。VFPは、垂直フロントポーチ期間です。

注： **LTDCが有効の場合、垂直同期エリアの最初の水平同期ピクセルとして、X/Y = 0/0位置でタイミング生成が開始し、その後バックポーチ、アクティブデータ表示エリア、フロントポーチと続きます。LTDCが無効の場合、タイミングジェネレータブロックはX = 全幅 - 1、Y = 全高 - 1にリセットされ、垂直同期フェーズの前の最後のピクセルを保持し、FIFOは一掃されます。したがって、連続的に出力されるのはブランキングデータのみとなります。**

同期タイミング設定の例

LTDCタイミング（パネルデータシートより抽出する必要あり）:

- 水平および垂直同期幅：0xAピクセルおよび0x2ライン
- 水平および垂直バックポーチ：0x14ピクセルおよび0x2ライン
- アクティブ幅とアクティブ高さ：0x140ピクセル、0xF0ライン（320x240）
- 水平フロントポーチ：0xAピクセル
- 垂直フロントポーチ：0x4ライン

LTDCタイミングレジスタでプログラムされる値は次のとおりです。

- LTDC_SSCRレジスタ：0x00090001にプログラム
（HSW[11:0] は0x9、VSH[10:0] は0x1）
- LTDC_BPCRレジスタ：0x001D0003にプログラム
（AHBP[11:0] は0x1D（0xA + 0x13）、AVBP[10:0]A は0x3（0x2 + 0x1））
- LTDC_AWCRレジスタ：0x015D00F3にプログラム
（AAW[11:0] は0x15D（0xA + 0x14 + 0x13F）、AAH[10:0] は0xF3（0x2 + 0x2 + 0xEF））
- LTDC_TWCRレジスタ：0x00000167にプログラム
（TOTALW[11:0] は0x167（0xA + 0x14 + 0x140 + 0x9））
- LTDC_THCRレジスタ：0x000000F7にプログラム
（TOTALH[10:0] は0xF7（0x2 + 0x2 + 0xF0 + 3））

プログラム可能な極性

水平／垂直同期信号、データイネーブル信号、ピクセルクロックの極性は、LTDC_GCRレジスタでアクティブハイまたはアクティブローにプログラムできます。

バックグラウンドカラー

一定の背景色（RGB888）をLTDC_BCCRレジスタでプログラムできます。ボトムレイヤとのブレンディングに使用します。

ディザリング

LFSRを使用した疑似乱数ディザリングを使用して、各ピクセルカラーチャネル（R、G、またはB）値に小さな乱数値（閾値）を加えますので、18bitディスプレイで24bitデータを表示する際に、MSBが切り上げられる場合があります。このようにして、フレームごとに異なるデータの四捨五入にはディザリングが使用されす。

疑似乱数ディザリングは、LSBを閾値と比較してLSBが閾値以上であればMSBにのみ1を加算することと同じです。ディザリングが適用されると、LSBは通常切り捨てられます。

加算される疑似乱数値の幅は、各カラーチャネルに対して2bit、つまり赤に対して2bit、緑に対して2bit、青に対して2bitです。

LTDCが有効になると、LFSRが最初のアクティブなピクセルで動作を開始し、ブランキング期間中やディザリングがオフになってもそのまま動作を続けます。LTDCが無効になると、LFSRはリセットされます。

ディザリングは、LTDC_GCRレジスタを使って動作中にオン／オフを切り替えることができます。

シャドウレジスタの再ロード

シャドウイングされる設定レジスタがいくつかあります。シャドウレジスタの値は、LTDC_SRCRレジスタでの設定に従い、アクティブなレジスタへの書込み時、または 垂直ブランキング期間の始めに、これらのレジスタに直ちに再ロードすることができます。即時再ロード設定が選択されている場合、すべての新規レジスタが書き込まれてから再ロードがアクティブになるようにする必要があります。

シャドウレジスタは、再ロードが完了するまでは再変更してはいけません。シャドウレジスタを読み出すと、実際にアクティブな値が読み出されます。新しく書き込まれた値は、再ロードが実行されるまでは読み出せません。

LTDC_IERレジスタで有効になっていれば、レジスタ再ロード割込みを生成することができます。

LTDC_LxCLUTWRを除き、レイヤ1とレイヤ2のすべてのレジスタはシャドウレジスタです。

割込み生成イベント

割込み設定については、セクション 43.5：LTDC割込みを参照してください。

43.4.2 レイヤのプログラム可能なパラメータ

最大2つのレイヤが有効化、無効化でき、また別々に設定することができます。レイヤの表示順序は固定で、下から上に順番に表示されます。2つのレイヤが有効になっている場合は、レイヤ2が上部に表示されるウィンドウとなります。

ウィンドウイング

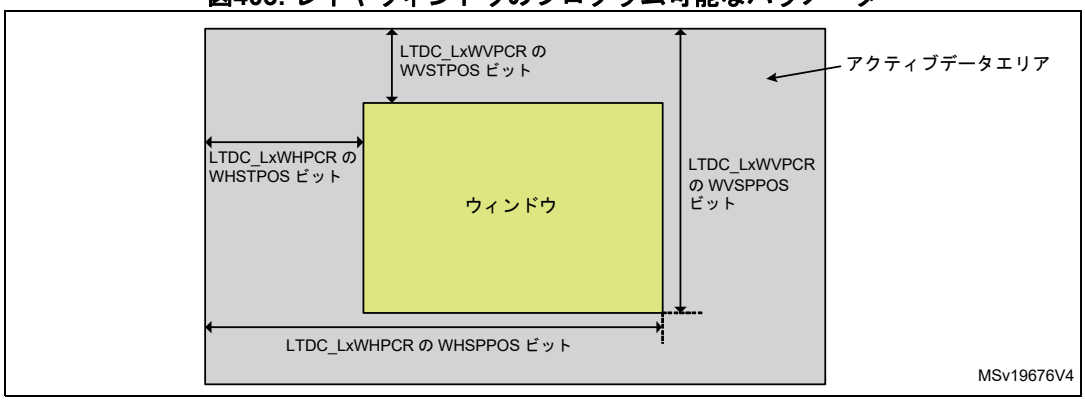
すべてのレイヤは位置決めとサイズ変更ができますが、アクティブ表示エリア内であることが必要です。

ウィンドウ位置およびサイズは、左上と右下のX/Y位置、および内部タイミングジェネレータによって設定され、この内部タイミングジェネレータは、同期、バックポーチサイズ、アクティブデータエリアから成っています。LTDC_LxWHPCRおよびLTDC_WVPCRレジスタを参照してください。

プログラム可能なレイヤ位置とサイズにより、ラインの最初／最後の見えるピクセルと、ウィンドウの最初／最後の見えるラインが決まります。画像フレームの全体または一部のいずれでも表示することができます（図403 参照）。

- レイヤ内の最初と最後の見えるピクセルは、LTDC_LxWHPCRレジスタ内のWHSTPOS[11:0] およびWHSPPOS[11:0] を設定することによりセットされます。
- レイヤ内の最初と最後の見えるラインは、LTDC_LxWVPCRレジスタ内の WVSTPOS[10:0] および WVSPPOS[10:0] を設定することによりセットされます。

図403. レイヤウィンドウのプログラム可能なパラメータ



ピクセル入力フォーマット

レイヤのフレームバッファ内に保存されるデータには、プログラム可能なピクセルフォーマットが使用されます。

LTDC_LxPFCRレジスタで、各レイヤに対して最大8種類までの入力ピクセルフォーマットを設定できます。

ピクセルデータはフレームバッファから読み出され、次のように内部8888（ARGB）フォーマットに変換されます。幅が8bit未満のコンポーネントは、ビットレプリケーションによって8bitまで拡張されます。選択されたビット範囲は、8bitより長くなるまで複数回連結されます。得られたベクタから、8つのMSBビットが選択されます。例：RGB565の赤チャネルの5bitは（ビット配置）43210432になります（3つのLSBに5bit中の3つのMSBが入る）。

下表で、選択したフォーマットに応じたピクセルデータのマッピングを説明しています。

表427.ピクセルデータマッピングとカラーフォーマット

ARGB8888			
@+3 A_x[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 A_{x+1}[7:0]	@+6 R_{x+1}[7:0]	@+5 G_{x+1}[7:0]	@+4 B_{x+1}[7:0]
RGB888			
@+3 B_{x+1}[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 G_{x+2}[7:0]	@+6 B_{x+2}[7:0]	@+5 R_{x+1}[7:0]	@+4 G_{x+1}[7:0]
RGB565			
@+3 R_{x+1}[4:0] G_{x+1}[5:3]	@+2 G_{x+1}[2:0] B_{x+1}[4:0]	@+1 R_x[4:0] G_x[5:3]	@ G_x[2:0] B_x[4:0]
@+7 R_{x+3}[4:0] G_{x+3}[5:3]	@+6 G_{x+3}[2:0] B_{x+3}[4:0]	@+5 R_{x+2}[4:0] G_{x+2}[5:3]	@+4 G_{x+2}[2:0] B_{x+2}[4:0]
ARGB1555			
@+3 A_{x+1}[0]R_{x+1}[4:0] G_{x+1}[4:3]	@+2 G_{x+1}[2:0] B_{x+1}[4:0]	@+1 A_x[0] R_x[4:0] G_x[4:3]	@ G_x[2:0] B_x[4:0]
@+7 A_{x+3}[0]R_{x+3}[4:0] G_{x+3}[4:3]	@+6 G_{x+3}[2:0] B_{x+3}[4:0]	@+5 A_{x+2}[0]R_{x+2}[4:0]G_{x+2}[4:3]	@+4 G_{x+2}[2:0] B_{x+2}[4:0]
ARGB4444			
@+3 A_{x+1}[3:0]R_{x+1}[3:0]	@+2 G_{x+1}[3:0] B_{x+1}[3:0]	@+1 A_x[3:0] R_x[3:0]	@ G_x[3:0] B_x[3:0]
@+7 A_{x+3}[3:0]R_{x+3}[3:0]	@+6 G_{x+3}[3:0] B_{x+3}[3:0]	@+5 A_{x+2}[3:0]R_{x+2}[3:0]	@+4 G_{x+2}[3:0] B_{x+2}[3:0]
L8			
@+3 L_{x+3}[7:0]	@+2 L_{x+2}[7:0]	@+1 L_{x+1}[7:0]	@ L_x[7:0]

表427.ピクセルデータマッピングとカラーフォーマット （続き）

@+7 L_{x+7}[7:0]	@+6 L_{x+6}[7:0]	@+5 L_{x+5}[7:0]	@+4 L_{x+4}[7:0]
AL44			
@+3 A_{x+3}[3:0] L_{x+3}[3:0]	@+2 A_{x+2}[3:0] L_{x+2}[3:0]	@+1 A_{x+1}[3:0] L_{x+1}[3:0]	@ A_x[3:0] L_x[3:0]
@+7 A_{x+7}[3:0] L_{x+7}[3:0]	@+6 A_{x+6}[3:0] L_{x+6}[3:0]	@+5 A_{x+5}[3:0] L_{x+5}[3:0]	@+4 A_{x+4}[3:0] L_{x+4}[3:0]
AL88			
@+3 A_{x+1}[7:0]	@+2 L_{x+1}[7:0]	@+1 A_x[7:0]	@ L_x[7:0]
@+7 A_{x+3}[7:0]	@+6 L_{x+3}[7:0]	@+5 A_{x+2}[7:0]	@+4 L_{x+2}[7:0]

カラールックアップテーブル（CLUT）

CLUTは、LTDC_LxCRレジスタですべてのレイヤに対してランタイム時に有効にできますが、L8、AL44、AL88入力ピクセルフォーマット使用時、インデックスカラーの場合にのみ役立ちます。

まず、CLUTにそのピクセルの本来のR、G、B値に代わるR、G、B値をロードする必要があります（インデックスカラー）。各色（RGB値）には、それ自体のアドレスがあり、CLUT内の位置を表します。

R、G、B値とそれぞれのアドレスは以下のようにLTDC_LxCLUTWRレジスタでプログラムされます。

- L8およびAL88入力ピクセルフォーマットの場合、CLUTは256色でロードする必要があります。各色のアドレスは、LTDC_LxCLUTWRレジスタ内のCLUTADDビットで設定されます。
- AL44入力ピクセルフォーマットの場合、CLUTは16色のみでロードする必要があります。各色のアドレスは、4bit Lチャネルを次のように8bitに複製することによって埋める必要があります。
  - L0（インデックスカラー 0）、アドレス0x00
  - L1、アドレス0x11
  - L2、アドレス0x22
  - .....
  - L15、アドレス0xFF

カラーフレームバッファアドレス

各レイヤには、LTDC_LxCFBARレジスタで設定されるカラーフレームバッファの開始アドレスがあります。

レイヤが有効になると、カラーフレームバッファからデータがフェッチされます。

カラーフレームバッファ長

すべてのレイヤには、バイト数単位でのカラーフレームバッファの総ライン長設定とフレームバッファ内のライン数設定があり、それぞれLTDC_LxCFBLRおよびLTDC_LxCFBLNRレジスタで設定可能です。

ライン長とライン数の設定は、フレームバッファの最後でレイヤFIFOへのデータのプリフェッチを停止するために使用されます。

- 必要なバイト数よりも少なく設定されている場合、FIFO　アンダーラン割込みが事前に有効になっていると、該当する割込みが生成されます。
- 実際に必要なバイト数よりも多く設定されている場合、FIFO から読み出された不要なデータは破棄されます。不要なデータは表示されません。

カラーフレームバッファピッチ

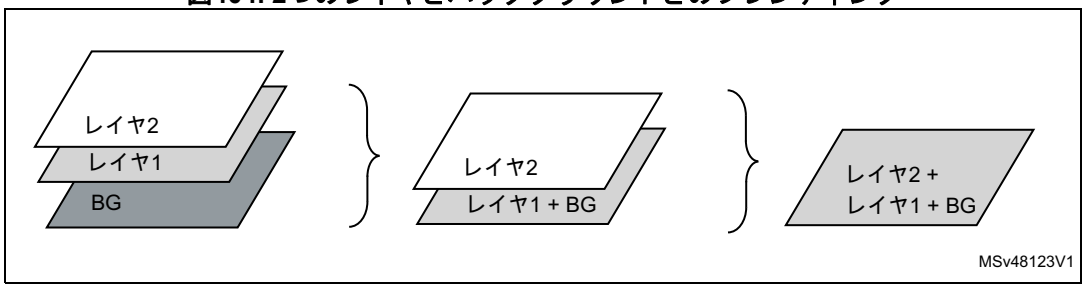
すべてのレイヤには、カラーフレームバッファに対して設定可能なピッチがあります。これはあるラインの開始から次のラインの始まりまでの距離をバイト数で表したものです。ピッチは、LTDC_LxCFBLRレジスタで設定します。

レイヤのブレンディング

ブレンディングは常にアクティブとなっており、2つのレイヤはLTDC_LxBFCRレジスタで設定されるブレンディング係数に従って混合することができます。

混合順序は固定で、下から上に順番に混合されます。2つのレイヤが有効になっている場合、まず レイヤ1がバックグラウンドカラーと混合され、その後レイヤ2がレイヤ1とバックグラウンドとの混合色と混合されます。次の図を参照してください。

図404. 2つのレイヤとバックグラウンドとのブレンディング



デフォルト色

すべてのレイヤには、フォーマットARGBでデフォルト色を設定することができ、このデフォルト色は定義されているレイヤウィンドウ外またはレイヤが無効になっているときに使用されます。

デフォルト色は、LTDC_LxDCCRレジスタで設定します。

ブレンディングは、レイヤの1つが無効になっていても必ず2つのレイヤ間で実施されます。レイヤが無効になっているときにデフォルト色が表示されないようにするには、LTDC_LxBFCRレジスタの該当レイヤのブレンディング係数をリセット値に保持します。

カラーキーイング

カラーキー（RGB）を設定し、透明なピクセルを表すことができます。

カラーキーイングが有効になっていると、現在のピクセル（フォーマット変換後でCLUT個別ブレンディング前）がカラーキーと比較されます。プログラムされたRGB値と一致した場合、そのピクセルのすべてのチャネル（ARGB）が0にセットされます。

カラーキー値を設定することにより、ランタイム時にその値を使用してピクセルRGB値を置換することができます。

カラーキーイングは、LTDC_LxCKCRレジスタで有効にします。

カラーキーイングは、LTDC_LxCKCRレジスタで設定します。プログラムされた値は、ARGB888へのピクセルフォーマット変換後の現在のピクセルと比較するピクセルフォーマットに依存しています。

例：ミッドイエロー色（赤50 % + 緑50 %）が透明カラーキーとして使用された場合：

- RGB565では、ミッドイエロー色は0x8400です。LTDC_LxCKCRを0x848200にセットします。
- ARGB8888では、ミッドイエロー色は0x808000です。LTDC_LxCKCRを0x808000にセットします。
- すべてのCLUTベースカラーモード（L8、AL88、AL44）で、パレットエントリの1つを0x808000のミッドイエロー色とし、LTDC_LxCKCRを0x808000に設定します。

43.5 LTDC割込み

LTDCにより、2つの割込みベクタの論理和をとった4つのマスク可能な割込みが使用できます。

割込みソースは、LTDC_IERレジスタを介して別々に有効または無効にできます。適切なマスクビットを1にセットすることで、対応する割込みが有効になります。

次のイベントによって2つの割込みが生成されます。

- ライン割込み：プログラムされたライン到達時に生成されます。ライン割込み位置は、LTDC_LIPCRレジスタでプログラムします。
- レジスタ再ロード割込み:垂直ブランキング期間中にシャドウレジスタの再ロードが実施されると生成されます。
- FIFOアンダーラン割込み：空のレイヤFIFOからピクセルが要求されると生成されます。
- 転送エラー割込み：データ転送中に AHB バスエラーが発生すると生成されます。

これらの割込みイベントは、下図に説明するようにNVICコントローラに接続されています。

図405. 割込みイベント

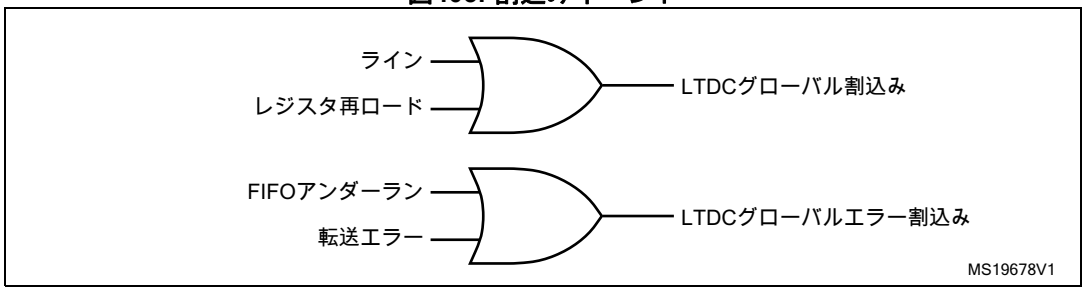


表428.LTDC割込みリクエスト

割込みイベント	イベント･フラグ	イネーブル制御ビット
ライン	LIF	LIE
レジスタ再ロード	RRIF	RRIEN
FIFOアンダーラン	FUDERRIF	FUDERRIE
転送エラー	TERRIF	TERRIE

43.6 LTDCプログラミング手順

LTDCをプログラムするには、以下の手順が必要です。

1. RCCレジスタでLTDCクロックを有効にします。
2. パネルデータシートに従って必要なピクセルクロックを設定します。
3. 同期タイミングを設定します。セクション43.4.1に説明されているように、パネルデータシートに従ってVSYNC、HSYNC、垂直／水平バックポーチ、アクティブデータエリア、フロントポーチなどのタイミングを設定します。
4. LTDC_GCRレジスタで、同期信号とクロック極性を設定します。
5. 必要であれば、LTDC_BCCRレジスタでバックグラウンドカラーを設定します。
6. LTDC_IERおよびLTDC_LIPCRレジスタで必要な割込みを設定します。
7. 以下によって、レイヤ1/2のパラメータを設定します。
   - LTDC_LxWHPCRおよびLTDC_WVPCRレジスタ内のレイヤウィンドウの水平／垂直位置をプログラムします。レイヤウィンドウは、アクティブデータエリア内にある必要があります。
   - LTDC_LxPFCRレジスタでピクセル入力フォーマットをプログラムします。
   - LTDC_LxCFBARレジスタでカラーフレームバッファ開始アドレスをプログラムします。
   - LTDC_LxCFBLR レジスタでカラーフレームバッファのライン長およびピッチをプログラムします。
   - LTDC_LxCFBLNRレジスタでカラーフレームバッファのライン数をプログラムします。
   - 必要であれば、LTDC_LxCLUTWRレジスタでRGB値とそのアドレスと共にCLUTをロードします。
   - 必要であれば、LTDC_LxDCCR および LTDC_LxBFCR レジスタでデフォルト色とブレンディング係数をそれぞれ設定します。
8. レイヤ1/2を有効にし、必要であればLTDC_LxCRレジスタでCLUTを有効にします。
9. 必要であれば、LTDC_GCRおよびLTDC_LxCKCRレジスタでディザリングとカラーキーイングをそれぞれ有効にします。それらは、動作中に有効にすることもできます。
10. LTDC_SRCRレジスタを介してシャドウレジスタをアクティブレジスタに再ロードします。
11. LTDC_GCRレジスタでLTDCコントローラを有効にします。
12. CLUT を除き、すべてのレイヤパラメータは動作中に変更できます。新しい設定は、直ちに再ロードするか、LTDC_SRCRレジスタを設定することにより、垂直ブランキング期間中に再ロードする必要があります。

注： **すべてのレイヤのレジスタはシャドウイングされます。レジスタに書き込まれたら、再ロードが完了するまでは再変更してはなりません。したがって、同じレジスタへの新しい書込みは、前の設定がまだ再ロードされていなければその設定を上書きします。**

43.7 LTDCレジスタ

43.7.1 LTDC同期サイズ設定レジスタ（LTDC_SSCR）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタによって、水平同期ピクセル数から1を引いた数と、垂直同期ライン数から1を引いた数を指定します。設定例については図402およびセクション43.4を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HSW[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	VSH[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **HSW[11:0]**：水平同期幅（ピクセルクロック周期の単位）
このビットフィールドによって、水平同期ピクセル数から1を引いた数を指定します。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **VSH[10:0]**：垂直同期高さ（水平スキャンラインの単位）
このビットフィールドによって、垂直同期の高さから1を引いた数を指定します。これは、水平同期ライン数を表します。

43.7.2 LTDCバックポーチ設定レジスタ（LTDC_BPCR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

このレジスタによって、水平同期およびバックポーチの累積ピクセル数から1を引いた数（HSYNC幅 + HBP - 1）と垂直同期およびバックポーチの累積ライン数から1を引いた数（VSYNC高さ + VBP - 1）を指定します。
設定例については図402およびセクション43.4を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AHBP[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	AVBP[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **AHBP[11:0]**：累積水平バックポーチ（ピクセルクロック周期の単位）
このビットにより、水平同期および水平バックポーチピクセル数から1を引いた数から成る累積水平バックポーチ幅を指定します。
水平バックポーチは、水平同期がインアクティブになってから次のスキャンラインのアクティブ表示エリアの開始までの期間です。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **AVBP[10:0]**：累積垂直バックポーチ（水平スキャンラインの単位）

このビットにより、垂直同期および垂直バックポーチライン数から1を引いた数から成る累積垂直バックポーチ幅を指定します。

垂直バックポーチは、フレーム開始時から次のフレームの最初のアクティブなスキャンライン開始までの水平スキャンライン数です。

43.7.3 LTDCアクティブ幅設定レジスタ（LTDC_AWCR）

アドレスオフセット：0x010

リセット値：0x0000 0000

このレジスタによって、水平同期、バックポーチ、アクティブピクセルの累積数から1を引いた数（HSYNC幅 + HBP + アクティブ幅 - 1）と垂直同期、バックポーチライン、アクティブラインの累積数から1を引いた数（VSYNC高さ + VBP + アクティブ高さ - 1）を指定します。設定例については図402およびセクション43.4を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AAW[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	AAH[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **AAW[11:0]**：累積アクティブ幅（ピクセルクロック周期の単位）

このビットにより、水平同期、水平バックポーチ、アクティブピクセル数から1を引いた数から成る累積アクティブ幅を定義します。

アクティブ幅は、パネルスキャンラインのアクティブ表示エリアにおけるピクセル数です。

最大ピクセルクロックに従った対応最大アクティブ幅については、デバイスデータシートを参照してください。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **AAH[10:0]**：累積アクティブ高さ（水平スキャンラインの単位）

このビットにより、垂直同期、垂直バックポーチ、アクティブ高さのライン数から1を引いた数から成る累積高さを指定します。アクティブ高さは、パネル内のアクティブなラインの数です。

最大ピクセルクロックに従った対応最大アクティブ高さについては、デバイスデータシートを参照してください。

43.7.4 LTDC全幅設定レジスタ（LTDC_TWCR）

アドレスオフセット：0x014

リセット値：0x0000 0000

このレジスタによって、水平同期、バックポーチ、アクティブ、フロントポーチピクセルの累積数から1を引いた数（HSYNC幅 + HBP + アクティブ幅 + HFP - 1）と垂直同期、バックポーチライン、アクティブ、フロントラインの累積数から1を引いた数（VSYNC高さ + VBP + アクティブ高さ + VFP-1）を指定します。設定例については図402 およびセクション43.4を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TOTALW[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TOTALH[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **TOTALW[11:0]**：全幅（ピクセルクロック周期の単位）

このビットにより、水平同期、水平バックポーチ、アクティブ幅、水平フロントポーチピクセル数から1を引いた数から成る累積全幅を指定します。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **TOTALH[10:0]**：全高（水平スキャンラインの単位）

このビットにより、垂直同期、垂直バックポーチ、アクティブ高さ、垂直フロントポーチ高さのライン数から1を引いた数から成る累積高さを指定します。

43.7.5 LTDCグローバル制御レジスタ（LTDC_GCR）

アドレスオフセット：0x018

リセット値：0x0000 2220

このレジスタにより、LCD-TFTコントローラのグローバル設定を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN
rw	rw	rw	rw												rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	r	r	r		r	r	r		r	r	r				rw

ビット31 **HSPOL**：水平同期極性

このビットは、ソフトウェアによってセット／クリアされます。

0：水平同期極性はアクティブローです。

1：水平同期極性はアクティブハイです。

ビット30 **VSPOL**：垂直同期極性

このビットは、ソフトウェアによってセット／クリアされます。

0：垂直同期はアクティブローです。

1：垂直同期はアクティブハイです。

ビット29 **DEPOL**：非データイネーブル極性
このビットは、ソフトウェアによってセット／クリアされます。
0：非データイネーブル極性はアクティブローです。
1：非データイネーブル極性はアクティブハイです。

ビット28 **PCPOL**：ピクセルクロック極性
このビットは、ソフトウェアによってセット／クリアされます。
0：ピクセルクロック極性はアクティブローです。
1：ピクセルクロックはアクティブハイです。

ビット27:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **DEN**：ディザリングイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ディザリングは無効です。
1：ディザリングは有効です。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **DRW[2:0]**：ディザ赤の幅
このビットはディザ赤のビットを返します。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **DGW[2:0]**：ディザ緑の幅
このビットはディザ緑のビットを返します。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **DBW[2:0]**：ディザ青の幅
このビットはディザ青のビットを返します。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **LTDCEN**：LCD-TFTコントローライネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：LTDCは無効です。
1：LTDCは有効です。

43.7.6 LTDCシャドウ再ロード設定レジスタ（LTDC_SRCR）

アドレスオフセット：0x024

リセット値：0x0000 0000

このレジスタにより、シャドウレジスタ値をアクティブレジスタに直ちに再ロードすることもできれば、垂直ブランキング期間中にすることもできます。LTDC_L1CLUTWRおよびLTDC_L2CLUTWRを除き、レイヤ1とレイヤ2のすべてのレジスタはシャドウレジスタです。

シャドウレジスタは、アクティブな値を読み出します。再ロードが完了するまでは、「古い」値が読み出されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBR	IMR
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **VBR**：垂直ブランキングの再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます（いったんセットされるとレジスタの書込みによってクリアすることはできません）。

0：影響なし。

1：シャドウレジスタは、垂直ブランキング期間中（アクティブ表示エリア後の最初のライン開始時）に再ロードされます。

ビット0 **IMR**：即時再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます。

0：影響なし。

1：シャドウレジスタが直ちに再ロードされます。

43.7.7 LTDCバックグラウンドカラー設定レジスタ（LTDC_BCCR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

このレジスタにより、バックグラウンド色（RGB888）を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCGREEN[7:0]								BCBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **BCRED[7:0]**：バックグラウンドカラー、赤の値

このビットにより、バックグラウンドの赤の値を設定します。

ビット15:8 **BCGREEN[7:0]**：バックグラウンドカラー、緑の値
このビットにより、バックグラウンドの緑の値を設定します。

ビット7:0 **BCBLUE[7:0]**：バックグラウンドカラー、青の値
このビットにより、バックグラウンドの青の値を設定します。

43.7.8 LTDC割込み有効レジスタ（LTDC_IER）

アドレスオフセット：0x034

リセット値：0x0000 0000

このレジスタは、対応するビットを1にセットすることで、どのステータスフラグが割込み要求を生成するかを決定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIE	TERRIE	FUIE	LIE
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RRIE**：レジスタ再ロード割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：レジスタ再ロード割込みは無効です。
1：レジスタ再ロード割込みは有効です。

ビット2 **TERRIE**：転送エラー割込みの有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：転送エラー割込みは無効です。
1：転送エラー割込みは有効です。

ビット1 **FUIE**：FIFOアンダーラン割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFOアンダーラン割込みは無効です。
1：FIFOアンダーラン割込みは有効です。

ビット0 **LIE**：ライン割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ライン割込みは無効です。
1：ライン割込みは有効です。

43.7.9 LTDC割込みステータスレジスタ（LTDC_ISR）

アドレスオフセット：0x038

リセット値：0x0000 0000

このレジスタは、割込みステータスフラグを返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIF	TERRIF	FUIF	LIF
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RRIF**：レジスタ再ロード割込みフラグ

0：レジスタ再ロード割込みは生成されていません。

1：垂直ブランキング再ロード発生時に（およびアクティブエリア到達後の最初のラインで）レジスタ再ロード割込みが生成されます。

ビット2 **TERRIF**：転送エラー割込みフラグ

0：転送エラー割込みは生成されていません。

1：バスエラー発生時に転送エラー割込みが生成されます。

ビット1 **FUIF**：FIFOアンダーラン割込みフラグ

0：FIFOアンダーラン割込みは生成されていません。

1：レイヤFIFOのいずれかが空で、FIFOからピクセルデータが読み出された場合に、FIFOアンダーラン割込みが生成されます。

ビット0 **LIF**：ライン割込みフラグ

0：ライン割込みは生成されていません。

1：プログラムされたラインに到達時にライン割込みが生成されます。

43.7.10 LTDC割込みクリアレジスタ（LTDC_ICR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRRIF	CTERRIF	CFUIF	CLIF
												w	w	w	w

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **CRRIF**：レジスタ再ロード割込みフラグをクリアします

0：影響なし

1：LTDC_ISRレジスタのRRIFフラグをクリアします。

ビット2 **CTERRIF**：転送エラー割込みフラグをクリアします

0：影響なし

1：LTDC_ISRレジスタのTERRIFフラグをクリアします。

ビット1 **CFUIF**：FIFOアンダーラン割込みフラグをクリアします

0：影響なし

1：LTDC_ISRレジスタのFUDERRIFフラグをクリアします。

ビット0 **CLIF**：ライン割込みフラグをクリアします

0：影響なし

1：LTDC_ISRレジスタのLIFフラグをクリアします。

43.7.11 LTDCライン割込み位置設定レジスタ（LTDC_LIPCR）

アドレスオフセット：0x040

リセット値：0x0000 0000

このレジスタにより、ライン割込みの位置を指定します。プログラムされるライン値は、タイミングパラメータに依存します。図402を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	LIPOS[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **LIPOS[10:0]**：ライン割込み位置

このビットにより、ライン割込み位置を設定します。

43.7.12 LTDC現在位置ステータスレジスタ（LTDC_CPSR）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CXPOS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CYPOS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **CXPOS[15:0]**：現在のX位置

このビットは、現在のX位置を返します。

ビット15:0 **CYPOS[15:0]**：現在のY位置

このビットは、現在のY位置を返します。

43.7.13 LTDC現在表示ステータスレジスタ（LTDC_CDSR）

アドレスオフセット：0x048

リセット値：0x0000 000F

このレジスタは、HSYNC、VSYNC、水平／垂直DE信号が制御する現在の表示フェーズのステータスを返します。

例：現在の表示フェーズが垂直同期であれば、VSYNCSビットがセットされます（アクティブハイ）。現在の表示フェーズが水平同期であれば、HSYNCSビットがアクティブハイになります。

返されるステータスは、LTDC_GCRレジスタで設定されている極性には依存しません。代わりに、現在のアクティブ表示フェーズを返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSYNCS	VSYNCS	HDES	VDES
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **HSYNCS**：水平同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット2 **VSYNCS**：垂直同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット1 **HDES**：水平データ有効表示ステータス
0：アクティブロー
1：アクティブハイ

ビット0 **VDES**：垂直データ有効表示ステータス
0：アクティブロー
1：アクティブハイ

43.7.14 LTDCレイヤx制御レジスタ（LTDC_LxCR）

アドレスオフセット：0x084 + 0x80 * (x-1)(x = 1から2)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
											rw			rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CLUTEN**：カラールックアップテーブルイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：カラールックアップテーブルは無効です。
1：カラールックアップテーブルは有効です。
CLUTは、L8、AL44、AL88ピクセルフォーマットの場合にのみ意味を持ちます。カラールックアップテーブル（CLUT）を参照してください。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **COLKEN**：カラーキーイングイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：カラーキーイングは無効です。
1：カラーキーイングは有効です。

ビット0 **LEN**：レイヤイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：レイヤは無効です。
1：レイヤは有効です。

43.7.15 LTDCレイヤxウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）

アドレスオフセット：0x088 + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0000

このレジスタにより、レイヤ1または2ウィンドウの水平位置（最初と最後のピクセル）を指定します。

ラインの最初の見えるピクセルは、LTDC_BPCRレジスタにおけるAHBP[11:0] ビット + 1のプログラム値です。

ラインの最後の見えるピクセルは、LTDC_AWCRレジスタにおけるAAW[11:0] ビット のプログラム値です。

例：LTDC_BPCRレジスタは0x000E0005（AHBP[11:0] は0xE）に設定され、LTDC_AWCRレジスタは0x028E01E5（AAW[11:0] は0x28E）に設定されます。アクティブデータエリアで水平開始オフセットを5ピクセルにして、サイズ630x460のウィンドウにおける水平位置を設定するには、

- レイヤウィンドウの最初のピクセルWHSTPOS[11:0] を0x14 （0xE+1+0x5）にプログラムする必要があります。
- レイヤウィンドウの最後のピクセルWHSPPOS[11:0] を0x28Aにプログラムする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	WHSPPOS[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **WHSPPOS[11:0]**：ウィンドウ水平停止位置
このビットにより、レイヤウィンドウにおけるラインの最後の見えるピクセルを設定します。
WHSPPOS[11:0] はAHBP[11:0] ビット + 1以上である必要があります（LTDC_BPCRレジスタにプログラム）。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **WHSTPOS[11:0]**：ウィンドウ水平開始位置

このビットにより、レイヤウィンドウにおけるラインの最初の見えるピクセルを設定します。
WHSTPOS[11:0] はAAW[11:0] ビット以下である必要があります（LTDC_AWCRレジスタにプログラム）。

43.7.16 LTDCレイヤxウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）

アドレスオフセット：0x08C + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0000

このレジスタにより、レイヤ1または2ウィンドウの垂直位置（最初と最後のライン）を指定します。

フレームの最初の見えるラインは、LTDC_BPCRレジスタにおける AVBP[10:0] ビット + 1のプログラム値です。

フレームの最後の見えるラインは、LTDC_AWCRレジスタにおける AAH[10:0] ビットのプログラム値です。

例：

LTDC_BPCR レジスタは0x000E0005（AVBP[10:0] は0x5）に設定され、LTDC_AWCR レジスタは0x028E01E5（AAH[10:0] は0x1E5）に設定されます。

アクティブデータエリアで垂直開始オフセットを8ラインにして、サイズ630x460のウィンドウにおける垂直位置を設定するには、

- レイヤウィンドウの最初のライン WVSTPOS[10:0] を0xE（0x5 + 1 + 0x8）にプログラムする必要があります。
- レイヤウィンドウの最後のライン WVSPPOS[10:0] を0x1DAにプログラムする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:16 **WVSPPOS[10:0]**：ウィンドウ垂直停止位置

このビットにより、レイヤウィンドウの最後の見えるラインを設定します。
WVSPPOS[10:0] はAVBP[10:0] ビット + 1以上である必要があります（LTDC_BPCRレジスタにプログラム）。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **WVSTPOS[10:0]**：ウィンドウ垂直開始位置

このビットにより、レイヤウィンドウの最初の見えるラインを設定します。
WVSTPOS[10:0] はAAH[10:0] ビット以下である必要があります（LTDC_AWCRレジスタにプログラム）。

43.7.17 LTDCレイヤxカラーキーイング設定レジスタ（LTDC_LxCKCR）

アドレスオフセット：0x090 + 0x80 * (x - 1)、（x = 1～2）

リセット値：0x0000 0000

このレジスタにより、カラーキーイングによって使用されるカラーキー値（RGB）を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKGREEN[7:0]								CKBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **CKRED[7:0]**：カラーキー、赤の値

ビット15:8 **CKGREEN[7:0]**：カラーキー、緑の値

ビット7:0 **CKBLUE[7:0]**：カラーキー、青の値

43.7.18 LTDCレイヤxピクセルフォーマット設定レジスタ（LTDC_LxPFCR）

アドレスオフセット：0x094 + 0x80 * (x - 1)、（x = 1～2）

リセット値：0x0000 0000

このレジスタにより、レイヤのフレームバッファ内に保存されるデータに使用するピクセルフォーマットを指定します。ピクセルデータはフレームバッファから読み出され、内部フォーマット8888(ARGB) に変換されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **PF[2:0]**：ピクセルフォーマット

このビットにより、ピクセルフォーマットを設定します。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

101：L8（8bit輝度）

110：AL44（4bitアルファ、4bit輝度）

111：AL88（8bitアルファ、8bit輝度）

43.7.19 LTDCレイヤx定数アルファ設定レジスタ（LTDC_LxCACR）

アドレスオフセット：0x098 + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 00FF

このレジスタにより、アルファブレンディングで使用する定数アルファ値（ハードウェアによって255で除される）を指定します。LTDC_LxBFCRレジスタを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **CONSTA[7:0]**：定数アルファ

このビットにより、ブレンディングに使用する定数アルファを設定します。定数アルファは、ハードウェアによって255で除されます。

例：プログラムされている定数アルファが0xFFであれば、定数アルファ値は255/255=1となります。

43.7.20 LTDCレイヤxデフォルト色設定レジスタ（LTDC_LxDCCR）

アドレスオフセット：0x09C + 0x80 * (x-1)(x = 1から2)

リセット値：0x0000 0000

このレジスタにより、フォーマットARGBにおけるレイヤのデフォルト色を指定します。このデフォルト色は指定されたレイヤウィンドウ外や、レイヤが無効になっているときに使用されます。リセット値0x00000000は、透明な黒を表します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DCALPHA[7:0]								DCRED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCGREEN[7:0]								DCBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **DCALPHA[7:0]**：デフォルト色アルファ

このビットにより、デフォルトのアルファ値を設定します。

ビット23:16 **DCRED[7:0]**：デフォルト色赤

このビットにより、デフォルト赤の値を設定します。

ビット15:8 **DCGREEN[7:0]**：デフォルト色緑

このビットにより、デフォルト緑の値を設定します。

ビット7:0 **DCBLUE[7:0]**：デフォルト色青

このビットにより、デフォルト青の値を設定します。

43.7.21 LTDCレイヤxブレンディング係数設定レジスタ（LTDC_LxBFCR）

アドレスオフセット：0x0A0 + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0607

このレジスタにより、ブレンディング係数F1およびF2を指定します。

基本的なブレンディングの計算式は、BC = BF1 x C + BF2 x Cs

- BC = ブレンドされた色
- BF1 = ブレンディング係数1
- C = 現在のレイヤの色
- BF2 = ブレンディング係数2
- Cs = ブレンドされた下側レイヤの色

定数アルファ値は、LTDC_LxCACRのプログラム値をハードウェアにより255で除したものです。

例：レイヤ1のみ有効、BF1を定数アルファに設定。BF2を 1 - 定数アルファに設定LTDC_LxCACRでプログラムされている定数アルファは240（0xF0）です。したがって、定数アルファ値は240 / 255 = 0.94となります。C：現在のレイヤ色は128。
Cs：バックグラウンドカラーは48。レイヤ1はバックグラウンド色とブレンドされます。
BC = 定数アルファ x C + （1 - 定数アルファ）x Cs = 0.94 x 128 + (1- 0.94) x 48 = 123

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2 [2:0]		
					rw	rw	rw						rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **BF1[2:0]**：ブレンディング係数1
このビットにより、ブレンディング係数F1を選択します。
100 ：定数アルファ
110：ピクセルアルファ x定数アルファ
その他：予約済み

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **BF2[2:0]**：ブレンディング係数2
このビットにより、ブレンディング係数F2を選択します。
101：1 - 定数アルファ
111：1 -（ピクセルアルファ x定数アルファ）
その他：予約済み

43.7.22 LTDCレイヤxカラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）

アドレスオフセット：0x0AC + 0x80 * (x - 1)、（x = 1～2）

リセット値：0x0000 0000

このレジスタにより、カラーフレームバッファ開始アドレスを指定します。このアドレスは、フレームバッファでレイヤの左上ピクセルのピクセルデータが保存されているアドレスを示している必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CFBADD[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFBADD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CFBADD[31:0]**：カラーフレームバッファ開始アドレス

このビットにより、カラーフレームバッファ開始アドレスを指定します。

43.7.23 LTDCレイヤxカラーフレームバッファ長レジスタ（LTDC_LxCFBLR）

アドレスオフセット：0x0B0 + 0x80 * (x - 1)、（x = 1～2）

リセット値：0x0000 0000

このレジスタにより、カラーフレームバッファのライン長とピッチを指定します。

例：

- フォーマットRGB565（ピクセル当たり2バイト）、幅256ピクセル（ライン当たりの総バイト数は256 * 2 = 512）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x02000203という値を書き込みます。
- フォーマットRGB888（ピクセル当たり3バイト）、幅320ピクセル（ライン当たりの総バイト数は320 * 3 = 960）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x03C003C3という値を書き込みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CFBP[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CFBLL[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:16 **CFBP[12:0]**：カラーフレームバッファピッチ（バイト単位）

このビットにより、あるピクセルのラインの開始から次のラインの開始までの増分であるピッチをバイト単位で指定します。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **CFBLL[12:0]**：カラーフレームバッファのライン長

このビットにより、バイト単位のピクセルライン長 + 3 を指定します。
ライン長は次のようにして計算します。
アクティブハイの幅 * ピクセル当たりのバイト数 + 3。

43.7.24 LTDCレイヤxカラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）

アドレスオフセット：0x0B4 + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0000

このレジスタにより、カラーフレームバッファ内のライン数を指定します。

このライン数とライン長の設定により、各レイヤに対してフレームごとのデータフェッチ数を指定します。必要なバイト数よりも少なく設定されている場合、FIFOアンダーラン割込みが有効になっていれば、該当する割込みが生成されます。

一方、開始アドレスとピッチ設定により、メモリ内の各ラインの正しい開始位置を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **CFBLNBR[10:0]**：フレームバッファライン数

このビットにより、アクティブハイ幅に相当するフレームバッファ内ライン数を指定します。

43.7.25 LTDCレイヤx CLUT書込みレジスタ（LTDC_LxCLUTWR）

アドレスオフセット：0x0C4 + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0000

このレジスタにより、CLUTアドレスとRGB値を指定します。

CLUT書込みレジスタは、ブランキング期間中またはレイヤが無効になっている場合にのみ設定する必要があります。CLUTは、LTDC_LxCRレジスタで有効または無効にできます。

CLUTは、L8、AL44、AL88ピクセルフォーマットの場合にのみ意味を持ちます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLUTADD[7:0]								RED[7:0]							
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:24 **CLUTADD[7:0]**：CLUTアドレス

このビットにより、各RGB値のCLUTアドレス（CLUT内のカラー位置）を指定します。

ビット23:16 **RED[7:0]**：赤の値
このビットにより、赤の値を設定します。

ビット15:8 **GREEN[7:0]**：緑の値
このビットにより、緑の値を設定します。

ビット7:0 **BLUE[7:0]**：青の値
このビットにより、青の値を設定します。

43.7.26 LTDCレジスタマップ

表429.LTDCレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x008	LTDC_SSCR	Res.	Res.	Res.	Res.	HSW[11:0]												Res.	Res.	Res.	Res.	Res.	VSH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x00C	LTDC_BPCR	Res.	Res.	Res.	Res.	AHBP[11:0]												Res.	Res.	Res.	Res.	Res.	AVBP[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x010	LTDC_AWCR	Res.	Res.	Res.	Res.	AAW[11:0]												Res.	Res.	Res.	Res.	Res.	AAH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x014	LTDC_TWCR	Res.	Res.	Res.	Res.	TOTALW[11:0]												Res.	Res.	Res.	Res.	Res.	TOTALH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x018	LTDC_GCR	HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN	Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	リセット値	0	0	0	0												0		0	1	0		0	1	0		0	1	0				0
0x024	LTDC_SRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBR	IMR
	リセット値																															0	0
0x02C	LTDC_BCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]								BCGREEN[7:0]								BCBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	予約済み	予約済み																															
0x034	LTDC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIE	TERRIE	FUIE	LIE
	リセット値																													0	0	0	0
0x038	LTDC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIF	TERRIF	FUIF	LIF
	リセット値																													0	0	0	0
0x03C	LTDC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRRIF	CTERRIF	CFUIF	CLIF
	リセット値																													0	0	0	0

表429.LTDCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x040	LTDC_LIPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LIPOS[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x044	LTDC_CPSR	CXPOS[15:0]																CYPOS[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	LTDC_CDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSYNCS	VSYNCS	HDES	VDES
	リセット値																													1	1	1	1
0x04C ~0x080	予約済み	予約済み																															
0x084	LTDC_L1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x088	LTDC_L1WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x08C	LTDC_L1WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x090	LTDC_L1CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094	LTDC_L1PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x098	LTDC_L1CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x09C	LTDC_L1DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0	LTDC_L1BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2 [2:0]		
	リセット値																						1	1	0						1	1	1
0x0A4 ~0x0A8	予約済み	予約済み																															
0x0AC	LTDC_L1CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B0	LTDC_L1CFBLR	Res.	Res.	Res.	CFBP[12:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B4	LTDC_L1CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x0B8 ~0x0C0	予約済み	予約済み																															
0x0C4	LTDC_L1CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C8 ~0x100	予約済み	予約済み																															

表429.LTDCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x104	LTDC_L2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x108	LTDC_L2WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x10C	LTDC_L2WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x110	LTDC_L2CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x114	LTDC_L2PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x118	LTDC_L2CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x11C	LTDC_L2DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x120	LTDC_L2BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2 [2:0]		
	リセット値																						1	1	0						1	1	1
0x124 ~0x128	予約済み	予約済み																															
0x12C	LTDC_L2CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x130	LTDC_L2CFBLR	Res.	Res.	Res.	CFBP[12:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x134	LTDC_L2CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x138 ~0x140	予約済み	予約済み																															
0x144	LTDC_L2CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

44 DSIホスト（DSI）

このセクションは、STM32U599/5A9/5F9/5G9デバイスにのみ適用されます。

44.1 概要

Portions Copyright (c) Synopsys, Inc. All rights reserved.掲載許可取得済み。

ディスプレイ･シリアル･インタフェース（DSI）は、MIPI® Allianceによって定義された通信プロトコル･グループの一部です。

MIPI® DSIホストコントローラは、MIPI® DSI仕様で定義されたすべてのプロトコル機能を実装するデジタルコアです。これにより、システムとMIPI® D-PHY間のインタフェースが提供され、ユーザがDSI準拠のディスプレイと通信できるようになります。

44.2 規格および参照資料

- ディスプレイ･シリアル･インタフェース（DSI）に関するMIPI® Alliance仕様
  v1.1 - 2011年11月22日
- ディスプレイ･バス･インタフェース（DBI-2）に関するMIPI® Alliance仕様
  v2.00 - 2005年11月16日
- ディスプレイ･コマンド･セット（DCS）に関するMIPI® Alliance仕様
  v1.1 - 2011年11月22日
- ディスプレイ･ピクセル･インタフェース（DPI-2）に関するMIPI® Alliance仕様
  v2.00 - 2005年9月15日
- 立体ディスプレイ形式（SDF）に関するMIPI® Alliance仕様
  v1.0 - 2011年11月22日
- D-PHYに関するMIPI® Alliance仕様
  v1.1 - 2011年11月7日

44.3 DSIホストの主な機能

- MIPI® Alliance規格に準拠（セクション 44.2：規格および参照資料を参照）
- MIPI® D-PHYとのインタフェース
- MIPI® Alliance仕様で定義されたDCSのすべてのコマンドをサポート：
  - APBインタフェースを経由したすべてのコマンドモード･パケットの送信
  - ビデオ･モード時の低電力かつ高速なコマンド送信
- 最大2つのD-PHYデータ･レーンをサポート
- データ･レーン0を介した双方向通信とエスケープ･モードのサポート
- D-PHYクロック･レーンの非連続クロックをサポートし、さらなる省電力化を実現
- PLLを無効にした超低電力モードをサポート
- ECCおよびチェックサム機能
- 送信終了パケット（EoTp）のサポート
- 障害回復スキーム
- システムインタフェースの設定可能な選択：
  - 制御用AMBA APBおよび汎用コマンドとDCSコマンドのオプションサポート
  - LTDC経由のビデオ･モード･インタフェース
  - LTDC経由のアダプテッド･コマンド･モード･インタフェース
  - ビデオ･モード、アダプテッド･コマンド･モード、APB スレーブにおいて、独立にプログラム可能な仮想チャネルID
- ビデオ･モード･インタフェースの機能：
  - 24bit･インタフェースにマッピングされるLTDCインタフェースのカラー･コーディング：
    - 16bit RGB、設定1、2、3
    - 18bit RGB、設定1、2
    - 24bit RGB
  - すべてのLTDCインタフェース信号の極性をプログラム可能
  - DPI標準の最大解像度800x480ピクセルを超える拡張解像度
  - 最大解像度は、利用可能なDSI物理リンク帯域幅によって制限されます。
    - レーン数：2
    - レーンあたり最大速度：500 M ビット/秒
    - セクション 44.4.3：サポートされている解像度およびフレームレートの例を参照。
- アダプテッド･インタフェースの機能：
  - memory_write_start（WMS）およびmemory_write_continue（WMC）DCSコマンドによる大容量データ送信のサポート
  - 24bit･インタフェースにマッピングされるLTDCインタフェースのカラー･コーディング：
    - 16bit RGB、設定1、2、3
    - 18bit RGB、設定1、2
    - 24bit RGB
- ビデオ･モード･パターン･ジェネレータの機能：
  - LTDC刺激なしの垂直および水平カラーバーの生成
  - LTDC刺激なしのBERパターン

44.4 DSIホストの機能説明

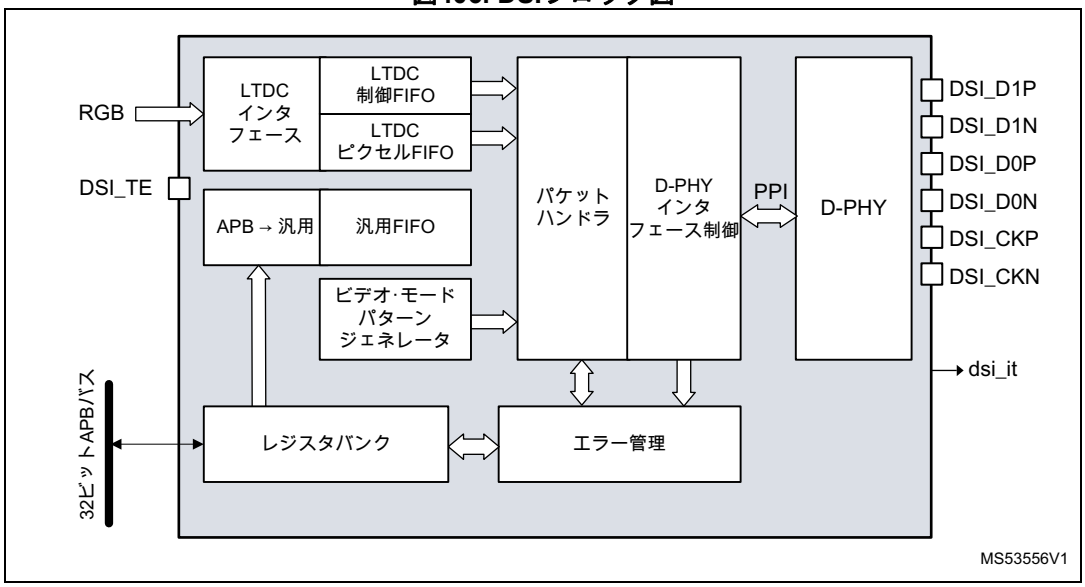
44.4.1 概要

MIPI® DSIホストには、LTDCに内部接続された専用ビデオ･インタフェースと、ディスプレイへの情報伝達に使用できる汎用APBインタフェースがあります。詳しくは以下のとおりです。

- LTDCインタフェース：
  - ビデオ･モードで情報を伝達するために使用され、ホスト･プロセッサからペリフェラルへの転送がリアルタイムのピクセル･ストリームの形式（DPI）となります。
  - カスタマイズされたモードにより、このインタフェースは、アダプテッド･コマンド･モード（DBI）で全帯域幅での情報伝達に使用できます。
- APBスレーブ･インタフェース：コマンド･モードで一般的な情報を送信でき、独自のレジスタ･インタフェースに従います。このインタフェースはビデオ･モードまたはアダプテッド･コマンド･モードのいずれかのLTDCインタフェースと同時に動作できます。
- ビデオ･モード･パターン･ジェネレータ：水平/垂直カラーバーやD-PHYのBERテスト･パターンを何らの刺激もなしに送信できます。

図406にブロック図を示します。

図406. DSIブロック図



44.4.2 DSIホストのピンと内部信号

表430および 表431に一覧、それぞれ、DSIのピン（オルタネート機能）と内部入出力信号を示します。

表430.DSIピン

信号名	信号タイプ	説明
DSI_D0P/D0N	入力／出力	差動データ･レーン0
DSI_D1P/D1N	出力	差動データ･レーン1
DSI_CKP DSI_CKN	出力	差動クロック
DSI_TE	入力	DSIティアリング効果ピン

表431.DSI内部入力／出力信号

信号名	信号タイプ	説明
dsi_it	出力	DSIグローバル割込み

44.4.3 サポートされている解像度およびフレームレート

DSI仕様では、サポートされる標準解像度やフレームレートは定義されていません。ディスプレイ解像度、ブランキング期間、同期イベント時間、フレームレート、ピクセル色深度は、必要な帯域幅を決める基本的な要因となります。さらに、リンクに関連するその他の属性によって、DSI固有のデバイスをサポートするリンクの能力が影響を受ける可能性があります。すなわち、ディスプレイ入力バッファリング容量、ビデオ送信モード（バーストまたは非バースト）、バス･ターンアラウンド（BTA）時間、ビデオ･モード送信における同時コマンド･モード･トラフィック、ディスプレイ･デバイスの仕様などの属性です。これらすべての変数が関わるため、特定のディスプレイ･デバイスをサポートする最小レーン･レートと最小レーン数を見積もる標準的な手順を定義することが難しくなっています。

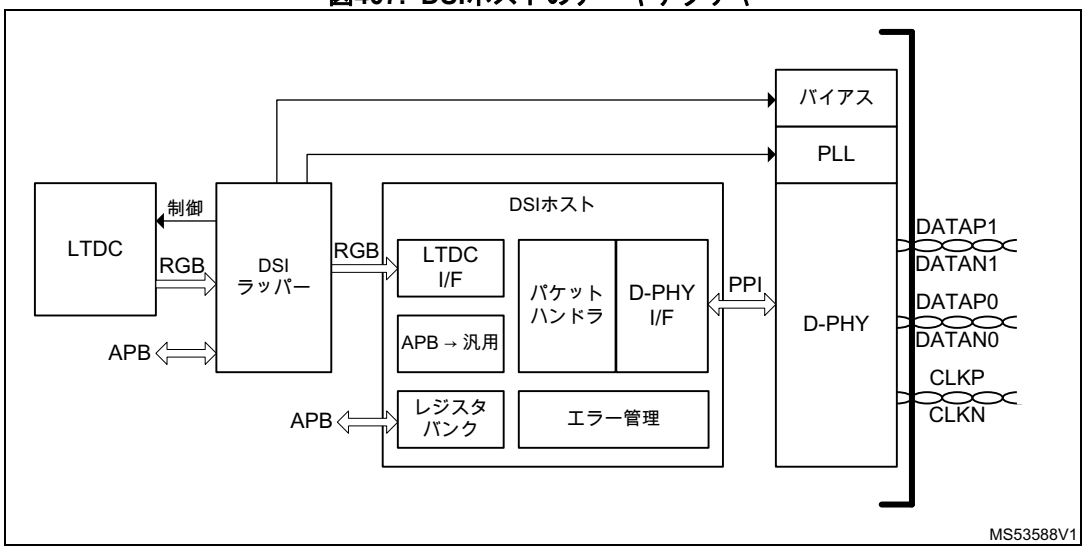
見積りの基本的な前提は以下のとおりです。

- クロック･レーンの周波数は 250MHzであり、各データ･レーンの帯域幅は 500Mbit/sとなります。
- ディスプレイは、DSIリンクで配信される速度でピクセルデータをバッファリングできなければなりません。
- ピクセルデータが送信されているとき、リンク上に重要な制御トラフィックは存在しないとします。

44.4.4 システム･レベルのアーキテクチャ

図407に、DSIホストのアーキテクチャを示します。

図407. DSIホストのアーキテクチャ



それぞれの部分の機能は以下のとおりです。

- DSIラッパーは、LTDCとDSIホストカーネル間のインタフェースを確保します。そして、カラーモード、信号極性を適応させることができ、アダプテッド･コマンド･モードでフレームバッファを自動更新するためのティアリング効果（TE）管理を行います。またDSIラッパーはDSI バイアス、DSI PLL、およびMIPI® D-PHYの特定機能も制御します。
- LTDCインタフェースは、LTDCからデータと制御信号を取り込み、ビデオ制御信号用のFIFOとピクセルデータ用の別のFIFOに伝えます。このデータは、以下のいずれかを作成するために使われます。
  - ビデオ･モードの場合のビデオ･パケット（セクション44.5 を参照）
  - アダプテッド･コマンド･モードの場合の memory_write_start と memory_write_continue のDCSコマンド（セクション44.6 を参照）
- レジスタバンクは、標準的なAMBA-APBスレーブ･インタフェースからアクセス可能で、DSIホストの設定や制御用のレジスタへのアクセスを提供します。また、特定のイベントをシステムに通知するための、完全にプログラム可能な割込みジェネレータもあります。
- PHYインタフェース制御は、D-PHYインタフェースの管理を担当します。現在の動作を確認し、低電力送受信やハイスピード送信を可能にします。また、ハイスピード送信のために、利用可能な複数のD-PHYレーン間でデータ分割を行います。
- パケットハンドラは、リンク内の動作をスケジューリングします。これは、現在動作しているインタフェースと、使用されているビデオ送信モード（バースト･モード、または同期パルスもしくは同期イベントでの非バースト･モード）に基づいて、いくつかの機能を実行します。対応するECCおよびCRCコードを生成してロングパケットまたはショートパケットを作成します。このブロックは、以下の機能も実行します。
  - パケットの受信
  - ECCチェックによるパケットヘッダの検証
  - ヘッダーの修正とシングル･ビットエラーの通知
  - 受信終了
  - 複数ヘッダー･エラーの通知

  - 受信パケットの仮想チャネルに応じて、ハンドラは出力データをそれぞれのポートにルーティングします。
- APB-to-generic ブロックは、APB 操作を汎用コマンドを保持する FIFO に橋渡しします。このブロックは、次のFIFOとのインタフェースを図ります。
  - コマンドFIFO
  - 書込みペイロードFIFO
  - 読出しペイロードFIFO
- エラー管理は、DSIリンク上のエラー状態を通知し、監視します。これは、タイムアウト状態が発生したかどうかを判断するために使用されるタイマを制御し、内部ソフトリセットを実行し、割込み通知をトリガします。

44.5 機能説明：LTDCインタフェースのビデオ･モード

LTDCインタフェースはデータと制御信号をキャプチャし、DSIリンクに送信するFIFOインタフェースに伝達します。

インタフェースには、ビデオ制御信号とピクセルデータという2種類のデータ･ストリームが存在します。インタフェースのカラー･コーディングに応じて、ピクセルデータはLTDCバス全体で異なる配置になります。

インタフェースのピクセル･カラー･コーディングを 表432にまとめます。

表432. LTDCインタフェースのカラー･コンポーネントの位置

場所	16bit			18bit		24bit
	設定1	設定2	設定3	設定1	設定2	
D23	-	-	-	-	-	R[7]
D22	-	-	-	-	-	R[6]
D21	-	-	R[4]	-	R[5]	R[5]
D20	-	R[4]	R[3]	-	R[4]	R[4]
D19	-	R[3]	R[2]	-	R[3]	R[3]
D18	-	R[2]	R[1]	-	R[2]	R[2]
D17	-	R[1]	R[0]	R[5]	R[1]	R[1]
D16	-	R[0]	-	R[4]	R[0]	R[0]
D15	R[4]	-	-	R[3]	-	G[7]
D14	R[3]	-	-	R[2]	-	G[6]
D13	R[2]	G[5]	G[5]	R[1]	G[5]	G[5]
D12	R[1]	G[4]	G[4]	R[0]	G[4]	G[4]
D11	R[0]	G[3]	G[3]	G[5]	G[3]	G[3]
D10	G[5]	G[2]	G[2]	G[4]	G[2]	G[2]
D9	G[4]	G[1]	G[1]	G[3]	G[1]	G[1]
D8	G[3]	G[0]	G[0]	G[2]	G[0]	G[0]
D7	G[2]	-	-	G[1]	-	B[7]
D6	G[1]	-	-	G[0]	-	B[6]
D5	G[0]	-	B[4]	B[5]	B[5]	B[5]
D4	B[4]	B[4]	B[3]	B[4]	B[4]	B[4]
D3	B[3]	B[3]	B[2]	B[3]	B[3]	B[3]
D2	B[2]	B[2]	B[1]	B[2]	B[2]	B[2]
D1	B[1]	B[1]	B[0]	B[1]	B[1]	B[1]
D0	B[0]	B[0]	-	B[0]	B[0]	B[0]

LTDCインタフェースは、柔軟性を高めて複数のシステムでこのインタフェースを正しく使用できるように設定できます。以下の設定オプションが利用できます。

- 極性の制御 : すべての制御信号は、LTDCの設定に応じて極性を変えるようにプログラム可能です。
- コアのリセットの後、DSI ホストは最初のVSYNC のアクティブ遷移を待ってピクセルデータを含む信号のサンプリングを開始するため、フレームの途中から画像データの送信を開始することを避けることができます。
- インタフェースのピクセルのカラー･コーディングが18bitで、18bitの軽度パック済みストリームが無効の場合、VPSIZEフィールドにプログラムされるピクセル数は4の倍数でなければなりません。つまり、このモードでは、構成内の2つのLSBは常にゼロと推測されます。仕様では、このモードではピクセル･ラインのサイズは4の倍数でなければならないと記載されています。
- FIFOのアンダーフローとオーバーフローを回避するために、設定されたピクセル数が常にLTDCから受信されると仮定します。
- パケットのスケジューリングに関してメモリを整理するために、パケットあたりのピクセル数のパラメータを使用して、異なるビデオ･パケットのメモリ空間を別にします。

SHTDNとCOLMのサンプリングと送信のためには、LTDCからのビデオ･ストリーミングがアクティブである必要があります。つまり、LTDCがVSYNCやHSYNCのようなビデオ信号をアクティブに生成していない場合、これらの信号はDSIリンクを通じて伝送されません。このような制約があるため、そしてコマンドを正しく伝送するためには、コマンドのサンプリングと伝送のために最初のVSYNCアクティブパルスが発生する必要があります。ディスプレイをシャットダウンする場合、コマンド発行後、1フレームの間はLTDCをアクティブにしておく必要があります。これにより、LTDCインタフェースで実際にビデオの生成を無効にする前に、コマンドが正しく送信されることが保証されます。

SHTDNとCOLMの値は、DSIラッパー制御レジスタ（DSI_WCR）に設定できます。

すべてのデータ型において、クロックサイクルごとに1ピクセル全体が受信されます。ペイロードのピクセル数は、表433に示した値の倍数に制約されます。

表433.各データ型におけるピクセル単位のペイロードサイズに対する倍数

値	データ型
1	16bit 18bit軽度パック済み 24bit
2	軽度パック済みピクセル･ストリーム
4	18bit非軽度パック済み

44.5.1 ビデオ送信モード

ビデオ送信モードには、以下の種類があります。

- バースト･モード
- 非バースト･モード
  - 同期パルスでの非バースト･モード
  - 同期イベントでの非バースト･モード

バースト･モード

このモードでは、アクティブなピクセル･ライン全体をFIFOにバッファリングし、割込みなしの1つのパケットで送信します。この送信モードでは、DPIピクセルFIFOにアクティブなピクセルデータのライン全体を格納できる容量が必要になります。このモードは、ピクセルに必要な帯域幅とDSIリンク帯域幅の差が大きい場合に適しており、このモードによって、DSIホストは1回のデータ･バーストでアクティブ･ビデオ･ライン全体を素早くディスパッチして、低電力モードに戻ることができます。

非バースト･モード

このモードでは、プロセッサはDSIホストのパーティション分割特性を利用して、ビデオ･ライン送信を複数のDSIパケットに分割します。この処理は、ピクセルに必要な帯域幅をDSIリンクの帯域幅と合わせるために実行されます。このモードでは、コントローラ構成はピクセルデータのライン全体をLTDCインタフェースのピクセルFIFO内に格納する必要はありません。必要なのは1つのビデオ･パケットの内容だけです。

バースト･モードまたは非バースト･モードを選択するガイドライン

バースト･モードと非バースト･モードの選択は、主にシステム構成とデバイスの要件に依存します。アプリケーションのシナリオに合ったビデオ送信モードを選択してください。バースト･モードでは、リンクはより多くの時間を低電力モードで済ませる確率が高まり、消費電力が減少するため、同モードのほうが有益です。バースト･モードの動作から最大の便益を得るためには、以下の条件が満たされている必要があります。

- DSI ホスト･コアは、内部 FIFO のオーバーフローを避けるために、ピクセル･ライン全体を格納するのに十分なピクセル･メモリを持つ必要があります。
- ディスプレイ･デバイスは、受信バッファでのオーバーフローを避けるために、1つのパケット･バーストでピクセル･ライン全体の受信をサポートする必要があります。
- DSI出力帯域幅は、リンクがラインごとに1回低電力に移行できるようにするために、LTDCインタフェースの入力帯域幅より高くする必要があります。

システムがこれらの要件を満たせない場合、バースト･モードの使用中にピクセルデータが失われ、ディスプレイ･デバイスの誤動作が発生する可能性があります。これらのエラーは、一時的なピクセルデータを保存するシステムの能力に関係しています。

バースト･モードを使用するための条件をすべては満たせない場合は、エラーを避けるために非バースト･モードを使用してください。非バースト･モードでは、以下のことが可能になるため、ピクセル送信の速度を合わせやすくなります。

- 特定数のピクセルだけメモリに格納し、ピクセル･ライン全体を必要としません（DSI ホストでのLTDCインタフェースのRAM要件を軽減します）。
- 少量のピクセル（ピクセル･ライン全体より少ない）バッファリングしかサポートしていないデバイスで動作します。

DSI非バースト･モードは、DSI出力ピクセル比率がLTDCインタフェースの入力ピクセル比率と一致するように設定する必要があり、ホスト側とデバイス側の両方でメモリ要件が軽減されます。この処

理は、ピクセル･ラインを複数のピクセル･チャンクに分けて、オプションでnullパケットを使ってインタリーブすることで行います。

以下の式は、DSIリンクのピクセル出力比率（「=」記号の左側）とLTDCインタフェース･ピクセル入力（「=」記号の右側）を一致させるために、DSIホスト･コアの送信パラメータを非バースト･モードでどのようにプログラムする必要があるかを示しています。

nullパケットが有効な場合：

lanebyteclkperiod × NUMC（VPSIZE × bytes_per_pixel + 12 + NPSIZE）/ number_of_lanes

= pixels_per_line × LTDC_Clock_period

nullパケットが無効な場合：

lanebyteclkperiod × NUMC（VPSIZE × bytes_per_pixel + 6 ）/ number_of_lanes

= pixels_per_line × LTDC_Clock_period

44.5.2 ビデオ･モードでのLTDCインタフェース設定の更新

現在のフレームに影響を与えることなくLTDCインタフェースの設定を動作中に更新することができます。これは、シャドウレジスタの助けによって行われます。この機能は、DSIホスト･ビデオ･シャドウ制御レジスタ（DSI_VSCR）によって制御されます。

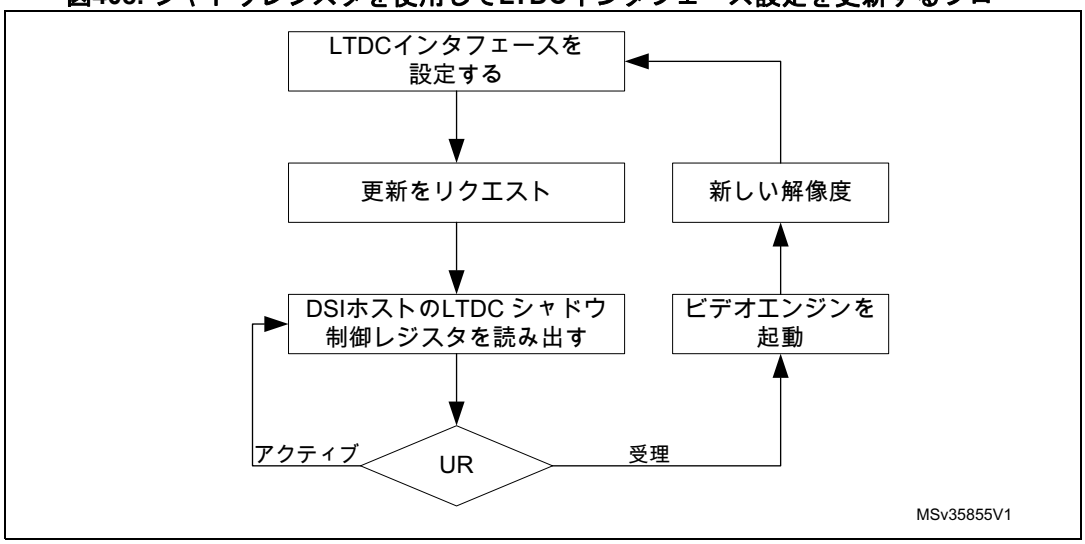
新しい設定は、システムが要求したときのみ使用されます。ビデオフレームの送信中にビデオ設定を更新するには、そのフレームの設定を補助レジスタに格納する必要があります。こうすることで、現在のフレームを破損することなく、APBインタフェースを通じて新しいフレーム設定を設定することができます。

デフォルトでは、この機能は無効です。この機能を有効にするには、DSIホスト･ビデオ･シャドウ制御レジスタ（DSI_VSCR）のイネーブル（EN）ビットを1にセットします。

この機能が有効になっていると、補助レジスタに格納されている設定がシステムによって適用されます。

図408 に、LTDCインタフェースの設定を更新するのに必要な手順を示します。

図408. シャドウレジスタを使用してLTDCインタフェース設定を更新するフロー

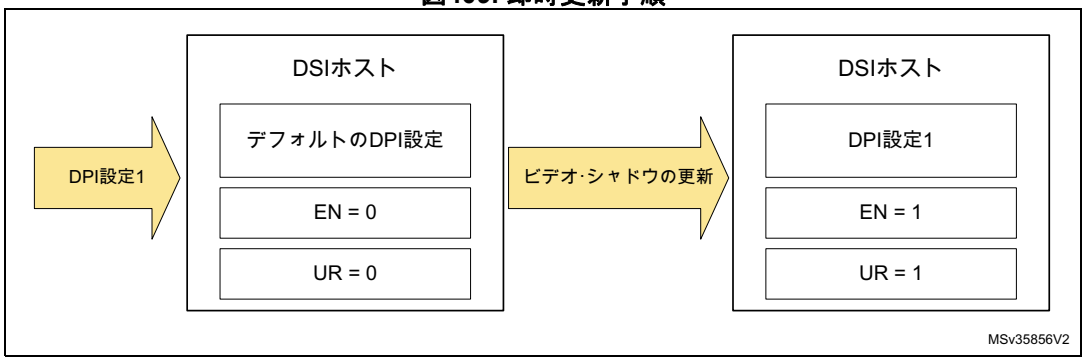


即時の更新

シャドウレジスタ機能がアクティブの場合、ビデオエンジンが起動する前に補助レジスタにLTDCを設定する必要があります。つまり、リセット後、すぐに更新レジスタ（UR）ビットが付与されます。

リセットなしでアクティブなレジスタを即座に更新する必要がある場合（図409のように）、DSIホスト·ビデオ·シャドウ制御レジスタ（DSI_VSCR）のイネーブル（EN）ビットと更新レジスタ（UR）ビットが0にセットされていることを確認します。

図409. 即時更新手順

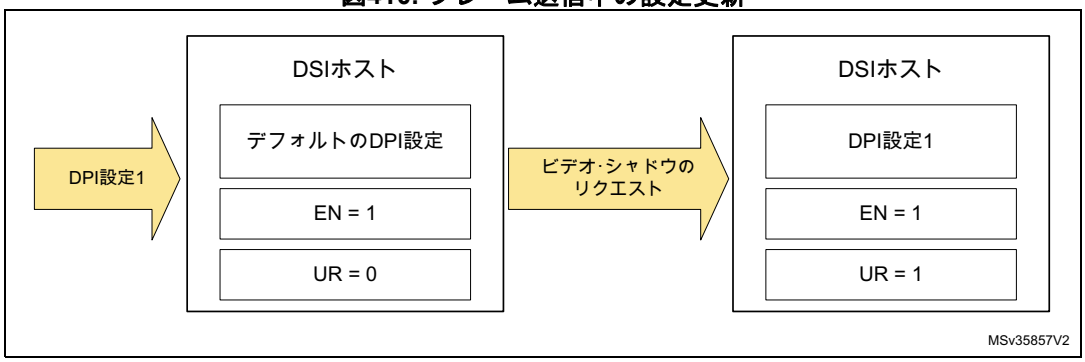


APBを使用したフレーム送信中の設定更新

LTDCインタフェースの設定を更新するには、図410に示す手順に従います。

1. DSIホスト·ビデオ·シャドウ制御レジスタ（DSI_VSCR）のイネーブル（EN）ビットが1にセットされていることを確認します。
2. DSI ホスト·ビデオ·シャドウ制御レジスタ（DSI_VSCR）の更新レジスタ（UR）ビットを 1 にセットします。
3. 更新レジスタ（UR）ビットを監視します。このビットは、更新が完了すると0にセットされます。

図410. フレーム送信中の設定更新



設定更新のリクエスト

フレームのどの部分でも、LTDCインタフェースの設定更新をリクエストすることができます。DSIホストはフレームの終わりまで待ってから設定を変更します。しかし、フレームの最初のラインで更新リクエストを送信すると、データはクロック·ドメイン間を伝搬する必要があるため、最初のラインでは避けてください。

44.6 機能説明：LTDCインタフェースのアダプテッド･コマンド･モード

アダプテッド･コマンド･モードによって、システムは、コマンド･モード送信を使用して（DCSパケットを使用して）DSIホストによって伝達されるLTDCからのピクセルのストリームを入力することができます。アダプテッド･コマンド･モードは、ピクセル入力制御レート信号とティアリング効果報告メカニズムもサポートしています。

アダプテッド･コマンド･モードでは、*memory_write_start*（WMS）および *memory_write_continue*（WMC）DCSコマンドによって大容量のデータを送信することができるようになります。これは、MIPI® ディスプレイにコマンド･モードで送信されるメモリ書込み動作や、高解像度ディスプレイでの大面積ピクセルのリフレッシュのために、より広いデータ帯域幅を提供するのに役立ちます。ディスプレイ設定コマンド、読戻しコマンド、ティアリング効果初期化などの追加コマンドを転送する場合は、アダプテッド･コマンド･モード機能を補完するためにAPBスレーブ汎用インタフェースを使用する必要があります。

アダプテッド･コマンド･モードの動作は16 bpp、18 bpp、24 bppのRGBに対応しています。

アダプテッド･コマンド･モードで画像データを送信するには、

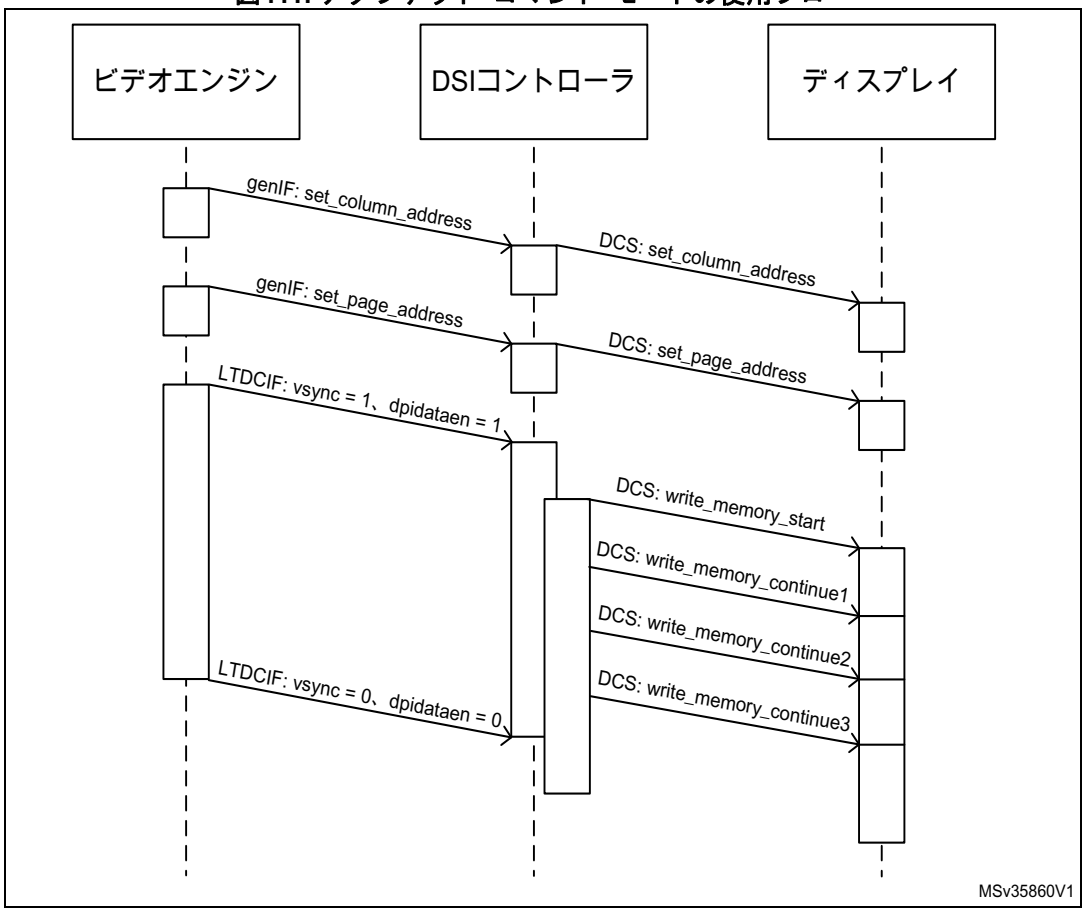
- DSIホスト･モード設定レジスタ（DSI_MCR）のコマンド･モード（CMDM）ビットを1にセットします。
- DSIラッパー設定レジスタ（DSI_WCFGR）でDSIモード（DSIM）ビットを1にセットします。

画像データを送信するには、以下の手順に従います。

- set_column_addressおよび set_page_address DCSコマンドを使って、リフレッシュする画像領域を定義します。画像領域は一度だけ定義する必要がありますが、異なる値が定義されるまで有効です。
- DSI ホスト LTDC カラー･コーディング･レジスタ（DSI_LCOLCR）のカラー･コーディング（COLC）フィールドを使用して、使用するピクセルのカラー･コーディングを定義します。
- DSIホストLTDC VCIDレジスタ（DSI_LVCIDR）の仮想チャネルID（VCID）フィールドを使用して、LTDCインタフェースが生成するパケットの仮想チャネルIDを定義します。これらも一度だけ定義する必要があります。
- DSI_WCRレジスタのLTDCイネーブル（LTDCEN）ビットをセットして、LTDCからのデータ送信を開始します。

図411に、アダプテッド･コマンド･モードの使用フローを示します。

図411. アダプテッド･コマンド･モードの使用フロー



DSIホスト･モード設定レジスタ（DSI_CFGR）のコマンド･モード（CMDM）ビットが1にセットされていると、LTDCインタフェースはアダプテッド･コマンド･モードに対応する動作を行います。

このモードでは、ホストプロセッサはLTDCインタフェースを使用して、ペリフェラルのローカル･フレームバッファに書き込まれるピクセルの連続ストリームを送信することができます。ペリフェラルは、ピクセル入力バスを使用してピクセルを受け取り、フローを自動的に制御して連続ピクセルのストリームを制限します。最初のピクセルが受信されると、DSIホストLTDCコマンド設定レジスタ（DSI_LCCR）のコマンド･サイズ（CMDSIZE）フィールドの現在値が、内部インタフェース機能にシャドウイングされます。このインタフェースは、有効なピクセルがインタフェースを通じて入力されるたびにカウンタをインクリメントします。このピクセル･カウンタがコマンド･サイズ（CMDSIZE）に達すると、コマンドがコマンドFIFOに書き込まれ、DSIリンクを通じてパケットを送信する準備が整います。

カウンタがシャドウイングされたコマンド･サイズ（CMDSIZE）の値に達する前に最後のピクセルが到着した場合、ワード･カウント（WC）がカウンタの値に対応するバイト量に設定されたWMSコマンドがコマンドFIFOに発行されます。CMDSIZE（シャドウ値）以上のピクセルを受信した場合、コマンド･サイズ（CMDSIZE）に対応するバイト数にWCが設定されてWMSコマンドがコマンドFIFOに送信され、カウンタはリスタートします。

最初のWMSコマンドがFIFOに書き込まれた後、回路は同様の動作をしますが、WMSコマンドの代わりにWMCコマンドを発行します。この処理は、画像の最後のピクセルが受信されるまで繰り返されます。コアは、画像の最後のピクセルが受信されて、下回るか、あるいはコマンド･サイズ（CMDSIZE）の上限に達すると、自動的に新しいパケットの送信を開始します。

LTDCとの同期

DSIラッパーは、以下のようにして、転送プロセスの同期を図ります。

- LTDCの開始/停止を制御する。
- LTDCとDSIホスト間のデータフロー制御を行う。

ディスプレイ･フレームバッファのリフレッシュへの移行は、以下のようにしてトリガするこすることができます。

- DSIラッパー制御レジスタ（DSI_WCR）のLTDCイネーブル（LTDCEN）ビットを手動でセットする。
- ティアリング効果（TEIF）イベントが発生し、自動リフレッシュ（AR）が有効になったとき、自動的に。

手動モードと自動モードの選択は、DSIラッパー設定レジスタ（DSI_WCFGR）の自動リフレッシュ（AR）ビットによって行われます。

手動リフレッシュ･モードでも自動リフレッシュ･モードでも、1フレームの転送が完了すると、DSIラッパーはTFTディスプレイ･コントローラ（LTDC）を停止させ、DSIラッパー制御レジスタ（DSI_WCR）のLTDCイネーブル（LTDCEN）ビットをリセットし、DSIラッパー･ステータスレジスタ（DSI_WSR）のリフレッシュ終了割込みフラグ（ERIF）をセットします。DSIラッパー設定レジスタ（DSI_WCFGR）のリフレッシュ終了割込みイネーブル（ERIE）ビットが設定されていれば、割込みが発生します。

DSIラッパー･ステータスレジスタ（DSI_WSR）のリフレッシュ終了割込みフラグ（ERIF）は、DSIラッパー割込みフラグ･クリアレジスタ（DSI_WCIFR）のリフレッシュ終了割込みフラグ･クリア（CERIF）ビットをセットすることで、リセットできます。

DSIラッパーによるTFTディスプレイ･コントローラー（LTDC）の停止は、DSIラッパー設定レジスタ（DSI_WCFGR）のVSync極性（VSPOL）ビットに従って、VSyncの立ち上がりエッジまたは立ち下がりエッジに同期して行われます。

ティアリング効果のサポート

DSI仕様では、コマンド･モードのディスプレイでのティアリング効果機能がサポートされています。これにより、ホスト･プロセッサは、ディスプレイ･ペリフェラルがフレームバッファの内容を読み取るプロセスにおいてどの位置にあるかについての正確なタイミング情報を受け取れるようになります。

ティアリング効果は以下によって管理できます。

- DSI仕様では記載されていない個別のピン
- DSIティアリング効果機能 : *set_tear_on* DCSコマンドは、汎用インタフェース･レジスタを使用してAPBインタフェースを通じて発行する必要があります。

GPIO経由のティアリング効果

DSIラッパー設定レジスタ（DSI_WCFGR）のティアリング効果ソース（TESRC）ビットが設定されていると、GPIOを経由してティアリング効果が通知されます。

入力信号の極性は、DSIラッパー設定レジスタ（DSI_WCFGR）のティアリング効果極性（TEPOL）ビットによって設定できます。

プログラムされたエッジが検出されると、DSIラッパー割込みおよびステータスレジスタ（DSI_WISR）のティアリング効果割込みフラグ（TEIF）ビットがセットされます。

DSIラッパー割込みイネーブル･レジスタ（DSI_WIER）のティアリング効果割込みイネーブル（TEIE）ビットがセットされていれば、割込みが発生します。

DSIリンク経由のティアリング効果

DSIラッパー設定レジスタ（DSI_WCFGR）のTESRCビットがリセットされていると、ティアリング効果はDSIリンクを経由して管理されます。

DSIホストは、set_tear_on コマンドを送信した後、2回のバス･ターンアラウンド（BTA）を実行してリンクの所有権をDSIディスプレイに与えます。ディスプレイは、D-PHYトリガイベントによってDSIホストに示されるティアイベントが発生するまで、バスの所有権を保持します。その後、DSIホストはトリガをデコードし、DSIラッパー割込みおよびステータスレジスタ（DSI_WISR）のティアリング効果割込みフラグ（TEIF）ビットをセットしてイベントを報告します。

DSIラッパー割込みイネーブル･レジスタ（DSI_WIER）のティアリング効果割込みイネーブル（TEIE）ビットがセットされていれば、割込みが発生します。

この機能を使用するには、WMSおよびWMC DCSコマンドを使用してディスプレイを更新した後、set_tear_on コマンドを発行する必要があります。この手順によって、ディスプレイが新しいフレーム更新を受信する準備ができるまで、DSIリンクが停止されます。

DSIホストは、柔軟性のために、WMS/WMCシーケンスの後に自動的にはティアリング効果リクエスト（ダブルBTA）を生成しないため、ディスプレイの複数の領域を更新することができ、DSI帯域幅の使用状況を改善できます。ティアリング効果のリクエストは、必ずDSIホストに実装されているset_tear_on コマンドによってトリガする必要があります。

ティアリング効果を有効にするには、次のレジスタを設定します。

- DSIホスト･コマンド･モード設定レジスタ（DSI_CMCR）: TEARE
- DSIホスト･プロトコル設定レジスタ（DSI_PCR）: BTAE

44.7 機能説明：APBスレーブ汎用インタフェース

APBスレーブ･インタフェースは、コマンド･モードで一般的な情報の送信を可能にし、独自のレジスタ･インタフェースに従います。このインタフェースを通じて送信されるコマンドは、DCS仕様に準拠するという制約を受けず、これにはDSI仕様にメーカー固有として記述されている汎用コマンドを含めることができます。

DSIホストは、DSI仕様に記述されているように、書込みと読出しのコマンド･モード･パケットの送信をサポートしています。これらのパケットはAPBレジスタのアクセスで構築されます。DSIホスト汎用ペイロード･データ･レジスタ（DSI_GPDR）には、操作に基づいた2つの異なる機能があります。このレジスタに書き込むと、コマンド･モード･パケットの送信時にデータがペイロードとして送信されます。このレジスタを読み出すと、読戻し操作のペイロードが返されます。DSIホスト汎用ヘッダ設定レジスタ（DSI_GHCR）には、コマンド･モードのパケット のヘッダタイプとヘッダデータが格納されます。このレジスタへ書き込むと、パケットの送信がトリガされます。これは、ロング･コマンド･モード･パケットの場合、パケットのペイロードをDSIホスト汎用ペイロード･データ･レジスタ（DSI_GPDR）に事前に書き込む必要があることを意味します。

汎用インタフェースを介して送信できる有効なパケットは以下のとおりです。

- 汎用書込みショートパケット0パラメータ
- 汎用書込みショートパケット1パラメータ
- 汎用書込みショートパケット2パラメータ
- 汎用読出しショートパケット0パラメータ
- 汎用読出しショートパケット1パラメータ
- 汎用読出しショートパケット2パラメータ
- 最大読出しパケット設定
- 汎用ロング書込みパケット
- DCS書込みショートパケット0パラメータ
- DCS書込みショートパケット1パラメータ
- DCS読出しショートパケット0パラメータ
- DCS書込みロング･パケット

DSIホスト汎用パケット･ステータスレジスタ（DSI_GPSR）の一連のビットは、APBインタフェースのサポートに関連するFIFOのステータスを報告します。

汎用インタフェースのパケットは、常にDSI送信モード（ビデオ･モードまたはコマンド･モード）のいずれかを使用して転送されます。これらのモードのどちらも選択されていない場合は、パケットはリンクを通じて送信されず、関連するFIFOは最終的にオーバーフローします。

44.7.1 汎用インタフェースを使ったパケットの送信

APBバスを介したパケットの転送は、以下の条件に基づいて行われます。

- APBプロトコルでは、書込みと読出しの手順はそれぞれ2クロックサイクルで実行されることが定義されている。つまり、APBインタフェースを介した最大入力データレートは、常にAPBクロックの半分の速度となります。
- データ入力バスは、最大 32bit 幅である。これにより、入力 APB クロック周波数と APB インタフェースで達成可能な最大ビットレートとの関係を定義することができます。
- APB のみを使用する場合の DSI リンクのピクセル･ビットレートは、（APB クロック周波数）× 16Mbit/sである。
- APBインタフェースのみを使用する場合、理論上のDSIリンク最大ビットレートは、DSIリンク最大ビットレート = APBクロック周波数（MHz）× 32 / 2Mbit/sで表すことができる。この式に

おいて、数値32はAPBデータバス幅を表し、2による除算が存在するのは、各APB書込み手順が実行されるのに2クロックサイクルを要するからです。

- 帯域幅はAPBクロック周波数に依存し、利用可能な帯域幅はクロック周波数とともに増加する。

DSIリンクによって伝送される高帯域幅のコマンド･モード･トラフィックを達成するためにAPBインタフェースを駆動するには、DSIホストはコマンド･モードのみで動作し、APBインタフェースは現在使用されている唯一のデータソースでなければなりません。したがって、APBインタフェースはDSIリンクの全帯域幅を保持し、他の入力インタフェース･ソースと帯域幅を共有することはありません。

メモリ書込みコマンドは、DSIリンクによって伝達される最も多くのデータを含むため、APBインタフェースに最大のスループットを要求します。パケット情報を書き込む際、まずDSIホスト汎用ペイロード･データレジスタ（DSI_GPDR）を使って、特定のパケットのペイロードをペイロードFIFOに書き込みます。ペイロードデータがコマンドパラメータの場合、送信する最初のバイトをAPBデータバスの最下位バイト位置に配置します。

ペイロードを書き込んだ後、パケットヘッダをコマンドFIFOに書き込みます。

ペイロードデータがメモリ書込みコマンド用である場合、それにはピクセル情報が含まれており、DCS仕様の付属書Aで言及されている、ピクセルからバイトへの変換構成に従う必要があります。

図 412～416 は、APBデータ書込みバス内でピクセルデータをどのように構成する必要があるかを示しています。

メモリ書込みコマンドは、DSIパケットにカプセル化されたDCSロング･パケットで伝達されます。DSIでは、DCSコマンドがパケットの最初のペイロード･バイトに存在しなければならないと規定されています。このことは各図にも示されています。図412 ～ 416において、write memory コマンドはDCSコマンドの write memory Start と write memory Continue に置き換えることができます。

図412. 24 bppのAPBピクセルからバイトへの構成

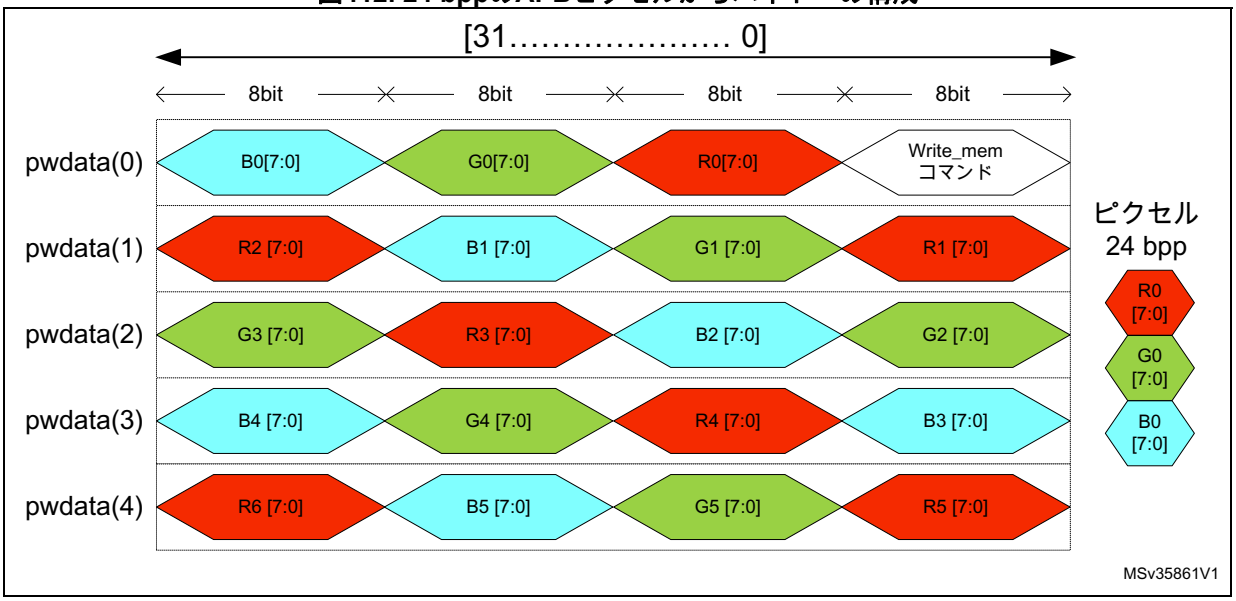


図413. 18 bppのAPBピクセルからバイトへの構成

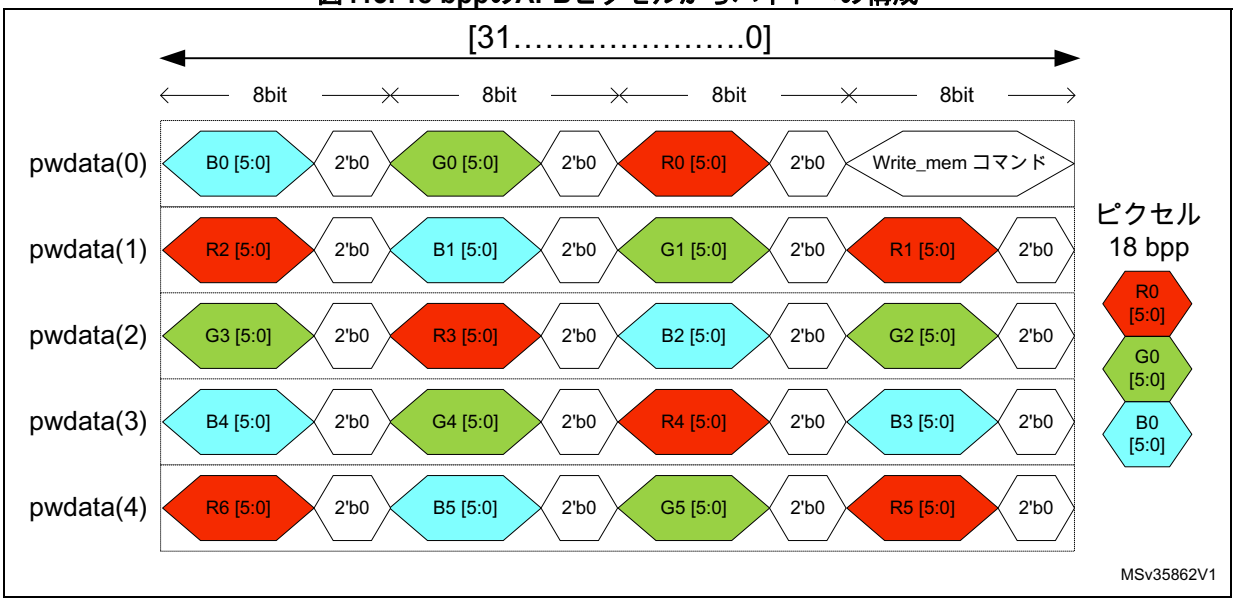


図414. 16 bppのAPBピクセルからバイトへの構成

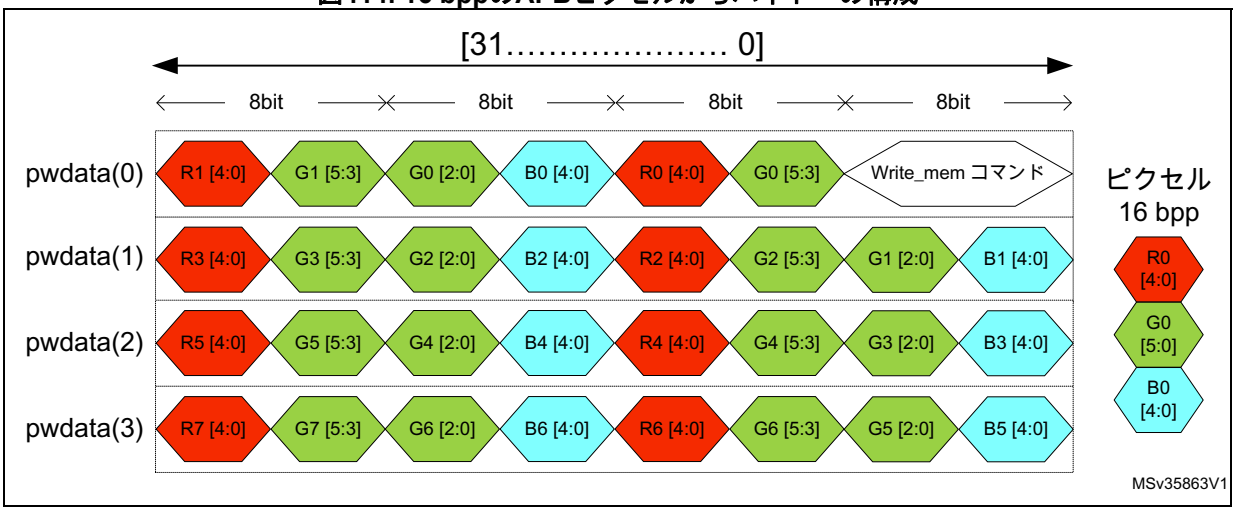


図415. 12 bppのAPBピクセルからバイトへの構成

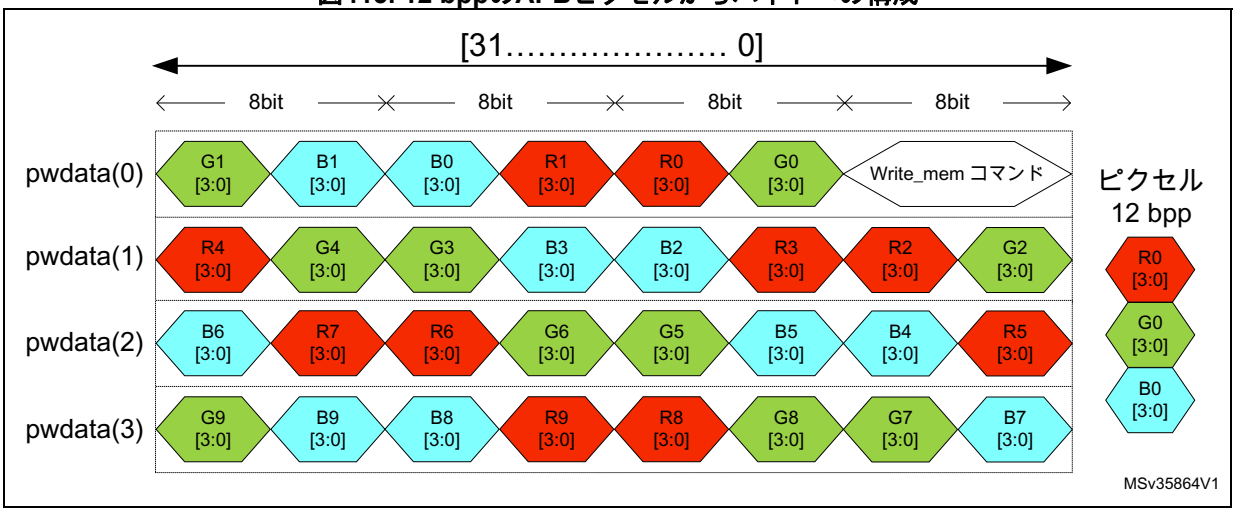
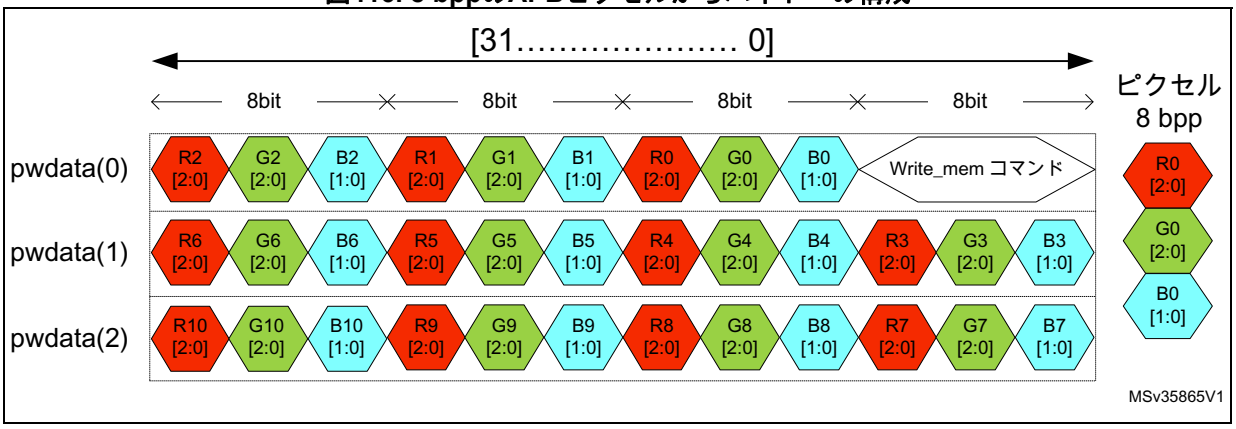


図416. 8 bppのAPBピクセルからバイトへの構成



44.8 機能説明：タイムアウトカウンタ

DSIホストには、さまざまな通信フェーズ中のタイムアウトを管理するためのカウンタが備わっています。各タイムアウト時間は、6つのタイムアウトカウンタ設定レジスタ（DSI_TCCR0～5）で設定できます。

カウンタには次の2種類があります。

- 競合エラー検出タイムアウトカウンタ（セクション44.8.1）
- ペリフェラルレスポンスタイムアウトカウンタ（セクション44.8.2）

44.8.1 競合エラー検出タイムアウトカウンタ

DSIホストには、エラーを通知するための一連のカウンタと条件が実装されています。これには、タイムアウトが発生したかどうかを判断するために使用されるタイマを制御するための一連のレジスタが備えられており、また、読出し操作でクリアされる割込みステータスレジスタのセットも含まれています（詳細は表434参照）。オプションとして、これらのレジスタは、DSI接続の中でエラーが発生したときに、システムをアクティブにするために使用できる割込み信号もトリガします。

表434.競合検出タイムアウトカウンタの設定

タイムアウトカウンタ	値レジスタ	値フィールド	フラグレジスタ	フラグフィールド
ハイスピード送信	DSI_TCCR0	TOHSTX	DSI_ISR1	TOHSTX
低電力受信	DSI_TCCR0	TOLPRX	DSI_ISR1	TOLPRX

これらの16bitカウンタの時間単位は、DSIホスト·クロック制御レジスタ（DSI_CCR）のタイムアウト·クロック分周（TOCKDIV）フィールドで定義されたサイクルで設定されます。

DSIホスト·クロック制御レジスタ（DSI_CCR）のタイムアウト·クロック分周（TOCKDIV）フィールドに書き込まれる値には、入力としてレーン·バイト·クロックを使ってタイムアウト限界値に対する時間単位を定義します。

この仕組みにより、これらの限界値を定義する範囲が広がります。

ハイスピード送信の競合検出

タイムアウト時間は、DSIホスト·タイムアウトカウンタ設定レジスタ1（DSI_TCCR0）のハイスピード送信タイムアウトカウンタ（HSTX_TOCNT）フィールドで設定されます。16bitカウンタは、ハイスピードモードがアクティブである時間を測定します。

このカウンタがDSIホスト·タイムアウトカウンタ設定レジスタ1（DSI_TCCR0）のハイスピード送信タイムアウトカウンタ（HSTX_TOCNT）フィールドで定義された値に達すると、DSIホスト割込みおよびステータスレジスタ1（DSI_ISR1）のタイムアウト·ハイスピード送信（TOHSTX）ビットがアサートされ、DSIホストに内部ソフトリセットが生成されます。

DSIホスト割込みイネーブル·レジスタ1（DSI_IER1）のタイムアウトハイスピード送信割込みイネーブル（TOHSTXIE）ビットがセットされていると、割込みが発生します。

低電力受信の競合検出

タイムアウトは、DSIホスト·タイムアウトカウンタ設定レジスタ0（DSI_TCCR0）の低電力受信タイムアウトカウンタ（LPRX_TOCNT）フィールドで設定されます。16bitカウンタは、低電力受信がアクティブである時間を測定します。

このカウンタがDSIホスト·タイムアウトカウンタ設定レジスタ0（DSI_TCCR0）の低電力受信タイムアウトカウンタ（LPRX_TOCNT）フィールドで定義された値に達すると、DSIホスト割込みおよびス

テータスレジスタ1（DSI_ISR1）のタイムアウト低電力受信（TOLPRX）ビットがアサートされ、DSIホストに内部ソフトリセットが生成されます。

DSIホスト割込みイネーブル·レジスタ1（DSI_IER1）のタイムアウト低電力受信割込みイネーブル（TOLPRXIE）ビットがセットされていると、割込みが発生します。ソフトウェアは、割込み通知を受けると、DSIホストPHY制御レジスタ（DSI_PCTLR）のデジタルイネーブル（DEN）ビットをデアサートしてからアサートすることにより、D-PHYをリセットする必要があります。

44.8.2 ペリフェラルレスポンスタイムアウトカウンタ

ペリフェラルは受信したパケットによっては即座に正しく応答できないこともあります。たとえば、あるペリフェラルが読出しリクエストを受信したが、そのアーキテクチャのためにしばらくの間RAMにアクセスできないことがあります（たとえば、パネルがリフレッシュ中で、応答するのに時間がかかるなど）。この場合、新しいデータを受信したり、新しいリクエストに正しく応答したりする前に、デバイスが前のデータを処理できるように、ホストが十分な時間待つようにタイムアウトをセットします。

表435 には、ペリフェラル応答のタイムアウトが関連付けられているさまざまなカテゴリに属するイベントが記載されています。

表435.PRESP_TOカウンタのさまざまなカテゴリのイベントの一覧

カテゴリ	イベント
BTA PRESP_TOを示唆する項目	バスターンアラウンド
PRESP_TOを示す読出しリクエスト（HSとLPの両方で発生）	（0x04）汎用読出し、パラメータなしのショート （0x14）汎用読出し、1パラメータのショート （0x24）汎用読出し、2パラメータのショート （0x06）DCS読出し、パラメータなしのショート
PRESP_TOを示す 書込みリクエスト（HSとLPの両方で発生）	（0x03）汎用ショート書込み、パラメータなしのショート （0x13）汎用ショート書込み、1パラメータのショート （0x23）汎用ショート書込み、2パラメータのショート （0x29）汎用ロング書込み ロング （0x05）DCSショート書込み、パラメータなしのショート （0x15）DCSショート書込み、1パラメータのショート （0x39）DCSロング書込み/write_LUT、コマンド·パケットのロング （0x37）最大戻りパケット·サイズのセット

DSIホストは、タイムアウトをトリガするイベントを送信すると、D-PHYが停止状態に切り替わり、そのタイムアウト値に達するまでカウンターが実行を開始するようにします。他のイベントを送信する準備ができている場合でも、リンクはLP-11状態のままで、タイムアウトが終了するまで使用されません。

図 417～419は、表435に記載された3つのカテゴリにおけるPRESP_TOカウンタのカウントのフローを示しています。

図417. バスターンアラウンド後のPRESP_TOのタイミング

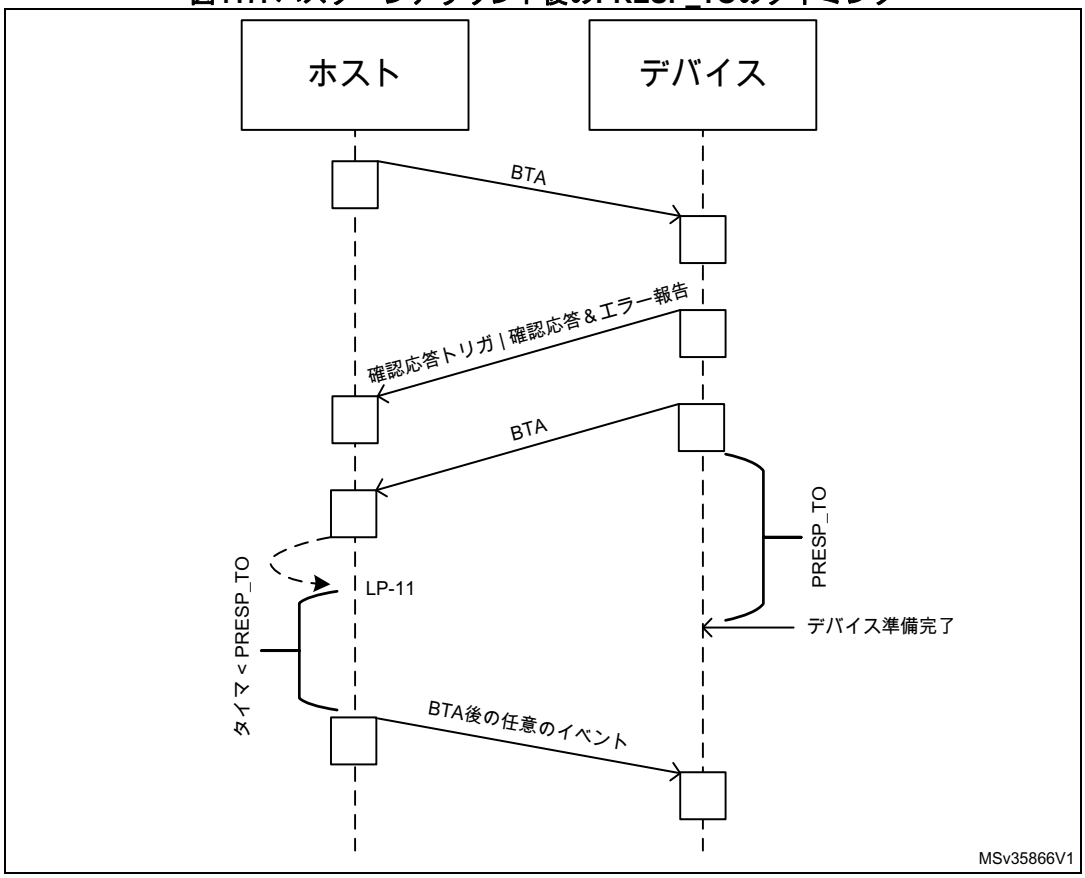


図418. 読出しリクエスト後のPRESP_TOのタイミング（HSまたはLP）

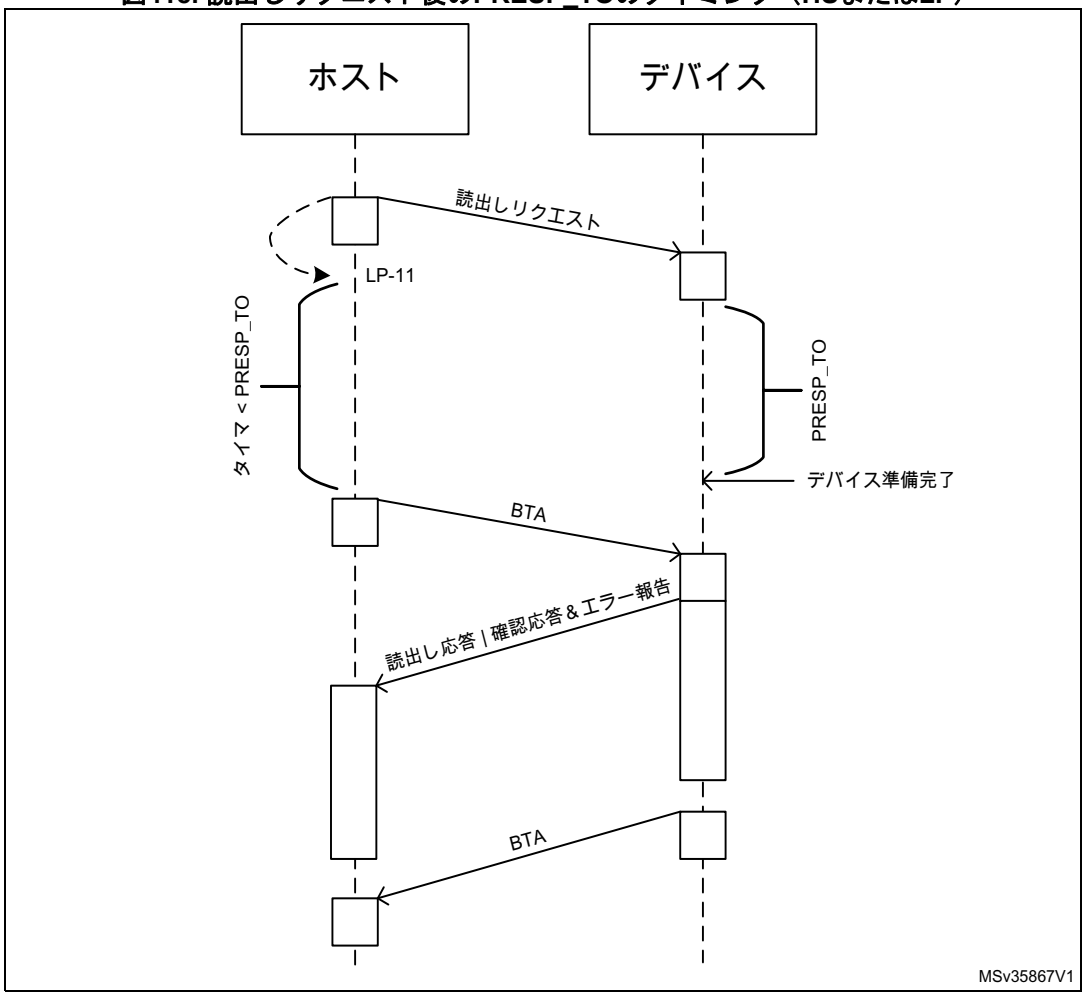


図419. 書込みリクエスト後のPRESP_TOのタイミング（HSまたはLP）

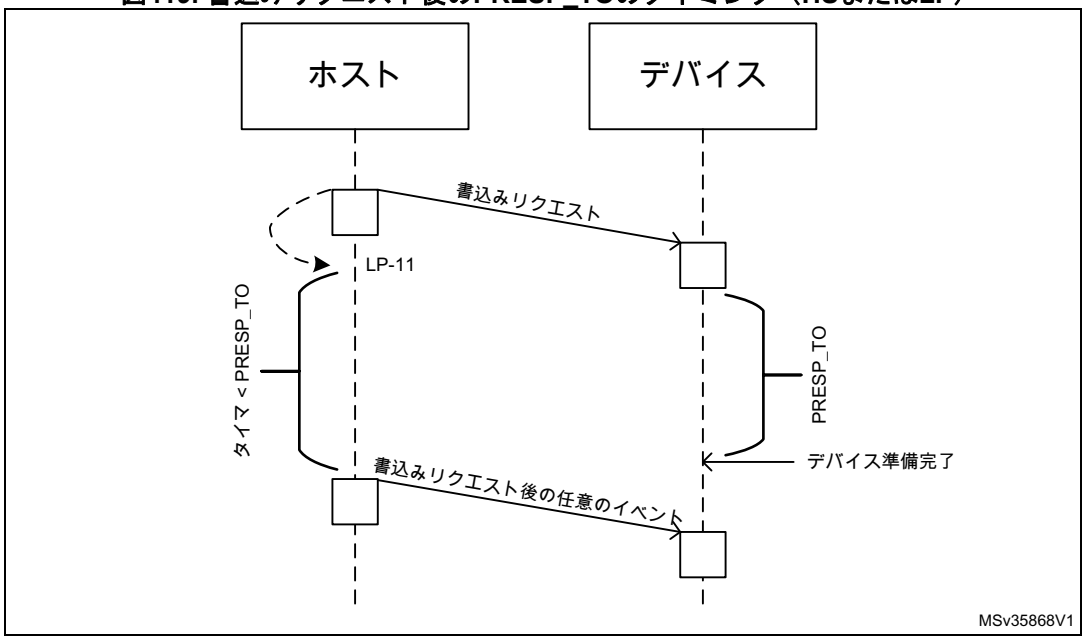


表436 に、PRESP_TOカウンタの設定に使用されるフィールドを説明します。

表436.PRESP_TOカウンタの設定

説明		レジスタ	フィールド
DSIホストがリンクをそのまま維持する期間	ハイスピード読出し操作の送信後	DSI_TCCR1	HSRD_TOCNT
	低電力読出し操作の送信後	DSI_TCCR2	LPRD_TOCNT
	バスターンアラウンド（BTA）完了後	DSI_TCCR5	BTA_TOCNT
DSIホストがリンクを非アクティブに保持する期間	ハイスピード書込み操作の送信後	DSI_TCCR3	HSWR_TOCNT
	低電力書込み操作の送信後	DSI_TCCR4	LPWR_TOCNT

これらのレジスタの値は、レーン·バイト·クロックのサイクル数で計測されます。これらのレジスタはコマンド·モードでのみ使用されます。ビデオ·モードでは、ディスプレイを適切にリフレッシュし続けるために満たすべき厳格なタイミングスケジュールがあり、これらのタイムアウトやその他のタイムアウトによって破られることがあってはならないからです。特定のタイムアウトを0に設定すると、そのカテゴリのイベントに対してLP-11状態への移行とタイムアウトが無効になります。

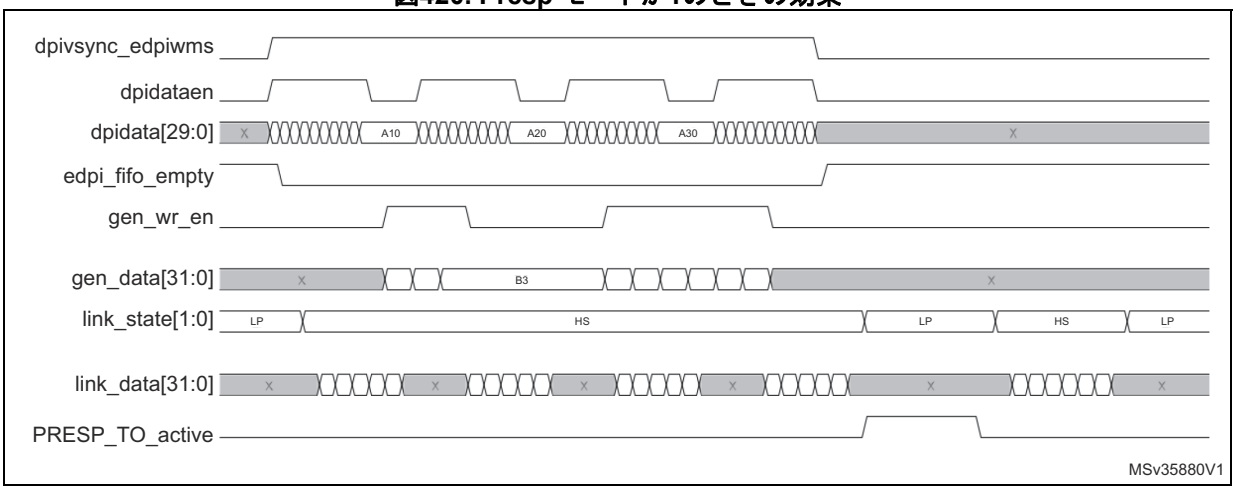
ハイスピードモードでの読取りおよび書込みリクエストは、低電力モードでの読取りおよび書込みリクエストとは別物です。たとえば、HSRD_TOCNTがゼロに設定され、LPRD_TOCNTがゼロ以外の値に設定されている場合、パラメータなしの汎用読出しではハイスピード時のPRESP_TOカウンタはアクティブにならないが、低電力時のPRESP_TOはアクティブになります。

DSIホスト·タイムアウトカウンタ設定レジスタ3（DSI_TCCR3）には、ハイスピード書込み動作タイムアウトのアダプティブ·コマンド·モードにおけるPRESP_TOの通常の動作を変更するための特別なPrespモード（PM）ビットが含まれています。このビットを1に設定すると、以下の両方の条件が満たされる場合に、HSWR_TOCNTからのPRESP_TOを一度だけ使用することが許されます。

- LTDC VSYNC信号が立ち上がり、そして立ち下がる。
- アダプテッド·コマンド·モードでLTDCインタフェースから発出されたパケットが送信され、そのFIFOが再び空になる。

このシナリオでは、送信準備のできた汎用インタフェースからのトラフィックがあっても、アダプテッド･コマンド･モードでないリクエストはD-PHYに送信されず、それらのリクエストは停止状態に戻されます。これが起こると、図420に示されているように、PRESP_TOカウンタが起動され、それが完了したときにのみ、DSIホストは、準備ができた他のトラフィックを送信します。

図420. Presp モードが1のときの効果

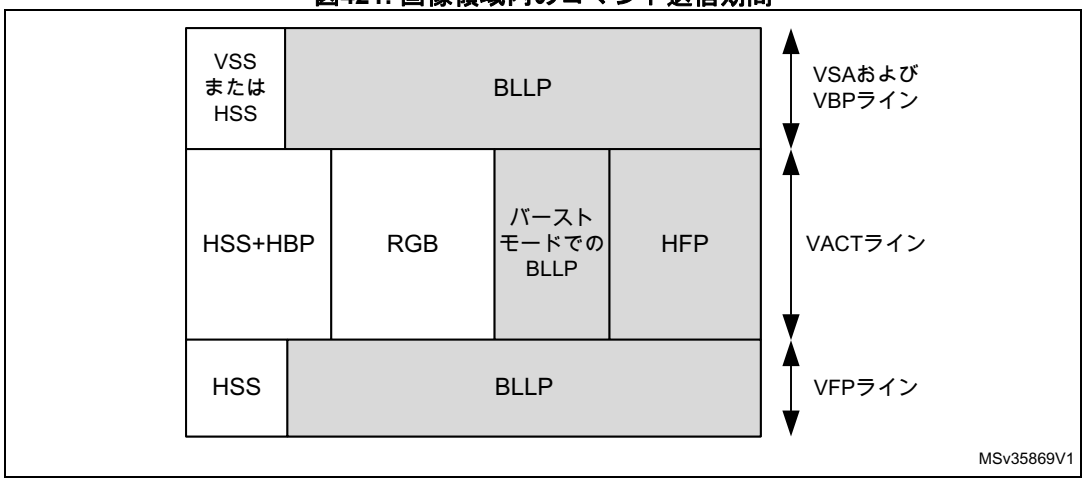


44.9 機能説明：コマンドの送信

44.9.1 ビデオ･モードでのコマンドの送信

DSIホストは、ビデオ･モードの間、ハイスピードおよび低電力でのコマンド送信をサポートしています。DSIホストは、APB汎用インタフェースを通して挿入されたコマンドを送信するために、ブランキングまたは低電力（BLLP）期間を使用します。これらの期間は、図421の灰色の領域に該当します。

図421. 画像領域内のコマンド送信期間

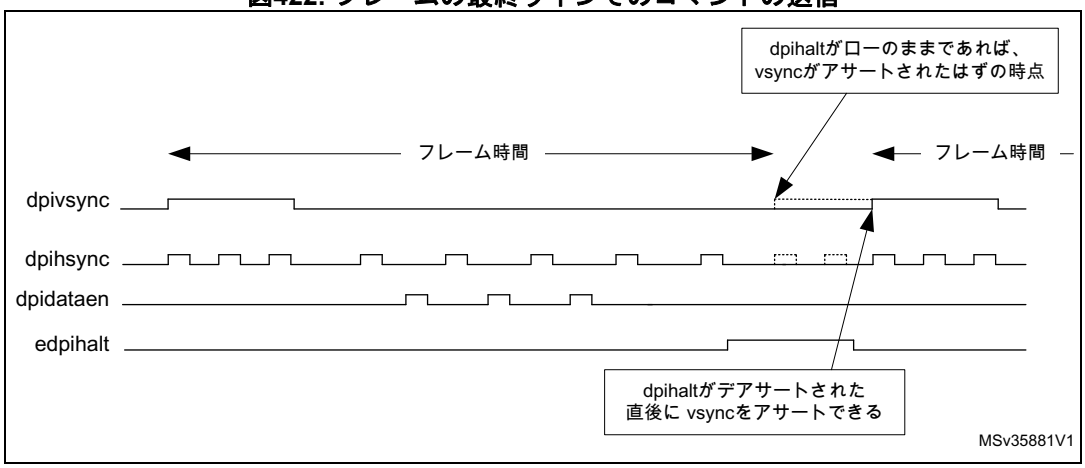


コマンドは、以下のパケット/状態の後のブランキング期間に送信されます。

- 垂直同期スタート（VSS）パケット（ビデオ同期パルスが有効でない場合）
- 水平同期終了（HSE）パケット（VSA、VBP、VFP領域の場合）
- 水平同期開始（HSS）パケット（VSA、VBP、VFP領域でビデオ同期パルスが有効でない場合）
- 水平アクティブ（HACT）状態

BLLPに対応する領域以外に、フレームの最終ラインでも大きなコマンドを送信することができます。この場合、ビデオ･モードのライン時間に違反するので、DPIビデオ･タイミング信号を非アクティブに保つよう要求するためにedpihalt信号をセットします。コマンドがBLLP領域に収まらない場合のみ、そのコマンドは最終ラインに延期され、図422に示すように、ビデオ･モードのライン時間違反の原因となります。

図422. フレームの最終ラインでのコマンドの送信



コマンドは、フレームの最終ラインの場合でも、ラインあたり1つしか送信されませんが、ラインごとに1つのコマンドが送信可能です。

低電力で送信できるコマンドは、ラインあたり1つだけです。しかし、ラインごとに1つの低電力コマンドが送信可能です。ハイスピードでは、DSIホストは、利用可能な時間内に収まると判断した数だけ、複数のコマンドを送信することができます。

DSIホストは、最終ラインが他のラインより短い可能性があるため、最終ラインでのコマンド送信を避けます。たとえば、ライン時間（t_L）は、LTDCインタフェースのt_L よりも半サイクル長くなる可能性があります。つまり、フレーム内の各ラインは、最後のライン時間から半サイクルずつ奪うことになります。この結果、最後のラインは（1/2サイクル）×（ライン数 -1）だけt_L より短くなります。

DSIラッパー制御レジスタ（DSI_WCR）のCOLMビットとSHTDNビットも、コマンド･パケットの送信をトリガできます。そのコマンドは、次のとおりです。

- カラーモード･オン
- カラーモード･オフ
- ペリフェラル･シャットダウン
- ペリフェラル･オン

これらのコマンドは、VACT領域では送信されません。DSIホスト･ビデオ･モード設定レジスタ（DSI_VMCR）の低電力コマンド･イネーブル（LPCE）ビットが設定されている場合、これらのコマンドは低電力モードで送信されます。

低電力モードでは、DSIホスト低電力モード設定レジスタ（DSI_LPMCR）の最大パケットサイズ（LPSIZE）フィールドが、これらのコマンドが送信可能かどうかを判断するために使用されます。DSIホストはこれらのコマンドを最終ラインでは送信しないため、最大パケットサイズ（LPSIZE）は4バイト（ショートパケットのバイト数）以上であると想定されています。

フレーム･バスターンアラウンド確認応答イネーブル（FBTAAE）ビットがDSIホスト低電力モード設定レジスタ（DSI_LPMCR）に設定されている場合、BTAはフレームの最終ラインの後にDSIホストによって生成されます。これは、書込みコマンドまたは読出しコマンドと同時に起きることがあります。いずれの場合も、確認応答が受信されるまでLTDCインタフェースは停止します（DSIバスの制御はホストに戻されます）。

44.9.2 低電力モードでのコマンドの送信

DSIホストは、ハイスピード･ビデオ･モードでの送信中に低電力コマンドを送信するように設定できます。

この機能を有効にするには、DSIホスト･ビデオ･モード設定レジスタ（DSI_VMCR）の低電力コマンド･イネーブル（LPCE）ビットを1にセットします。この場合、水平フロント･ポーチ（HFP）、垂直同期アクティブ（VSA）、垂直バック･ポーチ（VBP）、垂直フロント･ポーチ（VFP）の領域に、低電力モードでコマンドを送信できる時間（バイト単位）を計算する必要があります。

ビデオ･モード設定レジスタ（DSI_VMCR）のビット8から13は、アイドル時にDSIホストがLPに移行できるかどうかを示しています。低電力コマンド･イネーブル（LPCE）ビットが設定され、非ビデオ･パケットがキューにある場合、DSIホストは、特定の領域で低電力モードに入ることが許可されていなくても、低電力の設定を無視し、低電力コマンドを送信します。低電力コマンドを送信後、DSIホストは同期イベントが発生するまで低電力状態を維持します。

たとえば、VFPをハイスピード領域として選択し（LPVFPE = 1'b0）、コマンドをVFP領域で低電力で送信するとしてLPCEを設定しているとします。このコマンドは低電力で送信され、回線は新しいHSSが到着するまで低電力モードのままになります。

VSA、VBP、VFP領域でコマンドを低電力モードで送信するための時間の計算

DSIホスト低電力モード設定レジスタ（DSI_LPMCR）の最大パケット･サイズ（LPSIZE）フィールドは、VSA、VBP、VFP領域の期間中に、（エスケープ･クロックに基づく）低電力モードでコマンドを回線上に送信するのに利用可能な時間（バイト単位）を示します。

最大パケット･サイズ（LPSIZE）の計算は、使用するビデオ･モードに依存します。

図423は、同期パルスでの非バーストのビデオ･モードのタイミング間隔を示し、図424は、バーストおよび同期イベントでの非バーストのビデオ･モードについて示しています。

図423. 同期パルスでの非バーストのLPSIZE

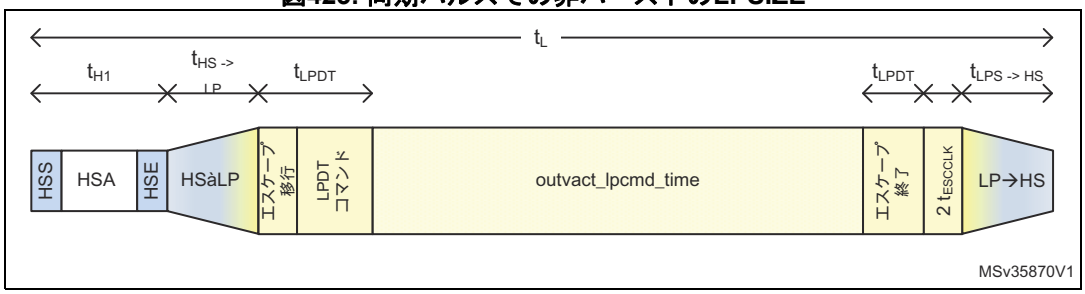
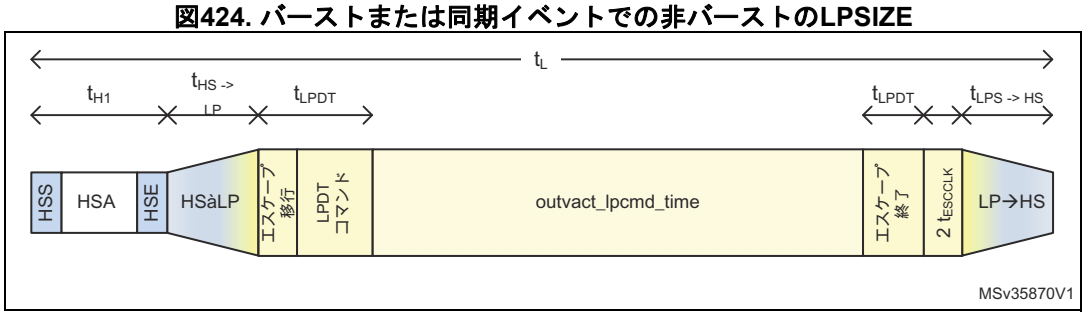


図424. バーストまたは同期イベントでの非バーストのLPSIZE



この時間は、次のように計算されます。

LPSIZE = (t_L - (t_{H1} + $t_{HS->LP}$ + t_{LPHS} + t_{LPDT} + 2 t_{ESCCLK})) / (2 × 8 × t_{ESCCLK})、ここで

- t_L = ライン時間
- t_{H1} = 同期パルスモードのHSAパルスの時間（図423）またはEoTpを含むHSSパケットの送信時間（図424）。
- $t_{HS->LP}$ = 低電力モードに移行する時間
- $t_{LP->HS}$ = 低電力モードを終了する時間
- t_{LPDT} = エスケープ･モード移行、LPDTコマンド、エスケープ終了に関連するD-PHYタイミング。D-PHY仕様によれば、この値はLPで常に11bit（または22 TXエスケープ･クロック･サイクル）です。
- t_{ESCCLK} = DSI_CCRレジスタのTXECKDIVフィールドでプログラムされたエスケープ･クロック周期
- t_{ESCCLK} = DSIホスト実装による遅延

上の式では、使用可能な時間をバイトに変換するために8で除算しています。2で除算しているのは、エスケープ･クロック2サイクルごとに1bitが送信されるためです。最大パケットサイズ（LPSIZE）フィールドは、コマンドを送信するのに十分な時間があるかどうかを判断するために、送信されるコマンドのサイズと直接比較することができます。低電力モードで送信できるコマンドの最大サイズは、このフィールドによって255バイトに制限されます。低電力モードでシャットダウンやカラーなどのDCTRLコマンドを送信するためには、このレジスタを4バイト以上の値にプログラムする必要があります。

ラインあたり12.4µsのフレームを例にとり、エスケープ･クロック周波数20MHz、レーンビットレート800Mbitsと仮定します。この場合、エスケープ･モードで124bitを送信することができます（つまり、124bit = 12.4µs × 20MHz / 2）。それでも、D-PHYプロトコルとPHYタイミングを考慮する必要があります。

以下の仮定を設けます。

- レーン･バイト･クロック周期は10ns（レーンあたり800Mbit）
- エスケープ･クロック周期は50ns（DSI_CCR.TXECKDIV = 5）
- HSSパケットとHSEパケットで区切られた同期パルスで、ビデオを非バースト･モードで送信します。
- DSIは2つのレーン用に設定します。
- D-PHYは低電力からハイスピードモードへの移行に180nsかかります（DSI_DLTCR.LS2HS_TIME = 18）
- D-PHYはハイスピードから低電力モードへの移行に200nsかかります（DSI_DLTCR.HS2LP_TIME = 20）
- t_{HSA} = 420ns.

この例では、次のように13バイトのコマンドが送信できます。

LPSIZE = (12.4 µs - (420ns + 180ns +200ns + (22 × 50ns + 2 × 50ns))) / (2 × 8 × 50ns) = 13バイト。

HFP領域における低電力モードでのコマンド送信時間の計算

DSIホスト低電力モード設定レジスタ（DSI_LPMCR）のVACT最大パケット･サイズ（VLPSIZE）フィールドは、垂直アクティブ（VACT）領域で、（エスケープ･クロックに基づく）低電力モードでコマンドを送信するのに利用可能な時間（バイト単位）を示します。

VACT最大パケット･サイズ（VLPSIZE）の値を計算するには、使用するビデオ･モードを考慮します。図425は同期パルスでの非バーストのビデオ･モードのタイミング間隔、図426は同期イベントでの非バーストのビデオ･モードのタイミング間隔、図427はバースト･ビデオ･モードのタイミング間隔を示します。

図425. 同期パルスでの非バーストのVLPSIZE

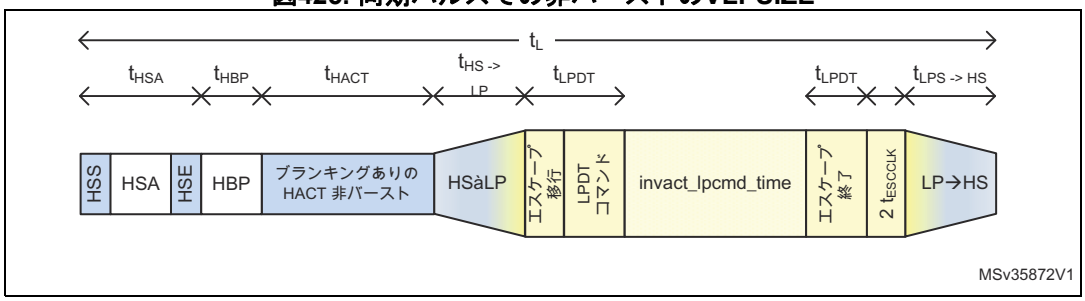


図426. 同期イベントでの非バーストのVLPSIZE

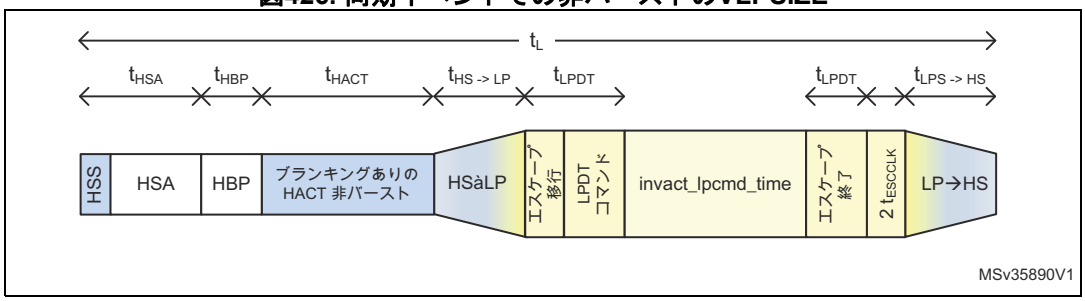
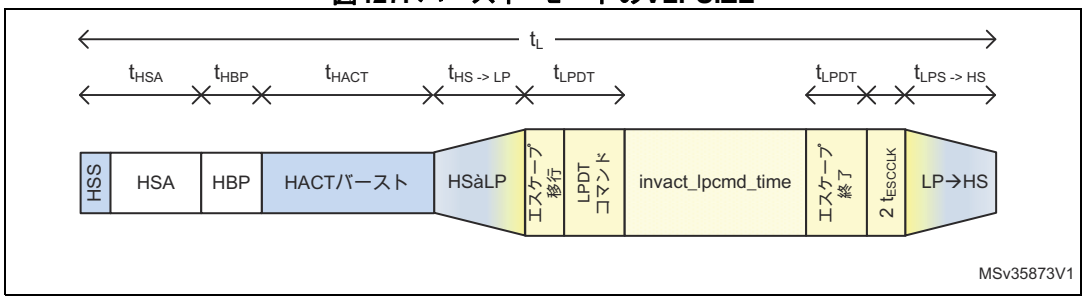


図427. バースト·モードのVLPSIZE



この時間は、次のように計算されます。

$VLPSIZE = (t_L - (t_{HSA} + t_{HBP} + t_{HACT} + t_{HS->LP} + t_{LP->HS} + t_{LPDT} + 2\ t_{ESCCLK})) / (2 \times 8 \times t_{ESCCLK})$

ここで、

- t_L = ライン時間
- t_{HSA} = HSAパルスの時間（DSI_VHSACR.HSA）
- t_{HBP} = 水平バック·ポーチの時間（DSI_VHBPCR.HBP）
- t_{HACT} = ビデオ·アクティブの時間。バースト·モードでは、ビデオ·アクティブは時間圧縮され、t_{HACT} = VPSIZE × Bytes_per_Pixel / Number_Lanes × $t_{Lane_byte_clk}$ として計算されます。
- t_{ESCCLK} = DSI_CCRレジスタのTXECKDIVフィールドでプログラムされたエスケープ·クロック周期

VLPSIZEフィールドは、コマンドを送信するための時間があるかどうかを判断するために、送信されるコマンドのサイズと直接比較することができます。

ラインあたり16.4µsのフレームを例にとり、エスケープ·クロック周波数20MHz、レーンビットレート800Mbit/sと仮定します。この場合、エスケープ·モードで420bitを送信することができます（つまり、164bit = 16.4µs × 20MHz / 2）。それでも、フレームの垂直アクティブ領域であるため、D-PHYプロトコルとPHYのタイミングとは別に、HSA、HBP、HACTのタイミングを考慮してください。以下の仮定を設けます。

- アクティブなレーン数は、4
- レーン·バイト·クロック周期（lanebyteclkperiod）は10ns（レーンあたり800Mbit）
- エスケープ·クロック周期は50ns（DSI_CCR.TXECKDIV = 5）
- D-PHYは低電力からハイスピードモードへの通過に180nsかかります（DSI_DLTCR.LP2HS_TIME = 18）
- D-PHYはハイスピードから低電力モードへの通過に200nsかかります（DSI_DLTCR.HS2LP_TIME = 20）
- t_{HSA} = 420ns
- t_{HBP} = 800ns
- t_{HACT} = 12800ns（24 bppで1280ピクセルを送信する場合）
- ビデオは非バースト·モードで送信されます
- DSIは4つのレーン用に設定します。

この例では、ビデオは 非バースト·モードで送信されると見なします。VLPSIZEは、次のように計算されます。

VLPSIZE = (16.4µs -(420ns + 800ns + 12.8µs + 180ns +200ns + (22 × 50ns + 2 × 50ns)) / (2 × 8 × 50ns) = 1バイト

この期間には、1バイトのみを送信できます。ショート·パケット（たとえば、汎用ショート書込み）には最低4バイトが必要です。したがって、この例では、コマンドはVACT領域では送信されません。

バースト･モードが有効な場合、HACTは時間圧縮されるため、VACT領域でコマンドを送信するのにより多くの時間が利用できます。

VLPSIZE = (16.4µs - (420ns + 800ns + (1280 × 3 / 4 × 10ns) + 180ns + 200ns + (22 × 50ns + 2 × 50ns) / (2 × 8 × 50ns) = 5バイト

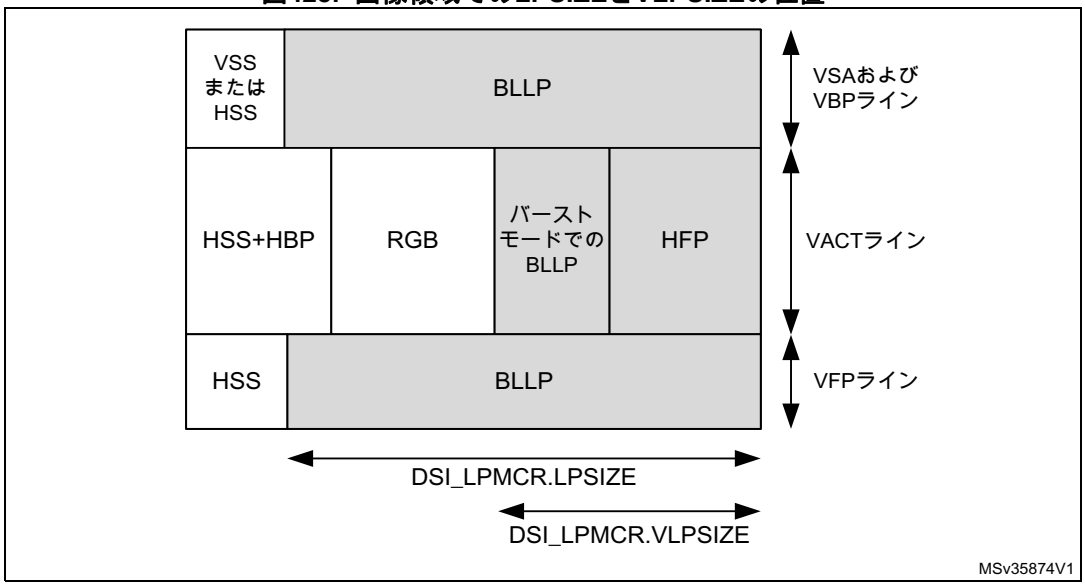
バースト･モードでは、VLPSIZEは5バイトであり、よって4バイトのショートパケットは送信することができます。

さまざまな期間でのコマンドの送信

LPSIZEフィールドとVLPSIZEフィールドによって、コマンドがBLLP期間のいずれかで送信可能かどうかを判断するための簡単な比較が可能になります。

図428には、VLPSIZEとLPSIZEの意味を、影付きの領域およびVACT領域と照らし合わせて示しています。

図428. 画像領域でのLPSIZEとVLPSIZEの位置



44.9.3 ハイスピードでのコマンドの送信

DSI_VMCRレジスタのLPCEビットが0の場合、コマンドはビデオ･モードでハイスピードで送信されます。この場合、DSIホストが各コマンドを送信できる領域を自動的に判定するため、プログラミングや計算は不要です。

44.9.4 読出しコマンドの送信

レジスタのMRD_TIMEフィールドには、読出しコマンドの実行に必要な最大時間をレーン･バイト･クロック･サイクルで設定します。これは次のように計算されます。

MRD_TIME = 読出しコマンドを低電力モードで送信する時間 + 低電力モードに移行および終了する時間 + ペリフェラル･デバイスから読出しデータ･パケットを返す時間

ペリフェラルから読出しデータ･パケットを返す時間は、読み出されたバイト数と、ホストのエスケープ･クロックではなくペリフェラルのエスケープ･クロック周波数に依存します。MRD_TIMEフィールドは、BLLP期間内に読出しコマンドを完了する時間があるかどうかを判断するために、ハイスピード･モードと低電力モードの両方で使用されます。

ハイスピード･モード（LPCE = 0）では、MRD_TIMEは次のように計算されます。

MRD_TIME = ($t_{HS->LP}$ + $t_{LP->HS}$ + t_{read} + 2 × t_{BTA}) / レーン･バイト･クロック周期

低電力モード（LPCE = 1）では、MRD_TIMEは次のように計算されます。

MRD_TIME = ($t_{HS->LP}$ + $t_{LP->HS}$ + t_{LPDT} + t_{lprd} + t_{read} + 2 × t_{BTA}) / レーン･バイト･クロック周期、ここで

- $t_{HS->LP}$ = 低電力モードに移行する時間
- $t_{LP->HS}$ = 低電力モードを終了する時間
- t_{LPDT} = エスケープ･モードへの移行、LPDTコマンド、エスケープ･モード終了に関連するD-PHYタイミング（D-PHY仕様によると、この値は常にLPで11bit、または22 TXエスケープ･クロック･サイクルです）
- t_{lprd} = 低電力モードでの読出しコマンドの時間（64 × TXエスケープ･クロック）
- t_{read} = ペリフェラルから読出しデータ･パケットを返す時間
- t_{BTA} = バスターンアラウンドを実行する時間（D-PHYに依存）

ライン時間内で読出しコマンドを発行するのに十分な時間を確保するために、ペリフェラルから読み出す最大バイト数を最小に保つことが推奨されます。MRD_TIME × レーン･バイト･クロック周期がLPSIZE × 16 × ホストのエスケープ･クロック周期より小さくなるようにしてください。そうでない場合、読出しコマンドはフレームの最終ラインでディスパッチされます。多数のパラメータ（＞16）を読み出す必要がある場合は、読出しコマンドの実行時にMRD_TIMEを増やしてください。読出しが完了したら、MRD_TIMEをより小さい値に減少させてください。

フレームの最終ラインで読出しコマンドが発行された場合、LTDCインタフェースは停止し、読出しコマンドが進行するまで停止したままになります。この期間中は、ビデオの送信は停止する必要があります。

44.9.5 低電力モードでのクロック･レーン

D-PHYの消費電力を削減するため、DSIホストはハイスピードモードで送信していないとき、クロック･レーンを低電力モードにします。コントローラは、ソフトウェアの直接介入なしに、クロック･レーンをHS（クロック･レーンがアクティブにクロックを送信）からLP状態へ自動的に遷移させます。この機能は、DSI_CLCRレジスタのDPCCおよびACRビットを設定することによって有効にできます。

コマンド･モードでは、DSIホストは、送信するHSパケットがない場合、クロック･レーンを低電力モードにすることができます。

ビデオ･モード（LTDCインタフェース）では、DSIホストコントローラは、内部のビデオとPHYのタイミング設定を使用して、クロックラインが低電力モードに入る時間があるかどうかを判断し、ピクセルデータと同期イベントのビデオデータ送信を損なわないようにします。

ビデオ･モードの正しい設定（セクション 44.5：機能説明：LTDCインタフェースのビデオ･モードを参照）と共に、DSIホストは、クロック･レーンがハイスピードモードから低電力モードに移行するのに必要な時間、およびその逆を行うのに必要な時間を知る必要があります。必要な値はD-PHY仕様から得ることができます。DSI_CLTCRレジスタを以下の値でプログラムしてください。

- HS2LP_TIME = クロック･レーンのHSからLPまでの時間／HSのバイトクロック周期（lanebyteclk）
- LP2HS_TIME = クロック･レーンのLPからHSまでの時間／HSのバイトクロック周期（lanebyteclk）

プログラムされた値に基づいて、DSIホストはビデオフレームの非アクティブ領域中にクロック･レーンが低電力モードに入るのに十分な時間があるかどうかを計算します。

DSIホストは、以下の3つの可能なシナリオのうち、省電力に関して従うべき最良の方法を決定します。

- 低電力モードに移行するのに十分な時間がない。したがって、図429に示すようにブランキング期間を追加する。
- データ・レーンが低電力モードに移行するのに十分な時間があるが、クロック・レーンが低電力モードに移行するのに十分な時間がない（図430参照）。
- 図431のように、データ・レーンおよびクロック・レーンの両方が低電力モードに移行するのに十分な時間がある。

図429. HSでのクロック・レーンとデータ・レーン

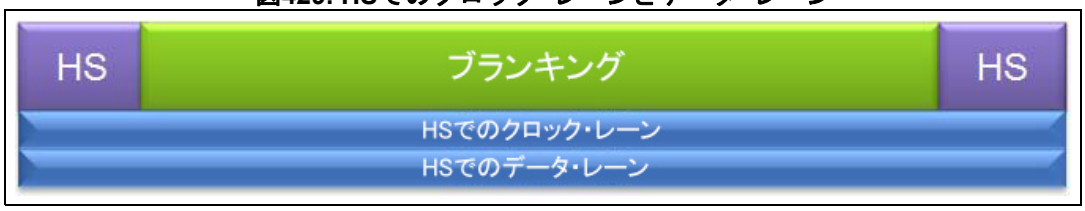


図430. HSでのクロック・レーンとLPでのデータ・レーン

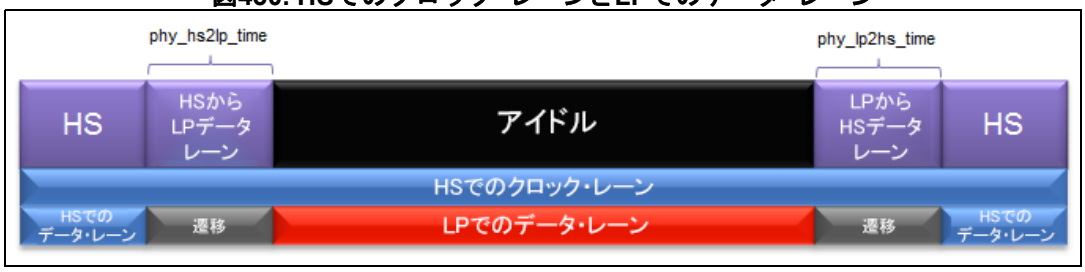
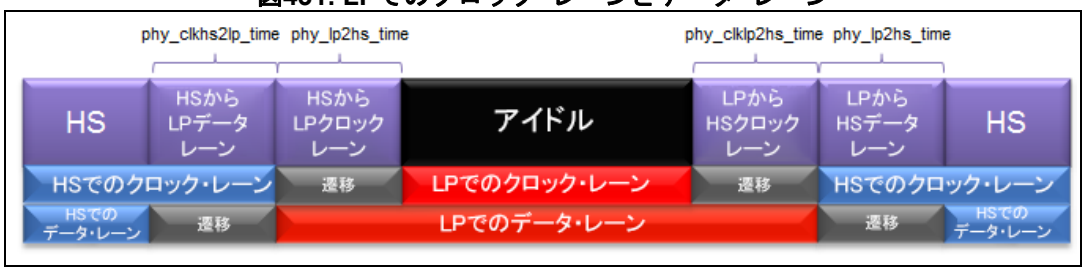


図431. LPでのクロック・レーンとデータ・レーン

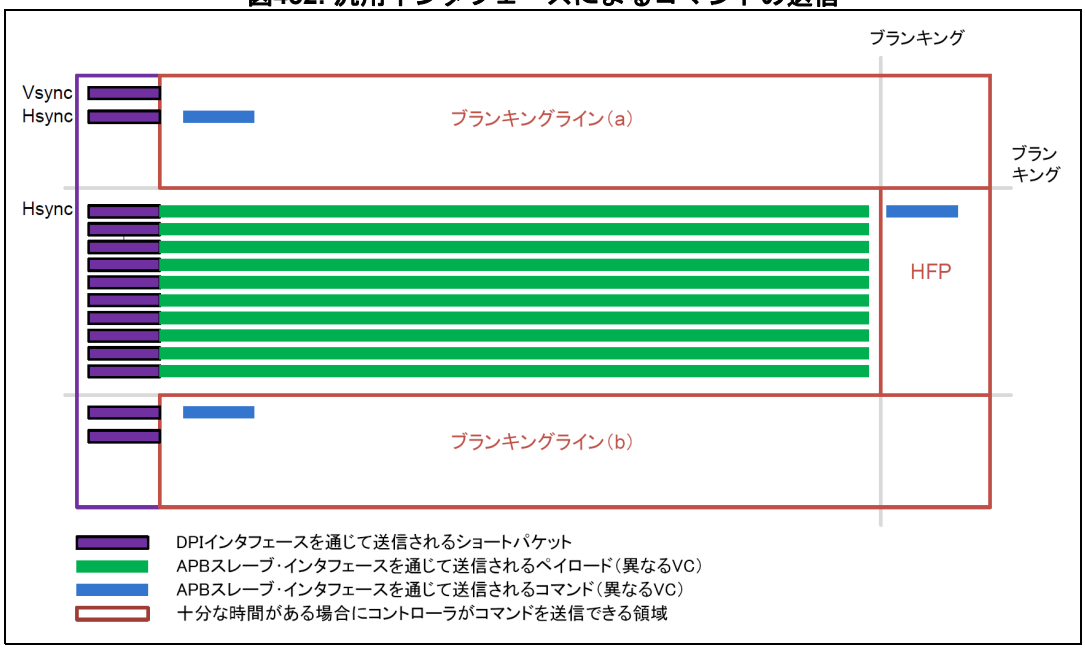


44.10 機能説明：仮想チャネル

DSIホストは、各インタフェースに使用する仮想チャネル（VC）の選択をサポートしています。複数のディスプレイがそれぞれ異なる仮想チャネル識別子を持つ場合、複数の仮想チャネルを使用することで、システムは複数のディスプレイを同時にアドレス指定することができます。

LTDCインタフェースが特定の仮想チャネル用に設定されている場合、APBスレーブ汎用インタフェースを使用して、ビデオストリームの送信中にコマンドを発行することができます。これにより、進行中のビデオストリームを通じてコマンドを送信し、異なる仮想チャネルに対応させることが可能になり、複数ディスプレイとのインタフェースが実現します。ビデオ·モード中は、ビデオストリーム送信が最優先されます。したがって、汎用インタフェースからのパケットのようなサイドバンド·パケットの送信は、ビデオストリームの送信中に時間が空いたときにのみ転送されます。DSIホストは利用可能な時間帯を識別し、それらを使用して汎用インタフェース·パケットを転送します。図432は、DSIホストがAPB汎用インタフェースからのパケットを、LTDCインタフェースから送信されるビデオストリーム内に挿入する場所を示しています。

図432. 汎用インタフェースによるコマンドの送信



また、異なる仮想チャネルを使用して、汎用インタフェースだけで複数のディスプレイに対応することも可能です。汎用インタフェースは設定によって特定の仮想チャネルに制限されないため、異なる仮想チャネルを使ってパケットを発行することができます。これによりインタフェースは、異なる仮想チャネルによってディスプレイに提供するパケットを時間多重化することができます。

以下の設定レジスタを使用して、LTDCおよびAPBスレーブ汎用インタフェースを介した送信に関連付けられた仮想チャネルIDを選択できます。

- DSI_LVCIDR.VCIDフィールドでは、LTDCインタフェースを使用するビデオ·モードのパケットに関連付けられる仮想チャネルIDを設定します。
- DSI_GHCR レジスタでは、APB スレーブ汎用インタフェースを使用する送信のためのパケットヘッダ（使用する仮想チャネルIDが格納されている）を設定します。
- DSI_GVCIDR レジスタでは、保存して汎用インタフェースに返す読出し応答の仮想チャネル ID を設定します。

44.11 機能説明：ビデオ･モード･パターン･ジェネレータ

ビデオ･モード･パターン･ジェネレータによって、水平/垂直カラーバーやD-PHYのBERテスト･パターンを刺激なしで送信できます。

フレーム要件を 表437に記載されているビデオレジスタに定義する必要があります。

表437.フレーム要件設定レジスタ

レジスタ名	説明
DSIホスト･ビデオ･モード設定	ビデオ･モードの設定
DSIホスト･ビデオ･パケット設定	ビデオ･パケットのサイズ
DSIホスト･ビデオ･チャンク設定	チャンク数
DSIホスト･ビデオnullパケット設定	nullパケットのサイズ
DSIホスト･ビデオHSA設定	水平同期アクティブ時間
DSIホスト･ビデオHBP設定	水平バック･ポーチ時間
DSIホスト･ビデオ･ライン設定	ライン時間
DSIホスト･ビデオVSA設定	垂直同期アクティブ期間
DSIホスト･ビデオVBP設定	垂直バック･ポーチ期間
DSIホスト･ビデオVFP設定	垂直フロント･ポーチ期間
DSIホスト･ビデオVA設定	垂直解像度

44.11.1 カラーバー･パターン

カラーバー･パターンは、白、黄、シアン、緑、マゼンタ、赤、青、黒の8色のバーで構成されます。

各色の幅は、ライン･ピクセル･サイズ（垂直パターン）またはライン数（水平パターン）を8で割って計算されます。垂直カラーバー･モード（図433）では、各単色バーの幅は、ラインのピクセル数を8で割った値です。ラインのピクセル数が8で割り切れない場合、最後の色（黒）に残りが含まれます。

水平カラーバー･モード（図434）では、各色ラインの色幅は、フレームのライン数を8で割った値です。フレームのライン数が8で割り切れない場合、最後の色（黒）に残りが含まれます。

図433. 垂直カラーバー・モード

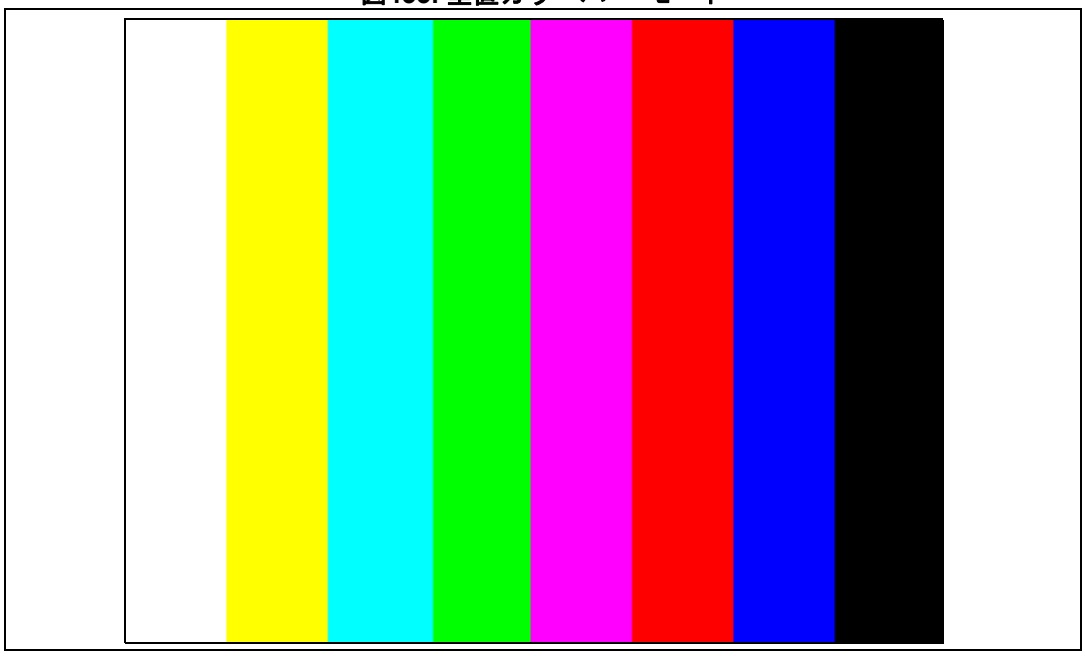


図434. 水平カラーバー・モード

44.11.2 カラー･コーディング

表438に、使用されているRGBの成分を示します。

表438.RGBの成分

	白	黄	シアン	緑	マゼンダ	赤	青	黒
R	高	高	低	低	高	高	低	低
G	高	高	高	高	低	低	低	低
B	高	低	高	低	高	低	高	低

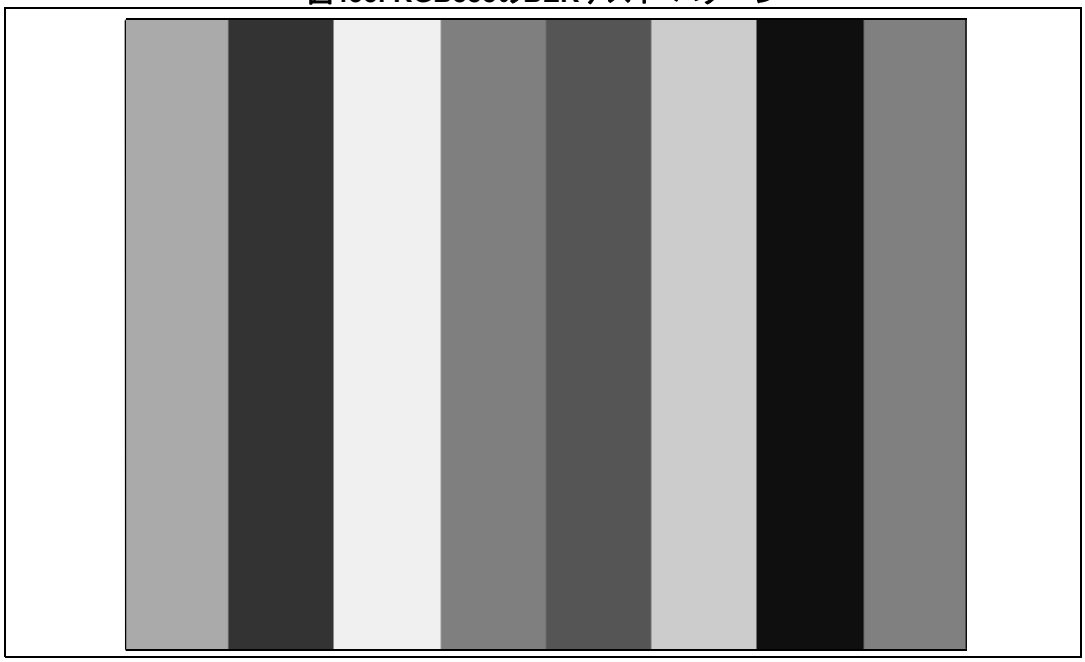
44.11.3 BERテスト･パターン

BERテスト･パターンによって、適合性テストが簡素化されます。このパターンでRX D-PHYがデータを正しく受信できる能力をテストできます。次のデータ･パターンが必要です。

- Xバイトの0xAA（高周波パターン、反転）
- Xバイトの0x33（中周波パターン）
- Xバイトの0xF0（低周波パターン、反転）
- Xバイトの0x7F（単独の0パターン）
- Xバイトの0x55（高周波パターン）
- Xバイトの0xCC（中周波パターン、反転）
- Xバイトの0x0F（低周波パターン）
- Yバイトの0x80（単独の1パターン）

ほとんどの場合、YはXと等しいですが、ラインの長さと使用されるカラー･コーディングによっては、YはXと異なる場合があります。RGB888カラー･コーディングと水平解像度が8の倍数の場合、DSIディスプレイには、図435に示すパターンが表示されます。

図435. RGB888のBERテスト･パターン



44.11.4 ビデオ･モード･パターン･ジェネレータの解像度

方向、BERモード、カラー･コーディングによって、ビデオ･モード･パターン･ジェネレータが受け付ける最小解像度は以下のとおりです。

- BERモード：8x8
- 水平カラーバー･モード：8x8
- 垂直カラーバー･モード：8x8

垂直パターン

各カラーバーの幅は、8色のテストパターン用の水平解像度（ピクセル）の分割によって決まります。水平解像度が8で割り切れない場合、最後の色（黒）が拡張されて解像度が満たされます。

図436 の例では、水平分解能は103です。

図436. 垂直パターン（103x15）

水平パターン

各カラーバーの幅は、8色のテストパターン用の垂直解像度（ライン）の数を分割することで決まります。垂直解像度が8で割り切れない場合、図437に示すように最後の色（黒）が拡張されて解像度が満たされます。

図437. 水平パターン（103x15）

44.12 機能説明：D-PHYの管理

埋め込みMIPI® D-PHYは、DSIホストによって直接制御され、DSIラッパーを介して設定されます。

DSIとD-PHYにクロックと電源を供給するために、専用のPLLと専用のバイアスも組み込まれています。

44.12.1 D-PHYの設定

D-PHYの設定は、DSI_WPCRxレジスタによりDSIラッパーを介して行われます。

ピンでのスルーレート調整

DSI通信を微調整するために、以下のスルーレートを調整することができます。

- DSI_DPCSRCRによるHS-TX速度のクロック·スルーレート
- DSI_DPDL0SRCRとDSI_DPDL1SRCRによるHS-TX速度のデータ·スルーレート

帯域の設定

D-PHYの周波数帯域は、クロック·レーンについてはDSI_DPCBCRレジスタで、データ·レーンについてはDSI_DPDL0BCRとDSI_DPDL1BCRで制御されるため、クロック·レーンとデータ·レーンについて調整する必要があります。

レーンのカスタム設定

DSIの統合を容易にするため、表440で説明するように、レーンピンをレーン上で入れ替えることができます。

表440.レーンのカスタム設定

機能	レーン	DSI_WPCR0のイネーブル·ビット
レーンピンのスワップ	クロック·レーン	SWCL
	データ·レーン0	SWDL0
	データ·レーン1	SWDL1

44.12.2 D-PHYのHS2LPおよびLP2HSの時間

DSIシステムは、2つのHS送信の間に十分な時間があれば、ブランキング期間中にLPモードに切り替えることができます。

スケジューリングを行い、切り替えが可能かどうかを推定するために、DSIホストにおいて、HSからLPへの遷移とLPからHSへの遷移の時間を、クロック·レーンについてはDSI_CLTCRレジスタに、データ·レーンについてはDSI_DLTCRレジスタにプログラムする必要があります。

表441に、レーン·バイト·クロックサイクル数で表された、これらのタイミングについてプログラムするべき値の概算を示します。

表441.帯域周波数（MHz）に対するHS2LPおよびLP2HSの値

	80～100	100～120	120～160	160～200	200～240	240～320	320～390	390～450	450～510
クロックHS2LP	4 + LPXO	4 + LPXO	5 + LPXO	6 + LPXO	7 + LPXO	7 + LPXO	9 + LPXO	9 + LPXO	11 + LPXO
クロックLP2HS	8 + 3 × LPXO	8 + 3 × LPXO	14 + 3 × LPXO	16 + 3 × LPXO	21 + 3 × LPXO	25 + 3 × LPXO	32 + 3 × LPXO	33 + 3 × LPXO	40 + 3 × LPXO

表441.帯域周波数（MHz）に対するHS2LPおよびLP2HSの値 （続き）

	80～100	100～120	120～160	160～200	200～240	240～320	320～390	390～450	450～510
データ HS2LP	5	5	6	7	8	8	10	10	12
データ LP2HS	8 + 2 × LPXO	9 + 2 × LPXO	11 + 2 × LPXO	14 + 2 × LPXO	15 + 2 × LPXO	17 + 2 × LPXO	19 + 2 × LPXO	20 + 2 × LPXO	23 + 2 × LPXO

44.12.3 D-PHYの特殊動作

DSIラッパーには、D-PHYを特定の状態や動作に強制するための制御ビットがいくつかあります。

レーン状態の強制

DSI_WPCR0レジスタのFTXSMDLおよびFTXSMCLビットによって、データ・レーンやクロック・レーンを強制的にTX STOPモードにすることができます。

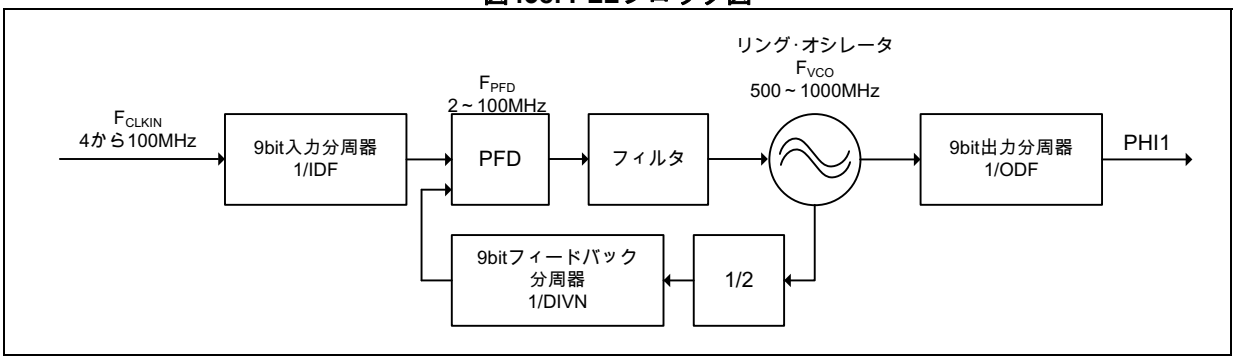
このビットを設定すると、それぞれのレーン・モジュールは直ちに送信制御モードにジャンプし、停止状態（LP-11）の送信を開始します。

この機能は、間違ったBTAシーケンスの後にTXモードに戻るために使用できます。

44.12.4 DSI PLLの制御

図438 に示すように、専用のDSI PLLはDSIラッパーを介して制御されます。

図438. PLLブロック図



PLL出力周波数はDSI_WRPCRレジスタのフィールドで設定されます。VCO周波数とPLL出力周波数は以下のように計算されます。

$$F_{VCO} = (CLK_{IN} / IDF) \times 2 \times NDIV$$

$$PHI = F_{VCO} / ODF$$

ここで、

- CLK_{IN} は 4～100MHzの範囲
- DSI_WRPCR.NDIVは 1～511 の範囲
- DSI_WRPCR.IDFは1～511 の範囲
- F_{PFD} は2～100MHzの範囲
- F_{VCO} は 500MHz～1GHzの範囲
- DSI_WRPCR.ODFは 1～511 となり得ます
- PHIは 31.25～500MHz の範囲

PLLは、DSI_WRPCRレジスタのPLLENビットをセットすることによって有効になります。

PLLがロックされると、DSI_WISRにPLLLIFビットがセットされます。PLLLIEビットがDSI_WIERにセットされていると割込みが発生します。

PLLの状態（ロックまたはアンロック）は、DSI_WISRレジスタのPLLLSフラグによって監視できます。

PLLがロック解除されると、DSI_WISRのPLLUIFビットがセットされます。DSI_WIERレジスタのPLLUIEビットがセットされていると割込みが発生します。

DSIのPLL設定を変更できるのは、PLLが無効なときだけです。

44.12.5 D-PHYのバイアス制御

D-PHYへ基準電圧を提供するバイアスは、DSI_BCFGRレジスタのPWRUPビットをセットすることで有効になります。

44.13 機能説明：割込みとエラー

割込みは、DSIホストまたはDSIラッパーによって生成される可能性があります。

すべての割込みは、割込みコントローラへの1つの割込みレーンに統合されます。

44.13.1 DSIラッパー割込み

割込みは次のイベントによって生成できます。

- ティアリング効果イベント
- リフレッシュ終了
- PLLロック
- PLLアンロック

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表442.DSIラッパー割込みリクエスト

割込みイベント	DSI_WISRのイベント･フラグ	DSI_WIERのイネーブル制御ビット
ティアリング効果	TEIF	TEIE
リフレッシュ終了	ERIF	ERIE
PLLロック	PLLLIF	PLLLIE
PLLアンロック	PLLUIF	PLLUIE

44.13.2 DSIホストの割込みとエラー

DSI_ISR0およびDSI_ISR1レジスタは、エラー状態報告に関連しています。これらのレジスタは、エラーの発生をシステムに知らせるために割込みをトリガすることができます。

DSIホストには、DSI_ISR0またはDSI_ISR1レジスタにエラーが発生したときにハイにセットされる1本の割込みラインがあります。

割込みのトリガは、DSI_IER0およびDSI_IER1マスクレジスタをプログラミングすることでマスクできます。デフォルトでは、すべてのエラーがマスクされています。これらのレジスタのいずれかのビットが1にセットされていると、特定のエラーに対する割込みが有効になります。エラービットは常にそれぞれのDSI_ISRレジスタにセットされます。DSI_ISR0およびDSI_ISR1レジスタは、常に読出し操作の後にクリアされます。割込みの原因となったすべてのレジスタが読み出されると、割込みラインはクリアされます。

割込み強制レジスタ（DSI_FIR0とDSI_FIR1）はテスト目的で使用され、割込みソースをトリガする条件をアクティブにすることなく、個別に割込みイベントをトリガすることができます（その目的のために刺激を作り出すのは非常に複雑です）。この機能により、割込みイベントに関連するソフトウェアの開発とテストも容易になります。これらのレジスタの任意のビットを1にセットすることで、対応する割込みがトリガされます。

図439中の淡黄色のボックスは、いくつかのエラーの場所を示しています。

図439. エラー発生源

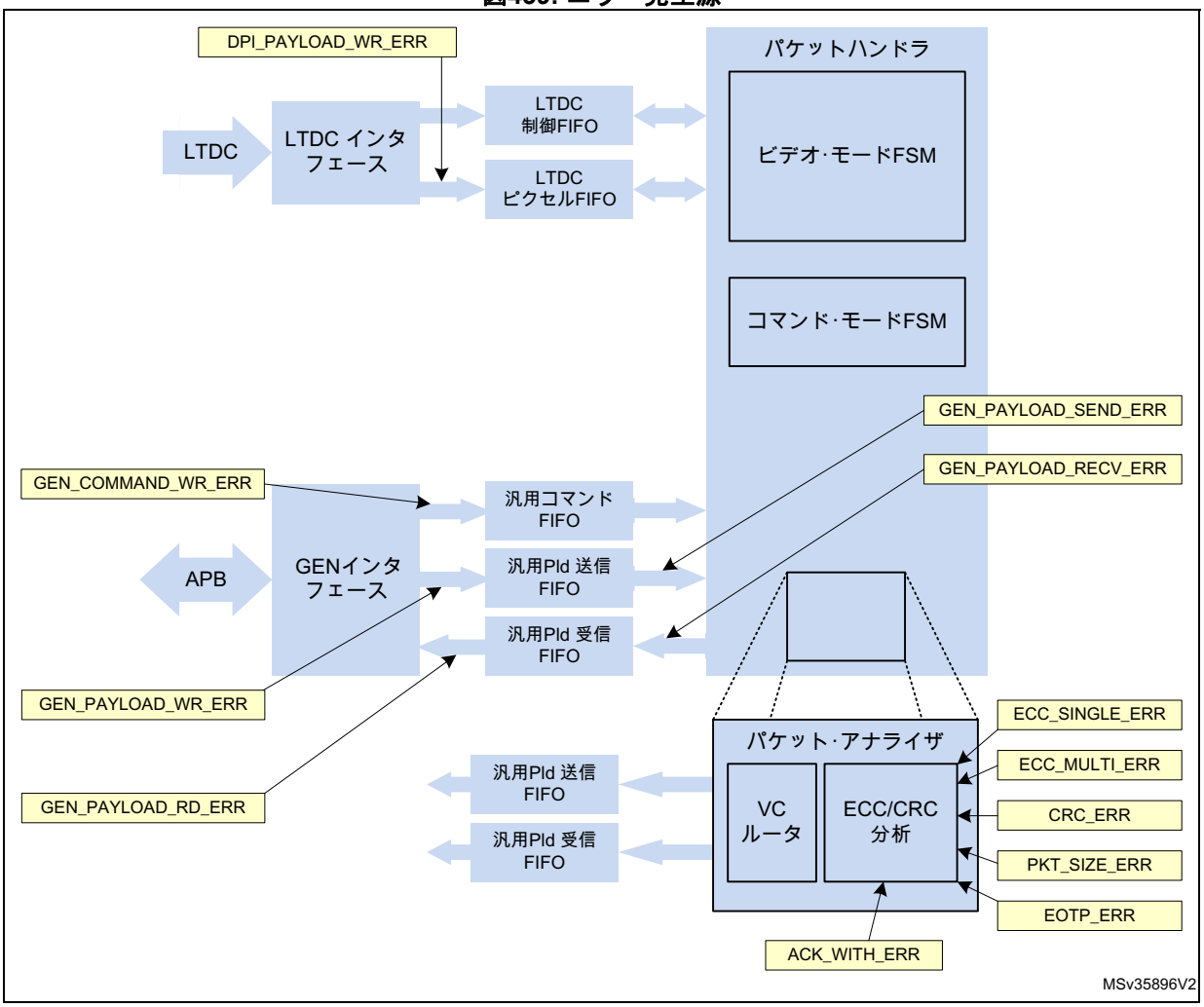


表443 では、これらの割込みが発生する理由と、割込みから回復する方法を説明しています。

表443.エラーの原因と回復方法

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
0	20	PE4	D-PHYがLP1の競合エラーを報告しています。 D-PHYホストは、ラインをハイに駆動しようとしたとき競合を検出しています。	D-PHYを競合から回復してください。 DSIホストをリセットしてパケットを再送信してください。このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。
0	19	PE3	D-PHYがLP0の競合エラーを報告しています。 D-PHYホストは、ラインをローに駆動しようとしたとき競合を検出しています。	D-PHYを競合から回復してください。 DSIホストをリセットしてパケットを再送信してください。このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。

表443.エラーの原因と回復方法 （続き）

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
0	18	PE2	D-PHYが誤った制御エラーを報告しています。 D-PHYは、レーン0ラインで不正なライン状態シーケンスを検出しています。	デバイスが期待どおりに動作しません。デバイスとの通信が適切に確立されていません。これは回復不能エラーです。 DSIホストとD-PHYをリセットしてください。このエラーが再発する場合は、デバイスの動作を分析してください。
0	17	PE1	D-PHYがLPDTエラーを報告しています。 D-PHYは、LPDTが8bitの倍数と一致しなかったことを検出しています。	データの受信は信頼できません。D-PHYは回復しますが、デバイスからの受信データが信頼できない可能性があります。 DSIホストをリセットしてRX転送を繰り返すことを推奨します。
0	16	PE0	D-PHYが エスケープ移行エラーを報告しています。 D-PHYは 受信したエスケープ移行コードを認識していません。	D-PHYホストはエスケープ移行コードを認識していません。送信は無視されます。D-PHYホストは回復しますが、システムはRX受信を繰り返す必要があります。
0	15	AE15	このエラーは、エラー付き確認応答パケットから直接取得されます。 デバイスは受信時にプロトコル違反を検出しました。	ディスプレイのドキュメントを参照してください。このエラーがアクティブな場合、デバイスには、このエラーに関する追加情報を報告する別の読み戻しコマンドがあるはずです。追加情報を読んで、適切な措置を講じてください。
0	14	AE14	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、エラー報告にこのビットを使用することを選択しました。	このエラーの考えられる理由についてデバイスのドキュメントを参照し、適切な措置を講じてください。
0	13	AE13	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、送信長がパケット長と一致しないことを報告しています。	この原因として考えられるのは、パケット·ヘッダに複数のエラー（2つ以上）が存在するため、エラー検出に失敗し、デバイスがパケットを破棄しないことです。この場合、パケット·ヘッダが破損しており、デコードの不一致が発生する可能性があります。 パケットを再送信してください。このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。
0	12	AE12	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、受信したパケットの少なくとも1つにVC IDを認識していません。	この原因として考えられるのは、パケット·ヘッダに複数のエラー（2つ以上）が存在するため、エラー検出に失敗し、デバイスがパケットを破棄しないことです。この場合、パケット·ヘッダが破損しており、デコードの不一致が発生する可能性があります。 パケットを再送信してください。このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。
0	11	AE11	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、受信したパケットの少なくとも1つのデータ型を認識していません。	デバイスの機能を確認してください。デバイスがサポートしていないパケットがいくつか存在する可能性があります。 送信を繰り返してください。

表443.エラーの原因と回復方法 （続き）

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
0	10	AE10	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、受信したパケットの少なくとも1つにCRCエラーを検出しています。	最後の確認応答リクエストの後に送信されたロング·パケットの一部で、ペイロードにCRCエラーが含まれている可能性があります。 ペイロードの内容が重要であるならば、パケットを再送信してください。 このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。
0	9	AE9	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、受信したパケットの少なくとも1つに複数ビットのECCエラーを検出しています。	デバイスは、最後の確認応答リクエストの後に送信されたパケットを解釈しません。 パケットが重要であるならば、パケットを再送信してください。 このエラーが再発する場合は、ホストとデバイス間の接続を注意深く分析してください。
0	8	AE8	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、受信したパケットの少なくとも1つに1bitのECCエラーを検出し、修正しています。	アクションは不要です。 デバイスはパケットに確認応答します。 このエラーが再発する場合は、リンクのシグナル·インテグリティまたはノイズ状態を分析してください。
0	7	AE7	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、LP0/LP1検出を通じて回線競合を検出しています。	このエラーにより、低電力データの送受信が破損する可能性があります。 パケットを無視して、再送信してください。 デバイスは自動的に回復します。 このエラーが再発する場合は、デバイスの機能と、ホストとデバイス間の接続を確認してください。 DSI仕様1.1のセクション7.2.1を参照してください。
0	6	AE6	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスが誤った制御エラーを検出しています。	デバイスは以下のいずれかを検出します。 – LP-10（LP要求）の後に、有効なエスケープまたはターンアラウンド·シーケンスの残りが続いていません。 – LP-01（HS 要求）の後に、ブリッジ状態（LP-00）が続いていません。 D-PHY通信が破損しています。このエラーは回復不能です。 DSIホストとD-PHYをリセットしてください。 DSI仕様1.1のセクション7.1.6を参照してください。

表443.エラーの原因と回復方法 （続き）

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
0	5	AE5	エラー付き確認応答パケットに、このエラーが含まれています。 HS受信およびLP送信のディスプレイ･タイムアウト･カウンタが時間切れです。	ホストとデバイスのタイムアウト･カウンタが正しく設定されていない可能性があります。デバイスのHS_TXタイムアウトは、ホストのHS_RXタイムアウトよりも短くする必要があります。ホストのLP_RXタイムアウトは、デバイスのLP_TXタイムアウトよりも長くする必要があります。ホスト設定がデバイスの仕様と一致していることをチェック･確認してください。このエラーは自動的に回復されますが、送信または受信のすべてのパケットが完全であるという保証はありません。このエラーの詳細情報については、DSI仕様1.1のセクション7.2.2を参照してください。
0	4	AE4	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、LPDTが8bit境界に揃えられていないことを報告しています。	デバイスがパケットを正しく受信する保証はありません。 パケットを再送信してください。このエラーの詳細情報については、DSI仕様のセクション7.1.5を参照してください。
0	3	AE3	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスはエスケープ･モード移行コマンドを認識していません。	デバイスはエスケープ･モード移行コードを認識していません。 デバイスの機能を確認してください。このエラーの詳細情報については、DSI仕様のセクション7.1.4を参照してください。 デバイスへの送信を繰り返してください。
0	2	AE2	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、EoTシーケンスが検出されたとき、HS送信が8bit境界で終了しなかったことを検出しています。	デバイスがパケットを正しく受信した保証はありません。再送する必要があります。パケットを再送信してください。このエラーの詳細情報については、DSI仕様1.1のセクション7.1.3を参照してください。
0	1	AE1	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、SoTリーダー･シーケンスが破損していることを検出しています。	デバイスは着信した送信を破棄します。ホストで再送する必要があります。このエラーの詳細情報については、DSI仕様1.1のセクション7.1.2を参照してください。
0	0	AE0	エラー付き確認応答パケットに、このエラーが含まれています。 デバイスは、SoTシーケンスがエラー付きで受信されたが、同期はまだ達成できていることを報告しています。	デバイスは、SoTシーケンス内のシングル･ビット･エラーおよび一部のマルチビット･エラーに対して耐性がありますが、パケットの正確性は損なわれます。パケットの内容が重要であったならば、パケットを再送信してください。このエラーの詳細情報については、DSI仕様1.1のセクション7.1.1を参照してください。
1	19	PBUE	LTDCのペイロード･バッファでアンダーフローが発生しています。	LTDCの周波数がDSIの帯域幅と比較して遅すぎます。LTDCの周波数を上げるか、DSIの帯域幅を減らしてください。

表443.エラーの原因と回復方法 （続き）

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
1	12	GPRXE	汎用読出しFIFOでオーバーフローが発生しています。	読出しFIFOサイズが、最大読み戻しパケットサイズに対して正しく設定されていません。ホストで設定したFIFOに適切なサイズの読出しデータを返すようにデバイスを設定してください。FIFOに格納されたデータは破損しています。 DSIホストをリセットして読出し手順を繰り返してください。
1	11	GPRDE	汎用読出しFIFOでアンダーフローが発生しています。	システムは読出し手順の終了を待たず、FIFOからデータの取り出しを開始します。読出しデータは、完全に受信される前に要求されます。データは破損しています。 DSIホストをリセットして読出し手順を繰り返してください。APBインタフェースを通してデータを読み出す前に、読出し手順が完了していることを確認してください。
1	10	GPTXE	汎用書込みペイロードFIFOでアンダーフローが発生しています。	システムは、それぞれのパケットのペイロードがペイロードFIFOに完全にロードされる前に、パケット·ヘッダを書き込みます。このエラーは回復不能であり、送信されたパケットは破損しています。 DSIホストをリセットして書込み手順を繰り返してください。
1	9	GPWRE	汎用書込みペイロードFIFOでオーバーフローが発生しています。	ペイロードFIFOサイズが、ロング·パケットのペイロード全体を格納するために正しく設定されていません。FIFOに格納されたデータは破損しています。 DSIホストをリセットして書込み手順を繰り返してください。
1	8	GCWRE	汎用コマンドFIFOでオーバーフローが発生しています。	コマンドFIFOサイズが、バースト·パケットのヘッダ全体を格納するために正しく設定されていません。FIFOに格納されたデータは破損しています。 DSIホストをリセットして書込み手順を繰り返してください。
1	7	LPWRE	DPIピクセル·ペイロードFIFOでオーバーフローが発生しています。	コントローラFIFOサイズが、動作解像度に対して正しく設定されていません。ビデオ·モード設定レジスタを確認してください。LTDCのビデオ解像度と一致していなければなりません。ピクセルデータのシーケンスは破損しています。 DSIホストをリセットしてビデオ送信を再開してください。
1	6	EOTPE	ホストが送信終了パケットで終わっていない送信を受信しています。	このエラーは、受信したパケットのデータの完全性に関しては重要ではありません。 デバイスがEoTpパケットの送信をサポートしているかどうかを確認してください。

表443.エラーの原因と回復方法 （続き）

DSIホスト割込みおよびステータスレジスタ	ビット	名前	エラーの原因	推奨されるエラー処理方法
1	5	PSE	ホストが、予想される境界で終わっていない送信を受信しています。	受信データの完全性は保証できません。 DSIホストをリセットして読出し手順を繰り返してください。
1	4	CRCE	ホストが、受信したロング･パケットのペイロードにCRCエラーがあることを報告しています。	受信したペイロードデータは破損しています。DSIホストをリセットして読出し手順を繰り返してください。このエラーが再発する場合は、DSI接続リンクのノイズレベルをチェックしてください。
1	3	ECCME	ホストが、受信したパケットに複数のECCエラーがあったことを報告しています。	受信したパケットは破損しています。DSIホストは、続くパケットすべてを無視します。DSIホストは読出し手順を繰り返す必要があります。
1	2	ECCSE	ホストが、受信したパケットにシングル･ビットエラーがあったことを報告しています。	DSIホストはエラーを修正し、パケットを適切にデコードできるため、このエラーは重要ではありません。 このエラーが再発する場合は、DSI接続リンクのシグナル･インテグリティとノイズレベルをチェックしてください。
1	1	TOLPRX	ホストが、低電力受信用に設定されたタイムアウト･カウンタが時間切れになったことを報告しています。	設定されたタイムアウト･カウンタが終了すると、DSIホストはコントローラ側を自動的にリセットし、通常の動作に回復します。このイベント中に発生したパケット送信は失われます。 このエラーが再発する場合は、タイマの設定に問題がないかをチェックしてください。このタイマは、デバイスによって生成される最大の低電力送信より大きくなければなりません。
1	0	TOHSTX	ホストが、ハイスピード送信用に設定されたタイムアウト･カウンタが時間切れになったことを報告しています。	設定されたタイムアウト･カウンタが終了すると、DSIホストはコントローラ側を自動的にリセットし、通常の動作に回復します。このイベント中に発生したパケット送信は失われます。 このエラーが再発する場合は、タイマの設定に問題がないかをチェックしてください。このタイマは、ホストによって生成される最大のハイスピード送信バーストより大きくなければなりません。
DSIラッパー	10	PLLUF	D-PHYのPLLのロックが解除されています。	このエラーは致命的です。 グラフィカル･サブシステムを再設定し、再起動する必要があります。

44.14 プログラミング手順

DSIホストを操作するには、MIPI® DSI仕様に精通している必要があります。ソフトウェアでプログラム可能なすべてのレジスタは、APBインタフェースを通じてアクセスできます。

44.14.1 プログラミング手順の概要

ビデオ·モードまたはアダプテッド·コマンド·モードのプログラミング手順は、以下の順序に従う必要があります。

1. RCCを設定します（RCCの章を参照）。
   - DSIとLTDC用のクロックを有効にします。
   - LTDC PLLを設定して、オンにし、そのロックを待ちます。
2. オプションでGPIOを設定します（たとえば、ティアリング効果でGPIOの使用が必要な場合など）。
3. オプションでISRを有効にします。
4. LTDCを設定します（LTDCの章を参照）。
   - パネル·タイミングをプログラムします。
   - 関連するレイヤを有効にします。
5. DSIのバイアスをオンにします。
6. セクション44.12.4で説明したように、DSI PLLを設定して、オンにし、そのロックを待ちます。
7. セクション44.14.2で詳しく説明するように、DSIホスト、DSI D_PHY、およびDSIラッパーのD-PHYパラメータを設定し、D-PHYの設定とタイミングを定義します。
8. セクション44.14.3で詳しく説明するように、DSIホストのタイミングを設定します。
9. セクション44.14.4で詳しく説明するように、DSIホストのフロー制御とDBIインタフェースを設定します。
10. セクション44.14.5で詳しく説明するように、DSIホストのLTDCインタフェースを設定します。
11. DSI ホストを、セクション 44.14.6 に詳しく説明されているビデオ·モード、またはセクション44.14.7に詳しく説明されているアダプテッド·コマンド·モードに設定します。
12. DSI_PCTLRのDENビットをセットしてD-PHYを有効にします。
13. DSI_PCTLRのCKENビットをセットしてD-PHYクロック·レーンを有効にします。
14. DSI_CRのENビットをセットしてDSIホストを有効にします。
15. DSI_WCRのDSIENビットをセットしてDSIラッパーを有効にします。
16. オプションで、APB汎用インタフェースを通じてDCSコマンドを送信し、ディスプレイを設定します。
17. LTDCでLTDCを有効にします。
18. DSIラッパーを通じてLTDCフローを開始します（CR.LTDCEN = 1）。

ビデオ·モードでは、LTDCが有効になるとすぐにデータ·ストリーミングが開始されます。

アダプテッド·コマンド·モードでは、CR.LTDCENビットがセットされるとすぐにフレームバッファの更新が開始されます。

44.14.2 D-PHYパラメータの設定

D-PHYでは、通信を開始する前に特定の設定が必要です。設定パラメータは、DSIホストかDSIラッパーのどちらかに格納されます。

D-PHYパラメータの設定

D-PHYは、以下の調整をするために設定する必要があります。

- DSI_DPCSRCレジスタ を介したクロック·レーンのスルーレート
- DSI_DPDL0SRCとDSIDPDL1SRCを介したデータ·レーンのスルーレート
- DSI_DPCLBCRを介したクロック·レーンの周波数帯域
- DSI_DPDL0BCRとDSI_DPBL1BCRを介したデータ·レーンの周波数帯域

DSIホストでのD-PHYパラメータの設定

DSIホストには、D-PHYのタイミング·パラメータの設定とレーン数が保存されます。

起動前に以下のフィールドを設定する必要があります。

- DSI_PCONFRレジスタのデータ·レーンの数
- DSI_CLCRレジスタの自動クロック·レーン制御（ACR）
- DSI_CLCRレジスタのクロック制御（DPCC）
- DSI_CLTCRおよびDSI_DLTCRレジスタのクロック·レーンおよびデータ·レーンのLP/HSおよびHS/LP遷移時間
- DSI_PCONFRレジスタのSTOPウェイト·タイム

44.14.3 DSIホストのタイミングの設定

すべてのプロトコルのタイミングをDSIホストで設定する必要があります。

クロック分周器の設定

2つのクロックが内部的に生成されます。

- タイムアウト·クロック
- TXエスケープ·クロック

タイムアウト·クロックは、HSからLP、LPからHSへの遷移エラーの設定におけるタイミングユニットとして使用されます。その分周比は、DSIホスト·クロック制御レジスタ（DSI_CCR）のタイムアウト·クロック分周（TOCKDIV）フィールドで設定されます。

TXエスケープ·クロックは低電力送信で使用されます。その分周比は、DSIホスト·クロック制御レジスタ（DSI_CCR）のTXエスケープ·クロック分周（TXECKDIV）フィールドでlanebyteclockを基準として設定されます。クロックの代表的な値は、20MHz前後でなければなりません。

タイムアウトの設定

セクション44.8で説明されているタイムアウト管理のタイミングは、DSIホスト·タイムアウトカウンタ設定レジスタ（DSI_TCCR0～DSI_TCCR5）で設定します。

44.14.4 フロー制御とDBIインタフェースの設定

フロー制御は、DSIホスト·プロトコル設定レジスタ（DSI_PCR）によって設定されます。設定パラメータは以下のとおりです。

- CRC受信イネーブル（CRCRXEビット）
- ECC受信イネーブル（ECCRXEビット）
- BTAイネーブル（BTAEビット）
- EoTp受信イネーブル（ETRXEビット）
- EoTp送信イネーブル（ETTXEビット）

これらの値は、DSIディスプレイとの通信に使用されるプロトコルに依存します。

汎用 DBI インタフェースに使用される仮想チャネル ID は、DSI ホスト汎用 VCID レジスタ（DSI_GVCIDR）の仮想チャネルID（VCID）フィールドで設定する必要があります。

すべてのDCSコマンドはそれぞれのタイプに応じて、ハイスピードと低電力のいずれかで送受信できます。それぞれに対する専用の設定ビットは、DSI ホスト·コマンド·モード設定レジスタ（DSI_CMCR）でプログラムする必要があります。

DSIホスト·コマンド·モード設定レジスタ（DSI_CMCR）で、パケットまたはティアリング効果イベントに対する確認応答リクエストも設定する必要があります。

44.14.5 DSIホストのLTDCインタフェースの設定

DSIホストは、ビデオ·モードやアダプテッド·コマンド·モードでは、LTDCを通してシステムとインタフェースを取るため、DSIラッパーは、その間で低レベルのインタフェースを実行します。

DSIラッパーにプログラムされたパラメータは、LTDCおよびDSIホストにプログラムされたパラメータと一致している必要があります。

以下のフィールドを設定する必要があります。

- DSIホストLTDC VCIDレジスタ（DSI_LVCIDR）の仮想チャネルID（VCID）フィールドの仮想チャネルID
- DSI ホスト LTDC カラー·コーディング·レジスタ（DSI_LCOLCR）のカラー·コーディング（COLC）フィールドと、DSIラッパー設定レジスタ（DSI_WCFGR）のカラー多重化（COLMUX）
- 軽度パケットが18bitモードで使用される場合は、DSIホストLTDCカラー·コーディング·レジスタ（DSI_LCOLCR）の軽度パケット·イネーブル（LPE）ビットを設定する必要があります。
- DSIホストLTDC極性設定レジスタ（DSI_LPCR）のHSync極性（HSP）ビットのHSYNC極性
- DSIホストLTDC極性設定レジスタ（DSI_LPCR）のVSync極性（VSP）ビットとDSIラッパー設定レジスタ（DSI_WCFGR）のVSync極性（VSPOL）ビットのVSYNC極性
- DSIホストLTDC極性設定レジスタ（DSI_LPCR）のデータイネーブル極性（DEP）ビットDATA ENABLE極性

44.14.6 ビデオ･モードの設定

ビデオ･モードの設定では、コマンド送信のための低電力時のコントローラの動作、ビデオ送信のタイプ（バースト･モードまたは非バースト･モード）、パネルの水平および垂直タイミングを定義します。

- ビデオ送信モードを選択し、プロセッサがDSI リンクを通じてビデオ･ラインをどのように転送するかを定義します。
  - DSI_VMCRで低電力遷移を設定して、十分な時間がある場合に低電力に移行できるビデオ期間を定義します。
  - コントローラがフレームの最後にペリフェラル確認応答メッセージを要求する必要があるかどうかを設定します（DSI_VMCR.FBTAAE）。
  - コマンドを低電力で送信するかどうかを設定します（DSI_VMCR.LPE）。
- ビデオ･モードのタイプを選択します
  - バースト･モード：

    ビデオ･モード･タイプ（DSI_VMCR.VMT）を値2'b1xで設定します。

    ビデオ･パケットのサイズ（DSI_VPCR.VPSIZE）を、ピクセル単位で測定されるアクティブ･ライン期間のサイズで設定します。

    レジスタDSI_VCCRとDSI_VNPCRはDSIホストによって無視されます。
  - 非バースト･モード：

    ビデオ･モード･タイプ（DSI_VMCR.VMT）を、同期パルスの送信を有効にする場合は2'b00で、同期イベントの送信を有効にする場合は2'b01で設定します。

    ビデオ･パケットのサイズ（DSI_VPCR.VPSIZE）を、単一のパケットで送信するピクセル数で設定します。この値の選択は、接続されたペリフェラルの利用可能なメモリ（データが最初に格納される場合）、またはDSIホストのFIFO用に選択するメモリによって変わります。

    チャンク数（DSI_VCCR.NUMC）をビデオ･ラインごとに送信するパケット数で設定します。VPSIZE × NUMCの値は、ビデオの1ラインあたりのピクセル数です。ただし、NUMCが0の場合（マルチパケットが無効になります）を除きます。NUMCを1に設定した場合、ラインあたり1つのパケットしかありませんが、チャンクの一部であり、その後にnullパケットが続く可能性があります。

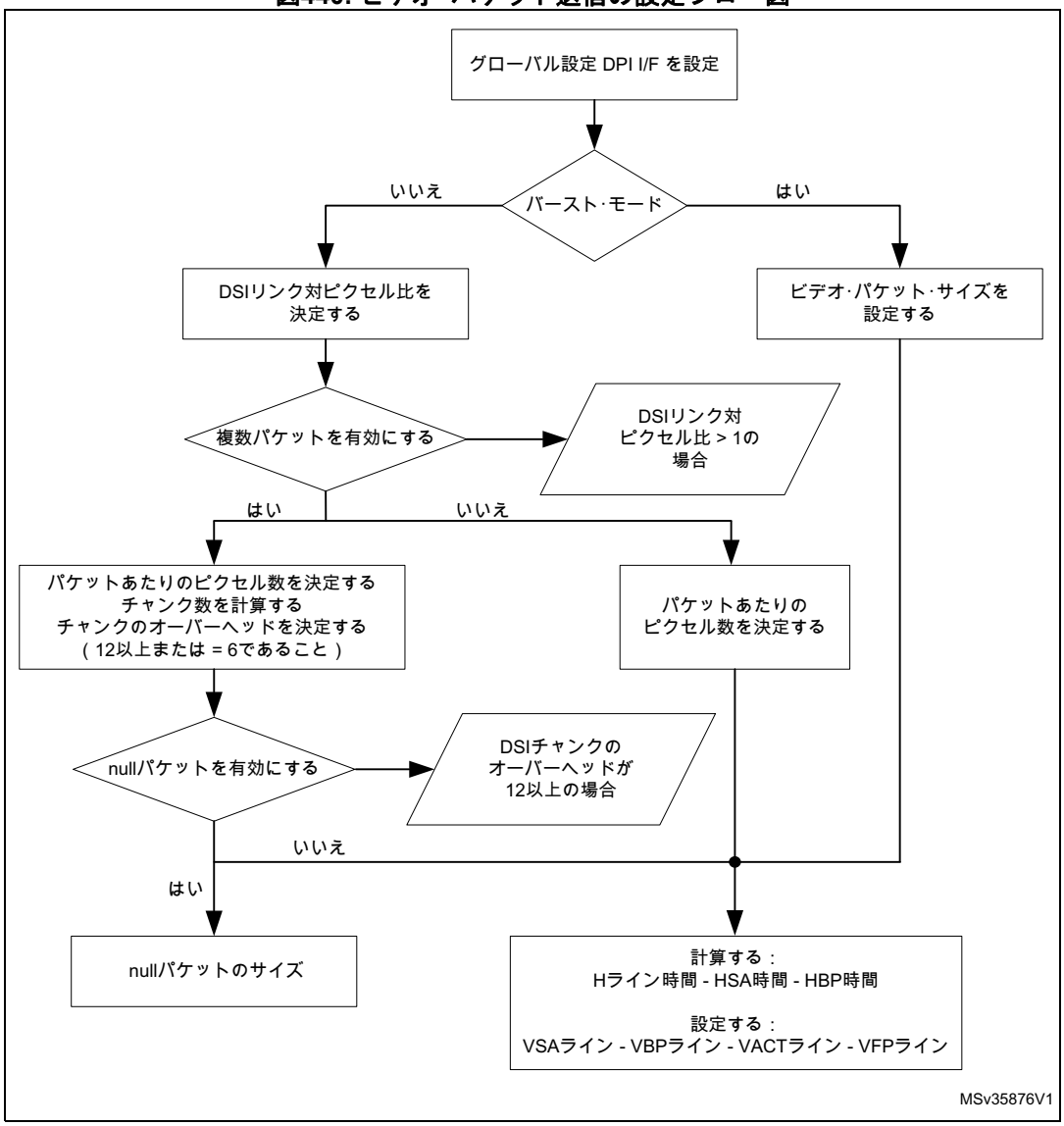
    nullパケットのサイズ（DSI_VNPCR.NPSIZE）を、チャンクの一部として挿入されるnullパケットのサイズで設定します。これを0に設定すると、nullパケットは無効となります。
- ビデオ水平タイミング設定を以下のように設定します。
  - 水平ライン時間（DSI_VLCR.HLINE）を、レーン･バイト･クロックのサイクルで計測されるLTDCビデオ･ラインにかかる時間で設定します（500MHzのクロック･レーンの場合、レーン･バイト･クロック周期は8ns）。LTDCクロック周期とレーン･バイト･クロック周期が倍数関係でない場合、DSI_VLCR.HLINEをプログラムする値は丸める必要があります。設定値の丸めにより、ライン間にタイミングの不一致がもたらされます。DSIホストが低電力に移行しないように設定されている場合、このタイミングのずれは各ラインで蓄積され、フレームの終端に向かってかなりの量の不一致が生じます。こうなる理由は、DSIホストはLTDCインタフェースで水平同期イベントが発生したときにブランキング･パケットを送信するため、新しいラインごとに再同期できないからです。しかし、蓄積された不一致は、フレームの最終ラインで消滅しなければなりません。DSI仕様によれば、垂直同期イベントで新しいフレームが始まるとき、リンクは常に低電力に戻り、同期を取り戻さなければならないことになっています。累積されたタイミングの不一致が最終ラインの低電力時間より大きければ、誤動作が発生します。この現象は、DSIホストが1ラインにつき1回低電力になるように設定することで回避できます。
  - 水平同期時間（DSI_VHSACR.HSA）を、レーン･バイト･クロックのサイクル（通常8nsの周期）で計測されるLTDC水平同期アクティブ期間にかかる時間で設定します。

– 水平バック･ポーチ時間（DSI_VHBPCR.HBP）を、レーン･バイト･クロック･サイクル（通常8nsの周期）で計測されるLTDC水平バック･ポーチ期間にかかる時間で設定します。このパラメータの計算には、特に注意を払う必要があります。

- 垂直ライン設定を以下のように定義します。
  – 垂直同期時間（DSI_VVSACR.VSA）は、LTDC垂直同期アクティブ期間に存在するライン数を設定します。
  – 垂直バック･ポーチ時間（DSI_VVBPCR.VBP）は、LTDC垂直バック･ポーチ期間に存在するライン数を設定します。
  – 垂直フロント･ポーチ時間（DSI_VVFPCR.VFP）は、LTDC垂直フロント･ポーチ期間に存在するライン数を設定します。
  – 垂直アクティブ時間（DSI_VVACR.VA）は、LTDC垂直アクティブ期間に存在するライン数を設定します。

DPIパケット送信の設定手順を 図440に示します。

図440. ビデオ･パケット送信の設定フロー図



ビデオ設定の例

以下にビデオ･パケット送信の設定例を示します。

ビデオ解像度：

- PCLK周期 = 50ns
- HSA = 8 PCLK
- HBP = 8 PCLK
- HACT = 480 PCLK
- HFP = 24 PCLK
- VSA = 2ライン
- VBP = 2ライン
- VACT = 640ライン
- VFP = 4ライン

設定手順：

- ビデオ送信モードの設定：
  a) 低電力遷移を設定します。
  DSI_VMCR[13:8] = 6'b111111（すべてのビデオ期間でLPを有効にするため）
  b) DSI_VMCR.FBTAAE = 1（DSIホストが各フレームの終わりにペリフェラルから確認応答メッセージをリクエストするため）
- バースト･モードを使用するには、次の手順に従います。
  DSI_VMCR.VMT = 2'b1x
  DSI_VPCR.VPSIZE = 480
- 水平タイミングの設定：
  – DSI_VLCR.HLINE =
  (HSA + HBP + HACT + HFP) × (PCLK周期 / クロック･レーン･バイト周期) =
  (8 + 8 + 480 + 24) * (50 / 8) = 3250
  – DSI_VHSACR.HSA = HSA × (PCLK周期 / クロック･レーン･バイト周期) =
  8 *(50 /8)=50
  – DSI_VHBPCR.HBP = HBP × (PCLK周期 / クロック･レーン･バイト周期) =
  8 *(50 /8)=50
- 垂直ライン設定：
  – DSI_VVSACR.VSA = 2
  – DSI_VVBPCR.VBP = 2
  – DSI_VVFPCR.VFP = 4
  – DSI_VVACR.VA = 640

44.14.7 アダプテッド･コマンド･モードの設定

アダプテッド･コマンド･モードでは、以下のパラメータを設定する必要があります。

- 書込みメモリ･コマンドの最大許容サイズを定義する DSI ホスト LTDC コマンド設定レジスタ（DSI_LCCR）のコマンド･サイズ（CMDSIZE）フィールド
- DSI ラッパー設定レジスタ（DSI_WCFGR）のティアリング効果ソース（TESRC）ビットとオプションのティアリング効果極性（TEPOL）ビット
- ティアリング効果イベントを受信するたびにディスプレイを自動的に更新する必要がある場合の、DSIラッパー設定レジスタ（DSI_WCFGR）の自動リフレッシュ（AR）ビット

44.14.8 ビデオ･モード･パターン･ジェネレータの設定

DSIホストは水平/垂直カラーバーのないカラーバー･パターンや、D-PHYのBERテストパターンを何らの刺激もなしに送信できます。

図441に、テストパターンを送信するためのプログラミング･シーケンスを示します。

1. DSI_MCRレジスタを設定して、ビデオ･モードを有効にします。DSI_VMCR.VMTを使用してビデオ･モード･タイプを設定します。
2. DSI_LCOLCRレジスタを設定します。
3. 図442に示すレジスタ（グレーの部分が転送されるピクセルを示す）を使用してフレームを設定します。
4. パターン生成モード（DSI_VMCR.PGM）とパターン方向（DSI_VMCR.PGO）を設定し、それらを有効にします（DSI_VMCR.PGE）。

図441. テストパターンを送信するためのプログラミング･シーケンス

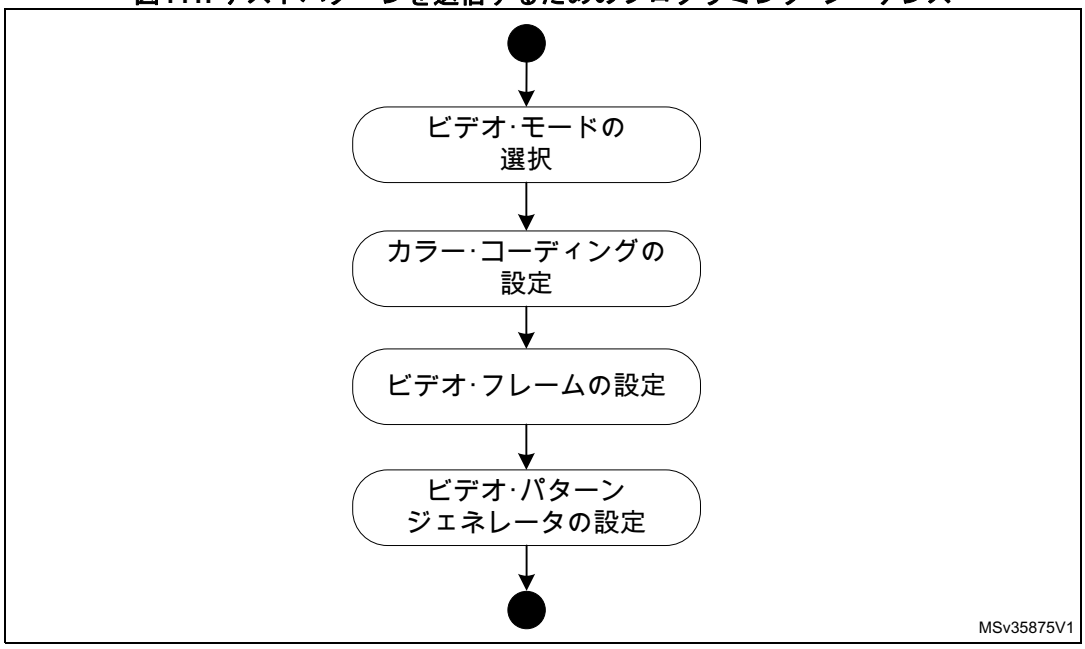
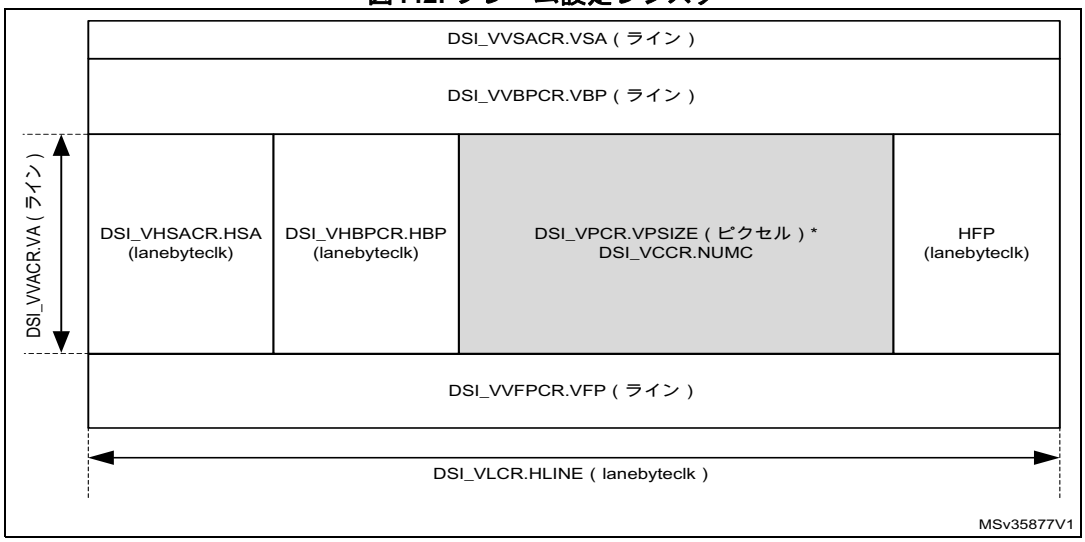


図442. フレーム設定レジスタ



注： **ペイロードのピクセル数は、表433に示した値の倍数に制約されます。**

44.14.9 ULPMの管理

ULPMに移行･終了するソフトウェアを設定するには、以下の2通りの方法があります。

- D-PHY PLLが動作している状態でULPMに移行および終了する（より高速なプロセス）
- D-PHY PLLがオフになっている状態でULPMに移行および終了する（電力消費の点でより効率的なプロセス）

ULPMシーケンスに対するクロック管理

ULPM管理のステートマシンは、D-PHYが提供するレーン･バイト･クロックで動作しています。

D-PHYはクロック･レーンがULPM状態でない時にのみレーン･バイト･クロックを提供するため、ULPMモード移行シーケンスを開始する前に、DSIホストのレーン･バイト･クロックのソースを切り替えることが必須です。

レーン･バイト･クロックのソースは、RCCによって制御されており、以下のクロックである可能性があります。

- D-PHYによって提供されるレーン･バイト･クロック（ULPM以外の全てのモードの場合）
- システムPLLによって生成されるクロック（ULPMの場合）

ULPMに移行するプロセス･フロー

クロック･レーンとデータ･レーンの両方でULPMに移行するには、以下の手順で詳しく説明するプロセスを実行します。

1. DSIホストの初期ステータスを確認します。
   - DSI_PCTLR[2:1] = 2'h3
   - DSI_WRPCR.PLLEN = 1'h1およびDSI_WRPCR.REGEN = 1'h1
   - DSI_PUCR[3:0] = 4'h0
   - DSI_PTTCR[3:0] = 4'h0
   - すべてのアクティブなレーンが停止状態で、D-PHY PLLがロックされていることを確認します。

     1レーン構成：DSI_PSR[6:4] = 3'h3、DSI_PSR[1] = 1'h0、DSI_WISR.PLLS = 1'h1

     2レーン構成：DSI_PSR[8:4] = 5'h1B、DSI_PSR[1] = 1'h0、DSI_WISR.PLLS = 1'h1
2. RCCのレーン･バイト･クロックのソースをD-PHYからシステムPLLに切り替えます。
3. DSI_PUCR[3:0] = 4'h5に設定して、データ･レーンとクロック･レーンでULPMに移行します。
4. D-PHYアクティブ･レーンがULPMに移行するまで待ちます。
   - 1レーン構成：DSI_PSR[6:1] = 6'h00
   - 2レーン構成：DSI_PSR[8:1] = 8'h00

   これでDSIホストはULPMになってています。
5. DSI_WRPCR.PLLEN = 1'b0にセットしてD-PHY PLLをオフにします。

ULPMを終了するプロセス･フロー

クロック･レーンとデータ･レーンの両方でULPMを終了するには、以下の手順で詳しく説明するプロセス･フローを実行します。

1. すべてのアクティブなレーンがULPMになっていることを確認します。
   - 1レーン構成：DSI_PSR[6:1] = 6'h00
   - 2レーン構成：DSI_PSR[8:1] = 8'h00
2. DSI_WRPCR.PLLEN = 1'b1に設定してD-PHY PLLをオンにします。
3. D-PHY PLLがロックされるまで待ちます。
   - DSI_WISR.PLLS = 1'b1
4. ULPM要求ビットをデアサートせずに、DSI_PUCR[3:0] = 4'hFを設定することによってULPM終了ビットをアサートします。
5. すべてのアクティブなレーンがULPMを終了するまで待ちます。
   - 1レーン構成：
     DSI_PSR[5] = 1'b1
     DSI_PSR[3] = 1'b1
   - 2レーン構成：
     DSI_PSR[8] = 1'b1
     DSI_PSR[5] = 1'b1
     DSI_PSR[3] = 1'b1
6. 1ms待ちます。
7. DSI_PUCR[3:0] = 4'h0に設定することで、ULPM要求とULPM終了ビットをデアサートします。
8. RCCのレーン･バイト･クロックのソースをシステムPLLからD-PHYに切り替えます。
9. これでDSIホストは停止状態で、D-PHY PLLはロックされています。
   - 1レーン構成：
     DSI_PSR[6:4] = 3'h3
     DSI_PSR[1] = 1'h0
     DSI_WRPCR.PLLEN = 1'b1
   - 2レーン構成：
     DSI_PSR[8:4] = 5'h1B
     DSI_PSR[1] = 1'h0
     DSI_WRPCR.PLLEN = 1'b1

44.15 DSIホスト･レジスタ

44.15.1 DSIホスト･バージョン･レジスタ（DSI_VR）

アドレスオフセット：0x0000

リセット値：0x3134 312A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VERSION[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VERSION[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **VERSION[31:0]**：DSIホストのバージョン

この読出し専用レジスタには、DSIホストのバージョンが格納されています。

44.15.2 DSIホスト制御レジスタ（DSI_CR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：有効化

このビットは、DSIホストをパワーアップ･モードまたはリセットのいずれかに設定します。

0：DSIホストは無効です（リセット状態）。

1：DSIホストは有効です。

44.15.3 DSIホスト･クロック制御レジスタ（DSI_CCR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOCKDIV[7:0]								TXECKDIV[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **TOCKDIV[7:0]**：タイムアウト･クロック分周

このフィールドは、HSからLP、LPからHSへの遷移エラーの設定におけるタイミングユニットとして使用されるタイムアウト･クロックの分周比を示します。

ビット7:0 **TXECKDIV[7:0]**：TXエスケープ･クロック分周

このフィールドは、TXエスケープ･クロック･ソース（lanebyteclk）の分周比を示します。0と1の値はTX_ESCクロックの生成を停止させます。

44.15.4 DSIホストLTDC VCIDレジスタ（DSI_LVCIDR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCID[1:0]	
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **VCID[1:0]**：仮想チャネルID

これらのビットにより、LTDCインタフェースのトラフィック用の仮想チャネルIDを設定します。

44.15.5 DSIホストLTDCカラー･コーディング･レジスタ（DSI_LCOLCR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPE	Res.	Res.	Res.	Res.	COLC[3:0]			
							rw					rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **LPE**：軽度パケット･イネーブル

このビットは、18bit設定に対する軽度パック済みバリアントを有効にします。

0：軽度パケット･バリアントは無効です。

1：軽度パケット･バリアントは有効です。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **COLC[3:0]**：カラー･コーディング

このフィールドでは、DPIカラー･コーディングを設定します。

0000：16bit設定1

0001：16bit設定2

0010：16bit設定3

0011：18bit設定1

0100：18bit設定2

0101：24bit

その他：予約済み

44.15.6 DSIホストLTDC極性設定レジスタ（DSI_LPCR）

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSP	VSP	DEP
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **HSP**：HSYNC極性

このビットでは、HSYNCピンの極性を設定します。

0：HSYNCピン はアクティブハイです（デフォルト）。

1：VSYNCピンはアクティブローです。

ビット1 **VSP**：VSYNC極性

このビットでは、VSYNCピンの極性を設定します。

0：シャットダウン･ピンはアクティブハイです（デフォルト）。

1：シャットダウン･ピンはアクティブローです。

ビット0 **DEP**：データイネーブル極性

このビットでは、データイネーブル･ピンの極性を設定します。

0：データイネーブル･ピンはアクティブハイです（デフォルト）。

1：データイネーブル･ピンはアクティブローです。

44.15.7 DSIホスト低電力モード設定レジスタ（DSI_LPMCR）

アドレスオフセット：0x0018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPSIZE[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLPSIZE[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **LPSIZE[7:0]**：最大パケット･サイズ

このフィールドは、低電力モードでのコマンドの送信に使用されます。VSA、VBP、VFPの領域の間でラインに適合する最大パケット･サイズ（バイト単位）を定義します。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **VLPSIZE[7:0]**：VACT最大パケット･サイズ

このフィールドは、低電力モードでのコマンドの送信に使用されます。VACT領域の期間でラインに適合する最大パケット･サイズ（バイト単位）を定義します。

44.15.8 DSIホスト･プロトコル設定レジスタ（DSI_PCR）

アドレスオフセット：0x002C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETTXLPE	CRCRXE	ECCRXE	BTAE	ETRXE	ETTXE
										rw	rw	rw	rw	rw	rw

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **ETTXLPE**：低電力でのEoTp送信の有効化

このビットは、低電力でのEoTP送信を有効にします。

0：低電力でのEoTp送信は無効です。

1：低電力でのEoTp送信は有効です。

ビット4 **CRCRXE**：CRC受信の有効化

このビットは、CRC受信とエラー報告を有効にします。

0：CRC受信は無効です。

1：CRC受信は有効です。

ビット3 **ECCRXE**：ECC受信の有効化

このビットは、ECC受信、エラー訂正と報告を有効にします。

0：ECC受信は無効です。

1：ECC受信は有効です。

ビット2 **BTAE**：バスターンアラウンド有効化

このビットは、バスターンアラウンド（BTA）要求を有効にします。

0：バスターンアラウンド要求は無効です。

1：バスターンアラウンド要求は有効です。

ビット1 **ETRXE**：EoTp受信の有効化

このビットは、EoTp受信を有効にします。

0：EoTp受信は無効です。

1：EoTp受信は有効です。

ビット0 **ETTXE**：EoTp送信の有効化

このビットは、EoTP送信を有効にします。

0：EoTp送信は無効です。

1：EoTp送信は有効です。

44.15.9 DSIホスト汎用VCIDレジスタ（DSI_GVCIDR）

アドレスオフセット：0x0030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCIDTX[1:0]	
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCIDRx[1:0]	
														rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **VCIDTX[1:0]**：送信用仮想チャネルID

このフィールドは、汎用パケットが自動的に生成され、送信される汎用インタフェース仮想チャネルの識別子を示します。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **VCIDRX[1:0]**：受信用仮想チャネルID

このフィールドは、汎用インタフェースの読み戻し仮想チャネルの識別子を示します。

44.15.10 DSIホスト･モード設定レジスタ（DSI_MCR）

アドレスオフセット：0x0034

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDM
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **CMDM**：コマンド･モード

このビットにより、DSIホストをビデオ･モードまたはコマンド･モードのいずれかに設定します。

0：DSIホストはビデオ･モードに設定されます。

1：DSIホストはコマンド･モードに設定されます。

44.15.11 DSIホスト･ビデオ･モード設定レジスタ（DSI_VMCR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PGO	Res.	Res.	Res.	PGM	Res.	Res.	Res.	PGE
							rw				rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPCE	FBTAAE	LPHFPE	LPHBPE	LPVAE	LPVFPE	LPVBPE	LPVSAE	Res.	Res.	Res.	Res.	Res.	Res.	VMT[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **PGO**：パターン･ジェネレータの向き

このビットでは、カラーバーの向きを設定します。

0：垂直カラーバー

1：水平カラーバー

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **PGM**：パターン･ジェネレータのモード

このビットでは、パターン･ジェネレータのモードを設定します。

0：カラーバー（水平または垂直）

1：BERパターン（垂直のみ）

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **PGE**：パターン･ジェネレータの有効化

このビットはビデオ･モード･パターン･ジェネレータを有効にします。

0：パターン･ジェネレータは無効です。

1：パターン･ジェネレータは有効です。

ビット15 **LPCE**：低電力コマンドの有効化

このビットは、コマンド送信を低電力モードでのみ有効にします。

0：低電力モードでのコマンド送信は無効です。

1：低電力モードでのコマンド送信は有効です。

ビット14 **FBTAAE**：フレーム･バスターンアラウンド確認応答の有効化

このビットは、フレームの終わりでの確認応答のリクエストを有効にします。

0：フレームの終わりでの確認応答は無効です。

1：フレームの終わりでの確認応答は有効です。

ビット13 **LPHFPE**：低電力水平フロント･ポーチの有効化

このビットは、タイミングが許せば、水平フロント･ポーチ（HFP）期間内での低電力への復帰を有効にします。

0：HFP期間内での低電力への復帰は無効です。

1：HFP期間内での低電力への復帰は有効です。

ビット12 **LPHBPE**：低電力水平バック･ポーチの有効化

このビットは、タイミングが許せば、水平バック･ポーチ（HBP）期間内での低電力への復帰を有効にします。

0：HBP期間内での低電力への復帰は無効です。

1：HBP期間内での低電力への復帰は有効です。

ビット11 **LPVAE**：低電力垂直アクティブ有効化

このビットは、タイミングが許せば、垂直アクティブ（VACT）期間内での低電力への復帰を有効にします。

0：VACT内での低電力への復帰は無効です。

1：VACT内での低電力への復帰は有効です。

ビット10 **LPVFPE**：低電力垂直フロント･ポーチ有効化

このビットは、タイミングが許せば、垂直フロント･ポーチ（VFP）期間内での低電力への復帰を有効にします。

0：VFP内での低電力への復帰は無効です。

1：VFP内での低電力への復帰は有効です。

ビット9 **LPVBPE**：低電力垂直バック･ポーチ有効化

このビットは、タイミングが許せば、垂直バック･ポーチ（VBP）期間内での低電力への復帰を有効にします。

0：VBP内での低電力への復帰は無効です。

1：VBP内での低電力への復帰は有効です。

ビット8 **LPVSAE**：低電力垂直同期アクティブ有効化

このビットは、タイミングが許せば、垂直同期時間（VSA）期間内での低電力への復帰を有効にします。

0：VSA内での低電力への復帰は無効です。

1：VSA内での低電力への復帰は有効です

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **VMT[1:0]**：ビデオ･モード･タイプ

このフィールドでは、ビデオ･モード送信タイプを設定します。

00：同期パルスでの非バースト

01：同期イベントでの非バースト

1x：バースト･モード

44.15.12 DSIホスト･ビデオ･パケット設定レジスタ（DSI_VPCR）

アドレスオフセット：0x003C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	VPSIZE[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **VPSIZE[13:0]**：ビデオ･パケット･サイズ

このフィールドでは、単一のビデオ･パケットのピクセル数を設定します。
軽度パック済みでない18bitのデータ型の場合、この数値は4の倍数でなければなりません。
YCbCrのデータ型の場合、DSI仕様に記載されているように2の倍数でなければなりません。

44.15.13 DSIホスト･ビデオ･チャンク設定レジスタ（DSI_VCCR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	NUMC[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **NUMC[12:0]**：チャンクの数

このレジスタでは、ライン期間中に送信されるチャンクの数を設定します（チャンクはビデオ･パケットとnullパケットで構成されます）。
0または1に設定すると、ビデオ･ラインは単一のパケットで送信されます。
1に設定すると、NPSIZE > 0の場合、パケットはチャンクの一部であり、nullパケットが続きます。そうでない場合、各ビデオ･ラインの送信には複数のチャンクが使用されます。

44.15.14 DSIホスト･ビデオnullパケット設定レジスタ（DSI_VNPCR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	NPSIZE[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **NPSIZE[12:0]**：nullパケットのサイズ

このフィールドでは、nullパケットのバイト数を設定します。
0に設定すると、nullパケットは無効となります。

44.15.15 DSIホスト･ビデオHSA設定レジスタ（DSI_VHSACR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HSA[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **HSA[11:0]**：水平同期アクティブ時間

このフィールドでは、水平同期アクティブ期間をレーン･バイト･クロックのサイクルで設定します。

44.15.16 DSIホスト･ビデオHBP設定レジスタ（DSI_VHBPCR）

アドレスオフセット：0x004C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HBP[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **HBP[11:0]**：水平バック･ポーチ時間

このフィールドでは、水平バック･ポーチ期間をレーン･バイト･クロックのサイクルで設定します。

44.15.17 DSIホスト･ビデオ･ライン設定レジスタ（DSI_VLCR）

アドレスオフセット：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	HLINE[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **HLINE[14:0]**：水平ライン時間

このフィールドでは、レーン･バイト･クロックのサイクルでカウントされる合計の水平ライン期間（HSA+HBP+HACT+HFP）を設定します。

44.15.18 DSIホスト･ビデオVSA設定レジスタ（DSI_VVSACR）

アドレスオフセット：0x0054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VSA[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VSA[9:0]**：垂直同期アクティブ時間

このフィールドでは、水平ラインの数で計測される垂直同期アクティブ期間を設定します。

44.15.19 DSIホスト･ビデオVBP設定レジスタ（DSI_VVBPCR）

アドレスオフセット：0x0058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBP[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VBP[9:0]**：垂直バック･ポーチ時間

このフィールドでは、水平ラインの数で計測される垂直バック･ポーチ期間を設定します。

44.15.20 DSIホスト･ビデオVFP設定レジスタ（DSI_VVFPCR）

アドレスオフセット：0x005C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VFP[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VFP[9:0]**：垂直フロント･ポーチ時間

このフィールドでは、水平ラインの数で計測される垂直フロント･ポーチ期間を設定します。

44.15.21 DSIホスト･ビデオVA設定レジスタ（DSI_VVACR）

アドレスオフセット：0x0060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	VA[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **VA[13:0]**：垂直アクティブ時間

このフィールドでは、水平ラインの数で計測される垂直アクティブ期間を設定します。

44.15.22 DSIホストLTDCコマンド設定レジスタ（DSI_LCCR）

アドレスオフセット：0x0064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMDSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CMDSIZE[15:0]**：コマンドサイズ

このフィールドでは、ピクセル単位で計測されるLTDC書込みメモリ･コマンドの最大許容サイズを設定します。LTDCから得られたデータの自動分割は恒久的に有効です。

44.15.23 DSIホスト･コマンド･モード設定レジスタ（DSI_CMCR）

アドレスオフセット：0x0068

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MRDPS	Res.	Res.	Res.	Res.	DLWTX	DSR0TX	DSW1TX	DSW0TX
							rw					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	GLWTX	GSR 2TX	GSR 1TX	GSR 0TX	GSW 2TX	GSW 1TX	GSW 0TX	Res.	Res.	Res.	Res.	Res.	Res.	ARE	TEARE
	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **MRDPS**：最大読出しパケット･サイズ

このビットでは、最大読出しパケット･サイズ･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **DLWTX**：DCSロング書込みの送信

このビットでは、DCSロング書込みパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット18 **DSR0TX**：DCSショート読出し0パラメータの送信

このビットでは、0パラメータのDCSショート読出しパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット17 **DSW1TX**：DCSショート読出し1パラメータの送信

このビットでは、1パラメータのDCSショート読出しパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット16 **DSW0TX**：DCSショート書込み0パラメータの送信

このビットでは、0パラメータのDCSショート書込みパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **GLWTX**：汎用ロング書込みの送信

このビットでは、汎用ロング書込みパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット13 **GSR2TX**：汎用ショート読出し2パラメータの送信

このビットでは、2パラメータの汎用ショート読出しパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット12 **GSR1TX**：汎用ショート読出し1パラメータの送信

このビットでは、1パラメータの汎用ショート読出しパケット･コマンドの送信タイプを設定します。

0：ハイスピード

1：低電力

ビット11 **GSR0TX**：汎用ショート読出し0パラメータの送信
このビットでは、0パラメータの汎用ショート読出しパケット･コマンドの送信タイプを設定します。
0：ハイスピード
1：低電力

ビット10 **GSW2TX**：汎用ショート書込み2パラメータの送信
このビットでは、2パラメータの汎用ショート書込みパケット･コマンドの送信タイプを設定します。
0：ハイスピード
1：低電力

ビット9 **GSW1TX**：汎用ショート書込み1パラメータの送信
このビットでは、1パラメータの汎用ショート書込みパケット･コマンドの送信タイプを設定します。
0：ハイスピード
1：低電力

ビット8 **GSW0TX**：汎用ショート書込み0パラメータの送信
このビットでは、0パラメータの汎用ショート書込みパケット･コマンドの送信タイプを設定します。
0：ハイスピード
1：低電力

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **ARE**：確認応答リクエストの有効化
このビットは、各パケット送信後の確認応答リクエストを有効にします。
0：確認応答リクエストは無効です。
1：確認応答リクエストは有効です。

ビット0 **TEARE**：ティアリング効果の確認応答リクエストの有効化
このビットは、ティアリング効果の確認応答リクエストを有効にします。
0：ティアリング効果の確認応答リクエストは無効です。
1：ティアリング効果の確認応答リクエストは有効です。

44.15.24 DSIホスト汎用ヘッダ設定レジスタ（DSI_GHCR）

アドレスオフセット：0x006C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WCMSB[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WCLSB[7:0]								VCID[1:0]		DT[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **WCMSB[7:0]**：ワードカウントのMSB
このフィールドでは、ロング･パケットの場合はヘッダ･パケットのワード数の最上位バイトを、ショート･パケットの場合はデータ1を設定します。

ビット15:8 **WCLSB[7:0]**：ワードカウントのLSB
このフィールドでは、ロング･パケットの場合はヘッダ･パケットのワード数の最下位バイトを、ショート･パケットの場合はデータ0を設定します。

ビット7:6 **VCID[1:0]**：チャネル
このフィールドでは、ヘッダ･パケットの仮想チャネルIDを設定します。

ビット5:0 **DT[5:0]**：タイプ
このフィールドでは、ヘッダ･パケットのパケット･データ･タイプを設定します。

44.15.25 DSIホスト汎用ペイロード･データ･レジスタ（DSI_GPDR）

アドレスオフセット：0x0070

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA4 [7:0]								DATA3 [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA2 [7:0]								DATA1 [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **DATA4[7:0]**：ペイロード･バイト4
このフィールドは、パケット･ペイロードのバイト4を示します。

ビット23:16 **DATA3[7:0]**：ペイロード･バイト3
このフィールドは、パケット･ペイロードのバイト3を示します。

ビット15:8 **DATA2[7:0]**：ペイロード･バイト2
このフィールドは、パケット･ペイロードのバイト2を示します。

ビット7:0 **DATA1[7:0]**：ペイロード･バイト1
このフィールドは、パケット･ペイロードのバイト1を示します。

44.15.26 DSIホスト汎用パケット･ステータスレジスタ（DSI_GPSR）

アドレスオフセット：0x0074

リセット値：0x00050015

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBF	PBE	CMDBF	CMDBE
												r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCB	PRDFF	PRDFE	PWRFF	PWRFE	CMDFF	CMDFE
									r	r	r	r	r	r	r

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PBF**：ペイロード･バッファ･フル
このビットは、汎用ペイロード内部バッファのフル状態を示します。
0：ペイロード内部バッファはフルではありません。
1：ペイロード内部バッファはフルです。

ビット18 **PBE**：ペイロード･バッファ･エンプティ

このビットは、汎用ペイロード内部バッファのエンプティ状態を示します。

0：ペイロード内部バッファはエンプティではありません。

1：ペイロード内部バッファはエンプティです。

ビット17 **CMDBF**：コマンド･バッファ･フル

このビットは、汎用コマンド内部バッファのフル状態を示します。

0：コマンド内部バッファはフルではありません。

1：コマンド内部バッファはフルです。

ビット16 **CMDBE**：コマンド･バッファ･エンプティ

このビットは、汎用ペイロード内部バッファのエンプティ状態を示します。

0：ペイロード内部バッファはフルではありません。

1：ペイロード内部バッファはフルです。

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **RCB**：読出しコマンド･ビジー

このビットは、読出しコマンドが発行されるとセットされ、レスポンス全体がFIFOに格納されるとクリアされます。

0：読出しコマンドは実行中ではありません。

1：読出しコマンドは実行中です。

ビット5 **PRDFF**：ペイロード読出しFIFOフル

このビットは、汎用読出しペイロードFIFOのフル状態を示します。

0：読出しペイロードFIFOはフルではありません。

1：読出しペイロードFIFOはフルです。

ビット4 **PRDFE**：ペイロード読出しFIFOエンプティ

このビットは、汎用読出しペイロードFIFOのエンプティ状態を示します。

0：読出しペイロードFIFOはエンプティではありません。

1：読出しペイロードFIFOはエンプティです。

ビット3 **PWRFF**：ペイロード書込みFIFOフル

このビットは、汎用書込みペイロードFIFOのフル状態を示します。

0：書込みペイロードFIFOはフルではありません。

1：書込みペイロードFIFOはフルです。

ビット2 **PWRFE**：ペイロード書込みFIFOエンプティ

このビットは、汎用書込みペイロードFIFOのエンプティ状態を示します。

0：書込みペイロードFIFOはエンプティではありません。

1：書込みペイロードFIFOはエンプティです。

ビット1 **CMDFF**：コマンドFIFOフル

このビットは、汎用コマンドFIFOのフル状態を示します。

0：書込みペイロードFIFOはフルではありません。

1：書込みペイロードFIFOはフルです。

ビット0 **CMDFE**：コマンドFIFOエンプティ

このビットは、汎用コマンドFIFOのエンプティ状態を示します。

0：書込みペイロードFIFOはエンプティではありません。

1：書込みペイロードFIFOはエンプティです。

44.15.27 DSIホスト･タイムアウトカウンタ設定レジスタ0（DSI_TCCR0）

アドレスオフセット：0x0078

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HSTX_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPRX_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **HSTX_TOCNT[15:0]**：ハイスピード送信タイムアウトカウンタ

このフィールドでは、ハイスピード送信タイムアウト競合検出（TOCKDIVサイクルで計測）をトリガするタイムアウトカウンタを設定します。

非バースト･モードを使用し、1ラインデータの終了から次のライン同期開始までの間にハイスピードから低電力へ切り替えて戻るのに十分な時間がない場合、DSIリンクはフレームごとに1回低電力状態に戻るので、TOCKDIVとHSTX_TOCNTを以下のように設定してください。

HSTX_TOCNT × レーン･バイト･クロック周期 × TOCKDIV ≥ 1フレーム･データの送信時間 × (1 + 10%)

バースト･モードでは、RGBピクセル･パケットは時間圧縮されているので、スキャンライン中に時間が残ります。したがって、バースト･モードで、1ラインデータの終了から次のライン同期開始までの間にハイスピードから低電力へ切り替えて戻るのに十分な時間がある場合、DSIリンクは節電するためにこの時間の間に低電力モードに移行して戻ることができます。そのためには、TOCKDIVとHSTX_TOCNTを以下のように設定してください。

HSTX_TOCNT × レーン･バイト･クロック周期 × TOCKDIV ≥ 1ライン･データの送信時間 × (1 + 10%)

ビット15:0 **LPRX_TOCNT[15:0]**：低電力受信タイムアウトカウンタ

このフィールドでは、低電力受信タイムアウト競合検出（TOCKDIVサイクルで計測）をトリガするタイムアウトカウンタを設定します。

44.15.28 DSIホスト･タイムアウトカウンタ設定レジスタ1（DSI_TCCR1）

アドレスオフセット：0x007C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSRD_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HSRD_TOCNT[15:0]**：ハイスピード読出しタイムアウトカウンタ

このフィールドでは、DSIホストがハイスピード読出し操作を送信した後、リンクをそのまま維持する期間を設定します。この期間はlanebyteclkのサイクルで計測されます。カウントは、D-PHYが 停止状態になったときに開始され、割込みは発生しません。

44.15.29 DSIホスト･タイムアウトカウンタ設定レジスタ2（DSI_TCCR2）

アドレスオフセット：0x0080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPRD_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LPRD_TOCNT[15:0]**：低電力読出しタイムアウトカウンタ

このフィールドでは、DSIホストが低電力読出し操作を送信した後、リンクをそのまま維持する期間を設定します。この期間はlanebyteclkのサイクルで計測されます。カウントは、D-PHYが 停止状態になったときに開始され、割込みは発生しません。

44.15.30 DSIホスト･タイムアウトカウンタ設定レジスタ3（DSI_TCCR3）

アドレスオフセット：0x0084

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSWR_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **PM**：Prespモード

このビットを1に設定すると、次の両方の条件が満たされた場合に、HSWR_TOCNTによって引き起こされるペリフェラルレスポンスタイムアウトがコマンド･モードでLTDCのフレームごとに一度だけ使用されるようになります。

–dpivsync_edpiwmsが立ち上がり、そして立ち下がった。

–コマンド･モードでLTDCから発出されたパケットが送信され、そのFIFOが再び空になった。

このシナリオでは、送信準備のできた汎用インタフェースからのトラフィックがあっても、LTDCコマンドでないリクエストはD-PHYに送信されず、トラフィックは停止状態に戻されます。これが起こると、PRESP_TOカウンタが起動され、それが完了したときにのみ、コントローラは、準備ができた他のトラフィックを送信します。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HSWR_TOCNT[15:0]**：ハイスピード書込みタイムアウトカウンタ

このフィールドでは、DSIホストがハイスピード書込み操作を送信した後、リンクを非アクティブに保持する期間を設定します。この期間はlanebyteclkのサイクルで計測されます。カウントは、D-PHYが 停止状態になったときに開始され、割込みは発生しません。

44.15.31 DSIホスト･タイムアウトカウンタ設定レジスタ4（DSI_TCCR4）

アドレスオフセット：0x0088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPWR_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **LPWR_TOCNT[15:0]**：低電力書込みタイムアウトカウンタ

このフィールドでは、DSIホストが低電力書込み操作を送信した後、リンクをそのまま維持する期間を設定します。この期間はlanebyteclkのサイクルで計測されます。カウントは、D-PHYが 停止状態になったときに開始され、割込みは発生しません。

44.15.32 DSIホスト･タイムアウトカウンタ設定レジスタ5（DSI_TCCR5）

アドレスオフセット：0x008C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BTA_TOCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BTA_TOCNT[15:0]**：バスターンアラウンド･タイムアウトカウンタ

このフィールドでは、DSIホストがバスターンアラウンドを完了した後、リンクをそのまま維持する期間を設定します。この期間はlanebyteclkのサイクルで計測されます。カウントは、D-PHYが 停止状態になったときに開始され、割込みは発生しません。

44.15.33 DSIホスト･クロック･レーン設定レジスタ（DSI_CLCR）

アドレスオフセット：0x0094

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACR	DPCC
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **ACR**：自動クロック･レーン制御

このビットにより、時間の許す限り自動メカニズムがクロック･レーンでのクロックの提供を停止できるようになります。

0：自動クロック･レーン制御は無効です。

1：自動クロック･レーン制御は有効です。

ビット0 **DPCC**：D-PHYクロック制御

このビットは、D-PHYのクロック状態を制御します。

0：クロック･レーンは低電力モードになります。

1：クロック･レーンはハイスピード･モードで動作します。

44.15.34 DSIホスト･クロック･レーン･タイマ設定レジスタ（DSI_CLTCR）

アドレスオフセット：0x0098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HS2LP_TIME [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LP2HS_TIME [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **HS2LP_TIME[9:0]**：ハイスピードから低電力への移行時間

このフィールドでは、ハイスピードから低電力の送信に移行するためにD-PHYクロック･レーンでかかる最大時間（レーン･バイト･クロック･サイクルで計測します）を設定します。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **LP2HS_TIME[9:0]**：低電力からハイスピードへの移行時間

このフィールドでは、低電力からハイスピードの送信に移行するためにD-PHYクロック･レーンでかかる最大時間（レーン･バイト･クロック･サイクルで計測します）を設定します。

44.15.35 DSIホスト･データ･レーン･タイマ設定レジスタ（DSI_DLTCR）

アドレスオフセット：0x009C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HS2LP_TIME [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LP2HS_TIME [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **HS2LP_TIME[9:0]**：ハイスピードから低電力への移行時間

このフィールドでは、ハイスピードから低電力の送信に移行するためにD-PHYデータ･レーンでかかる最大時間（レーン･バイト･クロック･サイクルで計測します）を設定します。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **LP2HS_TIME[9:0]**：低電力からハイスピードへの移行時間

このフィールドでは、低電力からハイスピードの送信に移行するためにD-PHYデータ･レーンでかかる最大時間（レーン･バイト･クロック･サイクルで計測します）を設定します。

44.15.36 DSIホストPHY制御レジスタ（DSI_PCTLR）

アドレスオフセット：0x00A0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKE	DEN	Res.
													rw	rw	

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CKE**：クロックの有効化

このビットは、D-PHYクロック･レーン･モジュールを有効にします。

0：D-PHYクロック･レーン･モジュールは無効です。

1：D-PHYクロック･レーン･モジュールは有効です。

ビット1 **DEN**：デジタル有効化

このビットを0にセットすると、D-PHYのデジタル･セクションがリセット状態に置かれます。

0：D-PHYのデジタル･セクションはリセット状態です。

1：D-PHYの デジタルセクションは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

44.15.37 DSIホストPHY設定レジスタ（DSI_PCONFR）

アドレスオフセット：0x00A4

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SW_TIME[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	NL[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **SW_TIME[7:0]**：STOP待ち時間

このフィールドでは、STOP状態の後にハイスピード送信をリクエストする最小待ち時間を設定します。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **NL[1:0]**：レーン数

このフィールドでは、アクティブなデータ・レーンの数を設定します。

00：1つのデータ・レーン（レーン0）

01：2つのデータ・レーン（レーン0と1）- リセット値

その他：予約済み

44.15.38 DSIホストPHY ULPS制御レジスタ（DSI_PUCR）

アドレスオフセット：0x00A8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UEDL	URDL	UECL	URCL
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **UEDL**：データ・レーンでのULPS終了

すべてのアクティブなデータ・レーンでULPSモードを終了します。

0：終了リクエストなし。

1：すべてのアクティブなデータ・レーンURDLでULPSモードを終了します。

ビット2 **URDL**：データ･レーンでのULPSリクエスト
すべてのアクティブなデータ･レーンでULPSモードをリクエストします。
0：ULPSリクエストなし
1：すべてのアクティブなデータ･レーン UECL でULPSモードをリクエストします。

ビット1 **UECL**：クロック･レーンでのULPS終了
クロック･レーンでULPSモードを終了します。
0：終了リクエストなし。
1：クロック･レーンでULPSモードを終了します。

ビット0 **URCL**：クロック･レーンでのULPSリクエスト
クロック･レーンでULPSモードをリクエストします。
0：ULPSリクエストなし
1：クロック･レーンでULPSモードをリクエストします。

44.15.39 DSIホストPHY TXトリガ設定レジスタ（DSI_PTTCR）

アドレスオフセット：0x00AC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TX_TRIG[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TX_TRIG[3:0]**：送信トリガ
エスケープ･モード送信トリガ0-3。
常にTX_TRIGの1bitだけがアサートされます。

44.15.40 DSIホストPHYステータスレジスタ（DSI_PSR）

アドレスオフセット：0x00B0

リセット値：0x0000 1528

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UAN1	PSS1	RUE0	UAN0	PSS0	UANC	PSSC	PD	Res.
							r	r	r	r	r	r	r	r	

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **UAN1**：レーン1でULPS非アクティブ
このビットはulpsactivenot1lane D-PHY信号の状態を示します。

ビット7 **PSS1**：レーン1でPHY停止状態
このビットはphystopstate1lane D-PHY信号の状態を示します。

ビット6 **RUE0**：レーン0でRX ULPSエスケープ
このビットはrxulpsesc0lane D-PHY信号の状態を示します。

ビット5 **UAN0**：レーン1でULPS非アクティブ
このビットはulpsactivenot0lane D-PHY信号の状態を示します。

ビット4 **PSS0**：レーン0でPHY停止状態
このビットはphystopstate0lane D-PHY信号の状態を示します。

ビット3 **UANC**：クロック・レーンでULPS非アクティブ
このビットは、ulpsactivenotclklane D-PHY信号の状態を示します。

ビット2 **PSSC**：クロック・レーンでPHY停止状態
このビットは、phystopstateclklane D-PHY信号の状態を示します。

ビット1 **PD**：PHYの方向
このビットは、phydirection D-PHY信号の状態を示します。

ビット0 予約済みであり、リセット値に保持する必要があります。

44.15.41 DSIホスト割込みおよびステータスレジスタ0（DSI_ISR0）

アドレスオフセット：0x00BC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PE4	PE3	PE2	PE1	PE0
											r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AE15	AE14	AE13	AE12	AE11	AE10	AE9	AE8	AE7	AE6	AE5	AE4	AE3	AE2	AE1	AE0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **PE4**：PHYエラー 4
このビットは レーン0からのLP1競合エラー ErrContentionLP1を示します。

ビット19 **PE3**：PHYエラー 3
このビットは レーン0からのLP0競合エラー ErrContentionLP0を示します。

ビット18 **PE2**：PHYエラー 2
このビットは レーン0からのErrControlエラーを示します。

ビット17 **PE1**：PHYエラー 1
このビットは レーン0からのErrSyncEsc低電力送信同期エラーを示します。

ビット16 **PE0**：PHYエラー 0
このビットは レーン0からのErrEscエスケープ移行エラーを示します。

ビット15 **AE15**：確認応答エラー 15
このビットは、確認応答エラー報告からDSIプロトコル違反を取得します。

ビット14 **AE14**：確認応答エラー 14
このビットは、確認応答エラー報告から予約済み（デバイスに固有）を取得します。

ビット13 **AE13**：確認応答エラー 13
このビットは、確認応答エラー報告から 無効な送信長を取得します。

ビット12 **AE12**：確認応答エラー 12
このビットは、確認応答エラー報告からDSI VC ID不正を取得します。

ビット11 **AE11**：確認応答エラー 11
このビットは、確認応答エラー報告から 認識されないDSIデータ型を取得します。

ビット10 **AE10**：確認応答エラー 10
このビットは、確認応答エラー報告からチェックサム･エラー（ロング･パケットのみ）を取得します。

ビット9 **AE9**：確認応答エラー 9
このビットは、確認応答エラー報告から 複数ビットのECCエラー（検出、未修正）を取得します。

ビット8 **AE8**：確認応答エラー 8
このビットは、確認応答エラー報告からシングル･ビットのECCエラー（検出、修正済み）を取得します。

ビット7 **AE7**：確認応答エラー 7
このビットは、確認応答エラー報告から予約済み（デバイスに固有）を取得します。

ビット6 **AE6**：確認応答エラー 6
このビットは、確認応答エラー報告から誤った制御エラーを取得します。

ビット5 **AE5**：確認応答エラー 5
このビットは、確認応答エラー報告からペリフェラル･タイムアウトエラーを取得します。

ビット4 **AE4**：確認応答エラー 4
このビットは、確認応答エラー報告からLP送信同期エラーを取得します。

ビット3 **AE3**：確認応答エラー 3
このビットは、確認応答エラー報告からエスケープ･モード移行コマンド･エラーを取得します。

ビット2 **AE2**：確認応答エラー 2
このビットは、確認応答エラー報告からEoT同期エラーを取得します。

ビット1 **AE1**：確認応答エラー 1
このビットは、確認応答エラー報告からSoT同期エラーを取得します。

ビット0 **AE0**：確認応答エラー 0
このビットは、確認応答エラー報告からSoTエラーを取得します。

44.15.42 DSIホスト割込みおよびステータスレジスタ1（DSI_ISR1）

アドレスオフセット：0x00C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBUE	Res.	Res.	Res.
												r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	GPRXE	GPRDE	GPTXE	GPWRE	GCWRE	LPWRE	EOTPE	PSE	CRCE	ECCME	ECCSE	TOLPRX	TOHSTX
			r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PBUE**：ペイロード･バッファ･アンダーフローエラー
このビットは、ビデオ･モード用のDSIパケットを構築するためにペイロードを読み出す際にアンダーフローが発生したことを示します。

ビット18:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **GPRXE**：汎用ペイロード受信エラー

このビットは、汎用インタフェース･パケットの読戻し中にペイロードFIFOが一杯になり、受信データが破損していることを示します。

ビット11 **GPRDE**：汎用ペイロード読出しエラー

このビットは、DCSのデータ読出し中にペイロードFIFOがエンプティになり、インタフェースに送信されたデータが破損していることを示します。

ビット10 **GPTXE**：汎用ペイロード送信エラー

このビットは、汎用インタフェース･パケットの構築中にペイロードFIFOがエンプティになり、破損したデータが送信されていることを示します。

ビット9 **GPWRE**：汎用ペイロード書込みエラー

このビットは、システムが汎用インタフェースを通してペイロード･データを書き込もうとしたが、FIFOが一杯であることを示します。したがって、ペイロードは書き込まれていません。

ビット8 **GCWRE**：汎用コマンド書込みエラー

このビットは、システムが汎用インタフェースを通してコマンドを書き込もうとしたが、FIFOが一杯であることを示します。したがって、コマンドは書き込まれていません。

ビット7 **LPWRE**：LTDCペイロード書込みエラー

このビットは、DPIのピクセル･ライン格納中にペイロードFIFOが一杯になり、格納したデータが破損していることを示します。

ビット6 **EOTPE**：EoTpエラー

このビットは、着信したペリフェラル送信の最後にEoTpパケットが受信されていないことを示します。

ビット5 **PSE**：パケット･サイズ･エラー

このビットは、パケット受信中にパケット･サイズ･エラーが検出されたことを示します。

ビット4 **CRCE**：CRCエラー

このビットは、受信したパケットのペイロードにCRCエラーが検出されたことを示します。

ビット3 **ECCME**：ECC複数ビットエラー

このビットは、受信したパケットにECCの複数のエラーが検出されたことを示します。

ビット2 **ECCSE**：ECCシングル･ビットエラー

このビットは、受信したパケットにECCシングルエラーが検出され、修正されたことを示します。

ビット1 **TOLPRX**：低電力受信タイムアウト

このビットは、低電力受信タイムアウトカウンタが終了に達し、競合が検出されたことを示します。

ビット0 **TOHSTX**：ハイスピード送信タイムアウト

このビットは、ハイスピード送信タイムアウトカウンタが終了に達し、競合が検出されたことを示します。

44.15.43 DSIホスト割込みイネーブル･レジスタ0（DSI_IER0）

アドレスオフセット：0x00C4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PE4IE	PE3IE	PE2IE	PE1IE	PE0IE
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AE15IE	AE14IE	AE13IE	AE12IE	AE11IE	AE10IE	AE9IE	AE8IE	AE7IE	AE6IE	AE5IE	AE4IE	AE3IE	AE2IE	AE1IE	AE0IE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **PE4IE**：PHYエラー 4割込みイネーブル
このビットは、PHYエラー 4での割込み生成を有効にします。
0：PHYエラー 4での割込みは無効です。
1：PHYエラー 4での割込みは有効です。

ビット19 **PE3IE**：PHYエラー 3割込みイネーブル
このビットは、PHYエラー 4 での割込み生成を有効にします。
0：PHYエラー 3での割込みは無効です。
1：PHYエラー 3での割込みは有効です。

ビット18 **PE2IE**：PHYエラー 2割込みイネーブル
このビットは、PHYエラー 2での割込み生成を有効にします。
0：PHYエラー 2での割込みは無効です。
1: PHYエラー 2での割込みは有効です。

ビット17 **PE1IE**：PHYエラー 1割込みイネーブル
このビットは、PHYエラー 1での割込み生成を有効にします。
0：PHYエラー 1での割込みは無効です。
1: PHYエラー 1での割込みは有効です。

ビット16 **PE0IE**：PHYエラー 0割込みイネーブル
このビットは、PHYエラー 0での割込み生成を有効にします。
0：PHYエラー 0での割込みは無効です。
1: PHYエラー 0での割込みは有効です。

ビット15 **AE15IE**：確認応答エラー 15割込みイネーブル
このビットは、確認応答エラー 15での割込み生成を有効にします。
0：確認応答エラー 15での割込みは無効です。
1: 確認応答エラー 15での割込みは有効です。

ビット14 **AE14IE**：確認応答エラー 14割込みイネーブル
このビットは、確認応答エラー 14での割込み生成を有効にします。
0：確認応答エラー 14での割込みは無効です。
1: 確認応答エラー 14での割込みは有効です。

ビット13 **AE13IE**：確認応答エラー 13割込みイネーブル
このビットは、確認応答エラー 13での割込み生成を有効にします。
0：確認応答エラー 13での割込みは無効です。
1: 確認応答エラー 13での割込みは有効です。

ビット12 **AE12IE**：確認応答エラー 12割込みイネーブル
このビットは、確認応答エラー 12での割込み生成を有効にします。
0：確認応答エラー 12での割込みは無効です。
1: 確認応答エラー 12での割込みは有効です。

ビット11 **AE11IE**：確認応答エラー 11割込みイネーブル
このビットは、確認応答エラー 11での割込み生成を有効にします。
0：確認応答エラー 11での割込みは無効です。
1: 確認応答エラー 11での割込みは有効です。

ビット10 **AE10IE**：確認応答エラー 10割込みイネーブル
このビットは、確認応答エラー 10での割込み生成を有効にします。
0：確認応答エラー 10での割込みは無効です。
1: 確認応答エラー 10での割込みは有効です。

ビット9 **AE9IE**：確認応答エラー 9割込みイネーブル
このビットは、確認応答エラー 9での割込み生成を有効にします。
0：確認応答エラー 9での割込みは無効です。
1: 確認応答エラー 9での割込みは有効です。

ビット8 **AE8IE**：確認応答エラー 8割込みイネーブル
このビットは、確認応答エラー 8での割込み生成を有効にします。
0：確認応答エラー 8での割込みは無効です。
1: 確認応答エラー 8での割込みは有効です。

ビット7 **AE7IE**：確認応答エラー 7割込みイネーブル
このビットは、確認応答エラー 7での割込み生成を有効にします。
0：確認応答エラー 7での割込みは無効です。
1: 確認応答エラー 7での割込みは有効です。

ビット6 **AE6IE**：確認応答エラー 6割込みイネーブル
このビットは、確認応答エラー 6での割込み生成を有効にします。
0：確認応答エラー 6での割込みは無効です。
1: 確認応答エラー 6での割込みは有効です。

ビット5 **AE5IE**：確認応答エラー 5割込みイネーブル
このビットは、確認応答エラー 5での割込み生成を有効にします。
0：確認応答エラー 5での割込みは無効です。
1: 確認応答エラー 5での割込みは有効です。

ビット4 **AE4IE**：確認応答エラー 4割込みイネーブル
このビットは、確認応答エラー 4での割込み生成を有効にします。
0：確認応答エラー 4での割込みは無効です。
1: 確認応答エラー 4での割込みは有効です。

ビット3 **AE3IE**：確認応答エラー 3割込みイネーブル
このビットは、確認応答エラー 3での割込み生成を有効にします。
0：確認応答エラー 3での割込みは無効です。
1: 確認応答エラー 3での割込みは有効です。

ビット2 **AE2IE**：確認応答エラー 2割込みイネーブル
このビットは、確認応答エラー 2での割込み生成を有効にします。
0：確認応答エラー 2での割込みは無効です。
1: 確認応答エラー 2での割込みは有効です。

ビット1 **AE1IE**：確認応答エラー 1割込みイネーブル
このビットは、確認応答エラー 1での割込み生成を有効にします。
0：確認応答エラー 1での割込みは無効です。
1: 確認応答エラー 1での割込みは有効です。

ビット0 **AE0IE**：確認応答エラー 0割込みイネーブル
このビットは、確認応答エラー 0での割込み生成を有効にします。
0：確認応答エラー 0での割込みは無効です。
1: 確認応答エラー 0での割込みは有効です。

44.15.44 DSIホスト割込みイネーブル･レジスタ1 （DSI_IER1）

アドレスオフセット：0x00C8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBUEIE	Res.	Res.	Res.
												rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	GPRXEIE	GPRDEIE	GPTXEIE	GPWREIE	GCWREIE	LPWREIE	EOTPEIE	PSEIE	CRCEIE	ECCMEIE	ECCSEIE	TOLPRXIE	TOHSTXIE
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PBUEIE**：ペイロード･バッファ･アンダーフローエラー割込みイネーブル
このビットは、ペイロード･バッファ･アンダーフローエラーでの割込み生成を有効にします。
0：ペイロード･バッファ･アンダーフローエラーでの割込みは無効です。
1：ペイロード･バッファ･アンダーフローエラーでの割込みは有効です。

ビット18:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **GPRXEIE**：汎用ペイロード受信エラー割込みイネーブル
このビットは、汎用ペイロード受信エラーでの割込み生成を有効にします。
0：汎用ペイロード受信エラーでの割込みは無効です。
1：汎用ペイロード受信エラーでの割込みは有効です。

ビット11 **GPRDEIE**：汎用ペイロード読出しエラー割込みイネーブル
このビットは、汎用ペイロード読出しエラーでの割込み生成を有効にします。
0：汎用ペイロード読出しエラーでの割込みは無効です。
1：汎用ペイロード読出しエラーでの割込みは有効です。

ビット10 **GPTXEIE**：汎用ペイロード送信エラー割込みイネーブル
このビットは、汎用ペイロード送信エラーでの割込み生成を有効にします。
0：汎用ペイロード送信エラーでの割込みは無効です。
1：汎用ペイロード送信エラーでの割込みは有効です。

ビット9 **GPWREIE**：汎用ペイロード書込みエラー割込みイネーブル
このビットは、汎用ペイロード書込みエラーでの割込み生成を有効にします。
0：汎用ペイロード書込みエラーでの割込みは無効です。
1：汎用ペイロード書込みエラーでの割込みは有効です。

ビット8 **GCWREIE**：汎用コマンド書込みエラー割込みイネーブル
このビットは、汎用コマンド書込みエラーでの割込み生成を有効にします。
0：汎用コマンド書込みエラーでの割込みは無効です。
1：汎用コマンド書込みエラーでの割込みは有効です。

ビット7 **LPWREIE**：LTDCペイロード書込みエラー割込みイネーブル
このビットは、LTDCペイロード書込みエラーでの割込み生成を有効にします。
0：LTDCペイロード書込みエラーでの割込みは無効です。
1：LTDCペイロード書込みエラーでの割込みは有効です。

ビット6 **EOTPEIE**：EoTpエラー割込みイネーブル
このビットは、EoTpエラーでの割込み生成を有効にします。
0：EoTpエラーでの割込みは無効です。
1：EoTpエラーでの割込みは有効です。

ビット5 **PSEIE**：パケット･サイズ･エラー割込みイネーブル
このビットは、パケット･サイズ･エラーでの割込み生成を有効にします。
0：パケット･サイズ･エラーでの割込みは無効です。
1：パケット･サイズ･エラーでの割込みは有効です。

ビット4 **CRCEIE**：CRCエラー割込みイネーブル
このビットは、CRCエラーでの割込み生成を有効にします。
0：CRCエラーでの割込みは無効です。
1：CRCエラーでの割込みは有効です。

ビット3 **ECCMEIE**：ECC複数ビットエラー割込みイネーブル
このビットは、ECC複数ビットエラーでの割込み生成を有効にします。
0：ECC複数ビットエラーでの割込みは無効です。
1：ECC複数ビットエラーでの割込みは有効です。

ビット2 **ECCSEIE**：ECCシングル･ビットエラー割込みイネーブル
このビットは、ECCシングル･ビットエラーでの割込み生成を有効にします。
0：ECCシングル･ビットエラーでの割込みは無効です。
1：ECCシングル･ビットエラーでの割込みは有効です。

ビット1 **TOLPRXIE**：低電力受信タイムアウト割込みイネーブル
このビットは、低電力受信タイムアウトでの割込み生成を有効にします。
0：低電力受信タイムアウトでの割込みは無効です。
1：低電力受信タイムアウトでの割込みは有効です。

ビット0 **TOHSTXIE**：ハイスピード送信タイムアウト割込みイネーブル
このビットは、ハイスピード送信タイムアウトでの割込み生成を有効にします。
0：ハイスピード送信タイムアウトでの割込みは無効です。
1：ハイスピード送信タイムアウトでの割込みは有効です。

44.15.45 DSIホスト強制割込みレジスタ0（DSI_FIR0）

アドレスオフセット：0x00D8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPE4	FPE3	FPE2	FPE1	FPE0
											w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FAE15	FAE14	FAE13	FAE12	FAE11	FAE10	FAE9	FAE8	FAE7	FAE6	FAE5	FAE4	FAE3	FAE2	FAE1	FAE0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **FPE4**：強制DPHYエラー 4
このビットに1を書き込むと、強制的にPHYエラー 4になります。

ビット19 **FPE3**：強制DPHYエラー 3
このビットに1を書き込むと、強制的にPHYエラー 3になります。

ビット18 **FPE2**：強制DPHYエラー 2
このビットに1を書き込むと、強制的にPHYエラー 2になります。

ビット17 **FPE1**：強制DPHYエラー 1
このビットに1を書き込むと、強制的にPHYエラー 1になります。

ビット16 **FPE0**：強制DPHYエラー 0
このビットに1を書き込むと、強制的にPHYエラー 0になります。

ビット15 **FAE15**：強制確認応答エラー 15
このビットに1を書き込むと、強制的に確認応答エラー 15になります。

ビット14 **FAE14**：強制確認応答エラー 14
このビットに1を書き込むと、強制的に確認応答エラー 14になります。

ビット13 **FAE13**：強制確認応答エラー 13
このビットに1を書き込むと、強制的に確認応答エラー 13になります。

ビット12 **FAE12**：強制確認応答エラー 12
このビットに1を書き込むと、強制的に確認応答エラー 12になります。

ビット11 **FAE11**：強制確認応答エラー 11
このビットに1を書き込むと、強制的に確認応答エラー 11になります。

ビット10 **FAE10**：強制確認応答エラー 10
このビットに1を書き込むと、強制的に確認応答エラー 10になります。

ビット9 **FAE9**：強制確認応答エラー 9
このビットに1を書き込むと、強制的に確認応答エラー 9になります。

ビット8 **FAE8**：強制確認応答エラー 8
このビットに1を書き込むと、強制的に確認応答エラー 8になります。

ビット7 **FAE7**：強制確認応答エラー 7
このビットに1を書き込むと、強制的に確認応答エラー 7になります。

ビット6 **FAE6**：強制確認応答エラー 6
このビットに1を書き込むと、強制的に確認応答エラー 6になります。

ビット5 **FAE5**：強制確認応答エラー 5
このビットに1を書き込むと、強制的に確認応答エラー 5になります。

ビット4 **FAE4**：強制確認応答エラー 4
このビットに1を書き込むと、強制的に確認応答エラー 4になります。

ビット3 **FAE3**：強制確認応答エラー 3
このビットに1を書き込むと、強制的に確認応答エラー 3になります。

ビット2 **FAE2**：強制確認応答エラー 2
このビットに1を書き込むと、強制的に確認応答エラー 2になります。

ビット1 **FAE1**：強制確認応答エラー 1
このビットに1を書き込むと、強制的に確認応答エラー 1になります。

ビット0 **FAE0**：強制確認応答エラー 0
このビットに1を書き込むと、強制的に確認応答エラー 0になります。

44.15.46 DSIホスト強制割込みレジスタ1（DSI_FIR1）

アドレスオフセット：0x00DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPBUE	Res.	Res.	Res.
												w			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FGP RXE	FGP RDE	FGP TXE	FGP WRE	FGC WRE	FLP WRE	FE OTPE	FPSE	FCRCE	FECC ME	FECC SE	FTOLPR X	FTOHST X
			w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **FPBUE**：強制ペイロード･バッファ･アンダーフローエラー
このビットに1を書き込むと、強制的にペイロード･アンダーフローエラーになります。

ビット18:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **FGPRXE**：強制汎用ペイロード受信エラー
このビットに1を書き込むと、強制的に汎用ペイロード受信エラーになります。

ビット11 **FGPRDE**：強制汎用ペイロード読出しエラー
このビットに1を書き込むと、強制的に汎用ペイロード読出しエラーになります。

ビット10 **FGPTXE**：強制汎用ペイロード送信エラー
このビットに1を書き込むと、強制的に汎用ペイロード送信エラーになります。

ビット9 **FGPWRE**：強制汎用ペイロード書込みエラー
このビットに1を書き込むと、強制的に汎用ペイロード書込みエラーになります。

ビット8 **FGCWRE**：強制汎用コマンド書込みエラー
このビットに1を書き込むと、強制的に汎用コマンド書込みエラーになります。

ビット7 **FLPWRE**：強制LTDCペイロード書込みエラー
このビットに1を書き込むと、強制的にLTDCペイロード書込みエラーになります。

ビット6 **FEOTPE**：強制EoTpエラー
このビットに1を書き込むと、強制的にEoTpエラーになります。

ビット5 **FPSE**：強制パケット･サイズ･エラー
このビットに1を書き込むと、強制的にパケット･サイズ･エラーになります。

ビット4 **FCRCE**：強制CRCエラー
このビットに1を書き込むと、強制的にCRCエラーになります。

ビット3 **FECCME**：強制ECC複数ビットエラー
このビットに1を書き込むと、強制的にECC複数ビットエラーになります。

ビット2 **FECCSE**：強制ECCシングル･ビットエラー
このビットに1を書き込むと、強制的にECCシングル･ビットエラーになります。

ビット1 **FTOLPRX**：強制低電力受信タイムアウト
このビットに1を書き込むと、強制的に低電力受信タイムアウトになります。

ビット0 **FTOHSTX**：強制ハイスピード送信タイムアウト
このビットに1を書き込むと、強制的にハイスピード送信タイムアウトになります。

44.15.47 DSIホスト･データ･レーン･タイマ読出し設定レジスタ（DSI_DLTRCR）

アドレスオフセット：0x00F4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MRD_TIME[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **MRD_TIME[14:0]**：最大読出し 時間
このフィールドでは、読出しコマンドの実行に必要な最大時間をレーン･バイト･クロック･サイクルで設定します。このレジスタは、読出しコマンドが実行されていないときにのみ変更できます。

44.15.48 DSIホスト･ビデオ･シャドウ制御レジスタ（DSI_VSCR）

アドレスオフセット：0x0100

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
							rw								rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **UR**：更新レジスタ

1に設定すると、LTDCレジスタが補助レジスタにコピーされます。コピー後、このビットは自動クリアされます。

0：更新はリクエストされません。

1：レジスタの更新がリクエストされます。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：有効化

1に設定すると、DSIホストLTDCインタフェースは補助レジスタからアクティブな設定を受け取ります。このビットがURビットとともにセットされると、補助レジスタは自動的に更新されます。

0：レジスタの更新は無効です。

1：レジスタの更新は有効です。

44.15.49 DSIホストLTDC現在VCIDレジスタ（DSI_LCVCIDR）

アドレスオフセット：0x010C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCID[1:0]	
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **VCID[1:0]**：仮想チャネルID

このフィールドは、LTDCインタフェースの仮想チャネルIDを返します。

44.15.50 DSIホストLTDC現在カラー･コーディング･レジスタ（DSI_LCCCR）

アドレスオフセット：0x0110

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPE	Res.	Res.	Res.	Res.	COLC[3:0]			
							r					r	r	r	r

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **LPE**：軽度パック済みイネーブル

このビットは、18bit設定に対する軽度パック済みバリアントの現在の状態を返します。

0：軽度パック済みバリアントは無効です。

1：軽度パック済みバリアントは有効です。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **COLC[3:0]**：カラー･コーディング

このフィールドは、現在のLTDCインタフェースのカラー･コーディングを返します。

0000：16bit設定1

0001：16bit設定2

0010：16bit設定3

0011：18bit設定1

0100：18bit設定2

0101：24bit

0110～1111：予約済み

コマンド･モードのLTDCインタフェースが選択され、現在、コマンド･モード（CMDM=1）で動作している場合は、0110～1111：24bit

44.15.51 DSIホスト低電力モード現在設定レジスタ（DSI_LPMCCR）

アドレスオフセット：0x0118

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPSIZE[7:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLPSIZE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **LPSIZE[7:0]**：最大パケット･サイズ

このフィールドは、低電力モードでのコマンド送信のために、VSA、VBP、およびVFP領域の期間内で1ラインに収まる最大のパケットの現在のサイズをバイト単位で返します。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **VLPSIZE[7:0]**：VACT最大パケット･サイズ

このフィールドは、低電力モードでのコマンド送信のために、VACT領域の期間内で1ラインに収まる最大のパケットの現在のサイズをバイト単位で返します。

44.15.52 DSIホスト･ビデオ･モード現在設定レジスタ（DSI_VMCCR）

アドレスオフセット：0x0138

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LPCE	FBTAAE	LPHFE	LPHBPE	LPVAE	LPVFPE	LPVBPE	LPVSAE	VMT[1:0]	
						r	r	r	r	r	r	r	r	r	r

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **LPCE**：低電力コマンドの有効化

このビットは、低電力モードでの現在のコマンド送信状態を返します。

0：低電力モードでのコマンド送信は無効です。

1：低電力モードでのコマンド送信は有効です。

ビット8 **FBTAAE**：フレームBTA確認応答の有効化

このビットは、フレームの終わりでの確認応答リクエストの現在の状態を返します。

0：フレームの終わりでの確認応答は無効です。

1：フレームの終わりでの確認応答は有効です。

ビット7 **LPHFE**：低電力水平フロント･ポーチの有効化

このビットは、タイミングが許す場合の水平フロント･ポーチ（HFP）期間内での低電力への復帰の現在の状態を返します。

0：HFP期間内での低電力への復帰は無効です。

1：HFP期間内での低電力への復帰は有効です。

ビット6 **LPHBPE**：低電力水平バック･ポーチの有効化

このビットは、タイミングが許す場合の水平バック･ポーチ（HBP）期間内での低電力への復帰の現在の状態を返します。

0：HBP期間内での低電力への復帰は無効です。

1：HBP期間内での低電力への復帰は有効です。

ビット5 **LPVAE**：低電力垂直アクティブ有効化

このビットは、タイミングが許す場合の垂直アクティブ（VACT）期間内での低電力への復帰の現在の状態を返します。

0：VACT内での低電力への復帰は無効です。

1：VACT内での低電力への復帰は有効です。

ビット4 **LPVFPE**：低電力垂直フロント･ポーチ有効化

このビットは、タイミングが許す場合の垂直フロント･ポーチ（VFP）期間内での低電力への復帰の現在の状態を返します。

0：VFP内での低電力への復帰は無効です。

1：VFP内での低電力への復帰は有効です。

ビット3 **LPVBPE**：低電力垂直バック･ポーチ有効化

このビットは、タイミングが許す場合の垂直バック･ポーチ（VBP）期間内での低電力への復帰の現在の状態を返します。

0：VBP内での低電力への復帰は無効です。

1：VBP内での低電力への復帰は有効です。

ビット2 **LPVSAE**：低電力垂直同期時間有効化

このビットは、タイミングが許す場合の垂直同期時間（VSA）期間内での低電力への復帰の現在の状態を返します。

0：VSA内での低電力への復帰は無効です。

1：VSA内での低電力への復帰は有効です

ビット1:0 **VMT[1:0]**：ビデオ･モード･タイプ

このフィールドは、現在のビデオ･モード送信タイプを返します。

00：同期パルスでの非バースト

01：同期イベントでの非バースト

1x：バースト･モード

44.15.53 DSIホスト･ビデオ･パケット現在設定レジスタ（DSI_VPCCR）

アドレスオフセット：0x013C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	VPSIZE[13:0]													
		r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **VPSIZE[13:0]**：ビデオ･パケット･サイズ

このフィールドは、単一のビデオ･パケットのピクセル数を返します。

44.15.54 DSIホスト･ビデオ･チャンク現在設定レジスタ（DSI_VCCCR）

アドレスオフセット：0x0140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	NUMC[12:0]												
			r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **NUMC[12:0]**：チャンクの数

このフィールドは、1ライン期間中に送信されるチャンクの数を返します。

44.15.55 DSIホスト･ビデオnullパケット現在設定レジスタ（DSI_VNPCCR）

アドレスオフセット：0x0144

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	NPSIZE[12:0]												
			r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **NPSIZE[12:0]**：nullパケットのサイズ

このフィールドは、nullパケットのバイト数を返します。

44.15.56 DSIホスト･ビデオHSA現在設定レジスタ（DSI_VHSACCR）

アドレスオフセット：0x0148

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HSA[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **HSA[11:0]**：水平同期アクティブ時間

このフィールドは、水平同期アクティブ期間をレーン･バイト･クロックのサイクルで返します。

44.15.57 DSIホスト･ビデオHBP現在設定レジスタ（DSI_VHBPCCR）

アドレスオフセット：0x014C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HBP[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **HBP[11:0]**：水平バック･ポーチ時間

このフィールドは、水平バック･ポーチ期間をレーン･バイト･クロックのサイクルで返します。

44.15.58 DSIホスト･ビデオ･ライン現在設定レジスタ（DSI_VLCCR）

アドレスオフセット：0x0150

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	HLINE[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **HLINE[14:0]**：水平ライン時間

このフィールドは、レーン･バイト･クロックのサイクルでカウントされる合計の水平ライン期間（HSA+HBP+HACT+HFP）の現在値を返します。

44.15.59 DSIホスト･ビデオVSA現在設定レジスタ（DSI_VVSACCR）

アドレスオフセット：0x0154

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VSA[9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VSA[9:0]**：垂直同期アクティブ時間

このフィールドは、水平ラインの数で計測される現在の垂直同期アクティブ期間を返します。

44.15.60 DSIホスト･ビデオVBP現在設定レジスタ（DSI_VVBPCCR）

アドレスオフセット：0x0158

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBP[9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VBP[9:0]**：垂直バック･ポーチ時間

このフィールドは、水平ラインの数で計測される現在の垂直バック･ポーチ期間を返します。

44.15.61 DSIホスト･ビデオVFP現在設定レジスタ（DSI_VVFPCCR）

アドレスオフセット：0x015C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VFP[9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **VFP[9:0]**：垂直フロント･ポーチ時間

このフィールドは、水平ラインの数で計測される現在の垂直フロント･ポーチ期間を返します。

44.15.62 DSIホスト･ビデオVA現在設定レジスタ（DSI_VVACCR）

アドレスオフセット：0x0160

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	VA[13:0]													
		r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **VA[13:0]**：垂直アクティブ時間

このフィールドは、水平ラインの数で計測される現在の垂直アクティブ期間を返します。

44.15.63 DSIホストFIFOおよびバッファ･ステータスレジスタ（DSI_FBSR）

アドレスオフセット：0x0168

リセット値：0x0005 0015

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBF	APBE	ACBF	ACBE	Res.	Res.	VPBF	VPBE
								r	r	r	r			r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APWFF	APWFE	ACWFF	ACWFE	VPWFF	VPWFE	VCWFF	VCWFE
								r	r	r	r	r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **APBF**：アダプテッド･コマンド･モードのペイロード･バッファ･フル

このビットは、アダプテッド･コマンド･モードのペイロード内部バッファのフル状態を示します。

0：ペイロード内部バッファはフルではありません。

1：ペイロード内部バッファはフルです。

ビット22 **APBE**：アダプテッド･コマンド･モードのペイロード･バッファ･エンプティ
このビットは、アダプテッド･コマンド･モードのペイロード内部バッファのエンプティ状態を示します。

0：ペイロード内部バッファはエンプティではありません。
1：ペイロード内部バッファはエンプティです。

ビット21 **ACBF**：アダプテッド･コマンド･モードのコマンド･バッファ･フル
このビットは、アダプテッド･コマンド･モードのコマンド内部バッファのフル状態を示します。

0：コマンド内部バッファはフルではありません。
1：コマンド内部バッファはフルです。

ビット20 **ACBE**：アダプテッド･コマンド･モードのコマンド･バッファ･エンプティ
このビットは、アダプテッド･コマンド･モードのコマンド内部バッファのエンプティ状態を示します。

0：コマンド内部バッファはエンプティではありません。
1：コマンド内部バッファはエンプティです。

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **VPBF**：ビデオ･モードのペイロード･バッファ･フル
このビットは、ビデオ･モードのペイロード内部バッファのフル状態を示します。
0：ペイロード内部バッファはフルではありません。
1：ペイロード内部バッファはフルです。

ビット16 **VPBE**：ビデオ･モードのペイロード･バッファ･エンプティ
このビットは、ビデオ･モードのペイロード内部バッファのエンプティ状態を示します。
0：ペイロード内部バッファはエンプティではありません。
1：ペイロード内部バッファはエンプティです。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **APWFF**：アダプテッド･コマンド･モードのペイロード書込みFIFOフル
このビットは、アダプテッド･コマンド･モードの書込みペイロードFIFOのフル状態を示します。
0：書込みペイロードFIFOはフルではありません。
1：書込みペイロードFIFOはフルです。

ビット6 **APWFE**：アダプテッド･コマンド･モードのペイロード書込みFIFOエンプティ
このビットは、アダプテッド･コマンド･モードの書込みペイロードFIFOのエンプティ状態を示します。
0：書込みペイロードFIFOはエンプティではありません。
1：書込みペイロードFIFOはエンプティです。

ビット5 **ACWFF**：アダプテッド･コマンド･モードのコマンド書込みFIFOフル
このビットは、アダプテッド･コマンド･モードの書込みコマンドFIFOのフル状態を示します。
0：書込みコマンドFIFOはフルではありません。
1：書込みコマンドFIFOはフルです。

ビット4 **ACWFE**：アダプテッド･コマンド･モードのコマンド書込みFIFOエンプティ
このビットは、アダプテッド･コマンド･モードの書込みコマンドFIFOのエンプティ状態を示します。
0：書込みコマンドFIFOはエンプティではありません。
1：書込みコマンドFIFOはエンプティです。

ビット3 **VPWFF**：ビデオ･モードのペイロード書込みFIFOフル
このビットは、ビデオ･モードの書込みペイロードFIFOのフル状態を示します。
0：書込みペイロードFIFOはフルではありません。
1：書込みペイロードFIFOはフルです。

ビット2 **VPWFE**：ビデオ･モードのペイロード書込みFIFOエンプティ

このビットは、ビデオ･モードの書込みペイロードFIFOのエンプティ状態を示します。

0：書込みペイロードFIFOはエンプティではありません。

1：書込みペイロードFIFOはエンプティです。

ビット1 **VCWFF**：ビデオ･モードのコマンド書込みFIFOフル

このビットは、ビデオ･モードの書込みコマンドFIFOのフル状態を示します。

0：書込みコマンドFIFOはフルではありません。

1：書込みコマンドFIFOはフルです。

ビット0 **VCWFE**：ビデオ･モードのコマンド書込みFIFOエンプティ

このビットは、ビデオ･モードの書込みコマンドFIFOのエンプティ状態を示します。

0：書込みコマンドFIFOはエンプティではありません。

1：書込みコマンドFIFOはエンプティです。

44.16 DSIラッパー･レジスタ

44.16.1 DSIラッパー設定レジスタ（DSI_WCFGR）

アドレスオフセット：0x0400

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VSPOL	AR	TEPOL	TESRC	COLMUX[2:0]			DSIM
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **VSPOL**：VSync極性

このビットによってLTDCが停止するVSyncエッジを選択します。

0：LTDCは立ち下がりエッジで停止します。

1：LTDCは立ち上がりエッジで停止します。

このビットは、DSIが停止しているとき（DSI_WCR.DSIEN = 0およびDSI_CR.EN = 0）にのみ変更する必要があります。

ビット6 **AR**：自動リフレッシュ

このビットでは、DBIモードのリフレッシュ･モードを選択します。

0：自動リフレッシュ･モードは無効です。

1：自動リフレッシュ･モードは有効です。

このビットは、DSIホストが停止しているとき（DSI_CR.EN = 0）にのみ変更する必要があります。

ビット5 **TEPOL**：TE極性

このビットでは、外部ピンのティアリング効果（TE）ソースの極性を選択します。

0：立ち上がりエッジ。

1：立ち下がりエッジ。

このビットは、DSIホストが停止しているとき（DSI_CR.EN = 0）にのみ変更する必要があります。

ビット4 **TESRC**：TEソース

このビットでは、ティアリング効果（TE）のソースを選択します。

0：DSIリンク

1：外部ピン

このビットは、DSIホストが停止しているとき（DSI_CR.EN = 0）にのみ変更する必要があります。

ビット3:1 **COLMUX[2:0]**：カラー多重化

このビットでは、DSIホストで使用されるカラー多重化を選択します。

000：16bit設定1

001：16bit設定2

010：16bit設定3

011：18bit設定1

100：18bit設定2

101：24bit

このフィールドは、DSIが停止しているとき（DSI_WCR.DSIEN = 0およびDSI_CR.EN = 0）にのみ変更する必要があります。

ビット0 **DSIM**：DSIモード

このビットでは、ビデオ送信のモードを選択します。

0：ビデオ･モード

1：アダプテッド･コマンド･モード

このビットは、DSIホストが停止しているとき（DSI_CR.EN = 0）にのみ変更する必要があります。

44.16.2 DSIラッパー制御レジスタ（DSI_WCR）

アドレスオフセット：0x0404

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSIEN	LTDCEN	SHTDN	COLM
												rw	rs	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **DSIEN**：DSIイネーブル

このビットは、DSIラッパーを有効にします。

0：DSIは無効です。

1：DSIは有効です。

ビット2 **LTDCEN**：LTDCイネーブル

このビットは、アダプテッド･コマンド･モードでのフレーム転送のためにLTDCを有効にします。

0：LTDCは無効です。

1：LTDCは有効です。

ビット1 **SHTDN**：シャットダウン

このビットは、ビデオ･モードでのディスプレイのシャットダウンを制御します。

0：ディスプレイはオンです。

1：ディスプレイはオフです。

ビット0 **COLM**：カラーモード

このビットは、ビデオ･モードでのディスプレイのカラーモードを制御します。

0：フル･カラーモード

1：8カラーモード

44.16.3 DSIラッパー割込みイネーブル･レジスタ（DSI_WIER）

アドレスオフセット：0x0408

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PLLUIE	PLLLIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERIE	TEIE
					rw	rw								rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **PLLUIE**：PLLアンロック割込みイネーブル
このビットは、PLLアンロック割込みを有効にします。
0：PLLアンロック割込みは無効です。
1：PLLアンロック割込みは有効です。

ビット9 **PLLLIE**：PLLロック割込みイネーブル
このビットは、PLLロック割込みを有効にします。
0：PLLロック割込みは無効です。
1：PLLロック割込みは有効です。

ビット8:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **ERIE**：リフレッシュ終了割込みイネーブル
このビットは、リフレッシュ終了割込みを有効にします。
0：リフレッシュ終了割込みは無効です。
1：リフレッシュ終了割込みは有効です。

ビット0 **TEIE**：ティアリング効果割込みイネーブル
このビットは、ティアリング効果割込みを有効にします。
0：ティアリング効果割込みは無効です。
1：ティアリング効果割込みは有効です。

44.16.4 DSIラッパー割込みおよびステータスレジスタ（DSI_WISR）

アドレスオフセット：0x040C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PLLUIF	PLLLIF	PLLLS	Res.	Res.	Res.	Res.	Res.	BUSY	ERIF	TEIF
					r	r	r						r	r	r

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **PLLUIF**：PLLアンロック割込みフラグ
このビットは、PLLがロック解除されたときにセットされます。
0：PLLアンロックイベントは発生していません。
1：PLLアンロックイベントが発生しました。

ビット9 **PLLLIF**：PLLロック割込みフラグ
このビットは、PLLがロックされたときにセットされます。
0：PLLロックイベントは発生していません。
1：PLLロックイベントが発生しました。

ビット8 **PLLLS**：PLLロックステータス
このビットはPLLがロックされるとセットされ、ロック解除されるとクリアされます。
0：PLLはロック解除されています。
1：PLLはロックされています。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **BUSY**：ビジーフラグ
このビットは、アダプテッド･コマンド･モードでのフレーム転送が進行中のときにセットされます。
0：転送は進行中ではありません。
1：転送は進行中です。

ビット1 **ERIF**：リフレッシュ終了割込みフラグ
このビットは、アダプテッド･コマンド･モードでのフレーム転送が完了したときにセットされます。
0：リフレッシュ終了イベントは発生していません。
1：リフレッシュ終了イベントが発生しました。

ビット0 **TEIF**：ティアリング効果割込みフラグ
このビットは、ティアリング効果イベントが発生したときに、セットされます。
0：ティアリング効果イベントは発生していません。
1：ティアリング効果イベントが発生しました。

44.16.5 DSIラッパー割込みフラグクリアレジスタ（DSI_WIFCR）

アドレスオフセット：0x0410

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CPLLUIF	CPLLLIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERIF	CTEIF
					w	w								w	w

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **CPLLUIF**：PLLアンロック割込みフラグのクリア
1を書き込むと、DSI_WSRレジスタのPLLUIFフラグがクリアされます。

ビット9 **CPLLLIF**：PLLロック割込みフラグのクリア
1を書き込むと、DSI_WSRレジスタのPLLLIFフラグがクリアされます。

ビット8:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CERIF**：リフレッシュ終了割込みフラグのクリア
1を書き込むと、DSI_WSRレジスタのERIFフラグがクリアされます。

ビット0 **CTEIF**：ティアリング効果割込みフラグのクリア
1を書き込むと、DSI_WSRレジスタのTEIFフラグがクリアされます。

44.16.6 DSIラッパー PHY設定レジスタ0（DSI_WPCR0）

アドレスオフセット：0x0418

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FTXSMDL	FTXSMCL	Res.	Res.	Res.	SWDL1	SWDL0	SWCL	Res.	Res.	Res.	Res.	Res.	Res.
		rw	rw				rw	rw	rw						

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **FTXSMDL**：データ･レーンをTX STOPモードに強制

このビットは、データ･レーンを強制的にTX STOPモードにします。これはレーン･モジュールを送信モードに初期化するために使用されます。これによって、レーン･モジュールは直ちに送信制御モードにジャンプし、停止状態（LP-11）の送信を開始します。これは、間違ったBTAシーケンスの後にTXモードに戻るために使用できます。

0：効果なし。

1：データ･レーンを強制的にTX STOPモードにします。

ビット12 **FTXSMCL**：クロック･レーンをTX STOPモードに強制

このビットは、クロック･レーンを強制的にTX STOPモードにします。これはレーン･モジュールを送信モードに初期化するために使用されます。これによって、レーン･モジュールは直ちに送信制御モードにジャンプし、停止状態（LP-11）の送信を開始します。これは、間違ったBTAシーケンスの後にTXモードに戻るために使用できます。

0：効果なし。

1：クロック･レーンを強制的にTX STOPモードにします。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SWDL1**：データ･レーン1のピンのスワップ

このビットによってクロック･レーンのピンがスワップされます。

0：クロック･レーンの通常のピン設定

1：スワップされたクロック･レーンのピン

ビット7 **SWDL0**：データ･レーン0のピンのスワップ

このビットによってデータ･レーン0のピンがスワップされます。

0：クロック･レーンの通常のピン設定

1：スワップされたクロック･レーンのピン

ビット6 **SWCL**：クロック･レーンのピンのスワップ

このビットによってクロック･レーンのピンがスワップされます。

0：クロック･レーンの通常のピン設定

1：スワップされたクロック･レーンのピン

ビット5:0 予約済みであり、リセット値に保持する必要があります。

44.16.7 DSIラッパー・レギュレータおよびPLL制御レジスタ（DSI_WRPCR）

アドレスオフセット：0x0430

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	BC[1:0]		ODF[8:0]									IDF[8:5]			
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDF[4:0]					NDIV[8:0]									Res.	PLLEN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **BC[1:0]**：帯域制御

このフィールドでは、VCO周波数帯域を選択します。

00：500から800MHz

01：800から1000MHz

その他：予約済み

ビット28:20 **ODF[8:0]**：PLL出力の分周比

このフィールドでは、PLL出力分周比を設定します。

0：PLL出力は1で分周されます。

1：PLL出力は1で分周されます。

2：PLL出力は2で分周されます。

...

511：PLL出力は511で分周されます。

ビット19:11 **IDF[8:0]**：PLL入力の分周比

このフィールドでは、PLL入力の分周比を設定します。

0：PLL入力は1で分周されます。

1：PLL入力は1で分周されます。

2：PLL入力は2で分周されます。

...

511：PLL入力は511で分周されます。

ビット10:2 **NDIV[8:0]**：PLLループの分周比

このフィールドでは、PLLループの分周比を設定します。

0：PLLループは1x2で分周されます。

1：PLLループは1x2で分周されます。

2：PLLループは2x2で分周されます。

...

511：PLLループは511x2で分周されます。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **PLLEN**：PLLイネーブル

このビットは、D-PHYのPLLを有効にします。

0：PLLは無効です。

1：PLLは有効です。

44.16.8 DSIラッパー PLLチューニング･レジスタ （DSI_WPTR）

アドレスオフセット：0x0434

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPF[3:0]				CP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw								

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **LPF[3:0]**：ループフィルタ

このフィールドは、PLLループフィルタを制御するもので、PFD周波数の範囲に従って設定する必要があります。

0000：2.0から4.4MHz

0001：4.4から30.9MHz

0010：30.9から50MHz

その他：予約済み

ビット11:8 **CP[3:0]**：チャージポンプ

このフィールドは、PLLチャージポンプを制御するもので、PFD周波数範囲およびLPFの値に従って設定する必要があります。

0000：2.0～4.4MHzおよび14.1～30.9MHz

0001：4.4から14.1MHz

0010：45.7から50MHz

0011：30.9から45.7MHz

その他：予約済み

ビット7:0 予約済みであり、リセット値に保持する必要があります。

44.17 DSIバイアス･レジスタ

44.17.1 DSIバイアス設定レジスタ（DSI_BCFGR）

アドレスオフセット：0x0808

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PWRUP	Res.	Res.	Res.	Res.	Res.	Res.
									rw						

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **PWRUP**：パワーアップ

このビットは、MIPI D-PHYの基準バイアスの電源を投入します。

0：基準バイアスはパワーダウン状態です。

1：基準バイアスはパワーアップ状態です。

ビット5:0 予約済みであり、リセット値に保持する必要があります。

44.18 D-PHYレジスタ

44.18.1 DSI D-PHYクロック帯域制御レジスタ（DSI_DPCBCR）

アドレスオフセット：0x0C04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BC[4:0]					Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:3 **BC[4:0]**：帯域制御

このフィールドでは、D-PHYで使用される周波数帯域を選択します。

00000：80から100MHz

00001：100から120MHz

00010：120から160MHz

00011：160から200MHz

00100：200から240MHz

00101：240から320MHz

00110：320から390MHz

00111：390から450MHz

01000：450から510MHz

その他：予約済み

ビット2:0 予約済みであり、リセット値に保持する必要があります。

44.18.2 DSI D-PHYクロック･スルーレート制御レジスタ（DSI_DPCSRCR）

アドレスオフセット：0x0C34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRC[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SRC[7:0]**：スルーレート制御

このフィールドでは、HS-TX速度のスルーレートを選択します。

0x0E：80～750Mbit/s

その他：予約済み

44.18.3 DSI D-PHYデータ･レーン0 HSオフセット制御レジスタ（DSI_DPDL0HSOCR）

アドレスオフセット：0x0C5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPRPO[3:0]				Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **HSPRPO[3:0]**：HS準備オフセット

このフィールドでは、HS準備タイミングに追加されるレーン･バイト･クロックでのオフセットを選択します。オフセットは、D-PHY用に選択された周波数帯域に依存します。

0000：100～120MHz - 120～160MHz - 240～320MHz

0001：80～100MHz - 160～200MHz - 200～240MHz - 320～390MHz

0010：390～450MHz - 450～510MHz

その他：予約済み

ビット3:0 予約済みであり、リセット値に保持する必要があります。

44.18.4 DSI D-PHYデータ･レーン0 HS LPXオフセット制御レジスタ（DSI_DPDL0LPXOCR）

アドレスオフセット：0x0C60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPXO[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **LPXO[3:0]**：LPXオフセット

このフィールドでは、次の状態に関連する遅延を微調整するために追加されるオフセットを選択します：INIT_STATE、STOP_STATE、LP01_STATE、LP11_STATE。

このフィールドは、2の補数形式の4bitの符号付き値（-8 ～ +7の範囲）です。

LPXタイミングは、D-PHYで選択された周波数帯域に依存する符号なし固定7bit値と、このフィールドの符号付き4bit値で構成されます。

LPXタイミングはレーン･バイト･クロック周期で表されます。

LPX固定値は次のとおりです。

80から120MHz: 7’h01

120から160MHz: 7’h02

160から320MHz: 7’h03

320から450MHz: 7’h04

450から510MHz: 7’h05

結果として得られるLPXタイミングは符号なし7bit値であるため、値が負（2の補数形式）の場合のアンダーフローに注意する必要があります。

44.18.5 DSI D-PHYデータ・レーン0帯域制御レジスタ（DSI_DPDL0BCR）

アドレスオフセット：0x0C70

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BC[4:0]				
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **BC[4:0]**：帯域制御

このフィールドでは、D-PHYで使用される周波数帯域を選択します。

00000：80から100MHz

00001：100から120MHz

00010：120から160MHz

00011：160から200MHz

00100：200から240MHz

00101：240から320MHz

00110：320から390MHz

00111：390から450MHz

01000：450から510MHz

その他：予約済み

44.18.6 DSI D-PHYデータ・レーン0 スルーレート制御レジスタ（DSI_DPDL0SRCR）

アドレスオフセット：0x0CA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRC[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SRC[7:0]**：スルーレート制御

このフィールドでは、HS-TX速度のスルーレートを選択します。

0x0E：80～750Mbit/s

その他：予約済み

44.18.7 DSI D-PHYデータ･レーン1 HSオフセット制御レジスタ（DSI_DPDL1HSOCR）

アドレスオフセット：0x0CF4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPRPO[3:0]				Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **HSPRPO[3:0]**：HS準備オフセット

このフィールドでは、HS準備タイミングに追加されるレーン･バイト･クロックでのオフセットを選択します。オフセットは、D-PHY用に選択された周波数帯域に依存します。

0000：100～120MHz - 120～160MHz - 240～320MHz

0001：80～100MHz - 160～200MHz - 200～240MHz - 320～390MHz

0010：390～450MHz - 450～510MHz

その他：予約済み

ビット3:0 予約済みであり、リセット値に保持する必要があります。

44.18.8 DSI D-PHYデータ･レーン1 HS LPXオフセット制御レジスタ（DSI_DPDL1LPXOCR）

アドレスオフセット：0x0CF8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPXO[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **LPXO[3:0]**：LPXオフセット

このフィールドでは、次の状態に関連する遅延を微調整するために追加されるオフセットを選択します：INIT_STATE、STOP_STATE、LP01_STATE、LP11_STATE。

このフィールドは、2の補数形式の4bitの符号付き値です。

LPXタイミングは、D-PHYで選択された周波数帯域に依存する符号なし固定7bit値と、このフィールドの符号付き4bit値で構成されます。

オフセットはレーン･バイト･クロック周期で表されます。

結果として得られるLPXタイミングは符号なし7bit値であるため、値が負（2の補数形式）の場合のアンダーフローに注意する必要があります。

44.18.9 DSI D-PHYデータ・レーン1帯域制御レジスタ（DSI_DPDL1BCR）

アドレスオフセット：0x0D08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BC[4:0]				
											rw	rw	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **BC[4:0]**：帯域制御

このフィールドでは、D-PHYで使用される周波数帯域を選択します。

00000：80から100MHz

00001：100から120MHz

00010：120から160MHz

00011：160から200MHz

00100：200から240MHz

00101：240から320MHz

00110：320から390MHz

00111：390から450MHz

01000：450から510MHz

その他：予約済み

44.18.10 DSI D-PHYデータ・レーン1スルーレート制御レジスタ（DSI_DPDL1SRCR）

アドレスオフセット：0x0D38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRC[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SRC[7:0]**：スルーレート制御

このフィールドでは、HS-TX速度のスルーレートを選択します。

0x0E：80～750Mbit/s

その他：予約済み

44.18.11 DSIレジスタマップ

表444.DSIレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DSI_VR	VERSION[31:0]																															
	リセット値	0	0	1	1	0	0	0	1	0	0	1	1	0	1	0	0	0	0	1	1	0	0	0	1	0	0	1	0	1	1	0	0
0x0004	DSI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																																0
0x0008	DSI_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOCKDIV[7:0]								TXECKDIV[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	DSI_LVCIDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCID[1:0]	
	リセット値																															0	0
0x0010	DSI_LCOLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPE	Res.	Res.	Res.	Res.	COLC[3:0]			
	リセット値																								0					0	0	0	0
0x0014	DSI_LPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSP	VSP	DEP
	リセット値																														0	0	0
0x0018	DSI_LPMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPSIZE[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLPSIZE[7:0]							
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x001C ~0x0028	予約済み	予約済み																															
0x002C	DSI_PCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETTXLPE	CRCRXE	ECCRXE	BTAE	ETRXE	ETTXE
	リセット値																											0	0	0	0	0	0
0x0030	DSI_GVCIDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCIDTX[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCIDRX[1:0]	
	リセット値															0	0															0	0
0x0034	DSI_MCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDM
	リセット値																																1
0x0038	DSI_VMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PGO	Res.	Res.	Res.	PGM	Res.	Res.	Res.	PGE	LPCE	FBTAAE	LPHFPE	LPHBPE	LVAE	LPVFPE	LPVBPE	LPVSAE	Res.	Res.	Res.	Res.	Res.	Res.	VMT[1:0]	
	リセット値								0				0				0	0	0	0	0	0	0	0	0							0	0
0x003C	DSI_VPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VPSIZE[13:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	DSI_VCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMC[12:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	DSI_VNPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NPSIZE[12:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	DSI_VHSACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSA[11:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	DSI_VHBPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HBP[11:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0

表444.DSIレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0050	**DSI_VLCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HLINE[14:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	**DSI_VVSACR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VSA[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0058	**DSI_VVBPCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBP[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x005C	**DSI_VVFPCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VFP[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0060	**DSI_VVACR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VA[13:0]													
	リセット値																			0	0	0		0	0	0	0	0	0	0	0	0	0
0x0064	**DSI_LCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDSIZE[15:0]															
	リセット値																	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0
0x0068	**DSI_CMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MRDPS	Res.	Res.	Res.	Res.	DLWTX	DSR0TX	DSW1TX	DSW0TX	Res.	GLWTX	GSR2TX	GSR1TX	GSR0TX	GSW2TX	GSW1TX	GSW0TX	Res.	Res.	Res.	Res.	Res.	Res.	ARE	TEARE
	リセット値								0					0	0	0	0		0	0	0	0	0	0	0							0	0
0x006C	**DSI_GHCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WCMSB[7:0]								WCLSB[7:0]								VCID		DT[5:0]					
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0070	**DSI_GPDR**	DATA4 [7:0]								DATA3 [7:0]								DATA2 [7:0]								DATA1 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0074	**DSI_GPSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBF	PBE	CMDBF	CMDBEB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCB	PRDFF	PRDFE	PWRFF	PWRFE	CMDFF	CMDFE
	リセット値													0	1	0	1										0	0	1	0	1	0	1
0x0078	**DSI_TCCR0**	HSTX_TOCNT[15:0]																LPRX_TOCNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x007C	**DSI_TCCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSRD_TOCNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0080	**DSI_TCCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPRD_TOCNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	**DSI_TCCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSWR_TOCNT[15:0]															
	リセット値								0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0088	**DSI_TCCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSWR_TOCNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008C	**DSI_TCCR5**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BTA_TOCNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0090	予約済み	予約済み																															
0x0094	**DSI_CLCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACR	DPCC
	リセット値																															0	0
0x0098	**DSI_CLTCR**	Res.	Res.	Res.	Res.	Res.	Res.	HS2LP_TIME [9:0]										Res.	Res.	Res.	Res.	Res.	Res.	LP2HS_TIME [9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x009C	**DSI_DLTCR**	Res.	Res.	Res.	Res.	Res.	Res.	HS2LP_TIME [9:0]										Res.	Res.	Res.	Res.	Res.	Res.	LP2HS_TIME [9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0

表444.DSIレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00A0	DSI_PCTLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKE	DEN	Res.
	リセット値																														0	0	
0x00A4	DSI_PCCONFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SW_TIME[7:0]												Res.	Res.	NL[1:0]	
	リセット値																	0	0	0	0	0	0	0	0							0	0
0x00A8	DSI_PUCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UEDL	URDL	UECL	URCL
	リセット値																													0	0	0	0
0x00AC	DSI_PTTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TX_TRIG[3:0]			
	リセット値																													0	0	0	0
0x00B0	DSI_PSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UAN1	PSS1	RUE0	UAN0	PSS0	UANC	PSSC	PD	Res.
	リセット値																								1	0	0	1	0	1	0	0	
0x00B4 ~0x00B8	予約済み	予約済み																															
0x00BC	DSI_ISR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PE4	PE3	PE2	PE1	PE0	AE15	AE14	AE13	AE12	AE11	AE10	AE9	AE8	AE7	AE6	AE5	AE4	AE3	AE2	AE1	AE0
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C0	DSI_ISR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBUE	Res.	Res.	Res.	Res.	Res.	Res.	GPRXE	GPRDE	GPTXE	GPWRE	GCWRE	LPWRE	EOTPE	PSE	CRCE	ECCME	ECCSE	TOLPRX	TOHSTX
	リセット値													0							0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C4	DSI_IER0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PE4IE	PE3IE	PE2IE	PE1IE	PE0IE	AE15IE	AE14IE	AE13IE	AE12IE	AE11IE	AE10IE	AE9IE	AE8IE	AE7IE	AE6IE	AE5IE	AE4IE	AE3IE	AE2IE	AE1IE	AE0IE
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8	DSI_IER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PBUEIE	Res.	Res.	Res.	Res.	Res.	Res.	GPRXEIE	GPRDEIE	GPTXEIE	GPWREIE	GCWREIE	LPWREIE	EOTPEIE	PSEIE	CRCEIE	ECCMEIE	ECCSEIE	TOLPRXIE	TOHSTXIE
	リセット値													0							0	0	0	0	0	0	0	0	0	0	0	0	0
0x00CC ~0x00D4	予約済み	予約済み																															
0x00D8	DSI_FIR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPE4	FPE3	FPE2	FPE1	FPE0	FAE15	FAE14	FAE13	FAE12	FAE11	FAE10	FAE9	FAE8	FAE7	FAE6	FAE5	FAE4	FAE3	FAE2	FAE1	FAE0
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00DC	DSI_FIR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FPBUE	Res.	Res.	Res.	Res.	Res.	Res.	FGPRXE	FGPRDE	FGPTXE	FGPWRE	FGCWRE	FLPWRE	FEOTPE	FPSE	FCRCE	FECCME	FECCSE	FTOLPRX	FTOHSTX
	リセット値													0							0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F0	予約済み	予約済み																															
0x00F4	DSI_DLTRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MRD_TIME[14:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F8 ~0x00FC	予約済み	予約済み																															
0x0100	DSI_VSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																								0								0
0x0104 ~0x0108	予約済み	予約済み																															
0x010C	DSI_LCVCIDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VCID[1:0]	
	リセット値																															0	0

表444.DSIレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0110	**DSI_LCCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPE	Res.	Res.	Res.	Res.	COLC[3:0]			
	リセット値																								0					0	0	0	0
0x0114	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0118	**DSI_LPMCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPSIZE[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLPSIZE[7:0]							
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x011C ~0x0134	予約済み	予約済み																															
0x0138	**DSI_VMCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPCE	FBTAAE	LPHFE	LPHBPE	LVAE	LPVFPE	LPVBPE	LPVSAE	VMT[1:0]	
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x013C	**DSI_VPCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VPSIZE[13:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0140	**DSI_VCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMC[12:0]														
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0144	**DSI_VNPCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NPSIZE[12:0]														
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0148	**DSI_VHSACCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSA[11:0]													
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x014C	**DSI_VHBPCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HBP[11:0]													
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x0150	**DSI_VLCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HLINE[14:0]														
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0154	**DSI_VVSACCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VSA[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0158	**DSI_VVBPCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBP[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x015C	**DSI_VVFPCCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VFP[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0160	**DSI_VVACCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VA[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0164	予約済みです。	予約済み																															
0x0168	**DSI_FBSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBF	APBE	ACBF	ACBE	Res.	Res.	VPBF	VPBE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APWFF	APWFE	ACWFF	ACWFE	VPWFF	VPWFE	VCWFF	VCWFE
	リセット値									0	0	0	0			0	1									0	0	0	1	0	1	0	1
0x0170 ~0x03FC	予約済み	予約済み																															
0x0400	**DSI_WCFGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VSPOL	AR	TEPOL	TESRC	COLMUX[2:0]			DSIM
	リセット値																									0	0	0	0	0	0	0	0

表444.DSIレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0404	**DSI_WCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSIEN	LTDCEN	SHTDN	COLM
	リセット値																													0	0	0	0
0x0408	**DSI_WIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLLUIE	PLLLIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERIE	TEIE
	リセット値																						0	0								0	0
0x040C	**DSI_WISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLLUIF	PLLLIF	PLLLS	Res.	Res.	Res.	Res.	Res.	BUSY	ERIF	TEIF
	リセット値																						0	0	0						0	0	0
0x0410	**DSI_WIFCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPLLUIF	CPLLLIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERIF	CTEIF
	リセット値																						0	0								0	0
0x0414	予約済み	予約済み																															
0x0418	**DSI_WPCR0**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDDL	Res.	CDOFFDL	FTXSMDL	FTXSMCL	Res.	Res.	Res.	SWDL1	SWDL0	SWCL	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																0		0	0	0				0	0	0						
0x0420 ～ 0x042C	予約済み	予約済み																															
0x0430	**DSI_WRPCR**	Res.	BC[1:0]		ODF[8:0]									IDF[8:0]									NDIV[8:0]								Res.	PLLEN
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x0434	**DSI_WPTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPF[3:0]				CP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0								
0x0434 ~0x0804	予約済み	予約済み																															
0x0808	**DSI_BCFGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PWRUP	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																										0						
0x080C ~0x0C00	予約済み	予約済み																															
0x0C04	**DSI_DPCBCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BC[4:0]					Res.	Res.	Res.
	リセット値																									0	0	0	0	0			
0x0C08 ~0x0C30	予約済み	予約済み																															
0x0C34	**DSI_DPCSRCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0C38 ~0x0C58	予約済み	予約済み																															
0x0C5C	**DSI_DPDL0HSOCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSPRPO [3:0]					Res.	Res.	Res.
	リセット値																										0	0	0				
0x0C60	**DSI_DPDL0LPXOCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPXO[3:0]			
	リセット値																																
0x0C64 ~0x0C6C	予約済み	予約済み																															
0x0C70	**DSI_DPDL0BCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BC[4:0]			
	リセット値																													0	0	0	0	0

表444.DSIレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C74 ~0x0C9C	予約済み	予約済み																															
0x0CA0	**DSI_DPDL0SRCR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	SRC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0CA4 ~0x0CF0	予約済み	予約済み																															
0x0CF4	**DSI_DPDL1HSOCR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	HSPRPO [3:0]				Res	Res	Res	Res
	リセット値																									0	0	0					
0x0CF8	**DSI_DPDL1LPXOCR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	LPXO[3:0]			
	リセット値																																
0x0CFC ~0x0D04	予約済み	予約済み																															
0x0D08	**DSI_DPDL1BCR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	BC[4:0]				
	リセット値																												0	0	0	0	0
0x0D0C ~0x0D34	予約済み	予約済み																															
0x0D38	**DSI_DPDL1SRCR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	SRC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

45 Neo-Chromグラフィック･プロセッサ（GPU2D）

このセクションは、STM32U599/5A9およびSTM32U5Fx/5Gxデバイスにのみ適用されます。

45.1 GPU2Dの概要

GPU2Dは、グラフィカル･ユーザ･インタフェース（GUI）、メニュー表示、アニメーションなどの多数の2.5Dグラフィック･アプリケーションを高速化する専用のグラフィックス処理ユニットです。

GPU2Dは、最先端のグラフィック･レンダリングのために設計された最適化されたソフトウェア･スタックと共に動作します。

45.2 GPU2Dの主な機能

主な特長

- VLIW（超長命令語）命令セットを備えたマルチスレッド･フラグメント（ピクセル）処理コア
- 固定小数点機能ユニット
- CPUオーバーヘッドを最小化するコマンドリストベースのDMA
- テクスチャ、コマンドリスト、フレームバッファアクセス用の2つの 32bit AHB マスタインタフェース
- レジスタバンクアクセス用の32bit AHBスレーブインタフェース
- システムレベルの同期用の最大4つの汎用フラグ
- TSC™4およびTSC™6/TSC™6aに対応したテクスチャ解凍ユニット

2D描画機能

- ピクセル／ライン描画
- 矩形塗りつぶし
- 三角形、四角形の描画
- アンチエイリアシング8xMSAA（マルチサンプル･アンチエイリアシング）

ベクタ･グラフィック･アクセラレーション

- パス描画（ライン、多角形、長方形、円弧、楕円、円）
- ベジェ曲線（3次曲線と2次曲線）
- パス変換（3x3行列）
- パス･ストローク
- 塗りつぶし（8x MSAAアンチエイリアシングによる偶数奇数規則と非ゼロ規則）
- グラデーション生成（線形、放射状、円錐形）

画像変換

- 3D透視正投影
- バイリニア･フィルタリングまたはポイント･サンプリングによるテクスチャ･マッピング

ビットマップ転送のサポート

- 回転、ミラーリング、ストレッチ（X軸とY軸で独立）
- 転送元や転送先でのカラーキーイング
- ピクセル･フォーマット変換

テキスト･レンダリングのサポート

- A1、A2、A4、およびA8bitマップ･アンチエイリアス
- サブサンプリング･アンチエイリアス

カラーフォーマット

- ABGR8888、ARGB8888、BGRA8888、RGBA8888
- xBGR8888、xRGB8888、BGRx8888、RGBx8888、RGB888、BGR888
- BGR565, RGB565
- ABGR1555、BGRA5551、RGBA5551、ARGB1555
- ABGR4444、BGRA4444、RGBA4444、ARGB4444
- RGB322, BGR322
- ABGR2222、BGRA2222、RGBA2222、ARGB2222
- TSC4、TSC6、TSC6A
- LUT1、LUT2、LUT4、LUT8
- L1、L2、L4、L8（グレースケール）
- A1、A2、A4、A8

ハードウェア･ブレンダによる完全アルファ･ブレンディング

- プログラム可能なブレンディングモード
- 転送元や転送先でのカラーキーイング

45.3 GPU2Dの実装

次の表に、GPU2Dのさまざまな実装を示します。

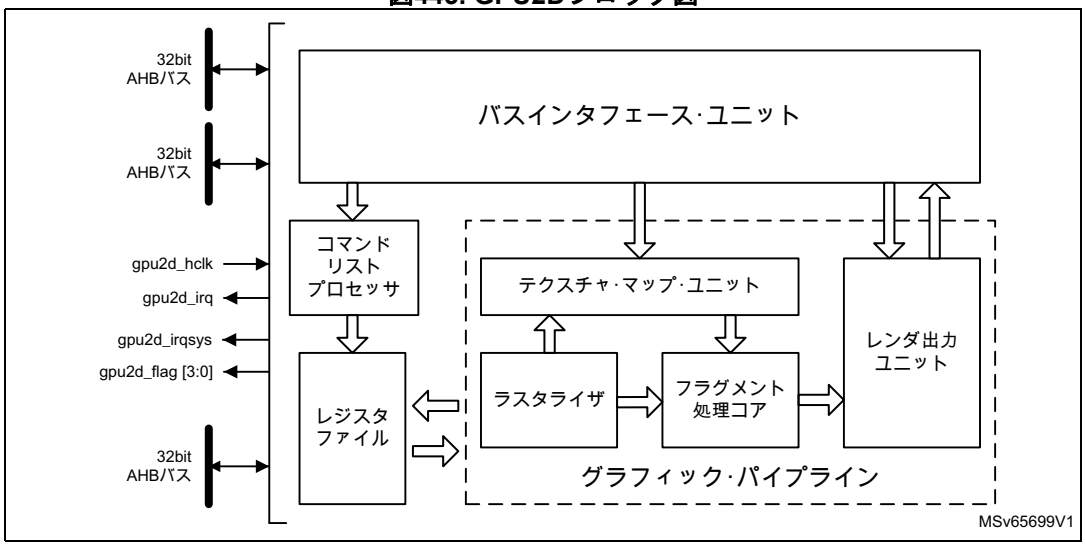
表445.GPU2Dの実装

GPU2Dの機能	STM32U599/5A9	STM32U5Fx/5Gx
ラスタ操作	X	X
ベクタ･グラフィック･アクセラレーション	-	X
追加カラーモード : ABGR1555、BGRA5551、RGBA5551、ARGB1555、ABGR4444、BGRA4444、RGBA4444、ARGB4444、ABGR2222、BGRA2222、RGBA2222、ARGB2222、LUT1、LUT2、LUT4、LUT8	-	X

45.4 GPU2Dの概要

45.4.1 GPU2Dブロック図

図443. GPU2Dブロック図



45.4.2 GPU2Dのピンおよび内部信号

GPU2Dの内部信号を次の表に示します。

表446.GPU2D内部入力／出力信号

内部信号名	信号タイプ	説明
gpu2d_hclk	入力	GPU2D AHBクロック
gpu2d_irq	出力	GPU2D割込みリクエスト
gpu2d_irqsys	出力	GPU2Dシステム割込みリクエスト
gpu2d_flag [3:0]	出力	GPU2D汎用フラグ

表447.GPU2Dトリガ接続

トリガ名	方向	トリガ転送元／転送先
gpu2d_flag[0]	出力	gpdma_trigsel[53]
gpu2d_flag[1]	出力	gpdma_trigsel[54]
gpu2d_flag[2]	出力	gpdma_trigsel[55]
gpu2d_flag[3]	出力	gpdma_trigsel[56]

46 JPEGコーデック（JPEG）

このセクションは、STM32U5Fx/5Gxにのみ適用されます。

46.1 はじめに

ハードウェア8bit JPEGコーデックでは、非圧縮画像データストリームのエンコードまたはJPEG圧縮された画像データストリームのデコードを行います。JPEGヘッダの完全な管理も行います。

46.2 JPEGコーデックの主な機能

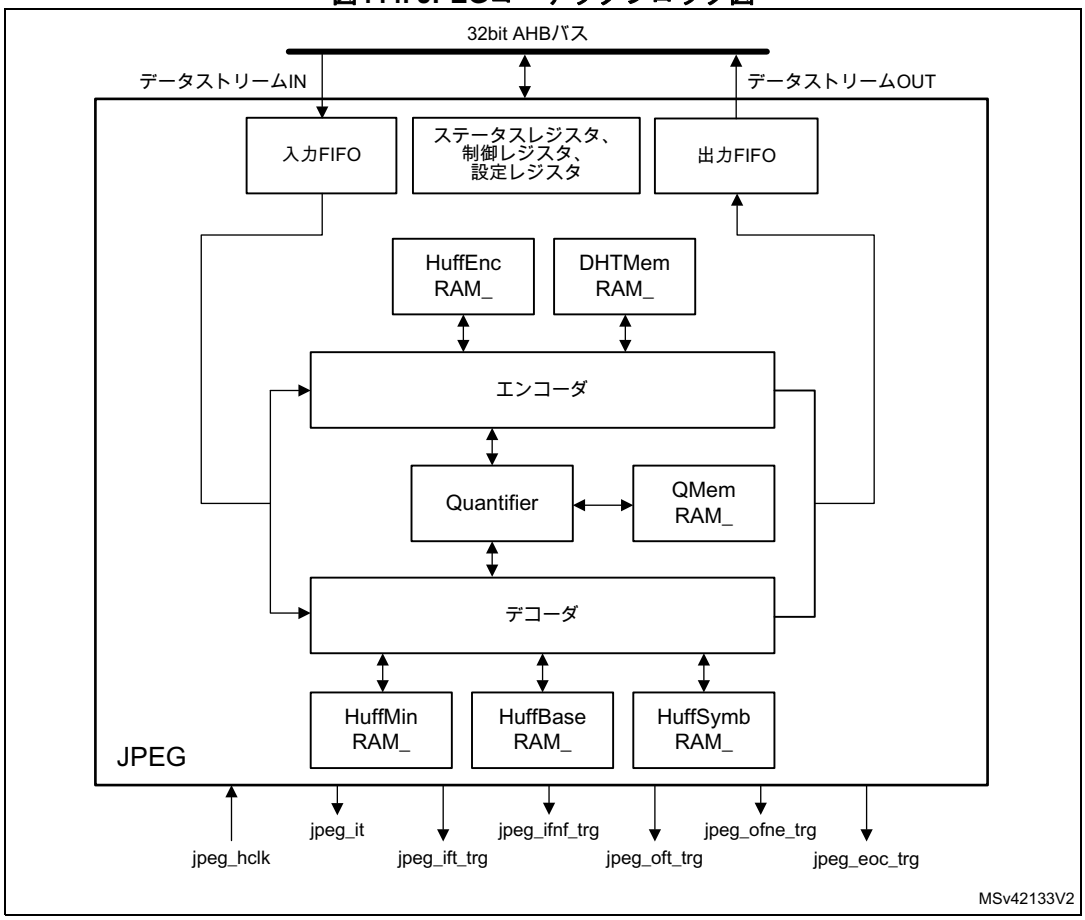
- ハイスピード完全同期動作
- エンコーダまたはデコーダとして設定可能
- ピクセルあたりシングルクロックのエンコード／デコード
- RGB、YCbCr、YCMK、BW（グレースケール）画像色空間対応
- エンコード／デコード時の画像成分あたり8bit深さ
- イネーブル／ディスエーブル付きのJPEGヘッダ生成器／パーサ
- 4つのプログラム可能な量子化テーブル
- シングルクロックのハフマンコーディング／デコーディング
- 完全プログラム可能なハフマンテーブル（2つのACと2つのDC）
- 完全プログラム可能な最小符号化単位（MCU）
- 同時入出力データストリームインタフェース

46.3 JPEGコーデックブロック の機能説明

46.3.1 概要

JPEGコーデックのブロック図を図444に示します。

図444. JPEGコーデックブロック図



46.3.2 JPEG内部信号

表448にJPEG内部信号を示します。

表448.JPEG内部信号

信号名	信号タイプ	説明
jpeg_hclk	デジタル入力	JPEGカーネル／レジスタインタフェースクロック
jpeg_it	デジタル出力	JPEGグローバル割込み
jpeg_ift_trg	デジタル出力	DMAトリガに対するJPEG入力FIFO閾値
jpeg_ifnf_trg	デジタル出力	DMAトリガに対するJPEG入力FIFO非フル
jpeg_oft_trg	デジタル出力	DMAトリガに対するJPEG出力FIFO閾値

表448.JPEG内部信号 （続き）

信号名	信号タイプ	説明
jpeg_ofne_trg	デジタル出力	DMAトリガに対するJPEG出力FIFO非エンプティ
jpeg_eoc_trg	デジタル出力	DMAトリガに対するJPEG変換終了

表449.JPEGトリガ接続

トリガ名	方向	トリガの転送元／転送先
jpeg_eoc_trg	出力	gpdma_trigsel[63]
jpeg_ifnf_trg	出力	gpdma_trigsel[64]
jpeg_ift_trg	出力	gpdma_trigsel[65]
jpeg_ofne_trg	出力	gpdma_trigsel[66]
jpeg_oft_trg	出力	gpdma_trigsel[67]

46.3.3 JPEGデコード手順

JPEGコーデックは、*ISO/IEC 10918-1* 規格で定義されているJPEGストリームをデコード可能です。

オプションとして、JPEGヘッダのパースと、それに応じてJPEGコーデックレジスタと量子化テーブルとハフマンテーブルの更新を行えます。

JPEGコーデックは、JPEG_CONFR1レジスタのDEビット（デコードイネーブル）をセットすることにより、デコードモードに設定します。

JPEG_CONFR0レジスタのスタートビットをセットすることによりJPEGデコードを開始します。

JPEGコーデックは、次のいずれかを生成することにより、その入力FIFOにデータを要求します。

- DMAリクエスト
- DMAトリガ
- 割込み

入力FIFO用DMA生成

DMAリクエストは、32バイトの入力FIFOが少なくともハーフエンプティになったとき、つまり16バイトのデータを書き込む空きがあるときに生成されます。

DMAリクエストの生成は、JPEG_CONFR0レジスタのSTARTビットには依存していません。入力FIFOが16バイトを受け入れ可能である場合、入力FIFOのDMAが有効になり（JPEG_CRレジスタのIDMAENビットをセット）、DMAリクエストがJPEGコーデックカーネルの状態とは無関係に生成されます。

16バイトのデータを書き込むため、DMAによってバースト転送が開始されます。

入力FIFOがフルであると、書込みは無視されます。

デコード処理の最後には、余分なバイトが入力FIFOに残っていたり、DMAリクエストが未処理となっていることがあります。JPEG_CRレジスタのIFFビット（入力FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前に、入力FIFOのDMAを無効として、FIFO一掃時の不要なDMAリクエストを防止してください。

デコード処理の最後にFIFOを一掃しないと、その結果として、残りデータが次回のJPEGデコードに使われてしまいます。

JPEG_SRレジスタのEOCFフラグがセットされると、DMAリクエストは生成されなくなります。

入力FIFOに対する割込みまたは DMAトリガの生成

入力FIFOは、FIFOの状態に応じた次の2つのフラグを介して、割込みまたは DMAトリガを用いて管理できます。

- 入力FIFO非フルフラグ：32bit値を書込み可能
- 入力FIFO閾値フラグ：8ワード（32バイト）を書込み可能。

割込みまたは DMAトリガの生成は、JPEG_CONFR0レジスタのSTARTビットには依存していません。入力FIFOフラグは、JPEGコーデックカーネルの状態とは無関係に生成されます。

入力FIFOがフルであると、書込みは無視されます。

デコード処理の最後には、余分なバイトが入力FIFOに残っていたり、割込みリクエスト/DMAトリガが未処理となっていることがあります。JPEG_CRレジスタのIFFビット（入力FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前には、次を行ってください。

- 入力FIFOの割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- DMAチャネルを停止して、不要な DMAトリガを防止してください。

デコード処理の最後にFIFOを一掃しないと、その結果として、残りデータが次回のJPEGデコードに使われてしまいます。

ヘッダの解析

ヘッダの解析は、JPEG_CONFR1レジスタのHDRビットをセットすることによってアクティブ化できます。

JPEGヘッダパーサは、ISO/IEC 10918-1規格附属書Bに示されているJPEGベースラインアルゴリズムに該当するすべてのマーカに対応しています。

対応マーカの解析時には、必要なパラメータをJPEGヘッダパーサが抽出して、シャドウレジスタに格納します。解析の最後に、JPEGコーデックレジスタは更新されます。

DQTマーカセグメントが位置している場合には、それに関連する量子化データが量子化テーブルメモリに書き込まれます。

DHTマーカセグメントが位置している場合には、それに関連するハフマンテーブルデータが3種類の異なるテーブル形式（HuffMin、HuffBase、HuffSymb）へと変換されて、それぞれのメモリに格納されます。

解析動作が完了すると、JPEG_SRレジスタのHPDF（ヘッダ解析完了フラグ）ビットがセットされます。JPEG_CRレジスタのEHPIE（ヘッダ解析完了割込みイネーブル）ビットがセットされていると、割込みが生成されます。

JPEGデコード

JPEGヘッダの解析が行われるか、JPEGコーデックのレジスタとメモリが正常にプログラムされるかすると、受信データストリームはデコードされ、結果として得られたMCUは出力FIFOに送られます。

画像2個を連続デコードする場合には、JPEG_CONFR0レジスタのスタートビットは、2番目の画像のヘッダ処理が干渉したあとに、（すでに1であったとしても）再びセットする必要があります。

出力FIFO用DMA生成

DMAリクエストは、32バイトの出力FIFOが少なくともハーフフルになったとき、つまり最低16バイトのデータがあるときに生成されます。

16バイトのデータを読み込むため、DMAによってバースト転送が開始されます。

出力FIFOがエンプティのときには、読むと0が返ります。

デコード処理が完了すると、JPEGデコードにより64バイトのブロックが生成されるため、余分なバイトが出力FIFOに残っていたり、DMAリクエストが未処理となっていてはなりません。

JPEG_CONFR0レジスタのSTARTビットをリセットすることにより、JPEGコーデック処理が異常終了した場合には、JPEG_CRレジスタのOFFビット（入力FIFO一掃）を設定することによって、出力FIFOを一掃できます。

FIFOを一掃する前に、出力FIFOのDMAを無効として、FIFO一掃時の不要なDMAリクエストを防止してください。

出力FIFOに対する割込みまたは DMAトリガの生成

出力FIFOは、FIFOの状態に応じた次の2つのフラグを介して、割込みまたは DMAトリガを用いて管理できます。

- 出力FIFO非エンプティフラグ：32bit値を読出し可能
- 出力FIFO閾値フラグ：8ワード（32バイト）を読出し可能。

出力FIFOがエンプティのときには、読むと0が返ります。

JPEG_CONFR0レジスタのスタートビットをリセットすることにより、JPEGコーデック処理が異常終了した場合には、出力FIFOを一掃できます。FIFOを一掃する必要がある場合には、JPEG_CRレジスタのFFビット（FIFO一掃）を設定することにより、ソフトウェアで行ってください。

FIFOを一掃する前には、次を行ってください。

- 出力FIFOの割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- DMAチャネルを停止して、不要な DMAトリガを防止してください。

JPEG設定が何か変化する前に、処理の最後で出力FIFOを一掃する必要があります。

46.3.4 JPEGエンコード手順

JPEGコーデックは、*ISO/IEC 10918-1* 規格で定義されているJPEGストリームをデエンコード可能です。

オプションで、JPEGヘッダの生成も行います。

JPEGコーデックは、JPEG_CONFR1レジスタのDEビット（デコードイネーブル）をリセットすることにより、エンコードモードに設定します。

JPEGのエンコードに使用された設定は、JPEGコーデックにロードされる必要があります。

- JPEGコーデック設定レジスタ
- 量子化テーブル
- ハフマンテーブル

JPEG_CONFR0レジスタのスタートビットをセットすることによりJPEGコーデックを開始します。

いったんJPEGコーデックが開始されると、次のいずれかを生成する入力FIFOのデータを要求します。

- DMAリクエスト
- DMAトリガ
- 割込み

入力FIFO用DMA生成

DMAリクエストは、32バイトの入力FIFOが少なくともハーフエンプティになったとき、つまり16バイトのデータを書き込む空きがあるときに生成されます。

DMAリクエストの生成は、JPEG_CONFR0レジスタのSTARTビットには依存していません。入力FIFOが16バイトを受け入れ可能である場合、入力FIFOのDMAが有効になり（JPEG_CRレジスタのIDMAENビットをセット）、DMAリクエストがJPEGコーデックカーネルの状態とは無関係に生成されます。

16バイトのデータを書き込むため、DMAによってバースト転送が開始されます。

入力FIFOがフルであると、書込みは無視されます。

エンコード処理の最後には、余分なバイトが入力FIFOに残っていたり、DMAリクエストが未処理となっていることがあります。JPEG_CRレジスタのIFFビット（入力FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前に、入力FIFOのDMAを無効として、FIFO一掃時の不要なDMAリクエストを防止してください。

エンコード処理の最後にFIFOを一掃しないと、その結果として、残りデータが次回のJPEGエンコードに使われてしまいます。

JPEG_SRレジスタのEOCFフラグがセットされると、DMAリクエストは生成されなくなります。

入力FIFOに対する割込みまたは DMAトリガの生成

入力FIFOは、FIFOの状態に応じた次の2つのフラグを介して、割込みまたは DMAトリガを用いて管理できます。

- 入力FIFO非フルフラグ：32bit値を書込み可能
- 入力FIFO閾値フラグ：8ワード（32バイト）を書込み可能。

割込みまたは DMAトリガの生成は、JPEG_CONFR0レジスタのSTARTビットには依存していません。入力FIFOフラグは、JPEGコーデックカーネルの状態とは無関係に生成されます。

入力FIFOがフルであると、書込みは無視されます。

エンコード処理の最後には、余分なバイトが入力FIFOに残っていたり、割込みリクエスト/DMAトリガが未処理となっていることがあります。JPEG_CRレジスタのIFFビット（入力FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前には、次を行ってください。

- 入力FIFOの割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- DMAチャネルを停止して、不要な DMAトリガを防止してください。

エンコード処理の最後にFIFOを一掃しないと、その結果として、残りデータが次回のJPEGエンコードに使われてしまいます。

JPEGエンコード

JPEGヘッダが生成されると、受信MCUはエンコードされて、結果として得られたデータストリームは出力FIFOに送られます。

出力FIFO用DMA生成

DMAリクエストは、32バイトの出力FIFOが少なくともハーフフルになったとき、つまり最低16バイトのデータがあるときに生成されます。

16バイトのデータを読み込むため、DMAによってバースト転送が開始されます。

出力FIFOがエンプティのときには、読むと0が返ります。

エンコード処理の最後には、ストリームパディングが16バイトにないことがあるため、最後のバイトが出力FIFOに残る場合があります。

これらの追加バイトは、CPUが出力FIFO非エンプティフラグを用いて管理する必要があります。

JPEG_CONFR0レジスタのスタートビットをリセットすることにより、JPEGコーデック処理が異常終了した場合には、出力FIFOを一掃できます。JPEG_CRレジスタのOFFビット（出力FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前に、入力FIFOのDMAを無効として、FIFO一掃時の不要なDMAリクエストを防止してください。

出力FIFOに対する割込みまたは DMAトリガの生成

出力FIFOは、FIFOの状態に応じた次の2つのフラグを介して、割込みまたは DMAトリガを用いて管理できます。

- 出力FIFO非エンプティフラグ：32bit値を読出し可能
- 出力FIFO閾値フラグ：8ワード（32バイト）を読出し可能。

出力FIFOがエンプティのときには、読むと0が返ります。

JPEG_CONFR0レジスタのスタートビットをリセットすることにより、JPEGコーデック処理が異常終了した場合には、出力FIFOを一掃できます。JPEG_CRレジスタのFFビット（FIFO一掃）を設定することによって、FIFOを一掃できます。

FIFOを一掃する前には、次を行ってください。

- 出力FIFOの割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- DMAチャネルを停止して、不要な DMAトリガを防止してください。

JPEG設定が何か変化する前に、処理の最後で出力FIFOを一掃する必要があります。

JPEG_SRレジスタのEOCFビット（変換終了フラグ）は、出力FIFOがエンプティであるときにのみクリアできます。

JPEG_CONFR1レジスタのHDRビット（ヘッダ処理）か、JPEG_CRレジスタのJCENビット（JPEGコーデックイネーブル）かをクリアできるのは、JPEG_SRレジスタのEOCFビットがクリアされているときに限られます。

46.4 JPEGコーデック割込み

割込みは次のイベントによって生成できます。

- 入力FIFO閾値到達
- 入力FIFO非フル
- 出力FIFO閾値到達
- 出力FIFO非エンプティ
- 変換終了
- ヘッダ解析終了

高い柔軟性を実現するため、個別の割込みイネーブル･ビットを使用できます。

表450.JPEGコーデック割込み

割込みイベント	イベント･フラグ	イネーブル制御ビット
入力FIFO閾値到達	IFTF	IFTIE
入力FIFO非フル	IFNFF	IFNFIE
出力FIFO閾値到達	OFTF	OFTIE
出力FIFO非エンプティ	OFNEF	OFNEIE
変換の終了	EOCF	EOCIE
ヘッダ解析終了	HPDF	HPDIE

46.5 JPEGコーデックレジスタ

46.5.1 JPEGコーデック制御レジスタ（JPEG_CONFR0）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START
															w

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **START**：開始

このビットは、エンコード処理またはデコード処理の開始または停止を行います。

0：停止/中止

1：開始

注： このビットを読み出すと常に0が返されます。

46.5.2 JPEGコーデック設定レジスタ1（JPEG_CONFR1）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
YSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	
							rw	rw	rw	rw	rw	rw		rw	rw

ビット31:16 **YSIZE[15:0]**：Yサイズ

このフィールドにより、ソース画像内のライン数を定義します。

ビット15:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **HDR**：ヘッダ処理

このビットにより、ヘッダ処理（生成／解析）が有効となります。

0：無効

1：有効

ビット7:6 **NS[1:0]**：スキャンする成分数

このフィールドにより、スキャンヘッダマーカセグメントの成分数から1を引いた値を定義します。

ビット5:4 **COLSPACE[1:0]**：色空間

このフィールドにより、出力ストリームに挿入する量子化テーブル数から1を引いた値を定義します。

00：グレースケール（量子化テーブル1つ）

01：YUV（量子化テーブル2つ）

10：RGB（量子化テーブル3つ）

11：CMYK（量子化テーブル4つ）

ビット3 **DE**：エンコーダまたはデコーダとしてのコーデック操作

このビットは、エンコード処理またはデコード処理を選択します。

0：エンコード

1：デコード

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **NF[1:0]**：色成分の数

このフィールドにより、色成分数から1を引いた値を定義します。

00：グレースケール（色成分1つ）

01：-（色成分2つ）

10：YUVまたはRGB（色成分3つ）

11：CMYK（色成分4つ）

46.5.3 JPEGコーデック設定レジスタ2（JPEG_CONFR2）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	NMCU[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NMCU[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:0 **NMCU[25:0]**：MCUサイクルの数

エンコード時；このフィールドにより、エンコードするMCU単位数から1を引いた値を定義します。

デコード時；このフィールドは、デコードされる完全なMCU単位数から1を引いた値を示します（このフィールドはJPEGヘッダの解析後に更新されます）。デコード画像のXやYのサイズが8または16（サブサンプリング処理に依存します）の倍数ではない場合には、生成されるMCUの総数を得るためには、結果として得られる不完全または空であるMCUをこの値に加算する必要があります。

46.5.4 JPEGコーデック設定レジスタ3（JPEG_CONFR3）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:16 **XSIZE[15:0]**：Xサイズ

このフィールドにより、ラインあたりのピクセル数を定義します。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

46.5.5 JPEGコーデック設定レジスタx（JPEG_CONFRx）

アドレスオフセット：0x000 + 0x4 * x（x = 4から7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **HSF[3:0]**：水平サンプリング係数
成分 {x-4} の水平サンプリング係数。

ビット11:8 **VSF[3:0]**：垂直サンプリング係数
成分 {x-4} の垂直サンプリング係数。

ビット7:4 **NB[3:0]**：ブロックの数
MCUの中である特定色に属するデータ単位数から1を引いた値

ビット3:2 **QT[1:0]**：量子化テーブル
成分 {x-4} の量子化テーブルを選択。
00：量子化テーブル0
01：量子化テーブル1
10：量子化テーブル2
11：量子化テーブル3

ビット1 **HA**：ハフマンAC
AC係数のエンコードのためのハフマンテーブルを選択
0：ハフマンACテーブル0
1：ハフマンACテーブル1

ビット0 **HD**：ハフマンDC
DC係数のエンコードのためのハフマンテーブルを選択
0：ハフマンDCテーブル0
1：ハフマンDCテーブル1

46.5.6 JPEG制御レジスタ（JPEG_CR）

アドレスオフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	オフ	IFF	ODMAEN	IDMAEN	Res.	Res.	Res.	Res.	HPDIE	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN
	w	w	rw	rw					rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **OFF**：出力FIFO一掃

このビットは、出力FIFOを一掃します。

0：影響なし

1：出力FIFOを一掃

注： このビットを読み出すと常に0が返されます。

ビット13 **IFF**：入力FIFO一掃

このビットは、入力FIFOを一掃します。

0：影響なし

1：入力FIFOを一掃

注： このビットを読み出すと常に0が返されます。

ビット12 **ODMAEN**：出力DMAイネーブル

出力FIFO用DMAリクエスト生成を有効化します。

0：無効

1：有効

ビット11 **IDMAEN**：入力DMAイネーブル

入力FIFO用DMAリクエスト生成を有効化します。

0：無効

1：有効

ビット 10:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **HPDIE**：ヘッダ解析終了割込みイネーブル

このビットは、ヘッダ解析動作の完了時の割込み生成を有効化します。

0：無効

1：有効

ビット5 **EOCIE**：変換終了割込み有効化

このビットは、変換終了時の割込み生成を有効にします。

0：無効

1：有効

ビット4 **OFNEIE**：出力FIFO非エンプティ割込み有効

このビットは、出力FIFOがエンプティではないときの割込み生成を有効にします。

0：無効

1：有効

ビット3 **OFTIE**：出力FIFO閾値割込み有効

このビットは、出力FIFOが閾値に達したときの割込み生成を有効にします。

0：無効

1：有効

ビット2 **IFNFIE**：入力FIFO非フル割込み有効

このビットは、入力FIFOがエンプティではないときの割込み生成を有効にします。

0：無効

1：有効

ビット1 **IFTIE**：入力FIFO閾値割込み有効

このビットは、入力FIFOが閾値に達したときの割込み生成を有効にします。

0：無効

1：有効

ビット0 **JCEN**：JPEGコアイネーブル

このビットは、JPEGコーデックコアを有効にします。

0：無効（内部レジスタがリセット）

1：有効（内部レジスタがアクセス可能）

46.5.7 JPEGステータスレジスタ（JPEG_SR）

アドレスオフセット：0x034

リセット値：0x0000 0006

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.
								r	r	r	r	r	r	r	

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **COF**：コーデック操作フラグ

このビットは、エンコード／デコード操作が進行中であることを示します。

0：非進行中

1：進行中

ビット6 **HPDF**：ヘッダ解析終了フラグ

デコードモードにおいて、このビットはヘッダ構文解析と内部レジスタの更新が完了したことを示します。

0：未完

1：完了

ビット5 **EOCF**：変換終了フラグ

このビットは、エンコード／デコード処理と出力FIFOに対するデータ転送が完了したことを示します。

0：未完

1：完了

ビット4 **OFNEF**：出力FIFO非エンプティフラグ

このビットは、データが出力FIFOにあることを示します。DMAを使用する場合は、このフラグを考慮してはなりません。

0：エンプティ（データなし）

1：非エンプティ（データあり）

ビット3 **OFTF**：出力FIFO閾値フラグ

このビットは、出力FIFOのデータ量が閾値に達したか超過したことを示します。DMAを使用する場合は、このフラグを考慮してはなりません。

0：閾値未満

1：閾値以上

ビット2 **IFNFF**：入力FIFO非フルフラグ

このビットは、入力FIFOがフルではないことを示します（データ書込み可）。DMAを使用する場合は、このフラグを考慮してはなりません。

0：フル

1：非フル

ビット1 **IFTF**：入力FIFO閾値フラグ

このビットは、入力FIFOのデータ量が閾値未満であることを示します。DMAを使用する場合は、このフラグを考慮してはなりません。

0：閾値以上

1：閾値未満

ビット0 予約済みであり、リセット値に保持する必要があります。

46.5.8 JPEGクリアフラグレジスタ（JPEG_CFR）

アドレスオフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
									rt_w1	rt_w1					

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CHPDF**：クリアヘッダ解析終了フラグ

“1”を書き込むとJPEG_SRレジスタのHPDFビットをクリアします。

0：影響なし

1：クリア

ビット5 **CEOCF**：クリア変換終了フラグ

“1”を書き込むとJPEG_SRレジスタのECFビットをクリアします。

0：影響なし

1：クリア

ビット4:0 予約済みであり、リセット値に保持する必要があります。

46.5.9 JPEGデータ入力レジスタ（JPEG_DIR）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAIN[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAIN[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **DATAIN[31:0]**：データ入力FIFO
入力FIFOデータレジスタ

46.5.10 JPEGデータ出力レジスタ（JPEG_DOR）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAOUT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAOUT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **DATAOUT[31:0]**：データ出力FIFO
出力FIFOデータレジスタ

46.5.11 JPEG量子化メモリx（JPEG_QMEMx_y）

アドレスオフセット：0x050 + 0x40 * x + 0x4 * y（x = 0から3、y = 0から15）

リセット値：0xXXXX XXXX

ISOドキュメントで指定されている4つの量子化テーブル。

ヘッダの解析によるデコードの場合、量子化テーブルのプログラミングは必要ありません。係数は、ヘッダパーサによって量子化メモリに直接書き込まれます。

ヘッダの解析なしのデコードまたはエンコードの場合、量子化テーブルはソフトウェアでジグザグの順に書き込む必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
QCOEF{4*y+3}[7:0]								QCOEF{4*y+2}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
QCOEF{4*y+1}[7:0]								QCOEF{4*y}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **QCOEF{4*y+3}[7:0]**：量子化係数 {4*y+3}
8bit量子化係数

ビット23:16 **QCOEF{4*y+2}[7:0]**：量子化係数 {4*y+2}
8bit量子化係数

ビット15:8 **QCOEF{4*y+1}[7:0]**：量子化係数 {4*y+1}
8bit量子化係数

ビット7:0 **QCOEF{4*y}[7:0]**：量子化係数 {4*y}
8bit量子化係数

46.5.12 JPEGハフマン最小（JPEG_HUFFMINx_y）

アドレスオフセット：0x150 + 0x10 * x + 0x4 * y（x = 0から3、y = 0から2）

リセット値：0xXXXX XXXX

このメモリには、JPEGデコーダによって内部的に使用される最小のハフマン値が格納されます。メモリの内容はヘッダの解析時にハードウェアによって書き込まれます。

- DATA0：最小AC0値
- DATA1：最小DC0値
- DATA2：最小AC1値
- DATA3：最小DC1値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA{x}[{32*y+31}:{32*y+16}]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA{x}[{32*y+15}:{32*y}]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATA{x}[{32*y+31}:{32*y}]**：最小ハフマン値

JPEGデコーダによって内部的に使用される100ビットの最小ハフマン値。

46.5.13 JPEGハフマン最小x（JPEG_HUFFMINx_y）

アドレスオフセット：0x150 + 0x10 * x+ 0x4 * y（(x = 0から3、y =3）

リセット値：0xXXXX XXXX

このメモリには、JPEGデコーダによって内部的に使用される最小のハフマン値が格納されます。メモリの内容はヘッダの解析時にハードウェアによって書き込まれます。

- DATA0：最小AC0値
- DATA1：最小DC0値
- DATA2：最小AC1値
- DATA3：最小DC1値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{x}[99: 96]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **DATA{x}[99:96]**：最小ハフマン値

JPEGデコーダによって内部的に使用される100ビットの最小ハフマン値。

46.5.14 JPEGハフマンベース（JPEG_HUFFBASEx）

アドレスオフセット：0x190 + 0x4 * x（x = 0から31）

リセット値：0xXXXX XXXX

このメモリには、JPEGデコーダによって内部的に使用されるベースのハフマン値が格納されます。メモリの内容はヘッダの解析時にハードウェアによって次のように書き込まれます。

- DATA0からDATA15：ベースAC0値
- DATA16からDATA31：ベースDC0値
- DATA32からDATA47：ベースAC1値
- DATA48からDATA63：ベースDC1値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{2*x+1}[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{2*x}[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24:16 **DATA{2*x+1}[8:0]**：データ {2*x+1}
ベースハフマン値

ビット15:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **DATA{2*x}[8:0]**：データ {2*x}
ベースハフマン値

46.5.15 JPEGハフマンシンボル（JPEG_HUFFSYMBx）

アドレスオフセット：0x210 + 0x4 * x（x = 0から83）

リセット値：0xXXXX XXXX

このメモリには、JPEGデコーダによって内部的に使用されるハフマンシンボルが格納されます。メモリの内容はヘッダの解析時にハードウェアによって次のように書き込まれます。

- DATA0からDATA161：AC0シンボル
- DATA162からDATA173：DC0およびDC1シンボル
- DATA174からDATA335：AC1シンボル

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA{4*x+3}[7:0]								DATA{4*x+2}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA{4*x+1}[7:0]								DATA{4*x}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **DATA{4*x+3}[7:0]**：データ {4*x+3}
ハフマンシンボル

ビット23:16 **DATA{4*x+2}[7:0]**：データ {4*x+2}
ハフマンシンボル

ビット15:8 **DATA{4*x+1}[7:0]**：データ {4*x+1}
ハフマンシンボル

ビット7:0 **DATA{4*x}[7:0]**：データ {4*x}
ハフマンシンボル

46.5.16 JPEG DHTメモリ（JPEG_DHTMEMx）

アドレスオフセット：0x360 + 0x4 * x（x = 0から102）

リセット値：0xXXXX XXXX

ヘッダ生成を伴うエンコードプロセスの場合、このメモリにはDHTマーカセグメントのACおよびDCハフマンテーブルがISO/IEC仕様形式で格納されます。

- DATA0からDATA27：DCハフマンテーブル0
- DATA28からDATA205：ACハフマンテーブル0
- DATA206からDATA233：DCハフマンテーブル1
- DATA234からDATA411：ACハフマンテーブル1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA{4*x+3}[7:0]								DATA{4*x+2}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA{4*x+1}[7:0]								DATA{4*x}[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **DATA{4*x+3}[7:0]**：ハフマンテーブルデータ {4*x+3}
DHTマーカセグメント生成のためのハフマンテーブルデータ。

ビット23:16 **DATA{4*x+2}[7:0]**：ハフマンテーブルデータ {4*x+2}
DHTマーカセグメント生成のためのハフマンテーブルデータ。

ビット15:8 **DATA{4*x+1}[7:0]**：ハフマンテーブルデータ {4*x+1}
DHTマーカセグメント生成のためのハフマンテーブルデータ。

ビット7:0 **DATA{4*x}[7:0]**：ハフマンテーブルデータ {4*x}
DHTマーカセグメント生成のためのハフマンテーブルデータ。

46.5.17 JPEGハフマンエンコードACx（JPEG_HUFFENC_ACx_y）

アドレスオフセット：0x500 + 0x160 * x + 0x4 * y（x = 0から1、y = 0から87）

リセット値：0xXXXX XXXX

このメモリは、ACコンポーネントのエンコードプロセス中に使用されるハフマンコードを定義します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]				HCODE{2*y}[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:24 **HLEN{2*y+1}[3:0]**：ハフマン長 {2*y+1}
ハフマンコードHCODE{2*y+1} のビット数 - 1。

ビット23:16 **HCODE{2*y+1}[7:0]**：ハフマンコード {2*y+1}
ハフマンコードの最下位8ビット。
ハフマンコードの長さが8ビット未満の場合、未使用のビットは0でなければなりません。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **HLEN{2*y}[3:0]**：ハフマン長 {2*y}
ハフマンコードHCODE{2*y} のビット数 - 1。

ビット7:0 **HCODE{2*y}[7:0]**：ハフマンコード {2*y}
ハフマンコードの最下位8ビット。
ハフマンコードの長さが8ビット未満の場合、未使用のビットは0でなければなりません。

46.5.18 JPEGハフマンエンコードDCx（JPEG_HUFFENC_DCx_y）

アドレスオフセット：0x7C0 + 0x20 * x + 0x4 * y（x = 0から1、y = 0から7）

リセット値：0xXXXX XXXX

このメモリは、DCコンポーネントのエンコードプロセス中に使用されるハフマンコードを定義します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]				HCODE{2*y}[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:24 **HLEN{2*y+1}[3:0]**：ハフマン長 {2*y+1}
ハフマンコードHCODE{2*y+1} のビット数 - 1。

ビット23:16 **HCODE{2*y+1}[7:0]**：ハフマンコード {2*y+1}
ハフマンコードの最下位8ビット。
ハフマンコードの長さが8ビット未満の場合、未使用のビットは0でなければなりません。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **HLEN{2*y}[3:0]**：ハフマン長 {2*y}
ハフマンコードHCODE{2*y} のビット数 - 1。

ビット7:0 **HCODE{2*y}[7:0]**：ハフマンコード {2*y}
ハフマンコードの最下位8ビット。
ハフマンコードの長さが8ビット未満の場合、未使用のビットは0でなければなりません。

46.5.19 JPEGコーデックレジスタマップ

次の表にJPEGコーデックレジスタの一覧を示します。JPEGコーデックレジスタのベースアドレスについては、レジスタ境界アドレス表を参照してください。

表451.JPEGコーデックレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	JPEG_CONFR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START
	リセット値																																0
0x004	JPEG_CONFR1	YSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0		0	0
0x008	JPEG_CONFR2	Res.	Res.	Res.	Res.	Res.	Res.	NMCU[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	JPEG_CONFR3	XSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
0x010	JPEG_CONFR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	JPEG_CONFR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	JPEG_CONFR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	JPEG_CONFR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020 ~ 0x02C	予約済み	予約済み																															
0x030	JPEG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	オフ	IFF	ODMAEN	IDMAEN	Res.	Res.	Res.	HPDIE	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN	
	リセット値																		0	0	0	0				0	0	0	0	0	0	0	0
0x034	JPEG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.	
	リセット値																								0	0	0	0	0	1	1		

表451.JPEGコーデックレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x038	JPEG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
	リセット値																										0	0					
0x040	JPEG_DIR	DATAIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	JPEG_DOR	DATAOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048 ~ 0x04C	予約済み	予約済み																															
0x050 ~0x08C	JPEG_QMEM0	QCOEF{4*y+3}[7:0]								QCOEF{4*y+2}[7:0]								QCOEF{4*y+1}[7:0]								QCOEF{4*y}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x090 ~0x0CC	JPEG_QMEM1	QCOEF{4*y+3}[7:0]								QCOEF{4*y+2}[7:0]								QCOEF{4*y+1}[7:0]								QCOEF{4*y}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0D0 ~0x10F	JPEG_QMEM2	QCOEF{4*y+3}[7:0]								QCOEF{4*y+2}[7:0]								QCOEF{4*y+1}[7:0]								QCOEF{4*y}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x110 ~0x14C	JPEG_QMEM3	QCOEF{4*y+3}[7:0]								QCOEF{4*y+2}[7:0]								QCOEF{4*y+1}[7:0]								QCOEF{4*y}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x150 ~0x18C	JPEG_HUFFMIN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{x}[99:96]			
		DATA{x}[95:64]																															
		DATA{x}[63:32]																															
		DATA{x}[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x190 ~ 0x20C	JPEG_HUFFBASE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{2*x+1}[8:0]									Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA{2*x}[8:0]								
	リセット値								X	X	X	X	X	X	X	X	X								X	X	X	X	X	X	X	X	X
0x210 ~0x35C	JPEG_HUFFSYMB	DATA{4*x+3}[7:0]								DATA{4*x+2}[7:0]								DATA{4*x+1}[7:0]								DATA{4*x}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x360 ~0x4F8	JPEG_DHTMEM	DATA{4*x+3}[7:0]								DATA{4*x+2}[7:0]								DATA{4*x+1}[7:0]								DATA{4*x}[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x4FC	予約済み	Res.																															
0x500 ~ 0x65C	JPEG_HUFFENC_AC0	Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]								Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]								HCODE{2*y}[7:0]			
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x660 ~0x7BC	JPEG_HUFFENC_AC1	Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]								Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]								HCODE{2*y}[7:0]			
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x7C0 ~0x7DC	JPEG_HUFFENC_DC0	Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]								Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]								HCODE{2*y}[7:0]			
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x7E0 ~0x7FC	JPEG_HUFFENC_DC1	Res.	Res.	Res.	Res.	HLEN{2*y+1}[3:0]				HCODE{2*y+1}[7:0]								Res.	Res.	Res.	Res.	HLEN{2*y}[3:0]								HCODE{2*y}[7:0]			
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、セクション2.3を参照してください。

47 タッチセンシングコントローラ（TSC）

47.1 概要

タッチセンシングコントローラを使用することで、どんなアプリケーションにも簡単に静電容量検出機能を追加することができます。静電容量検出技術とは、誘電材料（ガラス、プラスチックなど）により直接触れることがないよう保護されている電極の近くに近づけた指を検出することができる技術です。指（または何らかの導電性物体）によって発生する静電容量の変化は、表面電荷移動取得原理に基づき、実証済みの方法で測定されます。

タッチセンシングコントローラは、STMTouchタッチ検出ファームウェアライブラリによってフルサポートされています。この無料で利用できるライブラリにより、タッチ検出機能がエンドアプリケーションに確実に実装されます。

47.2 TSCの主な機能

タッチセンシングコントローラは、以下のような主な機能を備えています。

- 実績のある安定した表面電荷移動取得原理を採用。
- 最大 24 本の静電容量検出チャネルに対応。
- 最大8本の静電容量検出チャネルの 同時取得により、非常に優れた応答時間を実現。
- スペクトル拡散機能により、ノイズの多い環境でのシステムの安定性が向上。
- 電荷移動取得シーケンスのハードウェア完全管理
- プログラム可能な電荷移動周波数
- プログラム可能なサンプリングコンデンサのI/Oピン
- プログラム可能なチャネルのI/Oピン
- 最大カウント値をプログラムできるので、チャネル障害時の取得時間の延長を回避。
- 割込み機能を持つ専用の取得完了フラグおよび最大カウントエラーフラグ
- 最高3本の静電容量検出チャネルに対して1個のサンプリングコンデンサとすることで、システム構成部品の個数を削減。
- 近接、タッチキー、リニア、およびロータリタイプのタッチセンサに対応。
- STMTouchタッチ検出ファームウェアライブラリで動作。

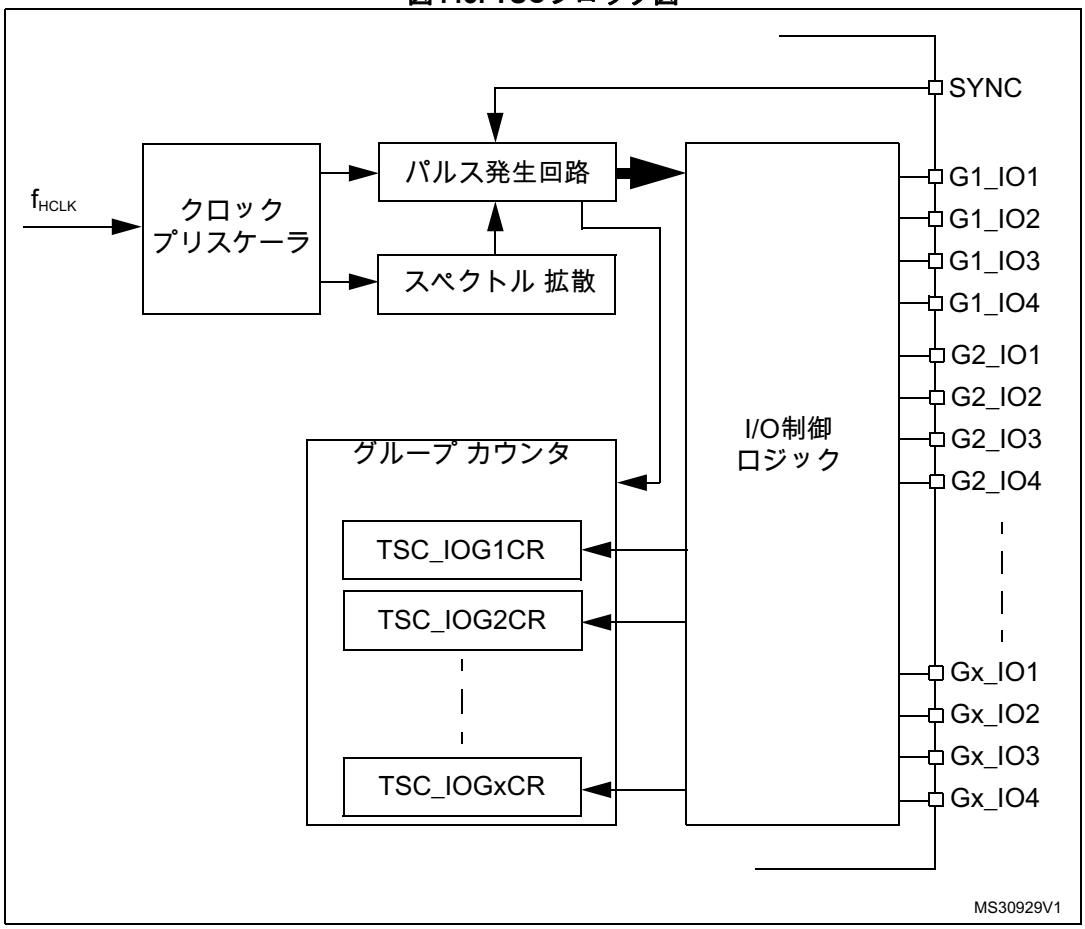
注： **静電容量検出チャネルの本数は、パッケージのサイズ、および使用可能なI/Oがあるかどうかによって決まります。**

47.3 TSCの機能詳細

47.3.1 TSCブロック図

タッチセンシングコントローラのブロック図を 図445に示します。

図445. TSCブロック図



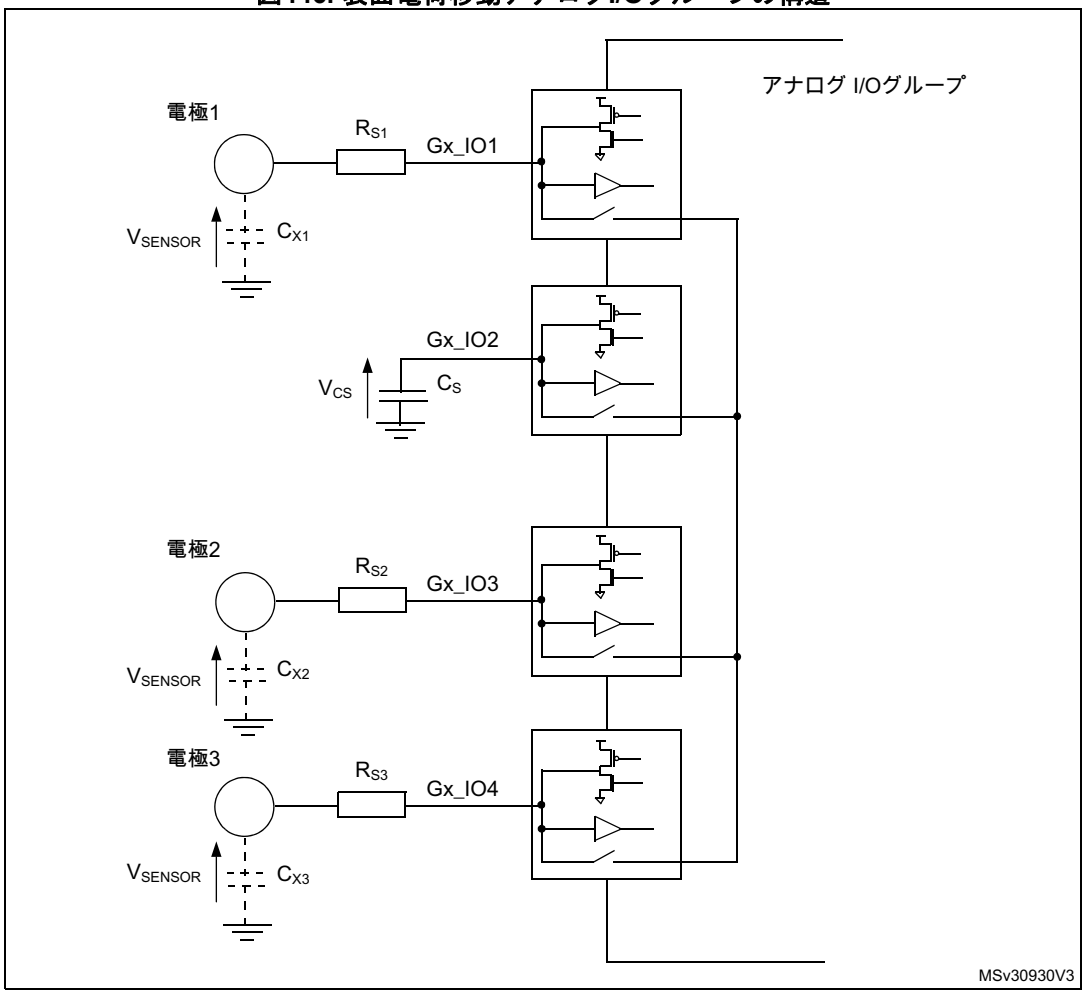
47.3.2 表面電荷移動取得の概要

表面電荷移動取得は、静電容量を計測するための実績のある、安定した効果的な方法です。この方法では、シングルエンド電極型で動作するために最低限必要な数の外部コンポーネントが使用されます。この取得は、最大4つのGPIOから成るアナログI/Oグループを中心に設計されています（図446を参照）。いくつかのアナログI/Oグループを使用することができ、それにより複数の静電容量検出チャネルを同時に取得し、サポートできる静電容量検出チャネルの数を増やすことができます。1つのアナログI/Oグループ内で行われる静電容量検出チャネルの取得はシーケンシャルです。

GPIOのうちの1つは、サンプリングコンデンサC_S 専用です。1つのアナログI/Oグループでは、一度に1つのサンプリングコンデンサI/Oだけを有効にしてください。

残りのGPIOは電極に使用され、一般にチャネルと呼ばれます。特定のニーズ（近接検出など）については、1つのアナログI/Oグループで同時に1つ以上のチャネルを有効にすることができます。

図446. 表面電荷移動アナログI/Oグループの構造



注： **Gx_IOyのxはアナログI/Oグループの番号、yは選択したグループ内でのGPIOの番号を表します。**

表面電荷移動取得の基本動作は、電極の静電容量（C_X）を充電することと、蓄積された電荷の一部をサンプリングコンデンサ（C_S）へ移動することです。このシーケンスは、C_S の電圧が所定の閾値（弊社の場合、V_{IH} ）に達するまで繰り返されます。閾値に達するために必要な電荷移動の回数は、電極の静電容量の大きさで直接表されます。

表452に、静電容量検出チャネル1の電荷移動取得シーケンスの詳細を示します。C_S の電圧が所定の閾値に達するまで、状態3～7が繰り返されます。その他のチャネルの取得にも同じシーケンスが適用されます。電極のシリアルレジスタ R_S によって、この方法のESD（静電気放電）イミュニティが向上します。

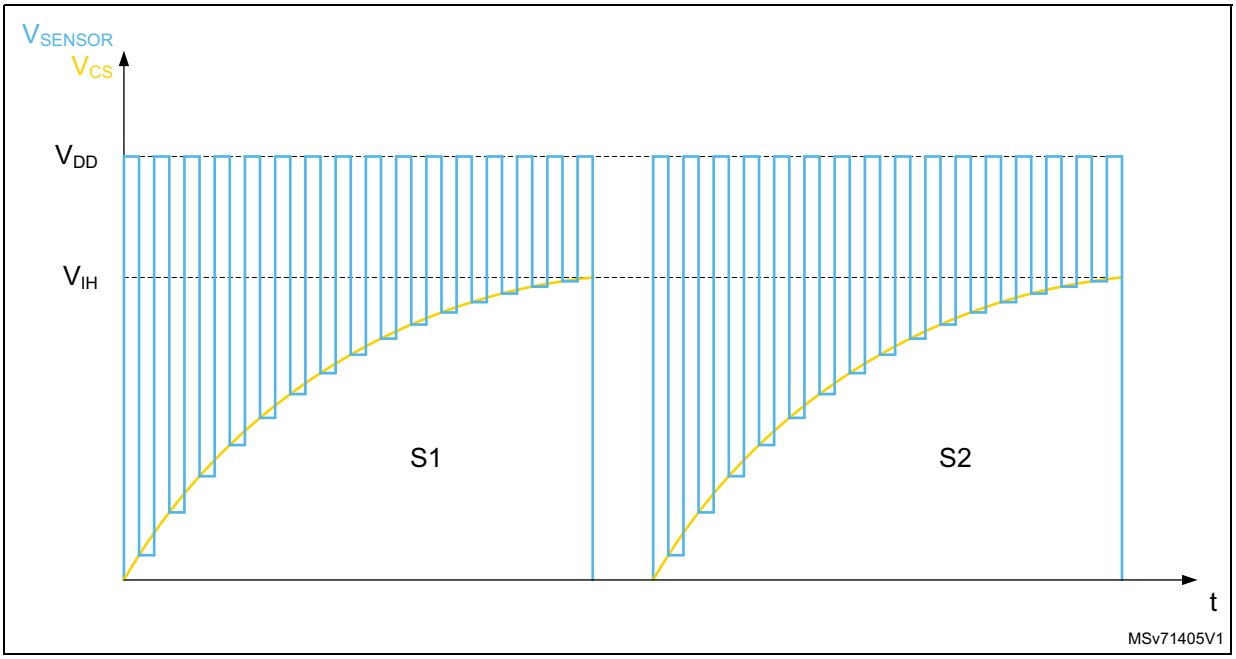
表452.取得シーケンスの概要

状態	Gx_IO1（チャネル）	Gx_IO2（サンプリング）	Gx_IO3（チャネル）	Gx_IO4（チャネル）	状態の説明
1	入力フローティング（アナログスイッチ閉）	出力オープンドレインロー（アナログスイッチ閉）	入力フローティング（アナログスイッチ閉）		すべてのC_XおよびC_Sを放電。
2	入力フローティング				デッドタイム
3	出力プッシュプルハイ	入力フローティング			C_{X1}を充電。
4	入力フローティング				デッドタイム
5	入力フローティング（アナログスイッチ閉）		入力フローティング		C_{X1}からC_Sへの電荷移動
6	入力フローティング				デッドタイム
7	入力フローティング				C_Sの電圧を測定。

注： **Gx_IOyのxはアナログI/Oグループの番号、yは選択したグループ内でのGPIOの番号を表します。**

サンプリングコンデンサC_Sの電圧の経時変化を以下に示します（V_{SENSOR}およびV_{CS}の定義については、図446を参照）。

図447. サンプリングコンデンサの電圧の変化



47.3.3 リセットおよびクロック

TSCクロックソースはAHBクロック（HCLK）です。2個のプログラム可能なプリスケーラを使用して、パルス発生回路およびスペクトル拡散内部クロックを生成します。

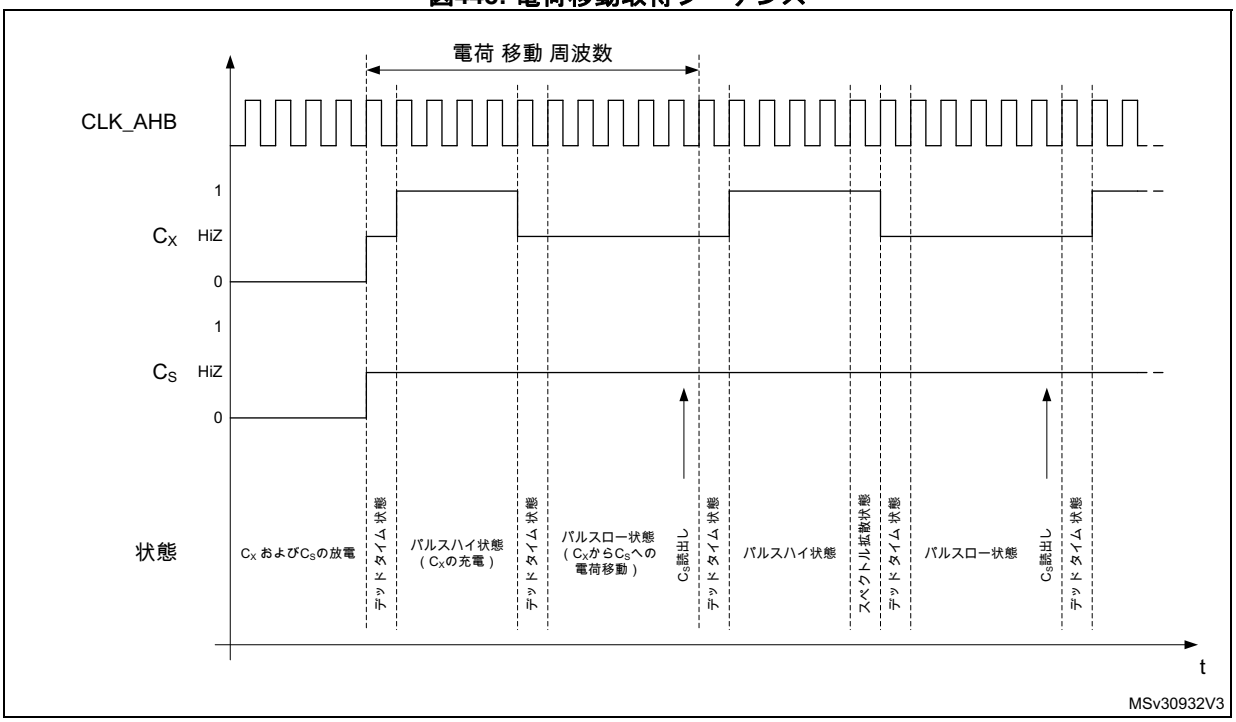
- パルス発生回路クロック（PGCLK）は、TSC_CRレジスタのPGPSC[2:0] ビットを使用して定義されます。
- スペクトル拡散クロック（SSCLK）は、TSC_CRレジスタのSSPSCビットを使用して定義されます。

リセットおよびクロックコントローラ（RCC）は、タッチセンシングコントローラのクロックを有効にし、このペリフェラルをリセットするための専用ビットを提供します。詳細については、セクション 11：リセットおよびクロック制御（RCC）を参照してください。

47.3.4 電荷移動取得シーケンス

電荷移動取得シーケンスの例を 図448に示します。

図448. 電荷移動取得シーケンス



より高い柔軟性を得るために、電荷移動周波数は完全に設定可能です。パルスのハイ状態（C_Xの充電）およびパルスのロー状態（C_X からC_Sへの電荷の移動）のどちらも、継続時間はTSC_CRレジスタのCTPH[3:0] ビットおよびCTPL[3:0] ビットを使用して定義することができます。パルスのハイ状態およびロー状態の継続時間の標準範囲は、500ns～2µsです。電極の静電容量を正確に測定できるようにするには、C_X が常にフル充電となるように、パルスのハイ状態の継続時間を設定する必要があります。

電荷移動取得シーケンスを最適化するため、パルスのハイ状態とロー状態の間にデッドタイムが挿入されます。デッドタイムとは、サンプリングコンデンサのI/Oとチャネルの I/Oの両方が入力フローティング状態にある時間のことです。この状態の継続時間はHCLKの 2 周期です。

スペクトル拡散機能が有効であれば、パルスのハイ状態の最後にSSCLKクロックの周期の変数が追加されます。

サンプリングコンデンサのI/Oの読出しは、C_S の電圧が所定の閾値に達したかどうかを確認するために、パルスのロー状態の最後に行われます。

注： **次のTSC制御レジスタ設定は禁止されています。**

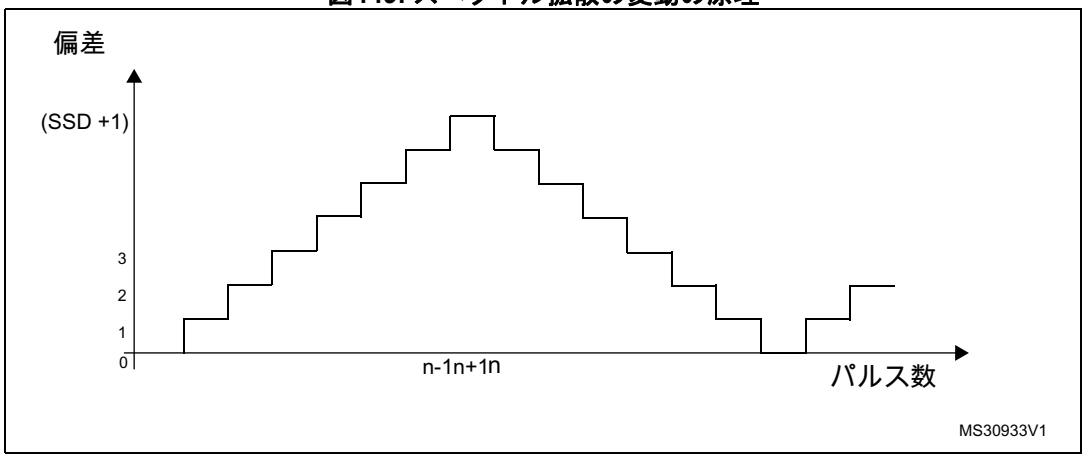
- **ビットPGPSCを“000”に、ビットCTPLを“0000”にセットすること**
- **ビットPGPSCを“000”に、ビットCTPLを“0001”にセットすること**
- **ビットPGPSCを“001”に、ビットCTPLを“0000”にセットすること**

47.3.5 スペクトル拡散機能

スペクトル拡散機能により、電荷移動周波数の変動が生成されます。これは、ノイズの多い環境において電荷移動取得の安定性を向上させ、さらに誘導放出を低減するために行われます。周波数の最大変動幅は、公称電荷移動周期の10～50% です。たとえば、公称電荷移動周波数が250kHz（4µs）の場合、典型的なスペクトル拡散偏差は10%（400ns）であり、この値から最小電荷移動周波数（227kHz以下）が導き出されます。

実際には、スペクトル拡散とは、以下に示す原理を用いて、SSCLK周期の変数をパルスのハイ状態に追加することを言います。

図449. スペクトル拡散の変動の原理



次の表に、異なるHCLK 設定での最大周波数偏差を示します。

表453.スペクトル拡散偏差とAHBクロック周波数

f_{HCLK}	スペクトル拡散の刻み	スペクトル拡散最大偏差
80MHz	12.5ns	3205.1ns
160MHz	6.25ns	1602.6ns

スペクトル拡散機能は、TSC_CRレジスタのSSEビットを使用して有効／無効にできます。周波数偏差は、TSC_CRレジスタのSSPSCおよびSSD[6:0] ビットを介して、デバイスHCLKクロック周波数および選択された電荷移動周波数を収容できるように設定することも可能です。

47.3.6 最大カウントエラー

最大カウントエラーは、静電容量検出チャネルの障害により取得時間が長くなるのを防ぎます。アナログI/Oグループのカウンタに対して最大カウント値を指定するのがこのエラーです。この最大カウント値はTSC_CRレジスタのMCV[2:0] ビットを使用して指定されます。取得グループのカウンタがこの最大値に到達するとすぐに、処理中の取得は停止し、取得完了（EOAFビット）および最大カウントエラー（MCEFビット）の両方のフラグがセットされます。対応する取得完了（EOAIEビット）や最大カウントエラー（MCEIEビット）の割込みイネーブル･ビットがセットされている場合は、割込みも生成することができます。

47.3.7 サンプリングコンデンサI/Oおよび チャネルI/Oのモード選択

タッチセンシングコントローラによるGPIOの制御を可能にするには、標準のGPIOレジスタおよびGPIOxAFRレジスタを介して、対応するオルタネート機能を有効にする必要があります。

TSCによって制御されるGPIOモードは、TSC_IOSCR およびTSC_IOCCR レジスタを使用して定義されます。

処理中の取得がない場合､タッチセンシングコントローラによって制御されるすべてのI/Oはデフォルト状態です。取得の処理中は、未使用I/O（サンプリングコンデンサI/Oとしても、チャネルI/Oとしても定義されない）のみがデフォルト状態です。TSC_CRレジスタのIODEFビットは、デフォルト状態にあるI/Oの設定を定義します。次の表にモードに応じたI/Oの設定を示します。

表454.モードおよびIODEFビットの値に応じたI/Oの状態

IODEFビット	取得 ステータス	未使用I/Oモード	チャネルI/Oモード	サンプリング コンデンサI/Oモード
0 （出力プッシュプルロー）	なし	出力プッシュプル ロー	出力プッシュプル ロー	出力プッシュプル ロー
0 （出力プッシュプルロー）	進行中	出力プッシュプル ロー	-	-
1 （入力フローティング）	なし	入力フローティング	入力フローティング	入力フローティング
1 （入力フローティング）	進行中	入力フローティング	-	-

未使用I/Oモード

未使用I/Oは、TSCペリフェラルによって制御されるGPIOに対応しますが、電極I/Oとしても、サンプリングコンデンサI/Oとしても定義されません。

サンプリングコンデンサI/Oモード

TSCペリフェラルによるサンプリングコンデンサI/Oの制御を可能にするには、まず対応するGPIOをオルタネート出力オープンドレインモードにセットし、次にTSC_IOSCR レジスタ の対応するGx_IOyビットをセットする必要があります。

1つのアナログI/Oグループでは、一度に1つのサンプリングコンデンサのみを有効にしてください。

チャネルI/Oモード

TSCペリフェラルによるチャネルI/Oの制御を可能にするには、まず対応するGPIOをオルタネート出力プッシュプルモードにセットし、次にTSC_IOCCR レジスタ の対応するGx_IOyビットをセットする必要があります。

より高い等価電極表面が要求される、あるいは取得プロセスを高速化する近接検出では、同じアナログI/Oグループに属するいくつかのチャネルを有効にし、同時に取得することが可能です。

注： **取得フェーズの間、およびTSCペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCRまたはTSC_IOCCRビットがセットされたらすぐに、タッチセンシングコントローラによって、対応するGPIOアナログスイッチが自動的に制御されます。**

47.3.8 取得モード

タッチセンシングコントローラには2種類の取得モードがあります。

- 通常の取得モード : TSC_CRレジスタのSTARTビットがセットされるとすぐに取得を開始します。
- 同期取得モード： TSC_CRレジスタのSTARTビットをセットすることで取得が有効になりますが、取得の開始は、立ち下がり／立ち上がりエッジおよびSYNC入力ピンのハイレベルが検出されたときのみです。このモードは、CPU負荷を追加することなく、静電容量検出チャネルの取得を外部信号に同期させる場合に役立ちます。

TSC_IOGCSRレジスタのGxEビットで、どのI/Oグループを有効（対応するカウンタがカウントされる状態）にするかを指定します。無効なアナログI/OグループのC_S電圧は監視されず、このグループは取得完了フラグのトリガに加わりません。ただし、無効なアナログI/Oグループにいくつかのチャネルが含まれる場合、それらはパルスを出力します。

有効なアナログI/OグループのC_S電圧が所定の閾値に達すると、TSC_IOGCSRレジスタの対応するGxSビットがセットされます。すべての有効なアナログI/Oグループの取得が完了した（すべての有効なアナログI/Oグループのすべてのビットがセットされた）時点で、TSC_ISRレジスタのEOAFフラグがセットされます。TSC_IERレジスタのEOAIEビットがセットされている場合、割込みリクエストが生成されます。

最大カウントエラーが検出された場合、処理中の取得は停止し、TSC_ISRレジスタのEOAFフラグとMCEFフラグの両方がセットされます。対応するビット（TSCIERレジスタのEOAIEおよびMCEIEビット）がセットされていれば、両方のイベントに対して割込みリクエストを生成することができます。最大カウントエラーが検出された場合、有効なアナログI/Oグループの残りのGxSビットはセットされませんので注意してください。

割込みフラグをクリアするには、TSC_ICRレジスタの対応するEOAICビットとMCEICビットをセットする必要があります。

アナログI/Oグループのカウンタは、新しい取得が開始された時点でクリアされます。カウンタは、取得完了と同時に、対応するチャネルで生成された電荷移動サイクルの数に更新されます。

47.3.9 I/Oヒステリシスおよびアナログスイッチの制御

より高い柔軟性を提供するために、タッチセンシングコントローラにより、各Gx_IOyのシュミットトリガヒステリシスおよびアナログスイッチを制御できます。タッチセンシングコントローラが有効であると仮定すれば、I/O制御モードの状態（標準のGPIOレジスタまたはその他のペリフェラルによって制御されている）にかかわらず、この制御は実行可能です。この制御は、異なる取得シーケンスを実行したり、他の目的のために実行する際に役立つことがあります。

システムのイミュニティを向上させる目的で、TSC_IOHCR レジスタの対応するGx_IOyビットをリセットすることによって、TSCによって制御されるGPIOのシュミットトリガヒステリシスを無効にする必要があります。

47.4 TSC低電力モード

表455.低電力モードがTSCに与える影響

モード	説明
SLEEP	影響はありません。ペリフェラル割込みによって、デバイスはSLEEPモードを終了します。
STOP	ペリフェラルレジスタの内容は保たれる。
STANDBY	パワーダウン状態。ペリフェラルは、STANDBYモードを終了した後に再初期化する必要がある。

47.5 TSC割込み

表456.割込み制御ビット

割込みイベント	イネーブル制御ビット	イベントフラグ	フラッグビットのクリア	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
取得完了	EOAIE	EOAIF	EOAIC	あり	なし	なし
最大カウントエラー	MCEIE	MCEIF	MCEIC	あり	なし	なし

47.6 TSC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスすることができます。

47.6.1 TSC制御レジスタ（TSC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTPH[3:0]				CTPL[3:0]				SSD[6:0]							SSE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSPSC	PGPSC[2:0]			Res.	Res.	Res.	Res.	MCV[2:0]			IODEF	SYNC POL	AM	START	TSCE
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 **CTPH[3:0]**：電荷移動パルスハイ

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、電荷移動パルスのハイ状態（C_Xの充電）の継続時間を定義します。

0000：1x t_{PGCLK}
0001：2x t_{PGCLK}
...
1111：16x t_{PGCLK}

注：　取得処理中はこれらのビットを変更しないでください。

ビット27:24 **CTPL[3:0]**：電荷移動パルスロー

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、電荷移動パルスのロー状態（C_X からC_Sへの電荷の移動）の継続時間を定義します。

0000：1x t_{PGCLK}
0001：2x t_{PGCLK}
...
1111：16x t_{PGCLK}

注：　取得処理中はこれらのビットを変更しないでください。

注：　一部の設定は禁止されています。詳細については、セクション 47.3.4：電荷移動取得シーケンスを参照してください。

ビット23:17 **SSD[6:0]**：スペクトル拡散偏差

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、SSCLKクロックの周期の変数を電荷移動パルスのハイ状態に追加する際のスペクトル拡散偏差を定義します。

0000000：1x t_{SSCLK}
0000001：2x t_{SSCLK}
...
1111111：128x t_{SSCLK}

注：　取得処理中はこれらのビットを変更しないでください。

ビット16 **SSE**：スペクトル拡散イネーブル

このビットは、スペクトル拡散機能を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：スペクトル拡散機能は無効です。

1：スペクトル拡散機能は有効です。

注： 取得処理中はこのビットを変更しないでください。

ビット15 **SSPSC**：スペクトル拡散プリスケーラ

このビットは、ソフトウェアによってセット／クリアされます。このビットは、スペクトル拡散クロック（SSCLK）の生成に使用されるAHBクロック分周器を選択します。

0：f_{HCLK}

1：f_{HCLK} /2

注： 取得処理中はこのビットを変更しないでください。

ビット14:12 **PGPSC[2:0]**：パルス発生回路プリスケーラ

これらのビットはソフトウェアによってセット／クリアされます。これらのビットは、パルス発生回路クロック（PGCLK）の生成に使用されるAHBクロック分周器を選択します。

000：f_{HCLK}

001：f_{HCLK} /2

010：f_{HCLK} /4

011：f_{HCLK} /8

100：f_{HCLK} /16

101：f_{HCLK} /32

110：f_{HCLK} /64

111：f_{HCLK} /128

注： 取得処理中はこれらのビットを変更しないでください。

注： 一部の設定は禁止されています。詳細については、セクション 47.3.4：電荷移動取得シーケンスを参照してください。

ビット11:8 予約済みであり、リセット値に保持する必要があります。

ビット7:5 **MCV[2:0]**：最大カウント値

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、最大カウントエラーが生成される前に生成され得る電荷移動パルスの最大数を定義します。

000:255

001：511

010：1023

011：2047

100：4095

101：8191

110：16383

111：予約済み

注： 取得処理中はこれらのビットを変更しないでください。

ビット4 **IODEF**：I/Oデフォルトモード

このビットは、ソフトウェアによってセット／クリアされます。このビットは、処理中の取得がない場合に、すべてのTSC I/Oの設定を定義します。処理中の取得がある場合、このビットは、すべての未使用I/O （サンプリングコンデンサI/Oやチャネル I/Oとして定義されていない）の設定を定義します。

0：I/Oは強制的に出力プッシュプルローにされます。

1：I/Oは入力フローティング状態です。

注： 取得処理中はこのビットを変更しないでください。

ビット3 **SYNCPOL**：同期ピンの極性

このビットは、同期入力ピンの極性を選択するために、ソフトウェアによってセット／クリアされます。

0：立ち下がりエッジのみ

1：立ち上がりエッジおよびハイレベル

ビット2 **AM**：取得モード

このビットは、取得モードを選択するために、ソフトウェアによってセット／クリアされます。

0：通常の取得モード（STARTビットがセットされた直後に取得を開始）

1：同期取得モード（STARTビットがセットされ、かつ選択された信号がSYNC入力ピンで検出された場合に、取得を開始）

注： 取得処理中はこのビットを変更しないでください。

ビット1 **START**：新しい取得を開始します。

このビットは、新しい取得を開始するために、ソフトウェアによってセットされます。このビットは、取得が完了した直後にハードウェアによって、または処理中の取得をキャンセルするためにソフトウェアによってクリアされます。

0：取得は開始されていません。

1：新しい取得を開始します。

ビット0 **TSCE**：タッチセンシングコントローライネーブル

このビットは、タッチセンシングコントローラを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：タッチセンシングコントローラは無効です。

1：タッチセンシングコントローラは有効です。

注： タッチセンシングコントローラが無効の場合、TSCレジスタの設定は無効です。

47.6.2 TSC割込み有効レジスタ（TSC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIE	EOAIE
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **MCEIE**：最大カウントエラー割込みイネーブル

このビットは、最大カウントエラーの割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：最大カウントエラー割込みは無効です。

1：最大カウントエラー割込みは有効です。

ビット0 **EOAIE**：取得完了割込みイネーブル

このビットは、取得完了割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：取得完了割込みは無効です。

1：取得完了割込みは有効です。

47.6.3 TSC割込みクリアレジスタ（TSC_ICR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIC	EOAIC
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **MCEIC**：最大カウントエラー割込みクリア

このビットは、最大カウントエラーフラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

0：影響なし

1：TSC_ISRレジスタの対応するMCEFをクリアします。

ビット0 **EOAIC**：取得完了割込みクリア

このビットは、取得完了フラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

0：影響なし

1：TSC_ISRレジスタの対応するEOAFをクリアします。

47.6.4 TSC割込みステータスレジスタ（TSC_ISR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEF	EOAF
														r	r

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **MCEF**：最大カウントエラーフラグ

このビットは、アナログI/Oグループのカウンタが指定された最大カウント値に達すると、ハードウェアによってセットされます。このビットは、ソフトウェアでTSC_ICRレジスタのMCEICビットに“1”を書き込むことによってクリアされます。

0：最大カウントエラー（MCE）は検出されていません。

1：最大カウントエラー（MCE）が検出されました。

ビット0 **EOAF**：取得完了フラグ

このビットは、すべての有効なグループの取得が完了した（すべての有効なアナログI/OグループのすべてのgxSビットがセットされた、または最大カウントエラーが検出された）時点で、ハードウェアによってセットされます。このビットは、ソフトウェアでTSC_ICRレジスタのEOAICビットに“1”を書き込むことによってクリアされます。

0：取得は処理中または開始されていません。

1：取得は完了しました。

47.6.5 TSC I/Oヒステリシス制御レジスタ（TSC_IOHCR）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOyシュミットトリガヒステリシスモード

これらのビットは、Gx_IOyシュミットトリガヒステリシスを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Gx_IOyシュミットトリガヒステリシスは無効です。

1：Gx_IOyシュミットトリガヒステリシスは有効です。

注： これらのビットは、I/O制御モードの状態にかかわらず（標準のGPIOレジスタによって制御されていても）、I/Oシュミットトリガヒステリシスを制御します。

47.6.6 TSC I/Oアナログスイッチ制御レジスタ（TSC_IOASCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOyアナログスイッチイネーブル

これらのビットは、Gx_IOyアナログスイッチを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Gx_IOyアナログスイッチは無効（開）です。

1：Gx_IOyアナログスイッチは有効（閉）です。

注： これらのビットは、I/O制御モードの状態にかかわらず（標準のGPIOレジスタによって制御されていても）、I/Oアナログスイッチを制御します。

47.6.7 TSC I/Oサンプリング制御レジスタ（TSC_IOSCR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOyサンプリングモード

これらのビットは、Gx_IOyをサンプリングコンデンサI/Oとして設定するために、ソフトウェアによってセット／クリアされます。1つのアナログI/Oグループでは、1つのI/Oだけをサンプリングコンデンサとして定義してください。

0：Gx_IOyは未使用です。

1：Gx_IOyはサンプリングコンデンサとして使用されます。

注： 取得処理中はこれらのビットを変更しないでください。

取得フェーズの間、およびTSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCRビットがセットされたらすぐに、タッチセンシングコントローラによって、対応するGPIOアナログスイッチが自動的に制御されます。

47.6.8 TSC I/Oチャネル制御レジスタ（TSC_IOCCR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOyチャネルモード

これらのビットは、Gx_IOyをチャネルI/Oとして設定するために、ソフトウェアによってセット／クリアされます。

0：Gx_IOyは未使用です。

1：Gx_IOyはチャネルとして使用されます。

注： 取得処理中はこれらのビットを変更しないでください。

取得フェーズの間、およびTSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOCCRビットがセットされたらすぐに、タッチセンシングコントローラによって、対応するGPIOアナログスイッチが自動的に制御されます。

47.6.9 TSC I/Oグループ制御ステータスレジスタ（TSC_IOGCSR）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8S	G7S	G6S	G5S	G4S	G3S	G2S	G1S
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8E	G7E	G6E	G5E	G4E	G3E	G2E	G1E
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **GxS**：アナログI/Oグループxステータス

これらのビットは、対応する有効なアナログI/Oグループxで取得が完了した時点で、ハードウェアによってセットされます。これらのビットは、新しい取得が開始した時点で、ハードウェアによってクリアされます。

0：アナログI/Oグループxでの取得は、処理中または開始されていません。

1：アナログI/Oグループxでの取得は完了しました。

注： 最大カウントエラーが検出された場合、有効なアナログI/Oグループの残りのGxSビットはセットされません。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **GxE**：アナログI/Oグループxイネーブル

これらのビットは、対応するアナログI/Oグループxでの取得を有効（カウンタがカウントされる状態）にするために、ソフトウェアによってセット／クリアされます。

0：アナログI/Oグループxでの取得は無効です。

1：アナログI/Oグループxでの取得は有効です。

47.6.10 TSC I/Oグループxカウンタレジスタ（TSC_IOGxCR）

xはアナログI/Oグループの番号を表します。

アドレスオフセット：0x30 + 0x04 * x（x = 1から8）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CNT[13:0]													
		r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:0 **CNT[13:0]**：カウンタ値

これらのビットは、アナログI/Oグループxの取得を完了させる（C_S の電圧が閾値に達する）ために、そこで生成された電荷移動サイクルの数を表します。

47.6.11 TSCレジスタマップ

表457.TSCレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	TSC_CR	CTPH[3:0]				CTPL[3:0]				SSD[6:0]							SSE	SSPSC	PGPSC[2:0]			Res.	Res.	Res.	Res.	MCV [2:0]			IODEF	SYNCPOL	AM	START	TSCE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x0004	TSC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIE	EOAIE
	リセット値																															0	0
0x0008	TSC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIC	EOAIC
	リセット値																															0	0
0x000C	TSC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEF	EOAF
	リセット値																															0	0
0x0010	TSC_IOHCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0014	予約済み																																
0x0018	TSC_IOASCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	予約済み																																
0x0020	TSC_IOSCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	予約済み																																
0x0028	TSC_IOCCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	予約済み																																
0x0030	TSC_IOGCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8S	G7S	G6S	G5S	G4S	G3S	G2S	G1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8E	G7E	G6E	G5E	G4E	G3E	G2E	G1E
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x0034	TSC_IOG1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0038	TSC_IOG2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	TSC_IOG3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表457.TSCレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0040	TSC_IOG4CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	TSC_IOG5CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	TSC_IOG6CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	TSC_IOG7CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050	TSC_IOG8CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

48 真性乱数発生器（RNG）

48.1 概要

RNGは、完全エントロピー出力を32bitサンプルとしてアプリケーションに提供する真性乱数発生器です。ライブエントロピーソース（アナログ）と内部conditioning component(条件付けコンポーネント)から構成されています。

RNGはNIST SP 800-90B準拠のエントロピーソースで、非決定論的乱数発生器（NDRBG）を構築するために使用できます。

RNG真性乱数発生器は、NIST SP800-90Bに従って事前認定されています。また、ドイツBSI文書の統計的検定AIS-31（T0 ~ T8）規格に従って試験を行いました。

48.2 RNGの主な機能

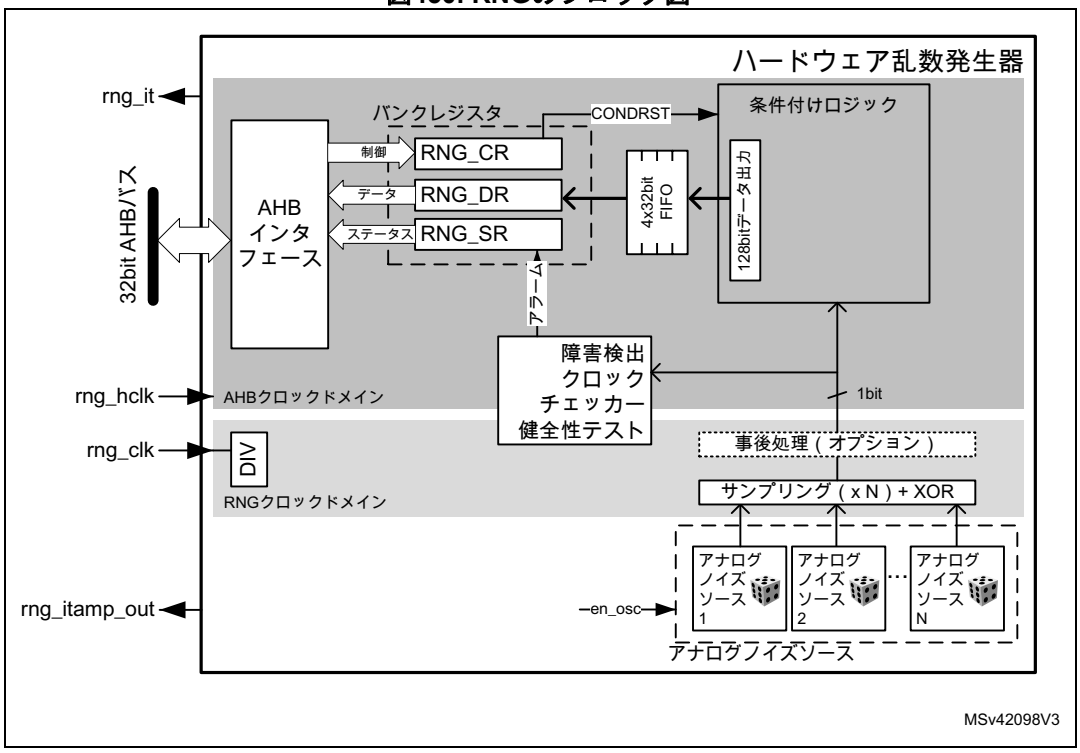
- RNGは、NIST SP800-90B承認条件付けステージで条件付けされたアナログエントロピーソースによって作成された32bitの真の乱数を供給します。
- 非決定論的乱数発生器（NDRBG）を構築するためのエントロピーソースとして使用できます。
- NIST構成では、f_{AHB} が$f_{threshold}$ 未満の場合、412 AHB クロックサイクル（それ以外の場合は256 RNGクロックサイクル）ごとに 4つの 32bitのランダムサンプルを生成します。
- RNGには、起動時テストおよびNIST SP800-90B承認の継続的健全性テスト（繰返し回数テストおよび適応比率テスト）が内蔵されており、特定のエラー管理と関連付けられています。
- 本機能を無効にして消費電力を低減できます。また、自動低電力モードで有効にできます（デフォルト設定）。
- 32bit･ワードのシングル･アクセスでのみアクセス可能なAMBA® AHBスレーブペリフェラルが搭載されています（それ以外はAHBバスエラーが発生し、書込みアクセスは無視されます）。

48.3 RNGの機能説明

48.3.1 RNGのブロック図

図450にRNGブロック図を示します。

図450. RNGのブロック図



48.3.2 RNG内部信号

表458では、RNGをより良く理解するために内部の信号のリストを記載しております。STM32（パッド上）の入出力の信号ではありません。

表458.RNG内部入力／出力信号

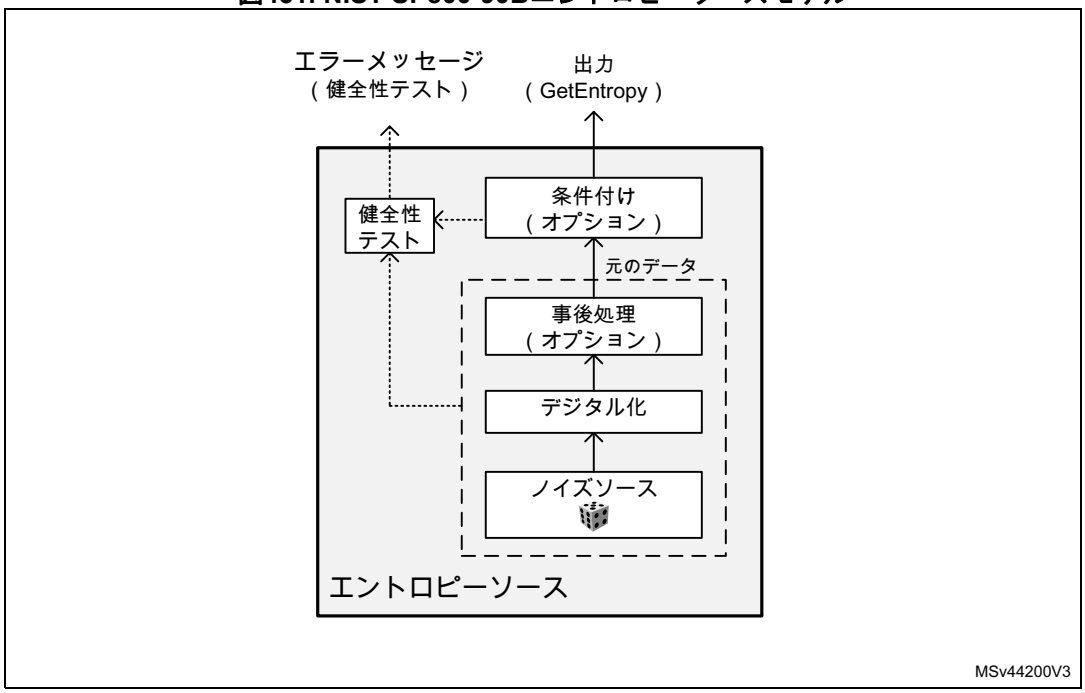
信号名	信号タイプ	説明
rng_it	デジタル出力	RNGグローバル割込みリクエスト
rng_hclk	デジタル入力	AHBクロック
rng_clk	デジタル入力	RNG専用クロック、rng_hclkと非同期
rng_itamp_out	デジタル出力	予期しないハードウェアフォールトが発生したとき、TAMPへのRNG内部タンパイベント信号（XORされます）。この信号がトリガされると、RNGはランダムサンプルの配信を停止し、再び使用可能になるにはリセットと新しい初期化が必要です。

48.3.3 乱数の生成

真性乱数発生器（RNG）は、決定論的な間隔でAHBインタフェースを介して真にランダムなデータを伝送します。

その境界内で、RNGには 図451に示されている必要なすべてのNISTコンポーネントが統合されています。これらのコンポーネントは、アナログノイズソース、デジタル化ステージ、条件付けアルゴリズム、健全性監視ブロック、およびエントロピーソースとの相互作用に使用される2つのインタフェース、すなわちGetEntropyとHealthTestです。

図451. NIST SP800-90Bエントロピーソースモデル



上に図示されたコンポーネントの詳細を以下に示します。

ノイズソース

ノイズソースは、非決定論的なエントロピーを生成する構成要素のノイズソースで、これは出力の不確実性に関連したものです(デジタル化されたノイズソースやソースデータ)このノイズソースは1ビットのサンプルを提供します。以下の要素で構成されます。

- それぞれ排他的論理和がとられたフリーランニングのリングオシレータの出力 3 つに基づく複数のアナログノイズソース（x6）。セクション 48.3.8 : RNGの低消費電力時の取り扱いに記載しているとおり、電力低減のためにそれらのアナログオシレータは無効化できます。
- 単一のアナログ出力に向けた、すべてのノイズソースの排他的論理和。
- 専用クロック入力（rng_clk（内蔵分周回路あり））によってクロック供給されたこの出力のサンプリングステージ。1bitの元データ出力を伝送します。

このノイズソースサンプリングは、AHBインタフェースクロック周波数（rng_hclk）から独立しています。ソフトウェアは統合された分周回路を使用してサンプリング周波数を減少させることができます。

注： **セクション 48.6 : RNGエントロピーソース検証では、推奨されるRNGクロック周波数および関連する分周値が示されています。**

後処理

NIST構成では、サンプリングされたノイズソースに後処理は適用されません。非NIST構成B（セクション48.6.2 で定義されている）では、正規化バイアス解除が適用されます。すなわち、ビットの半分はサンプルノイズソースから取得し、あとの半分は反転したサンプルノイズソースから取得します。

条件付け

RNGの条件付けコンポーネントは、結果として得られる固定長ビット列出力（128ビット）のエントロピー率を増加させる決定論的機能です。NIST SP800-90Bの目標は、出力での完全なエントロピーです（128bit）。

2つの乱数生成の間に必要な時間、RNGを初期化してから最初のサンプルを利用できるまでに必要な時間については、セクション 48.5：RNG処理時間に記載しています。

出力バッファ

データ出力バッファは、条件付けコンポーネントから出力された32bitワードを4つまで格納できます。RNG_DRレジスタを通じて出力FIFOから4ワード読み出されると、128bit条件付け出力レジスタの内容は出力FIFOにプッシュされ、新たな条件付けが自動的に始まります。に指定された数のクロックサイクルの後に、新たな4ワードが条件付け出力レジスタに追加されます。セクション 48.5：RNG処理時間

乱数がRNG_DRレジスタを介して使用できるときは必ず、DRDYフラグが0から1に変わります。このフラグは、RNG_DRレジスタから 4ワード読み出した後、出力バッファが空になるまでハイのままになります。

注： **割込みが有効の場合、このデータレディフラグが0から1に遷移するときに割込みが生成されます。その後、上記で説明したように割込みはRNGによって自動的にクリアされます。**

健全性チェック

このコンポーネントはエントロピーソース全体（そのノイズソースとともに）を開始し、その後期待通りに実行し、高い確立と信頼性ですばやく不具合の取り込みを保証することを確実にします。

RNGは、NIST SP800-90Bに準拠して、以下のような健全性チェック機能を実行します。記述されている閾値は、レジスタRNG_HTCRの推奨値に対応します（セクション48.6.2の設定Aを参照）。

1. リセットしてからRNGを初めてエントロピーソースとして使用するまでに実行される起動時健全性テスト:
   - 繰返し回数テストで、ノイズソースが 38 ビット以上連続して一定値（0または1）を提供した場合にエラーフラグを立てます。
   - 連続する1024bitのウィンドウで実行される適応比率テスト：RNGは、ノイズソース出力の最初のビットが 686 回以上繰り返されていないことを検証します。
   - 既知解テストで、条件付けステージを検証します。
2. ノイズソースの出力で無制限に実行する継続的健全性テスト：
   - 起動時テストで実行されるものと同様の繰返し回数テスト。
   - 起動時テストで実行されるものと同様の適応比率テスト。
3. ベンダ固有の継続テスト
   - 遷移回数テスト。ノイズソースが2bitパターン（01または10）を32回以上連続して提供した場合にエラーフラグを立てます。
   - リアルタイムの「遅すぎる」サンプリングクロック検出。1 RNGクロックサイクル（分周前）が 32 で分周されたAHBクロックサイクルより小さい場合にエラーフラグを立てます。
4. デジタル化ノイズソース（元のデータ）のオン･デマンド･テスト
   - エントロピーソースを再起動し、起動時テストを再実行することでサポートされます（セクション 48.3.4：RNG初期化 のソフトウェアリセットシーケンスを参照）。その他の必要に応じたテスト（ソフトウェアベース）はサポートされません。

エラーコンディションが検出された時に、RNG_SRレジスタのCECSおよびSECSステータスビットに表示されます。詳細はセクション 48.3.7：エラー管理を参照してください。

注： **エラーが検出された場合、割込みを生成することができます。**

上記の健全性テストの閾値は、RNG_HTCRレジスタの値を変更することで変更されます。詳細については、セクション 48.6：RNGエントロピーソース検証を参照してください。

48.3.4 RNG初期化

図452 にRNGの簡易ステートマシンを図示します。

RNGを有効化した後（RNG_CRのRNGEN = 1）、以下の一連のイベントが発生します。

1. アナログノイズソースが有効になると、デフォルトではRNGはアナログ出力のサンプリングを開始して128bitの条件付けシフトレジスタに入力するまで16 RNGクロックサイクル待ちます。
2. 条件付けハードウェアは起動すると､元のデータサンプルで起動時動作テストと既知解テストを自動的にトリガします。
3. 起動時健全性テストが完了するとき、この間に、3つの128bitのノイズソースサンプルが使用されます。
4. 条件付けステージの内部入力データバッファに128bitが再入力され、RNG設定（NISTまたは非NIST）によって定義された多数の条件付けが実行されます。その後、出力バッファには後処理の結果が入力されます。
5. 出力バッファはRNGの使用方法に応じて自動的に再入力されます。

関連する初期化時間については、セクション 48.5：RNG処理時間を参照してください。

図452. RNG初期化の概要

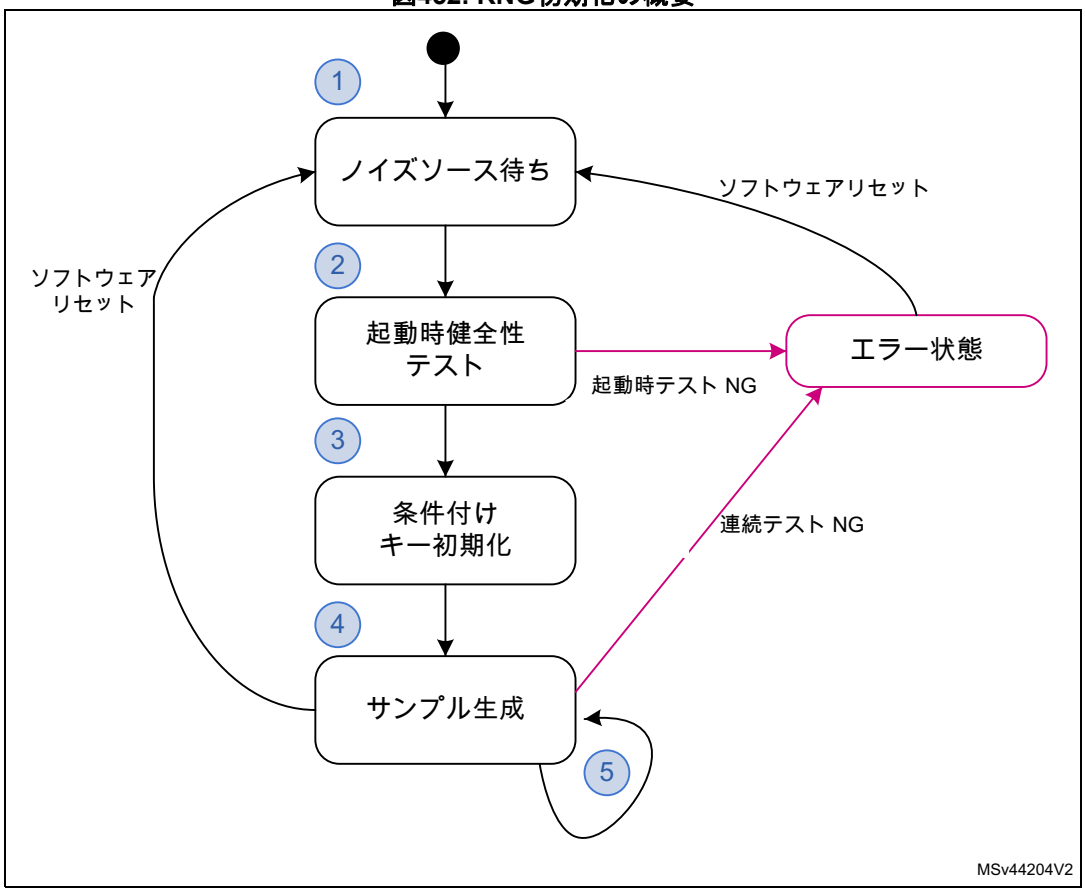


図452 では、以下によって実施される考えられるソフトウェアリセットのシーケンスも説明しています。

1. 同じRNG設定および新しいCLKDIVで、RNG_CRレジスタにビットRNGEN = 0およびCONDRST = 1を書き込みます（必要に応じて）。
2. 次に、RNG_CRレジスタにRNGEN = 1およびCONDRST = 0を書き込みます。
3. 初期化が完了した後、乱数が準備できるまで待ちます。

注： **RNGペリフェラルがRCC（ハードウェアリセット）を介してリセットされると、RNGレジスタで最適なランダム性のためのRNG設定が失われます。CONFIGLOCKをセットしたソフトウェアリセットでは、RNG設定が保持されます。**

48.3.5 RNG動作

通常動作

割込みを使用してRNGを作動させるには、次の手順を推奨します。

1. セクション 48.6 : RNGエントロピーソース検証を参照して、特定のRNG設定がアプリケーションに必要かどうかを確認します。
   - 必要であれば、RNG_CRレジスタに正しいRNG設定とともにビットCONDRST = 1を書き込みます。次に、RNG_CRレジスタにビットCONDRST = 0、割込み有効化ビットIE = 1、RNG有効化ビットRNGEN = 1を、さらに書き込みます。
   - 必要でない場合、RNG_CRレジスタに割込み有効化ビットIE = 1、RNG有効化ビットRNGEN1 = 1を書き込みます。
2. 乱数の準備ができたとき、またはエラーが発生したとき、割込みが生成されるようになります。したがって、割込みがあるたびに次のことをチェックします。
   - 発生したエラーはありません。RNG_SRレジスタでSEISおよびCEISビットを0にセットする必要があります。
   - 乱数が準備できています。RNG_SRレジスタでDRDYビットを1にセットする必要があります。
   - 上記2つの条件に当てはまる場合、最大で連続4回まで RNG_DRレジスタの内容が読み出せるようになります。条件付け出力バッファに有効なデータがある場合には、アプリケーションによってさらに4ワード読み出すことができます（この場合にはDRDYビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DRレジスタを読み出さないでください。エラーが発生した場合には、セクション48.3.7に記載されたリカバリシーケンスを使用する必要があります。

ポーリングモードでRNGを作動させるには、次の手順を推奨します。

1. セクション 48.6 : RNGエントロピーソース検証を参照して、特定のRNG設定がアプリケーションに必要かどうかを確認します。
   - 必要であれば、RNG_CRレジスタに正しいRNG設定とともにビットCONDRST = 1を書き込みます。次に、RNG_CRレジスタにビットCONDRST = 0、RNG有効化ビットRNGEN = 1を、さらに書き込みます。
   - 必要でない場合、RNG_CRレジスタのRNGENビットを1にセットして、RNGを有効にするだけです。
2. RNG_SRレジスタを読み出し、次を確認します。
   - 発生したエラーはありません（SEISおよびCEISビットは0にセットされていなければなりません）。
   - 乱数が準備できています（DRDYビットは1にセットされていなければなりません）。
3. 上記の条件に当てはまる場合、最大で連続4回まで RNG_DRレジスタの内容を読み出します。条件付け出力バッファに有効なデータがある場合には、アプリケーションによってさらに4 ワード読み出すことができます（この場合にはDRDYビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DRレジスタを読み出さないでください。エラーが発生した場合には、セクション48.3.7に記載されたリカバリシーケンスを使用する必要があります。

注： **データが準備できていない場合（DRDY = 0）、RNG_DRは0を返します。**
RNG_DRがゼロではないことを常に確認することをお勧めします。ゼロの場合、RNG_SRポーリングとRND_DR出力読出しの間にシードエラーが発生した（まれなイベント）ことを意味するからです。

乱数生成周期がアプリケーションに対して懸念となる場合、およびNISTの準拠が必須ではない場合、セクション 48.6 : RNGエントロピーソース検証に記述されているRNGの設定“B”を使用して、より高速なRNG設定を選択できます。乱数生成速度の増加に関する概要は、セクション 48.5 : RNG処理時間に示しています。

低消費電力動作

消費電力がアプリケーションに対して懸念となる場合、セクション 48.3.8 : RNGの低消費電力時の取り扱いに示すように低消費電力にする取り扱いを使用できます。

ソフトウェア後処理

AIS-31またはNIST SP800-90Bの承認を満たすために期待される特定のソフトウェア後処理／条件付けはありません。

内蔵されている健全性チェック機能の説明はセクション 48.3.3 : 乱数の生成を参照してください。

48.3.6 RNGクロック供給

RNGは、AHBバスクロックと専用RNGクロックの2種類の異なるクロックで動作します。

AHBクロックは、AHBバンクレジスタおよび条件付けコンポーネントにクロック供給するために使用されます。RNGクロックは、プログラム可能な分周回路（RNG_CRレジスタのCLKDIVビットフィールドを参照）と組合せてノイズソースサンプリングに使用されます。推奨されるクロック設定は、セクション 48.6 : RNGエントロピーソース検証で詳しく説明されています。

注： **RNG_CRレジスタのCEDビットが0にセットされている場合、内部分周器の前の RNGクロック周波数は 32 で分周されたAHBクロック周波数より高くする必要があります。そうしないと、クロックチェッカーが必ずクロックエラーのフラグ（RNG_SRレジスタのCECS = 1）を立てます。**

詳細（AHBおよびRNGクロックドメイン）については、セクション 48.3.1 : RNGのブロック図を参照してください。

48.3.7 エラー管理

このセクションに詳細を示しているとおり、乱数生成と並行して、健全性チェックブロックにより、正しいノイズソース動作とRNGソースクロックの周波数が検証されます。関連するエラー状態も記載されています。

クロックエラー検出

クロックエラー検出が有効なとき（CED = 0）、RNGクロック周波数が低すぎる場合、RNGは CEIS と CECS の両方のビットを1にセットして、クロックエラーが発生したことを示します。この場合、アプリケーションによってRNGクロックが正しく設定されていること（セクション 48.3.6 : RNGクロック供給を参照）をチェックして、CEISビットの割込みフラグをクリアする必要があります。CECSビットは、クロック条件が正常なときに自動的にクリアされます。

注： **クロックエラーは、生成された乱数には影響しないため、アプリケーションはRNG_DRレジスタを読み出すことができます。**

CEISは、RNGによってCECSが1にセットされているときのみセットされます。

ノイズソースエラー検出

ノイズソース（またはシード）エラーが発生すると、RNGは乱数生成を停止し、SEIS および SECS ビットを両方1にセットして、シードエラーが発生したことを示します。このときRNG_DRレジスタに値が用意されていても、エントロピーが十分ではない可能性があるので使用しないようにします。

シードエラーからの完全なリカバリを行うには、次のシーケンスを使用する必要があります。

1. CONDRSTに1、そして0を書き込んでソフトウェアリセットします（詳細はビットフィールドの説明を参照）。SECSがセットされている場合のみ、このステップ が必要です。実際、SEISがセットされ、SECSがクリアされていると、RNGが自動的にリセット（自動リセット）を実行し

たことを意味します。この場合、アプリケーションはSEISビット割込みフラグをクリアする必要があります。

2. 手順 1 で SECS がセットされていた（自動リセットではない）場合、RNG_CR レジスタでCONDRSTがクリアされるまで待ってから、RNG_SRレジスタでSEISがクリアされていることを確認します。そうでない場合、RNG_SRレジスタのSEISビットをクリアするだけです。
3. 手順1でSECSがセットされていた（自動リセットではない）場合、SECSがRNGによってクリアされるまで待ちます。乱数生成が正常に戻ります。

注： **シードエラーの後、SECSがクリアされると、RNGは乱数の生成を再開します。**

アプリケーションがRNG_CRレジスタのARDISビットを設定すると、自動リセットは無効になります。手順1でCONDRSTを使用する必要があります。

RNGタンパエラー

RNGによって予期しないエラーが検出されると、TAMPペリフェラルで内部タンパイベントがトリガされ、RNGは乱数データの提供を停止します。

このイベントが発生すると、セキュアアプリケーションが集中リセット管理またはグローバルSoCリセットのいずれかを使用してRNGペリフェラルをリセットする必要があります。その後、再びRNGを適切に初期化する必要があります。

48.3.8 RNGの低消費電力時の取り扱い

消費電力が懸念される場合、RNG_CRレジスタのRNGENビットを0にセットしてDRDYビットを1にセットするとすぐに、RNGを無効化できます。RNGEN = 0であっても後処理ロジックと出力バッファは動作を続けていますので、ソフトウェアは次の機能を使用できます。

- 出力バッファに有効なワードが存在する場合には、RNG_DRレジスタから乱数をもう4個読み出すことができます。
- 条件付け出力内部レジスタに有効なビットが存在する場合には、RNG_DR レジスタから乱数をもう4個読み出すことができます。そうではない場合には、期待される新しいノイズソースビット閾値（NISTモードでは128bit）に達して、条件付けが完了するまでアプリケーションでRNGを再度有効にする必要があります。
  期待される条件付けの回数に達した場合のみ（NISTC = 0の場合は2回）、新たな4つの乱数ワードを使用できるようになります。全体的な時間については、セクション 48.5： 1893ページのRNG処理時間を参照してください。

RNGを無効化する場合、すべてのアナログシード発生回路を無効にします。その消費電力は、データシートの電気的特性セクションに示されています。また、RNGクロックによってクロックの供給を受けるロジックはすべてクロック停止にします。この方法では、RNG初期化時間のために、RNG_DRレジスタでランダムサンプルが利用可能となるまでのレイテンシが増加することに注意してください。

初期化中に（すなわちDRDYビットが初めて立ち上がるよりも十分前に）RNGブロックが無効化された場合、RNGENビットが1にセットされると、初期化シーケンスは停止したところから再開されます（アプリケーションがRNG_CRレジスタのCONDRSTビットを使用して条件付けロジックをリセットしていない限り）。

アプリケーションがRNGクロックを停止する必要がある場合は、RNGENビットをクリアしてからRCCを使用してRNGカーネルクロックを停止するまでの間に2 RNGカーネルクロックサイクル待つことをお勧めします。

また、RNGが非アクティブになる電力モードにアプリケーションが入る必要がある場合は、RNGENビットをクリアしてからPWRを使用して低消費電力モードに入るまでに2 RNGカーネルクロックサイクル待つことをお勧めします。

上記の2つのケースでは、RNGアナログ発振器がアクティブなままのときの予期せぬ電力消費を避けるために、アプリケーションはRNG_CRレジスタのビット13をセットすることができます。このビットをセットすると、RNGENビットがセットされている（RNGがアクティブになっている）間、わずかな電力消費しか追加されません。

注： **RNG が非アクティブ化される（つまり、保留されるか、または使用できない）電力モードは、PWR セクションを参照してください。**

48.4 RNG割込み

RNGでは、割込みは次のイベントによって生成できます。

- データレディフラグ
- シードエラー、セクション 48.3.7：エラー管理を参照
- クロックエラー、セクション 48.3.7：エラー管理を参照

表459 に示すように、専用割込み有効化制御ビットが使用できます。

表459.RNG割込みリクエスト

項目（割込みの略称）	割込みイベント	イベント･フラグ	イネーブル制御ビット	割込みのクリア方法
RNG	データレディフラグ	DRDY	IE	なし（自動）
	シードエラーフラグ	SEIS	IE	SEISに0を書き込むか、CONDRSTに1を書き込んでから0を書き込みます。
	クロックエラーフラグ	CEIS	IE	CEISに0を書き込みます。

RNG_CRレジスタのマスクビットまたは全体的な割込み制御ビットIEを変更することにより、上記の割込みソースを個別に有効にしたり無効にしたりすることができます。個別の割込みソースのステータスはRNG_SRレジスタから読み出すことができます。

注： **割込みは、RNGが有効化されている場合にのみ生成されます。**

48.5 RNG処理時間

表460で説明されている推奨設定Aでは、4つの32bitデータの2セットの間の時間は次のいずれかです。

- $f_{AHB} < f_{threshold}$ の場合、206 x N AHBサイクル（条件付けステージが制限要因）、または
- $f_{AHB} \geq f_{threshold}$ の場合、128 x N RNGサイクル（ノイズソースステージが制限要因）

ここで、$f_{threshold} = 1.6 \times f_{RNG}$ の場合、たとえばf_{RNG}= 48 MHzであれば、77 MHzです。Nの値は 2 です。

注： **CLKDIVが0ではない場合、f_{RNG} は内部分周比を考慮する必要があります。**

設定Bが選択された場合、性能数値は次のようになります。

- $f_{AHB} < f_{threshold}$ の場合、206 AHBサイクル、または
- $f_{AHB} \geq f_{threshold}$ の場合、32 RNGサイクル

ここで、$f_{threshold} = 6.5 \times f_{RNG}$。

48.6 RNGエントロピーソース検証

48.6.1 概要

RNGから使用できるエントロピーの量を評価するために、STMicroelectronicsはドイツのBSI AIS-31統計テスト（T0 ~ T8）およびNIST SP800-90Bテストスイートを使用してペリフェラルをテストしました。結果は要求に応じて提供でき、顧客側でテストを再現できます。

48.6.2 検証条件

STMicroelectronicsは、以下の条件下でRNG真性乱数発生器をテストしました。

- RNGクロックrng_clk= 48MHz
- RNGの設定については、表 460 : RNGの設定で説明しています。設定AのみNIST SP800-90Bを認証できます。

表460.RNGの設定

RNG設定	RNG_CRビット						ループ数（N）	RNG_HTCRレジスタ	RNG_NCSR register
	NISTCビット	RNG_CONFIG1 [5:0]	CLKDIV [3:0]	RNG_CONFIG2 [2:0]	RNG_CONFIG3 [3:0]	CEDビット			
A	0	0x0F	0x6(1)	0x0(2)	0xD	0	2	0x0000 9AAE(3)	0x0000 0492
B	1	0x18	0x0	0x0	0x0	0	1	0x0000 A6BA	0x0003 FFFF

1. NIST認証の場合、ノイズソースのサンプリングは 750kHz 以下でなければなりません。したがって、RNGクロックが48MHzと異なる場合は、CLKDIVのこの値を調整する必要があります。詳細については、セクション48.7.1 のCLKDIVビットフィールドの説明を参照してください。
2. RNGの電力消費が重要な場合は、0x1値を推奨します。詳細については、セクション 48.3.8 : RNGの低消費電力時の取り扱いの最後を参照してください。
3. 繰返しテストでは 38 に、適応テストでは 686 に対応します。詳細については、1888ページの健全性チェックを参照してください。

48.6.3 データ収集

統計テストを行うためには、サンプルを元データレベルでのエントロピーソースからだけではなく、エントロピーソースの出力からも収集する必要があります。データ収集と統計テストスイートの実行の詳細については、www.st.com から入手できるアプリケーションノート「STM32 microcontrollers random number generation validation using NIST statistical test suite」（AN4230）を参照してください。

製品で上記サンプルを取得する必要がある場合は、STMicroelectronicsまでご連絡ください。

48.7 RNGレジスタ

RNGは、制御レジスタ、データレジスタ、ステータスレジスタと連動します。

48.7.1 RNG制御レジスタ（RNG_CR）

アドレスオフセット：0x000

リセット値：0x0080 0D00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CONFIGLOCK	CONDRST	Res.	Res.	Res.	Res.	RNG_CONFIG1 [5:0]						CLKDIV[3:0]			
rs	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RNG_CONFIG2 [2:0]			NISTC	RNG_CONFIG3 [3:0]				ARDIS	Res.	CED	Res.	IE	RNGEN	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw		rw	rw		

ビット31 **CONFIGLOCK**：RNG設定ロック

0：RNG_HTCRおよびRNG_CR設定ビット [29:4] への書込みが許可されます。

1：RNG_HTCR およびRNG_CR設定ビット [29:4] への書込みは、次のRNGリセットまで無視されます。

このビットフィールドは1回だけセットできます。このビットがセットされると、RNGがリセットされた場合のみ0にリセットできます。

ビット30 **CONDRST**：条件付きソフトリセット

1を書き込んでから0を書き込んで条件付けロジックをリセットし、すべてのFIFOをクリアして、RNG_SRがクリアされた状態で新しいRNG初期化処理を開始します。RNG_CR およびRNG_HTCR レジスタは、CONDRSTによっては変更されません。

設定ビット [29:4] をセットするのと同じアクセスで、このビットを1にセットする必要があります。言い換えると、CONDRSTビットが1にセットされるときには、ビット [29:4] にも正しい設定を書き込む必要があります。

CONDRSTがソフトウェアによって0にセットされると、その値はリセット処理が完了したときに0になります。これには約2 AHBクロックサイクルと2 RNGクロックサイクルを必要とします。

ビット29:26 予約済みであり、リセット値に保持する必要があります。

ビット25:20 **RNG_CONFIG1[5:0]**：RNG設定1

RNG設定に予約済みです（ビットフィールド1）。セクション 48.6：RNGエントロピーソース検証に記載された推奨される値を使用して初期化する必要があります。

RNG_CONFIG1のあらゆるビットへの書込みは、CONFIGLOCKが0になっている間に、同じアクセスでCONDRSTビットが1にセットされる場合のみ考慮されます。CONFIGLOCK = 1の場合、このビットへの書込みは無視されます。

ビット19:16 **CLKDIV[3:0]**：クロック分周回路ファクタ

この値は、受信RNGクロックで動作するプログラム可能な内部分周回路（1～16）の設定に使用されます。これらのビットに書き込めるのは、コアが無効（RNGEN = 0）のときだけです。

0x0：分周回路以降の内部RNGクロックは、受信RNGクロックと類似しています。

0x1：内部RNGクロックごとに2 RNGクロックサイクルです。

0x2：内部RNGクロックごとに2^2（= 4）RNGクロックサイクルです。

...

0xF：内部クロックごとに2^{15} RNGクロックサイクルです（例：48MHz受信すると、RNGクロックは1.5kHz内部RNGクロックになります）。

これらのビットへの書込みは、CONFIGLOCKが0になっている間に、同じアクセスでCONDRSTビットが1にセットされる場合のみ考慮されます。CONFIGLOCK = 1の場合、このビットへの書込みは無視されます。

ビット15:13 **RNG_CONFIG2[2:0]**：RNG設定2

RNG設定に予約済みです（ビットフィールド2）。RNGの電力消費が重要な場合は、ビット13をセットすることができます。セクション 48.3.8：RNGの低消費電力時の取り扱いを参照してください。詳細は、RNG_CONFIG1bitフィールドの説明を参照してください。

ビット12 **NISTC**：NISTカスタム

0：NIST準拠RNGのハードウェアデフォルト値。この設定では、128bit出力ごとの2回の条件付けループが実行され、256bitのノイズソースが使用されます。

1：NIST準拠RNGのカスタム値。提案される設定については、セクション 48.6：RNGエントロピーソース検証を参照してください。

このビットへの書込みは、CONFIGLOCKが0のままで同じアクセスでCONDRSTビットが1にセットされる場合のみ考慮されます。CONFIGLOCK = 1の場合、このビットへの書込みは無視されます。

ビット11:8 **RNG_CONFIG3[3:0]**：RNG設定3

RNG設定に予約済みです（ビットフィールド3）。詳細は、RNG_CONFIG1bitフィールドの説明を参照してください。

このレジスタのNISTCビットがクリアされている場合、RNG_CONFIG3のビットフィールドの値はRNGによって無視されます。

ビット7 **ARDIS**：オートリセットは無効です。

0:ノイズソースエラーが発生すると、RNGはオートリセットを実行して、SECSビットをクリアします。

1：ノイズソースエラーが発生したとき、乱数生成を再開始するには、アプリケーションはCONDRSTに1を書き込んでから0を書き込むことによって、RNGをリセットする必要があります。

オートリセットが有効なとき、アプリケーションは、やはり、ノイズソースエラー後にSEISビットをクリアする必要があります。

このビットへの書込みは、CONFIGLOCKが0のままで同じアクセスでCONDRSTビットが1にセットされる場合のみ考慮されます。CONFIGLOCK = 1の場合、このビットへの書込みは無視されます。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CED**：クロックエラー検出

0：クロックエラー検出は有効です。

1：クロックエラー検出は無効です。

RNGが有効になっている場合、動作中にクロックエラー検出を有効化／無効化できません。CEDを有効化／無効化するには、RNGを無効化する必要があります。

このビットへの書込みは、CONFIGLOCKが0になっている間に、同じアクセスでCONDRSTビットが1にセットされる場合のみ考慮されます。CONFIGLOCK = 1の場合、このビットへの書込みは無視されます。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 **IE**：割込み有効

0：RNG割込みは無効です。

1：RNG割込みは有効です。RNG_SRレジスタでDRDY = 1 、SEIS = 1またはCEIS = 1になると、割込みは直ちにペンディングとなります。

ビット2 **RNGEN**：真性乱数発生器イネーブル

0：真性乱数発生器は無効です。アナログノイズソースの電力がオフになり、RNGクロックによって供給されるロジックがゲートされます。

1：真性乱数発生器は有効です。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

48.7.2 RNGステータスレジスタ（RNG_SR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
									rc_w0	rc_w0			r	r	r

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **SEIS**：シードエラー割込みステータス

このビットは、SECSとして同じ時間でセットされます。0を書き込むことによってクリアされます（CONDRSTが使用されていない場合）。1を書き込んでも、ビットの値は変化しません。

0：障害シーケンスは検出されませんでした。

1：1つ以上の障害シーケンスが検出されました。詳細については、SECS ビットの説明を参照してください。

RNG_CRレジスタでIE = 1である場合、割込みが保留となります。

ビット5 **CEIS**：クロックエラー割込みステータス

このビットは、CECSとして同じ時間でセットされます。0を書き込むことによってクリアされます。1を書き込んでも、ビットの値は変化しません。

0：正しいRNGクロックです（$f_{RNGCLK} > f_{HCLK}/32$）。

1：内部分周回路より前の RNGクロックが遅すぎると検出されました（$f_{RNGCLK} < f_{HCLK}/32$）。

RNG_CRレジスタでIE = 1である場合、割込みが保留となります。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SECS**：シードエラーの現在のステータス

0：障害のあるシーケンスは現在のところ検出されていません。SEISビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。

1：次の障害シーケンスのうち1つ以上が検出されました。

- ランタイム時の繰返し回数テストに失敗しました（ノイズソースが24bit以上連続して一定値0または1を提供したか、32回以上連続して2bitパターン01または10を提供しました）。
- ノイズソースで起動時または継続の適応比率テストに失敗しました。
- 起動の後処理／条件付け正常性チェックに失敗しました。

ビット1 **CECS**：クロックエラーの現在のステータス

0：正しいRNGクロックです（$f_{RNGCLK} > f_{HCLK}/32$）。CEISビットがセットされている場合、これは遅いクロックが検出されたが、回復していることを意味します。

1：遅すぎるRNGクロックです（$f_{RNGCLK} < f_{HCLK}/32$）。

注： CECSビットは、RNG_CRレジスタのCEDビットが0にセットされたときにのみ有効です。

ビット0 **DRDY**：データレディ

0：RNG_DRレジスタがまだ有効でなく、乱数データは利用できません。

1：RNG_DRレジスタに有効な乱数データが入っています。

出力バッファが空になると（RNG_DRレジスタの読出し後）、新しい乱数値が生成されるまで、このビットは0に戻ります。

注： ペリフェラルが無効化されている場合（RNG_CRレジスタのRNGEN = 0）に、DRDYビットが立ち上がることがあります。

RNG_CRレジスタでIE= 1の場合、DRDY = 1のときに割込みが生成されます。

48.7.3 RNGデータレジスタ（RNG_DR）

アドレスオフセット：0x008

リセット値：0x0000 0000

RNG_DRレジスタは読出し専用レジスタであり、読み出されると32bitの乱数値を返します。このレジスタの内容は、RNGEN = 0の場合でも、DRDY = 1で、値が0x0でないときに有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RNDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RNDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RNDATA[31:0]**：ランダムデータ

DRDY = 1のときに有効な32bit乱数データです。DRDY = 0のとき、RNDATAの値は0です。

DRDYがセットされているとき、RNG_DRがゼロではないことを常に確認することをお勧めします。ゼロの場合、RNG_SRポーリングとRND_DR出力読出しの間にシードエラーが発生した（まれなイベント）ことを意味するからです。

48.7.4 RNG健全性テスト制御レジスタ（RNG_HTCR）

アドレスオフセット：0x010

リセット値：0x0000：72AC

RNG_HTCRへの書込みは、RNG_CRでCONDRSTビットがセットされ、CONFIGLOCKビットがクリアされている場合のみ考慮されます。CONFIGLOCK = 1の場合、このレジスタへの書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HTCFG[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTCFG[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **HTCFG[31:0]**：健全性テストの設定

この設定は 健全性テストを設定するためにRNGによって使用されます。推奨値については セクション 48.6：RNGエントロピーソース検証を参照してください。

注： 推奨値と異なる値が書き込まれた場合、このレジスタの読出しを含むRNGの動作は保証されません。

48.7.5 RNGレジスタマップ

表461.RNGレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RNG_CR	CONFIGLOCK	CONDRST	Res.	Res.	Res.	Res.	RNG_CONFIG1 [5:0]						CLKDIV [3:0]				RNG_ CONFIG2 [2:0]			NISTC	RNG_ CONFIG3 [3:0]				ARDIS	Res.	CED	Res.	IE	RNGEN	Res.	Res.
	リセット値	0	0					0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0		0		0	0		
0x004	RNG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
	リセット値																										0	0			0	0	0
0x008	RNG_DR	RNDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	RNG_HTCR	HTCFG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	1	0	1	0	1	1	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

49 AESハードウェアアクセラレータ（AES）

このセクションは、STM32U545/585/5Ax/5Gxデバイスにのみ適用されます。

49.1 概要

AESハードウェアアクセラレータ（AES）は、連邦情報処理規格公報（FIPS）197 (Federal information processing standards (FIPS) publication 197)によって規定されている高度暗号化標準（AES : Advanced Encryption Standard）に完全に準拠したアルゴリズムと実装を使用してデータを暗号化および復号化します。

ペリフェラルは、128または256bitのキーサイズの CTR、GCM、GMAC、CCM、ECB、およびCBC連鎖モードをサポートします。

AESは、32bitのシングルアクセスでのみアクセス可能なAMBA AHB スレーブペリフェラルです。その他のアクセスタイプは AHB エラーを生成し、32bit書込み以外では、レジスタの内容が破損する可能性があります。

ペリフェラルは、入力および出力データのDMAシングル転送をサポートします（2つのDMAチャネルが必要です）。

49.2 AES の主な特徴

- 2001年11月発効のNIST『高度暗号化標準（AES）、FIPS公報197』に準拠
- 128bitデータブロックの処理
- 128bitおよび256bitの暗号キー長に対応
- 次の複数の連鎖モードで暗号化と復号 :
  - 電子コードブック（ECB）モード
  - 暗号ブロック連鎖（CBC）モード
  - カウンタ（CTR）モード
  - ガロアカウンタモード（GCM）
  - ガロアメッセージ認証コード（GMAC）モード
  - CBC-MAC付きカウンタ（CCM）モード
- 128bit または 256bit キーをそれぞれ使用して 1 個の 128bit ブロックのデータを処理する場合はECBモードで 51 または 75 クロックサイクルの遅延
- ECB/CBC復号の最終ラウンドのキーの計算に使用する内蔵ラウンド･キー･スケジューラ
- 32bit･ワードのシングル･アクセスでのみアクセス可能なAMBA AHB スレーブペリフェラル
- 暗号化キーを格納するための256bit書込み専用レジスタ（32bitレジスタ8個）
- 初期化ベクタを格納するための128bitレジスタ（32bitレジスタ4個）
- データ入力および出力用の32bitバッファ
- 2つのチャネル（1チャネルは入力データ用、1チャネルは処理済みデータ用）を使用したシングル転送ダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御
- 1、8、16、32bitデータをサポートするデータスワッピングロジック
- AES がより優先順位の高い別メッセージを処理する必要がある場合は、ソフトウェアでメッセージをサスペンドしてから元のメッセージのレジュームが可能
- SAESで制御された、サイドチャネル攻撃に対する耐性を備えたSAES ペリフェラル（共有キーモード）とのハードウェアキーの共有

49.3 AES の実装

デバイスには、1つのAESと1つのSAESペリフェラルがあり、次の表に示すように実装されています。

表462. AES/SAESの機能

AES/SAESのモード／機能(1)	AES	SAES
ECB、CBC連鎖	X	X
CTR、CCM、GCM連鎖	X	-
AES 128bit ECB暗号化（サイクル単位）	51	528
DHUKおよびBHKキー選択	-	X
サイドチャネル攻撃耐性	-	X
SAESとAESの共有キー	X	

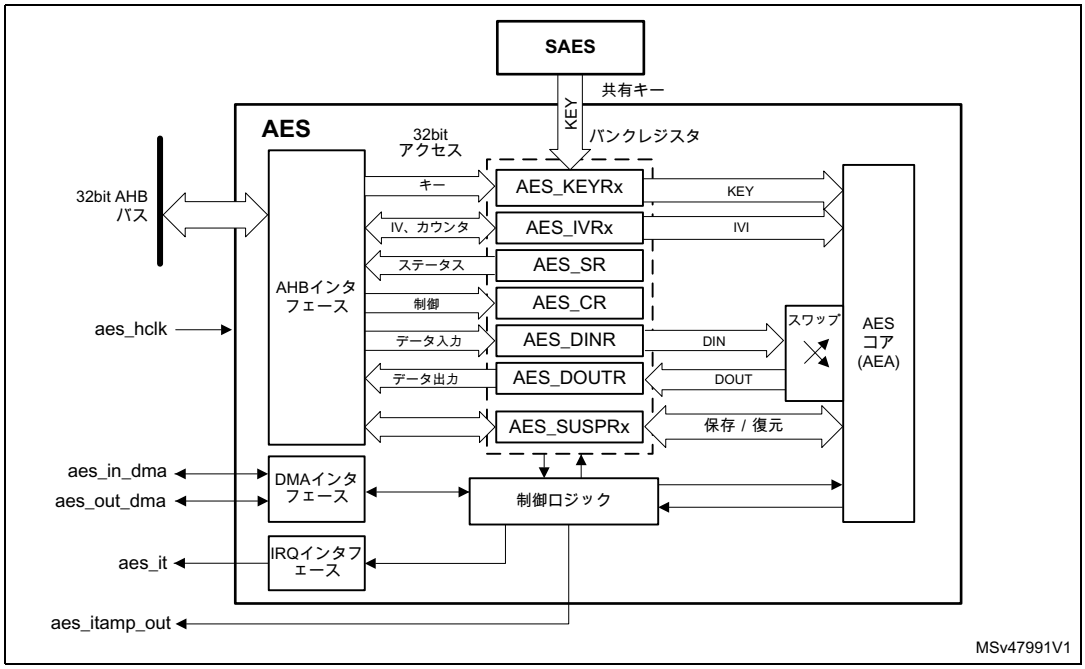
1. X : サポートされています。

49.4 AES 機能詳細

49.4.1 AES ブロック図

図453 に AES のブロック図を示します。

図453. AES ブロック図



49.4.2 AES 内部信号

表463には、AES ペリフェラルにインタフェース接続するユーザ関連内部信号を示しています。

表463.AES 内部入力／出力信号

信号名	信号タイプ	説明
aes_hclk	入力	AHB バスクロック
aes_it	出力	AES 割込みリクエスト
aes_in_dma	入力／出力	入力DMAシングル･リクエスト／確認応答
aes_out_dma	入力／出力	出力DMAシングル･リクエスト／確認応答
aes_itamp_out	出力	予期しないハードウェアフォールトが発生したときにトリガされる、TAMPへのタンパイベント信号（XORされます）。この信号がトリガされると、AES は自動的にキーレジスタをクリアします。AES を再び使用可能にするには、リセットする必要があります。

49.4.3 AES 暗号コア

概要

AES 暗号コアは、次のコンポーネントで構成されています。

- AESコア･アルゴリズム（AEA）
- バイナリガロアフィールド（GF2mul）の乗算器
- キー入力
- 初期化ベクタ（IV）入力
- 連鎖アルゴリズムロジック（XOR、フィードバック／カウンタ、マスク）

AES コアは128bitまたは256bitキー長を使用して128bitデータブロック（4ワード）で動作します。連鎖モード次第で、AES は0個または1個の128bit初期化ベクタIVを必要とします。

AES には、次の動作モードがあります。

- **モード1**：

  AES_KEYRxレジスタに格納されたキーを使用して平文を暗号化します。
- **モード2**：

  ECBまたはCBCの復号化キーを準備します。ECBまたはCBCの連鎖モードでモード3を選択する前に使用する必要があります。復号化のために準備されたキーは、AES_KEYRxレジスタに自動的に格納されます。これで、AES ペリフェラルをモード3に切り替えて、データ復号化を実行する準備ができます。
- **モード3**：

  AES_KEYRxレジスタに格納されたキーを使用して暗号文を復号化します。ECBおよびCBCの連鎖モードを選択した場合、モード2で事前にキーを準備する必要があります。

注： **モード2 は、ECBおよびCBCの復号化を実行する場合のみ使用します。**

動作モードは、AES_CRレジスタのMODE[1:0] ビットフィールドをプログラムして選択します。この操作は、AES ペリフェラルが無効である場合のみ実行できます。

特殊キー動作は、AES_CRレジスタのKMOD[1:0] ビットフィールドを使用して選択します。詳細については、セクション49.4.13を参照してください。

標準的データ処理

AES の標準的な使用方法をセクション 49.4.4：1908ページの暗号操作を行うための AES の手順に示します。

注： **IVIビットフィールドを除き、中間AEAステージの出力が暗号境界の外に漏れることは決してありません。**

連鎖モード

次の連鎖モードが AES でサポートされており、AES_CRレジスタのCHMOD[2:0] ビットフィールドで選択します。

- 電子コードブック（ECB）
- 暗号ブロック連鎖（CBC）
- カウンタ（CTR）
- ガロアカウンタモード（GCM）
- ガロアメッセージ認証コード（GMAC）
- CBC-MAC付きカウンタ（CCM）

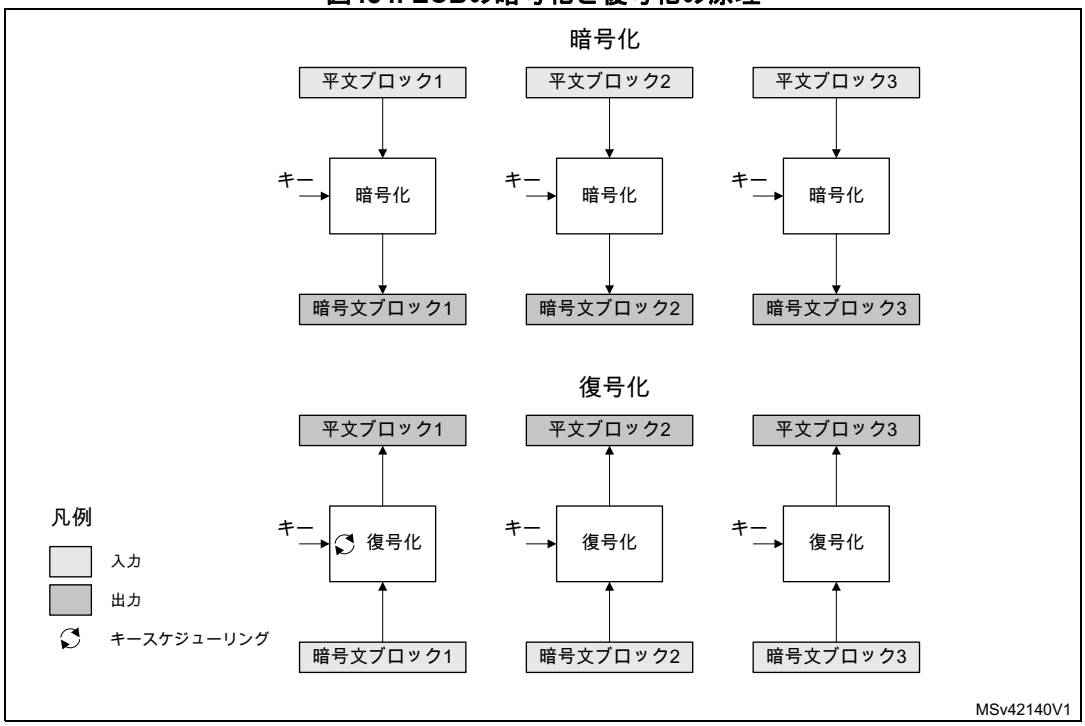
注： **連鎖モードは、AES が無効である（AES_CRレジスタのENビットがクリアされている）場合のみ変更できます。**

各 AES 連鎖モードの原理については、以降の項目に記載されています。

詳細については、セクション 49.4.8：AES 基本連鎖モード（ECB、CBC）以降の専用セクションに示されています。

電子コードブック（ECB）モード

図454. ECBの暗号化と復号化の原理

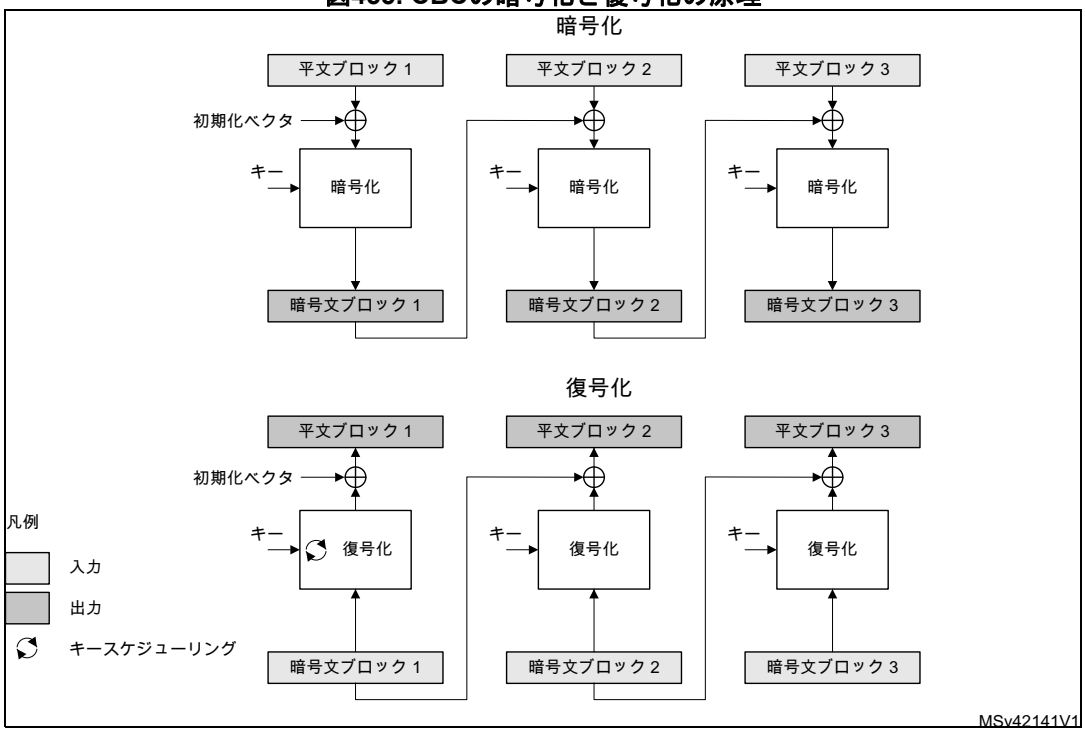


ECBは最も単純な動作モードです。連鎖操作も、特別な初期化ステージもありません。メッセージはブロックに分割され、各ブロックが個別に暗号化または復号化されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

暗号ブロック連鎖（CBC）モード

図455. CBCの暗号化と復号化の原理

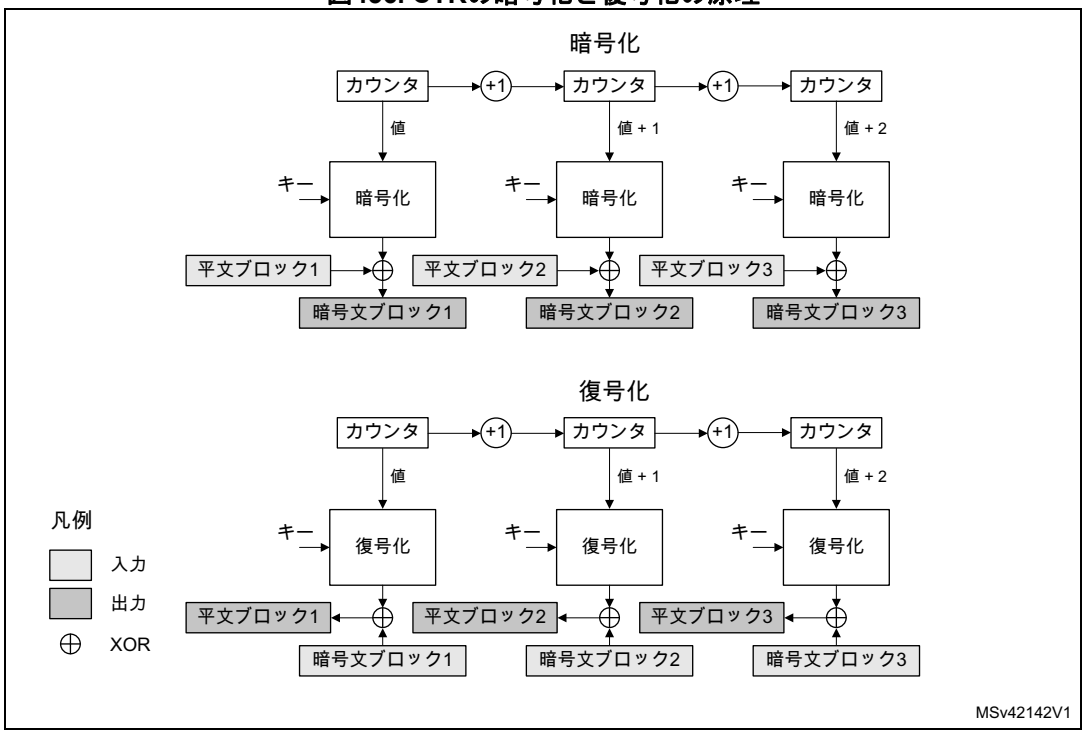


CBCモードでは、各ブロックの出力が次のブロックの入力で連鎖されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクタが使用されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

カウンタ（CTR）モード

図456. CTRの暗号化と復号化の原理

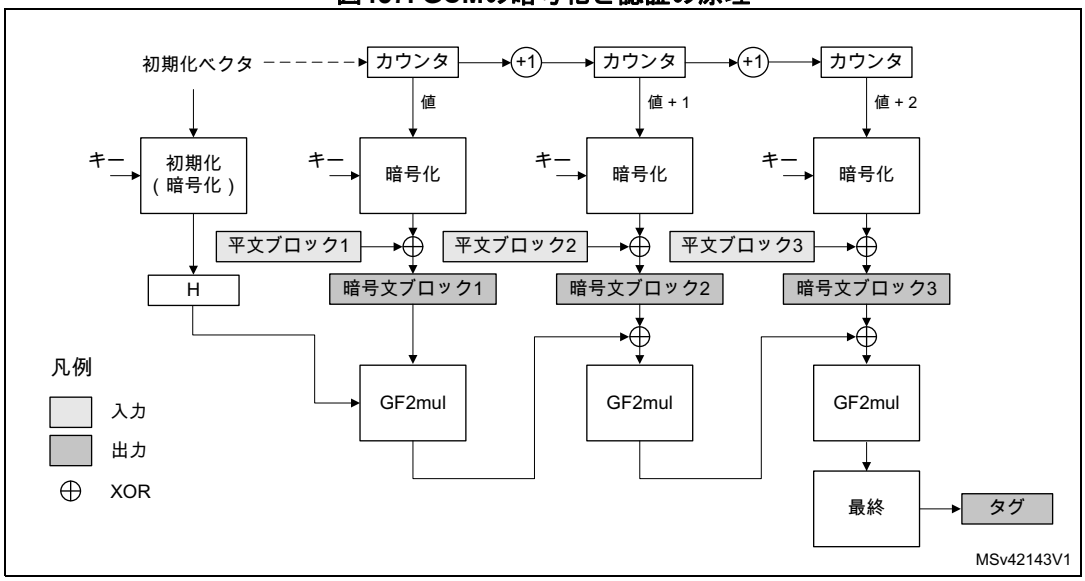


CTRモードでは、AESコアを使用してキーストリームを生成します。次に、このキーは、NIST 特別公報800-38A『ブロック暗号の推奨動作モード』（NIST Special Publication 800-38A,Recommendation for Block Cipher Modes of Operation）で指定されている暗号文を得るために、平文との排他的論理和をとられます。

注： **この連鎖スキームでは、キーストリームまたはカウンタブロックの生成に AES コアが暗号化モードで必ず使用されるため、ECBモードやCBCモードとは異なり、CTRの復号化にキースケジューリングは必要ありません。**

ガロア／カウンタモード（GCM）

図457. GCMの暗号化と認証の原理

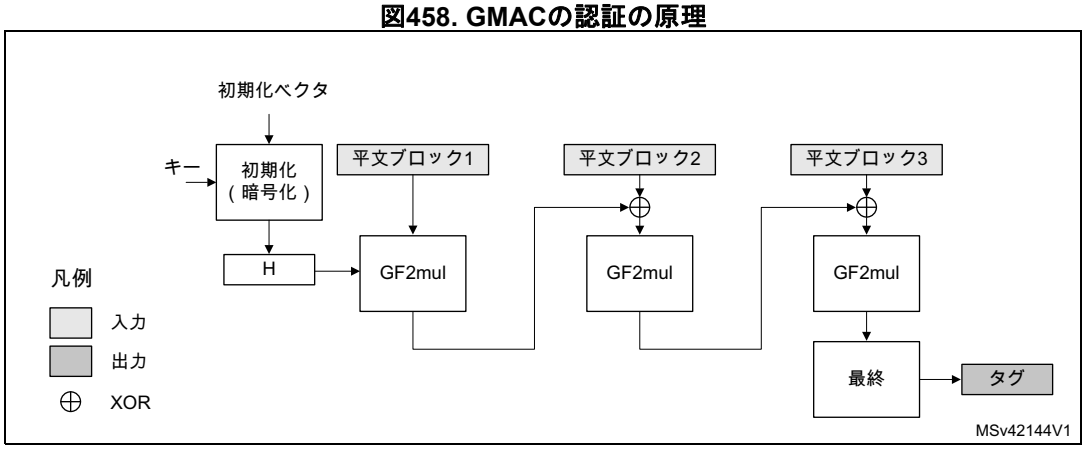


ガロア／カウンタモード（GCM）では、平文メッセージが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文とそのMAC（認証タグとも言います）が生成されます。これは、NIST特別公報800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』（NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation - Galois/Counter Mode (GCM) and GMAC）の中に定義されています。

GCMモードは、機密性のためにカウンタモードでの AES に基づいています。固定の有限体に乗算器を使用して、メッセージ認証コードを計算します。初期値ならびにメッセージの最後に特殊な128bitブロックを必要とします。

ガロアメッセージ認証コード（GMAC）の原理

図458. GMACの認証の原理

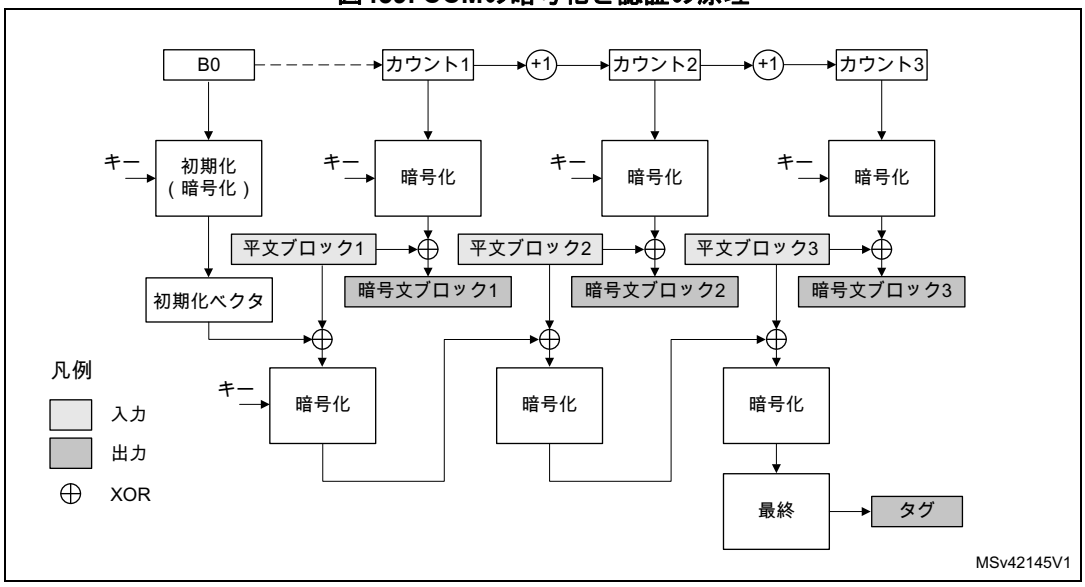


ガロアメッセージ認証コード（GMAC）を使用すると、メッセージの認証と、対応するメッセージ認証コード（MAC）の生成が可能となります。これは、NIST 特別公報800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』の中に定義されています。

平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみでペイロードなし）に適用されることを除けば、GMACはGCMと似ています。

CBC-MAC付きカウンタ（CCM）の原理

図459. CCMの暗号化と認証の原理



暗号ブロック連鎖-メッセージ認証コード付きカウンタ（CCM）モードでは、平文メッセージが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文と対応するMAC（タグとも言います）が生成されます。これは、NIST特別公報800-38C『ブロック暗号の推奨動作モード - 認証および機密性のためのCCMモード』（NIST in Special Publication 800-38C, Recommendation for Block Cipher Modes of Operation - The CCM Mode for Authentication and Confidentialityの中に説明されています。

CCMモードは、機密性のためにカウンタモードでの AES に基づいており、CBCを使用してメッセージ認証コードを計算します。初期値が必要です。

CCM連鎖モードは、GCMのように平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみでペイロードなし）に適用できます。CCMをこのように使用することはCMACとは呼ばれず（GCM/GMACとは異なります）、その使用方法はNISTによる推奨ではないことに注意してください。

49.4.4 暗号操作を行うための AES の手順

概要

標準的な暗号操作を以下に説明します。詳細については、セクション 49.4.8：AES 基本連鎖モード（ECB、CBC）以降のセクションに示されています。

AES の初期化

AES を初期化するには、まず AES_CRレジスタのENビットをクリアして無効化します。次に、任意の順番で次の手順を実行します。

- AES_CRレジスタのMODE[1:0] ビットフィールドをプログラムして、AES モードを設定します。
  - 暗号化の場合は、モード1（MODE[1:0] = 00）を選択します。
  - 復号化の場合は、ECBまたはCBCの連鎖モードを使用していない場合、モード3を選択します（MODE[1:0] = 10）。後者の場合、セクション 49.4.5：AES 復号化ラウンドキーの準備に記載されているとおり、暗号化キーの初期キー導出を実行する必要があります。
- AES_CRレジスタのCHMOD[2:0] ビットフィールドをプログラムして、連鎖モードを選択します。
- AES_CRレジスタのDATATYPE[1:0] ビットフィールドでデータ型（1、8、16、32bit）を設定します。
- 必要であれば（CBC またはCTR の連鎖モードの場合など）、初期化ベクタを AES_IVRxレジスタに書き込みます。
- AES_CRレジスタのKEYSIZEビットフィールドでキーサイズ（128bitまたは256bit）を設定します。この手順は、キーレジスタへの書込みを行う前に行う必要があります。
- 対称キーを AES_KEYRxレジスタ（キーサイズに応じて4個か8個のレジスタ）に書き込みます。

注： **AES は、KEYSIZEによって定義されたキー情報が AES_KEYRxにロードされている場合、AES_SRのKEYVALIDをセットします。**

データ追加

このセクションでは、KMOD[1:0] = 00のとき、処理するデータのサイズが128bitの倍数ではない場合に、処理するためにデータを追加するさまざまな方法について説明します。その他のKMOD[1:0] 値については、セクション49.4.13 を参照してください。

ECBまたはCBCモードについては、セクション 49.4.6：AES 暗号文借用およびデータパディング を参照してください。これらの場合の最終ブロックの管理は、このセクションで説明しているシーケンスより複雑になります。

ポーリングでのデータ追加

この方法では、フラグポーリングを使用して次のシーケンスでデータ追加を制御します。

1. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
2. ペイロードが完全に処理されるまで次のサブシーケンスを繰り返します。
   a) AES_DINRレジスタに4つの入力データワードを書き込みます。
   b) ステータスフラグCCFが AES_SRでセットされるまで待って、AES_DOUTRレジスタから4つのデータワードを読み出します。
   c) AES_ICR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
   d) 処理したばかりのデータブロックがメッセージの最後から2番目のブロックであり、処理する最終ブロックの有効データが128bitを下回る場合、最終ブロックの残り部分を0でパディングし、GCMのペイロード暗号化またはCCMのペイロード復号化の場合は、AES_CRレジスタのNPBLBビットフィールドで、AES が正しいタグを計算できるように有効ではないバイト数を指定します。

3. これは最終ブロックであるため、データの一部ではないデータを破棄してから、AES_CRレジスタのENビットをクリアして AES ペリフェラルを無効にします。

注： **ウェイトサイクルは AES_DINRレジスタに対する連続する2つの書込みの間に最大3つまで自動的に挿入され、AES プロセッサにキーを送信できます。**

NPBLBビットは、GCM、GMAC、CCMの連鎖モードのヘッダフェーズで使用されません。

割込みを使用したデータ追加

この方法は、AES ペリフェラルからの割込みを使用して次のシーケンスでデータ追加を制御します。

1. AES_IER レジスタのCCFIEビットをセットして、AES からの割込みを有効にします。
2. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
3. まず、AES_DINRレジスタに4つの入力データワードを書き込みます。
4. AES 割込みサービスルーチンでデータを処理します。割込み時には次の手順に従います。
   a) AES_DOUTRレジスタから4つの出力データワードを読み出します。
   b) AES_ICR レジスタの CCF ビットをセットして、CCFフラグ、保留中の割込みをクリアします。
   c) 処理したばかりのデータブロックがメッセージの最後から2番目のブロックであり、処理する最終ブロックの有効データが128bitを下回る場合、最終ブロックの残り部分を0でパディングし、GCMのペイロード暗号化またはCCMのペイロード復号化の場合は、AES_CRレジスタのNPBLBビットフィールドで、AES が正しいタグを計算できるように有効ではないバイト数を指定します。次に、4e) に進みます。
   d) 処理したばかりのデータブロックがメッセージの最終ブロックだった場合、データの一部ではないデータを破棄してから、AES_CRレジスタのENビットをクリアして AES ペリフェラルを無効にし、割込みサービスルーチンを中止します。
   e) AES_DINRレジスタに次の4つの入力データワードを書き込み、割込みサービスルーチンを中止します。

注： **AES は連続した読出し操作または書込み操作の間の遅延を許容するため、2つの AES 計算の間に別のペリフェラルからの割込みがある場合などに受け入れることができます。**

NPBLBビットは、GCM、GMAC、CCMの連鎖モードのヘッダフェーズで使用されません。

DMAを使用したデータ追加

この方法では、すべての転送と処理がDMAおよび AES によって管理されます。この方法を使用するには、次の手順に従います。

1. ブロックの残り部分を0でパディングして、最後の4ワードのデータブロックを準備します（処理するデータが完全に埋まっていない場合）。
2. 処理するデータをメモリから AES ペリフェラル入力へ、処理したデータを AES ペリフェラル出力からメモリに転送するようDMAコントローラを設定します（セクション 49.4.17：AES DMA インタフェース に記載）。転送完了時に割込みを生成するようDMAコントローラを設定します。GCMのペイロード暗号化またはCCMのペイロード復号化で0でパディングしている場合は、DMA転送に最後の4ワードブロックを**含まないでください**。代わりに、ポーリングでのデータ追加に示したシーケンスをこの最終ブロックに使用する必要があります。これは AES が正しいタグを計算できるように、NPBLBビットをセットアップしてからブロックを処理する必要があるためです。
3. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
4. AES_CRレジスタのDMAINENビットとDMAOUTENビットをセットして、DMAリクエストを有効にします。
5. 転送完了を示すDMA割込みが発生したら、メモリから AES 処理済みデータを取得します。

注： **AES_DOUTR レジスタの読出しは計算フェーズの最後にソフトウェア操作なしで自動的に DMA によって管理されるため、この方法ではCCFフラグを使用しません。**

NPBLBビットは、GCM、GMAC、CCMの連鎖モードのヘッダフェーズで使用されません。

49.4.5 AES 復号化ラウンドキーの準備

内部キースケジューリングはAESラウンドキーを生成するために使用されます。AES暗号化で、ラウンド0キーは、キーレジスタに格納されているキーです。AES復号化は、最後のラウンドのキーを使用して開始する必要があります。暗号化キーはメモリに格納されているため、復号化キーを取得するには、特別なキースケジューリングを行う必要があります。このキースケジューリングは、ECBおよびCBCモードでAES復号化を行う場合にのみ必要です。

AES_CRのMODE[1:0] ビットフィールドを01にセットしてモード2（キー処理のみ）を選択してから、MODE[1:0] を10（モード3、復号化のみ）にセットして復号化を進める方法が推奨されます。モード2の使用方法は、以下のとおりです。

1. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CRのMODE[1:0] ビットフィールドを01にセットして、モード2を選択します。このキー導出モードは選択された連鎖アルゴリズムとは無関係なので、この場合、CHMOD[2:0] ビットフィールドは重要ではありません。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0] 値による復号化については、セクション49.4.13 を参照してください。
3. AES_CRレジスタのKEYSIZEビットでキー長を128bitまたは256bitにセットします。
4. AES_KEYRxレジスタ（128bitまたは256bit）に暗号化キーを書き込みます。AES_IVRxレジスタへの書込みは無効です。
5. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
6. AES_SRレジスタでCCFフラグがセットされるまで待ちます。
7. CCFフラグをクリアします。導出されたキーはAESコアにあり、すぐに復号化に使用できます。

注： **導出キーが使用できるようになると、AES はハードウェアによって無効にされます。**

導出キーの計算を再開するには、手順 4、5、6、7 を繰り返します。

注： **キー準備の操作は、キーサイズ（128bitまたは256bit）に応じて 59 または 82 クロックサイクル続きます。**

49.4.6 AES 暗号文借用およびデータパディング

サイズがブロックサイズ（128bit）の倍数ではないメッセージを管理するためにECBまたはCBCモードで AES を使用するときには、NIST 特別公報800-38A『ブロック暗号の推奨動作モード：CBCモード用暗号文借用の3つの変種』に記載されているもののような、暗号文借用技術を使用します。AES ペリフェラルはこのような技術をサポートしていないため、アプリケーションは最後から2番目のブロックのデータを使用して、入力データの最終ブロックを完成させる必要があります。

注： **暗号文借用技術は、この参考マニュアルには*記載されていません。***

同様に、AES がECBとCBC以外のモードで使用される場合には、不完全な入力データブロック（128bitより短い入力データを持つブロック）を、暗号化の前に0でパディングする必要があります（データストリングの後端に余分なビットを付加する必要があります）。復号後は、余分なビットを破棄する必要があります。AES では、**最終ブロック**に対する自動データパディング操作が行われるため、アプリケーションがセクション 49.4.4：1908ページの暗号操作を行うための AES の手順に示された推奨に従って、128bitの倍数ではないメッセージを管理する必要があります。

注： **パディングデータは AES_CRレジスタのDATATYPE[1:0] フィールドに従って、通常のデータと同様の方法で入れ替えられます（詳細はセクション 49.4.14 : AES データレジスタおよびデータスワッピングを参照してください）。**

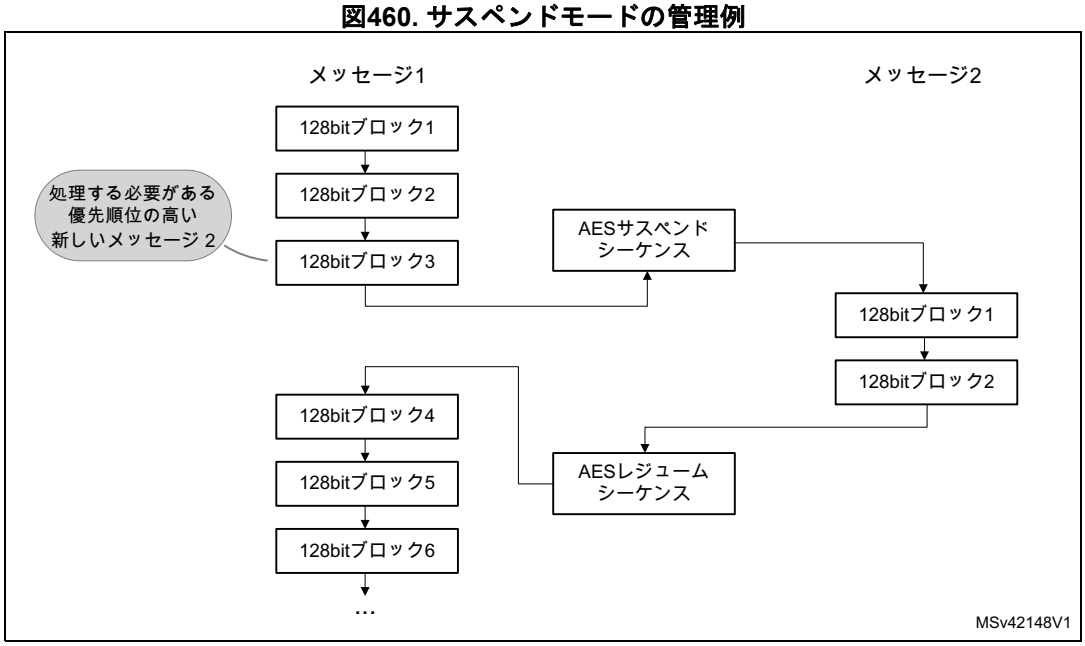
49.4.7 AES タスクのサスペンドとレジューム

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージは暗号化モードでも復号化モードでも再開できます。

サスペンド／レジューム操作によって連鎖操作が途切れることはなく、AES が再び有効になるとすぐにメッセージ処理を再開して、次のデータブロックを受信できます。

図460に、サスペンド／レジューム操作の例を示します。メッセージ1は、より短く優先順位の高いメッセージ2を送信するためにサスペンドされています。

図460. サスペンドモードの管理例



サスペンド／レジューム操作の詳細は、各 AES モード専用のセクションに説明があります。

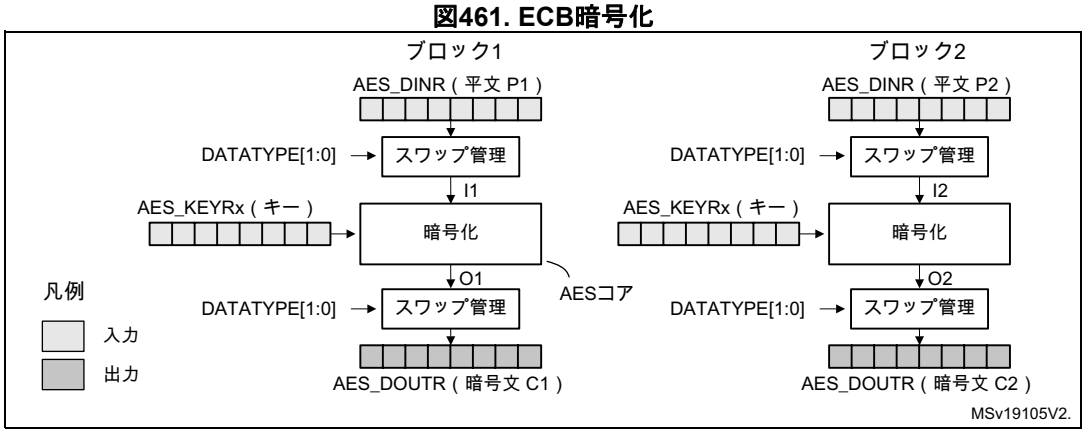
49.4.8 AES 基本連鎖モード（ECB、CBC）

概要

このセクションでは、AES コアが提供するECB暗号化、ECB復号化、CBC暗号化、CBC復号化の4つの基本動作モードについて簡単に説明しています。詳細については、2001年11月26日発行の『FIPS公報197』を参照してください。

図461に、電子コードブック（ECB）暗号化の図を示します。

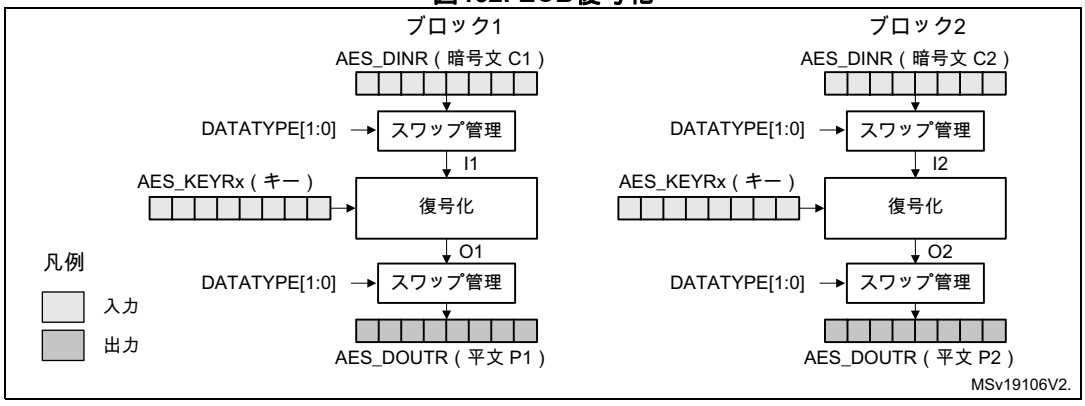
図461. ECB暗号化



ECB暗号化モードでは、まず AES_DINRレジスタの128bitの平文入力データブロックPxのビット／バイト／ハーフワードのスワッピングが行われます。スワップの結果Ixが、128bitまたは256bitキーを使って暗号化モードでセットされたAESコアで処理されます。暗号化の結果Oxのビット／バイト／ハーフワードのスワッピングが行われ、128bitの暗号文出力データブロックCxとして AES_DOUTRレジスタに格納されます。ECB暗号化は、最後の完全な平文ブロックが暗号化されるまで同様に続きます。

図462に、電子コードブック（ECB）復号化の図を示します。

図462. ECB復号化

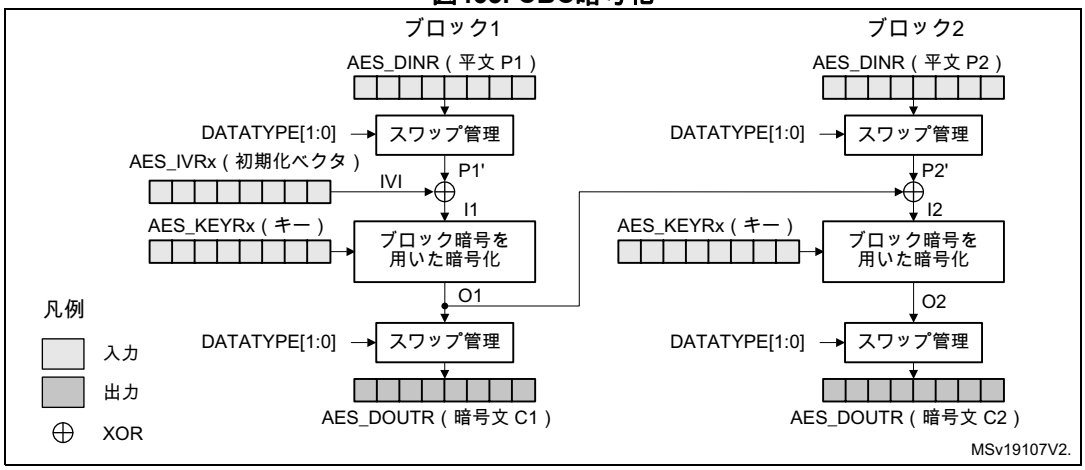


ECBモードでAES復号化を実施するには、最終ラウンドの暗号化キーを回収し、それを暗号文を復号化するための最初のラウンドのキーとして使用して、秘密鍵を準備する必要があります（まず暗号化の完全なキースケジュールを実行する必要があります）。この準備は、AESコアによってサポートされます。

ECB復号化モードでは、まず AES_DINRレジスタの128bitの暗号文入力データブロックC1のビット／バイト／ハーフワードのスワッピングが行われます。このキーイングシーケンスは、ECB暗号化のものに対し、逆になっています。スワップの結果I1は、事前に準備した復号化キーを使って復号化モードでセットされたAESコアで処理されます。復号化結果のビット／バイト／ハーフワードのスワッピングが行われ、128bitの平文出力データブロックP1として AES_DOUTRレジスタに格納されます。ECB復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

図463に、暗号ブロック連鎖（CBC）暗号化の図を示します。

図463. CBC暗号化

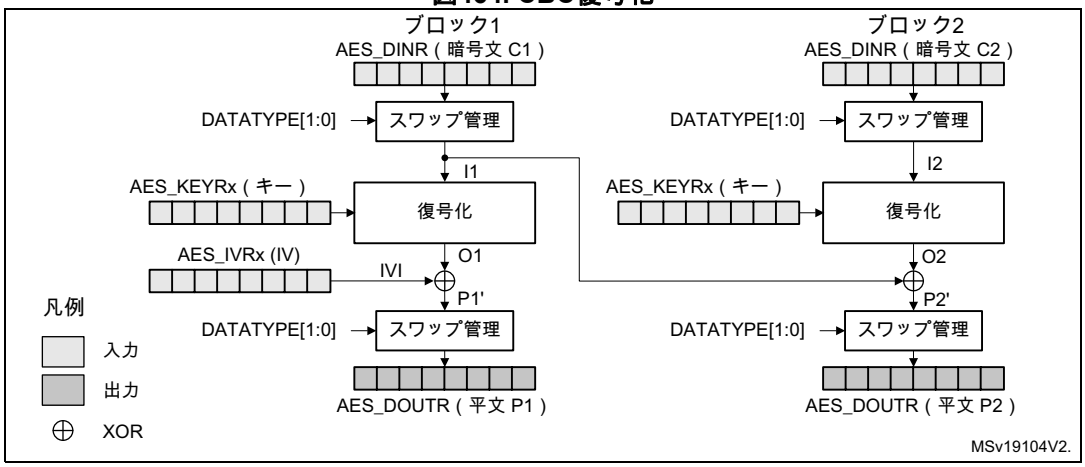


CBC暗号化モードでは、ビット／バイト／ハーフワードのスワッピング（P1’）後、最初の平文入力ブロックと128bitのIVIビットフィールド（初期化ベクタおよびカウンタ）の排他的論理和がとられ、128bitまたは256bitキーを使ってAESコアで暗号化するためのI1入力データを生成します。スワッピング操作後、その結果の128bitの出力ブロックO1が暗号文C1として使用されます。その後、O1データと2番目の平文データP2’ の排他的論理和がとられ、AESコアが暗号文データの2番目を生成するためのI2入力データを生成します。データブロックの連鎖は、メッセージ内の最後の平文ブロックが暗号化されるまで同様に続きます。

メッセージサイズが128bitの倍数ではない場合、最後の不完全なデータブロックは、セクション 49.4.6：AES 暗号文借用およびデータパディングに説明されている方法で暗号化されます。

図464 に、暗号ブロック連鎖（CBC）復号化の図を示します。

図464. CBC復号化



ECB復号化モードと同様に、CBC復号化モードでAES復号化を実施するには、秘密鍵を準備する必要があります。

キーの準備処理後、最初の128bitの暗号文ブロック（スワップ操作後）が128bitまたは256bitキーを使った復号化操作用のAESコア入力ブロックI1として直接使用され、復号化が進みます。その出力O1と、128bitのIVIフィールド（暗号化で使用したものと同じである必要があります）の排他的論理和をとって、最初の平文ブロックP1を生成します。

2番目の暗号文ブロックは、最初のブロックのI1データが初期化ベクタの代わりに使用されることを除けば、最初のブロックと同じ方法で処理されます。

復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

メッセージサイズが128bitの倍数ではない場合、最後の不完全なデータブロックは、セクション 49.4.6：AES 暗号文借用およびデータパディングに説明されている方法で復号化されます。

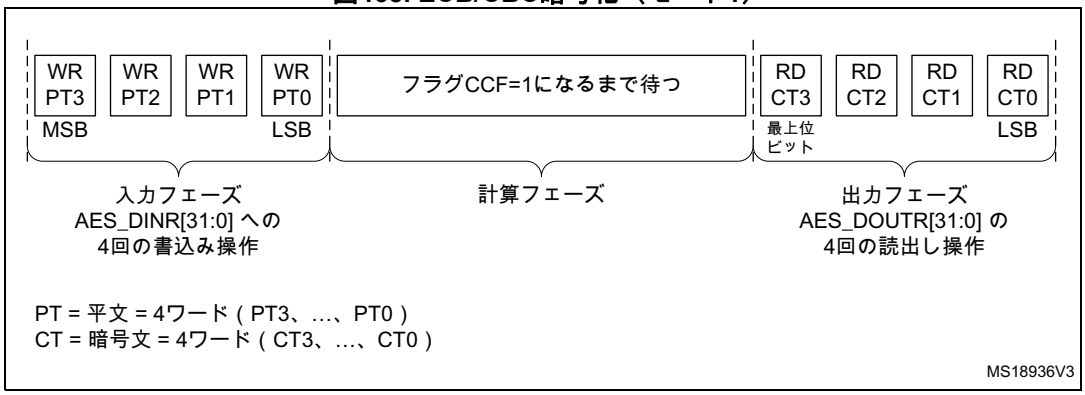
データスワッピングの詳細については、セクション 49.4.14：AES データレジスタおよびデータスワッピングを参照してください。

ECB/CBC暗号化シーケンス

ECB/CBC暗号化を行うためのイベントのシーケンス（詳細はセクション49.4.4 を参照）：

1. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CRレジスタのMODE[1:0] ビットフィールドを00にセットしてモード1を選択し、AES_CRレジスタのCHMOD[2:0] ビットフィールドをそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。データ型もDATATYPE[1:0] ビットフィールドで定義できます。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0] 値による暗号化については、セクション49.4.13を参照してください。
3. AES_CRレジスタのKEYSIZEビットで128bitまたは256bitのキー長を選択します。
4. AES_KEYRxレジスタ（128bitまたは256bit）に暗号化キーを書き込みます。CBCモードを選択した場合、AES_IVRxレジスタに初期化ベクタデータを書き込みます。
5. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
6. AES_DINRレジスタに図465に示すとおり4回書き込んで平文（MSBファースト）を入力します。
7. AES_SRレジスタでCCFフラグがセットされるまで待ちます。
8. AES_DOUTRレジスタを図465に示すとおり4回読み出して、暗号文（MSBファースト）を取得します。次に、AES_ICR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
9. 手順 6-7-8を繰り返して、同じ暗号化キーですべてのブロックを処理します。

図465. ECB/CBC暗号化（モード1）

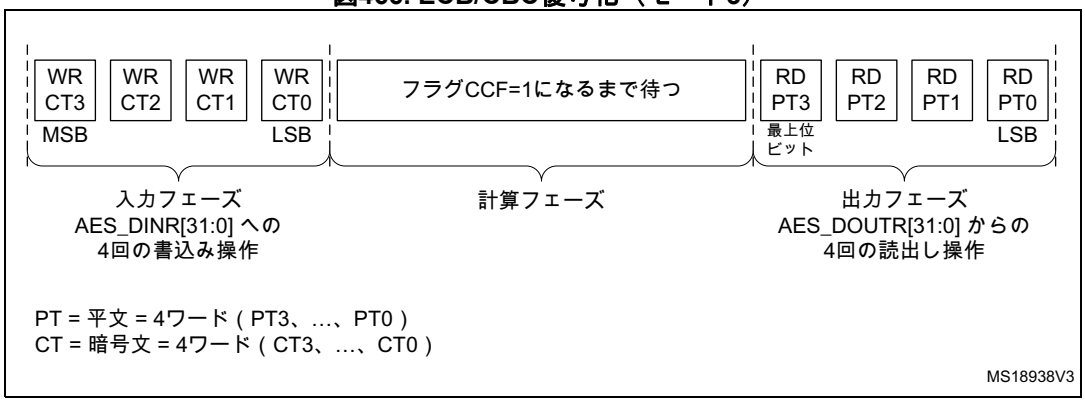


ECB/CBC復号化シーケンス

AES ECB/CBC復号化を行うためのイベントのシーケンス（詳細はセクション49.4.4を参照）は次のとおりです。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0]値による復号化については、セクション49.4.13を参照してください。

1. セクション 49.4.5：AES 復号化ラウンドキーの準備に示された手順に従って、AESコアで復号化キーを準備します。
2. AES_CRレジスタのMODE[1:0] ビットフィールドを10にセットしてモード3を選択し、AES_CRレジスタのCHMOD[2:0] ビットフィールドをそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。データ型もDATATYPE[1:0] ビットフィールドで定義できます。KEYSIZE ビットフィールドは、そのままの状態を維持する必要があります。
3. AES_IVRxレジスタに初期化ベクタを書き込みます（CBCモードのみ必要）。
4. AES_CRレジスタのENビットをセットして、AES を有効にします。
5. AES_DINRレジスタに図466に示すとおり4回書き込んで暗号文（MSBファースト）を入力します。
6. AES_SRレジスタでCCFフラグがセットされるまで待ちます。
7. AES_DOUTRレジスタを図466に示すとおり4回読み出して、平文（MSBファースト）を取得します。次に、AES_ICR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
8. 手順 5-6-7を繰り返して、同じキーで暗号化されたすべてのブロックを処理します。

図466. ECB/CBC復号化（モード3）



ECB/CBCモードでのサスペンド／レジューム動作

メッセージの処理をサスペンドするには、次の手順に従います。

1. DMAが使用されている場合には、AES_CRレジスタのDMAINENビットをクリアして、IN FIFOへの AES DMA転送を停止します。
2. DMAが使用されていない場合は、AES_DOUTRレジスタを4回読み出して、最後に処理されたブロックを保存します。DMAが使用されている場合には、AES_SRレジスタでCCFフラグがセットされるまで待ち、AES_CRレジスタのDMAOUTENビットをクリアして、OUT FIFOからのDMA転送を停止します。
3. DMA が使用されていない場合は、AES_SR レジスタの CCF フラグが 1（計算完了）になるまでポーリングします。
4. AES_ICR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
5. 初期化ベクタレジスタを保存します（データ処理中に AES_IVRxレジスタが変更されるCBCモードでのみ必要です）。

6. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
7. AES_CRレジスタを保存して、キーレジスタが優先順位の高いメッセージを処理するのに必要でなければ、それをクリアします。
8. DMAが使用されている場合には、DMAコントローラのステータス（INおよびOUTデータ転送のポインタ、残りバイト数など）を保存します。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMAが使用されている場合には、DMAコントローラを設定して残りのFIFO INとFIFO OUT転送を完了させます。
2. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
3. AES_CRレジスタを（正しいKEYSIZEで）復元してから、AES_KEYRxレジスタを復元します。
4. セクション 49.4.5：AES　復号化ラウンドキーの準備に示したように復号化キーを準備します（ECBまたはCBC復号化のみ必要）。
5. 保存した設定を使用して AES_IVRxレジスタを復元します（CBCモードのみ必要）。
6. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
7. DMAが使用されている場合には、AES_CRレジスタのDMAINENビットとDMAOUTENビットをセットして、AES DMA転送を有効にします。

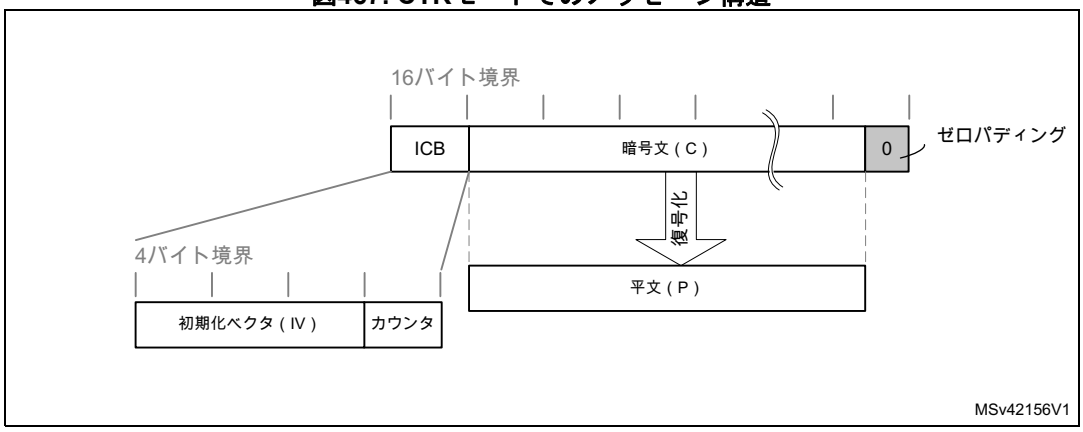
49.4.9 AES カウンタ（CTR）モード

概要

カウンタモード（CTR）では、AES をキーストリームジェネレータとして使用します。生成されたキーは、その後平文との排他的論理和をとって暗号化されます。

CTR連鎖は、NIST 特別公報800-38A『ブロック暗号の推奨動作モード』の中に定義されています。CTRモードでの典型的なメッセージ構造を図467に示します。

図467. CTRモードでのメッセージ構造



このメッセージの構造は次のとおりです。

- 16バイトの初期カウンタブロック（ICB）は、次の2つの異なるフィールドで構成されています。
  - **初期化ベクタ**（IV）：ある特定のキーでの暗号化サイクルごとに一意でなければならない96bitの値です。
  - **カウンタ**：ブロック処理が完了するたびにインクリメントされる32bitのビッグエンディアンの整数です。カウンタの初期値は1に設定する必要があります。
- 平文Pは、既知の長さの暗号文Cとして暗号化されます。この長さは16バイトの倍数でなくとも構いませんが、その場合には平文のパディングが必要です。

CTRの暗号化と復号化

図468および図469では、AES ペリフェラルで実行されるCTRの暗号化および復号化の処理についてそれぞれ示しています。CTRモードは、AES_CRレジスタのCHMOD[2:0] ビットフィールドに010を書き込んで選択します。

図468. CTR暗号化

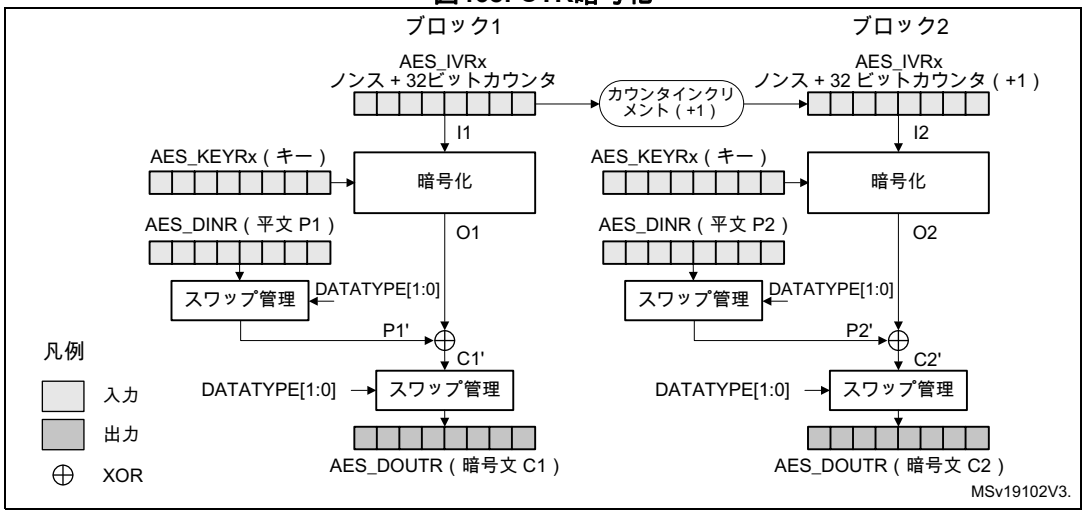
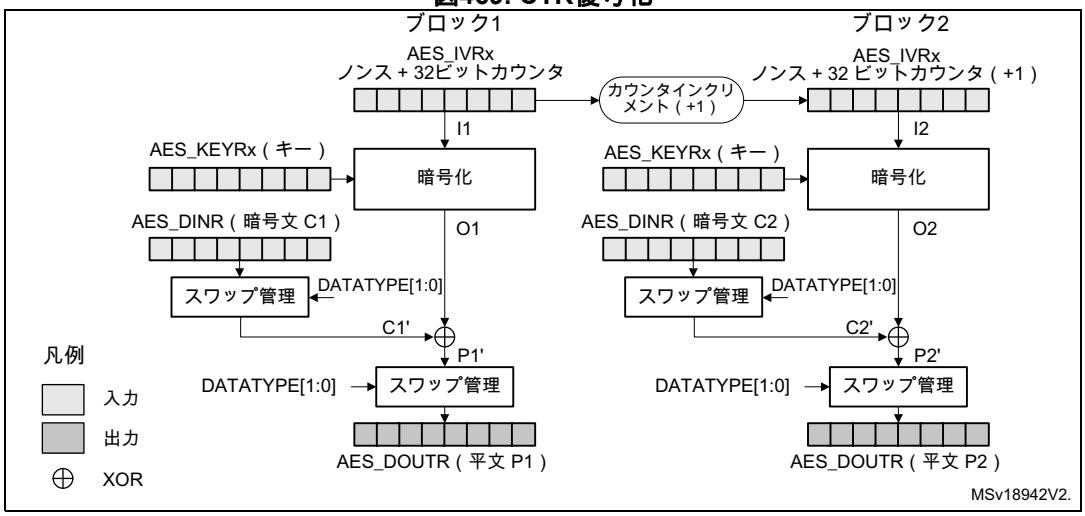


図469. CTR復号化



CTRモードでは、暗号コア出力（キーストリームとも呼ばれます）Oxと関連入力ブロック（暗号化ではPx'、復号化ではCx'）の排他的論理和がとられ、正しい出力ブロック（暗号化ではCx'、復号化ではPx'）を生成します。AES の初期化ベクタは、表464に示すように初期化する必要があります。

表464.CTRモード初期化ベクタの定義

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
IVI[127:96]	IVI[95:64]	IVI[63:32]	IVI[31:0] 32bitカウンタ = 0x0001

最初のデータブロックを処理するときに一度だけ AES_IVRx レジスタを使用するCBCモードとは違って、CTRモードでは、各データブロックを処理するたびに AES_IVRxレジスタが使用され、AES ペリフェラルによって初期化ベクタのカウンタビットがインクリメントされます（ノンスビットは変更されません）。

コアが常に現在のカウンタブロックを暗号化して、平文（CTR暗号化）や暗号文（CTR復号化）の入力と排他的論理和がとられるキーストリームを生成するので、CTR復号化はCTR暗号化と変わりません。CTRモードでは、MODE[1:0] ビットフィールドを01（キー導出）にセットすることが禁止されており、その他すべての設定はデフォルトで暗号化モードになります。

CTR連鎖モードで暗号化または復号化を行うためのイベントのシーケンス：

1. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CRレジスタのCHMOD[2:0] ビットフィールドを010にセットして、CTR連鎖モードを選択します。MODE[1:0] ビットフィールドを01以外の値にセットします。
3. AES_KEYRxレジスタを初期化して、AES_IVRxレジスタを表464に示されたとおりロードします。
4. AES_CRレジスタのENビットをセットして、現在のカウンタの暗号化を開始します（ENは計算が終了すると自動的にリセットされます）。
5. 最終ブロックの場合には、必要に応じてデータを0でパディングして完全なブロックとします。
6. AES でデータを追加して結果を読み出します。可能性のある3通りのケースはセクション 49.4.4 : 暗号操作を行うための AES の手順に記載されています。
7. 最後から2番目のブロックが処理されるまで前の手順を繰り返します。最終ブロックでは、前の2つの手順を適用して、ペイロードの一部ではないビットを破棄します（最終入力ブロックの有効なデータのサイズが16バイト未満の場合）。

CTRモードでのサスペンド／レジューム動作

CBCモードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。CBCサスペンド／レジュームシーケンスの詳細については、セクション 49.4.8 : AES 基本連鎖モード（ECB、CBC）を参照してください。

注： **CBCモードと同様に、レジューム動作中に AES_IVRxレジスタを再ロードする必要があります。**

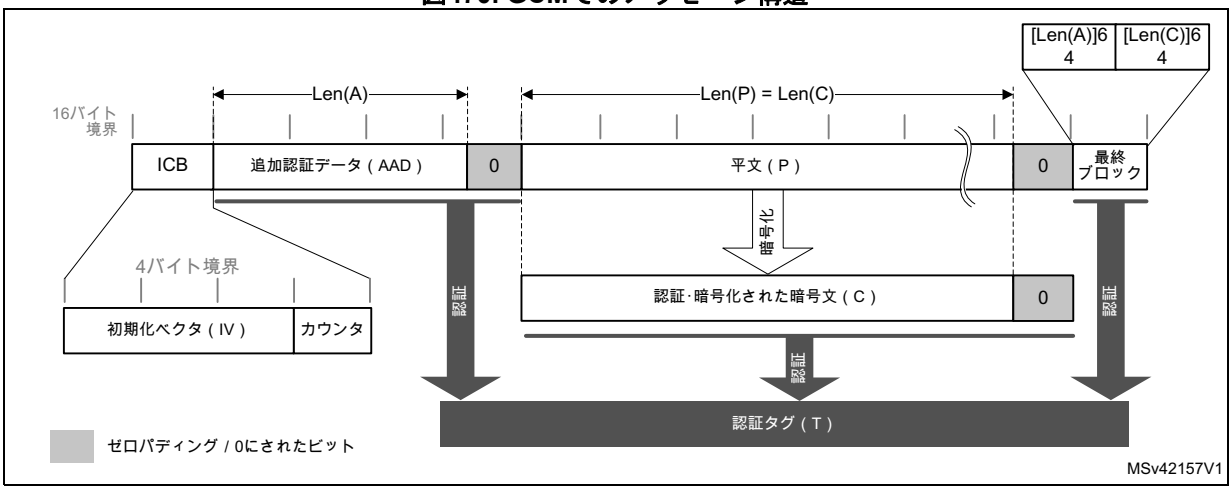
49.4.10 AES ガロア／カウンタモード（GCM）

概要

AESガロア／カウンタモード（GCM）では、平文メッセージを対応する暗号文およびタグ（メッセージ認証コードとも言います）に暗号化して認証できます。機密性の確保のため、GCMアルゴリズムはAESカウンタモードに基づいています。固定の有限フィールドを超えて乗算器を使用しタグを生成します。

GCM連鎖は、NIST 特別公報800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』の中に定義されています。GCMモードでの典型的なメッセージ構造を図470に示します。

図470. GCMでのメッセージ構造



メッセージには、次の構造があります。

- **16バイトの初期カウンタブロック（ICB）**は、次の2つの異なるフィールドで構成されています。
  - **初期化ベクタ（IV）**：ある特定のキーでの暗号化サイクルごとに一意でなければならない96bitの値です。GCM規格では96bit未満のIVに対応していますが、この場合には規則が厳格に適用されることに注意してください。
  - **カウンタ**：ブロック処理が完了するたびにインクリメントされる32bitのビッグエンディアンの整数です。NIST仕様によると、ペイロードの最初のブロックを処理するときのカウンタ値は0x2です。
- **認証済みヘッダAAD**（追加認証データとも言います）は、既知の長さであるLen(A) を持ちます。この長さは16バイトの倍数でなくても構いませんが、2^{64}1bitを超えてはなりません。メッセージのこの部分には認証だけ行われて、暗号化はされません。
- **平文メッセージP** には暗号文Cとして認証と暗号化の両方が行われ、既知の長さであるLen(P) を持ちます。この長さは16バイトの倍数でなくても構いませんが、128bitの2^{32}-2ブロック分を超えることはできません。
- **最終ブロック**には、表465に示すとおり、AADヘッダ長（ビット [32:63]）とペイロード長（ビット [96:127]）の情報が含まれます。

GCM規格では、暗号文Cは平文Pと同じビット長であることと規定されています。

メッセージ（AADまたはP）の長さが16バイトの倍数ではない場合には、特別なパディング処理が必要となります。

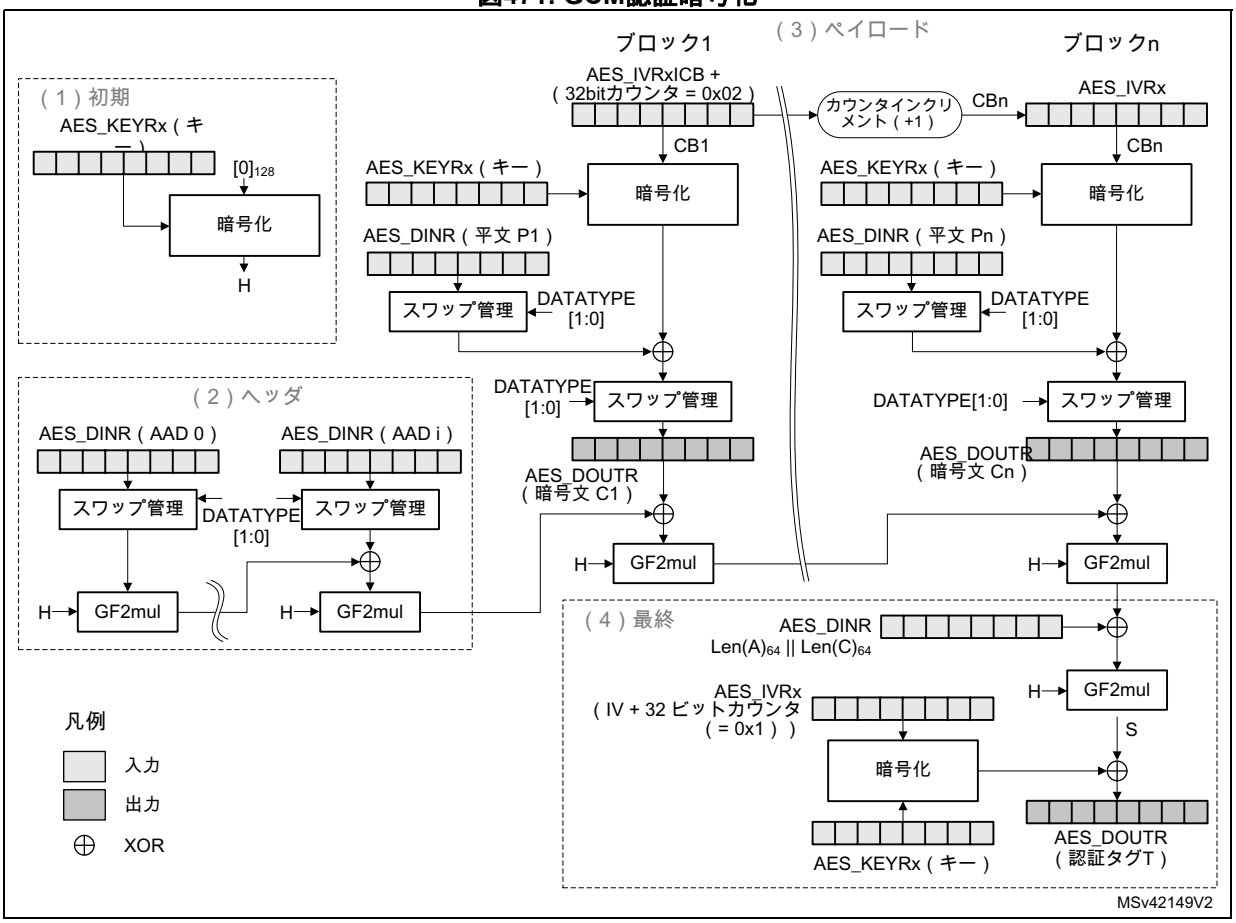
表465.GCM最終ブロックの定義

エンディアン	ビット [0] --- ビット [31]	ビット [32] --- ビット [63]	ビット [64] --- ビット [95]	ビット [96] --- ビット [127]
入力データ	0x0	AAD長 [31:0]	0x0	ペイロード長 [31:0]

GCM処理

図471には、AES ペリフェラルでのGCMの実装について示しています。GCMは、AES_CRレジスタのCHMOD[2:0] ビットフィールドに011を書き込んで選択します。

図471. GCM認証暗号化



GCMモードにおける平文の機密性のメカニズムはカウンタモードと類似しており、入力カウンタブロックのシーケンスを生成する特別なインクリメント機能（32bitインクリメントと示される）を備えています。

各データブロックの処理には AES_IVRxレジスタ（データの**カウンタブロック**を保持）が使用されます。AES ペリフェラルが、カウンタ [31:0] ビットフィールドを自動的にインクリメントします。最初のカウンタブロック（CB1）は、アプリケーションソフトウェアによって初期カウンタブロックICBから導出されます（表466参照）。

表466.GCMモードでの AES_IVRxレジスタの初期化

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
ICB[127:96]	ICB[95:64]	ICB[63:32]	ICB[31:0] 32bitカウンタ = 0x0002

注： **このモードでは、MODE[1:0] ビットフィールドを01（キー導出）にセットすることが禁止されています。**

GCMモードにおける認証メカニズムは **GF2mul** と呼ばれるハッシュ機能に基づいており、このハッシュ機能では、バイナリガロア体の中で、ハッシュサブキー（H）と呼ばれる固定パラメータによる乗算が行われます。

GCMメッセージは、次の項目に詳述されている以下のフェーズで処理されます。

- **初期フェーズ**：AES がGCMハッシュサブキー（H）を準備します。
- **ヘッダフェーズ**：AES が追加認証データ（AAD）を処理して、ハッシュの計算のみを行います。
- **ペイロードフェーズ**：AES が平文（P）を処理して、ハッシュの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- **最終フェーズ**：AES がメッセージの最終ブロックを使って認証タグ（T）を生成します。

GCM初期フェーズ

この最初のステップでは、GCMハッシュサブキー（H）が計算され、すべてのブロックの処理に使用されるように内部に保存されます。推奨シーケンスは、次のとおりです。

1. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CRレジスタのCHMOD[2:0] ビットフィールドを011にセットして、GCM連鎖モードを選択し、オプションでDATATYPE[1:0] フィールドをセットします。
3. AES_CRレジスタのGCMPH[1:0] ビットフィールドを00にセットして、初期フェーズを示します。
4. AES_CRレジスタのMODE[1:0]　ビットフィールドを00または10にセットします。このビットフィールドはペイロードフェーズでのみ使用されますが、初期フェーズで設定し、後続のすべてのフェーズで変更しないことをお勧めします。
5. キーで AES_KEYRxレジスタを初期化して、AES_IVRxレジスタを表466で定義された情報で初期化します。
6. AES_CRレジスタのENビットを1にセットしてハッシュキーの計算を開始します（ENは計算が終了すると自動的にリセットされます）。
7. AES_SRのCCFフラグが1に変わって計算が終了するまで待ちます。あるいは、対応する割込みを使用します。
8. AES_ICR レジスタの CCF ビットをセットして、AES_SRのCCFフラグをクリアします。

GCMヘッダフェーズ

このフェーズはGCM初期フェーズの後に発生し、ペイロードフェーズの前に完了させる必要があります。実行するシーケンスは、暗号化および復号化と同じで次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを01にセットして、ヘッダフェーズを示します。初期フェーズで設定したMODE[1:0] ビットフィールドは変更しないでください。
2. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、ブロックのAADサイズが128bitを下回る場合、ブロックの残り部分を0でパディングします。次に、AES に セクション 49.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加します。このフェーズ中に読み出されるデータはありません。
4. 最後の追加認証データブロックが処理されるまで手順 3を繰り返します。

注： **ヘッダフェーズは、AADがなければ（Len(A) = 0であれば）スキップできます。**

GCMペイロードフェーズ

このフェーズは、暗号化および復号化と同じでGCMヘッダフェーズの後に実行されます。このフェーズでは、暗号化／復号化したペイロードが AES_DOUTRレジスタに格納されます。実行するシーケンスは、次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを10にセットして、ペイロードフェーズを示します。初期フェーズで設定したMODE[1:0] ビットフィールドは変更しないでください。
2. ヘッダフェーズをスキップした場合、AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、平文（暗号化）または暗号文（復号化）のサイズが128bitを下回る場合、ブロックの残り部分を0でパディングします。
4. AES に セクション 49.4.4：1908ページの暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加して、結果を読み出します。
5. 最後から2番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号化されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の2つの手順を実行します。最終ブロックでは、最終ブロックのサイズが16バイト未満である場合に、ペイロードの一部ではないビットを破棄します。

注： **ペイロードフェーズは、ペイロードデータがなければ（Len(C) = 0であれば）スキップできます（GMACモード参照）。**

GCM最終フェーズ

この最終フェーズでは、AES ペリフェラルがGCM認証タグを生成して AES_DOUTRに格納します。実行するシーケンスは、次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを11にセットして、最終フェーズを示します。
2. 表465に示すとおり、AADとペイロードのビット長を連結してブロックのデータを構成します。AES_DINRレジスタにブロック書き込みます。
3. AES_SRのCCFフラグが1に変わって計算が終了するまで待ちます。
4. AES_DOUTRレジスタを4回読み出してGCM認証タグを取得します。
5. AES_ICR レジスタの CCF ビットをセットして、AES_SRのCCFフラグをクリアします。
6. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。認証された復号化である場合には、生成されたタグをメッセージとともに渡された予測タグと比較します。

注： **最終フェーズでは、データは通常 AES_DINRに書き込まれます（スワッピングなし）。一方、AES_DOUTRから読み出されたタグ・データにはスワッピングが適用されます。**

ヘッダフェーズまたはペイロードフェーズから最終フェーズに遷移する際に、AES を無効にしないでください。そうしないと、誤った結果になります。

GCMモードでのサスペンド／レジューム動作

メッセージの処理をサスペンドするには、次の手順に従います。

1. DMAが使用されている場合には、AES_CRレジスタのDMAINENビットをクリアして、IN FIFOへの AES DMA転送を停止します。DMAが使用されていない場合は、AES_SRレジスタのCCFフラグが1にセットされ、現在の計算が完了したことを確認します。
2. ペイロードフェーズで、DMAが使用されていない場合は、AES_DOUTRレジスタを4回読み出して、最後に処理されたブロックを保存します。DMAが使用されている場合には、AES_SRレジスタでCCFフラグがセットされるまで待ち、AES_CRレジスタのDMAOUTENビットをクリアして、OUT FIFOからのDMA転送を停止します。
3. AES_ICR レジスタの CCF ビットをセットして、AES_SRのCCFフラグをクリアします。
4. メモリに AES_SUSPxRレジスタを保存します（ここで、xは0から7です）。
5. ペイロードフェーズでは、AES_IVRxレジスタがデータ処理中に初期値から変化するため、それらを保存します。ヘッダフェーズでは、この手順は不要です。
6. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
7. メモリに現在の AES 設定を保存します（初期化ベクタレジスタ AES_IVRx以外）。アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。
8. DMAが使用されている場合には、DMAコントローラのステータス（ INデータ転送のポインタ、残りバイト数など）を保存します。ペイロードフェーズでは、OUTデータ転送のポインタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMAが使用されている場合には、DMAコントローラを設定して残りのFIFO IN転送を完了させます。ペイロードフェーズでは、残りのFIFO OUT転送もDMAコントローラで設定する必要があります。
2. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
3. メモリに事前に保存したサスペンドレジスタの値を、対応する AES_SUSPxRレジスタに書き戻します（ここで、xは0から7です）。
4. ペイロードフェーズでは、メモリに事前に保存した初期化ベクタレジスタの値を、対応するAES_IVRxレジスタに書き戻します。ヘッダフェーズでは、初期設定値を AES_IVRxレジスタに書き戻します。
5. AES_CRと AES_KEYRxのレジスタに初期設定値を復元します。
6. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。

DMAが使用されている場合には、AES_CR レジスタのDMAINENビット（ペイロードフェーズではDMAOUTENビットも）をセットして、AES DMAリクエストを有効にします。

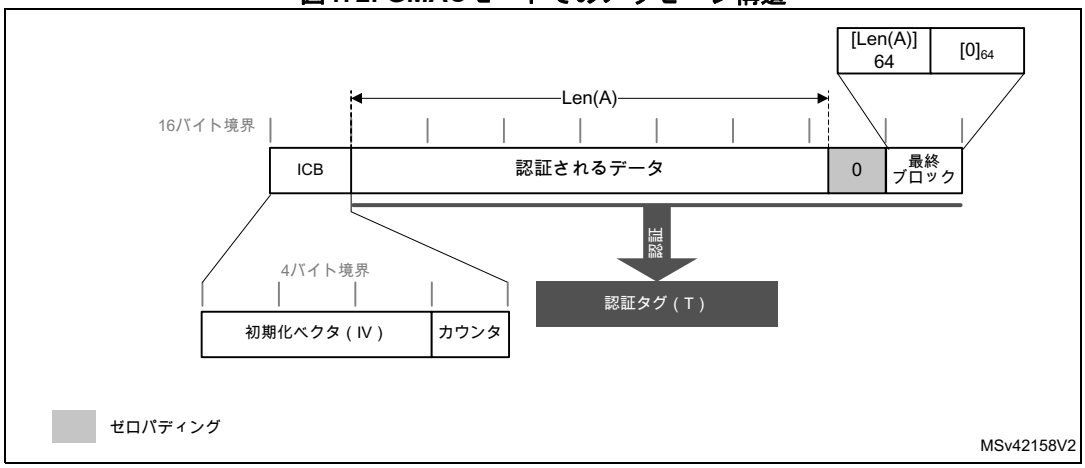
49.4.11 AES ガロアメッセージ認証コード（GMAC）

概要

ガロアメッセージ認証コード（GMAC）では、平文を認証して、対応するタグ情報（メッセージ認証コードとも言います）を生成することができます。NIST 特別公報800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』に定義されているGCMアルゴリズムに基づいています。

GMACでの典型的なメッセージ構造を図472に示します。

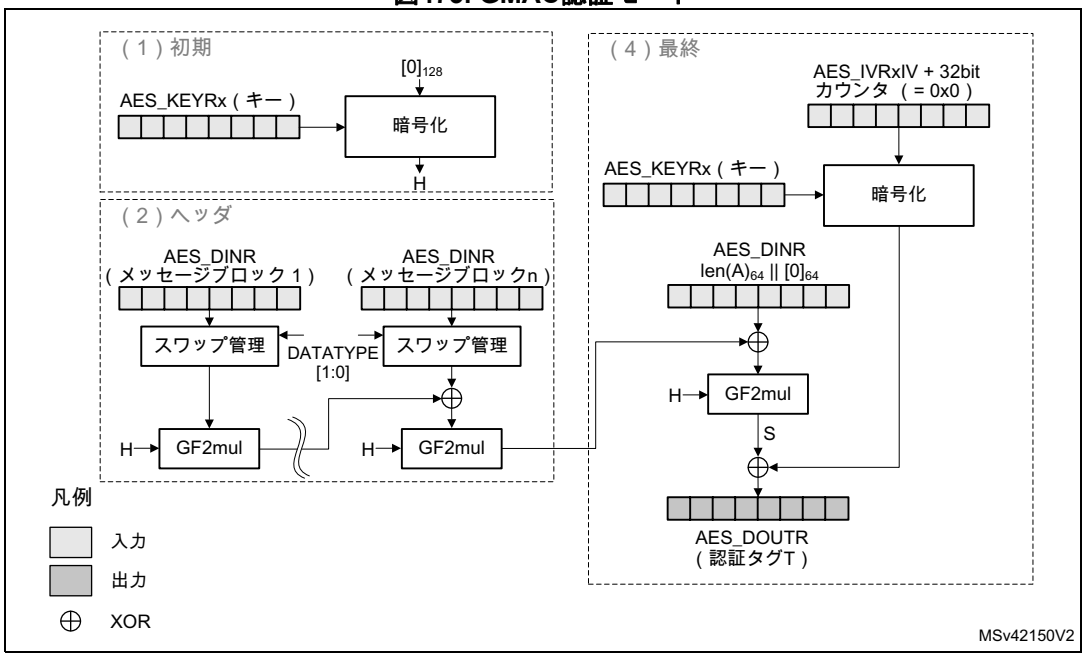
図472. GMACモードでのメッセージ構造



AES GMAC処理

図473には、AES ペリフェラルでのGMACモードの実装について示しています。このモードは、AES_CRレジスタのCHMOD[2:0] ビットフィールドに011を書き込んで選択します。

図473. GMAC認証モード



GMACアルゴリズムはヘッダのみ含まれているメッセージに適用されるGCMアルゴリズムに相当します。その結果、ペイロードフェーズが省略されること以外の手順と設定がすべてGCMと同じです。

GMACでのサスペンド／レジューム動作

GMACモードでは、ヘッダフェーズのみが割込み可能であることを除いて、GCMで示されたシーケンスが適用されます。

49.4.12 AESCBC-MAC付きカウンタ（CCM）

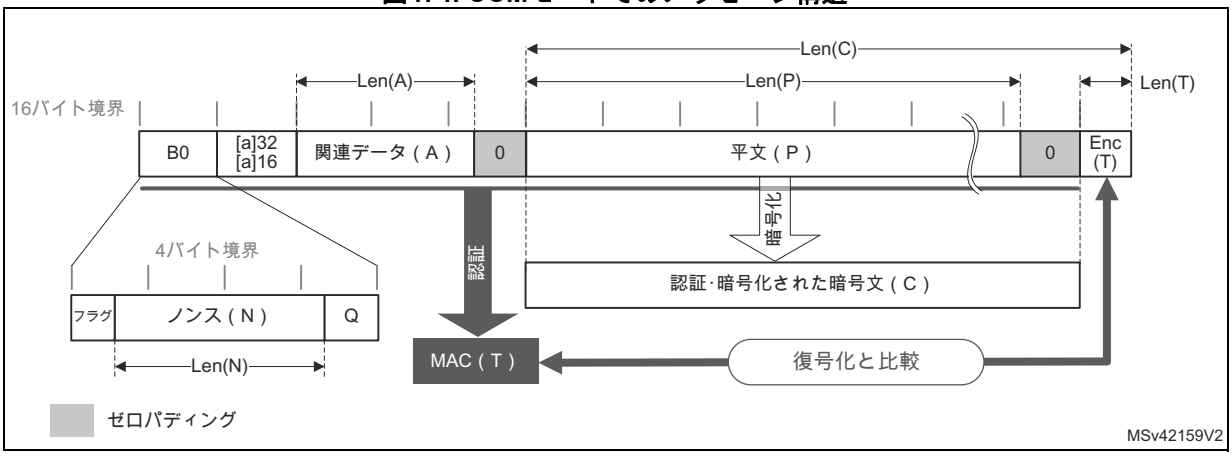
概要

AES **暗号ブロック連鎖-メッセージ認証コード付きカウンタ**（CCM）アルゴリズムでは、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、CCMアルゴリズムはカウンタモードでの AES に基づいています。暗号ブロック連鎖技術を使ってメッセージ認証コードを生成します。これは、一般的にCBC-MACと呼ばれます。

注： **NISTは、CCM仕様の関係以外ではCBC-MACを認証モードとして承認していません。**

CCM連鎖は、NIST 特別公報800-38C『ブロック暗号の推奨動作モード - 認証および機密性のためのCCMモード』の中に規定されています。CCMでの典型的なメッセージ構造を図474に示します。

図474. CCMモードでのメッセージ構造



メッセージの構造は次のとおりです。

- **16バイトの初期認証ブロック（B0）**は、次の3つの異なるフィールドで構成されています。
  - **Q**：Pのオクテット長（Len(P)）を表すビットストリングです。
  - **ノンス（N）**：Len(N) サイズの使い捨て値です（新しい通信のたびに新しいノンスを割り当てる必要があります）。Len(N) + Len(P) の合計は、15バイトにする必要があります。
  - **フラグ**：規格に規定されている、制御情報のための4つのフラグを格納する最上位オクテットです。値 **t**（バイト単位でのMACの長さ）と **Q**（Len(P) < 2^{8q} バイトとなる平文の長さ）をエンコードするための3bitのストリングが2つ格納されています。**Q** に関連付けられるカウンタブロックの範囲は2^{8Q-4} となり、**Q** の最大値が8である場合、暗号で使用されるカウンタブロックは60bitである必要があります。
- **16バイトブロック（B）**は関連データ（A）に関連付けられます。
  メッセージのこの部分には認証だけ行われて、暗号化はされません。この部分は既知の長さであるLen(A) を持っており、この長さは16バイトの倍数でなくとも構いません（図474参照）。規格

には、最初のメッセージブロック（B1）の上位側ビットにおいて、バイト数で表現された関連データ長（a）は次の定義のようにエンコードされなければならないとも示されています。

– $0 < a < 2^{16} - 2^8$ である場合には、$[a]_{16}$ のようにエンコードされ、2バイトとなります。
– $2^{16} - 2^8 < a < 2^{32}$ である場合には、0xff || 0xfe || $[a]_{32}$ のようにエンコードされ、6バイトとなります。
– $2^{32} < a < 2^{64}$ である場合には、0xff || 0xff || $[a]_{64}$ のようにエンコードされ、10バイトとなります。

- **16バイトブロック（B）**は既知の長さLen(P) を持ち、暗号文Cとして認証と暗号化の両方が行われている平文メッセージPに関連付けられます。この長さは16バイトの倍数でなくとも構いません（図474参照）。
- 長さLen(T) の**暗号化されたMAC (T)** が、全長がLen(C) である暗号文Cに追加されます。

メッセージ（AまたはP）の長さが16バイトの倍数ではない場合には、特別なパディング処理が必要となります。

注： **CCM連鎖モードは、関連データのみ（ペイロードなし）にも使用できます。**

例として、NIST特別公報800-38CのセクションC.1には次の値（16進数）が示されています。

N：10111213 141516（Len(N) = 56bitまたは7バイト）

A：00010203 04050607（Len(A) = 64bitまたは 8 バイト）

P：20212223（Len(P) = 32bitまたは 4 バイト）

T：6084341B（Len(T) = 32bitまたはt = 4）

B0：4F101112 13141516 00000000 00000004
B1：00080001 02030405 06070000 00000000
B2：20212223 00000000 00000000 00000000

CTR0：0710111213 141516 00000000 00000000

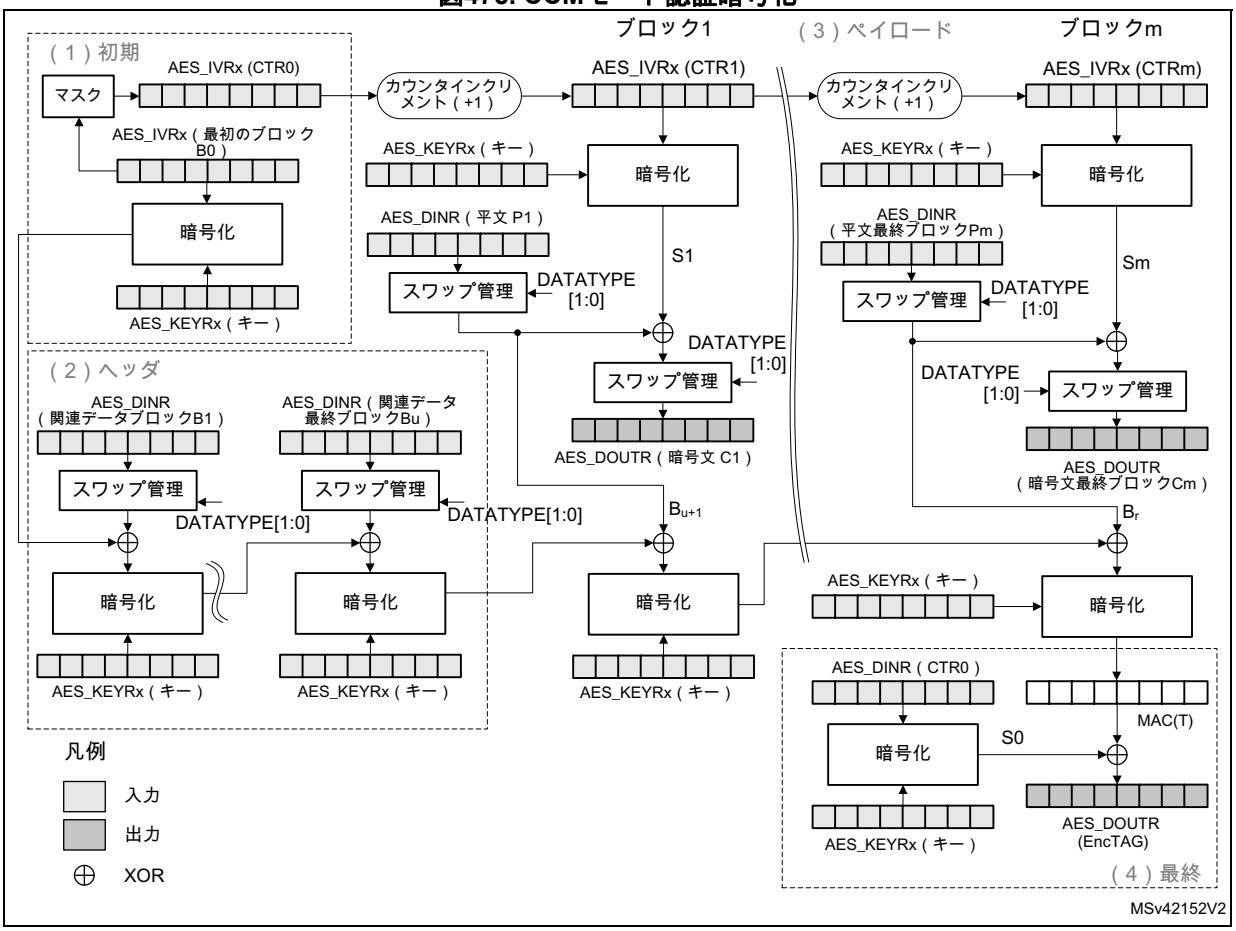
CTR1：0710111213 141516 00000000 00000001

フォーマットされた入力データブロックBx（特にB0とB1）の生成は、アプリケーションで管理する必要があります。

CCM処理

図475には、AES ペリフェラルでのCCMの実装（暗号化の例）について示しています。このモードは、AES_CRレジスタのCHMOD[2:0] ビットフィールドに100を書き込んで選択します。

図475. CCMモード認証暗号化



生成-暗号化処理に入力するデータは、有効なナンス、有効なペイロードストリング、有効な関連データストリングであり、すべて適切にフォーマットされているものです。CBC連鎖メカニズムはフォーマットされた平文データに適用されて、既知の長さを持つMACを生成します。入力として十分な長さのカウンタブロックのシーケンスが必要であるカウンタモードの暗号化は、ペイロードストリングとMACに個別に適用されます。結果の暗号文Cは、平文Pに対する生成-暗号化処理の出力です。

AES_IVRxレジスタは各データブロックを処理するために使用され、AES によって、最初のブロックB0により定義されるビット長を持つCTRカウンタが自動的にインクリメントされます。表467には、アプリケーションがB0データをどのように読み込むかについて示しています。

注： **CCMモードでの AES ペリフェラルは、NISTが指定する最大64bitのカウンタをサポートしています。**

表467.CCMモードでの AES_IVRxレジスタの初期化

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
B0 [127:96]	B0 [95:64]	B0 [63:32]	B0 [31:0]

注： **このモードでは、MODE[1:0] ビットフィールドを01（キー導出）にセットすることが禁止されています。**

CCMメッセージは、次の項目に詳述されている以下のフェーズで処理されます。

- **初期フェーズ**：AES が最初のブロックを処理して最初のカウンタブロックを準備します。
- **ヘッダフェーズ**：AES が関連データ（A）を処理して、タグの計算のみを行います。
- **ペイロードフェーズ**：IPが平文（P）を処理して、タグの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- **最終フェーズ**：AES がメッセージ認証コード（MAC）を生成します。

CCM初期フェーズ

このフェーズでは、CCMメッセージの最初のブロックB0が AES_IVRxレジスタに書き込まれます。AES_DOUTRレジスタに出力データは一切格納されません。推奨シーケンスは、次のとおりです。

1. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CRレジスタのCHMOD[2:0] ビットフィールドを100にセットして、CCM連鎖モードを選択し、オプションでDATATYPE[1:0] フィールドをセットします。
3. AES_CRレジスタのGCMPH[1:0] ビットフィールドを00にセットして、初期フェーズを示します。
4. AES_CRレジスタのMODE[1:0] ビットフィールドを00または10にセットします。このビットフィールドはペイロードフェーズでのみ使用されますが、初期フェーズで設定し、後続のすべてのフェーズで変更しないことをお勧めします。
5. キーで AES_KEYRxレジスタを初期化して、AES_IVRxレジスタを表467に示されたB0データで初期化します。
6. AES_CRレジスタのENビットを1にセットしてカウンタの計算を開始します（ENは計算が終了すると自動的にリセットされます）。
7. AES_SRのCCFフラグが1に変わって計算が終了するまで待ちます。あるいは、対応する割込みを使用します。
8. AES_ICR レジスタの CCF ビットに1をセットして、AES_SRレジスタのCCFフラグをクリアします。

CCMヘッダフェーズ

このフェーズはGCM初期フェーズの後に発生し、ペイロードフェーズの前に完了させる必要があります。このフェーズでは、AES_DOUTRレジスタには出力データは一切格納されません。

実行するシーケンスは、暗号化および復号化と同じで次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを01にセットして、ヘッダフェーズを示します。初期フェーズで設定したMODE[1:0] ビットフィールドは変更しないでください。
2. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、ブロックのAADサイズが128bitを下回る場合、ブロックの残り部分を0でパディングします。次に、AES に セクション 49.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加します。このフェーズ中に読み出されるデータはありません。
4. 最後の追加認証データブロックが処理されるまで手順 3 を繰り返します。

注： **ヘッダフェーズは、関連データがなければ（Len(A) = 0であれば）スキップできます。**

関連データ（B1）の最初のブロックは、ソフトウェアで関連データ長でフォーマットする必要があります。

CCMペイロードフェーズ（暗号化または復号化）

このフェーズは、暗号化および復号化と同じでCCMヘッダフェーズの後に実行されます。このフェーズでは、暗号化／復号化したペイロードが AES_DOUTRレジスタに格納されます。実行するシーケンスは、次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを10にセットして、ペイロードフェーズを示します。初期フェーズで設定したMODE[1:0] ビットフィールドは変更しないでください。
2. ヘッダフェーズをスキップした場合、AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
3. これが暗号化する最終データブロックであり、ブロックの平文のサイズが 128bit を下回る場合、ブロックの残り部分を0でパディングします。
4. AES に セクション 49.4.4：1908ページの暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加して、結果を読み出します。
5. 最後から2番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号化されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の2つの手順を適用します。最終ブロックでは、最終ブロックのサイズが16バイト未満である場合に、ペイロードの一部ではないデータを破棄します。

注： **ペイロードフェーズは、ペイロードデータがなければ（Len(P) = 0またはLen(C) = Len(T) であれば）スキップできます。**

暗号文Cを復号化するときには、暗号化されたタグ情報である$LSB_{Len(T)}$(C) を削除します。

CCM最終フェーズ

この最終フェーズでは、AES ペリフェラルがGCM認証タグを生成して AES_DOUTRに格納します。実行するシーケンスは、次のとおりです。

1. AES_CRレジスタのGCMPH[1:0] ビットフィールドを11にセットして、最終フェーズを示します。
2. AES_SRレジスタの計算終了フラグCCFがセットされるまで待ちます。
3. AES_DOUTRレジスタを4回読み出します。この出力がCCM認証タグに相当します。
4. AES_ICR レジスタの CCF ビットをセットして、AES_SRレジスタのCCFフラグをクリアします。
5. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
6. 認証された復号化である場合には、生成された暗号化タグを、暗号文でパディングされた暗号化タグと比較します。

注： **この最終フェーズでは、AES_DOUTR レジスタから読み出されたタグ･データにスワッピングが適用されます。**

ヘッダフェーズから最終フェーズに遷移する際に、AES ペリフェラルを無効にしないでください。そうしないと、誤った結果になります。

アプリケーションは、有効なタグを取得するために、タグの長さで認証タグ出力をマスクする必要があります。

CCMモードでのサスペンド／レジューム動作

ヘッダフェーズおよびペイロードフェーズで**メッセージの処理をサスペンドするには**、次の手順に従います。

1. DMAが使用されている場合には、AES_CRレジスタのDMAINENビットをクリアして、IN FIFO への AES DMA転送を停止します。DMAが使用されていない場合は、AES_SRレジスタのCCFフラグが1にセットされ、現在の計算が完了したことを確認します。
2. ペイロードフェーズで、DMAが使用されていない場合は、AES_DOUTRレジスタを4回読み出して、最後に処理されたブロックを保存します。DMAが使用されている場合には、AES_SRレジ

スタでCCFフラグがセットされるまで待ち、AES_CRレジスタのDMAOUTENビットをクリアして、OUT FIFOからのDMA転送を停止します。

3. AES_ICR レジスタの CCF ビットに1をセットして、AES_SRレジスタのCCFフラグをクリアします。
4. メモリに AES_SUSPxRレジスタを保存します（ここで、xは0から7です）。
5. AES_IVRxレジスタがデータ処理中に初期値から変化するため、それらを保存します。
6. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
7. メモリに現在の AES 設定を保存します（初期化ベクタレジスタ AES_IVRx以外）。アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。
8. DMAが使用されている場合には、DMAコントローラのステータス（ INデータ転送のポインタ、残りバイト数など）を保存します。ペイロードフェーズでは、OUTデータ転送のポインタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMAが使用されている場合には、DMAコントローラを設定して残りのFIFO IN転送を完了させます。ペイロードフェーズでは、残りのFIFO OUT転送もDMAコントローラで設定する必要があります。
2. AES_CRレジスタのENビットをクリアして、AES ペリフェラルを無効にします。
3. メモリに事前に保存したサスペンドレジスタの値を、対応する AES_SUSPxRレジスタに書き戻します（ここで、xは0から7です）。
4. メモリに事前に保存した初期化ベクタレジスタの値を、対応する AES_IVRxレジスタに書き戻します。
5. AES_CRと AES_KEYRxのレジスタに初期設定値を復元します。
6. AES_CRレジスタのENビットをセットして、AES ペリフェラルを有効にします。
7. DMAが使用されている場合には、AES_CRレジスタのDMAINENビット（ペイロードフェーズではDMAOUTENビットも）を1にセットして、AES DMAリクエストを有効にします。

49.4.13 共有キーを使った AES の動作

AES ペリフェラルは、SAESペリフェラルをセキュリティ·コプロセッサとして使用できます。この場合、セキュア·アプリケーションは、堅牢なSAESペリフェラルにキーを準備します。準備ができたら、AES アプリケーションは専用のハードウェアキーのバスを通じて、この準備されたキーをロードできます。

推奨シーケンスについては、以下の説明とSAESペリフェラルのセクション 50.4.10 : 共有キーを使った SAES の動作を参照してください。

1. SAESペリフェラル·アプリケーションでは、共有キーモード（KMOD[1:0] = 10）で共有するキーを一度暗号化します。
2. AESペリフェラルで共有キーが必要になるたびに、アプリケーションは同じ共有キーモード（KMOD[1:0] = 10）を選択して、共有キーをSAESペリフェラルで復号化する必要があります。
3. 共有キーを復号化し、SAES_KEYRxレジスタにロードすると、AESペリフェラルと共有できるようになります。実際、AESペリフェラルに共有キーをロードする必要がある場合、アプリケーションは正しいKEYSIZEをセットし、AES_CR レジスタのKMOD[1:0] = 10を書き込みます。KEYVALIDをクリアすると、キー情報は自動的にハードウェアによってAES_KEYRxに転送されます。AES_SRにBUSYがセットされます。
4. 転送が完了するとBUSYビットはクリアされ、AES_SRレジスタにはKEYVALIDビットがセットされます。BUSYビットがクリアされたときにKEYVALIDがセットされない場合、またはキーエラーフラグ（KEIF）がセットされた場合は、転送中に予期しないイベント（たとえばハードウェアフォールト）が発生したか、SAES_SRのKEYVALIDビットが転送終了前にクリアされていま

す。このようなエラーが発生した場合、SAESとAESの両方のペリフェラルの制御レジスタのIPRSTビットを通じて、SAESキーの共有プロセス全体を最初からやり直す必要があります（詳細はセクション49.4.18を参照）。

この時点で、AESは有効な共有キーで初期化されています。アプリケーションは、KMOD[1:0] を00に設定して、データの処理を進めることができます。

注： **KMOD[1:0] = 10 かつ BUSY = 1 のとき、AES ペリフェラルの KEYSIZE が SAES ペリフェラルのKEYSIZEと異なる場合は、キー共有が失敗して、両方のペリフェラルにKEIFがセットされます。**

49.4.14 AES データレジスタおよびデータスワッピング

データの入出力

128bitのデータブロックは、AES_DINRレジスタ（ビットフィールドDIN[31:0]）に4つの32bitワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）まで連続で書き込むことで、AES ペリフェラルに入力されます。

128bitのデータブロックは、AES_DOUTRレジスタ（ビットフィールドDOUT[31:0]）から4つの32bitワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）まで連続で読み出すことで、AES ペリフェラルから取得されます。

AES_DINRレジスタまたは AES_DOUTRレジスタの32bitのデータワードは、次のいずれかのビッグエンディアン順で整理されます。

- AES_DINRに書き込むワードの最上位バイトは、書き込むワードを保持する4つの隣接したメモリ位置のうち、一番小さいアドレスに配置する必要があります。
- AES_DOUTRから読み出すワードの最上位バイトは、ワードを受け取る4つの隣接したメモリ位置のうち、一番小さいアドレスになります。

AES への入力データブロック書込みにDMAを使用する場合は、入力ブロックの4ワードをビックエンディアン順（一番小さいアドレスに最上位ワード）で連続してメモリに格納する必要があります。セクション 49.4.17：AES DMAインタフェースを参照してください。

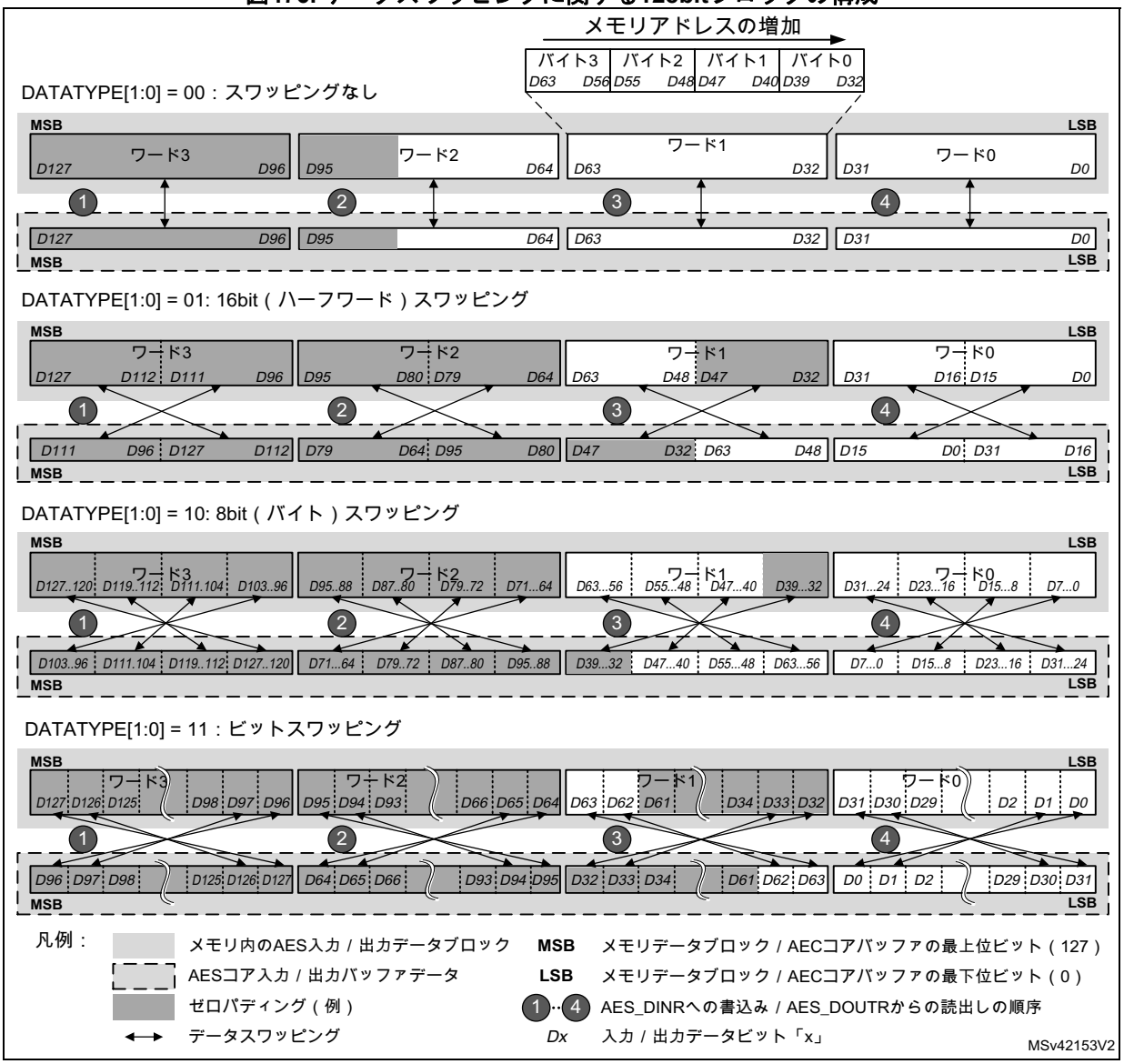
データスワッピング

AES ペリフェラルは、AES処理コアに読み込まれる前の AES_DINRレジスタの入力データワード、および AES_DOUTRレジスタに送られる前のAES処理コアから出力されたデータについて、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしに設定できます。データのタイプによって選択します。たとえば、ASCIIテキストストリームの場合、バイトスワッピングを使用します。

データスワッピングのタイプは、AES_CRレジスタのDATATYPE[1:0] ビットフィールドで選択します。この選択は、AESコアの入力と出力の両方に適用されます。

それぞれのデータスワッピングのタイプについては、図476に AES_DINRレジスタから入力されたデータのAES処理コアの入力バッファデータP127 ~ P0の構成、またはAES処理コアの出力バッファデータP127 ~ P0から AES_DOUTRレジスタで使用できる出力データの構成について示しています。

図476. データスワッピングに関する128bitブロックの構成



注： **AES キーレジスタ（AES_KEYRx）と初期化レジスタ（AES_IVRx）のデータは、選択されたスワップモードの影響を受けません。**

データパディング

図476では、データスワッピング後の0にされたビットによってAESコア入力バッファの最上位ビット側に隣接したゾーンが形成される場合など、0でパディングされるメモリデータブロックの例も示しています。この例では、次を含む入力データブロックのパディングが示されています。

- 48メッセージビット（DATATYPE[1:0] = 01）
- 56メッセージビット（DATATYPE[1:0] = 10）
- 34メッセージビット（DATATYPE[1:0] = 11）

49.4.15 AES キーレジスタ

書込み専用の AES_KEYRxレジスタは、暗号化または復号化キービットフィールドのKEY[127:0] や KEY[255:0] を格納します。各レジスタに書き込むデータは、リトルエンディアン順で最上位バイトをメモリ内の一番大きいアドレスに配置するように整理されます（セキュリティ上の理由で読出しはできません）。

表468 に示すように、キーは8個のレジスタにわたって分散されます。

表468.AES_KEYRxレジスタでのキーエンディアン（128bitまたは256bitのキー長）

AES_KEYR7 [31:0]	AES_KEYR6 [31:0]	AES_KEYR5 [31:0]	AES_KEYR4 [31:0]	AES_KEYR3 [31:0]	AES_KEYR2 [31:0]	AES_KEYR1 [31:0]	AES_KEYR0 [31:0]
-	-	-	-	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]
KEY[255:224]	KEY[223:192]	KEY[191:160]	KEY[159:128]	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]

暗号化または復号化のキーは、AES ペリフェラルが無効であるときに、AES_CRレジスタのENビットをクリアして、これらのレジスタに書き込むことができます。

キーレジスタは、AES_CRレジスタのDATATYPE[1:0] ビットフィールドで制御されるデータスワッピングによる影響を受けません。

キー全体を、AES計算を開始する前に書き込む必要があります。通常のキーモード（KMOD[1:0] = 00）では、AES_KEYRx（KEYSIZE = 0の場合x = 0 ~ 3、KEYSIZE = 1の場合x = 0 ~ 7）レジスタは、必ず昇順または降順のどちらかで書き込む必要があります。

注： **キーのローディングシーケンスを開始すると、BUSYフラグがセットされ、KEYVALIDフラグがクリアされます。KEYSIZEで定義されたビットの量が AES_KEYRxレジスタに転送されると、BUSYはクリアされ、KEYVALIDがセットされ、ENビットが書込み可能になります。エラーが発生すると、BUSYとKEYVALIDがクリアされ、KEIFがセットされます（詳細はセクション 49.4.18 : AES エラー管理を参照）。**
キーモードの詳細については、セクション49.4.13を参照してください。

49.4.16 AES 初期化ベクタレジスタ

4つの AES_IVRxレジスタで初期化ベクタの入力ビットフィールドIVI[127:0] を保持します。各レジスタで読み書きするデータは、リトルエンディアン順で最上位バイトをメモリ内の一番大きいアドレスに配置するように整理されます。レジスタも一番小さいアドレス（AES_IVR0）から一番大きいアドレス（AES_IVR3）の順になります。

ビットフィールドのデータの位は、選択された連鎖モードに基づきます。使用時は、ビットフィールドがAESコアの計算サイクルごとに更新されます。

AES ペリフェラルが有効であるときの AES_IVRxレジスタへの書込み操作は、レジスタの内容に影響を与えません。AES_IVRxレジスタの内容を変更する場合はまず AES_CRレジスタのENビットをクリアする必要があります。

AES_IVRxレジスタを読み出すと、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

AES_IVRxレジスタは、AES_CRレジスタのDATATYPE[1:0] ビットフィールドで制御されるデータスワッピング機能による影響を受けません。

49.4.17 AES DMAインタフェース

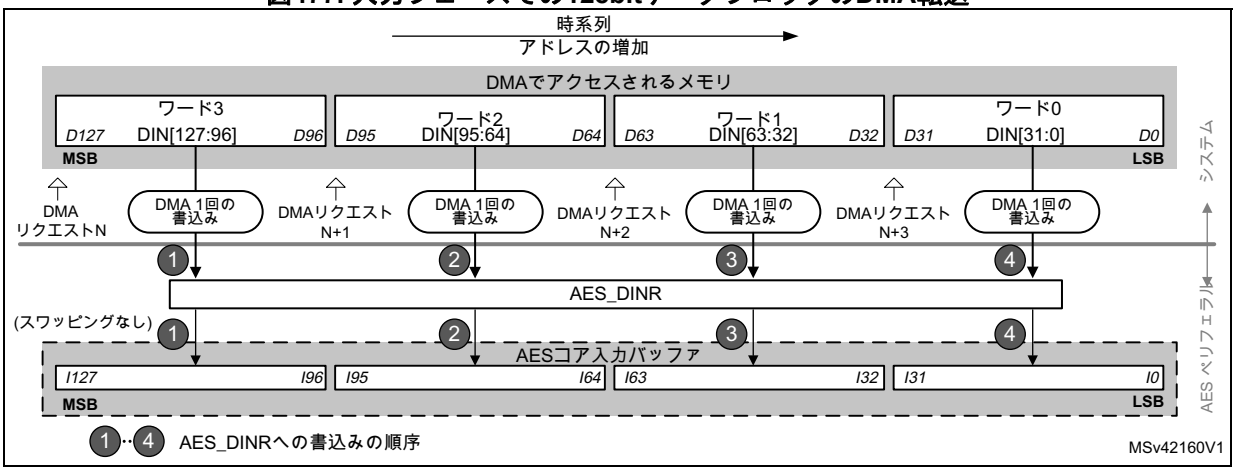
AES ペリフェラルには、DMA（ダイレクトメモリアクセス）コントローラに接続するインタフェースが搭載されています。DMA操作は AES_CRレジスタで制御します。

DMAを用いたデータ入力

AES_CRレジスタのDMAINENビットをセットすると、AES へのDMA書込みが有効になります。図477に示すように、AES ペリフェラルは、入力フェーズ中に AES_DINRレジスタに128bitのブロック（4倍ワード）を書き込む必要があるたびに、DMAリクエストを開始します。

注： **選択されたアルゴリズムとモードに応じて、特別なパディングや暗号文借用が必要となることがあります。たとえば、AES GCM暗号化やAES CCM復号化の場合は、DMA転送に最終ブロックを含まないでください。詳細については、セクション 49.4.4：暗号操作を行うための AES の手順を参照してください。**

図477. 入力フェーズでの128bitデータブロックのDMA転送

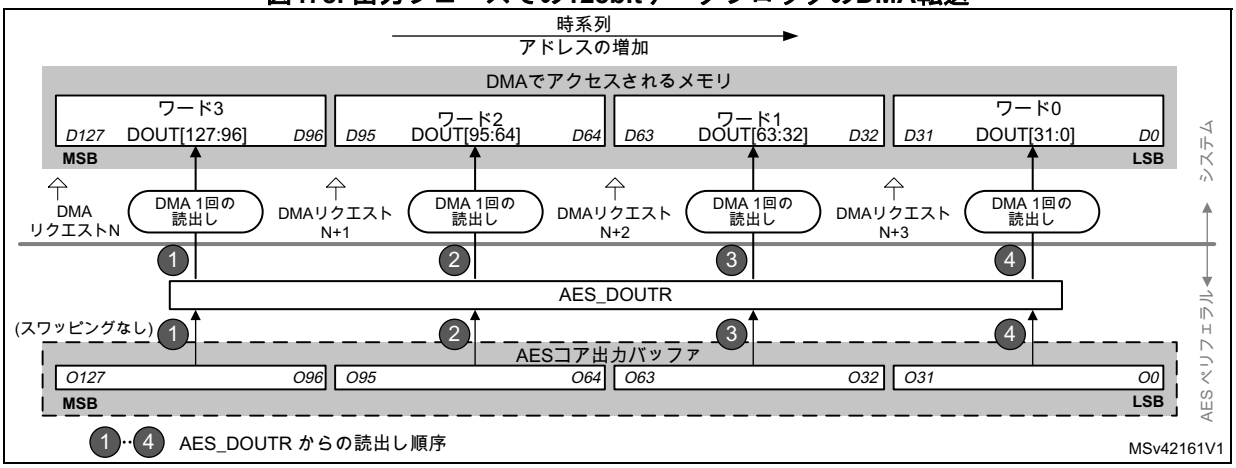


DMAを用いたデータ出力

AES_CRレジスタのDMAOUTENビットをセットすると、AES からのDMA読出しが有効になります。図478に示すように、AES ペリフェラルは、出力フェーズ中に AES_DINR レジスタから128bitのブロック（4倍ワード）を読み出す必要があるたびに、DMAリクエストを開始します。

注： **メッセージサイズによっては、最後のブロックでアプリケーションによって余分なバイトを破棄する必要が出ることがあります。**

図478. 出力フェーズでの128bitデータブロックのDMA転送



異なる動作モードにおけるDMA操作

DMA操作は、モード1（暗号化）またはモード3（復号化）が AES_CRレジスタのMODE[1:0] ビットフィールドで選択されている場合に使用できます。モード2（キー導出）では、AES_KEYRxレジスタはソフトウェアで書き込む必要があり、AES_CRレジスタのDMAINENおよびDMAOUTENビットでDMA転送を有効にしても、このモードでは影響がありません。

DMAシングル·リクエストは、無効にされるまで AES によって生成されます。そのため、128bitデータブロック処理終了時のデータ出力フェーズの後に、AES によって次のデータブロックの新しいデータ入力フェーズ（ある場合）に自動的に切り替えられます。

AES とメモリの間のデータ転送がDMAによって管理されている場合、AES_DOUTRレジスタの読出しは計算フェーズの最後に自動的にDMAによって管理されているため、CCFフラグは使用されません。CCFフラグは、ソフトウェアで管理されるデータ転送に再び遷移したときにのみクリアする必要があります。詳細については、セクション 49.4.4 : 暗号操作を行うための AES の手順のサブセクションデータ追加を参照してください。

49.4.18 AES エラー管理

AES 設定は、AES_CRレジスタのENビットをクリアすることでいつでも変更できます。

読出しエラーフラグ（RDERR）

AES_DOUTRレジスタが予期せず読出しを試みることで、AES_SRレジスタのRDERRフラグとAES_ISRレジスタのRWEIFフラグがセットされ、0が返されます。

RDERRは、計算フェーズ中、または入力フェーズ中にトリガされます。

注： **AES はRDERRエラー検出時に無効にならずに処理を続けます。**

AES_IER レジスタの RWEIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 49.5 : AES 割込みを参照してください。

RDERR およびRWEIF フラグは、AES_ICR レジスタの RWEIE ビットをセットしてクリアします。

書込みエラーフラグ（WDERR）

AES_DINRレジスタが予期せず書込みを試みることで、AES_SR レジスタのWRERR フラグとAES_ISRレジスタのRWEIFフラグがセットされます。AES_DINRへの影響はありません。WRERRは、計算フェーズ中、または出力フェーズ中にトリガされます。

注： **AES はWRERRエラー検出後に無効にならずに処理を続けます。**

AES_IER レジスタの RWEIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 49.5：AES 割込みを参照してください。

WRERR およびRWEIF フラグは、AES_ICR レジスタの RWEIF ビットをセットしてクリアします。

キーエラー割込みフラグ（KEIF）

キーレジスタへのキーのロードに失敗すると、AES_ISR レジスタのKEIF フラグがセットされ、AES_SRレジスタのKEYVALIDビットがクリアされます。

KEIFフラグは、AES_ICR レジスタの対応するビットをクリアします。AES_IER レジスタのKEIEビットがセットされている場合、割込みが生成されます。詳細については、セクション 49.5：AES 割込み を参照してください。

考えられるキーエラーの原因は次のとおりです。

- キー書込みシーケンスエラー:キーレジスタの書込み時に正しくないシーケンスが検出された場合。詳細については、セクション 49.4.15：AES キーレジスタを参照してください。
- キー共有サイズの不一致：KMOD[1:0] = 10 で、SAES ペリフェラルに格納されたKEYSIZE と一致しない AES ペリフェラルにアプリケーションがKEYSIZE情報をセットし、エラーがトリガされた場合。
- キー共有エラー：SAESペリフェラルからのキーレジスタのコピーがAES失敗した場合。詳細については、セクション 49.4.13：共有キーを使った AES の動作を参照してください。

キー共有エラーが発生したら、対応する制御レジスタのIPRSTビットを通じて AES とSAESの両方のペリフェラルをリセットし、その後キー共有シーケンスを最初からやり直します。

注： **キーエラーの発生時には、再設定を行う前にKEIFフラグをクリアしますAES。**

49.5 AES 割込み

AES ペリフェラルによって生成された個別にマスク可能な割込みソースが、次のイベントを通知します。

- 計算完了
- 読出しエラー
- 書込みエラー
- キーエラー

これらのソースは、Arm® Cortex® 割込みコントローラに接続された AES ペリフェラルからの共通の割込み信号に組み合わされます。AES_IER レジスタの対応するイネーブル･ビットを設定/クリアすることで、それぞれ個別に有効/無効にでき、また AES_ICR レジスタの対応するビットを設定することでクリアできます。

各ステータスは、AES_SR および AES_ISR レジスタから読み出すことができます。

割込みソースの概要、イベント･フラグ、イネーブル･ビットについては、表469を参照してください。

表469.AES 割込みリクエスト

項目（割込みの略称）	AES 割込みイベント	イベント･フラグ	イネーブル･ビット	割込みのクリア方法
AES	計算完了フラグ	CCF	CCFIE	CCF をセット(1)
	読出しエラーフラグ	RDERR(2)	RWEIE	RWEIF をセット（1）
	書込みエラーフラグ	WRERR（2）		
	キーエラーフラグ	KEIF	KEIE	KEIFをセット（1）

1. AES_ICR レジスタのビット。
2. AES_SRレジスタのフラグであり、AES_ISRレジスタのフラグRWEIFによって反映されます。

49.6 AES 処理遅延時間

以下の表では、各動作モードで128bitブロックを処理する際の遅延時間の概要を示します。

表470.ECB、CBC およびCTR での処理遅延時間

キー長	動作モード	アルゴリズム	クロックサイクル数
128bit	モード1：暗号化	ECB、CBC、CTR	51
	モード2：キー派生	-	59
	モード3：復号化	ECB、CBC、CTR	51
256bit	モード1：暗号化	ECB、CBC、CTR	75
	モード2：キー派生	-	82
	モード3：復号化	ECB、CBC、CTR	75

表471.GCMおよびCCMでの処理遅延時間（クロックサイクル数）

キー長	動作モード	アルゴリズム	初期フェーズ	ヘッダフェーズ(1)	ペイロードフェーズ（1）	タグフェーズ(1)
128bit	モード1：暗号化／モード3：復号化	GCM	64	35	51	59
		CCM	63	55	114	58
256bit	モード1：暗号化／モード3：復号化	GCM	88	35	75	75
		CCM	87	79	162	82

1. データ挿入では、AHB バスで AES により強制されるウェイトステートを含めることができます（最大3サイクル、通常は1サイクル）。

49.7 AES レジスタ

49.7.1 AES制御レジスタ（AES_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPRST	Res.			Res.		KMOD[1:0]		NPBLB[3:0]				Res.	KEYSIZE	Res.	CHMOD[2]
rw						rw	rw	rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	GCMPH[1:0]		DMAOUTEN	DMAINEN	Res.	Res.	Res.	Res.	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw

ビット31 **IPRST**：AES ペリフェラルのソフトウェアリセット

ビットを設定すると、IPRSTビット自体を除くすべてのレジスタがデフォルト値になり、AES ペリフェラルがリセットされます。したがって、キーに関連するデータはすべて失われます。このため、セキュリティの低いアプリケーションに AES を渡す前に、このビットをセットすることをお勧めします。

このビットは、設定レジスタの書込み中はローである必要があります。

ビット 30:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **KMOD[1:0]**：キーモード選択

このビットフィールドは、アプリケーションによる AES キーの使用方法を定義します。

00：通常のキー

10：SAESコプロセッサからの共有キー

その他：予約済みです。

通常のキーを選択すると、キーレジスタは自由に使用でき、AES_DINおよび AES_DOUTレジスタに固有の用途や保護を適用する必要はありません。

SAESコプロセッサからの共有キーを選択すると、AESペリフェラルはそのキーレジスタを、SAESペリフェラルのキーレジスタに格納されたデータとともに、自動的にロードします。キー値は、AES_SRレジスタのBUSYビットがクリアされ、KEYVALIDがセットされたときに、キーレジスタで使用できます。そうしないと、キーエラーフラグKEIFが AES_ISRレジスタにセットされます。

ビットフィールドは、KEYSIZEが正しい場合、および共有キーの復号化シーケンスがSAESコプロセッサで正常に完了した場合にのみ設定する必要があります。

ビットフィールドの書込みを試みても、AES_SRレジスタのBUSYフラグがセットされている場合や、AES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット23:20 **NPBLB[3:0]**：最終ブロックにおけるパディングバイト数

このビットフィールドで、ペイロードの最終ブロックにおけるパディングバイト数をセットします。

0000：全バイトが有効（パディングなし）

0001：最終ブロックの最下位バイト1個に対するパディング

.....

1111：最終ブロックの最下位バイト15個に対するパディング

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット18 **KEYSIZE**：キー長選択

このビットフィールドは、AES 暗号コアに使用するキーの長さ（ビット単位）を定義します。

0：128

1：256

ビットの書込みを試みても、AES_SRレジスタのBUSYフラグがセットされている場合や、AES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット14:13 **GCMPH[1:0]**：GCMまたはCCMフェーズの選択

このビットフィールドでGCM、GMAC、またはCCMアルゴリズムを選択します。

00：初期フェーズ

01：ヘッダフェーズ

10：ペイロードフェーズ

11：最終フェーズ

GCM、GMAC、またはCCMアルゴリズム以外が（ALGOMODEビットフィールドで）選択されている場合、このビットフィールドの影響はありません。

ビット12 **DMAOUTEN**：DMA出力有効

このビットは、出力フェーズでDMAでのデータ転送を有効／無効にします。

0：無効化

1：イネーブル

このビットがセットされると、出力データフェーズ中に AES によってDMAリクエストが自動生成されます。この機能は、モード1またはモード3がMODE[1:0] ビットフィールドで選択されている場合のみ有効です。モード2（キー導出）では有効ではありません。

ビット11 **DMAINEN**：DMA入力イネーブル

このビットは、入力フェーズでDMAでのデータ転送を有効／無効にします。

0：無効化

1：イネーブル

このビットがセットされると、入力データフェーズ中に AES によってDMAリクエストが自動生成されます。この機能は、モード1またはモード3がMODE[1:0] ビットフィールドで選択されている場合のみ有効です。モード2（キー導出）では有効ではありません。

ビット10:7 予約済みであり、リセット値に保持する必要があります。

ビット16、6:5 **CHMOD[2:0]**：連鎖モード選択

このビットフィールドで、AES 連鎖モードを選択します。

000：電子コードブック（ECB）

001：暗号ブロック連鎖（CBC）

010：カウンタモード（CTR）

011：ガロアカウンタモード（GCM）およびガロアメッセージ認証コード（GMAC）

100：CBC-MAC付きカウンタ（CCM）

上記以外：予約済みです。

ビットフィールドの書込みを試みても、AES_SRレジスタのBUSYフラグがセットされている場合や、AES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット4:3 **MODE[1:0]**：AES 動作モード

このビットフィールドで、AES 動作モードを選択します。

00：モード1：暗号化

01：モード2：キー導出（ECB/CBC復号化のキーの準備）

10：モード3：復号化

11：予約済みです。

ビットフィールドの書込みを試みても、AES_SRレジスタのBUSYフラグがセットされている場合や、AES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット2:1 **DATATYPE[1:0]**：データ型選択

このビットフィールドは、データスワッピングのモードを選択して、AES_DINRレジスタに書き込まれるデータまたは AES_DOUTRレジスタから読み出されるデータのフォーマットを定義します。

00：なし

01：ハーフワード（16bit）

10：バイト（8bit）

11：ビット

詳細については、セクション 49.4.14：AES データレジスタおよびデータスワッピングを参照してください。

ビットフィールドの書込みを試みても、AES_SRレジスタのBUSYフラグがセットされている場合や、AES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット0 **EN**：AES イネーブル

このビットは、AES ペリフェラルを有効または無効にします。

0：無効化

1：イネーブル

このビットをクリアしてからセットすれば、いつでも AES ペリフェラルを再初期化できます。

このビットは、キー準備（モード2）完了時、およびGCM/GMAC/CCM初期フェーズ完了時に、ハードウェアによって自動的にクリアされます。

このビットは、KEYVALID = 0 である限りセットできません。

注：00以外のKMOD[1:0] では、ビットENよりもIPRSTビットを使用してください。

49.7.2 AES ステータスレジスタ（AES_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEYVALID	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
								r				r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **KEYVALID**：キー有効フラグ

このビットは、AES_CRのKEYSIZEによって定義された量のキー情報が AES_KEYxキーレジスタにロードされると、ハードウェアによってセットされます。

0：有効なキー情報がキーレジスタにありません。AES_CRのENビットをセットできません。

1：AES_CRのKEYSIZEによって定義された有効なキー情報がキーレジスタにロードされています。

通常モードでKEYSEL = 0の場合、アプリケーションはキー･レジスタに正しい順序で書き込む必要があります。順序を誤ると AES_ISRレジスタのKEIFフラグがセットされ、KEYVALIDが0のままになります。

KEYSELが0以外の場合、AES によってBUSYフラグが自動的にセットされます。キーの読み込みに成功すると、BUSYフラグがクリアされ、KEYVALIDがセットされます。エラーが発生すると AES_ISRレジスタのKEIFフラグがセットされ、BUSYフラグがクリアされて、KEYVALIDが0のままになります。

KEIFフラグがセットされている場合、アプリケーションは AES_ICRレジスタを通じてこのフラグをクリアする必要があります。クリアしないとKEYVALIDをセットできません。詳細については、KEIFビットの説明を参照してください。

キー読込みの詳細については、セクション 49.4.15：AES キーレジスタを参照してください。

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **BUSY**：ビジー

このフラグは、AES がGCMペイロード**暗号化**フェーズ中にアイドルかビジーかを示します。

0：アイドル

1：ビジー

フラグが「アイドル」を示す場合、優先順位の高いメッセージを処理するために現在のGCM暗号化処理をサスペンドできます。連鎖モードや、ペイロード暗号化以外のGCMフェーズでは、サスペンド処理のためフラグを無視する必要があります。

フラグは、SAESペリフェラルから共有キーを転送するときにセットされます。

ビット2 **WRERR**：書込みエラー

このフラグは、AES_DINRレジスタへの予期しない書込み操作（計算中またはデータ出力フェーズ中）の検出を示します。

0：検出なし

1：検出

フラグはハードウェアによってセットされます。AES_ICR レジスタの RWEIF ビットをセットすると、ソフトウェアによってクリアされます。

フラグ設定時、AES_ICR レジスタの RWEIE ビットで有効になっている場合、割込みが生成されます。

フラグ設定は、AES 動作に影響しません。予期せぬ書込みは無視されます。

ビット1 **RDERR**：読出しエラーフラグ

このフラグは、AES_DOUTRレジスタからの予期しない読出し操作（計算中またはデータ入力フェーズ中）の検出を示します。

0：検出なし

1：検出

フラグはハードウェアによってセットされます。AES_ICR レジスタの RWEIF ビットをセットすると、ソフトウェアによってクリアされます。

フラグ設定時、AES_ICR レジスタの RWEIE ビットで有効になっている場合、割込みが生成されます。

フラグ設定は、AES 動作に影響しません。予期せぬ読出しにはゼロが返されます。

ビット0 **CCF**：計算完了フラグ

このビットは、AES_ISRレジスタのCCFビットを反映します。

49.7.3 AES データ入力レジスタ（AES_DINR）

アドレスオフセット：0x08

リセット値：0x0000 0000

32bit書込みアクセスタイプのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIN[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIN[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **DIN[31:0]**：入力データワード

入力フェーズでこのビットフィールドに4回連続で書き込むことによって、AES ペリフェラルに128bit･ブロックの入力データがすべて書き込まれます。1回目から4回目の書込みまで、対応するデータの重みは [127:96]、[95:64]、[63:32]、[31:0] となります。書込みごとに、32bit入力バッファのデータがDATATYPE[1:0] ビットフィールドに応じたデータ･スワッピング･ブロックによって処理されたうえで、AESコアの128bit入力バッファに書き込まれます。

入力データ･ブロックのデータの位は、AES 動作モードに基づきます。

- **モード1**（暗号化）：平文
- **モード2**（キー導出）：ビットフィールド未使用（AES_KEYRxレジスタを入力に使用）
- **モード3**（復号化）：暗号文

データスワッピング操作の手順は、セクション 49.4.14：1931ページのAES データレジスタおよびデータスワッピングを参照してください。

49.7.4 AES データ出力レジスタ（AES_DOUTR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

32bit読出しアクセス･タイプのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DOUT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOUT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **DOUT[31:0]**：出力データワード

読出し専用ビットフィールドは、32bit出力バッファをフェッチします。計算完了（CCFセット）時にこのビットフィールドを4回連続で読み出すことによって、実質的には ペリフェラルから128bit･ブロックの出力データがすべて読み出されます。出力バッファに達する前に、AESコアによって生成されたデータは、DATATYPE[1:0]ビットフィールドに応じたデータスワッピングブロックによって処理されます。

1回目から4回目までの読出し操作でのデータの重みは [127:96]、[95:64]、[63:32]、[31:0] になります。

出力データ･ブロックのデータの位は、AES 動作モードに基づきます。

- **モード1**（暗号化）：暗号文
- **モード2**（キー導出）：ビットフィールド未使用
- **モード3**（復号化）：平文

データスワッピング操作の手順は、セクション 49.4.14：1931ページのAES データレジスタおよびデータスワッピングを参照してください。

49.7.5 AES キーレジスタ0（AES_KEYR0）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[31:0]**：暗号化キー、ビット [31:0]

この書込み専用ビットフィールドには、動作モードに応じてAES暗号化または復号化キーのビット [31:0] が含まれます。

- **モード1**（暗号化）、**モード2**（キー導出）：ビットフィールドに書き込む値は暗号化キーです。
- **モード3**（復号化）：ビットフィールドに書き込む値は、復号化に使用する前に導出される暗号化キーです。

AES_KEYRxレジスタに書き込めるのは、KEYSIZEの値が正しく、AES ペリフェラルが無効化（AES_CRレジスタのENビットがクリア）されているときのみです。AES_SRレジスタのKEYVALIDビットに示すとおり、特殊な書込みシーケンスも必要です。KMOD[1:0] = 10（共有キー）の場合、キーは SAESペリフェラルからAES_KEYRxレジスタに直接ロードされることに注意してください（そのため、キーレジスタへの書込みは無視され、KEIFがセットされます）。

詳細については、セクション 49.4.15： 1933ページのAES キーレジスタを参照してください。

49.7.6 AES キーレジスタ1（AES_KEYR1）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[63:32]**：暗号化キー、ビット [63:32]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.7 AES キーレジスタ2（AES_KEYR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[95:80]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[79:64]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[95:64]**：暗号化キー、ビット [95:64]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.8 AES キーレジスタ3（AES_KEYR3）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[127:112]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[111:96]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[127:96]**：暗号化キー、ビット [127:96]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.9 AES 初期化ベクタレジスタ0（AES_IVR0）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[31:0]**：初期化ベクタ入力、ビット [31:0]

IVI[127:0] フィールドの説明については、セクション 49.4.16：1933ページのAES 初期化ベクタレジスタを参照してください。

初期化ベクタは、ECB以外の連鎖モードでのみ使用します。

AES_IVRxレジスタは、AES ペリフェラルが無効である場合のみ書き込むことができます。

49.7.10 AES 初期化ベクタレジスタ1（AES_IVR1）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[63:32]**：初期化ベクタ入力、ビット [63:32]

IVI[128:0] ビットフィールドの説明については、AES_IVR0レジスタを参照してください。

49.7.11 AES 初期化ベクタレジスタ2（AES_IVR2）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[95:80]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[79:64]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[95:64]**：初期化ベクタ入力、ビット [95:64]

IVI[128:0] ビットフィールドの説明については、AES_IVR0レジスタを参照してください。

49.7.12 AES 初期化ベクタレジスタ3（AES_IVR3）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[127:112]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[111:96]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[127:96]**：初期化ベクタ入力、ビット [127:96]

IVI[128:0] ビットフィールドの説明については、AES_IVR0レジスタを参照してください。

49.7.13 AES キーレジスタ4（AES_KEYR4）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[159:144]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[143:128]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[159:128]**：暗号化キー、ビット [159:128]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.14 AES キーレジスタ5（AES_KEYR5）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[191:176]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[175:160]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[191:160]**：暗号化キー、ビット [191:160]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.15 AES キーレジスタ6（AES_KEYR6）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[223:208]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[207:192]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[223:192]**：暗号化キー、ビット [223:192]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

49.7.16 AES キーレジスタ7（AES_KEYR7）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[255:240]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[239:224]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[255:224]**：暗号化キー、ビット [255:224]
KEY[255:0] ビットフィールドの説明については、AES_KEYR0レジスタを参照してください。

注： 4～7のキーレジスタは、256bitのキー長を選択した場合のみ使用されます。128bitのキー長を選択した場合は有効ではありません（この場合、0～3のキーレジスタのみが使用されます）。

49.7.17 AES サスペンドレジスタ（AES_SUSPxR）

アドレスオフセット：0x040 + 0x4 * x（x = 0から7）

リセット値：0x0000 0000

現在のタスクのAES処理が優先順位の高いタスクを処理するためにサスペンドされる場合、AESプロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。
サスペンド時、ソフトウェアは AES_SUSPxRレジスタ（xは0～7）の内容を読み出し、メモリに保存してから優先順位の高いタスクにAESプロセッサを使用します。完了時、ソフトウェアは保存した内容を、対応するサスペンドレジスタに復元してから、元のタスクを再開します。

注： **これらのレジスタは、GCM、GMACまたは CCM連鎖モードが選択されているときにのみ使用します。**

これらのレジスタは AES が有効である場合のみ読み出すことができます。AES が無効になっている状態で、これらのレジスタを読み出すと、0x0000 0000が返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SUSP[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUSP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SUSP[31:0]**：AES サスペンド
サスペンド動作時、対応する AES_SUSPxRレジスタのこのビットフィールドでは内部 AES レジスタのいずれかの値を取得します。

49.7.18 AES 割込み有効レジスタ（AES_IER）

アドレスオフセット：0x300

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIE	RWEIE	CCFIE
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIE**：キーエラー割込みイネーブル
このビットは、KEIF（キーエラーフラグ）がセットされている場合に AES 割込み生成を有効化または無効化（マスク）します。
0：無効化（マスク）
1：有効化（未マスク）

ビット1 **RWEIE**：読出しまたは書込みエラー割込みイネーブル

このビットは、RWEIF（読出しおよび/または書込みエラーフラグ）がセットされている場合に AES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

ビット0 **CCFIE**：計算完了フラグ割込みイネーブル

このビットは、CCF（計算完了フラグ）がセットされている場合に AES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

49.7.19 AES割込みステータスレジスタ（AES_ISR）

アドレスオフセット：0x304

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	RWEIF	CCF
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIF**：キーエラー割込みフラグ

この読出し専用ビットは、キー情報がキーレジスタへのロードに失敗した場合に、ハードウェアによってセットされます or key register usage is forbidden。

0：キーエラーは検出されていません。

1：キー情報がキーレジスタへのロードに失敗しました。, or key register use is forbidden

AES_ICRレジスタの対応するビットを設定すると、AES_IERレジスタのKEIEビットがセットされている場合は、KEIFがクリアされ、割込みが生成されます。

KEIFは、次のエラーによってトリガされます。

–AES_KEYRx レジスタへの書込みが、正しい順序で行われていません。（KEYSIZE = 0 の場合、AES_KEYR0、AES_KEYR1、AES_KEYR2、AES_KEYR3レジスタの順、またはその逆順。KEYSIZE = 1の場合、AES_KEYR0、AES_KEYR1、AES_KEYR2、AES_KEYR3、AES_KEYR4、AES_KEYR5、AES_KEYR6、AES_KEYR7、またはその逆順）。

KEIFはアプリケーションソフトウェアでクリアする必要があり、そうしないとKEYVALIDをセットできません。

ビット1 **RWEIF**：読出しまたは書込みエラー割込みフラグ

この読出し専用ビットは、AES_SRレジスタのRDERRまたはWRERRエラーフラグがセットされている場合に、ハードウェアによってセットされます。

0：読出しまたは書込みエラーは検出されていません。

1：読出しまたは書込みエラーが検出されています（詳細は AES_SRレジスタを参照）。

RWEIFビットは、アプリケーションが AES_ICRレジスタの対応するビットをセットしたときにクリアされます。AES_IERレジスタのRWEIEビットが事前にセットされている場合、割込みが生成されます。

このフラグは、キー派生モードが選択されている場合に意味を持ちません。

ビット0 **CCF**：計算完了フラグ

このフラグは計算が完了したかどうかを示します。

0：未完了

1：完了

フラグは、計算完了時にハードウェアによってセットされます。AES_ICR レジスタの CCF ビットをセットすると、このフラグがソフトウェアによってクリアされます。

フラグ設定時、AES_IER レジスタのCCFIEビットで有効になっている場合、割込みが生成されます。

フラグはDMAOUTENビットが0の場合のみ有効です。DMA_ENが1の場合、ハイのままとなります。

49.7.20 AES 割込みクリアレジスタ（AES_ICR）

アドレスオフセット：0x308

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	RWEIF	CCF
													w	w	w

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIF**：キーエラー割込みフラグクリア

このビットをセットすると、AES_ISRレジスタのKEIFステータスビットがクリアされます。

ビット1 **RWEIF**：読出しまたは書込みエラー割込みフラグクリア

このビットをセットすると、AES_ISRレジスタのRWEIFステータスビットと、AES_SRレジスタのRDERRとWRERRの両方のフラグがクリアされます。

ビット0 **CCF**：計算完了フラグクリア

このビットをセットすると、AES_SRと AES_ISRのレジスタのCCFステータスビットがクリアされます。

49.7.21 AES レジスタマップ

表472.AESレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	AES_CR	IPRST	Res.	Res.	Res.	Res.	Res.	KMOD[1]	KMOD[0]	NPBLB[3:0]				Res.	KEYSIZE	Res.	CHMOD[2]	Res.	GCMPH[1:0]		DMAOUTEN	DMAINEN	Res.	Res.	Res.	Res.	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	リセット値	0						0	0	0	0	0	0		0		0		0	0	0	0					0	0	0	0	0	0	0
0x004	AES_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEYVALID	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
	リセット値																									0				0	0	0	0
0x008	AES_DINR	DIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	AES_DOUTR	DOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	AES_KEYR0	KEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	AES_KEYR1	KEY[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	AES_KEYR2	KEY[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	AES_KEYR3	KEY[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	AES_IVR0	IVI[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	AES_IVR1	IVI[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	AES_IVR2	IVI[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	AES_IVR3	IVI[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	AES_KEYR4	KEY[159:128]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	AES_KEYR5	KEY[191:160]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	AES_KEYR6	KEY[223:192]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	AES_KEYR7	KEY[255:224]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	AES_SUSP0R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表472.AESレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x044	AES_SUSP1R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	AES_SUSP2R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	AES_SUSP3R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	AES_SUSP4R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	AES_SUSP5R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	AES_SUSP6R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	AES_SUSP7R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060-0x2FF	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x300	AES_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIE	RWEIE	CCFIE
	リセット値																														0	0	0
0x304	AES_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	RWEIF	CCF
	リセット値																														0	0	0
0x308	AES_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	RWEIF	CCF
	リセット値																														0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

50 セキュアAESコプロセッサ（SAES）

このセクションは、STM32U545/585/5Ax/5Gxデバイスにのみ適用されます。

50.1 概要

セキュアAESコプロセッサ（SAES）は、連邦情報処理規格公報（FIPS）197によって規定されている高度暗号化標準（AES：Advanced Encryption Standard）に完全に準拠したアルゴリズムと実装を使用してデータを暗号化および復号化します。これには、差動電力解析（DPA）を含むサイドチャネル攻撃（SCA）に対する保護、認定済みのSESIP、およびPSAセキュリティ保証レベル3などが組み込まれています。

ペリフェラルは、128または256bitのキーサイズの ECB、およびCBC連鎖モードに加え、ハードウェア秘密キーの暗号化／復号化（ラップキーモード）や高速 AES ペリフェラルとのキー共有（共有キーモード）などの特別なモードをサポートします。

SAES には、ハードウェアでSTM32のハードウェア秘密キー（ブート･ハードウェア･キー BHK および導出されたハードウェア･ユニーク･キー DHUK）をロードできる場合があります。これらはアプリケーションで使用できますが、読み出すことはできません。

SAES は、32bitのシングルアクセスでのみアクセス可能なAMBA AHB スレーブペリフェラルです。その他のアクセスタイプは AHB エラーを生成し、32bit書込み以外では、レジスタの内容が破損する可能性があります。

ペリフェラルは、入力および出力データのDMAシングル転送をサポートします（2つのDMAチャネルが必要です）。また、真性乱数発生器（TRNG）および AES ペリフェラルとハードウェア的に連動しています。

50.2 SAES の主な特徴

- 2001年11月発効のNIST『高度暗号化標準（AES）、FIPS公報197』に準拠
- 128bitデータブロックの処理
- 128bitおよび256bitの暗号キー長に対応
- 次の複数の連鎖モードで暗号化と復号：
  - 電子コードブック（ECB）モード
  - 暗号ブロック連鎖（CBC）モード
- 128bit または 256bit キーをそれぞれ使用して 1 個の 128bit ブロックのデータを処理する場合はECBモードで 528 または 743 クロックサイクルの遅延
- ECB/CBC復号の最終ラウンドのキーの計算に使用する内蔵ラウンド･キー･スケジューラ
- 32bit･ワードのシングル･アクセスでのみアクセス可能なAMBA AHB スレーブ･ペリフェラル
- 暗号化キーを格納するための256bit書込み専用レジスタ（32bitレジスタ8個）
  - 一緒にXOR演算できる2つのハードウェア秘密キー（BHK、DHUK）のハードウェアのロード（オプション）
- 初期化ベクタを格納するための128bitレジスタ（32bitレジスタ4個）
- データ入力および出力用の32bitバッファ
- 2つのチャネル（1チャネルは入力データ用、1チャネルは処理済みデータ用）を使用したシングル転送ダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御
- 1、8、16、32bitデータをサポートするデータスワッピングロジック
- SAES がより優先順位の高い別メッセージを処理する必要がある場合は、ソフトウェアでメッセージをサスペンドしてから元のメッセージのレジュームが可能
- キーのセキュリティコンテキスト強制
- ハードウェア秘密キー暗号化／復号（ラップキーモード）
- サイドチャネル攻撃（SCA：side-channel attacks）に対する保護、差分電力解析（DPA：differential power analysis）、認定SESIPおよびPSAセキュリティ保証レベル3を含む
- SAESで制御された、高速 AES ペリフェラル（共有キーモード）とのハードウェアキーの共有

50.3 SAES の実装

デバイスには、1つのSAESと1つのAESペリフェラルが、次の表に示すように実装されています。

表473. AES/SAESの機能

AES/SAESのモード／機能(1)	AES	SAES
ECB、CBC連鎖（チェーン動作）	X	X
CTR、CCM、GCM連鎖	X	-
AES 128bit ECB暗号化（サイクル単位）	51	528
DHUKおよびBHKキー選択	-	X
サイドチャネル攻撃耐性	-	X
SAESとAESの共有キー	X	

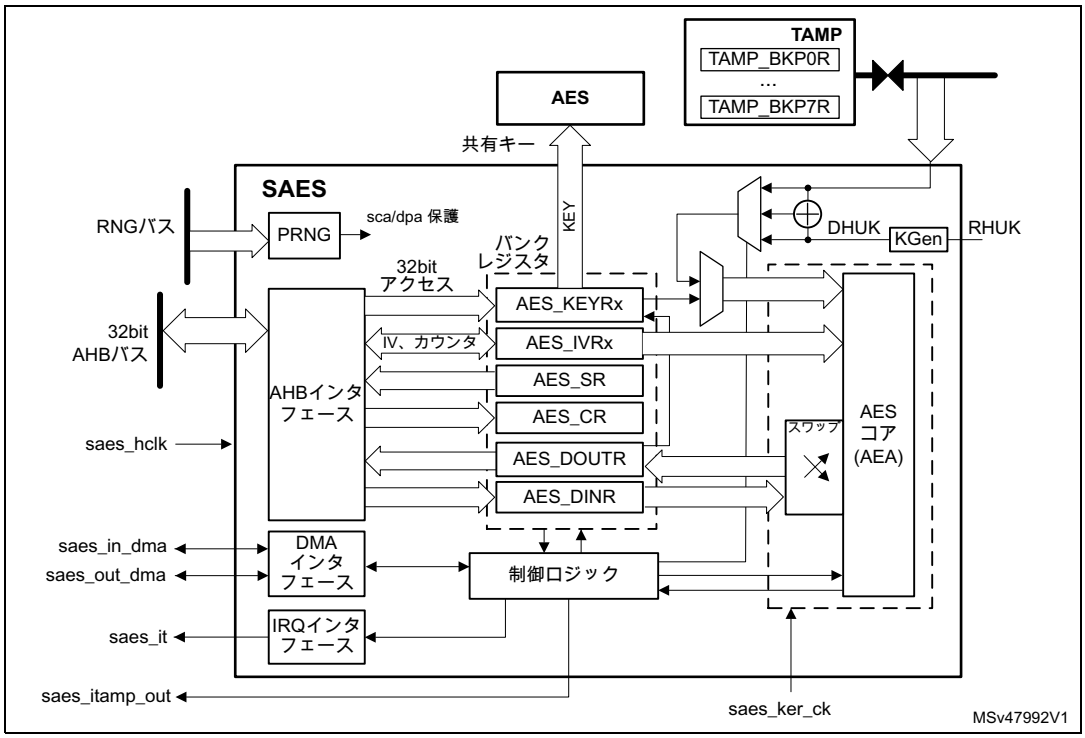
1. X：サポートされています。

50.4 SAES機能詳細

50.4.1 SAES ブロック図

図479に SAES のブロック図を示します。

図479. SAES ブロック図



注： **AESは AES ペリフェラルを表わします。saes_ker_ckはrcc_shsi_ckクロック信号を表します。**

50.4.2 SAES 内部信号

表474 には、SAES ペリフェラルにインタフェース接続するユーザ関連内部信号を示しています。

表474.SAES 内部入力／出力信号

信号名	信号タイプ	説明
saes_hclk	入力	AHB バスクロック
saes_it	出力	SAES 割込みリクエスト
saes_in_dma	入力／出力	入力DMAシングル･リクエスト／確認応答
saes_out_dma	入力／出力	出力DMAシングル･リクエスト／確認応答
saes_itamp_out	出力	予期しないハードウェアフォールトが発生したときにトリガされる、TAMPへのタンパイベント信号（XORされます）。この信号がトリガされると、SAES はキーレジスタを自動的にクリアします。SAES を再び使用可能にするには、リセットが必要です。
rcc_shsi_ck	入力	RCCからの専用SHSIクロック

表474.SAES 内部入力／出力信号 （続き）

信号名	信号タイプ	説明
RHUK	入力	256bitのルートハードウェア固有キー（不揮発性、デバイスごとに固有、およびソフトウェアに対して秘密）。導出ハードウェア固有キー（DHUK）を内部で計算するために使用されます。
BHK(1)	入力	改ざん防止のセキュアバックアップレジスタに格納されている256bitのブートハードウェアキー（BHK）で、ブート中にセキュアコードによって書き込まれます。このキーは、一度書き込まれると、製品をリセットするまではアプリケーションから読み書きができなくなります。

1. TAMPペリフェラル内の一連のバックアップレジスタに接続されており、アプリケーションソフトウェアによって書き込まれ、その後、読取り／書込みがロックされます（詳細については セクション50.4.12 を参照）。

50.4.3 SAES 暗号コア

概要

SAES 暗号コアは、次のコンポーネントで構成されています。

- AESコア･アルゴリズム（AEA）
- キー入力
- 初期化ベクタ（IV）入力
- 連鎖アルゴリズムロジック（XOR）

SAES コアは128bitまたは256bitのキー長を使用して128bit･データ･ブロック（4ワード）で動作します。連鎖モードに応じて、SAESは0個または1個の128bit初期化ベクタ（IV）を必要とします。

SAES には、次の動作モードがあります。

- **モード1**：

  SAES_KEYRxレジスタに格納されたキーを使用して平文を暗号化するか、専用のハードウェアバスを使用して読み出します。

- **モード2:**

  ECBまたはCBCの復号化キーを準備します。ECBまたはCBCの連鎖モードでモード3を選択する前に使用する必要があります。復号化のために準備されたキーは、SAES_KEYRxレジスタに自動的に格納されます。これで、SAES ペリフェラルをモード3に切り替えて、データ復号化を実行する準備ができます。

- **モード3**：

  SAES_KEYRxレジスタに格納されたキーを使用して暗号文を復号化します。ECBおよびCBCの連鎖モードを選択した場合、モード2で事前にキーを準備する必要があります。

注： **モード2 は、ECBおよびCBCの復号化を実行する場合のみ使用します。**

動作モードは、SAES_CRレジスタのMODE[1:0] ビットフィールドをプログラムして選択します。この操作は、SAES ペリフェラルが無効である場合のみ実行できます。

特殊キー動作は、SAES_CRレジスタのKMOD[1:0] ビットフィールドを使用して選択します。詳細については、セクション50.4.10およびセクション50.4.9を参照してください。

標準的データ処理

SAES の標準的な使用方法についてはセクション 50.4.4：1958ページの暗号操作を行うためのSAESの手順に説明があります。

注： **IVIビットフィールドを除き、中間AEAステージの出力が暗号境界の外に漏れることは決してありません。**

連鎖モード

次の連鎖モードが SAES でサポートされており、SAES_CRレジスタのCHMOD[2:0] ビットフィールドで選択します。

- 電子コードブック（ECB）
- 暗号ブロック連鎖（CBC）

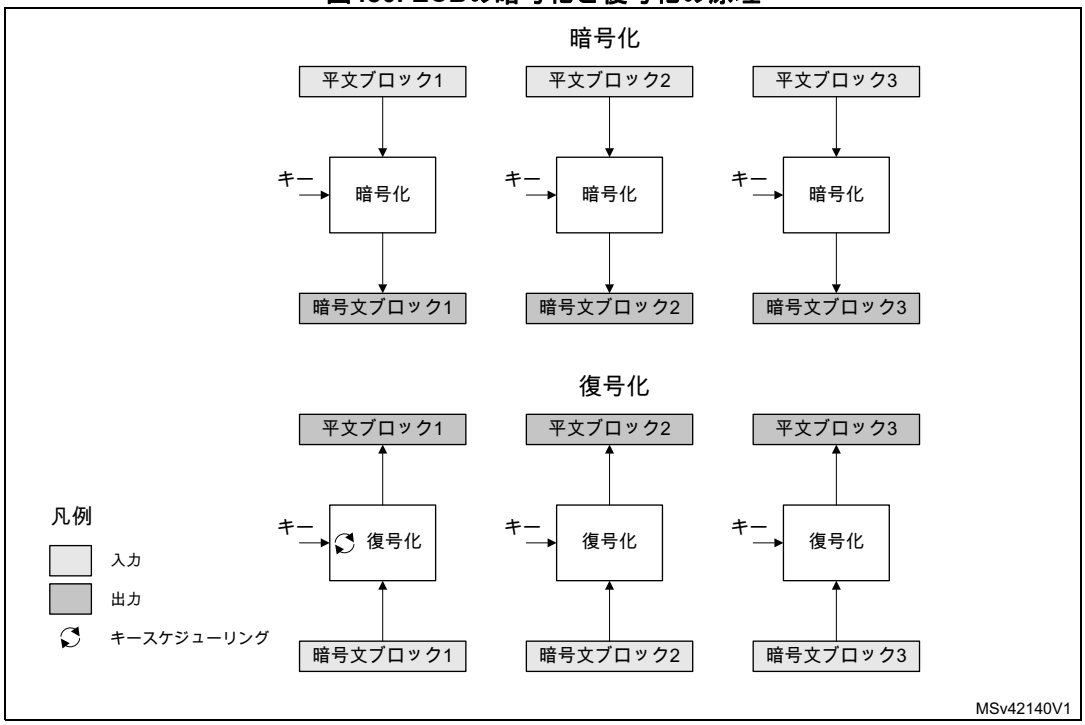
注： **連鎖モードは、SAES が無効である（SAES_CRレジスタのENビットがクリアされている）場合のみ変更できます。**

各 SAES 連鎖モードの原理については、以降の項目に記載されています。

詳細については、セクション 50.4.8 : SAES 基本連鎖モード（ECB、CBC）以降の専用セクションに示されています。

電子コードブック（ECB）モード

図480. ECBの暗号化と復号化の原理

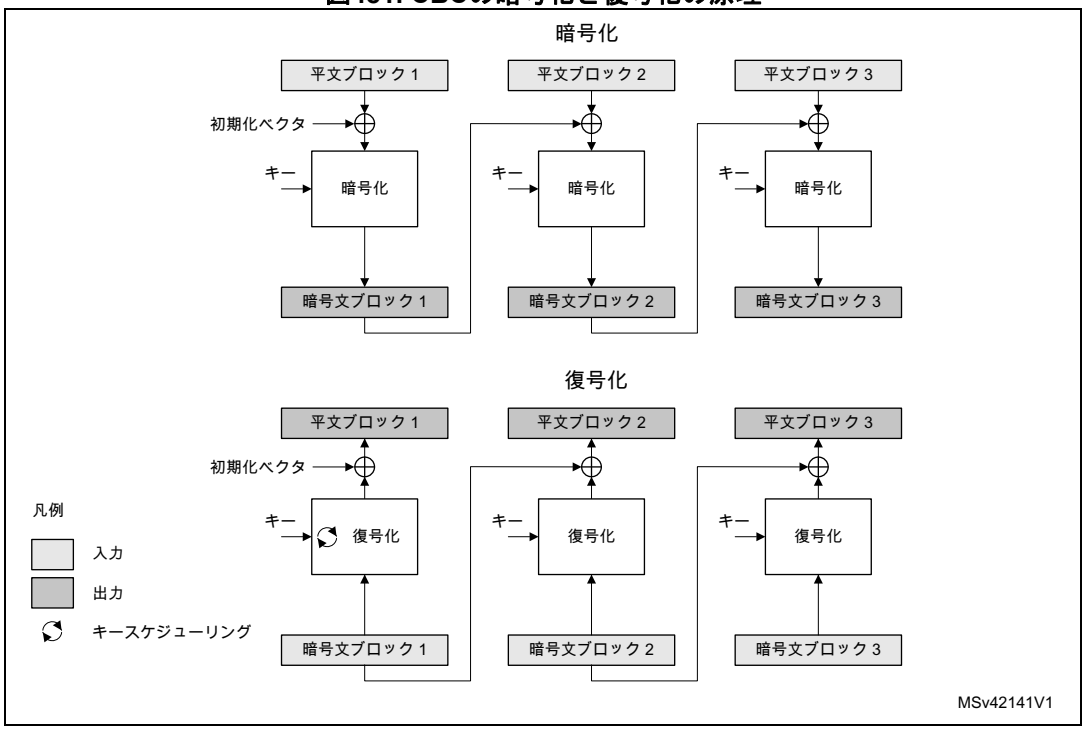


ECBは最も単純な動作モードです。連鎖操作も、特別な初期化ステージもありません。メッセージはブロックに分割され、各ブロックが個別に暗号化または復号されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

暗号ブロック連鎖（CBC）モード

図481. CBCの暗号化と復号化の原理



CBCモードでは、各ブロックの出力が次のブロックの入力で連鎖されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクタが使用されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

50.4.4 暗号操作を行うためのSAES の手順

はじめに

標準的な暗号操作を以下に説明します。詳細については、セクション 50.4.8 : SAES 基本連鎖モード（ECB、CBC）以降のセクションに示されています。

SAES の初期化

SAES を初期化するには、まず SAES_CRレジスタのENビットをクリアして無効化します。次に、任意の順番で次の手順を実行します（KEYSIZEを除く）。

- SAES_SRのBUSY = 0（RNGによる乱数の取得が実行されていない状態）およびSAES_ISRのRNGEIF = 0（乱数の取得エラーがフラグされている状態）であることを確認します。
- SAES_CRレジスタのMODE[1:0] ビットフィールドをプログラムして、SAES モードを設定します。
  - 暗号化の場合は、モード1（MODE[1:0] = 00）を選択します。
  - 復号化の場合は、セクション 50.4.8 : SAES 基本連鎖モード（ECB、CBC）に示すように、モード2（MODE[1:0] = 01）、モード3（MODE[1:0] = 10）の順に選択します。
- SAES_CRレジスタのCHMOD[2:0] ビットフィールドをプログラムして、連鎖モードを選択します。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0] 値については、セクション50.4.9 およびセクション50.4.10を参照してください。
- SAES_CRレジスタのDATATYPE[1:0] ビットフィールドでデータ型（1、8、16、32bit）を設定します。
- 必要であれば（CBC 連鎖モードの場合など）、初期化ベクタを SAES_IVRxレジスタに書き込みます。
- SAES_CRレジスタのKEYSIZEビットフィールドでキーサイズ（128bitまたは256bit）を設定します。この手順は、キーレジスタへの書込みを行う前に行うか、KEYSELを使用して異なるキーソースを選択する必要があります。キーを別のセキュリティコンテキスト（異なるセキュア属性）と共有してはならない場合は、SAES_CRレジスタのKEYPROTビットもセットする必要があります。
- 対称キーを SAES_KEYRxレジスタ（キーサイズに応じて4個か8個のレジスタ）に書き込みます。あるいは、0x0とは異なる値をKEYSEL[2:0] にセットすることによって、キーレジスタとは異なるキーソースを選択します。詳細については、セクション50.4.12を参照してください。

注： **SAES は、KEYSIZEによって定義されたキー情報が SAES_KEYRxにロードされている場合、SAES_SRのKEYVALIDをセットします。**

データ追加

このセクションでは、KMOD[1:0] = 00のとき、処理するデータのサイズが128bitの倍数ではない場合に、処理するためにデータを追加するさまざまな方法について説明します。その他のKMOD[1:0] 値については、セクション50.4.10 およびセクション50.4.9を参照してください。

ポーリングでのデータ追加

この方法では、フラグポーリングを使用して次のシーケンスでデータ追加を制御します。

1. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
2. ペイロードが完全に処理されるまで次のサブシーケンスを繰り返します。
   a) SAES_DINRレジスタに4つの入力データワードを書き込みます。
   b) ステータスフラグCCFが SAES_SRでセットされるまで待って、SAES_DOUTRレジスタから4つのデータワードを読み出します。
   c) SAES_ISR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
   d) 処理したばかりのデータブロックがメッセージの最後から2番目のブロックであり、処理する最終ブロックの有効データが128bitを下回る場合は、セクション 50.4.6 : SAES 暗号文借用およびデータパディングを参照してください。
3. これは最終ブロックであるため、セクション 50.4.6 : SAES 暗号文借用およびデータパディング、の指示に従い、その後 SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。

注： **ウェイトサイクルは SAES_DINRレジスタに対する連続する2つの書込みの間に最大3つまで自動的に挿入され、AES コプロセッサにキーを送信できます。**

割込みを使用したデータ追加

この方法は、SAES ペリフェラルからの割込みを使用して次のシーケンスでデータ追加を制御します。

1. SAES_IER レジスタのCCFIEビットをセットして、SAES からの割込みを有効にします。
2. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
3. まず、SAES_DINRレジスタに4つの入力データワードを書き込みます。
4. SAES 割込みサービスルーチンでデータを処理します。割込み時には次の手順に従います。
   a) SAES_DOUTRレジスタから4つの出力データワードを読み出します。
   b) SAES_ISR レジスタの CCF ビットをセットして、CCFフラグ、つまり保留中の割込みをクリアします。
   c) 処理したばかりのデータブロックがメッセージの最後から2番目のブロックであり、処理する最終ブロックの有効データが128bitを下回る場合は、セクション 50.4.6：SAES 暗号文借用およびデータパディング *を参照してください*。次に、4e) に進みます。
   d) 処理したばかりのデータブロックがメッセージの最終ブロックだった場合、必要に応じてセクション 50.4.6：SAES 暗号文借用およびデータパディングに従い、次にSAES_CRレジスタのENビットをクリアして SAES ペリフェラルを無効にし、割込みサービスルーチンを中止します。
   e) SAES_DINRレジスタに次の4つの入力データワードを書き込み、割込みサービスルーチンを中止します。

注： **SAES は連続した読出し操作または書込み操作の間の遅延を許容するため、2つの SAES 計算の間に別のペリフェラルからの割込みがある場合などに受け入れることができます。**

DMAを使用したデータ追加

この方法では、すべての転送と処理がDMAおよび SAES によって管理されます。この方法を使用するには、次の手順に従います。

1. 処理する最終ブロックが128bitを下回る場合は、最後の4ワードのデータブロックを準備するためにセクション 50.4.6：SAES 暗号文借用およびデータパディングを参照してください。
2. 処理するデータをメモリから SAES ペリフェラル入力へ、処理したデータを SAES ペリフェラル出力からメモリに転送するようDMAコントローラを設定します（セクション 50.4.14：SAES DMAインタフェース に記載）。転送完了時に割込みを生成するようDMAコントローラを設定します。
3. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
4. SAES_CRレジスタのDMAINENビットとDMAOUTENビットをセットして、DMAリクエストを有効にします。
5. 転送完了を示すDMA割込みが発生したら、メモリから SAES 処理済みデータを取得します。

注： **SAES_DOUTR レジスタの読出しは計算フェーズの最後にソフトウェア操作なしで自動的に DMA によって管理されるため、この方法ではCCFフラグを使用しません。**

50.4.5 SAES 復号ラウンド･キーの準備

AESラウンド･キーの生成には、内部キー･スケジュールを使用します。AES暗号化のラウンド0のキーは、キー･レジスタに格納されているキーです。AES復号は、最後のラウンド･キーを使用して開始する必要があります。暗号化キーはメモリに保存されているので、復号キーを得るには特別なキー･スケジューリングを実行する必要があります。

SAES_CRのMODE[1:0] ビットフィールドを01に設定してモード2（キー処理のみ）を選択してから、MODE[1:0] を10（モード3、復号のみ）に設定して復号を進める方法が推奨されます。モード2の使用方法は、以下のとおりです。

1. SAES_SRのBUSY = 0 （RNGによる乱数フェッチが実行されていない状態）を確認します。
2. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
3. SAES_CRのMODE[1:0] ビットフィールドを01にセットして、モード2を選択します。このキー導出モードは選択された連鎖アルゴリズムとは無関係なので、この場合、CHMOD[2:0] ビットフィールドは重要ではありません。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0] 値による復号化については、セクション50.4.10およびセクション50.4.9を参照してください。
4. SAES_CRレジスタのKEYSIZEビットでキー長を128bitまたは256bitにセットします。キーを別のセキュリティコンテキスト（異なるセキュア属性）と共有してはならない場合は、SAES_CRレジスタのKEYPROTビットもセットする必要があります。
5. SAES_KEYRxレジスタ（128または256bit）に暗号化キーを書き込むか、KEYSEL[2:0] を通じてキーレジスタとは異なるキーソースを選択します。詳細については、セクション 50.4.12 : SAES キーレジスタを参照してください。SAES_IVRxレジスタへの書込みは無効です。
6. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
7. SAES_SRレジスタでCCFフラグがセットされるまで待ちます。
8. CCFフラグをクリアします。導出されたキーはAESコアにあり、すぐに復号化に使用できます。

注： **導出キーが使用できるようになると、SAES はハードウェアによって無効にされます。**

導出キーの計算を再開するには、手順 5、6、7、8を繰り返します。

注： **キー準備の操作は、キーサイズ（128bitまたは256bit）に応じて 200 または 324 クロックサイクル続きます。**

50.4.6 SAES 暗号文借用およびデータパディング

サイズがブロックサイズ（128bit）の倍数ではないメッセージを管理するためにECBまたはCBCモードで SAES を使用するときには、NIST 特別公報800-38A『ブロック暗号の推奨動作モード : CBCモード用暗号文借用の3つの変種』に記載されているもののような、暗号文借用技術を使用します。SAES ペリフェラルはこのような技術をサポートしていないため、アプリケーションは最後から2番目のブロックのデータを使用して、入力データの最終ブロックを完成させる必要があります。

注： **暗号文借用技術は、この参考マニュアルには記載されていません。**

注： **パディングデータは SAES_CRレジスタのDATATYPE[1:0] フィールドに従って、通常のデータと同様の方法で入れ替えられます（詳細は、セクション 50.4.11 : SAES データレジスタおよびデータスワッピングを参照してください）。**

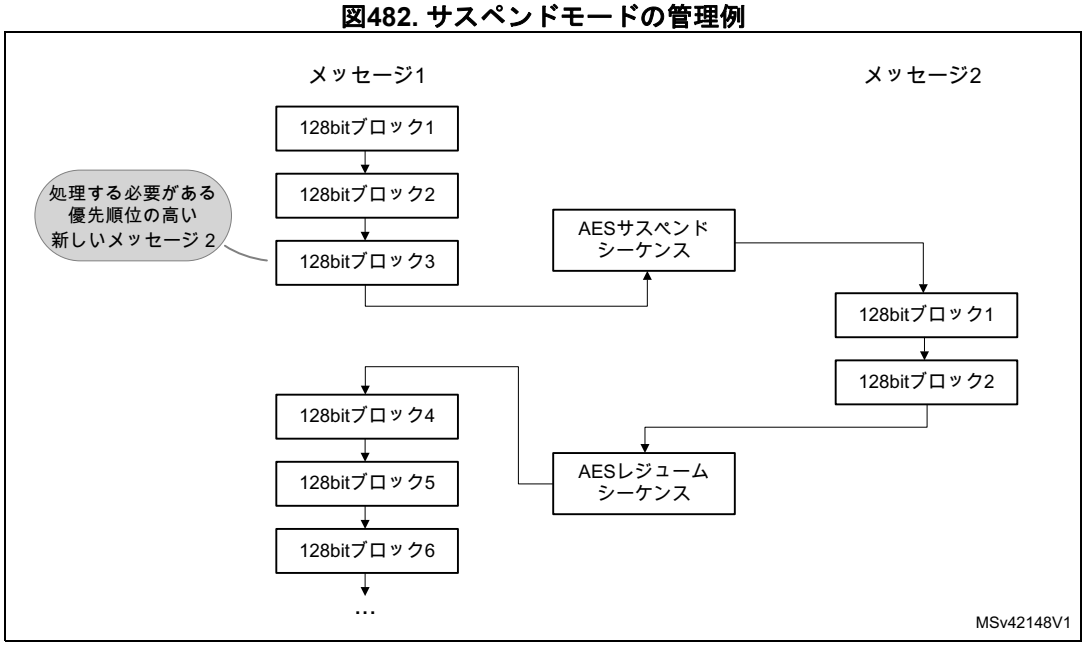
50.4.7 SAES タスクのサスペンドとレジューム

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージは暗号化モードでも復号化モードでも再開できます。

サスペンド／レジューム操作によって連鎖操作が途切れることはなく、SAES が再び有効になるとすぐにメッセージ処理を再開して、次のデータブロックを受信できます。

図482 に、サスペンド／レジューム操作の例を示します。メッセージ1は、より短く優先順位の高いメッセージ2を送信するためにサスペンドされています。

図482. サスペンドモードの管理例



サスペンド／レジューム操作の詳細は、各 SAES モード専用のセクションに説明があります。

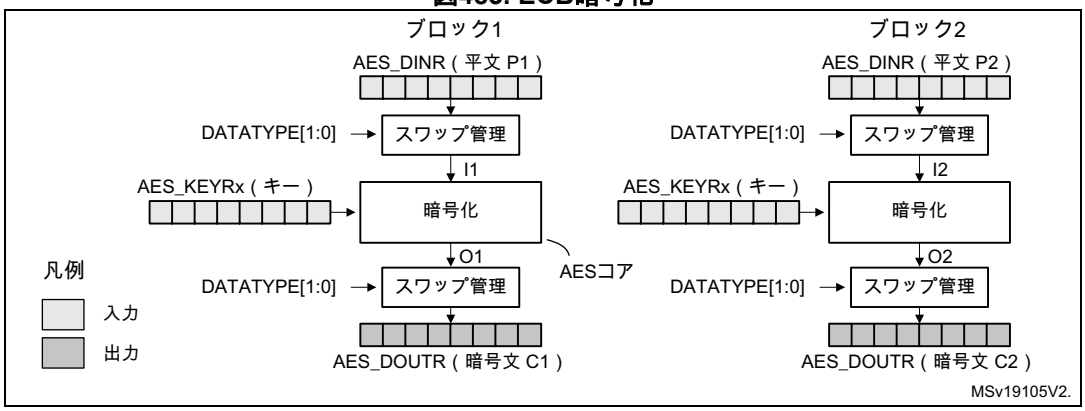
50.4.8 SAES 基本連鎖モード（ECB、CBC）

概要

このセクションでは、SAES コアが提供するECB暗号化、ECB復号、CBC暗号化、CBC復号の4つの基本動作モードについて簡単に説明します。詳細については、2001年11月26日発行の『FIPS公報197』を参照してください。

図483 に、電子コードブック（ECB）暗号化の図を示します。

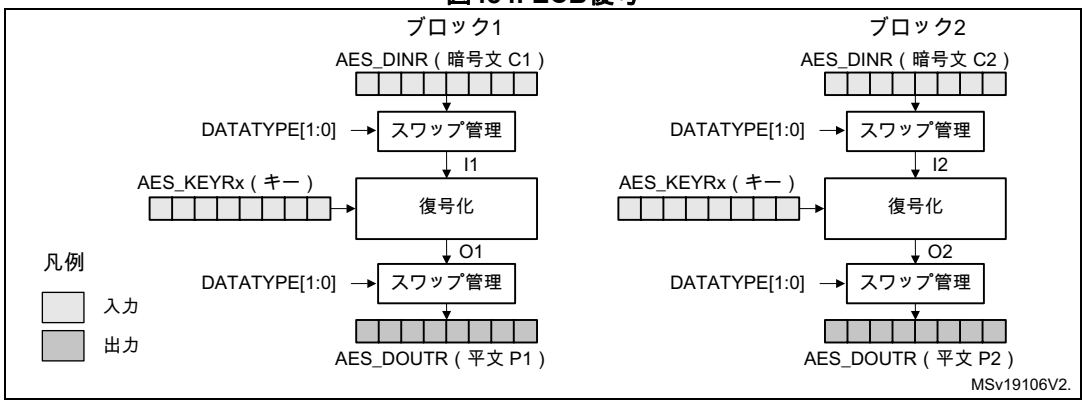
図483. ECB暗号化



ECB暗号化モードでは、まず SAES_DINRレジスタの128bitの平文入力データブロックPxのビット／バイト／ハーフワードのスワッピングが行われます。スワップの結果Ixが、128bitまたは256bitキーを使って暗号化モードでセットされたAESコアで処理されます。暗号化の結果Oxのビット／バイト／ハーフワードのスワッピングが行われ、128bitの暗号文出力データブロックCxとして SAES_DOUTRレジスタに格納されます。ECB暗号化は、最後の完全な平文ブロックが暗号化されるまで同様に続きます。

図484に、電子コードブック（ECB）復号化の図を示します。

図484. ECB復号

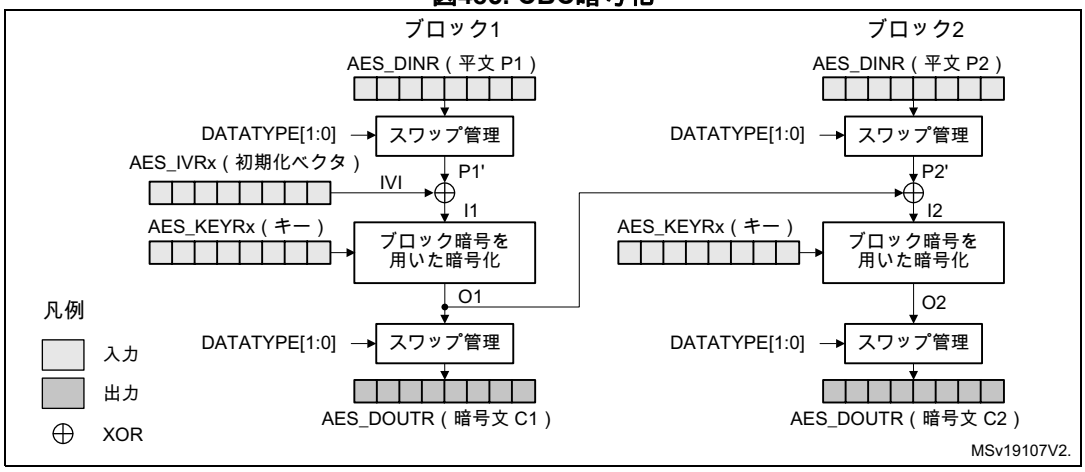


ECBモードでAES復号化を実施するには、最終ラウンドの暗号化キーを回収し、それを暗号文を復号化するための最初のラウンドのキーとして使用して、秘密鍵を準備する必要があります（まず暗号化の完全なキースケジュールを実行する必要があります）。この準備は、AESコアによってサポートされます。

ECB復号化モードでは、まず SAES_DINRレジスタの128bitの暗号文入力データブロックC1のビット／バイト／ハーフワードのスワッピングが行われます。このキーイングシーケンスは、ECB暗号化のものに対し、逆になっています。スワップの結果I1は、事前に準備した復号化キーを使って復号化モードでセットされたAESコアで処理されます。復号化結果のビット／バイト／ハーフワードのスワッピングが行われ、128bitの平文出力データブロックP1として SAES_DOUTRレジスタに格納されます。ECB復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

図485に、暗号ブロック連鎖（CBC）復号の図を示します。

図485. CBC暗号化

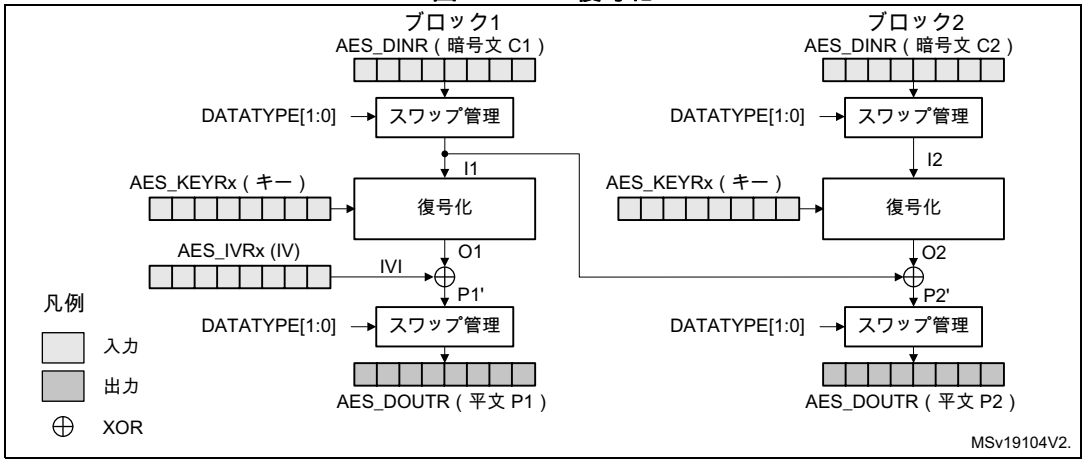


CBC暗号化モードでは、ビット／バイト／ハーフワードのスワッピング（P1’）後、最初の平文入力ブロックと128bitのIVIビットフィールド（初期化ベクタおよびカウンタ）の排他的論理和がとられ、128bitまたは256bitキーを使ってAESコアで暗号化するためのI1入力データを生成します。スワッピング操作後、その結果の128bitの出力ブロックO1が暗号文C1として使用されます。その後、O1データと2番目の平文データP2’ の排他的論理和がとられ、AESコアが暗号文データの2番目を生成するためのI2入力データを生成します。データブロックの連鎖は、メッセージ内の最後の平文ブロックが暗号化されるまで同様に続きます。

メッセージサイズが128bitの倍数ではない場合、最後の不完全なデータブロックは、セクション 50.4.6：SAES 暗号文借用およびデータパディングに説明されている方法で暗号化されます。

図486に、暗号ブロック連鎖（CBC）復号化の図を示します。

図486. CBC復号化



ECB復号化モードと同様に、CBC復号化モードでAES復号化を実施するには、秘密鍵を準備する必要があります。

キーの準備処理後、最初の128bitの暗号文ブロック（スワップ操作後）が128bitまたは256bitキーを使った復号化操作用のAESコア入力ブロックI1として直接使用され、復号化が進みます。その出力O1と、128bitのIVIフィールド（暗号化で使用したものと同じである必要があります）の排他的論理和をとって、最初の平文ブロックP1を生成します。

2番目の暗号文ブロックは、最初のブロックのI1データが初期化ベクタの代わりに使用されることを除けば、最初のブロックと同じ方法で処理されます。

復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

メッセージサイズが128bitの倍数ではない場合、最後の不完全なデータブロックは、セクション 50.4.6：SAES 暗号文借用およびデータパディングに説明されている方法で復号化されます。

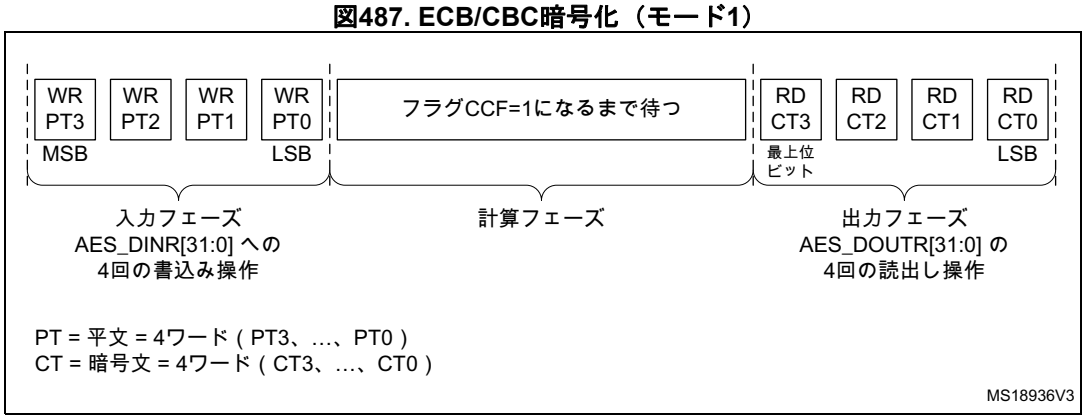
データスワッピングの詳細については、セクション 50.4.11：SAES データレジスタおよびデータスワッピングを参照してください。

ECB/CBC暗号化シーケンス

ECB/CBC暗号化を行うためのイベントのシーケンス（詳細はセクション50.4.4を参照）：

1. SAES_SRのBUSY = 0 （RNGによる乱数フェッチが実行されていない状態）を確認します。
2. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
3. SAES_CRレジスタのMODE[1:0] ビットフィールドを00にセットしてモード1を選択し、SAES_CRレジスタのCHMOD[2:0] ビットフィールドをそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。データ型もDATATYPE[1:0] ビットフィールドで定義できます。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0] 値による暗号化については、セクション50.4.10およびセクション50.4.9を参照してください。
4. SAES_CRレジスタのKEYSIZEビットで128bitまたは256bitのキー長を選択します。キーを別のセキュリティコンテキスト（異なるセキュア属性）と共有してはならない場合は、SAES_CRレジスタのKEYPROTビットもセットする必要があります。
5. SAES_KEYRx レジスタ（128bit または 256bit）に暗号化キーを書き込みます。あるいは、KEYSEL[2:0] を通じて、キーレジスタとは異なるキーソースを選択します。詳細については、セクション 50.4.12：SAES キーレジスタを参照してください。CBCモードを選択した場合、SAES_IVRxレジスタに初期化ベクタ·データを書き込みます。
6. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
7. SAES_DINRレジスタに図487に示すとおり4回書き込んで平文（MSBファースト）を入力します。
8. SAES_SRレジスタでCCFフラグがセットされるまで待ちます。
9. SAES_DOUTRレジスタを図487に示すとおり4回読み出して、暗号文（MSBファースト）を取得します。SAES_ISR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
10. 手順7、8、9を繰り返して、同じ暗号化キーですべてのブロックを処理します。

図487. ECB/CBC暗号化（モード1）

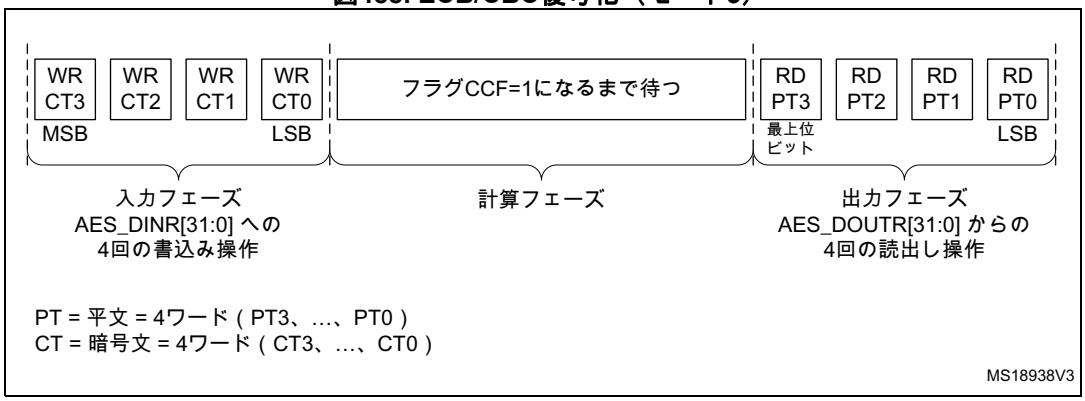


ECB/CBC復号化シーケンス

AES ECB/CBC復号を行うためのイベントのシーケンス（詳細はセクション50.4.4を参照）は次のとおりです。通常のキーモードを選択するには、KMOD[1:0] を00に設定します。その他のKMOD[1:0]値による復号化については、セクション50.4.10およびセクション50.4.9を参照してください。

1. セクション 50.4.5 : SAES 復号ラウンド･キーの準備に示された手順に従って、AESコアで復号キーを準備します。
2. SAES_CRレジスタのMODE[1:0] ビットフィールドを10にセットしてモード3を選択し、SAES_CRレジスタのCHMOD[2:0] ビットフィールドをそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。データ型もDATATYPE[1:0] ビットフィールドで定義できます。KEYSIZE およびKMOD[1:0] ビットフィールドは、そのままの状態を維持する必要があります。
3. SAES_IVRxレジスタに初期化ベクタを書き込みます（CBCモードのみ必要）。
4. SAES_CRレジスタのENビットをセットして、SAES を有効にします。
5. SAES_DINRレジスタに図488に示すとおり4回書き込んで暗号文（MSBファースト）を入力します。
6. SAES_SRレジスタでCCFフラグがセットされるまで待ちます。
7. SAES_DOUTRレジスタを図488に示すとおり4回読み出して、平文（MSBファースト）を取得します。SAES_ISR レジスタの CCF ビットをセットしてCCFフラグをクリアします。
8. 手順5、6、7を繰り返して、同じキーで暗号化されたすべてのブロックを処理します。

図488. ECB/CBC復号化（モード3）



ECB/CBCモードでのサスペンド／レジューム動作

次のシーケンスは通常キー･モード（KMOD[1:0] = 00）で有効です。

メッセージの処理をサスペンドするには、次の手順に従います。

1. DMAが使用されている場合には、SAES_CRレジスタのDMAINENビットをクリアして、IN FIFOへの SAES DMA転送を停止します。
2. DMAが使用されていない場合は、SAES_DOUTRレジスタを4回読み出して、最後に処理されたブロックを保存します。DMAが使用されている場合には、SAES_SRレジスタでCCFフラグがセットされるまで待ち、SAES_CRレジスタのDMAOUTENビットをクリアして、OUT FIFOからのDMA転送を停止します。
3. DMAが使用されていない場合は、SAES_SRレジスタのCCFフラグが1（計算完了）になるまでポーリングします。
4. SAES_ISR レジスタの CCF ビットをセットしてCCFフラグをクリアします。

5. 初期化ベクタ·レジスタを保存します（データ処理中に SAES_IVRxレジスタが変更されるCBCモードでのみ必要です）。
6. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
7. SAES_CR レジスタを保存し、キー·レジスタが必要でなければ、より優先順位の高いメッセージを処理するためにクリアします。
8. DMAが使用されている場合には、DMAコントローラのステータス（INおよびOUTデータ転送のポインタ、残りバイト数など）を保存します。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMAが使用されている場合には、DMAコントローラを設定して残りのFIFO INとFIFO OUT転送を完了させます。
2. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
3. SAES_CRレジスタを（正しいKEYSIZEで）復元してから、SAES_KEYRxレジスタを復元します。あるいは、KEYSEL[2:0] を使用して、キーレジスタとは異なるキーソースを選択します。詳細については、セクション 50.4.12 : SAES キーレジスタを参照してください。
4. セクション 50.4.5 : SAES 復号ラウンド·キーの準備に示したように復号化キーを準備します（ECBまたはCBC復号化のみ必要）。
5. 保存した設定を使用して SAES_IVRxレジスタを復元します（CBCモードのみ必要）。
6. SAES_CRレジスタのENビットをセットして、SAES ペリフェラルを有効にします。
7. DMAが使用されている場合には、SAES_CRレジスタのDMAINENビットとDMAOUTENビットをセットして、SAES DMA転送を有効にします。

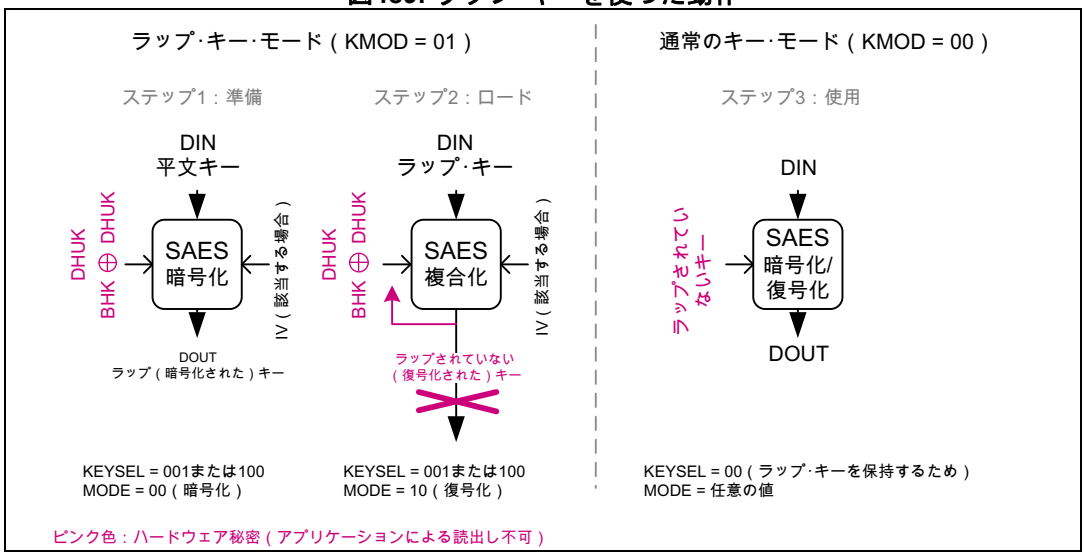
50.4.9 SAESラップキーを使った の動作

SAES ペリフェラルは、XOR演算を使用した、またはアプリケーションキー BHKなしのハードウェア秘密キー DHUKを使用して、アプリケーションキーをラップ（暗号化）またはアンラップ（復号化）することができます。この機能により、アプリケーションソフトウェアは、AESキーを平文（暗号化されていない状態）で公開することなく使用できるようにします。

ラップキーシーケンスは小さすぎるため、サスペンド／レジュームすることができません。SAES では、ラップされていないキーを使ってキーをアンラップすることはできません。

図489は、ラップ·キーを用いた動作を示します。ラップ·キーを保護するために、KEYSEL[2:0] を001または100に設定してDHUKを選択します。あるいは、対応するレジスタがTAMPペリフェラルで読取り／書込みロックされている場合は、KEYSELに010を設定してBHKを選択します。

図489. ラップ･キーを使った動作



注： **DHUK値は、特権、KMOD[1:0]、KEYSEL、CHMOD[2:0] と、SAES ペリフェラルがセキュアであるか非セキュアであるかに依存します。**

ラップ･キー･モードの暗号化

次は、キーをラップ（暗号化）する際の推奨シーケンスです。

1. SAES_SRのBUSY = 0 （RNGによる乱数フェッチが実行されていない状態）を確認します。
2. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
3. SAES_CRレジスタのMODE[1:0] ビットフィールドを00にセットしてモード1（暗号化）を選択し、CHMOD[2:0] ビットフィールドをそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。データタイプは32bitとして定義でき、このときDATATYPE[1:0] ビットフィールドには00がセットされています。キーサイズは、KEYSIZEビットを使用して適切に設定する必要があります。KMOD[1:0] ビットフィールドは01（ラップ･キー）にセットする必要があります。このKEYSIZE情報は、暗号化キーと暗号化されるキーの両方に使用されます。
4. 前の手順でCBCモードが選択された場合は、SAES_IVRxレジスタに初期化ベクタを書き込みます。
5. SAES_CRレジスタのKEYSEL[2:0] ビットフィールドを001または100に設定してDHUKを選択します。キーのロードが完了するとBUSY ビットはクリアされ、SAES_SR レジスタにはKEYVALIDビットがセットされます。KEYSEL = 100の使用方法の詳細については、セクション 50.4.12 : SAES キーレジスタを参照してください。
6. SAES_CRレジスタのENビットをセットして、SAES を有効にします。
7. SAES_DINRレジスタに4回書き込んで、暗号化するキーを入力します（MSBファースト、1974ページの表 475を参照）。
8. SAES_SRレジスタのCCFフラグがセットされるまで待ちます。
9. SAES_DOUTRレジスタを4回読み出して、暗号化されたキー（MSBファースト）を取得します。次に、SAES_ICRレジスタの CCF ビットをセットしてCCFフラグをクリアします。
10. KEYSIZEが256bitの場合は、手順6から8を繰り返します。

注： **ラップ･キー･モードの暗号化は、CHMOD[2:0] を通じてECBまたはCBCが選択されている場合にのみサポートされます。**

ラップ･キー･モードの暗号化

ラップ･キーをアンラップ（復号化）する際の推奨シーケンスは、以下のとおりです。

1. SAES_SRのBUSY = 0 （RNGによる乱数フェッチが実行されていない状態）を確認します。
2. SAES_CRレジスタのENビットをクリアして、SAES ペリフェラルを無効にします。
3. SAES_CRレジスタのMODE[1:0] ビットフィールドを01にセットしてモード2を選択し、CHMOD[2:0] をそれぞれ000または001にセットして、ECBまたはCBCの連鎖モードを選択します。キーサイズは、KEYSIZEビットを使用して適切に設定する必要があります。KMOD[1:0] ビットフィールドは01（ラップ･キー）にセットする必要があります。DATATYPE[1:0] ビットフィールドでのデータ型選択は、暗号化で使用されたものと同じである必要があります（つまり0x0）。このKEYSIZE情報は、復号化キーと復号化されるキーの両方に使用されます。

   復号化されたキーを別のセキュリティコンテキスト（異なるセキュア属性）と共有してはならない場合は、SAES_CRレジスタのKEYPROTビットもセットする必要があります。
4. SAES_CRレジスタのKEYSEL[2:0] ビットフィールドを001または100に設定してDHUKを選択します。キーのロードが完了すると、SAES_SR レジスタの BUSY ビットがクリアされ、KEYVALIDビットがセットされます。KEYSEL = 100の使用方法の詳細については、セクション 50.4.12 : SAES キーレジスタを参照してください。
5. SAES_CRレジスタのENビットをセットして、SAES を有効にします。
6. SAES_SRレジスタのCCFフラグがセットされるまで待ちます。
7. CCFフラグをクリアします。復号化キーはAESコアで使用可能となり、SAES は自動的に無効化されます。
8. MODE[1:0] ビットフィールドに10を設定し、SAES_CRレジスタで、モード3を選択します。
9. 前の手順でCBCモードが選択された場合は、SAES_IVRxレジスタに初期化ベクタを書き込みます。
10. SAES_CRレジスタのENビットをセットして、SAES を有効にします。
11. SAES_DINRレジスタに4回書き込んで、暗号されたランダムなキーを入力します（MSBファースト、1974ページの表 475を参照）。
12. SAES_SRレジスタのCCFフラグがセットされるまで待ちます。
13. CCFフラグをクリアし、KEYSIZEが256bitの場合は手順10および11を繰り返します。

キーレジスタに復号化されたキーがロードされている場合、SAES_CRレジスタのKEYSEL[2:0] が自動的にクリアされます。そのため、このシーケンス後、復号化されたラップ･キーが SAES_KEYRxレジスタで使用可能となり、任意のAES操作について、アプリケーションソフトウェアで直ちに使用可能になります（通常キー･モード）。

手順2でKEYPROTビットをクリアした場合、復号化されたラップ･キーは、別のセキュリティコンテキスト（異なるセキュリティ属性）で実行されているアプリケーションと共有できます。

注： **KMOD[1:0] = 01（ラップ･キー）およびMODE[1:0] = 10（復号化）の場合、SAES_DOUTRレジスタへの読出しアクセスにより、読出しエラー（RDERR）がトリガされます。**

KEYSEL[2:0] = 001（DHUK）または100（DHUK XOR BHK）の場合、アプリケーションソフトウェアは暗号化および復号化に同じ権限、セキュリティ、KMOD[1:0]、CHMOD[2:0]、およびKEYSIZEコンテキストを使用する必要があります。そうしないと、誤った結果になります。

50.4.10 共有キーを使った SAES の動作

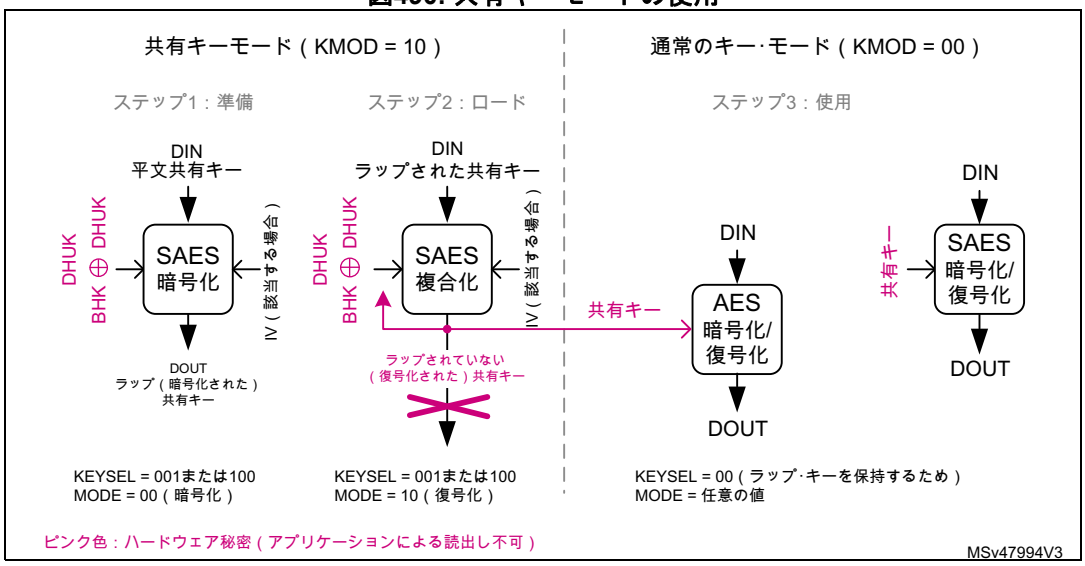
SAES ペリフェラルは、アプリケーションキー BHK とXOR演算された、またはされない、ハードウェア秘密キーDHUKを使用してラップされたアプリケーションキーを共有することができます。この機能を使用すると、アプリケーションソフトウェアはAESキーを平文（暗号化されていない文）で公開することなく、AES ペリフェラルで　使用できるようにすることができます。

共有キーシーケンスは小さすぎるため、サスペンド／レジュームすることができません。SAES では、ラップされていないキーを使って共有キーをアンラップすることはできません。

注： **SAES に格納されているキーが AES と共有されると、KEYPROTビットによる保護が失われます。保護の詳細については、セクション 50.4.12：SAES キーレジスタを参照してください。**

図490には、共有キーモードの使用方法を示します。共有キーを保護するために、KEYSEL[2:0] を 001 または100に設定してDHUKを選択します。あるいは、対応するレジスタがTAMPペリフェラルで読取り／書込みロックされている場合は、KEYSELに010を設定してBHKを選択します。

図490. 共有キーモードの使用



注： **DHUK値は、特権、KMOD[1:0]、KSHAREID、KEYSEL、CHMOD[2:0] と、SAES ペリフェラルがセキュアであるか非セキュアであるかに依存します。**

手順3で、AESは AES ペリフェラルを表わします。

共有キーモードの暗号化

SAES が　ペリフェラルとキーを共有できるようにするには、キーを一度暗号化（ラップ）する必要があります。共有キーの暗号化シーケンスはラップ·キーの場合と同じです。手順3でKMOD[1:0] は 10（共有キー）にセットされ、KSHAREID[1:0] は 00 で保持されます。図490詳細については、ラップ·キー·モードの暗号化を参照してください。

注： **共有キーモードの暗号化は、CHMOD[2:0]　を通じてECBまたはCBCが選択されている場合にのみサポートされます。**

共有キーモードの復号化

SAES で、ペリフェラルとキーを共有する必要があるたびに、共有された暗号化キーは SAES で復号化する必要があり、AES でロードしなければなりません。全体のシーケンスについては、次に記載しています。

ペリフェラルでのシーケンスSAES

共有キーの復号化シーケンスはラップ･キーの場合と同じです。手順3でKMOD[1:0] は10（共有キー）にセットされ、KSHAREID[1:0] は 00 で保持されます。図490詳細については、ラップ･キー･モードの暗号化を参照してください。

注： **KEYSEL[2:0] ビットフィールドは自動的にクリアされるため、SAES では、復号された共有キーを共有する代わりに直接使用できます。この場合、KMOD[1:0] は00（通常キー･モード）にセットする必要があります。**

AESペリフェラルでのシーケンス

共有キーが SAES キーレジスタで復号化されると、AES ペリフェラルと共有できます。SAES ペリフェラルはキー共有状態のままとなります。つまり SAES_SR レジスタのKMOD[1:0] = 10 で、KEYVALID = 1です。AES キー共有のターゲットペリフェラルのシーケンスは、次のとおりです。

1. 一部のデータを処理するために、AESプロセッサを初期化します。
2. キーをロードする必要がある場合は、SAES ペリフェラルと同じKEYSIZEをセットし、AES_CR レジスタのKMOD[1:0] ビットフィールドに10（共有キー）を書き込みます。SAES ペリフェラルで前のシーケンスが正常に完了している場合、KMOD[1:0] が10のまま保持され、KSHAREID[1:0] が 00 のまま保持された状態で、SAES_KEYRxレジスタには自動的に AES_KEYRx レジスタにコピーされ、AES_SR にBUSYがセットされます。
3. 転送が完了すると、BUSYフラグはクリアされ、AES_SR レジスタにKEYVALIDがセットされます。BUSYビットがクリアされたときにKEYVALIDがセットされない場合、またはキーエラーフラグ（KEIF）がセットされた場合は、転送中にDPAエラーなどの予期せぬイベントが発生したか、タンパイベントまたはKEYVALID SAES フラグが転送終了前にクリアされています。このようなエラーが発生した場合、両方の制御レジスタのIPRSTビットを通じて、SAES ペリフェラルから開始されたキーの共有プロセス全体を最初からやり直す必要があります。
4. KEYVALIDがセットされる際、キー共有のターゲットペリフェラルは有効な共有キーにより初期化されます。アプリケーションは、KMOD[1:0] を00に設定して、データの処理を進めることができます。

SAES が未使用であり、正しくキー共有状態にある限り、このシーケンスは複数回（たとえば、サスペンド／レジューム状況を管理するために）実行できます。

注： **SAES ペリフェラルで、KMOD[1:0] = 10（共有キー）およびMODE[1:0] = 10（復号化）の場合、SAES_DOUTRレジスタへの読出しアクセスにより、読出しエラー（RDERR）がトリガされます。**

KEYSEL[2:0] = 001（DHUK）または100（DHUK XOR BHK）の場合、アプリケーションソフトウェアは暗号化および復号化に同じ権限、セキュリティ、KMOD[1:0] / KSHAREID[1:0]、CHMOD[2:0]、およびKEYSIZEコンテキストを使用する必要があります。そうしないと、誤った結果になります。

AESペリフェラルでKMOD[1:0] = 10かつBUSY = 1で、AESSAESと のKEYSIZE値が異なる場合、キー共有は失敗し、両方のペリフェラルでKEIFフラグが立てられます。

50.4.11 SAES データレジスタおよびデータスワッピング

データの入出力

128bitのデータ･ブロックは、SAES_DINRレジスタ（ビットフィールドDIN[31:0]）に、4つの32bit･ワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）の順序で連続して書き込むことで SAES ペリフェラルに入力されます。

128bitのデータ･ブロックは、SAES_DOUTRレジスタ（ビットフィールドDOUT[31:0]）から、4つの32bit･ワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）の順序で連続して読み出すことで、SAES ペリフェラルから取得されます。

SAES_DINRレジスタまたは SAES_DOUTR レジスタの32bitのデータ･ワードは、次のいずれかのビッグ･エンディアン順で整理されます。

- SAES_DINRに書き込むワードの最上位バイトは、書き込むワードを保持する4つの隣接したメモリ位置のうち、一番小さいアドレスに配置する必要があります。
- SAES_DOUTRから読み出すワードの最上位バイトは、ワードを受け取る4つの隣接したメモリ位置のうち、一番小さいアドレスになります。

SAES への入力データブロック書込みにDMAを使用する場合は、入力ブロックの4ワードをビックエンディアン順（一番小さいアドレスに最上位ワード）で連続してメモリに格納する必要があります。セクション 50.4.14：SAES DMAインタフェースを参照してください。

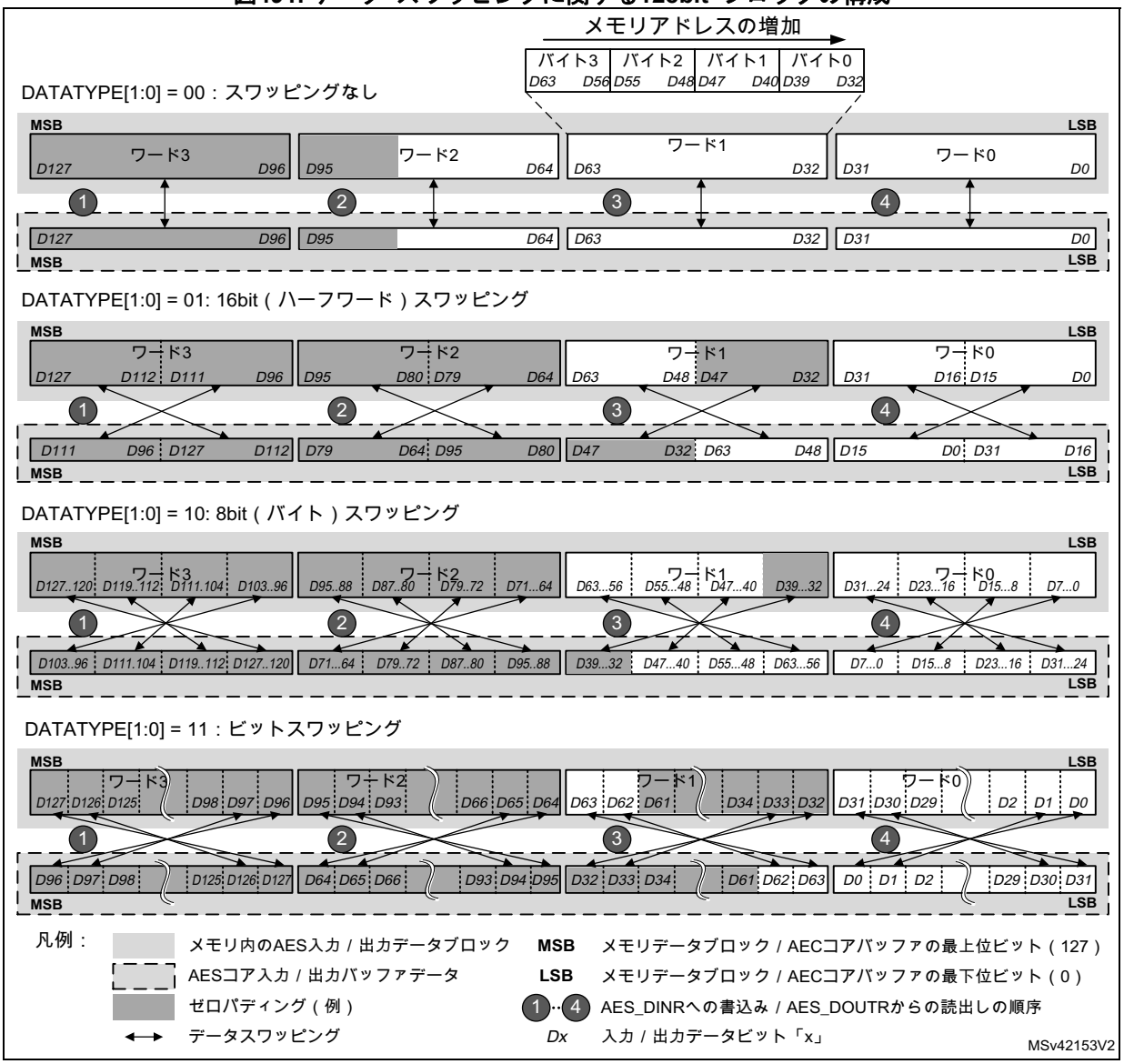
データスワッピング

SAES ペリフェラルは、AES処理コアに読み込まれる前の SAES_DINRレジスタの入力データワード、および SAES_DOUTRレジスタに送られる前のAES処理コアから出力されたデータについて、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしに設定できます。データのタイプによって選択します。たとえば、ASCIIテキストストリームの場合、バイトスワッピングを使用します。

データスワッピングのタイプは、SAES_CRレジスタのDATATYPE[1:0] ビットフィールドで選択します。この選択は、AESコアの入力と出力の両方に適用されます。

それぞれのデータ･スワッピングのタイプについては、図491に SAES_DINRレジスタから入力されたデータのAES処理コアの入力バッファ･データP127 ~ P0の構成、または SAES_DOUTRレジスタから得られる出力データのAES処理コアの出力バッファ･データP127 ~ P0の構成を示しています。

図491. データ･スワッピングに関する128bit･ブロックの構成



注： **SAES キーレジスタ（SAES_KEYRx）と初期化レジスタ（SAES_IVRx）のデータは、選択されたスワップモードの影響を受けません。**

データパディング

図491では、データスワッピング後の0にされたビットによってAESコア入力バッファの最上位ビット側に隣接したゾーンが形成される場合など、0でパディングされるメモリデータブロックの例も示しています。この例では、次を含む入力データブロックのパディングが示されています。

- 48メッセージビット（DATATYPE[1:0] = 01）
- 56メッセージビット（DATATYPE[1:0] = 10）
- 34メッセージビット（DATATYPE[1:0] = 11）

50.4.12 SAES キーレジスタ

書込み専用の SAES_KEYRxレジスタは、暗号化または復号化キービットフィールドのKEY[127:0] や KEY[255:0] を格納します。各レジスタに書き込むデータは、リトルエンディアン順で最上位バイトをメモリ内の一番大きいアドレスに配置するように整理されます（セキュリティ上の理由で読出しはできません）。

表475 に示すように、キーは8個のレジスタに跨ります。

表475.SAES_KEYRxレジスタでのキーのエンディアン設定（128bitまたは256bitのキー長）

SAES_KEYR7 [31:0]	SAES_KEYR6 [31:0]	SAES_KEYR5 [31:0]	SAES_KEYR4 [31:0]	SAES_KEYR3 [31:0]	SAES_KEYR2 [31:0]	SAES_KEYR1 [31:0]	SAES_KEYR0 [31:0]
-	-	-	-	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]
KEY[255:224]	KEY[223:192]	KEY[191:160]	KEY[159:128]	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]
TAMP_BKP7R [31:0]	TAMP_BKP6R [31:0]	TAMP_BKP5R [31:0]	TAMP_BKP4R [31:0]	TAMP_BKP3R [31:0]	TAMP_BKP2R [31:0]	TAMP_BKP1R [31:0]	TAMP_BKP0R [31:0]

暗号化または復号化のキーは、SAES ペリフェラルが無効であるときに、SAES_CRレジスタのENビットと KEYSEL[2:0] ビットフィールドをクリアして、これらのレジスタに書き込むことができます。

キー･レジスタは、SAES_CR レジスタのDATATYPE[1:0] ビットフィールドで制御されるデータ･スワッピングによる影響を受けません。

キー全体を、AES計算を開始する前に書き込む必要があります。通常のキーモード（KMOD[1:0] = 00）では、KEYSEL[2:0] = 000のとき、SAES_KEYRx（KEYSIZE = 0の場合x = 0 ~ 3、KEYSIZE = 1の場合x = 0 ~ 7）レジスタは、必ず昇順または降順のどちらかで書き込む必要があります。

KEYSEL[2:0] が001にセットされている場合、SAESエンジン内で不揮発性のOTPベースのルートのハードウェア･ユニーク･キーから計算された導出されたハードウェア･ユニーク･キー（DHUK）は、KEYSIZE情報に基づいて直接キーレジスタにロードされます。

KEYSEL[2:0] が010の場合、耐タンパ性のあるセキュアバックアップレジスタに格納されているブート･ハードウェア･キー（BHK）は、セキュア･アプリケーションがすべてのTAMP_BKPxR レジスタ（KEYSIZE = 0の場合x = 0 ~ 3、KEYSIZE = 1の場合x = 0 ~ 7）を昇順または降順で1回読み出すことによって、キーレジスタに完全に転送されます。表475を参照してください。

KEYSEL[2:0] が100の場合、DHUKとBHKのXOR演算の組み合わせは、セキュア･アプリケーションがすべてのTAMP_BKPxRレジスタ（KEYSIZE = 0の場合x = 0 ~ 3、KEYSIZE = 1の場合x = 0 ~ 7）を昇順または降順で1回読み出すことによって、キーレジスタに完全に転送されます。表475を参照してください。

KEYVALID = 0の場合、KEYSEL[2:0] にゼロ以外の同じ値を繰り返し書き込むと、DHUK またはBHKの読み込みがトリガされるだけになります。KEYVALIDをクリアするには、SAES_CRレジスタのIPRSTビットをセットする方法が推奨されます。この方法は、たとえば同じBHK（KEYSEL[2:0] = 010）を選択してECB復号化からECB暗号化に切り替える際に必要です。

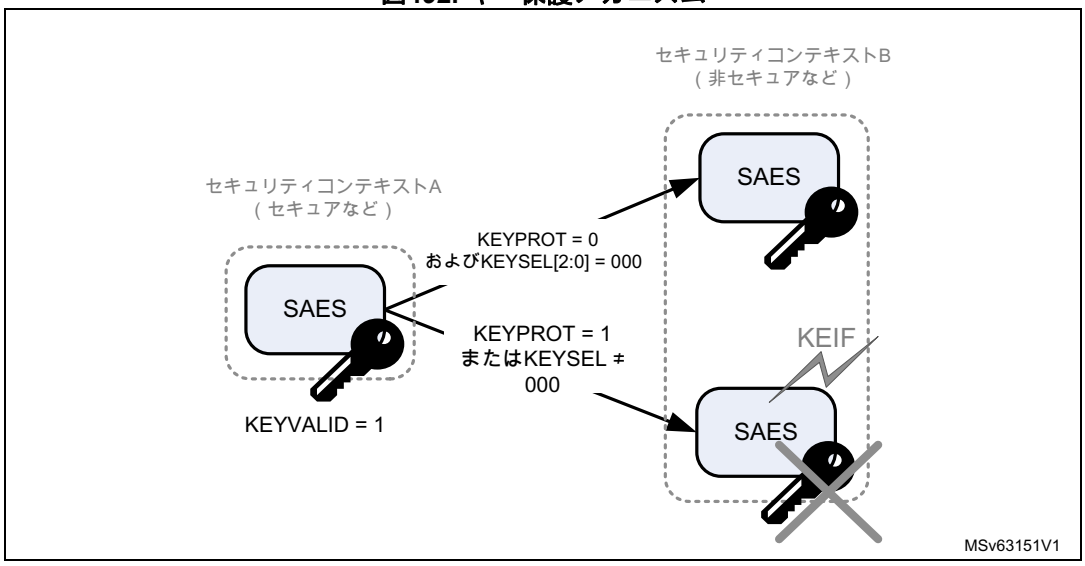
SAES_CRレジスタのKEYPROTビットは、キーレジスタにロードするキーを別のセキュリティコンテキスト（つまり、異なるセキュリティ属性）で実行されるアプリケーションと共有してはならない場合にセットする必要があります。KEYPROTとKEYVALIDを設定すると、キーをロードしたセキュリティコンテキストとは別のセキュリティコンテキストで実行されるアプリケーションが SAES アクセスを試みたときに、KEIFはエラーを示します。図492を参照してください。

注： **KEYSEL[2:0] 値が0（通常のキー）と異なる場合、キーレジスタは自動的に保護されます。**

DHUK、BHK、およびそれらのXOR演算の組み合わせは、どのソフトウェアでも（セキュアであっても）読み出せません。

セキュアSAESはセキュアDHUKを使用します。非セキュアSAESは非セキュアDHUKを使用します。

図492. キー保護メカニズム



注： **キーのローディングシーケンスを開始すると、BUSYフラグがセットされ、KEYVALIDフラグがクリアされます。KEYSIZEで定義されたビット量が SAES_KEYRxレジスタに転送されると、BUSYがクリアされ、KEYVALIDがセットされ、ENビットが書き込めるようなります。エラーが発生すると、BUSYとKEYVALIDがクリアされ、KEIFがセットされます（詳細はセクション 50.4.15：SAES エラー管理を参照）。これは、すべてのKEYSEL値を保持します。**
キーモードの詳細については、セクション50.4.10 およびセクション50.4.9を参照してください。

50.4.13 SAES 初期化ベクタレジスタ

4つの SAES_IVRxレジスタで初期化ベクタの入力ビットフィールドIVI[127:0] を保持します。各レジスタで読み書きするデータは、リトルエンディアン順で最上位バイトをメモリ内の一番大きいアドレスに配置するように整理されます。レジスタも一番小さいアドレス（SAES_IVR0）から一番大きいアドレス（SAES_IVR3）の順になります。

ビットフィールドのデータの位は、選択された連鎖モードに基づきます。使用時は、ビットフィールドがAESコアの計算サイクルごとに更新されます。

SAES ペリフェラルが有効であるときの SAES_IVRxレジスタへの書込み操作は、レジスタの内容に影響を与えません。SAES_IVRxレジスタの内容を変更する場合はまず SAES_CRレジスタのENビットをクリアする必要があります。

SAES_IVRxレジスタを読み出すと、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

SAES_IVRxレジスタは、SAES_CRレジスタのDATATYPE[1:0] ビットフィールドで制御されるデータ・スワッピング機能による影響を受けません。

50.4.14 SAES DMAインタフェース

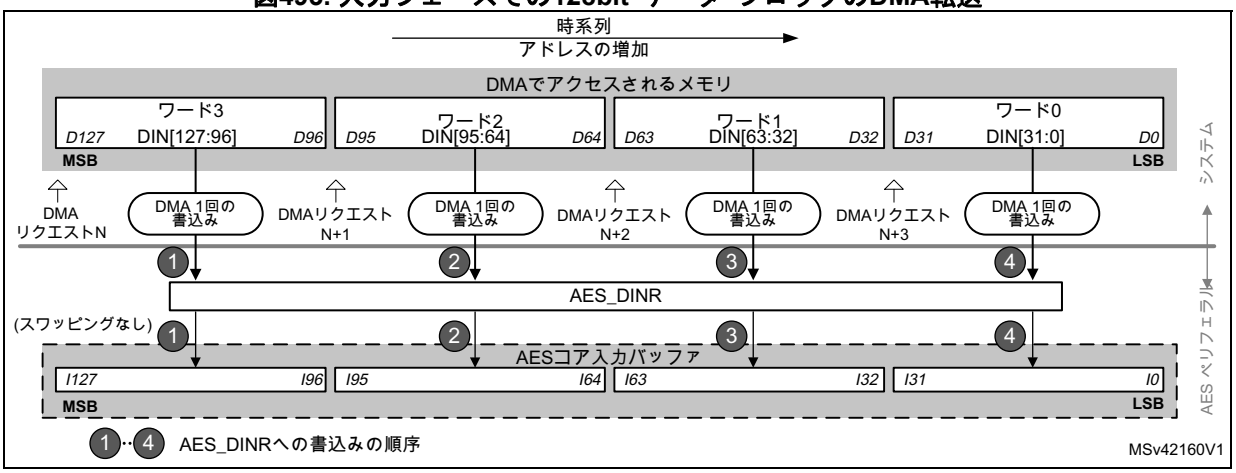
SAES ペリフェラルには、DMA（ダイレクトメモリアクセス）コントローラに接続するインタフェースが搭載されています。DMA操作は SAES_CRレジスタで制御します。

DMAを用いたデータ入力

SAES_CRレジスタのDMAINENビットをセットすると、SAES へのDMA書込みが有効になります。図493 に示すように、SAES ペリフェラルは、入力フェーズ中に SAES_DINRレジスタに128bitのブロック（4倍ワード）を書き込む必要があるたびに、DMAリクエストを開始します。

注： **選択されたアルゴリズムとモードに応じて、特別なパディングや暗号文借用が必要となることがあります。詳細については、セクション 50.4.6 : SAES 暗号文借用およびデータパディングを参照してください。**

図493. 入力フェーズでの128bit･データ･ブロックのDMA転送

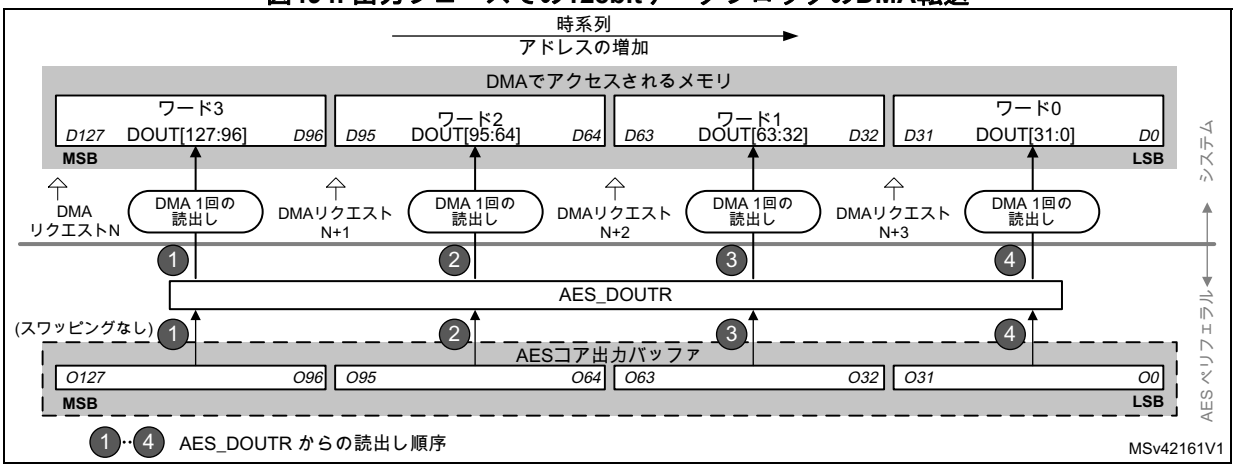


DMAを用いたデータ出力

SAES_CRレジスタのDMAOUTENビットをセットすると、SAES からのDMA読出しが有効になります。図494に示すように、SAES ペリフェラルは、出力フェーズ中に SAES_DINRレジスタから128bitのブロック（4倍ワード）を読み出す必要があるたびに、DMAリクエストを開始します。

注： **メッセージサイズによっては、最後のブロックでアプリケーションによって余分なバイトを破棄する必要が出ることがあります。**

図494. 出力フェーズでの128bitデータブロックのDMA転送



さまざまな動作モードにおけるDMA操作

DMA操作は、モード1（暗号化）またはモード3（復号化）が SAES_CRレジスタのMODE[1:0] ビットフィールドで選択されている場合に使用できます。モード2（キー導出）では、SAES_KEYRxレジスタはソフトウェアで書き込む必要があり、SAES_CRレジスタのDMAINENおよびDMAOUTENビットでDMA転送を有効にしても、このモードでは影響がありません。

DMAシングル·リクエストは、無効にされるまで SAES によって生成されます。そのため、128bitデータブロック処理終了時のデータ出力フェーズの後に、SAES によって次のデータブロックの新しいデータ入力フェーズ（ある場合）に自動的に切り替えられます。

SAES とメモリ間のデータ転送をDMAで処理する場合、CCFフラグは不要です。SAES_DOUTRレジスタの読出しが、計算フェーズの最後にDMAによって自動的に処理されるからです。CCFフラグのクリアが必要になるのは、ソフトウェア管理のデータ転送に戻るときだけです。詳細については、セクション 50.4.4：暗号操作を行うためのSAES の手順 のサブセクション データ追加を参照してください。

50.4.15 SAES エラー管理

特に指定がない限り、SAES 設定は、SAES_CRレジスタのENビットをクリアすることでいつでも変更できます。

読出しエラー·フラグ（RDERR）

SAES_DOUTRレジスタが予期せず読出しを試みることで、SAES_SRレジスタのRDERRフラグとSAES_ISRレジスタのRWEIFフラグがセットされ、0が返されます。

RDERRは、計算フェーズまたは入力フェーズでトリガされます。

注： **特に指定がない限り、SAES はRDERRエラー検出時に無効にならずに処理を続けます。**

SAES_IER レジスタの RWEIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 50.5：SAES割込みを参照してください。

RDERR およびRWEIF フラグは、SAES_ISR レジスタの RWEIE ビットをセットしてクリアします。

書込みエラーフラグ（WDERR）

SAES_DINRレジスタが予期せず書込みを試みることで、SAES_SRレジスタのWRERRフラグとSAES_ISRレジスタのRWEIFフラグがセットされます。SAES_DINRへの影響はありません。WRERRは、計算フェーズまたは出力フェーズでトリガされます。

注： **特に指定がない限り、SAES はWRERRエラー検出後に無効にならずに処理を続けます。**

SAES_IER レジスタの RWEIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 50.5：SAES割込みを参照してください。

WRERR およびRWEIF フラグは、SAES_ISR レジスタの RWEIF ビットをセットしてクリアします。

キーエラー割込みフラグ（KEIF）

キーレジスタへのキーのロードに失敗したり、キーが保護されているときにキーをロードしようとすると、SAES_ISRレジスタのKEIFフラグがセットされ、SAES_SRレジスタのKEYVALIDビットがクリアされます。

KEIFフラグは、SAES_ISR レジスタの対応するビットによってクリアされます。SAES_IER レジスタのKEIEビットがセットされている場合、割込みが生成されます。詳細については、セクション 50.5: SAES割込みを参照してください。

考えられるキーエラーの原因は次のとおりです。

- キー保護エラー：KEYVALIDがセットされているときに、KEYPROT = 1またはKEYSELが0以外の場合、キーをロードするために使用したセキュリティコンテキストとは別のセキュリティコンテキストで実行されるアプリケーションの SAES（つまり、異なるセキュリティ属性）へのアクセスが検出されると、このエラーがトリガされます。
- キー書込みシーケンスエラー：キーレジスタの書込み時に正しくないシーケンスが検出された場合。詳細については、セクション 50.4.12：SAES キーレジスタを参照してください。
- キー共有サイズの不一致：KMOD[1:0] = 10 で、SAESペリフェラルに格納されたKEYSIZE と一致しない AES ペリフェラルにアプリケーションがKEYSIZE情報をセットし、エラーがトリガされた場合。
- キー共有エラー：SAESペリフェラルからAES へのキーレジスタのコピーが失敗した場合。詳細については、セクション 50.4.10：共有キーを使った SAES の動作を参照してください。
- ハードウェア秘密キーのロードエラー：DHUK またはBHKが SAES へのロードに失敗しました。KEYSEL = 001（DHUK）、010（BHK）または100（DHUK XOR BHK）は機能しません。

キー共有エラーが発生したら、対応する制御レジスタのIPRSTビットを通じて AES とSAESの両方のペリフェラルをリセットし、その後キー共有シーケンスを最初からやり直します。

キー選択エラーが発生した場合、KEIFフラグをクリアすると、キー選択プロセスが自動的に再開されます。問題が解決しない場合（たとえば、RHUKロード失敗など）、パワーオンリセットが必要になることがあります。

注： **キーエラーの発生時には、再設定を行う前にKEIFフラグをクリアしてくださいSAES。**

RNGエラー割込みフラグ（RNGEIF）

SAES は、RCC でIPリセットがトリガされた後で、RNGペリフェラルから自動的に乱数をフェッチします。SAESは、RNGEIFがセットされているときは使用できません。

RNGペリフェラルから乱数を取得しているときにエラーが検出されると（たとえば、不正なエントロピーが原因で）、SAES_ISRレジスタのRNGEIFフラグがセットされます。このフラグは、SAES_ICR レジスタの対応ビットをセットしてクリアします。SAES_IERレジスタの RNGEIEビットがセットされている場合、割込みが生成されます。詳細については、セクション 50.5：SAES割込みを参照してください。

RNGエラーが発生した場合：

- RNGペリフェラルのAHBクロックが有効で、かつ、このペリフェラルにノイズソース（またはシード）エラーが保留されていないことを確認してください。
- SAES_CRレジスタのIPRSTビットをセットして、SAES_ISRレジスタのRNGEIFビットをクリアするか、ペリフェラルをリセットします。その後、SAES_SRレジスタのBUSYフラグがクリアされれば、RNGからの乱数の取得が完了したことになります。

注： **RNGEIFのエラーを避けるため、SAES AHB クロックが有効になるたびにRNG AHB クロックを有効にすることをお勧めします。**

DPAエラーについて

予期しないエラーが発生すると、TAMPペリフェラルの SAES 内部タンパイベントがトリガされ、SAES コプロセッサの処理が停止します。

通常の動作を再開するため、アプリケーションはRCCまたはグローバルリセットを通じて SAES ペリフェラルをリセットする必要があります。

50.5 SAES割込み

SAES ペリフェラルによって生成され、個別にマスク可能な割込みソースは、次のイベントを通知します。

- 計算完了
- 読出しエラー
- 書込みエラー
- キーエラー
- RNGエラー

これらのソースは、Arm® Cortex® 割込みコントローラに接続された SAES ペリフェラルからの共通の割込み信号に組み合わされます。各ソースは、SAES_IER レジスタの対応するイネーブル･ビットをセットまたはクリアすることで個別にそれぞれ有効化または無効化され、SAES_ISR レジスタの対応するビットをセットすることでクリアされます。

各ステータスは、SAES_SR および SAES_ISR レジスタから読み出すことができます。

割込みソースの概要、イベント･フラグ、イネーブル･ビットについては、表476を参照してください。

表476.SAES 割込みリクエスト

項目（割込みの略称）	SAES 割込みイベント	イベント･フラグ	イネーブル･ビット	割込みのクリア方法
SAES	計算完了フラグ	CCF	CCFIE	CCF をセット(1)
	読出しエラーフラグ	RDERR(2)	RWEIE	RWEIF をセット (1)
	書込みエラーフラグ	WRERR (2)		
	キーエラーフラグ	KEIF	KEIE	KEIFのセット (1)
	RNGエラーフラグ	RNGEIF	RNGEIE	RNGEIFをセット (1)

1. SAES_ISR レジスタのビット。
2. SAES_SRレジスタのフラグであり、SAES_ISRレジスタのフラグRWEIFによって反映されます。

50.6 SAES 処理遅延時間

以下の表では、各動作モードで128bit･ブロックを処理する際の遅延時間の概要を示します。

表477.ECB、CBCの処理遅延時間

キー長	動作モード	アルゴリズム	クロック サイクル数(1)
128bit	モード1：暗号化	ECB、CBC	**528**
	モード2：キー導出	-	**200**
	モード3：復号化	ECB、CBC	**528**
256bit	モード1：暗号化	ECB、CBC	**743**
	モード2：キー導出	-	**324**
	モード3：復号化	ECB、CBC	**743**

1. RCCからのSHSIクロック（48MHz、+/-15% ジッタ）

50.7 SAES レジスタ

50.7.1 SAES制御レジスタ（SAES_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPRST	KEYSEL[2:0]			KSHAREID[1:0]		KMOD[1:0]		Res.				KEYPROT	KEYSIZE	Res.	CHMOD[2]
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DMAOUTEN	DMAINEN	Res.	Res.	Res.	Res.	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
			rw	rw					rw	rw	rw	rw	rw	rw	rw

ビット31 **IPRST**：SAES ペリフェラルのソフトウェアリセット

ビットを設定すると、IPRSTビット自体と SAES_DPACFGレジスタを除くすべてのレジスタがデフォルト値になり、SAES ペリフェラルがリセットされます。したがって、キー関連のデータは、すべて失われます。このため、安全性の低いアプリケーションに SAES を引き渡す前にこのビットをセットすることをお勧めします。どの設定レジスタに書き込んでいるときも、このビットはローである必要があります。

ビット30:28 **KEYSEL[2:0]**：キー選択

このビットフィールドにより、AES暗号コアで使用するキー情報のソースを規定します。

000：キーレジスタ SAES_KEYxに読み込まれたソフトウェア･キー

001：導出されたハードウェア･ユニーク･キー（DHUK）

010：ブート･ハードウェア･キー（BHK）

100：DHUKとBHKのXOR演算

111：テスト･モード･キー（256bitのハードウェア定数0xA5A5...A5A5）

その他：予約済み（使用した場合は、IPRSTによりSAESを固定解除）

KEYSELがゼロ以外の場合、SAES_SRレジスタのBUSYビットがクリアされ、KEYVALIDがセットされていれば、選択したキーの値がキー･レジスタに存在しています。そうでないときは、キーエラーフラグKEIFがセットされます。KEYVALID = 0の場合、KEYSEL[2:0] にゼロ以外の同じ値を繰り返し書き込むと、DHUKまたはBHKの読み込みがトリガされるだけになります。

アプリケーションソフトウェアがKEYSEL[2:0] ビットフィールドに書き込むことによってキー選択を変更すると、キーレジスタは直ちに消去され、KEYVALIDフラグがクリアされます。

復号プロセスの最後に、KMOD[1:0] が0以外の場合、KEYSEL[2:0] はクリアされます。

ビットフィールドに0以外の値とKEYVALIDがセットされている場合、アプリケーションは、ロードされたキーを持つ SAES の所有権を、別のセキュリティコンテキスト（セキュア、非セキュアなど）で実行されているアプリケーションに転送することはできません。より具体的には、任意のレジスタへのアクセスのセキュリティがSAES によって記録された情報と一致しない場合、KEIFフラグがセットされます。

SAES_SRレジスタのBUSYフラグがセットされている場合、および書込みアクセスの前に SAES_CRレジスタのENビットがセットされていて、その書込みアクセスによってクリアされない場合、このビットフィールドへの書込みの試みは無視されます。

ビット27:26 **KSHAREID[1:0]**：キー共有識別

このビットフィールドは、KMOD[1:0] = 10（共有キー）とした復号プロセスの最後で、専用ハードウェア･バスを使用してどのターゲットが SAES のキー･レジスタを読み出せるかを定義します。

00：AESペリフェラル

その他：予約済み

SAES_SRレジスタのBUSYフラグがセットされている場合または SAES_CRレジスタのENビットがセットされている場合、このビットフィールドに書き込もうとしても無視されます。したがって、その書込みアクセスによって、このビットフィールドがクリアされることはありません。

ビット25:24 **KMOD[1:0]**：キーモード選択

このビットフィールドは、アプリケーションによる SAES キーの使用方法を定義します。

00：通常のキー

01：ラップ･キー

10：共有キー

その他：予約済み

通常キーを選択した場合、SAES_DINレジスタと SAES_DOUTレジスタには特別な使用方法や保護が適用されず、各キー･レジスタは自由に使用できます。

ラップ･キーの選択により、キーレジスタにロードされたキーはAESキーの暗号化または復号化にのみ使用することができます。そのため、ラップ･キー･モードで復号化が選択されると、復号化処理が成功した後で、0として読み出される SAES_DOUTレジスタが SAES キーレジスタに自動的にロードされます。

共有キーの選択により、復号化処理が成功した後で、SAES キーレジスタはKSHAREID[1:0] ビットフィールドに記述されているペリフェラルと共有されます。この共有は、KMOD[1:0] = 10およびKEYVALID = 1である間のみ有効です。復号化が選択されると、復号化処理が成功した後で、0として読み出される SAES_DOUTレジスタが SAES キーレジスタに自動的にロードされます。

KMOD[1:0]が0以外の場合、異なるセキュリティ･ドメイン（セキュアまたは非セキュアなど）に属するアプリケーションが使用できるようにSAESペリフェラルを設定しようとすると、自動的にキーが消去され、KEIFフラグがセットされます。

ビットの書込みを試みても、SAES_SRレジスタのBUSYフラグがセットされている場合や、SAES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **KEYPROT**：キー保護

これをセットすると、ハードウェアベースのキー保護が有効になります。

0：KEYVALIDがセットされ、KEYSEL = 0の場合、アプリケーションは SAES の所有権を、ロードしたキーとともに別のセキュリティコンテキスト（非セキュア、セキュアなど）で動作しているアプリケーションに移行できます。

1：KEYVALIDがセットされている場合、任意のレジスタへのアクセスが検出されるとキーエラーフラグ（KEIF）がセットされます。そのアクセスのセキュリティコンテキスト（セキュア、非セキュアなど）は、キーをロードしたアプリケーションのセキュリティコンテキストとは異なります。

ビットフィールドの書込みを試みても、SAES_SRレジスタのBUSYフラグがセットされている場合や、SAES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット18 **KEYSIZE**：キー長選択

このビットフィールドは、SAES 暗号コアに使用するキーの長さ（ビット単位）を定義します。

0：128

1：256

KMOD[1:0] = 01 または10 の場合、KEYSIZEでは暗号化または復号化するキーの長さも定義されます。

ビットフィールドの書込みを試みても、SAES_SRレジスタのBUSYフラグがセットされている場合や、SAES_CRレジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DMAOUTEN**：DMA出力イネーブル

このビットは、出力フェーズでDMAでのデータ転送を有効または無効にします。

0：無効

1：有効

このビットがセットされると、出力データフェーズ中に SAES によってDMAリクエストが自動生成されます。この機能は、モード1またはモード3がMODE[1:0] ビットフィールドで選択されている場合のみ有効です。モード2（キー導出）では有効ではありません。

ビット11 **DMAINEN**：DMA入力イネーブル

このビットは、入力フェーズでDMAでのデータ転送を有効／無効にします。

0：無効

1：有効

このビットがセットされると、入力データフェーズ中に SAES によってDMAリクエストが自動生成されます。この機能は、モード1またはモード3がMODE[1:0] ビットフィールドで選択されている場合のみ有効です。モード2（キー導出）では有効ではありません。

ビット10:7 予約済みであり、リセット値に保持する必要があります。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット16、6:5 **CHMOD[2:0]**：連鎖モード選択

このビットフィールドで、AES連鎖モードを選択します。

000：電子コードブック（ECB：Electronic codebook）

001：暗号ブロック連鎖（CBC：Cipher-block chaining）

その他：予約済み

SAES_SRレジスタのBUSYフラグがセットされている場合、および書込みアクセスの前に SAES_CRレジスタのENビットがセットされていて、その書込みアクセスによってクリアされない場合、このビットフィールドへの書込みの試みは無視されます。

ビット4:3 **MODE[1:0]**：SAES 動作モード

このビットフィールドで、SAES 動作モードを選択します。

00：モード1：暗号化

01：モード2：キー導出（ECB/CBC復号化のキーの準備）

10：モード3：暗号化

11：予約済み

ビットフィールドの書込みを試みても、SAES_SR レジスタのBUSY フラグがセットされている場合や、SAES_CR レジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット2:1 **DATATYPE[1:0]**：データ型選択

このビットフィールドは、データスワッピングのモードを選択して、SAES_DINR レジスタに書き込まれるデータまたは SAES_DOUTR レジスタから読み出されるデータのフォーマットを定義します。

00：なし

01：ハーフワード（16bit）

10：バイト（8bit）

11：ビット

詳細については、セクション 50.4.11：SAES データレジスタおよびデータスワッピングを参照してください。

ビットフィールドの書込みを試みても、SAES_SR レジスタのBUSY フラグがセットされている場合や、SAES_CR レジスタのENビットが書込みアクセス前にセットされ、当該書込みアクセスによってクリアされていない場合は、無視されます。

ビット0 **EN**：SAES イネーブル

このビットは、SAES ペリフェラルを有効または無効にします。

0：無効

1：有効

このビットをクリアしてからセットすれば、いつでも SAES ペリフェラルを再初期化できます。

このビットは、キー準備完了時に、ハードウェアによって自動的にクリアされます（モード2）。

このビットは、KEYVALID = 0 である限りセットできません。

注：00以外のKMOD[1:0] では、ビットENよりもIPRSTビットを使用してください。

50.7.2 SAES ステータスレジスタ（SAES_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEYVALID	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
								r				r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **KEYVALID**：キー有効フラグ

このビットは、SAES_CRのKEYSIZEによって定義された量のキー情報が SAES_KEYxキーレジスタにロードされると、ハードウェアによってセットされます。

0：有効なキー情報がキーレジスタにありません。SAES_CRのENビットをセットできません。

1：SAES_CRのKEYSIZEによって定義された有効なキー情報がキーレジスタにロードされています。

通常モードでKEYSEL = 0の場合、アプリケーションはキー･レジスタに正しい順序で書き込む必要があります。順序を誤ると SAES_ISRレジスタのKEIFフラグがセットされ、KEYVALIDが0のままになります。

KEYSELが0以外の場合、SAES によってBUSYフラグが自動的にセットされます。キーの読み込みに成功すると、BUSYフラグがクリアされ、KEYVALIDがセットされます。エラーが発生すると SAES_ISRレジスタのKEIFフラグがセットされ、BUSYフラグがクリアされて、KEYVALIDが0のままになります。

KEIFフラグがセットされている場合、アプリケーションは SAES_ICRレジスタを通じてこのフラグをクリアする必要があります。クリアしないとKEYVALIDをセットできません。詳細については、KEIFビットの説明を参照してください。

キー読込みの詳細については、セクション 50.4.12：SAES キーレジスタを参照してください。

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **BUSY**：ビジー

このフラグは、SAES がGCMペイロード**暗号化**フェーズ中にアイドルかビジーかを示します。

0：アイドル

1：ビジー

このフラグは SAES の初期化、RNGからの乱数フェッチ、共有キーのターゲット･ペリフェラルへの転送時にもセットされます。

ビット2 **WRERR**：書込みエラー

このフラグは、SAES_DINRレジスタへの予期しない書込み操作（計算中またはデータ出力フェーズ中）の検出を示します。

0：未検出

1：検出

フラグはハードウェアによってセットされます。SAES_ISR レジスタの RWEIF ビットをセットすると、ソフトウェアによってクリアされます。

フラグ設定時、SAES_ISR レジスタの RWEIE ビットで有効になっている場合、割込みが生成されます。

このフラグのセットは、SAES 動作に影響しません。予期せぬ書込みは無視されます。

ビット1 **RDERR**：読出しエラーフラグ

このフラグは、SAES_DOUTRレジスタからの予期しない読出し操作（計算中またはデータ入力フェーズ中）の検出を示します。

0：未検出

1：検出

フラグはハードウェアによってセットされます。SAES_ISR レジスタの RWEIF ビットをセットすると、ソフトウェアによってクリアされます。

フラグ設定時、SAES_ISR レジスタの RWEIE ビットで有効になっている場合、割込みが生成されます。

このフラグのセットは、SAES 動作に影響しません。予期せぬ読出しにはゼロが返されます。

ビット0 **CCF**：計算完了フラグ

このビットは、SAES_ISRレジスタのCCFビットを反映します。

50.7.3 SAES データ入力レジスタ（SAES_DINR）

アドレスオフセット：0x08

リセット値：0x0000 0000

32bit書込みアクセスタイプのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIN[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIN[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **DIN[31:0]**：入力データワード

入力フェーズでこのビットフィールドに4回連続で書き込むことによって、SAES ペリフェラルに128bit･ブロックの入力データがすべて書き込まれます。1回目から4回目の書込みまで、対応するデータの重みは [127:96]、[95:64]、[63:32]、[31:0] となります。書込みごとに、32bit入力バッファのデータがDATATYPE[1:0] ビットフィールドに応じたデータ･スワッピング･ブロックによって処理されたうえで、AESコアの128bit入力バッファに書き込まれます。

入力データ･ブロックのデータの意味は、SAES の動作モードによって異なります。

- **モード1**（暗号化）：平文
- **モード2**（キー導出）：ビットフィールド未使用（KEYSEL = 0の場合、SAES_KEYRxレジスタを入力に使用）
- **モード3**（復号化）：暗号文

データスワッピング操作の手順は、セクション 50.4.11： 1972ページのSAES データレジスタおよびデータスワッピングを参照してください。

50.7.4 SAES データ出力レジスタ（SAES_DOUTR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

32bit読出しアクセス･タイプのみサポートされます。KMOD[1:0] = 01または10で、MODE[1:0] = 10およびEN = 1のときに、読出しエラーがトリガされた際に読み出されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DOUT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOUT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **DOUT[31:0]**：出力データワード

読出し専用ビットフィールドは、32bit出力バッファをフェッチします。計算完了（CCFセット）時にこのビットフィールドを4回連続で読み出すことによって、実質的には SAES ペリフェラルから128bit･ブロックの出力データがすべて読み出されます。AES コアによって生成されたデータは、出力バッファに達する前にDATATYPE[1:0] ビットフィールドに応じたデータ･スワッピング･ブロックによって処理されます。

1回目から4回目までの読出し操作でのデータの重みは [127:96]、[95:64]、[63:32]、[31:0] になります。

出力データ･ブロックのデータの意味は、SAES の動作モードによって異なります。

- **モード1**（暗号化）：暗号文
- **モード2**（キー導出）：ビットフィールド未使用
- **モード3**（復号化）：平文

データスワッピング操作の手順は、セクション 50.4.11： 1972ページのSAES データレジスタおよびデータスワッピングを参照してください。

50.7.5 SAES キーレジスタ0（SAES_KEYR0）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[31:0]**：暗号化キー、ビット [31:0]

この書込み専用ビットフィールドには、動作モードに応じてAES暗号化または復号化キーのビット [31:0] が含まれます。

- **モード1**（暗号化）、**モード2**（キー導出）：ビットフィールドに書き込む値は暗号化キーです。
- **モード3**（復号化）：ビットフィールドに書き込む値は、復号化に使用する前に導出される暗号化キーです。

SAES_KEYRxレジスタに書き込めるのは、KEYSIZEの値が正しく、SAES ペリフェラルが無効化（SAES_CRレジスタのENビットがクリア）されているときのみです。SAES_SRレジスタのKEYVALIDビットに示すとおり、特殊な書込みシーケンスも必要です。KEYSELが0およびKEYVALID = 0と異なる場合、キーが SAES_KEYRxレジスタに直接ロードされることに注意してください（そのため、キーレジスタへの書込みは無視され、KEIFがセットされます）。

詳細については、セクション 50.4.12： 1974ページのSAES キーレジスタを参照してください。

50.7.6 SAES キーレジスタ1（SAES_KEYR1）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[63:32]**：暗号化キー、ビット [63:32]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.7 SAES キーレジスタ2（SAES_KEYR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[95:80]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[79:64]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[95:64]**：暗号化キー、ビット [95:64]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.8 SAES キーレジスタ3（SAES_KEYR3）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[127:112]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[111:96]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[127:96]**：暗号化キー、ビット [127:96]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.9 SAES 初期化ベクタレジスタ0（SAES_IVR0）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[31:0]**：初期化ベクタ入力、ビット [31:0]

IVI[127:0] フィールドの説明については、セクション 50.4.13： 1975ページのSAES 初期化ベクタレジスタを参照してください。

初期化ベクタは、ECB以外の連鎖モードでのみ使用します。

SAES_IVRxレジスタは、SAES ペリフェラルが無効である場合にのみ書き込むことができます。

50.7.10 SAES 初期化ベクタレジスタ1（SAES_IVR1）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[63:32]**：初期化ベクタ入力、ビット [63:32]
IVI[128:0] ビットフィールドの説明については、SAES_IVR0レジスタを参照してください。

50.7.11 SAES 初期化ベクタレジスタ2（SAES_IVR2）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[95:80]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[79:64]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[95:64]**：初期化ベクタ入力、ビット [95:64]
IVI[128:0] ビットフィールドの説明については、SAES_IVR0レジスタを参照してください。

50.7.12 SAES 初期化ベクタレジスタ3（SAES_IVR3）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[127:112]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[111:96]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **IVI[127:96]**：初期化ベクタ入力、ビット [127:96]
IVI[128:0] ビットフィールドの説明については、SAES_IVR0レジスタを参照してください。

50.7.13 SAES キーレジスタ4（SAES_KEYR4）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[159:144]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[143:128]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[159:128]**：暗号化キー、ビット [159:128]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.14 SAES キーレジスタ5（SAES_KEYR5）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[191:176]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[175:160]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[191:160]**：暗号化キー、ビット [191:160]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.15 SAES キーレジスタ6（SAES_KEYR6）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[223:208]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[207:192]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[223:192]**：暗号化キー、ビット [223:192]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

50.7.16 SAES キーレジスタ7（SAES_KEYR7）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[255:240]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[239:224]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **KEY[255:224]**：暗号化キー、ビット [255:224]

KEY[255:0] ビットフィールドの説明については、SAES_KEYR0レジスタを参照してください。

注： 4〜7のキーレジスタは、256bitのキー長を選択した場合のみ使用されます。128bitのキー長を選択した場合は有効ではありません（この場合、0〜3のキーレジスタのみが使用されます）。

50.7.17 SAES 割込み有効レジスタ（SAES_IER）

アドレスオフセット：0x300

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIE	KEIE	RWEIE	CCFIE
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RNGEIE：**RNGエラー割込みイネーブル

このビットは、RNGEIF（RNGエラーフラグ）がセットされた場合の SAES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

ビット2 **KEIE**：キーエラー割込みイネーブル

このビットは、KEIF（キーエラーフラグ）がセットされている場合に SAES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

ビット1 **RWEIE：**読出しまたは書込みエラー割込みイネーブル

このビットは、RWEIF（読出しおよび/または書込みエラーフラグ）がセットされている場合に SAES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

ビット0 **CCFIE**：計算完了フラグ割込みイネーブル

このビットは、CCF（計算完了フラグ）がセットされている場合に SAES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：有効化（未マスク）

50.7.18 SAES割込みステータスレジスタ（SAES_ISR）

アドレスオフセット：0x304

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIF	KEIF	RWEIF	CCF
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RNGEIF**：RNGエラー割込みフラグ

この読出し専用ビットは、RNGバスインタフェースでエラー（たとえば、不正なエントロピー）が検出されたときに、ハードウェアによってセットされます。

0：RNGバスは機能しています。

1：RNGバスインタフェースでエラー（ランダム·シード取得エラー）が検出されてます。

RNGEIEビットは、アプリケーションが SAES_ICRレジスタの対応するビットをセットしたときにクリアされます。SAES_IERレジスタのRNGEIEビットが事前にセットされている場合、割込みが生成されます。このビットをクリアすることで、RNGペリフェラルから新しい乱数のリロードがトリガされます。

ビット2 **KEIF**：キーエラー割込みフラグ

この読出し専用ビットは、キー情報がキーレジスタへのロードに失敗した場合、またはキーレジスタの使用が禁止されている場合に、ハードウェアによってセットされます。

0：キーエラーは検出されていません。

1：キー情報がキーレジスタへのロードに失敗しました。またはキーレジスタの使用が禁止されています。

SAES_ICRレジスタの対応するビットを設定すると、SAES_IERレジスタのKEIEビットがセットされている場合は、KEIFがクリアされ、割込みが生成されます。

KEIFは、次のエラーによってトリガされます。

–SAES がDHUKへのロードに失敗しました（KEYSEL = 001または100）。

–SAES が正しい順序でのBHKへのロードに失敗しました（KEYSEL = 010または100）。

–AES がSAESペリフェラルによって共有されたキーのロードに失敗しました（KMOD = 10）。

– KEYVALID = 1で（KEYPROT = 1またはKEYSELが0x0以外）の場合、キー（セキュアまたは非セキュア）をロードするアプリケーションのセキュリティ·コンテキストは、SAES_CRまたはSAES_DOUTへのアクセスのセキュリティ属性と一致しません。この場合、KEYVALIDおよびENビットはクリアされます。

–SAES_KEYRxレジスタへの書込みが、次の正しい順序で行われてない。(KEYSIZE = 0の場合、SAES_KEYR0、SAES_KEYR1、SAES_KEYR2、SAES_KEYR3 レジスタの順、またはその逆順。KEYSIZE = 1 の場合、SAES_KEYR0、SAES_KEYR1、SAES_KEYR2、SAES_KEYR3、SAES_KEYR4、SAES_KEYR5、SAES_KEYR6、SAES_KEYR7、またはその逆順)。

KEIFは、アプリケーションソフトウェアでクリアする必要があります。クリアしないと、KEYVALIDがセットされません。

ビット1 **RWEIF**：読出しまたは書込みエラー割込みフラグ

この読出し専用ビットは、SAES_SRレジスタのRDERRまたはWRERRエラーフラグがセットされている場合に、ハードウェアによってセットされます。

0：読出しまたは書込みエラーは検出されていません。

1：読出しまたは書込みエラーが検出されています（詳細は SAES_SRレジスタを参照）。

RWEIFビットは、アプリケーションが SAES_ICRレジスタの対応するビットをセットしたときにクリアされます。SAES_IERレジスタのRWEIEビットが事前にセットされている場合、割込みが生成されます。

このフラグは、キー派生モードが選択されている場合に意味を持ちません。

ビット0 **CCF**：CCF計算完了フラグ

このフラグは計算が完了したかどうかを示します。

0：未完

1：完了

フラグは、計算完了時にハードウェアによってセットされます。SAES_ISR レジスタの CCF ビットをセットすると、このフラグがソフトウェアによってクリアされます。

フラグ設定時、SAES_IER レジスタのCCFIEビットで有効になっている場合、割込みが生成されます。

フラグはDMAOUTENビットが0の場合のみ有効です。DMA_ENが1の場合、ハイのままとなります。

50.7.19 SAES 割込みクリアレジスタ（SAES_ICR）

アドレスオフセット：0x308

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIF	KEIF	RWEIF	CCF
												w	w	w	w

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RNGEIF**：RNGエラー割込みフラグクリア

SAES_ISRレジスタのRNGEIFステータスビットをクリアするためには、アプリケーションはこのビットをセットする必要があります。

ビット2 **KEIF**：キーエラー割込みフラグクリア

このビットをセットすると、SAES_ISRレジスタのKEIFステータスビットがクリアされます。

ビット1 **RWEIF**：読出しまたは書込みエラー割込みフラグクリア

このビットをセットすると、SAES_ISRレジスタのRWEIFステータスビットと、SAES_SRレジスタのRDERRとWRERRの両方のフラグがクリアされます。

ビット0 **CCF**：CCF計算完了フラグクリア

このビットをセットすると、SAES_SRと SAES_ISRのレジスタのCCFステータスビットがクリアされます。

50.7.20 SAES レジスタマップ

表478.SAESレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	SAES_CR	IPRST	KEYSEL[2:0]			KSHAREID[1:0]		KMOD[1:0]		Res.				KEYPROT	KEYSIZE	Res.	CHMOD[2]	Res.	Res.		DMAOUTEN	DMAINEN	Res.	Res.	Res.	Res.	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	リセット値	0	0	0	0	0	0	0	0					0	0		0				0	0					0	0	0	0	0	0	0
0x004	SAES_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEYVALID	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
	リセット値																									0				0	0	0	0
0x008	SAES_DINR	DIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	SAES_DOUTR	DOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	SAES_KEYR0	KEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	SAES_KEYR1	KEY[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	SAES_KEYR2	KEY[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	SAES_KEYR3	KEY[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	SAES_IVR0	IVI[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	SAES_IVR1	IVI[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	SAES_IVR2	IVI[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	SAES_IVR3	IVI[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	SAES_KEYR4	KEY[159:128]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	SAES_KEYR5	KEY[191:160]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	SAES_KEYR6	KEY[223:192]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	SAES_KEYR7	KEY[255:224]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040-0x2FF	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表478.SAESレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x300	SAES_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIE	KEIE	RWEIE	CCFIE
	リセット値																													0	0	0	0
0x304	SAES_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIF	KEIF	RWEIF	CCF
	リセット値																													0	0	0	0
0x308	SAES_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEIF	KEIF	RWEIF	CCF
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

51 ハッシュプロセッサ（HASH）

51.1 概要

ハッシュプロセッサは、セキュアハッシュアルゴリズム（SHA-1、SHA2-224、SHA2-256）、MD5（メッセージダイジェストアルゴリズム5）ハッシュアルゴリズムおよびHMAC（鍵付きハッシュメッセージ認証コード）アルゴリズムに完全に準拠した処理系となっています。HMACはメッセージ認証を必要とするアプリケーションに適しています。

ハッシュプロセッサは、最長（$2^{64} – 1$）ビットのメッセージに対して、FIPS（連邦情報処理規格）で承認された160、224、256 ビットの長さのダイジェストを生成します。また、MD5アルゴリズムの128bitのダイジェストも生成します。

51.2 HASHの主な機能

- 以下に準拠したデータ認証アプリケーションに好適
  - 連邦情報処理規格公報FIPS PUB 180-4、セキュアハッシュ標準（SHA-1およびSHA-2ファミリ）
  - 連邦情報処理規格公報FIPS PUB 186-4、デジタル署名標準（DSS）
  - インターネット技術タスクフォース（IETF）RFC 1321、MD5メッセージダイジェストアルゴリズム
  - インターネット技術タスクフォース（IETF）RFC 2104、HMAC : メッセージ認証のための鍵付きハッシング、および連邦情報処理規格公報FIPS PUB 198-1、鍵付きハッシュメッセージ認証コード（HMAC）
- SHA-1、SHA2-224、SHA2-256、およびMD5 の高速計算
  - SHA-1（またはSHA2-256）アルゴリズムを使用して1個の512bitブロックのデータを処理する場合は、それぞれ82（または66）クロックサイクル
  - MD5アルゴリズムを使用して1個の512bitブロックのデータを処理する場合は66クロックサイクル
- 連続するメッセージブロックから、対応する32bitワードのダイジェストが相互に追加され、メッセージ全体のダイジェストを形成
  - 入力ビット列の内部リトルエンディアン表現に準拠する自動32bitワードスワッピング
  - 対応ワードスワッピング：ビット、バイト、ハーフワードおよび32bitワード
- 512bit（16 × 32bit）のダイジェスト最低ブロックサイズに適合するよう入力ビット列を補完する自動パディング
- 内部入力FIFOに関連付けられた32bitのシングル入力レジスタ（1つのブロックサイズに対応）
- 32bitワードのアクセスでのみ（それ以外はAHBエラーが発生）アクセス可能なAHBスレーブペリフェラル。
- 出力メッセージダイジェストの場合 8 × 32bitワード（H0からH7）
- 1つのチャネルを使用したダイレクトメモリアクセス（DMA）をサポートする自動データフロー制御。
- 4ワードのシングルまたは固定DMAバースト転送
- ブロック単位での割込み可能なメッセージダイジェスト計算
  - 再ロード可能なダイジェストレジスタ
  - DMAを含むハッシュ計算サスペンド／レジュームメカニズム

51.3 ハッシュの実装

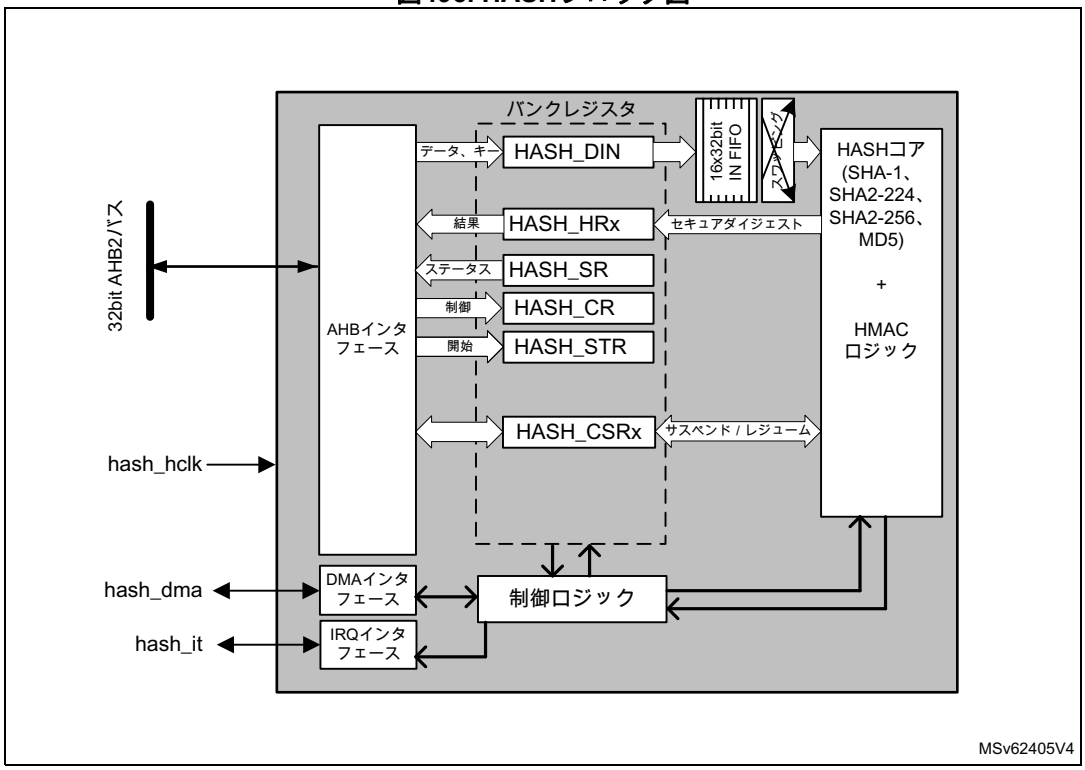
デバイスはHASHペリフェラルのシングルインスタンスを持ちます。

51.4 HASHの機能説明

51.4.1 HASHブロック図

図495に、ハッシュプロセッサのブロック図を示します。

図495. HASHブロック図



51.4.2 HASH内部信号

表479では、製品レベル（パッド上）ではなく、HASHレベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表479.HASH内部入力／出力信号

信号名	信号タイプ	説明
hash_hclk	デジタル入力	AHBバスクロック
hash_it	デジタル出力	ハッシュプロセッサグローバル割込みリクエスト
hash_dma	デジタル入出力	DMA転送リクエスト／確認応答

51.4.3 セキュアハッシュアルゴリズムについて

ハッシュプロセッサは、FIPS PUB 180-4 標準およびIETF RFC1321公報（MD5）によって定義されたセキュアハッシュアルゴリズムに完全に準拠した実装となっています。

各アルゴリズムで、HASHはメッセージやデータファイルの圧縮表記を生成します。より具体的には、入力メッセージの長さが2^{64} ビット未満の場合、HASH処理コアは、メッセージダイジェストと呼ばれる以下に定義された固定長の出力文字列をそれぞれ生成します。

- MD5の場合、ダイジェストサイズは128ビットです。
- SHA-1の場合、ダイジェストサイズは160ビットです。
- SHA2-224およびSHA2-256の場合、ダイジェストサイズはそれぞれ224ビットおよび256ビットです。

その後、メッセージの署名を作成したり検証したりするため、メッセージダイジェストはデジタル署名アルゴリズムを使用して処理することができるようになります。

メッセージ自体ではなくメッセージダイジェストに署名を行うことで、多くの場合処理効率が向上します。これは通常、メッセージダイジェストの方がメッセージよりもはるかにサイズが小さいからです。デジタル署名の検証者は、デジタル署名の作成者が使用したのと同じハッシュアルゴリズムを使用する必要があります。

ハッシュプロセッサがサポートしているSHA-2 機能はNISTによって「安全」であるとされています。それはあるメッセージダイジェストに対応するメッセージの発見、もしくは同じメッセージダイジェストを生成する2つの異なるメッセージの発見が計算上不可能だからです（SHA-1は2017年2月から安全ではないとされています）。メッセージの転送中に何らかの変更が加えられた場合、きわめて高い確率で異なるメッセージダイジェストになり、その署名は検証を通過できません。

51.4.4 メッセージデータの供給

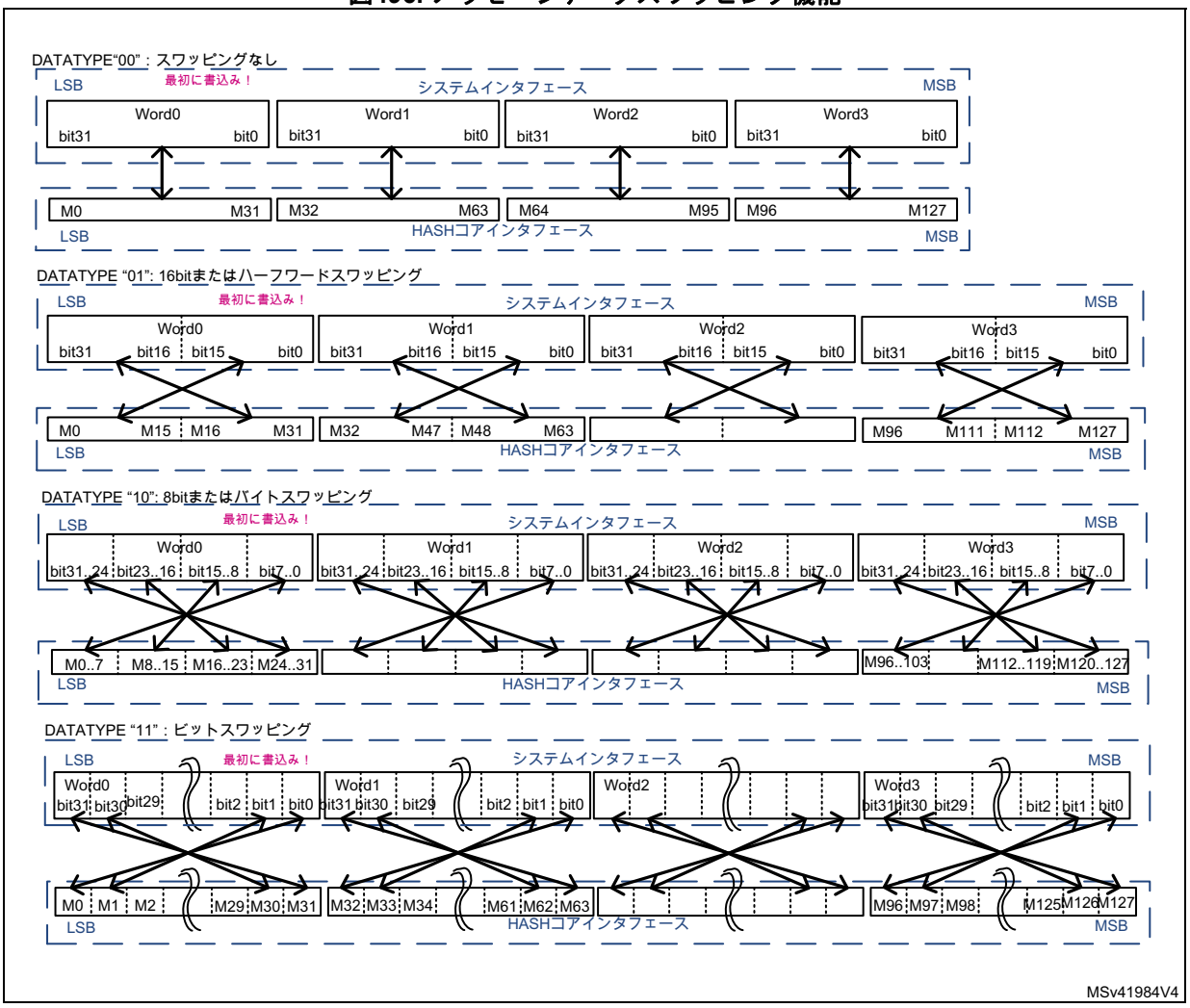
HASHで処理するメッセージ（またはデータファイル）はビット列とみなされます。FIPS PUB 180-4 標準により、このメッセージビット列はビッグエンディアン規約で表記された16進数ワードで左から右に表記されるため、最上位ビットは各ワード内で一番左のビット位置に格納されます。たとえば、「**01100001 01100010 01100011**」のビット列で表されるメッセージ文字列「abc」は、32bitワードで **0x00636261**、8bitワードで **0x61626300** として表されます。

データはHASH_DIN レジスタに書き込むことにより32bit ワードごとにHASH に入力されます。HASH_DINレジスタの現在の内容は、このレジスタに新しいデータが書き込まれるたびに、16ワードの入力FIFOに転送されます。そのため、HASH_DINとFIFOで、17の32bitワード長のFIFOを形成します（INバッファと呼びます）。

処理されるデータの種類（データがASCIIテキストストリームの場合のバイトスワッピングなど）に応じて、入力FIFOからのデータをリトルエンディアンハッシュ処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの動作を実行する必要があります。図496に、HASH制御レジスタ（HASH_CR）のDATATYPEビットフィールドに従って、ハッシュ処理コアの32bitデータブロックM0～31が、ドライバによって入力FIFOに書き込まれる1つの32bitワードからどのように構成されるかを示します。

ビットスワッピングが無効（DATATYPE = 00）になっている場合のHASH_DINデータのエンディアン形式は、次のように記載できます。メッセージの最下位ビットはハッシュプロセッサに入力される最初のワードのMSBの位置でなければならず、ビット列の32番目のビットはハッシュプロセッサに入力される2番目のワードのMSBの位置でなければなりません。それ以降は同様です。

図496. メッセージデータスワッピング機能



51.4.5 メッセージダイジェストの計算

ハッシュプロセッサは、メッセージダイジェストを計算する際に、いくつかのブロックを順次処理します。MD5、SHA1、およびSHA2では、ブロックサイズは512ビットです。

DMAまたはCPUがブロックをハッシュプロセッサに書き込むたびに、HASHは自動的にメッセージダイジェストの計算を開始します。この操作は、部分ダイジェスト計算として知られています。

セクション 51.4.4 : メッセージデータの供給に示すとおり、処理するメッセージは、HASH 32bitワードに入力され、入力FIFOを満たすためにHASH_DINレジスタに書き込まれます。

このデータでハッシュ計算を実行するために、以下の手順がアプリケーションによって使用されます。

1. HASH_CRレジスタを使用してハッシュプロセッサを初期化します。
   a) ALGOビットフィールドを使用して正しいアルゴリズムを選択します。必要に応じて、HASH_CRのDATATYPEビットフィールドを使用してメッセージ入力ワードに正しいスワッピング操作をプログラムします。

b) HMACモードが必要なときには、HMACキーサイズがアルゴリズムの既知のブロックサイズよりも大きい場合は、LKEYビットとともにMODEビットをセットします（そうでない場合は、LKEYはクリアされたままにする）。詳細については、セクション 51.4.7：HMAC操作を参照してください。

c) 有効ビット数が32bitではない場合、NBLW[4:0] を更新してメッセージの最後のワードで有効ビット数を定義します。NBLW[4:0] 情報は、最終メッセージダイジェスト計算の前に、正確に自動メッセージパッディングを行うために使用されます。

2. HASH_CRのINITビットを1にセットして、初期化を完了します。データをDMAで転送する場合、ビットDMAEも1にセットします。

注意： **ステップ2でプログラムする際に、その前あるいはその時に正しい設定値（ALGO、DATATYPE、HMACモード、キー長、NBLW[4:0]）をセットアップすることが重要です。**

3. データがDMAで自動で転送されない場合、HASH_DINレジスタに書き込んでデータ入力を開始します。ブロックの処理はブロックの最終値が 入力FIFOに入力された後で初めて開始できることに注意してください。部分または最終ダイジェスト計算の管理方法は、データのプロセッサへの供給方法によって異なります。

a) データがソフトウェアで入力される場合：

– アプリケーションが次のブロックの最初のワードを書き込むたびに、部分ダイジェスト計算がトリガされます。このとき、ブロックサイズはHASH_CRのNBWEビットとして定義されています。プロセッサが再びレディ（HASH_SRのDINIS = 1）になると、ソフトウェアは新しいデータをHASH_DINに書き込めるようになります。このメカニズムによって、HASHによるウェイトステートの導入を回避します。

– 最終ダイジェスト計算は、最終ブロックが入力され、ソフトウェアがDCALビットに1を書き込む際にトリガされます。メッセージ長がブロックサイズの整数倍ではない場合、DCALビットを書き込む前にHASH_STRレジスタのNBLW[4:0] ビットフィールドを書き込む必要があります（詳細はセクション51.4.6を参照）。

b) データがDMAによってシングルDMA転送（MDMATビット = 0）として入力される場合：

– 部分ダイジェスト計算は、FIFOがフルになるたびに自動的にトリガされます。

最終ダイジェスト計算は、最終ブロックがHASH_DINレジスタに転送された際に自動的にトリガされます（DCALビットがハードウェアによって1にセットされます）。メッセージ長がブロックサイズの整数倍ではない場合、DMAを有効にする前にHASH_STRレジスタのNBLW[4:0] フィールドを書き込む必要があります（詳細はセクション51.4.6を参照）。

c) データがDMAによって多重DMA転送で入力される（MDMAT = 1）場合：

– 部分ダイジェスト計算は、シングルDMA転送に対してトリガされます。ただし、最終ダイジェスト計算は、最終ブロックがDMAによってHASH_DINレジスタに転送された際に自動的にトリガされません（DCALビットがハードウェアによって1にセットされません）。これにより、ハッシュプロセッサは、新しいDMA転送をダイジェスト計算の一部として受信できます。最終ダイジェスト計算を開始するには、シングルDMA転送に対して行うように最終ダイジェスト計算をトリガするために、最後のDMA転送の前に、ソフトウェアでMDMATビットを0にセットする必要があります（上記参照）。

4. いったんダイジェスト計算が完了すると（DCIS = 1）、結果ダイジェストは表480に記載しているとおり、出力レジスタから読み出すことができます。

表480.ハッシュプロセッサ出力

アルゴリズム	有効な出力レジスタ	最上位ビット	ダイジェストサイズ（ビット）
MD5	HASH_H0からHASH_H3	HASH_H0[31]	128
SHA-1	HASH_H0からHASH_H4	HASH_H0[31]	160
SHA2-224	HASH_H0からHASH_H6	HASH_H0[31]	224
SHA2-256	HASH_H0からHASH_H7		256

HMACの詳細な説明については、セクション 51.4.7 : HMAC操作を参照してください。

51.4.6 メッセージのパッディング

概要

メッセージの圧縮表記を計算する際に、データをハッシュプロセッサに（ブロック転送ごとの自動部分ダイジェスト計算で）供給する処理は、オリジナルメッセージの最後のビットがHASH_DINレジスタに書き込まれるまでループします。

メッセージの長さ（ビット数）は任意の整数値をとることができるため、ハッシュプロセッサに書き込まれる最後のワードの有効ビット数は1から32まで可能です。最終メッセージダイジェスト計算の前にメッセージのパディングを正しく行うため、最後のワードの有効ビット数NBLW[4:0] をHASH_STRレジスタに書き込む必要があります。

パディング処理

DMAが有効または無効な状態での詳細なパディング手順については、セクション 51.4.5 : メッセージダイジェストの計算に記載しています。

パディングの例

連邦情報処理規格PUB 180-4 によって指定されているように、メッセージのパディングは、「1」の後に *k* 個の「0」が続き、その後にメッセージのビットの長さ *L* に等しい64bit整数値が続きます。この3つのパディング操作により、長さ *L* + 1 + *k* + 64のパディングされたメッセージが生成され、512bitの倍数となります。

ハッシュプロセッサでは、HASH_DINレジスタに対して最後に書き込まれたワードの中のNBLW[4:0]ビットフィールドで定義された位置に、「1」が追加され、残りの上位ビットは「0」にクリアされます。

FIPS PUB180-4からの例

オリジナルメッセージがASCIIバイナリコード形式の「abc」で、長さL = 24であると仮定します。

```
byte 0    byte 1   byte 2   byte 3
01100001 01100010 01100011 uuuuuuuu
<-- HASH_DIN に書かれた最初のワード -->
```

NBLW[4:0] に値24をロードする必要があります。ビット列のビット位置24（上記ビット列で左から右へカウントします）に**「1」**が付加されます。これはHASH_DINレジスタのビット31に対応します（リトルエンディアン規約）。

```
01100001 01100010 01100011 1uuuuuuu
```

L = 24であるため、上記のビット列のビット数は25bitであり、423個の「0」ビットが付加されて448bitになります。

その結果、16進数（ビッグエンディアン形式のバイトワード）で以下のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

2ワード形式のメッセージ長の値L（00000000 00000018）が付加されます。したがって、パディング処理により最終的に得られるメッセージは、16進数（ビッグエンディアン形式のバイトワード）で次のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

ハッシュプロセッサがHASH_DIN入力レジスタ内のバイトをスワップするようプログラムされている（HASH_CRでDATATYPE = 10）場合、次の手順に従って上記メッセージを入力する必要があります。

1. HASH_DINレジスタに **0xUU636261** を書き込みます（ここで、“U”は影響なしを意味します）。
2. HASH_STRレジスタに **0x18** を書き込みます（オリジナルメッセージの長さは24bitなのでHASH_DINレジスタに最後に書き込まれるワードの有効ビット数は24です）。
3. HASH_STRレジスタに **0x10** を書き込んでメッセージのパッディング処理（上記で説明）を開始し、ダイジェスト計算を実行します。
4. ハッシュ計算が完了すると、SHA-1アルゴリズムのメッセージダイジェストがHASH_HRxレジスタ（x = 0～4）で使用できる状態になります。このFIPSの例の場合の期待値は次のとおりです。

```
HASH_HR0 = 0xA9993E36
HASH_HR1 = 0x4706816A
HASH_HR2 = 0xBA3E2571
HASH_HR3 = 0x7850C26C
HASH_HR4 = 0x9CD0D89D
```

51.4.7 HMAC操作

概要

インターネット技術タスクフォースRFC2104およびNIST FIPS PUB 198-1で指定されているように、HMACアルゴリズムは、処理するメッセージをユーザが選択したキーに対して不可逆的に紐づけすることによるメッセージ認証に使用します。このアルゴリズムは2つのネストされたハッシュ操作で構成されています。

```
HMAC(message)= Hash((Key | pad) XOR opad |
                Hash((Key | pad) XOR ipad | message))
```

ここで、

- **opad** = $[0x5C]_n$（外側パッド）、**ipad** = $[0x36]_n$（内側パッド）
- $[X]_n$ はXを *n* 回繰り返すことを表し、*n* は基礎となるハッシュ関数データブロックのサイズと等しくなります（512bitブロックではn = 64）。
- **pad** は、上記で定義された長さ *n* までキーを拡張するために必要な0の連続です。キー長が *n* より大きくなる場合、アプリケーションはまずHash() 関数を使用してキーをハッシュ処理して、その結果のバイト文字列を実際のキーとしてHMACに使用します。
- | は連結演算子を表します。

注： **ハッシュプロセッサのHMACモードは、サポートされているすべてのアルゴリズムで使用できます。**

HMAC処理

HMACを計算するには、4つの各手順が必要です。

1. ソフトウェアは、MODEビットが1の状態でALGOビットを望みのアルゴリズムに対応する値にセットして、INITビットに1を書き込みます。使用するキーが64バイトを超える場合、LKEYビットも1にセットする必要があります。この場合、HMACの仕様で必要とされているように、ハッシュプロセッサは、実際のキーではなくキーのハッシュを使用します。
2. ソフトウェアは、メッセージ文字列のローディングと同じ方法を使用して、内部ハッシュ関数で使用するためのキーを提供します。これは、HASH_DINレジスタにキーデータを書き込んで、その後DCALビットに1を書き込み、HASH_STRレジスタに正しいNBLW[4:0] を書き込んで転送を完了する方法です。

注： **エンディアン形式の詳細については、セクション 51.4.4: メッセージデータの供給を参照してください。**

3. プロセッサが再びレディ（HASH_SRのDINIS = 1）になると、ソフトウェアはメッセージ列をHASH_DINに書き込めるようになります。最終ブロックの最後のワードが入力され、ソフトウェアがHASH_STRレジスタのDCALビットに1を書き込んだとき、メッセージ長がブロックサイズの整数倍ではない場合は、同時にNBLW[4:0] ビットフィールドにも0以外の値を書き込む必要があります。セクション 51.4.4 : メッセージデータの供給の説明のように、メッセージ列を提供するのにDMAを使用することもできます。
4. プロセッサが再びレディ（HASH_SRのDINIS = 1）になると、ソフトウェアは外部ハッシュ関数で使用するためのキーを提供します。これは、HASH_DINレジスタにキーデータを書き込んで、その後DCALビットに1を書き込み、HASH_STRレジスタに正しいNBLW[4:0] を書き込んで転送を完了する方法です。DCIS = 1になり次第、有効な出力レジスタ（HASH_HRx）内でHMACの計算結果を確認できます。

注： **HMAC初期計算遅延は、セクション 51.6 : HASH処理時間に記載しているとおりキーとメッセージの長さによって変わります。**

HMACの例

以下は、NISTによって指定されているHMAC SHA-1アルゴリズム（HASH_CRでALGO = 00およびMODE = 1）の例です。

オリジナルメッセージがASCIIバイナリコード形式の「**`Sample message for keylen = blocklen`**」で、長さL = 34バイトであると仮定します。HASHがスワップなしモードでプログラムされている場合（HASH_CRでDATATYPE = 00）、次のデータをHASH_DINレジスタに連続でロードする必要があります。

1. NISTによって指定されている**内部ハッシュキー**入力（長さ = 64、パディングなし）。キー長 = 64であるため、HASH_CRレジスタのLKEYビットは0にセットされます。

   `00010203 04050607 08090A0B 0C0D0E0F 10111213 14151617`

   `18191A1B 1C1D1E1F 20212223 24252627 28292A2B 2C2D2E2F`

   `30313233 34353637 38393A3B 3C3D3E3F`

2. **メッセージ**入力（長さ = 34、パディング必須）。HASH_STRを **`0x20`** にセットして、メッセージのパディングと内部ハッシュ計算を開始する必要があります（'U'は無視とみなします）。

   `53616D70 6C65206D 65737361 67652066 6F72206B 65796C65`

   `6E3D626C 6F636B6C 656EUUUU`

3. **外部ハッシュキー**入力（長さ = 64、パディングなし）。ここで、内部ハッシュキーと同一のキーを入力します。

4. **最終外部ハッシュ計算**が、HASHによって実行されます。以下のように、HMAC-SHA1ダイジェストの結果がHASH_HRxレジスタ（x = 0～4）で使用できます。

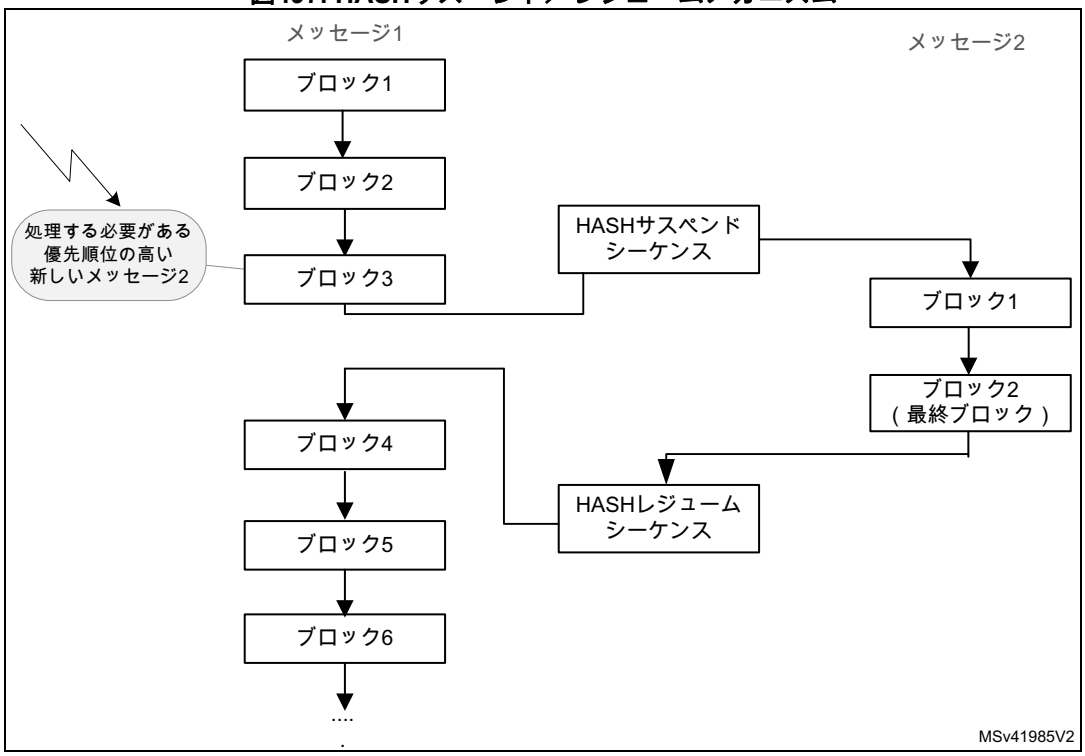
```
HASH_HR0 = 0x5FD596EE
HASH_HR1 = 0x78D5553C
HASH_HR2 = 0x8FF4E72D
HASH_HR3 = 0x266DFD19
HASH_HR4 = 0x2366DA29
```

51.4.8 HASHサスペンド／レジューム操作

概要

ハッシュ／HMAC操作に割り込んでより優先度の高い別の処理を行うことができます。図497に示すように、割り込まれた処理は優先度の高いタスクが処理された後で完了します。

図497. HASHサスペンド／レジュームメカニズム



これを行うには、割り込まれたタスクのコンテキストをHASHレジスタからメモリに保存し、その後HASHレジスタからメモリに復元する必要があります。

データフローをソフトウェアまたはDMAによって制御する手順を以下に示します。

ソフトウェアによってロードされるデータ

DMAがハッシュプロセッサへのメッセージのロードに使用されていない場合、進行中のブロック処理がないときのみコンテキストを保存できます。

メッセージの処理をサスペンドするには、NBWEで定義されているワード数を書き込んだ後に、次の手順に従います。

1. ポーリングモードでは、BUSY = 0になるまで待ち、DINISステータスビットが1にセットされているかどうかをポーリングします。

   割込みモードでは、DINIS割込みハンドラーで次のステップを実施します（推奨）。
2. 次のレジスタの内容をメモリに保存します。
   - HASH_IMR
   - HASH_STR
   - HASH_CR
   - HASH_CSR0からHASH_CSR37。HMAC操作が進行中であった場合は、HASH_CSR38からHASH_CSR53のレジスタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. メモリに保存した値を、HASH_IMR、HASH_STRおよびHASH_CRのレジスタに書き込みます。
2. HASH_CRレジスタのINITビットをセットすることによって、ハッシュプロセッサを初期化します。
3. メモリに保存した値を、HASH_CSRxレジスタに書き込みます。
4. 処理の割り込まれたところから再開します。

DMAによってロードされるデータ

DMAを使用してメッセージをハッシュプロセッサにロードしている場合、以下に説明するように安全なダイジェスト計算をサスペンドして復元することをお勧めします。このシーケンスでは、ハッシュペリフェラルに割り当てられたDMAチャネルは、メッセージ1の処理に割り当てられたままになります（図497を参照）。

DMAを使用してメッセージの処理をサスペンドするには、次の手順に従います。

1. DMAEビットをクリアしてDMAインタフェースを無効にします。ハッシュ·ペリフェラルは、現在の入力ブロックを完了し、ハッシュ処理を始めるために、DMAを使用して十分なデータを自動的にフェッチします。
2. 最後のDMA転送が完了するまで待ちます（HASH_SRでDMAS = 0になるまで）。
3. DMAチャネルを無効にします。
4. ポーリングまたは割込みモード（推奨）で、ハッシュプロセッサがレディになる（処理中のブロックがない状態）まで待ちます。つまり、HASH_SRでDINIS = 1になるのを待ちます。このレジスタにDCISもセットされている場合、ハッシュ結果は利用可能であり、コンテキストスワッピングは無用です。そうではない場合には、ステップ5に進みます。
5. HASH_IMR、HASH_STR、HASH_CR、およびHASH_CSR0からHASH_CSR37のレジスタを保存します。HMAC操作が進行中であった場合は、HASH_CSR38からHASH_CSR53のレジスタも保存する必要があります。

DMAを使用してメッセージの処理をレジュームするには、次の手順に従います。

1. 再度割り込まれないうちに、引き続きメッセージを最後まで転送するためにDMAコントローラを再設定します。
2. メモリに保存した値をHASH_IMR、HASH_STR、およびHASH_CRレジスタにプログラムします。
3. HASH_CRレジスタのINITビットをセットすることによって、ハッシュプロセッサを初期化します。
4. メモリに保存した値をHASH_CSRxレジスタにプログラムします。
5. DMAEビットをセットすることにより処理が割り込まれたところから再開します。

51.4.9 HASH DMAインタフェース

HASHでは、4ワードのシングルおよび固定DMAバースト転送のどちらもサポートしています。

ハッシュプロセッサには、DMAコントローラに接続するインタフェースが搭載されています。このDMAは、HASH_CRレジスタのDMAEビットをセットすることによってHASHにデータを書き込むために使用できます。このビットがセットされると、HASHは、HASH_DINレジスタにブロックを書き込む必要があるたびに、DMAリクエストを開始します。

4つの32bitワードを受信すると、HASHは自動的にDMAへの新しいリクエストをトリガします。詳細については、セクション 51.4.5：メッセージダイジェストの計算を参照してください。

DMA転送を開始する前に、ソフトウェアはHASH_DINレジスタにコピーされる最後のワードでの有効ビット数をプログラムする必要があります。これはHASH_STRレジスタに次の値を書き込むことによって行います。

`NBLW[4:0] = Len(Message)% 32`：ここで、「`x%32`」は、xを32で割った余りを示します。

HASH_SRレジスタのDMASビットから、DMAインタフェース動作に関する情報を得ることができます。このビットはDMAEと共にセットされ、DMAEがクリアされ、DMA転送中ではない場合にクリアされます。

注： **DMASビットに関連する割込みはありません。**

MDMATがセットされているとき、転送のサイズは4ワードの倍数でなければなりません。

51.4.10 HASHエラー管理

ハッシュプロセッサによって生成されるエラーフラグはありません。

51.5 HASH割込み

マスク可能な2つの割込みソースが個別にHASHプロセッサによって生成され、次のイベントが通知されます。

- ダイジェスト計算完了（DCIS）
- データ入力バッファレディ（DINIS）

両方の割込みソースは、同じグローバル割込みリクエスト信号（hash_it）に接続されており、これは、NVIC（ネスト化されたベクタ割込みコントローラ）に接続されています。HASH_IMRレジスタのマスクビットを変更することにより、それぞれの割込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを1にセットすることで、その割込みが有効になります。

マスク可能な各割込みソースのステータスはHASH_SRレジスタから読み出すことができます。表481に、使用可能な機能の概要を示します。

表481.HASH割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法
HASH	ダイジェスト計算完了	DCIS	DCIE	DCISのクリア、またはINITのセット
	新しいブロックを取得するためのデータ入力バッファレディ	DINIS	DINIE	DINISのクリア、またはHASH_DINへの書込み

51.6 HASH処理時間

表482に、各動作モードで中間ブロックの処理に必要な時間を示します。

表482.処理時間（クロックサイクル数）

動作モード	FIFOロード(1)	計算フェーズ	合計
MD5	16	50	**66**
SHA-1	16	66	**82**
SHA2-224	16	50	**66**
SHA2-256			

1. ブロックをプロセッサにロードするために必要な時間を追加してください。

メッセージ（またはHMACのキー）の最終ブロックを処理するのに必要な時間はこれより長い場合があります。この時間は最終ブロックの長さとキー（HMACモードの）の長さによって異なります。

中間ブロックの処理と比べて以下の倍数増える場合があります。

- ハッシュメッセージは **1～2.5倍**
- HMAC入力キーは**～2.5倍**
- HMACメッセージは **1～2.5倍**
- ショートキーの場合のHMAC出力キーは**～2.5倍**
- ロングキーの場合のHMAC出力キーは **3.5～5倍**

51.7 HASHレジスタ

HASHコアは、いくつかの制御レジスタ、ステータスレジスタ、およびいくつかのメッセージダイジェストレジスタと連携しています。これらのレジスタはすべて32bitワード単位でのみアクセス可能です。それ以外はAHBエラーが発生します。

51.7.1 HASH制御レジスタ（HASH_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1:0]		LKEY
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MDMAT	DINNE	NBW[3:0]				Res.	MODE	DATATYPE[1:0]		DMAE	INIT	Res.	Res.
		rw	r	r	r	r	r		rw	rw	rw	rw	rw		

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:17 **ALGO[1:0]**：アルゴリズム選択

このビットは、ハッシュアルゴリズムを選択します。

00：SHA-1

01：MD5

10：SHA2-224

11：SHA2-256

この選択は、INITビットがセットされている場合にのみ有効です。計算中にこのビットフィールドを変更しても影響はありません。

ALGOビットフィールドが更新され、INITビットがセットされると、HASH_SRのNBWEが自動的に0x11に更新されます。

ビット16 **LKEY**：ロングキー選択

このビットによって、HMACモードにおけるショートキー（≤ 64バイト）またはロングキー（> 64バイト）を選択します。

0：HMACキーは64バイト以下です。実際HASH_DINに書き込まれたキー値は、HMAC計算中に使用されます。

1：HMACキーは64バイトを超えます。HMACの計算中には、実際のキーの代わりにキーのハッシュが使用されます。

この選択は、INITとMODEのビットが両方ともセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **MDMAT**：多重DMA転送

多重DMA転送が必要な大容量のファイルをハッシングするときに、このビットをセットします。

0：DCALはDMA転送の最後に自動的にセットされます。

1：DCALはDMA転送の最後に自動的にはセットされません。

ビット12 **DINNE**：DINは空ではありません。

説明については、HASH_SRのDINNEビットを参照してください。

このビットは読出し専用です。

ビット11:8 **NBW[3:0]**：既にプッシュされたワード数

説明については、HASH_SRのNBWP[3:0] ビットフィールドを参照してください。

このビットフィールドは読出し専用です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **MODE**：モード選択

このビットによって、選択したアルゴリズムに対してHASHまたはHMACモードを選択します。

0：ハッシュモードを選択します

1：HMACモードを選択します。使用するキーが64バイトを超える場合、LKEYをセットする必要があります。

この選択は、INITビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット5:4 **DATATYPE[1:0]**：データ型選択

HASH_DINレジスタに入力したデータのフォーマットを定義します。

00：32bitデータ。HASH_DINに書き込まれたデータは順序を変えずに直接HASH処理に使用されます。

01：16bitデータ、またはハーフワード。HASH_DINに書き込まれたデータは2つのハーフワードとみなされ、HASH処理に使用される前にスワップされます。

10：8bitデータ、またはバイト。HASH_DINに書き込まれたデータは4バイトとみなされ、HASH処理に使用される前にスワップされます。

11：ビットデータ、またはビット列。HASH_DINに書き込まれたデータは32bit（ビット位置0におけるデータ列の最初のビット）とみなされ、HASH処理（ビット位置31におけるデータ列の最初のビット）に使用する前にスワップされます。

ビット3 **DMAE**：DMA有効化

0：DMA転送は無効です

1:DMA転送は有効です。DMA要求は、HASHコアがデータを受け取る準備ができるとすぐに送られます。

このビットがセットされると、メッセージの最後のデータがハッシュプロセッサに書き込まれている間、ハードウェアによってクリアされます。

DMA転送中にこのビットを0にセットすると、現在の転送はアボートされません。その代わり、転送が完了するかINITに1が書き込まれるまで、IPのDMAインタフェースは内部的に有効になっています。

INITビットを1にセットすることで、DMAEビットはクリアされません。

ビット2 **INIT**：メッセージダイジェスト計算の初期化

このビットに1を書き込むとハッシュプロセッサコアがリセットされるので、HASHは新しいメッセージのメッセージダイジェストを計算可能となります。

このビットに0を書き込んでも影響はありません。このビットを読み出すと常に0が返ります。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

51.7.2 HASHデータ入力レジスタ（HASH_DIN）

アドレスオフセット：0x04

リセット値：0x0000 0000

HASH_DINはデータ入力レジスタです。このレジスタは32bit幅です。このレジスタを使用して、ブロックごとにメッセージを入力します。HASH_DINレジスタにプログラムを行うと、AHBデータバス上の値がハッシュコアに「プッシュ」され、このレジスタにはAHBデータバス上の新しい値が入ります。正しいメッセージフォーマットを取得するには、事前にHASH_CRレジスタのDATATYPEビットを設定する必要があります。

HASH_DINレジスタにブロック全体が書き込まれると、以下のように中間ダイジェストの計算が開始されます。

- DMAを使用していない場合、HASH_DINレジスタに新しいデータを書き込む（次のブロックの最初のワード）ことにより開始されます（中間ダイジェストの計算）。
- DMAを使用している場合は自動的に開始されます。

HASH_DINレジスタに最後のブロックが書き込まれると、HASH_STRレジスタのDCALビットに1を書き込むこと（最終ダイジェストの計算）によって最終ダイジェストの計算（パディングを含む）が開始されます。この動作は、DMAを使用し、かつMDMATビットが0に設定されている場合は自動的に開始されます。

HASH_DINレジスタを読み出すと、0が返されます。

注： **HASHがビジーの場合、ダイジェスト計算（中間または最終）が完了していないと、HASH_DINレジスタへの書込みアクセスによってAHBバスが停止する可能性があります。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAIN[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAIN[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **DATAIN[31:0]**：データ入力

このレジスタに書き込むと、レジスタの現在の内容がIN FIFOにプッシュされ、レジスタにはAHBデータバス上の新しい値が入ります。

このレジスタを読み出すと、0が返されます。

51.7.3 HASHスタートレジスタ（HASH_STR）

アドレスオフセット：0x08

リセット値：0x0000 0000

HASH_STRレジスタには2つの機能があります。

- ハッシュプロセッサに入力されたメッセージの最後のワードにおける有効ビット数を定義します（HASH_DINレジスタに書き込まれた最後のデータの有効最下位ビット数）。
- DCALビットに1を書き込むことによりメッセージの最後のブロックの処理を開始します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW[4:0]				
							rw				rw	rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **DCAL**：ダイジェスト計算

このビットに1を書き込むと以前書き込まれたNBLW[4:0] の値を使用してメッセージのパッディングが開始され、INITビットに最後に1が書き込まれてから入力FIFOに書き込まれたすべてのデータワードを使用して最終メッセージダイジェストの計算が開始されます。

このビットを読み出すと0が返されます。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **NBLW[4:0]**：最後のワードの有効ビット数

メッセージのビット列の最後のワードがHASH_DINレジスタに書き込まれると、ハッシュプロセッサは内部データスワッピング後に以下に指定された有効ビットのみを取得します。

0x00：書き込まれた最後のデータのすべての32bitが有効メッセージビットです。（M[31:0]）

0x01：書き込まれた最後のデータ（スワッピング後）の1bitだけが有効です。（M[0]）

0x02：書き込まれた最後のデータ（スワッピング後）の2bitだけが有効です。（M[1:0]）

0x03：書き込まれた最後のデータ（スワッピング後）の3bit（すなわち、M[2:0]）だけが有効です。

.....

0x1F：書き込まれた最後のデータ（スワッピング後）の31bit（すなわち、M[30:0]）だけが有効です。

上記メカニズムは、DCAL = 0のときのみ有効です。DCALが1にセットされているときにNBLW[4:0] ビットフィールドに書き込まれた場合、NBLW[4:0] ビットフィールドは変更されません。つまり、同時にNBLW[4:0] を設定しつつDCALをセットすることはできません。

NBLW[4:0] ビットを読み出すと、最後にNBLW[4:0] に書き込まれた値が返されます。

51.7.4 HASHダイジェストレジスタ

これらのレジスタには、次のように指定されたメッセージダイジェストの計算結果が入っています。

- HASH_HR0、HASH_HR1、HASH_HR2、HASH_HR3、HASH_HR4のレジスタは、SHA-1ダイジェストの結果を返します。
- HASH_HR0、HASH_HR1、HASH_HR2、HASH_HR3はMD5で定義されたA、B、C、Dを（それぞれ）返します。
- HASH_HR0からHASH_HR6のレジスタは、SHA2-224 ダイジェストの結果を返します。
- HASH_HR0からHASH_HR7のレジスタは、SHA2-256 ダイジェストの結果を返します。

すべてのケースで、ダイジェストの最上位ビットはHASH_H0[31] に格納され、未使用のHASH_HRxレジスタはゼロとして読み出されます。

ハッシュコアが中間ダイジェストまたは最終メッセージダイジェストを計算している最中に（DCISビットが0のとき）、これらのいずれかのレジスタへの読出しアクセスが実行されると、この読出し操作は0を返します。

注： **（INIT ビットに 1 を書き込むことによって）新しいメッセージのダイジェスト計算を開始するとき、HASH_HRxレジスタは強制的にリセット値になります。**

HASH_HR0からHASH_HR4のレジスタには、2つの異なるアドレスを使用してアクセスできます。

HASHエイリアスダイジェストレジスタx（HASH_HRAx）

アドレスオフセット：0x0C + 0x4 * x（x = 0から4）

リセット値：0x0000 0000

HASH_HRAxレジスタの内容は、アドレスオフセット0x310にあるHASH_HRxレジスタの内容と同じです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Hx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Hx[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **Hx[31:0]**：ハッシュデータx

セクション 51.7.4：HASHダイジェストレジスタの概要を参照してください。

HASHダイジェストレジスタx（HASH_HRx）

アドレスオフセット：0x310 + 0x4 * x（x = 0から4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Hx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Hx[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **Hx[31:0]**：ハッシュデータx

セクション 51.7.4：HASHダイジェストレジスタの概要を参照してください。

HASH補助ダイジェストレジスタx（HASH_HRx）

アドレスオフセット：0x310 + 0x4 * x（x = 5から7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Hx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Hx[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **Hx[31:0]**：ハッシュデータx

セクション 51.7.4：HASHダイジェストレジスタの概要を参照してください。

51.7.5 HASH割込み有効レジスタ（HASH_IMR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCIE	DINIE
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DCIE**：ダイジェスト計算完了割込みイネーブル

0：ダイジェスト計算完了割込みは無効です

1：ダイジェスト計算完了割込みは有効です

ビット0 **DINIE**：データ入力割込みイネーブル

0：データ入力割込みは無効です

1：データ入力割込みは有効です

51.7.6 HASHステータスレジスタ（HASH_SR）

アドレスオフセット：0x24

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBWE[4:0]				
											r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DINNE	Res.	NBWP[4:0]					Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
r		r	r	r	r	r						r	r	rc_w0	rc_w0

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **NBWE[4:0]**：期待されたワード数

このビットフィールドは、部分計算をトリガするためにFIFOにプッシュする必要のあるメッセージ内のワード数を示します。NBWEは、HASH_DINレジスタに書込みアクセスが行われると値が1だけ減少します。NBWEは、HASH_CRでINITビットがセットされると、期待されるブロックサイズ +1（ワード単位）（0x11）に設定され、部分ダイジェスト計算が終了すると、期待されるブロックサイズに設定されます。

ビット15 **DINNE**：DINは空ではありません。

このビットはHASH_DINレジスタに有効なデータがある場合（少なくとも1度は書き込まれたあと）にセットされます。このビットはINITビット（初期化）またはDCALビット（前回のメッセージ処理の完了）のどちらかに1が書き込まれるとクリアされます。

0：データ入力バッファにデータがありません

1：入力バッファには少なくとも1ワードのデータがあります

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:9 **NBWP[4:0]**：既にプッシュされたワード数

このビットフィールドは、既にFIFOにプッシュされているメッセージ内の正確なワード数です。NBWPは、HASH_DINレジスタに書込みアクセスが行われると値が1だけ増加します。

ダイジェスト計算が始まると、NBWPがNBWP - ブロックサイズ（ワード単位）に更新され、INITビットに1が書き込まれるとNBWPはゼロになります。

ビット8:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **BUSY**：ビジービット

0：現在処理中のブロックはありません。

1：ハッシュコアが処理中のデータブロックがあります。

ビット2 **DMAS**：DMAステータス

このビットから、DMAインタフェース動作に関する情報を得ることができます。このビットはDMAEと共にセットされ、DMAE = 0、かつDMA転送中ではない場合にクリアされます。このビットに関連する割込みはありません。

0：DMAインタフェースは無効（DMAE = 0）、かつ転送中ではありません。

1：DMAインタフェースは有効（DMAE = 1）、または転送中です。

ビット1 **DCIS**：ダイジェスト計算完了割込みステータス

このビットは、ダイジェストがレディーになったとき（メッセージ全体が処理済）、ハードウェアによってセットされます。このビットは、そこに0を書き込むかHASH_CRレジスタのINITビットに1を書き込むことによってクリアされます。

0：HASH_HRxレジスタに利用可能なダイジェストはありません（ゼロが返されます）。

1：ダイジェスト計算完了。HASH_HRxレジスタに利用可能なダイジェストがあります。HASH_IMRレジスタのDCIEビットがセットされている場合、割込みが生成されます。

ビット0 **DINIS**：データ入力割込みステータス

このビットは、FIFOが新しいブロックを取得する準備ができたときに（空き領域が16か所）ハードウェアによってセットされます。このビットは、そこに0を書き込むかHASH_DINレジスタに書込みを行うことによってクリアされます。

0：入力バッファの空き領域は16か所未満です。

1：入力バッファに新しいブロックを取り込むことができます。HASH_IMRレジスタのDINIEビットがセットされている場合、割込みが生成されます。

DINIS=0のとき、HASH_CSRxレジスタはゼロとして読み出されます。

51.7.7 HASHコンテキストスワップレジスタ

ハッシュプロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。ハッシュプロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクがハッシュプロセッサを使用する必要があるため、サスペンド／レジューム動作を実行する必要がある場合に役立ちます。

そのようなことが発生した場合、HASH_CSRxレジスタを読み出し、その値をシステムメモリ空間に保存する必要があります。このようにして、優先権のあるタスクがハッシュプロセッサを使用することができ、ハッシュ計算が完了した時点で、保存したコンテキストをメモリから読み出し、HASH_CSRxレジスタに書き戻すことができます。

HASH_CSRxレジスタは、DINISが1に等しいときにのみ読み出せます。そうでないときは、ゼロが返されます。

HASHコンテキストスワップレジスタx（HASH_CSRx）

アドレスオフセット：0x0F8 + x * 0x4（x = 0から 53）

リセット値：0x0000 0002 (HASH_CSR0)

リセット値：0x0000 0000（HASH_CSR1から 53）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CSx[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSx[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CSx[31:0]**：コンテキストスワップx

セクション 51.7.7：HASHコンテキストスワップレジスタの概要を参照してください。

51.7.8 HASHレジスタマップ

表483に、HASHレジスタマップとリセット値の概要を示します。

表483.HASHレジスタのマップとリセット値

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	HASH_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	ALGO0	LKEY	Res.	Res.	MDMAT	DINNE	NBW[3:0]				Res.	MODE	DATATYPE		DMAE	INIT	Res.	Res.
	リセット値														0	0	0			0	0	0	0	0	0		0	0	0	0	0		
0x04	HASH_DIN	DATAIN[31:16]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	HASH_STR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW[4:0]				
	リセット値																								0				0	0	0	0	0
0x0C	HASH_HRA0	H0 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	HASH_HRA1	H1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	HASH_HRA2	H2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	HASH_HRA3	H3 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	HASH_HRA4	H4 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	HASH_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCIE	DINIE
	リセット値																														0	0	
0x24	HASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBWE4	NBWE3	NBWE2	NBWE1	NBWE0	DINNE	Res.	NBWP4	NBWP3	NBWP2	NBWP1	NBWP0	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
	リセット値												0	0	0	0	0	0		0	0	0	0	0						0	0	0	1
0x28- 0xF4	予約済みです。	予約済み																															
0x0F8	HASH_CSR0	CS0 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0x0F8 + 0x4 * x （x = 1から53） 最後のアドレス：0x1CC	HASH_CSRx	CSx[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
.....																																	
0x1D0- 0x30C	予約済みです。	予約済み																															
0x310	HASH_HR0	H0 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表483.HASHレジスタのマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x314	HASH_HR1	H1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x318	HASH_HR2	H2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x31C	HASH_HR3	H3 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x320	HASH_HR4	H4 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x324	HASH_HR5	H5 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x328	HASH_HR6	H6 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x32C	HASH_HR7	H7 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

52 オンザフライ復号化エンジン（OTFDEC）

このセクションは、STM32U545/585/5Ax/5Gxデバイスにのみ適用されます。

52.1 概要

オンザフライ復号化エンジン（On-The-Fly DECryption engine：OTFDEC）により、読出しリクエストのアドレス情報に基づいてAHBトラフィックをオンザフライで復号化できます。OTFDEC では、4つの独立して重複のない暗号化領域を定義できます。

OTFDECはカウンタモードでAES-128を使用して、遅延を最小限に抑えます。その結果、暗号化された領域の内容が変更されるたびに、領域全体を異なる暗号化コンテキスト（鍵または初期化ベクタ）で再度暗号化する必要があります。この制約により、OTFDECが外部NOR Flashに格納された読出し専用データまたはコードの復号化に適したものになっています。

注： **OTFDECをOCTOSPIと連携して使用する場合、Flashメモリコントローラのメモリマップドモードで Flashメモリにアクセスする必要があります。**

製品でセキュリティが有効になっている場合、安全なホストでのみOTFDECをプログラムできます。

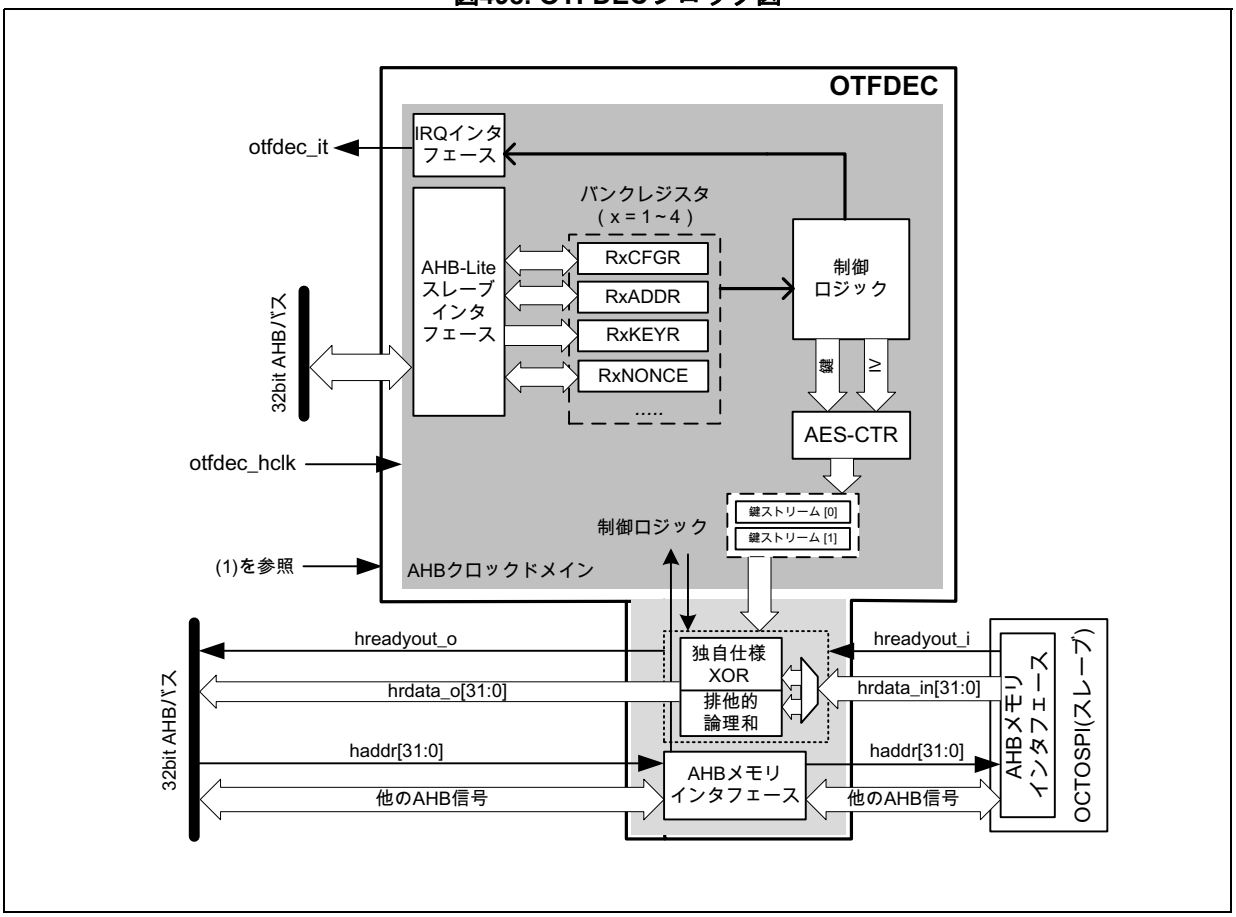
52.2 OTFDECの主な機能

- OCTOSPIのメモリマップド読出し操作（1回または複数回）時のオンザフライ128bit復号化
  - カウンタ（CTR）モードでのAESの使用（2つの128bit鍵ストリームバッファ）
  - あらゆる読出しサイズのサポート
  - 暗号化／復号化に読出しの物理アドレスを使用
- 最大4つの独立した暗号化領域
  - 領域定義の単位：4096バイト
  - 領域設定の書込みロックメカニズム
  - 各領域につき、独自の128bit鍵、2バイトのファームウェアバージョン、8バイトのアプリケーション定義ノンスあり。このうち最低1つはアプリケーションで暗号化するたびに変更が必要
- 暗号鍵の機密性と整合性による保護
  - 書込み専用レジスタ（ソフトウェアロックメカニズム付き）
  - 公開鍵情報として8bit CRCが使用可能
- OCTOSPIプリフェッチメカニズムのサポート
- AES ストリーム暗号の最上部に独自仕様の保護レイヤを追加するための高度な暗号化モードを選択可能（実行専用）
- 32bitワードのシングルアクセスでのみアクセス可能な特権対応の AMBA AHBスレーブペリフェラル（それ以外はAHBバスエラーが発生し、書込みアクセスは無視される）
- TrustZoneセキュリティが製品で有効になっている場合のみ安全なプログラミングが可能
- 暗号化モード

52.3 OTFDECの機能説明

52.3.1 OTFDECブロック図

図498. OTFDECブロック図



1. otfdec_tzen

52.3.2 OTFDEC内部信号

表484では、製品レベル（パッド上）ではなく、OTFDECレベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表484.OTFDEC内部入力／出力信号

信号名	信号タイプ	説明
otfdec_hclk	デジタル入力	AHBバスクロック
otfdec_it	デジタル出力	OTFDECグローバル割込みリクエスト
otfdec_tzen	デジタル入力	OTFDEC TrustZone有効、ペリフェラルのTrustZone機能を制御（TZEN）

FLASHのTZENオプションビットは、デバイスでTrustZoneをアクティブにするために使用します。

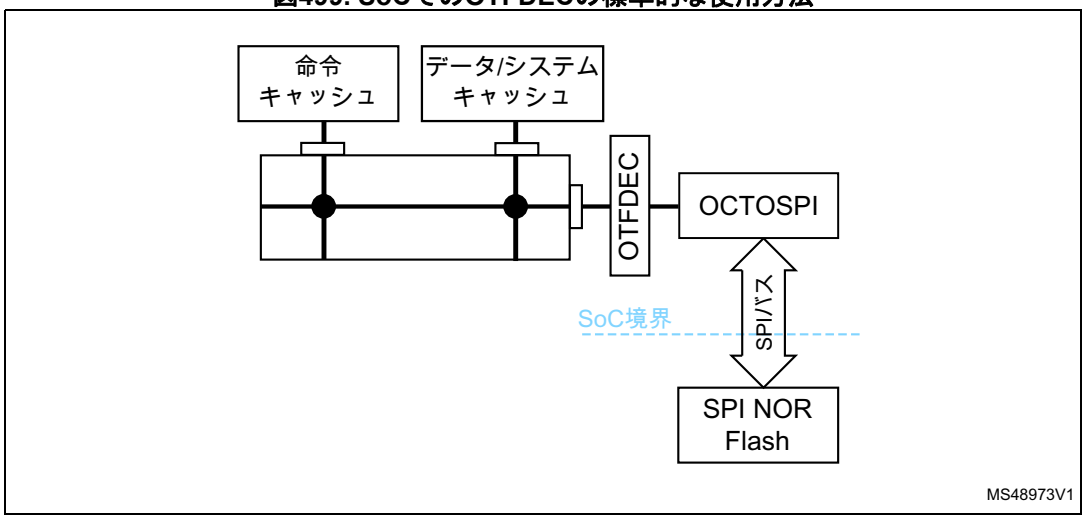
- TZEN = 1：TrustZoneセキュリティが製品で有効になっています。
- TZEN = 0：TrustZoneセキュリティが製品で無効になっています。

52.3.3 OTFDECオンザフライ復号化

概要

OTFDECの標準的な使用方法を図499に示します。

図499. SoCでのOTFDECの標準的な使用方法



OTFDECの本来の目的は、外部SPI NOR Flashデバイスに格納されている読出し専用ファームウェアライブラリの機密性を保護することです。

特別なロック処理がOTFDECで利用でき、復号鍵の整合性を保護し、ソフトウェアのDoS攻撃に対するその他の設定も保護します。ほとんどのレジスタへのOTFDECアクセスは、OTFDEC_PRIVCFGRレジスタのPRIVビットをセットすることで、特権のみにすることができます。OTFDECは、TrustZoneセキュリティが有効化されている場合、TrustZone CPUでのみ書き込むことができます。

OTFDECをOCTOSPIと連携して使用する場合、Flashコントローラのメモリマップドモードで Flashメモリに読出す必要があります。

オンザフライでの復号化に加え、OTFDECは一度に32bitワードずつ暗号化することもできます（詳細については、セクション 52.5.3：OTFDECでの暗号化を参照）。

OTFDECアーキテクチャ

OTFDECは、関連するAHBバスのAHB読出し転送をすべて解析します。読出しリクエストがOTFDECでプログラムされた4つの領域のいずれかにある場合、カウンタモードでのAESアルゴリズムに基づき、制御ロジックによって鍵ストリームの計算がトリガされます。この鍵ストリームを使用して、OCTOSPI AHBマスタからの読出し転送に存在するデータをオンザフライで復号化し、鍵ストリーム情報を計算している間にこのマスタのHREADYOUT信号をローレベルに保持します（これには最大11サイクルかかります）。有効化されたOTFDEC領域以外へのアクセスは、すべて非暗号化領域に属します。

各OTFDEC領域は、x = 1～4のOTFDEC_RxCFGR、OTFDEC_RxSTARTADDR、OTFDEC_RxENDADDR、OTFDEC_RxNONCER、およびOTFDEC_RxKEYRレジスタでプログラムされます。OTFDEC_RxCFGRで、MODEビットはOTFDEC動作モード（標準または高度な暗号化）を定義します。

領域判定の単位は 4096 バイトです。

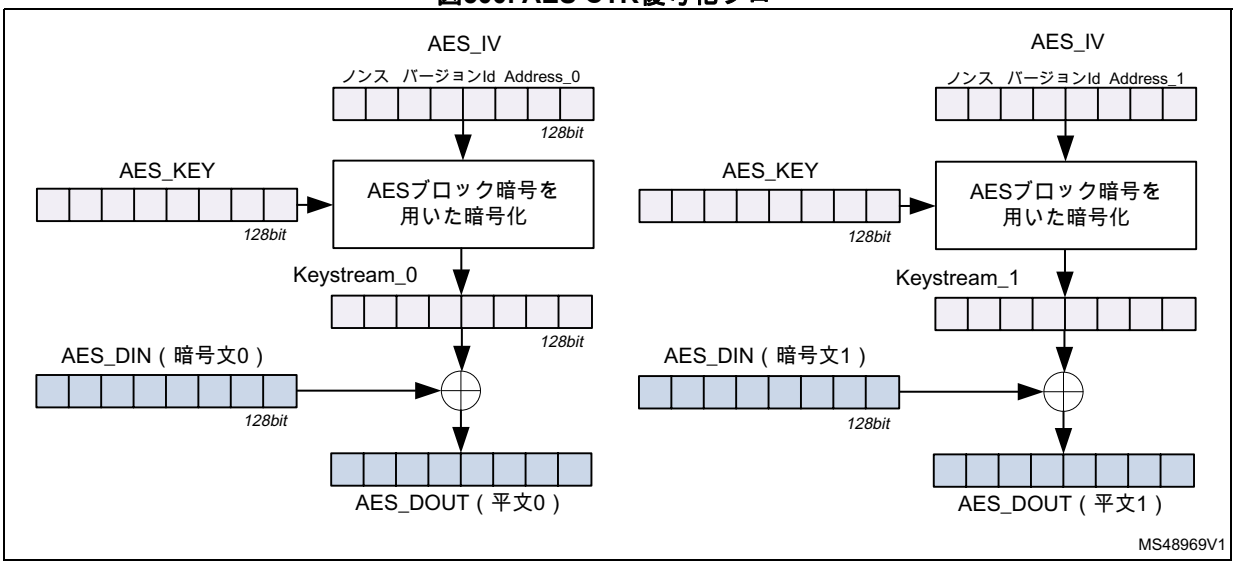
注： **OTFDECで領域の重複を防ぐことはできませんが、これは有効なプログラミングとならないため、アプリケーションソフトウェアで回避する必要があります。**

OTFDECは、4096 バイト整列のアドレス境界に跨らない限り、インクリメンタルバーストやラップバーストを復号化できます。

52.3.4 OTFDECを使用したカウンタモードでのAES復号化

図500に、OTFDECがカウンタ連鎖モードで業界標準の高度暗号化標準（AES）アルゴリズムを使用する流れについて示します。このモードは、NISTによって特別公報800-38A『ブロック暗号の推奨動作モード』で指定されています。

図500. AES CTR復号化フロー



特別な鍵ストリーム情報である128bitデータブロックは、それぞれ以下の定義に従ってAESブロック暗号を使用して生成されます。

- 初期化ベクタAES_IV[127:0] = RxNONCER1[31:0] || RxNONCER0[31:0] || 0b0000 0000 0000 0000 || RxCFGR[31:16] || 0b00 || (x-1) || ReadAddress[31:4]
- 鍵マテリアルAES_KEY[127:0] = R*x*KEYR3[31:0] || R*x*KEYR2[31:0] || R*x*KEYR1[31:0] || R*x*KEYR0[31:0]

注： **上記のxは、選択した暗号化領域のRegionIDです（x = 1～4）。**
ReadAddressは暗号化されたデータブロックのAHBアドレスで、モジュロ128bitです。

結果の128bit鍵ストリームは、128bit暗号文データとの排他的論理和がとられ、128bitの平文データが生成されます。

- AES_DINおよびAES_DOUTのデータブロックは次の規則に従って構成されます（「|」はバイナリ連結を表します）。
  AES_Dx[127:0]= AHB_word(@ | 0xC)[31:0] | AHB_word(@ | 0x8)[31:0] | AHB_word(@ | 0x4)[31:0] || AHB_word(@ | 0x0)[31:0]、ここで、@ は鍵ストリーム（上記ReadAddress[31:4]）の計算に使用される16進数アドレスを指します。

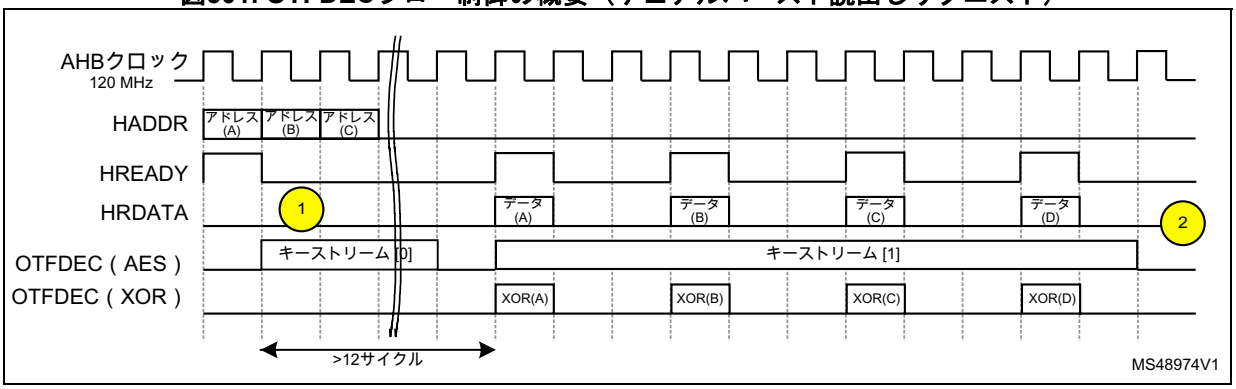
読出しリクエストが暗号化領域内にない場合や、復号化がこの領域で有効になっていない場合、AHBデータは変更されません。

注： アプリケーションによってOTFDEC_RxCFGRのMODEビットフィールドが11にセットされると、さらに保護レイヤがAESストリーム暗号の最上部に追加されます。この高度な暗号化モードは指示がある場合にのみ使用できます（実行専用領域）。

52.3.5 フロー制御管理

図501に、1つの128bit AESデータブロックに対応するINCR4 AHBバーストをOTFDECで管理する方法について示します。

図501. OTFDECフロー制御の概要（デュアルバースト読出しリクエスト）



以下の注意点があります。

1. OTFDECは、データの復号化（鍵ストリームの計算）の準備が整っていないため、AHBマスタからのHREADY信号を強制的にローにします。
2. 鍵ストリームバッファにより、OTFDECがこの設定（120MHz AHBクロック、SPIクロックごとに2バイトを伝達する104MHz SPIバス）で12サイクル以内に新しいデーター式を処理する準備が整います。

52.3.6 OTFDECエラー管理

OTFDECは以下のとおり定義されたエラーを自動的に管理します。

- OTFDEC_RxKEYRレジスタに対する不正読出し。
- OTFDEC_RxCFGRのCONFIGLOCKまたはKEYLOCKが「1」の場合のOTFDEC_RxKEYRレジスタに対する不正書込み（アクセスがセキュアである場合）。セキュリティが製品で無効になっている場合は、アクセスが非セキュアである場合に同じエラーが発生します。
- OTFDEC_RxCFGRのCONFIGLOCKが「1」の場合のOTFDEC_RxKEYR、OTFDEC_RxSTARTADDR、OTFDEC_RxENDADDR、またはOTFDEC_RxNONCERのレジスタに対する不正書込み（x = 1～4）（アクセスがセキュアである場合）。セキュリティが製品で無効になっている場合は、アクセスが非セキュアである場合に同じエラーが発生します。
- 実行専用領域に対する不正読出し（MODE[1:0] = 11）。このような不正リクエストはバスエラーなしで0x0が返されます。
- 暗号化有効時の領域に対する実行リクエスト（ENC = 1）。リクエストはバスエラーなしで0x0が返されます。
- 鍵エラー：鍵レジスタがnullまたは適切に初期化されていない場合（KEYCRC = 0x0）の暗号化領域に対する読出しリクエスト。エラーの原因は、誤った鍵のローディングシーケンス（OTFDEC_RxCFGRのKEYCRCを参照）であるか、アボートイベント（タンパ検出、許可されていないデバッグ接続、信頼できないブート、RDPのレベルのリグレッション）が考えられます。このような読出しリクエストはバスエラーなしで0x0が返されます。

- TrustZoneセキュリティが製品で有効になっており、アクセスが非セキュアである場合のあらゆるレジスタに対する書込み。

  この最後のエラーは、製品リファレンス·マニュアルのGTZCセクションに記載されているとおり、TrustZone割込みコントローラで管理およびクリアされます。

これらのエラーには（最後の1つを除く）、OTFDEC_IERレジスタにSEIE、XONEIE、またはKEIEビットがセットされている場合は、割込みを生成できます。（セクション52.4を参照）。

注： **鍵エラーの後、再度OTFDEC鍵を適切に初期化する必要があり、レジスタがロックされている場合は、OTFDECのリセットが必要になる場合があります。**

52.4 OTFDEC割込み

OTFDECによって生成される個別にマスク可能な割込みソースは3つあり、次のセキュリティイベントが通知されます。

- 鍵に対する不正な読出しまたは書込みアクセス（SEIFフラグ）。セクション52.3.6を参照してください。
- CONFIGLOCK = 1の場合の領域の設定に対する不正書込み（SEIFフラグ）。セクション52.3.6を参照してください。
- XONEIFフラグをトリガする、実行専用領域（MODE[1:0] = 11）への読出しアクセス。
- 暗号化有効時の実行（XONEIFフラグ）。
- KEIFフラグをトリガする鍵エラー（0で読み出される暗号化領域）。セクション52.3.6を参照してください。

割込みソースは、同じグローバル割込みリクエスト信号に接続されます。

表485に示すように、OTFDEC_IERの対応するSEIE、XONEIE、またはKEIEビットをセットすることにより、OTFDEC割込みソースを有効／無効にすることができます。割込みイベントのステータスはOTFDEC_ISRで確認でき、このイベントはOTFDEC_ICRでクリアできます。

表485.OTFDEC割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ(1)	イネーブル制御ビット	割込みのクリア方法
OTFDEC	セキュリティエラー	SEIF	SEIE	OTFDEC_ICRにSEIFをセット
	実行専用暗号化中に実行	XONEIF	XONEIE	OTFDEC_ICRにXONEIFをセット
	鍵エラー	KEIF	KEIE	OTFDEC_ICRにKEIFをセット

1. イベントフラグは、OTFDEC_ISRレジスタ内にあります。

52.5 OTFDECアプリケーション情報

52.5.1 OTFDEC初期化処理

概要

OTFDECの主な側面の1つは、秘密鍵に関わる、信頼性のあるレジスタの初期化です。以下に示す2種類の信頼性のある初期化処理を推奨します。

注： **これらのシーケンスは、ファームウェア開発時に鍵や領域設定のロックを推奨するわけではないため、製品コード向けとなります。**

設定レジスタへの書込みは、領域が有効化されていても、設定のロックによって許可されている場合は有効です。

初期化処理1：すべての領域に対する1つの鍵

この処理では、1つのエンティティが4つの保護された領域の復号化に使用する秘密鍵を所有します。推奨されるOTFDEC設定シーケンスは、以下のとおりです。

1. *x* = 1～4について、OTFDEC_RxCFGRに正しいMODE[1:0] の値を書き込みます。
2. *x* = 1～4について、（有効なCRCを得るために）KEYCRCに記載されているシーケンスでOTFDEC_RxKEYRレジスタをプログラムします。鍵レジスタは書込み専用ということに注意してください。
3. x = 1 ~ 4について、鍵CRCをチェックします。問題なければ、OTFDEC_RxCFGRのKEYLOCKビットをセットします。このビットはクリアできません（この領域の鍵レジスタ x は書き込めなくなります）。
4. 領域xを復号化するには（必ずしも復号鍵を所有するエンティティで実行する必要がないタスク）:
   a) 鍵CRCが領域に格納された暗号化バイナリに対応しているかどうか確認します。
   b) このバイナリに対応する詳細情報（ノンス、開始アドレス、終了アドレス、バージョン番号）を入力します。
   c) REG_ENでこの領域の復号化を有効にします。
   d) OTFDEC_RxCFGRのCONFIGLOCKビットをセットします。このビットはクリアできません（領域設定は書き込めなくなります）。

注意： **指定した領域について、MODEビットが変更されると、鍵レジスタと関連CRCがハードウェアによってクリアされます。そのため、上記の手順1は手順2の前に実行する必要があり、手順2以降はMODEビットを変更しないでください。**

初期化処理2：領域ごとに1つの鍵

この処理では、1つのエンティティが1つ（または複数）の保護された領域の復号化に使用する秘密鍵を所有できます。推奨されるOTFDEC設定シーケンスは、以下のとおりです。

1. 領域xを復号化するには（このタスクは、対応する鍵を所有するエンティティで実行する**必要があります**）：
   a) OTFDEC_RxCFGRに正しいMODE[1:0] の値を書き込みます。
   b) （有効なCRCを得るために）KEYCRCに記載されているシーケンスでOTFDEC_RxKEYRレジスタをプログラムします。鍵レジスタは書込み専用ということに注意してください。
   c) 鍵CRCをチェックします。問題なければ、OTFDEC_RxCFGRのKEYLOCKビットをセットします。このビットはクリアできません（鍵レジスタは書き込めなくなります）。
   d) 保護されたファームウェアに対応する詳細情報（ノンス、開始アドレス、終了アドレス、バージョン番号）を入力します。
   e) REG_ENでこの領域の復号化を有効にします。
   f) OTFDEC_RxCFGRのCONFIGLOCKビットをセットします。このビットはクリアできません（領域設定は書き込めなくなります）。

注意： **指定した領域について、MODEビットが変更されると、鍵レジスタと関連CRCがハードウェアによってクリアされます。そのため、上記の手順a) を手順b) の前に実行する必要があり、手順b) 以降はMODEビットを変更しないでください。**

52.5.2 OTFDECおよび電力管理

OTFDECがリセットされるたびに、セクション52.5.1に示した正しい鍵のローディングシーケンスを実行する必要があります（この場合では、OTFDEC_RxCFGRでKEYCRC = 0）。

OTFDECがハードウェアによってリセットされるたびに、アプリケーションソフトウェアでこのポイントを確認することを推奨します。

52.5.3 OTFDECでの暗号化

コードおよびデータ標準の暗号化

OTFDECは、MODE[1:0] = 10で保護された領域に格納されたバイナリを処理するとき、カウンタモードで標準のAESを使用します。このモードを選択すると、AES互換のハードウェアアクセラレータまたはライブラリを使用して、これらの保護されたライブラリを暗号化できます。以下の高度な暗号化セクションに示すように、OTFDECを使用することもできます（MODE[1:0] = 10）。

AESの入力および出力の定義とエンディアンは、セクション 52.3.4：OTFDECを使用したカウンタモードでのAES復号化で定義されています。

OTFDECの高度な暗号化

OTFDECは、MODE[1:0] = 11で保護された領域に格納されたコードを処理するとき、カウンタモードで標準のAESの最上位にある独自仕様の保護レイヤを使用します。

高度な暗号化モードを使用してタンパに対する堅牢性を高めることができます。

推奨されるOTFDECを使用した暗号化シーケンスは、以下のとおりです。

1. 暗号化を行うアプリケーションが、OTFDEC_CR の ENC ビットをセットします。TrustZone セキュリティが製品で有効になっている場合、このアプリケーションはTrustZoneセキュアモードで実行する必要があります。OTFDEC_PRIVCFGRにPRIVビットがセットされている場合、このアプリケーションには特権がなければなりません。

2. セクション 52.5.1：OTFDEC初期化処理に示すように、暗号化アプリケーションではOTFDECを初期化します。OCTOSPIも、OTFDECが暗号化モードで完全に機能するように、適切にクロック供給する必要があります。この手順は、手順1の前に行うこともできます。
3. 暗号化アプリケーションは32bitの平文データを期待された保護アドレスに書き込み、その後暗号化されたデータをRAMに格納するために同じアドレスに読み戻します。このデータは、暗号化モードではOTFDECによって傍受されるため、デバイス内に保持されます。
4. 暗号化アプリケーションは、すべてのバイナリが処理されるまで、前の手順（アドレスの変更）に戻ります。
5. 暗号化アプリケーションがOTFDEC_CRのENCビットをクリアします。その後、別のアプリケーションで、暗号化されたバイナリを外部Flashの正しいアドレスに一掃できます。

この手順について、いくつかの重要な注意点があります。

- 暗号化粒度は32bitです（32bitのシングルアクセスは必須です）。
- ENC ビットがセットされている場合は、非暗号化領域の読出しは通常のデータを返します（暗号化も復号化もされていないデータなど）。暗号化モードでは、OCTOSPI（レジスタを含む）にアクセスしてはなりません。これは、OTFDECがENCビットがセットされている間のOCTOPSIとの通信を遮断するためです。
- OTFDECは、ENC = 1の間は実行をサポートしません（暗号化されたデータの読出しのみ）。不正実行検出時に、XONEIFフラグが立てられ、ゼロが返されます。

52.5.4 OTFDEC鍵CRCのソースコード

以下は、OTFDEC_RxKEYRレジスタの鍵をロードした後にKEYCRCビットフィールドのOTFDECによって示される計算結果との比較に使用できるCRCソースコードです。

```
uint8_t getCRC(uint32_t * keyin)
{
  const uint8_t CRC7_POLY = 0x7;
  const uint32_t key_strobe[4] = {0xAA55AA55, 0x3, 0x18, 0xC0};
  uint8_t  i, j, k, crc = 0x0;
  uint32_t  keyval;

  for (j = 0; j < 4; j++)
  {
    keyval = *(keyin+j);
    if (j == 0)
    {
      keyval ^= key_strobe[0];
    }
    else
    {
      keyval ^= (key_strobe[j] << 24) | (crc << 16) | (key_strobe[j] << 8)
| crc;
    }

    for (i = 0, crc = 0; i < 32; i++)
    {
      k = (((crc >> 7) ^ (keyval >> (31-i))&0xF)) & 1;
      crc <<= 1;
      if (k)
      {
        crc ^= CRC7_POLY;
       }
    }
    crc^=0x55;
  }
  return crc;
}
```

52.6 OTFDECレジスタ

52.6.1 OTFDEC制御レジスタ（OTFDEC_CR）

アドレスオフセット：0x0

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **ENC**：暗号化モードビット

このビットがセットされると、OTFDECは暗号化モードで使用されます。この間、アプリケーションは平文データを書き、暗号化されたデータを読み戻すことができます。このビットがクリアされると（デフォルト）、OTFDECは復号化モードで使用されます。この間、アプリケーションは復号化されたデータのみを読み戻します。両方のモードで、暗号化コンテキスト（鍵、ノンス、ファームウェアバージョン）は適切に初期化する必要があります。

このビットをセットすると、データアクセスのみが許可されます（それ以外の場合はゼロが返され、XONEIFがセットされます）。MODE = 11の場合は、高度な暗号化モードが自動的に選択されます。

0：OTFDECが復号化モードで動作しています。

1：OTFDECが暗号化モードで動作しています。

注： ENCビットがセットされると、OCTOSPIにはアクセスできなくなります（レジスタおよびメモリマップド領域）。

52.6.2 OTFDEC特権アクセス制御設定レジスタ（OTFDEC_PRIVCFGR）

アドレスオフセット：0x10

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **PRIV**：特権アクセス保護。

0：OTFDECレジスタアクセスで保護が追加されません。

1：OTFDEC_PRIVCFGR以外のすべてのレジスタにアクセスする際に、保護が追加されます。

– レジスタに対する非特権読出しアクセスは0を返します。

– レジスタに対する非特権書込みアクセスは無視されます。

注： このビットは、特権モードでのみ書き込むことができます。読出しに制限はありません。

52.6.3 OTFDEC領域x設定レジスタ（OTFDEC_RxCFGR）

アドレスオフセット：0x20 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

CONFIGLOCKビットが1にセットされている場合、書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_VERSION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIG LOCK	REG_ EN
r	r	r	r	r	r	r	r			rw	rw		rs	rs	rw

ビット31:16 **REG_VERSION[15:0]**：領域のファームウェアバージョン

領域に対応するOTFDEC_RxCFGRのREG_ENビットがセットされる前に、この16bitのビットフィールドを正しく初期化する必要があります。

ビット15:8 **KEYCRC[7:0]**：領域鍵の8bit CRC

KEYLOCK = 0のとき、次のシーケンスで正確にこの領域の鍵をロードしている間に、KEYCRCビットフィールドがハードウェアによって自動計算されます。KEYR0、KEYR1、KEYR2、最後にKEYR3の順です（すべて1回の書込み）。新しい有効なシーケンスが開始するとすぐに新しい計算が開始され、有効なシーケンスが完了するまで、KEYCRCは0として読み出されます。

KEYLOCK = 1のとき、KEYCRCは次のリセットまで変わりません。

CRC計算は、標準のCRC-8-CCITTアルゴリズム$X^8 + X^2 + X + 1$（規則による）を使用した8bitチェックサムです。ソースコードは、セクション52.5.4で確認可能です。

このフィールドは読出し専用です。

注： CRC情報は、鍵の最後のビットが書き込まれた後にのみ更新されます。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **MODE[1:0]**：動作モード

このビットフィールドで、この領域でのOTFDEC動作モードを選択します。

10：すべての読出しアクセスが復号化されます（命令およびデータ）。

11：高度な暗号化モードが有効になり、命令アクセスのみが復号化されます。

その他：予約済み

MODE ≠ 11の場合、標準のAES暗号化モードが有効になります。

MODEビットのいずれかが変更されると、領域鍵と関連CRCが0になります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEYLOCK**：領域鍵のロック

0：この領域のKEYRxレジスタへの書込みが許可されます。

1：この領域のKEYRxレジスタへの書込みが、次のOTFDECリセットまで無視されます。KEYCRCビットフィールドはロックされます。

注： このビットは1回セットされます。すなわち、このビットがセットされると、OTFDECがリセットされた場合のみ0にリセットできます。

ビット1 **CONFIGLOCK**：領域設定のロック

0：この領域のOTFDEC_RxCFGR、OTFDEC_RxSTARTADDR、OTFDEC_RxENDADDR、およびOTFDEC_RxNONCERyのレジスタへの書込みが許可されます。

1：この領域のOTFDEC_RxCFGR、OTFDEC_RxSTARTADDR、OTFDEC_RxENDADDR、およびOTFDEC_RxNONCERyのレジスタへの書込みが、次のOTFDECリセットまで無視されます。

注： このビットは1回セットされます。すなわち、このビットがセットされると、OTFDECがリセットされた場合のみ0にリセットできます。このビットをセットすると、KEYLOCKビットが強制的に1になります。

ビット0 **REG_EN**：領域のオンザフライ復号化イネーブル

0：この領域でのオンザフライ復号化が無効になります。

1：この領域でのオンザフライ復号化が有効になります。データは対応する鍵ストリームとの排他的論理和がとられます。

注： このビットをセットしたときに領域のコンテキスト（バージョン、鍵、ノンス）が無効であれば、不要なデータが復号化されます。

52.6.4 OTFDEC領域x開始アドレスレジスタ（OTFDEC_RxSTARTADDR）

アドレスオフセット：0x24 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_START_ADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_START_ADDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REG_START_ADDR[31:0]**：領域AHB開始アドレス

このレジスタは、OTFDEC_RxCFGRレジスタで領域に対応するREG_ENビットがセットされる前に書き込む必要があります。

このレジスタへの書込みは、OTFDEC_RxCFGRレジスタの領域CONFIGLOCKビットがセットされている間に行われた場合、破棄されます。

注： 領域を決定すると、最初の 12 ビット（LSB）と最後の 4 ビット（MSB）は無視されます。

このレジスタに読出しでアクセスする場合、4 MSB ビットと 12 LSB ビットは0を返します。

52.6.5 OTFDEC領域x終了アドレスレジスタ（OTFDEC_RxENDADDR）

アドレスオフセット：0x28 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0FFF

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_END_ADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_END_ADDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REG_END_ADDR[31:0]**：領域AHB終了アドレス

このレジスタは、OTFDEC_RxCFGRレジスタで領域に対応するREG_ENビットがセットされる前に書き込む必要があり、OTFDEC_RxENDADDRは厳密にOTFDEC_RxSTARTADDRより大きい有効な値にする必要があります。

このレジスタへの書込みは、OTFDEC_RxCFGRの領域CONFIGLOCKビットがセットされている間に行われた場合、破棄されます。

注： 領域を決定すると、最初の 12 ビット（LSB）と最後の 4 ビット（MSB）は無視されます。

このレジスタに読出しでアクセスする場合、4 MSB ビットは0 を返し、12 LSB ビットは1を返します。

52.6.6 OTFDEC領域xノンスレジスタ0（OTFDEC_RxNONCER0）

アドレスオフセット：0x2C + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_NONCE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_NONCE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REG_NONCE[31:0]**：領域のノンス、ビット [31:0]

このレジスタは、OTFDEC_RxCFGRで領域に対応するREG_ENビットがセットされる前に書き込む必要があります。

このレジスタでは、OTFDEC_RxCFGRの領域CONFIGLOCKビットがセットされている間に行われた書込みは破棄されます。

52.6.7 OTFDEC領域xノンスレジスタ1（OTFDEC_RxNONCER1）

アドレスオフセット：0x30 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_NONCE[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_NONCE[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **REG_NONCE[63:32]**：領域のノンス、ビット [63:32]
NONCE[63:0] ビットフィールドの説明については、OTFDEC_RxNONCER0レジスタを参照してください。

52.6.8 OTFDEC領域x鍵レジスタ0（OTFDEC_RxKEYR0）

アドレスオフセット：0x34 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_KEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **REG_KEY[31:0]**：領域鍵、ビット [31:0]
このレジスタは、OTFDEC_RxCFGRで領域に対応するREG_ENビットがセットされる前に書き込む必要があります。
このレジスタを読み出すと0の値が返されます。このレジスタへの書込みは、OTFDEC_RxCFGRで領域CONFIGLOCKまたはKEYLOCKビットがセットされている間に行われた場合、破棄されます。

注： アプリケーションによって正常に OTFDEC_RxCFGRおよび OTFDEC_RxKEYRのMODEビットが変更されると、関連するKEYCRCは消去されます。

52.6.9 OTFDEC領域x鍵レジスタ1（OTFDEC_RxKEYR1）

アドレスオフセット：0x38 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_KEY[63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_KEY[47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **REG_KEY[63:32]**：領域鍵、ビット [63:32]

KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0レジスタを参照してください。

52.6.10 OTFDEC領域x鍵レジスタ2（OTFDEC_RxKEYR2）

アドレスオフセット：0x3C + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_KEY[95:80]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_KEY[79:64]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **REG_KEY[95:64]**：領域鍵、ビット [95:64]

KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0レジスタを参照してください。

52.6.11 OTFDEC領域x鍵レジスタ3（OTFDEC_RxKEYR3）

アドレスオフセット：0x40 + 0x30 * (x-1)(x = 1から4)

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REG_KEY[127:112]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REG_KEY[111:96]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **REG_KEY[127:96]**：領域鍵、ビット [127:96]

KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0レジスタを参照してください。

52.6.12 OTFDEC割込みステータスレジスタ（OTFDEC_ISR）

アドレスオフセット：0x300

リセット値：0x0000 0000

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権読出しは0を返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIF:** 鍵エラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。鍵レジスタがnullまたは適切に初期化されていない場合（KEYCRC = 0x0）に暗号化領域で読出しアクセスが発生すると、ビットがセットされます。

ビットは、アプリケーションがOTFDEC_ICRで対応するビットを1にセットするとクリアされます。

0：OTFDECは適切に動作します。

1：鍵レジスタがnullまたは適切に初期化されていない有効な暗号化領域（KEYCRC = 0x0）で読出しアクセスを検出しました。OTFDECは、読出しに対して0の値を返し、OTFDEC_IERでKEIEビットが1にセットされている場合はオプションで割込みを生成します。

KEIFがセットされると、誤った鍵レジスタでの領域に対する読出しは0の値を返します。これらの鍵レジスタが適切に初期化されるまで、この状態が続きます（KEYCRCが0ではない場合）。

ビット1 **XONEIF:** 実行専用、実行なしエラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。MODEビットが11にセットされた暗号化領域で、命令フェッチではない読出しアクセスが検出されると、ビットがセットされます。最後に、暗号化モードが有効で実行アクセスが検出された場合は、XONEIFもセットされます。

ビットは、アプリケーションがOTFDEC_ICRで対応するビットを1にセットするとクリアされます。

0：実行専用エラーステータスがありません。ペンディング状態の割込みはありません。

1：MODEビットが 11にセットされた ある領域で読出しアクセスが検出されたか、ENC = 1で実行アクセスが検出されました。OTFDECは、不正アクセスに対して0の値を返し、OTFDEC_IERでXONEIEビットが1にセットされている場合はオプションで割込みを生成します。

ビット0 **SEIF:** セキュリティエラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。1つ以上のセキュリティエラーが検出された場合に、ビットがセットされます。

ビットは、アプリケーションがOTFDEC_ICRで対応するビットを1にセットするとクリアされます。

0：セキュリティエラーステータスがありません。ペンディング状態の割込みはありません。

1：セキュリティエラーフラグステータスと、ペンディング状態の割込みがあります。OTFDEC_IERの対応するSEIEビットに応じて、実際の割込みが生成されます。

52.6.13 OTFDEC割込みクリアレジスタ（OTFDEC_ICR）

アドレスオフセット：0x304

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

PRIVビットがOTFDEC_PRIVCFGRでセットされている場合、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
													w	w	w

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIF**：鍵エラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に0が読み出されます。

0：KEIFフラグステータスは影響を受けません。

1：KEIFフラグステータスがOTFDEC_ISRでクリアされます。

注： KEIFをクリアしても、問題の原因（誤った鍵レジスタ）は解決しません。暗号化領域に再びアクセスできるようにするには、OTFDEC鍵レジスタを再度適切に初期化する必要があります。

ビット1 **XONEIF:** 実行専用、実行なしエラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に0が読み出されます。

0：XONEIFフラグステータスは影響を受けません。

1：XONEIFフラグステータスがOTFDEC_ISRレジスタでクリアされます。

ビット0 **SEIF:** セキュリティエラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に0が読み出されます。

0：SEIFフラグステータスは影響を受けません。

1：SEIFフラグステータスがOTFDEC_ISRレジスタでクリアされます。

52.6.14 OTFDEC割込み有効レジスタ（OTFDEC_IER）

アドレスオフセット：0x308

リセット値：0x0000 0000

TrustZoneセキュリティが製品で有効になっている場合、非セキュアAHB書込みアクセス（HNONSEC = 1）は破棄されます。

OTFDEC_PRIVCFGRでPRIVビットがセットされている場合、非特権読出しは0を返し、非特権書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIE	XONEIE	SEIE
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **KEIE：**鍵エラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。KEIFフラグステータスがセットされている場合に、OTFDEC割込み生成を制御します。

0：鍵エラーフラグKEIFでの割込み生成は無効（マスク）です。

1：鍵エラーフラグKEIFでの割込み生成は有効（未マスク）です。

ビット1 **XONEIE:** 実行専用、実行なしエラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。XONEIFフラグステータスがセットされている場合に、OTFDEC割込み生成を制御します。

0：実行専用エラー XONEIFでの割込み生成は無効（マスク）です。

1：実行専用エラー XONEIFでの割込み生成は有効（未マスク）です。

ビット0 **SEIE：**セキュリティエラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。SEIFフラグステータスがセットされている場合に、OTFDEC割込み生成を制御します。

0：セキュリティエラー SEIFでの割込み生成は無効（マスク）です。

1：セキュリティエラー SEIFでの割込み生成は有効（未マスク）です。

52.6.15 OTFDECレジスタマップ

表486.OTFDECレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	OTFDEC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC
	リセット値																																0
0x04 - 0x0C	予約済み	予約済み																															
0x10	OTFDEC_PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV
	リセット値																																0
0x14-0x1C	予約済み	予約済み																															
0x20	OTFDEC_R1CFGR1	REG1_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	OTFDEC_R1STARTADDR	REG1_START_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	OTFDEC_R1ENDADDR	REG1_END_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	OTFDEC_R1NONCER0	REG1_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	OTFDEC_R1NONCER1	REG1_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	OTFDEC_R1KEYR0	REG1_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	OTFDEC_R1KEYR1	REG1_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	OTFDEC_R1KEYR2	REG1_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	OTFDEC_R1KEYR3	REG1_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44 - 0x4C	予約済み	予約済み																															
0x50	OTFDEC_R2CFGR	REG2_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	OTFDEC_R2STARTADDR	REG2_START_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表486.OTFDECレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x58	OTFDEC_R2ENDADDR	REG2_END_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	OTFDEC_R2NONCER0	REG2_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	OTFDEC_R2NONCER1	REG2_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	OTFDEC_R2KEYR0	REG2_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	OTFDEC_R2KEYR1	REG2_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	OTFDEC_R2KEYR2	REG2_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	OTFDEC_R2KEYR3	REG2_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C	予約済み	予約済み																															
0x80	OTFDEC_R3CFGR	REG3_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0
0x84	OTFDEC_R3STARTADDR	REG3_START_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	OTFDEC_R3ENDADDR	REG3_END_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	OTFDEC_R3NONCER0	REG3_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	OTFDEC_R3NONCER1	REG3_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94	OTFDEC_R3KEYR0	REG3_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x98	OTFDEC_R3KEYR1	REG3_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x9C	OTFDEC_R3KEYR2	REG3_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA0	OTFDEC_R3KEYR3	REG3_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4 - 0xAC	予約済み	予約済み																															

表486.OTFDECレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB0	OTFDEC_R4CFGR	REG4_VERSION [15:0]																KEYCRC[7:0]									Res.	Res.	MODE[1:0]		KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0
0xB4	OTFDEC_R4STARTADDR	REG4_START_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB8	OTFDEC_R4ENDADDR	REG4_END_ADDR [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xBC	OTFDEC_R4NONCER0	REG4_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC0	OTFDEC_R4NONCER1	REG4_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC4	OTFDEC_R4KEYR0	REG4_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC8	OTFDEC_R4KEYR1	REG4_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xCC	OTFDEC_R4KEYR2	REG4_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD0	OTFDEC_R4KEYR3	REG4_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0D4 ~0x2FC	予約済み	予約済み																															
0x300	OTFDEC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
	リセット値																														0	0	0
0x304	OTFDEC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
	リセット値																														0	0	0
0x308	OTFDEC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIE	XONEIE	SEIE
	リセット値																														0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

53 公開鍵アクセラレータ（PKA）

このセクションは、STM32U545/585/5Ax/5Gxデバイスにのみ適用されます。

53.1 概要

PKA（Public Key Accelerator：公開鍵アクセラレータ）は、特にRSA、ディフィー - ヘルマン、GF（p）（ガロア体）上のECC（楕円曲線暗号）に関連する暗号公開鍵の基本エレメントの計算を目的としています。手頃なコストで高いパフォーマンスを実現するために、これらの演算はモンゴメリ定義域で実行されます。

特定の演算に対して、必要な計算はすべてアクセラレータ内で実行されるため、入力や出力の処理のために追加でハードウェア／ソフトウェアで何かを行う必要はありません。

秘密情報を扱う場合に、PKAには差動電力解析（DPA）、認定済みのSESIP、およびPSAセキュリティ保証レベル3などの、サイドチャネル攻撃（SCA）に対する保護が組み込まれています。

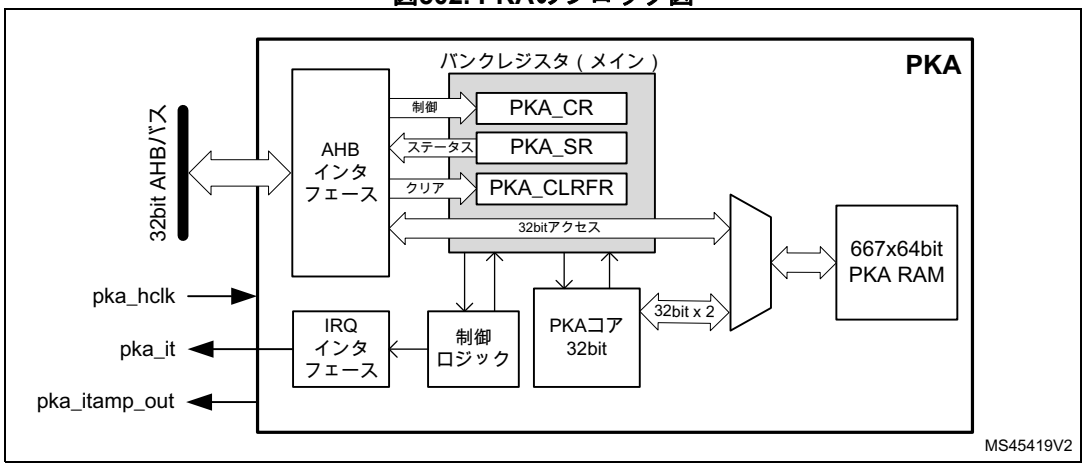
53.2 PKAの主な機能

- RSA、DH、GF（p）でのECCの演算の高速化は、高速剰余乗算に関するモンゴメリ法に基づきます。具体的には次のとおりです。
  - RSAべき剰余、RSA中国剰余定理（CRT）べき乗
  - ECCスカラー乗算、曲線上の点の確認、完全加算、ダブルベースラダー、射影からアフィンへ
  - ECDSA署名の生成および検証
- RSA/DHでは最大4160bit、ECCでは最大640bitのオペランドを処理可能
- 秘密情報を扱う場合：差動電力解析（DPA）、認定済みのSESIP、およびPSAセキュリティ保証レベル3などの、サイドチャネル攻撃（SCA）に対する保護
  - べき剰余、ECCスカラー乗算、ECDSA署名生成に適用可能
- 加算、減算、乗算、モジュラリダクション、モジュラ逆数、比較、モンゴメリ乗算などの算術演算および剰余演算
- モンゴメリ定義域内外への変換を搭載
- 32bitワードのシングルアクセスでのみアクセス可能なAMBA AHBスレーブペリフェラル（それ以外はAHBバスエラーが発生し、書込みアクセスは無視されます）

53.3 PKAの機能詳細

53.3.1 PKAブロック図

図502. PKAのブロック図



53.3.2 PKA内部信号

表487に、PKAレベルで使用可能な内部信号をリストしています。これらの信号は製品接続パッドでは必ずしも使用できるわけではありません。

表487.内部入力／出力信号

信号名	信号タイプ	説明
pka_hclk	デジタル入力	AHBバスクロック
pka_it	デジタル出力	公開鍵アクセラレータIPグローバル割込みリクエスト
pka_itamp_out	デジタル出力	TAMPへのPKA内部改ざんイベント信号（XOR演算されます）。PKAが秘密情報を操作しているときに予期しない障害が発生した場合、またはプログラムされた入力ポイントが入力曲線上で見つからなかった（ECDSA署名およびECCスカラー乗算のみ）場合にトリガされます。 この信号は、障害が検出されるとすぐにアサートされます。アサートされると、PKAレジスタへの読取りアクセスは0にリセットされ、書込みは無視されます。 PKAメモリがクリアされると、この信号はデアサートされます。

53.3.3 PKAのリセットおよびクロック

PKAは、AHBバスクロックでクロック供給されます。PKAペリフェラルのリセット信号がリリースされると、PKA_RAMは667クロックサイクルをかけて自動的にクリアされます。この時間の間、PKA_CRレジスタのENビットのセットは無視されます。

デバイスに適用されるセキュリティポリシーに従って、改ざんイベントの後にPKA RAMをリセットすることもできます。「システムセキュリティ」セクションの「改ざんの検出と対応」を参照してください（本製品に該当する場合）。

53.3.4 PKA公開鍵の高速化

概要

公開鍵アクセラレータ（PKA）は、リベスト、シャミア、エーデルマン（RSA）、ディフィー - ヘルマン（DH）、および素体上のECCの演算を高速化するために使用されます。サポートされるオペランドサイズは、RSAおよびDHでは最大4160bit、ECCでは最大640bitです。

PKAは、素体上で定義されたすべての非特異楕円曲線をサポートします。これは、短いワイエルシュトラス方程式$y^2 = x^3 + ax + b$ (mod p)で記述できます。詳細については、セクション 53.5.1：サポートされている楕円曲線を参照してください。

注： **2値曲線、エドワーズ曲線、およびCurve25519は、PKAではサポートされていません。**

PKA RAMと呼ばれる5336バイト（64bitの667ワード）のメモリが、PKAへの初期データ供給と、計算完了後の結果の保持に使用されます。PKA AHBインタフェースでアクセスします。

PKA演算モード

PKAが実行できる演算のリストを、整数算術機能と素体（Fp）楕円曲線機能についてそれぞれ表488および表489 に示します。

表488.PKA整数算術機能リスト

PKA_CR.MODE[5:0]		実行される演算	参照
16進数	2進数		
0x01	000001	モンゴメリパラメータ計算R2 mod n	セクション53.4.2
0x0E	001110	剰余加算（A+B）mod n	セクション53.4.3
0x0F	001111	剰余減算（A-B）mod n	セクション53.4.4
0x10	010000	モンゴメリ乗算（AxB）mod n	セクション53.4.5
0x00	000000	べき剰余A^e mod n	セクション53.4.6
0x02	000010	べき剰余A^e mod n（高速モード）	
0x03	000011	べき剰余A^e mod n（保護）	セクション53.4.6
0x08	001000	モジュラ逆数A^{-1} mod n	セクション53.4.7
0x0D	001101	モジュラリダクションA mod n	セクション53.4.8
0x09	001001	算術加算A+B	セクション53.4.9
0x0A	001010	算術減算A-B	セクション53.4.10
0x0B	001011	算術乗算AxB	セクション53.4.11
0x0C	001100	算術比較（A=B、A>B、A<B）	セクション53.4.12
0x07	000111	RSA CRTべき乗	セクション53.4.13

表489.PKA素体（Fp）楕円曲線機能リスト

PKA_CR.MODE[5:0]		実行される演算	参照
16進数	2進数		
0x28	101000	楕円曲線Fp上の点の確認	セクション53.4.14
0x20	100000	ECCスカラー乗算kP（保護）	セクション53.4.15
0x23	100011	ECC完全加算	セクション53.4.18
0x24	100100	ECDSA署名 （保護）	セクション53.4.16
0x26	100110	ECDSA検証	セクション53.4.17
0x27	100111	ECCダブルベースラダー	セクション53.4.19
0x2F	101111	ECC射影からアフィンへ	セクション53.4.20

これらの演算モードにはそれぞれ関連するコードがあり、PKA_CRレジスタのMODEフィールドに書き込む必要があります。アプリケーションが上記に記載されていない値を選択すると、MODEビットフィールドへの書込みは無視され、演算エラー（OPERRF）がトリガされます。これが起きた場合、エラーをクリアした後、新しい演算を選択する必要があります。

表488および表489の一部の演算は保護されていると表示されます。これらの演算は、秘密鍵を操作するとき（RSA復号でのべき剰余、ECCでのスカラー乗算および署名）に使用されます。これらの秘密情報（サイドチャネル攻撃から保護されている）は、保護された演算の終了（BUSYがローになる）時にPKA RAMから自動的に消去されます。また、それらはサイドチャネル攻撃から保護されます。

注意： **セキュリティ上の理由から、RSA復号を実行する場合は、保護されたべき剰余演算（MODE = 0x3）を選択することが非常に重要です。**

モンゴメリ空間と高速モードの演算

効率面の理由で、PKAはモンゴメリ定義域でモジュラ乗算演算を内部で実行し、自動的に内外への変換を実行します。

モンゴメリパラメータ計算は時間がかかるため、アプリケーションでは演算開始前に、事前に計算されたモンゴメリパラメータを供給する高速な演算モードを使用することを決定できます。パフォーマンスの改善については、セクション 53.5.2：計算時間で説明します。

高速モードを使用する演算は、べき剰余（MODE = 0x02）だけです。

53.3.5 PKAの標準的なアプリケーション

概要

PKAは、多数の公開鍵暗号化機能の高速化に使用できます。特に、

- RSAの暗号化と復号
- RSAのキー完成
- CRT-RSAの復号
- DSAとECDSA署名の生成および検証
- DHおよびECDHのキーの合意

上記の機能の仕様については、次の公報に示されています。

- FIPS PUB 186-4、デジタル署名標準（DSS）、2013年7月、NIST発行
- PKCS #1、RSA暗号標準v1.5、v2.1およびv2.2.、RSAラボ発行
- IEEE1363-2000、公開鍵暗号向けIEEE標準仕様、2000年1月
- ANSI X9.62-2005、金融サービス業界向け公開鍵暗号、楕円曲線デジタル署名アルゴリズム（ECDSA）、2005年11月

主な機能の原理については、このセクションで説明しています。詳細については、上記に引用した資料を参照してください。

RSAのキーペア

次のRSA演算では、公開鍵と秘密鍵の情報が以下のように定義されています。

- アリスが公開鍵（*n*、*e*）をボブに送信します。数値 *n* および *e*は非常に大きな正の整数です。
- アリスは自身の秘密鍵d（これも非常に大きな正の整数）を公開しないでおきます。その代わりに、この秘密鍵を5つのエレメント（p、q、dp、dq、qInv）で表すこともできます。

上記の表現の詳細については、RSAの仕様を参照してください。

RSAの暗号化と復号の原理

PKCS#1仕様で推奨されるように、ボブはアリスの公開鍵（*n*、*e*）を使用してメッセージMを暗号化するために、次の手順で進める必要があります。

1. エンコードされたメッセージEM = ENCODE(M) を計算します。ここでENCODEはエンコードメソッドを指します。
2. EMを整数*m*（$0 \le m < n$）に変換します。（*m*、*n*）は互いに素です。
3. 暗号文c = m^e mod *n* を計算します。
4. 整数cを文字列の暗号文Cに変換します。

アリスは自身の秘密鍵*d*を使用して暗号文cを復号するために、以下に示した手順に従います。

1. 暗号文Cを整数の暗号文を表すcに変換します。
2. 必要に応じて、(n, e, d）の情報を使用して素因数（p, q）を取得し、phi = (p - 1) * (q - 1) を計算します。詳細は、NIST SP800-56B付録Cを参照してください。
3. 平文*m* = c^d mod *n* = $(m^e)^d$ mod *n*を復元します。秘密鍵が5つのエレメント（p、q、dp、dq、qInv）である場合、平文 *m* を次の演算を実行して取得します。
   a) $m_1 = c^{dp} \bmod p$
   b) $m_2 = c^{dq} \bmod q$
   c) $h = qInv\ (m_1 - m_2) \bmod p$
   d) $m = m_2 + h\ q$
4. 整数メッセージを表す*m*を、エンコードされたメッセージEMに変換します。
5. メッセージM = DECODE(EM) を復元します。ここでDECODEはデコードメソッドを指します。

上記の操作は、PKAで べき剰余 A^e mod *n*（秘密鍵が *d* である場合）、または RSA CRTべき乗（秘密鍵が5つのエレメント（p、q、dp、dq、qInv）である場合）を使用することにより高速化できます。

注： **メッセージと整数のデコード操作と変換操作は、PKCS#1標準で指定されています。**

注： **復号プロセスでは、セキュリティ上の理由から、べき剰余演算の保護されたバージョン（MODE = 0x3）を強くお勧めします。暗号化プロセスでは秘密キーの知識が必要なため、MODE = 0x3は使用できません。**

楕円曲線の選択

次のECC演算では、曲線パラメータが以下のように定義されます。

- 曲線は関係者（アリスとボブ）が合意した楕円曲線フィールドに対応します。サポートされる曲線パラメータの概要をセクション 53.5.1：サポートされている楕円曲線に示しています。
- G は、大きな素数位数 n を持つ選択された楕円曲線の基点（ジェネレータとも言います）です（例：*n* x G = 単位元O）。

ECDSAメッセージの署名生成

ECDSA（楕円曲線デジタル署名アルゴリズム）署名の生成機能の原理は次のとおりです。アリスはメッセージ *m* に自身の秘密鍵の整数d_Aを使用して署名するために、以下の手順に従います。

1. e = HASH(*m*) を計算します。ここでHASHは暗号ハッシュ関数を指します。
2. eの左端のL_n ビットをzとします。ここで、L_n は群位数 *n* のビット長を指します。
3. 暗号として安全なランダムの整数kを選択します（$0 < k < n$）。
4. 曲線の点 $(x_1, y_1) = k \times G$を計算します。
5. $r = x_1 \bmod n$を計算します。*r = 0*の場合、手順3に戻ります。
6. $s = k^{-1} (z + rd_A) \bmod n$を計算します。*s = 0*の場合、手順3に戻ります。
7. 署名はペア（*r*、*s*）となります。

手順4～7はPKAが次のいずれかを使用して高速化します。

- ECDSA署名
- 以下の演算すべて：
  - ECC Fpスカラー乗算 k x P
  - モジュラリダクション A mod *n*
  - モジュラ逆数 $A^{-1} \bmod n$
  - 剰余加算および剰余乗算とモンゴメリ乗算

ECDSA署名の検証

ECDSA（楕円曲線デジタル署名アルゴリズム）署名の検証機能の原理は次のとおりです。ボブはアリスの署名を認証するために、彼女の公開鍵の曲線の点 Q_A のコピーを手に入れる必要があります

ボブは、次の手順で Q_A が有効な曲線の点であることを検証できます。

1. Q_A が単位元Oと等しくないことを確認します。
2. Q_A が合意した曲線上にあることを確認します。
3. n x Q_A = Oであることを確認します。

続いて、ボブは以下に示した手順に従います。

1. *r*と*s*が [1, n-1] 内で整数であることを検証します。
2. *e*= HASH(*m*) を計算します。ここでHASHは合意した暗号ハッシュ関数を指します。
3. *e* の左端のL_n ビットをzとします。
4. $w = s^{-1} \bmod n$ を計算します。
5. $u_1 = zw \bmod n$ および$u_2 = rw \bmod n$ を計算します。
6. 曲線の点 $(x_1, y_1) = u_1 \times G + u_2 \times Q_A$を計算します。
7. $r = x_1 \pmod n$ であれば、署名は有効です。そうでなければ、無効です。

手順4～7はPKAが ECDSA検証を使用して高速化します。

53.3.6 演算を行うためのPKAの手順

PKA有効化／無効化

PKA_CRレジスタでENビットを1にセットすると、PKAペリフェラルが有効になります。PKAはPKA_SRのINITOKビットがセットされたときに使用可能になります。EN = 0のとき、PKAペリフェラルはリセット状態で保持され、PKAメモリにはアプリケーションによってAHBインタフェース経由で引き続き接続可能となります。

注： **PKAがメモリをクリアしている最中は、ENビットをセットできません。**

注： **PKA_CRのENビットをセットするときは、MODEビットフィールドの値が許可されたPKA演算に対応していることを確認してください（セクション53.3.7のOPERRFを参照）。**

計算中にENビットを0にクリアすると、演算がアボートされます。この場合、PKAモードが0x03、0x20、0x24の場合を除き、PKAメモリの内容は保証されません。これらの演算では、PKAメモリはアボート後にクリアされ、メモリは667サイクルの間、使用できなくなります。このクリア時間中はPKAレジスタのみにアクセスでき、ENビットへの書込みは無視されます。

INITOKビットが0のままの場合は、RNGペリフェラルがクロック供給され、適切に初期化されていることを確認してから、PKAを再度有効にしてみてください。

データフォーマット

PKA RAMでの入力データと結果のフォーマットはセクション53.4 に各演算に対して指定されています。

PKA演算の実行

サポートされる各PKA演算は、次の手順によって実行されます。

1. アドレスオフセット0x400にあるPKA内部RAMに初期データをロードします。
2. PKA_CR レジスタの MODE フィールドに実行する演算を指定して書き込んでから、同じくPKA_CRレジスタでSTARTビットをアサートします。
3. PKA_SRレジスタのPROCENDFビットが1にセットされ、計算が完了するまで待ちます。
4. PKA内部RAMから結果データを読み出し、PKA_CLRFRのPROCENDFCビットをセットすることによって、PROCENDFビットをクリアします。

注： **PKAがビジー（BUSY = 1）の場合、アプリケーションによるPKA RAMへのアクセスはすべて無視され、PKA_SRのフラグRAMERRFがセットされます。**

手順 2 で不正または不明な演算を選択すると、OPERRFエラーがトリガされ、手順 3（PROCENDF = 1）は発生しません。詳細については、セクション53.3.7を参照してください。

事前に計算されたモンゴメリパラメータの使用（PKA高速モード）

セクション53.3.4 に示すとおり、同じ係数で多数の演算を行う場合、アプリケーションでは対応するモンゴメリパラメータを一度だけ計算することが有益と考えられます（例：セクション53.4.5を参照）。これは「高速モード」と言われています。

高速モードの使用を管理するためには、以下に説明する手順に従うことを推奨します。

1. PKA RAMで係数のサイズと値の情報をロードします。これらの情報はセクション53.5.1にまとめられています。
2. PKA_CR レジスタで PKA をモンゴメリパラメータ計算モード（MODE=”0x1”）にプログラムして、STARTビットをアサートします。
3. PKA_SRレジスタのPROCENDFビットが1にセットされるまで待ってから、PKAメモリから対応するモンゴメリパラメータを読み出します。その後、PKA_CLRFRのPROCENDFCビットをセットすることによりPROCENDFビットをクリアします。

4. 必要なPKA操作を進め、レギュラ入力データの最上部でモンゴメリの情報R2 mod mをロードします。すべてのアドレスは に示しています。すべてのアドレスをセクション53.4に示しています。

53.3.7 PKAエラー管理

PKAの使用時に、次のいくつかのエラーが発生することがあります。

- PKA RAMへのアクセスが予想の範囲外です。この場合、PKA_SRレジスタのアドレスエラーフラグ（ADDRERRF）がセットされます。
- PKAコアがPKA RAMを使用しているときに、PKA RAMへのAHBアクセスが発生しました。この場合、RAMエラーフラグ（RAMERRF）がPKA_SRレジスタにセットされ、PKA RAMの読出しにはゼロが返され、書込みは無視されます。
- MODEビットフィールドを使用して選択された演算モードは、PKA演算モードのリスト（またはビットフィールドの説明）に含まれません。この場合、演算エラーフラグ（OPERRF）がPKA_SRレジスタにセットされ、MODEビットフィールドへの書込みは無視されます。

上記の各エラーフラグに対して、アプリケーションがPKA_CRレジスタの対応ビットをセットしていれば、PKAによって割込みが生成されます（詳細はセクション53.6を参照してください）。

ADDRERRF、OPERRF、およびRAMERRFエラーは、PKA_CLRFRの対応するビットをセットすることによってクリアされます。

PKAは、PKA_CRレジスタでENビットをリセットすることでいつでも再初期化できます。

新しい演算をPKA_CRレジスタに書き込む前に、PKA_CLRFRのOPERRFCビットを使用してOPERRFエラーをクリアする必要があります。

53.4 PKA演算モード

53.4.1 概要

PKAでサポートされるさまざまな演算について、以下のサブセクションで説明し、PKA RAMに格納される入力データおよび結果のフォーマットを定義します。

警告： **PKAでは、すべての入力パラメータが有効であり、相互に一貫性があると想定しているため、演算を開始する前にPKAへのすべての入力パラメータの有効性をチェックする必要があります。入力パラメータは、演算テーブルで指定されたオペランドのサイズを超えてはなりません。**

以下の情報は、すべてのPKA演算に適用されます。

- PKAコアでは、そのRAMに含まれる64bitワードを処理します。そのため、以降のすべてのワードサイズは64bitになります。
- エレメントをPKAのRAMに入力として書き込む場合、すべてのビットを0にした追加ワードを最上位の入力ワードの後に追加する必要があります。このルールは、オペランドのサイズが1に固定されている場合には適用されません。
- 報告されるすべての RAM ストレージアドレスはデータの最下位ワードを指しており、アプリケーションで使用する実際のアドレスを取得するには、示されたオフセットにPKAのベースアドレスを加える必要があります。

- サポートされるオペランドの「サイズ」は次のとおりです。
  - ROS（RSAのオペランドサイズ）: データサイズは（*rsa_size*/64 + 1）ワードです。*rsa_size* は選択した係数の長さ（ビット単位）です。たとえば、1024bitのオペランドサイズでRSAを計算する場合、ROSは17ワード（1088bit）となります。
  - EOS（ECCのオペランドサイズ）: データサイズは（*ecc_size*/64 + 1）ワードです。*ecc_size* は選択した素数の係数の長さ（ビット単位）です。たとえば、192bitのオペランドサイズでECCを計算する場合、EOSは4ワード（256bit）となります。
  - ROSおよびEOSの値には、必要なオール0の追加ワードが含まれています。
- 特に指定がない限り、表のすべてのオペランドは整数です。

注： **PKA コアは 64bit の複数ワードを処理するため、上記の計算式で端数の結果となった場合は、最も近い整数に繰り上げる必要があります。**

注： **最大ROSは66ワード（4160bitの最大指数サイズ）であり、最大EOSは11ワード（640bitの最大オペランドサイズ）です。**

最初の例として（PKAメモリのエンディアンをよりよく理解するために）、ECC Fpスカラー乗算の演算を準備するために、アプリケーションがECC P256曲線の点Pのx座標（EOS = 5ワード）を書き込むときには、最下位ビットをアドレスオフセット0x578のビット0に、最上位ビットをアドレスオフセット0x590のビット63に配置する必要があります。次に、上述したように、アプリケーションは空のワード0x0000000000000000をアドレスオフセット0x598に書き込む必要があります。

2番目の例として、やはりECC Fpスカラー乗算の演算を準備するために、アプリケーションが224bitの長さの係数（つまり 切り上げて4つの64bitワードに1を加えたもの）を持つ曲線に、a = -3という情報を書く必要があるとき、以下のデータをPKAメモリに書き込む必要があります。

@RAM+410	0x0000000000000001	/* 余分なワードを含まない曲線係数「a」の符号 */
@RAM+418	0x0000000000000011	/* |a| のLSBの値 */
@RAM+420	0x0000000000000000	.....
@RAM+428	0x0000000000000000	.....
@RAM+430	0x0000000000000000	/* |a| のMSBの値 */
@RAM+438	0x0000000000000000	/* すべて0の追加のワード */

53.4.2 モンゴメリパラメータ計算

この機能は、PKAで使用されるモンゴメリパラメータ（R^2 mod *n*）の計算に使用され、オペランドをモンゴメリ剰余系表現に変換します。この演算は、高速モード演算が使用される場合に非常に役立ちます。この場合、モンゴメリパラメータが入力として渡され、計算時間が節約されるためです。

注： **この演算はECC曲線でも使用できます。この場合、素数の係数の長さとEOSサイズを使用する必要があります。**

モンゴメリパラメータ計算に関する演算方法の概要を表490 に示しています。

表490.モンゴメリパラメータ計算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x01	PKA_CR	6bit
	係数の長さ	（ビット単位、0 ≤ 値 < 4160）	RAM@0x408	64bit
	係数の値n	（奇数の整数のみ、n < 2^{4160}）	RAM@0x1088	ROS
OUT	結果：R^2 mod n	-	RAM@0x620	

53.4.3 剰余加算

剰余加算の演算は、A + B mod *n*の計算式となります。演算方法の概要を表491に示しています。

表491.剰余加算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0E	PKA_CR	6bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（0 ≤ A < n）	RAM@0xA50	ROS
	オペランドB	（0 ≤ B < n）	RAM@0xC68	
	係数の値n	（n < 2^{4160}）	RAM@0x1088	
OUT	結果：A+B mod n	（0 ≤ 結果 < n）	RAM@0xE78	

53.4.4 剰余減算

剰余減算の演算は、次の計算式となります。

- A ≥ Bの場合：A - B mod *n*
- A < Bの場合：A + n - B mod *n*

演算方法の概要を表492に示しています。

表492.剰余減算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0F	PKA_CR	6bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（0 ≤ A < n）	RAM@0xA50	ROS
	オペランドB	（0 ≤ B < n）	RAM@0xC68	
	係数の値n	（n < 2^{4160}）	RAM@0x1088	
OUT	結果：A-B mod n	（0 ≤ 結果 < n）	RAM@0xE78	

53.4.5 剰余乗算とモンゴメリ乗算

一連の乗算を実行する場合に効率性を高めるため、PKAはモンゴメリ定義域に少なくとも1つの入力がある乗算を高速化します。この演算の主な2つの使用方法は次のとおりです。

- 自然な定義域からモンゴメリ定義域へ、およびその逆の値のマッピング
- 剰余乗算A x B mod *n* の実行

上記の演算を実行する方法について、以下に記載します。「x」はこの演算を示し、A、B、Cのオペランドは自然な定義域内にあります。

1. モンゴメリ定義域への（からの）変換
   a) Aは自然な定義域内の整数とします。
      r2modn をモンゴメリパラメータ計算で計算します。
      結果AR= A x *r2modn* mod *n*は、モンゴメリ定義域内のAとなります。
   b) BRはモンゴメリ定義域内の整数とします。
      結果B = BR x 1 mod *n*は、自然な定義域内のBとなります。
      同様に、a) で計算される値ARは、A = AR x 1 mod *n*を計算することで自然な定義域に変換できます。
2. 単純な剰余乗算A x B mod *n*
   a) *r2modn* をモンゴメリパラメータ計算で計算します。
   b) AR= A x *r2modn* mod *n* を計算します。出力は、モンゴメリ定義域内です。
   c) AB= AR x *B* mod n を計算します。出力は、自然な定義域内です。
3. 複数の剰余乗算A x B x C mod *n*
   a) *r2modn* をモンゴメリパラメータ計算で計算します。
   b) AR= A x *r2modn* mod *n* を計算します。出力は、モンゴメリ定義域内です。
   c) BR= B x *r2modn* mod *n* を計算します。出力は、モンゴメリ定義域内です。
   d) ABR= AR x *BR* mod n を計算します。出力は、モンゴメリ定義域内です。
   e) CR= C x *r2modn* mod *n* を計算します。出力は、モンゴメリ定義域内です。
   f) ABCR= ABR x *CR* mod n を計算します。出力は、モンゴメリ定義域内です。
   g) （オプション）乗算するオペランドがまだある場合は、上の2つの手順を繰り返します。
   h) ABC= ABCR x 1 mod *n*を計算して、自然な定義域内で結果を取得します。

モンゴメリ乗算に関する演算方法の概要を表493に示しています。

表493.モンゴメリ乗算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x10	PKA_CR	6bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（0 ≤ A < n）	RAM@0xA50	ROS
	オペランドB	（0 ≤ B < n）	RAM@0xC68	
	係数の値n	（奇数の整数のみ、n < 2^{4160}）	RAM@0x1088	
OUT	結果：AxB mod n(1)	-	RAM@0xE78	

1. 入力の性質に応じて、モンゴメリ定義域内または自然な定義域内の結果になります（例 2 と 3 を参照）。

53.4.6 べき剰余

べき剰余の演算は、シングルステップのRSA演算を実行するために通常使用します。これは、A^e mod *n* の計算式となります。

公開情報を含むRSA演算（RSA暗号化）では、表494および 表495に詳述する通常モードまたは高速モードを使用できます。秘密情報を含むRSA演算（RSA復号）では、セキュリティ上の理由から表496に詳述する保護モードを使用する必要があります。

注： **この演算が開始されると、PKA 制御レジスタと PKA メモリは使用できなくなります。PKA によって BUSYビットが0にセットされると、アクセスは回復します。**

この演算が予期しないハードウェアイベントによるエラーで完了すると、TAMPペリフェラルに対してPKA改ざんイベントがトリガされ、ハードウェアによって消去されるまでPKA RAMへのアクセスがブロックされます。

注： **MODE = 0x03のとき、エラー出力が0xD60Dと異なる場合、PKAによってすべてのメモリの内容がクリアされ、秘密鍵に関する情報の漏えいが回避されます。**

べき剰余に関する演算方法の概要を、表494（通常モード）、表495（高速モード）、および表496（保護モード）に示します。高速モードの使用方法は、セクション53.3.6 に示しています。

表494.べき剰余（通常モード）

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x00	PKA_CR	6bit
IN	指数の長さ	（ビット単位、null以外）	RAM@0x400	64bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	
IN/OUT	オペランドA （べき乗の底）	（0 ≤ A < n）	RAM@0xC68	ROS
IN	指数e	（0 ≤ e < n）	RAM@0xE78	
	係数の値n	（奇数の整数のみ、$n < 2^{4160}$）	RAM@0x1088	
OUT	結果：A^e mod n	（0 ≤ 結果 < n）	RAM@0x838	

表495.べき剰余（高速モード）

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x02	PKA_CR	6bit
IN	指数の長さ	（ビット単位、null以外）	RAM@0x400	64bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	
IN/OUT	オペランドA （べき乗の底）	（0 ≤ A < n）	RAM@0xC68	ROS
IN	指数e	（0 ≤ e < n）	RAM@0xE78	
	係数の値n	（奇数の整数のみ、$n < 2^{4160}$）	RAM@0x1088	
IN/OUT	モンゴメリパラメータ R2 mod n	（必須）	RAM@0x620	
OUT	結果：A^e mod n	（0 ≤ 結果 < n）	RAM@0x838	

表496.べき剰余（保護モード）

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x03	PKA_CR	6bit
	指数eの長さ	（ビット単位、null以外）	RAM@0x400	64bit
	係数またはオペランドの長さ	（ビット単位、null以外）	RAM@0x408	
	オペランドA （べき乗の底）	（0 ≤ A < n）	RAM@0x16C8	ROS
	指数e	（0 ≤ e < n）	RAM@0x14B8	
	係数の値n	（奇数の整数のみ、$n < 2^{4096}$）	RAM@0x0838	
	Phi値(1)	-	RAM@0x0C68	
OUT	結果：$A^e \bmod n$	（0 ≤ 結果 < n）	RAM@0x838	
ERROR	$A^e \bmod n$のエラー	– エラーなし：0xD60D – エラーあり：0xCBC9	RAM@0x1298	64bit

1. phi = (p - 1) * (q - 1) によるn^1 のオイラーのトーシェント関数。ここで、pとqは係数nの素因数です（詳細については、NIST SP800-56B付録CまたはRSAの暗号化と復号の原理を参照）。最適化として、キーペア生成の一部としてphi情報を保持することをお勧めします。別の方法は、キーを（*N*, *e*, *d*, *phi*）ではなく（*p*, *q*, *e*, *d*）として保存することです。この場合、PKA算術乗算 *N* = *p* * *q*、および *phi* = (*p* - 1) * (*q* - 1) を使用して *N* と *phi* を導出します。

53.4.7 モジュラ逆数

モジュラ逆数の演算は、逆数$A^{-1} \bmod n$ の計算式となります。係数 *n* が素数の場合、Aのすべての値（1 ≤ A < *n*）で逆数の出力が有効です。係数*n*が素数ではない場合、Aと*n*の最大公約数が1の場合のみAが逆数を持ちます。

オペランドAが係数*n*の約数である場合、結果は*n*の因数の倍数になります。

モジュラ逆数に関する演算方法の概要を表497に示しています。

表497.モジュラ逆数

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x08	PKA_CR	6bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（0 ≤ A < n）	RAM@0xA50	ROS
	係数の値n	（奇数の整数のみ、$n < 2^{4160}$）	RAM@0xC68	
OUT	結果：$A^{-1} \bmod n$	0 < 結果 < n	RAM@0xE78	

53.4.8 モジュラリダクション

モジュラリダクションの演算は、Aをnで割った余りで計算されます。演算方法の概要を 表498に示しています。

表498.モジュラリダクション

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0D	PKA_CR	6bit
	オペランドの長さ	（ビット単位、null以外）	RAM@0x400	64bit
	係数の長さ	（ビット単位、8 < 値 < 4160）	RAM@0x408	
	オペランドA	（$0 \le A < 2n < 2^{4160}$）	RAM@0xA50	ROS
	係数の値n	（奇数の整数のみ、$n < 2^{4160}$）	RAM@0xC68	
OUT	結果A mod n	（0 < 結果 < n）	RAM@0xE78	

53.4.9 算術加算

算術加算の演算は、A + Bの計算式となります。演算方法の概要を表499に示しています。

表499.算術加算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x09	PKA_CR	6bit
	オペランドの長さM	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（$0 \le A < 2^M$）	RAM@0xA50	ROS
	オペランドB	（$0 \le B < 2^M$）	RAM@0xC68	
OUT	結果：A+B	（$0 \le$ 結果 $< 2^{M+1}$）	RAM@0xE78	ROS + 1

53.4.10 算術減算

算術減算の演算は、次の計算式となります。

- A ≥ Bの場合：A - B
- A < BかつM/32の剰余が0を超える場合：$A + 2^{int(M/32)*32+1} - B$
- A < BかつM/32の剰余が0となる場合：$A + 2^{int(M/32)*32} - B$

最後の2つの箇条書きでは、結果が負であるため、出力の最上位ワードに続く 32 ビットワードは 0xFFFF FFFF となります。

演算方法の概要を表500に示しています。

表500.算術減算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0A	PKA_CR	6bit
	オペランドの長さM	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	（$0 \le < A < 2^M$）	RAM@0xA50	ROS
	オペランドB	（$0 \le < B < 2^M$）	RAM@0xC68	
OUT	結果: A-B	（$0 \le$ 結果 $< 2^M$）	RAM@0xE78	

53.4.11 算術乗算

算術乗算の演算は、AxBの計算式となります。演算方法の概要を表501に示しています。

表501.算術乗算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0B	PKA_CR	6bit
	オペランドの長さM	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	$(0 \le A < 2^M)$	RAM@0xA50	ROS
	オペランドB	$(0 \le B < 2^M)$	RAM@0xC68	
OUT	結果：AxB	$(0 \le 結果 < 2^M)$	RAM@0xE78	2xROS

53.4.12 算術比較

算術比較の演算は、次の計算式となります。

- A = B：結果 = 0xED2C
- A > B：結果 = 0x7AF8
- A < B：結果 = 0x916A

算術比較に関する演算方法の概要を表502に示しています。

表502.算術比較

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x0C	PKA_CR	6bit
	オペランドの長さM	（ビット単位、null以外）	RAM@0x408	64bit
	オペランドA	$(0 \le A < 2^M)$	RAM@0xA50	ROS
	オペランドB	$(0 \le B < 2^M)$	RAM@0xC68	
OUT	結果A?B	0xED2C、0x7AF8または0x916A	RAM@0xE78	64bit

53.4.13 RSA CRTべき乗

効率性のために、OpenSSL RSAのような人気のある数多くの暗号ライブラリでは、中国剰余定理（CRT）に基づき、復号化および署名に次の最適化を使用しています。

- *p*および*q*は事前に計算された素数（秘密鍵の一部として格納）
- $d_P = d \bmod (p-1)$
- $d_Q = d \bmod (q-1)$
- $q_{inv} = q^{-1} \bmod p$

これらの値によって、次のようにべき乗$m = A^d \pmod{pq}$をより効率的に計算できます。

- $m_1 = A^{dP} \bmod p$
- $m_2 = A^{dQ} \bmod p$
- $h = q_{inv}(m_1 - m_2) \bmod p$（ここでは$m_1 > m_2$）
- $m = m_2 + hq$

CRTべき乗$A^d \bmod pq$に関する演算方法の概要を表503に示しています。

表503.CRTべき乗

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x07	PKA_CR	6bit
IN	オペランドの長さ	（ビット単位、null以外）	RAM@0x408	64bit
IN	オペランドd_P	（$0 \le d_P < 2^{M/2}$）	RAM@0x730	ROS/2
	オペランドd_Q	（$0 \le d_Q < 2^{M/2}$）	RAM@0xE78	
	オペランドq_{inv}	（$0 \le q_{inv} < 2^{M/2}$）	RAM@0x948	
	素数p(1)	（$0 \le p < 2^{M/2}$）	RAM@0xB60	
	素数 q (1)	（$0 \le q < 2^{M/2}$）	RAM@0x1088	
IN	オペランドA	（$0 \le A < 2^{M/2}$）	RAM@0x12A0	ROS
OUT	結果：$A^d \bmod pq$	（$0 \le$ 結果 $< pq$）	RAM@0x838	

1. 2以外にする必要があります。

53.4.14 楕円曲線Fp上の点の確認

この演算では、指定された点P（x, y）が素体の式 $y^2 = (x^3 + ax + b) \bmod p$（ここで a および b は曲線のエレメントです）の曲線に当てはまるものかどうかを確認します。

楕円曲線Fp上の点の確認に関する演算方法の概要を表504に示しています。

表504.楕円曲線Fp上の点の確認

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x28	PKA_CR	6bit
	係数の長さ	（ビット単位、null以外、 8 < 値 < 640）	RAM@0x408	64bit
	曲線係数 a の符号	0x0：正 0x1：負	RAM@0x410	
	曲線係数 |a|	（絶対値、|a| < p）	RAM@0x418	EOS
	曲線係数b	（|b| < p）	RAM@0x520	
	曲線の係数の値 p	（奇数の整数素数、0 < p < 2640）	RAM@0x470	
	点Pの座標 x	（x < p）	RAM@0x578	
	点Pの座標 y	（y < p）	RAM@0x5D0	
	モンゴメリパラメータR2 mod n	-	RAM@0x4C8	
OUT	結果：曲線上の点P	– 0xD60D：曲線上の点 – 0xA3B7：曲線上にない点 – 0xF946：xまたはy座標が係数pより小さくない	RAM@0x680	64bit

53.4.15 ECC Fpスカラー乗算

この演算は、a k x P (x_P, y_P) の計算式となります。ここで、Pは素体上の曲線の点であり、「x」は楕円曲線のスカラーと点の乗算です。計算結果は、同じ曲線に属する点または無限遠点となります。

ECC Fpスカラー乗算に関する演算方法の概要を表505に示しています。

注： **この演算が開始されると、PKA制御レジスタとPKAメモリは使用できなくなります。PKAによってBUSYビットが0にセットされると、アクセスは回復します。**

この演算が予期しないハードウェアイベントによるエラーで完了すると、TAMPペリフェラルに対してPKA改ざんイベントがトリガされ、ハードウェアによって消去されるまでPKA RAMへのアクセスがブロックされます。PKA改ざんは、プログラムされた入力ポイントが入力したECC曲線上に見つからない場合にもトリガされます。これは、PKAの演算「楕円曲線上の点」を使用することで回避できます。

表505.ECC Fpスカラー乗算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x20	PKA_CR	6bit
IN	曲線の素数位数 *n* の長さ	（ビット単位、null以外）	RAM@0x400	64bit
	曲線の係数pの長さ	（ビット単位、null以外、8 < 値 < 640）	RAM@0x408	
	曲線係数aの符号	– 0x0：正 – 0x1：負	RAM@0x410	
IN	曲線係数 |*a*|	（絶対値、|a| < p）	RAM@0x418	EOS
	曲線係数b	（正の整数）	RAM@0x520	
	曲線の係数の値 *p*	（奇数の整数素数、$0 < p < 2^{640}$）	RAM@0x1088	
	スカラー倍数k	（$0 \le k < 2^{640}$）	RAM@0x12A0	
	点Pの座標x_P	（x < *p*）	RAM@0x578	
	点Pの座標y_P	（y < *p*）	RAM@0x470	
	曲線の素数位数 *n*	（整数素数）	RAM@0xF88	
OUT	結果：k x Pの座標 *x'*	（結果 < *p*）	RAM@0x578	
	結果：k x Pの座標 *y'*	（結果 < *p*）	RAM@0x5D0	
ERROR	k x Pのエラー	– エラーなし：0xD60D – エラーあり：0xCBC9	RAM@0x680	64bit

この演算を実行する場合、次の特殊なケースに注意する必要があります。

- k = 0のとき、曲線パラメータbが0以外の場合、この関数は（0, 0）の無限遠点を返し、そうでない場合は（0, 1）を返します。kが0以外のとき、無限遠点が返される場合があります。アプリケーションがこの動作を検出したとき、新しい計算を実行する必要があります。
- k < 0のとき（負のスカラー乗算が必要とされる場合）、倍数の絶対値k = |-k| をPKAに渡す必要があります。計算完了後、-P = (x, -y) の式を使用して、有効な最終結果のy座標を計算できます（x座標は同じままです）。

注： **エラー出力が0xD60Dと異なる場合、PKAによってすべてのメモリの内容がクリアされ、秘密鍵に関する情報の漏えいが回避されます。**

53.4.16 ECDSA署名

ECDSA署名演算（セクション53.3.5で説明）の概要は 表506（入力パラメータ）と表507（出力パラメータ）に示します。

アプリケーションで出力エラーが0xD60Dかどうかを確認し、異なっている場合は、新しいkを生成して、ECDSA署名演算を繰り返す必要があります。

注： **この演算が開始されると、PKA 制御レジスタと PKA メモリは使用できなくなります。PKA によって BUSYビットが0にセットされると、アクセスは回復します。**

この演算が予期しないハードウェアイベントによるエラーで完了すると、TAMPペリフェラルに対してPKA改ざんイベントがトリガされ、ハードウェアによって消去されるまでPKA RAMへのアクセスがブロックされます。PKA改ざんは、プログラムされた入力ポイントが入力したECC曲線上に見つからない場合にもトリガされます。これは、PKAの演算「楕円曲線上の点」を使用することで回避できます。

表506.ECDSA署名 - 入力

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x24	PKA_CR	6bit
	曲線の素数位数 n の長さ（*nlen*）	（ビット単位、null以外）	RAM@0x400	64bit
	曲線の係数pの長さ	（ビット単位、8 < 値 < 640）	RAM@0x408	
	曲線係数aの符号	0x0：正 0x1：負	RAM@0x410	
	曲線係数 |a|	（絶対値、|a| < p）	RAM@0x418	EOS
	曲線係数b	（正の整数）	RAM@0x520	
	曲線の係数の値 *p*	（奇数の整数素数、$0 < p < 2^{640}$）	RAM@0x1088	
	整数k(1)	（$0 \leq k < 2^{640}$）	RAM@0x12A0	
	曲線の基点Gの座標x	（x < *p*）	RAM@0x578	
	曲線の基点Gの座標y	（y < *p*）	RAM@0x470	
	メッセージのハッシュ z	（*nlen* に等しいハッシュサイズ）(2)	RAM@0xFE8	
	秘密鍵d	（0 < d）	RAM@0xF28	
	曲線の素数位数n	（整数素数）	RAM@0xF88	

1. 通常、この整数は暗号として安全な乱数ですが、場合によっては決定論的に生成されることがあります。
2. ハッシュパラメータのサイズを曲線の素数位数nの長さと等しくするには、ゼロによるパディングまたはハッシュの切捨てを使用する必要があります。

表507.ECDSA署名 - 出力

方向とパラメータ		値（注）	ストレージ	サイズ
OUT	署名部分 *r*	（0 < *r* < n）	RAM@0x730	EOS
	署名部分 *s*	（0 < *s* < n）	RAM@0x788	
ERROR	署名の結果	– 0xD60D：計算成功、エラーなし – 0xCBC9：計算失敗 – 0xA3B7：署名部分rが0 – 0xF946：署名部分sが0	RAM@0xFE0	64bit

注： **エラー出力が0xD60Dまたは0xCBC9に等しい場合、PKAによってすべてのメモリの内容がクリアされ、秘密鍵に関する情報の漏えいが回避されます。エラー出力が0xA3B7または0xF946に等しい場合、PKAのメモリの内容は部分的に消去され、エラーコードは読取り可能に保たれます。**

拡張ECDSAサポート

PKAは、拡張ECDSA署名もサポートしており、入力および出力はECDSA署名と同じ（それぞれ 表506および表507）ですが、点kGの座標が追加されます。この追加出力は表508で定義されています。

表508.拡張ECDSA署名 - 追加出力

方向とパラメータ		値（注）	ストレージ	サイズ
OUT	曲線の点kGの座標x_1	（$0 \le x_1 < p$）	RAM@0x1400	EOS
	曲線の点kGの座標y_1	（$0 \le y_1 < p$）	RAM@0x1458	

53.4.17 ECDSA検証

ECDSA検証演算（セクション53.3.5で説明）の概要は 表509（入力パラメータ）と表510（出力パラメータ）に示します。

アプリケーションで出力エラーが0xD60Dかどうかを確認する必要があり、異なっていた場合は、署名は検証されていません。

表509.ECDSA検証 - 入力

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x26	PKA_CR	6bit
	曲線の素数位数 n の長さ（*nlen*）	（ビット単位、null以外）	RAM@0x408	64bit
	曲線の係数pの長さ	（ビット単位、null以外、 8 < 値 < 640）	RAM@0x4C8	
	曲線係数aの符号	0x0：正 0x1：負	RAM@0x468	
	曲線係数 |a|	（絶対値、|a| < p）	RAM@0x470	EOS
	曲線の係数の値 *p*	（奇数の整数素数、$0 < p < 2^{640}$）	RAM@0x4D0	
	曲線の基点Gの座標x	（x < *p*）	RAM@0x678	
	曲線の基点Gの座標y	（y < *p*）	RAM@0x6D0	
	公開鍵曲線の点Qの座標x_Q	（$x_Q < p$）	RAM@0x12F8	
	公開鍵曲線の点Qの座標y_Q	（$y_Q < p$）	RAM@0x1350	
	署名部分 *r*	（0 < *r* < n）	RAM@0x10E0	
	署名部分 *s*	（0 < *s* < n）	RAM@0xC68	
	メッセージのハッシュ *z*	（*nlen* に等しいハッシュサイズ）(1)	RAM@0x13A8	
	曲線の素数位数 *n*	（整数素数）	RAM@0x1088	

1. ハッシュパラメータのサイズを曲線の素数位数nの長さと等しくするには、ゼロによるパディングまたはハッシュの切捨てを使用する必要があります。

表510.ECDSA検証 - 出力

方向とパラメータ		値（注）	ストレージ	サイズ
OUT	結果：ECDSA検証	– 0xD60D：有効な署名 – 0xA3B7：無効な署名	RAM@0x5D0	64bit
	計算された署名部分 *r*	–（0 < *r* < n）	RAM@0x578	EOS

53.4.18 ECC完全加算

ECC完全加算では、楕円曲線上の2つの指定された点の加算を計算します。

演算方法の概要を表511に示しています。

注： **2つの入力点と結果の点は、ヤコビアン座標（X, Y, Z）で表されます。アフィン座標（x, y）で点を入力するには、(X, Y, Z) = (x, y, 1) 変換を使用することができます。結果の点をヤコビアン座標に変換するには、(x, y) = (X/Z^2, Y/Z^3) 変換を使用することができます。**

表511.ECC完全加算

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x23	PKA_CR	6bit
	曲線の係数 *p* の長さ	（ビット単位、null以外、 8 < 値 < 640）	RAM@0x408	64bit
	曲線係数 *a* の符号	– 0x0：正 0x1：負	RAM@0x410	
	曲線の係数の値 *p*	（奇数の整数素数、 $0 < p < 2^{640}$）	RAM@0x470	EOS
	曲線係数 |*a*|	（絶対値、|a| < p）	RAM@0x418	
	最初の点Pの座標X	（x < *p*）	RAM@0x628	
	最初の点Pの座標Y	（y < *p*）	RAM@0x680	
	最初の点Pの座標Z	（z < *p*）	RAM@0x6D8	
	2番目の点Qの座標X	（x < *p*）	RAM@0x730	
	2番目の点Qの座標Y	（y < *p*）	RAM@0x788	
	2番目の点Qの座標Z	（z < *p*）	RAM@0x7E0	
OUT	結果の座標X	（x < *p*）	RAM@0xD60	
	結果の座標Y	（y < *p*）	RAM@0xDB8	
	結果の座標Z	（z < *p*）	RAM@0xE10	

53.4.19 ECCダブルベースラダー

ECCダブルベースラダー演算は、k*P+m*Qの計算で構成されます。ここで、（P, Q）は楕円曲線上の2点であり、（k, m）は2つのスカラーです。演算方法の概要を表512に示しています。

結果の点が無限遠点である場合（エラーコード0xA3B7）、結果の座標は（0, 0）となります。

注： **2つの入力点はヤコビアン座標（X, Y, Z）で表されます。アフィン座標（x, y）で点を入力するには、(X, Y, Z) = (x, y, 1) 変換を使用することができます。結果はアフィン座標（x, y）で表されます。**

表512.ECCダブルベースラダー

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x27	PKA_CR	6bit
	曲線の素数位数 *n* の長さ	（ビット単位、null以外）	RAM@0x400	64bit
	曲線の係数pの長さ	（ビット単位、null以外、8 < 値 < 640）	RAM@0x408	
	曲線係数aの符号	– 0x0：正 – 0x1：負	RAM@0x410	EOS
	曲線係数 |*a*|	（絶対値、|a| < p）	RAM@0x418	
	曲線の係数の値 *p*	（奇数の整数素数、$0 < p < 2^{640}$）	RAM@0x470	
	整数k	（$0 < k < 2^{640}$）	RAM@0x520	
	整数m	（$0 < m < 2^{640}$）	RAM@0x578	
	最初の点Pの座標X	（x < *p*）	RAM@0x628	
	最初の点Pの座標Y	（y < *p*）	RAM@0x680	
	最初の点Pの座標Z	（z < *p*）	RAM@0x6D8	
	2番目の点Qの座標X	（x < *p*）	RAM@0x730	
	2番目の点Qの座標Y	（y < *p*）	RAM@0x788	
	2番目の点Qの座標Z	（z < *p*）	RAM@0x7E0	
OUT	結果の座標x	（x < *p*）	RAM@0x578	EOS
	結果の座標y	（y < *p*）	RAM@0x5D0	
	エラーコード	– 点は無限遠点ではない：0xD60D – 無限遠点：0xA3B7	RAM@0x520	64bit

53.4.20 ECC射影からアフィンへ

ECC射影からアフィンへの演算では、同次射影座標での点Pの表現とアフィン座標での点Pの表現の間の変換を計算します。つまり、点が（X, Y, Z）のトリプルで表される場合、アフィン座標 (x, y) = (X/Y, Y/Z) を計算します。

すべての演算は、点が属する曲線の係数pをモジュロとして行われます。結果の点が無限遠点である場合（エラーコード0xA3B7）、結果の座標は（0, 0）となります。

演算方法の概要を表513に示しています。

表513.ECC射影からアフィンへ

方向とパラメータ		値（注）	ストレージ	サイズ
IN	MODE	0x2F	PKA_CR	6bit
	曲線の係数 *p* の長さ	（ビット単位、8 < 値 < 640）	RAM@0x408	64bit
	曲線の係数の値p	（奇数の整数素数、$0 < p < 2^{640}$）	RAM@0x470	EOS
	点Pの座標X（射影）	（x < *p*）	RAM@0xD60	
	点Pの座標Y（射影）	（y < *p*）	RAM@0xDB8	
	点Pの座標Z（射影）	（z < *p*）	RAM@0xE10	
	モンゴメリパラメータR2 mod n	-	RAM@0x4C8	
OUT	点Pの座標x（アフィン）	（x < *p*）	RAM@0x578	
	点Pの座標y（アフィン）	（y < *p*）	RAM@0x5D0	
ERROR	エラーコード	– 点は無限遠点ではない：0xD60D – 無限遠点：0xA3B7	RAM@0x680	64bit

53.5 設定と処理時間の例

53.5.1 サポートされている楕円曲線

PKAは、素体上で定義されたすべての非特異楕円曲線をサポートしています。これらの曲線は、短いワイエルシュトラス方程式、$y^2 = x^3 + ax + b \pmod p$ で記述できます。

注： **2値曲線、エドワーズ曲線、およびCurve25519は、PKAではサポートされていません。**
ECC演算でサポートされる最大オペランドサイズは640bitです。

これらの楕円曲線のECC定義域パラメータを公開する場合、標準化団体では以下のようなパラメータを定義しています。

- 有限体GF(*p*) のすべての点の算術演算における係数として使用される素数整数*p*
- 以下に定義するGによって生成される群の位数である（通常は素数の）整数*n*
- 座標（*Gx*, *Gy*）で定義される曲線Gの基点
- 短いワイエルシュトラス方程式の係数である整数*a*と*b*

最後の箇条書きについて、標準化団体が*a*を負であると定義している場合、PKAは以下の2つの表現をサポートしています。

1. 有限体GF(*p*) において **p-|a|として定義されたa**、たとえば**p-3**：
曲線係数p = 0xFFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 00000000 FFFFFFFF FFFFFFF**F**

曲線係数aの符号= 0x0（正）
曲線係数a = 0xFFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 00000000 FFFFFFFF FFFFFFF**C**

2. **負として定義された*a***、たとえば**-3**：
曲線係数p = 0xFFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 00000000 FFFFFFFF FFFFFFFF
曲線係数aの符号= 0x**1**（負）
曲線係数a = 0x00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000**3**

表514に、ECC演算でPKAがサポートする曲線群をまとめます。

表514.ECC演算でサポートされる曲線群

曲線名	標準	参照	
P-192	NIST	デジタル署名標準（DSS）、NIST FIPS 186-4	
P-224			
P-256			
P-384			
P-521			
brainpoolP224r1, brainpoolP224t1	IETF	– Brainpool楕円曲線、IETF RFC 5639 – インターネット鍵交換（IKE）グループ記述レジストリのBrainpool楕円曲線、IETF RFC 6932	https://tools.ietf.org
brainpoolP256r1, brainpoolP256t1			
brainpoolP320r1, brainpoolP320t1			
brainpoolP384r1, brainpoolP384t1			
brainpoolP512r1, brainpoolP512t1			
secp192k1, secp192r1	SEC	効率的な暗号化の標準SEC 2曲線	https://www.secg.org
secp224k1, secp224r1			
secp256k1, secp256r1			
secp384r1			
secp521r1			
公開鍵暗号化アルゴリズムSM2の推奨曲線パラメータ	OSCCA	– 楕円曲線に基づいた公開鍵暗号アルゴリズム SM2、中国国家商務管理機構 OSCCAのSM2、2010年12月 – デジタル署名 - パート3離散対数ベースのメカニズム、ISO/IEC 14888-3、2018年11月	

53.5.2 計算時間

次の表に、AHBクロックサイクルで表したPKAの計算時間の概要を示します。

表515.べき剰余

指数の長さ（ビット単位）	モード	係数の長さ（ビット単位）			
		1024	2048	3072	4096
3	ノーマル	124600	491000	684000	1133200
	高速	22700	82000	178000	311000
17	ノーマル	135700	531400	772400	1288000
	高速	33800	122500	266500	465800
$2^{16}+1$	ノーマル	180000	693700	1126200	1907200
	高速	78200	284700	620400	1085000
1024	保護	9958000	-	-	-
	ノーマル	5850000	-	-	-
	高速	5748000	-	-	-
	CRT(1)	1775000	-	-	-
2048	保護	-	63886000	-	-
	ノーマル	-	42240000	-	-
	高速	-	41832000	-	-
	CRT（1）	-	11670000	-	-
3072	保護	-	-	199403000	-
	ノーマル	-	-	136830000	-
	高速	-	-	136325000	-
	CRT（1）	-	-	36886000	-
4096	保護	-	-	-	454318000
	ノーマル	-	-	-	316000000
	高速	-	-	-	315226000
	CRT（1）	-	-	-	84577000

1. CRTは中国剰余定理最適化の略です（MODEビットフィールド = 0x07）。

表516.ECCスカラー乗算(1)

係数の長さ（ビット単位）							
160	192	256	320	384	512	521	640
-	1590000	3083000	5339000	8518000	17818000	21053000	31826000

1. これらの時間は、スカラーパラメータに含まれる1の数に影響され、モンゴメリパラメータR2の計算が含まれています。

表517.ECDSA署名の平均計算時間(1)(2)

係数の長さ（ビット単位）							
160	192	256	320	384	512	521	640
-	1500000	2744000	4579000	7184000	14455000	16685000	24965000

1. これらの値は、指定された長さのランダムな係数による平均実行時間です。係数の長さや値に影響されます。
2. NIST楕円曲線の有限体を定義する係数の実行時間は、Brainpool楕円曲線に使用する係数や同じサイズのランダムな係数に対してかかる時間より短くなります。

表518.ECDSA検証の平均計算時間

係数の長さ（ビット単位）							
160	192	256	320	384	512	521	640
1011000	1495000	2938000	5014000	7979000	16804000	19254000	29582000

表519.ECCダブルベースラダーの平均計算時間

係数の長さ（ビット単位）							
160	192	256	320	384	512	521	640
967000	1419000	2768000	4784000	7547000	15854000	18257000	28257000

表520.ECC射影からアフィンへの平均計算時間

係数の長さ（ビット単位）						
160	192	256	320	384	512	640
47600	78000	148300	253000	419000	838400	1049300

表521.ECC完全加算の平均計算時間

係数の長さ（ビット単位）						
160	192	256	320	384	512	640
10000	12000	18000	26000	39000	53000	89000

表522.楕円曲線Fp上の点の確認の平均計算時間

係数の長さ（ビット単位）							
160	192	256	320	384	512	521	640
3400	4200	6100	8300	10900	17200	-	-

表523.モンゴメリパラメータの平均計算時間(1)

係数の長さ（ビット単位）							
192	256	320	512	1024	2048	3072	4096
8600	8710	11870	17000	102000	410000	506000	822000

1. 計算時間は、係数の長さと値に依存します。したがって、これらの値は、与えられた長さのランダム係数の平均実行時間です。

53.6 PKA割込み

公開鍵アクセラレータによって生成され個別にマスク可能な割込みソースは 4 つあり、次のイベントが通知されます。

1. PKAでサポートされていない演算のエラー（OPERRF）（セクション53.3.7を参照）
2. マップされていないアドレスへのアクセス（ADDRERRF）（セクション53.3.7を参照）
3. PKA演算中のPKA RAMのアクセス（RAMERRF）（セクション53.3.7を参照）
4. PKA演算終了（PROCENDF）

割込みソースは、同じグローバル割込みリクエスト信号pka_itに接続されます。

PKA制御レジスタ （PKA_CR）のマスクビットを変更することにより、上記の割込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを1にセットすることで、その割込みが有効になります。個別の割込みイベントのステータスは PKAステータスレジスタ（PKA_SR）から読み出すことができ、PKA_CLRFRレジスタでクリアされます。

表524に、使用可能な機能の概要を示します。

表524.PKA割込みリクエスト

項目（略称）	イベント	イベントフラグ	イネーブル制御ビット	クリア方法
PKA	サポートされていない演算	OPERRF	OPERRIE	OPERRFCビットのセット
	マップされていないアドレスへのアクセスエラー	ADDRERRF	ADDRERRIE	ADDRERRFCビットのセット
	PKA RAMアクセスエラー	RAMERRF	RAMERRIE	RAMERRFCビットのセット
	PKA演算終了	PROCENDF	PROCENDIE	PROCENDFCビットのセット

53.7 PKAレジスタ

53.7.1 PKA制御レジスタ （PKA_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OP ERRIE	ADDR ERRIE	RAM_ ERRIE	Res.	PROC ENDIE	Res.
										rw	rw	rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MODE[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	START	EN
		rw	rw	rw	rw	rw	rw							rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OPERRIE**：演算エラー割込みイネーブル

0：PKA_SRのOPERRFフラグがセットされても、割込みは生成されません。

1：PKA_SRのOPERRFフラグがセットされると、割込みが生成されます。

ビット20 **ADDRERRIE**：アドレスエラー割込みイネーブル

0：PKA_SRのADDRERRFフラグがセットされても、割込みは生成されません。

1：PKA_SRのADDRERRFフラグがセットされると、割込みが生成されます。

ビット19 **RAMERRIE**：RAMエラー割込みイネーブル

0：PKA_SRのRAMERRFフラグがセットされても、割込みは生成されません。

1：PKA_SRのRAMERRFフラグがセットされると、割込みが生成されます。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17 **PROCENDIE**：演算終了割込みイネーブル

0：PKA_SRのPROCENDFフラグがセットされても、割込みは生成されません。

1：PKA_SRのPROCENDFフラグがセットされると、割込みが生成されます。

ビット16:14 予約済みであり、リセット値に保持する必要があります。

ビット13:8 **MODE[5:0]**：PKA演算コード

000000：モンゴメリパラメータ計算後にべき剰余

000001：モンゴメリパラメータ計算のみ

000010：べき剰余のみ（モンゴメリパラメータを最初にロードする必要があります）

000011：べき剰余（保護、秘密情報を扱う場合に使用）

100000：モンゴメリパラメータ計算後にECCスカラー乗算（保護）

100100：ECDSA署名 （保護）

100110：ECDSA検証

101000：楕円曲線Fp上の点の確認

000111：RSA CRTべき乗

001000：モジュラ逆数

001001：算術加算

001010：算術減算

001011：算術乗算

001100：算術比較

001101：モジュラリダクション

001110：剰余加算

001111：剰余減算

010000：モンゴメリ乗算

100011：ECC完全加算

100111：ECCダブルベースラダー

101111：ECC射影からアフィンへ

ここにリストされていない演算が、ENビットがセットされた状態でアプリケーションによって書き込まれると、OPERRFビットがPKA_SRレジスタにセットされ、MODEビットフィールドへの書込みは無視されます。

ビット7:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **START**：演算開始

このビットに1を書き込むと、PKA RAMに既に書き込まれたオペランドとデータを使用して、MODE[5:0]で選択された演算を開始します。このビットは常に0として読み出されます。

STARTビットがセットされているときに不正な演算を選択すると、演算は開始されず、PKA_SRでOPERRFビットがセットされます。

注： PKAがビジーであると、STARTは無視されます。

ビット0 **EN**：PKAイネーブル

0：PKAは無効です。

1：PKAは有効です。PKAはハードウェアによってPKA_SRのINITOKがセットされたときに機能します。EN = 1のときに不正な演算が選択されると、PKA_SRのOPERRFビットがセットされます。詳細は、PKA_CR.MODEビットフィールドを参照してください。

注： EN = 0の場合、PKA RAMはアプリケーションでアクセスできます。

53.7.2 PKAステータスレジスタ （PKA_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OP ERRF	ADDR ERRF	RAM_ ERRF	Res.	PROC ENDF	BUSY
										r	r	r		r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INITOK
															r

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OPERRF**：演算エラーフラグ

0：イベントのエラーはありません。

1：PKA_CRレジスタで不正または不明な演算が選択されました。

このビットは、PKA_CLRFRのOPERRFCビットを使用してクリアします。

ビット20 **ADDRERRF**：アドレスエラーフラグ

0：アドレスエラーはありません。

1：アドレスアクセスが範囲外（マップされていないアドレス）です。

このビットは、PKA_CLRFRのADDRERRFCビットを使用してクリアします。

ビット19 **RAMERRF**：PKA RAMエラーフラグ

0：PKA RAMアクセスエラーはありません。

1：PKAコアが計算中に内部RAMを使用して、PKA RAMへのAHBアクセスが発生しました（AHB PKA_RAMはPKA演算中にアクセスできません）。

このビットは、PKA_CLRFRのRAMERRFCビットを使用してクリアします。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17 **PROCENDF**：PKA演算終了フラグ

0：演算が進行中です。

1：PKA演算が完了しました。このフラグは、BUSYビットがネゲートされたときにセットされます。

ビット16 **BUSY**：ビジーフラグ

このビットは、PKA演算が進行中（PKA_CRのSTART = 1）に必ずセットされます。計算が完了すると自動的にクリアされ、PKA RAMが再びアクセス可能になります。

0：進行中の演算がありません（デフォルト）。

1：演算が進行中です。

PKAが誤ったOPコードで開始された場合、数サイクルの間ビジー状態に留まり、その後、演算を自動的にアボートしてレディ状態（BUSY = 0）に戻ります。

ビット15:2 予約済みであり、リセット値に保持する必要があります。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **INITOK**：PKA初期化OK

このビットは、PKAの初期化が完了するとアサートされます。RNGが適切な乱数を出力できなかった場合、INITOKは0のままとなります。

0：PKAは正しく初期化されません。STARTビットはセットできません。

1：PKAは正しく初期化され、正常に使用できます。

53.7.3 PKAクリアフラグレジスタ（PKA_CLRFR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OP ERRFC	ADDR ERRFC	RAM_ ERRFC	Res.	PROC ENDFC	Res.
										w	w	w		w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **OPERRFC**：演算エラーフラグのクリア
0：影響なし。
1：PKA_SRのOPERRFフラグをクリアします。

ビット20 **ADDRERRFC**：アドレスエラーフラグのクリア
0：影響なし。
1：PKA_SRのADDRERRFフラグをクリアします。

ビット19 **RAMERRFC**：PKA RAMエラーフラグのクリア
0：影響なし。
1：PKA_SRのRAMERRFフラグをクリアします。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17 **PROCENDFC**：PKA演算終了フラグのクリア
0：影響なし。
1：PKA_SRのPROCENDFフラグをクリアします。

ビット16:0 予約済みであり、リセット値に保持する必要があります。

注： **PKA_CLRFRを読み出すとすべて0で返されます。**

53.7.4 PKA RAM

PKA RAMは、PKAベースアドレスと比較して0x0400のオフセットアドレスでマップされています。PKA.AHBインタフェースでは、32bitワードのシングルアクセスのみがサポートされています。

RAMサイズは5336バイトです（最大ワードオフセット：0x14D0)

注： **セクション 53.3.3：PKAのリセットおよびクロックで説明したように、PKA RAMは、PKAリセットまたは製品リセットの直後には使用できません。**

53.7.5 PKAレジスタマップ

表525.PKAレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	PKA_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OPERRIE	ADDRERRIE	RAMERRIE	Res.	PROCENDIE	Res.	Res.	Res.	MODE[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	START	EN
	リセット値											0	0	0		0				0	0	0	0	0	0							0	0
0x004	PKA_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OPERRF	ADDRERRF	RAMERRF	Res.	PROCENDF	BUSY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INITOK
	リセット値											0	0	0		0	0																0
0x008	PKA_CLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OPERRFC	ADDRERRFC	RAMERRFC	Res.	PROCENDFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0		0																	

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

54 高機能制御タイマ（TIM1/TIM8）

54.1 TIM1/TIM8 の概要

高機能制御タイマ（TIM1/TIM8）は、プログラム可能なプリスケーラによって駆動される16bitの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラとRCCクロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

高機能制御タイマ（TIM1/TIM8）と汎用タイマ（TIMy）は、互いに独立しており、リソースを共有しません。これらのタイマは、セクション 54.3.30：タイマの同期に示すように、相互に同期させることができます。

54.2 TIM1/TIM8 の主な特徴

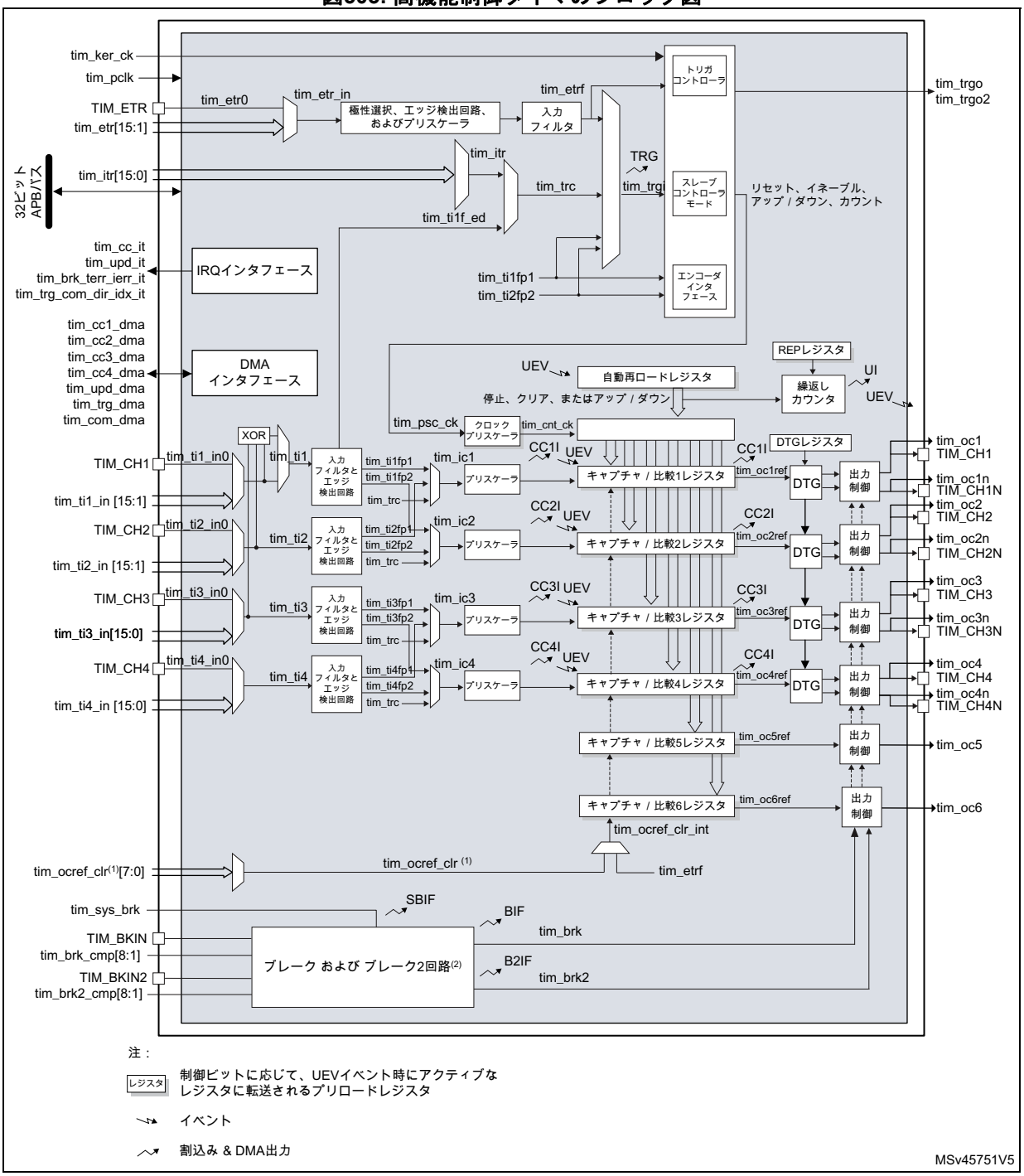
TIM1/TIM8 タイマの主な機能は、次のとおりです。

- 16bitのアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- 16bitのプログラム可能なプリスケーラ。カウンタクロック周波数を1から65536の間で分周でき、分周比の動作中の変更も可能。
- 次の機能を持つ、最大6つの独立チャネル：
  - 入力キャプチャ（ただしチャネル5および6）
  - 出力比較
  - PWM生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰返しカウンタ。
- タイマの出力信号をユーザが選択可能な安全な設定にする2つのブレーク入力。
- 以下のイベント時の割込み／DMA生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

54.3 TIM1/TIM8機能詳細

54.3.1 ブロック図

図503. 高機能制御タイマのブロック図



1. この機能は一部のタイマでは利用できません（セクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照）。
2. 詳細については、図 550：ブレークおよびブレーク2回路の概要を参照してください。

54.3.2 TIM1/TIM8ピンおよび内部信号

このセクションの以下の表は、TIMの入力および出力の一覧です。

表526.TIMの入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2 TIM_CH3 TIM_CH4	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、またはPWMで使用できます。 TIM_CH1とTIM_CH2は、外部クロック（tim_ker_ckクロックの1/4未満）、外部トリガ、および直交エンコーダの入力としても使用できます。 TIM_CH1、TIM_CH2、およびTIM_CH3は、デジタルホール効果センサとのインタフェースに使用できます。
TIM_CH1N TIM_CH2N TIM_CH3N TIM_CH4N	出力	デッドタイム挿入の可能性があるTIM_CHx出力から導き出されるタイマ相補出力。
TIM_ETR	入力	外部トリガ入力。この入力は、外部トリガまたは外部クロックソースとして使用できます。tim_etr_inプリスケーラが使用されている場合、この入力はtim_ker_ckよりも高い周波数のクロックを受信できます。
TIM_BKIN TIM_BKIN2	入力／出力	ブレークおよびブレーク2入力。これらの入力は双方向モードにも設定することができます。

表527.TIM内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0] tim_ti3_in [15:0] tim_ti4_in [15:0]	入力	内部タイマ入力バス。tim_ti1_in[15:0] および tim_ti2_in[15:0] の入力は、キャプチャ用、外部クロック（tim_ker_ckクロックの1/4未満）、直交エンコーダ信号用として使用できます。
tim_etr[15:0]	入力	。外部トリガ内部入力バス。これらの入力は、トリガ、外部クロックとして、またはハードウェアのサイクルごと（サイクルバイサイクル）のパルス幅制御に使用できます。tim_etr_inプリスケーラが使用されている場合、これらの入力はtim_ker_ckよりも高い周波数のクロックを受信できます。
tim_itr[15:0]	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ckクロックの1/4未満）として使用できます。
tim_trgo/tim_trgo2	出力	内部トリガ出力。これらのトリガは、他のタイマおよび／またはその他のペリフェラルによって使用されます。

表527.TIM内部入力／出力信号 （続き）

内部信号名	信号タイプ	説明
tim_ocref_clr[7:0]	入力	タイマのtim_ocref_clr入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim ocxref信号をクリアするために使用できます。
tim_brk_cmp[8:1]	入力	内部信号のブレーク入力
tim_brk2_cmp [8:1]	入力	内部信号のブレーク2入力
tim_sys_brk[n:0]	入力	システムブレーク入力この入力はMCUのシステムレベルのエラーを集めます。
tim_pclk	入力	タイマAPBクロック
tim_ker_ck	入力	タイマカーネルクロック
tim_cc_it	出力	タイマのキャプチャ／比較割込み
tim_upd_it	出力	タイマ更新イベント割込み
tim_brk_terr_ierr_it	出力	タイマブレーク、ブレーク2、遷移エラー、およびインデックスエラー割込み
tim_trg_com_dir_idx_it	出力	タイマトリガ、転流、方向、およびインデックス割込み
tim_cc1_dma tim_cc2_dma tim_cc3_dma tim_cc4_dma	出力	タイマキャプチャ／比較1～4 DMAリクエスト
tim_upd_dma	出力	タイマ更新DMAリクエスト
tim_trg_dma	出力	タイマトリガDMAリクエスト
tim_com_dma	出力	タイマ転流DMAリクエスト

および に、tim_ti[4:1] 入力マルチプレクサに接続されるソースを列挙します。

表528.tim_ti1入力マルチプレクサへの相互接続

tim_ti1入力	ソース	
	TIM1	TIM8
tim_ti1_in0	TIM1_CH1	TIM8_CH1
tim_ti1_in1	comp1_out	comp1_out
tim_ti1_in2	comp2_out(1)	comp2_out (1)
tim_ti1_in [15:3]	予約済み	

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表529.tim_ti2入力マルチプレクサへの相互接続

tim_ti2入力	ソース	
	TIM1	TIM8
tim_ti2_in0	TIM1_CH2	TIM8_CH2
tim_ti2_in [15:1]	予約済み	

表530.tim_ti3入力マルチプレクサへの相互接続

tim_ti3入力	ソース	
	TIM1	TIM8
tim_ti3_in0	TIM1_CH3	TIM8_CH3
tim_ti2_in [15:1]	予約済み	

表531.tim_ti4入力マルチプレクサへの相互接続

tim_ti4入力	ソース	
	TIM1	TIM8
tim_ti4_in0	TIM1_CH4	TIM8_CH4
tim_ti4_in [15:1]	予約済み	

以下の表に、tim_etr入力マルチプレクサに接続される内部ソースを列挙します。

表532. 内部トリガ接続

タイマ内部トリガ入力信号	TIM1	TIM8
tim_itr0	予約済み	tim1_trgo
tim_itr1	tim2_trgo	tim2_trgo
tim_itr2	tim3_trgo	tim3_trgo
tim_itr3	tim4_trgo	tim4_trgo
tim_itr4	tim5_trgo	tim5_trgo
tim_itr5	tim8_trgo	予約済み
tim_itr6	tim15_trgo	tim15_trgo
tim_itr7	tim16_oc1	tim16_oc1
tim_itr8	tim17_oc1	tim17_oc1
tim_itr[15:9]	予約済み	

以下の表に、tim_etr入力マルチプレクサに接続される内部ソースを列挙します。

表533.STM32U535/545/575/585向けの tim_etr入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て	
	TIM1	TIM8
tim_etr0	TIM1_ETR	TIM8_ETR
tim_etr1	comp1_out	comp1_out
tim_etr2	comp2_out(1)	comp2_out (1)
tim_etr3	MSIK	MSIK
tim_etr4	HSI	HSI
tim_etr5	MSIS	MSIS
tim_etr6	予約済み	
tim_etr7		
tim_etr8	adc1_awd1	adc1_awd1
tim_etr9	adc1_awd2	adc1_awd2
tim_etr10	adc1_awd3	adc1_awd3
tim_etr11	adc4_awd1	adc4_awd1
tim_etr12	adc4_awd2	adc4_awd2
tim_etr13	adc4_awd3	adc4_awd3
tim_etr[15:14]	予約済み	

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表534.STM2U59x/5Ax/5Fx/5Gx向けのtim_etr入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て	
	TIM1	TIM8
tim_etr0	TIM1_ETR	TIM8_ETR
tim_etr1	comp1_out	comp1_out
tim_etr2	comp2_out	comp2_out
tim_etr3	MSIK	MSIK
tim_etr4	HSI	HSI
tim_etr5	MSIS	MSIS
tim_etr6	adc2_awd2	adc2_awd2
tim_etr7	adc2_awd3	adc2_awd3
tim_etr8	adc1_awd1	adc1_awd1
tim_etr9	adc1_awd2	adc1_awd2
tim_etr10	adc1_awd3	adc1_awd3
tim_etr11	adc4_awd1	adc4_awd1
tim_etr12	adc4_awd2	adc4_awd2
tim_etr13	adc4_awd3	adc4_awd3

表534.STM2U59x/5Ax/5Fx/5Gx向けのtim_etr入力マルチプレクサへの相互接続 （続き）

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て	
	TIM1	TIM8
tim_etr14	adc2_awd1	adc2_awd1
tim_etr15	予約済み	

以下の表に、tim_brkおよびtim_brk2入力に接続されるソースを列挙します。

表535.タイマブレーク相互接続

tim_brk入力	TIM1	TIM8
TIM_BKIN	TIM1_BKINピン	TIM8_BKINピン
tim_brk_cmp1	comp1_out	comp1_out
tim_brk_cmp2	comp2_out(1)	comp2_out (1)
tim_brk_cmp3	予約済み	
tim_brk_cmp4		
tim_brk_cmp5		
tim_brk_cmp6		
tim_brk_cmp7	mdf1_break0	mdf1_break2
tim_brk_cmp8	予約済み	

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表536.タイマbreak2相互接続

tim_brk2入力	TIM1	TIM8
TIM_BKIN2	TIM1_BKIN2ピン	TIM8_BKIN2ピン
tim_brk2_cmp1	comp1_out	comp1_out
tim_brk2_cmp2	comp2_out(1)	comp2_out (1)
tim_brk2_cmp3	予約済み	
tim_brk2_cmp4		
tim_brk2_cmp5		
tim_brk2_cmp6		
tim_brk2_cmp7	mdf1_break1	mdf1_break3
tim_brk2_cmp8	予約済み	

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表537.システムブレーク相互接続

tim_sys_brk入力	TIM1/TIM8/TIM20	SYSCFG_CFGR2レジスタのイネーブルビット
tim_sys_brk0	Cortex®-M33ロックアップ	CLL
tim_sys_brk1	プログラム可能な電圧検出器（PVD）	PVDL

表537.システムブレーク相互接続 （続き）

tim_sys_brk入力	TIM1/TIM8/TIM20	SYSCFG_CFGR2レジスタのイネーブルビット
tim_sys_brk2	SRAMダブルECCエラー	SPL
tim_sys_brk3	FlashダブルECCエラー	ECCL
tim_sys_brk4	クロック·セキュリティ·システム（CSS）	なし（常に有効）

以下の表に、tim_ocref_clr入力マルチプレクサに接続される内部ソースを列挙します。

表538.ocref_clr入力マルチプレクサへの相互接続

タイマOCREFクリア信号	タイマOCREFクリア信号割り当て	
	TIM1	TIM8
tim_ocref_clr0	comp1_out	comp1_out
tim_ocref_clr1	comp2_out(1)	comp2_out（1）
tim_ocref_clr2	予約済み	
tim_ocref_clr3		
tim_ocref_clr4		
tim_ocref_clr5		
tim_ocref_clr6		
tim_ocref_clr7		

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

54.3.3 タイムベースユニット

プログラム可能な高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ16bitカウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、カウンタが動作中でも、ソフトウェアで読み書きができます。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1レジスタのUDISビットが0の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力tim_cnt_ckから供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

注： **TIMx_CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があります。**

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を1から65536の間の値で分周します。16bitレジスタ（TIMx_PSCレジスタ）を使って制御される16bitカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図504と 図505に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図504. プリスケーラ分周比が1から2に変化したときのカウンタのタイミング図

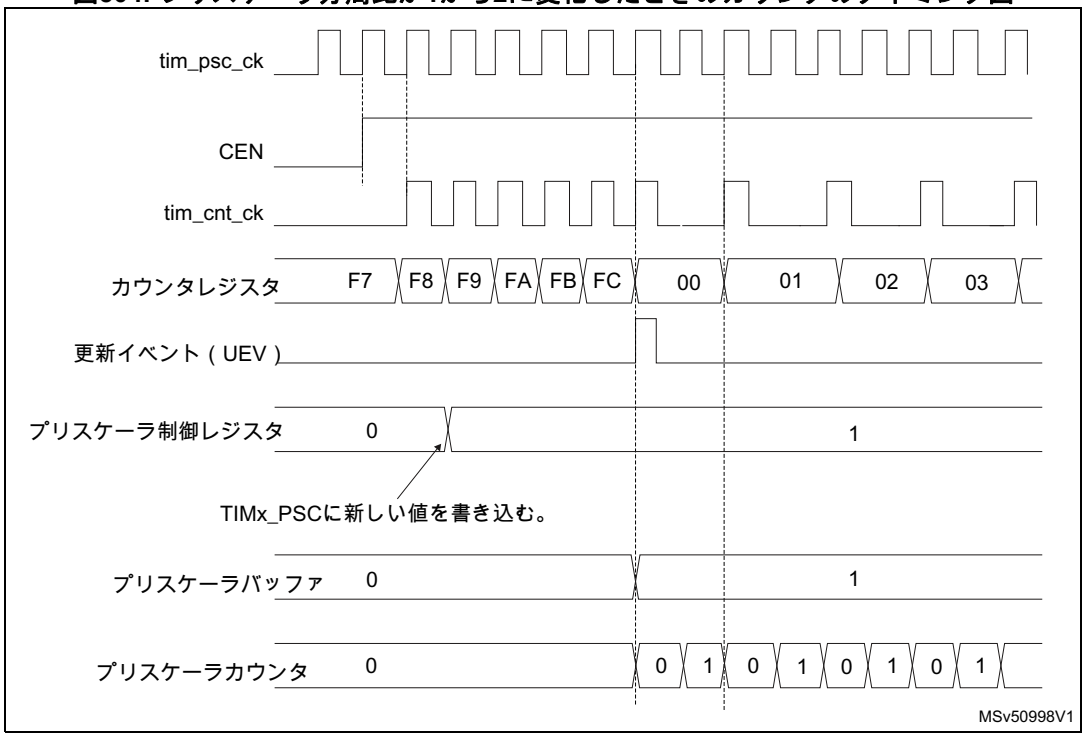
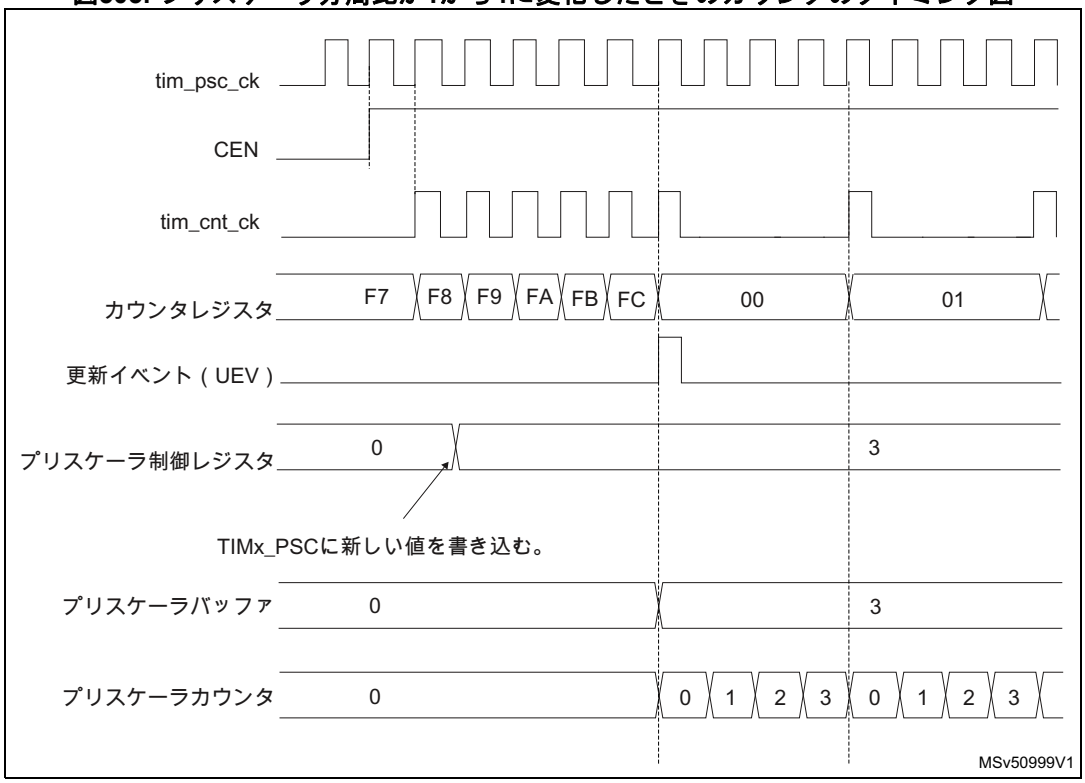


図505. プリスケーラ分周比が1から4に変化したときのカウンタのタイミング図



54.3.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは0から自動再ロード値（TIMx_ARRレジスタの内容）までカウントし、0からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰返しカウンタが使用されている場合には、繰返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることでも更新イベントが生成されます。

UEVイベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは0からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- 繰返しカウンタにはTIMx_RCRレジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図506. 内部クロック分周比が1の場合のカウンタのタイミング図

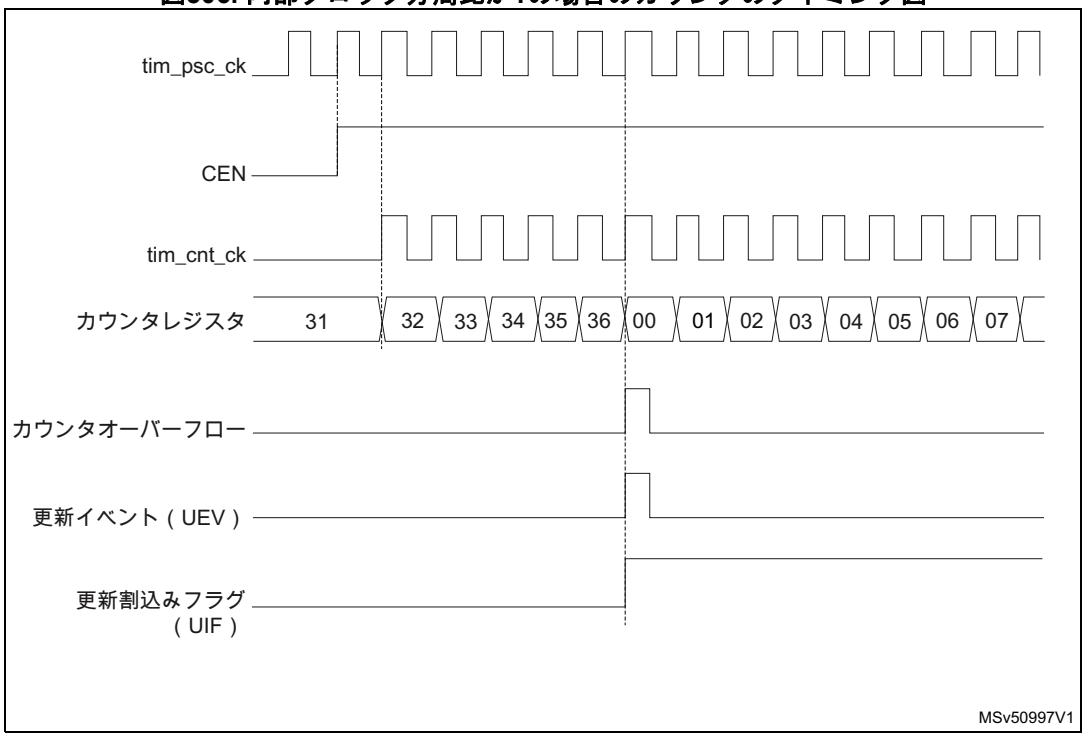


図507. 内部クロック分周比が2の場合のカウンタのタイミング図

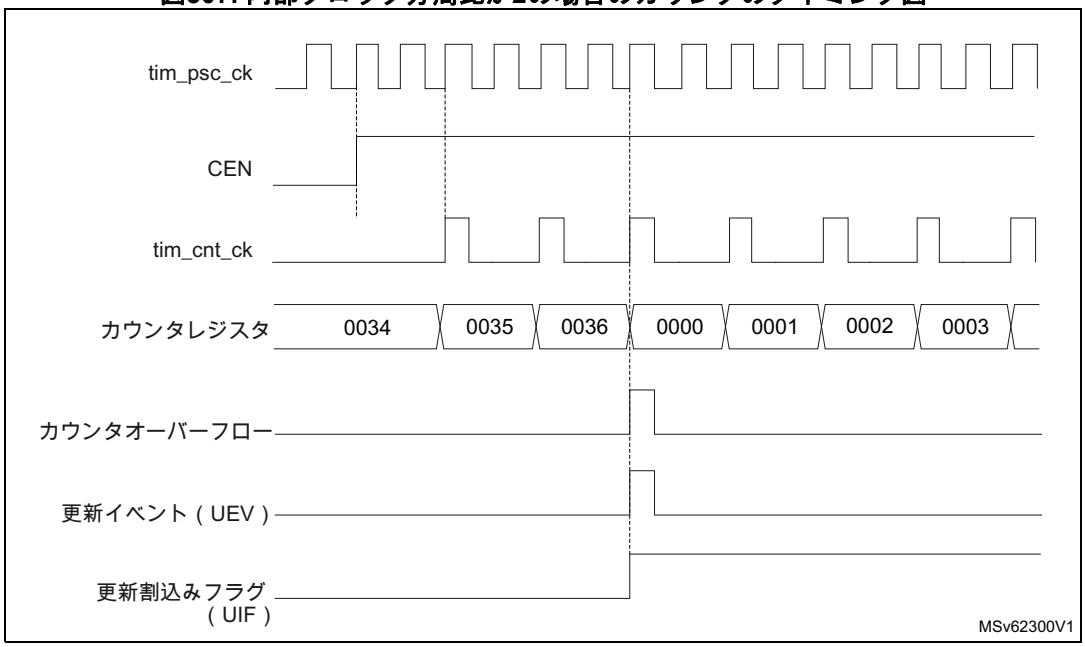


図508. 内部クロック分周比が4の場合のカウンタのタイミング図

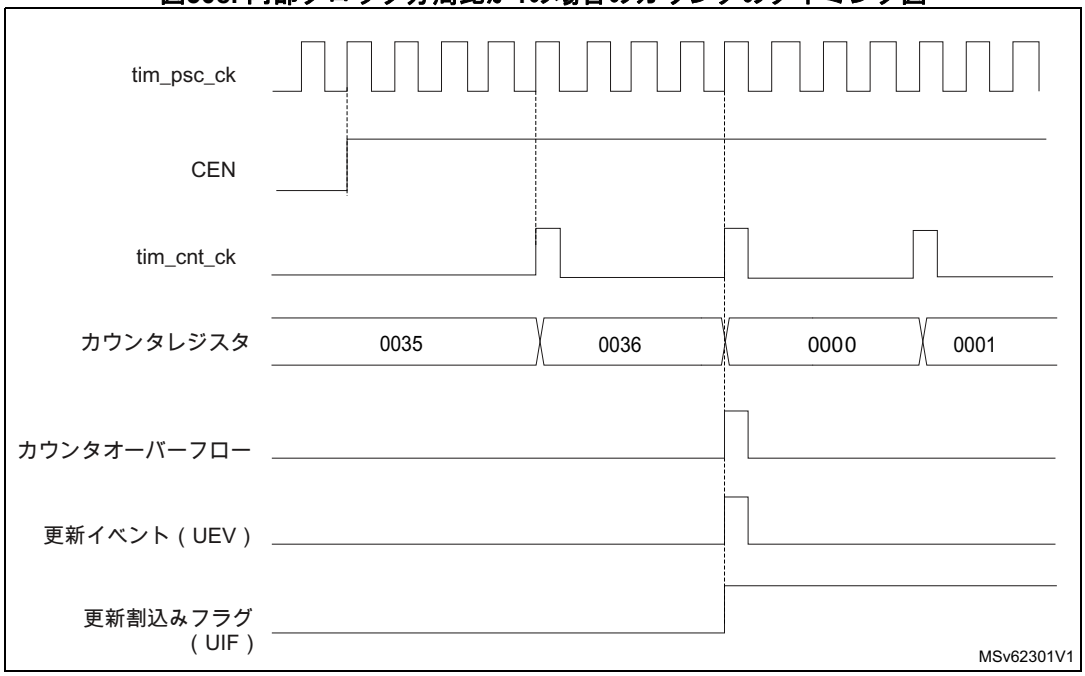


図509. 内部クロック分周比がNの場合のカウンタのタイミング図

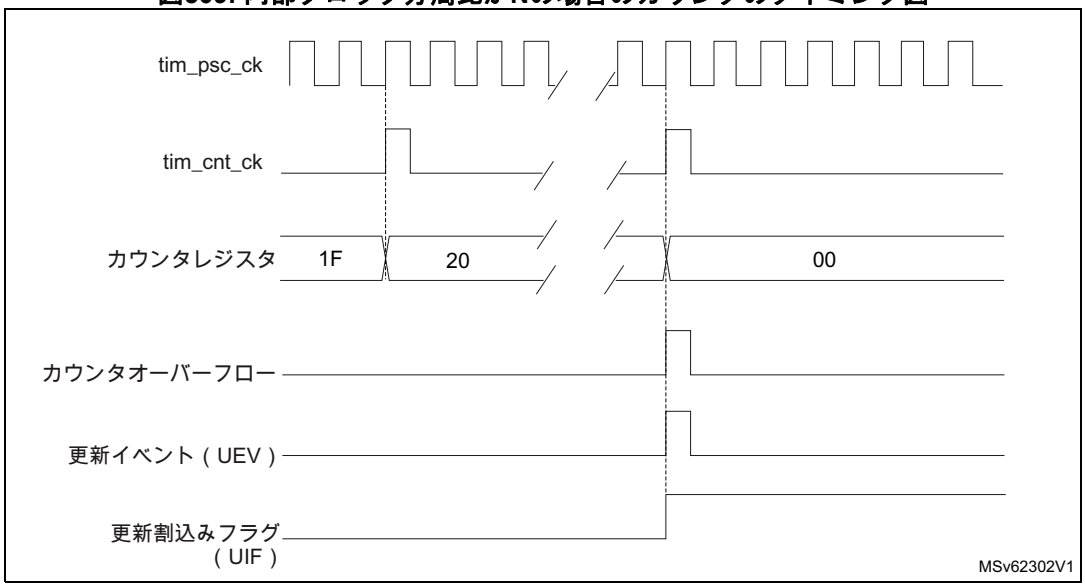


図510. ARPE=0の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARRはプリロードされない）

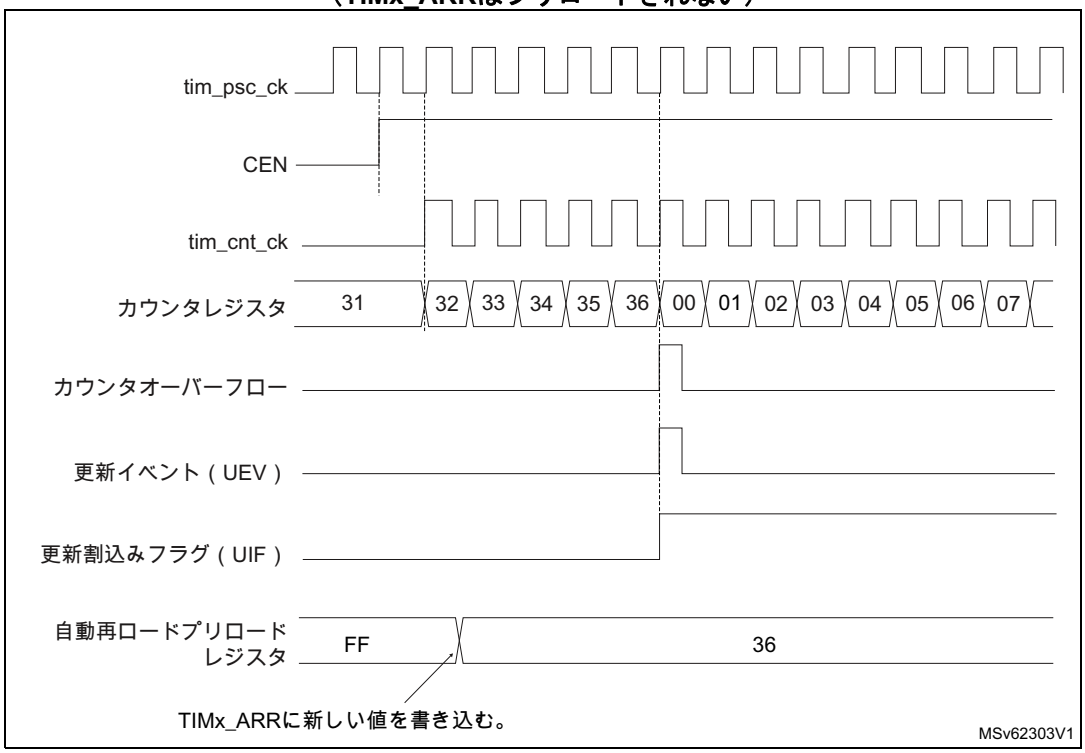
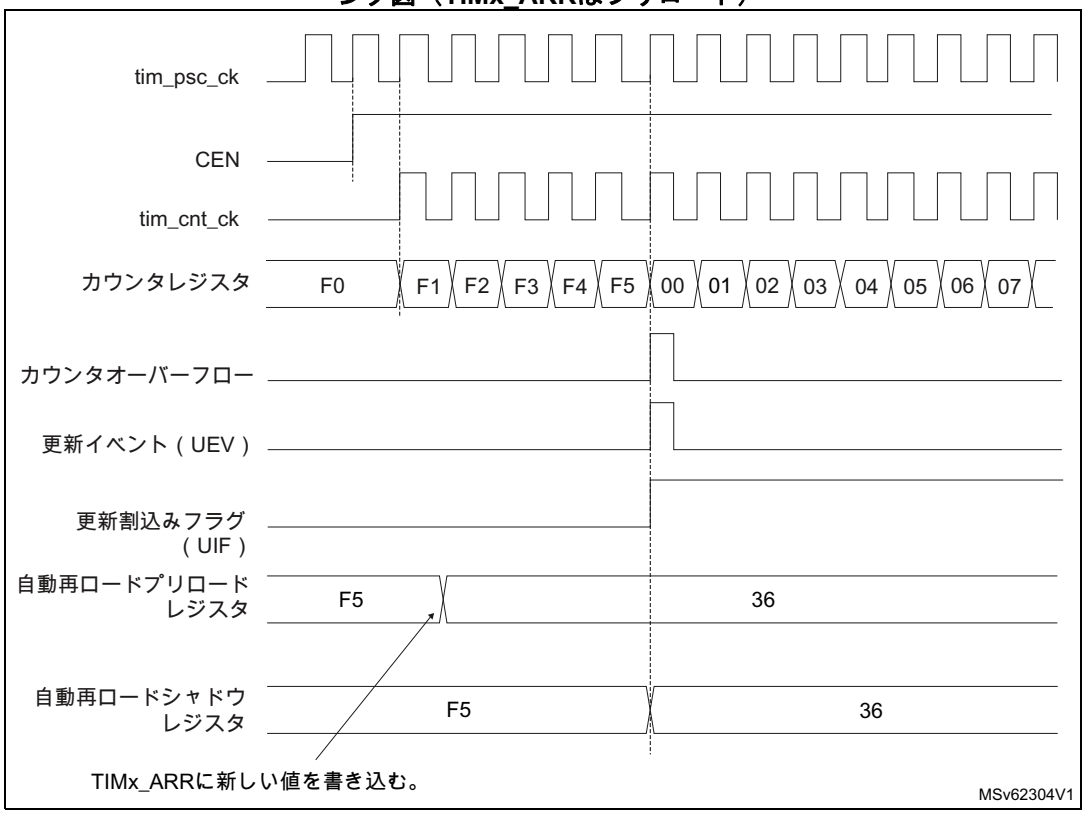


図511. ARPE=1（TIMx_ARRはプリロード）の場合の更新イベント時のカウンタのタイミング図（TIMx_ARRはプリロード）



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARRレジスタの内容）から0までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

繰返しカウンタが使用されている場合には、繰返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1までダウンカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰返しカウンタが使用されていないときには、カウンタのアンダーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることでも更新イベントが生成されます。

UEV更新イベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは0からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- 繰返しカウンタにはTIMx_RCRレジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARRレジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図512. 内部クロック分周比が1の場合のカウンタのタイミング図

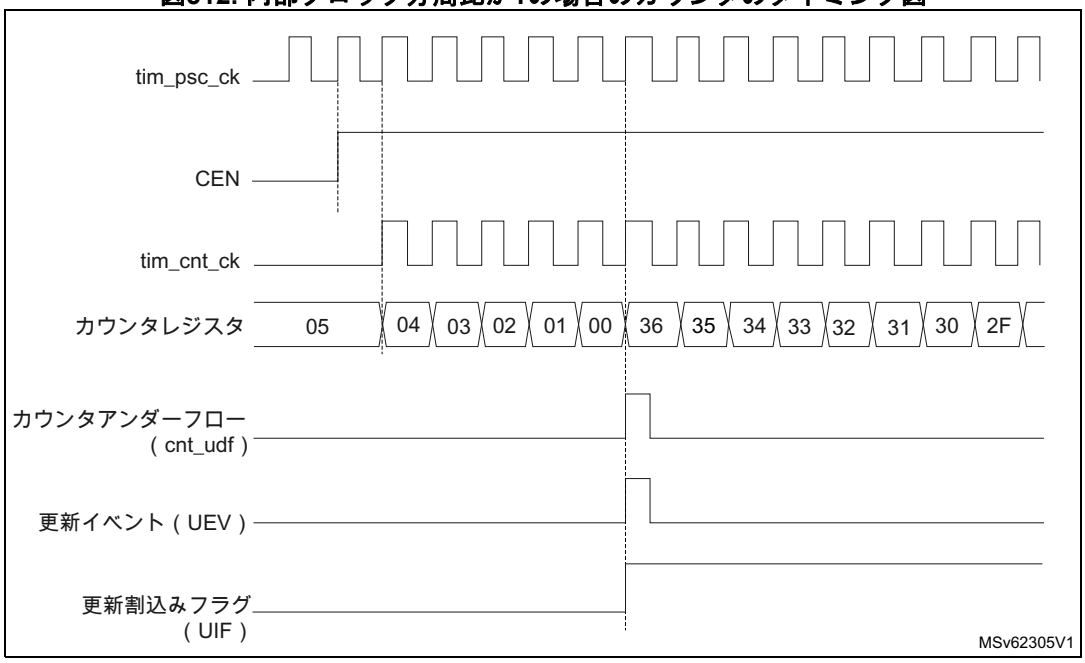


図513. 内部クロック分周比が2の場合のカウンタのタイミング図

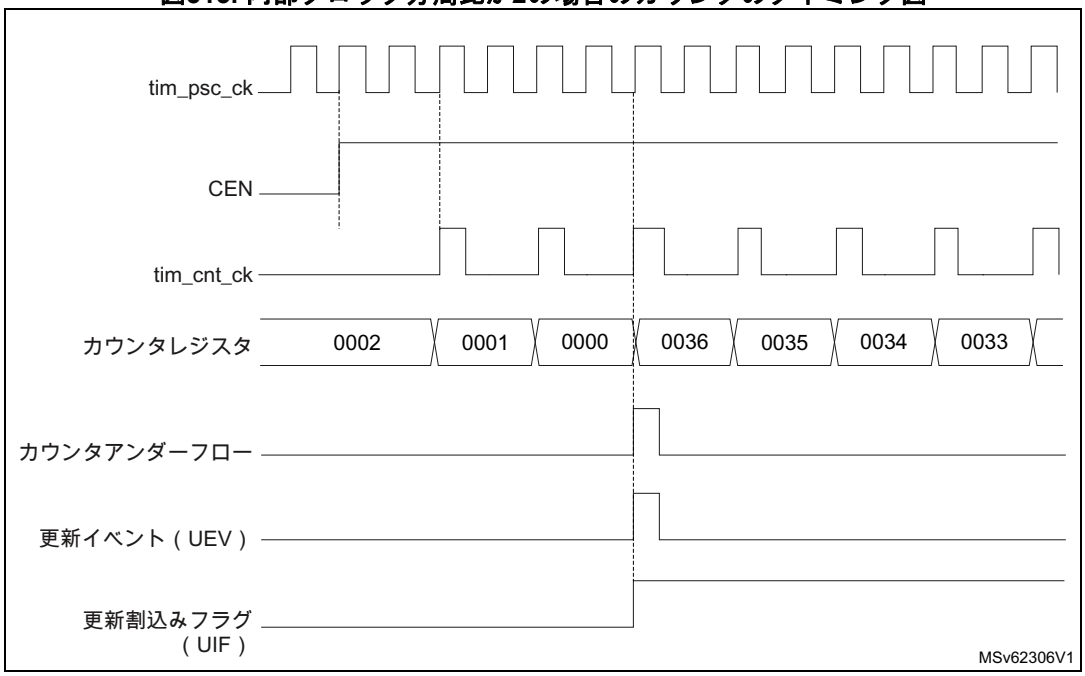


図514. 内部クロック分周比が4の場合のカウンタのタイミング図

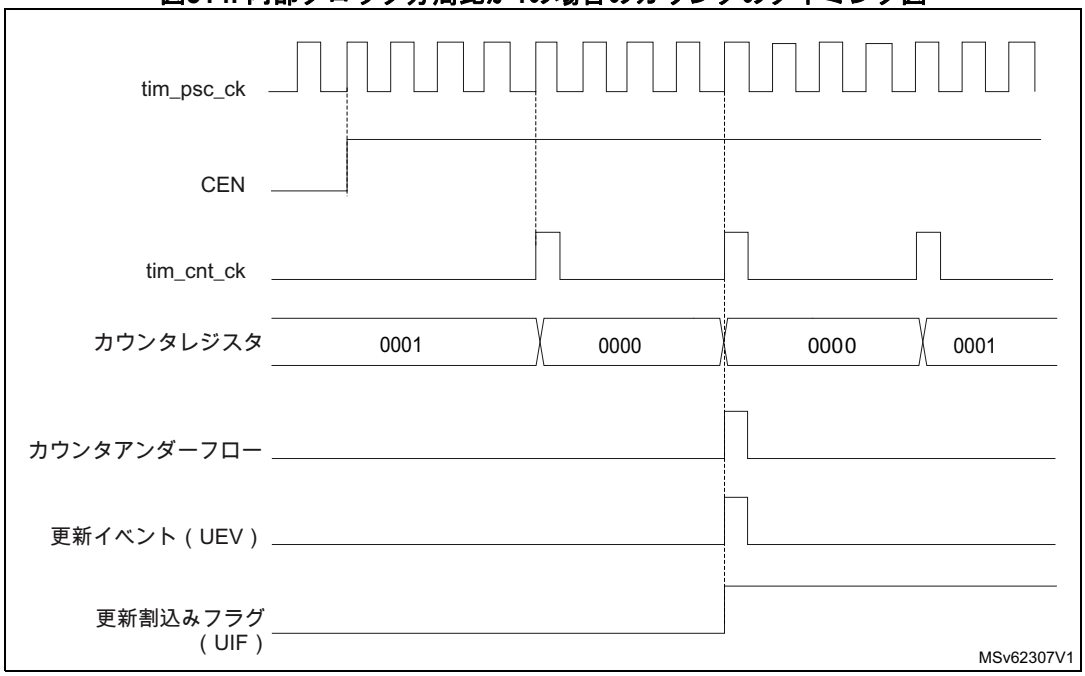


図515. 内部クロック分周比がNの場合のカウンタのタイミング図

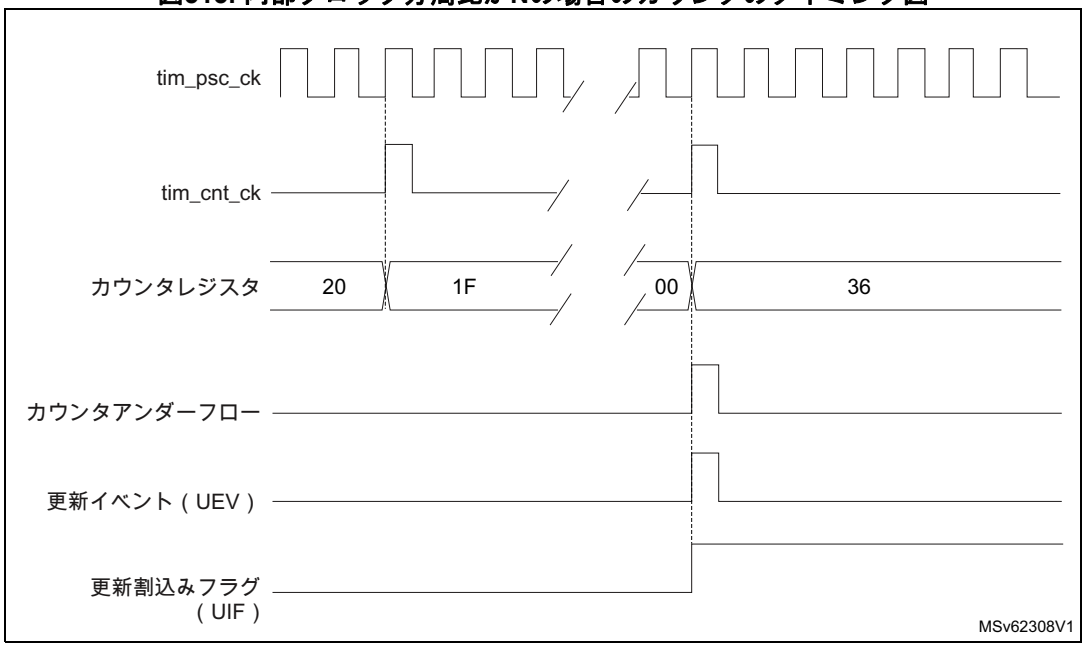
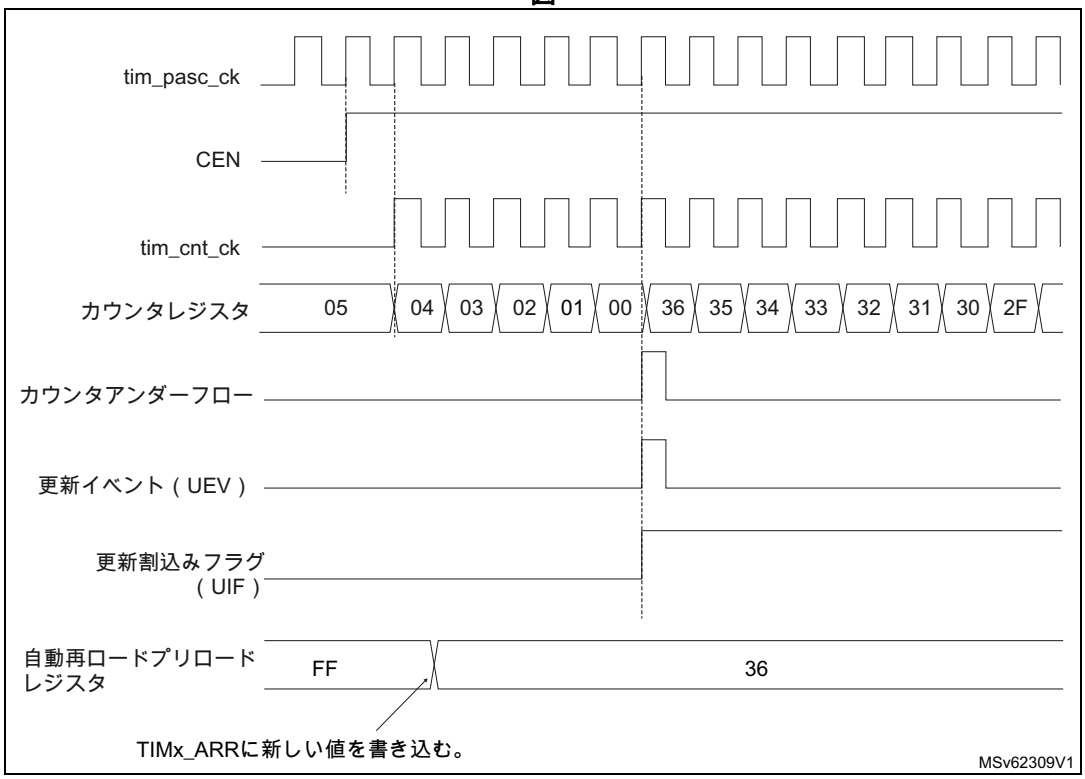


図516. 繰返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは0から自動再ロード値（TIMx_ARRレジスタの内容）－1までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から1までカウントして、カウンタアンダーフローイベントを生成します。その後、0からカウントをリスタートします。

センターアラインモードは、TIMx_CR1レジスタのCMSビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード3、CMS=11）にセットされます。

このモードでは、TIMx_CR1レジスタの方向ビット（DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは0からカウントをリスタートします。

UEV更新イベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

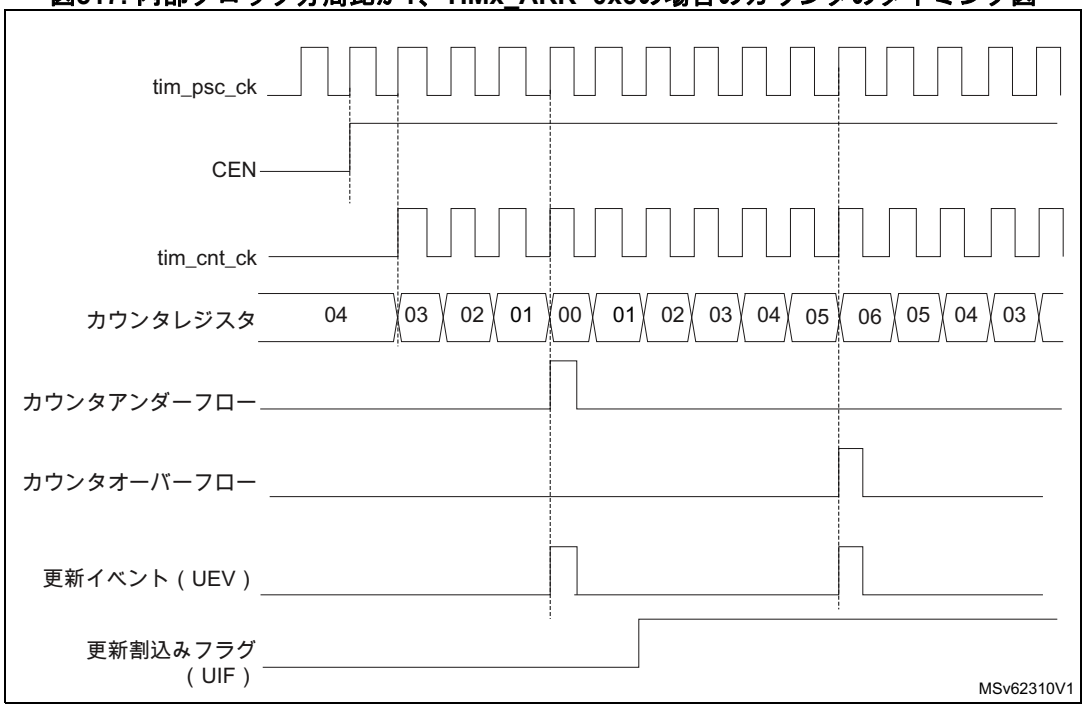
さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合、UGビットをセットするとUEV更新イベントが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みが発生するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- 繰返しカウンタにはTIMx_RCRレジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARRレジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図517. 内部クロック分周比が1、TIMx_ARR=0x6の場合のカウンタのタイミング図



1. ここでは、センターアラインモード1が使用されています（詳細については、セクション 54.6：TIM1/TIM8 レジスタを参照）。

図518. 内部クロック分周比が2の場合のカウンタのタイミング図

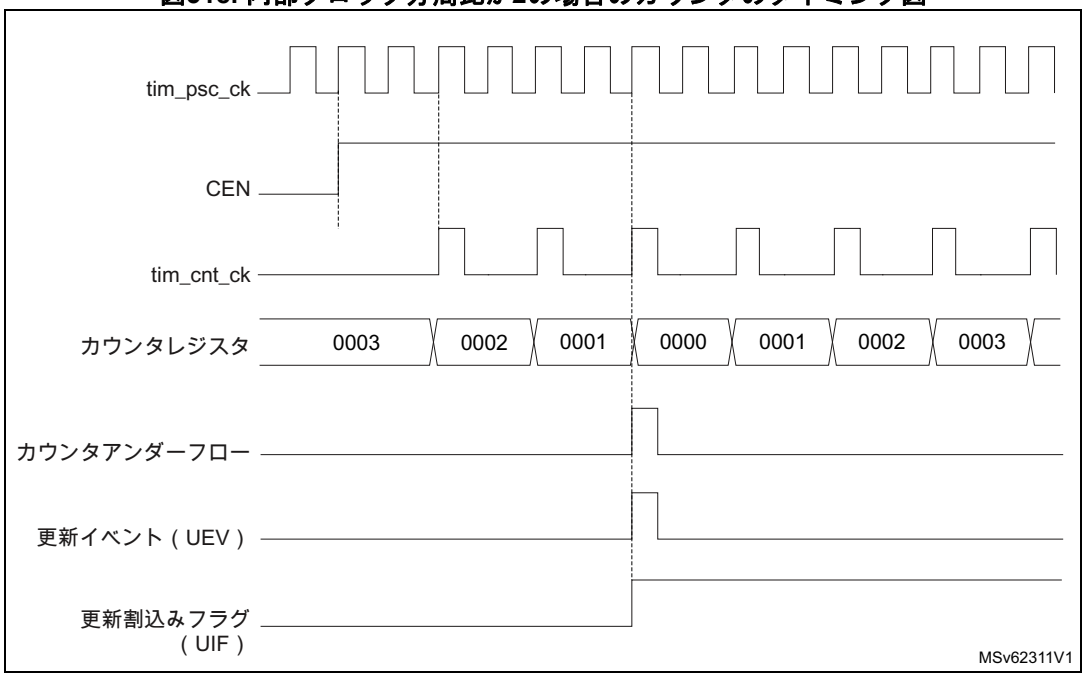


図519. 内部クロック分周比が4、TIMx_ARR=0x36の場合のカウンタのタイミング図

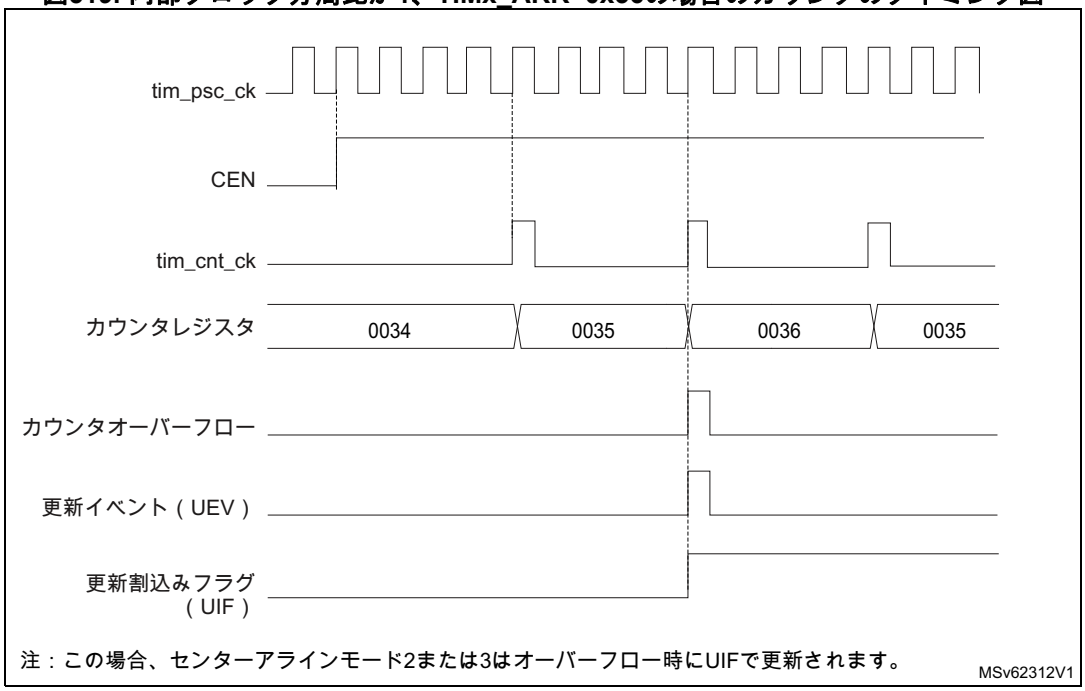


図520. 内部クロック分周比がNの場合のカウンタのタイミング図

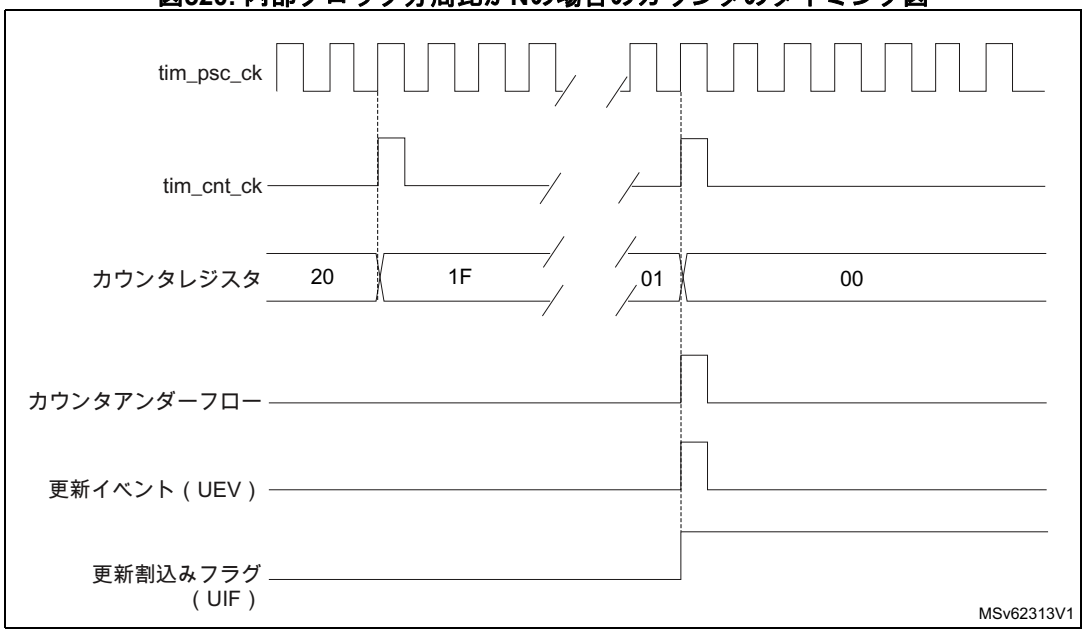


図521. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図

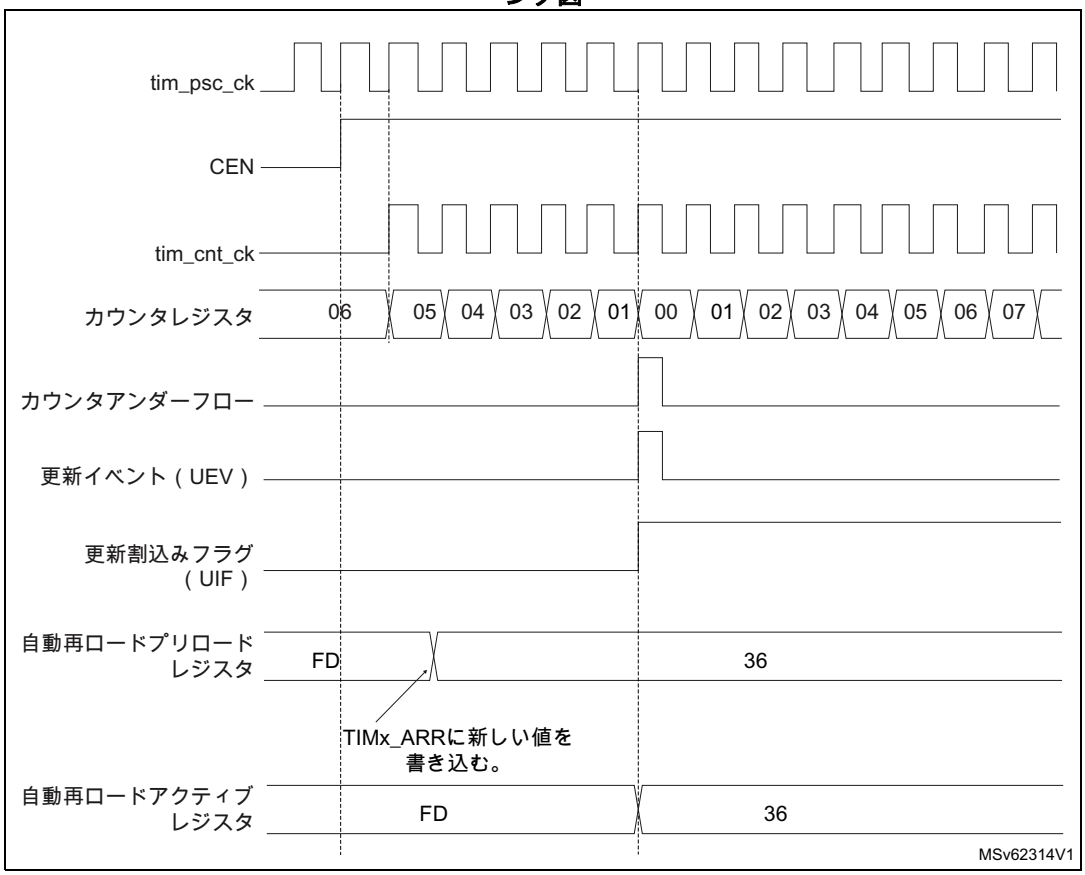
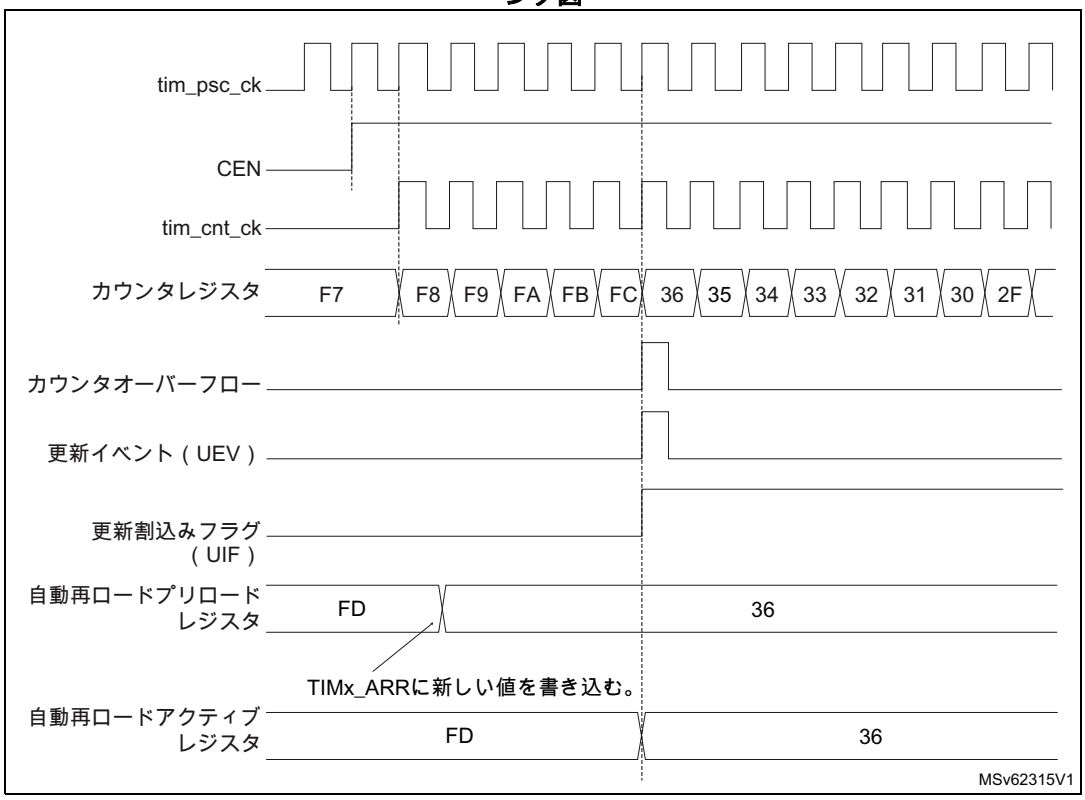


図522. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



54.3.5 繰返しカウンタ

セクション 54.3.3 : タイムベースユニットに、カウンタオーバーフロー/アンダーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰返しカウンタが0に達したときにのみ、更新イベントが生成されます。これは、PWM信号を生成する際に役立ちます。

これは、TIMx_RCR繰返しカウンタレジスタの値をNとすると、N+1回目のカウンタオーバーフローまたはアンダーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR自動再ロードレジスタ、TIMx_PSCプリスケーラレジスタ、比較モードのTIMx_CCRxキャプチャ/比較レジスタ）へデータが転送されることを意味します。

繰返しカウンタは、次の場合にデクリメントします。

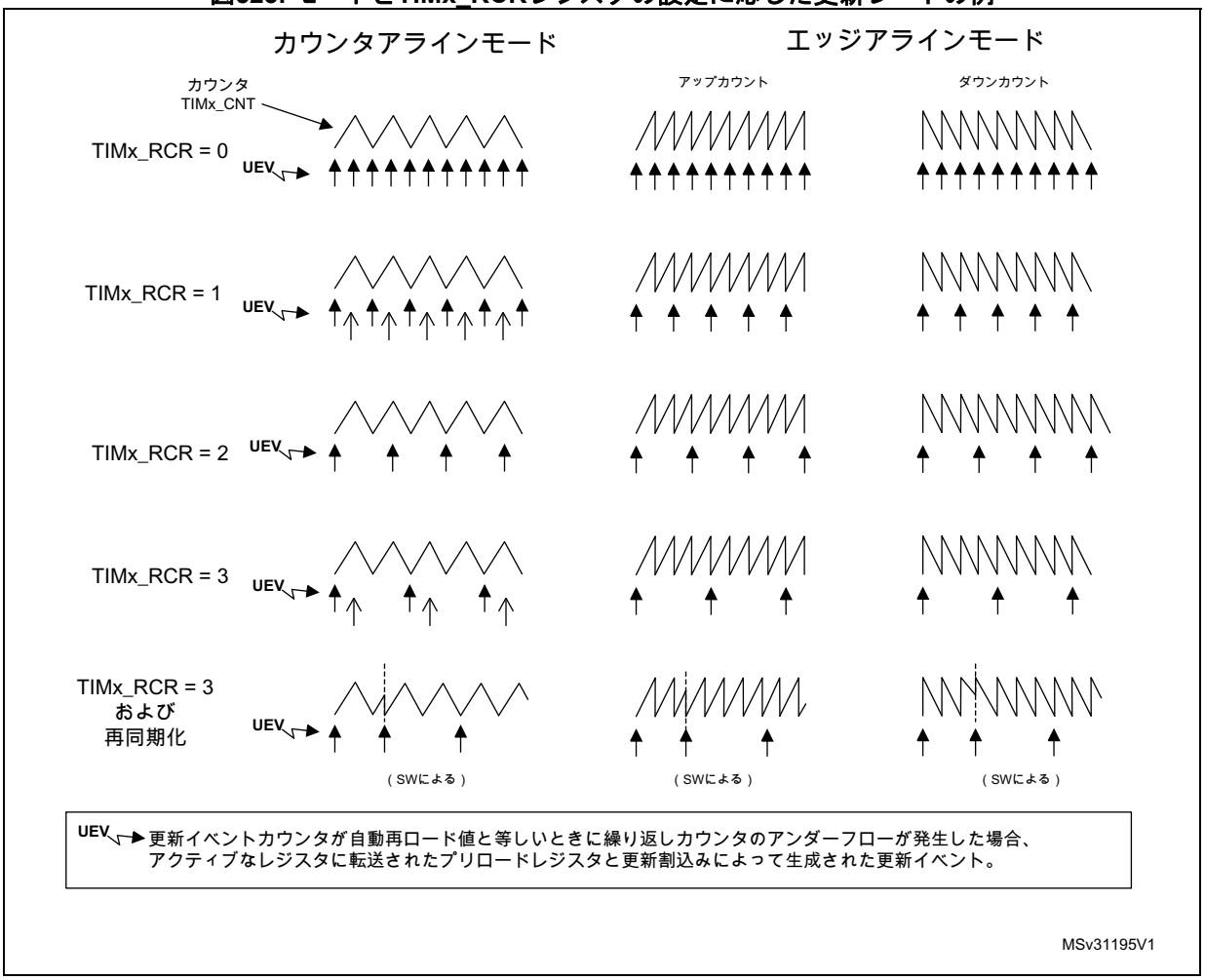
- アップカウントモードで、カウンタオーバーフローごと
- ダウンカウントモードで、カウンタアンダーフローごと
- センターアラインモードで、カウンタオーバーフローとカウンタアンダーフローごと最大繰返し回数は32768 PWMサイクルに限られますが、PWM周期ごとにデューティサイクルを2回更新することが可能になります。センターアラインモードで比較レジスタの値をPWM周期あたり1回のみ更新するときには、パターンが対称なので、最大精度は$2xT_{ck}$ です。

繰返しカウンタは自動再ロードタイプです。繰返しの回数は、TIMx_RCRレジスタの値によって定義されたとおりに維持されます（図523を参照してください）。ソフトウェアによって（TIMx_EGRレジスタのUGビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰返しカウンタの値にかかわらず直ちにイベントが発生し、繰返しカウンタにTIMx_RCRレジスタの内容が再ロードされます。

センターアラインモードでは、RCRが奇数の場合、RCRレジスタが書き込まれたタイミングおよびカウンタが開始されたタイミングに応じてオーバーフローまたはアンダーフロー時に更新イベントが発生します。カウントの開始前にRCRが書き込まれた場合は、アンダーフローで、UEVが発生します。カウントの開始後にRCRが書き込まれた場合は、オーバーフローでUEVが発生します。

たとえば、TIMx_RCR = 3の場合、TIMx_RCRの書込みタイミングに応じて4回目のオーバーフローイベントまたはアンダーフローイベントごとにUEVが発生します。

図523. モードとTIMx_RCRレジスタの設定に応じた更新レートの例



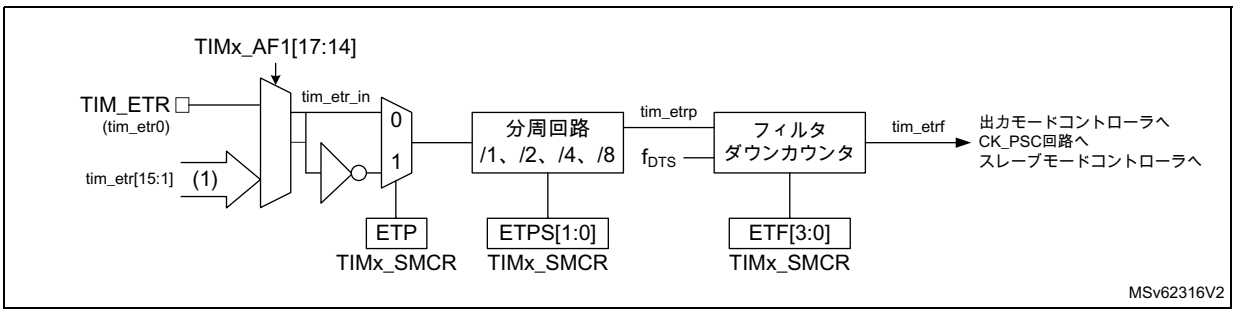
54.3.6 外部トリガ入力

タイマには外部トリガ入力tim_etr_in機能があります。以下の目的で使用できます。

- 外部クロック（外部クロックモード2、セクション54.3.7を参照）
- スレーブモードのトリガ（セクション54.3.30を参照）
- サイクルごとの電流調整のPWMリセット入力（セクション54.3.9を参照）

以下の図524は、tim_etr_in入力の前処理について説明しています。入力の極性は、TIMxSMCRレジスタのETPビットで定義されています。トリガはETPS[1:0] ビットフィールドでプログラムされた分周器でプリスケールし、ETF[3:0] ビットフィールドでデジタル的にフィルタリングすることができます。結果の信号（tim_etrf）は、次の3つの目的に使用できます。すなわち、外部クロックとして、出力を調整する（通常は電流制限のためにPWM出力をリセットする）ため、およびスレーブモードコントローラのトリガとしてです。

図524. 外部トリガ入力ブロック



tim_etr_in入力は、入力ピン（デフォルト設定）、内部ソースといった複数のソースから入力されます。この選択は、TIMx_AF1レジスタのETRSEL[3:0] ビットフィールドで行います。

製品のetr_in入力へ接続されるソースの一覧についてはセクション 54.3.2: TIM1/TIM8ピンおよび内部信号を参照してください。

54.3.7 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

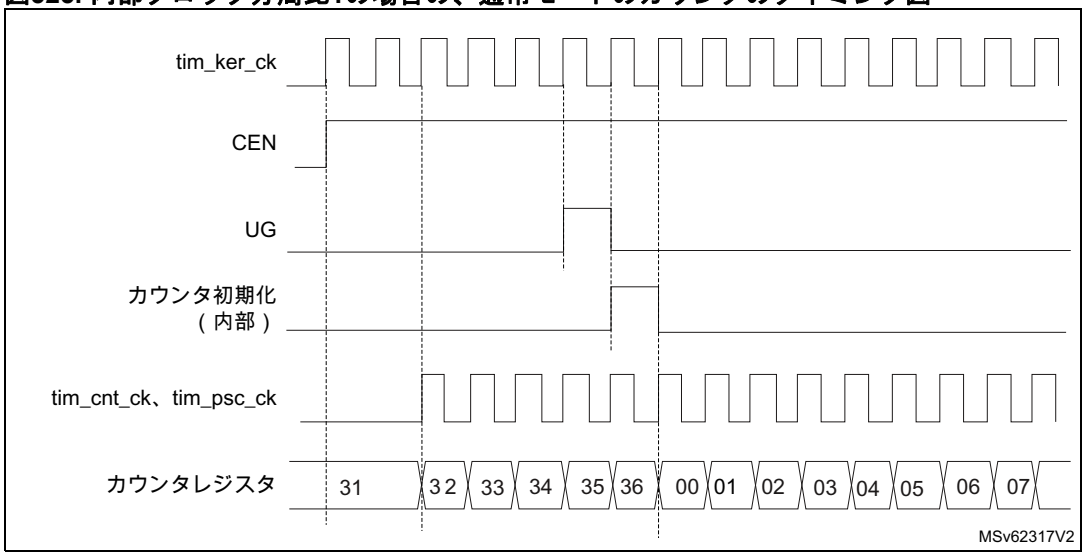
- 内部クロック（tim_ker_ck）
- 外部クロックモード1：外部入力ピン（tim_ti1またはtim_ti2）
- 外部クロックモード2：外部トリガ入力（tim_etr_in）
- エンコーダモード

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（SMS=000）、CEN、DIR（TIMx_CR1レジスタ）、およびUGビット（TIMx_EGRレジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれるUGビットを除きます）。CENビットに1が書き込まれるとすぐに、プリスケーラには内部クロックtim_ker_ckが供給されます。

図525に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

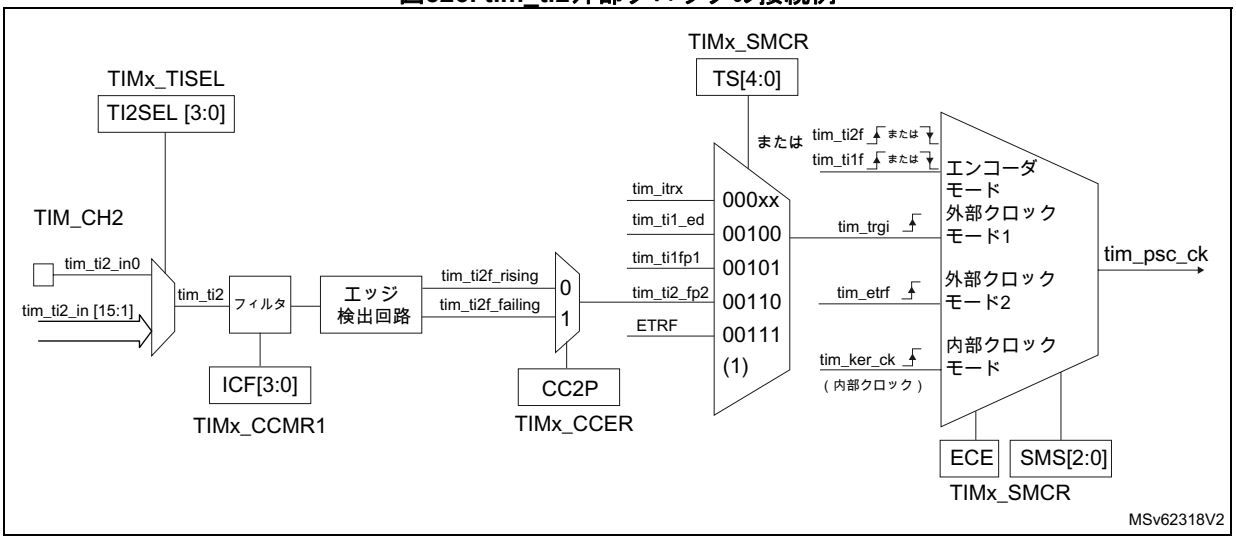
図525. 内部クロック分周比1の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード1

このモードはTIMx_SMCRレジスタのSMS=111のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図526. tim_ti2外部クロックの接続例



1. 01000から11111の範囲のコードは予約済み。

たとえば、tim_ti2入力の立上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

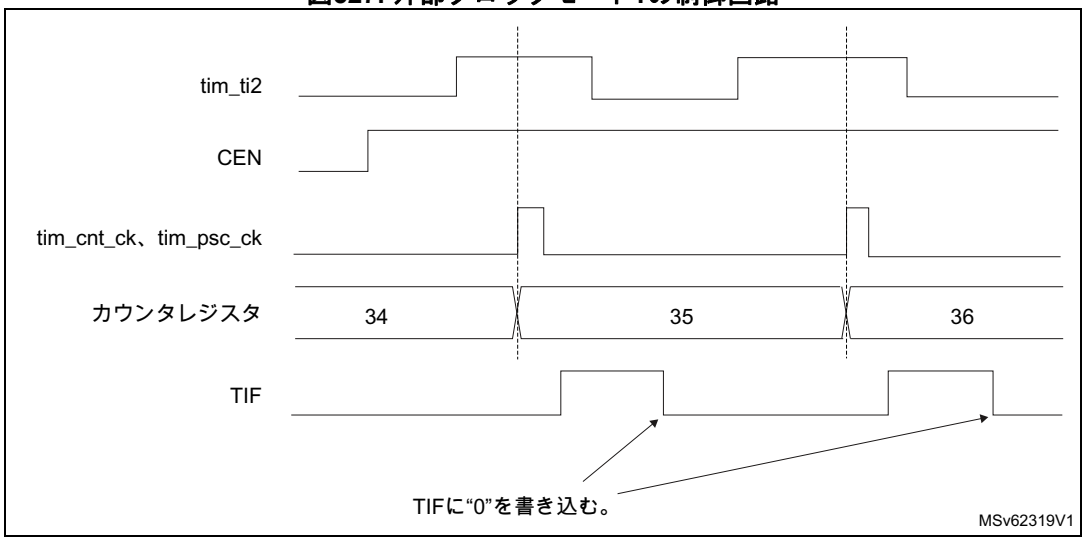
1. TIMx_CCMR1レジスタのCC2Sビットに“01”を書き込むことによって、チャネル2がtim_ti2入力の立上がりエッジを検出するように設定します。
2. TIMx_CCMR1レジスタのIC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000にしておきます）。
3. CC2P=0とCC2NP=0をTIMx_CCERレジスタに書き込んで、立上がりエッジ極性を選択します。
4. TIMx_SMCRレジスタにSMS=111を書き込むことによって、タイマを外部クロックモード1に設定します。
5. TIMx_SMCRレジスタにTS=00110を書き込むことによって、トリガ入力ソースとしてtim_ti2を選択します。*/
6. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。**

tim_ti2の立上がりエッジが発生すると、カウンタは1カウントを行い、TIFフラグがセットされます。

tim_ti2の立上がりエッジから実際のカウンタクロックまでの間には、tim_ti2入力の再同期回路による遅延があります。

図527. 外部クロックモード1の制御回路



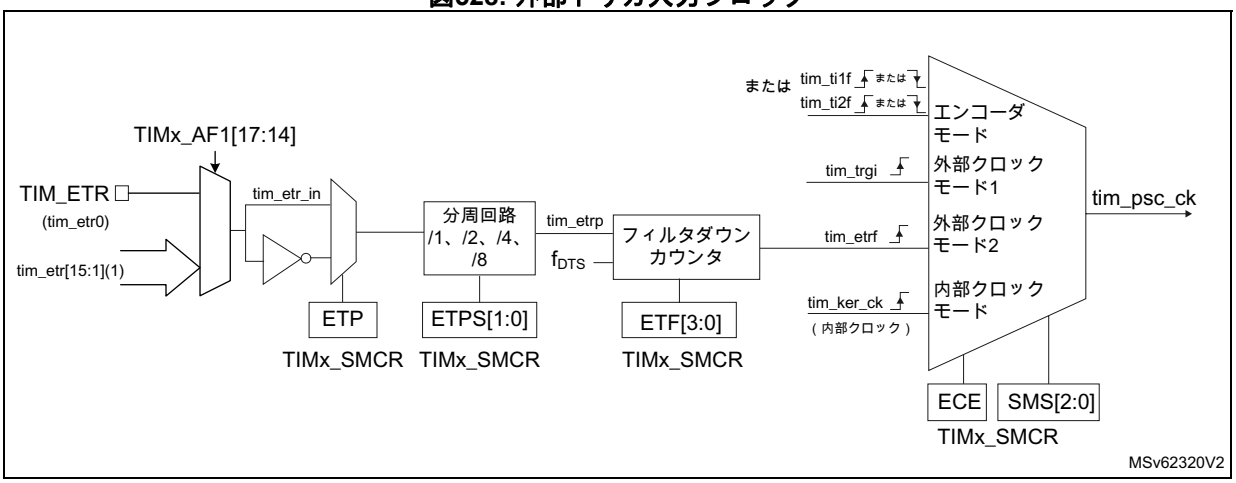
外部クロックソースモード2

このモードは、TIMx_SMCRレジスタのECE=1を書き込むことによって選択されます。

カウンタは、外部トリガ入力tim_etr_inの立上がりまたは立下がりエッジごとにカウントします。

図528に、外部トリガ入力ブロックの概要を示します。

図528. 外部トリガ入力ブロック



1. セクション 54.3.2 : TIM1/TIM8ピンおよび内部信号を参照してください。

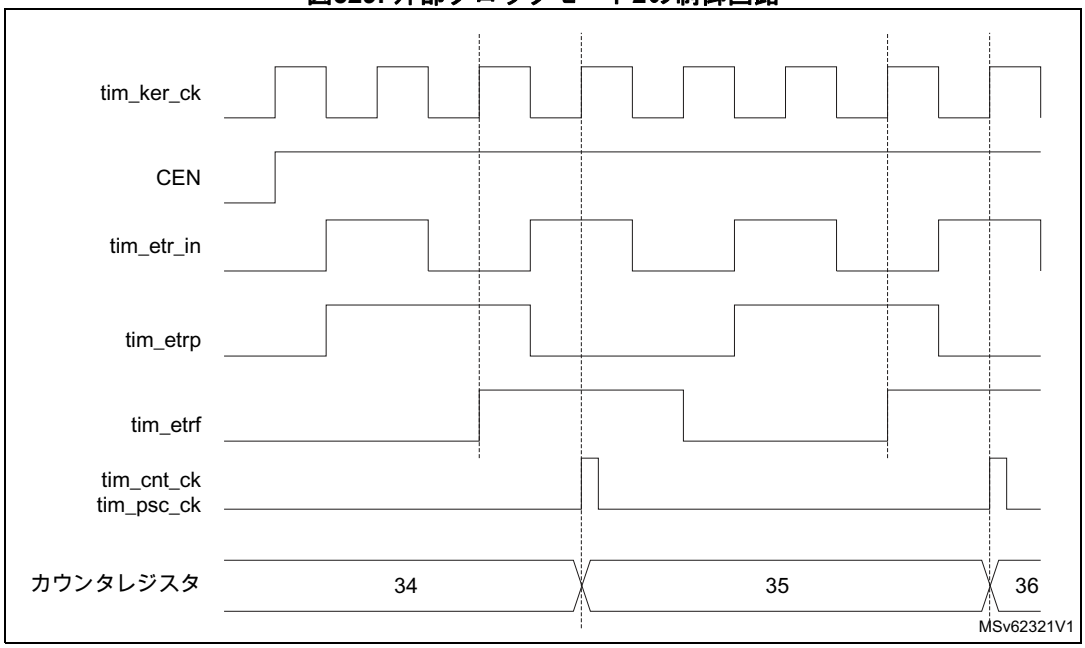
たとえば、tim_etr_inの2回の立上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCRレジスタのETF[3:0] に0000を書き込みます。
2. TIMx_SMCRレジスタにETPS[1:0]=01を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCRレジスタにETP=0を書き込むことによって、tim_etr_in入力の立上がりエッジ検出を選択します。
4. TIMx_SMCRレジスタにECE=1を書き込むことによって、外部クロックモード2を有効にします。
5. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを有効にします。

カウンタは2回のtim_etr_in立上がりエッジごとに1回カウントします。

tim_etr_inの立上がりエッジから実際のカウンタクロックまでの間に、tim_etrp信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大でtim_ker_ck周波数の1/4です。ETRP信号が高速の場合、ユーザは適切なETPSプリスケーラー設定によって外部信号の分周比を適用する必要があります。

図529. 外部クロックモード2の制御回路



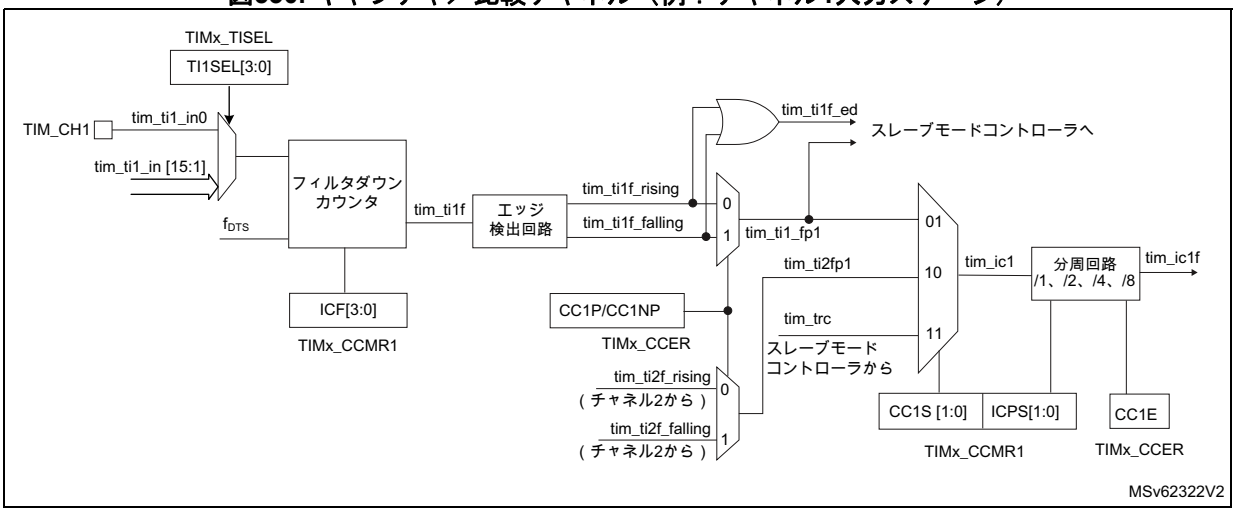
54.3.8 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（チャネル5および6を除くデジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図530から 図533に、1つのキャプチャ／比較 チャネルの概要を示します。

入力ステージは、対応するtim_tix入力をサンプリングして、フィルタリングを行った信号tim_tixfを生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図530. キャプチャ／比較チャネル（例：チャネル1入力ステージ）



出力ステージは、その後、基準として使用される中間波形tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図531. キャプチャ／比較チャネル1メイン回路

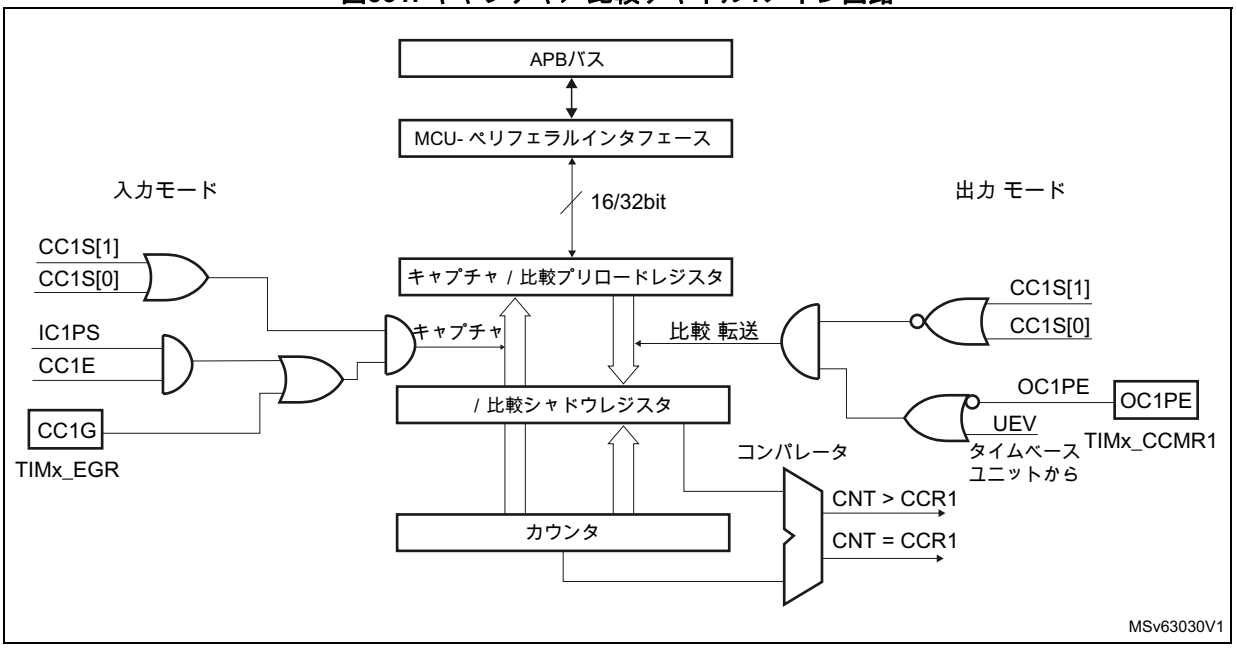
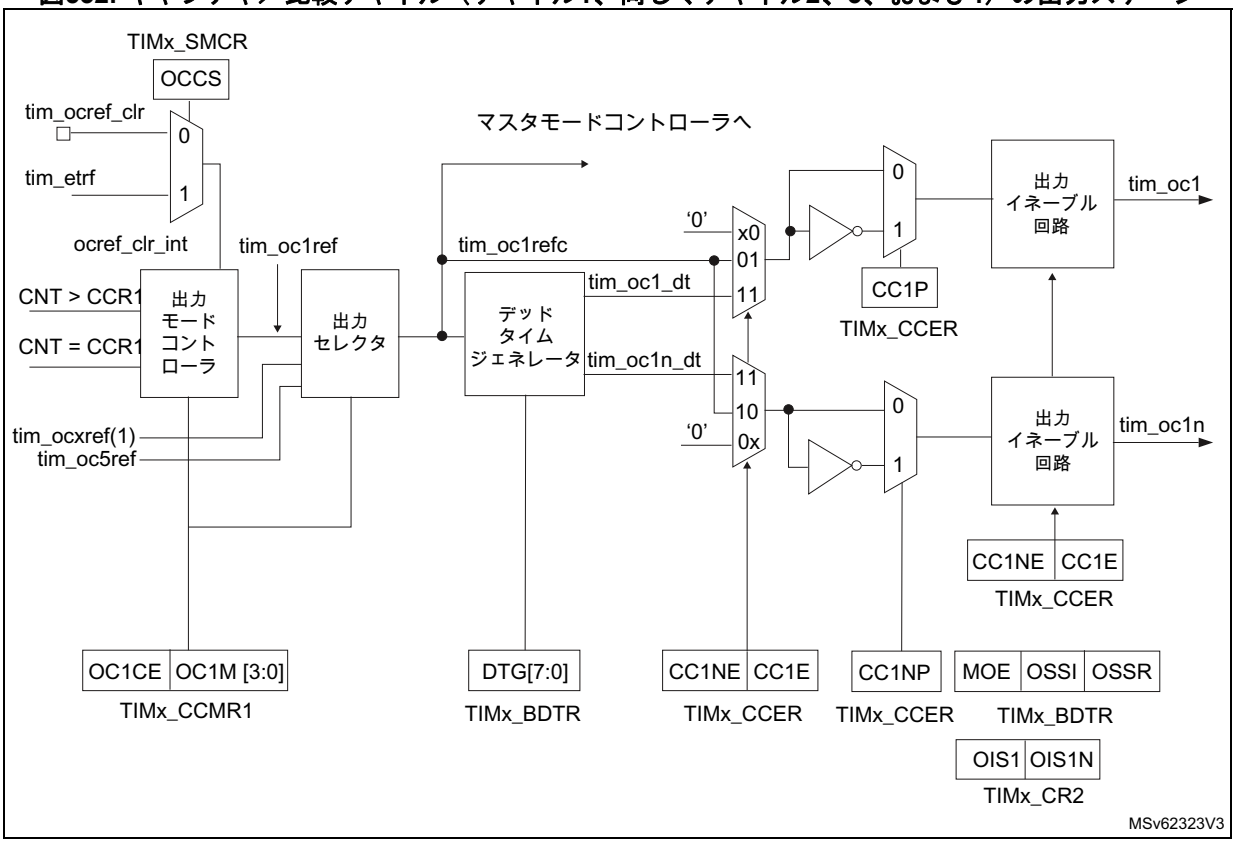
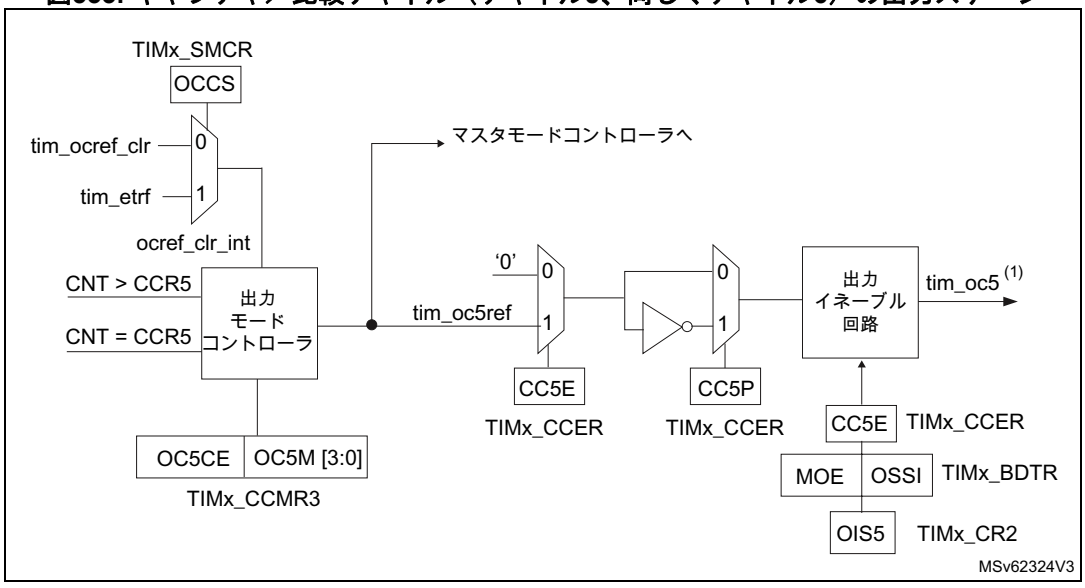


図532. キャプチャ／比較チャネル（チャネル1、同じくチャネル2、3、および4）の出力ステージ



1. tim_ocxref、ここでxは相補チャネルのランク

図533. キャプチャ／比較チャネル（チャネル5、同じくチャネル6）の出力ステージ



1. 外部的には使用できません。

キャプチャ／比較ブロックは、1つのプリロードレジスタと1つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

54.3.9 入力キャプチャモード

入力キャプチャモードでは、対応するICx信号によって遷移が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応するCCXIFフラグ（TIMx_SRレジスタ）がセットされ、割込みまたはDMAリクエストを送信できます（有効な場合）。CCxIFフラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグCCxOF（TIMx_SRレジスタ）がセットされます。CCxIFフラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRxレジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOFは、0を書き込むとクリアされます。

次の例は、tim_ti1入力が立ち上がったときに、カウンタの値をTIMx_CCR1にキャプチャする方法を示します。このためには、次の手順を使用します。

- アクティブ入力を選択します。TIMx_CCR1はtim_ti1入力とリンクされていなければならず、このためにはTIMx_CCMR1レジスタのCC1Sビットに01を書き込みます。CC1Sの値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1レジスタは読出し専用になります。
- タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力がtim_tixの1つである場合、TIMx_CCMRxレジスタのICxFビットによって）。入力信号の反転時、最大で内部クロックの5サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を5クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した8個のサンプルが検出されたときに、tim_ti1の遷移を検証できます（周波数f_{DTS} でサンプリング）。この場合、TIMx_CCMR1レジスタのIC1Fビットに0011を書き込みます。
- tim_ti1チャネルのアクティブ変化のエッジを選択します。このためには、TIMx_CCERレジスタのCC1PビットとCC1NPビットに“0”を書き込みます（この場合、立上がりエッジの選択）。
- 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1レジスタのIC1PSビットに“00”を書き込む）。
- TIMx_CCERレジスタのCC1Eビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
- 必要な場合は、TIMx_DIERレジスタのCC1IEビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIERレジスタのCC1DEビットをセットすることによって、DMAリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値がTIMx_CCR1レジスタに格納されます。
- CC1IFフラグがセットされます（割込みフラグ）。CC1OFビットは、少なくとも2回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IEビットに応じて、割込みが生成されます。
- CC1DEビットに応じて、DMAリクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これは、フラグ読出し後、データ読出し前に発生するかもしれないオーバーキャプチャの見落としを避けるためです。

注： **IC割込みとDMAリクエストは、TIMx_EGRレジスタの対応するCCxGビットをセットすることによって、ソフトウェアによって生成することができます。**

54.3.10 PWM入力モード

このモードは、単一のtim_tix入力に接続されたPWM信号の周期とデューティサイクルの両方を以下のように測定するために使用されます。

- TIMx_CCR1レジスタは周期値（2つの連続した立上がりエッジ間の間隔）を保持します。
- TIM_CCR2レジスタはパルス幅（2つの連続した立上がりと立下がりエッジ間の間隔）を保持します。

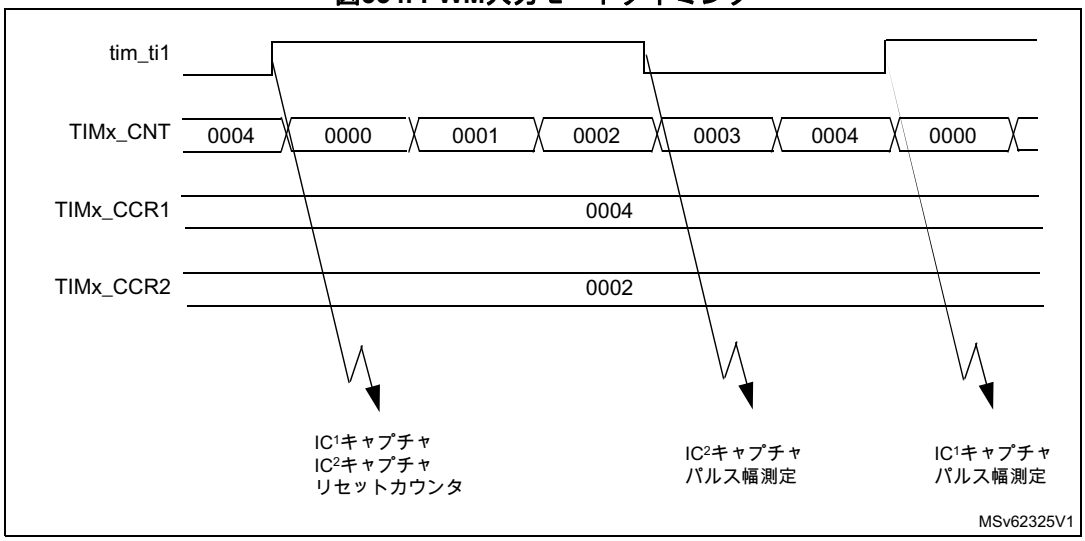
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2つのICx信号が同じtim_tixfp1入力にマッピングされます。
- この2つのICx信号は、逆の極性のエッジでアクティブです。
- 2つのtim_tixfp信号の1つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

tim_ti1に適用されるPWM信号の周期とパルス幅は、以下の手順を使用して測定できます。

- TIMx_CCMR1 レジスタの CC1S ビットに “01” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
- CC1P ビットと CC1NP ビットに“0” を書き込むことによって（立上がりエッジでアクティブ）、tim_ti1fp1のアクティブな極性を選択します（TIMx_CCR1のキャプチャとカウンタクリアの両方に使用）。
- TIMx_CCMR1 レジスタの CC2S ビットに “10” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
- CC2PビットとCC2NPビットにCC2P/CC2NP=“10”を書き込むことによって（立下がりエッジでアクティブ）、tim_ti1fp2のアクティブ極性を選択します（TIMx_CCR2のキャプチャに使用されます）。
- TIMx_SMCRレジスタのTSビットに00101を書き込むことによって（tim_ti1fp1を選択）、有効なトリガ入力を選択します。
- TIMx_SMCR レジスタの SMS ビットに 0100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
- TIMx_CCERレジスタのCC1EとCC2Eビットに“1”を書き込むことによって、キャプチャを有効にします。

図534. PWM入力モードタイミング



54.3.11 強制出力モード

このモード（TIMx_CCMRxレジスタのCCxSビット = 00）では、各出力比較信号（tim_ocxref、そしてtim_ocx/tim_ocxn）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応するTIMx_CCMRxレジスタのOCxMビットに0101を書き込む必要があります。これにより、tim_ocxrefは強制的にハイレベルになり（tim_ocxrefは常にアクティブハイ）、tim_ocxはCCxP極性ビットと逆の値になります。

例：CCxP=0（tim_ocxアクティブハイ）=> tim_ocxは強制的にハイレベルになります。

tim_ocxref信号は、TIMx_CCMRxレジスタのOCxMビットに0100を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRxシャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みやDMAリクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

54.3.12 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。マイクロコントロール内でチャネル5および6のみが使用可能である場合に、チャネル1から4を出力できます（たとえば、合成波形生成またはADCトリガのため）。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCERレジスタのCCxPビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SRレジスタのCCxIFビット）。
- 対応する割込みマスク（TIMx_DIERレジスタのCCXIEビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIERレジスタのCCxDEビット）がセットされている場合は、DMAリクエストを送信します（DMAリクエスト選択には、TIMx_CR2レジスタのCCDSビットが使用されます）。

TIMx_CCRxレジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRxレジスタのOCxPEビットを使用してプログラミングできます。

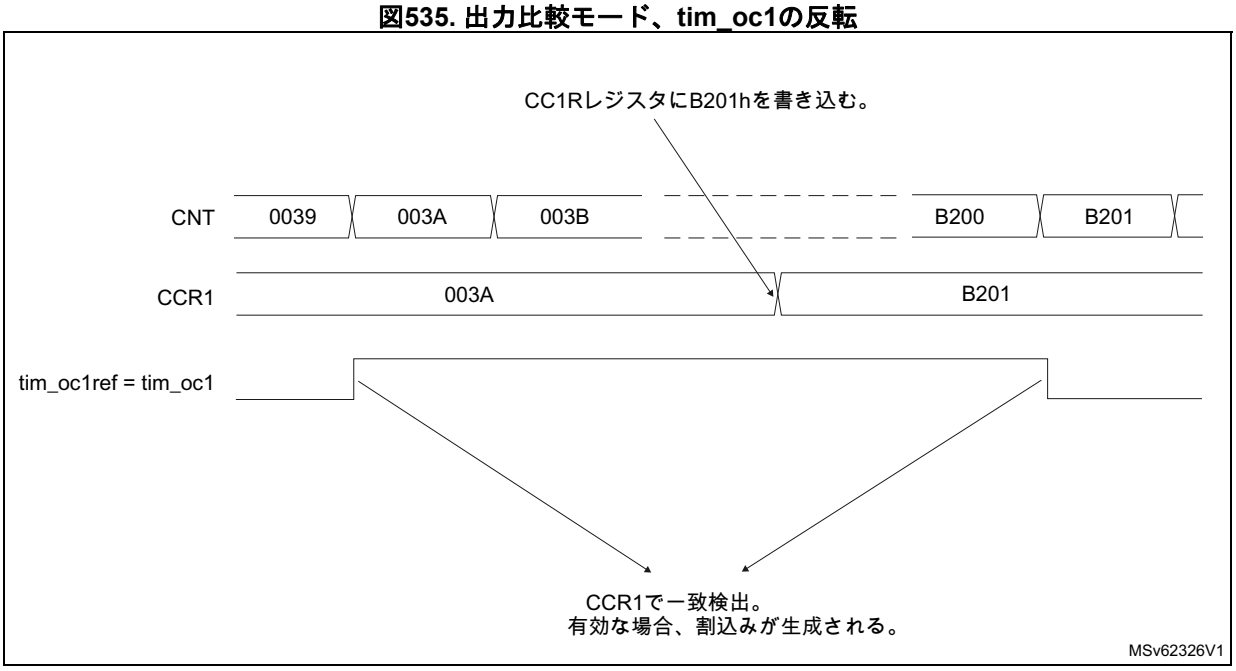
出力比較モードでは、更新イベントUEVはtim_ocxrefおよびtim_ocx出力には影響を与えません。タイミングの分解能はカウンタの1カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARRレジスタとTIMx_CCRxレジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIEビットをセットします。
4. 出力モードを選択します。例：
   - CNTとCCRxが一致したときにtim_ocx出力ピンを反転するには、OCxMビットに0011を書き込みます。
   - プリロードレジスタを無効にするには、OCxPEビットに0を書き込みます。
   - アクティブハイ極性を選択するには、CCxPビットに0を書き込みます。
   - 出力を有効にするには、CCxEビットに1を書き込みます。
5. TIMx_CR1レジスタのCENビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによってTIMx_CCRxレジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRxシャドウレジスタは、次の更新イベントUEVでのみ更新されます。例を 図535に示します。

図535. 出力比較モード、tim_oc1の反転



54.3.13 PWMモード

パルス幅変調（PWM）モードは、TIMx_ARRレジスタの値によって決められた周波数とTIMx_CCRxレジスタの値によって決められたデューティサイクルの信号を生成するために使用されます。

PWMモードは、TIMx_CCMRxレジスタのOCxMビットに“0110”（PWMモード1）または“0111”（PWMモード2）を書き込むことによって、チャネルごとに選択できます（tim_ocx出力ごとに1つのPWM）。TIMx_CCMRxレジスタのOCxPEビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1レジスタのARPEビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGRレジスタのUGビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

tim_ocx極性は、TIMx_CCERレジスタのOCxPビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx出力は、CCxE、CCxNE、MOE、OSSI、およびOSSRビット（TIMx_CCERおよびTIMx_BDTRレジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERレジスタの説明を参照してください。

PWMモード（1または2）では、TIMx_CNTとTIMx_CCRxが常に比較されて、TIMx_CCRx ≤ TIMx_CNTまたはTIMx_CNT ≤ TIMx_CCRxかどうかが判断されます（カウントの方向によります）。

タイマは、TIMx_CR1レジスタのCMSビットに応じて、エッジアラインモードまたはセンターアラインモードでPWMを生成できます。

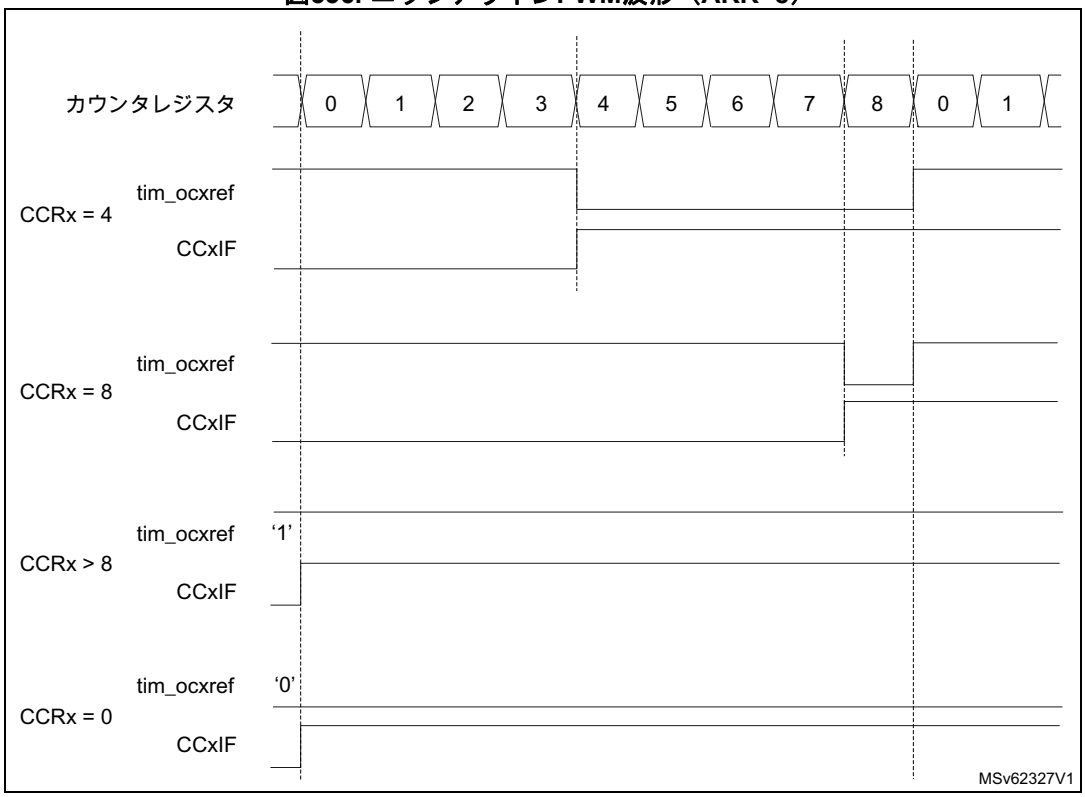
PWMエッジアラインモード

- アップカウント構成

  TIMx_CR1レジスタのDIRビットがローのときには、アップカウントがアクティブです。2081ページのアップカウントモードを参照してください。

  次の例では、PWMモード1を使用しています。PWM基準信号tim_ocxrefは、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値がTIMx_ARRの自動再ロード値より大きい場合、tim_ocxrefは“1”に保持されます。比較値が0の場合、tim_ocxrefは“0”に保持されます。図536にTIMx_ARR=8のときのエッジアラインPWM波形の例を示します。

図536. エッジアラインPWM波形（ARR=8）



- ダウンカウント構成

  TIMx_CR1レジスタのDIRビットがハイのときには、ダウンカウントがアクティブです。2085ページのダウンカウントモードを参照してください。

  PWMモード1では、基準信号tim_ocxrefは、TIMx_CNT > TIMx_CCRxの間はローであり、そうでない場合はハイになります。TIMx_CCRxの比較値がTIMx_ARRの自動再ロード値より大きい場合、tim_ocxrefは“1”に保持されます。このモードでは、0 % のPWM信号を生成することはできません。

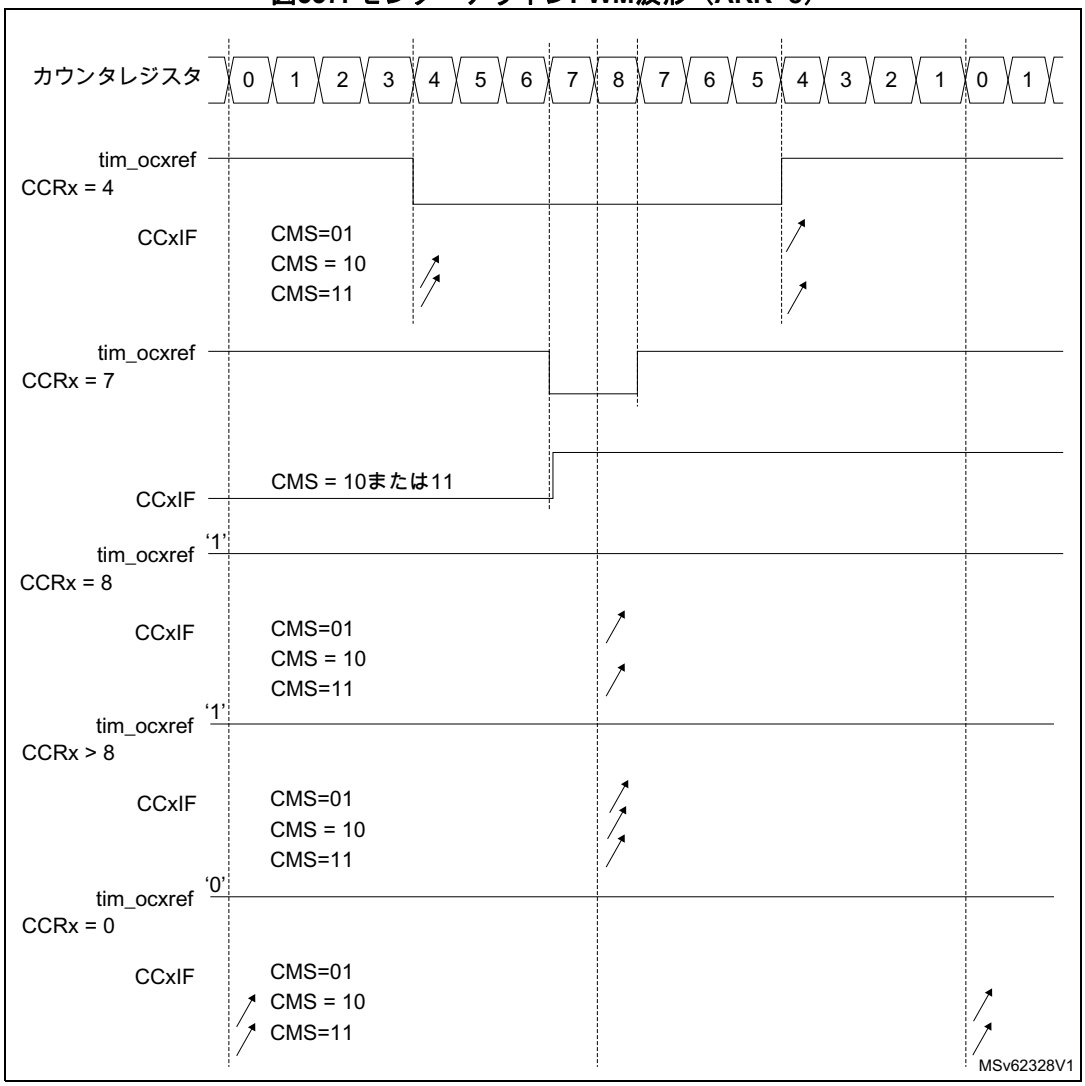
PWMセンターアラインモード

センターアラインモードは、TIMx_CR1レジスタのCMSビットが“00”でないときにアクティブです（その他すべての設定は、tim_ocxref/tim_ocx信号に対して同じ効果を持ちます）。比較フラグは、CMSビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。2089ページのセンターアラインモード（アップ／ダウンカウント）を参照してください。

図537 に、次の条件でのセンターアラインPWM波形の例を示します。

- TIMx_ARR=8
- PWMモードはPWMモード1
- フラグは、TIMx_CR1レジスタのCMS=01で選択されたセンターアラインモード1に対応して、カウンタがカウントダウンするときにセットされます。

図537. センターアラインPWM波形（ARR=8）



センターアラインモードの使用に関するヒント：

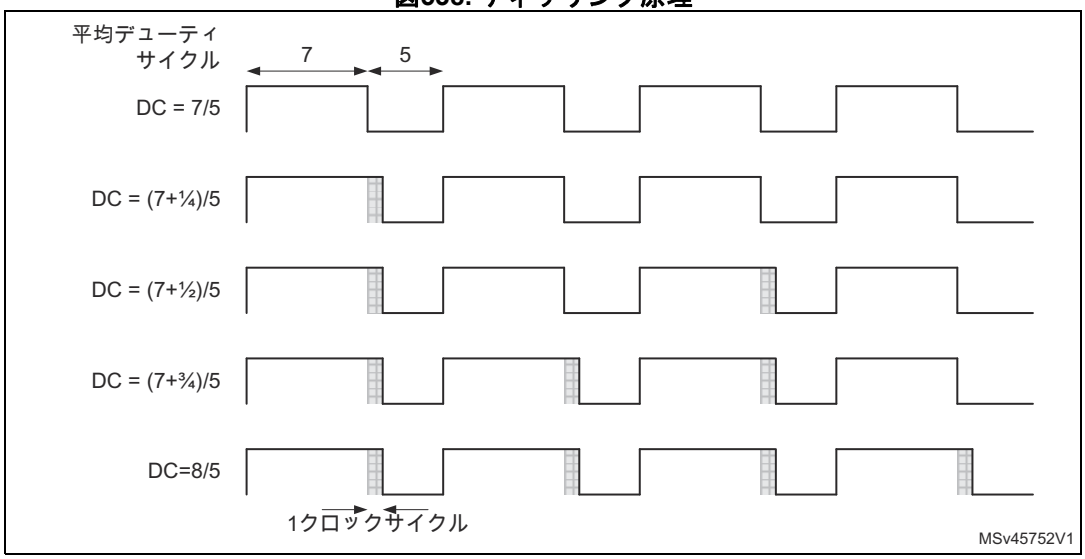
- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1レジスタのDIRビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIRビットとCMSビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに0またはTIMx_ARR値が書き込まれた場合、方向は更新されますが、更新イベントUEVは生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGRレジスタのUGビットをセットする）、動作中はカウンタへの書込みを行わないことです。

ディザリングモード

PWMモードの有効分解能は、TIMx_CR1レジスタのDITHENビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）とARR（PWM周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16個の連続したPWM周期にわたって実際のCCR（またはARR）値をわずかに変更する（1タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたはPWM周期を考えると、16倍の分解能の向上が可能になります。下記の図538は、4つの連続するPWMサイクルに適用されたディザリング原理を示します。

図538. ディザリング原理

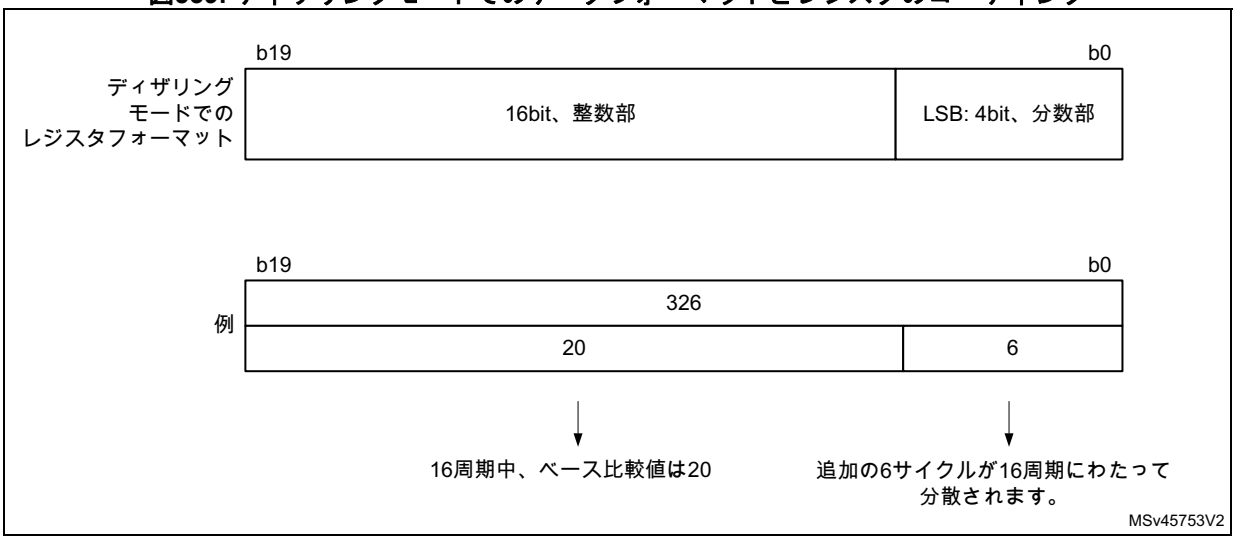


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図539を参照）。

- 下位4bitは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット19:4に左シフトされ、ベース値用のコーディングになります。

注： **DITHENビットをリセットするときは、次の手順に従う必要があります。**
1を参照してください。CENおよびARPEビットをリセットする必要があります。
2を参照してください。DITHENビットをリセットする必要があります。
3を参照してください。CCIFフラグをクリアする必要があります。
4を参照してください。CENビットをセットすることができます（最終的にARPE = 1とともに）。

図539. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

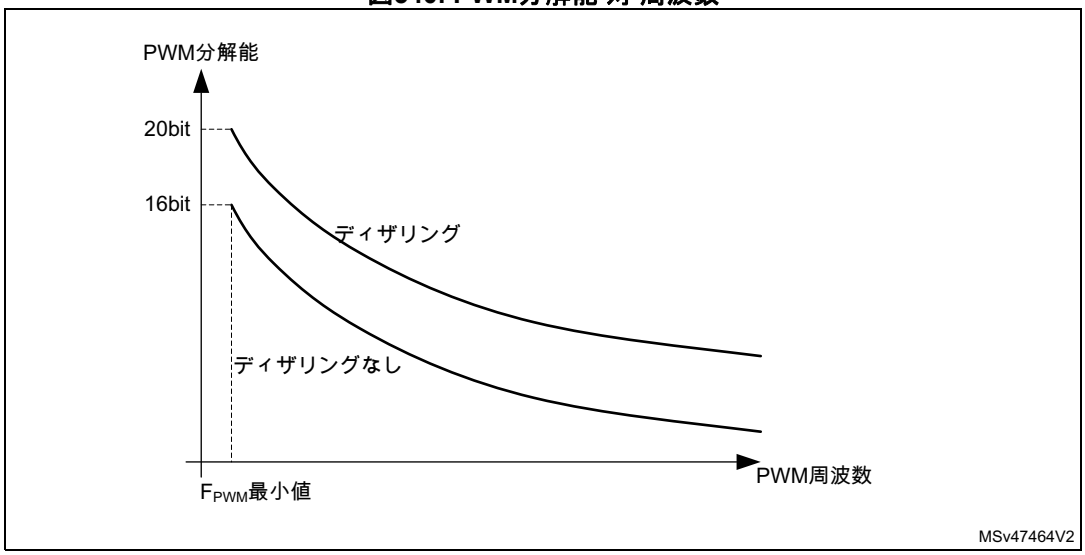
$$\text{ディザリングモードが無効の場合: } F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモードが有効な場合: } F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARRおよびTIMxCCRyの最大値は、ディザリングモードでは0xFFFEFに制限されています（整数部では65534、ディザリング部分では15に対応します）。**

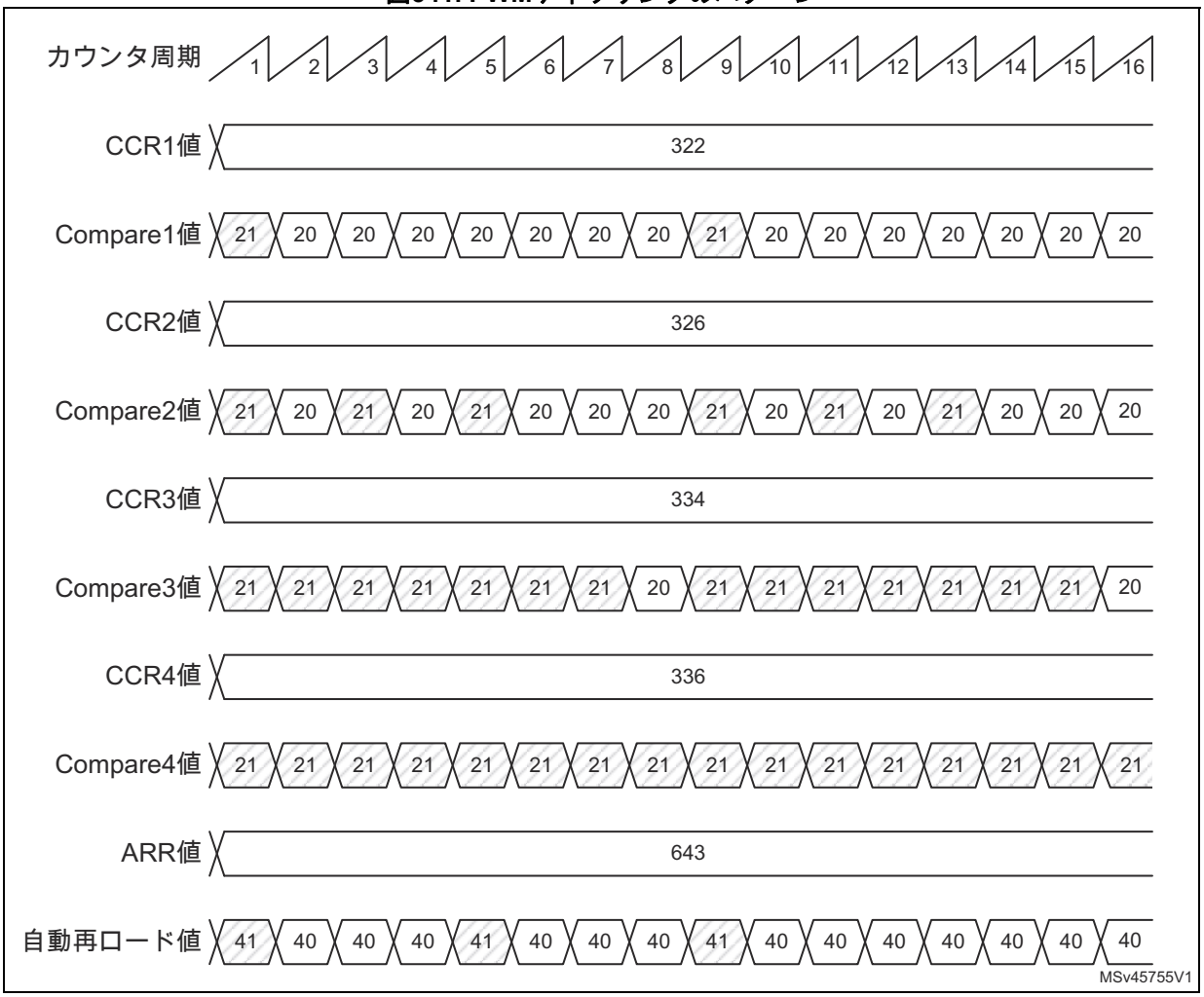
下記の図540に示すように、ディザリングモードはPWM周波数に関係なく、PWM分解能を増加するために使用されます。

図540. PWM分解能 対 周波数



下記の図541で説明するように、デューティサイクルや周期の変更は、連続した16周期にわたって広がります。

図541. PWMディザリングのパターン



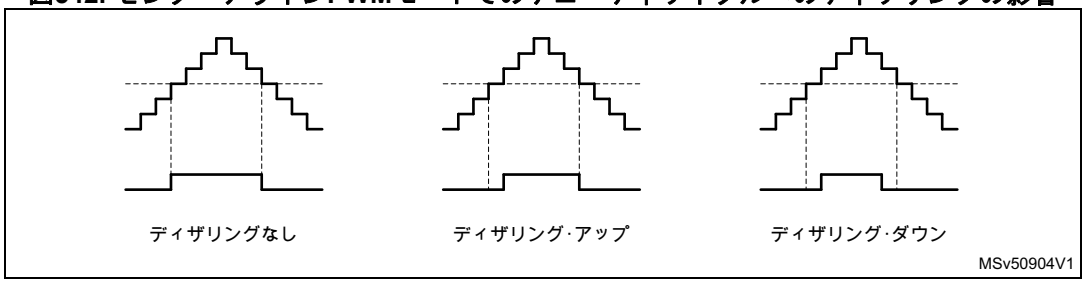
自動再ロード値と比較値の増分は、下記の表539に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表539.CCRおよびARRレジスタの変更のディザリングパターン

LSB値	PWM周期															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

ディザリングモードはセンターアラインPWMモード（TIMx_CR1レジスタのCMSビットが“00”に等しくないとき）でも使用できます。この場合、ディザリングパターンは、以下の図542に示すように、アップおよびダウンカウントフェーズを考慮して、8つの連続したPWM周期にわたって適用されます。

図542. センターアラインPWMモードでのデューティサイクルへのディザリングの影響



下記の表540は、センターアラインPWMモードでどのようにディザリングパターンが追加されるかを示しています。

表540.センターアラインPWMモードでのCCRレジスタ変更のディザリングパターン

LSB値	PWM周期															
	1		2		3		4		5		6		7		8	
	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

54.3.14 非対称PWMモード

非対称モードでは、プログラム可能な位相シフトによって2つのセンターアラインPWM信号の生成を可能にします。周波数がTIMx_ARRレジスタの値で決定されるのに対し、デューティサイクルや位相シフトはTIMx_CCRxレジスタペアで決定されます。1つ目のレジスタがアップカウント時のPWMを制御し、2つ目のレジスタがダウンカウント時のPWMを制御することで、PWMはPWMハーフサイクルごとに調整されます。

- tim_oc1refc（またはtim_oc2refc）は、TIMx_CCR1およびTIMx_CCR2によって制御されます。
- tim_oc3refc（またはtim_oc4refc）は、TIMx_CCR3およびTIMx_CCR4によって制御されます。

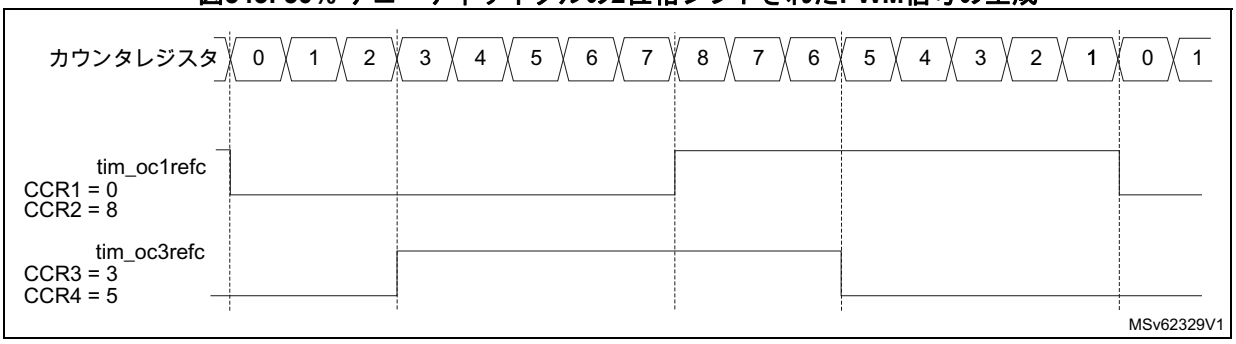
非対称PWMモードは、TIMx_CCMRxレジスタのOCxMビットに“1110”（非対称PWMモード1）または“1111”（非対称PWMモード2）を書き込むことによって、2チャネルごとに選択できます（CCRレジスタペアごとに1つのtim_ocx出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

特定のチャネルが非対称のPWMチャネルとして使用されると、その相補チャネルも使用できます。たとえば、tim_oc1refc信号がチャネル1（非対称PWMモード1）に生成されると、チャネル2のtim_oc2ref信号、または非対称PWMモード1の結果として得られるtim_oc2refc信号を出力できます。

図543は、非対称PWMモードを使用して生成される信号の例を表します（チャネル1から4は非対称PWMモード2として設定されます）。これにより、デッドタイムジェネレータとともにフルブリッジ位相シフトDC-DCコンバータを制御できます。

図543. 50% デューティサイクルの2位相シフトされたPWM信号の生成



54.3.15 組み合わせPWMモード

組み合わせPWMモードでは、2つのエッジアラインまたはセンターアラインPWM信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数がTIMx_ARRレジスタの値で決定されるのに対し、デューティサイクルや遅延は2つのTIMx_CCRxレジスタで決定されます。結果として得られる信号tim_ocxrefcは、2つのPWM基準信号のORまたはANDによる論理結合から成ります。

- tim_oc1refc（またはtim_oc2refc）は、TIMx_CCR1およびTIMx_CCR2によって制御されます。
- tim_oc3refc（またはtim_oc4refc）は、TIMx_CCR3およびTIMx_CCR4によって制御されます。

組合せPWMモードは、TIMx_CCMRxレジスタのOCxMビットに“1100”（組合せPWMモード1）または“1101”（組合せPWMモード2）を書き込むことによって、2チャネルごとに選択できます（CCRレジスタペアごとに1つのtim_ocx出力）。

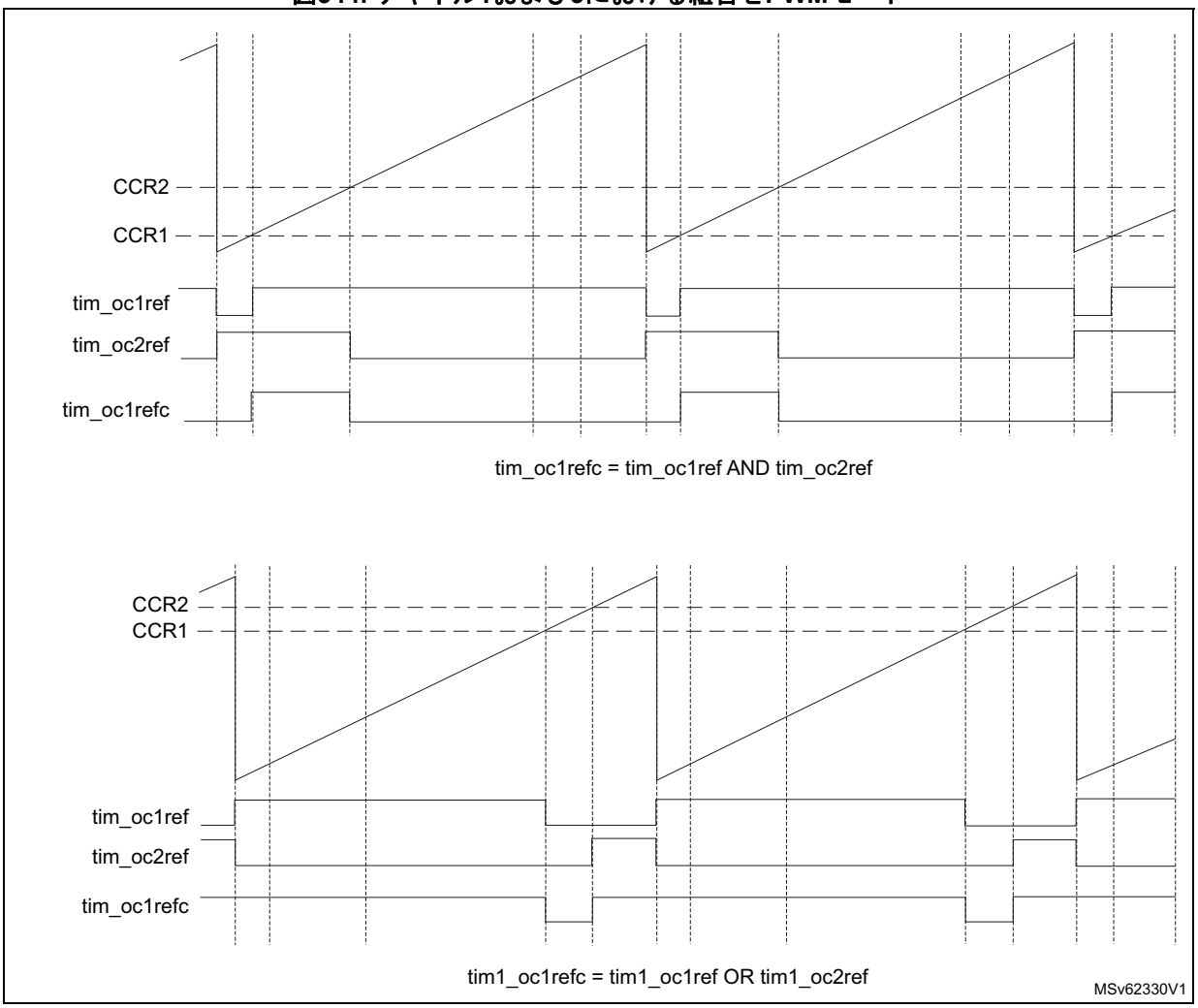
特定のチャネルが組合せPWMチャネルとして使用されている場合、相補チャネルを反対のPWMモードに設定する必要があります（たとえば、1つを組合せPWMモード1、もう1つを組合せPWMモード2にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

図544は、次の設定で取得可能な組合せPWMモードを使用して生成される信号の例を表します。

- チャネル1が組み合わせPWMモード2で設定されている場合
- チャネル2がPWMモード1で設定されている場合
- チャネル3が組み合わせPWMモード2で設定されている場合
- チャネル4がPWMモード1で設定されている場合

図544. チャネル1および3における組合せPWMモード



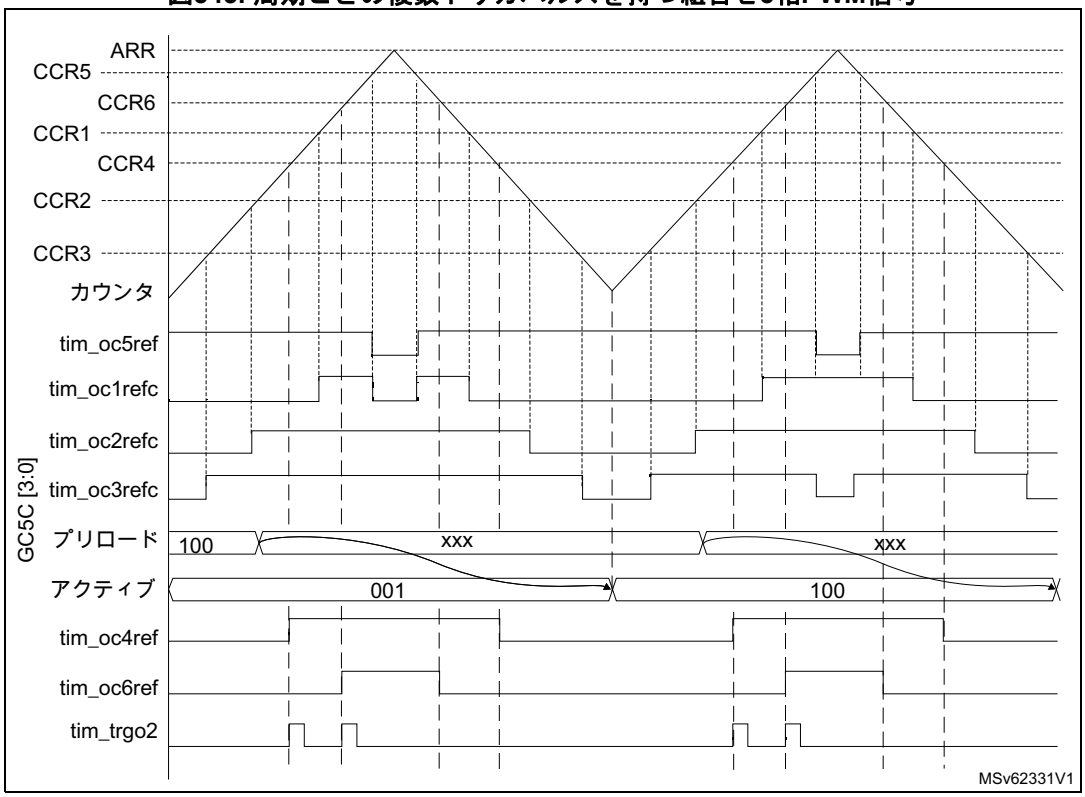
54.3.16 組合せ3相PWMモード

組合せ3相PWMモードでは、パルスの中間で論理積を取った単一のプログラム可能な信号とともに1つから3つのセンターアラインPWM信号を生成できます。結果として得られる組合せ信号の定義には、tim_oc5ref信号が使用されます。TIMx_CCR5の3bit GC5C[3:1] では、tim_oc5refを組合せる基準信号を選択できます。結果として得られる信号tim_ocxrefcは、2つのPWM基準信号のANDによる論理結合から生成されます。

- GC5C1がセットされると、tim_oc1refcはTIMx_CCR1およびTIMx_CCR5によって制御されます。
- GC5C2がセットされると、tim_oc2refcはTIMx_CCR2およびTIMx_CCR5によって制御されます。
- GC5C3がセットされると、tim_oc3refcはTIMx_CCR3およびTIMx_CCR5によって制御されます。

組合せ3相PWMモードは、少なくとも3bit GC5C[3:1] の 1 つをセットすることでチャネル1から3で個別に選択できます。

図545. 周期ごとの複数トリガパルスを持つ組合せ3相PWM信号



tim_trgo2波形は、特定の3相PWM信号でのADCの同期方法を示します。詳細については、セクション 54.3.31：ADCトリガを参照してください。

54.3.17 相補出力とデッドタイム挿入

高機能制御タイマ（TIM1/TIM8）は、2つの相補信号を出力して、出力時のスイッチオンおよびスイッチオフを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 tim_ocx または相補 tim_ocxn）の極性は出力ごとに独自に選択できます。これはTIMx_CCERレジスタのCCxPビットおよびCCxNPビットへの書込みによって行います。

相補信号 tim_ocx および tim_ocxn は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタとTIMx_CR2 レジスタのMOE、OISx、OISxN、OSSI、およびOSSRビットといった複数の制御ビットの組合せによって有効になります。詳細については、表 548：2188ページのブレーク機能を持つ相補tim_ocxおよびtim_ocxnチャネルの出力制御ビットを参照してください。特に、IDLE状態に切り替わるとき（MOEが0になるとき）に、デッドタイムが挿入されます。

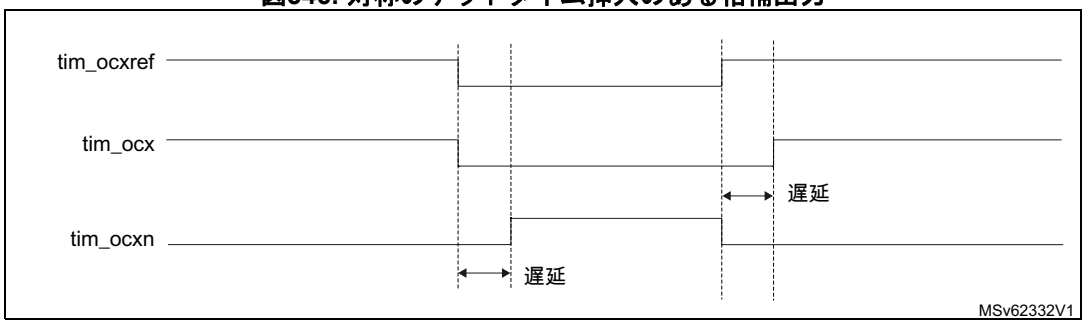
デッドタイム挿入は、CCxEビットとCCxNEビットの両方をセットし、ブレーク回路がある場合は、さらにMOEビットをセットすることによって有効になります。各チャネルに1つの10bitデッドタイムジェネレータがあります。この回路は、基準波形tim_ocxrefからtim_ocxとtim_ocxnの2つの出力を生成します。tim_ocxとtim_ocxnがアクティブハイの場合、

- tim_ocx出力信号は基準信号と同じですが、立上がりエッジが基準の立上がりエッジより遅い点が異なります。
- tim_ocxn出力信号は、立上がりエッジが基準波形の立下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（tim_ocxまたはtim_ocxn）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号tim_ocxrefとの関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、およびCCxNE=1を想定しています。）

図546. 対称のデッドタイム挿入のある相補出力



TIMx_DTR2のDTAEビットは、図547に示すように、デッドタイム値を、基準信号の立上がりエッジと立下がりエッジに対して変えるために使用されます。

非対称モード（DTAE = 1）では、立上がりエッジ基準デッドタイムはTIMx_BDTRレジスタのDTG[7:0]ビットフィールドによって定義され、立下がりエッジ基準はTIMx_DTR2レジスタのDTGF[7:0] ビットフィールドによって定義されます 。DTAEビットは、カウンタを有効にする前に書き込む必要があり、CEN=1の間は変更してはなりません。

プリロードメカニズムを使用して、PWM運転中にデッドタイム値を動作中に更新することができます。デッドタイムビットフィールドDTG[7:0] とDTGF[7:0] は、TIMX_DTR2レジスタでDTPEビットがセットされるとプリロードされます。プリロード値は、次の更新イベントでアクティブレジスタにロードされます。

注： **カウンタが有効なときにDTPE ビットを有効にした場合、最後の更新以降に書き込まれた新しい値は破棄され、以前の値が使用されます。**

図547. 非対称のデッドタイム

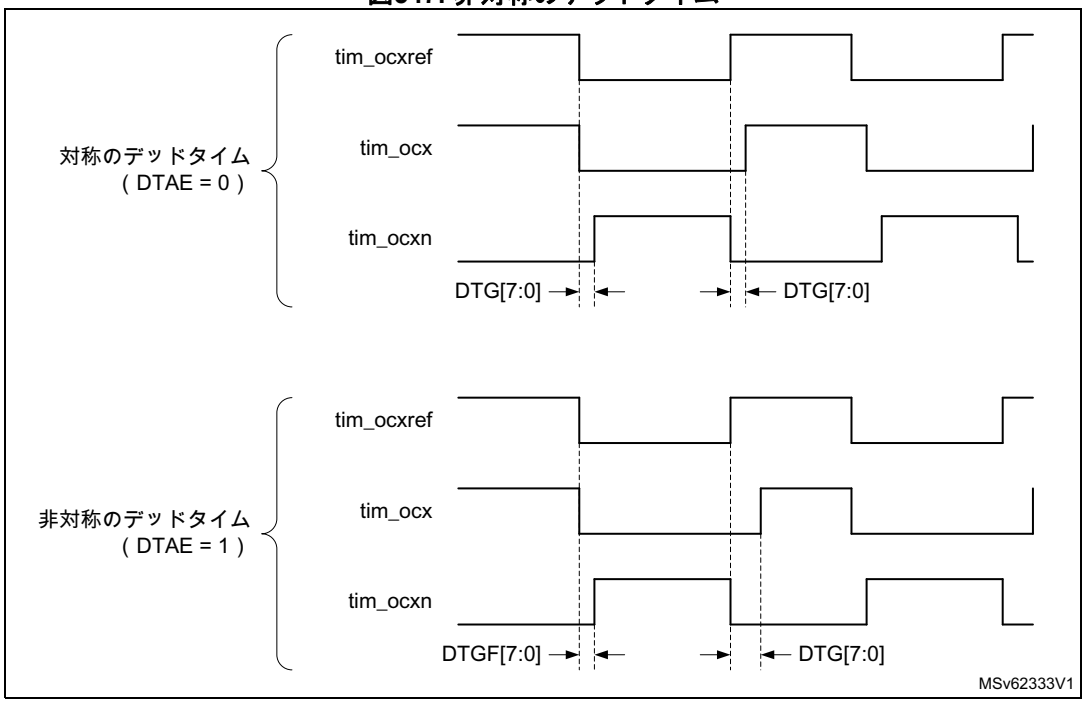


図548. 負のパルスより長い遅延があるときのデッドタイムの波形

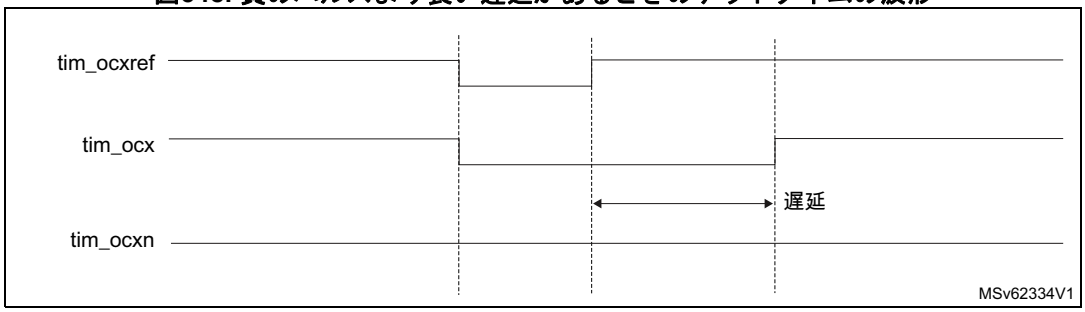
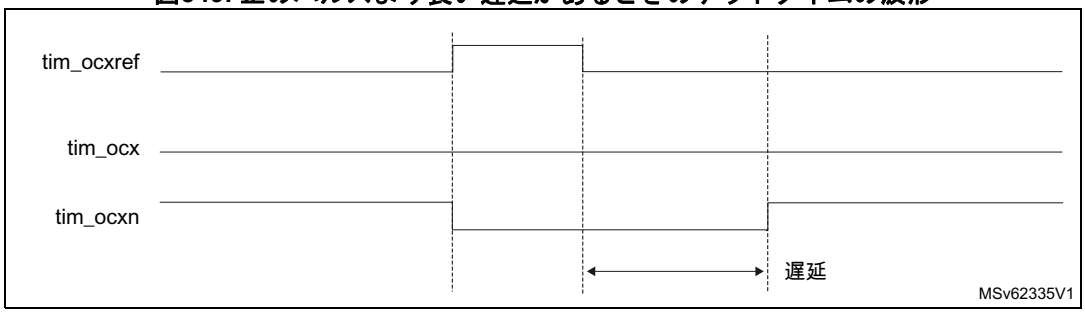


図549. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTRレジスタのDTGビットでプログラム可能です。遅延計算については、セクション 54.6.20：TIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 1、8）を参照してください。

tim_ocxref信号のtim_ocxまたはtim_ocxnへのリダイレクト

出力モード（強制、出力比較、またはPWM）では、TIMx_CCERレジスタのCCxEビットおよびCCxNEビットを設定することによって、tim_ocxref信号をtim_ocx出力またはtim_ocxn出力にリダイレクトできます。

これは、特定の波形（PWMまたは静的アクティブレベルなど）を一方の出力に送信し、相補信号をインアクティブレベルに固定するために使用されます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **tim_ocxnのみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、tim_ocxrefがハイレベルになるとすぐにアクティブになります。たとえば、CCxNP=0の場合は、tim_ocxn=tim_ocxrefになります。他方、tim_ocxとtim_ocxnの両方が有効なときには（CCxE=CCxNE=1）、tim_ocxrefがハイになるとtim_ocxはアクティブになり、tim_ocxnは相補信号となり、tim_ocxrefがローのときにアクティブになります。**

54.3.18 ブレーク機能の使用

ブレーク機能の目的は、タイマによって生成されるPWM信号によって駆動する電源スイッチを保護することです。2つのブレーク入力は通常、パワーステージおよび3相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路はPWM出力を遮断し、強制的に事前定義された安全な状態に移行させます。出力の遮断をトリガするために、いくつかの内部MCUイベントを選択することも可能です。

このブレークには、2つのチャネルがあります。システムレベル障害（クロック障害、ECC/パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵コンパレータからの障害）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できるブレークチャネル。アプリケーション障害のみを含み、出力をインアクティブ状態に強制できるブレーク2チャネル。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタのMOEビット。ソフトウェアによって出力を有効／無効にするために使用され、ブレーク または ブレーク2イベント時にリセットされます。
- TIMx_BDTRレジスタのOSSIビット。出力をインアクティブ状態で制御するか、GPIOコントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2レジスタのOISxおよびOISxNビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISxおよびOISxNの値にかかわらず、一度にtim_ocx出力とtim_ocxn出力を両方ともアクティブレベルにセットすることはできません。詳細については、表 548：2188ページのブレーク機能を持つ相補tim_ocxおよびtim_ocxnチャネルの出力制御ビットを参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOEビットはローになります。ブレーク機能は、TIMx_BDTRレジスタのBKEビットおよびBK2Eビットをセットすることによって有効にできます。ブレーク入力の極性は、同じレジスタのBKPおよびBK2Pビットを設定することによって選択できます。BKExとBKPxは、同時に変更できます。BKExおよびBKPxビットが書き込まれるとき、書込みが有効になるまでに1 APBクロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには1 APBクロックサイクル待つ必要があります。

MOEの立下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTRレジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOEがローになった後で1に設定された場合、MOEを正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

ブレーク（tim_brk）チャネルのソースは以下のいずれかです。

- （GPIOオルタネート機能選択レジスタでの選択に従って）TIMx_BKINピンの1つに接続された外部ソースで、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - tim_brk_cmpx入力から着信するもの（製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照）
  - システムブレークリクエストから着信するもの（製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照）

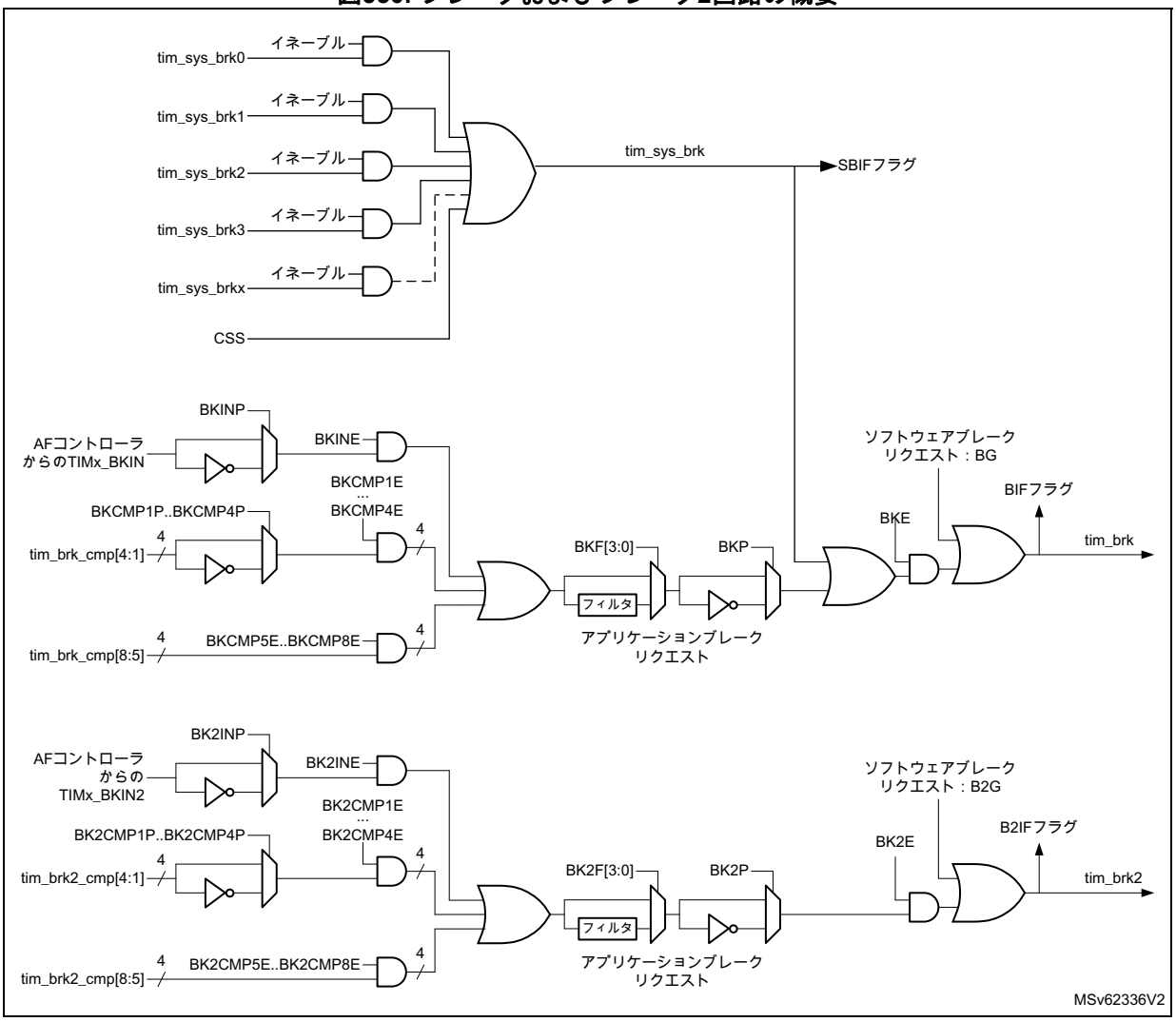
ブレーク2（tim_brk2）のソースは以下のいずれかです。

- （GPIOオルタネート機能選択レジスタでの選択に従って）TIMx_BKIN2ピンの1つに接続された外部ソースで、極性選択およびオプションのデジタルフィルタリングあり
- tim_brk2_cmpx 入力から着信する内部ソース（製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照）

ブレークイベントは、TIMx_EGRレジスタでBGおよびB2Gビットを使用して、ソフトウェアによって生成することもできます。

以下の図550のとおり、すべてのソースはタイマtim_brkまたはtim_brk2入力に入る前に論理和がとられます。

図550. ブレークおよびブレーク2回路の概要



注： **非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部PLLやCSSを使用）を使用する必要があります。**

ブレークが1つ発生すると（ブレーク入力の1つで選択されたレベル）、

- MOEビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、GPIOコントローラへの制御が解除されます（OSSIビットで選択）。これは、MCUオシレータがオフの場合に有効です。
- 各出力チャネルは、MOE=0になったとき、TIMx_CR2レジスタのOISxビットでプログラミングされたレベルで駆動されます。OSSI=0の場合、タイマは出力の制御（GPIOコントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後にOISxおよびOISxNビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、tim_ocxとtim_ocxnを同時にアクティブレベルに駆動することはできま

せん。MOEの再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約2 tim_ker_ckクロックサイクル）。

– OSSI=0の場合、タイマは出力の制御（ハイインピーダンス状態を強制するGPIOコントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxEまたはCCxNEビットのどちらかがハイになったときにハイになります。

- ブレーク状態フラグ（TIMx_SRレジスタのSBIF、BIF、およびB2IFビット）がセットされます。TIMx_DIERレジスタのBIEビットがセットされている場合、割込みが生成されます。TIMx_DIERレジスタのBDEビットがセットされている場合、DMAリクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント（UEV）で再び自動的にセットされます。たとえば、これを使用してレギュレーションを行うことができます。そうでない場合、MOEビットはアプリケーションが再び“1”をセットするまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

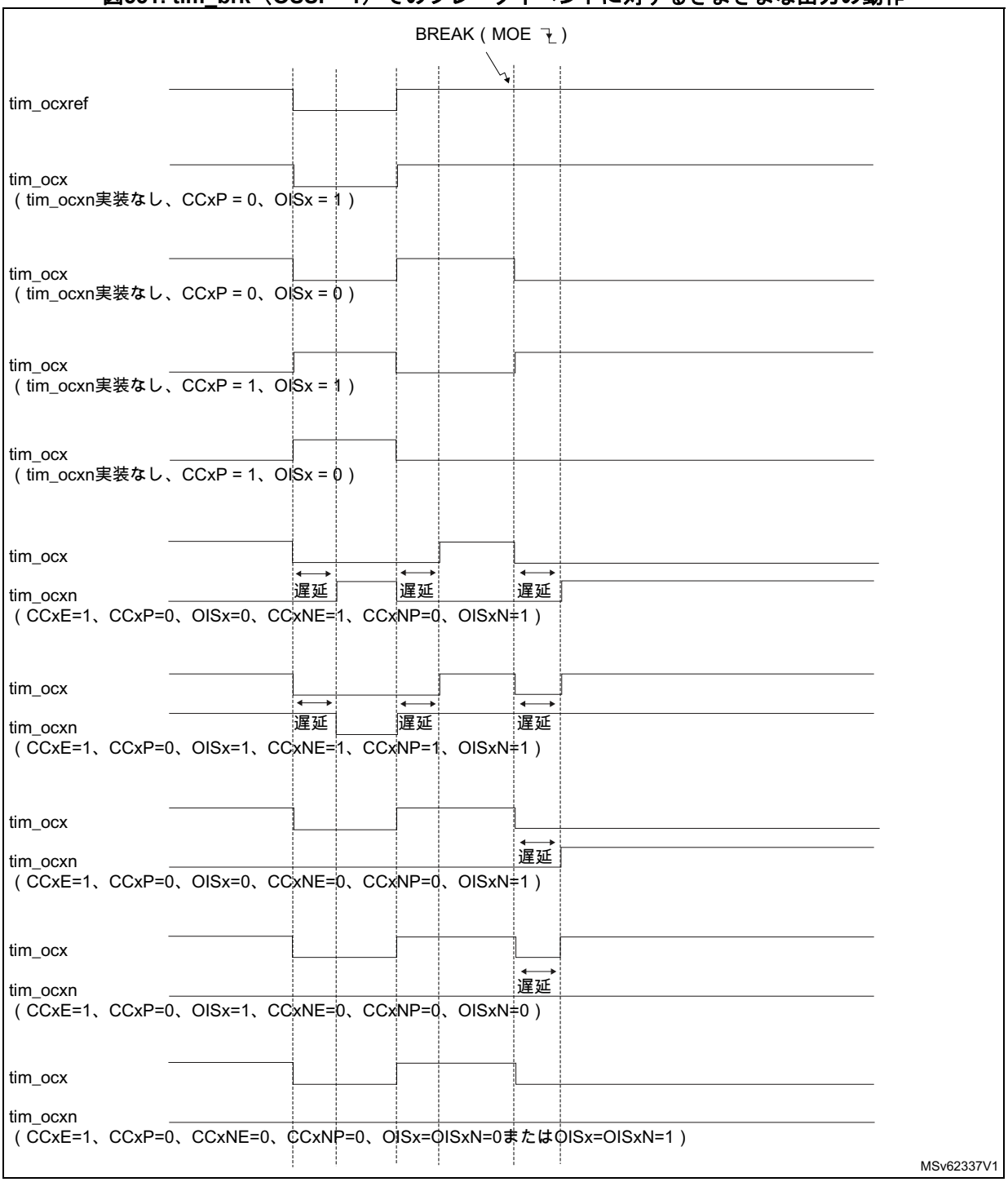
注： **AOEビットが設定されている状態でCPUによりMOEがリセットされると、出力はアイドル状態になり、OSSI値に応じてインアクティブレベルまたはハイインピーダンスに強制されます。MOEビットとAOEビットの両方がCPUによってリセットされた場合、出力は無効状態になり、TIMx_CR2レジスタのOISxビットでプログラムされたレベルで駆動されます。**

注： **ブレーク入力は、信号レベルに対してアクティブです。このため、ブレーク入力がアクティブな間は、MOEをセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグBIFおよびB2IFをクリアできません。**

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これは、いくつかのパラメータ（デッドタイムの長さ、tim_ocx/tim_ocxn極性、無効時の状態、OCxM構成、ブレークイネーブルと極性）の設定を固定するために使用されます。TIMx_BDTRレジスタのLOCKビットによって、3レベルの保護を選択することができます。セクション 54.6.20 : TIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 1、8）を参照してください。LOCKビットは、MCUリセット後に1回だけ書き込むことができます。

図551に、ブレークに対する出力の動作例を示します。

図551. tim_brk（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作



2つのブレーク入力は、タイマ出力で異なる動作を示します。

- tim_brk 入力は、無効化（インアクティブ状態）するか、PWM 出力を強制的に事前定義した安全な状態に移行できます。
- tim_brk2は、PWM出力の無効化（インアクティブ状態）のみ可能です。

表 541に示すように、tim_brkの優先順位はtim_brk2入力よりも高いです。

注： **tim_brk2はOSSR = OSSI = 1の場合にのみ使用してください。**

表541.タイマ出力とtim_brk/tim_brk2入力の動作

tim_brk	tim_brk2	タイマ出力状態	通常の使用例	
			tim_ocxn出力（ローサイドスイッチ）	tim_ocx出力（ハイサイドスイッチ）
アクティブ	X	– 非アクティブから強制される出力状態（デッドタイム後） – OSSI = 0の場合、出力は無効（GPIOロジックが制御を引き継ぐ）	デッドタイム挿入後ON	OFF
非アクティブ	アクティブ	非アクティブ	OFF	OFF

図552では、tim_brkおよびtim_brk2入力で信号がアクティブな場合の、tim_ocxおよびtim_ocx出力の動作の例を示します。この場合、両方の出力がアクティブハイ極性になります（TIMx_CCERレジスタのCCxP = CCxNP = 0）。

図552. tim_brkおよびtim_brk2のアサート後のPWM出力状態（OSSI=1）

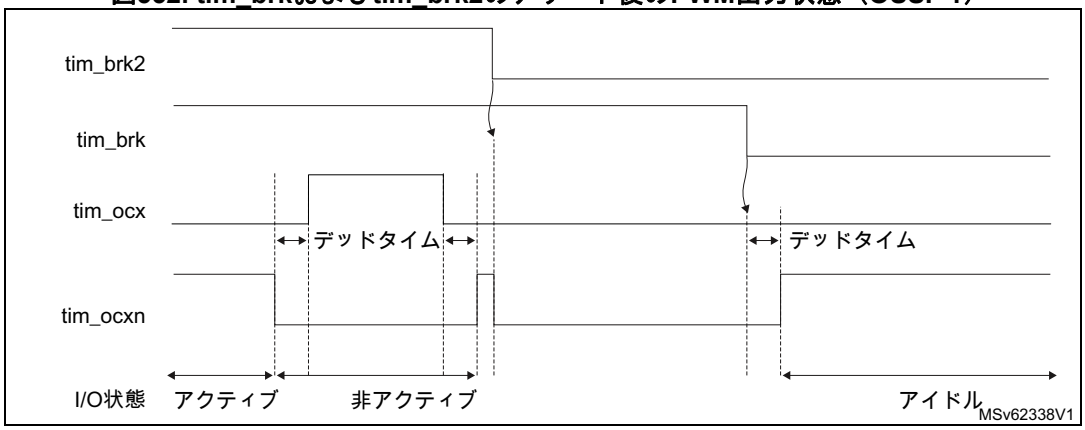
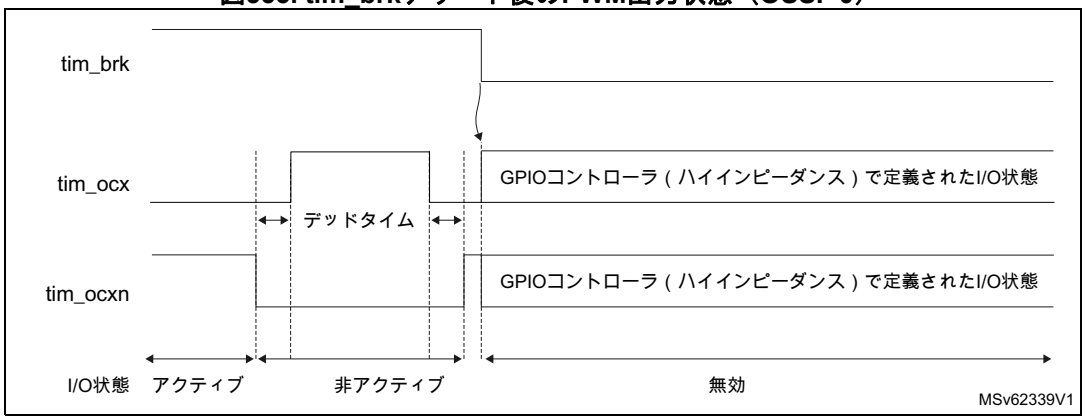


図553. tim_brkアサート後のPWM出力状態（OSSI=0）



54.3.19 双方向ブレーク入力

TIM1/TIM8 には、図554に示すとおり、双方向ブレークI/Oが搭載されています。

これによって、以下がサポートされます。

- 入力および出力ステータスピンどちらにもなるユニークなピンを持ち、外部のMCUやゲートドライバに障害を通知するために使用できる幅広いレベルのグローバルブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、ユニークなブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインソース

tim_brkおよびtim_brk2入力は、TIMxBDTRレジスタのBKBIDおよびBK2BIDビットを使用して双方向モードに設定されます。BKBIDプログラミングビットは、TIMxBDTRレジスタのLOCKビットを使用して読出し専用モードにロックできます（LOCKレベル1以上）。

双方向モードは、tim_brkとtim_brk2入力の両方で使用でき、I/Oをアクティブロー極性でオープンドレインモードに設定する必要があります（BKINP、BKP、BK2INP、BK2Pビットを使用）。システム（たとえばCSS）、内部ペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローにされます。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

ブレークソフトウェアイベント（BGおよびB2G）も、ブレークI/Oを強制的に0にして、タイマがブレーク状態に移行されたことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BKEまたはB2KE = 1）にのみ有効です。ソフトウェアブレークイベントがBKEまたはB2KE = 0で生成された場合、出力が安全な状態になり、ブレークフラグはセットされますが、TIMx_BKINおよびTIMx_BKIN2 I/Oには効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRM（BK2DSRM）ビットが1にセットされると、この機能によりブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM（BK2DSRM）ビットがセットされており、オーブンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM出力の再開を防ぐことができます。
- 出力が有効になっている（MOEビットがセットされている）限り、BKDSRM（BK2DSRM）ビットでブレーク保護を解除できません（表542を参照）。

表542.ブレーク保護解除条件

MOE	BKBID（BK2BID）	BKDSRM（BK2DSRM）	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

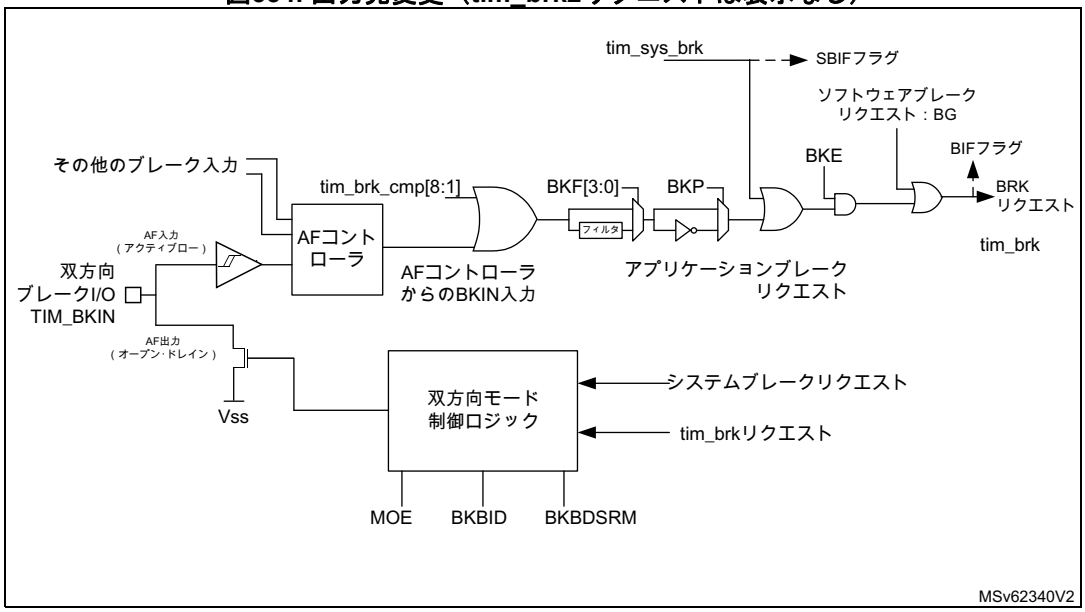
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレーク（ブレーク2）イベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRM（BK2DSRM）ビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM（BK2DSRM）ビットをハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOEビットをセットして、PWM出力を再び有効にできます。

図554. 出力先変更（tim_brk2リクエストは表示なし）

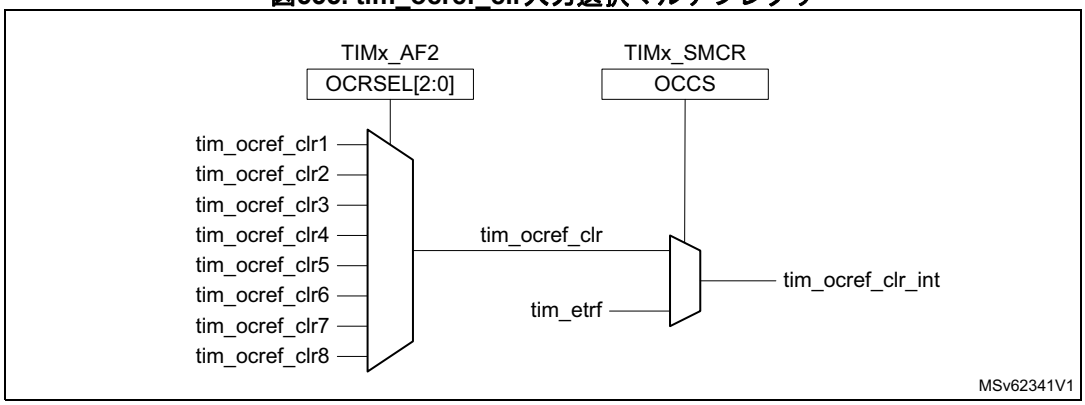


54.3.20 外部イベントによるtim_ocxref信号のクリア

特定のチャネルのtim_ocxref信号はtim_ocref_clr_int入力にハイレベルを適用するとクリアされます（対応するTIMx_CCMRxレジスタのOCxCEイネーブルビットを1にセットする）。tim_ocxrefは、次のPWMサイクルでの次のアクティブステートへの遷移まで、ローのまま残ります。この機能は、出力比較モードとPWMモードでのみ使用可能です。強制モードでは動作しません。tim_ocref_clr_int入力は、TIMx_SMCRレジスタでOCCSビットを設定することで、tim_ocref_clr入力とtim_etrf（フィルタ後のtim_etr_in）の間で選択できます。

tim_ocref_clr入力は、以下の 図555に示すように、TIMx_AF2レジスタのOCRSEL[2:0] ビットフィールドを使用して複数の入力から選択できます。製品で利用できるソースの一覧については、セクション 54.3.2：TIM1/TIM8ピンおよび内部信号 を参照してください。

図555. tim_ocref_clr入力選択マルチプレクサ

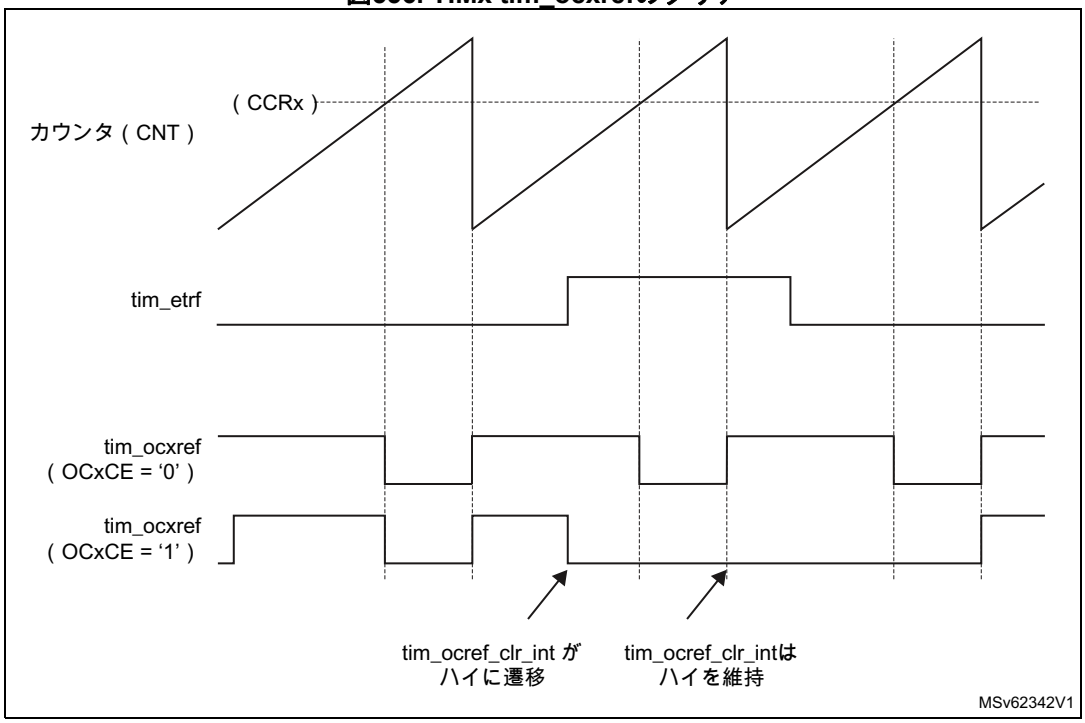


tim_etrfが選択された場合、tim_etr_inは次のように設定する必要があります。

1. 外部トリガプリスケーラはオフ状態に維持する必要があります（TIMx_SMCRレジスタのETPS[1:0] ビットを”00”にセット）。
2. 外部クロックモード2を無効にします（TIMx_SMCRレジスタのECEビットを”0”にセット）。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて（トリガに接続されたソースの極性、およびフィルタを使ってノイズを除去する最終的なニーズに従って）設定できます。

図556 に、イネーブルビットOCxCE の両方の値について、tim_etrf 入力がハイになったときのtim_ocxref信号の動作を示します。この例では、TIMxタイマはPWMモードにプログラミングされています。

図556. TIMx tim_ocxrefのクリア



注： **100% デューティサイクルのPWMの場合 （CCRx > ARRの場合）、次のカウンタオーバーフローでtim_ocxrefが再度有効になります。**

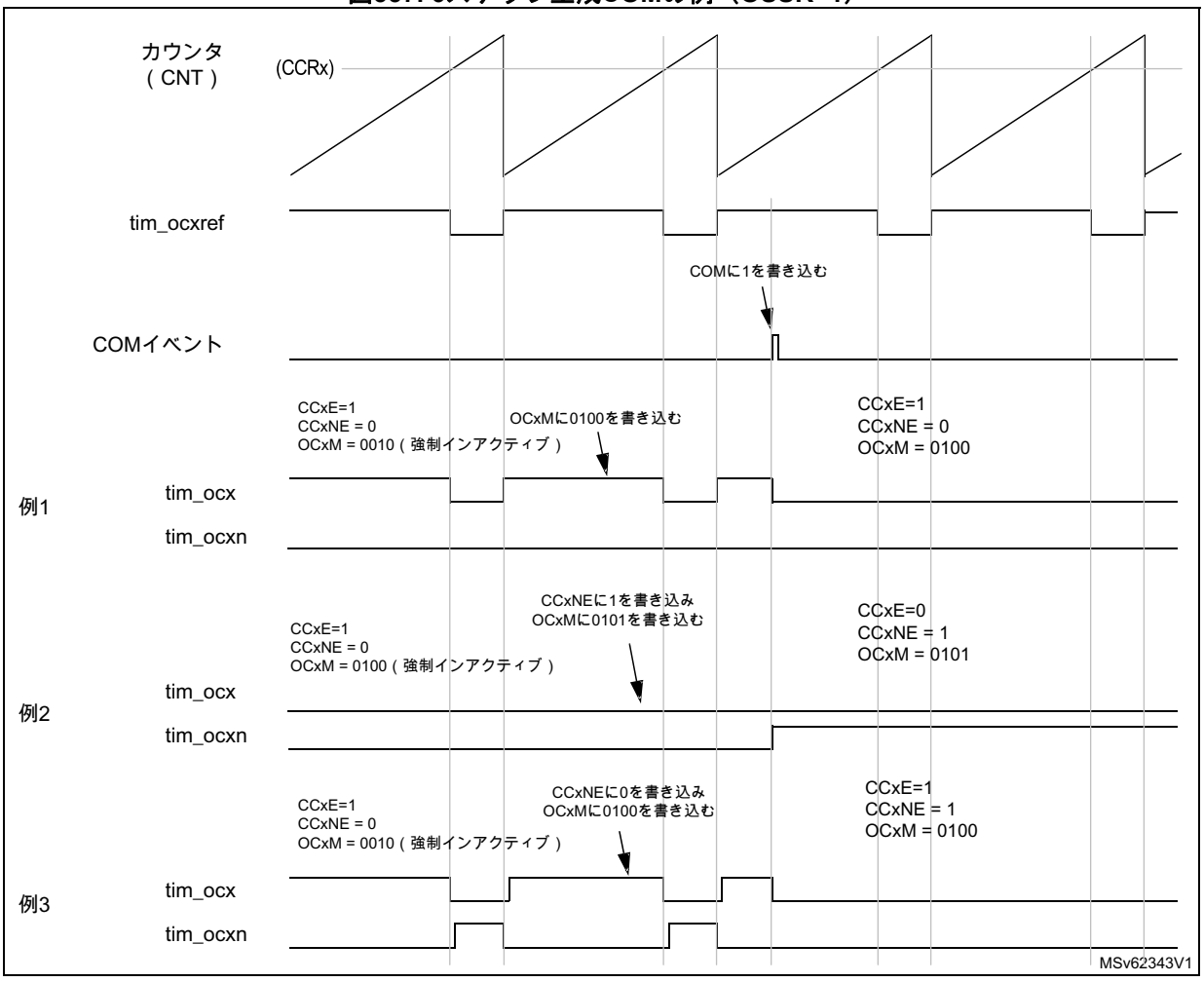
54.3.21 6ステップPWM生成

チャネルで相補出力が使用されているときには、OCxM、CCxE、およびCCxNEビットでプリロードビットが使用できます。プリロードビットは、COM転流イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COMは、TIMx_EGRレジスタのCOMビットをセットすることによってソフトウェアによって、またはハードウェアによって（tim_trgi立上がりエッジで）生成することができます。

フラグは、COMイベントが発生したときにセットされ（TIMｘ_SRレジスタのCOMIFビット）、これによって割込み（TIMx_DIERレジスタのCOMIEビットがセットされている場合）またはDMAリクエスト（TIMx_DIERレジスタのCOMDEビットがセットされている場合）を生成できます。

図557に、COMイベントが発生したときのtim_ocxとtim_ocxn出力の動作を、3種類のプログラミング構成の例で示します。

図557. 6ステップ生成COMの例（OSSR=1）



54.3.22 ワンパルスモード

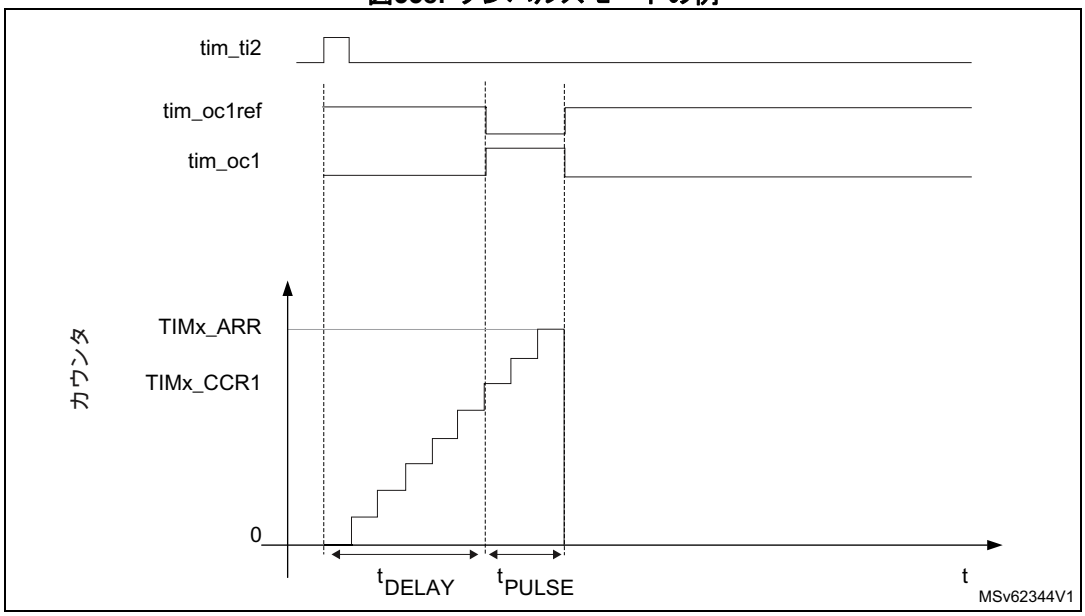
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたはPWMモードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1レジスタのOPMビットをセットします。これによって、カウンタは、次の更新イベントUEVで自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- アップカウント時：CNT < CCRx ≤ ARR（特に、0 < CCRx）
- ダウンカウント時：CNT > CCRx

図558. ワンパルスモードの例



たとえば、tim_ti2入力ピンで立上がりエッジが検出されたときに、tim_oc1にパルス幅がt_{PULSE} の正のパルスを遅延時間t_{DELAY} 後に生成することもできます。

tim_ti2fp2をトリガ1として使用します。

- TIMx_CCMR1レジスタのCC2Sビットに”01”を書き込むことによって、tim_ti2fp2をtim_ti2に配置します。
- tim_ti2fp2は、立上がりエッジを検出して、TIMx_CCERレジスタでCC2P=“0”とCC2NP =“0”を書き込みます。
- tim_ti2fp2をスレーブモードコントローラのトリガ（tim_trgi）として構成するために、TIMx_SMCRレジスタのTS=“00110”を書き込みます。
- tim_ti2fp2を使用してカウンタを開始します。このためには、TIMx_SMCRレジスタのSMSビットに“110”（トリガモード）を書き込みます。

OPM波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに0から1へ遷移し、カウンタが自動再ロード値に達したときに1から0へ遷移する波形を生成するとします。このためには、TIMx_CCMR1レジスタにOC1M=111を書き込むことによって、PWMモード2を有効にする必要があります。オプションで、TIMx_CCMR1レジスタにOC1PE='1' を書き込み、TIMx_CR1レジスタのARPEビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1レジスタに比較値を書き込み、TIMx_ARRレジスタに自動再ロード値を書き込む必要があります。次に、UGビットをセットすることによって更新を生成し、tim_ti2で外部トリガイベントを待つ必要があります。この例では、CC1Pに“0”を書き込みます。

上の例では、TIMx_CR1レジスタのDIRおよびCMSビットはローでなければなりません。

必要なパルスは1つだけなので（シングルモード）、TIMx_CR1レジスタのOPMビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1レジスタのOPMビットが“0”にセットされると、繰り返しモードが選択されます。

特殊ケース：tim_ocx高速イネーブル：

ワンパルスモードでは、tim_tix入力のエッジ検出によって、カウンタを有効にするCENビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRxレジスタのOCxFEビットをセットします。こうすると、tim_ocxref（およびtim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFEは、チャネルがPWM1またはPWM2モードに設定された場合のみ機能します。

54.3.23 再トリガ可能なワンパルスモード

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション54.3.22で説明した再トリガ不可能なワンパルスモードとは、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

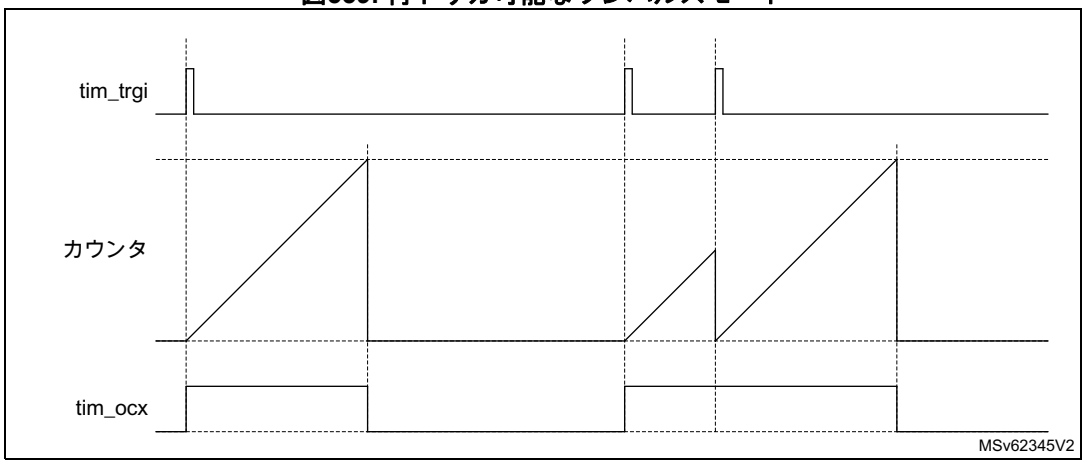
タイマはスレーブモードである必要があり、このときビットはTIMx_SMCRレジスタでSMS[3:0]=“1000”（リセットモードとトリガモードの組合せ）、および再トリガ可能なOPMモード1または2でOCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応するCCRxを0にセットする必要があります（ARRレジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRxはARR以上である必要があります。

注： **OCxM[3:0] およびSMS[3:0]ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

このモードをセンターアラインPWMモードと組合せて使用することはできません。TIMx_CR1では、CMS[1:0] = 00にする必要があります。

図559. 再トリガ可能なワンパルスモード

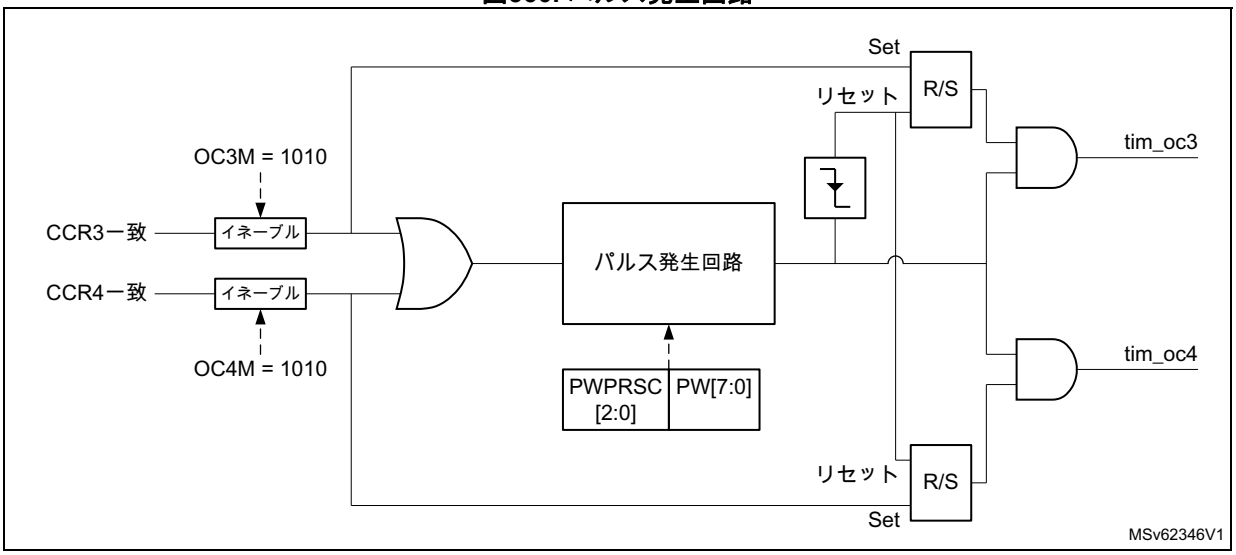


54.3.24 比較モードでのパルス

比較一致イベント時にパルスを生成できます。デバッグまたは同期のために、カウンタ値が特定の比較値と等しいときに、プログラム可能なパルス幅を持つ信号が生成されます。

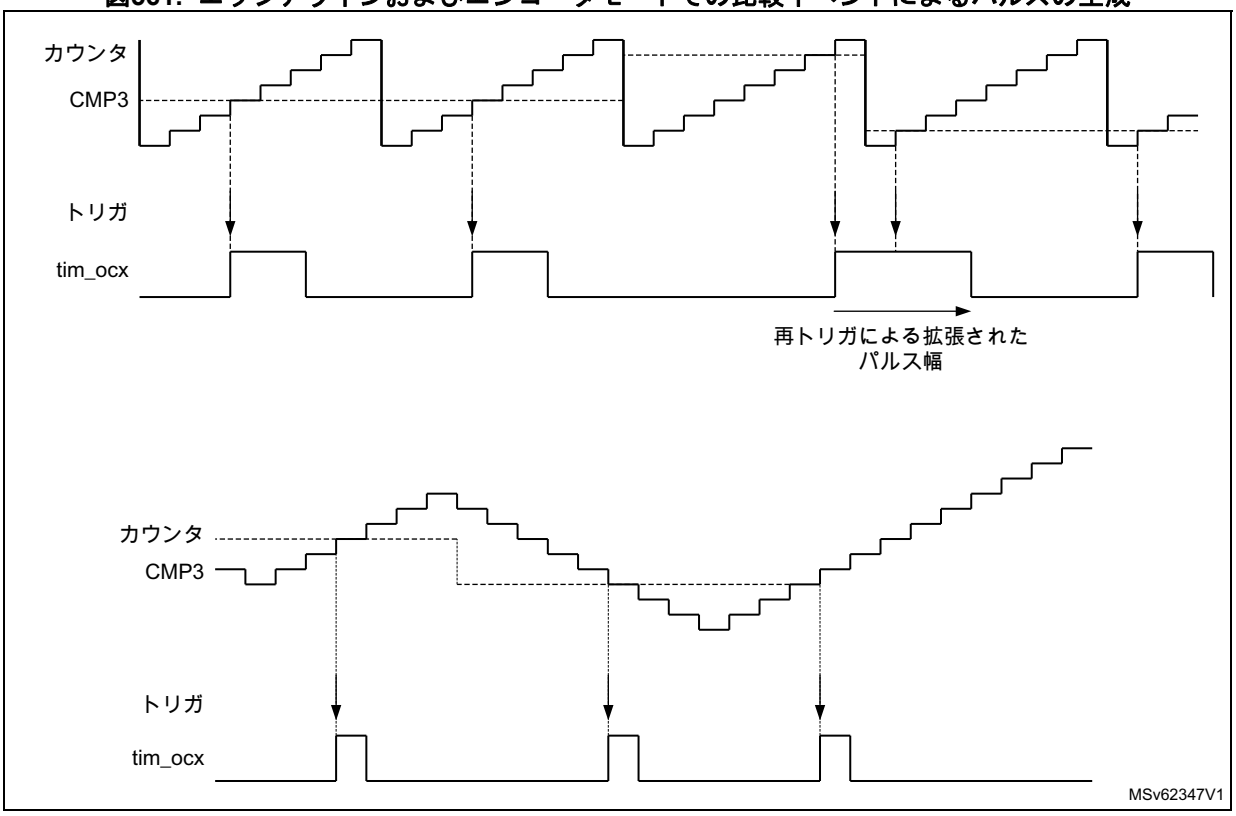
このモードは、エンコーダモードを含むすべてのスレーブモードの選択で、エッジアラインおよびセンタアラインのカウントモードで使用できます。これは、チャネル3とチャネル4のみで使用できます。以下の図560に示すように、パルス発生回路はただ1つであり、2つのチャネルで共有されます。

図560. パルス発生回路



下記の図561は、エッジアラインおよびエンコーダ動作モードでパルスがどのように生成されるかを示しています。

図561. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成



この出力比較モードは、TIMx_CCMR2レジスタのOC3M[3:0] およびOC4M[3:0] ビットフィールドを使用して選択されます。

パルス幅は、次のように、PWPRSC[2:0] ビットに従ってプリスケールされた特定のクロックを使用し、レジスタのPW[7:0] ビットフィールドを使用してプログラムします。

$t_{PW} = PW[7:0] \times t_{PWG}$

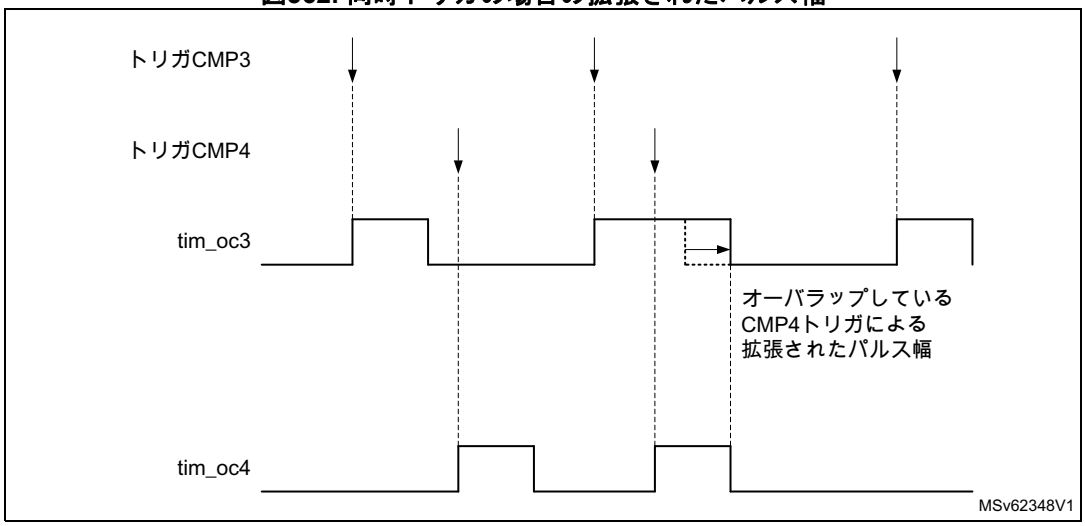
ここで、$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$.

上記は、プリスケーラの値に応じた分解能と最大値を示します。

パルスは再トリガ可能です。パルスが進行中に新しいトリガが入ると、パルスは拡張されます。

注： **2つのチャネルが同時に有効になっている場合、1つのチャネルのトリガが並行出力上に生成されたパルスとオーバーラップしない限り、パルスは独立して発行されます。反対に、2つのトリガがオーバーラップしている場合、最初に到着したトリガに関連するパルス幅は（再トリガのために）拡張されますが、後に到着したトリガのパルス幅は正しくなります（以下の図562に示すとおり）。**

図562. 同時トリガの場合の拡張されたパルス幅



54.3.25 エンコーダインタフェースモード

直交エンコーダ

エンコーダインタフェースモードを選択するには、TIMx_SMCRレジスタで、カウンタがtim_ti1エッジのみをカウントしている場合は SMS=“0001” を、tim_ti2 エッジのみをカウントしている場合はSMS=“0010”を、tim_ti1とtim_ti2の両方のエッジをカウントしている場合はSMS=“0011”を書き込みます。

tim_ti1とtim_ti2の極性を選択するには、TIMx_CCERレジスタのCC1PビットとCC2Pビットをプログラミングします。必要なときには、入力フィルタもプログラミングできます。CC1NPとCC2NPはローに維持する必要があります。

2つの入力tim_ti1とtim_ti2は、直交エンコーダとのインタフェースに使用されます。表543を参照してください。カウンタのクロック供給は、tim_ti1fp1またはtim_ti2fp2（入力フィルタおよび極性選択の後はtim_ti1とtim_ti2、フィルタされず反転されない場合はtim_ti1fp1=tim_ti1、フィルタされず反転されない場合はtim_ti2fp2=tim_ti2）での有効な遷移ごとに行われます。ただし、カウンタは有効である（TIMx_CR1レジスタのCENビットに“1”が書き込まれている）ことが前提です。2つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1レジスタのDIRビットがハードウェアによって変更されます。カウンタがtim_ti1のみ、tim_ti2のみ、またはtim_ti1とtim_ti2の両方をカウントしている場合でも、DIRビットは、いずれかの入力（tim_ti1またはtim_ti2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0とTIMx_ARRレジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0からARRまで、またはARRから0まで）。したがって、開始前にTIMx_ARRを設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、繰返しカウンタ、トリガ出力の機能は通常どおりに機能を続けます。エンコーダモードと外部クロックモード2は互換性がないので、同時に選択することはできません。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（tim_ti1とtim_ti2は同時に切り替わらないと想定しています）。

表543.カウント方向とエンコーダ信号（CC1P = CC2P = 0）

アクティブエッジ	SMS[3:0]	他方の信号のレベル（tim_ti2に対するtim_ti1fp1、tim_ti1に対するtim_ti2fp2）	tim_ti1fp1信号		tim_ti2fp2信号	
			立ち上がり	立下がり	立上がり	立下がり
tim_ti1のみカウント x1モード	1110	高	ダウン	アップ	カウントなし	カウントなし
		低	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti2のみカウント x1モード	1111	高	カウントなし	カウントなし	アップ	ダウン
		低	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti1のみカウント x2モード	0001	高	ダウン	アップ	カウントなし	カウントなし
		低	アップ	ダウン	カウントなし	カウントなし
tim_ti2のみカウント x2モード	0010	高	カウントなし	カウントなし	アップ	ダウン
		低	カウントなし	カウントなし	ダウン	アップ
tim_ti1とtim_ti2の両方をカウント x4モード	0011	高	ダウン	アップ	アップ	ダウン
		低	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCUに直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す3番目のエンコーダ出力は、外部トリガ入力に接続して、カウンタのリセットをトリガできます。

図563 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=“01”（TIMx_CCMR1レジスタ、tim_ti1fp1はtim_ti1に配置）
- CC2S=“01”（TIMx_CCMR2レジスタ、tim_ti1fp2はtim_ti2に配置）
- CC1P=“0”、CC1NP =“0”（TIMx_CCERレジスタ、tim_ti1fp1非反転、tim_ti1fp1=tim_ti1）
- CC2P=“0”、CC2NP =“0”（TIMx_CCERレジスタ、tim_ti1fp2非反転、tim_ti1fp2=tim_ti2）
- SMS=“0011”（TIMx_SMCRレジスタ、両方の入力が立上がりと立下がりの両エッジでアクティブ）
- CEN=“1”（TIMx_CR1レジスタ、カウンタ有効）

図563. エンコーダインタフェースモードにおけるカウンタの動作例

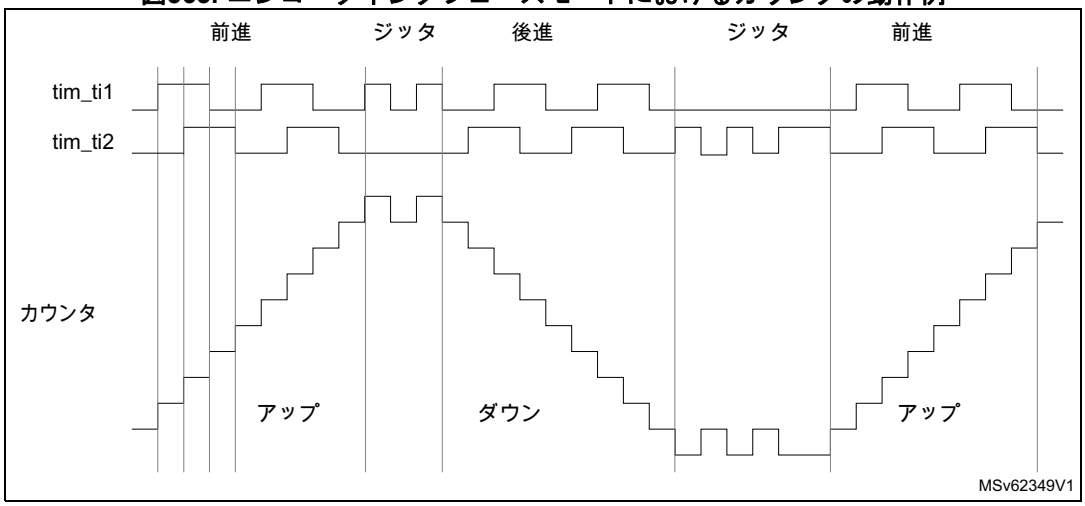
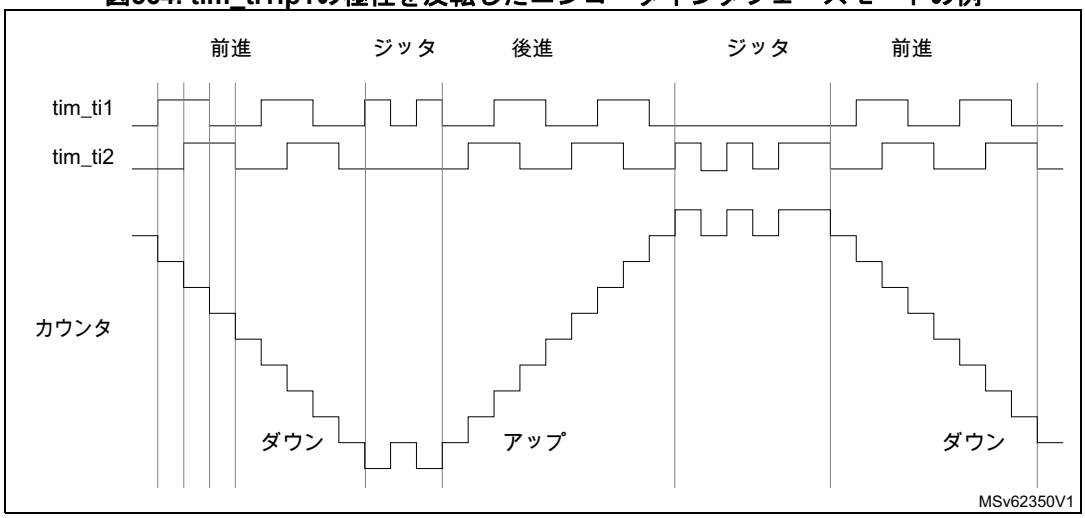


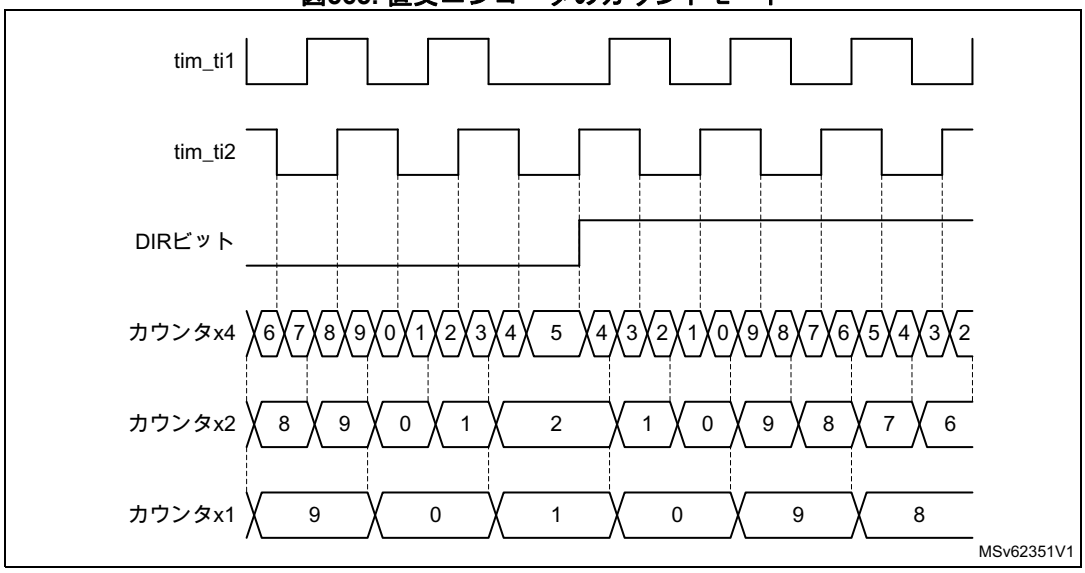
図564に、tim_ti1fp1の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P="1"）。

図564. tim_ti1fp1の極性を反転したエンコーダインタフェースモードの例



以下の図565は、さまざまなカウントモードでの速度反転中のタイマカウンタ値を示しています。

図565. 直交エンコーダのカウントモード



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した2番目のタイマを使用して、2つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合は、カウンタの値を3番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。また、DMAリクエストを通じて、この値を読み出すこともできます。

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPYフラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIFとUIFCPYフラグのアサートの間には、遅延はありません。

32bitのタイマの実装で、IUFREMAPビットがセットされている場合、カウンタのビット31は読出しアクセス時にUIFCPYフラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

クロックプラス方向エンコーダモード

直交エンコーダモードのほか、タイマは他のタイプのエンコーダもサポートしています。

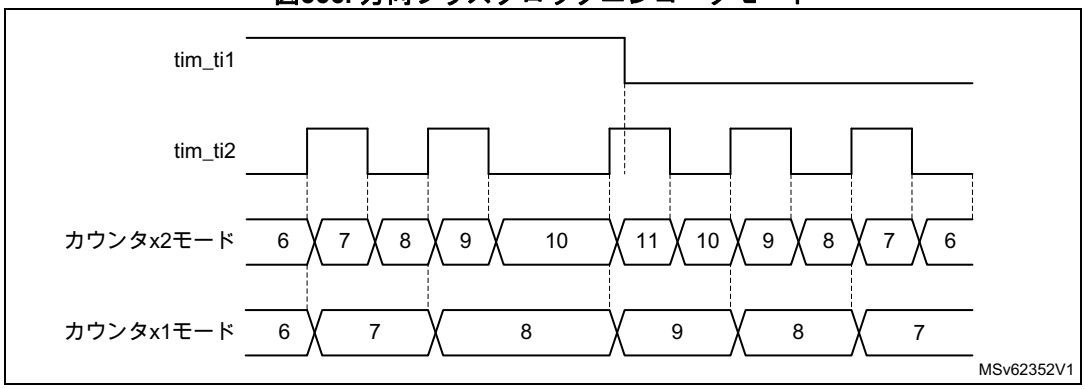
図566に示す「クロックプラス方向」モードでは、tim_ti2の単一ラインでクロックが提供され、tim_ti1入力を使用して方向が強制されます。

このモードはTIMx_SMCRレジスタのSMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1010：X2モード。カウンタはクロックの立上がりと立下がりの両エッジで更新されます。
- 1011：x1 モード。カウンタは CC2P ビット値に従って単一のクロックエッジで更新されます。CC2P = 0は立ち上がりエッジ検出に対応し、CC2P = 1は立ち下がりエッジ検出に対応します。

tim_ti1の方向信号の極性は、CC1Pビットで設定されます。0は正の極性（tim_ti1がハイのときアップカウントで、tim_ti1がローのときダウンカウント）に対応し、CC1P = 1は負の極性（tim_ti1がローのときアップカウント）に対応します。

図566. 方向プラスクロックエンコーダモード



方向クロックエンコーダモード

図567の「方向クロック」モードでは、クロックは2本のラインで提供され、1本のアップカウントクロックラインと1本のダウンカウントクロックラインを持つというように、方向に応じて一度に1つが提供されます。

このモードはTIMx_SMCRレジスタのSMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1100 : x2モード。カウンタは、2つのクロックラインのいずれかの立上がりと立下がりの両エッジで更新されます。CC1PおよびCC2Pビットは、クロックアイドル状態用のコーディングです。CCxP = 0はハイレベルアイドル状態に対応し（下記図567を参照）、CCxP = 1はローレベルアイドル状態に対応します（下記図568を参照）。
- 1101 : x1モード。カウンタは、CC1PおよびCC2Pビット値に従って、単一のクロックエッジで更新されます。CCxP = 0は立下がり検出とハイレベルアイドル状態に対応し（下記図567を参照）、CCxP = 1は立上がり検出とローレベルアイドル状態に対応します（下記図568を参照）。

図567. 方向クロックエンコーダモード（CC1P = CC2P = 0）

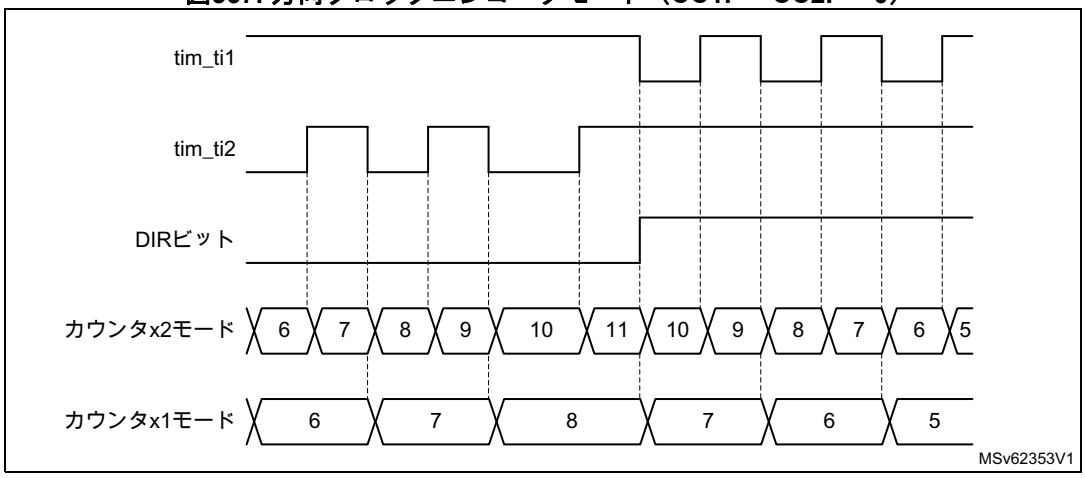
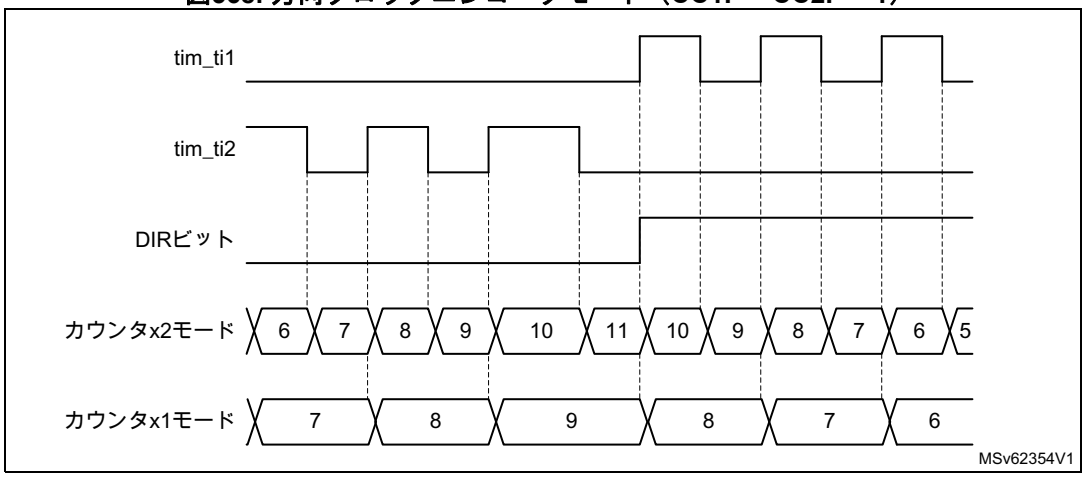


図568. 方向クロックエンコーダモード（CC1P = CC2P = 1）



以下の表 544に、任意の入力遷移に対して方向クロックモードがどのように動作するかを詳しく説明します。

表544.カウント方向 対 エンコーダ信号および極性設定

方向クロックモード	SMS[3:0]	他方の信号のレベル（tim_ti2に対するtim_ti1fp1、tim_ti1に対するtim_ti2fp2）	tim_ti1fp1信号		tim_ti2fp2信号	
			立ち上がり	立下がり	立上がり	立下がり
x2モード CCxP=0	1100	高	ダウン	ダウン	アップ	アップ
		低	カウントなし	カウントなし	カウントなし	カウントなし
x2モード CCxP=1	1100	高	カウントなし	カウントなし	カウントなし	カウントなし
		低	ダウン	ダウン	アップ	アップ
x1モード CCxP=0	1101	高	カウントなし	ダウン	カウントなし	アップ
		低	カウントなし	カウントなし	カウントなし	カウントなし
x1モード CCxP=1	1101	高	カウントなし	カウントなし	カウントなし	カウントなし
		低	ダウン	カウントなし	アップ	カウントなし

インデックス入力

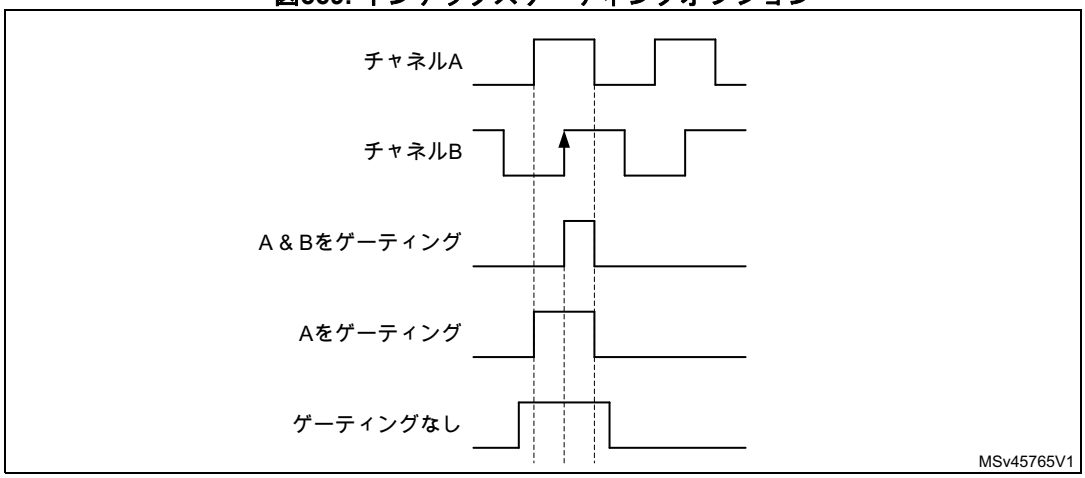
カウンタは、絶対基準位置を示すエンコーダからのインデックス信号によってリセットできます。インデックス信号は、tim_etr_in入力に接続する必要があります。これは、デジタル入力フィルタを使用してフィルタリングできます。

インデックス機能は、TIMX_ECRレジスタのIEビットで有効にされます。IEビットは、SMS[3:0] ビットフィールドが0001、0010、011、1010、1011、1100、1101、1110、1111の値であるときに、エンコーダモードでのみセットする必要があります。

以下の図569に示すように、インデックスパルス調整のいくつかのオプションを備えた市販のエンコーダが提案されています。

- AおよびBでゲート処理：パルス幅は1つのチャネル周期の1/4であり、AおよびBの両方のエッジにそろえられます。
- Aでゲート処理（またはBでゲート処理）：パルス幅は1つのチャネル周期の1/2であり、チャネルA（または チャネルB）の2つのエッジにそろえられます。
- ゲート処理なし：パルス幅は最大1チャネル周期であり、どのエッジにもそろえられません。

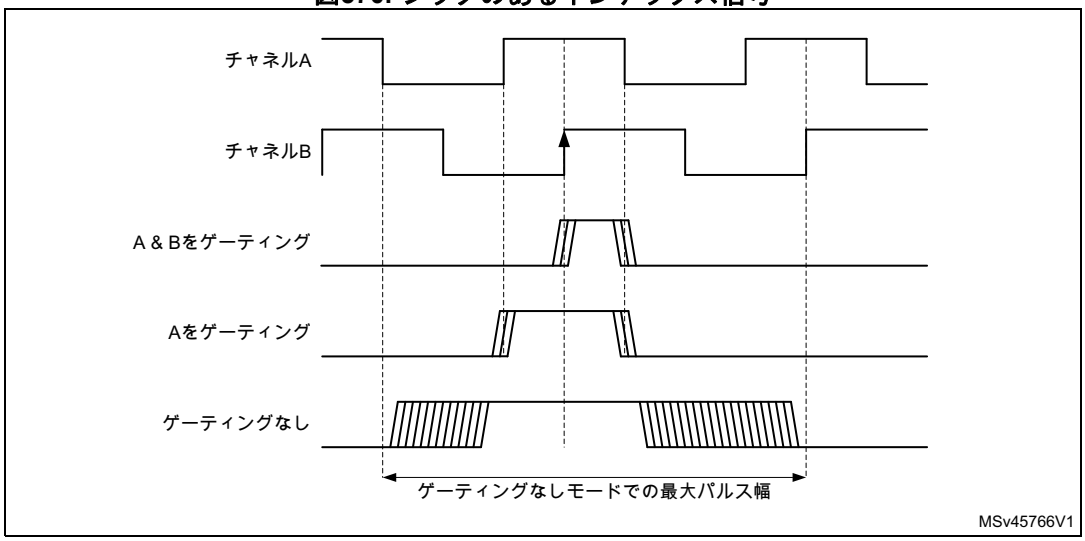
図569. インデックスゲーティングオプション



以下の図570に示すように、ゲーティングモードに関係なく、回路はインデックス信号のジッタを許容します。

ゲート処理なしモードでは、信号はエンコーダの2周期より厳密に小さくなければなりません。パルス幅がエンコーダの2周期以上の場合、カウンタは複数回リセットされます。

図570. ジッタのあるインデックス信号



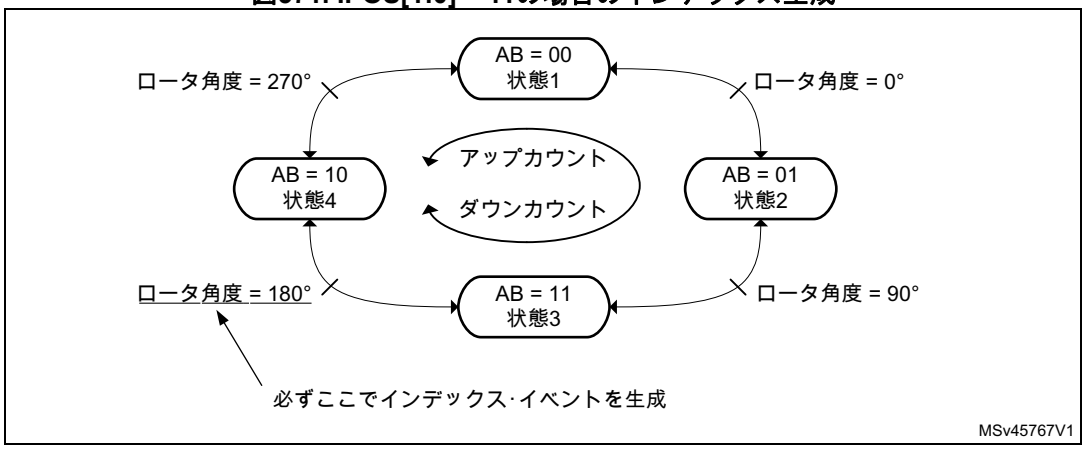
タイマは、特定のプログラミングを必要とせずに、3つのゲーティングオプションを同様にサポートします。TIMx_ECRレジスタのIPOS[1:0] ビットフィールドを使用して、インデックスを同期する必要があるエンコーダ状態（チャネルAとチャネルBの状態の組み合わせ）を定義するだけです。

インデックス検出イベントは、速度の反転中に対称動作を確保するために、カウント方向に応じて異なる動作をします。

- カウンタはアップカウント中（DIRビット = 0）に、リセットされます。
- カウンタはダウンカウント時に、TIMx_ARRに設定されます。

これにより、カウント方向に関係なく、まったく同じ機械的角度位置でインデックスを生成できます。以下の図571は、単純な例（機械的回転ごとに4つのエッジを提供するエンコーダ）について、生成されるインデックスの位置を示しています。

図571. IPOS[1:0] = 11の場合のインデックス生成



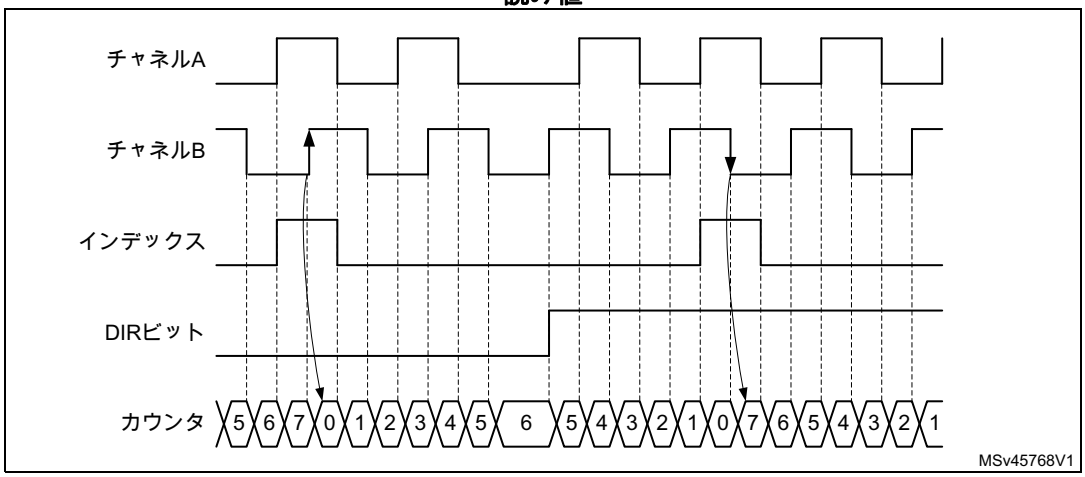
以下の図572に、IPOS[1:0] = 11の場合の波形と対応する値を示します。カウンタの値が強制される瞬間は、カウント方向に応じて自動的に調整されることを示しています。

- アップカウント（DIRビット= 0）の場合、エンコーダの状態が「11」（ChA = 1、ChB = 1）のとき、カウンタは0にセットされます。
- ダウンカウント（DIRビット= 1）の場合、「11」状態を終了するとき、カウンタはTIMx_ARRにセットされます。

インデックス検出イベント時に割込みを発行できます。

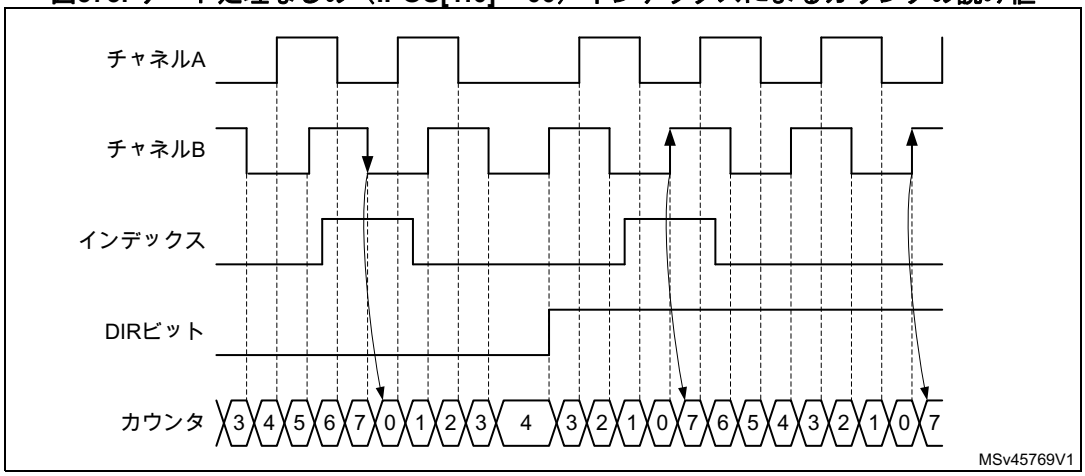
矢印は、どの遷移でインデックスイベント割込みが生成されるかを示しています。

図572. チャネルAでゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値



以下の図573に、ゲート処理なしモードの場合の波形と対応する値を示します。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図573. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値



以下の図574は、さまざまなパルスアライメントシナリオでの「A & Bでゲート処理」モードの処理方法を示しています。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図574. チャネルAおよびBでゲート処理されたインデックスによるカウンタの読み値

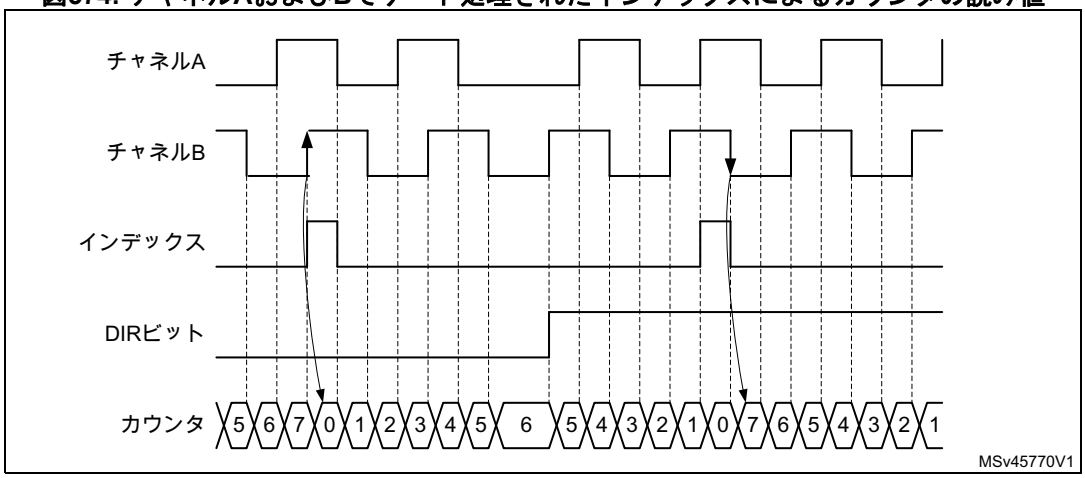


図575と図576には、後続のインデックスパルスが、エンコーダクロック周期の4分の1よりも狭いことがあるケースが詳しく示されています。

図575. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11）

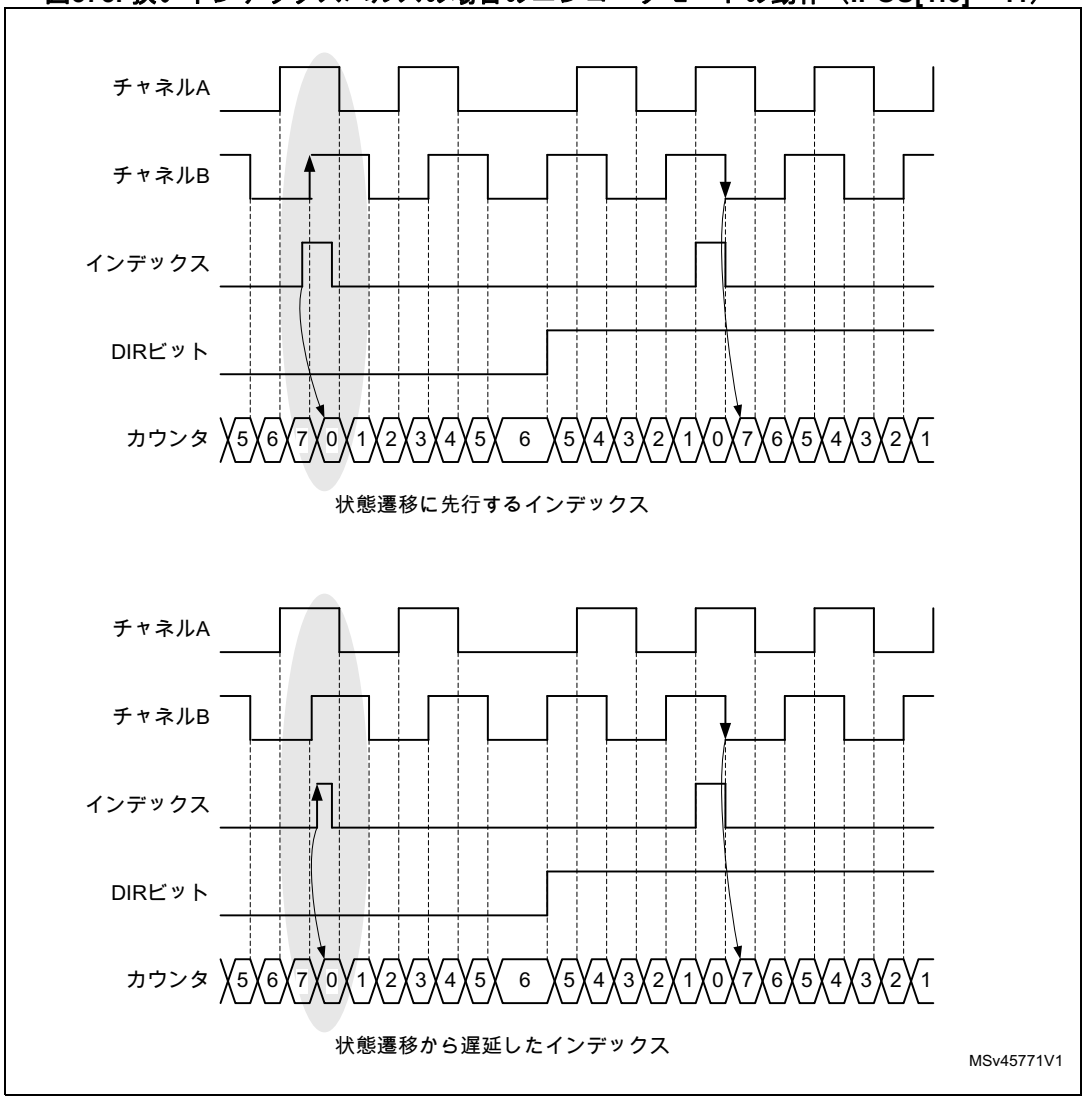
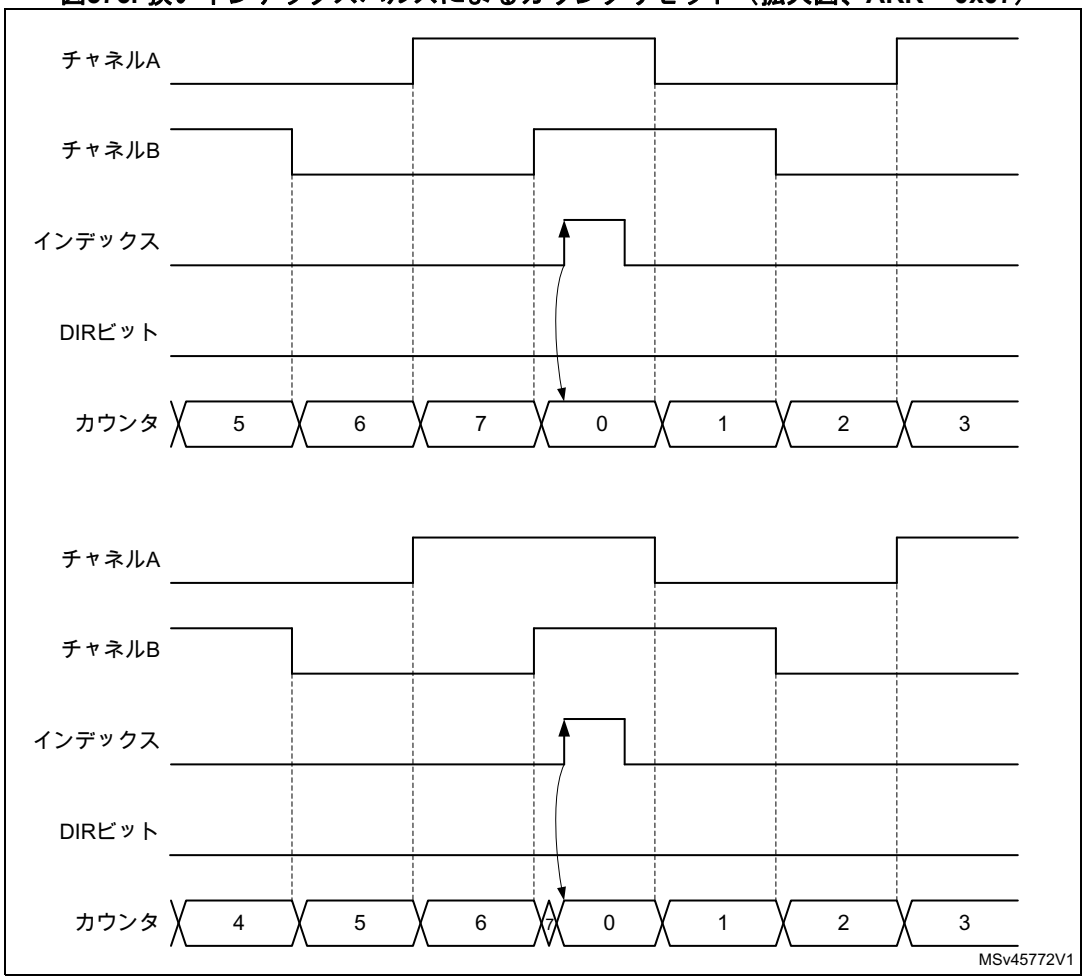
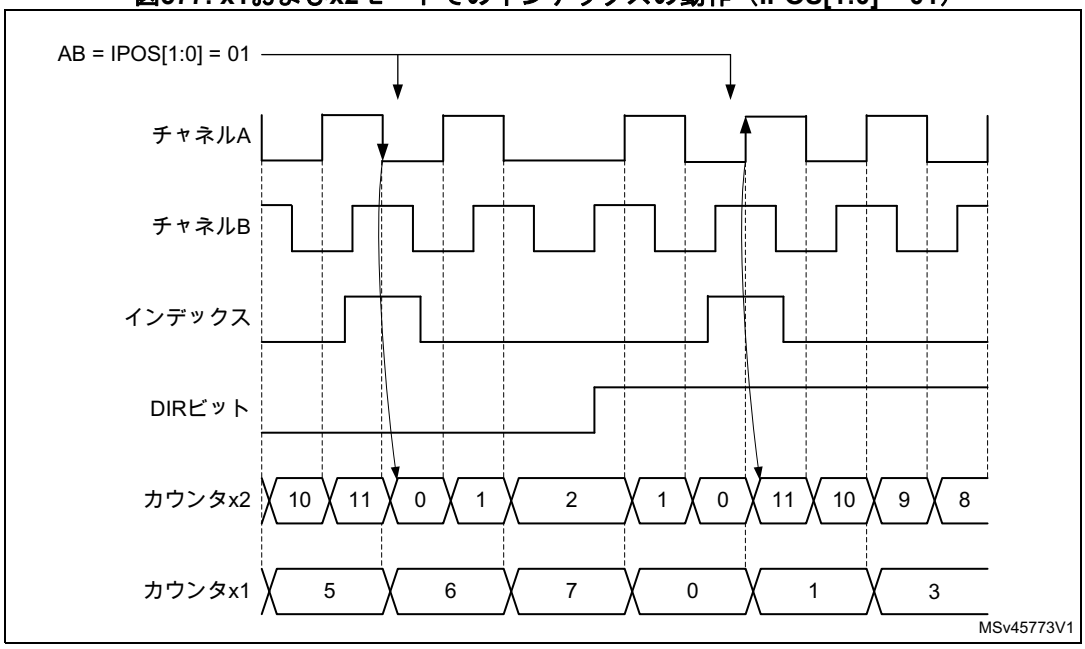


図576. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07）



下記の図577は、x1およびx2モードでインデックスがどのように管理されるかを示します。

図577. x1およびx2モードでのインデックスの動作（IPOS[1:0] = 01）

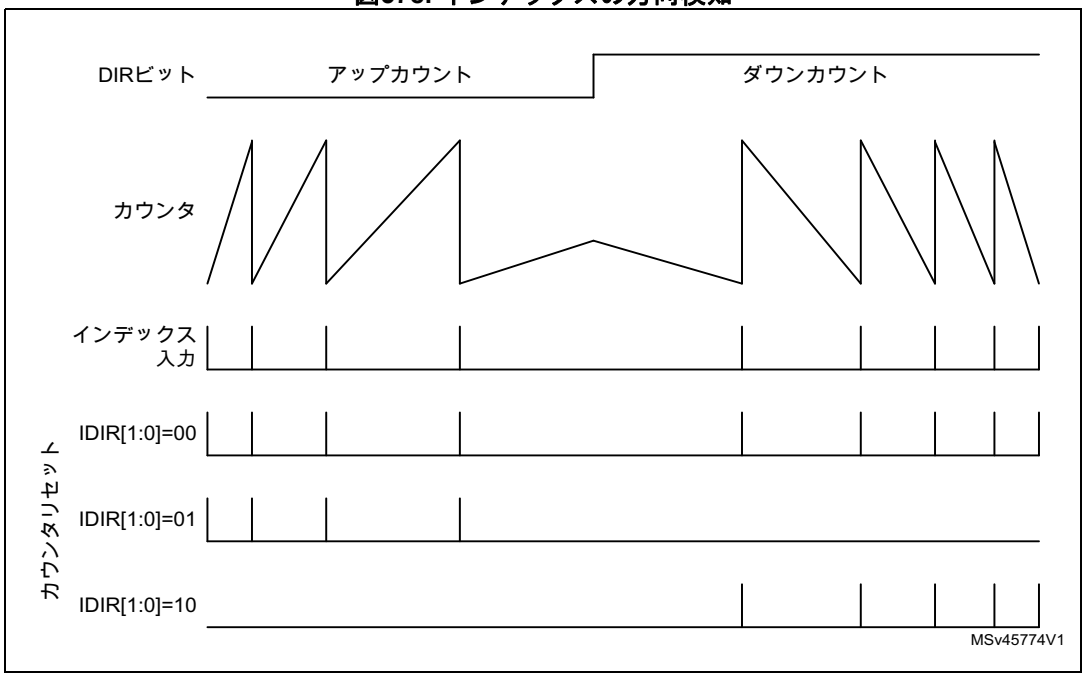


インデックスの方向検知

TIMx_ECRレジスタのIDIR[1:0] ビットフィールドにより、選択したカウント方向でのみインデックスをアクティブにできます。

以下の図578は、IDIR[1:0] 値に応じた、インデックスとカウンタのリセットイベントの関係を示しています。

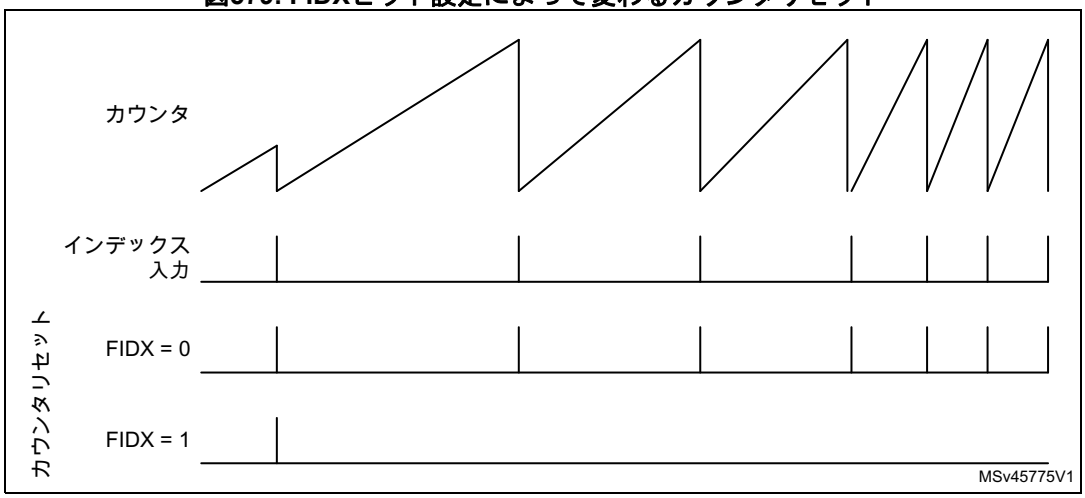
図578. インデックスの方向検知



特殊な最初のインデックスイベント管理

TIMx_ECRレジスタのFIDXビットによって、以下の図579に示すように、インデックスを1回だけ取得できます。最初のインデックスが到着すると、その後のインデックスは無視されます。必要に応じて、FIDXビット に0を書き込み、再度1にセットすることにより、回路を再設定できます。

図579. FIDXビット設定によって変わるカウンタリセット



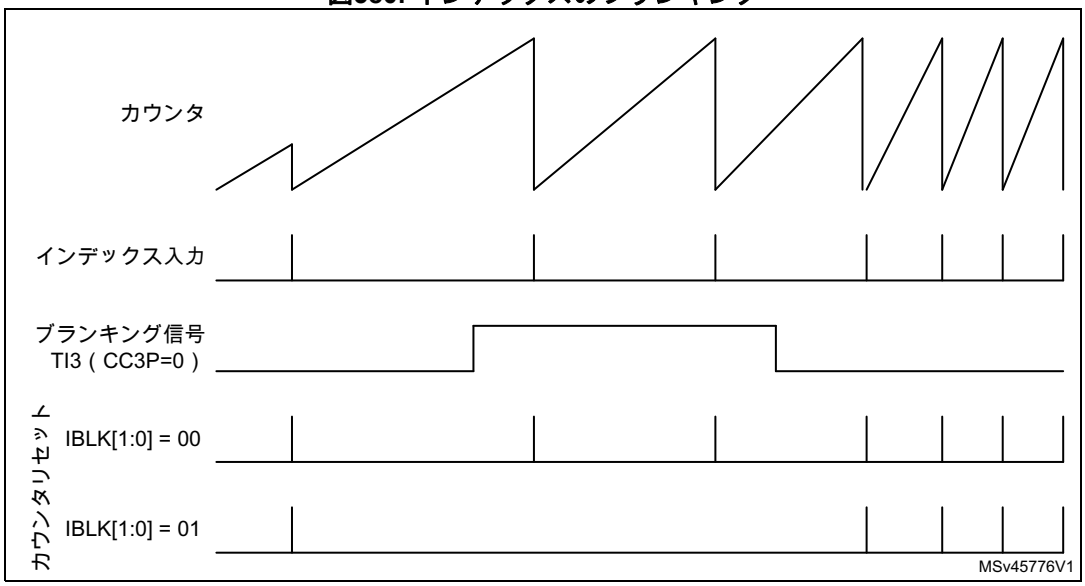
インデックスのブランキング

インデックスイベントは、tim_ti3またはtim_ti4の入力を使用してブランキングすることができます。ブランキング期間中は、以下の図580に示すように、インデックスイベントがカウンタをリセットすることはありません。

このモードはTIMx_ECRレジスタのIBLK[1:0] ビットフィールドを使用して、以下のように有効にされます。

- IBLK[1:0] = 00：インデックス信号は常にアクティブです。
- IBLK[1:0] = 01：インデックス信号は、tim_ti3入力時にブランキング処理されます。
- IBLK[1:0] = 10：インデックス信号は、tim_ti4入力時にブランキング処理されます。

図580. インデックスのブランキング



非直交モードでのインデックス管理

以下の図581および図582では、SMS[3:0] ビットフィールドが1010、1011、1100、1101に等しい場合に、方向クロックモードおよびクロックプラス方向モードでインデックスがどのように管理されるかを詳しく説明します。

これらのモードの両方で、インデックスの検出は次のようにIPOS[0] ビットで設定されます。

- IPOS[0] = 0：インデックスはクロックのローレベルで検出されます。
- IPOS[0] = 1：インデックスはクロックのハイレベルで検出されます。

IPOS[1] ビットは意味がありません。

図581. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1）

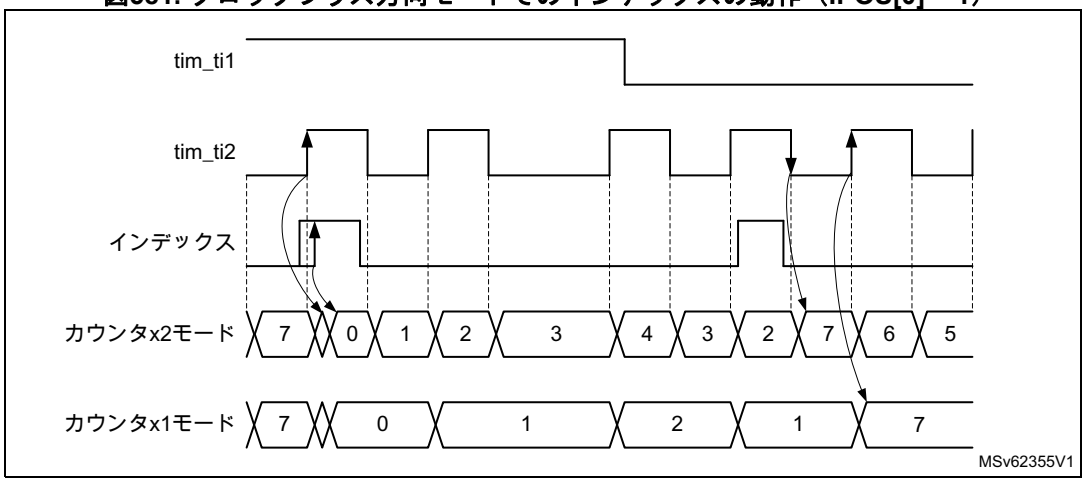
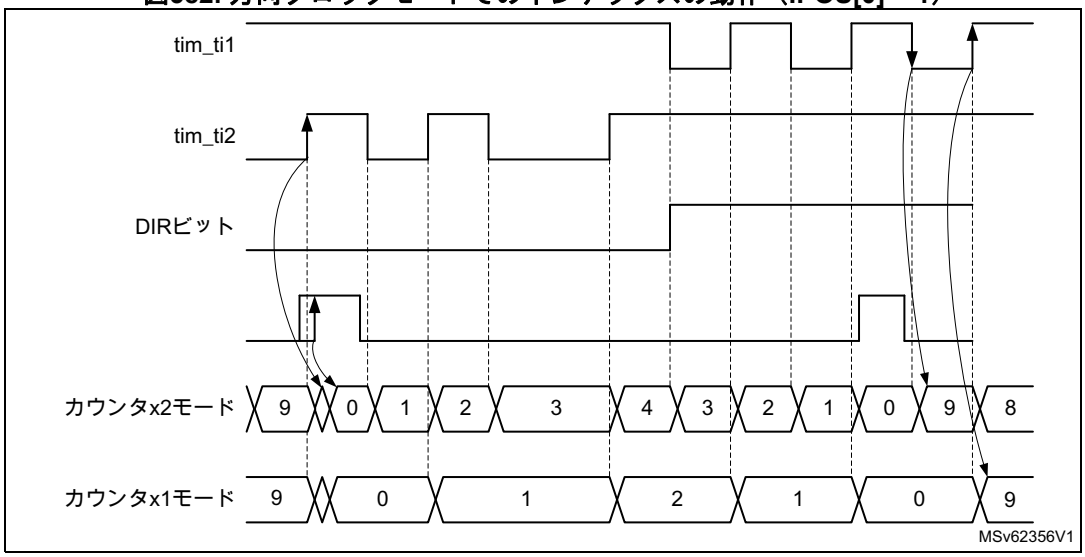


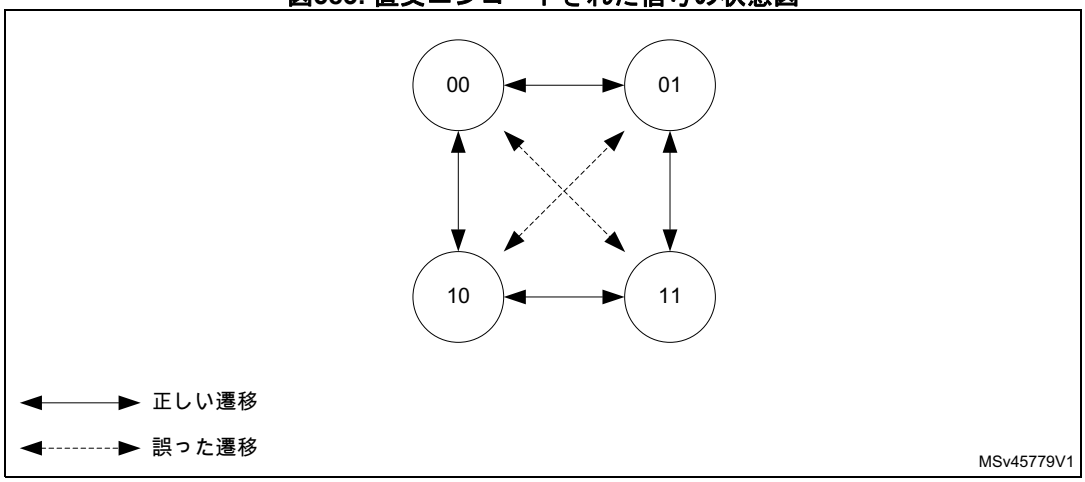
図582. 方向クロックモードでのインデックスの動作（IPOS[0] = 1）



エンコーダのエラー管理

2つの直交信号が使用できるエンコーダ構成の場合、遷移エラーを検出できます。2つの入力の読み値は、以下の図583で、状態図として表すことができる2bitのグレーコードに対応しています。一度に1つのビットだけが変化することが予測されています。間違った遷移があると、TIMx_SRステータスレジスタのTERRF割込みフラグがセットされます。TIMx_DIERレジスタのTERRIEビットがセットされていると、遷移エラー割込みが生成されます。

図583. 直交エンコードされた信号の状態図



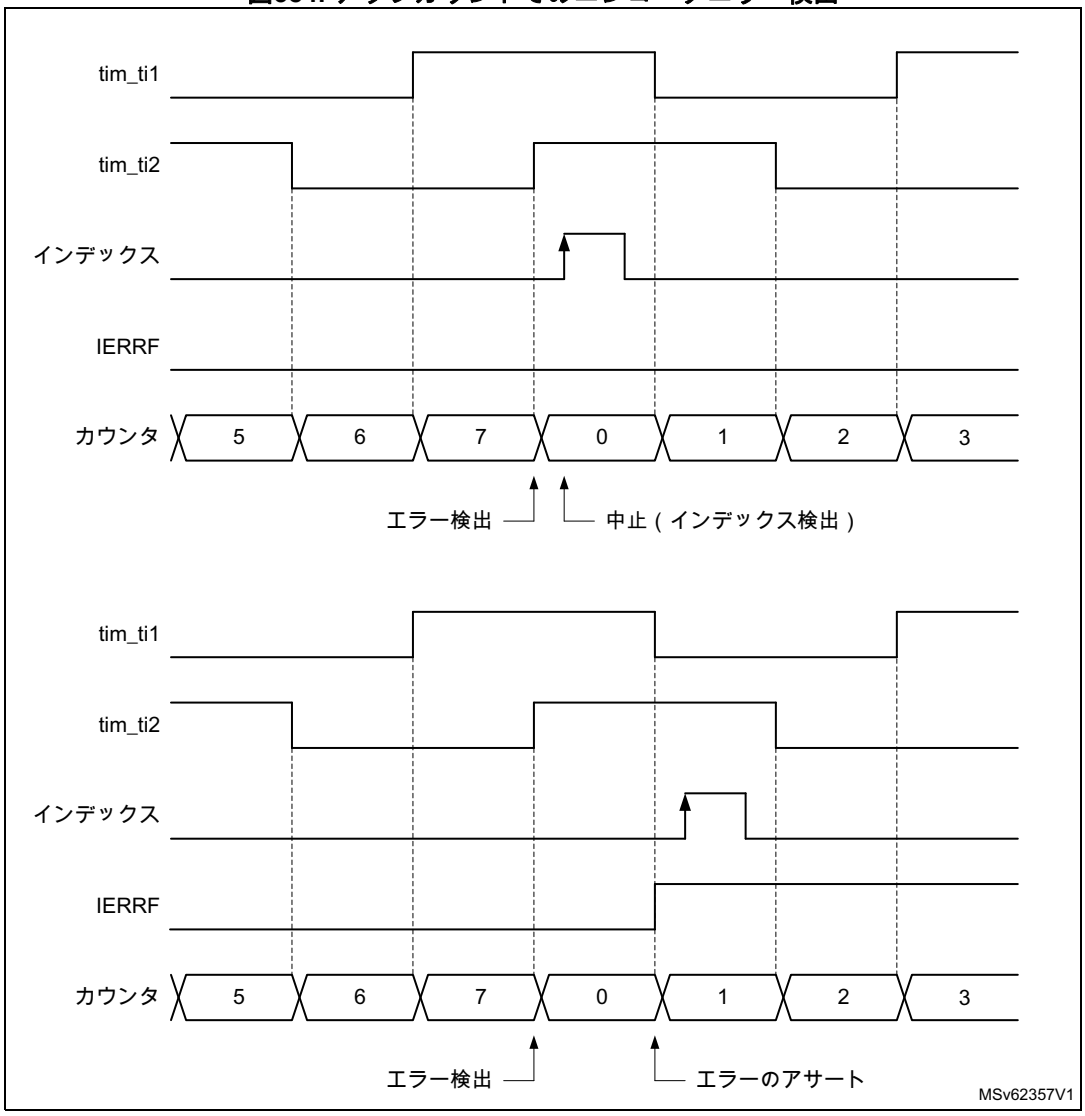
インデックス信号を持つエンコーダの場合、1回転あたりのパルスが過剰になる異常動作を検出できます。1回転あたりN個のパルスを持つエンコーダでは、1回転あたり4xNカウントが提供されます。インデックス信号によって、4xNクロック周期ごとにカウンタがリセットされます。

カウンタ値がインデックスイベントなしにTIMx_ARRから0にインクリメントされるか、0からTIMxARR値にデクリメントされる場合、これはインデックス位置エラーとして報告されます。

オーバーフロー閾値は、TIMx_ARRレジスタを使ってプログラムします。1000行のエンコーダでは、カウンタ値が0?3999になります（4x読み取りモードの場合）。オーバーフロー検出閾値は、TIMx_ARR = 3999 + 1 = 4000を設定してプログラムする必要があります。

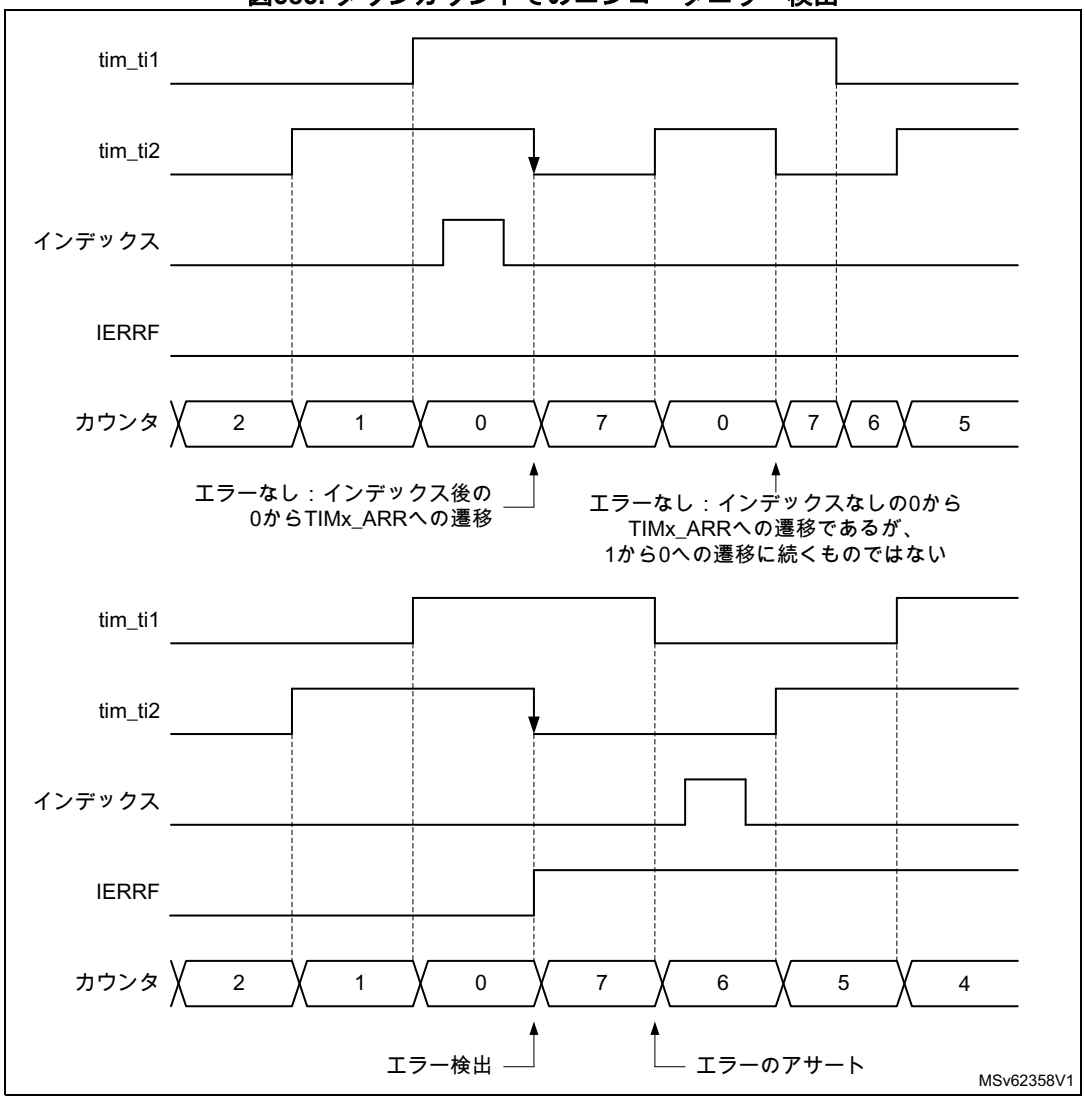
アップカウントでは、エラーアサーションは0から1への遷移まで遅延されます。これは、以下の図584に示すように、AおよびBゲート処理モードでの狭いインデックスパルスへの対処です。

図584. アップカウントでのエンコーダエラー検出



ダウンカウントモードでは、検出は1から0への事前遷移によって条件付けられます。これは、以下の図585に示すように、AおよびBゲート処理モードでの狭いインデックスパルスに対処し、インデックス検出直後にエンコーダがTIMx_ARRと0の間でディザリングする場合の誤ったエラー検出を回避するためです。

図585. ダウンカウントでのエンコーダエラー検出



インデックスエラーによって、TIMx_SRステータスレジスタのIERRF割込みフラグがセットされます。TIMx_DIERレジスタのIERRIEビットがセットされていると、インデックスエラー割込みが生成されます。

エンコーダの機能的割込み

エンコーダモードでは、以下の割込みも使用できます。

- 方向変更：エンコーダモードでのカウント方向の変更によって、TIMx_CR1レジスタのDIRビットが反転します。方向変更によって、TIMx_SRステータスレジスタのDIRF割込みフラグがセットされます。TIMx_DIERレジスタのDIRIEビットがセットされていると、方向変更割込みが生成されます。

- インデックスイベント：インデックスイベントによって、TIMx_SRステータスレジスタのIDXF割込みフラグがセットされます。TIMx_DIERレジスタのIDXIEビットがセットされていると、インデックス割込みが生成されます。

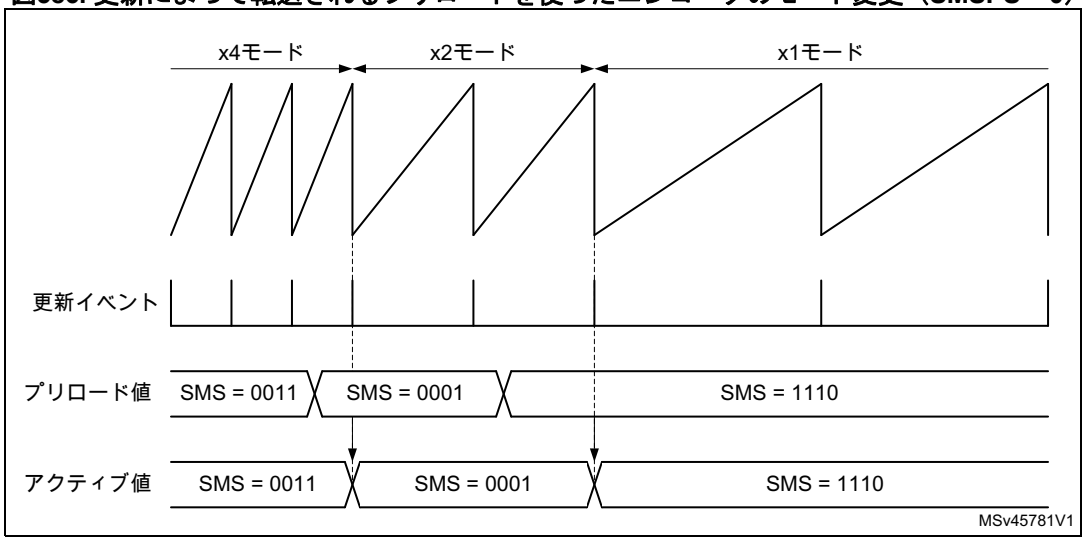
実行時エンコーダモード更新のためのスレーブモード選択プリロード

実行中に、あるエンコーダモードを別のモードに切り替える必要があることがあります。以下の図586に示すように、これは通常、高速時に更新割込みレートを低下させるために、x4モードからx2に、x1に切り替えることにより、実行されます。

この目的のために、SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCRレジスタのSMSPEイネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、TIMx_SMCRレジスタのSMSPSビットで選択できます。

- SMSPS = 0：転送は、アップカウント時にカウンタがオーバーフローするとき、およびダウンカウント時にアンダーフローするときに発生する更新イベント（UEV）によってトリガされます。
- SMSPS = 1：転送は、インデックスイベントによってトリガされます。

図586. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）



エンコーダクロック出力

エンコーダモードの動作原理は、十分な数のクロックエッジ取得と正確な測定を行うために比較的長い積分時間を必要とするため、低速においては、高分解能の速度測定には完全には適していません。

低速でのより良いソリューションは、エッジ間のクロック周期測定を行うことです。これは、スレーブタイマを使用して達成できます。タイマによって、tim_trgo出力にエンコーダクロック情報を出力できます。その後、スレーブタイマはエンコーダクロックエッジごとに周期測定を実行し、速度情報を提供できます。

このモードはTIMx_CR2レジスタのMMS[3:0] ビットフィールドを1000に設定することによって有効になります。これは、SMS[3:0] 値、0001、0010、0011、1010、1011、1100、1101、1110、1111に対して有効です。その他のSMS[3:0] コードは許されず、予想外の動作につながることがあります。

54.3.26 方向ビットの出力

tim_oc3nおよびtim_oc4出力信号に、タイマから方向信号を出力できます（TIMx_CR1レジスタのDIRビットのコピー）。これを達成するには、TIMx_CCMR2レジスタのOC3M[3:0] またはOC4M[3:0] ビットフィールドを1011にセットします。

この機能を使用すると、エンコーダモードでカウント方向（または回転方向）を監視すること、またはセンタアラインPWMモードでアップ／ダウンフェーズを示す信号を得ることができます。

54.3.27 UIFビットの再配置

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPYフラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特殊なケースでは、バックグラウンドタスク（カウンタの読出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

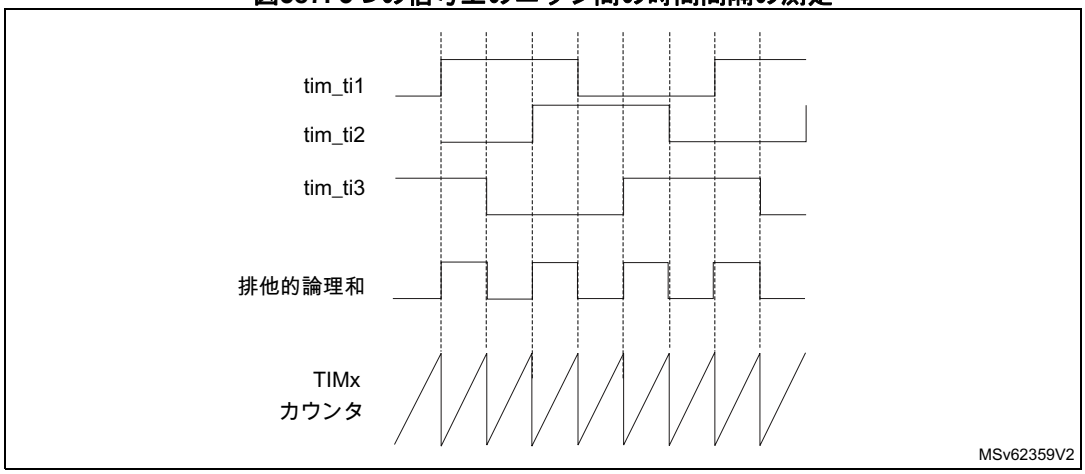
UIFとUIFCPYフラグのアサートの間には、遅延はありません。

54.3.28 タイマ入力XOR機能

TIMx_CR2レジスタのTI1Sビットによって、チャネル1の入力フィルタを3つの入力ピンtim_ti1、tim_ti2、およびtim_ti3を結合するXORゲートの出力に接続できます。

XOR出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図587に示すように、2つの入力信号上のエッジ間の間隔を測定するのに便利です。

図587. 3つの信号上のエッジ間の時間間隔の測定



54.3.29 ホールセンサとのインタフェース

これは、高機能制御タイマを使用してPWM信号を生成し、モータと図588で「インタフェースタイマ」と記されている別のタイマTIMxを駆動することによって実現します。「インタフェースタイマ」は、XORを通じてtim_ti1入力チャネル（TIMx_CR2レジスタのTI1Sビットをセットすることで選択できます）に接続された3つのタイマ入力ピン（tim_ti1、tim_ti2、およびtim_ti3）をキャプチャします。

スレーブモードコントローラはリセットモードに設定され、スレーブ入力はtim_ti1f_edです。したがって、3つの入力のいずれかが反転するごとに、カウンタは0からカウントをリスタートします。これが、ホール入力の変化によってトリガされるタイムベースとなります。

「インタフェースタイマ」上で、キャプチャ／比較チャネル1がキャプチャモードで設定され、キャプチャ信号はtim_trcです（図 530：2099ページのキャプチャ／比較チャネル（例：チャネル1入力ステージ）を参照）。キャプチャされた値は、入力の2回の変化の間の経過時間に対応し、モータの速度情報を与えます。

「インタフェースタイマ」を出力モードで使用して、（COMイベントをトリガすることによって）高機能制御タイマのチャネルの設定を変更するパルスを生成できます。高機能制御タイマは、モータを駆動するPWM信号を生成するために使用されます。このためには、プログラミングした遅延の後に正パルスが生成されるように（出力比較モードまたはPWMモードで）インタフェースタイマチャネルをプログラミングする必要があります。このパルスは、tim_trgo出力を通じて高機能制御タイマに送信されます。

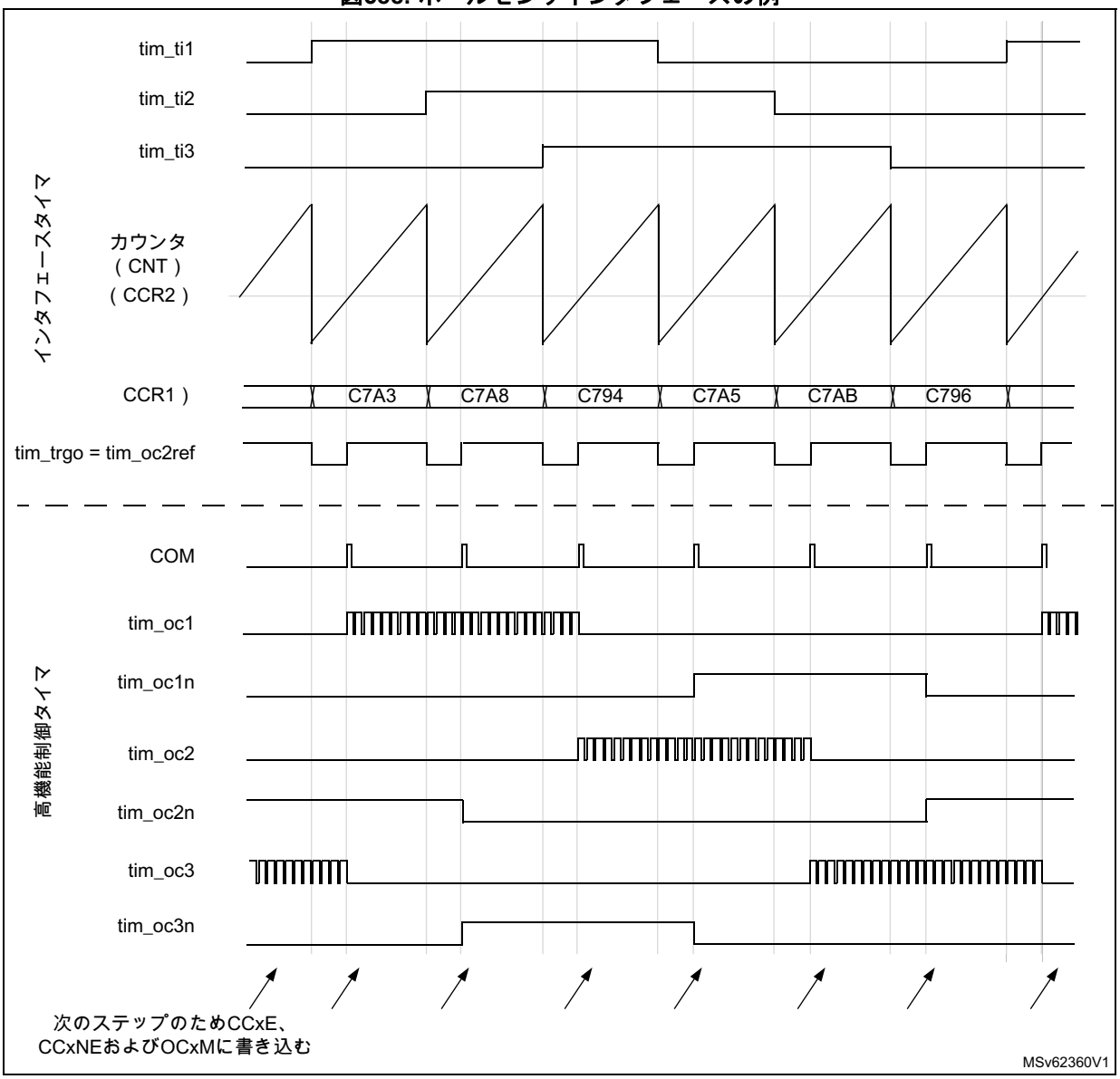
例：TIMxタイマの1つに接続されているホール入力が変化するたびに、プログラミングした遅延の後に高機能制御タイマのPWM設定を変更するとします。

- 3つのタイマ入力をtim_ti1入力チャネルにOR接続します。このためには、TIMx_CR2レジスタのTI1Sビットに“1”を書き込みます。
- タイムベースをプログラムします。このためには、TIMx_ARRに最大値を書き込み、tim_ti1の変化でカウンタがクリアされるようにします。最大カウンタ時間がセンサの2回の変化の間の時間より長くなるように、プリスケーラを設定します。
- チャネル1をキャプチャモード（tim_trc選択）にプログラムします。すなわち、TIMx_CCMR1レジスタのCC1Sビットに“01”を書き込みます。必要な場合は、デジタルフィルタをプログラムすることもできます。
- チャネル2をPWM 2モードにプログラミングし、希望の遅延を指定します。このためには、TIMx_CCMR1レジスタのOC2Mビットに“111”を、CC2Sビットに“00”を書き込みます。
- tim_trgo上のトリガ出力としてtim_oc2refを選択します。このためには、TIMx_CR2 レジスタのMMSビットに“101”を書き込みます。

高機能制御タイマで、トリガ入力として適切なtim_itrx入力を選択する必要があり、タイマがPWM信号を生成するようにプログラミングし、キャプチャ/比較制御信号がプリロードされ（TIMx_CR2レジスタのCCPC=1）、COMイベントがトリガ入力によって制御されなければなりません（TIMx_CR2レジスタのCCUS=1）。PWM制御ビット（CCxE、OCxM）は、COMイベント後に次のステップのために書き込まれます（これは、tim_oc2refの立上がりエッジによって生成される割込みサブルーチンで行うことができます）。

図588に、この例を示します。

図588. ホールセンサインタフェースの例



54.3.30 タイマの同期

タイマの同期や連携した動作のために、TIMxタイマを内部で相互リンクすることができます。詳細については、セクション 55.4.23 : タイマの同期を参照してください。これらは、リセットモード、ゲートモード、トリガモード、リセット + トリガモード、およびゲート + リセットモードで同期できます。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1レジスタのURSビットがローの場合は、更新イベントUEVが生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

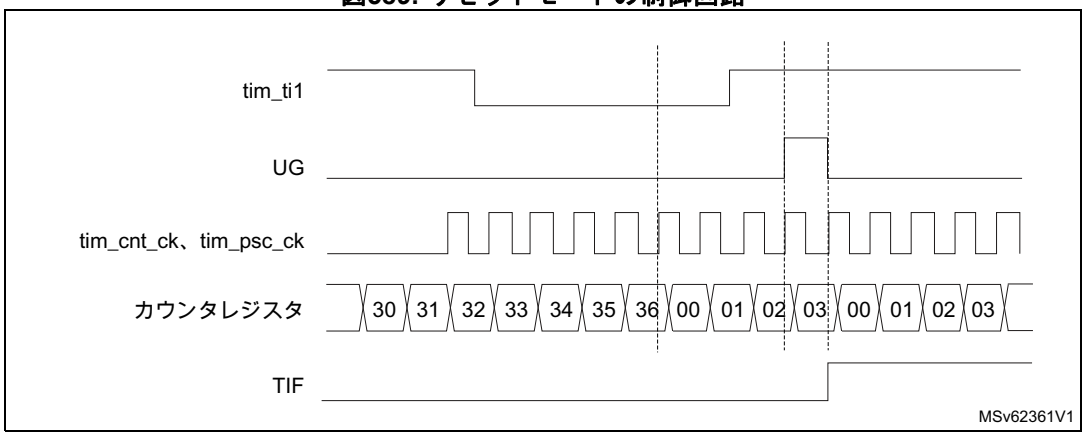
次の例では、tim_ti1入力の立上がりエッジに応じて、アップカウンタがクリアされます。

- tim_ti1の立上がりエッジを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01）。TIMx_CCERレジスタにCC1P=0とCC1NP = 0を書き込んで、極性を有効にします（そして、立上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
- TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1の立上がりエッジまで通常の動作を行います。tim_ti1が立ち上がると、カウンタはクリアされ、0からリスタートします。同時に、トリガフラグがセットされ（TIMx_SRレジスタのTIFビット）、有効な場合は割込みリクエストまたはDMAリクエストを送信できます（TIMx_DIERレジスタのTIEおよびTDEビットに依存）。

次の図は、自動再ロードレジスタTIMx_ARR=0x36の場合の動作を示します。tim_ti1の立上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1入力の同期回路によるものです。

図589. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

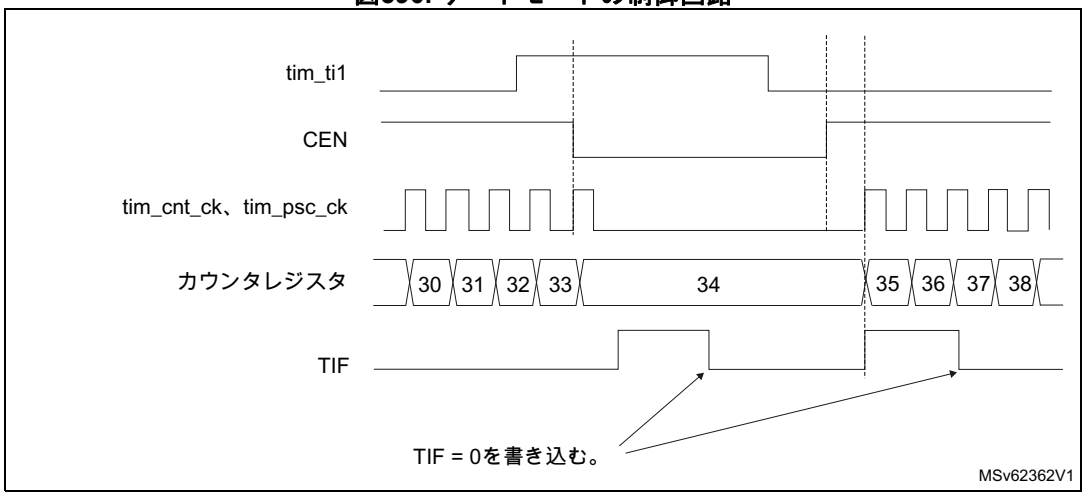
次の例では、アップカウンタはtim_ti1入力がローのときだけカウントします。

- tim_ti1のローレベルを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01bit）。TIMx_CCERレジスタでCC1P= 1とCC1NP = 0を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタにSMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1がローになると内部クロックでカウントを開始して、tim_ti1がハイになると停止します。TIMx_SRレジスタのTIFフラグは、カウンタの開始時と停止時にセットされます。

tim_ti1の立上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1入力の再同期回路によるものです。

図590. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

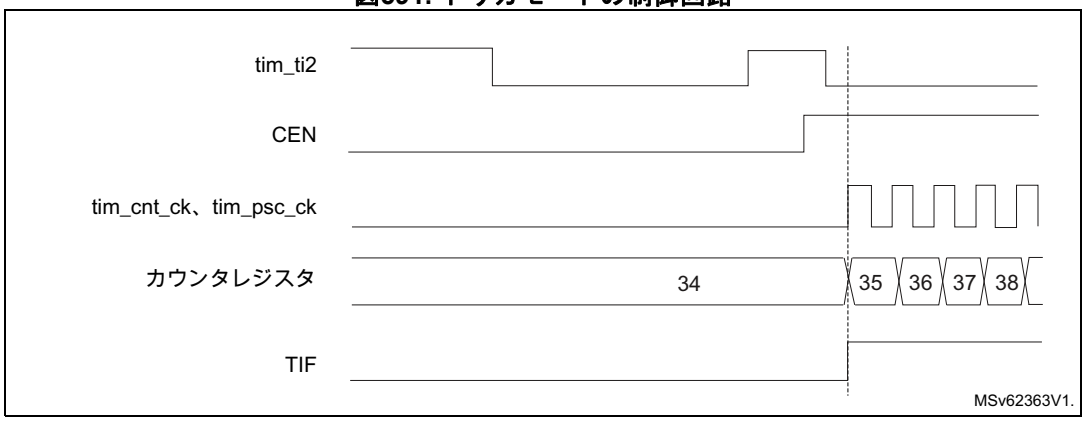
次の例では、アップカウンタは、tim_ti2入力の立上がりエッジに応じて開始します。

- tim_ti2の立上がりエッジを検出するように、チャネル2を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2Sビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1レジスタのCC2S=01）。TIMx_CCERレジスタでCC2P = 1とCC2NP = 0を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタにSMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCRレジスタにTS=00110を書き込むことによって、入力ソースとしてtim_ti2を選択します。

tim_ti2で立上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIFフラグがセットされます。

tim_ti2の立上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2入力の再同期回路によるものです。

図591. トリガモードの制御回路



スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（tim_trgi）の立上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード - ゲートモードとリセットモードの組み合わせ

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードは、範囲外のPWM信号（最大期待値を超えるデューティサイクル）を検出するために使用されます。

スレーブモード：外部クロックモード2 + トリガモード

外部クロックモード2は、他のスレーブモードとともに使用できます（ただし、外部クロックモード1とエンコーダモードは除きます）。この場合、tim_etr_in信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、またはトリガモード）。TIMx_SMCRレジスタのTSビットを通じてtim_trgiとしてtim_etr_inを選択しないようにしてください。

次の例では、アップカウンタは、tim_ti1の立上がりエッジが発生すると、tim_etr_in信号の立上がりエッジのたびにインクリメントされます。

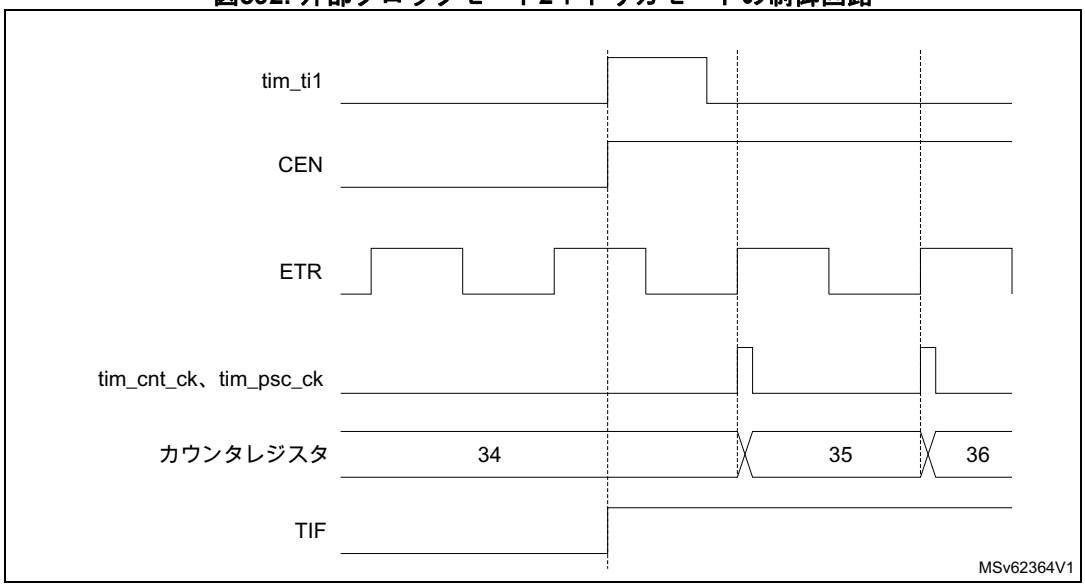
1. TIMx_SMCRレジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0 : tim_etr_inの立上がりエッジを検出。ECE=1で外部クロックモード2を有効にします。

2. TI1の立上がりエッジを検出するように、チャネル1を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1レジスタのCC1S=01で、入力キャプチャソースのみを選択します。
   - TIMx_CCERレジスタのCC1P=0とCC1NP=0で、極性を有効にします（そして、立上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。

tim_ti1の立上がりエッジでカウンタが有効になり、TIFフラグがセットされます。カウンタは、その後tim_etr_inの立上がりエッジでカウントします。

tim_etr_in信号の立上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_etrp入力の再同期回路によるものです。

図592. 外部クロックモード2＋トリガモードの制御回路



注： **tim_trgoまたはtim_trgo2の信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

54.3.31 ADCトリガ

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。以下のような内部エッジ検出から発行されたパルスを生成することもできます。

- OC4refの立上がりおよび立下がりエッジ
- OC5refの立上がりエッジまたはOC6refの立下がりエッジ

トリガは、ADCにリダイレクトされるtim_trgo2内部ラインで発行されます。使用できるイベントは全部で16個あり、TIMx_CR2レジスタのMMS2[3:0] ビットを使用して選択できます。

3相モータ駆動のアプリケーションの例については、2117 ページの図 545を参照してください。

注： **tim_trgoまたはtim_trgo2の信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

注： **ADCのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。**

54.3.32 DMAバーストモード

TIMxタイマには、1つのイベントで多重DMAリクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMAコントローラの転送先は一意で、仮想レジスタTIMx_DMARを示している必要があります。特定のタイマイベントで、タイマは一連のDMAリクエスト（バースト）を開始します。TIMx_DMARレジスタへの各書込みは、実際にタイマレジスタの1つにリダイレクトされます。

TIMx_DCRレジスタのDBL[4:0] ビットによって、DMAバースト長がセットされます。タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCRレジスタのDBA[4:0] ビットは、DMA転送のDMAベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し／書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

TIMx_DCRレジスタのDBSS[3:0] ビットでは、DMAバースト転送をトリガする割込みソースを定義します（詳細はセクション 54.6.29：DMA制御レジスタ（TIMx_DCR）（x = 1、8）を参照）。

たとえば、更新イベント時にCCRxレジスタ値の内容を更新するためにタイマDMAバースト機能を使用します（x = 2、3、4）。このとき、DMAはCCRxレジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応するDMAチャネルを次のように設定します。
   - DMAチャネルペリフェラルアドレスを、DMARレジスタアドレスとします。
   - DMAチャネルメモリアドレスを、DMAによってCCRxレジスタに転送されるデータを格納するRAM内のバッファアドレスとします。
   - 転送データ数 = 3とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBAとDBLのビットフィールドを次のように設定することによって、DCRレジスタを設定します。DBL = 3回転送、DBA = 0xE、およびDBSS = 1。
3. TIMx更新DMAリクエストを有効にします（DIERレジスタのUDEビットをセット）。
4. TIMxを有効にします。
5. DMAチャネルを有効にします。

この例は、各CCRxレジスタが1回更新される場合です。たとえば、各CCRxレジスタが2回更新される場合は、転送データ数は6でなければなりません。データ1、データ2、データ3、データ4、データ5、データ6を格納するRAMのバッファを例にします。データは、CCRxレジスタに次のように転送されます。最初の更新DMAリクエストでデータ1がCCR2に転送され、データ2はCCR3に、データ3は

CCR4にそれぞれ転送され、2番目の更新DMAリクエストでデータ4がCCR2に、データ5がCCR3に、データ6がCCR4にそれぞれ転送されます。

注： **null値を予約済みレジスタに書き込むことができます。**

54.3.33 TIM1/TIM8DMAリクエスト

以下の表に示すように、TIM1/TIM8 はDMAリクエストを生成できます。

表545.DMAリクエスト

DMAリクエスト信号	DMAリクエストあり	有効制御ビット
tim_upd_dma	更新	UDE
tim_cc1_dma	キャプチャ／比較1	CC1DE
tim_cc2_dma	キャプチャ／比較2	CC2DE
tim_cc3_dma	キャプチャ／比較3	CC3DE
tim_cc4_dma	キャプチャ／比較4	CC4DE
tim_com_dma	転流（COM）	COMDE
tim_trg_dma	トリガ	TDE

54.3.34 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M33 コアは停止状態）、TIMxカウンタは、DBGモジュールのDBG_TIMx_STOP設定ビットに応じて、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

安全のため、カウンタが停止すると、出力は無効になります（MOEビットのリセット時と同じ）。通常強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSIビット = 1）、GPIOコントローラで制御することができます（OSSIビット = 0）。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

54.4 TIM1/TIM8 低消費電力モード

表546.低消費電力モードが TIM1/TIM8 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスはSLEEPモードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

54.5 TIM1/TIM8 割込み

表547に示すように、TIM1/TIM8 は複数の割込みを生成できます。

表547.割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
TIM_UP	更新	UIF	UIE	UIFに0を書き込む	あり	不可
TIM_CC	キャプチャ／比較1	CC1IF	CC1IE	CC1IFに0を書き込む	あり	不可
	キャプチャ／比較2	CC2IF	CC2IE	CC2IFに0を書き込む	あり	不可
	キャプチャ／比較3	CC3IF	CC3IE	CC3IFに0を書き込む	あり	不可
	キャプチャ／比較4	CC4IF	CC4IE	CC4IFに0を書き込む	あり	不可
TIM_TRG_COM	転流（COM）	COMIF	COMIE	COMIFに0を書き込む	あり	不可
	トリガ	TIF	TIE	TIFに0を書き込む	あり	不可
TIM_DIR_IDX	インデックス	IDXF	IDXIE	IDXFに0を書き込む	あり	不可
	方向	DIRF	DIRIE	DIRFに0を書き込む	あり	不可
TIM_BRK	ブレーク	BIF	BIE	BIFに0を書き込む	あり	不可
	Break2	B2IF		B2IFに0を書き込む	あり	不可
	システムブレーク	SBIF		SBIFに0を書き込む	あり	不可
TIM_IERR	インデックスエラー	IERRF	IERRIE	IERRFに0を書き込む	あり	不可
TIM_TER	遷移エラー	TERRF	TERRIE	TERRFに0を書き込む	あり	不可

54.6 TIM1/TIM8 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション1.2を参照してください。

54.6.1 制御レジスタ1（TIMx_CR1）（x = 1、8）

アドレスオフセット：0x000

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DITHEN**：ディザリングイネーブル

0：ディザリングは無効です。
1：ディザリングは有効です。

注： DITHENビットは、CENビットがリセットされているときのみ変更できます。

ビット11 **UIFREMAP**：UIFステータスビットの再配置

0：再配置なし。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされません。
1：再配置は有効です。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_etr_in、tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{tim_ker_ck}$
01：$t_{DTS}=2*t_{tim_ker_ck}$
10：$t_{DTS}=4*t_{tim_ker_ck}$
11：予約済み - この値をプログラミングしないでください。

ビット7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARRレジスタはバッファされません。
1：TIMx_ARRレジスタはバッファされます。

ビット6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモード。カウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01： センターアラインモード1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット3 **OPM**：ワンパルスモード。

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CENビットをクリア）。

ビット2 **URS**：更新リクエストソース

このビットは、UEVイベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたはDMAリクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたはDMAリクエストを生成します（有効な場合）。

ビット1 **UDIS**：更新ディセーブル

このビットは、UEVイベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEVは有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UGビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CENビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的にCENビットをセットできます。

54.6.2 制御レジスタ2（TIMx_CR2）（x = 1、8）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5
						rw		rw	rw	rw	rw		rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OIS4N	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット24 予約済みであり、リセット値に保持する必要があります。

ビット23:20 **MMS2[3:0]**：マスタモード選択2

これらのビットにより、選択される同期（tim_trgo2）について、ADCに情報を送信できるようになります。組み合わせは、次のとおりです。

0000：**リセット**- TIMx_EGRレジスタのUGビットがトリガ出力（tim_trgo2）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo2信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタイネーブル信号CNT_ENがトリガ出力（tim_trgo2）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、tim_trgo2には遅延が存在します（TIMx_SMCRレジスタのMSMビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo2）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IFフラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（tim_trgo2）。

0100：**比較** - tim_oc1refc信号がトリガ出力（tim_trgo2）として使用されます。

0101：**比較** - tim_oc2refc信号がトリガ出力（tim_trgo2）として使用されます。

0110：**比較** - tim_oc3refc信号がトリガ出力（tim_trgo2）として使用されます。

0111：**比較** - tim_oc4refc信号がトリガ出力（tim_trgo2）として使用されます。

1000：**比較** - tim_oc5refc信号がトリガ出力（tim_trgo2）として使用されます。

1001：**比較** - tim_oc6refc信号がトリガ出力（tim_trgo2）として使用されます。

1010：**パルス比較** - tim_oc4refcの立上がりエッジまたは立下がりエッジによって、tim_trgo2にパルスが生成されます。

1011：**パルス比較** - tim_oc6refcの立ち上がりエッジまたは立ち下がりエッジによって、tim_trgo2にパルスが生成されます。

1100：**パルス比較** - tim_oc4refcまたはtim_oc6refcの立ち上がりエッジによって、tim_trgo2にパルスが生成されます。

1101：**パルス比較** - tim_oc4refcの立ち上がりエッジまたはtim_oc6refcの立ち下がりエッジによって、tim_trgo2にパルスが生成されます。

1110：**パルス比較** - tim_oc5refcまたはtim_oc6refcの立ち上がりエッジによって、tim_trgo2にパルスが生成されます。

1111：**パルス比較** - tim_oc5refcの立ち上がりエッジまたはtim_oc6refcの立ち下がりエッジによって、tim_trgo2にパルスが生成されます。

注： スレーブタイマまたはADCのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **OIS6**：出力アイドル状態6（tim_oc6出力）
OIS1bitの説明を参照してください。

ビット17 予約済みであり、リセット値に保持する必要があります。

ビット16 **OIS5**：出力アイドル状態5（tim_oc5出力）
OIS1bitの説明を参照してください。

ビット15 **OIS4N**：出力アイドル状態4（tim_oc4n出力）
OIS1Nビットの説明を参照してください。

ビット14 **OIS4**：出力アイドル状態4（tim_oc4出力）
OIS1bitの説明を参照してください。

ビット13 **OIS3N**：出力アイドル状態3（tim_oc3n出力）
OIS1Nビットの説明を参照してください。

ビット12 **OIS3**：出力アイドル状態3（tim_oc3n出力）
OIS1bitの説明を参照してください。

ビット11 **OIS2N**：出力アイドル状態2（tim_oc2n出力）
OIS1Nビットの説明を参照してください。

ビット10 **OIS2**：出力アイドル状態2（tim_oc2出力）
OIS1bitの説明を参照してください。

ビット9 **OIS1N**：出力アイドル状態1（tim_oc1n出力）
0：MOE=0のとき、デッドタイム後にtim_oc1n=0
1：MOE=0のとき、デッドタイム後にtim_oc1n=1
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット8 **OIS1**：出力アイドル状態1（tim_oc1出力）
0：MOE=0のとき、tim_oc1=0（デッドタイム後に）
1：MOE=0のとき、tim_oc1=1（デッドタイム後に）
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット7 **TI1S**：tim_ti1選択
0：tim_ti1_in[15:0] マルチプレクサ出力がtim_ti1入力に接続されます。
1：tim_ti1_in[15:0]、tim_ti2_in[15:0]、およびtim_ti3_in[15:0] マルチプレクサ出力がXOR演算されてtim_ti1入力に接続されます。

ビット25、6:4 **MMS[3:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択します（tim_trgo）。組み合わせは、次のとおりです。

0000：**リセット** - TIMx_EGRレジスタのUGビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタ·イネーブル信号CNT_ENがトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、tim_trgoには遅延が存在します（TIMx_SMCRレジスタのMSMビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IFフラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（tim_trgo）。

0100：**比較** - tim_oc1refc信号がトリガ出力（tim_trgo）として使用されます。

0101：**比較** - tim_oc1ref信号がトリガ出力（tim_trgo）として使用されます。

0110：**比較** - tim_oc1ref信号がトリガ出力（tim_trgo）として使用されます。

0111：**比較** - tim_oc4refc信号がトリガ出力（tim_trgo）として使用されます。

1000：**エンコーダ·クロック出力** - エンコーダ·クロック信号がトリガ出力（tim_trgo）として使用されます。このコードはSMS[3:0] 値、0001、0010、0011、1010、1011、1100、1101、1110、1111に対して有効です。その他のSMS[3:0] コードは許されず、予想外の動作につながることがあります。

その他のコードは予約済み

注： スレーブタイマまたはADCのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット3 **CCDS**：キャプチャ／比較DMA選択

0：CCx DMAリクエストは、CCxイベントが発生すると送信されます。

1：CCx DMAリクエストは、更新イベントが発生すると送信されます。

ビット2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMGビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされているときに（CCPC=1）、COMGビットをセットすることによって、またはtim_trgiの立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、およびOCxMビットはプリロードされません。

1：CCxE、CCxNE、およびOCxMビットがプリロードされます。書込みの後、これらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUSビットに応じて、COMGビットがセットまたはtim_trgiで立ち上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

54.6.3 TIMxスレーブモード制御レジスタ (TIMx_SMCR)（x = 1、8）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
						rw	rw			rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **SMSPS**：SMSプリロードソース

このビットは、プリロードからアクティブへのSMS[3:0] ビットフィールドの転送をトリガするイベントのいずれかを選択します。

0：転送は、タイマの更新イベントによってトリガされます。

1：転送は、インデックスイベントによってトリガされます。

ビット24 **SMSPE**：SMSプリロードイネーブル

このビットは、SMS[3:0] ビットフィールドをプリロードするかどうかを選択します。

0：SMS[3:0] ビットフィールドはプリロードされません。

1：SMS[3:0] プリロードは有効です。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **TS[4:3]**：トリガ選択 - ビット4:3

TS[2:0] 説明を参照 - ビット6:4

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **ETP**：外部トリガ極性

このビットは、tim_etr_inとtim_$\overline{\text{etr_in}}$ のいずれがトリガ動作に使用されるかを選択します。

0：tim_etr_inは反転されず、ハイレベルまたは立上がりエッジでアクティブです。

1：tim_etr_inは反転され、ローレベルまたは立下がりエッジでアクティブです。

ビット14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード2を有効にします。

0：外部クロックモード2は無効です。

1：外部クロックモード2は有効です。カウンタは、tim_etrf信号のアクティブエッジによってクロック供給されます。

注： ECEビットをセットすることは、tim_trgiがtim_etrfに接続された状態で外部クロックモード1を選択することと同じ効果があります（SMS=111、TS=00111）。

外部クロックモード2と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、tim_trgiをtim_etrfに接続することはできません（TSビットが00111でないことが必要）。

外部クロックモード1と外部クロックモード2が同時に有効な場合、外部クロック入力はtim_etrfです。

ビット13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号tim_etrpの周波数は、TIMxCLK周波数の1/4までに制限されます。プリスケーラを有効にすると、tim_etrp周波数を低減できます。これは、高速な外部クロックをtim_etr_inに入力するときに役立ちます。

00：プリスケーラオフ

01：tim_etr_in周波数は2分周されます。

10：tim_etr_in周波数は4分周されます。

11：tim_etr_in周波数は8分周されます。

ビット11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、tim_etrp信号をサンプルする周波数と、tim_etrpに適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS} で行われます。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット7 **MSM：**マスタ/スレーブモード

0：影響なし。

1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgoを通じて）完全な同期状態が得られるまで遅延されます。これは、1つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット6:4 **TS[2:0]**：トリガ選択

このビットフィールドはTS[4:3] ビットと結合されます。

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ0（tim_itr0）

00001：内部トリガ1（tim_itr1）

00010：内部トリガ2（tim_itr2）

00011：内部トリガ3（tim_itr3）

00100：tim_ti1エッジ検出回路（tim_ti1f_ed）

00101：フィルタ･タイマ入力1（tim_ti1fp1）

00110：フィルタタイマ入力2（tim_ti2fp2）

00111：外部トリガ入力（tim_etrf）

01000：内部トリガ4（tim_itr4）

01001：内部トリガ5（tim_itr5）

01010：内部トリガ6（tim_itr6）

01011：内部トリガ7（tim_itr7）

01100：内部トリガ8（tim_itr8）

01101：内部トリガ9（tim_itr9）

01110：内部トリガ10（tim_itr10）

01111：内部トリガ11（tim_itr11）

10000：内部トリガ12（tim_itr12）

10001：内部トリガ13（tim_itr13）

10010：内部トリガ14（tim_itr14）

10011：内部トリガ15（tim_itr15）

その他：予約済み

各タイマでのtim_itrxの詳細については、表 532：内部トリガ接続を参照してください。

注： 遷移時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（たとえば、SMS = 000のとき）にのみ変更しなければなりません。

ビット3 **OCCS**：OCREFクリア選択

このビットは、OCREFクリアソースを選択するために使用されます。

0：tim_ocref_clr_intはtim_ocref_clr入力に接続されます。

1：tim_ocref_clr_intはtim_etrfに接続されます。

ビット16、2:0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：直交エンコーダ·モード1、x2モード - カウンタは、tim_ti2fp2のレベルに応じて、tim_ti1fp1のエッジでカウント·アップ／ダウンします。

0010：直交エンコーダ·モード2、x2モード - カウンタは、tim_ti1fp1のレベルに応じて、tim_ti2fp2のエッジでカウント·アップ／ダウンします。

0011：直交エンコーダ·モード3、x4モード - カウンタは、他の入力のレベルに応じて、tim_ti1fp1とtim_ti2fp2の両方のエッジでカウント·アップ/ダウンします。

0100：リセット·モード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲート·モード - カウンタ·クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ·モード - カウンタは、トリガtim_trgiの立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロック·モード1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセット·モードとトリガ·モードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲート·モードとリセット·モードの組み合わせ - カウンタ·クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

1010：エンコーダモード：クロックプラス方向、x2モード。

1011：クロックプラス方向、x1モード。tim_ti2fp2エッジ検出はCC2Pによって設定されます。

1100：エンコーダモード：方向クロック、x2モード。

1101：方向クロック、x1モード。tim_ti1fp1とtim_ti2fp2エッジ検出はCC1PとCC2Pによって設定されます。

1110：直交エンコーダ·モード：x1モード、tim_ti1fp1エッジのみでカウント、エッジ検出はCC1Pによって設定されます。

1111：直交エンコーダ·モード：x1モード、tim_ti1fp2エッジのみでカウント、エッジ検出はCC2Pによって設定されます。

注： トリガ入力としてtim_ti1f_edが選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_edはTI1Fの変化ごとに1パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgoまたはtim_trgo2の信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

54.6.4 DMA／割込み有効レジスタ（TIMx_DIER）（x = 1、8）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.
								rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TERRIE**：転送エラー割込みイネーブル
0：遷移エラー割込みは無効です。
1：遷移エラー割込みは有効です。

ビット22 **IERRIE**：インデックスエラー割込みイネーブル
0：インデックスエラー割込みは無効です。
1：インデックスエラー割込みは有効です。

ビット21 **DIRIE**：方向変更割込みイネーブル
0：方向変更割込みは無効です。
1：方向変更割込みは有効です。

ビット20 **IDXIE**：インデックス割込みイネーブル
0：インデックス割込みは無効です。
1：インデックス変更割込みは有効です。

ビット19:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TDE**：トリガDMAリクエストイネーブル
0：トリガDMAリクエストは無効です。
1：トリガDMAリクエストは有効です。

ビット13 **COMDE**：COM DMAリクエストイネーブル
0：COM DMAリクエストは無効です。
1：COM DMAリクエストは有効です。

ビット12 **CC4DE**：キャプチャ /比較4 DMAリクエストイネーブル
0：CC4 DMAリクエストは無効です。
1：CC4 DMAリクエストは有効です。

ビット11 **CC3DE**：キャプチャ /比較3 DMAリクエストイネーブル
0：CC3 DMAリクエストは無効です。
1：CC3 DMAリクエストは有効です。

ビット10 **CC2DE**：キャプチャ /比較2 DMAリクエストイネーブル
0：CC2 DMAリクエストは無効です。
1：CC2 DMAリクエストは有効です。

ビット9 **CC1DE**：キャプチャ /比較1 DMAリクエストイネーブル
0：CC1 DMAリクエストは無効です。
1：CC1 DMAリクエストは有効です。

ビット8 **UDE**：更新DMAリクエストイネーブル
0：更新DMAリクエストは無効です。
1：更新DMAリクエストは有効です。

ビット7 **BIE**：ブレーク割込みイネーブル
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット5 **COMIE**：イベントCOM割込み有効
0：COM割込みは無効です。
1：COM割込みは有効です。

ビット4 **CC4IE**：キャプチャ／比較4割込みイネーブル
0：CC4割込みは無効です。
1：CC4割込みは有効です。

ビット3 **CC3IE**：キャプチャ／比較3割込みイネーブル
0：CC3割込みは無効です。
1：CC3割込みは有効です。

ビット2 **CC2IE**：キャプチャ／比較2割込みイネーブル
0：CC2割込みは無効です。
1：CC2割込みは有効です。

ビット1 **CC1IE**：キャプチャ／比較1割込みイネーブル
0：CC1割込みは無効です。
1：CC1割込みは有効です。

ビット0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

54.6.5 TIMxステータスレジスタ（TIMx_SR）（x = 1、8）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	CC6IF	CC5IF
								rc_w0	rc_w0	rc_w0	rc_w0			rc_w0	rc_w0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TERRF**：転送エラー割込みフラグ
このフラグは、エンコーダモードで遷移エラーが検出されたときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：エンコーダ遷移エラーは検出されていません。
1：エンコーダ遷移エラーが検出されています。

ビット22 **IERRF**：転送エラー割込みフラグ
このフラグは、インデックスエラーが検出されたときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：インデックスエラーは検出されていません。
1：インデックスエラーが検出されています。

ビット21 **DIRF**：方向変更割込みフラグ

このフラグは、エンコーダモードで方向が変わった（TIMx_CRのDIRビット値が変わった）ときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：方向の変更なし

1：方向の変更

ビット20 **IDXF**：インデックス割込みフラグ

このフラグは、インデックスイベントが検出されたときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：インデックスイベントは発生していません。

1：インデックスイベントが発生しています。

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **CC6IF**：キャプチャ／比較6割込みフラグ

CC1IFの説明を参照してください。

注： チャネル6は出力としてのみ設定できます。

ビット16 **CC5IF**：キャプチャ／比較5割込みフラグ

CC1IFの説明を参照してください。

注： チャネル5は出力としてのみ設定できます。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **SBIF**：システムブレーク割込みフラグ

このフラグは、システムブレーク入力がアクティブになると、ハードウェアによってセットされます。システムブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

PWM動作をリスタートするには、このフラグをリセットする必要があります。

0：ブレークイベントは発生していません。

1：システムブレーク入力でアクティブレベルが検出されました。TIMx_DIERレジスタのBIE=1の場合、割込みが生成されます。

ビット12 **CC4OF**キャプチャ／比較4オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット11 **CC3OF**：キャプチャ／比較3オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット10 **CC2OF**：キャプチャ／比較2オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット9 **CC1OF**：キャプチャ／比較1オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IFフラグがすでにセットされているときに、カウンタの値がTIMx_CCR1レジスタにキャプチャされました。

ビット8 **B2IF**：ブレーク2割込みフラグ

このフラグは、ブレーク2入力がアクティブになると、ハードウェアによってセットされます。ブレーク2入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク2入力でアクティブレベルが検出されました。TIMx_DIERレジスタのBIE=1の場合、割込みが生成されます。

ビット7 **BIF**：ブレーク割込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。TIMx_DIERレジスタのBIE=1の場合、割込みが生成されます。

ビット6 **TIF**：トリガ割込みフラグ

このフラグは、TRGトリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット5 **COMIF**：COM割込みフラグ

このフラグは、COMイベント時にハードウェアによってセットされます（キャプチャ/比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COMイベントは発生していません。

1：COM割込みがペンディング中です。

ビット4 **CC4IF**：キャプチャ／比較4割込みフラグ

CC1IFの説明を参照してください。

ビット3 **CC3IF**：キャプチャ／比較3割込みフラグ

CC1IFの説明を参照してください。

ビット2 **CC2IF**：キャプチャ／比較2割込みフラグ

CC1IFの説明を参照してください。

ビット1 **CC1IF**：キャプチャ／比較1割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、またはTIMx_CCR1レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネルCC1が出力として設定されている場合：カウンタTIMx_CNTの内容がTIMx_CCR1レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1の内容がTIMx_ARRの内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）にCC1IFビットはハイになります。センターアラインモードでのフラグの設定には3つのオプションがあります。詳細については、TIMx_CR1レジスタのCMSビットを参照してください。

CC1チャネルが入力として設定されている場合：このビットは、TIMx_CCR1レジスタでカウンタ値がキャプチャされた（TIMx_CCERのCC1PおよびCC1NPビット設定で定義されたエッジ検知に従って、IC1でエッジが検出された）ときに設定されます。

ビット0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰返しカウンタ値に関するオーバーフローまたはアンダーフロー（繰返しカウンタ=0の場合の更新）、およびTIMx_CR1レジスタのUDIS=0の場合。

–TIMx_CR1レジスタのURS=0かつUDIS=0であり、TIMx_EGRレジスタのUGビットを使用して、CNTがソフトウェアによって再初期化されたとき。

–TIMx_CR1レジスタのURS=0かつUDIS=0であり、トリガイベントによってCNTが再初期化されたとき（セクション 54.6.3：TIMxスレーブモード制御レジスタ (TIMx_SMCR)（x = 1、8）を参照）。

54.6.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 1、8）

アドレスオフセット：0x014

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

ビット15:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **B2G**：ブレーク2生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレーク2イベントが生成されます。MOEビットがクリアされ、B2IFフラグがセットされます。有効な場合は、関連する割込みが発生します。

ビット7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOEビットがクリアされ、BIFフラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SRレジスタのTIFフラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCxE、CCxNE、OCxMビットの更新（CCPCビットが設定されている場合）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット4 **CC4G**：キャプチャ /比較4生成

CC1Gの説明を参照してください。

ビット3 **CC3G**：キャプチャ /比較3生成

CC1Gの説明を参照してください。

ビット2 **CC2G**：キャプチャ /比較2生成

CC1Gの説明を参照してください。

ビット1 **CC1G**：キャプチャ／比較1生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル1でキャプチャ／比較イベントが生成されます。

CC1チャネルが出力として設定されている場合：

CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。

CC1チャネルが入力として設定されている場合：

カウンタの現在値がTIMx_CCR1レジスタにキャプチャされます。CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。CC1IFフラグがすでにハイの場合、CC1OFフラグがセットされます。

ビット0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

54.6.7 TIMx キャプチャ／比較モードレジスタ1（TIMx_CCMR1）（x = 1、8）

アドレスオフセット：0x018

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード：

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **IC2F[3:0]**：入力キャプチャ 2フィルタ

ビット11:10 **IC2PSC[1:0]**：入力キャプチャ 2プリスケーラ

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2にマッピングされます。

10：CC2チャネルは入力として設定され、tim_ic1はtim_ti2にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC2E=0）のときにのみ書込み可能です。

ビット7:4 **IC1F[3:0]**：入力キャプチャ 1フィルタ

このビットフィールドは、tim_ti1入力をサンプリングする周波数と、tim_ti1に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS} で行われます。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット3:2 **IC1PSC[1:0]**：入力キャプチャ 1プリスケーラ

このビットフィールドは、CC1入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCERレジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2イベントごとに行われます。

10：キャプチャは、4イベントごとに行われます。

11：キャプチャは、8イベントごとに行われます。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。

10：CC1チャネルは入力として設定され、tim_ic2はtim_ti1にマッピングされます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

54.6.8 TIMx キャプチャ／比較モード･レジスタ1 [オルタネート]（TIMx_CCMR1）（x = 1、8）

アドレスオフセット：0x018

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		OC1 CE	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC2CE**：出力比較2クリアイネーブル

ビット24、14:12 **OC2M[3:0]**：出力比較2モード

ビット11 **OC2PE**：出力比較2プリロードイネーブル

ビット10 **OC2FE**：出力比較2高速イネーブル

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2にマッピングされます。

10：CC2チャネルは入力として設定され、tim_ic1はtim_ti2にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC2E=0）のときにのみ書込み可能です。

ビット7 **OC1CE**：出力比較1クリアイネーブル

0：tim_oc1refはtim_ocref_clr_int信号の影響を受けません。

1：tim_oc1refはtim_ocref_clr_int信号のハイレベルが検出されるとクリアされます（tim_ocref_clr入力またはtim_etrf入力）。

ビット16、6:4 **OC1M[3:0]**：出力比較1モード

これらのビットは、tim_oc1およびtim_oc1nが導き出される出力基準信号tim_oc1refの動作を定義します。tim_oc1refはアクティブハイですが、tim_oc1およびtim_oc1nのアクティブレベルはCC1PおよびCC1NPビットに依存します。

0000：停止 - 出力比較レジスタTIMx_CCR1とカウンタTIMx_CNTとの間の比較結果は出力に影響しません。このモードは、タイマをソフトウェアのタイムベースとして使用する場合に使用できます。タイマ動作中に停止モードが有効になると、出力は停止状態になる前の状態（アクティブまたはインアクティブ）を維持します。

0001：一致時にチャネル1をアクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル1を非アクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的に低になります。

0011：反転 - TIMx_CNT = TIMx_CCR1のとき、tim_oc1refは反転します。

0100：強制非アクティブレベル - tim_oc1refは強制的にローになります。

0101：強制アクティブレベル - tim_oc1refは強制的にハイになります。

0110：PWMモード1 - カウントアップ時、チャネル1は、TIMx_CNT < TIMx_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル1は、TIMx_CNT> TIMx_CCR1の場合はインアクティブ（tim_oc1ref=“0”）に、そうでない場合はアクティブ（tim_oc1ref=“1”）になります。

0111：PWMモード2 - カウントアップ時、チャネル1は、TIMx_CNT < TIMx_CCR1の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル1は、TIMx_CNT> TIMx_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能なOPMモード1 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能なOPMモード2 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWMモード2と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み

1011：予約済み

1100：組み合わせPWMモード1 - tim_oc1refは、PWMモード1と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ORです。

1101：組み合わせPWMモード2 - tim_oc1refは、PWMモード2と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ANDです。

1110：非対称PWMモード1 - tim_oc1refは、PWMモード1と同様に挙動します。tim_oc1refcは、カウンタがカウントアップするときにtim_oc1refを出力し、カウントダウンするときにtim_oc2refを出力します。

1111：非対称PWMモード2 - tim_oc1refは、PWMモード2と同様に挙動します。tim_oc1refcは、カウンタがカウントアップするときにtim_oc1refを出力し、カウントダウンするときにtim_oc2refを出力します。

注： これらのビットは、LOCKレベル3がプログラムされていて（TIMx_BDTRレジスタのLOCKビット）、かつCC1S=“00”（チャネルは出力に設定）のときには、変更できません。

注： PWMモードでは、比較結果が変化したとき、出力比較モードが「停止」モードから「PWM」モードに切り替わったとき、および出力比較モードが「強制アクティブ／インアクティブ」モードから「PWM」モードに切り替わったときに、OCREFのレベルが変化します。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2レジスタのCCPCビットがセットされた場合、OC1Mアクティブビットは、COMが生成されたときにのみプリロードから新しい値をとります。

ビット3 **OC1PE**：出力比較1プリロードイネーブル

0：TIMx_CCR1のプリロードレジスタは無効です。TIMx_CCR1は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCKレベル3がプログラムされていて（TIMx_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力に設定）のときには、変更できません。

ビット2 **OC1FE**：出力比較1高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1パルスモード（TIMx_CR1レジスタのOPMビットのセット）で使用する必要があります。

0：CC1の動作は、トリガがオンのときでも、通常、カウンタとCCR1の値に依存します。トリガ入力のエッジ発生からCC1出力が有効になるまでの最小遅延は、5クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1出力に対して、比較一致のように働きます。このような場合、OCは、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1出力を有効にするまでの遅延は、3クロックサイクルに短縮されます。OCFEは、チャネルがPWM1またはPWM2モードに設定されている場合のみ機能します。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。

10：CC1チャネルは入力として設定され、tim_ic1はtim_ti2にマッピングされます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

54.6.9 TIMxキャプチャ /比較モードレジスタ2（TIMx_CCMR2）（x = 1、8）

アドレスオフセット：0x01C

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル3と出力比較モードでのチャネル4）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **IC4F[3:0]**：入力キャプチャ 4フィルタ

ビット11:10 **IC4PSC[1:0]**：入力キャプチャ 4プリスケーラ

ビット9:8 **CC4S[1:0]**：キャプチャ／比較4選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4チャネルは出力として設定されます。

01：CC4チャネルは入力として設定され、tim_ic4はtim_ti4にマッピングされます。

10：CC4チャネルは入力として設定され、tim_ic4はtim_ti3にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic4はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC4E=0）のときにのみ書込み可能です。

ビット7:4 **IC3F[3:0]**：入力キャプチャ 3フィルタ

ビット3:2 **IC3PSC[1:0]**：入力キャプチャ 3プリスケーラ

ビット1:0 **CC3S[1:0]**：キャプチャ／比較3選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3チャネルは出力として設定されます。

01：CC3チャネルは入力として設定され、tim_ic3はtim_ti3にマッピングされます。

10：CC3チャネルは入力として設定され、tim_ic3はtim_ti4にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic3はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC3E=0）のときにのみ書込み可能です。

54.6.10 TIMx キャプチャ／比較モードレジスタ2 [オルタネート]（TIMx_CCMR2）（x = 1、8）

アドレスオフセット：0x01C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル3と出力比較モードでのチャネル4）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4 CE	OC4M [2:0]			OC4 PE	OC4 FE	CC4S [1:0]		OC3 CE	OC3M [2:0]			OC3 PE	OC3 FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC4CE**：出力比較4クリアイネーブル

ビット24、14:12 **OC4M[3:0]**：出力比較4モード

OC3M[3:0] ビットの説明を参照してください。

ビット11 **OC4PE**：出力比較4プリロードイネーブル

ビット10 **OC4PE**：出力比較4高速イネーブル

ビット9:8 **CC4S[1:0]**：キャプチャ／比較4選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4チャネルは出力として設定されます。

01：CC4チャネルは入力として設定され、tim_ic4はtim_ti4にマッピングされます。

10：CC4チャネルは入力として設定され、tim_ic4はtim_ti3にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic4はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC4E=0）のときにのみ書込み可能です。

ビット7 **OC3CE**：出力比較3クリアイネーブル

ビット16、6:4 **OC3M[3:0]**：出力比較3モード

これらのビットは、tim_oc3およびtim_oc3nが導き出される出力基準信号tim_oc3refの動作を定義します。tim_oc3refはアクティブハイですが、tim_oc3およびtim_oc3nのアクティブレベルはCC3PおよびCC3NPビットに依存します。

0000：停止 - 出力比較レジスタTIMx_CCR3とカウンタTIMx_CNTとの間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル3をアクティブレベルに設定します。tim_oc3ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ3（TIMx_CCR3）と一致したときに、強制的にハイになります。

0010：一致時にチャネル3を非アクティブレベルに設定します。tim_oc3ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ3（TIMx_CCR3）と一致したときに、強制的に低になります。

0011：反転 - TIMx_CNT = TIMx_CCR3のとき、tim_oc3refは反転します。

0100：強制インアクティブレベル - tim_oc3refは強制的にローになります。

0101：強制アクティブレベル - tim_oc3refは強制的にハイになります。

0110：PWMモード1 - カウント･アップ時、チャネル3は、TIMx_CNT < TIMx_CCR3の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル3は、TIMx_CNT> TIMx_CCR3 の場合はインアクティブ（tim_oc3ref=“0”）に、そうでない場合はアクティブ（tim_oc3ref=“1”）になります。

0111：PWMモード2 - カウントアップ時、チャネル3は、TIMx_CNT < TIMx_CCR3の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル3は、TIMx_CNT> TIMx_CCR3の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能なOPMモード1 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能なOPMモード2 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWMモード2と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：比較によるパルス：TIMxECRのPWPRSC[2:0] ビットフィールドとPW[7:0] ビットフィールドのプログラミング内容に従って、CCR3一致イベント時にパルスがtim_oc3refに生成されます。

1011：方向の出力。tim_oc3ref信号はDIRビットのコピーによって上書きされます。

1100：組み合わせPWMモード1 - tim_oc3refは、PWMモード1と同様に挙動します。tim_oc3refcは、tim_oc3refとtim_oc4refとの論理ORです。

1101：組み合わせPWMモード2 - tim_oc3refは、PWMモード2と同様に挙動します。tim_oc3refcは、tim_oc3refとtim_oc4refとの論理ANDです。

1110：非対称PWMモード1 - tim_oc3refは、PWMモード1と同様に挙動します。tim_oc3refcは、カウンタがカウントアップするときにtim_oc3refを出力し、カウントダウンするときにtim_oc4refを出力します。

1111：非対称PWMモード2 - tim_oc3refは、PWMモード2と同様に挙動します。tim_oc3refcは、カウンタがカウントアップするときにtim_oc3refを出力し、カウントダウンするときにtim_oc4refを出力します。

注： これらのビットは、LOCKレベル3がプログラムされていて（TIMx_BDTRレジスタのLOCKビット）、かつCC1S=“00”（チャネルは出力に設定）のときには、変更できません。

注： PWMモードでは、比較結果が変化したとき、または出力比較モードが停止モードからPWMモードに変更されたときにのみ、OCREFのレベルが変化します。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2レジスタのCCPCビットがセットされた場合、OC3Mアクティブビットは、COMが生成されたときにのみプリロードから新しい値をとります。

ビット3 **OC3PE**：出力比較3プリロードイネーブル

ビット2 **OC3FE**：出力比較3高速イネーブル

ビット1:0 **CC3S[1:0]**：キャプチャ／比較3選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic3はtim_ti3にマッピングされます。

10：CC2チャネルは入力として設定され、tim_ic3はtim_ti4にマッピングされます。

11：CC3チャネルは入力として設定され、tim_ic3はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC3E=0）のときにのみ書込み可能です。

54.6.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E
										rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	CC4NE	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **CC6P**：キャプチャ／比較6出力極性
CC1Pの説明を参照してください。

ビット20 **CC6E**：キャプチャ /比較6出力イネーブル
CC1Eの説明を参照してください。

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **CC5P**：キャプチャ／比較5出力極性
CC1Pの説明を参照してください。

ビット16 **CC5E**：キャプチャ／比較5出力イネーブル
CC1Eの説明を参照してください。

ビット15 **Cc4NP**：キャプチャ /比較4相補出力極性
CC1NPの説明を参照してください。

ビット14 **CC4NE**：キャプチャ /比較4相補出力イネーブル
CC1NEの説明を参照してください。

ビット13 **CC4P**：キャプチャ /比較4出力極性
CC1Pの説明を参照してください。

ビット12 **CC4E**：キャプチャ／比較4出力イネーブル
CC1Eの説明を参照してください。

ビット11 **CC3NP**：キャプチャ /比較3相補出力極性
CC1NPの説明を参照してください。

ビット10 **CC3NE**：キャプチャ /比較3相補出力イネーブル
CC1NEの説明を参照してください。

ビット9 **CC3P**」キャプチャ／比較3出力極性
CC1Pの説明を参照してください。

ビット8 **CC3E**：キャプチャ／比較3出力イネーブル
CC1Eの説明を参照してください。

ビット7 **CC2NP**：キャプチャ /比較2相補出力極性
CC1NPの説明を参照してください。

ビット6 **CC2NE**：キャプチャ /比較2相補出力イネーブル
CC1NEの説明を参照してください。

ビット5 **CC2P**：キャプチャ／比較2出力極性
CC1Pの説明を参照してください。

ビット4 **CC2E**：キャプチャ／比較2出力イネーブル
CC1Eの説明を参照してください。

ビット3 **CC1NP**：キャプチャ／比較1相補出力極性
CC1チャネルが出力として設定されている場合：
0：tim_oc1nはアクティブハイです。
1：tim_oc1nはアクティブローです。
CC1チャネルが入力として設定されている場合：
このビットは、tim_ti1fp1とtim_ti2fp1の極性を定義するためにCC1Pと組合せて使用されます。CC1Pの説明を参照してください。

注： このビットは、LOCKレベル2または3がプログラムされ、（TIMx_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力として設定）になった直後は書込みできません。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NPアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット2 **CC1NE**：キャプチャ /比較1相補出力イネーブル
0：オフ - tim_oc1nはアクティブではありません。tim_oc1nのレベルは、MOE、OSSI、OSSR、OIS1、OIS1NおよびCC1Eビットによって決まります。
1：オン - tim_oc1n信号は、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1Eビットにより、対応する出力ピンに出力されます。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NEアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット1 **CC1P**：キャプチャ／比較1出力極性

0：OC1はアクティブハイ（出力モード）／エッジ検知を選択（入力モード、下記を参照）

1：OC1はアクティブロー（出力モード）／エッジ検知を選択（入力モード、下記を参照）

CC1チャネルが入力として設定されているときは、CC1NP/CC1Pの両ビットは、トリガまたはキャプチャ動作に対するTI1FP1とTI2FP1のアクティブ極性を選択します。

CC1NP=0、CC1P=0：非反転/立上がりエッジ。この回路はTIxFP1の立上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1：反転／立下がりエッジ。この回路はTIxFP1の立下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1：非反転／両エッジ／この回路はTIxFP1の立上がりエッジと立下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCKレベル2または3がプログラムされた直後は書込みできません（TIMx_BDTRレジスタのLOCKビット）。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1Pアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット0 **CC1E**：キャプチャ／比較1出力イネーブル

0：キャプチャモードは無効／OC1は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1信号は、対応する出力ピンに出力されます。

CC1チャネルが出力として設定されている場合、OC1レベルは、CC1Eビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1NEビットに依存します。詳細については、表548を参照してください。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1Eアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

表548.ブレーク機能を持つ相補tim_ocxおよびtim_ocxnチャネルの出力制御ビット

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	tim_ocx出力状態	tim_ocxn出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0、tim_ocxn=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0	tim_ocxref + 極性 tim_ocxn = tim_ocxref xor CCxNP
		0	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocxn=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREFに対する相補（OCREFではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（非アクティブ状態で出力有効） tim_ocx=CCxP	tim_ocxref + 極性 tim_ocxn = tim_ocxref xor CCxNP
		1	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	オフ状態（インアクティブ状態で出力有効） tim_ocxn=CCxNP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイ･インピーダンス）	
	1		0	0		
			0	1	オフ状態（非アクティブ状態で出力有効） 非同期に：tim_ocx=CCxP、tim_ocxn=CCxNP（tim_brkまたはtim_brk2がトリガされた場合）。 クロックが存在する場合、（tim_brkがトリガされた場合にのみ有効）：アクティブな状態でOISxとOISxNがOCXとtim_ocxnにそれぞれ対応しないとみなされる場合（ハーフブリッジ設定でスイッチを駆動した場合に短絡の原因となる）、デッドタイム後にtim_ocx=OISxおよびtim_ocxn=OISxNとなります。 **注： tim_brk2はOSSI = OSSR = 1の場合にのみ使用できます。**	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIOが制御を引き継いだ場合）、OISx、OISxN、CCxP、およびCCxNPはクリアされたままでなければなりません。

注： 相補 tim_ocx および tim_ocxn チャネルに接続されている外部入出力ピンの状態は、tim_ocx および tim_ocxnチャネルの状態と、GPIO レジスタに依存します。

54.6.12 TIMx カウンタ（TIMx_CNT）（x = 1、8）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UIFCPY**：UIFコピー

このビットはTIMx_ISRレジスタのUIFビットの読出し専用コピー。TIMxCR1のUIFREMAPビットがリセットされると、ビット31は予約済みで、0で読み出されます。

ビット30:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分はありません。

54.6.13 TIMx プリスケーラ（TIMx_PSC）（x = 1、8）

アドレスオフセット：0x028

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（$f_{tim_cnt_ck}$）は$f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSCは、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGRレジスタのUGビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

54.6.14 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 1、8）

アドレスオフセット：0x02C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **ARR[19:0]**：自動再ロード値

ARRは、実際の自動再ロードレジスタにロードされる値です。

ARRの更新と動作の詳細については、セクション 54.3.3：2079ページのタイムベースユニットを参照してください。

自動再ロード値がnullのときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、自動再ロード値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

54.6.15 TIMx 繰返しカウンタレジスタ（TIMx_RCR）（x = 1、8）

アドレスオフセット：0x030

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **REP[15:0]**：繰り返しカウンタ再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP値からカウントをリスタートします。繰返しカウンタは、繰返し更新イベントUEVでのみ、REP値で再ロードされるので、TIMx_RCRレジスタへの書込みは、次の繰返し更新イベントまで有効になりません。

PWMモードでは、（REP+1）は次のことを意味します。

–エッジアラインモードでは、PWM周期の数

–センターアラインモードでは、PWMの1/2周期の数

54.6.16 TIMx キャプチャ／比較レジスタ1 （TIMx_CCR1）（x = 1、8）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR1[19:0]**：キャプチャ／比較1値

CC1チャネルが出力として設定されている場合：CCR1は、実際のキャプチャ／比較1レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1レジスタのOC1PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較1レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc1出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1チャネルが入力として設定されている場合：CR1の値は、最後の入力キャプチャ1イベント（tim_ic1）によって転送されたカウンタ値です。TIMx_CCR1レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

54.6.17 TIMx キャプチャ／比較レジスタ2 （TIMx_CCR2）（x = 1、8）

アドレスオフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR2[19:0]**：キャプチャ／比較2値

CC2チャネルが出力として設定されている場合：CCR2は、実際のキャプチャ／比較2レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1レジスタのOC2PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較2レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc2出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] に比較値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2チャネルが入力として設定されている場合：CCR2の値は、最後の入力キャプチャ2イベント（tim_ic2）によって転送されたカウンタ値です。TIMx_CCR2レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] にキャプチャ値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

54.6.18 TIMx キャプチャ／比較レジスタ3 （TIMx_CCR3）（x = 1、8）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR3[19:0]**：キャプチャ／比較値

CC3チャネルが出力として設定されている場合：CCR3は、実際のキャプチャ／比較3レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2レジスタのOC3PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較3レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc3出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR3[15:0] に比較値を保持します。CCR3[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:4] に整数部を保持します。CCR3[3:0] ビットフィールドにはディザリング部分が含まれます。

CC3チャネルが入力として設定されている場合：CCR3の値は、最後の入力キャプチャ 3イベント（tim_ic3）によって転送されたカウンタ値です。TIMx_CCR3レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR3[15:0] にキャプチャ値を保持します。CCR3[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:4] にキャプチャを保持します。CCR3[3:0] ビットはリセットされています。

54.6.19 TIMx キャプチャ／比較レジスタ4 （TIMx_CCR4）（x = 1、8）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR4[19:0]**：キャプチャ／比較値

CC4チャネルが出力として設定されている場合：CCR4は、実際のキャプチャ／比較4レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2レジスタのOC4PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較4レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較され、tim_oc4出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR4[15:0] に比較値を保持します。CCR4[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:4] に整数部を保持します。CCR4[3:0] ビットフィールドにはディザリング部分が含まれます。

CC4チャネルが入力として設定されている場合：CCR4の値は、最後の入力キャプチャ4イベント（tim_ic4）によって転送されたカウンタ値です。TIMx_CCR4レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR4[15:0] にキャプチャ値を保持します。CCR4[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:4] にキャプチャを保持します。CCR4[3:0] ビットはリセットされています。

54.6.20 TIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 1、8）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	BK2BID	BKBID	BK2 DSRM	BK DSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ビットBKBID/BK2BID/BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、およびDTG[7:0] は、LOCK設定に応じて書込みがロックされるので、TIMx_BDTRレジスタへの最初のアクセス時に、これらすべてを設定しなければならないことがあります。

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **BK2BID**：ブレーク2双方向

BKBIDの説明を参照してください

ビット28 **BKBID**：ブレーク双方向

0：ブレーク入力tim_brkは入力モードです。

1：ブレーク入力tim_brkは双方向モードです。

双方向モード（BKBIDビットが1にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット27 **BK2DSRM**：ブレーク2解除

BKDSRMの説明を参照してください

ビット26 **BKDSRM**：ブレーク解除

0：ブレーク入力tim_brkが設定されます。

1：ブレーク入力tim_brkは解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRMビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット25 **BK2P**：ブレーク2極性

0：ブレーク入力tim_brk2はアクティブローです。

1：ブレーク入力tim_brk2はアクティブハイです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット24 **BK2E**：ブレーク2イネーブル

このビットは、完全なブレーク2保護を有効にします（図 550：ブレークおよびブレーク2回路の概要のようにbk_acthおよびBKINソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク2機能は無効です。

1：ブレーク2機能は有効です。

注： BRKIN2はOSSR = OSSI = 1の場合にのみ使用してください。

注： このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット23:20 **BK2F[3:0]**：ブレーク2フィルタ

このビットフィールドは、tim_brk2入力をサンプリングする周波数と、tim_brk2に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、tim_brk2は非同期として動作します。

0001：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6

0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8

0110：$f_{SAMPLING} = f_{DTS}/4$、N = 6

0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8

1000：$f_{SAMPLING} = f_{DTS}/8$、N = 6

1001：$f_{SAMPLING} = f_{DTS}/8$、N = 8

1010：$f_{SAMPLING} = f_{DTS}/16$、N = 5

1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6

1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8

1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5

1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6

1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

注： このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk入力をサンプリングする周波数と、tim_brkに適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、tim_brkは非同期で動作します。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注：　このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力の1つ（tim_brkまたはtim_brk2）がアクティブになると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOEビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：ブレーク2イベントへの対応。OCおよびOCN出力は無効です。
ブレークイベントへの対応、またはMOEが0に書き込まれた場合：OCおよびOCN出力が無効か、OSSIビットによって強制的にアイドル状態になります。

1：OCおよびOCN出力は、それぞれのイネーブルビット（TIMx_CCERレジスタのCCxE、CCxNEビット）がセットされている場合は有効です。
詳細については、OC/OCNイネーブルの説明を参照してください（セクション 54.6.11 : TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8））。

ビット14 **AOE**：自動出力イネーブル

0：MOEはソフトウェアによってのみセットできます。

1：MOEは、ソフトウェアでセットできるほか、次回の更新イベントで自動的にセットできます（ブレーク入力tim_brkとtim_brk2のどちらもアクティブでないこと）。

注：　このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット13 **BKP**：ブレーク極性

0：ブレーク入力tim_brkはアクティブ･ローです。

1：ブレーク入力tim_brkはアクティブ･ハイです。

注：　このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注：　このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット12 **BKE**：ブレークイネーブル

このビットは、完全なブレーク保護を有効にします（図 550：ブレークおよびブレーク2回路の概要のようにbk_acthおよびBKINソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク機能は無効です。

1：ブレーク機能は有効です。

注： このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット11 **OSSR**：RUNモードのオフ状態の選択

このビットは、MOE=1のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSRは、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCNイネーブルの説明を参照してください（セクション 54.6.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8））。

0：インアクティブのとき、OC/OCN出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制するGPIOロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1またはCCxNE=1になると、OC/OCN出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCKレベル2がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット10 **OSSI**：アイドル･モードのオフ状態の選択

このビットは、MOE=0のとき、出力として設定されているチャネルでのブレークイベントやソフトウェアの書込みによって使用されます。

詳細については、OC/OCNイネーブルの説明を参照してください（セクション 54.6.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8））。

0：インアクティブのとき、OC/OCN出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制するGPIOロジックによって引き継がれます）。

1：インアクティブのとき、OC/OCN出力はまず強制的にインアクティブレベルにされ、次にデッドタイム後に強制的にアイドルレベルにされます。タイマは出力の制御を保持します。

注： このビットは、LOCKレベル2がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCKオフ - どのビットも書き込み保護されません。

01：LOCKレベル1 ー TIMx_BDTRレジスタのDTGビット、TIMx_CR2レジスタのOISxビットとOISxNビット、およびTIMx_BDTRレジスタのBKBID/BK2BID/BKE/BKP/AOEの各ビットには書込みできなくなります。

10：LOCKレベル2 ー LOCKレベル1に加えて、CC極性ビット（関連するチャネルがCCxSビットを通じて出力に設定されている場合は、TIMx_CCERレジスタのCCxP/CCxNPビット）とOSSRおよびOSSIビットも書き込めなくなります。

11：LOCKレベル3 ー LOCKレベル2に加えて、CC制御ビット（関連するチャネルがCCxSビットを通じて出力に設定されている場合は、TIMx_CCMRレジスタのOCxMおよびOCxPEビット）が書き込めなくなります。

注： LOCKビットは、リセット後に一度だけ書込みができます。いったんTIMx_BDTRレジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで$t_{dtg}=t_{DTS}$.

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで$T_{dtg}=2xt_{DTS}$.

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=8xt_{DTS}$.

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns（250ns単位）

32 µsから63 µs（1 µs単位）

64 µsから126 µs（2 µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされているとき、変更できません（TIMx_BDTRレジスタのLOCKビット）。

54.6.21 TIMx キャプチャ／比較レジスタ5 （TIMx_CCR5）（x = 1、8）

アドレスオフセット：0x048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[19:16]			
rw	rw	rw										rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR5 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **GC5C3**：チャネル5およびチャネル3のグループ化

チャネル3出力におけるひずみ：

0：tim_oc5refのtim_oc3refcへの影響はありません。

1：tim_oc3refcは、tim_oc3refとtim_oc5refの論理ANDです。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR2でプリロード機能を選択している場合）。

注： このひずみを組合せPWM信号に適用することもできます。

ビット30 **GC5C2**：チャネル5およびチャネル2のグループ化

チャネル2出力におけるひずみ：

0：tim_oc5refのtim_oc2refcへの影響はありません。

1：tim_oc2refcは、tim_oc2refとtim_oc5refの論理ANDです。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1でプリロード機能を選択している場合）。

注： このひずみを組合せPWM信号に適用することもできます。

ビット29 **GC5C1**：チャネル5およびチャネル1のグループ化

チャネル1出力におけるひずみ：

0：oc5refのoc1refcへの影響はありません。

1：oc1refcは、oc1refとoc5refの論理ANDです。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1でプリロード機能を選択している場合）。

注： このひずみを組合せPWM信号に適用することもできます。

ビット28:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR5[19:0]**：キャプチャ／比較5値

CCR5は、実際のキャプチャ／比較5レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3レジスタのOC5PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較5レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc5出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR5[15:0] に比較値を保持します。CCR5[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR5[19:4] に整数部を保持します。CCR5[3:0] ビットフィールドにはディザリング部分が含まれます。

54.6.22 TIMx キャプチャ／比較レジスタ6 （TIMx_CCR6）（x = 1、8）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR6 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR6[19:0]**：キャプチャ／比較6値

CCR6は、実際のキャプチャ／比較6レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3レジスタのOC6PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較6レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc6出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR6[15:0] に比較値を保持します。CCR6[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR6[19:4] に整数部を保持します。CCR6[3:0] ビットフィールドにはディザリング部分が含まれます。

54.6.23 TIMx キャプチャ／比較モードレジスタ3（TIMx_CCMR3）（x = 1、8）

アドレスオフセット：0x050

リセット値：0x0000 0000

上記のCCMR1レジスタの説明を参照してください。チャネル5および6は出力としてのみ設定できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6 CE	OC6M [2:0]			OC6 PE	OC6FE	Res.	Res.	OC5 CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC6CE**：出力比較6クリアイネーブル

ビット24、14:12 **OC6M[3:0]**：出力比較6モード

ビット11 **OC6PE**：出力比較6プリロードイネーブル

ビット10 **OC6FE**：出力比較6高速イネーブル

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **OC5CE**：出力比較5クリアイネーブル

ビット16、6:4 **OC5M[3:0]**：出力比較5モード

ビット3 **OC5PE**：出力比較5プリロードイネーブル

ビット2 **OC5FE**：出力比較5高速イネーブル

ビット1:0 予約済みであり、リセット値に保持する必要があります。

54.6.24 TIMx タイマデッドタイムレジスタ2（TIMx_DTR2）（x = 1、8）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DTPE**：デッドタイムプリロードイネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット16 **DTAE**：デッドタイム非対称イネーブル

0：立ち上がりおよび立ち下がりエッジでのデッドタイムは同一で、DTG[7:0] レジスタで定義されます。

1：立ち上がりエッジのデッドタイムはDTG[7:0] レジスタで定義され、立ち下がりエッジのデッドタイムはDTGF[7:0] ビットで定義されます。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジジェネレータのセットアップ

このビットフィールドでは、立下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここで$t_{dtg}=t_{DTS}$.

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここで$T_{dtg}=2xt_{DTS}$.

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで$T_{dtg}=8xt_{DTS}$.

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで$T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns（250ns単位）

32 µsから63 µs（1 µs単位）

64 µsから126 µs（2 µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされているとき、変更できません（TIMx_BDTRレジスタのLOCKビット）。

54.6.25 TIMx タイマエンコーダ制御レジスタ（TIMx_ECR）（x = 1、8）

アドレスオフセット：0x058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PWPRSC[2:0]			PW[7:0]							
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS[1:0]		FIDX	IBLK[1:0]		IDIR[1:0]		IE
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **PWPRSC[2:0]**：パルス生成回路（幅）プリスケーラ

このビットフィールドは、パルス発生回路用のクロックプリスケーラを次のように設定します。

$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$

ビット23:16 **PW[7:0]**：パルス幅

このビットフィールドは、パルスの長さを次のように定義します。

$t_{PW} = PW[7:0] \times t_{PWG}$

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:6 **IPOS[1:0]**：インデックス位置決め

直交エンコーダモード（SMS[3:0] = 0001、0010、0011、1110、1111）で、このビットは、インデックスイベントがカウンタをリセットするAB入力の設定を示します。

00：AB = 00のときにインデックスがカウンタをリセットします。
01：AB = 01のときにインデックスがカウンタをリセットします。
10：AB = 10のときにインデックスがカウンタをリセットします。
11：AB = 11のときにインデックスがカウンタをリセットします。

方向クロックモードまたはクロックプラス方向モード（SMS[3:0] = 1010、1011、1100、1101）で、これらのビットは、インデックスイベントがカウンタをリセットするレベルを示します。双方向クロックモードでは、これは両方のクロック入力に適用されます。

x0：クロックが0のときにインデックスがカウンタをリセットします。
x1：クロックが1のときにインデックスがカウンタをリセットします。

注： IPOS[1] ビットは意味がありません。

ビット5 **FIDX**：最初のインデックス

このビットは、最初のインデックスのみが考慮されるかどうかを示します。

0：インデックスは常にアクティブです。
1：最初のインデックスのみがカウンタをリセットします。

ビット4:3 **IBLK[1:0]**：インデックスのブランキング

このビットは、インデックスイベントがtim_ti3またはtim_ti4の入力によって調整されるかどうかを示します。

00：インデックスは常にアクティブです。
01：インデックスは、CC3Pビットフィールドに従って、tim_ti3入力がアクティブなときに無効になります。
10：インデックスは、CC4Pビットフィールドに従って、tim_ti4入力がアクティブなときに無効になります。
11：予約済み

ビット2:1 **IDIR[1:0]**：インデックスの方向
このビットは、どちらの方向のときにインデックスイベントがカウンタをリセットするかを示します。
00：方向に関係なく、インデックスがカウンタをリセットします。
01：アップカウントのときのみ、インデックスがカウンタをリセットします。
10：ダウンカウントのときのみ、インデックスがカウンタをリセットします。
11：予約済み

ビット0 **IE**：インデックスイネーブル
このビットは、インデックスイベントがカウンタをリセットするかどうかを示します。
0：インデックスは無効です。
1：インデックスは有効です。

54.6.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 1、8）

アドレスオフセット：0x05C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:24 **TI4SEL[3:0]**：tim_ti4[15:0] の入力を選択します。
0000 :tim_ti4_in0：TIMx_CH4
0001：tim_ti4_in1
.....
1111：tim_ti4_in15
相互接続一覧については、セクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **TI3SEL[3:0]**：tim_ti3[15:0] の入力を選択します。
0000 :tim_ti3_in0：TIMx_CH2
0001：tim_ti3_in1
.....
1111：tim_ti3_in15
相互接続一覧については、セクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **TI2SEL[3:0]**：tim_ti2_in[15:0] の入力を選択します。
0000 :tim_ti2_in0：TIMx_CH2
0001：tim_ti2_in1
.....
1111：tim_ti2_in15
相互接続一覧については、セクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TI1SEL[3:0]**：tim_ti1[15:0] の入力を選択します。
0000 :tim_ti1_in0：TIMx_CH1
0001：tim_ti1_in1
.....
1111：tim_ti1_in15
相互接続一覧については、セクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

54.6.27 TIMx オルタネート機能オプションレジスタ1 （TIMx_AF1）（x = 1、8）

アドレスオフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17:14 **ETRSEL[3:0]**：etr_inソース選択
これらのビットは、etr_in入力ソースを選択します。
0000 :tim_etr0：TIMx_ETR入力
0001：tim_etr1
.....
1111：tim_etr15
製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。
注： これらのビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット13 **BKCMP4P**：tim_brk_cmp4入力極性
このビットは、tim_brk_cmp4入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。
0：tim_brk_cmp4入力の極性は反転されません（BKP = 0の場合はアクティブロー、BKP = 1の場合はアクティブハイ）。
1：tim_brk_cmp4入力の極性を反転させます（BKP = 0の場合はアクティブハイ、BKP = 1の場合はアクティブロー）。
注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット12 **BKCMP3P**：tim_brk_cmp3入力極性

このビットは、tim_brk_cmp3入力の感度を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp3入力の極性は反転されません（BKP = 0の場合はアクティブロー、BKP = 1の場合はアクティブハイ）。

1：tim_brk_cmp3入力の極性を反転させます（BKP = 0の場合はアクティブハイ、BKP = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット11 **BKCMP2P**：tim_brk_cmp2入力極性

このビットは、tim_brk_cmp2入力検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2入力の極性は反転されません（BKP = 0の場合はアクティブロー、BKP = 1の場合はアクティブハイ）。

1：tim_brk_cmp2入力の極性を反転させます（BKP = 0の場合はアクティブハイ、BKP = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット10 **BKCMP1P**：tim_brk_cmp1入力極性

このビットは、tim_brk_cmp1入力の感度を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1入力の極性は反転されません（BKP = 0の場合はアクティブロー、BKP = 1の場合はアクティブハイ）。

1：tim_brk_cmp1入力の極性を反転させます（BKP = 0の場合はアクティブハイ、BKP = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット9 **BKINP**：TIMx_BKIN入力極性

このビットは、TIMx_BKINオルタネート機能入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN入力の極性を反転しません（BKP = 0ならばアクティブ･ロー、BKP = 1ならばアクティブ･ハイ）

1：TIMx_BKIN入力の極性を反転します（BKP = 0ならばアクティブ･ハイ、BKP = 1ならばアクティブ･ロー）

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット8 **BKCMP8E**：tim_brk_cmp8イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp8を有効化します。tim_brk_cmp8出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp8入力は無効です。

1：tim_brk_cmp8入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット7 **BKCMP7E**：tim_brk_cmp7イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp7を有効化します。tim_brk_cmp7出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp7入力は無効です。

1：tim_brk_cmp7入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット6 **BKCMP6E**：tim_brk_cmp6イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp6を有効化します。tim_brk_cmp6出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp6入力は無効です。

1：tim_brk_cmp6入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット5 **BKCMP5E**：tim_brk_cmp5イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp5を有効化します。tim_brk_cmp5出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp5入力は無効です。

1：tim_brk_cmp5入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット4 **BKCMP4E**：tim_brk_cmp4イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp4を有効化します。tim_brk_cmp4出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp4入力は無効です。

1：tim_brk_cmp4入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット3 **BKCMP3E**：tim_brk_cmp3イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp3を有効化します。tim_brk_cmp3出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp3入力は無効です。

1：tim_brk_cmp3入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット2 **BKCMP2E**：tim_brk_cmp2イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp2を有効化します。tim_brk_cmp2出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp2入力は無効です。

1：tim_brk_cmp2入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット1 **BKCMP1E**：tim_brk_cmp1イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp1を有効化します。tim_brk_cmp1出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp1入力は無効です。

1：tim_brk_cmp1入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット0 **BKINE**：TIMx_BKIN入力イネーブル

このビットは、タイマのtim_brk入力に対してTIMx_BKINオルタネート機能入力を有効化します。TIMx_BKIN入力は、ほかのtim_brkソースとの「論理和」がとられます。

0：TIMx_BKIN入力は無効です。

1：TIMx_BKIN入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注： 製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

54.6.28 TIMx オルタネート機能レジスタ2 （TIMx_AF2）（x = 1、8）

アドレスオフセット：0x064

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK2CMP4P	BK2CMP3P	BK2CMP2P	BK2CMP1P	BK2INP	BK2CMP8E	BK2CMP7E	BK2CMP6E	BK2CMP5E	BK2CMP4E	BK2CMP3E	BK2CMP2E	BK2CMP1E	BK2INE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **OCRSEL[2:0]**：ocref_clrソース選択

これらのビットは、ocref_clr入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

.....

111：tim_ocref_clr7

製品固有の情報についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

注： これらのビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **BK2CMP4P**：tim_brk2_cmp4入力極性

このビットは、tim_brk2_cmp4入力の検知を選択します。BK2P極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp4入力の極性は反転されません（BK2P = 0の場合はアクティブロー、BK2P = 1の場合はアクティブハイ）。

1：tim_brk2_cmp4入力の極性を反転させます（BK2P = 0の場合はアクティブハイ、BK2P = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット12 **BK2CMP3P**：tim_brk2_cmp3入力極性

このビットは、tim_brk2_cmp3入力の感度を選択します。BK2P極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp3入力の極性は反転されません（BK2P = 0の場合はアクティブロー、BK2P = 1の場合はアクティブハイ）。

1：tim_brk2_cmp3入力の極性を反転させます（BK2P = 0の場合はアクティブハイ、BK2P = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット11 **BK2CMP2P**：tim_brk2_cmp2入力極性

このビットは、tim_brk2_cmp2入力の感度を選択します。BK2P極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp2入力の極性は反転されません（BK2P = 0の場合はアクティブロー、BK2P = 1の場合はアクティブハイ）。

1：tim_brk2_cmp2入力の極性を反転させます（BK2P = 0の場合はアクティブハイ、BK2P = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット10 **BK2CMP1P**：tim_brk2_cmp1入力極性

このビットは、tim_brk2_cmp1入力の感度を選択します。BK2P極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp1入力の極性は反転されません（BK2P = 0の場合はアクティブロー、BK2P = 1の場合はアクティブハイ）。

1：tim_brk2_cmp1入力の極性を反転させます（BK2P = 0の場合はアクティブハイ、BK2P = 1の場合はアクティブロー）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット9 **BK2INP**：TIMx_BKIN2入力極性

このビットは、TIMx_BKIN2オルタネート機能入力の検知を選択します。BK2P極性ビットとともにプログラムする必要があります。

0： TIMx_BKIN2入力の極性を反転しません（BK2P = 0ならばアクティブ･ロー、BK2P = 1ならばアクティブ･ハイ）

1： TIMx_BKIN2入力の極性を反転します（BK2P = 0ならばアクティブ･ハイ、BK2P = 1ならばアクティブ･ロー）

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット8 **BK2CMP8E**：tim_brk2_cmp8イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp8を有効化します。tim_brk2_cmp8出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp8入力は無効です。

1：tim_brk2_cmp8入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット7 **BK2CMP7E**：tim_brk2_cmp7イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp7を有効化します。tim_brk2_cmp7出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp7入力は無効です。

1：tim_brk2_cmp7入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット6 **BK2CMP6E**：tim_brk2_cmp6イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp6を有効化します。tim_brk2_cmp6出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp6入力は無効です。

1：tim_brk2_cmp6入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット5 **BK2CMP5E**：tim_brk2_cmp5イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp5を有効化します。tim_brk2_cmp5出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp5入力は無効です。

1：tim_brk2_cmp5入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット4 **BK2CMP4E**：tim_brk2_cmp4イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp4を有効化します。tim_brk2_cmp4出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp4入力は無効です。

1：tim_brk2_cmp4入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット3 **BK2CMP3E**：tim_brk2_cmp3イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp3を有効化します。tim_brk2_cmp3出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp3入力は無効です。

1：tim_brk2_cmp3入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット2 **BK2CMP2E**：tim_brk2_cmp2イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp2を有効化します。tim_brk2_cmp2出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp2入力は無効です。

1：tim_brk2_cmp2入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット1 **BK2CMP1E**：tim_brk2_cmp1イネーブル

このビットは、タイマのtim_brk2入力に対してtim_brk2_cmp1を有効化します。tim_brk2_cmp1出力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：tim_brk2_cmp1入力は無効です。

1：tim_brk2_cmp1入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット0 **BK2INE**：TIMx_BKIN2入力イネーブル

このビットは、タイマのtim_brk2入力に対してTIMx_BKIN2オルタネート機能入力を有効化します。TIMx_BKIN2入力は、ほかのtim_brk2ソースとの「論理和」がとられます。

0：TIMx_BKIN2入力は無効です。

1：TIMx_BKIN2入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注： 製品固有の実装についてはセクション 54.3.2：TIM1/TIM8ピンおよび内部信号を参照してください。

54.6.29 DMA制御レジスタ（TIMx_DCR）（x = 1、8）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **DBSS[3:0]**：DMAバーストソース選択

このビットフィールドは、DMAバースト転送をトリガする割込みソースを定義します（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）。

0000：予約済み

0001：更新

0010：CC1

0011：CC2

0100：CC3

0101：CC4

0110：COM

0111：トリガ

その他：予約済み

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **DBL[4:0]**：DMAバースト長

この5bitのベクタは、転送数などDMA転送長（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：転送1回

00001：転送2回

00010：転送3回

.....

11010：転送26回

例：次の転送を考えます：DBL = 7バイトかつDBA = TIM2_CR1。

–DBL = 7バイトおよびDBA = TIM2_CR1が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1アドレス）+ DBA +（DMAインデックス）、ここでDMAインデックス = DBL

この例では、（TIMx_CR1アドレス）+ DBAに7バイトが追加され、データのコピー元／コピー先アドレスが与えられます。この場合、転送は、（TIMx_CR1アドレス）+ DBAから始めて、7つのレジスタに対して行われます。

DMAデータサイズの設定に応じて、いくつかのケースが想定されます。

–DMAデータサイズをハーフワードで設定した場合、7つのレジスタにそれぞれ16bitのデータが転送されます。

–DMA データサイズをバイトで設定した場合も、データは7つのレジスタに転送されます（最初のレジスタには最初のMSBバイトが含まれ、2番目のレジスタには最初のLSBバイトが含まれるなど、以降同様）。タイマへの転送で、DMAによって転送されるデータサイズを指定する必要もあります。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DBA[4:0]**：DMAベースアドレス

この5bitのベクタは、DMA転送のベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し/書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

.....

54.6.30 完全転送の TIMx DMAアドレス（TIMx_DMAR）（x = 1、8）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAB[31:0]**：DMAバーストアクセスレジスタ

DMARレジスタへの読出しまたは書込み動作は、（TIMx_CR1アドレス）+（DBA + DMAインデックス）x 4のアドレスにあるレジスタへのアクセスとなります。

ここで、TIMx_CR1アドレスは制御レジスタ1のアドレスであり、DBAはTIMx_DCRレジスタで設定されたDMAベースアドレスであり、DMAインデックスはDMA転送によって自動的に制御され、範囲は0からDBLです（DBLはTIMx_DCR内で設定）。

54.6.31 TIMx レジスタマップ

TIMx レジスタは、次の表のように、16bitアドレス可能レジスタとして配置されます。

表549.TIMxレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x004	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5	OIS4N	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値							0		0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x008	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
	リセット値							0	0			0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値									0	0	0	0						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値									0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x018	**TIMx_CCMR1** 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	**TIMx_CCMR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F [3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	CC4NP	CC4NE	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	**TIMx_CNT**	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表549.TIMxレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x030	**TIMx_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	**TIMx_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	**TIMx_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	**TIMx_CCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	**TIMx_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	**TIMx_BDTR**	Res.	Res.	BK2BID	BKBID	BK2DSRM	BKDSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	**TIMx_CCR5**	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5 [19:0]																			
	リセット値	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	**TIMx_CCR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	**TIMx_CCMR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M [2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		

表549.TIMxレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x054	**TIMx_DTR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x058	**TIMx_ECR**	Res.	Res.	Res.	Res.	Res.	PWPRSC[2:0]			PW[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS[1:0]		FIDX	IBLK[1:0]		IDIR[1:0]		IE
	リセット値						0	0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x05C	**TIMx_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]		
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x060	**TIMx_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res ETRSEL [3:0]				BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x064	**TIMx_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]			Res.	Res.	BK2CMP4P	BK2CMP3P	BK2CMP2P	BK2CMP1P	BK2INP	BK2CMP8E	BK2CMP7E	BK2CMP6E	BK2CMP5E	BK2CMP4E	BK2CMP3E	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値														0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x068.. 0x3D8	予約済み	Res.																															
0x3DC	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]				Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値													0	0	0	0				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIMx_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては136ページのセクション 2.3を参照してください。

55 汎用タイマ（TIM2/TIM3/TIM4/TIM5）

55.1 TIM2/TIM3/TIM4/TIM5 の概要

この汎用タイマは、プログラマブルなプリスケーラによって駆動される16bitまたは32bitの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較とPWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラとRCCクロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、セクション 55.4.23：タイマの同期に示すように、相互に同期させることができます。

55.2 TIM2/TIM3/TIM4/TIM5 の主な特徴

汎用TIMxタイマの主な機能は、次のとおりです。

- 16bitまたは32bitのアップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1から65535の間で分周する16bitプログラム可能プリスケーラ。
- 次の機能を持つ、最大4つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割込み／DMA生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

55.3 TIM2/TIM3/TIM4/TIM5 の実装

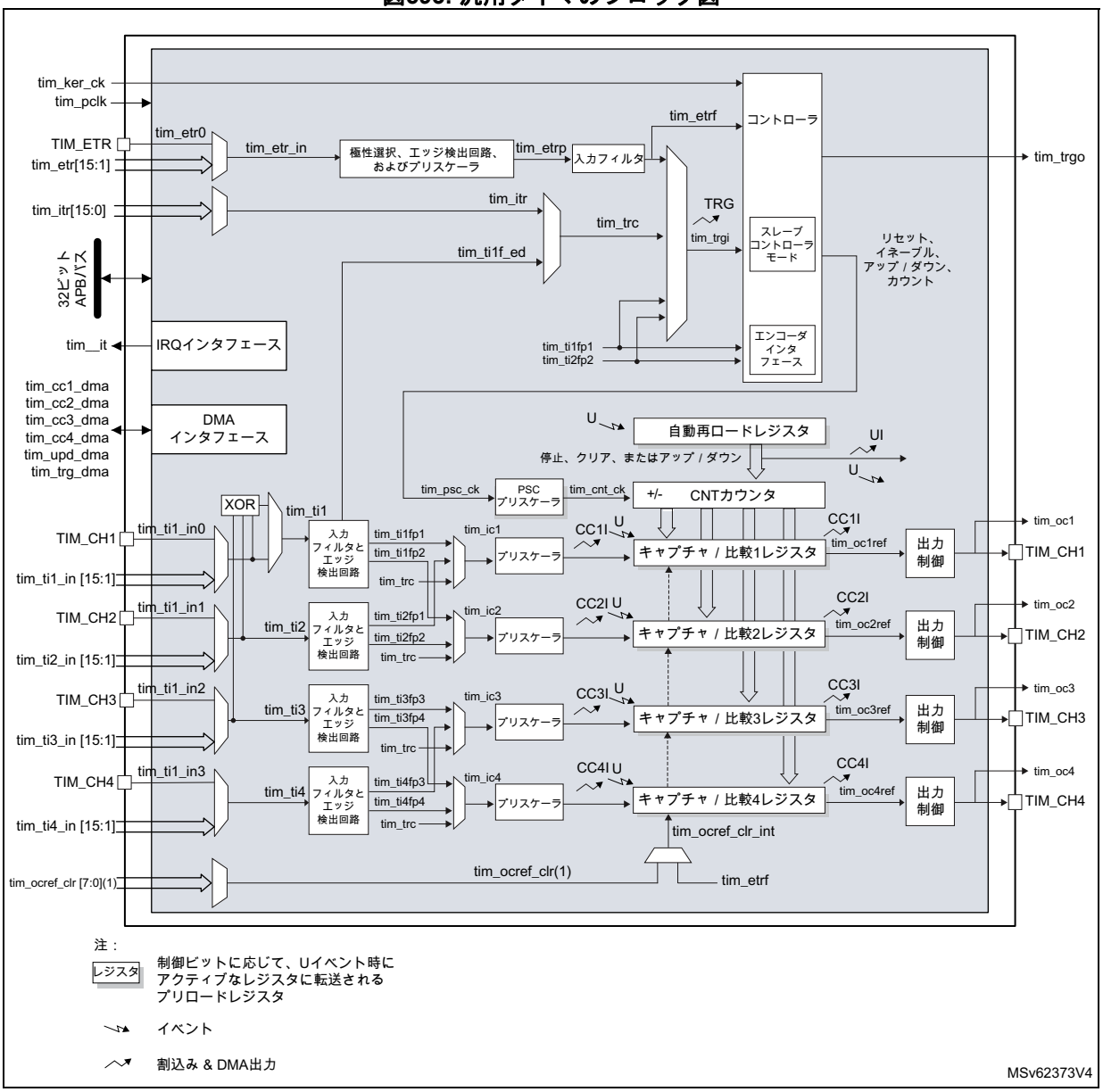
表550.STM32U5シリーズ汎用タイマ

タイマインスタンス	TIM2	TIM3	TIM4	TIM5
分解能	32bit	32bit	32bit	32bit
OCREFクリア選択ソース	可能 tim_etrf tim_ocref_clr[7:0]	あり tim_etrf tim_ocref_clr[7:0]	あり tim_etrf tim_ocref_clr[7:0]	あり tim_etrf tim_ocref_clr[7:0]

55.4 TIM2/TIM3/TIM4/TIM5機能詳細

55.4.1 ブロック図

図593. 汎用タイマのブロック図



1. この機能は一部のタイマでは利用できません（セクション 55.3：TIM2/TIM3/TIM4/TIM5 の実装を参照）。

55.4.2 TIM2/TIM3/TIM4/TIM5ピンおよび内部信号

このセクションの表 551および表 552は、TIMの入力および出力の一覧です。

表551.TIMの入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2 TIM_CH3 TIM_CH4	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、またはPWMで使用できます。 TIM_CH1とTIM_CH2は、外部クロック（tim_ker_ckクロックの1/4未満）、外部トリガ、および直交エンコーダの入力としても使用できます。 TIM_CH1、TIM_CH2、およびTIM_CH3は、デジタルホール効果センサとのインタフェースに使用できます。
TIM_ETR	入力	外部トリガ入力。この入力は、外部トリガまたは外部クロックソースとして使用できます。tim_etr_inプリスケーラが使用されている場合、この入力はtim_ker_ckよりも高い周波数のクロックを受信できます。

表552.TIM内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0] tim_ti3_in [15:0] tim_ti4_in [15:0]	入力	内部タイマ入力バス。tim_ti1_in[15:0] および tim_ti2_in[15:0] の入力は、キャプチャ用、外部クロック（tim_ker_ckクロックの1/4未満）、直交エンコーダ信号用として使用できます。
tim_etr[15:0]	入力	外部トリガ内部入力バス。これらの入力は、トリガ、外部クロックとして、またはハードウェアのサイクルごと（サイクルバイサイクル）のパルス幅制御に使用できます。tim_etr_inプリスケーラが使用されている場合、これらの入力はtim_ker_ckよりも高い周波数のクロックを受信できます。
tim_itr[15:0]	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ckクロックの1/4未満）として使用できます。
tim_trgo	出力	内部トリガ出力。このトリガは他のオンチップペリフェラルをトリガできます。
tim_ocref_clr[7:0]	入力	タイマのtim_ocref_clr入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim_ocxref信号をクリアするために使用できます。
tim_pclk	入力	タイマAPBクロック。
tim_ker_ck	入力	タイマカーネルクロック
tim_it	出力	キャプチャ／比較、更新、およびブレークトリガリクエストを取りまとめるグローバルタイマ割込み。

表552.TIM内部入力／出力信号 （続き）

内部信号名	信号タイプ	説明
tim_cc1_dma tim_cc2_dma tim_cc3_dma tim_cc4_dma	出力	タイマキャプチャ／比較1～4 DMAリクエスト。
tim_upd_dma	出力	タイマ更新DMAリクエスト。
tim_trg_dma	出力	タイマトリガDMAリクエスト。

および に、tim_ti[4:1] 入力マルチプレクサに接続されるソースを列挙します。

表553.tim_ti1入力マルチプレクサへの相互接続

tim_ti1入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti1_in0	TIM2_CH1	TIM3_CH1	TIM4_CH1	TIM5_CH1
tim_ti1_in1	comp1_out	comp1_out	comp1_out	LSI
tim_ti1_in2	comp2_out(1)	comp2_out（1）	comp2_out（1）	LSE
tim_ti1_in3	予約済み			rtc_wut_trg
tim_ti1_in4				comp1_out
tim_ti1_in5				comp2_out（1）
tim_ti1_in[15:6]	予約済み			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表554.tim_ti2入力マルチプレクサへの相互接続

tim_ti2入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti2_in0	TIM2_CH2	TIM3_CH2	TIM4_CH2	TIM5_CH2
tim_ti2_in1	comp1_out	comp1_out	comp1_out	comp1_out
tim_ti2_in2	comp2_out(1)	comp2_out（1）	comp2_out（1）	comp2_out（1）
tim_ti2_in [15:3]	予約済み			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表555.tim_ti3入力マルチプレクサへの相互接続

tim_ti3入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti3_in0	TIM2_CH3	TIM3_CH3	TIM4_CH3	TIM5_CH3
tim_ti3_in [15:1]	予約済み			

表556.tim_ti4入力マルチプレクサへの相互接続

tim_ti4入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti4_in0	TIM2_CH4	TIM3_CH4	TIM4_CH4	TIM5_CH4
tim_ti4_in1	comp1_out	予約済み		
tim_ti4_in2	comp2_out(1)			
tim_ti4_in [15:3]	予約済み			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

以下の表に、tim_itr入力マルチプレクサに接続される内部ソースを列挙します。

表557.TIMx内部トリガ接続

TIMx	TIM2	TIM3	TIM4	TIM5
tim_itr0	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo
tim_itr1	予約済み	tim2_trgo	tim2_trgo	tim2_trgo
tim_itr2	tim3_trgo	予約済み	tim3_trgo	tim3_trgo
tim_itr3	tim4_trgo	tim4_trgo	予約済み	tim4_trgo
tim_itr4	tim5_trgo	tim5_trgo	tim5_trgo	予約済み
tim_itr5	tim8_trgo	tim8_trgo	tim8_trgo	tim8_trgo
tim_itr6	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo
tim_itr7	tim16_oc1	tim16_oc1	tim16_oc1	tim16_oc1
tim_itr8	tim17_oc1	tim17_oc1	tim17_oc1	tim17_oc1
tim_itr9	予約済み	予約済み		
tim_itr10				
tim_itr11	OTG_FS/OTG_HSSOF			
tim_itr[15:12]	予約済み			

以下の表に、tim_etr入力マルチプレクサに接続される内部ソースを列挙します。

表558.STM32U535/545/575/585向けのtim_etr入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て			
	TIM2	TIM3	TIM4	TIM5
tim_etr0	TIM2_ETR	TIM3_ETR	TIM4_ETR	TIM5_ETR
tim_etr1	comp1_out	comp1_out	comp1_out	comp1_out
tim_etr2	comp2_out(1)	comp2_out (1)	comp2_out (1)	comp2_out (1)
tim_etr3	MSIK	MSIK	MSIK	MSIK
tim_etr4	HSI	HSI	HSI	HSI
tim_etr5	MSIS	MSIS	MSIS	MSIS

表558.STM32U535/545/575/585向けのtim_etr入力マルチプレクサへの相互接続 （続き）

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て			
	TIM2	TIM3	TIM4	TIM5
tim_etr6	予約済み			
tim_etr7				
tim_etr8	TIM3_ETR	TIM2_ETR	TIM3_ETR	TIM2_ETR
tim_etr9	TIM4_ETR	TIM4_ETR	TIM5_ETR	TIM3_ETR
tim_etr10	TIM5_ETR	予約済み	予約済み	
tim_etr11	LSE	adc1_awd1		
tim_etr12	予約済み	adc1_awd2		
tim_etr13		adc1_awd3		
tim_etr[15:14]	予約済み			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表559.STM32U59x/5Ax/5Fx/5Gx向けのtim_etr入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て			
	TIM2	TIM3	TIM4	TIM5
tim_etr0	TIM2_ETR	TIM3_ETR	TIM4_ETR	TIM5_ETR
tim_etr1	comp1_out	comp1_out	comp1_out	comp1_out
tim_etr2	comp2_out	comp2_out	comp2_out	comp2_out
tim_etr3	MSIK	MSIK	MSIK	MSIK
tim_etr4	HSI	HSI	HSI	HSI
tim_etr5	MSIS	MSIS	MSIS	MSIS
tim_etr6	DCMI_VSYNC	DCMI_VSYNC	DCMI_VSYNC	DCMI_VSYNC
tim_etr7	LTDC_VSYNC	LTDC_VSYNC	LTDC_VSYNC	LTDC_VSYNC
tim_etr8	TIM3_ETR	TIM2_ETR	TIM3_ETR	TIM2_ETR
tim_etr9	TIM4_ETR	TIM4_ETR	TIM5_ETR	TIM3_ETR
tim_etr10	TIM5_ETR	DSI_TE	DSI_TE	DSI_TE
tim_etr11	LSE	adc1_awd1	adc2_awd1	予約済み
tim_etr12	予約済み	adc1_awd2	adc2_awd2	
tim_etr13	DSI_TE	adc1_awd3	adc2_awd3	
tim_etr14	DCMI_HSYNC	DCMI_HSYNC	DCMI_HSYNC	DCMI_HSYNC
tim_etr15	LTDC_HSYNC	LTDC_HSYNC	LTDC_HSYNC	LTDC_HSYNC

以下の表に、tim_ocref_clr入力マルチプレクサに接続される内部ソースを列挙します。

表560.tim_ocref_clr入力マルチプレクサへの相互接続

タイマtim_ocref_clr信号	タイマtim_ocref_clr信号割当て			
	TIM2	TIM3	TIM4	TIM5
tim_ocref_clr0	comp1_out	comp1_out	comp1_out	comp1_out
tim_ocref_clr1	comp2_out(1)	comp2_out（1）	comp2_out（1）	comp2_out（1）
tim_ocref_clr[7:2]	予約済み			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

55.4.3 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ16bit／32bitカウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）:
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1レジスタのUDISビットが0の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力tim_cnt_ckから供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号CNT_ENは、CENの1クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を1から65536の間の値で分周することができます。16ビット／32bitレジスタ（TIMx_PSCレジスタ）を使って制御される16bitカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図594と図595に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図594. プリスケーラ分周比が1から2に変化したときのカウンタのタイミング図

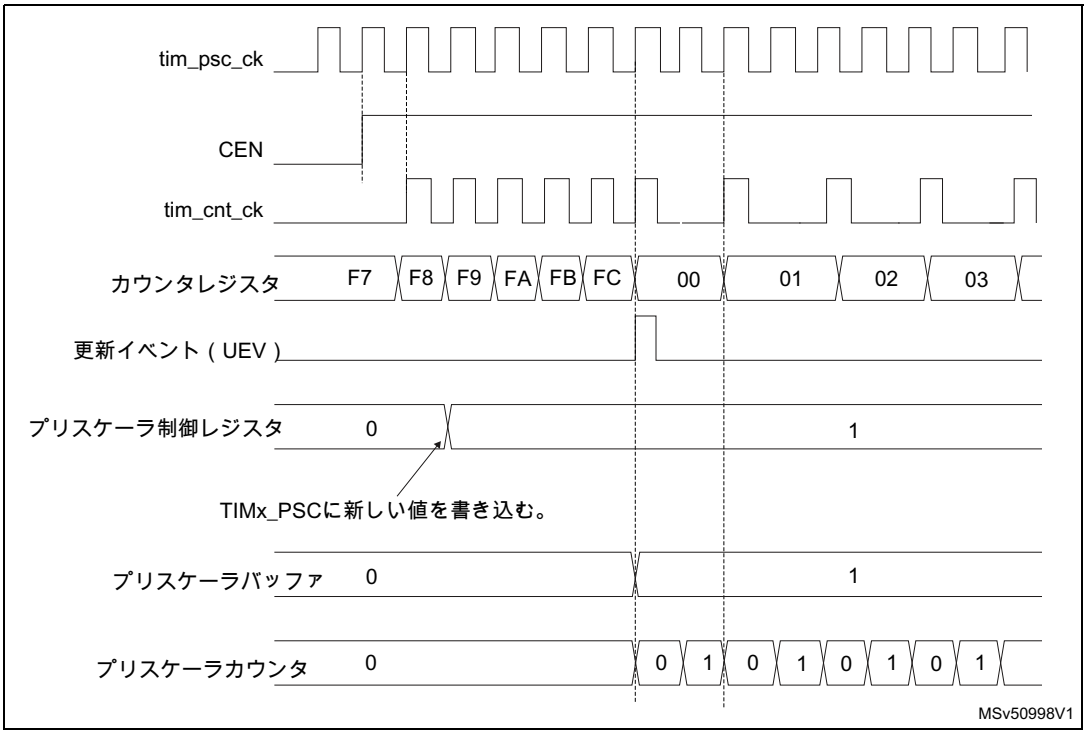
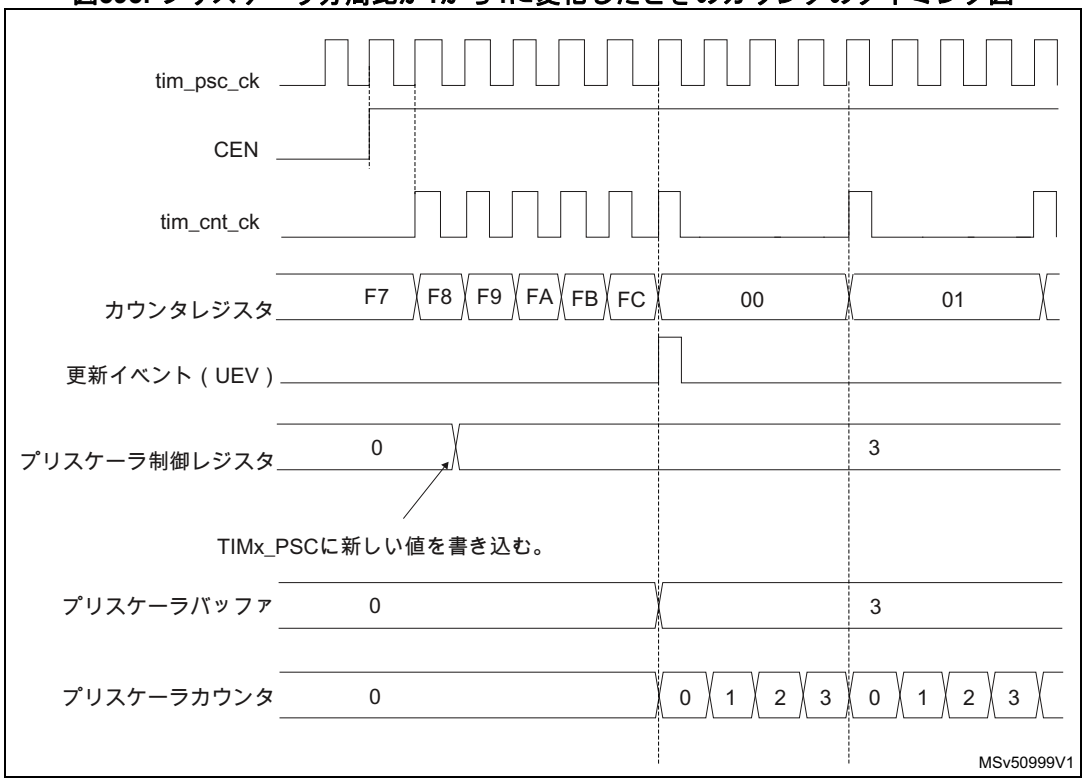


図595. プリスケーラ分周比が1から4に変化したときのカウンタのタイミング図



55.4.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは0から自動再ロード値（TIMx_ARRレジスタの内容）までカウントし、0からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることで生成できます。

UEVイベントは、TIMx_CR1レジスタのUDISビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは0からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図596. 内部クロック分周比が1の場合のカウンタのタイミング図

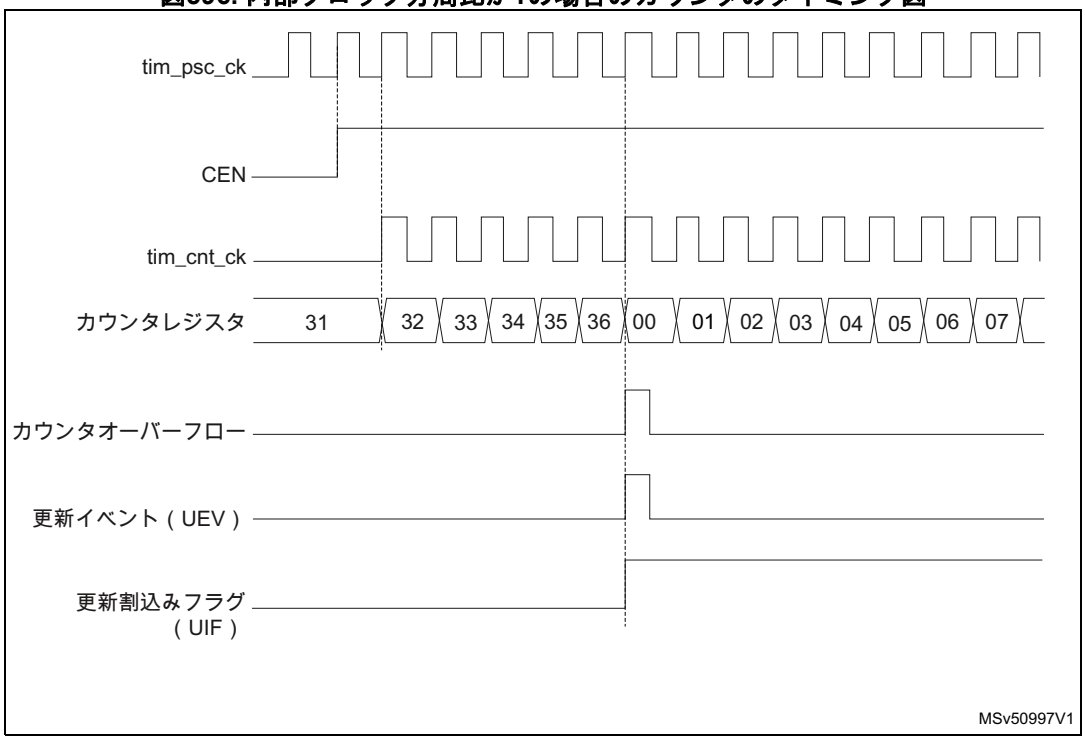


図597. 内部クロック分周比が2の場合のカウンタのタイミング図

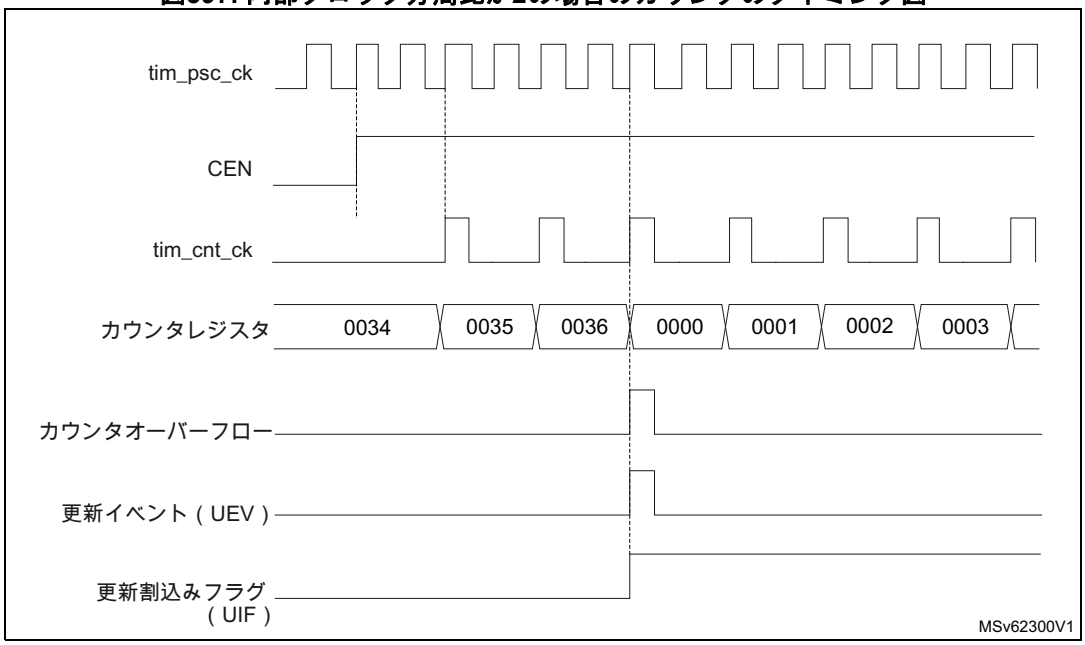


図598. 内部クロック分周比が4の場合のカウンタのタイミング図

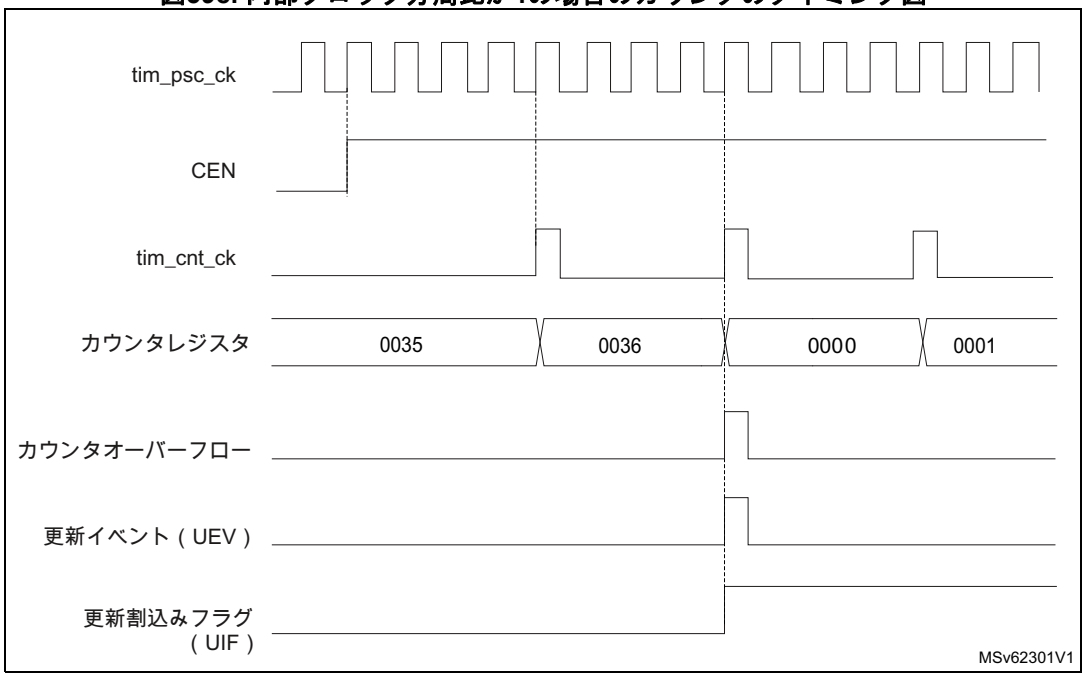


図599. 内部クロック分周比がNの場合のカウンタのタイミング図

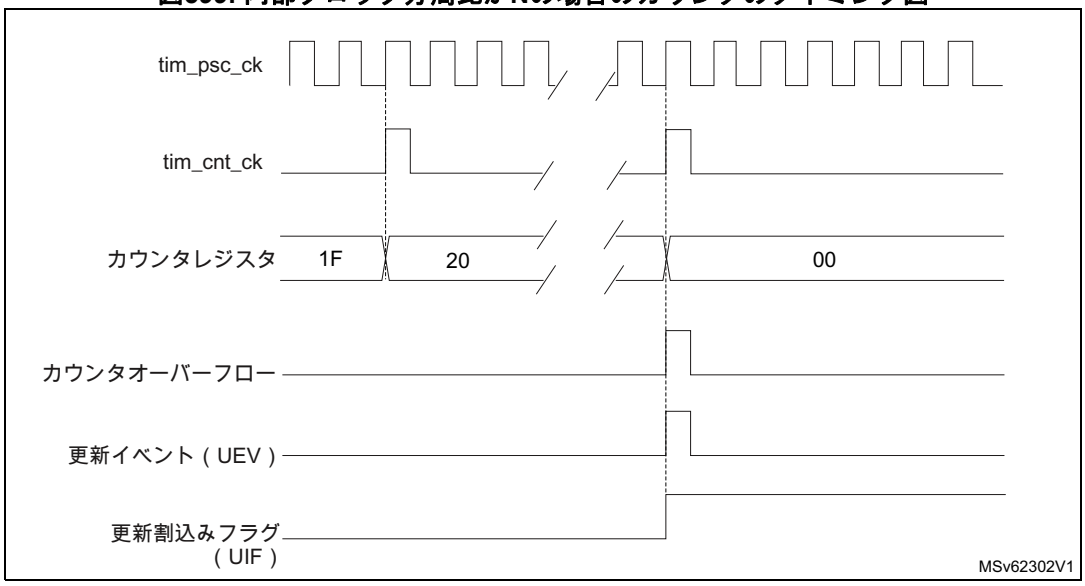


図600. ARPE=0（TIMx_ARRはプリロードされない）の場合の更新イベント時のカウンタのタイミング図

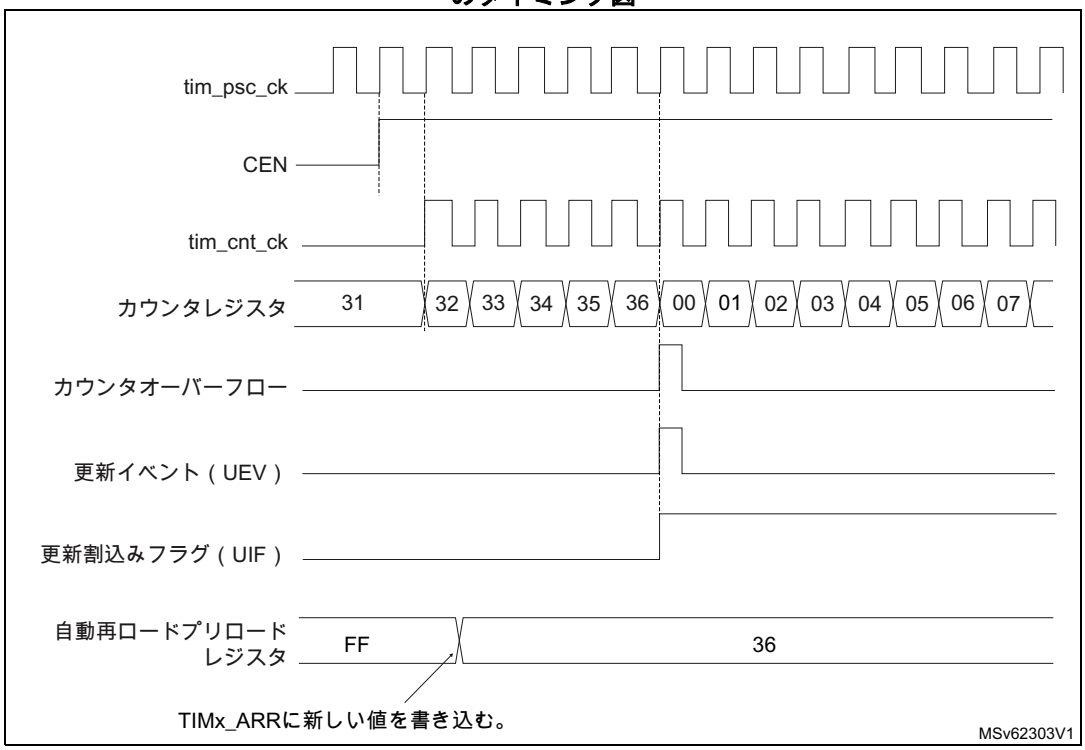
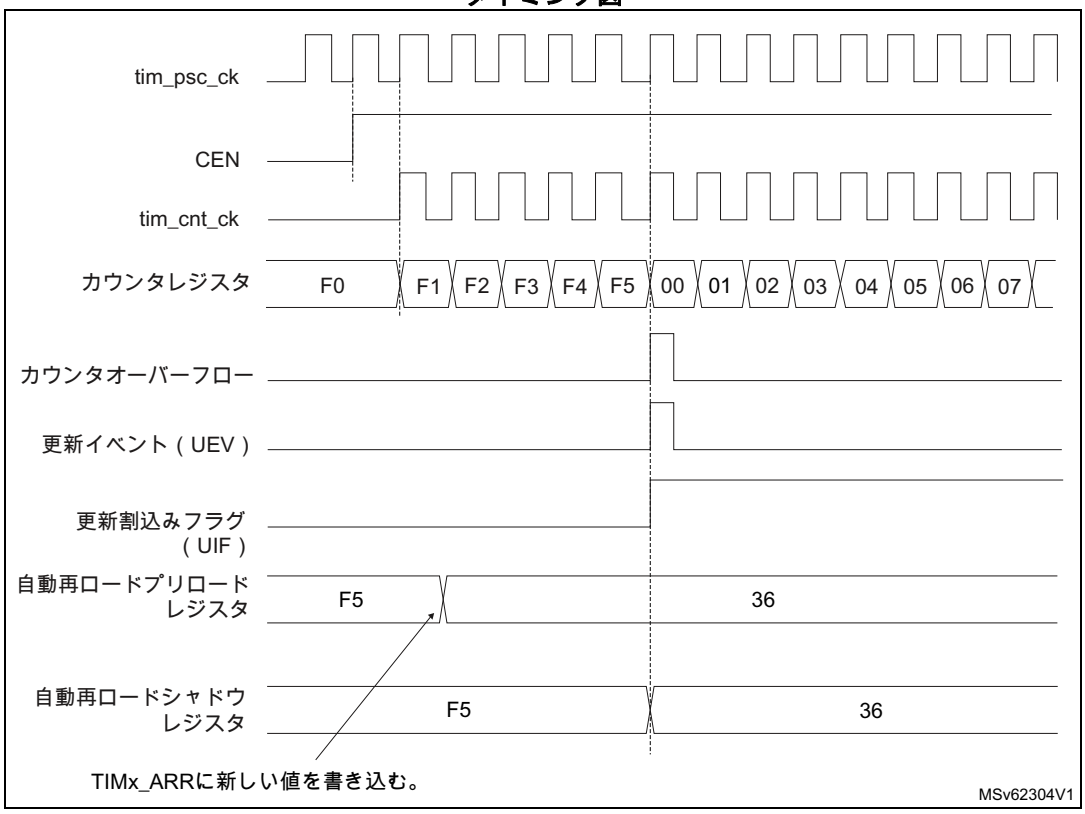


図601. ARPE=1（TIMx_ARRはプリロードされる）のときの更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARRレジスタの内容）から0までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることにより生成できます。

UEV更新イベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは0からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARRレジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図602. 内部クロック分周比が1の場合のカウンタのタイミング図

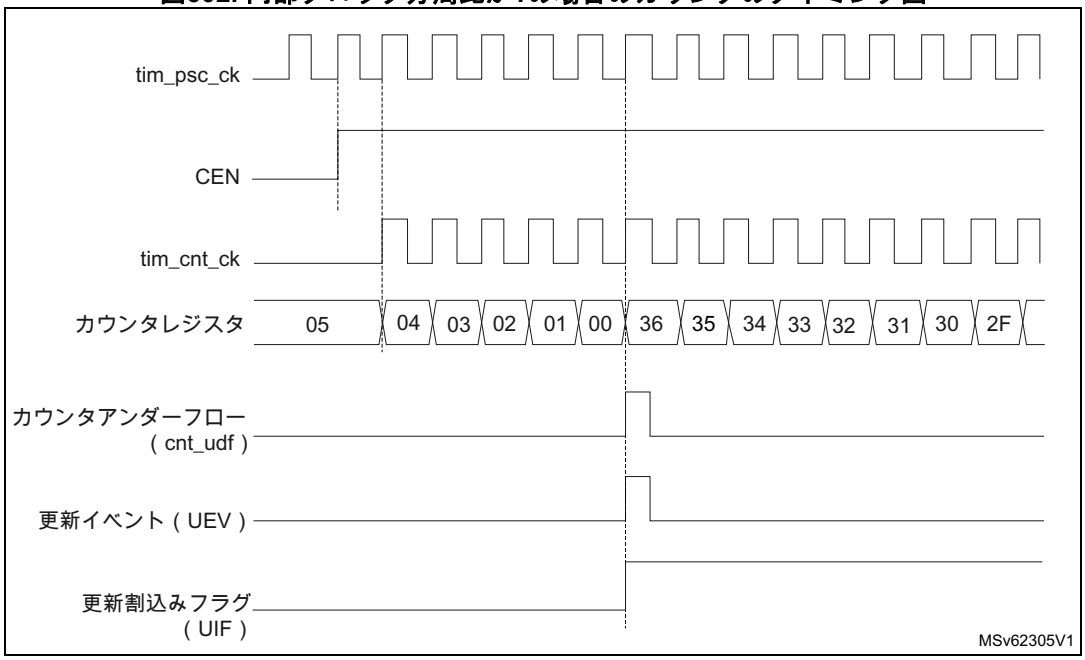


図603. 内部クロック分周比が2の場合のカウンタのタイミング図

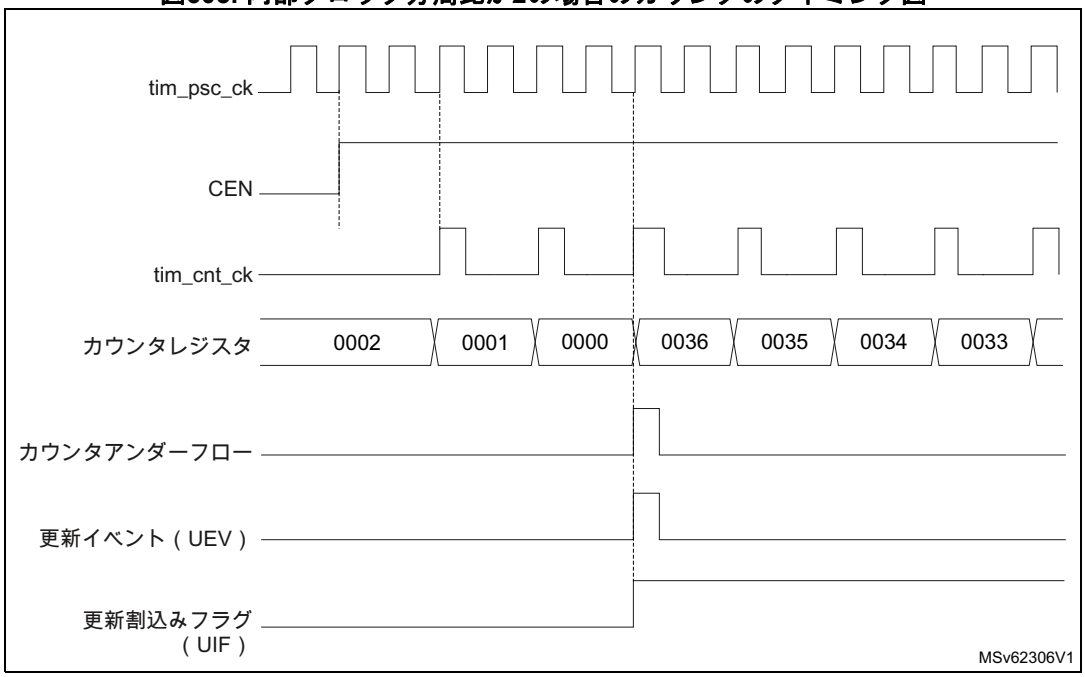


図604. 内部クロック分周比が4の場合のカウンタのタイミング図

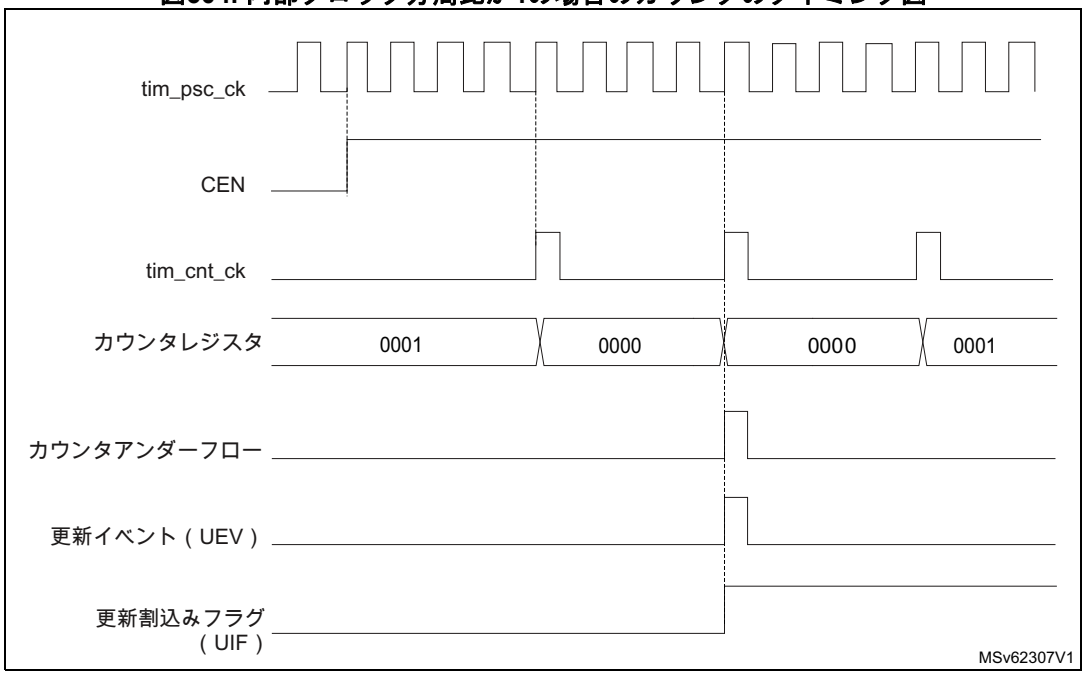


図605. 内部クロック分周比がNの場合のカウンタのタイミング図

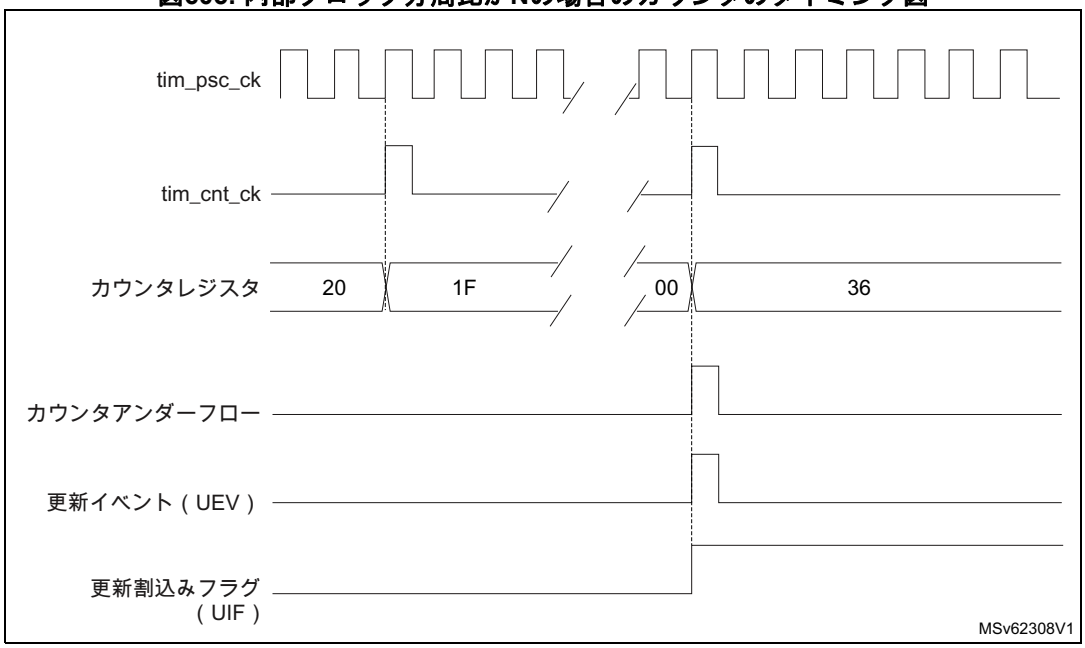
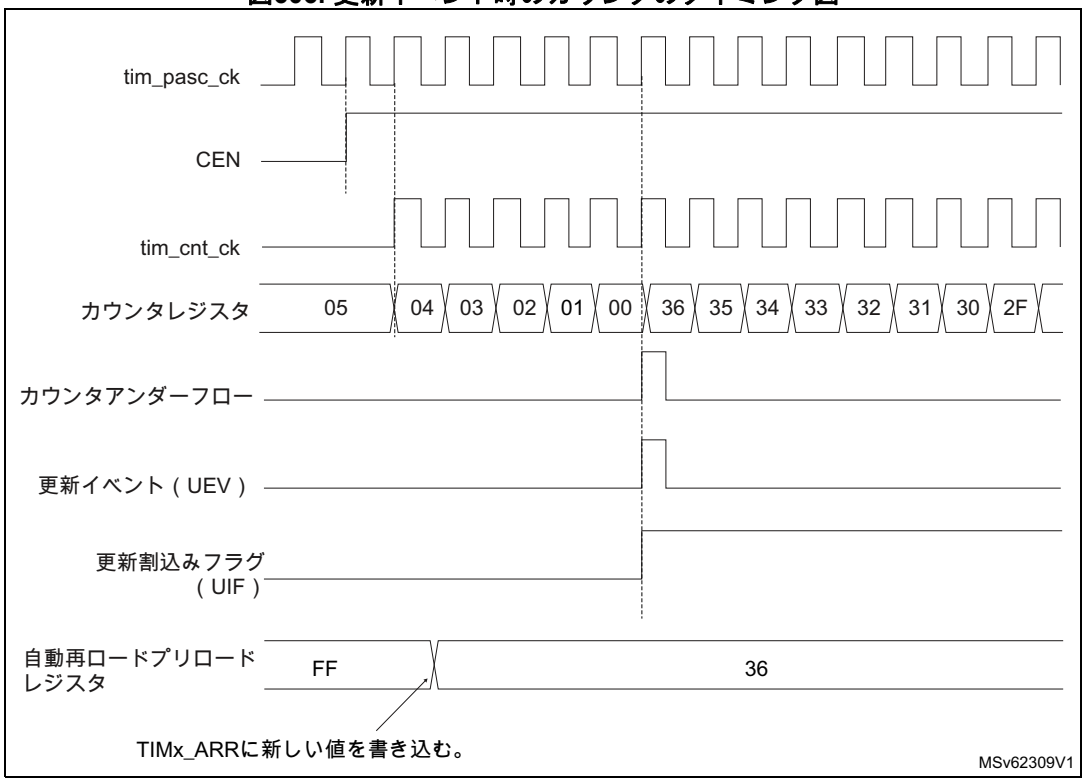


図606. 更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは0から自動再ロード値（TIMx_ARRレジスタの内容）－1までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から1までカウントして、カウンタアンダーフローイベントを生成します。その後、0からカウントをリスタートします。

センターアラインモードは、TIMx_CR1レジスタのCMSビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード3、CMS=11）にセットされます。

このモードでは、方向ビット（TIMx_CR1レジスタのDIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは0からカウントをリスタートします。

UEV更新イベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがっ

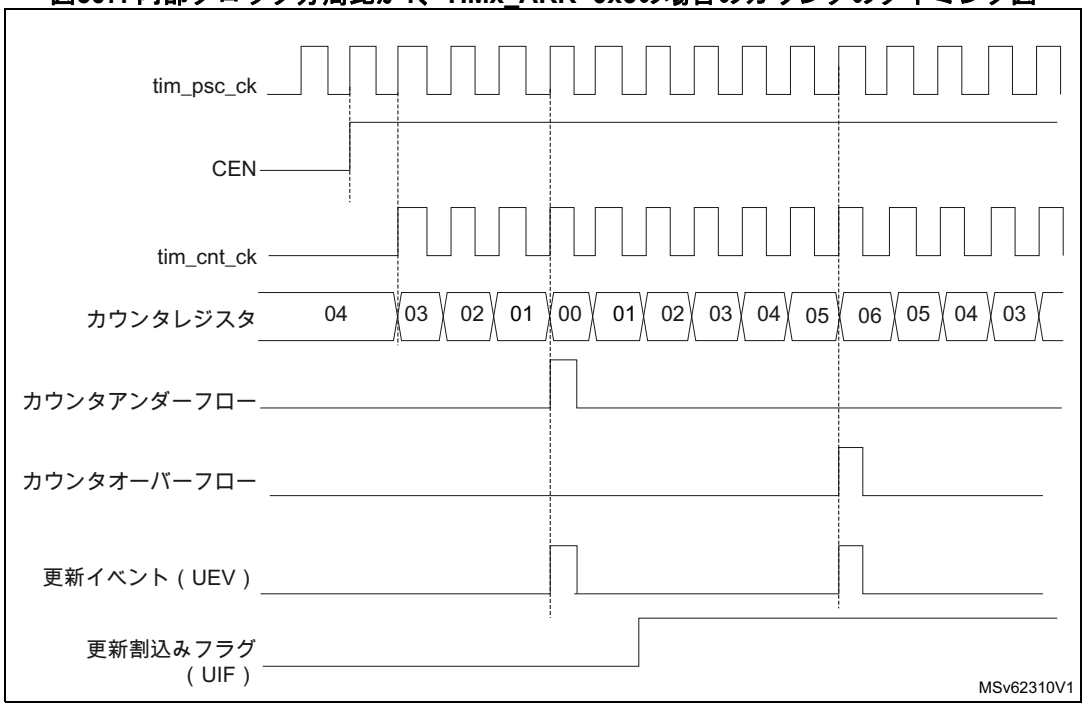
て、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARRレジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図607. 内部クロック分周比が1、TIMx_ARR=0x6の場合のカウンタのタイミング図



1. ここでは、センターアラインモード1が使用されています（詳細については、セクション 55.5.1： 2291ページのTIMx制御レジスタ1（TIMx_CR1）（x = 2〜5）を参照）。

図608. 内部クロック分周比が2の場合のカウンタのタイミング図

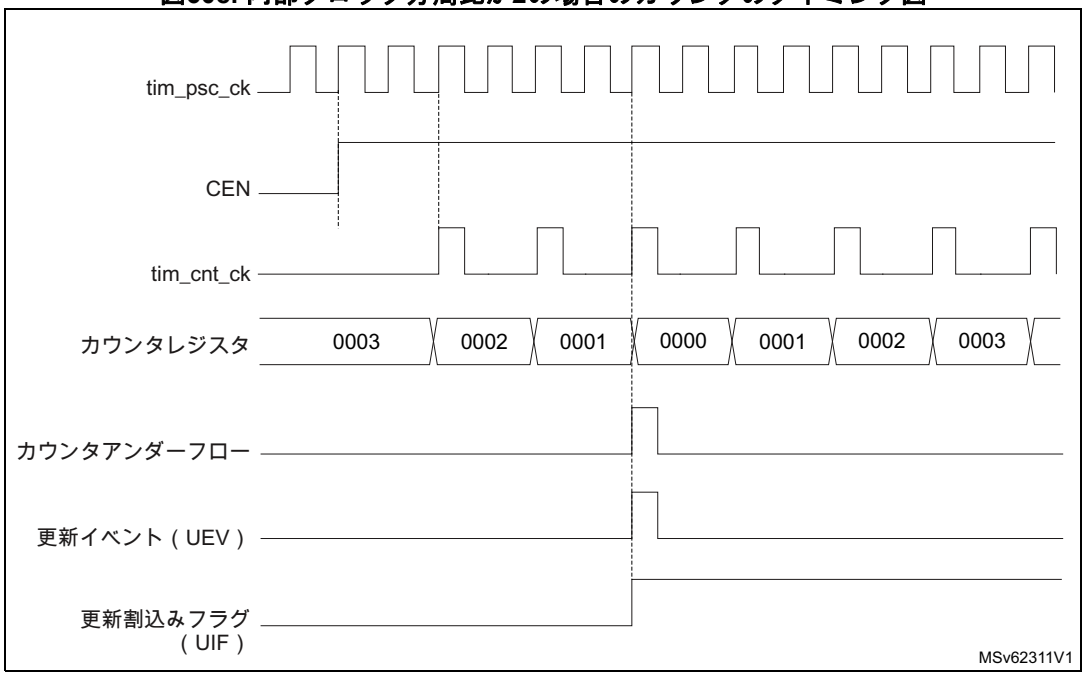
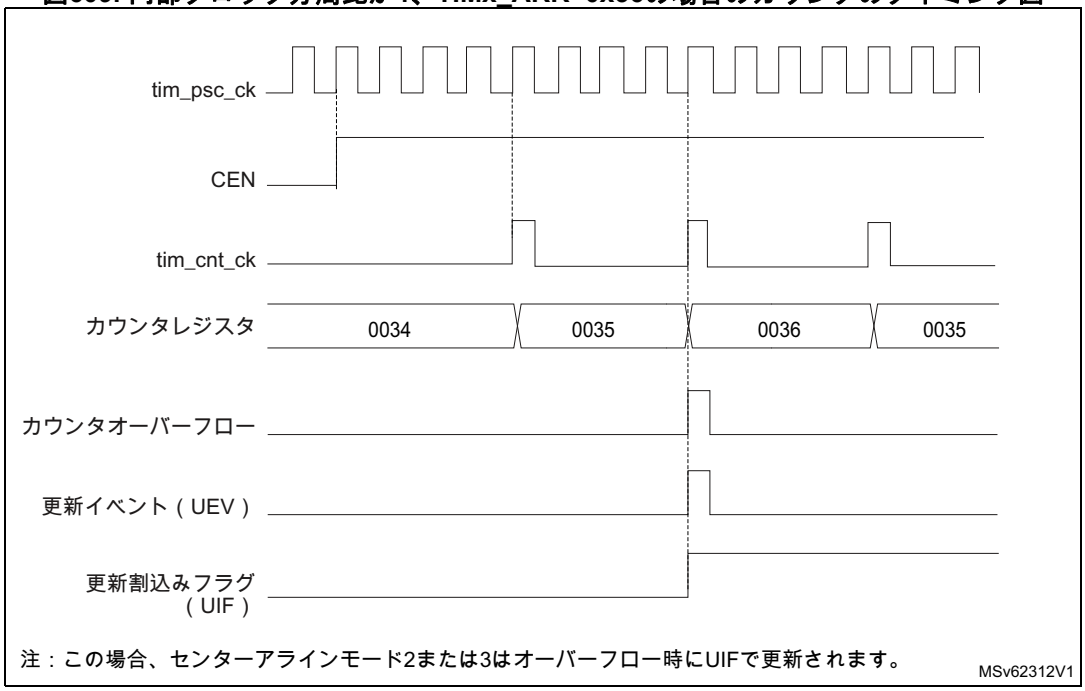


図609. 内部クロック分周比が4、TIMx_ARR=0x36の場合のカウンタのタイミング図



1. センターアラインモード2または3が使用され、オーバーフロー時にUIFがセットされます。

図610. 内部クロック分周比がNの場合のカウンタのタイミング図

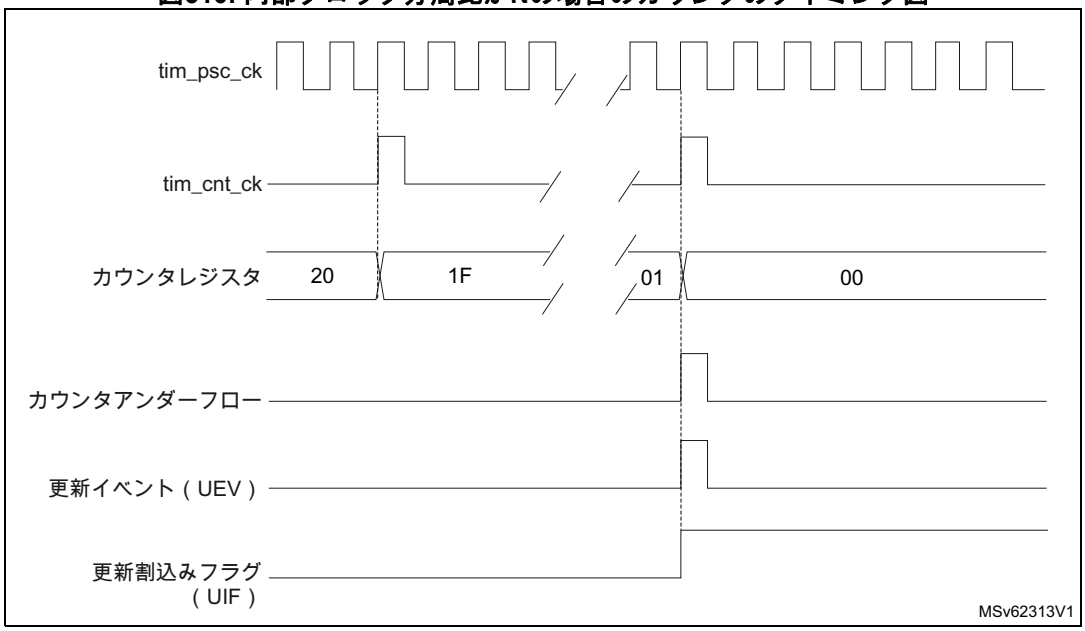


図611. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図

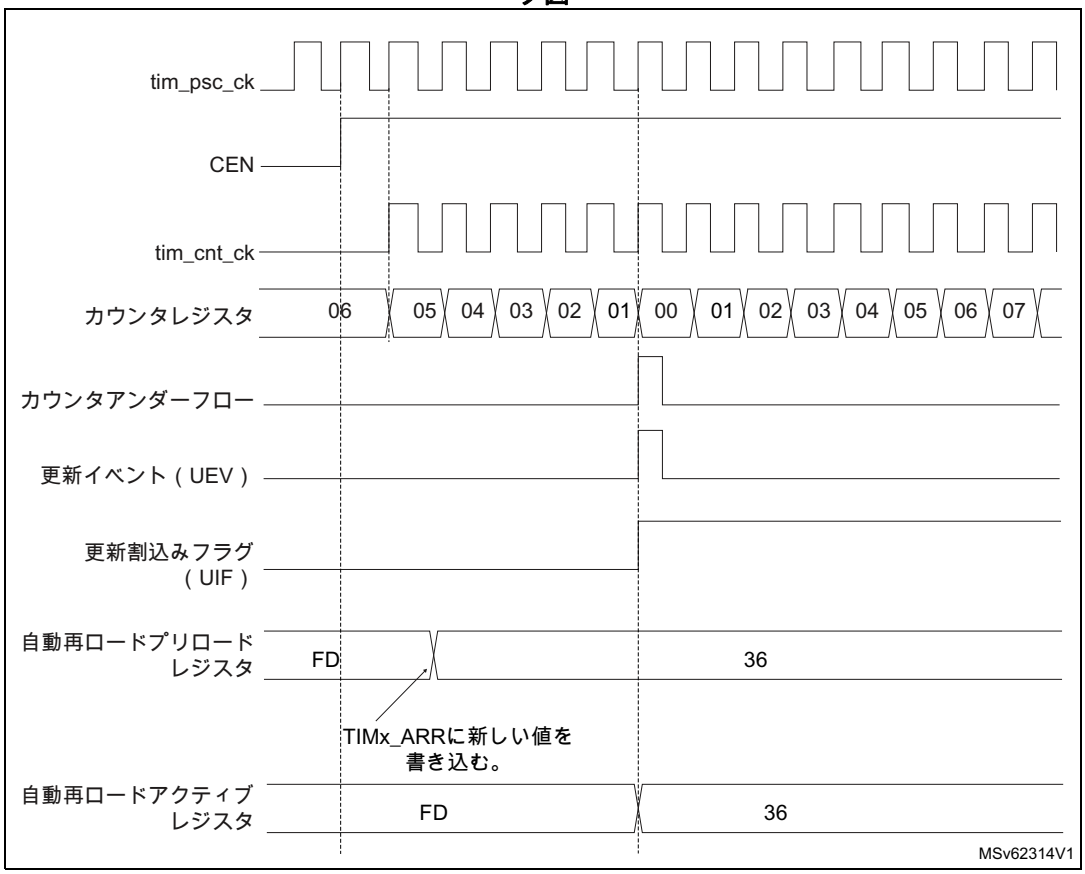
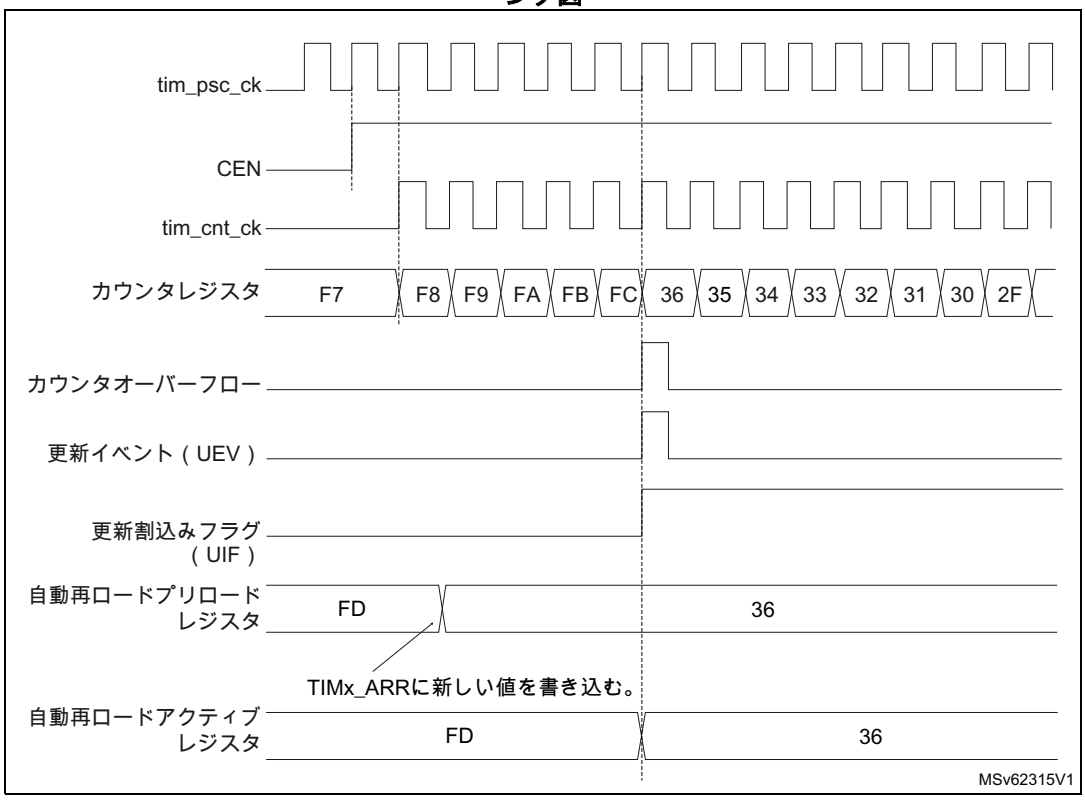


図612. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



55.4.5 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

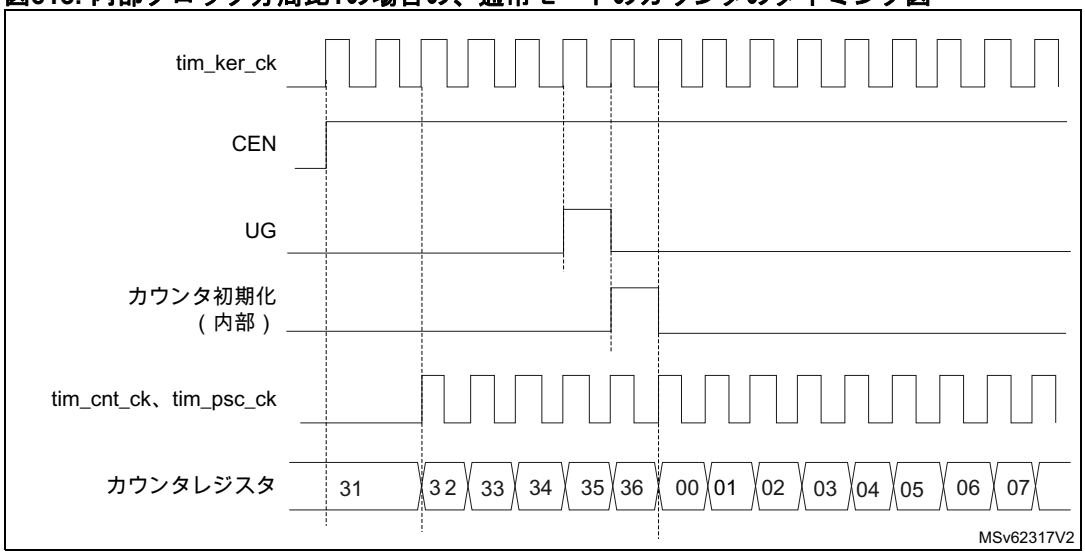
- 内部クロック（tim_ker_ck）
- 外部クロックモード1：外部入力ピン（tim_ti1またはtim_ti2）
- 外部クロックモード2：外部トリガ入力（tim_etr_in）
- 内部トリガ入力（tim_itr）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、タイマ1がタイマ2のプリスケーラとして機能するように設定できます。詳細については、2284ページのタイマを別のタイマのプリスケーラとして使用するを参照してください。

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（TIMx_SMCRレジスタのSMS=000）、CEN、DIRビット（TIMx_CR1レジスタ）とUGビット（TIMx_EGRレジスタ）が実際の制御ビットであり、ソフトウェアでのみ変更できます（自動的にクリアされたままのUGビットを除きます）。CENビットに1が書き込まれるとすぐに、プリスケーラには内部クロックtim_ker_ckが供給されます。

図613 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

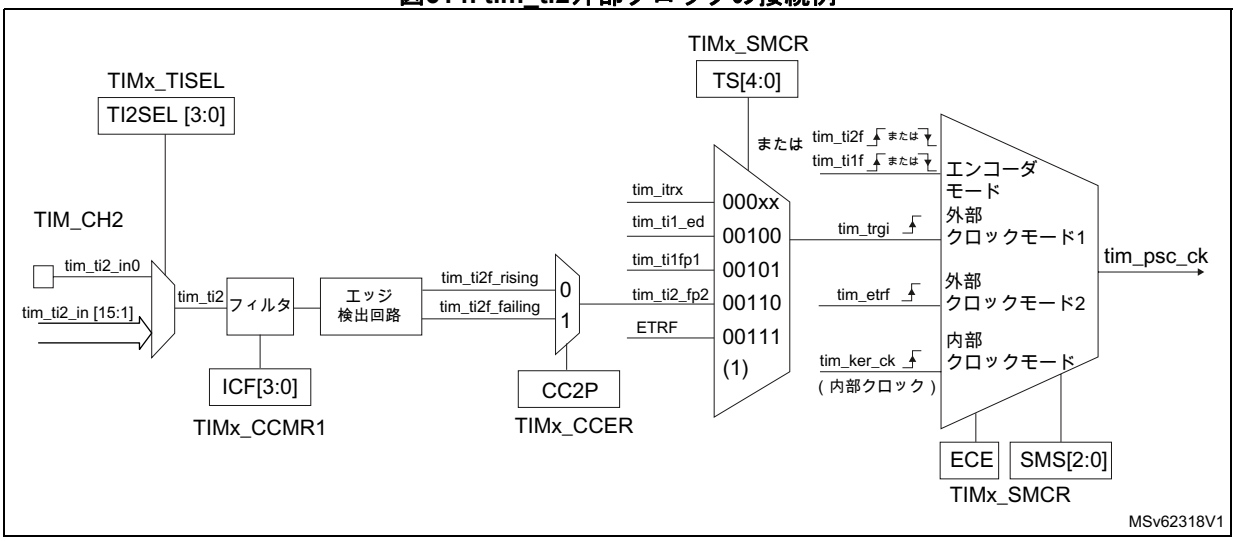
図613. 内部クロック分周比1の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード1

このモードはTIMx_SMCRレジスタのSMS=111のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図614. tim_ti2外部クロックの接続例



1. 01000から11111の範囲のコード：tim_itr[15:0]。

たとえば、tim_ti2入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

たとえば、tim_ti2入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_TISELレジスタのTI2SEL[3:0] ビットで、適切なtim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1レジスタにCC2S=01を書き込むことによって、チャネル2がtim_ti2入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1レジスタのIC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000にしておきます）。

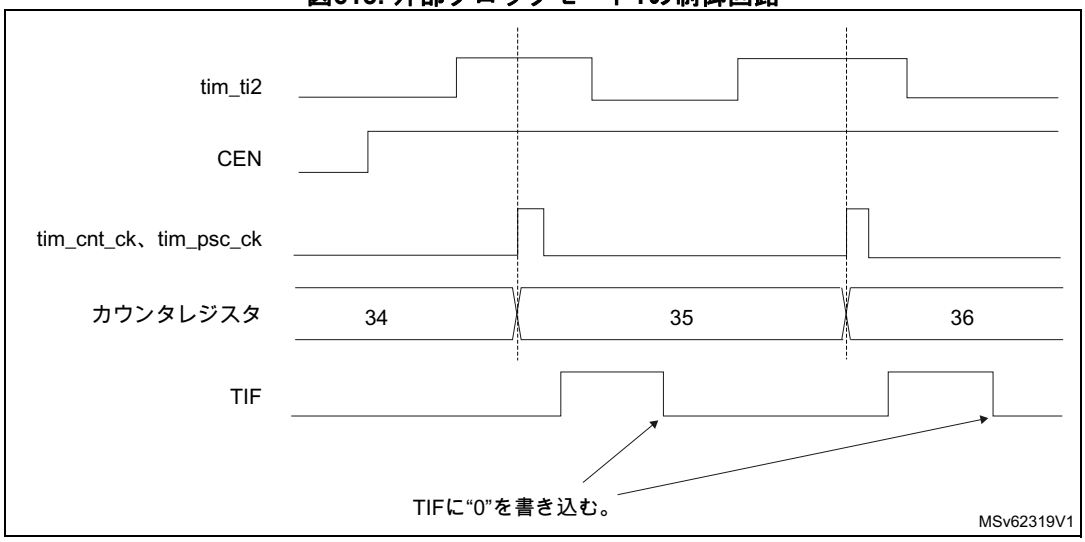
注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

4. CC2P=0、CC2NP=0、およびCC2NP=0をTIMx_CCERレジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCRレジスタにSMS=111を書き込むことによって、タイマを外部クロックモード1に設定します。
6. TIMx_SMCRレジスタにTS=00110を書き込むことによって、入力ソースとしてtim_ti2を選択します。
7. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを有効にします。

tim_ti2の立ち上がりエッジが発生すると、カウンタは1カウントを行い、TIFフラグがセットされます。

tim_ti2の立ち上がりエッジから実際のカウンタクロックまでの間には、tim_ti2入力の再同期回路による遅延があります。

図615. 外部クロックモード1の制御回路



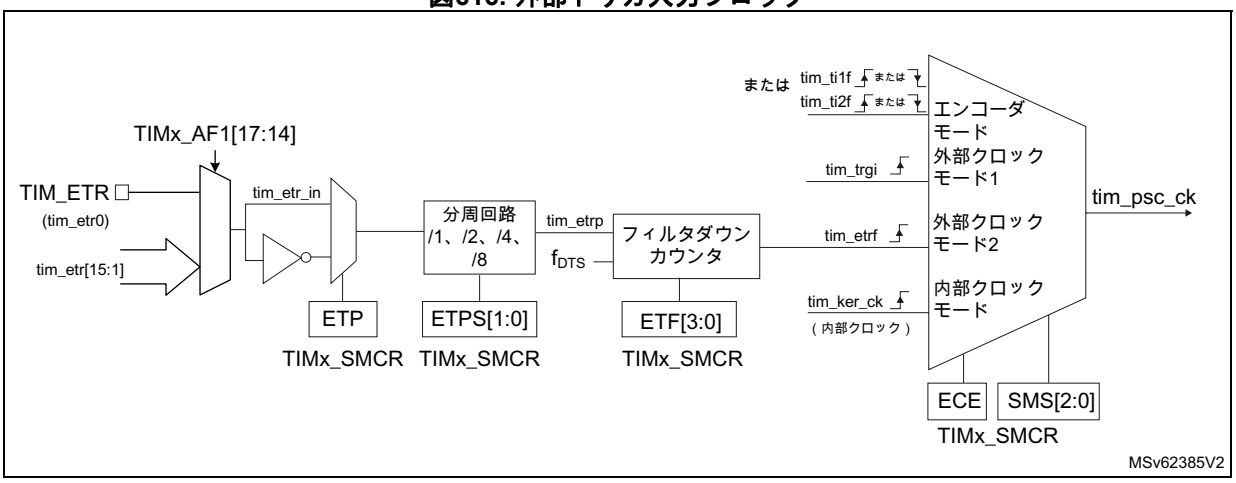
外部クロックソースモード2

このモードは、TIMx_SMCRレジスタのECE=1を書き込むことによって選択されます。

カウンタは、外部トリガ入力tim_etr_inの立ち上がりまたは立ち下がりエッジごとにカウントできます。

図616に、外部トリガ入力ブロックの概要を示します。

図616. 外部トリガ入力ブロック



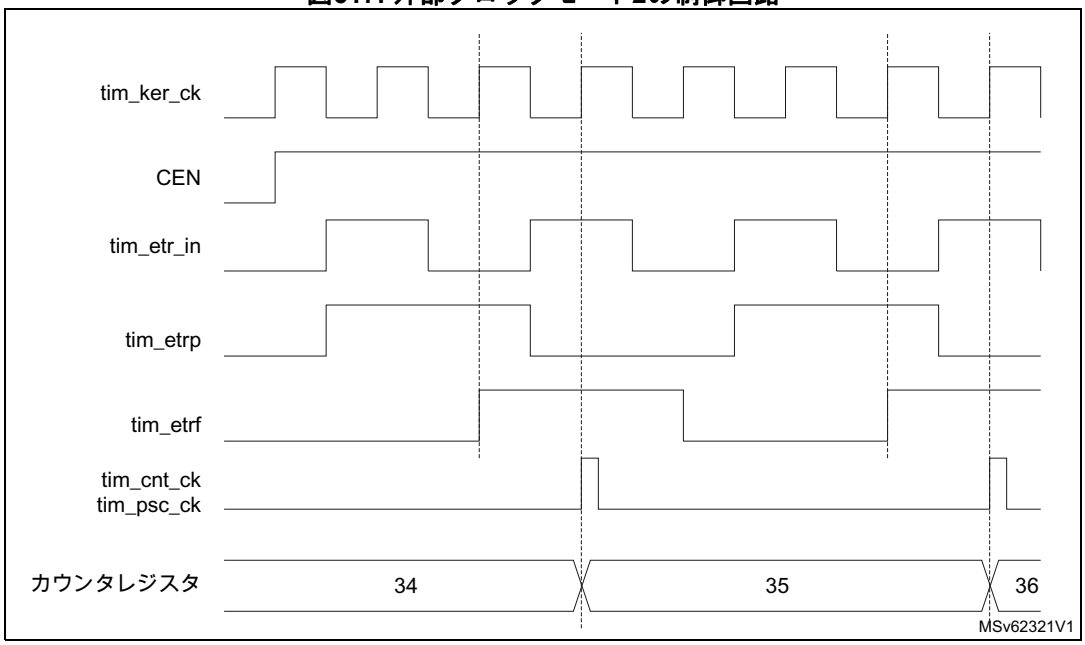
たとえば、tim_etr_inの2回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. TIMx_AF1レジスタのETRSEL[3:0] ビットで、適切なtim_etr_inソース（内部または外部）を選択します。
2. この例ではフィルタは不要なので、TIMx_SMCRレジスタのETF[3:0] に0000を書き込みます。
3. TIMx_SMCRレジスタにETPS[1:0]=01を書き込むことによって、プリスケーラを設定します。
4. TIMx_SMCRレジスタにETP=0を書き込むことによって、tim_etr_inの立ち上がりエッジ検出を選択します。
5. TIMx_SMCRレジスタにECE=1を書き込むことによって、外部クロックモード2を有効にします。
6. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを有効にします。

カウンタは2回のtim_etr_in立上がりエッジごとに1回カウントします。

tim_etr_inの立上がりエッジから実際のカウンタクロックまでの間に、tim_etrp信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大でTIMxCLK周波数の1/4です。ETRP信号が高速の場合、ユーザは適切なETPSプリスケーラー設定によって外部信号の分周比を適用する必要があります。

図617. 外部クロックモード2の制御回路



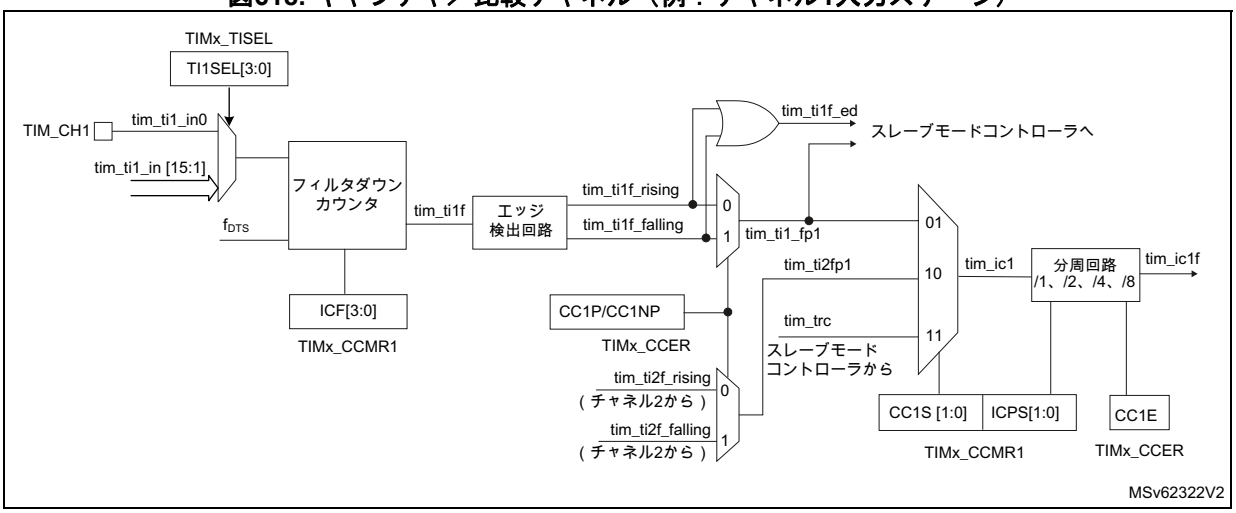
55.4.6 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

入力ステージは、対応するtim_tix入力をサンプリングして、フィルタリングを行った信号tim_tixfを生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図618. キャプチャ／比較チャネル（例：チャネル1入力ステージ）



出力ステージは、その後、基準として使用される中間波形tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図619. キャプチャ／比較チャネル1メイン回路

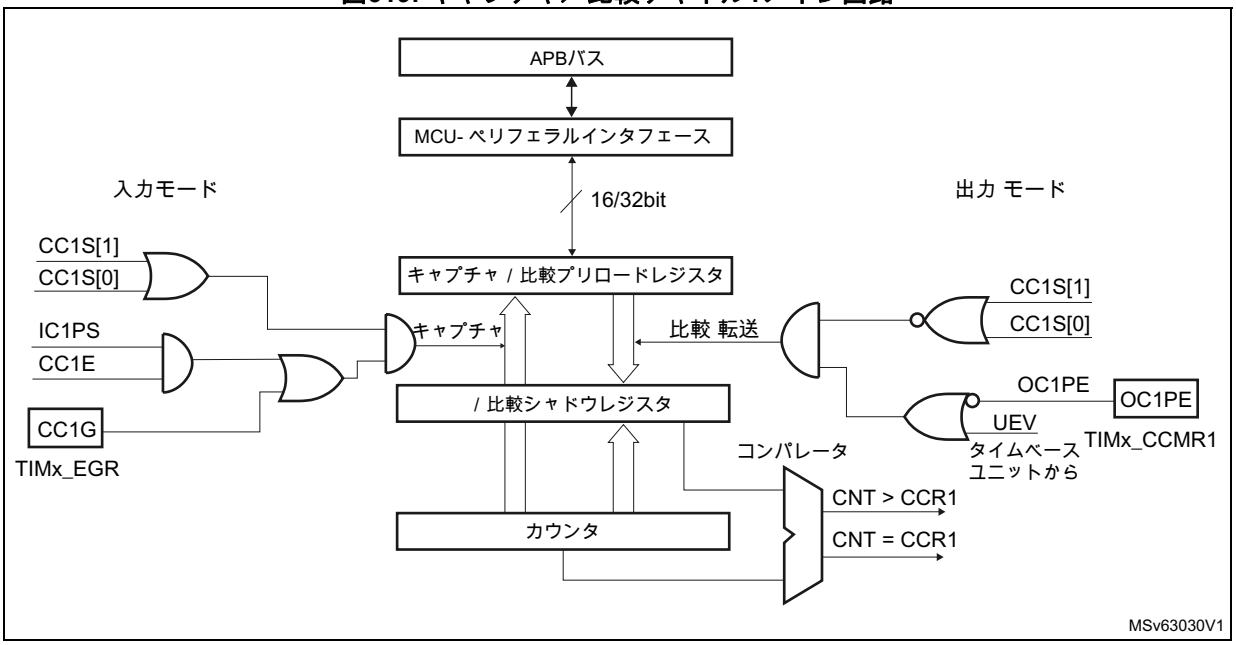
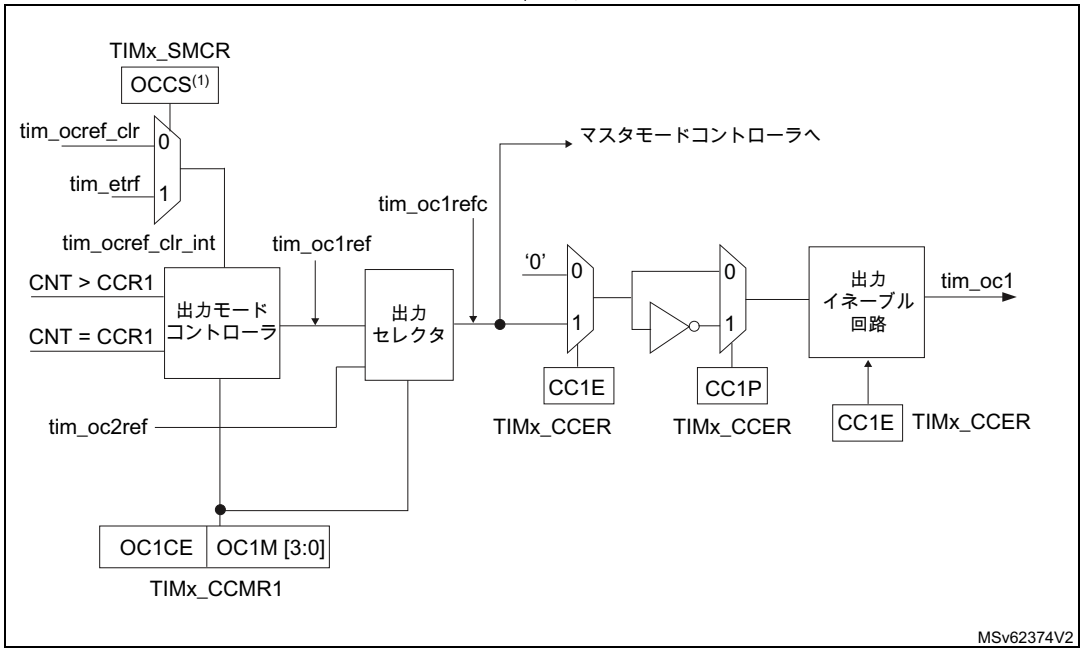


図620. キャプチャ／比較チャネル（チャネル1、同じくチャネル2、3、および4）の出力ステージ



1. いくつかのインスタンスのみで使用できます。使用できない場合は、tim_etrfは直接tim_ocref_clr_intに接続されます。

キャプチャ／比較ブロックは、1つのプリロードレジスタと1つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

55.4.7 入力キャプチャモード

入力キャプチャモードでは、対応するICx信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応するCCXIFフラグ（TIMx_SRレジスタ）がセットされ、割込みまたはDMAリクエストを送信できます（有効な場合）。CCxIFフラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグCCxOF（TIMx_SRレジスタ）がセットされます。CCxIFフラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRxレジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOFは、0を書き込むとクリアされます。

次の例は、tim_ti1入力が立ち上がったときに、カウンタの値をTIMx_CCR1にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISELレジスタのTI1SEL[3:0] ビットで、適切なtim_tix_in[15:0] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1はtim_ti1入力とリンクされていなければならず、このためにはTIMx_CCMR1レジスタのCC1Sビットに01を書き込みます。CC1Sの値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1レジスタは読出し専用になります。
3. タイマに接続される信号に関して、必要な入力フィルタ時間をプログラムします（入力がtim_tixの1つである場合、TIMx_CCMRxレジスタのICxFビットによって）。入力信号の反転時、最大で内部クロックの5サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を5クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した8個のサンプルが検出されたときに、tim_ti1の遷移を検証できます（周波数f_{DTS} でサンプリング）。この場合、TIMx_CCMR1レジスタのIC1Fビットに0011を書き込みます。
4. tim_ti1チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCERレジスタのCC1P、CC1NP、およびCC1NPビットに“000”を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1レジスタのIC1PSビットに00を書き込みます）。
6. TIMx_CCERレジスタのCC1Eビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIERレジスタのCC1IEビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIERレジスタのCC1DEビットをセットすることによって、DMAリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値がTIMx_CCR1レジスタに格納されます。
- CC1IFフラグがセットされます（割込みフラグ）。CC1OFビットは、少なくとも2回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IEビットに応じて、割込みが生成されます。
- CC1DEビットに応じて、DMAリクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これは、フラグ読出し後、データ読出し前に発生するかもしれないオーバーキャプチャの見落としを避けるためです。

注： IC割込みとDMAリクエストは、TIMx_EGRレジスタの対応するCCxGビットをセットすることによって、ソフトウェアによって生成することができます。

55.4.8 PWM入力モード

このモードは、単一のtim_tix入力に接続されたPWM信号の周期とデューティサイクルの両方を測定するために使用されます。

- TIMx_CCR1レジスタは周期値（2つの連続した立ち上がりエッジ間の間隔）を保持します。
- TIM_CCR2レジスタはパルス幅（2つの連続した立ち上がりと立ち下がりエッジ間の間隔）を保持します。

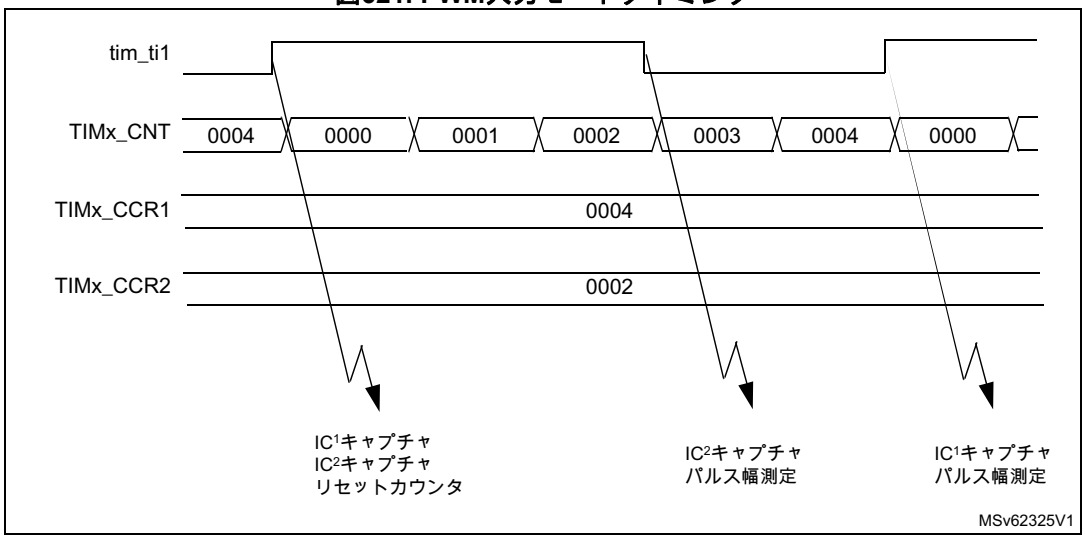
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2つのICx信号が同じtim_tix入力にマッピングされます。
- この2つのICx信号は、逆の極性のエッジでアクティブです。
- 2つのTIxFP信号の1つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

tim_ti1に適用されるPWM信号の周期とパルス幅は、以下の手順を使用して測定できます。

1. TIMx_TISELレジスタのTI1SEL[3:0] ビットで、適切なtim_tix_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに “01” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1PビットとCC1NPビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、tim_ti1fp1のアクティブ極性を選択します（TIMx_CCR1のキャプチャとカウンタクリアの両方に使用します）。
4. TIMx_CCMR1 レジスタの CC2S ビットに “10” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2Pビットに“1”を、CC2NPビットに“0”を書き込むことによって（立ち下がりエッジでアクティブ）、tim_ti1fp2のアクティブ極性を選択します（TIMx_CCR2のキャプチャに使用されます）。
6. TIMx_SMCRレジスタのTSビットに00101を書き込むことによって（tim_ti1fp1を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCRレジスタのSMSビットに100を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCERレジスタのCC1EとCC2Eビットに1を書き込むことによって、キャプチャを有効にします。

図621. PWM入力モードタイミング



1. tim_ti1fp1とtim_ti2fp2のみがスレーブモードコントローラに接続されているので、PWM入力モードはTIMx_CH1/TIMx_CH2信号でのみ使用できます。

55.4.9 強制出力モード

このモード（TIMx_CCMRxレジスタのCCxSビット = 00）では、各出力比較信号（tim_ocxref、そしてtim_ocx）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応するTIMx_CCMRxレジスタのOCxMビットに101を書き込む必要があります。これにより、tim_ocxrefは強制的にハイレベルになり（tim_ocxrefは常にアクティブハイ）、tim_ocxはCCxP極性ビットと逆の値になります。

例：CCxP=0（tim_ocxアクティブハイ）=> tim_ocxは強制的にハイレベルになります。

TIMx_CCMRxレジスタのOCxMビットに100を書き込むことによって、tim_ocxref信号を強制的にローにできます。

いずれにしても、TIMx_CCRxシャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みやDMAリクエストを送信できます。これについては、出力比較モードのセクションで説明します。

55.4.10 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCERレジスタのCCxPビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SRレジスタのCCxIFビット）。
- 対応する割込みマスク（TIMx_DIERレジスタのCCXIEビット）がセットされている場合は、割込みを生成します。

- 対応するイネーブルビット（TIMx_DIERレジスタのCCxDEビット）がセットされている場合は、DMAリクエストを送信します（DMAリクエスト選択には、TIMx_CR2レジスタのCCDSビットが使用されます）。

TIMx_CCRxレジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRxレジスタのOCxPEビットを使用してプログラミングできます。

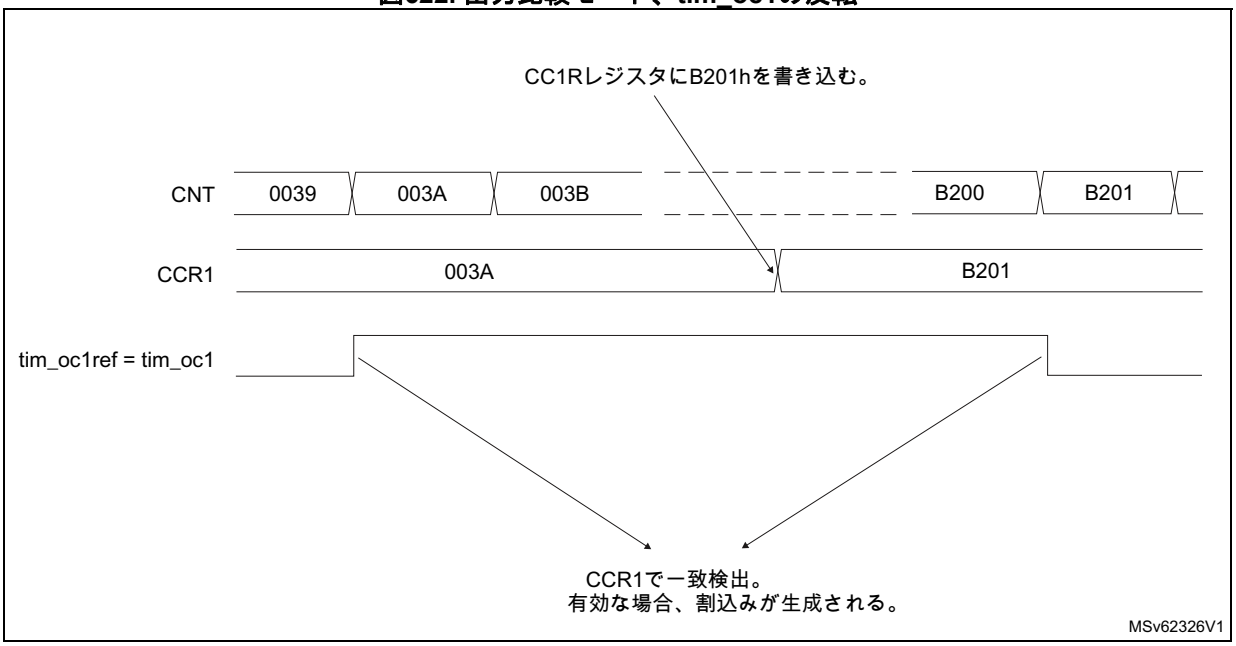
出力比較モードでは、更新イベントUEVはtim_ocxrefおよびtim_ocx出力には影響を与えません。タイミングの分解能はカウンタの1カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARRレジスタとTIMx_CCRxレジスタに目的のデータを書き込みます。
3. 割込みリクエスト／DMAリクエストを生成する場合は、CCxIEビット／CCxDEビットをセットします。
4. 出力モードを選択します。例：
   a) CNTとCCRxが一致したときにtim_ocx出力ピンを反転するには、OCxMビットに0011を書き込みます。
   b) プリロードレジスタを無効にするには、OCxPEビットに0を書き込みます。
   c) アクティブハイ極性を選択するには、CCxPビットに0を書き込みます。
   d) 出力を有効にするには、CCxEビットに1を書き込みます。
5. TIMx_CR1レジスタのCENビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアでTIMx_CCRxレジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRxシャドウレジスタは、次の更新イベントUEVでのみ更新されます。例を 図622に示します。

図622. 出力比較モード、tim_oc1の反転



55.4.11 PWMモード

パルス幅変調（PWM）モードは、TIMx_ARRレジスタの値によって決められた周波数とTIMx_CCRxレジスタの値によって決められたデューティサイクルの信号を生成するために使用されます。

PWMモードは、個々のチャネル（tim_ocx出力ごとにPWM 1波形）で、TIMx_CCMRxレジスタのOCxMビットに“110”（PWMモード1）や“111”（PWMモード2）を書き込むことで、独自に選択できます。TIMx_CCMRxレジスタのOCxPEビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1レジスタのARPEビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGRレジスタのUGビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

tim_ocx極性は、TIMx_CCERレジスタのOCxPビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx出力は、TIMx_CCERレジスタのCCxEビットによって有効になります。詳細については、TIMx_CCERxレジスタの説明を参照してください。

PWMモード（1または2）では、TIMx_CNTとTIMx_CCRxが常に比較されて、TIMx_CCRx ≤ TIMx_CNTまたはTIMx_CNT ≤ TIMx_CCRxかどうかが判断されます（カウントの方向によります）。tim_ocref_clrは、tim_etr_inまたはtim_oceref_clr信号を介した外部イベントによってクリアできます。この場合、tim_ocref_clr信号は、以下の場合にのみ、アサートされます。

- 比較一致イベントの後。
- 出力比較モード（TIMx_CCMRxレジスタのOCxMビット）が停止構成（比較なし、OCxM=000）からPWMモードの1つ（OCxM=110または111）へ切り替えられたとき。タイマの動作中は、ソフトウェアで強制的にPWMになります。

タイマは、TIMx_CR1レジスタのCMSビットに応じて、エッジアラインモードまたはセンターアラインモードでPWMを生成できます。

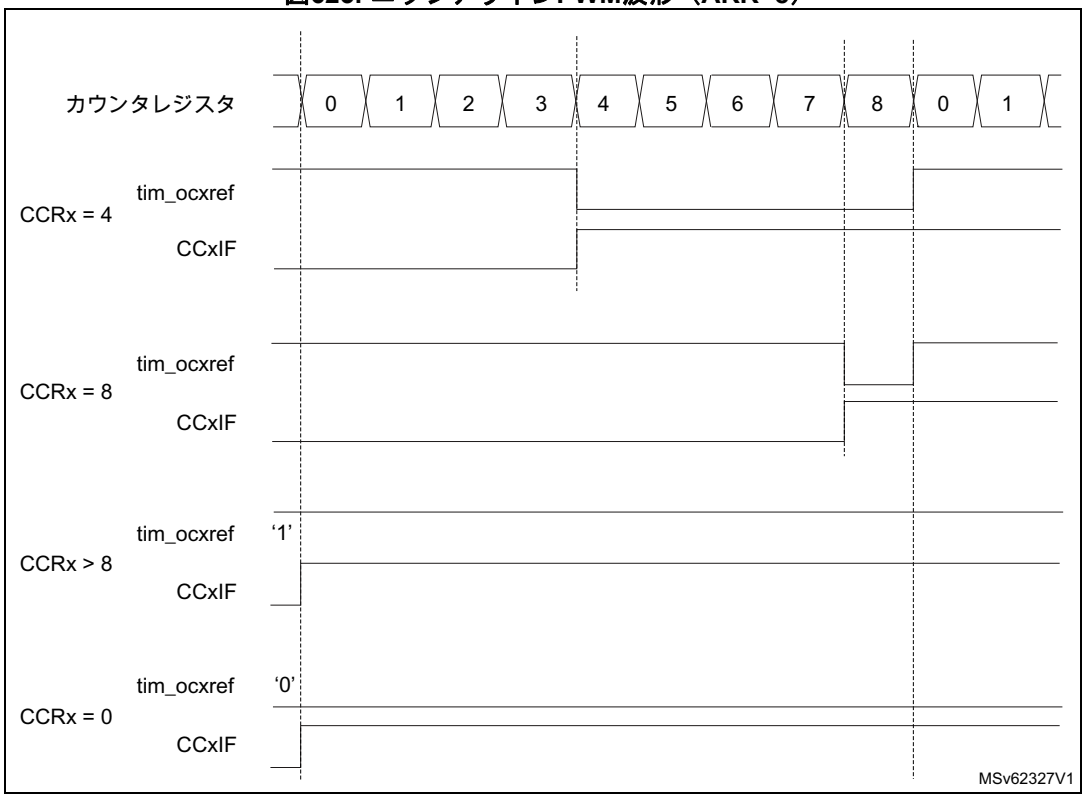
PWMエッジアラインモード

アップカウント構成

TIMx_CR1レジスタのDIRビットがローのときには、アップカウントがアクティブです。2225ページのアップカウントモードを参照してください。

次の例では、PWMモード1を使用しています。PWM基準信号tim_ocxrefは、TIMx_CNT < TIMx_CCRxの間はハイに、そうでない場合はローになります。TIMx_CCRxの比較値がTIMx_ARRの自動再ロード値より大きい場合、tim_ocxrefは“1”に保持されます。比較値が0の場合、tim_ocxrefは“0”に保持されます。図623にTIMx_ARR=8のときのエッジアラインPWM波形の例を示します。

図623. エッジアラインPWM波形（ARR=8）



ダウンカウント構成

TIMx_CR1レジスタのDIRビットがハイのときには、ダウンカウントがアクティブです。2228ページのダウンカウントモードを参照してください。

PWMモード1では、基準信号tim_ocxrefは、TIMx_CNT > TIMx_CCRxの間はローであり、そうでない場合はハイになります。TIMx_CCRxの比較値がTIMx_ARRの自動再ロード値より大きい場合、tim_ocxrefは100% に保持されます。このモードでは、PWM信号を生成することはできません。

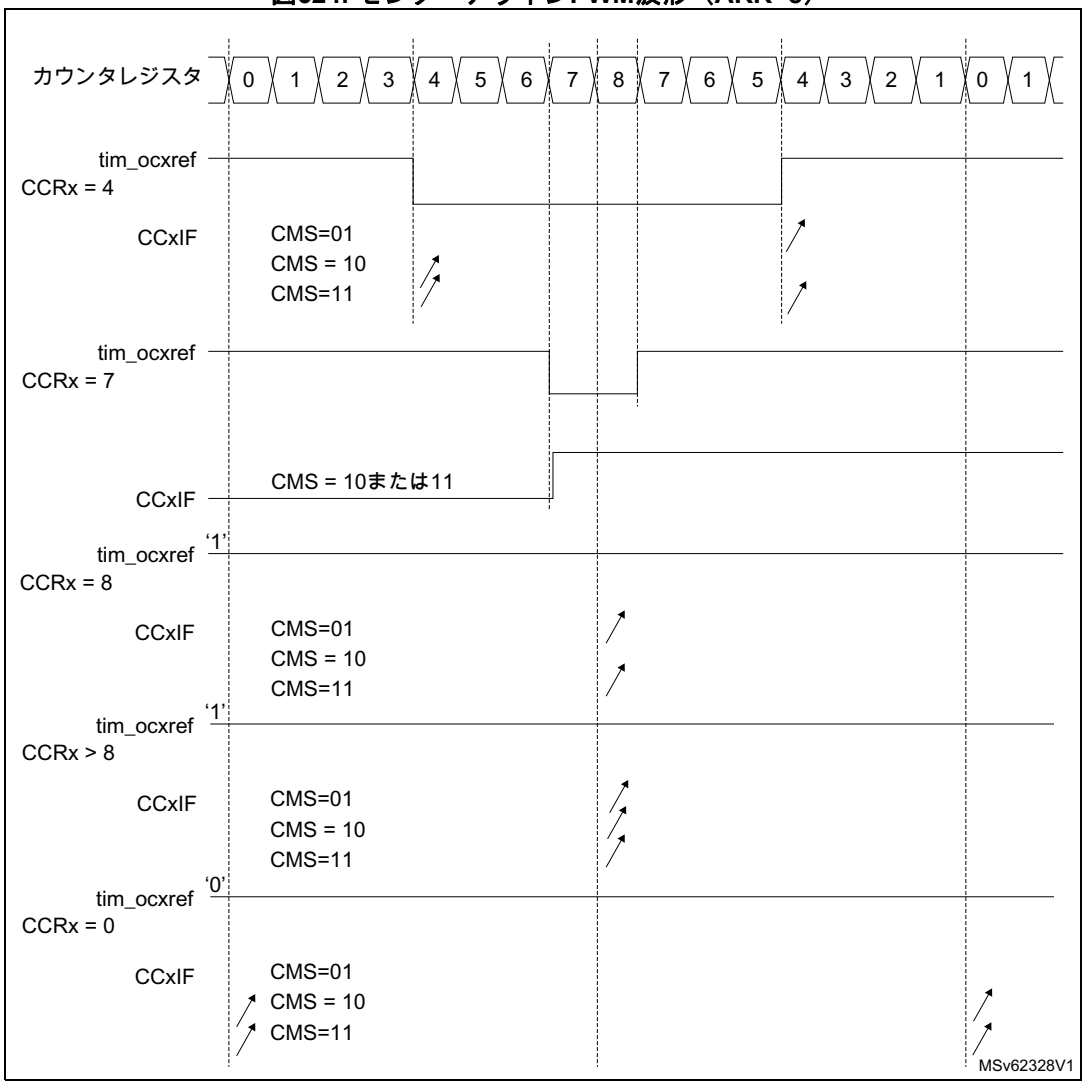
PWMセンターアラインモード

センターアラインモードは、TIMx_CR1レジスタのCMSビットが“00”でないときにアクティブです（その他すべての設定は、tim_ocxref/tim_ocx信号に対して同じ効果を持ちます）。比較フラグは、CMSビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。2231ページのセンターアラインモード（アップ／ダウンカウント）を参照してください。

図624に、次の条件でのセンターアラインPWM波形の例を示します。

- TIMx_ARR=8
- PWMモードはPWMモード1
- フラグは、TIMx_CR1レジスタのCMS=01で選択されたセンターアラインモード1に対応して、カウンタがカウントダウンするときにセットされます。

図624. センターアラインPWM波形（ARR=8）



センターアラインモードの使用に関するヒント：

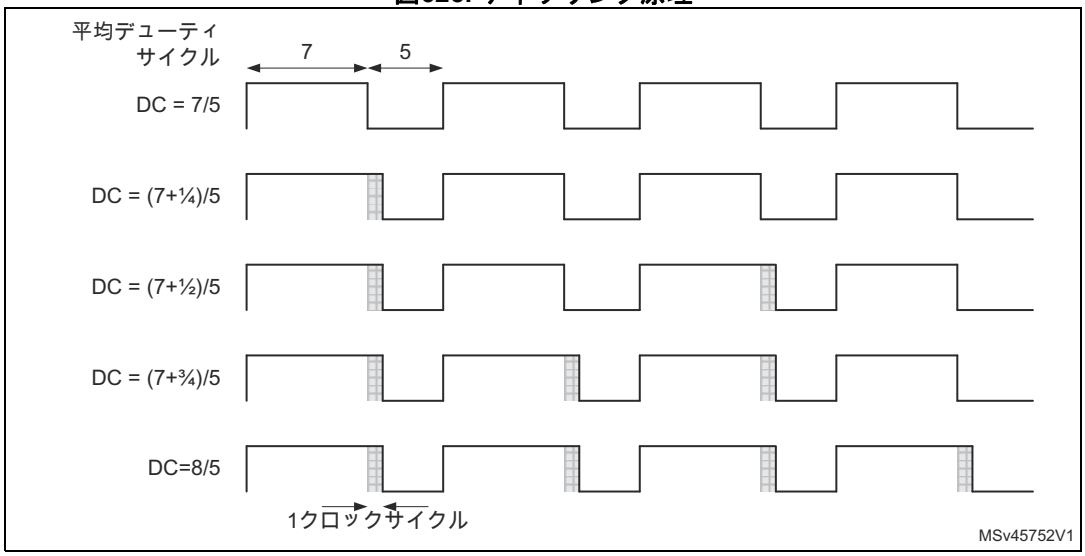
- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1レジスタのDIRビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIRビットとCMSビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに0またはTIMx_ARR値が書き込まれた場合、方向は更新されますが、更新イベントUEVは生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGRレジスタのUGビットをセットする）、動作中はカウンタへの書込みを行わないことです。

ディザリングモード

PWMモードの有効分解能は、TIMx_CR1レジスタのDITHENビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）とARR（PWM周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16個の連続したPWM周期にわたって実際のCCR（またはARR）値をわずかに変更する（1タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたはPWM周期を考えると、16倍の分解能の向上が可能になります。下記の図625は、4つの連続するPWMサイクルに適用されたディザリング原理を示します。

図625. ディザリング原理



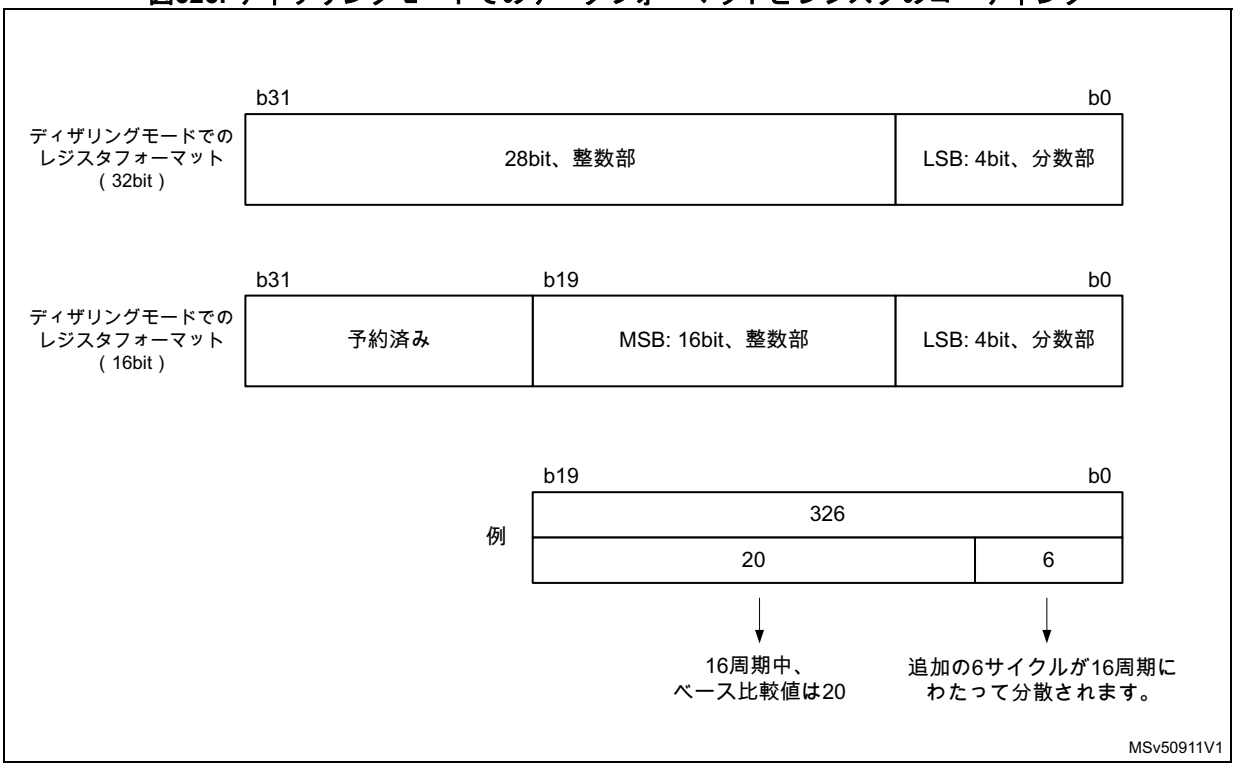
ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図626を参照）。

- 下位4bitは、拡張分解能部分（分数部分）用です。
- 上位ビットは、4桁だけ左にシフトされ、ベース値用のコーディングになります。16bitモードでは、16bitのフォーマットが維持されます。

注： **DITHENビットをリセットするときは、次の手順に従う必要があります。**

1. CENおよびARPEビットをリセットする必要があります。
2. DITHENビットをリセットする必要があります。
3. CCIFフラグをクリアする必要があります。
4. CENビットをセットすることができます（最終的にARPE = 1とともに）。

図626. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

$$\text{ディザリングモードが無効の場合: } F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモード（16bitタイマ）: } F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

$$\text{ディザリングモード（32bitタイマ）: } F_{pwmMin} = \frac{F_{Tim}}{268435454 + \frac{15}{16}}$$

注： **16bitタイマの場合、TIMx_ARRおよびTIMxCCRyの最大値は、ディザリングモードでは0xFFFEFに制限されています（整数部では65534、ディザリング部分では15に対応します）。**
32bitタイマの場合、TIMx_ARRおよびTIMxCCRyの最大値は、ディザリングモードでは0xFFFFFFEFに制限されています（整数部では264435454、ディザリング部分では15に対応します）。

下記の図627と図628に示すように、ディザリングモードはPWM分解能を増加するために使用されます。

図627. PWM分解能 対 周波数（16bitモード）

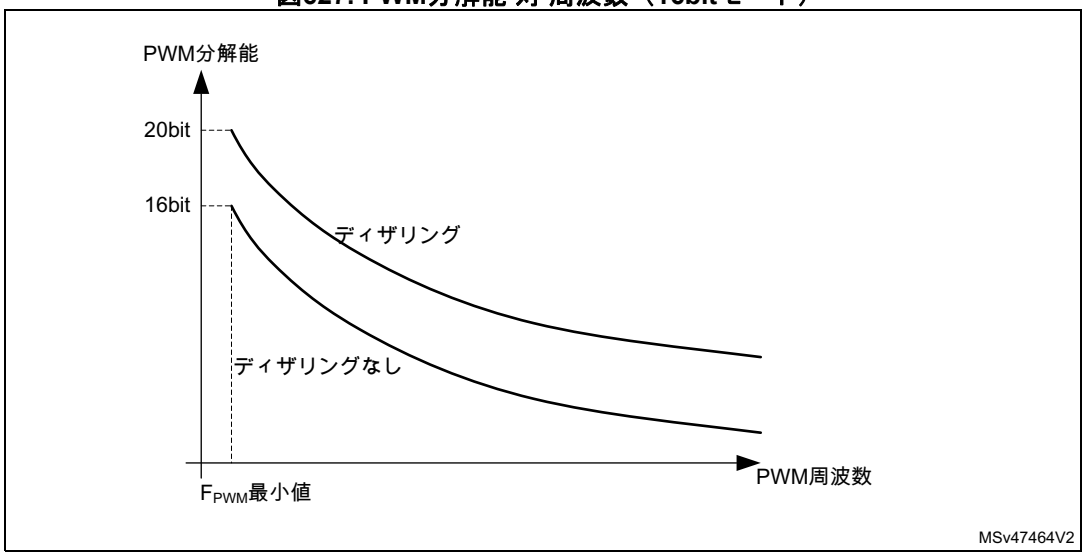
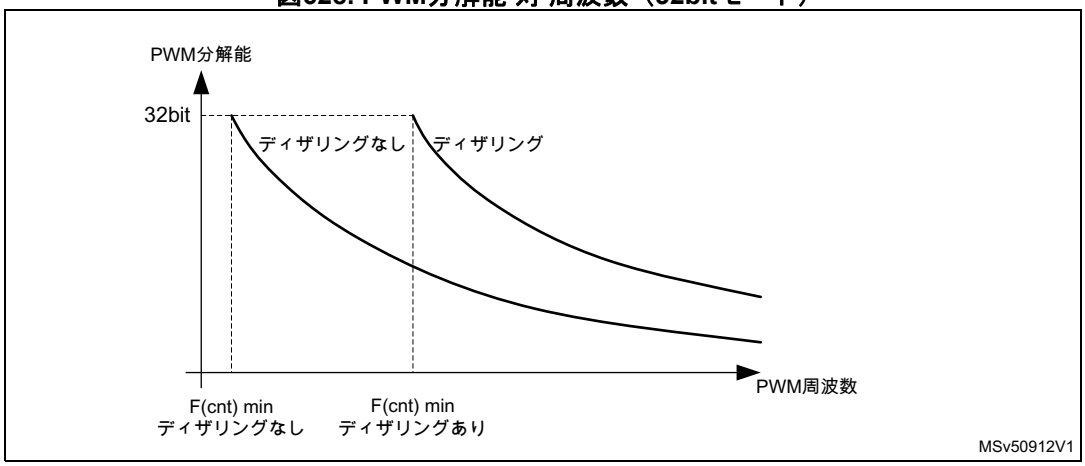
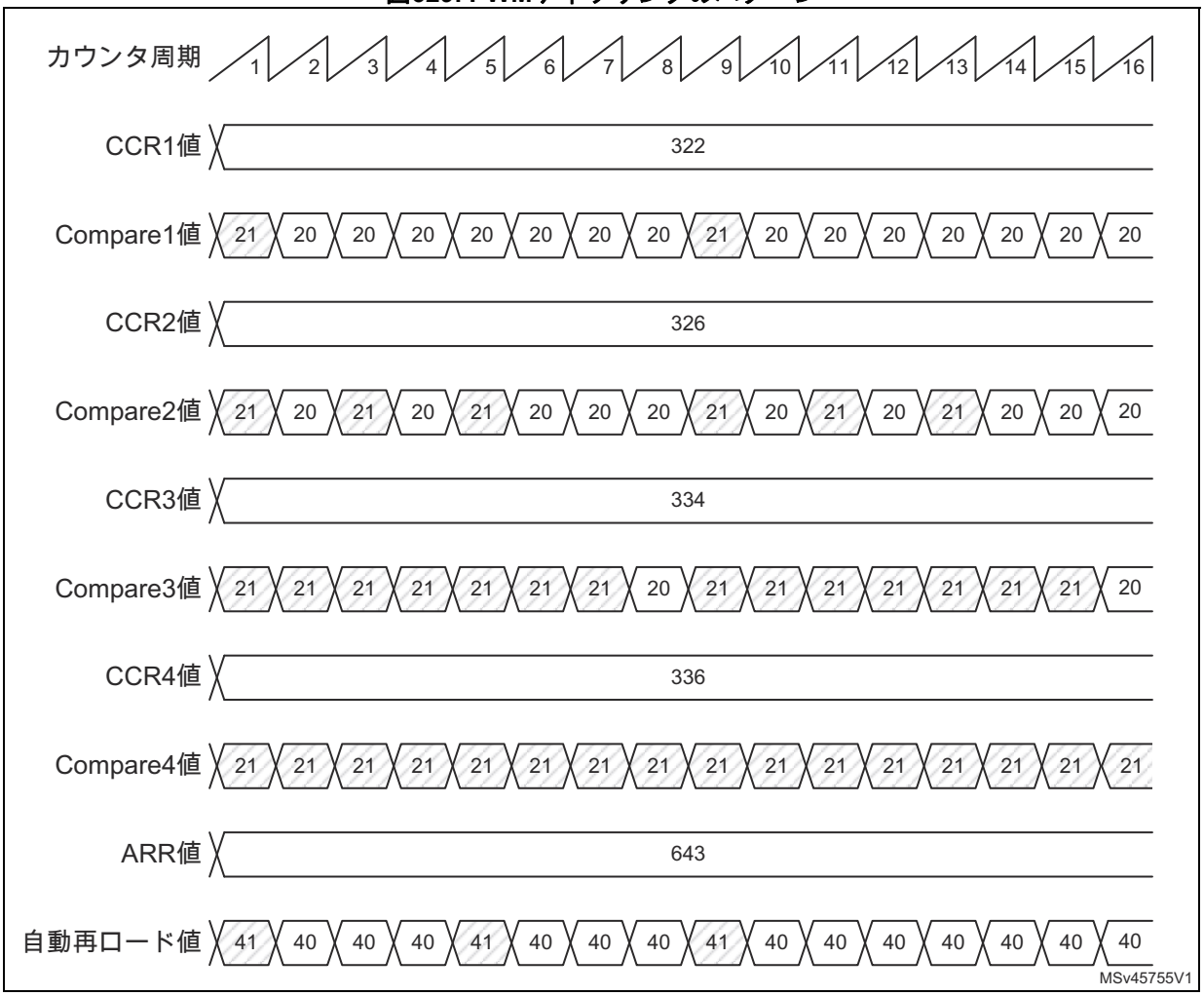


図628. PWM分解能 対 周波数（32bitモード）



下記の図629で説明するように、デューティサイクルや周期の変更は、連続した16周期にわたって広がります。

図629. PWMディザリングのパターン



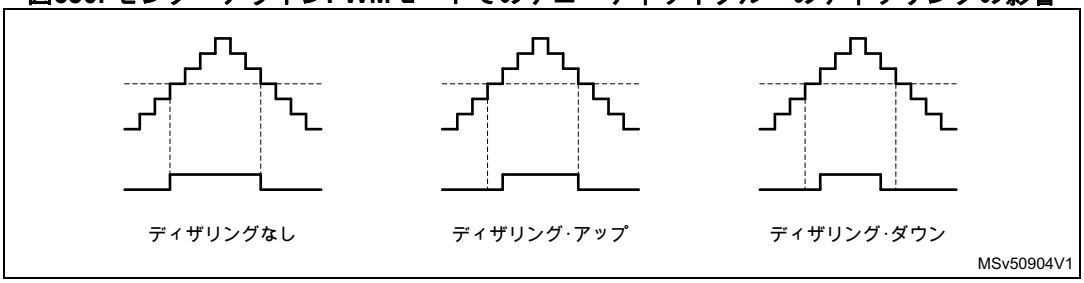
自動再ロード値と比較値の増分は、下記の表561に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表561.CCRおよびARRレジスタの変更のディザリングパターン

LSB値	PWM周期															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

ディザリングモードはセンターアラインPWMモード（TIMx_CR1レジスタのCMSビットが“00”に等しくないとき）でも使用できます。この場合、ディザリングパターンは、以下の図630に示すように、アップおよびダウンカウントフェーズを考慮して、8つの連続したPWM周期にわたって適用されます。

図630. センターアラインPWMモードでのデューティサイクルへのディザリングの影響



下記の表562は、センターアラインPWMモードでどのようにディザリングパターンが追加されるかを示しています。

表562.センターアラインPWMモードでのCCRレジスタ変更のディザリングパターン

LSB値	PWM周期															
	1		2		3		4		5		6		7		8	
	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

55.4.12 非対称PWMモード

非対称モードでは、プログラム可能な位相シフトによって2つのセンターアラインPWM信号の生成を可能にします。周波数がTIMx_ARRレジスタの値で決定されるのに対し、デューティサイクルや位相シフトはTIMx_CCRxレジスタペアで決定されます。1つ目のレジスタがアップカウント時のPWMを制御し、2つ目のレジスタがダウンカウント時のPWMを制御することで、PWMはPWMハーフサイクルごとに調整されます。

- tim_oc1refc（またはtim_oc2refc）は、TIMx_CCR1およびTIMx_CCR2によって制御されます。
- tim_oc3refc（またはtim_oc4refc）は、TIMx_CCR3およびTIMx_CCR4によって制御されます。

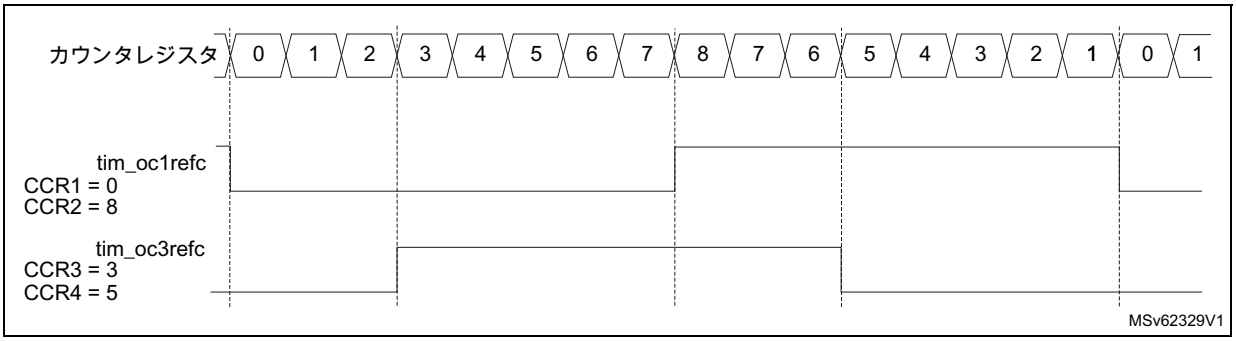
非対称PWMモードは、TIMx_CCMRxレジスタのOCxMビットに“1110”（非対称PWMモード1）または“1111”（非対称PWMモード2）を書き込むことによって、2チャネルごとに選択できます（CCRレジスタペアごとに1つのtim_ocx出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

特定のチャネルが非対称のPWMチャネルとして使用されると、その2次チャネルも使用できます。たとえば、tim_oc1refc信号がチャネル1（非対称PWMモード1）に生成されると、チャネル2のtim_oc2ref信号、または非対称PWMモード2の結果として得られるtim_oc2refc信号を出力できます。

図631は、非対称PWMモードを使用して生成される信号の例を表します（チャネル1から4は非対称PWMモード2として設定されます）。

図631. 50% デューティサイクルの2位相シフトされたPWM信号の生成



55.4.13 組み合わせPWMモード

組み合わせPWMモードでは、2つのエッジアラインまたはセンターアラインPWM信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数がTIMx_ARRレジスタの値で決定されるのに対し、デューティサイクルや遅延は2つのTIMx_CCRxレジスタで決定されます。結果として得られる信号tim_ocxrefcは、2つのPWM基準信号のORまたはANDによる論理結合から成ります。

– tim_oc1refc（またはtim_oc2refc）は、TIMx_CCR1およびTIMx_CCR2によって制御されます。

– tim_oc3refc（またはtim_oc4refc）は、TIMx_CCR3およびTIMx_CCR4によって制御されます。

組み合わせPWMモードは、TIMx_CCMRxレジスタのOCxMビットに“1100”（組み合わせPWMモード1）または“1101”（組み合わせPWMモード2）を書き込むことによって、2チャネルごとに選択できます（CCRレジスタペアごとに1つのtim_ocx出力）。

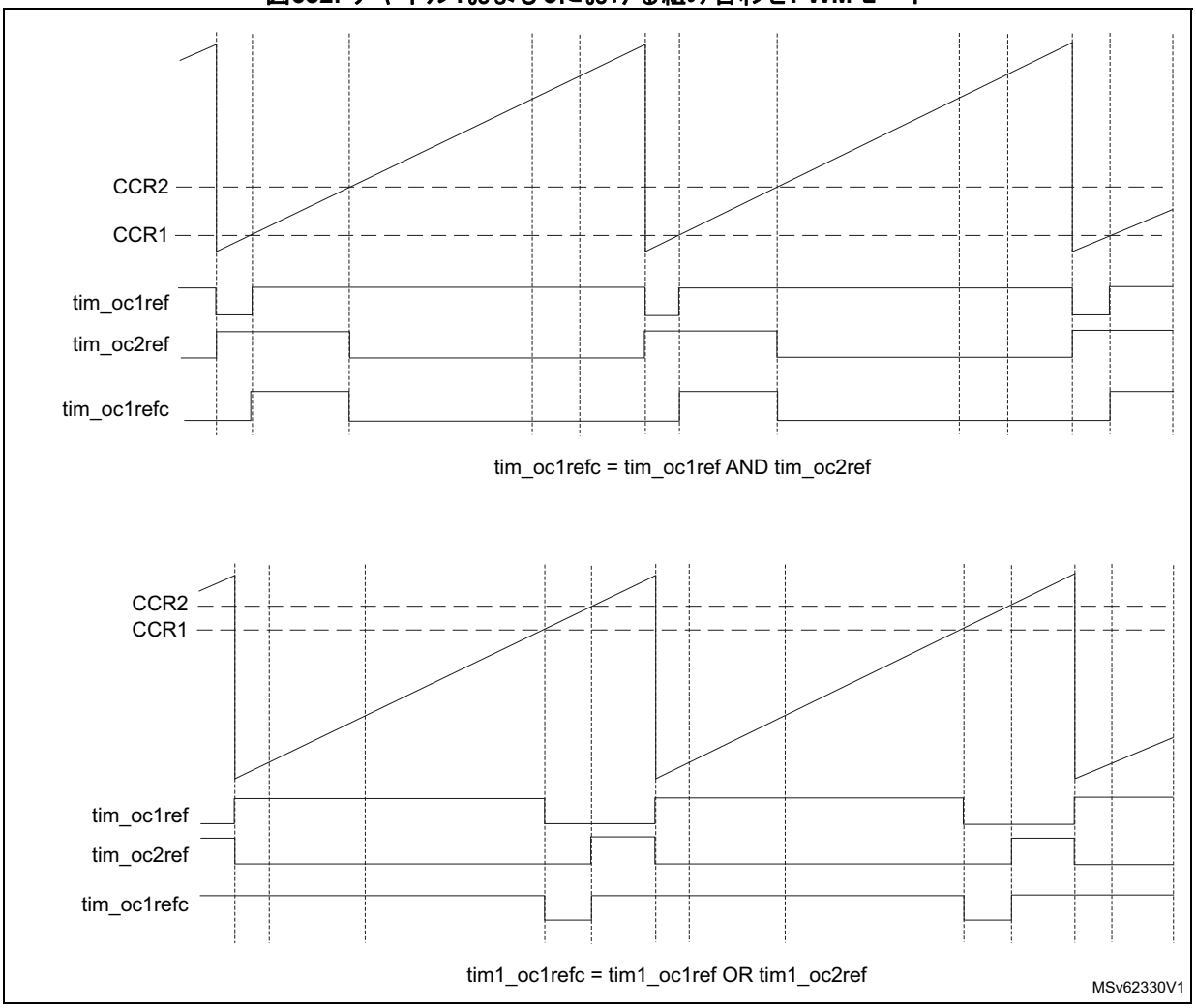
特定のチャネルが組み合わせPWMチャネルとして使用されている場合、2次チャネルを反対のPWMモードに設定する必要があります（たとえば、1つを組み合わせPWMモード1、もう1つを組み合わせPWMモード2にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

図632は、次の設定で取得可能な組み合わせPWMモードを使用して生成される信号の例を表します。

- チャネル1が組み合わせPWMモード2で設定されている場合
- チャネル2がPWMモード1で設定されている場合
- チャネル3が組み合わせPWMモード2で設定されている場合
- チャネル4がPWMモード1で設定されている場合

図632. チャネル1および3における組み合わせPWMモード



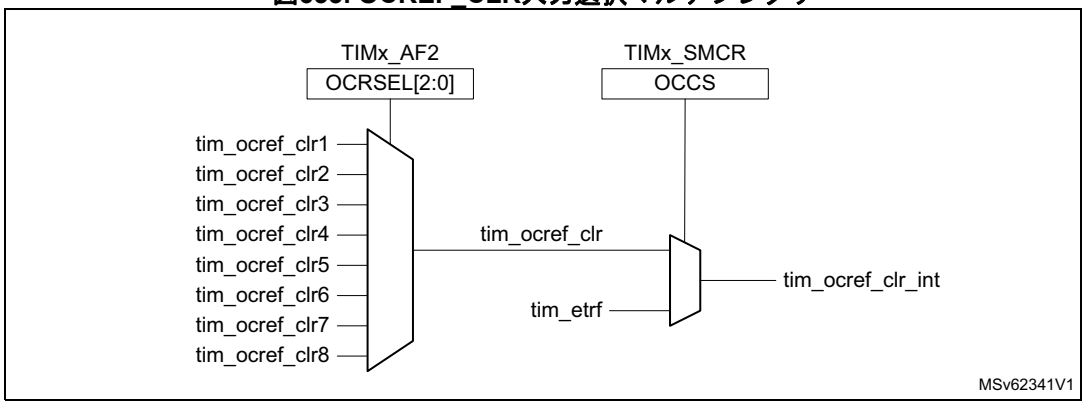
55.4.14 外部イベントによるtim_ocxref信号のクリア

特定のチャネルのtim_ocxref信号はtim_ocref_clr_int入力にハイレベルを適用するとクリアされます（対応するTIMx_CCMRxレジスタのOCxCEイネーブルビットを1にセットする）。tim_ocxrefは、次のPWMサイクルでの次のアクティブステートへの遷移まで、ローのまま残ります。この機能は、出力比較モードとPWMモードでのみ使用可能です。強制モードでは動作しません。

tim_ocref_clr_int のソースは、OCREF クリア選択機能の実装に依存します。セクション 55.3 : TIM2/TIM3/TIM4/TIM5 の実装を参照してください。

OCREFクリア選択機能が実装されている場合は、tim_ocref_clr_intは、TIMx_SMCRレジスタでOCCSビットを設定することで、tim_ocref_clr入力とtim_etrf入力（フィルタ後のtim_etr_in）の間で選択できます。tim_ocref_clr入力は、以下の図633に示すように、TIMx_AF2レジスタのOCRSEL[2:0] ビットフィールドを使用して複数のtim_ocref_clr[7:0] 入力から選択できます。

図633. OCREF_CLR入力選択マルチプレクサ



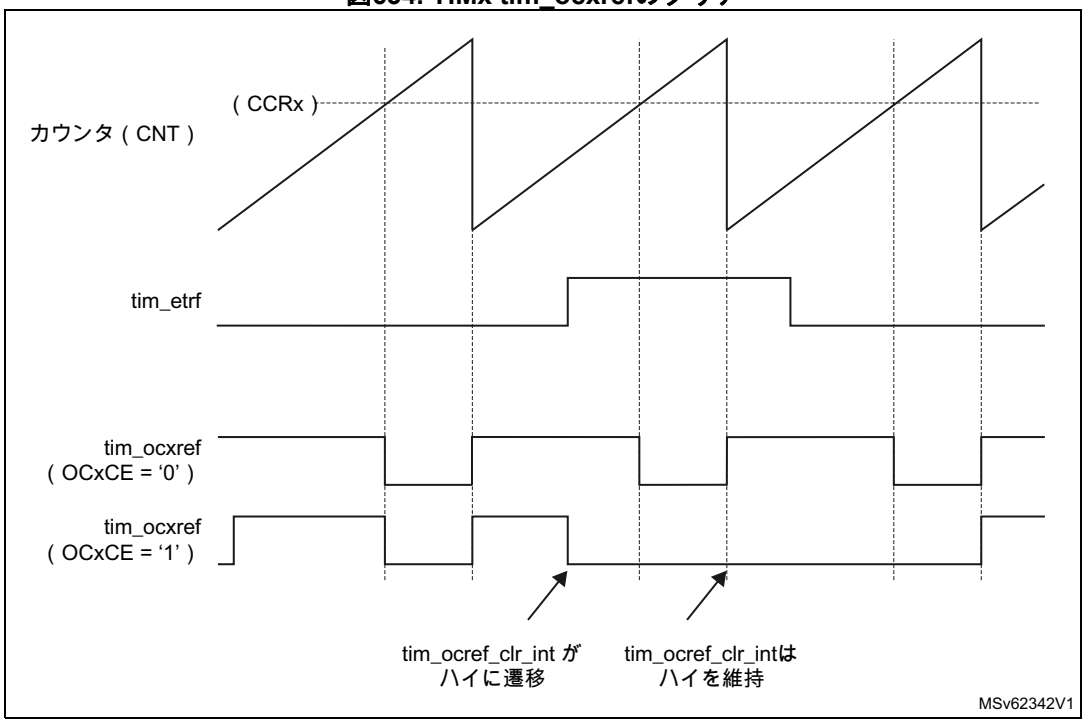
OCREFクリア選択機能が実装されていない場合は、tim_ocref_clr_int入力は直接tim_etrf入力に接続されます。

たとえば、tim_ocref_clr_int信号をコンパレータの出力に接続して、電流処理に使用することができます。この場合、tim_etr_inは次のように設定する必要があります。

1. 外部トリガプリスケーラをオフ状態に保持する必要があります。すなわち、TIMx_SMCRレジスタのビットETPS[1:0] が00にクリアされます。
2. 外部クロックモード2を無効にする必要があります。すなわち、TIM1_SMCRレジスタのビットECEが0にクリアされます。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて設定できます。

図634 に、OCxCE イネーブルビットの両方の値について、tim_etrf 入力がハイになったときのtim_ocxref信号の動作を示します。この例では、TIMxタイマはPWMモードにプログラミングされています。

図634. TIMx tim_ocxrefのクリア



注： 100% デューティサイクルのPWMの場合 （CCRx > ARRの場合）、次のカウンタオーバーフローでtim_ocxrefが再度有効になります。

55.4.15 ワンパルスモード

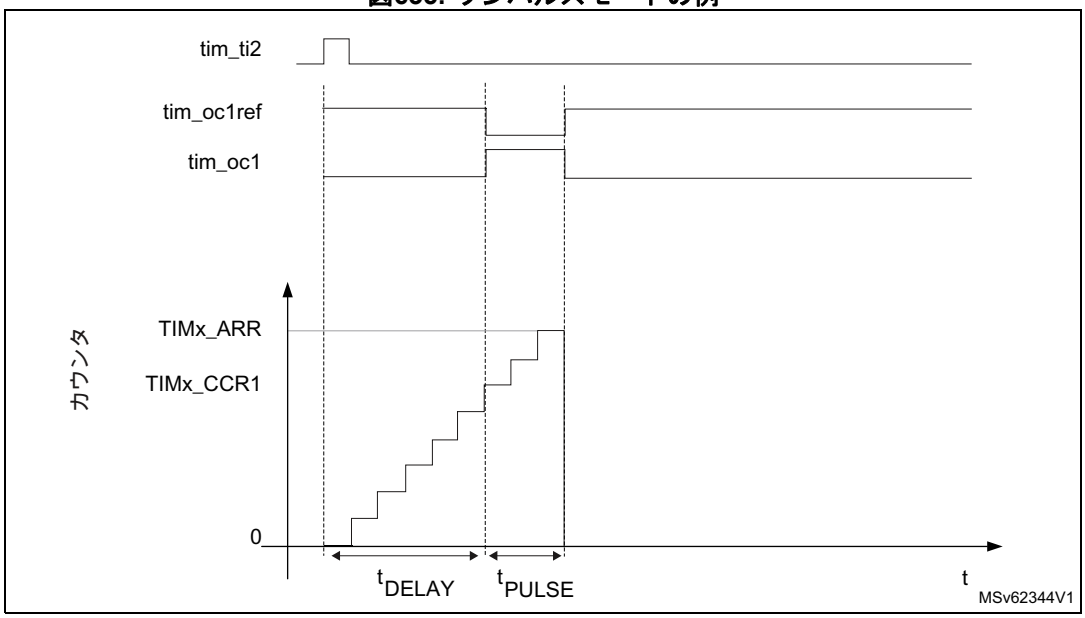
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたはPWMモードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1レジスタのOPMビットをセットします。これによって、カウンタは、次の更新イベントUEVで自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図635. ワンパルスモードの例



たとえば、tim_ti2入力ピンで立ち上がりエッジが検出されたときに、tim_oc1にパルス幅がt_{PULSE} の正のパルスを遅延時間t_{DELAY} 後に生成することもできます。

tim_ti2fp2をトリガ1として使用します。

1. TIMx_TISELレジスタのTI2SEL[3:0] ビットで、適切なtim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1レジスタにCC2S=01を書き込むことによって、tim_ti2fp2をtim_ti2に配置します。
3. tim_ti2fp2は、立ち上がりエッジを検出して、TIMx_CCERレジスタのCC2PビットとCC2NPビットに“0” を書き込みます。
4. tim_ti2fp2をスレーブモードコントローラのトリガ（tim_trgi）として構成するために、TIMx_SMCRレジスタのTS=“00110”を書き込みます。
5. tim_ti2fp2を使用してカウンタを開始するために、TIMx_SMCRレジスタのSMSビットに“110”（トリガモード）を書き込みます。

OPM波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに0から1へ遷移し、カウンタが自動再ロード値に達したときに1から0へ遷移する波形を生成するとします。このためには、TIMx_CCMR1レジスタにOC1M=111を書き込むことによって、PWMモード2を有効にする必要があります。オプションで、TIMx_CCMR1レジスタにOC1PE=1を書き込み、TIMx_CR1レジスタのARPEビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1レジスタに比較値を書き込み、TIMx_ARRレジスタに自動再ロード値を書き込む必要があります。次に、UGビットをセットすることによって更新を生成し、tim_ti2で外部トリガイベントを待つ必要があります。この例では、CC1Pに“0”を書き込みます。

上の例では、TIMx_CR1レジスタのDIRおよびCMSビットはローにする必要があります。

必要なパルスは1つだけなので（シングルモード）、TIMx_CR1レジスタのOPMビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1レジスタのOPMビットが“0”にセットされると、繰り返しモードが選択されます。

特殊ケース：tim_ocx高速イネーブル：

ワンパルスモードでは、tim_tix入力のエッジ検出によって、カウンタを有効にするCENビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRxレジスタのOCxFEビットをセットします。こうすると、tim_ocxref（およびtim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFEは、チャネルがPWM1またはPWM2モードに設定された場合のみ機能します。

55.4.16 再トリガ可能なワンパルスモード

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション55.4.15で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

タイマはスレーブモードである必要があり、このときビットはTIMx_SMCRレジスタでSMS[3:0] =「1000」（リセットモードとトリガモードの組み合わせ）、および再トリガ可能なOPMモード1または2でOCxM[3:0] が「1000」または「1001」にセットされています。

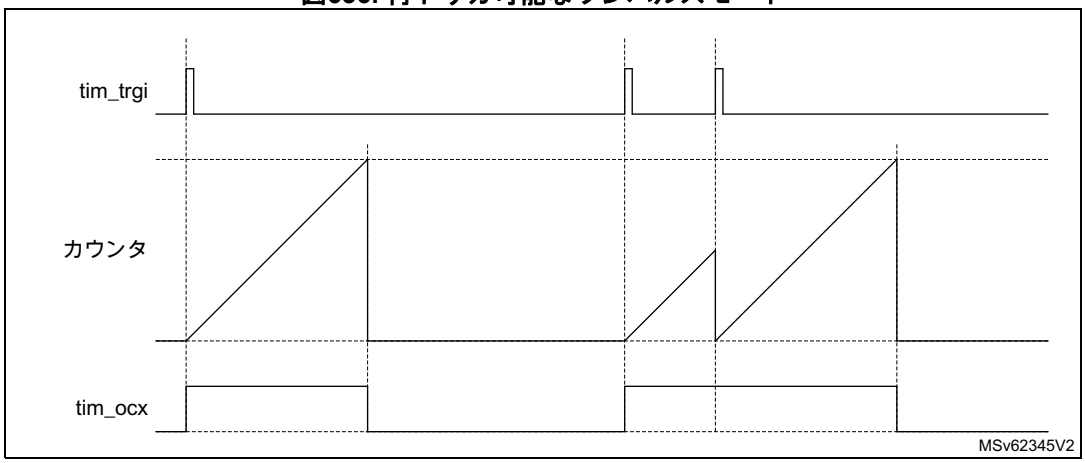
タイマをアップカウントモードで設定した場合、対応するCCRxを0にセットする必要があります（ARRレジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRxはARR以上である必要があります。

注： **再トリガ可能なワンパルスモードでは、CCxIFフラグは意味を持ちません。**

OCxM[3:0] およびSMS[3:0]ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。

このモードをセンターアラインPWMモードと組み合わせて使用することはできません。TIMx_CR1では、CMS[1:0] = 00にする必要があります。

図636. 再トリガ可能なワンパルスモード

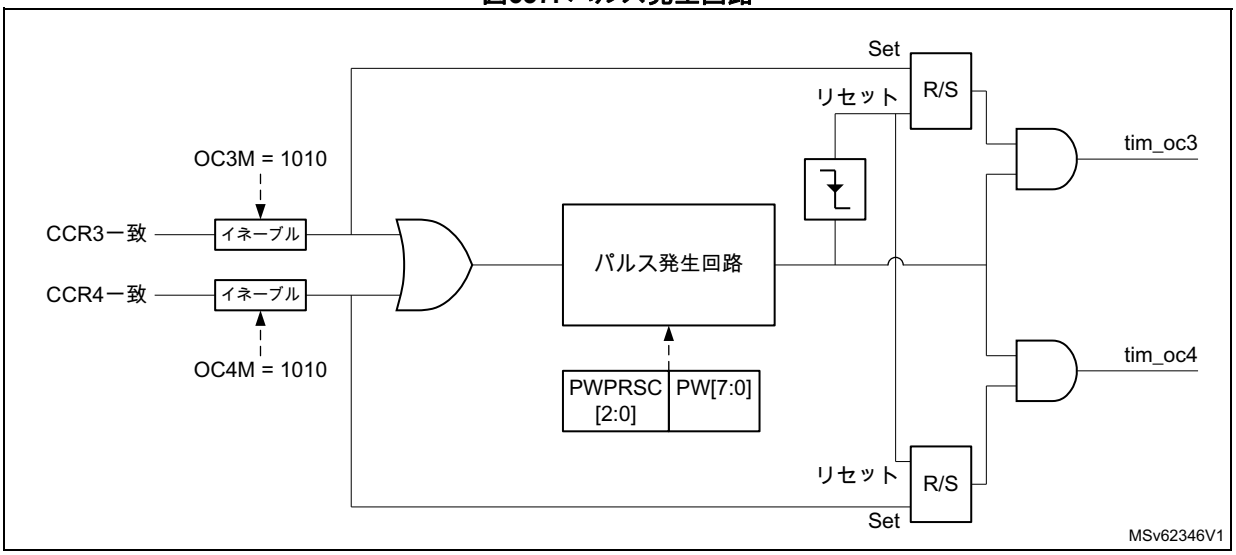


55.4.17 比較モードでのパルス

比較一致イベント時にパルスを生成できます。デバッグまたは同期のために、カウンタ値が特定の比較値と等しいときに、プログラム可能なパルス幅を持つ信号が生成されます。

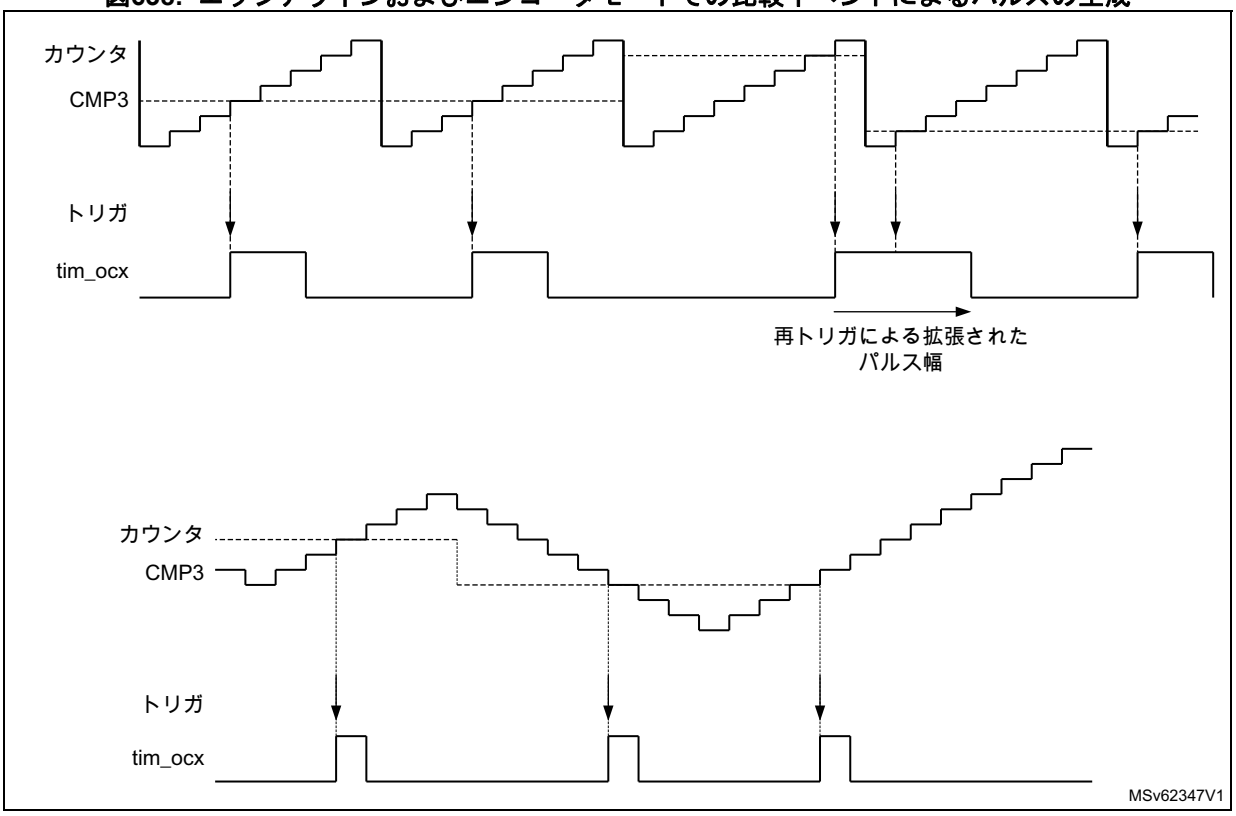
このモードは、エンコーダモードを含むすべてのスレーブモードの選択で、エッジアラインおよびセンタアラインのカウントモードで使用できます。これは、チャネル3とチャネル4のみで使用できます。以下の図637に示すように、パルス発生回路はただ1つであり、2つのチャネルで共有されます。

図637. パルス発生回路



下記の図638 は、エッジアラインおよびエンコーダ動作モードでパルスがどのように生成されるかを示しています。

図638. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成



この出力比較モードは、TIMx_CCMR2 レジスタのOC3M[3:0] およびOC4M[3:0] ビットフィールドを使用して選択されます。

パルス幅は、次のように、PWPRSC[2:0] ビットに従ってプリスケールされた特定のクロックを使用し、レジスタのPW[7:0] ビットフィールドを使用してプログラムします。

$t_{PW} = PW[7:0] \times t_{PWG}$

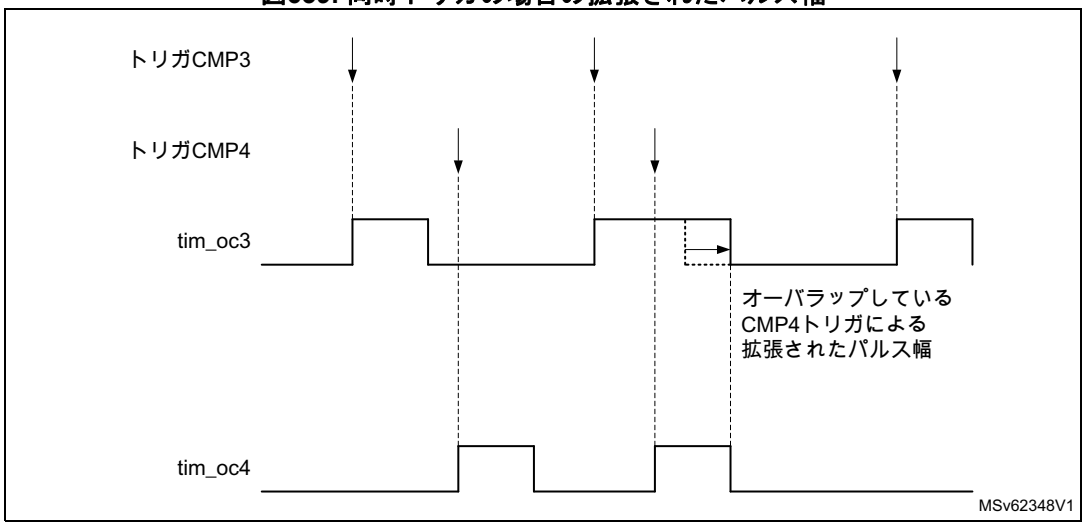
ここで、$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$.

上記は、プリスケーラの値に応じた分解能と最大値を示します。

パルスは再トリガ可能です。パルスが進行中に新しいトリガが入ると、パルスは拡張されます。

注： **2つのチャネルが同時に有効になっている場合、1つのチャネルのトリガが並行出力上に生成されたパルスとオーバーラップしない限り、パルスは独立して発行されます。反対に、2つのトリガがオーバーラップしている場合、最初に到着したトリガに関連するパルス幅は（再トリガのために）拡張されますが、後に到着したトリガのパルス幅は正しくなります（以下の図639に示すとおり）。**

図639. 同時トリガの場合の拡張されたパルス幅



55.4.18 エンコーダインタフェースモード

直交エンコーダ

エンコーダインタフェースモードを選択するには、TIMx_SMCRレジスタで、カウンタがtim_ti1エッジのみをカウントしている場合はSMS=0001を、tim_ti2エッジのみをカウントしている場合はSMS=0010を、tim_ti1とtim_ti2の両方のエッジをカウントしている場合はSMS=0011を書き込みます。

tim_ti1とtim_ti2の極性を選択するには、TIMx_CCERレジスタのCC1PビットとCC2Pビットをプログラミングします。CC1NPとCC2NPはクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。

2つの入力tim_ti1とtim_ti2は、インクリメンタルエンコーダとのインタフェースに使用されます。表563を参照してください。カウンタのクロックは、tim_ti1fp1またはtim_ti2fp2（入力フィルタおよび極性選択後はtim_ti1およびtim_ti2。フィルタされず、反転されない場合はtim_ti1fp1=tim_ti1、フィルタされず、反転されない場合はtim_ti2fp2=tim_ti2）の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります（TIMx_CR1レジスタのCENビットが“1”）。2つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1レジスタのDIRビットがハードウェアによって変更されます。カウンタがtim_ti1のみ、tim_ti2のみ、またはtim_ti1とtim_ti2の両方をカウントしている場合でも、DIRビットは、いずれかの入力（tim_ti1またはtim_ti2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0とTIMx_ARRレジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0からARRまで、またはARRから0まで）。したがって、開始前にTIMx_ARRを設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、およびトリガ出力機能は、通常動作を続けます。エンコーダモードと外部クロックモード2は互換性がないので、同時に選択することはできません。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（tim_ti1とtim_ti2は同時に切り替わらないと想定しています）。

表563.カウント方向とエンコーダ信号（CC1P = CC2P = 0）

アクティブエッジ	SMS[3:0]	他方の信号のレベル（tim_ti2に対するtim_ti1fp1、tim_ti1に対するtim_ti2fp2）	tim_ti1fp1信号		tim_ti2fp2信号	
			立ち上がり	立下がり	立上がり	立下がり
tim_ti1のみのx1モードでカウント	1110	高	ダウン	アップ	カウントなし	カウントなし
		低	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti2のみのx1モードでカウント	1111	高	カウントなし	カウントなし	アップ	ダウン
		低	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti1のみのx2モードでカウント	0001	高	ダウン	アップ	カウントなし	カウントなし
		低	アップ	ダウン	カウントなし	ダウン
tim_ti2のみのx2モードでカウント	0010	高	カウントなし	カウントなし	アップ	ダウン
		低	カウントなし	カウントなし	ダウン	アップ
tim_ti1およびtim_ti2 x4モードでカウント	0011	高	ダウン	アップ	アップ	ダウン
		低	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCUに直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す3番目のエンコーダ出力は、外部トリガ入力に接続して、カウンタのリセットをトリガできます。

図640に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S = 01（TIMx_CCMR1レジスタ、tim_ti1fp1はtim_ti1に配置）
- CC2S = 01（TIMx_CCMR2レジスタ、tim_ti2fp2はtim_ti2に配置）
- CC1PおよびCC1NP = 0（TIMx_CCERレジスタ、tim_ti1fp1非反転、tim_ti1fp1=tim_ti1）
- CC2PおよびCC2NP = 0（TIMx_CCERレジスタ、tim_ti2fp2非反転、tim_ti2fp2=tim_ti2）
- SMS=0011（TIMx_SMCR　レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=1（TIMx_CR1レジスタ、カウンタ有効）

図640. エンコーダインタフェースモードにおけるカウンタの動作例

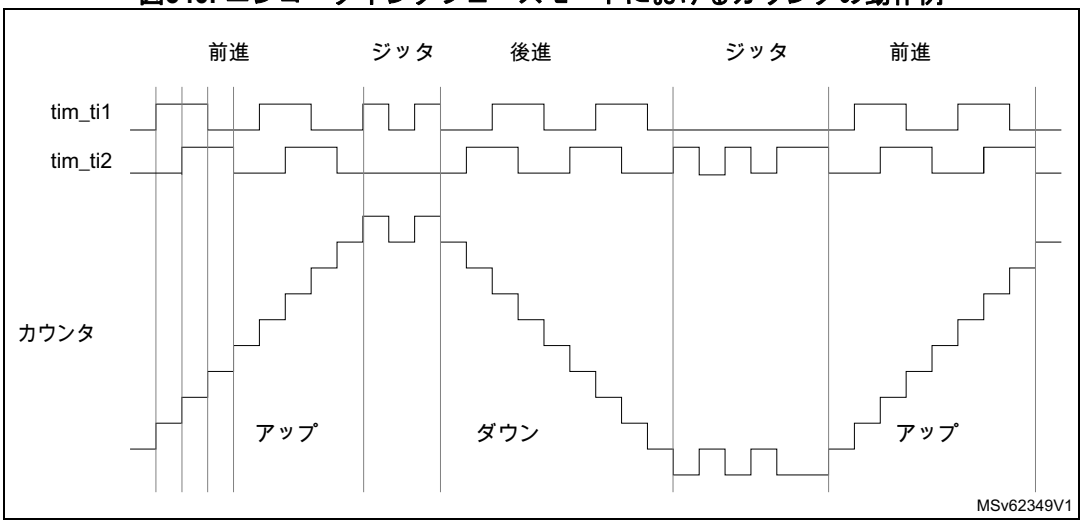
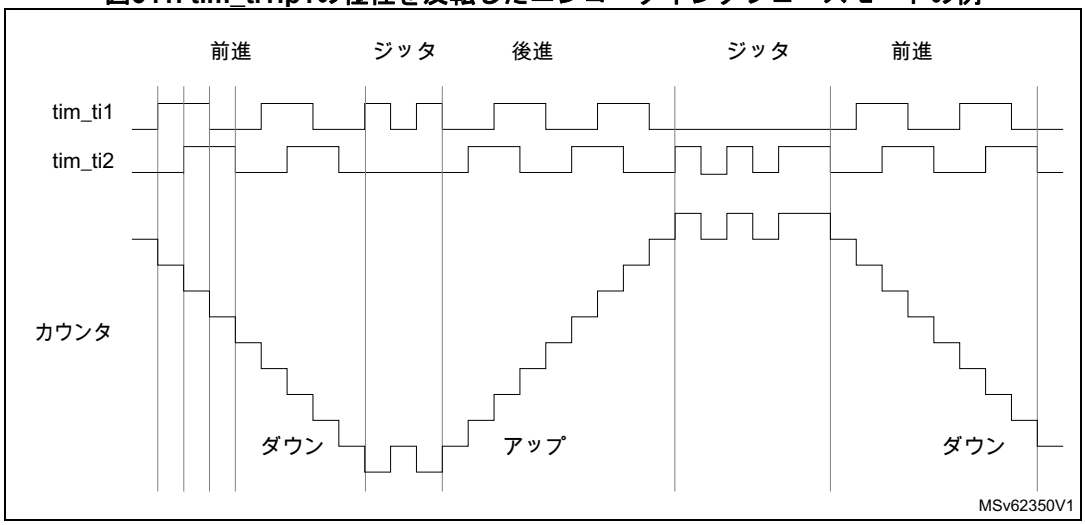


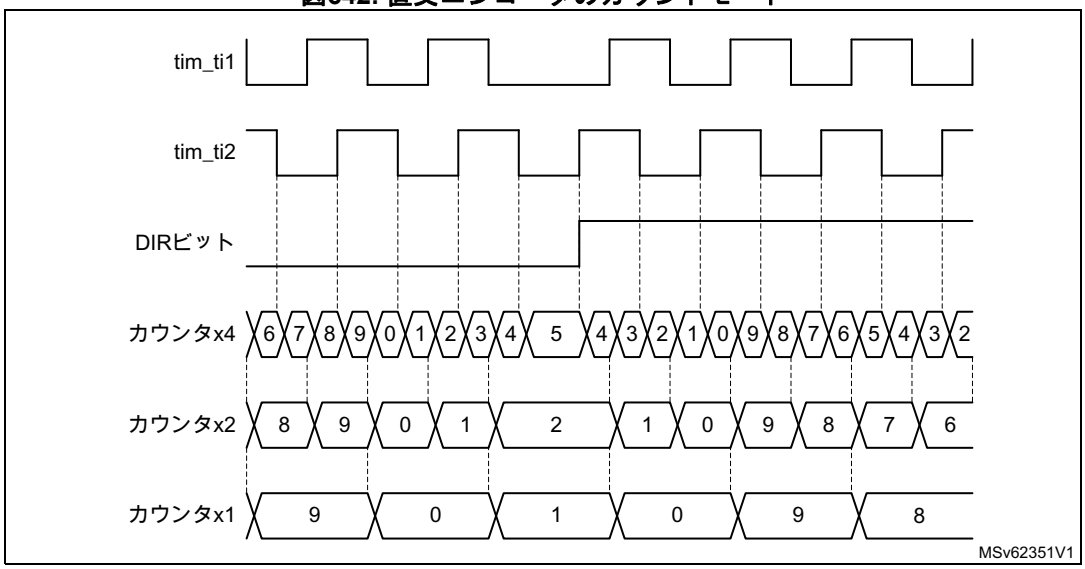
図641に、tim_ti1fp1の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図641. tim_ti1fp1の極性を反転したエンコーダインタフェースモードの例



以下の図642 は、さまざまなカウントモードでの速度反転中のタイマカウンタ値を示しています。

図642. 直交エンコーダのカウントモード



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した2番目のタイマを使用して、2つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合は、カウンタの値を3番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。また、DMAリクエストを通じて、この値を読み出すこともできます。

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPYフラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIFとUIFCPYフラグのアサートの間には、遅延はありません。

32bitのタイマの実装で、IUFREMAPビットがセットされている場合、カウンタのビット31は読出しアクセス時にUIFCPYフラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

クロックプラス方向エンコーダモード

直交エンコーダモードのほか、タイマは他のタイプのエンコーダもサポートしています。

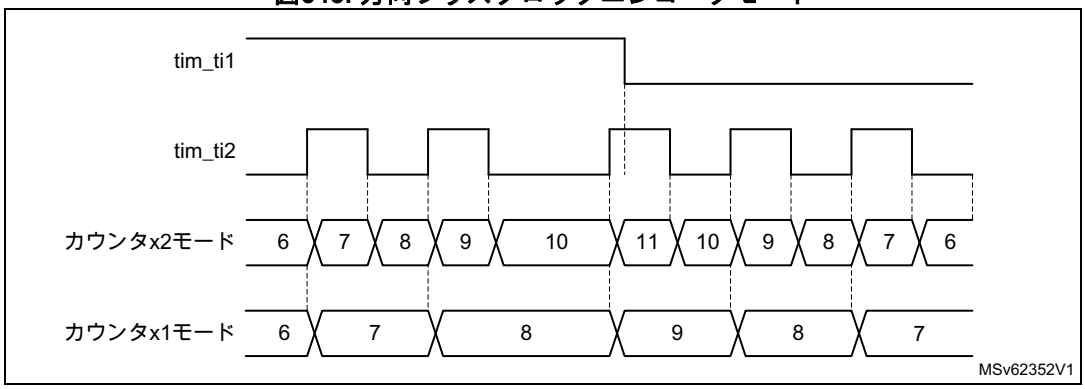
図643に示す「クロックプラス方向」モードでは、tim_ti2の単一ラインでクロックが提供され、tim_ti1入力を使用して方向が強制されます。

このモードはTIMx_SMCRレジスタのSMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1010：X2モード。カウンタはクロックの立上がりと立下がりの両エッジで更新されます。
- 1011：x1 モード。カウンタは CC2P ビット値に従って単一のクロックエッジで更新されます。CC2P = 0は立ち上がりエッジ検出に対応し、CC2P = 1は立ち下がりエッジ検出に対応します。

tim_ti1の方向信号の極性は、CC1Pビットで設定されます。0は正の極性（tim_ti1がハイのときアップカウントで、tim_ti1がローのときダウンカウント）に対応し、CC1P = 1は負の極性（tim_ti1がローのときアップカウント）に対応します。

図643. 方向プラスクロックエンコーダモード



方向クロックエンコーダモード

図644の「方向クロック」モードでは、クロックは2本のラインで提供され、1本のアップカウントクロックラインと1本のダウンカウントクロックラインを持つというように、方向に応じて一度に1つが提供されます。

このモードはTIMx_SMCRレジスタのSMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1100 : x2モード。カウンタは、2つのクロックラインのいずれかの立上がりと立下がりの両エッジで更新されます。CC1PおよびCC2Pビットは、クロックアイドル状態用のコーディングです。CCxP = 0はハイレベルアイドル状態に対応し（下記図644を参照）、CCxP = 1はローレベルアイドル状態に対応します（下記図645を参照）。
- 1101 : x1モード。カウンタは、CC1PおよびCC2Pビット値に従って、単一のクロックエッジで更新されます。CCxP = 0は立下がり検出とハイレベルアイドル状態に対応し（下記図644を参照）、CCxP = 1は立上がり検出とローレベルアイドル状態に対応します（下記図645を参照）。

図644. 方向クロックエンコーダモード（CC1P = CC2P = 0）

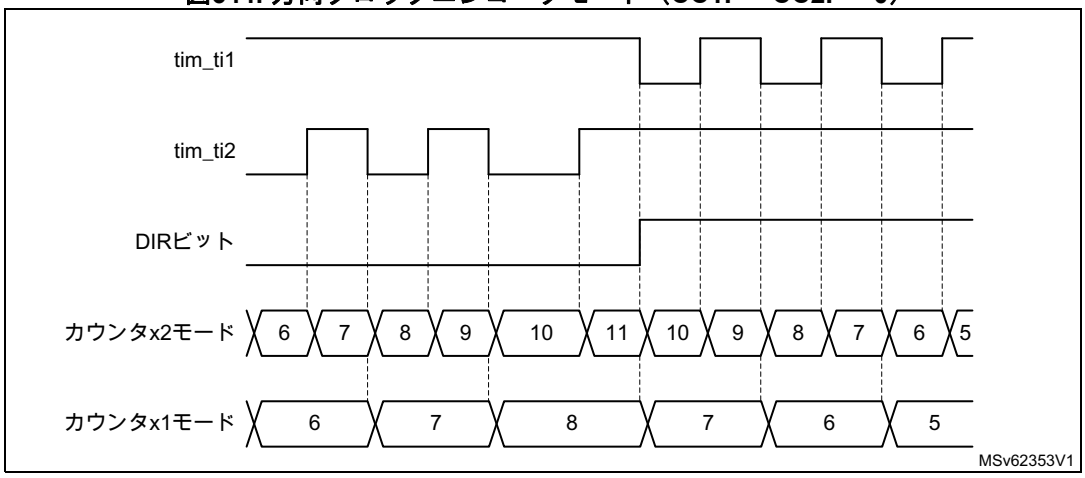
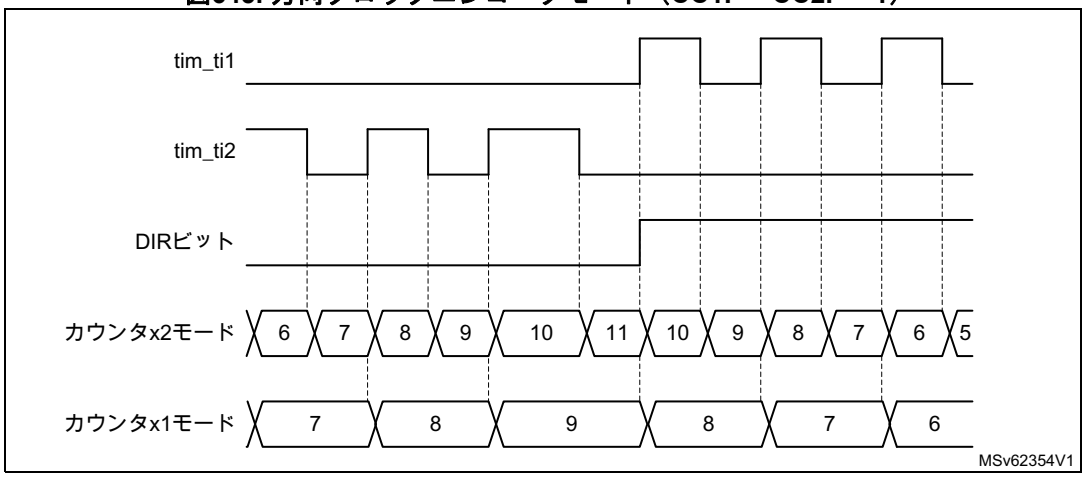


図645. 方向クロックエンコーダモード（CC1P = CC2P = 1）



以下の表 564に、任意の入力遷移に対して方向クロックモードがどのように動作するかを詳しく説明します。

表564.カウント方向 対 エンコーダ信号および極性設定

方向クロックモード	SMS[3:0]	他方の信号のレベル（tim_ti2に対するtim_ti1fp1、tim_ti1に対するtim_ti2fp2）	tim_ti1fp1信号		tim_ti2fp2信号	
			立ち上がり	立下がり	立上がり	立下がり
x2モード CCxP=0	1100	高	ダウン	ダウン	アップ	アップ
		低	カウントなし	カウントなし	カウントなし	カウントなし
x2モード CCxP=1	1100	高	カウントなし	カウントなし	カウントなし	カウントなし
		低	ダウン	ダウン	アップ	アップ
x1モード CCxP=0	1101	高	カウントなし	ダウン	カウントなし	アップ
		低	カウントなし	カウントなし	カウントなし	カウントなし
x1モード CCxP=1	1101	高	カウントなし	カウントなし	カウントなし	カウントなし
		低	ダウン	カウントなし	アップ	カウントなし

インデックス入力

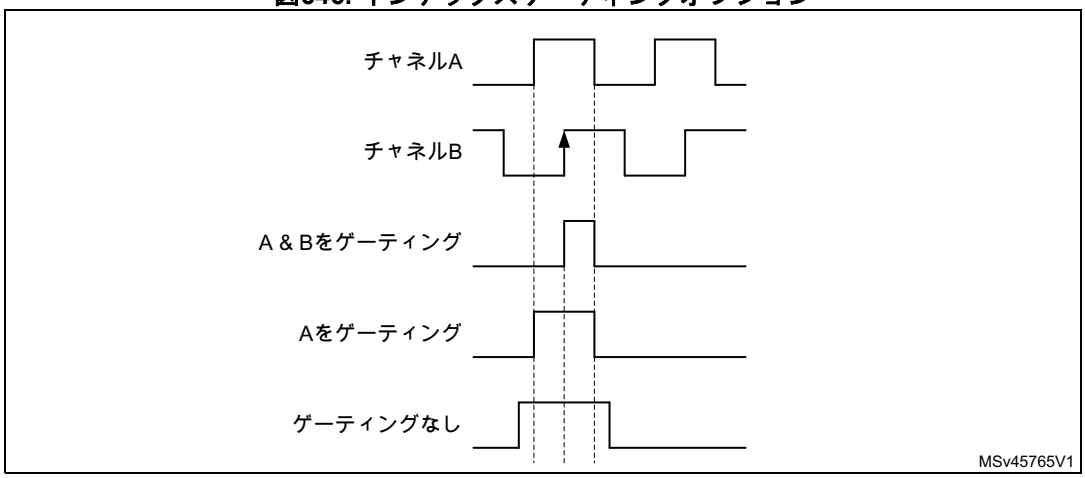
カウンタは、絶対基準位置を示すエンコーダからのインデックス信号によってリセットできます。インデックス信号は、tim_etr_in入力に接続する必要があります。これは、デジタル入力フィルタを使用してフィルタリングできます。

インデックス機能は、TIMx_ECRレジスタのIEビットで有効にされます。IEビットは、SMS[3:0] ビットフィールドが0001、0010、011、1010、1011、1100、1101、1110、1111の値であるときに、エンコーダモードでのみセットする必要があります。

以下の図646に示すように、インデックスパルス調整のいくつかのオプションを備えた市販のエンコーダが提案されています。

- AおよびBでゲート処理：パルス幅は1つのチャネル周期の1/4であり、AおよびBの両方のエッジにそろえられます。
- Aでゲート処理（またはBでゲート処理）：パルス幅は1つのチャネル周期の1/2であり、チャネルA（またはチャネルB）の2つのエッジにそろえられます。
- ゲート処理なし：パルス幅は最大1チャネル周期であり、どのエッジにもそろえられません。

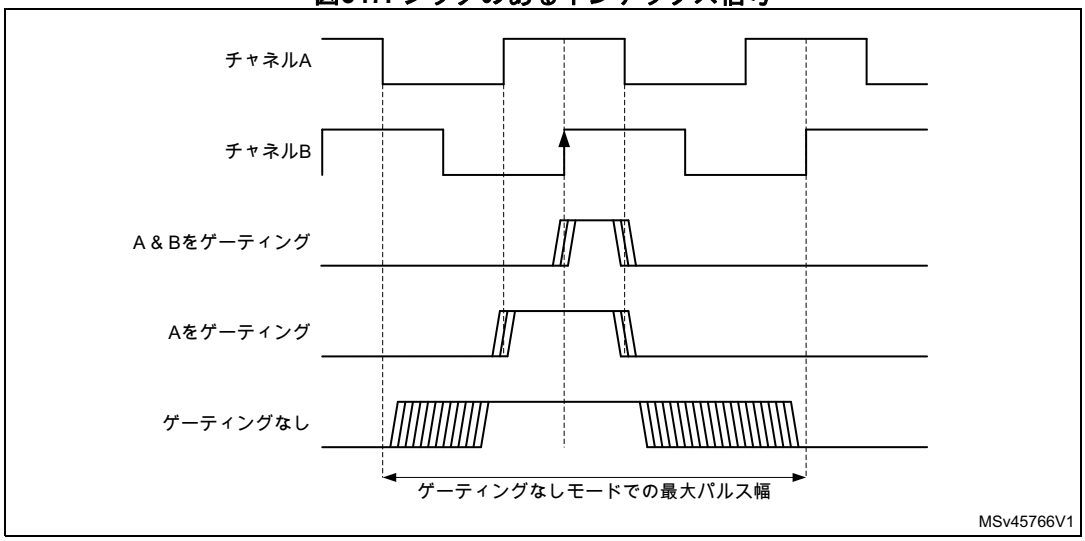
図646. インデックスゲーティングオプション



以下の図647に示すように、ゲーティングモードに関係なく、回路はインデックス信号のジッタを許容します。

ゲート処理なしモードでは、信号はエンコーダの2周期より厳密に小さくなければなりません。パルス幅がエンコーダの2周期以上の場合、カウンタは複数回リセットされます。

図647. ジッタのあるインデックス信号



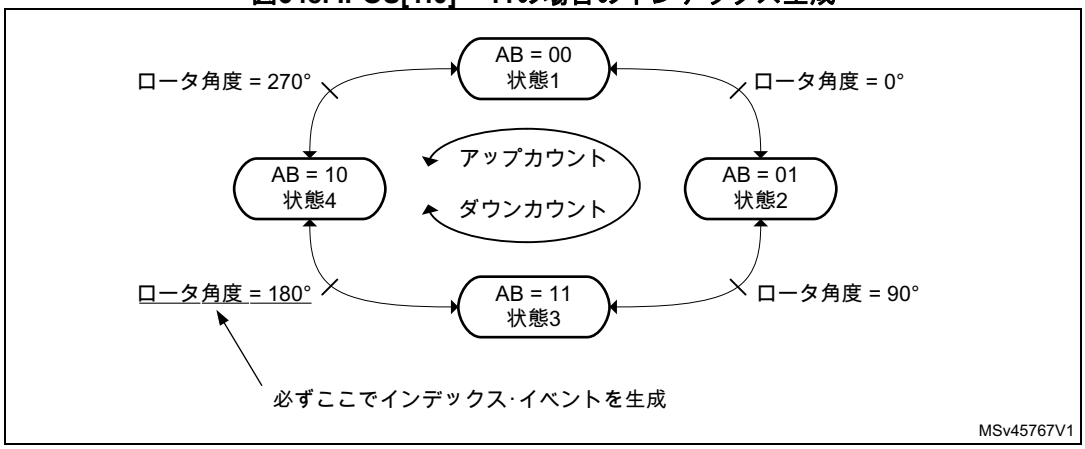
タイマは、特定のプログラミングを必要とせずに、3つのゲーティングオプションを同様にサポートします。TIMx_ECRレジスタのIPOS[1:0] ビットフィールドを使用して、インデックスを同期する必要があるエンコーダ状態（チャネルAとチャネルBの状態の組み合わせ）を定義するだけです。

インデックス検出イベントは、速度の反転中に対称動作を確保するために、カウント方向に応じて異なる動作をします。

- カウンタはアップカウント中（DIRビット = 0）に、リセットされます。
- カウンタはダウンカウント時に、TIMx_ARRに設定されます。

これにより、カウント方向に関係なく、まったく同じ機械的角度位置でインデックスを生成できます。以下の図648は、単純な例（機械的回転ごとに4つのエッジを提供するエンコーダ）について、生成されるインデックスの位置を示しています。

図648. IPOS[1:0] = 11の場合のインデックス生成



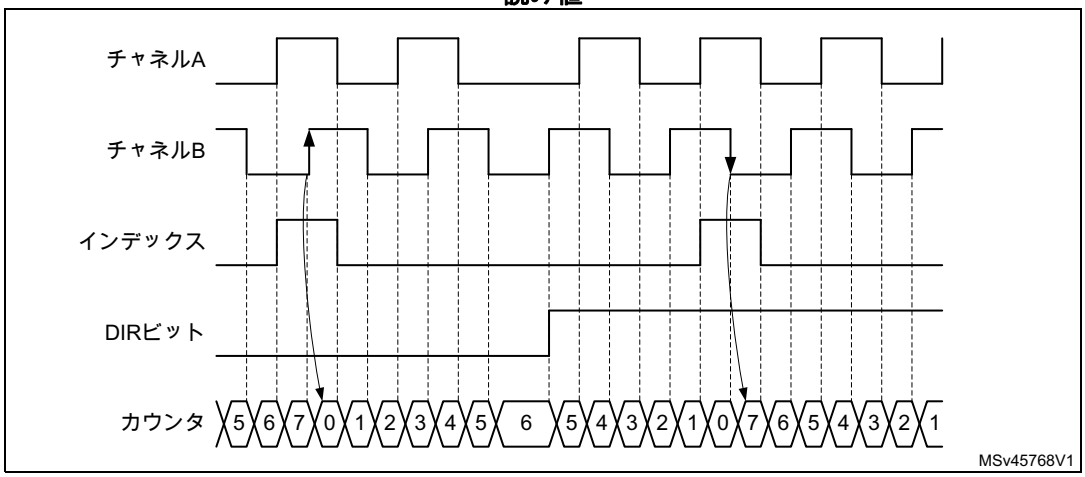
以下の図649に、IPOS[1:0] = 11の場合の波形と対応する値を示します。カウンタの値が強制される瞬間は、カウント方向に応じて自動的に調整されることを示しています。

- アップカウント（DIRビット = 0）の場合、エンコーダの状態が「11」（ChA = 1、ChB = 1）のとき、カウンタは0にセットされます。
- ダウンカウント（DIRビット = 1）の場合、「11」状態を終了するとき、カウンタはTIMx_ARRにセットされます。

インデックス検出イベント時に割込みを発行できます。

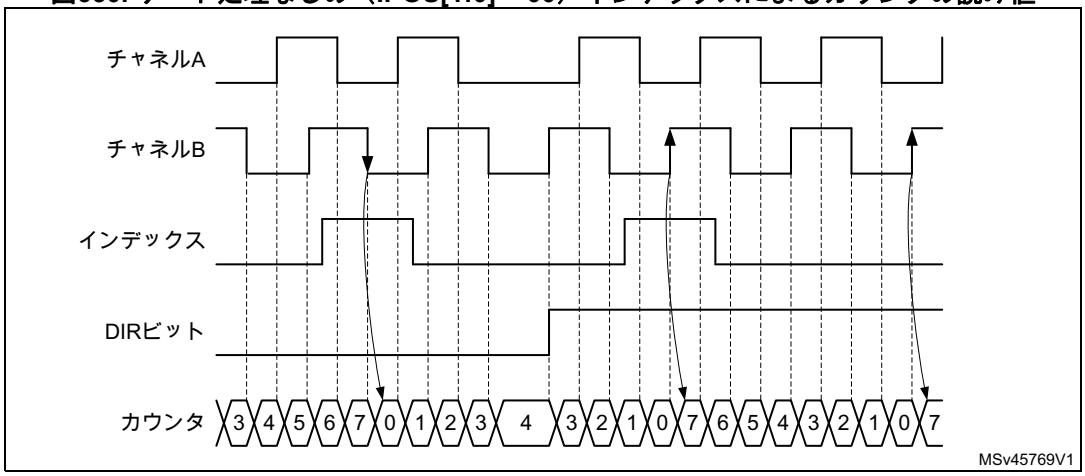
矢印は、どの遷移でインデックスイベント割込みが生成されるかを示しています。

図649. チャネルAでゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値



以下の図650に、ゲート処理なしモードの場合の波形と対応する値を示します。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図650. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値



以下の図651は、さまざまなパルスアライメントシナリオでの「A & Bでゲート処理」モードの処理方法を示しています。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図651. チャネルAおよびBでゲート処理されたインデックスによるカウンタの読み値

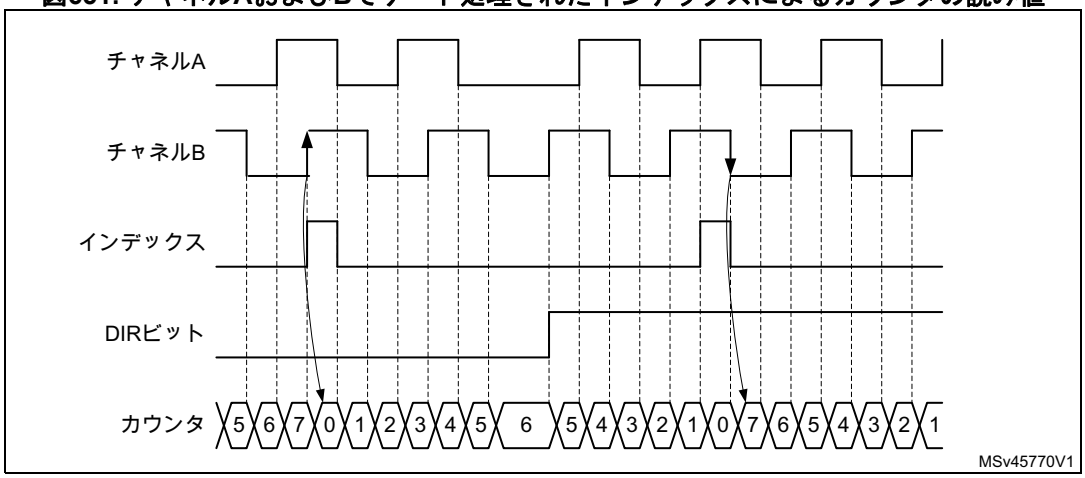


図652と図653には、後続のインデックスパルスが、エンコーダクロック周期の4分の1よりも狭いことがあるケースが詳しく示されています。

図652. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11）

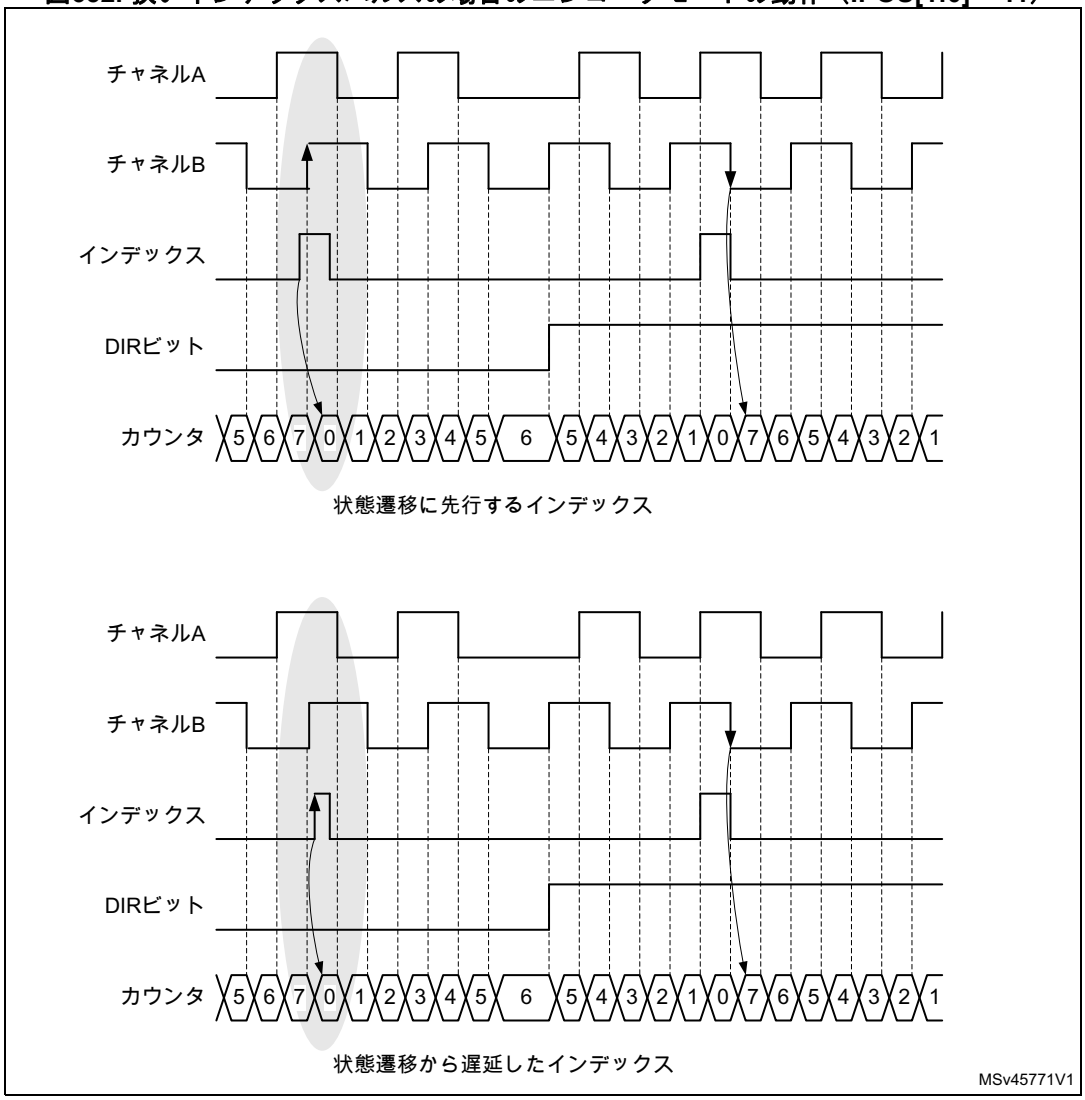
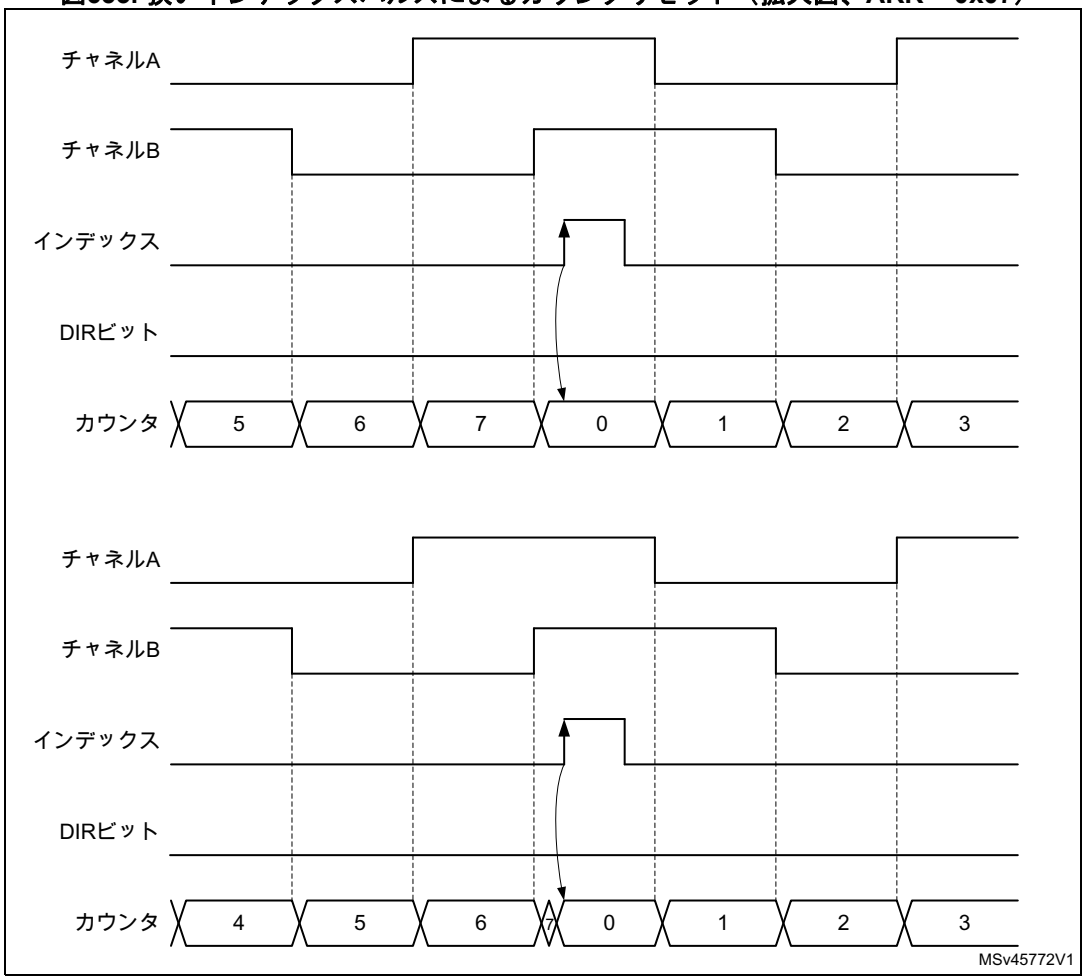
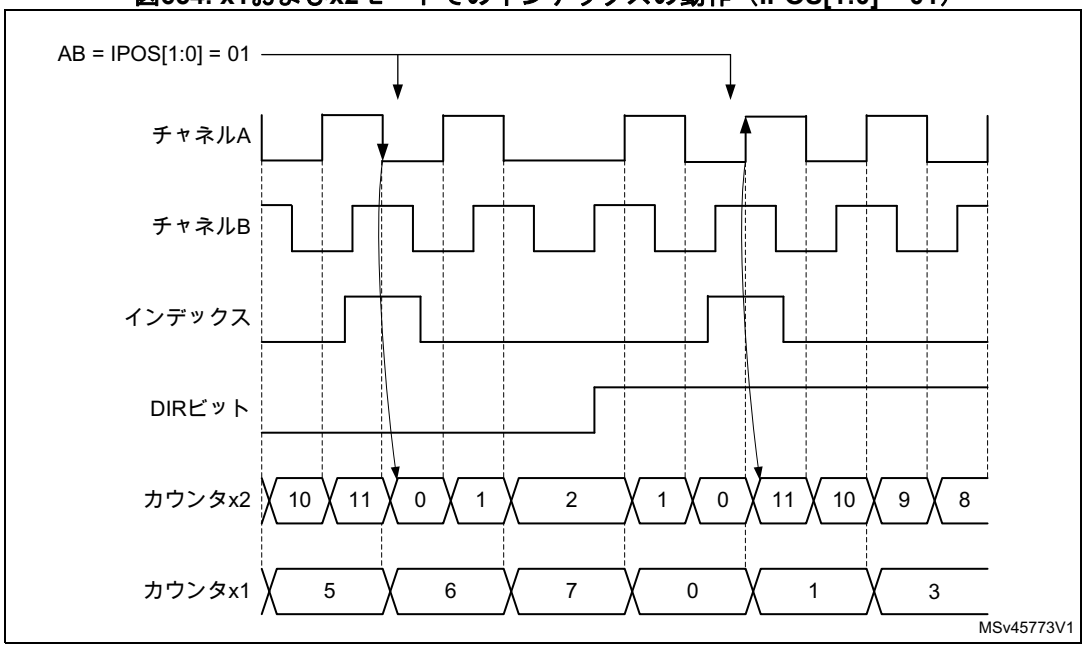


図653. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07）



下記の図654は、x1およびx2モードでインデックスがどのように管理されるかを示します。

図654. x1およびx2モードでのインデックスの動作（IPOS[1:0] = 01）

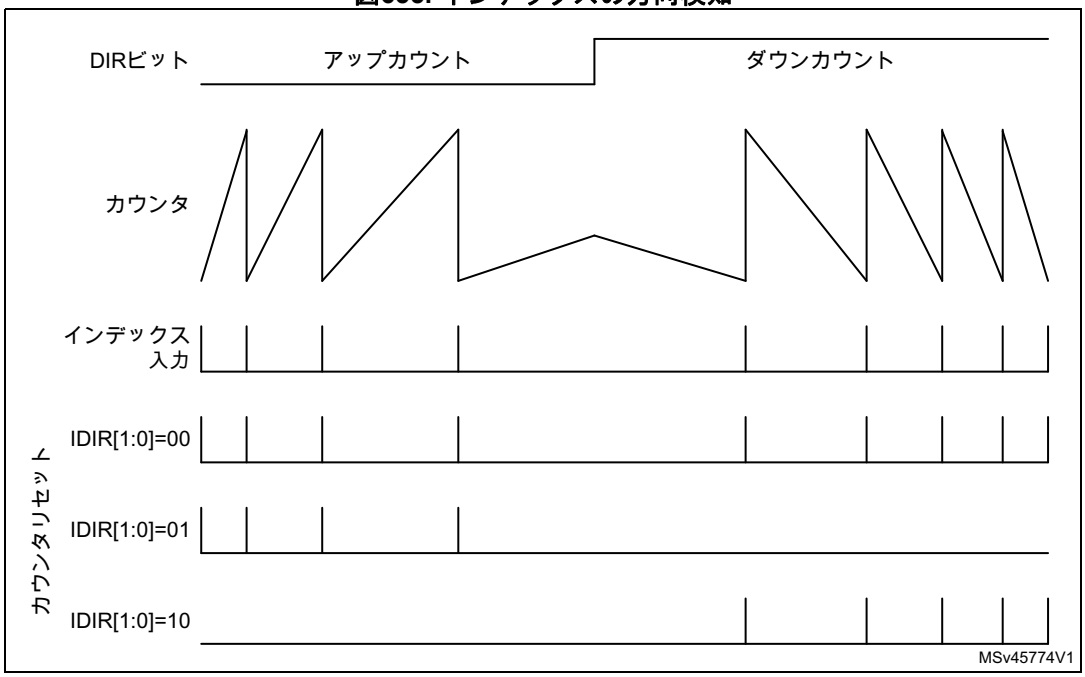


インデックスの方向検知

TIMx_ECRレジスタのIDIR[1:0] ビットフィールドにより、選択したカウント方向でのみインデックスをアクティブにできます。

以下の図655は、IDIR[1:0] 値に応じた、インデックスとカウンタのリセットイベントの関係を示しています。

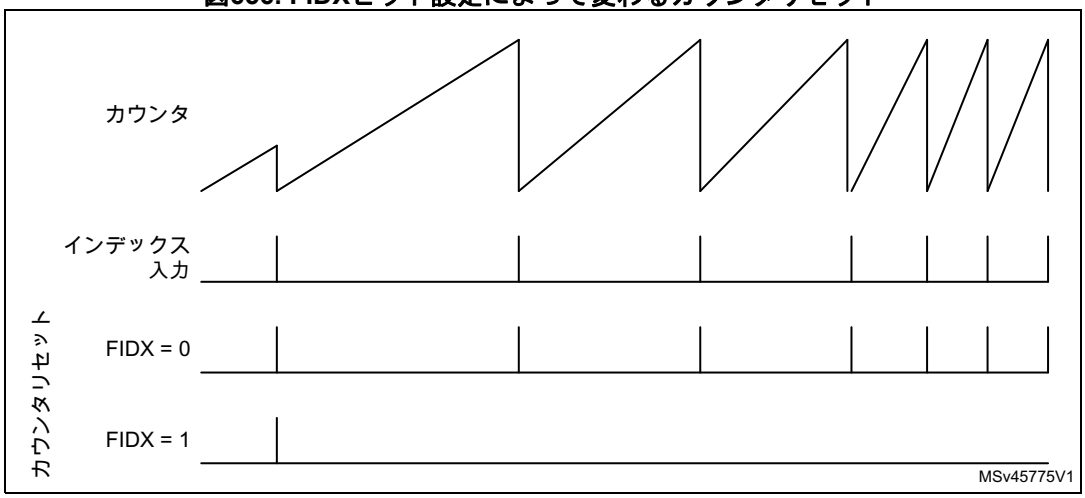
図655. インデックスの方向検知



特殊な最初のインデックスイベント管理

TIMx_ECRレジスタのFIDXビットによって、以下の図656に示すように、インデックスを1回だけ取得できます。最初のインデックスが到着すると、その後のインデックスは無視されます。必要に応じて、FIDXビット に0を書き込み、再度1にセットすることにより、回路を再設定できます。

図656. FIDXビット設定によって変わるカウンタリセット



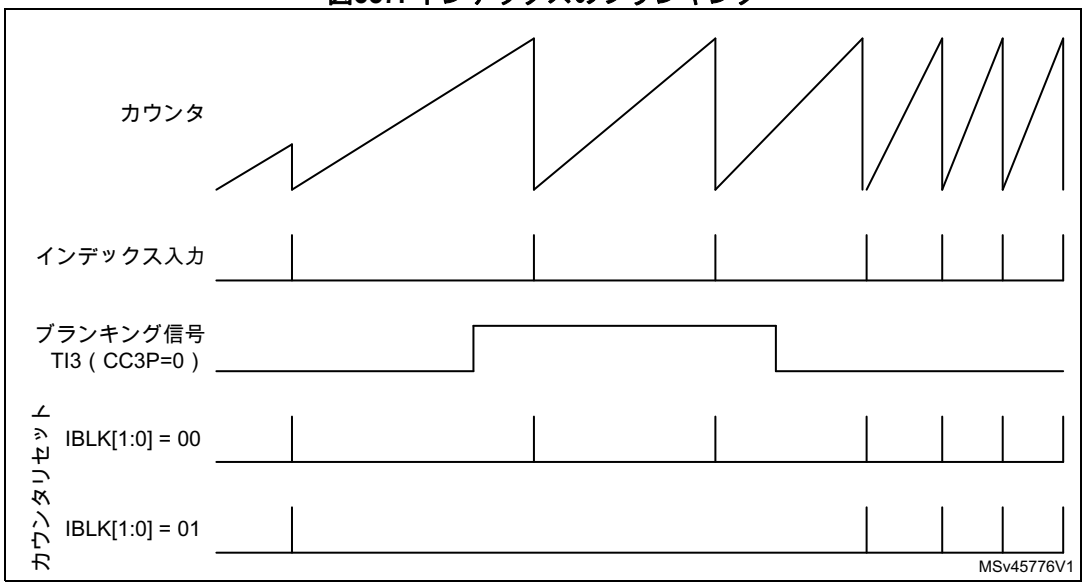
インデックスのブランキング

インデックスイベントは、tim_ti3またはtim_ti4の入力を使用してブランキングすることができます。ブランキング期間中は、以下の図657に示すように、インデックスイベントがカウンタをリセットすることはありません。

このモードはTIMx_ECRレジスタのIBLK[1:0] ビットフィールドを使用して、以下のように有効にされます。

- IBLK[1:0] = 00：インデックス信号は常にアクティブです。
- IBLK[1:0] = 01：インデックス信号は、tim_ti3入力時にブランキング処理されます。
- IBLK[1:0] = 10：インデックス信号は、tim_ti4入力時にブランキング処理されます。

図657. インデックスのブランキング



非直交モードでのインデックス管理

以下の図658および図659では、SMS[3:0] ビットフィールドが1010、1011、1100、1101に等しい場合に、方向クロックモードおよびクロックプラス方向モードでインデックスがどのように管理されるかを詳しく説明します。

これらのモードの両方で、インデックスの検出は次のようにIPOS[0] ビットで設定されます。

- IPOS[0] = 0：インデックスはクロックのローレベルで検出されます。
- IPOS[0] = 1：インデックスはクロックのハイレベルで検出されます。

IPOS[1] ビットは意味がありません。

図658. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1）

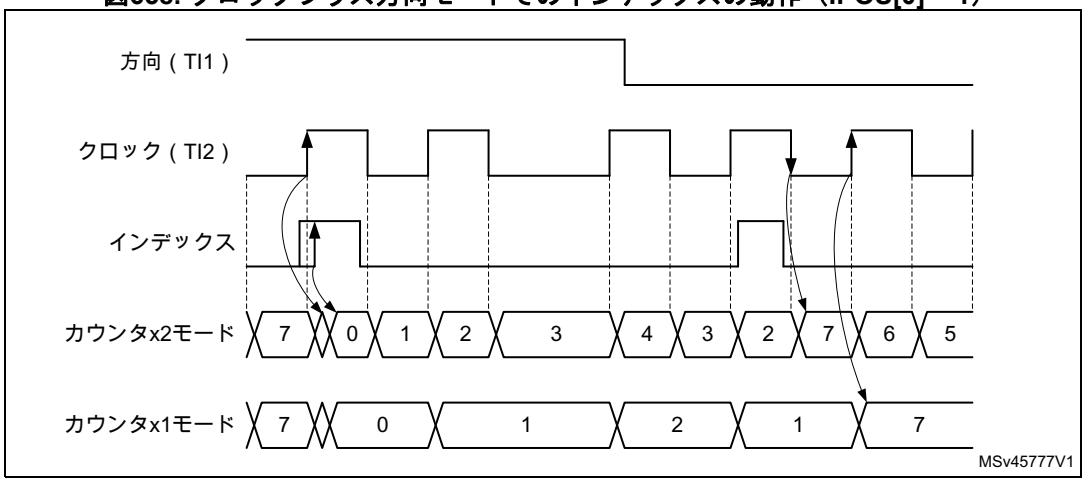
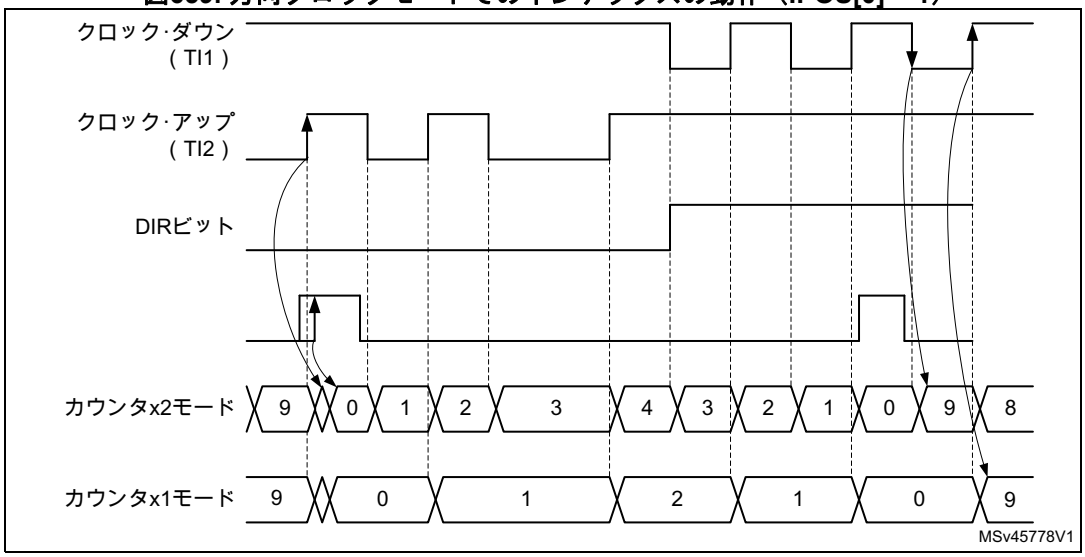


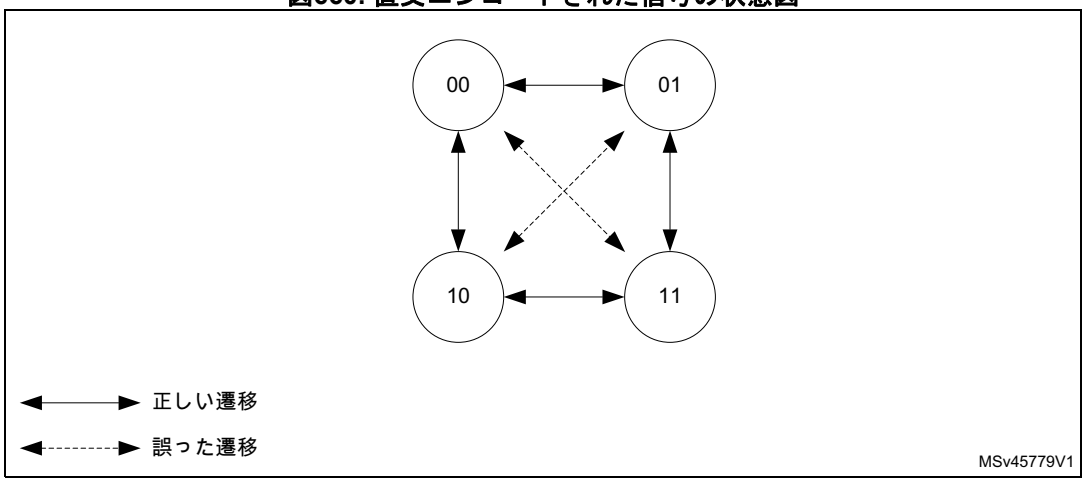
図659. 方向クロックモードでのインデックスの動作（IPOS[0] = 1）



エンコーダのエラー管理

2つの直交信号が使用できるエンコーダ構成の場合、遷移エラーを検出できます。2つの入力の読み値は、以下の図660で、状態図として表すことができる2bitのグレーコードに対応しています。一度に1つのビットだけが変化することが予測されています。間違った遷移があると、TIMx_SRステータスレジスタのTERRF割込みフラグがセットされます。TIMx_DIERレジスタのTERRIEビットがセットされていると、遷移エラー割込みが生成されます。

図660. 直交エンコードされた信号の状態図



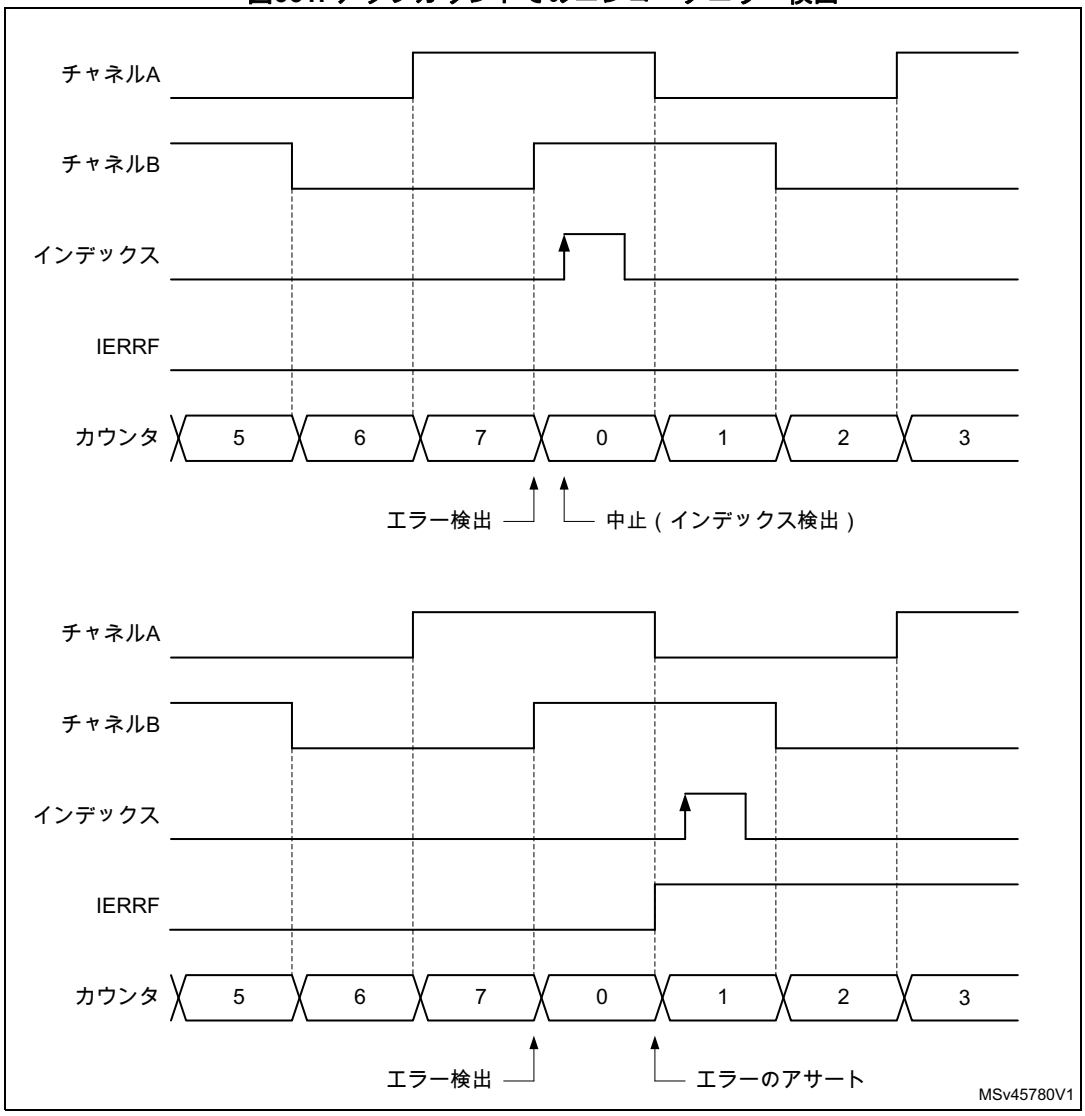
インデックス信号を持つエンコーダの場合、1回転あたりのパルスが過剰になる異常動作を検出できます。1回転あたりN個のパルスを持つエンコーダでは、1回転あたり4xNカウントが提供されます。インデックス信号によって、4xNクロック周期ごとにカウンタがリセットされます。

カウンタ値がインデックスイベントなしにTIMx_ARRから0にインクリメントされるか、0からTIMxARR値にデクリメントされる場合、これはインデックス位置エラーとして報告されます。

オーバーフロー閾値は、TIMx_ARRレジスタを使ってプログラムします。1000行のエンコーダでは、カウンタ値が0?3999になります（4x読み取りモードの場合）。オーバーフロー検出閾値は、TIMx_ARR = 3999 + 1 = 4000を設定してプログラムする必要があります。

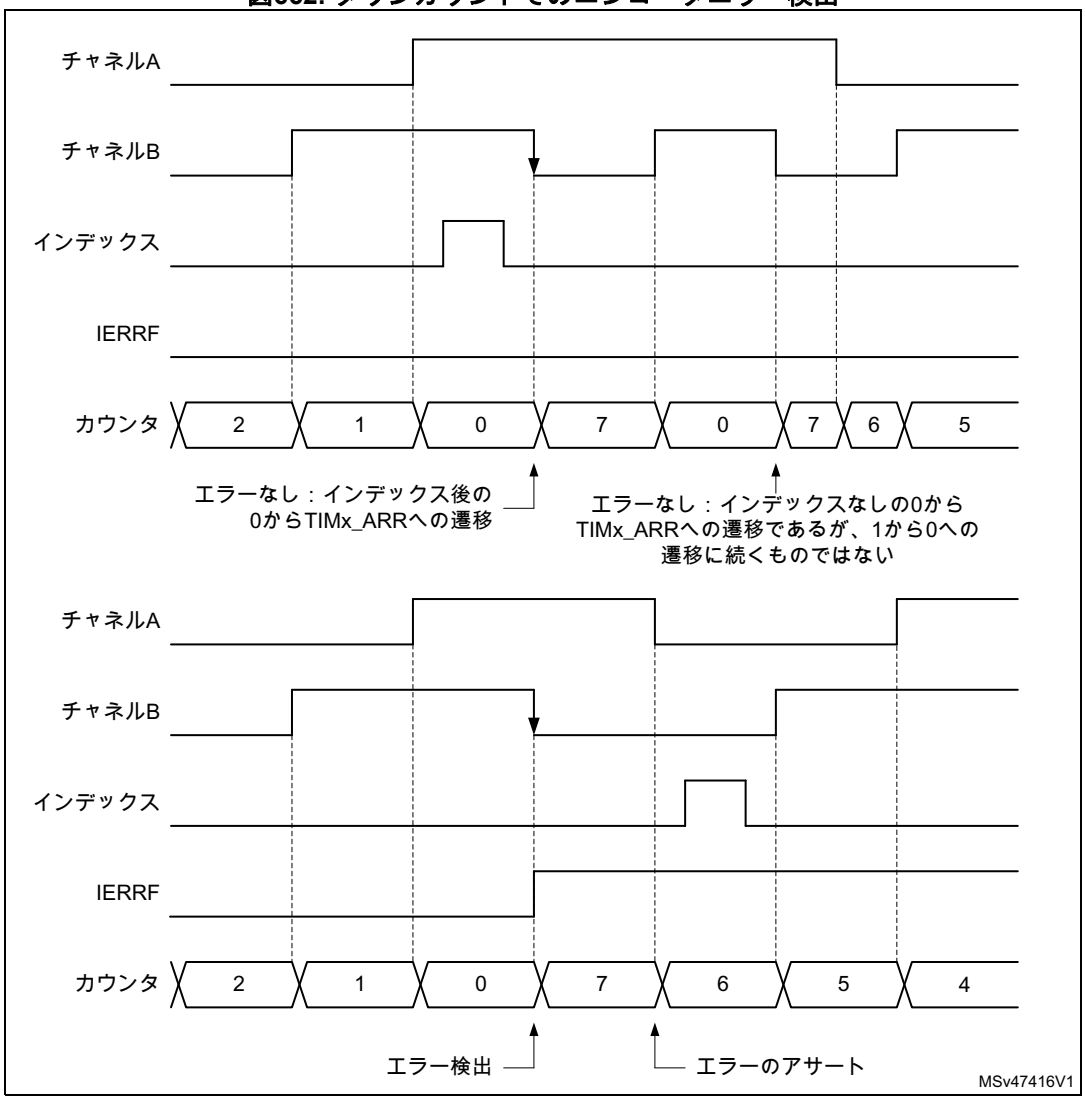
アップカウントでは、エラーアサーションは0から1への遷移まで遅延されます。これは、以下の図661に示すように、AおよびBゲート処理モードでの狭いインデックスパルスへの対処です。

図661. アップカウントでのエンコーダエラー検出



ダウンカウントモードでは、検出は1から0への事前遷移によって条件付けられます。これは、以下の図662に示すように、AおよびBゲート処理モードでの狭いインデックスパルスに対処し、インデックス検出直後にエンコーダがTIMx_ARRと0の間でディザリングする場合の誤ったエラー検出を回避するためです。

図662. ダウンカウントでのエンコーダエラー検出



インデックスエラーによって、TIMx_SRステータスレジスタのIERRF割込みフラグがセットされます。TIMx_DIERレジスタのIERRIEビットがセットされていると、インデックスエラー割込みが生成されます。

エンコーダの機能的割込み

エンコーダモードでは、以下の割込みも使用できます。

- 方向変更：エンコーダモードでのカウント方向の変更によって、TIMx_CR1レジスタのDIRビットが反転します。方向変更によって、TIMx_SRステータスレジスタのDIRF割込みフラグがセットされます。TIMx_DIERレジスタのDIRIEビットがセットされていると、方向変更割込みが生成されます。
- インデックスイベント：インデックスイベントによって、TIMx_SRステータスレジスタのIDXF割込みフラグがセットされます。TIMx_DIERレジスタのIDXIEビットがセットされていると、インデックス割込みが生成されます。

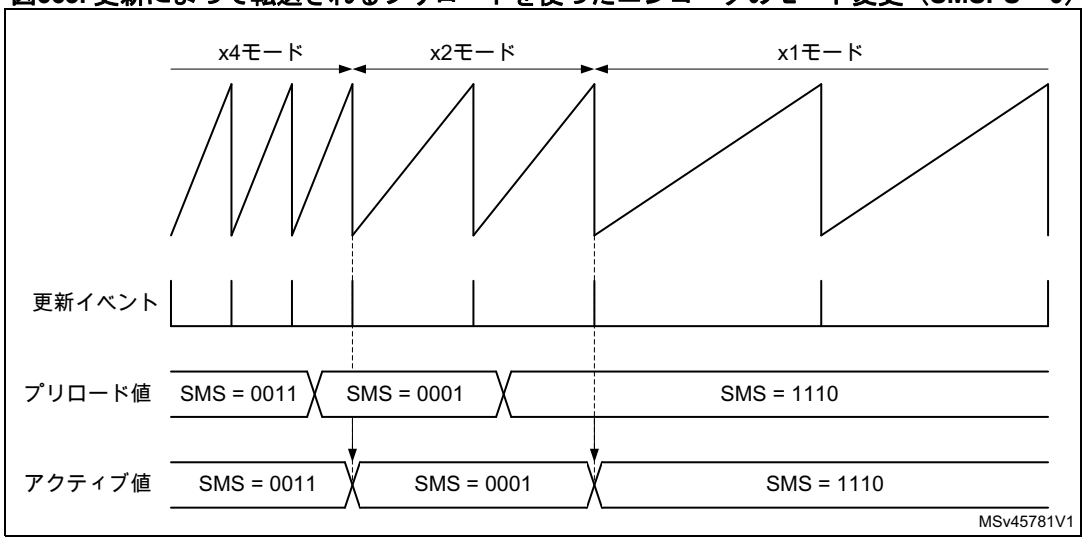
実行時エンコーダモード更新のためのスレーブモード選択プリロード

実行中に、あるエンコーダモードを別のモードに切り替える必要があることがあります。以下の図663に示すように、これは通常、高速時に更新割込みレートを低下させるために、x4モードからx2に、x1に切り替えることにより、実行されます。

この目的のために、SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCRレジスタのSMSPEイネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、TIMx_SMCRレジスタのSMSPSビットで選択できます。

- SMSPS = 0：転送は、アップカウント時にカウンタがオーバーフローするとき、およびダウンカウント時にアンダーフローするときに発生する更新イベント（UEV）によってトリガされます。
- SMSPS = 1：転送は、インデックスイベントによってトリガされます。

図663. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）



エンコーダクロック出力

エンコーダモードの動作原理は、十分な数のクロックエッジ取得と正確な測定を行うために比較的長い積分時間を必要とするため、低速においては、高分解能の速度測定には完全には適していません。

低速でのより良いソリューションは、エッジ間のクロック周期測定を行うことです。これは、スレーブタイマを使用して達成できます。タイマによって、tim_trgo出力にエンコーダクロック情報を出力できます。その後、スレーブタイマはエンコーダクロックエッジごとに周期測定を実行し、速度情報を提供できます。

このモードはTIMx_CR2レジスタのMMS[3:0] ビットフィールドを1000に設定することによって有効になります。これは、SMS[3:0] 値、0001、0010、0011、1010、1011、1100、1101、1110、1111に対して有効です。その他のSMS[3:0] コードは許されず、予想外の動作につながることがあります。

55.4.19 方向ビットの出力

tim_oc3およびtim_oc4出力信号に、タイマから方向信号を出力できます（TIMx_CR1レジスタのDIRビットのコピー）。これを達成するには、TIMx_CCMR2レジスタのOC3M[3:0] またはOC4M[3:0] ビットフィールドを1011にセットします。

この機能を使用すると、エンコーダモードでカウント方向（または回転方向）を監視すること、またはセンタアラインPWMモードでアップ／ダウンフェーズを示す信号を得ることができます。

55.4.20 UIFビットの再配置

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）を連続コピーします。これは、UIFCPYフラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取るために使用されます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIFとUIFCPYフラグのアサートの間には、遅延はありません。

32bitのタイマの実装で、IUFREMAPビットがセットされている場合、カウンタのビット31は読出しアクセス時にUIFCPYフラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

55.4.21 タイマ入力XOR機能

TIM1xx_CR2レジスタのTI1Sビットによって、チャネル1の入力フィルタを3つの入力ピンtim_ti1、tim_ti2、およびtim_ti3を結合するXORゲートの出力に接続できます。

XOR出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

この機能をホールセンサのインタフェースに使用した例をセクション 54.3.29: 2154ページのホールセンサとのインタフェースに示します。

55.4.22 タイマと外部トリガの同期

TIMxタイマは、リセットモード、ゲートモード、トリガモード、リセット + トリガモード、およびゲート + リセットモードで、外部トリガに同期できます。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1レジスタのURSビットがローの場合は、更新イベントUEVが生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

次の例では、tim_ti1入力の立上がりエッジに応じて、アップカウンタがクリアされます。

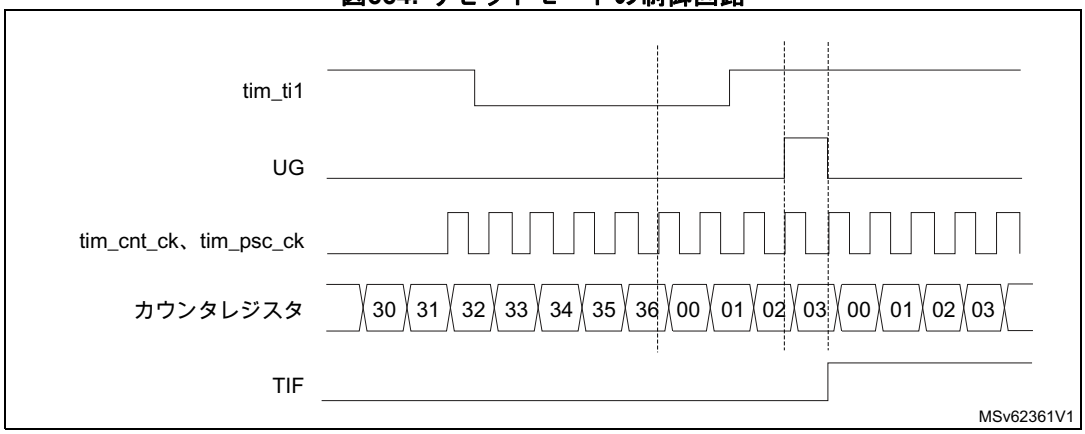
1. tim_ti1の立上がりエッジを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01）。TIMx_CCERレジスタでCC1P= 0とCC1NP = 0を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。

2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
3. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1の立上がりエッジまで通常の動作を行います。tim_ti1が立ち上がると、カウンタはクリアされ、0からリスタートします。同時に、トリガフラグがセットされ（TIMx_SRレジスタのTIFビット）、有効な場合は割込みリクエストまたはDMAリクエストを送信できます（TIMx_DIERレジスタのTIEおよびTDEビットに依存）。

次の図は、自動再ロードレジスタTIMx_ARR=0x36の場合の動作を示します。tim_ti1の立上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1入力の同期回路によるものです。

図664. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

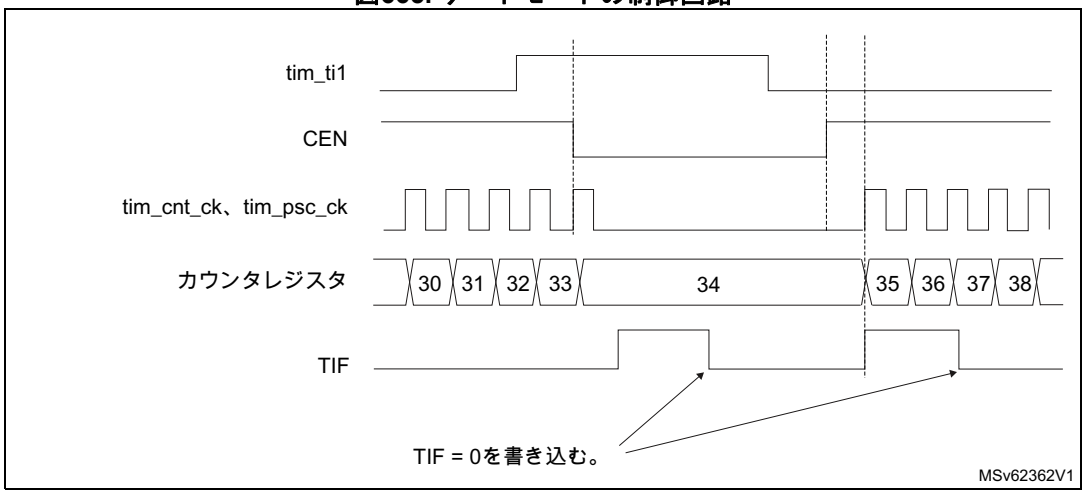
次の例では、アップカウンタはtim_ti1入力がローのときだけカウントします。

1. tim_ti1のローレベルを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01bit）。TIMx_CCERレジスタでCC1P = 1とCC1NP = 0を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1がローになると内部クロックでカウントを開始して、tim_ti1がハイになると停止します。TIMx_SRレジスタのTIFフラグは、カウンタの開始時と停止時にセットされます。

tim_ti1の立上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1入力の再同期回路によるものです。

図665. ゲートモードの制御回路



注： **ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。**

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

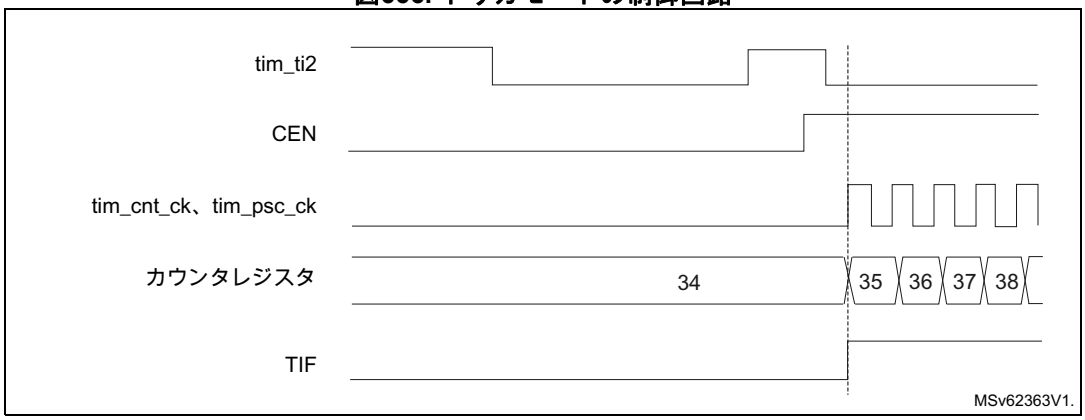
次の例では、アップカウンタは、tim_ti2入力の立上がりエッジに応じて開始します。

1. tim_ti2の立上がりエッジを検出するように、チャネル2を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC2S=01）。TIMx_CCERレジスタでCC2P = 1とCC2NP = 0を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCRレジスタにTS=00110を書き込むことによって、入力ソースとしてtim_ti2を選択します。

tim_ti2で立上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIFフラグがセットされます。

tim_ti2の立上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2入力の再同期回路によるものです。

図666. トリガモードの制御回路



実行時エンコーダモード更新のためのスレーブモード選択プリロード

SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCRレジスタのSMSPEイネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、カウンタがオーバーフローするときに発生する更新イベント（UEV）です。

スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード - ゲートモードとリセットモードの組み合わせ

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードは、範囲外のPWM信号（最大期待値を超えるデューティサイクル）を検出するために使用されます。

スレーブモード – 外部クロックモード2 + トリガモード

外部クロックモード2は、他のスレーブモードとともに使用できます（ただし、外部クロックモード1とエンコーダモードは除きます）。この場合、tim_etr_in信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCRレジスタのTSビットを通じてtim_trgiとしてtim_etr_inを選択しないようにしてください。

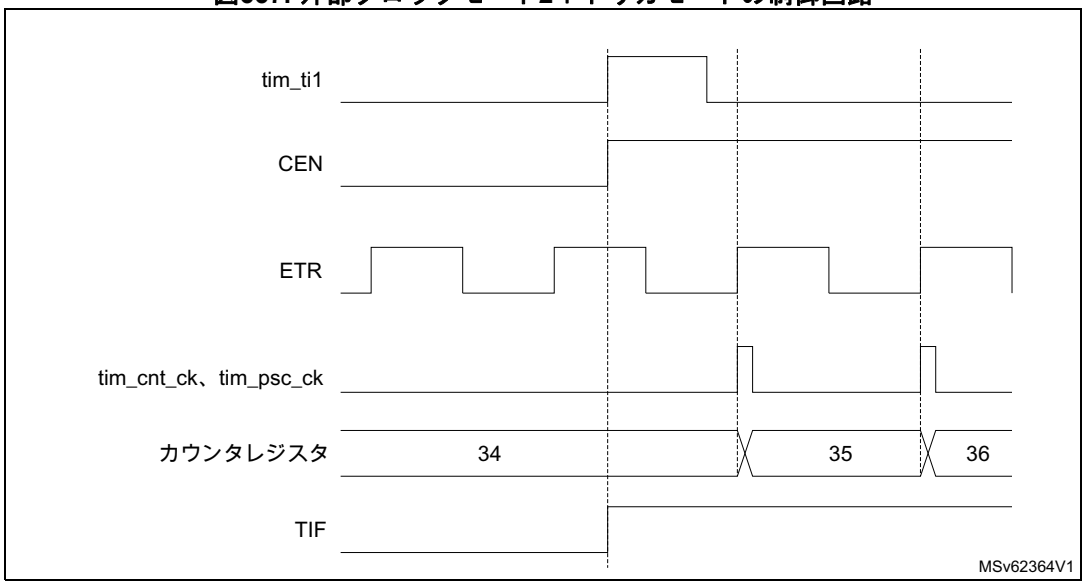
次の例では、アップカウンタは、tim_ti1の立上がりエッジが発生すると、tim_etr_in信号の立上がりエッジのたびにインクリメントされます。

1. TIMx_SMCRレジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0 : tim_etr_inの立上がりエッジを検出。ECE=1で外部クロックモード2を有効にします。
2. TI1の立上がりエッジを検出するように、チャネル1を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1レジスタのCC1S=01で、入力キャプチャソースのみを選択します。
   - TIMx_CCERレジスタのCC1P=0とCC1NP=0で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。

tim_ti1の立上がりエッジでカウンタが有効になり、TIFフラグがセットされます。カウンタは、その後tim_etr_inの立上がりエッジでカウントします。

tim_etr_in信号の立上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_etrp入力の再同期回路によるものです。

図667. 外部クロックモード2＋トリガモードの制御回路



55.4.23 タイマの同期

タイマの同期や連携した動作のために、TIMxタイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

図668と図669に、マスタ／スレーブタイマー接続の例を示します。

図668. マスタ／スレーブタイマの例

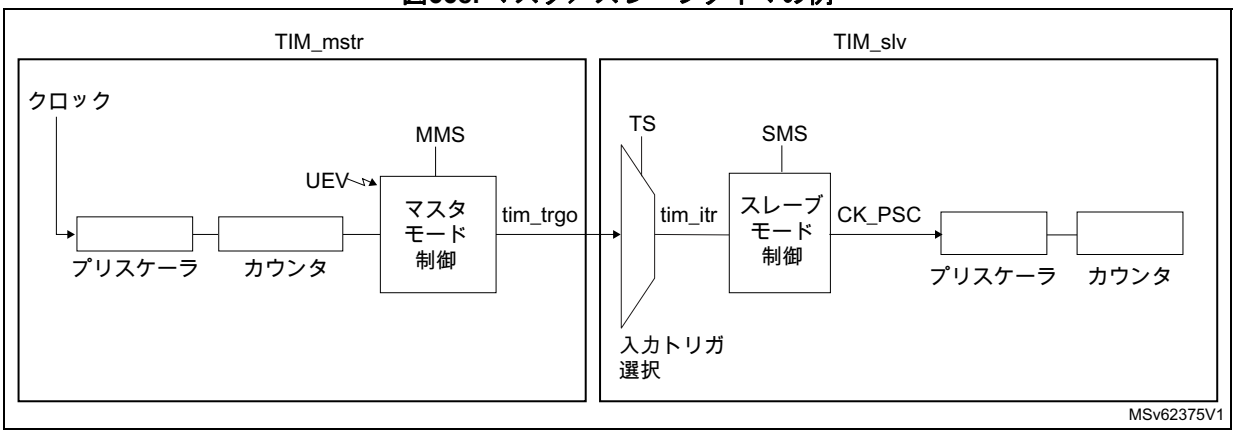
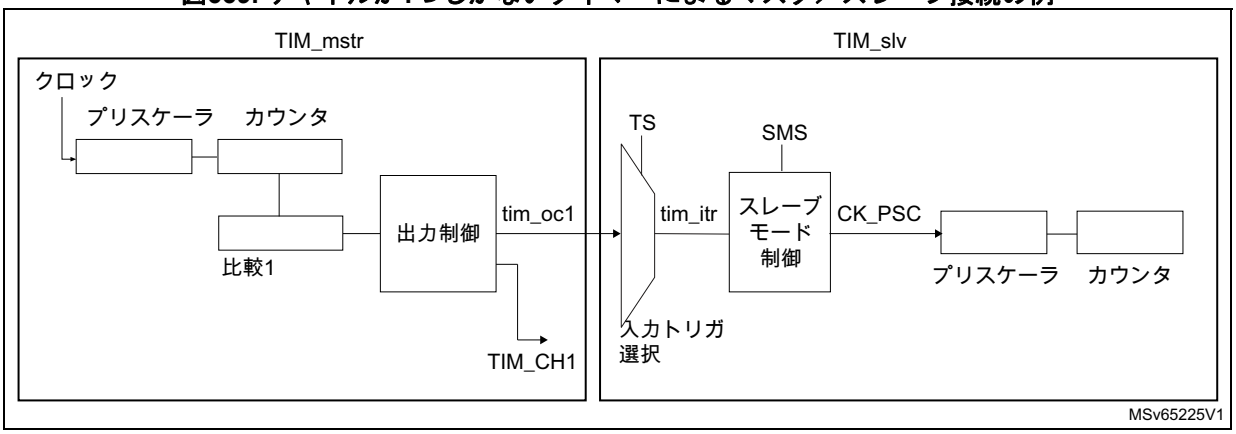


図669. チャネルが1つしかないタイマーによるマスタ／スレーブ接続の例



注： **チャネルが1つしかないタイマー（図669を参照）には、マスタモードはありません。ただし、tim_oc1出力信号をスレーブタイマのトリガとして使用できます（セクション 55.4.2 : TIM2/TIM3/TIM4/TIM5ピンおよび内部信号のTIMx内部トリガ接続の表を参照）。**
tim_oc1信号パルス幅は、スレーブタイマがトリガを検出するように、転送先のタイマの少なくとも2クロックサイクルになるようにプログラムする必要があります。
たとえば、転送先のタイマtim_ker_ckのクロックが、転送元のタイマより4倍遅い場合、OC1パルス幅は8クロックサイクルである必要があります。

タイマを別のタイマのプリスケーラとして使用する

たとえば、TIM_mstrがTIM_slvのプリスケーラとして動作するように設定できます。図668を参照してください。このためには、次の操作を行います。

1. 更新イベントUEVごとに定期的なトリガ信号を出力するように、TIM_mstrをマスタモードに設定します。TIM_mstr_CR2レジスタにMMS=010を書き込んだ場合、更新イベントが生成されるたびに、tim_trgoで立ち上がりエッジが出力されます。
2. TIM_mstrのtim_trgo出力をTIM_slvに接続するためには、TIM_slvを、ITR2を内部トリガとして使用するスレーブモードに設定する必要があります。これは、TIM_slv_SMCRレジスタのTSビットを通して選択されます（TS=00010を書き込みます）。
3. 次に、スレーブモードコントローラを外部クロックモード1に設定しなければなりません（TIM_slv_SMCRレジスタでSMS=111を書き込みます）。これによりTIM_slvは、TIM_mstrの定期的なトリガ信号の立ち上がりエッジ（TIM_mstrカウンタのオーバーフローに対応しています）をクロックとして動作します。
4. 最後に、それぞれのCENビット（TIMx_CR1レジスタ）をセットすることによって、両方のタイマを有効にする必要があります。

注： **TIM_mstr のトリガ出力として tim_ocx が選択された場合（MMS=1xx）、その立ち上がりエッジがTIM_slvカウンタのクロックとして使用されます。**

タイマを使用して別のタイマを有効にする

この例では、TIM_slvの有効化を、TIM_mstrの出力比較1で制御します。接続については、図668を参照してください。TIM_slvは、分周された内部クロックでカウントします（TIM_mstrのtim_oc1refがハイのときのみ）。両方のカウンタクロック周波数は、tim_ker_ckをプリスケーラで3分周したものです（$f_{tim_cnt_ck} = f_{tim_ker_ck}/3$）。

1. TIM_mstrをマスタモードに設定して、その出力比較1基準（tim_oc1ref）信号をトリガ出力として送信します（TIM_mstr_CR2レジスタのMMS=100）。
2. TIM_mstrのtim_oc1ref波形を設定します（TIM_mstr_CCMR1レジスタ）。

3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slvをゲートモードに設定します（TIM_slv_SMCRレジスタのSMS=101）。
5. CENビット（TIM_slv_CR1レジスタ）に“1”を書き込んで、TIM_slvを有効にします。
6. CENビット（TIM_mstr_CR1レジスタ）に“1”を書き込んで、TIM_mstrを開始します。

注： スレーブタイマのカウンタクロックはマスタタイマのカウンタクロックと同期しないので、このモードはTIM_slvカウンタのイネーブル信号にのみ影響を与えます。

図670. TIM_mstrのtim_oc1refによるTIM_slvのゲート操作

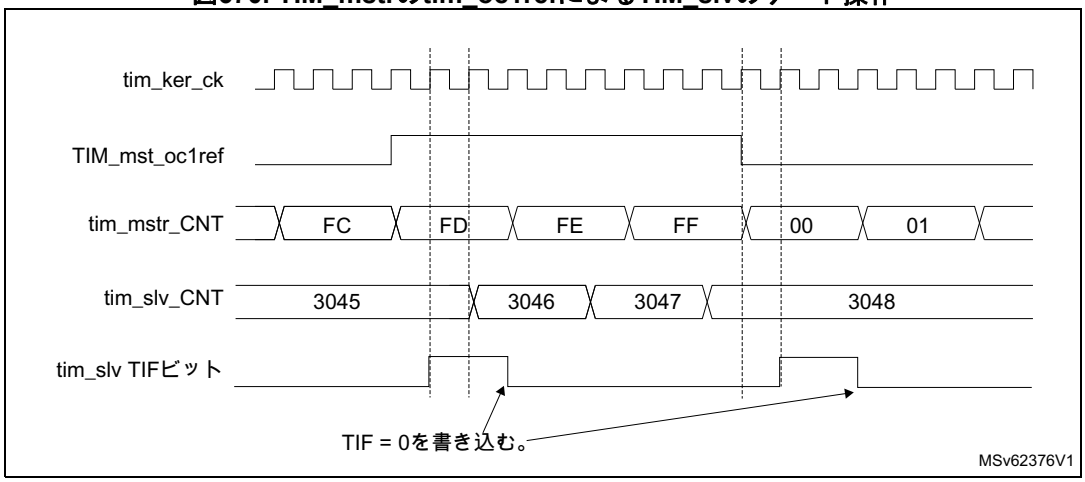
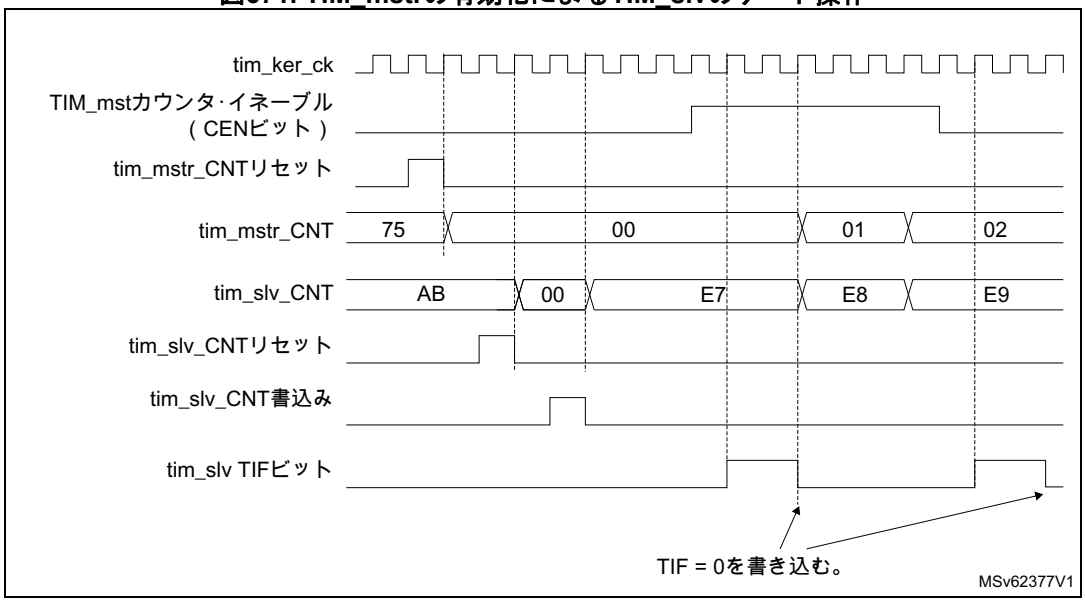


図670の例では、TIM_slvのカウンタとプリスケーラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM_mstrを開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。その後、任意の値をタイマカウンタに書き込むことができます。TIMx_EGRレジスタのUGビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では（図671を参照）、TIM_mstrとTIM_slvを同期させます。TIM_mstrはマスタであり、0からスタートします。TIM_slvはスレーブであり、0xE7から開始します。プリスケーラの分周比は両方のタイマで同じです。TIM_mstr_CR1レジスタのCENビットに0を書き込むことによってTIM_mstrを無効にすると、TIM_slvは停止します。

1. TIM_mstrをマスタモードに設定して、その出力比較1基準（tim_oc1ref）信号をトリガ出力として送信します（TIM_mstr_CR2レジスタのMMS=100）。
2. TIM_mstrのtim_oc1ref波形を設定します（TIM_mstr_CCMR1レジスタ）。
3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slvをゲートモードに設定します（TIM_slv_SMCRレジスタのSMS=101）。
5. UGビット（TIM_mstr_EGRレジスタ）に“1”を書き込むことによって、TIM_mstrをリセットします。
6. UGビット（TIM_slv_EGRレジスタ）に“1”を書き込むことによって、TIM_slvをリセットします。
7. TIM_slvカウンタ（TIM_slv_CNT）に“0xE7”を書き込んで、TIM_slvを0xE7に初期化します。
8. CENビット（TIM_slv_CR1レジスタ）に“1”を書き込んで、TIM_slvを有効にします。
9. CENビット（TIM_mstr_CR1レジスタ）に“1”を書き込んで、TIM_mstrを開始します。
10. CENビット（TIM_mstr_CR1レジスタ）に“0”を書き込んで、TIM_mstrを停止します。

図671. TIM_mstrの有効化によるTIM_slvのゲート操作

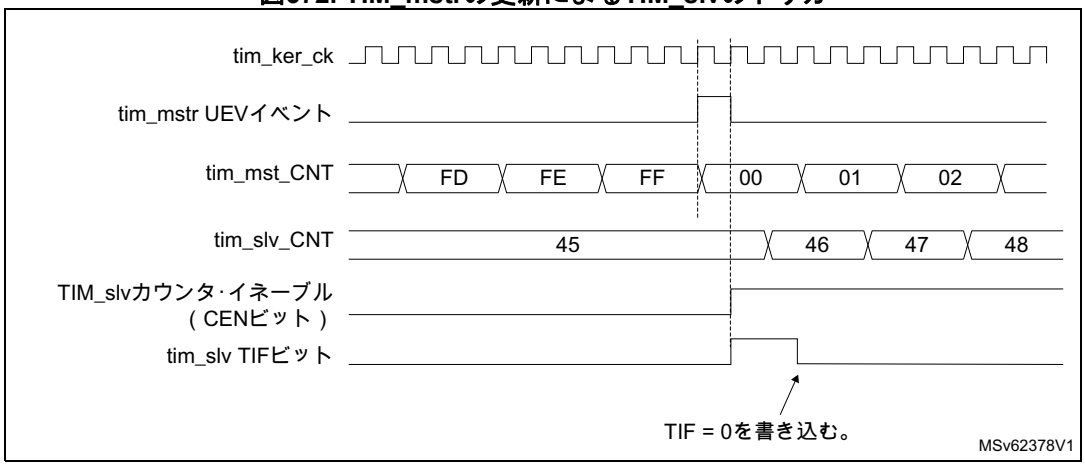


タイマを使用して別のタイマを開始する

この例では、TIM_mstrの更新イベントによってTIM_slvの有効化を設定します。接続については、図668を参照してください。TIM_mstrによって更新イベントが生成されると、TIM_slvは、分周された内部クロックで現在値（ゼロである必要はありません）からカウントを開始します。TIM_slvがトリガ信号を受信すると、そのCENビットが自動的にセットされ、カウンタはTIM_slv_CR1レジスタのCENビットに0が書き込まれるまでカウントします。両方のカウンタクロック周波数は、tim_ker_ckをプリスケーラで3分周したものです（$f_{tim_cnt_ck} = f_{tim_ker_ck}/3$）。

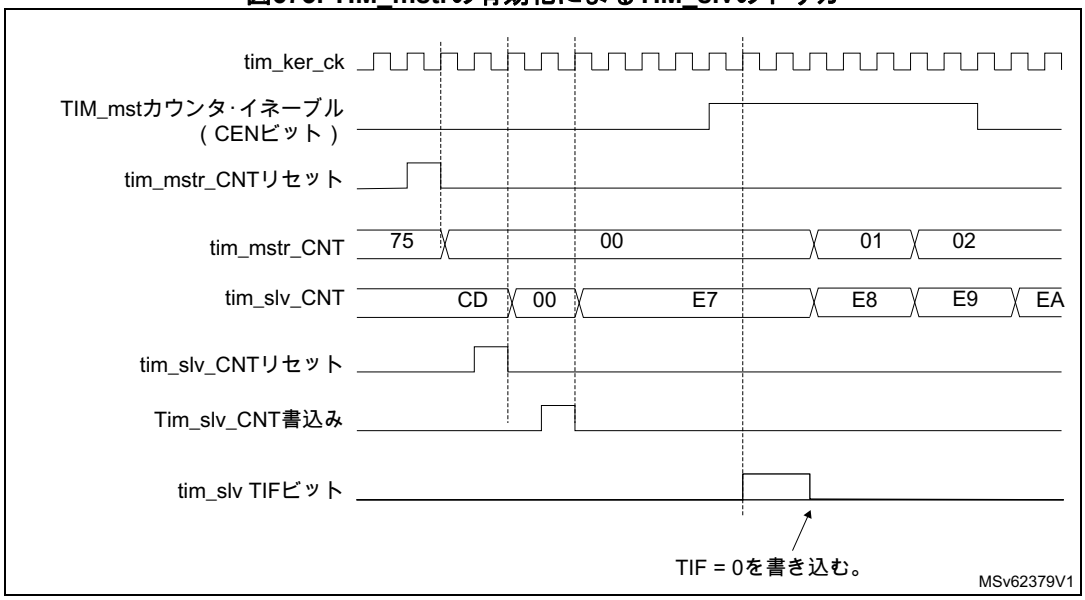
1. TIM_mstr をマスタモードに設定して、その更新イベント（UEV）をトリガ出力として送信します（TIM_mstr_CR2レジスタのMMS=010）。
2. TIM_mstrの周期を設定します（TIM_mstr_ARRレジスタ）。
3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slvをトリガモードに設定します（TIM_slv_SMCRレジスタのSMS=110）。
5. CENビット（TIM_mstr_CR1レジスタ）に“1”を書き込んで、TIM_mstrを開始します。

図672. TIM_mstrの更新によるTIM_slvのトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化できます。図673は図672と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します（TIM_slv_SMCRレジスタのSMS=110）。

図673. TIM_mstrの有効化によるTIM_slvのトリガ



外部トリガに対応して2つのタイマを同期して開始します。

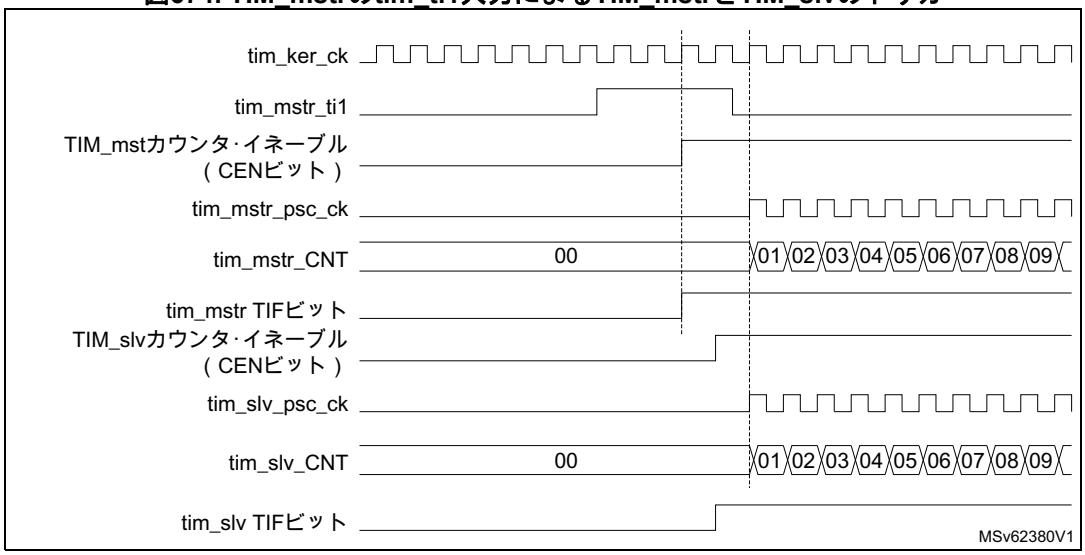
この例では、tim_ti1入力の立ち上がりエッジでTIM_mstrを有効にし、TIM_mstrが有効になると、TIM_slvを有効にします。接続については、図668を参照してください。カウンタの同時性を確保するため、TIM_mstrはマスタ／スレーブモードに設定する必要があります（tim_ti1に対してはスレーブ、TIM_slvに対してはマスタ）。

1. TIM_mstrをマスタモードに設定して、その有効化をトリガ出力として送信します（TIM_mstr_CR2レジスタのMMS=001）。
2. TIM_mstrをスレーブモードに設定して、tim_ti1から入力トリガを受け取るようにします（TIM_mstr_SMCRレジスタのTS=00100）。
3. TIM_mstrをトリガモードに設定します（TIM_mstr_SMCRレジスタのSMS=110）。
4. MSM=1（TIM_mstr_SMCR レジスタ）を書き込むことによって、TIM_mstr をマスタ / スレーブモードに設定します。
5. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタのTS=00000）。
6. TIM_slvをトリガモードに設定します（TIM_slv_SMCRレジスタのSMS=110）。

tim_ti1（TIM_mstr）で立ち上がりエッジが発生すると、両方のカウンタが同期して内部クロックによるカウントを開始し、両方のTIFフラグがセットされます。

注： **この例では、両方のタイマが開始前に初期化されます（それぞれのUGビットをセットすることによって）。両方のカウンタは0からスタートしますが、カウンタレジスタ（TIMx_CNT）のいずれかに書き込むことによって、両者間に容易にオフセットを挿入できます。マスタ／スレーブモードでは、TIM_mstrのCNT_ENとCK_PSCの間に遅延が挿入されます。**

図674. TIM_mstrのtim_ti1入力によるTIM_mstrとTIM_slvのトリガ



注： **tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

55.4.24 ADCトリガ

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。

注： **tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

55.4.25 DMAバーストモード

TIMxタイマには、1つのイベントで多重DMAリクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMAコントローラの転送先は一意で、仮想レジスタTIMx_DMARを示している必要があります。特定のタイマイベントで、タイマは一連のDMAリクエスト（バースト）を開始します。TIMx_DMARレジスタへの各書込みは、実際にタイマレジスタの1つにリダイレクトされます。

TIMx_DCRレジスタのDBL[4:0] ビットによって、DMAバースト長がセットされます。タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCRレジスタのDBA[4:0] ビットは、DMA転送のDMAベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し／書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

TIMx_DCRレジスタのDBSS[3:0] ビットでは、DMAバースト転送をトリガする割込みソースを定義します（詳細はセクション 55.5.23：TIMx DMA制御レジスタ（TIMx_DCR）（x = 2～5）を参照）。

たとえば、更新イベント時にCCRxレジスタ値の内容を更新するためにタイマDMAバースト機能を使用します（x = 2、3、4）。このとき、DMAはCCRxレジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応するDMAチャネルを次のように設定します。
   - DMAチャネルペリフェラルアドレスを、DMARレジスタアドレスとします。
   - DMAチャネルメモリアドレスを、DMAによってCCRxレジスタに転送されるデータを格納するRAM内のバッファアドレスとします。
   - 転送データ数 = 3とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBAとDBLのビットフィールドを次のように設定することによって、DCRレジスタを設定します。DBL = 3回転送、DBA = 0xE、およびDBSS = 1。
3. TIMx更新DMAリクエストを有効にします（DIERレジスタのUDEビットをセット）。
4. TIMxを有効化
5. DMAチャネルを有効化注：

この例は、各CCRxレジスタが1回更新される場合です。たとえば、各CCRxレジスタが2回更新される場合は、転送データ数は6である必要があります。データ1、データ2、データ3、データ4、データ5、データ6を格納するRAMのバッファを例にします。データは、CCRxレジスタに次のように転送されます。最初の更新DMAリクエストでデータ1がCCR2に転送され、データ2はCCR3に、データ3はCCR4にそれぞれ転送され、2番目の更新DMAリクエストでデータ4がCCR2に、データ5がCCR3に、データ6がCCR4にそれぞれ転送されます。

注： **null値を予約済みレジスタに書き込むことができます。**

55.4.26 TIM2/TIM3/TIM4/TIM5DMAリクエスト

表565に示すように、TIM2/TIM3/TIM4/TIM5 はDMAリクエストを生成できます。

表565.DMAリクエスト

DMAリクエスト信号	DMAリクエストあり	有効制御ビット
tim_upd_dma	更新	UDE
tim_cc1_dma	キャプチャ／比較1	CC1DE
tim_cc2_dma	キャプチャ／比較2	CC2DE
tim_cc3_dma	キャプチャ／比較3	CC3DE
tim_cc4_dma	キャプチャ／比較4	CC4DE
tim_trg_dma	トリガ	TDE

注： **一部のタイマの DMA リクエストを、DMA コントローラに接続してはなりません。詳細については、DMAのセクションを参照してください。**

55.4.27 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M33 コアは停止状態）、TIMxカウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

55.4.28 TIM2/TIM3/TIM4/TIM5 低消費電力モード

表566.低消費電力モードがTIM2/TIM3/TIM4/TIM5　に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。 割込みによって、デバイスはSLEEPモードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

55.4.29 TIM2/TIM3/TIM4/TIM5 割込み

表567に示すように、TIM2/TIM3/TIM4/TIM5 は複数の割込みを生成できます。

表567.割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
TIM_UP	更新	UIF	UIE	UIFに0を書き込む	あり	不可
TIM_CC	キャプチャ／比較1	CC1IF	CC1IE	CC1IFに0を書き込む	あり	不可
	キャプチャ／比較2	CC2IF	CC2IE	CC2IFに0を書き込む	あり	不可
	キャプチャ／比較3	CC3IF	CC3IE	CC3IFに0を書き込む	あり	不可
	キャプチャ／比較4	CC4IF	CC4IE	CC4IFに0を書き込む	あり	不可
TIM_TRG	トリガ	TIF	TIE	TIFに0を書き込む	あり	不可
TIM_DIR_IDX	インデックス	IDXF	IDXIE	IDXFに0を書き込む	あり	不可
	方向	DIRF	DIRIE	DIRFに0を書き込む	あり	不可
TIM_IERR	インデックスエラー	IERRF	IERRIE	IERRFに0を書き込む	あり	不可
TIM_TER	遷移エラー	TERRF	TERRIE	TERRFに0を書き込む	あり	不可

55.5 TIM2/TIM3/TIM4/TIM5 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション1.2を参照してください。

ペリフェラルレジスタには、ハーフワード（16bit）またはワード（32bit）単位でアクセスする必要があります。

55.5.1 TIMx制御レジスタ1（TIMx_CR1）（x = 2〜5）

アドレスオフセット：0x000

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DITHEN**：ディザリングイネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHENビットは、CENビットがリセットされているときのみ変更できます。

ビット11 **UIFREMAP**：UIFステータスビットの再配置

0：再配置なし。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされません。

1：再配置は有効です。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デジタルフィルタ（tim_etr_in、tim_tix）によって使用されるサンプリングクロックとの間の分周比を示します。

00：$t_{DTS} = t_{tim_ker_ck}$

01：$t_{DTS} = 2 \times t_{tim_ker_ck}$

10：$t_{DTS} = 4 \times t_{tim_ker_ck}$

11：予約済み

ビット7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARRレジスタはバッファされません。

1：TIMx_ARRレジスタはバッファされます。

ビット6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモード。カウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。

01：センターアラインモード1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。

10：センターアラインモード2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。

11：センターアラインモード3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRxレジスタのCCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット3 **OPM**：ワンパルスモード。

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CENビットをクリア）。

ビット2 **URS**：更新リクエストソース

このビットは、UEVイベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたはDMAリクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたはDMAリクエストを生成します（有効な場合）。

ビット1 **UDIS**：更新ディセーブル

このビットは、UEVイベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEVは有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEVは無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UGビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CENビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的にCENビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CENは自動的にクリアされます。

55.5.2 TIMx制御レジスタ2（TIMx_CR2）（x = 2～5）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット24:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TI1S**：tim_ti1選択

0：tim_ti1_in[15:0] マルチプレクサ出力がtim_ti1入力に接続されます。
1：tim_ti1_in[15:0]、tim_ti2_in[15:0]、およびtim_ti3_in[15:0] マルチプレクサ出力がXOR演算されてtim_ti1入力に接続されます。セクション 54.3.29：2154ページのホールセンサとのインタフェースも参照してください。

ビット25、6、5、4 **MMS[3:0]**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（tim_trgo）を選択するために使用します。組み合わせは、次のとおりです。

0000：**リセット** - TIMx_EGRレジスタのUGビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタ·イネーブル信号CNT_ENがトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN制御ビットとトリガ入力との論理和（AND）によって生成されます。
カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、tim_trgoには遅延が存在します（TIMx_SMCRレジスタのMSMビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - CC1IFフラグがすでにハイであっても、それがセットされるキャプチャまたは比較一致が発生すると、トリガ出力として正のパルスがただちに送信されます（tim_trgo）。

0100：**比較** - tim_oc1refc信号がトリガ出力（tim_trgo）として使用されます。

0101：**比較** - tim_oc1ref信号がトリガ出力（tim_trgo）として使用されます。

0110：**比較** - tim_oc1ref信号がトリガ出力（tim_trgo）として使用されます。

0111：**比較** - tim_oc1ref信号がトリガ出力（tim_trgo）として使用されます。

1000：**エンコーダ·クロック出力** - エンコーダ·クロック信号がトリガ出力（tim_trgo）として使用されます。このコードはSMS[3:0] 値、0001、0010、0011、1010、1011、1100、1101、1110、1111に対して有効です。その他のSMS[3:0] コードは許されず、予想外の動作につながることがあります。

その他：予約済み

注： スレーブタイマまたはADCのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット3 **CCDS**：キャプチャ／比較DMA選択

0：CCx DMAリクエストは、CCxイベントが発生すると送信されます。
1：CCx DMAリクエストは、更新イベントが発生すると送信されます。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

55.5.3 TIMxスレーブモード制御レジスタ（TIMx_SMCR）（x = 2～5）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
						rw	rw			rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **SMSPS**：SMSプリロードソース

このビットは、プリロードからアクティブへのSMS[3:0] ビットフィールドの転送をトリガするイベントのいずれかを選択します。

0：転送は、タイマの更新イベントによってトリガされます。

1：転送は、インデックス･イベントによってトリガされます。

ビット24 **SMSPE**：SMSプリロードイネーブル

このビットは、SMS[3:0] ビットフィールドをプリロードするかどうかを選択します。

0：SMS[3:0] ビットフィールドはプリロードされません。

1：SMS[3:0] プリロードは有効です。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **ETP**：外部トリガ極性

このビットは、tim_etr_inと $\overline{\text{tim_etr_in}}$ のいずれがトリガ動作に使用されるかを選択します。

0：tim_etr_inは反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：tim_etr_inは反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード2を有効にします。

0：外部クロックモード2は無効です。

1：外部クロックモード2は有効です。カウンタは、tim_etrf信号のアクティブエッジによってクロック供給されます。

注： ECEビットをセットすることは、tim_trgiがtim_etrfに接続された状態で外部クロックモード1を選択することと同じ効果があります（SMS=111、TS=00111）。
外部クロックモード2と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、tim_trgiをtim_etrfに接続することはできません（TSビットが00111でないことが必要）。
外部クロックモード1と外部クロックモード2が同時に有効な場合、外部クロック入力はtim_etrfです。

ビット13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号tim_etrpの周波数は、tim_ker_ck周波数の1/4までに制限されます。プリスケーラを有効にすると、tim_etrp周波数を低減できます。これは、高速な外部クロックをtim_etr_inに入力するときに役立ちます。

00：プリスケーラオフ

01：tim_etrp周波数は2分周されます。

10：tim_etrp周波数は4分周されます。

11：tim_etrp周波数は8分周されます。

ビット11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、tim_etrp信号をサンプルする周波数と、tim_etrpに適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS}で行われます。
0001： $f_{SAMPLING}=f_{tim_ker_ck}$, N=2
0010： $f_{SAMPLING}=f_{tim_ker_ck}$, N=4
0011： $f_{SAMPLING}=f_{tim_ker_ck}$, N=8
0100： $f_{SAMPLING}=f_{DTS}/2$、N = 6
0101： $f_{SAMPLING}=f_{DTS}/2$、N = 8
0110： $f_{SAMPLING}=f_{DTS}/4$, N=6
0111： $f_{SAMPLING}=f_{DTS}/4$, N=8
1000： $f_{SAMPLING}=f_{DTS}/8$, N=6
1001： $f_{SAMPLING}=f_{DTS}/8$, N=8
1010： $f_{SAMPLING}=f_{DTS}/16$, N=5
1011： $f_{SAMPLING}=f_{DTS}/16$, N=6
1100： $f_{SAMPLING}=f_{DTS}/16$, N=8
1101： $f_{SAMPLING}=f_{DTS}/32$, N=5
1110： $f_{SAMPLING}=f_{DTS}/32$, N=6
1111： $f_{SAMPLING}=f_{DTS}/32$, N=8

ビット7 **MSM**：マスタ/スレーブモード

0：影響なし。
1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgoを通じて）完全な同期状態が得られるまで遅延されます。これは、1つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ0（tim_itr0）
00001：内部トリガ1（tim_itr1）
00010：内部トリガ2（tim_itr2）
00011：内部トリガ3（tim_itr3）
00100：tim_ti1エッジ検出回路（tim_ti1f_ed）
00101：フィルタタイマ入力1（tim_ti1fp1）
00110：フィルタタイマ入力2（tim_ti2fp2）
00111：外部トリガ入力（tim_etrf）
01000：内部トリガ4（tim_itr4）
01001：内部トリガ5（tim_itr5）
01010：内部トリガ6（tim_itr6）
01011：内部トリガ7（tim_itr7）
01100：内部トリガ8（tim_itr8）
01101：内部トリガ9（tim_itr9）
01110：内部トリガ10（tim_itr10）
01111：内部トリガ11（tim_itr11）
10000：内部トリガ12（tim_itr12）
10001：内部トリガ13（tim_itr13）
10010：内部トリガ14（tim_itr14）
10011：内部トリガ15（tim_itr15）
その他：予約済み

製品固有の実装の詳細はセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS = 000のときなど）にのみ変更しなければなりません。

ビット3 **OCCS**：OCREFクリア選択

このビットは、OCREFクリアソースを選択するために使用されます。

0：tim_ocref_clr_intはtim_ocref_clr入力に接続されます。

1：tim_ocref_clr_intはtim_etrfに接続されます。

注： OCREFクリア選択機能がサポートされていない場合、このビットは予約済みであり、ハードウェアによって強制的に "0" に設定されます。セクション 55.3：TIM2/TIM3/TIM4/TIM5 の実装を参照してください。

ビット16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード1 - カウンタは、tim_ti2fp2のレベルに応じて、tim_ti1fp1のエッジでカウントアップ／ダウンします。

0010：直交エンコーダ･モード2、x2モード - カウンタは、tim_ti1fp1のレベルに応じて、tim_ti2fp2のエッジでカウント･アップ／ダウンします。0100：

0011：エンコーダモード3 - カウンタは、他の入力のレベルに応じて、tim_ti1fp1とtim_ti2fp2の両方のエッジでカウントアップ/ダウンします。

0100：リセット･モード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲート･モード - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ･モード - カウンタは、トリガtim_trgiの立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロック･モード1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセット･モードとトリガ･モードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲート･モードとリセット･モードの組み合わせ - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

1010：エンコーダモード：クロックプラス方向、x2モード。

1011：クロックプラス方向、x1モード。tim_ti2fp2エッジ検出はCC2Pによって設定されます。

1100：エンコーダモード：方向クロック、x2モード。

1101：方向クロック、x1モード。tim_ti1fp1とtim_ti2fp2エッジ検出はCC1PとCC2Pによって設定されます。

1110：直交エンコーダ･モード：x1モード、tim_ti1fp1エッジのみでカウント、エッジ検出はCC1Pによって設定されます。

1111：直交エンコーダ･モード：x1モード、tim_ti1fp2エッジのみでカウント、エッジ検出はCC2Pによって設定されます。

注： トリガ入力としてtim_ti1f_edが選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_edはtim_ti1fの変化ごとに1パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

55.5.4 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 2～5）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.
								rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TERRIE**：転送エラー割込みイネーブル
0：遷移エラー割込みは無効です。
1：遷移エラー割込みは有効です。

ビット22 **IERRIE**：インデックスエラー割込みイネーブル
0：インデックスエラー割込みは無効です。
1：インデックスエラー割込みは有効です。

ビット21 **DIRIE**：方向変更割込みイネーブル
0：方向変更割込みは無効です。
1：方向変更割込みは有効です。

ビット20 **IDXIE**：インデックス割込みイネーブル
0：インデックス割込みは無効です。
1：インデックス割込みは有効です。

ビット19:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TDE**：トリガDMAリクエストイネーブル
0：トリガDMAリクエストは無効です。
1：トリガDMAリクエストは有効です。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CC4DE**：キャプチャ／比較4 DMAリクエストイネーブル
0：CC4 DMAリクエストは無効です。
1：CC4 DMAリクエストは有効です。

ビット11 **CC3DE**：キャプチャ／比較3 DMAリクエストイネーブル
0：CC3 DMAリクエストは無効です。
1：CC3 DMAリクエストは有効です。

ビット10 **CC2DE**：キャプチャ／比較2 DMAリクエストイネーブル
0：CC2 DMAリクエストは無効です。
1：CC2 DMAリクエストは有効です。

ビット9 **CC1DE**：キャプチャ／比較1 DMAリクエストイネーブル
0：CC1 DMAリクエストは無効です。
1：CC1 DMAリクエストは有効です。

ビット8 **UDE**：更新DMAリクエストイネーブル
0：更新DMAリクエストは無効です。
1：更新DMAリクエストは有効です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TIE**：トリガ割込みイネーブル

0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CC4IE**：キャプチャ／比較4割込みイネーブル

0：CC4割込みは無効です。
1：CC4割込みは有効です。

ビット3 **CC3IE**：キャプチャ／比較3割込みイネーブル

0：CC3割込みは無効です。
1：CC3割込みは有効です。

ビット2 **CC2IE**：キャプチャ／比較2割込みイネーブル

0：CC2割込みは無効です。
1：CC2割込みは有効です。

ビット1 **CC1IE**：キャプチャ／比較1割込みイネーブル

0：CC1割込みは無効です。
1：CC1割込みは有効です。

ビット0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。
1：更新割込みは有効です。

55.5.5 TIMxステータスレジスタ（TIMx_SR）（x = 2〜5）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	Res.	Res.
								rc_w0	rc_w0	rc_w0	rc_w0				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TERRF**：転送エラー割込みフラグ

このフラグは、エンコーダモードで遷移エラーが検出されたときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：エンコーダ遷移エラーは検出されていません。
1：エンコーダ遷移エラーが検出されています。

ビット22 **IERRF**：インデックスエラー割込みフラグ

このフラグは、インデックスエラーが検出されたときに、ハードウェアによってセットされます。ソフトウェアによって
“0”を書き込むことによってクリアされます。
0：インデックスエラーは検出されていません。
1：インデックスエラーが検出されています。

ビット21 **DIRF**：方向変更割込みフラグ

このフラグは、エンコーダモードで方向が変わった（TIMx_CRのDIRビット値が変わった）ときに、ハードウェアによってセットされます。

これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：方向の変更なし

1：方向の変更

ビット20 **IDXF**：インデックス割込みフラグ

このフラグは、インデックスイベントが検出されたときに、ハードウェアによってセットされます。ソフトウェアによって

“0”を書き込むことによってクリアされます。

0：インデックスイベントは発生していません。

1：インデックスイベントが発生しています。

ビット19:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **CC4OF**：キャプチャ／比較4オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット11 **CC3OF**：キャプチャ／比較3オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット10 **CC2OF**：キャプチャ／比較2オーバーキャプチャフラグ

CC1OFの説明を参照してください。

ビット9 **CC1OF**：キャプチャ／比較1オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IFフラグがすでにセットされているときに、カウンタの値がTIMx_CCR1レジスタにキャプチャされました。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TIF**：トリガ割込みフラグ

このフラグは、TRGトリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CC4IF**：キャプチャ／比較4割込みフラグ

CC1IFの説明を参照してください。

ビット3 **CC3IF**：キャプチャ／比較3割込みフラグ

CC1IFの説明を参照してください。

ビット2 **CC2IF**：キャプチャ／比較2割込みフラグ

CC1IFの説明を参照してください。

ビット1 **CC1IF**：キャプチャ／比較1割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、またはTIMx_CCR1レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネルCC1が出力として設定されている場合：カウンタTIMx_CNTの内容がTIMx_CCR1レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1の内容がTIMx_ARRの内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）にCC1IFビットはハイになります。センターアラインモードでのフラグの設定には3つのオプションがあります。詳細については、TIMx_CR1レジスタのCMSビットを参照してください。

CC1チャネルが入力として設定されている場合：このビットは、TIMx_CCR1レジスタでカウンタ値がキャプチャされた（TIMx_CCERのCC1PおよびCC1NPビット設定で定義されたエッジ検知に従って、IC1でエッジが検出された）ときに設定されます。

ビット0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

オーバーフローまたはアンダーフロー時と、TIMx_CR1レジスタでUDIS=0の場合。

TIMx_CR1レジスタのURS=0かつUDIS=0であり、TIMx_EGRレジスタのUGビットを使用して、CNTがソフトウェアによって再初期化されたとき。

TIMx_CR1レジスタのURS=0かつUDIS=0であり、トリガイベントによってCNTが再初期化されたとき（同期制御レジスタの説明を参照）。

55.5.6 TIMxイベント生成レジスタ（TIMx_EGR）（x = 2〜5）

アドレスオフセット：0x014

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SRレジスタのTIFフラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CC4G**：キャプチャ /比較4生成

CC1Gの説明を参照してください。

ビット3 **CC3G**：キャプチャ /比較3生成

CC1Gの説明を参照してください。

ビット2 **CC2G**：キャプチャ /比較2生成

CC1Gの説明を参照してください。

ビット1 **CC1G**：キャプチャ／比較1生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル1でキャプチャ／比較イベントが生成されます。

CC1チャネルが出力として設定されている場合：

CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。

CC1チャネルが入力として設定されている場合：

カウンタの現在値がTIMx_CCR1レジスタにキャプチャされます。CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。CC1IFフラグがすでにハイの場合、CC1OFフラグがセットされます。

ビット0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

55.5.7 TIMxキャプチャ／比較モードレジスタ1（TIMx_CCMR1）（x = 2〜5）

アドレスオフセット：0x018

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **IC2F[3:0]**：入力キャプチャ 2フィルタ

ビット11:10 **IC2PSC[1:0]**：入力キャプチャ 2プリスケーラ

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2に配置されます。

10：CC2チャネルは入力として設定され、tim_ic1. はtim_ti2に配置されます。

11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC2E=0）のときにのみ書込み可能です。

ビット7:4 **IC1F[3:0]**：入力キャプチャ 1フィルタ

このビットフィールドは、tim_ti1入力をサンプリングする周波数と、tim_ti1に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS} で行われます。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット3:2 **IC1PSC[1:0]**：入力キャプチャ 1プリスケーラ

このビットフィールドは、CC1入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E=0（TIMx_CCERレジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2イベントごとに行われます。

10：キャプチャは、4イベントごとに行われます。

11：キャプチャは、8イベントごとに行われます。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。

10：CC1チャネルは入力として設定され、tim_ic1はtim_ti2にマッピングされます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

55.5.8 TIMxキャプチャ／比較モードレジスタ1 [オルタネート]（TIMx_CCMR1）（x = 2〜5）

アドレスオフセット：0x018

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC2CE**：出力比較2クリアイネーブル

ビット24、14:12 **OC2M[3:0]**：出力比較2モード
ビット6:4のOC1M説明を参照

ビット11 **OC2PE**出力比較2プリロードイネーブル

ビット10 **OC2FE**：出力比較2高速イネーブル

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2チャネルは出力として設定されます。
01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2にマッピングされます。
10：CC2チャネルは入力として設定され、tim_ic1はtim_ti2に配置されます。
11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。
注： CC2Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC2E=0）のときにのみ書込み可能です。

ビット7 **OC1CE**：出力比較1クリアイネーブル
0：tim_oc1refはtim_ocref_clr_int入力の影響を受けません。
1：tim_oc1refはtim_ocref_clr_int入力のハイレベルが検出されるとクリアされます。

ビット16、6:4 **OC1M[3:0]**：出力比較1モード

これらのビットは、tim_oc1が導き出される出力基準信号tim_oc1refの動作を定義します。tim_oc1refはアクティブハイですが、tim_oc1のアクティブレベルはCC1Pビットに依存します。

0000：停止 - 出力比較レジスタTIMx_CCR1とカウンタTIMx_CNTとの間の比較結果は出力に影響しません。このモードは、タイマをソフトウェアタイムベースとして使用する場合に使用できます。タイマ動作中に停止モードが有効になると、出力は停止状態になる前の状態（アクティブまたはインアクティブ）を維持します。

0001：一致時にチャネル1をアクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル1を非アクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的に低になります。

0011：反転 - TIMx_CNT = TIMx_CCR1のとき、tim_oc1refは反転します。

0100：強制非アクティブレベル - tim_oc1refは強制的にローになります。

0101：強制アクティブレベル - tim_oc1refは強制的にハイになります。

0110：PWMモード1 - カウントアップ時、チャネル1は、TIMx_CNT < TIMx_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル1は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（tim_oc1ref=0）に、そうでない場合はアクティブ（tim_oc1ref=1）になります。

0111：PWMモード2 - カウントアップ時、チャネル1は、TIMx_CNT < TIMx_CCR1の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル1は、TIMx_CNT> TIMx_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能なOPMモード1 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能なOPMモード2 - アップカウント･モードでは、tim_trgi信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWMモード2と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み

1011：予約済み

1100：組み合わせPWMモード1 - tim_oc1refは、PWMモード1と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ORです。

1101：組み合わせPWMモード2 - tim_oc1refは、PWMモード2と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ANDです。

1110：非対称PWMモード1 - tim_oc1refは、PWMモード1と同様に挙動します。tim_oc1refcは、カウンタがカウントアップするときにtim_oc1refを出力し、カウントダウンするときにtim_oc2refを出力します。

1111：非対称PWMモード2 - tim_oc1refは、PWMモード2と同様に挙動します。tim_oc1refcは、カウンタがカウントアップするときにtim_oc1refを出力し、カウントダウンするときにtim_oc2refを出力します。

注： PWMモードでは、比較結果が変化したとき、出力比較モードが「停止」モードから「PWM」モードに切り替わったとき、および出力比較モードが「強制アクティブ／インアクティブ」モードから「PWM」モードに切り替わったときに、OCREFのレベルが変化します。

ビット3 **OC1PE**：出力比較1プリロードイネーブル

0：TIMx_CCR1のプリロードレジスタは無効です。TIMx_CCR1は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

ビット2 **OC1FE**出力比較1高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1パルスモード（TIMx_CR1レジスタのOPMビットのセット）で使用する必要があります。

0：CC1の動作は、トリガがオンのときでも、通常、カウンタとCCR1の値に依存します。トリガ入力のエッジ発生からCC1出力が有効になるまでの最小遅延は、5クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1出力に対して、比較一致のように働きます。このような場合、OCは、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1出力を有効にするまでの遅延は、3クロックサイクルに短縮されます。OCFEは、チャネルがPWM1またはPWM2モードに設定されている場合のみ機能します。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1に配置されます。

10：CC1チャネルは入力として設定され、tim_ic2. はtim_ti1に配置されます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_ti1に配置されます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

55.5.9 TIMxキャプチャ／比較モードレジスタ2（TIMx_CCMR2）（x = 2〜5）

アドレスオフセット：0x01C

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **IC4F[3:0]**：入力キャプチャ 4フィルタ

ビット11:10 **IC4PSC[1:0]**：入力キャプチャ 4プリスケーラ

ビット9:8 **CC4S[1:0]**：キャプチャ／比較4選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4チャネルは出力として設定されます。

01：CC4チャネルは入力として設定され、tim_ic4はtim_ti4にマッピングされます。

10：CC4チャネルは入力として設定され、tim_ic4はtim_ti3にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic4はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC4E=0）のときにのみ書込み可能です。

ビット7:4 **IC3F[3:0]**：入力キャプチャ 3フィルタ

ビット3:2 **IC3PSC[1:0]**：入力キャプチャ 3プリスケーラ

ビット1:0 **CC3S[1:0]**：キャプチャ／比較3選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3チャネルは出力として設定されます。

01：CC3チャネルは入力として設定され、tim_ic3はtim_ti3にマッピングされます。

10：CC3チャネルは入力として設定され、tim_ic3はtim_ti4にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic3はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC3E=0）のときにのみ書込み可能です。

55.5.10 TIMxキャプチャ／比較モードレジスタ2 [オルタネート]（TIMx_CCMR2）（x = 2〜5）

アドレスオフセット：0x01C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC4CE**：出力比較4クリアイネーブル

ビット24、14:12 **OC4M[3:0]**：出力比較4モード

OC1Mの説明（TIMx_CCMR1レジスタのビット6:4）を参照

ビット11 **OC4PE**：出力比較4プリロードイネーブル

ビット10 **OC4FE**：出力比較4高速イネーブル

ビット9:8 **CC4S[1:0]**：キャプチャ／比較4選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4チャネルは出力として設定されます。

01：CC4チャネルは入力として設定され、tim_ic4はtim_ti4にマッピングされます。

10：CC4チャネルは入力として設定され、tim_ic4はtim_ti3にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic4はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC4E=0）のときにのみ書込み可能です。

ビット7 **OC3CE**：出力比較3クリアイネーブル

ビット16、6:4 **OC3M[3:0]**：出力比較3モード
OC1Mの説明（TIMx_CCMR1レジスタのビット6:4）を参照してください。

ビット3 **OC3PE**：出力比較3プリロードイネーブル

ビット2 **OC3FE**：出力比較3高速イネーブル

ビット1:0 **CC3S[1:0]**：キャプチャ／比較3選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3チャネルは出力として設定されます。
01：CC3チャネルは入力として設定され、tim_ic3はtim_ti3にマッピングされます。
10：CC3チャネルは入力として設定され、tim_ic3はtim_ti4にマッピングされます。
11：CC2チャネルは入力として設定され、tim_ic3はtim_trcにマッピングされます。このモードは、TSビット（TIMx_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC3E=0）のときにのみ書込み可能です。

55.5.11 TIMxキャプチャ／比較有効レジスタ（TIMx_CCER）（x = 2〜5）

アドレスオフセット：0x020

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット15 **CC4NP**：キャプチャ /比較4出力極性
CC1NPの説明を参照してください。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CC4P**：キャプチャ /比較4出力極性
CC1Pの説明を参照してください。

ビット12 **CC4E**：キャプチャ／比較4出力イネーブル
CC1Eの説明を参照してください。

ビット11 **CC3NP**：キャプチャ /比較3出力極性
CC1NPの説明を参照してください。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC3P**：キャプチャ /比較3出力極性
CC1Pの説明を参照してください。

ビット8 **CC3E**：キャプチャ／比較3出力イネーブル
CC1Eの説明を参照してください。

ビット7 **CC2NP**：キャプチャ／比較2出力極性
CC1NPの説明を参照してください。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CC2P**：キャプチャ／比較2出力極性
CC1Pの説明を参照してください。

ビット4 **CC2E**：キャプチャ／比較2出力イネーブル
CC1Eの説明を参照してください。

ビット3 **CC1NP**：キャプチャ /比較1出力極性

CC1チャネルが出力として設定されている場合：この場合、CC1NPはクリアされたままでなければなりません。

CC1チャネルが入力として設定されている場合：このビットは、tim_ti1fp1/tim_ti2fp1の極性を定義するためにCC1Pと組み合わせて使用されます。CC1Pの説明を参照してください。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CC1P**：キャプチャ /比較1出力極性

0：OC1はアクティブハイ（出力モード）／エッジ検知を選択（入力モード、下記を参照）

1：OC1はアクティブロー（出力モード）／エッジ検知を選択（入力モード、下記を参照）

CC1チャネルが入力として設定されているときは、CC1NP/CC1Pの両ビットは、トリガまたはキャプチャ動作に対するTI1FP1とTI2FP1のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立上がりエッジ。この回路はTIxFP1の立上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立下がりエッジ。この回路はTIxFP1の立下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転/両エッジ。この回路はTIxFP1の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

ビット0 **CC1E**：キャプチャ／比較1出力イネーブル

0：キャプチャモードは無効／OC1は非アクティブ。

1：キャプチャモードは有効／OC1信号は、対応する出力ピンに出力されます。

表568.標準tim_ocxチャネルの出力制御ビット

CCxEビット	tim_ocx出力状態
0	出力無効（タイマによって駆動されない：ハイインピーダンス）
1	出力は有効（tim_ocx = tim_ocxref + 極性）

注： CCxE=0のとき、標準tim_ocxチャネルに接続されている外部IOピンの状態は、GPIOレジスタのみに依存します。

55.5.12 TIMx カウンタ（TIMx_CNT）（x = 2～5）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIFCPY_CNT[31]	CNT[30:16]														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UIFCPY_CNT[31]**：値はTIMx_CR1のIUFREMAPに依存します。

UIFREMAP = 0の場合：

CNT[31]：カウンタ値の最上位ビット

UIFREMAP = 1の場合：

UIFCPY：UIFコピー

このビットはTIMx_ISRレジスタのUIFビットの読出し専用コピー

ビット30:0 **CNT[30:0]**：カウンタ値の最下位部分

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[30:0] に非ディザリング部分を保持します。分数部分は得られません。

55.5.13 TIMxプリスケーラ（TIMx_PSC）（x = 2～5）

アドレスオフセット：0x028

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数ftim_cnt_ckは$f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSCは、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGRレジスタのUGビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

55.5.14 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 2〜5）

アドレスオフセット：0x02C

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **ARR[31:0]**：自動再ロード値

ARRは、実際の自動再ロードレジスタにロードされる値です。

ARRの更新と動作の詳細については、セクション 55.4.3：2223ページのタイムベースユニットを参照してください。

自動再ロード値がnullのときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、自動再ロード値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[31:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

55.5.15 TIMx キャプチャ／比較レジスタ1（TIMx_CCR1）（x = 2〜5）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CCR1[31:0]**：キャプチャ／比較1値

CC1チャネルが出力として設定されている場合：

CCR1は、実際のキャプチャ／比較1レジスタにロードされる値（プリロード値）です。TIMx_CCMR1レジスタのOC1PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較1レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc1出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[31:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1チャネルが入力として設定されている場合：

CCR1は、最後の入力キャプチャ1イベント（tim_ic1）によって転送されたカウンタ値です。TIMx_CCR1レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[31:0] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

55.5.16 TIMx キャプチャ／比較レジスタ2（TIMx_CCR2）（x = 2〜5）

アドレスオフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CCR2[31:0]**：キャプチャ／比較2値

CC2チャネルが出力として設定されている場合：

CCR2は、実際のキャプチャ／比較2レジスタにロードされる値（プリロード値）です。TIMx_CCMR2レジスタのOC2PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較2レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc2出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[31:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2チャネルが入力として設定されている場合：

CCR2は、最後の入力キャプチャ2イベント（tim_ic2）によって転送されたカウンタ値です。TIMx_CCR2レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[31:0] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

55.5.17 TIMx キャプチャ／比較レジスタ3（TIMx_CCR3）（x = 2～5）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CCR3[31:0]**：キャプチャ／比較3値

CC3チャネルが出力として設定されている場合：

CCR3は、実際のキャプチャ／比較3レジスタにロードされる値（プリロード値）です。TIMx_CCMR3レジスタのOC3PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較3レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc3出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[31:4] に整数部を保持します。CCR3[3:0] ビットフィールドにはディザリング部分が含まれます。

CC3チャネルが入力として設定されている場合：

CCR3は、最後の入力キャプチャ3イベント（tim_ic3）によって転送されたカウンタ値です。TIMx_CCR3レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[31:0] にキャプチャを保持します。CCR3[3:0] ビットはリセットされています。

55.5.18 TIMx キャプチャ／比較レジスタ4（TIMx_CCR4）（x = 2〜5）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CCR4[31:0]**：キャプチャ／比較4値

CC4チャネルが出力として設定されている場合：

CCR4は、実際のキャプチャ／比較4レジスタにロードされる値（プリロード値）です。TIMx_CCMR4レジスタのOC4PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較4レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc4出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[31:4] に整数部を保持します。CCR4[3:0] ビットフィールドにはディザリング部分が含まれます。

CC4チャネルが入力として設定されている場合：

CCR4は、最後の入力キャプチャ4イベント（tim_ic4）によって転送されたカウンタ値です。TIMx_CCR4レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[31:0] にキャプチャを保持します。CCR4[3:0] ビットはリセットされています。

55.5.19 TIMxタイマエンコーダ制御レジスタ（TIMx_ECR）（x = 2～5）

アドレスオフセット：0x058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PWPRSC[2:0]			PW[7:0]							
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS[1:0]		FIDX	IBLK[1:0]		IDIR[1:0]		IE
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **PWPRSC[2:0]**：パルス生成回路（幅）プリスケーラ

このビットフィールドは、パルス発生回路用のクロックプリスケーラを次のように設定します。

$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$

ビット23:16 **PW[7:0]**：パルス幅

このビットフィールドは、パルスの長さを次のように定義します。

$t_{PW} = PW[7:0] \times t_{PWG}$

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:6 **IPOS[1:0]**：インデックス位置決め

直交エンコーダモード（SMS[3:0] = 0001、0010、0011、1110、1111）で、このビットは、インデックスイベントがカウンタをリセットするAB入力の設定を示します。

00：AB = 00のときにインデックスがカウンタをリセットします。
01：AB = 01のときにインデックスがカウンタをリセットします。
10：AB = 10のときにインデックスがカウンタをリセットします。
11：AB = 11のときにインデックスがカウンタをリセットします。

方向クロックモードまたはクロックプラス方向モード（SMS[3:0] = 1010、1011、1100、1101）で、これらのビットは、インデックスイベントがカウンタをリセットするレベルを示します。双方向クロックモードでは、これは両方のクロック入力に適用されます。

x0：クロックが0のときにインデックスがカウンタをリセットします。
x1：クロックが1のときにインデックスがカウンタをリセットします。

注： IPOS[1] ビットは意味がありません。

ビット5 **FIDX**：最初のインデックス

このビットは、最初のインデックスのみが考慮されるかどうかを示します。

0：インデックスは常にアクティブです。
1：最初のインデックスのみがカウンタをリセットします。

ビット4:3 **IBLK[1:0]**：インデックスのブランキング

このビットは、インデックスイベントがtim_ti3入力によって調整されるかどうかを示します。

00：インデックスは常にアクティブです。
01：インデックスは、CC3Pビットフィールドに従って、tim_ti3入力がアクティブなときに無効になります。
10：インデックスは、CC4Pビットフィールドに従って、tim_ti4入力がアクティブなときに無効になります。
11：予約済み

ビット2:1 **IDIR[1:0]**：インデックスの方向
このビットは、どちらの方向のときにインデックスイベントがカウンタをリセットするかを示します。
00方向に関係なく、インデックスがカウンタをリセットします。
01：アップカウントのときのみ、インデックスがカウンタをリセットします。
10：ダウンカウントのときのみ、インデックスがカウンタをリセットします。
11：予約済み

ビット0 **IE**：インデックスイネーブル
このビットは、インデックスイベントがカウンタをリセットするかどうかを示します。
0：インデックスは無効です。
1：インデックスは有効です。

55.5.20 TIMxタイマ入力選択レジスタ（TIMx_TISEL）（x = 2〜5）

アドレスオフセット：0x05C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:24 **TI4SEL[3:0]**：tim_ti4[15:0] の入力を選択します。
0000：tim_ti4_in0：TIMx_CH4
0001：tim_ti4_in1
.....
1111：tim_ti4_in15
製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **TI3SEL[3:0]**：tim_ti3[15:0] の入力を選択します。
0000：tim_ti3_in0：TIMx_CH3
0001：tim_ti3_in1
.....
1111：tim_ti3_in15
製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **TI2SEL[3:0]**：tim_ti2_in[15:0] の入力を選択します。
0000：tim_ti2_in0：TIMx_CH2
0001：tim_ti2_in1
.....
1111：tim_ti2_in15
製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TI1SEL[3:0]**：tim_ti1[15:0] の入力を選択します。
0000：tim_ti1_in0：TIMx_CH1
0001：tim_ti1_in1
.....
1111：tim_ti1_in15
製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

55.5.21 TIMxオルタネート機能レジスタ1 （TIMx_AF1）（x = 2〜5）

アドレスオフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17:14 **ETRSEL[3:0]**：etr_inソース選択
これらのビットは、etr_in入力ソースを選択します。
0000：tim_etr0：TIMx_ETR入力
0001：tim_etr1
.....
1111：tim_etr15
製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

ビット13:0 予約済みであり、リセット値に保持する必要があります。

55.5.22 TIMxオルタネート機能レジスタ2 （TIMx_AF2）（x = 2～5）

アドレスオフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **OCRSEL[2:0]**：ocref_clrソース選択

これらのビットは、ocref_clr入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

.....

111：tim_ocref_clr7

製品固有の実装についてはセクション 55.4.2：TIM2/TIM3/TIM4/TIM5ピンおよび内部信号を参照してください。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

55.5.23 TIMx DMA制御レジスタ（TIMx_DCR）（x = 2～5）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **DBSS[3:0]**：DMAバーストソース選択

このビットフィールドは、DMAバースト転送をトリガする割込みソースを定義します（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）。

0000：予約済み
0001：更新
0010：CC1
0011：CC2
0100：CC3
0101：CC4
0110：COM
0111：トリガ
その他：予約済み

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **DBL[4:0]**：DMAバースト長

この5bitのベクタは、転送数などDMA転送長（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1転送
00001：2転送
00010：3転送
...
11010：26転送

例：次の転送を考えます：DBL = 7バイトかつDBA = TIM2_CR1。

–DBL = 7バイトおよびDBA = TIM2_CR1が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1アドレス）+ DBA +（DMAインデックス）、ここでDMAインデックス = DBL

この例では、（TIMx_CR1アドレス）+ DBAに7バイトが追加され、データのコピー元／コピー先アドレスが与えられます。この場合、転送は、（TIMx_CR1アドレス）+ DBAから始めて、7つのレジスタに対して行われます。

DMAデータサイズの設定に応じて、いくつかのケースが想定されます。

–DMAデータサイズをハーフワードで設定した場合、7つのレジスタにそれぞれ16bitのデータが転送されます。

–DMA データサイズをバイトで設定した場合も、データは7つのレジスタに転送されます（最初のレジスタには最初のMSBバイトが含まれ、2番目のレジスタには最初のLSBバイトが含まれるなど、以降同様）。タイマへの転送で、DMAによって転送されるデータサイズを指定する必要もあります。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DBA[4:0]**：DMAベースアドレス

この5bitのベクタは、DMA転送のベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し/書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

.....

55.5.24 完全転送のTIMx DMAアドレス（TIMx_DMAR）（x = 2〜5）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAB[31:0]**：DMAバーストアクセスレジスタ

DMARレジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1アドレス）+ （DBA + DMAインデックス）x 4

ここで、TIMx_CR1アドレスは制御レジスタ1のアドレスであり、DBAはTIMx_DCRレジスタで設定されたDMAベースアドレスであり、DMAインデックスはDMA転送によって自動的に制御され、範囲は0からDBLです（DBLはTIMx_DCR内で設定）。

55.5.25 TIMxレジスタマップ

TIMxレジスタは、次の表のようにマップされます。

表569.TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x004	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
	リセット値							0																		0	0	0	0	0			
0x008	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値							0	0			0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x00C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.	Res.	Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値								0	0	0	0							0		0	0	0	0	0		0		0	0	0	0	0
0x010	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値								0	0	0	0									0	0	0	0			0		0	0	0	0	0
0x014	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																										0		0	0	0	0	0
0x018	**TIMx_CCMR1** 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	**TIMx_CCMR2** 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F [3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	O24CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																	0		0	0	0		0	0	0		0	0	0		0	0
0x024	**TIMx_CNT**	UIFCPY_CNT[31]	CNT[30:16]（32bitタイマでのみCNT[31:16]）															CNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表569.TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	**TIMx_ARR**（x = 2〜5）	ARR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x030	予約済み																																
0x034	**TIMx_CCR1**	CCR1 [31:20]（32bitタイマのみ）												CCR1 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	**TIMx_CCR2**	CCR2 [31:20]（32bitタイマのみ）												CCR2 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	**TIMx_CCR3**	CCR3 [31:20]（32bitタイマのみ）												CCR3 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	**TIMx_CCR4**	CCR4 [31:20]（32bitタイマのみ）												CCR4 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044.. 0x054	予約済み	Res.																															
0x058	**TIMx_ECR**	Res.	Res.	Res.	Res.	Res.	PWPRSC [2:0]			PW[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS [1:0]		FIDX	Res.	IBLK [1:0]		IDIR [1:0]		IE
	リセット値						0	0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0	0
0x05C	**TIMx_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.		TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x060	**TIMx_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x064	**TIMx_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x068.. 0x3D8	予約済み	Res.																															
0x3DC	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]				Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値													0	0	0	0				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIMx_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

56 汎用タイマ（TIM15/TIM16/TIM17）

56.1 TIM15/TIM16/TIM17 の概要

TIM15/TIM16/TIM17 タイマは、プログラム可能なプリスケーラによって駆動される16bitの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラとRCCクロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM15/TIM16/TIM17 タイマは完全に独立していて、いかなるリソースも共有しません。TIM15は、セクション 56.4.26：タイマ同期（TIM15のみ）に示すように同期させることができます。

56.2 TIM15の主な機能

TIM15には以下の機能があります。

- 16bit自動再ロードアップカウンタ
- 16bitのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を1から65535の間の値で分周可能
- 次の機能を持つ、最大2つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM生成（エッジモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力（チャネル1のみ）
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA生成：
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力（割込みリクエスト）

56.3 TIM16/TIM17 の主な特長

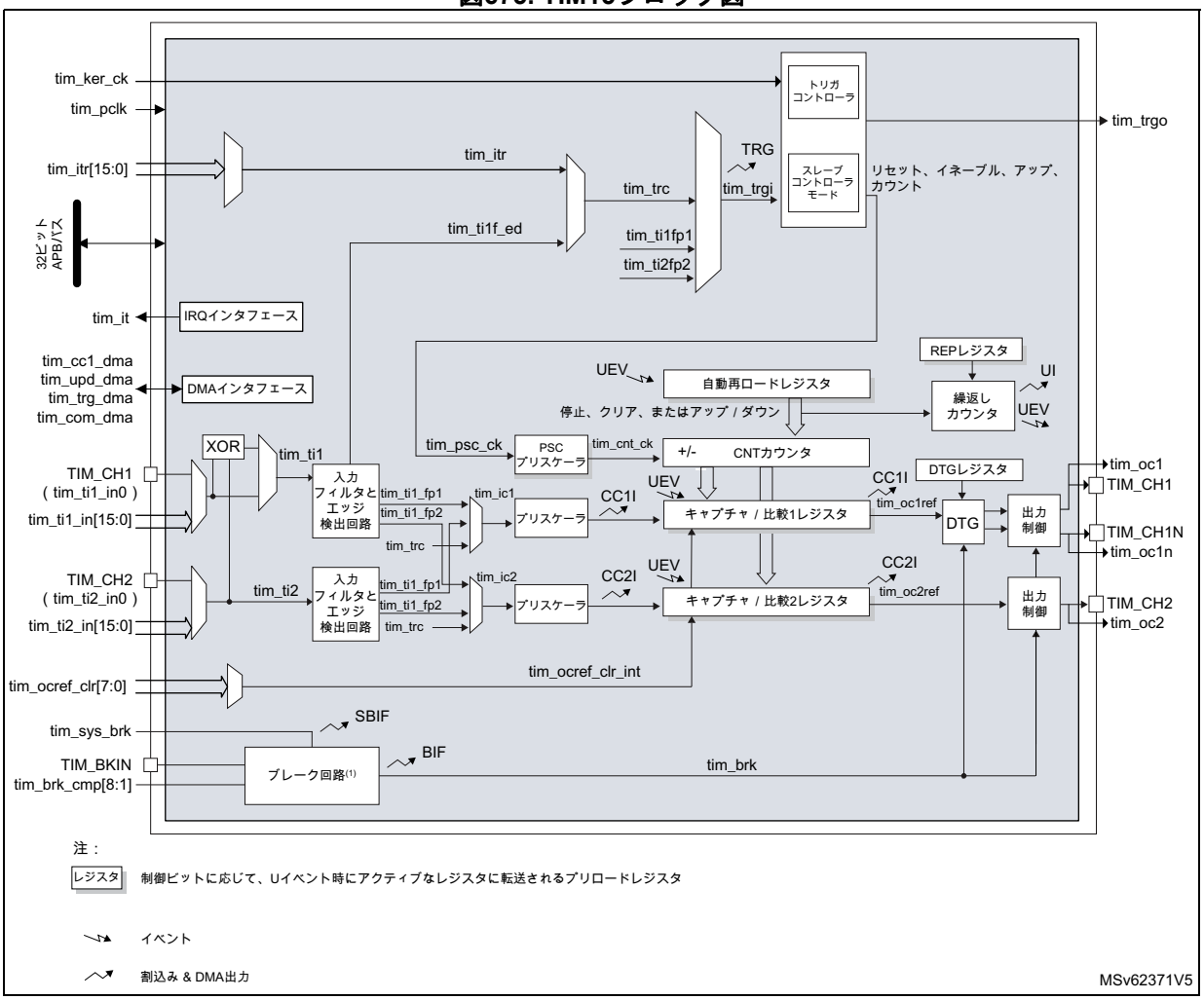
TIM16/TIM17 タイマには以下の機能があります。

- 16bit自動再ロードアップカウンタ
- 16bitのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を1から65535の間の値で分周可能
- 次の機能を持つ、1チャネルタイマ。
  - 入力キャプチャ
  - 出力比較
  - PWM生成（エッジアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA生成：
  - 更新：カウンタオーバーフロー
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力

56.4 TIM15/TIM16/TIM17機能詳細

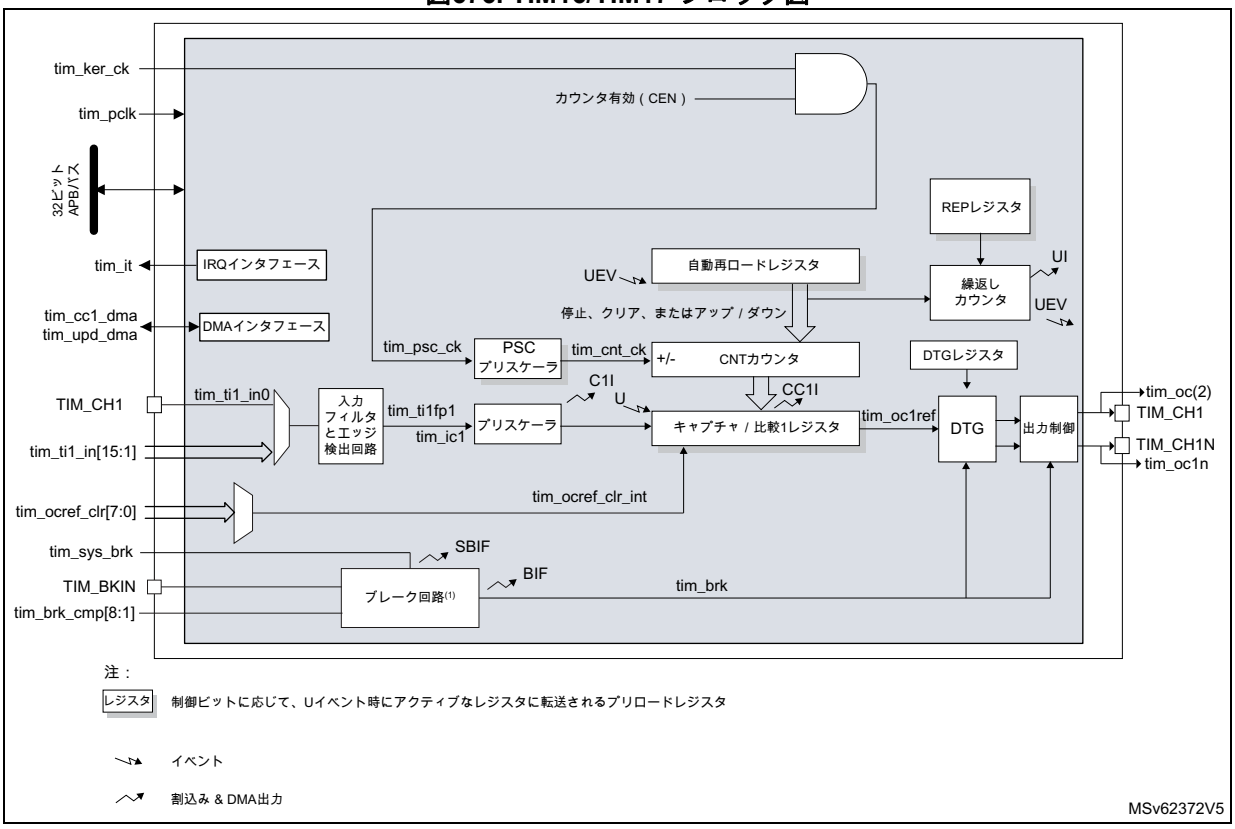
56.4.1 ブロック図

図675. TIM15ブロック図



1. 詳細については、セクション 56.4.15：ブレーク機能の使用を参照してください。

図676. TIM16/TIM17 ブロック図



1. 詳細については、セクション 56.4.15：ブレーク機能の使用を参照してください。
2. この信号は一部のスレーブタイマのトリガとして使用できます（次のセクションの内部トリガ接続表を参照）。詳細については、セクション 56.4.27：タイマ出力を他のタイマのトリガとして使用（TIM16/TIM17 のみ）を参照してください。

56.4.2 TIM15/TIM16/TIM17ピンおよび内部信号

このセクションの表 570および表 571は、TIMの入力および出力の一覧です。

表570.TIMの入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2(1)	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、またはPWMで使用できます。 TIM_CH1 とTIM_CH2 は、外部クロック（tim_ker_ckクロックの1/4未満）、外部トリガの入力としても使用できます。
TIM_CH1N	出力	デッドタイム挿入の可能性があるTIM_CH1出力から導き出されるタイマ相補出力。
TIM_BKIN	入力／出力	ブレーク入力。この入力は双方向モードにも設定することができます。

1. TIM15でのみ使用できます。

表571.TIM内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0](1)	入力	内部タイマ入力バス。これらの入力は、キャプチャ用に、または外部クロック（tim_ker_ckクロックの1/4未満）として使用できます。
tim_itr[15:0]（1）	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ckクロックの1/4未満）として使用できます。
tim_trgo（1）	出力	内部トリガ出力。このトリガは他のオンチップペリフェラルをトリガできます。
tim_ocref_clr[7:0]	入力	タイマのtim_ocref_clr入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim_ocxref信号をクリアするために使用できます。
tim_brk_cmp[8:1]	入力	内部信号のブレーク入力
tim_sys_brk[n:0]	入力	システムブレーク入力この入力はMCUのシステムレベルのエラーを集めます。
tim_pclk	入力	タイマAPBクロック
tim_ker_ck	入力	タイマカーネルクロック。このクロックは、tim_pclk（同じソースから導出）と同期する必要があります。クロック比tim_ker_ck/tim_pclkは整数1、2、3、･･･16（最大値）でなければなりません。
tim_it	出力	キャプチャ／比較、更新、ブレークトリガ、および転流リクエストを取りまとめるグローバルタイマ割込み。
tim_cc1_dma	出力	タイマキャプチャ／比較1 DMAリクエスト
tim_upd_dma	出力	タイマ更新DMAリクエスト
tim_trg_dma	出力	タイマトリガDMAリクエスト
tim_com_dma	出力	タイマ転流DMAリクエスト

1. TIM15でのみ使用できます。

以降の表に、tim_ti[2:1] 入力マルチプレクサに接続されるソースを列挙します。

表572.tim_ti1入力マルチプレクサへの相互接続

tim_ti1入力	ソース		
	TIM15	TIM16	TIM17
tim_ti1_in0	TIM15_CH1	TIM16_CH1	TIM17_CH1
tim_ti1_in1	LSE	予約済み	
tim_ti1_in2	comp1_out	MCO	
tim_ti1_in3	comp2_out(1)	HSE/32	
tim_ti1_in4	予約済み	rtc_wut_trg	
tim_ti1_in5		LSE	
tim_ti1_in6		LSI	

表572.tim_ti1入力マルチプレクサへの相互接続 （続き）

tim_ti1入力	ソース		
	TIM15	TIM16	TIM17
tim_ti1_in7	予約済み	MSIS/1024	
tim_ti1_in8		MSIS/4	
tim_ti1_in9		HSI/256	
tim_ti1_in[15:10]	予約済み		

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表573.tim_ti2入力マルチプレクサへの相互接続

tim_ti2入力	TIM15ソース
tim_ti2_in0	TIM15_CH2
tim_ti2_in1	comp2_out(1)
tim_ti2_in [15:2]	予約済み

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

以下の表に、tim_itr入力マルチプレクサに接続される内部ソースを列挙します。

表574.TIMx内部トリガ接続

Tim_itrx入力	TIM15
tim_itr0	tim1_trgo
tim_itr1	tim2_trgo
tim_itr2	tim3_trgo
tim_itr3	tim4_trgo
tim_itr4	tim5_trgo
tim_itr5	tim8_trgo
tim_itr6	予約済み
tim_itr7	tim16_oc1
tim_itr8	tim17_oc1
tim_itr[15:9]	予約済み

以下の表に、tim_brkおよびtim_brk2入力に接続されるソースを列挙します。

表575.タイマブレーク相互接続

tim_brk入力	TIM15	TIM16	TIM17
TIM_BKIN	TIM15_BKINピン	TIM16_BKINピン	TIM17_BKINピン
tim_brk_cmp1	comp1_out	comp1_out	comp1_out
tim_brk_cmp2	comp2_out(1)	comp2_out（1）	comp2_out（1）
tim_brk_cmp[8:3]	予約済み		

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表576.システムブレーク相互接続

tim_sys_brk入力	TIM15 / TIM16 / TIM17	SYSCFG_CFGR2のイネーブルビット
tim_sys_brk0	Cortex®-M33ロックアップ	CLL
tim_sys_brk1	プログラム可能な電圧検出器（PVD）	PVDL
tim_sys_brk2	SRAMダブルECCエラー	SPL
tim_sys_brk3	Flashメモリダブル ECCエラー	ECCL
tim_sys_brk4	クロックセキュリティシステム（CSS）	なし（常に有効）

以下の表に、tim_ocref_clr入力マルチプレクサに接続される内部ソースを列挙します。

表577.ocref_clr入力マルチプレクサへの相互接続

タイマOCREFクリア信号	タイマOCREFクリア信号割り当て		
	TIM15	TIM16	TIM17
tim_ocref_clr0	comp1_out	comp1_out	comp1_out
tim_ocref_clr1	comp2_out(1)	comp2_out（1）	comp2_out（1）
tim_ocref_clr[7:2]	予約済み		

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

56.4.3 タイムベースユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ16bitアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1レジスタのUDISビットが0である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力tim_cnt_ckから供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIMｘ_CR1レジスタのCENビットがセットされてから、カウンタがカウントを開始するまでに1クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を1から65536の間の値で分周することができます。16bitレジスタ（TIMx_PSCレジスタ）を使って制御される16bitカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図677と図678に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図677. プリスケーラ分周比が1から2に変化したときのカウンタのタイミング図

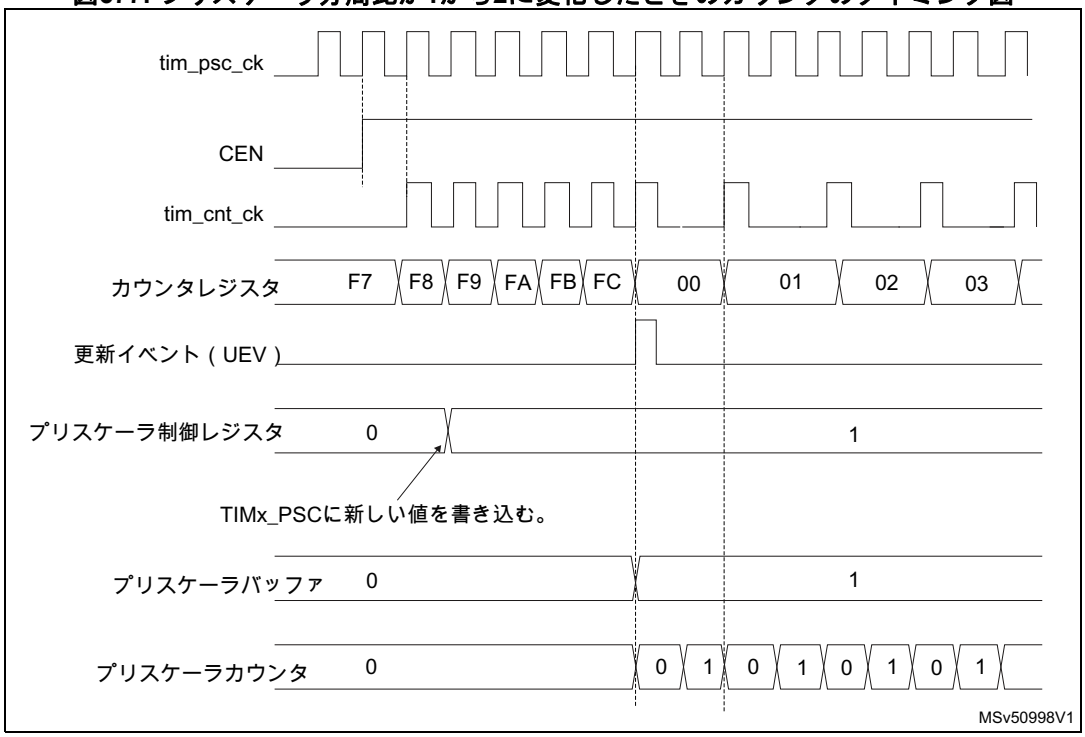
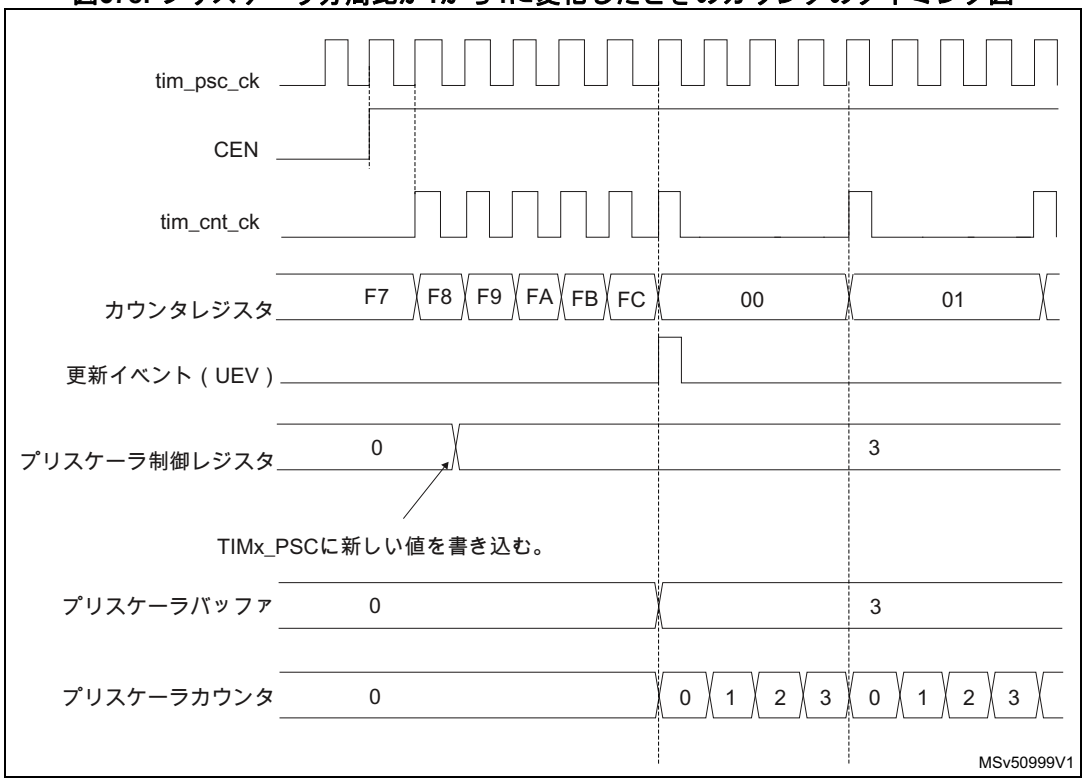


図678. プリスケーラ分周比が1から4に変化したときのカウンタのタイミング図



56.4.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは0から自動再ロード値（TIMx_ARRレジスタの内容）までカウントし、0からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰返しカウンタが使用されている場合には、繰返しカウンタレジスタにプログラムされている回数（TIMx_RCR）までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGRレジスタのUGビットをセットすることでも更新イベントが生成されます。

UEVイベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDISビットに0が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは0からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1レジスタのURSビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（したがって、割込みやDMAリクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットに応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- 繰返しカウンタにはTIMx_RCRレジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図679. 内部クロック分周比が1の場合のカウンタのタイミング図

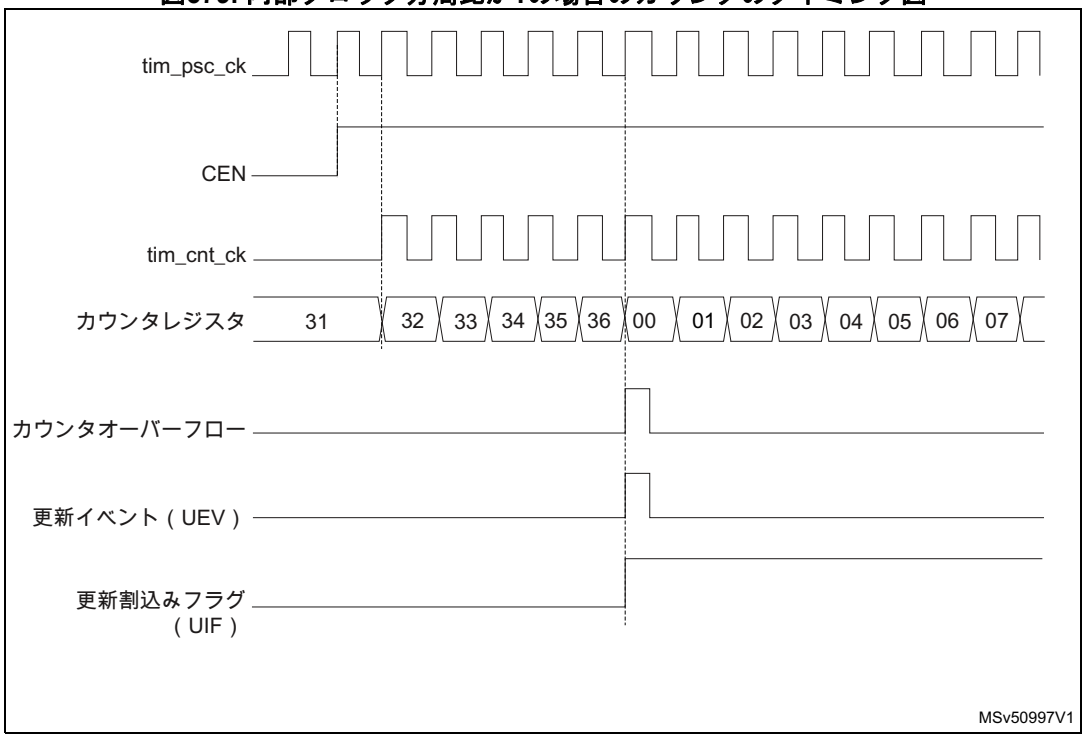


図680. 内部クロック分周比が2の場合のカウンタのタイミング図

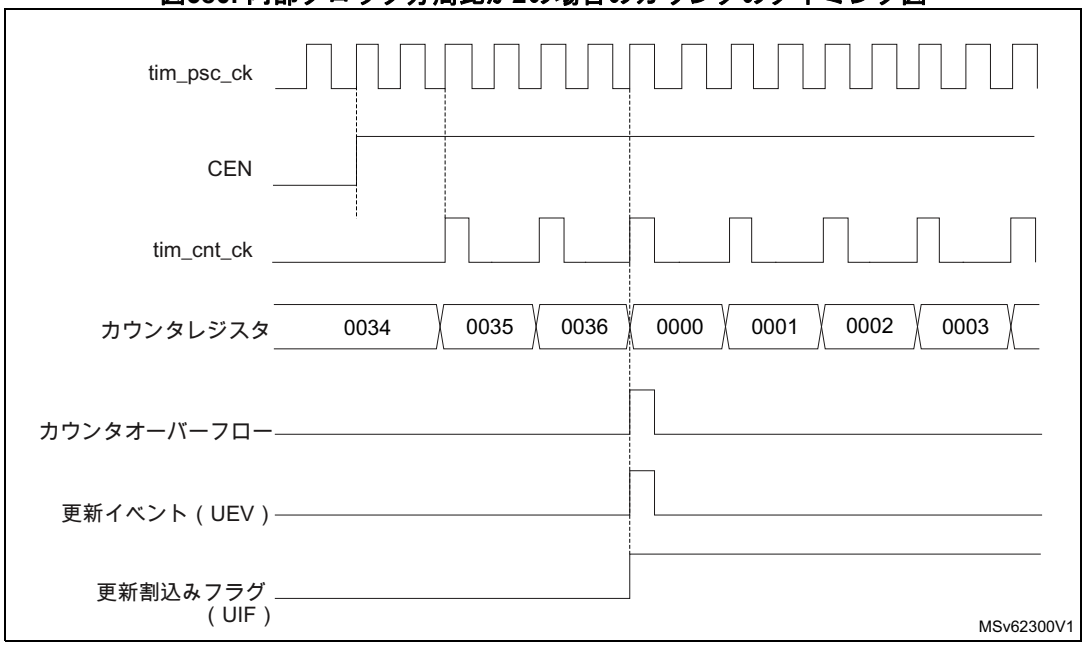


図681. 内部クロック分周比が4の場合のカウンタのタイミング図

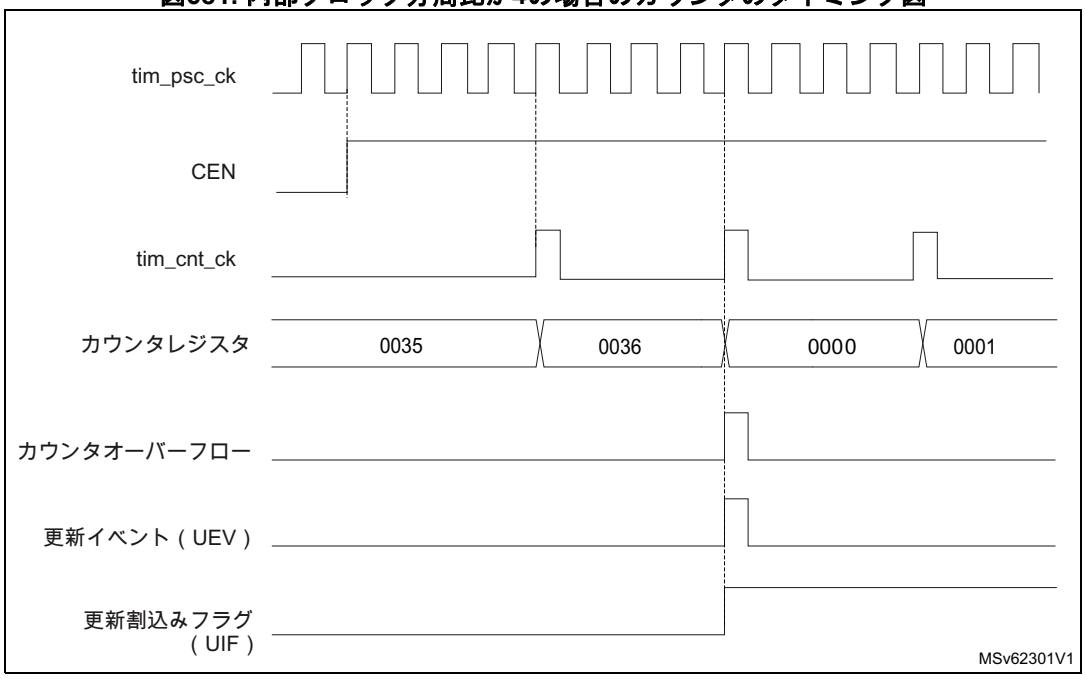


図682. 内部クロック分周比がNの場合のカウンタのタイミング図

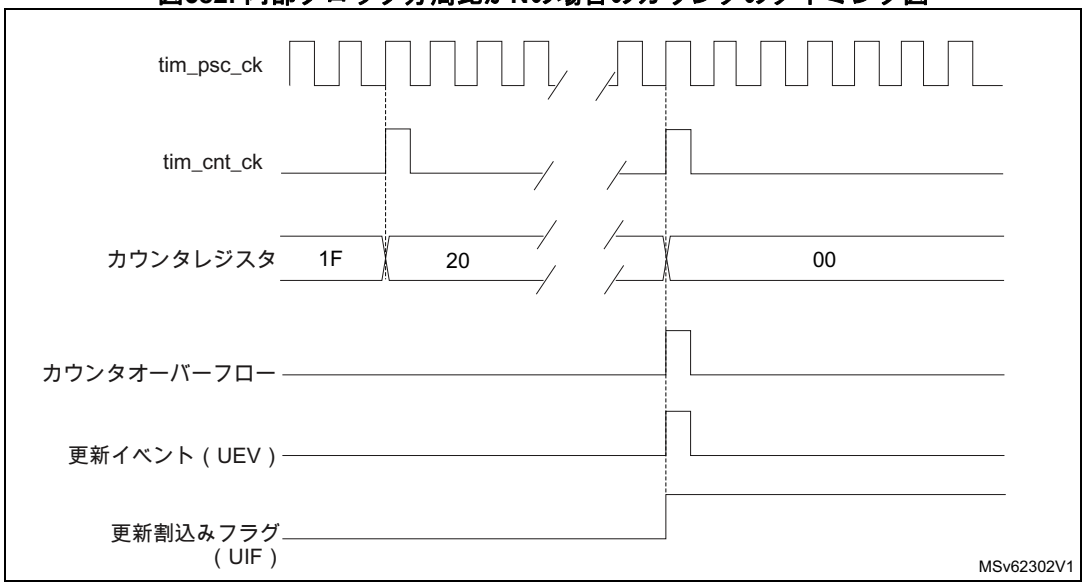


図683. ARPE=0の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARRはプリロードされない）

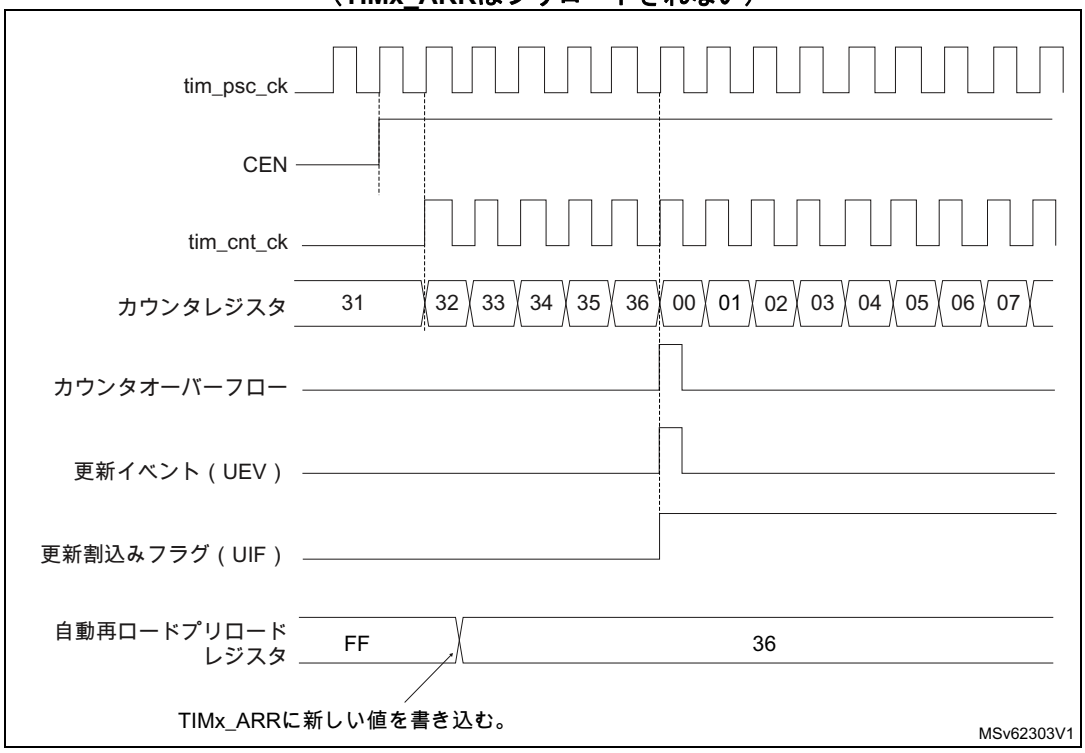
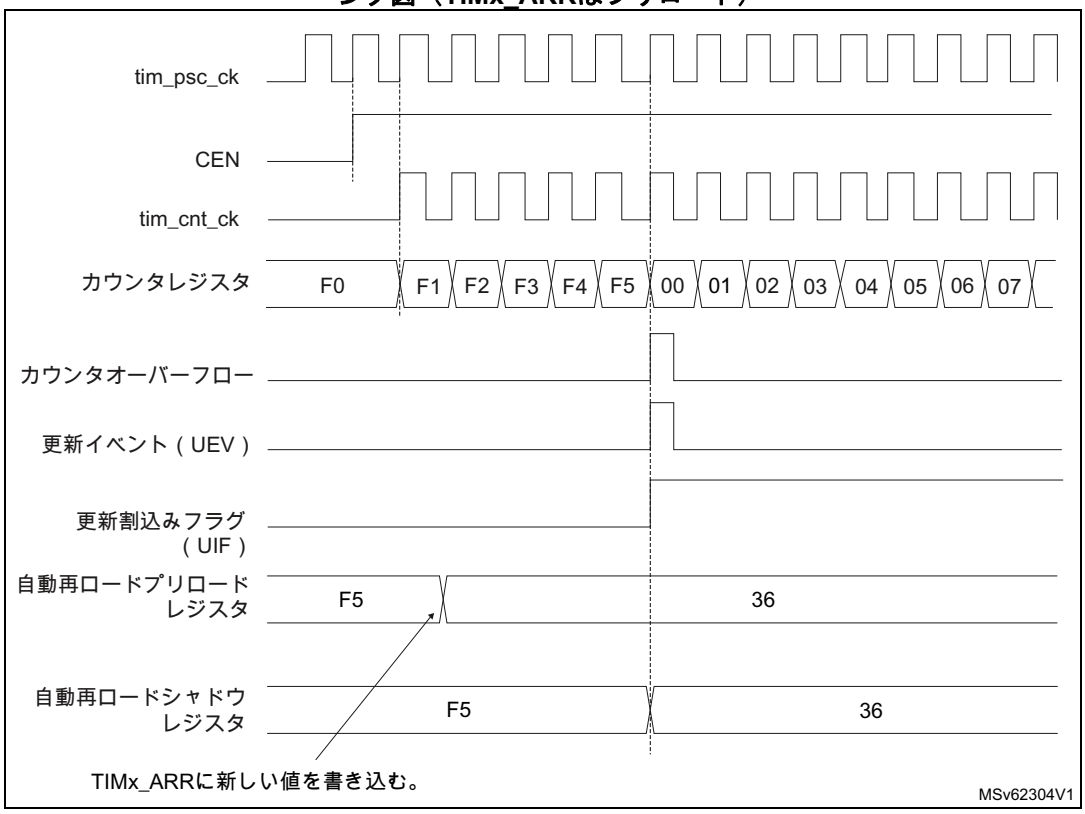


図684. ARPE=1（TIMx_ARRはプリロード）の場合の更新イベント時のカウンタのタイミング図（TIMx_ARRはプリロード）



56.4.5 繰返しカウンタ

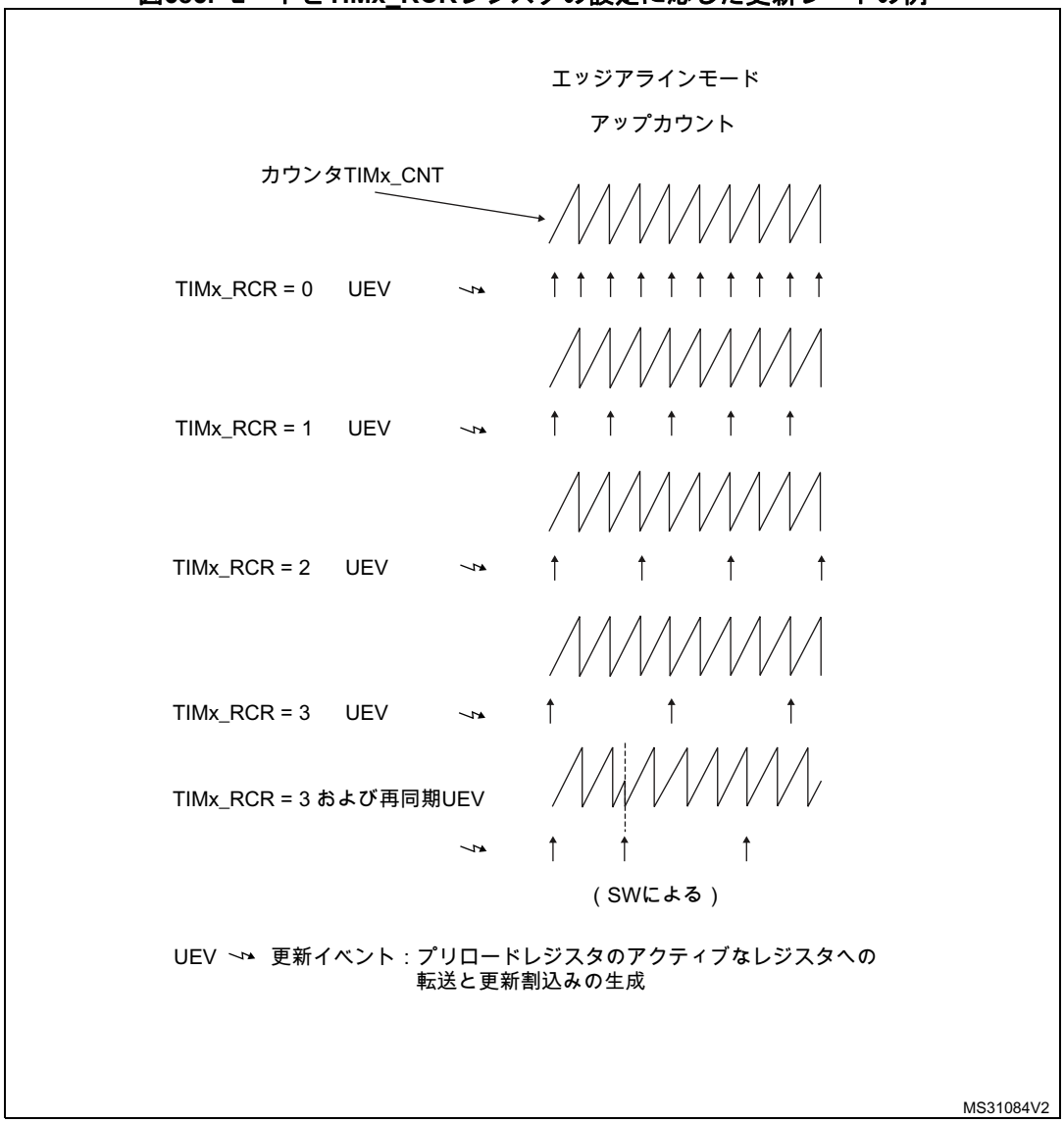
セクション 56.4.3 : タイムベースユニットに、カウンタオーバーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰返しカウンタが0に達したときにのみ、更新イベントが生成されます。これは、PWM信号を生成する際に役立ちます。

これは、TIMx_RCR繰返しカウンタレジスタの値をNとすると、N回目のカウンタオーバーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR自動再ロードレジスタ、TIMx_PSCプリスケーラレジスタ、比較モードのTIMx_CCRxキャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰返しカウンタは、カウンタオーバーフローごとにデクリメントされます。

繰返しカウンタは自動再ロードタイプです。繰返しの回数は、TIMx_RCRレジスタの値によって定義されたとおりに維持されます（図685を参照してください）。ソフトウェアによって（TIMx_EGRレジスタのUGビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰返しカウンタの値にかかわらず直ちにイベントが発生し、繰返しカウンタにTIMx_RCRレジスタの内容が再ロードされます。

図685. モードとTIMx_RCRレジスタの設定に応じた更新レートの例



56.4.6 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

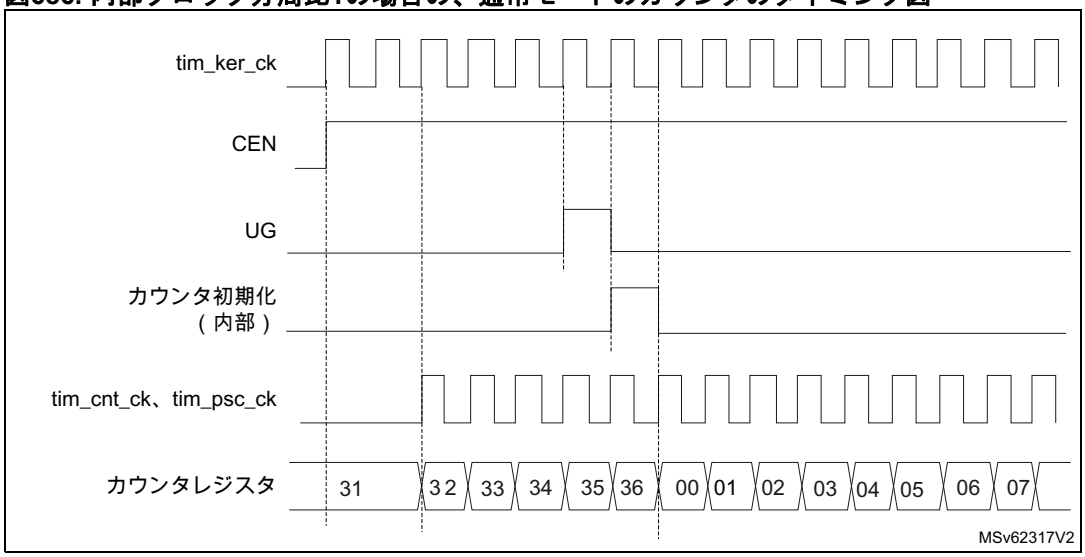
- 内部クロック（tim_ker_ck）
- 外部クロックモード1：外部入力ピン（使用可能な場合、tim_ti1またはtim_ti2）
- 内部トリガ入力（tim_itrx）（TIM15 の場合のみ）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、TIM1がTIM15のプリスケーラとして機能するように設定できます。詳細については、タイマを使用して別のタイマを有効にするを参照してください。

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（SMS=000）、CEN（TIMx_CR1レジスタ）、およびUGビット（TIMx_EGRレジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれるUGビットを除きます）。CENビットに1が書き込まれるとすぐに、プリスケーラには内部クロックtim_ker_ckが供給されます。

図686に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

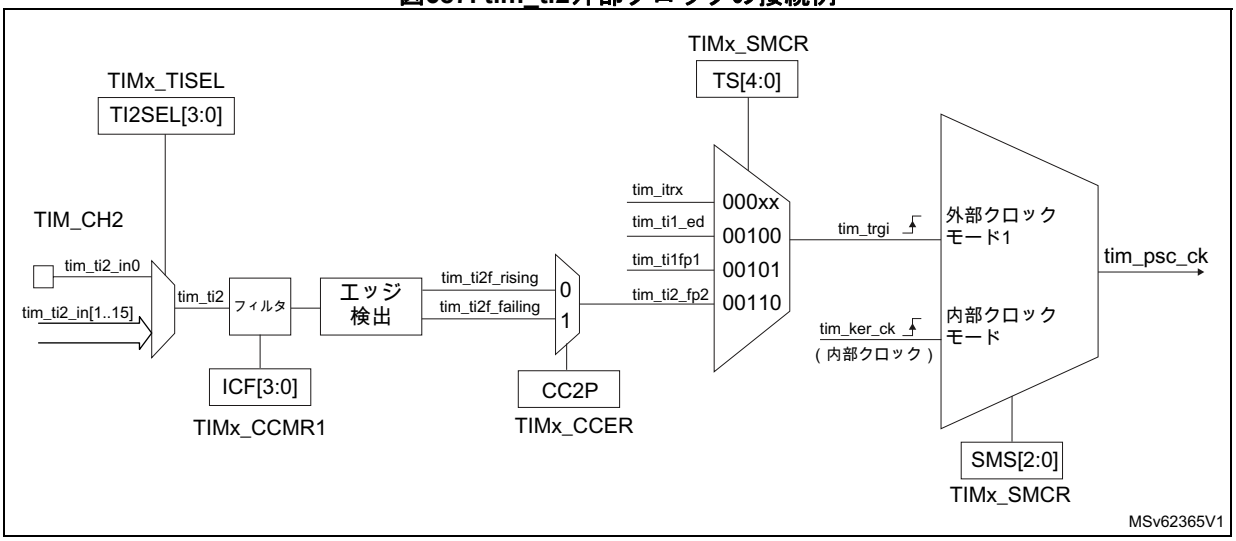
図686. 内部クロック分周比1の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード1

このモードはTIMx_SMCRレジスタのSMS=111のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図687. tim_ti2外部クロックの接続例



たとえば、tim_ti2入力の立上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

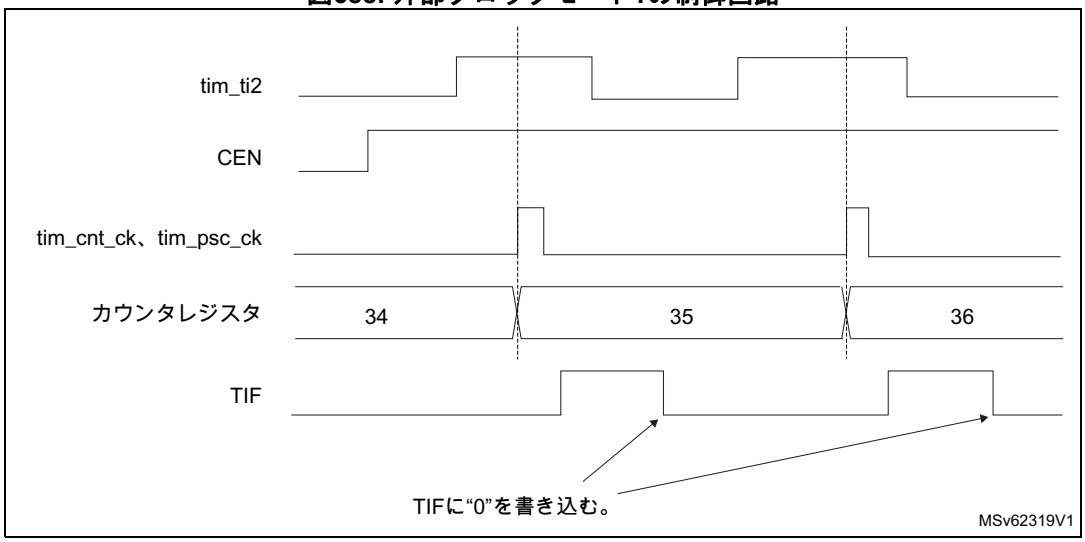
1. TIMx_TISELレジスタのTI2SEL[3:0] ビットで、適切なtim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1レジスタのCC2Sビットに“01”を書き込むことによって、チャネル2がtim_ti2入力の立上がりエッジを検出するように設定します。
3. TIMx_CCMR1レジスタのIC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000にしておきます）。
4. CC2P=0をTIMx_CCERレジスタに書き込んで、立上がりエッジ極性を選択します。
5. TIMx_SMCRレジスタにSMS=111を書き込むことによって、タイマを外部クロックモード1に設定します。
6. TIMx_SMCRレジスタにTS=00110を書き込むことによって、トリガ入力ソースとしてtim_ti2を選択します。*/
7. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。**

tim_ti2の立上がりエッジが発生すると、カウンタは1カウントを行い、TIFフラグがセットされます。

tim_ti2の立上がりエッジから実際のカウンタクロックまでの間には、tim_ti2入力の再同期回路による遅延があります。

図688. 外部クロックモード1の制御回路



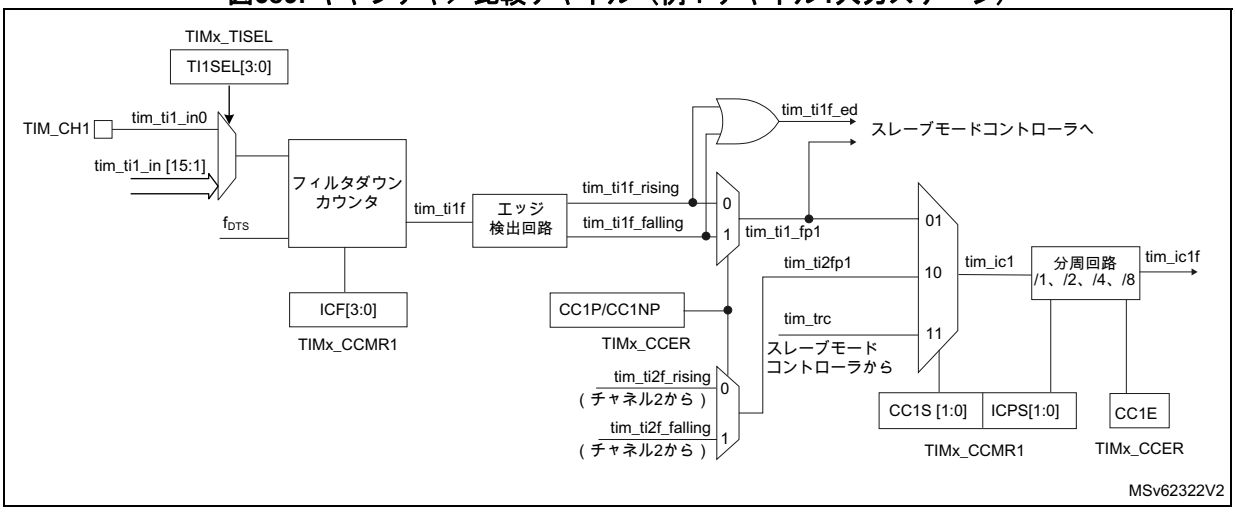
56.4.7 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図689から図692に、1つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応するtim_tix入力をサンプリングして、フィルタリングを行った信号tim_tixfを生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図689. キャプチャ／比較チャネル（例：チャネル1入力ステージ）



出力ステージは、その後、基準として使用される中間波形tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図690. キャプチャ／比較チャネル1メイン回路

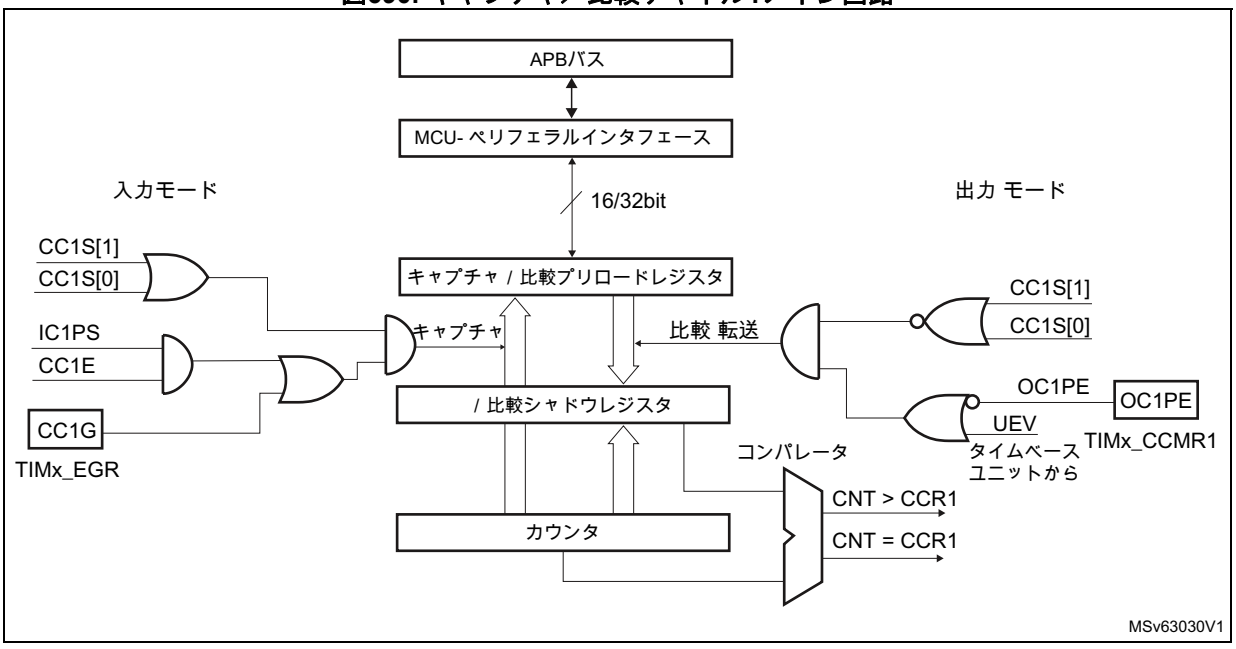


図691. キャプチャ／比較チャネル（チャネル1）の出力ステージ

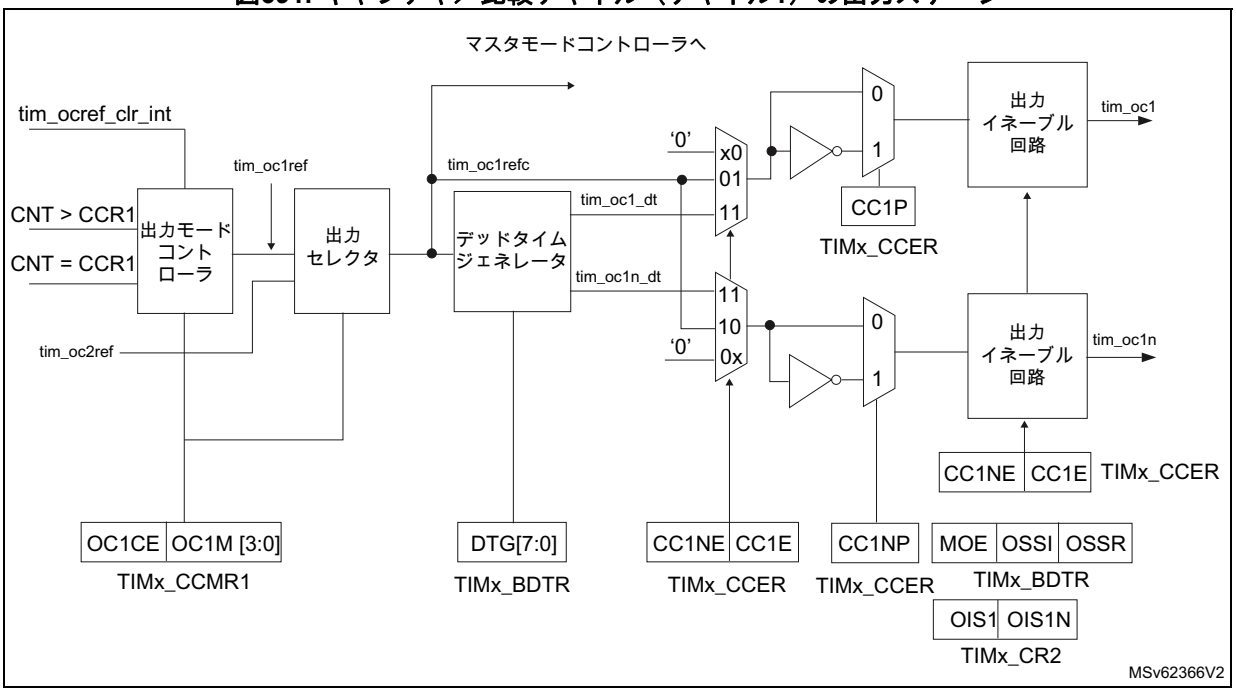
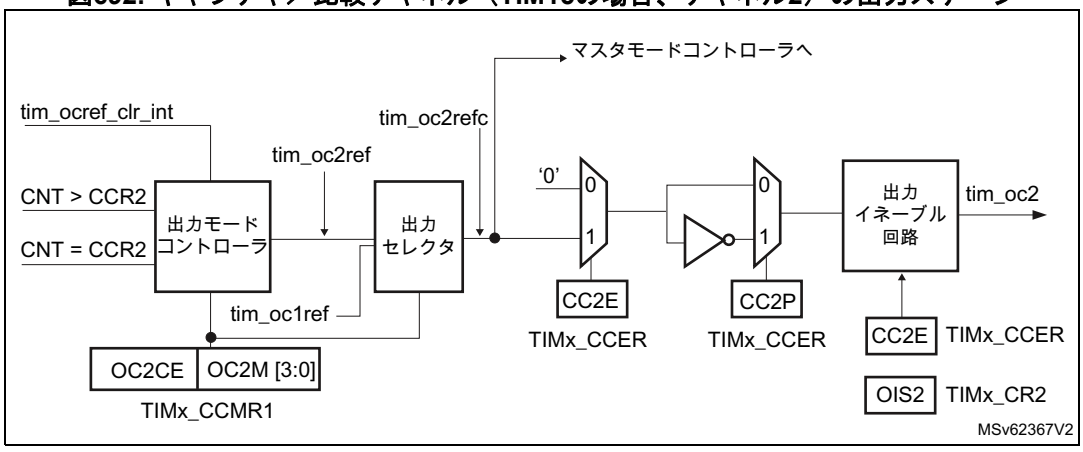


図692. キャプチャ／比較チャネル（TIM15の場合、チャネル2）の出力ステージ



キャプチャ／比較ブロックは、1つのプリロードレジスタと1つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

56.4.8 入力キャプチャモード

入力キャプチャモードでは、対応するtim_icx信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応するCCXIFフラグ（TIMx_SRレジスタ）がセットされ、割込みまたはDMAリクエストを送信できます（有効な場合）。CCxIFフラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグCCxOF（TIMx_SRレジスタ）がセットされます。CCxIFフラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRxレジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOFは、0を書き込むとクリアされます。

次の例は、tim_ti1入力が立ち上がったときに、カウンタの値をTIMx_CCR1にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISELレジスタのTI1SEL[3:0] ビットで、適切なtim_ti1_in[15:1] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1はtim_ti1入力とリンクされていなければならず、このためにはTIMx_CCMR1レジスタのCC1Sビットに01を書き込みます。CC1Sの値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1レジスタは読出し専用になります。
3. タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力がtim_tixの1つである場合、TIMx_CCMRxレジスタのICxFビットによって）。入力信号の反転時、最低でも内部クロックの5サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を5クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した8個のサンプルが検出されたときに、tim_ti1の遷移を検証できます（周波数f_{DTS} でサンプリング）。この場合、TIMx_CCMR1レジスタのIC1Fビットに0011を書き込みます。
4. tim_ti1チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCERレジスタのCC1Pビットに0を書き込みます（この場合、立上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1レジスタのIC1PSビットに“00”を書き込む）。
6. TIMx_CCERレジスタのCC1Eビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIERレジスタのCC1IEビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIERレジスタのCC1DEビットをセットすることによって、DMAリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値がTIMx_CCR1レジスタに格納されます。
- CC1IFフラグがセットされます（割込みフラグ）。CC1OFビットは、少なくとも2回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IEビットに応じて、割込みが生成されます。
- CC1DEビットに応じて、DMAリクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これは、フラグ読出し後、データ読出し前に発生するかもしれないオーバーキャプチャの見落としを避けるためです。

注： **IC割込みとDMAリクエストは、TIMx_EGRレジスタの対応するCCxGビットをセットすることによって、ソフトウェアによって生成することができます。**

56.4.9 PWM入力モード（TIM15の場合のみ）

このモードは、単一のtim_tix入力に接続されたPWM信号の周期とデューティサイクルの両方を以下のように測定するために使用されます。

- TIMx_CCR1レジスタは周期値（2つの連続した立上がりエッジ間の間隔）を保持します。
- TIMx_CCR2 レジスタはパルス幅（2 つの連続した立上がりと立下がりエッジ間の間隔）を保持します。

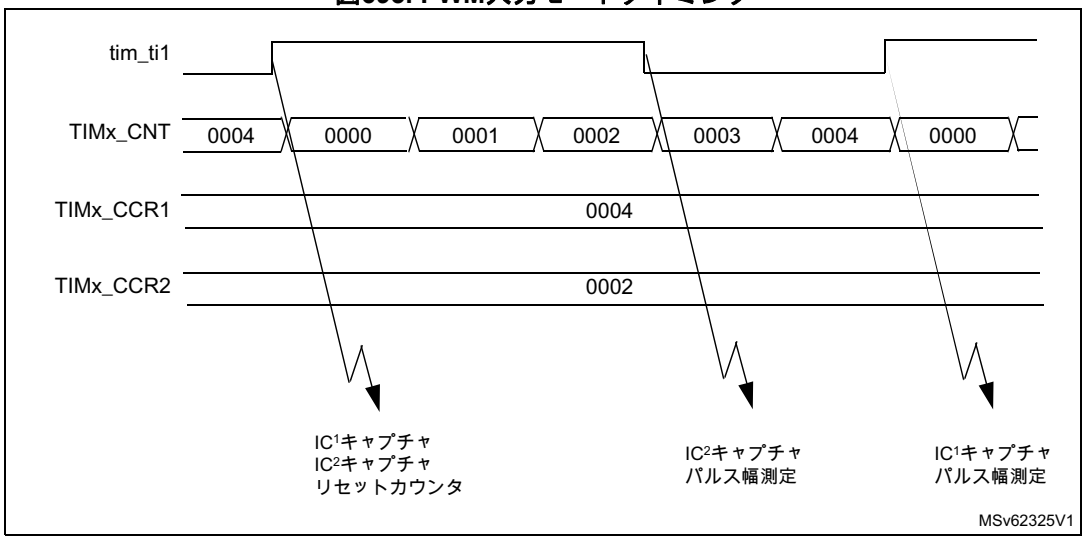
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2つのtim_icx信号が同じtim_tix入力にマッピングされます。
- この2つのtim_icx信号は、逆の極性のエッジでアクティブです。
- 2つのtim_tixfpy信号の1つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、tim_ti1に適用されたPWMの周期（TIMx_CCR1レジスタ）とデューティサイクル（TIMx_CCR2レジスタ）を測定できます（手順は、tim_ker_ck周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISELレジスタのTI1SEL[3:0] ビットで、適切なtim_ti1_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに “01” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0” を書き込むことによって（立上がりエッジでアクティブ）、tim_ti1fp1のアクティブな極性を選択します（TIMx_CCR1のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに “10” を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2PビットとCC2NPビットに“10”を書き込むことによって（立下がりエッジでアクティブ）、tim_ti1fp2のアクティブ極性を選択します（TIMx_CCR2のキャプチャに使用されます）。
6. TIMx_SMCRレジスタのTSビットに00101を書き込むことによって（tim_ti1fp1を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCRレジスタのSMSビットに100を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCERレジスタのCC1EとCC2Eビットに“1”を書き込むことによって、キャプチャを有効にします。

図693. PWM入力モードタイミング



1. tim_ti1fp1 と tim_ti2fp2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2信号でのみ使用できます。

56.4.10 強制出力モード

このモード（TIMx_CCMRxレジスタのCCxSビット = 00）では、各出力比較信号（tim_ocxref、そしてtim_ocx/tim_ocxn）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応するTIMx_CCMRxレジスタのOCxMビットに101を書き込む必要があります。これにより、tim_ocxrefは強制的にハイレベルになり（tim_ocxrefは常にアクティブハイ）、tim_ocxはCCxP極性ビットと逆の値になります。

例：CCxP=0（tim_ocxアクティブハイ）=> tim_ocxは強制的にハイレベルになります。

tim_ocxref信号は、TIMx_CCMRxレジスタのOCxMビットに100を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRxシャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みやDMAリクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

56.4.11 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCERレジスタのCCxPビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか

（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。

- 割込みステータスレジスタのフラグをセットします（TIMx_SRレジスタのCCxIFビット）。
- 対応する割込みマスク（TIMx_DIERレジスタのCCXIEビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIERレジスタのCCxDEビット）がセットされている場合は、DMAリクエストを送信します（DMAリクエスト選択には、TIMx_CR2レジスタのCCDSビットが使用されます）。

TIMx_CCRxレジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRxレジスタのOCxPEビットを使用してプログラミングできます。

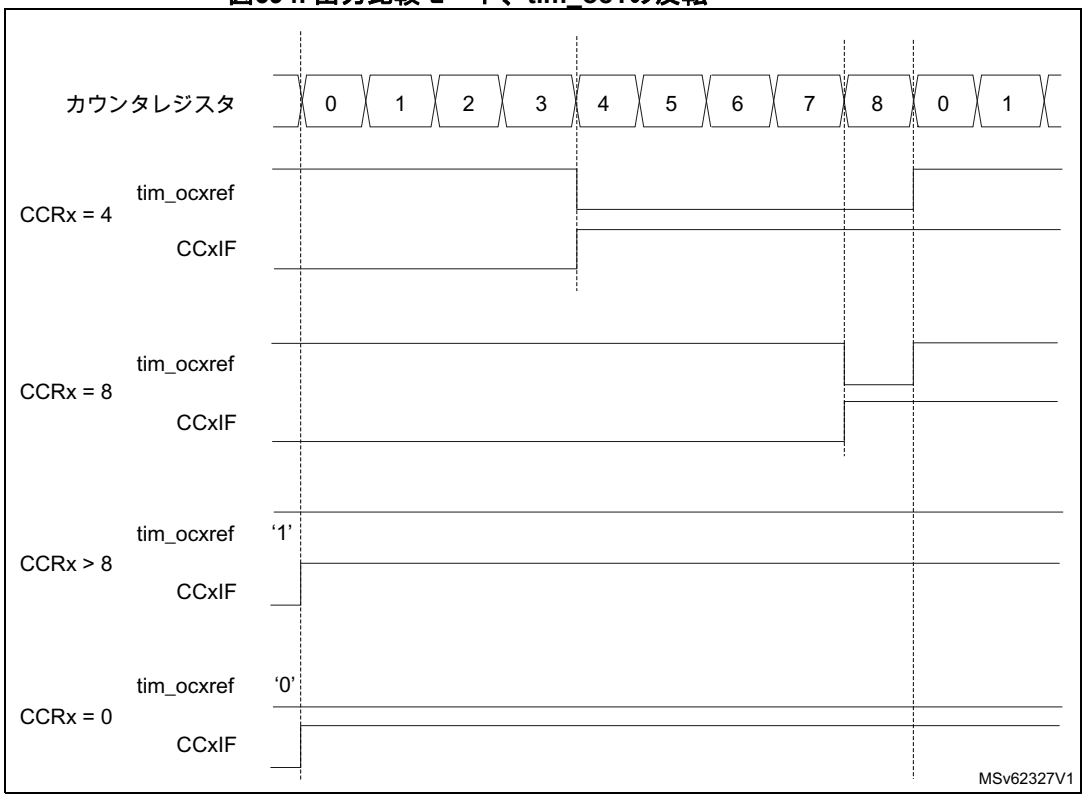
出力比較モードでは、更新イベントUEVはtim_ocxrefおよびtim_ocx出力には影響を与えません。タイミングの分解能はカウンタの1カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARRレジスタとTIMx_CCRxレジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIEビットをセットします。
4. 出力モードを選択します。例：
   - CNTとCCRxが一致したときにtim_ocx出力ピンを反転するには、OCxMビットに011を書き込みます。
   - プリロードレジスタを無効にするには、OCxPEビットに0を書き込みます。
   - アクティブハイ極性を選択するには、CCxPビットに0を書き込みます。
   - 出力を有効にするには、CCxEビットに1を書き込みます。
5. TIMx_CR1レジスタのCENビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによってTIMx_CCRxレジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRxシャドウレジスタは、次の更新イベントUEVでのみ更新されます。例を図694に示します。

図694. 出力比較モード、tim_oc1の反転



56.4.12 PWMモード

パルス幅変調（PWM）モードは、TIMx_ARRレジスタの値によって決められた周波数とTIMx_CCRxレジスタの値によって決められたデューティサイクルの信号を生成するために使用されます。

PWMモードは、TIMx_CCMRxレジスタのOCxMビットに“110”（PWMモード1）または“111”（PWMモード2）を書き込むことによって、チャネルごとに選択できます（tim_ocx出力ごとに1つのPWM）。TIMx_CCMRxレジスタのOCxPEビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1レジスタのARPEビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGRレジスタのUGビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

tim_ocx極性は、TIMx_CCERレジスタのOCxPビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx出力は、CCxE、CCxNE、MOE、OSSI、およびOSSRビット（TIMx_CCERおよびTIMx_BDTRレジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERレジスタの説明を参照してください。

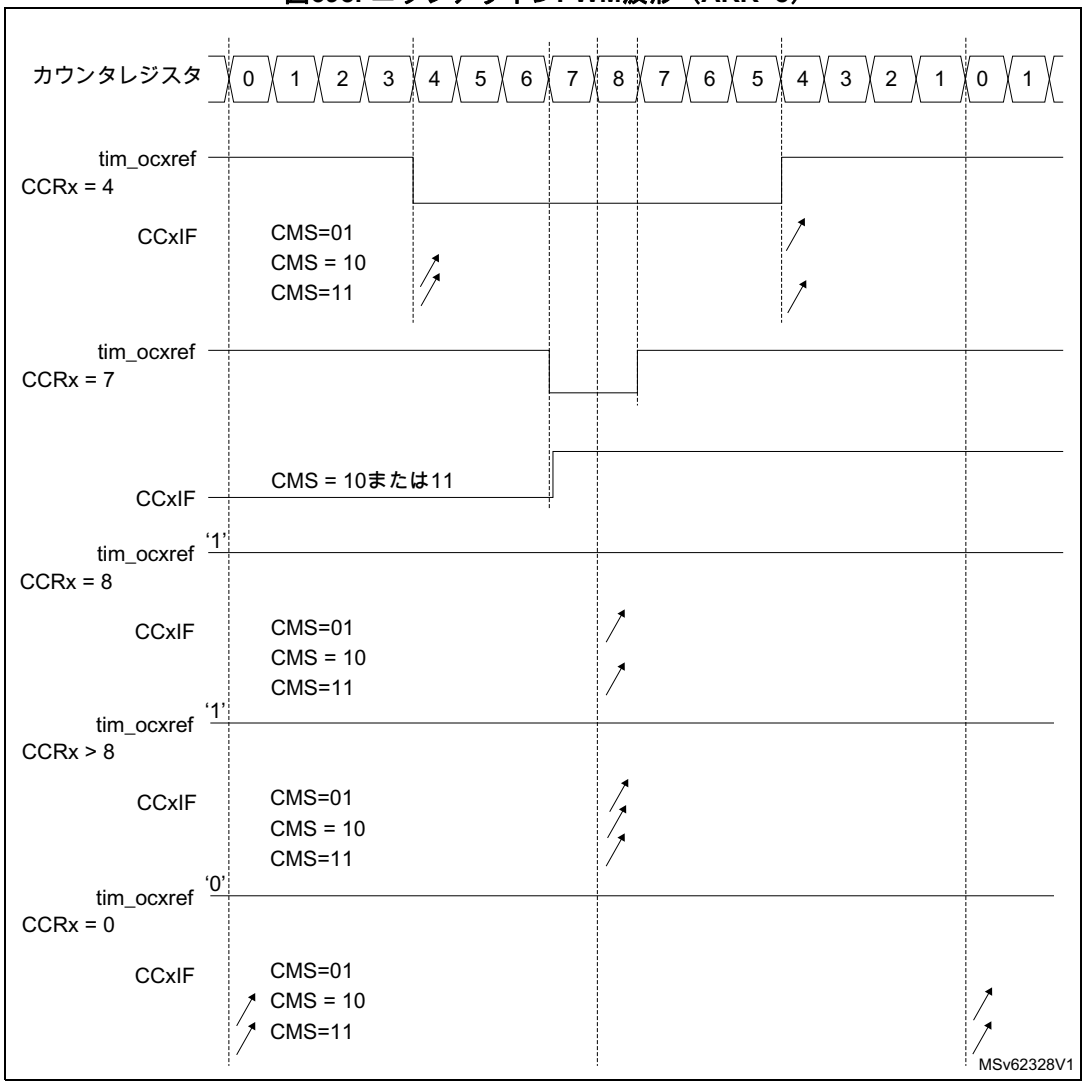
PWMモード（1または2）では、TIMx_CNTとTIMx_CCRxが常に比較されて、TIMx_CCRx ≤ TIMx_CNTまたはTIMx_CNT ≤ TIMx_CCRxかどうかが判断されます（カウントの方向によります）。

TIM15/TIM16/TIM17 はアップカウント動作でのみ使用可能です。2331ページのアップカウントモードを参照してください。

次の例では、PWMモード1を使用しています。PWM基準信号tim_ocxrefは、TIMx_CNT < TIMx_CCRxの間はハイに、そうでない場合はローになります。TIMx_CCRxの比較値がTIMx_ARRの自動再ロード

値より大きい場合、tim_ocxrefは“1”に保持されます。比較値が0の場合、tim_ocxrefは“0”に保持されます。図695にTIMx_ARR=8のときのエッジアラインPWM波形の例を示します。

図695. エッジアラインPWM波形（ARR=8）

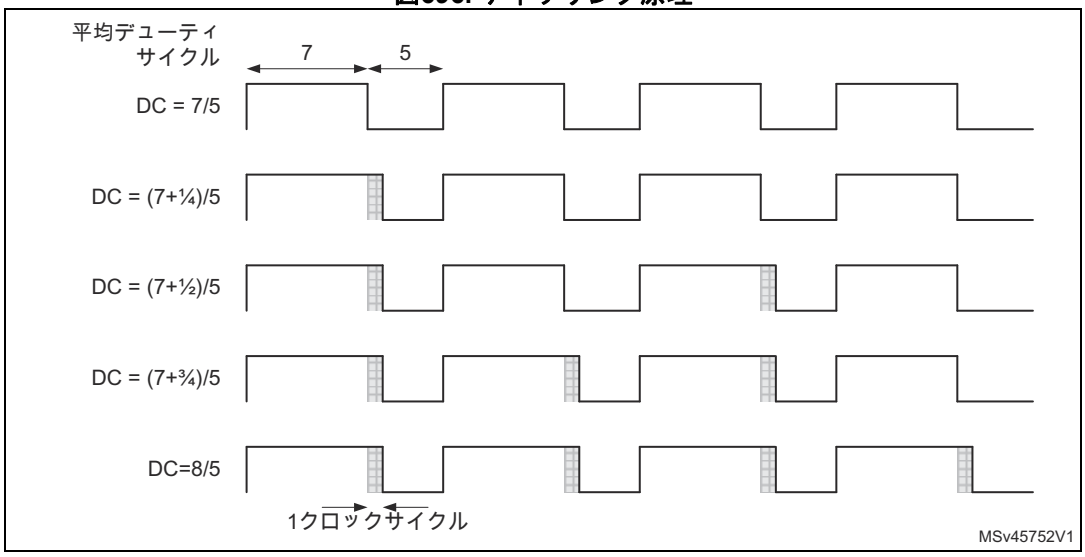


ディザリングモード

PWMモードの有効分解能は、TIMx_CR1レジスタのDITHENビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）とARR（PWM周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16個の連続したPWM周期にわたって実際のCCR（またはARR）値をわずかに変更する（1タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたはPWM周期を考えると、16倍の分解能の向上が可能になります。下記の図696は、4つの連続するPWMサイクルに適用されたディザリング原理を示します。

図696. ディザリング原理

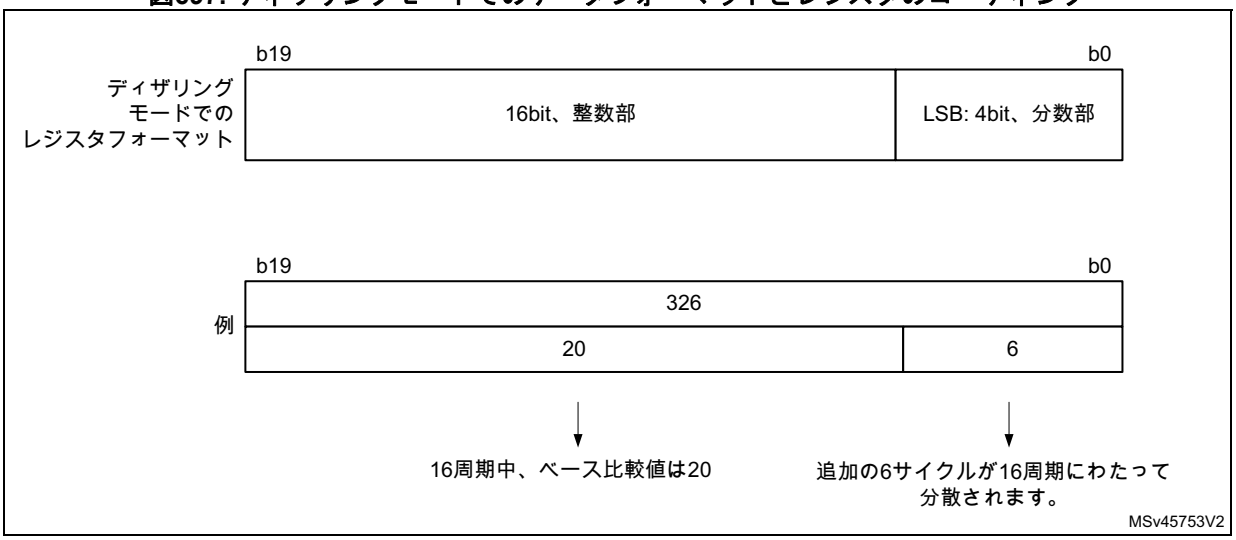


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図697を参照）。

- 下位4bitは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット19:4に左シフトされ、ベース値用のコーディングになります。

注： **DITHENビットをリセットするときは、次の手順に従う必要があります。**
1を参照してください。CENおよびARPEビットをリセットする必要があります。
2を参照してください。ARR[3:0] ビットをリセットする必要があります。
3を参照してください。CCIFフラグをクリアする必要があります。
4を参照してください。CENビットをセットすることができます（最終的にARPE = 1とともに）。

図697. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

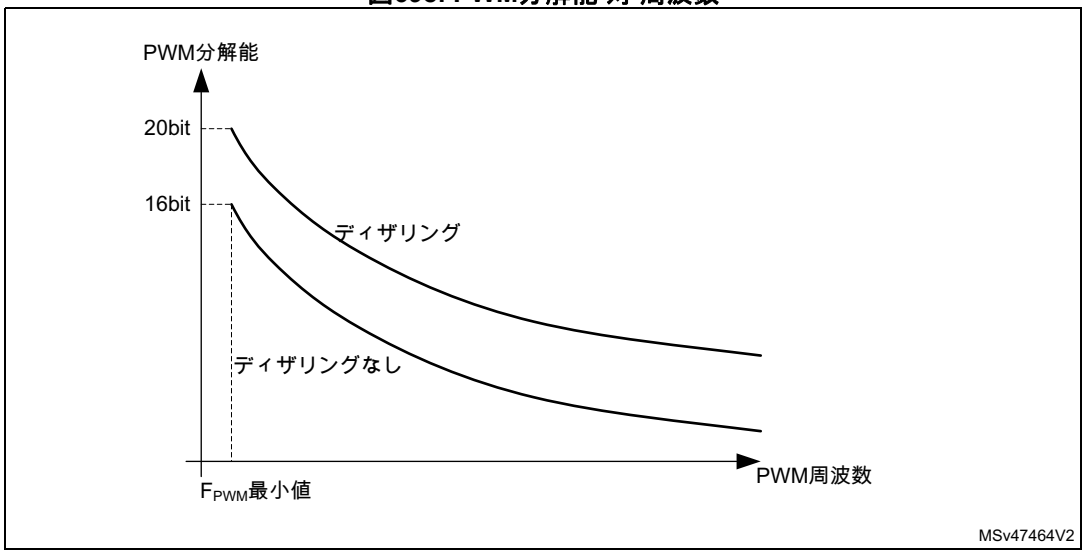
$$\text{ディザリングモードが無効の場合: } F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモードが有効な場合: } F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARRおよびTIMxCCRyの最大値は、ディザリングモードでは0xFFFEFに制限されています（整数部では65534、ディザリング部分では15に対応します）。**

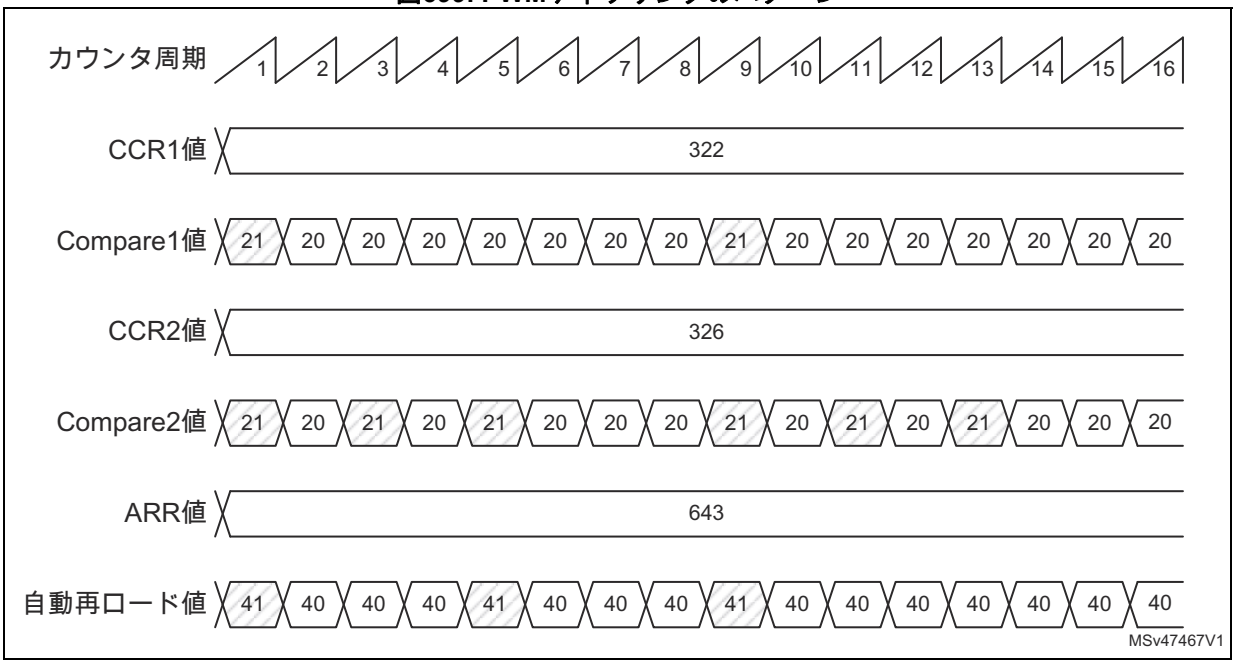
下記の図698に示すように、ディザリングモードはPWM周波数に関係なく、PWM分解能を増加するために使用されます。

図698. PWM分解能 対 周波数



下記の図699で説明するように、デューティサイクルや周期の変更は、連続した16周期にわたって広がります。

図699. PWMディザリングのパターン



自動再ロード値と比較値の増分は、下記の表578に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表578.CCRおよびARRレジスタの変更のディザリングパターン

-	PWM周期															
LSB値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-

表578.CCRおよびARRレジスタの変更のディザリングパターン （続き）

-	PWM周期															
LSB値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

56.4.13 組合せPWMモード（TIM15のみ）

組み合わせPWMモードでは、2つのエッジアラインまたはセンターアラインPWM信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数がTIMx_ARRレジスタの値で決定されるのに対し、デューティサイクルや遅延は2つのTIMx_CCRxレジスタで決定されます。結果として得られる信号tim_ocxrefcは、2つのPWM基準信号のORまたはANDによる論理結合から成ります。

- tim_oc1refc（またはtim_oc2refc）は、TIMx_CCR1およびTIMx_CCR2レジスタによって制御されます。

組合せPWMモードは、TIMx_CCMRxレジスタのOCxMビットに“1100”（組合せPWMモード1）または“1101”（組合せPWMモード2）を書き込むことによって、2チャネルごとに選択できます（CCRレジスタペアごとに1つのtim_ocx出力）。

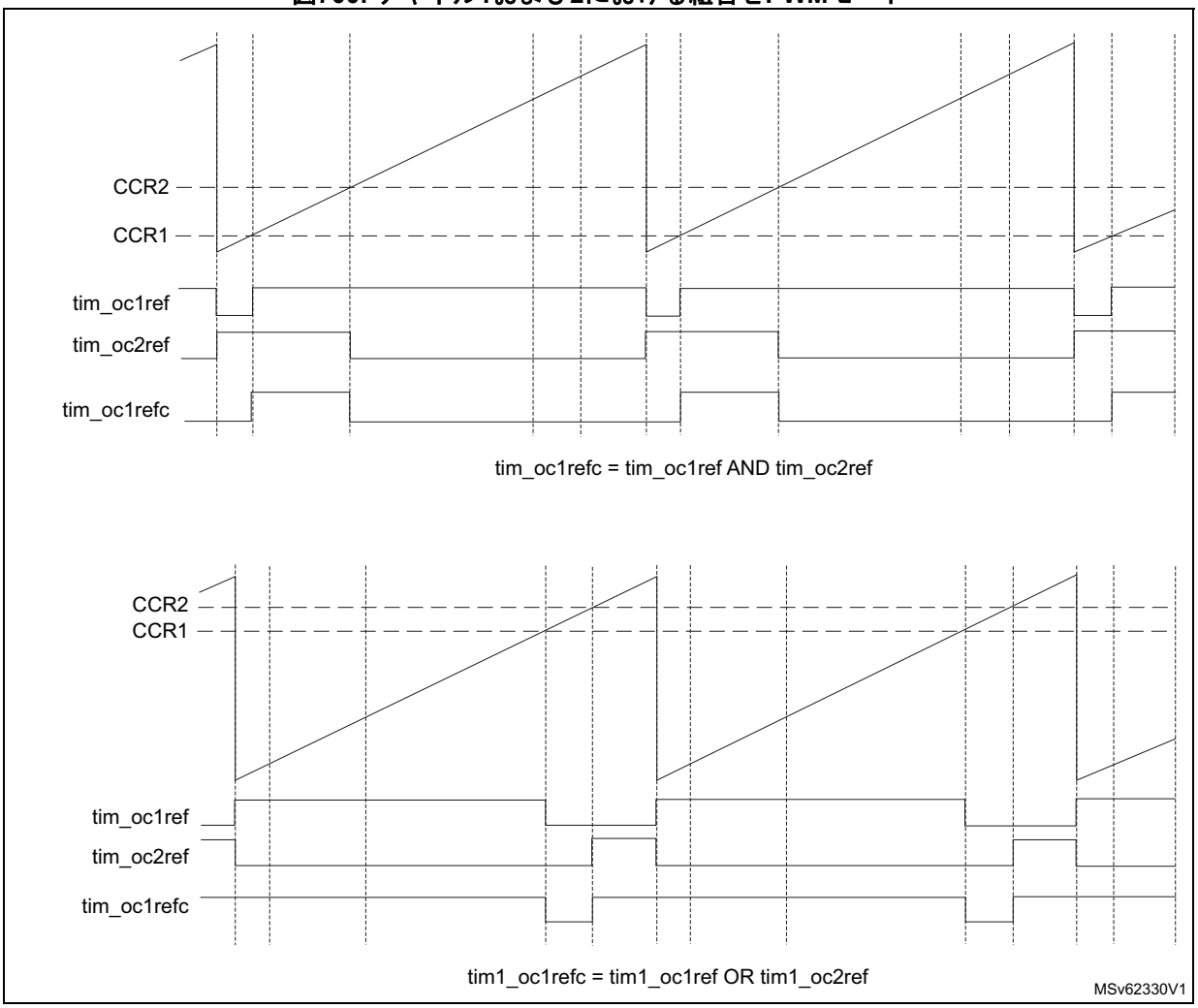
特定のチャネルが組合せPWMチャネルとして使用されている場合、相補チャネルを反対のPWMモードに設定する必要があります（たとえば、1つを組合せPWMモード1、もう1つを組合せPWMモード2にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

図700は、次の設定で取得可能な組合せPWMモードを使用して生成される信号の例を表します。

- チャネル1が組み合わせPWMモード2で設定されている場合
- チャネル2がPWMモード1で設定されている場合

図700. チャネル1および2における組合せPWMモード



56.4.14 相補出力とデッドタイム挿入

TIM15/TIM16/TIM17 汎用タイマは、1つの相補信号を出力して、出力時のスイッチオフおよびスイッチオンを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 tim_ocx または相補 tim_ocxn）の極性は出力ごとに独自に選択できます。これはTIMx_CCERレジスタのCCxPビットおよびCCxNPビットへの書込みによって行います。

相補信号 tim_ocx および tim_ocxn は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタとTIMx_CR2 レジスタのMOE、OISx、OISxN、OSSI、およびOSSR ビットといった複数の制御ビットの組合せによって有効になります。詳細については、表 585：2413ページのブレーク機能を持つ相補tim_oc1およびtim_oc1nチャネルの出力制御ビット（TIM16/TIM17）を参照してください。特に、IDLE状態に切り替わるとき（MOEが0になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxEビットとCCxNEビットの両方をセットし、ブレーク回路がある場合は、さらにMOEビットをセットすることによって有効になります。各チャネルに1つの10bitデッドタイム

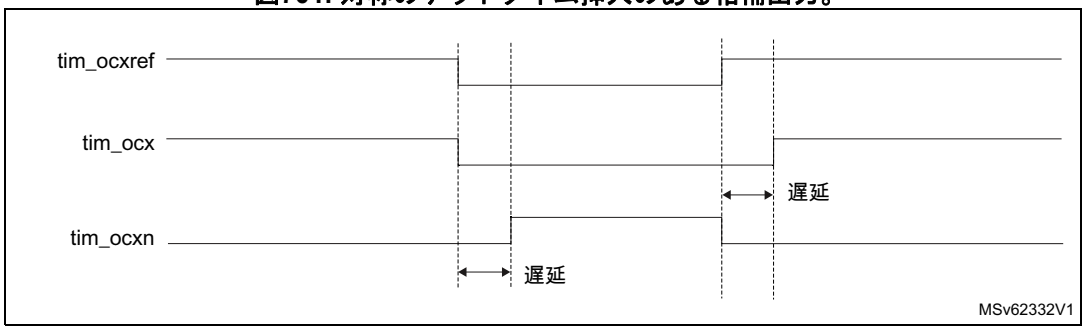
ジェネレータがあります。この回路は、基準波形tim_ocxrefからtim_ocxとtim_ocxnの2つの出力を生成します。tim_ocxとtim_ocxnがアクティブハイの場合、

- tim_ocx出力信号は基準信号と同じですが、立上がりエッジが基準の立上がりエッジより遅い点が異なります。
- tim_ocxn出力信号は、立上がりエッジが基準波形の立下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（tim_ocxまたはtim_ocxn）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号tim_ocxrefとの関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、およびCCxNE=1を想定しています。）

図701. 対称のデッドタイム挿入のある相補出力。



TIMx_DTR2のDTAEビットは、図702に示すように、デッドタイム値を、基準信号の立上がりエッジと立下がりエッジに対して変えるために使用されます。

非対称モード（DTAE = 1）では、立上がりエッジ基準デッドタイムはTIMx_BDTRレジスタのDTG[7:0]ビットフィールドによって定義され、立下がりエッジ基準はTIMx_DTR2レジスタのDTGF[7:0] ビットフィールドによって定義されます 。DTAEビットは、カウンタを有効にする前に書き込む必要があり、CEN=1の間は変更してはなりません。

プリロードメカニズムを使用して、PWM運転中にデッドタイム値を動作中に更新することができます。デッドタイムビットフィールドDTG[7:0] とDTGF[7:0] は、TIMX_DTR2レジスタでDTPEビットがセットされるとプリロードされます。プリロード値は、次の更新イベントでアクティブレジスタにロードされます。

注： **カウンタが有効なときにDTPE ビットを有効にした場合、最後の更新以降に書き込まれた新しい値は破棄され、以前の値が使用されます。**

図702. 非対称のデッドタイム

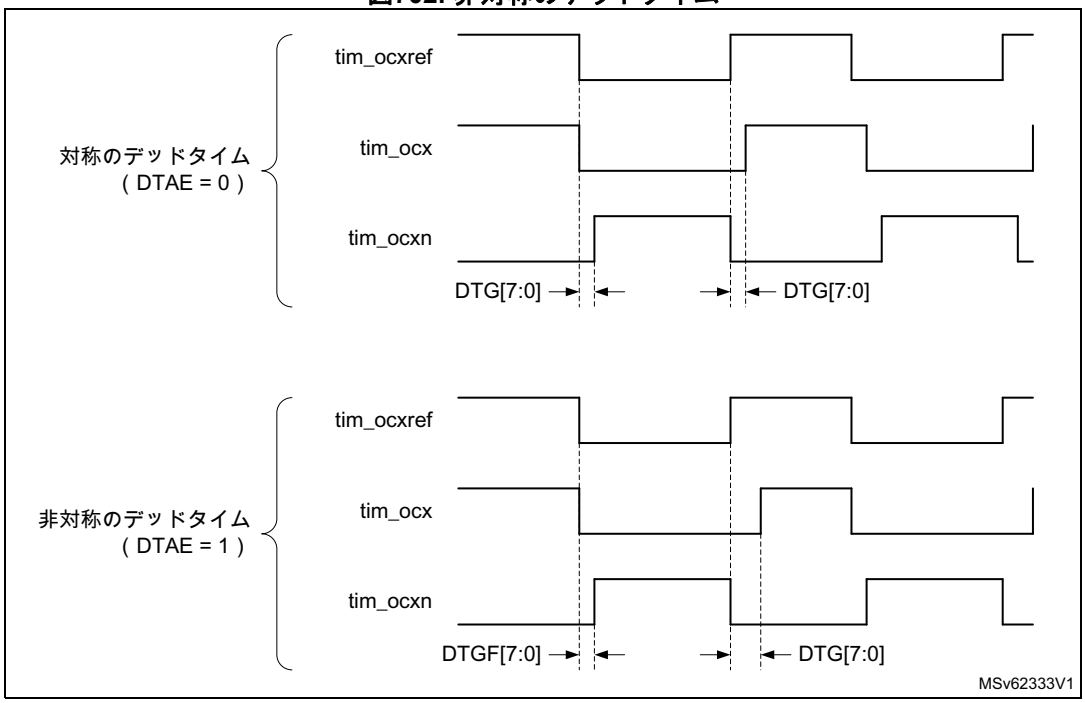


図703. 負のパルスより長い遅延があるときのデッドタイムの波形

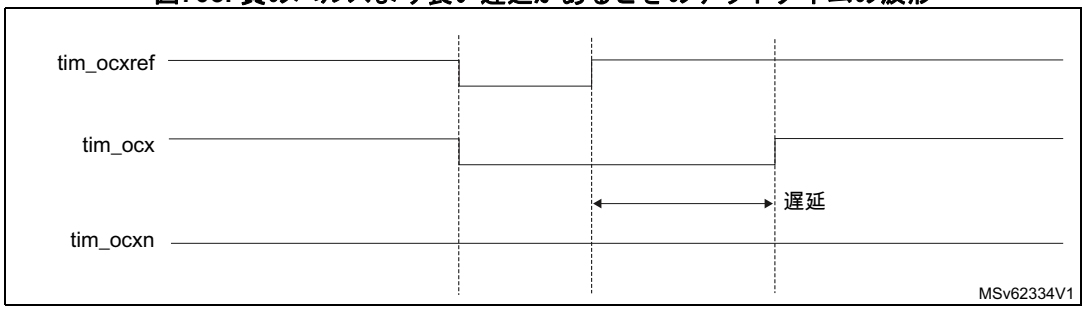
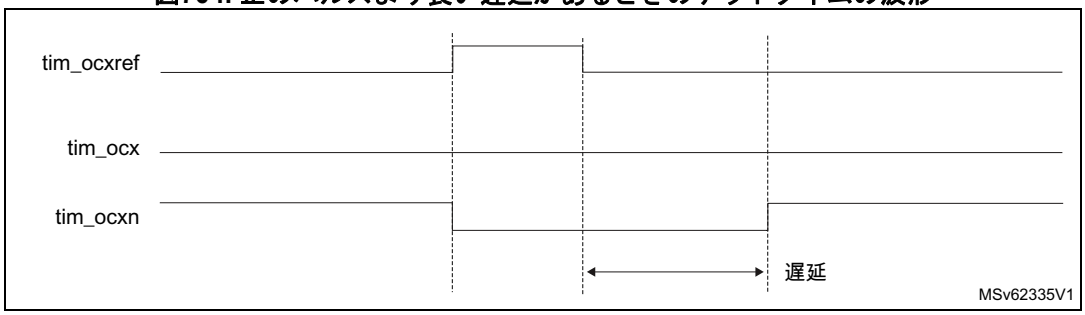


図704. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTRレジスタのDTGビットでプログラム可能です。遅延計算については、セクション 56.8.14： 2417ページのTIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）を参照してください。

tim_ocxref信号のtim_ocxまたはtim_ocxnへのリダイレクト

出力モード（強制、出力比較、またはPWM）では、TIMx_CCERレジスタのCCxEビットおよびCCxNEビットを設定することによって、tim_ocxref信号をtim_ocx出力またはtim_ocxn出力にリダイレクトできます。

これは、特定の波形（PWMまたは静的アクティブレベルなど）を一方の出力に送信し、相補信号を非アクティブレベルに固定するために使用されます。他の例としては、両方の出力を非アクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **tim_ocxnのみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、tim_ocxrefがハイレベルになるとすぐにアクティブになります。たとえば、CCxNP=0の場合は、tim_ocxn=tim_ocxrefになります。他方、tim_ocxとtim_ocxnの両方が有効なときには（CCxE=CCxNE=1）、tim_ocxrefがハイになるとtim_ocxはアクティブになり、tim_ocxnは相補信号となり、tim_ocxrefがローのときにアクティブになります。**

56.4.15 ブレーク機能の使用

ブレーク機能の目的は、タイマによって生成されるPWM信号によって駆動する電源スイッチを保護することです。ブレーク入力は通常、パワーステージおよび3相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路はPWM出力を遮断し、強制的に事前定義された安全な状態に移行させます。

このブレークチャネルは、システムレベル障害（クロック障害、ECC/パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵比較回路からの障害）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたは非アクティブ）に強制できます。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTRレジスタのMOEビット。ソフトウェアによって出力を有効／無効にするために使用され、ブレーク または ブレーク2イベント時にリセットされます。
- TIMx_BDTRレジスタのOSSIビット。出力を非アクティブ状態で制御するか、GPIOコントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2レジスタのOISxおよびOISxNビット。アクティブまたは非アクティブな出力遮断レベルをセットします。OISxおよびOISxNの値にかかわらず、一度にtim_ocx出力とtim_ocxn出力を両方ともアクティブレベルにセットすることはできません。詳細については、表 585：2413ページのブレーク機能を持つ相補tim_oc1およびtim_oc1nチャネルの出力制御ビット（TIM16/TIM17）を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOEビットはローになります。ブレーク機能は、TIMx_BDTRレジスタのBKEビットをセットすることによって有効になります。ブレーク入力の極性は、同じレジスタのBKPビットを設定することによって選択できます。BKEとBKPは、同時に変更できます。BKEおよびBKPビットが書き込まれるとき、書込みが有効になるまでに1 APBクロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには1 APBクロックサイクル待つ必要があります。

MOEの立下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTRレジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOEがローになった後で1に設定された場合、MOEを正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

ブレークは、次を持つtim_brk入力によって生成されます。

- プログラム可能な極性（TIMx_BDTRレジスタのBKPビット）
- プログラム可能なイネーブルビット（TIMx_BDTRレジスタのBKEビット）
- 誤ったイベントを避けるためのプログラム可能なフィルタ（TIMx_BDTR レジスタの BKF[3:0] ビット）

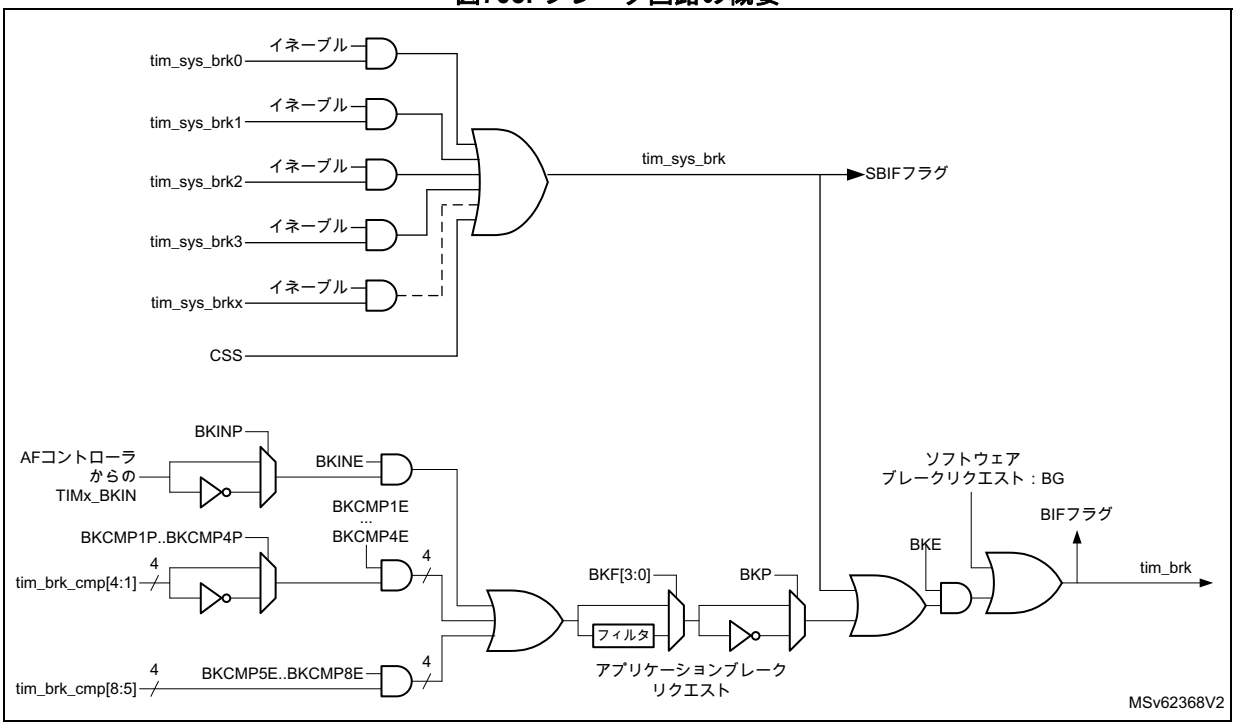
ブレークは、TIMx_AF1レジスタを使用して、個別に有効化でき、プログラム可能なエッジ検知を持つ複数のソースから生成できます。

ブレーク（tim_brk）チャネルのソースは以下のいずれかです。

- （GPIOオルタネート機能選択レジスタでの選択に従って）TIM_BKINピンの1つに接続された外部ソースで、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - tim_brk_cmpx 入力から着信するもの（製品固有の実装についてはセクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照）
  - システムブレークリクエストからtim_sys_brk入力へ着信するもの（製品固有の実装についてはセクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照）

ブレークイベントは、TIMx_EGRレジスタでBGビットを使用して、ソフトウェアによって生成することもできます。以下の図705のとおり、すべてのソースはタイマのtim_brk入力に入る前に論理和がとられます。

図705. ブレーク回路の概要



注意： **非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部PLLやCSSを使用）を使用する必要があります。**

ブレークが発生すると（ブレーク入力で選択されたレベル）、

- MOE ビットは非同期にクリアされ、出力は、非アクティブ状態またはアイドル状態になるか、GPIOへの制御が解除されます（OSSIビットで選択）。これは、MCUオシレータがオフの場合も同様です。
- 各出力チャネルは、MOE=0になったとき、TIMx_CR2レジスタのOISxビットでプログラミングされたレベルで駆動されます。OSSI=0の場合、タイマは出力の制御（GPIOによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずリセット状態の非アクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後にOISxおよびOISxNビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、tim_ocxとtim_ocxnを同時にアクティブレベルに駆動することはできません。MOEの再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約2 tim_ker_ckクロックサイクル）。
  - OSSI=0の場合、タイマはイネーブル出力（ハイインピーダンス状態を強制するGPIOによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxEまたはCCxNEビットのどちらかがハイになったときにハイになります。
- ブレーク状態フラグ（TIMx_SRレジスタのBIFビット）がセットされます。TIMx_DIERレジスタのBIEビットがセットされている場合は、割込みを生成できます。TIMx_DIERレジスタのBDEビットがセットされている場合、DMAリクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント UEV で再び自動的にセットされます。これを使用して、たとえば、レギュレーションを行うことができます。そうでない場合、MOEは、再び1が書き込まれるまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

注： **AOEビットが設定されている状態でCPUによりMOEがリセットされると、出力はアイドル状態になり、OSSI値に応じて非アクティブレベルまたはハイインピーダンスに強制されます。MOEビットとAOEビットの両方がCPUによってリセットされた場合、出力は無効状態になり、TIMx_CR2レジスタのOISxビットでプログラムされたレベルで駆動されます。**

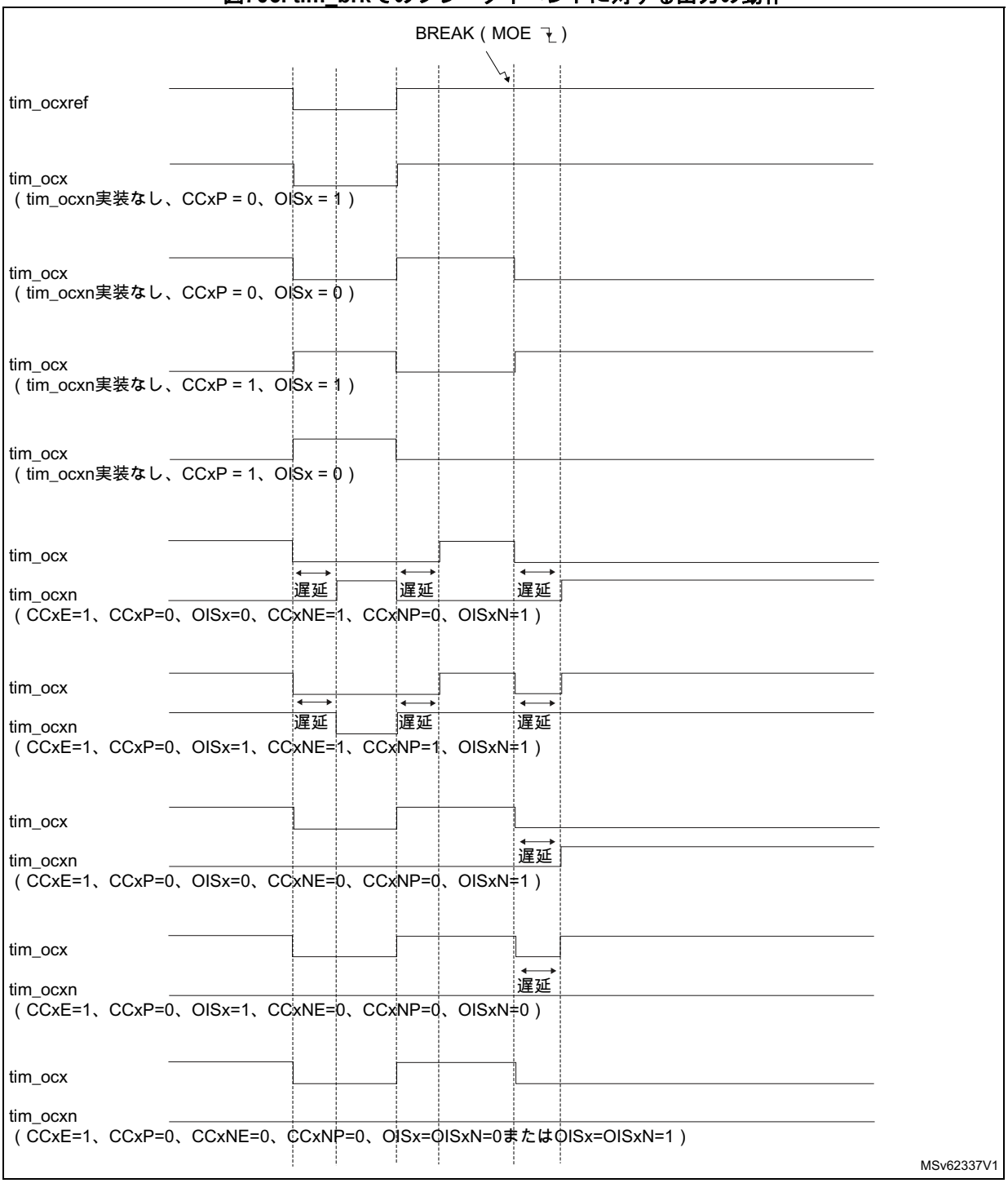
注： **ブレーク入力は、信号レベルに対して動作します。このため、ブレーク入力がアクティブな間は、MOEをセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグBIFをクリアできません。**

ブレークは、tim_brk入力によって生成でき、tim_brkはプログラミング可能な極性を持ち、TIMx_BDTRレジスタのBKEがイネーブルビットです。

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これは、いくつかのパラメータ（デッドタイムの長さ、tim_ocx/tim_ocxn極性、無効時の状態、OCxM構成、ブレークイネーブルと極性）の設定を固定するために使用されます。TIMx_BDTRレジスタのLOCKビットによって、保護を3つのレベルから選択することができます。セクション 56.8.14：TIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）を参照してください。LOCKビットは、MCUリセット後に1回だけ書き込むことができます。

図706に、ブレークに対する出力の動作例を示します。

図706. tim_brkでのブレークイベントに対する出力の動作



56.4.16 双方向ブレーク入力

TIM15/TIM16/TIM17 には、図707に示すとおり、双方向ブレークI/Oが搭載されています。

それらは下記のために使用されます。

- 入力および出力ステータスピンどちらにもなる単一のピンによる、外部のMCUやゲートドライバに障害を通知するために使用できる幅広いレベルのグローバルブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、単一のブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインソース

tim_brk入力は、TIMxBDTRレジスタのBKBIDビットを使用して双方向モードに設定されます。BKBIDプログラミングビットは、TIMxBDTRレジスタのLOCKビットを使用して読出し専用モードにロックできます（LOCKレベル1以上）。

双方向モードでは、I/Oをアクティブロー極性でオープンドレインモードに設定する必要があります（BKINPおよびBKPビットを使用）。システム（たとえばCSS）、内部ペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローにされます。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

（BGビットのセットによってトリガされる）ブレークソフトウェアイベントも、ブレークI/Oを強制的に0にして、タイマがブレーク状態に移行したことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BKE = 1）にのみ有効です。ソフトウェアブレークイベントがBKE = 0で生成された場合、出力は安全な状態になり、ブレークフラグはセットされますが、TIM_BKIN I/Oには効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRMビットが1にセットされると、ブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM ビットがセットされており、オープンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM出力の再開を防ぐことができます。
- 出力が有効になっている（MOEビットがセットされている）限り、BKDSRMビットでブレーク保護を解除できません（表579を参照）。

表579.ブレーク保護解除条件

MOE	BKBID	BKDSRM	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

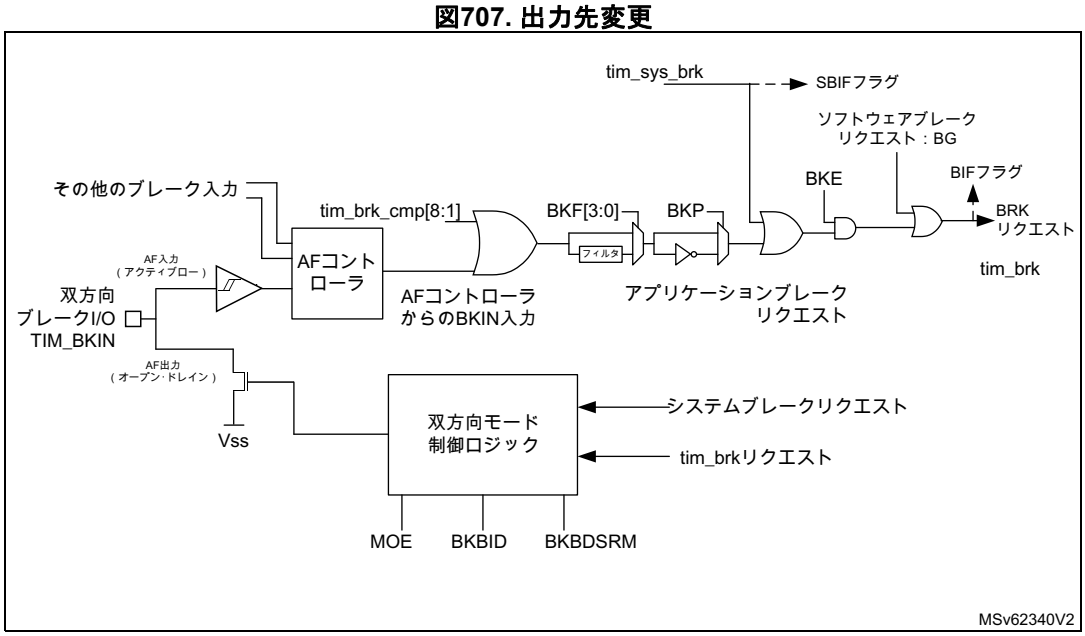
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレークイベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRMビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM ビットをハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOEビットをセットして、PWM出力を再び有効にできます。

図707. 出力先変更

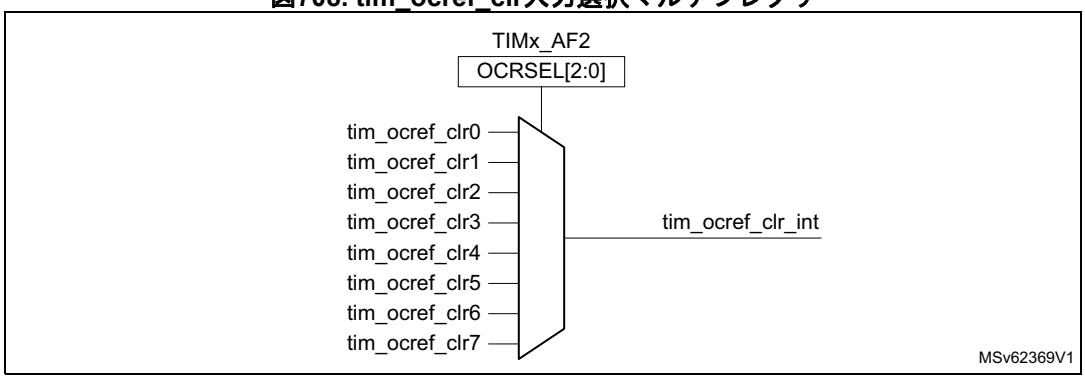


56.4.17 外部イベントによるtim_ocxref信号のクリア

特定のチャネルのtim_ocxref信号はtim_ocref_clr_int入力にハイレベルを適用するとクリアされます（対応するTIMx_CCMRxレジスタのOCxCEイネーブルビットを1にセットする）。tim_ocxrefは、次のPWMサイクルでの次のアクティブステートへの遷移まで、ローのまま残ります。この機能は、出力比較モードとPWMモードでのみ使用可能です。強制モードでは動作しません。

tim_ocref_clr_int入力は、以下の図708に示すように、複数の入力から選択できます。

図708. tim_ocref_clr入力選択マルチプレクサ



56.4.18 6ステップPWM生成

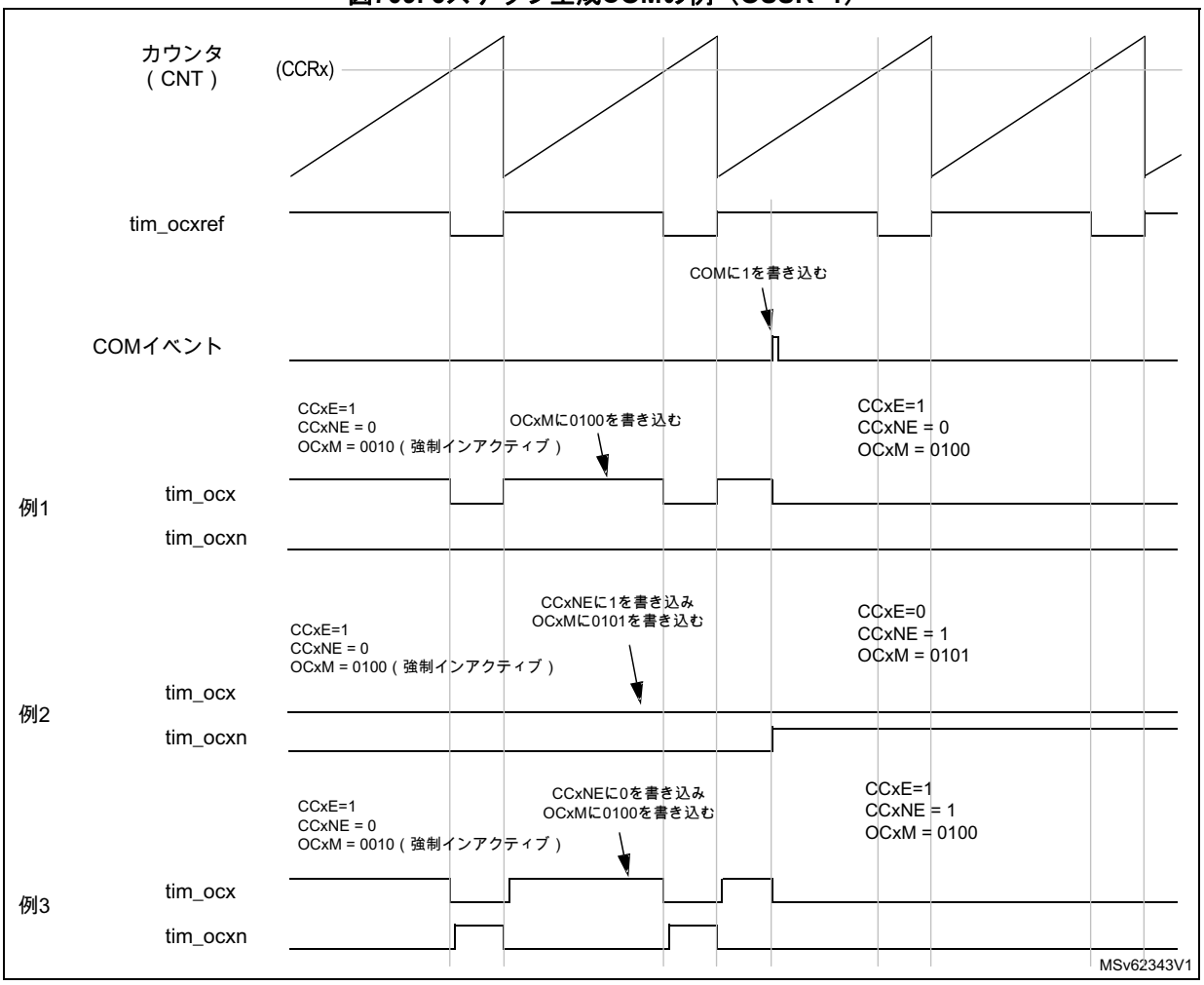
チャネルで相補出力が使用されているときには、OCxM、CCxE、およびCCxNEビットでプリロードビットが使用できます。プリロードビットは、COM転流イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COMは、TIMx_EGRレジスタのCOMビットをセットすることによってソフトウェアによって、またはハードウェアによって（tim_trgi立上がりエッジで）生成することができます。

フラグは、COMイベントが発生したときにセットされ（TIMｘ_SRレジスタのCOMIFビット）、これによって割込み（TIMx_DIERレジスタのCOMIEビットがセットされている場合）またはDMAリクエスト（TIMx_DIERレジスタのCOMDEビットがセットされている場合）を生成できます。

図709に、COMイベントが発生したときのtim_ocxとtim_ocxn出力の動作を、

3種類のプログラミング構成の例で示します。

図709. 6ステップ生成COMの例（OSSR=1）



56.4.19 ワンパルスモード

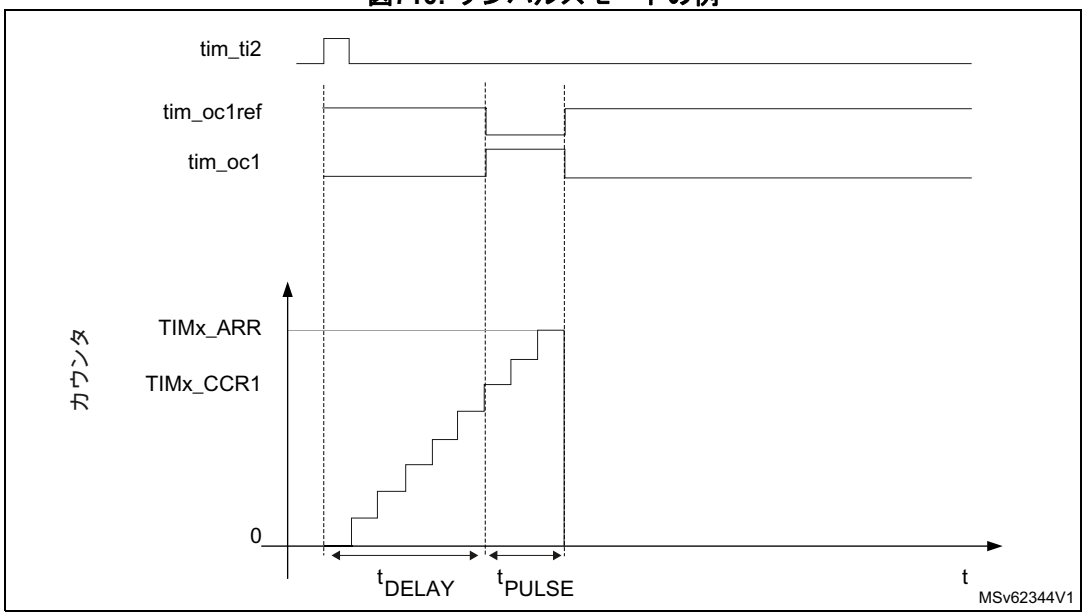
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたはPWMモードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1レジスタのOPMビットをセットします。これによって、カウンタは、次の更新イベントUEVで自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図710. ワンパルスモードの例



たとえば、tim_ti2入力ピンで立上がりエッジが検出されたときに、tim_oc1にパルス幅がt_{PULSE} の正のパルスを遅延時間t_{DELAY} 後に生成することもできます。

tim_ti2fp2をトリガ1として使用します。

1. TIMx_TISELレジスタのTI2SEL[3:0] ビットで、適切なtim_ti2_in[15:1] ソース（内部または外部）を選択します。
2. TIMx_CCMR1レジスタのCC2Sビットに”01”を書き込むことによって、tim_ti2fp2をtim_ti2に配置します。
3. tim_ti2fp2は、立上がりエッジを検出して、TIMx_CCERレジスタでCC2P=“0”とCC2NP =“0”を書き込みます。
4. tim_ti2fp2をスレーブモードコントローラのトリガ （tim_trgi） として構成するために、TIMx_SMCRレジスタのTS=“00110”を書き込みます。
5. tim_ti2fp2を使用してカウンタを開始します。このためには、TIMx_SMCRレジスタのSMSビットに“110”（トリガモード）を書き込みます。

OPM波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに0から1へ遷移し、カウンタが自動再ロード値に達したときに1から0へ遷移する波形を生成するとします。このためには、TIMx_CCMR1レジスタにOC1M=111を書き込むことによって、PWMモード2を有効にする必要があります。オプションで、TIMx_CCMR1レジスタにOC1PE='1' を書き込み、TIMx_CR1レジスタのARPEビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1レジスタに比較値を書き込み、TIMx_ARRレジスタに自動再ロード値を書き込む必要があります。次に、UGビットをセットすることによって更新を生成し、tim_ti2で外部トリガイベントを待つ必要があります。この例では、CC1Pに“0”を書き込みます。

必要なパルスは1つだけなので、TIMx_CR1レジスタのOPMビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。

特殊ケース：tim_ocx高速イネーブル

ワンパルスモードでは、tim_tix入力のエッジ検出によって、カウンタを有効にするCENビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRxレジスタのOCxFEビットをセットします。こうすると、tim_ocxref（およびtim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFEは、チャネルがPWM1またはPWM2モードに設定された場合のみ機能します。

56.4.20 再トリガ可能なワンパルスモード（TIM15のみ）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション56.4.19で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

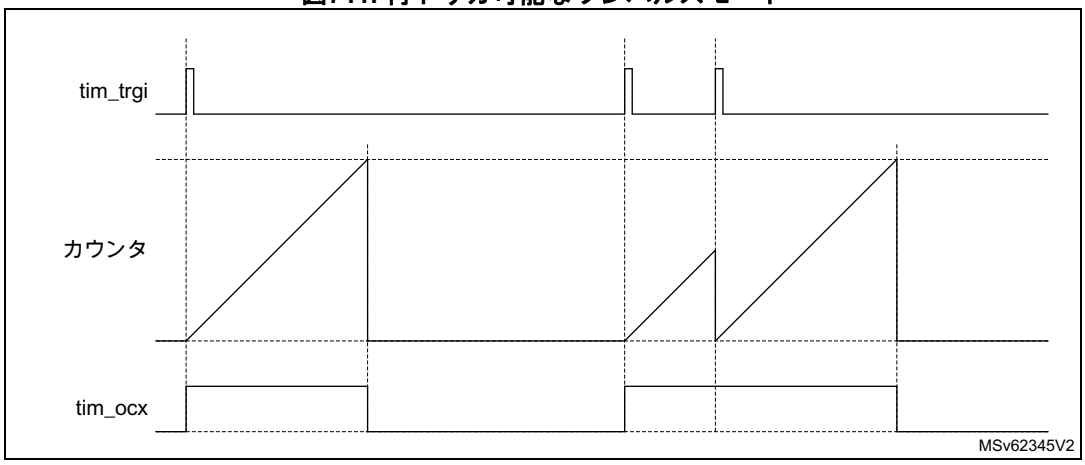
タイマはスレーブモードである必要があり、このときビットはTIMx_SMCRレジスタでSMS[3:0] =“1000”（リセットモードとトリガモードの組合せ）、および再トリガ可能なOPMモード1または2でOCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応するCCRxを0にセットする必要があります（ARRレジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRxはARR以上である必要があります。

注： **OCxM[3:0] およびSMS[3:0]ビットフィールドは互換性を確保するために2つのパーツに分割され、最上位ビットと3つの最下位ビットとは隣接していません。**

このモードをセンターアラインPWMモードと組合せて使用することはできません。TIMx_CR1では、CMS[1:0] = 00にする必要があります。

図711. 再トリガ可能なワンパルスモード



56.4.21 UIFビットの再配置

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これは、UIFCPYフラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取るために使用されます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

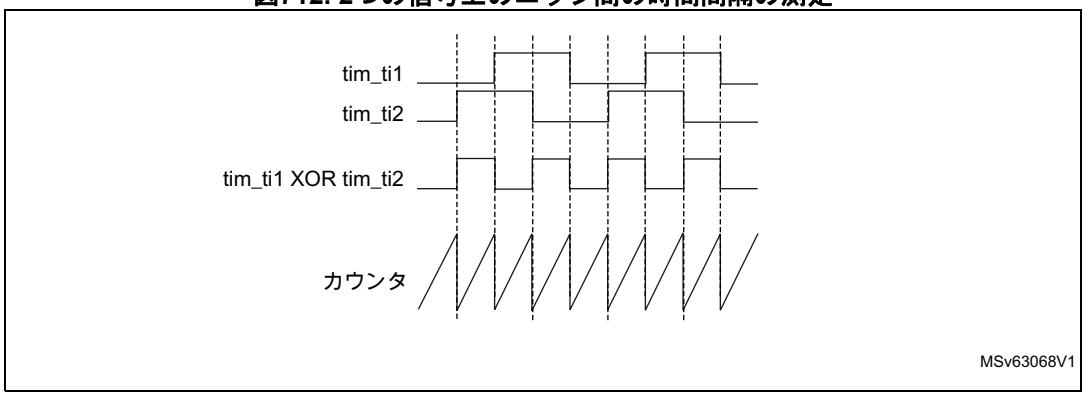
UIFとUIFCPYフラグのアサートの間には、遅延はありません。

56.4.22 タイマ入力XOR機能（TIM15のみ）

TIMx_CR2レジスタのTI1Sビットによって、チャネル1の入力フィルタを2つの入力ピンtim_ti1およびtim_ti2を結合するXORゲートの出力に接続できます。

XOR出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図712に示すように、2つの入力信号上のエッジ間の間隔を測定するのに便利です。

図712. 2つの信号上のエッジ間の時間間隔の測定



56.4.23 外部トリガ同期（TIM15のみ）

TIMタイマは、タイマの同期または連結のために、内部で互いにリンクされます。

TIM15タイマは、リセットモード、ゲートモード、トリガモード、リセット + トリガモード、およびゲート + リセットモードで、外部トリガに同期できます。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1レジスタのURSビットがローの場合は、更新イベントUEVが生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

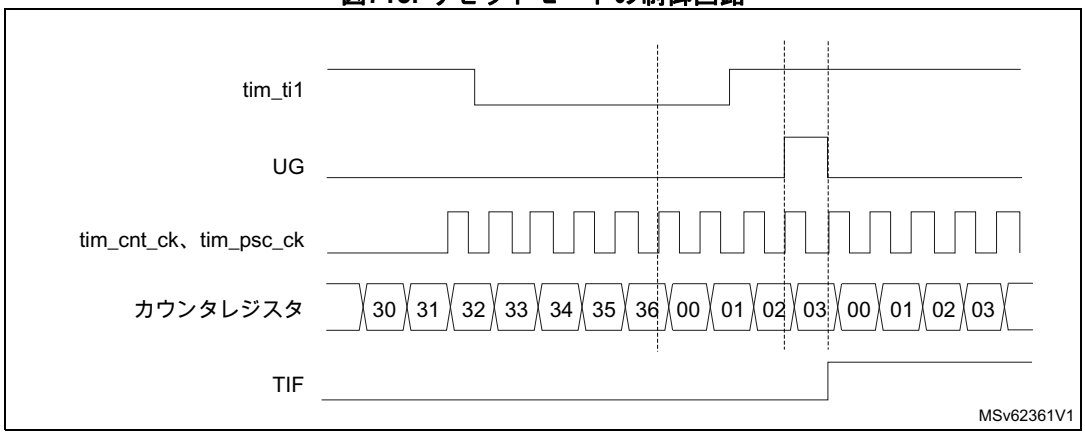
次の例では、tim_ti1入力の立上がりエッジに応じて、アップカウンタがクリアされます。

1. tim_ti1の立上がりエッジを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01）。TIMx_CCERレジスタでCC1P=0とCC1NP = 0を書き込んで、極性を有効にします（そして、立上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
3. TIMx_CR1レジスタにCEN=1を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1の立上がりエッジまで通常の動作を行います。tim_ti1が立ち上がると、カウンタはクリアされ、0からリスタートします。同時に、トリガフラグがセットされ（TIMx_SRレジスタのTIFビット）、有効な場合は割込みリクエストまたはDMAリクエストを送信できます（TIMx_DIERレジスタのTIEおよびTDEビットに依存）。

次の図は、自動再ロードレジスタTIMx_ARR=0x36の場合の動作を示します。tim_ti1の立上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1入力の同期回路によるものです。

図713. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

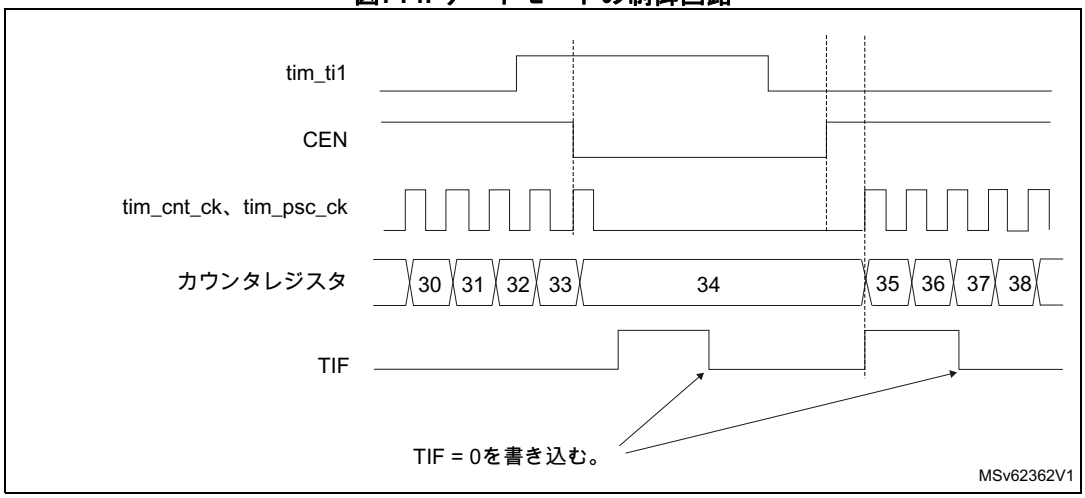
次の例では、アップカウンタはtim_ti1入力がローのときだけカウントします。

1. tim_ti1のローレベルを検出するように、チャネル1を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1Sビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1レジスタのCC1S=01bit）。TIMx_CCERレジスタでCC1P = 1とCC1NP = 0を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタにSMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCRレジスタにTS=00101を書き込むことによって、入力ソースとしてtim_ti1を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1がローになると内部クロックでカウントを開始して、tim_ti1がハイになると停止します。TIMx_SRレジスタのTIFフラグは、カウンタの開始時と停止時にセットされます。

tim_ti1の立上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1入力の再同期回路によるものです。

図714. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

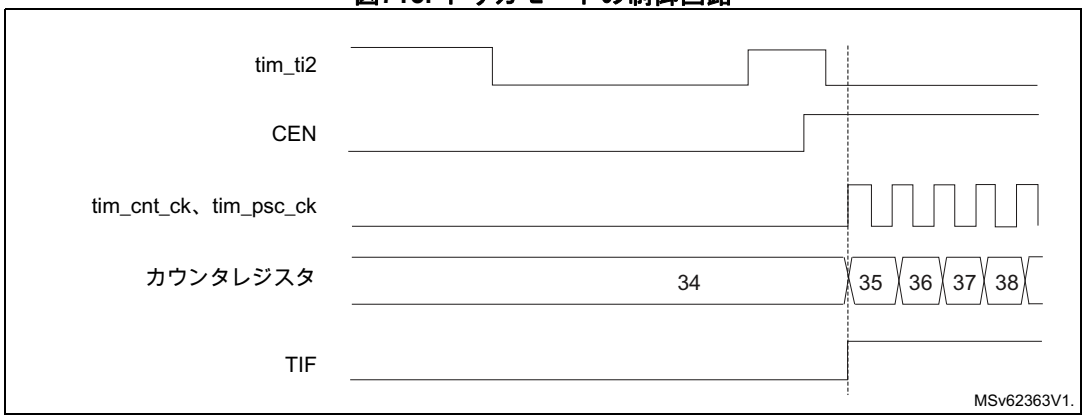
次の例では、アップカウンタは、tim_ti2入力の立上がりエッジに応じて開始します。

1. tim_ti2の立上がりエッジを検出するように、チャネル2を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2Sビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1レジスタのCC2S=01）。TIMx_CCERレジスタでCC2Pに「1」、CC2NPに「0」を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCRレジスタにTS=00110を書き込むことによって、入力ソースとしてtim_ti2を選択します。

tim_ti2で立上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIFフラグがセットされます。

tim_ti2の立上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2入力の再同期回路によるものです。

図715. トリガモードの制御回路



実行時更新のためのスレーブモード選択プリロード

SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCRレジスタのSMSPEイネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、カウンタがオーバーフローするときに発生する更新イベント（UEV）です。

56.4.24 スレーブモード - リセットモードとトリガモードの組合せ（TIM15のみ）

この場合、選択されたトリガ入力（tim_trgi）の立上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

56.4.25 スレーブモード - リセットモードとゲートモードの組合せ（TIM15のみ）

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードは、範囲外のPWM信号（最大期待値を超えるデューティサイクル）を検出するために使用されます。

56.4.26 タイマ同期（TIM15のみ）

タイマの同期や連携した動作のために、TIMxタイマを内部で相互リンクすることができます。詳細については、セクション 55.4.23：タイマの同期を参照してください。

注： **tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

56.4.27 タイマ出力を他のタイマのトリガとして使用（TIM16/TIM17 のみ）

1チャンネルのみのタイマは、マスタモードを備えていません。しかし、OC1出力信号は、他のいくつかのタイマ（本書の他のセクションで説明されているタイマを含む）のトリガとして使用できます。どのタイマがスレーブとして使用できるかを知るには、デバイスのいずれかのタイマの「TIMx内部トリガ接続」テーブルを確認してください。

OC1信号のパルス幅は、スレーブタイマが確実にトリガを検出できるように、転送先タイマの少なくとも2クロックサイクルになるようにプログラムする必要があります。

たとえば、転送先タイマのCK_INTクロックが転送元のタイマより4倍遅い場合、OC1のパルス幅は8クロックサイクルでなければなりません。

56.4.28 ADCトリガ（TIM15のみ）

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。

注： **tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

56.4.29 DMAバーストモード

TIMxタイマには、1つのイベントで多重DMAリクエストを生成する機能があります。主な目的は、いくつかのタイマレジスタをソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMAコントローラの転送先は一意で、仮想レジスタTIMx_DMARを示している必要があります。特定のタイマイベントで、タイマは一連のDMAリクエスト（バースト）を開始します。TIMx_DMARレジスタへの各書込みは、実際にタイマレジスタの1つにリダイレクトされます。

TIMx_DCR レジスタのDBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタのDBA[4:0] ビットは、DMA 転送のDMA ベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し／書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

TIMx_DCRレジスタのDBSS[3:0] ビットでは、DMAバースト転送をトリガする割込みソースを定義します（詳細はセクション 56.8.20：TIMx DMA制御レジスタ（TIMx_DCR）（x = 16～17）を参照）。

たとえば、更新イベント時にCCRxレジスタ値（x = 2、3、4）の内容を更新するためにタイマDMAバースト機能を使用できます。このとき、DMAはCCRxレジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応するDMAチャネルを次のように設定します。
   - DMAチャネルペリフェラルアドレスを、DMARレジスタアドレスとします。
   - DMAチャネルメモリアドレスを、DMAによってCCRxレジスタに転送されるデータを格納するRAM内のバッファアドレスとします。
   - 転送データ数 = 3とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBAとDBLのビットフィールドを次のように設定することによって、DCRレジスタを設定します。
   DBL = 3回転送、DBA = 0xE、およびDBSS = 1。
3. TIMx更新DMAリクエストを有効にします（DIERレジスタのUDEビットをセット）。
4. TIMxを有効化
5. DMAチャネルを有効化注：

この例は、各CCRxレジスタが1回更新される場合です。たとえば、各CCRxレジスタが2回更新される場合は、転送データ数は6でなければなりません。データ1、データ2、データ3、データ4、データ5、データ6を格納するRAMのバッファを例にします。データは、CCRxレジスタに次のように転送されます。最初の更新DMAリクエストでデータ1がCCR2に転送され、データ2はCCR3に、データ3はCCR4にそれぞれ転送され、2番目の更新DMAリクエストでデータ4がCCR2に、データ5がCCR3に、データ6がCCR4にそれぞれ転送されます。

注： **null値を予約済みレジスタに書き込むことができます。**

56.4.30 TIM15/TIM16/TIM17DMAリクエスト

表 580に示すように、TIM15/TIM16/TIM17 はDMAリクエストを生成できます。

表580.DMAリクエスト

DMAリクエスト信号	DMAリクエストあり	有効制御ビット
tim_upd_dma	更新	UDE
tim_cc1_dma	キャプチャ／比較1	CC1DE
tim_com_dma(1)	転流（COM）	COMDE
tim_trg_dma (1)	トリガ	TDE

1. TIM15でのみ使用できます。

56.4.31 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M33 コアは停止状態）、TIMxカウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

安全のため、カウンタが停止すると、出力は無効になります（MOEビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的に非アクティブ状態にするか（OSSIビット = 1）、GPIOコントローラで制御することができます（OSSIビット = 0）。

詳細については、デバッグセクションを参照してください。

56.5 TIM15/TIM16/TIM17 低消費電力モード

表581.低消費電力モードが TIM15/TIM16/TIM17 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスはSLEEPモードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

56.6 TIM15/TIM16/TIM17 割込み

表582 に示すように、TIM15/TIM16/TIM17 は複数の割込みを生成できます。

表582.割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
TIM	更新	UIF	UIE	UIFに0を書き込む	あり	不可
	キャプチャ／比較1	CC1IF	CC1IE	CC1IFに0を書き込む	あり	不可
	キャプチャ／比較2(1)	CC2IF	CC2IE	CC2IFに0を書き込む	あり	不可
	転流（COM）	COMIF	COMIE	COMIFに0を書き込む	あり	不可
	トリガ (1)	TIF	TIE	TIFに0を書き込む	あり	不可
	ブレーク	BIF	BIE	BIFに0を書き込む	あり	不可

1. TIM15でのみ使用できます。

56.7 TIM15レジスタ

レジスタの説明で使用されている略語のリストについては、セクション1.2を参照してください。

56.7.1 TIM15制御レジスタ1 （TIM15_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw				rw	rw	rw	rw

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 DITHEN：ディザリングイネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHENビットは、CENビットがリセットされているときのみ変更できます。

ビット11 **UIFREMAP**：UIFステータスビットの再配置

0：再配置なし。UIFステータスビットはTIM15_CNTレジスタのビット31にコピーされません。

1：再配置は有効です。UIFステータスビットはTIM15_CNTレジスタのビット31にコピーされます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}= t_{tim_ker_ck}$

01：$t_{DTS}= 2*t_{tim_ker_ck}$

10：$t_{DTS}= 4*t_{tim_ker_ck}$

11：予約済み

ビット7 **ARPE**：自動再ロードプリロードイネーブル

0：TIM15_ARRレジスタはバッファされません。

1：TIM15_ARRレジスタはバッファされます。

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **OPM**：ワンパルスモード。

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CENビットをクリア）。

ビット2 **URS**：更新リクエストソース

このビットは、UEVイベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UGビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みを生成します（有効な場合）。

ビット1 **UDIS**：更新ディセーブル

このビットは、UEVイベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEVは有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UGビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UGビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット0 **CEN**：カウンタイネーブル

0：カウンタは無効です。
1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CENビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的にCENビットをセットできます。

56.7.2 TIM15制御レジスタ2（TIM15_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
					rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **OIS2**：出力アイドル状態2（tim_oc2出力）

0：MOE = 0のとき、tim_oc2=0
1：MOE = 0のとき、tim_oc2=1

注： このビットは、LOCKレベル1、2、または3がプログラムされている場合（TIM15_BKRレジスタのLOCKビット）、変更できません。

ビット9 **OIS1N**：出力アイドル状態1（tim_oc1n出力）

0：MOE=0のとき、デッドタイム後にtim_oc1n=0
1：MOE=0のとき、デッドタイム後にtim_oc1n=1

注： このビットは、LOCKレベル1、2、または3がプログラムされている場合（TIM15_BKRレジスタのLOCKビット）、変更できません。

ビット8 **OIS1**：出力アイドル状態1（tim_oc1出力）

0：MOE=0のとき、デッドタイム後にtim_oc1=0
1：MOE=0のとき、デッドタイム後にtim_oc1=1

注： このビットは、LOCKレベル1、2、または3がプログラムされている場合（TIM15_BKRレジスタのLOCKビット）、変更できません。

ビット7 **TI1S**：tim_ti1選択

0：tim_ti1_in[15:0] マルチプレクサ出力がtim_ti1入力に接続されます。
1：tim_ti1_in[15..0] およびtim_ti2_in[15..0] マルチプレクサ出力がtim_ti1入力に接続されます（XOR接続）。

ビット6:4 **MMS[2:0]**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（tim_trgo）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット** - TIM15_EGRレジスタのUGビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号CNT_ENがトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、tim_trgoには遅延が存在します（TIM15_SMCRレジスタのMSMビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると（tim_trgo）、CC1IFフラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。

100：**比較** - tim_oc1refc信号がトリガ出力（tim_trgo）として使用されます。

101：**比較** - tim_oc2refc信号がトリガ出力（tim_trgo）として使用されます。

ビット3 **CCDS**：キャプチャ／比較DMA選択

0：CCx DMAリクエストは、CCxイベントが発生すると送信されます。

1：CCx DMAリクエストは、更新イベントが発生すると送信されます。

ビット2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMGビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMGビットをセットすることによって、またはtim_trgiの立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、およびOCxMビットはプリロードされません。

1：CCxE、CCxNE、およびOCxMビットがプリロードされます。書込みの後、これらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUSビットに応じて、COMGビットがセットまたはtim_trgiで立ち上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

56.7.3 TIM15スレーブモード制御レジスタ（TIM15_SMCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
							rw			rw	rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
								rw	rw	rw	rw		rw	rw	rw

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **SMSPE**：SMSプリロードイネーブル

このビットは、SMS[3:0] ビットフィールドをプリロードするかどうかを選択します。

0：SMS[3:0] ビットフィールドはプリロードされません。

1：SMS[3:0] プリロードは有効です。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **MSM**：マスタ/スレーブモード

0：影響なし。

1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgo を通じて）完全な同期状態が得られるまで遅延されます。これは、1つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット21、20、6、5、**TS[4:0]**：トリガ選択

4 このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000 : 内部トリガ0（tim_itr0）
00001 : 内部トリガ1（tim_itr1）
00010 : 内部トリガ2（tim_itr2）
00011 : 内部トリガ3（tim_itr3）
00100 : tim_ti1エッジ検出回路（tim_ti1f_ed）
00101 : フィルタタイマ入力1（tim_ti1fp1）
00110 : フィルタタイマ入力2（tim_ti2fp2）
00111 : 予約済み
01000 : 内部トリガ4（tim_itr4）
01001 : 内部トリガ5（tim_itr5）
01010 : 内部トリガ6（tim_itr6）
01011 : 内部トリガ7（tim_itr7）
01100 : 内部トリガ8（tim_itr8）
01101 : 内部トリガ9（tim_itr9）
01110 : 内部トリガ10（tim_itr10）
10000 : 内部トリガ12（tim_itr12）
10001 : 内部トリガ13（tim_itr13）
10010 : 内部トリガ14（tim_itr14）
10011 : 内部トリガ15（tim_itr15）
その他 : 予約済み
各タイマでのtim_itrxの詳細については、セクション 56.4.2 : TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

注： 電圧変化時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（たとえば、SMS = 000のとき）にのみ変更しなければなりません。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済み

0010：予約済み

0011：予約済み

0100：リセットモード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ･モード - カウンタは、トリガtim_trgiの立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲートモードとリセットモードの組み合わせ - カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

その他：予約済み

注： トリガ入力としてtim_ti1f_edが選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_edはtim_ti1fの変化ごとに1パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

56.7.4 TIM15 DMA／割込み有効レジスタ（TIM15_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	rw	rw				rw	rw	rw	rw	rw			rw	rw	rw

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **TDE**：トリガDMAリクエストイネーブル

0：トリガDMAリクエストは無効です。

1：トリガDMAリクエストは有効です。

ビット13 **COMDE**：COM DMAリクエストイネーブル

0：COM DMAリクエストは無効です。

1：COM DMAリクエストは有効です。

ビット12:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC1DE**：キャプチャ／比較1 DMAリクエストイネーブル

0：CC1 DMAリクエストは無効です。

1：CC1 DMAリクエストは有効です。

ビット8 **UDE**：更新DMAリクエストイネーブル
0：更新DMAリクエストは無効です。
1：更新DMAリクエストは有効です。

ビット7 **BIE**：ブレーク割込み有効
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット5 **COMIE**：COM割込み有効
0：COM割込みは無効です。
1：COM割込みは有効です。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CC2IE**：キャプチャ／比較2割込みイネーブル
0：CC2割込みは無効です。
1：CC2割込みは有効です。

ビット1 **CC1IE**：キャプチャ／比較1割込みイネーブル
0：CC1割込みは無効です。
1：CC1割込みは有効です。

ビット0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

56.7.5 TIM15ステータスレジスタ（TIM15_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0		rc_w0	rc_w0	rc_w0			rc_w0	rc_w0	rc_w0

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **CC2OF**：キャプチャ／比較2オーバーキャプチャフラグ
CC1OFの説明を参照してください。

ビット9 **CC1OF**：キャプチャ／比較1オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IFフラグがすでにセットされているときに、カウンタの値がTIM15_CCR1レジスタにキャプチャされました。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **BIF**：ブレーク割込みフラグ
このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。
0：ブレークイベントは発生していません。
1：ブレーク入力でアクティブレベルが検出されました。

ビット6 **TIF**：トリガ割込みフラグ

このフラグは、TRGトリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi入力でアクティブエッジが検出されたとき、またはゲートモードが選択されている場合は、両方のエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット5 **COMIF**：COM割込みフラグ

このフラグは、COMイベント時にハードウェアによってセットされます（キャプチャ/比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COMイベントは発生していません。

1：COM割込みがペンディング中です。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CC2IF**：キャプチャ／比較2割込みフラグ

CC1IFの説明を参照してください。

ビット1 **CC1IF**：キャプチャ／比較1割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、またはTIMx_CCR1レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネルCC1が出力として設定されている場合：カウンタTIMx_CNTの内容がTIMx_CCR1レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1の内容がTIMx_ARRの内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）にCC1IFビットはハイになります。センターアラインモードでのフラグの設定には3つのオプションがあります。詳細については、TIMx_CR1レジスタのCMSビットを参照してください。

CC1チャネルが入力として設定されている場合：このビットは、TIMx_CCR1レジスタでカウンタ値がキャプチャされた（TIMx_CCERのCC1PおよびCC1NPビット設定で定義されたエッジ検知に従って、IC1でエッジが検出された）ときに設定されます。

ビット0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰返しカウンタ値に関するオーバーフロー（繰返しカウンタ=0の場合の更新）、およびTIM15_CR1レジスタのUDIS=0の場合。

–TIM15_CR1レジスタのURS=0かつUDIS=0の場合に、TIM15_EGRレジスタのUGビットを使用して、CNTがソフトウェアによって再初期化されたとき。

–TIM15_CR1レジスタのURS=0かつUDIS=0であり、トリガイベントによってCNTが再初期化されたとき（セクション 56.7.3：TIM15スレーブモード制御レジスタ（TIM15_SMCR）を参照）。

56.7.6 TIM15イベント生成レジスタ（TIM15_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
								w	w	rw			w	w	w

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIM15_SRレジスタのTIFフラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPCビットがセットされているときには、CCxE、CCxNE、およびOCxMビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット4:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CC2G**：キャプチャ／比較2生成

CC1Gの説明を参照してください。

ビット1 **CC1G**：キャプチャ／比較1生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル1でキャプチャ／比較イベントが生成されます。

CC1チャネルが出力として設定されている場合：

CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。

CC1チャネルが入力として設定されている場合：

カウンタの現在値がTIM15_CCR1レジスタにキャプチャされます。CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。CC1IFフラグがすでにハイの場合、CC1OFフラグがセットされます。

ビット0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

56.7.7 TIM15キャプチャ／比較モードレジスタ1（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 **IC2F[3:0]**：入力キャプチャ 2フィルタ

ビット11:10 **IC2PSC[1:0]**：入力キャプチャ 2プリスケーラ

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2にマッピングされます。

10：CC2チャネルは入力として設定され、tim_ic1はtim_ti2に配置されます。

11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIM15_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2Sビットは、チャネルがオフ（TIM15_CCERレジスタのCC2E="0"）のときのみ、書込みができます。

ビット7:4 **IC1F[3:0]**：入力キャプチャ 1フィルタ

このビットフィールドは、tim_ti1入力をサンプリングする周波数と、tim_ti1に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS} で行われます。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット3:2 **IC1PSC[1:0]**：入力キャプチャ 1プリスケーラ

このビットフィールドは、CC1入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E="0"（TIM15_CCERレジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2イベントごとに行われます。

10：キャプチャは、4イベントごとに行われます。

11：キャプチャは、8イベントごとに行われます。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。

10：CC1チャネルは入力として設定され、tim_ic1はtim_ti2にマッピングされます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_trcにマッピングされます。このモードは、TSビット（TIM15_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIM15_CCERレジスタのCC1E="0"）のときのみ、書込みができます。

56.7.8 TIM15キャプチャ／比較モード･レジスタ1 [オルタネート]（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		OC1 CE	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット23:17 予約済みであり、リセット値に保持する必要があります。

ビット15 **OC2CE**：出力比較2クリアイネーブル

ビット24、14:12 **OC2M[3:0]**：出力比較2モード

ビット11 **OC2PE**出力比較2プリロードイネーブル

ビット10 **OC2FE**：出力比較2高速イネーブル

ビット9:8 **CC2S[1:0]**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2チャネルは出力として設定されます。

01：CC2チャネルは入力として設定され、tim_ic2はtim_ti2に配置されます。

10：C2チャネルは入力として設定され、tim_ic2はtim_ti1にマッピングされます。

11：CC2チャネルは入力として設定され、tim_ic2はtim_trcにマッピングされます。このモードは、TSビット（TIM15_SMCRレジスタ）を通じて内部トリガ入力が選択されている場合のみ機能します。

注： CC2Sビットは、チャネルがオフ（TIM15_CCERレジスタのCC2E="0"）のときのみ、書込みができます。

ビット7 **OC1CE**：出力比較1クリアイネーブル

0：tim_oc1refはtim_ocref_clr_int入力の影響を受けません。

1：tim_oc1refはtim_ocref_clr_int入力のハイレベルが検出されるとクリアされます。

ビット16、6:4 **OC1M[3:0]**：出力比較1モード

これらのビットは、tim_oc1およびtim_oc1nが導き出される出力基準信号tim_oc1refの動作を定義します。tim_oc1refはアクティブハイですが、tim_oc1およびtim_oc1nのアクティブレベルはCC1PおよびCC1NPビットに依存します。

0000：停止 - 出力比較レジスタTIM15_CCR1とカウンタTIM15_CNTとの間の比較結果は出力に影響しません。このモードは、タイマをソフトウェアタイムベースとして使用する場合に使用できます。タイマ動作中に停止モードが有効になると、出力は停止状態になる前の状態（アクティブまたは非アクティブ）を維持します。

0001：一致時にチャネル1をアクティブレベルに設定します。tim_oc1ref信号は、カウンタTIM15_CNTがキャプチャ／比較レジスタ1（TIM15_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。tim_oc1ref 信号は、カウンタTIM15_CNTがキャプチャ／比較レジスタ1（TIM15_CCR1）と一致したときに、強制的に低になります。

0011：反転 - TIM15_CNT = TIM15_CCR1のとき、tim_oc1refは反転します。

0100：強制非アクティブレベル - tim_oc1refは強制的にローになります。

0101：強制アクティブレベル - tim_oc1refは強制的にハイになります。

0110：PWMモード1 - チャネル1は、TIM15_CNT < TIM15_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。

0111：PWMモード2 - チャネル1は、TIM15_CNT < TIM15_CCR1の場合は非アクティブに、そうでない場合はアクティブになります。

1000：再トリガ可能な OPMモード1 - アップカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルは非アクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再び非アクティブになります。

1001：再トリガ可能な OPMモード2 - アップカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルは非アクティブです。その後、PWMモード2と同様に比較が行われ、チャネルは次の更新時に再び非アクティブになります。ダウンカウントモードでは、tim_trgi信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWMモード1と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み

1011：予約済み

1100：組み合わせPWMモード1 - tim_oc1refは、PWMモード1と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ORです。

1101：組み合わせPWMモード2 - tim_oc1refは、PWMモード2と同様に挙動します。tim_oc1refcは、tim_oc1refとtim_oc2refとの論理ANDです。

1110：予約済み

1111：予約済み

注：　これらのビットは、LOCKレベル3がプログラムされていて（TIM15_BDTRレジスタのLOCKビット）、CC1S="00"（チャネルは出力に設定）のときには、変更できません。

PWMモードでは、比較結果が変化したとき、出力比較モードが「停止」モードから「PWM」モードに切り替わったとき、および出力比較モードが「強制アクティブ／非アクティブ」モードから「PWM」モードに切り替わったときに、OCREFのレベルが変化します。

相補出力を持つチャネルでは、このビットはプリロードされます。TIM15_CR2レジスタのCCPCビットがセットされた場合、OC1Mアクティブビットは、COMが生成されたときにのみプリロードから新しい値をとります。

ビット3 **OC1PE**：出力比較1プリロードイネーブル

0 :TIM15_CCR1のプリロードレジスタは無効です。TIM15_CCR1は、いつでも書込み可能であり、新しい値はただちに有効になります。

1 :TIM15_CCR1のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIM15_CCR1プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCKレベル3がプログラムされていて（TIM15_BDTRレジスタのLOCKビット）、CC1S="00"（チャネルは出力に設定）のときには、変更できません。

ビット2 **OC1FE**：出力比較1高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1パルスモード（TIMx_CR1レジスタのOPMビットのセット）で使用する必要があります。

0：CC1の動作は、トリガがオンのときでも、通常、カウンタとCCR1の値に依存します。トリガ入力のエッジ発生からCC1出力が有効になるまでの最小遅延は、5クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1出力に対して、比較一致のように働きます。このため、tim_ocxは、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1出力を有効にするまでの遅延は、3クロックサイクルに短縮されます。OCFEは、チャネルがPWM1またはPWM2モードに設定されている場合のみ機能します。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1に配置されます。

10：CC1チャネルは入力として設定され、tim_ic2. はtim_ti1に配置されます。

11：CC1チャネルは入力として設定され、tim_ic1はtim_ti1に配置されます。このモードは、TSビット（TIM15_SMCRレジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1Sビットは、チャネルがオフ（TIM15_CCERレジスタのCC1E="0"）のときのみ、書込みができます。

56.7.9 TIM15キャプチャ／比較有効レジスタ（TIM15_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
								rw		rw	rw	rw	rw	rw	rw

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **CC2NP**：キャプチャ／比較2相補出力極性
CC1NPの説明を参照してください。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **CC2P**：キャプチャ／比較2出力極性
CC1Pの説明を参照してください。

ビット4 **CC2E**：キャプチャ／比較2出力イネーブル
CC1Eの説明を参照してください。

ビット3 **CC1NP**：キャプチャ／比較1相補出力極性

CC1チャネルが出力として設定されている場合：

0：tim_oc1nはアクティブハイです。

1：tim_oc1nはアクティブローです。

CC1チャネルが入力として設定されている場合：

このビットは、tim_ti1fp1とtim_ti2fp1の極性を定義するためにCC1Pと組合せて使用されます。CC1Pの説明を参照してください。

注： このビットは、LOCKレベル2または3がプログラムされ（TIM15_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力として設定）になった直後に書込みできなくなります。

相補出力を持つチャネルでは、このビットはプリロードされます。TIM15_CR2レジスタのCCPCビットがセットされた場合、CC1NPアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット2 **CC1NE**：キャプチャ／比較1相補出力イネーブル

0：オフ - tim_oc1nはアクティブではありません。tim_oc1nのレベルは、MOE、OSSI、OSSR、OIS1、OIS1NおよびCC1Eビットによって決まります。

1：オン - tim_oc1n信号は、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1Eビットにより、対応する出力ピンに出力されます。

ビット1 **CC1P**：キャプチャ／比較1出力極性

CC1チャネルが出力として設定されている場合：

0：OC1はアクティブハイ（出力モード）／エッジ検知を選択（入力モード、下記を参照）

1：OC1はアクティブロー（出力モード）／エッジ検知を選択（入力モード、下記を参照）

CC1チャネルが入力として設定されているときは、CC1NP/CC1Pの両ビットは、トリガまたはキャプチャ動作に対するTI1FP1とTI2FP1のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立上がりエッジ。この回路はTIxFP1の立上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立下がりエッジ。この回路はTIxFP1の立下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1が反転されます（ゲートモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立上がりエッジと立下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCKレベル2または3がプログラムされた（TIM15_BDTRレジスタのLOCKビット）直後は書込みできません。

相補出力を持つチャネルでは、このビットはプリロードされます。TIM15_CR2レジスタのCCPCビットがセットされた場合、CC1Pアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット0 **CC1E**：キャプチャ／比較1出力イネーブル

0：キャプチャモードは無効／OC1は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1信号は、対応する出力ピンに出力されます。

CC1チャネルが出力として設定されている場合、OC1レベルは、CC1Eビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1NEビットに依存します。詳細については、表583を参照してください。

表583.ブレーク機能を持つ相補tim_ocxおよびtim_ocxnチャネルの出力制御ビット（TIM15）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	tim_ocx出力状態	tim_ocxn出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0 tim_ocxn=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0	tim_ocxref + 極性 tim_ocxn = tim_ocxref XOR CCxNP
		0	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocxn=0
		X	1	1	tim_ocxref + 極性 + デッドタイム	tim_ocxrefに対する相補（OCREFではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（非アクティブ状態で出力有効） tim_ocx=CCxP	tim_ocxref + 極性 tim_ocxn=tim_ocxref XOR CCxNP
		1	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	オフ状態（非アクティブ状態で出力有効） tim_ocxn=CCxNP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイインピーダンス）	
	1		0	0		
			0	1	オフ状態（非アクティブ状態で出力有効） 非同期に：tim_ocx=CCxP、tim_ocxn=CCxNP クロックが存在する場合：デッドタイム後、tim_ocx=OISxおよびtim_ocxn=OISxN。ただし、OISxとOISxNは、アクティブ状態におけるtim_ocxとtim_ocxnの両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIOコントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、およびCCxNPビットはクリアされたままでなければなりません。

注： 相補 tim_ocx および tim_ocxn チャネルに接続されている外部入出力ピンの状態は、tim_ocx および tim_ocxnチャネルの状態と、GPIO制御 および オルタネート機能選択レジスタに依存します。

56.7.10 TIM15カウンタ（TIM15_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UIFCPY**：UIFコピー
このビットはTIM15_ISRレジスタのUIFビットのコピーで読出し専用です。

ビット30:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）
このレジスタは、カウンタ値を保持します。
ディザリングモード（DITHEN = 1）
このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分はありません。

56.7.11 TIM15プリスケーラ（TIM15_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（$f_{tim_cnt_ck}$）は$f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。
PSCは、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIM15_EGRレジスタのUGビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

56.7.12 TIM15の自動再ロードレジスタ（TIM15_ARR）

アドレスオフセット：0x2C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **ARR[19:0]**：自動再ロード値

ARRは、実際の自動再ロードレジスタにロードされる値です。

ARRの更新と動作の詳細については、セクション 56.4.3: 2329ページのタイムベースユニット を参照してください。

自動再ロード値がnullのときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

56.7.13 TIM15繰返しカウンタレジスタ（TIM15_RCR）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **REP[7:0]**：繰り返しカウンタ再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP値からカウントをリスタートします。繰返しカウンタは、繰返し更新イベントUEVでのみ、REP値で再ロードされるので、TIM15_RCRレジスタへの書込みは、次の繰返し更新イベントまで有効になりません。

PWMモードでは、（REP+1）はエッジアラインモードでPWM周期の数を意味します。

- エッジアラインモードでは、PWM周期の数
- センターアラインモードでは、PWMの1/2周期の数

56.7.14 TIM15のキャプチャ／比較レジスタ1（TIM15_CCR1）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR1[19:0]**：キャプチャ／比較1値

CC1チャネルが出力として設定されている場合：

CCR1は、実際のキャプチャ／比較1レジスタにロードされる値（プリロード値）です。

TIM15_CCMR1レジスタのOC1PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較1レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIM15_CNTと比較され、tim_oc1出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1チャネルが入力として設定されている場合：

CR1は、最後の入力キャプチャ1イベント（tim_ic1）によって転送されたカウンタ値です。TIMx_CCR1レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

56.7.15 TIM15のキャプチャ／比較レジスタ2（TIM15_CCR2）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR2[19:0]**：キャプチャ／比較2値

CC2チャネルが出力として設定されている場合：

CCR2は、実際のキャプチャ／比較2レジスタにロードされる値（プリロード値）です。

TIM15_CCMR2レジスタのOC2PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較2レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIM15_CNTと比較され、tim_oc2出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] に比較値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2チャネルが入力として設定されている場合：

CCR2は、最後の入力キャプチャ1イベント（tim_ic2）によって転送されたカウンタ値です。TIMx_CCR2レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] にキャプチャ値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

56.7.16 TIM15ブレークおよびデッドタイムレジスタ（TIM15_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、およびDTG[7:0] ビットは、LOCK設定に応じて書込みがロックされるので、TIM15_BDTRレジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **BKBID**：ブレーク双方向

0：ブレーク入力tim_brkは入力モードです。

1：ブレーク入力tim_brkは双方向モードです。

双方向モード（BKBIDビットが1にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 **BKDSRM**：ブレーク解除

0：ブレーク入力tim_brkが設定されます。

1：ブレーク入力tim_brkは解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRMビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット25:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk入力信号をサンプリングする周波数と、tim_brkに適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個のイベント発生が必要です。

0000：フィルタなし、tim_brkは非同期として動作します。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： このビットは、LOCKレベル1がプログラムされているとき（TIM15_BDTRレジスタのLOCKビット）には変更できません。

ビット15 **MOE**：メイン出力イネーブル

このビットは、tim_brk入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOEビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OSSIビットに応じて、tim_ocxおよびtim_ocxn出力が無効か、強制的にアイドル状態になります。

1：tim_ocxおよびtim_ocxn出力は、それぞれのイネーブルビット（TIM15_CCERレジスタのCCxE、CCxNEビット）がセットされている場合は有効です。

詳細については、tim_ocx/tim_ocxnイネーブルの説明を参照してください（セクション 56.7.9： 2384ページのTIM15キャプチャ／比較有効レジスタ（TIM15_CCER））。

ビット14 **AOE**：自動出力イネーブル

0：MOEはソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット13 **BKP**：ブレーク極性

0：ブレーク入力tim_brkはアクティブ･ローです。

1：ブレーク入力tim_brkはアクティブ･ハイです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット12 **BKE**：ブレークイネーブル

0：ブレーク入力（tim_brkおよびtim_sys_brkクロック障害イベント）は無効です。

1：ブレーク入力（tim_brkおよびtim_sys_brkクロック障害イベント）は有効です。

このビットは、LOCKレベル1がプログラムされているとき（TIM15_BDTRレジスタのLOCKビット）には変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット11 **OSSR**：RUNモードのオフ状態の選択

このビットは、MOE=1のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSRは、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、tim_ocx/tim_ocxnイネーブルの説明を参照してください（セクション 56.7.9： 2384ページのTIM15キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：非アクティブのとき、tim_ocx/tim_ocxn出力は無効です（タイマは出力の制御を解放し、制御はハイインピーダンス状態を強制するGPIOによって引き継がれます）。

1：非アクティブのとき、CCxE=1またはCCxNE=1になると、tim_ocx/tim_ocxn出力は、非アクティブレベルで有効になります（出力は引き続きタイマで制御されます）。

注： このビットは、LOCKレベル2がプログラムされているとき（TIM15_BDTRレジスタのLOCKビット）には変更できません。

ビット10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0のとき、出力として設定されているチャネルで使用されます。

詳細については、tim_ocx/tim_ocxnイネーブルの説明を参照してください（セクション 56.7.9： 2384ページのTIM15キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：非アクティブのとき、tim_ocx/tim_ocxn出力は無効です（tim_ocx/tim_ocxnイネーブル出力信号 = 0）。

1：非アクティブのとき、tim_ocx/tim_ocxn出力は、CCxE=1またはCCxNE=1になると、まず、強制的にアイドルレベルになります。（tim_ocx/tim_ocxnイネーブル出力信号 = 1）

注： このビットは、LOCKレベル2がプログラムされているとき（TIM15_BDTRレジスタのLOCKビット）には変更できません。

ビット9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCKオフ - どのビットも書込み保護されません。

01：LOCKレベル1 - TIM15_BDTRレジスタのDTGビット、TIM15_CR2レジスタのOISxおよびOISxNビット、およびTIM15_BDTRレジスタのBKBID/BKE/BKP/AOEビットは、書込みができなくなります。

10：LOCKレベル2 - LOCKレベル1に加えて、CC極性ビット（関連するチャネルがCCxSビットを通じて出力に設定されている場合は、TIM15_CCERレジスタのCCxP/CCxNPビット）とOSSRおよびOSSIビットも書き込めなくなります。

11：OCKレベル3 - LOCKレベル2に加えて、CC制御ビット（関連するチャネルがCCxSビットによって出力に設定されている場合は、TIM15_CCMRxレジスタのOCxMおよびOCxPEビット）が書き込めなくなります。

注： LOCKビットは、リセット後に一度だけ書込みができます。いったんTIM15_BDTRレジスタに書込みが行われると、それらの内容は次のリセットまで凍結されます。

ビット7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで$t_{dtg}=t_{DTS}$

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで$T_{dtg}=2xt_{DTS}$

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=8xt_{DTS}$

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=16xt_{DTS}$

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns （250ns単位）

32µsから63µs （1µs単位）

64µsから126µs （2µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされている（TIM15_BDTRレジスタのLOCKビット）とき、変更できません。

56.7.17 TIM15タイマデッドタイムレジスタ2（TIM15_DTR2）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DTPE**：デッドタイムプリロードイネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCKレベル1、2、または3がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット16 **DTAE**：デッドタイム非対称イネーブル

0：立上がりおよび立下がりエッジでのデッドタイムは同一で、DTG[7:0] レジスタで定義されます。

1：立上がりエッジのデッドタイムはDTG[7:0] レジスタで定義され、立下がりエッジのデッドタイムはDTGF[7:0] ビットで定義されます。

注： このビットは、LOCKレベル1、2、または3がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジジェネレータのセットアップ

このビットフィールドでは、立下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここで$t_{dtg}=t_{DTS}$.

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここで$T_{dtg}=2xt_{DTS}$.

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで$T_{dtg}=8xt_{DTS}$.

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで$T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns（250ns単位）

32 µsから63 µs（1 µs単位）

64 µsから126 µs（2 µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされている（TIM15_BDTRレジスタのLOCKビット）とき、変更できません。

56.7.18 TIM15入力選択レジスタ（TIM15_TISEL）

アドレスオフセット：0x5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **TI2SEL[3:0]**：tim_ti2_in[15:0] の入力を選択します。

0000：TIM15_CH2入力（tim_ti2_in0）

0001：tim_ti2_in1

.....

1111：tim_ti2_in15

相互接続一覧については、セクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TI1SEL[3:0]**：tim_ti1_in[15:0] の入力を選択します。

0000：TIM15_CH1入力（tim_ti1_in0）

0001：tim_ti1_in1

.....

1111：tim_ti1_in15

相互接続一覧については、セクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

56.7.19 TIM15オルタネート機能レジスタ1 （TIM15_AF1）

アドレスオフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

製品固有の実装についてはセクション 56.4.2: TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **BKCMP4P**：tim_brk_cmp4入力極性

このビットは、tim_brk_cmp4入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp4入力はアクティブハイです。

1：tim_brk_cmp4入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット12 **BKCMP3P**：tim_brk_cmp3入力極性

このビットは、tim_brk_cmp3入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp3入力はアクティブハイです。

1：tim_brk_cmp3入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット11 **BKCMP2P**：tim_brk_cmp2入力極性

このビットは、tim_brk_cmp2入力検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2入力はアクティブハイです。

1：tim_brk_cmp2入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット10 **BKCMP1P**：tim_brk_cmp1入力極性

このビットは、tim_brk_cmp1入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1入力はアクティブハイです。

1：tim_brk_cmp1入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット9 **BKINP**：TIMx_BKIN入力極性

このビットは、TIMx_BKINオルタネート機能入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN入力はアクティブハイです。

1：TIMx_BKIN入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット8 **BKCMP8E**：tim_brk_cmp8イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp8を有効化します。mdf_brkx出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp8入力は無効です。

1：tim_brk_cmp8入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット7 **BKCMP7E**：tim_brk_cmp7イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp7を有効化します。COMP7出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp7入力は無効です。

1：tim_brk_cmp7入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット6 **BKCMP6E**：tim_brk_cmp6イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp6を有効化します。tim_brk_cmp6出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp6入力は無効です。

1：tim_brk_cmp6入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット5 **BKCMP5E**：tim_brk_cmp5イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp5を有効化します。tim_brk_cmp5出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp5入力は無効です。

1：tim_brk_cmp5入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット4 **BKCMP4E**：tim_brk_cmp4イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp4を有効化します。tim_brk_cmp4出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp4入力は無効です。

1：tim_brk_cmp4入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット3 **BKCMP3E**：tim_brk_cmp3イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp3を有効化します。tim_brk_cmp3出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp3入力は無効です。

1：tim_brk_cmp3入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット2 **BKCMP2E**：tim_brk_cmp2イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp2を有効化します。tim_brk_cmp2出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp2入力は無効です。

1：tim_brk_cmp2入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット1 **BKCMP1E**：tim_brk_cmp1イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp1を有効化します。tim_brk_cmp1出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp1入力は無効です。

1：tim_brk_cmp1入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット0 **BKINE**：TIMx_BKIN入力イネーブル

このビットは、タイマのtim_brk入力に対してTIMx_BKINオルタネート機能入力を有効化します。TIMx_BKIN入力は、ほかのtim_brkソースとの「論理和」がとられます。

0：TIMx_BKIN入力は無効です。

1：TIMx_BKIN入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

56.7.20 TIM15オルタネート機能レジスタ2 （TIM15_AF2）

アドレスオフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **OCRSEL[2:0]**：ocref_clrソース選択

これらのビットは、ocref_clr入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

010：tim_ocref_clr2

011：tim_ocref_clr3

100：tim_ocref_clr4

101：tim_ocref_clr5

110：tim_ocref_clr6

111：tim_ocref_clr7

製品固有の実装についてはセクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

注： これらのビットは、LOCKレベル1がプログラムされている場合（TIM15_BDTRレジスタのLOCKビット）、変更できません。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

56.7.21 TIM15 DMA制御レジスタ（TIM15_DCR）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **DBSS[3:0]**：DMAバーストソース選択

このビットフィールドは、DMA バースト転送をトリガする割込みソースを定義します（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）。

0000：予約済み

0001：更新

0010：CC1

0110：COM

0111：トリガ

その他：予約済み

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **DBL[4:0]**：DMAバースト長

この5bitのフィールドは、DMA転送長（タイマは、TIM15_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：転送1回
00001：2転送
00010：3転送
...
10001：18転送

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DBA[4:0]**：DMAベースアドレス

この5bitのフィールドは、DMA転送のベースアドレスを指定します（TIM15_DMARアドレスを通じて読出し／書込みアクセスが行われるとき）。DBAは、TIM15_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIM15_CR1
00001：TIM15_CR2
00010：TIM15_SMCR
.....

56.7.22 完全転送のTIM15 DMAアドレス（TIM15_DMAR）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAB[31:0]**：DMAバーストアクセスレジスタ

DMARレジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIM15_CR1アドレス）+ （DBA + DMAインデックス）x 4

ここで、TIM15_CR1アドレスは制御レジスタ1のアドレスであり、DBAはTIM15_DCRレジスタで設定されたDMAベースアドレスであり、DMAインデックスはDMA転送によって自動的に制御され、範囲は0からDBLです（DBLはTIM15_DCR内で設定）。

56.7.23 TIM15レジスタマップ

TIM15レジスタは、次の表のように、16bitのアドレス可能レジスタとしてマップされます。

表584.TIM15レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM15_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIM15_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値																						0	0	0	0	0	0	0	0	0		0
0x08	TIM15_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値								0			0	0				0									0	0	0	0		0	0	0
0x0C	TIM15_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																		0	0				0	0	0	0	0			0	0	0
0x10	TIM15_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0		0	0	0			0	0	0
0x14	TIM15_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
	リセット値																									0	0	0			0	0	0

表584.TIM15レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x18	**TIM15_CCMR1** 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2 PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM15_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIM15_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値																									0		0	0	0	0	0	0
0x24	**TIM15_CNT**	UIFCPYまたはRes.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIM15_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIM15_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	**TIM15_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	**TIM15_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIM15_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38 ～ 0x40	予約済み	Res.																															
0x44	**TIM15_BDTR**	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48 ～ 0x50	予約済み	Res.																															
0x54	**TIM15_DTR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x58	予約済み	Res.																															

表584.TIM15レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	**TIM15_TISEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																					0	0	0	0					0	0	0	0
0x60	**TIM15_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x64	**TIM15_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCR SEL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x68 - 0x3D8	予約済み	Res.																															
0x3DC	**TIM15_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]				Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値													0	0	0	0				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIM15_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

56.8 TIM16/TIM17 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション1.2 を参照してください。

56.8.1 TIMx制御レジスタ1（TIMx_CR1）（x = 16～17）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw				rw	rw	rw	rw

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DITHEN**：ディザリングイネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHENビットは、CENビットがリセットされているときのみ変更できます。

ビット11 **UIFREMAP**：UIFステータスビットの再配置

0：再配置なし。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされません。

1：再配置は有効です。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{tim_ker_ck}$

01：$t_{DTS}=2*t_{tim_ker_ck}$

10：$t_{DTS}=4*t_{tim_ker_ck}$

11：予約済み

ビット7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARRレジスタはバッファされません。

1：TIMx_ARRレジスタはバッファされます。

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **OPM**：ワンパルスモード。

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CENビットをクリア）。

ビット2 **URS**：更新リクエストソース

このビットは、UEVイベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたはDMAリクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット1 **UDIS**：更新ディセーブル

このビットは、UEVイベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEVは有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UGビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UGビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CENビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的にCENビットをセットできます。

56.8.2 TIMx制御レジスタ2（TIMx_CR2）（x = 16～17）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **OIS1N**：出力アイドル状態1（tim_oc1n出力）

0：MOE=0のとき、デッドタイム後にtim_oc1n=0

1：MOE=0のとき、デッドタイム後にtim_oc1n=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKRレジスタのLOCKビット）。

ビット8 **OIS1**：出力アイドル状態1（tim_oc1出力）

0：MOE=0のとき、デッドタイム後にtim_oc1=0

1：MOE=0のとき、デッドタイム後にtim_oc1=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKRレジスタのLOCKビット）。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **CCDS**：キャプチャ／比較DMA選択

0：CCx DMAリクエストは、CCxイベントが発生すると送信されます。

1：CCx DMAリクエストは、更新イベントが発生すると送信されます。

ビット2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMGビットをセットすることによってのみ更新されます。。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMGビットをセットすることによって、またはtim_trgiの立ち上がりエッジで更新されます（使用可能な場合）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、およびOCxMビットはプリロードされません。

1 :CCxE、CCxNE、およびOCxMビットがプリロードされます。書込みの後、これらのビットはCOMビットがセットされたときにのみ更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

56.8.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 16〜17）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
						rw	rw	rw		rw				rw	rw

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC1DE**：キャプチャ／比較1 DMAリクエストイネーブル

0：CC1 DMAリクエストは無効です。

1：CC1 DMAリクエストは有効です。

ビット8 **UDE**：更新DMAリクエストイネーブル

0：更新DMAリクエストは無効です。

1：更新DMAリクエストは有効です。

ビット7 **BIE**：ブレーク割込み有効

0：ブレーク割込みは無効です。

1：ブレーク割込みは有効です。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **COMIE**：COM割込み有効

0：COM割込みは無効です。

1：COM割込みは有効です。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CC1IE**：キャプチャ／比較1割込みイネーブル

0：CC1割込みは無効です。

1：CC1割込みは有効です。

ビット0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

56.8.4 TIMxステータスレジスタ（TIMx_SR）（x = 16～17）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC1OF**：キャプチャ／比較1オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバーキャプチャは検出されていません。

1：CC1IFフラグがすでにセットされているときに、カウンタの値がTIMx_CCR1レジスタにキャプチャされました。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **BIF**：ブレーク割込みフラグ

このフラグは、tim_brk入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **COMIF**：COM割込みフラグ

このフラグは、COMイベント時にハードウェアによってセットされます（キャプチャ/比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COMイベントは発生していません。

1：COM割込みがペンディング中です。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CC1IF**：キャプチャ／比較1割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、またはTIMx_CCR1レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネルCC1が出力として設定されている場合：カウンタTIMx_CNTの内容がTIMx_CCR1レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1の内容がTIMx_ARRの内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）にCC1IFビットはハイになります。センターアラインモードでのフラグの設定には3つのオプションがあります。詳細については、TIMx_CR1レジスタのCMSビットを参照してください。

CC1チャネルが入力として設定されている場合：このビットは、TIMx_CCR1レジスタでカウンタ値がキャプチャされた（TIMx_CCERのCC1PおよびCC1NPビット設定で定義されたエッジ検知に従って、IC1でエッジが検出された）ときに設定されます。

ビット0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰返しカウンタ値に関するオーバーフロー（繰返しカウンタ =0 の場合の更新）、およびTIMx_CR1レジスタのUDIS=0の場合。

– TIMx_CR1レジスタのURS=0かつUDIS=0であり、TIMx_EGRレジスタのUGビットを使用して、CNTがソフトウェアによって再初期化されたとき。

56.8.5 TIMxイベント生成レジスタ（TIMx_EGR）（x = 16～17）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
								w		w				w	w

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたはDMA転送が発生します。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPCビットがセットされているときには、CCxE、CCxNE、およびOCxMビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CC1G**：キャプチャ／比較1生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル1でキャプチャ／比較イベントが生成されます。

CC1チャネルが出力として設定されている場合：

CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。

CC1チャネルが入力として設定されている場合：

カウンタの現在値がTIMx_CCR1レジスタにキャプチャされます。CC1IFフラグがセットされ、対応する割込みまたはDMAリクエストが送信されます（有効な場合）。CC1IFフラグがすでにハイの場合、CC1OFフラグがセットされます。

ビット0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

56.8.6 TIMxキャプチャ／比較モードレジスタ1（TIMx_CCMR1）（x = 16～17）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **IC1F[3:0]**：入力キャプチャ 1フィルタ

このビットフィールドは、tim_ti1入力をサンプリングする周波数と、tim_ti1に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個の連続イベント発生が必要です。

0000：フィルタなし、サンプリングはf_{DTS} で行われます。
0001：$f_{SAMPLING}$= $f_{tim_ker_ck}$、N = 2
0010：$f_{SAMPLING}$= $f_{tim_ker_ck}$、N = 4
0011：$f_{SAMPLING}$= $f_{tim_ker_ck}$、N = 8
0100：$f_{SAMPLING}$= f_{DTS}/2、N =
0101：$f_{SAMPLING}$= f_{DTS}/2、N = 8
0110：$f_{SAMPLING}$= f_{DTS}/4、N = 6
0111：$f_{SAMPLING}$= f_{DTS}/4、N = 8
1000：$f_{SAMPLING}$= f_{DTS}/8、N = 6
1001：$f_{SAMPLING}$= f_{DTS}/8、N = 8
1010：$f_{SAMPLING}$= f_{DTS}/16、N = 5
1011：$f_{SAMPLING}$= f_{DTS}/16、N = 6
1100：$f_{SAMPLING}$= f_{DTS}/16、N = 8
1101：$f_{SAMPLING}$= f_{DTS}/32、N = 5
1110：$f_{SAMPLING}$= f_{DTS}/32、N = 6
1111：$f_{SAMPLING}$= f_{DTS}/32、N = 8

ビット3:2 **IC1PSC[1:0]**：入力キャプチャ 1プリスケーラ

このビットフィールドは、CC1入力（tim_ic1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCERレジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2イベントごとに行われます。
10：キャプチャは、4イベントごとに行われます。
11：キャプチャは、8イベントごとに行われます。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1に配置されます。

その他：予約済み

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

56.8.7 TIMxキャプチャ／比較モードレジスタ1 [オルタネート]（TIMx_CCMR1）（x = 16～17）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応するCCxSビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル1と出力比較モードでのチャネル2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **OC1CE**：出力比較1クリアイネーブル

0：tim_oc1refはtim_ocref_clr入力の影響を受けません。

1：tim_oc1refはtim_ocref_clr入力のハイレベルが検出されるとクリアされます。

ビット16、6:4 **OC1M[3:0]**：出力比較1モード

これらのビットは、tim_oc1およびtim_oc1nが導き出される出力基準信号tim_oc1refの動作を定義します。tim_oc1refはアクティブハイですが、tim_oc1およびtim_oc1nのアクティブレベルはCC1PおよびCC1NPビットに依存します。

0000：停止 - 出力比較レジスタTIMx_CCR1とカウンタTIMx_CNTとの間の比較結果は出力に影響しません。このモードは、タイマをソフトウェアタイムベースとして使用する場合に使用できます。タイマ動作中に停止モードが有効になると、出力は停止状態になる前の状態（アクティブまたは非アクティブ）を維持します。

0001：一致時にチャネル1をアクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル1を非アクティブレベルに設定します。tim_oc1ref信号は、カウンタTIMx_CNTがキャプチャ／比較レジスタ1（TIMx_CCR1）と一致したときに、強制的に低になります。

0011：反転 - TIMx_CNT = TIMx_CCR1のとき、tim_oc1refは反転します。

0100：強制非アクティブレベル - tim_oc1refは強制的にローになります。

0101：強制アクティブレベル - tim_oc1refは強制的にハイになります。

0110：PWMモード1 - チャネル1は、TIMx_CNT < TIMx_CCR1の場合はアクティブに、そうでない場合は非アクティブになります。

0111：PWMモード2 - チャネル1は、TIMx_CNT < TIMx_CCR1の場合は非アクティブに、そうでない場合はアクティブになります。

その他：予約済み

注： これらのビットは、LOCKレベル3がプログラムされていて（TIMx_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力に設定）のときには、変更できません。

PWMモードでは、比較結果が変化したとき、出力比較モードが「停止」モードから「PWM」モードに切り替わったとき、および出力比較モードが「強制アクティブ／非アクティブ」モードから「PWM」モードに切り替わったときに、OCREFのレベルが変化します。

ビット3 **OC1PE**：出力比較1プリロードイネーブル

0：TIMx_CCR1のプリロードレジスタは無効です。TIMx_CCR1は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCKレベル3がプログラムされていて（TIMx_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力に設定）のときには、変更できません。

ビット2 **OC1FE**：出力比較1高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1パルスモード（TIMx_CR1レジスタのOPMビットのセット）で使用する必要があります。

0：CC1の動作は、トリガがオンのときでも、通常、カウンタとCCR1の値に依存します。トリガ入力のエッジ発生からCC1出力が有効になるまでの最小遅延は、5クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1出力に対して、比較一致のように働きます。このため、tim_ocxは、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1出力を有効にするまでの遅延は、3クロックサイクルに短縮されます。OC1FEは、チャネルがPWM1またはPWM2モードに設定されている場合のみ機能します。

ビット1:0 **CC1S[1:0]**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1チャネルは出力として設定されます。

01：CC1チャネルは入力として設定され、tim_ic1はtim_ti1にマッピングされます。

その他：予約済み

注： CC1Sビットは、チャネルがオフ（TIMx_CCERレジスタのCC1E=0）のときにのみ書込み可能です。

56.8.8 TIMxキャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

ビット15:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **CC1NP**：キャプチャ／比較1相補出力極性

CC1チャネルが出力として設定されている場合：

0：tim_oc1nlはアクティブハイです。

1：tim_oc1nlはアクティブローです。

CC1チャネルが入力として設定されている場合：

このビットは、tim_ti1fp1の極性を定義するためにCC1Pと組合せて使用されます。CC1Pの説明を参照してください。

注： このビットは、LOCKレベル2または3がプログラムされ、（TIMx_BDTRレジスタのLOCKビット）、かつCC1S="00"（チャネルは出力として設定）になった直後は書込みできません。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2レジスタのCCPCビットがセットされた場合、CC1NPアクティブビットは、転流イベント（COM）が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット2 **CC1NE**：キャプチャ／比較1相補出力イネーブル

0：オフ - tim_oc1nはアクティブではありません。tim_oc1nのレベルは、MOE、OSSI、OSSR、OIS1、OIS1NおよびCC1Eビットによって決まります。

1：オン - tim_oc1n信号は、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1Eビットにより、対応する出力ピンに出力されます。

ビット1 **CC1P**：キャプチャ／比較1出力極性

0：OC1はアクティブハイ（出力モード）／エッジ検知を選択（入力モード、下記を参照）

1：OC1はアクティブロー（出力モード）／エッジ検知を選択（入力モード、下記を参照）

CC1チャネルが入力として設定されているときは、CC1NP/CC1Pの両ビットは、トリガまたはキャプチャ動作に対するTI1FP1とTI2FP1のアクティブ極性を選択します。

CC1NP=0、CC1P=0：非反転/立上がりエッジ。この回路はTIxFP1の立上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。

CC1NP=0、CC1P=1：反転／立下がりエッジ。この回路はTIxFP1の立下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1が反転されます（ゲートモードでのトリガ動作）。

CC1NP=1、CC1P=1：非反転／両エッジ／この回路はTIxFP1の立上がりエッジと立下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1は反転されません（ゲートモードでのトリガ動作）。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCKレベル2または3がプログラムされた直後は書込みできません（TIMx_BDTRレジスタのLOCKビット）。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2レジスタのCCPCビットがセットされた場合、CC1Pアクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット0 **CC1E**：キャプチャ／比較1出力イネーブル

0：キャプチャモードは無効／OC1は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1信号は、対応する出力ピンに出力されます。

CC1チャネルが出力として設定されている場合、OC1レベルは、CC1Eビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、およびCC1NEビットに依存します。詳細については、表585を参照してください。

表585.ブレーク機能を持つ相補tim_oc1およびtim_oc1nチャネルの出力制御ビット（TIM16/TIM17）

制御ビット					出力状態(1)	
MOEビット	OSSIビット	OSSRビット	CC1Eビット	CC1NEビット	tim_oc1出力状態	tim_oc1n出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1 = 0 tim_oc1n = 0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1 = 0	tim_oc1ref + 極性tim_oc1n = tim_oc1ref XOR CC1NP
		0	1	0	tim_oc1ref + 極性 tim_oc1=tim_oc1ref XOR CC1P	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1n = 0
		X	1	1	tim_oc1ref + 極性 + デッドタイム	tim_oc1refに対する相補（tim_oc1refではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（非アクティブ状態で出力有効） tim_oc1=CC1P	tim_oc1ref + 極性 tim_oc1n=tim_oc1ref XOR CC1NP
		1	1	0	tim_oc1ref + 極性 tim_oc1=tim_oc1ref XOR CC1P	オフ状態（非アクティブ状態で出力有効） tim_oc1n=CC1NP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイインピーダンス）	
	1		0	0		
			0	1	オフ状態（非アクティブ状態で出力有効） 非同期に：tim_oc1=CC1P、tim_oc1n=CC1NP クロックが存在する場合：デッドタイム後、tim_oc1=OIS1およびtim_oc1n=OIS1N。ただし、OIS1とOIS1Nは、アクティブ状態におけるtim_oc1とtim_oc1nの両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIOコントローラが制御を引き継いだ場合）、OIS1、OIS1N、CC1P、およびCC1NPビットはクリアされたままでなければなりません。

注： 相補 tim_oc1 および tim_oc1n チャネルに接続されている外部入出力ピンの状態は、tim_oc1 および tim_oc1nチャネルの状態と、GPIO制御 および オルタネート機能選択レジスタに依存します。

56.8.9 TIMxカウンタ（TIMx_CNT）（x = 16～17）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UIFCPY**：UIFコピー

このビットはTIMx_ISRレジスタのUIFビットの読出し専用コピー。TIMx_CR1のUIFREMAPビットがリセットされている場合、ビット31は予約済みです。

ビット30:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分はありません。

56.8.10 TIMxプリスケーラ（TIMx_PSC）（x = 16～17）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（ftim_cnt_ck）は$f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSCは、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGRレジスタのUGビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

56.8.11 TIMx自動再ロードレジスタ（TIMx_ARR）（x = 16～17）

アドレスオフセット：0x2C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **ARR[19:0]**：自動再ロード値

ARRは、実際の自動再ロードレジスタにロードされる値です。

ARRの更新と動作の詳細については、セクション 56.4.3：2329ページのタイムベースユニット を参照してください。

自動再ロード値がnullのときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

56.8.12 TIMx繰返しカウンタレジスタ（TIMx_RCR）（x = 16～17）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **REP[7:0]**：繰り返しカウンタ再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP値からカウントをリスタートします。繰返しカウンタは、繰返し更新イベントUEVでのみ、REP値で再ロードされるので、TIMx_RCRレジスタへの書込みは、次の繰返し更新イベントまで有効になりません。

PWMモードでは、（REP+1）はエッジアラインモードでPWM周期の数を意味します。

- エッジアラインモードでは、PWM周期の数
- センターアラインモードでは、PWMの1/2周期の数

56.8.13 TIMxキャプチャ／比較レジスタ1（TIMx_CCR1）（x = 16～17）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **CCR1[19:0]**：キャプチャ／比較1値

CC1チャネルが出力として設定されている場合：

CCR1は、実際のキャプチャ／比較1レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1レジスタのOC1PEビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較1レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタTIMx_CNTと比較されて、tim_oc1出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1チャネルが入力として設定されている場合：

CCR1は、最後の入力キャプチャ 1イベント（tim_ic1）によって転送されたカウンタ値です。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

56.8.14 TIMxブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、およびDTG[7:0] ビットは、LOCK設定に応じて書込みがロックされるので、TIMx_BDTRレジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **BKBID** ブレーク双方向

0：ブレーク入力tim_brkは入力モードです。

1：ブレーク入力tim_brkは双方向モードです。

双方向モード（BKBIDビットが1にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 **BKDSRM** ブレーク解除

0：ブレーク入力tim_brkが設定されます。

1：ブレーク入力tim_brkは解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRMビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット25:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk入力をサンプリングする周波数と、tim_brkに適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすにはN個のイベント発生が必要です。

0000：フィルタなし、tim_brkは非同期として動作します。
0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2
0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4
0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111: $f_{SAMPLING}=f_{DTS}/4$, N=8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
0111: $f_{SAMPLING}=f_{DTS}/4$, N=8
1010: $f_{SAMPLING}=f_{DTS}/16$, N=5
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100: $f_{SAMPLING}=f_{DTS}/16$, N=8
1101: $f_{SAMPLING}=f_{DTS}/32$, N=5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111: $f_{SAMPLING}=f_{DTS}/32$, N=8

このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット15 **MOE**：メイン出力イネーブル

このビットは、tim_brk入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOEビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OSSIビットに応じて、tim_oc1およびtim_oc1n出力が無効か、強制的にアイドル状態になります。
1：tim_oc1およびtim_oc1n出力は、それぞれのイネーブルビット（TIMx_CCERレジスタのCC1E、CC1NEビット）がセットされている場合は有効です。

詳細については、tim_oc1/tim_oc1nイネーブルの説明を参照してください（セクション 56.8.8： 2411ページのTIMxキャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

ビット14 **AOE**：自動出力イネーブル

0：MOEはソフトウェアによってのみセットできます。
1：MOEは、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（tim_brk入力がアクティブでない場合）。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット13 **BKP**：ブレーク極性

0：ブレーク入力tim_brkはアクティブ･ローです。
1：ブレーク入力tim_brkはアクティブ･ハイです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット12 **BKE**：ブレークイネーブル

0：ブレーク入力（tim_brkおよびtim_sys_brkイベント）は無効です。
1：ブレーク入力（tim_brkおよびtim_sys_brkイベント）は有効です。

注： このビットは、LOCKレベル1がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

このビットへの書込み操作では、書込みが有効になるまでに1 APBクロックサイクルの遅延が生じます。

ビット11 **OSSR**：RUNモードのオフ状態の選択

このビットは、MOE=1のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSRは、相補出力がタイマに実装されていない場合には、実装されません。
詳細については、tim_oc1/tim_oc1nイネーブルの説明を参照してください（セクション 56.8.8： 2411ページのTIMxキャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

0：非アクティブのとき、tim_oc1/tim_oc1n出力は無効です（タイマは出力の制御を解放し、制御はハイインピーダンス状態を強制するGPIOによって引き継がれます）。

1：非アクティブのとき、CC1E=1またはCC1NE=1になると、tim_oc1/tim_oc1n出力は、非アクティブレベルで有効になります（出力は引き続きタイマで制御されます）。

注： このビットは、LOCKレベル2がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0のとき、出力として設定されているチャネルで使用されます。
詳細については、tim_oc1/tim_oc1nイネーブルの説明を参照してください（セクション 56.8.8： 2411ページのTIMxキャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

0：非アクティブのとき、tim_oc1/tim_oc1n出力は無効です（tim_oc1/tim_oc1nイネーブル出力信号 = 0）。

1：非アクティブのとき、tim_oc1/tim_oc1n出力は、CC1E=1またはCC1NE=1になると、まず、強制的にアイドルレベルになります。（tim_oc1/tim_oc1nイネーブル出力信号 = 1）

注： このビットは、LOCKレベル2がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCKオフ - どのビットも書込み保護されません。

01：LOCKレベル1 - TIMx_BDTRレジスタのDTGビット、TIMx_CR2レジスタのOISxおよびOISxNビット、およびTIMx_BDTRレジスタのBKBID/BKE/BKP/AOEビットは、書込みができなくなります。

10：LOCKレベル2 - LOCKレベル1に加えて、CC極性ビット（関連するチャネルがCCxSビットを通じて出力に設定されている場合は、TIMx_CCERレジスタのCCxP/CCxNPビット）とOSSRおよびOSSIビットも書き込めなくなります。

11：LOCKレベル3 - LOCKレベル2に加えて、CC制御ビット（関連するチャネルがCCxSビットを通じて出力に設定されている場合は、TIMx_CCMRxレジスタのOCxMおよびOCxPEビット）が書き込めなくなります。

注： LOCKビットは、リセット後に一度だけ書込みができます。いったんTIMx_BDTRレジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで$t_{dtg}=t_{DTS}$

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで$T_{dtg}=2xt_{DTS}$

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=8xt_{DTS}$

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで$T_{dtg}=16xt_{DTS}$

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns （250ns単位）

32µsから63µs （1µs単位）

64µsから126µs （2µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされているとき、変更できません（TIMx_BDTRレジスタのLOCKビット）。

56.8.15 TIMxタイマデッドタイムレジスタ2（TIMx_DTR2）（x = 16～17）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DTPE**：デッドタイムプリロードイネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット16 **DTAE**：デッドタイム非対称イネーブル

0：立上がりおよび立下がりエッジでのデッドタイムは同一で、DTG[7:0] レジスタで定義されます。

1：立上がりエッジのデッドタイムはDTG[7:0] レジスタで定義され、立下がりエッジのデッドタイムはDTGF[7:0] ビットで定義されます。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTRレジスタのLOCKビット）。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジジェネレータのセットアップ

このビットフィールドでは、立下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここでt_{dtg}=t_{DTS}.

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここでT_{dtg}=2xt_{DTS}.

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここでT_{dtg}=8xt_{DTS}.

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここでT_{dtg}=16xt_{DTS}.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0から15875ns（125ns単位）

16µsから31750ns（250ns単位）

32 µsから63 µs（1 µs単位）

64 µsから126 µs（2 µs単位）

注： このビットフィールドは、LOCKレベル1、2、または3がプログラムされているとき、変更できません（TIMx_BDTRレジスタのLOCKビット）。

56.8.16 TIMx入力選択レジスタ（TIMx_TISEL）（x = 16～17）

アドレスオフセット：0x5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TI1SEL[3:0]**：tim_ti1_in[15:0] の入力を選択します。

0000：TIMx_CH1入力（tim_ti1_in0）

0001：tim_ti1_in1

.....

1111：tim_ti1_in15

相互接続一覧については、セクション 56.4.2：TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

56.8.17 TIMxオルタネート機能レジスタ1 （TIMx_AF1）（x = 16～17）

アドレスオフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

製品固有の実装についてはセクション 56.4.2: TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **BKCMP4P**：tim_brk_cmp4入力極性

このビットは、tim_brk_cmp4入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp4入力はアクティブハイです。

1：tim_brk_cmp4入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット12 **BKCMP3P**：tim_brk_cmp3入力極性

このビットは、tim_brk_cmp3入力の感度を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp3入力はアクティブハイです。

1：tim_brk_cmp3入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット11 **BKCMP2P**：tim_brk_cmp2入力極性

このビットは、tim_brk_cmp2入力検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2入力はアクティブハイです。

1：tim_brk_cmp2入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット10 **BKCMP1P**：tim_brk_cmp1入力極性

このビットは、tim_brk_cmp1入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1入力はアクティブハイです。

1：tim_brk_cmp1入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット9 **BKINP**：TIMx_BKIN入力極性

このビットは、TIMx_BKINオルタネート機能入力の検知極性を選択します。BKP極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN入力はアクティブハイです。

1：TIMx_BKIN入力はアクティブローです。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット8 **BKCMP8E**：tim_brk_cmp8イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp8を有効化します。mdf_brkx出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp8入力は無効です。

1：tim_brk_cmp8入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット7 **BKCMP7E**：tim_brk_cmp7イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp7を有効化します。tim_brk_cmp7出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp7入力は無効です。

1：tim_brk_cmp7入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット6 **BKCMP6E**：tim_brk_cmp6イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp6を有効化します。tim_brk_cmp6出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp6入力は無効です。

1：tim_brk_cmp6入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット5 **BKCMP5E**：tim_brk_cmp5イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp5を有効化します。tim_brk_cmp5出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp5入力は無効です。

1：tim_brk_cmp5入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット4 **BKCMP4E**：tim_brk_cmp4イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp4を有効化します。tim_brk_cmp4出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp4入力は無効です。

1：tim_brk_cmp4入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット3 **BKCMP3E**：tim_brk_cmp3イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp3を有効化します。tim_brk_cmp3出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp3入力は無効です。

1：tim_brk_cmp3入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット2 **BKCMP2E**：tim_brk_cmp2イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp2を有効化します。tim_brk_cmp2出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp2入力は無効です。

1：tim_brk_cmp2入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット1 **BKCMP1E**：tim_brk_cmp1イネーブル

このビットは、タイマのtim_brk入力に対してtim_brk_cmp1を有効化します。tim_brk_cmp1出力は、ほかのtim_brkソースとの「論理和」がとられます。

0：tim_brk_cmp1入力は無効です。

1：tim_brk_cmp1入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット0 **BKINE**：TIMx_BKIN入力イネーブル

このビットは、タイマのtim_brk入力に対してTIMx_BKINオルタネート機能入力を有効化します。TIMx_BKIN入力は、ほかのtim_brkソースとの「論理和」がとられます。

0：TIMx_BKIN入力は無効です。

1：TIMx_BKIN入力は有効です。

注： このビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

56.8.18 TIMxオルタネート機能レジスタ2 （TIMx_AF2）（x = 16～17）

アドレスオフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **OCRSEL[2:0]**：tim_ocref_clrソース選択

これらのビットは、tim_ocref_clr入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

010：tim_ocref_clr2

011：tim_ocref_clr3

100：tim_ocref_clr4

101：tim_ocref_clr5

110：tim_ocref_clr6

111：tim_ocref_clr7

製品固有の実装についてはセクション 56.4.2: TIM15/TIM16/TIM17ピンおよび内部信号を参照してください。

注： これらのビットは、LOCKレベル1がプログラムされている場合（TIMx_BDTRレジスタのLOCKビット）、変更できません。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

56.8.19 TIMxオプションレジスタ1（TIMx_OR1）（x = 16～17）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSE32EN	Res.
														rw	

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **HSE32EN**：32分周されたHSEイネーブル

このビットは、tim_ti1_in3用に32分周されたHSEを有効にします。詳細については、表 572 : tim_ti1入力マルチプレクサへの相互接続を参照してください。

0：32分周されたHSEは無効です。

1：32分周されたHSEは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

56.8.20 TIMx DMA制御レジスタ（TIMx_DCR）（x = 16～17）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **DBSS[3:0]**：DMAバーストソース選択

このビットフィールドは、DMAバースト転送をトリガする割込みソースを定義します（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）。

0000：予約済み

0001：更新

0010：CC1

その他：予約済み

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **DBL[4:0]**：DMAバースト長

この5bitのフィールドは、転送数などDMA転送長（タイマは、TIMx_DMARアドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：転送1回
00001：2転送
00010：3転送
...
10001：18転送

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **DBA[4:0]**：DMAベースアドレス

この5bitのフィールドは、DMA転送のベースアドレスを指定します（TIMx_DMARアドレスを通じて読出し／書込みアクセスが行われるとき）。DBAは、TIMx_CR1レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

例：次の転送を考えます：DBL = 7転送かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1アドレスから始めて、7つのレジスタに対して行われます。

56.8.21 TIM16/TIM17 完全転送用のDMAアドレス（TIMx_DMAR）（x = 16～17）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAB[31:0]**：DMAバーストアクセスレジスタ

DMARレジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1アドレス）+ （DBA + DMAインデックス）x 4

ここで、TIMx_CR1アドレスは制御レジスタ1のアドレスであり、DBAはTIMx_DCRレジスタで設定されたDMAベースアドレスであり、DMAインデックスはDMA転送によって自動的に制御され、範囲は0からDBLです（DBLはTIMx_DCR内で設定）。

56.8.22 TIM16/TIM17 レジスタマップ

TIM16/TIM17 レジスタは、次の表のように、16bitアドレス可能レジスタとして配置されます。

表586.TIM16/TIM17レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
	リセット値																							0	0					0	0		0
0x08	予約済み	Res.																															
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
	リセット値																							0	0	0		0				0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0		0		0				0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
	リセット値																									0		0				0	0
0x18	**TIMx_CCMR1** 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
	リセット値																									0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	OC1CE	OC1M [2:0]				OC1PE	OC1FE	CC1S [1:0]	
	リセット値																0									0	0	0	0	0	0	0	0
0x1C	予約済み	Res.																															
0x20	**TIMx_CCER**	Res.	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
	リセット値																													0	0	0	0
0x24	**TIMx_CNT**	UIFCPYまたはRes.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表586.TIM16/TIM17レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	TIMx_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38 - 0x40	予約済み	Res.																															
0x44	TIMx_BDTR	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48 - 0x50	予約済み	Res.																															
0x54	TIMx_DTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x58	予約済み	Res.																															
0x5C	TIMx_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0
0x60	TIMx_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x64	TIMx_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCR SEL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x68	TIMx_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSE32EN	Res.
	リセット値																															0	
0x6C - 0x3D8	予約済み	Res.																															
0x3DC	TIMx_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBSS[3:0]				Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値													0	0	0	0				0	0	0	0	0				0	0	0	0	0
0x3E0	TIMx_DMAR	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

57 基本タイマ（TIM6/TIM7）

57.1 TIM6/TIM7 の概要

基本タイマ TIM6/TIM7 は、プログラム可能なプリスケーラによって駆動される16bit自動再ロードカウンタで構成されています。

これらは、タイムベース生成のための汎用タイマとして使用できます。

基本タイマは、デジタル/アナログコンバータをトリガするために使用することもできます。これはタイマのトリガ出力で行われます。

タイマは完全に独立していて、いかなるリソースも共有しません。

57.2 TIM6/TIM7 の主な特徴

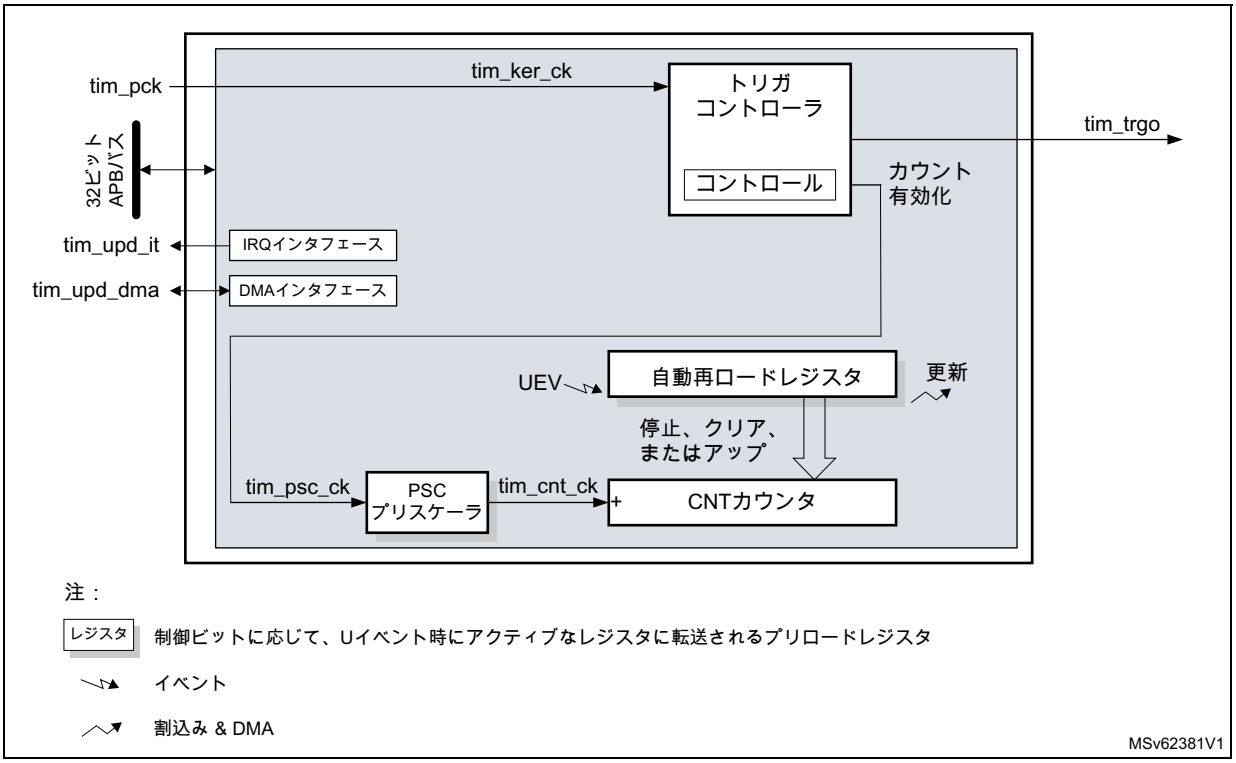
基本タイマ（TIM6/TIM7）の機能は、次のとおりです。

- 16bit自動再ロードアップカウンタ
- 16bitのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を1から65535の間の値で分周可能
- DACをトリガする同期回路
- 更新イベント時の割込み／DMA生成：カウンタオーバーフロー

57.3 TIM6/TIM7機能詳細

57.3.1 TIM6/TIM7 ブロック図

図716. 基本タイマブロック図



57.3.2 TIM6/TIM7 内部信号

このセクションの以下の表は、TIMの入力および出力の一覧です。

表587.TIM内部入力／出力信号

内部信号名	信号タイプ	説明
tim_pclk	入力	タイマAPBクロック
tim_ker_ck	入力	タイマカーネルクロック。このクロックは、tim_pclk（同じソースから導出）と同期する必要があります。クロック比tim_ker_ck/tim_pclkは整数1、2、3、･･･16（最大値）でなければなりません。
tim_trgo	出力	内部トリガ出力。このトリガは他の内部ペリフェラル（DAC）をトリガできます。
tim_upd_it	出力	タイマ更新イベント割込み
tim_upd_dma	出力	タイマ更新DMAリクエスト

57.3.3 TIM6/TIM7 クロック

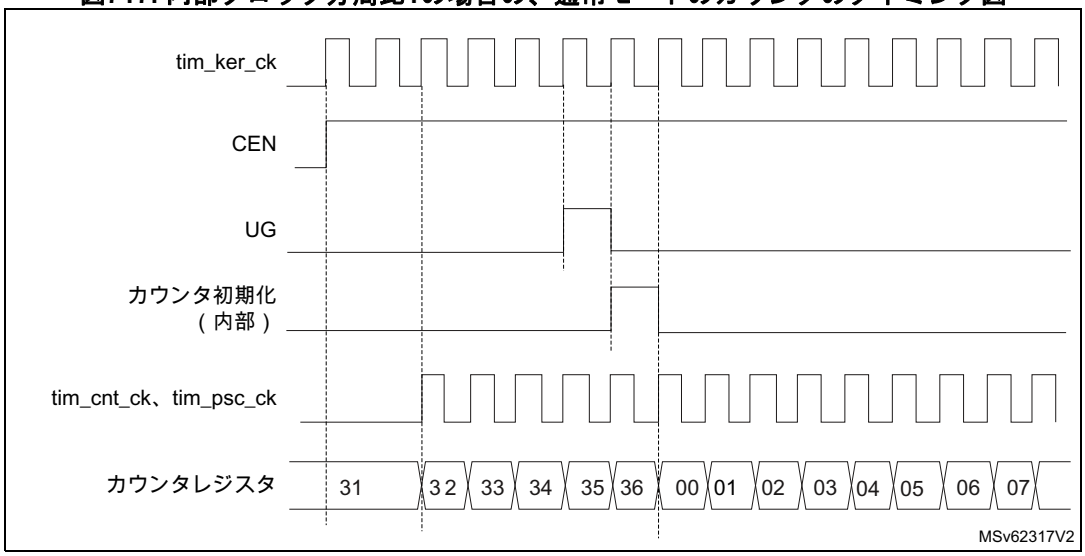
タイマバスインタフェースはtim_pclk APBクロックによってクロック供給されます。

カウンタクロックtim_ker_ckはtim_pclk入力に接続されます。

TIMx_CR1レジスタのCENビットとTIMx_EGRレジスタのUGビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます（ただし、自動的にクリア状態が保持されるUGビットを除く）。CENビットに1が書き込まれるとすぐに、プリスケーラには内部クロックtim_ker_ckが供給されます。

図717に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

図717. 内部クロック分周比1の場合の、通常モードのカウンタのタイミング図



57.3.4 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ16bitアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベントUEVごとに、シャドウレジスタに転送されます。TIMx_CR1レジスタのUDISビットが0である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1レジスタのカウンタイネーブルビット（CEN）がセットされているときのみ、プリスケーラ出力tim_cnt_ckから供給されます。

実際のカウンタイネーブル信号tim_cnt_enは、CENビットセットの1クロックサイクル後にセットされることにご注意ください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を1から65536の間の値で分周することができます。16bitレジスタ（TIMx_PSCレジスタ）を使って制御される16bitカウンタをベースとしています。TIMx_PSC制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図718と図719に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図718. プリスケーラ分周比が1から2に変化したときのカウンタのタイミング図

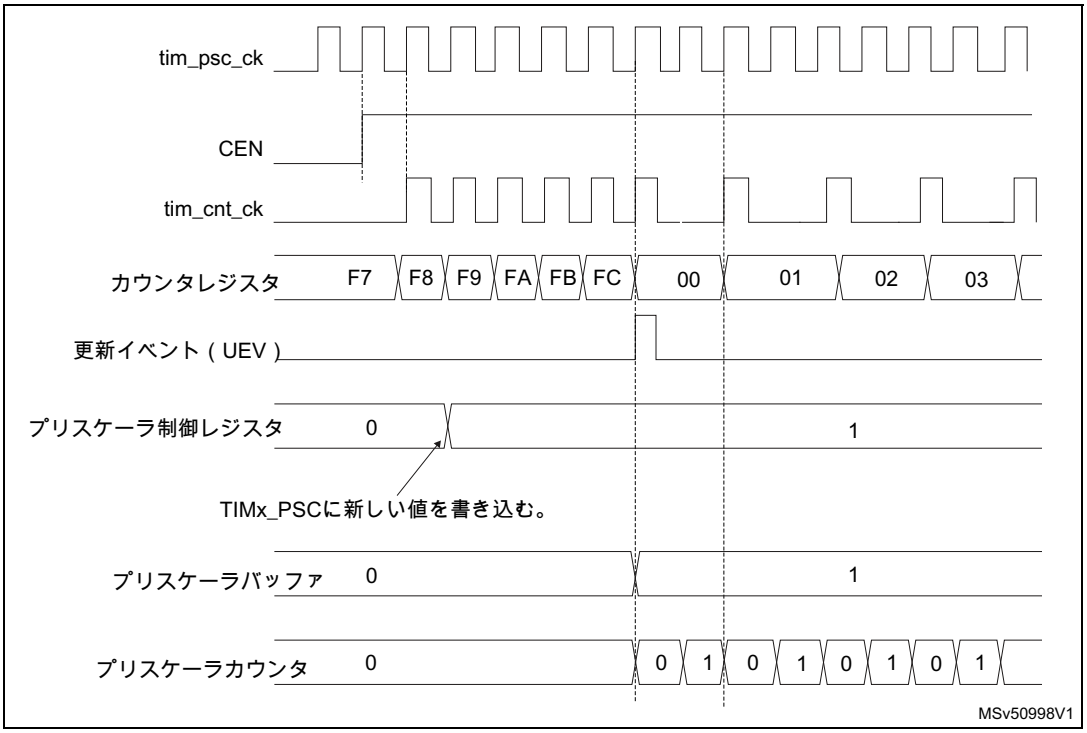
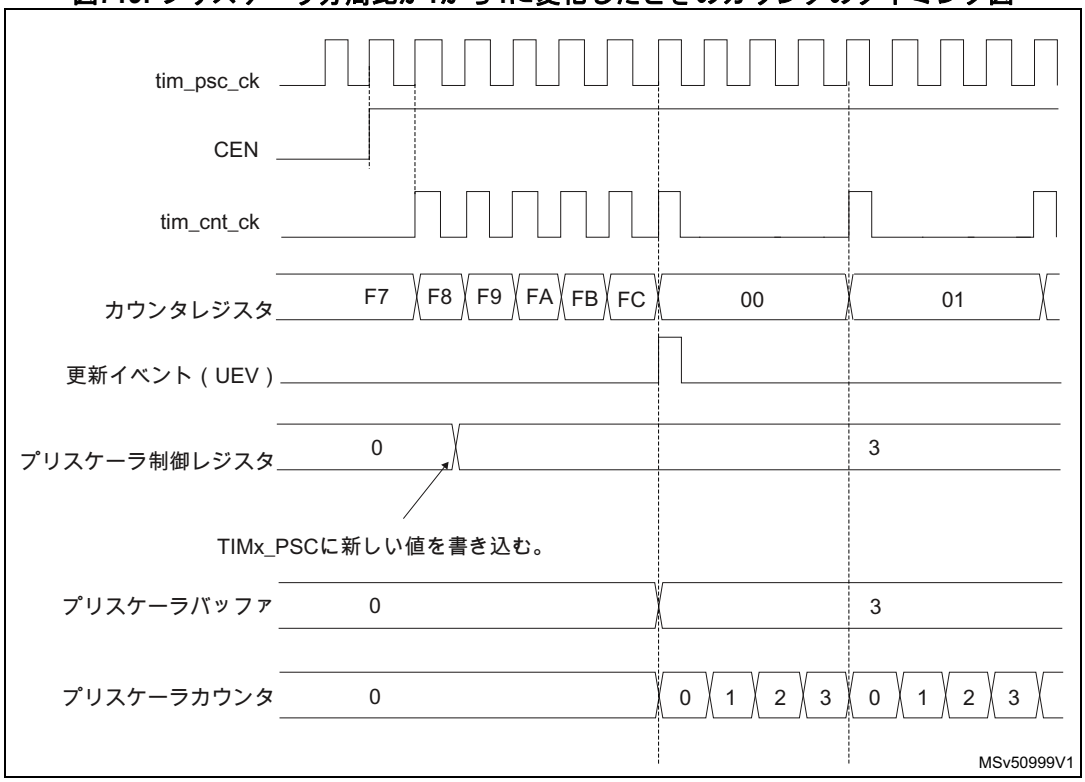


図719. プリスケーラ分周比が1から4に変化したときのカウンタのタイミング図



57.3.5 カウントモード

カウンタは、0から自動再ロード値（TIMx_ARRレジスタの内容）までカウントした後、0からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGRレジスタのUGビットをセットすることによって（ソフトウェアで、または、スレーブモードコントローラを使用して）生成できます。

UEVイベントは、ソフトウェアでTIMx_CR1レジスタのUDISビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDISビットに0が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケーラカウンタは両方とも0からリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1レジスタのURS（更新リクエスト選択）ビットがセットされている場合、UGビットをセットすると更新イベントUEVが生成されますが、UIFフラグはセットされません（割込みやDMAリクエストは送信されません）。

更新イベントが発生すると、すべてのレジスタが更新され、URSビットの設定に応じて、更新フラグ（TIMx_SRレジスタのUIFビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSCレジスタの内容）が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図720. 内部クロック分周比が1の場合のカウンタのタイミング図

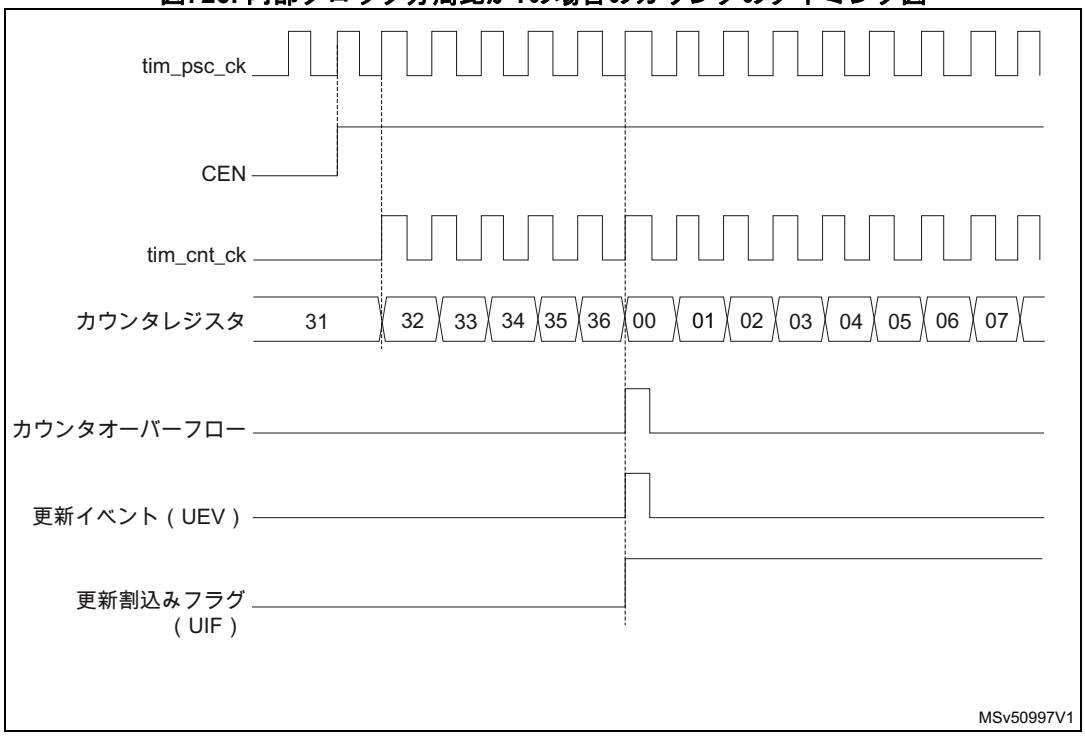


図721. 内部クロック分周比が2の場合のカウンタのタイミング図

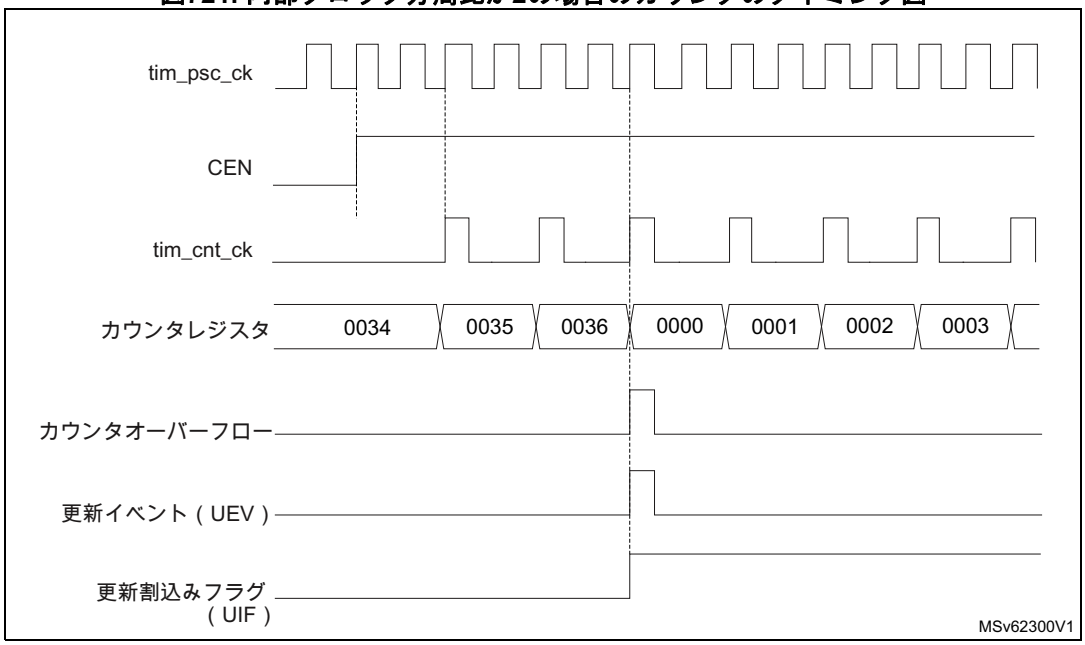


図722. 内部クロック分周比が4の場合のカウンタのタイミング図

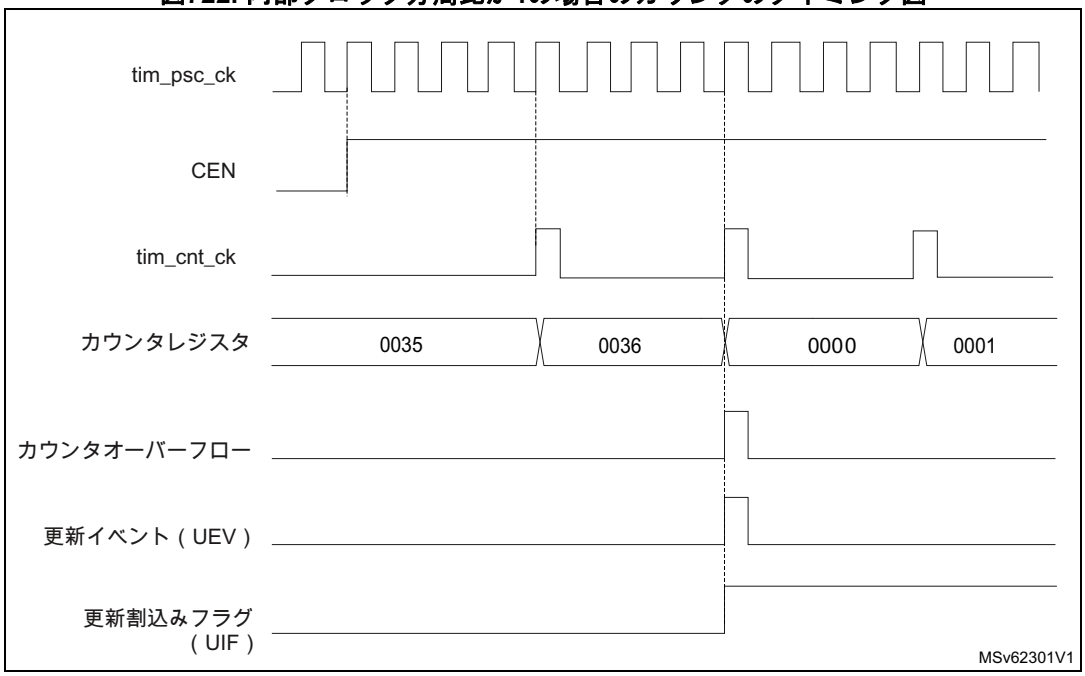


図723. 内部クロック分周比がNの場合のカウンタのタイミング図

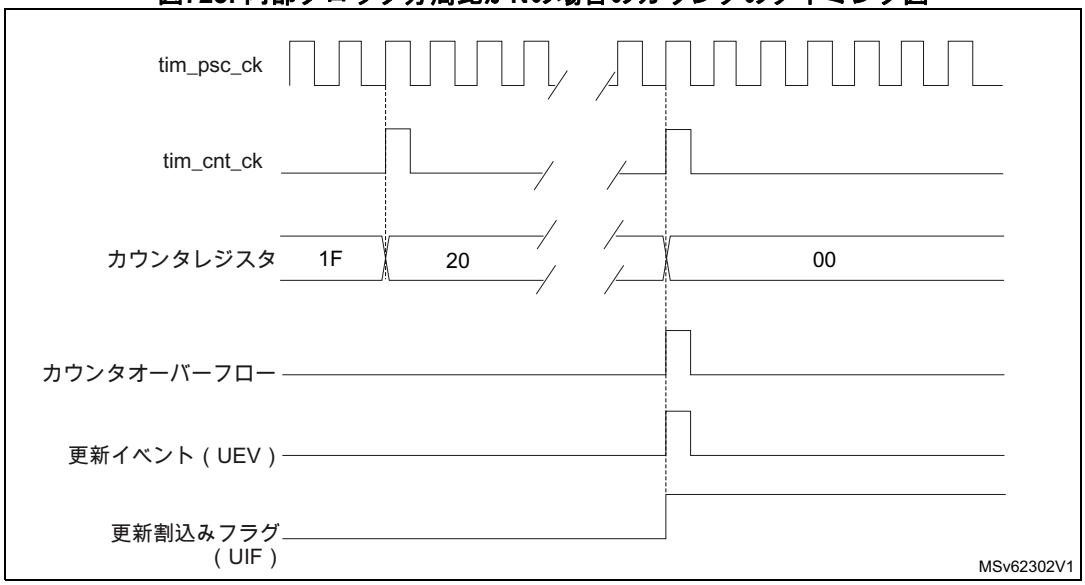


図724. ARPE = 0（TIMx_ARRはプリロードされない）の場合の更新イベント時のカウンタのタイミング図

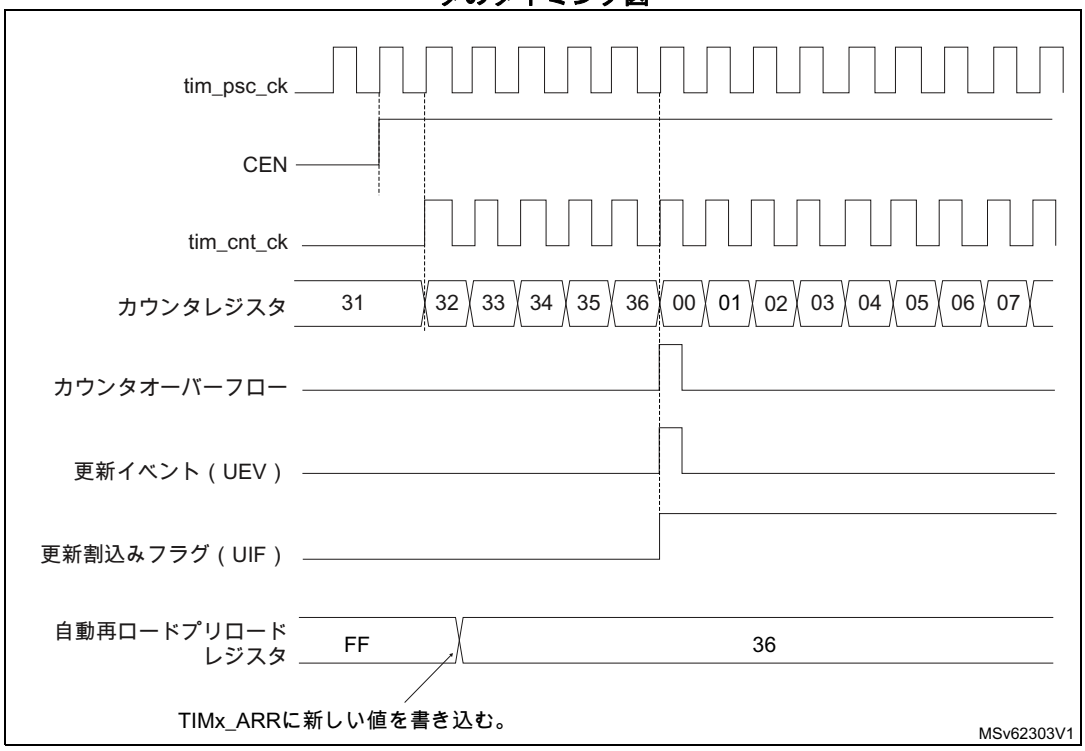
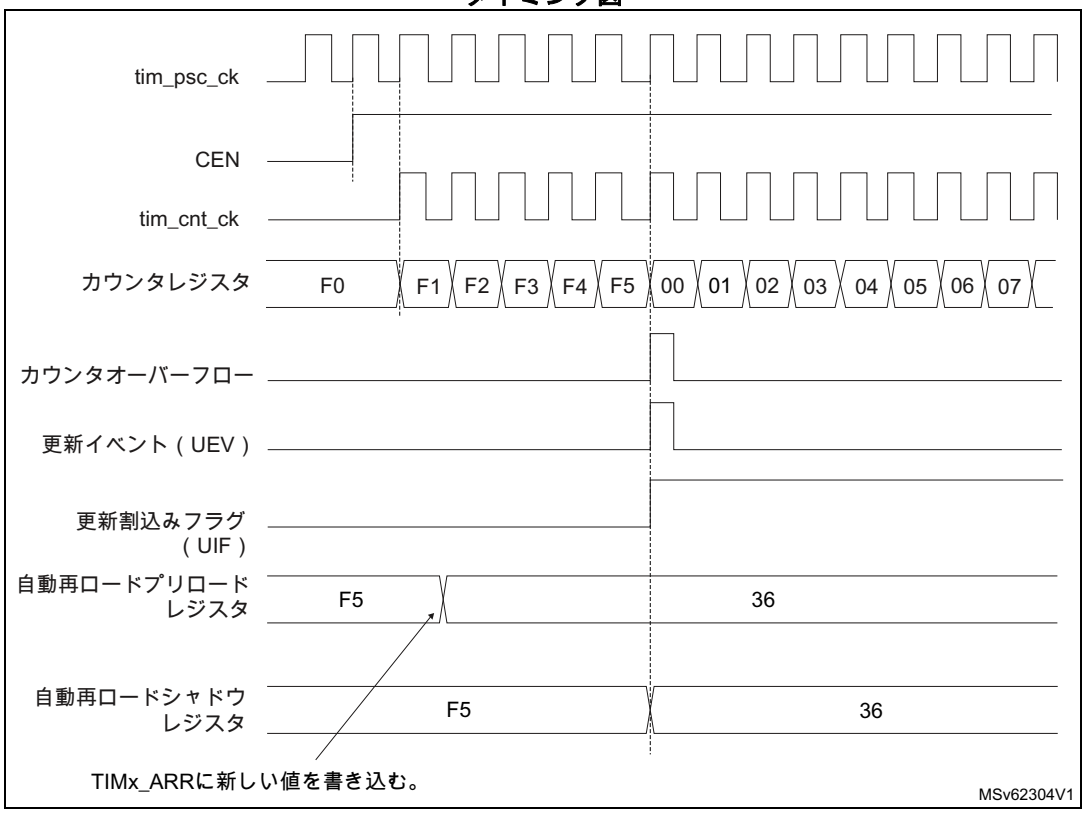


図725. ARPE=1（TIMx_ARRがプリロードされる）の場合の更新イベント時のカウンタのタイミング図



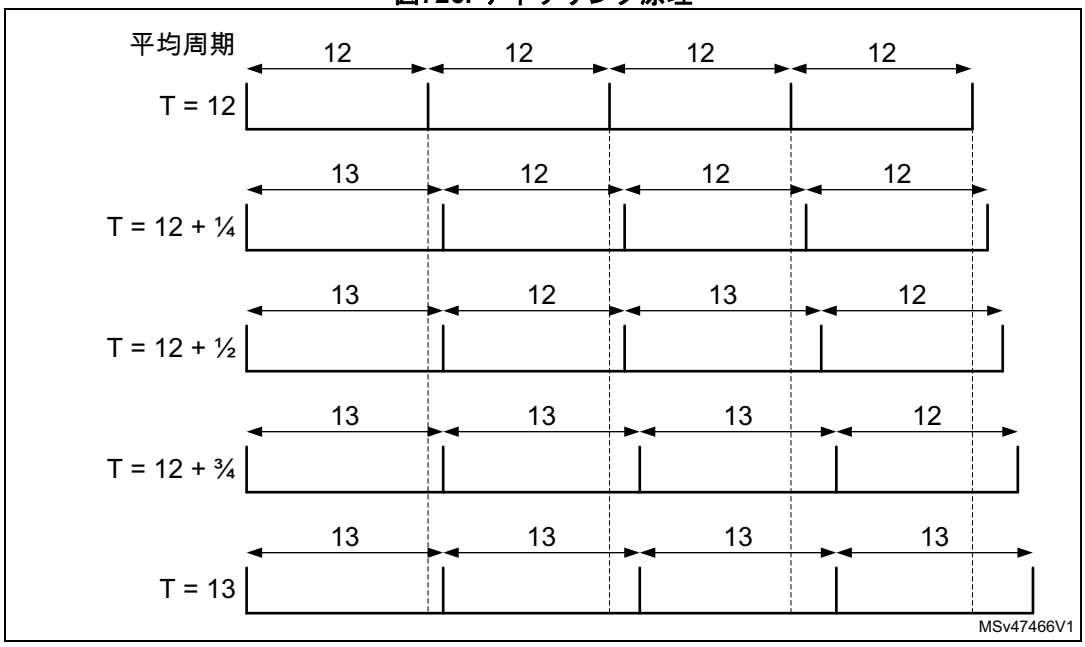
ディザリングモード

タイムベースの有効分解能は、TIMx_CR1レジスタのDITHENビットを使用して、ディザリングモードを有効にすることで増加できます。これは、TIMx_ARRの動作に影響し、タイマがトリガとして使用される場合（通常はDACの場合）に平均カウンタ周期を調整するのに役立ちます。

動作原理は、事前に定義されたパターンで、16個の連続した カウント 周期にわたって実際のARR値をわずかに変更する（1タイマクロック周期を追加する、またはしない）ことです。これにより、平均のカウント周期を考えると、16倍の分解能の向上が可能になります。

下記の図726は、4つの連続するカウント周期に適用されたディザリング原理を示します。

図726. ディザリング原理

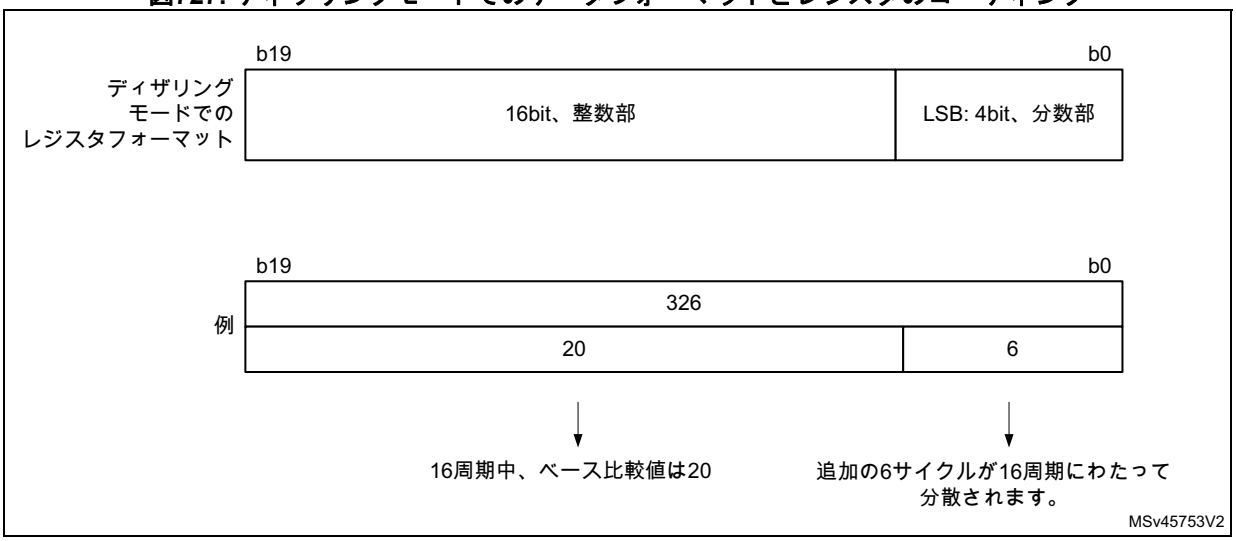


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図727を参照）。

- 下位4bitは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット19:4に左シフトされ、ベース値用のコーディングになります。

注： **DITHENビットをリセットするときは、次の手順に従う必要があります。**
1を参照してください。CENおよびARPEビットをリセットする必要があります。
2を参照してください。ARR[3:0] ビットをリセットする必要があります。
3を参照してください。DITHENビットをリセットする必要があります。
4を参照してください。CENビットをセットすることができます（最終的にARPE = 1とともに）。

図727. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

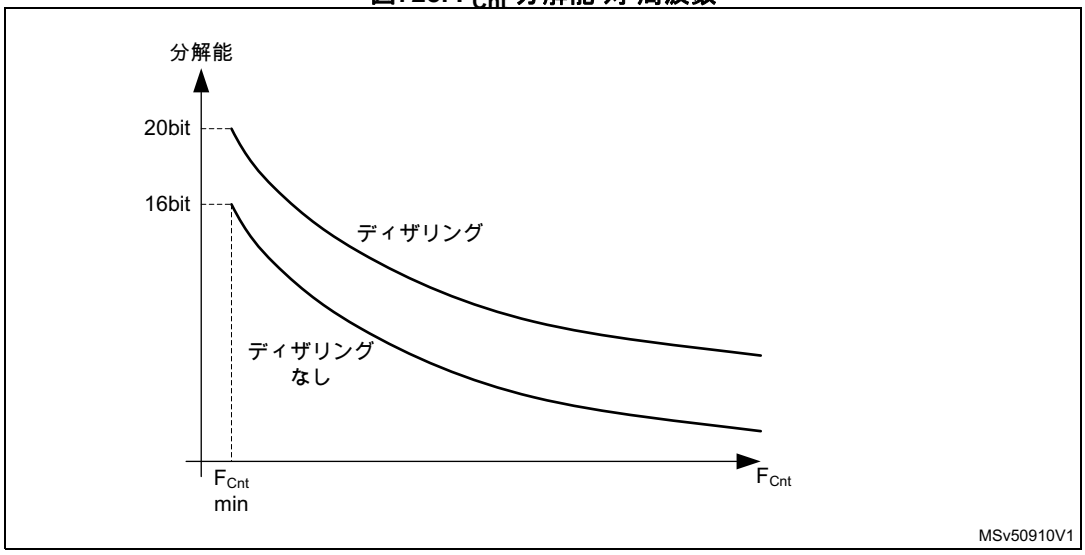
$$\text{ディザリングモードが無効の場合: } F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモードが有効な場合: } F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARRの最大値は、ディザリングモードでは0xFFFEFに制限されています（整数部では65534、ディザリング部分では15に対応します）。**

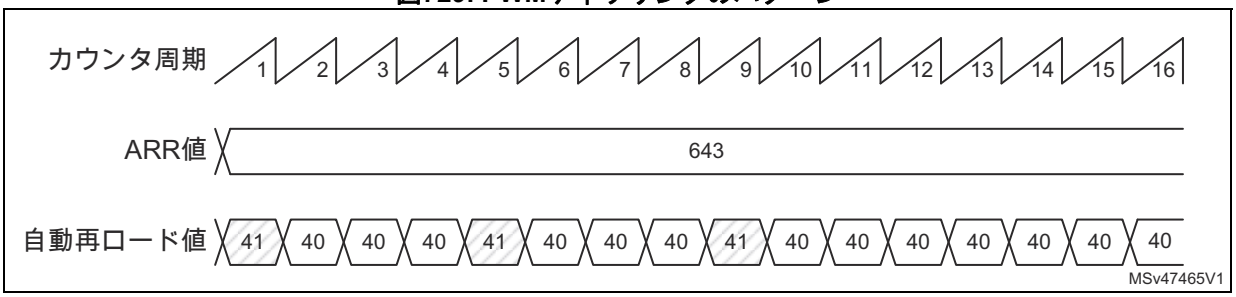
下記の図728に示すように、ディザリングモードはPWM周波数に関係なく、PWM分解能を増加するために使用されます。

図728. F_{Cnt} 分解能 対 周波数



下記の図729で説明するように、周期の変更は、連続した16周期にわたって広がります。

図729. PWMディザリングのパターン



自動再ロード値と比較値の増分は、下記の表588に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表588.TIMx_ARRレジスタの変更のディザリングパターン

-	PWM周期															
LSB値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

57.3.6 UIFビットの再配置

TIMx_CR1レジスタのIUFREMAPビットでは、タイマカウンタレジスタのビット31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これは、UIFCPYフラグによって信号伝達される潜在的なロールオーバー条件とカウンタ値を一度に読み取るために使用されます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

UIFとUIFCPYフラグのアサートの間には、遅延はありません。

57.3.7 ADCトリガ

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。

注： **tim_trgo信号を受信するスレーブペリフェラル（タイマ、ADCなど）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

57.3.8 TIM6/TIM7DMAリクエスト

表589に示すように、TIM6/TIM7 は単一のDMAリクエストを生成できます。

表589.DMAリクエスト

DMA項目（略称）	DMAリクエストあり	有効制御ビット
tim_upd_dma	更新	UDE

57.3.9 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M33 コアは停止状態）、TIMxカウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

57.3.10 TIM6/TIM7 低消費電力モード

表590.低消費電力モードが TIM6/TIM7 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスはSLEEPモードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

57.3.11 TIM6/TIM7 割込み

表591に示すように、TIM6/TIM7 は単一の割込みを生成できます。

表591.割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPおよびSTANDBYモードの終了
TIM6 TIM7	更新	UIF	UIE	UIFに0を書き込む	あり	不可

57.4 TIM6/TIM7 レジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ハーフワード（16bit）またはワード（32bit）単位でアクセスする必要があります。

57.4.1 TIMx制御レジスタ1（TIMx_CR1）（x = 6〜7）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw				rw				rw	rw	rw	rw

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DITHEN**：ディザリングイネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHENビットは、CENビットがリセットされているときのみ変更できます。

ビット11 **UIFREMAP**：UIFステータスビットの再配置

0：再配置なし。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされません。

1：再配置は有効です。UIFステータスビットはTIMx_CNTレジスタのビット31にコピーされます。

ビット10:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARRレジスタはバッファされません。

1：TIMx_ARRレジスタはバッファされます。

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **OPM**：ワンパルスモード。

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CENビットをクリア）。

ビット2 **URS**：更新リクエストソース

このビットは、UEVイベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたはDMAリクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー
– UGビットのセット
– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたはDMAリクエストを生成します（有効な場合）。

ビット1 **UDIS**：更新ディセーブル

このビットは、UEVイベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEVは有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UGビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEVは無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC）はそれぞれの値を維持します。ただし、UGビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

ワンパルスモードでは、更新イベントが発生すると、CENは自動的にクリアされます。

57.4.2 TIMx制御レジスタ2（TIMx_CR2）（x = 6～7）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS[2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **MMS[2:0]**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（TRGO）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット** - TIMx_EGRレジスタのUGビットがトリガ出力（tim_trgo）として使用されます。

001：**イネーブル** - カウンタ·イネーブル信号tim_cnt_enがトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、CEN制御ビットが書き込まれると生成されます。

010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

注： スレーブタイマまたはtim_trgoを受信するペリフェラルのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

57.4.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 6～7）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
							rw								rw

ビット15:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **UDE**：更新DMAリクエストイネーブル

0：更新DMAリクエストは無効です。

1：更新DMAリクエストは有効です。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

57.4.4 TIMxステータスレジスタ（TIMx_SR）（x = 6～7）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
															rc_w0

ビット15:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–TIMx_CR1レジスタのUDIS = 0の場合、カウンタのオーバーフローが発生したとき。

–TIMx_CR1レジスタのURS = 0かつUDIS = 0の場合に、TIMx_EGRレジスタのUGビットを使用して、ソフトウェアでCNTが再初期化されたとき。

57.4.5 TIMxイベント生成レジスタ（TIMx_EGR）（x = 6～7）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
															w

ビット15:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

57.4.6 TIMxカウンタ（TIMx_CNT）（x = 6～7）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UIFCPY**：UIFコピー

このビットはTIMx_ISRレジスタのUIFビットの読出し専用コピー。TIMx_CR1のUIFREMAPビットがリセットされると、ビット31は予約済みで、0で読み出されます。

ビット30:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分はありません。

57.4.7 TIMxプリスケーラ（TIMx_PSC）（x = 6～7）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数$f_{tim_cnt_ck}$は$f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSCには、更新イベントのたびにアクティブプリスケーラレジスタにロードされる値が含まれます。

（更新イベントには、TIMx_EGRレジスタのUGビットを通じて、カウンタがクリアされる場合も含まれます）。

57.4.8 TIMx自動再ロードレジスタ（TIMx_ARR）（x = 6〜7）

アドレスオフセット：0x2C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **ARR[19:0]**：自動再ロード値

ARRは、実際の自動再ロードレジスタにロードされる値です。

ARRの更新と動作の詳細については、セクション 57.3.4： 2431ページのタイムベースユニットを参照してください。

自動再ロード値がnullのときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットは予約されています。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

57.4.9 TIMxレジスタマップ

TIMxレジスタは、次の表のように、16bitアドレス可能レジスタとして配置されます。

表592.TIMxレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																				0	0				0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS [2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x08	予約済み																																
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
	リセット値																								0								0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
	リセット値																																0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
	リセット値																																0
0x18-0x20	予約済み																																
0x24	TIMx_CNT	UIFCPYまたはRes.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

58 低消費電力タイマ（LPTIM）

58.1 概要

LPTIMは、消費電力削減の究極的な進展を利用した16bitタイマです。クロックソースの多様性により、LPTIMはSTANDBYモードを除くすべての電力モードで実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIMは、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIMはシステムを低消費電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIMの柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

58.2 LPTIMの主な機能

- 16bitアップカウンタ
- 8つの分周比（1、2、4、8、16、32、64、128）を持つ3bitプリスケーラ
- 選択可能なクロック
  - 内部クロックソース：設定可能な内部クロックソース（RCCセクションを参照）
  - LPTIM入力経由の外部クロックソース（LPオシレータが実行していないときに動作、パルスカウンタアプリケーションによって使用）
- 16bit ARR自動再ロードレジスタ
- 16bitキャプチャ／比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能なI/O極性
- エンコーダモード
- 繰返しカウンタ
- 次の機能を持つ、最大2つの独立チャネル：
  - 入力キャプチャ
  - PWM生成（エッジアラインモード）
  - ワンパルスモード出力
- 10種のイベント時の割込み生成
- 以下のイベント時のDMAリクエスト生成：
  - 更新イベント
  - 入力キャプチャ

58.3 LPTIMの実装

下の表に、STM32U5シリーズ デバイスでのLPTIMの実装を示します。LPTIM1、LPTIM2およびLPTIM3には、その機能のすべてが実装されています。LPTIM4では、それよりも少ない機能がサポートされています。

表593.STM32U5シリーズ LPTIM機能

LPTIMのモード／機能(1)	LPTIM1	LPTIM2	LPTIM3	LPTIM4
エンコーダモード	X	X	-	-
PWMモード	X	X	X	X
入力キャプチャ	X	X	X	-
チャネル数	2	2	2	-
DMAリクエスト数	3	3	3	-
STOPモードからのウェイクアップ	X(2)	X(3)	X (2)	X (2)
自律モード	X	X	X	-

1. X : サポートされています。
2. STOP 0、STOP 1およびSTOP 2モードからサポートされるウェイクアップ。
3. STOP 0およびSTOP 1モードからサポートされるウェイクアップ。

58.4 LPTIMの機能説明

58.4.1 LPTIMブロック図

図730. LPTIM1/2/3タイマブロック図

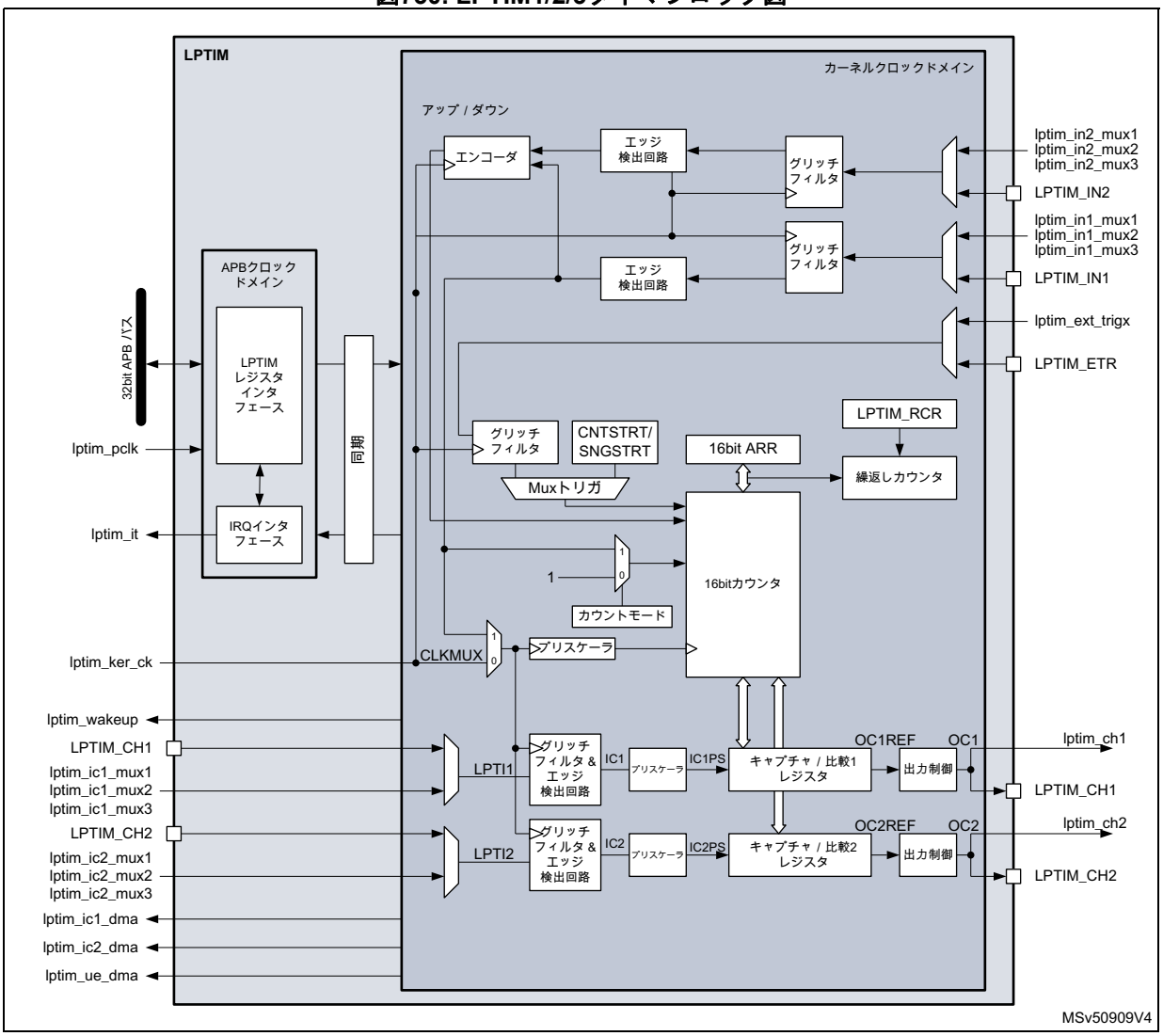
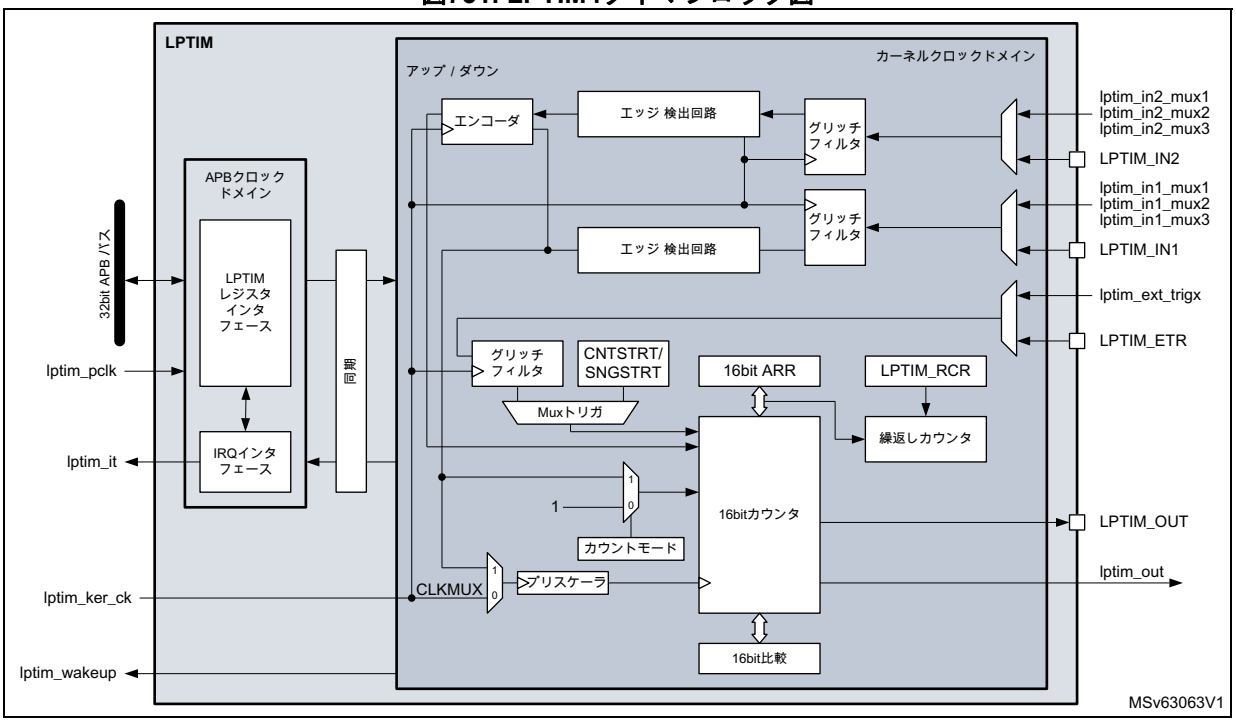


図731. LPTIM4タイマブロック図



58.4.2 LPTIMピンおよび内部信号

次の表に、LPTIMのピンと内部信号の一覧を示します。

表594.LPTIM1/2/3の入出力ピン

名前	信号タイプ	説明
LPTIM_IN1	デジタル入力	マルチプレクサ入力0のLPTIMx_IN1ピンからのLPTIM入力1
LPTIM_IN2(1)	デジタル入力	マルチプレクサ入力0のLPTIMx_IN2ピンからのLPTIM入力2
LPTIM_ETR	デジタル入力	LPTIM外部トリガLPTIMx_ETRピン
LPTIM_CH1	デジタル入出力	LPTIMチャネル1入出力LPTIMx_IN1ピン
LPTIM_CH2	デジタル入出力	LPTIMチャネル2入出力LPTIMx_IN2ピン

1. LPTIM3には入力1のみがあります（入力2はありません）。

表595.LPTIM4の入出力ピン

名前	信号タイプ	説明
LPTIM_IN1	デジタル入力	マルチプレクサ入力0のLPTIMx_IN1ピンからのLPTIM入力1
LPTIM_ETR	デジタル入力	LPTIM外部トリガLPTIMx_ETRピン
LPTIM_OUT	デジタル出力	LPTIM出力LPTIMx_OUTピン

表596.LPTIM1/2/3内部信号

名前	信号タイプ	説明
lptim_pclk	デジタル入力	LPTIM APBクロックドメイン
lptim_ker_ck	デジタル入力	LPTIMカーネルクロック
lptim_in1_mux1	デジタル入力	マルチプレクサ入力1に接続される内部LPTIM入力1
lptim_in1_mux2	デジタル入力	マルチプレクサ入力2に接続される内部LPTIM入力1
lptim_in1_mux3	デジタル入力	マルチプレクサ入力3に接続される内部LPTIM入力1
lptim_in2_mux1(1)	デジタル入力	マルチプレクサ入力1に接続される内部LPTIM入力2
lptim_in2_mux2（1)	デジタル入力	マルチプレクサ入力2に接続される内部LPTIM入力2
lptim_in2_mux3（1)	デジタル入力	マルチプレクサ入力3に接続される内部LPTIM入力2
lptim_ic1_mux1	デジタル入力	マルチプレクサ入力1に接続される内部LPTIM入力キャプチャ 1
lptim_ic1_mux2	デジタル入力	マルチプレクサ入力2に接続される内部LPTIM入力キャプチャ 1
lptim_ic1_mux3	デジタル入力	マルチプレクサ入力3に接続される内部LPTIM入力キャプチャ 1
lptim_ic2_mux1	デジタル入力	マルチプレクサ入力1に接続される内部LPTIM入力キャプチャ 2
lptim_ic2_mux2	デジタル入力	マルチプレクサ入力2に接続される内部LPTIM入力キャプチャ 2
lptim_ic2_mux3	デジタル入力	マルチプレクサ入力3に接続される内部LPTIM入力キャプチャ 2
lptim_ext_trigx	デジタル入力	LPTIM外部トリガ入力x
lptim_it	デジタル出力	LPTIMグローバル割込み
lptim_wakeup	デジタル出力	LPTIMウェイクアップイベント
lptim_ic1_dma	デジタル出力	LPTIM入力キャプチャ 1 DMAリクエスト
lptim_ic2_dma	デジタル出力	LPTIM入力キャプチャ 2 DMAリクエスト
lptim_ue_dma	デジタル出力	LPTIM更新イベントのDMAリクエスト

1. LPTIM3には入力1のみがあります（入力2はありません）。

表597.LPTIM4内部信号

名前	信号タイプ	説明
lptim_pclk	デジタル入力	LPTIM APBクロックドメイン
lptim_ker_ck	デジタル入力	LPTIMカーネルクロック
lptim_in1_mux1	デジタル入力	マルチプレクサ入力1に接続される内部LPTIM入力1
lptim_in1_mux2	デジタル入力	マルチプレクサ入力2に接続される内部LPTIM入力1
lptim_in1_mux3	デジタル入力	マルチプレクサ入力3に接続される内部LPTIM入力1
lptim_ext_trigx	デジタル入力	LPTIM外部トリガ入力x
lptim_out	デジタル出力	LPTIMカウンタ出力
lptim_it	デジタル出力	LPTIMグローバル割込み
lptim_wakeup	デジタル出力	LPTIMウェイクアップイベント

58.4.3 LPTIM入力およびトリガマッピング

LPTIM外部トリガおよび入力の接続について、以下に詳しく説明します。

表598.LPTIM1/2/3/4外部トリガ接続

TRIGSEL	外部トリガ			
	LPTIM1	LPTIM2	LPTIM3	LPTIM4
lptim_ext_trig0	GPIO			
lptim_ext_trig1	rtc_alra_trg			
lptim_ext_trig2	rtc_alrb_trg			
lptim_ext_trig3	tamp_trg1			lpdma_ch1_tc
lptim_ext_trig4	tamp_trg2	gpdma_ch0_tc	lpdma_ch1_tc	tamp_trg2
lptim_ext_trig5	lpdma_ch2_tc	gpdma_ch4_tc	tamp_trg3	tamp_trg3
lptim_ext_trig6	comp1_out			
lptim_ext_trig7	comp2_out(1)			

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表599.LPTIM1/2/3/4入力1接続

lptim_in1_mux	LPTIM1/2/3/4の入力1の接続先
lptim_in1_mux0	GPIO
lptim_in1_mux1	comp1_out
lptim_in1_mux2	-
lptim_in1_mux3	-

表600.LPTIM1/2入力2接続

lptim_in2_mux	LPTIM1/2の入力2の接続先
lptim_in2_mux0	GPIO
lptim_in2_mux1	comp2_out(1)
lptim_in2_mux2	-
lptim_in2_mux3	-

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表601.LPTIM1/2/3入力キャプチャ 1接続

lptim_ic1_mux	LPTIM1/2/3入力キャプチャ 1の接続先
lptim_ic1_mux0	GPIO
lptim_ic1_mux1	comp1_out
lptim_ic1_mux2	comp2_out(1)
lptim_ic1_mux3	-

1. COMP2が使用できないため、この接続はSTM32U535/545には存在しません。

表602.LPTIM1入力キャプチャ 2接続

lptim_ic2_mux	LPTIM1入力キャプチャ 2の接続先
lptim_ic2_mux0	I/O
lptim_ic2_mux1	LSI
lptim_ic2_mux2	LSE
lptim_ic2_mux3	-

表603.LPTIM2入力キャプチャ 2接続

lptim_ic2_mux	LPTIM2入力キャプチャ 2の接続先
lptim_ic2_mux0	I/O
lptim_ic2_mux1	HSI/256
lptim_ic2_mux2	MSIS/1024
lptim_ic2_mux3	MSIS/4

表604.LPTIM3入力キャプチャ 2接続

lptim_ic2_mux	LPTIM3入力キャプチャ 2の接続先
lptim_ic2_mux0	I/O
lptim_ic2_mux1	-
lptim_ic2_mux2	-
lptim_ic2_mux3	-

58.4.4 LPTIMのリセットとクロック

LPTIMのクロックには、いくつかのクロックソースを使用できます。RCCを介して選択できる任意の設定可能な内部クロックソースになることができる内部クロック信号を使用してクロックを供給できます（詳細については、RCCのセクションを参照してください）。また、LPTIMのクロックには、外部Input1に入力された外部クロック信号を使用することもできます。外部クロックソースでクロック供給されるときには、LPTIMは、次の2つの構成のいずれかで動作します。

- 最初の構成では、LPTIMのクロックは外部信号によって供給されますが、同時に内部クロック信号も、設定可能な内部クロックソースからLPTIMに供給されます（RCCのセクションを参照）。
- 2番目の構成では、LPTIMのクロックは外部Input1を通じて外部クロックソースによってのみ供給されます。この構成は、低消費電力モードになった後、すべての埋め込みオシレータがオフになるときに、タイムアウト機能またはパルスカウンタ機能を実現するために使用されます。

CKSELビットおよびCOUNTMODEビットに書き込むことによって、LPTIMが外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOLビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給する必要があります（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の4倍以上である必要があります。

58.4.5 グリッチフィルタ

LPTIM入力の外部入力（GPIOにマップされたもの）または内部入力（チップレベルで他の内蔵ペリフェラルにマップされたもの）はデジタルフィルタによって保護され、グリッチとノイズの影響がLPTIM内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースをLPTIMに供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

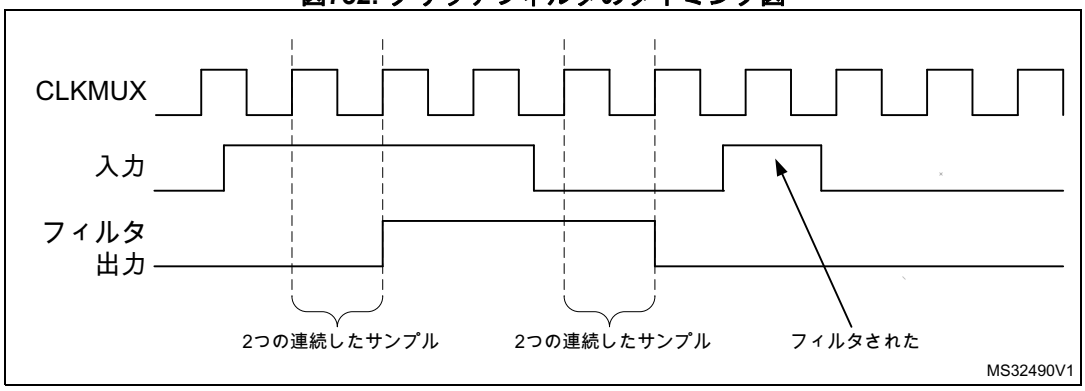
デジタルフィルタは、3 つのグループに分けられます。

- 最初のグループのデジタルフィルタは、LPTIM の内部または外部入力を保護します。デジタルフィルタの期間はCKFLTビットによって制御されます。
- 2番目のグループのデジタルフィルタは、LPTIMの内部または外部トリガ入力を保護します。デジタルフィルタの期間はTRGFLTビットによって制御されます。
- 3番目のグループのデジタルフィルタは､LPTIMの内部または外部入力キャプチャを保護します。デジタルフィルタの期間はICxFビットによって制御されます。

注： **デジタルフィルタの期間は 、グループごとに制御されます。同じグループ内の各デジタルフィルタの期間を個別に設定することはできません。**

フィルタの期間は、信号のレベル変化を有効な遷移とみなすために、LPTIM入力の1つで検出される連続した等しいサンプルの数に基づいて決められます。図732に、2つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図732. グリッチフィルタのタイミング図



注： **内部クロック信号が供給されない場合は、CKFLT、ICxF、およびTRGFLTビットを0にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIMの外部入力をグリッチから保護できます。**

58.4.6 プリスケーラ

LPTIM 16bitカウンタの前には、設定可能な2のべき乗プリスケーラがあります。プリスケーラの分周比はPRESC[2:0] 3bitフィールドによって制御されます。下の表に、可能な分周比を示します。

表605.プリスケーラ分周比

プログラミング	分周比
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

58.4.7 トリガマルチプレクサ

LPTIMカウンタは、ソフトウェアによって、または 8 つのトリガ入力の1つのアクティブエッジの検出後に開始できます。

LPTIMのトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が00の場合、LPTIMカウンタは、CNTSTRTまたはSNGSTRTビットがソフトウェアによってセットされるとすぐに開始します。TRIGEN[1:0] の残りの3つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIMカウンタは、アクティブエッジが検出されるとすぐに開始します。
- TRIGEN[1:0] が00以外のときには、カウンタの開始に使用される8 つのトリガ入力の1つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIMの場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに2カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： **SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。**

注： **ソフトウェアによってカウンタを開始する場合（TRIGEN[1:0] = 00）、LPTIM_CRレジスタが更新（SNGSTRTまたはCNTSTRTビットのいずれかをセット）されてからカウンタが有効に開始されるまで3カーネルクロックサイクルの遅延が発生します。**

58.4.8 動作モード

LPTIMには2つの動作モードがあります。

- 連続モード : タイマはフリーランニングし、トリガイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード : タイマはトリガイベントによって開始し、LPTIM更新イベントが生成されると停止します。

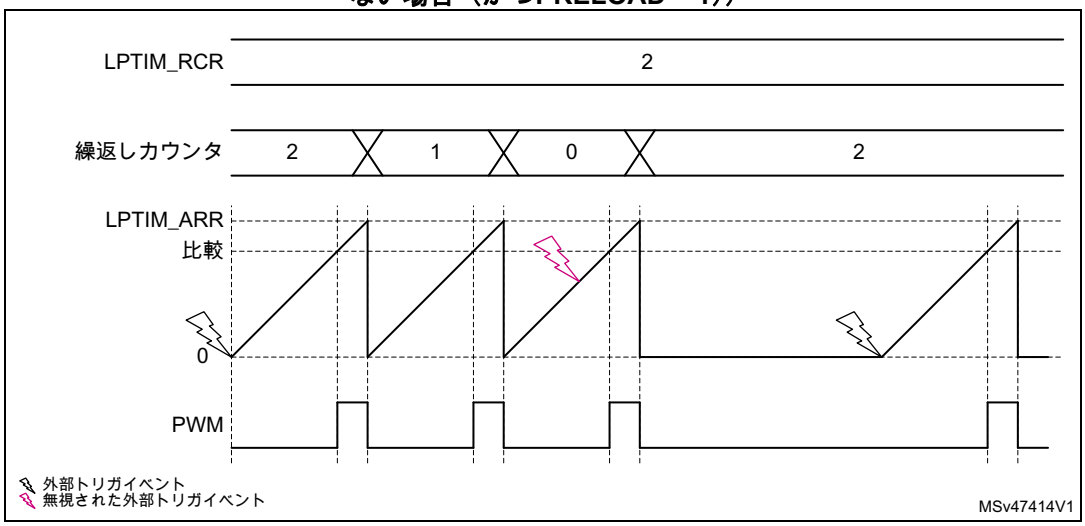
ワンショットモード

ワンショットカウントを有効にするには、SNGSTRTビットをセットする必要があります。

新しいトリガイベント発生はタイマを再開します。カウンタが開始してから、次のLPTIM更新イベントまでに発生したトリガイベントは破棄されます。

外部トリガが選択された場合、SNGSTRTビットがセットされた後で外部トリガイベントを受信し、繰返しカウンタが停止した後（更新イベント後）、かつ繰返しレジスタの内容が0ではない場合、繰返しカウンタは繰返しレジスタによってすでに入力された値を再ロードし、新たなワンショットカウントサイクルを開始します（図733を参照）。

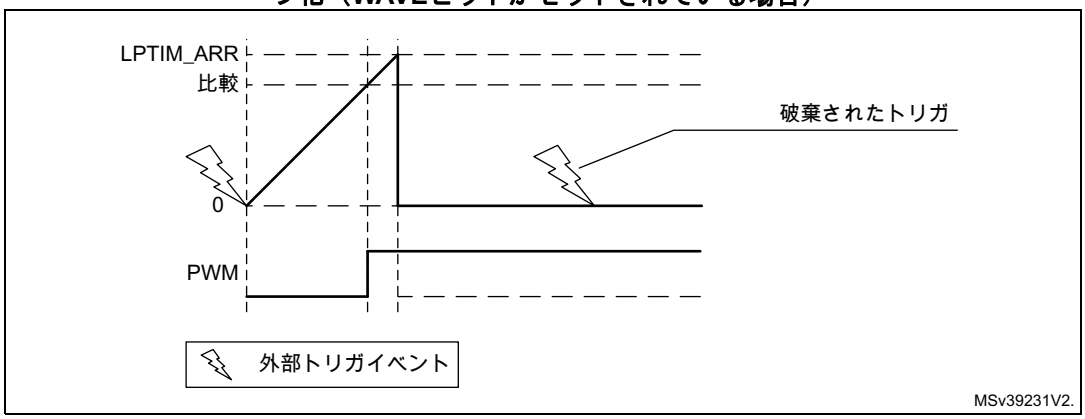
図733. LPTIM出力波形、シングルカウントモードの設定（繰返しレジスタの内容が0ではない場合（かつPRELOAD = 1））



- セットワンスモードのアクティブ化 :

LPTIM_CFGRレジスタのWAVEビットフィールドがセットされると、セットワンスモードがアクティブになることに注意してください。この場合、最初のトリガに続いてカウンタが一度だけ開始され、その後発生したトリガイベントはすべて破棄されます（図734を参照）。

図734. LPTIM出力波形、シングルカウントモードの設定、セットワンスモードのアクティブ化（WAVEビットがセットされている場合）



ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、SNGSTRTをセットすると、カウンタはワンショットカウントを開始します。

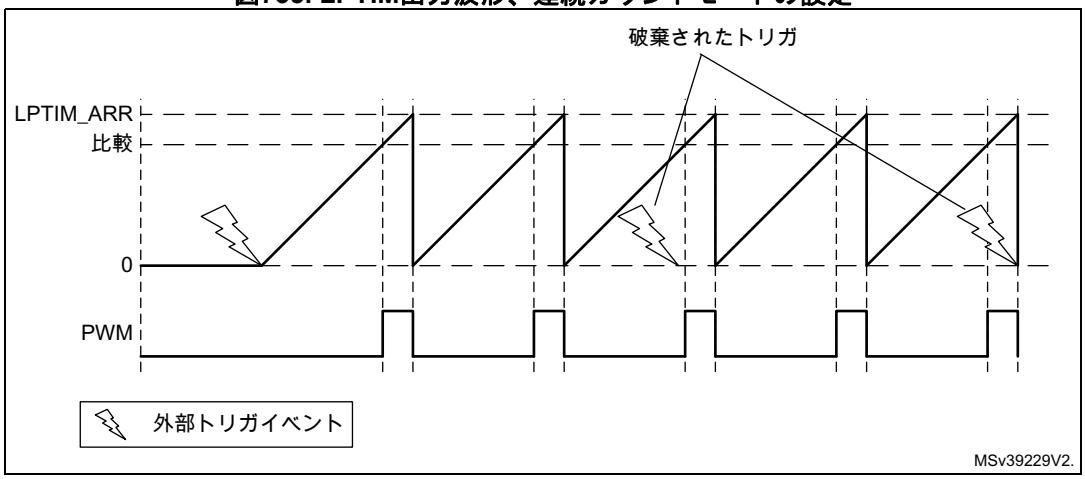
連続モード

連続カウントを有効にするには、CNTSTRTビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRTがセットされた後に外部トリガイベントが着信すると、カウンタは連続カウントを開始します。その後発生した外部トリガイベントはすべて、図735に示すように破棄されます。

ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、CNTSTRTをセットすると、カウンタは連続カウントを開始します。

図735. LPTIM出力波形、連続カウントモードの設定



SNGSTRTおよびCNTSTRTビットは、タイマが有効なときのみ（イネーブルビットが1にセットされている）、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRTをセットすると、LPTIMはワンショットモードに切り替わります。カウンタ（アクティブな場合）は、LPTIM更新イベントが生成されるとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRTをセットすると、LPTIMは連続モードに切り替わります。カウンタ（アクティブな場合）は、ARRに達するとすぐに再開始します。

58.4.9 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIMカウンタをリセットできます。この機能は、TIMOUTビットで制御されます。

最初のトリガイベントでタイマが開始し、その後のトリガイベントでLPTIMカウンタと繰返しカウンタがリセットされ、タイマが再開始します。

低消費電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによってMCUがウェイクアップします。

58.4.10 波形生成

2つの16bitレジスタLPTIM_ARR（自動再ロードレジスタ）と LPTIM_CCRx（キャプチャ／比較レジスタ）は、LPTIM出力にいくつかの異なった波形を生成するために使用されます。

タイマは次の波形を生成できます。

- PWMモード：LPTIM出力は、LPTIM_CNTのカウンタ値がLPTIM_CCRxの比較値を超えるとすぐにセットされます。LPTIM出力は、LPTIM_ARRレジスタとLPTIM_CNTレジスタの間で一致が発生するとすぐにリセットされます。詳細については、セクション 58.4.19：PWMモードを参照してください。
- ワンパルスモード：出力波形は、最初のパルスについてはPWMモードの波形と同様であり、その後、出力はリセットされたままです。
- セットワンスモード：出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます（設定された出力極性に応じて）。

上記のモードでは、LPTIM_ARRレジスタの値は LPTIM_CCRx レジスタの値より厳密に大きい必要があります。

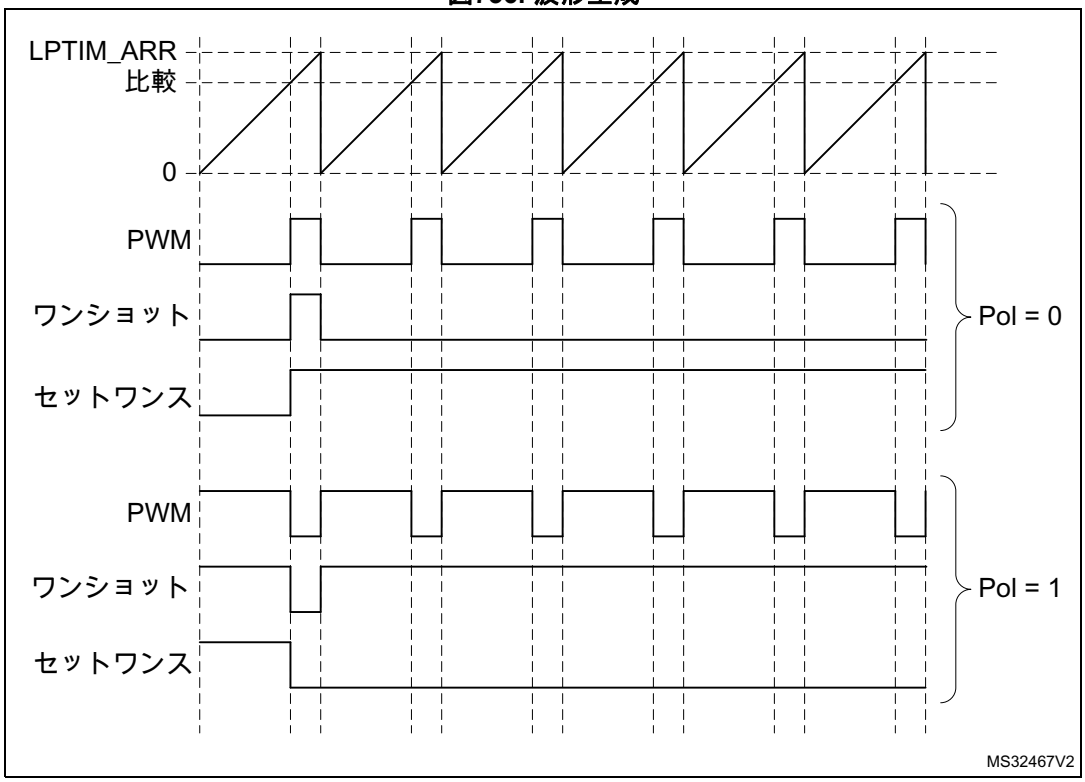
LPTIM出力波形は、WAVEビットによって次のように設定できます。

- WAVEビットを0にリセットすると、LPTIMはCNTSTRTまたはSNGSTRTのセットされているビットに応じて、PWM波形またはワンパルス波形のいずれかを生成します。
- WAVEビットを1にセットすると、LPTIMはセットワンスモード波形を生成します。

WAVPOL/CCxP ビットは、LPTIM出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力のデフォルト値はただちに変更されます。

LPTIMクロック周波数を2分周した周波数までの信号を生成できます。図736に、LPTIM出力で生成できる3つの波形を示します。また、WAVPOL/CCxP ビットを使用して極性を変更したときの効果も示します。

図736. 波形生成



58.4.11 レジスタの更新

LPTIM_ARRレジスタ、LPTIM_RCRレジスタ、LPTIM_CCRxレジスタは、APBバス書込み操作の直後、またはタイマがすでに開始していた場合は次のLPTIM更新イベントでの同期時に更新されます。

PRELOADビットによって、LPTIM_ARR 、LPTIM_RCR、およびLPTIM_CCRxレジスタの更新方法を制御します。

- PRELOADビットを“0”にリセットすると、LPTIM_ARRレジスタ、LPTIM_RCRレジスタ、およびLPTIM_CCRxレジスタは書込みアクセスの直後に更新されます。
- PRELOADビットを“1”にセットすると、LPTIM_ARRレジスタ、LPTIM_RCRレジスタ、およびLPTIM_CCRxレジスタは、次のLPTIM更新イベント時に更新されます（タイマがすでに開始していた場合）。

LPTIM APBインタフェースとLPTIMカーネルロジックは異なるクロックを使用するので、APB書込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延時間内は、これらのレジスタに追加で書き込めません。

LPTIM_ISRレジスタのARROKフラグ、REPOKフラグ、CMPxOKフラグは、それぞれ、LPTIM_ARRレジスタ、LPTIM_RCRレジスタ、LPTIM_CCRxレジスタへの書込み操作が完了したことを示します。

LPTIM_ARR、LPTIM_RCR、またはLPTIM_CCRxレジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。ARROKフラグ、REPOKフラグ、またはCMPxOKフラグがそれぞれセットされる前に、続けて書込みを行うと、予測不能な結果になります。

58.4.12 カウンタモード

LPTIMカウンタを使用して、LPTIM Input1の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。CKSELビットおよびCOUNTMODEビットは、カウンタの更新にどのソースを使用するかを決定します。

LPTIMがInput1の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立上がりエッジ、立下がりエッジ、または両方のエッジで更新できます。

CKSELおよびCOUNTMODEの値に応じて、以下に示すカウントモードを選択できます。

- CKSEL = 0：LPTIMのクロックは、内部クロックソースによって供給されます。
  - COUNTMODE = 0

    LPTIMが内部クロックソースによってクロック供給されるように設定され、LPTIMカウンタが各内部クロックパルス後に更新されるように設定されます。
  - COUNTMODE = 1

    LPTIM外部Input1は、LPTIMに供給される内部クロックでサンプリングされます。

    したがって、イベントをミスしないためには、外部Input1信号の変化の周波数が、LPTIMに供給される内部クロックの周波数を超えてはなりません。また、LPTIMに供給される内部クロックを分周してはなりません（PRESC[2:0] = 000）。
- CKSEL = 1：LPTIMのクロックは、外部クロックソースによって供給されます。

  COUNTMODEの 値は無視されます。

  この構成では、LPTIMは内部クロックソースを必要としません（グリッチフィルタ有効である場合を除く）。LPTIM外部Input1に入力された信号がLPTIMのシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。

  この構成の場合、LPTIMカウンタは、Input1クロック信号の立上がりエッジまたは立下がりエッジで更新できますが、立上がりと立下がりの両方のエッジで更新することはできません。

  LPTIM外部Input1に入力された信号はLPTIMカーネルロジックのクロック動作にも使用されるので、カウンタがインクリメントされる前（LPTIMが有効にされた後）、初期遅延があります。より正確には、LPTIM外部Input1の（LPTIMが有効になった後の）最初の5つのアクティブエッジは失われます。

58.4.13 タイマ有効

LPTIM_CRレジスタのイネーブルビットは、LPTIMカーネルロジックを有効化／無効化するために使用されます。イネーブルビットをセットした後、LPTIMが実際に有効になるまで、2カウンタクロックの遅延が必要です。

LPTIM_CFGRレジスタの変更は、LPTIMが無効なときにのみ行う必要があります。

58.4.14 タイマカウンタのリセット

LPTIM_CNTレジスタの内容をゼロにリセットするために、以下の2つのリセットメカニズムが実装されています。

- 同期リセットメカニズム：同期リセットは、LPTIM_CRレジスタのCOUNTRSTビットによって制御されます。COUNTRSTビットフィールドを“1”にセットした後、リセット信号がLPTIMカーネルクロックドメイン内に伝播されます。したがって、リセットが考慮される前に、LPTIMカーネルロジックの若干のクロックパルスが経過することに注意することが重要です。このため、リセットがトリガされてから有効になるまでの間に、LPTIMカウンタは若干の余分なパルスをカウントすることになります。COUNTRSTビットはAPBクロックドメインに位置し、LPTIMカウンタはLPTIMカーネルクロックドメインに位置しているので、COUNTRSTビットに“1”を書き込んだ時にAPBクロックによって発行されるリセット信号に同期するのに、カーネルクロックの3クロックサイクルの遅延が必要となります。
- 非同期リセットメカニズム：非同期リセットは、LPTIM_CR レジスタにある RSTARE ビットによって制御されます。このビットが“1”にセットされると、LPTIM_CNTレジスタに対するあらゆる読出しアクセスによって、その内容がゼロにリセットされます。非同期リセットは、LPTIMコアクロックが提供されない時間枠内にトリガする必要があります。たとえば、LPTIM入力1が外部クロックソースとして使用されているとき、非同期リセットは、LPTIM入力1に反転が起きないことが十分確保されるときにのみ適用する必要があります。
  LPTIM_CNTレジスタの内容を信頼できるように読み出すためには、2回の連続した読出しアクセスを実行して比較する必要があることに注意してください。2回の読出しアクセスで得られた値が同じであるとき、各読出しアクセスは信頼できると考えられます。しかし残念ながら、非同期リセットが有効になっているとき、LPTIM_CNTレジスタを2回読み出すことはできません。

警告： **LPTIM内部には、2つのリセットメカニズムを同時に使用することを防ぐメカニズムはありません。したがって、開発者はこれらの2つのメカニズムを排他的に使用することを確実にする必要があります。**

58.4.15 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが0とLPTIM_ARRレジスタでプログラムされた自動再ロード値の間で（方向に応じて、0からARRまで、またはARRから0まで）連続的にカウントすることを意味します。したがって、カウンタを開始する前にLPTIM_ARRを設定する必要があります。Input1とInput2の2つの外部入力信号から、LPTIMカウンタのクロックのためのクロック信号が生成されます。この2つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIMが内部クロックソースからクロック供給されるときにのみ使用できます。Input1とInput2の両方の信号周波数は、LPTIM内部クロック周波数を4分周したものを超えてはなりません。これは、LPTIMの正しい動作を保証するために必要です。

方向の変更は、LPTIM_ISR レジスタのDownとUpの2つのフラグによって通知されます。また、DOWNIEビットを通じて有効化された場合、両方の方向変更イベントで割込みを生成できます。

エンコーダモードを有効にするには、ENCビットを1にセットする必要があります。LPTIMを、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIMカウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウント方向は、UpおよびDownフラグによって通知され、エンコーダのロータの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ検出に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組合せを示します（Input1とInput2は同時に切り替わらないと想定しています）。

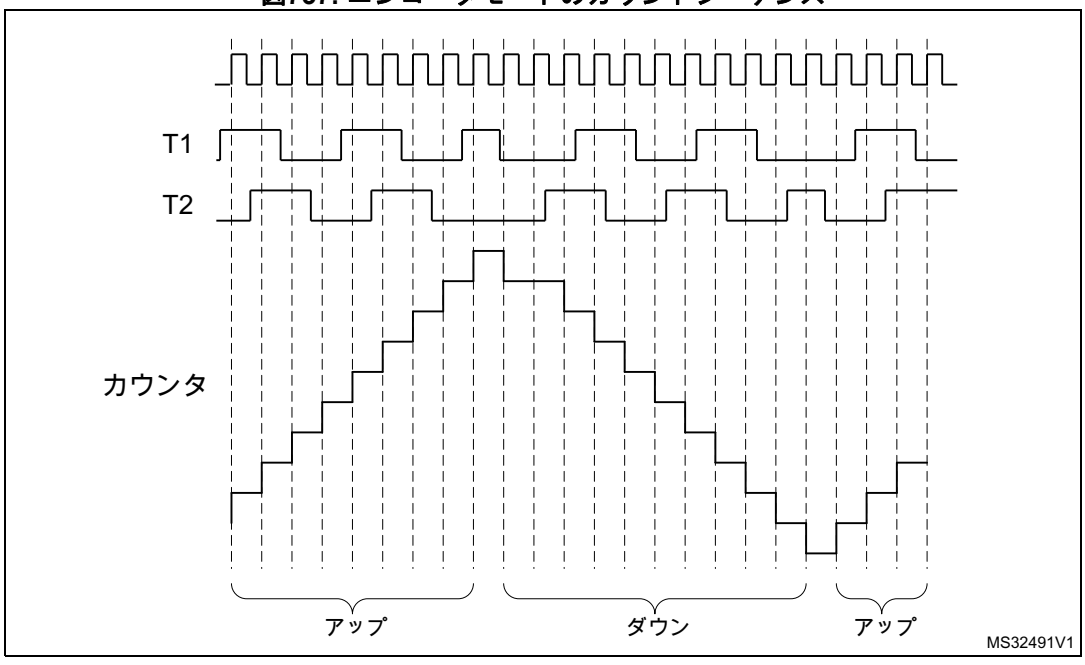
表606.エンコーダのカウントシナリオ

アクティブエッジ	他方の信号のレベル（Input2に対するInput1、Input1に対するInput2）	Input1信号		Input2信号	
		立ち上がり	立下がり	立上がり	立下がり
立上がりエッジ	高	ダウン	カウントなし	アップ	カウントなし
	低	アップ	カウントなし	ダウン	カウントなし
立下がりエッジ	高	カウントなし	アップ	カウントなし	ダウン
	低	カウントなし	ダウン	カウントなし	アップ
両エッジ	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

次の図に、両方のエッジ検出が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： **このモードでは、LPTIMのクロックは内部クロックソースによって供給される必要があるため、CKSELビットをリセット値（0）に維持する必要があります。また、プリスケーラの分周比はリセット値である1に等しくなければなりません（PRESC[2:0] ビットが000である必要があります）。**

図737. エンコーダモードのカウントシーケンス



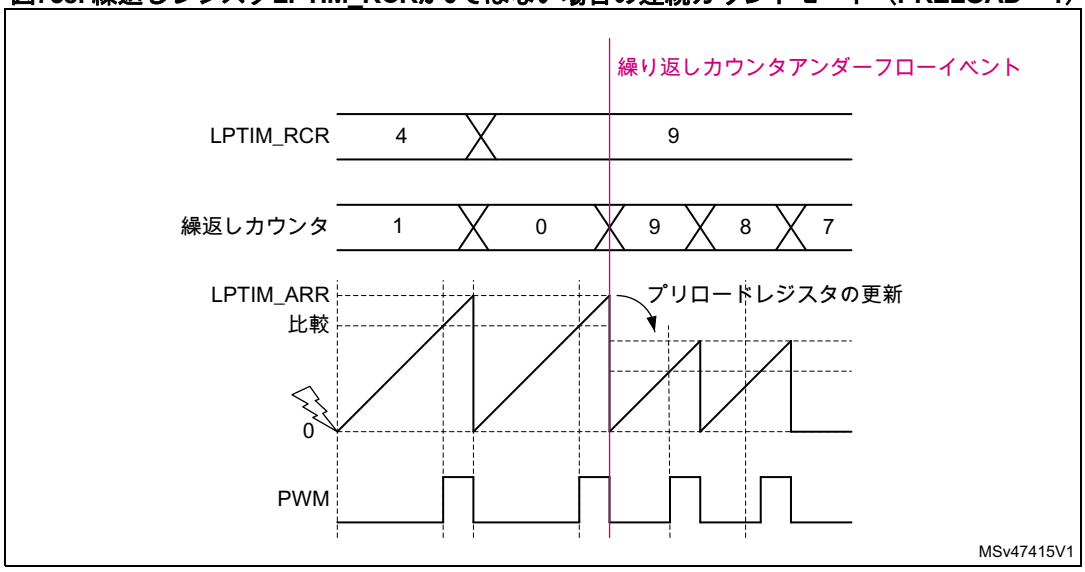
58.4.16 繰返しカウンタ

LPTIMには、LPTIMカウンタオーバーフローイベントが発生するたびに1ずつデクリメントする繰返しカウンタが搭載されています。繰返しカウンタアンダーフローイベントは、繰返しカウンタに0が含まれ、LPTIMカウンタがオーバーフローすると生成されます。各繰返しカウンタアンダーフローイベントの後、繰返しカウンタには、繰返しレジスタLPTIM_RCRに属するREP[7:0] ビットフィールドの内容がロードされます。

繰返しアンダーフローイベントは、REP[7:0] レジスタが0にセットされる場合、各LPTIMカウンタオーバーフロー時に生成されます。

PRELOAD = 1の場合、REP[7:0] ビットフィールドに書き込んでも、次の繰返しアンダーフローイベントが発生するまで繰返しカウンタの内容に影響しません。繰返しカウンタが各LPTIMカウンタオーバーフローイベントでデクリメントし続け、繰返しアンダーフローイベントが生成された場合のみ、REP[7:0] に書き込まれた新しい値が繰返しカウンタにロードされます。この動作を図738 に示します。

図738. 繰返しレジスタLPTIM_RCRが0ではない場合の連続カウントモード（PRELOAD = 1）



繰返しカウンタアンダーフローイベントは、LPTIMプリロードレジスタの更新と体系的に関連付けられています（詳細はセクション「レジスタの更新」を参照）。

繰返しカウンタアンダーフローイベントは、LPTIM_ISRレジスタに配置された更新イベント（UE）フラグでソフトウェアに通知されます。LPTIM_DIERレジスタに配置された、それぞれの更新イベント割込みイネーブル（UEIE）制御ビットがセットされていれば、UEフラグがセットされたとき、LPTIM割込みがトリガされます。

繰返しレジスタLPTIM_RCRはAPBバスインタフェースクロックドメインにあり、繰返しカウンタ自体はLPTIMカーネルクロックドメインにあります。新しい値がLPTIM_RCRレジスタに書き込まれるたびに、新しい内容がAPBバスインタフェースクロックドメインからLPTIMカーネルクロックドメインに伝播され、新たに書き込まれた値は繰返しカウンタアンダーフローイベント直後に繰返しカウンタにロードされます。新たに書き込まれた内容の同期遅延は4 APBクロックサイクルと3 LPTIMカーネルクロックサイクルとなり、経過するとLPTIM_ISRレジスタにあるREPOKフラグによって通知されます。LPTIMカーネルがLSIクロックソースからクロック供給されている場合など、LPTIMカーネルクロックサイクルが比較的遅くなる場合、LPTIM_RCRレジスタの内容の同期が完了したことをソフトウェアで検出する REPOK フラグでしばらくポーリングすることができます。そのため、LPTIM_DIERレジスタの関連するREPOKIE制御ビットをセットしておけば、REPOKフラグがセットされた時に割込みを生成することができます。

注： **LPTIM_RCR レジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。REPOKフラグがセットされる前に続けて書込みを行うと、予測不能な結果になります。**

注意： **PRELOAD = 0で繰返しカウンタを使用する場合、自動再ロード一致イベントの少なくとも5カウンタサイクル前にLPTIM_RCRレジスタを変更する必要があります。そうしないと、予測不能な動作が発生することがあります。**

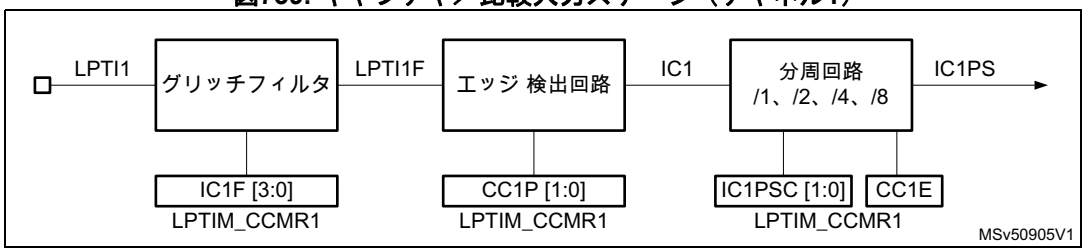
58.4.17 キャプチャ／比較チャネル

各キャプチャ /比較チャネルは、PWM用の、キャプチャ /比較レジスタ、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクサ、プリスケーラ）、および出力ステージ（比較回路と出力制御）から構成されています。

入力ステージ

入力ステージは、対応するLPTIx入力をサンプリングして、フィルタリングを行った信号LPTIxFを生成します。次に、極性選択付きのエッジ検出回路が、キャプチャコマンドとして使用されるICx信号を生成します。この信号はプリスケーラを通じて、キャプチャコマンド信号（ICxPS）を生成します。

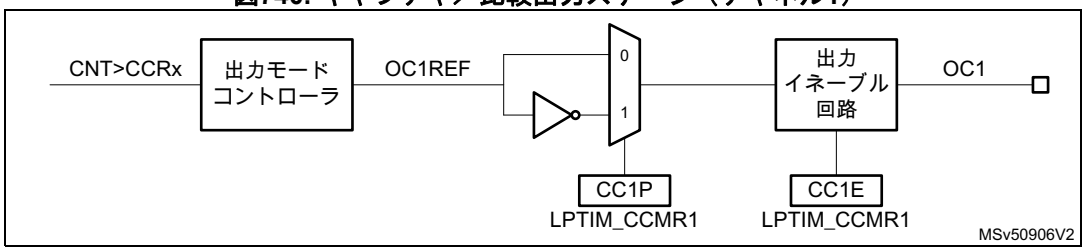
図739. キャプチャ／比較入力ステージ（チャネル1）



出力ステージ

出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成しますOCxREF（アクティブハイ）信号の極性は最終出力に影響を与えます。

図740. キャプチャ／比較出力ステージ（チャネル1）



58.4.18 入力キャプチャモード

入力キャプチャモードでは、対応するICx信号によって遷移が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（LPTIM_CCRx）が使用されます。チャネルxで入力キャプチャが有効（CCxE セット）になっていて、キャプチャが発生すると、対応する CCxIF フラグ（LPTIM_ISRレジスタ）がセットされ、割込みまたはDMAリクエストが送信されます（有効な場合）。CCxIFフラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグCCxOF（LPTIM_ISRレジスタ）がセットされます。CCxIFフラグは、ソフトウェアでCCxICFに1を書き込むことによって、または、LPTIM_CCRxレジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOFは、CCxOCFに1を書き込むことによってクリアされます。

注： **DMAモードでは、IC DMAリクエストを有効にした後に、カウンタを開始した後、最後に入力キャプチャチャネルを有効にする（CCxEビットをセットする）必要があります。これは、カウンタがまだ開始されていないときに、入力キャプチャのDMAリクエストが発生するのを防ぐためです。**

入力キャプチャのグリッチフィルタの遅延

トリガーイベントがチャネルx入力（LPTIx）に到着する時、設定されたグリッチフィルタ（CCMRxレジスタのICxF[1:0] フィールド）とカーネルクロックプレスケーラ値（CFGRレジスタのPRESC[2:0] フィールド）に応じて、可変の遅延が発生します。これによって、CCRxレジスタに格納されたキャプチャ値とキャプチャトリガに対応した実際の値との間に系統的なオフセット（表607を参照）が生じます。

このオフセットは系統的であって、2つのキャプチャ間で補正されるため、パルス幅測定には影響しません。

入力キャプチャトリガに対応する実際のキャプチャ値は、以下の式を使用して計算できます。

実際のキャプチャ値 = キャプチャされた（LPTIM_CCRx）値 - オフセット

グリッチフィルタとカーネルクロックのプリスケーラー値（CFGRレジスタのPRESCフィールド）に応じて、関連するオフセットを使用する必要があります。

例：PRESC[2:0] = 0x2でICxF = 0x3の場合の実際のキャプチャ値の決定

この設定（PRESC[2:0] = 0x2でICxF = 0x3）の場合、表607によれば、オフセットは5です。

CCRxでキャプチャされた値が9（LPTIM_CNT = 9）であるとすれば、これは、LPTIM_CNTが9 - 5 = 4に等しかったときにキャプチャトリガが発生したことになります。

表607.入力キャプチャのグリッチフィルタの遅延（カウンタステップ単位）

プリスケーラのPRESC[2:0]	ICxF[1:0]	オフセット
0	0	2
	1	7
	2	9
	3	13
1	0	3
	1	5
	2	6
	3	8
2	0	2
	1	3
	2	4
	3	5
3	0	2
	1	2
	2	3
	3	3

表607.入力キャプチャのグリッチフィルタの遅延（カウンタステップ単位）

プリスケーラのPRESC[2:0]	ICxF[1:0]	オフセット
4	0	2
	1	2
	2	2
	3	2
5	0	2
	1	2
	2	2
	3	2
6	0	2
	1	2
	2	2
	3	2
7	0	2
	1	2
	2	2
	3	2

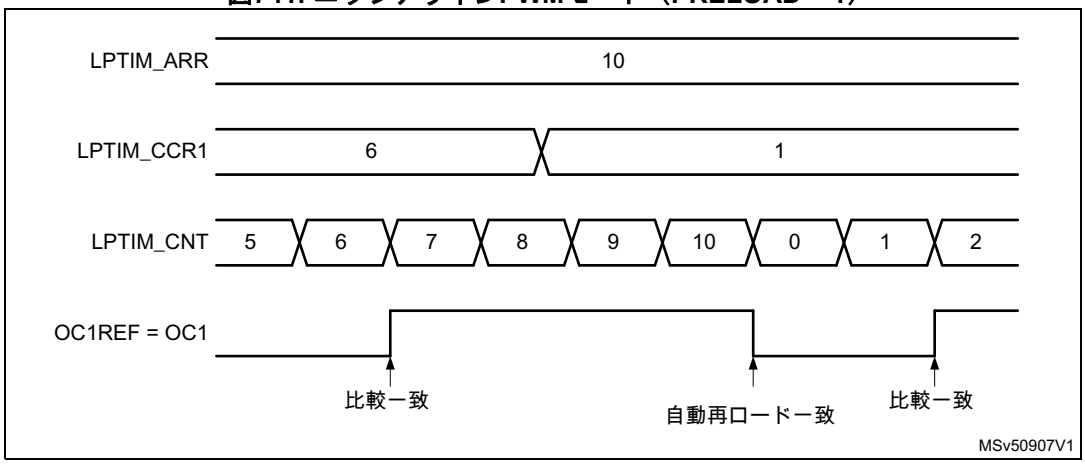
58.4.19 PWMモード

PWMモードでは、LPTIM_ARRレジスタの値によって決められた周波数とLPTIM_CCRxレジスタの値によって決められたデューティサイクルで信号を生成できます。LPTIMはエッジアラインモードでPWMを生成できます。

OCx極性は、LPTIM_CCMRxレジスタのCCxPビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx出力は、LPTIM_CCMRxレジスタのCCxEビットによって有効になります。詳細については、LPTIM_CCMRxレジスタの説明を参照してください。

図741に、LPTIMチャネル1が、LPTIM_CCR1 = 6、その後1、およびLPTIM_ARR=10でPWMモードに設定されている例を示します。

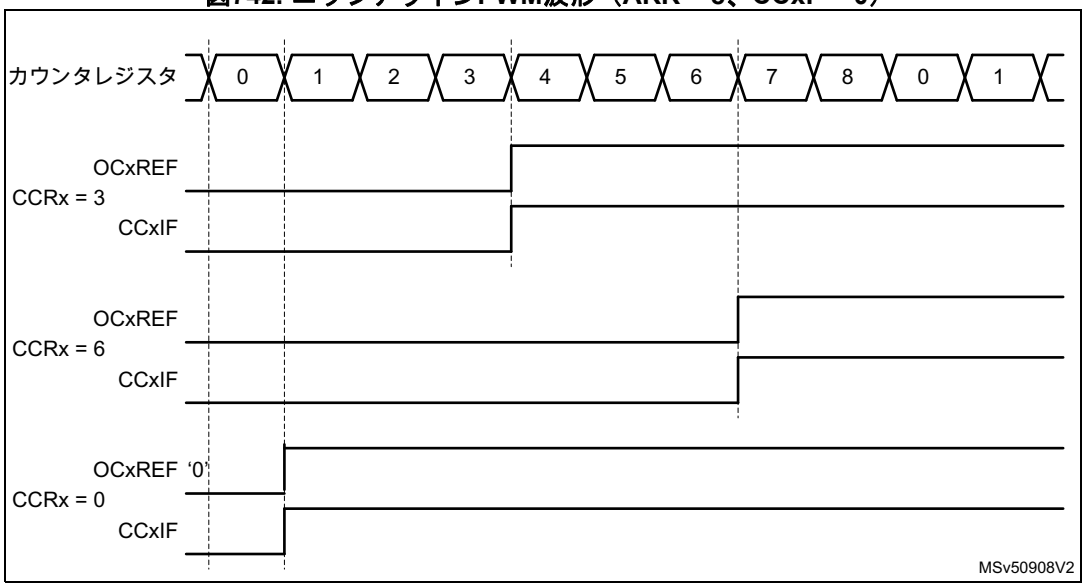
図741. エッジアラインPWMモード（PRELOAD = 1）



次の例では、基準PWM信号OCxREFは、LPTIM_CNT ≤ LPTIM_CCRxの間はローであり、そうでない場合はハイになります。

図742にLPTIM_ARR = 8の場合の例のいくつかのエッジアラインPWM波形を示します。

図742. エッジアラインPWM波形（ARR = 8、CCxP = 0）



即時更新（PRELOAD = 0）のPWMモード

PRELOAD = 0のPWMモードでは、現在のPWMサイクル内で出力レベルを早期に変更できます。LPTIM_CCRxレジスタの即時更新（PRELOAD = 0）、およびLPTIM_CNTレジスタとLPTIM_CCRxレジスタの継続的な比較に基づいて、現在のPWM周期の完了を待たずに、PWMサイクル内でできるだけ早く新しいデューティサイクル値を適用させることができます。

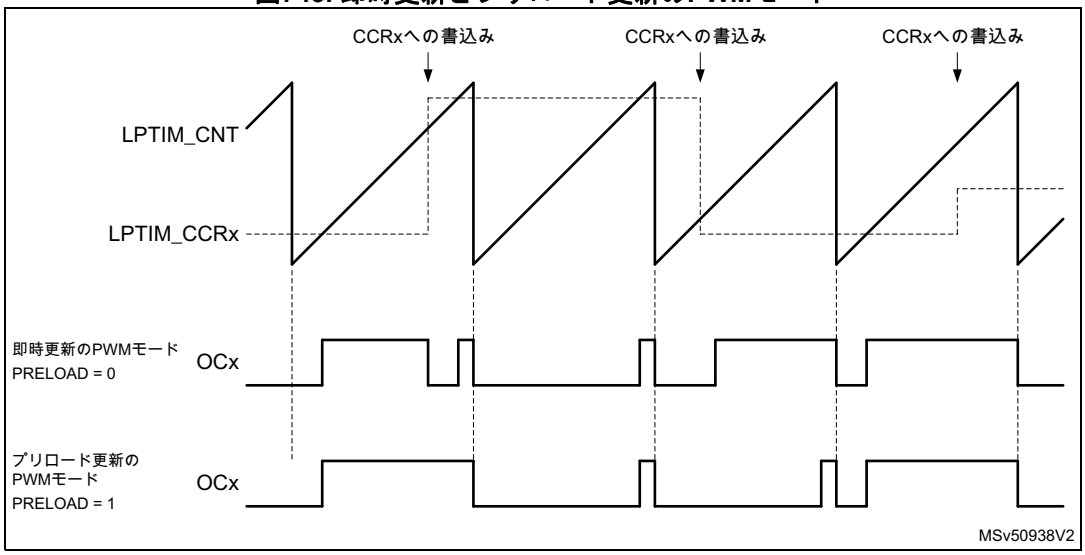
（PRELOAD = 0）の場合、LPTIM_CCRxレジスタの比較値を更新することにより、ソフトウェア（またはDMA）によってOCxREF信号レベルを動作中に変更できます。

書き込まれた比較値、および現在のカウンタ値と比較値に応じて、OCxREFレベルは次のように再割り当てされます。

- 新しい比較値が現在のカウンタ値を超えておらず、現在の比較値がカウンタ値を超えている場合、新しい比較値が書き込まれるとすぐにOCxREFレベルはハイに再割り当てされます。
- 新しい比較値がカウンタ値を超えており、現在の比較値がカウンタ値を超えていない場合、新しい比較値が書き込まれるとすぐにOCxREFレベルはローに再割り当てされます。

新しい比較値と現在の比較値が両方ともカウンタ値を超えていない場合、出力基準信号OCxREFレベルは変更されません。図743に、PRELOAD = 0およびPRELOAD = 1の場合のOCxREF信号レベルの動作を示します。

図743. 即時更新とプリロード更新のPWMモード



注： **どちらのPWMモードでも、比較一致、自動再ロード一致、更新イベントの各フラグは、対応するイベントの1 LPTIMカウンタサイクル後にセットされ、OCxREFレベルも、対応するイベントの1 LPTIMカウンタサイクル後に変更されます。たとえば、LPTIM_CCRxが3にセットされている場合、CCxIFはLPTIM_CNT = 4の時にセットされます。この動作を図741に示します。**

58.4.20 自律モード

このモードで使用可能なオシレータでクロックされている場合（RCCを参照）、LPTIMは自律モードで動作できます。これにより、APBクロックが停止していてもSTOPモードで完全に機能する状態を維持できます。APBクロックは、SRAMとの間でデータの転送が必要な場合、毎回ペリフェラルによってリクエストされます。ペリフェラルがAPBクロックを受信すると、LPTIM設定に応じて割込みまたはDMAリクエストが生成されます。

CPUのオフロード（RUNモード）またはSTOPモード時のウェイクアップを回避するには、LPTIM DMAリクエストを使用して、入力キャプチャモードの場合はキャプチャされた値を転送し、PWMモードの場合はLPTIMレジスタを更新できます。

STOPモード時、LPTIMカウンタは、その外部入力トリガの1つにアクティブエッジが検出された後で、自動的に開始できます。

入力キャプチャモード

STOPモードで自律的に動作するには、LPTIM_DIERレジスタのCCxDEビットをセットして、入力キャプチャ DMAリクエストを有効化する必要があります。

カウンタ値がキャプチャされ、LPTIM_CCRxレジスタで使用可能になるたびに、ペリフェラルからAPBクロックがリクエストされ、DMAリクエストが生成されます。キャプチャされた値は、その後SRAMに転送されます。キャプチャされた値がAPB（CPUやDMAなどの任意のバスマスタ）に読み出されると、CCxIFフラグがハードウェアによって自動的にクリアされます。

PWMモード

LPTIMは、出力波形パルス幅および/またはデューティサイクルを、CPUの介入なく、更新イベントのたびに自律的に変更するように設定できます。この自律モードを有効にするには、対応するUEDEビットをLPTIM_DIERレジスタにセットする必要があります。

更新イベントごとに、ペリフェラルからAPBクロックがリクエストされ、DMAリクエストが生成されます。DMA方向をメモリからペリフェラルに設定する必要があります。これにより、DMAリクエストごとにSRAMに格納された値でLPTIMレジスタを更新できます。

LPTIM_ARRレジスタがCPUやDMAなどのバスマスタによって書き込まれると、UEフラグがハードウェアによって自動的にクリアされます。そのため、UEフラグのハードウェアによる自動クリアを有効にするには、LPTIM_ARRレジスタを（リストの）最後に書き込むようにアプリケーションを設定する必要があります。たとえば、LPTIM_CCR1とLPTIM_RCRレジスタをDMAによってSTOPモードで更新する必要がある場合、更新シーケンスはLPTIM_CCR1、LPTIM_RCR、LPTIM_ARRの順でなければなりません。

UEフラグは、LPTIM_ICRレジスタの対応するクリアビットUECFでもクリアできます。これには、LPTIM_ICRレジスタをレジスタ更新の最後に書き込むようDMAを設定します。

58.4.21 DMAリクエスト

LPTIMには、次の2つのカテゴリのDMAリクエストを生成する機能があります。

- 入力キャプチャカウンタ値を取得するために使用されるDMAリクエスト
- DMA更新リクエストは、ソフトウェアのオーバーヘッドを発生させずに、LPTIMERの一部を定期的に複数回再プログラムするために使用されます。

入力キャプチャ DMAリクエスト

各LPTIMチャネルには専用の入力キャプチャ DMAリクエストがあります。CCRxレジスタでキャプチャの準備ができるたびに、DMAリクエストが生成され（LPTIM_DIERでCCxDEビットがセットされている場合）、CCxIFがセットされます。その後、CCRxにキャプチャされた値は、DMAによって目的のメモリの転送先に定期的に転送できます。CCRxレジスタにキャプチャされた値が読み出されると、CCxIFはハードウェアによって自動的にクリアされます。

注： **ICx DMAリクエスト信号lptim_icx_dmaは、次の状況でリセットされます。**

- **該当するDMAリクエストが無効になった場合（LPTIM_DIERレジスタのCCxDEビットのクリア）**
- **または、チャネルxが無効になった場合（CCxEビットのクリア）**
- **または、LPTIMが無効になった場合（LPTIM_CRレジスタのENABLEビットのクリア）**

更新イベントDMAリクエスト

更新イベントごとにDMAリクエストが生成され（LPTIM_DIERにUEDEが設定されている場合）、UEフラグがセットされます。DMA リクエストを使用すると、LPTIM_ARR、LPTIM_RCR、またはLPTIM_CCRxレジスタを定期的に更新し、カスタムPWM波形を生成できるようになります。

UEは、LPTIM_ARRレジスタへのバスマスタ（CPUやDMAなど）の書込みアクセス時にハードウェアによって自動的にクリアされます。

注： **UE DMAリクエスト信号lptim_ue_dmaは、次の状況でリセットされます。**

- **該当するDMAリクエストが無効になった場合（LPTIM_DIERレジスタのUEDEビットのクリア）**
- **または、LPTIMが無効になった場合（LPTIM_CRレジスタのENABLEビットのクリア）**

58.4.22 デバッグモード

マイクロコントローラがデバッグモードになると（コアは停止状態）、LPTIMカウンタは、DBGモジュールのDBG_LPTIM_STOP設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

58.5 LPTIM低消費電力モード

表608.低消費電力モードがLPTIMに与える影響

モード	説明
SLEEP	影響はありません。LPTIM割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	STOPモードで使用可能なオシレータからLPTIMのクロックが供給されている場合、LPTIMは動作を続け、割込みによりデバイスはSTOPモードを終了します。DMAリクエストは、インスタンスが自律モードをサポートする場合に機能します（セクション 58.3：LPTIMの実装を参照）。
STANDBY	LPTIMペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. LPTIM1およびLPTIM3のみが自律モードをサポートし、STOP 2モードでウェイクアップ機能を使用できます。LPTIM4は自律モードをサポートしていませんが、STOP 2モードでウェイクアップ機能を使用できます。STOP 2モードに入る前にLPTIM2を無効にする必要があります。

58.6 LPTIM割込み

LPTIM_DIERレジスタで有効化されていた場合、次のイベントが発生すると、割込み／ウェイクアップイベントが生成されます。

- 比較一致
- 自動再ロード一致（エンコーダモードの場合は方向にかかわらず）
- 外部トリガイベント
- 自動再ロードレジスタへの書込み完了
- 比較レジスタへの書込み完了
- 方向変更（エンコーダモード）、プログラム可能（アップ／ダウン／両方）
- 更新イベント
- 繰返しレジスタ更新OK
- 入力キャプチャの発生
- オーバーキャプチャの発生
- 割込みイネーブルレジスタ更新OK

注： **LPTIM_ISRレジスタ（ステータスレジスタ）の対応するフラグがセットされた後に、LPTIM_DIERレジスタのビットをセットしても、割込みはアサートされません。**

表609.割込みイベント

割込みベクタ	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了(1)
LPTIMx	比較一致	CCxIF	CCxIE	CCxCFに1を書き込む	可	はい
	入力キャプチャ(2)	CCxIF	CCxIE	CCxCFに1を書き込む	可	はい
	オーバーキャプチャ (2)	CCxOF	CCxOIE	CCxOCFに1を書き込む	可	はい
	自動再ロード一致	ARRM	ARRMIE	ARRMCFに1を書き込みます。	可	はい
	外部トリガイベント	EXTTRIG	EXTTRIGIE	EXTTRIGCFに1を書き込む	可	はい
	自動再ロードレジスタ更新OK	ARROK	ARROKIE	ARROKCFに1を書き込む	可	はい
	キャプチャ／比較レジスタ更新OK	CMPxOK	CMPxOKIE	CMPxOKCFに1を書き込む	可	はい
	アップ方向への変更(3)	UP	UPIE	UPCFに1を書き込みます。	可	はい
	ダウン方向への変更 (3)	DOWN	DOWNIE	DOWNCFに1を書き込みます。	可	はい
	更新イベント	UE	UEIE	UECFに1を書き込みます。	可	はい
	繰返しレジスタ更新OK	REPOK	REPOKIE	REPOKCFに1を書き込む	可	はい

1. 各LPTIMイベントは、LPTIMインスタンスがSTOPモードからのウェイクアップ機能をサポートしている場合にのみデバイスをSTOPモードからウェイクアップすることができます。セクション 58.3：LPTIMの実装を参照してください。
2. LPTIMがチャネルを実装していない場合、このイベントは存在しません。セクション 58.3：LPTIMの実装を参照してください。
3. LPTIMがエンコーダモード機能をサポートしていない場合、このイベントは存在しません。セクション 58.3：LPTIMの実装を参照してください。

58.7 LPTIMレジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2: 123ページのレジスタに関する略記を参照してください。

ペリフェラルレジスタには、ワード（32-bit）単位でアクセスすることができます。

58.7.1 LPTIM4 割込みおよびステータスレジスタ（LPTIM4_ISR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIER OK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							r								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP OK	UE	DOWN	UP	ARR OK	CMP1 OK	EXT TRIG	ARRM	CC1IF
							r	r	r	r	r	r	r	r	r

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROK**：割込みイネーブルレジスタ更新OK

DIEROKは、LPTIM_DIERレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。DIEROKフラグは、LPTIM_ICRレジスタのDIEROKCFビットに1を書き込むことでクリアできます。

ビット23:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **REPOK**：繰返しレジスタ更新OK

REPOKは、LPTIM_RCRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。REPOKフラグは、LPTIM_ICRレジスタのREPOKCFビットに1を書き込むことでクリアできます。

ビット7 **UE**：LPTIM更新イベントの発生

UEは、更新イベントが生成されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UEフラグは、LPTIM_ICRレジスタのUECFビットに1を書き込むことでクリアできます。

ビット6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWNビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。DOWNフラグは、LPTIM_ICRレジスタのDOWNCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 58.3：LPTIMの実装 を参照してください。

ビット5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UPビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UPフラグは、LPTIM_ICRレジスタのUPCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 58.3：LPTIMの実装 を参照してください。

ビット4 **ARROK**：自動再ロードレジスタ更新OK

ARROKは、LPTIM_ARRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARROKフラグは、LPTIM_ICRレジスタのARROKCFビットに1を書き込むことでクリアできます。

ビット3 **CMP1OK**：比較レジスタ1更新OK

CMP1OKは、LPTIM_CCR1レジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CMP1OKフラグは、LPTIM_ICRレジスタのCMP1OKCFビットに1を書き込むことでクリアできます。

ビット2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIGは、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。EXTTRIGフラグは、LPTIM_ICRレジスタのEXTTRIGCFビットに1を書き込むことでクリアできます。

ビット1 **ARRM**：自動再ロード一致

ARRMは、LPTIM_CNTレジスタの値がLPTIM_ARRレジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARRMフラグは、LPTIM_ICRレジスタのARRMCFビットに1を書き込むことでクリアできます。

ビット0 **CC1IF**：比較1割込みフラグ

CC1IFフラグは、LPTIM_CNTレジスタの値が比較レジスタの値に一致したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CC1IFフラグは、LPTIM_ICRレジスタのCC1CFビットに1を書き込むことでクリアできます。

0：一致していません。

1：カウンタLPTIM_CNTレジスタの値の内容がLPTIM_CCR1レジスタの値と一致しました。

58.7.2 LPTIMx 割込みとステータスレジスタ [オルタネート]（LPTIMx_ISR）（x = 1～3）

レジスタのこの説明は、出力比較モードに対してのみ使用できます。入力キャプチャモードについては次のセクションを参照してください。

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIER OK	Res.	Res.	Res.	Res.	CMP2 OK	Res.	Res.	Res.
							r					r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC2IF	REP OK	UE	DOWN	UP	ARR OK	CMP1 OK	EXT TRIG	ARRM	CC1IF
						r	r	r	r	r	r	r	r	r	r

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROK**：割込みイネーブルレジスタ更新OK

DIEROKは、LPTIM_DIERレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。DIEROKフラグは、LPTIM_ICRレジスタのDIEROKCFビットに1を書き込むことでクリアできます。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CMP2OK**：比較レジスタ2更新OK

CMP2OKは、LPTIM_CCR2レジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CMP2OKフラグは、LPTIM_ICRレジスタのCMP2OKCFビットに1を書き込むことでクリアできます。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット18:12 予約済みであり、リセット値に保持する必要があります。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2IF**：比較2割込みフラグ

CC2チャネルが出力として設定されている場合：

CC2IFフラグは、LPTIM_CNTレジスタの値が比較レジスタの値に一致したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

CC2IFフラグは、LPTIM_ICRレジスタのCC2CFビットに1を書き込むことでクリアできます。

0：一致していません。

1：カウンタLPTIM_CNTレジスタの値の内容がLPTIM_CCR2レジスタの値と一致しました。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOK**：繰返しレジスタ更新OK

REPOKは、LPTIM_RCRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。REPOKフラグは、LPTIM_ICRレジスタのREPOKCFビットに1を書き込むことでクリアできます。

ビット7 **UE**：LPTIM更新イベントの発生

UEは、更新イベントが生成されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。対応する割込みまたはDMAリクエストが生成されます（有効な場合）。UEフラグは、LPTIM_ICRレジスタのUECFビットに1を書き込むことでクリアできます。LPTIM_ARRレジスタがCPUやDMAなどのバスマスタによって書き込まれると、UEフラグがハードウェアによって自動的にクリアされます。

ビット6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWNビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。DOWNフラグは、LPTIM_ICRレジスタのDOWNCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UPビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UPフラグは、LPTIM_ICRレジスタのUPCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROK**：自動再ロードレジスタ更新OK

ARROKは、LPTIM_ARRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARROKフラグは、LPTIM_ICRレジスタのARROKCFビットに1を書き込むことでクリアできます。

ビット3 **CMP1OK**：比較レジスタ1更新OK

CMP1OKは、LPTIM_CCR1レジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CMP1OKフラグは、LPTIM_ICRレジスタのCMP1OKCFビットに1を書き込むことでクリアできます。

ビット2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIGは、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。EXTTRIGフラグは、LPTIM_ICRレジスタのEXTTRIGCFビットに1を書き込むことでクリアできます。

ビット1 **ARRM**：自動再ロード一致

ARRMは、LPTIM_CNTレジスタの値がLPTIM_ARRレジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARRMフラグは、LPTIM_ICRレジスタのARRMCFビットに1を書き込むことでクリアできます。

ビット0 **CC1IF**：比較1割込みフラグ

CC1チャネルが出力として設定されている場合：

CC1IFフラグは、LPTIM_CNTレジスタの値が比較レジスタの値に一致したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CC1IFフラグは、LPTIM_ICRレジスタのCC1CFビットに1を書き込むことでクリアできます。

0：一致していません。

1：カウンタLPTIM_CNTレジスタの値の内容がLPTIM_CCR1レジスタの値と一致しました。

58.7.3 LPTIMx 割込みとステータスレジスタ [オルタネート]（LPTIMx_ISR）（x = 1～3）

レジスタのこの説明は、入力キャプチャモードに対してのみ使用できます。出力比較モードについては前のセクションを参照してください。

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							r								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CC2OF	CC1OF	Res.	Res.	CC2IF	REPOK	UE	DOWN	UP	ARROK	Res.	EXTTRIG	ARRM	CC1IF
		r	r			r	r	r	r	r	r		r	r	r

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROK**：割込みイネーブルレジスタ更新OK

DIEROKは、LPTIM_DIERレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。DIEROKフラグは、LPTIM_ICRレジスタのDIEROKCFビットに1を書き込むことでクリアできます。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CC2OF**：キャプチャ 2オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。LPTIM_ICRレジスタのCC2OCFビットに1を書き込むことで、ソフトウェアによってクリアされます。

0 :オーバーキャプチャは検出されていません。

1 :CC2IFフラグがすでにセットされているときに、カウンタの値がLPTIM_CCR2レジスタにキャプチャされました。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット12 **CC1OF**：キャプチャ 1オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。LPTIM_ICRレジスタのCC1OCFビットに1を書き込むことで、ソフトウェアによってクリアされます。

0 :オーバーキャプチャは検出されていません。

1 :CC1IFフラグがすでにセットされているときに、カウンタの値がLPTIM_CCR1レジスタにキャプチャされました。

注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2IF**：キャプチャ 2割込みフラグ

CC2チャネルが入力として設定されている場合：

CC2IFは、カウンタの現在値がLPTIM_CCR2レジスタにキャプチャされたことをアプリケーションに知らせるために、ハードウェアによってセットされます。対応する割込みまたはDMAリクエストが生成されます（有効な場合）。CC2IFフラグがすでにハイの場合、CC2OFフラグがセットされます。

0 :入力キャプチャは発生していません。

1 :カウンタ値がLPTIM_CCR2レジスタにキャプチャされました。（選択された極性に一致するエッジがIC2で検出されました）。キャプチャされた値が（CPUまたはDMAによって）読み出されると、CC2IFフラグはハードウェアによって自動的にクリアされます。CC2IFフラグは、LPTIM_ICRレジスタのCC2CFビットに1を書き込むことでクリアできます。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOK**：繰返しレジスタ更新OK

REPOKは、LPTIM_RCRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。REPOKフラグは、LPTIM_ICRレジスタのREPOKCFビットに1を書き込むことでクリアできます。

ビット7 **UE**：LPTIM更新イベントの発生

UEは、更新イベントが生成されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UEフラグは、LPTIM_ICRレジスタのUECFビットに1を書き込むことでクリアできます。

ビット6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWNビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。DOWNフラグは、LPTIM_ICRレジスタのDOWNCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UPビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UPフラグは、LPTIM_ICRレジスタのUPCFビットに1を書き込むことでクリアできます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROK**：自動再ロードレジスタ更新OK

ARROKは、LPTIM_ARRレジスタへのAPBバス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARROKフラグは、LPTIM_ICRレジスタのARROKCFビットに1を書き込むことでクリアできます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIGは、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。EXTTRIGフラグは、LPTIM_ICRレジスタのEXTTRIGCFビットに1を書き込むことでクリアできます。

ビット1 **ARRM**：自動再ロード一致

ARRMは、LPTIM_CNTレジスタの値がLPTIM_ARRレジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARRMフラグは、LPTIM_ICRレジスタのARRMCFビットに1を書き込むことでクリアできます。

ビット0 **CC1IF**：キャプチャ 1割込みフラグ

CC1チャネルが入力として設定されている場合：

CC1IFは、カウンタの現在値がLPTIM_CCR1レジスタにキャプチャされたことをアプリケーションに知らせるために、ハードウェアによってセットされます。対応する割込みまたはDMAリクエストが生成されます（有効な場合）。CC1IFフラグがすでにハイの場合、CC1OFフラグがセットされます。

0：入力キャプチャは発生していません。

1：カウンタ値がLPTIM_CCR1レジスタにキャプチャされました。（選択された極性に一致するエッジがIC1で検出されました）。キャプチャされた値が（CPUまたはDMAによって）読み出されると、CC1IFフラグはハードウェアによって自動的にクリアされます。CC1IFフラグは、LPTIM_ICRレジスタのCC1CFビットに1を書き込むことでクリアできます。

58.7.4 LPTIM4 割込みクリアレジスタ（LPTIM4_ICR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							w								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPOKCF	UECF	DOWNCF	UPCF	ARROKCF	CMP1OKCF	EXTTRIGCF	ARRMCF	CC1CF
							w	w	w	w	w	w	w	w	w

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROKCF**：割込みイネーブルレジスタ更新OKクリアフラグ

このビットに1を書き込むと、LPTIM_ISRレジスタのDIEROKフラグがクリアされます。

ビット23:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **REPOKCF**：繰返しレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのREPOKフラグがクリアされます。

ビット7 **UECF**：更新イベントクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのUEフラグがクリアされます。

ビット6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに1を書き込むと、LPTIM_ISRレジスタのDOWNフラグがクリアされます。
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPCF**：アップへの方向変更フラグクリア
このビットに1を書き込むと、LPTIM_ISRレジスタのUPフラグがクリアされます。
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKCF**：自動再ロードレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのARROKフラグがクリアされます。

ビット3 **CMP1OKCF**：比較レジスタ1更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCMP1OKフラグがクリアされます。

ビット2 **EXTTRIGCF**：外部トリガ有効エッジクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのEXTTRIGフラグがクリアされます。

ビット1 **ARRMCF**：自動再ロード一致クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのARRMフラグがクリアされます。

ビット0 **CC1CF**：キャプチャ／比較1クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC1IFフラグがクリアされます。

58.7.5 LPTIMx 割込みクリアレジスタ [オルタネート]（LPTIMx_ICR）（x = 1～3）

レジスタのこの説明は、出力比較モードに対してのみ使用できます。入力キャプチャ比較モードについては次のセクションを参照してください。

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	CMP2OKCF	Res.	Res.	Res.
							w					w			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC2CF	REPOKCF	UECF	DOWNCF	UPCF	ARROKCF	CMP1OKCF	EXTTRIGCF	ARRMCF	CC1CF
						w	w	w	w	w	w	w	w	w	w

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROKCF**：割込みイネーブルレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのDIEROKフラグがクリアされます。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CMP2OKCF**：比較レジスタ2更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCMP2OKフラグがクリアされます。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット18:12 予約済みであり、リセット値に保持する必要があります。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2CF**：キャプチャ／比較2クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC2IFフラグがクリアされます。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOKCF**：繰返しレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのREPOKフラグがクリアされます。

ビット7 **UECF**：更新イベントクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのUEフラグがクリアされます。

ビット6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに1を書き込むと、LPTIM_ISRレジスタのDOWNフラグがクリアされます。
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPCF**：アップへの方向変更フラグクリア
このビットに1を書き込むと、LPTIM_ISRレジスタのUPフラグがクリアされます。
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKCF**：自動再ロードレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのARROKフラグがクリアされます。

ビット3 **CMP1OKCF**：比較レジスタ1更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCMP1OKフラグがクリアされます。

ビット2 **EXTTRIGCF**：外部トリガ有効エッジクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのEXTTRIGフラグがクリアされます。

ビット1 **ARRMCF**：自動再ロード一致クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのARRMフラグがクリアされます。

ビット0 **CC1CF**：キャプチャ／比較1クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC1IFフラグがクリアされます。

58.7.6 LPTIMx 割込みクリアレジスタ [オルタネート]（LPTIMx_ICR）（x = 1～3）

レジスタのこの説明は、入力キャプチャモードに対してのみ使用できます。出力比較モードについては前のセクションを参照してください。

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							w								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CC2OCF	CC1OCF	Res.	Res.	CC2CF	REPOKCF	UECF	DOWNCF	UPCF	ARROKCF	Res.	EXTTRIGCF	ARRMCF	CC1CF
		w	w			w	w	w	w	w	w		w	w	w

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **DIEROKCF**：割込みイネーブルレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのDIEROKフラグがクリアされます。

ビット23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CC2OCF**：キャプチャ／比較2オーバーキャプチャクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC2OFフラグがクリアされます。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット12 **CC1OCF**：キャプチャ／比較1オーバーキャプチャクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC1OFフラグがクリアされます。
注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2CF**：キャプチャ／比較2クリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのCC2IFフラグがクリアされます。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOKCF**：繰返しレジスタ更新OKクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのREPOKフラグがクリアされます。

ビット7 **UECF**：更新イベントクリアフラグ
このビットに1を書き込むと、LPTIM_ISRレジスタのUEフラグがクリアされます。

ビット6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに1を書き込むと、LPTIM_ISRレジスタのDOWNフラグがクリアされます。
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPCF**：アップへの方向変更フラグクリア

このビットに1を書き込むと、LPTIM_ISRレジスタのUPフラグがクリアされます。

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKCF**：自動再ロードレジスタ更新OKクリアフラグ

このビットに1を書き込むと、LPTIM_ISRレジスタのARROKフラグがクリアされます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **EXTTRIGCF**：外部トリガ有効エッジクリアフラグ

このビットに1を書き込むと、LPTIM_ISRレジスタのEXTTRIGフラグがクリアされます。

ビット1 **ARRMCF**：自動再ロード一致クリアフラグ

このビットに1を書き込むと、LPTIM_ISRレジスタのARRMフラグがクリアされます。

ビット0 **CC1CF**：キャプチャ／比較1クリアフラグ

このビットに1を書き込むと、LPTIM_ISRレジスタのCC1IFフラグがクリアされます。

58.7.7 LPTIM4 割込み有効レジスタ（LPTIM4_DIER）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPOKIE	UEIE	DOWNIE	UPIE	ARROKIE	CMP1OKIE	EXTTRIGIE	ARRMIE	CC1IE
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **REPOKIE**：繰返しレジスタ更新OK割込みイネーブル

0：繰返しレジスタ更新OK割込みは無効です。

1：繰返しレジスタ更新OK割込みは有効です。

ビット7 **UEIE**：更新イベント割込みイネーブル

0：更新イベント割込みは無効です。

1：更新イベント割込みは有効です。

ビット6 **DOWNIE**：ダウンへの方向変更割込み有効化

0： DOWN割込み無効

1： DOWN割込み有効

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPIE**：アップへの方向変更割込み有効化

0： UP割込み無効

1： UP割込み有効

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKIE**：自動再ロードレジスタ更新OK割込み有効化

0： ARROK割込み無効

1： ARROK割込み有効

ビット3 **CMP1OKIE**：比較レジスタ1更新OK割込みイネーブル
0： CMPOKレジスタ1割込みは無効です。
1： CMPOKレジスタ1割込みは有効です。

ビット2 **EXTTRIGIE**：外部トリガ有効エッジ割込み有効化
0： EXTTRIG割込み無効
1： EXTTRIG割込み有効

ビット1 **ARRMIE**：自動再ロード一致割込み有効化
0： ARRM割込み無効
1： ARRM割込み有効

ビット0 **CC1IE**：キャプチャ／比較1割込みイネーブル
0： キャプチャ／比較1割込みは無効です。
1： キャプチャ／比較1割込みは有効です。

注意： LPTIMx_DIERレジスタの変更は、LPTIMが有効な（ENABLEビットが 1にセットされている）ときにのみ行う必要があります。LPTIMx_DIERレジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。DIEROKフラグがセットされる前に続けて書込みを行うと、予測不能な結果になります。

58.7.8 LPTIMx 割込み有効レジスタ [オルタネート]（LPTIMx_DIER）（x = 1～3）

レジスタのこの説明は、出力比較モードに対してのみ使用できます。入力キャプチャ比較モードについては次のセクションを参照してください。

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UEDE	Res.	Res.	Res.	CMP2OKIE	Res.	Res.	Res.
								rw				rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC2IE	REPOKIE	UEIE	DOWNIE	UPIE	ARROKIE	CMP1OKIE	EXTTRIGIE	ARRMIE	CC1IE
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **UEDE**：更新イベントDMAリクエストイネーブル
0： UE DMAリクエストは無効です。このUEDEビットに "0" を書き込むと、関連するue_dma_req信号がリセットされます。
1： UE DMAリクエストは有効です。
注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CMP2OKIE**：比較レジスタ2更新OK割込みイネーブル
0： CMPOKレジスタ2割込みは無効です。
1： CMPOKレジスタ2割込みは有効です。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット18:12 予約済みであり、リセット値に保持する必要があります。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2IE**：キャプチャ／比較2割込みイネーブル

0： キャプチャ／比較2割込みは無効です。

1： キャプチャ／比較2割込みは有効です。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOKIE**：繰返しレジスタ更新OK割込みイネーブル

0：繰返しレジスタ更新OK割込みは無効です。

1：繰返しレジスタ更新OK割込みは有効です。

ビット7 **UEIE**：更新イベント割込みイネーブル

0：更新イベント割込みは無効です。

1：更新イベント割込みは有効です。

ビット6 **DOWNIE**：ダウンへの方向変更割込み有効化

0： DOWN割込み無効

1： DOWN割込み有効

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPIE**：アップへの方向変更割込み有効化

0： UP割込み無効

1： UP割込み有効

注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKIE**：自動再ロードレジスタ更新OK割込み有効化

0： ARROK割込み無効

1： ARROK割込み有効

ビット3 **CMP1OKIE**：比較レジスタ1更新OK割込みイネーブル

0： CMPOKレジスタ1割込みは無効です。

1： CMPOKレジスタ1割込みは有効です。

ビット2 **EXTTRIGIE**：外部トリガ有効エッジ割込み有効化

0： EXTTRIG割込み無効

1： EXTTRIG割込み有効

ビット1 **ARRMIE**：自動再ロード一致割込み有効化

0： ARRM割込み無効

1： ARRM割込み有効

ビット0 **CC1IE**：キャプチャ／比較1割込みイネーブル

0： キャプチャ／比較1割込みは無効です。

1： キャプチャ／比較1割込みは有効です。

注意： LPTIMx_DIERレジスタの変更は、LPTIMが有効な（ENABLEビットが1にセットされている）ときにのみ行う必要があります。LPTIMx_DIERレジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。DIEROKフラグがセットされる前に続けて書込みを行うと、予測不能な結果になります。

58.7.9 LPTIMx 割込み有効レジスタ [オルタネート]（LPTIMx_DIER）（x = 1～3）

レジスタのこの説明は、入力キャプチャモードに対してのみ使用できます。出力比較モードについては前のセクションを参照してください。

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	CC2DE	Res.	UEDE	Res.	Res.	Res.	Res.	Res.	Res.	CC1DE
						rw		rw							rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CC2OIE	CC1OIE	Res.	Res.	CC2IE	REPOKIE	UEIE	DOWNIE	UPIE	ARROKIE	Res.	EXTTRIGIE	ARRMIE	CC1IE
		rw	rw			rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25 **CC2DE**：キャプチャ／比較2 DMAリクエストイネーブル

0：CC2 DMAリクエストは無効です。このCC2DEビットに "0" を書き込むと、関連するic2_dma_req信号がリセットされます。

1：CC2 DMAリクエストは有効です。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット24 予約済みであり、リセット値に保持する必要があります。

ビット23 **UEDE**：更新イベントDMAリクエストイネーブル

0：UE DMAリクエストは無効です。このUEDEビットに "0" を書き込むと、関連するue_dma_req信号がリセットされます。

1：UE DMAリクエストは有効です。

注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット22:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **CC1DE**：キャプチャ／比較1 DMAリクエストイネーブル

0：CC1 DMAリクエストは無効です。このCC1DEビットに "0" を書き込むと、関連するic1_dma_req信号がリセットされます。

1：CC1 DMAリクエストは有効です。

注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CC2OIE**：キャプチャ／比較2オーバーキャプチャ割込みイネーブル

0：CC2オーバーキャプチャ割込みは無効です。

1：CC2オーバーキャプチャ割込みは有効です。

注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット12 **CC1OIE**：キャプチャ／比較1オーバーキャプチャ割込みイネーブル
0：CC1オーバーキャプチャ割込みは無効です。
1：CC1オーバーキャプチャ割込みは有効です。
注： LPTIMが少なくとも1つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CC2IE**：キャプチャ／比較2割込みイネーブル
0：キャプチャ／比較2割込みは無効です。
1：キャプチャ／比較2割込みは有効です。
注： LPTIMが少なくとも2つのチャネルを実装していない場合、このビットは予約済みです。セクション58.3を参照してください。

ビット8 **REPOKIE**：繰返しレジスタ更新OK割込みイネーブル
0：繰返しレジスタ更新OK割込みは無効です。
1：繰返しレジスタ更新OK割込みは有効です。

ビット7 **UEIE**：更新イベント割込みイネーブル
0：更新イベント割込みは無効です。
1：更新イベント割込みは有効です。

ビット6 **DOWNIE**：ダウンへの方向変更割込み有効化
0：DOWN割込み無効
1：DOWN割込み有効
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット5 **UPIE**：アップへの方向変更割込み有効化
0：UP割込み無効
1：UP割込み有効
注： LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3を参照してください。

ビット4 **ARROKIE**：自動再ロードレジスタ更新OK割込み有効化
0：ARROK割込み無効
1：ARROK割込み有効

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **EXTTRIGIE**：外部トリガ有効エッジ割込み有効化
0：EXTTRIG割込み無効
1：EXTTRIG割込み有効

ビット1 **ARRMIE**：自動再ロード一致割込み有効化
0：ARRM割込み無効
1：ARRM割込み有効

ビット0 **CC1IE**：キャプチャ／比較1割込みイネーブル
0：キャプチャ／比較1割込みは無効です。
1：キャプチャ／比較1割込みは有効です。

注意： LPTIMx_DIERレジスタの変更は、LPTIMが有効な（ENABLEビットが1にセットされている）ときにのみ行う必要があります。LPTIMx_DIERレジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。DIEROKフラグがセットされる前に続けて書込みを行うと、予測不能な結果になります。

58.7.10 LPTIM設定レジスタ（LPTIM_CFGR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNT MODE	PRE LOAD	WAV POL	WAVE	TIMOUT	TRIGEN[1:0]		Res.
							rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGSEL[2:0]			Res.	PRESC[2:0]			Res.	TRGFLT[1:0]		Res.	CKFLT[1:0]		CKPOL[1:0]		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ENC**：エンコーダモード有効化

ENCビットは、エンコーダモードを制御します。

0： エンコーダモード無効

1： エンコーダモード有効

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション58.3 を参照してください。

ビット23 **COUNTMODE**：カウンタモード有効化

COUNTMODEビットは、LPTIMがカウンタのクロックに使用するクロックソースを選択します。

0：カウンタは各内部クロックのパルスに従ってインクリメントされます。

1：カウンタはLPTIM外部Input1の各有効なクロックパルスに従ってインクリメントされます。

ビット22 **PRELOAD**：レジスタ更新モード

PRELOADビットは、LPTIM_ARR、LPTIM_RCR、および LPTIM_CCRx レジスタの更新方法を制御します。

0： レジスタは、各APBバス書込みアクセス後に更新されます。

1： レジスタは、現在のLPTIM周期の終了時に更新されます。

ビット21 **WAVPOL**：波形極性

WAVPOLビットは、出力の極性を制御します。

0： LPTIM出力はLPTIM_CNTレジスタとLPTIM_CCRxレジスタの比較結果を反映します。

1： LPTIM出力はLPTIM_CNTレジスタとLPTIM_CCRxレジスタの比較結果の反転を反映します。

注： LPTIMが少なくとも1つのキャプチャ／比較チャネルを実装している場合、このビットは予約済みです。セクション58.3 を参照してください。

ビット20 **WAVE**：波形

WAVEビットは、出力波形を制御します。

0： セットワンスモードを非アクティブにします。

1： セットワンスモードをアクティブにします。

ビット19 **TIMOUT**：タイムアウト有効化

TIMOUTビットは、タイムアウト機能を制御します。

0： タイマがすでに開始しているときに着信したトリガイベントは無視されます。

1： タイマがすでに開始しているときにトリガイベントが着信すると、LPTIM カウンタと繰返しカウンタがリセットされて、再開始されます。

ビット18:17 **TRIGEN[1:0]**：トリガ有効化および極性

TRIGENビットは、LPTIMカウンタが外部トリガによって開始されるかどうかを制御します。外部トリガオプションが選択された場合、トリガのアクティブエッジについて3つの構成が可能です。

00：ソフトウェアトリガ（カウントの開始はソフトウェアによって行われます）。

01：立上がりエッジがアクティブエッジです。

10：立下がりエッジがアクティブエッジです。

11：両方のエッジがアクティブエッジです。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15:13 **TRIGSEL[2:0]**：トリガセレクタ

TRIGSELビットは、次の8つの利用可能なソースの中からLPTIMのトリガイベントの役目を果たすトリガソースを選択します。

000：lptim_ext_trig0

001：lptim_ext_trig1

010：lptim_ext_trig2

011：lptim_ext_trig3

100：lptim_ext_trig4

101：lptim_ext_trig5

110：lptim_ext_trig6

111：lptim_ext_trig7

詳細については、セクション 58.4.3：LPTIM入力およびトリガマッピングを参照してください。

ビット12 予約済みであり、リセット値に保持する必要があります。

ビット11:9 **PRESC[2:0]**：クロックプリスケーラ

PRESCビットは、プリスケーラ分周比を設定します。次の分周比から選択できます。

000：/1

001：/2

010：/4

011：/8

100：/16

101：/32

110：/64

111：/128

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7:6 **TRGFLT[1:0]**：設定可能なトリガ用デジタルフィルタ

TRGFLT値は、内部トリガ上でレベルの変化が発生した時に、有効なレベル遷移とみなすまで検出する同じサンプルの連続数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：トリガのアクティブレベル変更は、有効なトリガとみなされます。

01：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも2クロック周期にわたって安定している必要があります。

10：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも4クロック周期にわたって安定している必要があります。

11：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも8クロック周期にわたって安定している必要があります。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:3 **CKFLT[1:0]**：設定可能な外部クロック用デジタルフィルタ

CKFLT値は、外部クロック信号上でレベル変化が発生した時に、有効なレベル遷移とみなすまで検出する同じサンプルの連続数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：外部クロック信号のレベル変更は、有効な遷移とみなされます。

01：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも2クロック周期にわたって安定している必要があります。

10：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも4クロック周期にわたって安定している必要があります。

11：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも8クロック周期にわたって安定している必要があります。

ビット2:1 **CKPOL[1:0]**：クロック極性

LPTIMのクロックが外部クロックソースによって供給されるとき、CKPOLビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。

00：立上がりエッジが、カウントに使用されるアクティブエッジです。
LPTIMがエンコーダモードで設定されている（ENCビットがセットされている）場合、エンコーダサブモード1がアクティブです。

01：立下がりエッジが、カウントに使用されるアクティブエッジです。
LPTIMがエンコーダモードで設定されている（ENCビットがセットされている）場合、エンコーダサブモード2がアクティブです。

10：両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIMのクロックは内部クロックソースからも供給される必要があり、その周波数は外部クロック周波数の4倍以上である必要があります。
LPTIMがエンコーダモードで設定されている（ENCビットがセットされている）場合、エンコーダサブモード3がアクティブです。

11：使用できません。

エンコーダモードのサブモードの詳細については、セクション 58.4.15：エンコーダモード を参照してください。

ビット0 **CKSEL**：クロックセレクタ

CKSELビットは、LPTIMが使用するクロックソースを選択します。

0： LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。

1： LPTIMのクロックは、LPTIM外部Input1を通じて外部クロックソースによって供給されます。

注意： LPTIM_CFGRレジスタの変更は、LPTIMが無効（イネーブルビットが0にリセットされている）のときにのみ行う必要があります。

58.7.11 LPTIM制御レジスタ（LPTIM_CR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
											rw	rs	rw	rw	rw

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **RSTARE**：読出し後リセット有効化

このビットは、ソフトウェアによってセット／クリアされます。RSTAREが“1”にセットされると、LPTIM_CNTレジスタに対するあらゆる読出しアクセスによって、LPTIM_CNTレジスタの内容が非同期的にリセットされます。

このビットをセットできるのは、LPTIMが有効なときだけです。

ビット3 **COUNTRST**：カウンタリセット

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。“1”にセットされると、このビットはLPTIM_CNTカウンタレジスタの同期リセットをトリガします。このリセットの同期特性のため、3 LPタイマコアクロックサイクルの同期遅延の後にのみ、この状況が発生します（LPタイマコアクロックはAPBクロックと異なる場合があります）。

このビットをセットできるのは、LPTIMが有効なときだけです。これはハードウェアによって自動的にリセットされます。

注意： COUNTRSTは、ハードウェアによって“0”にクリアされる前に、ソフトウェアによって“1”にセットしてはなりません。したがって、ソフトウェアでCOUNTRSTビットを“1”にセットしようとする前に、それがすでに“0”にクリアされていることを確認する必要があります。

ビット2 **CNTSTRT**：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIMは連続モードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が00以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。

シングルパルスモードでのカウント中にこのビットがセットされた場合、LPTIM_ARRレジスタとLPTIM_CNTレジスタが次に一致したときにタイマは停止せず、LPTIMカウンタは連続モードでのカウントを続行します。

このビットをセットできるのは、LPTIMが有効なときだけです。これはハードウェアによって自動的にリセットされます。

ビット1 **SNGSTRT**：シングルモードでLPTIM開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIMはシングルパルスモードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が00以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIMはシングルパルスモードで開始します。

LPTIMが連続カウントモードのときにこのビットがセットされた場合、LPTIMはLPTIM_ARRレジスタとLPTIM_CNTレジスタが次に一致したときに停止します。

このビットをセットできるのは、LPTIMが有効なときだけです。これはハードウェアによって自動的にリセットされます。

ビット0 **ENABLE**：LPTIM有効化

イネーブルビットは、ソフトウェアによってセット／クリアされます。

0：LPTIMは無効です。ENABLEビットに "0" を書き込むと、すべてのDMAリクエスト信号（入力キャプチャおよび更新イベントDMAリクエスト）がリセットされます。

1：LPTIMは有効です。

58.7.12 LPTIM比較レジスタ 1（LPTIM_CCR1）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CCR1[15:0]**：キャプチャ／比較1値

CC1チャネルが出力として設定されている場合：

CCR1は、キャプチャ／比較1レジスタにロードされる値です。

PRELOADオプションに応じて、CCR1レジスタはPRELOADビットがリセットされた場合は即座に更新され、PREOADビットがリセットされた場合は次のLPTIM更新イベント時に更新されます。

キャプチャ／比較レジスタ1は、カウンタLPTIM_CNTと比較されて、OC1出力に送信される値を含みます。

CC1チャネルが入力として設定されている場合：

CCR1は読出し専用となり、最後の入力キャプチャ1イベントによって転送されたカウンタ値を含みます。LPTIM_CCR1レジスタは読出し専用レジスタで、プログラムできません。

LPTIMがチャネルを実装していない場合：

比較レジスタ1は、カウンタのLPTIM_CNTと比較されて、LPTIM出力に送信される値を含みます。

注意： LPTIM_CCR1 レジスタの変更は、LPTIMが有効な（ENABLEビットが1にセットされている）ときにのみ行う必要があります。

58.7.13 LPTIM自動再ロードレジスタ（LPTIM_ARR）

アドレスオフセット：0x018

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **ARR[15:0]**：自動再ロード値

ARRは、LPTIMの自動再ロード値です。

この値は、CCRx[15:0] 値より大きくなければなりません。

注意： **LPTIM_ARRレジスタの変更は、LPTIMが有効（イネーブルビットが1にセットされている）のときにのみ行う必要があります。**

58.7.14 LPTIMカウンタレジスタ（LPTIM_CNT）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CNT[15:0]**：カウンタ値

LPTIMが非同期クロックで実行しているとき、LPTIM_CNTレジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2つの連続した読出しアクセスを実行して、返された2つの値が同じかどうかを確認する必要があります。

58.7.15 LPTIM設定レジスタ2（LPTIM_CFGR2）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2SEL [1:0]		Res.	Res.	IC1SEL [1:0]	
										rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL [1:0]		Res.	Res.	IN1SEL [1:0]	
										rw	rw			rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21:20 **IC2SEL[1:0]**：LPTIM入力 キャプチャ 2選択

IC2SELビットは、LPTIM入力キャプチャ 2を利用可能な入力のうちの1つに接続するLPTIM入力キャプチャ 2マルチプレクサを制御します。

00：lptim_ic2_mux0

01：lptim_ic2_mux1

10：lptim_ic2_mux2

11：lptim_ic2_mux3

接続の詳細については、セクション 58.4.3：LPTIM入力およびトリガマッピングを参照してください。

ビット19:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **IC1SEL[1:0]**：LPTIM入力 キャプチャ 1選択

IC1SELビットは、LPTIM入力キャプチャ 1を利用可能な入力のうちの1つに接続するLPTIM入力キャプチャ 1マルチプレクサを制御します。

★

00：lptim_ic1_mux0

01：lptim_ic1_mux1

10：lptim_ic1_mux2

11：lptim_ic1_mux3

接続の詳細については、セクション 58.4.3：LPTIM入力およびトリガマッピングを参照してください。

ビット15:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **IN2SEL[1:0]**：LPTIM入力2選択

IN2SELビットは、LPTIM入力2を利用可能な入力のうちの1つに接続するLPTIM入力2マルチプレクサを制御します。

00：lptim_in2_mux0

01：lptim_in2_mux1

10：lptim_in2_mux2

11：lptim_in2_mux3

接続の詳細については、セクション 58.4.3：LPTIM入力およびトリガマッピングを参照してください。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **IN1SEL[1:0]**：LPTIM入力1選択

IN1SELビットは、LPTIM入力1を利用可能な入力のうちの1つに接続するLPTIM入力1マルチプレクサを制御します。

00：lptim_in1_mux0

01：lptim_in1_mux1

10：lptim_in1_mux2

11：lptim_in1_mux3

接続の詳細については、セクション 58.4.3：LPTIM入力およびトリガマッピングを参照してください。

58.7.16 LPTIM繰返しレジスタ（LPTIM_RCR）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **REP[7:0]**：繰返しレジスタ値
REPはLPTIMの繰返し値です。

注意： **LPTIM_RCRレジスタの変更は、LPTIMが有効（イネーブルビットが1にセットされている）のときにのみ行う必要があります。PRELOAD = 0で繰返しカウンタを使用する場合、自動再ロード一致イベントの少なくとも5カウンタサイクル前にLPTIM_RCRレジスタを変更する必要があります。そうしないと、予測不能な動作が発生することがあります。**

58.7.17 LPTIMキャプチャ／比較モードレジスタ1（LPTIM_CCMR1）

アドレスオフセット：0x02C

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（PWMモード）で使用できます。チャネルの方向は対応するCCxSELビットを設定することで定義されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	IC2F [1:0]		Res.	Res.	IC2PSC [1:0]		Res.	Res.	Res.	Res.	CC2P [1:0]		CC2E	CC2 SEL
		rw	rw			rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	IC1F [1:0]		Res.	Res.	IC1PSC [1:0]		Res.	Res.	Res.	Res.	CC1P [1:0]		CC1E	CC1 SEL
		rw	rw			rw	rw					rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29:28 **IC2F[1:0]**：入力キャプチャ 2フィルタ
このビットフィールドにより、外部入力キャプチャ信号上でレベル変化が発生した時に、有効なレベル遷移とみなすまで検出する同じサンプルの連続数を定義します。この機能を使用するには、内部クロックソースが必要です。
00：外部入力キャプチャ信号のいかなるレベル変化も、有効な遷移とみなされます。
01：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも2クロック周期にわたって安定している必要があります。
10：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも4クロック周期にわたって安定している必要があります。
11：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも8クロック周期にわたって安定している必要があります。

ビット27:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **IC2PSC[1:0]**：入力キャプチャ 2プリスケーラ

このビットフィールドは、CC2入力（IC2）に作用するプリスケーラの比を定義します。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2イベントごとに行われます。

10：キャプチャは、4イベントごとに行われます。

11：キャプチャは、8イベントごとに行われます。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:18 **CC2P[1:0]**：キャプチャ /比較2出力極性。

条件：出力としてのCC2

出力モードが有効になっている場合、ビット2のみを使用して極性を設定します。ビット3は無視されます。

0：OC2はアクティブハイです。

1：OC2はアクティブローです。

条件：入力としてのCC2

このフィールドは、キャプチャ動作でのIC2の極性を選択するために使用されます。

00：立上がりエッジ。回路はIC2の立上がりエッジに反応します。

01：立下がりエッジ。回路はIC2の立下がりエッジに反応します。

10：予約済み。この設定は使用しないでください。

11：両方のエッジ。回路はIC2の立上がりエッジと立下がりエッジの両方に反応します。

ビット17 **CC2E**：キャプチャ /比較2出力イネーブル。

条件：出力としてのCC2

0：オフ - OC2はアクティブではありません。CC2Eビットに "0" を書き込むと、他のすべてのLPTIMチャネルが無効になっている場合にのみ、ue_dma_req信号がリセットされます。

1：オン - OC2信号は、対応する出力ピンに出力されます。

条件：入力としてのCC2

このビットによって、カウンタ値のキャプチャ／比較レジスタ2（LPTIM_CCR2）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。このCC2Eビットに "0" を書き込むと、関連するic2_dma_req信号がリセットされます。

1：キャプチャは有効です。

ビット16 **CC2SEL**：キャプチャ／比較2選択

このビットフィールドは、チャネルの方向、すなわち、入力（キャプチャ）モードか、出力モードかを定義します。

0：CC2チャネルは出力PWMモードに設定されます。

1：CC2チャネルは入力キャプチャモードに設定されます。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **IC1F[1:0]**：入力キャプチャ 1フィルタ

このビットフィールドにより、外部入力キャプチャ信号上でレベル変化が発生した時に、有効なレベル遷移とみなすまで検出する同じサンプルの連続数を定義します。この機能を使用するには、内部クロックソースが必要です。

00：外部入力キャプチャ信号のいかなるレベル変化も、有効な遷移とみなされます。

01：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも2クロック周期にわたって安定している必要があります。

10：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも4クロック周期にわたって安定している必要があります。

11：外部入力キャプチャ信号のレベル変化が有効な遷移とみなされるためには、少なくとも8クロック周期にわたって安定している必要があります。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **IC1PSC[1:0]**：入力キャプチャ 1プリスケーラ

このビットフィールドは、CC1入力（IC1）に作用するプリスケーラの比を定義します。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2イベントごとに行われます。

10：キャプチャは、4イベントごとに行われます。

11：キャプチャは、8イベントごとに行われます。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:2 **CC1P[1:0]**：キャプチャ /比較1出力極性。

条件：出力としてのCC1

出力モードが有効になっている場合、ビット2のみを使用して極性を設定します。ビット3は無視されます。

0 :OC1はアクティブハイです。LPTIM出力はLPTIM_ARRレジスタとLPTIM_CCRxレジスタの比較結果を反映します。

1 :OC1はアクティブローです。LPTIM出力はLPTIM_ARRレジスタとLPTIM_CCRxレジスタの比較結果の反転を反映します。

条件：入力としてのCC1

このフィールドは、キャプチャ動作でのIC1の極性を選択するために使用されます。

00：立上がりエッジ。回路はIC1の立上がりエッジに反応します。

01：立下がりエッジ。回路はIC1の立下がりエッジに反応します。

10：予約済み。この設定は使用しないでください。

11：両方のエッジ。回路はIC1の立上がりエッジと立下がりエッジの両方に反応します。

ビット1 **CC1E**：キャプチャ /比較1出力イネーブル。

条件：出力としてのCC1

0 : オフ - OC1はアクティブではありません。CC1Eビットに "0" を書き込むと、他のすべてのLPTIMチャネルが無効になっている場合にのみ、ue_dma_req信号がリセットされます。

1 : オン - OC1信号は、対応する出力ピンに出力されます。

条件：入力としてのCC1

このビットによって、カウンタ値のキャプチャ／比較レジスタ1（LPTIM_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0 : キャプチャは無効です。このCC1Eビットに "0" を書き込むと、関連するic1_dma_req信号がリセットされます。

1 : キャプチャは有効です。

ビット0 **CC1SEL**：キャプチャ／比較1選択

このビットフィールドは、チャネルの方向、すなわち、入力（キャプチャ）モードか、出力モードかを定義します。

0：CC1チャネルは出力PWMモードに設定されます。

1：CC1チャネルは入力キャプチャモードに設定されます。

注意： **LPTIM_CCMRxレジスタへの書込み後、同じレジスタへの新しい書込み操作は、（PRESC × 3）カーネルクロックサイクルの値以上の遅延の後にのみ実行できます。PRESC[2:0] は、クロックの10進数の分周比（1、2、4、..128）です。この遅延を守らずに続けて書込みを行うと、予測不能な結果になります。**

注意： **CCxSEL、ICxF[1:0]、CCxP[1:0]、およびICxPSC[1:0] フィールドは、チャネルxが無効になっている（CCxEビットが0にリセットされている）場合にのみ変更する必要があります。**

LPTIMがチャネルを実装していない場合、このレジスタは予約済みです。セクション58.3 を参照してください。

58.7.18 LPTIM比較レジスタ2（LPTIM_CCR2）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **CCR2[15:0]**：キャプチャ／比較2値

CC2チャネルが出力として設定されている場合：

CCR2は、キャプチャ／比較2レジスタにロードされる値です。

PRELOADオプションに応じて、CCR2レジスタはPRELOADビットがリセットされた場合は即座に更新され、PREOADビットがリセットされた場合は次のLPTIM更新イベント時に更新されます。

キャプチャ／比較レジスタ2は、カウンタLPTIM_CNTと比較されて、OC2出力に送信される値を含みます。

CC2チャネルが入力として設定されている場合：

CCR2は読出し専用となり、最後の入力キャプチャ2イベントによって転送されたカウンタ値を含みます。

LPTIM_CCR2レジスタは読出し専用レジスタで、プログラムできません。

注意： **LPTIM_CCR2レジスタの変更は、LPTIMが有効な（ENABLEビットが1にセットされている）ときにのみ行う必要があります。**

注： **LPTIM が実装しているチャネルが 2 未満の場合、このレジスタは予約済みです。セクション 58.3：LPTIMの実装を参照してください。**

58.7.19 LPTIMレジスタマップ

次の表にLPTIMレジスタの一覧を示します。

表610.LPTIMレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	LPTIM4_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPOK	UE	DOWN (2)	UP (2)	ARROK	CMP1OK	EXTTRIG	ARRM	CC1IF
	リセット値								0																0	0	0	0	0	0	0	0	0
0x000	LPTIMx_ISR（x = 1～3）出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROK	Res.	Res.	Res.	Res.	CMP2OK (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IF (1)	REPOK	UE	DOWN (2)	UP (2)	ARROK	CMP1OK	EXTTRIG	ARRM	CC1IF
	リセット値								0					0										0	0	0	0	0	0	0	0	0	0
	LPTIMx_ISR（x = 1～3）入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF (1)	CC1OF	Res.	Res.	CC2IF (1)	REPOK	UE	DOWN (2)	UP (2)	ARROK	Res.	EXTTRIG	ARRM	CC1IF
	リセット値								0											0	0			0	0	0	0	0	0		0	0	0
0x004	LPTIM4_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPOKCF	UECF	DOWNCF (2)	UPCF (2)	ARROKCF	CMP1OKCF	EXTTRIGCF	ARRMCF	CC1CF
	リセット値								0																0	0	0	0	0	0	0	0	0
0x004	LPTIMx_ICR（x = 1～3）出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	CMP2OKCF (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2CF (1)	REPOKCF	UECF	DOWNCF (2)	UPCF (2)	ARROKCF	CMP1OKCF	EXTTRIGCF	ARRMCF	CC1CF
	リセット値								0					0										0	0	0	0	0	0	0	0	0	0
	LPTIMx_ICR（x = 1～3）入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIEROKCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OCF (1)	CC1OCF	Res.	Res.	CC2CF (1)	REPOKCF	UECF	DOWNCF (2)	UPCF (2)	ARROKCF	Res.	EXTTRIGCF	ARRMCF	CC1CF
	リセット値								0											0	0			0	0	0	0	0	0		0	0	0
0x008	LPTIM4_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPOKIE	UEIE	DOWNIE (2)	UPIE (2)	ARROKIE	CMP1OKIE	EXTTRIGIE	ARRMIE	CC1IE
	リセット値																								0	0	0	0	0	0	0	0	0
0x008	LPTIMx_DIER（x = 1～3）出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UEDE	Res.	Res.	Res.	CMP2OKIE (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IE (1)	REPOKIE	UEIE	DOWNIE (2)	UPIE (2)	ARROKIE	CMP1OKIE	EXTTRIGIE	ARRMIE	CC1IE
	リセット値									0				0										0	0	0	0	0	0	0	0	0	0
	LPTIMx_DIER（x = 1～3）入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	CC2DE (1)	Res.	UEDE	Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	Res.	Res.	CC2OIE (1)	CC1OIE	Res.	Res.	CC2IE (1)	REPOKIE	UEIE	DOWNIE (2)	UPIE (2)	ARROKIE	Res.	EXTTRIGIE	ARRMIE	CC1IE
	リセット値							0		0							0			0	0			0	0	0	0	0	0		0	0	0
0x00C	LPTIM_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC (2)	COUNTMODE	PRELOAD	WAVPOL (3)	WAVE	TIMOUT	TRIGEN		Res.	TRIGSEL[2:0]			Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL
	リセット値								0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0		0	0	0	0	0
0x010	LPTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
	リセット値																												0	0	0	0	0

表610.LPTIMレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x014	LPTIM_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	LPTIM_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x01C	LPTIM_CNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	LPTIM_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2SEL [1:0]		Res.	Res.	IC1SEL [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL [1:0]		Res.	Res.	IN1SEL [1:0]	
	リセット値											0	0			0	0											0	0			0	0
0x028	LPTIM_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x02C	LPTIM_CCMR1(4)	Res.	Res.	IC2F [1:0]		Res.	Res.	IC2PSC [1:0]		Res.	Res.	Res.	Res.	CC2P [1:0]		CC2E	CC2SEL	Res.	Res.	IC1F [1:0]		Res.	Res.	IC1PSC [1:0]		Res.	Res.	Res.	Res.	CC1P [1:0]		CC1E	CC1SEL
	リセット値			0	0			0	0					0	0	0	0			0	0			0	0					0	0	0	0
0x034	LPTIM_CCR2(5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. LPTIMが少なくとも2つのチャネルを実装していない場合､このビットは予約済みです。セクション 58.3:LPTIMの実装を参照してください。
2. LPTIMがエンコーダモード機能をサポートしない場合、このビットは予約済みです。セクション 58.3 : LPTIMの実装を参照してください。
3. LPTIMが少なくとも1つのキャプチャ／比較チャネルを実装している場合、このビットは予約済みです。セクション 58.3 : LPTIMの実装を参照してください。
4. LPTIMがチャネルを実装していない場合、このレジスタは予約済みです。セクション 58.3 : LPTIMの実装 を参照してください。
5. LPTIMが実装しているチャネルが2未満の場合、このレジスタは予約済みです。セクション 58.3 : LPTIMの実装 を参照してください。

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

59 グラフィック･タイマ（GFXTIM）

このセクションは、STM32U5Fx/5Gxデバイスにのみ適用されます。

59.1 概要

グラフィック･タイマ（GFXTIM）はグラフィック指向のタイマで、フレームまたはラインのカウントのためのグラフィカル･イベントをスマートに管理できるようにします。

59.2 GFXTIMの主な機能

- フレームおよびラインの統合されたクロック生成
- 1つの比較チャネルを持つ1つの絶対フレーム･カウンタ
- 2つの自動再ロード相対フレーム･カウンタ
- 2つの比較チャネルを持つ1つのライン･タイマ
- 外部ティアリング効果によるライン管理および同期
- 外部トリガ生成機能のある4つのプログラム可能なイベント･ジェネレータ
- 1つのウォッチドッグカウンタ

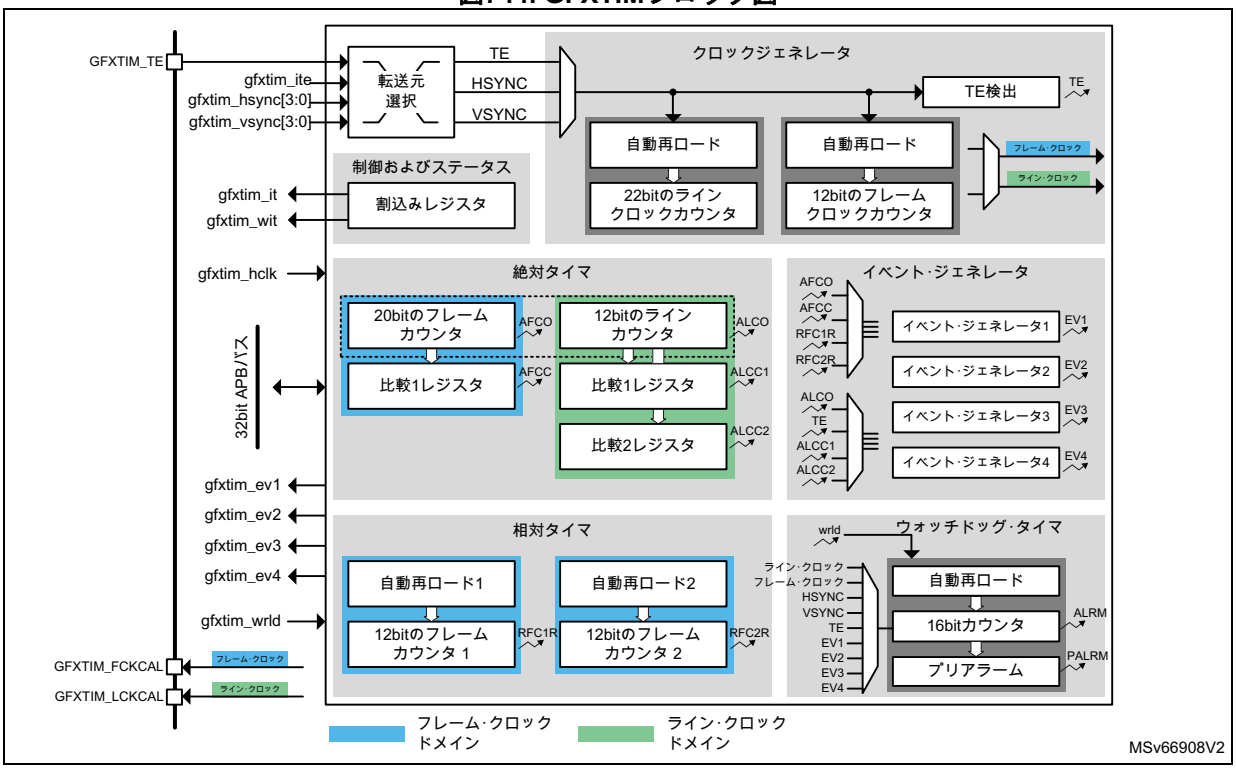
59.3 GFXTIMの機能詳細

59.3.1 ブロック図

グラフィック･タイマは6つの機能ブロックに分割されます。

- クロックジェネレータ
- 絶対タイマ
- 相対タイマ
- ティアリング効果の検出
- イベント･ジェネレータ
- ウォッチドッグ･タイマ

図744. GFXTIMブロック図



59.3.2 GFXTIMピンおよび内部信号

表611.GFXTIMの入出力ピン

ピン名	信号タイプ	説明
GFXTIM_TE	入力	引裂効果
GFXTIM_FCKCAL	出力	フレーム･クロック較正出力
GFXTIM_LCKCAL	出力	ライン･クロック較正出力

表612.GFXTIM内部信号

信号名	信号タイプ	説明
gfxtim_hclk	デジタル入力	カーネル／レジスタインタフェースクロック
gfxtim_it	デジタル出力	グローバル割込み
gfxtim_wit	デジタル出力	ウォッチドッグ･グローバル割込み
gfxtim_ev1	デジタル出力	グラフィック･タイマ･イベント1
gfxtim_ev2	デジタル出力	グラフィック･タイマ･イベント2
gfxtim_ev3	デジタル出力	グラフィック･タイマ･イベント3
gfxtim_ev4	デジタル出力	グラフィック･タイマ･イベント4
gfxtim_wrld	デジタル入力	ウォッチドッグの再ロード
gfxtim_ite	デジタル入力	内部ティアリング効果

表612.GFXTIM内部信号 （続き）

信号名	信号タイプ	説明
gfxtim_hsync[3:0]	デジタル入力	水平同期
gfxtim_vsync[3:0]	デジタル入力	垂直同期

次の表に、GFXTIMトリガの接続方法を示します。

表613.GFXTIMトリガの相互接続

トリガ名	方向	トリガの転送元/転送先
gfxtim_ev1	出力	gpdma_trigsel[62]
gfxtim_ev2	出力	gpdma_trigsel[61]
gfxtim_ev3	出力	gpdma_trigsel[60]
gfxtim_ev4	出力	gpdma_trigsel[59]
gfxtim_wrld	入力	予約済み
gfxtim_ite	入力	予約済み
gfxtim_hsync[0]	入力	LCD_HSYNC
gfxtim_hsync[1]	入力	予約済み
gfxtim_hsync[2]	入力	予約済み
gfxtim_hsync[3]	入力	予約済み
gfxtim_vsync[0]	入力	LCD_VSYNC
gfxtim_vsync[1]	入力	予約済み
gfxtim_vsync[2]	入力	予約済み
gfxtim_vsync[3]	入力	予約済み

59.3.3 クロックジェネレータ

2つのクロックが、絶対タイマと相対タイマを供給するために内部的に生成されています。

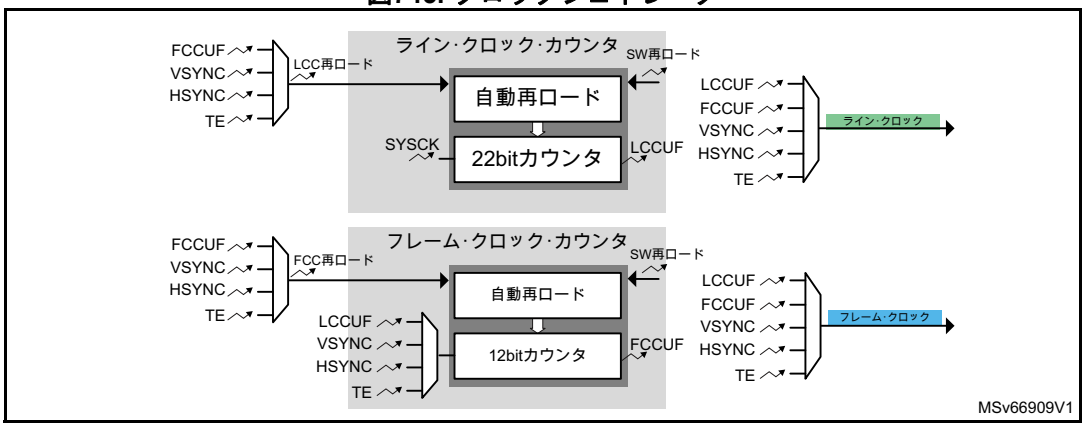
- フレーム･クロック：フレーム･カウンタのクロック供給
- ライン･クロック：ライン･カウンタのクロック供給

タイムベースの生成用の内部カウンタ

GFXTIMには、2つのクロック生成カウンタが組み込まれています。

- システムクロックの22bitの自動再ロードダウンカウンタ
- 選択可能な内部または外部イベントの12bitの自動再ロードダウンカウンタ

図745. クロックジェネレータ



ライン･クロック･カウンタ

ライン･クロック･カウンタは、システムクロックの22bitの自動再ロードダウンカウンタです。

ライン･クロック･カウンタは、GFXTIM_CGCR（クロックジェネレータ制御レジスタ）のLCCCS（ライン･クロック･カウンタのクロックソース）でクロックソースを選択して有効にします。

ライン･クロック･カウンタは、次のイベントのいずれかが発生したときに、自動的に再ロードできます。

- フレーム･クロック･カウンタ･アンダーフロー
- VSYNCエッジ（極性は制御される）
- HSYNCエッジ（極性は制御される）
- TEエッジ（極性は制御される）

ハードウェアの再ロード機能およびソースは、GFXTIM_CGCR（クロックジェネレータ制御レジスタ）のLCCHRS（ライン･クロック･カウンタのハードウェア再ロードのソース）で選択可能です。

ライン･クロック･カウンタは、GFXTIM_CGCR（クロックジェネレータ制御レジスタ）のLCCFR（ライン･クロック･カウンタの強制再ロード）をセットすることで、ソフトウェアによって再ロードできます。

再ロード値は、GFXTIM_LCCRR（ライン･クロック･カウンタの再ロードレジスタ）でプログラムされます。

フレーム･クロック･カウンタ

12bitの自動再ロード･ダウンカウンタが、TE、HYSNC、VSYNC、またはライン･クロック･カウンタ･アンダーフローのいずれかによってクロック供給される場合は、フレーム･クロック･カウンタです。

フレーム･クロック･カウンタは、GFXTIM_CGCRのFCCCSでクロックソースを選択して有効にします。

フレーム･クロック･カウンタは、次のイベントのいずれかが発生したときに、自動的に再ロードできます。

- ライン･クロック･カウンタ･アンダーフロー
- VSYNCエッジ（極性は制御される）
- HSYNCエッジ（極性は制御される）
- TEエッジ（極性は制御される）

ハードウェアの再ロード機能およびソースはGFXTIM_CGCRレジスタのFCCHRSで選択可能です。

フレーム･クロック･カウンタは、GFXTIM_CGCRのFCCFRをセットすることで、ソフトウェアによって再ロードできます。

再ロード値はGFXTIM_FCCRRでプログラムされます。

クロック生成

ラインのクロックソースは次のいずれかです。

- 内部22bitダウンカウンタのアンダーフロー･フラグ
- 内部12bitダウンカウンタのアンダーフロー･フラグ
- HSYNC、VSYNC、またはTEピン（極性は制御される）

フレームのクロックソースは次のいずれかです。

- 内部22bitダウンカウンタのアンダーフロー･フラグ
- 内部12bitダウンカウンタのアンダーフロー･フラグ
- HSYNC、VSYNC、またはTEピン（極性は制御される）

クロック較正出力

較正/デバッグ目的で、フレーム･クロックとライン･クロックを特定のI/Oに出力できます。

フレーム･クロックの較正出力は、GFXTIM_CR（設定レジスタ）のFCCOE（フレーム･クロック較正出力イネーブル）をセットすることによって有効になります。

ライン･クロック較正出力は、GFXTIM_CRのLCCOEをセットすることによって有効になります。

同期およびティアリング効果のソース

GFXTIMは、LCD-TFTコントローラやカメラ･インタフェースなどのHSYNCおよび/またはVSYNC同期信号を提供することでペリフェラルに接続できます。

HSYNCおよび/またはVSYNCのソースは、GFXTIM_CRのSYNCS（同期ソース）を通じて選択されます。

ティアリング効果のソースは、外部ピン、またはこのインタフェースに組み込まれたシステムのMIPI® DSI Hostが提供するものである場合があります。同期信号に対して直交性を拡張するには、選択されたHSYNCまたはVSYNC入力をティアリング効果のソースとすることもできます。

ティアリング効果のソースは、GFXTIM_CRのTES（TEソース）で選択可能です。

59.3.4 クロックジェネレータ設定の例

クロックジェネレータでは、次の複数の作業用設定を使用できます。

- スタンドアロン（外部同期なし）
- 外部HSYNCおよびVSYNC
- 外部HSYNCのみ
- 外部VSYNCのみ
- 外部CSYNC（TE）のみ

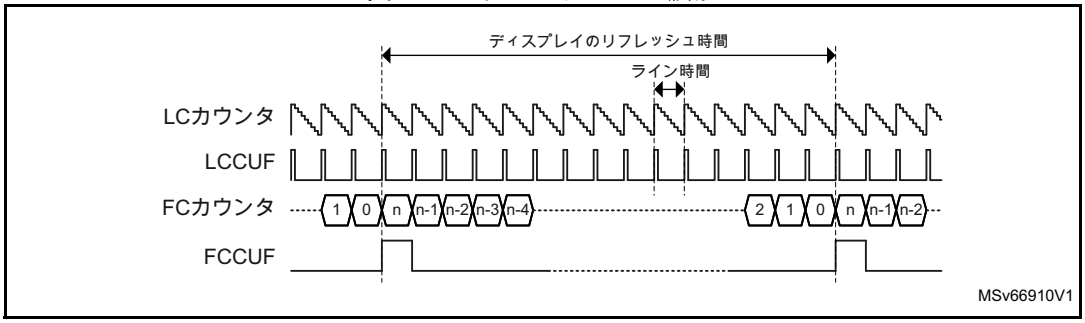
同期信号HSYNC、VSYNC、およびTEは、対称的に実装され、必要に応じて入れ替えることもできます。

参考として、以下に例を示します。クロックジェネレータで他の組み合わせをプログラムすることもできます。

スタンドアロン

スタンドアロン設定では、クロックジェネレータが外部信号なしでGFXTIMにフレーム·クロックとライン·クロックを提供します。

図746. スタンドアロンの波形

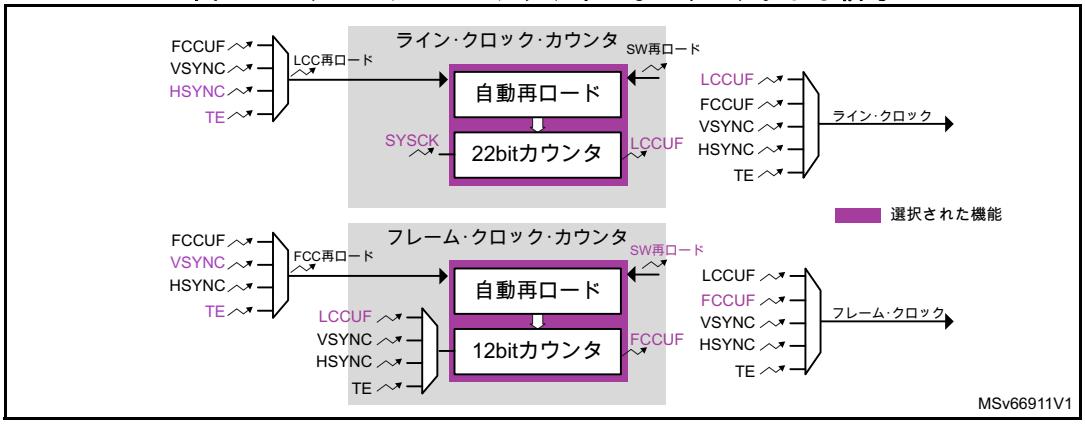


LCCUF（ライン·クロック·カウンタ·アンダーフロー）イベントはライン·クロックとして動作し、FCCUF（フレーム·クロック·カウンタ·アンダーフロー）はフレーム·クロックとして動作します。

クロックジェネレータは、次のいずれかの方法で同期させることができます。

- GFXTIM_CGCRのFCCFRをセットしてソフトウェアによって再ロードする
- 外部VSYNC（あるいはTE）でハードウェアによって再ロードする

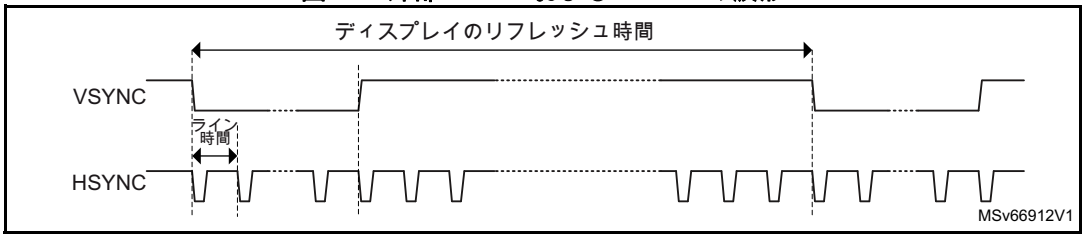
図747. スタンドアロンでアクティブなカウンタおよび信号



外部HSYNCおよびVSYNC

外部HSYNCおよびVSYNCを使用するときは、カウンタはラインおよびフレームのクロック生成に使用されません。クロックジェネレータは、直接ライン·クロックにHSYNCを、フレーム·クロックにVSYNCをコピーします。

図748. 外部HSYNCおよびVSYNCの波形

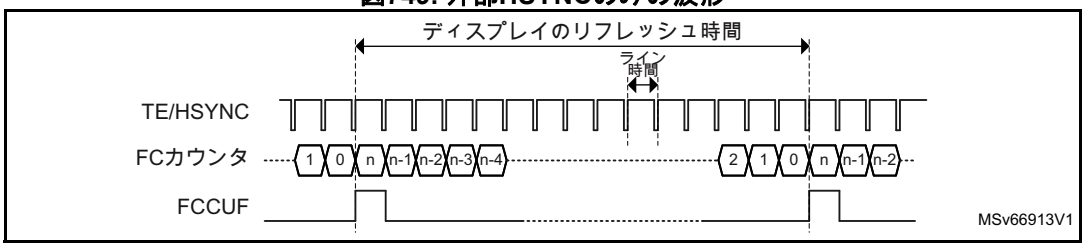


HSYNCはライン·クロックとして動作し、VSYNCはフレーム·クロックとして動作します。

外部HSYNCのみ

外部HSYNCのみの設定では、クロックジェネレータがHSYNCのみに基づいてGFXTIMにフレーム·クロックとライン·クロックを提供します。

図749. 外部HSYNCのみの波形

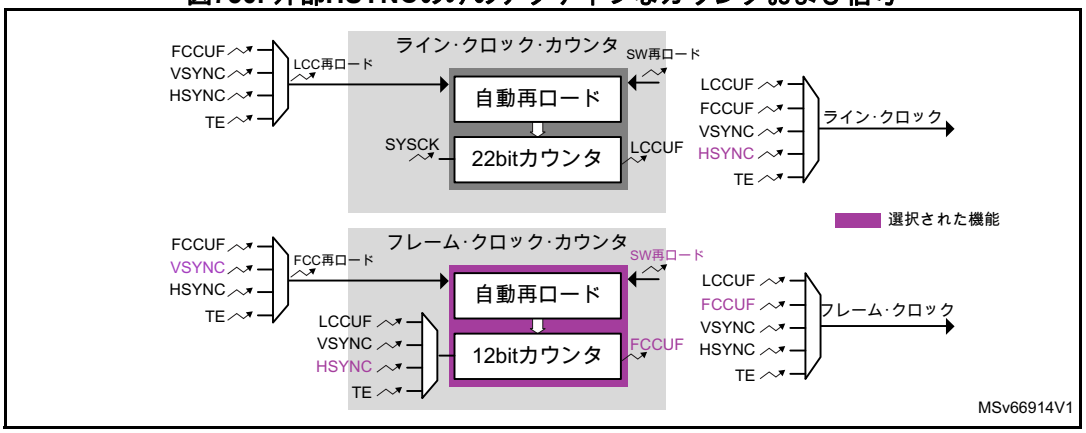


HSYNCはライン·クロックとして動作し、FCCUFはフレーム·クロックとして動作します。

クロックジェネレータは、次のいずれかの方法で同期させることができます。

- GFXTIM_CGCRのFCCFRをセットしてソフトウェアによって再ロードする
- 外部VSYNC（TE）でハードウェアによって再ロードする

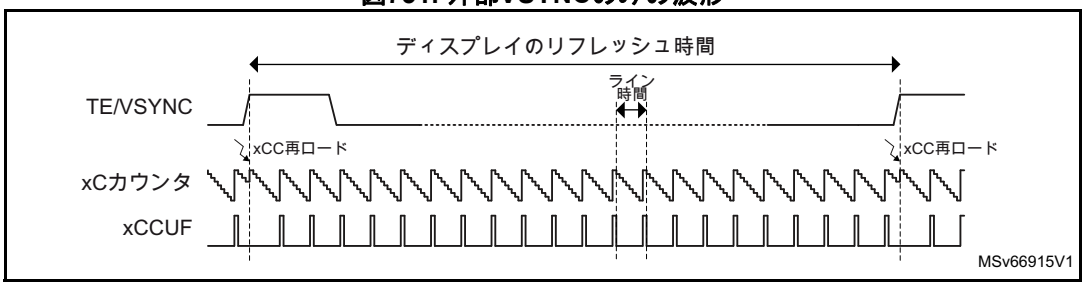
図750. 外部HSYNCのみのアクティブなカウンタおよび信号



外部VSYNCのみ

外部VSYNCのみの設定では、クロックジェネレータがVSYNCのみに基づいてGFXTIMにフレーム·クロックとライン·クロックを提供します。

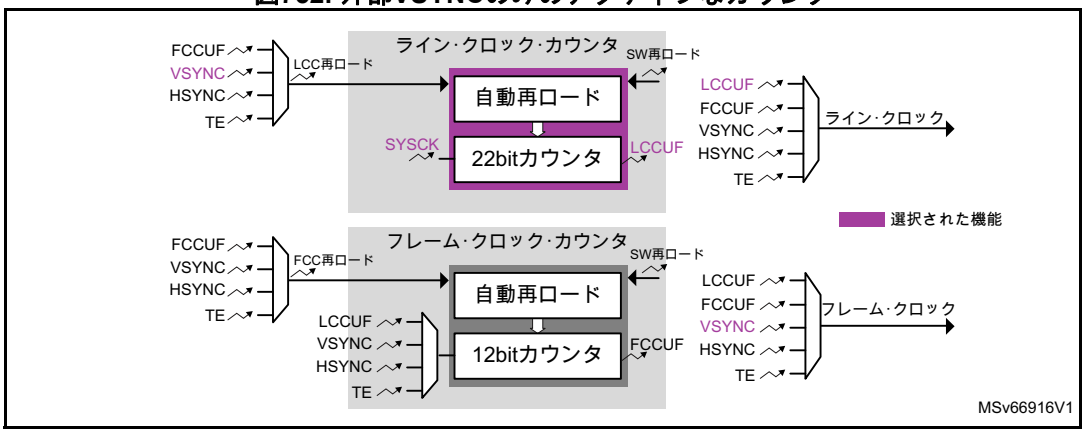
図751. 外部VSYNCのみの波形



LCCUF（ライン·クロック·カウンタ·アンダーフロー）はライン·クロックとして動作し、VSYNCはフレーム·クロックとして動作します。

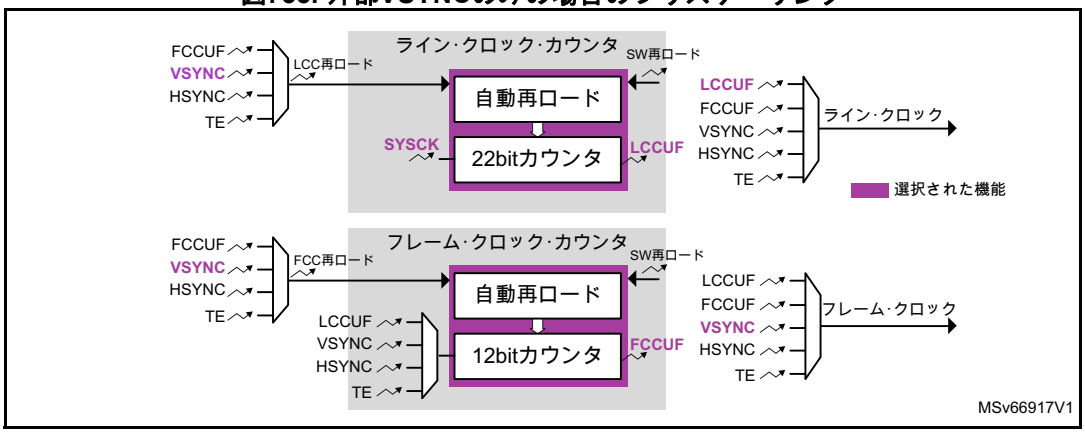
ライン·クロックを生成するライン·クロック·カウンタがVSYNCイベントで再ロードされます。

図752. 外部VSYNCのみのアクティブなカウンタ



ライン·クロック·カウンタは、フレーム·クロック·カウンタのカウント範囲を広げるために、プリスケーラとしても動作します。FCCUFはライン·クロックとして動作します。

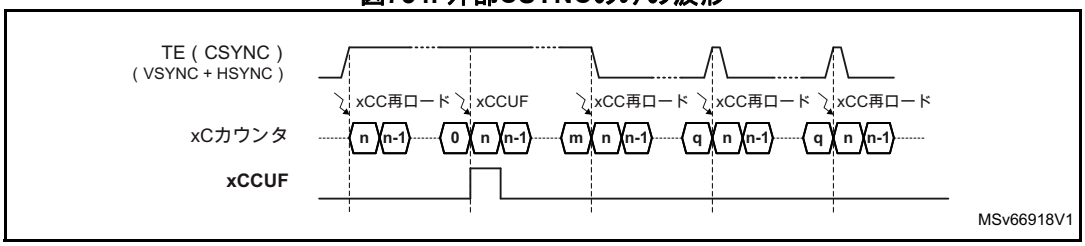
図753. 外部VSYNCのみの場合のプリスケーリング



外部CSYNCのみ

外部CSYNCのみの設定では、クロックジェネレータがCSYNCのみに基づいてGFXTIMにフレーム·クロックとライン·クロックを提供します（以下の図で、CSYNCはTEに入力される）。

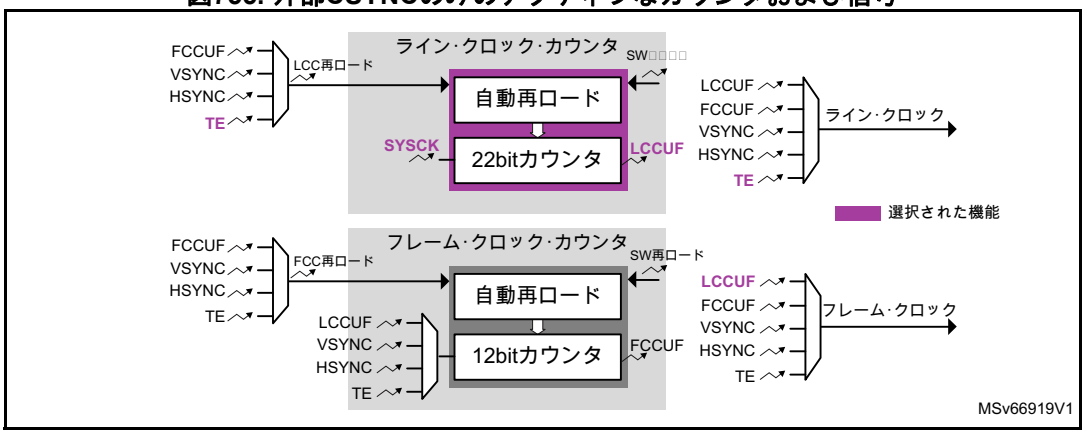
図754. 外部CSYNCのみの波形



CSYNC（この例のTEピン上）はライン·クロックとして動作し、LCCUFはフレーム·クロックとして動作します。

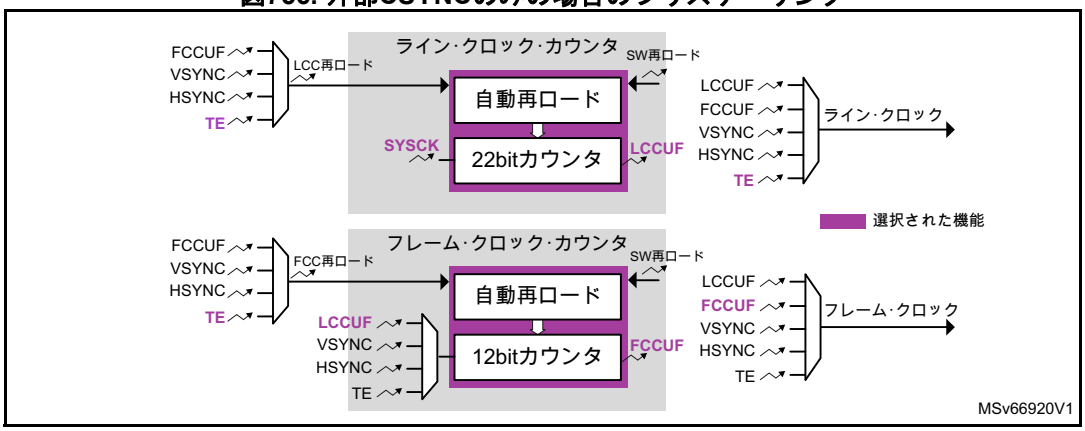
フレーム･クロックを生成するライン･クロック･カウンタがCSYNCイベントで再ロードされます。

図755. 外部CSYNCのみのアクティブなカウンタおよび信号



ライン･クロック･カウンタは、フレーム･クロック･カウンタのカウント範囲を広げるために、プリスケーラとしても動作します。FCCUFはフレーム･クロックとして動作します。

図756. 外部CSYNCのみの場合のプリスケーリング



59.3.5 絶対タイマ

絶対時間

絶対時間の32bit値は、グローバルタイムを提供し、グラフィカル･ソフトウェアを同期して正確な絶対時間のイベントを生成するために、直接GFXTIM_ATR（絶対時間レジスタ）で読み出すことができます。

絶対時間は以下の組み合わせで生成されます。

- 絶対フレーム･カウンタ
- 絶対ライン･カウンタ

絶対フレーム･カウンタ

絶対フレーム･カウンタは20bitのフリーランニングアップカウンタで、リセットはソフトウェアで制御されます。

GFXTIM_TCR（タイマ設定レジスタ）のAFCEN（絶対フレーム･カウンタ･イネーブル）に1を書き込むことによって有効にできます。

カウンタは、フレーム･クロックの立ち上がりエッジのたびにインクリメントされ、GFXTIM_TCRのFAFCR（絶対フレーム･カウンタの強制リセット）をセットすることによってリセットされます。

絶対フレーム･カウンタがオーバーフローした場合は、GFXTIM_ISR（割込みステータスレジスタ）のAFCOF（オーバーフローフラグ）がセットされます。GFXTIM_IER（割込み有効レジスタ）のAFCOIE（オーバー割込みイネーブル）がセットされている場合は、割込みが生成されます。

カウンタ値が比較値と一致する場合、20bitの比較レジスタで、絶対フレーム･カウンタの比較イベントを生成できます。比較イベントが発生すると、GFXTIM_ISRのAFCC1F（比較1フラグ）がセットされます。GFXTIM_IERのAFCC1IE（比較1割込みイネーブル）がセットされている場合は、割込みが生成されます。

絶対フレーム･カウンタは、GFXTIM_TDR（タイマ無効化レジスタ）のAFCDIS（カウンタディスエーブル）に1を書き込むことによって無効にできます。

ステータス（有効/無効）は、GFXTIM_TSR（タイマのステータスレジスタ）のAFCS（カウンタのステータス）を通じて監視できます。

絶対ライン･カウンタ

絶対ライン･カウンタは12bitのフリーランニングアップカウンタで、リセットはハードウェアで制御されます。

GFXTIM_TCRのALCENに1を書き込むことによって有効にできます。

カウンタは、ライン･クロックの立ち上がりエッジのたびにインクリメントされ、フレーム･クロックの立ち上がりエッジのたびに、またはGFXTIM_TCRのFALCRをセットすることによってリセットされます。

絶対ライン･カウンタがオーバーフローした場合は、GFXTIM_ISRのALCOFがセットされます。GFXTIM_IERのAFCOIEがセットされている場合は、割込みが生成されます。

カウンタ値が比較値と一致する場合、2つの12bitの比較レジスタで、絶対ライン･カウンタの比較イベントを生成できます。比較イベントが発生すると、GFXTIM_ISRのALCC1FまたはALCC2Fがセットされます。GFXTIM_IERのALCC1IEまたはALCC2IEがセットされている場合は、割込みが生成されます。

絶対ライン･カウンタは、GFXTIM_TDRのALCDISに1を書き込むことによって無効にできます。

ステータス（有効/無効）は、GFXTIM_TSRのAFCSを通じて監視できます。

59.3.6 相対タイマ

相対タイマは、グラフィカル･タスクをフレーム･レベルで同期するために、周期的なイベントを生成できます。

相対フレーム･カウンタは12bitの自動再ロードダウンカウントタイマで、フレーム･クロックの立ち上がりエッジのたびにデクリメントされます。カウンタはGFXTIM_TCRのRFCxEN（相対フレーム･カウンタxイネーブル）に1を書き込むことによって開始されます。開始値は、GFXTIM_RFCxRR（相対フレーム･カウンタx再ロードレジスタ）から自動的にロードされます。

現在の相対フレーム･カウンタの値は、直接GFXTIM_RFCxRで読み出すことができます。

カウンタが0になると、GFXTIM_RFCxRRの値が自動的に再ロードされます。GFXTIM_ISRのRFCxRF（再ロード･フラグ）がセットされ、GFXTIM_IERのRFCxRIE（割込みイネーブル）がセットされている場合は、割込みが生成されます。

再ロード操作が発生すると、タイマは自動的に無効にすることができますが、GFXTIM_TCRのRFCxCM（連続モード）がセットされている場合は、カウントを続けることができます。

ステータス（有効/無効）は、GFXTIM_TSRのRFCxSを通じて監視できます。

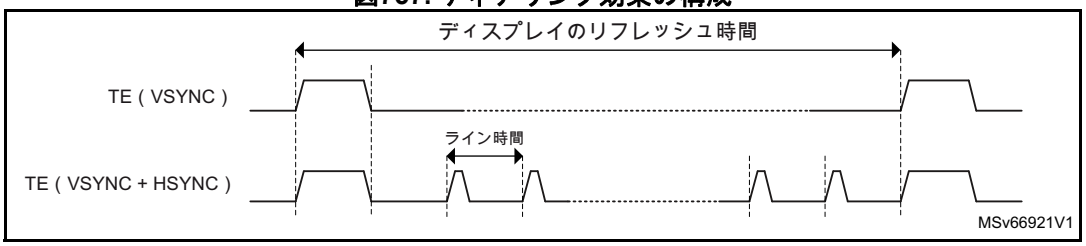
カウンタは、GFXTIM_TCRのFRFCRxをセットすることによって、オンザフライで再ロードできます。この強制再ロードにより、タイマが停止したり（GFXTIM_TCRのRFCxCM = 0であった場合も）、GFXTIM_ISRのRFCxRF（再ロード･フラグ）がセットされたりすることはありません。

相対フレーム･カウンタは、GFXTIM_TDRのRFCxDIS（ディセーブル）をセットすることによって無効にできます。

59.3.7 ティアリング効果の検出

ティアリング効果ラインは、下の図に示す2つの構成のうちの1つで動作します。

図757. ティアリング効果の構成



ティアリング効果イベントは、GFXTIM_CRのTEPOL（ティアリング効果極性）に応じて、立ち上がりエッジまたは立ち下がりエッジで生成することができます。

ティアリング効果イベントが検出されると、GFXTIM_ISRのTEF（ティアリング効果フラグ）がセットされます。GFXTIM_IERのTEIE（割込みイネーブル）がセットされている場合は、割込みが生成されます。

59.3.8 イベント･ジェネレータ

イベント･ジェネレータは、復号イベントにタイマイベントを組み込むことができます。最大4つの組み合わされたイベントを生成できます。

イベントの用途は以下のとおりです。

- 割込み生成
- ウォッチドッグへのクロック供給
- 外部トリガ生成

復号イベントは、フレーム･イベントとライン･イベントを組み合わせたものです。

フレーム･イベントが発生すると、GFXTIMは復号イベントを生成する前に、ライン･イベントの発生を待ちます。

フレーム･イベントは、GFXTIM_EVSR（イベント選択yレジスタ）の対応するFESx（フレーム･イベント選択x）によって選択されます。ライン･イベントは、GFXTIM_EVSRの対応するLESxによって選択されます。

復号イベントの生成は、GFXTIM_EVCR（イベント制御レジスタ）の対応するEVxEN（イベントxイネーブル）をセットすることによって有効になります。

復号イベントの誤生成を避けるため、イベント設定前にイベント生成を無効にすることをお勧めします。

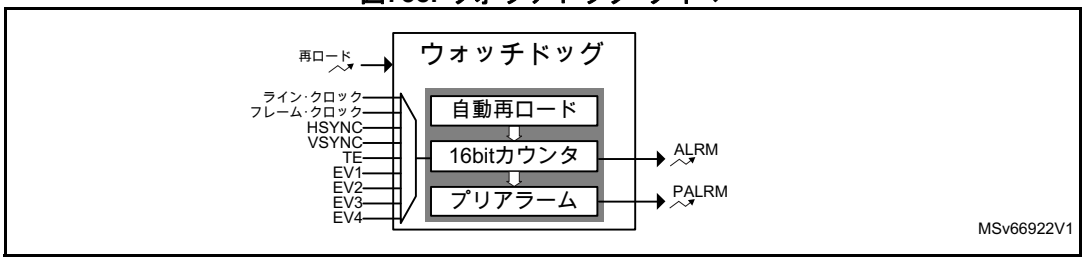
復号イベントが発生すると、GFXTIM_ISRの対応するEVxF（イベントxフラグ）がセットされます。GFXTIM_IERのEVxIE（イベントx割込みイネーブル）がセットされている場合は、割込みが生成されます。

ハードウェアトリガを生成するために、各イベントを別のペリフェラル（DMAなど）に接続できます。

59.3.9 ウォッチドッグ･タイマ

ウォッチドッグ･タイマは、プログラム可能なクロックソースを持つ16bitの自動再ロード･ダウンカウンタです。

図758. ウォッチドッグ･タイマ



クロックソース

ウォッチドッグ･クロック･ソースは、以下のいずれかの間のGTXTIM_WDGTCR（ウォッチドッグ･タイマ設定レジスタ）のWDGCS（クロックソース）を通じて選択できます。

- ライン･クロック
- フレーム･クロック
- HSYNC
- VSYNC
- TE
- イベント1
- イベント2
- イベント3
- イベント4

スタートアップ

ウォッチドッグは、GFXTIM_WDGTCRのWDGENをセットすることによって開始され、0に達すると自動的に停止します。

開始時に、ウォッチドッグカウンタはGFXTIM_WDGRR（ウォッチドッグの再ロードレジスタ）でプログラムされた自動再ロード値によって自動的にロードされます。

ウォッチドッグの現在値は、GFXTIM_WDGCRを通じて読み出すことができます。

自動再ロード

自動再ロードは、以下のいずれかの方法で強制的に実行できます。

- ソフトウェアによる場合は、GTXTIM_WDGTCR（ウォッチドッグ･タイマ設定レジスタ）のFWDGR（ウォッチドッグの強制再ロード）をセットする
- ハードウェアによる場合は、外部トリガ（gfxtim_wrld信号）を通じて行う

  トリガの極性はGFXTIM_WDGTCRのWDGHRC（ウォッチドッグ･ハードウェア再ロード設定）で設定されます。

イベントと割込み生成

ウォッチドッグは2つのイベントを生成できます。

- ウォッチドッグ･ダウンカウンタが0になったときのアラーム

  ウォッチドッグカウンタは自動的に停止し、GFXTIM_ISRのWDGAF（アラーム･フラグ）がセットされます。GFXTIM_IERのWDGAIE（アラーム割込みイネーブル）がセットされている場合は、割込みが生成されます。
- ウォッチドッグカウンタがGFXTIM_WDGPAR（ウォッチドッグ･プリアラーム･レジスタ）のプリアラーム･レジスタ値と一致するときのプリアラーム

  GFXTIM_ISRのWDGPF（プリアラーム･フラグ）がセットされます。GFXTIM_IERのWDGPIE（プリアラーム割込みイネーブル）がセットされている場合は、割込みが生成されます。

59.4 GFXTIM割込み

割込みは次のイベントによって生成できます。

- 絶対フレーム･カウンタ比較イベント
- 絶対フレーム･カウンタ･オーバーフロー
- 絶対ライン･カウンタ比較イベント
- 絶対ライン･カウンタ･オーバーフロー
- 相対フレーム･カウンタ再ロード･イベント
- 外部ティアリング効果
- 組み合わされたイベント
- ウォッチドッグ･アラーム･イベント
- ウォッチドッグ･プリアラーム･イベント

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表614.グラフィック･タイマ割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法
GFXTIM	絶対フレーム･カウンタ･オーバーフロー	AFCOF	AFCOIE	CAFCOFに1を書き込む
	絶対フレーム･カウンタ比較1	AFCC1F	AFCC1IE	CAFCC1Fに1を書き込む
	絶対ライン･カウンタ･オーバーフロー	ALCOF	ALCOIE	CALCOFに1を書き込む
	絶対ライン･カウンタ比較1	ALCC1F	ALCC1IE	CALCC1Fに1を書き込む
	絶対ライン･カウンタ比較2	ALCC2F	ALCC2IE	CALCC2Fに1を書き込む
	相対フレーム･カウンタ1再ロード	RFC1RF	RFC1RIE	CRFC1RFに1を書き込む
	相対フレーム･カウンタ2再ロード	RFC2RF	RFC2RIE	CRFC2RFに1を書き込む
	外部ティアリング効果	TEF	TEIE	CTEFに1を書き込む
	イベント1	EV1F	EV1IE	CEV1Fに1を書き込む
	イベント2	EV2F	EV2IE	CEV2Fに1を書き込む
	イベント3	EV3F	EV3IE	CEV3Fに1を書き込む
	イベント4	EV4F	EV4IE	CEV4Fに1を書き込む
GFXTIMW	ウォッチドッグ･アラーム	WDGAF	WDGAIE	CWDGAFに1を書き込む
	ウォッチドッグ･プリアラーム	WDGPF	WDGPIE	CWDGPFに1を書き込む

59.5 GFXTIMレジスタ

59.5.1 GFXTIM設定レジスタ（GFXTIM_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCCOE	FCCOE
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	SYNCS[1:0]		Res.	Res.	Res.	TEPOL	Res.	Res.	TES[1:0]	
						rw	rw				rw			rw	rw

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **LCCOE**：ライン･クロック較正出力イネーブル
このビットは、ライン･クロック出力を有効化します。
0：ライン･クロック出力は無効です。
1：ライン･クロック出力は有効です。

ビット16 **FCCOE**：フレーム･クロック較正出力イネーブル
このビットは、フレーム･クロック出力を有効化します。
0：フレーム･クロック出力は無効です。
1：フレーム･クロック出力は有効です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **SYNCS[1:0]**：同期ソース
このビットフィールドは、同期信号（HSYNCおよびVSYNC）のソースを選択します。
00：gfxtim_hsync[0] およびgfxtim_vsync[0] が選択されます。
01：gfxtim_hsync[1] およびgfxtim_vsync[1] が選択されます。
10：gfxtim_hsync[2] およびgfxtim_vsync[2] が選択されます。
11：gfxtim_hsync[3] およびgfxtim_vsync[3] が選択されます。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **TEPOL**：ティアリング効果極性
このビットは、ティアリング効果の極性を選択します。
0：立ち上がりエッジでアクティブなティアリング効果
1：立ち下がりエッジでアクティブなティアリング効果

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **TES[1:0]**：ティアリング･ソース
このビットフィールドは、ティアリング効果のソースを選択します。
00：TE入力パッドを選択します。
01：gfxtim_iteを選択します。
10：SYNCS[1:0] で選択されたHSYNC入力
11：SYNCS[1:0] で選択されたVSYNC入力

59.5.2 GFXTIMクロックジェネレータ設定レジスタ（GFXTIM_CGCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FCCHRS[2:0]			Res.	Res.	Res.	FCCFR	Res.	FCCCS[2:0]			Res.	FCS[2:0]		
	rw	rw	rw				w		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LCCHRS[2:0]			Res.	Res.	Res.	LCCFR	Res.	Res.	Res.	LCCCS	Res.	LCS[2:0]		
	rw	rw	rw				w				rw		rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:28 **FCCHRS[2:0]**：フレーム･クロック･カウンタのハードウェア再ロードのソース
このビットフィールドは、フレーム･クロック･カウンタのハードウェア再ロードのソースを設定します。
000：ハードウェアの再ロードなし
001：ライン･クロック･カウンタ･アンダーフロー
010：HSYNC立ち上がりエッジ
011：HSYNC立ち下がりエッジ
100：VSYNC立ち上がりエッジ
101：VSYNC立ち下がりエッジ
110：TE立ち上がりエッジ
111：TE立ち下がりエッジ

ビット27:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **FCCFR**：フレーム･クロック･カウンタの強制再ロード
このビットは、フレーム･クロック･カウンタを強制的に再ロードします。
0：影響なし
1：フレーム･クロック･カウンタの強制再ロード

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:20 **FCCCS[2:0]**：フレーム･クロック･カウンタのクロックソース
このビットフィールドは、フレーム･クロック･カウンタのクロックソースを設定します。
000：フレーム･クロック･カウンタは無効です。
001：ライン･クロック･カウンタ･アンダーフロー
010：HSYNC立ち上がりエッジ
011：HSYNC立ち下がりエッジ
100：VSYNC立ち上がりエッジ
101：VSYNC立ち下がりエッジ
110：TE立ち上がりエッジ
111：TE立ち下がりエッジ

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **FCS[2:0]**：フレーム･クロック･ソース
このビットフィールドは、フレーム･クロック･ソースを設定します。
000：ライン･クロック･カウンタ･アンダーフロー
001：フレーム･クロック･カウンタ･アンダーフロー
010：HSYNC立ち上がりエッジ
011：HSYNC立ち下がりエッジ
100：VSYNC立ち上がりエッジ
101：VSYNC立ち下がりエッジ
110：TE立ち上がりエッジ
111：TE立ち下がりエッジ

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **LCCHRS[2:0]**：ライン･クロック･カウンタのハードウェア再ロードのソース
このビットフィールドは、ライン･クロック･カウンタのハードウェア再ロードのソースを設定します。
000：ハードウェアの再ロードなし
001：フレーム･クロック･カウンタ･アンダーフロー
010：HSYNC立ち上がりエッジ
011：HSYNC立ち下がりエッジ
100：VSYNC立ち上がりエッジ
101：VSYNC立ち下がりエッジ
110：TE立ち上がりエッジ
111：TE立ち下がりエッジ

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **LCCFR**：ライン･クロック･カウンタの強制再ロード
このビットは、ライン･クロック･カウンタを強制的に再ロードします。
0：影響なし
1：ライン･クロック･カウンタの強制再ロード

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **LCCCS**：ライン･クロック･カウンタのクロックソース
このビットは、ライン･クロック･カウンタのクロックソースを設定します。
0：ライン･クロック･カウンタは無効です。
1：システムクロックを選択します。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **LCS[2:0]**：ライン･クロック･ソース
このビットフィールドは、ライン･クロック･ソースを設定します。
000：ライン･クロック･カウンタ･アンダーフロー
001：フレーム･クロック･カウンタ･アンダーフロー
010：HSYNC立ち上がりエッジ
011：HSYNC立ち下がりエッジ
100：VSYNC立ち上がりエッジ
101：VSYNC立ち下がりエッジ
110：TE立ち上がりエッジ
111：TE立ち下がりエッジ

59.5.3 GFXTIMタイマ設定レジスタ（GFXTIM_TCR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRFC2R	RFC2CM	RFC2EN	Res.	FRFC1R	RFC1CM	RFC1EN
									w	rw	w		w	rw	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FALCR	ALCEN	Res.	Res.	FAFCR	AFCEN
										w	w			w	w

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **FRFC2R**：相対フレーム･カウンタ2の強制再ロード
このビットは、相対フレーム･カウンタ2を強制的に再ロードします。
0：影響なし
1：相対フレーム･カウンタ2の強制再ロード

ビット21 **RFC2CM**：相対フレーム･カウンタ2の連続モード
このビットは、相対フレーム･カウンタ2の連続モードを有効化します。
0：相対フレーム･カウンタ2はワンショットです。
1：相対フレーム･カウンタ2は連続モードです。

ビット20 **RFC2EN**：相対フレーム･カウンタ2イネーブル
このビットは、相対フレーム･カウンタ2を有効化します。
0：影響なし
1：相対フレーム･カウンタ2は有効です。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **FRFC1R**：相対フレーム･カウンタ1の強制再ロード
このビットは、相対フレーム･カウンタ1を強制的に再ロードします。
0：影響なし
1：相対フレーム･カウンタ1の強制再ロード

ビット17 **RFC1CM**：相対フレーム･カウンタ1の連続モード
このビットは、相対フレーム･カウンタ1の連続モードを有効化します。
0：相対フレーム･カウンタ1はワンショットです。
1：相対フレーム･カウンタ1は連続モードです。

ビット16 **RFC1EN**：相対フレーム･カウンタ1イネーブル
このビットは、相対フレーム･カウンタ1を有効化します。
0：影響なし
1：相対フレーム･カウンタは有効です。

ビット15:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **FALCR**：絶対ライン･カウンタの強制リセット
このビットは、絶対ライン･カウンタを強制的にリセットします。
0：影響なし
1：絶対ライン･カウンタの強制リセット

ビット4 **ALCEN**：絶対ライン･カウンタ･イネーブル
このビットは、絶対ライン･カウンタを有効化します。
0：影響なし
1：絶対ライン･カウンタは有効です。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **FAFCR**：絶対フレーム･カウンタの強制
このビットは、絶対フレーム･カウンタを強制的にリセットします。
0：影響なし
1：絶対フレーム･カウンタのリセットを強制します。

ビット0 **AFCEN**：絶対フレーム･カウンタ･イネーブル
このビットは、絶対フレーム･カウンタを有効化します。
0：影響なし
1：絶対フレーム･カウンタは有効です。

59.5.4 GFXTIMタイマ無効化レジスタ（GFXTIM_TCR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFC2DIS	Res.	Res.	Res.	RFC1DIS
											w				w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALCDIS	Res.	Res.	Res.	AFCDIS
											w				w

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **RFC2DIS**：相対フレーム･カウンタ2ディスエーブル
このビットは、相対フレーム･カウンタ2を無効化します。
0：影響なし
1：相対フレーム･カウンタ2は無効です。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RFC1DIS**：相対フレーム･カウンタ1ディスエーブル
このビットは、相対フレーム･カウンタ1を無効化します。
0：影響なし
1：相対フレーム･カウンタ1は無効です。

ビット15:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ALCDIS**：絶対ライン･カウンタ･ディスエーブル
このビットは、絶対ライン･カウンタを無効化します。
0：影響なし
1：絶対ライン･カウンタは無効です。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **AFCDIS**：絶対フレーム･カウンタ･ディスエーブル
このビットは、絶対フレーム･カウンタを無効化します。
0：影響なし
1：絶対フレーム･カウンタは無効です。

59.5.5 GFXTIMイベント制御レジスタ（GFXTIM_EVCR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EV4EN	EV3EN	EV2EN	EV1EN
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **EV4EN**：イベント4イネーブル
このビットは、復号イベント4の生成を有効化します。
0：イベント4生成は無効です。
1：イベント4生成は有効です。

ビット2 **EV3EN**：イベント3イネーブル
このビットは、復号イベント3の生成を有効化します。
0：イベント3生成は無効です。
1：イベント3生成は有効です。

ビット1 **EV2EN**：イベント2イネーブル
このビットは、復号イベント2の生成を有効化します。
0：イベント2生成は無効です。
1：イベント2生成は有効です。

ビット0 **EV1EN**：イベント1イネーブル
このビットは、復号イベント1の生成を有効化します。
0：イベント1生成は無効です。
1：イベント1生成は有効です。

59.5.6 GFXTIMイベント選択レジスタ（GFXTIM_EVCR）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FES4 [2:0]			Res.	LES4 [2:0]			Res.	FES3 [2:0]			Res.	LES3 [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FES2 [2:0]			Res.	LES2 [2:0]			Res.	FES1 [2:0]			Res.	LES1 [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:28 **FES4[2:0]**：フレーム･イベント選択4

このビットフィールドは、復号イベント4の生成に関するフレーム･イベント選択について定義します。

000：フレーム･イベントなし

001：絶対フレーム･カウンタ･オーバーフロー

010：絶対フレーム･カウンタ比較

100：相対フレーム･カウンタ1再ロード

101：相対フレーム･カウンタ2再ロード

その他：予約済み

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **LES4[2:0]**：ライン･イベント選択4

このビットフィールドは、復号イベント4の生成に関するライン･イベント選択について定義します。

000：ライン･イベントなし

001：絶対ライン･カウンタ･オーバーフロー

010：ティアリング効果

100：絶対ライン･カウンタ1比較

101：絶対ライン･カウンタ2比較

その他：予約済み

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:20 **FES3[2:0]**：フレーム･イベント選択3

このビットフィールドは、復号イベント3の生成に関するフレーム･イベント選択について定義します。

000：フレーム･イベントなし

001：絶対フレーム･カウンタ･オーバーフロー

010：絶対フレーム･カウンタ比較

100：相対フレーム･カウンタ1再ロード

101：相対フレーム･カウンタ2再ロード

その他：予約済み

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **LES3[2:0]**：ライン･イベント選択3

このビットフィールドは、復号イベント3の生成に関するライン･イベント選択について定義します。

000：ライン･イベントなし

001：絶対ライン･カウンタ･オーバーフロー

010：ティアリング効果

100：絶対ライン･カウンタ1比較

101：絶対ライン･カウンタ2比較

その他：予約済み

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **FES2[2:0]**：フレーム･イベント選択2

このビットフィールドは、復号イベント2の生成に関するフレーム･イベント選択について定義します。

000：フレーム･イベントなし

001：絶対フレーム･カウンタ･オーバーフロー

010：絶対フレーム･カウンタ比較

100：相対フレーム･カウンタ1再ロード

101：相対フレーム･カウンタ2再ロード

その他：予約済み

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **LES2[2:0]**：ライン･イベント選択2

このビットフィールドは、復号イベント2の生成に関するライン･イベント選択について定義します。

000：ライン･イベントなし

001：絶対ライン･カウンタ･オーバーフロー

010：ティアリング効果

100：絶対ライン･カウンタ1比較

101：絶対ライン･カウンタ2比較

その他：予約済み

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **FES1[2:0]**：フレーム･イベント選択1

このビットフィールドは、復号イベント1の生成に関するフレーム･イベント選択について定義します。

000：フレーム･イベントなし

001：絶対フレーム･カウンタ･オーバーフロー

010：絶対フレーム･カウンタ比較

100：相対フレーム･カウンタ1再ロード

101：相対フレーム･カウンタ2再ロード

その他：予約済み

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **LES1[2:0]**：ライン･イベント選択1

このビットフィールドは、復号イベント1の生成に関するライン･イベント選択について定義します。

000：ライン･イベントなし

001：絶対ライン･カウンタ･オーバーフロー

010：ティアリング効果

100：絶対ライン･カウンタ1比較

101：絶対ライン･カウンタ2比較

その他：予約済み

59.5.7 GFXTIMウォッチドッグ･タイマ設定レジスタ（GFXTIM_WDGTCR）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FWDGR
															w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	WDGCS[3:0]				Res.	Res.	WDGHRC[1:0]		Res.	WDGS	WDGDIS	WDGEN
				rw	rw	rw	rw			rw	rw		r	w	w

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **FWDGR**：ウォッチドッグの強制再ロード
このビットは、グラフィック･ウォッチドッグを強制的に再ロードします。
0：影響なし
1：グラフィック･ウォッチドッグの強制再ロード

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **WDGCS[3:0]**：ウォッチドッグ･クロック･ソース
このビットフィールドは、ウォッチドッグ･クロック･ソースを選択します。
0000：ライン･クロック
0001：フレーム･クロック
0010：HSYNC立ち上がりエッジ
0011：HSYNC立ち下がりエッジ
0100：VSYNC立ち上がりエッジ
0101：VSYNC立ち下がりエッジ
0110：TE立ち上がりエッジ
0111：TE立ち下がりエッジ
1000：イベント1
1001：イベント2
1010：イベント3
1011：イベント4
その他：予約済み

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **WDGHRC[1:0]**：ウォッチドッグ･ハードウェア再ロード設定
このビットフィールドは、ウォッチドッグ･ハードウェアの再ロードを設定します。
00：ウォッチドッグ･ハードウェアの再ロードは無効です。
01：ウォッチドッグはgfxtim_wrldの立ち上がりエッジを再ロードします。
10：ウォッチドッグはgfxtim_wrldの立ち下がりエッジを再ロードします。
11：予約済み

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **WDGS**：ウォッチドッグ･ステータス
このビットは、グラフィック･ウォッチドッグのステータスを返します。
0：グラフィック･ウォッチドッグは無効です。
1：グラフィック･ウォッチドッグは有効です。

ビット1 **WDGDIS**：ウォッチドッグ･ディスエーブル

このビットは、グラフィック･ウォッチドッグを無効化します。

0：影響なし

1：グラフィック･ウォッチドッグは無効です。

ビット0 **WDGEN**：ウォッチドッグ･イネーブル

このビットは、グラフィック･ウォッチドッグを有効化します。

0：影響なし

1：グラフィック･ウォッチドッグは有効です。

59.5.8 GFXTIM割込みステータスレジスタ（GFXTIM_ISR）

アドレスオフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	WDGPF	WDGAF	Res.	Res.	Res.	Res.	EV4F	EV3F	EV2F	EV1F
						r	r					r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	RFC2RF	RFC1RF	Res.	Res.	ALCC2F	ALCC1F	Res.	Res.	Res.	AFCC1F	Res.	TEF	ALCOF	AFCOF
		r	r			r	r				r		r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **WDGPF**：ウォッチドッグ･プリアラーム･フラグ

このビットは、グラフィック･ウォッチドッグ･プリアラームが発生したことを示します。

0：グラフィック･ウォッチドッグ･プリアラームは発生していません。

1：グラフィック･ウォッチドッグ･プリアラームが発生しました。

ビット24 **WDGAF**：ウォッチドッグ･アラーム･フラグ

このビットは、グラフィック･ウォッチドッグ･アラームが発生したことを示します。

0：グラフィック･ウォッチドッグ･アラームは発生していません。

1：グラフィック･ウォッチドッグ･アラームが発生しました。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **EV4F**：イベント4のフラグ

このビットは、復号イベント4が発生したことを示します。

0：復号イベント4は発生していません。

1：復号イベント4が発生しました。

ビット18 **EV3F**：イベント3のフラグ

このビットは、復号イベント3が発生したことを示します。

0：復号イベント3は発生していません。

1：復号イベント3が発生しました。

ビット17 **EV2F**：イベント2のフラグ

このビットは、復号イベント2が発生したことを示します。

0：復号イベント2は発生していません。

1：復号イベント2が発生しました。

ビット16 **EV1F**：イベント1のフラグ
このビットは、復号イベント1が発生したことを示します。
0：復号イベント1は発生していません。
1：復号イベント1が発生しました。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **RFC2RF**：相対フレーム･カウンタ2再ロード･フラグ
このビットは、相対フレーム･カウンタ2が再ロードされたことを示します。
0：相対フレーム･カウンタ2の再ロードは発生していません。
1：相対フレーム･カウンタ2の再ロードが発生しました。

ビット12 **RFC1RF**：相対フレーム･カウンタ1再ロード･フラグ
このビットは、相対フレーム･カウンタ1が再ロードされたことを示します。
0：相対フレーム･カウンタ1の再ロードは発生していません。
1：相対フレーム･カウンタ1の再ロードが発生しました。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **ALCC2F**：絶対ライン･カウンタ比較2フラグ
このビットは、絶対ライン･カウンタ比較2での一致について示します。
0：絶対ライン･カウンタ比較2で一致は発生していません。
1：絶対ライン･カウンタ比較2で一致が発生しました。

ビット8 **ALCC1F**：絶対ライン･カウンタ比較1フラグ
このビットは、絶対ライン･カウンタ比較1での一致について示します。
0：絶対ライン･カウンタ比較1で一致は発生していません。
1：絶対ライン･カウンタ比較1で一致が発生しました。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **AFCC1F**：絶対フレーム･カウンタ比較1フラグ
このビットは、絶対フレーム･カウンタ比較1での一致について示します。
0：絶対フレーム･カウンタ比較1で一致は発生していません。
1：絶対フレーム･カウンタ比較1で一致が発生しました。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **TEF**：ティアリング効果フラグ
このビットは、ティアリング効果イベントが発生したことを示します。
0：ティアリング効果は発生していません。
1：ティアリング効果が発生しました。

ビット1 **ALCOF**：絶対ライン･カウンタ･オーバーフロー･フラグ
このビットは、絶対ライン･カウンタで発生したオーバーフローについて示します。
0：絶対ライン･カウンタでオーバーフローは発生していません。
1：絶対ライン･カウンタでオーバーフローが発生しました。

ビット0 **AFCOF**：絶対フレーム･カウンタ･オーバーフロー･フラグ
このビットは、絶対フレーム･カウンタで発生したオーバーフローについて示します。
0：絶対フレーム･カウンタでオーバーフローは発生していません。
1：絶対フレーム･カウンタでオーバーフローが発生しました。

59.5.9 GFXTIM割込みクリアレジスタ（GFXTIM_ICR）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	CWDGPF	CWDGAF	Res.	Res.	Res.	Res.	CEV4F	CEV3F	CEV2F	CEV1F
						w	w					w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CRFC2RF	CRFC1RF	Res.	Res.	CALCC2F	CALCC1F	Res.	Res.	Res.	CAFCC1F	Res.	CTEF	CALCOF	CAFCOF
		w	w			w	w				w		w	w	w

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **CWDGPF**：ウォッチドッグ･プリアラーム･フラグのクリア
このビットは、GXFXTIM_ISRのWDGPFをクリアします。
0：影響なし
1：WDGPFのクリア

ビット24 **CWDGAF**：ウォッチドッグ･アラーム･フラグのクリア
このビットは、GXFXTIM_ISRのWDGAFをクリアします。
0：影響なし
1：WDGAFのクリア

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **CEV4F**：イベント4フラグのクリア
このビットは、GXFXTIM_ISRのEV4Fをクリアします。
0：影響なし
1：EV4Fのクリア

ビット18 **CEV3F**：イベント3フラグのクリア
このビットは、GXFXTIM_ISRのEV3Fをクリアします。
0：影響なし
1：EV3Fのクリア

ビット17 **CEV2F**：イベント2フラグのクリア
このビットは、GXFXTIM_ISRのEV2Fをクリアします。
0：影響なし
1：EV2Fのクリア

ビット16 **CEV1F**：イベント1フラグのクリア
このビットは、GXFXTIM_ISRのEV1F をクリアします。
0：影響なし
1：EV1Fのクリア

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **CRFC2RF**：相対フレーム･カウンタ2再ロード･フラグのクリア
このビットは、GXFXTIM_ISRのRFC2RFをクリアします。
0：影響なし
1：RFC2RFのクリア

ビット12 **CRFC1RF**：相対フレーム･カウンタ1再ロード･フラグのクリア
このビットは、GXTIM_ISRのRFC1RFをクリアします。
0：影響なし
1：RFC1RFのクリア

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CALCC2F**：絶対ライン･カウンタ比較2フラグのクリア
このビットは、GXTIM_ISRのALCC2Fをクリアします。
0：影響なし
1：ALCC2Fのクリア

ビット8 **CALCC1F**：絶対ライン･カウンタ比較1フラグのクリア
このビットは、GXTIM_ISRのALCC1Fをクリアします。
0：影響なし
1：ALCC1Fのクリア

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **CAFCC1F**：絶対フレーム･カウンタ比較1フラグのクリア
このビットは、GXTIM_ISRのAFCC1Fをクリアします。
0：影響なし
1：AFCC1Fのクリア

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CTEF**：ティアリング効果フラグのクリア
このビットは、GXTIM_ISRのTEFをクリアします。
0：影響なし
1：TEFのクリア

ビット1 **CALCOF**：絶対ライン･カウンタ･オーバーフロー･フラグのクリア
このビットは、GXTIM_ISRのALCOFをクリアします。
0：影響なし
1：ALCOFのクリア

ビット0 **CAFCOF**：絶対フレーム･カウンタ･オーバーフロー･フラグのクリア
このビットは、GXTIM_ISRのAFCOFをクリアします。
0：影響なし
1：AFCOFのクリア

59.5.10 GFXTIM割込み有効レジスタ（GFXTIM_IER）

アドレスオフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	WDGPIE	WDGAIE	Res.	Res.	Res.	Res.	EV4IE	EV3IE	EV2IE	EV1IE
						rw	rw					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	RFC2RIE	RFC1RIE	Res.	Res.	ALCC2IE	ALCC1IE	Res.	Res.	Res.	AFCC1IE	Res.	TEIE	ALCOIE	AFCOIE
		rw	rw			rw	rw				rw		rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **WDGPIE**：ウォッチドッグ･プリアラーム割込みイネーブル
このビットは、ウォッチドッグ･プリアラーム割込み生成を有効化します。
0：ウォッチドッグ･プリアラーム割込みは無効です。
1：ウォッチドッグ･プリアラーム割込みは有効です。

ビット24 **WDGAIE**：ウォッチドッグ･アラーム割込みイネーブル
このビットは、ウォッチドッグ･アラーム割込み生成を有効化します。
0：ウォッチドッグ･アラーム割込みは無効です。
1：ウォッチドッグ･アラーム割込みは有効です。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **EV4IE**：イベント4割込みイネーブル
このビットは、復号イベント4の割込み生成を有効化します。
0：イベント4割込みは無効です。
1：イベント4割込みは有効です。

ビット18 **EV3IE**：イベント3割込みイネーブル
このビットは、復号イベント3の割込み生成を有効化します。
0：イベント3割込みは無効です。
1：イベント3割込みは有効です。

ビット17 **EV2IE**：イベント2割込みイネーブル
このビットは、復号イベント2の割込み生成を有効化します。
0：イベント2割込みは無効です。
1：イベント2割込みは有効です。

ビット16 **EV1IE**：イベント1割込みイネーブル
このビットは、復号イベント1の割込み生成を有効化します。
0：イベント1割込みは無効です。
1：イベント1割込みは有効です。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **RFC2RIE**：相対フレーム･カウンタ2再ロード割込みイネーブル
このビットは、相対フレーム･カウンタ2の再ロード割込み生成を有効化します。
0：相対フレーム･カウンタ2再ロード割込みは無効です。
1：相対フレーム･カウンタ2再ロード割込みは有効です。

ビット12 **RFC1RIE**：相対フレーム･カウンタ1再ロード割込みイネーブル
このビットは、相対フレーム･カウンタ1の再ロード割込み生成を有効化します。
0：相対フレーム･カウンタ1再ロード割込みは無効です。
1：相対フレーム･カウンタ1再ロード割込みは有効です。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **ALCC2IE**：絶対ライン･カウンタ比較2割込みイネーブル
このビットは、絶対ライン･カウンタ比較2の割込み生成を有効化します。
0：絶対ライン･カウンタ比較2割込みは無効です。
1：絶対ライン･カウンタ比較2割込みは有効です。

ビット8 **ALCC1IE**：絶対ライン･カウンタ比較1割込みイネーブル
このビットは、絶対ライン･カウンタ比較1の割込み生成を有効化します。
0：絶対ライン･カウンタ比較1割込みは無効です。
1：絶対ライン･カウンタ比較1割込みは有効です。

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **AFCC1IE**：絶対フレーム･カウンタ比較1割込みイネーブル
このビットは、絶対フレーム･カウンタ比較の割込み生成を有効化します。
0：絶対フレーム･カウンタ比較1割込みは無効です。
1：絶対フレーム･カウンタ比較1割込みは有効です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **TEIE**：ティアリング効果割込みイネーブル
このビットは、ティアリング効果割込み生成を有効化します。
0：ティアリング効果割込みは無効です。
1：ティアリング効果割込みは有効です。

ビット1 **ALCOIE**：絶対ライン･カウンタ･オーバーフロー割込みイネーブル
このビットは、絶対ライン･カウンタ･オーバーフローの割込み生成を有効化します。
0：絶対ライン･カウンタ･オーバーフロー割込みは無効です。
1：絶対ライン･カウンタ･オーバーフロー割込みは有効です。

ビット0 **AFCOIE**：絶対フレーム･カウンタ･オーバーフロー割込みイネーブル
このビットは、絶対フレーム･カウンタ･オーバーフローの割込み生成を有効化します。
0：絶対フレーム･カウンタ･オーバーフロー割込みは無効です。
1：絶対フレーム･カウンタ･オーバーフロー割込みは有効です。

59.5.11 GFXTIMタイマ･ステータス･レジスタ（GFXTIM_ISR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFC2S	Res.	Res.	Res.	RFC1S
											r				r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALCS	Res.	Res.	Res.	AFCS
											r				r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **RFC2S**：相対フレーム･カウンタ2のステータス
このビットは、相対フレーム･カウンタ2のステータスを返します。
0：相対フレーム･カウンタ2は無効です。
1：相対フレーム･カウンタ2は有効です。

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RFC1S**：相対フレーム･カウンタ1のステータス
このビットは、相対フレーム･カウンタ1のステータスを返します。
0：相対フレーム･カウンタ1は無効です。
1：相対フレーム･カウンタ1は有効です。

ビット15:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ALCS**：絶対ライン･カウンタのステータス
このビットは、絶対ライン･カウンタのステータスを返します。
0：絶対ライン･カウンタは無効です。
1：絶対ライン･カウンタは有効です。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **AFCS**：絶対フレーム･カウンタのステータス
このビットは、絶対フレーム･カウンタのステータスを返します。
0：絶対フレーム･カウンタは無効です。
1：絶対フレーム･カウンタは有効です。

59.5.12 GFXTIMライン･クロック･カウンタの再ロードレジスタ（GFXTIM_LCCRR）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RELOAD[21:16]					
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21:0 **RELOAD[21:0]**：再ロード値

ライン･クロック･カウンタの再ロード値。

59.5.13 GFXTIMフレーム･クロック･カウンタの再ロードレジスタ（GFXTIM_FCCRR）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RELOAD[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **RELOAD[11:0]**：再ロード値

フレーム･クロック･カウンタの再ロード値。

59.5.14 GFXTIM絶対時間レジスタ（GFXTIM_ATR）

アドレスオフセット：0x050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FRAME[19:4]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRAME[3:0]				LINE[11:0]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 **FRAME[19:0]**：フレーム番号
絶対フレーム･カウンタの現在値。

ビット11:0 **LINE[11:0]**：ライン番号
絶対ライン･カウンタの現在値。

59.5.15 GFXTIM絶対フレーム･カウンタ･レジスタ（GFXTIM_AFCR）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRAME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **FRAME[19:0]**：フレーム番号
絶対フレーム･カウンタの現在値。
注： このビットフィールドは、絶対フレーム･カウンタが無効の場合にのみ書込み可能です。

59.5.16 GFXTIM絶対ライン･カウンタ･レジスタ（GFXTIM_ALCR）

アドレスオフセット：0x058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LINE[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LINE[11:0]**：ライン番号

絶対ライン･カウンタの現在値。

注： このビットフィールドは、絶対ライン･カウンタが無効の場合にのみ書込み可能です。

59.5.17 GFXTIM絶対フレーム･カウンタ比較1レジスタ（GFXTIM_AFCC1R）

アドレスオフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRAME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **FRAME[19:0]**：フレーム番号

絶対フレーム･カウンタの比較1の値。

59.5.18 GFXTIM絶対ライン･カウンタ比較1レジスタ（GFXTIM_ALCC1R）

アドレスオフセット：0x070

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LINE[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LINE[11:0]**：ライン番号
絶対ライン･カウンタの比較値1。

59.5.19 GFXTIM絶対ライン･カウンタ比較2レジスタ（GFXTIM_ALCC2R）

アドレスオフセット：0x074

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LINE[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **LINE[11:0]**：ライン番号
絶対ライン･カウンタの比較値2。

59.5.20 GFXTIM相対フレーム･カウンタ1レジスタ（GFXTIM_RFC1R）

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FRAME[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **FRAME[11:0]**：フレーム番号
相対フレーム･カウンタ1の現在値。

59.5.21 GFXTIM相対フレーム･カウンタ1の再ロード･レジスタ（GFXTIM_RFC1RR）

アドレスオフセット：0x084

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FRAME[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **FRAME[11:0]**：フレーム再ロード値

相対フレーム･カウンタ1の再ロード値。

59.5.22 GFXTIM相対フレーム･カウンタ2レジスタ（GFXTIM_RFC2R）

アドレスオフセット：0x088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FRAME[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **FRAME[11:0]**：フレーム番号

相対フレーム･カウンタ2の現在値。

59.5.23 GFXTIM相対フレーム･カウンタ2の再ロード･レジスタ（GFXTIM_RFC2RR）

アドレスオフセット：0x08C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FRAME[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **FRAME[11:0]**：フレーム再ロード値

相対フレーム･カウンタ2の再ロード値。

59.5.24 GFXTIMウォッチドッグ･カウンタ･レジスタ（GFXTIM_WDGCR）

アドレスオフセット：0x0A0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VALUE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **VALUE[15:0]**：値

ウォッチドッグカウンタの現在値。

59.5.25 GFXTIMウォッチドッグ再ロード･レジスタ（GFXTIM_WDGRR）

アドレスオフセット：0x0A4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RELOAD[15:0]**：再ロード値
ウォッチドッグカウンタの再ロード値。

59.5.26 GFXTIMウォッチドッグ･プリアラーム･レジスタ（GFXTIM_WDGPAR）

アドレスオフセット：0x0A8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PREALARM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **PREALARM[15:0]**：プリアラーム値
ウォッチドッグカウンタのプリアラーム値。

59.5.27 GFXTIMレジスタマップ

表615.GFXTIMレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	GFXTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCCOE	FCCOE	Res.	Res.	Res.	Res.	Res.	Res.	SYNCS [1:0]		Res.	Res.	Res.	TEPOL	Res.	Res.	TES [1:0]	
	リセット値															0	0							0	0				0			0	0
0x004	GFXTIM_CGCR	Res.	FCCHRS [2:0]			Res.	Res.	Res.	FCCFR		FCCCS [2:0]			Res.	FCS [2:0]			Res.	LCCHRS [2:0]			Res.	Res.	Res.	LCCFR	Res.	Res.	Res.	LCCCS	Res.	LCS[2:0]		
	リセット値		0	0	0				0		0	0	0		0	0	0		0	0	0				0				0		0	0	0
0x008	GFXTIM_TCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRFC2R	RFC2CM	RFC2EN	Res.	FRFC1R	RFC1CM	RFC1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FALCR	ALCEN	Res.	Res.	FAFCR	AFCEN
	リセット値										0	0	0		0	0	0											0	0			0	0
0x00C	GFXTIM_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFC2DIS	Res.	Res.	Res.	RFC1DIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALCDIS	Res.	Res.	Res.	AFCDIS
	リセット値												0				0												0				0
0x010	GFXTIM_EVCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EV4EN	EV3EN	EV2EN	EV1EN
	リセット値																													0	0	0	0
0x014	GFXTIM_EVSR	Res.	FES4 [2:0]			Res.	LES4 [2:0]			Res.	FES3 [2:0]			Res.	LES3 [2:0]			Res.	FES2 [2:0]			Res.	LES2 [2:0]			Res.	FES1 [2:0]			Res.	Res.	LES1 [2:0]	
	リセット値		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0			0	0	0
0x018 ~0x01C	予約済み	予約済み																															
0x020	GFXTIM_ WDGTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FWDGR	Res.	Res.	Res.	Res.	WDGCS[3:0]				Res.	Res.	WDGHRC [1:0]		Res.	WDGS	WDGDIS	WDGEN
	リセット値																0					0	0	0	0			0	0		0	0	0
0x024 ~0x02C	予約済み	予約済み																															
0x030	GFXTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	WDGPF	WDGAF	Res.	Res.	Res.	Res.	EV4F	EV3F	EV2F	EV1F	Res.	Res.	RFC2RF	RFC1RF	Res.	Res.	ALCC2F	ALCC1F	Res.	Res.	Res.	AFCC1F	Res.	TEF	ALCOF	AFCOF
	リセット値							0	0					0	0	0	0			0	0			0	0				0		0	0	0
0x034	GFXTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	CWDGPF	CWDGAF	Res.	Res.	Res.	Res.	CEV4F	CEV3F	CEV2F	CEV1F	Res.	Res.	CRFC2RF	CRFC1RRF	Res.	Res.	CALCC2F	CALCC1F	Res.	Res.	Res.	CAFCC1F	Res.	CTEF	CALCOF	CAFCOF
	リセット値							0	0					0	0	0	0			0	0			0	0				0		0	0	0
0x038	GFXTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	WDGPIE	WDGAIE	Res.	Res.	Res.	Res.	EV4IE	EV3IE	EV2IE	EV1IE	Res.	Res.	RFC2RIE	RFC1RIE	Res.	Res.	ALCC2IE	ALCC1IE	Res.	Res.	Res.	AFCC1IE	Res.	TEIE	ALCOIE	AFCOIE
	リセット値							0	0					0	0	0	0			0	0			0	0				0		0	0	0
0x03C	GFXTIM_TSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFC2S	Res.	Res.	Res.	RFC1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALCS	Res.	Res.	Res.	AFCS
	リセット値												0				0												0				0
0x040	GFXTIM_LCCRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RELOAD[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	GFXTIM_FCCRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RELOAD[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x048 ~0x04C	予約済み	予約済み																															
0x050	GFXTIM_ATR	FRAME[19:0]																				LINE[11:0]											
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表615.GFXTIMレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x054	GFXTIM_AFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	GFXTIM_ALCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x05C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x060	GFXTIM_AFCC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x064 ~0x06C	予約済み	予約済み																															
0x070	GFXTIM_ALCC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x074	GFXTIM_ALCC2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x078 ~0x07C	予約済み	予約済み																															
0x080	GFXTIM_RFC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x084	GFXTIM_RFC1RR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x088	GFXTIM_RFC2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x08C	GFXTIM_RFC2RR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRAME[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x090 ~0x09C	予約済み	予約済み																															
0x0A0	GFXTIM_WDGCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALUE[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A4	GFXTIM_WDGRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RELOAD[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A8	GFXTIM_WDGPAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREALARM[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

60 赤外線インタフェース（IRTIM）

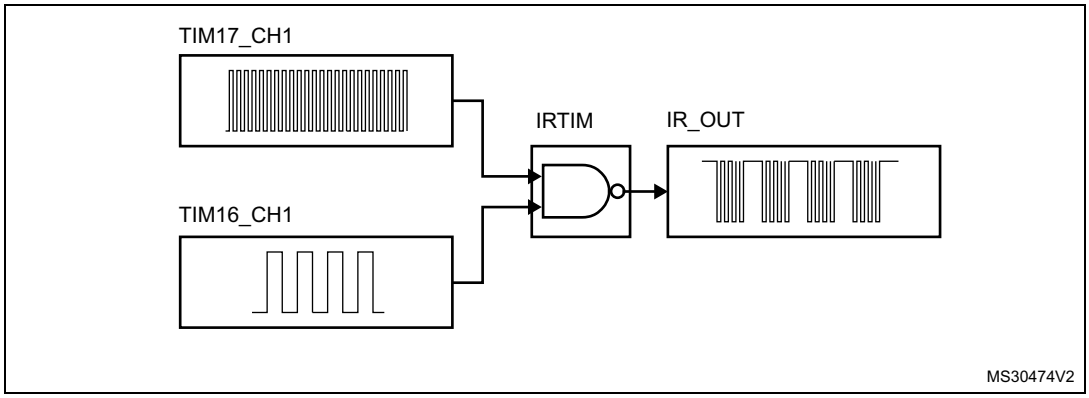
このデバイスではリモートコントロールで赤外線インタフェース（IRTIM）を使用できます。リモートコントロール機能を実行するために、赤外線LEDと併せて使用できます。

図759に示すように、TIM16 およびTIM17 との間で内部接続を使用します。

赤外線リモートコントロール信号を生成するには、正しい波形を生成するためにIRインタフェースを有効にし、TIM16チャネル1 （TIM16_OC1）とTIM17チャネル1 （TIM17_OC1）を適切に設定する必要があります。

赤外線レシーバは、基本的な入力キャプチャモードで簡単に実装できます。

図759. TIM16 およびTIM17 との IRTIM内部ハードウェア接続



標準のIRパルス変調モードはすべて2つのタイマ出力比較チャネルをプログラミングすることで取得できます。

TIM16 は変調包絡線を生成し、TIM17 は高周波キャリア信号を生成するために使用されます。

赤外線機能はIR_OUTピンから出力。この機能の有効化は、GPIOx_AFRxレジスタを通じ、関連するオルタネート機能ビットを有効にすることで行われます。

大電流シンクLEDドライバ機能（PB9 ピンでのみ使用可能）の有効化はSYSCFG_CFGR1レジスタのPB9_FMPビットを通じて行われ、赤外線LEDを直接制御するために必要な大電流を流すために使用されます。

61 独立型ウォッチドッグ（IWDG）

61.1 概要

独立型ウォッチドック（IWDG）ペリフェラルは、ソフトウェアまたはハードウェア障害による不具合を検出する機能により、高い安全レベルを実現しています。

IWDGは独立型クロックによりクロック供給されているため、メインクロックの障害時も機能に影響が出ません。

また、ウォッチドッグ機能はV_{DD}電圧ドメインで実行され、低電力モードでもIWDGが機能します。この製品のIWDGの機能を確認するには、セクション61.3を参照してください。

IWDGは、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、予期しない行動の検知が確実に行えます。

61.2 IWDGの主な機能

- 12bitダウンカウンタ
- デュアル電圧ドメイン、そのため低電力モードでの動作が可能
- 独立クロック
- 早期ウェイクアップ割込みの生成
- 生成のリセット
  - タイムアウトの場合
  - 想定したウィンドウの外でのリフレッシュの場合

61.3 IWDGの実装

表616.STM32U5シリーズIWDGの機能(1)

IWDGのモード／機能	IWDG
IWDGカーネルクロックとして使用されるLSI（iwdg_ker_ck）	X
ウィンドウ関数	X
早期ウェイクアップ割込みの生成	X
システムリセット生成(2)	X
システムSTOPで有効な機能	X
システムSTANDBYで有効な機能	X
システムSTOP時にウェイクアップ割込みを生成する機能(3)	X
システムSTANDBY時にウェイクアップ割込みを生成する機能	-
マイクロコントローラがデバッグモードに切り替わったときにフリーズする機能(4)	X
STOPモードでアクティビティを制御するためのオプションバイト(5)	X
STANDBYモードでアクティビティを制御するためのオプションバイト(6)	X
ハードウェアモードを制御するためのオプションバイト(7)	X

1. “X” はサポートされています。“-” はサポートされていません。
2. 詳細については、RCCのセクションを参照してください。
3. 割込みによるSTOPからのウェイクアップはSTM32U575/585のSTOP 0、STOP 1、およびSTOP 2モード、およびSTM32U535/545/59x/5Ax/5Fx/5GxのSTOP 0、STOP 1、STOP 2、およびSTOP 3モードでのみサポートされています。
4. DBGセクションのDBG_IWDG_STOPにより制御されます。

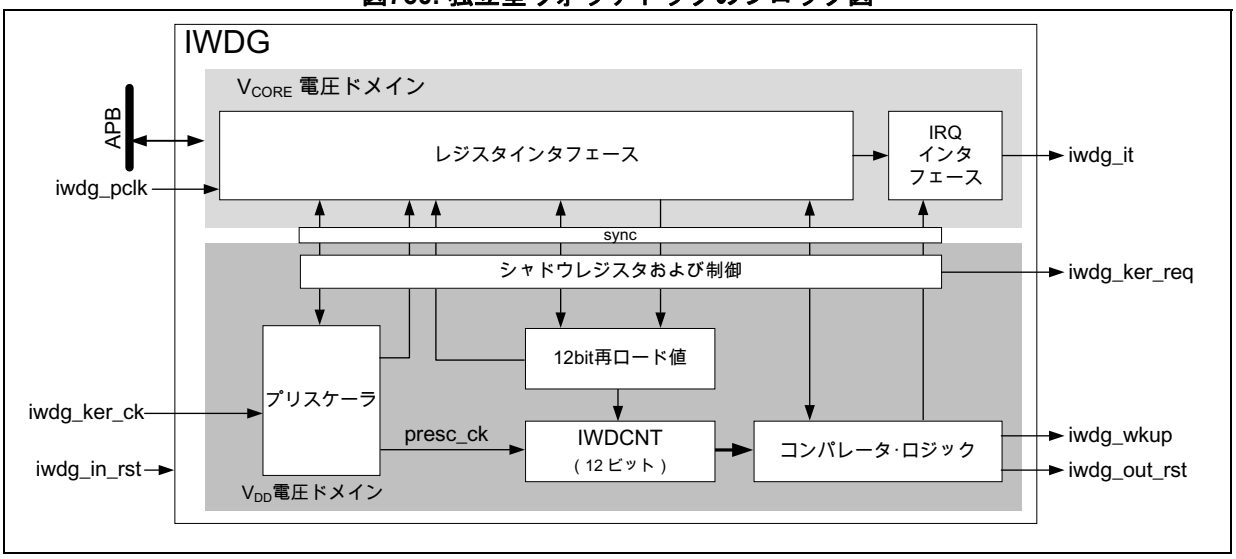
5. FLASHセクションのオプションバイトIWDG_STOPにより制御されます。
6. FLASHセクションのオプションバイトIWDG_STDBYにより制御されます。
7. FLASHセクションのオプションバイトIWDG_SWにより制御されます。

61.4 IWDGの機能説明

61.4.1 IWDGブロック図

図760に、独立型ウォッチドッグモジュールの機能ブロックを示します。

図760. 独立型ウォッチドッグのブロック図



レジスタとIRQインタフェースはV_{CORE} 電圧ドメインに配置されています。ウォッチドッグ機能自体は、低電力モードで機能できるように、V_{DD} 電圧ドメインに配置されています。IWDGの機能については、セクション61.3を参照してください。

レジスタとIRQインタフェースは主にAPBクロック（iwdg_pclk）によってクロック供給されますが、ウォッチドッグ機能は専用のカーネルクロック（iwdg_ker_ck）によってクロック供給されます。2つのドメイン間のデータ交換は、同期メカニズムにより行われます。レジスタインタフェースに配置されたレジスタの大部分は、V_{DD} 電圧ドメインにシャドウイングされます。

IWDGダウンカウンタ（IWDCNT）はプリスケーラクロック（presc_ck）によってクロック供給されます。プリスケーラクロックは、PR[3:0] ビットフィールドに従って、カーネルクロックiwdg_ker_ckをプリスケーラで割って生成されます。

61.4.2 IWDG内部信号

IWDG内部信号のリストは表617を参照してください。

表617.IWDG内部入力／出力信号

信号名	信号タイプ	説明
iwdg_ker_ck	入力	IWDGカーネルクロック
iwdg_ker_req	入力	IDGカーネルクロックリクエスト
iwdg_pclk	入力	IWDG APBクロック
iwdg_out_rst	出力	IWDGリセット出力
iwdg_in_rst	入力	IWDGリセット入力
iwdg_wkup	出力	IWDGウェイクアップ·イベント
iwdg_it	出力	IWDG早期ウェイクアップ割込み

61.4.3 ソフトウェアおよびハードウェアウォッチドッグモード

ウォッチドッグモードを使用して、IWDGを有効化する方法としてソフトウェアコマンド（ソフトウェアウォッチドッグモード）または自動（ハードウェアウォッチドッグモード）を選択できます。それ以外の機能はすべて、ソフトウェアモードでもハードウェアモードでも同様に動作します。

ソフトウェアウォッチドッグモードはデフォルトの動作モードです。独立型ウォッチドッグは IWDGキーレジスタ（IWDG_KR）に値0x0000 CCCCが書き込まれることによって開始され、IWDCNTはリセット値（0xFFF）からカウントダウンを開始します。

ハードウェアウォッチドッグモードでは、独立型ウォッチドッグは起動時に自動的に開始されるか、リセットのたびに（iwdg_in_rst経由で）開始されます。IWDCNTダウンカウンタはリセット値0xFFFからカウントダウンを開始します。ハードウェアウォッチドッグモード機能は、デバイスのオプションビットを通じて有効になります。詳細はセクション61.3を参照してください。

IWDCNTが0x000に達すると、リセット信号が生成されます（iwdg_out_rstをアサート）。

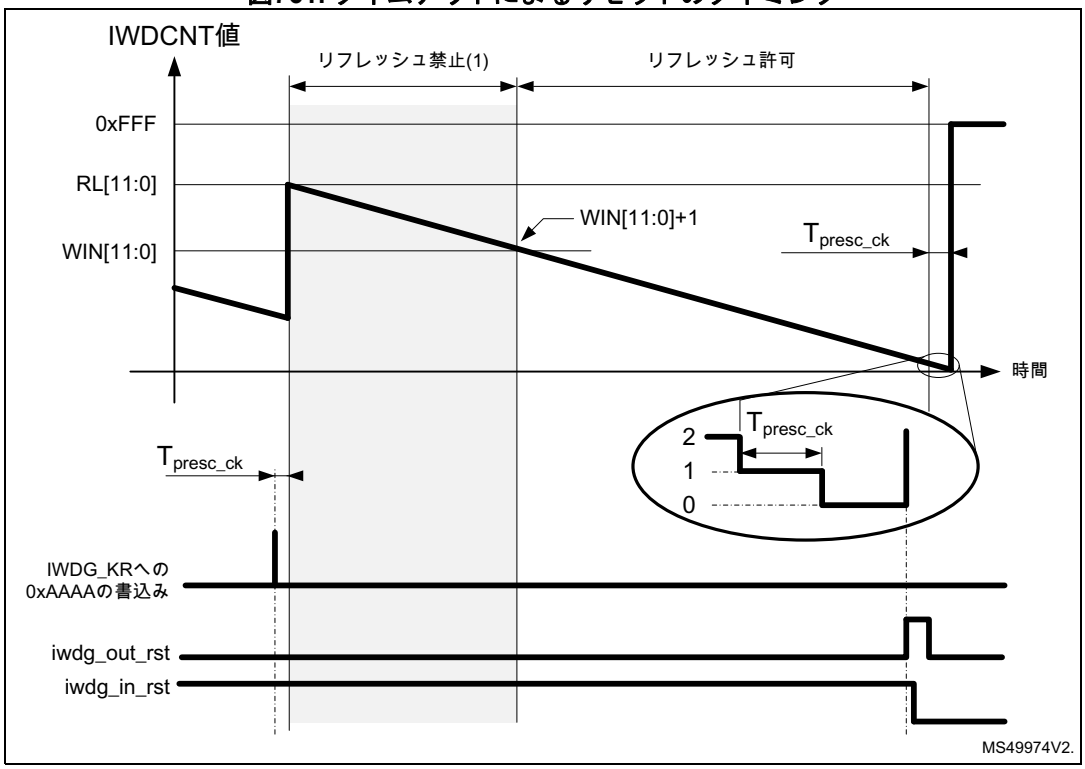
IWDGキーレジスタ（IWDG_KR）にキー値0x0000 AAAAが書き込まれると、IWDG_RLRの値がIWDCNTに再ロードされ、ウォッチドッグのリセットが防止されます。

再同期化の遅延のため、IWDCNTダウンカウンタが1に達する前にIWDGをリフレッシュする必要があります。

IWDGが開始された後、停止できるのはリセット時のみです（iwdg_in_rstをアサート）。

図761に示すように、リフレッシュコマンドが実行されると、presc_ckの1周期後で、RL[11:0] の内容によりIWDCNTが再ロードされます。

図761. タイムアウトによるリセットのタイミング



1. ウィンドウオプションが有効化されている場合

IWDCNTが1に達する前にIWDGがリフレッシュされない場合、IWDGによりリセットが生成されます（iwdg_out_rstをアサート）。それに対して、RCCによりIWDGがリセットされ（iwdg_in_rstをアサート）、リセットソースがクリアされます。

61.4.4 ウィンドウオプション

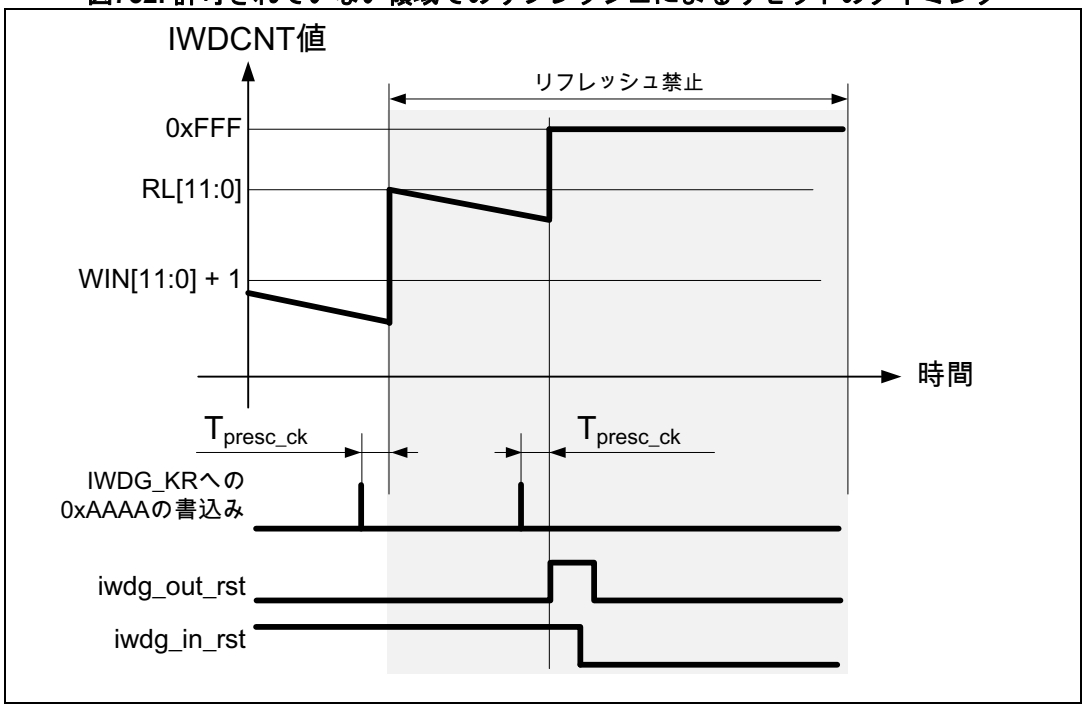
IWDGは、IWDGウィンドウレジスタ（IWDG_WINR）に適切なウィンドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタがWIN[11:0] + 1より大きい間に再ロード操作が行われると、リセットが生成されます。WIN[11:0] は IWDGウィンドウレジスタ（IWDG_WINR）に配置されています。図762に示すように、想定しないリフレッシュコマンドからpresc_ckの1周期後にリセットが生成されます。

IWDGウィンドウレジスタ（IWDG_WINR）のデフォルト値は0x0000 0FFFです。そのため、この値が更新されない場合は、ウィンドウオプションは無効化されます。

ウィンドウ値が変更されるとすぐに、次のリフレッシュが必要な箇所を容易に予測できるように、ダウンカウンタ（IWDCNT）にRL[11:0] 値が再ロードされます。

図762. 許可されていない領域でのリフレッシュによるリセットのタイミング



ウィンドウオプションが有効な場合のIWDGの設定

1. IWDGキーレジスタ（IWDG_KR）に0x0000 CCCCを書き込むことによって、IWDGを有効にします。
2. IWDGキーレジスタ（IWDG_KR）に0x0000 5555を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDGプリスケーラレジスタ（IWDG_PR）をプログラムすることによって、IWDGのプリスケーラに書込みを行います。
4. IWDG再ロードレジスタ（IWDG_RLR）に書き込みます。
5. 必要に応じて、早期ウェイクアップ割込みを有効化し、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）に適切な値を書き込んで早期ウェイクアップ·コンパレータをプログラムします。
6. IWDGウィンドウレジスタ（IWDG_WINR）に書き込みます。これにより、自動的にIWDCNTダウンカウンタにRL[11:0] 値が再ロードされます。
7. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
8. IWDGキーレジスタ（IWDG_KR）に0x0000 0000を書き込んで、レジスタを書込み保護します。

注： **このシーケンスの完了後にアプリケーションでAPBクロックを無効化するつもりがない場合、手順 7 はスキップできます。**

ウィンドウオプションが無効な場合のIWDGの設定

ウィンドウオプションが使用されていない場合、IWDGは以下のように設定することができます。

1. IWDGキーレジスタ（IWDG_KR）に0x0000 CCCCを書き込むことによって、IWDGを有効にします。
2. IWDGキーレジスタ（IWDG_KR）に0x0000 5555を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDGプリスケーラレジスタ（IWDG_PR）をプログラムすることによって、プリスケーラに書込みを行います。
4. IWDG再ロードレジスタ（IWDG_RLR）に書き込みます。
5. 必要に応じて、早期ウェイクアップ割込みを有効化し、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）に適切な値を書き込んで早期ウェイクアップ·コンパレータをプログラムします。
6. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
7. RL[11:0] 値によりカウンタをリフレッシュし、IWDGキーレジスタ（IWDG_KR）に0x0000 AAAAを書き込むことによってレジスタを書込み保護します。

ウィンドウコンパレータの更新

IWDGの動作中は、ウィンドウコンパレータを更新できます。IWDCNTも再ロードされます。ウィンドウコンパレータを更新するためのシーケンスは次のとおりです。

1. IWDGキーレジスタ（IWDG_KR）に0x0000 5555を書き込むことにより、レジスタアクセスを有効化します。
2. IWDGウィンドウレジスタ（IWDG_WINR）に書き込みます。これにより、自動的にIWDCNTダウンカウンタにRL[11:0] 値が再ロードされます。
3. WVU = 0を待ちます。
4. IWDG_KRに0x0000 0000を書き込むことにより、レジスタをロックします。

このシーケンスの完了後にアプリケーションでAPBクロックを無効化するつもりがない場合、手順 3 はスキップできます。

61.4.5 デバッグ

プロセッサがデバッグモードになると（コアは停止状態）、IWDCNTダウンカウンタは、製品のデバッグ機能に応じて、通常どおりに動作を続けるか、または停止します。この製品の機能の詳細については、セクション61.3を参照してください。

61.4.6 レジスタのアクセス保護

IWDGプリスケーラレジスタ（IWDG_PR）、IWDG再ロードレジスタ（IWDG_RLR）、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）、およびIWDGウィンドウレジスタ（IWDG_WINR）への書込みアクセスは保護されます。これを変更するには、まず IWDGキーレジスタ（IWDG_KR）に0x0000 5555を書き込みます。このレジスタに別の値で書込みアクセスすると、シーケンスがブレークされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作（0x0000 AAAAの書込み）の場合に相当します。

ステータスレジスタは、プリスケーラ、あるいはダウンカウンタ再ロード値やウィンドウ値の更新が行われていることを示すために使用されます。

61.5 IWDG低電力モード

オプション·バイトの設定によって、IWDGでは低電力モード時にカウントを継続することも、継続しないこともできます。詳細については、セクション61.3を参照してください。

表618.低消費電力モードがIWDGに与える影響

モード	説明
SLEEP	影響はありません。IWDG割込みによって、デバイスはSLEEPモードから復帰します。
STOP	オプションバイトの設定によって、IWDGは有効または無効になります。 詳細については、セクション61.3を参照してください。 IWDG割込みによって、デバイスはSTOPモードを終了します。
STANDBY	オプションバイトの設定によって、IWDGは有効または無効になります。 詳細については、セクション61.3を参照してください。 IWDG割込みにより、デバイスがSTANDBYモードを終了することはありません。
SHUTDOWN	IWDGは機能しません。

61.6 IWDG割込み

IWDGには、ダウンカウンタの値に応じて早期割込みを生成する機能があります。早期割込みを有効にするには、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）のEWIEビットを1に設定します。

コンパレータ値（EWIT[11:0]）を使用すると、早期割込みを生成する箇所をアプリケーションで定義できます。

IWDCNTダウンカウンタがEWIT[11:0] - 1の値に達すると、iwdg_wkupが有効化され、必要に応じて低電力モードを終了することが可能になります。

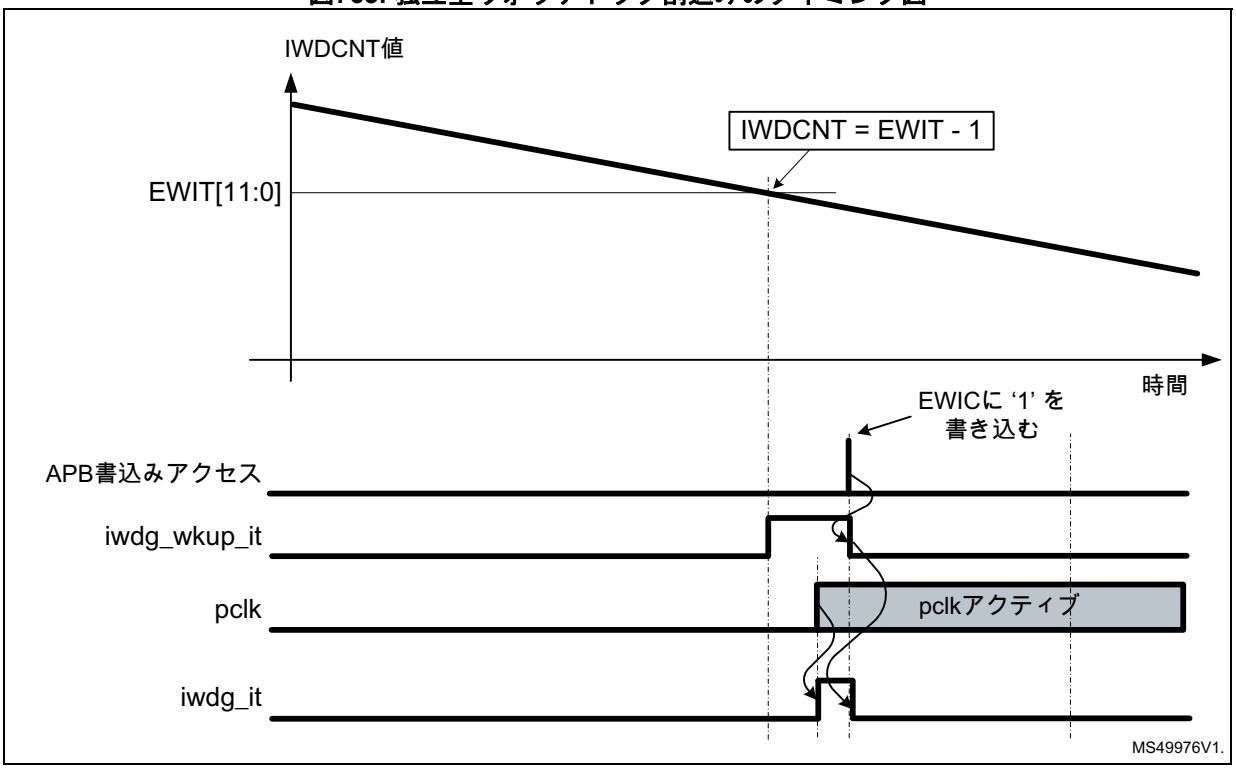
APBクロックが利用できる場合、iwdg_itも有効化されます。

また、IWDGステータスレジスタ（IWDG_SR）のフラグEWIFが1に設定されます。

EWI割込みの確認応答をするには、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）のEWICビットに“1”を書き込みます。

また、IWDG_EWCRレジスタに書き込むと、ダウンカウンタ（IWDCNT）が再ロード値RL[11:0] でリフレッシュされます。

図763. 独立型ウォッチドッグ割込みのタイミング図



ウォッチドッグのリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割込み（EWI）が使用できます。

早期ウェイクアップ·コンパレータ値の変更

次のシーケンスにより、いつでも早期ウェイクアップ·コンパレータ値を変更したり、割込み生成を有効化/無効化したりできます。

1. IWDGキーレジスタ（IWDG_KR）に0x0000 5555を書き込むことによって、レジスタのアクセスを有効にします。
2. 早期ウェイクアップ割込みを有効化 / 無効化したり、IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）に適切な値を書き込んで早期ウェイクアップ·コンパレータをプログラムしたりします。
3. EWU = ‘0’ を待ちます。EWUはIWDGステータスレジスタ（IWDG_SR）に配置されています。
4. IWDGキーレジスタ（IWDG_KR）に0x0000 0000を書き込むことにより、レジスタを書込み保護します。

このシーケンスの完了後にアプリケーションでAPBクロックを無効化するつもりがない場合、手順 3はスキップできます。

表619にIWDG割込みリクエストをまとめます。

表619.IWDG割込みリクエスト

割込みイベント	イベントフラグ	割込みのクリア方法	割込み有効制御ビット	有効な割込み	
				iwdg_it	iwdg_wkup_it
IWDCNTがEWIT値に到達	EWIF	EWICに ‘1’ を書き込む	EWIE	Y(1)	Y(2)

1. iwdg_pclk入力にクロックがある場合に生成されます。
2. iwdg_ker_ck入力にクロックがある場合に生成されます。

61.7 IWDGレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ハーフワード（16bit）またはワード（32bit）単位でアクセスする必要があります。

レジスタインタフェースに配置されたレジスタの大部分は、V_{DD} 電圧ドメインにシャドウイングされます。iwdg_in_rstがアサートされると、V_{DD} 電圧ドメインに配置されたウォッチドッグのロジックとシャドウレジスタがリセットされます。

アプリケーションでウォッチドッグレジスタが読み戻されると、対応するシャドウレジスタの値がハードウェアによりレジスタインタフェースに転送されます。

アプリケーションでウォッチドッグレジスタが書き込まれると、対応するシャドウレジスタがハードウェアで更新されます。

61.7.1 IWDGキーレジスタ（IWDG_KR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **KEY[15:0]：**キー値（書込み専用、読出しでは0x0000）

これらのビットは、アプリケーションによって書き込まれた値に応じて、いくつかの機能に使用できます。

- 0xAAAA：IWDCNTダウンカウンタにRL[11:0] 値を再ロードし（ウォッチドッグのリフレッシュ）、レジスタを書込み保護します。この値は、ソフトウェアによって一定間隔で書き込まれなければなりません。そうしないと、カウンタが0に達したときにウォッチドッグによりリセットが生成されます。
- 0x5555：レジスタへの書込みアクセスを有効にします。
- 0xCCCC：レジスタのウォッチドッグ（ハードウェアウォッチドッグオプションが選択されている場合を除く）と書込み保護を有効にします。
- 0x5555以外の値：レジスタを書込み保護します。

書込み保護メカニズムを備えているのは、IWDG_PR、IWDG_RLR、IWDG_EWCR、およびIWDG_WINRレジスタのみです。

61.7.2 IWDGプリスケーラレジスタ（IWDG_PR）

アドレスオフセット：0x04

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PR[3:0]：**プリスケーラ分周回路

これらのビットは、書込みアクセス保護されています。セクション61.4.6を参照してください。カウンタクロックを供給するプリスケーラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケーラ分周回路を変更できるようにするには、IWDGステータスレジスタ（IWDG_SR）のPVUビットがリセットされる必要があります。

0000：4分周
0001：8分周
0010：16分周
0011：32分周
0100：64分周
0101：128分周
0110：256分周
0111：512分周
その他：1024分周

注： このレジスタを読み出すと、V_{DD} 電圧ドメインからプリスケーラ値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDGステータスレジスタ（IWDG_SR）のPVUビットがリセットされているときのみとなります。

61.7.3 IWDG再ロードレジスタ（IWDG_RLR）

アドレスオフセット：0x08

リセット値：0x0000：0FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RL[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **RL[11:0]**：ウォッチドッグカウンタ再ロード値

これらのビットは、書込みアクセス保護されています。セクション61.4.6を参照してください。IWDGキーレジスタ（IWDG_KR）に値0xAAAAが書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とプリスケーラクロックによって決まります。RL[11:0] に2未満の値を設定することは推奨しません。

再ロード値を変更できるようにするには、IWDGステータスレジスタ（IWDG_SR）のRVUビットがリセットされる必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあるため、このレジスタから読み出された値が有効なのは、IWDGステータスレジスタ（IWDG_SR）のRVUビットがリセットされているときのみとなります。

61.7.4 IWDGステータスレジスタ（IWDG_SR）

アドレスオフセット：0x0C

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	EWIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWU	WVU	RVU	PVU
	r											r	r	r	r

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **EWIF**：ウォッチドッグの早期割込みフラグ

このビットは、早期割込みが保留中であることを示すため、ハードウェアによって ”1” にセットされます。このビットをソフトウェアでクリアするには、IWDG_EWCR レジスタのビットEWICに ”1” を書き込む必要があります。

ビット13:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **EWU**：ウォッチドッグ割込みコンパレータ値の更新

このビットは、割込みコンパレータ値（EWIT[11:0]）の更新またはEWIDの更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで更新操作が完了したときに、ハードウェアによってリセットされます（IWDGカーネルクロックiwdg_ker_ckの最大 3 周期を要する）。

EWIT[11:0] およびEWIEフィールドは、EWUビットがリセットされているときのみ更新できます。

ビット2 **WVU**：ウォッチドッグカウンタウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（IWDGカーネルクロックiwdg_ker_ckの最大 3 周期を要する）。

ウィンドウ値は、WVUビットがリセットされているときのみ更新できます。

このビットは、一般のウィンドウが1の場合のみ生成されます。

ビット1 **RVU**：ウォッチドッグカウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（IWDGカーネルクロックiwdg_ker_ckの最大 3 周期を要する）。

再ロード値は、RVUビットがリセットされているときのみ更新できます。

ビット0 **PVU**：ウォッチドッグプリスケーラ値の更新

このビットは、プリスケーラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインでプリスケーラの更新操作が完了したときに、ハードウェアによってリセットされます（IWDGカーネルクロックiwdg_ker_ckの最大 3 周期を要する）。

プリスケーラ値は、PVUビットがリセットされているときのみ更新できます。

注： **複数の再ロード値、プリスケーラ値、早期割込み位置、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前にRVUビットがリセットされるまで待つか、プリスケーラ値を変更する前にPVUビットがリセットされるまで待つか、ウィンドウ値を変更する前にWVUビットがリセットされるまで待つか、早期割込み位置値を変更する前にEWUビットがリセットされるまで待つ必要があります。プリスケーラ値、再ロード値、ウィンドウ値、早期割込み値を更新した後は、RVU、PVU 、WVU、EWUがリセットされるのを待たずに、コード実行を続けることができます（低電力モードが開始された場合を除く）。**

61.7.5 IWDGウィンドウレジスタ（IWDG_WINR）

アドレスオフセット：0x10

リセット値：0x0000：0FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	WIN[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **WIN[11:0]**：ウォッチドッグカウンタウィンドウ値

これらのビットは書込みアクセス保護されており（セクション61.4.6を参照）、ダウンカウンタと比較されるウィンドウ値の上限を格納しています。

リセットを防ぐには、カウンタの値がWIN[11:0] + 1よりも小さく、1よりも大きいIWDCNTダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、IWDGステータスレジスタ（IWDG_SR）のWVUビットがリセットされる必要があります。

注： **このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDGステータスレジスタ（IWDG_SR）のWVUビットがリセットされているときのみとなります。**

61.7.6 IWDG早期ウェイクアップ割込みレジスタ（IWDG_EWCR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EWIE	EWIC	Res.	Res.	EWIT[11:0]											
rw	w			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EWIE**：ウォッチドッグの早期割込みイネーブル
ソフトウェアでセット／クリアされます。
0：早期割込みインタフェースは無効です。
1：早期割込みインタフェースは有効です。
このビットの値を変更できるようにするには、IWDGステータスレジスタ（IWDG_SR）のEWUビットがリセットされる必要があります。

ビット14 **EWIC**：ウォッチドッグの早期割込みの確認応答
早期ウェイクアップ割込みを確認応答し、EWIFフラグをクリアするには、ソフトウェアでこのビットに1を書き込む必要があります。0を書き込んだ場合は無効で、このフラグを読み出すと0が返されます。

ビット13:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **EWIT[11:0]**：ウォッチドッグカウンタウィンドウ値
これらのビットは、書込みアクセス保護されています（ セクション61.4.6を参照）。早期ウェイクアップ割込みを生成するIWDCNTダウンカウンタの位置を定義するために、ソフトウェアによって書き込まれます。早期割込みは、IWDCNTがEWIT[11:0] - 1以下の場合に生成されます。
EWIT[11:0] は1より大きくなければなりません。
EWIE = 1の場合のみ割込みが発生します。
再ロード値を変更できるようにするには、IWDGステータスレジスタ（IWDG_SR）のEWUビットがリセットされる必要があります。

注： このレジスタを読み出すと、早期ウェイクアップのコンパレータ値とV_{DD} 電圧ドメインの割込み有効ビットが返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあるため、このレジスタから読み出された値が有効なのは、IWDGステータスレジスタ（IWDG_SR）のEWUビットがリセットされているときのみとなります。

61.7.7 IWDGレジスタマップ

表620.IWDGレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	IWDG_KR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	IWDG_PR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[3:0]			
	リセット値																													0	0	0	0
0x08	IWDG_RLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RL[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	IWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWU	WVU	RVU	PVU
	リセット値																		0											0	0	0	0
0x10	IWDG_WINR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WIN[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x14	IWDG_EWCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIE	EWIC	Res.	Res.	EWIT[11:0]											
	リセット値																	0	0			0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

62 システムウィンドウ型ウォッチドッグ（WWDG）

62.1 概要

システムウィンドウ型ウォッチドッグ（WWDG）は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。

ウォッチドッグ回路は、T6bitがクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後にリセットを生成します。リセットは、ダウンカウンタがウィンドウレジスタ値に達する前に7bitのダウンカウンタの値（制御レジスタ内）がリフレッシュされた場合にも生成されます。このことは、限られた時間枠（time-window）の間にカウンタがリフレッシュされなければならないことを意味します。

WWDGクロックは、APB クロックから分周され、また設定可能な時間枠（time-window）があるので、これをプログラムしてアプリケーション動作の異常な進み･遅れを検出できます。

WWDGは、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

62.2 WWDGの主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
  - ダウンカウンタの値が0x40より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）（図765を参照）。
- 早期ウェイクアップ割込み（EWI）: ダウンカウンタが0x40になったときにトリガ（有効であり、ウォッチドッグがアクティブな場合）

62.3 WWDGの実装

表621.WWDGの機能(1)

WWDGのモード／機能	WWDG
ウィンドウ関数	X
早期ウェイクアップ割込みの生成	X
システムリセット生成(2)	X
システムSTOPで有効な機能	-
システムSTANDBYで有効な機能	-
マイクロコントローラがデバッグモードに切り替わったときにフリーズする機能(3)	X
ハードウェアモードを制御するためのオプションバイト(4)	X

1. “X” はサポートされています。“-” はサポートされていません。
2. 詳細については、RCCのセクションを参照してください。
3. DBGブロックのDBG_WWDG_STOPにより制御されます。
4. オプションバイトWWDG_SWで制御されます。WWDG_SWがHWモードでセットされている場合、CPUがRUNモードまたはSLEEPモードに入るとすぐにWWDGが実行されます。

62.4 WWDGの機能説明

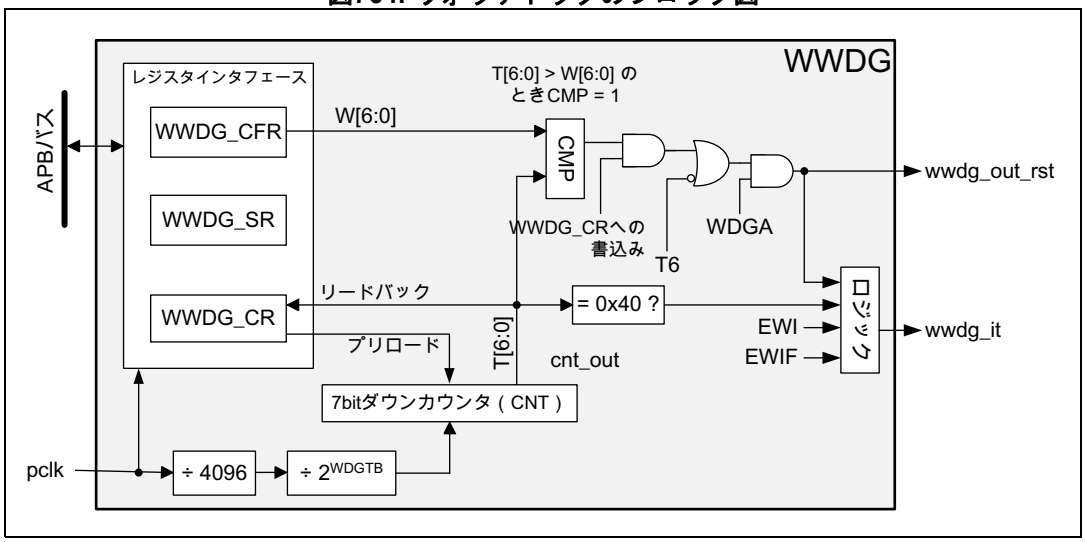
ウォッチドッグが有効な（WWDG_CRレジスタのWDGAビットがセットされている）場合、7bitのダウンカウンタ（T[6:0] ビット）が0x40に達して0x3Fにデクリメントされた（T6がクリアされた）時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

アプリケーションプログラムは、通常動作時には定期的にWWDG_CRレジスタへの書込みを行って、リセットを防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値以下のとき、かつ0x3Fより高いときにのみ実行できます。WWDG_CRレジスタに格納される値は、0xFFから0xC0の間でなければなりません。

WWDGのブロック図については、図764を参照してください。

62.4.1 WWDGブロック図

図764. ウォッチドッグのブロック図



62.4.2 WWDG内部信号

表622にWWDGの内部信号のリストを示します。

表622.WWDG内部入力／出力信号

信号名	信号タイプ	説明
pclk	デジタル入力	APBバスクロック
wwdg_out_rst	デジタル出力	WWDGリセット信号出力
wwdg_it	デジタル出力	WWDG早期割込み出力

62.4.3 ウォッチドッグの有効化

ユーザオプションWWDG_SWで “ソフトウェアウィンドウウォッチドッグ”を選択すると、リセット後、ウォッチドッグは常に無効です。これを有効にするには、WWDG_CRレジスタのWDGAビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

ユーザオプションWWDG_SWで “ハードウェアウィンドウウォッチドッグ”を選択すると、リセット後、ウォッチドッグは常に有効であり、無効にできません。

62.4.4 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6bitをセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。このタイミングは、WWDG_CRレジスタへの書込み時のプリスケーラの状態が不明なので最小値から最大値の間で変化します（図765 参照）。WWDG設定レジスタ（WWDG_CFR）はウィンドウの上限値を含みます。リセットを防ぐには、ダウンカウンタの値がウィンドウレジスタの値以下のとき、かつ0x3Fより高いときに、これを再ロードする必要があります。図765に、ウィンドウ型ウォッチドッグのプロセスを示します。

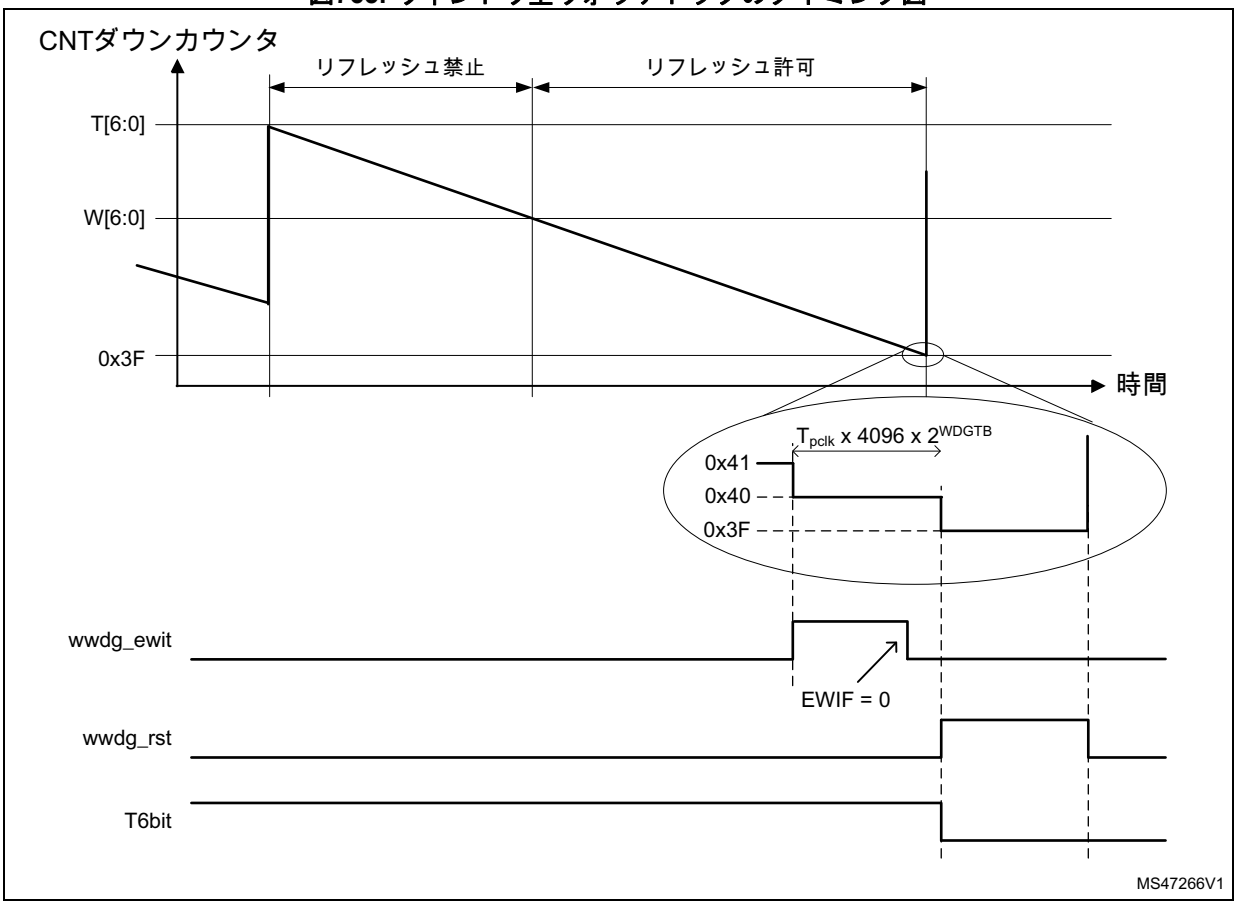
注： **T6bitを使用して、ソフトウェアリセットを生成することができます（WDGAビットはセット、T6bitはクリアされる）。**

62.4.5 ウォッチドッグタイムアウトをプログラムする方法

図765の式を使用して、WWDGのタイムアウトを計算します。

警告： **WWDG_CRレジスタに書き込むときには、ただちにリセットされるのを防ぐために、常にT6bitに1を書き込んでください。**

図765. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGTB[2:0]} \times (T[5:0] + 1) \qquad (ms)$$

ここで、

t_{WWDG}：WWDGタイムアウト

t_{PCLK}：APB クロック周期の測定値（ms）

4096：内部分周器に対応する値

たとえば、APB 周波数が 48MHzの場合、WDGTB[2:0] は3にセットされ、T[5:0] は63にセットされます。

$$t_{WWDG} = (1/48000) \times 4096 \times 2^{3} \times (63 + 1) = 43.69ms$$

t_{WWDG} の最小値と最大値については、データシートを参照してください。

62.4.6 デバッグモード

デバイスがデバッグモードになると（プロセッサは停止状態）、WWDGカウンタは、DBGモジュールの設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、セクション 75：デバッグサポート（DBG）を参照してください。

62.5 WWDG割込み

リセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割込み（EWI）が使用できます。早期ウェイクアップ割込みを有効にするには、アプリケーションで以下の操作を行う必要があります。

- WWDG_SRレジスタのEWIFビットに0を書き込み、望ましくない保留中の割込みをクリアする
- WWDG_CFRレジスタのEWIビットに1を書き込み、割込みを有効にする

ダウンカウンタ値が0x40に到達すると、ウォッチドッグ割込みが生成され、対応する割込みサービスルーチン（ISR）を使用して、デバイスをリセットする前に特定の処理（通信やデータロギングなど）をトリガすることができます。

アプリケーションによっては、EWI割込みを使用して、WWDGリセットを生成せずにソフトウェアのシステムチェックやシステム復旧／グレースフルデグラデーションを管理することができます。この場合、対応するISRでWWDGカウンタを再ロードし、WWDGリセットを回避してから必要な操作をトリガしてください。

ウォッチドッグ割込みは、WWDG_SRレジスタのEWIFビットに“0”を書き込むことによってクリアされます。

注： **優先順位の高いタスクにおけるシステムロックなどによってウォッチドッグ割込みが使用できない場合、最終的にはWWDGリセットが生成されます。**

表623.WWDG割込みリクエスト

割込み		イベントフラグ	イネーブル制御ビット	割込みのクリア方法	モードの終了		
ベクタ	イベント				SLEEP	STOP(1)	STANDBY(1)
WWDG(2)	早期ウェイクアップ割込み	EWIF	EWI	EWIFフラグに0を書き込む	可能	不可	なし

1. WWDG割込みでは追加の機能が利用できます。詳細は、セクション62.3 を参照してください。
2. WWDGベクタはwwdg_it信号のアサートに相当します。

62.6 WWDGレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ハーフワード（16bit）またはワード（32bit）単位でアクセスすることができます。

62.6.1 WWDG制御レジスタ（WWDG_CR）

アドレスオフセット：0x000

リセット値：0x0000 007F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
								rs	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WDGA**：有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。WDGA = 1のとき、ウォッチドッグはリセットを生成できます。

0：ウォッチドッグは無効です。

1：ウォッチドッグは有効です。

ビット6:0 **T[6:0]:** 7bitカウンタ（MSBからLSBまで）

これらのビットは、ウォッチドッグカウンタの値を含みます。

（4096 x $2^{WDGTB[2:0]}$）PCLKサイクルごとにデクリメントされます。0x40に達して0x3Fにデクリメントされると（T6がクリアされると）、リセットが生成されます。

62.6.2 WWDG設定レジスタ（WWDG_CFR）

アドレスオフセット：0x004

リセット値：0x0000 007F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	WDGTB[2:0]			Res.	EWI	Res.	Res.	W[6:0]						
		rw	rw	rw		rs			rw	rw	rw	rw	rw	rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13:11 **WDGTB[2:0]：**タイマベース

プリスケーラのタイムベースは、次のように変更できます。

000：CKカウンタクロック（PCLK/4096）1分周

001：CKカウンタクロック（PCLK/4096）2分周

010：CKカウンタクロック（PCLK/4096）4分周

011：CKカウンタクロック（PCLK/4096）8分周

100：CKカウンタクロック（PCLK/4096）16分周

101：CKカウンタクロック（PCLK/4096）32分周

110：CKカウンタクロック（PCLK/4096）64分周

111：CKカウンタクロック（PCLK/4096）128分周

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **EWI：**早期ウェイクアップ割込みイネーブル

ソフトウェアでセットされ、リセット後はハードウェアによってクリアされます。このビットがセットされているときには、カウンタの値が0x40に達したときに割込みが発生します。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **W[6:0]:** 7bitウィンドウ値

これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

62.6.3 WWDGステータスレジスタ（WWDG_SR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
															rc_w0

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EWIF：**早期ウェイクアップ割込みフラグ

このビットは、カウンタの値が0x40に達したときにハードウェアによってセットされます。0を書き込んでソフトウェアでクリアする必要があります。1を書き込んでも、ビットの値は変化しません。このビットは、割込みが有効でない場合にもセットされます。

62.6.4 WWDGレジスタマップ

次の表に、WWDGレジスタマップとリセット値を示します。

表624.WWDGレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	WWDG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
	リセット値																									0	1	1	1	1	1	1	1
0x004	WWDG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGTB [2:0]			Res.	EWI	Res.	Res.	W[6:0]						
	リセット値																			0	0	0		0			1	1	1	1	1	1	1
0x008	WWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
	リセット値																																0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

63 リアルタイムクロック（RTC）

63.1 概要

RTCは、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック（RTC）は、独立したBCDタイマ／カウンタです。RTCは、プログラム可能なアラーム割込みを備えた時刻クロック／カレンダを搭載しています。

供給電圧が動作範囲内にある間は、デバイスのステータス（実行モード、低消費電力モード、またはリセット中）に関係なく、RTCが停止することはありません。

RTCはV_{BAT} モードで機能します。

63.2 RTCの主な機能

RTCは、次の機能をサポートしています（図 766：RTCのブロック図を参照）。

- サブセカンド、秒、分、時（12または24時間形式）、曜日、日、月、年にBCD（2進化10進数）形式で対応するカレンダ。
- 32bitのフリーランニングカウンタを使用するバイナリモード。
- 月の28日、29日（うるう年）、30日、31日の自動補正
- 2つのプログラム可能なアラーム
- 1～32767のRTCクロックパルスの動作中の補正。マスタクロックとの同期に使用可能
- リファレンスクロック検出：より正確な秒のクロックソース（50または60Hz）の使用で、カレンダの精度を向上。
- 0.95 ppmの分解能を持つデジタル較正回路で、クオーツクリスタルの精度を補償
- カレンダの内容の保存に使用できるタイムスタンプ機能。この機能は、タイムスタンプピンでのイベント、またはタンパイベント、あるいはV_{BAT}モードへの切り替えでトリガ可能
- 分解能と周期がプログラム可能な周期的なイベント用 17bit 自動再ロードウェイクアップタイマ（WUT）
- TrustZoneサポート：
  - RTCの完全なセキュリティ
  - アラームA、アラームB、ウェイクアップタイマおよびタイムスタンプの個別のセキュア/非セキュア設定
- アラームA、アラームB、ウェイクアップタイマおよびタイムスタンプの個別の特権保護

RTCの電源は、V_{DD} 供給（ある場合）またはVBATピンから取得するスイッチ経由で供給されます。

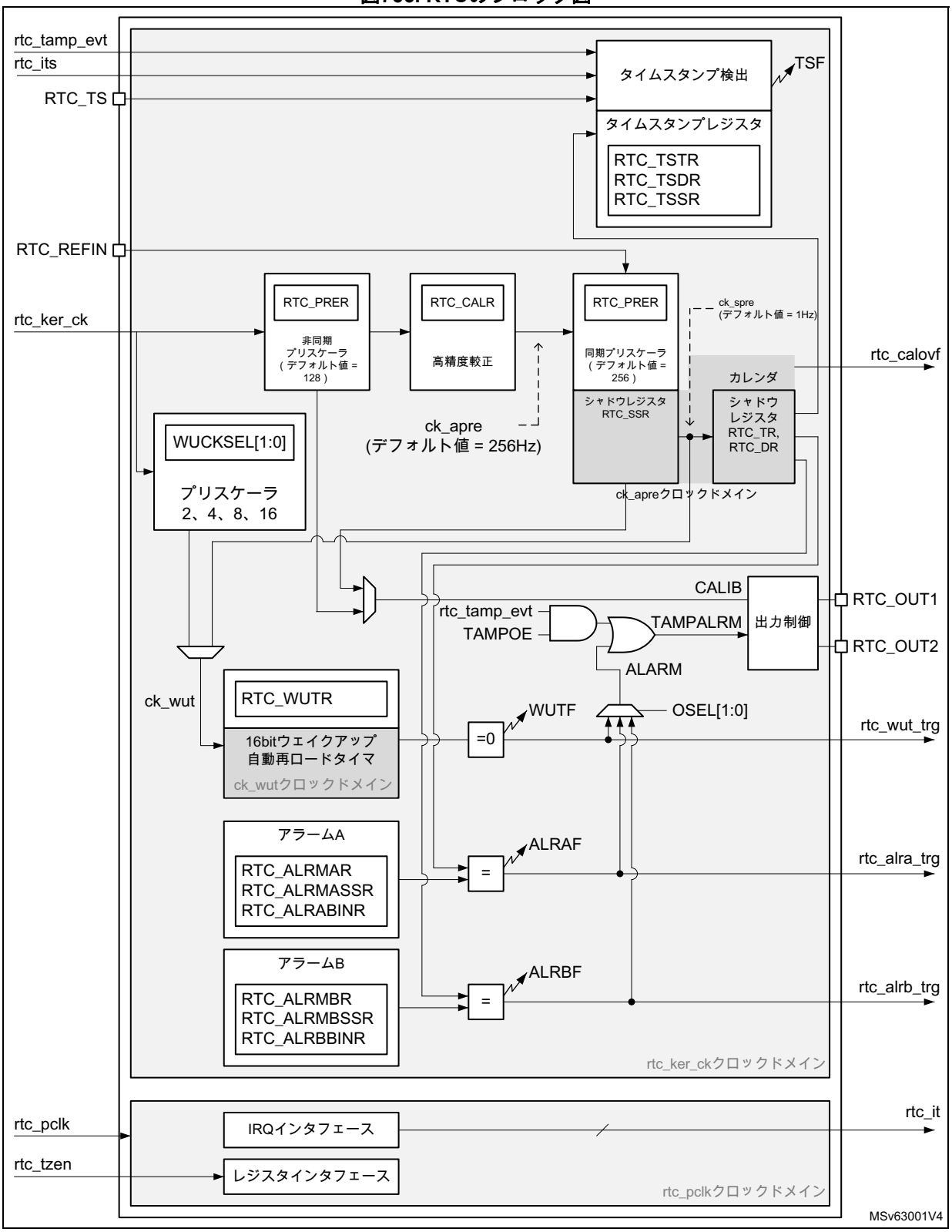
RTCはLSEによってクロック供給されている場合、V_{BAT} モードおよびすべての低消費電力モードで機能します。

すべてのRTCイベント（アラーム、ウェイクアップタイマ、タイムスタンプ）で割込みを生成して、デバイスを低消費電力モードからウェイクアップさせることができます。

63.3 RTCの機能説明

63.3.1 RTCのブロック図

図766. RTCのブロック図



63.3.2 RTCピンおよび内部信号

表625.RTCの入出力ピン

ピン名	信号タイプ	説明
RTC_TS	入力	RTCタイムスタンプ入力
RTC_REFIN	入力	RTC 50または60 Hzリファレンスクロック入力
RTC_OUT1	出力	RTC出力1
RTC_OUT2	出力	RTC出力2

RTC_OUT1 およびRTC_OUT2 は、次の2つの出力のうち1つを選択します。

- CALIB:512 Hzまたは1 Hzのクロック出力（LSE周波数32.768 kHzの場合）。この出力は、RTC_CRレジスタのCOEビットをセットして有効にします。
- TAMPALRM：この出力は、rtc_tamp_evt信号とALARM信号の論理和（OR）です。

ALARMは、アラームA、アラームB、またはウェイクアップ出力を選択するRTC_CRレジスタのOSEL[1:0] ビットを設定することで有効にします。rtc_tamp_evtは、タンパイベント出力を選択するRTC_CRレジスタのTAMPOEビットをセットすることで有効にします。

表626.RTC内部入力／出力信号

内部信号名	信号タイプ	説明
rtc_ker_ck	入力	RTCカーネルクロック（本書ではRTCCLKとも言います）
rtc_pclk	入力	RTC APBクロック
rtc_its	入力	RTC内部タイムスタンプイベント
rtc_tamp_evt	入力	TAMPペリフェラルで検出されたタンパイベント（内部または外部）
rtc_tzen	入力	RTC TrustZone有効
rtc_it	出力	RTC割込み（詳細については、セクション 63.5 : RTC割込みを参照してください）
rtc_alra_trg	出力	RTCアラームAイベント検出トリガ
rtc_alrb_trg	出力	RTCアラームBイベント検出トリガ
rtc_wut_trg	出力	RTCウェイクアップタイマイベント検出トリガ
rtc_calovf	出力	RTCカレンダオーバーフロー：この信号は、RTCカレンダが2099年12月31日23:59:59に最大値に達すると生成されます。このとき、カレンダは停止し、オーバーフローしません。

通常、RTCカーネルクロックは、RCCで他のクロックソースを選択できますが、32.768kHzでのLSEです（詳細はRCCを参照してください）。選択したクロックがLSEではない場合、一部の低電力モードやV_{BAT} で使用できない機能があります。詳細については、セクション 63.4 : RTC低消費電力モードを参照してください。

表627.RTC相互接続

信号名	転送元／転送先
rtc_its	電源制御（PWR）から：V_{BAT} 検出出力の主電源の損失／切り替え
rtc_tamp_evt	TAMPペリフェラルから：tamp_evt
rtc_tzen	FLASHオプションバイトから：TZEN
rtc_calovf	TAMPペリフェラルへ：tamp_itamp5

TZENオプションビットは、デバイスでTrustZoneをアクティブにするために使用します。

TZEN = 1：TrustZoneは有効です。

TZEN = 0：TrustZoneは無効です。

TrustZoneが無効である場合、RTCレジスタへのAPBアクセスは非セキュアとなります。

トリガ出力は、その他のペリフェラルのトリガとして使用できます。

63.3.3 RTCおよびTAMPによって制御されるGPIO

バッテリバックアップドメイン（V_{BAT}）に含まれるGPIOは、どのGPIO設定であっても、これらのI/Oでの機能を備えたペリフェラルで直接制御されます。

RTC_OUT1、RTC_TS、TAMP_IN1およびTAMP_OUT2は、同じピン（PC13）に配置されます。PC13に配置されたRTCおよびTAMPの機能は、すべての低消費電力モードとV_{BAT} モードで使用できます。

出力形式は 表628に示す優先順位に従います。

表628.RTCピンPC13の設定(1)

PC13ピンの機能	OSEL[1:0]（ALARM出力イネーブル）	TAMPOE（TAMPER出力イネーブル）	COE（CALIB出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	ATOSHARE=0でTAMP2E=TAMP2AM=1、またはATOSHARE=1およびATOSELx=1でTAMPxE=TAMPxAM=1	TAMP1E（TAMP_IN1入力イネーブル）	TSE（RTC_TS入力イネーブル）
TAMPALRM出力プッシュプル	01、10、または11	0	無視	無視	0	0	無視	無視	無視
	00	1							
	01、10、または11	1							

表628.RTCピンPC13の設定(1) （続き）

PC13ピンの機能		OSEL[1:0]（ALARM出力イネーブル）	TAMPOE（TAMPER出力イネーブル）	COE（CALIB出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	ATOSHARE=0でTAMP2E=TAMP2AM=1、またはATOSHARE=1およびATOSELx=1でTAMPxE=TAMPxAM=1	TAMP1E（TAMP_IN1入力イネーブル）	TSE（RTC_TS入力イネーブル）
TAMPALRM出力オープンドレイン(2)	プルアップ・プルダウンなし	01、10、または11	0	無視	無視	1	0	無視	無視	無視
		00	1							
		01、10、または11	1							
	内部プルアップ	01、10、または11	0	無視	無視	1	1	無視	無視	無視
		00	1							
		01、10、または11	1							
CALIB出力PP		00	0	1	0	無視	無視	無視	無視	無視
TAMP_OUT2出力PP		00	0	0	0	無視	無視	1	無視	無視
TAMP_IN1入力フローティング		00	0	0	無視	無視	無視	0	1	0
		00	0	1	1					
		無視	無視	0						
RTC_TSおよびTAMP_IN1入力フローティング		00	0	0	無視	無視	無視	0	1	1
		00	0	1	1			0		
		無視	無視	0				0		
RTC_TS入力フローティング		00	0	0	無視	無視	無視	0	0	1
		00	0	1	1			0		
		無視	無視	0				0		
ウェイクアップピンまたは標準GPIO		00	0	0	無視	無視	無視	0	0	0
		00	0	1	1			0		
		無視	無視	0				0		

1. OD：オープンドレイン、PP：プッシュプル
2. この設定では、GPIOを入力で設定する必要があります。

さらに、OUT2ENビットにより、RTC_OUT2を PB2 ピンに出力することも可能です。表629の表に示すとおり、OSEL、COEおよびOUT2ENの設定に応じて、異なる機能がRTC_OUT1やRTC_OUT2に配置されます。

表629.RTC_OUTの配置

OSEL[1:0] ビット（ALARM出力イネーブル）	COEビット（CALIB出力イネーブル）	OUT2ENビット	PC13でのRTC_OUT1	PB2 でのRTC_OUT2
00	0	0	-	-
00	1		CALIB	-
01、10、または11	無視		TAMPALRM	-
00	0	1	-	-
00	1		-	CALIB
01、10、または11	0		-	TAMPALRM
01、10、または11	1		TAMPALRM	CALIB

63.3.4 RTCセキュア保護モード

デフォルトでは、バックアップドメインのパワーオンリセット後、セキュアモードのみで書込み可能なRTCセキュア設定レジスタ（RTC_SECCFGR）を除いて、すべてのRTCレジスタがセキュアモードと非セキュアモードの両方で読出しまたは書込みできます。RTC保護設定は、システムリセットの影響を受けません。

RTC_SECCFGRレジスタのSECビットがセットされている場合：

- RTCレジスタへの書込みはセキュアモードでのみ可能です。
- RTC_SECCFGR、RTC_PRIVCFGR、RTC_MISR、RTC_TR、RTC_DR、RTC_SSR、RTC_PRER、およびRTC_CALR の読出しは、セキュアモードと非セキュアモードで常に可能です。その他のすべてのRTCレジスタは、セキュアモードでのみ読出しできます。

SECビットがクリアされている場合でも、専用のINITSEC、CALSEC、TSSEC、WUTSEC、ALRASECまたはALRBSECの制御ビットをセットすることで、いくつかのレジスタを保護できます。これらすべてのビットもセットされている場合、すべてのRTCレジスタをセキュアモードおよび非セキュアモードで読み書きできます。

- INITSECがセットされている場合：
  - RTC_TR、RTC_DR、RTC_PRERのレジスタ、INIT、BINおよびBCDU（RTC_ICSR）、FMT制御ビット（RTC_CR）、INITPRIV（RTC_PRIVCFGR）は、セキュアモードでのみ書込みできます。
  - これらのレジスタと制御ビットは、セキュアモードおよび非セキュアモードで読出しできます。
- CALSECがセットされている場合：
  - RTC_SHIFTRとRTC_CALRのレジスタ、およびADD1H、SUB1H、REFCKONの制御ビット（RTC_CR）、CALPRIV（RTC_PRIVCFGR）は、セキュアモードでのみ書込みできます。
  - これらのレジスタと制御ビットは、セキュアモードおよび非セキュアモードで読出しできます。
- ALRASECがセットされている場合：
  - RTC_ALRMAR、RTC_ALRMASSR、RTC_ALRABINR のレジスタ、およびALRAE、ALRAFCLR、ALRAIE、SSRUIE（RTC_CR）、CALRAF、CSSRUF（RTC_SCR）、ALRAF、SSRUF

（RTC_SR）、およびALRAMF、SSRUMF（RTC_SMISR）は、セキュアモードでのみ読み書きできます。

– ALRAPRIV（RTC_PRIVCFGR）はセキュアモードでのみ書込みできます。

- ALRBSECがセットされている場合：
  – RTC_ALRMBR、RTC_ALRMBSSR、RTC_ALRBBINR のレジスタ、およびALRBE ALRBFCLR、ALRBIE（RTC_CR）、CALRBF（RTC_SCR）、ALRBF（RTC_SR）、ALRBMF（RTC_SMISR）は、セキュアモードでのみ読み書きできます。
  – ALRBPRIV（RTC_PRIVCFGR）は、セキュアモードでのみ書込みできます。
- WUTSECがセットされている場合：
  – RTC_WUTR レジスタ、および WUTE、WUTIE、WUCKSEL の制御ビット（RTC_CR）、CWUTF（RTC_SCR）、WUTF（RTC_SR）、WUTMF（RTC_SMISR）は、セキュアモードでのみ読み書きできます。
  – WUTPRIV（RTC_PRIVCFGR）は、セキュアモードでのみ書込みできます。
- TSSECがセットされている場合：
  – RTC_TSTR、RTC_TSDR およびRTC_TSSSR のレジスタ、およびTAMPTS、ITSE、TSE、TSIE、TSEDGEの制御ビット（RTC_CR）、CITSF、CTSOVF、CTSFのビット（RTC_SCR）、TSF、TSOVF、ITSF（RTC_SR）、TSMF、TSOVMF、ITSMF（RTC_SMISR）は、セキュアモードでのみ読み書きできます。
  – TSPRIV（RTC_PRIVCFGR）は、セキュアモードでのみ書込みできます。

セキュア保護されたレジスタへの非セキュアアクセスは拒否されます。

- 生成されるバスエラーはありません。
- レジスタがグローバル保護されている場合：TZIC（TrustZone不正アクセスコントローラ）でのフラグ/割込みを通じて通知が生成されます。レジスタの数ビットだけが保護されている場合（RTC_CRなど混合機能を持ったレジスタ）、通知は生成されません。
- 書込み保護されている場合、そのビットは書き込まれません。
- 読出し保護されている場合、そのビットは0として読み出されます。

1つ以上の機能がセキュアに設定されると、RTCリセットおよびクロック制御もRCCでセキュアになります。

63.3.5 RTC特権保護モード

デフォルトでは、バックアップドメインのパワーオンリセット後、特権モードのみで書込み可能な RTC特権モード制御レジスタ（RTC_PRIVCFGR）を除いて、すべてのRTCレジスタが特権と非特権モードの両方で読出しまたは書込みできます。RTC保護設定は、システムリセットの影響を受けません。

RTC_PRIVCFGRレジスタのPRIVビットがセットされている場合：

- RTCレジスタへの書込みは特権モードでのみ可能です。
- RTC_SECCFGR、RTC_PRIVCFGR、RTC_TR、RTC_DR、RTC_SSR、RTC_PRERおよびRTC_CALR の読出しは、特権と非特権モードで常に可能です。その他のすべてのRTCレジスタは、特権モードでのみ読出しできます。

PRIVビットがクリアされている場合、専用のINITPRIV、CALPRIV、TSPRIV、WUTPRIV、ALRAPRVまたはALRBPRIVの制御ビットをセットすることで、いくつかのレジスタを保護できます。これらすべてのビットもクリアされている場合、すべてのRTCレジスタを特権および非特権モードで読み書きできます。

- INITPRIVがセットされている場合：
  – RTC_TR、RTC_DR、RTC_PRERのレジスタ、INIT、BINおよびBCDU（RTC_ICSR）、FMT制御ビット（RTC_CR）、INITSEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。

  - これらのレジスタと制御ビットは、特権および非特権モードで読出しできます。
- CALPRIVがセットされている場合：
  - RTC_SHIFTR と RTC_CALR のレジスタ、および ADD1H、REFCKON の制御ビット（RTC_CR）、CALDSEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。
  - これらのレジスタと制御ビットは、特権および非特権モードで読出しできます。
- ALRAPRIVがセットされている場合：
  - RTC_ALRMAR、RTC_ALRMASSRおよびRTC_ALRABINRレジスタ、ALRAE ALRAFCLR、ALRAIE およびSSRUIE（RTC_CR）、およびCALRAF およびCSSRUF（RTC_SCR）、ALRAF およびSSRUF（RTC_SR）、ALRAMF およびSSRUMF（RTC_SMISR）、RTC_SMISR は特権モードでのみ読み書きできます。
  - ALRASEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。
- ALRBPRIVがセットされている場合：
  - RTC_ALRMBR、RTC_ALRMBSSR、RTC_ALRBBINRのレジスタ、ALRBE ALRBFCLR、ALRBIE（RTC_CR）、CALRBF（RTC_SCR）、ALRBF（RTC_SR）、ALRBMF（RTC_MISR およびRTC_SMISR）は、特権モードでのみ読み書きできます。
  - ALRBSEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。
- WUTPRIVがセットされている場合：
  - RTC_WUTR レジスタ、および WUTE、WUTIE、WUCKSEL の制御ビット（RTC_CR）、CWUTF（RTC_SCR）、WUTF（RTC_SR）、WUTMF（RTC_MISR およびRTC_SMISR）は、特権モードでのみ読み書きできます。
  - WUTSEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。
- TSPRIVがセットされている場合：
  - RTC_TSTR、RTC_TSDR およびRTC_TSSSR のレジスタ、およびTAMPTS、, ITSE、TSE、TSIE、TSEDGEの制御ビット（RTC_CR）、CITSF、CTSOVF、CTSFのビット（RTC_SCR）、TSF、TSOVF、ITSF（RTC_SR）、TSMF、TSOVMF、ITSMF（RTC_MISR）、およびRTC_SMISR は、特権モードでのみ読み書きできます。
  - TSSEC（RTC_SECCFGR）は、特権モードでのみ書込みできます。

特権保護されたレジスタへの非特権アクセスは拒否されます。

- 生成されるバスエラーはありません。
- 書込み保護されている場合、そのビットは書き込まれません。
- 読出し保護されている場合、そのビットは0として読み出されます。

63.3.6 クロックとプリスケーラ

RTCクロック（RTCCLK）ソース設定の詳細については、「リセットおよびクロック制御（RCC）」を参照してください。

BCDモード（BIN=00）

プログラム可能なプリスケーラステージで、カレンダの更新に使用する1Hzのクロックを生成します。消費電力を最少に抑えるため、プリスケーラは以下に示す2つのプログラム可能なプリスケーラに分割されます（ 図 766：RTCのブロック図を参照）。

- RTC_PRERレジスタのPREDIV_Aビットで設定される7bitの非同期プリスケーラ
- RTC_PRERレジスタのPREDIV_Sビットで設定される15bitの同期プリスケーラ

注： **両方のプリスケーラを使用する場合は、非同期プリスケーラを高い値に設定して消費を最低限に抑えることをお勧めします。**

LSE周波数32.768kHzで1Hz（ck_spre）の内部クロック周波数を得るため、非同期プリスケーラ分周比は128、同期プリスケーラの分周比は256に設定されます。

最低分周比は1、最大分周比は2^{22} です。

これは、約4 MHzの最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apreクロックは、サブセカンドダウンカウンタであるバイナリRTC_SSRにクロックを供給するために使用されます。値がゼロになると、RTC_SSRは、PREDIV_Sの内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spreクロックは、カレンダの更新に、または16bitウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16bitウェイクアップ自動再ロードタイマを、プログラム可能な4bit非同期プリスケーラで分周したRTCCLKで動作させることもできます（詳細は セクション 63.3.10：周期的自動ウェイクアップを参照）。

バイナリモード（BIN=01）

SSRバイナリダウンカウンタは32bit長に拡張され、フリーランニングです。日付と時刻のカレンダであるBCDレジスタは機能しません。

このダウンカウンタは、ck_apreによってクロック供給され、RTC_PRERレジスタのPREDIV_Aビットを通して設定される7bit非同期プリスケーラの出力です。

PREDIV_Sの値は無視されます。

混合モード（BIN=10または11）

SSRバイナリダウンカウンタは32bit長に拡張され、フリーランニングです。日付と時刻のカレンダであるBCDレジスタも利用できます。

このダウンカウンタは、ck_apreによってクロック供給され、RTC_PRERレジスタのPREDIV_Aビットを通して設定される7bit非同期プリスケーラの出力です。ビットBCDU[2:0] は、SSR最下位ビットを使用して、カレンダを1秒ずつインクリメントするタイミングを定義するのに使用します。

63.3.7 リアルタイムクロックとカレンダ

RTCカレンダの時刻および日付レジスタには、PCLK（APBクロック）と同期するシャドウレジスタからアクセスします。同期するための待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用RTC_SSR
- 時刻用RTC_TR
- 日付用RTC_DR

RTCCLKサイクルごとに現在のカレンダ値がシャドウレジスタにコピーされ、RTC_ICSRレジスタのRSFビットがセットされます（セクション 63.6.12 : RTCシフト制御レジスタ（RTC_SHIFTR）を参照）。STOPモードおよびSTANDBYモードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 4サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CRレジスタのBYPSHAD制御ビットをセットすることにより、カレンダレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TRまたはRTC_DRレジスタをBYPSHAD = 0の状態で読み出す際は、APBクロックの周波数 （f_{APB}）は、RTCクロック（f_{RTCCLK}）の周波数の7倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

63.3.8 カレンダ超低消費電力モード

RTC_CALRレジスタのLPCALビットをセットすることによって、RTC電力消費を大幅に削減できます。この設定では、RTC全体がRTCCLKとck_apreの両方ではなく、ck_apreのみによってクロック供給されます。その結果、一部のフラグ遅延が長くなり、較正時間範囲も長くなります（セクション : RTC超低消費電力モードを参照）。

非同期プリスケーラ分周比（PREDIV_A+1）が2の累乗の数ではない場合、LPCALビットは無視されます（0とみなされます）。

LPCAL=0からLPCAL=1へ、またはLPCAL=1からLPCAL=0への切り替えは即時ではなく、完了にck_apre数周期かかります。

63.3.9 プログラム可能なアラーム

RTCユニットにはプログラム可能なアラーム（アラームAとアラームB）が搭載されています。以下にアラームAの説明を示していますが、アラームBにも転用できます。

プログラム可能なアラーム機能は、RTC_CRレジスタのALRAEビットを通じて有効にします。

ALRAFは、カレンダのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタRTC_ALRMASSRおよびRTC_ALRMARにプログラムされている値と一致する場合は1にセットされます。各カレンダ項目は、RTC_ALRMARレジスタのMSKxビットおよびRTC_ALRMASSRレジスタのMASKSSxビットで個別に選択できます。

バイナリモードを使用する場合、アラームバイナリレジスタRTC_ALRABINRでサブセカンドフィールドをプログラムできます。

アラームの割込みは、RTC_CRレジスタのALRAIEビットを通じて有効にします。

アラームを使用して別のペリフェラル用にトリガーイベントを生成する場合、RTC_CRレジスタでALRAFCLRビットに1を設定すると、ハードウェアにより自動的にALRAFが作成されます。この設定では、ALRAFフラグをクリアするためにソフトウェアによる介入は必要はありません。

注意： **秒の項目が選択されている場合（RTC_ALRMARでMSK1bitがリセットされている場合）、正しい動作を保証するため、RTC_PRERレジスタでセットされる同期プリスケーラの分周比は3以上である必要があります。**

アラームAおよびアラームB（RTC_CRレジスタのOSEL[1:0] ビットで有効になっている場合）は、TAMPALRM出力に送ることができます。TAMPALRM出力の極性は、RTC_CRレジスタのPOLビットを通じて設定できます。

63.3.10 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16bitのプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は17bitまで拡張できます。

ウェイクアップ機能は、RTC_CRレジスタのWUTEビットを通じて有効にします。

ウェイクアップタイマクロック入力ck_wutには、次のものが使用できます。

- 2、4、8、または16分周したRTCクロック（RTCCLK）

  RTCCLKがLSE（32.768kHz）である場合、最小分解能61µsで、ウェイクアップ割込み周期を122µsから32秒の範囲で設定できます。
- BCDモードのck_spre（通常は1 Hzの内部クロック）、またはバイナリモードまたは混合モード（BCD-binary）でBCDUでの定義に従ってカレンダを更新するのに使用されるクロック。

  ck_spreの周波数が1 Hzの場合、1秒の分解能により、1秒から約36時間までのウェイクアップタイムが得られます。このプログラム可能な広い時間範囲は、2つの部分に分かれます。
  - WUCKSEL[2:1] = 10の場合は1秒から18時間です。
  - WUCKSEL[2:1] = 11の場合は約18時間から36時間です。この最後のケースでは、2^{16} が16bitカウンタの現在値に追加されます。初期化シーケンスが完了すると（2575ページのウェイクアップタイマのプログラミングを参照）、タイマがカウントダウンを開始します。ウェイクアップ機能が有効である場合、カウントダウンは低消費電力モードでアクティブのままとなります。さらに、カウンタが0に到達すると、RTC_SRレジスタのWUTFフラグがセットされ、ウェイクアップカウンタが再ロード値（RTC_WUTRレジスタ値）で自動的に再ロードされます。

RTC_WUTRレジスタのWUTOCLRに応じて、WUTFフラグはソフトウェアでクリアする必要があり（WUTOCLR = 0x0000）、そうしない場合は自動再ロードダウンカウンタがWUTOCLRの値に達したときに（0x0000 < WUTOCLR ≤ WUT）、WUTFがハードウェアによって自動でクリアされます。

ウェイクアップフラグは、その他のペリフェラルで使用できる内部信号rtc_wutで出力されます（セクション 63.3.1：RTCのブロック図のセクションを参照）。

RTC_CRレジスタのWUTIEビットをセットして周期的ウェイクアップ割込みを有効にすると、デバイスは低消費電力モードを終了できます。

周期的ウェイクアップフラグは、RTC_CRレジスタのOSEL[1:0] ビットを通じて有効になっている場合に、TAMPALRM出力に送ることができます。TAMPALRM出力の極性は、RTC_CRレジスタのPOLビットを通じて設定できます。

低消費電力モード（SLEEP、STOP、STANDBY）と同様に、システムリセットもウェイクアップタイマには影響しません。

63.3.11 RTCの初期化と設定

RTCバイナリ、BCD、または混合モード

デフォルトではRTCはBCDモード（RTC_ICSRレジスタでBIN = 00）: RTC_SSRレジスタにはサブセカンドフィールドSS[15:0] があり、ck_apreによってクロック供給され、1 Hzのクロックを生成してBCD形式でカレンダのレジスタ（RTC_TRおよびRTC_DR）を更新することができます。

RTCがバイナリモードに設定されている場合（RTC_ICSRレジスタでBIN = 01）: RTC_SSRレジスタにはバイナリカウンタSS[31:0] があり、ck_apreによってクロック供給されます。BCD形式のカレンダのレジスタ（RTC_TRおよびRTC_DR）は使用されません。

RTCがバイナリモードに設定されている場合（RTC_ICSRレジスタでBIN = 10または11）: RTC_SSRレジスタにはバイナリカウンタSS[31:0] があり、ck_apreによってクロック供給されます。カレンダは、SSR[BCDU+7:0] が0に達するたびに更新されます（1秒のインクリメント）。

RTCレジスタ書込み保護

システムリセット後、電源制御ペリフェラルの DBP ビットによって、RTCレジスタは不要な書込みアクセスから保護されます（PWR電源制御のセクションを参照）。RTCレジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

バックアップドメインリセット後、次の一部のRTCレジスタは書込み保護されます。RTC_TR、RTC_DR、RTC_PRER、RTC_CALR、RTC_SHIFTR、ビットINIT、BIN、BCDU（RTC_ICSR）、およびビットFMT、SUB1H、ADD1H、REFCKON（RTC_CR）。

保護されたRTCレジスタの書込み保護を解除するには、次のステップが必要です。

1. RTC_WPRレジスタに0xCAを書き込みます。
2. RTC_WPRレジスタに0x53を書き込みます。

誤ったキーを書き込むと、書込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

INITPRIVによって保護されたレジスタは、INIT KEYで書込み保護されます。

CALDPRIVによって保護されたレジスタは、CAL KEYで書込み保護されます。

PRIV または INITPRIV が RTC_PRIVCFGR でセットされている場合、SEC または INITSEC が RTC_SECCFGRでセットされている場合：RTC_WPRレジスタへの書込みアクセスが特権モードおよびセキュリティモード（PRIV、INITPRIV、SEC、INITSEC の設定で定義）で実行される場合のみ、INIT KEYはロック解除またはロックされます。

PRIV またはCALPRIV がRTC_PRIVCFGR でセットされている場合、またはSEC やCALSEC がRTC_SECCFGRでセットされている場合：RTC_WPRレジスタへの書込みアクセスが特権モードおよびセキュリティモード（PRIV、CALPRIV、SEC、CALSEC の設定で定義）で実行される場合のみ、CAL KEYはロック解除またはロックされます。

カレンダの初期化と設定

時間形式やプリスケーラ設定を含むカレンダ時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ICSRレジスタでINITビットを1にセットして、初期化モードに入ります。このモードでは、カレンダカウンタが停止し、その値を更新することができます。
2. RTC_ICSR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。
   - LPCAL = 0の場合：INITFは、INITビットがセットされた後、およそ2 RTCCLKサイクルでセットされます。
   - LPCAL = 1の場合：INITFは、INITビットがセットされた後、最大2 ck_apreサイクルでセットされます。
3. カレンダカウンタのための1 Hzクロックを生成するには、RTC_PRERレジスタで両方のプリスケーラ分周比をプログラムし、RTC_ICSRレジスタのBINおよびBCDU を設定します。
4. シャドウレジスタ（RTC_TRおよびRTC_DR）に時刻と日付の初期値をロードし、RTC_CRレジスタのFMTビットを介して時間形式（12時間または24時間）を設定します。
5. INITビットをクリアして初期化モードを終了します。その後、カレンダカウンタの実際の値が自動的にロードされます。
   - LPCAL=0の場合：4 RTCCLKクロックサイクル後にカウントが再開します。
   - LPCAL=1の場合：最大2 RTCCLK + 1 ck_apre後にカウントが再開します。

初期化シーケンスが完了すると、カレンダがカウントを開始します。RTC_SSRの内容は次のように初期化します。

- PREDIV_SはBCDモード（BIN=00）
- バイナリで0xFFFF FFFFまたは混合（BCD-binary）モード（BIN=01、10または11）

BCDモードでは、RTC_SSRには、同期プリスケーラのカウンタの値が入っています。これにより、RTCによって維持されている正確な時刻を1/(PREDIV_S + 1) 秒の分解能まで計算することができます。その結果、同期プリスケーラ値（PREDIV_S[14:0]）を増加させることにより分解能を改善できます。許可されている最大分解能（32768Hzクロックで30.52µs）は、PREDIV_Sを0x7FFFにセットすることにより得られます。

ただし、PREDIV_Sを増加させるということは、同期プリスケーラの出力を1 Hzに維持するためPREDIV_Aを減らす必要があることを意味します。このように、非同期プリスケーラの出力周波数が増加すると、RTCの動的消費電力が増加する場合があります。RTCの動的消費電力は、2の累乗の数になるPREDIV_A+1に最適化されます。

注： **システムリセット後、アプリケーションはRTC_ICSR レジスタのINITS フラグを読み出し、カレンダが初期化されたか否かを確認できるようになります。このフラグが0であれば、カレンダの年の項目がバックアップドメインリセットのデフォルト値（0x00）にセットされているため、初期化されていません。**

注： **初期化後にカレンダを読み出すには、まずソフトウェアでRTC_ICSRレジスタのRSFフラグがセットされていることを確認する必要があります。**

サマータイム

サマータイム管理は、RTC_CRレジスタのSUB1Hビット、ADD1Hビット、BKPビットを介して行われます。

SUB1HまたはADD1Hを使用すると、ソフトウェアは初期化手順を踏まずに1度の操作で、カレンダから1時間引いたり足したりすることができます。

さらに、ソフトウェアはBKPビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下にアラームAの手順を示していますが、アラームBにも転用できます。

1. RTC_CRのALRAEをクリアしてアラームAを無効にします。
2. アラームAレジスタ（RTC_ALRMASSR/RTC_ALRMAR またはRTC_ALRABINR）をプログラムします。
3. RTC_CRレジスタでALRAEをセットしてアラームAを再び有効にします。

注： **RTC_CRレジスタの各変更は、クロック同期のためRTCCLKクロック約2サイクル後に有効になります。**

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値（RTC_WUTRのWUT[15:0]）の設定または変更には、次の手順が必要です。

1. RTC_CRのWUTEをクリアしてウェイクアップタイマを無効にします。
2. RTC_ICSRのWUTWFがセットされ、ウェイクアップ自動再ロードカウンタおよびWUCKSEL[2:0]ビットへのアクセスが許可されていることが確認されるまでWUTWFをポーリングします。このステップは、カレンダ初期化モードではスキップする必要があります。
   - WUCKSEL[2] = 0の場合：WUTWFは、WUTEビットがクリアされた後、およそ1 ck_wut + 1 RTCCLKサイクルでセットされます。
   - WUCKSEL[2] = 1の場合：WUTWFは、WUTEビットがクリアされた後、最大1 ck_apre + 1 RTCCLKサイクルでセットされます。
3. ウェイクアップ自動再ロード値WUT[15:0]、WUTOCLR[15:0] およびウェイクアップクロック選択（RTC_CRのWUCKSEL[2:0] ビット）をプログラムします。RTC_CRでWUTEをセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウンを再開します。
   - WUCKSEL[2] = 0の場合：WUTWFは、WUTEビットがセットされた後、およそ1 ck_wut + 1 RTCCLKサイクルでクリアされます。
   - WUCKSEL[2] = 1の場合：WUTWFは、WUTEビットがクリアされた後、最大1 ck_apre + 1 RTCCLKサイクルでセットされます。

63.3.12 カレンダの読出し

RTC_CRレジスタのBYPSHAD 制御ビットがクリアされている場合

RTCカレンダレジスタ（RTC_SSR、RTC_TR、およびRTC_DR）を正しく読み出すには、APBクロック周波数（f_{PCLK}）がRTCクロック周波数（f_{RTCCLK}）の7倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APBクロック周波数がRTCクロック周波数の7倍未満である場合、ソフトウェアによってカレンダ時間と日付レジスタを2回読み出す必要があります。RTC_TRの2回目の読出しが1回目の読出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3回目の読出しアクセスを行う必要があります。どの場合も、APBクロック周波数は必ずRTCクロック周波数以上でなければなりません。

RTC_ICSRレジスタのRSFビットは、カレンダレジスタがRTC_SSR、RTC_TR、およびRTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーは、RTCCLKサイクルごと行われます。3つの値における一貫性を保証するため、RTC_SSRまたはRTC_TRのどちらかを読み出すと、高次カレンダシャドウレジスタの値はRTC_DRが読み出されるまでロックされます。ソフトウェアが1 RTCCLK周期未満の間隔でカレンダの読出しアクセスを行う場合、最初のカレンダ読出し後にRSFをソフトウェアでクリアする必要があり、その後ソフトウェアは、RSFがセットされるまで待ってから、RTC_SSR、RTC_TR、およびRTC_DR レジスタを再度読み出す必要があります。

低消費電力モード（STOPまたはSTANDBY）からのウェイクアップ後は、RSFをソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度RSFがセットされるまで待ってから、RTC_SSR、RTC_TR、およびRTC_DR レジスタを再度読み出す必要があります。

RSFビットは、ウェイクアップ後にクリアする必要がありますが、低消費電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアはRSFがセットされるまで待ってから、RTC_SSR、RTC_TR、およびRTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化（2574ページのカレンダの初期化と設定を参照）後、ソフトウェアはRSFがセットされるまで待ってから、RTC_SSR、RTC_TR、およびRTC_DR レジスタを読み出す必要があります。

同期（セクション 63.3.14 : RTCの同期を参照）後、ソフトウェアはRSFがセットされるまで待ってから、RTC_SSR、RTC_TR、およびRTC_DR レジスタを読み出す必要があります。

RTC_CRレジスタ（バイパスシャドウレジスタ）のBYPSHAD制御ビットがセットされている場合

カレンダレジスタを読み出すと、カレンダカウンタの値が直接与えられるため、RSFビットがセットされるのを待つ必要がありません。シャドウレジスタは低消費電力モード（STOPまたはSTANDBY）では更新されないため、低消費電力モード終了後に特にこのような読出しが有用です。

BYPSHADビットが1にセットされている場合、レジスタへの2回の読出しアクセス間でRTCCLKエッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読出し操作中にRTCCLKエッジが発生した場合、レジスタの1つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを2回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わり、ソフトウェアはカレンダレジスタの最下位の数値を2回比較するだけで構いません。

注： **BYPSHAD = 1の間、カレンダレジスタの読出し命令が完了するには1 APBサイクルだけ余計に必要となります。**

63.3.13 RTCのリセット

カレンダシャドウレジスタ（RTC_SSR、RTC_TR、RTC_DR）およびRTCステータスレジスタ（RTC_ICSR）の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタはバックアップドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTCの現在のカレンダレジスタ、RTC制御レジスタ（RTC_CR）、プリスケーラレジスタ（RTC_PRER）、RTC較正レジスタ（RTC_CALR）、RTCシフトレジスタ（RTC_SHIFTR）、RTCタイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、およびRTC_TSDR）、ウェイクアップタイマレジスタ（RTC_WUTR）、およびアラームAおよびアラームBレジスタ（RTC_ALRMASSR/RTC_ALRMAR/RTC_ALRABINR および RTC_ALRMBSSR/RTC_ALRMBR/RTC_ALRBBINR）。

さらに、LSEクロックによって駆動されている際にリセットソースがバックアップドメインリセットと異なる場合、システムリセットがかかってもRTCは動作を続けます（システムリセットの影響を受けないRTCクロックソースの詳細については、RCCを参照してください）。バックアップドメインリセットが発生すると、RTCは停止し、すべてのRTCレジスタがリセット値にセットされます。

63.3.14 RTCの同期

RTCは、高精度でリモートクロックと同期できます。サブセカンドフィールド（RTC_SSRまたはRTC_TSSSR）を読み出した後、リモートクロックとRTCによって維持されている時間の正確なオフセットを計算できます。その後、RTC_SHIFTRを使用してほんの一瞬クロックを「シフト」することによってRTCを調整し、このオフセットを取り除くことができます。

RTCは、RTCシフト制御レジスタ（RTC_SHIFTR）を使って微調整できます。RTC_SHIFTRに書き込むことにより、1 ck_apre周期の分解能でシフト（遅れ／進み）させることができます。

このシフト操作の本質は、同期プリスケーラのカウンタSS[15:0] にSUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。

同時に BCDまたは混合モードで ADD1Sビットがセットされた場合、1秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。バイナリモードではADD1Sは影響を及ぼしません。

RTC_SHIFTRレジスタへの書込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示すSHPFフラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： **混合モードでは（BIN=10または11）、SUBFS[14:BCDU+8] には0を書き込む必要があります。**

注意： **BCDモードでシフト操作を始める前に、ユーザはSS[15] = 0であることを確認し、オーバーフローが発生しないようにする必要があります。混合モードでは、SS[BCDU+8] のビットが0であることを確認する必要があります。**

注意： **この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1のときにファームウェアからRTC_SHIFTRへの書込みはできません。**

63.3.15 RTCリファレンスクロック検出

この機能は、BCDモード（BIN=00）でのみ利用できます。

RTCカレンダの更新は、リファレンスクロックであるRTC_REFINに同期させることができます。通常は商用電源（50または60Hz）です。RTC_REFINリファレンスクロックには、32.768kHz LSEクロックより高い精度が必要です。RTC_REFIN検出を有効にした際（RTC_CRのREFCKONビットが1にセット）、カレンダは引き続きLSEクロックによって駆動され、RTC_REFINはカレンダ更新周波数（1Hz）の誤差の補正に使用されます。

各1Hzクロックエッジは、一番近いRTC_REFINクロックエッジ（所与の時間枠内に見つかった場合）と比較されます。ほとんどの場合、2つのクロックエッジは正しく整列しています。LSEクロックが不正確なために1Hzのクロックがずれた場合、RTCは1Hzのクロックを少しシフトさせ、その後の1Hzのクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTCは、32.768kHzクォーツから生成される256Hzクロック（ck_apre）を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新（1秒ごと）程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠はck_apre 7周期に等しくなります。その後のカレンダ更新では、ck_apre 3周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_spreクロックを出力する非同期プリスケーラは強制的に再ロードされます。プリスケーラは同時に再ロードされるので、リファレンスクロックおよび1Hzのクロックが整列するタイミングには影響しません。クロックが整列していない場合、後の1Hzクロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した（ck_apre 3周期の枠内でリファレンスクロックエッジが発生しない）場合、カレンダはLSEクロックのみを基準にして更新が継続されます。その後RTCはck_spreエッジを中心としてck_apre 7周期という広い検出時間枠でリファレンスクロックを待ちます。

RTC_REFIN検出を有効にした場合、PREDIV_AおよびPREDIV_Sを以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREVID_S = 0x00FF

注： **RTC_REFINクロック検出は、STANDBYモードでは利用できません。**

63.3.16 RTCの高精度デジタル較正

RTC周波数の精度は、-487.1 ppmから +488.5 ppmの範囲で、分解能約0.954 ppmでデジタル的に較正できます。周波数の修正は、一連の微調整（個々のck_calパルスの追加や削除）によって行われます。

LPCAL=0の場合：ck_cal = RTCCLK

LPCAL=1の場合：ck_cal = ck_apre

このような調整は、短い期間で観測された場合でもRTCが十分に較正されるように、かなり広範に分散して行われます。

RTC超低消費電力モード

RTCの消費電力は、RTC較正レジスタ（RTC_CALR）のLPCALビットをセットすることで低減できます。この場合、較正メカニズムはRTCCLKではなくck_apreで適用されます。結果となる精度は同じですが、LPCAL = 0の場合は2^{20} RTCCLKパルスではなく、およそ2^{20} x PREDIV_A x RTCCLKパルスの較正サイクルの間に較正が実行されます。

高精度較正メカニズム

高精度較正レジスタ（RTC_CALR）によって較正サイクル中にマスクされるck_calクロックサイクル数を指定します。

- CALM[0] ビットを1にセットすると較正サイクルの中でちょうど1パルスがマスクされます。
- CALM[1] ビットを1にセットすると、さらに2サイクルがマスクされます。
- CALM[2] ビットを1にセットすると、さらに4サイクルがマスクされます。
- CALM[8] ビットを1にセットするまで続けると、256クロックがマスクされます。

注： **CALM[8:0]（RTC_CALR）によって較正サイクル中にマスクされるck_calパルス数を指定します。CALM[0] ビットを1にセットすると、cal_cnt[19:0] = 0x80000になった時点で、較正サイクル中でちょうど1パルスがマスクされます。CALM[1] = 1では、さらに2サイクルがマスクされ（cal_cnt = 0x40000および0xC0000）、CALM[2] = 1では、さらに4サイクルがマスクされ（cal_cnt = 0x20000/0x60000/ 0xA0000/ 0xE0000）、CALM[8] = 1まで続けると、256クロックがマスクされます（cal_cnt = 0xXX800）。**

CALMでは、細かい分解能でRTC周波数を最大487.1 ppm 負の方向に調整することができ、CALPビットでは周波数を488.5 ppm正の方向に調整することができます。CALP を1にセットすることにより、実質上は、ck_cal 2^{11} サイクルごとに、ck_calパルスが1パルス追加で挿入されます。すなわち較正サイクルごとに512クロックが追加されることとなります。

CALMをCALPと合わせて使用すると較正サイクルの間にck_cal -511から +512 ck_calサイクルまでのオフセットが追加でき、これは約0.954 ppmの分解能で較正範囲 -487.1～+488.5 ppmに換算されます。

有効較正周波数（FCAL）を入力周波数（FRTCCLK）に対して求める計算式は次のとおりです。

$$F_{CAL} = F_{RTCCLK} \times [1 + (CALP \times 512 - CALM) / (2^{20} + CALM - CALP \times 512)]$$

注意： **PREDIV_Aは3以上でなければなりません。**

PREDIV_A < 3の場合の較正

非同期プリスケーラ値（RTC_PRERレジスタのPREDIV_Aビット）が3未満の場合、CALPビットを1にセットすることはできません。CALPがすでに1にセットされていて、PREDIV_Aビットが3未満の値にセットされた場合、CALPの設定値は無視され、CALPが0に設定された場合と同じように較正されます。

BCDモードでは PREDIV_Aが3未満の状態で較正できますが、各秒のカウントが8 ck_calクロックサイクル分早められるように同期プリスケーラ値（PREDIV_S）を小さくする必要があります。これは、較正サイクルごとに256クロックサイクル追加することに相当します。結果として、CALMビットのみを使用して較正周期の間に255～256クロックパルス（243.3～244.1 ppmの較正範囲に相当）を追加することができます。

公称RTCCLK周波数が32768 Hzで、PREDIV_Aが1（分周比2）の場合、PREDIV_Sを16383ではなく16379（4少ない）にセットする必要があります。また、PREDIV_Aが0の場合、PREDIV_Sを32767ではなく32759（8少ない）にセットする必要がありますので注意してください。

PREDIV_Sをこのように減少させた場合、較正された入力クロックの有効周波数の式は次のようになります。

$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$

この場合、RTCCLKが正確に32768.00 Hzであれば、CALM[7:0] が0x100（CALM設定範囲の中間値）と等しくなるのが正しい設定です。

RTC較正値の確認

32秒の較正サイクルにするために、LPCAL = 0でRTC較正を検証することを推奨します。

RTCの精度は、RTCCLKの正確な周波数を測定し、正しいCALM値およびCALP値を計算することにより保証されます。オプションの1 Hz出力が搭載されており、アプリケーションによってRTC精度の測定と確認を行うことができます。

ある時間間隔でRTCの周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大2 RTCCLKクロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は32秒です。

  このモードを使用して正確に32秒で1Hz出力の精度を測定すると、その精度は0.477ppm（較正分解能の制限により32秒で0.5 RTCCLKサイクル）以内となることが保証されます。
- RTC_CALRレジスタのCALW16bitを1にセットして、較正サイクル周期を強制的に16秒にすることができます。

  この場合、RTC精度は最大誤差0.954 ppm（16秒で0.5 RTCCLKサイクル）で16秒間で測定できます。ただし、較正分解能が下がるため、長期的なRTC精度もまた0.954 ppmに下がります。CALW16が1にセットされると、CALM[0] ビットは0のままとなります。
- RTC_CALRレジスタのCALW8bitを1にセットして、較正サイクル周期を強制的に8秒にすることができます。

  この場合、RTC精度は最大誤差1.907 ppm（8秒で0.5 RTCCLKサイクル）で8秒間で測定できます。長期的なRTC精度も1.907 ppmに下がります。CALW8が1にセットされると、CALM[1:0] ビットは00のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ICSR/INITF = 0の間でも、較正レジスタ（RTC_CALR）を動作中に更新することができます。

1. RTC_ICSR/RECALPF（再較正保留フラグ）をポーリングします。
2. このフラグが0にセットされている場合は、必要に応じて新しい値をRTC_CALRに書き込みます。すると、RECALPFが自動的に1にセットされます。
3. RTC_CALRへの書込み動作後ck_apre 3サイクル以内に、新しい較正設定が有効になります。

63.3.17 タイムスタンプ機能

タイムスタンプは、RTC_CRレジスタのTSEビットまたはITSE ビットを1にセットすることにより有効になります。

TSEがセットされている場合：

RTC_TSピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

TAMPTS がセットされている場合：

内部または外部タンパイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。RTC制御レジスタ（RTC_CR）およびセクション ：タンパイベント時のタイムスタンプを参照してください。

ITSEがセットされている場合：

内部タイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。内部タイムスタンプイベントはV_{BAT}電源への切り替えによって生成されます。

タイムスタンプイベントが発生すると、内部または外部イベントによって、RTC_SRレジスタのタイムスタンプフラグビット（TSF）がセットされます。イベントが内部の場合、RTC_SRレジスタのITSFフラグもセットされます。

RTC_CRレジスタのTSIEビットをセットすることにより、タイムスタンプイベントが発生したときに割込みが生成されます。

タイムスタンプフラグ（TSF）がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ（TSOVF）がセットされ、タイムスタンプレジスタ（RTC_TSTRおよびRTC_TSDR）は、その前のイベントの結果を維持します。

注： **同期プロセスにより、RTC_TSピンまたはrtc_its内部信号からのタイムスタンプイベントが発生してから、最大でck_apreの2サイクル後に、TSFがセットされます。TSFは、タンパフラグから最大でck_apreの3サイクル後にセットされます。**

TSOVFは、イベントが発生してから最大でck_apreの1サイクル後にセットされます。これは、2つのタイムスタンプイベントの発生したタイミングが近い場合、TSFがまだ“0”であってもTSOVFが“1”と検出される可能性があることを意味します。よって、TSOVFのポーリングはTSFがセットされた後に実施することをお勧めします。

注意： **TSFビットのクリア処理を行った直後にタイムスタンプイベントが発生した場合、TSFビットおよびTSOVFビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、アプリケーションはTSFがすでに1と読み出されていない限り、0クリアの処理を行ってはなりません。**

63.3.18 較正クロック出力

RTC_CRレジスタでCOEビットが1にセットされると、CALIBデバイス出力にリファレンスクロックが供給されます。

RTC_CRレジスタのCOSELビットがリセットされ、かつPREDIV_A = 0x7Fである場合、CALIB周波数は$f_{RTCCLK/64}$ です。これは32.768 kHzのRTCCLK周波数に対する512 Hzの較正出力に相当します。立下りエッジには軽いジッタがあるため、CALIBのデューティサイクルは不規則になります。したがって、立上りエッジの使用が推奨されます。

COSELがセットされ、「PREDIV_S+1」が0以外の256の倍数である場合（PREDIV_S[7:0] = 0xFFなど）、CALIB周波数はfRTCCLK/（256 *（PREDIV_A+1））となります。これは、RTCCLK周波数が32.768kHzで、プリスケーラデフォルト値（PREDIV_A = Ox7F、PREDIV_S = 0xFF）に対する1Hzの較正出力に相当します。

注： **COSELがクリアされると、CALIB出力は非同期プリスケーラの6番目のステージの出力になります。LPCALを0から1に変更する場合、LPCAL切り替え中は出力が不規則（グリッチ）になる可能性があります。LPCAL = 1の場合、この出力は常に使用できます。LPCAL = 0の場合、PREDIV_Aが0x20未満のとき出力はありません。**

COSELがセットされると、CALIB出力は同期プリスケーラの8番目のステージの出力になります。

63.3.19 タンパおよびアラーム出力

RTC_CRレジスタのOSEL[1:0] 制御ビットを使用してアラーム出力TAMPALRMを有効にし、出力となる機能を選択します。これらの機能は、RTC_SRレジスタの該当するフラグの内容を反映します。

TAMPOE制御ビットがRTC_CRでセットされている場合、すべての外部および内部のタンパフラグは論理和がとられ、TAMPALRM出力に送られます。OSEL = 00の場合、TAMPALRM出力はタンパフラグのみ反映します。OSEL ≠ 00の場合、TAMPALRMの信号でタンパフラグと、アラームA、B、ウェイクアップフラグのいずれかがともに供給されます。

RTC_CRのPOL制御ビットは、POLが1にセットされているときに選択されたフラグビットの逆が出力されるよう、TAMPALRM出力の極性を決定します。

TAMPALRM出力

TAMPALRMピンは、制御ビットTAMPALRM_TYPE（RTC_CRレジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。TAMPALRM_PU（RTC_CR）によって、出力モードで内部プルアップを適用できます。

注： **TAMPALRM出力が有効になると、RTC_OUT1でCALIBより優先順位が高くなります。**

TAMPALRMがRTCでオープンドレインに設定されている場合、RTC_OUT1 GPIOを入力として設定する必要があります。

63.4 RTC低消費電力モード

表630.低消費電力モードがRTCに与える影響

モード	説明
SLEEP	影響なし。 RTC割込みによって、デバイスはSLEEPモードから復帰します。
STOP	RTCクロックソースがLSEまたはLSIの場合、RTCはアクティブのままです。RTC割込みによって、デバイスはSTOPモードを終了します。
STANDBY	RTCクロックソースがLSEまたはLSIの場合、RTCはアクティブのままです。RTC割込みによって、デバイスはSTANDBYモードを終了します。
SHUTDOWN	RTCクロックソースがLSEの場合、RTCはアクティブのままです。RTC割込みによって、デバイスはSHUTDOWNモードを終了します。

次の表は、すべてのモードでのRTCピンと機能の一覧です。

表631.RTCピン機能（モード共通）

機能	STOP 3モード、STANDBYモードおよびSHUTDOWNモード以外のすべての低電力モードで使用可能な機能	STOP 3モード、STANDBYモードおよびSHUTDOWNモードで使用可能な機能	V_{BAT} モードで使用可能な機能
RTC_TS	可能	あり	はい
RTC_REFIN	可能	不可	なし
RTC_OUT1	可能	あり	はい
RTC_OUT2	可能	あり	はい

63.5 RTC割込み

割込みチャネルは、マスク済み割込みステータスレジスタ、またはセキュリティモード設定に応じたセキュアマスク済み割込みステータスレジスタで設定されます。また、非セキュア割込み出力やセキュア割込み出力も有効化されます。

表632.非セキュア割込みリクエスト

項目（割込みの略称）	割込みイベント	イベント・フラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	低消費電力モードの終了
RTC	アラームA	ALRAF	ALRAIE および（ALRASEC=0かつ SEC=0）	CALRAFに1を書き込む	あり(3)
	アラームB	ALRBF	ALRBIE および（ALRBSEC=0かつ SEC=0）	CALRBFに1を書き込む	あり (3)
	タイムスタンプ	TSF	TSIE および（TSSEC=0かつ SEC=0）	CTSFに1を書き込む	あり (3)
	ウェイクアップ・タイマ	WUTF	WUTIE および（WUTSEC=0かつ SEC=0）	CWUTFに1を書き込む	あり (3)
	SSRアンダーフロー	SSRUF	SSRUIE および（ALRASEC=0かつ SEC=0）	CSSRUFに1を書き込む	あり (3)

1. イベントフラグは、RTC_SRレジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、RTC_MISRレジスタ内です。
3. RTCが低消費電力モードでオシレータ機能によってクロック供給される場合。

表633.セキュア割込みリクエスト

項目（割込みの略称）	割込みイベント	イベント・フラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	低消費電力モードの終了
RTC_S	アラームA	ALRAF	ALRAIEおよび（ALRASEC=1または SEC=1）	CALRAFに1を書き込む	あり(3)
	アラームB	ALRBF	ALRBIEおよび（ALRBSEC=1または SEC=1）	CALRBFに1を書き込む	あり (3)
	タイムスタンプ	TSF	TSIEおよび（TSSEC=1または SEC=1）	CTSFに1を書き込む	あり (3)
	ウェイクアップ・タイマ	WUTF	WUTIEおよび（WUTSEC=1または SEC=1）	CWUTFに1を書き込む	あり (3)
	SSRアンダーフロー	SSRUF	SSRUIEおよび（ALRASEC=1または SEC=1）	CSSRUFに1を書き込む	あり (3)

1. イベントフラグは、RTC_SRレジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、RTC_SMISRレジスタ内です。
3. RTCが低消費電力モードでオシレータ機能によってクロック供給される場合。

63.6 RTCレジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルのセクション1.2を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスすることができます。

63.6.1 RTC時刻レジスタ（RTC_TR）

RTC_TRは、カレンダ時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。2574ページのカレンダの初期化と設定 および 2575ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは、非セキュアアクセスから書込み保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから書込み保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x00

バックアップドメインリセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0の場合。BYPSHAD = 1の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **PM**: AM/PM表記
0：AMまたは24時間形式
1：PM

ビット21:20 **HT[1:0]**：BCD形式での時の十の位

ビット19:16 **HU[3:0]**：BCD形式での時の一の位

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **MNT[2:0]**：BCD形式での分の十の位

ビット11:8 **MNU[3:0]**：BCD形式での分の一の位

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **ST[2:0]**：BCD形式での秒の十の位

ビット3:0 **SU[3:0]**：BCD形式での秒の一の位

63.6.2 RTC日付レジスタ（RTC_DR）

RTC_DRは、カレンダ日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。2574ページのカレンダの初期化と設定および2575ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは、非セキュアアクセスから書込み保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから書込み保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x04

バックアップドメインリセット値：0x0000 2101

システムリセット値：0x0000 2101（BYPSHAD = 0の場合。BYPSHAD = 1の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:20 **YT[3:0]**：BCD形式での年の十の位

ビット19:16 **YU[3:0]**：BCD形式での年の一の位

ビット15:13 **WDU[2:0]**：曜日

000：禁止
001：月曜日
.....
111：日曜日

ビット12 **MT**：BCD形式での月の十の位

ビット11:8 **MU[3:0]**：BCD形式での月の一の位

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **DT[1:0]**：BCD形式での日の十の位

ビット3:0 **DU[3:0]**：BCD形式での日の一の位

注： **カレンダは最大値に達すると停止し、ロールオーバーできません。**

63.6.3 RTCサブセカンドレジスタ（RTC_SSR）

アドレスオフセット：0x08

バックアップドメインリセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0の場合。BYPSHAD = 1の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SS[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **SS[31:0]**：同期バイナリカウンタ

SS[31:16]：同期バイナリカウンタのMSB値
バイナリモードまたは混合モードが選択されている場合（BIN = 01、10、11のいずれか）：
SS[31:16] は、SS[31:0] フリーランニングダウンカウンタの16 MSBです。
BCDモードが選択されている場合（BIN=00）：
SS[31:16] はハードウェアによって0x0000に固定されています。
SS[15:0]：サブセカンド値/同期バイナリカウンタのLSB値
バイナリモードが選択されている場合（BIN = 01、10、11のいずれか）：
SS[15:0] は、SS[31:0] フリーランニングダウンカウンタの16 MSBです。
BCDモードが選択されている場合（BIN=00）：
SS[15:0] は、同期プリスケーラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。
秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)
SSは、シフト操作後に限り、PREDIV_Sより大きな値となる場合があります。この場合、正確な時刻/日付は、RTC_TR/RTC_DRで示される値よりも1秒少ない値となります。

63.6.4 RTC初期化制御およびステータスレジスタ（RTC_ICSR）

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非セキュアアクセスから個別に保護することもできます。セクション 63.3.4：RTCセキュア保護モードを参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x0C

バックアップドメインリセット値：0x0000 0007

システムリセット：0にクリアされるINIT、INITF、RSFビット以外は影響されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RECALPF
															r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	BCDU[2:0]			BIN[1:0]		INIT	INITF	RSF	INITS	SHPF	WUTWF	Res.	Res.
			rw	rw	rw	rw	rw	rw	r	rc_w0	r	r	r		

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RECALPF**：再較正保留フラグ

ソフトウェアによってRTC_CALRレジスタに書込みが行われると、RECALPFステータスフラグが自動的に1にセットされ、RTC_CALRレジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは0に戻ります。動作中の再較正を参照してください。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:10 **BCDU[2:0]**：BCD更新（BIN = 10または11）

BCDカレンダおよびバイナリ拡張カウンタを使用した混合モード（BIN = 10または11）では、カレンダの秒はSSR最下位ビットでインクリメントされます。

0x0：SS[7:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x1：SS[8:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x2：SS[9:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x3：SS[10:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x4：SS[11:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x5：SS[12:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x6：SS[13:0] = 0のたびに1秒のカレンダインクリメントが生成されます。
0x7：SS[14:0] = 0のたびに1秒のカレンダインクリメントが生成されます。

ビット9:8 **BIN[1:0]**：バイナリモード

00：フリーランニングBCDカレンダモード（バイナリモード無効）
01：フリーランニングバイナリモード（BCDモード無効）
10：フリーランニングBCDカレンダモードおよびバイナリモード
11：フリーランニングBCDカレンダモードおよびバイナリモード

ビット7 **INIT**：初期化モード

0：フリーランニングモード
1：時刻と日付レジスタ（RTC_TRとRTC_DR）、およびプリスケーラレジスタ（RTC_PRER）に加え、BINおよびBCDUフィールドのプログラムに使用する初期化モードです。INITがリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット6 **INITF**：初期化フラグ

このビットが1にセットされると、RTCは初期化状態となり、時刻、日付およびプリスケーラレジスタが更新できます。

0：カレンダレジスタを更新できません。

1：カレンダレジスタを更新できます。

ビット5 **RSF**：レジスタ同期フラグ

このビットは、カレンダレジスタがシャドウレジスタ（RTC_SSR、RTC_TRおよびRTC_DR）にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中（SHPF = 1）の初期化モードで、またはバイパスシャドウレジスタモード（BYPSHAD = 1）で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0：カレンダシャドウレジスタはまだ同期していません。

1：カレンダシャドウレジスタは同期しています。

ビット4 **INITS**：初期化ステータスフラグ

このビットは、カレンダの年の項目が0ではないとき（バックアップドメインリセット状態）にハードウェアによってセットされます。

0：カレンダは初期化されていません。

1：カレンダは初期化されています。

ビット3 **SHPF**：シフト操作保留

このフラグは、RTC_SHIFTRへの書込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPFビットに書き込んでも影響はありません。

0：保留中のシフト操作はありません。

1：保留中のシフト操作があります。

ビット2 **WUTWF**：ウェイクアップタイマ書込みフラグ

このビットは、RTC_CRでWUTEビットが0にセットされた後、WUT値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：初期化モード以外で、ウェイクアップタイマ設定は更新できません。

1：ウェイクアップタイマ設定が更新できます。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

63.6.5 RTCプリスケーラレジスタ（RTC_PRER）

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2回の書込みアクセスに分けて行う必要があります。2574ページのカレンダの初期化と設定を参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは、非セキュアアクセスから書込み保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから書込み保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x10

バックアップドメインリセット値：0x007F：00FF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PREDIV_S[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **PREDIV_A[6:0]**：非同期プリスケーラ分周比

非同期分周比です。

ck_apre周波数 = RTCCLK周波数／(PREDIV_A+1)

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **PREDIV_S[14:0]**：同期プリスケーラ分周比

同期分周比です。

ck_spre周波数 = ck_apre周波数／(PREDIV_S+1)

63.6.6 RTCウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ICSRのWUTWFが1にセットされているときにのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x14

バックアップドメインリセット値：0x0000 FFFF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUTOCLR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **WUTOCLR[15:0]**：ウェイクアップ自動再ロード出力クリア値

WUTOCLR[15:0] が0x0000ではない場合、WUTFは自動再ロードダウンカウンタが0に達するとハードウェアによってセットされ、自動再ロードダウンカウンタがWUTOCLR[15:0] に達するとハードウェアによってクリアされます。

WUTOCLR[15:0] = 0x0000の場合、WUTFはWUTダウンカウンタが0に達するとハードウェアによってセットされ、

ソフトウェアによってクリアされます。

ビット15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTEが1にセット）なとき、ck_wutの（WUT[15:0] + 1）サイクルごとにWUTFフラグがセットされます。ck_wutの周期は、RTC_CRレジスタのWUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1のとき、ウェイクアップタイマは17bitとなり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットであるWUT[16] となります。

WUTFの最初のアサートは、WUTEがセットされてからWUTとck_wutの（WUT + 2）サイクルの間に発生します。WUCKSEL[2:0] = 011 (RTCCLK/2) のときにWUT[15:0] を0x0000にセットすることはできません。

63.6.7 RTC制御レジスタ（RTC_CR）

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非セキュアアクセスから個別に保護することもできます。セクション 63.3.4：RTCセキュア保護モードを参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x18

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	ALRBFCLR	ALRAFCLR	TAMPOE	TAMPTS	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	SSRUIE	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **OUT2EN**：RTC_OUT2出力イネーブル

このビットがセットされている場合、RTC_OUT2で次のようにRTC出力を再配置できます。

OUT2EN = 0: RTC出力2無効

OSEL ≠ 00またはTAMPOE = 1の場合：RTC_OUT1にTAMPALRMが出力されます。

OSEL = 00およびTAMPOE = 0およびCOE = 1の場合：RTC_OUT1にCALIBが出力されます。

OUT2EN = 1: RTC出力2有効

（OSEL ≠ 00またはTAMPOE = 1）およびCOE = 0の場合：RTC_OUT2にTAMPALRMが出力されます。

OSEL = 00およびTAMPOE = 0およびCOE = 1の場合：RTC_OUT2にCALIBが出力されます。

（OSEL ≠ 00またはTAMPOE = 1）およびCOE = 1の場合：RTC_OUT2にCALIBが出力され、RTC_OUT1にTAMPALRMが出力されます。

ビット30 **TAMPALRM_TYPE**：TAMPALRM出力形式

0：TAMPALRMはプッシュプル出力になります。

1：TAMPALRMはオープンドレイン出力になります。

ビット29 **TAMPALRM_PU**：TAMPALRMプルアップイネーブル

0：TAMPALRM出力でプルアップは適用されません。

1：TAMPALRM出力でプルアップは適用されます。

ビット28 **ALRBFCLR**：アラームBフラグ自動クリア

0：アラームBイベントでトリガイベントが生成され、次のアラームイベントを許可するためにALRBFをソフトウェアでクリアする必要があります。

1：アラームBイベントによってトリガイベントが生成されます。ALRBFはck_apreの1サイクル後にハードウェアにより自動的にクリアされます。

ビット27 **ALRAFCLR**：アラームAフラグ自動クリア

0：アラームAイベントでトリガイベントが生成され、次のアラームイベントを許可するためにALRAFをソフトウェアでクリアする必要があります。

1：アラームAイベントによってトリガイベントが生成されます。ALRAFはck_apreの1サイクル後にハードウェアにより自動的にクリアされます。

ビット26 **TAMPOE**：TAMPALRMでのタンパ検出出力イネーブル

0：タンパフラグがTAMPALRMで送られません。

1：タンパフラグがOSELによる信号およびPOLによる極性とともにTAMPALRMで送られます。

ビット25 **TAMPTS**：タンパ検出イベント時のタイムスタンプの有効化

0：タンパ検出イベントがあっても、RTCタイムスタンプは保存されません。

1：タンパ検出イベント時、RTCタイムスタンプが保存されます。

RTC_CRレジスタでTSE = 0であってもTAMPTSは有効です。タイムスタンプフラグ は、タンパフラグから最大でck_apreの3サイクル後にセットされます。

注： RTC初期化モードに入る前にTAMPTSをクリアにする必要があります。

ビット24 **ITSE**：内部イベントでのタイムスタンプイネーブル

0：内部イベントでのタイムスタンプは無効です。

1：内部イベントでのタイムスタンプは有効です。

ビット23 **COE**：較正出力イネーブル

このビットは、CALIB出力を有効にします。

0：較正出力は無効です。

1：較正出力は有効です。

ビット22:21 **OSEL[1:0]**：出力選択

これらのビットは、TAMPALRM出力に送られるフラグの選択に使用します。

00：出力は無効です。

01：アラームA出力は有効です。

10：アラームB出力は有効です。

11：ウェイクアップ出力は有効です。

ビット20 **POL**：出力極性

このビットは、TAMPALRM出力の極性の設定に使用します。

0：ALRAF/ALRBF/WUTFが（OSEL[1:0] に応じて）アサートされた場合、またはTAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはハイになります。

1：ALRAF/ALRBF/WUTFが（OSEL[1:0] に応じて）アサートされた場合、またはTAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはローになります。

ビット19 **COSEL**：較正出力選択

COE = 1のとき、このビットによってCALIBに出力される信号を選択します。

0：較正出力は512 Hzです。

1：較正出力は1 Hzです。

これらの周波数は、RTCCLKが32.768kHzで、プリスケーラがデフォルト値（PREDIV_A = 127およびPREDIV_S = 255）の場合に有効です。セクション 63.3.18：較正クロック出力を参照してください。

ビット18 **BKP**：バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザが書き込むことができます。

ビット17 **SUB1H**：1時間差し引き（冬時間変更）

このビットを初期化モード以外のときにセットすると、現在時刻が0でない場合にカレンダ時刻から1時間を差し引きます。このビットは常に0として読み出されます。

現在時間が0のときにこのビットをセットしても、影響はありません。

0：影響なし

1：現在時刻から1時間差し引きます。これは、冬時間変更に使用できます。

ビット16 **ADD1H**：1時間加算（サマータイム変更）

このビットを初期化モード以外のときにセットすると、カレンダ時刻に1時間加算します。このビットは常に0として読み出されます。

0：影響なし

1：現在時刻に1時間加算します。これは、サマータイム変更に使用できます。

ビット15 **TSIE**：タイムスタンプ割込みイネーブル
0：タイムスタンプ割込みは無効です。
1：タイムスタンプ割込みは有効です。

ビット14 **WUTIE**：ウェイクアップ･タイマ割込みイネーブル
0：ウェイクアップタイマ割込みは無効です。
1：ウェイクアップタイマ割込みは有効です。

ビット13 **ALRBIE**：アラームB割込みイネーブル
0：アラームB割込みは無効です。
1：アラームB割込みは有効です。

ビット12 **ALRAIE**：アラームA割込みイネーブル
0：アラームA割込みは無効です。
1：アラームA割込みは有効です。

ビット11 **TSE**：タイムスタンプイネーブル
0：タイムスタンプは無効です。
1：タイムスタンプは有効です。

ビット10 **WUTE**：ウェイクアップ･タイマ･イネーブル
0：ウェイクアップタイマは無効です。
1：ウェイクアップタイマは有効です。
注： ウェイクアップタイマが無効の場合は、再度有効にする前にWUTWF=1になるまで待ちます。

ビット9 **ALRBE**：アラームBイネーブル
0：アラームBは無効です。
1：アラームBは有効です。

ビット8 **ALRAE**: アラームAイネーブル
0：アラームAは無効です。
1：アラームAは有効です。

ビット7 **SSRUIE**：SSRアンダーフロー割込みイネーブル
0：SSRアンダーフロー割込みは無効です。
1：SSRアンダーフロー割込みは有効です。

ビット6 **FMT**：AM/PM時間形式
0：24時間/日形式
1：AM/PM時間形式

ビット5 **BYPSHAD**：シャドウレジスタをバイパスします。
0：カレンダ値は（RTC_SSR、RTC_TR、RTC_DRから読み出す場合）、シャドウレジスタから取得され、これらは2 RTCCLKサイクルごとに1回更新されます。
1：カレンダ値は（RTC_SSR、RTC_TR、RTC_DRから読み出す場合）、カレンダカウンタから直接取得されます。
注： APBクロックの周波数がRTCCLKの7倍未満である場合、BYPSHADは1にセットする必要があります。

ビット4 **REFCKON**：RTC_REFINリファレンスクロック検出イネーブル（50または60Hz）
0：RTC_REFIN検出は無効です。
1：RTC_REFIN検出は有効です。
注： BINは0x00、PREDIV_Sは0x00FFでなければなりません。

ビット3 **TSEDGE**：タイムスタンプイベントアクティブエッジ
0：RTC_TS入力の立ち上がりエッジによってタイムスタンプイベントを生成します。
1：RTC_TS入力の立ち下がりエッジによってタイムスタンプイベントを生成します。
不要なTSF設定を回避するため、TSEDGEが変化した場合にはTSEをリセットする必要があります。

ビット2:0 **WUCKSEL[2:0]**：ck_wutウェイクアップクロックの選択

000：RTC/16クロックが選択されます。

001：RTC/8クロックが選択されます。

010：RTC/4クロックが選択されます。

011：RTC/2クロックが選択されます。

10x：BCDモードでは ck_spre（通常は1Hz）クロックが選択されます。バイナリまたは混合モードでは、このクロックがBCDUによって選択されます。

11x：BCDモードでは ck_spre（通常は1Hz）クロックが選択されます。バイナリまたは混合モードでは、このクロックがBCDUによって選択されます。さらに、2^{16} がWUTカウンタ値に追加されます。

注： **初期化モード（RTC_ICSR/INITF = 1）の場合のみ、このレジスタのビット6および4が書き込めます。**

WUT = ウェイクアップユニットカウンタ値。WUT = (0x0000～0xFFFF) + 0x10000 (WUCKSEL[2:1] = 11の場合追加されます。)

このレジスタのビット2～0は、RTC_CRのWUTEビット = 0かつRTC_ICSRのWUTWFビット = 1の場合にのみ書き込めます。

カレンダの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1HおよびSUB1Hの変更は、次の秒から有効になります。

63.6.8 RTC特権モード制御レジスタ（RTC_PRIVCFGR）

このレジスタに書き込めるのは、APBアクセスが特権のときだけです。このレジスタは書込み保護することができ、このレジスタの各ビットは、RTC_SECCFGRの設定に応じて、非セキュアアクセスから個別に書込み保護することもできます（セクション 63.3.5：RTC特権保護モードを参照）。

アドレスオフセット：0x1C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV	INIT PRIV	CAL PRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS PRIV	WUT PRIV	ALRB PRIV	ALRA PRIV
rw	rw	rw										rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **PRIV**：RTC特権保護

0：APBアクセスが特権でも非特権でも、その他の特権保護ビットで保護されているレジスタ以外、すべてのRTCレジスタに書き込めます。

1：APBアクセスが特権の場合のみ、すべてのRTCレジスタに書き込めます。

ビット14 **INITPRIV**：初期化の特権保護

0：APBアクセスが特権でも非特権でも、RTC初期化モード、カレンダ、プリスケーラのレジスタに書き込めます。

1：APBアクセスが特権の場合のみ、RTC初期化モード、カレンダ、プリスケーラのレジスタに書き込めます。

ビット13 **CALPRIV**：シフトレジスタ、サマータイム、較正およびリファレンスクロックの特権保護

0：APBアクセスが特権でも非特権でも、シフトレジスタ、サマータイム、較正およびリファレンスクロックを書き込めます。

1：APBアクセスが特権の場合のみ、シフトレジスタ、サマータイム、較正およびリファレンスクロックを書き込めます。

ビット12:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **TSPRIV**：タイムスタンプの特権保護

0：APBアクセスが特権でも非特権でも、RTCタイムスタンプの設定と割込みクリアを書き込めます。

1：APBアクセスが特権の場合のみ、RTCタイムスタンプの設定と割込みクリアを書き込めます。

ビット2 **WUTPRIV**：ウェイクアップタイマ特権保護

0：APBアクセスが特権でも非特権でも、RTCウェイクアップタイマの設定と割込みクリアを書き込めます。

1：APBアクセスが特権の場合のみ、RTCウェイクアップタイマの設定と割込みクリアを書き込めます。

ビット1 **ALRBPRIV**：アラームBの特権保護

0：APBアクセスが特権でも非特権でも、RTCアラームBの設定と割込みクリアを書き込めます。

1：APBアクセスが特権の場合のみ、RTCアラームBの設定と割込みクリアを書き込めます。

ビット0 **ALRAPRIV**：アラームA およびSSRアンダーフロー特権保護

0：APBアクセスが特権でも非特権でも、RTCアラームA およびSSRアンダーフローの設定と割込みクリアを書き込めます。

1：APBアクセスが特権の場合のみ、RTCアラームA およびSSRアンダーフローの設定と割込みクリアを書き込めます。

注：　読出し保護についての詳細は、セクション 63.3.5：RTC特権保護モードを参照してください。

63.6.9 RTCセキュア設定レジスタ（RCC_SECCFGR）

このレジスタに書き込めるのは、APBアクセスがセキュアなときだけです。

このレジスタは、全体的に書込み保護することができ、このレジスタの各ビットは、RTC_PRIVCFGRの設定に応じて、非特権アクセスから個別に書込み保護することもできます（セクション 63.3.5：RTC特権保護モードを参照）。

アドレスオフセット：0x20

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC	INIT SEC	CAL SEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS SEC	WUT SEC	ALRB SEC	ALRA SEC
rw	rw	rw										rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **SEC**：RTCグローバル保護

0：APBアクセスがセキュアでも非セキュアでも、その他のセキュア保護ビットで保護されているレジスタ以外、すべてのRTCレジスタに書き込めます。

1：APBアクセスがセキュアな場合のみ、すべてのRTCレジスタに書き込めます。

ビット14 **INITSEC**：初期化の保護

0：APBアクセスがセキュアでも非セキュアでも、RTC初期化モード、カレンダ、プリスケーラのレジスタに書き込めます。

1：APBアクセスがセキュアな場合のみ、RTC初期化モード、カレンダ、プリスケーラのレジスタに書き込めます。

ビット13 **CALSEC**：シフトレジスタ、サマータイム、較正およびリファレンスクロックの保護

0：APBアクセスがセキュアでも非セキュアでも、シフトレジスタ、サマータイム、較正およびリファレンスクロックを書き込めます。

1：APBアクセスがセキュアな場合のみ、シフトレジスタ、サマータイム、較正およびリファレンスクロックを書き込めます。

ビット12:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **TSSEC**：タイムスタンプの保護

0：APBアクセスがセキュアでも非セキュアでも、RTCタイムスタンプの設定と割込みクリアを書き込めます。

1：APBアクセスがセキュアな場合のみ、RTCタイムスタンプの設定と割込みクリアを書き込めます。

ビット2 **WUTSEC**：ウェイクアップタイマ保護

0：APBアクセスがセキュアでも非セキュアでも、RTCウェイクアップタイマの設定と割込みクリアを書き込めます。

1：APBアクセスがセキュアな場合のみ、RTCウェイクアップタイマの設定と割込みクリアを書き込めます。

ビット1 **ALRBSEC**：アラームBの保護

0:APBアクセスがセキュアでも非セキュアでも､RTCアラームBの設定と割込みクリアを書き込めます。

1：APBアクセスがセキュアな場合のみ、RTCアラームBの設定と割込みクリアを書き込めます。

ビット0 **ALRASEC**：アラームA およびSSRアンダーフロー保護

0：APBアクセスがセキュアでも非セキュアでも、RTCアラームA およびSSRアンダーフローの設定と割込みクリアを書き込めます。

1：APBアクセスがセキュアな場合のみ、RTCアラームA およびSSRアンダーフローの設定と割込みクリアを書き込めます。

注：　読出し保護についての詳細は、セクション 63.3.4：RTCセキュア保護モードを参照してください。

63.6.10 RTC書込み保護レジスタ（RTC_WPR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
								w	w	w	w	w	w	w	w

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **KEY[7:0]**：書込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に0x00が返されます。

RTCレジスタの書込み保護解除方法については、RTCレジスタ書込み保護を参照してください。

63.6.11 RTC較正レジスタ（RTC_CALR）

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは、非セキュアアクセスから書込み保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから書込み保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x28

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	CALW8	CALW16	LPCAL	Res.	Res.	Res.	CALM[8:0]								
rw	rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **CALP**：RTC周波数を488.5 ppm増加

0：RTCCLKパルスは加えられません。

1：RTCCLKの2^{11} パルスごとに1パルス効果的に挿入されます（周波数が488.5 ppm増加）。

この機能は、CALMと共に使用されることが想定されており、カレンダの周波数を高分解能で下げることができます。入力周波数が32768Hzの場合、32秒枠の間に追加されるRTCCLKパルスの数 z は z（512 × CALP）- CALMのように算出されます。

セクション 63.3.16：RTCの高精度デジタル較正を参照してください。

ビット14 **CALW8**：8秒較正サイクル周期の使用

CALW8が1にセットされると、8秒較正サイクル周期が選択されます。

注： CALW8 = 1の場合、CALM[1:0] は00に固定されます。セクション 63.3.16 : RTCの高精度デジタル較正を参照してください。

ビット13 **CALW16**：16秒較正サイクル周期の使用

CALW16が1にセットされると、16秒較正サイクル周期が選択されます。CALW8=1の場合、このビットを1にセットすることはできません。

注： CALW16 = 1の場合、CALM[0] は0に固定されます。セクション 63.3.16 : RTCの高精度デジタル較正を参照してください。

ビット12 **LPCAL**：RTC低電力モード

0：較正時間範囲が2^{20} RTCCLKとなり、高消費電力モードとなります。このモードは、32秒未満の較正時間範囲が必要な場合のみセットする必要があります。

1：較正時間範囲が2^{20} ck_apreとなり、超低電力モードで必要な設定となります。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **CALM[8:0]**：較正マイナス

RTCCLK 2^{20} パルス（入力周波数が32768 Hzの場合32秒）内のCALMをマスクすることによって、カレンダの周波数が下げられます。この方法により、カレンダの周波数を0.9537 ppmの分解能で下げることができます。

カレンダの周波数を高めるには、この機能をCALPと組み合わせて使用する必要があります。セクション 63.3.16： 2578ページのRTCの高精度デジタル較正を参照してください。

63.6.12 RTCシフト制御レジスタ（RTC_SHIFTR）

このレジスタは書込み保護されています。書込みアクセスの手順は、2573ページのRTCレジスタ書込み保護を参照してください。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x2C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS[14:0]														
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31 **ADD1S**：1秒加算

0：影響なし

1：時計/カレンダに1秒加算します。

このビットは書込み専用であり、常に0として読み出されます。シフト操作が保留中（RTC_ICSRでSHPF = 1）の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS（下記説明を参照）と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット30:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **SUBFS[14:0]**：秒の小数部を差し引きます。

このビットは書込み専用であり、常に0として読み出されます。シフト操作が保留中（RTC_ICSRでSHPF = 1）の場合、このビットに書き込んでも影響はありません。

SUBFSに書き込まれた値は、同期プリスケーラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き（遅延）されます。

遅れ（秒）= SUBFS / (PREDIV_S + 1)

ADD1S機能がSUBFSと共に用いられた場合、秒の小数部を効果的にクロックに加算する（クロックを進める）ことができ、実際のクロックの進みは次の式のとおりとなります。

進み（秒）= (1 - (SUBFS / (PREDIV_S + 1)))

混合BCD-binayモードでは（BIN=10または11）、SUBFS[14:BCDU+8] には0を書き込む必要があります。

注： SUBFSに書き込むことによりRSFはクリアされます。その後、ソフトウェアがRSF = 1まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実になります。

63.6.13 RTCタイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_SRでTSFが1にセットされている場合にのみ有効です。また、TSFビットがリセットされるとクリアされます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x30

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **PM**: AM/PM表記
0：AMまたは24時間形式
1：PM

ビット21:20 **HT[1:0]**：BCD形式での時の十の位

ビット19:16 **HU[3:0]**：BCD形式での時の一の位

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **MNT[2:0]**：BCD形式での分の十の位

ビット11:8 **MNU[3:0]**：BCD形式での分の一の位

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **ST[2:0]**：BCD形式での秒の十の位

ビット3:0 **SU[3:0]**：BCD形式での秒の一の位

63.6.14 RTCタイムスタンプ日付レジスタ（RTC_TSDR）

このレジスタの内容は、RTC_SRでTSFが1にセットされている場合にのみ有効です。また、TSFビットがリセットされるとクリアされます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x34

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:13 **WDU[2:0]**：曜日

ビット12 **MT**：BCD形式での月の十の位

ビット11:8 **MU[3:0]**：BCD形式での月の一の位

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **DT[1:0]**：BCD形式での日の十の位

ビット3:0 **DU[3:0]**：BCD形式での日の一の位

63.6.15 RTCタイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_SRでTSFが1にセットされている場合にのみ有効です。また、TSFビットがリセットされるとクリアされます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x38

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SS[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 SS[31:0]：サブセカンド値/同期バイナリカウンタ値

SS[31:0] は、タイムスタンプイベントが発生したときの同期プリスケーラのカウンタの値です。

63.6.16 RTCアラームAレジスタ（RTC_ALRMAR）

このレジスタは、RTC_CRレジスタのALRAIEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x40

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **MSK4**：アラームA日付マスク

0：日付/曜日が一致すると、アラームAがセットされます。

1：アラームAの比較では日付/曜日を無視します。

ビット30 **WDSEL**：曜日選択
0：DU[3:0] は日の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット29:28 **DT[1:0]**：BCD形式での日の十の位

ビット27:24 **DU[3:0]**：BCD形式での日の一の位または曜日

ビット23 **MSK3**：アラームA時マスク
0：時が一致すると、アラームAがセットされます。
1：アラームAの比較では時を無視します。

ビット22 **PM**: AM/PM表記
0：AMまたは24時間形式
1：PM

ビット21:20 **HT[1:0]**：BCD形式での時の十の位

ビット19:16 **HU[3:0]**：BCD形式での時の一の位

ビット15 **MSK2**：アラームA分マスク
0：分が一致すると、アラームAがセットされます。
1：アラームAの比較では分を無視します。

ビット14:12 **MNT[2:0]**：BCD形式での分の十の位

ビット11:8 **MNU[3:0]**：BCD形式での分の一の位

ビット7 **MSK1**：アラームA秒マスク
0：秒が一致すると、アラームAがセットされます。
1：アラームAの比較では秒を無視します。

ビット6:4 **ST[2:0]**：BCD形式での秒の十の位

ビット3:0 **SU[3:0]**：BCD形式での秒の一の位

63.6.17 RTCアラームAサブセカンドレジスタ（RTC_ALRMASSR）

このレジスタは、RTC_CRレジスタのALRAIEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x44

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SSCLR	Res.	MASKSS[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw		rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット31 **SSCLR**：アラームでの同期カウンタのクリア（バイナリモードのみ）

0：同期バイナリカウンタ（RTC_SSRのSS[31:0]）はフリーランニングです。

1：同期バイナリカウンタ（RTC_SSRのSS[31:0]）は、0xFFFF FFFFからRTC_ALRABINR .SS[31:0]値まで動作し、RTC_ALRABINR .SS[31:0] に達した1 ck_apreサイクル後に自動的に0xFFFF FFFFで再ロードされます。

注： BCDまたは混合モードが使用されている場合（BIN = 00、10、11のいずれか）、SSCLRは0のままにしておく必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29:24 **MASKSS[5:0]**：このビットから始まる最上位ビットのマスク

0：アラームAに対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。

1：アラームAの比較ではSS[31:1] を無視します。SS[0] のみ比較されます。

2：アラームAの比較ではSS[31:2] を無視します。SS[1:0] のみ比較されます。

.....

31：アラームAの比較ではSS[31] を無視します。SS[30:0] のみ比較されます。

32から63：アラームをアクティブにするには、32のすべてのSSビットを比較して一致させる必要があります。

注： BCDモード（BIN=00）では、同期カウンタのオーバーフロービット（ビット31:15）が比較されることはありません。これらのビットは、シフト操作後に限り、0でなくなる場合があります。

ビット23:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **SS[14:0]**: サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラームAをアクティブ化するかどうかを決定します。0～MASKSS-1のビットだけが比較されます。

このフィールドは、RTC_ALRABINRのSS[14:0] のミラーであり、RTC_ALRABINRでも読出しまたは書込みできます。

注： TAMP_ATCR1でSSCLRにATCKSEL[3] = 1がセットされている場合、SS[3:0] は0000でなければなりません。

63.6.18 RTCアラームBレジスタ（RTC_ALRMBR）

このレジスタは、RTC_CRレジスタのALRBEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x48

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WD SEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **MSK4**：アラームB日付マスク
0：日付/曜日が一致すると、アラームBがセットされます。
1：アラームBの比較では日付/曜日を無視します。

ビット30 **WDSEL**：曜日選択
0：DU[3:0] は日の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット29:28 **DT[1:0]**：BCD形式での日の十の位

ビット27:24 **DU[3:0]**：BCD形式での日の一の位または曜日

ビット23 **MSK3**：アラームB時マスク
0：時が一致すると、アラームBがセットされます。
1：アラームBの比較では時を無視します。

ビット22 **PM**: AM/PM表記
0：AMまたは24時間形式
1：PM

ビット21:20 **HT[1:0]**：BCD形式での時の十の位

ビット19:16 **HU[3:0]**：BCD形式での時の一の位

ビット15 **MSK2**：アラームB分マスク
0：分が一致すると、アラームBがセットされます。
1：アラームBの比較では分を無視します。

ビット14:12 **MNT[2:0]**：BCD形式での分の十の位

ビット11:8 **MNU[3:0]**：BCD形式での分の一の位

ビット7 **MSK1**：アラームB秒マスク
0：秒が一致すると、アラームBがセットされます。
1：アラームBの比較では秒を無視します。

ビット6:4 **ST[2:0]**：BCD形式での秒の十の位

ビット3:0 **SU[3:0]**：BCD形式での秒の一の位

63.6.19 RTCアラームBサブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CRレジスタのALRBEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x4C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SSCLR	Res.	MASKSS[5:4]		MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw		rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット31 **SSCLR**：アラームでの同期カウンタのクリア（バイナリモードのみ）

0：同期バイナリカウンタ（RTC_SSRのSS[31:0]）はフリーランニングです。

1：同期バイナリカウンタ（RTC_SSRのSS[31:0]）は、0xFFFF FFFFからRTC_ALRBBINRのSS[31:0]値まで動作し、RTC_ALRBBINRのSS[31:0] に達した1 ck_apreサイクル後に自動的に0xFFFF FFFFで再ロードされます。

注： BCDまたは混合モードが使用されている場合（BIN = 00、10、11のいずれか）、SSCLRは0のままにしておく必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29:24 **MASKSS[5:0]**：このビットから始まる最上位ビットのマスク

0：アラームBに対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。

1：アラームBの比較ではSS[31:1] を無視します。SS[0] のみ比較されます。

2：アラームBの比較ではSS[31:2] を無視します。SS[1:0] のみ比較されます。

.....

31：アラームBの比較ではSS[31] を無視します。SS[30:0] のみ比較されます。

32から63：アラームをアクティブにするには、32のすべてのSSビットを比較して一致させる必要があります。

注： BCDモード（BIN=00）では、同期カウンタのオーバーフロービット（ビット15）が比較されることはありません。このビットは、シフト操作後に限り、0でなくなる場合があります。

ビット23:15 予約済みであり、リセット値に保持する必要があります。

ビット14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラームBをアクティブ化するかどうかを決定します。0～MASKSS-1のビットだけが比較されます。

このフィールドは、RTC_ALRBBINRのSS[14:0] のミラーであり、RTC_ALRBBINRでも読出しまたは書込みできます。

注： TAMP_ATCR1でSSCLRにATCKSEL[3] = 1がセットされている場合、SS[3:0] は0000でなければなりません。

63.6.20 RTCステータスレジスタ（RTC_SR）

このレジスタは全体的に保護することができ、このレジスタの各ビットは非セキュアアクセスから個別に保護することもできます。セクション 63.3.4：RTCセキュア保護モードを参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x50

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUF	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
									r	r	r	r	r	r	r

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **SSRUF**：SSRアンダーフローフラグ

このフラグは、SSRが0を下回る場合にハードウェアによってセットされます。SSCLR=1の場合、SSRUFはセットされません。

ビット5 **ITSF**：内部タイムスタンプフラグ

このフラグは、内部イベントでタイムスタンプが発生したときに、ハードウェアによってセットされます。

ビット4 **TSOVF**：タイムスタンプオーバーフローフラグ

このフラグは、TSFが既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

TSOVFは、TSFビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSFビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット3 **TSF**：タイムスタンプフラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

ITSFフラグがセットされている場合、ITSFとともにTSFをクリアする必要があります。

注： TAMPTS = 1で、タンパイベントに続くck_apreの3サイクルの間にタンパフラグが読み出される場合、TSFはセットされません。詳細については、タンパイベント時のタイムスタンプを参照してください。

ビット2 **WUTF**：ウェイクアップタイマフラグ

このフラグは、ウェイクアップ自動再ロードカウンタが0に到達したときに、ハードウェアによってセットされます。

WUTOCLR[15:0] が0x0000でない場合、ウェイクアップ自動再ロードカウンタがWUTOCLR値に達すると、ハードウェアによってWUTFがクリアされます。

WUTOCLR[15:0] が0x0000である場合、WUTFはソフトウェアでクリアする必要があります。

このフラグは、WUTFが再び1にセットされる前、RTCCLK 1.5周期以上前にソフトウェアでクリアする必要があります。

ビット1 **ALRBF**：アラームBフラグ

このフラグは、時刻／日付レジスタ（RTC_TRおよびRTC_DR）がアラームBレジスタ（RTC_ALRMBR）と一致したときにハードウェアによってセットされます。

ビット0 **ALRAF**：アラームAフラグ

このフラグは、時刻／日付レジスタ（RTC_TRおよびRTC_DR）がアラームAレジスタ（RTC_ALRMAR）と一致したときにハードウェアによってセットされます。

注： このレジスタのビットは、RTC_SCRレジスタで対応するクリアビットをセットしてから2 APBクロックサイクル後にクリアされます。

63.6.21 RTC 非セキュアマスク済み割込みステータスレジスタ（RTC_MISR）

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x54

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUMF	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
									r	r	r	r	r	r	r

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **SSRUMF**：SSRアンダーフロー非セキュアマスク済みフラグ

このフラグは、SSRアンダーフロー非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット5 **ITSMF**：内部タイムスタンプ非セキュアマスク済みフラグ

このフラグは、内部イベントでタイムスタンプが発生して、タイムスタンプ非セキュア割込みが発生したときに、ハードウェアによってセットされます。

ビット4 **TSOVMF**：タイムスタンプオーバーフロー非セキュアマスク済みフラグ

このフラグは、TSMFがすでにセットされている間にタイムスタンプ割込みが発生した場合に、ハードウェアによってセットされます。

TSOVFは、TSFビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSFビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット3 **TSMF**：タイムスタンプ非セキュアマスク済みフラグ

このフラグは、タイムスタンプ非セキュア割込みが発生した場合にハードウェアによってセットされます。

ITSFフラグがセットされている場合、ITSFとともにTSFをクリアする必要があります。

ビット2 **WUTMF**：ウェイクアップタイマ非セキュアマスク済みフラグ

このフラグは、ウェイクアップタイマ非セキュア割込みが発生した場合にハードウェアによってセットされます。

このフラグは、WUTFが再び1にセットされる前、RTCCLK 1.5周期以上前にソフトウェアでクリアする必要があります。

ビット1 **ALRBMF**：アラームB 非セキュアマスク済みフラグ

このフラグは、アラームB 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット0 **ALRAMF**：アラームAマスク済みフラグ

このフラグは、アラームA 非セキュア割込みが発生した場合にハードウェアによってセットされます。

63.6.22 RTCセキュアマスク済み割込みステータスレジスタ（RTC_SMISR）

このレジスタは全体的に保護することができ、このレジスタの各ビットは非セキュアアクセスから個別に保護することもできます。セクション 63.3.4：RTCセキュア保護モードを参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x58

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUMF	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
									r	r	r	r	r	r	r

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **SSRUMF**：SSRアンダーフローセキュアマスク済みフラグ

このフラグは、SSRアンダーフローセキュア割込みが発生した場合にハードウェアによってセットされます。

ビット5 **ITSMF**：内部タイムスタンプ割込みセキュアマスク済みフラグ

このフラグは、内部イベントでタイムスタンプが発生して、タイムスタンプセキュア割込みが発生したときに、ハードウェアによってセットされます。

ビット4 **TSOVMF**：タイムスタンプオーバーフロー割込みセキュアマスク済みフラグ

このフラグは、TSMFがすでにセットされている間にタイムスタンプセキュア割込みが発生した場合に、ハードウェアによってセットされます。

TSOVFは、TSFビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSFビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット3 **TSMF**：タイムスタンプ割込みセキュアマスク済みフラグ

このフラグは、タイムスタンプセキュア割込みが発生した場合にハードウェアによってセットされます。

ITSFフラグがセットされている場合、ITSFとともにTSFをクリアする必要があります。

ビット2 **WUTMF**：ウェイクアップタイマ割込みセキュアマスク済みフラグ

このフラグは、ウェイクアップタイマセキュア割込みが発生した場合にハードウェアによってセットされます。

このフラグは、WUTFが再び1にセットされる前、RTCCLK 1.5周期以上前にソフトウェアでクリアする必要があります。

ビット1 **ALRBMF**：アラームB割込みセキュアマスク済みフラグ

このフラグは、アラームBセキュア割込みが発生した場合にハードウェアによってセットされます。

ビット0 **ALRAMF**：アラームA割込みセキュアマスク済みフラグ

このフラグは、アラームAセキュア割込みが発生した場合にハードウェアによってセットされます。

63.6.23 RTCステータスクリアレジスタ（RTC_SCR）

このレジスタは全体的に保護することができ、このレジスタの各ビットは非セキュアアクセスから個別に保護することもできます。セクション 63.3.4：RTCセキュア保護モードを参照してください。

このレジスタは全体的に保護することができ、このレジスタの各ビットは非特権アクセスから個別に保護することもできます。セクション 63.3.5：RTC特権保護モードを参照してください。

アドレスオフセット：0x5C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSSRUF	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
									w	w	w	w	w	w	w

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CSSRUF**：SSRアンダーフローフラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのSSRUFがクリアされます。

ビット5 **CITSF**：内部タイムスタンプ·フラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのITSFビットがクリアされます。

ビット4 **CTSOVF**：タイムスタンプ·オーバーフロー·フラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのTSOVFビットがクリアされます。
TSOVFは、TSFビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSFビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット3 **CTSF**：タイムスタンプ·フラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのTSFビットがクリアされます。
ITSFフラグがセットされている場合、CRSFとCITSFをセットして、ITSFとともにTSFをクリアする必要があります。

ビット2 **CWUTF**：ウェイクアップタイマフラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのWUTFビットがクリアされます。

ビット1 **CALRBF**：アラームBフラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのALRBFビットがクリアされます。

ビット0 **CALRAF**：アラームAフラグのクリア
このビットに1を書き込むと、RTC_SRレジスタのALRAFビットがクリアされます。

63.6.24 RTCアラームAバイナリモードレジスタ（RTC_ALRABINR）

このレジスタは、RTC_CRレジスタのALRAIEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x70

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SS[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SS[31:0]**：バイナリモードでの同期カウンタアラーム値

この値が同期カウンタの内容と比較され、アラームAをアクティブ化するかどうかを決定します。0～MASKSS-1のビットだけが比較されます。

SS[14:0] はRTC_ALRMASSRRのSS[14:0] のミラーであり、RTC_ALRMASSRでも読出しまたは書込みできます。

注： TAMP_ATCR1でSSCLRにATCKSEL[3] = 1がセットされている場合、SS[3:0] は0000でなければなりません。

63.6.25 RTCアラームBバイナリモードレジスタ（RTC_ALRBBINR）

このレジスタは、RTC_CRレジスタのALRBEがリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは、非セキュアアクセスから保護できます。セクション 63.3.4 : RTCセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 63.3.5 : RTC特権保護モードを参照してください。

アドレスオフセット：0x74

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SS[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **SS[31:0]**：バイナリモードでの同期カウンタアラーム値

この値が同期カウンタの内容と比較され、アラームBをアクティブ化するかどうかを決定します。0～MASKSS-1のビットだけが比較されます。

SS[14:0] はRTC_ALRMBSSRRのSS[14:0] のミラーであり、RTC_ALRMBSSRでも読出しまたは書込みできます。

注： TAMP_ATCR1でSSCLRにATCKSEL[3] = 1がセットされている場合、SS[3:0] は0000でなければなりません。

63.6.26 RTCレジスタマップ

表634.RTCレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RTC_TR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT [1:0]		HU[3:0].				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x04	RTC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]				WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0].			
	リセット値									0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1			0	0	0	0	0	1
0x08	RTC_SSR	SS[31:16]																SS[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	RTC_ICSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RECALPF	Res.	Res.	Res.	BCDU [2:0]			BIN [1:0]		INIT	INITF	RSF	INITS	SHPF	WUT WF	Res.	Res.
	リセット値																0				0	0	0	0	0	0	0	0	0	0	1		
0x10	RTC_PRER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]							PREDIV_S[14:0]															
	リセット値										1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x14	RTC_WUTR	WUTOCLR[15:0]																WUT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x18	RTC_CR	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	ALRBFCLR	ALRAFCLR	TAMPOE	TAMPTS	ITSE	COE	O SEL [1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	SSRUIE	FMT	BYPSHAD	REFCKON	TSEDGE	WUCK SEL[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	RTC_ PRIVCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIV	INITPRIV	CALPRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSPRIV	WUTPRIV	ALRBPRIV	ALRAPRIV
	リセット値																	0	0	0										0	0	0	0
0x20	RTC_ SECCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC	INITSEC	CALSEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSSEC	WUTSEC	ALRBSEC	ALRASEC
	リセット値																	0	0	0										0	0	0	0
0x24	RTC_WPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x28	RTC_ CALR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALP	CALW8	CALW16	LPCAL	Res.	Res.	Res.	CALM[8:0]								
	リセット値																	0	0	0	0				0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBFS[14:0]														
	リセット値	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	RTC_TSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0

表634.RTCレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x34	RTC_TSDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0].			
	リセット値																	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x38	RTC_TSSSR	SS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	RTC_ALRMAR	MSK4	WDSEL	DT [1:0]		DU[3:0].				MSK3	PM	HT [1:0]		HU[3:0].				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	RTC_ ALRMASSR	SSCLR	Res.	MASKSS [5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値	0		0	0	0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	RTC_ALRMBR	MSK4	WDSEL	DT [1:0]		DU[3:0].				MSK3	PM	HT [1:0]		HU[3:0].				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	RTC_ALRMBS SR	SSCLR	Res.	MASKSS [5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値	0		0	0	0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	RTC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUF	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
	リセット値																										0	0	0	0	0	0	0
0x54	RTC_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUMF	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
	リセット値																										0	0	0	0	0	0	0
0x58	RTC_SMISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSRUMF	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
	リセット値																										0	0	0	0	0	0	0
0x5C	RTC_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSSRUF	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
	リセット値																										0	0	0	0	0	0	0
0x70	RTC_ ALRABINR	SS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	RTC_ ALRBBINR	SS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

64 タンパおよびバックアップレジスタ（TAMP）

64.1 概要

耐タンパ検出回路は、機密データを外部攻撃から保護するために使用されます。3232bitバックアップレジスタは、すべての低電力モードとV_{BAT} モードで保持されます。バックアップレジスタとデバイスのその他の機密情報は、8 タンパピンと11内部タンパを備えたこの耐タンパ検出回路によって保護されます。外部のタンパピンは、エッジ検出、フィルタあり／なしのレベル検出、タンパピンが外部にオープンまたは短絡したことを自動チェックすることで、セキュリティレベルを向上させるアクティブタンパに対して設定できます。

64.2 TAMPの主な機能

- タンパ検出によって バックアップレジスタ、バックアップSRAM、SRAM2、キャッシュ、および暗号ペリフェラル を消去することもできます。タンパによって保護されるデバイスリソースは、「デバイス機密情報」と呼ばれます。
- 3232bitバックアップレジスタ：
  - バックアップレジスタ（TAMP_BKPxR）は、V_{DD} 電源がオフに切り替わったときにV_{BAT} による電源がオンの状態になる バックアップドメインに実装されます。
- 8 の外部タンパ検出イベント用の最大8 個のタンパピン：
  - アクティブタンパモード：タンパ出力とタンパ入力を連続的に比較して、物理的なオープン/ショート攻撃から保護します。
  - フレキシブルアクティブタンパI/O管理：メッシュ 4枚（各入力が専用の出力に関連付けられる）からメッシュ 7枚（1つの出力を最大7つのタンパ入力で共有）
  - パッシブタンパ：内部プルアップハードウェア管理により超低消費電力エッジまたはレベル検出を行います。
  - 設定可能なデジタルフィルタ。
- 過渡または環境変動攻撃に対する保護のための 11 の内部タンパイベント
- 各タンパは次の2つのモードで設定できます。
  - 確認モード：タンパ検出時に、バックアップレジスタを含めて、機密情報がすぐに消去されます。
  - 潜在モード：タンパ検出の後に、大部分の機密情報の消去がソフトウェアによって起動されます。
- あらゆるタンパ検出によってRTCタイムスタンプイベントを生成可能
- TrustZoneサポート：
  - タンパのセキュアまたは非セキュア設定
  - 3種類の設定可能なサイズを持つエリアでのバックアップレジスタ設定

    読出し／書込みがセキュアなエリア1個

    書込みがセキュア/読出しが非セキュアのエリア1個

    読出し/書込みが非セキュアなエリア1個
  - バックアップレジスタに格納され、読出し/書込みアクセスに対して保護されたセキュアなAES用のブートハードウェアキー
- タンパ設定とバックアップレジスタ特権保護
- 単調カウンタ

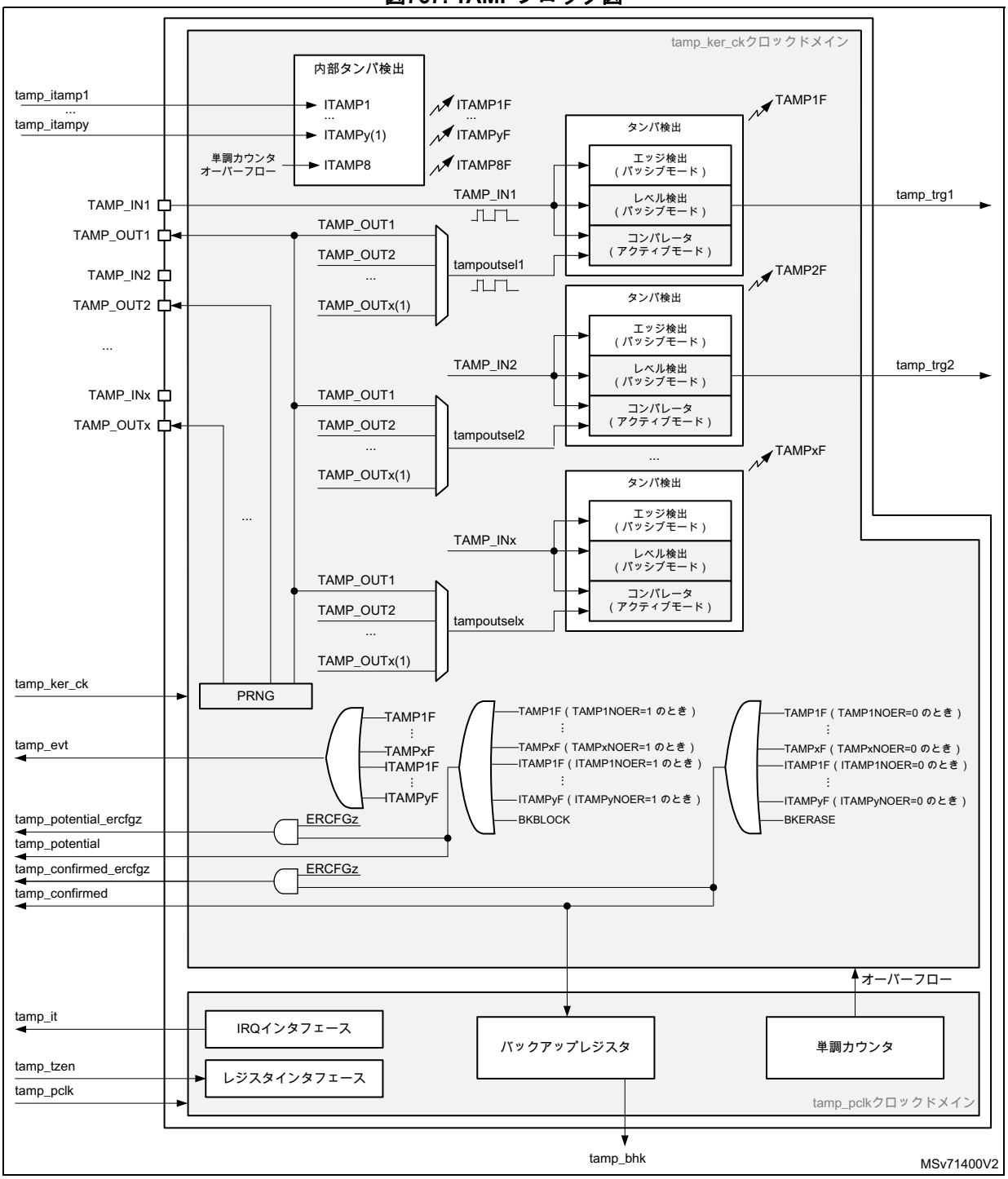
64.3 TAMPの実装

STM32U575/585のリビジョンXデバイスには、アクティブなタンパプリスケーラ拡張は実装されていません。この機能は、その他すべてのSTM32U575/585のリビジョン、およびその他のSTM32U5シリーズのデバイスに実装されています。

64.4 TAMPの機能説明

64.4.1 TAMPブロック図

図767. TAMPブロック図



1. 外部タンパおよび内部タンパの数は製品によります。

64.4.2 TAMPピンおよび内部信号

表635.TAMPの入出力ピン

ピン名	信号タイプ	説明
TAMP_INx（x = ピンインデックス）	入力	タンパ入力ピン
TAMP_OUTx（x = ピンインデックス）	出力	タンパ出力ピン（アクティブモードのみ）

表636.TAMP内部入力／出力信号

内部信号名	信号タイプ	説明
tamp_ker_ck	入力	TAMPカーネルクロック（rtc_ker_ckに接続、本書ではRTCCLKとも言います）
tamp_pclk	入力	TAMP APBクロック（rtc_pclkに接続）
tamp_itamp[y] （y = 信号インデックス）	入力	内部タンパイベントソース
tamp_tzen	入力	TAMP TrustZone有効
tamp_evt	出力	タンパイベント検出フラグ（内部または外部タンパ）。確認済みモード設定と潜在モード設定の両方。
tamp_potential	出力	潜在タンパ検出信号。デバイス機密情報(1) の保護のために使用されます。 この信号は次の場合にアクティブです。 – タンパイベント検出フラグ（内部または外部タンパ）が潜在モードで生成されている場合 – BKBLOCK に 1 を書き込んでソフトウェアリクエストを行った場合
tamp_confirmed	出力	確認済みタンパ検出信号。デバイス機密情報 (1) 保護のために使用されます。 この信号は次の場合にアクティブです。 – タンパイベント検出フラグ（内部または外部タンパ）が確認済みモードで生成されている場合 – BKERASE に 1 を書き込んでソフトウェアリクエストを行った場合
tamp_potential_ercfgz （z = 信号インデックス）	出力	潜在タンパ検出信号は ERCFGz = 1の場合のみ生成されます。 この信号は次の場合にアクティブです。 – タンパイベント検出フラグ（内部または外部タンパ）が潜在モードで生成されている場合 – BKBLOCK に 1 を書き込んでソフトウェアリクエストを行った場合
tamp_confirmed_ercfgz （z = 信号インデックス）	出力	確認済みタンパ検出信号は ERCFGz = 1の場合のみ生成されます。 この信号は次の場合にアクティブです。 – タンパイベント検出フラグ（内部または外部タンパ）が確認済みモードで生成されている場合 – BKERASE に 1 を書き込んでソフトウェアリクエストを行った場合
tamp_it	出力	TAMP割込み（詳細については、セクション 64.6 : TAMP割込みを参照してください）

表636.TAMP内部入力／出力信号 （続き）

内部信号名	信号タイプ	説明
tamp_trg[x] （x = 信号インデックス）	出力	タンパ検出トリガ
tamp_bhk	出力	タンパブートハードウェアキーバス

1. 表 637：TAMP相互接続を参照してください。

通常、TAMPカーネルクロックは、RCCで他のクロックソースを選択できますが、32.768 kHzでのLSEです（詳細はRCCを参照してください）。選択したクロックによって、一部の低消費電力モードやV_{BAT}で使用できない検出モードがあります（詳細は、セクション 64.5：TAMP低消費電力モードを参照してください）。

表637.TAMP相互接続

信号名	転送元／転送先
tamp_tzen	FLASHオプションバイトから：TZEN
tamp_evt	タイムスタンプイベントの生成に使用されるrtc_tamp_evt
tamp_potential	tamp_potential信号は、以下に示すデバイス機密情報への読出し/書込みアクセスをブロックするために使用されます。 – バックアップレジスタ – SRAM2 システムFlashメモリのRHUK（ルート･ハードウェア･ユニーク･キー）とSAESへのBHK（ブート･ハードウェア･キー）ハードウェアバスはブロックされます。 tamp_potential信号は、以下に示すデバイス機密情報を消去するために使用されます。 – ICACHEの内容 – SAES、AES、HASHペリフェラル – PKA SRAM 消去の実行中は、デバイス機密情報のアクセスがブロックされます。
tamp_confirmed	tamp_confirmed信号は、以下に示すデバイス機密情報を消去するために使用されます。 – バックアップレジスタ – SRAM2 – ICACHE/DCACHE1の内容 – OTFDECキーおよびCRCレジスタ – SAES、AES、HASHペリフェラル – PKA SRAM 消去の実行中は、デバイス機密情報のアクセスがブロックされます。 SAESへのシステムFlashメモリのRHUK（ルート･ハードウェア･ユニーク･キー）ハードウェアバスはブロックされます。
tamp_potential_ercfg0	TAMP_ERCFGRでビット ERCFG0がセットされている場合、以下に示すデバイス機密情報への読出し/書込みアクセスをブロックするために、tamp_potential_ercfg0信号が使用されます。 – バックアップSRAM
tamp_confirmed_ercfg0	TAMP_ERCFGRでビット ERCFG0がセットされている場合、以下に示すデバイス機密情報を消去するために、tamp_confirmed_ercfg0信号が使用されます。 – バックアップSRAM 消去の実行中は、デバイス機密情報のアクセスがブロックされます。

表637.TAMP相互接続 （続き）

信号名	転送元／転送先
tamp_itamp1	バックアップドメイン電圧閾値監視 (1)
tamp_itamp2	温度監視(1)
tamp_itamp3	LSE監視（LSECSS)(2)
tamp_itamp5	RTCカレンダオーバーフロー（rtc_calovf）
tamp_itamp6	RDP > 0の場合のJTAG/SWDアクセス
tamp_itamp7	ADC4（adc4_awd1）アナログウォッチドッグ監視1
tamp_itamp8(3)	単調カウンタ1オーバーフロー
tamp_itamp9	暗号化ペリフェラル･フォールト（SAESまたはAESまたはPKAまたはTRNG）
tamp_itamp11	タンパフラグ（潜在タンパタイムアウト）のセット時、IWDG のリセット
tamp_itamp12	ADC4（adc4_awd2）アナログウォッチドッグ監視2
tamp_itamp13	ADC4（adc4_awd3）アナログウォッチドッグ監視3
tamp_bhk	saes_bhk。このバスは、セキュアなAESコプロセッサにブート･ハードウェア･キーをロードするために使用されます。

1. この監視を有効にするには、PWRバックアップドメイン制御レジスタ1（PWR_BDCR1）でMONENをセットする必要があります。
2. この監視を有効にするには、RCCバックアップドメイン制御レジスタ（RCC_BDCR）で LSECSSON をセットする必要があります。
3. この信号はTAMPペリフェラルで生成されます。

TZENオプションビットは、デバイスでTrustZoneをアクティブにするために使用します。

TZEN = 1：TrustZoneは有効です。

TZEN = 0：TrustZoneは無効です。

TrustZoneが無効である場合、TAMPレジスタへのAPBアクセスは非セキュアとなります。

64.4.3 RTCおよびTAMPによって制御されるGPIO

セクション 63.3.3：RTCおよびTAMPによって制御されるGPIOを参照してください。

64.4.4 TAMPレジスタ書込み保護

システムリセット後、電源制御ペリフェラルの DBP ビットによって、TAMPレジスタ（バックアップレジスタを含む）は不要な書込みアクセスから保護されます（PWR電源制御のセクションを参照）。TAMPレジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

64.4.5 TAMPセキュア保護モード

デフォルトでは、バックアップドメインのパワーオンリセット後、セキュアモードのみで書込み可能なTAMPセキュア設定レジスタ（TAMP_SECCFGR）を除いて、すべてのTAMPレジスタがセキュアモードと非セキュアモードの両方で読出しまたは書込みできます。TAMP保護設定は、システムリセットの影響を受けません。

- TAMP_SECCFGR レジスタのTAMPSECビットがセットされている場合：
  - TAMP レジスタの書込みは、独自の保護設定を持つバックアップレジスタを除き、セキュアモードでのみ可能です。
  - TAMP_SECCFGR、TAMP_PRIVCFGRおよびTAMP_MISRの読出しは、セキュアモードと非セキュアモードで常に可能です。その他のすべてのTAMPレジスタは、独自の保護設定を持つバックアップレジスタと単調カウンタを除き、セキュアモードでのみ読出しできます。
- TAMP_SECCFGR レジスタにCNT1SECビットがセットされている場合：TAMP_COUNT1Rはセキュアモードでのみ読み書きできます。

セキュア保護されたレジスタへの非セキュアアクセスは拒否されます。

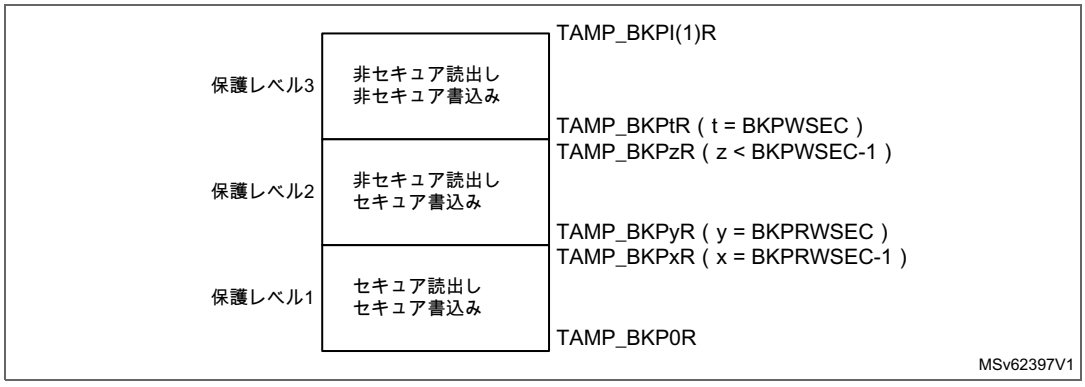
- 生成されるバスエラーはありません。
- TZIC（TrustZone不正アクセスコントローラ）でのフラグ/割込みを通じて通知が生成されます。
- 書込み保護されている場合、そのビットは書き込まれません。
- 読出し保護されている場合、そのビットは0として読み出されます。

1つ以上の機能がセキュアに設定されると、TAMPリセットおよびクロック制御もRCCでセキュアになります。

64.4.6 バックアップレジスタ保護ゾーン

バックアップレジスタの保護は、BKPRWSEC[7:0] および BKPWSEC[7:0] によって設定されます（下図を参照してください）。

図768. バックアップレジスタ保護ゾーン



1. I = 最後のバックアップレジスタインデックス

TZEN =1の場合、TAMP_PRIVCFGRのBKPWPRIVおよびBKPRWPRIVのビットは、セキュアモードでのみ書込みできます。

64.4.7 TAMP特権保護モード

デフォルトでは、バックアップドメインのパワーオンリセット後、特権モードのみで書込み可能なTAMP特権設定レジスタ（TAMP_PRIVCFGR）を除いて、すべてのTAMPレジスタが特権モードと非特権モードの両方で読出しまたは書込みできます。TAMP保護設定は、システムリセットの影響を受けません。

TAMP_PRIVCFGRレジスタのTAMPPRIVビットがセットされている場合：

- TAMP レジスタの書込みは、独自の保護設定を持つバックアップレジスタと単調カウンタを除き、特権モードでのみ可能です。
- TAMP_PRIVCFGRレジスタにCNT1PRIVビットがセットされている場合：TAMP_COUNT1Rは特権モードでのみ読み書きできます。
- TAMP_SECCFGR、TAMP_PRIVCFGRの読出しは、特権と非特権モードで常に可能です。その他のすべてのTAMPレジスタは、独自の保護設定を持つバックアップレジスタと単調カウンタを除き、特権モードでのみ読出しできます。

バックアップレジスタの保護は、保護ゾーン1では BKPRWSEC[7:0] およびBKPRWPRIV、保護ゾーン2では BKPRWSEC[7:0]、BKPWSEC[7:0] およびBKPWPRIVによって設定されます（図768を参照）。BHKLOCKビットは、BKPRWPRIVビットがセットされている場合に、特権モードでのみ書込みできます。

特権保護されたレジスタへの非特権アクセスは拒否されます。

- 生成されるバスエラーはありません。
- 書込み保護されている場合、そのビットは書き込まれません。
- 読出し保護されている場合、そのビットは0として読み出されます。

64.4.8 ブート･ハードウェア･キー（BHK）

TAMP_BKP0RからTAMP_BKP7Rまでの最初の8個のバックアップレジスタを使用して、セキュアなAESのためのブート･ハードウェア･キーを格納できます。

このため、これらのレジスタは保護ゾーン1に属している必要があります。BKPRWSEC は8以上である必要があります。

ブート･ハードウェア･キーによるバックアップレジスタの書込みが完了したら、TAMP_SECCFGRレジスタでBHKLOCKビットをセットする必要があります。BHKLOCKのセットが完了すると、ソフトウェアから8個のバックアップレジスタへのアクセスはできなくなります。0として読み出され、これらのレジスタへの書込みは無視されます。BHKLOCKはソフトウェアではクリアできず、タンパイベントの後で、または読出し保護（RDP）の無効化時にハードウェアによってクリアされます。また、BKERASEコマンドでもクリアされます （いずれの場合も、バックアップレジスタも消去されます）。

セキュアなAESコプロセッサ（SAES）にブート･ハードウェア･キーをダウンロードする手順については、SAESのセクションを参照してください。

64.4.9 タンパ検出

タンパ検出の主な目的は、デバイスに対する外部攻撃からデバイス機密情報を保護することです。検出は、TAMP_INx（x = ピンインデックス）I/O、または範囲外のデバイス状態を検出する内部モニタでのイベントに対して行います。

タンパ検出は次の目的で設定できます。

- バックアップレジスタおよび表 637 : TAMP相互接続に示したSRAMまたはペリフェラルに格納されたその他のデバイス機密情報を消去
- バックアップレジスタおよび表 637 : TAMP相互接続に示したSRAMまたはペリフェラルに格納されたその他のデバイス機密情報への読出し/書込みアクセスをブロック
- 低消費電力モードからのウェイクアップのための割込みを生成
- 低消費電力タイマのためのハードウェアトリガ、またはRTCタイムスタンプイベントを生成

外部I/Oタンパ検出サポートは次の2種類のメイン設定に対応しています。

- パッシブモード : TAMP_INx I/Oが監視され、タンパはエッジまたはレベルで検出されます。
- アクティブモード : TAMP_INx（x = ピンインデックス）がTAMP_OUTy（y = ピンインデックス）と継続的に比較され、オープン/ショートが検出されます。

誤検出を防止するため、外部タンパ検出にデジタルフィルタを適用できます。また、潜在モードで各タンパソースを設定し、タンパ検出時にハードウェアによって機密情報の消去が開始されないようにすることができます。機密情報の消去は、ソフトウェアチェック後に、ソフトウェアによって開始することができます。

64.4.10 TAMPバックアップレジスタとその他のデバイス機密情報の消去

バックアップレジスタ（TAMP_BKPxR）は、システムリセットや STANDBYモードからのウェイクアップではリセットされません。

バックアップレジスタとその他のデバイス機密情報は、対応するマスクが設定されている（TAMP_CR2レジスタでTAMPxMSK=1）場合はリセットされません。

注 : **バックアップレジスタは、Flashの読出し保護がレベル1からレベル0に変わったときにも消去されます。**

タンパ検出 – 確認済みモード

確認済みモードは、TAMP_CR2レジスタでTAMPxNOER = 0がセットされているときに、TAMPx（外部タンパx）のために選択されます。確認済みモードは、TAMP_CR3レジスタでTAMPxNOER = 0がセットされているときに、ITAMPx（内部タンパx）のために選択されます。確認済みモードでのタンパ検出の影響は、表 637 : TAMP相互接続のtamp_confirmed およびtamp_confirmed_ercfgx 信号の項を参照してください。

このモードは、タンパの検出時にデバイス機密情報を自動的に消去するために選択されます。

タンパ検出 – 潜在モード

潜在モードは、TAMP_CR2レジスタでTAMPxNOER = 1がセットされているときに、TAMPx（外部タンパx）のために選択されます。潜在タンパモードは、TAMP_CR3レジスタでITAMPxNOER = 1がセットされているときに、ITAMPx（内部タンパx）のために選択されます。潜在モードでのタンパ検出の影響は、表 637 : TAMP相互接続のtamp_potential およびtamp_potential_ercfgx 信号の項を参照してください。

このモードを選択すると、タンパの検出時に一部のデバイス機密情報が完全に消去されて復旧できなくなるのを防止できます。このモードでは、該当するタンパイベントの検出時に、一部のデバイス機

密情報は消去されません。また、潜在モードでタンパ検出フラグがセットさせるとすぐに、TAMP_SCRレジスタで該当するクリアフラグを設定することでこのフラグがクリアされるまで、これらのデバイス機密情報への読出し/書込みアクセスはブロックされます。そのため、真のタンパと誤検出を区別するためのチェックが行われ、潜在タンパが真のタンパであると確認された場合に、機密情報の消去が開始されます。TAMP_CR2のBKERASEビットをセットすると、ソフトウェアによりデバイス機密情報が消去されます。

潜在タンパから確認済みタンパのタイムアウトまで

一部の内部タンパでは、別のタンパフラグがセットされているときに独立したウォッチドッグリセットが発生すると、タンパイベントが生成されます（表 637 : TAMP相互接続を参照）。ITAMPxNOER=0により、IWDGタンパを設定する必要があります。これにより、以前のタンパイベントが潜在モードであった場合に、タイムアウト後にハードウェアによって強制的にデバイス機密情報が消去されます。潜在タンパイベントに続いてソフトウェア決定の前にウォッチドッグリセットが発生する場合、これは「潜在タンパ」から「確認済みタンパ」への移行に相当します。

デバイスリソース保護設定

デバイスリソースを設定して、タンパ検出によって保護されるデバイス機密事項のリストに組み込むことができます。

TAMP_ERCFGRに ERCFGz = 0がセットされている場合、ERCFGzに関連付けられたデバイスリソースはTAMPペリフェラルによって保護されません。

- タンパ検出の影響を受けません（確認済みモードでも潜在モードでも）。
- BKERASEソフトウェアコマンドの影響を受けません。
- BKBLOCKソフトウェアコマンドの影響を受けません。

TAMP_ERCFGRに ERCFGz = 1がセットされている場合、ERCFGzに関連付けられたデバイスリソースはTAMPペリフェラルによって保護されます。

- 表 637 : TAMP相互接続のtamp_confirmed_ercfgz信号の項に示すように、確認済みタンパ検出とBKERASEソフトウェアコマンドの影響を受けます。
- 表 637 : TAMP 相互接続の tamp_potential_ercfgz 信号の項に示すように、潜在タンパ検出とBKBLOCKソフトウェアコマンドの影響を受けます。

表638.デバイスリソースxのタンパ保護

-	潜在タンパまたはBKBLOCK	確認済みタンパまたはBKERASE
RPCFGx = 0	デバイスリソースxへの影響なし	デバイスリソースxへの影響なし
RPCFGx = 1	tamp_potential_ercfgxの項に示すように、デバイス機密情報xを保護(1)	tamp_confirmed_ercfgx (1) の項に示すように、デバイス機密情報xを保護

1. 表 637 : TAMP相互接続を参照してください。

ソフトウェアによりデバイス機密情報へのアクセスをブロック

デフォルトでは、タンパイベントフラグが検出されなければ、アプリケーションによってデバイス機密情報にアクセスできます。タンパフラグがセットされている場合は、デバイス機密情報にはアクセスできません。

ソフトウェアによりデバイス機密情報へのアクセスをブロックするには、TAMP_CR2 レジスタのBKBLOCKビットをセットします。デバイス機密情報にアクセスできるのは、BKBLOCK = 0で、タンパフラグがセットされていない場合のみです。

64.4.11 タンパ検出設定と初期化

各入力は、TAMP_CRレジスタにおいて該当するTAMPxEビットを1にセットすることにより有効にできます。

各TAMP_INxタンパ検出入力は、TAMP_SRレジスタのフラグTAMPxFに関連付けられています。

TAMP_IERレジスタのTAMPxIEビットをセットすることにより、タンパ検出イベント発生時に割込みが生成されます（TAMPxFがセットされている場合）。該当するTAMPxMSKがセットされている場合、TAMPxIEをセットすることはできません。

タンパイベント時のトリガ出力生成

タンパイベント検出は、低消費電力タイマによるトリガ入力として使用できます。

TAMP_CRレジスタでTAMPxMSKビットがクリアされている場合、同じピンで新しいタンパ検出ができるようにするには、ソフトウェアによってTAMPxFフラグをクリアする必要があります。

TAMPxMSKビットがセットされている場合、TAMPxFフラグはマスクされ、TAMP_SRレジスタでクリアされたままとなります。この設定を使用することで、システムのウェイクアップによりTAMPxFをクリアしなくても、STOPモードで低消費電力タイマを自動的にトリガできます。この場合、バックアップレジスタはクリアされません。

この機能は、タンパが タンパ入力でのフィルタを使ったレベル検出（パッシブモード）モードで設定された場合にのみ利用可能です（TAMPFLT ≠ 00、およびアクティブモードが非選択の場合）。

タンパイベント時のタイムスタンプ

RTC_CRでTAMPTSを1にセットすると、すべての内部または外部タンパイベントがタイムスタンプを発生させるようになります。タンパイベントによりタイムスタンプが発生する場合、通常のタイムスタンプイベント発生時と同様にTSFビットまたはTSOVFビットがRTC_SRでセットされます。

注： **TSFは、TAMP x Fフラグから最大でck_apreの3サイクル後にセットされます。RTCCLKが停止した場合、TSFはセットされません（RTCCLKの再起動時にセットされます）。**

注： **TSFの立上りの前にTAMPxFがクリアされた場合、TSFはセットされません。結果的に、TAMPTS = 1 の場合、ソフトウェアではタンパフラグをクリアする前にタイムスタンプフラグを待つか、TAMP割込みルーチンでRTCカウンタ値を読み出す必要があります。**

タンパ入力でのエッジ検出（パッシブモード）

TAMPFLTビットが00の場合、該当するTAMPxTRGビットに応じて立上りエッジまたは立下りエッジが観測されると、TAMP_INxピンがタンパ検出イベントを生成します。エッジ検出を選択すると、TAMP_INx入力の内部プルアップ抵抗が無効になります。

注意： **TAMPFLT = 00およびTAMPxTRG = 0（立上りエッジ検出）のとき、タンパ入力がタンパ検出を有効にする前にすでにハイレベルになっている場合、タンパイベントがハードウェアによって検出される場合があります。**

タンパイベントが検出されクリアされた後に、バックアップレジスタ（TAMP_BKPxR）を再プログラムする場合には、事前に、TAMP_INxを無効にしてから再度有効にする（TAMPxEを1にセット）必要があります。これによって、TAMP_INxの入力値がタンパ検出を示している期間中に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、TAMP_INxの入力でのレベル検出に相当します。

注： **タンパ検出は、V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不必要なリセットを避けるには、TAMPxが設定されているピンを外部で適切な信号レベルに接続しておく必要があります。**

タンパ入力でのフィルタを使ったレベル検出（パッシブモード）

フィルタを使ったレベル検出は、TAMPFLTを0以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLTに応じて）2、4または8回のいずれかの連続したサンプルがTAMPxTRGビットで指定するレベルで観測されたときに生成されます。

TAMP_INx入力は、TAMPPUDISが1にセットされて無効な状態になっていない限り、その状態がサンプリングされる前にI/Oの内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間はTAMPPRCHビットによって決定され、TAMP_INx入力により大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQを使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： **プルアップ抵抗の電気的特性については、マイクロコントローラデータシートを参照してください。**

アクティブタンパの検出

TAMPxAMビットがTAMP_ATCRでセットされている場合、タンパイベントはアクティブモードに設定され、これはTAMP_OUTyピンとTAMP_INxピンの比較に基づきます。デフォルト（ATOSHARE = 0）では、TAMP_INxとTAMP_OUTx（y = x）が比較されます。ATOSHAREビットがセットされている場合、同じ出力を複数のタンパ入力に使用できます。TAMP_OUTy機能は、ATOSHAREおよびATOSELxビットにより、アクティブなタンパ入力TAMP_INx（TAMPxEN = TAMPxAM = 1）との比較のために選択されるとすぐに、I/Oで有効になります。TAMP_ATCRx（x = 1、2）レジスタのATOSHAREビットおよびATOSELビットの説明を参照してください。

2 CK_ATPERサイクルごと（CK_ATPER = 2^{ATPER} x CK_ATPRE）に、TAMP_OUTy出力ピンは疑似乱数発生器（PRNG）によって生成される値を出力します。この値を出力した後、TAMP_OUTyピンは、1 CK_ATPERサイクル後に逆の値を出力します。

表639.アクティブタンパ出力の変更周期

ATCKSEL[3:0]	CK_ATPRE周波数	ATPER[2:0]	タンパ出力の変更（CK_ATPER）周波数	タンパ出力の変更周期(1)（ms）
0x0	f_{RTCCLK}	0x0	f_{RTCCLK}	0.030
		0x1	$f_{RTCCLK}/2$	0.061
		0x2	$f_{RTCCLK}/4$	0.122
		0x3	$f_{RTCCLK}/8$	0.244
		0x4	$f_{RTCCLK}/16$	0.488
		0x5	$f_{RTCCLK}/32$	0.977
		0x6	$f_{RTCCLK}/64$	1.953
		0x7	$f_{RTCCLK}/128$	3.906
...	...	...	...	...
0x7	$f_{RTCCLK}/128$	0x0	$f_{RTCCLK}/128$	3.906
		0x1	$f_{RTCCLK}/256$	7.8125
		0x2	$f_{RTCCLK}/512$	15.625
		0x3	$f_{RTCCLK}/1024$	31.250
		0x4	$f_{RTCCLK}/2048$	62.5
		0x5	$f_{RTCCLK}/4096$	125
		0x6	$f_{RTCCLK}/8192$	250
		0x7	$f_{RTCCLK}/16384$	500

表639.アクティブタンパ出力の変更周期 （続き）

ATCKSEL[3:0]	CK_ATPRE周波数	ATPER[2:0]	タンパ出力の変更（CK_ATPER）周波数	タンパ出力の変更周期(1)（ms）
0xB(2)	$f_{RTCCLK}/2048$(3)	0x0	$f_{RTCCLK}/2048$	62.5
		0x1	$f_{RTCCLK}/4096$	125
		0x2	$f_{RTCCLK}/8192$	250
		0x3	$f_{RTCCLK}/16384$	500
		0x4	$f_{RTCCLK}/32768$	1000
		0x5	$f_{RTCCLK}/65536$	2000
		0x6	$f_{RTCCLK}/131072$	4000
		0x7	$f_{RTCCLK}/262144$	8000

1. f_{RTCCLK} = 32768 Hzと仮定します。
2. これらの値は、アクティブタンパプリスケーラ拡張がサポートされている場合にのみサポートされます。セクション 64.3：TAMPの実装を参照してください。
3. この設定では、(PREDIV_A+1) = 128かつ (PREDIV_S+1) が16の倍数であることが前提となります。

PRNGは、選択したタンパ出力によってその数に応じた異なる周波数で使用されます。選択した出力の数は、TAMPxAM、TAMPxE、ATOSEL、ATOSHAREによります。

- 出力が1つだけ選択されている場合：PRNGは16 CK_ATPER周期ごとに使用されます。
- 出力が2つ選択されている場合：PRNGは8 CK_ATPER周期ごとに使用されます。
- 出力が3つまたは4つ選択されている場合：PRNGは4 CK_ATPER周期ごとに使用されます。
- 出力が5つ以上選択されている場合：PRNGは2 CK_ATPER周期ごとに使用されます。

PRNGは新しい値の出力に最低9 CK_ATPREサイクルを必要とします。そのため、正しく機能するための最低ATPER値を以下の表に示します。

表640.最低ATPER値

選択した出力数	最低ATPER
1	0
2	1
3または4	2
5以上	3

TAMP_INxピンはTAMP_OUTyピンに外部接続されます。CK_ATPREサイクルごとに、TAMP_OUTy出力値とTAMP_INx受信値が比較されます。比較して不一致となった場合、TAMPxFビットがTAMP_SRレジスタでセットされます。

たとえば、アクティブモードで TAMP1 および TAMP2 の両方を設定して有効にすることで、TAMP_OUT1をTAMP_IN1とTAMP_IN2との比較に使用できます（ATOSHARE = 1、ATOSEL1 = 000、ATOSEL2 = 000）。

アクティブタンパは、FLTEN = 1の場合に入力フィルタと組み合わせることができます。この場合、連続する4つの比較サンプルで2つの比較が偽となった場合のみタンパが検出されます。

図769. アクティブタンパフィルタ

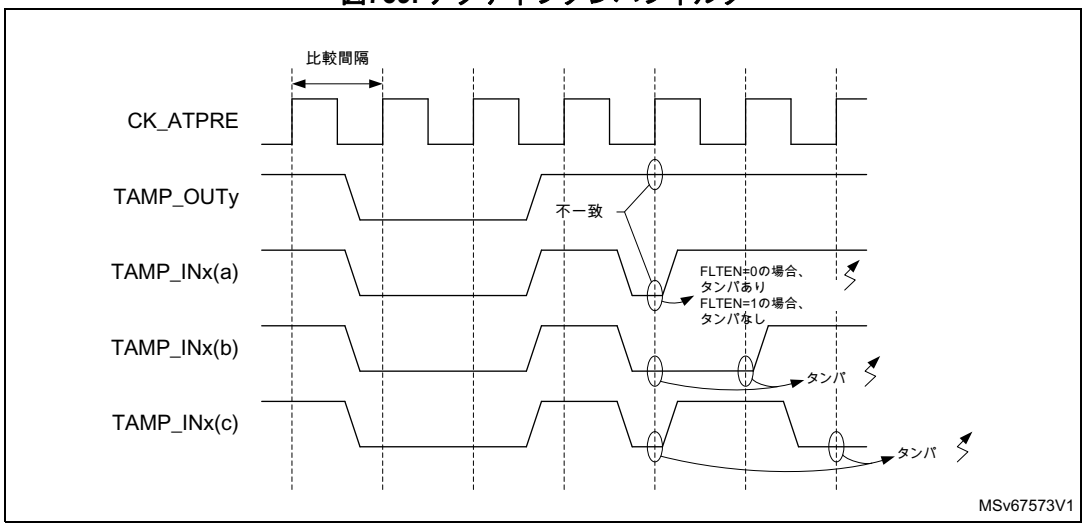


図769に示すように、FLTEN = 0の場合、TAMP_OUTy出力と関連するTAMP_INx入力の間に不一致があると、後者のサンプリング時にタンパが生成されます。これが3つの例（a）、（b）、（c）に示した状態です。

FLTEN = 1の場合、例（a）では連続する4回の比較で不一致が1回しか検出されないため、タンパが生成されません。例（b）では、連続する2回の不一致が検出されているため、タンパが生成されます。例（c）では、連続するサンプルで不一致は発生していなくても、連続する4回の比較で2回の不一致が発生しているため、ここでもタンパが生成されます。

FLTEN = 1をセットすると、TAMP_ATCR1レジスタのATCKSELフィールドでプログラムされるCK_ATPREの1周期よりも短い単一パルスを無視することにより、TAMP_INx入力でのグリッチ、バウンス、または一次的な状態による望ましくないタンパの検出を回避できます。f_{RTCCLK} = 32.768 kHzと仮定して、設定可能な各ATCKSELについて、最小フィルタパルス幅を表641に示します。

表641.アクティブタンパのフィルタパルス時間

ATCKSEL[3:0]	CK_ATPRE周波数	最小フィルタパルス幅（ms）
0x0	f_{RTCCLK}	0.030
0x1	$f_{RTCCLK}/2$	0.061
0x2	$f_{RTCCLK}/4$	0.122
0x3	$f_{RTCCLK}/8$	0.244
0x4	$f_{RTCCLK}/16$	0.488
0x5	$f_{RTCCLK}/32$	0.977
0x6	$f_{RTCCLK}/64$	1.953
0x7	$f_{RTCCLK}/128$	3.906
0xB(1)	$f_{RTCCLK}/2048$	62.500(2)

1. これらの値は、アクティブタンパプリスケーラ拡張がサポートされている場合にのみサポートされます。セクション 64.3：TAMPの実装を参照してください。
2. この設定では、(PREDIV_A+1) = 128かつ (PREDIV_S+1) が16の倍数であることが前提となります。

注：　複数のパルスがCK_ATPREの1周期分より短い場合でも、連続する4回の比較で2回の不一致が発生した場合、タンパが生じることがあります。

注意： **ATCKSEL[3] = 1の場合、RTC初期化モードが開始されると、CK_ATPREおよびCK_ATPERクロックが停止します。そのため、INITモードが終了するまで、TAMP_OUTyピンは停止します。**

セクション ：カレンダの初期化と設定を参照してください。

ATCKSEL[3] = 1と一緒にRTCバイナリモードを使用する場合は、RTCアラームAサブセカンドレジスタ（RTC_ALRMASSR）、RTCアラームBサブセカンドレジスタ（RTC_ALRMBSSR）、RTCアラームAバイナリモードレジスタ（RTC_ALRABINR）、RTCアラームBバイナリモードレジスタ（RTC_ALRBBINR）も参照してください。

注意： **注意：ATCKSEL[3] = 1 の場合、カレンダオーバーフローではアクティブタンパ検出は機能しなくなります。タンパ保護を行うには、カレンダオーバーフローで内部タンパ5を有効化する必要があります。**

疑似乱数発生器は、シードで初期化する必要があります。これはTAMP_ATSEEDRレジスタに連続して4つの32bit乱数値を書き込むことによって行います。シードをプログラムすると、自動的にPRNGに送信されます。新しいシードが転送され、PRNGによって処理されている場合は、TAMP_ATORのSEEDFビットがセットされ、TAMP APBクロックはオフにできません。処理時間は、4つ目のシードが書き込まれてから最大184 APBクロックサイクルです。そのため、ユーザは新しいシードを書き込んでから、SEEDFがクリアされるのを待ち、低消費電力モードに移行する必要があります。

アクティブタンパ出力は、1つ目のシードが書き込まれて処理を完了した後でのみアクティブになります。その後、アクティブタンパ動作中は、新しいシードを書き込んで処理できます。

アクティブタンパの初期化

ここに、システムリセット後にソフトウェアでアクティブタンパを初期化する手順を示します。

TAMP_ATORレジスタのINITSを読み出します。

- INITS = 0x0（初期化されていない場合）：
  a) TAMP_ATCRを書き込んでアクティブタンパクロック、フィルタ、出力共有（ある場合）、アクティブモードを設定します。
  b) TAMP_CR1を書き込んでタンパを有効にします（すべての必要なタンパを同じ書込み処理で有効にする必要があります）。
  c) TAMP_ATSEEDRにSEEDを4回書き込みます。
  d) TAMP_ATORのSEEDF = 0まで待ちます。これでバックアップレジスタがアクティブタンパで保護されます。
- INITS = 0x1（すでに初期化されている場合）：
  初期化しません。ランダム性を高めるには、新しいSEEDを定期的に供給する必要があります。新しいSEEDが供給されると、SEEDF = 0になるまで待ってから、TAMP APBクロックをオフに切り替える低消費電力モードに移行します。
- タンパをソフトウェアで無効にして、後で再び有効にする場合、タンパを有効にした後でSEEDを書き込む必要があります。
  a) TAMP_CR1を書き込んでタンパを有効にします（すべての必要なタンパを同じ書込み処理で有効にする必要があります）。
  b) TAMP_ATSEEDRにSEEDを4回書き込みます。
  c) TAMP_ATORのSEEDF = 0まで待ちます。これでバックアップレジスタがアクティブタンパで保護されます。

64.5 TAMP低消費電力モード

表642.低消費電力モードがTAMPに与える影響

モード	説明
SLEEP	影響はありません。 TAMP割込みによって、デバイスはSLEEPモードから復帰します。
STOP	クロックソースがLSEまたはLSIである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、すべての機能に影響はありません。 タンパイベントによって、デバイスはSTOPモードを終了します。
STANDBY	クロックソースがLSEまたはLSIである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、すべての機能に影響はありません。 タンパイベントによって、デバイスはSTANDBYモードを終了します。
SHUTDOWN	クロックソースがLSEである場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、すべての機能に影響はありません。タンパイベントによって、デバイスはSHUTDOWNモードを終了します。

表643.TAMPピン機能（モード共通）

ピン名	すべての低消費電力モードで機能するか	V_{BAT} モードで使用可能な機能
TAMP_IN[8:1]	可能	あり
TAMP_OUT[8:1]	可能	あり

64.6 TAMP割込み

割込みチャネルは、マスク済み割込みステータスレジスタ、またはセキュリティモード設定に応じたセキュアマスク済み割込みステータスレジスタで設定されます（TAMPSEC）。

表644.割込みリクエスト

項目（割込みの略称）	割込みイベント	イベント・フラグ(1)	イネーブル制御ビット	割込みのクリア方法	低消費電力モードの終了
TAMP	タンパx(2)	TAMPxF	TAMPxIE	CTAMPxFに1を書き込む	あり(3)
	内部タンパy (2)	ITAMPyF	ITAMPyIE	CITAMPyFに1を書き込む	あり (3)

1. イベントフラグは、TAMP_SRレジスタ内です。
2. タンパおよび内部タンパイベントの数は製品によります。
3. 低消費電力モードで利用可能な機能については、表 642：低消費電力モードがTAMPに与える影響を参照してください。

64.7 TAMPレジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルのセクション1.2を参照してください。ペリフェラルレジスタには、ワード（32bit）単位でアクセスすることができます。

64.7.1 TAMP制御レジスタ1（TAMP_CR1）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x00

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ITAMP13E	ITAMP12E	ITAMP11E	Res.	ITAMP9E	ITAMP8E	ITAMP7E	ITAMP6E	ITAMP5E	Res.	ITAMP3E	ITAMP2E	ITAMP1E
			rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8E	TAMP7E	TAMP6E	TAMP5E	TAMP4E	TAMP3E	TAMP2E	TAMP1E
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ITAMP13E**：内部タンパ13イネーブル
0：内部タンパ13は無効です。
1：内部タンパ13は有効です。

ビット27 **ITAMP12E**：内部タンパ12イネーブル
0：内部タンパ12は無効です。
1：内部タンパ12は有効です。

ビット26 **ITAMP11E**：内部タンパ11イネーブル
0：内部タンパ11は無効です。
1：内部タンパ11は有効です。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ITAMP9E**：内部タンパ9イネーブル
0：内部タンパ9は無効です。
1：内部タンパ9は有効です。

ビット23 **ITAMP8E**：内部タンパ8イネーブル
0：内部タンパ8は無効です。
1：内部タンパ8は有効です。

ビット22 **ITAMP7E**：内部タンパ7イネーブル
0：内部タンパ7は無効です。
1：内部タンパ7は有効です。

ビット21 **ITAMP6E**：内部タンパ6イネーブル
0：内部タンパ6は無効です。
1：内部タンパ6は有効です。

ビット20 **ITAMP5E**：内部タンパ5イネーブル
0：内部タンパ5は無効です。
1：内部タンパ5は有効です。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ITAMP3E**：内部タンパ3イネーブル
0：内部タンパ3は無効です。
1：内部タンパ3は有効です。

ビット17 **ITAMP2E**：内部タンパ2イネーブル
0：内部タンパ2は無効です。
1：内部タンパ2は有効です。

ビット16 **ITAMP1E**：内部タンパ1イネーブル
0：内部タンパ1は無効です。
1：内部タンパ1は有効です。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8E**：TAMP_IN8のタンパ検出イネーブル (1)
0：TAMP_IN8のタンパ検出は無効です。
1：TAMP_IN8のタンパ検出は有効です。

ビット6 **TAMP7E**：TAMP_IN7のタンパ検出イネーブル (1)
0：TAMP_IN7のタンパ検出は無効です。
1：TAMP_IN7のタンパ検出は有効です。

ビット5 **TAMP6E**：TAMP_IN6のタンパ検出イネーブル (1)
0：TAMP_IN6のタンパ検出は無効です。
1：TAMP_IN6のタンパ検出は有効です。

ビット4 **TAMP5E**：TAMP_IN5のタンパ検出イネーブル (1)
0：TAMP_IN5のタンパ検出は無効です。
1：TAMP_IN5のタンパ検出は有効です。

ビット3 **TAMP4E**：TAMP_IN4のタンパ検出イネーブル (1)
0：TAMP_IN4のタンパ検出は無効です。
1：TAMP_IN4のタンパ検出は有効です。

ビット2 **TAMP3E**：TAMP_IN3のタンパ検出イネーブル (1)
0：TAMP_IN3のタンパ検出は無効です。
1：TAMP_IN3のタンパ検出は有効です。

ビット1 **TAMP2E**：TAMP_IN2のタンパ検出イネーブル (1)
0：TAMP_IN2のタンパ検出は無効です。
1：TAMP_IN2のタンパ検出は有効です。

ビット0 **TAMP1E**：TAMP_IN1のタンパ検出イネーブル(1)
0：TAMP_IN1のタンパ検出は無効です。
1：TAMP_IN1のタンパ検出は有効です。

1. タンパ検出モード（TAMP_FLTCR、TAMP_ATCR1、TAMP_ATCR2 レジスタおよびTAMP_CR2のTAMPxTRGビットで選択）は、タンパ検出を有効化する前に設定する必要があります。

64.7.2 TAMP制御レジスタ2（TAMP_CR2）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x04

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TAMP8 TRG	TAMP7 TRG	TAMP6 TRG	TAMP5 TRG	TAMP4 TRG	TAMP3 TRG	TAMP2 TRG	TAMP1 TRG	BK ERASE	BK BLOCK	Res.	Res.	Res.	TAMP3 MSK	TAMP2 MSK	TAMP1 MSK
rw	rw	rw	rw	rw	rw	rw	rw	w	rw				rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8 NOER	TAMP7 NOER	TAMP6 NOER	TAMP5 NOER	TAMP4 NOER	TAMP3 NOER	TAMP2 NOER	TAMP1 NOER
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **TAMP8TRG**：タンパ8の入力のアクティブレベル（アクティブモード無効）

0 : TAMPFLT ≠ 00の場合、タンパ8の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ8の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ8の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ8の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット30 **TAMP7TRG**：タンパ7の入力のアクティブレベル（アクティブモード無効）

0 : TAMPFLT ≠ 00の場合、タンパ7の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ7の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ7の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ7の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット29 **TAMP6TRG**：タンパ6の入力のアクティブレベル（アクティブモード無効）

0 : TAMPFLT ≠ 00の場合、タンパ6の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ6の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ6の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ6の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット28 **TAMP5TRG**：タンパ5の入力のアクティブレベル（アクティブモード無効）

0 : TAMPFLT ≠ 00の場合、タンパ5の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ5の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ5の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ5の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット27 **TAMP4TRG**：タンパ4の入力のアクティブレベル（アクティブモード無効）

0 : TAMPFLT ≠ 00の場合、タンパ4の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ4の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ4の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ4の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット26 **TAMP3TRG**：タンパ3の入力のアクティブレベル

0 : TAMPFLT ≠ 00の場合、タンパ3の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ3の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 : TAMPFLT ≠ 00の場合、タンパ3の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ3の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット25 **TAMP2TRG**：タンパ2の入力のアクティブレベル

0 :TAMPFLT ≠ 00の場合、タンパ2の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ2の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 :TAMPFLT ≠ 00の場合、タンパ2の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ2の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット24 **TAMP1TRG**：タンパ1の入力のアクティブレベル

0 :TAMPFLT ≠ 00の場合、タンパ1の入力がローのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ1の入力が立上りエッジのとき、タンパ検出イベントがトリガされます。

1 :TAMPFLT ≠ 00の場合、タンパ1の入力がハイのままのとき、タンパ検出イベントがトリガされます。
TAMPFLT = 00の場合、タンパ1の入力が立下りエッジのとき、タンパ検出イベントがトリガされます。

ビット23 **BKERASE**：バックアップレジスタとデバイス機密情報 (1) の消去

このビットに 「1」を書き込むと、バックアップレジスタとデバイス機密情報(1)がリセットされます。
0を書き込んでも、ビットの値は変化しません。このビットは常に0として読み出されます。

ビット22 **BKBLOCK**：バックアップレジスタとデバイス機密情報 (1) のアクセスのブロック

0：タンパフラグがセットされていない場合、バックアップレジスタとデバイス機密情報 (1) にアクセスできます。

1：バックアップレジスタとデバイス機密情報 (1) にアクセスできません。

ビット21:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **TAMP3MSK**：タンパ3マスク

0：タンパ3イベントによってトリガイベントが生成され、TAMP3Fはソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ3イベントによってトリガイベントが生成されます。TAMP3Fはマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタとデバイス機密情報 (1) は消去されません。
TAMP3MSKがセットされている場合、タンパ3割込みを有効にできません。

ビット17 **TAMP2MSK**：タンパ2マスク

0：タンパ2イベントによってトリガイベントが生成され、TAMP2Fはソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ2イベントによってトリガイベントが生成されます。TAMP2Fはマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタとデバイス機密情報 (1) は消去されません。
TAMP2MSKがセットされている場合、タンパ2割込みを有効にできません。

ビット16 **TAMP1MSK**：タンパ1マスク

0：タンパ1イベントによってトリガイベントが生成され、TAMP1Fはソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ1イベントによってトリガイベントが生成されます。TAMP1Fはマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタとデバイス機密情報 (1) は消去されません。
TAMP1MSKがセットされている場合、タンパ1割込みを有効にできません。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8NOER**：タンパ8消去なし

0：タンパ8イベント検出は確認済みモードです (1)。

1：タンパ8イベント検出は潜在モードです (2)。

ビット6 **TAMP7NOER**：タンパ7消去なし

0：タンパ7イベント検出は確認済みモードです (1)。

1：タンパ7イベント検出は潜在モードです (2)。

ビット5 **TAMP6NOER**：タンパ6消去なし

0：タンパ6イベント検出は確認済みモードです (1)。

1：タンパ6イベント検出は潜在モードです (2)。

ビット4 **TAMP5NOER**：タンパ5消去なし
0：タンパ5イベント検出は確認済みモードです (1)。
1：タンパ5イベント検出は潜在モードです (2)。

ビット3 **TAMP4NOER**：タンパ4消去なし
0：タンパ4イベント検出は確認済みモードです (1)。
1：タンパ4イベント検出は潜在モードです (2)。

ビット2 **TAMP3NOER**：タンパ3消去なし
0：タンパ3イベント検出は確認済みモードです (1)。
1：タンパ3イベント検出は潜在モードです (2)。

ビット1 **TAMP2NOER**：タンパ2消去なし
0：タンパ2イベント検出は確認済みモードです (1)。
1：タンパ2イベント検出は潜在モードです (2)。

ビット0 **TAMP1NOER**：タンパ1消去なし
0：タンパ1イベント検出は確認済みモードです(1)。
1：タンパ1イベント検出は潜在モードです(2)。

1. 確認済みモードでのタンパ検出の影響は、表 637 : TAMP相互接続のtamp_confirmed およびtamp_confirmed_ercfgx 信号の項を参照してください。
2. 潜在モードでのタンパ検出の影響は、表 637 : TAMP相互接続のtamp_potential およびtamp_potential_ercfgx 信号の項を参照してください。

64.7.3 TAMP制御レジスタ3（TAMP_CR3）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モード を参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モード を参照してください。

アドレスオフセット：0x08

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ITAMP 13NOER	ITAMP 12NOER	ITAMP 11NOER	Res.	ITAMP9 NOER	ITAMP8 NOER	ITAMP7 NOER	ITAMP6 NOER	ITAMP5 NOER	Res.	ITAMP3 NOER	ITAMP2 NOER	ITAMP1 NOER
			rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ITAMP13NOER**：内部タンパ13 消去なし
0：内部タンパ13イベント検出は確認済みモードです (1)。
1：内部タンパ13イベント検出は潜在モードです (2)。

ビット11 **ITAMP12NOER**：内部タンパ12 消去なし
0：内部タンパ12イベント検出は確認済みモードです (1)。
1：内部タンパ12イベント検出は潜在モードです (2)。

ビット10 **ITAMP11NOER**：内部タンパ11 消去なし
0：内部タンパ11イベント検出は確認済みモードです (1)。
1：内部タンパ11イベント検出は潜在モードです (2)。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **ITAMP9NOER**：内部タンパ9 消去なし
0：内部タンパ9イベント検出は確認済みモードです (1)。
1：内部タンパ9イベント検出は潜在モードです (2)。

ビット7 **ITAMP8NOER**：内部タンパ8 消去なし
0：内部タンパ8イベント検出は確認済みモードです (1)。
1：内部タンパ8イベント検出は潜在モードです (2)。

ビット6 **ITAMP7NOER**：内部タンパ7 消去なし
0：内部タンパ7イベント検出は確認済みモードです (1)。
1：内部タンパ7イベント検出は潜在モードです (2)。

ビット5 **ITAMP6NOER**：内部タンパ6 消去なし
0：内部タンパ6イベント検出は確認済みモードです(1)。
1：内部タンパ6イベント検出は潜在モードです(2)。

ビット4 **ITAMP5NOER**：内部タンパ5 消去なし
0：内部タンパ5イベント検出は確認済みモードです (1)。
1：内部タンパ5イベント検出は潜在モードです (2)。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **ITAMP3NOER**：内部タンパ3 消去なし
0：内部タンパ3イベント検出は確認済みモードです (1)。
1：内部タンパ3イベント検出は潜在モードです (2)。

ビット1 **ITAMP2NOER**：内部タンパ2 消去なし
0：内部タンパ2イベント検出は確認済みモードです (1)。
1：内部タンパ2イベント検出は潜在モードです (2)。

ビット0 **ITAMP1NOER**：内部タンパ1 消去なし
0：内部タンパ1イベント検出は確認済みモードです (1)。
1：内部タンパ1イベント検出は潜在モードです (2)。

1. 確認済みモードでの内部タンパ検出の影響は、表 637 : TAMP相互接続のtamp_confirmed およびtamp_confirmed_ercfgx 信号の項を参照してください。
2. 潜在モードでの内部タンパ検出の影響は、表 637 : TAMP相互接続のtamp_potential およびtamp_potential_ercfgx 信号の項を参照してください。

64.7.4 TAMPフィルタ制御レジスタ（TAMP_FLTCR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x0C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP PUDIS	TAMPPRCH [1:0]		TAMPFLT [1:0]		TAMPFREQ [2:0]		
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMPPUDIS**：TAMP_INxプルアップディスエーブル

このビットにより、毎回のサンプリング前に各TAMPxピンをプリチャージするかどうか決定します。
0：サンプリング前にTAMP_INxピンをプリチャージします（内部プルアップを有効化）。
1：TAMP_INxピンのプリチャージを無効化します。

ビット6:5 **TAMPPRCH[1:0]**：TAMP_INxプリチャージ持続時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCHは、各TAMP_INx入力に対して有効です。
0x0：1 RTCCLKサイクル
0x1：2 RTCCLKサイクル
0x2：4 RTCCLKサイクル
0x3：8 RTCCLKサイクル

ビット4:3 **TAMPFLT[1:0]**：TAMP_INxフィルタカウント

これらのビットにより、タンパイベントをアクティブにするのに必要な指定のレベル（TAMP*TRG）での連続サンプリングの数を決定します。TAMPFLTは、各TAMP_INx入力に対して有効です。
0x0：TAMP_INx入力がアクティブレベル（TAMP_INx入力における内部プルアップなし）に変化するときのエッジで、タンパイベントがアクティブになります。
0x1:アクティブレベルでの連続した2回のサンプリングの後、タンパイベントがアクティブになります。
0x2:アクティブレベルでの連続した4回のサンプリングの後、タンパイベントがアクティブになります。
0x3:アクティブレベルでの連続した8回のサンプリングの後、タンパイベントがアクティブになります。

ビット2:0 **TAMPFREQ[2:0]**：タンパサンプリング周波数

これらのビットにより、各TAMP_INx入力がサンプリングされる周波数を決定します。
0x0：RTCCLK / 32768（RTCCLK = 32768Hzの場合1Hz）
0x1：RTCCLK / 16384（RTCCLK = 32768Hzの場合2Hz）
0x2：RTCCLK / 8192（RTCCLK = 32768Hzの場合4Hz）
0x3：RTCCLK / 4096（RTCCLK = 32768Hzの場合8Hz）
0x4：RTCCLK / 2048（RTCCLK = 32768Hzの場合16Hz）
0x5：RTCCLK / 1024（RTCCLK = 32768Hzの場合32Hz）
0x6：RTCCLK / 512（RTCCLK = 32768Hzの場合64Hz）
0x7：RTCCLK / 256（RTCCLK = 32768Hzの場合128Hz）

注： このレジスタは、パッシブモードでのタンパ入力のみ関係します。

64.7.5 TAMPアクティブタンパ制御レジスタ1（TAMP_ATCR1）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モード を参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モード を参照してください。

アドレスオフセット：0x10

バックアップドメインリセット値：0x0007 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLTEN	ATO SHARE	Res.	Res.	Res.	ATPER[2:0]			Res.	Res.	Res.	Res.	ATCKSEL[3:0]			
rw	rw				rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATOSEL4 [1:0]		ATOSEL3 [1:0]		ATOSEL2 [1:0]		ATOSEL1 [1:0]		TAMP8 AM	TAMP7 AM	TAMP6 AM	TAMP5 AM	TAMP4 AM	TAMP3 AM	TAMP2 AM	TAMP1 AM
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **FLTEN**：アクティブタンパフィルタイネーブル

0：アクティブタンパフィルタは無効です。

1：アクティブタンパフィルタは有効です。連続した4つのサンプルのうち比較の不一致が2つ発生した場合にタンパイベントが検出されます。

ビット30 **ATOSHARE**：アクティブタンパ出力の共有

0：各アクティブタンパ入力TAMP_INiが、専用の出力TAMP_OUTiと比較されます。

1：各アクティブタンパ入力TAMP_INiが、ATOSELiビットで定義されたTAMPOUTSELiと比較されます。

ビット29:27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **ATPER[2:0]**：アクティブタンパ出力の変更周期

タンパ出力は、CK_ATPER = (2^{ATPER} x CK_ATPRE) サイクルごとに変化します。表 640：最低ATPER値を参照してください。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **ATCKSEL[3:0]**：アクティブタンパRTC非同期プリスケーラクロックの選択

これらのビットでRTC非同期プリスケーラステージ出力を選択します。選択されたクロックはCK_ATPREです。アクティブタンパプリスケーラ拡張が実装されていない場合、ATCKSEL[3] は予約済み、読出し専用、かつ0に固定されます。

0000：RTCCLKが選択されます。

0001：RTCCLK/2が選択されます。

0010：RTCCLK/4が選択されます。

0011：RTCCLK/8が選択されます。

0100：RTCCLK/16が選択されます。

0101：RTCCLK/32が選択されます。

0110：RTCCLK/64が選択されます。

0111：RTCCLK/128が選択されます。

1011：（PREDIV_A+1）= 128かつ（PREDIV_S+1）が16の倍数の場合、RTCCLK/2048が選択されます。この値は、アクティブタンパプリスケーラ拡張がサポートされている場合にのみサポートされます。セクション 64.3：TAMPの実装を参照してください。

その他：予約済み

注：　これらのビットに書き込めるのは、アクティブタンパがすべて無効なときだけです。書込み保護は、すべてのアクティブタンパが無効になってから最大1.5 CK_ATPREサイクル保持されます。

ビット15:14 **ATOSEL4[1:0]**：アクティブタンパ共有出力4の選択
00: TAMPOUTSEL4 = TAMP_OUT1
01: TAMPOUTSEL4 = TAMP_OUT2
10: TAMPOUTSEL4 = TAMP_OUT3
11: TAMPOUTSEL4 = TAMP_OUT4
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット13:12 **ATOSEL3[1:0]**：アクティブタンパ共有出力3の選択
00: TAMPOUTSEL3 = TAMP_OUT1
01: TAMPOUTSEL3 = TAMP_OUT2
10: TAMPOUTSEL3 = TAMP_OUT3
11: TAMPOUTSEL3 = TAMP_OUT4
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット11:10 **ATOSEL2[1:0]**：アクティブタンパ共有出力2の選択
00: TAMPOUTSEL2 = TAMP_OUT1
01: TAMPOUTSEL2 = TAMP_OUT2
10: TAMPOUTSEL2 = TAMP_OUT3
11: TAMPOUTSEL2 = TAMP_OUT4
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット9:8 **ATOSEL1[1:0]**：アクティブタンパ共有出力1の選択
00: TAMPOUTSEL1 = TAMP_OUT1
01: TAMPOUTSEL1 = TAMP_OUT2
10: TAMPOUTSEL1 = TAMP_OUT3
11: TAMPOUTSEL1 = TAMP_OUT4
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット7 **TAMP8AM**：タンパ8アクティブモード
0：タンパ8検出モードはパッシブです。
1：タンパ8検出モードはアクティブです。

ビット6 **TAMP7AM**：タンパ7アクティブモード
0：タンパ7検出モードはパッシブです。
1：タンパ7検出モードはアクティブです。

ビット5 **TAMP6AM**：タンパ6アクティブモード
0：タンパ6検出モードはパッシブです。
1：タンパ6検出モードはアクティブです。

ビット4 **TAMP5AM**：タンパ5アクティブモード
0：タンパ5検出モードはパッシブです。
1：タンパ5検出モードはアクティブです。

ビット3 **TAMP4AM**：タンパ4アクティブモード
0：タンパ4検出モードはパッシブです。
1：タンパ4検出モードはアクティブです。

ビット2 **TAMP3AM**：タンパ3アクティブモード
0：タンパ3検出モードはパッシブです。
1：タンパ3検出モードはアクティブです。

ビット1 **TAMP2AM**：タンパ2アクティブモード

0：タンパ2検出モードはパッシブです。

1：タンパ2検出モードはアクティブです。

ビット0 **TAMP1AM**：タンパ1アクティブモード

0：タンパ1検出モードはパッシブです。

1：タンパ1検出モードはアクティブです。

注： **TAMPxAMビットがセットされている場合、対応するTAMPxEビットがTAMP_CR1レジスタですべてクリアされていない限り、このレジスタのアクティブタンパ設定を変更できません。**

アクティブモードに設定されたすべてのタンパは、（同じTAMP_CR1書込みで関連するすべてのTAMPxEをセットして）同時に有効にする必要があります。

アクティブモードに設定されたすべてのタンパは、（同じTAMP_CR1書込みで関連するすべてのTAMPxEをクリアして）同時に無効にする必要があります。

アクティブタンパを無効にしてから再び有効にするまで、最低CK_ATPRE 1周期を待つ必要があります。

64.7.6 TAMPアクティブタンパシードレジスタ（TAMP_ATSEEDR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モード を参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モード を参照してください。

アドレスオフセット：0x14

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEED[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEED[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **SEED[31:0]**：疑似乱数発生器のシード値

このレジスタは、128bitのシードをPRNGに供給する32bit値を4回書き込む必要があります。このレジスタに書き込むと、シード値が自動的にPRNGに送信されます。

64.7.7 TAMPアクティブタンパ出力レジスタ（TAMP_ATOR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モード を参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モード を参照してください。

アドレスオフセット：0x18

バックアップドメインリセット値：0x0000 0000

システムリセット：0にリセットされるSEEDFを除いて影響されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INITS	SEEDF	Res.	Res.	Res.	Res.	Res.	Res.	PRNG[7:0]							
r	r							r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **INITS**：アクティブタンパの初期化ステータス

このフラグは、PRNGが最初の128bitのシードを受け取ったときにハードウェアによってセットされ、有効にされたアクティブタンパが機能することを示します。このフラグは、アクティブタンパが無効になるとクリアされます。

ビット14 **SEEDF**：シードランニングフラグ

このフラグは、新しいシードがTAMP_ATSEEDRに書き込まれたときに、ハードウェアによってセットされます。PRNGがこの新しいシードを受け取ったときにハードウェアによってクリアされます。またシステムリセットによってクリアされます。SEEDFがセットされている場合、TAMP APBクロックはオフにしないでください。

ビット13:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PRNG[7:0]**：疑似乱数発生器の値

このフィールドは、PRNG出力の値を提供します。同期遅延による潜在的な不一致があるため、PRNGは最低2回読み出す必要があります。前の読出し値と一致する場合、その読出し値が正しいものとなります。

このフィールドは、APBがセキュアモードのときのみ読み出すことができます。

64.7.8 TAMPアクティブタンパ制御レジスタ2（TAMP_ATCR2）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モード を参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モード を参照してください。

アドレスオフセット：0x1C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ATOSEL8 [2:0]			ATOSEL7 [2:0]			ATOSEL6 [2:0]			ATOSEL5 [2:0]			ATOSEL4 [2:0]			ATOSEL3 [2]
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATOSEL3 [1:0]		ATOSEL2 [2:0]			ATOSEL1 [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw								

ビット31:29 **ATOSEL8[2:0]**：アクティブタンパ共有出力8の選択

000: TAMPOUTSEL8 = TAMP_OUT1
001: TAMPOUTSEL8 = TAMP_OUT2
010: TAMPOUTSEL8 = TAMP_OUT3
011: TAMPOUTSEL8 = TAMP_OUT4
100: TAMPOUTSEL8 = TAMP_OUT5
101: TAMPOUTSEL8 = TAMP_OUT6
110: TAMPOUTSEL8 = TAMP_OUT7
111: TAMPOUTSEL8 = TAMP_OUT8

パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット28:26 **ATOSEL7[2:0]**：アクティブタンパ共有出力7の選択

000: TAMPOUTSEL7 = TAMP_OUT1
001: TAMPOUTSEL7 = TAMP_OUT2
010: TAMPOUTSEL7 = TAMP_OUT3
011: TAMPOUTSEL7 = TAMP_OUT4
100: TAMPOUTSEL7 = TAMP_OUT5
101: TAMPOUTSEL7 = TAMP_OUT6
110: TAMPOUTSEL7 = TAMP_OUT7
111: TAMPOUTSEL7 = TAMP_OUT8

パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット25:23 **ATOSEL6[2:0]**：アクティブタンパ共有出力6の選択
000: TAMPOUTSEL6 = TAMP_OUT1
001: TAMPOUTSEL6 = TAMP_OUT2
010: TAMPOUTSEL6 = TAMP_OUT3
011: TAMPOUTSEL6 = TAMP_OUT4
100: TAMPOUTSEL6 = TAMP_OUT5
101: TAMPOUTSEL6 = TAMP_OUT6
110: TAMPOUTSEL6 = TAMP_OUT7
111: TAMPOUTSEL6 = TAMP_OUT8
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット22:20 **ATOSEL5[2:0]**：アクティブタンパ共有出力5の選択
000: TAMPOUTSEL5 = TAMP_OUT1
001: TAMPOUTSEL5 = TAMP_OUT2
010: TAMPOUTSEL5 = TAMP_OUT3
011: TAMPOUTSEL5 = TAMP_OUT4
100: TAMPOUTSEL5 = TAMP_OUT5
101: TAMPOUTSEL5 = TAMP_OUT6
110: TAMPOUTSEL5 = TAMP_OUT7
111: TAMPOUTSEL5 = TAMP_OUT8
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット19:17 **ATOSEL4[2:0]**：アクティブタンパ共有出力4の選択
000: TAMPOUTSEL4 = TAMP_OUT1
001: TAMPOUTSEL4 = TAMP_OUT2
010: TAMPOUTSEL4 = TAMP_OUT3
011: TAMPOUTSEL4 = TAMP_OUT4
100: TAMPOUTSEL4 = TAMP_OUT5
101: TAMPOUTSEL4 = TAMP_OUT6
110: TAMPOUTSEL4 = TAMP_OUT7
111: TAMPOUTSEL4 = TAMP_OUT8
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。
ビット18:17は、TAMP_ATCR1のATOSEL2[1:0] のミラーであり、TAMP_ATCR1でも読出しまたは書込みできます。

ビット16:14 **ATOSEL3[2:0]**：アクティブタンパ共有出力3の選択
000: TAMPOUTSEL3 = TAMP_OUT1
001: TAMPOUTSEL3 = TAMP_OUT2
010: TAMPOUTSEL3 = TAMP_OUT3
011: TAMPOUTSEL3 = TAMP_OUT4
100: TAMPOUTSEL3 = TAMP_OUT5
101: TAMPOUTSEL3 = TAMP_OUT6
110: TAMPOUTSEL3 = TAMP_OUT7
111: TAMPOUTSEL3 = TAMP_OUT8
パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。
ビット15:14は、TAMP_ATCR1のATOSEL3[1:0] のミラーであり、TAMP_ATCR1でも読出しまたは書込みできます。

ビット13:11 **ATOSEL2[2:0]**：アクティブタンパ共有出力2の選択

000: TAMPOUTSEL2 = TAMP_OUT1
001: TAMPOUTSEL2 = TAMP_OUT2
010: TAMPOUTSEL2 = TAMP_OUT3
011: TAMPOUTSEL2 = TAMP_OUT4
100: TAMPOUTSEL2 = TAMP_OUT5
101: TAMPOUTSEL2 = TAMP_OUT6
110: TAMPOUTSEL2 = TAMP_OUT7
111: TAMPOUTSEL2 = TAMP_OUT8

パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット12:11は、TAMP_ATCR1のATOSEL2[1:0] のミラーであり、TAMP_ATCR1でも読出しまたは書込みできます。

ビット10:8 **ATOSEL1[2:0]**：アクティブタンパ共有出力1の選択

000: TAMPOUTSEL1 = TAMP_OUT1
001: TAMPOUTSEL1 = TAMP_OUT2
010: TAMPOUTSEL1 = TAMP_OUT3
011: TAMPOUTSEL1 = TAMP_OUT4
100: TAMPOUTSEL1 = TAMP_OUT5
101: TAMPOUTSEL1 = TAMP_OUT6
110: TAMPOUTSEL1 = TAMP_OUT7
111: TAMPOUTSEL1 = TAMP_OUT8

パッケージのピン配置でTAMP_OUTx出力が利用できない場合、出力選択値は予約済みとなり、使用できません。

ビット9:8は、TAMP_ATCR1のATOSEL1[1:0] のミラーであり、TAMP_ATCR1でも読出しまたは書込みできます。

ビット7:0 予約済みであり、リセット値に保持する必要があります。

注： **TAMPxAMビットがセットされている場合、対応するTAMPxEビットがTAMP_CR1レジスタですべてクリアされていない限り、このレジスタのアクティブタンパ設定を変更できません。**

アクティブモードに設定されたすべてのタンパは、（同じTAMP_CR1書込みで関連するすべてのTAMPxEをセットして）同時に有効にする必要があります。

アクティブモードに設定されたすべてのタンパは、（同じTAMP_CR1書込みで関連するすべてのTAMPxEをクリアして）同時に無効にする必要があります。

アクティブタンパを無効にしてから再び有効にするまで、最低CK_ATPRE 1周期を待つ必要があります。

64.7.9 TAMP セキュア設定レジスタ（TAMP_SECCFGR）

TZEN = 1の場合、このレジスタに書き込めるのは、APBアクセスがセキュアなときだけです。TZEN = 0の場合、BKPRWSEC[7:0]、BKPWSEC[7:0]、BHKLOCKは非セキュアAPBアクセスにより書込みができ、TAMPSEC、CNT1SEC は書込みができません。

このレジスタは全体的に書込み保護することができ、このレジスタの各ビットは、TAMP_PRIVCFGRの設定に応じて、非特権アクセスから個別に書込み保護することもできます（セクション 64.4.7 : TAMP特権保護モードを参照）。

アドレスオフセット：0x20

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TAMPSEC	BHKLOCK	Res.	Res.	Res.	Res.	Res.	Res.	BKPWSEC[7:0]							
rw	rs							rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT1SEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKPRWSEC[7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **TAMPSEC**：タンパ保護（単調カウンタとバックアップレジスタを除く）

0：APBアクセスがセキュアでも非セキュアでも、タンパの設定と割込みを書き込めます。

1：APBアクセスがセキュアな場合のみ、タンパの設定と割込みを書き込めます。

注： 読出し保護についての詳細は、セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

ビット30 **BHKLOCK**：ブート･ハードウェア･キーのロック

このビットはソフトウェアによって読出しと、1の値の書込みのみが行えます。タンパ検出イベントに続いて、または読出し保護（RDP）が無効な状態で、バックアップレジスタと一緒にハードウェアによってクリアされます。

0：TAMP_BKP0RからTAMP_BKP7Rまでのバックアップレジスタには、所属する保護ゾーンに従ってアクセスできます。

1：TAMP_BKP0RからTAMP_BKP7Rまでのバックアップレジスタには、読出しでも書込みでもアクセスできません（0として読み出され、書込みは無視されます）。

ビット29:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **BKPWSEC[7:0]**：バックアップレジスタ書込み保護オフセット

BKPWSEC値は0 ～ 32 でなければなりません。

保護ゾーン2 はTAMP_BKPyR（y = BKPRWSEC）からTAMP_BKPzR（z = BKPWSEC-1で、BKPWSEC > BKPRWSEC）までのバックアップレジスタ用に定義されています。

– TZEN=1の場合、これらのバックアップレジスタは、セキュアアクセスでのみ書き込めます。

– 読出しは、セキュアアクセスと非セキュアアクセスで可能です。

BKPWSEC = 0またはBKPWSEC ≤ BKPRWSECの場合：保護ゾーン2はありません。

保護ゾーン3 はTAMP_BKPtR（BKPWSEC ≥ BKPRWSECの場合はt = BKPWSEC、それ以外の場合はt = BKPRWSEC）からのバックアップレジスタ用に定義されています。

– 読出しと書込みは、セキュアアクセスと非セキュアアクセスで可能です。

BKPWSEC = 32の場合：保護ゾーン3はありません。

図 768：バックアップレジスタ保護ゾーンを参照してください。

注： TZEN=0の場合：非セキュアアクセスで保護ゾーン2の読み書きが可能です。

注： BKPWPRIVがセットされている場合、BKPRWSEC[7:0] は特権モードでのみ書込みできます。

ビット15 **CNT1SEC**：単調カウンタ1のセキュア保護

0：APBアクセスがセキュアでも非セキュアでも、単調カウンタ1（TAMP_COUNT1R）を読み書きできます。

1：単調カウンタ1（TAMP_COUNT1R）は、APBアクセスがセキュアの場合のみ読み書きできます。

ビット14:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **BKPRWSEC[7:0]**：バックアップレジスタ読み書き保護オフセット

BKPRWSEC値は0 ～ 32 でなければなりません。

保護ゾーン1 はTAMP_BKP0RからTAMP_BKPxR（x = BKPRWSEC-1で、BKPRWSEC ≥ 1）までのバックアップレジスタ用に定義されています。

– TZEN=1の場合、これらのバックアップレジスタは、セキュアアクセスでのみ読み書きできます。

BKPRWSEC = 0の場合：保護ゾーン1はありません。

図 768：バックアップレジスタ保護ゾーンを参照してください。

注： TZEN=0の場合：非セキュアアクセスで保護ゾーン1の読み書きが可能です。

注： BKPRWPRIVがセットされている場合、BKPRWSEC[7:0] は特権モードでのみ書込みできます。

64.7.10 TAMP特権設定レジスタ（TAMP_PRIVCFGR）

このレジスタに書き込めるのは、APBアクセスが特権のときだけです。

TZEN = 1の場合、このレジスタは書込み保護することができ、このレジスタの各ビットは、TAMP_SECCFGR の設定に応じて、非セキュアアクセスから個別に書込み保護することもできます（セクション 64.4.5：TAMPセキュア保護モードを参照）。

アドレスオフセット：0x24

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TAMPPRIV	BKPWPRIV	BKPRWPRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT1PRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															

ビット31 **TAMPPRIV**：タンパ特権保護（バックアップレジスタを除く）

0：タンパ設定および割込みは、特権アクセスまたは非特権アクセスで書込みできます。

1：タンパ設定および割込みは、特権アクセスでのみ書込みできます。

注： 読出し保護についての詳細は、セクション 64.4.7：TAMP特権保護モードを参照してください。

ビット30 **BKPWPRIV**：バックアップレジスタゾーン2の特権保護

0：バックアップレジスタゾーン2は、特権アクセスまたは非特権アクセスで書込みできます。

1：バックアップレジスタゾーン2は、特権アクセスでのみ書込みできます。

ビット29 **BKPRWPRIV**：バックアップレジスタゾーン1の特権保護

0：バックアップレジスタゾーン1は、特権アクセスまたは非特権アクセスで読み書きできます。

1：バックアップレジスタゾーン1は、特権アクセスでのみ読み書きできます。

ビット28:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **CNT1PRIV**：単調カウンタ1の特権保護

0：APBアクセスが特権でも非特権でも、単調カウンタ1（TAMP_COUNT1R）を読み書きできます。

1：APBアクセスが特権の場合のみ、単調カウンタ1（TAMP_COUNT1R）を読み書きできます。

ビット14:0 予約済みであり、リセット値に保持する必要があります。

64.7.11 TAMP割込み有効レジスタ（TAMP_IER）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x2C

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ITAMP13IE	ITAMP12IE	ITAMP11IE	Res.	ITAMP9IE	ITAMP8IE	ITAMP7IE	ITAMP6IE	ITAMP5IE	Res.	ITAMP3IE	ITAMP2IE	ITAMP1IE
			rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8IE	TAMP7IE	TAMP6IE	TAMP5IE	TAMP4IE	TAMP3IE	TAMP2IE	TAMP1IE
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ITAMP13IE**：内部タンパ13割込みイネーブル
0：内部タンパ13割込みは無効です。
1：内部タンパ13割込みは有効です。

ビット27 **ITAMP12IE**：内部タンパ12割込みイネーブル
0：内部タンパ12割込みは無効です。
1：内部タンパ12割込みは有効です。

ビット26 **ITAMP11IE**：内部タンパ11割込みイネーブル
0：内部タンパ11割込みは無効です。
1：内部タンパ11割込みは有効です。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ITAMP9IE**：内部タンパ9割込みイネーブル
0：内部タンパ9割込みは無効です。
1：内部タンパ9割込みは有効です。

ビット23 **ITAMP8IE**：内部タンパ8割込みイネーブル
0：内部タンパ8割込みは無効です。
1：内部タンパ8割込みは有効です。

ビット22 **ITAMP7IE**：内部タンパ7割込みイネーブル
0：内部タンパ7割込みは無効です。
1：内部タンパ7割込みは有効です。

ビット21 **ITAMP6IE**：内部タンパ6割込みイネーブル
0：内部タンパ6割込みは無効です。
1：内部タンパ6割込みは有効です。

ビット20 **ITAMP5IE**：内部タンパ5割込みイネーブル
0：内部タンパ5割込みは無効です。
1：内部タンパ5割込みは有効です。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ITAMP3IE**：内部タンパ3割込みイネーブル
0：内部タンパ3割込みは無効です。
1：内部タンパ3割込みは有効です。

ビット17 **ITAMP2IE**：内部タンパ2割込みイネーブル
0：内部タンパ2割込みは無効です。
1：内部タンパ2割込みは有効です。

ビット16 **ITAMP1IE**：内部タンパ1割込みイネーブル
0：内部タンパ1割込みは無効です。
1：内部タンパ1割込みは有効です。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8IE**：タンパ8割込みイネーブル
0：タンパ8割込みは無効です。
1：タンパ8割込みは有効です。

ビット6 **TAMP7IE**：タンパ7割込みイネーブル
0：タンパ7割込みは無効です。
1：タンパ7割込みは有効です。

ビット5 **TAMP6IE**：タンパ6割込みイネーブル
0：タンパ6割込みは無効です。
1：タンパ6割込みは有効です。

ビット4 **TAMP5IE**：タンパ5割込みイネーブル
0：タンパ5割込みは無効です。
1：タンパ5割込みは有効です。

ビット3 **TAMP4IE**：タンパ4割込みイネーブル
0：タンパ4割込みは無効です。
1：タンパ4割込みは有効です。

ビット2 **TAMP3IE**：タンパ3割込みイネーブル
0：タンパ3割込みは無効です。
1：タンパ3割込みは有効です。

ビット1 **TAMP2IE**：タンパ2割込みイネーブル
0：タンパ2割込みは無効です。
1：タンパ2割込みは有効です。

ビット0 **TAMP1IE**：タンパ1割込みイネーブル
0：タンパ1割込みは無効です。
1：タンパ1割込みは有効です。

64.7.12 TAMPステータスレジスタ（TAMP_SR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x30

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ITAMP13F	ITAMP12F	ITAMP11F	Res.	ITAMP9F	ITAMP8F	ITAMP7F	ITAMP6F	ITAMP5F	Res.	ITAMP3F	ITAMP2F	ITAMP1F
			r	r	r		r	r	r	r	r		r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8F	TAMP7F	TAMP6F	TAMP5F	TAMP4F	TAMP3F	TAMP2F	TAMP1F
								r	r	r	r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ITAMP13F**：内部タンパ13フラグ

このフラグは、内部タンパ13にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット27 **ITAMP12F**：内部タンパ12フラグ

このフラグは、内部タンパ12にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット26 **ITAMP11F**：内部タンパ11フラグ

このフラグは、内部タンパ11にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ITAMP9F**：内部タンパ9フラグ

このフラグは、内部タンパ9にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット23 **ITAMP8F**：内部タンパ8フラグ

このフラグは、内部タンパ8にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット22 **ITAMP7F**：内部タンパ7フラグ

このフラグは、内部タンパ7にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット21 **ITAMP6F**：内部タンパ6フラグ

このフラグは、内部タンパ6にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット20 **ITAMP5F**：内部タンパ5フラグ

このフラグは、内部タンパ5にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ITAMP3F**：内部タンパ3フラグ

このフラグは、内部タンパ3にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット17 **ITAMP2F**：内部タンパ2フラグ

このフラグは、内部タンパ2にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット16 **ITAMP1F**：内部タンパ1フラグ

このフラグは、内部タンパ1にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8F**：TAMP8検出フラグのクリア

このフラグは、TAMP8入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット6 **TAMP7F**：TAMP7検出フラグのクリア

このフラグは、TAMP7入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット5 **TAMP6F**：TAMP6検出フラグのクリア

このフラグは、TAMP6入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット4 **TAMP5F**：TAMP5検出フラグのクリア

このフラグは、TAMP5入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット3 **TAMP4F**：TAMP4検出フラグのクリア

このフラグは、TAMP4入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット2 **TAMP3F**：TAMP3検出フラグのクリア

このフラグは、TAMP3入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット1 **TAMP2F**：TAMP2検出フラグのクリア

このフラグは、TAMP2入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット0 **TAMP1F**：TAMP1検出フラグのクリア

このフラグは、TAMP1入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

64.7.13 TAMP 非セキュアマスク済み割込みステータスレジスタ（TAMP_MISR）

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7：TAMP特権保護モードを参照してください。

アドレスオフセット：0x34

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ITAMP13MF	ITAMP12MF	ITAMP11MF	Res.	ITAMP9MF	ITAMP8MF	ITAMP7MF	ITAMP6MF	ITAMP5MF	Res.	ITAMP3MF	ITAMP2MF	ITAMP1MF
			r	r	r		r	r	r	r	r		r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8MF	TAMP7MF	TAMP6MF	TAMP5MF	TAMP4MF	TAMP3MF	TAMP2MF	TAMP1MF
								r	r	r	r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ITAMP13MF**：内部タンパ13 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ13 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット27 **ITAMP12MF**：内部タンパ12 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ12 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット26 **ITAMP11MF**：内部タンパ11 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ11 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ITAMP9MF**：内部タンパ9 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ9 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット23 **ITAMP8MF**：内部タンパ8 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ8 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット22 **ITAMP7MF**：内部タンパ7タンパ非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ7 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット21 **ITAMP6MF**：内部タンパ6 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ6 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット20 **ITAMP5MF**：内部タンパ5 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ5 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ITAMP3MF**：内部タンパ3 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ3 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット17 **ITAMP2MF**：内部タンパ2 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ2 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット16 **ITAMP1MF**：内部タンパ1 非セキュア割込みマスク済みフラグ
このフラグは、内部タンパ1 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8MF**：TAMP8 非セキュア割込みマスク済みフラグ
このフラグは、タンパ8 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット6 **TAMP7MF**：TAMP7 非セキュア割込みマスク済みフラグ
このフラグは、タンパ7 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット5 **TAMP6MF**：TAMP6 非セキュア割込みマスク済みフラグ
このフラグは、タンパ6 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット4 **TAMP5MF**：TAMP5 非セキュア割込みマスク済みフラグ
このフラグは、タンパ5 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット3 **TAMP4MF**：TAMP4 非セキュア割込みマスク済みフラグ
このフラグは、タンパ4 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット2 **TAMP3MF**：TAMP3 非セキュア割込みマスク済みフラグ
このフラグは、タンパ3 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット1 **TAMP2MF**：TAMP2 非セキュア割込みマスク済みフラグ
このフラグは、タンパ2 非セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット0 **TAMP1MF**：TAMP1 非セキュア割込みマスク済みフラグ
このフラグは、タンパ1 非セキュア割込みが発生した場合にハードウェアによってセットされます。

64.7.14 TAMPセキュアマスク済み割込みステータスレジスタ（TAMP_SMISR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x38

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ITAMP13MF	ITAMP12MF	ITAMP11MF	Res.	ITAMP9MF	ITAMP8MF	ITAMP7MF	ITAMP6MF	ITAMP5MF	Res.	ITAMP3MF	ITAMP2MF	ITAMP1MF
			r	r	r		r	r	r	r	r		r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8MF	TAMP7MF	TAMP6MF	TAMP5MF	TAMP4MF	TAMP3MF	TAMP2MF	TAMP1MF
								r	r	r	r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ITAMP13MF**：内部タンパ13 セキュア割込みマスク済みフラグ
このフラグは、内部タンパ13 セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット27 **ITAMP12MF**：内部タンパ12 セキュア割込みマスク済みフラグ
このフラグは、内部タンパ12 セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット26 **ITAMP11MF**：内部タンパ11 セキュア割込みマスク済みフラグ
このフラグは、内部タンパ11 セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **ITAMP9MF**：内部タンパ9 セキュア割込みマスク済みフラグ
このフラグは、内部タンパ9 セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット23 **ITAMP8MF**：内部タンパ8セキュア割込みマスク済みフラグ
このフラグは、内部タンパ8セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット22 **ITAMP7MF**：内部タンパ7セキュア割込みマスク済みフラグ
このフラグは、内部タンパ7セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット21 **ITAMP6MF**：内部タンパ6セキュア割込みマスク済みフラグ
このフラグは、内部タンパ6セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット20 **ITAMP5MF**：内部タンパ5セキュア割込みマスク済みフラグ
このフラグは、内部タンパ5セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ITAMP3MF**：内部タンパ3セキュア割込みマスク済みフラグ
このフラグは、内部タンパ3セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット17 **ITAMP2MF**：内部タンパ2セキュア割込みマスク済みフラグ
このフラグは、内部タンパ2セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット16 **ITAMP1MF**：内部タンパ1セキュア割込みマスク済みフラグ
このフラグは、内部タンパ1セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TAMP8MF**：TAMP8セキュア割込みマスク済みフラグ
このフラグは、タンパ8セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット6 **TAMP7MF**：TAMP7セキュア割込みマスク済みフラグ
このフラグは、タンパ7セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット5 **TAMP6MF**：TAMP6セキュア割込みマスク済みフラグ
このフラグは、タンパ6セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット4 **TAMP5MF**：TAMP5セキュア割込みマスク済みフラグ
このフラグは、タンパ5セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット3 **TAMP4MF**：TAMP4セキュア割込みマスク済みフラグ
このフラグは、タンパ4セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット2 **TAMP3MF**：TAMP3セキュア割込みマスク済みフラグ
このフラグは、タンパ3セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット1 **TAMP2MF**：TAMP2セキュア割込みマスク済みフラグ
このフラグは、タンパ2セキュア割込みが発生した場合にハードウェアによってセットされます。

ビット0 **TAMP1MF**：TAMP1セキュア割込みマスク済みフラグ
このフラグは、タンパ1セキュア割込みが発生した場合にハードウェアによってセットされます。

64.7.15 TAMPステータスクリアレジスタ（TAMP_SCR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x3C

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CITAMP13F	CITAMP12F	CITAMP11F	Res.	CITAMP9F	CITAMP8F	CITAMP7F	CITAMP6F	CITAMP5F	Res.	CITAMP3F	CITAMP2F	CITAMP1F
			w	w	w		w	w	w	w	w		w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP8F	CTAMP7F	CTAMP6F	CTAMP5F	CTAMP4F	CTAMP3F	CTAMP2F	CTAMP1F
								w	w	w	w	w	w	w	w

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 予約済みであり、リセット値に保持する必要があります。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **CITAMP13F**：ITAMP13検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP13Fビットがクリアされます。

ビット27 **CITAMP12F**：ITAMP12検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP12Fビットがクリアされます。

ビット26 **CITAMP11F**：ITAMP11検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP11Fビットがクリアされます。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **CITAMP9F**：ITAMP9検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP9Fビットがクリアされます。

ビット23 **CITAMP8F**：ITAMP8検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP8Fビットがクリアされます。

ビット22 **CITAMP7F**：ITAMP7検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP7Fビットがクリアされます。

ビット21 **CITAMP6F**：ITAMP6検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP6Fビットがクリアされます。

ビット20 **CITAMP5F**：ITAMP5検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP5Fビットがクリアされます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **CITAMP3F**：ITAMP3検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP3Fビットがクリアされます。

ビット17 **CITAMP2F**：ITAMP2検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP2Fビットがクリアされます。

ビット16 **CITAMP1F**：ITAMP1検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのITAMP1Fビットがクリアされます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **CTAMP8F**：TAMP8検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP8Fビットがクリアされます。

ビット6 **CTAMP7F**：TAMP7検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP7Fビットがクリアされます。

ビット5 **CTAMP6F**：TAMP6検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP6Fビットがクリアされます。

ビット4 **CTAMP5F**：TAMP5検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP5Fビットがクリアされます。

ビット3 **CTAMP4F**：TAMP4検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP4Fビットがクリアされます。

ビット2 **CTAMP3F**：TAMP3検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP3Fビットがクリアされます。

ビット1 **CTAMP2F**：TAMP2検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP2Fビットがクリアされます。

ビット0 **CTAMP1F**：TAMP1検出フラグのクリア
このビットに1を書き込むと、TAMP_SRレジスタのTAMP1Fビットがクリアされます。

64.7.16 TAMP単調カウンタ1レジスタ（TAMP_COUNT1R）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x040

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
COUNT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COUNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **COUNT[31:0]：**

このレジスタは読出し専用で、このレジスタに対して書込みアクセスがあると、1ずつインクリメントされます。このレジスタは、最大値に達するとロールオーバーせずに停止します。

64.7.17 TAMP 消去設定レジスタ（TAMP_ERCFGR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5 : TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7 : TAMP特権保護モードを参照してください。

アドレスオフセット：0x54

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERCFG0
															rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:8 予約済みであり、リセット値に保持する必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **ERCFG0**：リソース0保護を設定可能(1)

0：リソース0は、TAMPペリフェラルによって保護されるデバイス機密情報には含まれません。

1：リソース0は、TAMPペリフェラルによって保護されるデバイス機密情報に含まれます。

1. 表 635：TAMPの入出力ピンと表 637：TAMP相互接続のtamp_confirmed_ercfg0およびtamp_potential_ercfg0信号を参照してください。

64.7.18 TAMPバックアップxレジスタ（TAMP_BKPxR）

このレジスタは、非セキュアアクセスから保護できます。セクション 64.4.5：TAMPセキュア保護モードを参照してください。

このレジスタは、非特権アクセスから保護できます。セクション 64.4.7：TAMP特権保護モードを参照してください。

アドレスオフセット：0x100 + 0x04 * x, (x = 0から31)

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKP[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BKP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット31:0 **BKP[31:0]：**

アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。

デフォルト（ERASE）設定で、このレジスタはタンパ検出イベントでリセットされます。最低1つの内部または外部タンパフラグがセットされている限り、強制的にリセット値となります。このレジスタは、読出し保護（RDP）が無効である場合もリセットされます。

64.7.19 TAMPレジスタマップ

表645.TAMPレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TAMP_CR1	Res.	Res.	Res.	ITAMP13E	ITAMP12E	ITAMP11E	Res.	ITAMP9E	ITAMP8E	ITAMP7E	ITAMP6E	ITAMP5E	Res.	ITAMP3E	ITAMP2E	ITAMP1E	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8E	TAMP7E	TAMP6E	TAMP5E	TAMP4E	TAMP3E	TAMP2E	TAMP1E
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x04	TAMP_CR2	TAMP8TRG	TAMP7TRG	TAMP6TRG	TAMP5TRG	TAMP4TRG	TAMP3TRG	TAMP2TRG	TAMP1TRG	BKERASE	BKBLOCK	Res.	Res.	Res.	TAMP3MSK	TAMP2MSK	TAMP1MSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8NOER	TAMP7NOER	TAMP6NOER	TAMP5NOER	TAMP4NOER	TAMP3NOER	TAMP2NOER	TAMP1NOER
	リセット値	0	0	0	0	0	0	0	0	0	0				0	0	0									0	0	0	0	0	0	0	0
0x08	TAMP_CR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP13NOER	ITAMP12NOER	ITAMP11NOER	Res.	ITAMP9NOER	ITAMP8NOER	ITAMP7NOER	ITAMP6NOER	ITAMP5NOER	Res.	ITAMP3NOER	ITAMP2NOER	ITAMP1NOER
	リセット値																				0	0	0		0	0	0	0	0		0	0	0
0x0C	TAMP_FLTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]		
	リセット値																									0	0	0	0	0	0	0	0
0x10	TAMP_ATCR1	FLTEN	ATOSHARE	Res.	Res.	Res.	AT PER[2:0]			Res.	Res.	Res.	Res.	ATCK SEL[3:0]				ATO SEL4 [1:0]		ATO SEL3 [1:0]		ATO SEL2 [1:0]		ATO SEL1 [1:0]		TAMP8AM	TAMP7AM	TAMP6AM	TAMP5AM	TAMP4AM	TAMP3AM	TAMP2AM	TAMP1AM
	リセット値	0	0				0	0	0					0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TAMP_ATSEEDR	SEED[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	TAMP_ATOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INITS	SEEDF	Res.	Res.	Res.	Res.	Res.	Res.	PRNG[7:0]							
	リセット値																	0	0							0	0	0	0	0	0	0	0
0x1C	TAMP_ATCR2	ATO SEL8 [2:0]			ATO SEL7 [2:0]			ATO SEL6 [2:0]			ATO SEL5 [2:0]			ATO SEL4 [2:0]				ATO SEL3 [2:0]		ATO SEL2 [2:0]			ATO SEL1 [2:0]										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x20	TAMP_SECCFGR	TAMPSEC	BHKLOCK	Res.	Res.	Res.	Res.	Res.	Res.	BKPWSEC[7:0]								CNT1SEC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKPRWSEC[7:0]							
	リセット値	0	0							0	0	0	0	0	0	0	0	0								0	0	0	0	0	0	0	0
0x24	TAMP_PRIVCFGR	TAMPPRIV	BKPWPRIV	BKPRWPRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT1PRIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0														0															

表645.TAMPレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	TAMP_IER	Res.	Res.	Res.	ITAMP13IE	ITAMP12IE	ITAMP11IE	Res.	ITAMP9IE	ITAMP8IE	ITAMP7IF	ITAMP6IE	ITAMP5IE	Res.	ITAMP3IE	ITAMP2IE	ITAMP1IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8IE	TAMP7IE	TAMP6IE	TAMP5IE	TAMP4IE	TAMP3IE	TAMP2IE	TAMP1IE
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x30	TAMP_SR	Res.	Res.	Res.	ITAMP13F	ITAMP12F	ITAMP11F	Res.	ITAMP9F	ITAMP8F	ITAMP7F	ITAMP6F	ITAMP5F	Res.	ITAMP3F	ITAMP2F	ITAMP1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8F	TAMP7F	TAMP6F	TAMP5F	TAMP4F	TAMP3F	TAMP2F	TAMP1F
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x34	TAMP_MISR	Res.	Res.	Res.	ITAMP13MF	ITAMP12MF	ITAMP11MF	Res.	ITAMP9MF	ITAMP8MF	ITAMP7MF	ITAMP6MF	ITAMP5MF	Res.	ITAMP3MF	ITAMP2MF	ITAMP1MF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8MF	TAMP7MF	TAMP6MF	TAMP5MF	TAMP4MF	TAMP3MF	TAMP2MF	TAMP1MF
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x38	TAMP_SMISR	Res.	Res.	Res.	ITAMP13F	ITAMP12F	ITAMP11F	Res.	ITAMP9F	ITAMP8F	ITAMP7F	ITAMP6MF	ITAMP5MF	Res.	ITAMP3MF	ITAMP2MF	ITAMP1MF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP8MF	TAMP7MF	TAMP6MF	TAMP5MF	TAMP4MF	TAMP3MF	TAMP2MF	TAMP1MF
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x3C	TAMP_SCR	Res.	Res.	Res.	CITAMP13F	CITAMP12F	CITAMP11F	Res.	CITAMP9F	CITAMP8F	CITAMP7F	CITAMP6F	CITAMP5F	Res.	CITAMP3F	CITAMP2F	CITAMP1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP8F	CTAMP7F	CTAMP6F	CTAMP5F	CTAMP4F	CTAMP3F	CTAMP2F	CTAMP1F
	リセット値				0	0	0		0	0	0	0	0		0	0	0									0	0	0	0	0	0	0	0
0x40	TAMP_ COUNTR	COUNT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	TAMP_ ERCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERCFG0
	リセット値																																0
0x100 + 0x04*x、（x= 0～31）	TAMP_BKPxR	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

65 I²C（Inter-integrated circuit）インタフェース

65.1 はじめに

I²C（inter-integrated circuit）バスインタフェースは、マイクロコントローラとシリアルI²Cバスの間の通信を処理します。このインタフェースはマルチマスタ機能を有し、すべてのI²Cバス固有のシーケンス、プロトコル、アービトレーション、タイミングを制御することができます。標準モード（Sm）、高速モード（Fm）、および高速モードプラス（Fm+）をサポートします。

また、SMBus（System Management Bus）およびPMBus®（Power Management Bus）と互換性があります。

DMAを使用して、CPUの負荷を軽減できます。

65.2 I²Cの主な機能

- I²Cバス仕様リビジョン03との互換性：
  - スレーブおよびマスタモード
  - マルチマスタ機能
  - 標準モード（最大100kHz）
  - 高速モード（最大400kHz）
  - 高速モードプラス（最大1MHz）
  - 7bitおよび10bitアドレッシングモード
  - 複数の7bitスレーブアドレス（2つのアドレス、1つは設定可能なマスク付き）
  - すべての7bitアドレス確認応答モード
  - 同報（General call）コール
  - プログラム可能なセットアップおよびホールド時間
  - 使いやすいイベント管理
  - クロックストレッチオプション
  - ソフトウェアリセット
- DMA機能付きの1バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます（セクション65.3を参照）。

- SMBus仕様rev 3.0との互換性：
  - ハードウェアPEC（Packet Error Checking）の生成とACK制御による確認
  - コマンドおよびデータ確認応答制御
  - アドレス解決プロトコル（ARP）サポート
  - ホストおよびデバイスのサポート
  - SMBusアラート
  - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.3規格との互換性
- 独立したクロック：独立したクロックソースの選択により、I2Cの通信速度は i2c_pclk の再プログラミングから独立
- STOPモードからのウェイクアップ
- STOPモードでの自律機能

65.3 I²Cの実装

表646.STM32U535/545/575/585のI²Cの実装

I2Cの機能(1)	I2C1	I2C2	I2C3	I2C4
7bitアドレスモード	X	X	X	X
10bitアドレスモード	X	X	X	X
標準モード（最大100 kbit/s）	X	X	X	X
高速モード（最大400 kbit/s）	X	X	X	X
20mA出力駆動I/O搭載高速モードプラス（最大1 Mbit/s）	X	X	X	X
独立クロック	X	X	X	X
自律モード	X	X	X	X
STOPモードからのウェイクアップ	X(2)	X (2)	X(3)	X (2)
SMBus/PMBus	X	X	X	X

1. X : サポートされています。
2. STOP 0およびSTOP 1モードからサポートされるウェイクアップ。
3. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

表647.STM32U59x/5Ax/5Fx/5GxのI2Cの実装

I2Cの機能(1)	I2C1	I2C2	I2C3	I2C4	I2C5	I2C6
7bitアドレスモード	X	X	X	X	X	X
10bitアドレスモード	X	X	X	X	X	X
標準モード（最大100 kbit/s）	X	X	X	X	X	X
高速モード（最大400 kbit/s）	X	X	X	X	X	X
20mA出力駆動I/O搭載高速モードプラス（最大1 Mbit/s）	X	X	X	X	X	X
独立クロック	X	X	X	X	X	X
自律モード	X	X	X	X	X	X
STOPモードからのウェイクアップ	X(2)	X (2)	X(3)	X (2)	X (2)	X (2)
SMBus/PMBus	X	X	X	X	X	X

1. X : サポートされています。
2. STOP 0およびSTOP 1モードからサポートされるウェイクアップ。
3. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

65.4 I²Cの機能詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によってI²Cバスに接続されます。標準（最大100kHz）、高速モード（最大400kHz）、または高速モードプラス（最大1MHz）のI²Cバスで接続できます。

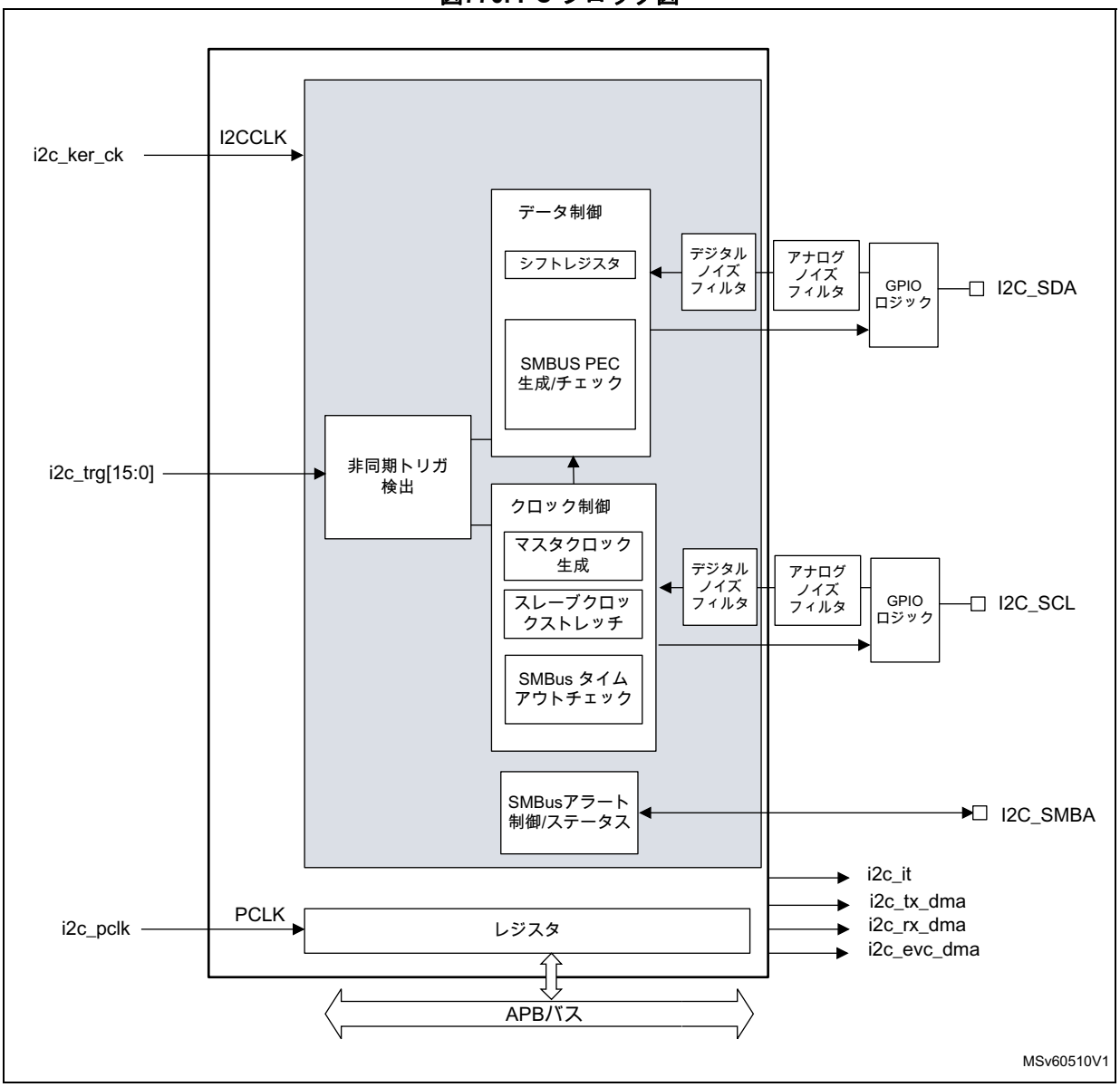
このインタフェースは、データ（SDA）ピンとクロック（SCL）ピンによってSMBusに接続することもできます。

SMBus機能がサポートされる場合、オプションのSMBusアラートピン（SMBA）も使用できます。

65.4.1 I2C ブロック図

I2C インタフェースのブロック図を図770に示します。

図770. I2C ブロック図



I2C は、独立したクロックソースによってクロック供給されるため、I2Cは i2c_pclk 周波数から独立して動作できます。

65.4.2 I²Cピンおよび内部信号

表648.I2Cの入出力ピン

ピン名	信号タイプ	説明
I2C_SDA	双方向	I²Cデータ
I2C_SCL	双方向	I²Cクロック
I2C_SMBA	双方向	SMBusアラート

表649.I2C内部入力／出力信号

内部信号名	信号タイプ	説明
i2c_ker_ck	入力	I²Cカーネルクロック（本書ではI2CCLKとも言います）
i2c_pclk	入力	I²C APBクロック
i2c_trg [15:0]	入力	I²Cトリガ
i2c_it	出力	I²C割込み。割込みソースの全リストは、表664を参照してください。
i2c_rx_dma	出力	I²C受信データDMAリクエスト（I2C_RX）
i2c_tx_dma	出力	I²C送信データDMAリクエスト（I2C_TX）
i2c_evc_dma	出力	I²Cイベント制御DMAリクエスト（I2C_EVC）

表650.I2C1、I2C2、I2C4、I2C5、I2C6相互接続

信号名	転送元／転送先
i2c_trg0	gpdma1_ch0_tc
i2c_trg1	gpdma1_ch1_tc
i2c_trg2	gpdma1_ch2_tc
i2c_trg3	gpdma1_ch3_tc
i2c_trg4	exti5
i2c_trg5	exti9
i2c_trg6	lptim1_ch1
i2c_trg7	lptim2_ch1
i2c_trg8	comp1_out
i2c_trg9	comp2_out
i2c_trg10	rtc_alra_trg
i2c_trg11	rtc_wut_trg
i2c_trg12	-
i2c_trg13	-
i2c_trg14	-
i2c_trg15	-

表651.I2C3相互接続

信号名	転送元／転送先
i2c_trg0	lpdma1_ch0_tc
i2c_trg1	lpdma1_ch1_tc
i2c_trg2	lpdma1_ch2_tc
i2c_trg3	lpdma1_ch3_tc
i2c_trg4	exti5
i2c_trg5	exti8
i2c_trg6	lptim1_ch1
i2c_trg7	lptim3_ch1
i2c_trg8	comp1_out
i2c_trg9	comp2_out
i2c_trg10	rtc_alra_trg
i2c_trg11	rtc_wut_trg
i2c_trg12	-
i2c_trg13	-
i2c_trg14	-
i2c_trg15	-

65.4.3 I²Cクロックの要件

I²Cカーネルは i2c_ker_ck によってクロック供給されます。

i2c_ker_ck の周期t_{I2CCLK} は、次の条件を満たす必要があります。

- $t_{I2CCLK} < (t_{LOW} - t_{filters}) / 4$
- $t_{I2CCLK} < t_{HIGH}$

ここで：

t_{LOW}：SCLロー時間およびt_{HIGH}：SCLハイ時間

$t_{filters}$:有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

デジタルフィルタの遅延は、DNF x t_{I2CCLK} です。

i2c_pclk クロックの周期t_{PCLK} は、次の条件を満たす必要があります。

- $t_{PCLK} < 4 / 3\ t_{SCL}$（$t_{SCL}$：SCL周期）

注意： **I²Cカーネルが i2c_pclk によってクロック供給されるとき、このクロックはt_{I2CCLK} の条件を満たす必要があります。**

65.4.4 モード選択

このインタフェースは、次の4つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、STARTコンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失またはSTOP生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

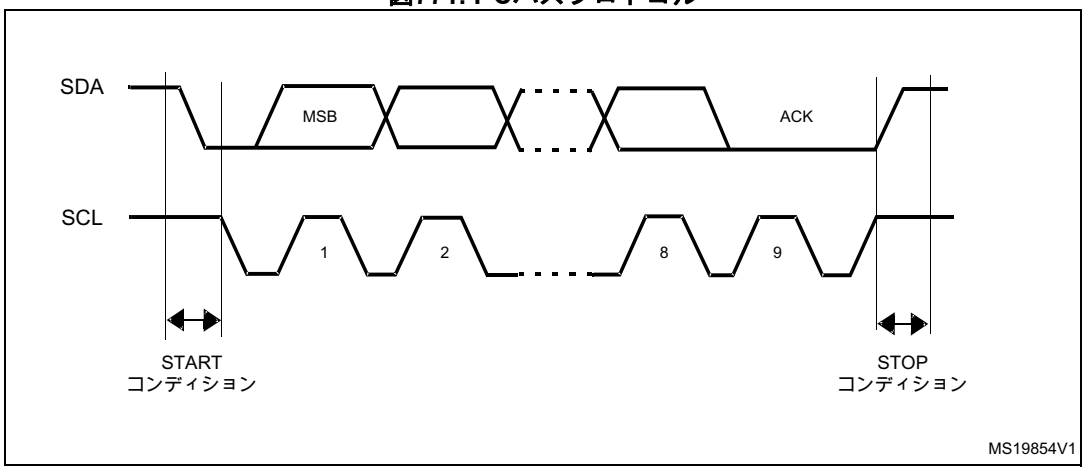
マスタモードでは、I²C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常にSTARTコンディションで開始され、STOPコンディションで終わります。STARTおよびSTOPコンディションは、マスタモードではソフトウェアによって生成されます。

スレーブモードでは、このインタフェースは、自己アドレス（7または10bit）と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みのSMBusアドレスもソフトウェアによって有効にできます。

データとアドレスは、MSBが先の8bitバイトとして転送されます。STARTコンディションの後に続く最初のバイト（7bitモードでは1バイト、10bitモードでは2バイト）にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8つの クロックサイクルのバイト転送の後には9番目のクロックパルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります（図771を参照）。

図771. I²Cバスプロトコル



確認応答（Acknowledge）は、ソフトウェアによって有効または無効にできます。I²Cインタフェースのアドレスは、ソフトウェアによって選択できます。

65.4.5 I²Cの初期化

ペリフェラルの有効化と無効化

クロックコントローラでI²Cペリフェラルクロックを設定して有効にする必要があります。その後 I²Cは、I2C_CR1レジスタのPEビットをセットすることで有効にできます。

I²Cが無効にされると（PE = 0）、I²Cはソフトウェアリセットを実行します。詳細については、セクション65.4.6を参照してください。

ノイズフィルタ

I2C_CR1レジスタのPEビットをセットすることによって I2Cペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDAおよびSCL入力にアナログノイズフィルタがあります。このフィルタはI²C仕様に準拠しており、高速モードおよび高速モードプラスで最大50 nsのパルス幅を持つスパイクを抑制します。ANFOFFビットをセットすることによって、このアナログフィルタを無効にし、I2C_CR1レジスタのDNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCLまたはSDAラインのレベルは、DNF x i2c_ker_ck 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な1～15 i2c_ker_ck 周期の長さを持つスパイクを抑制できます。

表652.アナログフィルタとデジタルフィルタの比較

-	アナログフィルタ	デジタルフィルタ
抑制されるスパイクのパルス幅	≥ 50ns	長さを1 ～ 15 I²Cペリフェラルクロックにプログラム可能
利点	STOPモードで使用可能	– プログラム可能な長さ：標準要件に対して追加のフィルタリング機能 – 安定した長さ
欠点	温度、電圧、プロセスのばらつき	デジタルフィルタが有効なときには、STOPモードでの機能はサポートされません。

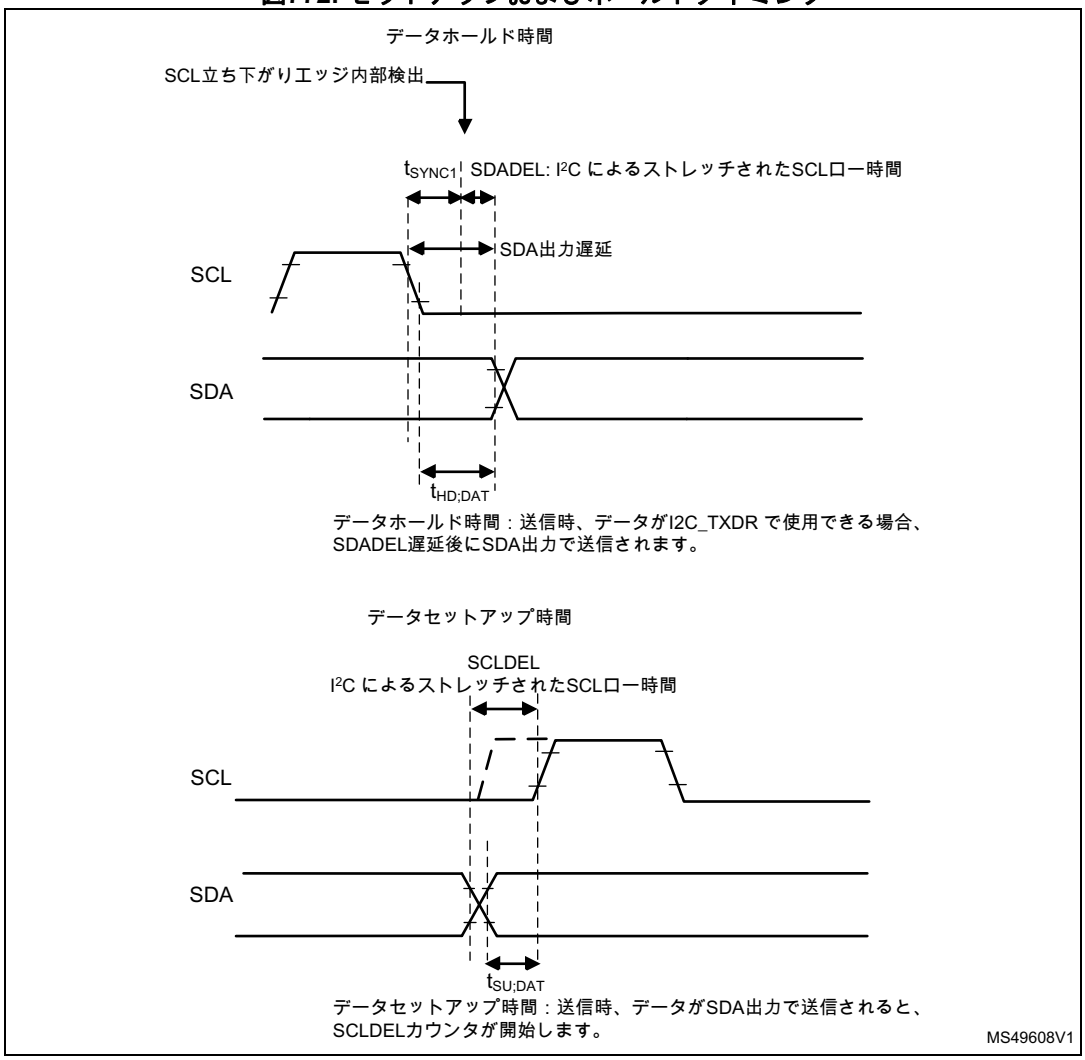
注意： **I²Cが有効なときには、フィルタ設定は変更できません。**

I²Cのタイミング

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間を保証するためには、タイミングを設定する必要があります。これを行うには、I2C_TIMINGR レジスタのPRESC[3:0]、SCLDEL[3:0]、およびSDADEL[3:0] ビットをプログラムします。

STM32CubeMXツールは、I²C設定ウィンドウのI2C_TIMINGRの内容を計算し、提供します。

図772. セットアップおよびホールドタイミング



SCLの立ち下がりエッジが内部で検出されると、遅延（ホールド時間 $t_{HD;DAT}$ に影響を与える t_{SDADEL}）がSDA出力を送信する前に挿入されます。t_{SDADEL} = SDADEL x t_{PRESC} + t_{I2CCLK}、ここで t_{PRESC} = (PRESC + 1) x t_{I2CCLK} です。

SDA出力遅延の合計は、次のとおりです：

t_{SYNC1} + {[SDADEL x (PRESC + 1) + 1] x t_{I2CCLK} }

t_{SYNC1} の長さは、以下に依存します。

- SCL立下がり傾斜
- アナログフィルタが有効なときの入力遅延：$t_{AF(min)} < t_{AF} < t_{AF(max)}$
- デジタルフィルタが有効なときの入力遅延：t_{DNF} = DNF x t_{I2CCLK}
- i2c_ker_ck クロックへのSCLの同期による遅延（2 ～ 3 i2c_ker_ck 周期）

SCL立下がりエッジの未定義の領域をブリッジするためには、SDADELを次のようにプログラムする必要があります。

$\{t_{f\,(max)} + t_{HD;DAT\,(min)} - t_{AF(min)} - [(DNF + 3) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\} \leq SDADEL$

$SDADEL \leq \{t_{HD;DAT\,(max)} - t_{AF(max)} - [(DNF + 4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： **$t_{AF(min)}$ / $t_{AF(max)}$ は、アナログフィルタが有効なときのみ、等式に含まれます。t_{AF} の値については、デバイスのデータシートを参照してください。**

最大$t_{HD;DAT}$ は、標準モード、高速モード、および高速モードプラスでそれぞれ3.45、0.9、および0.45 µsです。これは、遷移時間による$t_{VD;DAT}$ の最大値より短くなければなりません。この最大値を満たす必要があるのは、デバイスがSCL信号のLOW周期（t_{LOW}）をストレッチしない場合だけです。クロックがSCLをストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

通常、SDAの立上がりエッジが最悪ケースです。この場合、前の式は次のようになります。

$SDADEL \leq \{t_{VD;DAT\ (max)} - t_{r\ (max)} - t_{AF\ (max)} - [(DNF + 4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$。

注： **NOSTRETCH = 0のときには、SCLDELの値に従って、デバイスはセットアップ時間を保証するためにSCLローをストレッチするので、この条件に違反することがあります。**

t_f、t_r、$t_{HD;DAT}$、および$t_{VD;DAT}$ の規格値については、表653を参照してください。

- t_{SDADEL} の後や、データがI2C_TXDRレジスタにまだ書き込まれていないためスレーブがクロックをストレッチしなければならなかった場合のSDA出力送信後、SCLラインはセットアップ時間中、ローレベルで保持されます。このセットアップ時間は、$t_{SCLDEL} = (SCLDEL+ 1) \times t_{PRESC}$ であり、ここで、$t_{PRESC} = (PRESC+ 1) \times t_{I2CCLK}$ です。t_{SCLDEL} はセットアップ時間 $t_{SU;DAT}$ に影響を与えます。

SDA遷移（立上がりエッジは通常、最悪のケース）の未定義の領域をブリッジするためには、SCLDELを次のようにプログラムする必要があります。

$\{[t_{r\ (max)} + t_{SU;DAT\ (min)}] / [(PRESC+ 1)] \times t_{I2CCLK}]\} - 1 \leq SCLDEL$

t_r および $t_{SU;DAT}$ の規格値については、表653を参照してください。

使用するSDAおよびSCL遷移時間の値は、アプリケーションでの値です。I_2C規格から最大値を使用すると、SDADELとSCLDELの計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

注： **各クロックパルスで、SCL立下がりエッジの検出後、I²Cマスタまたはスレーブは、最低 $[(SDADEL + SCLDEL + 1) \times (PRESC + 1) + 1] \times t_{I2CCLK}$ の間、送信および受信の両モードでSCLローをストレッチします。送信モードで、SDADELカウンタ終了時にデータがまだI2C_TXDRに書き込まれていない場合、I²Cは次のデータが書き込まれるまでSCLローをストレッチし続けます。その時、新しいデータMSBがSDA出力で送信され、SCLDELカウンタが開始し、SCLローのストレッチを継続して、データセットアップ時間を保証します。**

スレーブモードでNOSTRETCH = 1の場合、SCLはストレッチされません。そのため、十分なセットアップ時間が保証されるように、SDADELをプログラムする必要があります。

表653.I²C-SMBus仕様のデータのセットアップおよびホールド時間

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
$t_{HD;DAT}$	データホールド時間	0	-	0	-	0	-	0.3	-	µs
$t_{VD;DAT}$	データ有効時間	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	データセットアップ時間	250	-	100	-	50	-	250	-	ns
t_r	SDAおよびSCL信号の立上がり時間	-	1000	-	300	-	120	-	1000	
t_f	SDAおよびSCL信号の立下がり時間	-	300	-	300	-	120	-	300	

また、マスタモードでは、I2C_TIMINGRレジスタのPRESC[3:0]、SCLH[7:0]、およびSCLL[7:0] ビットフィールドをプログラムすることによって、SCLクロックのハイおよびローレベルを設定する必要があります。

- SCL立下がりエッジが内部で検出されると、SCL出力をリリースする前に遅延が挿入されます。この遅延は、t_{SCLL} = (SCLL + 1) x t_{PRESC} であり、t_{PRESC} = (PRESC + 1) x t_{I2CCLK} です。
  t_{SCLL} は、SCLロー時間 t_{LOW} に影響を与えます。
- SCL立上がりエッジが内部で検出されると、SCL出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、t_{SCLH} = (SCLH + 1) x t_{PRESC}であり、t_{PRESC} = (PRESC+ 1) x t_{I2CCLK} です。t_{SCLH}はSCLハイ時間 t_{HIGH} に影響を与えます。

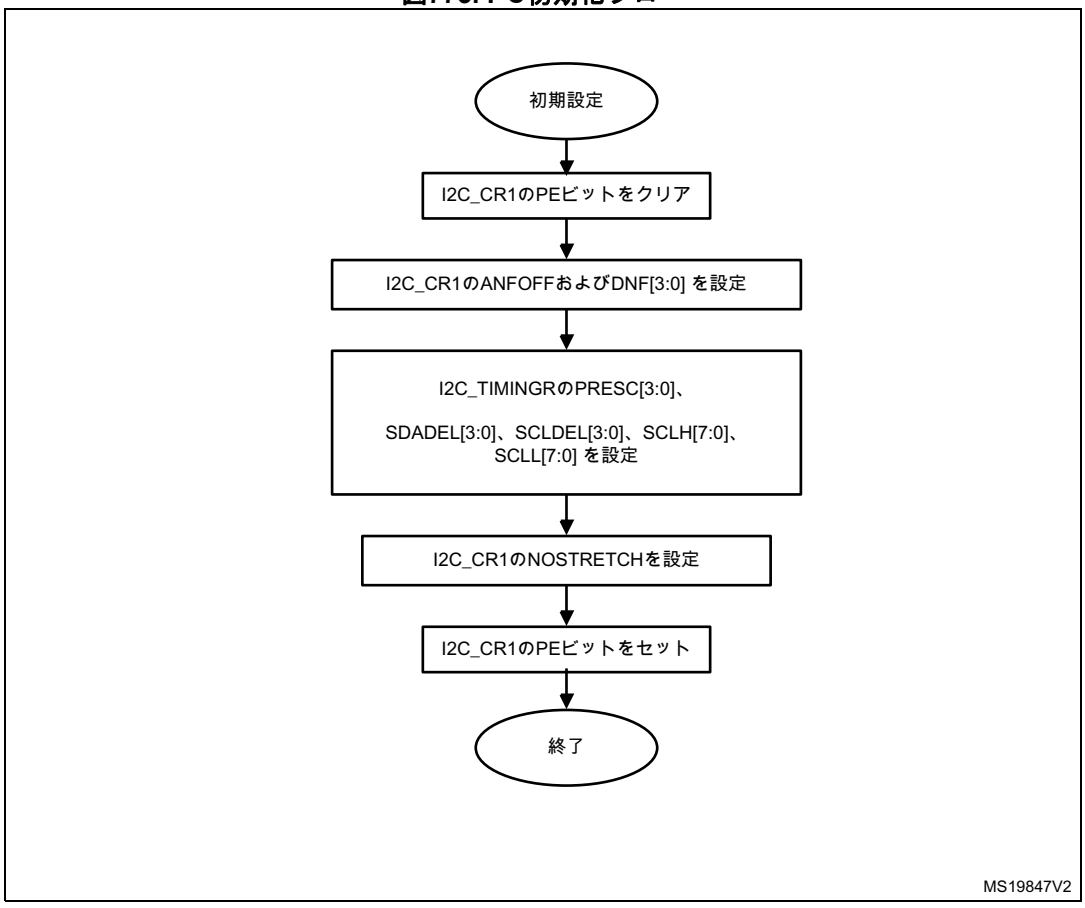
詳細については、I2Cマスタ初期化を参照してください。

注意： **I²Cが有効なときには、タイミング設定の変更はできません。**

ペリフェラルを有効にする前に、I_2CスレーブNOSTRETCHモードも設定する必要があります。詳細については、I2Cスレーブ初期化を参照してください。

注意： **I²Cが有効なときには、NOSTRETCH設定の変更はできません。**

図773. I²C初期化フロー



65.4.6 ソフトウェアリセット

ソフトウェアリセットを行うには、I2C_CR1レジスタのPEビットをクリアします。その場合、I²Cのラインの SCLおよびSDAがリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2C_CR2レジスタ：START、STOP、NACK
2. I2C_ISRレジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus機能がサポートされるときには、以下も影響を受けます：

1. I2C_CR2レジスタ：PECBYTE
2. I2C_ISRレジスタ：PECERR、TIMEOUT、ALERT

ソフトウェア·リセットを実行するには、少なくとも3 APBクロック·サイクルの間、PEをローに保持する必要があります。このためには、次のソフトウェアシーケンスを書き込みます：

1. PE = 0を書き込む。
2. PE = 0であることを確認します。
3. PE = 1を書き込む。

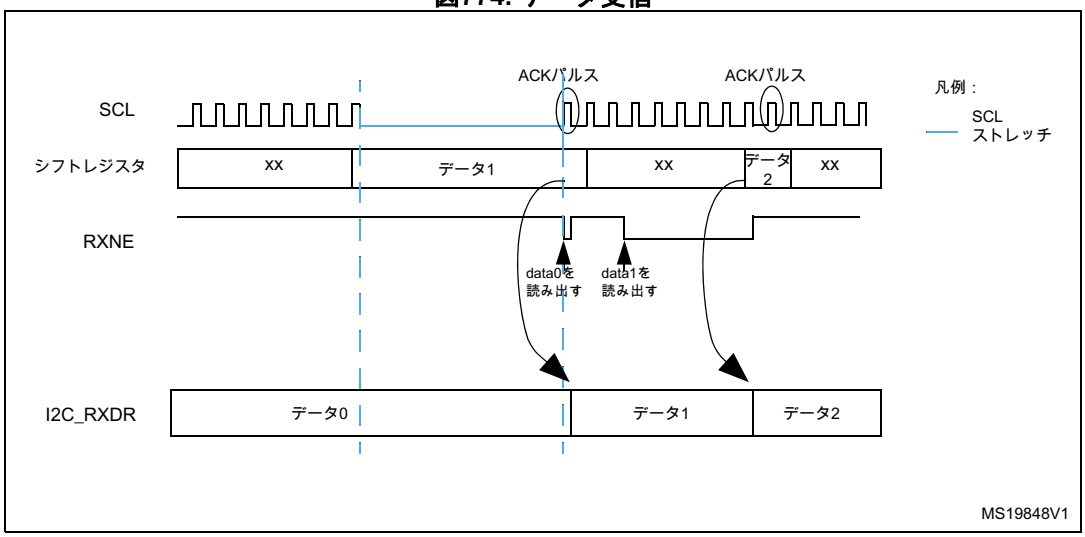
65.4.7 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA入力はシフトレジスタに送られます。8番目のSCLパルスの後（完全なデータバイトの受信後）、シフトレジスタは、I2C_RXDRレジスタが空の場合（RXNE = 0）、このレジスタにコピーされます。RXNE = 1の場合、すなわち、前に受信されたデータバイトがまだ読み出されていなかった場合、SCLラインは I2C_RXDRが読み出されるまでストレッチされます。ストレッチは、8番目と9番目のSCLパルスの間（確認応答パルスの前）に挿入されます。

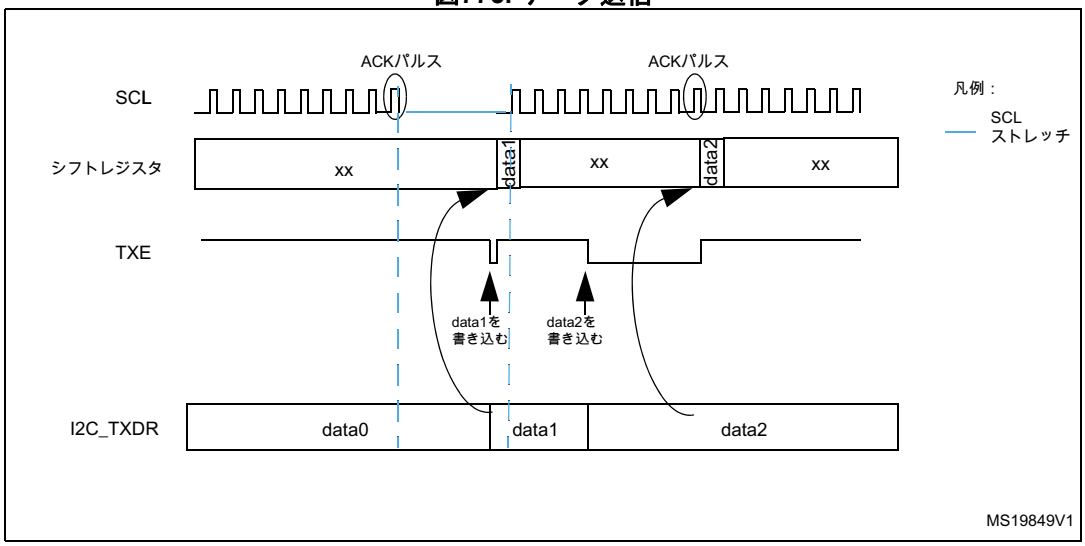
図774. データ受信



送信

I2C_TXDRレジスタが空でない場合（TXE = 0）、その内容が9番目のSCLパルス（確認応答パルス）の後、シフトレジスタにコピーされます。次に、シフトレジスタの内容がSDAラインにシフトアウトされます。TXE = 1の場合、すなわち、I2C_TXDRにデータがまだ書き込まれていない場合、SCLラインは I2C_TXDRに書き込まれるまでストレッチされます。ストレッチは、9番目のSCLパルスの後で行われます。

図775. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2Cにはハードウェアにバイトカウンタが組み込まれています。

- マスタモードでのNACK、STOP、およびReSTART生成
- スレーブレシーバモードでのACK制御
- SMBus機能がサポートされているときのPEC生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2C_CR1レジスタのSBC（スレーブバイト制御）ビットをセットすることにより、ソフトウェアによって有効にできます。

転送されるバイト数は、I2C_CR2レジスタのNBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数（NBYTES）が255より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2C_CR2レジスタのRELOADビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTESでプログラムされたバイト数が転送されると、TCRフラグがセットされ、TCIEがセットされている場合は割込みが生成されます。SCLは、TCRフラグがセットされている間、ストレッチされます。TCRは、NBYTESにゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTESカウンタに最後のバイト数が再ロードされたときには、RELOADビットがクリアされる必要があります。

マスタモードでRELOAD = 0のときには、カウンタは以下の2つのモードで使用できます。

- **自動終了モード**（I2C_CR2レジスタのAUTOEND = “1”）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタはSTOPコンディションを自動的に送信します。
- **ソフトウェア終了モード**（I2C_CR2レジスタのAUTOEND = 0）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェアアクションが求められます。TCフラグがセットされ、TCIEビットがセットされている場合は割込みが生成されます。SCL信号は、TCフラグがセットされている間、ストレッチされます。TCフラグは、I2C_CR2レジスタのSTARTまたはSTOPビットがセットされたときに、ソフトウェアによってクリアされます。マスタがRESTARTコンディションを送信したいときには、このモードを使用する必要があります。

注意： **AUTOENDビットは、RELOADビットがセットされているときには効果がありません。**

表654.I2C設定

機能	SBCビット	RELOADビット	AUTOENDビット
マスタTx/Rx NBYTES + STOP	x	0	1
マスタTx/Rx + NBYTES + RESTART	x	0	0
スレーブTx/Rx、すべての受信バイトにACK	0	x	x
スレーブRxおよびACK制御	1	1	x

65.4.8 I²Cスレーブモード

I²Cスレーブ初期化

スレーブモードで動作するには、少なくとも1つのスレーブアドレスを有効にする必要があります。レジスタ I2C_OAR1と I2C_OAR2を使用して、スレーブ専用アドレスOA1およびOA2をプログラムできます。

- OA1は、I2C_OAR1レジスタのOA1MODEビットをセットすることによって、7bitモード（デフォルト）または10bitアドレッシングモードに設定できます。

  OA1を有効にするには、I2C_OAR1レジスタのOA1ENビットをセットします。
- 追加のスレーブアドレスが必要な場合は、2番目のスレーブアドレスOA2を設定できます。I2C_OAR2レジスタのOA2MSK[2:0] ビットを設定することによって、最大7つのOA2 LSBをマスクできます。したがって、OA2MSKが1から6まで設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、またはOA2[7] のみが受信アドレスと比較されます。OA2MSKが0に等しくなくなるとすぐに、OA2のアドレスコンパレータは、確認応答されない I2C予約済みアドレス（0000 XXXおよび1111 XXX）を除外します。OA2MSK = 7の場合、受信されたすべての7bitのアドレスが確認応答されます（予約済みアドレスを除く）。OA2は常に7bitアドレスです。

  これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2C_OAR1または I2C_OAR2レジスタがOA2MSK = 0でプログラムされた場合、確認応答できます。

  OA2を有効にするには、I2C_OAR2レジスタのOA2ENビットをセットします。
- 同報アドレスは、I2C_CR1レジスタのGCENビットをセットすることで有効になります。

I2Cが有効アドレスの1つによって選択されると、ADDR割込みステータスフラグがセットされ、ADDRIEビットがセットされている場合は割込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用します。これは、必要なときには、ソフトウェアアクションを実行するために、SCL信号をローレベルでストレッチすることを意味します。

マスタがクロックストレッチをサポートしない場合、I2C_CR1レジスタのNOSTRETCH = 1で I2Cを設定する必要があります。

ADDR割込みの受信後、いくつかのアドレスが有効な場合は、I2C_ISRレジスタのADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIRフラグも確認する必要があります。

スレーブクロックストレッチ（NOSTRETCH = 0）

デフォルトモードでは、I2Cスレーブは次の状況でSCLクロックをストレッチします：

- ADDR フラグがセットされると：受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCFビットをセットすることによりソフトウェアによってADDRフラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了し、新しいデータが I2C_TXDRレジスタに書き込まれなかった場合、またはADDRフラグがクリアされたときに（TXE = 1）、最初のデータバイトが書き込まれていなかった場合。このストレッチは、データが I2C_TXDRレジスタに書き込まれたときにリリースされます。
- 受信時、I2C_RXDRレジスタがまだ読み出されておらず、新しいデータ受信が完了したとき。このストレッチは、I2C_RXDRが読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード（SBC = 1およびRELOAD = 1）でTCR = 1のとき、すなわち、最後のデータバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによってTCRがクリアされたときにリリースされます。
- SCL立下がりエッジの検出後、I2Cは、[(SDADEL + SCLDEL + 1) x (PRESC+ 1) + 1] x t_{I2CCLK} の間、SCLローをストレッチします。

クロックストレッチなしのスレーブ（NOSTRETCH = 1）

I2C_CR1レジスタのNOSTRETCH = 1のとき、I2CスレーブはSCL信号をストレッチしません。

- ADDRフラグがセットされている間、SCLクロックはストレッチされません。
- 送信時、転送に対応する最初のSCLパルスが発生する前に、I2C_TXDRレジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2C_ISRレジスタでOVRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。OVRフラグは、最初のデータ送信が開始し、STOPFビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後でのみ、前に転送のSTOPFフラグをクリアすることによって、送信される最初のデータについても、OVRステータスが提供されることを確実にできます。
- 受信時、次のデータバイトの9番目のSCLパルス（ACKパルス）が発生する前に、I2C_RXDRレジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2C_ISRレジスタのOVRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

スレーブバイト制御モード

スレーブ受信モードでバイトACK制御を可能にするには、I2C_CR1レジスタのSBCビットをセットすることによって、スレーブ·バイト制御モードを有効にする必要があります。これは、SMBus規格に準拠する必要があります。

スレーブ受信モードでバイトACK制御を可能にするには、再ロード·モードを選択する必要があります（RELOAD = 1）。各バイトの制御を得るには、ADDR割込みサブルーチンでNBYTESを0x1に初期化し、各受信バイト後に0x1に再ロードする必要があります。バイトが受信されると、TCRビットがセットされ、8番目と9番目のSCLパルスの間で、SCL信号ローをストレッチします。I2C_RXDRレジスタからデータを読み出すことができ、その後、I2C_CR2レジスタのACKビットを設定することによって、確認応答するかどうかを決定できます。SCLストレッチは、NBYTESをゼロ以外の値にプログラムすることによってリリースされ、確認応答または否認応答が送信され、次のバイトを受信できます。

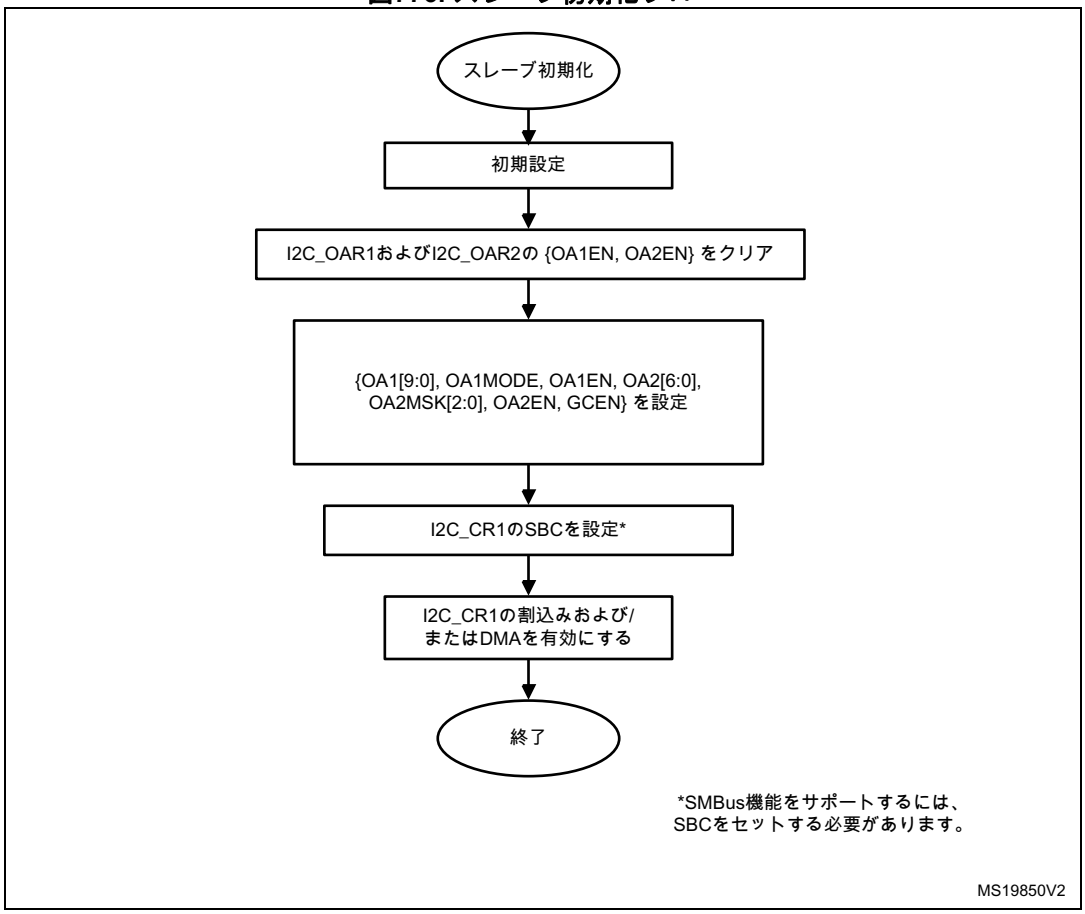
NBYTESに0x1より大きい値をロードでき、この場合、受信フローはNBYTESデータ受信中、継続します。

注： **SBCビットは、I²Cが無効なとき、またはスレーブがアドレス指定されていないとき、またはADDR = 1のときに設定する必要があります。**

RELOADビットの値は、ADDR = 1のとき、またはTCR = 1のときに変更できます。

注意： **スレーブバイト制御モードは、NOSTRETCHモードと互換性がありません。NOSTRETCH = 1のときにSBCをセットすることはできません。**

図776. スレーブ初期化フロー



スレーブトランスミッタ

I2C_TXDRレジスタが空になると、送信割込みステータス（TXIS）が生成されます。I2C_CR1レジスタのTXIEビットがセットされている場合は、割込みが生成されます。

TXISビットは、I2C_TXDRレジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACKが受信されると、I2C_ISRレジスタのNACKFビットがセットされ、I2C_CR1レジスタのNACKIEビットがセットされていた場合は割込みが生成されます。マスタがSTOPまたはRESTARTコンディションを実行できるように、スレーブはSCLおよびSDAラインを自動的にリリースします。TXISビットは、NACK受信時にはセットされません。

STOPが受信され、I2C_CR1レジスタのSTOPIEビットがセットされると、I2C_ISRレジスタのSTOPFフラグがセットされ、割込みが生成されます。ほとんどのアプリケーションでは、SBCは通常、0にプログラムされます。この場合、スレーブアドレスが受信されたときに（ADDR = 1）、TXE = 0であった場合、I2C_TXDRレジスタの内容を最初のデータバイトとして送信するか、新しいデータバイトをプログラムするためにTXEビットをセットすることによって I2C_TXDRレジスタを一掃するかを選択できます。

スレーブバイト制御モード（SBC = 1）では、送信バイト数をアドレス一致割込みサブルーチンのNBYTESでプログラムする必要があります（ADDR = 1）。この場合、転送中のTXISイベントの数は、NBYTESでプログラムされた値に対応します。

注意： **NOSTRETCH = 1のとき、SCLクロックはADDRフラグがセットされている間はストレッチされないので、最初のデータバイトをプログラムするためにADDRサブルーチンで I2C_TXDRレジスタの内容を一掃することはできません。最初に送信されるデータバイトは、I2C_TXDRレジスタで前もってプログラムされている必要があります。**

- このデータは、前の送信メッセージの最後のTXISイベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするためにTXEビットをセットすることによって I2C_TXDRレジスタを一掃できます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPFビットのクリアは、これらのアクションの後でのみ行う必要があります。

  最初のデータ送信が開始したときにSTOPFがまだセットされていた場合、アンダーランエラーが生成されます（OVRフラグがセットされます）。

  TXISイベントが必要な場合（送信割込みまたは送信DMAリクエスト）、TXISイベントを生成するためには、TXEビットに加えてTXISビットもセットする必要があります。

図777. I²Cスレーブトランスミッタの転送シーケンスフロー（NOSTRETCH = 0）

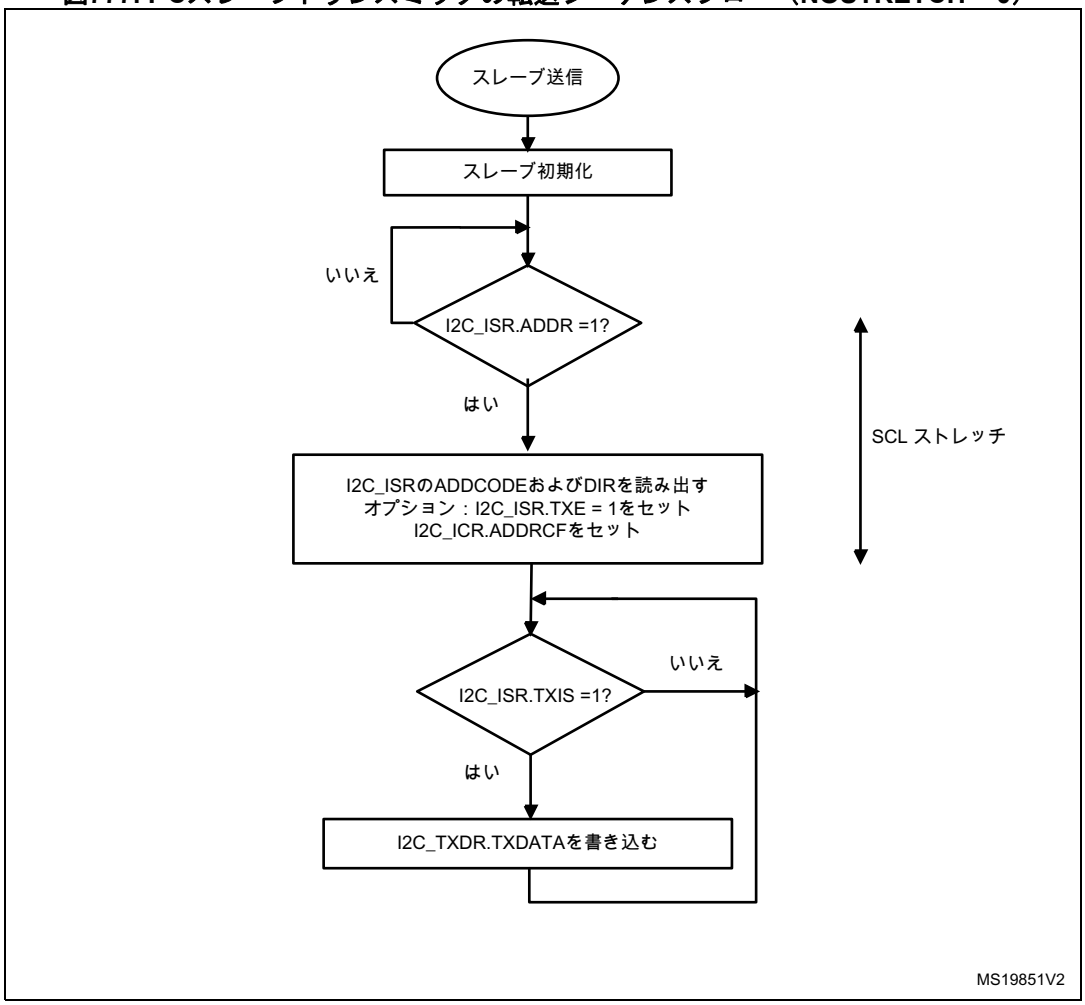


図778. I2Cスレーブトランスミッタの転送シーケンスフロー（NOSTRETCH = 1）

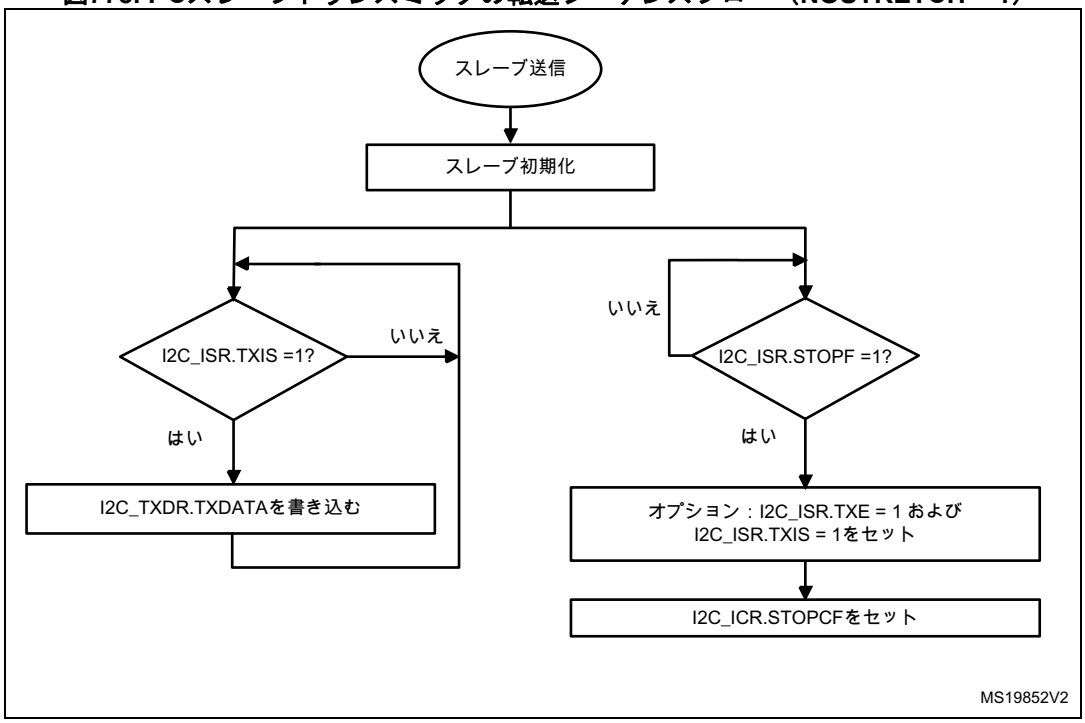
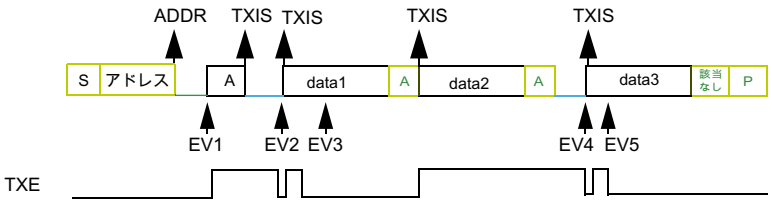


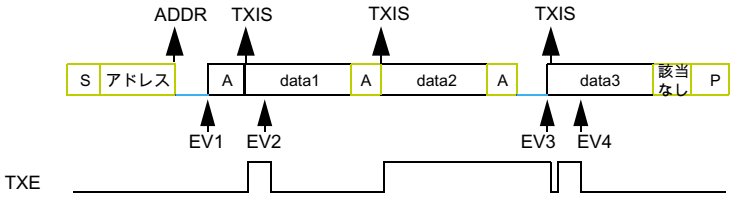
図779. I²Cスレーブトランスミッタの転送バス図（必須イベントのみ）

最初のデータをFlashするI²Cスレーブトランスミッタ3バイト、NOSTRETCH = 0の例：



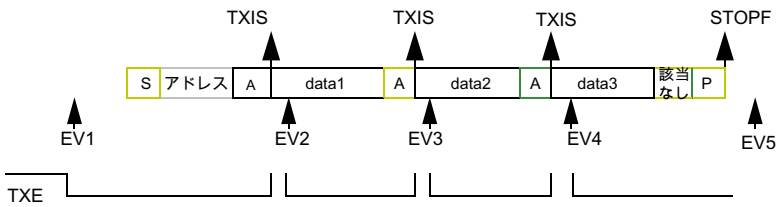
EV1：ADDR ISR：ADDCODEおよびDIRをチェック、TXEをセット、ADDRCFをセット
EV2：TXIS ISR：data1を書き込む
EV3：TXIS ISR：data2を書き込む
EV4：TXIS ISR：data3を書き込む
EV5：TXIS ISR：data4を書き込む（送信されない）

最初のデータをFlashしないI²Cスレーブトランスミッタ3バイト、NOSTRETCH = 0の例：



EV1：ADDR ISR：ADDCODEおよびDIRをチェック、ADDRCFをセット
EV2：TXIS ISR：data2を書き込む
EV3：TXIS ISR：data3を書き込む
EV4：TXIS ISR：data4を書き込む（送信されない）

I²Cスレーブトランスミッタ3バイト、NOSTRETCH=1の例：



EV1：data1を書き込む
EV2：TXIS ISR：data2を書き込む
EV3：TXIS ISR：data3を書き込む
EV4：TXIS ISR：data4を書き込む（送信されない）
EV5： STOPF ISR:（オプション：TXEおよびTXISをセット）、STOPCFをセット

MS19853V2

スレーブレシーバ

I2C_RXDRがフルのときには、I2C_ISRのRXNEがセットされ、I2C_CR1のRXIEがセットされている場合は割込みが生成されます。RXNEは、I2C_RXDRが読み出されたときにクリアされます。

STOPが受信され、I2C_CR1レジスタのSTOPIEビットがセットされると、I2C_ISRのSTOPFがセットされ、割込みが生成されます。

図780. スレーブレシーバの転送シーケンスフロー（NOSTRETCH = 0）

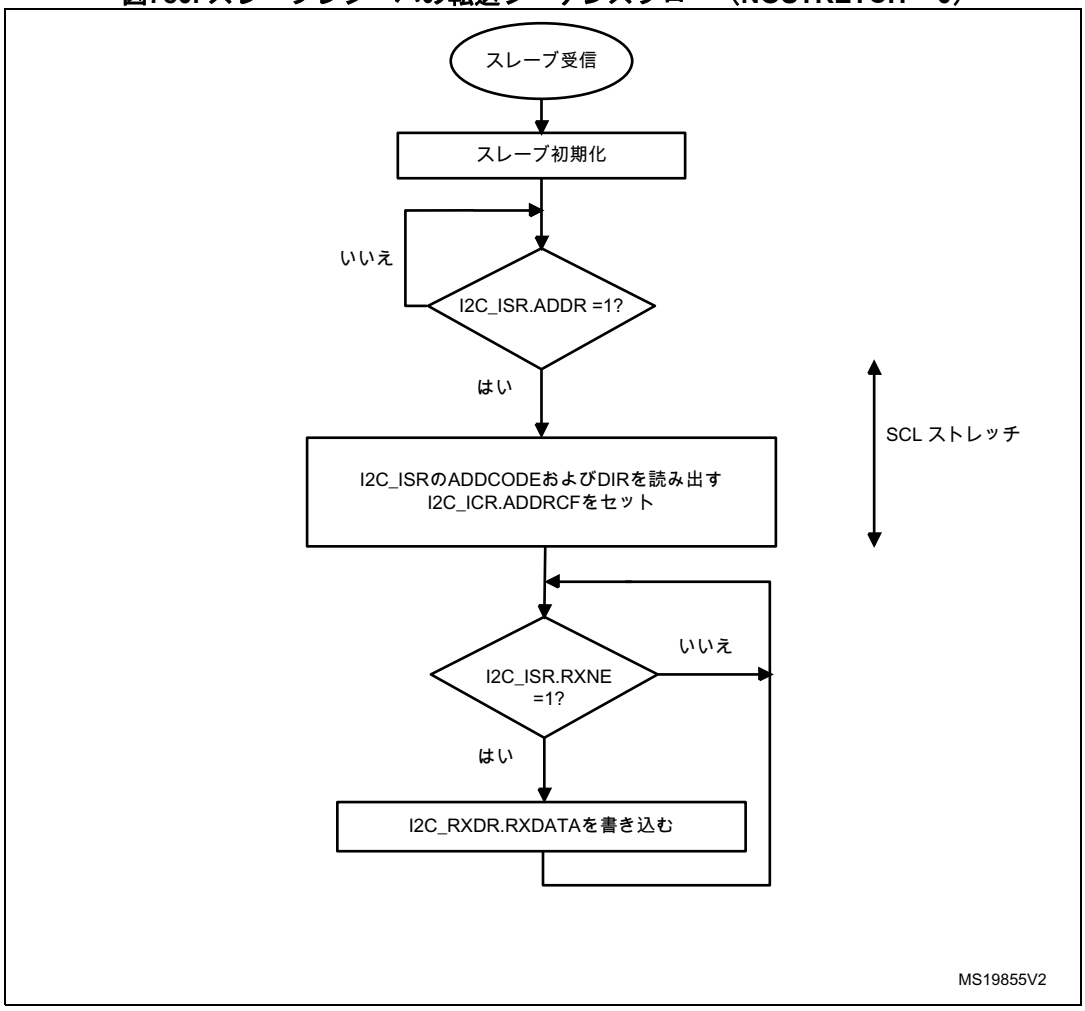


図781. スレーブレシーバの転送シーケンスフロー（NOSTRETCH = 1）

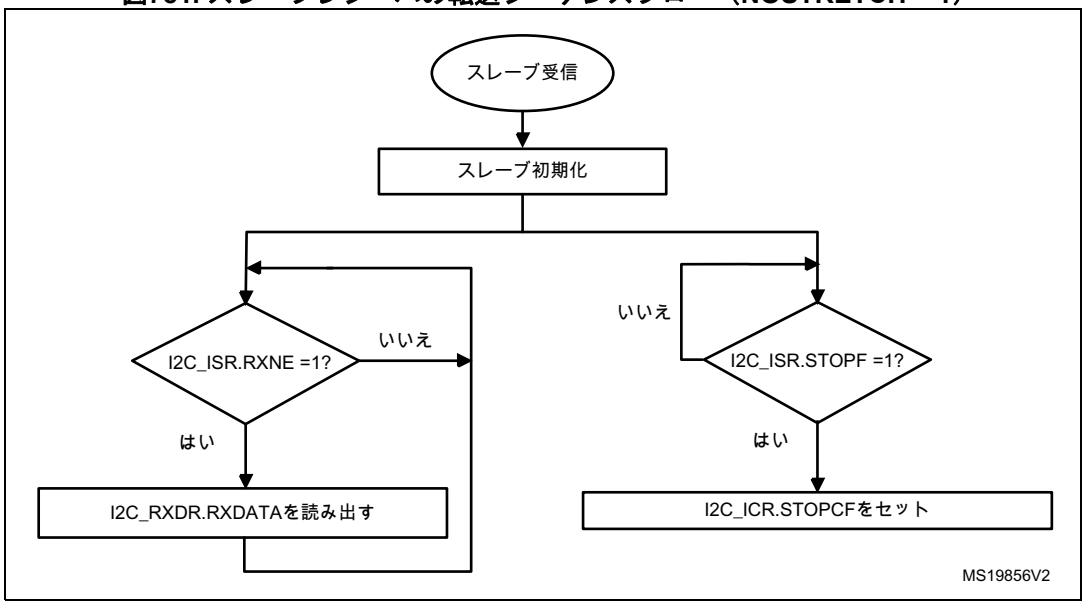
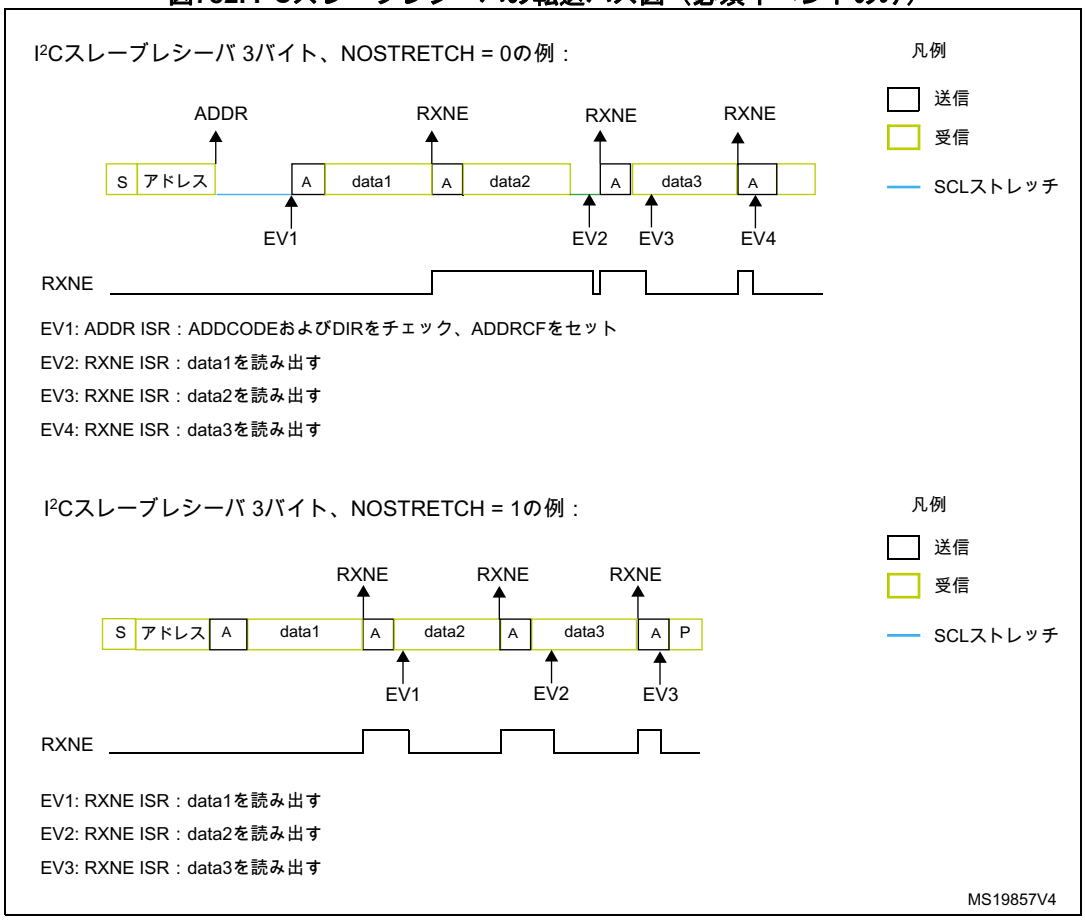


図782. I²Cスレーブレシーバの転送バス図（必須イベントのみ）



65.4.9 I²Cマスタモード

I²Cマスタ初期化

ペリフェラルを有効にする前に、I2C_TIMINGRレジスタのSCLHおよびSCLLビットをセットすることによって、I²Cマスタクロックを設定する必要があります。

STM32CubeMXツールは、I²C設定ウィンドウのI2C_TIMINGRコンテンツを計算し、提供します。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルはSCLLカウンタを使用してカウントされ、SCLローレベル内部検出から開始されます。
- クロックのハイレベルはSCLHカウンタを使用してカウントされ、SCLハイレベル内部検出から開始されます。

I²Cは、SCL立ち下がりエッジ、SCL入力ノイズフィルタ（アナログ + デジタル）、およびI2CxCLKクロックとのSCL同期に応じた遅延 t_{SYNC1} の後にSCLローレベルを検出します。SCLLカウンタがI2C_TIMINGRレジスタのSCLL[7:0] ビットでプログラムされた値に達すると、I²CはSCLをハイレベルにリリースします。

I²Cは、SCL立ち上がりエッジ、SCL入力ノイズフィルタ（アナログ + デジタル）、およびI2CxCLKクロックとのSCL同期に応じた遅延 t_{SYNC2} の後にSCLハイレベルを検出します。SCLHカウンタがI2C_TIMINGRレジスタのSCLH[7:0] ビットでプログラムされた値に達すると、I2CはSCLをローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{[(SCLH+ 1) + (SCLL+ 1)] \times (PRESC+ 1) \times t_{I2CCLK}\}$

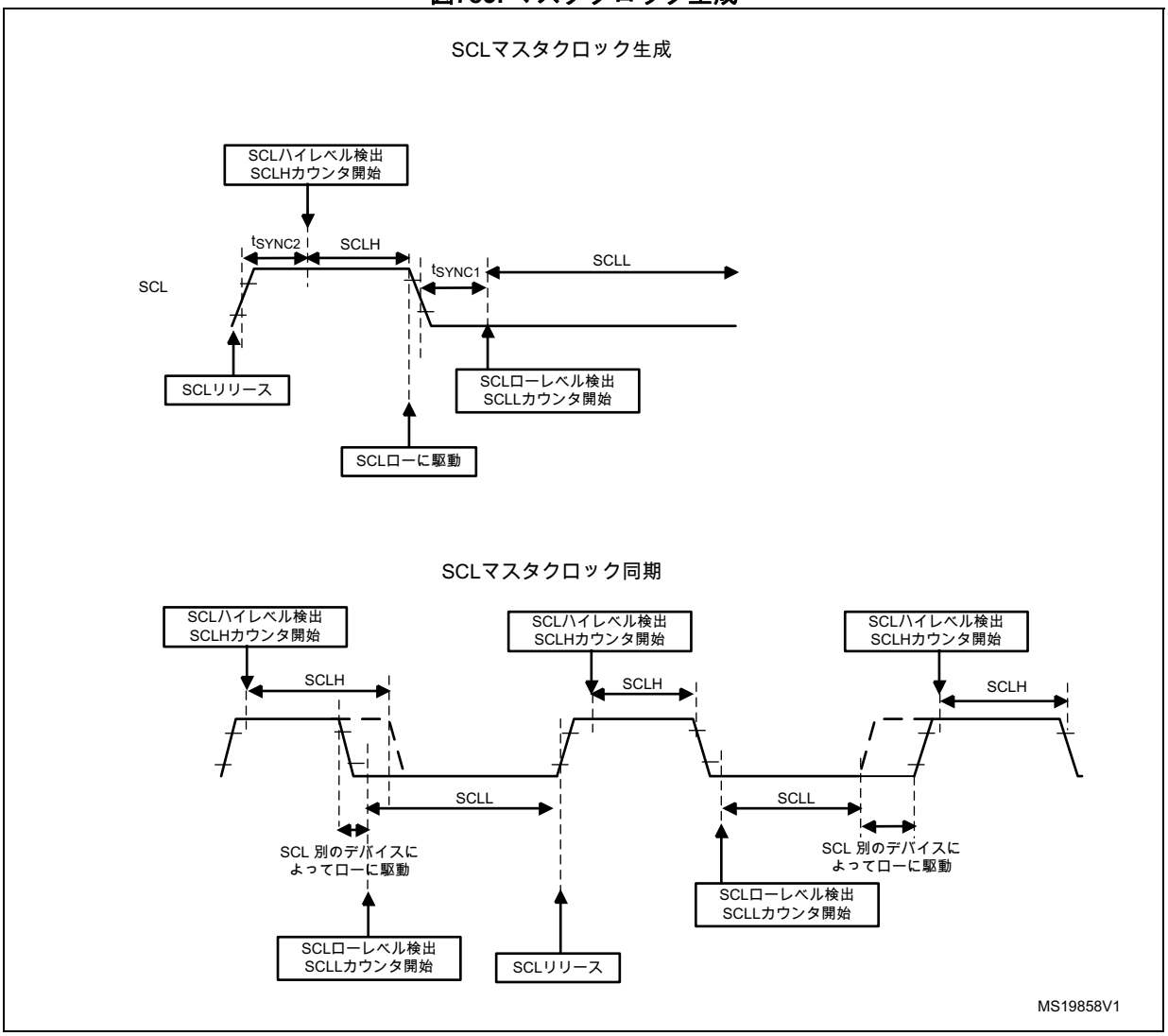
t_{SYNC1} の長さは、次のパラメータに依存します：

- SCL立下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- デジタルフィルタが有効なときの入力遅延：DNF x t_{I2CCLK}
- SCLと i2c_ker_ck クロックの同期による遅延（2 ～ 3 i2c_ker_ck 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL立上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- デジタルフィルタが有効なときの入力遅延：DNF x t_{I2CCLK}
- SCLと i2c_ker_ck クロックの同期による遅延（2 ～ 3 i2c_ker_ck 周期）

図783. マスタクロック生成



注意： **I²CまたはSMBus準拠のためには、マスタクロックは次の表のタイミングを満たす必要があります。**

表655.I²C-SMBus仕様のクロックタイミング

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス (Fm+)		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
f_{SCL}	SCLクロック周波数	-	100	-	400	-	1000	-	100	kHz
$t_{HD:STA}$	（反復）STARTコンディションのホールド時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
$t_{SU:STA}$	反復STARTコンディションのセットアップ時間	4.7	-	0.6	-	0.26	-	4.7	-	
$t_{SU:STO}$	STOPコンディションのセットアップ時間	4.0	-	0.6	-	0.26	-	4.0	-	
t_{BUF}	STOP コンディションと START コンディションの間のバスフリー時間	4.7	-	1.3	-	0.5	-	4.7	-	
t_{LOW}	SCLクロックのロー周期	4.7	-	1.3	-	0.5	-	4.7	-	
t_{HIGH}	SCLクロックの周期	4.0	-	0.6	-	0.26	-	4.0	50	
t_r	SDAおよびSCL信号の立上がり時間	-	1000	-	300	-	120	-	1000	ns
t_f	SDAおよびSCL信号の立下がり時間	-	300	-	300	-	120	-	300	

注： **SCLLは、t_{BUF}および$t_{SU:STA}$タイミングの生成にも使用され、SCLHは、$t_{HD:STA}$および$t_{SU:STO}$タイミングの生成にも使用されます。**

I2C_TIMINGR設定と i2c_ker_ck 周波数の例については、セクション65.4.10を参照してください。

マスタ通信の初期化（アドレスフェーズ）

通信を初期化するためには、I2C_CR2レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります。

- アドレッシングモード（7bitまたは10bit）：ADD10
- 送信されるスレーブアドレス：SADD[9:0]
- 転送方向：RD_WRN
- 10bitアドレスが読み出される場合：HEAD10Rビット。HEAD10Rを設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ（方向の変更の場合）かを示す必要があります。
- 転送されるバイト数：NBYTES[7:0]。バイト数が255バイト以上の場合は、NBYTES[7:0] に0xFFを書き込む必要があります。

次に、I2C_CR2レジスタのSTARTビットをセットする必要があります。STARTビットがセットされているとき、上記のすべてのビットを変更することはできません。

その場合、マスタは、バスがフリーである（BUSY = 0）ことを検出すると、t_{BUF} の遅延後に、自動的にSTARTコンディションとスレーブアドレスを送信します。

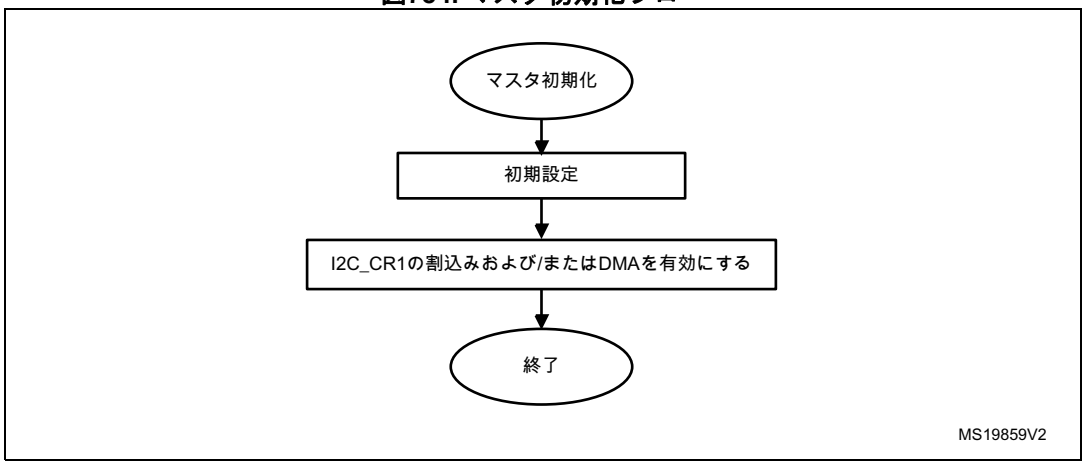
アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

注： **STARTビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。STARTビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。**

10bitアドレッシングモードでは、スレーブアドレスの最初の7bitがスレーブによってNACKされている場合、マスタはACKが受信されるまで自動的にスレーブアドレスの送信を再開します。この場合、NACKをスレーブから受信するかどうか、ADDRCFをセットしてスレーブアドレスの送信を停止する必要があります。
STARTビットがセットされているときに、I2Cがスレーブとしてアドレス指定された場合（ADDR = 1）、I²Cはスレーブモードに切り替わり、STARTビットがクリアされます。

注： **反復スタートコンディションにも同じ手順が適用されます。この場合、BUSY = 1です。**

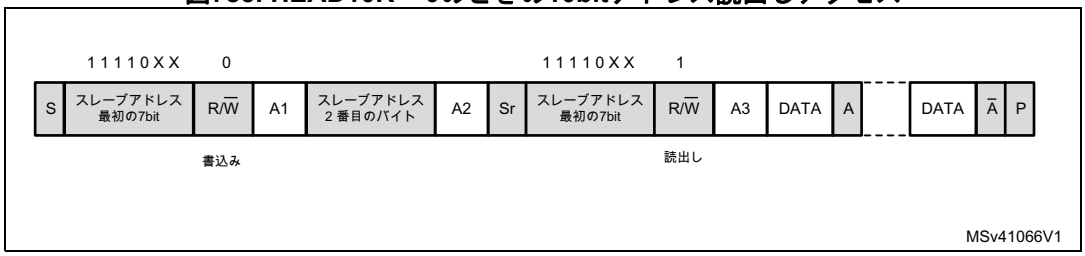
図784. マスタ初期化フロー



10bitアドレススレーブをアドレス指定するマスタレシーバの初期化

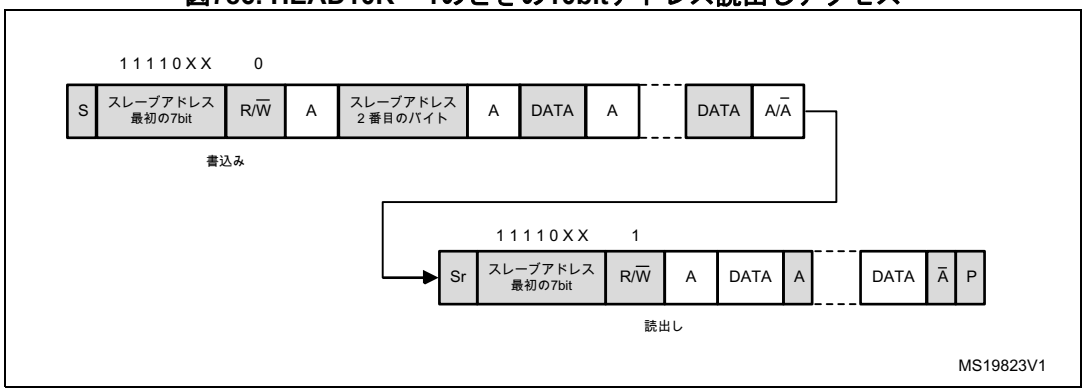
- スレーブアドレスが10bit形式の場合、I2C_CR2レジスタのHEAD10Rビットをクリアすることによって、完全な読出しシーケンスを送信することができます。この場合、マスタは、STARTビットがセットされた後、（Re）START + スレーブアドレス10bitヘッダ書込み + スレーブアドレスの2番目のバイト + REStart + スレーブアドレス10bitヘッダ読出しのような完全なシーケンスを自動的に送信します。

図785. HEAD10R = 0のときの10bitアドレス読出しアクセス



- マスタが10bitアドレススレーブをアドレス指定して、このスレーブアドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その後、HEAD10R = 1で設定された10bit･スレーブ･アドレスで反復開始が設定されます。この場合、マスタはReStart + スレーブアドレス10bitヘッダ読出しのシーケンスを送信します。

図786. HEAD10R = 1のときの10bitアドレス読出しアクセス



マスタトランスミッタ

書込み転送の場合、ACKが受信されたときの9番目のSCLパルス後、各バイトの送信後にTXISフラグがセットされます。

I2C_CR1レジスタのTXIEビットがセットされている場合、TXISイベント時に割込みが生成されます。このフラグは、I2C_TXDRレジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中のTXISイベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が255より大きい場合、I2C_CR2レジスタのRELOADビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTESデータが転送されると、TCRフラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCLラインはローでストレッチされます。

TXISフラグは、NACK受信時にはセットされません。

- RELOAD = 0でNBYTESデータが転送されたとき：
  - 自動終了モード（AUTOEND = 1）では、STOPが自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND = 0）では、TCフラグがセットされ、ソフトウェアアクションを実行するためにSCLラインがローでストレッチされます：

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2レジスタのSTARTビットをセットすることによって、RESTARTコンディションをリクエストできます。STARTビットをセットすると、TCフラグがクリアされ、STARTコンディションがバスに送信されます。

    I2C_CR2レジスタのSTOPビットをセットすることによって、STOPコンディションをリクエストできます。STOPビットをセットすると、TCフラグがクリアされ、STOPコンディションがバスに送信されます。
- NACKが受信された場合：TXISフラグはセットされず、NACK受信後、自動的にSTOPコンディションが送信されます。I2C_ISRレジスタでNACKFフラグがセットされ、NACKIEビットがセットされていた場合は割込みが生成されます。

図787. N ≤ 255バイトの場合の I2Cマスタトランスミッタの転送シーケンスフロー

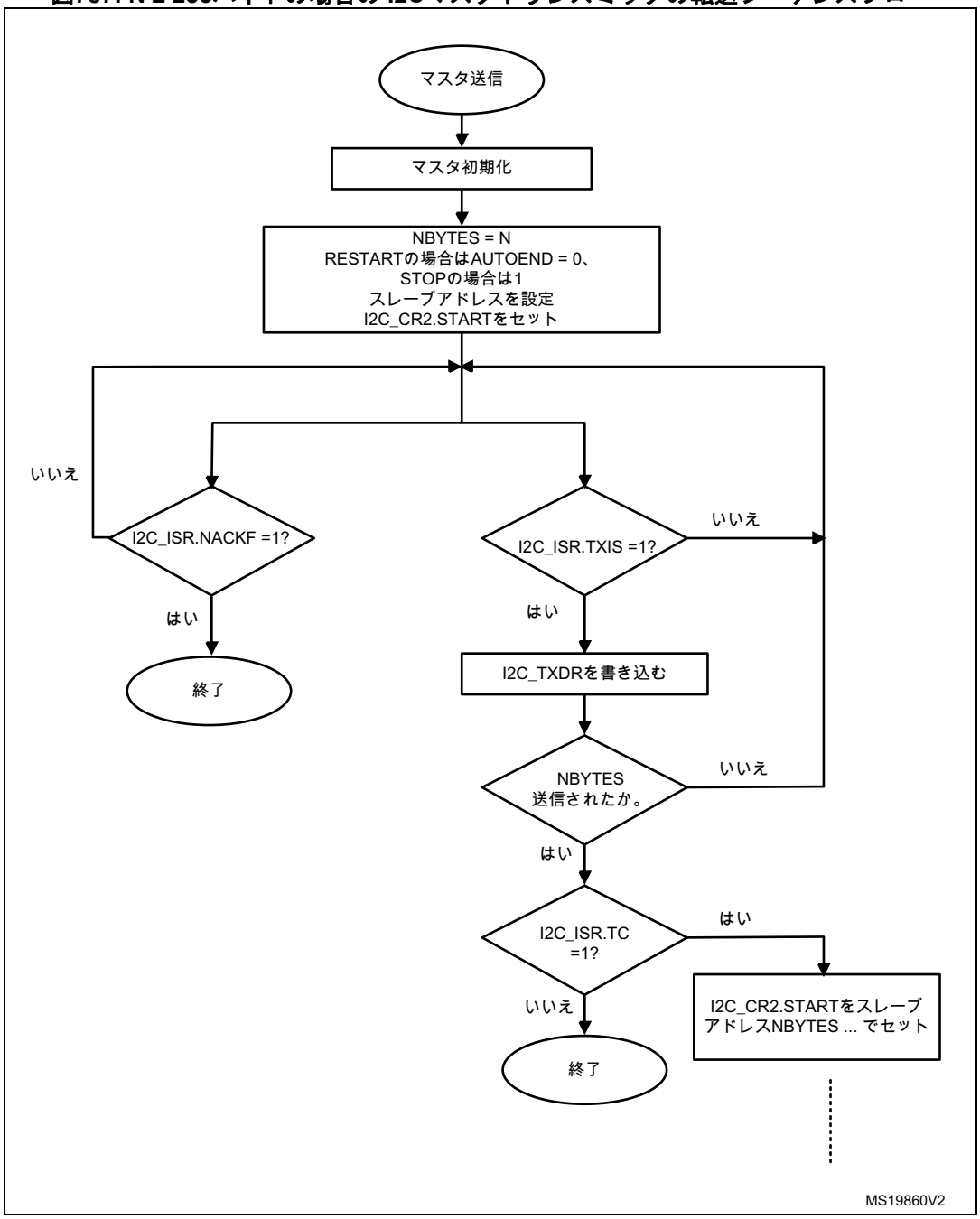


図788. N > 255バイトの場合の I²Cマスタトランスミッタの転送シーケンスフロー

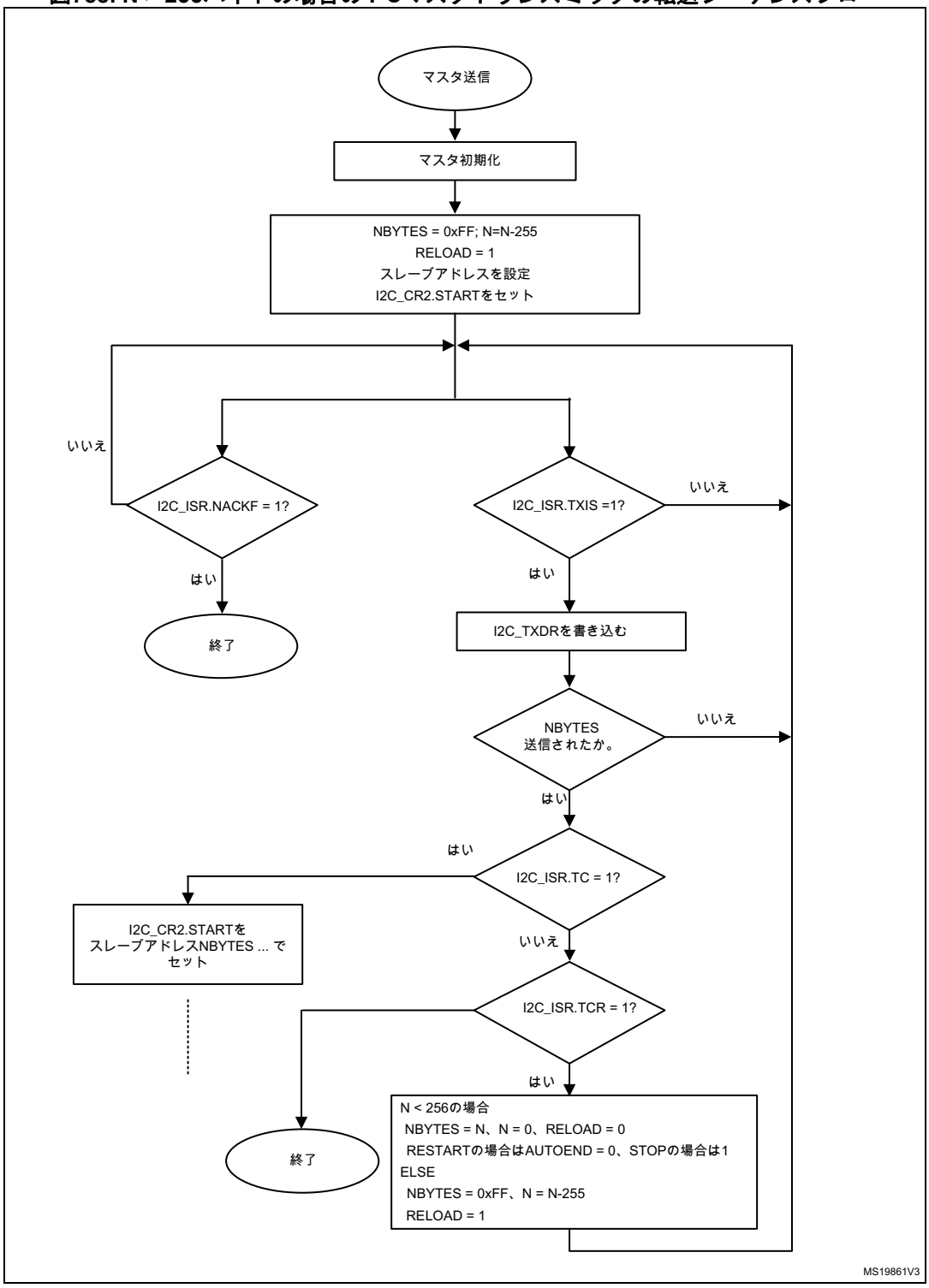
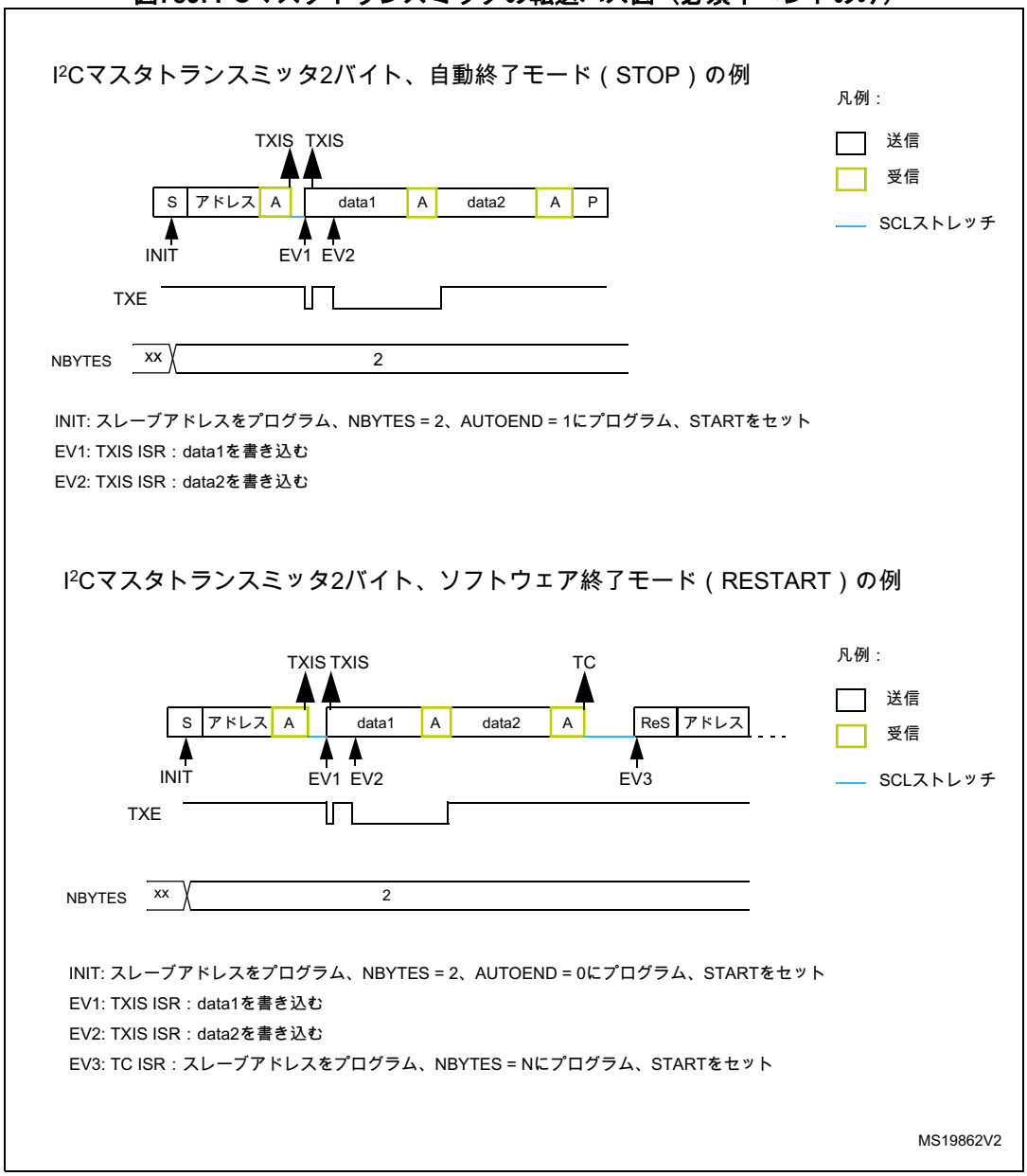


図789. I²Cマスタトランスミッタの転送バス図（必須イベントのみ）



マスタレシーバ

読出し転送の場合、各バイトの受信後や8番目のSCLパルス後にRXNEフラグがセットされます。I2C_CR1レジスタのRXIEビットがセットされている場合、RXNEイベント時に割込みが生成されます。このフラグは、I2C_RXDRが読み出されたときにクリアされます。

受信されるデータバイト数の合計が255より大きい場合、I2C_CR2レジスタのRELOADビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCRフラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCLラインはローでストレッチされます。

- RELOAD = 0でNBYTES[7:0] データが転送されたとき：
  - 自動終了モード（AUTOEND = 1）では、最後の受信バイト後にNACKおよびSTOPが自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND = 0）では、最後の受信バイト後にNACKが自動的に送信され、TCフラグがセットされ、ソフトウェアアクションを実行できるように、SCLラインがローでストレッチされます。

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2レジスタのSTARTビットをセットすることによって、RESTARTコンディションをリクエストできます。STARTビットをセットすると、TCフラグがクリアされ、STARTコンディションとスレーブアドレスがバスに送信されます。

    I2C_CR2レジスタのSTOPビットをセットすることによって、STOPコンディションをリクエストできます。STOPビットをセットすると、TCフラグがクリアされ、STOPコンディションがバスに送信されます。

図790. N ≤ 255バイトの場合の I²Cマスタレシーバの転送シーケンスフロー

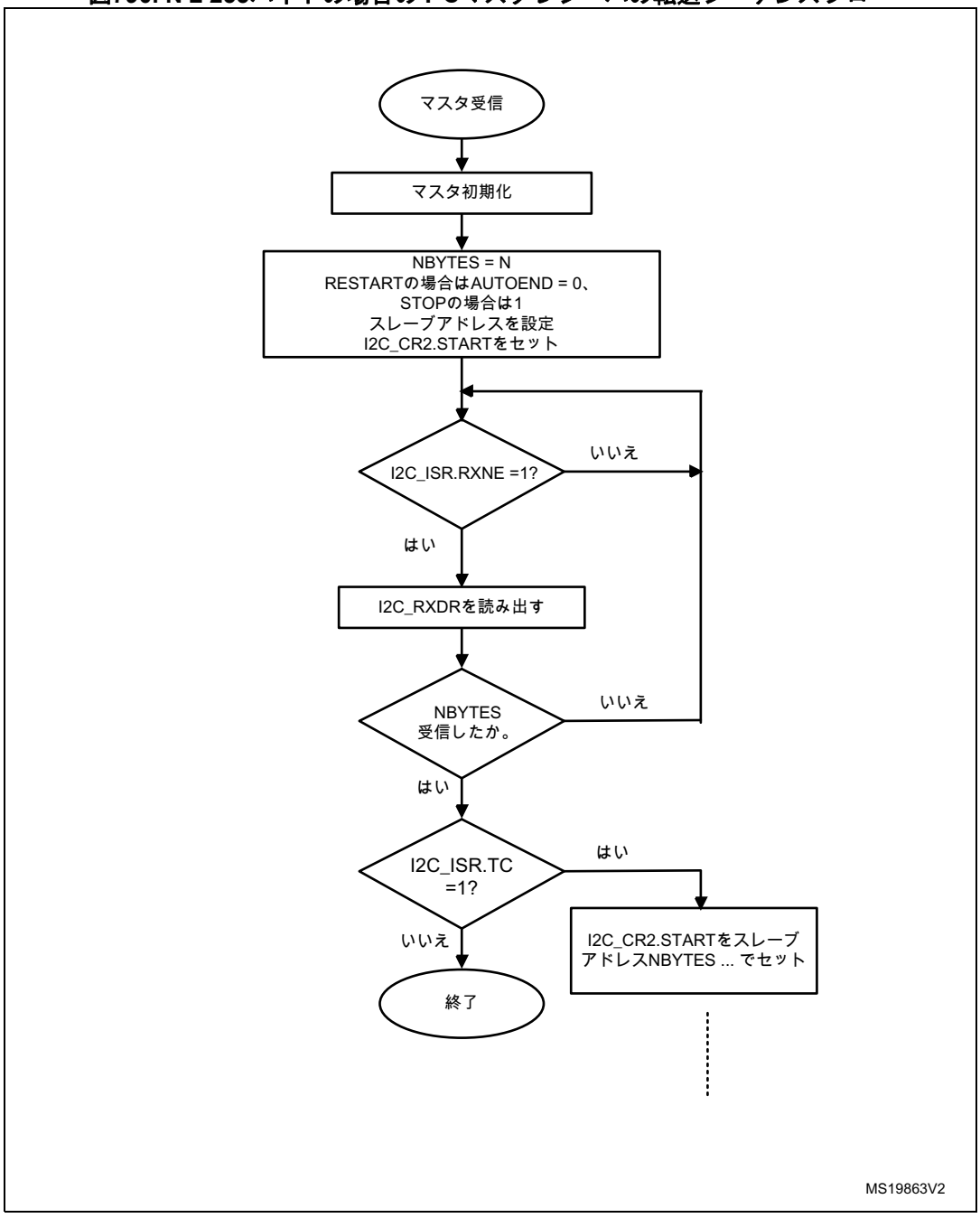


図791. N > 255バイトの場合の I²Cマスタレシーバの転送シーケンスフロー

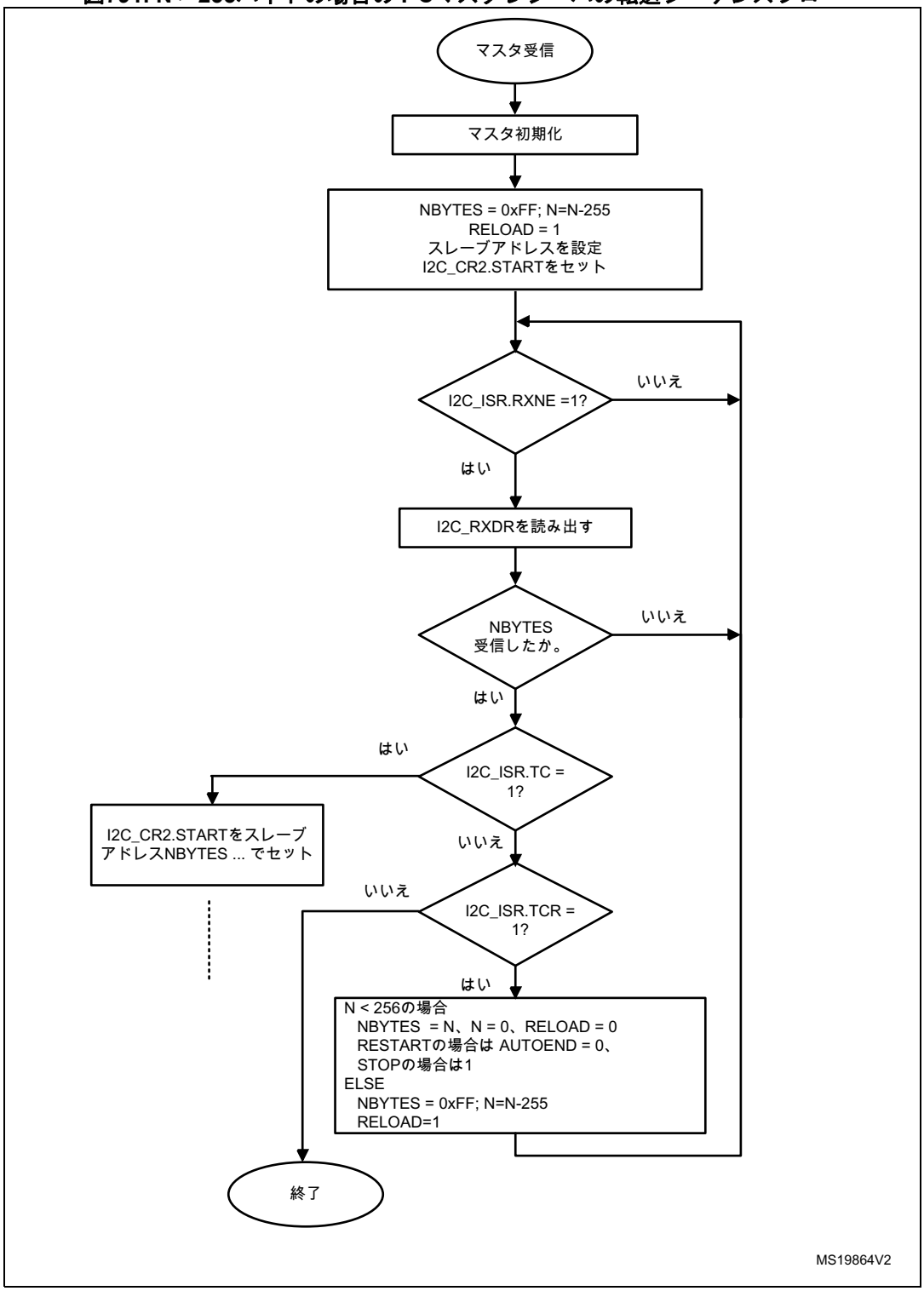
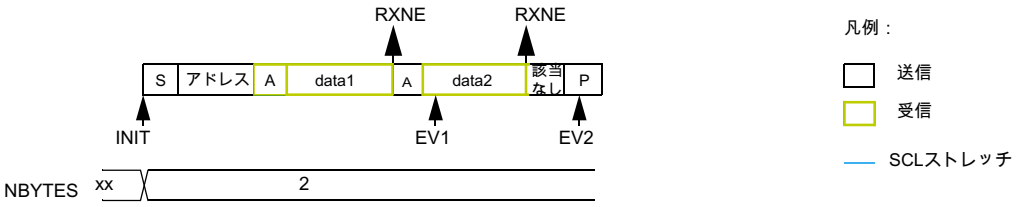


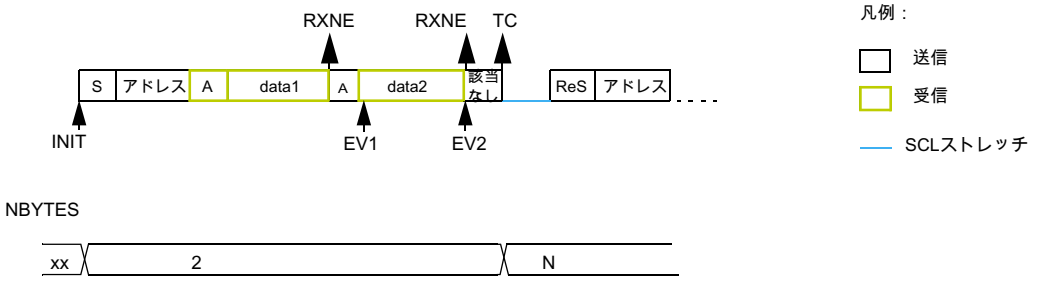
図792. I²Cマスタレシーバの転送バス図（必須イベントのみ）

I²Cマスタレシーバ2バイト、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=1にプログラム、STARTをセット
EV1：RXNE ISR：data1を読み出す
EV2：RXNE ISR：data2を読み出す

I²Cマスタレシーバ2バイト、ソフトウェア終了モード（RESTART）の例



INIT：スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=0にプログラム、STARTをセット
EV1：RXNE ISR：data1を読み出す
EV2：RXNE ISR：data2を読み出す
EV3：TC ISR：スレーブアドレスをプログラム、NBYTES = Nにプログラム、STARTをセット

MS19865V1

65.4.10 I2C_TIMINGRレジスタの設定例

次の表に、I²C仕様に準拠したタイミングを得るための I2C_TIMINGRをプログラムする方法の例を示します。より正確な設定値を得るには、STM32CubeMXツール（I2C設定ウィンドウ）を使用してください。

表656.f_{I2CCLK} = 8 MHzでのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10kHz	100kHz	400kHz	500kHz
PRESC	0x1	0x1	0x0	0x0
SCLL	0xC7	0x13	0x9	0x6
t_{SCLL}	200 x 250ns = 50µs	20 x 250ns = 5.0µs	10 x 125ns = 1250ns	7 x 125ns = 875ns
SCLH	0xC3	0xF	0x3	0x3
t_{SCLH}	196 x 250ns = 49µs	16 x 250ns = 4.0µs	4 x 125ns = 500ns	4 x 125ns = 500ns
t_{SCL}(1)	～100µs(2)	～10µs (2)	～2500ns(3)	～2000ns(4)
SDADEL	0x2	0x2	0x1	0x0
t_{SDADEL}	2 x 250ns = 500ns	2 x 250ns = 500ns	1 x 125ns = 125ns	0ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250ns = 1250ns	5 x 250ns = 1250ns	4 x 125ns = 500ns	2 x 125ns = 250ns

1. t_{SCL} は、SCL内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 nsです。例：t_{SYNC1} + t_{SYNC2} = 1000ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 nsです。例：t_{SYNC1} + t_{SYNC2} = 750ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 nsです。例：t_{SYNC1} + t_{SYNC2} = 655ns

表657.f_{I2CCLK} = 16 MHzでのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10kHz	100kHz	400kHz	1000kHz
PRESC	0x3	0x3	0x1	0x0
SCLL	0xC7	0x13	0x9	0x4
t_{SCLL}	200 x 250ns = 50µs	20 x 250ns = 5.0µs	10 x 125ns = 1250ns	5 x 62.5ns = 312.5ns
SCLH	0xC3	0xF	0x3	0x2
t_{SCLH}	196 x 250ns = 49µs	16 x 250ns = 4.0µs	4 x 125ns = 500ns	3 x 62.5ns = 187.5ns
t_{SCL}(1)	～100µs(2)	～10µs (2)	～2500ns(3)	～1000ns(4)
SDADEL	0x2	0x2	0x2	0x0
t_{SDADEL}	2 x 250ns = 500ns	2 x 250ns = 500ns	2 x 125ns = 250ns	0ns
SCLDEL	0x4	0x4	0x3	0x2
t_{SCLDEL}	5 x 250ns = 1250ns	5 x 250ns = 1250ns	4 x 125ns = 500ns	3 x 62.5ns = 187.5ns

1. t_{SCL} は、SCL内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 nsです。例：t_{SYNC1} + t_{SYNC2} = 1000ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 nsです。例：t_{SYNC1} + t_{SYNC2} = 750ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 nsです。例：t_{SYNC1} + t_{SYNC2} = 500ns

65.4.11 SMBus固有の機能

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3 を参照してください）。

概要

システム管理バス（SMBus）は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる2線インタフェースです。I²Cの動作原理に基づきます。SMBusにより、システムおよびパワーマネージメント関連のタスク向けの制御バスが実現できます。

このペリフェラルは、SMBus仕様（http://smbus.org）と互換性があります。

システム管理バス仕様では、3種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムのCPUにメインインタフェースを提供する特殊なマスタです。ホストは、マスタ/スレーブとすることができ、SMBusホスト通知プロトコルをサポートする必要があります。システム内では、ただ1つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

バスプロトコル

特定のデバイスについて、11の可能なコマンドプロトコルがあります。デバイスは、11のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、およびBlock Write-Block Read Process Callです。これらのプロトコルは、ユーザのソフトウェアによって実装する必要があります。

これらのプロトコルの詳細については、SMBus仕様（http://smbus.org）を参照してください。

アドレス解決プロトコル（ARP）

SMBusスレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子（UDID）を実装する必要があります。128bitの番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル（ARP）をサポートします。SMBusデバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1レジスタのSMBDENビットをセットすることによって有効になります。ARPコマンドは、ユーザのソフトウェアによって実装する必要があります。

ARPサポートのために、スレーブモードでアービトレーションも行われます。

SMBusアドレス解決プロトコルの詳細については、SMBus仕様（http://smbus.org）を参照してください。

受信コマンドおよびデータ確認応答制御

SMBusレシーバは、受信した各コマンドまたはデータをNACKできなければなりません。スレーブモードでACK制御を可能にするためには、I2C_CR1レジスタのSBCビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、スレーブバイト制御モードを参照してください。

ホスト通知プロトコル

このペリフェラルは、I2C_CR1レジスタのSMBHENビットをセットすることによって、Host Notify（ホスト通知）プロトコルをサポートします。この場合、ホストはSMBusホストアドレス（0b0001 000）を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBusアラート

SMBus ALERTオプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストのSMBALERT# ピンを通じてホストに信号を送信します。ホストは、割込みを処理し、アラート応答アドレス（0b0001 100）を通じて全SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブデバイスとして設定されたとき（SMBHEN = 0）、I2C_CR1レジスタのALERTENビットをセットすることによって、SMBAピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき（SMBHEN = 1）、SMBAピンで立ち下がりエッジが検出され、ALERTEN = 1のとき、I2C_ISRレジスタのALERTフラグがセットされます。I2C_CR1レジスタのERRIEビットがセットされている場合は、割込みが生成されます。ALERTEN = 0のときには、外部SMBAピンがローの場合でも、ALERTラインはハイとみなされます。

SMBus ALERTピンが不要な場合には、ALERTEN = 0の場合、SMBAピンを標準GPIOとして使用できます。

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus仕様にパケットエラーチェックメカニズムが導入されました。パケットエラーチェックは、各メッセージ転送の終わりにパケットエラーコード（PEC）を付加することによって実装されます。PECは、すべてのメッセージバイト（アドレスと読出し／書込みビットを含む）に対して$C(x) = x_8 + x^2 + x + 1$ CRC-8多項式を使用して計算されます。

ペリフェラルにはハードウェアPEC計算器が組み込まれており、受信バイトがハードウェアによって計算されたPECに一致しないときには自動的に否認応答を送信できます。

タイムアウト

このペリフェラルは、SMBus仕様で定義された3つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

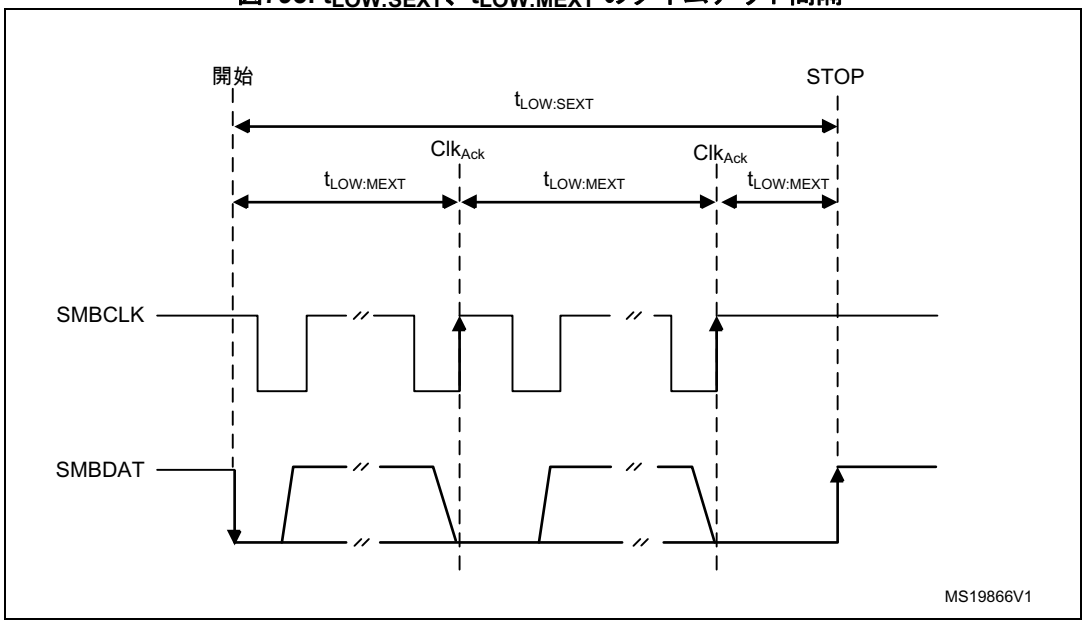
表658.SMBusタイムアウト仕様

記号	パラメータ	リミット		単位
		最小値	最大値	
$t_{TIMEOUT}$	クロックロータイムアウト検出	25	35	ms
$t_{LOW:SEXT}$(1)	累積クロックロー延長時間（スレーブデバイス）	-	25	
$t_{LOW:MEXT}$(2)	累積クロックロー延長時間（マスタデバイス）	-	10	

1. $t_{LOW:SEXT}$ は、特定のスレーブデバイスが初めのSTARTからSTOPまでの1つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が$t_{LOW:SEXT}$より大きくなることがあります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。

2. $t_{LOW:MEXT}$ は、マスタデバイスがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、またはACK-to-STOPから定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が$t_{LOW:MEXT}$ より大きくなることがあります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図793. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が$t_{HIGH,MAX}$ より大きいt_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます（ 表653を参照）。

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLKまたはSMBDATラインで状態遷移を検出しなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

65.4.12 SMBus初期化

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照）。

SMBus通信を行うためには、I²C初期化に加えて、他にも特定の初期化を行う必要があります。

受信コマンドおよびデータ確認応答制御（スレーブモード）

SMBus レシーバは、受信した各コマンドまたはデータをNACKできなければなりません。スレーブ·モードでACK制御を可能にするには、I2C_CR1レジスタのSBCビットをセットすることによって、スレーブ·バイト制御モードを有効にする必要があります。詳細については、スレーブバイト制御モードを参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定のSMBusアドレスを有効にしてください。詳細については、バスアイドル検出を参照してください。

- SMBusデバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1レジスタのSMBDENビットをセットすることによって有効になります。
- SMBusホストアドレス（0b0001 000）は、I2C_CR1レジスタのSMBHENビットをセットすることによって有効になります。
- アラート応答アドレス（0b0001100）は、I2C_CR1 レジスタのALERTENビットをセットすることによって有効になります。

パケットエラーチェック

PEC計算を有効にするには、I2C_CR1レジスタのPECENビットをセットします。その場合、PEC転送はハードウェアバイトカウンタ（I2C_CR2レジスタのNBYTES[7:0]）を使用して管理されます。PECENビットは、I2Cを有効にする前に設定する必要があります。

PEC転送はハードウェアバイトカウンタによって管理されるので、スレーブモードでSMBusとインタフェースするときにはSBCビットをセットする必要があります。PECは、PECBYTEビットがセットされ、RELOADビットがクリアされたとき、NBYTES - 1データの転送後に転送されます。RELOADがセットされた場合、PECBYTEは効果がありません。

注意： **I2Cが有効なときには、PECEN設定の変更はできません。**

表659.SMBusのPEC設定

モード	SBCビット	RELOADビット	AUTOENDビット	PECBYTEビット
マスタTx/Rx NBYTES + PEC+ STOP	x	0	1	1
マスタTx/Rx NBYTES + PEC + ReSTART	x	0	0	1
スレーブTx/RxとPEC	1	0	x	1

タイムアウト検出

タイムアウト検出は、I2C_TIMEOUTRレジスタのTIMOUTENおよびTEXTENビットをセットすることによって有効になります。SMBus仕様で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- $t_{TIMEOUT}$ チェック

  $t_{TIMEOUT}$ チェックを有効にするためには、12bitのTIMEOUTA[11:0] ビットを$t_{TIMEOUT}$ パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCLローレベルタイムアウトを検出するためには、TIDLEビットを0に設定する必要があります。

  その場合、タイマは、I2C_TIMEOUTRレジスタのTIMOUTENをセットすることによって有効になります。

  SCLが (TIMEOUTA1) x 2048 x t_{I2CCLK} より長い間ローのままであった場合、I2C_ISRレジスタのTIMEOUTフラグがセットされます。

  表660を参照してください。

注意： **TIMEOUTENビットがセットされているときには、TIMEOUTA[11:0] ビットおよびTIDLEビットの設定変更はできません。**

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ チェック

  ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12bitのTIMEOUTBタイマは、スレーブの場合は $t_{LOW:SEXT}$ をチェックするために、マスタの場合は $t_{LOW:MEXT}$ をチェックするために、設定する必要があります。規格では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

  その場合、タイマは、I2C_TIMEOUTRレジスタのTEXTENビットをセットすることによって有効になります。

  SMBusペリフェラルが、（TIMEOUTB+1）x 2048 x t_{I2CCLK} より長い間、および バスアイドル検出セクションで述べられているタイムアウト間隔で、累積SCLストレッチを実行した場合、I2C_ISRレジスタのTIMEOUTフラグがセットされます。

  表661を参照してください。

注意： **TEXTENビットがセットされているときには、TIMEOUTB設定変更はできません。**

バスアイドル検出

t_{IDLE} チェックを有効にするためには、12bitのTIMEOUTA[11:0] フィールドをt_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCLおよびSDAハイレベルタイムアウトを検出するためには、TIDLEビットを1に設定する必要があります。

その場合、タイマは、I2C_TIMEOUTRレジスタのTIMEOUTENビットをセットすることによって有効になります。

SCLおよびSDAの両方のラインが（TIMEOUTA+1）x 4 x t_{I2CCLK} より長い間ハイのままであった場合、I2C_ISRレジスタのTIMEOUTフラグがセットされます。

表662を参照してください。

注意： **TIMEOUTENがセットされているときに、TIMEOUTAおよびTIDLE設定を変更することはできません。**

65.4.13 SMBus：I2C_TIMEOUTRレジスタの設定例

このセクションは、SMBus機能がサポートされるときにのみ適用されます。セクション65.3 を参照してください。

- $t_{TIMEOUT}$ の最大時間を25 msに設定：

表660.TIMEOUTAの設定例（最大$t_{TIMEOUT}$ = 25ms）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLEビット	TIMEOUTENビット	$t_{TIMEOUT}$
8MHz	0x61	0	1	98 x 2048 x 125ns = 25ms
16MHz	0xC3	0	1	196 x 2048 x 62.5ns = 25ms

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ の最大時間を8 msに設定：

表661.TIMEOUTBの設定例

f_{I2CCLK}	TIMEOUTB[11:0] ビット	TEXTENビット	$t_{LOW:EXT}$
8MHz	0x1F	1	32 x 2048 x 125ns = 8ms
16MHz	0x3F	1	64 x 2048 x 62.5ns = 8ms

- t_{IDLE} の最大時間を50 µsに設定

表662.TIMEOUTAの設定例（最大t_{IDLE} = 50µs）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLEビット	TIMEOUTENビット	t_{TIDLE}
8MHz	0x63	1	1	100 x 4 x 125ns = 50µs
16MHz	0xC7	1	1	200 x 4 x 62.5ns = 50µs

65.4.14 SMBusスレーブモード

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照してください）。

I²Cスレーブ転送管理（セクション65.4.8を参照）に加えて、SMBusをサポートするために、いくつか追加のソフトウェア·フローがあります。

SMBusスレーブトランスミッタ

IPがSMBusで使用されるときには、SBCは、プログラムされたデータバイト数の終わりのPEC送信を有効にするため、1にプログラムする必要があります。PECBYTEビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数にはPEC送信が含まれます。その場合、TXIS割込みの合計数はNBYTES - 1であり、NBYTES - 1のデータ転送後にマスタが追加のバイトをリクエストすると、I2C_PECRレジスタの内容が自動的に送信されます。

注意： **PECBYTEビットは、RELOADビットがセットされているときには効果がありません。**

図794. Nバイト + PECの場合のSMBusスレーブトランスミッタの転送シーケンスフロー

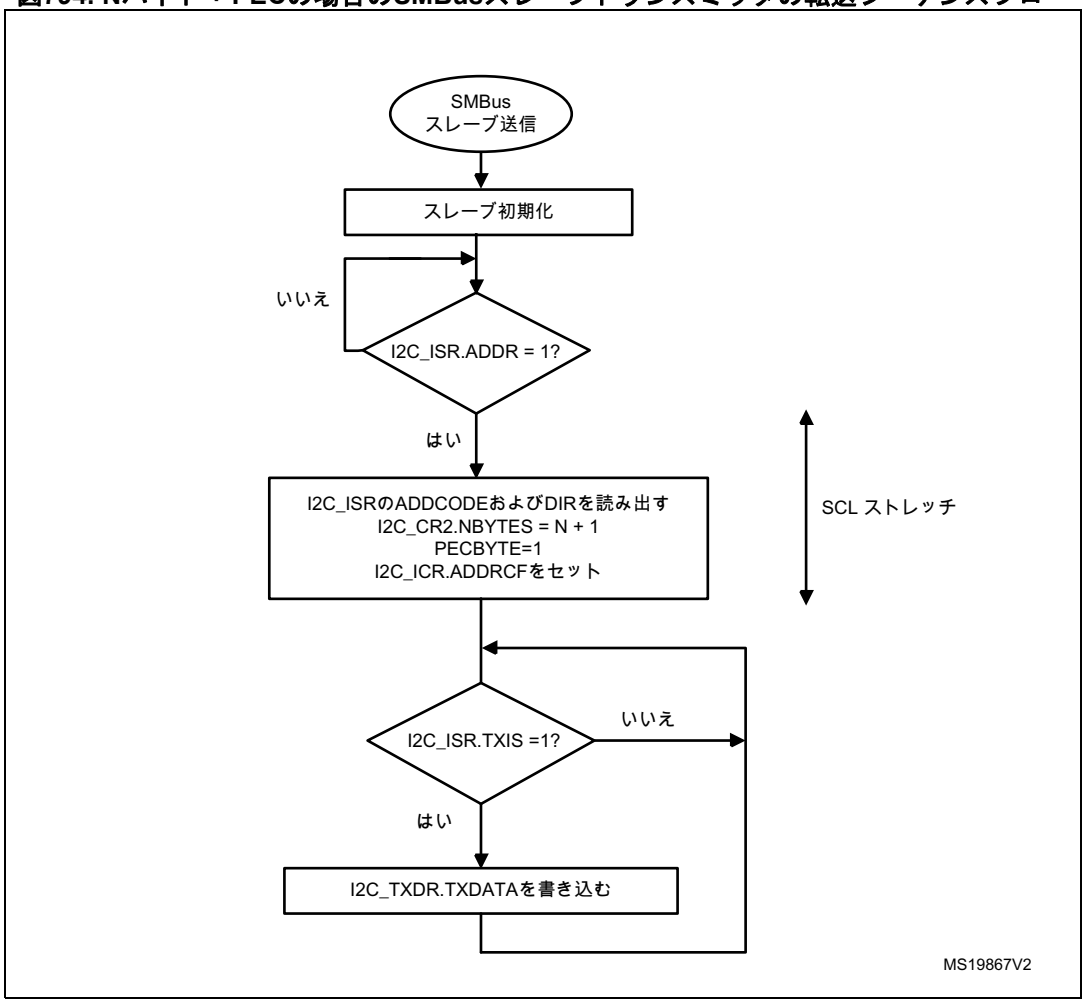
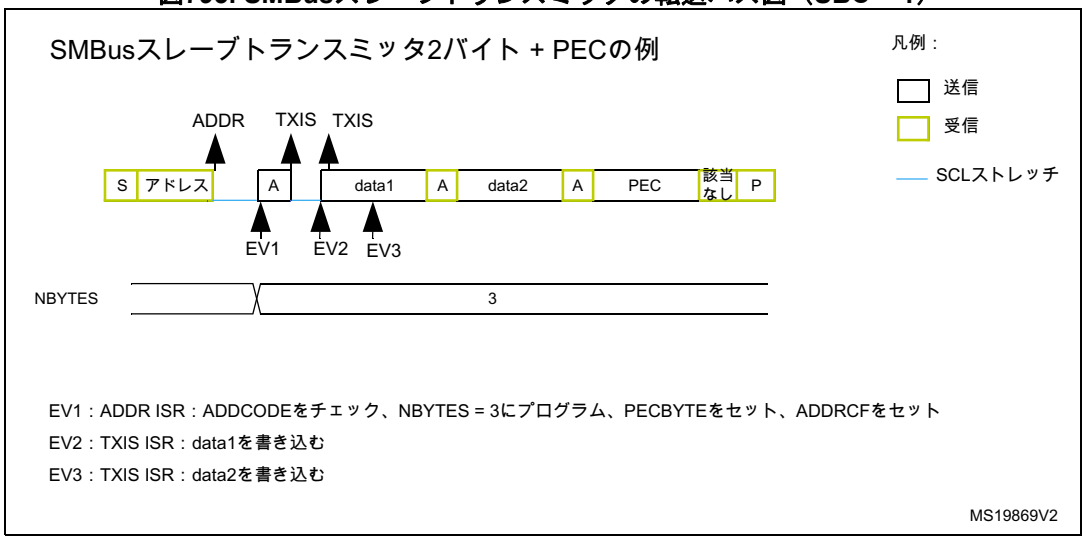


図795. SMBusスレーブトランスミッタの転送バス図（SBC = 1）



SMBusスレーブレシーバ

I2CがSMBusモードで使用されるときには、SBCは、プログラムされたデータバイト数の終わりのPECチェックを可能にするため、1にプログラムする必要があります。各バイトのACK制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD = 1）。詳細については、スレーブバイト制御モードを参照してください。

PECバイトをチェックするためには、RELOADビットをクリアして、PECBYTEビットをセットする必要があります。この場合、NBYTES - 1のデータが受信された後、次の受信バイトが内部 I2C_PECR レジスタの内容と比較されます。ACKビットの値にかかわらず、比較が一致しなかった場合はNACKが自動的に生成され、比較が一致した場合はACKが自動的に生成されます。PECバイトが受信されると、他のデータと同様に I2C_RXDRレジスタにコピーされ、RXNEフラグがセットされます。

PEC不一致の場合、PECERRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

ACKソフトウェア制御が不要な場合は、PECBYTE = 1をプログラムし、同じ書込み操作でNBYTESを連続フローで受信するバイト数にプログラムします。NBYTES - 1が受信された後、次の受信バイトがPECであるかどうかがチェックされます。

注意： **PECBYTEビットは、RELOADビットがセットされているときには効果がありません。**

図796. Nバイト + PECの場合のSMBusスレーブレシーバの転送シーケンスフロー

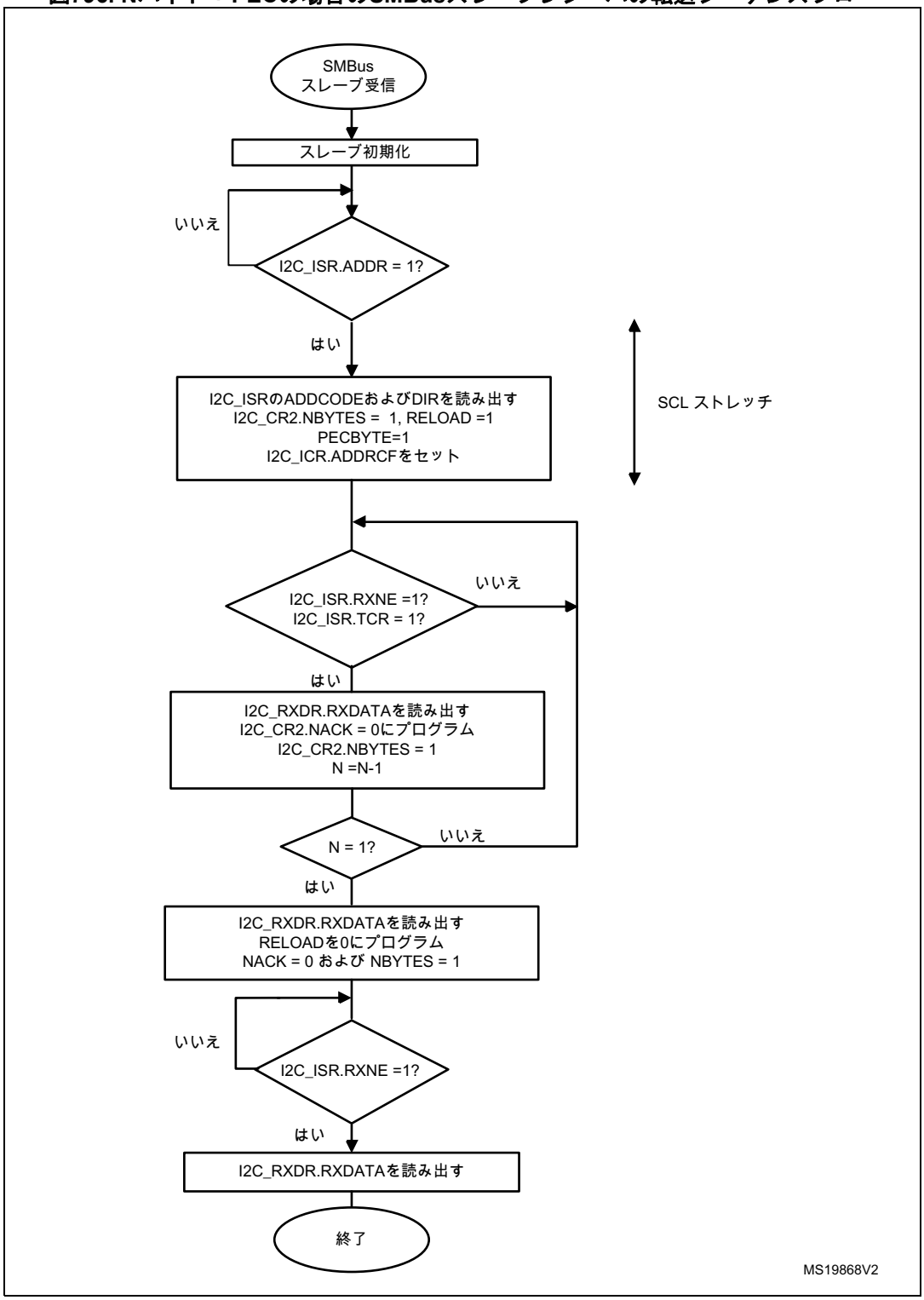
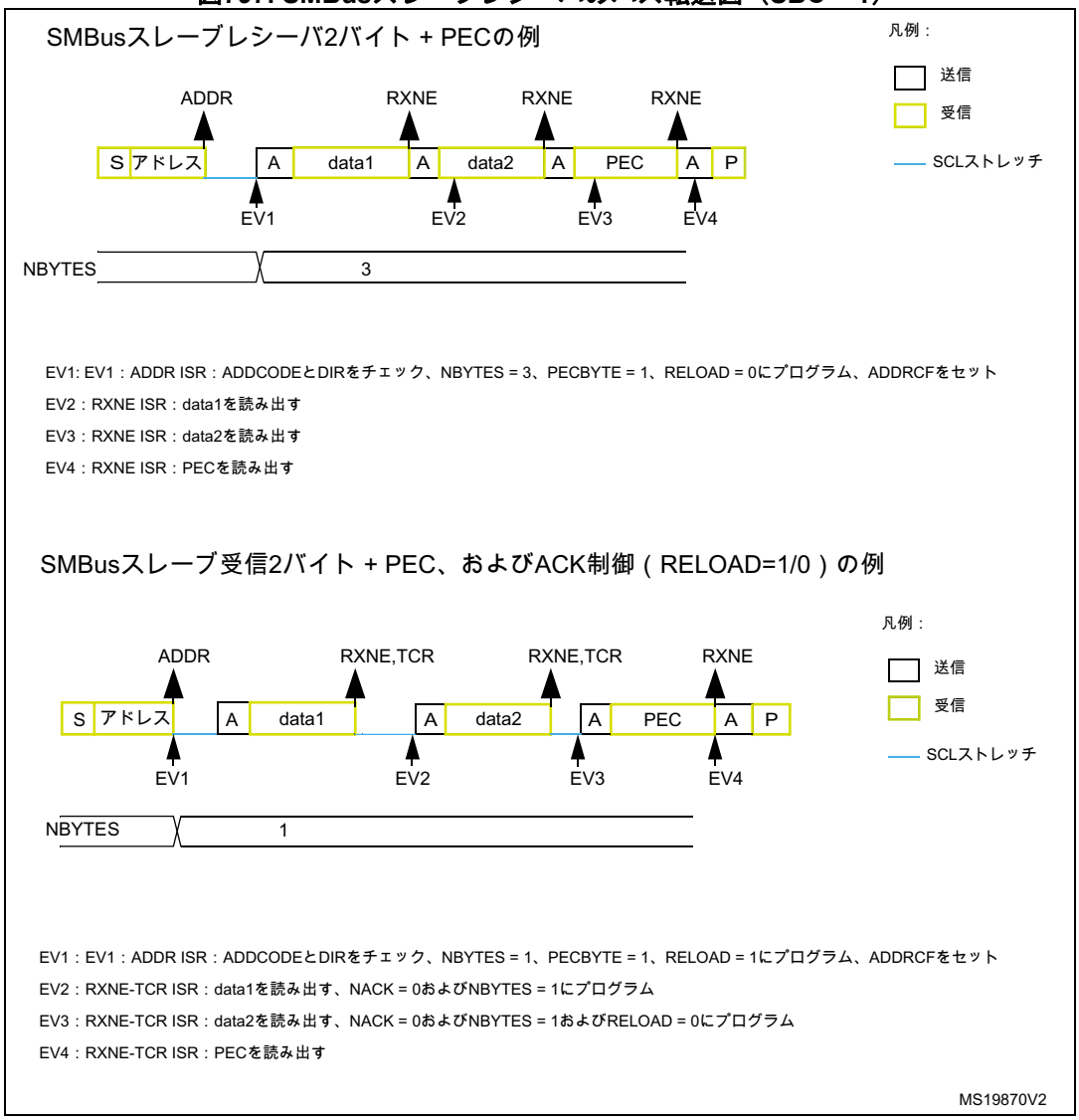


図797. SMBusスレーブレシーバのバス転送図（SBC = 1）



このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照してください）。

I²Cマスタ転送管理（セクション65.4.9を参照）に加えて、SMBusをサポートするために、いくつか追加のソフトウェア·フローがあります。

SMBusマスタ トランスミッタ

SMBusマスタがPECを送信したいときには、STARTビットをセットする前に、PECBYTEビットをセットする必要があり、バイト数をNBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS割込みの合計数はNBYTES - 1になります。したがって、NBYTES = 0x1のときにPECBYTEビットがセットされた場合、I2C_PECRレジスタの内容が自動的に送信されます。

SMBusマスタがPEC後にSTOPコンディションを送信したい場合は、自動終了モードを選択してください（AUTOEND = 1）。この場合、PEC送信に続いて、STOPコンディションが自動的に送信されます。

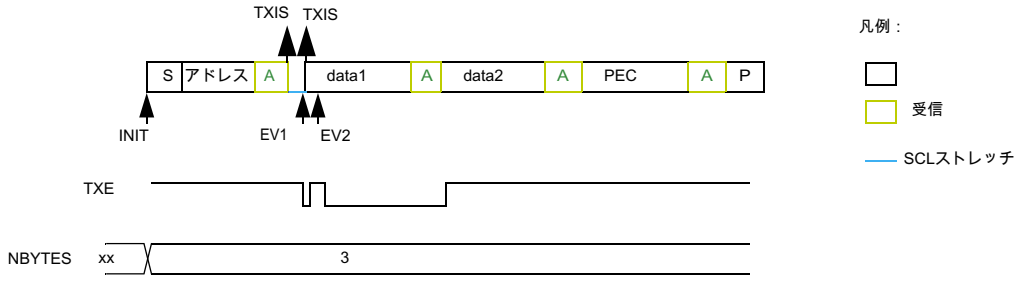
SMBusマスタがPEC後にRESTARTコンディションを送信したい場合は、ソフトウェアモードを選択してください（AUTOEND = 0）。この場合、NBYTES - 1が送信されると、PEC送信後に I2C_PECR

レジスタの内容が送信され、TCフラグがセットされ、SCLラインローをストレッチします。RESTARTコンディションをTC割込みサブルーチンでプログラムする必要があります。

注意： **PECBYTEビットは、RELOADビットがセットされているときには効果がありません。**

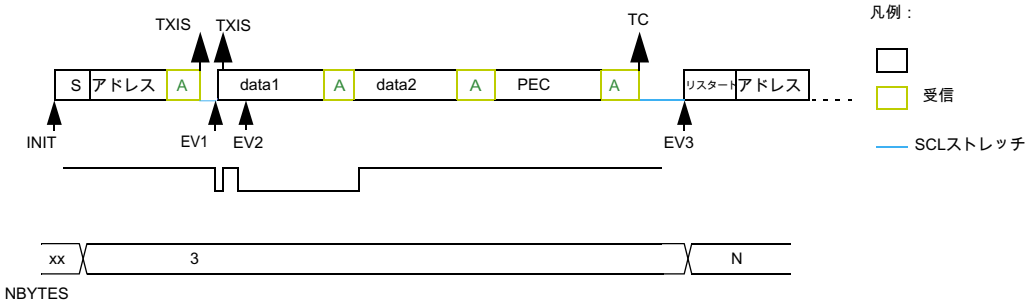
図798. SMBusマスタトランスミッタのバス転送図

SMBusマスタトランスミッタ 2 バイト + PEC、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND = 1にプログラム、PECBYTEをセット、STARTをセット
EV1：TXIS ISR：data1を書き込む
EV2：TXIS ISR：data2を書き込む

SMBusマスタトランスミッタ 2 バイト + PEC、ソフトウェア終了モード（RESTART）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND = 0にプログラム、PECBYTEをセット、STARTをセット
EV1：TXIS ISR：data1を書き込む
EV2：TXIS ISR：data2を書き込む
EV3：TC ISR：スレーブアドレスをプログラム、NBYTES = Nにプログラム、STRDをセット

MS19871V2

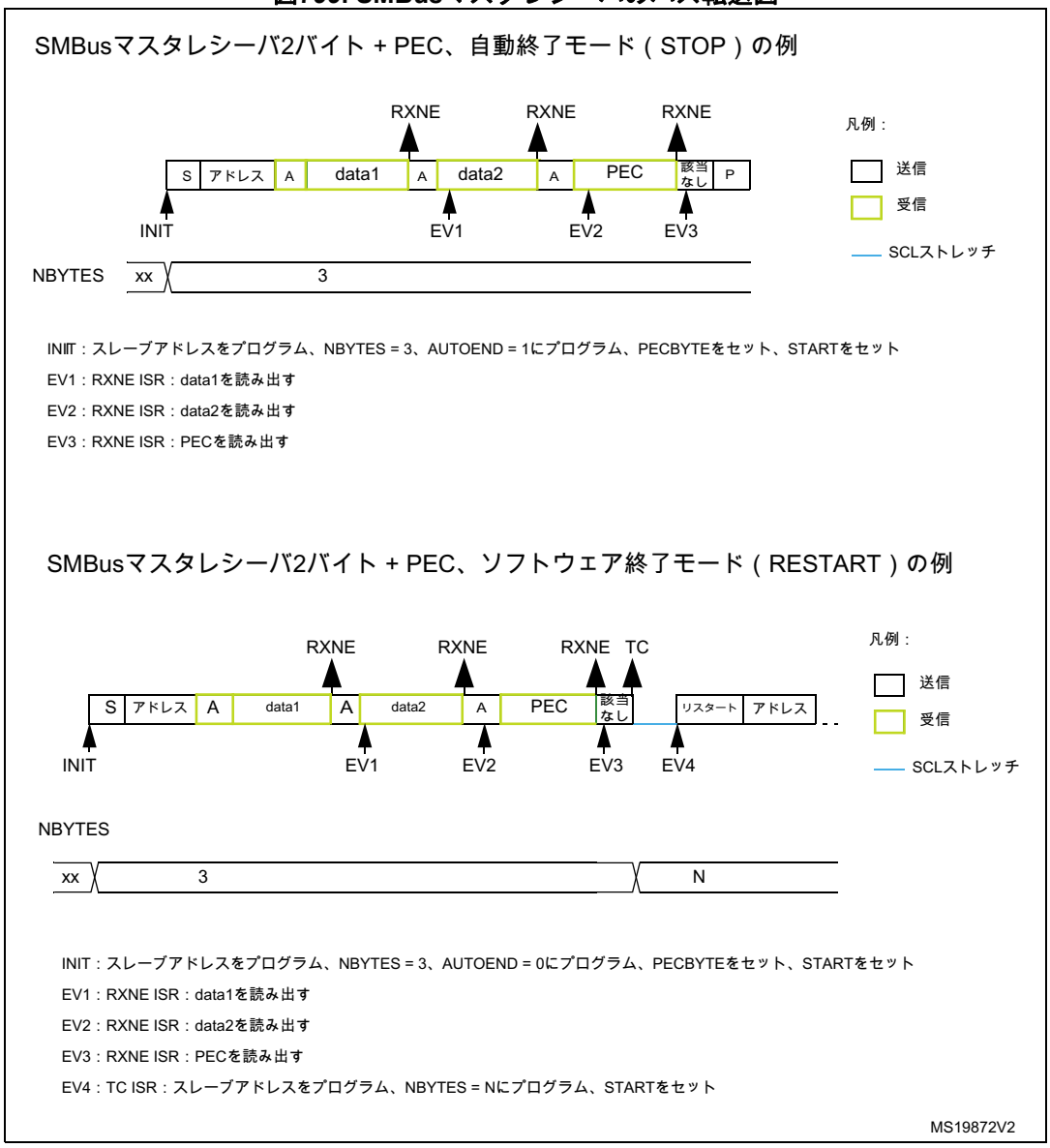
SMBusマスタレシーバ

SMBusマスタが転送終了時にPECを受信してからSTOPを受信したいときには、自動終了モードを選択できます（AUTOEND = 1）。STARTビットをセットする前に、PECBYTEビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES - 1のデータが受信された後、次の受信バイトが I2C_PECRレジスタの内容と自動的に照合されます。PECバイトに対してNACK応答が与えられた後、STOPコンディションが送信されます。

SMBusマスタが転送終了時にPECバイトを受信してからRESTARTを受信したいときには、ソフトウェアモードを選択する必要があります（AUTOEND = 0）。START ビットをセットする前に、PECBYTEビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES - 1のデータが受信された後、次の受信バイトが I2C_PECRレジスタの内容と自動的に照合されます。PECバイト受信後にTCフラグがセットされ、SCLラインローをストレッチします。RESTARTコンディションは、TC割込みサブルーチンでプログラムできます。

注意： **PECBYTEビットは、RELOADビットがセットされているときには効果がありません。**

図799. SMBusマスタレシーバのバス転送図



65.4.15 自律モード

I²Cペリフェラルは、NOSTRETCH = 0のマスタおよびスレーブモードでサポートされている自律モードにより、STOPモードで機能します。自律モードは、NOSTRETCH = 1のときはサポートされません。このモードは、RUN、SLEEP、またはSTOPモードで使用できます。

APBクロックは、I²Cステータスの更新が必要になるたびにペリフェラルによってリクエストされます。ペリフェラルがAPBクロックを受信すると、I²C設定に応じて割込みまたはDMAリクエストが生成されます。

割込みが生成された場合、デバイスはSTOPモードからウェイクアップします。

割込みがない場合、デバイスはSTOPモードのままですが、カーネルとAHB/APBクロックはI2CとRCCで有効化されたすべての自律型ペリフェラルで使用可能です。DMA リクエストが有効な場合、製品がSTOPモードのまま、データはDMAによってSRAMに/から直接転送されます。

スレーブモード

スレーブモードでは、自律モードはI2C_CR1のWUPEN = 1のときにSTOPモードで有効になります。このモードは、NOSTRETCH = 0の場合にのみサポートされます。

カーネルクロックは、スレーブがアドレス指定されている場合、すべての転送のSTART検出中に、STOPコンディションが発生するまで、ペリフェラルによってリクエストされます。スレーブがアドレス指定されていない場合、カーネルクロックリクエストはアドレスフェーズの後で解除されます。

アドレス一致（ADDR）およびSTOP検出（STOPF）イベントの処理を避けるため、スレーブモードでDMA転送のみを使用して機能を最適化するには、I2C_CR1のADDRACLRおよびSTOPFACLRビットをセットすることができます。

注： **I²Cクロックがシステムクロックの場合、またはWUPEN = 0の場合、スレーブモードでは、START受信後も内部オシレータはオンになりません。**

マスタモード

RUN、SLEEP、またはSTOPモードで非同期トリガが検出されると、マスタモードでは転送を自動的に起動できます。I2C_AUTOCRレジスタのTRIGSELによって選択されたトリガは、転送を許可するカーネルクロックリクエストを生成し、STARTコンディションおよびI2C_CR2レジスタ で定義されているとおりにI2C転送を起動します。カーネルクロックは、STOPコンディションが発生するまでリクエストされます。

制御レジスタの書込みのため頻繁にCPUをウェイクアップしないよう、DMAリクエストによって一部の割込みを代替可能です。マスタモードで、I2C_AUTOCRレジスタの対応するTCDMAENまたはTCRDMAENがセットされたときに、転送完了（TC）および転送完了再ロード（TCR）イベントでDMAリクエストを生成できます。その結果、I2C_CR2はDMA転送によって、新しい転送をプログラムするため（TCイベント）、またはバイト数を再ロードするため（TCRイベント）に書き込むことができます。

トリガが有効で、ただし別の転送を開始するためにアプリケーションが制御を取り戻す必要がある場合、新しい転送を開始するためにI2C_CR2レジスタに書き込む前に、以下の手順が必要です。

1. I2C_ISRのBUSY = 0を待ちます。
2. I2C_AUTOCRレジスタのTRIGENビットを無効にします。
3. t_{BUF} タイミングより長い遅延を待ちます（STOPおよびSTARTコンディションの間のバスフリー時間）。
4. I2C_ISRのBUSY = 0を待ちます。

注意： **製品がSTOPモードになると、I²Cは転送に関連付けられている場合のみ、そのカーネルクロックを受信します。その結果、一部の機能（例：タイムアウト、バスアイドル検出）はSTOPモードで信頼できなくなります。**

注意： **デジタルフィルタは、STOP モードからの機能と互換性がありません。DNF ビットが 0 でない場合、WUPENビットをセットしても効果はありません。**

注意： **STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロックストレッチを有効にする必要があります（NOSTRETCH = 0）。**

注意： **STOPモードからのウェイクアップが無効な場合（WUPEN = 0）、STOPモードに入る前に、I2Cペリフェラルを無効にする必要があります（PE = 0）。**

65.4.16 エラー条件

次のエラーは、通信エラーを引き起こす可能性のある条件です。

バスエラー（BERR）

バスエラーは、STARTまたはSTOPコンディションが検出され、それが9 SCLクロックパルスの倍数の後に位置していなかったときに検出されます。STARTまたはSTOPコンディションは、SCLがハイとのときにSDAエッジが発生した場合に検出されます。

バスエラーフラグは、I²Cがマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ（すなわち、スレーブモードのアドレスフェーズでないとき）、セットされます。

スレーブモードでSTARTまたはRESTARTの誤配置が検出された場合、I2Cは、正しいSTARTコンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2C_ISRレジスタのBERRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

アービトレーション喪失（ARLO）

アービトレーション喪失は、SDAラインでハイレベルが送信されたが、SCL立上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDAおよびSCLラインはリリースされ、START制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCLおよびSDAラインがリリースされます。

アービトレーション喪失が検出されると、I2C_ISRレジスタのARLOフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

オーバーラン／アンダーランエラー（OVR）

オーバーランまたはアンダーランエラーは、スレーブモードでNOSTRETCH = 1のとき、および次のときに検出されます：

- 受信時、新しいバイトが受信され、RXDRレジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答としてNACKが自動的に送信されます。
- 送信時：
  - STOPF = 1で、最初のデータバイトが送信されなければならないとき。TXE = 0の場合、I2C_TXDRレジスタの内容が送信され、そうでない場合は0xFFが送信されます。
  - 新しいバイトが送信されるべきときに、I2C_TXDR レジスタがまだ書き込まれていなかった場合、0xFFが送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2C_ISRレジスタのOVRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

パケットエラーチェックエラー（PECERR）

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照してください）。

PECエラーは、受信したPECバイトが I2C_PECRレジスタの内容と一致しなかったときに検出されます。正しくないPECの受信後、NACKが自動的に送信されます。

PECエラーが検出されると、I2C_ISRレジスタのPECERRフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

タイムアウトエラー（TIMEOUT）

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照してください）。

タイムアウトエラーは、次のような条件で発生します：

- TIDLE = 0であり、SCLがTIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これはSMBusタイムアウトの検出に使用されます。
- TIDLE = 1であり、SDAおよびSCLがTIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間がTIMEOUTB[11:0] ビットで定義された時間（SMBus $t_{LOW:MEXT}$ パラメータ）に達した場合。
- スレーブ累積クロックロー延長時間がTIMEOUTB[11:0] ビットで定義された時間（SMBus $t_{LOW:SEXT}$ パラメータ）に達した場合。

マスタモードでタイムアウト違反が検出されると、STOPコンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDAおよびSCLラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2C_ISRレジスタのTIMEOUTフラグがセットされ、I2C_CR1レジスタのERRIEビットがセットされていた場合は割込みが生成されます。

アラート（ALERT）

このセクションは、SMBus機能がサポートされるときにのみ適用されます（セクション65.3を参照してください）。

ALERTフラグは、I2Cインタフェースがホストとして設定され（SMBHEN=1）、アラートピン検出が有効であり（ALERTEN=1）、SMBAピンで立ち下がりエッジが検出されたときにセットされます。I2C_CR1レジスタのERRIEビットがセットされている場合は、割込みが生成されます。

65.4.17 DMAリクエスト

DMAを使用した送信

送信についてDMA（Direct Memory Access）を有効にするには、I2C_CR1レジスタのTXDMAENビットをセットします。TXISビットがセットされるたびに、データは、DMAペリフェラル（を参照）を使用して設定されたSRAM領域から I2C_TXDRレジスタにロードされます。

データのみがDMAで転送されます。

- マスタモード：初期化、スレーブアドレス、方向、バイト数、およびスタートビットはソフトウェアによってプログラムされます（送信されたスレーブアドレスをDMAで転送することはできません）。すべてのデータがDMAを使用して転送されるときには、STARTビットをセットする前に、DMAを初期化する必要があります。転送の終了は、NBYTESカウンタによって管理されます。マスタトランスミッタを参照してください。

- スレーブモードでは：
  - NOSTRETCH = 0では、すべてのデータがDMAを使用して転送されるときには、アドレス一致イベントの前、またはADDR割込みサブルーチンで、ADDRをクリアする前にDMAを初期化する必要があります。
  - NOSTRETCH = 1では、アドレス一致イベントの前にDMAを初期化する必要があります。
- SMBusをサポートする場合：PEC転送はNBYTESカウンタによって管理されます。SMBusスレーブトランスミッタ および SMBusマスタトランスミッタを参照してください。

注： **DMAが送信に使用される場合、TXIEビットが有効である必要はありません。**

DMAを使用した受信

受信についてDMA（Direct Memory Access）を有効にするには、I2C_CR1レジスタのRXDMAENビットをセットします。RXNEビットがセットされているときには、データは、I2C_RXDRレジスタからDMAペリフェラル（ を参照）を使用して設定されたSRAM領域にロードされます。データのみ（PECを含む）がDMAで転送されます。

- マスタモード、初期化、スレーブアドレス、方向、バイト数、およびスタートビットはソフトウェアによってプログラムされます。すべてのデータがDMAを使用して転送されるときには、STARTビットをセットする前に、DMAを初期化する必要があります。転送の終了は、NBYTESカウンタによって管理されます。
- NOSTRETCH = 0のスレーブモードでは、すべてのデータがDMAを使用して転送されるときには、アドレス一致イベントの前、またはADDR割込みサブルーチンで、ADDRをクリアする前にDMAを初期化する必要があります。
- SMBusがサポートされる場合（セクション65.3を参照）：PEC転送はNBYTESカウンタによって管理されます。SMBusスレーブレシーバおよびSMBusマスタレシーバを参照してください。

注： **DMAが受信に使用される場合、RXIEビットが有効である必要はありません。**

DMAを使用したマスタ･イベント制御

マスタモードで、転送は製品がRUN、SLEEP、またはSTOPモードのときにTCおよびTCR DMAリクエストによって自動的に管理できます。

- I2C_AUTOCRレジスタのTCDMAENがセットされている場合、I2C_EVC（I²C制御イベント）のDMAリクエストは、I2C_ISRレジスタのTCがセットされたときに生成されます。DMAは、I2C_CR2レジスタに次のコマンドを書き込むためにプログラムする必要があります。新しいI2C_CR2コマンドでSTOPとSTARTの両方のビットがセットされている場合、STOPコンディションに続いてSTARTコンディションが送信され、その後I2C_CR2で定義されたアドレス、方向、およびバイト数が送信されます。新しいI2C_CR2コマンドでSTARTビットのみがセットされている場合、ReSTARTコンディションが送信され、その後I2C_CR2で定義されたアドレス、方向、およびバイト数が送信されます。
- I2C_AUTOCRレジスタのTCRDMAENがセットされている場合、I2C_EVC（I2C制御イベント）のDMAリクエストは、I2C_ISRレジスタのTCRがセットされたときに生成されます。DMAは、I2C_CR2レジスタに転送する残りのバイト数を書き込むためにプログラムする必要があります。

65.4.18 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールのDBG_I2Cx_ 設定ビットに応じて、SMBusタイムアウトは、通常の動作を続行するか、あるいは停止します。

65.5 I²C低消費電力モード

表663.低消費電力モードがI2Cに与える影響

モード	説明
SLEEP	影響はありません。I²C割込みによって、デバイスはSLEEPモードから復帰します。
STOP(1)	I²Cレジスタの内容は保たれます。自律モードが有効で、I²CがSTOPモードで使用可能な内部オシレータによってクロック供給された場合：マスタモードおよびスレーブモードで転送が機能します。DMAリクエストが機能するとき、割込みにより、デバイスはSTOPモードを終了します。WUPEN = 0の場合：STOPモードに入る前にI²Cを無効にする必要があります。
STANDBY	I²Cペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. 各インスタンスでサポートされるSTOPモードについては、I²C実装表を参照してください。特定のSTOPモードからのウェイクアップがサポートされていない場合、このSTOPモードに入る前に、インスタンスを無効にする必要があります。

65.6 I²C割込み

次の表に、I2C割込みリクエストの一覧を示します。

表664.I2C割込みリクエスト

項目（割込みの略称）		割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了	STOP3、STANDBY、SHUTDOWN モードの終了
I2C	I2C_EV	受信バッファノットエンプティ	RXNE	RXIE	I2C_RXDRレジスタを読み出す	あり	あり(1)	なし
		送信バッファ割込みステータス	TXIS	TXIE	I2C_TXDRレジスタに書き込む			
		STOP検出割込みフラグ	STOPF	STOPIE	STOPCF = 1を書き込む。			
		転送完了再ロード	TCR	TCIE	I2C_CR2のNBYTES[7:0] に0を書き込む			
		転送完了	TC		START = 1またはSTOP = 1を書き込む			
		アドレス一致	ADDR	ADDRIE	ADDRCF=1を書き込む			
		NACK受信	NACKF	NACKIE	NACKCF=1を書き込む			
		アドレス一致	ADDR	ADDRIE	ADDRCF=1を書き込む			
		NACK受信	NACKF	NACKIE	NACKCF=1を書き込む			
	I2C_ER	バスエラー	BERR	ERRIE	BERRCF=1を書き込む	あり	あり (1)	なし
		アービトレーション喪失	ARLO		ARLOCF=1を書き込む			
		オーバーラン／アンダーラン	OVR		OVRCF=1を書き込む			
		PECエラー	PECERR		PECERRCF=1を書き込む			
		タイムアウト／t_{LOW} エラー	TIMEOUT		TIMEOUTCF=1を書き込む			
		SMBusアラート	ALERT		ALERTCF=1を書き込む			

1. 各インスタンスでサポートされるSTOPモードからのウェイクアップの詳細については、I²C実装表を参照してください。

65.7 I²Cレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタは、ワード（32bit）単位でアクセスされます。

65.7.1 I²C制御レジスタ1（I2C_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大2 x i2c_pclk + 6 x i2c_ker_ckです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STOPFACLR	ADDRACLR	Res.	Res.	Res.	Res.	Res.	FMP	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC
rw	rw						rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **STOPFACLR**：STOP検出フラグ（STOPF）自動クリア

0：STOPFフラグは、ハードウェアによってセットされ、ソフトウェアによってSTOPCFビットをセットすることでクリアされます。

1：STOPFフラグは、ハードウェアによってクリア状態に保たれます。このモードはNOSTRETCHスレーブモードで使用でき、次のデータ送信の前にSTOPFフラグがクリアされない場合のオーバーランエラーを回避できます。これにより、ペリフェラルからの割込みなしで、DMAのみによるスレーブデータ管理が可能になります。

ビット30 **ADDRACLR**：アドレス一致フラグ（ADDR）自動クリア

0：ADDRフラグは、ハードウェアによってセットされ、ソフトウェアによってADDRCFビットをセットすることでクリアされます。

1：ADDRフラグは、ハードウェアによってクリア状態に保たれます。このモードはスレーブモードで使用でき、I²Cが1つのスレーブアドレスのみを有効にする場合に、ADDRクロックストレッチを回避できます。これにより、ペリフェラルからの割込みなしで、DMAのみによるスレーブデータ管理が可能になります。

ビット 29:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **FMP**：GPIO高速モードプラス20 mA駆動イネーブル

0：20mA I/O駆動は無効です。

1：20mA I/O駆動は有効です。

ビット23 **PECEN**：PECイネーブル

0：PEC計算は無効です。

1：PEC計算は有効です。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット22 **ALERTEN**：SMBusアラート有効

0：SMBusアラートピン（SMBA）はホストモード（SMBHEN = 1）ではサポートされません。デバイスモード（SMBHEN = 0）では、SMBAピンが解放され、アラート応答アドレスヘッダが無効になります（0001100xの後にはNACKが続きます）。

1:SMBusアラートピンはホストモード(SMBHEN = 1)でサポートされます。デバイスモード(SMBHEN = 0）では、SMBAピンがローに駆動され、アラート応答アドレスヘッダが有効になります（0001100xの後にはACKが続きます）。

注： ALERTEN = 0のときには、SMBAピンを標準GPIOとして使用できます。

SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット21 **SMBDEN**：SMBusデバイスデフォルトアドレス有効

0：デバイスデフォルトアドレスは無効です。アドレス0b1100001xはNACKされます。

1：デバイスデフォルトアドレスは有効です。アドレス0b1100001xはACKされます。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット20 **SMBHEN**：SMBusホストアドレス有効

0：ホストアドレスは無効です。アドレス0b0001000xはNACKされます。

1：ホストアドレスは有効です。アドレス0b0001000xはACKされます。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット19 **GCEN**：同報イネーブル

0：同報は無効です。アドレス0b00000000はNACKされます。

1：同報は有効です。アドレス0b00000000はACKされます。

ビット18 **WUPEN**：STOPモードからのウェイクアップの有効化

0：STOPモードからのウェイクアップは無効です。

1：STOPモードからのウェイクアップは有効です。

注： STOPモードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

注： WUPENは、DNF = 0000のときのみセットできます。

ビット17 **NOSTRETCH**：クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。マスタモードではクリアされたままでなければなりません。

0：クロックストレッチは有効です。

1：クロックストレッチは無効です。

注： このビットは、I2Cが無効（PE = 0）のときのみプログラムできます。

ビット16 **SBC**：スレーブバイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

0：スレーブバイト制御は無効です。

1：スレーブバイト制御は有効です。

ビット15 **RXDMAEN**：DMA受信リクエスト有効

0：受信のためDMAモードを無効にします。

1：受信のためDMAモードを有効にします。

ビット14 **TXDMAEN**：DMA送信リクエスト有効

0：送信のためDMAモードを無効にします。

1：送信のためDMAモードを有効にします。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **ANFOFF**：アナログノイズフィルタOFF

0：アナログノイズフィルタは有効です。

1：アナログノイズフィルタは無効です。

注： このビットは、I2Cが無効（PE = 0）のときのみプログラムできます。

ビット11:8 **DNF[3:0]**：デジタルノイズフィルタ

これらのビットは、SDAおよびSCL入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大DNF[3:0] * t_{I2CCLK} の長さのスパイクを除去します。

0000：デジタルフィルタは無効です。

0001：デジタルフィルタは有効であり、最大1 t_{I2CCLK} の除去能力を持ちます。

.....

1111：デジタルフィルタは有効であり、最大15 t_{I2CCLK} の除去能力を持ちます。

注： アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。
このフィルタは、I2Cが無効（PE = 0）のときのみプログラムできます。

ビット7 **ERRIE**：エラー割込みイネーブル

0：エラー検出割込みは無効です。

1：エラー検出割込みは有効です。

注： 次のようなエラーが発生すると、割込みが生成されます：
アービトレーション喪失（ARLO）
バスエラー検出（BERR）
オーバーラン／アンダーラン（OVR）
タイムアウト検出（TIMEOUT）
PECエラー検出（PECERR）
アラートピンイベント検出（ALERT）

ビット6 **TCIE**：転送完了割込み有効化

0：転送完了割込みは無効です。

1：転送完了割込みは有効です。

注： 次のようなイベントが発生すると、割込みが生成されます：
転送完了（TC）
転送完了再ロード（TCR）

ビット5 **STOPIE**：STOP検出割込みイネーブル

0：STOP検出（STOPF）割込みは無効です。

1：STOP検出（STOPF）割込みは有効です。

ビット4 **NACKIE**：否認応答受信割込みイネーブル

0：否認応答（NACKF）受信割込みは無効です。

1：否認応答（NACKF）受信割込みは有効です。

ビット3 **ADDRIE**：アドレス一致割込みイネーブル（スレーブのみ）

0：アドレス一致（ADDR）割込みは無効です。

1：アドレス一致（ADDR）割込みは有効です。

ビット2 **RXIE**：RX割込みイネーブル

0：受信（RXNE）割込みは無効です。

1：受信（RXNE）割込みは有効です。

ビット1 **TXIE**：TX割込みイネーブル

0：送信（TXIS）割込みは無効です。

1：送信（TXIS）割込みは有効です。

ビット0 **PE**：ペリフェラルイネーブル

0：ペリフェラルは無効です。

1：ペリフェラルは有効です。

注： PE = 0のとき、I2C SCLおよびSDAラインは解放されます。内部ステートマシンおよびステータスビットはリセット値に戻されます。クリアされたときには、PEは少なくとも3 APBクロックサイクルの間、ローに保たれる必要があります。

65.7.2 I²C制御レジスタ2（I2C_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大2 x i2c_pclk + 6 x i2c_ker_ckです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PEC BYTE	AUTO END	RE LOAD	NBYTES[7:0]							
					rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
rs	rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:27 予約済みであり、リセット値に保持する必要があります。

ビット26 **PECBYTE**：パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PECが転送されたとき、またはSTOPコンディションあるいはアドレス一致を受信したとき、また、PE = 0のとき、ハードウェアによってクリアされます。

0：PEC転送なし。

1：PEC送信/受信がリクエストされます。

注： このビットに0を書き込んでも、効果はありません。

このビットは、RELOADがセットされているときには効果がありません。

このビットは、SBC = 0のとき、スレーブモードでは効果がありません。

SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット25 **AUTOEND**：自動終了モード（マスタモード）

このビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェア終了モード：NBYTESデータが転送されるとTCフラグがセットされ、SCLローをストレッチします。

1：自動終了モード：NBYTESデータが転送されると、STOPコンディションが自動的に送信されます。

注： このビットは、スレーブモードまたはRELOADビットがセットされているときには効果がありません。

ビット24 **RELOAD**：NBYTES 再ロードモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送は、NBYTESデータの転送後、完了します（STOPまたはRESTARTが続きます）。

1：転送は、NBYTESデータの転送後に完了しません（NBYTESが再ロードされます）。NBYTESデータが転送されるとTCRフラグがセットされ、SCLローをストレッチします。

ビット23:16 **NBYTES[7:0]**：バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC = 0のスレーブモードでは効果がありません。

注： STARTビットがセットされているときに、これらのビットを変更することはできません。

ビット15 **NACK**：NACK生成（スレーブモード）

このビットはソフトウェアによってセットされ、NACKが送信されたとき、またはSTOPコンディションあるいはアドレス一致を受信したとき、またはPE = 0のとき、ハードウェアによってクリアされます。

0：現在の受信バイト後にACKが送信されます。

1：現在の受信バイト後にNACKが送信されます。

注： このビットに0を書き込んでも、効果はありません。

このビットは、スレーブモードでのみ使用されます：マスタレシーバモードでは、NACKビットの値にかかわらず、STOPまたはRESTARTコンディション前の最後のバイト後にNACKが自動的に生成されます。

スレーブレシーバNOSTRETCHモードでオーバーランが発生すると、NACKビットの値にかかわらず、NACKが自動的に生成されます。

ハードウェアPECチェックが有効なとき（PECBYTE = 1）、PEC確認応答値はNACK値に依存しません。

ビット14 **STOP**：STOP生成（マスタモード）

このビットはソフトウェアによってセットされ、STOPコンディションが検出されたとき、またはPE=0のときにハードウェアによってクリアされます。

マスタモード：

0：STOP生成なし。

1：現在のバイト転送後のSTOP生成。

注： このビットに0を書き込んでも、効果はありません。

ビット13 **START**：START生成

このビットはソフトウェアによってセットされ、STARTとアドレスシーケンスが送信された後、アービトレーション喪失によって、スレーブモードでのアドレス一致によって、タイムアウトエラー検出によって、またはPE = 0のときに、ハードウェアによってクリアされます。

0：START生成なし。

1：RESTART/START生成。

I2Cがすでにマスタモードであり、AUTOEND = 0の場合、このビットをセットすると、RELOAD = 0のとき、NBYTES転送の終了後にREPEATED STARTコンディションが生成されます。

そうでない場合、バスがフリーになると、このビットをセットすることでSTARTコンディションが生成されます。

注： このビットに0を書き込んでも、効果はありません。

バスがBUSYの場合、または I2Cがスレーブモードの場合でも、STARTビットをセットできます。

このビットは、RELOADがセットされているときには効果がありません。

ビット12 **HEAD10R**：読出し方向のみの10bitアドレスヘッダ（マスタレシーバモード）

0：マスタは完全な10bitスレーブアドレス読出しシーケンスを送信します。START + 2バイトの書込み方向の10bitアドレス + RESTART + 読出し方向の10bitアドレスの最初の7bit。

1：マスタは10bitアドレスの最初の7bitのみを送信し、その後に読出し方向を送信します。

注： STARTビットがセットされているときに、このビットを変更することはできません。

ビット11 **ADD10**：10bitアドレッシングモード（マスタモード）

0：マスタは7bitアドレッシングモードで動作します。

1：マスタは10bitアドレッシングモードで動作します。

注： STARTビットがセットされているときに、このビットを変更することはできません。

ビット10 **RD_WRN**：転送方向（マスタモード）

0：マスタは書込み転送をリクエストします。

1：マスタは読出し転送をリクエストします。

注： STARTビットがセットされているときに、このビットを変更することはできません。

ビット9:0 **SADD[9:0]**：スレーブアドレス（マスタモード）

7bitアドレッシングモード（ADD10=0）：

SADD[7:1] には、送信される7bitのスレーブアドレスを書き込む必要があります。SADD[9]、SADD[8]、およびSADD[0] ビットは無視されます。

10bitアドレッシングモード（ADD10=1）：

SADD[9:0] には、送信される10bitのスレーブアドレスを書き込む必要があります。

注： STARTビットがセットされているときに、これらのビットを変更することはできません。

65.7.3 I²C Own Address 1レジスタ（I2C_OAR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA1EN	Res.	Res.	Res.	Res.	OA1 MODE	OA1 [9:0]									
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **OA1EN**：Own Address 1イネーブル

0：Own Address 1は無効です。受信されたスレーブアドレスOA1はNACKされます。

1：Own Address 1は有効です。受信されたスレーブアドレスOA1はACKされます。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **OA1MODE**：Own Address 1 10bitモード

0：Own Address 1は7bitアドレスです。

1：Own Address 1は10bitアドレスです。

注： このビットは、OA1EN = 0のときのみ書き込むことができます。

ビット9:0 **OA1[9:0]**：インタフェース専用スレーブアドレス

7bitアドレスモード：OA1[7:1] には7bitの専用スレーブアドレスが含まれます。OA1[9]、OA1[8]、およびOA1[0] ビットは無視されます。

10bitアドレスモード：OA1[9:0] には10bitの専用スレーブアドレスが含まれます。

注： これらのビットは、OA1EN = 0のときのみ書き込むことができます。

65.7.4 I2C Own Address 2レジスタ（I2C_OAR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大2 x i2c_pclk + 6 x i2c_ker_ckです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2 [7:1]							Res.
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **OA2EN**：Own Address 2イネーブル

0：Own Address 2は無効です。受信されたスレーブアドレスOA2はNACKされます。

1：Own Address 2は有効です。受信されたスレーブアドレスOA2はACKされます。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **OA2MSK[2:0]**：Own Address 2マスク

000：マスクなし

001：OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。

010：OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。

011：OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。

100：OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。

101：OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。

110：OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。

111：OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7bit受信アドレスが確認応答されます。

注： これらのビットは、OA2EN = 0のときのみ書き込むことができます。

OA2MSKが0でなくなると、予約済み I2Cアドレス（0b0000xxxおよび0b1111xxx）は、比較が一致した場合でも確認応答されません。

ビット7:1 **OA2[7:1]**：インタフェースアドレス

7bitアドレスモード：7bitアドレス

注： これらのビットは、OA2EN = 0のときのみ書き込むことができます。

ビット0 予約済みであり、リセット値に保持する必要があります。

65.7.5 I²Cタイミングレジスタ（I2C_TIMINGR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]			
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLH[7:0]								SCLL[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 **PRESC[3:0]**：タイミングプリスケーラ

このフィールドは、データのセットアップおよびホールドカウンタ（I2Cのタイミングを参照）とSCLハイおよびローレベルカウンタ（I2Cマスタ初期化 を参照）に使用されるクロック周期t_{PRESC} を生成するために、i2c_ker_ck をプリスケールするのに使用されます。

t_{PRESC} = (PRESC + 1) x t_{I2CCLK}

ビット27:24 予約済みであり、リセット値に保持する必要があります。

ビット23:20 **SCLDEL[3:0]**：データセットアップ時間

このフィールドは、SDAエッジとSCL立上がりエッジの間に遅延t_{SCLDEL} を生成するために使用されます。NOSTRETCH = 0でのマスタモードおよびスレーブモードでは、SCLラインは、t_{SCLDEL} の間ローにストレッチされます。

t_{SCLDEL} = (SCLDEL + 1) x t_{PRESC}

注： t_{SCLDEL} は、$t_{SU:DAT}$ タイミングを生成するために使用されます。

ビット19:16 **SDADEL[3:0]**：データホールド時間

このフィールドは、SCL立下がりエッジとSDAエッジの間に遅延t_{SDADEL} を生成するために使用されます。NOSTRETCH = 0でのマスタモードおよびスレーブモードでは、SCLラインは、t_{SDADEL} の間ローにストレッチされます。

t_{SDADEL}= SDADEL x t_{PRESC}

注： SDADELは、$t_{HD:DAT}$ タイミングを生成するために使用されます。

ビット15:8 **SCLH[7:0]**：SCLハイ周期（マスタモード）

このフィールドは、マスタモードでSCLハイ周期を生成するために使用されます。

t_{SCLH} = (SCLH + 1) x t_{PRESC}

注： SCLHは、$t_{SU:STO}$ および$t_{HD:STA}$タイミングを生成するためにも使用されます。

ビット7:0 **SCLL[7:0]**：SCLロー周期（マスタモード）

このフィールドは、マスタモードでSCLロー周期を生成するために使用されます。

t_{SCLL} = (SCLL + 1) x t_{PRESC}

注： SCLLは、t_{BUF} および$t_{SU:STA}$ タイミングを生成するためにも使用されます。

注： このレジスタは、I²Cが無効（PE=0）のときに設定する必要があります。

注： STM32CubeMXツールは、I2C設定ウィンドウのI2C_TIMINGRコンテンツを計算し、提供します。

65.7.6 I²Cタイムアウトレジスタ（I2C_TIMEOUTR）

アドレスオフセット：0x14

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大2 x i2c_pclk + 6 x i2c_ker_ckです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **TEXTEN**：拡張クロックタイムアウト有効

0：拡張クロックタイムアウト検出は無効です。

1：拡張クロックタイムアウト検出は有効です。$t_{LOW:EXT}$ を超える累積SCLストレッチが I²Cインタフェースによって行われると、タイムアウトエラーが検出されます（TIMEOUT = 1）。

ビット30:28 予約済みであり、リセット値に保持する必要があります。

ビット27:16 **TIMEOUTB[11:0]**：バスタイムアウトB

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます：

マスタモードでは、マスタ累積クロックロー拡張時間（$t_{LOW:MEXT}$）が検出されます。

スレーブモードでは、スレーブ累積クロックロー拡張時間（$t_{LOW:SEXT}$）が検出されます。

$t_{LOW:EXT}$ = (TIMEOUTB + TIDLE = 01) x 2048 x t_{I2CCLK}

注： これらのビットは、TEXTEN = 0のときのみ書き込むことができます。

ビット15 **TIMOUTEN**：クロックタイムアウト有効

0：SCLタイムアウト検出は無効です。

1：SCLタイムアウト検出は有効です。SCLが$t_{TIMEOUT}$（TIDLE = 0）を超えてローであるか、t_{IDLE}（TIDLE = 1）を超えてハイであった場合、タイムアウトエラーが検出されます（TIMEOUT = 1）。

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **TIDLE**：アイドルクロックタイムアウト検出

0：TIMEOUTAは、SCLロータイムアウトの検出に使用されます。

1：TIMEOUTAは、SCLとSDAの両方のハイタイムアウト（バスアイドル条件）の検出に使用されます。

注： このビットは、TIMOUTEN = 0のときのみ書き込むことができます。

ビット11:0 **TIMEOUTA[11:0]**：バスタイムアウトA

このフィールドは、以下を設定するために使用されます：

TIDLE = 0のときのSCLロータイムアウト条件$t_{TIMEOUT}$

$t_{TIMEOUT}$ = (TIMEOUTA + 1) x 2048 x t_{I2CCLK}

TIDLE = 1のときのバスアイドル条件（SCLとSDAの両方のハイ）

t_{IDLE} = (TIMEOUTA + 1) x 4 x t_{I2CCLK}

注： これらのビットは、TIMOUTEN = 0のときのみ書き込むことができます。

注： SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に0x00000000に設定されます。セクション65.3を参照してください。

65.7.7 I²C割込みおよびステータスレジスタ（I2C_ISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR
								r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.	ALERT	TIME OUT	PEC ERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
r		r	r	r	r	r	r	r	r	r	r	r	r	rs	rs

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:17 **ADDCODE[6:0]**：アドレス一致コード（スレーブモード）

これらのビットは、アドレス一致イベントが発生したときに（ADDR = 1）、受信したアドレスで更新されます。

10bitアドレスの場合、ADDCODEは10bitのヘッダとその後のアドレスの2つのMSBを示します。

ビット16 **DIR**：転送方向（スレーブモード）

このフラグは、アドレス一致イベントが発生したときに（ADDR = 1）、更新されます。

0：書込み転送、スレーブはレシーバモードになります。

1：読出し転送、スレーブはトランスミッタモードになります。

ビット15 **BUSY**：バスビジー

このフラグは、バス上で通信が進行中であることを表示します。STARTコンディションが検出されたときにハードウェアによってセットされます。STOPコンディションが検出されたとき、またはPE = 0のときにハードウェアによってクリアされます。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 **ALERT**：SMBusアラート

このフラグは、SMBHEN = 1（SMBusホスト設定）、ALERTEN = 1、およびSMBALERTイベント（立下がりエッジ）がSMBAピンで検出されたときに、ハードウェアによってセットされます。ALERTCFビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE = 0のとき、ハードウェアによってクリアされます。

SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット12 **TIMEOUT**：タイムアウトまたはt_{LOW}検出フラグ

このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCFビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE = 0のとき、ハードウェアによってクリアされます。

SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット11 **PECERR**：受信時のPECエラー

このフラグは、受信したPECがPECレジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくないPECの受信後、NACKが自動的に送信されます。PECCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット10 **OVR**：オーバーラン/アンダーラン（スレーブモード）

このフラグは、NOSTRETCH = 1のスレーブモードで、オーバーラン／アンダーランエラーが発生したときに、ハードウェアによってセットされます。OVRCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット9 **ARLO**：アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット8 **BERR**：バスエラー

このフラグは、ペリフェラルが転送に関与しているので、STARTまたはSTOPコンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット7 **TCR**：転送完了再ロード

このフラグは、RELOAD = 1で、NBYTESデータが転送されたときに、ハードウェアによってセットされます。NBYTESにゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

このフラグは、マスタモード、またはSBCビットがセットされているときのスレーブモードでのみ使用されます。

ビット6 **TC**：転送完了（マスタモード）

このフラグは、RELOAD = 0、AUTOEND = 0、およびNBYTESデータが転送されたときに、ハードウェアによってセットされます。STARTビットまたはSTOPビットがセットされたときに、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット5 **STOPF**：STOP検出フラグ

このフラグは、バス上でSTOPコンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます：

– マスタとして。ただし、STOPコンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット4 **NACKF**：否認応答受信フラグ

このフラグは、バイト送信後にNACKを受信したときに、ハードウェアによってセットされます。NACKCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット3 **ADDR**：アドレス一致（スレーブモード）

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの1つに一致したときに、ハードウェアによってセットされます。ADDRCFビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット2 **RXNE**：受信データレジスタノットエンプティ（レシーバ）

このビットは、受信データが I2C_RXDRレジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2C_RXDRが読み出されたときにクリアされます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット1 **TXIS**：送信割込みステータス（トランスミッタ）

このビットは、I2C_TXDRレジスタが空であり、送信データを I2C_TXDRレジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2C_TXDRレジスタに書き込まれたときにクリアされます。

NOSTRETCH = 1の場合にのみ、このビットにソフトウェアで1を書き込んで、TXISイベント（TXIE = 1の場合は割込み、TXDMAEN = 1の場合はDMAリクエスト）を生成できます。

注： このビットは、PE = 0のとき、ハードウェアによってクリアされます。

ビット0 **TXE**：送信データレジスタエンプティ（トランスミッタ）

このビットは、I2C_TXDRレジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2C_TXDRレジスタに書き込まれたときにクリアされます。

このビットは、ソフトウェアによって1を書き込んで、送信データレジスタ I2C_TXDRをフラッシュできます。

注： このビットは、PE = 0のとき、ハードウェアによってセットされます。

65.7.8 I2C割込みクリアレジスタ（I2C_ICR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
		w	w	w	w	w	w			w	w	w			

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **ALERTCF**：アラートフラグクリア

このビットに1を書き込むと、I2C_ISRレジスタのALERTフラグがクリアされます。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット12 **TIMOUTCF**：タイムアウト検出フラグクリア

このビットに1を書き込むと、I2C_ISRレジスタのTIMEOUTフラグがクリアされます。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット11 **PECCF**：PECエラーフラグクリア

このビットに1を書き込むと、I2C_ISRレジスタのPECERRフラグがクリアされます。

注： SMBus機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション65.3を参照してください。

ビット10 **OVRCF**：オーバーラン/アンダーランフラグクリア

このビットに1を書き込むと、I2C_ISRレジスタのOVRフラグがクリアされます。

ビット9 **ARLOCF**：アービトレーション喪失フラグクリア
このビットに1を書き込むと、I2C_ISRレジスタのARLOフラグがクリアされます。

ビット8 **BERRCF**：バスエラーフラグクリア
このビットに1を書き込むと、I2C_ISRレジスタのBERRFフラグがクリアされます。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **STOPCF**：STOP検出フラグクリア
このビットに1を書き込むと、I2C_ISRレジスタのSTOPFフラグがクリアされます。

ビット4 **NACKCF**：否認応答フラグクリア
このビットに1を書き込むと、I2C_ISRレジスタのNACKFフラグがクリアされます。

ビット3 **ADDRCF**：アドレス一致フラグクリア
このビットに1を書き込むと、I2C_ISRレジスタのADDRフラグがクリアされます。このビットに1を書き込むと、I2C_CR2レジスタのSTARTビットもクリアされます。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

65.7.9 I²C PECレジスタ（I2C_PECR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PEC[7:0]**：パケットエラーチェックレジスタ
PECEN=1のとき、このフィールドは内部PECを含みます。
PECは、PE = 0のとき、ハードウェアによってクリアされます。

注： **SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に0x00000000に設定されます。セクション65.3を参照してください。**

65.7.10 I²C受信データレジスタ（I2C_RXDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **RXDATA[7:0]：**8bitのレシーバのデータ
I²Cバスから受信したデータバイト。

65.7.11 I²C送信データレジスタ（I2C_TXDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **TXDATA[7:0]：**8bitの送信のデータ
I²Cバスに送信されるデータバイト。

注： これらのビットは、TXE = 1のときのみ書き込むことができます。

65.7.12 I²C自律モード制御レジスタ（I2C_AUTOCR）

アドレスオフセット：0x2C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGEN	TRIGPOL	TRIGSEL[3:0]			
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCRDMAEN	TCDMAEN	Res.	Res.	Res.	Res.	Res.	Res.
								rw	rw						

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **TRIGEN**：トリガ有効

0：トリガは無効です。

1：トリガは有効です。

トリガが検出されると、STARTコンディションが送信され、I2C_CR2で定義されたように転送が開始されます。

ビット20 **TRIGPOL**：トリガ極性

0：立ち上がりエッジでアクティブなトリガ

1：立ち下がりエッジでアクティブなトリガ

注： このビットは、PE = 0のときのみ書き込むことができます。

ビット19:16 **TRIGSEL[3:0]**：トリガ選択（セクション 65.4.2：I2Cピンおよび内部信号 I²C相互接続表を参照）。

0000：i2c_trg0を選択します。

0001：i2c_trg1を選択します。

.....

1111：i2c_trg15を選択します。

注： このビットは、PE = 0のときのみ書き込むことができます。

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TCRDMAEN**：転送完了再ロードイベントでのDMAリクエストイネーブル

0：転送完了再ロードイベントでDMAリクエストが生成されません。

1：転送完了再ロードイベントでDMAリクエストが生成されます。

ビット6 **TCDMAEN**：転送完了イベントでのDMAリクエストイネーブル

0：転送完了イベントでDMAリクエストが生成されません。

1：転送完了イベントでDMAリクエストが生成されます。

ビット5:0 予約済みであり、リセット値に保持する必要があります。

65.7.13 I²Cレジスタマップ

次の表に、I²Cのレジスタマップとリセット値を示します。

表665.I²Cレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	I2C_CR1	STOPFACLR	ADDRACLR	Res.	Res.	Res.	Res.	Res.	FMP	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC	RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
	リセット値	0	0						0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x4	I2C_CR2	Res.	Res.	Res.	Res.	Res.	PECBYTE	AUTOEND	RELOAD	NBYTES[7:0]								NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	I2C_OAR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA1EN	Res.	Res.	Res.	Res.	OA1MODE	OA1 [9:0]									
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0
0xC	I2C_OAR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2 [7:1]							Res.
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	
0x10	I2C_ TIMINGR	PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL [3:0]				SDADEL [3:0]				SCLH[7:0].								SCLL[7:0]							
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	I2C_ TIMEOUTR	TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]												TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	I2C_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR	BUSY	Res.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
	リセット値									0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	I2C_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
	リセット値																			0	0	0	0	0	0			0	0	0			
0x20	I2C_PECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x24	I2C_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x28	I2C_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表665.I²Cレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	I2C_AUTOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGEN	TRIGPOL	TRIGSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCRDMAEN	TCDMAEN	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0									0	0						

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

66 Universal synchronous/asynchronous receiver transmitter (USART/UART)

このセクションでは、ユニバーサル同期非同期レシーバトランスミッタ（USART/UART）について説明します。

66.1 概要

USARTを使用すると、業界標準のNRZ非同期シリアルデータフォーマットを必要とする外部機器と柔軟性の高い全二重データ交換を行うことができます。フラクショナルボーレートジェネレータによって、非常に広範囲のボーレートが達成できます。

USARTは同期式単方向通信と半二重単線通信の両方のほか、LIN（Local Interconnection Network）、スマートカードプロトコル、IrDA（infrared data association）SIR ENDEC仕様、およびモデム動作（CTS/RTS）もサポートします。マルチプロセッサ通信もサポートされています。

マルチバッファ設定でDMA（直接メモリアクセス）を使用することによって、ハイスピードデータ通信が可能です。

66.2 USARTの主な機能

- 全二重非同期通信
- NRZ標準フォーマット（マーク／スペース）
- 16倍または8倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の最適な調整を達成
- ボーレートジェネレータシステム
- データ送信および受信用の2つの内部FIFO
  各FIFOはソフトウェアで有効／無効にすることができ、ステータスフラグを装備
- 共通にプログラム可能な送信および受信ボーレート
- PCLKから独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- 自動ボーレート検出
- プログラム可能なデータワード長（7または8または9bit）
- データ順序をプログラム可能（MSBファースト／LSBファーストのシフト）
- 設定可能なストップビット（1または2個のストップビット）
- 同期通信のための同期マスタ／スレーブモードとクロック出力／入力
- SPIスレーブ送信アンダーランエラーフラグ
- 単線半二重通信
- DMAを使用した連続通信
- 送受信バイトは集中型DMAを使用して予約済みSRAMにバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能なTx/Rxピン設定
- モデムとRS-485トランシーバのハードウェアフロー制御
- 通信制御／エラー検出フラグ

- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- フラグ付き割込みソース
- マルチプロセッサ通信:アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ
- STOPモードからのウェイクアップ
- STOPモードでの自律機能

66.3 USARTの拡張機能

- LINマスタの同期ブレーク送信機能とLINスレーブのブレーク検出機能
  - USARTがLIN用にハードウェア設定されている場合、13bitのブレーク生成と10/11bitのブレーク検出
- 通常モードで3/16bit期間をサポートするIrDA SIRエンコーダデコーダ
- スマートカードモード
  - ISO/IEC 7816-3標準で定義されているスマートカードのT=0およびT=1非同期プロトコルをサポート
  - スマートカード動作用に0.5個および1.5個のストップビット
- Modbus通信のサポート
  - タイムアウト機能
  - CR/LFキャラクタ認識

66.4 USARTの実装

以下の表では、USARTの実装について説明します。比較のため、表にはLPUARTも含めています。

表666.STM32U5シリーズでのUSART/UART/LPUARTの実装

インスタンス	STM32U535/545	STM32U575/585	STM32U59x/5Ax/5Fx/5Gx
USART1	フル	フル	フル
USART2	-	フル	フル
USART3	フル	フル	フル
USART6	-	-	フル
UART4	基本	基本	基本
UART5	基本	基本	基本
LPUART1	低消費電力	低消費電力	低消費電力

表667.USART/LPUARTの機能

モードまたは機能(1)	フル機能	基本機能	低電力機能
モデムのハードウェアフロー制御	X	X	X
DMAを使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X

表667.USART/LPUARTの機能 （続き）

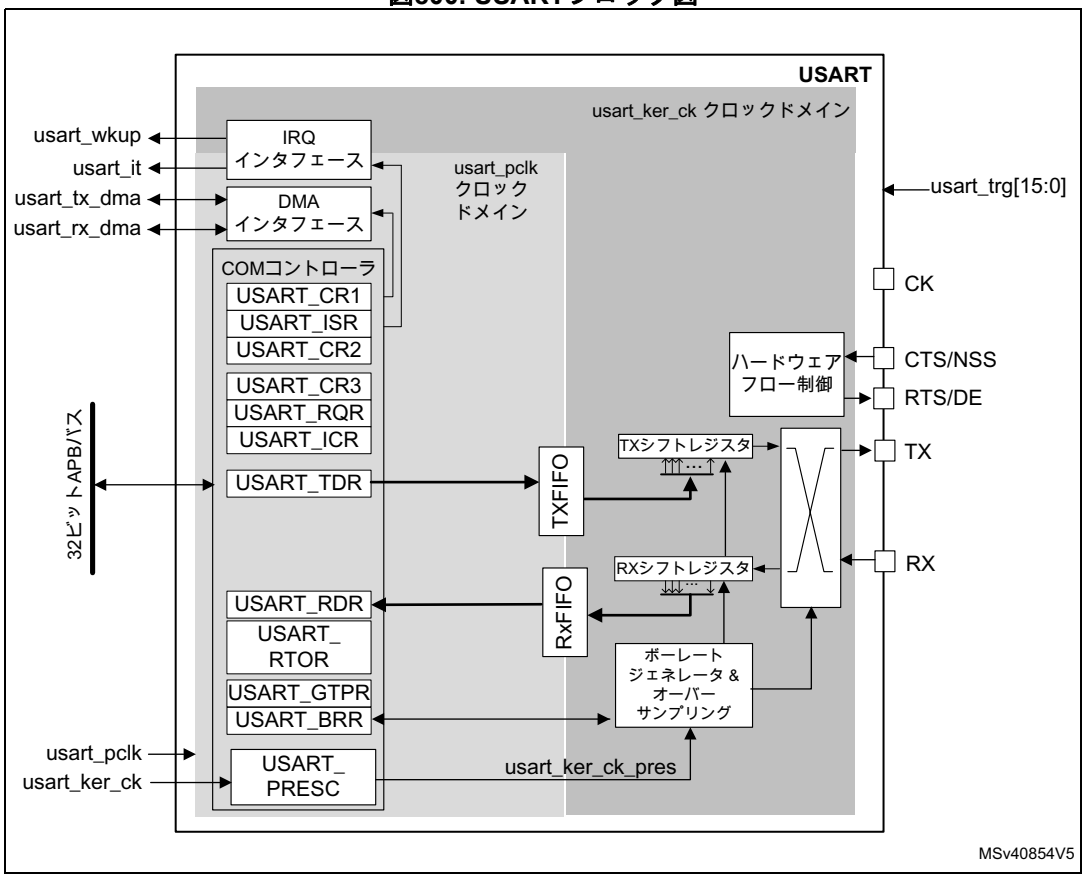
モードまたは機能[1]	フル機能	基本機能	低電力機能
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDECブロック	X	X	-
LINモード	X	X	-
デュアルクロックドメイン	X	X	X
レシーバタイムアウト割込み	X	X	-
Modbus通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USARTデータ長	7、8、および9bit		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFOサイズ	8		
低消費電力モードからのウェイクアップ	X[2]	X [2]	X[3]
自律モード	X	X	X

1. X : サポートされています。
2. STOP 0およびSTOP 1モードからサポートされるウェイクアップ。
3. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

66.5 USARTの機能説明

66.5.1 USARTブロック図

図800. USARTブロック図



66.5.2 USARTピンおよび内部信号

USARTの入出力ピンの説明

- USART双方向通信

  USART双方向通信では、少なくとも受信データ入力（RX）と送信データ出力（TX）の2個のピンが必要です。

  – **RX**（受信データ入力）

  RXはシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用されています。それによって、有効な入力データとノイズを区別しています。

  – **TX**（送信データ出力）

  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、データを送信する必要がないとき、TXピンはハイになります。単線およびスマートカードモードでは、このI/Oはデータの送受信に使用されます。

- RS232ハードウェアフロー制御モード

  RS232ハードウェアフロー制御モードでは、以下のピンが必要です。

  – **CTS**（Clear To Send）

    ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。

  – **RTS**（Request To Send）

    ローレベルのとき、この信号はUSARTがデータを受信する準備ができたことを示します。

- RS485ハードウェア制御モード

  RS485ハードウェア制御モードでは、**DE**（Driver Enable）ピンが必要です。この信号は、外部トランシーバの送信モードを有効にします。

- 同期マスタ／スレーブモードおよびスマートカードモード

  同期マスタ／スレーブモードおよびスマートカードモードでは、以下のピンが必要です。

  – CK

    このピンは同期マスタモードおよびスマートカードモードで、クロック出力として機能します。

    同期スレーブモードでは、クロック入力として機能します。

    同期マスタモードでは、このピンは、SPIマスタモードに対応する同期送信用のトランスミッタデータクロックを出力します（スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します）。これと並行して、RXピンでデータを同期受信できます。このメカニズムを使用して、シフトレジスタを持つペリフェラル（LCDドライバなど）を制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。

    スマートカードモードでは、CK出力はスマートカードにクロックを供給します。

  – NSS

    このピンは、同期スレーブモードでスレーブ選択入力として機能します。

USART入出力ピンおよび内部信号のリストについては、表668および表669を参照してください。

表668.USARTの入出力ピン

ピン名	信号タイプ	説明
USART_RX	入力	シリアルデータ受信入力。
USART_TX	出力	送信データ出力。
USART_CTS	入力	送信可否
USART_RTS	出力	送信要求
USART_DE(1)	出力	ドライバイネーブル
USART_CK	出力	同期マスタモードおよびスマートカードモードでのクロック出力。
USART_NSS(2)	入力	同期スレーブモードでのスレーブ選択入力。

1. USART_DEとUSART_RTSは同じピンを共有します。
2. USART_NSSとUSART_CTSは同じピンを共有します。

USARTの入出力信号の説明

表669.USART内部入力／出力信号

ピン名	信号タイプ	説明
usart_pclk	入力	APBクロック
usart_ker_ck	入力	USARTカーネルクロック
usart_wkup	出力	USARTはウェイクアップ割込みに対応
usart_it	出力	USARTグローバル割込み
usart_tx_dma	入力／出力	USART送信DMAリクエスト
usart_rx_dma	入力／出力	USART受信DMAリクエスト
usart_trg[15:0]	入力	USARTトリガ

USARTの相互作用の説明

表670.USARTの相互接続（USART1/2/3/6およびUART4/5）

信号名	転送元
usart_trg0	gpdma1_ch0_tc
usart_trg1	gpdma1_ch1_tc
usart_trg2	gpdma1_ch2_tc
usart_trg3	gpdma1_ch3_tc
usart_trg4	exti6
usart_trg5	exti9
usart_trg6	lptim1_ch1
usart_trg7	lptim2_ch1
usart_trg8	comp1_out
usart_trg9	comp2_out
usart_trg10	rtc_alra_trg
usart_trg11	rtc_wut_trg
usart_trg12	-
usart_trg13	-
usart_trg14	-
usart_trg15	-

66.5.3 USARTクロック

図800に示した簡易ブロック図には、以下の完全に独立した2つのクロック・ドメインが示されています。

- **usart_pclk** クロックドメイン

  usart_pclkクロック信号はペリフェラルバスインタフェースにクロック供給します。USARTレジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **usart_ker_ck** カーネルクロックドメイン

  usart_ker_ck はUSARTクロックソースです。これは **usart_pclk** から独立しており、RCCによって供給されます。したがって、**usart_ker_ck** クロックが停止しているときでも、USARTレジスタは読み書きできます。

  デュアルクロックドメイン機能がサポートされていないときは、**usart_ker_ck** クロックは **usart_pclk** クロックと同じになります。

usart_pclk と**usart_ker_ck** の間には何の制約もありません。**usart_ker_ck**は**usart_pclk** より速くすることも遅くすることもできます。唯一の限界は、十分に速く通信を管理できるソフトウェアの能力です。

USARTがSPIスレーブモードで動作するときは、USARTは、外部マスタSPIデバイスによって提供される外部CK信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。**usart_ker_ck** クロックはCK入力のクロックより少なくとも3倍は速くなければなりません。

66.5.4 USARTキャラクタの説明

ワード長は、USART_CR1レジスタのMビット（M0：ビット12およびM1：ビット28）をプログラムすることによって、7、8、または9bitに設定できます（図801を参照）。

- 7bitのキャラクタ長：M[1:0] = 10
- 8bitのキャラクタ長：M[1:0] = 00
- 9bitのキャラクタ長：M[1:0] = 01

注： **7bitデータ長モードでは、スマートカードモード、LINマスタモード、および自動ボーレート（0x7Fおよび0x55フレーム検出）はサポートされません。**

デフォルトでは、信号（TXまたはRX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

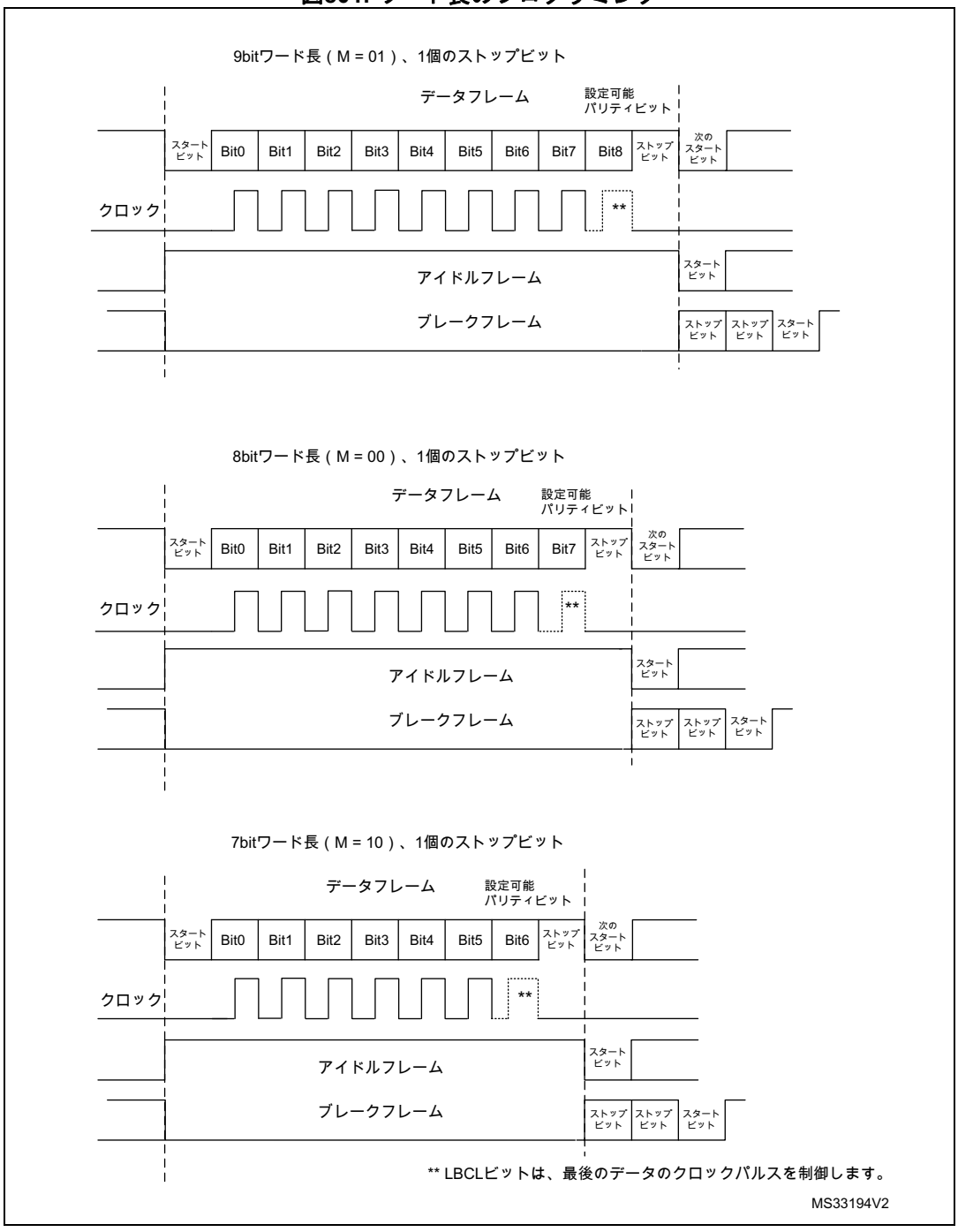
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは2個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図801. ワード長のプログラミング



66.5.5 USARTのFIFOと閾値

USARTはFIFOモードで動作できます。

USARTは送信FIFO（TXFIFO）と受信FIFO（RXFIFO）を備えています。FIFOモードは、USART_CR1レジスタのFIFOEN（ビット29）をセットすることによって有効になります。このモードは、UART、SPI、およびスマートカードモードでのみ、サポートされます。

最大のデータワード長が9bitなので、TXFIFOは9bit幅です。しかし、RXFIFOのデフォルト幅は12bitです。この理由は、レシーバはFIFOにデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともにRXFIFOに格納されます。ただし、RDRを読み出すときは、データのみが読み出されます。**

ステータスフラグは、USART_ISRレジスタで入手可能です。

TxおよびRX割込みがトリガされるTXFIFOおよびRXFIFOのレベルを設定することができます。これらの閾値は、USART_CR3制御レジスタのRXFTCFGおよびTXFTCFGビットフィールドによってプログラムされます。

このとき、

- RXFIFOに受信したデータの数がRXFTCFGビットフィールドでプログラムされた閾値に達すると、Rx割込みが生成されます。

  この場合、USART_ISRレジスタのRXFTフラグがセットされます。これはRXFTCFG分のデータ、すなわちUSART_RDRに1データ、RXFIFOに（RXFTCFG - 1）データを受信したことを意味します。例として、RXFTCFGが101にプログラムされている場合、FIFOサイズに相当する量のデータが受信された（FIFOサイズ -1のデータがRXFIFOに入り、1データがUSART_RDRに入った）時に、RXFTフラグがセットされます。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO 内の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、Tx割込みが生成されます。

66.5.6 USARTトランスミッタ

トランスミッタは、Mビットのステータスに応じて、7または8または9bitのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータはTXピンで出力され、対応するクロックパルスはCKピンで出力されます。

キャラクタ送信

USART送信時、データはLSBファースト（デフォルト設定）でTXピンにシフトアウトされます。このモードでは、USART_TDRレジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます。

FIFOモードが有効なとき、送信データレジスタ（USART_TDR）に書き込まれたデータはTXFIFOのキューに入ります。

各キャラクタの前には、スタートビット（1bit周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は0.5、1、1.5、または2に設定できます。

注： **送信データをUSART_TDRに書き込む前に、TEビットをセットする必要があります。**

データの送信中にTEビットをリセットしないでください。送信中にTEビットをリセットすると、ボーレートカウンタが停止されるため、TXピンのデータが破壊されます。送信中の現在のデータは失われます。

TEビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

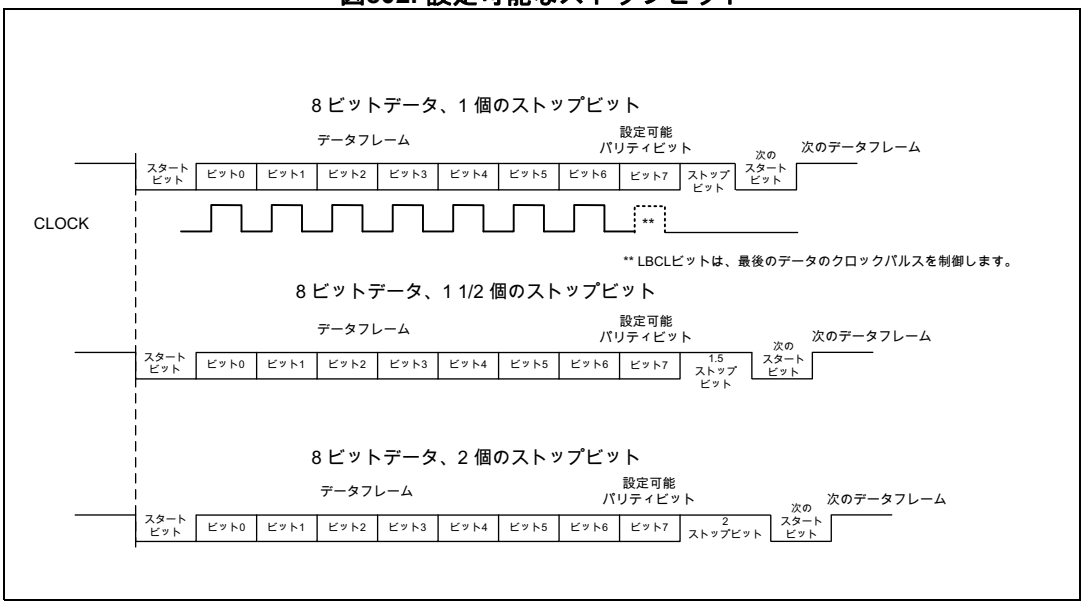
各キャラクタとともに送信されるストップビットの数は、USART_CR2のビット13、12でプログラミングできます。

- **1つのストップビット**：これはストップビット数のデフォルト値です。
- **2つのストップビット**：これは通常のUSARTモード、単線モード、およびモデムモードでサポートされます。
- **1.5 個のストップビット**：スマートカードモードで使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10個のロービット（M[1:0] = “00”のとき）または11個のロービット（M[1:0] = “01”のとき）または9個のロービット（M[1:0] = “10”のとき）の後に2個のストップビットが続きます（図802を参照）。長いブレーク（9/10/11個のロービットを超える長さのブレーク）を送信することはできません。

図802. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. USART_CR1のMビットをプログラムして、ワード長を定義します。
2. USART_BRRレジスタを使用して、希望するボーレートを選択します。
3. USART_CR2レジスタでストップビットの数をプログラミングします。
4. USART_CR1レジスタのUEビットに1を書き込んで、USARTを有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAT）を選択します。セクション 66.5.20：USARTおよびDMAを使用した連続通信の説明に基づいて、DMAレジスタを設定します。

6. USART_CR1のTEビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータをUSART_TDRレジスタに書き込みます。シングルバッファの場合、送信される各データにこれを繰り返します。
   – FIFOモードが無効になっている場合は、USART_TDRに1つのデータを書き込むと、TXEフラグがクリアされます。
   – FIFOモードが有効になっている場合は、USART_TDRに1つのデータを書き込むと、1つのデータがTXFIFOに追加されます。TXFNFフラグがセットされていると、USART_TDRへの書込み動作が行われます。このフラグはTXFIFOがフルになるまでセットされたままになります。
8. USART_TDRレジスタに最後のデータを書き込んだら、TC=1になるまで待ちます。
   – FIFOモードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   – FIFOモードが有効になっている場合、これは、TXFIFOとシフトレジスタの両方が空になっていることを示します。

   このチェックは、USARTが無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1バイト通信

- FIFOモードが無効の場合

  送信データレジスタに書き込むと、必ずTXEビットがクリアされます。TXEフラグがハードウェアによってセットされます。同フラグは以下を示します。
  – データはUSART_TDRレジスタからシフトレジスタへ移動され、データ送信が開始しています。
  – USART_TDRレジスタは空です。
  – 次のデータを、前のデータに上書きせずに、USART_TDRレジスタに書き込めます。

  TXEIEビットがセットされている場合、このフラグは割込みを生成します。

  送信が行われているとき、USART_TDRレジスタへの書込み命令によってデータがTDRバッファに格納されます。そして、データは現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、USART_TDRレジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXEビットがセットされます。
- FIFOモードが有効になっている場合は、以下を示すためにハードウェアによってTXFNF（TXFIFOはフルではない）フラグがセットされます。
  – TXFIFOはフルではありません。
  – USART_TDRレジスタは空です。
  – 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。送信が行われているとき、USART_TDRレジスタへの書込み操作によってデータがTXFIFOに格納されます。データは、現在の送信の最後にTXFIFOからシフトレジスタにコピーされます。

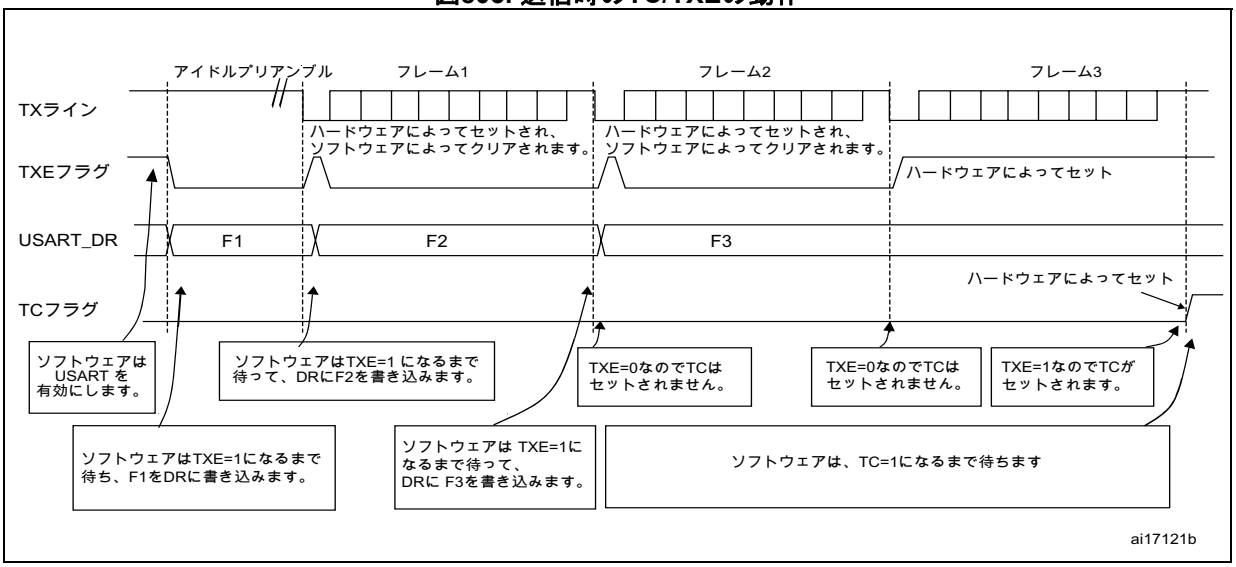
  TXFIFOがフルではない場合、USART_TDRレジスタへの書込み動作が終わった後も、TXFNFフラグは1のままに留まります。TXFIFOがフルになると、クリアされます。TXFNFIEビットがセットされている場合、このフラグは割込みを生成します。

  あるいは、TXFIFO閾値に達した時、割込みが生成され、データをFIFOに書き込むことができます。この場合、CPUは、プログラムされたトリガレベルによって定義されたデータのブロックを書き込むことができます。

  フレームが送信され（ストップビットの後）、TXEフラグ（FIFOモードの場合はTXFE）がセットされると、TCフラグはハイレベルになります。USART_CR1レジスタのTCIEビットがセットされると、割込みが生成されます。

USART_TDRレジスタに最後のデータを書き込んだ後は、USARTを無効にしたり、マイクロコントローラを低消費電力モードにする前にTCがセットされるまで待つ必要があります（図 803 : 送信時のTC/TXEの動作を参照）。

図803. 送信時のTC/TXEの動作



注： **FIFO管理が有効になっているときは、TXFNFフラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、Mビットに依存します（図801を参照）。

SBKRQビットに1が書き込まれた場合、現在のキャラクタ送信の完了後、TXラインにブレークキャラクタが送信されます。SBKFビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。USARTは、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに2bitの長さの間、ロジック1信号（ストップ）を挿入します。

SBKRQビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFOモードが有効になっている場合、TXFIFOがフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TEビットをセットすると、USARTは、最初のデータフレームの前にアイドルフレームを送信します。

66.5.7 USARTレシーバ

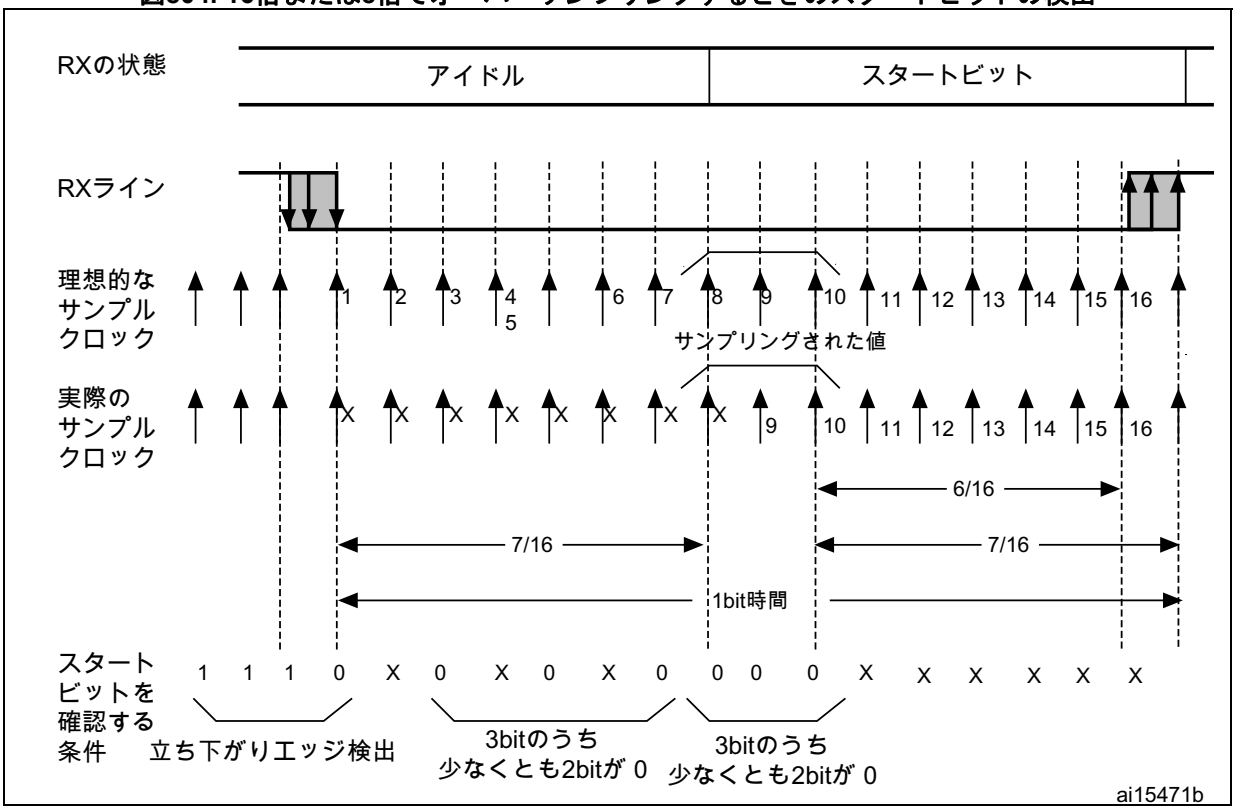
USARTは、USART_CR1レジスタのMビットに応じて、7または8または9bitのデータワードを受信できます。

スタートビット検出

スタートビット検出シーケンスは、16倍または8倍でオーバーサンプリングするときと同じです。

USARTでは、サンプルの特定シーケンスが認識されると、スタートビットが検出されます。このシーケンスは、1 1 1 0 X 0 X 0X 0X 0 X 0X 0です。

図804. 16倍または8倍でオーバーサンプリングするときのスタートビットの検出



注： **シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。**

サンプリングされた3つのビットが“0”の場合（第3、第5、および第7bitでの最初のサンプリングで3つのビットが“0”であり、第8、第9、および第10bitでの2回目のサンプリングでも3つのビットが“0”）、スタートビットが確認されます（RXNEフラグがセットされ、RXNEIE=1の場合は割込みが生成されるか、またはFIFOモードが有効な場合は、RXFNEフラグがセットされ、RXFNEIE=1の場合は割込みが生成されます）。

次の場合、スタートビットは確認されますが、NEノイズフラグがセットされます。

a) 両方のサンプリングについて、サンプリングされた3bitのうちの2bitが0の場合（第3、第5、および第7bitのサンプリングと、第8、第9、および第10bitのサンプリング）。

または

b) いずれか一方のサンプリングで（第3、第5、および第7bitでのサンプリングまたは第8、第9、および第10bitでのサンプリング）、3つのビットのうち2つが0の場合。

上記のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

キャラクタの受信

USARTの受信時には、データはRXピンを通じてLSBファースト（デフォルトの設定）でシフトアウトされます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. USART_CR1のMビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタUSART_BRRを使用して、希望するボーレートを選択します。
3. USART_CR2レジスタでストップビットの数をプログラミングします。
4. USART_CR1レジスタのUEビットに1を書き込んで、USARTを有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAR）を選択します。セクション 66.5.20：USARTおよびDMAを使用した連続通信の説明に基づいて、DMAレジスタを設定します。
6. USART_CR1レジスタのREビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFOモードが無効の場合、RXNEビットは、シフトレジスタの内容がRDRレジスタに転送されたことを示すためにセットされます。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFOモードが有効な場合、RXFIFOが空ではないことを示すためにRXFNEビットがセットされます。USART_RDRを読み出すと、RXFIFOに入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともにRXFIFOに格納されます。
- RXNEIE（FIFOモードが有効な場合はRXFNEIE） ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズ、パリティ、またはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効な場合は、RXNE フラグはバイトを受信するたびにセットされます。DMAが受信データレジスタを読み出すと、同フラグはクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMAリクエストのたびに、RXFIFOから1データが取り出されます。DMAリクエストは、RXFIFOが空ではないとき、すなわち、RXFIFOから読み出されるべきデータがあるときに、トリガされます。
- シングル-バッファモードでは、
  - FIFOモードが無効の場合、RXNEフラグのクリアは、ソフトウェアによるUSART_RDRレジスタからの読出しによって行われます。RXNE フラグは、USART_RQR レジスタの RXFRQビットに“1”をプログラムすることによってクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNEフラグをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE がセットされます。USART_RDRからの読出し動作のたびに、RXFIFOから1つのデータが取り出されます。RXFIFOが空になると、RXFNEフラグがクリアされます。RXFNEフラグは、USART_RQRのRXFRQビットに1をプログラムすることによってクリアすることもできます。RXFIFOがフルのとき、オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、

RXFIFO内の最初のエントリを読み出す必要があります。RXFNEIEビットがセットされている場合、RXFNEフラグは割込みを生成します。あるいは、RXFIFO閾値に達した時、割込みが生成され、データをRXFIFOから読み出すことができます。この場合、CPUは、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USARTはブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIEビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFOモードが無効の場合

  RXNEビットがリセットされていなくてキャラクタを受信すると、オーバーランエラーが発生します。

  RXNEビットがクリアされない限り、データをシフトレジスタからRDRレジスタに転送することはできません。RXNEフラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときにRXNEフラグがセットされていた場合、または前回のDMAリクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – OREビットがセットされます。
  – RDRの内容は失われません。前回のデータはUSART_RDRレジスタを読み出すことによって、入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIEビットまたはEIEビットがセットされている場合、割込みが生成されます。

- FIFOモードが有効な場合

  シフトレジスタが転送される準備ができていて、受信FIFOがフルのとき、オーバーランエラーが発生します。

  RXFIFOに1つの空き場所ができるまで、データをシフトレジスタからUSART_RDRレジスタに転送することはできません。RXFIFOが空でないとき、RXFNEフラグがセットされます。

  RXFIFOがフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – OREビットがセットされます。
  – RXFIFO の最初のエントリは失われません。それは、USART_RDR レジスタを読み出すことによって入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIEビットまたはEIEビットがセットされている場合、割込みが生成されます。

OREビットは、USART_ICRレジスタのORECFビットをセットすることによってリセットされます。

注： **OREビットがセットされた場合、少なくとも1個のデータが失われています。**

FIFOモードが無効になっているとき、2つの可能性があります。

- **RXNE=1の場合、有効な最後のデータは、受信レジスタRDRに格納され、読出しが可能です。**
- **RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR レジスタには読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータがRDRレジスタで読み出されると同時に新しい（そして失われた）データが受信されると発生します。**

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（セットおよびクロック制御（RCC）のセクションを参照）。クロックソースは、UEビットのセットによってUSARTを有効にする前に選ぶ必要があります。

クロックソースは、次の2つの基準に従って選択する必要があります。

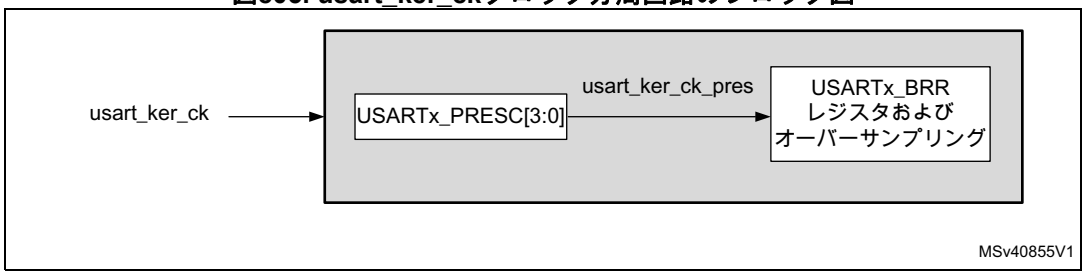
- USARTを低消費電力モードで使用できること
- 通信速度

クロックソース周波数は、usart_ker_ckです。

デュアルクロックドメインと低消費電力モードからのウェイクアップ機能がサポートされるときには、usart_ker_ckクロックソースはRCCで設定できます（**リセットおよびクロック制御（RCC）**のセクションを参照）。そうしない場合、usart_ker_ckクロックはusart_pclkと同じになります。

usart_ker_ckクロックは、USART_PRESCレジスタで定義されたプログラム可能な係数によって分周できます。

図805. usart_ker_ckクロック分周回路のブロック図



usart_ker_ckソースによっては、USARTはMCUが低消費電力モードのときにデータを受信することができます。受信データと選択されたウェイクアップモードに応じて、USARTより必要なときにMCUをウェイクアップし、ソフトウェアによるUSART_RDRレジスタの読出しによって、またはDMAによって受信データを転送できます。

他のクロックソースの場合、USART通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています（同期モードの場合を除く）。これにより、最大通信速度とノイズ／クロック精度の耐性との間の最適なトレードオフが図られます。

オーバーサンプリング方式は、USART_CR1レジスタのOVER8bitをボーレートクロックの16倍または8倍にプログラムすることによって選択できます（図806および図807を参照）。

アプリケーションに応じて、

- 高速（最大usart_ker_ck_pres/8）を達成するには8倍（OVER8=1）のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます（セク

ション 66.5.9： 2750ページのクロック偏差に対するUSARTレシーバの許容誤差を参照）。

- クロック偏差に対するレシーバの許容誤差を増やすには、16倍（OVER8=0）のオーバーサンプリングを選択します。この場合、最大速度は 最大 usart_ker_ck_pres/16に制限されます（ここで、usart_ker_ck_presはUSART入力クロックをプリスケーラで分周したものです）。

論理レベルの評価方法を選択するには、USART_CR3レジスタのONEBITビットをプログラミングします。次の2つのオプションが利用できます。

- 受信されたビットの中央にある3つのサンプルの多数決。この場合、多数決に使用された3つのサンプルが等しくないとき、NEビットがセットされます。
- 受信されたビットの中央にある1つのサンプル。

  アプリケーションに応じて、

  – ノイズの多い環境で操作するときは3つのサンプルの多数決方式（ONEBIT=0）を選択します。そしてノイズが検出された場合（表671を参照）は、サンプリング中にグリッチが発生していることになりますので、そのデータを除去します。

  – ラインがノイズフリーであるときは、1つのサンプルによる方式(ONEBIT=1)を選択し、クロック偏差に対するレシーバの許容誤差を増やします（セクション 66.5.9： 2750ページのクロック偏差に対するUSARTレシーバの許容誤差を参照）。この場合、NEビットはセットされません。

フレーム内でノイズが検出された場合：

- RXNEビット（FIFOモードが有効な場合はRXFNEビット）の立ち上がりエッジで、NEビットがセットされます。
- 無効なデータがシフトレジスタからUSART_RDRレジスタへ転送されます。
- シングルバイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成するRXNEビット（FIFOモードが有効な場合はRXFNEビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3レジスタのEIEビットがセットされている場合に割込みが発行されます。

NEビットは、ICRレジスタのNFCFビットをセットすることによってリセットされます。

注： **ノイズエラーはSPIモードではサポートされていません。**

8倍のオーバーサンプリングは、スマートカード、IrDA、およびLINの各モードでは使用できません。これらのモードでは、OVER8bitはハードウェアによって “0” に固定されています。

図806. データサンプリング（16倍のオーバーサンプリング）

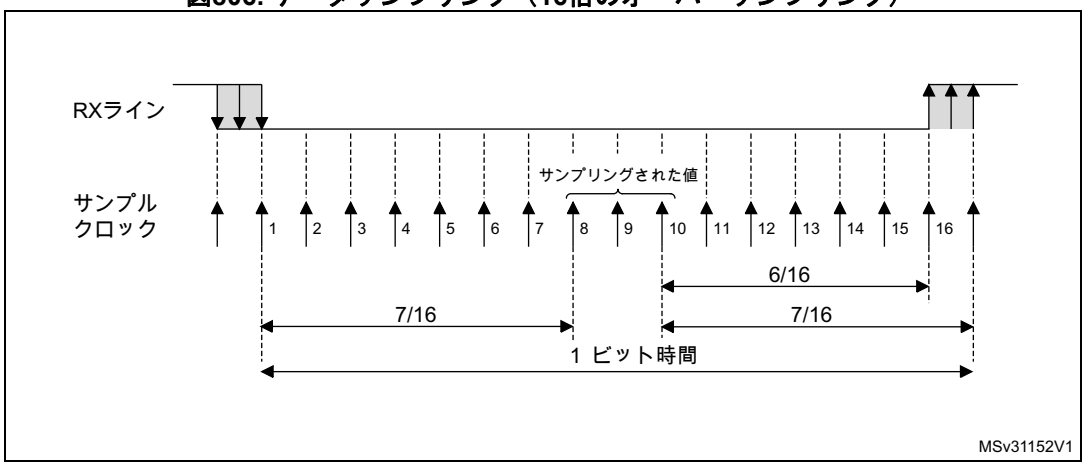


図807. データサンプリング（8倍のオーバーサンプリング）

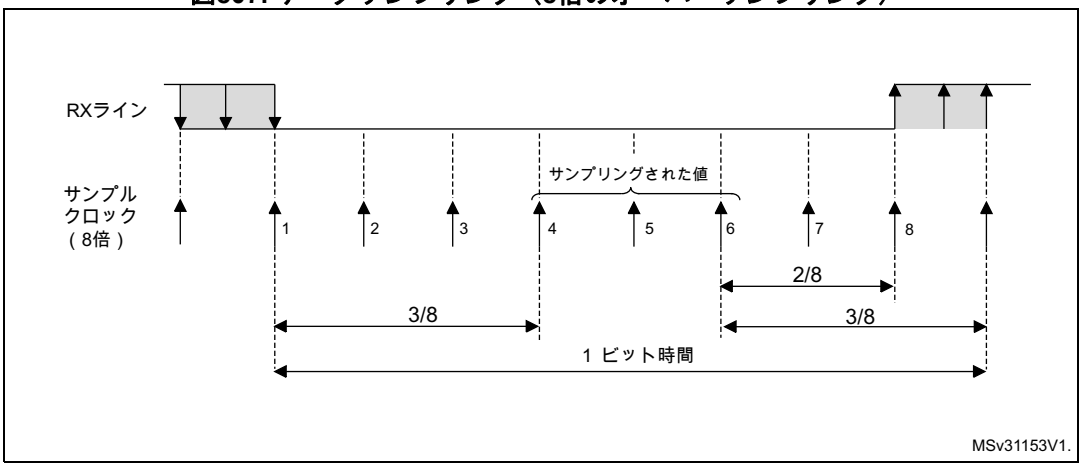


表671.サンプリングされたデータからのノイズ検出

サンプリングされた値	NEステータス	受信ビットの値
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FEビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタ（FIFO モードが有効な場合はRXFIFO）へ転送されます。
- シングルバイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成するRXNEビット（FIFOモードが有効な場合はRXFNEビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3レジスタのEIEビットがセットされている場合に割込みが発行されます。

USART_ICRレジスタのFECFに1を書き込むことによって、FEビットがリセットされます。

注： **フレーミングエラーはSPIモードではサポートされていません。**

受信時の設定可能なストップビット

受信するストップビット数は、USART_CRの制御ビットを通じて設定でき、通常モードでは1または2、スマートカードモードでは0.5または1.5にできます。

- **0.5個のストップビット（スマートカードモードでの受信）**：0.5個のストップビットでは、サンプリングは行われません。したがって、0.5個のストップビットが選択されている場合、フレーミングエラーやブレークフレームは検出されません。
- **1個のストップビット**：ストップビット1個のサンプリングは、8番目、9番目、および10番目のサンプルで行われます。
- 1.5個のストップビット（スマートカードモード）

  スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバブロックを有効にし（USART_CR1のRE=1）、ストップビットをチェックして、スマートカードがパリティエラーを検出したかどうかをテストする必要があります。

  パリティエラーが発生した場合、スマートカードはサンプリング時のデータ信号を強制的にローレベルにします（これは、フレーミングエラーとしてフラグされるNACK信号です）。その後、1.5個のストップビットの最後に、RXNEフラグ（FIFOモードが有効な場合はRXFNEフラグ）を通してFEフラグがセットされます。ストップビット1.5個のサンプリングは、16番目、17番目、および18番目のサンプルで（ストップビットの開始から1ボークロック周期後に）行われます。1.5個のストップビットは、2つの部分に分解できます。すなわち、何も起こらない0.5ボークロック周期と、途中でサンプリングが行われる通常の1ストップビット周期です（詳細はセクション 66.5.17： 2762ページのUSARTレシーバタイムアウトを参照）。
- 2個のストップビット

  ストップビット2個のサンプリングは、最初のストップビットの8番目、9番目、および10番目のサンプルで行われます。

  最初のストップビット中にフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。

  2番目のストップビットでは、フレーミングエラーの検査は行われません。最初のストップビットの最後に、RXNEフラグ（FIFOモードが有効な場合はRXFNEフラグ）がセットされます。

66.5.8 USARTボーレート生成

レシーバとトランスミッタ（RxとTx）のボーレートは、USART_BRRレジスタでプログラムされた値に設定されます。

式 1：標準USARTのボーレート（SPIモードを含む）（OVER8 = “0”または“1”）

16倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ck_pres}}{\text{USARTDIV}}$$

8倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{2 \times \text{usart_ker_ck_pres}}{\text{USARTDIV}}$$

式 2：スマートカード、LIN、およびIrDAモードのボーレート（OVER8 = 0）

ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ck_pres}}{\text{USARTDIV}}$$

USARTDIVは、符号なしの固定小数点数であり、USART_BRRレジスタでコード化されます。

- OVER8 = 0のとき、BRR = USARTDIVです。
- OVER8 = 1のとき、
  - BRR[2:0] = USARTDIV[3:0] であり、右に1bitシフトされます。
  - BRR[3] は、クリアされたままにする必要があります。
  - BRR[15:4] = USARTDIV[15:4]

注： **ボーカウンタは、USART_BRRへの書込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更してはなりません。**

16倍および8倍のオーバーサンプリングの場合、USARTDIVは16以上である必要があります。

USART_BRRレジスタの値からUSARTDIVを得る方法

例 1

usart_ker_ck_pres = 8MHzで9600baudを得るには、

- 16倍のオーバーサンプリングの場合：
  USARTDIV = 8 000 000/9600
  BRR = USARTDIV = 0d833 = 0x0341
- 8倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 8 000 000/9600
  USARTDIV = 1666,66 (0d1667 = 0x683)
  BRR[3:0] = 0x3 >> 1 = 0x1
  BRR = 0x681

例 2

usart_ker_ck_pres = 48MHzで921.6kbaudを得るには、

- 16倍のオーバーサンプリングの場合：
  USARTDIV = 48 000 000/921 600
  BRR = USARTDIV = 0x52 = 0x34
- 8倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 48 000 000/921 600
  USARTDIV = 104 (0d104 = 0x68)
  BRR[3:0] = USARTDIV[3:0] >> 1 = 0x8 >> 1 = 0x4
  BRR = 0x64

66.5.9 クロック偏差に対するUSARTレシーバの許容誤差

USARTの非同期レシーバは、クロックシステムの合計偏差がUSARTレシーバの許容誤差未満の場合のみ、正しく動作します。

合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART receiver 許容誤差}$$

ここで、

DWUは、低消費電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01の場合：

$$DWU = \frac{t_{WUUSART}}{11 \times Tbit}$$

M[1:0] = 00の場合：

$$DWU = \frac{t_{WUUSART}}{10 \times Tbit}$$

M[1:0] = 10の場合：

$$DWU = \frac{t_{WUUSART}}{9 \times Tbit}$$

$t_{WUUSART}$ は、スタートビットの立ち下がりエッジが検出され、クロック（ペリフェラルによるリクエスト）が動作可能状態になってペリフェラルに達し、レギュレータが動作可能状態になるまでの時間です。

USARTレシーバは、次の設定に応じて、表672および表673で指定された最大許容偏差まで、データを正しく受信できます。

- USART_CR1レジスタのMビットによって定義された9、10または11bitのキャラクタ長
- USART_CR1レジスタのOVER8bitによって定義された8倍または16倍のオーバーサンプリング
- USART_BRRレジスタのビットBRR[3:0] が0000に等しいかどうか。
- データのサンプリングに1bitを使用するか3bitを使用するか（USART_CR3 レジスタのONEBITビットの値に依存）。

表672.BRR [3:0] = 0000のときのUSARTレシーバの許容誤差

Mビット	OVER8bit = 0		OVER8bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.75%	4.375%	2.50%	3.75%
01	3.41%	3.97%	2.27%	3.41%
10	4.16%	4.86%	2.77%	4.16%

表673.BRR[3:0] が0000でないときのUSARTレシーバの許容誤差

Mビット	OVER8bit = 0		OVER8bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.33%	3.88%	2%	3%
01	3.03%	3.53%	1.82%	2.73%
10	3.7%	4.31%	2.22%	3.33%

注： **表672および表673で指定されたデータは、Mビット = 00のとき受信フレームに正確に10bit時間（M = 01の場合は11bit時間、M = 10の場合は9bit時間）のアイドル·フレームがいくつか含まれる特殊なケースでは若干異なることがあります。**

66.5.10 USART自動ボーレート検出

USARTは、1キャラクタの受信に基づいて、USART_BRRレジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません。

- 16倍のオーバーサンプリングのとき、ボーレートはusart_ker_ck_pres/65535からusart_ker_ck_pres/16までの範囲内です。
- 8倍のオーバーサンプリングのとき、ボーレートはusart_ker_ck_pres/65535からusart_ker_ck_pres/8までの範囲内です。

自動ボーレート検出を有効にする前に、USART_CR2レジスタのABRMOD[1:0] フィールドによって自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、4つのモードがあります。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

これらのモードは以下のとおりです。

- **モード0**：1のビットで始まるキャラクタ。

  この場合、USARTはスタートビットの時間を測定します（立ち下がりエッジから立ち上がりエッジまで）。
- **モード1**：10xxビットパターンで始まるキャラクタ。

  この場合、USARTはスタートと最初のデータビットの時間を測定します。低速な信号スロープの場合の精度を高めるために、測定は立ち下がりエッジから立ち下がりエッジまでで行われます。
- **モード2**：0x7Fキャラクタフレーム（LSBファーストモードでは0x7Fキャラクタ、またはMSBファーストモードでは0xFE）。

  この場合、ボーレートは、最初はスタートビットの終了時に更新され（BRs）、次にビット6の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて：BR6）。ビット0からビット6までがBRsでサンプリングされ、キャラクタの残りのビットはBR6でサンプリングされます。
- **モード3**：0x55キャラクタフレーム。

  この場合、ボーレートは、最初はスタートビットの終了時に更新され（BRs）、次にビット0の終了時に更新され（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて：BR0）、最後にビット6の終了時に更新されます（BR6）。ビット0はBRsでサンプリングされ、ビット1からビット6まではBR0でサンプリングされ、キャラクタの残りのビットはBR6でサンプリングさ

れます。並行して、RXラインの中間遷移ごとに別のチェックが行われます。RXの遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット0で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USART_BRRレジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USART_CR2レジスタのABRENビットをセットします。USARTは、RXラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USART_ISRレジスタのABRFフラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR値が破損して、ABREエラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が16から65536までのクロック周期でなく（16倍のオーバーサンプリング）、8から65536までのクロック周期でない（8倍のオーバーサンプリング））。

その後、ABRFフラグをリセットすることによって（“0” を書き込むことによって）、自動ボーレート検出を再開できます。

FIFOマネージメントが無効になっていて、自動ボーレートエラーが発生した場合、RXNEおよびFEビットを通してABREフラグがセットされます。

FIFOマネージメントが有効になっていて、自動ボーレートエラーが発生した場合、RXFNEおよびFEビットを通してABREフラグがセットされます。

FIFOモードが有効な場合、自動ボーレート検出は、RXFIFOの最初の場所のデータを使用して行う必要があります。したがって、自動ボーレート検出を開始する前に、USART_ISRレジスタのRXFNEフラグをチェックしてRXFIFOが空であることを確認してください。

注： **自動ボーレート操作中にUSARTが無効化された場合（UE=0）、BRR値が破損することがあります。**

66.5.11 USARTマルチプロセッサ通信

USARTのマルチプロセッサ通信が可能です（ネットワーク内で複数のUSARTを接続して）。たとえば、1つのUSARTをマスタとして、そのTX出力を別のUSARTのRX入力に接続することができます。一方、他のUSARTは、それぞれのTX出力の論理和をとってマスタのRX入力に接続して、スレーブとなります。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対するUSARTサービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USART_CR1レジスタのMMEビットをセットする必要があります。

注： **FIFOマネージメントが有効になっていてMMEがすでにセットされている場合は、MMEビットはクリアしてはなりません。クリアした場合はすぐに（2 usart_ker_ckサイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- USART_ISRレジスタのRWUビットは“1”にセットされます。特定の条件下では、USART_RQRレジスタのMMRQビットを通じて、RWUをハードウェアまたはソフトウェアによって自動的に制御できます。

USARTは、USART_CR1レジスタのWAKEビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

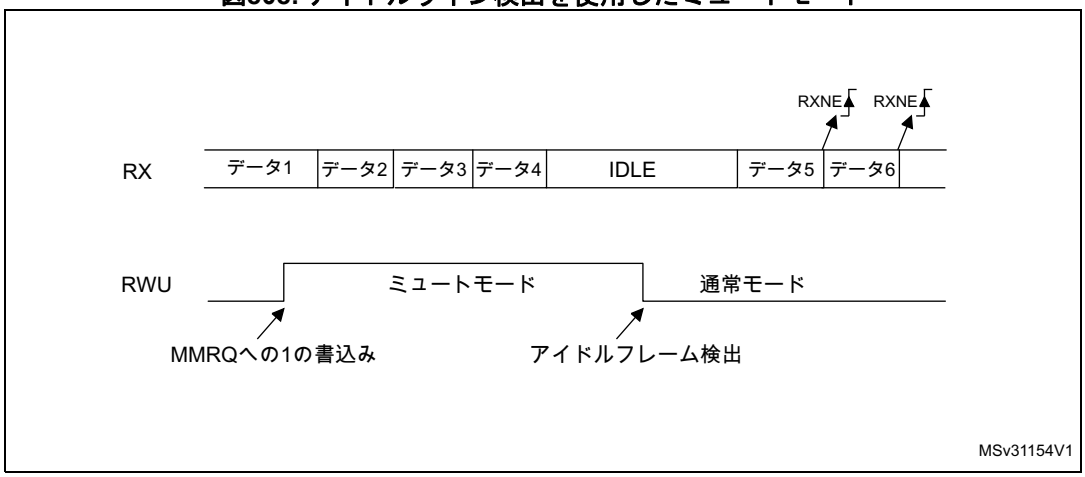
- WAKEビットがリセットされている場合は、アイドルライン検出
- WAKEビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQビットに1が書き込まれ、RWUが自動的にセットされたときには、USARTはミュートモードに入ります。

USARTは、アイドルフレームを検出するとウェイクアップします。その後、RWUビットはハードウェアによってクリアされますが、USART_ISRレジスタのIDLEビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図808に示します。

図808. アイドルライン検出を使用したミュートモード



注： **IDLEキャラクタがすでに経過しているときにMMRQがセットされた場合は、ミュートモードに入りません（RWUはセットされません）。**

ラインがIDLEのときにUSARTが有効にされた場合、1 IDLEフレーム後にアイドル状態が検出されます（1キャラクタフレームの受信後だけでなく）。

4bit／7bitアドレスマーク検出（WAKE=1）

このモードでは、MSBが ”1” の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは4または7 LSBです。7または4bitアドレス検出の選択は、ADDM7bitを使用して行われます。この4bit／7bitワードは、レシーバによって、USART_CR2レジスタのADDビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7bitおよび9bitデータモードでは、アドレス検出は、それぞれ6bitおよび8bitアドレス（ADD[5:0] およびADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USARTはミュートモードに入ります。この場合、RWUビットはハードウェアによってセットされます。USARTがミュートモードに入ったときには、このアドレスバイトに対してRXNEフラグはセットされず、割込みもDMAリクエストも発行されません。FIFO管理が有効になっている場合、ミュートモードに入る前に、RXFIFOに少なくとも1つの空き場所があることをソフトウェアによって確認する必要があります。

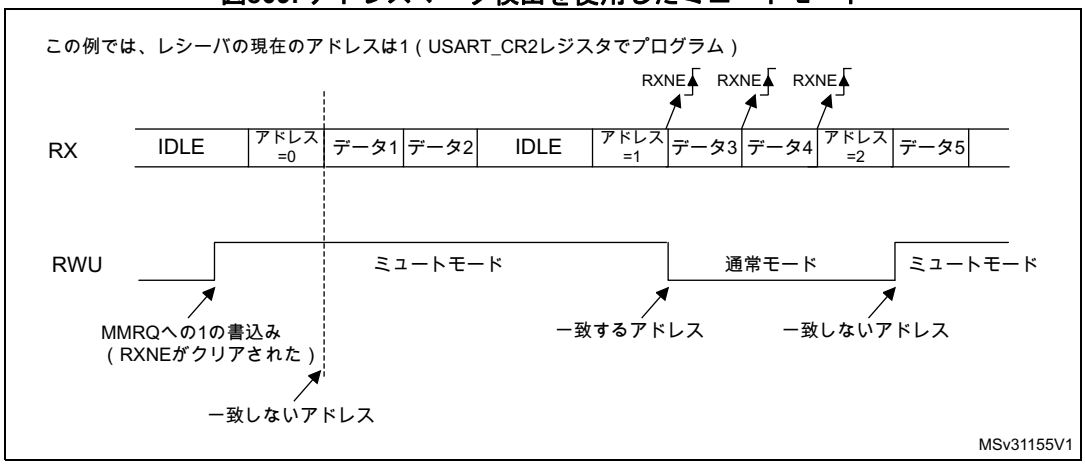
MMRQビットに1が書き込まれたときにも、USARTはミュートモードに入ります。この場合、RWUビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USARTはミュートモードを終了します。続いてRWUビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWUビットはクリアされているので、アドレスキャラクタに対してRXNE/RXFNEビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間にMMRQがセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を 図809に示します。

図809. アドレスマーク検出を使用したミュートモード



66.5.12 USART Modbus通信

USARTは、Modbus/RTUおよびModbus/ASCIIプロトコルの実装に対する基本的サポートを備えています。Modbus/RTUは、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USARTは、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1つのブロックの終了は2キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割込みは、USART_CR2レジスタのRTOENビットと、USART_CR1レジスタのRTOIEを通じて有効にする必要があります。2キャラクタ時間に対応する値（たとえば、22 xビット時間）をRTOレジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USARTは、キャラクター致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドでLF ASCIIコードをプログラムし、キャラクター致割込みを有効にすることによって（CMIE=1）、LFが受信されたときにソフトウェアに通知し、ソフトウェアはDMAバッファのCR/LFをチェックできます。

66.5.13 USARTパリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、USART_CR1レジスタのPCEビットをセットします。Mビットによって定義されたフレーム長に応じて、可能なUSARTフレームフォーマットを表674に示します。

表674.USARTフレームのフォーマット

Mビット	PCEビット	USARTフレーム(1)
00	0	| SB | 8bitデータ | STB |
00	1	| SB | 7bitデータ | PB | STB |
01	0	| SB | 9bitデータ | STB |
01	1	| SB | 8bitデータ | PB | STB |
10	0	| SB | 7bitデータ | STB |
10	1	| SB | 6bitデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。データレジスタでは、PBは常にMSB位置を取ります（Mビットの値に応じて、8または7番目）。

偶数パリティ

パリティビットは、6、7、または8 LSBビット（Mビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101であり、4bitがセットされた場合、偶数パリティが選択された場合（USART_CR1のPSビット = 0）、パリティビットは0になります。

奇数パリティ

パリティビットは、6、7、または8 LSBビット（Mビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101であり、4bitがセットされた場合、奇数パリティが選択された場合（USART_CR1のPSビット = 1）、パリティビットは1になります。

受信中のパリティチェック

パリティチェックに失敗した場合、USART_ISRレジスタのPEフラグがセットされ、USART_CR1レジスタのPEIEビットがセットされている場合は割込みが生成されます。PEフラグは、USART_ICRレジスタのPECFに1を書き込むことによってクリアされます。

送信中のパリティ生成

USART_CR1のPCEビットがセットされている場合、データレジスタに書き込まれたデータのMSBビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

66.5.14 USART LIN（Local Interconnection Network）モード

このセクションは、LINモードがサポートされるときにのみ適用されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

LINモードを選択するには、USART_CR2レジスタのLINENビットをセットします。LINモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのCLKEN
- USART_CR3レジスタのSTOP[1:0]、SCEN、HDSEL、およびIREN

LIN送信

LINマスタ送信の場合、セクション66.5.5に記述されている手順を適用する必要があります。通常のUSART送信と同じですが、次のような違いがあります。

- 8bitのワード長を設定するにはMビットをクリアします。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13個の “0” ビットがブレークキャラクタとして送信されます。その後、値 “1” の2bitが送信され、次のSTART検出が可能になります。

LIN受信

LINモードが有効になると、ブレーク検出回路が有効になります。この検出は、通常のUSARTレシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生のたびにブレークが検出できます。

レシーバが有効になると（USART_CR1レジスタのRE=1）、RX入力のSTART信号を探します。スタートビットの検出方法は、ブレークキャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます（8、9、および10番目のサンプル）。10個（USART_CR2レジスタのLBDL=0の場合）または11個（USART_CR2レジスタのLBDL=1の場合）の連続したビットが0として検出され、その後にデリミタキャラクタが続く場合、USART_ISRレジスタのLBDFフラグがセットされます。LBDIEビット=1の場合、割込みが生成されます。ブレークを確認する前に、RXラインがハイレベルに戻ったことを知らせるデリミタが検査されます。

この10または11が発生する前に“1”がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LINモードが無効にされた場合（LINEN=0）、レシーバは、ブレーク検出を考慮することなく、通常のUSARTとして機能し続けます。

LINモードが有効にされた場合（LINEN=1）、フレーミングエラーが発生（つまり、ブレークフレームの場合と同様に、“0” の位置でストップビットを検出）すると、レシーバは停止し、ブレーク検出回路が “1”（ブレークワードが完全でなかった場合）またはデリミタキャラクタ（ブレークが検出された場合）を受信するまで停止状態を維持します。

ブレーク検出回路ステートマシンの動作とブレークフラグを図 810：2757ページのLINモードでのブレーク検出（11bitブレーク長、LBDL=1）に示します。

ブレークフレームの例を図 811：2758ページのLINモードでのブレーク検出とフレーミングエラー検出に示します。

図810. LINモードでのブレーク検出（11bitブレーク長、LBDL=1）

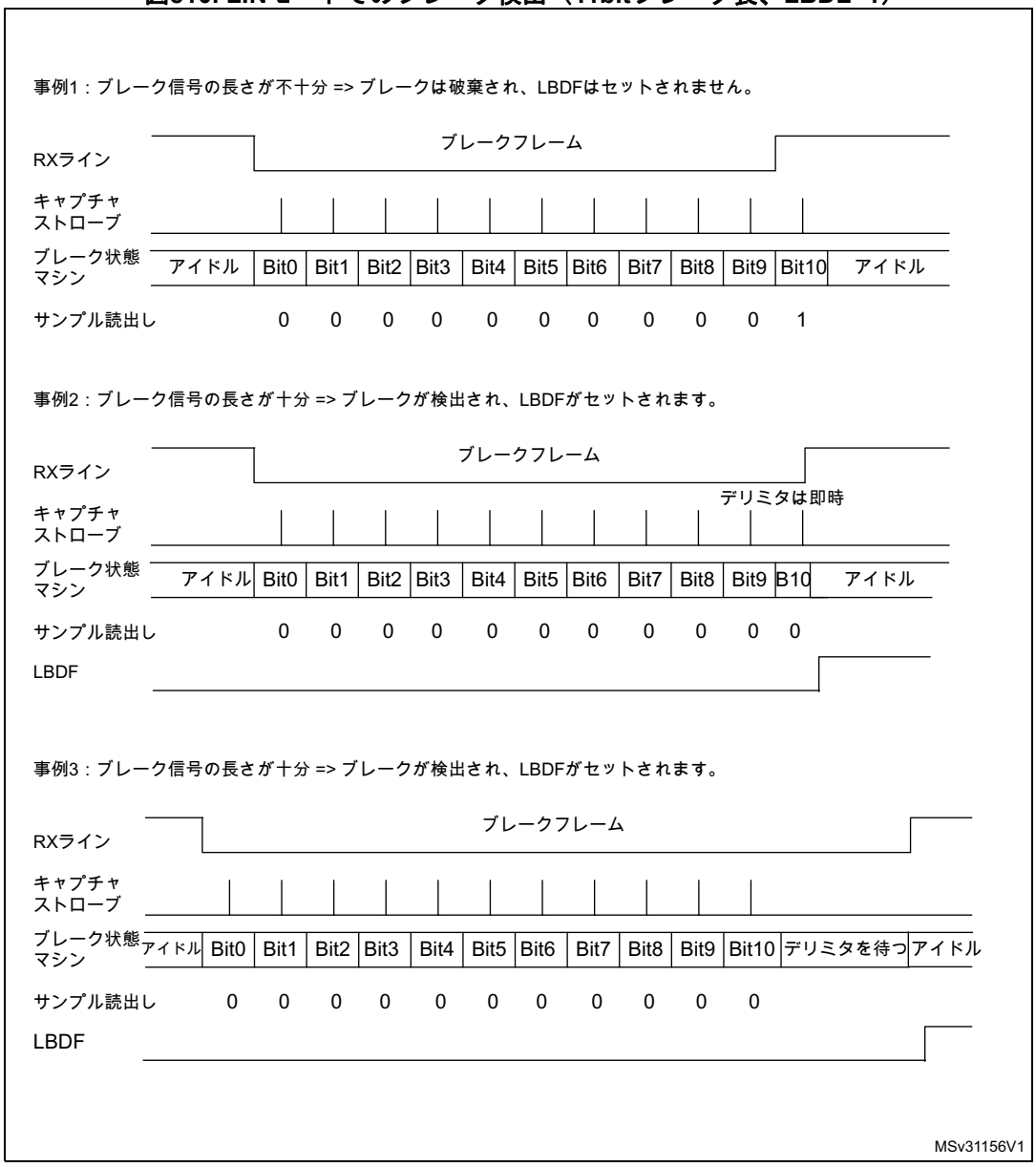
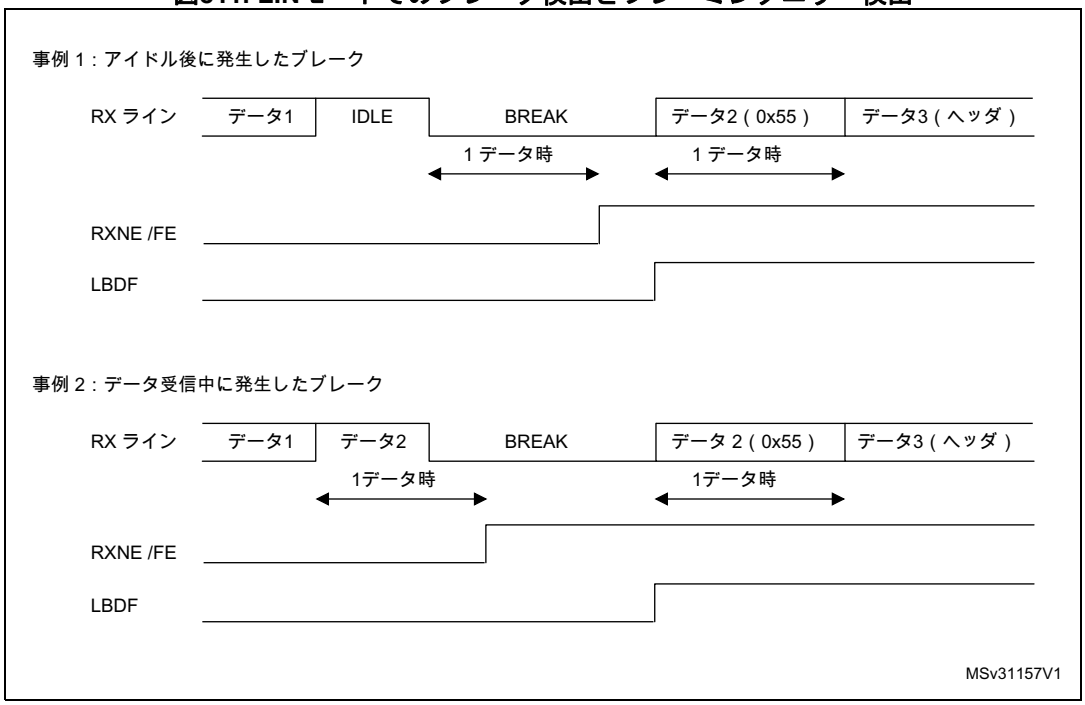


図811. LINモードでのブレーク検出とフレーミングエラー検出



66.5.15 USART同期モード

マスタモード

同期マスタモードを選択するには、USART_CR2レジスタのCLKENビットを“1”にプログラムします。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのLINENビット
- USART_CR3レジスタのSCEN、HDSEL、およびIRENビット

このモードでは、USARTを使用して、双方向同期シリアル通信をマスタモードで制御できます。CKピンはUSARTトランスミッタクロックの出力です。スタートビットとストップビットの処理中には、CKピンにクロックパルスは送信されません。USART_CR2レジスタのLBCLビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USART_CR2レジスタのCPOLビットは、クロック極性を選択するために使用され、USART_CR2レジスタのCPHAビットは、外部クロックの位相を選択するために使用されます（図812、図813、および図814を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、外部CKクロックは起動されません。

同期マスタモードでは、USARTトランスミッタは非同期モードの場合とまったく同じように動作します。しかし、CPOLとCPHAに基づいてCKがTXと同期するので、TX上のデータが同期します。

同期マスタモードでは、USARTレシーバは非同期モードの場合とは異なる動作をします。REが1にセットされた場合、データはオーバーサンプリングなしで、CK（CPOLとCPHAに応じて立ち上がりまたは立ち下がりエッジ）でサンプリングされます。ボーレート（1/16bit時間）に応じたセットアップ時間とホールド時間に従う必要があります。

注： マスタモードでは、CKピンはTXピンと連携して動作します。したがって、クロックが供給されるのは、トランスミッタが有効であり（TE=1）、データが送信中（USART_TDRデータレジスタへの書込み）の場合に限られます。つまり、データ送信を行わずに同期データを受信することはできません。

図812. USARTの同期マスタ送信の例

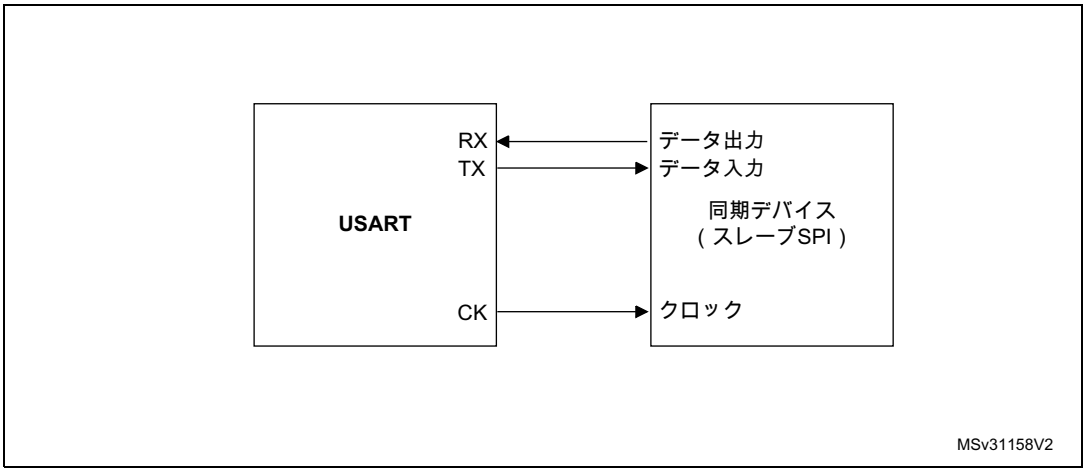


図813. 同期マスタモードでのUSARTデータクロックタイミング図（Mビット = 00）

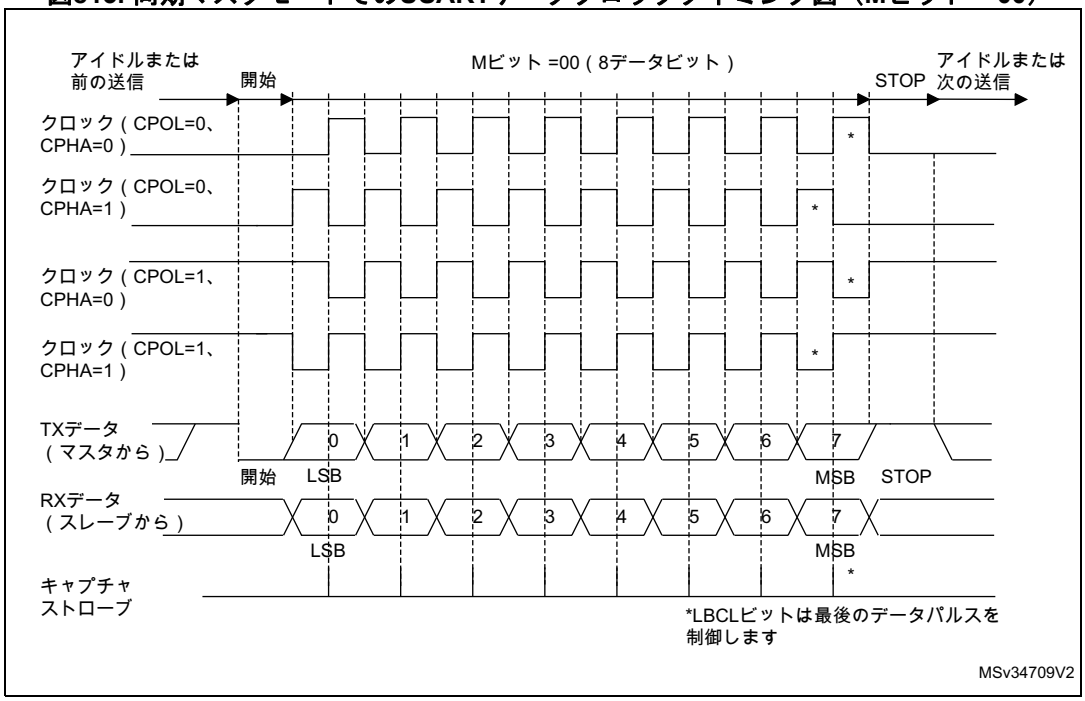
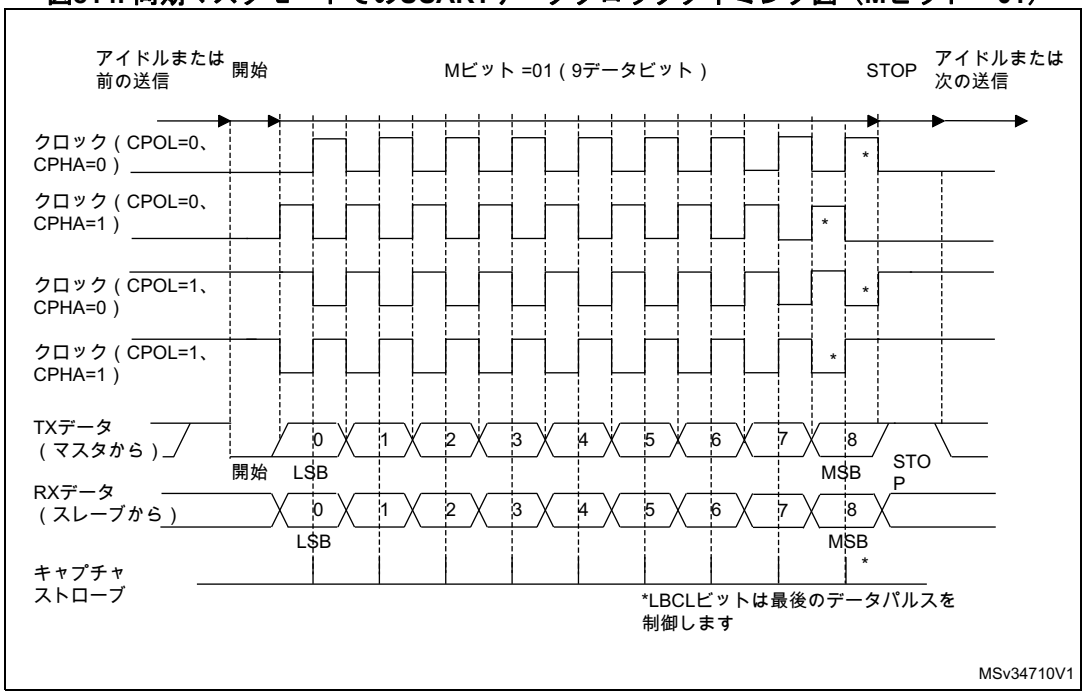


図814. 同期マスタモードでのUSARTデータクロックタイミング図（Mビット = 01）



スレーブモード

同期スレーブモードを選択するには、USART_CR2レジスタのSLVENビットを ”1” にプログラムします。同期スレーブモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのLINENおよびCLKENビット
- USART_CR3レジスタのSCEN、HDSEL、およびIRENビット

このモードでは、USARTを使用して、双方向同期シリアル通信をスレーブモードで制御できます。CKピンは、スレーブモードのUSARTの入力です。

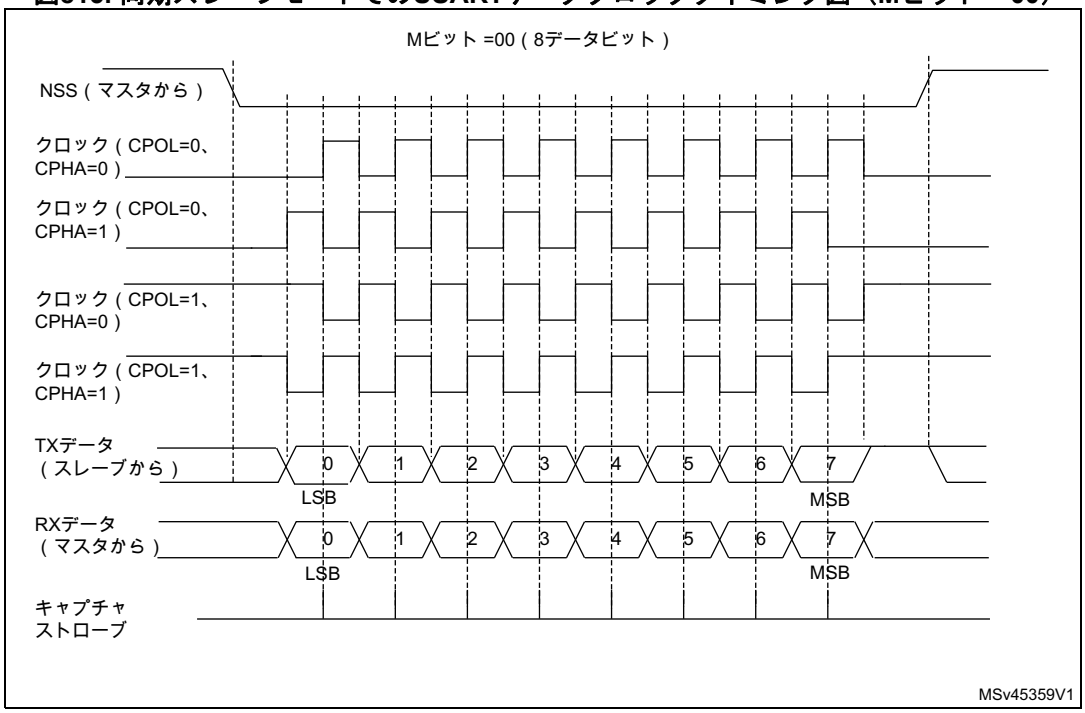
注： **ペリフェラルが SPI のスレーブモードで使用されるとき、ペリフェラルのクロックソースの周波数（usart_ker_ck_pres）はCK入力周波数の3倍より大きくなければなりません。**

USART_CR2レジスタのCPOLビットとCPHAビットは、それぞれクロック極性と外部クロックの位相を選択するために使用されます（図815を参照）。

アンダーランエラーフラグはスレーブ送信モードで使用できます。ソフトウェアがUSART_TDRにまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。

スレーブはハードウェアおよびソフトウェアのNSS管理をサポートしています。

図815. 同期スレーブモードでのUSARTデータクロックタイミング図（Mビット = 00）



スレーブ選択（NSS）ピンの管理

ハードウェアまたはソフトウェアのスレーブ選択管理は、USART_CR2レジスタのDIS_NSSビットを使用して設定することができます。

- ソフトウェアNSS管理（DIS_NSS = 1）

  SPIスレーブが常に選択され、NSS入力ピンは無視されます。

  外部NSSピンは他のアプリケーションで使用できます。

- ハードウェアNSS管理（DIS_NSS = 0）

  SPIスレーブ選択はNSS入力ピンに依存します。NSSがローレベルのときにスレーブが選択され、NSSがハイレベルのときには選択されません。

注： **USARTが無効にされたときには（UE="0"）、クロックパルスが正常に機能するように、LBCL（SPIマスタモードでのみ使用）、CPOL、およびCPHAビットを選択する必要があります。**

SPIスレーブモードでは、マスタ通信を開始する前に（またはクロックが安定しているときのフレーム間で）、USARTを有効にする必要があります。そうしないと、マスタがフレームの中央に位置しているときにUSARTスレーブが有効化されると、スレーブはマスタとの同期がとれなくなります。スレーブのデータレジスタは、通信クロックの最初のエッジまたは現在の通信の終了より前に、準備ができている必要があります。そうしないと、SPIスレーブはゼロを送信します。

SPIスレーブのアンダーランエラー

アンダーランエラーが起きると、USART_ISRレジスタのUDRフラグがセットされ、ソフトウェアによってアンダーランエラーフラグがクリアされるまでSPIスレーブは最後のデータを送信し続けます。

アンダーランフラグはフレームの開始時にセットされます。USART_CR3レジスタのEIEビットがセットされている場合、アンダーランエラー割込みがトリガされます。

アンダーランエラーフラグは、USART_ICRレジスタのビット UDRCF をセットすることによってクリアされます。

アンダーランエラーの場合、TDRレジスタに書き込むことは可能です。アンダーランエラーをクリアすると、新しいデータを送信できます。

アンダーランエラーが起きて、TDRに書き込まれる新しいデータがない場合は、フレームの終りにTCフラグがセットされます。

注： **データがUSART_TDRに書き込まれた時点がCKの最初の送信エッジに近すぎると、アンダーランエラーが起きることがあります。このアンダーランエラーを避けるために、USART_TDRへの書込みは、最初のCKエッジよりusart_ker_ckの3サイクル前に行う必要があります。**

66.5.16 USART単線半二重通信

単線半二重モードを選択するには、USART_CR3レジスタのHDSELビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのLINENおよびCLKENビット
- USART_CR3レジスタのSCENおよびIRENビット

USARTは、単線半二重のプロトコルに従うように設定できます。この場合、TXラインとRXラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3レジスタの制御ビットHDSELで行います。

HDSELビットに1が書き込まれると、

- TXラインとRXラインが内部接続されます。
- RXピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TXが外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/Oを設定する必要があります。

この点を除くと、通信プロトコルは通常のUSARTモードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TEビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

66.5.17 USARTレシーバタイムアウト

レシーバタイムアウト機能を有効にするには、USART_CR2制御レジスタのRTOENビットをセットします。

タイムアウト経過時間はUSART_RTORレジスタのRTOビットフィールドを使用してプログラムします。

レシーバタイムアウトカウンタは次の時点からカウントを開始します。

- STOP = 00またはSTOP = 11の場合、ストップビットの最後から
- STOP = 10の場合、2番目のストップビットの最後から。
- STOP = 01の場合、ストップビットの最初から。

タイムアウト経過時間が過ぎると、USART_ISR レジスタの RTOF フラグがセットされます。USART_CR1レジスタのRTOIEビットがセットされている場合、タイムアウトが生成されます。

66.5.18 USARTスマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

スマートカードモードを選択するには、USART_CR3レジスタのSCENビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのLINENビット
- USART_CR3レジスタのHDSELおよびIRENビット

スマートカードにクロックを供給するためにCLKENビットもセットすることができます。

スマートカードインタフェースは、ISO 7816-3標準で定義された非同期スマートカードプロトコルをサポートするように設計されています。T=0（キャラクタモード）とT=1（ブロックモード）の両方がサポートされます。

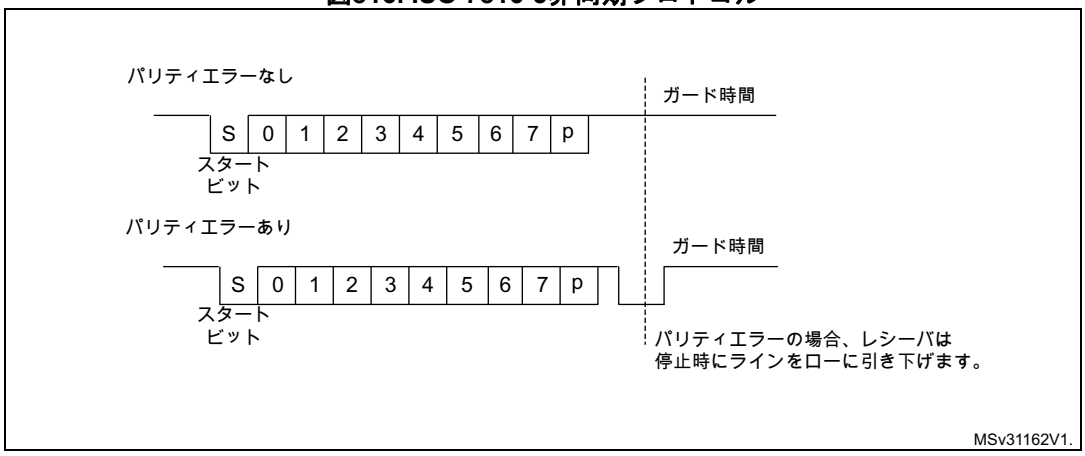
USARTは次のように設定してください。

- 8bit＋パリティ：USART_CR1レジスタのM=1およびPCE=1
- データ送受信時の1.5個のストップビット：USART_CR2レジスタのSTOP=11受信にはストップビット0.5個を選択することも可能です。

T=0（キャラクタ）モードでは、パリティエラーはガードタイム中の各キャラクタの終わりに示されます。

図816に、パリティエラーの有無によるデータラインの状況の変化の例を示します。

図816. ISO 7816-3非同期プロトコル



スマートカードと接続されると、USARTのTX出力は、やはりスマートカードによって駆動される双方向ラインを駆動します。TXピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

- 送信シフトレジスタからのデータの送信は、少なくとも1/2ボークロックの遅れが保証されます。通常動作では、フルの送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された1/2ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって（NACK）、この条件をUSARTに知らせます。このNACK信号（1ボークロックの間、送信ラインをローに引き下げ）は、1.5個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USARTは、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNTビットフィールドでプログラムされます。プログラムされた再試行回数後もUSARTがNACKを受信し続けた場合は、送信を停止して、エラーをフレーミ

ングエラーとして通知します。TXEビット（FIFOモードが有効な場合はTXFNFビット）は、USART_RQRレジスタのTXFRQビットを使用してセットできます。

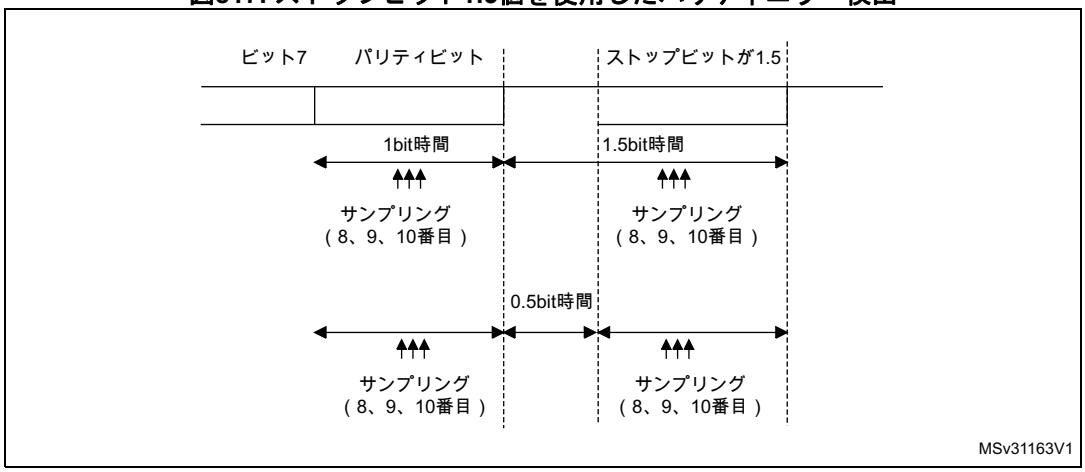
- 送信時のスマートカード自動再試行：USARTによるNACKの検出と反復キャラクタのスタートビットの間に2.5ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちにTCビットがセットされます（ガードタイムはありません）。ソフトウェアで再び繰り返したい場合は、規格によって指定されている2ボー周期以上を確保する必要があります。
- 1.5個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後1ボークロック周期間、送信ラインがローに引き下げられます。これは、USARTに送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK制御ビットがセットされている場合、パリティエラーはレシーバによって“NACK” され、そうでない場合、NACKは送信されません（T=1モードで使用されます）。受信したキャラクタにエラーがあった場合、RXNE（FIFOモードが有効な場合はRXFNE）／受信DMAリクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNTビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USARTはNACKの送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行：USARTがカードをNACKしたが、カードがキャラクタを繰り返さなかった場合、BUSYフラグはセットされたままです。
- 送信時、USARTは2つの連続するキャラクタの間にガードタイム（ガードタイムレジスタでプログラム）を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的のCGT（7816-3仕様で定義されているCharacter Guard Time）から12（1キャラクタの時間）を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TCがアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TCは強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TCがハイにアサートされます。TCBGTフラグは、ガード時間完了を待たずにデータ転送の終わりを検出するために使用できます。このフラグは、フレーム送信終了後およびカードからNACKを受信しなかった場合にのみセットされます。
- TCフラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからのNACKによってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、このNACKをスタートビットとして検出しません。ISOプロトコルによれば、受信されるNACKの期間は1または2ボークロック周期です。
- レシーバ側では、パリティエラーが検出されてNACKが送信された場合、レシーバはこのNACKをスタートビットとして検出しません。

注： **スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ0x00は、ブレークではなくデータとして処理されます。**

TEビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISOプロトコルでは定義されていません。

図817に、USARTによるNACK信号のサンプリング方法を示します。この例では、USARTはデータを送信中であり、ストップビットが1.5個組み込まれています。データとNACK信号の整合性を検査するために、USARTのレシーバ部が有効にされます。

図817. ストップビット1.5個を使用したパリティエラー検出



USARTは、CK出力を通じてスマートカードにクロックを供給できます。スマートカードモードでは、CKは通信に関係せず、5bitのプリスケーラを通じて単に内部のペリフェラル入力クロックから取得されます。この分周比は、USART_GTPRレジスタで設定されます。CKの周波数はusart_ker_ck_pres/2からusart_ker_ck_pres/62の間でプログラミングできます。ここで、usart_ker_ck_presは、プログラムされたプリスケーラで分周されたペリフェラル入力クロックです。

ブロックモード（T=1）

T=1（ブロック）モードでは、パリティエラー送信は、UART_CR3レジスタのNACKビットをクリアすることによって無効化されます。

ブロックモードでスマートカードからの読出しをリクエストするときには、ソフトウェアはRTORレジスタをBWT（ブロックウェイトタイム）- 11の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、タイムアウト割込みが生成されます。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE/RXFNE割込みによって通知されます。

注： **ブロックモードのスマートカードからの読出しにUSARTをDMAモードで使用するときでも、RXNE/RXFNE割込みを有効にする必要があります。並行して、DMAは最初の受信バイトの後でのみ有効にする必要があります。**

2つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後（RXNE/RXFNE割込み）、RTOレジスタをCWT（キャラクタウェイトタイム -11）の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT未満の時間内に新しいキャラクタを送信しなかった場合、USARTはRTOFフラグと割込み（RTOIEビットがセットされているとき）によって、これをソフトウェアに通知します。

注： **スマートカードプロトコルの定義にあるように、BWT/CWT値は最後のキャラクタの開始（スタートビット）から定義する必要があります。RTOレジスタは、最後のキャラクタ自体の長さを考慮して、それぞれBWT -11またはCWT -11にプログラムする必要があります。**

ブロック長カウンタは、USARTが受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USARTの送信時にリセットされます。ブロックの長さは、スマートカードによってブロックの3番目のバイト（プロローグフィールド）で伝えられます。この値をUSART_RTORレジスタのBLENフィールドでプログラムする必要があります。DMAモードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値（0x0）にプログラムする必要があります。この値では、4番目の受信キャラクタの後に割込みが生成されます。ソフトウェアはLENフィールド（3番目のバイト）を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割込み駆動受信モードでは、ブロックの長さはソフトウェアによって、またはBLEN値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLENの最大値（0xFF）をプログラムすることができます。実際の値は、3番目のキャラクタの受信後にプログラムされます。

ブロックがLRC水平冗長検査（1エピローグバイト）を使用している場合は、BLEN=LENです。ブロックがCRCメカニズム（2エピローグバイト）を使用している場合は、BLEN=LEN+1をプログラムする必要があります。合計ブロック長（プロローグ、エピローグ、および情報フィールドを含む）は、BLEN+4に等しくなります。ブロックの終わりはEOBFフラグと割込み（EOBIEビットがセットされているとき）によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりはRTO割込みによって通知されます（キャラクタウェイトタイムオーバーフロー）。

注： **エラーチェックコード（LRC/CRC）は、ソフトウェアによって計算／確認されなければなりません。**

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの2つコンベンションを定義しています。

ダイレクトコンベンションは、LSBファースト、論理ビットの値1がラインのH状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビットMSBFIRST=0、DATAINV=0（デフォルト値）をプログラムする必要があります。

インバースコンベンションは、MSBファースト、論理ビットの値1が単線のL状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビットMSBFIRST=1、DATAINV=1をプログラムする必要があります。

注： **論理データ値が反転されると（0=H、1=L）、パリティビットも同じように反転されます。**

カードのコンベンションを認識するために、カードは最初のキャラクタTSをATR（Answer To Reset）フレームの最初のキャラクタとして送信します。TSには、LHHL LLL LLHとLHHL HHH LLHの2つのパターンがあります。

- (H) LHHL LLL LLHは、インバースコンベンションをセットアップします。状態Lが値1にエンコードされ、モーメント2は最上位ビットを含みます（MSBファースト）。インバースコンベンションによってデコードされると、送受信されたバイトは3Fに等しくなります。
- (H) LHHL HHH LLHは、ダイレクトコンベンションをセットアップします。状態Hが値1にエンコードされ、モーメント2は最下位ビットを含みます（LSBファースト）。ダイレクトコンベンションによってデコードされると、送受信されたバイトは3Bに等しくなります。

キャラクタパリティは、2から10までの9個のモーメントに1にセットされた偶数個のビットがあったときに正しいとみなされます。

USARTはカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USARTがダイレクトコンベンション（デフォルト）で設定され、カードがインバースコンベンションで応答した場合、TS = LHHL LLL LLHになり、USARTが受信したキャラクタは03になり、パリティは奇数です。

したがって、TSパターン認識には2つの方式を使用できます。

方式1

USARTは、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TSパターンの受信によってパリティエラー割込みと、カードに対するエラー信号が生成されます。

- パリティエラー割込みは、カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USARTをインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割込みに応答して、ソフトウェアはUSARTを再プログラムし、カードに対して新しいリセットコマンドを生成してから、TSを再び待つこともできます。

方式2

USARTは9bit／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように2つのTSパターンのいずれかを受信します。

(H) LHHL LLL LLH = 0x103 -> インバースコンベンションを選択

(H) LHHL HHH LLH = 0x13B -> ダイレクトコンベンションを選択

ソフトウェアは受信されたキャラクタをこの2つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えてUSARTをプログラムします。

2つのうちのどちらも認識されなかった場合、ネゴシエーションを再開始するためにカードリセットが生成されます。

66.5.19 USART IrDA SIR ENDECブロック

このセクションは、IrDAモードがサポートされるときにのみ適用されます。セクション 66.4： 2731 ページのUSARTの実装を参照してください。

IrDAモードを選択するには、USART_CR3レジスタのIRENビットをセットします。IrDAモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2レジスタのLINEN、STOP、およびCLKENビット
- USART_CR3レジスタのSCENおよびHDSELビット

IrDA SIR物理層は、ロジック0を赤外光パルスとして表現するRZI（Return to Zero, Inverted）変調方式の使用を指定します（図818を参照）。

SIR送信エンコーダは、USARTからのNRZ（Non Return to Zero）送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線LEDに送信されます。SIR ENDECの場合、USARTは最大115.2 kbaudのビットレートしかサポートしません。通常モードでは、送信パルス幅はビット周期の3/16に規定されます。

SIR受信デコーダは、赤外線検出回路からのRZ（Return to Zero）ビットストリームを復調し、受信したNRZシリアルビットストリームをUSARTに出力します。デコーダの入力は、アイドル状態でノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDAは半二重通信プロトコルです。トランスミッタがビジーである場合（USARTがIrDAエンコーダにデータを送信しているとき）、IrDAデコーダはIrDA受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USARTがデコードされたデータを受信している）場合、IrDAはUSARTからIrDAへのTX上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。

- 0はハイパルスとして送信され、1は0として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の3/16と規定されます（図819を参照）。
- SIRデコーダは、IrDA準拠の受信信号をUSART用のビットストリームに変換します。
- SIR受信ロジックは、ハイ状態を論理値1とみなし、ローパルスを論理値0とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR出力は、アイドル時にロー状態になります。
- IrDA仕様では、1.41 usより大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ端のグリッチ検出回路は、PSC　2周期（PSCはUSART_GTPRでプログラムされたプリスケーラ値）より小さな幅のパルスをフィルタします。PSC 1周期より小さな幅のパルスは常に拒否されますが、1周期以上2周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2周期より大きな幅のパルスは、パルスとして受け入れられます。PSC=0のとき、IrDAエンコーダ/デコーダは機能しません。
- レシーバは、低消費電力トランスミッタと通信できます。
- IrDAモードでは、USART_CR2レジスタのストップビットを“1ストップビット”に設定する必要があります。

IrDA低消費電力モード

- トランスミッタ

  低消費電力モードでは、パルス幅はビット周期の3/16に維持されません。代わりに、パルス幅は低消費電力ボーレート（最小で1.42MHz）の3倍となります。一般に、この値は1.8432MHz（1.42MHz < PSC < 2.12MHz）です。低消費電力モードのプログラム可能な分周器は、この値を得るためにシステムクロックを分周します。

- レシーバ

  低消費電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USARTは1/PSCよりも短いパルスを破棄する必要があります。有効なローレベルは、その期間がIrDA低電力ボークロック（USART_GTPRのPSC値）の2周期分を超える場合にのみ受け入れられます。

注： **PSC 2周期未満1周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。**

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA物理層仕様では、送信と受信の間に最小10 msの遅延を指定しています（IrDAは半二重プロトコルです）。

図818. IrDA SIR ENDECブロック図

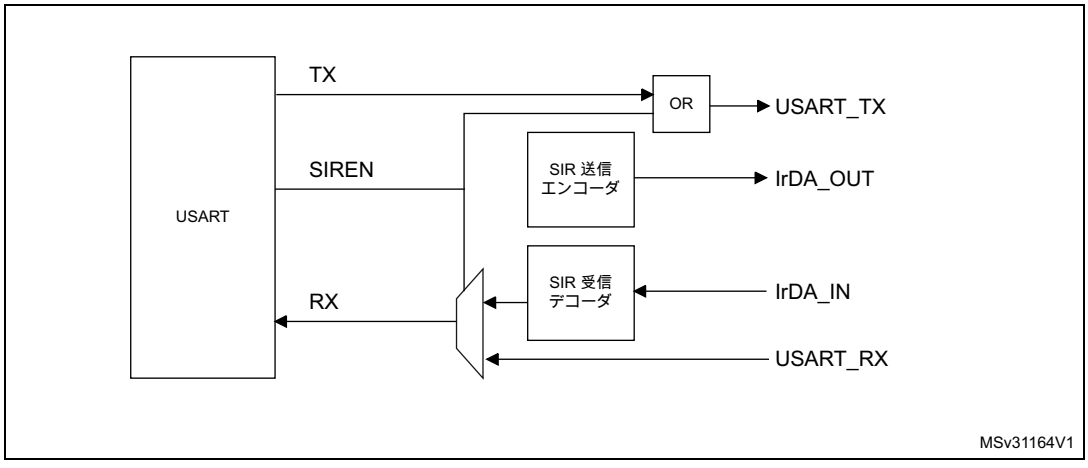
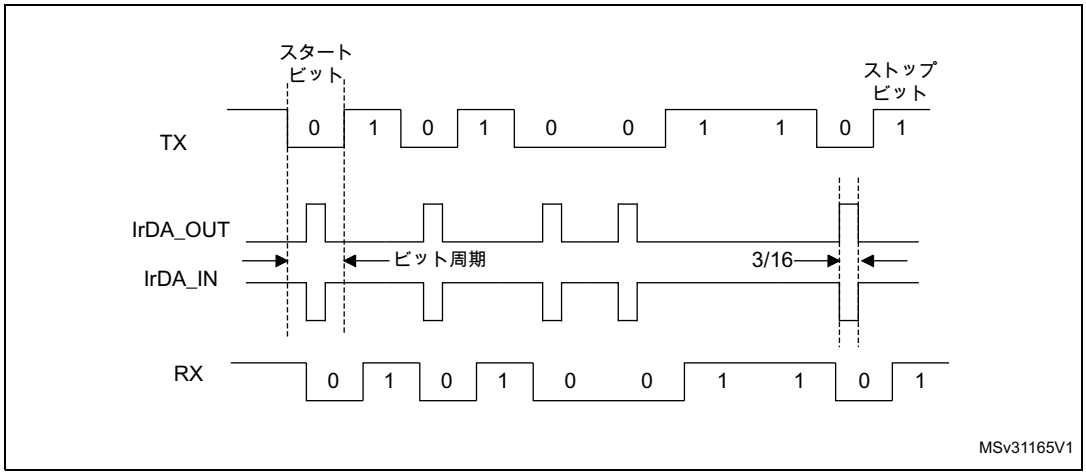


図819. IrDAデータ変調（3/16）- 通常モード



66.5.20 USARTおよびDMAを使用した連続通信

USARTは、DMAを使用して連続通信を行うことができます。RxバッファとTxバッファに対するDMAリクエストは、それぞれ独立して生成できます。

注： **DMAモードがサポートされるかどうかについては、セクション 66.4：2731ページのUSARTの実装を参照してください。DMAがサポートされない場合は、セクション66.5.7の説明に従ってUSARTを使用してください。FIFOが無効の場合、連続通信を行うには、USART_ISRレジスタのTXE/ RXNEフラグをクリアします。**

DMAを使用した送信

DMAモードでの送信を有効にするには、USART_CR3レジスタのDMATビットをセットします。TXEフラグ（FIFOモードが有効な場合はTXFNFフラグ）がセットされるたびに、データは、DMAペリフェラル（ダイレクトメモリアクセスコントローラ（DMA）セクション）を使用して設定されたSRAM領域からUSART_TDRレジスタにロードされます。DMAチャネルをUSART送信用に割り付けるには、次の手順を実行します（xはチャネル番号を示します）

1. DMA制御レジスタにUSART_TDRレジスタのアドレスを書き込み、これを転送先として設定します。データは、各TXE（またはFIFOモードが有効な場合はTXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域からUSART_TDRレジスタにロードされます。
3. 転送すべきバイト総数をDMA制御レジスタに設定します。
4. チャネル優先順位をDMAレジスタで設定します。
5. アプリケーションで必要とされる1/2転送終了、転送完了後のDMA割込み生成を設定します。
6. USART_ICRレジスタのTCCFビットをセットすることによって、USART_ISRレジスタのTCフラグをクリアします。
7. DMAレジスタのチャネルを有効にします。

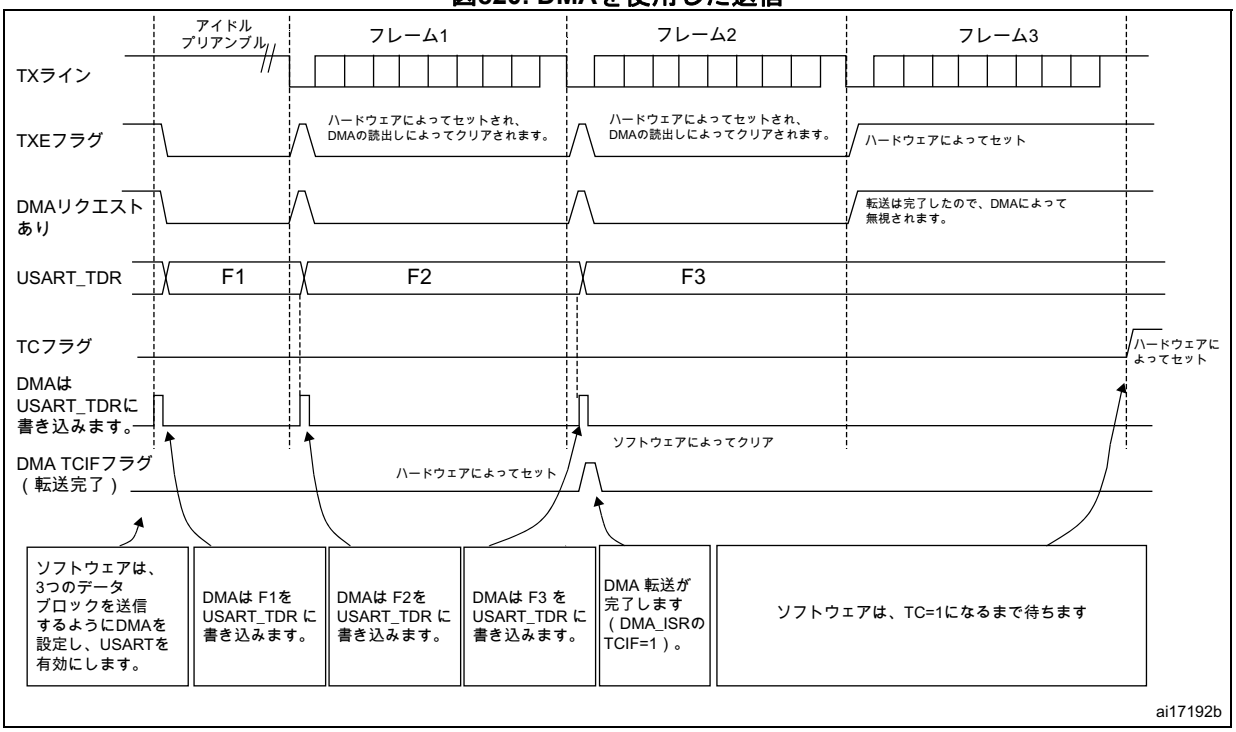
DMAコントローラにプログラミングされたデータ転送数に達すると、DMAコントローラは、DMAチャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータをDMAが書き込むと（DMA_ISRレジスタのTCIFフラグがセットされます）、TCフラグを観察してUSART通信の完了を確認することができます。これは、USARTを無効にする前に、またはペリフェラルクロックが無効のときシステムが低消費電力モードに

入る前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1になるまで待つ必要があります。TCフラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

注： DMAの転送終了前にDMATビットをクリアしないでください。

図820. DMAを使用した送信



注： FIFO管理が有効になっているときは、DMAリクエストは、送信FIFOノットフル（すなわち、TXFNF = 1）によってトリガされます。

DMAを使用した受信

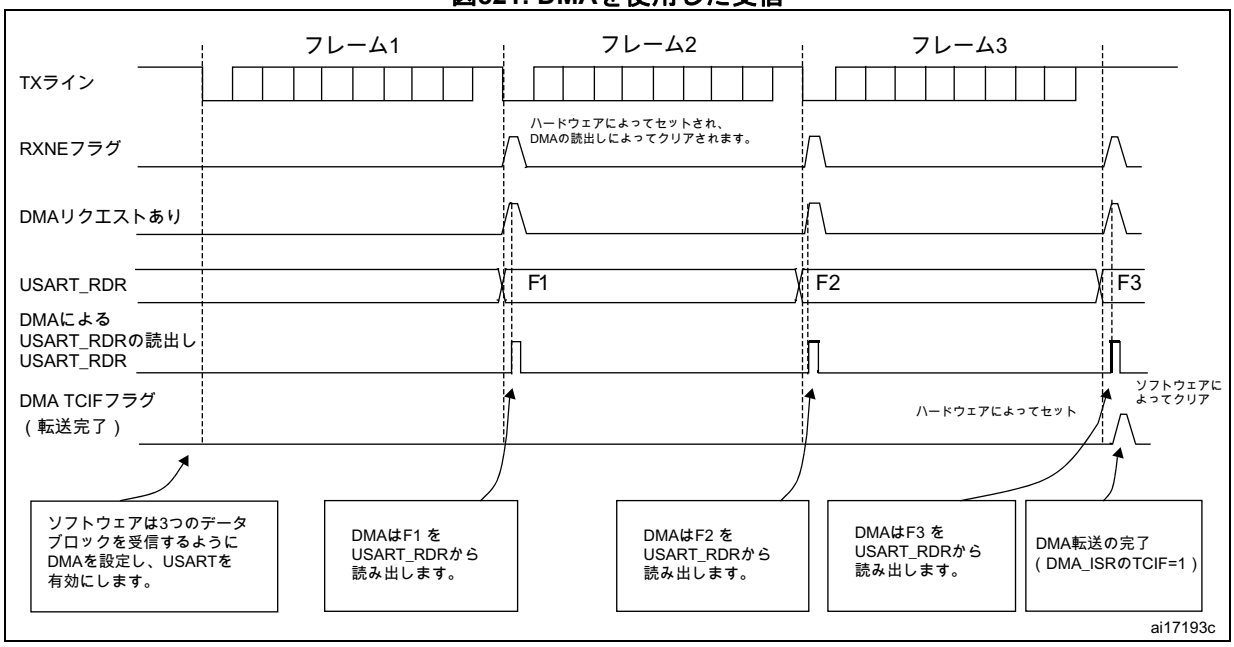
DMAモードでの受信を有効にするには、USART_CR3レジスタのDMARビットをセットします。データバイトが受信されるたびに、データは、USART_RDRレジスタからDMAペリフェラル（ダイレクトメモリアクセスコントローラ*(DMA)*セクション）を使用して設定されたSRAM領域にロードされます。DMAチャネルをUSART受信用に割り付けるには、次の手順を実行します。

1. DMA制御レジスタにUSART_RDRレジスタのアドレスを書き込み、これを転送元として設定します。データは、各RXNE（FIFOモードが有効な場合はRXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各RXNE（FIFOモードが有効な場合はRXFNE）イベント後に、USART_RDRからこのメモリ領域にロードされます。
3. 転送すべきバイト総数をDMA制御レジスタに設定します。
4. チャネル優先順位をDMA制御レジスタで設定します。
5. アプリケーションで必要とされる1/2転送終了、転送完了後の割込み生成を設定します。
6. DMA制御レジスタのチャネルを有効にします。

DMAコントローラにプログラミングされたデータ転送数に達すると、DMAコントローラは、DMAチャネルの割込みベクタに基づいて割込みを生成します。

注：　DMAの転送終了前にDMARビットをクリアしないでください。

図821. DMAを使用した受信



注：　FIFO管理が有効になっているときは、DMAリクエストは、受信FIFOノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。

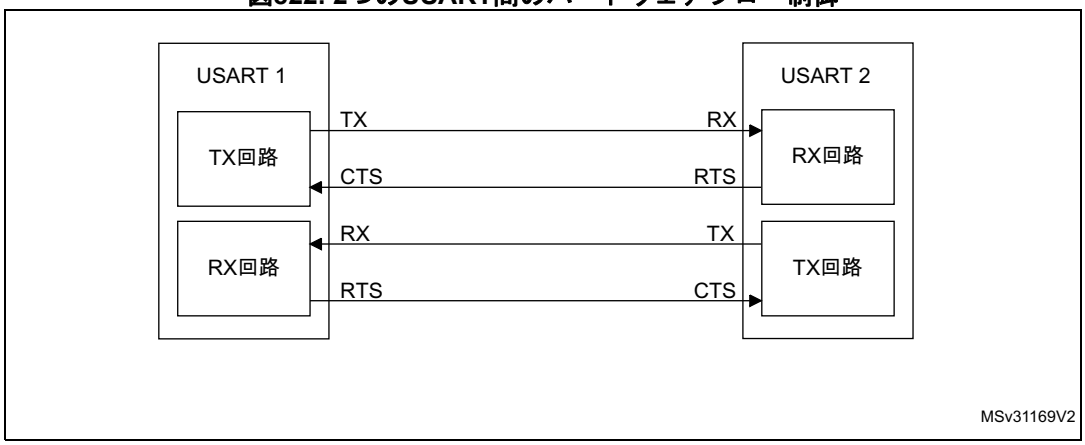
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1バイト受信においてRXNE（FIFOモードが有効な場合はRXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（USART_CR3レジスタのEIEビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

66.5.21 RS232ハードウェアフロー制御およびRS485ドライバ有効

CTS入力とRTS出力を使用すると、2つのデバイス間でシリアルデータフローを制御できます。図822に、このモードで2つのデバイスを接続する方法を示します。

図822. 2つのUSART間のハードウェアフロー制御

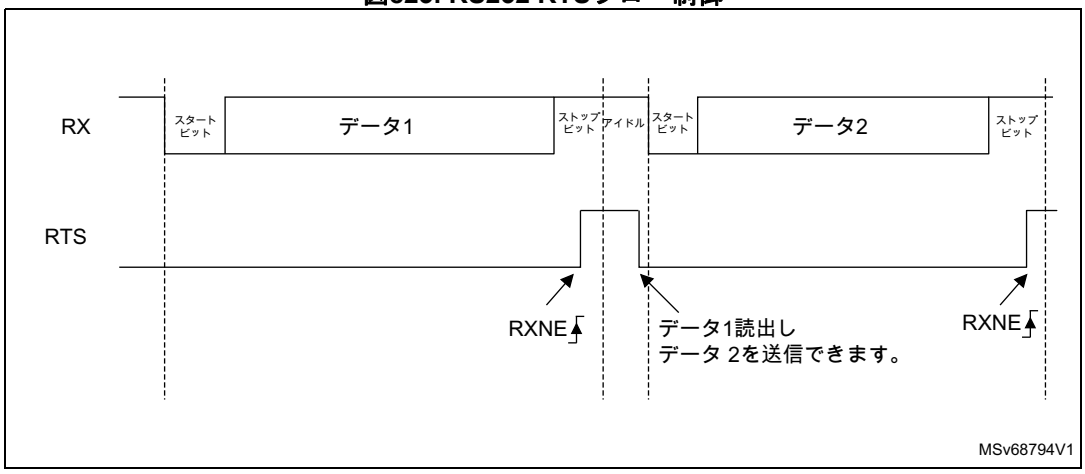


RS232 RTSとCTSのフロー制御は、USART_CR3レジスタのRTSEビットとCTSEビットに “1” を書き込むことによって、個別に有効にできます。

RS232 RTSフロー制御

RTSフロー制御が有効な場合（RTSE=1）、USARTレシーバが新しいデータを受信可能である限り、RTSがネゲートされます（ローレベル接続）。受信レジスタがフルになるとRTSがアサートされ、現在のフレームの終わりに送信が停止する予定であることを示します。図823に、RTSフロー制御が有効な場合の通信例を示します。

図823. RS232 RTSフロー制御



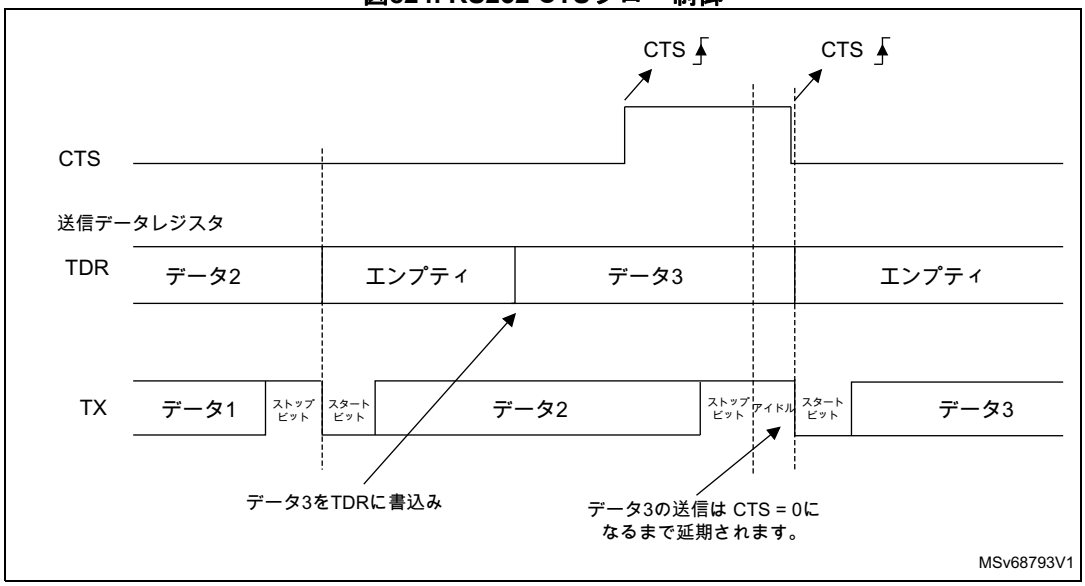
注： **FIFOモードが有効な場合は、RXFIFOがフルのときにのみ、RTSがアサートされます。**

RS232 CTSフロー制御

CTSフロー制御が有効な場合（CTSE=1）、トランスミッタは、CTS入力をチェックしてから、次のフレームを送信します。CTSがネゲートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまりTXE/TXFE=0の場合）。そうでない場合、送信は行われません。送信中にCTSがアサートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1の場合、CTS入力がトグルすると、CTSIFステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。USART_CR3レジスタのCTSIEビットがセットされている場合、割込みが生成されます。図824に、CTSフロー制御が有効な場合の通信例を示します。

図824. RS232 CTSフロー制御



注： **正しい動作のために、CTSは、現在のキャラクタの終了の少なくとも3 USARTクロックソース周期前にネゲートする必要があります。さらに、2 x PCLK周期より短いパルスではCTSCFフラグがセットされない場合があることに注意してください。**

RS485ドライバ有効

ドライバ有効機能を有効にするには、USART_CR3制御レジスタのビットDEMをセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE信号の有効化から スタートビットの開始までの時間です。USART_CR1制御レジスタのDEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了からDE信号の無効化までの時間です。USART_CR1制御レジスタのDEDT [4:0] ビットフィールドを使用してプログラムされます。DE信号の極性は、USART_CR3制御レジスタのDEPビットを使用して設定できます。

USARTでは、DEATおよびDEDTはサンプル時間単位（オーバーサンプリングレートに応じて1/8または1/16bit時間）で表されます。

66.5.22 USART自律モード

USARTペリフェラルは、自律モードによってSTOPモードでも機能します。自律モードは、RUNモードおよびSLEEPモードでも使用できます。低電力モードに入る前にUESMビットをセットする必要があります。

APBクロックは、USARTステータスを更新する必要があるたびにペリフェラルによって要求されます。USARTはAPBクロックを受信すると、ペリフェラルの設定に応じて割込みまたはDMAリクエストを生成します。

割込みが生成された場合、デバイスはSTOPモードからウェイクアップします。割込みが生成されない場合、デバイスはSTOPモードのままですが、カーネルクロックとAPBクロックは、リセットおよびクロックコントローラ（RCC）で有効になっているUSARTおよびすべての自律ペリフェラルで引き続き使用可能です。DMAリクエストが有効になっている場合は、製品がSTOPモードに留まっている間、データはDMAによってSRAMとの間で直接転送されます。

送信モード

送信時、APBクロックはTEビットがセットされていて以下の場合にのみ要求されます。

- FIFOモードが有効な場合は、次の場合にAPBクロックが要求されます。
  - TxFIFOが空で（TXFE = 1）、対応する割込みが有効になっている（TXFEIE = 1）
  - TxFIFOの閾値に達していて（TXFT = 1）、対応する割込みが有効になっている（TXFTIE = 1）
  - TxFIFOがフルでなく（TXFNF = 1）、対応する割込みまたはDMAが有効になっている（TXFNFIE = 1またはDMAT = 1）
- FIFOモードが無効の場合、データがシフトレジスタに転送されるとすぐにAPBクロックが要求されます。DMAまたは関連する割込みを有効にしておく必要があります。

TEビットは、非同期トリガが検出されると、ハードウェアによってセットされます。

送信は、RUN、SLEEP、またはSTOPモードにおいて、非同期トリガが検出されると自動的に開始されます。トリガは、USART_AUTOCR レジスタの TRIGSEL ビットで選択します。これにより、USART_CR1レジスタのTEビットがセットされ、転送を有効にするAPBクロック要求が生成されます。送信が完了するまでAPBクロックが要求され、プログラムされた送信データ数（USART_AUTOCRレジスタのTDNビットフィールド）に達するとハードウェアによってTEビットがクリアされます。この場合、送信すべきデータ数に達し、最後のバイトを送信した時点でTCフラグがセットされます。

受信モード

- FIFOモードが有効な場合は、次の場合にAPBクロックが要求されます。
  - RxFIFIOがフルで（RXFF = 1）、対応する割込みが有効になっている（RXFFIE = 1）
  - RxFIFIOの閾値に達していて（RXFT = 1）、対応する割込みが有効になっている（RXFTIE = 1）
  - RxFIFIOが空でなく（RXFNE = 1）、対応する割込みまたはDMAが有効になっている（RXFNEIE = 1またはDMAR = 1）
- FIFOモードが無効な場合、APBクロックはUSARTがデータのサンプリングを終了し、USART_RDRに書き込む準備ができた時に要求されます。DMAまたは関連する割込みを有効にしておく必要があります。

注： **受信モードでは、オーバーランエラーが発生する（ORE = 1）と、APBクロックが要求されます。割込みの発生とMCUのウェイクアップを有効にするには、EIEビットを設定し、OVRDISビットをクリ**

アした状態にする必要があります。APBクロックリクエストは、割込みフラグがクリアされるまで、維持されます。

受信モードでは、パリティ/ノイズ/フレーミングエラーが発生して、DMAが受信に使用されるときにもAPBクロックが要求されます。APBクロックリクエストは、割込みフラグがクリアされるまで、維持されます。

UARTおよびSPIマスタモードのみで自律モードがサポートされます。

マイクロコントローラの低消費電力モードからの正常なウェイクアップを可能にする最大USARTボーレートの決定

マイクロコントローラの低消費電力モードからの正常なウェイクアップを可能にする最大USARTボーレートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）とUSARTレシーバの許容誤差（セクション 66.5.9：クロック偏差に対するUSARTレシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、Mビット = 01、ONEBIT = 0、BRR [3:0] = 0000とします。

この条件では、表 672：BRR [3:0] = 0000のときのUSARTレシーバの許容誤差によると、USARTレシーバの許容誤差は3.41% です。

DTRA + DQUANT + DREC + DTCL + DWU < USARTレシーバの許容誤差

$D_{WUmax} = t_{WUUSART} / (11 \times T_{bitmin})$

$T_{bitmin} = t_{WUUSART} / (11 \times D_{WUmax})$

ここでは、$t_{WUUSART}$ が低消費電力モードからのウェイクアップ時間です。

パラメータDTRA、DQUANT、DREC、およびDTCLが0% であるときに理想的なケースを考えた場合、DWUの最大値は3.41% です。実際には、最低usart_ker_ck精度を考慮する必要があります(DREC)。

たとえばHSIがusart_ker_ckとして使用され、HSI精度が1% の場合、以下を得ることができます。

$t_{WUUSART}$ = 3µs（この値は１例です。正確な値については、デバイスのデータシートを参照してください）

D_{WUmax} = USARTレシーバの許容誤差 – DREC = 3.41% - 1% = 2.41%

T_{bitmin} = 3µs/ (11 x 2.41%) = 11.32µs.

その結果、低電力モードからの正しいウェイクアップを可能にする最大ボーレートは、1/11.32µs = 88.36kbaudとなります。

66.6 低消費電力モードのUSART

表675.低消費電力モードがUSARTに与える影響

モード	説明
SLEEP	影響はありません。USART割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	USARTレジスタの内容は保持されます。 USARTがSTOPモードで利用可能なオシレータによってクロック供給されている場合、非同期モードおよびSPIマスタモードでの転送は機能します。DMAリクエストは機能しており、割込みによってデバイスはSTOPモードを終了します。
STANDBY	USARTペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. STOPモードからのウェイクアップが指定のペリフェラルインスタンスでサポートされているかどうかについては、セクション 66.4 : USARTの実装を参照してください。インスタンスが指定のSTOPモードで機能しない場合、このSTOPモードを開始する前に無効化する必要があります。

66.7 USART割込み

すべてのUSART割込みリクエストの詳細な説明については、表676を参照してください。

表676.USART割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOP(1)モードの終了	STANDBYモードの終了
USARTまたはUART	送信データレジスタエンプティ	TXE	TXEIE	TDRを書き込む	可能	あり	なし
	送信FIFOノットフル	TXFNF	TXFNFIE	TXFIFOフル		あり	
	送信FIFOエンプティ	TXFE	TXFEIE	TXFRQにTDRまたは1を書き込む		可能	
	送信FIFO閾値到達	TXFT	TXFTIE	TDRを書き込む		可能	
	CTS割込み	CTSIF	CTSIE	CTSCFに1を書き込む		不可	
	送信完了	TC	TCIE	TCCFにTDRまたは1を書き込む		可能	
	ガード時間前に送信完了	TCBGT	TCBGTIE	TCBGTにTDRまたは1を書き込む		不可	
USARTまたはUART	受信データレジスタノットエンプティ（データの読出し可能）	RXNE	RXNEIE	RXFRQでRDRまたは1を読み出す	可能	あり	なし
	受信FIFOノットエンプティ	RXFNE	RXFNEIE	RXFIFOエンプティまでRDRを読み出すか、RXFRQに1を書き込む		可能	
	受信FIFOフル	RXFF(2)	RXFFIE	読出しはRDRです。		可能	
	受信FIFO閾値到達	RXFT	RXFTIE	読出しはRDRです。		可能	
	オーバーランエラー検出	ORE	RXNEIE/RXFNEIE	ORECFに1を書き込む		**あり**	
	アイドルライン検出	IDLE	IDLEIE	IDLECFに1を書き込む		不可	
	パリティエラー	PE	PEIE	PECFに1を書き込む		**あり**(3)	
	LINブレーク	LBDF	LBDIE	LBDCFに1を書き込む		**不可**	
	マルチバッファ通信でノイズエラー。	NE	EIE	NFCFに1を書き込む		**あり** (3)	
	マルチバッファ通信でオーバーランエラー。	ORE(4)		ORECFに1を書き込む		**あり**	
	マルチバッファ通信でフレーミングエラー。	FE		FECFに1を書き込む		**あり** (3)	
	キャラクター致	CMF	CMIE	CMCFに1を書き込む		**あり**(5)	
	レシーバタイムアウト	RTOF	RTOFIE	RTOCCFに1を書き込む		**不可**	
	ブロックの終了	EOBF	EOBIE	EOBCFに1を書き込む		**不可**	
	SPI スレーブのアンダーランエラー	UDR	EIE	UDRCFに1を書き込む		**不可**	

1. USARTは、ペリフェラルインスタンスがSTOPモードからのウェイクアップ機能をサポートしている場合にのみデバイスをSTOPモードからウェイクアップすることができます。サポートされているSTOPモードのリストについては、セクション 66.4 : USARTの実装を参照してください。

2. RXFFフラグは、USARTがn+1個のデータ、すなわち、RXFIFOにn個のデータ、USART_RDRに1個のデータを受信した場合にアサートされます（nはRXFIFOのサイズ）。STOPモードでは、USART_RDRはクロック供給されません。その結果として、このレジスタは書き込まれず、n個のデータが受信されてRXFIFOに書き込まれた後に、RXFF割込みがアサートされます（RXFFフラグはセットされません）。
3. パリティ / ノイズ / フレーミング・エラー割込みが発生すると、DMAが使用されている場合にSTOPモードからのウェイクアップが可能になります。
4. OVRDIS = 0の場合。
5. FIFOモードが有効な場合、DMAを使用する必要があります。

66.8 USARTレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスする必要があります。

66.8.1 USART制御レジスタ1（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタがFIFOモード有効（このセクション）でも、FIFOモード無効（次のセクション）でも使用できます。

FIFOモードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **RXFFIE**：RXFIFOフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのRXFF = 1のときに、USART割込みが生成されます。

ビット30 **TXFEIE**：TXFIFOエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのTXFE = 1のときに、USART割込みが生成されます。

ビット29 **FIFOEN**：FIFOモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFOモードは無効です。

1：FIFOモードは有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： FIFOモードは、SPIマスタ／スレーブモードおよびスマートカードモードでのみ、標準UART通信で使用できます。IrDAおよびLINのモードでは有効にしてはいけません。

ビット28 **M1**：ワード長

このビットはビット12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00：スタートビット1個、データビット8個、ストップビットn個

M[1:0] = 01：スタートビット1個、データビット9個、ストップビットn個

M[1:0] = 10：スタートビット1個、データビット7個、ストップビットn個

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitデータ長モードでは、スマートカードモード、LINマスタモード、および自動ボーレート（0x7Fおよび0x55フレーム検出）はサポートされません。

ビット27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのEOBFフラグがセットされると、USART割込みが生成されます。

注： USARTがスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット26 **RTOIE**：レシーバタイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのRTOFビットがセットされると、USART割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4: 2731ページのUSARTの実装を参照してください。

ビット25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この5bit値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8または1/16bit時間）で表されます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この5bit値は、送信メッセージの最後のストップビットの終了からDE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8または1/16bit時間）で表されます。

DEDT時間中にUSART_TDRレジスタに書込みが行われた場合、DEDT時間とDEAT時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット15 **OVER8**：オーバーサンプリングモード

0：16倍のオーバーサンプリング

1：8倍のオーバーサンプリング

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。

ビット14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのCMFビットがセットされると、USART割込みが生成されます。

ビット13 **MME**：ミュートモード有効

このビットはUSARTミュートモード機能を有効にします。セットされると、USARTは、WAKEビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット12 **M0**：ワード長

このビットはビット28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット28（M1）の説明を参照）。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

ビット11 **WAKE**：レシーバのウェイクアップ方式

このビットによって、ミュートモードからのUSARTのウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティはMSB位置（M=1の場合は9番目のビット、M=0の場合は8番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後でPCEが有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCEビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット8 **PEIE**：PE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのPE=1のときには、USART割込みが生成されます。

ビット7 **TXFNFIE**：TXFIFOノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのTXFNF=1のときには、USART割込みが生成されます。

ビット6 **TCIE**：送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのTC=1のときには、USART割込みが生成されます。

ビット5 **RXFNEIE**：RXFIFOノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのOREが1またはRXFNEが1になるとUSART割込みが生成されます。

ビット4 **IDLEIE**：IDLE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのIDLE=1のときには、USART割込みが生成されます。

ビット3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。自律モードが使用されていない場合、TEビットは、ソフトウェアによってセット／クリアされます。自律モードが使用されている場合、TEビットはステータスビットになり、ハードウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0” に続けて “1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐにはTEに1を書き込まないでください。必要な時間を確保するために、ソフトウェアはUSART_ISRレジスタのTEACKビットをポーリングできます。

スマートカードモードでは、TEがセットされると、送信が開始されるまでに1bit時間の遅れが生じます。

ビット2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット1 **UESM**：USART低電力モードで有効

このビットがクリアされると、USARTは そのカーネルクロックを要求できず、低電力モードでは機能しません。

このビットがセットされると、USARTはMCUを 低消費電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USARTは低電力モードで機能しません。

1：USARTは低電力モードで機能します。

注： UESMビットは初期化フェーズでセットする必要があります。

USARTが 低電力モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4：2731ページのUSARTの実装を参照してください。

ビット0 **UE**：USARTイネーブル

このビットがクリアされると、USARTプリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USARTの設定は保たれますが、USART_ISRのステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USARTプリスケーラと出力は無効であり、低電力モードです。

1：USARTは有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには、TE ビットを事前にリセットする必要があり、ソフトウェアはUSART_ISRのTCビットがセットされるのを待ってから、UEビットをリセットする必要があります。

UE=0のときにはDMAリクエストもリセットされるので、UEビットをリセットする前にDMAチャネルを無効にする必要があります。

スマートカードモードでは（SCEN = 1）、CKはUEビット値にかかわらず、CLKEN = 1の場合に常に使用可能です。

66.8.2 USART制御レジスタ1[オルタネート]（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタがFIFOモード有効（前のセクション）でも、FIFOモード無効（このセクション）でも使用できます。

FIFOモードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **FIFOEN**：FIFOモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFOモードは無効です。

1：FIFOモードは有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： FIFOモードは、SPIマスタ／スレーブモードおよびスマートカードモードでのみ、標準UART通信で使用できます。IrDAおよびLINのモードでは有効にしてはいけません。

ビット28 **M1**：ワード長

このビットはビット12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00：スタートビット1個、データビット8個、ストップビットn個

M[1:0] = 01：スタートビット1個、データビット9個、ストップビットn個

M[1:0] = 10：スタートビット1個、データビット7個、ストップビットn個

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitデータ長モードでは、スマートカードモード、LINマスタモード、および自動ボーレート（0x7Fおよび0x55フレーム検出）はサポートされません。

ビット27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのEOBFフラグがセットされると、USART割込みが生成されます。

注： USARTがスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット26 **RTOIE**：レシーバタイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのRTOFビットがセットされると、USART割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この5bit値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8または1/16bit時間）で表されます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この5bit値は、送信メッセージの最後のストップビットの終了からDE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8または1/16bit時間）で表されます。

DEDT時間中にUSART_TDRレジスタに書込みが行われた場合、DEDT時間とDEAT時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット15 **OVER8**：オーバーサンプリングモード

0：16倍のオーバーサンプリング

1：8倍のオーバーサンプリング

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。

ビット14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのCMFビットがセットされると、USART割込みが生成されます。

ビット13 **MME**：ミュートモード有効

このビットはUSARTミュートモード機能を有効にします。セットされると、USARTは、WAKEビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット12 **M0**：ワード長

このビットはビット28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット28（M1）の説明を参照）。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

ビット11 **WAKE**：レシーバのウェイクアップ方式

このビットによって、ミュートモードからのUSARTのウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティはMSB位置（M=1の場合は9番目のビット、M=0の場合は8番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後でPCEが有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCEビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット8 **PEIE**：PE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのPE=1のときには、USART割込みが生成されます。

ビット7 **TXEIE**：送信データレジスタエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのTXE=1のときには、USART割込みが生成されます。

ビット6 **TCIE**：送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのTC=1のときには、USART割込みが生成されます。

ビット5 **RXNEIE**：受信データレジスタノットエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのOREが1またはRXNEが1になるとUSART割込みが生成されます。

ビット4 **IDLEIE**：IDLE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISRレジスタのIDLE=1のときには、USART割込みが生成されます。

ビット3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。自律モードが使用されていない場合、TEビットは、ソフトウェアによってセット／クリアされます。自律モードが使用されている場合、TEビットはステータスビットになり、ハードウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0” に続けて “1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐにはTEに1を書き込まないでください。必要な時間を確保するために、ソフトウェアはUSART_ISRレジスタのTEACKビットをポーリングできます。

スマートカードモードでは、TEがセットされると、送信が開始されるまでに1bit時間の遅れが生じます。

ビット2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット1 **UESM**：USART低電力モードで有効

このビットがクリアされると、USARTは そのカーネルクロックを要求できず、低電力モードでは機能しません。

このビットがセットされると、USARTはMCUを 低消費電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USARTは低電力モードで機能しません。

1：USARTは低電力モードで機能します。

注： UESMビットは初期化フェーズでセットする必要があります。

USARTが 低電力モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット0 **UE**：USARTイネーブル

このビットがクリアされると、USARTプリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USARTの設定は保たれますが、USART_ISRのステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USARTプリスケーラと出力は無効であり、低電力モードです。

1：USARTは有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには、TE ビットを事前にリセットする必要があり、ソフトウェアはUSART_ISRのTCビットがセットされるのを待ってから、UEビットをリセットする必要があります。

UE=0のときにはDMAリクエストもリセットされるので、UEビットをリセットする前にDMAチャネルを無効にする必要があります。

スマートカードモードでは（SCEN = 1）、CKはUEビット値にかかわらず、CLKEN = 1の場合に常に使用可能です。

66.8.3 USART制御レジスタ2（USART_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw			rw

ビット31:24 **ADD[7:0]**：USARTノードのアドレス

これらのビットは、ミュートモードのUSARTノードのアドレス、または低消費電力モードまたはRUNモードで認識されるキャラクタコードを指定します。

- ミュートモードでは、マルチプロセッサ通信で4bit／7bitアドレスマーク検出によってミュートモードからウェイクアップするために使用されます。トランスミッタによって送信されるキャラクタのMSBは1でなければなりません。4bitアドレスマーク検出では、ADD[3:0] ビットのみが使用されます。
- 低消費電力モードでは、キャラクター致時に低消費電力モードからのウェイクアップに使用されます。低電力モード中に受信したキャラクタがADD[7:0] ビットフィールドでプログラムされたキャラクタと一致する場合、CMFフラグがセットされ、CMIEビットをセットすることで対応する割込みが有効になっていれば、低電力モードからデバイスがウェイクアップします。
- ミュートモードが無効化された RUN モードでは（ModBus プロトコルでのブロックの終了の検出など）、受信したキャラクタ全体（8bit）がADD[7:0] 値と比較され、一致している場合はCMFフラグがセットされます。CMIEビットがセットされている場合、割込みが生成されます。

これらのビットフィールドは、受信が無効のとき（RE=0）またはUSARTが無効のとき（UE=0）のみ、書き込むことができます。

ビット23 **RTOEN**：レシーバタイムアウト有効

このビットは、ソフトウェアによってセット／クリアされます。

0：レシーバタイムアウト機能は無効です。

1：レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR（レシーバタイムアウトレジスタ）でプログラムされた時間にわたってRXラインがアイドル（受信なし）であった場合、USART_ISRレジスタのRTOFフラグがセットされます。

注： USARTがレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット22:21 **ABRMOD[1:0]**：自動ボーレートモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：スタートビットの測定がボーレートの検出に使用されます。

01：立ち下がりエッジから立ち下がりエッジまでの測定（受信されたフレームはシングルビット = 1で始まらなければならず、その場合、フレーム = Start10xxxxxx）

10：0x7Fフレーム検出

11：0x55フレーム検出

このビットフィールドは、ABREN=0またはUSARTが無効（UE=0）のときのみ書き込むことができます。

注： DATAINV=1および／またはMSBFIRST=1の場合、パターンはライン上で同じである必要があります（たとえば、MSBFIRSTの場合は0xAA）。

USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット20 **ABREN**：自動ボーレート有効

このビットは、ソフトウェアによってセット／クリアされます。

0：自動ボーレート検出は無効です。

1：自動ボーレート検出は有効です。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット19 **MSBFIRST**：MSBファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット0から順に送受信されます。

1：スタートビットに続いて、データはMSB（ビット7/8）から順に送受信されます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット17 **TXINV**：TXピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TXピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TXピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TXラインで外部インバータを使用できます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット16 **RXINV**：RXピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RXピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RXピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RXラインで外部インバータを使用できます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット15 **SWAP**：TX/RXピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RXピンは標準ピンアウトでの定義に従って使用されます。

1：TXおよびRXピンの機能はスワップされます。これにより、別のUARTへのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット14 **LINEN**：LINモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LINモードは無効です。

1：LINモードは有効です。

LINモードでは、USART_CR1レジスタのSBKRQビットを使用してLIN同期ブレーク（下位13bit）を送信し、LIN同期ブレークを検出することができます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： USARTがLINモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1個のストップビット

01：0.5個のストップビット

10：2個のストップビット

11：1.5個のストップビット

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

ビット11 **CLKEN**：クロック有効

このビットによって、CKピンを有効にできます。

0：CKピンは無効です。

1：CKピンは有効です。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　同期モードもスマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

スマートカードモードでは、スマートカードにCKクロックを正しく供給するには、次のステップを順守する必要があります。
UE = 0
SCEN = 1
GTPR設定
CLKEN= 1
UE = 1

ビット10 **CPOL**：クロック極性

このビットによって、同期モードにおけるCKピンのクロック出力の極性を選択できます。CPHAビットと連携して動作し、希望するクロック／データ関係になるようにします。

0：送信ウィンドウの外で、CKピンはローレベルを維持します。

1：送信ウィンドウの外で、CKピンはハイレベルを維持します。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット9 **CPHA**：クロックフェーズ

このビットは、同期モードでのCKピンのクロック出力の位相を選択するために使用されます。CPOLビットと連携して動作し、希望するクロック／データ関係になるようにします（図806および図807を参照）。

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット8 **LBCL**：最終ビットのクロックパルス

このビットは、同期モードで送信される最終データビット（MSB）に関連するクロックパルスを、CKピンに出力する必要があるかどうかを選択するために使用されます。

0：最終データビットのクロックパルスはCKピンに出力されません。

1：最終データビットのクロックパルスがCKピンに出力されます。

注意：　最終ビットは、USART_CR1レジスタのMビットによって選択された7または8または9bitフォーマットに応じて送信された7番目または8番目または9番目のデータビットです。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LBDIE**：LINブレーク検出割込み有効

ブレーク割込みマスクです（ブレークデリミタを使用したブレーク検出）。

0：割込みは禁止されています。

1：USART_ISRレジスタのLBDF=1のときには、割込みが生成されます。

注： LINモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット5 **LBDL**：LIN ブレーク検出長

このビットでは、10bitと11bitのブレーク検出を選択します。

0：10bitブレーク検出

1：11bitブレーク検出

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： LINモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット4 **ADDM7**：7bitアドレス検出／4bitアドレス検出

このビットは、4bitアドレス検出と7bitアドレス検出の選択に使用されます。

0：4bitアドレス検出

1：7bitアドレス検出（8bitデータモード）

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitおよび9bitデータモードでは、アドレス検出は、それぞれ6bitおよび8bitアドレス（ADD[5:0] およびADD[7:0]）に対して行われます。

ビット3 **DIS_NSS**：

DIS_NSSビットがセットされているとき、NSSピンの入力は無視されます。

0：SPIスレーブ選択はNSS入力ピンに依存します。

1：SPIスレーブが常に選択され、NSS入力ピンは無視されます。

注： SPIスレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット2:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SLVEN**：同期スレーブモード有効

SLVENビットがセットされると、同期スレーブモードが有効になります。

0：スレーブモードは無効です。

1：スレーブモードは有効です。

注： SPIスレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

注： トランスミッタが有効なときには、CPOL、CPHA、LBCLのビットに書き込まないでください。

66.8.4 USART制御レジスタ3（USART_CR3）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXF TIE	RXFTCFG[2:0]			TCBG TIE	TXFTIE	Res.	Res.	Res.	SCARCNT[2:0]			Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	ONE BIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HD SEL	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 **TXFTCFG[2:0]**：TXFIFO閾値設定

000：TXFIFOはその深さの1/8に達します。
001：TXFIFOはその深さの1/4に達します。
010：TXFIFOはその深さの1/2に達します。
011：TXFIFOはその深さの3/4に達します。
100：TXFIFOはその深さの7/8に達します。
101：TXFIFOは空になります。
残りの組み合わせ：予約済み

ビット28 **RXFTIE**：RXFIFO閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信FIFOがRXFTCFGでプログラムされた閾値に達すると、USART割込みが生成されます。

ビット27:25 **RXFTCFG[2:0]**：受信FIFO閾値設定

000：受信FIFOはその深さの1/8に達します。
001：受信FIFOはその深さの1/4に達します。
010：受信FIFOはその深さの1/2に達します。
011：受信FIFOはその深さの3/4に達します。
100：受信FIFOはその深さの7/8に達します。
101：受信FIFOはフルになります。
残りの組み合わせ：予約済み

ビット24 **TCBGTIE**：ガード時間前送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：USART_ISRレジスタのTCBGT=1のときには、USART割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4：2731ページのUSARTの実装を参照してください。

ビット23 **TXFTIE**：TXFIFOフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFOがTXFTCFGでプログラムされた閾値に達すると、USART割込みが生成されます。

ビット22:20 予約済みであり、リセット値に保持する必要があります。

ビット19:17 **SCARCNT[2:0]**：スマートカード自動再試行カウント

このビットフィールドは、スマートカードモードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します（FEビットをセット）。

受信モードでは、受信エラーが生成される（RXNE/RXFNEおよびPEビットのセット）までの受信の試行エラー回数を指定します。

このビットフィールドは、USARTが無効（UE=0）のときのみプログラムする必要があります。

USARTが有効になると（UE=1）、このビットフィールドは再送信を停止するために0x0にのみ書込み可能です。

0x0：再送信無効 - 送信モードでの自動再送信禁止

0x1から0x7：自動再送信試行回数（信号エラーの生成前）

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **DEP**：ドライバ有効極性選択

0：DE信号はアクティブハイです。

1：DE信号はアクティブローです。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット14 **DEM**：ドライバ有効モード

このビットにより、DE信号によって外部トランシーバ制御を有効にできます。

0：DE機能は無効です。

1：DE機能は有効です。DE信号はRTSピンで出力されます。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット13 **DDRE**：受信エラー時DMA無効

0：受信エラーの場合、DMAは無効になりません。対応するエラーフラグはセットされますが、RXNEは0に保たれ、オーバーランを防ぎます。結果として、DMAリクエストはアサートされないので、エラーのあるデータは転送されず（DMAリクエストなし）、次の正しい受信データが転送されます。（スマートカードモードで使用されます）

1：受信エラーの後、DMAは無効化されます。対応するエラーフラグとRXNEがセットされます。エラーフラグがクリアされるまで、DMAリクエストはマスクされます。つまり、ソフトウェアは、まず、DMAリクエストを無効にするか（DMAR=0）、RXNE（FIFOモードが有効な場合はRXFNE）をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、OREは、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNEフラグがまだセットされている間に新しいデータを受信した場合、

OREフラグはセットされず、新しく受信されたデータがUSART_RDRレジスタの前の内容に上書きされます。FIFOモードが有効なとき、RXFIFOはバイパスされ、データは直接、USART_RDRレジスタに書き込まれます。FIFO管理が有効になっているときでも、RXNEフラグが使用されます。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット11 **ONEBIT**：1サンプルビット方式有効

このビットによって、サンプル方式を選択できます。1サンプルビット方式が選択されると、ノイズ検出フラグ（NE）が無効になります。

0：3サンプルビット方式

1：1サンプルビット方式

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

ビット10 **CTSIE**：CTS割込み有効

0：割込みは禁止されています。

1：USART_ISRレジスタのCTSIF=1のときには、割込みが生成されます。

注：　ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット9 **CTSE**：CTS有効

0：CTSハードウェアフロー制御は無効です。

1：CTSモードは有効です。データはCTS入力がネゲート（0に固定）されている場合にのみ転送されます。データの送信中にCTS入力がアサートされた場合、送信は停止前に完了します。CTSがアサートされている間にデータがデータレジスタに書き込まれた場合、CTSがネゲートされるまで送信は延期されます。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット8 **RTSE**：RTS有効

0：RTSハードウェアフロー制御は無効です。

1：RTS出力は有効です。データは受信バッファに空きがある場合にのみリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS出力がネゲートされます（0にプルされます）。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注：　ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7 **DMAT**：DMA有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMAモードは送信に有効です。

0：DMAモードは送信に無効です。

ビット6 **DMAR**：DMA有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMAモードは受信に有効です。

0：DMAモードは受信に無効です。

ビット5 **SCEN**：スマートカードモード有効

このビットはスマートカードモードを有効にするために使用します。

0：スマートカードモードは無効です。

1：スマートカードモードは有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注：　USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット4 **NACK**：スマートカードNACK有効

0：パリティエラーの際のNACK転送は無効です。

1：パリティエラー時のNACK転送は有効です。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

ビット2 **IRLP**：IrDA低電力

このビットは、通常と低消費電力のIrDAモードの選択に使用されます。

0：通常モード

1：低電力モード

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： IrDAモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット1 **IREN**：IrDAモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：IrDAは無効です。

1：IrDAは有効です。

このビットは、USARTが無効（UE=0）のときのみ書き込むことができます。

注： IrDAモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット0 **EIE**：エラー割込み有効

エラー割込み有効ビットは、フレーミングエラー、オーバーランエラー、ノイズフラグ、またはSPIスレーブアンダーランエラー（USART_ISRレジスタのFE=1またはORE=1またはNE=1またはUDR=1）の場合に割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：USART_ISRレジスタのFE=1またはORE=1またはNE=1またはUDR=1（SPIスレーブモードで）のときに、割込みが生成されます。

66.8.5 USARTボーレートレジスタ（USART_BRR）

このレジスタは、USARTが無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **BRR[15:0]**：USARTボーレート

BRR[15:4]

BRR[15:4] はUSARTDIV[15:4] に対応します。

BRR[3:0]

OVER8 = 0のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = 1のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に1bitシフトされます。

BRR[3] は、クリアされたままにする必要があります。

66.8.6 USARTガード時間およびプリスケーラレジスタ（USART_GTPR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								PSC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **GT[7:0]**：ガード時間の値

このビットフィールドは、ガード時間値をボークロック周期数でプログラムするために使用します。

これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7:0 **PSC[7:0]**：プリスケーラ値

条件：IrDA低電力およびIrDA通常モード

PSC[7:0] = IrDA通常および低消費電力ボーレート

このビットフィールドは、USARTクロックソースを分周して低消費電力周波数を得るためのプリスケーラのプログラミングに使用します。

クロックソースは、レジスタに与えられた値（上位8bit）で分周されます。

00000000：予約済み - この値はプログラミングしないでください。

00000001：クロックソースは1で分周されます。

00000010：クロックソースは2で分周されます。

.....

条件：スマートカードモード

PSC[4:0]：プリスケーラ値

このビットフィールドは、USARTクロックソースを分周してスマートカードのクロックを提供するプリスケーラのプログラミングに使用します。

レジスタで指定された値（上位5bit）を2倍して、クロックソース周波数の分周比を求めます。

00000：予約済み - この値はプログラミングしないでください。

00001：クロックソースは2で分周されます。

00010：クロックソースは4で分周されます。

00011：クロックソースは6で分周されます。

.....

このビットフィールドは、USARTが無効（UE = 0）のときのみ書き込むことができます。

注： スマートカードモードが使用される場合、ビット [7:5] はクリアされたままにする必要があります。スマートカードモードやIrDAモードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

66.8.7 USARTレシーバタイムアウトレジスタ（USART_RTOR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLEN[7:0]								RTO[23:16]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 **BLEN[7:0]**：ブロック長

このビットフィールドは、受信時のスマートカードT=1のブロック長を指定します。この値は、情報文字の数 + エピローグフィールドの長さ（1-LEC/2-CRC）- 1と等しくなります。

例：

BLEN = 0 -> 0情報文字 + LEC

BLEN = 1 -> 0情報文字 + CRC

BLEN = 255 -> 254情報文字 + CRC（合計256文字）

スマートカードモードでは、TXE=0（FIFOモードが有効な場合はTXFE =0）のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。この場合、RE=0（受信無効）のとき、および/またはEOBCFビットが1に書き込まれたときに、ブロック長カウンタがリセットされます。

注： この値は、ブロック受信の開始後にプログラムできます（プロローグフィールドのLEN文字のデータを使用）。必ず受信したブロックにつき一度だけプログラムするようにしてください。

ビット23:0 **RTO[23:0]**：レシーバタイムアウト値

このビットフィールドは、レシーバタイムアウト値をビット期間数で指定します。

標準モードでは、最後の受信キャラクタの後、RTO値を超える間、新しいスタートビットが検出されなかった場合、RTOFフラグがセットされます。

スマートカードモードでは、この値はCWTおよびBWTを実装するために使用されます。詳細については、スマートカードの章を参照してください。この標準では、CWT/BWT測定は最後の受信キャラクタのスタートビットから開始して行われます。

注： この値は、受信キャラクタごとにプログラムされる必要があります。

注： RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって0x00000000に強制的に設定されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

66.8.8 USARTリクエストレジスタ（USART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
											w	w	w	w	w

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **TXFRQ**：送信データー掃リクエスト

FIFOモードが無効になっている場合は、このビットに “1” を書き込むと、TXEフラグがセットされます。これにより、送信データを破棄できます。このビットは、エラー（NACK）によりデータが送信されなかった場合、およびUSART_ISRレジスタでFEフラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。USARTがスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

FIFOモードが有効な場合、FIFO全体を一掃するためにTXFRQビットがセットされます。これによってTXFEフラグ（送信FIFOエンプティ、USART_ISRレジスタのビット23）がセットされます。送信FIFOの一掃は、UARTモードおよびスマートカードモードの両方でサポートされています。

注： FIFOモードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFOが空になるまでTXFNFフラグはリセットされます。

ビット3 **RXFRQ**：受信データー掃リクエスト

このビットに1を書き込むと、受信FIFO全体を空にします（すなわち、RXFNEビットをクリアします）。これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット2 **MMRQ**：ミュートモードリクエスト

このビットに1を書き込むと、USARTはミュートモードになり、RWUフラグはリセットされます。

ビット1 **SBKRQ**：ブレーク送信リクエスト

このビットに1を書き込むと、SBKFフラグがセットされ、送信マシンが使用可能になるとすぐに、ラインでBREAKを送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアはSBKRQビットをセットする前に、TXEフラグのアサートを待つ必要があります。

ビット0 **ABRRQ**：自動ボーレートリクエスト

このビットに1を書き込むと、USART_ISRのABRFフラグとABREフラグがリセットされ、次の受信データフレームでの自動ボーレート測定をリクエストします。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4: 2731ページのUSARTの実装を参照してください。

66.8.9 USART割込みおよびステータスレジスタ（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0XX0 00C0

XX = 28（FIFO／スマートカードモードが有効の場合）

XX = 08（FIFOが有効で、スマートカードモードが無効の場合）

同じレジスタがFIFOモード有効（このセクション）でも、FIFOモード無効（次のセクション）でも使用できます。

FIFOモードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY
				r	r	r	r	r	r	r		r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **TXFT**：TXFIFO閾値フラグ

このビットは、TXFIFOがUSART_CR3レジスタのTXFTCFGでプログラムされた閾値に達したとき、すなわち、TXFIFOにTXFTCFGの空き場所ができたときに、ハードウェアによってセットされます。USART_CR3レジスタのTXFTIEビット =1（ビット31）の場合、割込みが生成されます。

0：TXFIFOはプログラムされた閾値に達していません。

1：TXFIFOはプログラムされた閾値に達しました。

ビット26 **RXFT**：RXFIFO閾値フラグ

このビットは、USART_CR3レジスタのRXFTCFGでプログラムされた閾値に達したときに、ハードウェアによってセットされます。これは、受信FIFOに（RXFTCFG - 1）個のデータがあり、USART_RDRレジスタに1個のデータがあることを意味します。USART_CR3レジスタのRXFTIEビット =1（ビット27）の場合、割込みが生成されます。

0：受信FIFOはプログラムされた閾値に達していません。

1：受信FIFOはプログラムされた閾値に達しました。

注： RXFTCFG閾値が101に設定されているとき、16個のデータが利用可能になった場合（すなわち、RXFIFOに15個のデータが入り、USART_RDRに1個のデータが入った場合）、RXFTフラグがセットされます。したがって、17番目の受信データによってオーバーランエラーが起きることはありません。オーバーランエラーは18番目のデータを受信した後に発生します。

ビット25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDRに書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードがNACKを返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3レジスタのTCBGTIE=1の場合、割込みが生成されます。

このビットは、USART_ICRレジスタのTCBGTCFに1を書き込むことによって、またはUSART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信が完了していないか、または送信が正常に完了していません（つまり、カードからNACKを受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードからNACKなし）。

注： USARTがスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USARTがスマートカードモードをサポートする場合でこのモードが有効な場合、TCBGTのリセット値は1です。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット24 **RXFF**：RXFIFOフル

このビットは、受信したデータの数がRXFIFOサイズ + 1と一致したときに（RXFIFOがフルで、USART_RDRレジスタに1個のデータ）、ハードウェアによってセットされます。

USART_CR1レジスタのRXFFIEビット =1の場合、割込みが生成されます。

0：RXFIFOはフルではありません。

1：RXFIFOはフルです。

ビット23 **TXFE**：TXFIFOエンプティ

このビットは、TXFIFOが空のとき、ハードウェアによってセットされます。TXFIFOに少なくとも1データが入ったとき、このフラグはクリアされます。TXFEフラグは、USART_RQRレジスタのビットTXFRQ（ビット4）に1を書き込むことによってセットすることもできます。

USART_CR1レジスタのTXFEIEビット =1（ビット30）の場合、割込みが生成されます。

0：TXFIFOは空ではありません。

1：TXFIFOは空です。

ビット22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値がUSARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、USARTが受信できる状態であることを確認できます。

注： USARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値がUSARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

USART_CR1レジスタでTE=0を書き込んだ後、TE=1を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0の最小周期を満たすために使用できます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **RWU**：レシーバのミュートモードからのウェイクアップ

このビットは、USARTがミュートモードかどうかを示します。ウェイクアップ／ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたはIDLE）は、USART_CR1レジスタのWAKEビットによって選択されます。

IDLEモードでのウェイクアップが選択されたとき、このビットはUSART_RQRレジスタのMMRQビットに ”1” を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3レジスタのSBKRQビットに1を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタが送信されます。

ビット17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICRレジスタのCMCFに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1レジスタのCMIE=1の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RXラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USARTはアイドルです（受信なし）。

1：受信中です。

ビット15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたとき（RXFNEもセットされ、RXFNEIE=1の場合は割込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにハードウェアによってセットされます（ABRE=1）（この場合、ABRE、RXFNE、およびFEもセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQRレジスタのABRRQに1を書き込むことによって、ソフトウェアによってクリアされます。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_RQRレジスタのABRRQビットに1を書き込むことによって、ソフトウェアによってクリアされます。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット13 **UDR**：SPIスレーブアンダーランエラーフラグ

スレーブ送信モードでは、ソフトウェアがUSART_TDRにまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICRレジスタのUDRCFビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USARTがSPIスレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1スマートカードモード）。検出は、受信バイト数がBLEN + 4以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR1レジスタのEOBIE=1の場合、割込みが生成されます。

USART_ICRレジスタのEOBCFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット11 **RTOF**：レシーバタイムアウト

このビットは、RTORレジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICRレジスタのRTOCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2レジスタのRTOIE=1の場合、割込みが生成されます。

スマートカードモードでは、タイムアウトはCWTまたはBWTタイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間がRTORレジスタでプログラムされた値に等しい場合、2つのキャラクタが分離され、RTOFはセットされません。この時間がこの値に2サンプル時間（オーバーサンプリング方式によって2/16または2/8）を加えた値を超える場合、RTOFフラグがセットされます。

カウンタはRE=0の場合でもカウントしますが、RTOFはRE=1のときのみセットされます。REがセットされたときにタイムアウトがすでに経過していた場合、RTOFはセットされます。

USARTがレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット10 **CTS**：CTSフラグ

このビットは、ハードウェアによってセット／リセットされます。CTS入力ピンのステータスの反転コピーです。

0：CTSラインはセットされました。

1：CTSラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット9 **CTSIF**：CTS割込みフラグ

このビットは、CTSEビットがセットされていた場合、CTS入力がトグルしたときにハードウェアによってセットされます。USART_ICRレジスタのCTSCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3レジスタのCTSIE=1の場合、割込みが生成されます。

0：CTSステータスラインでの変化はありません。

1：CTSステータスラインでの変化がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット8 **LBDF**：LINブレーク検出フラグ

このビットは、LINブレークが検出されると、ハードウェアによってセットされます。USART_ICRレジスタのLBDCFに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2レジスタのLBDIE=1である場合、割込みが生成されます。

0：LINブレークは検出されませんでした。

1：LINブレークが検出されました。

注： USARTがLINモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7 **TXFNF**：TXFIFOノットフル

TXFIFOがフルではない、つまりUSART_TDRにデータを書き込めるとき、TXFNFはハードウェアによってセットされます。USART_TDRレジスタへの書込み動作ごとにデータがTXFIFOに格納されます。このフラグはTXFIFOがフルになるまでセットされたままになります。TXFIFOがフルになると、このフラグはクリアされ、データをUSART_TDRに書き込むことができないことを示します。

USART_CR1レジスタのTXFNFIEビット =1の場合、割込みが生成されます。

0：送信FIFOはフルです。

1：送信FIFOはフルではありません。

注： 一掃リクエスト中、TXFIFOが空になるまで、TXFNFはリセットに維持されます。一掃リクエストを（TXFRQビットをセットすることによって）送信した後、TXFIFOに書き込む前にTXFNFフラグをチェックする必要があります（TXFNFおよびTXFEは同時にセットされます）。

このビットは、シングルバッファ送信時に使用されます。

ビット6 **TC**：送信完了

このビットは、USART_TDRに書き込まれた最後のデータがシフトレジスタから送信されたことを示します。TCフラグの動作は次のようになります。

- TDN =0の場合、データを含むフレームの送信が完了し、TXE/TXFEがセットされたとき、TCフラグがセットされます。
- TDNがTXFIFOのデータの数と等しい場合、TXFIFOが空でTDNに達すると、TCフラグがセットされます。
- TDNがTXFIFOのデータの数より多い場合、送信するプログラム済みデータの数に達するようにTXFIFOが再び満たされるまで、TCはクリアされたままとなります。
- TDNがTXFIFOのデータの数より少ない場合、TXFIFOが空でなくても、TDNに達したときにTCがセットされます。

USART_CR1レジスタのTCIE=1の場合、割込みが生成されます。

TCビットは、USART_ICRレジスタのTCCFに ”1” を書き込むことによって、またはUSART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

ビット5 **RXFNE**：RXFIFOノットエンプティ

RXFNEビットは、RXFIFOが空でないとき、つまりデータがUSART_RDRレジスタから読み出せるときに、ハードウェアによってセットされます。USART_RDRからの読出し動作のたびに、RXFIFOの1つの場所が解放されます。

RXFIFOが空になると、RXFNEがクリアされます。RXFNEフラグは、USART_RQRレジスタのRXFRQに1を書き込むことによってクリアすることもできます。

USART_CR1レジスタのRXFNEIE=1の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1レジスタのIDLEIE=1の場合、割込みが生成されます。USART_ICRレジスタのIDLECFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXFNEビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLEビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、USARTがミュートでない場合（RWU=0）、WAKEビットによって選択されたミュートモードに関係なく、IDLEはセットされます。RWU=1の場合、IDLEはセットされません。

ビット3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFOモードが有効な場合はRXFF=“1”）のときに、
現在シフトレジスタに受信中のデータをUSART_RDRレジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICRレジスタのORECFに1を書き込むことによって、ソフトウェアによってクリアされます。
USART_CR1レジスタでRXFNEIE=1、またはUSART_CR3レジスタでEIE=1の場合、割込みが生成されます。
0：オーバーランエラーはありません。
1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDRレジスタの内容は失われませんが、シフトレジスタは上書きされます。EIEビットがセットされている場合、マルチバッファ通信中にOREフラグがセットされた場合、割込みが生成されます。

USART_CR3レジスタのOVRDISビットがセットされると、このビットは永続的に0に強制設定されます（オーバーラン検出なし）。

ビット2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICRレジスタのNFCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。
0：ノイズは検出されていません。
1：ノイズが検出されました。

注： このビットは、割込みを生成するRXFNEビットと同時に出現するため、割込みを生成しません。EIEビットがセットされている場合、マルチバッファ通信中にNEフラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NEフラグを無効にして、ONEBITビットに1をプログラミングして偏差に対するUSARTの許容誤差を増加させることができます（セクション 66.5.9： 2750 ページのクロック偏差に対するUSARTレシーバの許容誤差を参照）。

このエラーはUSART_RDR内のキャラクタに関連します。

ビット1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICRレジスタのFECFビットに1を書き込むことによって、ソフトウェアによってクリアされます。
スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームをNACK）最大送信試行回数に達すると、このビットがセットされます。
USART_CR3レジスタのEIE=1の場合、割込みが生成されます。
0：フレーミングエラーは検出されていません。
1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはUSART_RDR内のキャラクタに関連します。

ビット0 **PE**：パリティエラー

このビットは、受信モードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICRレジスタのPECFに1を書き込むことによって、ソフトウェアによってクリアされます。
USART_CR1レジスタのPEIE=1である場合、割込みが生成されます。
0：パリティエラーはありません。
1：パリティエラー

注： このエラーはUSART_RDR内のキャラクタに関連します。

66.8.10 USART割込みおよびステータスレジスタ [オルタネート]（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタがFIFOモード有効（前のセクション）でも、FIFOモード無効（このセクション）でも使用できます。

FIFOモードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	RE ACK	TE ACK	Res.	RWU	SBKF	CMF	BUSY
						r			r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDRに書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードがNACKを返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3レジスタのTCBGTIE=1の場合、割込みが生成されます。

このビットは、USART_ICRレジスタのTCBGTCFに1を書き込むことによって、またはUSART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信が完了していないか、または送信が正常に完了していません（つまり、カードからNACKを受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードからNACKなし）。

注： USARTがスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USARTがスマートカードモードをサポートする場合でこのモードが有効な場合、TCBGTのリセット値は1です。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット24:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値がUSARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、USARTが受信できる状態であることを確認できます。

注： USARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値がUSARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

USART_CR1レジスタでTE=0を書き込んだ後、TE=1を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0の最小周期を満たすために使用できます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **RWU**：レシーバのミュートモードからのウェイクアップ

このビットは、USARTがミュートモードかどうかを示します。ウェイクアップ／ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたはIDLE）は、USART_CR1レジスタのWAKEビットによって選択されます。

IDLEモードでのウェイクアップが選択されたとき、このビットはUSART_RQRレジスタのMMRQビットに ”1” を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3レジスタのSBKRQビットに1を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタが送信されます。

ビット17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICRレジスタのCMCFに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1レジスタのCMIE=1の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RXラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USARTはアイドルです（受信なし）。

1：受信中です。

ビット15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたとき（RXNEもセットされ、RXNEIE=1の場合は割込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにハードウェアによってセットされます（ABRE=1）（この場合、ABRE、RXNE、およびFEもセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQRレジスタのABRRQに1を書き込むことによって、ソフトウェアによってクリアされます。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_RQRレジスタのABRRQビットに1を書き込むことによって、ソフトウェアによってクリアされます。

注： USARTが自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット13 **UDR**：SPIスレーブアンダーランエラーフラグ

スレーブ送信モードでは、ソフトウェアがUSART_TDRにまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICRレジスタのUDRCFビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USARTがSPIスレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1スマートカードモード）。検出は、受信バイト数がBLEN + 4以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR1レジスタのEOBIE=1の場合、割込みが生成されます。

USART_ICRレジスタのEOBCFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット11 **RTOF**：レシーバタイムアウト

このビットは、RTORレジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICRレジスタのRTOCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2レジスタのRTOIE=1の場合、割込みが生成されます。

スマートカードモードでは、タイムアウトはCWTまたはBWTタイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間がRTORレジスタでプログラムされた値に等しい場合、2つのキャラクタが分離され、RTOFはセットされません。この時間がこの値に2サンプル時間（オーバーサンプリング方式によって2/16または2/8）を加えた値を超える場合、RTOFフラグがセットされます。

カウンタはRE=0の場合でもカウントしますが、RTOFはRE=1のときのみセットされます。REがセットされたときにタイムアウトがすでに経過していた場合、RTOFはセットされます。

USARTがレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット10 **CTS**：CTSフラグ

このビットは、ハードウェアによってセット／リセットされます。CTS入力ピンのステータスの反転コピーです。

0：CTSラインはセットされました。

1：CTSラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット9 **CTSIF**：CTS割込みフラグ

このビットは、CTSEビットがセットされていた場合、CTS入力がトグルしたときにハードウェアによってセットされます。USART_ICRレジスタのCTSCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3レジスタのCTSIE=1の場合、割込みが生成されます。

0：CTSステータスラインでの変化はありません。

1：CTSステータスラインでの変化がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット8 **LBDF**：LINブレーク検出フラグ

このビットは、LINブレークが検出されると、ハードウェアによってセットされます。USART_ICRレジスタのLBDCFに1を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2レジスタのLBDIE=1である場合、割込みが生成されます。

0：LINブレークは検出されませんでした。

1：LINブレークが検出されました。

注： USARTがLINモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7 **TXE**：送信データレジスタエンプティ

このビットは、USART_TDRレジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_TDRレジスタへの書込みによってクリアされます。TXEフラグは、USART_RQRレジスタのTXFRQに1を書き込んでセットし、データを破棄することもできます（スマートカードT=0モードでの送信失敗の場合のみ）。

USART_CR1レジスタのTXEIEビット =1の場合、割込みが生成されます。

0：データレジスタはフルです。

1：データレジスタはエンプティです。

ビット6 **TC**：送信完了

このビットは、USART_TDRに書き込まれた最後のデータがシフトレジスタから送信されたことを示します。データを含むフレームの送信が完了し、TXEがセットされたとき、TCフラグがセットされます。

USART_CR1レジスタのTCIE=1の場合、割込みが生成されます。

TCビットは、USART_ICRレジスタのTCCFに ”1” を書き込むことによって、またはUSART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

ビット5 **RXNE**：読出しデータレジスタノットエンプティ

RXNEビットは、USART_RDRシフトレジスタの内容がUSART_RDRレジスタに転送されると、ハードウェアによってセットされます。これは、USART_RDRレジスタからの読出しによってクリアされます。RXNEフラグは、USART_RQRレジスタのRXFRQに1を書き込むことによってクリアすることもできます。

USART_CR1レジスタのRXNEIE=1の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1レジスタのIDLEIE=1の場合、割込みが生成されます。USART_ICRレジスタのIDLECFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNEビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLEビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、USARTがミュートでない場合（RWU=0）、WAKEビットによって選択されたミュートモードに関係なく、IDLEはセットされます。RWU=1の場合、IDLEはセットされません。

ビット3 **ORE**：オーバーランエラー

このビットは、RXNE="1"（FIFOモードが有効な場合はRXFF="1"）のときに、
現在シフトレジスタで受信中のデータをUSART_RDRレジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICRレジスタのORECFに1を書き込むことによって、ソフトウェアによってクリアされます。
USART_CR1レジスタでRXNEIE=1、またはUSART_CR3レジスタでEIE=1の場合、割込みが生成されます。
1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDRレジスタの内容は失われませんが、シフトレジスタは上書きされます。EIEビットがセットされている場合、マルチバッファ通信中にOREフラグがセットされた場合、割込みが生成されます。

USART_CR3レジスタのOVRDISビットがセットされると、このビットは永続的に0に強制設定されます（オーバーラン検出なし）。

ビット2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICRレジスタのNFCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。
0：ノイズは検出されていません。
1：ノイズが検出されました。

注： このビットは、割込みを生成する RXNE ビットと同時に出現するため、割込みを生成しません。EIEビットがセットされている場合、マルチバッファ通信中にNEフラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NEフラグを無効にして、ONEBITビットに1をプログラミングして偏差に対するUSARTの許容誤差を増加させることができます（セクション 66.5.9： 2750 ページのクロック偏差に対するUSARTレシーバの許容誤差を参照）。

このエラーはUSART_RDR内のキャラクタに関連します。

ビット1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICRレジスタのFECFビットに1を書き込むことによって、ソフトウェアによってクリアされます。
スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームをNACK）最大送信試行回数に達すると、このビットがセットされます。
USART_CR3レジスタのEIE=1の場合、割込みが生成されます。
0：フレーミングエラーは検出されていません。
1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはUSART_RDR内のキャラクタに関連します。

ビット0 **PE**：パリティエラー

このビットは、受信モードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICRレジスタのPECFに1を書き込むことによって、ソフトウェアによってクリアされます。
USART_CR1レジスタのPEIE=1である場合、割込みが生成されます。
0：パリティエラーはありません。
1：パリティエラー

注： このエラーはUSART_RDR内のキャラクタに関連します。

66.8.11 USART割込みフラグクリアレジスタ（USART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMCF	Res.
														w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
		w	w	w		w	w	w	w	w	w	w	w	w	w

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **CMCF**：キャラクター致フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのCMFフラグがクリアされます。

ビット16:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **UDRCF**：SPIスレーブアンダーランフラグクリア

このビットに1を書き込むと、USART_ISRレジスタのUDRFフラグがクリアされます。

注： USARTがSPIスレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4: 2731ページのUSARTの実装を参照してください。

ビット12 **EOBCF**：ブロック終了フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのEOBFフラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4: 2731ページのUSARTの実装を参照してください。

ビット11 **RTOCF**：レシーバタイムアウトフラグクリア

このビットに1を書き込むと、USART_ISRレジスタのRTOFフラグがクリアされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CTSCF**：CTSフラグクリア

このビットに1を書き込むと、USART_ISRレジスタのCTSIFフラグがクリアされます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット8 **LBDCF**：LINブレーク検出フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのLBDFフラグがクリアされます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 66.4： 2731ページのUSARTの実装を参照してください。

ビット7 **TCBGTCF**：ガード時間前送信完了フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのTCBGTフラグがクリアされます。

ビット6 **TCCF**：送信完了フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのTCフラグがクリアされます。

ビット5 **TXFECF**：TXFIFOエンプティフラグクリア

このビットに1を書き込むと、USART_ISRレジスタのTXFEフラグがクリアされます。

ビット4 **IDLECF**：アイドルライン検出フラグクリア

このビットに1を書き込むと、USART_ISRレジスタのIDLEフラグがクリアされます。

ビット3 **ORECF**：オーバーランエラーフラグクリア
このビットに1を書き込むと、USART_ISRレジスタのOREフラグがクリアされます。

ビット2 **NECF**：ノイズ検出フラグクリア
このビットに1を書き込むと、USART_ISRレジスタのNEフラグがクリアされます。

ビット1 **FECF**：フレーミングエラーフラグクリア
このビットに1を書き込むと、USART_ISRレジスタのFEフラグがクリアされます。

ビット0 **PECF**：パリティエラーフラグクリア
このビットに1を書き込むと、USART_ISRレジスタのPEフラグがクリアされます。

66.8.12 USARTレシーバデータレジスタ（USART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **RDR[8:0]**：受信データ値
受信データキャラクタを含みます。
RDRレジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図800を参照）。
パリティを有効にして受信する場合、MSBビットで読み出される値が受信したパリティビットです。

66.8.13 USARTトランスミッタデータレジスタ（USART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **TDR[8:0]**：送信データ値
送信されるデータキャラクタを含みます。
USART_TDRレジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図800を参照）。
パリティを有効にして（USART_CR1レジスタのPCEビットに1をセット）送信しているとき、MSB（データ長に応じてビット7または8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF=1のときのみ書き込む必要があります。

66.8.14 USARTプリスケーラレジスタ（USART_PRESC）

このレジスタは、USARTが無効（UE=0）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PRESCALER[3:0]**：クロックプリスケーラ

USART入力クロックは、以下のようにプリスケーラ分周比によって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが2分周されます。
0010：入力クロックが4分周されます。
0011：入力クロックが6分周されます。
0100：入力クロックが8分周されます。
0101：入力クロックが10分周されます。
0110：入力クロックが12分周されます。
0111：入力クロックが16分周されます。
1000：入力クロックが32分周されます。
1001：入力クロックが64分周されます。
1010：入力クロックが128分周されます。
1011：入力クロックが256分周されます。
残りの組み合わせ：予約済み

注： PRESCALERが許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は1011になります。すなわち、入力クロックは256分周されます。

66.8.15 USART自律モード制御レジスタ（USART_AUTOCR）

アドレスオフセット：0x30

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGSEL[3:0]				IDLEDIS	TRIGEN	TRIGPOL
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TDN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22:19 **TRIGSEL[3:0]**：トリガ選択ビット

USARTの相互作用の説明を参照してください。

このビットフィールドに書き込めるのは、USART_CR1レジスタのUEビットがクリアされているときだけです。

0000：usart_trg0を選択します。

0001：usart_trg1を選択します。

.....

1111：usart_trg15を選択します。

注： このビットフィールドに書き込めるのは、USART_CR1 レジスタの UE ビットがクリアされているときだけです。

ビット18 **IDLEDIS**：トランスミッタ有効化後のアイドルフレーム送信無効化ビット

0：トランスミッタを有効（USART_CR1のTE = 1）にした後にアイドルフレームを送信します。

1：トランスミッタを有効にした後にアイドルフレームを送信しません。

注： このビットフィールドに書き込めるのは、USART_CR1 レジスタの UE ビットがクリアされているときだけです。

ビット17 **TRIGEN**：トリガ有効化ビット

0：トリガは無効です。

1：トリガは有効です。

注： このビットフィールドに書き込めるのは、USART_CR1 レジスタの UE ビットがクリアされているときだけです。

トリガが検出されると、USART_CR1のTEに1がセットされ、データ転送が開始されます。

ビット16 **TRIGPOL**：トリガ極性ビット

このビットフィールドに書き込めるのは、USART_CR1レジスタのUEビットがクリアされているときだけです。

0：トリガ は立ち上がりエッジでアクティブです。

1：トリガ は立ち下がりエッジでアクティブです。

ビット15:0 **TDN[15:0]**：TDN送信データ数

このビットフィールドでは、送信するデータの数をプログラミングできます。これに書き込めるのは、USART_CR1のUEがクリアされているときだけです。

66.8.16 USARTレジスタマップ

表677.USARTレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	USART_CR1 FIFOモードが有効な場合	RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	USART_CR1 FIFOモードが無効の場合	Res.	Res.	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	USART_CR2	ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0			0
0x08	USART_CR3	TXFTCFG[2:0]			RXFTIE[2:0]		RXFTCFG		TCBGTIE	TXFTIE	Res.	Res.	Res.	SCAR CNT[2:0]			Res.	DEP	DEM	DDRE	OVRDIS	ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	リセット値	0	0	0	0	0	0	0	0	0				0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	USART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	USART_GTPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GT[7:0]								PSC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	USART_RTOR	BLEN[7:0]								RTO[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	USART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
	リセット値																												0	0	0	0	0
0x1C	USART_ISR FIFOモードが有効な場合	Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値					0	0	X	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0x1C	USART_ISR FIFOモードが無効の場合	Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値							0			0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0x20	USART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMCF	Res.	Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
	リセット値															0				0	0	0		0	0	0	0	0	0	0	0	0	0
0x24	USART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表677.USARTレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	USART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x2C	USART_ PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0
0x30	USART_ AUTOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGSEL [3:0]				IDLEDIS	TRIGEN	TRIGPOL	TDN[15:0]															
	リセット値										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

67 低消費電力ユニバーサル非同期レシーバトランスミッタ（LPUART）

このセクションでは、低消費電力ユニバーサル非同期レシーバトランスミッタ（LPUART）について説明します。

67.1 概要

LPUARTは、限られた消費電力で双方向UART通信が可能なUARTです。わずか32.768kHz LSEクロックで最大9600 baudのUART通信が可能です。LSEクロックとは別のクロックソースによってLPUARTにクロック供給すると、さらに高いボーレートを達成できます。

マイクロコントローラが低消費電力モードのときでも、LPUARTはUARTフレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUARTには、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェアサポートが含まれています。

半二重単線通信とモデム操作（CTS/RTS）をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信にDMA（直接メモリアクセス）を使用できます。

67.2 LPUARTの主な機能

- 全二重非同期通信
- NRZ標準フォーマット（マーク／スペース）
- プログラム可能なボーレート
- 32.768kHzクロックソースを使用して、300baudから9600baudまで。
- より高い周波数のクロックソースを使用することにより、より高いボーレートを達成可能
- データを送信および受信する2つの内部FIFO
  各FIFOはソフトウェアで有効／無効にすることができ、FIFOの状態用のステータスフラグを装備
- PCLKから独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- プログラム可能なデータワード長（7または8または9bit）
- データ順序をプログラム可能（MSBファースト／LSBファーストのシフト）
- 設定可能なストップビット（1または2個のストップビット）
- 単線半二重通信
- DMAを使用した連続通信
- 送受信バイトは集中型DMAを使用して予約済みSRAMにバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能なTx/Rxピン設定
- モデムとRS-485トランシーバのハードウェアフロー制御
- 転送検出フラグ：
  - 受信バッファフル
  - 送信バッファエンプティ
  - ビジーおよび送信終了フラグ

- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 4つのエラー検出フラグ：
  - オーバーランエラー
  - ノイズ検出
  - フレームエラー
  - パリティエラー
- フラグ付き割込みソース
- マルチプロセッサ通信:アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ
- STOPモードからのウェイクアップ
- STOPモードでの自律機能

67.3 LPUARTの実装

以下の表では、LPUARTの実装について説明します。ここには、比較のためにUSARTとUARTも示します。

表678.STM32U5シリーズでのUSART/UART/LPUARTの実装

インスタンス	STM32U535/545	STM32U575/585	STM32U59x/5Ax/5Fx/5Gx
USART1	フル	フル	フル
USART2	-	フル	フル
USART3	フル	フル	フル
USART6	-	-	フル
UART4	基本	基本	基本
UART5	基本	基本	基本
LPUART1	低消費電力	低消費電力	低消費電力

表679.USART/LPUARTの機能

モードまたは機能(1)	フル機能	基本機能	低電力機能
モデムのハードウェアフロー制御	X	X	X
DMAを使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDECブロック	X	X	-
LINモード	X	X	-
デュアルクロックドメイン	X	X	X

表679.USART/LPUARTの機能 （続き）

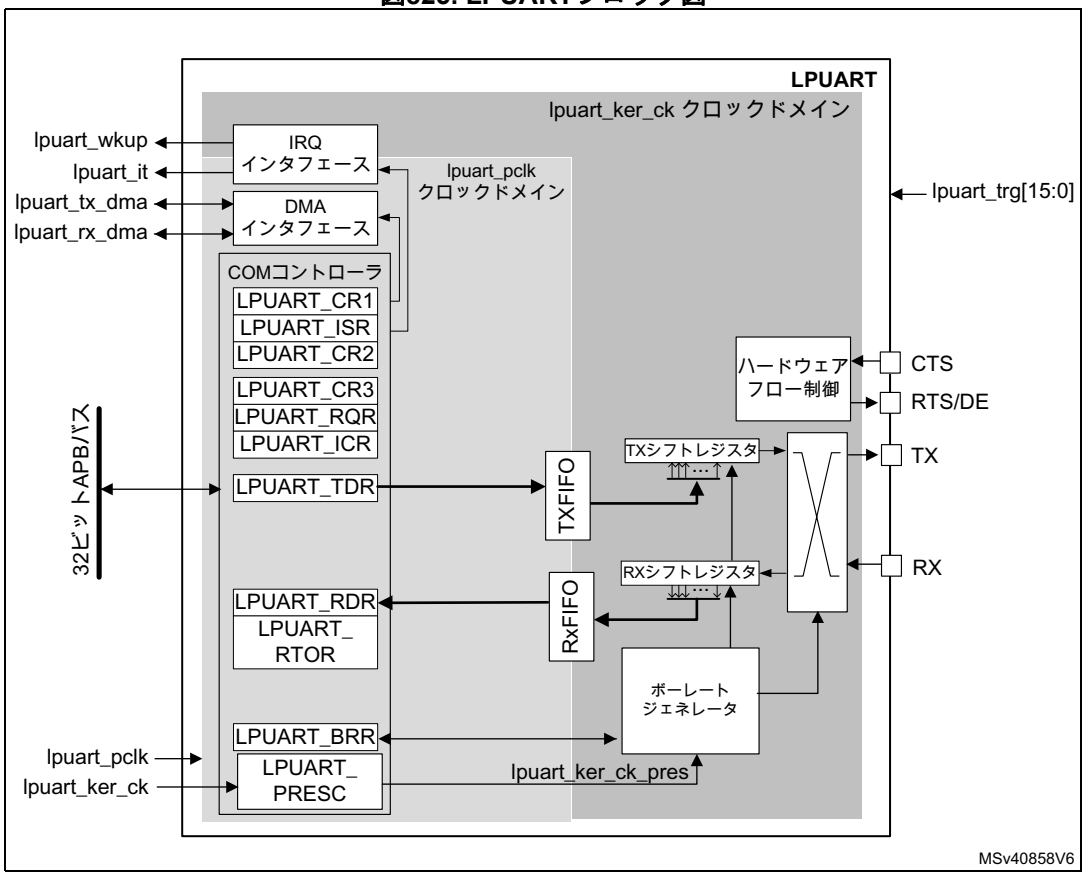
モードまたは機能(1)	フル機能	基本機能	低電力機能
レシーバタイムアウト割込み	X	X	-
Modbus通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USARTデータ長	7、8、および9bit		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFOサイズ	8		
低消費電力モードからのウェイクアップ	X(2)	X (2)	X(3)
自律モード	X	X	X

1. X：サポートされています。
2. STOP 0とSTOP 1モードからのウェイクアップをサポート
3. STOP 0、STOP 1、STOP 2モードからのウェイクアップをサポート

67.4 LPUART機能の詳細

67.4.1 LPUARTブロック図

図825. LPUARTブロック図



67.4.2 LPUARTピンおよび内部信号

LPUARTの入出力ピンの説明

- LPUART双方向通信

  LPUART双方向通信では、少なくとも受信データ入力（RX）と送信データ出力（TX）の2個のピンが必要です。

  – **RX**（受信データ入力）：

    RXはシリアルデータ入力です。

  – **TX**（送信データ出力）

    トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TXピンはハイレベルになります。単線モードでは、このI/Oはデータの送受信に使用されます。

- RS232ハードウェアフロー制御モード

  RS232ハードウェアフロー制御モードでは、以下のピンが必要です。

  – **CTS**（Clear To Send）

    ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。

  – **RTS**（Request To Send）

    ローレベルのとき、この信号はUSARTがデータを受信する準備ができたことを示します。

- RS485ハードウェアフロー制御モード

  RS485ハードウェア制御モードでは、**DE**（Driver Enable）ピンが必要です。この信号は、外部トランシーバの送信モードを有効にします。

LPUART入出力ピンおよび内部信号のリストについては、表680および表681を参照してください。

表680.LPUARTの入出力ピン

ピン名	信号タイプ	説明
LPUART_RX	入力	シリアルデータ受信入力。
LPUART_TX	出力	送信データ出力。
LPUART_CTS	入力	送信可否
LPUART_RTS	出力	送信要求
LPUART_DE(1)	出力	ドライバイネーブル

1. LPUART_DEとLPUART_RTSは同じピンを共有します。

LPUARTの入出力信号の説明

表681.LPUART内部入力／出力信号

ピン名	信号タイプ	説明
usart_pclk	入力	APBクロック
lpuart_ker_ck	入力	LPUARTカーネルクロック
lpuart_wkup	出力	LPUARTはウェイクアップ割込みに対応
lpuart_it	出力	LPUARTグローバル割込み
lpuart_tx_dma	入力／出力	LPUART送信DMAリクエスト

表681.LPUART内部入力／出力信号

ピン名	信号タイプ	説明
lpuart_rx_dma	入力／出力	LPUART受信DMAリクエスト
lpuart_trg[15:0]	入力	LPUARTトリガ

LPUARTの相互接続の説明

表682.LPUARTの相互接続（LPUART1）

信号名	転送元
lpuart_trg0	lpdma1_ch0_tc
lpuart_trg1	lpdma1_ch1_tc
lpuart_trg2	lpdma1_ch2_tc
lpuart_trg3	lpdma1_ch3_tc
lpuart_trg4	exti6
lpuart_trg5	exti8
lpuart_trg6	lptim1_ch1
lpuart_trg7	lptim3_ch1
lpuart_trg8	comp1_out
lpuart_trg9	comp2_out
lpuart_trg10	rtc_alra_trg
lpuart_trg11	rtc_wut_trg
lpuart_trg12	-
lpuart_trg13	-
lpuart_trg14	-
lpuart_trg15	-

67.4.3 LPUARTクロック

図825に示した簡易ブロック図には、以下の完全に独立した2つのクロックドメインが示されています。

- **lpuart_pclk** クロックドメイン

  lpuart_pclkクロック信号はペリフェラルバスインタフェースにクロック供給します。LPUARTレジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **lpuart_ker_ck** カーネルクロックドメイン

  lpuart_ker_ck はLPUARTクロックソースです。これは **lpuart_pclk** から独立しており、RCCによって供給されます。したがって、**lpuart_ker_ck** が停止しているときでも、LPUARTレジスタは読み書きできます。

  デュアルクロックドメイン機能がサポートされていないときは、**lpuart_ker_ck** はlpuart_pclkクロックと同じになります。

lpuart_pclk と **lpuart_ker_ck** の間には何も制約はありません。**lpuart_ker_ck** は **lpuart_pclk** より速いことも遅いこともあり、十分に速く通信を管理できるソフトウェアの能力以外に制限はありません。

67.4.4 LPUARTキャラクタの説明

ワード長は、LPUART_CR1レジスタのMビット（M0：ビット12およびM1：ビット28）をプログラムすることによって、7、8、または9bitに設定できます（図801を参照）。

- 7bitのキャラクタ長：M[1:0] = 10
- 8bitのキャラクタ長：M[1:0] = 00
- 9bitのキャラクタ長：M[1:0] = 01

デフォルトでは、信号（TXまたはRX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

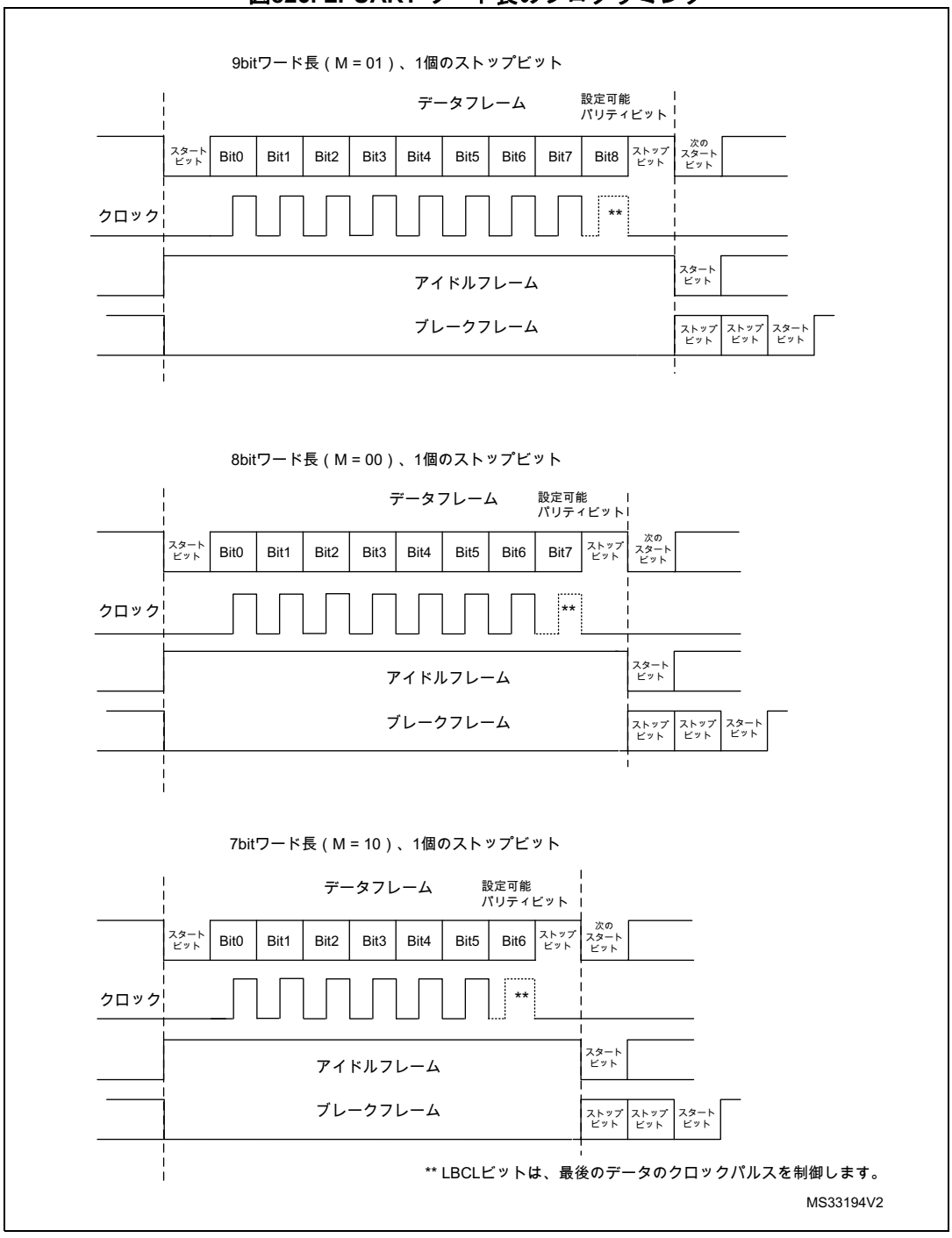
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは2個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図826. LPUART ワード長のプログラミング



67.4.5 LPUARTのFIFOと閾値

LPUARTはFIFOモードで動作できます。

LPUARTは送信FIFO（TXFIFO）と受信FIFO（RXFIFO）を備えています。FIFOモードは、LPUART_CR1レジスタのFIFOENビット（ビット29）をセットすることによって有効になります。

最大のデータワード長が9bitなので、TXFIFOは9bit幅です。しかし、RXFIFOのデフォルト幅は12bitです。この理由は、レシーバはFIFOにデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともにRXFIFOに格納されます。ただし、RDRを読み出すときは、データのみが読み出されます。**

ステータスフラグは、LPUART_ISRレジスタで入手可能です。

TxおよびRX割込みがトリガされるTXFIFOおよびRXFIFOのレベルを定義することができます。これらの閾値は、LPUART_CR3制御レジスタのRXFTCFGおよびTXFTCFGビットフィールドによってプログラムされます。

このとき、

- RXFIFOに受信したデータの数がRXFTCFGビットフィールドでプログラムされた閾値に達すると、Rx割込みが生成されます。

  この場合、LPUART_ISRレジスタのRXFTフラグがセットされます。これはRXFTCFG分のデータ、すなわちLPUART_RDRに1データ、RXFIFOに（RXFTCFG - 1）データを受信したことを意味します。例として、RXFTCFGが 101 にプログラムされている場合、FIFOサイズに相当する量のデータが受信された（FIFOサイズ - 1のデータがRXFIFOに入り、1データがLPUART_RDRに入った）時に、RXFTフラグがセットされます。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO 内の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、Tx割込みが生成されます。

67.4.6 LPUARTトランスミッタ

トランスミッタは、Mビットのステータスに応じて、7または8または9bitのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは、TXピンで出力されます。

キャラクタ送信

LPUART送信時、データはLSBファースト（デフォルト設定）でTXピンにシフトアウトされます。このモードでは、LPUART_TDRレジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（図825を参照）。

FIFOモードが有効なとき、LPUART_TDRレジスタに書き込まれたデータはTXFIFOのキューに入ります。

各キャラクタの前には、スタートビット（1bit周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は1または2にできます。

注： **送信データをLPUART_TDRに書き込む前に、TEビットをセットする必要があります。**

データの送信中にTEビットをリセットしないでください。送信中にTEビットをリセットすると、ボーレートカウンタが停止されるため、TXピンのデータが破壊されます。送信中の現在のデータは失われます。

TEビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

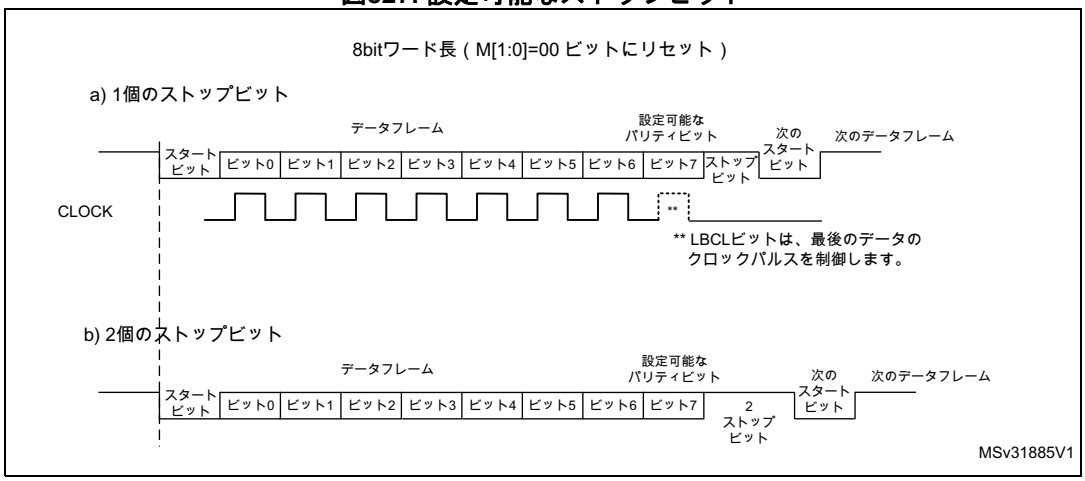
各キャラクタとともに送信されるストップビットの数は、LPUART_CR2（ビット13、12）でプログラミングできます。

- **1つのストップビット**：これはストップビット数のデフォルト値です。
- **2つのストップビット**：これは通常のLPUARTモード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10個のロービット（M[1:0] = “00”のとき）または11個のロービット（M[1:0] = “01”のとき）または9個のロービット（M[1:0] = “10”のとき）の後に2個のストップビットが続きます。長いブレーク（9/10/11個のロービットを超える長さのブレーク）を送信することはできません。

図827. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. LPUART_CR1のMビットをプログラムして、ワード長を定義します。
2. LPUART_BRRレジスタを使用して、目的のボーレートを選択します。
3. LPUART_CR2レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1レジスタのUEビットに1を書き込んで、LPUARTを有効にします。
5. マルチバッファ通信を行う場合は、LPUART_CR3レジスタのDMA有効（DMAT）を選択します。セクション 67.4.13 : DMAおよびLPUARTを使用した連続通信の説明に基づいて、DMAレジスタを設定します。
6. LPUART_CR1のTEビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータをLPUART_TDRレジスタに書き込みますシングルバッファの場合、送信される各データにこの操作を繰り返します。
   - FIFOモードが無効になっている場合は、LPUART_TDRに1つのデータを書き込むと、TXEフラグがクリアされます。
   - FIFOモードが有効になっている場合は、LPUART_TDRに1つのデータを書き込むと、1つのデータがTXFIFOに追加されます。TXFNFフラグがセットされていると、LPUART_TDRへの書込み動作が行われます。このフラグはTXFIFOがフルになるまでセットされたままになります。

8. LPUART_TDRレジスタに最後のデータを書き込んだら、TC=1になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。
   – FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   – FIFOモードが有効になっている場合、これは、TXFIFOとシフトレジスタの両方が空になっていることを示します。

   このチェックは、LPUARTが無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1バイト通信

- FIFOモードが無効の場合

  送信データレジスタに書き込むと、必ずTXEビットがクリアされます。TXEフラグがハードウェアによってセットされ、以下のことを示します。
  – データはLPUART_TDRレジスタからシフトレジスタへ移動され、データ送信が開始しています。
  – LPUART_TDRレジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDRレジスタに書き込めます。

  TXEIEビットがセットされている場合、TXEフラグは割込みを生成します。

  送信が行われているとき、LPUART_TDRレジスタへの書込み命令によってデータがTDRレジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、LPUART_TDRレジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXEビットがセットされます。

- FIFOモードが有効になっている場合は、以下を示すためにハードウェアによってTXFNF（TXFIFOはフルではない）フラグがセットされます。
  – TXFIFOはフルではありません。
  – LPUART_TDRレジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。送信が行われているとき、LPUART_TDRレジスタへの書込み操作によってデータがTXFIFOに格納されます。データは、現在の送信の最後にTXFIFOからシフトレジスタにコピーされます。

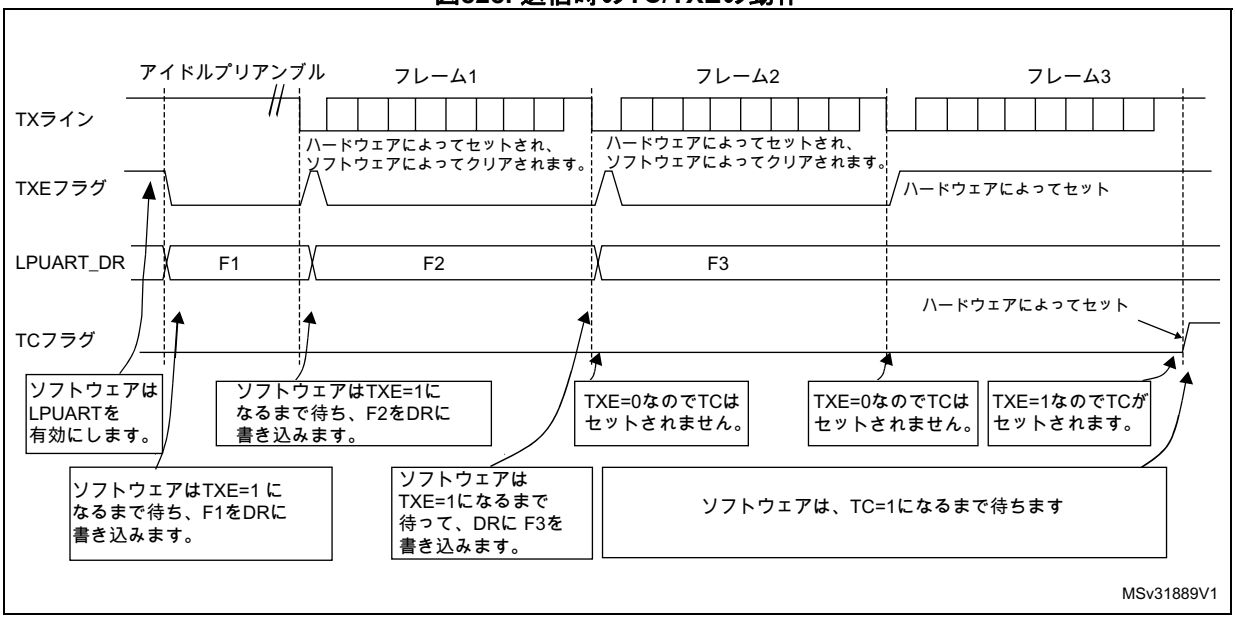
  TXFIFOがフルではない場合、LPUART_TDRレジスタへの書込み後も、TXFNFフラグは1のままに留まります。TXFIFOがフルになると、クリアされます。TXFNEIEビットがセットされている場合、このフラグは割込みを生成します。

  あるいは、TXFIFO閾値に達した時、割込みが生成され、データをTXFIFOに書き込むことができます。この場合、CPUは、プログラムされた閾値によって定義されたデータのブロックを書き込むことができます。

  フレームが送信され（ストップビットの後）、TXEフラグ（FIFOモードの場合はTXFE）がセットされると、TCビットはハイレベルになります。LPUART_CR1レジスタのTCIEビットがセットされると、割込みが生成されます。

  LPUART_TDRレジスタに最後のデータを書き込んだ後は、LPUARTを無効にしたり、マイクロコントローラを低消費電力モードにする前にTC=1になるまで待つ必要があります（図 828：送信時のTC/TXEの動作を参照）。

図828. 送信時のTC/TXEの動作



注： **FIFO管理が有効になっているときは、TXFNFフラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、Mビットに依存します（図826を参照）。

SBKRQビットに1が書き込まれた場合、現在のキャラクタ送信の完了後、TXラインにブレークキャラクタが送信されます。SBKFビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。LPUARTは、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに2bitの長さの間、ロジック1信号（STOP）を挿入します。

SBKRQビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFOモードが有効になっている場合、TXFIFOがフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TEビットをセットすると、LPUARTは、最初のデータフレームの前にアイドルフレームを送信します。

67.4.7 LPUARTレシーバ

LPUARTは、LPUART_CR1レジスタのMビットに応じて、7または8または9bitのデータワードを受信できます。

スタートビット検出

LPUARTでは、立ち下がりエッジがRxラインで発生したときスタートビットが検出され、その後、スタートビットの中央でサンプルが採取され、まだ“0”であることが確認されます。スタートサンプルが“1” であった場合、ノイズエラーフラグ（NE）がセットされ、スタートビットが破棄され、レシーバは新しいスタートビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUARTの受信時には、データはRXピンを通じてLSBファースト（デフォルト設定）でシフトインされます。このモードでは、LPUART_RDRレジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. LPUART_CR1のMビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタLPUART_BRRを使用して、目的のボーレートを選択します。
3. LPUART_CR2レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1レジスタのUEビットに1を書き込んで、LPUARTを有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3レジスタのDMA有効（DMAR）を選択します。セクション 67.4.13 : DMAおよびLPUARTを使用した連続通信の説明に基づいて、DMAレジスタを設定します。
6. LPUART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効な場合は、RXNE ビットがセットされます。同ビットは、シフトレジスタの内容がRDRに転送されたことを示します。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示す RXFNE ビットがセットされます。LPUART_RDRを読み出すと、RXFIFOに入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともにRXFIFOに格納されます。
- RXNEIE（FIFOモードの場合はRXFNEIE） ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFOモードが無効のとき、RXNEフラグはバイト受信ごとにセットされ、受信データレジスタのDMA読出しによってクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMAリクエストのたびに、RXFIFOから1データが取り出されます。DMAリクエストは、RXFIFOが空ではないとき、すなわち、RXFIFOから読み出されるべきデータがあるときに、RXFIFOによってトリガされます。
- シングル-バッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる LPUART_RDR レジスタからの読出しによって行われます。RXNEフラグは、LPUART_RQR レジスタの RXFRQに1を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNEビットをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。LPUART_RDRからの読出し動作のたびに、RXFIFOから1つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、LPUART_RQRレジスタのRXFRQビットに1を書き込むことによってもクリアすることもできます。RXFIFOがフルのとき、オーバーランエラーを避けるため、次のキャラクタの受信が終了する前に、RXFIFO内の最初のエントリを読み出す必要があります。RXFNEIEビットがセットされている場合、RXFNEフラグは割込みを生成します。あるいは、RXFIFO

閾値に達した時、割込みが生成され、データをRXFIFOから読み出すことができます。この場合、CPUは、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USARTはブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIEビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFOモードが無効の場合

  RXNEビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。

  RXNEビットがクリアされない限り、データをシフトレジスタからRDRレジスタに転送することはできません。RXNEフラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときにRXNEフラグがセットされていた場合、または前回のDMAリクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – OREビットがセットされます。
  – RDR の内容は失われません。LPUART_RDR の読出しが行われると、前のデータが使用できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIEビットまたはEIEビットがセットされている場合、割込みが生成されます。

- FIFOモードが有効な場合

  受信FIFOがフルのときにシフトレジスタが転送される準備ができると、オーバーランエラーが発生します。

  RXFIFOに1つの空き場所ができるまで、データをシフトレジスタからLPUART_RDRレジスタに転送することはできません。RXFIFOが空でないとき、RXFNEフラグがセットされます。

  RXFIFOがフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – OREビットがセットされます。
  – RXFIFOの最初のエントリは失われません。LPUART_RDRの読出しを行うと、そのデータが入手できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIEビットまたはEIEビットがセットされている場合、割込みが生成されます。

OREビットは、ICRレジスタのORECFビットをセットすることによってリセットされます。

注： **OREビットがセットされた場合、少なくとも1個のデータが失われています。**

FIFOモードが無効になっているとき、2つの可能性があります。

- **RXNE=1の場合、有効な最後のデータは、受信レジスタRDRに格納され、読出しが可能です。**
- **RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータがRDRで読み出されると同時に新しい（そして失われた）データが受信されると発生します。**

クロックソースの選択

クロックソースの選択は、クロック制御システムを通じて行われます（リセットおよびクロック制御（RCC）のセクションを参照）。クロックソースは、UEビットのセットによってLPUARTを有効にする前に選ぶ必要があります。

クロックソースは、次の2つの基準に従って選択する必要があります。

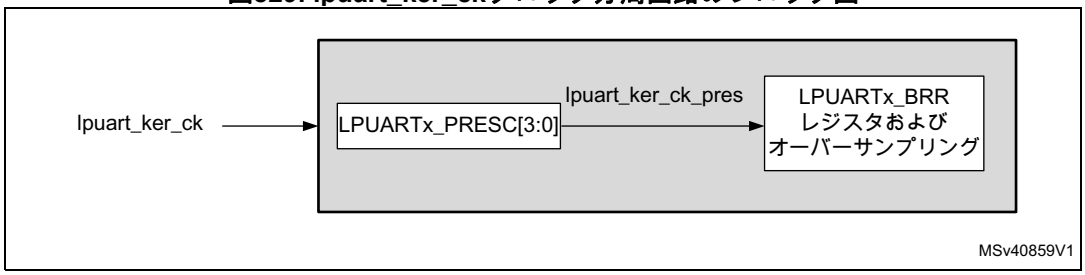
- LPUARTを低消費電力モードで使用できること
- 通信速度

クロックソース周波数は、lpuart_ker_ckです。

デュアルクロックドメインと低消費電力モードからのウェイクアップ機能がサポートされるときには、lpuart_ker_ckクロックソースはRCCで設定できます（**リセットおよびクロック制御（RCC）**のセクションを参照）。そうしない場合、lpuart_ker_ckクロックはlpuart_pclkと同じになります。

lpuart_ker_ckクロックは、LPUART_PRESCレジスタのプログラム可能な係数によって分周できます。

図829. lpuart_ker_ckクロック分周回路のブロック図



lpuart_ker_ckソースによっては、LPUARTはMCUが低消費電力モードのときにデータを受信することができます。受信データとウェイクアップモードの選択に応じて、LPUARTは必要なときにMCUをウェイクアップして、ソフトウェアがLPUART_RDRレジスタを読み出すことによって、またはDMAによって受信データを転送します。

他のクロックソースの場合、LPUART通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは各受信ビットをビット周期のできるだけ中央でサンプリングします。各受信ビットについて1つのサンプルだけが採取されます。

注： **データのノイズ検出は行われません。**

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FEビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタからLPUART_RDRレジスタへ転送されます。
- シングルバイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成するRXNEビットと同時に立ち上がります。マルチバッファ通信の場合、LPUART_CR3レジスタのEIEビットがセットされている場合に割込みが発行されます。

FEビットは、LPUART_ICRレジスタのFECFに1を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、LPUART_CR2の制御ビットを通じて設定でき、通常モードでは1または2にできます。

- **1個のストップビット**：ストップビット1個のサンプリングは、8番目、9番目、および10番目のサンプルで行われます。
- **2個のストップビット**：2個のストップビットのサンプリングは、2番目のストップビットの中央で行われます。RXNEおよびFEフラグは、このサンプルの直後、すなわち2番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

67.4.8 LPUARTボーレート生成

レシーバとトランスミッタ（RxとTx）のボーレートは、LPUART_BRRレジスタでプログラムされた値に設定されます。

$$\text{Tx/Rx ボー} = \frac{256 \times \text{lpuart_ker_ck_pres}}{\text{LPUARTDIV}}$$

LPUARTDIVは、LPUART_BRRレジスタで定義されます。

注： **ボーカウンタは、LPUART_BRRへの書込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更してはなりません。**

LPUART_BRRレジスタに0x300未満の値を書き込むことは禁じられています。

f_{CK} は、3 xボーレートから4096 xボーレートの範囲内でなければなりません。

LPUARTクロックソースがLSEの場合に達成できる最大ボーレートは、9600 baudです。LSEクロックとは別のクロックソースによってLPUARTにクロック供給すると、さらに高いボーレートを達成できます。たとえば、LPUARTクロックソース周波数が100MHzの場合、達成できる最大ボーレートは約33Mbaudです。

表683.lpuart_ker_ck_pres = 32.768kHzでプログラムされたボーレートのエラー計算

ボーレート		lpuart_ker_ck_pres= 32.768kHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） Bレート / 目標のBレート
1	0.3 Kbaud	0.3 Kbaud	0x6D3A	0
2	0.6 Kbaud	0.6 Kbaud	0x369D	0
3	1200 baud	1200.087 baud	0x1B4E	0.007
4	2400 baud	2400.17 baud	0xDA7	0.007
5	4800 baud	4801.72 baud	0x6D3	0.035
6	9600 Kbaud	9608.94 baud	0x369	0.093

表684.f_{CK} = 100MHzでプログラムされたボーレートのエラー計算

ボーレート		f_{CK} = 100MHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） Bレート / 目標のBレート
1	38400 baud	38400.04 baud	A2C2A	0.0001
2	57600 baud	57600.06 baud	6C81C	0.0001
3	115200 baud	115200.12 baud	3640E	0.0001
4	230400 baud	230400.23 baud	1B207	0.0001
5	460800 baud	460804.61 baud	D903	0,001
6	921600 baud	921625.81 baud	6C81	0.0028
7	4000 Kbaud	4000000.00 baud	1900	0
8	10000 Kbaud	10000000.00 baud	A00	0
9	20000 Kbaud	20000000.00 baud	500	0
10	30000 Kbaud	33032258.06 baud	307	0.1

67.4.9 クロック偏差に対するLPUARTレシーバの許容誤差

LPUARTの非同期レシーバは、クロックシステムの合計偏差がLPUARTレシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{LPUARTレシーバの許容誤差}$$

ここで、

DWUは、低消費電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01の場合：

$$DWU = \frac{t_{WULPUART}}{11 \times Tbit}$$

M[1:0] = 00の場合：

$$DWU = \frac{t_{WULPUART}}{10 \times Tbit}$$

M[1:0] = 10の場合：

$$DWU = \frac{t_{WULPUART}}{9 \times Tbit}$$

$t_{WULPUART}$ は、スタートビットの立ち下がりエッジが検出され、クロック（ペリフェラルによるリクエスト）が動作可能状態になってペリフェラルに達し、レギュレータが動作可能状態になるまでの時間です。

LPUARTレシーバは、表685で指定された最大許容偏差まで、データを正しく受信できます。

- LPUART_CR2レジスタのSTOP[1:0] ビットによって定義されたストップビットの数
- LPUART_BRRレジスタの値

表685.LPUARTレシーバの許容誤差

Mビット	768 < BRR < 1024	1024 < BRR < 2048	2048 < BRR < 4096	4096 ≤ BRR
8bit（M=00）、ストップビット1個	1.82%	2.56%	3.90%	4.42%
9bit（M=01）、ストップビット1個	1.69%	2.33%	2.53%	4.14%
7bit（M=10）、ストップビット1個	2.08%	2.86%	4.35%	4.42%
8bit（M=00）、ストップビット2個	2.08%	2.86%	4.35%	4.42%

表685.LPUARTレシーバの許容誤差

Mビット	768 < BRR < 1024	1024 < BRR < 2048	2048 < BRR < 4096	4096 ≤ BRR
9bit（M=01）、ストップビット2個	1.82%	2.56%	3.90%	4.42%
7bit（M=10）、ストップビット2個	2.34%	3.23%	4.92%	4.42%

注： **表685で指定されたデータは、Mビット = 00のとき、受信フレームに正確に10bit時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（Mビット = 01のときには11bit時間、またはMビット = 10のときには9bit時間）。**

67.4.10 LPUARTマルチプロセッサ通信

LPUARTのマルチプロセッサ通信が可能です（ネットワーク内で複数のLPUARTを接続して）。たとえば、1つのLPUARTをマスタとして、そのTX出力を別のLPUARTのRX入力に接続することができます。別のLPUARTはスレーブであり、それぞれのTX出力の論理積をとった上でマスタのRX入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対するLPUARTサービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUART_CR1レジスタのMMEビットをセットする必要があります。

注： **FIFOマネージメントが有効になっていてMMEがすでにセットされている場合は、MMEビットはクリアしてはなりません。クリアした場合はすぐに（2 lpuart_ker_ckサイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- LPUART_ISRレジスタのRWUビットは“1”にセットされます。特定の条件下では、LPUART_RQRレジスタのMMRQビットを通じて、RWUをハードウェアまたはソフトウェアによって自動的に制御できます。

LPUARTは、LPUART_CR1レジスタのWAKEビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

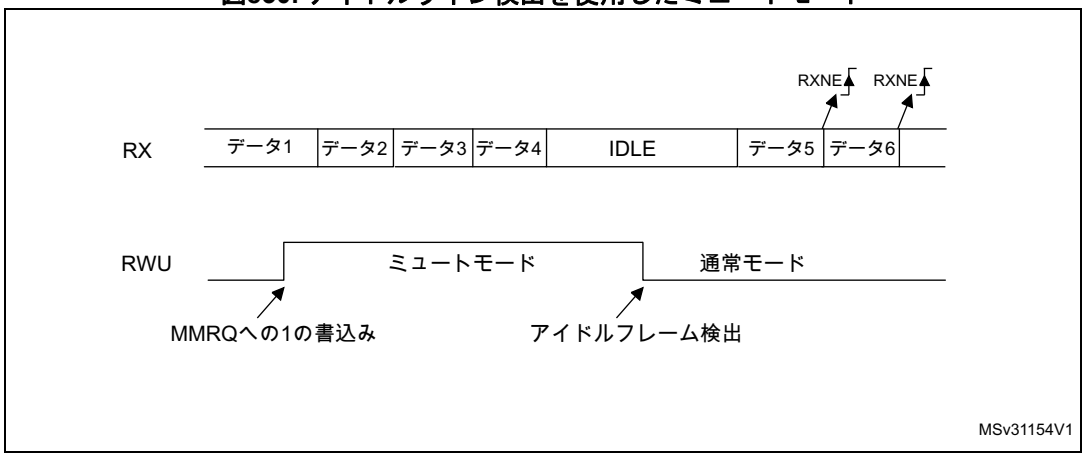
- WAKEビットがリセットされている場合は、アイドルライン検出
- WAKEビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQビットに1が書き込まれ、RWUが自動的にセットされたときには、LPUARTはミュートモードに入ります。

LPUARTは、アイドルフレームを検出するとウェイクアップします。その後、RWUビットはハードウェアによってクリアされますが、LPUART_ISRレジスタのIDLEビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図830に示します。

図830. アイドルライン検出を使用したミュートモード



注： **IDLEキャラクタがすでに経過しているときにMMRQがセットされた場合は、ミュートモードに入りません（RWUはセットされません）。**

ラインがIDLEのときにLPUARTが有効にされた場合、1 IDLEフレーム後にアイドル状態が検出されます（1キャラクタフレームの受信後だけでなく）。

4bit／7bitアドレスマーク検出（WAKE=1）

このモードでは、MSBが ”1” の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは4または7 LSBです。7または4bitアドレス検出の選択は、ADDM7bitを使用して行われます。この4bit／7bitワードは、レシーバによって、LPUART_CR2レジスタのADDビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7bitおよび9bitデータモードでは、アドレス検出は、それぞれ6bitおよび8bitアドレス（ADD[5:0] およびADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUARTはミュートモードに入ります。この場合、RWUビットはハードウェアによってセットされます。LPUARTがミュートモードに入ったときには、このアドレスバイトに対してRXNEフラグはセットされず、割込みもDMAリクエストも発行されません。

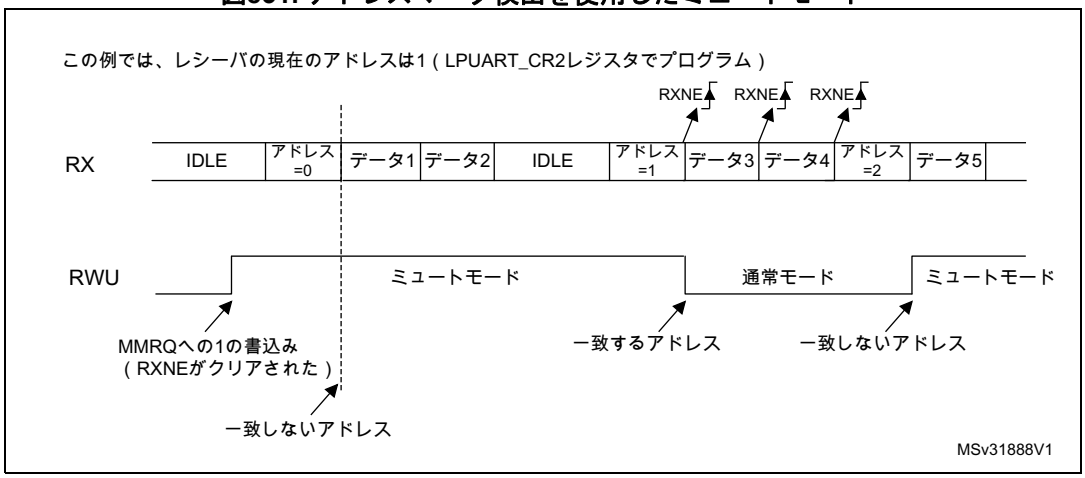
MMRQビットに1が書き込まれたときにも、LPUARTはミュートモードに入ります。この場合、RWUビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUARTはミュートモードを終了します。続いてRWUビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWUビットはクリアされているので、アドレスキャラクタに対してRXNE/RXFNEビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間にMMRQビットがセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を 図831に示します。

図831. アドレスマーク検出を使用したミュートモード



67.4.11 LPUARTパリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、LPUART_CR1レジスタのPCEビットをセットします。Mビットによって定義されたフレーム長に応じて、可能なLPUARTフレームフォーマットを表686に示します。

表686.LPUARTフレームのフォーマット

Mビット	PCEビット	LPUARTフレーム(1)
00	0	| SB | 8bitデータ | STB |
00	1	| SB | 7bitデータ | PB | STB |
01	0	| SB | 9bitデータ | STB |
01	1	| SB | 8bitデータ | PB | STB |
10	0	| SB | 7bitデータ | STB |
10	1	| SB | 6bitデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。

2。 データレジスタでは、PBは常にMSB位置を取ります（Mビットの値に応じて、8または7番目）。

偶数パリティ

パリティビットは、6、7、または8 LSBビット（Mビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101であり、4bitがセットされた場合、偶数パリティが選択された場合（LPUART_CR1のPSビット = 0）、パリティビットは0になります。

奇数パリティ

パリティビットは、6、7、または8 LSBビット（Mビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101であり、4bitがセットされた場合、奇数パリティが選択された場合（LPUART_CR1のPSビット = 1）、パリティビットは1になります。

受信中のパリティチェック

パリティチェックに失敗した場合、LPUART_ISRレジスタのPEフラグがセットされ、LPUART_CR1レジスタのPEIEビットがセットされている場合は割込みが生成されます。PEフラグは、LPUART_ICRレジスタのPECFに1を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUART_CR1のPCEビットがセットされている場合、データレジスタに書き込まれたデータのMSBビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

67.4.12 LPUART単線半二重通信

単線半二重モードを選択するには、LPUART_CR3レジスタのHDSELビットをセットします。

LPUARTは、単線半二重のプロトコルに従うように設定できます。この場合、TXラインとRXラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3レジスタの制御ビットHDSELで行います。

HDSELビットに1が書き込まれると、

- TXラインとRXラインが内部接続されます。
- RXピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TXが外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/Oを設定する必要があります。

この点を除くと、通信プロトコルは通常のLPUARTモードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TEビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： **LPUART通信では、1個のストップビット設定の場合、ストップビットの中央でRXNEフラグがセットされます。**

67.4.13 DMAおよびLPUARTを使用した連続通信

LPUARTは、DMAを使用して連続通信を行うことができます。RxバッファとTxバッファに対するDMAリクエストは、それぞれ独立して生成できます。

注： **DMAモードがサポートされるかどうかについては、セクション 66.4：2731ページのUSARTの実装を参照してください。DMAがサポートされない場合は、セクション66.5.7の説明に従ってLPUSRTを使用してください。連続通信を行うには。FIFOが無効のとき、LPUART_ISRレジスタのTXE/ RXNEフラグをクリアします。**

DMAを使用した送信

DMAモードでの送信を有効にするには、LPUART_CR3レジスタのDMATビットをセットします。TXEフラグ（FIFOモードが有効な場合はTXFNFフラグ）がセットされるたびに、データは、DMAペリフェラル（ダイレクトメモリアクセスコントローラのセクションを参照）を使用してSRAM領域からLPUART_TDRレジスタにロードされます。DMAチャネルをLPUART送信用に割り付けるには、次の手順を実行します（xはチャネル番号を示します）

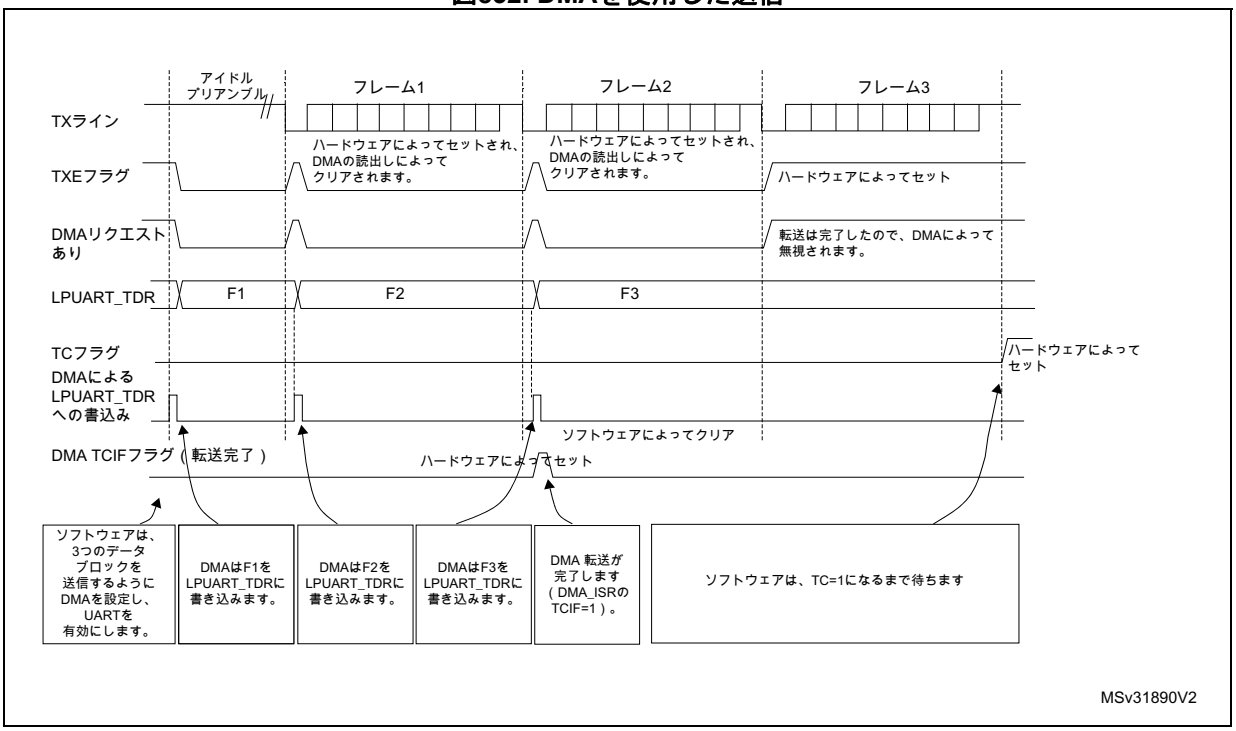
1. DMA制御レジスタにLPUART_TDRレジスタのアドレスを書き込み、これを転送先として設定します。データは、各TXE（またはFIFOモードが有効な場合はTXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域からLPUART_TDRレジスタにロードされます。
3. 転送すべきバイト総数をDMA制御レジスタに設定します。
4. チャネル優先順位をDMA レジスタで設定します。
5. アプリケーションで必要とされる1/2転送終了、転送完了後のDMA割込み生成を設定します。
6. LPUART_ICRレジスタのTCCFビットをセットすることによって、LPUART_ISRレジスタのTCフラグをクリアします。
7. DMAレジスタのチャネルを有効にします。

DMAコントローラにプログラミングされたデータ転送数に達すると、DMAコントローラは、DMAチャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータをDMAが書き込むと（DMA_ISRレジスタのTCIFフラグがセットされます）、TCフラグを観察してLPUART通信の完了を確認することができます。これは、LPUARTを無効にしたり 低消費電力モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1になるまで待つ必要があります。TCフラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

注： **DMAの転送終了前にDMATビットをクリアしないでください。**

図832. DMAを使用した送信



注： **FIFO管理が有効になっているときは、DMAリクエストは、送信FIFOノットフル（すなわち、TXFNF = 1）によってトリガされます。**

DMAを使用した受信

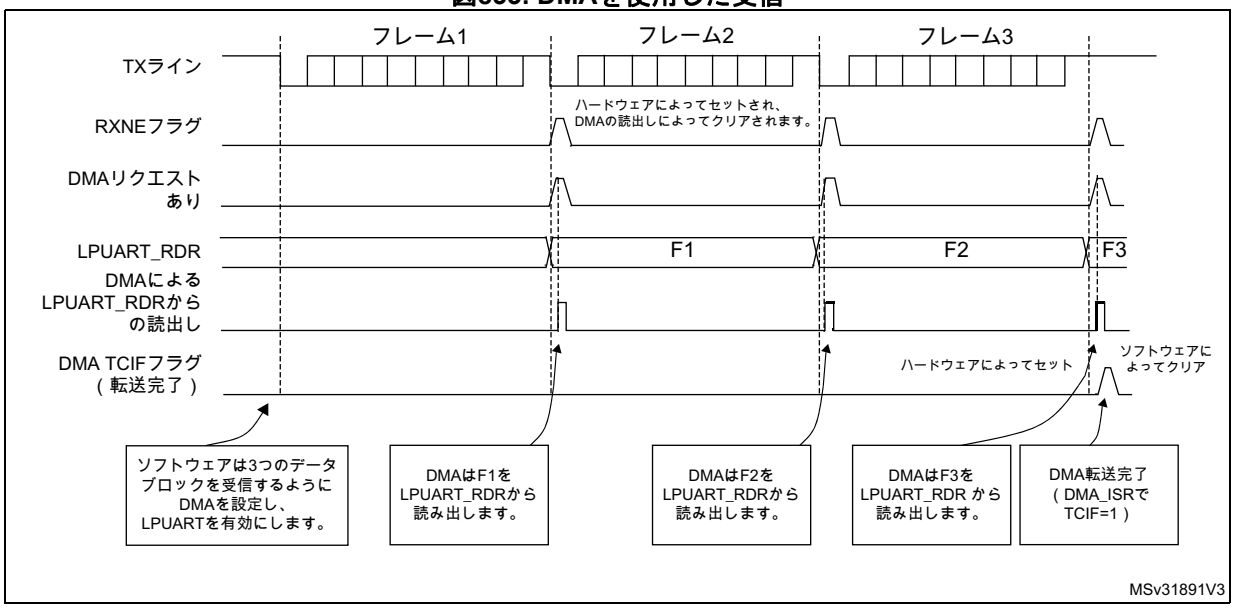
DMAモードでの受信を有効にするには、LPUART_CR3レジスタのDMARビットをセットします。データバイトが受信されるたびに、データは、LPUART_RDRレジスタからDMAペリフェラル（Direct memory access controller (DMA)（DMA）のセクション）を使用して設定されたSRAM領域にロードされます。DMAチャネルをLPUART受信用に割り付けるには、次の手順を実行します。

1. DMA制御レジスタにLPUART_RDRレジスタのアドレスを書き込み、これを転送元として設定します。データは、各RXNE（FIFOモードが有効な場合はRXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各RXNE（FIFOモードが有効な場合はRXFNE）イベント後に、LPUART_RDRからこのメモリ領域にロードされます。
3. 転送すべきバイト総数をDMA制御レジスタに設定します。
4. チャネル優先順位をDMA制御レジスタで設定します。
5. アプリケーションで必要とされる1/2転送終了、転送完了後の割込み生成を設定します。
6. DMA制御レジスタのチャネルを有効にします。

DMAコントローラにプログラミングされたデータ転送数に達すると、DMAコントローラは、DMAチャネルの割込みベクタに基づいて割込みを生成します。

注： **DMAの転送終了前にDMARビットをクリアしないでください。**

図833. DMAを使用した受信



注： **FIFO管理が有効になっているときは、DMAリクエストは、受信FIFOノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。**

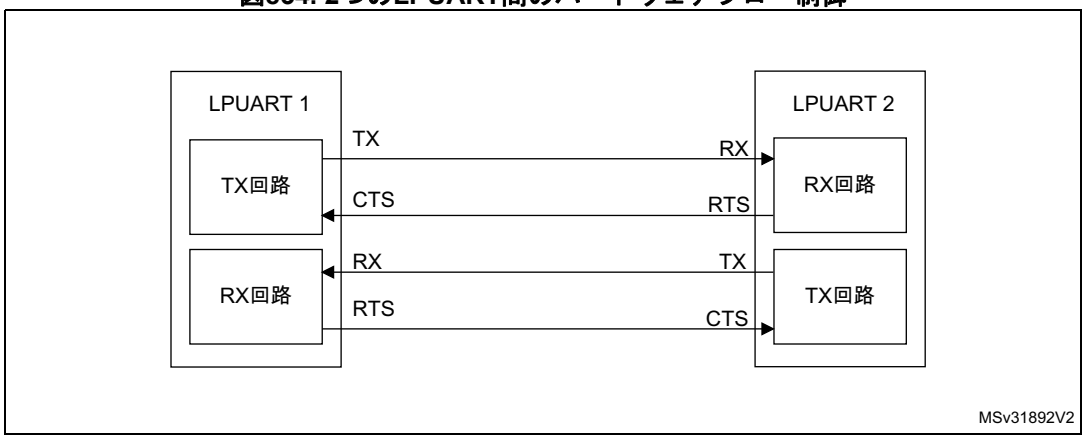
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1バイト受信においてRXNE（FIFOモードが有効な場合はRXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（LPUART_CR3レジスタのEIEビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

67.4.14 RS232ハードウェアフロー制御およびRS485ドライバ有効

CTS入力とRTS出力を使用すると、2つのデバイス間でシリアルデータフローを制御できます。図834に、このモードで2つのデバイスを接続する方法を示します。

図834. 2つのLPUART間のハードウェアフロー制御

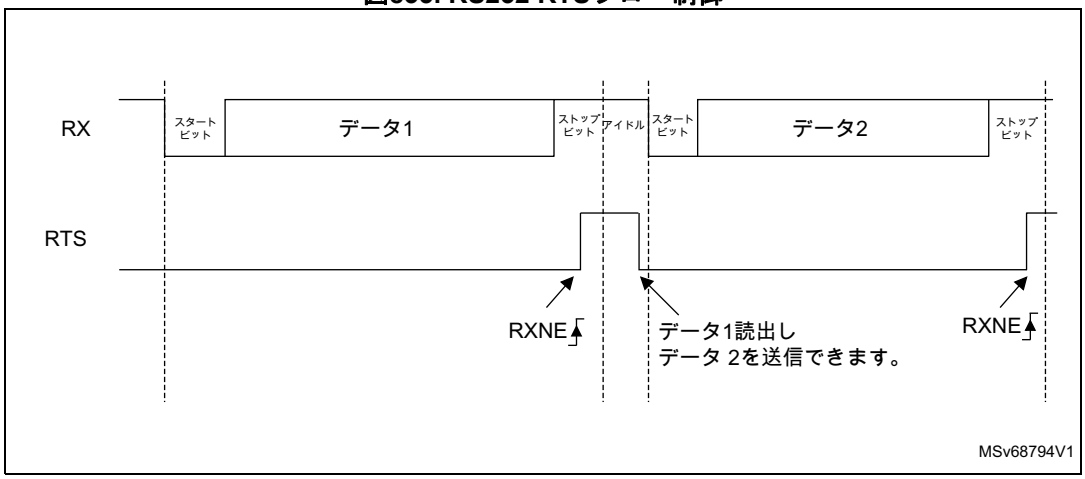


RS232 RTSとCTSのフロー制御は、LPUART_CR3レジスタのRTSEビットとCTSEビットにそれぞれ1を書き込むことによって、個別に有効にできます。

RS232 RTSフロー制御

RTSフロー制御が有効な場合（RTSE=1）、LPUARTレシーバが新しいデータを受信可能である限り、RTSがネゲートされます（ローレベル接続）。受信レジスタがフルになるとRTSがアサートされ、現在のフレームの終わりに送信が停止する予定であることを示します。図835に、RTSフロー制御が有効な場合の通信例を示します。

図835. RS232 RTSフロー制御



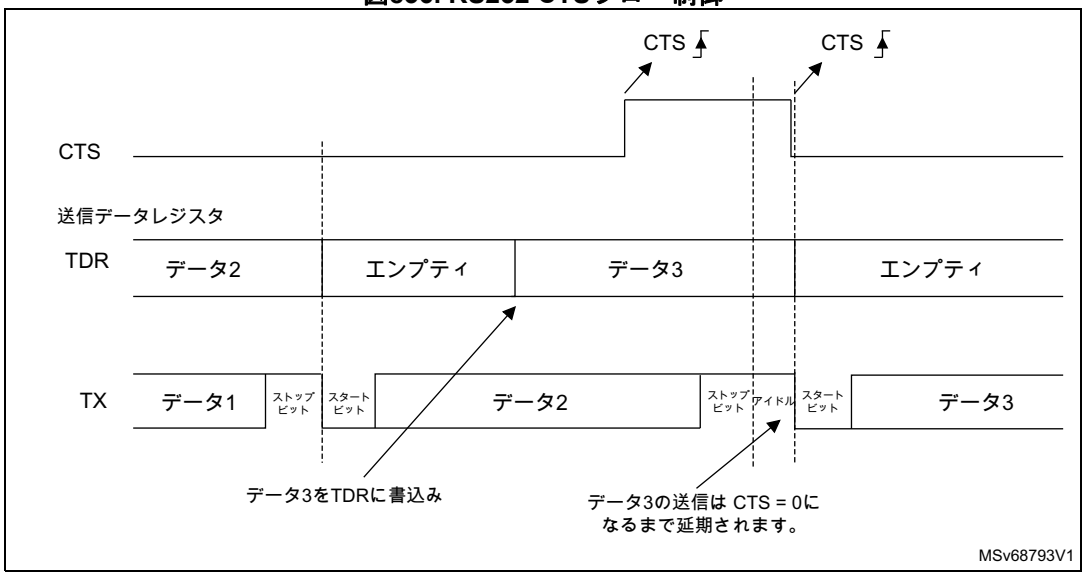
注： **FIFOモードが有効な場合は、RXFIFOがフルのときにのみ、RTSがアサートされます。**

RS232 CTSフロー制御

CTSフロー制御が有効な場合（CTSE=1）、トランスミッタは、CTS入力をチェックしてから、次のフレームを送信します。CTSがネゲートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまりTXE/TXFE=0の場合）。そうでない場合、送信は行われません。送信中にCTSがアサートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1の場合、CTS入力がトグルすると、CTSIFステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUART_CR3レジスタのCTSIEビットがセットされている場合、割込みが生成されます。図836に、CTSフロー制御が有効な場合の通信例を示します。

図836. RS232 CTSフロー制御



注： **正しい動作のために、CTSは、現在のキャラクタの終了の少なくとも3 LPUARTクロックソース周期前にネゲートする必要があります。さらに、2 x PCLK周期より短いパルスではCTSCFフラグがセットされない場合があることに注意してください。**

RS485ドライバ有効

ドライバ有効機能を有効にするには、LPUART_CR3制御レジスタのビットDEMをセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE信号の有効化から スタートビットの開始までの時間です。LPUART_CR1制御レジスタのDEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了からDE信号の無効化までの時間です。LPUART_CR1制御レジスタのDEDT [4:0] ビットフィールドを使用してプログラムされます。DE信号の極性は、LPUART_CR3制御レジスタのDEPビットを使用して設定できます。

LPUARTのDEATおよびDEDTはLPUARTクロックソース（f_{CK}）サイクルで表されます。

- ドライバ有効アサーション時間は次のようになります。
  - (1 + (DEAT x P)) x f_{CK} 、P ≠ 0の場合
  - (1 + DEAT) x f_{CK}、P = 0の場合
- ドライバ有効ネゲート時間は次のようになります。
  - (1 + (DEDT x P)) x f_{CK} 、P ≠ 0の場合
  - (1 + DEDT) x f_{CK}、P = 0の場合

ここで、P = BRR[20:11]

67.4.15 LPUART自律モード

LPUARTペリフェラルは、自律モードによってSTOPモードでも機能します。自律モードは、RUNモードおよびSLEEPモードでも使用できます。低電力モードに入る前にUESMビットをセットする必要があります。

APBクロックは、LPUARTステータスを更新する必要があるたびにペリフェラルによって要求されます。LPUARTは カーネルクロックとAPBクロックを受信すると、ペリフェラルの設定に応じて割込みまたはDMAリクエストを生成します。

割込みが生成された場合、デバイスはSTOPモードからウェイクアップします。割込みが生成されない場合、デバイスはSTOPモードのままですが、APBクロックは、リセットおよびクロックコントローラ（RCC）で有効になっているLPUARTおよびすべての自律ペリフェラルで引き続き使用可能です。DMAリクエストが有効になっている場合は、製品がSTOPモードに留まっている間、データはDMAによってSRAMとの間で直接転送されます。

送信モード

送信時、APBクロックはTEビットがセットされていて以下の場合にのみ要求されます。

- FIFOモードが有効な場合は、次の場合にAPBクロックが要求されます。
  - TxFIFOが空で（TXFE = 1）、対応する割込みが有効になっている（TXFEIE = 1）
  - TxFIFOの閾値に達していて（TXFT = 1）、対応する割込みが有効になっている（TXFTIE = 1）
  - TxFIFOがフルでなく（TXFNF = 1）、対応する割込みまたはDMAが有効になっている（TXFNFIE = 1またはDMAT = 1）
- FIFOモードが無効の場合、データがシフトレジスタに転送されるとすぐにAPBクロックが要求されます。DMAまたは関連する割込みを有効にしておく必要があります。

TEビットは、非同期トリガが検出されると、ハードウェアによってセットされます。

送信は、RUN、SLEEP、またはSTOPモードにおいて、非同期トリガが検出されると自動的に開始されます。トリガは、LPUART_AUTOCR レジスタのTRIGSEL ビットで選択します。これにより、LPUART_CR1レジスタのTEビットがセットされ、転送を有効にするAPBクロック要求が生成されます。送信が完了するまで APB クロックが要求され、プログラムされた送信データ数（LPUART_AUTOCRレジスタのTDNビットフィールド）に達するとハードウェアによってTEビットがクリアされます。この場合、送信すべきデータ数に達し、最後のバイトを送信した時点でTCフラグがセットされます。

受信モード

- FIFOモードが有効な場合は、次の場合にAPBクロックが要求されます。
  - RxFIFIOがフルで（RXFF = 1）、対応する割込みが有効になっている（RXFFIE = 1）
  - RxFIFIOの閾値に達していて（RXFT = 1）、対応する割込みが有効になっている（RXFTIE = 1）
  - RxFIFIOが空でなく（RXFNE = 1）、対応する割込みまたはDMAが有効になっている（RXFNEIE = 1）
- FIFOモードが無効な場合、APBクロックはLPUARTがデータのサンプリングを終了し、LPUART_RDRに書き込む準備ができた時に要求されます。DMAまたは関連する割込みを有効にしておく必要があります。

注： **受信モードでは、オーバーランエラーが発生する（ORE = 1）と、APBクロックが要求されます。割込みの発生とMCUのウェイクアップを有効にするには、EIEビットを設定し、OVRDISビットをクリ**

アした状態にする必要があります。APBクロックリクエストは、割込みフラグがクリアされるまで、維持されます。

受信モードでは、パリティ /ノイズ/フレーミングエラーが発生して、DMAが受信に使用されるときにAPBクロックが要求されます。APBクロックリクエストは、割込みフラグがクリアされるまで、維持されます。

MCUの低消費電力モードからの正常なウェイクアップを可能にする最大LPUARTボーレートの決定

マイクロコントローラの低消費電力モードからの正常なウェイクアップを可能にする最大LPUARTボーレートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）とLPUARTレシーバの許容誤差（セクション 67.4.9 : クロック偏差に対するLPUARTレシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、Mビット = 01、ONEBIT = 0、BRR [3:0] = 0000とします。

この条件では、表 685 : LPUARTレシーバの許容誤差によると、LPUARTレシーバの許容誤差は3.41%です。

DTRA + DQUANT + DREC + DTCL + DWU < LPUARTレシーバの許容誤差

$D_{WUmax} = t_{WULPUART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min} = t_{WULPUART} / (11 \times D_{WUmax})$

ここでは、$t_{WULPUART}$ が低消費電力モードからのウェイクアップ時間です。

パラメータDTRA、DQUANT、DREC、およびDTCLが0% であるときに理想的なケースを考えた場合、DWUの最大値は3.41% です。実際には、最低lpuart_ker_ck精度を考慮する必要があります。

たとえばHSIがlpuart_ker_ckとして使用され、HSI精度が1% の場合、以下を得ることができます。

$t_{WULPUART}$ = 3µs（値は例です。正確な値については、デバイスのデータシートを参照してください）。

D_{WUmax} = 3.41% - 1% = 2.41%

$T_{bit\ min}$ = 3µs/ (11 x 2.41%) = 11.32µs

その結果、低電力モードからの正しいウェイクアップを可能にする最大ボーレートは、1/11.32µs = 88.36kbaudとなります。

67.5 低消費電力モードのLPUART

表687.低消費電力モードがLPUARTに与える影響

モード	説明
SLEEP	影響はありません。LPUART割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	LPUARTレジスタの内容は保持されます。 LPUARTがSTOPモードで利用可能なオシレータによってクロック供給されている場合、非同期モードでの転送は機能します。DMAリクエストは機能しており、割込みによってデバイスはSTOPモードを終了します。
STANDBY	LPUARTペリフェラルはパワーダウンされ、STANDBYモード終了後に再初期化する必要があります。

1. STOPモードからのウェイクアップが指定のペリフェラルインスタンスでサポートされているかどうかについては、セクション 67.3 : LPUARTの実装を参照してください。インスタンスが指定のSTOPモードで機能しない場合、このSTOPモードを開始する前に無効化する必要があります。

67.6 LPUART割込み

すべてのLPUART割込みリクエストの詳細な説明については、表688を参照してください。

表688.LPUART割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEPモードの終了	STOP(1)モードの終了	STANDBYモードの終了
LPUART	送信データレジスタエンプティ	TXE	TXEIE	TDRを書き込む	可能	あり	なし
	送信FIFOノットフル	TXFNF	TXFNFIE	TXFIFOフル		あり	
	送信FIFOエンプティ	TXFE	TXFEIE	TXFRQにTDRまたは1を書き込む		可能	
	送信FIFO閾値到達	TXFT	TXFTIE	TDRを書き込む		可能	
	CTS割込み	CTSIF	CTSIE	CTSCFに1を書き込む		不可	
	送信完了	TC	TCIE	TCCFにTDRまたは1を書き込む		可能	
	受信データレジスタノットエンプティ（データの読出し可能）	RXNE	RXNEIE	RXFRQでRDRまたは1を読み出す	可能	あり	
	受信FIFOノットエンプティ	RXFNE	RXFNEIE	RXFIFOエンプティまでRDRを読み出すか、RXFRQに1を書き込む		可能	
	受信FIFOフル	RXFF(2)	RXFFIE	読出しはRDRです。		可能	
	受信FIFO閾値到達	RXFT	RXFTIE	読出しはRDRです。		可能	
	オーバーランエラー検出	ORE	RXNEIE/RXFNEIE	ORECFに1を書き込む		**あり**	
	アイドルライン検出	IDLE	IDLEIE	IDLECFに1を書き込む		不可	
	パリティエラー	PE	PEIE	PECFに1を書き込む		あり(3)	
	マルチバッファ通信でノイズエラー。	NE	EIE	NFCFに1を書き込む		あり (3)	
	マルチバッファ通信でオーバーランエラー。	ORE(4)		ORECFに1を書き込む		あり	
	マルチバッファ通信でフレーミングエラー。	FE		FECFに1を書き込む		あり (3)	
	キャラクター致	CMF	CMIE	CMCFに1を書き込む		あり(5)	

1. LPUARTは、ペリフェラルインスタンスがSTOPモードからのウェイクアップ機能をサポートしている場合にのみデバイスをSTOPモードからウェイクアップすることができます。サポートされているSTOPモードのリストについては、セクション 67.3 : LPUARTの実装を参照してください。
2. RXFFフラグは、LPUARTがn+1個のデータ、すなわち、RXFIFOにn個のデータ、LPUART_RDRに1個のデータを受信した場合にアサートされます（nはRXFIFOのサイズ）。STOPモードでは、LPUART_RDRはクロック供給されません。その結果として、このレジスタは書き込まれず、n個のデータが受信されてRXFIFOに書き込まれた後に、RXFF割込みがアサートされます（RXFFフラグはセットされません）。
3. パリティ/ノイズ/フレーミング･エラー割込みが発生すると、DMAが使用されている場合にSTOPモードからのウェイクアップが可能になります。
4. OVRDIS = 0の場合。
5. FIFOモードが有効な場合、DMAを使用する必要があります。

67.7 LPUARTレジスタ

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスする必要があります。

67.7.1 LPUART制御レジスタ1（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタがFIFOモード有効（このセクション）でも、FIFOモード無効（次のセクション）でも使用できます。

FIFOモードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **RXFFIE**：RXFIFOフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのRXFFが1になるとLPUART割込みが生成されます。

ビット30 **TXFEIE**：TXFIFOエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのTXFEが1になるとLPUART割込みが生成されます。

ビット29 **FIFOEN**：FIFOモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFOモードは無効です。

1：FIFOモードは有効です。

ビット28 **M1**：ワード長

このビットはビット12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00：スタートビット1個、データビット8個、ストップビットn個

M[1:0] = 01：スタートビット1個、データビット9個、ストップビットn個

M[1:0] = ‘10：スタートビット1個、データビット7個、ストップビットn個

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitデータ長モードでは、スマートカードモード、LINマスタモード、および自動ボーレート（0x7Fおよび0x55フレーム検出）はサポートされません。

ビット27:26 予約済みであり、リセット値に保持する必要があります。

ビット25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この5bit値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 66.5.21：RS232ハードウェアフロー制御およびRS485ドライバ有効を参照してください。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この5bit値は、送信メッセージの最後のストップビットの終了からDE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 67.4.14：RS232ハードウェアフロー制御およびRS485ドライバ有効を参照してください。

DEDT時間中にLPUART_TDRレジスタに書込みが行われた場合、DEDT時間とDEAT時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのCMFビットがセットされると、LPUART割込みが生成されます。

ビット13 **MME**：ミュートモード有効

このビットは、LPUARTのミュートモード機能を有効にします。セットされると、LPUARTは、WAKEビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット12 **M0**：ワード長

このビットはビット28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット28（M1）の説明を参照）。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット11 **WAKE**：レシーバのウェイクアップ方式

このビットによって、ミュートモードからのLPUARTのウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティはMSB位置（M=1の場合はビット9、M=0の場合はビット8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後でPCEが有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCEビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット8 **PEIE**：PE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのPEが1になるとLPUART割込みが生成されます。

ビット7 **TXFNFIE**：TXFIFOノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタでTXFNF=1のときにはLPUART割込みが生成されます。

ビット6 **TCIE**：送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのTCが1になるとLPUART割込みが生成されます。

ビット5 **RXFNEIE**：RXFIFOノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのORE=1またはRXFNE=1のときには、LPUART割込みが生成されます。

ビット4 **IDLEIE**：IDLE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのIDLEが1になるとLPUART割込みが生成されます。

ビット3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。自律モードが使用されていない場合、TEビットは、ソフトウェアによってセット／クリアされます。自律モードが使用されている場合、TEビットはステータスビットになり、ハードウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： **スマートカードモードの場合を除いて、送信中にTEビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐにはTEに1を書き込まないでください。必要な時間を確保するために、ソフトウェアはUSART_ISRレジスタのTEACKビットをポーリングできます。**

スマートカードモードでは、TEがセットされると、送信が開始されるまでに1bit時間の遅れが生じます。

ビット2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット1 **UESM**：LPUART低電力モードで有効

このビットがクリアされると、LPUART は そのカーネルクロックを要求できず、低電力モードでは機能しません。

このビットがセットされると、LPUART はMCUを低消費電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低電力モードで機能しません。

1：LPUART は低電力モードで機能します。

注： **UESMビットは初期化フェーズでセットする必要があります。**

LPUART**が 低電力モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。****セクション 67.3： 2815ページのLPUARTの実装を参照してください。**

ビット0 **UE**：LPUART有効

このビットがクリアされると、LPUARTプリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUARTの設定は保たれますが、LPUART_ISRのステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUARTプリスケーラと出力は無効であり、低電力モードです。

1：LPUARTは有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには、TEビットをその前にリセットする必要があり、ソフトウェアは 、UEビットをリセットする前にLPUART_ISRのTCビットがセットされるのを待つ必要があります。

UE=0のときにはDMAリクエストもリセットされるので、UEビットをリセットする前にDMAチャネルを無効にする必要があります。

67.7.2 LPUART制御レジスタ1[オルタネート]（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタがFIFOモード有効（前のセクション）でも、FIFOモード無効（このセクション）でも使用できます。

FIFOモードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
		rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **FIFOEN**：FIFOモード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFOモードは無効です。

1：FIFOモードは有効です。

ビット28 **M1**：ワード長

このビットはビット12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00：スタートビット1個、データビット8個、ストップビットn個

M[1:0] = 01：スタートビット1個、データビット9個、ストップビットn個

M[1:0] = ‘10：スタートビット1個、データビット7個、ストップビットn個

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitデータ長モードでは、スマートカードモード、LINマスタモード、および自動ボーレート（0x7Fおよび0x55フレーム検出）はサポートされません。

ビット27:26 予約済みであり、リセット値に保持する必要があります。

ビット25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この5bit値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 66.5.21：RS232ハードウェアフロー制御およびRS485ドライバ有効を参照してください。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この5bit値は、送信メッセージの最後のストップビットの終了からDE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 67.4.14：RS232ハードウェアフロー制御およびRS485ドライバ有効を参照してください。

DEDT時間中にLPUART_TDRレジスタに書込みが行われた場合、DEDT時間とDEAT時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのCMFビットがセットされると、LPUART割込みが生成されます。

ビット13 **MME**：ミュートモード有効

このビットは、LPUARTのミュートモード機能を有効にします。セットされると、LPUARTは、WAKEビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット12 **M0**：ワード長

このビットはビット28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット28（M1）の説明を参照）。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット11 **WAKE**：レシーバのウェイクアップ方式

このビットによって、ミュートモードからのLPUARTのウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティはMSB位置（M=1の場合はビット9、M=0の場合はビット8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後でPCEが有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCEビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット8 **PEIE**：PE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのPEが1になるとLPUART割込みが生成されます。

ビット7 **TXEIE**：送信データレジスタエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタでTXE=1のときにはLPUART割込みが生成されます。

ビット6 **TCIE**：送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのTCが1になるとLPUART割込みが生成されます。

ビット5 **RXNEIE**：受信データレジスタノットエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのORE=1またはRXNE=1のときには、LPUART割込みが生成されます。

ビット4 **IDLEIE**：IDLE割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのIDLEが1になるとLPUART割込みが生成されます。

ビット3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。自律モードが無効な場合、TEビットは、ソフトウェアによってセット／クリアされます。自律モードが有効な場合、TEビットはステータスビットになり、ハードウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： **スマートカードモードの場合を除いて、送信中にTEビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐにはTEに1を書き込まないでください。必要な時間を確保するために、ソフトウェアはLPUART_ISRレジスタのTEACKビットをポーリングできます。**

スマートカードモードでは、TEがセットされると、送信が開始されるまでに1bit時間の遅れが生じます。

ビット2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット1 **UESM**：LPUART低電力モードで有効

このビットがクリアされると、LPUART は そのカーネルクロックを要求できず、低電力モードでは機能しません。

このビットがセットされると、LPUART はMCUを低消費電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低電力モードで機能しません。

1：LPUART は低電力モードで機能します。

注： **UESMビットは初期化フェーズでセットする必要があります。**

LPUARTが 低電力モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 67.3： 2815ページのLPUARTの実装を参照してください。

ビット0 **UE**：LPUART有効

このビットがクリアされると、LPUARTプリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUARTの設定は保たれますが、LPUART_ISRのステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUARTプリスケーラと出力は無効であり、低電力モードです。

1：LPUARTは有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには、TEビットをその前にリセットする必要があり、ソフトウェアは 、UEビットをリセットする前にLPUART_ISRのTCビットがセットされるのを待つ必要があります。

UE=0のときにはDMAリクエストもリセットされるので、UEビットをリセットする前にDMAチャネルを無効にする必要があります。

67.7.3 LPUART制御レジスタ2（LPUART_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
rw		rw	rw								rw				

ビット31:24 **ADD[7:0]**：LPUARTノードのアドレス

これらのビットは、ミュートモードのLPUARTノードのアドレス、または低消費電力モードまたはRUNモードで認識されるキャラクタコードを指定します。

- ミュートモードでは、マルチプロセッサ通信で4bit／7bitアドレスマーク検出によってミュートモードからウェイクアップするために使用されます。トランスミッタによって送信されるキャラクタのMSBは1でなければなりません。4bitアドレスマーク検出では、ADD[3:0] ビットのみが使用されます。
- 低消費電力モードでは、キャラクター致時に低消費電力モードからのウェイクアップに使用されます。低電力モード中に受信したキャラクタがADD[7:0] ビットフィールドでプログラムされたキャラクタと一致する場合、CMFフラグがセットされ、CMIEビットをセットすることで対応する割込みが有効になっていれば、低電力モードからデバイスがウェイクアップします。
- ミュートモードが無効化された RUN モードでは（ModBus プロトコルでのブロックの終了の検出など）、受信したキャラクタ全体（8bit）がADD[7:0] 値と比較され、一致している場合はCMFフラグがセットされます。CMIEビットがセットされている場合、割込みが生成されます。

これらのビットフィールドは、受信が無効のとき（RE=0）またはUSARTが無効のとき（UE=0）のみ、書き込むことができます。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **MSBFIRST**：MSBファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット0から順に送受信されます。

1：スタートビットに続いて、データはMSB（ビット7/8）から順に送受信されます。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット17 **TXINV**：TXピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TXピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TXピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TXラインで外部インバータを使用できます。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット16 **RXINV**：RXピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RXピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RXピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RXラインで外部インバータを使用できます。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット15 **SWAP**：TX/RXピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RXピンは標準ピンアウトでの定義に従って使用されます。

1：TXおよびRXピンの機能はスワップされます。これにより、別のUARTへのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1個のストップビット

01：予約済み

10：2個のストップビット

11：予約済み

このビットフィールドは、LPUARTが無効（UE = 0）のときのみ書き込むことができます。

ビット11:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ADDM7**：7bitアドレス検出／4bitアドレス検出

このビットは、4bitアドレス検出と7bitアドレス検出の選択に使用されます。

0：4bitアドレス検出

1：7bitアドレス検出（8bitデータモード）

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

注： 7bitおよび9bitデータモードでは、アドレス検出は、それぞれ6bitおよび8bitアドレス（ADD[5:0] およびADD[7:0]）に対して行われます。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

67.7.4 LPUART制御レジスタ3（LPUART_CR3）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw		rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

ビット31:29 **TXFTCFG[2:0]**：TXFIFO閾値設定

000：TXFIFOはその深さの1/8に達します。
001：TXFIFOはその深さの1/4に達します。
110：TXFIFOはその深さの1/2に達します。
011：TXFIFOはその深さの3/4に達します。
100：TXFIFOはその深さの7/8に達します。
101：TXFIFOは空になります。
残りの組み合わせ：予約済み。

ビット28 **RXFTIE**：RXFIFO閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信FIFOがRXFTCFGでプログラムされた閾値に達すると、LPUART割込みが生成されます。

ビット27:25 **RXFTCFG[2:0]**：受信FIFO閾値設定

000：受信FIFOはその深さの1/8に達します。
001：受信FIFOはその深さの1/4に達します。
110：受信FIFOはその深さの1/2に達します。
011：受信FIFOはその深さの3/4に達します。
100：受信FIFOはその深さの7/8に達します。
101：受信FIFOはフルになります。
残りの組み合わせ：予約済み。

ビット24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TXFTIE**：TXFIFO閾値割込みの有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFOがTXFTCFGでプログラムされた閾値に達すると、LPUART割込みが生成されます。

ビット22:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **DEP**：ドライバ有効極性選択

0：DE信号はアクティブハイです。
1：DE信号はアクティブローです。
このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット14 **DEM**：ドライバ有効モード

このビットにより、DE信号によって外部トランシーバ制御を有効にできます。
0：DE機能は無効です。
1：DE機能は有効です。DE信号はRTSピンで出力されます。
このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット13 **DDRE**：受信エラー時DMA無効

0：受信エラーの場合、DMAは無効になりません。対応するエラーフラグはセットされますが、RXNEは0に保たれ、オーバーランを防ぎます。結果として、DMAリクエストはアサートされないので、エラーのあるデータは転送されず（DMAリクエストなし）、次の正しい受信データが転送されます。

1：受信エラーの後、DMAは無効化されます。対応するエラーフラグとRXNEがセットされます。エラーフラグがクリアされるまで、DMAリクエストはマスクされます。つまり、ソフトウェアは、まず、DMAリクエストを無効にするか（DMAR=0）、RXNEをクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、OREは、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNEフラグがまだセットされている間に新しいデータを受信した場合、

OREフラグはセットされず、新しく受信されたデータがLPUART_RDRレジスタの前の内容に上書きされます。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10 **CTSIE**：CTS割込み有効

0：割込みは禁止されています。

1：LPUART_ISRレジスタのCTSIF=1のときには、割込みが生成されます。

ビット9 **CTSE**：CTS有効

0：CTSハードウェアフロー制御は無効です。

1：CTSモードは有効です。データはCTS入力がネゲート（0に固定）されている場合にのみ転送されます。データの送信中にCTS入力がアサートされた場合、送信は停止前に完了します。CTSがアサートされている間にデータがデータレジスタに書き込まれた場合、CTSがネゲートされるまで送信は延期されます。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット8 **RTSE**：RTS有効

0：RTSハードウェアフロー制御は無効です。

1：RTS出力は有効です。データは受信バッファに空きがある場合にのみリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS出力がネゲートされます（0にプルされます）。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット7 **DMAT**：DMA有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMAモードは送信に有効です。

0：DMAモードは送信に無効です。

ビット6 **DMAR**：DMA有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMAモードは受信に有効です。

0：DMAモードは受信に無効です。

ビット5:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

ビット2:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EIE**：エラー割込み有効

エラー割込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（LPUART_ISRレジスタのFE=1またはORE=1またはNE=1）の場合に割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：LPUART_ISRレジスタのFE=1、ORE=1、またはNE=1の場合、割込みが生成されます。

67.7.5 LPUARTボーレートレジスタ（LPUART_BRR）

このレジスタは、LPUARTが無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **BRR[19:0]**：LPUARTボーレート分周（LPUARTDIV）

注： **LPUART_BRRレジスタに0x300未満の値を書き込むことは禁じられています。**

ただし、LPUART_BRRは ≥ 0x300でなければならず、LPUART_BRRは20bitであり、高いfck値を使用して高いボーレートを生成するときには、注意が必要です。fckは、[3 xボーレート ～ 4096 xボーレート] の範囲内でなければなりません。

67.7.6 LPUARTリクエストレジスタ（LPUART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
											w	w	w	w	

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **TXFRQ**：送信データー掃リクエスト

このビットはFIFOモードが有効なときに使用されます。FIFO全体を一掃するためにTXFRQビットがセットされます。これによってTXFEフラグ（TXFIFOエンプティ、LPUART_ISRレジスタのビット23）がセットされます。

注： FIFOモードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFOが空になるまでTXFNFフラグはリセットされます。

ビット3 **RXFRQ**：受信データー掃リクエスト

このビットに1を書き込むと、RXNEフラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット2 **MMRQ**：ミュートモードリクエスト

このビットに1を書き込むと、LPUARTはミュートモードになり、RWUフラグはリセットされます。

ビット1 **SBKRQ**：ブレーク送信リクエスト

このビットに1を書き込むと、SBKFフラグがセットされ、送信マシンが使用可能になるとすぐに、ラインでBREAKを送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアはSBKRQビットをセットする前に、TXEフラグのアサートを待つ必要があります。

ビット0 予約済みであり、リセット値に保持する必要があります。

67.7.7 LPUART割込みおよびステータスレジスタ（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0080 00C0

同じレジスタがFIFOモード有効（このセクション）でも、FIFOモード無効（次のセクション）でも使用できます。

FIFOモードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFE	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY
				r	r		r	r	r	r		r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **TXFT**：TXFIFO閾値フラグ

このビットは、TXFIFOがLPUART_CR3レジスタのTXFTCFGでプログラムされた閾値に達したとき、すなわち、TXFIFOにTXFTCFGの空き場所ができたときに、ハードウェアによってセットされます。LPUART_CR3レジスタのTXFTIEビット =1（ビット31）の場合、割込みが生成されます。

0：TXFIFOはプログラムされた閾値に達していません。

1：TXFIFOはプログラムされた閾値に達しました。

ビット26 **RXFT**：RXFIFO閾値フラグ

このビットは、RXFIFOがLPUART_CR3レジスタのRXFTCFGでプログラムされた閾値に達したとき、すなわち、受信FIFOにRXFTCFG分のデータが入ったときに、ハードウェアによってセットされます。LPUART_CR3レジスタのRXFTIEビット =1（ビット27）の場合、割込みが生成されます。

0：受信FIFOはプログラムされた閾値に達していません。

1：受信FIFOはプログラムされた閾値に達しました。

ビット25 予約済みであり、リセット値に保持する必要があります。

ビット24 **RXFF**：RXFIFOフル

このビットは、受信したデータの数がRXFIFOサイズ + 1と一致したときに（RXFIFOがフルで、LPUART_RDRレジスタに1個のデータ）、ハードウェアによってセットされます。

LPUART_CR1レジスタのRXFFIEビット =1の場合、割込みが生成されます。

0：RXFIFOはフルではありません。

1：RXFIFOはフルです。

ビット23 **TXFE**：TXFIFOエンプティ

このビットは、TXFIFOが空のとき、ハードウェアによってセットされます。TXFIFOに少なくとも1データが入ったとき、このフラグはクリアされます。TXFEフラグは、LPUART_RQRレジスタのビットTXFRQ（ビット4）に1を書き込むことによってセットすることもできます。

LPUART_CR1レジスタのTXFEIEビット =1（ビット30）の場合、割込みが生成されます。

0：TXFIFOは空ではありません。

1：TXFIFOは空です。

ビット22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値がLPUARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、LPUARTが受信できる状態であることを確認できます。

注： LPUARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

ビット21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値がLPUARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1レジスタでTE=0を書き込んだ後、TE=1を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0の最小周期を満たすために使用できます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、LPUARTがミュートモードかどうかを示します。ウェイクアップ／ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたはIDLE）は、LPUART_CR1レジスタのWAKEビットによって選択されます。

IDLEモードでのウェイクアップが選択されたとき、このビットはLPUART_RQRレジスタのMMRQビットに ”1” を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

ビット18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3レジスタのSBKRQビットに1を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタが送信されます。

ビット17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのCMCFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタのCMIE=1の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RXラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUARTはアイドルです（受信なし）。

1：受信中です。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **CTS**：CTSフラグ

このビットは、ハードウェアによってセット／リセットされます。CTS入力ピンのステータスの反転コピーです。

0：CTSラインはセットされました。

1：CTSラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット9 **CTSIF**：CTS割込みフラグ

このビットは、CTSEビットがセットされていた場合、CTS入力がトグルしたときにハードウェアによってセットされます。LPUART_ICRレジスタのCTSCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3レジスタのCTSIE=1の場合、割込みが生成されます。

0：CTSステータスラインでの変化はありません。

1：CTSステータスラインでの変化がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TXFNF**：TXFIFOノットフル

TXFIFOがフルではないとき、TXFNFがハードウェアによってセットされ、したがってデータをLPUART_TDRに書き込むことができます。LPUART_TDRレジスタへの書込み動作ごとにデータがTXFIFOに格納されます。このフラグはTXFIFOがフルになるまでセットされたままになります。TXFIFOがフルになると、このフラグはクリアされ、データをLPUART_TDRに書き込むことができないことを示します。

一掃リクエスト中、TXFIFOが空になるまで、TXFNFはリセットに維持されます。一掃リクエストを（TXFRQビットをセットすることによって）送信した後、TXFIFOに書き込む前にTXFNFフラグをチェックする必要があります（TXFNFおよびTXFEは同時にセットされます）。

LPUART_CR1レジスタのTXFNFIEビット =1の場合、割込みが生成されます。

0：データレジスタはフル／送信FIFOはフルです。

1：データレジスタ／送信FIFOはフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット6 **TC**：送信完了

このビットは、LPUART_TDRに書き込まれた最後のデータがシフトレジスタから送信されたことを示します。TCフラグの動作は次のようになります。

- TDN =0の場合、データを含むフレームの送信が完了し、TXFEがセットされたとき、TCフラグがセットされます。
- TDNがTXFIFOのデータの数と等しい場合、TXFIFOが空でTDNに達すると、TCフラグがセットされます。
- TDNがTXFIFOのデータの数より多い場合、送信するプログラム済みデータの数に達するようにTXFIFOが再び満たされるまで、TCはクリアされたままとなります。
- TDNがTXFIFOのデータの数より少ない場合、TXFIFOが空でなくても、TDNに達したときにTCがセットされます。

LPUART_CR1レジスタのTCIE=1の場合、割込みが生成されます。

TCビットは、LPUART_ICRレジスタのTCCFに ”1” を書き込むことによって、またはLPUART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

ビット5 **RXFNE**：RXFIFOノットエンプティ

RXFNEビットは、RXFIFOが空でなく、したがってデータがLPUART_RDRレジスタから読み出せるときに、ハードウェアによってセットされます。LPUART_RDRからの読出しのたびに、RXFIFOの1つの場所が解放されます。このビットはRXFIFOが空になると、クリアされます。

RXFNEフラグは、LPUART_RQRレジスタのRXFRQに1を書き込むことによってもクリアすることもできます。

LPUART_CR1レジスタのRXFNEIE=1の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1レジスタのIDLEIE=1の場合、割込みが生成されます。LPUART_ICRレジスタのIDLECFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXFNEビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLEビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、LPUARTがミュートでない場合（RWU=0）、WAKEビットによって選択されたミュートモードに関係なく、IDLEはセットされます。RWU=1の場合、IDLEはセットされません。

ビット3 **ORE**：オーバーランエラー

このビットは、RXNE="1"（FIFOモードが有効な場合はRXFF="1"）のときに、

現在シフトレジスタに受信中のデータをLPUART_RDRレジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのORECFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタでRXFNEIE=1、またはLPUART_CR3レジスタでEIE=1の場合、割込みが生成されます。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIEビットがセットされている場合、マルチバッファ通信中にOREフラグがセットされた場合、割込みが生成されます。

LPUART_CR3レジスタのOVRDISビットがセットされると、このビットは永続的に0に強制設定されます（オーバーラン検出なし）。

ビット2 **NE**：スタートビットノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICRレジスタのNFCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成するRXFNEビットと同時に出現するため、割込みを生成しません。EIEビットがセットされている場合、マルチバッファ通信中にNEフラグがセットされると、割込みが生成されます。

このエラーはLPUART_RDR内のキャラクタに関連します。

ビット1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのFECFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームをNACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR3レジスタのEIE=1の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはLPUART_RDR内のキャラクタに関連します。

ビット0 **PE**：パリティエラー

このビットは、受信モードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICRレジスタのPECFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタのPEIE=1である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： このエラーはLPUART_RDR内のキャラクタに関連します。

67.7.8 LPUART割込みとステータスレジスタ [オルタネート]（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタがFIFOモード有効（前のセクション）でも、FIFOモード無効（このセクション）でも使用できます。

FIFOモードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY
									r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値がLPUARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、LPUARTが受信できる状態であることを確認できます。

注： LPUARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

ビット21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値がLPUARTによって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1レジスタでTE=0を書き込んだ後、TE=1を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0の最小周期を満たすために使用できます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19 **RWU**：レシーバのミュートモードからのウェイクアップ

このビットは、LPUARTがミュートモードかどうかを示します。ウェイクアップ／ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたはIDLE）は、LPUART_CR1レジスタのWAKEビットによって選択されます。

IDLEモードでのウェイクアップが選択されたとき、このビットはLPUART_RQRレジスタのMMRQビットに ”1” を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUARTがSTOPモードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

ビット18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3レジスタのSBKRQビットに1を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタが送信されます。

ビット17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのCMCFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタのCMIE=1の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RXラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUARTはアイドルです（受信なし）。

1：受信中です。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **CTS**：CTSフラグ

このビットは、ハードウェアによってセット／リセットされます。CTS入力ピンのステータスの反転コピーです。

0：CTSラインはセットされました。

1：CTSラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット9 **CTSIF**：CTS割込みフラグ

このビットは、CTSEビットがセットされていた場合、CTS入力がトグルしたときにハードウェアによってセットされます。LPUART_ICRレジスタのCTSCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3レジスタのCTSIE=1の場合、割込みが生成されます。

0：CTSステータスラインでの変化はありません。

1：CTSステータスラインでの変化がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TXE**：送信データレジスタエンプティ

TXEは、LPUART_TDRレジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUART_TDRレジスタへの書込みによってクリアされます。

LPUART_CR1レジスタのTXEIEビット =1の場合、割込みが生成されます。

0：データレジスタはフル／送信FIFOはフルです。

1：データレジスタ／送信FIFOはフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット6 **TC**：送信完了

このビットは、USART_TDRに書き込まれた最後のデータがシフトレジスタから送信されたことを示します。データを含むフレームの送信が完了し、TXEがセットされたとき、TCフラグがセットされます。

LPUART_CR1レジスタのTCIE=1の場合、割込みが生成されます。

TCビットは、USART_ICRレジスタのTCCFに ”1” を書き込むことによって、またはUSART_TDRレジスタに書き込むことによって、ソフトウェアによってクリアされます。

ビット5 **RXNE**：読出しデータレジスタノットエンプティ

RXNEビットは、LPUART_RDRシフトレジスタの内容がLPUART_RDRレジスタに転送されると、ハードウェアによってセットされます。LPUART_RDRレジスタの読出しによってクリアされます。DLLRDYフラグを

RXNEフラグは、LPUART_RQRレジスタのRXFRQに1を書き込むことによってもクリアすることもできます。

RXFNEフラグは、LPUART_RQRレジスタのRXFRQに1を書き込むことによってもクリアすることもできます。

LPUART_CR1レジスタのRXNEIE=1の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1レジスタのIDLEIE=1の場合、割込みが生成されます。LPUART_ICRレジスタのIDLECFに1を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNEビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLEビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、LPUARTがミュートでない場合（RWU=0）、WAKEビットによって選択されたミュートモードに関係なく、IDLEはセットされます。RWU=1の場合、IDLEはセットされません。

ビット3 **ORE**：オーバーランエラー

このビットは、RXNE="1"（FIFOモードが有効な場合はRXFF="1"）のときに、

シフトレジスタで現在受信中のデータをLPUART_RDRレジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのORECFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタでRXNEIE=1、またはLPUART_CR3レジスタでEIE=1の場合、割込みが生成されます。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIEビットがセットされている場合、マルチバッファ通信中にOREフラグがセットされた場合、割込みが生成されます。

LPUART_CR3レジスタのOVRDISビットがセットされると、このビットは永続的に0に強制設定されます（オーバーラン検出なし）。

ビット2 **NE**：スタートビットノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICRレジスタのNFCFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成するRXNE/RXFNEビットと同時に出現するため、割込みを生成しません。EIEビットがセットされている場合、マルチバッファ通信中にNEフラグがセットされると、割込みが生成されます。

FIFOモードでは、このエラーはLPUART_RDR内のキャラクタに関連します。

ビット1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICRレジスタのFECFビットに1を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームをNACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR3レジスタのEIE=1の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： FIFOモードでは、このエラーはLPUART_RDR内のキャラクタに関連します。

ビット0 **PE**：パリティエラー

このビットは、受信モードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICRレジスタのPECFに1を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1レジスタのPEIE=1である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： FIFOモードでは、このエラーはLPUART_RDR内のキャラクタに関連します。

67.7.9 LPUART割込みフラグクリアレジスタ（LPUART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMCF	Res.
														w	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
						w			w		w	w	w	w	w

ビット31:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **CMCF**：キャラクター致フラグクリア

このビットに1を書き込むと、LPUART_ISRレジスタのCMFフラグがクリアされます。

ビット16:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **CTSCF**：CTSフラグクリア

このビットに1を書き込むと、LPUART_ISRレジスタのCTSIFフラグがクリアされます。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TCCF**：送信完了フラグクリア

このビットに1を書き込むと、LPUART_ISRレジスタのTCフラグがクリアされます。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **IDLECF**：アイドルライン検出フラグクリア

このビットに1を書き込むと、LPUART_ISRレジスタのIDLEフラグがクリアされます。

ビット3 **ORECF**：オーバーランエラーフラグクリア

このビットに1を書き込むと、LPUART_ISRレジスタのOREフラグがクリアされます。

ビット2 **NECF**：ノイズ検出フラグクリア
このビットに1を書き込むと、LPUART_ISRレジスタのNEフラグがクリアされます。

ビット1 **FECF**：フレーミングエラーフラグクリア
このビットに1を書き込むと、LPUART_ISRレジスタのFEフラグがクリアされます。

ビット0 **PECF**：パリティエラーフラグクリア
このビットに1を書き込むと、LPUART_ISRレジスタのPEフラグがクリアされます。

67.7.10 LPUART受信データレジスタ（LPUART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **RDR[8:0]**：受信データ値
受信データキャラクタを含みます。
RDRレジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図825を参照）。
パリティを有効にして受信する場合、MSBビットで読み出される値が受信したパリティビットです。

67.7.11 LPUART送信データレジスタ（LPUART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8:0 **TDR[8:0]**：送信データ値
送信されるデータキャラクタを含みます。
TDRレジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図825を参照）。
パリティを有効にして（LPUART_CR1レジスタのPCEビットに1をセット）送信しているとき、MSB（データ長に応じてビット7または8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF=1のときのみ書き込む必要があります。

67.7.12 LPUARTプリスケーラレジスタ（LPUART_PRESC）

このレジスタは、LPUARTが無効（UE=0）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PRESCALER[3:0]**：クロックプリスケーラ

LPUART入力クロックは、以下のようにプリスケーラによって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが2分周されます。
0010：入力クロックが4分周されます。
0011：入力クロックが6分周されます。
0100：入力クロックが8分周されます。
0101：入力クロックが10分周されます。
0110：入力クロックが12分周されます。
0111：入力クロックが16分周されます。
1000：入力クロックが32分周されます。
1001：入力クロックが64分周されます。
1010：入力クロックが128分周されます。
1011：入力クロックが256分周されます。
残りの組み合わせ：予約済み。

注： PRESCALERが許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は1011になります。すなわち、入力クロックは256分周されます。

67.7.13 LPUART自律モード制御レジスタ（LPUART_AUTOCR）

アドレスオフセット：0x30

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGSEL[3:0]				IDLEDIS	TRIGEN	TRIGPOL
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TDN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22:19 **TRIGSEL[3:0]**：トリガ選択ビット

セクション ：LPUARTの相互接続の説明を参照してください。

このビットフィールドに書き込めるのは、LPUART_CR1レジスタのUEビットがクリアされているときだけです。

0000：lpuart_trg0を選択します。

0001：lpuart_trg1を選択します。

.....

1111：lpuart_trg15を選択します。

注： このビットフィールドに書き込めるのは、LPUART_CR1レジスタのUEビットがクリアされているときだけです。

ビット18 **IDLEDIS**：トランスミッタ有効化後のアイドルフレーム送信無効化ビット

0：トランスミッタを有効（LPUART_CR1のTE = 1）にした後にアイドルフレームを送信します。

1：トランスミッタを有効にした後にアイドルフレームを送信しません。

注： このビットフィールドに書き込めるのは、LPUART_CR1レジスタのUEビットがクリアされているときだけです。

ビット17 **TRIGEN**：トリガ有効化ビット

0：トリガは無効です。

1：トリガは有効です。

注： このビットフィールドに書き込めるのは、LPUART_CR1レジスタのUEビットがクリアされているときだけです。

トリガが検出されると、LPUART_CR1のTEに1がセットされ、データ転送が開始されます。

ビット16 **TRIGPOL**：トリガ極性ビット

このビットフィールドに書き込めるのは、LPUART_CR1レジスタのUEビットがクリアされているときだけです。

0：トリガ は立ち上がりエッジでアクティブです。

1：トリガ は立ち下がりエッジでアクティブです。

ビット15:0 **TDN[15:0]**：TDC送信データ数

このビットフィールドでは、送信するデータの数をプログラミングできます。これに書き込めるのは、LPUART_CR1のUEがクリアされているときだけです。

67.7.14 LPUARTレジスタマップ

表689.LPUARTレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPUART_CR1 FIFOモード有効	RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]						DEDT[4:0]				Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	LPUART_CR1 FIFOモード無効	Res.	Res.	FIFOEN	M1	Res.	Res.	DEAT[4:0]						DEDT[4:0]				Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	LPUART_CR2	ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	Res.	STOP [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0		0	0								0				
0x08	LPUART_CR3	TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	Res.	EIE
	リセット値	0	0	0	0	0	0	0		0								0	0	0	0		0	0	0	0	0			0			0
0x0C	LPUART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10 ~0x14	予約済み																																
0x18	LPUART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
	リセット値																												0	0	0	0	
0x1C	LPUART_ISR FIFOモード有効	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFF	TXFF	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値								0	1	0	0		0	0	0	0						0	0		1	0	0	0	0	0	0	0
0x1C	LPUART_ISR FIFOモード無効	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	Res.	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値										0	0		0	0	0	0						0	0		1	0	0	0	0	0	0	0
0x20	LPUART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
	リセット値															0								0			0		0	0	0	0	0
0x24	LPUART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	LPUART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表689.LPUARTレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	LPUART_PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0
0x30	LPUART_AUTOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGSEL[3:0]				IDLEDIS	TRIGEN	TRIGPOL	TDN[15:0]															
	リセット値										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

68 シリアルペリフェラルインタフェース（SPI）

68.1 概要

シリアルペリフェラルインタフェース（SPI）を使用して、特定の同期プロトコルを利用して外部デバイスと通信することができます。SPIプロトコルは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタまたはスレーブとして設定することが可能で、マルチスレーブまたはマルチマスタの設定で動作することができます。マスタとして設定されたデバイスは、スレーブデバイスに通信クロック（SCK）を供給します。スレーブ選択（SS）およびレディ（RDY）信号は、具体的なスレーブとの通信をセットアップし、データフローが適切に処理されていることを確認するためだけにオプションで適用できます。デフォルトでモトローラのデータフォーマットが使用されますが、その他の特定のモードもサポートされています。

68.2 SPIの主な機能

- 3本のラインでの全二重同期転送
- 2本のラインでの半二重同期転送（双方向データライン有り）
- 2本のラインでの単方向同期転送（単方向データライン有り）
- 4から32bitまでのデータサイズ選択または8bitの倍数に固定
- マルチマスタまたはマルチスレーブモード機能
- デュアルクロックドメイン、ペリフェラルカーネルクロックはAPBバスクロックから独立
- ボーレートプリスケーラ、カーネル周波数/2まで、またはマスタモードでRCCからバイパス
- 構成と設定の保護
- マスタとスレーブの両方に対するハードウェア／ソフトウェアによるSS管理
- データ間、およびSSとデータフロー間の最小遅延を調節可能
- 設定可能なSS信号の極性とタイミング、MISOとMOSIのスワップ機能
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能（MSBファースト／LSBファーストのシフト）
- SSおよびCRCを制御するトランザクション内のデータ数をプログラム可能
- 専用の送受信フラグ（割込み機能付き）
- SPIのモトローラおよびTIフォーマットをサポート
- ハードウェアのCRC機能では、以下によってトランザクションの最後の通信の整合性を確認できます。
  - 送信モードでCRC値の追加
  - 受信モードでCRCエラーの自動チェック
- 動作モードに応じて、データオーバーラン、CRC エラー、データアンダーラン、モードフォールトおよびフレームエラーの場合に、割込み機能のあるエラー検出
- 8bitの内蔵RxおよびTx FIFOの倍数（FIFOサイズはインスタンスによって異なる）
- 設定可能なFIFO閾値（データパッキング）
- システムDMAコントローラでデータストリームを処理する機能
- スレーブのアンダーラン状態での設定可能な動作（カスケードサーキュラバッファ）

- STOPモードでの自律機能（トランザクションフローと必要なクロック分配の処理）およびSTOPからのウェイクアップ機能
- データフローを処理できるようになったことをスレーブデバイスに知らせる、オプションのステータスピンRDY

68.3 SPIの実装

表690に、SPIの実装を示します。インスタンスには、すべてまたは一部の機能が適用されます。

表690.SPIの機能

SPIの機能	SPI1およびSPI2（全機能）	SPI3（限定機能）
データサイズ	4～32bitで設定可能	8/16bit
CRC計算	CRC多項式の長さを5～33bitに設定可能	CRC多項式の長さ9/17bit
FIFOのサイズ	16 × 8bit	8 × 8bit
データ数の制御	最大65536	最大1024、残りデータカウンタ（CTSIZE）なし
ウェイクアップ付きのSTOPモードでの自律機能	あり	あり
ウェイクアップ付きのSTANDBYモードでの自律機能	なし	いいえ

注： **STOPおよびSTANDBYモードを終了するためのインスタンスの機能について、詳しくは表 696 : SPIのウェイクアップおよび割込みリクエストおよび表 100 : 動作モードに応じた機能を参照してください。**

68.4 SPIの機能詳細

68.4.1 SPIのブロック図

SPIにより、MCUと外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用のSPI割込みを使用することで、通信を管理することができます。SPIの主要要素およびそれらの相互作用を図837に示します。

図837. SPI のブロック図

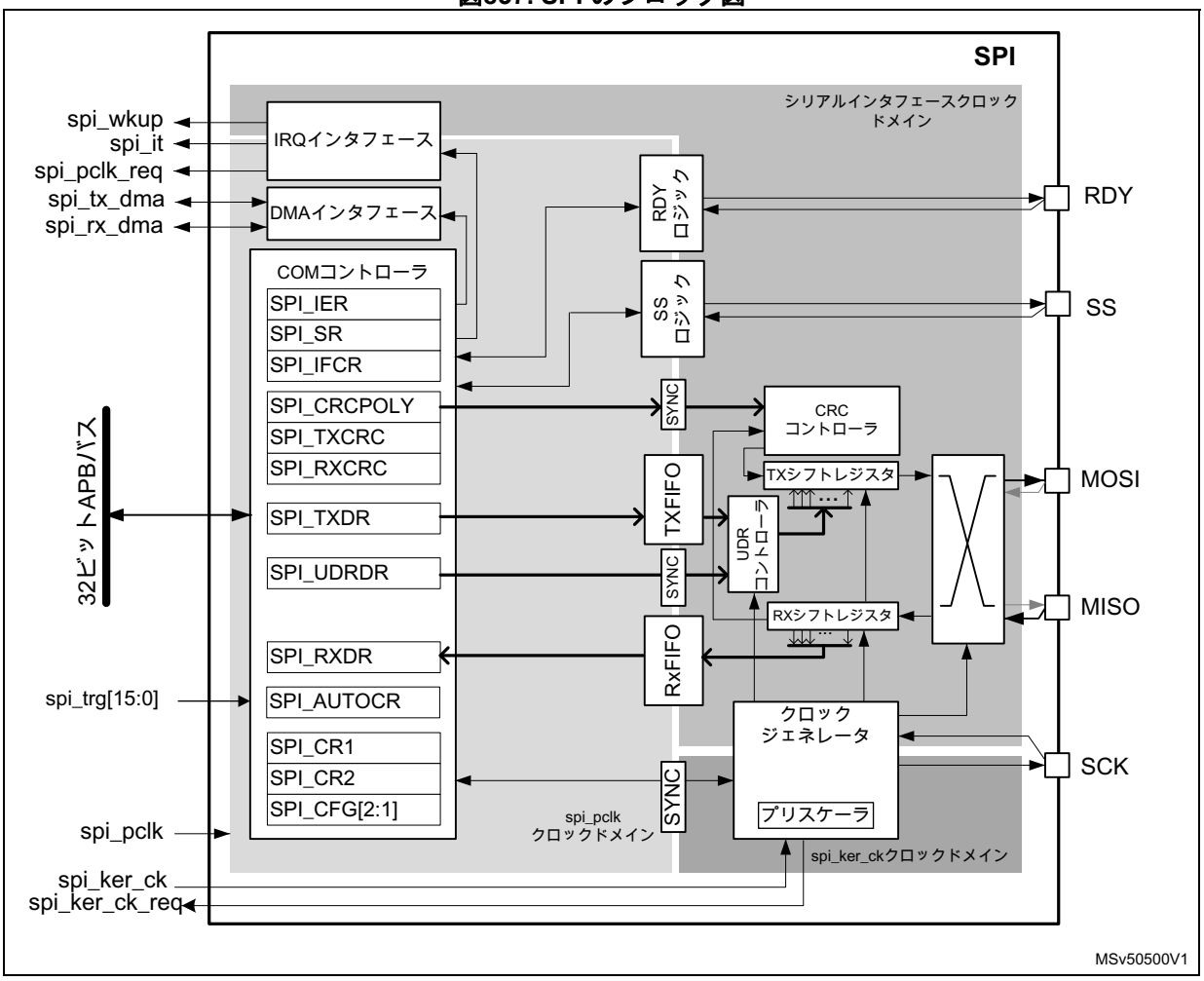


図837の簡易体系図には、以下の完全に独立した3つのクロックドメインが示されています。

- **spi_pclk** クロックドメイン
- **spi_ker_ck** カーネルクロックドメイン
- シリアルインタフェースクロックドメイン

これらのドメイン間のすべての制御およびステータス信号は厳密に同期されています。これらのクロック信号間の周波数の比に関しては特定の制約はありません。ユーザは、データのアンダーランやオーバーランイベントを避けるために、データフロー速度と互換性のある周波数比を考慮することが必要です。

spi_pclk クロック信号はペリフェラルバスインタフェースにクロックを供給します。SPIレジスタへアクセスが必要なとき、アクティブになっていなければなりません。

スレーブモードで動作しているSPIは、外部マスタSPIデバイスによって提供される外部SCK信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。このため、SPIスレーブは、**spi_pclk** および **spi_ker_ck** 信号が非アクティブになっている場合でもデータの送受信ができます。結果的に、シリアルインタフェースクロックドメイン内で動作する特定のスレーブのロジックでは、追加のトラフィックを正しく設定する必要があります（たとえば、アンダーランやオーバーランを評価する場合。詳細については、セクション68.5.2を参照）。バスがアイドルになると、これを行うことはできません。一部のケースでは、スレーブはクロックジェネレータの動作も必要とします（セクション68.5.1を参照）。

SPIがマスタとして機能する場合は、プリスケーラによって分周可能またはオプションでバイパス可能な、クロックジェネレータによってシリアルインタフェースクロックに供給するための通信中に有効なRCCからの **spi_ker_ck** カーネルクロックが必要です。信号はその後、SCKピンを介してスレーブに供給され、また内部的にマスタのシリアル·インタフェース·ドメインにも供給されます。

68.4.2 SPIピンおよび内部信号

最大5本のI/OピンがSPI外部デバイスとの通信専用に使用されます。

- **MISO**：マスタイン/スレーブアウトデータ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使用されます。
- **MOSI**：マスタアウト/スレーブインデータ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使用されます。
- **SCK**：このピンは、SPIマスタではシリアルクロックの出力に、SPIスレーブでは入力に使用されます。
- **SS**：スレーブ選択用のピンです。このピンは、SPIおよびSSの設定に応じて、以下のいずれかに使用できます。
  - 個々の通信用スレーブデバイスを選択する
  - データフレームを同期させる
  - 複数のマスタ間での競合を検出する

  詳細については、セクション68.4.7を参照してください。
- **RDY**：スレーブのFIFOの占有状況、つまり、データフローを損なうことなくトランザクションを続行できるスレーブの可用性を示すオプションのステータスピン。マスタは、現在の通信の一時的な中断を制御するためにこれをチェックできます。

SPIバスを使用することで、1つのマスタデバイスと1つ以上のスレーブデバイスとの間で通信することができます。バスは2本以上の線から成り、1本はクロック信号用、その他はデータの同期転送用です。その他の信号はオプションで、SPIノード間でのデータ交換とそれらの通信制御管理に応じて、追加することができます。

SPI入出力ピンおよび内部信号のリストについては、表691と表692を参照してください。

表691.SPI の入出力ピン

ピン名	I/Oタイプ	説明
MISO(1)	入力／出力	マスタデータ入力/スレーブデータ出力
MOSI (1)	入力／出力	マスタデータ出力/スレーブデータ入力
SCK	入力／出力	マスタクロック出力/スレーブクロック入力
SS	入力／出力	マスタ出力/スレーブ選択入力
RDY	入力／出力	SPIマスタの入力/スレーブFIFOのステータス占有率の出力

1. MOSI およびMISO ピンの機能はスワップできます。これらの方向は、SPI双方向半二重モードで異なる場合があります。

SPI入出力信号の説明

表692.SPI内部入力／出力信号

信号名	信号タイプ	説明
spi_pclk	入力	SPIクロック信号はペリフェラルバスインタフェースにクロックを供給
spi_ker_ck	入力	SPIカーネルクロック
spi_ker_ck_req	出力	SPIカーネルクロックリクエスト
spi_pclk_req	出力	SPIクロックリクエスト
spi_wkup	出力	SPIはウェイクアップ割込みを提供
spi_it	出力	SPIグローバル割込み
spi_tx_dma	入力／出力	SPI送信DMAリクエスト
spi_rx_dma	入力／出力	SPI受信DMAリクエスト
spi_trg[15:0]	入力	SPIトリガ

SPIの相互作用の説明

表693.SPIの相互作用（SPI1およびSPI2）

信号名	トリガソース
spi_trg0	gpdma1_ch0_tc
spi_trg1	gpdma1_ch1_tc
spi_trg2	gpdma1_ch2_tc
spi_trg3	gpdma1_ch3_tc
spi_trg4	exti4
spi_trg5	exti9
spi_trg6	lptim1_ch1
spi_trg7	lptim2_ch1
spi_trg8	comp1_out
spi_trg9	comp2_out(1)
spi_trg10	rtc_alra_trg
spi_trg11	rtc_wut_trg
spi_trg12	-
spi_trg13	-
spi_trg14	-
spi_trg15	-

1. STM32U535/545ではCOMP2が使用できないので、この接続はありません。

表694.SPIの相互接続（SPI3）

信号名	トリガソース
spi_trg0	lpdma1_ch0_tc
spi_trg1	lpdma1_ch1_tc
spi_trg2	lpdma1_ch2_tc
spi_trg3	lpdma1_ch3_tc
spi_trg4	exti4
spi_trg5	exti8
spi_trg6	lptim1_ch1
spi_trg7	lptim3_ch1
spi_trg8	comp1_out
spi_trg9	comp2_out(1)
spi_trg10	rtc_alra_trg
spi_trg11	rtc_wut_trg
spi_trg12	-
spi_trg13	-
spi_trg14	-
spi_trg15	-

1. STM32U535/545ではCOMP2が使用できないので、この接続はありません。

68.4.3 SPI通信の一般的側面

SPIを使用することで、MCUは対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2または3本の線（ソフトウェアSS管理あり）、あるいは3または4本の線（ハードウェアSS管理あり）が使われます。通信は常にマスタによって開始され制御されます。マスタはSCKラインにクロック信号を供給し、SSライン（ハードウェアによって管理される場合）によって通信相手のスレーブを選択したり、同期させたりします。マスタとスレーブ間のデータはMOSIおよび/またはMISOライン上を流れます。

68.4.4 マスタとスレーブの1対1の通信

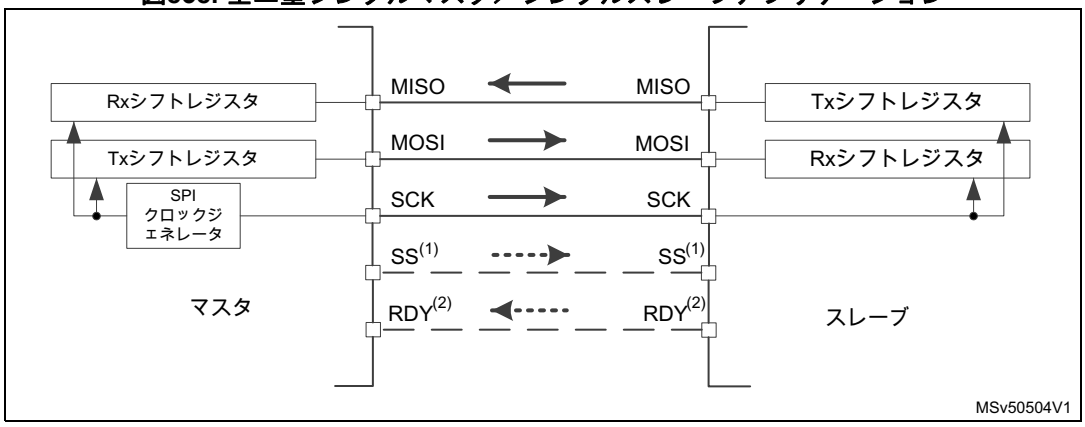
通信の流れは、全二重（3線）モード、半二重（2線）モード、単方向（2線）モードの3つのモードのうちの1つを使用します。SS信号は、シングルマスタ-スレーブ設定ではオプションであり、2つの通信ノード間で接続されないことがしばしばあります。しかしながら、SS信号は、この設定でデータフローを同期させるのに役立ち、特定のSPIモード（たとえば、TIモード）ではデフォルトで使用されます。

次のオプションのRDY信号は、スレーブ側でトランザクションされたすべてのデータの正しい管理を保証するのに役立ちます。

全二重通信

SPIは、デフォルトで全二重通信に設定されます（SPI_CFG2レジスタのビットCOMM[1:0] = 00）。この設定では、マスタおよびスレーブのシフトレジスタは、MOSIピンとMISOピンの間で2本の単方向ラインを介してリンクされます。SPI通信の間、データはマスタから供給されるSCKクロックのエッジに同期してシフトされます。マスタは、送信すべきデータをMOSIライン経由でスレーブに送信し、同時にMISOライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図838. 全二重シングルマスタ／シングルスレーブアプリケーション

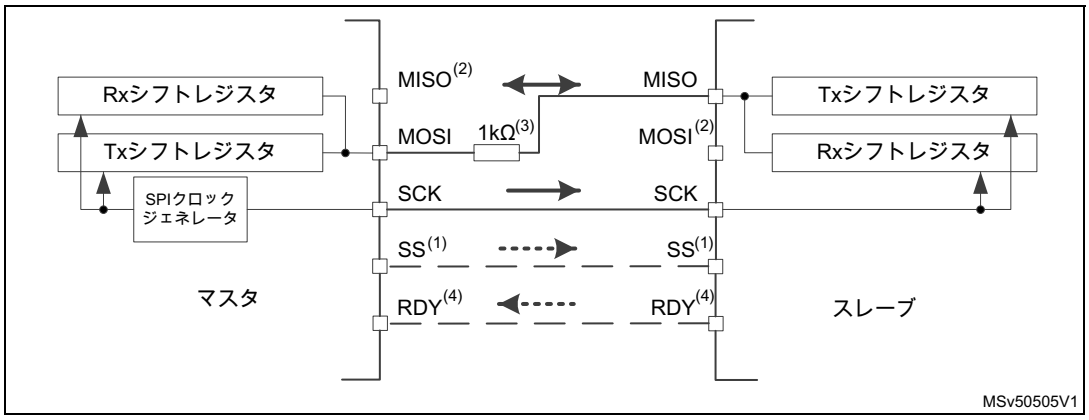


1. SPI インタフェースを機能させるための SS ピンの相互接続の適用は必須ではありません（詳細については、セクション 68.4.7を参照）。
2. スレーブによって供給されたRDY信号は、オプションでマスタによって読み出すことができます。

半二重通信

SPI_CFG2レジスタでCOMM[1:0] = 11にセットすることで、SPIは半二重モードで通信できます。この設定では、1本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データはSCKクロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方がSPI_CR1レジスタのHDDIRビットを使用して相互に選択された転送方向となります。通信の方向を変えるときはSPIを無効にする必要があるので、注意してください。この設定では、マスタのMISOピンとスレーブのMOSIピンは、他のアプリケーションで使用でき、GPIOとして機能します。

図839. 半二重シングルマスタ／シングルスレーブアプリケーション



1. SPI インタフェースを機能させるための SS ピンの相互接続の適用は必須ではありません（詳細については、セクション 68.4.7を参照）。
2. この設定では、マスタのMISOピンとスレーブのMOSIピンは、GPIOとして使用できます。
3. 双方向モードで動作している2つのノード間で通信方向が同時に変更されなかった場合は、重大な事態が発生する可能性があります。前のトランスミッタが逆の値をライン上に保持し続ける中、新しいトランスミッタは共通のデータラインにアクセスします（値はSPI設定と通信データによって決まります）。前のトランスミッタがデータ方向設定を変えるまで、ノードは一時的にラインの逆の出力レベルと競合する場合があります。このモードのMISOピンとMOSIピンの間に直列抵抗を挿入して競合する出力を保護し、その間の電流の流れを制限することをお奨めします。
4. スレーブによって供給されたRDY信号は、オプションでマスタによって読み出すことができます。

単方向通信

SPIは、SPI_CFG2レジスタのCOMM[1:0] フィールドを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは1ラインのみです。残りのMISOピンまたはMOSIピンのペアは通信には使用されず、標準のGPIOとして使用できます。

- **送信専用モード：COMM[1:0] = 01**

  送信専用モードのマスタは、TxFIFOに準備できたデータがあり、マスタの転送が進行中である限り、クロックを生成します。

  送信専用モードのスレーブは、SCKピンにクロックを受信し、SSピン（またはソフトウェア管理の内部信号）がアクティブである限り、データを送信します（セクション68.4.7を参照）。

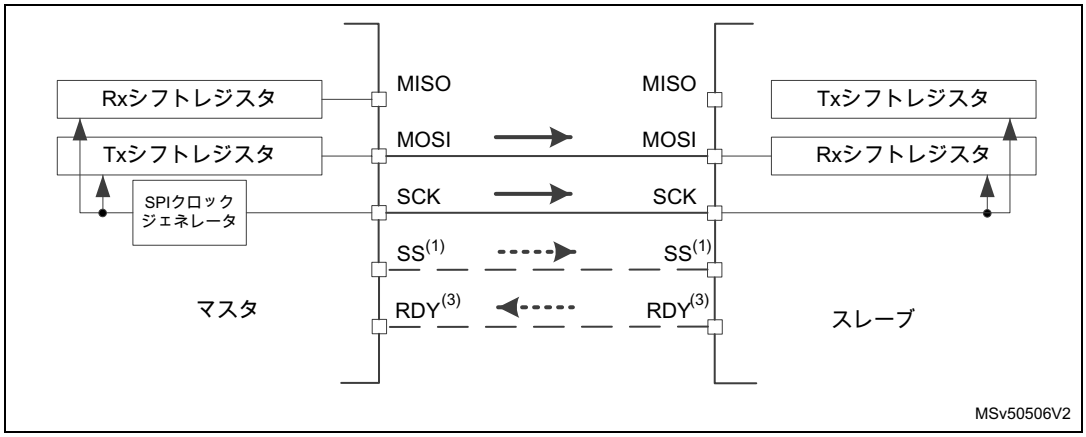
- **受信専用モード：COMM[1:0] = 10**

  マスタモードでは、MOSI出力が無効化され、GPIOとして使用できます。SPIが有効で、SPI_CR1レジスタのCSTARTビットがセットされている限り、クロック信号が生成され続けます。SPI_CR1レジスタのCSUSPビットをセットすることによってソフトウェアが明示的に停止を要求することによって、またはSPI_CR1のMASRXビットがセットされているとき、RxFIFOがフルになったときに自動的にクロックが停止されます。

  スレーブ設定では、MISO出力が無効化され、ピンをGPIOとして使用できます。スレーブ選択信号がアクティブな間は、スレーブはMOSIピンからデータを受信し続けます（セクション68.4.7を参照）。

注： **マスタおよびスレーブモードのどちらであっても、SPI_CFG2レジスタのIOSWPビットの値を変更することによって、送信専用のデータピンを受信専用のデータピンで置き換えること、またその逆もできます（このビットはSPIが無効なときにのみ変更できます）。**
すべての単方向通信は、トランザクション方向の設定を固定した（双方向モードは有効でHDDIRビットは決して変更されない）半二重通信の変形型か、未使用データラインと対応するデータフローが無視された場合は全二重制御で置き換えることができます。

図840. 単方向シングルマスタ／シングルスレーブのアプリケーション（送信専用モードのマスタ/受信専用モードのスレーブ）

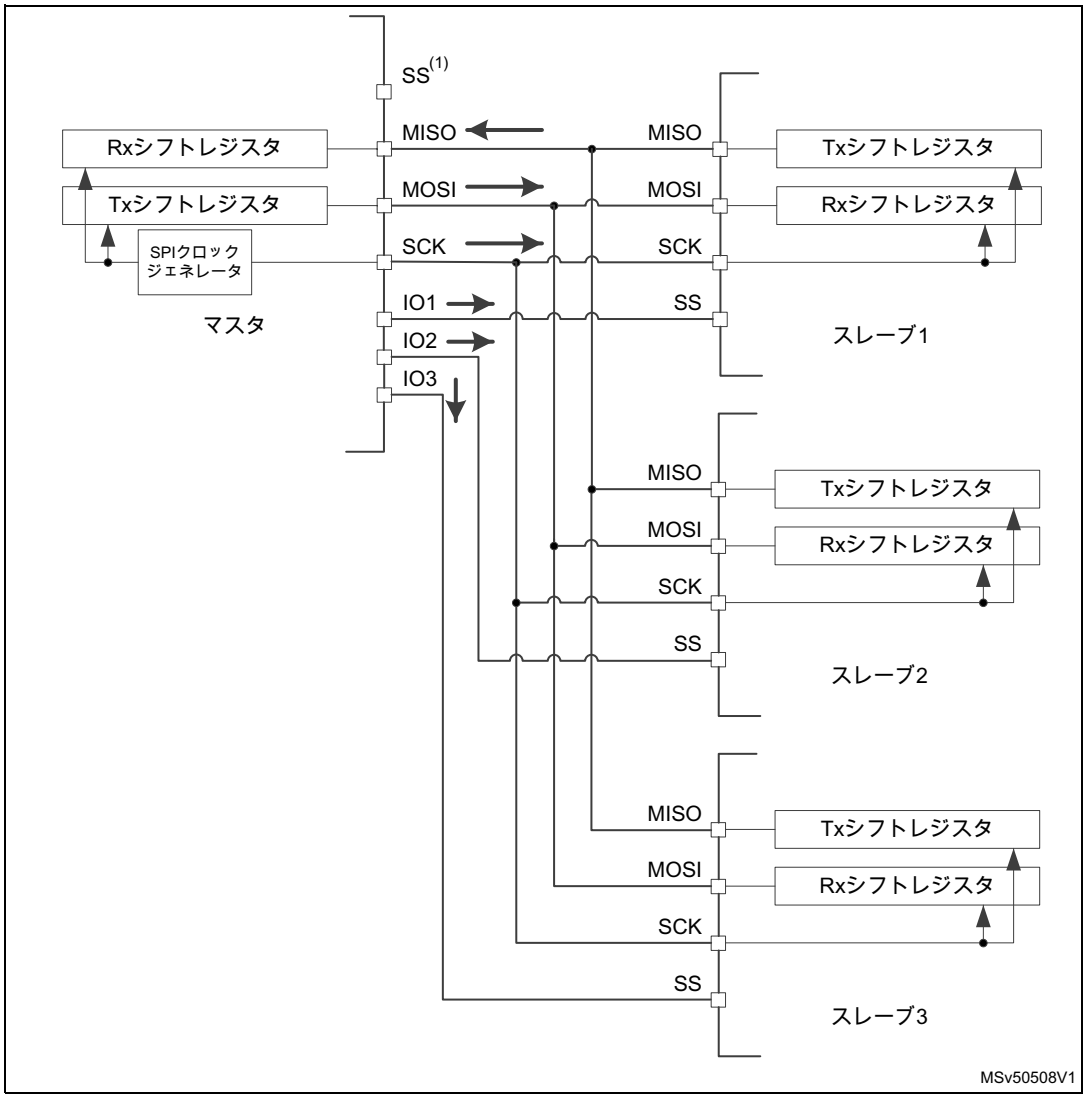


1. SPIインタフェースを機能させるためのSSピンの相互接続は必須ではありません（セクション68.4.7 を参照）。
2. この設定では、両方のMISOピンをGPIOとして使用できます。
3. スレーブによって供給されたRDY信号は、オプションでマスタによって読み出すことができます。

68.4.5 標準のマルチスレーブ通信

2つ以上の独立したスレーブがある設定の場合、マスタはスター接続形態を使用し、GPIOピンを使用して、各スレーブのチップセレクトラインを別々に管理します（図841を参照）。マスタは、スレーブのSS入力に接続されているGPIOをローレベルにプルダウンすることによって、スレーブの1つを個別に選択する必要があります（一度に共通MISOライン上のデータを制御できるのは1つのスレーブのみです）。これを行うことにより、マスタと選択されたスレーブ間の通信が確立します。単純さ以外に、この接続形態の長所は、すべての通信セッションがシングルマスタ-スレーブペア内で別々に行われるので、特定のSPI設定がスレーブごとに適用できることです。オプションとして、スレーブから情報を読み取る必要がないときは、マスタは同じ情報を複数のスレーブに送信することができます。

図841. スター接続形態で接続されたマスタと3つの独立したスレーブ



1. マスタの1つのSSピンのハードウェア出力機能ではこの接続形態をサポートしていません（ソフトウェア制御されている一連のGPIOに置き換えられます）。ピンでのSPI AFの設定は避けてください（詳細については、セクション68.4.7を参照）。
2. アプリケーションが、一度に 1 つの SS アクティブ信号だけがマスタによって供給されていることを確認できない場合、MISOラインの外部プルアップでMISOピンをオープンドレイン設定にして、スレーブの相互出力における競合を防ぐことをお勧めします。そうでない場合、外部レジスタなしでプッシュプル設定を適用できます（I/Oオルタネート機能入力/出力（GPIO）選択を参照）。
3. RDY信号は、オプションでスレーブからマスタによって読み出すことができます。

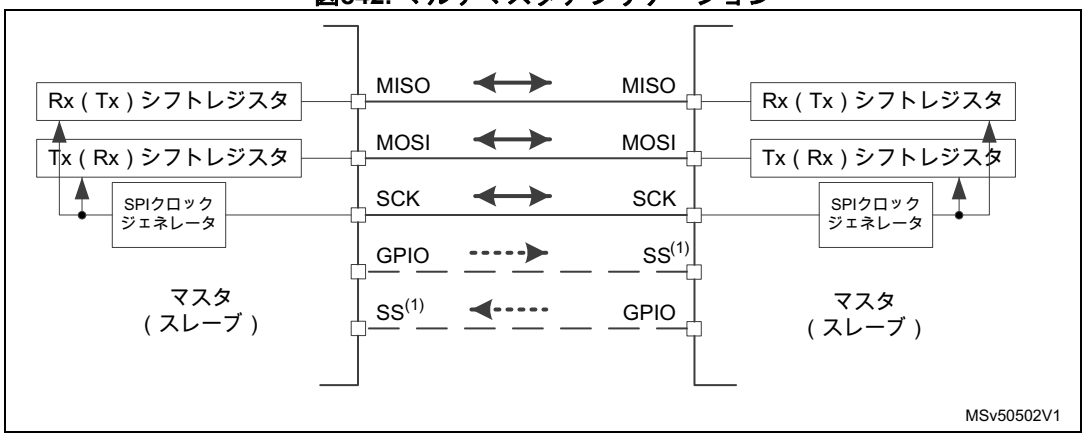
68.4.6 マルチマスタ通信

SPIバスがもともとマルチマスタ機能向けに設計されていない限り、同時にバスを支配しようとする2つのノード間での競合の可能性を検出する内蔵機能を使用できます。この検出では、SSピンをハードウェア入力モードに設定して使用します。共通のデータラインで一度に出力を適用できるノードは1つだけであるため、このモードで動作している2つ以上のSPIノードの接続は、不可能です。

ノードが非アクティブであるとき、両ノードはデフォルトでスレーブモードのままとなります。1つのノードがバスでの制御を得る場合、自身をマスタモードに切り替えて、専用のGPIOピンを介して他のノードのスレーブ選択入力にアクティブレベルを適用します。セッションが完了すると、アクティブスレーブ選択信号が解除され、バスを一時的に支配するノードがパッシブスレーブモードに戻り、次のセッション開始を待機します。

両ノードのマスタリングリクエストが同時に上がった場合、バス競合イベントが発生します（モードフォールトMODFイベントを参照）。いくつかの簡単なアービトレーションプロセスを適用できます（たとえば、両方のノードに適用する事前定義された異なるタイムアウトによって次の試行を延期します）。

図842. マルチマスタアプリケーション



1. SS ピンは両ノードともハードウェア入力モードで設定されます。アクティブレベルにすることで、パッシブノードがスレーブとして設定されるため、MISOライン出力の制御が可能になります。
2. RDY信号はこの通信では使用されません。

68.4.7 スレーブ選択（SS）ピンの管理

スレーブモードでは、SSは標準の「チップセレクト」入力として機能し、スレーブをマスタと通信させます。マスタモードでは、SSは出力としても入力としても使用できます。入力として使用する場合、SSはマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は1つのスレーブのスレーブ選択信号を駆動させることができます。SS信号は内部的に管理することも（SS入力のソフトウェアによる管理）、SS入力および出力の両方がSSピンに関連付けられているときは外部的に管理することもできます（ハードウェアによるSS管理）。この入力／出力外部信号（SSピンに存在）のどちらのレベルがアクティブとみなされるのかをSSIOPビットのセットによって設定することができます。SSレベルはSSIOPと等しい場合、アクティブとみなされます。

ハードウェアまたはソフトウェアのスレーブ選択管理は、SPI_CFG2レジスタのSSMビットを使用して以下のように設定することができます。

- **ソフトウェアSS管理（SSM = 1）**：この設定では、スレーブ選択情報はSPI_CR1レジスタのSSIビットの値によって内部で駆動されます。外部SSピンは他のアプリケーションで使用できます（GPIOまたはその他のオルタネート機能として）。
- **ハードウェアSS管理（SSM = 0）**：この場合、2通りの設定が可能です。次のどちらの設定を使用するかは、SS出力設定（SPI_CFG2レジスタのSSOEビット）によって決まります。
  - **SS出力有効（SSOE = 1）**：この設定は、MCUがマスタとしてセットされている場合にのみ使用します。SSピンはハードウェアによって管理されます。この機能はCSTARTおよびEOT制御に接続されています。そのため、マスタはSS出力を正しく制御するために、適切なTSIZE>0設定を適用する必要があります。SPI AFがSSピンで適用されていない場合でも（その場合は標準のGPIOとして使用可能）、デフォルトのSS入力レベルを保証するためにSSOE = 1は保持し、マルチマスタ接続形態にのみ適用可能なマスタSS内部ロジック入力時のモードフォールト評価を防いでください。

    a) SSOM = 0で、SP = 000のとき、SS信号は、マスタの転送が開始されると（CSTART = 1）すぐにアクティブレベルになり、EOTフラグがセットされるか、送信がサスペンドされるまでアクティブに保たれます。

    b) SP = 001のとき、TIモードで定義されたパルスが生成されます。

    c) SSOM = 1、SP = 000、およびMIDI > 1のとき、SSはデータフレーム間で非アクティブにパルス出力され、MIDI値から1を減じた値（1～14）によって定義されるSPIクロック周期の間、非アクティブに保たれます。

    d) SS入力は、モードフォールトを防ぐため、マスタ内部で強制的に非アクティブ状態になります。

  - **SS出力無効（SSM = 0、SSOE = 0）**：

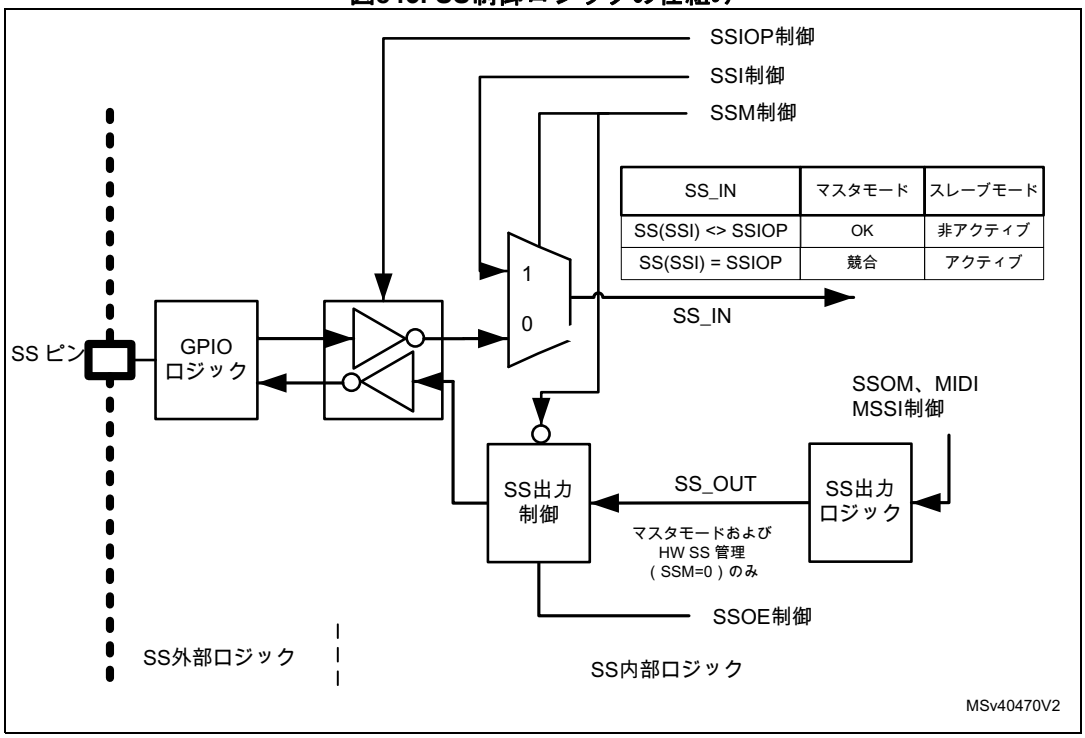
    a) マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、SSピンがアクティブレベルに引き込まれた場合、SPIはマスタモードのフォールト状態に入り、SPIデバイスは自動的にスレーブモードに再設定（MASTER = 0）されます。

    b) スレーブモードでは、SS ピンは標準の「チップセレクト」入力として機能し、SS ラインがアクティブレベルの間はスレーブが選択されます。

注： **モードフォールト状態で自動的にスレーブモードに切り替える目的は、データおよびクロックラインでの競合の可能性を避けるためです。SPEはこの状態で自動的にリセットされるため、RxとTxの両方のFIFOは一掃され、現在のデータは失われます。**

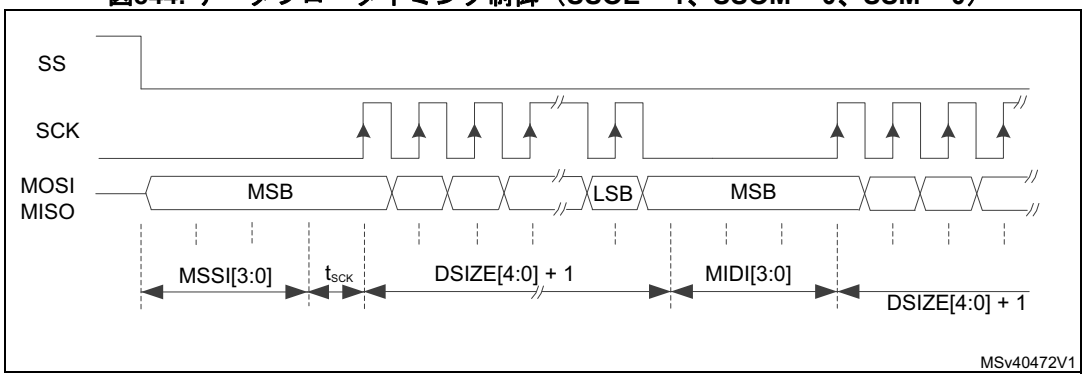
SPIスレーブがハードウェアSS管理モードで有効化されたとき、SSがアクティブレベルであることがわかった場合でも、すべてのトラフィックが無視されます。これらは、スレーブをマスタと同期させるSS信号の開始（非アクティブからアクティブレベルへのトランザクション）がスレーブによって検出されるまで無視されます。こういうわけで、外部SSピンが固定されているときは、ハードウェア管理モードは使用できません。SSのソフトウェア管理ではそのような保護はありません。そこで、この場合に限って、バスにトラフィックが無くて、SCK信号が転送間でアイドル状態レベルにあるときに、SSIビットを変更すべきです。

図843. SS制御ロジックの仕組み



ハードウェア出力SS制御が適用されるとき（SSM = 0、SSOE = 1）、MIDI[3:0] およびMSSI[3:0] ビットフィールドの設定によって、データフレーム間のSS信号のタイミングを制御し、（SSとクロック開始を分離するために）すべてのトランザクションの初めに余分の遅延時間を挿入することができます。これは、スレーブが正しいデータ処理のための十分な余裕を得るために流れの速度を落とす必要があるときに有用です（図844を参照）。

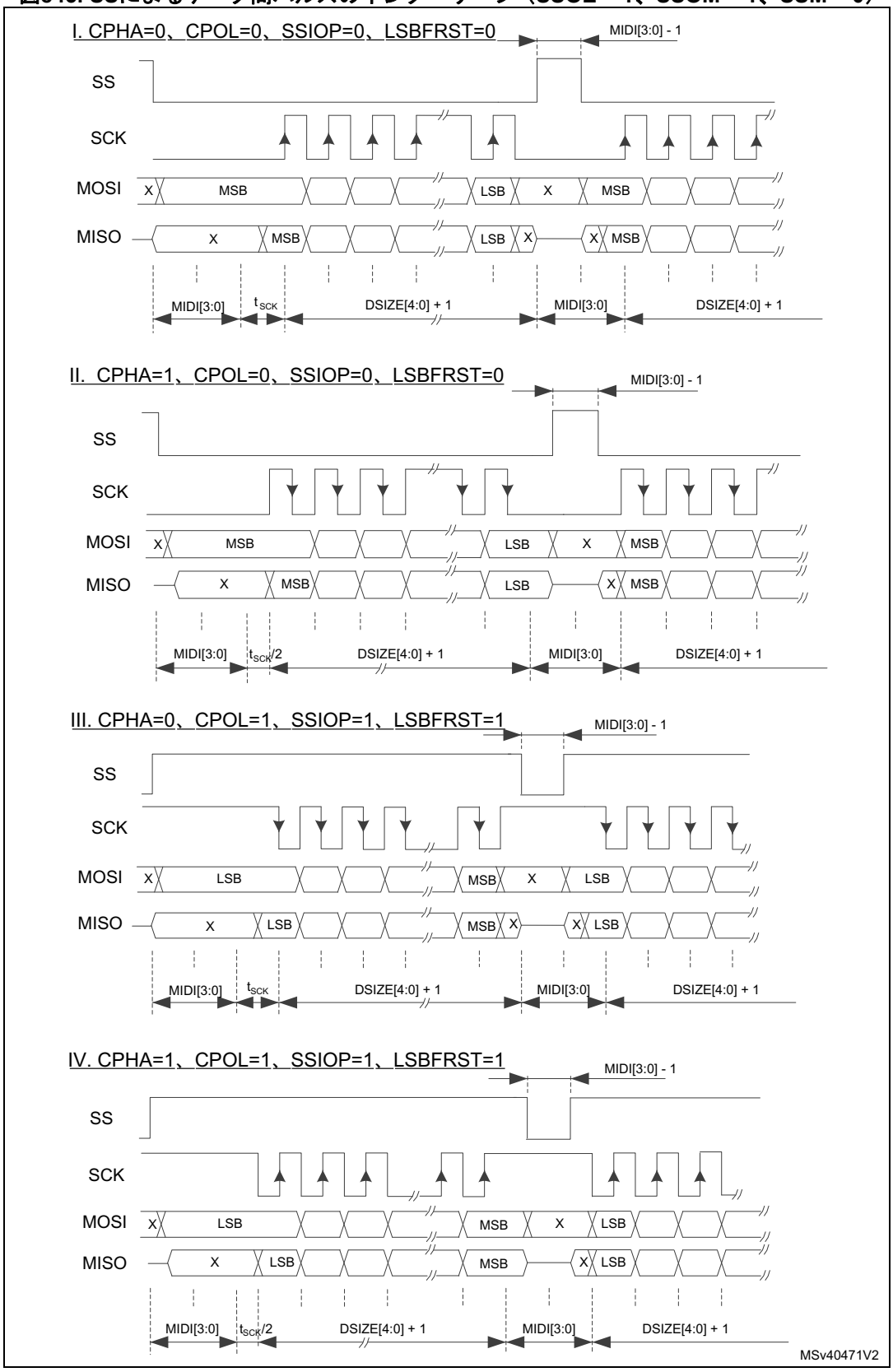
図844. データフロータイミング制御（SSOE = 1、SSOM = 0、SSM = 0）



1. MSSI[3:0] = 0011、MIDI[3:0] = 0011（MIDI[3:0] = 0のときは、SCKフローは連続です）。
2. CPHA = 0、CPOL = 0、SSIOP = 0、LSBFRST = 0。

さらに、ビットSSOM = 1の設定により、パルスを挿入する十分なスペースがあればデータフレーム間にパルスをインターリーブする特定のモードが呼び出されます（MIDI[3:0] はSPIの1周期より大きく設定する必要があります）。設定のいくつかの例を図845に示します。

図845. SSによるデータ間パルスのインターリーブ（SSOE = 1、SSOM = 1、SSM = 0）



1. MSSI[3:0] = 0010、MIDI[3:0] = 0010。
2. インターリーブ･パルスの幅がMIDI[3:0] > 1の場合のデータ間のSSインターリーブは、常に（MIDIパラメータによって定義された）フレーム間に設けられたギャップより、1 SCK周期少なくなります。MIDIがセットされている場合、フレームは1つのSCK周期で分けられますが、SSにインターリーブ･パルスは現れません。

68.4.8 レディピン（RDY）の管理

データを処理するためのスレーブ機能のステータスは、RDYピンで確認できます。デフォルトで、ローレベルはスレーブのトランザクションの準備ができていないことを示します。理由として、TxFIFOが空であること、RxFIFOがフルであること、またはSPIが無効化されていることがあげられます。信号のアクティブレベルはRDIOPビットで選択できます。ノット･レディ･ステータスを示していても、マスタがスレーブとの通信を続けたり、開始したりする場合、トランザクションはほぼ間違いなく大失敗します。

RDY出力を制御するロジックはやや複雑で、TSIZEおよびDSIZEの設定と密接につながっています。RDYの反応は、TxFIFOがほぼ空になったり、フレームのトランザクション中にRxFIFOがほぼフルになったりすることについて悲観的で、影響を受けやすいです。この悲観的なロジックは、RDYがアクティブなままであれば、TxFIFOが完全に空になっても、RxFIFOが完全に占有されても、トランザクションの最後に抑制されます。目的は、データ破損を防ぐことと、次にトランザクションされたデータをまた安全に処理できるようになるまで、一時的にトランザクションをサスペンドする必要があることをマスタに間に合うように通知することです。RDY信号入力がマスタ側で有効な場合、マスタはスレーブがノット･レディ･ステータスを示したときに通信をサスペンドします。これは、マスタが処理中のフレームのトランザクションを完了するのを防ぎます。単にスレーブのTxFIFOが空になるか、RxFIFOに次のデータが書き込まれるか読み出されるまで完全に満たされます（フレームにかかわらず、制約なしでも完了できます）。スレーブにTSIZE = 0設定を適用すると、スレーブがトランザクションの終了をまったく評価しなくなるため、問題になる場合があります（これにより、最後のデータの送信時にノット･レディ･ステータスが抑制されます）。その場合、ユーザはRxFIFOを解放し、ソフトウェアによってスレーブ側でTxFIFOに追加の（ダミーも可）データを書き込んでノットRDY信号を解除し、STマスタをブロック解除することで、しばしばフレームの中央でサスペンドされる通信で続けられるようにします。

マスタでRDYが使用されない場合は、無効にする必要があります（RDIOM = 0）。その後、マスタの内部ロジックがすでにレディ状態のスレーブステータスのシミュレーションを実施します。この場合、RDIOPビット設定は意味を持ちません。

マスタとスレーブの両側で、クロックドメインとRDYロジックの評価の間の同期により、RDYピン機能は信頼できません。また、データフレームのサイズが8bitより短く設定されている場合は使用できません。

68.4.9 通信フォーマット

SPI通信中は受信と送信の操作が同時に行われます。シリアルクロック（SCK）は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従い、正しく同期される必要があります。

クロックの位相および極性の制御

SPI_CFG2レジスタのCPOLビットとCPHAビットを使用することによって、考えられる4つのタイミングの関係をソフトウェアで選択できます。CPOL（クロック極性）ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOLがリセットされると、SCKピンはローレベルのアイドル状態になります。CPOLがセットされると、SCKピンはハイレベルのアイドル状態になります。

CPHAビットがセットされると、SCKピンの2番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOLビットがリセットされていれば立下りエッジ、CPOLビットがセットされていれば立上りエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。CPHAビットがリセットされている場合、SCKピンの1番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOLビットがセットされていれば立下りエッジ、CPOLビットがリセットされていれば立上りエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。

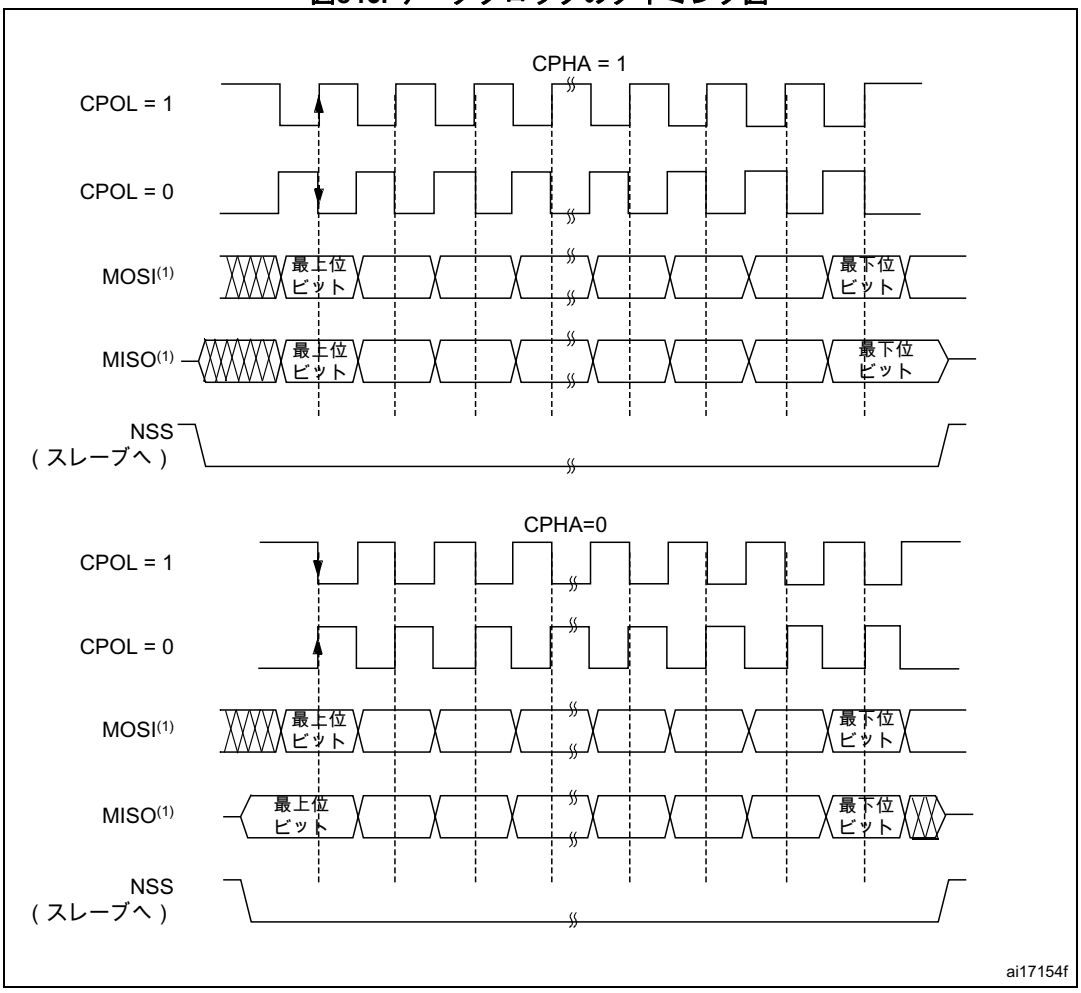
CPOL（クロック極性）ビットとCPHA（クロック位相）ビットの組み合わせによって、データキャプチャのクロックエッジ（図846の点線）を選択できます。

図846は、CPHAビットとCPOLビットの4つの組み合わせによるSPI全二重転送を示しています。

注： **CPOLまたはCPHAビットを変更する前に、SPEビットをリセットすることによって、SPIを無効にする必要があります。**

SCKのアイドル状態は、SPI_CFG2レジスタで（CPOL = 1ならSCKピンのプルアップ、CPOL = 0ならプルダウンによって）選択された極性に一致する必要があります。

図846. データクロックのタイミング図



1. データビットの順序はLSBFRSTビットの設定値に依存します。

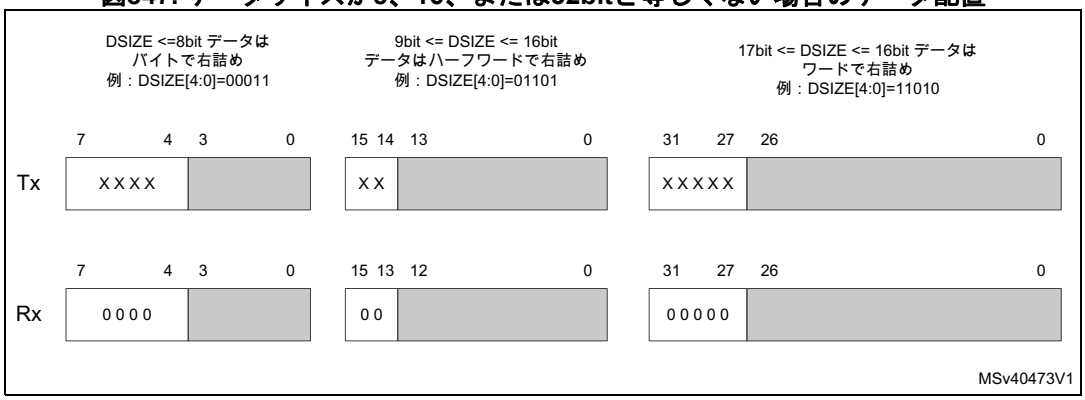
データフレームフォーマット

SPI_CFG2レジスタのLSBFRSTビットの値に応じて、SPIシフトレジスタを設定することで、シフトをMSBファーストまたはLSBファーストに設定することができます。

全機能がセットされたインスタンスで、データフレームサイズはSPI_CFG1レジスタのDSIZE[4:0]ビットを使用して選択できます。4bitから 32bit長までの間にセットでき、この設定は送信と受信の両方に適用できます。SPI_TXDR/SPI_RXDRレジスタにアクセスすると、データフレームはバイト（データがバイトに適合する場合）、ハーフワード、またはワードのいずれかに常に右詰めされます（図847を参照）。

アクセスが設定されたデータサイズの倍数の場合、データパッキングは自動的に適用されます。通信中、データフレーム内のビットのみがクロック供給され、送信されます。

図847. データサイズが8、16、または32bitと等しくない場合のデータ配置



注： **最小データ長は4bitです。4bit未満のデータ長が選択された場合は、4bitのデータフレームサイズに固定されます。**

機能が制限されているインスタンスの場合、DSIZE[4:3] のビット値に従って、データサイズは8bitの倍数から最大データ長に固定されます（インスタンスによる）。SPI_TXDRまたはSPI_RXDRへのアクセスが拡大されている場合（設定されたデータサイズの倍数）、データパッキングは自動的に適用されます。

68.4.10 SPIの設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関する章を参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

1. 適切なGPIOレジスタの書込み：適用されている場合は、MOSI、MISO、SCK、SS、およびRDYピンでGPIOオルタネート機能を設定します。
2. SPI_CFG1とSPI_CFG2レジスタに書き込んで、前のSPIが有効な場合、以下を除く「予約済み以外」のビットおよびビットフィールドすべてに適切な値を設定します。
   a) SSOM、MASRX、SSOE、RDIOM、MBR[2:0]、BPASS、MIDI[3:0]、MSSI[3:0] ビットはマスタモードでのみ考慮されます。MSSI[3:0] ビットはSSOEビットがセットされたときのみ有効になり、RDIOPビットはマスタモードでRDIOMビットがセットされていない場合は有効になりません。スレーブがTIモードで設定されると、MBR[2:0] 設定も考慮されます。
   b) UDRCFGはスレーブモードでのみ考慮されます。
   c) CRCSIZE[4:0] はCRCENがセットされた場合に必要です。
   d) CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、SSM、RDIOP、RDIOM、MSSI、およびMIDIはTIモードでは不要です。

e) SPI_CFG2レジスタでAFCNTRビットがセットされると、ペリフェラルが有効にもかかわらず、すべてのSPI出力は関連するGPIOピンに伝播され始めます。したがって、その後のSPI_CFG1およびSPI_CFG2レジスタの設定変更は、これらのピンの信号のレベルに影響を与えます。

3. 転送の長さを選択するためにSPI_CR2レジスタへ書き込みます。それが未知の場合は、TSIZEはゼロにプログラムしなければなりません。
4. 必要に応じてCRC多項式とCRC計算を設定するために、SPI_CRCPOLYと、SPI_CR1レジスタのTCRCINI、RCRCINI、およびCRC33_17bitに書き込みます。
5. DMAストリームが使用されている場合は、DMAレジスタにSPI TxおよびRx専用のDMAストリームを設定します（DMAを使用した通信の章を参照）。
6. 必要に応じて、SPI_CR1レジスタでSSI、HDDIR、およびMASRXを設定します。
7. 設定の保護が必要な場合は（安全のため）、SPI_CFG1レジスタのIOLOCKビットをプログラムします。

68.4.11 SPIの有効化

マスタがクロックを送信する前に、SPIスレーブを設定し、有効にすることを推奨します。ただし、トラフィックがバスで進行中に設定および有効化手順を実行しても悪影響はありません。このとき、SS信号はスレーブでハードウェアによって管理されるか、SS信号がソフトウェアによって管理されているときにスレーブのソフトウェアによって非アクティブ状態で保持されるものとします（セクション68.4.7を参照）。スレーブトランスミッタのデータレジスタには、マスタがクロック供給を開始する前に、送信データが含まれていなければなりません。SPIスレーブがSSによって選択される前に、SCK信号は選択した極性に対応するアイドル状態レベルに安定させる必要があります。そうしないと、続くトランザクションの同期が取れなくなってしまうことがあります。

SPIスレーブがハードウェアSS管理モードで有効化されたとき、SSがアクティブレベルであることがわかった場合でも、すべてのトラフィックが無視されます。これらは、スレーブをマスタと同期させるSS信号の開始（非アクティブからアクティブレベルへのトランザクション）がスレーブによって検出されるまで無視されます。こういうわけで、外部SSピンが固定されているときは、ハードウェア管理モードは使用できません。SSのソフトウェア管理ではそのような保護はありません。そこで、この場合に限って、バスにトラフィックが無くて、SCK信号が転送間でアイドル状態レベルにあるときに、SSIビットを変更すべきです。

全二重（または送信専用モード）のマスタは、SPIが有効でCSTARTビットがセットされ、TXFIFOが空でない場合、または次のTXFIFOへの書込み時に通信を開始します。

マスタ受信専用モードでは、SPIが有効化され、CSTARTビットがセットされた後、マスタは通信を開始し、クロックが動作を開始します。

DMAの扱いについては、セクション68.4.15を参照してください。

68.4.12 SPIのデータ送受信手順

データ通信フォーマットの設定は、柔軟性の高いサイズの確実なデータ数をセッション内で転送する（トランザクション）という基本原理に従います。一方、オプションで、データ処理はSPIデータレジスタの1回のアクセス（データパッキング）に効率的に累積したり、一貫したより大きなデータパケットにデータが収集されている場合は、このようなサービスのシーケンスにグループ化したりすることもできます。データ処理サービスは、FIFOパケットの占有イベントに基づいています。このために、完全なデータパケットは、専用パケットフラグによってのみ処理する必要があります。

このセクションの次の詳細な内容についてより的確に理解するには、最初に以下の項目の設定の影響と意味を把握してください。

データサイズ（DSIZE）- データフレームを定義します（1つのデータフレームのビット数をセットします）。

FIFO閾値（FTHLV）- データパケットを定義し、1つのデータパケットのデータフレーム数をセットして、ソフトウェアまたはDMAによって処理されるSPIデータレジスタのパケットの占有イベントの発生をセットします。

データアクセス – パケットイベント時に転送データがアプリケーションとSPI FIFOの間にある場合のSPIデータレジスタの内容の処理方法。これは、パケットサイズの設定によって決まります。オプションとして、レジスタに1回アクセスすることで（データパッキング）、または一連のそのようなアクセスにより（大きなデータパケットを処理する場合）、複数のデータを処理できます。

FIFOサイズ – 使用可能なデータを吸収する容量またはスペース。これは、データサイズと、データがスペース内でどのように圧縮され、構成されるかという内部ハードウェアの効率によって決定します。FTHLV設定では、最低2つのデータパケットを格納するというFIFOの機能を考慮する必要があります。

トランザクションサイズ（TSIZE） – 複数のデータパケットサービスでカバーされる、全体的なトランザクションセッションに関わるデータフレームの総数を定義します。この数字をパケットサイズに調整する必要はありません（最後の調整されていないデータパケットの処理は、TSIZEが適切にプログラムされている場合にサポートされます）。

RxFIFOおよびTxFIFOによるデータ処理

すべてのSPIデータトランザクションは複数バイト（N x 8bit）で構成された内蔵FIFOを通過します。FIFOのサイズ（N）は、製品およびペリフェラルのインスタンスに依存します。これにより、SPIは連続フローで動作できるようになり、またデータフレームサイズが短い場合や、割込み／DMA遅延時間が長すぎる場合にオーバーランを防ぐことができます。方向ごとにそれぞれTxFIFOとRxFIFOと呼ばれる固有のFIFOがあります。

FIFOの内容の処理は、専用のFIFOパケットの占有フラグ（TXP、RXP、またはDXP）によって発生するデータ･パケット･イベントの処理に基づきます。フラグの発生は、データ変換モード（二重、単方向）、データフレームサイズ（フレーム内のビット数）、およびデータパケットでのデータの構成方法に依存します。パケット･イベントの頻度は、FIFOの閾値を定義して、データをパケットにまとめた場合は大幅に低減できます。これにより、1つのFIFO占有パケットイベントに基づいて、1つのSPIデータレジスタアクセスまたはシステム性能の消費量が低いそれらのシーケンスによって、パケットでグループ化されたいくつかのデータフレームを効率的に処理できます。ユーザは、レジスタの読出しまたは書込みに適用された具体的なCPU命令を強制するために、データレジスタのアドレスを割り当てることで、アクセスタイプを制御できます。これにより、アクセスは8bit、16bit、または32bitにすることができますが、少なくとも1つのデータフレームに常にアクセスする必要があります。パケットサイズ（FTHVL）およびデータサイズ（DSIZE）の設定は、イベント時に1つのデータパケットの処理を適用して完了するためだけに、適用されたデータレジスタのアクセス（1つのアクセスかそれらのシーケンスが適用されているかにかかわらず）と常にバランスが取れた状態で保持することが重要です。FIFOの占有フラグの原理、発生、およびクリア機能は、DMA、割込み、またはポーリングのいずれが適用されていても共通です。

SPI_RXDRレジスタへの読出しアクセスからは、RxFIFOに保管された、まだ読み出されていない一番古い値が返されます。SPI_TXDRへの書込みアクセスでは、送信キューの最後にTxFIFOに書き込まれるデータを保管します。

SPI_RXDRレジスタへの読出しアクセスはRXPイベントで管理する必要があります。このフラグは、受信がアクティブになっているとき、少なくとも1つの完全なデータパケット（SPI_CFG1レジスタのFTHLV[3:0] ビットによってレシーバの閾値として定義されている）が受信FIFOで利用可能になったときにハードウェアによってセットされます。RXPは、SPI_RXDRがソフトウェアまたはDMAによって読み出されたとき、RxFIFOに利用可能なデータが、完全な1つのパケットよりも少なくなるとすぐにクリアされます。

RXPは、RXPIEビットがセットされている場合は割込みを、および/またはRXDMAENビットがセットされている場合はDMAリクエストをトリガします。

RXPフラグがセットされると、アプリケーションはSPIデータレジスタの読出しを所定回数行って1つのデータパケットの内容をダウンロードします。完全なデータパケットがダウンロードされると、アプリケーションソフトウェアまたはDMAはRXPの値をチェックして受信FIFO内に他のパケットが保留されてないかを調べ、もしあれば、RXPが0になるまでそれらをパケットごとにダウンロードします。Nをバイト単位のFIFOのサイズとして、RxFIFOが格納できる最大データフレーム数は、N個（フレームサイズ ≤ 8bitの場合）、N/2個（8bit < フレームサイズ ≤ 16bitの場合）、N/3個（16bit < フレームサイズ ≤ 24bitの場合）、またはN/4個（データフレームサイズ > 24bitの場合）です。

受信の最後に、FTHLVレベルに達しておらず、いくらかのデータがRxFIFO内でまだ利用可能になっていて、RXPがセットされないことがあります。この場合、FIFO内に残っている受信データフレームの数がSPI_SRレジスタのRXWNEおよびRXPLVLフィールドによって示されます。これは、転送サイズとパケットサイズが整合していない場合に、転送で受信した最後のデータの数が設定パケットサイズを完全に満たすことができないときに発生します。それにもかかわらず、アプリケーションソフトウェアはそれまでの完全なデータパケットに使用された、RxFIFOからの標準の数の読出しを問題なく行うことができます。矛盾のないデータ（完了したデータフレーム）のみがRxFIFOから取り出され、冗長な読み出し（または不完全なデータ）は0となります。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの受信を予知することや、RxFIFOから取り出すべき読出し所要数を計算することの負担から解放されます。

同じように、送信するデータフレームの書込みアクセスはTXPイベントで管理されます。このフラグは、送信がアクティブになっているとき、少なくとも1つの完全なデータパケット（SPI_CFG1レジスタのFTHLV[3:0] ビットで定義されている）をアプリケーションが送信FIFOにプッシュするのに十分なスペースができたときに、ハードウェアによってセットされます。TXPは、TxFIFOがソフトウェアおよび／またはDMAによって充填されるとすぐにクリアされます。次の完全なデータパケット用に現在利用できるスペースはなくなります。これによって、データがTxFIFOから放出され、新しいパケットがフレームごとに格納されるとき、TXP信号が振動することになります。少なくとも1つのデータフレームを格納するだけの十分なスペースがないとき（TXPイベントは無視されます）、TXTFがセットされているとき、またはSPIが無効になっているとき、TxFIFOへの書込みは無視されます。

TXPは、TXPIEビットがセットされている場合は割込みを、および/またはTXDMAENビットがセットされている場合はDMAリクエストをトリガします。TXPIEマスクは、TXTFフラグがセットされると、ハードウェアによってクリアされます。

TXPフラグがセットされると、アプリケーションはSPIデータレジスタへの書込みを所定回数行って1つのデータパケット全体の内容をアップロードします。新しい完全なデータパケットがアップロードされると、アプリケーションソフトウェアまたはDMAはTXPの値をチェックしてTxFIFOに他のパケットをプッシュできるかどうかを調べ、もしできるならば、TXPが0になるまで、それらをパケットごとにアップロードします。

転送サイズとパケットサイズが整合していない場合、転送の最後のデータの数が設定パケットサイズより短いことがあります。それにもかかわらず、アプリケーションソフトウェアはそれまでのパケットに使用された標準の数のデータレジスタ書込みを問題なく行うことができます。矛盾のないデータのみがTxFIFOにプッシュされ、冗長な書込みは破棄されます。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの送信を予知することや、最後のデータをTxFIFOにプッシュするための書込み所要数を計算することの負担から解放されます。最後のデータの場合、現在の転送を完了するために残りのデータを格納するのに十分なスペースがTxFIFOにあれば、SPIによってTXPイベントがアサートされます。

TXPイベントとRXPイベントの両方を割込みによってポーリングまたは処理できます。全二重モードではTXPとRXPの共通のイベントとしてDXPビットを監視することができます。

DXPフラグがセットされると、アプリケーションはSPIデータレジスタへの書込みを所要回数行って送信用の1つのデータパケット全体の内容をアップロードし、その後、SPIデータレジスタから同じ数

の読出しを行い、1つのデータパケットの内容をダウンロードします。1つのデータパケットがアップロードされ、1つがダウンロードされると、アプリケーションソフトウェアまたはDMAはDXPの値をチェックして他のパケットがプッシュおよびポップできないか順に調べ、もしできれば、DXPが0になるまでそれらをパケットごとにアップロード/ダウンロードします。

DXPIEビットがセットされている場合、DXPは割込みをトリガします。DXPIEマスクは、TXTFフラグがセットされると、ハードウェアによってクリアされます。

DXPは、全二重通信においてデータのアップロード/ダウンロードの性能を最適化するために有用であり、SPIの転送をサポートするのに必要な割込み回数またはDMAシーケンスを削減するか、その結果、特にSPIがSTOPモードで動作しているとき、CPUの帯域幅およびシステム電力の要求を最小化します。

リレーがDXP割込み専用の場合、TXPサービスが意図的に遅延され、TXPとRXPのイベントが共通手順で処理される場合に、ユーザはそのような簡略化の欠点を考慮する必要があります。これは、TXPイベントが受信RXPイベントの前に行われるためで、これにより通常は最後のフレームの前のトランザクションをTXPで処理してTxFIFOを完全に空にすることができます。それ以外の場合、マスタは連続SCKクロックの流れを供給できず、スレーブはアンダーラン条件に対応することも可能です。考えられるソリューションとして、セッションを開始する前にTxFIFOをいくつかのデータパケットで事前に書き込むこと、またトランザクションの最後にあるEOTによってTXTFイベント後に受信したすべてのデータを処理すること（トランザクションの最後にTXTFがDXP割込みを抑制するため）があげられます。CRC計算が有効である場合、トランザクションの最後のEOTにのみ基づいて、RxFIFOでCRCフレームに対応するため、追加のスペースありで計算する必要があります。

データ交換を管理するもう一つの方法は、DMAを使用することです（セクション68.4.15を参照）。

RxFIFOがフルのときに次のデータを受信した場合、オーバーランイベントが発生します（セクション68.5.2のOVRフラグの説明を参照）。オーバーランイベントは割込みによってポーリングまたは処理できます。

これは、MASRX = 0のときにスレーブモードまたはマスタ受信モードで発生することがあります。マスタレシーバでMASRXビットがセットされている場合、RxFIFOがフルになったとき、生成されたクロックが自動的に停止し、したがってオーバーランは回避できます。

RxFIFOとTxFIFOの両方の内容は、SPIが無効（SPE = 0）のときには一掃された状態に保たれ、アクセスできません。

トランザクションの処理

いくつかのデータフレームを単一シーケンスに渡してメッセージを完成させることができます。TSIZEに格納された値によって、メッセージ内のデータ数量を処理することができます。原則として、メッセージのトランザクションはCSTARTビットのセットによってSPIが有効になった時に開始し、要求された数量のデータがトランザクション処理された時に終了します。トランザクションの終わりに、CRCおよびハードウェアSS管理（適用される場合）が制御されます。内部ステートマシンを適切に再開するには、設定を変更しない場合でも、次のトランザクションが始まる前にSPIを無効にして再度有効化することを強く推奨します。

CSTARTがセットされる時、TSIZEがゼロに維持されている場合、終わりのないトランザクションが開始されます（転送サイズの制御は適用されません）。終わりのないトランザクションの間、FIFOの閾値と整合が取れているトランザクションされたデータ数のみがサポートされます。データ数（またはパケットへのグループ化）が予測不可能な場合、ユーザは、データフレームごとに、アプリケーションまたはDMAによって処理される独自のパケットイベントが発生するようにするために、FIFOの閾値設定（パケットサイズ）をシングルデータ（FTHLV = 0）に保つ必要があります。トランザクションは、CSTARTビットをクリアするCSUSPによっていつでも中断できます。SPIは、そのようなソフトウェアの中断後は常に無効にし、次のトランザクションが開始する前に再度有効化する必要があります。

送信が有効な場合、シーケンスは開始し、マスタ側のTxFIFOに何らかのデータが存在する間、続行します。TxFIFOがエンプティになるまで、マスタによってクロック信号が永続的に供給され、その後停止して追加のデータを待ちます。

受信専用モード、半二重（COMM[1:0] = 11、HDDIR = 0）、または単方向（COMM[1:0] = 10）モードでは、SPIが有効化され、CSTARTビットによってトランザクションがリリースされると、マスタによってシーケンスが開始されます。マスタによってクロック信号が供給されますが、この信号はマスタがSPIまたは受信専用モードを無効／サスペンド状態にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。ソフトウェア制御によってSPI_CR1レジスタのCSUSPビットに1を書き込むことで、またはMASRX=1でRxFIFOがフルになったときに自動的に、もしくはこの信号が適用されている場合はRDYステータス時に、受信がサスペンドされます（セクション68.4.8を参照）。また、TSIZEにプログラムされたフレームの数が完了した時にも、受信は自動的に停止します。

マスタのレシーブ専用モードを無効にするには、最初にSPIをサスペンドする必要があります。SPIがサスペンドされると、現在のフレームが完了してから、設定を変更します。

注意： **マスタモードで、受信が中断せずに進行しているときにSPEビットがクリアされると、現在のフレームを完了することなくクロックが停止し、RxFIFOが一掃されます。**

マスタはあらゆるトランザクションを連続モードで供給できる（SCK信号は連続的）一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。スレーブにRDY信号オプション機能がある場合、マスタは通信の流れを制御するために、スレーブによって発行されたRDY信号を監視できます。RDYピンを使用しない場合、スレーブは常にマスタと通信できる状態であるとみなされます。

必要に応じて、マスタは通信速度を下げ、より低速のクロックか、またはMIDI[3:0] ビットの設定によって十分な遅延を含む分離したフレームやデータセッションを供給する必要があります。あるいは、スレーブがデータの準備のために十分な余裕を持てるように、トランザクションの開始を延期するMSSI[1:0] の設定によって初期遅延時間を提供する必要があります。スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されることに注意してください。スレーブがDMAを使用することが望ましく、特に、データフレームが短く、FIFOがバイトでアクセスされ、そしてSPIバスが高速の場合に有効です。

スレーブのトランスミッタノードからSPI通信のフローに何らかのソフトウェア制御を加えるためには、SPI_UDRDR（SPIアンダーランデータレジスタ）に書き込んだ特定の値を使用することができます。スレーブ側で、TxFIFOが空になると、この値が次のデータとして自動的に送出され、マスタレシーバ側のソフトウェアで解釈されます（単に読み捨てられるか、ソフトウェアによってマスタレシーバをサスペンドするためにXOFFのようなコマンドとして解釈されます）。

マルチスレーブのスター接続形態では、一度に有効にできるのは出力データの1つのスレーブのみです。マスタとの通信用に選択されたスレーブは、マスタとの通信を開始する前に、SS入力のアクティブレベルへの変化を検出する必要があります。単一のスレーブシステムではSSによってスレーブを制御する必要はありませんが、スレーブを各データ転送シーケンスの開始と同期させるために、ここにもパルスを供給することをお勧めします。SSは、ソフトウェアとハードウェアの両方で管理できます（セクション68.4.7を参照）。

68.4.13 SPIの無効化

SPIを無効にするには、本項に記載されている無効化手順に従ってください。

マスタモードでは、この手順はペリフェラルクロックが停止し、システムが低電力モードに入る前に行うことが重要です。そうしないと、進行中のトランザクションが破壊されることがあります。

スレーブモードでは、**spi_pclk** および **spi_ker_ck** クロックが停止したとき、通信の終了またはデータサービスリクエスト状態に至るまで、SPI通信は中断することなく続行できます。**spi_pclk** は一般

にシステムをSTOPモードに設定することによって停止できます。詳細については、RCCのセクションを参照してください。

全二重または送信専用モードでは、マスタは、送信するデータの供給を停止した時点でいかなるトランザクションも終了することができます。この場合、クロックは最後のデータトランザクション後に停止します。最後のデータフレームが送信されるのを待つために、TXCフラグをポーリングする（またはEOTIE = 1によって割込みを有効にする）ことができます。

マスタが受信専用モードになっている場合、ペリフェラルを停止するためには、CSUSPビットを1にセットして、まずSPI通信をサスペンドする必要があります。

SPIがサスペンドされたとき、受信されているが、まだ読み出されていないデータはRxFIFOに格納されたままになっています。

SPIは、そのようなソフトウェアの中断後は常に無効にし、内部ステートマシンを適切に再開する必要があります。

SPIが無効化されると、RxFIFOが一掃されます。未読のデータが失われないようにするためには、SPIを無効化するとき、残っているすべてのデータ（SPI_SRレジスタのRXP、RXWNEおよびRXPLVLフィールドに示されています）を読み出してRxFIFOが空であることを確認する必要があります。

標準的な無効化手順は、送信セッションが（完全に）終わったかどうかをチェックするために、EOTおよび／またはTXCステータスをポーリングすることに基づきます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- マスタがSS信号をSPIに関連しないGPIOで処理する場合（マルチスレーブのスター接続形態の場合など）で、スレーブに適切なSSパルスの終わりを提供する必要がある場合、または
- 最後のデータフレームまたはCRCフレームのトランザクションがまだペリフェラルバスで進行している間に、DMAまたはFIFOからのトランザクションのストリームが完了した場合

TSIZE>0の場合、EOTとTXCの信号は等しいため、EOTのポーリングは、バス動作の終了を確認するために任意のSPI通信モードで信頼できます。TSIZE = 0の場合、適用されたSPIモードとデータフローの終了に従ってTXC、SUSP、またはFIFOの占有フラグを確認する必要があります。

マスタモードでの正しい無効化手順を以下に示します（受信専用モードが使用されている場合を除く）。

1. TXC = 1および/またはEOT = 1（送信するデータがなくなり、最後のデータフレームが送信される）まで待ちます。CRCが使用されている場合、ブロックの最後のデータが処理された後にCRCが自動的に送信されます。この場合、CRCフレームが完了した時にTXC/EOTがセットされます。送信がサスペンドされるときは、ソフトウェアはCSTARTビットがクリアされるまで待つ必要があります。
2. すべてのRxFIFOデータを読み出します（RXWNE = 0およびRXPLVL = 00になるまで）。
3. SPIを無効にします（SPE = 0）。

マスタ受信専用モードの場合の正しい無効化手順を以下に示します。

1. EOTまで待つか、またはSPIをサスペンド（CSUSP = 1）することによって受信フローを中断します。
2. 受信フローがサスペンドされたら、SUSP = 1（最後のデータフレームが処理される）まで待ちます。
3. すべてのRxFIFOデータを読み出します（RXWNE = 0およびRXPLVL = 00になるまで）。
4. SPIを無効にします（SPE = 0）。

スレーブモードでは、SPIを無効にすると、処理中のデータが失われます。

I/Oの制御

SPIが無効化されるとすぐに、AFCNTR設定に応じて関連する有効化されたAF出力をデバイスによって駆動できます。アクティブな出力制御が適用されていて（AFCNTR = 1）、SPIがちょうど無効化（SPE = 0）されたときに、有効化された出力はSPI制御信号と関連付けられ（マスタのSSとSCK、スレーブのRDY）、直ちに非アクティブレベルにトグルすることができます（それぞれマスタのSSIOPおよびCPOLの設定、およびスレーブのRDIOPに応じて）。それどころか、データライン出力（マスタではMOSI、スレーブではMISO）は、実際のTxFIFOの内容に応じてそのレベルが即座に変更される可能性があり、その結果、バス上の最新のトランザクションビットの値が無効になり、保証されなくなる可能性があります。必要に応じて、ユーザはデータラインでの適切なデータホールド時間に注意し、最後のデータトランザクションが完了した直後に高速SPIが無効になることを回避する必要があります。

注： **最新ビットの安定性がサンプリングクロックエッジを通じて設計保証されているにもかかわらず、一部のデバイスでは、サンプリング中にこのデータビットの安定性の間隔を均等に拡張する必要がある場合があります。これは、たとえばEOTイベント発生とSPI無効化操作の間に、ソフトウェアディレイをわずかに挿入することで実行できます。**

68.4.14 データパッキング

ユーザの観点から、互いに重なり合う以下の2つのデータパッキングの方法があります。

- データをTxFIFOに書き込む、またはRxFIFOから読み出すときのアクセスタイプ
  データサイズが、SPI_TXDRまたはSPI_RXDRレジスタで実行されるアクセス単位より倍率が小さい場合、複数のデータを1回のアクセスによって効率的にプッシュまたはフェッチすることができます。
- 1回のソフトウェアサービス中に扱われるデータの数
  データをフレームごとに別々に取り扱う代わりに、データをパケットにグループ化し、もっぱらデータパケットの内容全体でFIFOサービスを累積するほうが便利です。ユーザはFIFO閾値の設定によってパケットを定義できます。そうして、すべてのFIFO占有イベントはこの閾値レベルに関連し、要求されるサービスは割込みやウェイクアップ機能による適切なフラグによって通知されます。

データフレームサイズが1バイト（8bit以下）におさまる場合、SPI_RXDR/SPI_TXDRレジスタで16bitまたは32bitの読出しまたは書込みアクセスが実行されたときに、自動的にデータパッキングが使用されます。この場合、多重データフレームパターンが並行して処理されます。最初に、SPIはアクセスしたワードのLSBに保管されたパターンを使用して動作します。次に、MSBに保管された残りデータを使用して動作します。

図848に、全機能がセットされたインスタンスでの、データパッキングモードのシーケンス処理の例を示します。そこでは DSIZE[4:0] が4bitに設定され、トランスミッタのSPI_TXDRレジスタへの16bitまたは32bitの1回のアクセスの後、2つまたは4つのデータフレームがTxFIFOに書き込まれます。データフレームサイズが9bitと16bitの間である場合、32bitアクセスが実行されたとき、自動的にデータパッキングが使用されます。最下位ハーフワードが最初に使用されます（LSBFRSTの値にかかわらず）。

このシーケンスでは、RxFIFO閾値がフレームに設定されていた（データはパックされずにフレームベースで読み出される）場合、レシーバで2または4つのRXPイベントが生成される可能性があります。あるいは、SPI_CFG1レジスタのFTHLV[3:0] フィールドが、複数フレームをパックモードで読み出す（16bitまたは32bit読出しアクセス）設定にプログラムされていた場合は、1つのRXPイベントが生成されます。

データは 図847に従って配置されています。有効なビットだけがバスで実行されます。未使用のビットはトランスミッタでは考慮されず、レシーバではゼロで埋められます。

短いデータフレーム（8bit未満または16bit未満）を、より大きいデータアクセスモード（16bitまたは32bit）で使用する場合、FTHLVの値はフレームの数/データアクセスの倍数としてプログラムする必要

があります（最大8bitのフレームに32bitアクセスを使用する場合は4の倍数、あるいは、最大8bitのフレームに16bitアクセスを使用する場合、または最大16bitのフレームに32bitアクセスの場合は2の倍数）。

RxFIFO閾値の設定は、次の読出しアクセスサイズより必ず大きくなければなりません（最低でも同じでなければなりません）。そうしないと、誤った余分のデータが読み出されるからです。

設定したデータサイズより小さいFIFOデータアクセスは禁止されます。必ず最低でも1つの完了したデータフレームにアクセスする必要があります。

FTHLVビットでセットされた閾値未満の不完全なデータパケットがFIFOで利用可能になっている場合は、特定の問題が現れます。

この問題を解決するには、2つの方法があります。

A. TSIZEフィールドを使用しない方法

トランスミッタ側では、SPI_TXDRへの8bit/16bitアクセスによって任意の奇数シーケンスの最後のデータフレームを書き込むだけで十分です。

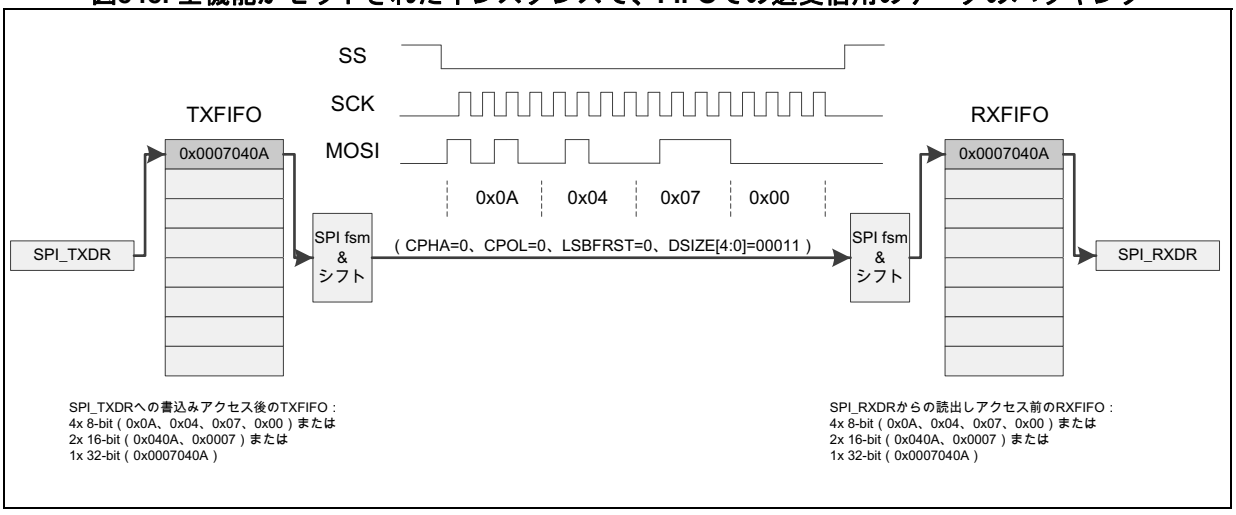
レシーバ側では、残りのデータは任意のアクセスによって読み出すことができます。余分なデータの読出しはゼロで埋められます。受信データがRxFIFOに残っているかどうかを検出するためには、RXWNEおよびRXPLVLのポーリングが使用できます（ポーリングを検出するためにシステムレベルではタイムアウトが使用されることがあります）。

B. TSIZEフィールドを使用する方法

トランスミッタ側では、EOTイベントが発生するとトランザクションはマスタによって停止されます。

受信側では、EOTがセットされた時、RXPフラグはセットされません。受信するデータの数（TSIZE）がパケットサイズの倍数でなかった場合、残っているデータの数がSPI_SRレジスタのRXWNEおよびRXPLVLフィールドによって示されます。残りのデータは任意のアクセスによって読み出すことができます。余分な読出しはゼロで埋められます。

図848. 全機能がセットされたインスタンスで、FIFOでの送受信用のデータのパッキング



1. DSIZE[4:0] は4bitに設定され、データは右詰め、有効なビットがバス上でのみ実行され、その順序はLSBFRSTに依存し、セットされている場合はすべてのデータフレームでその順序が逆になります。

68.4.15 DMA（ダイレクトメモリアクセス）を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読出し／書込み処理を容易にするために、SPIは簡単なリクエスト／確認応答プロトコルを実現するDMA機能を備えています。

SPI_CFG1レジスタのTXDMAENまたはRXDMAEN有効ビットをセットすると、DMAアクセスがリクエストされます。TxバッファとRxバッファには、定義されたパケットのサービスを満たすために、別々のリクエストを発行する必要があります

- 送信では、TXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMAはSPI_TXDRレジスタに一連の書込みを行います。
- 受信では、RXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMAはSPI_RXDRレジスタから一連の読出しを行います。トランザクションの最後にEOTがセットされ、最後のデータパケットが不完全だった場合、RXWNEおよびRXPLVL[1:0] の設定に従ってDMAリクエストが自動的にアクティブにされ、残りのデータが読み出されます。

SPIが受信専用モードにプログラムされている場合、UDRはセットされません。

SPIが送信モードにプログラムされている場合、送信データが利用できないことがあるので、スレーブ側で最終的にTXPおよびUDRがセットされることがあります。この場合、UDR管理の選択によっていくらかのデータがTXラインに送信されます。

SPIが単方向モードで使用されている場合、相補的な未使用チャネルを無効にしたまま、適切なDMAチャネルのみを有効にする必要があります。

SPIが送信専用モードにプログラムされている場合、RXPおよびOVRはセットされません。

SPIが全二重モードにプログラムされている場合、受信データが読み出されないので、最終的にRXPおよびOVRがセットされます。

送信モードで、DMAまたはユーザがすべての送信データを書き込んだとき（SPI_SRレジスタのTXTFフラグがセットされます）、EOT（TSIZE = 0の場合はTXC）フラグを監視することでSPI通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPIを無効にする前、またはマスタモードで **spi_pclk** を無効にする前にこの操作を行う必要があります。ソフトウェアは、まずEOT = 1および/またはTXC = 1になるまで待つ必要があります。

DMAを使用して通信を開始する場合、DMAチャネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rxが使用される場合、SPI_CFG1レジスタのRXDMAENビットのDMA Rxバッファを有効にします。
2. DMAが使用される場合、DMAレジスタにてTxおよびRxのDMAリクエストを有効にします。
3. DMA Txが使用される場合、SPI_CFG1レジスタのTXDMAENビットにてDMA Txバッファを有効にします。
4. SPEビットをセットして、SPIを有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. DMAが発行されていた場合、DMAレジスタにてTxおよびRxのDMAリクエストを無効にします。
2. SPI無効化手順に従ってSPIを無効にします。
3. DMA Txおよび（または）DMA Rxが使用されている場合、SPI_CFG1レジスタのTXDMAENビットおよびRXDMAENビットをクリアすることにより、DMA TxバッファおよびRxバッファを無効にします。

DMAによるデータパッキング

DMA（SPI_CFG1レジスタのTXDMAENおよびRXDMAENをセット）で転送が管理される場合、パッキングモードはSPI TXおよびSPI RX DMAチャネルで設定されるPSIZE値によって、自動的に有効／無効になります。

DMAチャネルのPSIZE値が16bitと等しい場合、SPIデータサイズは8bit以下で、パッキングモードは有効になります。同様に、DMAチャネルのPSIZE値が32bitと等しく、SPIデータサイズが16bit以下の場合、パッキングモードは有効になります。その後、DMAはSPI_TXDRレジスタへの書込み動作を管理します。

フレームサイズが小さいとき、データパッキングモードが使用されていて、送信データ数がDMAデータサイズ（16bitまたは32bit）の倍数になっていない場合、DMAはTSIZEフィールドの設定に従って自動的に転送を完了します。

あるいは、シングル／アンパックモードで最後のデータフレームをソフトウェアで書き込むこともできます。

設定したデータサイズより小さいDMAデータアクセスを設定することは禁止されます。必ず最低でも1つの完了したデータフレームにアクセスする必要があります。

68.4.16 自律モード

SPIは、進行中のトランザクションが終了するまで、特定のシステムの実行とのやり取りを必要とせずに、トランザクションを自律的に処理したり初期化したりできます。
このような自律的なトランザクションは、RUNモードやSLEEPモードで処理できるだけでなく、SPIロジックがリセットおよびクロックコントローラ（RCC）宛ての一時的なクロック要求を提供して、SPIのモードに応じてメモリとペリフェラルインタフェースとの間のデータフローを処理するために必要なSPIドメインのクロック供給を確保できる場合は、STOPモードでも処理できます。

STOPモードでは、特定のトラフィックイベント（主にTXPとRXP）に基づいてSPIレジスタを更新する必要があるたびに、ペリフェラルによってAPBクロックが要求されます。RCCの設定でSPI自律モードが有効になっていて、SPIがSTOPモードで利用可能な内部オシレータによってクロック供給されている場合、要求されたクロックはRCCから供給されます。

その後、SPIの設定に応じて、割込みまたはDMAリクエストが生成されます。割込みが無効な場合は、デバイスはSTOPモードに留まります。DMAリクエストが有効になっている場合は、デバイスがSTOPモードに留まっている間、データはDMAによってSRAMとの間で直接転送されます。有効になっている割込みが生成された場合、デバイスはSTOPモードからウェイクアップします。

注： **有効になっている割込みのフラグがセットされている間、ペリフェラルクロック要求は保留のままになります。このため、特に低電力消費に敏感なシステムでは、これらの保留中の要求を処理してできるだけ早くフラグをクリアすることが重要であり、アプリケーションはSPIを低電力モードに切り替える前に、保留中のすべての割込みイベントを確認する必要があります。**

スレーブモード

SPIが標準スレーブとして設定され、デバイスがSTOPモードの場合、SPIカーネルクロックとSPI APBクロックは永続的に提供されません。SPIインタフェースと関連FIFOとの間のすべてのデータフローは、シリアルインタフェース·クロックドメイン内の外部マスタデバイスから提供される外部SCKクロックによって処理されます。したがって、APBクロックの一時的な要求は、SPIの設定に応じて特定のトラフィックイベントに基づいて行われます。スレーブはトランザクションを初期化しないため、このモードではトランザクションの開始を同期させる必要はありません。

注： **有効になっている割込みのフラグがセットされている間、ペリフェラルクロック要求は保留のままになります。このため、特に低電力消費に敏感なシステムでは、これらの保留中の要求を処理してできるだけ早くフラグをクリアすることが重要であり、アプリケーションはSPIを低電力モードに切り替える前に、保留中のすべての割込みイベントを確認する必要があります。**

マスタモード

マスタモードで動作するSPIは、トランザクションが完了するまで外部スレーブにSCK信号を提供します。SCK信号は、デバイスがSTOPモードのときに そのSPIから出されるカーネルクロックの要求に応じて、RCCから供給されるカーネルクロックのドメイン内で実行されるSPIクロックジェネレータから生成されます。したがって、APBクロックの一時的な要求は、SPIの設定に応じて特定のトラフィックイベントに基づいて行われます。SPIマスタは常にトランザクションを初期化します。

STOPモードでの電力消費を最小限に抑えるには、ハードウェアによってトリガされる通信開始（SPI_AUTOCRレジスタのTRIGENビットの設定）と、事前定義されたデータサイズ（SPI_CR2レジスタのTSIZE > 0）の転送を組み合わせることをお勧めします。これにより、EOT処理と次のトリガイベントの間で、APBクロック要求が抑制されるようになります。

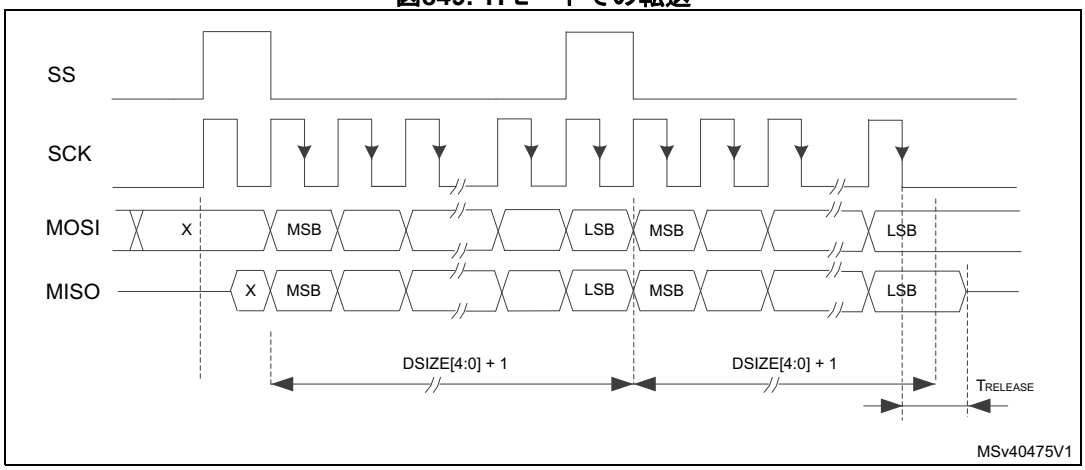
CSTARTビットがセットされると、トランザクションが開始されます。マスタトランスミッタの場合、TxFIFOもデータで埋める必要があります。RUN、SLEEP、またはSTOPモードで同期トリガが検出されると、CSTARTビットはソフトウェアまたはハードウェアによって書き込むことができます。トリガソースはSPI_AUTOCRレジスタのTRIGSELビットによって選択し、TRIGENビットによって有効にします。有効なトリガが検出されると、転送が開始され、EOTイベントまたはトランザクションの一時停止までデータを処理して転送が継続されます。TRIGENビットを変更する時は、トリガイベントの発生を防止する必要があります。それを防ぐことができない場合は、SPIが無効になっている間にTRIGENビットを書き込まなければなりません。そうしなければ、ペリフェラルの動作は保証されません。

68.5 SPIの特定モードおよび制御

68.5.1 TIモード

SPI_CFG2レジスタの特定のSP[2:0] ビットフィールド設定で、SPIはTIプロトコルと一致するように設定できます。SPIがTIモードの設定になっているときは、SCKおよびSS信号の極性、位相、フロー、およびビット順序は固定されているので、CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、SSM、RDIOP、RDIOM、MSSI、およびMIDIの設定は必要ありません。SS信号は、図849に示すように、LSBデータビット上のパルスによってプロトコルを同期させます。

図849. TIモードでの転送



スレーブモードでは、現在のトランザクションが終了した時点で、MISOピンでのスレーブ出力がハイインピーダンスになるタイミングを定義するためにクロックジェネレータが使用されます。マスタのボーレート設定（SPI_CFG1のMBR[2:0]）が適用され、このタイミングを非常に柔軟に決定するために任意のボーレートを使用できます。MISO信号がハイインピーダンス（TRELEASE）になるまでの

遅延も、内部再同期によって変わります。これには、ジェネレータに供給されるクロック信号がさらに2～4周期必要です。この値は次式で求められます：

$$\frac{Tbaud}{2} + 2 \times Tspi_ker_ck \leq Trelease \leq \frac{Tbaud}{2} + 4 \times Tspi_ker_ck$$

スレーブがデータトランザクション中にSSパルスの誤配置を検出すると、TIFREフラグがセットされます。

68.5.2 SPIエラーフラグ

次のいずれかのエラーフラグがセットされて、対応する割込み有効ビットをセットすることによって割込みが有効になっている場合、SPI割込みが生成されます。

オーバーランフラグ（OVR）

マスタまたはスレーブでデータを受信し、RxFIFOに受信データを格納するための十分なスペースがない場合に、オーバーラン状態が発生します。これは、ソフトウェアまたはDMAがRxFIFOに格納された前の受信データを読み出すために必要な時間が十分になかった場合に発生する可能性があります。

オーバーラン状態が発生すると、OVRフラグがセットされ、RxFIFOにある前の値が新しく受信された値で上書きされることはありません。新しく受信された値は破棄され、それ以降に送信されるすべてのデータは失われます。OVRIEビットがセットされている場合、OVRフラグは割込みをトリガします。OVRビットのクリアは、SPI_IFCRのOVRCビットに1を書き込むことによって行われます。次のオーバーランイベントを防ぐには、ソフトウェアの読出しによってRxFIFOが空になってからビットのクリアをすべきです。RxFIFOのスペースを最大限解放することをお勧めします。これは、RXPフラグの指示に基づいて、使用可能なすべてのデータパケットを読み出すことを意味します。

マスタモードでは、自動通信サスペンド（MASRXビット）によってRxFIFOオーバーランを防ぐことができます。

アンダーランフラグ（UDR）

スレーブ送信モードで、一般的にスレーブのTxFIFOに送信できるデータがなかった場合、アンダーラン状態がハードウェアによって内部的にキャプチャされます。その後、UDRフラグの設定はハードウェアによってステータスレジスタに伝えられます（下記の注を参照）。UDRIEビットがセットされている場合、UDRは割込みをトリガします。

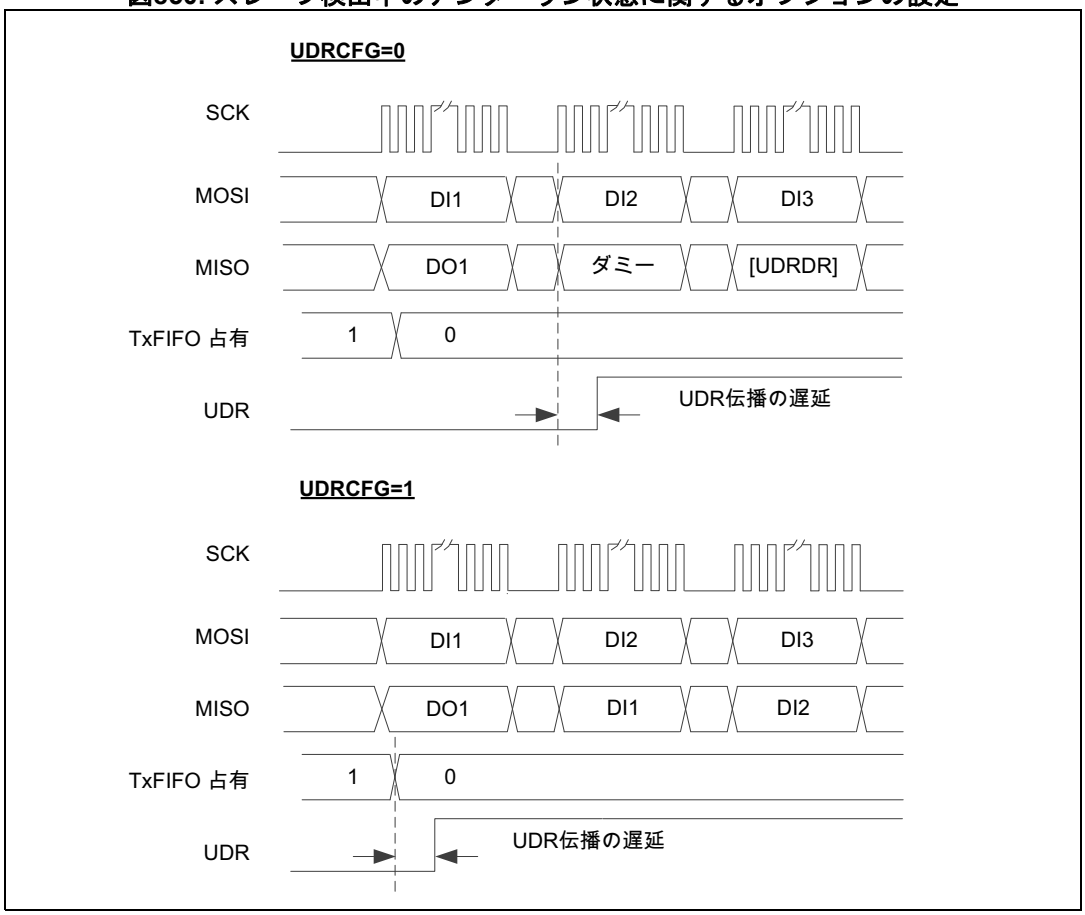
アンダーラン検出ロジックとシステム動作はUDRCFGビットに依存します。アンダーランがスレーブに検出されると、ユーザによってUDRDRレジスタに格納された一定のパターン、もしくは前にマスタから受信したデータを提供できます。最初の設定（UDRCFG = 0）が適用されると、TxFIFOが空の時に、マスタが新しいデータフレームの通信を開始したときにいつでもアンダーラン状態が評価されます。次に、最後の有効なデータと、UDRDRレジスタで定義された一定パターンの間に、必ず1つの追加のダミー（不測の）データが挿入されます（図850を参照）。マスタが通信を開始したときにTxFIFOが空でないものとして、アンダーラン状態は次のデータフローでFIFOが空になったときにだけ評価されます。TxFIFOからの有効なデータは、最後の受信データによって直ちに付加されます。

ソフトウェアによってUDRフラグがクリアされると、標準の送信が再有効化され、このクリア情報はハードウェアによってSPIロジックに伝えられます。UDRフラグをクリアする前に、次のアンダーラン状態発生のキャプチャを防ぐために、いくらかのデータをTxFIFOに書き込む必要があります。

スレーブによってトランザクションされるデータは、特にTxFIFOが空で、アンダーラン状態がまだキャプチャされていないか、クリアされたばかりの時に、トランザクションが開始または続行されたときに、予測不可能です。一般的にこれはSPIが有効になったばかりのときか、あるいは定義された

サイズでトランザクションが始まったばかりのときの場合です。さらに、この場合、データのトランザクションが開始される、あまりに直前にスレーブソフトウェアが空のTxFIFOに最初のデータを書き込んだとき、最初の数ビットは破壊される可能性があります（データをTxFIFOに伝播するのにAPBクロックの数サイクルがかかります）。

図850. スレーブ検出中のアンダーラン状態に関するオプションの設定



注： **UDRイベントのハードウェア伝播には、バス上で追加のトラフィックが必要です。これには、イベントが発生した後（ハードウェアによって捕捉されたアンダーランとソフトウェアによってクリアされたアンダーランの両方）、SPIクロックの数サイクルが常に余分にかかります。ソフトウェアによるUDRフラグのクリアがデータフレームトランザクションの終わり近くに適用された場合、またはSCKラインがフレーム間でアイドル状態の場合、次の追加のアンダーラン･パターンが、最初にスレーブによって、TxFIFOからの有効なデータが再度トランザクションされる前に送信されます。ユーザはアンダーランロジックを再開するセッション間でSPI無効化/有効化操作により、これを防ぐことができます。これにより、次のセッションを有効なデータで開始できます。**

モードフォールト（MODF）

モードフォールトは、マスタデバイスが内部SS信号（SSハードウェアモードではSSピン、SSソフトウェアモードではSSIビット）をローレベルにプルダウンしたときに発生します。これは、自動的にSPIインタフェースに次のような影響を与えます。

- MODFビットがセットされ、MODFIEビットがセットされている場合はSPI割込みがトリガされます。
- SPEビットは、MODFビットがセットされるまで、強制的にゼロになります。これにより、SPIが無効になり、MODF割込みリクエスト（有効な場合）を除くすべてのペリフェラル出力がブロックされます。
- MASTERビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODFは、SPI_IFCRレジスタのMODFCビットに1を書き込むことによってクリアされます。

複数のMCUで構成されるシステムでスレーブ間の競合を避けるには、SPEビットをセットすることによってSPIを再有効化する前に、SSピンを非アクティブレベルにプルする必要があります。

安全のため、MODFビットがセットされている間、ハードウェアはSPEビットのセットを許可しません。スレーブデバイスでは、MODFビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

マルチマスタシステムにおいてマスタがバスを支配するときの正しいソフトウェア手順は次のとおりです。

- SSOE = 0のときにマスタモードに切り替えます（別のマスタがバスを占有すると、潜在的な競合が発生する可能性があります。この場合、MODFが発生し、次のノードがマスタモードに切り替わるのを防ぎます）。
- 別のマスタのSS制御専用のGPIOピンをアクティブレベルにします。
- データのトランザクションを実行します。
- 別のマスタのSS制御専用のGPIOピンを非アクティブレベルにします。
- スレーブモードに切り替えて戻します。

CRCエラー（CRCE）

このフラグを使用して、SPI_CFG1レジスタのCRCENビットがセットされているときに受信された値の有効性を検証します。TSIZEによって定義された最後のデータが受信された後、シフトレジスタに受信された値が、レシーバのSPI_RXCRCの値と一致しなかった場合、SPI_SRレジスタのCRCEフラグがセットされます。CRCEIEビットがセットされている場合、CRCEフラグは割込みをトリガします。CRCEビットのクリアは、SPI_IFCRのCRCECビットに1を書き込むことによって行われます。

TIモードフレームフォーマットエラー（TIFRE）

SPIがスレーブモードで動作し、かつTIモードプロトコルに準拠した設定となっている場合、通信の実行中にSSパルスが発生すると、TIモードフレームフォーマットエラーが検出されます。このエラー発生すると、SPI_SRレジスタのTIFREフラグがセットされます。エラー発生時にはSPIは無効にされず、このSSパルスは無視されます。SPIは次のSSパルスを待ってから新規の転送を開始します。このエラーの検出によりいくつかのデータフレームが失われるため、データは破壊される可能性があります。

TIFREフラグは、SPI_IFCRのTIFRECビットに1を書き込むことによってクリアされます。TIFREIEビットがセットされている場合、SSエラー検出時に割込みが生成されます。データの一貫性がもはや保証されなくなるため、ソフトウェアによってマスタとスレーブ間で通信を再起動するべきです。

68.5.3 CRC計算

2つの別々の33bitの、または2つの別々の17bitのCRC計算機が、送信データおよび受信データの信頼性をチェックするために実装されています。全機能がセットされたインスタンスで、SPIではCRC多項式の長さが提供されており、その長さはデータの最大サイズが32bitの場合は5bitから33bitまで、データサイズが16bitに限られるペリフェラルインスタンスに対しては9bitから17bitまでです。機能が制限されているインスタンスでは、データサイズが16bitに制限されている場合、CRC多項式の長さは9または17のどちらかにセットできます。また、データサイズが32bitに拡張されると、オプションで33にセットできます。多項式の長さはCRCPOLYレジスタに格納されている値の最上位ビットによって定義されます。その長さはDSIZEフィールドで定義されているデータフレーム長より大きくセットする必要があります。最大のデータサイズが適用されているときは、多項式ストリングのサイズを常にデータより大きく保ちながらその最上位ビットを定義するために、さらにCRC33_17bitをセットする必要があります。SPI_CFG1のCRCSIZEフィールドでは、CRC計算レジスタからいくつの最上位ビットがCRCフレームとしてトランザクションされ、比較されるのかを定義します。これはデータフレーム長とは独立に定義されますが、データフレームサイズと同じか、その倍数でなければなりません。ただし、サイズがインスタンスの最大データサイズを超えることはできません。

CRC計算機能を最大限活用するには、多項式の長さ設定をCRCパターンサイズに一致させる必要があります。サイズが多項式の長さより大きくセットされている場合は、計算時の未使用ビットがトランザクションされ、CRCパターンの最後がすべてゼロになることが予想されます。

CRCの原理

CRC計算は、SPIが有効（SPE = 1）になる前に、SPI_CFG1レジスタのCRCENビットをセットすることによって有効になります。CRC値は、CRCPOLYレジスタとCRC33_17bitによって定義されたCRC多項式を使用して計算されます。SPIが有効になっているとき、CRC多項式は変更できますが、バス上にトラフィックがないときに限られます。

CRC計算は、SPI_CR1レジスタのCPHAビットとCPOLビットによって定義されたサンプリングクロックエッジで、ビットごとに行われます。計算されたCRC値は、SPI_CR2レジスタによって定義されたデータブロックの最後にのみ自動的にチェックされます。

受信データをもとに内部で計算されたCRC値とトランスミッタから受信したCRC値の間に不一致が検出された場合、データ破壊エラーを示すためにCRCEフラグがセットされます。CRCを処理する正しい手順は、SPIの設定および選択された転送管理によって変わります。

CRC転送管理

通信が開始され、最後のデータフレームがSPI_DRレジスタに送信または受信されるまで正常に続きます。

転送の長さはTSIZEによって定義される必要があります。要求されたデータ数がトランザクション処理されると、TXCRCが送信され、ラインで受信されたデータがRXCRC値と比較されます。

CRCSIZEの設定にかかわらず、TSIZEを全機能がセットされたインスタンスで0xFFFFにセットしたり、CRCが有効な場合に機能が制限されたインスタンスで0x3FFの値にセットしたりすることはできません。

送信では、CRCのトランザクション処理中、CRC計算は停止し、TXCRCはCRCSIZEフィールドの値に等しい長さのフレームで送信されます。

受信では、必要な数のデータがトランザクションされるとき、RXCRCは同じく停止します。その後、RXCRCレジスタの内容と比較される情報が、CRCSIZE値と等しい長さのフレームで受信されます。

CRCフレームの受信が完了すると、受信したCRC値とSPI_RXCRCレジスタで計算された値を比較する自動チェックが行われます。ソフトウェアは、SPI_SRレジスタのCRCEフラグをチェックして、

データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCECに1を書き込んでCRCEフラグをクリアします。

ユーザは冗長なCRC情報を一掃する面倒をみる必要はなく、それは自動的に行われます。

SPI_TXCRCおよびSPI_RXCRCの値のリセット

CRCフェーズのあとで新しいデータがサンプリングされた時、SPI_TXCRC値およびSPI_RXCRC値は自動的に初期化されます。これにより、データを中断することなく転送するために、DMAサーキュラモードが使用可能になります（いくつかのデータブロックは中間のCRCチェックフェーズによってカバーされます）。レシーバおよびトランスミッタ用の初期化パターンは、SPI_CR1レジスタのTCRCINIおよびRCRCINIビットの設定によってゼロまたはすべて1のどちらかに設定できます。

SPIが無効化されると、CRCの値はリセットされます。

68.6 SPI低電力モード

SPIはSTOPモードまでの自律動作をサポートします。セクション 68.4.16 : 自律モードを参照してください。

表695.低電力モードがSPIに与える影響

モード	説明
SLEEP	影響はありません。SPI割込みによって、デバイスはSLEEPモードを終了します。
STOP(1)	SPIレジスタの内容は保たれます。 RCCの設定で自律モードが有効になっていて、SPIがSTOPモードで利用可能な内部オシレータによってクロック供給される場合、転送は機能します。DMAリクエストは機能しており、これらのモードでの割込みによってデバイスはSTOPモードを終了します。
STANDBY（1）	SPIのインスタンスはこのモードのパワーダウン時に機能せず、STANDBYモードの終了後に再初期化する必要があります。このモードで動作できるSPIインスタンスには、DMAリクエストとウェイクアップ機能が備わっています。

1. インスタンスごとのSTOPモードからのウェイクアップのサポートと、STANDBYモードを使用できるかどうかについては、セクション 68.3 : SPIの実装を参照してください。インスタンスがSTOPモードで機能しない場合は、STOPモードに移行する前に無効化する必要があります。

68.7 SPI割込み

表696に、有効な場合は割込みイベント（spi_it）を生成できるSPIイベントの概要を示します。そのうちのいくつかは、さらに低電力モードからのウェイクアップ機能を備えています。それらの多くは、特定の割込み有効制御ビットを使用することによって、独立に有効化および無効化することができます。イベントに関連するフラグは、特定の方法でクリアされます。イベントフラグについての詳細は、SPIレジスタの説明を参照してください。SPIが無効化されると、割込みサービスへの伝播を防ぐために、MODF割込みリクエストを除く保留中の割込みリクエストがすべてブロックされます。

表696.SPIのウェイクアップおよび割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	イベントのクリア方法	STOPおよびSTANDBYモードの機能の終了 (1)(2)
SPI	TxFIFOロード準備済み（1つのデータパケット - FIFO閾値のスペースが利用可能）	TXP	TXPIE	TXPは、TxFIFOの空き容量がFTHLVより少なくなった時にハードウェアによってクリアされます。	可能
	データをRxFIFOに受信（1つのデータパケットが利用可能 - FIFO閾値）	RXP	RXPIE	RXPは、RxFIFOに入っているサンプルがFTHLVより少なくなった時にハードウェアによってクリアされます。	可能
	TXPとRXPの両方がアクティブ	DXP	DXPIE	TXPまたはRXPがクリアされた時	可能
	送信の転送フル	TXTF	TXTFIE	TXTFCに1を書き込む	不可
	アンダーラン	UDR	UDRIE	UDRCに1を書き込む	可能
	オーバーラン	OVR	OVRIE	OVRCに1を書き込む	可能
	CRCエラー	CRCE	CRCEIE	CRCECに1を書き込む	可能
	TIフレームフォーマットエラー	TIFRE	TIFREIE	TIFRECに1を書き込む	不可
	モードフォールト	MODF	MODFIE	MODFCに1を書き込む	不可
	転送の終了（転送の全シーケンス完了 - TSIZE値に基づいて）	EOT	EOTIE	EOTCに1を書き込む	可能
	マスタモードサスペンド	SUSP		SUSPCに1を書き込む	可能
	TXFIFO送信完了（TxFIFOエンプティ）	TXC(3)		TXCは、バス上で送信動作が開始された時にハードウェアによってクリアされます。	不可

1. すべての割込みイベントで、各インスタンスでシステムをSLEEPモードからウェイクアップできます。具体的なSTOPおよびSTANDBYモードを終了するためのインスタンスの機能について、詳しくは「動作モードに応じた機能」の表を参照してください。
2. STANDBYモードを使用できるかどうかについては、セクション 68.3：SPIの実装を参照してください。
3. TXCフラグの動作はTSIZE設定に依存します。TSIZE>0の場合、フラグはEOTCのよるクリアを含め、完全にEOTに従います。

68.8 SPI レジスタ

68.8.1 SPI 制御レジスタ1（SPI_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOLOCK
															rs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRCINI	RCRCINI	CRC33_17	SSI	HDDIR	CSUSP	CSTART	MASRX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
rw	rw	rw	rw	rw	w	rs	rw								rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **IOLOCK**：関連I/OのAF設定のロック

このビットは、SPIが無効（SPE = 0）の場合にのみ、ソフトウェアで設定できます。ハードウェアまたはソフトウェアによってSPEビットが1から0に変更されるたびに、ハードウェアによってのみクリアされます。

0：AF設定はロックされません。

1：AF設定はロックされます。

このビットをセットすると、SPI_CFG2レジスタの内容は変更できません。このビットは、SPIが有効（SPE = 1）な場合に書込み保護されます。

ビット15 **TCRCINI**：トランスミッタ用CRC計算の初期化パターン制御

0：すべて0のパターンを適用します。

1：すべて1のパターンを適用します。

ビット14 **RCRCINI**：レシーバ用CRC計算の初期化パターン制御

0：すべて0のパターンを適用します。

1：すべて1のパターンを適用します。

ビット13 **CRC33_17**：32bitのCRC多項式設定

0：フルサイズ（33bitまたは17bit）のCRC多項式は使用されません。

1：フルサイズ（33bitまたは17bit）のCRC多項式が使用されます。

ビット12 **SSI**：内部SS信号入力レベル

このビットは、SSMビットがセットされているときにのみ有効です。このビットの値はペリフェラル内部のSS入力に強制的に設定され、SSピンのI/O値は無視されます。

ビット11 **HDDIR**：半二重モードでのRx/Txの方向

半二重の設定において、HDDIRビットはRx/Txのデータ転送方向を確定します。このビットは、全二重または単方向の設定では無視されます。

0：SPIはレシーバです。

1：SPIはトランスミッタです。

ビット10 **CSUSP**：マスタのサスペンドリクエスト

このビットは0として読み出されます。

マスタモードにおいて、このビットがソフトウェアによってセットされると、現在のフレームの終了時点でCSTARTビットがリセットされ、通信がサスペンドされます。フレームのトランザクションの終了を確認するために、ユーザはSUSPフラグをチェックする必要があります。

低電力モードに移行する前に、（このビットを使用するか、またはTXDRを空に保持して）マスタモードの通信をサスペンドする必要があります。

ソフトウェアの中断後、SUSPフラグをクリアし、次のトランザクションが始まる前にSPIを無効化して再度有効化する必要があります。

ビット9 **CSTART**：マスタ転送の開始

このビットは、SPI 通信を開始するためだけに、SPIを有効化する場合に、ソフトウェアによってセットできます。これは、転送の終了（EOT）フラグがセットされたとき、またはトランザクションのサスペンドリクエストが受理されたときに、ハードウェアによってクリアされます。

0：マスタ転送はアイドル状態です。

1：マスタ転送は動作中、または自動サスペンドによって一時的にサスペンドされています。

SPIモードでは、このビットはマスタモードでのみ考慮されます。送信が有効になっている場合、送信FIFOでデータが利用可能になった場合のみ、通信が開始、または続行します。

ビット8 **MASRX**：受信モードでのマスタ自動サスペンド

このビットは、ソフトウェアによってセット／クリアされ、オーバーラン状態を避けるために、マスタレシーバモードでSPIの連続転送と自動管理を制御します。

0：オーバーラン状態にかかわらず、SPIのフロー／クロック生成は続行します（データは失われます）。

1：オーバーラン状態になる前に、RxFIFOがフル状態になった時、SPIフローは一時的にサスペンドされます。SPI通信がサスペンドされたとき、SUSPフラグがセットされます。

ハードウェアによって自動的にSPI通信がサスペンドされたとき、内部の同期遅延のために次のフレームの数ビットがすでにクロックアウトされている可能性があります。

このため、データのサイズが8bit未満のとき、自動サスペンドの信頼性は低くなります。この場合、MIDIパラメータが非ゼロ値であるときに適用される、データフレーム間に挿入された遅延時間と組み合わせることによって安全なサスペンドが達成されます。データサイズとインターリーブされたSPIサイクルの合計により、少なくともSPIクロックの8周期の長さの間隔が常に生成されます。SUSPビットのソフトウェアをクリアした後で、通信はその後の制約なく後続ビットのトランザクションによって再開され、継続します。SUSPビットをクリアする前に、ユーザはRxFIFO空間を最大限解放する必要があります。これには、その後の中断を防ぐために、RXPフラグの指示に基づいて、RxFIFOで使用可能なすべてのデータパケットを読み出します。

ビット7:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SPE**：シリアルペリフェラル有効

このビットは、ソフトウェアによってセット／クリアされます。

0：シリアルペリフェラルは無効です。

1：シリアルペリフェラルは有効です。

SPE = 1の場合、SPIデータ転送は有効化され、SPI_CFG1およびSPI_CFG2の設定レジスタ、CRCPOLY、UDRDR、SPI_AUTOCRレジスタの一部、およびSPI_CR1レジスタのIOLOCKビットは書込み保護されます。それらはSPE = 0のときのみ変更できます。

SPE = 0の場合、任意のSPI動作が停止され、無効化されると、MODF割込みリクエストを除いて、割込みが有効化されたイベントのすべての保留中のリクエストがブロックされ（ただし保留によりspi_plckクロックのリクエストが引き続き伝播される）、SS出力がマスタで無効化され、RDY信号はスレーブでノットレディステータスを保持し、内部ステートマシンはリセットされ、FIFOの内容はすべて一掃され、CRC計算は初期化され、受信データレジスタはゼロを読み出します。

MODFエラーフラグがアクティブになっているとき、SPEはクリアされ、セットできません。

68.8.2 SPI 制御レジスタ2（SPI_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **TSIZE[15:0]**：現在の転送でのデータ数

これらのビットがソフトウェアによって変更される場合、SPIを無効にする必要があります。

TSIZEにゼロ値が格納されているときにCSTARTがセットされると、終わりのないトランザクションが開始されます。TSIZEを0xFFFFの値、またはCRCが有効な場合には0x3FFFの値にセットすることはできません。

注： TSIZE[15:10] ビットは、制限された機能がセットされたインスタンスで予約され、リセット値で保持する必要があります。

68.8.3 SPI 設定レジスタ1（SPI_CFG1）

アドレスオフセット：0x08

リセット値：0x0007 0007

このレジスタの内容は、TXDMAENビットとRXDMAENビットを除き、SPIが有効になっているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BPASS	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	CRCSIZE[4:0]				
rw	rw	rw	rw						rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXDMAEN	RXDMAEN	Res.	Res.	Res.	Res.	UDRCFG	FTHLV[3:0]				DSIZE[4:0]				
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **BPASS**：マスタボーレートクロックジェネレータのプリスケーラのバイパス

0：バイパスは無効です。

1：バイパスは有効です。

ビット30:28 **MBR[2:0]**：マスタボーレートのプリスケーラの設定

000：SPIマスタクロック／2
001：SPIマスタクロック／4
010：SPIマスタクロック／8
011：SPIマスタクロック／16
100：SPIマスタクロック／32
101：SPIマスタクロック／64
110：SPIマスタクロック／128
111：SPIマスタクロック／256

注： MBR設定もTIモードで動作するスレーブで考慮されます（セクション 68.5.1：TIモードを参照）。

ビット27:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **CRCEN**：ハードウェアCRC計算有効

0：CRC計算は無効です。
1：CRC計算は有効です。

ビット21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **CRCSIZE[4:0]**：トランザクションおよび比較されるCRCフレームの長さ

CRC結果がトランザクションまたは比較されるとき、多項式の計算から最上位の数ビットが考慮されます。多項式の長さは、この設定の影響を受けません。

00000：予約済み
00001：予約済み
00010：予約済み
00011：4bit
00100：5bit
00101：6bit
00110：7bit
00111：8bit
.....
11101：30bit
11110：31bit
11111：32bit

値は、データサイズ（DSIZE[4:0]）と同じか、その倍数でセットする必要があります。最大サイズは、インスタンスでのDSIZEの最大値に対応します。

注： CRCSIZEビットフィールドの最上位ビットは、データサイズが16bitに限定されているペリフェラルのインスタンスで予約済みです。

ビット15 **TXDMAEN**：Tx DMAストリーム有効

0：Tx DMAは無効です。
1：Tx DMAは有効です。

ビット14 **RXDMAEN**：Rx DMAストリーム有効

0：Rx-DMAは無効です。
1：Rx-DMAは有効です。

ビット13:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **UDRCFG**：アンダーラン状態でのスレーブトランスミッタの動作

0：スレーブは、ユーザによってSPI_UDRDRレジスタに定義された一定のパターンを送信します。
1：スレーブは、マスタから最後に受信したデータを繰り返します。スレーブが送信専用モード（COMM[1:0] = 01）で設定されている場合、すべてゼロのパターンが繰り返されます。
詳細については、図 850：スレーブ検出中のアンダーラン状態に関するオプションの設定を参照してください。

ビット8:5 **FTHLV[3:0]**：FIFO閾値レベル

1つのデータパケットのデータフレーム数を定義します。パケットのサイズはFIFOスペースの1/2を超えてはいけません。

0000：1データ
0001：2データ
0010：3データ
0011：4データ
0100：5データ
0101：6データ
0110：7データ
0111：8データ
1000：9データ
1001：10データ
1010：11データ
1011：12データ
1100：13データ
1101：14データ
1110：15データ
1111：16データ

以下のように、設定されたパケットサイズがデータレジスタのアクセス並行度に整合する場合、SPIインタフェースはより効率的になります。

– SPIデータレジスタが16bitレジスタとしてアクセスされ、DSIZE ≤ 8bitの場合、FTHLV=2、4、6を選択することを推奨します。
– SPIデータレジスタが32bitレジスタとしてアクセスされ、DSIZE > 8bitの場合、FTHLV=2、4、6を選択し、DSIZE ≤ 8bitの場合はFTHLV=4、8、12を選択することを推奨します。

注： FTHLV[3:2] ビットは機能が制限されているインスタンスで予約済みです。

ビット4:0 **DSIZE[4:0]**：1つのSPIデータフレームのビット数

00000：未使用
00001：未使用
00010：未使用
00011：4bit
00100：5bit
00101：6bit
00110：7bit
00111：8bit
.....
11101：30bit
11110：31bit
11111：32bit

注： 最大データサイズは、一部のインスタンスでは16bitまで制限可能です。機能が制限されているインスタンスでは、DSIZE[2:0] ビットは予約済みで、リセット状態で保持する必要があります。その後、DSIZE[4:3] ビットは、次のデータサイズの設定を制御します。

00xxx：8bit
01xxx：16bit
10xxx：24bit
11xxx：32bit

68.8.4 SPI 設定レジスタ2（SPI_CFG2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このレジスタの内容は、SPIが有効になっているか、またはSPI_CR1レジスタでIOLOCKビットがセットされているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFCNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSBFRST	マスタ	SP[2:0]			COMM[1:0]		Res.
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOSWP	RDIOP	RDIOM	Res.	Res.	Res.	Res.	Res.	MIDI[3:0]				MIDI[3:0]			
rw	rw	rw						rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **AFCNTR :** GPIOオルタネート機能制御

このビットは、SPE = 0のときのみ考慮されます。

0：ペリフェラルは無効になっているとき、GPIOの制御をしません。

1：ペリフェラルは常に、関連するすべてのGPIOを制御します。

特定の設定の理由（たとえば、CRCリセット、CPHA、またはHDDIR変更）のためにSPIを一時的に無効にする必要があるときに、このビットをセットすると、オルタネート機能モードで設定された関連出力を、現在のSPI設定に対応する状態に強制することによって、そのグリッチを防止できます。

ビット30 **SSOM**：マスタモードでのSS出力管理

このビットはSSOEが有効なときにマスタモードで考慮されます。2つの連続したデータ転送の間でSS出力を設定することが可能になります。

0：データ転送が完了するまで、SSはアクティブレベルに保持され、EOTフラグによって非アクティブになります。

1：MIDI[3:0] >1のとき、SPIデータフレームはSSの非アクティブパルスによってインターリーブされます。

ビット29 **SSOE**：SS出力有効

このビットは、マスタモードでのみ考慮されます。

0：SS出力は無効にされ、SPIはマルチマスタ設定で機能できます。

1：SS出力は有効です。SPIはマルチマスタ環境では機能できません。SSOM、MIDI、MSSI、SSIOPビットの設定に従って転送を完了またはSPIを無効化した後で、SSピンを非アクティブレベルに強制します。

ビット28 **SSIOP**：SS入出力の極性

0：SS信号に関してローレベルがアクティブです。

1：SS信号に関してハイレベルがアクティブです。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 **SSM :** SS信号入力のソフトウェア管理

0：SS入力値はSS PADによって決定されます。

1：SS入力値はSSIビットによって決定されます。

マスタがハードウェアのSS出力（SSM = 0およびSSOE = 1）を使用する場合、SS信号入力はマスタモードのフォールトエラーを防ぐために内部的に非アクティブ状態に強制されます。

ビット25 **CPOL :** クロック極性

0：アイドル時にSCK信号は0となります。

1：アイドル時にSCK信号は1となります。

ビット24 **CPHA**：クロックフェーズ

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2番目のクロック遷移が最初のデータキャプチャエッジです。

ビット23 **LSBFRST**：データフレームフォーマット

0：MSBが最初に送信されます。

1：LSBが最初に送信されます。

ビット22 **MASTER**：SPIマスタ

0：SPIはスレーブです。

1：SPIはマスタです。

ビット21:19 **SP[2:0]**：シリアルプロトコル

000：SPIモトローラ

001：SPI TI

その他。予約済み。使用できません。

ビット18:17 **COMM[1:0]**：SPI通信モード

00：全二重

01：単方向トランスミッタ

10：単方向レシーバ

11：半二重

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **IOSWP**：MISOおよびMOSIピンのスワップ機能

0：スワップなし。

1：MOSIおよびMISOはスワップされます。

このビットがセットされた場合、MISOおよびMOSIピンのオルタネートの機能は逆になります。

元のMISOピンがMOSIになり、元のMOSIピンがMISOになります。

ビット14 **RDIOP**：RDY信号の入出力の極性

RDIOPが0の場合、信号がハイレベルにおいてスレーブが通信できる状態であることを意味します。

RDIOPが1の場合、信号がローレベルにおいてスレーブが通信できる状態であることを意味します。

ビット13 **RDIOM**：RDY信号の入出力の管理

0：RDY信号は、永続的にアクティブとして内部的に固定されていると定義されます（RDIOP設定は無効です）。

1：RDY信号は専用ピン（RDIOP設定が有効）のオルタネート機能入力（マスタの場合）または出力（スレーブの場合）から得られます。

注： SPI_CFG1レジスタのDSIZEが8bit未満で設定されている場合、RDIOMビットはゼロで保持する必要があります。

ビット12:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **MIDI[3:0]**：マスタのデータ間アイドル状態

マスタモードで2つの連続するデータフレーム間に挿入される最小時間遅延（SPIクロックサイクルの周期で表される）を指定します。

0000：遅延なし

0001：1クロックサイクル周期の遅延

.....

1111：15クロックサイクル周期の遅延

注： このビットは、TIモードではサポートされていません。

ビット3:0 **MSSI[3:0]**：マスタのSSアイドル状態

SSOEが有効なとき、マスタモードで、セッションを開始するSSのアクティブエッジと、セッションの最初のデータ･フレームの初めの部分との間に追加で挿入される余分な遅延（SPIクロックサイクル周期の数で表される）を指定します。

0000：余分な遅延なし

0001：1クロックサイクル周期の遅延を付加

.....

1111：15クロックサイクル周期の遅延を付加

注： このビットは、TIモードではサポートされていません。
遅延を含めるには、SPIを無効にし、セッションを再度有効にする必要があります。

68.8.5 SPI 割込み有効レジスタ（SPI_IER）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MODFIE	TIFREIE	CRCEIE	OVRIE	UDRIE	TXTFIE	EOTIE	DXPIE	TXPIE	RXPIE
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **MODFIE**：モードフォールト割込み有効

0：MODF割込みは無効です。

1：MODF割込みは有効です。

ビット8 **TIFREIE**：TIFRE割込み有効

0：TIFRE割込みは無効です。

1：TIFRE割込みは有効です。

ビット7 **CRCEIE**：CRCエラー割込み有効

0：CRC割込みは無効です。

1：CRC割込みは有効です。

ビット6 **OVRIE**：OVR割込み有効

0：OVR割込みは無効です。

1：OVR割込みは有効です。

ビット5 **UDRIE**：UDR割込み有効

0：UDR割込みは無効です。

1：UDR割込みは有効です。

ビット4 **TXTFIE**：TXTFIE 割込み有効

0：TXTF割込みは無効です。

1：TXTF割込みは有効です。

ビット3 **EOTIE**：EOT、SUSP、およびTXC割込み有効

0：EOT/SUSP/TXC割込みは無効です。

1：EOT/SUSP/TXC割込みは有効です。

ビット2 **DXPIE**：DXP割込み有効

DXPIEは、ソフトウェアによってセットされ、TXTFフラグのセットイベントによってクリアされます。

0：DXP割込みは無効です。

1：DXP割込みは有効です。

ビット1 **TXPIE**：TXP割込み有効

TXPIEは、ソフトウェアによってセットされ、TXTFフラグのセットイベントによってクリアされます。

0：TXP割込みは無効です。

1：TXP割込みは有効です。

ビット0 **RXPIE**：RXP割込み有効

0：RXP割込みは無効です。

1：RXP割込みは有効です。

68.8.6 SPI ステータスレジスタ（SPI_SR）

アドレスオフセット：0x14

リセット値：0x0000 1002

SPIを再度有効にしたときに、このレジスタのフラグはすべて、自動的にはクリアされません。以下のビットの説明に示された、フラグクリアレジスタ経由の、クリアするための特定の専用アクセスが必要です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTSIZE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXWNE	RXPLVL[1:0]		TXC	SUSP	Res.	MODF	TIFRE	CRCE	OVR	UDR	TXTF	EOT	DXP	TXP	RXP
r	r	r	r	r		r	r	r	r	r	r	r	r	r	r

ビット31:16 **CTSIZE[15:0]**：現在のTSIZEセッションで残っているデータフレームの数

この値は、トラフィックがバスで進行中、または低電力モードでの自律動作中は、完全には信頼できません。

注： CTSIZE[15:0] ビットは機能が制限されているインスタンスでは使用できません。

ビット15 **RXWNE**：RxFIFOワードノットエンプティ

0：RxFIFOスペースの4バイト未満がデータによって占有されます。

1：RxFIFOスペースの4バイト以上がデータによって占有されます。

注： このビット値は、DSIZE設定に依存せず、残留データによるRxFIFOの占有に関するRXPLVL[1:0]の情報と一緒に保持されます。

ビット14:13 **RXPLVL[1:0]**：RxFIFOパッキングレベル

RXWNE = 0で、データサイズが最大16bitの場合、この値はRxFIFOで続いている残りのデータフレーム数を示します。

00：RxFIFOで使用可能な次のフレームはありません。

01：使用可能なフレームが1つあります。

10：使用可能なフレームが2つあります*。

11：使用可能なフレームが3つあります*。

注： (*)：データサイズが最大8bitにセットされている場合のみ可能な値。

データサイズが16bitより大きい場合、これらのビットは常に00として読み出されます。その場合、FIFOで受信した1つのデータフレームは、データサイズが17 ~ 24bitの場合、RWNEビットでもRXPLVLビットでも検出することはできません。ユーザは、受信したデータ数を検出するため、TSIZE > 0の場合のEOTイベントやFTHLV = 0の場合のRXPイベントの監視など、その他の方法を適用する必要があります。

ビット12 **TXC**：TxFIFO送信完了

フラグの動作はTSIZE設定に依存します。

TSIZE=0のとき、TXCはハードウェアによってのみ変更され、TxFIFOが空になってバス上での動作がなくなるたびに発生します。

TSIZE ≠ 0の場合、ソフトウェアのクリアを含むEOTフラグ値をコピーするだけのTXCを監視する理由は特にありません。EOTIEがセットされているとき、TXCは割込みを生成します。

このフラグは、SPIがリセットまたは無効化されるとセットされます。

0：現在のデータのトランザクションはまだ進行中です。データはTxFIFOで利用可能になっているか、または最後のフレームの送信が進行中です。

1：最後のTxFIFOフレーム送信完了

ビット11 **SUSP**：サスペンド･ステータス

マスタモードにおいて、CSUSPリクエストが出され、現在のフレームが完了したとき、またはマスタ自動サスペンド受信モード（SPI_CR1レジスタのMASRXビットがセットされている）でRxFIFOがフル状態になったとき、直ちにSUSPがハードウェアによってセットされます。

EOTIEがセットされているとき、SUSPは割込みを生成します。

このビットは、SPIを無効にする前にクリアする必要があります。これを行うには、SPI_IFCRのSUSPCビットに1だけを書き込みます。

0：SPIはサスペンドされていません（マスタモードまたは他のモードがアクティブです）。

1：マスタモードはサスペンドされています（現在のフレームは完了しています）。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **MODF**：モードフォールト

0：モードフォールトはありません。

1：モードフォールトを検出しました。

MODFをセットすると、SPI_CR1レジスタのSPEおよびIOLOCKビットがリセットされ、MODFがクリアされるまで、SPEの再設定はブロックされます。

このビットは、排他的にSPI_IFCRのMODFCビットに1を書き込むことでクリアされます。

ビット8 **TIFRE**：TIフレームフォーマットエラー

0：TIフレームエラーはありません。

1：TIフレームエラーを検出しました。

このビットは、排他的にSPI_IFCRのTIFRECビットに1を書き込むことでクリアされます。

ビット7 **CRCE**：CRC エラー

0：CRCエラーはありません。

1：CRCエラーが検出されました。

このビットは、オプションでSPIを再度有効にするか、SPI_IFCRのCRCECビットに1を書き込むことでクリアされます。

ビット6 **OVR**：オーバーラン

0：オーバーランはありません。

1：オーバーランが検出されました。

このビットは、オプションでSPIを再度有効にするか、SPI_IFCRのOVRCビットに1を書き込むことでクリアされます。

ビット5 **UDR**：アンダーラン

0：アンダーランはありません。

1：アンダーランが検出されました。

このビットは、オプションでSPIを再度有効にするか、SPI_IFCRのUDRCビットに1を書き込むことでクリアされます。

注： SPIモードで、UDRフラグはスレーブモードにのみ適用されます。

ビット4 **TXTF**：送信の転送フル

0：TxFIFOのアップロードは処理中または開始されていません。

1：TxFIFOのアップロードは終了しています。

アプリケーションソフトウェアまたはDMAによって転送内のすべてのデータパケットが送信用に提出されると、つまりTSIZE数のデータがTxFIFOにプッシュされた時、すぐにハードウェアによってTXTFがセットされます。

このビットは、排他的にソフトウェアによってSPI_IFCRのTXTFCビットに1を書き込むことでクリアされます。

TXTFIEビットがセットされている場合、TXTFフラグは割込みをトリガします。

TXTFをセットするとTXPIEおよびDXPIEマスクがクリアされるので、アプリケーションソフトウェアはTXPおよびDXP割込みをいつ無効にするかの計算の負担を免れます。

ビット3 **EOT**：転送終了

全転送が完了すると、つまりSPIが再度有効にされたか、またはSPIでTSIZE数のデータが送信および／または受信された時、すぐにハードウェアによってEOTがセットされます。EOTは、SPIを再度有効にしたとき、またはオプションで SPI_IFCRのEOTCビットに1を書き込むことでクリアされます。

EOTIEビットがセットされている場合、EOTフラグは割込みをトリガします。

TXTFフラグがセットされるまでDXPフラグが使用され、DXPIEがクリアされている場合、RxFIFOに入っている最後のパケットを1回でダウンロードするためにEOTを使用することができます。

0：転送は処理中または開始されていません。

1：転送が完了しました。

マスタでは、EOTイベントによって、データのトランザクションが終了し、オプションでSS出力が処理されます。CRCが適用されると、EOTイベントはCRCフレームのトランザクションまで拡張されます。

内部ステートマシンを適切に再開するには、設定を変更しない場合でも、次のトランザクションが始まる前にSPIを無効にして再度有効化することを強く推奨します。

ビット2 **DXP**：二重パケット

0：TxFIFOはフルで、および／またはRxFIFOは空です。

1：TxFIFOには書き込むスペースがあり、RxFIFOには少なくとも1つの読み出すパケットがあります。

DXPフラグは、TXPとRXPの両フラグがセットされたとき、SPIモードにかかわらずセットされます。

ビット1 **TXP**：送信パケットスペース利用可能

0：TxFIFOに次のデータパケットを提供するための十分な空きスペースがありません。

1：TxFIFOに1つ以上のデータパケットを提供するための十分な空きスペースがあります。

TXPフラグはハードウェアによってのみ変更できます。この値は、FIFOの物理的サイズとその閾値（FTHLV[3:0]）、データフレームサイズ（SPIモードでのDSIZE[4:0]）、および実際の通信フローに依存します。SPI_TXDRへの連続した書込み動作によってデータパケットが格納される場合、TxFIFOに完全なデータパケットが格納されると、TXPフラグを再度確認する必要があります。SPIがリセットされるか､または無効化されるとSPI TxFIFOにアクセスできなくなるにもかかわらず､TXPはセットされます。

ビット0 **RXP**：受信パケット利用可能

0：RxFIFOは空であるか、または不完全なデータパケットが受信されています。

1：RxFIFOには少なくとも1つのデータパケットが入っています。

このフラグは、ハードウェアによって変更されます。SPIが有効な場合、RxFIFOで現在使用可能な全体データ数を監視します。RXPの値は、FIFOの閾値（FTHLV[3:0]）、データフレームサイズ（SPIモードでのDSIZE[4:0]）、および実際の通信フローに依存します。データパケットがSPI_RXDRから連続読出しを行うことで読み出される場合、完了したデータパケットがRxFIFOから読み出されたら、RXPフラグを再度確認する必要があります。

68.8.7 SPI 割込み/ステータスフラグクリアレジスタ（SPI_IFCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SUSPC	Res.	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
				w		w	w	w	w	w	w	w			

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **SUSPC**：サスペンドフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのSUSPフラグがクリアされます。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 **MODFC**：モードフォールトフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのMODFフラグがクリアされます。

ビット8 **TIFREC**：TIフレームフォーマットエラーフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのTIFREフラグがクリアされます。

ビット7 **CRCEC**：CRC エラーフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのCRCEフラグがクリアされます。

ビット6 **OVRC**：オーバーランフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのOVRフラグがクリアされます。

ビット5 **UDRC**：アンダーランフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのUDRフラグがクリアされます。

ビット4 **TXTFC**：送信転送フルフラグクリア

このビットに1を書き込むと、SPI_SRレジスタのTXTFフラグがクリアされます。

ビット3 **EOTC**：転送の終了フラグクリア

このビットに1を書き込むと、SPI_SRレジスタのEOTフラグがクリアされます。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

68.8.8 SPI 自律モード制御レジスタ（SPI_AUTOCR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGEN	TRIGPOL	TRIGSEL[3:0]			
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **TRIGEN**：CSTARTトリガのHW制御有効

0：HW制御は無効です。

1：HW制御は有効です。

注： 書込み中のトリガイベントを防止できない場合は、SPIが無効になっているときにTRIGENを変更する必要があります。

ビット20 **TRIGPOL**：トリガ極性

0：トリガは立ち上がりエッジでアクティブです。

1：トリガは立ち下がりエッジでアクティブです。

注： このビットは、SPE=0のときのみ書き込むことができます。

ビット19:16 **TRIGSEL[3:0]**：トリガ選択（セクション ：SPIの相互作用の説明を参照）

0000：spi_trg0を選択します。

0001：spi_trg1を選択します。

.....

1111：spi_trg15を選択します。

注： これらのビットは、SPE=0のときのみ書き込むことができます。

ビット15:0 予約済みであり、リセット値に保持する必要があります。

68.8.9 SPI 送信データレジスタ（SPI_TXDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXDR[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXDR[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **TXDR[31:0]**：送信データレジスタ

このレジスタは、TxFIFOとのインタフェースとして使用できます。これに書き込むとTxFIFOがアクセスされます。

注： データは常に右詰めです。未使用のビットは、レジスタへの書込み時に無視され、レジスタの読出し時にゼロとして読み出されます。

注： DRはバイト単位に（8bitアクセスで）アクセス可能です。この場合、1回のアクセスで1データバイトのみが書き込まれます。

ハーフワード単位（16bitアクセス）の場合、1回のアクセスで2データバイトまたは1ハーフワードデータが書き込めます。

ワード単位（32bitアクセス）の場合、1回のアクセスで4データバイト、または2ハーフワードデータ、またはワードデータが書き込めます。

設定したデータサイズより小さいレジスタの書込みアクセスは禁止されます。

68.8.10 SPI 受信データレジスタ（SPI_RXDR）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXDR[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RXDR[31:0]**：受信データレジスタ

このレジスタは、RxFIFOとのインタフェースとして使用できます。これを読み出すとRxFIFOがアクセスされます。

注： データは常に右詰めです。未使用のビットは、レジスタの読出し時にはゼロとして読み出されます。このレジスタへの書込みは無視されます。

注： DRはバイト単位（8bitアクセス）でアクセス可能です。この場合、1回のアクセスで1データバイトのみが読み出されます。

ハーフワード単位（16bitアクセス）の場合、1回のアクセスで2データバイトまたは1ハーフワードデータが読み出されます。

ワード単位（32bitアクセス）この場合、1回のアクセスで4データバイト、または2ハーフワードデータ、またはワードデータが読み出されます。

設定したデータサイズより小さいレジスタの読出しアクセスは禁止されます。

68.8.11 SPI 多項式レジスタ（SPI_CRCPOLY）

アドレスオフセット：0x40

リセット値：0x0000 0107

このレジスタの内容は、SPIが有効になっているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRCPOLY[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRCPOLY[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **CRCPOLY[31:0]**：CRC多項式レジスタ

このレジスタは、CRC計算用の多項式を格納します。

デフォルトの9bit多項式の設定0x107はDSIZEのデフォルトの8bit設定に対応しています。これは、ストリングの最上位ビットが常に隠されている固定長の多項式ストリングを備えた他のST製品で使用されている設定0x07と互換性があります。

多項式の長さはこのレジスタに格納されている値の最上位ビットによって与えられます。これはDSIZEよりも大きくなければなりません。DSIZEが最大の32bitまたは16bitのサイズに設定され、CRCが有効になっているとき、多項式の長さをデータサイズより大きく保つために、CRCPOLYレジスタとともにさらにCRC33_17bitをセットする必要があります。

注： CRCPOLY[31:16] ビットは、16bitに制限されたデータサイズのインスタンスで予約済みです。これらのアドレスに32bitアクセスが適用される場合、何も制約はありません。予約済みのビット31～16は常にゼロが読み出され、それらへの書込みは無視されます。

68.8.12 SPI トランスミッタCRCレジスタ（SPI_TXCRC）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXCRC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXCRC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **TXCRC[31:0]**：トランスミッタ用CRCレジスタ

CRC計算が有効なとき、TXCRC[31:0] ビットには、その後に送信されたバイトから算出されたCRC値が格納されます。CRC計算は、SPI_CR1のCRCENビットがセットされたとき、またはデータブロックが完全にトランザクションされたときに初期化されます。CRCは、SPI_CRCPOLYレジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLYレジスタと、SPI_CFG1レジスタのCRCSIZEビットの設定に依存します。

注： 通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。

注： TXCRC[31-16] ビットは、16bitに制限されたデータサイズのインスタンスで予約済みです。これらのアドレスに32bitアクセスが適用される場合、何も制約はありません。予約済みのビット31～16は常にゼロが読み出され、それらへの書込みは無視されます。

注： このレジスタの内容がソフトウェアによって読み出された場合、CRCSIZEビットフィールドの設定は考慮されません。この場合、未使用ビットにマスクは適用されません。

68.8.13 SPI レシーバCRCレジスタ（SPI_RXCRC）

アドレスオフセット：0x48

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXCRC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCRC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RXCRC[31:0]**：レシーバ用CRCレジスタ

CRC計算が有効なとき、RXCRC[31:0] ビットには、その後に受信したバイトから算出されたCRC値が格納されています。CRC計算は、SPI_CR1のCRCENビットがセットされたとき、またはデータブロックが完全にトランザクションされたときに初期化されます。CRCは、SPI_CRCPOLYレジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLYレジスタと、SPI_CFG1レジスタのCRCSIZEビットの設定に依存します。

注： 通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。

RXCRC[31-16] ビットは､16bitに制限されたデータサイズのペリフェラルインスタンスで予約済みです。これらのアドレスに32bitアクセスが適用される場合、何も制約はありません。予約済みのビット31～16は常にゼロが読み出され、それらへの書込みは無視されます。

注： このレジスタの内容がソフトウェアによって読み出された場合､CRCSIZEビットフィールドの設定は考慮されません。この場合、未使用ビットにマスクは適用されません。

68.8.14 SPI アンダーランデータレジスタ（SPI_UDRDR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

このレジスタの内容は、SPIが有効になっているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UDRDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UDRDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **UDRDR[31:0]**：スレーブのアンダーラン状態でのデータ

このレジスタは、スレーブモードで、アンダーラン状態でのみ考慮されます。考慮されるビット数は、SPI_CFG1レジスタのDSIZEビットの設定に依存します。アンダーラン状態の処理はSPI_CFG1レジスタのUDRCFGビットの設定に依存します。

注： UDRDR[31-16] ビットは､16bitに制限されたデータサイズのペリフェラルインスタンスで予約済みです。これらのアドレスに32bitアクセスが適用される場合、何も制約はありません。予約済みのビット31～16は常にゼロが読み出され、それらへの書込みは無視されます。

68.8.15 SPIレジスタマップ

表697.SPIレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SPI_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOLOCK	TCRCINI	RCRCINI	CRC33_17	SSI	HDDIR	CSUSP	CSTART	MASRX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
	リセット値																0	0	0	0	0	0	0	0	0								0
0x04	SPI_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSIZE[15:0]															
	リセット値																	0 (1)	0 (1)	0 (1)	0 (1)	0 (1)	0 (1)	0	0	0	0	0	0	0	0	0	0
0x08	SPI_CFG1	BPASS	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	CRCSIZE[4:0]					TXDMAEN	RXDMAEN	Res.	Res.	Res.	Res.	UDRCFG	FTHLV[3:0]				DSIZE[4:0]				
	リセット値	0	0	0	0						0		1	1	1	1	1	0	0					0	0(1)	0 (1)	0	0	0	0 (1)	1 (1)	1 (1)	1 (1)
0x0C	SPI_CFG2	AFCNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSBFRST	マスタ	SP[2:0]			COMM [1:0]		Res.	IOSWP	RDIOP	RDIOM	Res.	Res.	Res.	Res.	Res.	MIDI[3:0]				MIDI[3:0]				
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0	0	0						0	0	0	0	0	0	0	0
0x10	SPI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODFIE	TIFREIE	CRCEIE	OVRIE	UDRIE	TXTFIE	EOTIE	DXPIE	TXPIE	RXPIE
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x14	SPI_SR	CTSIZE[15:0] (1)																RXWNE	RXPLVL [1:0]		TXC	SUSP	Res.	MODF	TIFRE	CRCE	OVR	UDR	TXTF	EOT	DPXP	TXP	RXP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0		0	0	0	0	0	0	0	0	1	0
0x18	SPI_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSPC	Res.	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
	リセット値																					0		0	0	0	0	0	0	0			
0x1C	SPI_AUTOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGEN	TRIGPOL	TRIGSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0																
0x20	SPI_TXDR	TXDR[31:16]																TXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24-0x2C	予約済み	予約済みです。																															
0x30	SPI_RXDR	RXDR[31:16]																RXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 - 0x3C	予約済み	予約済みです。																															
0x40	SPI_CRCPOLY	CRCPOLY[31:16](2)																CRCPOLY[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
0x44	SPI_TXCRC	TXCRC[31:16] (2)																TXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	SPI_RXCRC	RXCRC[31:16] (2)																RXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	SPI_UDRDR	UDRDR[31:16] (2)																UDRDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. ビットフィールドは、機能が制限されているインスタンスのために予約済みで、リセット値で保持する必要があります。詳細については、セクション 68.8 : SPI レジスタの具体的なレジスタの説明を参照してください。
2. ビット31～16は、データサイズが16bitに限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに32bitアクセスが適用される場合、何も制約はありません。予約済みのビット31～16は常にゼロが読み出され、それらへの書込みは無視されます。

レジスタ境界アドレスについては、セクション2.3を参照してください。

69 シリアルオーディオインタフェース（SAI）

69.1 概要

このSAIインタフェース（シリアルオーディオインタフェース）は、柔軟性があり幅広い構成が可能なため、多様なオーディオプロトコルに対応できます。多くのステレオまたはモノラルオーディオアプリケーションが対象になります。例えば、I2S規格、LSB/MSB詰め、PCM/DSP、TDM、AC’97の各プロトコルに対応することができます。オーディオブロックがトランスミッタとして設定された場合、SPDIF出力が提供されます。

このレベルの柔軟性と再設定機能を実現するため、SAIは独立した2つのオーディオサブブロックを内蔵しています。各ブロックには専用のクロックジェネレータとI/Oラインコントローラが備わっています。

このSAIは、マスタまたはスレーブの設定で動作します。このオーディオサブブロックはレシーバにもトランスミッタにもなり、他の一方に対して同期または非同期で動作します。

SAIを他のSAIと接続することで、互いに同期して動作することができます。

69.2 SAIの主な機能

- 個別にFIFOを内蔵し、トランスミッタまたはレシーバとなることができる2つの独立したオーディオサブブロック。
- 各オーディオサブブロックに8ワードFIFOを内蔵。
- オーディオサブブロック間での同期モードまたは非同期モードが可能。
- 複数のSAI間の同期が可能。
- 両オーディオサブブロックに対して独立してマスタ設定またはスレーブ設定が可能。
- 両オーディオサブブロックをマスタモードに設定した場合に、個別のオーディオサンプリング周波数に対応する各オーディオブロック用クロックジェネレータ。
- 設定可能なデータサイズ：8、10、16、20、24、32bit。
- オーディオプロトコル：I2S、LSB/MSB詰め、PCM/DSP、TDM、AC'97。
- 最大4台のマイクロフォンのペアをサポートするPDMインタフェース。
- 必要に応じて、SPDIF出力を利用可能。
- サイズ設定が可能な最大16個のスロットが利用可能。
- フレームごとのビット数が設定可能。
- フレーム同期アクティブレベルが設定可能（オフセット、ビット長、レベル）。
- スロット内の先頭アクティブビット位置が設定可能。
- LSBが先にまたはMSBが先にデータ転送が可能。
- ミュートモード。
- ステレオ／モノラルオーディオフレーム機能。
- 通信クロックストローブエッジが設定可能（SCK）。
- 対応する割込み付きのエラーフラグ（それぞれイネーブル付き）。
  - オーバーラン検出およびアンダーラン検出、
  - スレーブモードにおける予測フレーム同期信号検出、
  - スレーブモードにおけるレイトフレーム同期信号検出、
  - 受信時AC’97モードに対するコーデックノットレディ

- 割込みソース（有効時）:
  - エラー
  - FIFOリクエスト
- 2チャネルDMAインタフェース

69.3 SAIの実装

表698.SAIの機能(1)

SAIの機能	SAI1	SAI2(2)
I2S、LSB/MSB詰め、PCM/DSP、TDM、AC'97	X	X
FIFOサイズ	8ワード	8ワード
SPDIF	X	X
PDM	X(3)	-

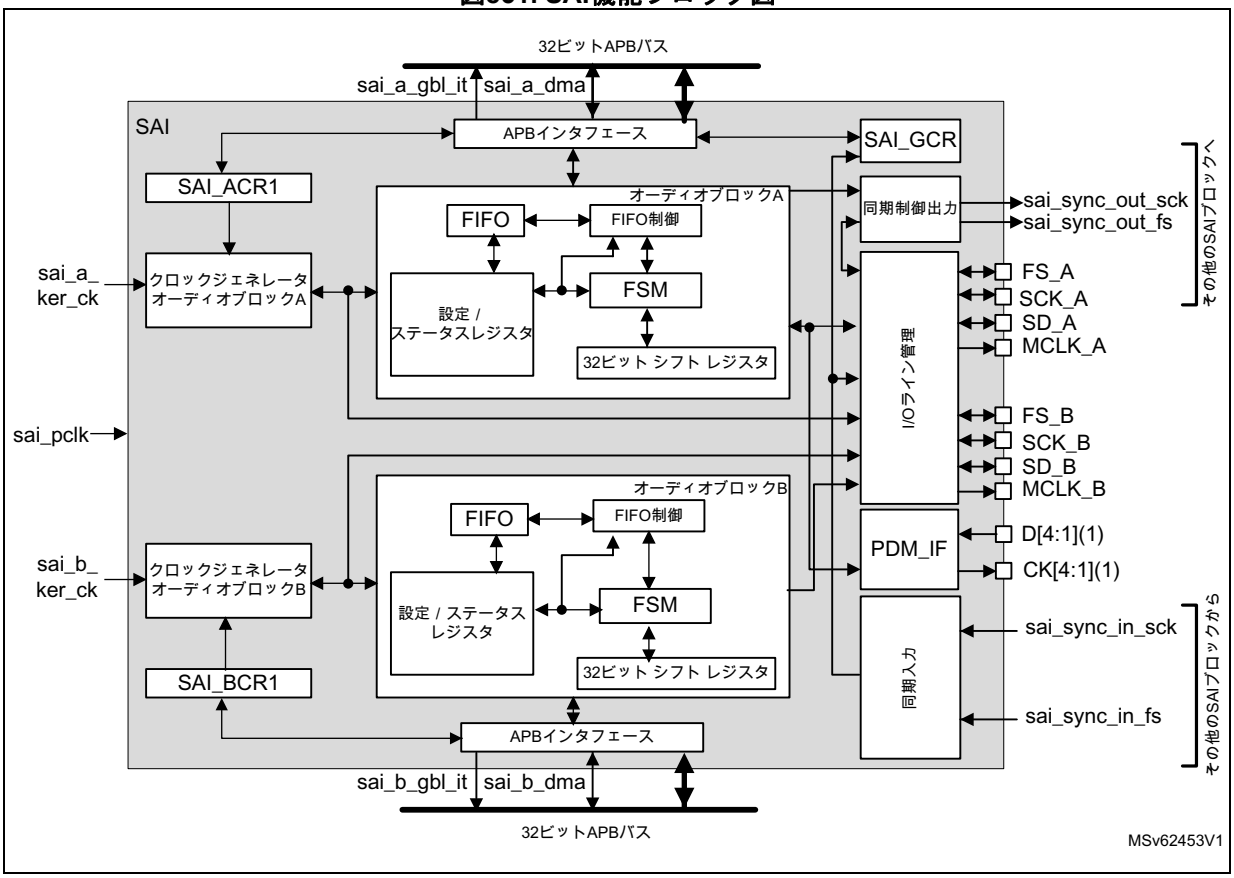
1. “X” はサポートされています。“-” はサポートされていません。
2. このインスタンスはSTM32U535/545デバイスでは使用できません。
3. D[3:1] およびCK[2:1] の信号のみ使用可能です。

69.4 SAIの機能説明

69.4.1 SAIブロック図

図851にSAIブロック図を、表699および表700にSAI内部および外部信号のリストを示します。

図851. SAI機能ブロック図



1. これらの信号は、すべてのSAIインタフェースで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

このSAIは大部分が2つのオーディオサブブロックから構成されており、それぞれに固有のクロックジェネレータを装備しています。各オーディオブロックは、それぞれの機能ステートマシンによって制御される32bitシフトレジスタを内蔵しています。データは、専用のFIFOに保存されたり、そこから読み出されたりします。FIFOはCPUからアクセスすることもできれば、通信時にCPUを開放するためDMAからアクセスすることもできます。各オーディオブロックは独立しています。互いに同期することもできます。

I/Oラインコントローラは、SAI内の指定オーディオブロックに対する4本の専用ピン（SD、SCK、FS、MCLK）を管理します。この2つのサブブロックを同期するように指定すると、いくつかのピンを共用していくつかのピンを開放し、汎用I/Oとして使用することができます。アプリケーション、デコーダ要件、オーディオブロックがマスタとして設定されているか否かによって、MCLKピンを出力にしたり、それ以外にしたりすることができます。

1つのSAIが別のSAIと同期して動作するように設定されている場合、より多くのI/Oを開放することができます（SD_xピンを除く）。

この機能ステートマシンは、広範囲なオーディオプロトコルを処理するように設定することができます。所望のプロトコルを設定するため、いくつかのレジスタがあります（オーディオフレーム波形ジェネレータ）。

オーディオサブブロックは、マスタモードまたはスレーブモードでトランスミッタまたはレシーバになることができます。マスタモードでは、SCK_xビットクロックとフレーム同期信号がSAIで生成されますが、スレーブモードでは、これらの信号を別の外部マスタまたは内部マスタから受け取ります。FS信号の方向がマスタモードまたはスレーブモードの定義に直接結びつかない特別なケースがあります。AC'97プロトコルでは、SAI（リンクコントローラ）がSCKクロックを入力するように設定されている場合でも（つまりスレーブモードでも）、FS信号はSAIの出力です。

注： **読みやすくするために、このセクションではSAI_AまたはSAI_BをSAI_xと表記しています。「x」とはSAI Aサブブロックまたは SAI Bサブブロックのことです。**

69.4.2 SAIピンおよび内部信号

表699.SAI内部入力／出力信号

内部信号名	信号タイプ	説明
sai_a_gbl_it/ sai_b_gbl_it	出力	オーディオブロックAおよびBグローバル割込み
sai_a_dma、sai_b_dma	入力／出力	オーディオブロックAおよびB DMA確認応答およびリクエスト
sai_sync_out_sck、sai_sync_out_fs	出力	他のSAIブロックと交換される内部クロックおよびフレーム同期出力信号
sai_sync_in_sck、sai_sync_in_fs	入力	他のSAIブロックと交換される内部クロックおよびフレーム同期入力信号
sai_a_ker_ck/ sai_b_ker_ck	入力	オーディオブロックA/Bカーネルクロック
sai_pclk	入力	APBクロック

表700.SAIの入出力ピン

名前	信号タイプ	コメント
SAI_SCK_A/B	入力／出力	オーディオブロックA/Bビットクロック
SAI_MCLK_A/B	出力	オーディオブロックA/Bマスタクロック
SAI_SD_A/B	入力／出力	ブロックA/Bのデータライン
SAI_FS_A/B	入力／出力	オーディオブロックA/Bのフレーム同期ライン
SAI_CK[4:1]	出力	PDMビットストリームクロック(1)
SAI_D[4:1]	入力	PDMビットストリームデータ (1)

1. これらの信号は、すべてのSAIインタフェースで使用できるわけではありません。詳細については、セクション 69.3 : SAIの実装を参照してください。

69.4.3 主要なSAIモード

SAIの各オーディオサブブロックは、選択したオーディオブロックのSAI_xCR1レジスタのMODEビットを使ってマスタまたはスレーブに設定することができます。

マスタモード

マスタモードでは、SAIは接続された外部デバイスへタイミング信号を送り出します。

- ビットクロックとフレーム同期はそれぞれSCK_xピンとFS_xピンに出力されます。
- 必要であれば、SAIはMCLK_xピンにマスタクロックを生成することもできます。

SCK_x、FS_x、およびMCLK_xのいずれも出力として設定されます。

スレーブモード

SAIは、外部デバイスからのタイミング信号の受信を待ちます。

- SAIサブブロックが非同期モードに設定されている場合、SCK_xピンおよびFS_xピンは入力として設定されます。
- SAIサブブロックが他のSAIインタフェースまたは2番目のオーディオサブブロックと同期して動作するよう設定されている場合、対応するSCK_xおよびFS_xピンは汎用I/Oとして使用できます。

MCLK_xピンはスレーブモードでは使用されず、別の機能に割り当てることができます。

スレーブデバイスを有効にした後でマスタを有効にすることをお勧めします。

SAIモードの設定と有効化

各オーディオサブブロックは、該当するオーディオブロックにあるSAI_xCR1レジスタのMODEビットを使って個別にトランスミッタまたはレシーバに指定することができます。そのため、SAI_SD_xピンは出力または入力としてそれぞれ設定されます。

同じSAIにある2個のマスタオーディオブロックは、2つの異なるMCLKおよびSCKクロック周波数で設定することができます。この場合、オーディオブロックは非同期モードに設定する必要があります。

SAI内の各オーディオブロックは、SAI_xCR1レジスタのSAIENビットにより有効化されます。このビットがアクティブになると直ちに、トランスミッタまたはレシーバはスレーブモードでクロックライン、データライン、同期ラインの動作に反応するようになります。

マスタTXモードでは、オーディオブロックを有効化すると直ちに、FIFOにデータがなくとも、外部スレーブに対するビットクロックを生成しますが、FS信号の生成は、FIFO内のデータの存在が条件となります。FIFOが先頭の送信データを受け取った後、このデータが外部スレーブに出力されます。FIFOに送信データがない場合、オーディオフレームにおいては0値が送信されて、アンダーランフラグが生成されます。

スレーブモードでは、オーディオブロックが有効化されて、フレームの開始が検出されたときに、オーディオフレームが開始されます。

スレーブTXモードでは、オーディオブロックを有効化した後の最初のフレームでアンダーランイベントが生成されることはありません。理由は、次の動作シーケンスが必要となるためです。

1. ソフトウェアまたはDMAによりSAI_xDRへ書込みを行います。
2. FIFO閾値（FLH）フラグが0b000（FIFOエンプティ）でなくなるまで待ちます。
3. スレーブ送信ミッタモードでオーディオブロックを有効化します。

69.4.4 SAI同期モード

オーディオサブブロックレベルとSAIレベルの2種類の同期レベルがあります。

内部同期

オーディオサブブロックは、同じSAI内の2番目のオーディオサブブロックと同期して動作できるよう設定できます。この場合、ビットクロック信号とフレーム同期信号が共用され、通信に使用する外部ピン数を少なくすることができます。同期モードに設定されたオーディオブロックでは、そのSCK_x、FS_x、およびMCLK_xピンはGPIOとして開放状態に戻されています。非同期モードに設定されたオーディオブロックのほうが、FS_x、SCK_x、MCLK_xのI/Oピンに適したブロックとなります（オーディオブロックをマスタと考えた場合）。

一般に、同期モードのオーディオブロックを使って、SAIを全二重モードに設定することができます。2個のオーディオブロックの片方をマスタとして設定し、他方をスレーブとして設定することができます。あるいは、両方をスレーブとして設定し、1つは非同期ブロック(SAI_xCR1 の対応するSYNCEN[1:0] ビットを00にセット)に、もう一方を同期ブロック（SAI_xCR1の対応するSYNCEN[1:0] ビットを01にセット）にすることもできます。

注： **内部再同期ステージのため、PCLK APB周波数はビットレートクロック周波数の2倍以上である必要があります。**

外部同期

オーディオサブブロックは他のSAIと同期して動作するよう設定することもできます。この方法は以下のとおりです。

1. SAIが、他のSAIの同期元として設定されている場合は、その相手のSAIにFSおよびSCK信号を提供するオーディオサブブロックを定義する必要があります。これはSYNCOUT[1:0] ビットを設定することによって行います。
2. 同期信号を受信するSAIは、SYNCIN[1:0] ビットに適正な値をセットすることによって同期を提供するSAIを選択する必要があります。2個のSAIオーディオサブブロックのそれぞれに対して、ユーザはそのサブブロックが他方のSAIと同期して動作するかどうかをSYNCENビットで指定します。

注： **SYNCIN[1:0] およびSYNCOUT[1:0] ビットはSAI_GCRレジスタ内にあり、SYNCENビットはSAI_xCR1レジスタ内にあります。**

SAIの2個のオーディオサブブロックがどちらも他のSAIと同期して動作する必要がある場合は、次の設定のいずれかを選択することができます。

- 他のSAIブロックと同期させる各オーディオブロックをSYNCEN[1:0] ビットにより設定します。
- 他のSAIと同期させる1個のオーディオブロックをSYNCEN[1:0] ビットにより設定します。このとき、もう一方のオーディオブロックは2番目のSAIオーディオブロックと同期させるように、SYNCEN[1:0] ビットによって設定されます。

下表は、使用されるSAIブロックに応じた適切な同期信号の選び方を示します。たとえば、SAI2はSAI2 SYNCINを0に設定することで、SAI1からの同期を選択できます。SAI1に SAI2からの同期を選択させたい場合は、SAI1 SYNCINを 1にセットする必要があります。「予約済み」と記載されている位置は使用できません。

表701.外部同期の選択

ブロックの例	SYNCIN = 1	SYNCIN = 0
SAI1	SAI2 同期	予約済みです。
SAI2	予約済み	SAI1 同期

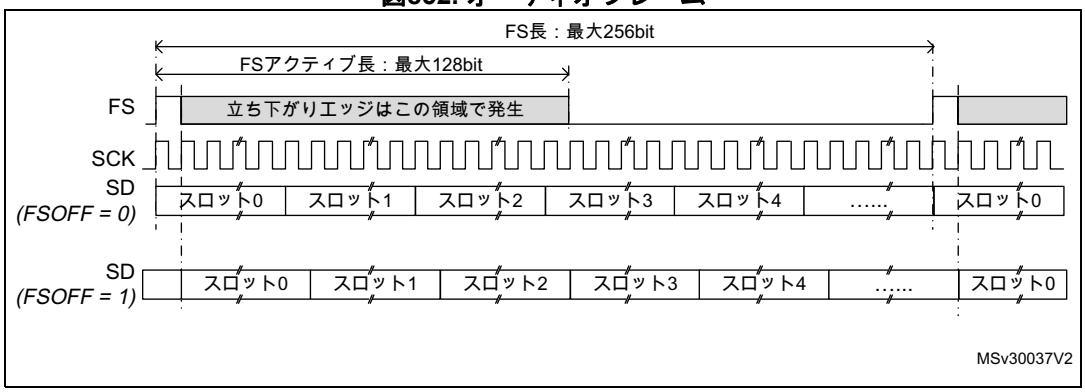
69.4.5 オーディオデータサイズ

オーディオフレームは、SAI_xCR1レジスタのDS[2:0] ビットを設定することにより、様々なデータサイズを対象とすることができます。データサイズとしては、8、10、16、20、24、32bitが可能です。転送時、SAI_xCR1レジスタのLSBFIRSTビットの設定によってデータのMSBを先頭にして送信することもできればLSBを先頭にすることもできます。

69.4.6 フレーム同期

FS信号は、オーディオフレームにおいてフレーム同期信号として機能します（フレーム開始）。この信号の波形は、このフレーム同期動作に関する独自の仕様を持つ様々なオーディオプロトコルに対応するため、全面的に設定可能となっています。この再設定はSAI_xFRCRレジスタを使って実行されます。図852にこの柔軟性について図示します。

図852. オーディオフレーム



AC’97モードまたはSPDIFモード（SAI_xCR1レジスタのPRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01）では、フレーム同期波形を強制的にAC’97プロトコルに適合させます。SAI_xFRCRレジスタ値は無視されます。

各オーディオブロックは独立であるため、それぞれに固有の設定が必要となります。

フレーム長

- マスタモード

  オーディオフレーム長は、SAI_xFRCRレジスタのFRL[7:0] ビットを設定して、最大256bitクロックに設定することができます。

  フレーム長がフレームに対して指定されたスロット数より大きい場合、SAI_xCR2レジスタのTRISビットの状態に応じて、残りの送信ビットを0に拡張するか、SDラインをハイインピーダンス（HI-Z）状態にします（FS信号の機能を参照）。受信モードでは、残りのビットは無視されます。

  NODIV ビットがクリアされている場合、(FRL + 1)は8から256のうちの2の累乗の数に調整する必要があります。これは、オーディオフレームがビットクロックサイクル当たり整数個のMCLKパルスを含むようにするためです。

  NODIV ビットがセットされている場合、(FRL + 1) フィールドは8から256のうちの任意の値をとることができます。セクション 69.4.8：SAIクロックジェネレータを参照してください。

- スレーブモード

  オーディオフレーム長は主に、外部マスタから送信されるオーディオフレーム当たりのビットクロック数をスレーブに対して指定するために使用されます。また、処理中のオーディオフレームでマスタからの想定されるフレーム同期信号または遅延したフレーム同期信号の発生を検出す

るためにも使用されます。この場合、エラーが生成されます。詳細については、セクション 69.4.14：エラーフラグを参照してください。

スレーブモードでは、SAI_xFRCRレジスタのFRL[7:0] ビット設定に制約はありません。

フレーム内のビット数はFRL[7:0] + 1に等しくなります。

オーディオフレーム内で転送する最小ビット数は8です。

フレーム同期極性

SAI_xFRCRレジスタのFSPOLビットにより、フレームが開始されるFSピンのアクティブ極性が設定されます。フレーム開始は、エッジに反応します。

スレーブモードにおいて、オーディオブロックは有効なフレームを待って、送信または受信を開始します。フレーム開始はこの信号に同期します。これは、通信中のフレーム開始が検出されず、予測のフレーム開始と一致しない場合にのみ有効です（セクション 69.4.14：エラーフラグを参照）。

マスタモードでは、SAI_xCR1レジスタのSAIENビットがクリアされるまで、オーディオフレームが完了するごとにフレーム同期が毎回送信されます。前のオーディオフレームの終了時にFIFOにデータがない場合、アンダーラン状態はセクション 69.4.14：エラーフラグの説明のように扱われますが、オーディオ通信フローは中断されません。

フレーム同期アクティブレベル長

SAI_xFRCRレジスタのFSALL[6:0] ビットにより、フレーム同期信号のアクティブレベル長の設定が可能になります。この長さは、1から128のビットクロックに設定することができます。

例えば、I2S、LSB詰めまたはMSB詰めモードでは、アクティブ長をフレーム長の1/2に、PCM/DSP、TDMでは1bit幅に、それぞれ設定することができます。

フレーム同期オフセット

アプリケーションで対象とするオーディオプロトコルに応じて、オーディオフレームの最終ビットまたは先頭ビットを送信したときフレーム同期信号をアサートすることができます（例えば、それぞれI2S規格プロトコルおよびMSB詰めプロトコルの場合）。SAI_xFRCRレジスタのFSOFFビットにより、2つの設定のいずれかを選択可能です。

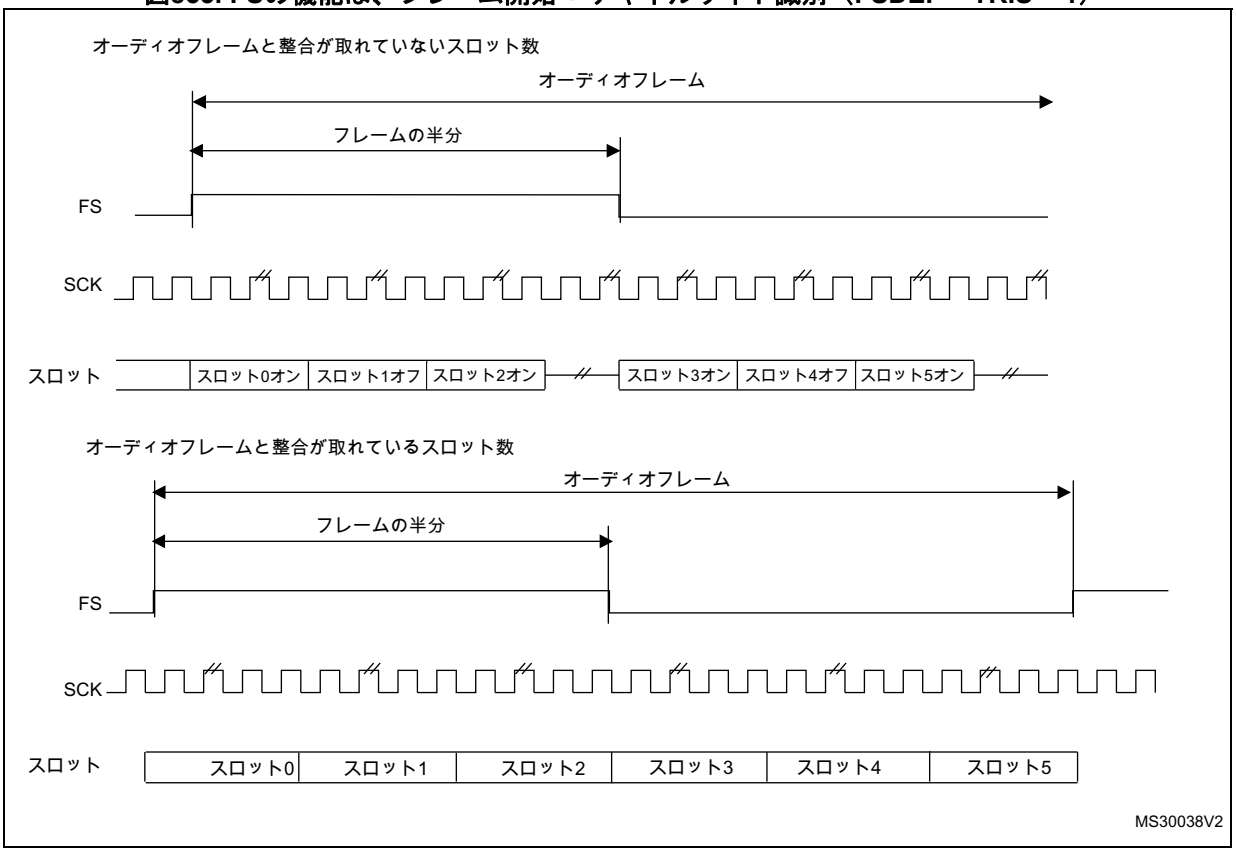
FS信号の機能

FS信号は、FSの機能によって意味が異なります。SAI_xFRCRレジスタのFSDEFビットにより意味を選択します。

- 0：例えばPCM/DSP、TDM、AC’97オーディオプロトコルのようなフレーム開始
- 1：例えばI2S、MSB詰めまたはLSB詰めプロトコルのような、オーディオフレーム内のチャネルサイド識別およびフレーム開始

FS信号をフレーム開始およびフレーム内チャネルサイド識別と見なす場合、指定スロット数は半分を左チャネル用、もう半分を右チャネル用と見なす必要があります。ハーフオーディオフレームのビットクロック数がチャネルサイド専用スロット数より大きい場合、TRIS = 0であれば、SAI_xCR2レジスタの残りのビットクロックに対して0が送信されます。そうしないと、TRIS = 1の場合、SDラインがハイインピーダンスに開放されます。受信モード時、残りのビットクロックサイクルはチャネルサイドが変化するまで無視されます。

図853. FSの機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1）

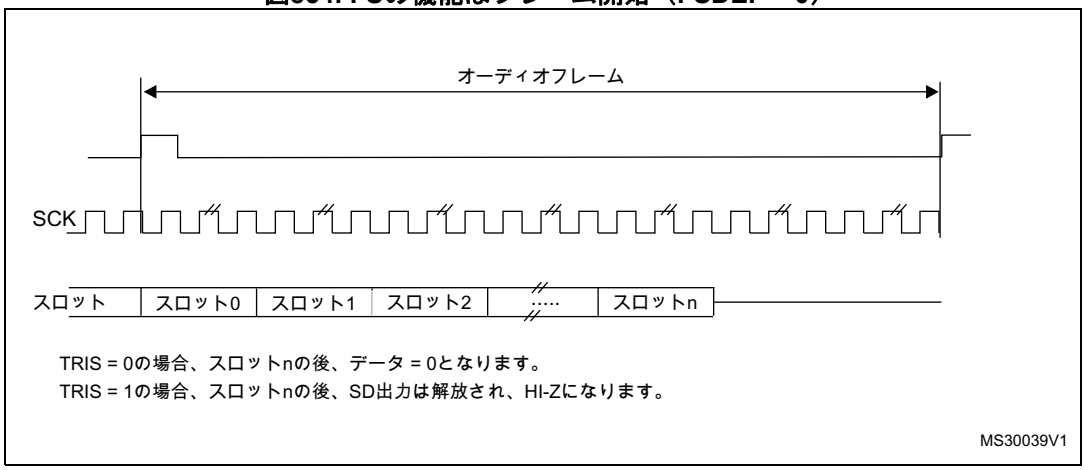


1. フレーム長は偶数である必要があります。

SAI_xFRCRのFSDEFビットがクリアされたままの場合、FS信号はフレーム開始に相当し、さらにSAI_xSLOTRのNBSLOT[3:0] ビットで指定されたスロット数にSAI_xSLOTRのSLOTSZ[1:0] ビットで設定されたスロット当たりのビット数を乗算した値が、フレームサイズ （SAI_xFRCRレジスタのFRL[7:0] ビット）より小さい場合、

- SAI_xCR2レジスタでTRIS = 0のとき、送信の場合はフレームの終わりまで、最終スロット後の残りのビットは強制的に0に設定されます。
- TRIS = 1のとき、これら残りのビットの転送中、ラインはハイインピーダンス（HI-Z）になります。受信モードでは、これらのビットは破棄されます。

図854. FSの機能はフレーム開始（FSDEF = 0）



送信モードでオーディオブロックがSDラインのSPDIF出力を取得するように設定されている場合は、FS信号は使用されません。対応するFSのI/Oは開放され、他の目的のために空けられます。

69.4.7 スロットの設定

スロットは、オーディオフレームの基本要素です。オーディオフレーム内のスロット数はNBSLOT[3:0] + 1に等しくなります。

オーディオフレーム当たりの最大スロット数は16に固定されています。

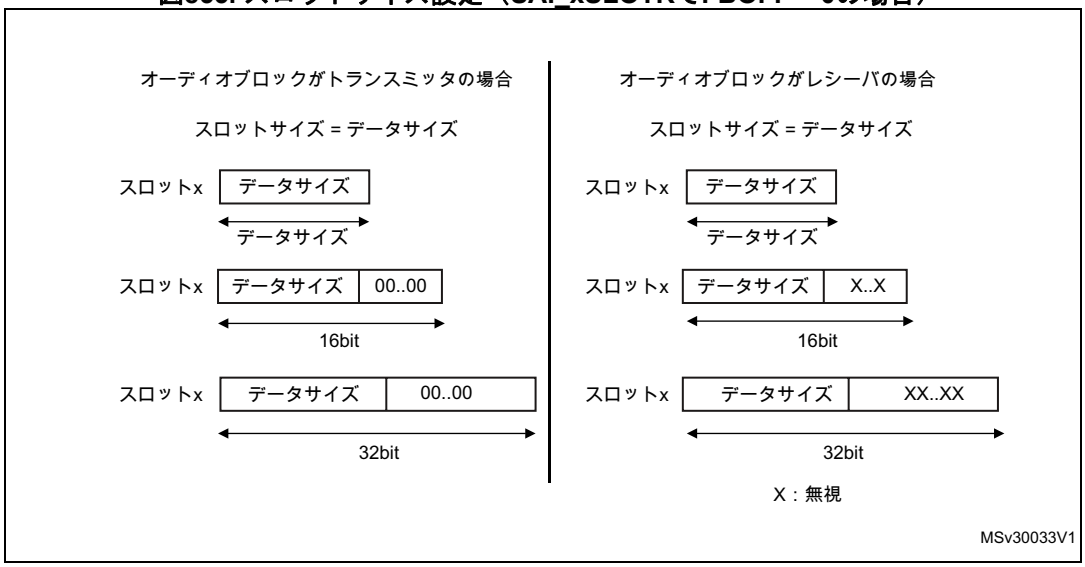
AC'97プロトコルまたはSPDIF（PRTCFG[1:0] ビット = 10 またはPRTCFG[1:0] ビット = 01の場合）に対しては、スロット数は、そのプロトコル仕様を対象とするよう自動的に設定され、NBSLOT[3:0]の値は無視されます。

各スロットは、SAI_xSLOTRレジスタのSLOTEN[15:0] ビットにより有効/無効を指定することができます。

送信モードでは、無効なスロットが転送されると、SDデータラインはTRISビットの設定に応じて強制的に0に設定されるか、または開放されてハイインピーダンス（HI-Z）状態になります（非アクティブスロットでの出力データライン管理を参照）。受信モードでは、このスロットの終わりからの受信した値は無視されます。結果として、FIFOアクセスがないため、この非アクティブなスロット状態にリンクしたFIFOからの読出しリクエストやFIFOへの書込みリクエストは起こりません。

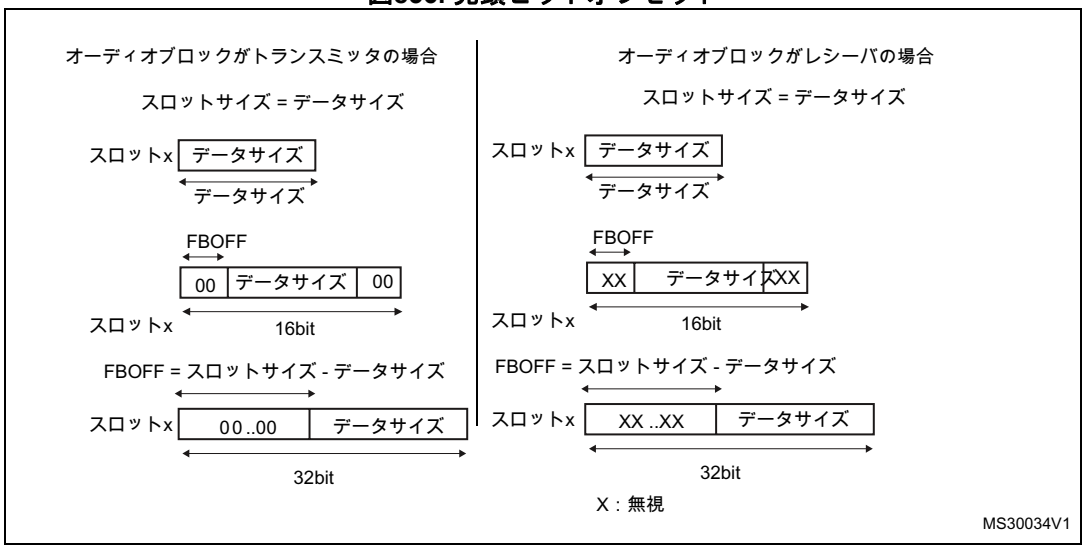
また、スロットのサイズも図855に示すように設定可能です。スロットのサイズは、SAI_xSLOTRレジスタのSLOTSZ[1:0] ビットを設定することで選択できます。サイズは、オーディオフレームの各スロットに同じように適用されます。

図855. スロットサイズ設定（SAI_xSLOTRでFBOFF = 0の場合）



スロット内で転送する先頭データビットの位置を選択することができます。このオフセットは、SAI_xSLOTRレジスタのFBOFF[4:0] ビットによって設定されます。送信モードでは、スロットの開始からこのオフセット位置に到達するまでの間0値が挿入されます。受信時、オフセットフェーズ内のビットは無視されます。この機能はLSB詰めプロトコルを対象としています （オフセットが、スロットサイズ - データサイズに等しい場合）。

図856. 先頭ビットオフセット



SAIの誤動作を避けるためには、次の条件を満足する必要があります。

FBOFF ≤(SLOTSZ - DS)、

DS ≤SLOTSZ、

NBSLOT x SLOTSZ ≤FRL（フレーム長）、

SAI_xFRCRレジスタのFSDEFビットがセットされている場合でも、スロット数は偶数でなければなりません。

AC’97 およびSPDIFプロトコル（PRTCFG[1:0] ビット = 10 またはPRTCFG[1:0] ビット = 01）では、スロットサイズは、セクション 69.4.11 : AC’97リンクコントローラで示されるように自動的に設定されます。

69.4.8 SAIクロックジェネレータ

各オーディオブロックには専用のクロックジェネレータが備わっています。クロックジェネレータは、sai_x_ker_ckからマスタクロック（MCLK_x）およびビットクロック（SCK_x）信号を作成します。sai_x_ker_ckクロックは、製品のクロックコントローラ（RCC）によって供給されます。

マスタクロック（MCLK_x）の生成

クロックジェネレータは、オーディオブロックがマスタまたはスレーブとして定義されている場合にマスタクロック（MCLK_x）を供給します。マスタクロックは、該当ブロックのSAIENビットが0にセットされていても、MCKENビットを1にセットするとすぐに、生成されます。この機能は、オーディオストリームをアクティブにする前にMCLK_xを生成可能にするため、MCLK_xクロックが外部オーディオデバイスのシステムクロックとして使用される場合に役立ちます。

オーディオサンプルを転送する前にMCLK_x出力にマスタクロックを生成するには、ユーザアプリケーションは以下のシーケンスに従う必要があります。

1. SAIEN = 0であることを確認します。
2. MCKDIV[5:0] 分周器を必要な値にプログラムします。
3. MCKENビットを1にセットします。
4. 後に、アプリケーションはSAIの他の部分を構成し、SAIENビットを1に設定してオーディオサンプルの転送を開始できます。

MCLK_x出力に生成されたクロックの乱れを避けるため、次の操作は推奨されません。

- MCKEN = 1のときに、MCKDIVを変更すること
- SAIEN = 1の場合に、MCKENを0にセットすること

SAIでは、MCLK_xが（SAIEN = 0の状態で）MCKENビットによってオンおよびオフに切り替えられたときに、MCLK_x出力にスプールがないことを保証します。

表702に、MCLK_xの有効化条件を示します。

表702.MCLK_xの有効化条件

MCLKEN	NODIV	ブロックxのSAIEN	MCLK_x
0	X	0	無効
1			有効
0	1	1	無効
1			有効
X	0		有効

注： **MCLKENが1にセットされている場合、MCLK_xはAC’97モードで生成することもできます。**

ビットクロック（SCK_x）の生成

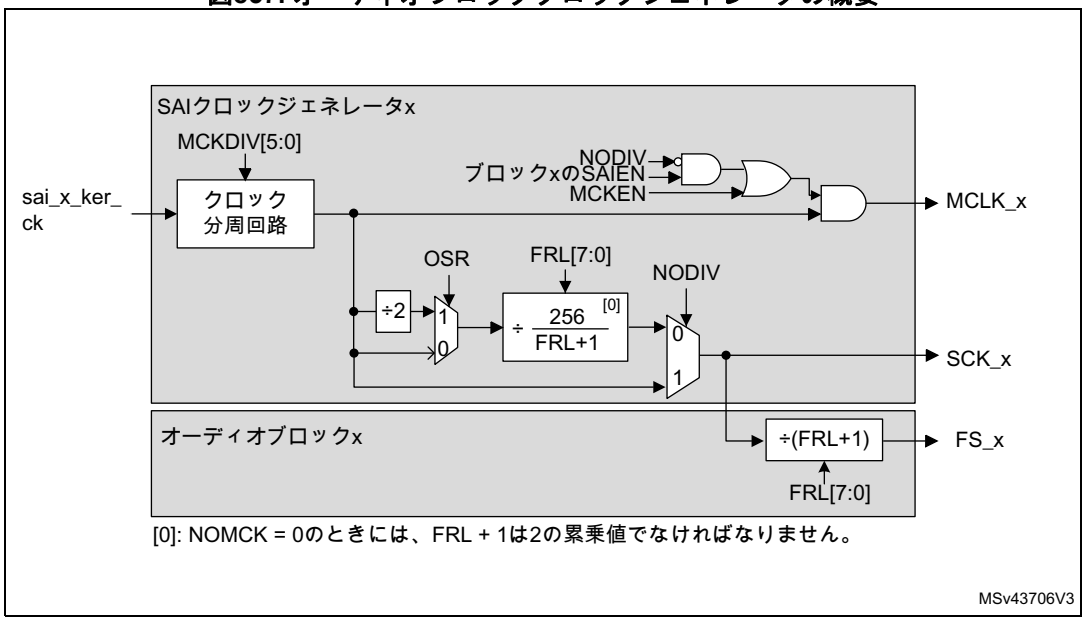
クロックジェネレータは、オーディオブロックがマスタとして定義されている場合にビットクロック（SCK_x）を供給します。フレーム同期（FS_x）は、クロックジェネレータから供給される信号からも得られます。

スレーブモードでは、NODIVおよびOSRフィールドの値は無視され、SCK_xクロックは生成されません。

SCK_xのビットクロックストローブエッジは、CKSTRフィールドで設定することができ、これはマスタモードとスレーブモードの両方で機能します。

図857に、オーディオブロッククロックジェネレータのアーキテクチャを示します。

図857. オーディオブロッククロックジェネレータの概要



マスタクロック（MCLK_x）とフレーム同期（FS_x）周波数の比率を256または512に強制するために、NODIVビットを使用する必要があります。

- NODIVを0にセットした場合、フレーム同期とマスタクロックとの周波数比は、OSR値に従って512または256に固定されますが、フレーム長は2の累乗値でなければなりません。詳細については、以下を参照してください。
- NODIVを1にセットした場合、ビットクロック（SCK_x）の周波数は、アプリケーションによって、MCKDIVを介して調整できます。さらに、フレーム長が8以上（すなわち、FRL[7:0] > 6）であるかぎり、フレーム長の値に何ら制約はありません。フレーム同期周波数はMCKDIVおよびフレーム長（FRL[7:0]）に依存します。この場合、MCLK_xの周波数はSCK_xに等しくなります。

NODIV、MCKEN、SAIEN、OVR、CKSTR、およびMCKDIV[5:0] ビットはSAI_xCR1レジスタに属しますが、FRL[7:0] ビットはSAI_xFRCRレジスタに属します。

NODIV = 0のときのクロックジェネレータのプログラミング

この場合、MCLK_x周波数は次のようになります。

- $F_{MCLK_x} = 256 \times F_{FS_x}$ （OSR = 0の場合）
- $F_{MCLK_x} = 512 \times F_{FS_x}$ （OSR = 1の場合）

MCKDIVが0ではない場合、MCLK_x周波数は下の計算式によって与えられます。

$$F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期周波数は次のようになります。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV \times (OSR + 1) \times 256}$$

ビットクロック周波数（SCK_x）は次の計算式で与えられます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck} \times (FRL + 1)}{MCKDIV \times (OSR + 1) \times 256}$$

注： **NODIVが0に等しい場合、（FRL+1）は2の累乗値でなければなりません。加えて、（FRL+1）は8～256でなければなりません。（FRL+1）はオーディオフレーム内のビットクロック数を表します。**

MCKDIV分周比が奇数である場合、MCLKのデューティサイクルは50%ではありません。MCKDIVが奇数、OSRが0に等しく、かつ（FRL+1）= 2^8であると、ビットクロック信号（SCK_x）についてもデューティサイクルが50% とは異なることがあります。

MCKDIVを偶数か大きな値（10を超える値）にプログラミングすることを推奨します。

MCKDIV = 0であると、MCKDIV = 1のときと同じ結果となることに注意してください。

NODIV = 1のときのクロックジェネレータのプログラミング

MCKDIVが0ではない場合、ビットクロック（SCK_x）周波数は下の計算式によって与えられます。

$$F_{SCK_x} = F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期（FS_x）周波数は次の計算式によって与えられます。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{(FRL + 1) \times MCKDIV}$$

注： **NODIVが1にセットされている場合、（FRL+1）は8から256のうちの任意の値をとることができます。**

MCKDIV = 0であると、MCKDIV = 1のときと同じ結果となることに注意してください。

クロックジェネレータのプログラミング例

表703に、48、96、192kHzに対するプログラミング例を示します。

表703.クロックジェネレータのプログラミング例

入力sai_x_ker_ck クロック周波数	MCLK	F_{MCLK}/ F_{FS}	FRL (1)	OSR	NODIV	MCKEN	MCKDIV[5:0]	オーディオ サンプリング周波数 (F_{FS})
98.304MHz	Y	512	2^N-1	1	0	1	0または1	192kHz
		512	2^N-1	1	0	1	2	96kHz
		512	2^N-1	1	0	1	4	48kHz
		256	2^N-1	0	0	1	2	192kHz
		256	2^N-1	0	0	1	4	96kHz
		256	2^N-1	0	0	1	8	48kHz
	N	-	63	-	1	0	8	192kHz
		-	63	-	1	0	16	96kHz
		-	63	-	1	0	32	48kHz

1. Nは3～8の整数値です。

69.4.9 内部FIFO

SAI内の各オーディオブロックには固有のFIFOが搭載されています。そのブロックがトランスミッタに指定されるかレシーバに指定されるかによって、FIFOにそれぞれ書き込んだり読み出したりすることができます。したがって、SAI_xSRレジスタのFREQビットに対応したFIFOリクエストは1つしかありません。

SAI_xIMレジスタのFREQIEビットが有効化されている場合、割込みが生成されます。これは次の条件によって変わります：

- FIFO閾値設定（SAI_xCR2のFLVLビット）
- 通信方向（トランスミッタまたはレシーバ）送信モードでの割込み生成および受信モードでの割込み生成を参照してください。

送信モードでの割込み生成

送信モードでの割込みの生成は、次のようにFIFOの設定によって変わります：

- SAI_xCR2レジスタのFIFO閾値ビットがFIFOエンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDRレジスタに有効なデータがない（SAI_xSRのFLVL[2:0] ビット<001b）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOがエンプティ（空）でなくなった場合（SAI_xSRのFLVL[2:0] ビット≠0b000）、すなわちFIFOに1個以上のデータが格納されている状態になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフルの1/4（FTH[2:0] = 001b）に設定されている場合、データがFIFOの1/4より少ない（SAI_xSRのFLVL[2:0] ビット<0b010）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOの1/4以上にデータがある状態（SAI_xSRのFLVL[2:0] ビット≧0b010）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOフルの1/2（FTH[2:0] = 0b010）に設定されている場合、データがFIFOの1/2より少ない（SAI_xSRのFLVL[2:0] ビット<011b）という条件で、割込み

が生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOの1/2以上にデータがある状態（SAI_xSRのFLVL[2:0] ビット≧011b）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。

- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフルの3/4（FTH[2:0] = 011b）に設定されている場合、データがFIFOの3/4より少ない（SAI_xSRのFLVL[2:0] ビット<0b100）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOの3/4以上にデータがある状態（SAI_xSRのFLVL[2:0] ビット≧0b100）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフル（FTH[2:0] = 0b100）に設定されている場合、FIFOがフルでない（SAI_xSRのFLVL[2:0] ビット<101b）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOデータがフル（SAI_xSRのFLVL[2:0] ビット = 101b）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。

受信モードでの割込み生成

受信モードでの割込みの発生は、次のようにFIFOの設定によって変わります：

- SAI_xCR2レジスタのFIFO閾値ビットがFIFOエンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDRレジスタに1個以上のデータがある（SAI_xSRのFLVL[2:0] ビット ≧ 001b）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOがエンプティになる（SAI_xSRのFLVL[2:0] ビット = 0b000）、すなわちFIFOにデータがなくなると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフルの1/4（FTH[2:0] = 001b）に設定されている場合、FIFOデータロケーションの1/4以上が使用可能（SAI_xSRのFLVL[2:0] ビット ≧ 0b010）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。使用可能なFIFOデータロケーションが1/4未満（SAI_xSRのFLVL[2:0] ビット < 0b010）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフルの1/2（FTH[2:0] = 0b010）に設定されている場合、FIFOデータロケーションの1/2以上が使用可能（SAI_xSRのFLVL[2:0] ビット≧011b）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。使用可能なFIFOデータロケーションが1/2未満（SAI_xSRのFLVL[2:0] ビット < 011b）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフルの3/4（FTH[2:0] = 011b）に設定されている場合、FIFOデータロケーションの3/4以上が使用可能（SAI_xSRのFLVL[2:0] ビット≧0b100）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。使用可能なFIFOデータロケーションが3/4未満（SAI_xSRのFLVL[2:0] ビット < 0b100）になると、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。
- SAI_xCR2レジスタのFIFO閾値ビットがFIFOがフル（FTH[2:0] = 0b100）に設定されている場合、FIFOがフル（SAI_xSRのFLVL[2:0] ビット = 101b）という条件で、割込みが生成されます（SAI_xSRレジスタのFREQビットがハードウェアによって1にセットされます）。FIFOデータがフルでなくなると（SAI_xSRのFLVL[2:0] ビット < 101b）、この割込み（SAI_xSRレジスタのFREQビット）はハードウェアによってクリアされます。

SAI_xCR1レジスタのDMAENビットがセットされている場合、SAIは、割込み生成と同様にDMAを使用することができます。FREQビットアサーションメカニズムは、上記FREQIEについて説明した割込み生成メカニズムと同じです。

各FIFOは8ワードのFIFOです。FIFOに対する書込み操作や読出し操作は、そのアクセスサイズによらず、1回につきFIFOの1ワード分を対象とします。FIFOの1ワードには1オーディオスロットが格納されます。FIFOポインタは、SAI_xDRレジスタに対する各アクセスの後に1ワードずつインクリメントされます。

SAI_xDRに書き込む際、データは右詰めである必要があります。

受信データはSAI_xDRで右詰めとなっています。

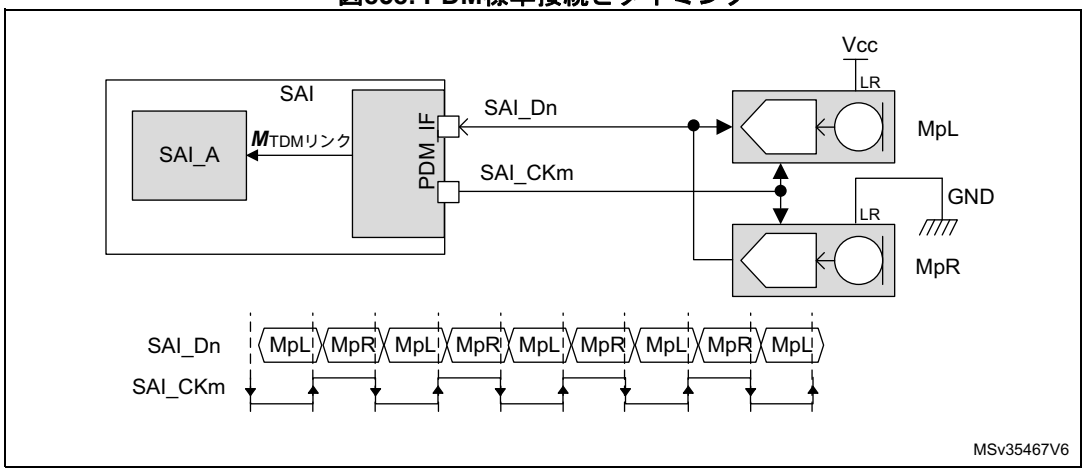
SAI_xCR2レジスタのFFLUSHビットをセットしてSAIを無効化すると、FIFOポインタを再初期化することができます。SAIが有効なときにFFLUSHをセットすると、FIFO内にあるデータは自動的に失われます。

69.4.10 PDMインタフェース

PDM（パルス密度変調）インタフェースは、デジタルマイクロフォン対応のために提供されています。最大4組のデジタルマイクロフォンのペアに並列接続可能です。製品の実装に応じて、対応可能なマイクロフォンの数は少なくなります（セクション 69.3：SAIの実装を参照）。

図858に、PDMインタフェースを介したデジタルマイクロフォンのペアの標準的な接続を示します。どちらのマイクロフォンも同じビットストリームクロックとデータのラインを共有しています。設定ピン（LR）によって、片方のマイクロフォンがSAI_CK[m]の立上りエッジで有効なデータを供給するのと同時に、もう一方がSAI_CK[m]の立下りエッジで有効なデータを供給します（mはクロックライン数）。

図858. PDM標準接続とタイミング



1. **n**はデータライン数、**p**はマイクロフォンのペア数を表します。

PDM機能は、TDMマスタモードで設定されたSAI_Aサブクロックとともに使用されるためのものです。SAI_Bサブクロックとは使用できません。PDMインタフェースは、SAI_AのTDMインタフェースから供給されるタイミング信号を使用して、それを適合させてビットストリームクロック（SAI_CK[m]）を生成します。

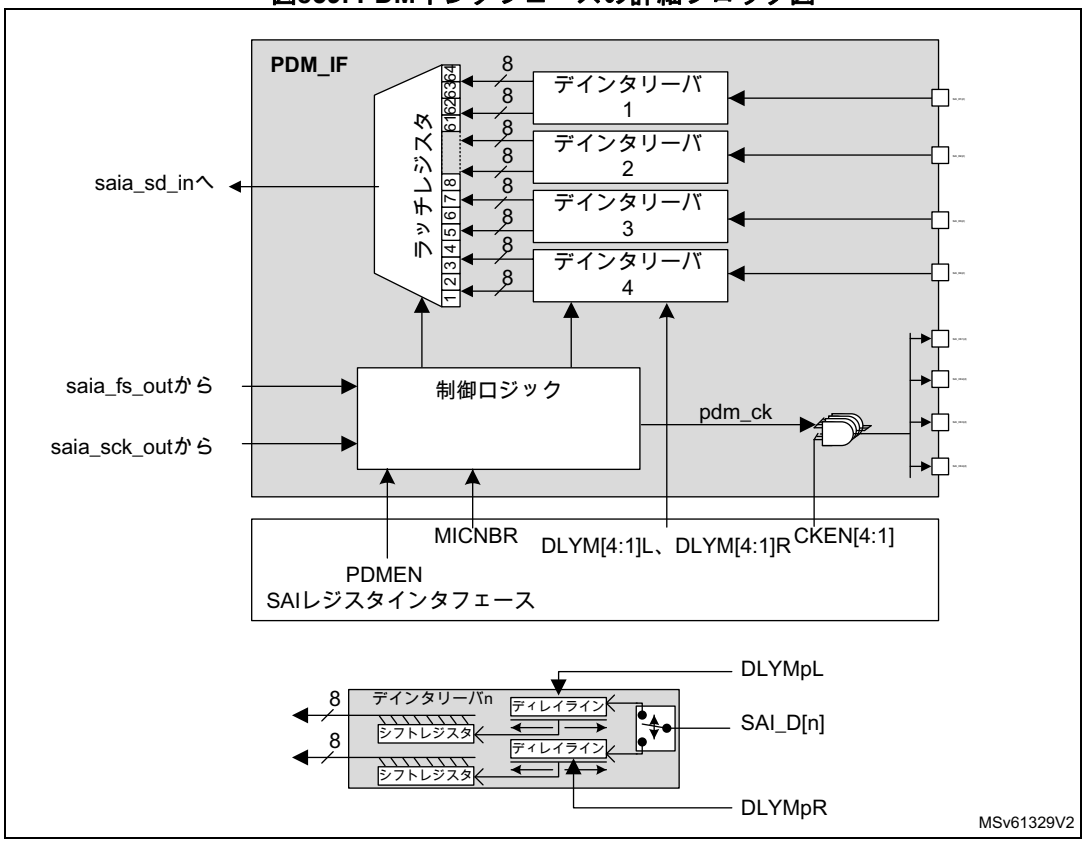
PDMへのデータ処理シーケンスは、次のとおりです。

1. PDMインタフェースは、SAI_AのTDMインタフェースから受信したビットクロックからビットストリームクロックを構築します。
2. マイクロフォンから受信したビットストリームデータ（SAI_D[n]）はインタリーブを解かれ、ビットストリームクロック精度で各マイクロフォンの遅延を微調整するために、7bit遅延ラインを通ります。
3. シフトレジスタによって、それぞれのシリアルビットストリームがバイト列に変換されます。
4. 最後の操作では、TDM インタフェースのシリアルデータラインを通じて、結果として得られたバイト列がSAI_Aにシフトアウトされます。

以下の図859に、PDMインタフェースのブロック図をデインタリーバの詳細図とともに示します。

注： **PDMインタフェースには、ビットストリームからPCMオーディオサンプルを構築するために必要なデシメーションフィルタは組み込まれていません。この操作を行うかどうかは、アプリケーションソフトウェア次第です。**

図859. PDMインタフェースの詳細ブロック図



1. **n**はデータライン数、**p**はマイクロフォンのペア数を表します。
2. これらの信号は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3 : SAIの実装を参照してください。

PDMインタフェースは、SAI_PDMCRレジスタのPDMENビットを通じて有効にできます。ただし、SAI_Aブロックを有効にする前にPDMインタフェースを有効にする必要があります。

メモリのフットプリントを削減するため、アプリケーションが必要とするマイクロフォンの本数を選択できます。この選択は、MICNBR[1:0] ビットで行うことができます。マイクロフォンの本数は、2本、4本、6本、8本から選択可能です。たとえば、アプリケーションがマイクロホンを3本使用するのであれば、4本を選択する必要があります。

PDMインタフェースの有効化

PDMインタフェースを有効にするには、以下のシーケンスに従います。

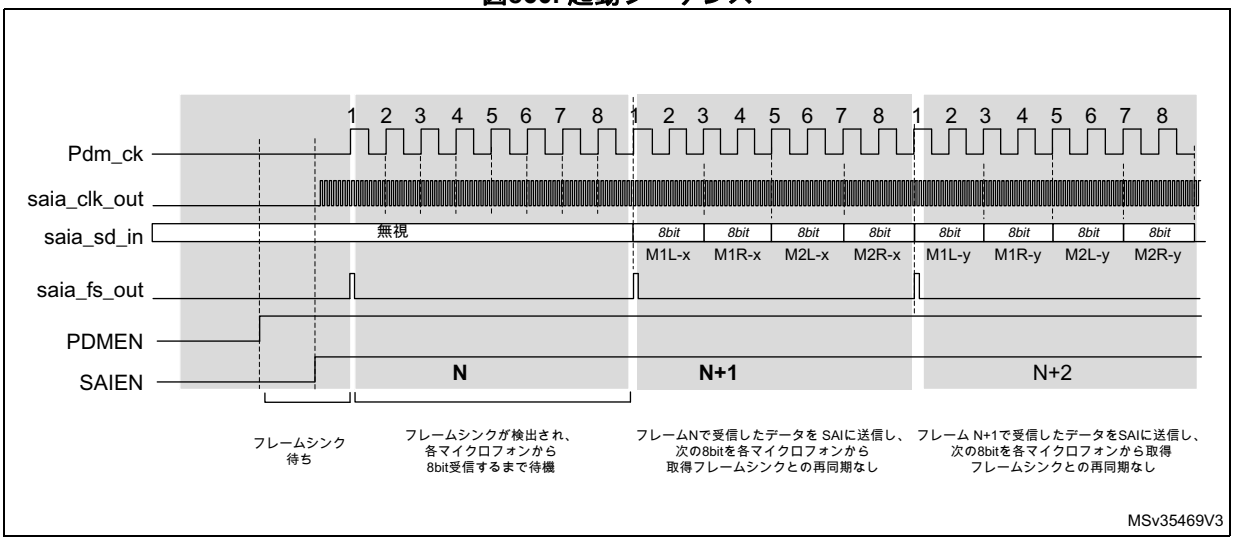
1. SAI_AをTDMマスタモードに設定します（表704参照）。
2. 次のようにPDMインタフェースを設定します。
   a) MICNBRにデジタルマイクロフォンの本数を設定します。
   b) CKENの対応するビットを1にセットして、アプリケーションで必要なビットストリームクロックを有効にします。
3. PDMENビットでPDMインタフェースを有効にします。
4. SAI_Aを有効にします。

注： **PDMインタフェースとSAI_Aが有効になると、SAI_ADRで受信する最初の2つのTDMAフレームは無効ですので、破棄する必要があります。**

起動シーケンス

図860に起動シーケンスを示します。PDMインタフェースが有効になると、マイクロフォンサンプルの取得を始める前に、フレーム同期イベントを待ちます。SAI_CKクロック8周期後、各マイクロフォンから入力されるデータバイトが利用可能となり、TDMインタフェースを通じてSAIに転送されます。

図860. 起動シーケンス



SAI_ADRデータフォーマット

マイクロフォンからSAI_ADRレジスタに入力されるデータの配列は、次のパラメータに依存しています。

- マイクロフォンの本数
- 選択されたスロット幅
- LSBFIRSTビット

スロット幅によって、SAI_ADRに入力可能な各ワードの有効ビット数が設定されます。

32bitのスロット幅が選択された場合には、SAI_ADRに入力可能な各データには32bitの有効なビットが含まれます。これによって、メモリに格納されるワード数が減ります。ただし、対応するには、ソフトウェアが各マイクロフォンのデータのインタリーブを解くための何らかの操作を行う必要があります。

その一方で、8bitのスロット幅が選択された場合には、SAI_ADRに入力可能な各データには8bitの有効ビットが含まれます。これによって、メモリに格納されるワード数が増えます。ただし、各ワードにはマイクロフォン1本からの情報が含まれているため、これには余計な処理を避けるという利点があります。

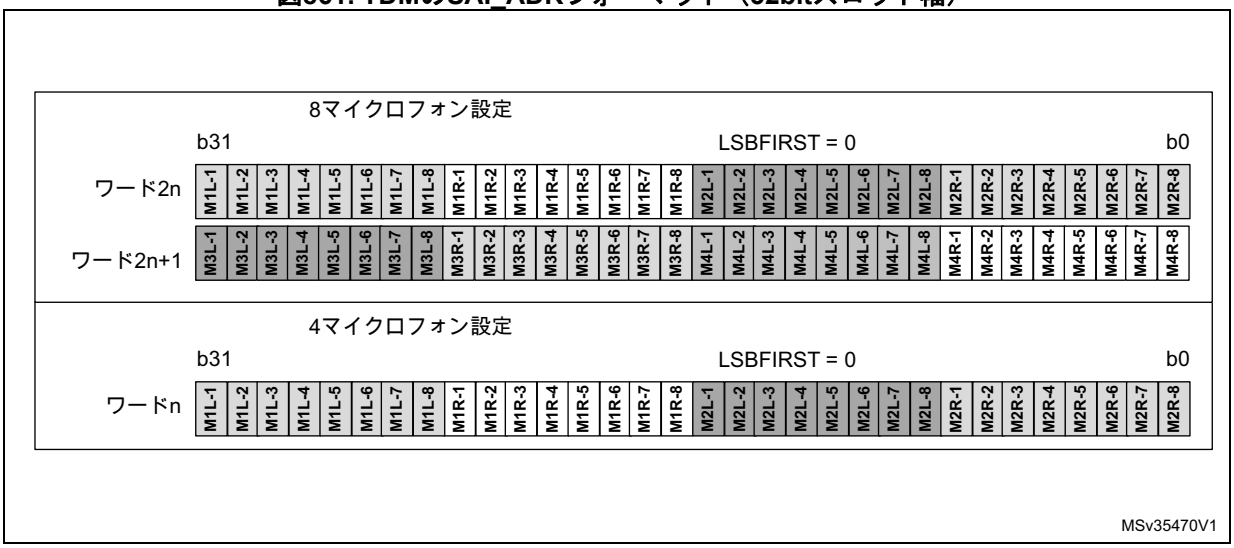
SAI_ADRデータフォーマット例

- **32bitスロット幅**（DS = 0b111かつSLOTSZ = 0）図861を参照してください。

  マイクロフォン8本の設定では、SAI_ADRレジスタから読み込まれた連続する2ワードに、各マイクロフォンからの1データバイトが含まれています。

  マイクロフォン4本の設定では、SAI_ADRレジスタから読み込まれた各ワードに、各マイクロフォンからの1データバイトが含まれています。

図861. TDMのSAI_ADRフォーマット（32bitスロット幅）

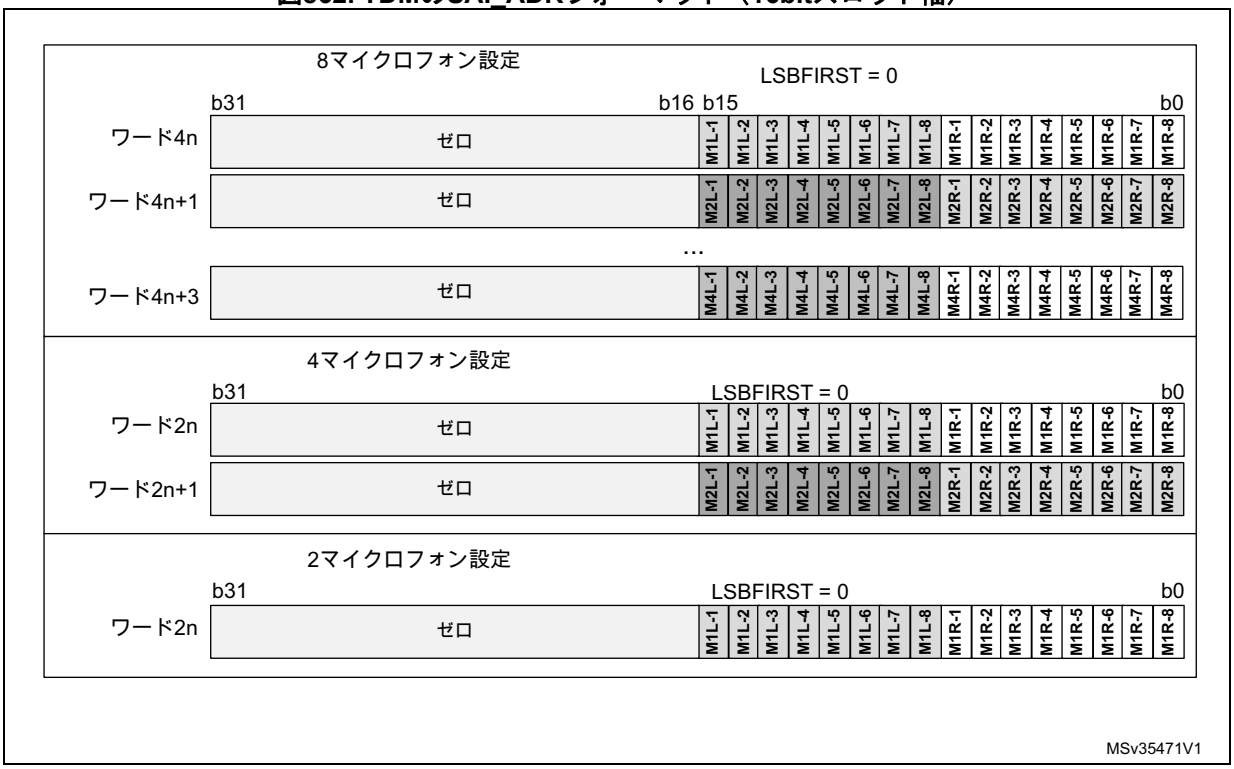


- **16bitスロット幅**（DS = 0b100かつSLOTSZ = 0）図862を参照してください。

  マイクロフォン8本の設定では、SAI_ADRレジスタから読み込まれた連続する4ワードに、各マイクロフォンからの1データバイトが含まれています。SAI_ADRの16bitデータは右詰めされていることに注意してください。

  マイクロフォン4本または2本の設定では、SAIの動作はマイクロフォン8本の設定とほぼ同じです。マイクロフォン4本からバイトを取得するには16bitワードが最大で2ワード必要であり、マイクロフォン2本からであれば1ワード必要です。

図862. TDMのSAI_ADRフォーマット（16bitスロット幅）

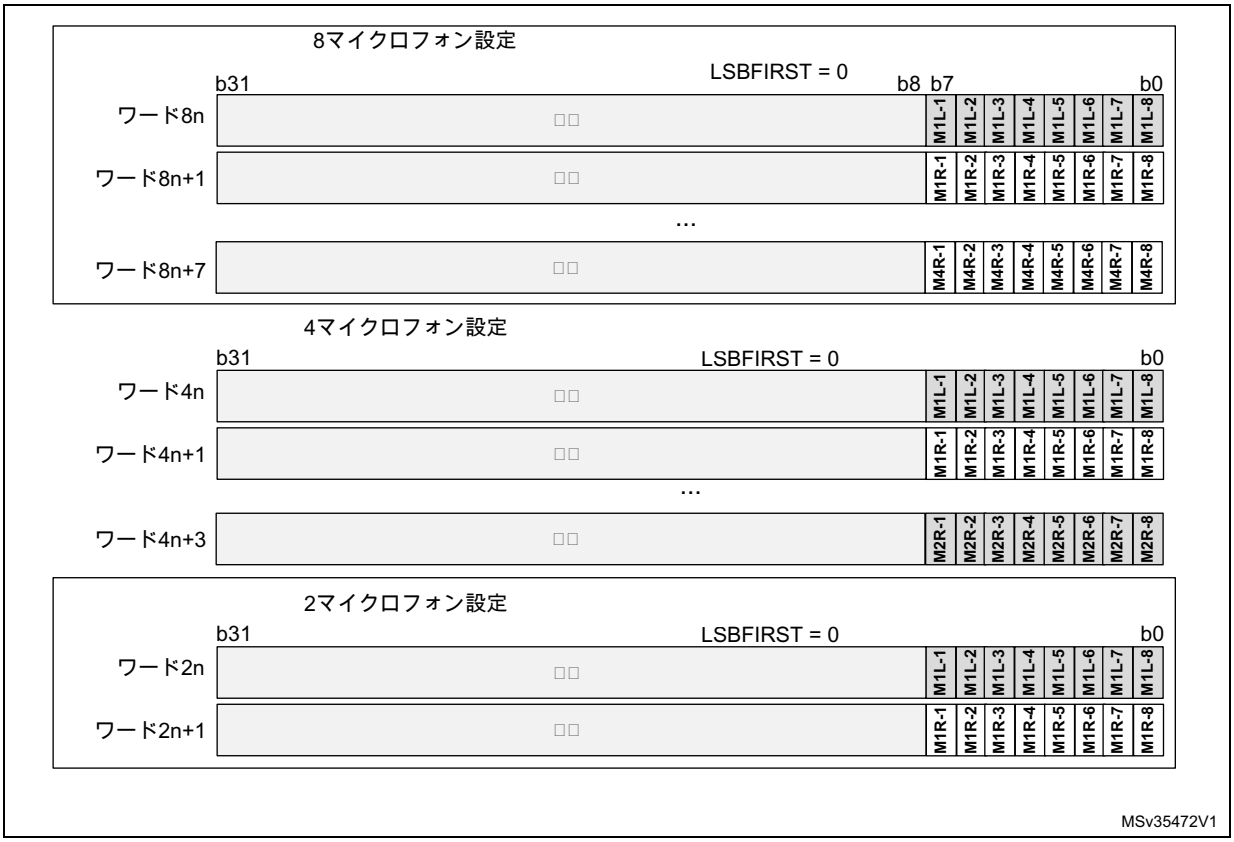


- **8bitスロット幅を使用**（DS = 0b010かつSLOTSZ = 0）図863を参照してください。

  マイクロフォン8本の設定では、SAI_ADRレジスタから読み込まれた連続する8ワードに、各マイクロフォンからのデータが1バイト含まれています。SAI_ADRの8bitデータは右詰めされていることに注意してください。

  マイクロフォン4本または2本の設定では、SAIの動作はマイクロフォン8本の設定とほぼ同じです。マイクロフォン4本からバイトを取得するには8bitワードが最大で4ワード必要であり、マイクロフォン2本からであれば2ワード必要です。

図863. TDMのSAI_ADRフォーマット（8bitスロット幅）



PDMインタフェースのTDM設定

SAI_A TDMインタフェースは、マイクロフォンのサンプルを取得するために、内部でPDMインタフェースに接続されています。ユーザアプリケーションは、PDMインタフェースを表704に示すように設定して、PDMインタフェースとの良好な接続を保証する必要があります。

表704.TDM設定

ビット フィールド	値	コメント
MODE	0b01	モードはMASTERレシーバである必要があります。
PRTCFG	0b00	TDM用フリープロトコル
DS	X	フレーム長とスロット数（FRLとNBSLOT）に基づいて、必要なデータフォーマットに応じて調整します。表705を参照してください。
LSBFIRST	X	このパラメータは必要なデータフォーマットに応じて使用します。
CKSTR	0	信号の遷移は、SCK_Aビットクロックの立ち上りエッジで発生します。信号はビットクロックの立下りでは安定しています。
MONO	0	ステレオモード
FRL	X	マイクロフォンの本数（MICNBR）に応じて調整します。表705を参照してください。
FSALL	0	ハルス幅は1bitクロックサイクルです。
FSDEF	0	FS信号がフレームの開始です。

表704.TDM設定 （続き）

ビット フィールド	値	コメント
FSPOL	1	FSはアクティブハイです。
FSOFF	0	FSはスロット0の先頭ビットでアサートされます。
FBOFF	0	スロットにはオフセットはありません。
SLOTSZ	0	スロットサイズ = データサイズ
NBSLOT	X	スロットサイズとフレーム長（FRLとDS）に基づいて、必要なデータフォーマットに応じて調整します。表705を参照してください。
SLOTEN	X	NBSLOTに応じて設定します。
NODIV	1	マスタクロックMCLKの生成は不要です。
MCKDIV	X	sai_a_ker_ck入力に供給される周波数に依存します。 このパラメータは、適切なビットストリームクロック周波数を生成するために調整する必要があります。表705を参照してください。

ビットストリームクロックレートの調整

SAI TDMインタフェースを適切にプログラムするには、ユーザアプリケーションは表704に記載された設定を考慮し、次の手順に従う必要があります。

1. 次の式を使用して、PDM ビットストリームクロックに必要な周波数に応じてビットクロック周波数（F_{SCK_A}）を調整します。

$$F_{SCK_A} = F_{PDM_CK} \times (MICNBR + 1) \times 2$$

MICNBRは、0、1、2、または3の値を取ります（0 = マイクロフォン2本、セクション69.6.18参照）。

2. 次の式を使用して、フレーム長（FRL）を設定します。

$$FRL = (16 \times (MICNBR + 1)) - 1$$

3. スロットサイズ（DS）を（FRL+1）の倍数に設定します。

表705.TDMフレーム設定の例(1)(2)

マイクロフォンサンプリングレート	マイクロフォン本数	必要なSAI_CKn周波数 (3)	ビットクロック（SCK_A）	フレームシンク（FS_A）	FRL	DS	NBSLOT	コメント
48kHz	最大8	3.072MHz	24.576MHz	384kHz	63	0b111	1	フレームあたり32bit 2スロット
		3.072MHz	24.576MHz	384kHz	63	0b100	3	フレームあたり16bit 4スロット
		3.072MHz	24.576MHz	384kHz	63	0b010	7	フレームあたり8bit 8スロット
	最大6	3.072MHz	18.432MHz	384kHz	47	0b110	1	フレームあたり24bit 2スロット
		3.072MHz	18.432MHz	384kHz	47	0b100	2	フレームあたり16bit 3スロット
		3.072MHz	18.432MHz	384kHz	47	0b010	5	フレームあたり8bit 6スロット
	最大4	3.072MHz	12.288MHz	384kHz	31	0b111	0	フレームあたり32bit 1スロット
		3.072MHz	12.288MHz	384kHz	31	0b100	1	フレームあたり16bit 2スロット
		3.072MHz	12.288MHz	384kHz	31	0b010	3	フレームあたり8bit 4スロット
	最大2	3.072MHz	6.144MHz	384kHz	15	0b100	0	フレームあたり16bit 1スロット
		3.072MHz	6.144MHz	384kHz	15	0b010	1	フレームあたり8bit 2スロット
16kHz	最大8	1.024MHz	8.192MHz	128kHz	63	0b111	1	フレームあたり32bit 2スロット
		1.024MHz	8.192MHz	128kHz	63	0b100	3	フレームあたり16bit 4スロット
		1.024MHz	8.192MHz	128kHz	63	0b010	7	フレームあたり8bit 8スロット
	最大6	1.024MHz	6.144MHz	128kHz	47	0b110	1	フレームあたり24bit 2スロット
		1.024MHz	6.144MHz	128kHz	47	0b010	5	フレームあたり8bit 6スロット
	最大4	1.024MHz	4.096MHz	128kHz	31	0b111	0	フレームあたり32bit 1スロット
		1.024MHz	4.096MHz	128kHz	31	0b100	1	フレームあたり16bit 2スロット
		1.024MHz	4.096MHz	128kHz	31	0b010	3	フレームあたり8bit 4スロット
	最大2	1.024MHz	2.048MHz	128kHz	15	0b100	0	フレームあたり16bit 1スロット
		1.024MHz	2.048MHz	128kHz	15	0b010	1	フレームあたり8bit 2スロット

1. TDM設定の詳細については、表 704：TDM設定を参照してください。SAIに供給されるsai_a_ker_ckクロック周波数は、SCK_A周波数の倍数である必要があり、MCKDIVはそれに応じてプログラムされる必要があります。
2. 上記のsai_a_ker_ck周波数は、あくまで一例です。Reset and clock controller (RCC) to check if they can be generated on the deviceのセクションを参照してください。
3. 上表は、デシメーション比が64である場合に許容される設定です。

ディレイラインの調整

PDMインタフェースが有効になると、アプリケーションは、SAI_PDMDLYレジスタを通じて各マイクロフォン入力のディレイセルを実行中に調整できます。

新たなディレイ値は、TDMフレームの2周期後に有効になります。

69.4.11 AC’97リンクコントローラ

SAIはAC’97リンクコントローラとしても機能することができます。このプロトコルの仕様は次のとおりです：

- スロット番号とスロットサイズは固定です。
- フレーム同期信号は完全に定義されており波形は固定です。

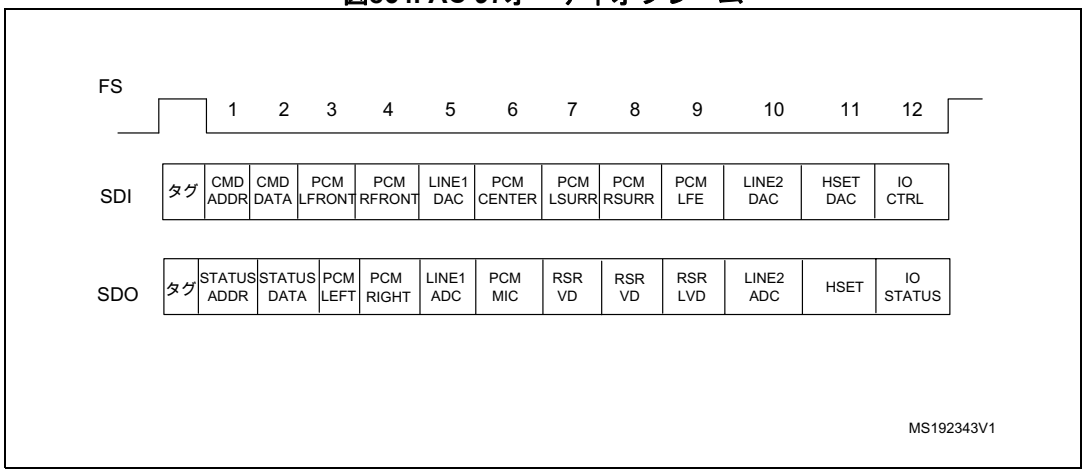
このプロトコルを選択するには、SAI_xCR1レジスタのPRTCFG[1:0] ビットを10に設定します。AC’97モードを選択した場合、使用可能なデータサイズは16bitまたは20bitのみであり、その他の場合、SAIの動作は保証されません。

- したがって、NBSLOT[3:0] ビットとSLOTSZ[1:0] ビットは無視されます。
- スロット数は13スロットに固定されています。最初のスロットは16bit幅で、他はすべて20bit幅（データスロット）です。
- SAI_xSLOTRレジスタのFBOFF[4:0] ビットは無視されます。
- SAI_xFRCRレジスタは無視されます。
- MCLKは使用されません。

非同期指定されたブロックからのFS信号は、自動的に出力として設定されます。これは、マスタ/スレーブの設定にかかわらずAC’97コントローラリンクがFS信号を駆動するためです。

図864に、AC’97オーディオフレーム構造を示します。

図864. AC’97オーディオフレーム



注： **AC’97プロトコルでは、タグのビット2は予約済みです（常に0）。したがって、SAI FIFOに書き込まれる値によらず、タグのビット2は0レベルに固定されます。**

タグ 表示についての詳細は、AC’97プロトコル 規格を参照してください。

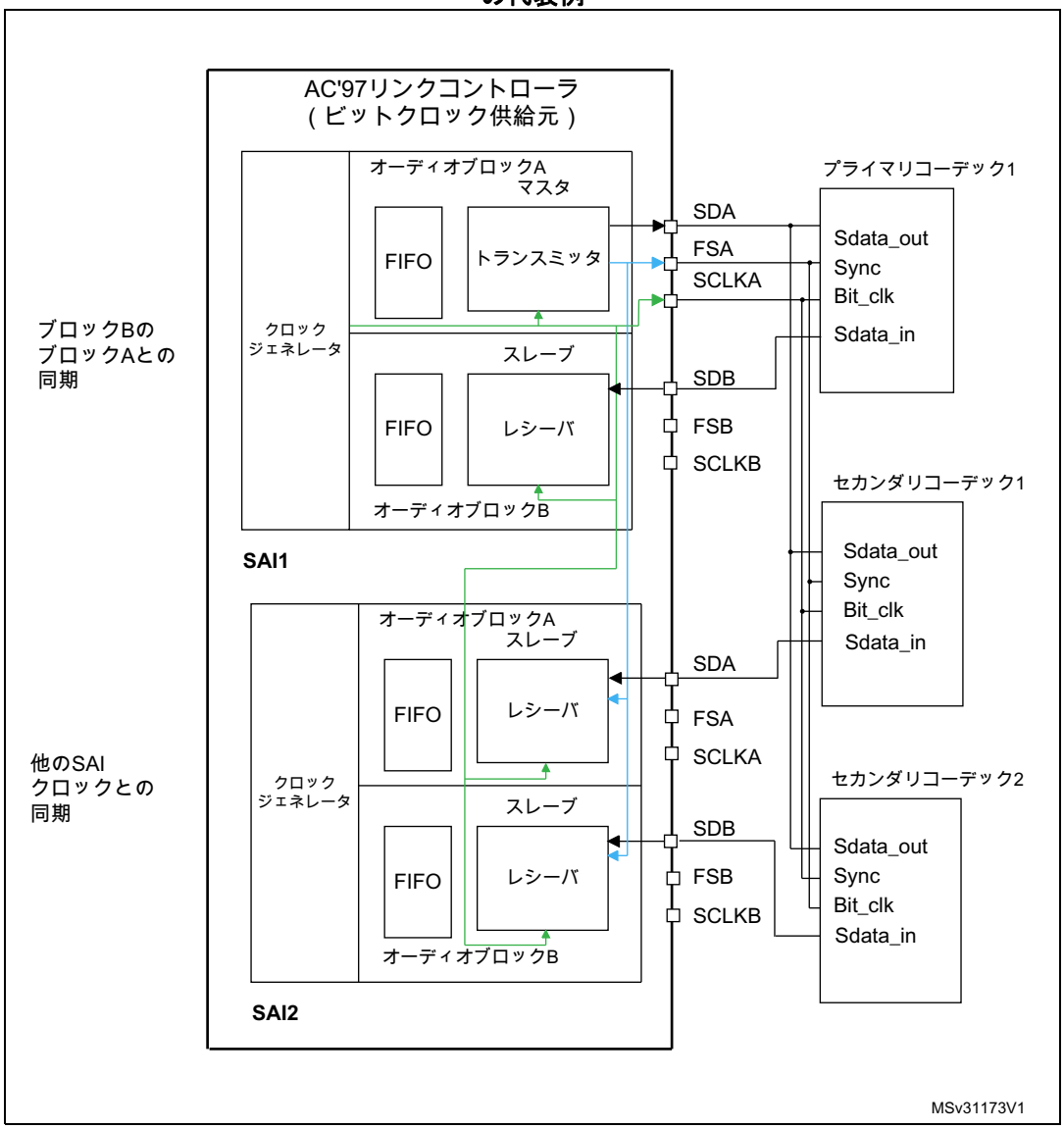
1つのSAIをAC’97の2点間通信に的を絞って使用することができます。

2つのSAIを使用することで（デバイスに2つの内蔵SAIを搭載の場合）、図865に示す3個の外部AC’97デコーダが有効化されます。

SAI1では、オーディオブロックAを非同期マスタトランスミッタに指定する必要がありますが、オーディオブロックBはスレーブレシーバに指定され、オーディオブロックAと内部的に同期します。

SAI2では、オーディオブロックAとBがどちらもスレーブ受信モードで外部SAI1と同期するように設定されます。

図865. 2つ以上の内蔵SAIを搭載したデバイス（3個の外部AC'97デコーダ）のAC'97設定の代表例



受信モードでは、スロット0のコーデックレディビットがローにデコードされた場合、AC'97リンクコントローラとして機能するSAIは、FIFOリクエストを必要としないためFIFO内のデータストレージも不要です。SAI_xIMレジスタのCNRDYIEビットが有効化された場合、SAI_xSRレジスタのCNRDYフラグがセットされ、割込みが生成されます。このフラグは、AC'97プロトコル専用です。

AC'97モードでのクロックジェネレータのプログラミング

AC'97モードでは、フレーム長は256bitに固定され、その周波数は48kHzに設定する必要があります。セクション 69.4.8 : SAIクロックジェネレータに示す式には、適切なフレームレート（F_{FS_x}）を生成するために、FRL = 255という条件を適用する必要があります。

69.4.12 SPDIF出力

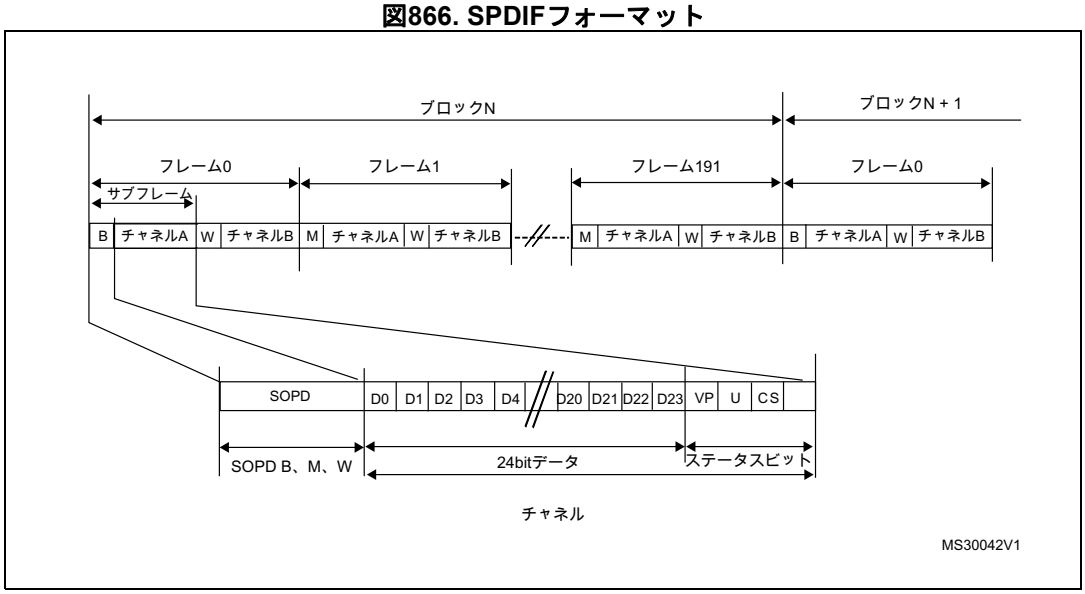
SPDIFインタフェースは送信モードでのみ使用できます。このインタフェースはオーディオIEC60958をサポートします。

SPDIFモードを選択するには、SAI_xCR1レジスタのPRTCFG[1:0] ビットに01をセットします。

SPDIFプロトコルを使用する場合

- SD データラインのみが有効です。
- I/OピンのFS、SCK、MCLKは空けてあります。
- MODE[1] ビットは、SAIのクロックジェネレータを有効にしたりSDラインでのデータレートを管理するために強制的に0にして、マスタモードを選択します。
- データサイズは強制的に24bitに設定されます。SAI_xCR1レジスタのDS[2:0] ビットにセットされた値は無視されます。
- クロックジェネレータのシンボルレートを、ビットクロックがシンボルレートの2倍でなければならないことを考慮して、定義する必要があります。データはマンチェスタープロトコルでコード化されます。
- SAI_xFRCRレジスタおよびSAI_xSLOTRレジスタは無視されます。SAIは、SPDIFプロトコル要件（図866を参照）に適合するように内部的に設定されます。

図866. SPDIFフォーマット



SPDIFブロックには192個のフレームが含まれています。各フレームは、2個の32bitサブフレーム（一般に左チャネル用と右チャネル用に1つずつ）で構成されています。各サブフレームはSOPDパターン（4bit）で構成されています。このパターンを使用して、サブフレームがブロックの先頭である（つまり、チャネルAを識別している）、ブロックのどこかでチャネルAを識別している、またはチャネルBを参照しているのいずれかを指定します（表706を参照）。チャネル情報の次の28bitは24bitのデータと4bitのステータス情報から成ります。

表706.SOPDパターン

SOPD	プリアンブルのコード化		説明
	最終ビットが0	最終ビットが1	
B	11101000	00010111	ブロックの始めにあるチャネルAデータ
W	11100100	00011011	ブロックのどこかにあるチャネルBデータ
M	11100010	00011101	チャネルAデータ

SAI_xDR に格納されているデータは次のような内容でなければなりません。

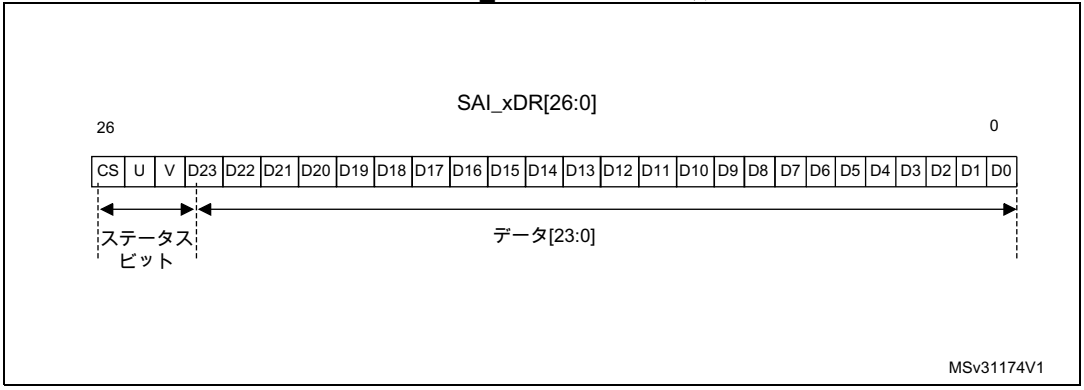
- SAI_xDR[26:24] は、チャネルのステータスビット、ユーザビット、有効性ビットを含みます。
- SAI_xDR[23:0] は、検討されたチャネルに関するデータ24bitを含みます。

データサイズが20bitの場合、データはSAI_xDR[23:4] に配置する必要があります。

データサイズが16bitの場合、データはSAI_xDR[23:8] に配置する必要があります。

SAI_xDR[23] は常にMSBを表します。

図867. SAI_xDRレジスタの順序



注： **必ずLSB（最下位ビット）から転送されます。**

SAIはまず最初にブロック内の各サブフレームの適切なプリアンブル部を送信します。次に、SAI_xDRがSDラインに送信されます（マンチェスターコード化）。SAIは、表707の記載通りに計算されたパリティビットを転送することで、サブフレームを終了させます。

表707.パリティビットの計算

SAI_xDR[26:0]	転送されるパリティビットPの値
奇数個の”0 ”	0
奇数個の”1 ”	1

SAIは送信モードでしか動作できないため、アンダーランはSPDIFモードのSAI_xSRレジスタで有効な唯一のエラーフラグです。そのため、アンダーラン割込みまたはアンダーランステータスビットによって検出されたアンダーランエラーから復帰するためには、次のシーケンスを実行する必要があります。

1. DMAが使用されている場合は、DMAストリームを無効にします（DMAペリフェラルを使用）。
2. SAIを無効にし、SAI_xCR1レジスタのSAIENビットをポーリングすることでペリフェラルが物理的に無効化されていることを確認します。

3. SAI_xCLRFRレジスタのCOVRUNDRフラグをクリアします。
4. FIFOを一掃するために、SAI_xCR2のFFLUSHビットをセットします。

   ソフトウェアで、新しいブロックの始めに今後格納するデータ（プリアンブル部B用のデータ）のアドレスを指し示す必要があります。DMAが使用されている場合、DMAソースベースアドレスのポインタは適宜更新する必要があります。
5. DMAを使用して、新しいソースベースアドレスに従ってデータ転送を管理する場合は、DMAストリーム（DMAペリフェラル）を再び有効にします。
6. SAI_xCR1レジスタのSAIENビットをセットして、再びSAIを有効にします。

SPDIFジェネレータモードでのクロックジェネレータのプログラミング

SPDIFジェネレータモードの場合、SAIはシンボルレートの2倍の速度のビットクロックを供給します。下表では、オーディオサンプリングレートに対するシンボルレートの一般例を示します。

表708.オーディオサンプリング周波数とシンボルレート

オーディオサンプリング周波数（F_S）	シンボルレート
44.1kHz	2.8224MHz
48kHz	3.072MHz
96kHz	6.144MHz
192kHz	12.288MHz

より一般的には、オーディオサンプリング周波数（F_S）とビットクロックレート（F_{SCK_X}）の関係は、次の式で求められます。

$$F_S = \frac{F_{SCK_x}}{128}$$

ビットクロックレートは次のように求められます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

注： **上式は、SAI_ACR1レジスタで NODIV が1にセットされている場合にのみ有効です。**

69.4.13 特有の機能

SAIインタフェースには、選択したオーディオプロトコルに応じて役立つ特定の機能が組み込まれています。これらの機能へは、SAI_xCR2レジスタの特定ビットを使ってアクセスすることができます。

ミュートモード

ミュートモードは、オーディオサブブロックがトランスミッタまたはレシーバであるとき使用することができます。

送信モード時のオーディオサブブロック

送信モード時、ミュートモードはいつでも選択可能です。ミュートモードは、オーディオフレーム全体に対して有効です。処理中のフレーム内でSAI_xCR2レジスタのMUTEビットがセットされると、ミュートモードが有効になります。

ミュートモードビットは、フレームの終わりでのみ認識されます。この時点でミュートモードビットがセットされた場合は、ミュートモードは新しいオーディオフレームの先頭から次に来るフレームの

終了までフレーム全体に対して有効になります。その後、このビットを認識して、次のフレームもミュートフレームか否かを決定します。

SAI_xSLOTRレジスタのNBSLOT[3:0] ビットに設定されたスロット数が2以下の場合、ミュートモード中に送信される値が0なのか、または各スロットの最後の値なのかを指定することができます。この選択は、SAI_xCR2レジスタのMUTEVALビットを使って行います。

SAI_xSLOTRレジスタのNBSLOT[3:0] ビットに設定されたスロット数が2より大きい場合、各スロットの各ビットで0値が送信されるため、SAI_xCR2のMUTEVALビットは無意味となります。

ミュートモード時は、FIFOポインタのインクリメントは続きます。つまり、FIFOにデータは供給され続けますが、ミュートモードの間は破棄されます。

受信モード時のオーディオサブブロック

受信モードでは、指定されかつ有効なオーディオフレームのすべてのスロットが、指定された連続した数のオーディオフレームに対して0を受信したとき、外部 トランスミッタから送信されたミュートモードを検出することが可能です（ SAI_xCR2レジスタのMUTECNT[5:0] ビット）。

ミュートフレーム数が検出されると、SAI_xSR レジスタのMUTEDET フラグがセットされて、SAI_xCR2のMUTEDETIEビットがセットされている場合は割込みを生成することができます。

オーディオサブブロックが無効化されたとき、または有効なスロットがオーディオフレーム内で1つ以上のデータを受信したとき、ミュートフレームカウンタがクリアされます。カウンタがMUTECNT[5:0] ビットに指定された値に到達すると、割込みが1回だけ生成されます。次に、カウンタがクリアされると割込みイベントが再初期化されます。

注： **ミュートモードはSPDIFオーディオブロックには使用できません。**

モノラル／ステレオモード

送信モードでは、スロット数が2と想定し（SAI_xSLOTRでNBSLOT[3:0] = 0001）、メモリ内でデータを前処理することなくモノラルモードを指定できます。このような場合、スロット0のデータがデータスロット1へ複製されるため、FIFOへのアクセスタイムは半減されます。

モノラルモードを有効にするには、

1. SAI_xCR1レジスタのMONOビットに1をセットします。
2. SAI_xSLOTRレジスタのNBSLOTビットに1、SLOTENビットに3をセットします。

受信モードでは、MONOビットをセットすることができますが、送信モードの場合と同様にスロット数が2のときだけ意味があります。このビットをセットした場合、スロット0のデータだけがFIFOに格納されます。この場合、スロット1に属するデータは、前のスロットと同じと見なされるため、破棄されます。受信時のデータの流れが左右でデータが異なる真のステレオオーディオフローである場合、MONOビットは意味を成しません。出力ステレオファイルから等価なモノラルファイルへの変換は、ソフトウェアで行われます。

コンパンディングモード

通信アプリケーションでは、データコンパンディングアルゴリズムにより送信データまたは受信データの処理が必要となることがあります。

図868に示すように、SAI_xCR2レジスタのCOMP[1:0] ビットに応じて（フリープロトコルモードを選択した場合のみ使用）、SDシリアル出力ラインに送信する前に処理（圧縮）するか否か、あるいはSDシリアル入力ラインから受信した後にデータを処理（伸長）するか否かをソフトウェアで選択することができます。サポートされている2つのコンパンディングモードは μ-LawとA-Law(l対数) で、これらはCCITT G.711勧告の一部となっています。

米国と日本で採用されているコンパンディング規格は μ-Lawであり、14bitのダイナミックレンジが確保できます（SAI_xCR2レジスタでCOMP[1:0] = 10）。

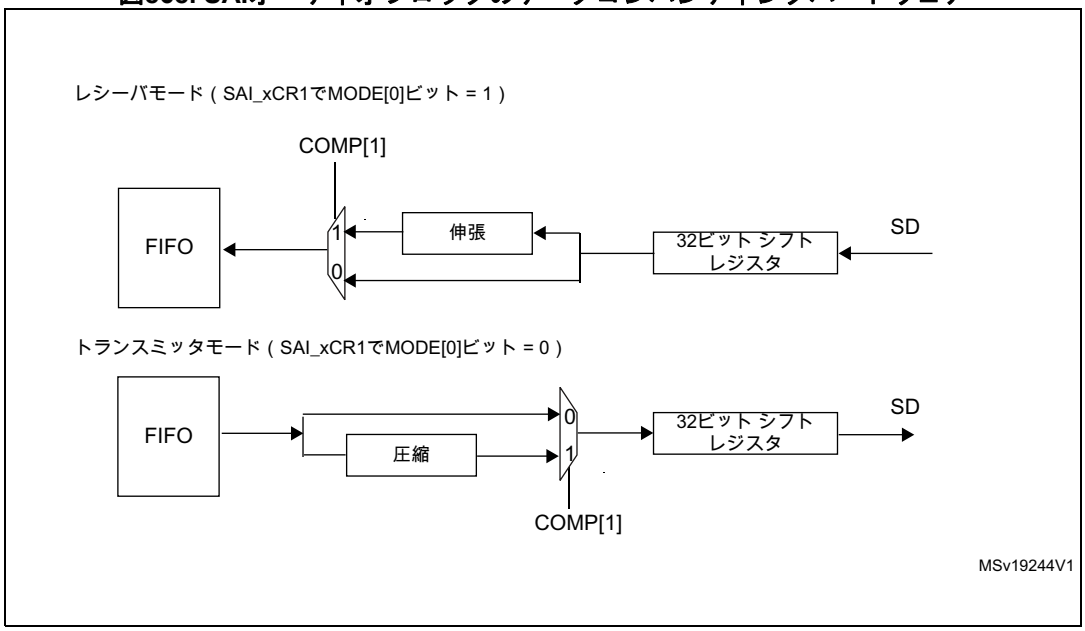
ヨーロッパのコンパンディング規格はA-Lawであり、13bitのダイナミックレンジが確保できます（SAI_xCR2レジスタでCOMP[1:0] = 11）。

コンパンディング規格（μ-LawまたはA-Law）は、SAI_xCR2レジスタのCPLビットの設定に応じて、1の補数表現または2の補数表現に基づいて計算することができます。

μ-LawおよびA-Law規格では、データをMSB詰めの8bitコードにコード化します。コンパンドされたデータは常に8bit幅です。このため、SAIオーディオブロックが有効化されていて（SAI_xCR1レジスタでSAIENビット = 1）、さらに、COMP[1:0] ビットによりこれら2つのコンパンディングモードのいずれかを選択している場合、SAI_xCR1レジスタのDS[2:0] ビットは強制的に010に固定されます。

コンパンディングが不要な場合は、COMP[1:0] ビットをクリアしたままにする必要があります。

図868. SAIオーディオブロックのデータコンパンディングハードウェア



1. AC'97 またはSPDIF を選択した場合は適用されません。

伸張モードまたは圧縮モードはSAI_xCR2により自動的に選択されます。

- SAIオーディオブロックがトランスミッタに設定され、かつSAI_xCR2レジスタのCOMP[1] ビットがセットされている場合、圧縮モードが適用されます。
- SAIオーディオブロックがレシーバに指定された場合、伸長アルゴリズムが適用されます。

非アクティブスロットでの出力データライン管理

送信モードでは、非アクティブスロットがデータラインに送信されたときに出力となっているSDラインの動作を選択することができます（TRISビットを使用）。

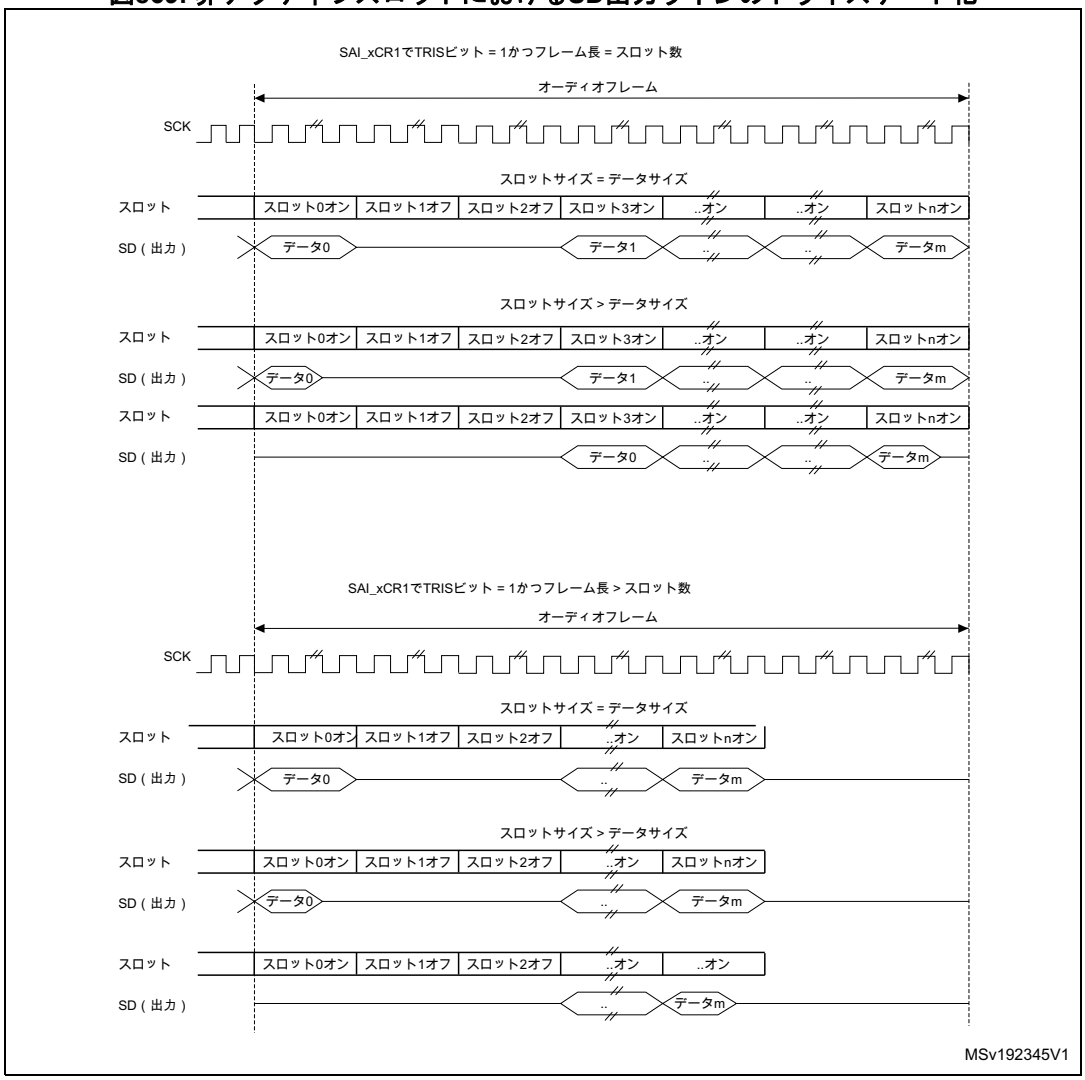
- 非アクティブスロットが送信されるとき、SAIがSD出力ラインを0に固定することもできれば、
- 送信データの最終ビットの終わりでラインをHI-Z状態にして、このノードに接続された他のトランスミッタへラインを開放することもできます。

2つのトランスミッタが同じSD出力ピンを同時に駆動しないように注意することが重要です。同時駆動した場合、短絡の原因となります。送信間の間隔を確保するため、データが32bitより短い場合は、SAI_xSLOTRレジスタのSLOTSZ[1:0] ビット = 10に設定することにより、データを32bitに拡張することができます。すると、次のスロットが非アクティブに指定された場合、アクティブスロットのLSBの終わりで（データを32bitに拡張するため0を詰めるフェーズで）SD出力ピンがトライステートになります。

さらに、スロット数とスロットサイズの積がフレーム長より小さい場合、0を詰めてオーディオフレームを完成させるときにSD出力ラインがトライステートになります。

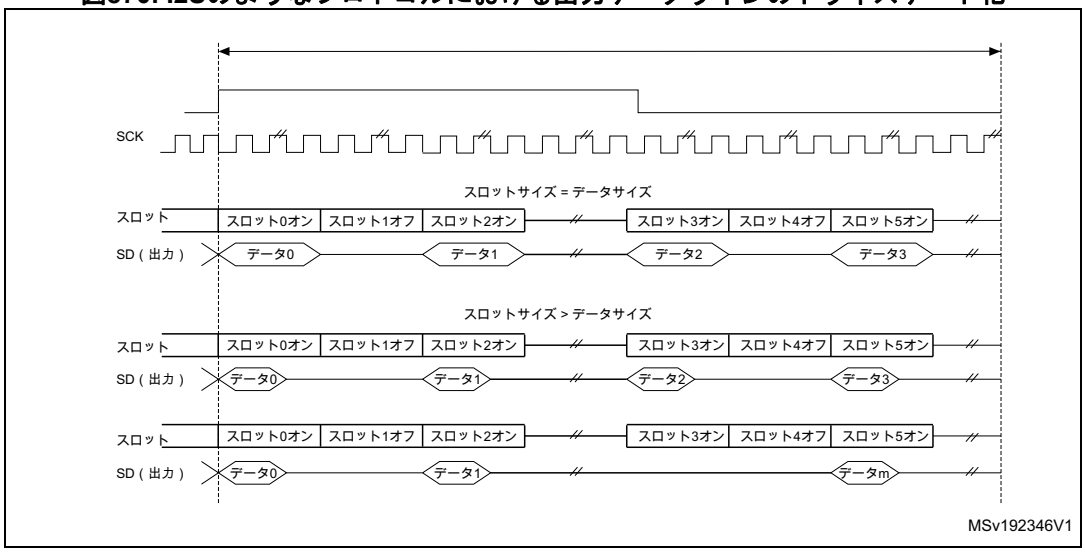
これらの動作を図869に示します。

図869. 非アクティブスロットにおけるSD出力ラインのトライステート化



選択したオーディオプロトコルがFS信号をフレーム開始とチャネルサイド識別として使用する場合（SAI_xFRCRレジスタでビットFSDEF = 1）、図870に従ってトライステートモードが管理されます（この図では、SAI_xCR1レジスタでビットTRIS = 1、FSDEF = 1、ハーフフレーム長 > スロット数 / 2、NBSLOT = 6）。

図870. I2SのようなプロトコルにおけるSD出力データラインのトライステート化



SAI_xCR2レジスタのTRISビットがクリアされている場合、図869と図870に示すSD出力ライン上のすべてのハイインピーダンス状態は0値での駆動として置き換えます。

69.4.14 エラーフラグ

SAIは次のエラーフラグを実装しています。

- FIFOオーバーラン／アンダーラン
- 予測フレーム同期検出
- レイトフレーム同期検出
- コーデックノットレディ（AC’97専用）
- マスタモードにおける不正クロック設定

FIFOオーバーラン/アンダーラン（OVRUDR）

FIFOオーバーラン/アンダーランビットは、SAI_xSRレジスタでOVRUDRと呼ばれます。

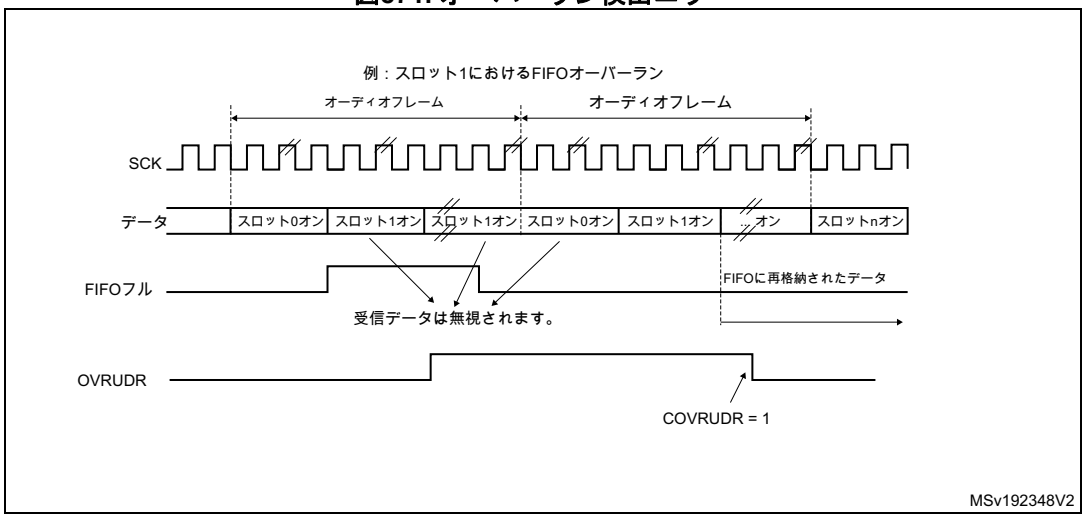
オーディオブロックはレシーバまたはトランスミッタのいずれかであり、SAI内の各オーディオブロックに固有のSAI_xSRレジスタを搭載しているため、オーバーランエラーとアンダーランエラーは、同じビットを使用しています。

オーバーラン

オーディオブロックがレシーバに設定されていて、FIFOがフルで受信データを保存できないときにオーディオフレームでデータが受信された場合に、オーバーラン状態が発生します。この場合、受信データが失われ、SAI_xSRレジスタのOVRUDRフラグがセットされ、SAI_xIMレジスタのOVRUDRIEビットがセットされている場合は割込みが生成されます。オーバーランが発生したスロット番号が、内部に保存されます。FIFOが新しいデータを格納できるようになるまで、データはFIFOに保存されません。FIFOに1データ分以上の空きがある場合、SAIオーディオブロックレシーバは、オーバーラン状態が検出されたとき内部に保存されたスロット番号からの新しいデータ（新しいオーディオフレームの）を保存します。これにより、転送先メモリでのデータスロットの配置の狂いを回避します（図871を参照）。

SAI_xCLRFRレジスタのCOVRUDRビットがセットされると、OVRUDRフラグがクリアされます。

図871. オーバーラン検出エラー



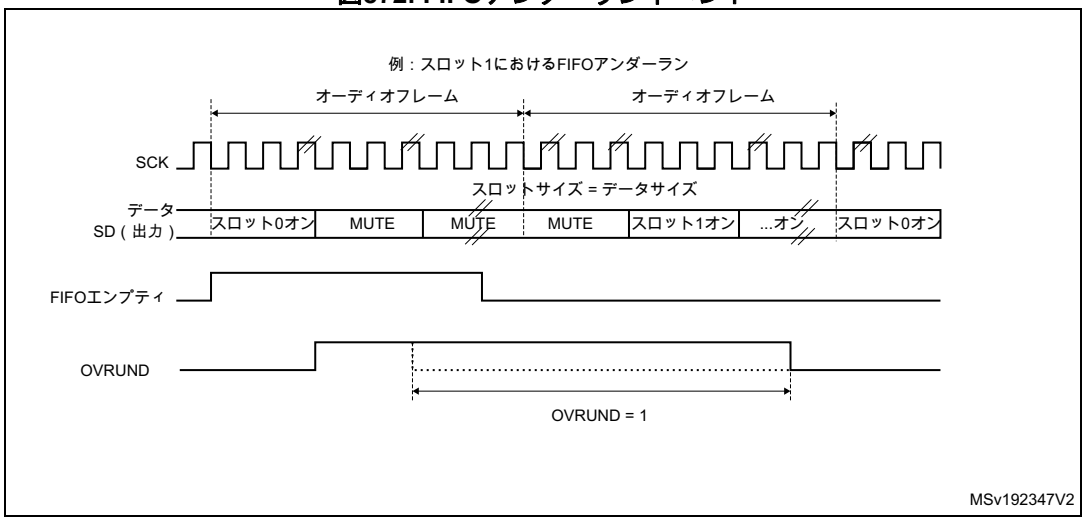
アンダーラン

SAI内のオーディオブロックがトランスミッタで、かつデータの送信が必要なときFIFOがエンプティ（空）の場合に、アンダーランが発生します。アンダーランが検出されると、イベントが発生したスロット番号が保存され、FIFOがアンダーランが検出されたスロットに対応するデータを送信する準備ができるまでMUTE値（00）が送信されます（図872を参照）。これにより、メモリポインタとオーディオフレーム内のスロットの同期はずれを回避します。

アンダーランイベントによりSAI_xSRレジスタのOVRUDRフラグがセットされ、SAI_xIMレジスタのOVRUDRIEビットがセットされている場合、割込みが生成されます。このフラグをクリアするには、SAI_xCLRFRレジスタのCOVRUDRビットをセットします。

アンダーランイベントは、オーディオサブブロックがマスタまたはスレーブとして設定されている場合に発生します。

図872. FIFOアンダーランイベント



予測フレーム同期検出（AFSDET）

AFSDETフラグはスレーブモードでのみ使用されます。マスタモードでは決してアサートされません。フレーム長、フレーム極性、フレームオフセットが設定されていて既知であるため、フレーム同期 （FS）が予測より早く検出されることを示します。

予測フレームの検出により、SAI_xSRレジスタのAFSDETフラグがセットされます。

この検出は、現在のオーディオフレームに影響を与えません。現在のオーディオフレームは予測FSに影響されません。つまり、現在のオーディオフレームを乱すことなく、FS信号に関する「寄生」イベントフラグが立つということです。

SAI_xIMレジスタのAFSDETIEビットがセットされている場合は、割込みが生成されます。AFSDETフラグをクリアするには、SAI_xCLRFRレジスタのCAFSDETビットをセットする必要があります。

予測フレーム検出エラーの後にマスタと再同期させるには、次の4ステップが必要です。

1. SAI_xCR1レジスタのSAIENビットをリセットして、SAIブロックを無効化します。SAIが無効であることを確認するには、SAIENビットを読み返し、それが0にセットされていることを確認します。
2. SAI_xCR2レジスタのFFLUSビットを使ってFIFOを一掃します。
3. SAIペリフェラルを再度有効化します（SAIENビットを1に設定）。
4. SAIブロックはFSの開始を待ってマスタとの同期を再開します。

注： **AFSDETフラグはAC’97モードではアサートされません。これはSAIオーディオブロックがリンクコントローラとして機能し、スレーブに指定された場合でもFS信号を生成するためです。SPDIFモードではFS信号は使用されないため、このフラグは意味を持ちません。**

レイトフレーム同期検出

SAIオーディオブロックがスレーブとして動作する場合のみ、SAI_xSRレジスタのLFSDETフラグをセットすることができます。フレーム長、フレーム極性、フレームオフセット設定は、SAI_xFRCRレジスタから知ることができます。

外部マスタがFS信号を予想タイミングで送信しない（信号の発生が遅すぎる）場合、LFSDETフラグがセットされ、SAI_xIMレジスタのLFSDETIEビットがセットされている場合は割込みが生成されます。

SAI_xCLRFRレジスタのCLFSDETビットがセットされると、LFSDETフラグがクリアされます。

レイトフレーム同期検出フラグは、対応するエラーが検出されるとセットされます。SAIはマスタと再同期させる必要があります（シーケンスは、予測フレーム同期検出（AFSDET）を参照）。

ノイズの多い環境では、オーディオブロックステートマシンがSCKクロック上のグリッチを間違って検出し、SAIデータが間違ったフレーム位置にシフトされることがあります。このようなイベントはSAIによって検出され、レイトフレーム同期検出エラーとして報告することができます。

外部マスタが連続モードでオーディオデータフレーム転送を管理していない場合、破壊は起こりませんが、ほとんどの場合、アプリケーションでこのような使い方はできません。この場合、LFSDETフラグがセットされます。

注： **LFSDETフラグはAC’97モードではアサートされません。これはSAIオーディオブロックがリンクコントローラとして機能し、スレーブと指定された場合でもFS信号を生成するためです。FS信号はプロトコルで使用されないため、このフラグはSPDIFモードでは 意味を持ちません。**

コーデックノットレディ（CNRDY AC’97）

SAIオーディオブロックがAC’97モードで動作するように設定されている場合にのみSAI_xSRレジスタのCNRDYフラグが関係します（SAI_xCR1レジスタのPRTCFG[1:0] ビット = 10）。SAI_xIMレジスタのCNRDYIEビットがセットされている場合、CNRDYフラグがセットされたときに割込みが生成されます。

AC’97オーディオフレームのタグ0（スロット0）の受信中にコーデックが通信の準備ができていないとき、CNRDYフラグがアサートされます。この場合、コーデックが動作可能であることをタグ0が示すまではコーデックは動作可能状態にはないため、FIFOに自動的に保存されるデータはありません。コーデックが動作可能状態になると、SAI_xSLOTRレジスタで指定されたすべてのアクティブスロットがキャプチャされます。

CNRDYフラグをクリアするには、SAI_xCLRFRレジスタのCCNRDYビットをセットする必要があります。

マスタモードでの不正なクロック設定（ NODIV = 0の場合）

オーディオブロックがマスタとして動作し（MODE[1] = 0）、NODIV ビットが0に等しい場合、次の条件が満たされてSAIが有効になるとすぐに、WCKCFGフラグがセットされます。

- (FRL + 1) は2の累乗値ではない、かつ
- (FRL + 1) は8～256の範囲外である。

MODE、NODIV、SAIENビットはSAI_xCR1レジスタに、FRLビットはSAI_xFRCRレジスタに 属します。

WCKCFGIEビットがセットされている場合、SAI_xSRレジスタでWCKCFGフラグがセットされると割込みが生成されます。このフラグをクリアするには、SAI_xCLRFRレジスタのCWCKCFGビットをセットします。

WCKCFGビットがセットされると、オーディオブロックが自動的に無効化され、ハードウェアによってSAIENビットがクリアされます。

69.4.15 SAIの無効化

SAI_xCR1レジスタのSAIENビットをクリアすることによって、SAIオーディオブロックをいつでも無効化することができます。すでに開始しているすべてのフレームは、SAIが動作を停止する前に自動的に完了します。現在のオーディオフレーム転送の最後にSAIが完全にオフするまで、SAIENビットはハイを保持します。

他のSAIと同期しているSAI内にオーディオブロックがある場合、マスタになっているSAIを先に無効化する必要があります。

69.4.16 SAI DMAインタフェース

CPUを開放し、バスの帯域幅を最適化するため、各SAIオーディオブロックにはSAI_xDRレジスタ（内部FIFOへのアクセス用）に対して読み書きを行うための独立したDMAインタフェースを搭載しています。オーディオサブブロック毎に1つのDMAチャネルがあり、基本的なDMAリクエスト／確認応答プロトコルに対応しています。

DMA転送を行うようにオーディオサブブロックを設定するには、SAI_xCR1レジスタのDMAENビットをセットします。DMAリクエストは、FIFO閾値レベルに応じてFIFOコントローラによって直接管理されます（詳細は、セクション 69.4.9 : 内部FIFOを参照してください）。DMA転送方向は、SAIオーディオサブブロック設定と次の様に関連付けされています。

- オーディオブロックがトランスミッタとして動作する場合、オーディオブロックのFIFOコントローラがDMAリクエストを出力してSAI_xDRレジスタに書き込まれたデータをFIFOにロードします。

- オーディオブロックがレシーバとして動作する場合、DMAリクエストはSAI_xDRレジスタからの読出し動作に関わります。

SAIインタフェースをDMAモードに設定するには、次のシーケンスに従ってください。

1. SAIとFIFO閾値レベルを設定して、DMAリクエストを開始するタイミングを指定します。
2. SAI DMAチャネルを設定します。
3. DMAを有効にします。
4. SAIインタフェースを有効にします。

注： **SAIブロックを設定する前に、SAI DMAチャネルを無効化する必要があります。**

69.5 SAI割込み

SAIは表709に示す7つの割込みソースに対応しています。

表709.SAI割込みソース

項目（割込みの略称）	割込みソース	割込みグループ	オーディオブロックモード	割込みイネーブル	割込みクリア
SAI	FREQ	FREQ	マスタ／スレーブ 受信または送信モード	SAI_xIMレジスタのFREQIE	下記に応じてクリアします。 – FIFO 閾値設定（SAI_xCR2 の FLVL ビット） – 通信方向（トランスミッタ／レシーバ） 詳細については、セクション 69.4.9 : 内部FIFOを参照してください。
	OVRUDR	ERROR	マスタ／スレーブ 受信または送信モード	SAI_xIMレジスタのOVRUDRIE	SAI_xCLRFRレジスタのCOVRUDR = 1
	AFSDET	ERROR	スレーブ （AC'97　モードおよびSPDIFモードでは使用しません。）	SAI_xIMレジスタのAFSDETIE	SAI_xCLRFRレジスタでCAFSDET = 1
	LFSDET	ERROR	スレーブ （AC'97　モードおよびSPDIFモードでは使用しません。）	SAI_xIMレジスタのLFSDETIE	SAI_xCLRFRレジスタでCLFSDET = 1
	CNRDY	ERROR	スレーブ （AC'97 モードの場合のみ）	SAI_xIMレジスタのCNRDYIE	SAI_xCLRFRレジスタでCCNRDY = 1
	MUTEDET	MUTE	マスタ／スレーブ 受信モードのみ	SAI_xIMレジスタのMUTEDETIE	SAI_xCLRFRレジスタでCMUTEDET = 1
	WCKCFG	ERROR	マスタ（SAI_xCR1レジスタで NODIV = 0の場合）	SAI_xIMレジスタのWCKCFGIE	SAI_xCLRFRレジスタでCWCKCFG = 1

割込みを有効にするには、次のシーケンスに従ってください。

1. SAI割込みを無効化します。
2. SAIを設定します。
3. SAI割込みソースを設定します。
4. SAIを有効化します。

69.6 SAIレジスタ

ペリフェラルレジスタには、ワード（32bit）単位でアクセスする必要があります。

69.6.1 SAIグローバル設定レジスタ（SAI_GCR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCOUT[1:0]		Res.	Res.	SYNCIN[1:0]	
										rw	rw			rw	rw

ビット31:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **SYNCOUT[1:0]**：同期出力

これらのビットは、ソフトウェアによってセット／クリアされます。

00：同期信号は出力されません。オーディオブロックがSPDIFとして設定されている場合、SYNCOUT[1:0] は「同期信号出力なし」に設定する必要があります。

01：ブロックAは他のSAI用に追加された同期に使用します。

10：ブロックBは他のSAI用に追加された同期に使用します。

11：予約済み両方のオーディオブロック（AおよびB）を無効化して、これらのビットを設定する必要があります。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **SYNCIN[1:0]**：同期入力

これらのビットは、ソフトウェアによってセット／クリアされます。

このフィールドの設定方法については、表 701：外部同期の選択を参照してください。

両方のオーディオブロック（AおよびB）を無効化して、これらのビットを設定する必要があります。

2つのオーディオブロックのうち1つが外部SAIで同期モードで動作するよう設定されている場合（SAI_ACR1またはSAI_BCR1レジスタでSYNCEN[1:0] = 10）、これらのビットは意味を持ちます。

69.6.2 SAI設定レジスタ1（SAI_ACR1）

アドレスオフセット：0x004

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **MCKEN**：マスタクロック生成イネーブル

0：マスタクロックは生成されません。

1：マスタクロックがSAIENビットから独立して生成されます。

ビット26 **OSR**：マスタクロックのオーバーサンプリング比

このビットは、NODIVビットが0にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比1のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタクロック周波数はセクション 69.4.8 : SAIクロックジェネレータで指定された式で計算されます。

オーディオブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット19 **NODIV**：分周回路なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が256または512に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率がFRL[7:0] に依存します。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DMAEN**:DMAイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMAは無効です。

1：DMAは有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでのDMAリクエストを回避するために、MODE[1:0] ビットはDMAENをセットする前に設定する必要があります。

ビット16 **SAIEN**：オーディオブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを0に設定し、このビットが0に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを1にセットする前に、それが0にセットされていることを確認します。そうしないと、有効化コマンドは考慮されません。

このビットは、SAIオーディオブロックの状態を制御できるようにします。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAIオーディオブロックは無効です。

1：SAIオーディオブロックは有効です。

注： SAIブロック（AまたはB）をマスタモードに設定した場合、SAIENビットをセットする前にSAIの入力にクロックが存在する必要があります。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIENがセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が2のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット0のデータがスロット1に複製されます。受信モードでは、スロット1は破棄され、スロット0から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット11:10 **SYNCEN[1:0]**：同期イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他のSAI内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済み

注： オーディオサブブロックは、SPDIFモードが有効な場合、非同期として設定する必要があります。

ビット9 **CKSTR**：クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIFオーディオプロトコルでは意味を持ちません。

0：SCK立ち上がりエッジでのSAIの変化によって信号が生成され、SCK立ち下がりエッジでのSAIのサンプリングによって信号が受信されます。

1：SCK立ち下がりエッジでのSAIの変化によって信号が生成され、SCK立ち上がりエッジでのSAIのサンプリングによって信号が受信されます。

ビット8 **LSBFIRST**：LSBファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC’97ではデータは常にMSBが先で転送されるため、このビットはAC’97オーディオプロトコルでは意味を持ちません。SPDIFではデータは常にLSB先頭で転送されるため、このビットはSPDIFオーディオプロトコルでは意味を持ちません。

0：データはMSBから先に転送されます。

1：データはLSBから先に転送されます。

ビット7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって8bitに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済み

001：予約済み

010：8bit

011：10bit

100：16bit

101：20bit

110：24bit

111：32bit

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用できるようになり、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（たとえば、I2S、LSB/MSB詰め、TDM、PCM/DSPなど）を処理することができます。

01：SPDIFプロトコル

10：AC'97プロトコル

11：予約済み

ビット1:0 **MODE[1:0]**：SAIxオーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIxオーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックをSPDIFモードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。

69.6.3 SAI設定レジスタ1（SAI_BCR1）

アドレスオフセット：0x024

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **MCKEN**：マスタクロック生成イネーブル

0：マスタクロックは生成されません。

1：マスタクロックがSAIENビットから独立して生成されます。

ビット26 **OSR**：マスタクロックのオーバーサンプリング比

このビットは、NODIVビットが0にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比1のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタクロック周波数はセクション 69.4.8 : SAIクロックジェネレータで指定された式で計算されます。

オーディオブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット19 **NODIV**：分周回路なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が256または512に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率がFRL[7:0] に依存します。

ビット18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DMAEN**:DMAイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMAは無効です。

1：DMAは有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでのDMAリクエストを回避するために、MODE[1:0] ビットはDMAENをセットする前に設定する必要があります。

ビット16 **SAIEN**：オーディオブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを0に設定し、このビットが0に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを1にセットする前に、それが0にセットされていることを確認します。そうしないと、有効化コマンドは考慮されません。

このビットは、SAIオーディオブロックの状態を制御できるようにします。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAIオーディオブロックは無効です。

1：SAIオーディオブロックは有効です。

注： SAIブロック（AまたはB）をマスタモードに設定した場合、SAIENビットをセットする前にSAIの入力にクロックが存在する必要があります。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIENがセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が2のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット0のデータがスロット1に複製されます。受信モードでは、スロット1は破棄され、スロット0から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット11:10 **SYNCEN[1:0]**：同期イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他のSAI内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済み

注： オーディオサブブロックは、SPDIFモードが有効な場合、非同期として設定する必要があります。

ビット9 **CKSTR**：クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIFオーディオプロトコルでは意味を持ちません。

0：SCK立ち上がりエッジでのSAIの変化によって信号が生成され、SCK立ち下がりエッジでのSAIのサンプリングによって信号が受信されます。

1：SCK立ち下がりエッジでのSAIの変化によって信号が生成され、SCK立ち上がりエッジでのSAIのサンプリングによって信号が受信されます。

ビット8 **LSBFIRST**：LSBファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC’97ではデータは常にMSBが先で転送されるため、このビットはAC’97オーディオプロトコルでは意味を持ちません。SPDIFではデータは常にLSB先頭で転送されるため、このビットはSPDIFオーディオプロトコルでは意味を持ちません。

0：データはMSBから先に転送されます。

1：データはLSBから先に転送されます。

ビット7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって8bitに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済み

001：予約済み

010：8bit

011：10bit

100：16bit

101：20bit

110：24bit

111：32bit

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用できるようになり、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（たとえば、I2S、LSB/MSB詰め、TDM、PCM/DSPなど）を処理することができます。

01：SPDIFプロトコル

10：AC'97プロトコル

11：予約済み

ビット1:0 **MODE[1:0]**：SAIxオーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIxオーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックをSPDIFモードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。マスタ送信モードでは、オーディオブロックはただちにFSおよびクロックの生成を開始します。

69.6.4 SAI設定レジスタ2（SAI_ACR2）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTEVAL	MUTE	TRIS	FFLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:14 **COMP[1:0]**：コンパンディングモード

これらのビットは、ソフトウェアによってセット／クリアされます。μ-LawとA-Law（対数）はCCITT G.711勧告の一部であり、使用される補数のタイプは *CPLビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み

10：μ-Lawアルゴリズム

11：A-Lawアルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1の補数表現。

1：2の補数表現。

注： コンパンディングモードが μ-LawアルゴリズムまたはA-Lawアルゴリズムの場合にのみ、このビットは有効です。

ビット12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDETフラグがセットされ、MUTEDETIEビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。このビットはオーディオブロックを有効化（SAIEN）する前に書込む必要があります。オーディオブロックがトランスミッタとして動作し、スロット数が2以下で、かつMUTEビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、このMUTEVALビットの値とは無関係に0になります。

スロット数が2以下でMUTEVAL = 1の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値0を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIFオーディオブロックには使用できません。

ビット5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が2以下の場合、ミュート値はMUTEVAL値に関連付けられますが、スロット数が2より大きい場合、ミュート値は0になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIFオーディオブロックには使用できません。

ビット4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックがSPDIFモードに設定されている場合、このビットは使用されません。このビットはSAIを無効にして設定する必要があります。

詳細については、セクション ：非アクティブスロットでの出力データライン管理を参照してください。

0：スロットが非アクティブのときでも、SD出力ラインはSAIによって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりでSD出力ラインを開放（HI-Z）します。

ビット3 **FFLUSH**：FIFOのフラッシュ。

このビットはソフトウェアでセットされます。常に0が読み出されます。このビットはSAIが無効なときに設定する必要があります。

0：FIFOをフラッシュしません。

1：FIFOをフラッシュします。このビットを1に設定すると、FIFOのフラッシュがトリガされます。すべての内部FIFOポインタ（読出しと書込み）がクリアされます。この場合、FIFOに残っているデータが失われます（送受信データはそれ以上失われません）。FIFOを一掃する前に、SAI DMAストリーム／割込みを無効化する必要があります。

ビット2:0 **FTH[2:0]**：FIFO閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFOエンプティ

001：¼ FIFO

010：½ FIFO

011：¾ FIFO

100：FIFOフル

101：予約済み

110：予約済み

111：予約済み

69.6.5 SAI設定レジスタ2（SAI_BCR2）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTEVAL	MUTE	TRIS	FFLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:14 **COMP[1:0]**：コンパンディングモード

これらのビットは、ソフトウェアによってセット／クリアされます。μ-LawとA-Law（対数）はCCITT G.711勧告の一部であり、使用される補数のタイプは *CPLビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み

10：μ-Lawアルゴリズム

11：A-Lawアルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1の補数表現。

1：2の補数表現。

注： コンパンディングモードが μ-LawアルゴリズムまたはA-Lawアルゴリズムの場合にのみ、このビットは有効です。

ビット12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDETフラグがセットされ、MUTEDETIEビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。このビットはオーディオブロックを有効化（SAIEN）する前に書込む必要があります。オーディオブロックがトランスミッタとして動作し、スロット数が2以下で、かつMUTEビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、このMUTEVALビットの値とは無関係に0になります。

スロット数が2以下でMUTEVAL = 1の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値0を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIFオーディオブロックには使用できません。

ビット5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が2以下の場合、ミュート値はMUTEVAL値に関連付けられますが、スロット数が2より大きい場合、ミュート値は0になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIFオーディオブロックには使用できません。

ビット4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックがSPDIFモードに設定されている場合、このビットは使用されません。このビットはSAIを無効にして設定する必要があります。

詳細については、セクション ：非アクティブスロットでの出力データライン管理を参照してください。

0：スロットが非アクティブのときでも、SD出力ラインはSAIによって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりでSD出力ラインを開放（HI-Z）します。

ビット3 **FFLUSH**：FIFOのフラッシュ。

このビットはソフトウェアでセットされます。常に0が読み出されます。このビットはSAIが無効なときに設定する必要があります。

0：FIFOをフラッシュしません。

1：FIFOをフラッシュします。このビットを1に設定すると、FIFOのフラッシュがトリガされます。すべての内部FIFOポインタ（読出しと書込み）がクリアされます。この場合、FIFOに残っているデータが失われます（送受信データはそれ以上失われません）。FIFOを一掃する前に、SAI DMAストリーム／割込みを無効化する必要があります。

ビット2:0 **FTH[2:0]**：FIFO閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFOエンプティ

001：¼ FIFO

010：½ FIFO

011：¾ FIFO

100：FIFOフル

101：予約済み

110：予約済み

111：予約済み

69.6.6 SAIフレーム設定レジスタ（SAI_AFRCR）

アドレスオフセット：0x00C

リセット値：0x0000 0007

注： このレジスタは、AC'97およびSPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FSはスロット0の先頭ビットで開始されます。

1：FSはスロット0の先頭ビットの1bit前でアサートされます。

ビット17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS信号上のフレーム開始のレベルを設定します。AC'97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FSはアクティブロー（立ち下がりエッジ）になります

1：FSはアクティブハイ（立ち上がりエッジ）になります

ビット16 **FSDEF**：フレーム同期定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS信号がフレーム開始信号になります

1：FS信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTRレジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（たとえば、このビットは、I2SプロトコルまたはMSB/LSB詰めプロトコルなどに対して設定する必要があります）。

AC'97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおけるFS信号のアクティブレベルの長さを、オーディオフレームにおけるFS信号のアクティブレベルの長さを、ビットクロック数（SCK）+ 1（FSALL[6:0] + 1）で、指定します。

これらのビットは意味を持たないので、AC'97 またはSPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長をSCKクロックサイクル数で定義します。フレーム内のビット数はFRL[7:0] + 1に等しくなります。

オーディオフレームで転送する最小ビット数は8でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが8bitで、SAI_xSLOTRレジスタのNBSLOT[4:0] でスロット0が1つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。

マスタモードで、マスタクロック（MCLK_xピンで使用可能）が使用されている場合、フレーム長は、8から256のうちの2の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を8から256のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC’97 またはSPDIF オーディオブロックの設定には使用されません。オーディオブロックを無効化して、これらのビットを設定する必要があります。

69.6.7 SAIフレーム設定レジスタ（SAI_BFRCR）

アドレスオフセット：0x02C

リセット値：0x0000 0007

注： このレジスタは、AC’97 およびSPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC’97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FSはスロット0の先頭ビットで開始されます。

1：FSはスロット0の先頭ビットの1bit前でアサートされます。

ビット17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS信号上のフレーム開始のレベルを設定します。AC’97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FSはアクティブロー（立ち下がりエッジ）になります

1：FSはアクティブハイ（立ち上がりエッジ）になります

ビット16 **FSDEF**：フレーム同期定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS信号がフレーム開始信号になります

1：FS信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTRレジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（たとえば、このビットは、I2SプロトコルまたはMSB/LSB詰めプロトコルなどに対して設定する必要があります）。

AC’97 またはSPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおけるFS信号のアクティブレベルの長さを､オーディオフレームにおけるFS信号のアクティブレベルの長さを、ビットクロック数（SCK）+ 1（FSALL[6:0] + 1）で、指定します。

これらのビットは意味を持たないので、AC'97 またはSPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長をSCKクロックサイクル数で定義します。フレーム内のビット数はFRL[7:0] + 1に等しくなります。

オーディオフレームで転送する最小ビット数は8でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが8bitで、SAI_xSLOTRレジスタのNBSLOT[4:0] でスロット0が1つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。

マスタモードで、マスタクロック（MCLK_xピンで使用可能）が使用されている場合、フレーム長は、8から256のうちの2の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を8から256のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC'97 またはSPDIF オーディオブロックの設定には使用されません。

69.6.8 SAIスロットレジスタ（SAI_ASLOTR）

アドレスオフセット：0x010

リセット値：0x0000 0000

注： このレジスタは、AC'97およびSPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:16 **SLOTEN[15:0]**：スロットイネーブル。

これらのビットは、ソフトウェアによってセット／クリアされます。

各SLOTENビットは0から15までのスロット位置に対応します（最大16スロット）。

0：非アクティブスロット。

1：アクティブスロット。

オーディオブロックを無効化して、スロットを有効にする必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1を表します（非アクティブスロット数を含む）。最大スロット数は16です。

SAI_xFRCRレジスタのFSDEFビットがセットされている場合、スロット数は偶数である必要があります。

オーディオブロックを無効化して、スロット数を設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

ビット7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAIの動作は不確定です。

SDラインの駆動方法については、非アクティブスロットでの出力データライン管理を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1レジスタのDS[3:0] で指定）に一致させます。

01：16bit

10：32bit

11：予約済み

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に0に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

69.6.9 SAIスロットレジスタ（SAI_BSLOTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

注： このレジスタは、AC'97およびSPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット31:16 **SLOTEN[15:0]**：スロットイネーブル。

これらのビットは、ソフトウェアによってセット／クリアされます。

各SLOTENビットは0から15までのスロット位置に対応します（最大16スロット）。

0：非アクティブスロット。

1：アクティブスロット。

オーディオブロックを無効化して、スロットを有効にする必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1を表します（非アクティブスロット数を含む）。最大スロット数は16です。

SAI_xFRCRレジスタのFSDEFビットがセットされている場合、スロット数は偶数である必要があります。

オーディオブロックを無効化して、スロット数を設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

ビット7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAIの動作は不確定です。

SDラインの駆動方法については、非アクティブスロットでの出力データライン管理を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1レジスタのDS[3:0] で指定）に一致させます。

01：16bit

10：32bit

11：予約済み

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に0に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 またはSPDIF モードでは無視されます。

69.6.10 SAI割込みマスクレジスタ（SAI_AIM）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LFSDETIE**：レイトフレーム同期検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのLFSDETビットがセットされている場合は割込みが生成されます。

このビットは、AC'97、SPDIFモードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット5 **AFSDETIE**：予測フレーム同期検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのAFSDETビットがセットされている場合は割込みが生成されます。

このビットは、AC'97、SPDIFモードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット4 **CNRDYIE**：コーデックノットレディ割込みイネーブル（AC'97）。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC'97フレームのスロット0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSRレジスタのCNRDYフラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介してAC'97モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット3 **FREQIE**：FIFOリクエスト割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのFREQビットがセットされている場合は割込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割込みを回避するために、MODEビットはFREQIEをセットする前に設定する必要があります。

ビット2 **WCKCFGIE**：不正クロック設定割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0かつ NODIV = 0）にのみ考慮されます。

SAI_xSRレジスタのWCKCFGフラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット1 **MUTEDETIE**：ミュート検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのMUTEDETビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット0 **OVRUDRIE**：オーバーラン/アンダーラン割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのOVRUDRビットがセットされている場合は割込みが生成されます。

69.6.11 SAI割込みマスクレジスタ（SAI_BIM）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LFSDETIE**：レイトフレーム同期検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのLFSDETビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット5 **AFSDETIE**：予測フレーム同期検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのAFSDETビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット4 **CNRDYIE**：コーデックノットレディ割込みイネーブル（AC'97）。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC’97フレームのスロット0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSRレジスタのCNRDYフラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介してAC’97モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット3 **FREQIE**：FIFOリクエスト割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのFREQビットがセットされている場合は割込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割込みを回避するために、MODEビットはFREQIEをセットする前に設定する必要があります。

ビット2 **WCKCFGIE**：不正クロック設定割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0かつ NODIV = 0）にのみ考慮されます。

SAI_xSRレジスタのWCKCFGフラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット1 **MUTEDETIE**：ミュート検出割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのMUTEDETビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット0 **OVRUDRIE**：オーバーラン/アンダーラン割込みイネーブル。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSRレジスタのOVRUDRビットがセットされている場合は割込みが生成されます。

69.6.12 SAIステータスレジスタ（SAI_ASR）

アドレスオフセット：0x018

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **FLVL[2:0]**：FIFOレベル閾値

このビットは読出し専用です。FIFOレベル閾値フラグはハードウェアにより管理され、その設定はSAIブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFOエンプティ（送信モードおよび受信モード）

001：FIFO ＜ ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFOただしフルではない（送信モード）、¾ ≤ FIFOただしフルではない（受信モード）

101：FIFOフル（送信モードおよび受信モード）

その他：予約済み

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LFSDET**：レイトフレーム同期検出。

このビットは読出し専用です。

0：エラーはありません。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは、AC'97 またはSPDIF モードでは使用されません。

SAI_xIMレジスタのLFSDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCLFSDETビットをセットすると、このフラグはクリアされます。

ビット5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーはありません。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットはAC'97 またはSPDIF モードでは使用されません。

SAI_xIMレジスタのAFSDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCAFSDETビットをセットすると、このフラグはクリアされます。

ビット4 **CNRDY**：コーデックノットレディ。

このビットは読出し専用です。

0：外部AC'97コーデックが動作可能状態です

1：外部AC'97コーデックが動作可能状態ではありません

SAI_xCR1レジスタでAC'97オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIMレジスタのCNRDYIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCCNRDYビットをセットすると、このフラグはクリアされます。

ビット3 **FREQ**：FIFOリクエスト。

このビットは読出し専用です。

0：FIFOリクエストはありません。

1：SAI_xDRに対する読出しまたは書込みのFIFOリクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFOリクエストはSAI_xDR内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFOリクエストはSAI_xDR内の読出しリクエスト動作に関わります。

SAI_xIMレジスタのFREQIEビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 69.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCRレジスタのFRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作し（MODE[1] = 0）かつ NODIV = 0の場合にのみ使用されます。

SAI_xIMレジスタのWCKCFGIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCWCKCFGビットをセットすると、このフラグはクリアされます。

ビット1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対してSD入力ラインでミュート値が検出されました（0値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する0値が受信された場合、このフラグがセットされます（SAI_xCR2レジスタのMUTECNTビットをセット）。

SAI_xIMレジスタのMUTEDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCMUTEDETビットをセットすると、このフラグはクリアされます。

ビット0 **OVRUDR**：オーバーラン／アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIMレジスタのOVRUDRIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCOVRUDRビットをセットすると、このフラグはクリアされます。

69.6.13 SAIステータスレジスタ（SAI_BSR）

アドレスオフセット：0x038

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **FLVL[2:0]**：FIFOレベル閾値

このビットは読出し専用です。FIFOレベル閾値フラグはハードウェアにより管理され、その設定はSAIブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFOエンプティ（送信モードおよび受信モード）

001：FIFO < ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFOただしフルではない（送信モード）、¾ ≤ FIFOただしフルではない（受信モード）

101：FIFOフル（送信モードおよび受信モード）

その他：予約済み

ビット15:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **LFSDET**：レイトフレーム同期検出。

このビットは読出し専用です。

0：エラーはありません。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは、AC'97またはSPDIFモードでは使用されません。

SAI_xIMレジスタのLFSDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCLFSDETビットをセットすると、このフラグはクリアされます。

ビット5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーはありません。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットはAC'97またはSPDIFモードでは使用されません。

SAI_xIMレジスタのAFSDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCAFSDETビットをセットすると、このフラグはクリアされます。

ビット4 **CNRDY**：コーデックノットレディ。

このビットは読出し専用です。

0：外部AC'97コーデックが動作可能状態です

1：外部AC'97コーデックが動作可能状態ではありません

SAI_xCR1レジスタでAC'97オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIMレジスタのCNRDYIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCCNRDYビットをセットすると、このフラグはクリアされます。

ビット3 **FREQ**：FIFOリクエスト。

このビットは読出し専用です。

0：FIFOリクエストはありません。

1：SAI_xDRに対する読出しまたは書込みのFIFOリクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFOリクエストはSAI_xDR内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFOリクエストはSAI_xDR内の読出しリクエスト動作に関わります。

SAI_xIMレジスタのFREQIEビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 69.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCRレジスタのFRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作し（MODE[1] = 0）かつNODIV= 0の場合にのみ使用されます。

SAI_xIMレジスタのWCKCFGIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCWCKCFGビットをセットすると、このフラグはクリアされます。

ビット1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対してSD入力ラインでミュート値が検出されました（0値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する0値が受信された場合、このフラグがセットされます（SAI_xCR2レジスタのMUTECNTビットをセット）。

SAI_xIMレジスタのMUTEDETIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCMUTEDETビットをセットすると、このフラグはクリアされます。

ビット0 **OVRUDR**：オーバーラン／アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIMレジスタのOVRUDRIEビットがセットされている場合は割込みが生成されます。

ソフトウェアによってSAI_xCLRFRレジスタのCOVRUDRビットをセットすると、このフラグはクリアされます。

69.6.14 SAIクリアフラグレジスタ（SAI_ACLRFR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CLFSDET**：レイトフレーム同期検出フラグクリア。

このビットは書込み専用です。

このビットを1に設定するとSAI_xSRレジスタのLFSDETフラグがクリアされます。

このビットは、AC’97 またはSPDIF モードでは使用されません。

このビットを読み出すと、常に0値が返されます。

ビット5 **CAFSDET**：予測フレーム同期検出フラグクリア。

このビットは書込み専用です。

このビットを1に設定するとSAI_xSRレジスタのAFSDETフラグがクリアされます。

このビットはAC’97 またはSPDIF モードでは使用されません。

このビットを読み出すと、常に0値が返されます。

ビット4 **CCNRDY**：コーデックノットレディフラグクリア。

このビットは書込み専用です。

このビットを1に設定するとSAI_xSRレジスタのCNRDYフラグがクリアされます。

SAI_xCR1レジスタでAC’97オーディオプロトコルが選択された場合にのみ、このビットが使用されます。

このビットを読み出すと、常に0値が返されます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのWCKCFGフラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつSAI_xCR1レジスタで NODIV = 0の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に0値が返されます。

ビット1 **CMUTEDET**：ミュート検出フラグ。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのMUTEDETフラグがクリアされます。
このビットを読み出すと、常に0値が返されます。

ビット0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのOVRUDRフラグがクリアされます。
このビットを読み出すと、常に0値が返されます。

69.6.15 SAIクリアフラグレジスタ（SAI_BCLRFR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのLFSDETフラグがクリアされます。
このビットは、AC’97またはSPDIF モードでは使用されません。
このビットを読み出すと、常に0値が返されます。

ビット5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのAFSDETフラグがクリアされます。
このビットはAC’97またはSPDIF モードでは使用されません。
このビットを読み出すと、常に0値が返されます。

ビット4 **CCNRDY**：コーデックノットレディフラグクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのCNRDYフラグがクリアされます。
SAI_xCR1レジスタでAC’97オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に0値が返されます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのWCKCFGフラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつSAI_xCR1レジスタで NODIV = 0の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に0値が返されます。

ビット1 **CMUTEDET**：ミュート検出フラグ。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのMUTEDETフラグがクリアされます。
このビットを読み出すと、常に0値が返されます。

ビット0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを1に設定するとSAI_xSRレジスタのOVRUDRフラグがクリアされます。
このビットを読み出すと、常に0値が返されます。

69.6.16 SAIデータレジスタ（SAI_ADR）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATA[31:0]**：データ
FIFOがフルでない場合、このレジスタへ書き込むと、FIFOへの書込みが行われます。
FIFOがエンプティでない場合、このレジスタを読み出すと、FIFOからの読出しが行われます。

69.6.17 SAIデータレジスタ（SAI_BDR）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DATA[31:0]**：データ
FIFOがフルでない場合、このレジスタへ書き込むと、FIFOへの書込みが行われます。
FIFOがエンプティでない場合、このレジスタを読み出すと、FIFOからの読出しが行われます。

69.6.18 SAI PDM制御レジスタ（SAI_PDMCR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
				rw	rw	rw	rw			rw	rw				rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **CKEN4**：ビットストリームクロック4番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK4クロックは無効です。

1：SAI_CK4クロックは有効です。

注： このビットをPDMEN = 1のときに設定するのは推奨しません。

SAI_CK4は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

ビット10 **CKEN3**：ビットストリームクロック3番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK3クロックは無効です。

1：SAI_CK3クロックは有効です。

注： このビットをPDMEN = 1のときに設定するのは推奨しません。

SAI_CK3は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

ビット9 **CKEN2**：ビットストリームクロック2番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK2クロックは無効です。

1：SAI_CK2クロックは有効です。

注： このビットをPDMEN = 1のときに設定するのは推奨しません。

SAI_CK2は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

ビット8 **CKEN1**：ビットストリームクロック1番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK1クロックは無効です。

1：SAI_CK1クロックは有効です。

注： このビットをPDMEN = 1のときに設定するのは推奨しません。

SAI_CK1は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **MICNBR[1:0]**：マイクロフォンの本数

このビットは、ソフトウェアによってセット／クリアされます。

00：マイクロフォン2本の設定

01：マイクロフォン4本の設定

10：マイクロフォン6本の設定

11：マイクロフォン8本の設定

注： このフィールドをPDMEN = 1のときに設定するのは推奨しません。*

データライン一式は、すべてのSAIインスタンスで使用できるわけではありません。詳細については、セクション 69.3：SAIの実装を参照してください。

ビット3:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **PDMEN**：PDMイネーブル

このビットは、ソフトウェアによってセット／クリアされます。このビットを使って、PDMインタフェースブロックの状態を制御できます。

PDMインタフェースを有効にする前にSAIがすでにTDMマスタモードで動作していることを確認してください。

0：PDMインタフェースは無効です。

1：PDMインタフェースは有効です。

69.6.19 SAI PDMディレイレジスタ（SAI_PDMDLY）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:28 **DLYM4R[2:0]**：**ペアNo.4** の2番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

000：ディレイなし

001：1 T_{SAI_CK} 周期のディレイ

010：2 T_{SAI_CK} 周期のディレイ

.....

111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D4ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26:24 **DLYM4L[2:0]**：ペアNo.4の1番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D4ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット23 予約済みであり、リセット値に保持する必要があります。

ビット22:20 **DLYM3R[2:0]**：ペアNo.3の2番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D3ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **DLYM3L[2:0]**：ペアNo.3の1番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D3ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14:12 **DLYM2R[2:0]**：ペアNo.2の2番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D2ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:8 **DLYM2L[2:0]**：ペアNo.2の1番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D2ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **DLYM1R[2:0]**：ペアNo.1の2番目のマイクロフォンのディレイライン調整
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D1ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **DLYM1L[2:0]**：ペアNo.1の1番目のマイクロフォンのディレイライン調整
このビットは、ソフトウェアによってセット／クリアされます。
000：ディレイなし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D1ラインが使用可能な場合にのみ使用できます。使用可能かどうか確認するには、セクション 69.3：SAIの実装を参照してください。

69.6.20 SAIレジスタマップ

表710.SAIレジスタマップとリセット値

オフセット	レジスタ名 リセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	SAI_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCOUT[1:0]		Res.	Res.	SYNCIN[1:0]	
	リセット値																											0	0			0	0
0x0004 または0x0024	SAI_xCR1	Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN	Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]		Res.	PRTCFG[1:0]		MODE[1:0]	
	リセット値					0	0	0	0	0	0	0	0	0		0	0			0	0	0	0	0	0	0	1	0		0	0	0	0
0x0008 または0x0028	SAI_xCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMP[1:0]		CPL	MUTECN[5:0]						MUTE VAL	MUTE	TRIS	FFLUS	FTH[2:0]		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C または0x002C	SAI_xFRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF	Res.	FSALL[6:0]							FRL[7:0]							
	リセット値														0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0010 または0x0030	SAI_xSLOTR	SLOTEN[15:0]																Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0014 または0x0034	SAI_xIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
	リセット値																										0	0	0	0	0	0	0
0x0018 または0x0038	SAI_xSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
	リセット値														0	0	0										0	0	0	0	1	0	0
0x001Cまたは0x003C	SAI_xCLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
	リセット値																										0	0	0		0	0	0
0x0020 または0x0040	SAI_xDR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	SAI_PDMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
	リセット値																					0	0	0	0			0	0				0

表710.SAIレジスタマップとリセット値 （続き）

オフセット	レジスタ名 リセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0048	SAI_PDMDLY	Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]			Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	リセット値		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0

レジスタ境界アドレスについては、セクション2.3を参照してください。

70 FDコントローラエリアネットワーク（FDCAN）

70.1 概要

コントローラエリアネットワーク（CAN）のサブシステム（図873を参照）は、1つのCANモジュール、共有メッセージRAM、および設定ブロックで構成されます。これらの部分の各ベースアドレスについてはメモリマップを参照してください。

モジュール（FDCAN）は、ISO 11898-1:2015（CANプロトコル仕様バージョン2.0パートA、B）およびCAN FDプロトコル仕様バージョン1.0に準拠しています。

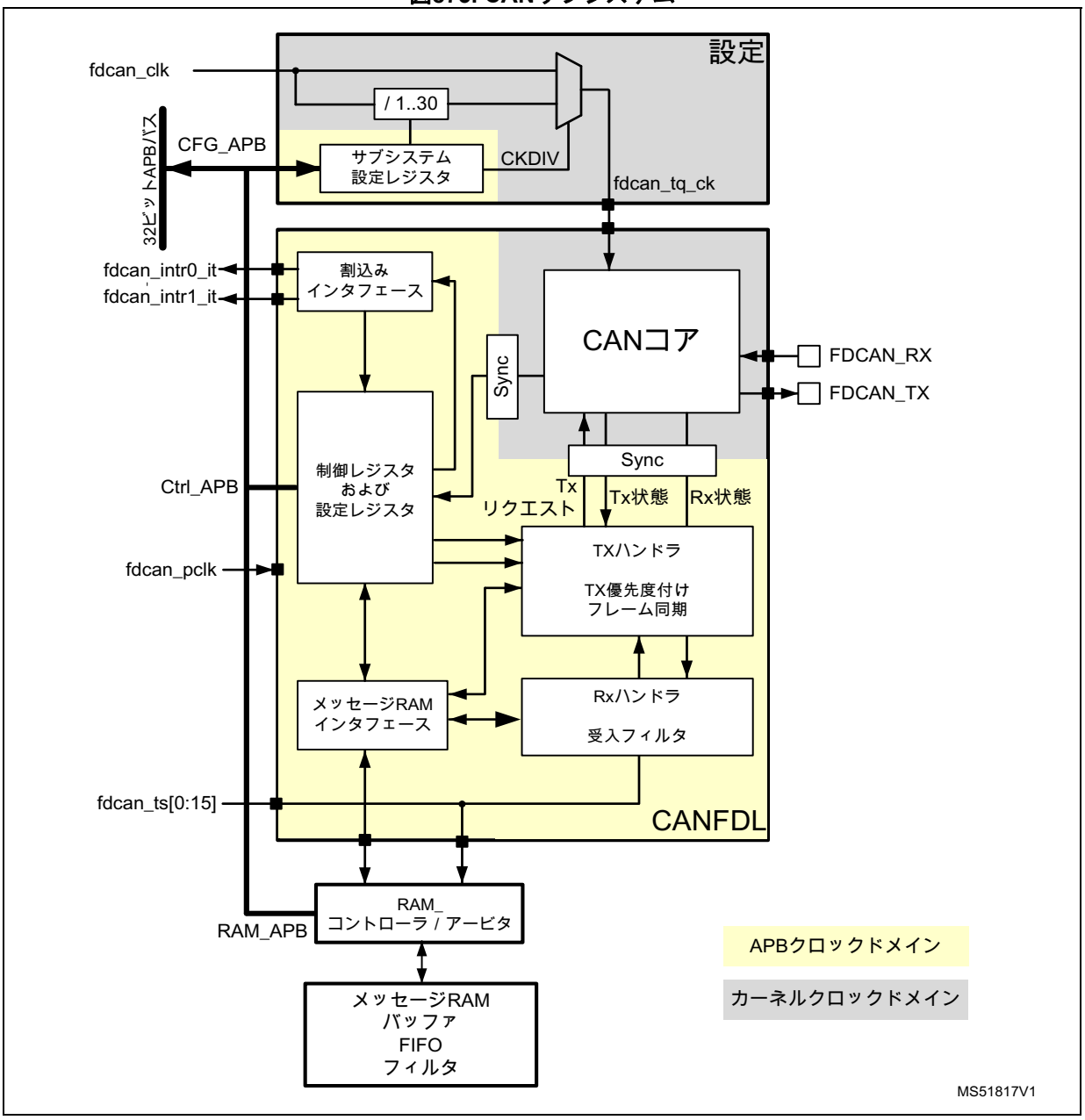
FDCANインスタンスあたり0.8 KBのメッセージRAMには、フィルタ、受信FIFO、送信イベントFIFO、および送信FIFOが実装されます。

CANサブシステムI/O信号およびピンの詳細は、それぞれ表711および図873に示します。

表711.CANサブシステムI/O信号

名前	タイプ	説明
fdcan_ck	デジタル入力	CANサブシステムカーネルクロック入力
fdcan_pclk		CANサブシステムAPBインタフェースクロック入力
fdcan_intr0_it	デジタル出力	FDCAN interrupt0
fdcan_intr1_it		FDCAN interrupt1
fdcan_ts[0:15]	-	外部タイムスタンプベクトル
FDCAN_RX	デジタル入力	FDCAN受信ピン
FDCAN_TX	デジタル出力	FDCAN送信ピン
APBインタフェース	デジタル入出力	設定、制御、およびRAMアクセス用の複数のpselを持つシングルAPP

図873. CANサブシステム

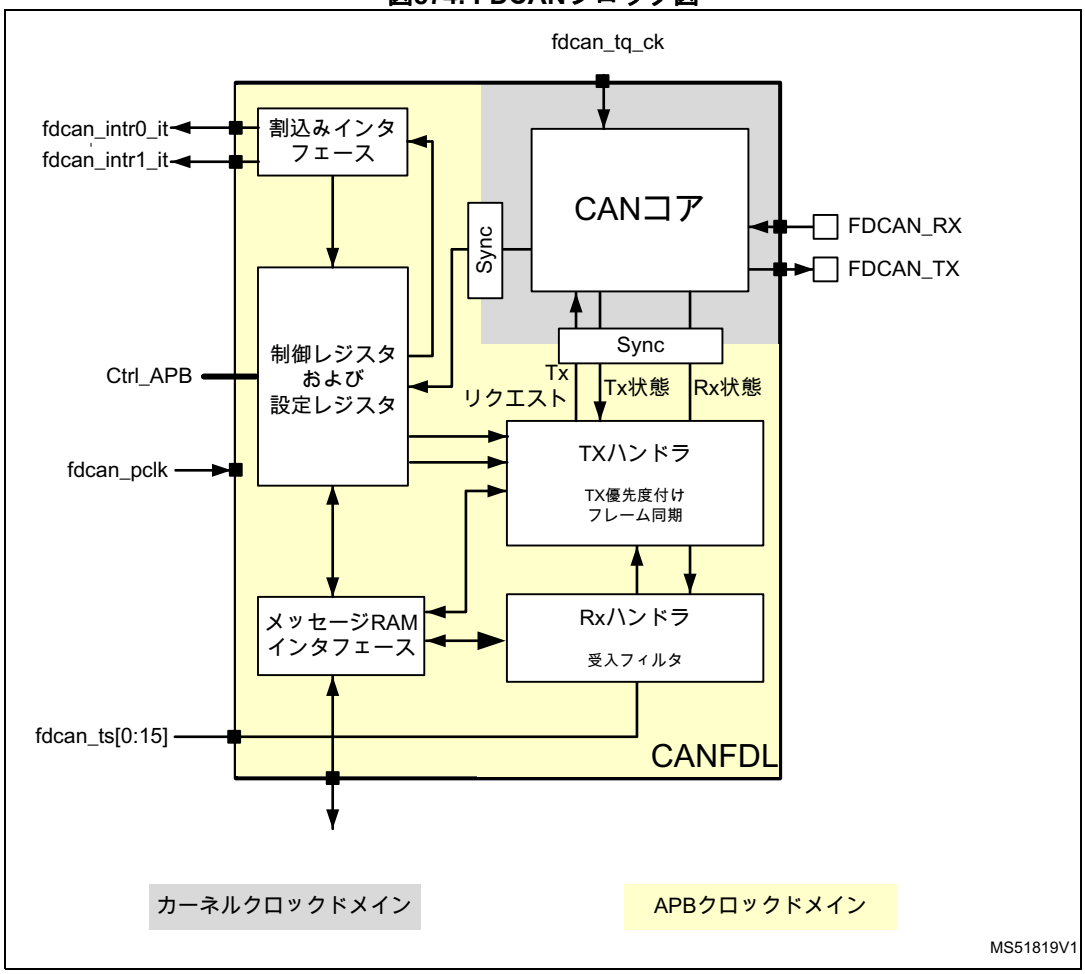


70.2 FDCANの主な機能

- CANプロトコルバージョン2.0パートA、BおよびISO 11898-1:2015、-4に準拠
- 最大64データバイトをサポートするCAN FD
- CANエラーログ記録
- AUTOSARおよびJ1939サポート
- 改良された受入フィルタ
- それぞれ3つのペイロードを持つ2つの受信FIFO（ペイロード当たり最大64バイト）
- 高優先メッセージの受信で個別の信号
- 3つのペイロードの設定可能な送信FIFO（ペイロード当たり最大64バイト）
- 送信イベントFIFO
- プログラム可能なループバックテストモード
- マスク可能なモジュール割込み
- 2つのクロックドメイン：APBバスインタフェースおよびCANコアカーネルクロック
- パワーダウンサポート

70.3 FDCANの機能詳細

図874. FDCANブロック図



2つの割込みライン

FDCANペリフェラルには、fdcan_intr0_itおよびfdcan_intr1_itの2つの割込みラインがあります。

FDCAN_ILEレジスタでEINT0およびEINT1bitをプログラムすることで、割込みラインを個別に有効化または無効化できます。

CANコア

CANコアには、プロトコルコントローラおよび受信／送信シフトレジスタがあります。CANコアは、すべてのISO 11898-1:2015プロトコル機能を処理し、11bitおよび29bitの両識別子をサポートしています。

Sync

同期ブロックは、APBクロックドメインとCANカーネルクロックドメインとの信号を同期します。

Txハンドラ

メッセージRAMからCANコアへのメッセージ転送を制御します。送信に対して、最大3個のTxバッファを使用できます。Txバッファは、Tx FIFOまたはTxキューとして使用できます。TxイベントFIFO

には、対応するメッセージIDとともにTxタイムスタンプを格納できます。送信キャンセルもサポートしています。

Rxハンドラ

CANコアから受信したメッセージの外部メッセージRAMへの転送を制御します。Rxハンドラは、受入フィルタを通過したすべてのメッセージのストレージ用に、2つの受信FIFOをサポートしています。Rxタイムスタンプが、各メッセージとともに格納されます。11bitのIDには最大28個のフィルタ、29bitのIDには最大8個のフィルタを定義できます。

APBインタフェース

設定レジスタ、コントローラの設定、およびRAMアクセス用のAPBバスにFDCANを接続します。

メッセージRAMインタフェース

FDCANにはRAMコントローラ／アービタ経由で外部の1 KBメッセージRAMが接続されています。

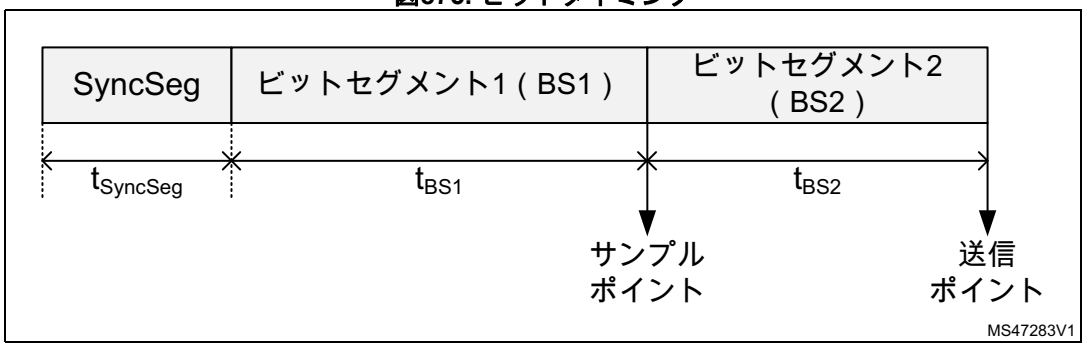
70.3.1 ビットタイミング

ビットタイミング回路は、シリアルバスラインを監視し、スタートビットエッジでの同期と後続エッジでの再同期によって、サンプリングとサンプリングポイントの調整を行います。

図875に示すように、この動作は、ビット時間を次のように3つのセグメントに分割するとわかりやすく説明できます。

- 同期セグメント（SYNC_SEG）：この1タイムクォンタム（1 x tq）に固定された長さの時間セグメント内にビット変化が生じることが想定されます。
- ビットセグメント 1（BS1）：サンプルポイントの位置を定義します。これには、CAN 規格のPROP_SEGとPHASE_SEG1が含まれます。セグメントの時間は1～16タイムクォンタムの間でプログラミング可能ですが、ネットワークのさまざまなノード間の周波数の差に起因する正の位相ドリフトを補正するために、自動的に延長されることがあります。
- ビットセグメント 2（BS2）：送信ポイントの位置を定義します。これは、CAN 規格のPHASE_SEG2に相当します。セグメントの時間は1～8タイムクォンタムの間でプログラミング可能ですが、負の位相ドリフトを補正するために、自動的に短縮されることもあります。

図875. ビットタイミング



ボーレートはビット時間の逆数（ボーレート = 1／ビット時間）で、3つのコンポーネントの和です。図875は、ビット時間 = $t_{SyncSeg}$ + t_{BS1} + t_{BS2} を示します。ここで、

- 公称ビット時間の場合
  - tq = (FDCAN_NBTP.NBRP[8:0] + 1) * $t_{fdcan_tq_clk}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_NBTP.NTSEG1[7:0] + 1)
  - t_{BS2} = tq * (FDCAN_NBTP.NTSEG2[6:0] + 1)
- データビット時間の場合
  - tq = (FDCAN_DBTP.DBRP[4:0] + 1) * $t_{fdcan_tq_clk}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_DBTP.DTSEG1[4:0] + 1)
  - t_{BS2} = tq * (FDCAN_DBTP.DTSEG2[3:0] + 1)

（再）同期ジャンプ幅（SJW）は、ビットセグメントの延長／短縮量の上限を定め、1～4タイムクォンタムの間でプログラミング可能です。

有効なエッジは、コントローラ自体はリセッシブビットを送信しないという条件で、あるビット時間において、ドミナントからリセッシブへのバスレベルの最初の遷移と定義されます。

有効なエッジがSYNC_SEGではなくBS1で検出された場合、サンプルポイントが遅くなるように、BS1は最大でSJW延長されます。

逆に、有効なエッジがSYNC_SEGではなくBS2で検出された場合、送信ポイントが早くなるように、BS2は最大でSJW短縮されます。

プログラミングエラーに対する安全対策として、ビットタイミングレジスタは、デバイスがSTANDBYモードの場合にのみ設定可能です。レジスタFDCAN_DBTPおよびFDCAN_NBTP（それぞれデータビット時間および公称ビット時間専用）は、CCCR.CCEとCCCR.INITがセットされている場合にのみアクセス可能です。

注： **CANビットタイミングと再同期メカニズムの詳細については、ISO 11898-1規格を参照してください。**

70.3.2 動作モード

設定

IPバージョン、ハードウェア、および入力クロック分周回路を設定します。クロック分周回路を0にセットすると、主な入力クロックはそのまま使用されます。

ソフトウェアの初期化

ソフトウェアの初期化は、FDCAN_CCCRレジスタでINITビットをセットするか、ソフトウェアやハードウェアリセット、Bus_Offへの移行で開始されます。FDCAN_CCCRレジスタでINITビットをセットしている間、CANバスとのメッセージ転送は停止し、CANバス出力FDCAN_TXのステータスはリセッシブ（ハイ）になります。EML（エラー管理ロジック）のカウンタは変更されません。FDCAN_CCCRのINITビットのセットで、設定レジスタは変更されません。FDCAN_CCCRのINITビットのクリアで、ソフトウェア初期化は終了します。その後、ビットストリームプロセッサ（BSP）は11個の連続したリセッシブビットの発生を待つことで（Bus_Idle）、CANバスでのデータ転送と同期してから、バスアクティビティに移行してメッセージ転送を開始できます。

FDCAN設定レジスタへのアクセスは、FDCAN_CCCRレジスタのINITビットとFDCAN_CCCRレジスタのCCEビットの両方がセットされているときのみ有効です。

FDCAN_CCCRレジスタのCCEビットは、FDCAN_CCCRのINITビットがセットされている間のみセット／クリアできます。FDCAN_CCCRレジスタのCCEビットは、FDCAN_CCCRのINITビットがクリアされているとき、自動的にクリアされます。

FDCAN_CCCRレジスタのCCEビットがセットされている場合、次のレジスタはリセットされます。

- FDCAN_HPMS - 高優先メッセージステータス
- FDCAN_RXF0S - Rx FIFO 0ステータス
- FDCAN_RXF1S - Rx FIFO 1ステータス
- FDCAN_TXFQS - Tx FIFO／キューステータス
- FDCAN_TXBRP - Txバッファリクエストペンディング
- FDCAN_TXBTO - Txバッファ送信発生
- FDCAN_TXBCF - Txバッファキャンセル終了
- FDCAN_TXEFS - TxイベントFIFOステータス

FDCAN_TOCVレジスタのタイムアウトカウンタ値TOCビットは、FDCAN_CCCRのCCEビットがセットされているとき、FDCAN_TOCCレジスタのTOPビットによって設定された値にプリセットされます。

さらに、TxハンドラおよびRxハンドラのステートマシンは、FDCAN_CCCRのCCEビットがセットされている間はアイドル状態で保持されます。

FDCAN_CCCRレジスタのCCEビットがクリアされている場合のみ、次のレジスタは書込み可能です。

- TXBAR - Txバッファ追加リクエスト
- TXBCR - Txバッファキャンセルリクエスト

FDCAN_CCCRのTESTビットおよびFDCAN_CCCRのMONビットは、CCCRのINITビットとCCCRレジスタのCCEビットの両方がセットされているときのみ、ソフトウェアによってセットできます。両ビットは常時リセットできます。FDCAN_CCCRのDARビットは、FDCAN_CCCRのINITビットとFDCAN_CCCRレジスタのCCEビットの両方がセットされているときのみセット／クリアできます。

通常動作

ハードウェアリセット後のFDCANのデフォルト動作モードは、イベント駆動CAN通信です。TT動作モードはサポートされていません。

FDCANが初期化され、FDCAN_CCCRレジスタのINITビットがクリアされると、FDCANはCANバスと同期し、通信できるようになります。

受入フィルタを通過した後、メッセージIDとDLCを含む受信メッセージは、Rx FIFO 0やRx FIFO 1に格納されます。

送信されるメッセージの場合、Tx FIFOやTxキューを初期化または更新できます。リモートフレーム受信時の自動送信はサポートされません。

CAN FD動作

FDCANプロトコルは2種類あります。

1. CANフレームのデータフィールドが8バイトより長くなるロングフレームモード（LFM）
2. CANフレームの制御フィールド、データフィールド、およびCRCフィールドがフレームの開始と終了時より高いビットレートで送信される高速フレームモード（FFM）

高速フレームモードは、ロングフレームモードと組み合わせて使用できます。

11bitの識別子のCANフレームで以前に予約済みであったビットと、29bitの識別子のCANフレームで以前予約済みであった一番最初のビットは、FDFビットとしてデコードされます。FDFリセッシブは、CAN FDフレームを示し、FDFドミナントはクラシックCANフレームを示します。

CAN FDフレームでは、FDFに続く2つのビットのresおよびBRSによって、このCAN FDフレーム内のビットレートを切り替えるかどうかが決まります。CAN FDビットレートの切り替えは、resドミナントとBRSリセッシブによって示されます。resリセッシブのコーディングは、プロトコルを今後拡張するために予約済みとなります。FDCANがFDFリセッシブおよびresリセッシブでフレームを受信した場合、ビットPSR.PXEをセットすることでプロトコル例外イベントの信号を送信します。プロトコル例外処理が有効である場合（CCCR.PXHD = 0）、動作状態が次のサンプルポイントでReceiver（PSR.ACT = 10）からIntegrating（PSR.ACT = 00）に変わります。プロトコル例外処理が無効である場合（CCCR.PXHD = 1）、FDCANはリセッシブのresビットをフォームエラーとして扱い、エラーフレームで応答します。

CAN FD動作は、CCCR.FDOEをプログラムすることで有効になります。CCCR.FDOE = 1の場合、CAN FDフレームの送受信は有効です。クラシックCANフレームの送受信はいつでも可能です。CAN FDフレームまたはクラシックCANフレームが送信されるかどうかは、それぞれのTxバッファ要素のビットFDF経由で設定できます。CCCR.FDOE = 0では、受信フレームはクラシックCANフレームとして解釈され、CAN FDフレーム受信時にエラーフレームを送信します。CAN FD動作が無効である場合、Txバッファ要素のビットFDFがセットされている場合でもCAN FDフレームは送信されません。CCCR.FDOEおよびCCCR.BRSEは、CCCR.INITおよびCCCR.CCEが両方セットされている間のみ変更できます。

CCCR.FDOE = 0では、ビットFDFおよびBRSの設定は無視され、フレームはクラシックCANフォーマットで送信されます。CCCR.FDOE = 1かつCCCR.BRSE = 0では、Txバッファ要素のビットFDFだけが評価されます。CCCR.FDOE = 1かつCCCR.BRSE = 1では、ビットレート切り替えでのCAN FDフレームの送信は有効です。ビットFDFおよびBRSがセットされたすべてのTxバッファ要素は、ビットレート切り替えしてCAN FDフォーマットで送信されます。

CAN動作中のモード変更は、次の条件下でのみ推奨されます。

- CAN FDデータフェーズでの失敗率は、CAN FDアービトレーションフェーズより大幅に高くなります。この場合、送信用のCAN FDビットレート切り替えオプションを無効にします。
- システム起動中、すべてのノードがCAN FDフォーマットで通信できることを検証するまでクラシックCANメッセージを送信します。これが当てはまる場合、すべてのノードはCAN FD動作に切り替わります。
- CANパーシャルネットワーキングのウェイクアップメッセージは、クラシックCANフォーマットで送信する必要があります。
- すべてのノードがCAN FDに対応していない場合のプログラミングの終了。非CAN FDノードは、プログラミングが完了するまでサイレントモードで保持されます。その後、すべてのノードがクラシックCAN通信に切り替わります。

FDCANフォーマットでは、DLCのコーディングは標準のCANフォーマットの1つとは異なります。DLCコード0～8は標準CANと同じコーディングで、コード9～15（標準CANではすべて8バイトのデータフィールドのコーディング）は表712に従ってコーディングされます。

表712.FDCANでのDLCコーディング

DLC	9	10	11	12	13	14	15
データバイト数	12	16	20	24	32	48	64

CAN FD高速フレームでは、BRS（ビットレート切り替え）ビットがリセッシブである場合、このビットの後のフレーム内でビットタイミングが切り替わります。FDCANアービトレーションフェーズでBRSビットまでは、ビットタイミングおよびプリスケーラレジスタBTPによって定義されたとおり

に、標準のCANビットタイミングが使用されます。次のFDCANデータフェーズでは、高速ビットタイミングおよびプリスケーラレジスタFBTPによって定義されたとおりに、高速CANビットタイミングが使用されます。CRCデリミタまたはエラー検出時にどちらが先に発生しても、ビットタイミングは高速タイミングから切り替わって戻ります。

CAN FDデータフェーズで設定可能な最大ビットレートは、FDCANカーネルクロック周波数に依存します。たとえば、20 MHzのFDCANカーネルクロック周波数で4タイムクォンタム（tq）の設定可能な最小ビット時間とした場合、データフェーズのビットレートは5Mbit/sとなります。

両方のデータフレームフォーマット（CAN FDロングフレームおよびCAN FD高速フレーム）で、ビットESI（エラーステータスインジケータ）の値は送信開始時のトランスミッタのエラー状態によって決まります。トランスミッタがエラーパッシブである場合、ESIはリセッシブで送信され、その他の場合はドミナントで送信されます。CAN FDリモートフレームでは、ESIビットはトランスミッタのエラー状態にかかわらず、常にドミナントで送信されます。CAN FDリモートフレームのデータ長コードは0で送信されます。

FDCAN TxバッファがDLC > 8でのFDCAN送信に設定されている場合、最初の8バイトはTxバッファで設定されたとおりに送信され、データフィールドの残りの部分は0xCCでパディングされます。FDCANがDLC > 8でFDCANフレームを受信する場合、フレームの最初の8バイトは適合するRx FIFOに格納されます。残りのバイトは破棄されます。

トランシーバ遅延補正

FDCAN送信のデータフェーズの間、1つのノードだけが送信して、その他すべてはレシーバになります。バスラインの長さに影響はありません。ピンFDCAN_TX経由で送信する場合、プロトコルコントローラはピンFDCAN_RX経由でローカルCANトランシーバから送信されたデータを受信します。受信されるデータは、CANトランシーバループ遅延により遅れます。この遅延がTSEG1（サンプルポイント前の時間セグメント）より大きくなる場合、ビットエラーが検出されます。トランシーバ遅延補正がない場合、FDCANフレームのデータフェーズのビットレートはトランシーバループ遅延によって制限されます。

FDCANは遅延補正メカニズムを実装して、CANトランシーバループ遅延を補正することで、特定のCANトランシーバの遅延にかかわらず、FDCANデータフェーズ中により高いビットレートでの送信を可能にします。

ノード送信のデータフェーズ中にビットエラーをチェックするために、遅延した送信データはセカンダリサンプルポイント（SSP）で受信したデータと比較されます。ビットエラーが検出される場合、トランスミッタは次のレギュラサンプルポイントでこのビットエラーに反応します。アービトレーションフェーズ中、遅延補正は常に無効です。

トランスミッタ遅延補正は、データビット時間がトランスミッタ遅延より短い場合に設定が可能になります。詳細は、新しいISO11898-1に記載されています。ビットDBTP.TDCをセットすることで有効になります。

受信したビットは、送信したビットとSSPで比較されます。SSPの位置は、トランシーバ経由でFDCAN の送信出力ピン FDCAN_TX から受信入力ピン FDCAN_RX までに測定された遅延と、TDCR.TDCOによって設定されたトランスミッタ遅延補正オフセットの合計として定義されます。トランスミッタ遅延補正オフセットは、受信したビット内のSSPの位置を調整するために使用します（例：データフェーズのビット時間の半分）。セカンダリサンプルポイントの位置は、次のmtq（fdcan_tq_ckクロックの1周期である最小タイムクォンタム）の整数に切り捨てられます。

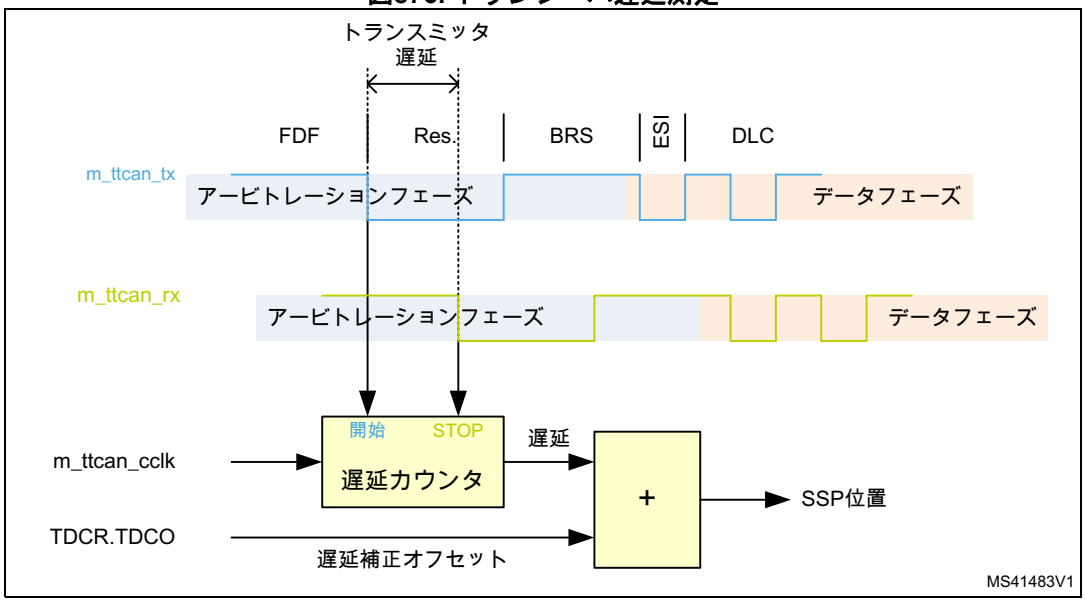
PSR.TDCVは、実際のトランスミッタ遅延補正値を示します。PSR.TDCVは、CCCR.INITがセットされているとクリアされ、DBTP.TDCがセットされている間FDフレームの各送信で更新されます。

FDCANに実装されたトランスミッタ遅延補正では、次の境界条件を考慮する必要があります。

- FDCAN_TxからFDCAN_Rxまでの測定された遅延と、設定されたトランスミッタ遅延補正オフセットTDCR.TDCOの合計は、データフェーズで6bit時間未満でなければなりません。
- FDCAN_TXからFDCAN_RXまでの測定された遅延と、設定されたトランスミッタ遅延補正オフセットTDCR.TDCOの合計は、127 mtq以下でなければなりません。合計がこの値を超える場合、最大値（127 mtq）はトランスミッタ遅延補正に使用されます。
- データフェーズは、CRCデリミタのサンプルポイントで終了し、SSPで受信したビットのチェックを停止します。

トランスミッタ遅延補正が、DBTP.TDC = 1をプログラムすることで有効になった場合、ビットFDFからビットresの立下りエッジで送信した各CAN FDフレーム内で測定が開始されます。このエッジがトランスミッタの受信入力ピンFDCAN_TXで見られると測定は停止します。この測定の解析時間は、1 mtqです。

図876. トランシーバ遅延測定



受信したFDFビット内のドミナントグリッチによって、受信したresビットの立ち下がりエッジ前に遅延補正測定を終了する（早いSSPの位置となる）ことを避けるために、TDCR.TDCFをプログラムすることでトランスミッタ遅延補正フィルタウィンドウの使用が有効になります。これでSSPの位置の最小値を定義します。より早いSSPの位置につながるFDCAN_RXのドミナントエッジは、トランスミッタ遅延測定では無視されます。SSPの位置が少なくともTDCR.TDCFで、FDCAN_RXがローになると、測定は停止します。

制限動作モード

制限動作モードでは、ノードはデータとリモートフレームを受信でき、有効なフレームに確認応答を付与できますが、データフレーム、リモートフレーム、アクティブエラーフレーム、またはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待ってCAN通信と再同期します。エラーカウンタ（ECR.REC、ECR.TEC）は、エラーログ記録（ECR.CEL）がアクティブになっている間は停止します。ソフトウェアは、ビットCCCR.ASMをセットすることでFDCANを制限動作モードにセットできます。このビットは、CCCR.CCEおよびCCCR.INITの両方が1にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってクリアできます。

Txハンドラがメッセージ RAMから時間内にデータを読み出せなかったとき、自動的に制限動作モードに移行します。制限動作モードを終了するには、ソフトウェアでCCCR.ASMをリセットする必要があります。

制限動作モードは、異なるCANビットレートに適応できるアプリケーションで使用できます。この場合、アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。

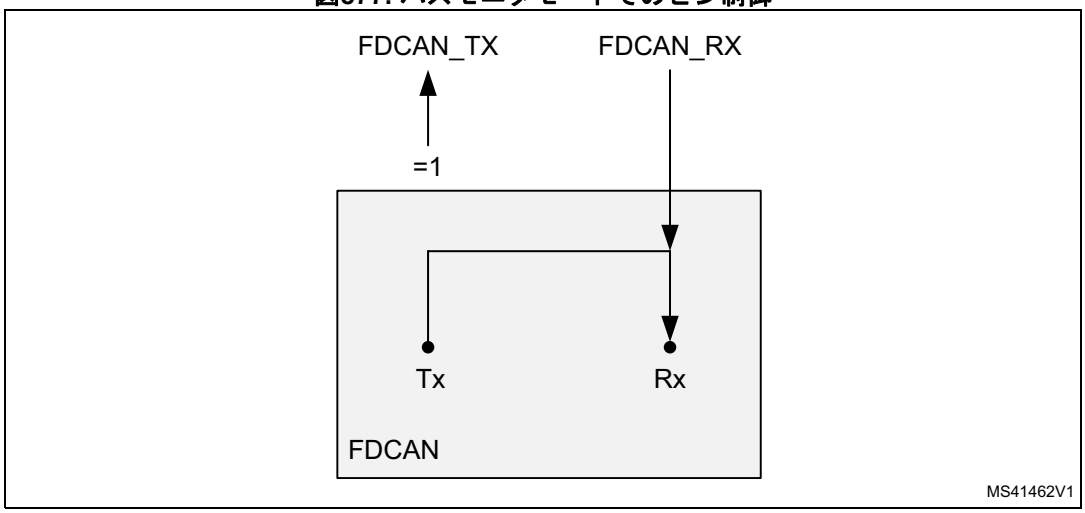
注： **制限動作モードは、ループバックモード（内部または外部）と組み合わせないでください。**

バスモニタモード

FDCANは、CCCR.MONビットをセットすることでバスモニタモードにセットされます。バスモニタモード（詳細はISO11898-1、10.12 Bus monitoringを参照してください）では、FDCANは有効なデータフレームとリモートフレームを受信できますが、送信は開始できません。このモードでは、CANバスでリセッシブビットのみを送信します。FDCANがドミナントビット（ACKビット、オーバーロードフラグ、アクティブエラーフラグ）を送信する必要がある場合、CANバスがリセッシブ状態にとどまることがあってもFDCANがこのドミナントビットを監視できるように、このビットは内部でルート変更されます。バスモニタモードでは、TXBRPレジスタがリセット状態のまま保持されます。

バスモニタモードを使用すると、ドミナントビットの送信による影響を受けずに、CANバス上のトラフィックを解析できます。図877では、バスモニタモードでのFDCANへのFDCAN_TXおよびFDCAN_RX信号の接続を示します。

図877. バスモニタモードでのピン制御



自動再送信モード無効（DAR）

CAN仕様（ISO11898-1、6.3.3 Recovery Management）に従い、FDCANはアービトレーションを喪失したフレームや送信中にエラーで中断されたフレームを自動的に再送信する方法を提供します。デフォルトでは自動再送信は有効化されています。

DARモードでのフレーム送信

DARモードでは、すべての送信がCANバスで開始した後で自動的にキャンセルされます。Txバッファ Txリクエストペンディングビット TXBRP.TRPxは、転送が成功した後、転送がまだ始まっていない時点でのキャンセル時、ロストアービトレーションによるアボート時、もしくはフレーム転送中にエラーが発生した時にリセットされます。

- 送信成功
  - 対応するTxバッファ送信発生ビットTXBTO[TOx] をセット
  - 対応するTxバッファキャンセル終了ビットTXBCF[CFx] はセットされない
- キャンセル時に送信成功
  - 対応するTxバッファ送信発生ビットTXBTO[TOx] をセット
  - 対応するTxバッファキャンセル終了ビットTXBCF[CFx] をセット
- アービトレーション喪失またはフレーム送信中断
  - 対応するTxバッファ送信発生ビットTXBTO[TOx] はセットされない
  - 対応するTxバッファキャンセル終了ビットTXBCF[CFx] をセット

フレーム送信が成功した場合、かつTxイベントのストレージが有効になっている場合、TxイベントFIFO要素はイベントタイプET = 10（キャンセル時送信）で書き込まれます。

パワーダウン（SLEEPモード）

FDCANは、CC制御レジスタCCCR[CSR] 経由のクロック停止リクエスト入力によって制御されるパワーダウンモードにセットできます。クロック停止リクエストがアクティブである限り、ビットCCCR[CSR] は1で読み出されます。

すべてのペンディングされた送信リクエストが完了すると、FDCANはバスアイドル状態が検出されるまで待ちます。その後、FDCANはCCCR[INIT] を1にセットして、追加のCAN転送を防ぎます。FDCANは、CCCR[CSA] を1にセットすることでパワーダウンの準備が整ったことを確認します。この状態では、クロックがオフに切り替わる前に、追加のレジスタアクセスが可能です。CCCR[INIT] への書込みアクセスは効果がありません。これで、モジュールクロック入力はオフに切り替えられます。

パワーダウンモードを終了するには、アプリケーションがCC制御レジスタフラグCCCR.CSRをリセットする前にモジュールクロックをオンにしなければなりません。FDCANは、CCCR[CSA] をリセットすることでこれを確認します。その後、アプリケーションは、ビットCCCR[INIT] をリセットすることでCAN通信を再開できます。

テストモード

FDCANテストレジスタ（FDCAN_TEST）への書込みアクセスを有効にするには、ビットCCCR.TESTを1にセットして、テストモードおよび機能の設定を有効にしなければなりません。

TEST.TXをプログラムすることで、CAN送信ピンFDCAN_TXで4つの出力機能が使用できます。デフォルト機能（シリアルデータ出力）に加えて、CANサンプルポイント信号を駆動してFDCANビットタイミングを監視し、一定のドミナントまたはリセッシブ値を駆動できます。ピンFDCAN_RXでの実際の値は、TEST.RXから読み出せます。両方の機能を使用して、CANバス物理層をチェックできます。

CANカーネルクロックとAPBクロックドメインとの同期メカニズムによって、新しい設定がFDCAN_TX出力ピンで確認できるようになるまで、TEST.TXへの書込み間でAPBクロック周期数周期分の遅延が発生する場合があります。これは、TEST.RX経由でFDCAN_RX入力ピンを読み出すときにも適用されます。

注： **テストモードは、製造テストまたは自己診断機能にのみ使用する必要があります。FDCAN_TXピンのソフトウェア制御は、すべてのCANプロトコル機能に干渉します。アプリケーションにテストモードを使用することは推奨されません。**

外部ループバックモード

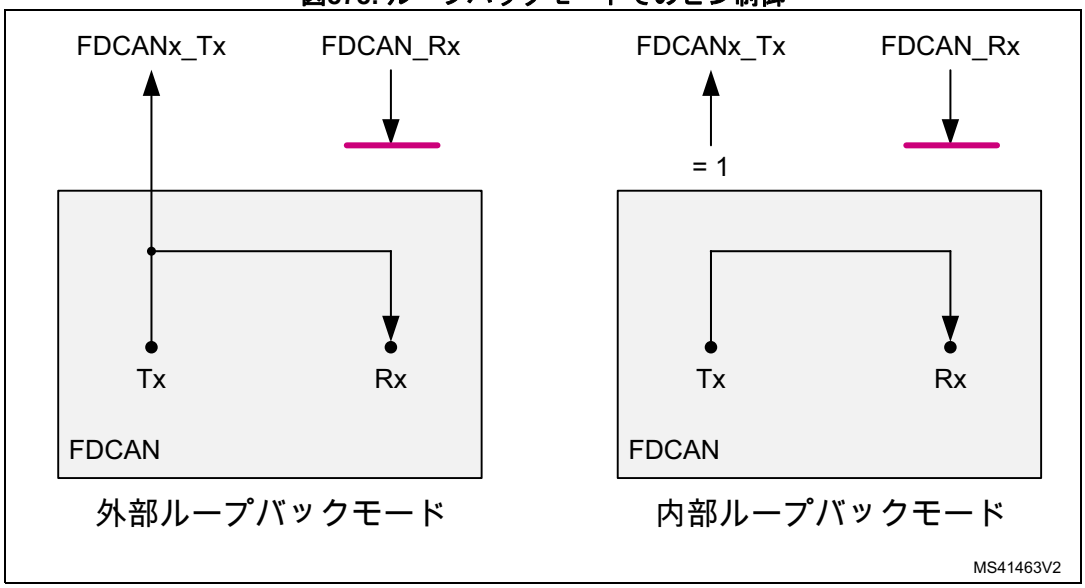
FDCANを外部ループバックモードに設定するには、TEST.LBCKを1にセットします。ループバックモードでは、FDCANはそれ自体が送信したメッセージを受信メッセージとして扱い、それらをRx FIFOに格納します（受入フィルタを通過した場合）。図878では、外部ループバックモードでのFDCANへの送受信信号FDCAN_TXおよびFDCAN_RXの接続を示します。

このモードは、ハードウェアの自己診断機能のために用意されています。外部通信から独立させるために、ループバックモードでは、FDCANはACKエラーを無視します（データ／リモートフレームのACKスロットではリセッシブビットがサンプリングされます）。このモードでは、FDCANは送信出力から受信入力へ内部フィードバックを行います。FDCANは、FDCAN_RX入力ピンの実際の値を無視します。送信メッセージはFDCAN_TX送信ピンで監視できます。

内部ループバックモード

TEST.LBCKとCCCR.MONを1にプログラムすることによって、内部ループバックモードに移行します。このモードは「ホットセルフテスト」に使用できます。つまり、FDCAN_TXおよびFDCAN_RXピンに接続されている実行中のCANシステムに影響を与えずに、FDCANをテストできます。このモードでは、FDCAN_RXピンはFDCANから切断され、FDCAN_TXピンはリセッシブに保持されます。図878では、内部ループバックモードの場合のFDCANへのFDCAN_TXおよびFDCAN_RXピンの接続を示します。

図878. ループバックモードでのピン制御



タイムスタンプの生成

タイムスタンプの生成について、FDCANは16bitラップアラウンドカウンタを提供します。プリスケーラTSCC.TCPを設定して、CANビット時間の倍数（1～16）でカウンタにクロック供給できます。カウンタは、TSCV[TCV] で読み出せます。レジスタTSCVへの書込みアクセスによって、カウンタは0にリセットされます。タイムスタンプカウンタが0に戻ると、割込みフラグIR[TSW] がセットされます。

フレーム受信／送信の開始時、カウンタ値がキャプチャされ、Rx FIFO（RXTS[15:0]）またはTxイベントFIFO（TXTS[15:0]）要素のタイムスタンプセクションに格納されます。

ビットTSCC.TSSをプログラムすることで、16bitタイムスタンプを使用できます。

デバッグモードの動作

デバッグモードでは、読出し機能のセット／リセットはデバッガレジスタアクセス中に自動的に無効になり、通常のMCU動作中に有効になります。

タイムアウトカウンタ

Rx FIFO 0、Rx FIFO 1、およびTxイベントFIFOのタイムアウト条件の信号を送信するために、FDCANは16-bitタイムアウトカウンタを提供します。ダウンカウンタとして動作し、TSCC[TCP] で制御される同じプリスケーラをタイムスタンプカウンタとして使用します。タイムアウトカウンタは、レジスタTOCC経由で設定されます。実際のカウンタ値は、TOCV[TOC] から読み出せます。タイムアウトカウンタはCCCR[INIT] = 0の間だけ開始できます。FDCANがBus_Off状態に移行したときなどCCCR[INIT] = 1の場合は停止します。

動作モードはTOCC[TOS] で選択されます。連続モードで動作時、CCCR[INIT] がリセットされるとカウンタが開始します。TOCVへの書込みによって、カウンタはTOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。

タイムアウトカウンタがFIFOのうち1つで制御されている場合、空のFIFOはTOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初のFIFO要素が格納されたときに開始します。TOCVへの書込みは無効です。

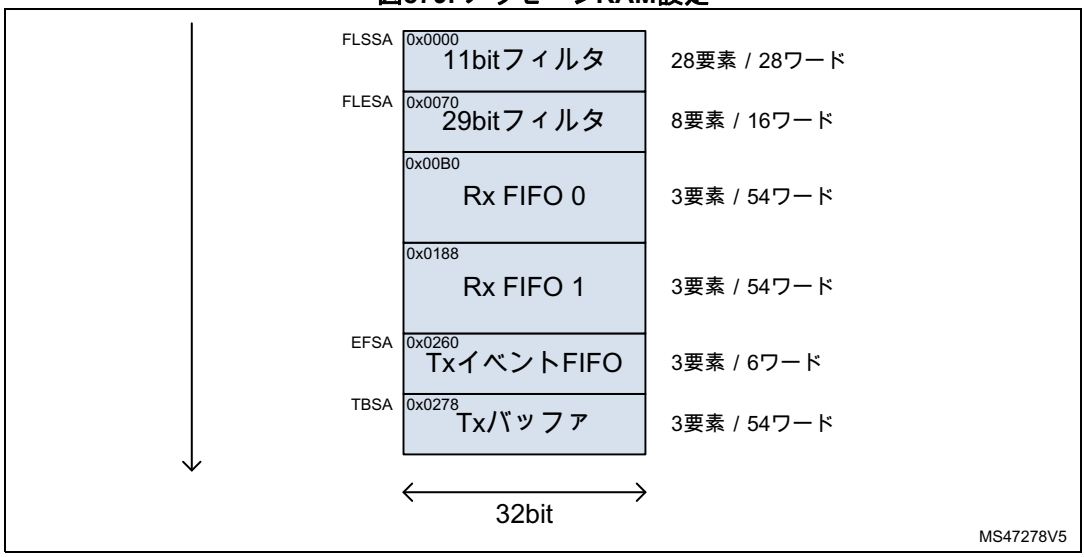
カウンタが0に達すると、割込みフラグIR[TOO] がセットされます。連続モードでは、カウンタはTOCC[TOP] ですぐに再開されます。

注： **タイムアウトカウンタのクロック信号は、CANコアサンプルポイント信号から生成されます。そのため、タイムアウトカウンタがデクリメントされる時間内のポイントは、CANコアの同期／再同期メカニズムによって異なる場合があります。FDCANのボーレートスイッチ機能が使用されている場合、タイムアウトカウンタはアービトレーションおよびデータフィールドで別にクロック供給されます。**

70.3.3 メッセージRAM

メッセージRAMは32bit幅で、これに最大212ワードを割り当てるためにFDCANモジュールが設定されます。図879に示された各セクションを設定する必要はありません。

図879. メッセージRAM設定



FDCANがメッセージRAMのアドレスを指定する場合、シングルバイトではなく32bitワード（整列）を指定します。RAMアドレスは32bitワードで、ビット15～2だけが評価され、2つの最下位ビットは無視されます。

インスタンスが複数ある場合、FDCANnのRAM開始アドレスはFDCANn-1の終了アドレス + 4で計算され、FDCANnの終了アドレスはFDCANn開始アドレス + 0x0350 - 4で計算されます。

たとえば、次の2つのインスタンスの場合：

- FDCAN1：
  - 開始アドレス0x0000
  - 終了アドレス0x034C（図879を参照）
- FDCAN2：
  - 開始アドレス = 0x034C（FDCAN1終了アドレス）+ 4 = 0x0350
  - 終了アドレス = 0x0350（FDCAN2開始アドレス）+ 0x0350 - 4 = 0x069C

Rxの処理

Rxハンドラは、受入フィルタ、2つのRx FIFOのうちの1つへの受信メッセージの転送、およびRx FIFOのPutとGetのインデックスを制御します。

受入フィルタ

FDCANは、2セットの受入フィルタを設定でき、1つは標準識別子用で、もう1つは拡張識別子用です。これらのフィルタは、Rx FIFO 0またはRx FIFO 1に割り当てられます。受入フィルタでは、フィルタの各リストが要素 #0から最初に一致する要素まで実行されます。受入フィルタは最初に一致する要素で停止します。次のフィルタ要素はこのメッセージでは評価されません。

主な機能：

- 各フィルタ要素は次のように設定できます。
  - 範囲フィルタ（～から～まで）
  - 1つまたは2つの専用IDのフィルタ
  - クラシックビットマスクフィルタ
- 各フィルタ要素は、受入フィルタまたは拒否フィルタで設定できます。
- 各フィルタ要素は個別に有効化／無効化できます。
- フィルタは連続でチェックされ、最初の一致フィルタ要素で実行を停止します。

関連する設定レジスタ：

- グローバルフィルタ設定（RXGFC）
- 拡張ID ANDマスク（XIDAM）

フィルタ要素の設定（SFEC/EFEC）に応じて、一致すると次の操作のいずれかがトリガされます。

- FIFO 0またはFIFO 1に受信フレームを格納
- 受信フレームを拒否
- 高優先メッセージ割込みフラグIR[HPM] のセット
- 高優先メッセージ割込みフラグIR[HPM] をセットし、FIFO 0またはFIFO 1に受信フレームを格納

受入フィルタは、完了識別子を受信した後に開始されます。受入フィルタが完了した後、一致するRx FIFOが見つかった場合、メッセージハンドラが32bitの単位で受信したメッセージデータを一致するRx FIFOに書込み始めます。CANプロトコルコントローラがエラー状態（CRCエラーなど）を検出した場合、このメッセージは破棄され、次の影響が生じます。

- **Rx FIFO**
  一致するRx FIFOの プット インデックスは更新されませんが、関連するFIFO要素（一部）が受信したデータで上書きされます。エラータイプについては、PSR.LECおよびPSR.DLECを参照してください。一致するRx FIFOが上書きモードで動作している場合、Rx FIFO上書きモードに記載された境界条件を考慮する必要があります。

注： **許可したメッセージが2つのRx FIFOのうちの1つに書き込まれると、変更されていない受信IDは、使用されるフィルタとは別に格納されます。受入フィルタプロセスの結果は、設定されたフィルタ要素のシーケンスに強く依存します。**

範囲フィルタ

フィルタがSF1ID/SF2IDおよびEF1ID/EF2IDで定義された範囲のメッセージIDを持つすべての受信フレームに一致します。

範囲フィルタを拡張フレームと使用する場合、2つのことが可能です。

- EFT = 00：範囲フィルタが適用される前に、受信フレームのメッセージIDは拡張ID ANDマスク（XIDAM）とAND演算されます。
- EFT = 11：拡張ID ANDマスク（XIDAM）は、範囲フィルタに使用されません。

専用IDのフィルタ

フィルタ要素を設定して、1つまたは2つの固有IDのフィルタでフィルタできます。1つの固有メッセージIDでフィルタするには、フィルタをSF1ID = SF2IDおよびEF1ID = EF2IDで設定する必要があります。

クラシックビットマスクフィルタ

クラシックビットマスクフィルタは、受信したメッセージIDのシングルビットをマスクして、メッセージIDのグループをフィルタすることを目的としています。クラシックビットマスクフィルタで、SF1ID/EF1IDはメッセージIDフィルタとして使用され、SF2ID/EF2IDはフィルタマスクとして使用されます。

フィルタマスクでの0bitは、設定されたIDフィルタの対応するビット位置をマスクします。たとえば、そのビット位置で受信したメッセージIDは受入フィルタの影響を受けません。対応するマスクビットが1の場合、受信したメッセージIDのそれらのビットだけが受入フィルタの影響を受けます。

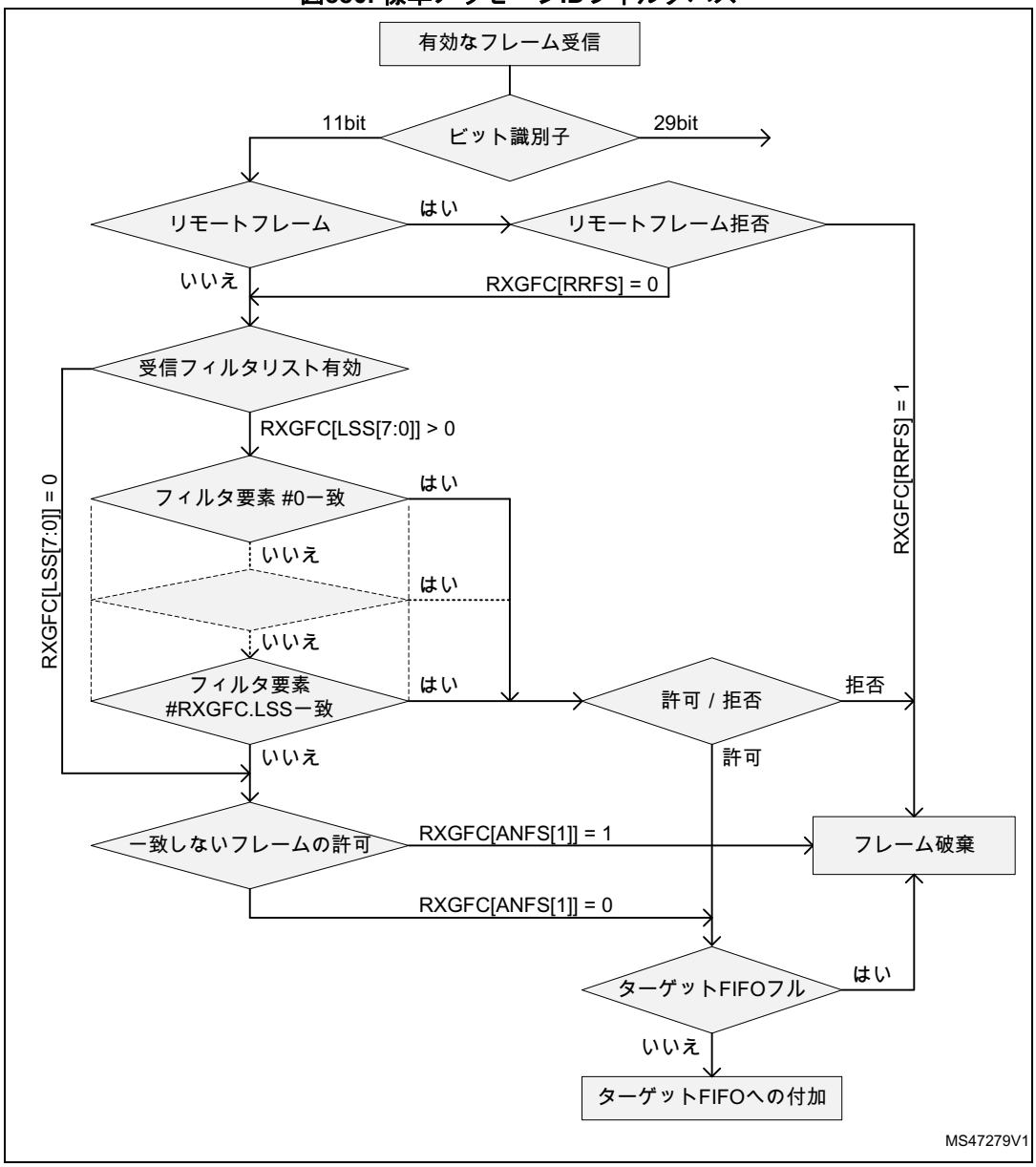
すべてのマスクビットが1の場合、受信したメッセージIDとメッセージIDフィルタが同一である場合のみ一致が発生します。すべてのマスクビットが0の場合、すべてのメッセージIDが一致します。

標準メッセージIDフィルタ

図880では、標準メッセージID（11-bit識別子）フィルタのフローを示します。標準メッセージIDフィルタ要素は、セクション70.3.8を参照してください。

グローバルフィルタ設定（RXGFC）メッセージIDで制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）は設定されたフィルタ要素のリストと比較されます。

図880. 標準メッセージIDフィルタパス

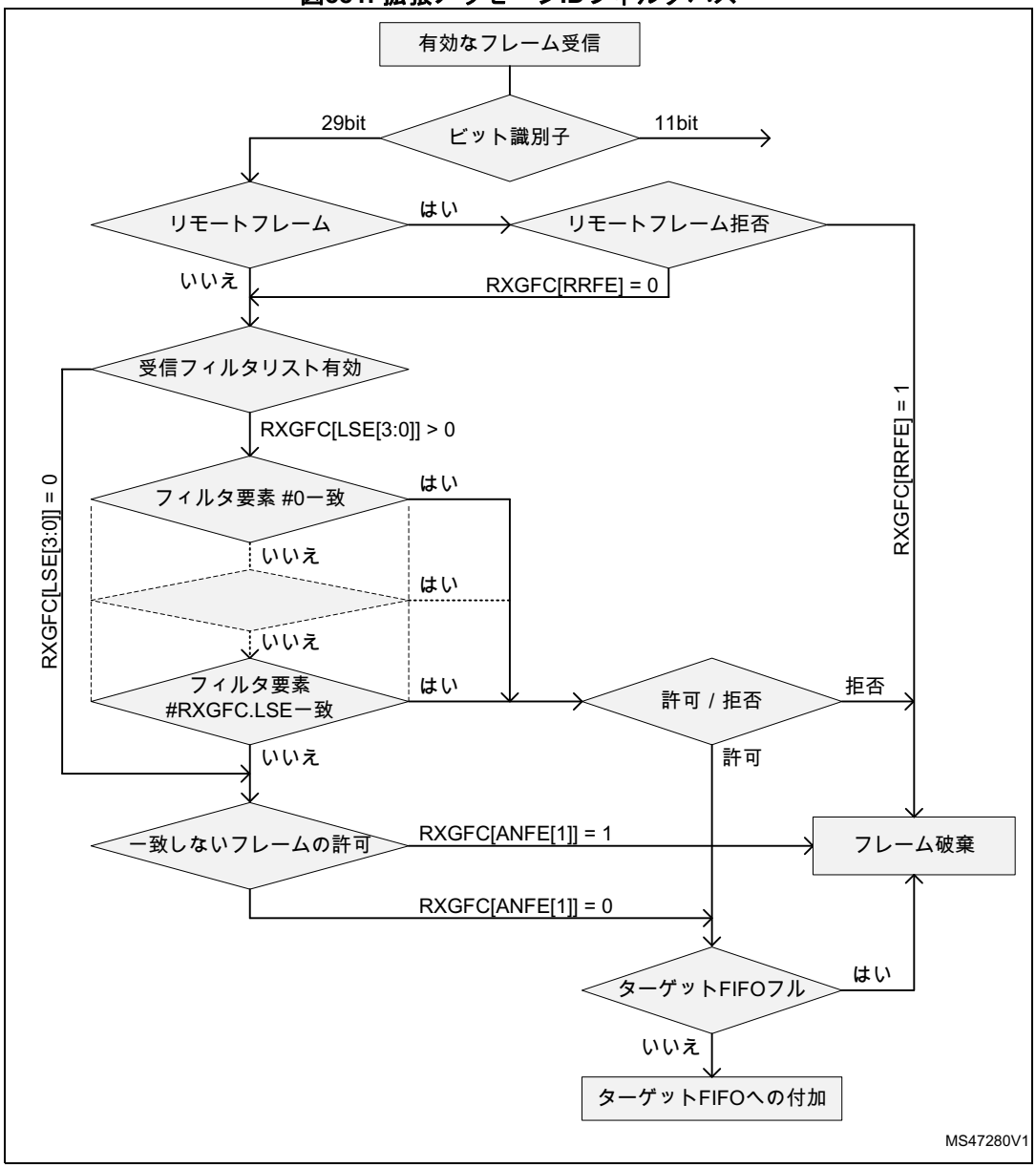


拡張メッセージIDフィルタ

図881では、拡張メッセージID（29-bit識別子）フィルタのフローを示します。拡張メッセージIDフィルタ要素は、セクション70.3.9を参照してください。

グローバルフィルタ設定RXGFCおよび拡張IDフィルタ設定RXGFCメッセージIDで制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）が設定されたフィルタ要素のリストと比較されます。

図881. 拡張メッセージIDフィルタパス



フィルタリストが実行される前に、拡張ID ANDマスク（XIDAM）は受信識別子とANDでつながれます。

Rx FIFO

Rx FIFO 0およびRx FIFO 1は、それぞれ最大3個の要素を保持できます。

受入フィルタを通過した受信メッセージは、一致フィルタ要素で設定されたとおりにRx FIFOに転送されます。Rx FIFO 0およびRx FIFO 1で使用できるフィルタメカニズムの詳細については、受入フィルタを参照してください。Rx FIFO要素は、セクション70.3.5を参照してください。

Rx FIFOフル条件がIR[RFnF] によって信号送信されると、最低1つのメッセージが読み出され、Rx FIFO Getインデックスがインクリメントされるまで、対応するRx FIFOに追加のメッセージは書き込まれません。対応するRx FIFOがフルになっているときにメッセージを受信した場合、このメッセージは破棄され、割込みフラグIR[RFnL] がセットされます。

Rx FIFOから読み出すときに、Rx FIFO GetインデックスRXFnS[FnGI] + FIFO要素サイズを対応するRx FIFO開始アドレス [FnSA] に追加する必要があります。

Rx FIFOブロックモード

Rx FIFOブロックモードは、RXGFC.FnOM = 0で設定されます。これは、Rx FIFOのデフォルトの動作モードです。

Rx FIFOフル条件（RXFnS.FnPI = RXFnS.FnGI）に達すると、最低1つのメッセージが読み出され、Rx FIFO Getインデックスがインクリメントされるまで、対応するRx FIFOに追加のメッセージは書き込まれません。Rx FIFOフル条件は、RXFnS.FnF = 1で信号送信されます。さらに、割込みフラグIR.RFnFがセットされます。

対応するRx FIFOがフルになっているときにメッセージを受信した場合、このメッセージは破棄され、メッセージ喪失条件がRXFnS.RFnL = 1で信号送信されます。さらに、割込みフラグIR.RFnLがセットされます。

Rx FIFO上書きモード

Rx FIFO上書きモードは、RXGFC.FnOM = 1で設定されます。

Rx FIFOフル条件（RXFnS.FnPI = RXFnS.FnGI）がRXFnS.FnF = 1によって信号送信されると、FIFOに許可された次のメッセージは一番古いFIFOメッセージを上書きします。PutおよびGetインデックスは両方1インクリメントされます。

Rx FIFOが上書きモードで動作し、Rx FIFOフル条件の信号を送信したとき、Rx FIFO要素の読出しは最低Getインデックス + 1で開始する必要があります。これは、CPUがメッセージRAM（Getインデックス）から読み出しつつ、受信メッセージがメッセージRAM（Putインデックス）に書き込まれる状況になり得るためです。この場合、一貫していないデータがそれぞれのRx FIFO要素から読み出される場合があります。Rx FIFOから読出し時に 取得インデックスにオフセットを加えることで、この問題を回避しています。このオフセットは、CPUがRx FIFOにアクセスする速度に依存します。

Rx FIFOから読み出した後、最後の要素読出しの数をRx FIFO確認応答インデックスRXFnA.FnAに書き込む必要があります。これにより、Getインデックスがその要素数にインクリメントされます。プット インデックスがこのRx FIFO要素にインクリメントされなかった場合、Rx FIFOフル条件はリセットされます（RXFnS.FnF = 0）。

Txの処理

Txハンドラは、Tx FIFOおよびTxキューの送信リクエストを処理します。CANコア、PutおよびGetインデックス、TxイベントFIFOへの送信メッセージの転送を制御します。最大3個のTxバッファをメッセージ送信にセットアップできます。CAN メッセージデータフィールドは64バイトに設定され、Tx FIFOにはTx要素の格納用に18個の32bitワードが割当てられます。

表713.フレーム送信の可能な設定

CCCR		Txバッファ要素		フレーム送信
BRSE	FDOE	FDF	BRS	
無視	0	無視	無視	クラシックCAN
0	1	0	無視	クラシックCAN
0	1	1	無視	ビットレート切り替えなしのFD
1	1	0	無視	クラシックCAN
1	1	1	0	ビットレート切り替えなしのFD
1	1	1	1	ビットレート切り替えありのFD

注： **AUTOSARは最低3つのTxキューバッファと送信キャンセルのサポートが必須です。**

Txハンドラは、Txバッファリクエストペンディングレジスタ TXBRPが更新されたときや、送信が開始されたときに、Txスキャンを開始して、最優先のペンディング中のTxリクエスト（一番小さいメッセージIDのTxバッファ）をチェックします。

送信一時停止

送信一時停止機能は、CANメッセージ識別子が（完全に）固有の値に指定され、容易に変更できないCANシステムで使用することを目的とします。これらのメッセージ識別子は、他の定義されたメッセージより高いCANアービトレーション優先度である場合がありますが、個別のアプリケーションでは関連するアービトレーション優先度が逆でなければなりません。これにより、他のメッセージが低いCANアービトレーション優先度であるため、あるECUが一気にCANメッセージを送信して別のECU CANメッセージを遅延させる場合があります。

たとえば、CAN ECU-1が機能を有効にして、アプリケーションソフトウェアによって4件のメッセージを送信するようリクエストされた場合、最初のメッセージ送信が成功した後、次のリクエストされたメッセージの開始が許可される前に2 CANビット時間のバスアイドルを待ちます。ペンディング中のメッセージを保持する他のECUがある場合、それらのメッセージはアイドル時間で開始され、ECU-1の次のメッセージとアービトレーションする必要はありません。メッセージを受信した後、受信メッセージがCANバスを開放すると、ECU-1は次の送信開始が許可されます。

この機能は、CCCR レジスタのTXP ビットで制御されます。このビットがセットされている場合、FDCANがメッセージの送信に成功するたびに、次の送信開始前の2 CANビット時間、一時停止します。これにより、ネットワーク内の他のCANノードのメッセージが低い優先度識別子を持つものであってもメッセージを送信できます。デフォルトでは無効です（CCCR.TXP = 0）。

この機能は、1つのノードからのバースト送信を緩和し、アプリケーションプログラムが誤って大量の送信をリクエストする「バブリングイディオット」シナリオから保護します。

Tx FIFO

Tx FIFO動作は、TXBC[TFQM] を0にプログラムすることによって設定されます。Tx FIFOに格納されたメッセージは、GetインデックスTXFQS[TFGI] によって参照されるメッセージから送信されます。各送信後、GetインデックスはTx FIFOがエンプティになるまで周期的にインクリメントされます。Tx FIFOは、異なるTxバッファから同じメッセージIDを持つメッセージを、これらのメッセージがTx FIFOに書き込まれた順番で送信できるようにします。FDCANは、GetインデックスとPutインデックスの差としてTx FIFOのフリーレベルTXFQS[TFFL] を計算します。これは、使用可能な（空き）Tx FIFO要素の数を示します。

新しい送信メッセージは、PutインデックスTXFQS[TFQPI] によって参照されるTxバッファから始まるTx FIFOに書き込まれる必要があります。追加リクエストは、次の空きTx FIFO要素にPutインデッ

クスをインクリメントします。PutインデックスがGetインデックスに達すると、Tx FIFOフル（TXFQS[TFQF]= 1）の信号が送信されます。この場合、次のメッセージが送信され、Getインデックスがインクリメントされるまで、Tx FIFOに追加のメッセージを書き込まないでください。

1つのメッセージがTx FIFOに追加される場合、Tx FIFO Putインデックスによって参照されるTxバッファに関連したTXBARビットに1を書き込むことで送信がリクエストされます。

複数（n）件のメッセージがTx FIFOに追加される場合、Putインデックスから始まるn個の連続したTxバッファに書き込まれます。次に送信がTXBAR経由でリクエストされます。その後、Putインデックスは周期的にnずつインクリメントされます。リクエストされるTxバッファの数は、Tx FIFOフリーレベルで示された空きのTxバッファの数を超えてはなりません。

Getインデックスによって参照されるTxバッファに対する送信リクエストがキャンセルされた場合、Getインデックスは送信リクエストをペンディングして次のTxバッファにインクリメントされ、Tx FIFOフリーレベルが再計算されます。送信キャンセルが他のあらゆるTxバッファに適用される場合、GetインデックスとFIFOフリーレベルに変化はありません。

Tx FIFO要素はメッセージRAMに18個の32bitワードを割り当てます。そのため、次に使用可能な（空き）Tx FIFOバッファの開始アドレスは、PutインデックスTXFQS[TFQPI]（0～2）の4倍をTxバッファ開始アドレス [TBSA] に追加することで計算されます。

Txキュー

Txキュー動作は、TXBC[TFQM] を1にプログラムすることによって設定されます。Txキューに格納されたメッセージは、一番小さいメッセージID（最優先）のメッセージから送信されます。

標準メッセージIDと拡張メッセージIDが混在している場合、標準メッセージIDは拡張メッセージIDのビット [28:18] と比較されます。

複数のキューバッファが同じメッセージIDで設定される場合、一番小さいバッファ数のキューバッファが先に送信されます。

新しいメッセージは、PutインデックスTXFQS[TFQPI] によって参照されるTxバッファに書き込まれる必要があります。追加リクエストは、次の空きTxバッファにPutインデックスをインクリメントします。Txキューがフル（TXFQS[TFQF]= 1）の場合、Putインデックスは有効ではなく、リクエストされるメッセージのうち最低1件が送信されるか、ペンディング中の送信リクエストがキャンセルされるまで、追加のメッセージをTxキューに書き込まないでください。

アプリケーションはPutインデックスの代わりにレジスタTXBRPを使用し、送信リクエストをペンディングせずにTxバッファにメッセージを配置する場合があります。

Txキューバッファは、メッセージRAMに18個の32bitワードを割り当てます。そのため、次に使用可能な（空き）Txキューバッファの開始アドレスは、Txキュー PutインデックスTXFQS[TFQPI]（0～2）の4倍をTxバッファ開始アドレスTBSAに追加することで計算されます。

送信キャンセル

FDCANは送信キャンセルをサポートします。Txキューバッファからリクエストされた送信をキャンセルするには、ホストはレジスタTXBCRの対応するビット位置（= Txバッファの数）に1を書き込む必要があります。送信キャンセルはTx FIFO動作を対象としません。

成功したキャンセルは、レジスタTXBCFの対応するビットを1にセットすることで信号送信されます。

Txバッファからの送信がすでに進行中で、送信キャンセルがリクエストされた場合、対応するTXBRPビットは送信が進行している限りセットされたままとなります。送信に成功した場合、対応するTXBTOおよびTXBCFビットがセットされます。送信に成功しなかった場合、繰り返されず、対応するTXBCFビットのみセットされます。

注： **ペンディング中の送信が開始直前にキャンセルされた場合、別のメッセージがこのノードでペンディングされている場合でも送信が開始されない短いタイムウィンドウが続きます。これにより、別のノードはそのノードの2番目の優先度のメッセージより低い優先度のメッセージを送信できます。**

Txイベントの処理

Txイベントの処理をサポートするために、FDCANはTxイベントFIFOを実装しました。FDCANがCANバス上にメッセージを送信すると、メッセージIDおよびタイムスタンプがTxイベントFIFO要素に格納されます。TxイベントをTxイベントFIFO要素にリンクするために、送信されたTxバッファのメッセージマーカーがTxイベントFIFO要素にコピーされます。

TxイベントFIFOは、最大3個の要素に設定できます。TxイベントFIFO要素は、Tx FIFOを参照してください。

TxイベントFIFOの目的は、Txバッファが送信すべきメッセージのみを保持し、送信ステータスはTxイベントFIFOに別に格納されるなど、送信ステータス情報の処理を送信メッセージ処理から切り離すことです。これには、特に動的に管理される送信キューを操作する際に、Txバッファが送信成功直後に新しいメッセージに使用できるという利点があります。Txバッファを上書きする前に、そのTxバッファからの送信ステータス情報を保存する必要はありません。

TxイベントFIFOフル条件がIR[TEFF] によって信号送信されると、最低1つの要素が読み出され、TxイベントFIFO Getインデックスがインクリメントされるまで、TxイベントFIFOに追加の要素は書き込まれません。TxイベントFIFOがフルになっているときにTxイベントが発生した場合、このイベントは破棄され、割込みフラグIR[TEFL] がセットされます。

TxイベントFIFOから読み出すときに、2倍のTxイベントFIFO GetインデックスTXEFS[EFGI] をTxイベントFIFO開始アドレスEFSAに追加する必要があります。

70.3.4 FIFO確認応答の処理

Rx FIFO 0、Rx FIFO 1、およびTxイベントFIFOのGetインデックスは、対応するFIFO確認応答インデックスへの書込みによって制御されます。セクション70.4.23およびセクション70.4.25を参照してください。FIFO確認応答インデックスへの書込みはFIFO GetインデックスをFIFO確認応答インデックス +1にセットし、FIFOフィルレベルを更新します。これには、2つの使用例があります。

1. 要素が1つだけFIFO（Getインデックスによって指定されているもの）から読み出された場合、このGetインデックス値はFIFO確認応答インデックスに書き込まれます。
2. 一連の要素がFIFOから読み出された場合、読出しシーケンスの最後に1回だけFIFO確認応答インデックス（値：最後の要素読出しのインデックス）を書き込んで、FIFO Getインデックスを更新するだけで十分です。

CPUがFDCANメッセージRAMに自由にアクセスできるため、任意の（Getインデックスが考慮されない）順番でFIFO要素を読み出す際に特別な注意が必要です。これは、2つのRx FIFOのうち1つから高優先メッセージを読み出すときに役立つ場合があります。この場合、Getインデックスが誤った位置にセットされ、FIFOフィルレベルも変わるため、FIFO確認応答インデックスへの書込みは行わないでください。この場合、古いほうのFIFO要素の一部は失われます。

注： **アプリケーションは、有効な値がFIFO確認応答インデックスに書き込まれていることを確認する必要があります。FDCANは誤った値をチェックしません。**

70.3.5 FDCAN Rx FIFO要素

2つのRx FIFOがメッセージRAMに設定されています。それぞれのRx FIFOセクションは、最大3個の受信メッセージを格納するように設定できます。Rx FIFO要素の構造は、表714に示され、表715に説明が示されています。

表714.Rx FIFO要素

ビット	31			24	23			16	15	8	7	0
R0	ESI	XTD	RTR	ID[28:0]								
R1	ANMF	FIDX[6:0]			Res.	FDF	BRS	DLC[3:0]	RXTS[15:0]			
R2	DB3 [7:0]				DB2 [7:0]				DB1 [7:0]		D[7:0]	
R3	DB7 [7:0]				DB6 [7:0]				DB5 [7:0]		DB4 [7:0]	
⋮	⋮				⋮				⋮			
Rn	DBm[7:0]				DBm-1[7:0]				DBm-2[7:0]		DBm-3[7:0]	

CAN FDメッセージの格納について設定された要素サイズは、64バイトのデータフィールドに設定されます。

表715.Rx FIFO要素の説明

フィールド	説明
R0bit 31 ESI	エラー状態インジケータ – 0：送信中のノードはエラーアクティブです。 – 1：送信中のノードはエラーパッシブです。
R0bit 30 XTD	拡張識別子 受信フレームに標準識別子または拡張識別子があるかどうかをホストに信号送信します。 – 0：11bit標準識別子です。 – 1：29bit拡張識別子です。
R0bit 29 RTR	リモート送信リクエスト 受信フレームがデータフレームかリモートフレームかどうかをホストに信号送信します。 – 0：受信フレームはデータフレームです。 – 1：受信フレームはリモートフレームです。
R0bit 28:0 ID[28:0]	識別子 ビットXTDに応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に保存されます。
R1bit 31 ANMF	許可される一致しないフレーム 一致しないフレームの許可は、RXGFC[ANFS] およびRXGFC[ANFE] 経由で有効にできます。 – 0：受信フレームがフィルタインデックスFIDXに一致します。 – 1：受信フレームがどのRxフィルタ要素にも一致しません。
R1bit 30:24 FIDX[6:0]	フィルタインデックス 0-27 = 一致しているRx受入フィルタ要素のインデックス（ANMF = 1の場合は無効）。 範囲は、0～RXGFC[LSS] - 1またはRXGFC[LSE] - 1です。
R1bit 21 FDF	FDフォーマット – 0：標準フレームフォーマットです。 – 1：FDCANフレームフォーマット（新しいDLCコーディングおよびCRC）

表715.Rx FIFO要素の説明 （続き）

フィールド	説明
R1bit 20 BRS	ビットレート切り替え – 0：ビットレート切り替えなしで受信したフレームです。 – 1：ビットレート切り替えありで受信したフレームです。
R1bit 19:16 DLC[3:0]	データ長コード – 0～8：クラシックCAN + CAN FD：受信フレームに0～8のデータバイトがあります。 – 9～15：クラシックCAN：受信フレームに8データバイトがあります。 – 9～15：CAN FD：受信フレームに12/16/20/24/32/48/64のデータバイトがあります。
R1bit 15:0 RXTS[15:0]	Rxタイムスタンプ フレーム受信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラTSCC[TCP] の設定に依存します。
R2bit 31:24 DB3 [7:0]	データバイト3
R2bit 23:16 DB2 [7:0]	データバイト2
R2bit 15:8 DB1 [7:0]	データバイト1
R2bit 7:0 D[7:0]	データバイト0
R3bit 31:24 DB7 [7:0]	データバイト7
R3bit 23:16 DB6 [7:0]	データバイト6
R3bit 15:8 DB5 [7:0]	データバイト5
R3bit 7:0 DB4 [7:0]	データバイト4
⋮	⋮
Rnビット 31:24 DBm[7:0]	データバイトm
Rnビット 23:16 DBm-1[7:0]	データバイトm-1
Rnビット15:8 DBm-2[7:0]	データバイトm-2
Rnビット7:0 DBm-3[7:0]	データバイトm-3

70.3.6 FDCANのTxバッファ要素

Txバッファセクション（3つの要素）はTx FIFOまたはTxキューを保持するように設定できます。Txハンドラは Tx FIFOとTxキューをTxバッファ設定FDCAN_TXBC.TFQMを使用して区別します。要素サイズは、最大64バイトのデータがあるCAN FDメッセージの格納について設定されます。

表716.TxバッファおよびFIFO要素

ビット	31			24	23					16	15	8	7	0
T0	ESI	XTD	RTR	ID[28:0]										
T1	MM[7:0]				EFC	Res.	FDF	BPS	DLC[3:0]		Res.			
T2	DB3 [7:0]				DB2 [7:0]						DB1 [7:0]		D[7:0]	
T3	DB7 [7:0]				DB6 [7:0]						DB5 [7:0]		DB4 [7:0]	
⋮	⋮				⋮						⋮			
Tn	DBm[7:0]				DBm-1[7:0]						DBm-2[7:0]		DBm-3[7:0]	

表717.Txバッファ要素の説明

フィールド	説明
T0bit 31 ESI(1)	エラー状態インジケータ – 0：CAN FDフォーマットのESIビットはエラーパッシブフラグにのみ依存します。 – 1：CAN FDフォーマットのESIビットはリセッシブで送信されました。
T0bit 30 XTD	拡張識別子 – 0：11bit標準識別子です。 – 1：29bit拡張識別子です。
T0bit 29 RTR(2)	リモート送信リクエスト – 0：データフレームを送信します。 – 1：リモートフレームを送信します。
T0bit 28:0 ID[28:0]	識別子 ビットXTDに応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
T1bit 31:24 MM[7:0]	メッセージマーカー Txバッファ設定中にCPUによって書き込まれます。Txメッセージステータスの識別のためにTxイベントFIFO要素にコピーされます。
T1bit 23 EFC	イベントFIFO制御 – 0：Txイベントを保存しません。 – 1：Txイベントを保存します。
T1bit 21 FDF	FDフォーマット – 0：クラシックCANフォーマットでフレームが送信されます。 – 1：CAN FDフォーマットでフレームが送信されます。
T1bit 20 BRS(3)	ビットレート切り替え – 0：ビットレート切り替えなしで送信されたCAN FDフレームです。 – 1：ビットレート切り替えありで送信されたCAN FDフレームです。

表717.Txバッファ要素の説明 （続き）

フィールド	説明
T1bit 19:16 DLC[3:0]	データ長コード – 0～8：クラシックCAN + CAN FD：受信フレームに0～8のデータバイトがあります。 – 9～15：クラシックCAN：受信フレームに8データバイトがあります。 – 9～15：CAN FD：受信フレームに12/16/20/24/32/48/64のデータバイトがあります。
T2bit 31:24 DB3 [7:0]	データバイト3
T2bit 23:16 DB2 [7:0]	データバイト2
T2bit 15:8 DB1 [7:0]	データバイト1
T2bit 7:0 D[7:0]	データバイト0
T3bit 31:24 DB7 [7:0]	データバイト7
T3bit 23:16 DB6 [7:0]	データバイト6
T3bit 15:8 DB5 [7:0]	データバイト5
T3bit 7:0 DB4[7:0]	データバイト4
⋮	⋮
Tnビット31:24 DBm[7:0]	データバイトm
Tnビット23:16 DBm-1[7:0]	データバイトm-1
Tnビット15:8 DBm-2[7:0]	データバイトm-2
Tnビット7:0 DBm-3[7:0]	データバイトm-3

1. 送信バッファのESIビットは、エラーパッシブフラグで論理和がとられ、送信されるFDフレームでのESIビットの値を決定します。CAN FDプロトコルの仕様によって必須とされるとおり、エラーアクティブノードではオプションでESIビットリセッシブを送信できますが、エラーパッシブノードでは常にESIビットリセッシブを送信します。
2. RTR = 1のとき、CCCR.FDOEがCAN FDフォーマットで送信を有効にしている場合でも、FDCANはISO11898-1に従ってリモートフレームを送信します。
3. ビットESI、FDF、BRSはCAN FD動作が有効（CCCR.FDOE = 1）のときのみ評価されます。ビットBRSはさらにCCCR.BRSE = 1のときのみ評価されます。

70.3.7 FDCAN TxイベントFIFO要素

各要素には、送信されたメッセージに関する情報が保存されています。TxイベントFIFOを読み出すことで、ホストCPUはメッセージが送信された順にこの情報を取得します。TxイベントFIFOに関するステータス情報は、レジスタTXEFSから取得できます。

表718.TxイベントFIFO要素

ビット	31			24	23					16	15	8	7	0
E0	ESI	XTD	RTR	ID[28:0]										
E1	MM[7:0]				ET[1:0]		EDL	BRS	DLC[3:0]		TXTS[15:0]			

表719.TxイベントFIFO要素の説明

フィールド	説明
E0bit 31 ESI	エラー状態インジケータ – 0：送信中のノードはエラーアクティブです。 – 1：送信中のノードはエラーパッシブです。
E0bit 30 XTD	拡張識別子 – 0：11bit標準識別子です。 – 1：29bit拡張識別子です。
E0bit 29 RTR	リモート送信リクエスト – 0：データフレームを送信します。 – 1：リモートフレームを送信します。
E0bit 28:0 ID[28:0]	識別子 ビットXTDに応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
E1bit 31:24 MM[7:0]	メッセージマーカー Txメッセージステータスの識別のためにTxバッファからTxイベントFIFO要素にコピーされます。
E1bit 23:22 EFC	イベントタイプ – 00：予約済み – 01：Txイベント – 10：キャンセル時でも送信（DARモードでの送信では常にセット） – 11：予約済み
E1bit 21 EDL	拡張データ長 – 0：標準フレームフォーマットです。 – 1：FDCANフレームフォーマット（新しいDLCコーディングおよびCRC）
E1bit 20 BRS	ビットレート切り替え – 0：ビットレート切り替えなしで送信されたフレームです。 – 1：ビットレート切り替えありで送信されたフレームです。
T1bit 19:16 DLC[3:0]	データ長コード 0～8：0～8のデータバイトのあるフレームが送信されます。 9～15：8データバイトのあるフレームが送信されます。
E1bit 15:0 TXTS[15:0]	Txタイムスタンプ フレーム送信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラTSCC[TCP] の設定に依存します。

70.3.8 FDCAN標準メッセージIDフィルタ要素

最大28のフィルタ要素を11bit標準IDに設定できます。標準メッセージIDフィルタ要素にアクセスするとき、そのアドレスはフィルタリストの標準開始アドレスFLSSAとフィルタ要素のインデックス（0～27）です。

表720.標準メッセージIDフィルタ要素

ビット	31		24	23	16	15	8	7	0
S0	SFT[1:0]	SFEC[2:0]		SFID1 [10:0]		Res.		SFID2 [10:0]	

表721.標準メッセージIDフィルタ要素フィールドの説明

フィールド	説明
ビット31:30 SFT[1:0](1)	標準フィルタタイプ – 00：SFID1からSFID2までの範囲のフィルタ – 01：SFID1またはSFID2のデュアルIDフィルタ – 10：クラシックフィルタ：SFID1 = フィルタ、SFID2 = マスク – 11：フィルタ要素無効
ビット29:27 SFEC[2:0]	標準フィルタ要素の設定 すべての有効化されたフィルタ要素は、標準フレームの受入フィルタに使用されます。受入フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。SFEC = 100、101、または110の場合、一致すると割込みフラグIR.HPMがセットされ、有効であれば、割込みが生成されます。この場合、レジスタHPMSは優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合にRx FIFO 0に格納 – 010：フィルタが一致した場合にRx FIFO 1に格納 – 011：フィルタが一致した場合にIDを拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットしてFIFO 0に格納 – 110：フィルタが一致した場合に優先度をセットしてFIFO 1に格納 – 111：未使用
ビット26:16 SFID1 [10:0]	標準フィルタID 1 標準IDフィルタ要素の最初のID
ビット10:0 SFID2 [10:0]	標準フィルタID 2 標準IDフィルタ要素の2番目のID

1. SFT = 11でフィルタ要素は無効化され、受入フィルタは続行します（SFEC = 000と同じ動作）。

注： **予約済みの値が設定されている場合、フィルタ要素は無効とみなされます。**

70.3.9 FDCAN拡張メッセージIDフィルタ要素

最大8のフィルタ要素を29bit拡張IDに設定できます。拡張メッセージIDフィルタ要素にアクセスするとき、そのアドレスはフィルタリストの拡張開始アドレスFLESAにフィルタ要素のインデックス（0～7）の2倍を加えたものです。

表722.拡張メッセージIDフィルタ要素

ビット	31		24	23 16	15 8	7 0
F0	EFEC[2:0]			EFID1 [28:0]		
F1	EFTI[1:0]	Res.		EFID2 [28:0]		

表723.拡張メッセージIDフィルタ要素フィールドの説明

フィールド	説明
F0bit 31:29 EFEC[2:0]	拡張フィルタ要素の設定 すべての有効化されたフィルタ要素は、拡張フレームの受入フィルタに使用されます。受入フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。EFEC = 100、101、または110の場合、一致すると割込みフラグIR[HPM] がセットされ、有効であれば、割込みが生成されます。この場合、レジスタHPMSは優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合にRx FIFO 0に格納 – 010：フィルタが一致した場合にRx FIFO 1に格納 – 011：フィルタが一致した場合にIDを拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットしてFIFO 0に格納 – 110：フィルタが一致した場合に優先度をセットしてFIFO 1に格納 – 111：未使用
F0bit 28:0 EFID1 [28:0]	拡張フィルタID 1 拡張IDフィルタ要素の最初のID Rx FIFOのフィルタ時、このフィールドで格納する拡張メッセージのIDを定義します。受信した識別子は完全一致とし、XIDAMマスキングメカニズムのみです。
F1bit 31:30 EFT[1:0]	拡張フィルタタイプ – 00：EF1IDからEF2IDまでの範囲のフィルタ（EF2ID ≥ EF1ID） – 01：EF1IDまたはEF2IDのデュアルIDフィルタ – 10：クラシックフィルタ：EF1ID = フィルタ、EF2ID = マスク – 11：EF1IDからEF2IDまでの範囲のフィルタ（EF2ID ≥ EF1ID）、XIDAMマスク適用なし
F1bit 29	未使用
F1bit 28:0 EFID2 [28:0]	拡張フィルタID 2 標準IDフィルタ要素の2番目のID

70.4 FDCANレジスタ

70.4.1 FDCANコア開放レジスタ（FDCAN_CREL）

アドレスオフセット：0x0000

リセット値：0x3214 1218

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 **REL[3:0]**：3

ビット27:24 **STEP[3:0]**：2

ビット23:20 **SUBSTEP[3:0]：**1

ビット19:16 **YEAR[3:0]**：4

ビット15:8 **MON[7:0]**：12

ビット7:0 **DAY[7:0]**：18

70.4.2 FDCANエンディアンレジスタ（FDCAN_ENDN）

アドレスオフセット：0x0004

リセット値：0x8765 4321

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ETV[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **ETV[31:0]**：エンディアンテスト値

エンディアンテスト値は0x8765 4321です。

注： **レジスタ読出しでは、エンディアンに問題がないことを確認するために、リセット値を示す必要があります。**

70.4.3 FDCANデータビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP）

アドレスオフセット：0x000C

リセット値：0x0000 0A33

このレジスタは、CCCR.CCEおよびCCCR.INITがセットされている場合のみ書込みできます。CANタイムクォンタムは、1～32 FDCANクロック周期の範囲でプログラム可能です。tq = (DBRP + 1) FDCANクロック周期。

DTSEG1は、Prop_SegとPhase_Seg1の合計です。DTSEG2は、Phase_Seg2です。そのため、ビット時間の長さは（プログラムされた値）[DTSEG1 + DTSEG2 + 3] tqまたは（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tqとなります。

情報処理時間（IPT）は0で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]				
								rw			rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0]			
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **TDC**：トランシーバ遅延補正

0：トランシーバ遅延補正は無効です。

1：トランシーバ遅延補正は有効です。

ビット22:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **DBRP[4:0]**：データビットレートプリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。ボーレートプリスケーラに有効な値は0～31です。ハードウェアはこの値をプログラムされた値 +1として解釈します。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **DTSEG1[4:0]**：サンプルポイント前のデータ時間セグメント

有効な値は0～31です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：t_{BS1} = (DTSEG1 + 1) x tqです。

ビット7:4 **DTSEG2[3:0]**：サンプルポイント後のデータ時間セグメント

有効な値は0～15です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：t_{BS2} = (DTSEG2 + 1) x tqです。

ビット3:0 **DSJW[3:0]**：同期ジャンプ幅

有効な値は0～15です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：t_{SJW} = (DSJW + 1) x tqです。

注： 8 MHzのFDCANクロックで、0x00000A33のリセット値がFDCANを500 kbit/sの高速ビットレートに設定します。

注： データフェーズのビットレートは、公称ビットレート以上である必要があります。

70.4.4 FDCANテストレジスタ（FDCAN_TEST）

ビットCCCR[TEST] を1にセットすると、このレジスタへの書込みアクセスが有効になります。CCCR[TEST] がリセットされると、すべてのレジスタ機能がリセット値にセットされます。

TxピンFDCANx_TXのループバックモードおよびソフトウェア制御はハードウェアテストモードです。TXを00以外にプログラムすると、CANバスでのメッセージ転送を妨害します。

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX[1:0]		LBCK	Res.	Res.	Res.	Res.
								r	rw	rw	rw				

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **RX**：受信ピン

ピンFDCANx_RXの実際の値を監視します。

0：CANバスはドミナントです（FDCANx_RX = 0）。

1：CANバスはリセッシブです（FDCANx_RX = 1）。

ビット6:5 **TX[1:0]**：送信ピンの制御

00：リセット値、FDCANx_TX TXはCANコアによって制御され、CANビット時間の最後に更新されます。

01：サンプルポイントはピンFDCANx_TXで監視できます。

10：ピンFDCANx_TXでドミナント（0）レベルです。

11：ピンFDCANx_TXでリセッシブ（1）です。

ビット4 **LBCK**：ループバックモード

0：リセット値、ループバックモードは無効です。

1：ループバックモードは有効です（パワーダウン（SLEEPモード）を参照）。

ビット3:0 予約済みであり、リセット値に保持する必要があります。

70.4.5 FDCAN RAMウォッチドッグレジスタ（FDCAN_RWD）

RAMウォッチドッグはメッセージRAMのREADY出力を監視します。メッセージRAMにアクセスすると、RWD[WDC] ビットで設定された値でメッセージRAMウォッチドッグカウンタが開始されます。

メッセージRAMがREADY出力を有効にすることで完了成功の信号を送信すると、カウンタはRWD[WDC] ビットで再ロードされます。カウンタが0までのカウントダウンを完了するまでにメッセージRAMからレスポンスがない場合、カウンタは停止し、割込みフラグIR[WDI] ビットがセットされます。RAMウォッチドッグカウンタは、fdcan_pclkクロックによってクロック供給されます。

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDV[7:0]								WDC[7:0]							
r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:8 **WDV[7:0]**：ウォッチドッグ値

実際のメッセージRAMウォッチドッグカウンタの値です。

ビット7:0 **WDC[7:0]**：ウォッチドッグ設定

メッセージRAMウォッチドッグカウンタの開始値です。00のリセット値で、カウンタは無効になります。これらは書込み保護（P）ビットで、FDCAN_CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

70.4.6 FDCAN CC制御レジスタ（FDCAN_CCCR）

アドレスオフセット：0x0018

リセット値：0x0000 0001

シングルビットのセットおよびリセットの詳細については、ソフトウェアの初期化を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INIT
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	r	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **NISO**：非ISO動作

このビットがセットされた場合、FDCANはBosch CAN FD Specification V1.0によって指定されたCAN FDフレームフォーマットを使用します。

0：ISO11898-1に準拠したCAN FDフレームフォーマットです。

1：Bosch CAN FD Specification V1.0に準拠したCAN FDフレームフォーマットです。

ビット14 **TXP**：

このビットがセットされた場合、FDCANはフレームの送信成功後、次の送信を開始する前に2 CANビット時間、一時停止します。

0：無効

1：有効

ビット13 **EFBI**：バス統合中のエッジフィルタ

0：エッジフィルタは無効です。

1：2つの連続したドミナントtqは、ハード同期のためにエッジを検出する必要があります。

ビット12 **PXHD**：プロトコル例外処理ディスエーブル

0：プロトコル例外処理は有効です。

1：プロトコル例外処理は無効です。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **BRSE**：FDCANビットレート切り替え

0：送信のビットレート切り替えは無効です。

1：送信のビットレート切り替えは有効です。

ビット8 **FDOE**：FD動作イネーブル

0：FD動作は無効です。

1：FD動作は有効です。

ビット7 **TEST**：テストモードイネーブル

0：通常動作、レジスタTESTはリセット値を保持します。

1：テストモード、レジスタTESTへの書込みアクセスは有効です。

ビット6 **DAR**：自動再送信の無効化

0：正常に送信されなかったメッセージの自動再送信は有効です。

1：自動再送信は無効です。

ビット5 **MON**：バスモニタモード

MONビットは、CCEおよびINITの両方が1にセットされているときのみソフトウェアによってセットできます。このビットはいつでもホストによってリセットできます。

0：バスモニタモードは無効です。

1：バスモニタモードは有効です。

ビット4 **CSR**：クロック停止リクエスト

0：クロック停止はリクエストされません。

1：クロック停止がリクエストされます。クロック停止がリクエストされたとき、すべてのペンディング中の転送リクエストが完了してCANバスがアイドルに達した後、まずINITがセットされ、次にCSAがセットされます。

ビット3 **CSA**：クロック停止の確認応答

0：クロック停止の確認応答はありません。

1：FDCANは、APBクロックおよびカーネルクロックを停止することで、パワーダウンにセットできます。

ビット2 **ASM**：ASM制限動作モード

制限動作モードは、異なるCANビットレートに適応するアプリケーションを対象とします。アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。オプションの制限動作モードでは、ノードはデータとリモートフレームを送信および受信でき、有効なフレームに確認応答を付与しますが、アクティブエラーフレームまたはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待ってCAN通信と再同期します。エラーカウンタはインクリメントされません。ASMビットは、CCEおよびINITの両方が1にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってリセットできます。

0：通常CAN動作です。

1：制限動作モードはアクティブです。

ビット1 **CCE**：設定変更イネーブル

0：CPUには、保護された設定レジスタへの書込みアクセス権がありません。

1：CPUには、保護された設定レジスタへの書込みアクセス権があります（CCCR.INIT = 1のとき）。

ビット0 **INIT**：初期化

0：通常動作

1：初期化を開始します。

注： 2つのクロックドメインの同期メカニズムによって、INITに書き込まれた値を読み戻せるようになるまでに遅延が発生する場合があります。そのため、プログラマはINITを新しい値にセットする前に、INITを読み出してINITに書き込まれた前の値が受理されたことを確認する必要があります。

70.4.7 FDCAN公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）

アドレスオフセット：0x001C

リセット値：0x0600 0A03

このレジスタは、CCCR[CCE] およびCCCR[INIT] がセットされている場合のみ書込みできます。CANビット時間は、4～81 tqの範囲でプログラム可能です。CANタイムクォンタムは、[1～1024] FDCANカーネルクロック周期の範囲でプログラム可能です。

tq = (BRP + 1) FDCANクロック周期fdcan_clk

NTSEG1は、Prop_SegとPhase_Seg1の合計です。NTSEG2は、Phase_Seg2です。そのため、ビット時間の長さは（プログラムされた値）[NTSEG1 + NTSEG2 + 3] tqまたは（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tqとなります。

情報処理時間（IPT）は0で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSJW[6:0]							NBRP[8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NTSEG1 [7:0]								Res.	NTSEG2 [6:0]						
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:25 **NSJW[6:0]**：公称（再）同期ジャンプ幅

有効な値は0～127です。この値のハードウェアによる実際の解釈は、使用される値が1でインクリメントされたプログラムされた値です。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

ビット24:16 **NBRP[8:0]**：ビットレートプリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。有効な値は0～511です。この値のハードウェアによる実際の解釈は、ここでプログラムされた値より1つ大きな値が使用されます。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

ビット15:8 **NTSEG1[7:0]**：サンプルポイント前の公称時間セグメント

有効な値は0～255です。この値のハードウェアによる実際の解釈は、プログラムされた値より1つ大きな値が使用されます。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **NTSEG2[6:0]**：サンプルポイント後の公称時間セグメント

有効な値は0～127です。この値のハードウェアによる実際の解釈は、プログラムされた値より1つ大きな値が使用されます。

注： 48MHzのCANカーネルクロックで、0x06000A03のリセット値がFDCANを3Mbit/sのビットレートに設定します。

70.4.8 FDCANタイムスタンプカウンタ設定レジスタ（FDCAN_TSCC）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS[1:0]	
														rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **TCP[3:0]**：タイムスタンプカウンタプリスケーラ

タイムスタンプとタイムアウトカウンタの時間単位をCANビット時間の倍数 [1～16] で設定します。

この値のハードウェアによる実際の解釈は、ここでプログラムされた値より1つ大きな値が使用されます。

CAN FDモードでは、アービトレーションフェーズとデータフェーズで異なるCANビット時間のため、内部タイムスタンプカウンタTCPは一定の時間ベースを提供しません。CAN FDでは、タイムスタンプ生成用の外部カウンタが必要です（TSS = 10）。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

ビット15:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **TSS[1:0]**：タイムスタンプ選択

00：タイムスタンプカウンタ値は常に0x0000です。

01：タイムスタンプカウンタ値はTCPに従ってインクリメントされます。

10：TIM3の値（tim3_cnt[0:15]）から始まる外部タイムスタンプカウンタ。

11：00と同じです。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

70.4.9 FDCANタイムスタンプカウンタ値レジスタ（FDCAN_TSCV）

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSC[15:0]															
rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **TSC[15:0]**：タイムスタンプカウンタ

内部／外部タイムスタンプカウンタ値は、フレーム受信の開始時（RxおよびTxの両方）にキャプチャされます。TSCC[TSS] = 01の場合、タイムスタンプカウンタは、TSCC[TCP] の設定に応じてCANビット時間 [1～16] の倍数でインクリメントされます。ラップアラウンドによって、割込みフラグIR[TSW] がセットされます。書込みアクセスによって、カウンタは0にリセットされます。

TSCC.TSS = 10の場合、TSCは外部タイムスタンプカウンタ値を反映します。書込みアクセスによる影響はありません。

注： **「ラップアラウンド」とは、TSCVへの書込みアクセス以外で、0以外から0へのタイムスタンプカウンタ値の変化を指します。**

70.4.10 FDCANタイムアウトカウンタ設定レジスタ（FDCAN_TOCC）

アドレスオフセット：0x0028

リセット値：0xFFFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TOP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS[1:0]		ETOC
													rw	rw	rw

ビット31:16 **TOP[15:0]**：タイムアウト期間

タイムアウトカウンタ（ダウンカウンタ）の開始値です。タイムアウト期間を設定します。

ビット15:3 予約済みであり、リセット値に保持する必要があります。

ビット2:1 **TOS[1:0]**：タイムアウト選択

連続モードで動作している場合、TOCVへの書込みによって、カウンタはTOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。タイムアウトカウンタがFIFOのうち1つで制御されている場合、空のFIFOはTOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初のFIFO要素が格納されたときに開始します。

00：連続動作

01：TxイベントFIFOによって制御されるタイムアウト

10：Rx FIFO 0によって制御されるタイムアウト

11：Rx FIFO 1によって制御されるタイムアウト

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

ビット0 **ETOC**：タイムアウトカウンタ有効化

0：タイムアウトカウンタは無効です。

1：タイムアウトカウンタは有効です。

これは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ書込みアクセス可能です。

詳細については、タイムアウトカウンタを参照してください。

70.4.11 FDCANタイムアウトカウンタ値レジスタ（FDCAN_TOCV）

アドレスオフセット：0x002C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOC[15:0]															
rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **TOC[15:0]**：タイムアウトカウンタ

タイムアウトカウンタは、TSCC.TCPの設定に応じてCANビット時間 [1～16] の倍数でデクリメントされます。0までデクリメントされると、割込みフラグIR.TOOがセットされ、タイムアウトカウンタは停止します。開始およびリセット／再開条件は、TOCC.TOSで設定します。

70.4.12 FDCANエラーカウンタレジスタ（FDCAN_ECR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]							
								rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RP	REC[6:0]							TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:16 **CEL[7:0]**：CANエラーロギング

CANプロトコルエラーによって送信エラーカウンタまたは受信エラーカウンタがインクリメントされるたびにカウンタがインクリメントされます。これはCELへの読出しアクセスによってリセットされます。カウンタは0xFFで停止し、TECまたはRECの次のインクリメントによって割込みフラグIR[ELO] がセットされます。

アクセスタイプはRXで、読出し時にリセットされます。

ビット15 **RP**：受信エラーパッシブ

0：受信エラーカウンタが128のエラーパッシブレベル未満です。

1：受信エラーカウンタが128のエラーパッシブレベルに達しています。

ビット14:8 **REC[6:0]：**受信エラーカウンタ

受信エラーカウンタの実際の状態、値の範囲は0～127です。

ビット7:0 **TEC[7:0]**：送信エラーカウンタ

送信エラーカウンタの実際の状態、値の範囲は0～255です。

CCCR.ASMがセットされている場合、CANプロトコルエラーが検出されているが、CELがインクリメントされているときに、CANプロトコルコントローラはTECとRECをインクリメントしません。

70.4.13 FDCANプロトコルステータスレジスタ（FDCAN_PSR）

アドレスオフセット：0x0044

リセット値：0x0000 0707

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]						
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	rc_r	rc_r	rc_r	rc_r	rs	rs	rs	r	r	r	r	r	rs	rs	rs

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22:16 **TDCV[6:0]**：トランスミッタ遅延補正値

セカンダリサンプルポイントの位置であり、FDCAN_TXからFDCAN_RXまでで測定された遅延とTDCR.TDCOの合計で定義されます。SSPの位置は、データフェーズでは送信ビットの開始とセカンダリサンプルポイントの間の最小タイムクオンタ（mtq）の数です。有効な値は0～127 mtqです。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 **PXE**：プロトコル例外イベント

0：最後の読出しアクセス以降に発生したプロトコル例外イベントはありません。

1：プロトコル例外イベントが発生しました。

ビット13 **REDL**：受信FDCANメッセージ

このビットは、受入フィルタに関係なくセットされます。

0：このビットがCPUによってリセットされるまでに、受信したFDCANメッセージはありません。

1：EDLフラグがセットされたFDCANフォーマットのメッセージを受信しました。

アクセスタイプはRXで、読出し時にリセットされます。

ビット12 **RBRS**：最後に受信したFDCANメッセージのBRSフラグ

このビットは、受入フィルタに関係なくREDLとともにセットされます。

0：最後に受信したFDCANメッセージにはBRSフラグがセットされていません。

1：最後に受信したFDCANメッセージにはBRSフラグがセットされています。

アクセスタイプはRXで、読出し時にリセットされます。

ビット11 **RESI**：最後に受信したFDCANメッセージのESIフラグ

このビットは、受入フィルタに関係なくREDLとともにセットされます。

0：最後に受信したFDCANメッセージにはESIフラグがセットされていません。

1：最後に受信したFDCANメッセージにはESIフラグがセットされています。

アクセスタイプはRXで、読出し時にリセットされます。

ビット10:8 **DLEC[2:0]**：データの最後のエラーコード

BRSフラグをセットしたFDCANフォーマットフレームのデータフェーズで発生した最後のエラーのタイプです。コーディングはLECと同じです。このフィールドは、BRSフラグをセットしたFDCANフォーマットフレームがエラーなしで転送（受信または送信）されたときに0にクリアされます。

アクセスタイプはRSで、読出し時にセットされます。

ビット7 **BO**：Bus_Offステータス

0：FDCANは、Bus_Offではありません。

1：FDCANは、Bus_Off状態です。

ビット6 **EW**：警告ステータス

0：両方のエラーカウンタは96のError_Warning制限未満です。

1：少なくとも1つのエラーカウンタが96のError_Warning制限に達しました。

ビット5 **EP**：エラーパッシブ

0：FDCANは、Error_Active状態です。通常、バス通信に参加して、エラーを検出した場合にアクティブエラーフラグを送信します。

1：FDCANは、Error_Passive状態です。

ビット4:3 **ACT[1:0]**：動作

モジュールのCAN通信状態を監視します。

00：同期中：ノードがCAN通信で同期しています。

01：アイドル：ノードはレシーバでもトランスミッタでもありません。

10：レシーバ：ノードはレシーバとして動作しています。

11：トランスミッタ：ノードはトランスミッタとして動作しています。

ビット2:0 **LEC[2:0]**：最後のエラーコード

LECは、CANバスで発生した最後のエラーのタイプを示します。このフィールドは、メッセージがエラーなしで転送（受信または送信）されたときに0にクリアされます。

000：エラーなし：正常な受信または送信によってLECがリセットされた以降に発生したエラーはありません。

001：スタッフエラー：シーケンス上で許容されていない6bit以上の同一レベルの連続ビットが受信メッセージに含まれています。

010：フォームエラー：受信フレームの固定フォーマット部分に誤ったフォーマットがあります。

011：確認応答エラー：FDCANによって送信されたメッセージが他のノードによって確認応答されませんでした。

100：ビット1エラー：メッセージ（アービトレーションフィールドを除く）送信中、デバイスがリセッシブレベル（論理値1のビット）の送信を求めたが、監視されたバスの値がドミナントでした。

101：ビット0エラー：メッセージ（または確認応答ビット、アクティブエラーフラグ、オーバーロードフラグ）送信中、デバイスがドミナントレベル（データまたは識別子ビットの論理値0）の送信を求めたが、監視されたバスの値がリセッシブでした。Bus_Offリカバリ中、11のリセッシブビットのシーケンスが監視されるたびに、このステータスがセットされます。これにより、CPUが（バスがドミナントでスタックしていない、または継続的に妨害されていないことを示す）Bus_Offリカバリシーケンスの進行を監視できます。

110：CRCエラー：受信メッセージのCRCチェックサムが正しくありませんでした。着信メッセージのCRCが、受信したデータから計算されたCRCと一致しません。

111：変更なし：プロトコルステータスレジスタの読出しアクセスによって、LECが「7」に再初期化されます。LECが値 '7' を示している場合、プロトコルステータスレジスタの最後のCPU読出しアクセス以降にCANバスイベントは検出されませんでした。

アクセスタイプはRSで、読出し時にセットされます。

注： **FDCAN フォーマットのフレームが、BRS フラグがセットされたデータフェーズに達したとき、次のCANイベント（エラーまたは有効なフレーム）はLECの代わりにFLECで示されます。FDCAN CRCシーケンスの固定スタッフビットでのエラーは、スタッフエラーではなく、フォームエラーとして示されます。**

注： **Bus_Offリカバリシーケンス（CAN Specification Rev. 2.0またはISO11898-1参照）は、CCCR[INIT]のセットやリセットで短縮できません。デバイスがBus_Offになると、それ自体でCCCR.INITをセットし、すべてのバス動作を停止します。CCCR[INIT] がCPUによってクリアされると、デバイスは通常動作を再開する前に、129のバスアイドルの発生（129 × 11の連続リセッシブビット）を待ちます。Bus_Offリカバリシーケンスの最後に、エラー管理カウンタがリセットされます。CCCR[INIT] リセット後の待機時間中、11のリセッシブビットのシーケンスは監視され、ビット0エラーコードはPSR[LEC] に書き込まれ、CPUはCANバスがドミナントで詰まっていないかどうかや、継続的に妨害されていないかをすぐにチェックし、Bus_Offリカバリシーケンスを監視できます。ECR[REC] はこれらのシーケンスをカウントするために使用されます。**

70.4.14 FDCANトランスミッタ遅延補正レジスタ（FDCAN_TDCR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14:8 **TDCO[6:0]**：トランスミッタ遅延補正オフセット

FDCAN_TXからFDCAN_RXまでに測定された遅延とセカンダリサンプルポイントの距離を定義するオフセット値です。有効な値は0～127 mtqです。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **TDCF[6:0]**：トランスミッタ遅延補正フィルタのウィンドウ長

SSPの位置の最小値を定義し、より早いSSPの位置につながるFDCAN_RXのドミナントエッジは、トランスミッタ遅延測定では無視されます。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

70.4.15 FDCAN割込みレジスタ（FDCAN_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に1を書き込むことによってクリアされます。

0を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。IEの設定は、割込みが生成されるかどうかを制御します。ILSの設定は、割込みが信号送信される割込みラインを制御します。

アドレスオフセット：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO
								rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOO	MRAF	TSW	TEFL	TEFF	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1N	RF0L	RF0F	RF0N
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **ARA**：予約済みアドレスへのアクセス

0：予約済みアドレスへのアクセスは発生していません。

1：予約済みアドレスへのアクセスが発生しました。

ビット22 **PED**：データ･フェーズでのプロトコルエラー（データビット時間が使用されます）
0：データフェーズでのプロトコルエラーはありませんでした。
1：データフェーズでのプロトコルエラーが検出されました（0、7ではないPSR.DLEC）。

ビット21 **PEA**：アービトレーションフェーズでのプロトコルエラー（公称ビット時間が使用されます）
0：アービトレーションフェーズでのプロトコルエラーはありませんでした。
1：アービトレーションフェーズでのプロトコルエラーが検出されました（0、7ではないPSR.LEC）。

ビット20 **WDI**：ウォッチドッグ割込み
0：メッセージRAMウォッチドッグイベントは発生していません。
1：READYの欠落によるメッセージRAMウォッチドッグイベントがありました。

ビット19 **BO**：Bus_Offステータス
0：Bus_Offステータスは変更されていません。
1：Bus_Offステータスが変更されました。

ビット18 **EW**：警告ステータス
0：Error_Warningステータスは変更されていません。
1：Error_Warningステータスが変更されました。

ビット17 **EP**：エラーパッシブ
0：Error_Passiveステータスは変更されていません。
1：Error_Passiveステータスが変更されました。

ビット16 **ELO**：エラーロギングオーバーフロー
0：CANエラーロギングカウンタはオーバーフローしていません。
1：CANエラーロギングカウンタのオーバーフローが発生しました。

ビット15 **TOO**：タイムアウト発生
0：タイムアウトなし
1：タイムアウトに到達

ビット14 **MRAF**：メッセージRAMアクセス失敗
このフラグは、Rxハンドラが次の場合にセットされます。

- 次のメッセージのアービトレーションフィールドを受信するまでに、受入フィルタまたは許可メッセージの格納を完了しなかった場合です。この場合、受入フィルタまたはメッセージ格納はアボートされ、Rxハンドラが次のメッセージの処理を開始します。
- メッセージをメッセージRAMに書き込めなかった場合です。この場合、メッセージの格納は中止されます。

どちらの場合も、FIFO Putインデックスは更新されません。部分的に格納されたメッセージは、次のメッセージがこの位置に格納される際に上書きされます。
Txハンドラがメッセージ RAMから時間内にメッセージを読み出せなかったときもこのフラグがセットされます。この場合、メッセージ送信はアボートされます。Txハンドラアクセス失敗時、FDCANは制限動作モードに切り替わります（制限動作モードを参照）。制限動作モードを終了するには、ホストCPUでCCCR.ASMをリセットする必要があります。
0：メッセージRAMアクセス失敗は発生していません。
1：メッセージRAMアクセス失敗が発生しました。

ビット13 **TSW**：タイムスタンプラップアラウンド
0：タイムスタンプカウンタのラップアラウンドはありません。
1：タイムスタンプカウンタのラップアラウンドが発生しました。

ビット12 **TEFL**：TxイベントFIFO要素の喪失
0：TxイベントFIFO要素の喪失はありません。
1：TxイベントFIFO要素の喪失がありました。

ビット11 **TEFF**：TxイベントFIFOフル
0：TxイベントFIFOはフルではありません。
1：TxイベントFIFOはフルです。

ビット10 **TEFN**：TxイベントFIFO新規エントリ
0：TxイベントFIFOは変更されていません。
1：TxハンドラがTxイベントFIFO要素を書き込みました。

ビット9 **TFE**：Tx FIFOエンプティ
0：Tx FIFOはエンプティではありません。
1：Tx FIFOはエンプティです。

ビット8 **TCF**：送信キャンセル終了
0：終了した送信キャンセルはありません。
1：送信キャンセルが終了しました。

ビット7 **TC**：送信完了
0：完了した送信はありません。
1：送信が完了しました。

ビット6 **HPM**：高優先メッセージ
0：受信した高優先メッセージはありません。
1：高優先メッセージを受信しました。

ビット5 **RF1L**：Rx FIFO 1メッセージ喪失
0：喪失したRx FIFO 1メッセージはありません。
1：Rx FIFO 1メッセージを喪失しました。

ビット4 **RF1F**：Rx FIFO 1フル
0：Rx FIFO 1はフルではありません。
1：Rx FIFO 1はフルです。

ビット3 **RF1N**：Rx FIFO 1新規メッセージ
0：Rx FIFO 1に書き込まれた新規メッセージはありません。
1：Rx FIFO 1に新規メッセージが書き込まれました。

ビット2 **RF0L**：Rx FIFO 0メッセージ喪失
0：喪失したRx FIFO 0メッセージはありません。
1：Rx FIFO 0メッセージを喪失しました。

ビット1 **RF0F**：Rx FIFO 0フル
0：Rx FIFO 0はフルではありません。
1：Rx FIFO 0はフルです。

ビット0 **RF0N**：Rx FIFO 0新規メッセージ
0：Rx FIFO 0に書き込まれた新規メッセージはありません。
1：Rx FIFO 0に新規メッセージが書き込まれました。

70.4.16 FDCAN割込み有効レジスタ（FDCAN_IE）

割込み有効レジスタの設定によって、割込みレジスタでのどのステータス変更が割込みラインで信号送信されるかが決まります。

アドレスオフセット：0x0054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1NE	RF0LE	RF0FE	RF0NE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **ARAE**：予約済みアドレスへのアクセスイネーブル

ビット22 **PEDE**：データフェーズでのプロトコルエラーイネーブル

ビット21 **PEAE**：アービトレーションフェーズでのプロトコルエラーイネーブル

ビット20 **WDIE**：ウォッチドッグ割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット19 **BOE**：Bus_Offステータス
0：割込みは無効です。
1：割込みは有効です。

ビット18 **EWE**：警告ステータスの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット17 **EPE**：エラーパッシブ割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット16 **ELOE**：エラーロギングオーバーフローの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット15 **TOOE**：タイムアウト発生の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット14 **MRAFE**：メッセージRAMアクセス失敗の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット13 **TSWE**：タイムスタンプラップアラウンドの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット12 **TEFLE**：TxイベントFIFO要素の喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット11 **TEFFE**：TxイベントFIFOフルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット10 **TEFNE**：TxイベントFIFO新規エントリの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット9 **TFEE**：Tx FIFOエンプティ割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット8 **TCFE**：送信キャンセル終了の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット7 **TCE**：送信完了の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット6 **HPME**：高優先メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット5 **RF1LE**：Rx FIFO 1メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット4 **RF1FE**：Rx FIFO 1フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット3 **RF1NE**：Rx FIFO 1新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット2 **RF0LE**：Rx FIFO 0メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット1 **RF0FE**：Rx FIFO 0フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット0 **RF0NE**：Rx FIFO 0新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

70.4.17 FDCAN割込みライン選択レジスタ（FDCAN_ILS）

このレジスタは、割込みレジスタの割込みフラグの特定のグループによって生成される割込みを、2つのモジュール割込みラインのうちの1つに割り当てます。割込み生成には、それぞれの割込みラインをILE[EINT0] およびILE[EINT1] で有効にする必要があります。

アドレスオフセット：0x0058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERR	BERR	MISC	TFERR	SMSG	RXFIFO1	RXFIFO0
									rw	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **PERR**：次の割込みをグループ化したプロトコルエラー
- ARAL：予約済みアドレスへのアクセスライン
- PEDL：データフェーズでのプロトコルエラーのライン
- PEAL：アービトレーションフェーズでのプロトコルエラーのライン
- WDIL：ウォッチドッグ割込みライン
- BOL：Bus_Offステータス
- EWL：警告ステータスの割込みライン

ビット5 **BERR:**：次の割込みをグループ化したビットおよびラインのエラー
- EPL：エラーパッシブの割込みライン
- ELOL：エラーロギングオーバーフローの割込みライン

ビット4 **MISC**：次の割込みを再グループ化した割込み
- TOOL：タイムアウト発生の割込みライン
- MRAFL：メッセージRAMアクセス失敗の割込みライン
- TSWL：タイムスタンプラップアラウンドの割込みライン

ビット3 **TFERR**：次の割込みをグループ化したTx FIFOエラー
- TEFLL：TxイベントFIFO要素の喪失の割込みライン
- TEFFL：TxイベントFIFOフルの割込みライン
- TEFNL：TxイベントFIFO新規エントリの割込みライン
- TFEL：Tx FIFOエンプティの割込みライン

ビット2 **SMSG**：次の割込みをグループ化したステータスメッセージビット
- TCFL：送信キャンセル終了の割込みライン
- TCL：送信完了の割込みライン
- HPML：高優先メッセージの割込みライン

ビット1 **RXFIFO1**：次の割込みをグループ化したRX FIFOビット
- RF1LL：Rx FIFO 1メッセージ喪失の割込みライン
- RF1FL：Rx FIFO 1フルの割込みライン
- RF1NL：Rx FIFO 1新規メッセージの割込みライン

ビット0 **RXFIFO0**：次の割込みをグループ化したRX FIFOビット
- RF0LL：Rx FIFO 0メッセージ喪失の割込みライン
- RF0FL：Rx FIFO 0フルの割込みライン
- RF0NL：Rx FIFO 0新規メッセージの割込みライン

70.4.18 FDCAN割込みライン有効レジスタ（FDCAN_ILE）

CPUへの2つの割込みラインは、それぞれビットEINT0およびEINT1をプログラムすることで有効化／無効化できます。

アドレスオフセット：0x005C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **EINT1**：割込みライン1有効化

0：割込みラインfdcan_intr0_itlは無効です。

1：割込みラインfdcan_intr0_itlは有効です。

ビット0 **EINT0**：割込みライン0有効化

0：割込みラインfdcan_intr1_itlは無効です。

1：割込みラインfdcan_intr1_itlは有効です。

70.4.19 FDCANグローバルフィルタ設定レジスタ（FDCAN_RXGFC）

メッセージIDフィルタのグローバル設定です。グローバルフィルタ設定は、図880および図881に示されたとおり、標準メッセージおよび拡張メッセージのフィルタパスを制御します。

アドレスオフセット：0x0080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	LSE[3:0]				Res.	Res.	Res.	LSS[4:0]				
				rw	rw	rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F0OM	F1OM	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
						rw	rw			rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27:24 **LSE[3:0]**：リストサイズ拡張

0：拡張メッセージIDフィルタはありません。

1～8：拡張メッセージIDフィルタ要素の数です。

>8：8を超える値は8として解釈されます。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット23:21 予約済みであり、リセット値に保持する必要があります。

ビット20:16 **LSS[4:0]**：リストサイズ標準

0：標準メッセージIDフィルタはありません。

1～28：標準メッセージIDフィルタ要素の数です。

>28：28を超える値は28として解釈されます。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **F0OM**：FIFO 0動作モード（上書きまたはブロック）

これは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット8 **F1OM**：FIFO 1動作モード（上書きまたはブロック）

これは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ANFS[1:0]**：一致しないフレーム（標準）の許可

フィルタリストの要素と一致しない11bit IDを持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0で受入れ

01：Rx FIFO 1で受入れ

10：拒否

11：拒否

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット3:2 **ANFE[1:0]**：一致しないフレーム（拡張）の許可

フィルタリストの要素と一致しない29bit IDを持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0で受入れ

01：Rx FIFO 1で受入れ

10：拒否

11：拒否

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット1 **RRFS**：リモートフレーム（標準）の拒否

0：11bit標準IDを持つリモートフレームをフィルタします。

1：11bit標準IDを持つすべてのリモートフレームを拒否します。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット0 **RRFE**：リモートフレーム（拡張）の拒否

0：29bit標準IDを持つリモートフレームをフィルタします。

1：29bit標準IDを持つすべてのリモートフレームを拒否します。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

70.4.20 FDCAN拡張IDおよびマスクレジスタ（FDCAN_XIDAM）

アドレスオフセット：0x0084

リセット値：0x1FFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EIDM[28:16]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIDM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:0 **EIDM[28:0]**：拡張IDマスク

拡張フレームの受入フィルタでは、拡張ID ANDマスクは受信フレームのメッセージIDとANDでつながれます。SAE J1939での29bit IDのマスキングを対象とします。すべてのビットのリセット値を1にすると、マスクはアクティブではなくなります。

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

70.4.21 FDCAN高優先メッセージステータスレジスタ（FDCAN_HPMS）

このレジスタは、メッセージIDフィルタ要素が優先度イベント一致を生成するように設定するたびに更新されます。これを使用して、着信高優先メッセージのステータスを監視し、これらのメッセージへの高速アクセスが可能になります。

アドレスオフセット：0x0088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLST	Res.	Res.	FIDX[4:0]					MSI[1:0]		Res.	Res.	Res.	BIDX[2:0]		
r			r	r	r	r	r	r	r				r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **FLST**：フィルタリスト

一致フィルタ要素のフィルタリストを示します。

0：標準フィルタリスト

1：拡張フィルタリスト

ビット14:13 予約済みであり、リセット値に保持する必要があります。

ビット12:8 **FIDX[4:0]**：フィルタインデックス

一致フィルタ要素のインデックスです。範囲は、0～RXGFC[LSS] - 1またはRXGFC[LSE] - 1です。

ビット7:6 **MSI[1:0]**：メッセージストレージインジケータ
00：FIFOが選択されていません。
01：FIFOオーバーランです。
10：メッセージがFIFO 0に格納されます。
11：メッセージがFIFO 1に格納されます。

ビット5:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **BIDX[2:0]**：バッファインデックス
メッセージが格納されたRx FIFO要素のインデックスです。MSI[1] = 1の場合のみ有効です。

70.4.22 FDCAN Rx FIFO 0ステータスレジスタ（FDCAN_RXF0S）

アドレスオフセット：0x0090

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	Res.	Res.	Res.	Res.	F0PI [1:0]	
						r	r							r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F0GI [1:0]		Res.	Res.	Res.	Res.	F0FL [3:0]			
						r	r					r	r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **RF0L**：Rx FIFO 0メッセージ喪失
このビットは割込みレジスタIR[RF0L] のコピーです。IR[RF0L] がリセットされると、このビットもリセットされます。
0：喪失したRx FIFO 0メッセージはありません。
1：Rx FIFO 0メッセージを喪失しました。サイズ0のRx FIFO 0への書込み試行後にもセットされます。

ビット24 **F0F**：Rx FIFO 0フル
0：Rx FIFO 0はフルではありません。
1：Rx FIFO 0はフルです。

ビット23:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **F0PI[1:0]**：Rx FIFO 0 Putインデックス
Rx FIFO 0書込みインデックスポインタで、範囲は0～2です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **F0GI[1:0]**：Rx FIFO 0 Getインデックス
Rx FIFO 0読出しインデックスポインタで、範囲は0～2です。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **F0FL[3:0]**：Rx FIFO 0フィルレベル
Rx FIFO 0に格納される要素の数で、範囲は0～3です。

70.4.23 CAN Rx FIFO 0確認応答レジスタ（FDCAN_RXF0A）

アドレスオフセット：0x0094

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **F0AI[2:0]**：Rx FIFO 0確認応答インデックス

ホストはRx FIFO 0からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 0から読み出された最後の要素のバッファインデックスをF0AIに書き込む必要があります。これにより、Rx FIFO 0 GetインデックスRXF0S[F0GI] がF0AI + 1にセットされ、FIFO 0フィルレベルRXF0S[F0FL] が更新されます。

70.4.24 FDCAN Rx FIFO 1ステータスレジスタ（FDCAN_RXF1S）

アドレスオフセット：0x0098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.	Res.	Res.	Res.	F1PI [1:0]	
						r	r							r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F1GI [1:0]		Res.	Res.	Res.	Res.	F1FL [3:0]			
						r	r					r	r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **RF1L**：Rx FIFO 1メッセージ喪失

このビットは割込みレジスタIR[RF1L] のコピーです。IR[RF1L] がリセットされると、このビットもリセットされます。

0：喪失したRx FIFO 1メッセージはありません。

1：Rx FIFO 1メッセージを喪失しました。サイズ0のRx FIFO 1への書込み試行後にもセットされます。

ビット24 **F1F**：Rx FIFO 1フル

0：Rx FIFO 1はフルではありません。

1：Rx FIFO 1はフルです。

ビット23:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **F1PI[1:0]**：Rx FIFO 1 Putインデックス

Rx FIFO 1書込みインデックスポインタで、範囲は0～2です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **F1GI[1:0]**：Rx FIFO 1 Getインデックス

Rx FIFO 1読出しインデックスポインタで、範囲は0～2です。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **F1FL[3:0]**：Rx FIFO 1フィルレベル

Rx FIFO 1に格納される要素の数で、範囲は0～3です。

70.4.25 FDCAN Rx FIFO 1確認応答レジスタ（FDCAN_RXF1A）

アドレスオフセット：0x009C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **F1AI[2:0]**：Rx FIFO 1確認応答インデックス

ホストはRx FIFO 1からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 1から読み出された最後の要素のバッファインデックスをF1AIに書き込む必要があります。これにより、Rx FIFO 1 GetインデックスRXF1S[F1GI] がF1AI + 1にセットされ、FIFO 1フィルレベルRXF1S[F1FL] が更新されます。

70.4.26 FDCAN Txバッファ設定レジスタ（FDCAN_TXBC）

アドレスオフセット：0x00C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **TFQM**：Tx FIFO／キューモード

0：Tx FIFO動作

1：Txキュー動作

これは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット23:0 予約済みであり、リセット値に保持する必要があります。

70.4.27 FDCAN Tx FIFO／キューステータスレジスタ（FDCAN_TXFQS）

Tx FIFO／キューステータスは、レジスタTXBRPに示されたペンディング中のTxリクエストに関連します。そのため、追加／キャンセルリクエストの効果は、実行中のTxスキャン（TXBRPがまだ更新されていない）によって遅延する場合があります。

アドレスオフセット：0x00C4

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	Res.	Res.	Res.	TFQPI[1:0]	
										r				r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TFGI[1:0]		Res.	Res.	Res.	Res.	Res.	TFFL[2:0]		
						r	r						r	r	r

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **TFQF**：Tx FIFO／キューフル

0：Tx FIFO／キューはフルではありません。

1：Tx FIFO／キューはフルです。

ビット20:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **TFQPI[1:0]**：Tx FIFO／キュー Putインデックス

Tx FIFO／キュー書込みインデックスポインタで、範囲は0～3です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **TFGI[1:0]**：Tx FIFO Getインデックス

Tx FIFO読出しインデックスポインタで、範囲は0～3です。Txキュー動作が設定された場合（TXBC.TFQM = 1）、0で読み出されます。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TFFL[2:0]**：Tx FIFOフリーレベル

連続するフリーTx FIFO要素の数はTFGIから開始し、範囲は0～3です。Txキュー動作が設定された場合（TXBC[TFQM] = 1）、0で読み出されます。

70.4.28 FDCAN Txバッファリクエストペンディングレジスタ（FDCAN_TXBRP）

アドレスオフセット：0x00C8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRP[2:0]		
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TRP[2:0]**：送信リクエストペンディング

各Txバッファには独自の送信リクエストペンディングビットがあります。このビットはレジスタTXBAR経由でセットされます。リクエストされた送信が完了した後か、レジスタTXBCR経由でキャンセルされた後に、ビットがリセットされます。

TXBRPビットがセットされた後、Txスキャンが開始され、最高優先度を持つペンディング中のTxリクエスト（一番小さいメッセージIDのTxバッファ）をチェックします。

キャンセルリクエストにより、レジスタTXBRPの対応する送信リクエストペンディングビットがリセットされます。キャンセルがリクエストされたときに送信をすでに開始していた場合、送信が成功したかどうかに関係なく、送信の最後に実行されます。対応するTXBRPビットがリセットされた直後に、キャンセルリクエストビットがリセットされます。

キャンセルがリクエストされた後、終了したキャンセルは次のタイミングでTXBCF経由で信号送信されます。

対応するTXBTOビットとともに送信が成功した後

キャンセル時点で送信がまだ開始されていなかったとき

ロストアービトレーションによって送信がアボートされたとき

フレーム送信中にエラーが発生したとき

DARモードでは、すべての送信が成功しなかった場合、自動的にキャンセルされます。すべての成功しなかった送信に対応するTXBCFビットがセットされます。

0：送信リクエストペンディングはありません。

1：送信リクエストペンディングがあります。

注： **Txスキャンが進行中のときにセットされたTXBRPビットは、この特定のTxスキャン中は考慮されません。このようなTxバッファに対してキャンセルがリクエストされた場合、この追加リクエストはすぐにキャンセルされ、対応するTXBRPビットはリセットされます。**

70.4.29 FDCAN Txバッファ追加リクエストレジスタ（FDCAN_TXBAR）

アドレスオフセット：0x00CC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR[2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **AR[2:0]**：追加リクエスト

各Txバッファには独自の追加リクエストビットがあります。1を書き込むと、対応する追加リクエストビットがセットされ、0を書き込んでも影響はありません。これにより、ホストは複数のTxバッファに対する送信リクエストをTXBARへの1回の書込みでセットできます。実行中のTxスキャンがないときはビットがすぐにリセットされ、それ以外の場合はTxスキャンプロセスが完了するまでビットがセットされたままとなります。

0：追加された送信リクエストはありません。

1：送信リクエストが追加されました。

注： **追加リクエストがペンディング中の送信リクエストを持つTxバッファに適用される（対応するTXBRPがすでにセットされている）場合、リクエストは無視されます。**

70.4.30 FDCAN Txバッファキャンセルリクエストレジスタ（FDCAN_TXBCR）

アドレスオフセット：0x00D0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CR[2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **CR[2:0]**：キャンセルリクエスト

各Txバッファには独自のキャンセルリクエストビットがあります。1を書き込むと、対応するCRビットがセットされ、0を書き込んでも影響はありません。

これにより、ホストは複数のTxバッファに対するキャンセルリクエストをTXBCRへの1回の書込みでセットできます。対応するTXBRPビットがリセットされるまで、このビットはセットされたままとなります。

0：ペンディング中のキャンセルはありません。

1：キャンセルがペンディングされています。

70.4.31 FDCAN Txバッファ送信発生レジスタ（FDCAN_TXBTO）

アドレスオフセット：0x00D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TO[2:0]		
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TO[2:0]**：送信発生

各Txバッファには独自のTOビットがあります。送信成功後に対応するTXBRPビットがクリアされたとき、ビットがセットされます。レジスタTXBARの対応するビットに1を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：発生した送信はありません。

1：送信が発生しました。

70.4.32 FDCAN Txバッファキャンセル終了レジスタ（FDCAN_TXBCF）

アドレスオフセット：0x00D8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CF[2:0]		
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **CF[2:0]**：キャンセル終了

各Txバッファには独自のCFビットがあります。キャンセルがTXBCRビットによってリクエストされた後に対応するTXBRPビットがクリアされたとき、このビットがセットされます。対応するTXBRPビットがキャンセル時点でセットされていなかった場合、CFはすぐにセットされます。レジスタTXBARの対応するビットに1を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：送信バッファキャンセルはありません。

1：送信バッファキャンセルが終了しました。

70.4.33 FDCAN Txバッファ送信割込み有効レジスタ（FDCAN_TXBTIE）

アドレスオフセット：0x00DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE[2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TIE[2:0]**：送信割込みイネーブル

各Txバッファには独自のTIEビットがあります。

0：送信割込みは無効です。

1：送信割込みは有効です。

70.4.34 FDCAN Txバッファ･キャンセル終了割込みイネーブル･レジスタ（FDCAN_TXBCIE）

アドレスオフセット：0x00E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFIE[2:0]		
													rw	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **CFIE[2:0]**：キャンセル終了の割込みイネーブル

各Txバッファには独自のCFIEビットがあります。

0：キャンセル終了割込みは無効です。

1：キャンセル終了割込みは有効です。

70.4.35 FDCAN TxイベントFIFOステータスレジスタ（FDCAN_TXEFS）

アドレスオフセット：0x00E4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	Res.	Res.	Res.	EFPI[1:0]	
						r	r							r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EFGI[1:0]		Res.	Res.	Res.	Res.	Res.	EFFL[2:0]		
						r	r						r	r	r

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25 **TEFL**：TxイベントFIFO要素の喪失

このビットは割込みレジスタIR[TEFL] のコピーです。IR[TEFL] がリセットされると、このビットもリセットされます。

0：TxイベントFIFO要素の喪失はありません。

1：TxイベントFIFO要素を喪失しました。サイズ0のTxイベントFIFOへの書込み試行後にもセットされます。

ビット24 **EFF**：イベントFIFOフル

0：TxイベントFIFOはフルではありません。

1：TxイベントFIFOはフルです。

ビット23:18 予約済みであり、リセット値に保持する必要があります。

ビット17:16 **EFPI[1:0]**：イベントFIFO Putインデックス

TxイベントFIFO書込みインデックスポインタで、範囲は0～3です。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **EFGI[1:0]**：イベントFIFO Getインデックス

TxイベントFIFO読出しインデックスポインタで、範囲は0～3です。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **EFFL[2:0]**：イベントFIFOフィルレベル

TxイベントFIFOに格納される要素の数で、範囲は0～3です。

70.4.36 FDCAN TxイベントFIFO確認応答レジスタ（FDCAN_TXEFA）

アドレスオフセット：0x00E8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[1:0]	
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **EFAI[1:0]**：イベントFIFO確認応答インデックス

ホストはTxイベントFIFOから要素または一連の要素を読み出した後、TxイベントFIFOから読み出された最後の要素のインデックスをEFAIに書き込む必要があります。これにより、TxイベントFIFO GetインデックスTXEFS[EFGI] がEFAI + 1にセットされ、FIFO 0フィルレベルTXEFS[EFFL] が更新されます。

70.4.37 FDCAN CFGクロック分周回路レジスタ（FDCAN_CKDIV）

アドレスオフセット：0x0100

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PDIV[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PDIV[3:0]**：入力クロック分周回路

APBクロックは、CANサブシステムで使用する前に分周できます。レートは、分周回路出力クロックを使用して計算する必要があります。

0000：1分周
0001：2分周
0010：4分周
0011：6分周
0100：8分周
0101：10分周
0110：12分周
0111：14分周
1000：16分周
1001：18分周
1010：20分周
1011：22分周
1100：24分周
1101：26分周
1110：28分周
1111：30分周

これらは書込み保護（P）ビットで、CCCRレジスタのビット1 [CCE] およびビット0 [INIT] が1にセットされているときのみ、このビットによる書込みアクセスが可能です。

70.4.38 FDCANレジスタマップ

表724.FDCANレジスタのマップとリセット値(1)

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	FDCAN_CREL	REL[3:0]				STEP[3:0]				SUBSTEP [3:0]				YEAR[3:0].				MON[7:0]								DAY[7:0]							
	リセット値	0	0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	1	1	0	1	1	1	1	1	0	1	0	0	0	1	0
0x0004	FDCAN_ENDN	ETV[31:0]																															
	リセット値	1	0	0	0	0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1
0x0008	予約済み	予約済み																															
0x000C	FDCAN_DBTP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]					Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0].			
	リセット値									0	0	0	0	0	0	0	0				0	1	0	1	0	0	0	1	1	0	0	1	1
0x0010	FDCAN_TEST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX [1:0]		LBCK	Res.	Res.	Res.	Res.
	リセット値																									0	0	0	0				
0x0014	FDCAN_RWD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDV[7:0]								WDC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	FDCAN_CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INT
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x001C	FDCAN_NBTP	NSJW[6:0]								NBRP[8:0]								NTSEG1 [7:0]								Res.	NTSEG2 [6:0]						
	リセット値	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	1
0x0020	FDCAN_TSCC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	FDCAN_TSCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	FDCAN_TOCC	TOP[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS [1:0]		ETOC
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	FDCAN_TOCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOC[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0030 - 0x003C	予約済み	予約済み																															
0x0040	FDCAN_ECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]								RP	REC[6:0]							TEC[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	FDCAN_PSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]							Res.	PXE	REDL	RBRSRESI1	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

表724.FDCANレジスタのマップとリセット値(1)（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0048	FDCAN_TDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	リセット値																		0	0	0	0	0	0		0	0	0	0	0	0	0	0
0x004C	予約済み	予約済み																															
0x0050	FDCAN_IR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	TOO	MRAF	TSW	TEFL	TEFF	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1N	RF0L	RF0F	RF0N
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	FDCAN_IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1NE	RF0LE	RF0FE	RF0NE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0058	FDCAN_ILS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERR	BERR	MISC	TFERR	SMSG	RXFIFO1	RXFIFO0
	リセット値	0																									0	0	0	0	0	0	0
0x005C	FDCAN_ILE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
	リセット値																															0	0
0x0060 - 0x007C	予約済み	予約済み																															
0x0080	FDCAN_RXGFC	Res.	Res.	Res.	Res.	LSE[3:0]				Res.	Res.	Res.	LSS[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	F0OM	F1OM	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
	リセット値					0	0	0	0				0	0	0	0	0							0	0			0	0	0	0	0	0
0x0084	FDCAN_XIDAM	Res.	Res.	Res.	EIDM[28:0]																												
	リセット値				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0088	FDCAN_HPMS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLST	Res.	Res.	FIDX[4:0]					MSI [1:0]		Res.	Res.	Res.	BIDX [2:0]		
	リセット値																	0			0	0	0	0	0	0	0				0	0	0
0x0090	FDCAN_RXF0S	Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	Res.	Res.	Res.	Res.	F0PI [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	F0GI [1:0]		Res.	Res.	Res.	Res.	F0FL [3:0]			
	リセット値							0	0							0	0							0	0					0	0	0	0
0x0094	FDCAN_RXF0A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [2:0]		
	リセット値																														0	0	0
0x0098	FDCAN_RXF1S	Res.	Res.	Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.	Res.	Res.	Res.	F1PI [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	F1GI [1:0]		Res.	Res.	Res.	Res.	F1FL [3:0]			
	リセット値							0	0							0	0							0	0					0	0	0	0
0x009C	FDCAN_RXF1A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [2:0]		
	リセット値																														0	0	0
0x00A0 - 0x00BC	予約済み	予約済み																															

表724.FDCANレジスタのマップとリセット値(1)（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00C0	FDCAN_TXBC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0	0																							
0x00C4	FDCAN_TXFQS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	Res.	Res.	Res.	TFQPI[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	TFGI[1:0]		Res.	Res.	Res.	Res.	Res.	TFFL[2:0]		
	リセット値											0				0	0							0	0						0	1	1
0x00C8	FDCAN_TXBRP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRP2	TRP1	TRP0
	リセット値																														0	0	0
0x00CC	FDCAN_TXBAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR2	AR1	AR0
	リセット値																														0	0	0
0x00D0	FDCAN_TXBCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CR2	CR1	CR0
	リセット値																														0	0	0
0x00D4	FDCAN_TXBTO	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TO2	TO1	TO0
	リセット値																														0	0	0
0x00D8	FDCAN_TXBCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CF2	CF1	CF0
	リセット値																														0	0	0
0x00DC	FDCAN_TXBTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE2	TIE1	TIE0
	リセット値																														0	0	0
0x00E0	FDCAN_TXBCIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFIE2	CFIE1	CFIE0
	リセット値																														0	0	0
0x00E4	FDCAN_TXEFS	Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	Res.	Res.	Res.	EFPI[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	EFG[1:0]		Res.	Res.	Res.	Res.	Res.	EFFL[2:0]		
	リセット値							0	0							0	0							0	0						0	0	0
0x00E8	FDCAN_TXEFA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[1:0]	
	リセット値																															0	0
0x0100	FDCAN_CKDIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PDIV[3:0]			
	リセット値																													0	0	0	0

1. R = 読出し、S = 読出し時にセット、X = 読出し時にリセット、W = 書込み、P = 書込み保護、p = セット保護、C = 書込み時にクリア／プリセット

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

71 ユニバーサルシリアルバスフルスピードホスト/デバイスインタフェース（USB）

このセクションは、STM32U535/545デバイスにのみ適用されます。

71.1 概要

USBペリフェラルは、フルスピードのUSB 2.0バスと APB2 バスの間のインタフェースを実装します。

USBサスペンド/レジュームがサポートされるため、節電のためにデバイスのクロックを停止できます。

71.2 USBの主な機能

- USB仕様バージョン2.0フルスピード準拠
- ホストモードとデバイスモードの両方に対応
- 1から8まで設定可能なエンドポイント数
- 2048バイトの専用パケットバッファメモリ（SRAM）
- 巡回冗長検査（CRC）の生成／チェック、NRZI（Non-return-to-zero Inverted）方式のエンコード／デコード、およびビットスタッフィング
- アイソクロナス転送サポート
- ダブルバッファバルク/アイソクロナスエンドポイント/チャネルに対応
- USBサスペンド／レジューム操作
- フレームロッククロックパルス生成
- USB 2.0リンク電源管理サポート（デバイスモードのみ）
- バッテリ充電仕様リビジョン1.2のサポート（デバイスモードのみ）
- USB接続／切断機能（USB DPライン上の制御可能な組み込みプルアップレジスタ）

71.3 USBの実装

表725に、デバイスでのUSB実装を示します。

表725.STMU535/545でのUSB実装

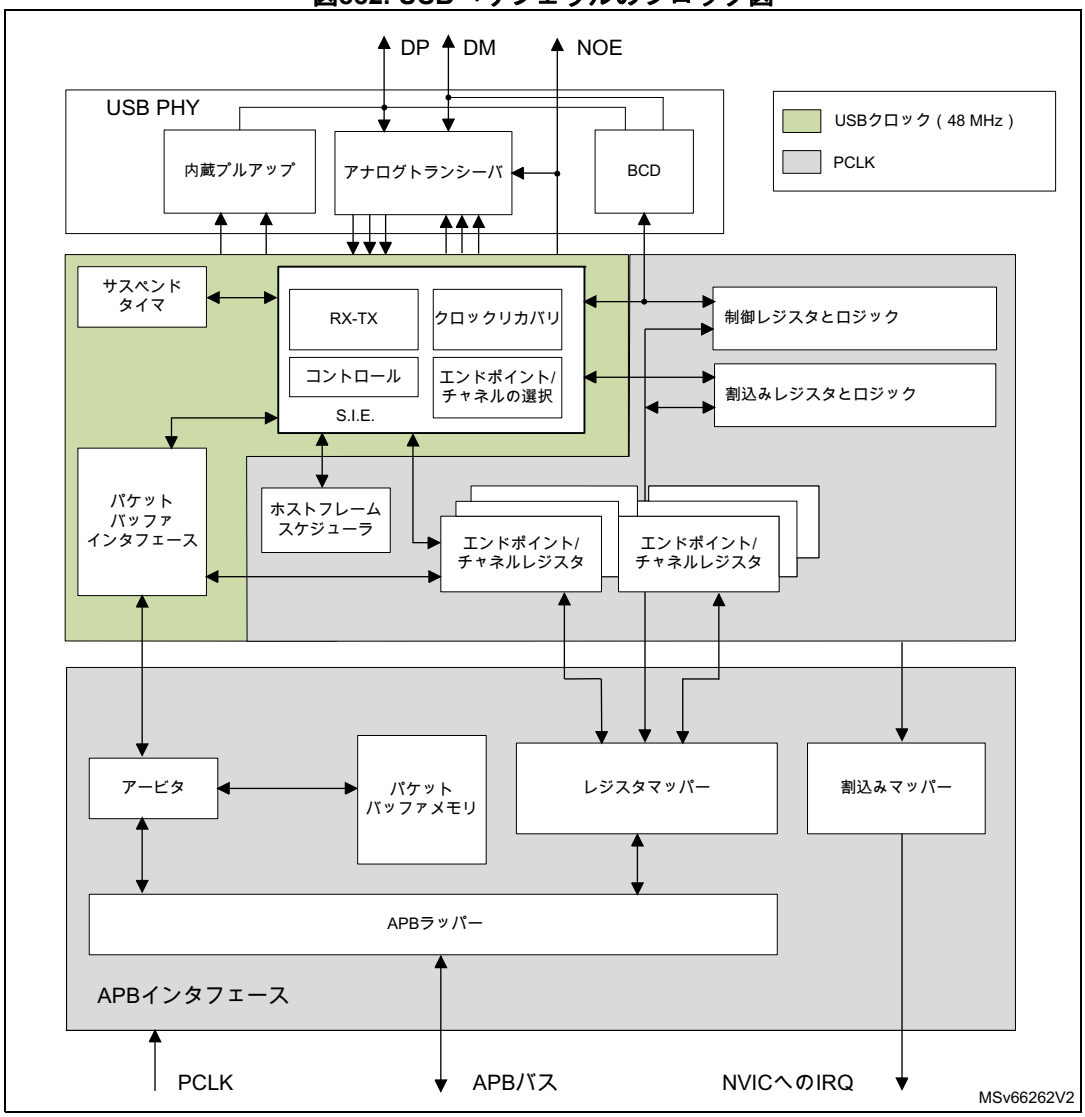
USBの機能(1)	USB
ホストモード	X
エンドポイント数	8
専用パケットバッファメモリSRAMのサイズ	2048バイト
専用パケットバッファメモリSRAMのアクセス方式	32bit
デバイス内でのUSB 2.0リンク電源管理（LPM）のサポート	X
デバイス用のバッテリ充電検出（BCD）のサポート	X
USB_DPライン上の組み込みプルアップレジスタ	X

1. X = サポート。

71.4 USBの機能説明

図882に、USBペリフェラルのブロック図を示します。

図882. USBペリフェラルのブロック図



概要とデバイスモードの機能

USBペリフェラルを使用すると、マイクロコントローラによって実装される機能と、ホストPCとUSBデバイスを兼ねる外部USB機能との間のUSB準拠接続が可能になります。外部USBホストまたはデバイスとシステムメモリ間のデータ転送は、USBペリフェラルによって直接アクセスされる専用パケットバッファメモリを通じて行われます。この専用メモリのサイズは2048バイトであり、最大16の単方向または8つの双方向エンドポイントを使用できます。USBペリフェラルは 外部USBホストまたはデバイスとインタフェースして、トークンパケットを検出し、データの送受信を処理し、USB標準の要求に従ってハンドシェイクパケットを処理します。トランザクションのフォーマット化は、CRCの生成とチェックも含めて、ハードウェアによって行われます。

各エンドポイント/チャネルは、エンドポイント/チャネル関連のメモリ領域の位置、その大きさ、または送信バイト数を示すバッファ記述ブロックに関連付けられています。有効な機能/エンドポイント

のペアのトークンがUSBペリフェラルによって検出されると、関連するデータ転送が行われます（必要であり、エンドポイント/チャネルが設定されている場合）。USBペリフェラルによってバッファされたデータは、内部16bitレジスタに配置され、専用バッファへのメモリアクセスが行われます。すべてのデータが転送されると、必要な場合には、転送方向に従って、適切なハンドシェイクパケットがUSB経由で生成され、または転送の方向によって期待されます。

トランザクションの終了時には、エンドポイント/チャネル固有の割込みが生成され、ステータスレジスタを読み出すか、別の割込み応答ルーチンを使用します。マイクロコントローラは、以下を決めることができます。

- 処理するエンドポイント/チャネル
- エラーが発生した場合に行うトランザクションのタイプ（ビットスタッフィング、フォーマット、CRC、プロトコル、ACKの欠如、オーバーラン／アンダーランなど）

アイソクロナス転送と高スループットバルク転送に対する特別なサポートが提供され、ダブルバッファ使用が実装されることにより、USBペリフェラルが使用できるバッファが常にあり、マイクロコントローラ用のバッファが別にあります。

レジスタUSB_ISTRの特別ビットTHR512を使用すると、512バイトの通知をバッファとの間で送受信することができます。このビットは、データの早期開始または読出し/書込みを促進する働きを持つため、長いISOパケットで（512 ～ 1023バイト）使用する必要があります。このように、最初の512バイトは、ダブルバッファモードの使用を回避しながら、ソフトウェアで処理することができます。このビットは、1つのISOエンドポイントのみが設定されている場合に有効です。

必要なときには、制御レジスタに書き込むことによって、ユニットを低電力モード（SUSPENDモード）にできます。このとき、すべての静的消費電流が回避され、USBクロックをスローダウンまたは停止できます。低電力モード中にUSB入力でアクティビティが検出されると、デバイスは非同期でウェイクアップします。特殊な割込みソースをウェイクアップラインに直接接続することにより、システムは通常のクロック生成をただちに再開し、直接クロックの開始/停止をサポートできます。

ホストモードと専用機能

レジスタUSB_CNTRの単一のHOSTビット を使用して、ホストモードを有効化できます。ホストモード機能により、USBはリモートペリフェラルと通信できます。サポートされる機能は、デバイスモードに対応し、同じレジスタ構造を使用してバッファを管理します。ホストモードでは同数のエンドポイントをサポートできますが、ホストモードでは「チャネル」という用語がよく使用されます。実際、各チャネルは接続されたデバイスとそのデバイス上のエンドポイントの組み合わせであるためです。パケット送受信用の基本メカニズムは、デバイスモードでサポートされているメカニズムと同様です。

ホストモードでの動作時、USBにはバスの役割があり、その役割を果たすために、有効な周期的/非周期的エンドポイントに対応したトランザクションリクエストを発行する必要があります。ホストフレームスケジューラでは、フレームを効率的に使用することが前提となります。ハブへの接続がサポートされています。直接接続とハブ経由の接続の両方で、ロースピードデバイスへの接続が可能です。

デバイスモードの項で説明したように、バルクチャネルとアイソクロナスチャネルの両方で、ダブルバッファモードもホストモードでサポートされています。THR512機能も（ただし、デバイスモードと同様に）ISOトラフィックに対してのみサポートされています。

注： **デバイスモードではバッテリ充電機能が検出されますが（高速充電を促進するために）、ホストモードではバッテリ充電容量（標準のCDPまたはDCPケース）を示すための統合サポートはなく、ホストポートが常にデフォルトの標準データポート（SDP）として提示されます。**

注： **LPM（リンク電源管理）の場合、ホストモードではこの機能がサポートされていません。**

71.4.1 デバイスモードとホストモードの両方で使用されるUSBブロックの説明

USBペリフェラルはUSBインタフェースに関するすべての機能を実装し、これには以下のブロックが含まれます。

- USB物理インタフェース（USB PHY）：このブロックは、外部USBホストへの電気インタフェースを維持します。ディファレンシャルアナログトランシーバ本体、制御可能な組み込みプルアップレジスタ（USB_DPラインに接続）、およびバッテリ充電検出（BCD）のサポート、同じUSB_DPおよびUSB_DMラインでの多重化を含みます。アナログトランシーバの出力有効化制御信号（アクティブロー）は、USB_NOEで外部的に提供されます。これを使用して、アクティビティLEDを駆動したり、実際の通信方向に関する情報を他の回路に提供することができます。
- シリアルインタフェースエンジン（SIE）：このブロックの機能には、同期パターン認識、ビットスタッフィング、CRC生成およびチェック、PID確認/生成、およびハンドシェイク評価が含まれます。USBトランシーバとインタフェースし、ローカルデータストレージ用のパケットバッファインタフェースによって提供される仮想バッファを使用する必要があります。このユニットは、フレーム開始（SOF）、USB_Reset、データエラーなどのUSBペリフェラルイベントと、送信終了、パケットの正しい受信などのエンドポイント関連イベントに従って信号を生成し、これらの信号は割込みの生成に使用されます。
- タイマ：このブロックは、フレーム開始ロッククロックパルスを生成し、トラフィックが3msの間、受信されなかったときにはグローバルサスペンドを（ホストから）検出します。
- パケットバッファインタフェース：このブロックは、ローカルメモリを管理して、送信と受信の両方のためのバッファのセットを柔軟に実装します。SIEから着信したリクエストに従って適切なバッファを選び、エンドポイント/チャネルレジスタによって示されるメモリアドレスに配置します。パケットの終わりまで、各バイトの交換後にアドレスをインクリメントして、交換バイト数を追跡し、バッファが最大容量までオーバーランするのを防ぎます。
- エンドポイント/チャネル関連レジスタ：各エンドポイント/チャネルには、エンドポイント/チャネルのタイプと現在のステータスを含んだレジスタが関連付けられます。単方向／シングルバッファエンドポイントの場合、1つのレジスタで2つのエンドポイントを実装できます。レジスタの数は8つであり、最大16の単方向／シングルバッファまたは最大7つのダブルバッファエンドポイントを任意の組み合わせで使用できます。たとえば、USBペリフェラルをプログラムして、4つのダブルバッファエンドポイントと8つのシングルバッファ／単方向エンドポイントを使用できます。
- 制御レジスタ：これらはUSBペリフェラル全体のステータスに関する情報を含むレジスタであり、レジュームやパワーダウンなどのUSBイベントを強制するために使用されます。
- 割込みレジスタ：これらは割込みマスクとイベントのレコードを含みます。割込み理由、割込みステータスの問い合わせや、保留中の割込みのステータスをクリアするために使用できます。

注： *** エンドポイント/チャネル0は、常にシングルバッファモードでの転送の制御に使用されます。**

USBペリフェラルはAPB2バスにAPB2インタフェースを通じて接続され、以下のブロックを含みます。

- パケットメモリ：これは、パケットバッファを物理的に含むローカルメモリです。パケットバッファインタフェースによって使用して、データ構造を作成し、アプリケーションソフトウェアから直接アクセスできます。パケットメモリのサイズは、2048バイトであり、32bitの512個のワードで構成されます。
- アービタ：このブロックは、APB2バスおよびUSBインタフェースから着信するリクエストを受け入れます。APB2アクセスに優先順位を付けることによってコンフリクトを解決するとともに、常にメモリ帯域幅の半分をすべてのUSB転送を完了するために確保しています。このタイムデュプレックス方式によって仮想デュアルポートSRAMが実装され、USBトランザクション中のメモリアクセスを可能にします。この方式により、任意の長さのマルチワードAPB2転送も可能です。
- レジスタマッパー：このブロックは、USBペリフェラルのさまざまなバイト長およびビット長のレジスタをAPB2によってアドレス指定され構造化された32bit長のワードセットに集めます。

- APB2ラッパー：これは、メモリとレジスタに APB2 へのインタフェースを提供します。また、USBペリフェラル全体を APB2 アドレス空間にマップします。
- 割込みマッパー：このブロックは、USB イベントによって割込みを生成する方法を選択し、それらをNVICにマップするために使用されます。

71.4.2 ホストモードに固有のホストフレームスケジューラ（HFS）の説明

ホストフレームスケジューラは、USBの優先順位とバンド幅のアクセス規則に従ってバスのホストチャネルリクエストを送信する働きを持つハードウェアマシンです。

ホストチャネルは次の2つのカテゴリに分類されます。

- 周期的チャネル：アイソクロナスおよび割込みトラフィックタイプ。保証されたバンド幅アクセスを使用します。
- 非周期的チャネル：バルクおよび制御トラフィックタイプ。ベストエフォート型サービスを使用します。

ホストフレームスケジューラは、次の3つのシーケンシャルウィンドウでフルスピードフレームを構成します。

- 周期的サービスウィンドウ
- 非周期的サービスウィンドウ
- ブラックセキュリティウィンドウ

新しいフレームの開始時にホストスケジュールは次のように機能します。

1. まず、フレームの開始時に有効だった（STAT ビットが VALID）すべての周期的チャネルについて検討します。
2. 周期的チャネルのサービスの単一ラウンドである周期的サービスウィンドウを、CH#1 から CH#8の優先順位で実行します。双方向チャネルの場合、OUT方向を最初に実行します。
3. 周期的ラウンドが終了すると、一部の周期的チャネルが最有効化された場合や、SOF の後に一部の新しいチャネルが有効化された場合でも、HFSによって周期的サービスウィンドウが閉じられ、周期的トラフィックのサービスが停止されます。
4. CH#1からCH#8のハードウェア優先順位で現在有効になっている（STATビットがVALID）非周期的チャネルのすべてのサービスを開始します。双方向チャネルの場合、OUT方向を最初に実行します。
5. ほぼフレームの終了に達するまで、非周期的チャネルの複数のラウンドロビンサービスサイクルを実行します。
6. 非周期的トラフィックはいつでもリクエストでき、HFSによりベストエフォートの遅延でサービスされます。ただし、バブルを避けるために新しいインジェクトリクエストが次のフレームまで直接延期されるフレームの最後のブラックセキュリティウィンドウは例外です。これは、セキュリティウィンドウの前にサービスされていない保留中のトランザクションにも適用されます。

71.5 デバイスモードおよびホストモードでのプログラミングの考慮事項

以下のセクションでは、アプリケーションソフトウェア開発を容易にするために、USBペリフェラルとアプリケーションプログラムの間の予期される相互作用について説明します。

71.5.1 汎用USBデバイスのプログラミング

このパートでは、USB準拠動作を得るためにアプリケーションソフトウェアに必要な主なタスクについて説明します。ほとんどの汎用USBイベントに関連するアクションが考慮に入れられており、ダブルバッファエンドポイントとアイソクロナス転送という特殊ケースについて、それぞれのパラグラフで説明します。システムリセットを別にすると、アクションは常にUSBペリフェラルによって開始され、以下に述べるUSBイベントの1つによって駆動されます。

71.5.2 システムリセットとパワーオンリセット

システムリセットおよびパワーオンリセットの際、アプリケーションソフトウェアが最初に実行すべき操作は、必要なクロック信号のすべてをUSBペリフェラルに与えることであり、その後、リセット信号をネゲートして、レジスタにアクセスできるようにしなければなりません。初期化シーケンス全体は、以下のとおりです。

最初のステップとして、アプリケーションソフトウェアは、デバイスクロック管理ロジックによって提供される関連の制御ビットを使用して、レジスタマクロセルクロックを有効にし、マクロセル固有のリセット信号をネゲートする必要があります。

その後、CNTRレジスタのPDWNビットを使用して、USBトランシーバに関連するデバイスのアナログ部分をオンにする必要があり、これには特別な処理が必要です。このビットは、ポートトランシーバに供給される内部電圧基準をオンにすることを目的としています。この回路にはスタートアップ時間が定義されており（データシートで指定されている$t_{STARTUP}$）、この間のUSBトランシーバの動作は定義されていません。したがって、CNTRレジスタのPDWNビットをセットした後、この時間だけ待つ必要があり、その後、USB部分のリセット条件を解除します（CNTRレジスタのUSBRSTビットをクリアすることによって）。ISTRレジスタをクリアすると、他のマクロセル動作が有効になる前に、誤って保留中の割込みが削除されます。

システムリセット時には、マイクロコントローラは必要なすべてのレジスタとパケットバッファ記述テーブルを初期化して、USBペリフェラルが割込みとデータ転送を正しく生成できるようにする必要があります。エンドポイント/チャネル固有ではないレジスタはすべて、アプリケーションソフトウェアの必要に応じて初期化する必要があります（有効にする割込みの選択、パケットバッファのアドレスの選択など）。USBリセットの場合、その後、プロセスが続行します（以下のパラグラフを参照）。

デバイスモードでのUSBバスリセット（RST_DCON割込み）

このイベントが発生すると、USBペリフェラルは、前のパラグラフで説明した初期化後のシステムリセットと同じ条件になります。すなわち、すべてのエンドポイントレジスタで通信が無効になります（USBペリフェラルはパケットに応答しません）。USBリセットイベントに対して、USB機能を有効にして、USBアドレスを0にし、デフォルトのコントロールエンドポイントのみを実装します（エンドポイントアドレスも0）。これは、USB_DADDRレジスタの機能有効（EF）ビットをセットし、CHEP0Rレジスタと関連するパケットバッファを初期化することによって行われます。USBエニュメレーションプロセス時、ホストは一意なアドレスをこのデバイスに割り当て、これがUSB_DADDRレジスタのADD[6:0] ビットに書き込まれなければならず、他に必要なエンドポイントを設定する必要があります。

RST_DCON割込みが受信されると、アプリケーションソフトウェアは、割込みをトリガしたリセットシーケンスの終了から10ms以内に、USB機能0のデフォルトのエンドポイントを再び有効にする必要があります。

ホストモードでのUSBバスリセット

ホストモードでバスリセットを有効化するには、USB_CNTRレジスタのUSBRSTビットをセットします。標準からの最小有効リセット時間が順守されると、ソフトウェアによってクリアされます。

パケットバッファの構造と用途

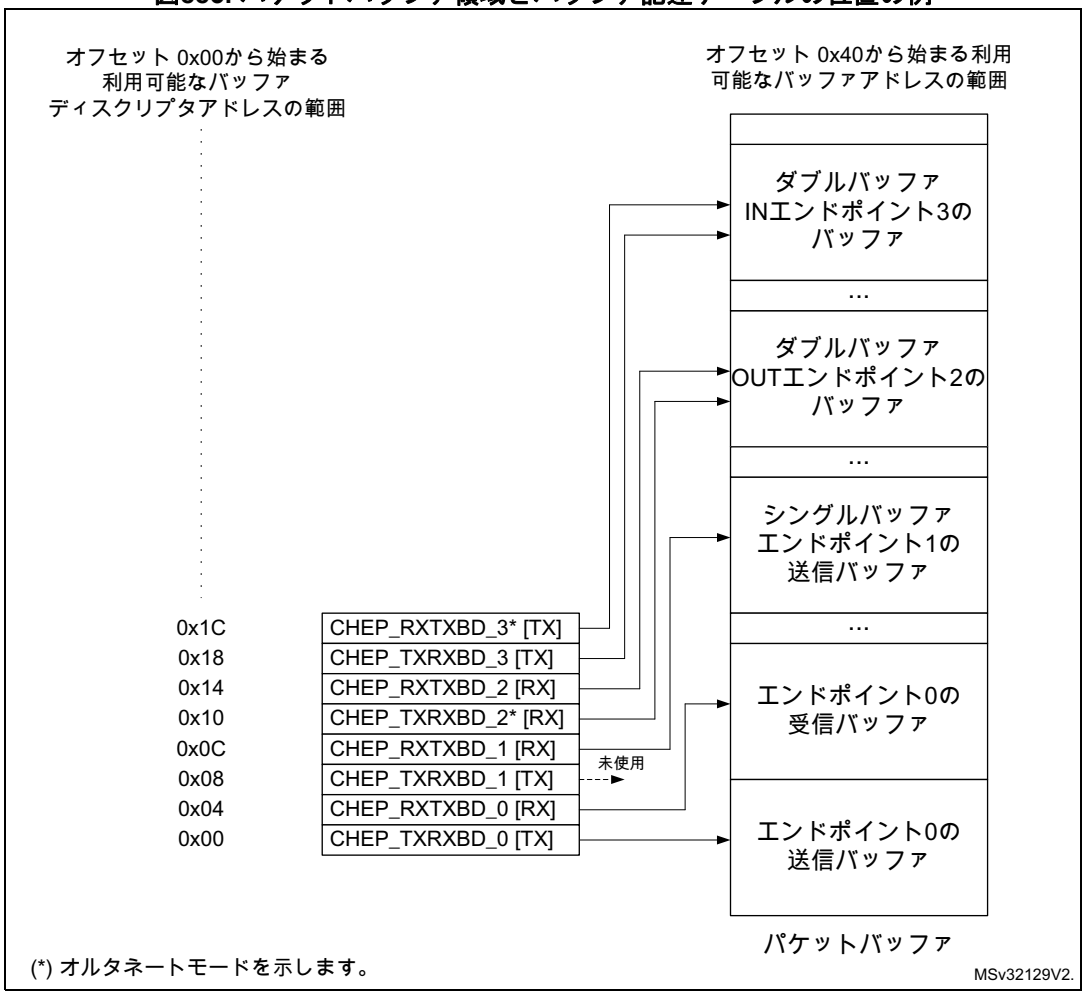
各双方向エンドポイントは、バス上でデータを送受信できます。受信データは、そのエンドポイント用に予約された専用のメモリバッファに格納され、別のメモリバッファにはエンドポイントによって送信されるデータが格納されます。このメモリへのアクセスは、パケットバッファインタフェースブロックによって行われ、これはメモリアクセスリクエストを配信し、確認応答を待ちます。マイクロコントローラもパケットバッファメモリにアクセスする必要があるので、アクセスコンフリクトを避けるためにアービトレーションロジックが使用され、APB2 サイクルの半分がマイクロコントローラのアクセスに使用され、残りの半分がUSBペリフェラルのアクセスに使用されます。このように、パケットメモリがデュアルポートSRAMであるかのように両方のエージェントが動作でき、マイクロコントローラが連続アクセスを実行中でも、コンフリクトを気にする必要がありません。USBペリフェラルロジックは、専用クロックを使用します。この専用のクロックの周波数はUSB標準の要件により、48MHzに固定され、これは APB2 バスのインタフェース用のクロックとは異なることがあります。APB2クロックの周波数がUSBペリフェラルの周波数よりも高くなる、または低くなる可能性がある場合、別のクロック設定を使用することができます。

注： **USBデータレートとパケットメモリインタフェース要件により、APB2　クロックは、データのオーバーラン／アンダーラン問題を回避するために、少なくとも 12 MHz以上の周波数でなければなりません。**

各エンドポイントには2つのパケットバッファが関連付けられます（通常は、送信用に1つ、受信用に1つ）。バッファの位置とサイズはバッファ記述テーブルで指定され、このテーブルもパケットメモリ内に配置されるので、バッファはパケットメモリ内の任意の場所に配置できます。各テーブルエントリがエンドポイントレジスタに関連付けられ、2つの32bitワードで構成されるため、テーブル開始アドレスは常に8バイトの境界に揃える必要があります。バッファディスクリプタテーブルのエントリについては、セクション 71.6.2 : バッファディスクリプタテーブルで説明します。エンドポイントが単方向であり、アイソクロナスでもダブルバッファバルクでもない場合、必要なパケットバッファは1つだけです（サポートされる転送方向に応じた1つのバッファ）。サポートされない転送方向または未使用のエンドポイントに関連するテーブル位置は、ユーザが使用できます。アイソクロナスおよびダブルバッファバルクエンドポイントには、パケットバッファの特別な取り扱いがあります（それぞれ、セクション 71.5.5 : デバイスモードでのアイソクロナス転送 および セクション 71.5.3 : ダブルバッファエンドポイントとデバイスモードでの使用を参照）。バッファ記述テーブルのエントリとパケットバッファ領域の関係を 図883に示します。

ホストモードについては、別のセクション、特にセクション 71.5.6 : ホストモードでのアイソクロナス転送とセクション 71.5.4 : ダブルバッファチャネル : ホストモードでの使用でバッファ使用モデルについて説明します。

図883. パケットバッファ領域とバッファ記述テーブルの位置の例



各パケットバッファは、始めから終わりまで、受信または送信のいずれかに使用されます。USBペリフェラルは、割り当てられたメモリバッファに隣接するメモリ位置の内容を変えることはありません。割り当てられたバッファ長を超えるパケットが受信された場合（バッファオーバーラン条件）、使用可能な最期の位置までしか、データはメモリにコピーされません。

エンドポイントの初期化

エンドポイントを初期化するための最初のステップは、CHEP_TXBD_nおよびCHEP_RXBD_nレジスタ（SRAM内）のADDRn_TX/ADDRn_RXフィールドに適切な値を書き込んで、送信データがすでに使用可能になったことと、受信データをバッファできることをUSBペリフェラルに知らせることです。USB_CHEPnRレジスタのUTYPEビットをエンドポイントのタイプに応じてセットし、最終的にはEPKINDビットを使用して、特に必要な機能を有効にする必要があります。送信側では、USB_CHEPnRレジスタのSTATTXビットを使用してエンドポイントを有効にする必要があり、COUNTn_TXを初期化する必要があります。受信の場合、STATRXビットをセットして受信を有効にする必要があり、BLSIZEおよびNUM_BLOCKフィールドを使用して、割り当てられたバッファサイズをCOUNTn_RXに書き込む必要があります。単方向エンドポイントは、アイソクロナスおよびダブルバッファバルクエンドポイントを除き、サポートされる方向に関するビットとレジスタのみを初期化する必要があります。送信または受信が有効になると、レジスタUSB_CHEPnRと位置ADDRn_TX/ADDRn_RX、COUNTn_TX/COUNTn_RX（それぞれ）をアプリケーションソフトウェアで変更しないでください。これらの値はハードウェアが直接変更できます。データ転送操作が完了し、CTR割込みイベントによって通知されると、再びアクセスできるようになり、新しい操作を再び有効にできます。

デバイスモードでのデータ送信（INパケット）

INトークンパケットを受信するとき、受信したアドレスが設定済みの有効なエンドポイントと一致した場合、USBペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内のCHEP_TXBD_n（フィールドADDRn_TXおよびCOUNTn_TX）にアクセスします。これらの位置の内容は、内部16bitレジスタADDRおよびCOUNTに格納されます（ソフトウェアからはアクセスできません）。パケットメモリへのアクセスが再度行われて、送信する最初のバイトが読み出され（3056ページのパケットバッファの構造と用途を参照）、USBペリフェラルが、USB_CHEPnRのDTOGTXビットに従ってDATA0またはDATA1 PIDの送信を開始します。PIDが完了すると、バッファメモリから読み出した最初のバイトが、USBバスで送信される出力シフトレジスタにロードされます。最後のデータバイトが送信された後に、計算されたCRCが送信されます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_CHEPnRレジスタのSTATTXビットに従って、NAKまたはSTALLハンドシェイクパケットがデータパケットの代わりに送信されます。

内部レジスタCHEP_TXBD_nのADDRn_TXフィールドは、現在のバッファメモリ位置を指すポインタとして使用され、COUNTは、残りの送信バイト数をカウントするために使用されます。パケットバッファメモリから読み出された各ハーフワードは、最下位バイトから順にUSBバスで送信されます。送信バッファメモリは、COUNTn_TX/4ワードの場合、ADDRn_TXによって示されるアドレスから順に読み出されます。送信パケットが奇数個のバイトで構成される場合、最後にアクセスされたハーフワードの下位半分のみが使用されます。

ホストによって受信されるACKの受信時に、USB_CHEPnRレジスタが更新されます。つまり、DTOGRXビットがトグルされ、STATRX = 10（NAK）をセットすることでエンドポイントが無効化され、VTTXビットがセットされます。アプリケーションソフトウェアは、まず、USB_ISTRレジスタのIDNおよびDIRビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。VTTXイベントの処理は、割込みビットをクリアすることから始まります。次に、アプリケーションソフトウェアは、送信データでいっぱいの別のバッファを準備し、COUNTn_TXテーブル位置を次の転送時に送信されるバイト数で更新し、最後に、STATTXを11（VALID）にセットして、送信を再び有効にします。STATTXビットが10（NAK）に等しいときには、そのエンドポイントにアドレス指定されたINリクエストはNAKされ、フロー制御条件を示します。USBホストは、成功するまで、トランザクションを再試行します。CTR割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された2番目のINトランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。

ホストモードでのデータ送信（OUTパケット）

ホストモードでのデータ送信は、デバイスモードの場合と同じ一般原則に従います。主な違いは、プロトコルによるものです。たとえば、ホストは送信を開始しますが、デバイスは着信トークンに応答します。

パケットの送信用に予約されたパケットメモリ内の位置に、ADDRn_TXをセットする必要があります。これにより、OUTパケットの内容がパケットメモリ内のそのアドレスに書き込まれ、パケット内のバイト数を示すために（必要に応じて）COUNTn_TXを更新する必要があります。

適切なエンドポイント用にDEVADDRを書き込む必要があり、送信をトリガするためにSTATTXを11（VALID）にセットする必要があります。これにより、HFSによって送信がスケジュールされます。

送信が正常に行われると、CTR割込み（適切な転送）がトリガされます。IDNおよびDIRを検査することにより、該当するチャネルと方向が特定できます。示されたチャネルで、STATTXフィールドはDISABLEに遷移しています。NAKを受信している場合（ペリフェラルがレディ状態でないとき）、STATTXはNAKになっています。STALL応答の場合、STATTXはSTALLになっています。この最後の場合、バスをリセットする必要があります。

デバイスがACKを受信した場合、USB_CHEPnRレジスタのDTPGTXビットがトグルされます。

次の場合、ビットVTTXおよびERR_TXを通じてエラー条件が通知されます。

- ハンドシェイクを時間内に受信しない場合
- 偽のEOP
- ビットスタッフィングエラー
- 無効なハンドシェイクPID

デバイスモードでのデータ受信（OUTおよびSETUPパケット）

この2つのトークンは、USBペリフェラルによっておおむね同じように扱われます。SETUPパケットの取り扱いにおける違いについては、制御転送に関する以下のパラグラフで詳しく述べます。OUT/SETUP PIDを受信するとき、アドレスが有効なエンドポイントと一致した場合、USBペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内のADDRn_RXおよびCOUNTn_RXフィールドの内容にアクセスします。ADDRn_RXフィールドの内容は、内部レジスタADDRに直接格納されます。内部レジスタCOUNTがリセットされ、BLSIZEおよびNUM_BLOCKビットフィールドの値がUSB_CHEP_RXBD_n内に読み出されると、それを使用してBUF_COUNTが初期化され、これは、バッファオーバーラン条件のチェックに使用される内部16bitカウンタです（これらの内部レジスタはすべて、ソフトウェアからはアクセスできません）。その後、USBペリフェラルによって受信されるデータバイトは、ハーフワード単位でパケット化され（最初の受信バイトが最下位バイトとして格納されます）、内部ADDRレジスタに含まれるアドレスから順にパケットバッファに転送され、各バイト転送により、BUF_COUNTはデクリメントされ、COUNTはインクリメントされます。DATAパケットの終わりが検出されると、受信されたCRCの正しさがテストされ、受信時にエラーが発生しなかった場合のみ、ACKハンドシェイクパケットが送信元ホストに送り返されます。

CRCが正しくなかった場合、または他の種類のエラー（ビットスタッフ違反、フレームエラーなど）が発生した場合、データバイトは、少なくともエラー検出ポイントまではパケットメモリにコピーされますが、ACKパケットは送信されず、USB_ISTRレジスタのERRビットがセットされます。ただし、この場合、通常はソフトウェアのアクションは不要です。USBペリフェラルは受信エラーから回復し、次のトランザクションを処理する準備ができた状態を保ちます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_CHEPnRレジスタのSTATRXビットに従って、NAKまたはSTALLハンドシェイクパケットがACKの代わりに送信され、データは受信メモリバッファに書き込まれません。

受信メモリバッファの位置は、ADDRn_RXに含まれるアドレスから受信データパケット長に対応するバイト数だけ、または、BLSIZEおよびNUM_BLOCKによって定義された割り当て済みの最後のメモリ位置のいずれか先に来る方まで書き込まれます。このように、USBペリフェラルは、割り当てられた受信メモリバッファ領域の終わりを超えて書き込むことはありません。データパケットペイロードの長さ（アプリケーションによって使用される実際のバイト数）が割り当てられたバッファを超える場合、USBペリフェラルによってバッファオーバーラン条件が検出されます。この場合、通常のACKの代わりにSTALLハンドシェイクが送信されて問題がホストに通知され、割込みは生成されず、トランザクションはエラーとみなされます。

ACKハンドシェイクパケットを送信することにより、トランザクションが正常に完了すると、内部COUNTレジスタがバッファ記述テーブルエントリ内のCOUNTn_RX位置にコピーされ、影響を受けなかったBLSIZEおよびNUM_BLOCKフィールドはそのままに保たれ（通常は再書込みの必要はありません）、USB_EPnRレジスタが次のように更新されます。つまり、DTOGRXビットがトグルされ、STATRX = 10（NAK）をセットすることでエンドポイントが無効化され、VTRXビットがセットされます。エラーまたはバッファオーバーランによってトランザクションが失敗した場合、上記のアクションはいずれも実行されません。アプリケーションソフトウェアは、まず、USB_ISTRレジスタのIDNおよびDIRビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。VTRXイベントの処理は、まず、トランザクションのタイプを決定することによって行われます（USB_CHEPnRレジスタのSETUPビット）。アプリケーションソフトウェアは、割込みフラグビットをクリアして、処理されるエンドポイントに関連するバッファ記述テーブルエントリ内のCOUNTn_RX位置を読み出して、受信バイト数を取得する必要があります。受

信データが処理された後、アプリケーションはUSB_CHEPnRのSTATRXビットを11（VALID）にセットして、次のトランザクションを有効にする必要があります。STATRXビットが10（NAK）に等しいときには、そのエンドポイントにアドレス指定されたOUTリクエストはNAKされ、フロー制御条件を示します。USBホストは、成功するまで、トランザクションを再試行します。CTR割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された2番目のOUTトランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。

ホストモードでのデータ受信（INパケット）

ホストモードでのデータ受信は、デバイスモードの場合と同じ一般原則に従います。この場合も、主な違いはプロトコルによるものです。デバイスで、前の操作後の準備状況によって、データの受信が可能な場合と可能でない場合がありますが、ホストでは、準備が完了していてデータを格納できる場合のみ、受信データのリクエストをします。

パケットの送信用に予約されたパケットメモリ内の位置に、ADDRn_TXをセットする必要があります。INトーケンパケットへのデータフェーズ応答で受信した内容は、パケットメモリ内のアドレスに書き込まれ、パケット内のバイト数を示すため、このプロセス中にCOUNTn_TXがハードウェアによって更新されます。

適切なエンドポイント用にDEVADDRを書き込む必要があり、受信をトリガするためにSTATRXをVALIDにセットする必要があります。これにより、HFSによって受信がスケジュールされます。

受信が正常に行われると、割込みCTR（適切な転送）がトリガされます。IDNおよびDIRを検査することにより、該当するチャネルと方向が特定できます。示されたチャネルで、STATRXフィールドはDISABLEに遷移しています。NAKを受信している場合（ペリフェラルがレディ状態でないとき）、STATRXはNAKになっています。STALL応答の場合、STATRXはSTALLになっています。この最後の場合、バスをリセットする必要があります。次の場合、INパケット中にビットVTRXおよびERR_RXを通じてエラー条件が通知されます。

- 偽のEOP
- ビットスタッフィングエラー
- CRCの誤り

デバイスモードでの制御転送

制御転送は、SETUPトランザクション、すべて同じ方向のゼロ個以上のデータステージ、およびステータスステージ（反対方向のゼロバイトの転送）で構成されます。SETUPトランザクションは制御エンドポイントのみによって処理され、OUTトランザクション（データ受信）と非常に似ていますが、制御転送を初期化するために、アドレス指定されたエンドポイントレジスタのDTOGTXおよびDTOGRXビットの値がそれぞれ1と0にセットされる点と、ソフトウェアが後続のトランザクションがINかOUTかをSETUPの内容に応じて決定できるように、STATTXとSTATRXの両方が10（NAK）にセットされる点が異なります。制御エンドポイントはVTRXイベントごとにUSB_CHEPnRレジスタのSETUPビットをチェックして、通常のOUTトランザクションとSETUPトランザクションを区別する必要があります。USBデバイスは、SETUPステージで転送されたデータを解釈することによって、データステージの数と方向を判断でき、エラーの場合にはトランザクションをSTALLする必要があります。そのためには、最後以外のすべてのデータステージで、未使用の方向をSTALLにセットして、ホストが転送方向を反転するのが速過ぎた場合にはステータスステージとしてSTALLされるようにする必要があります。

最後のデータステージを有効にするときには、反対方向をNAKにセットして、ホストが（ステータスステージを実行するために）転送方向をすぐに反転した場合、制御操作の完了を待つようにする必要があります。制御動作が正常に完了した場合はソフトウェアによってNAKがVALIDに変更され、それ以外の場合はSTALLに変更されます。同時に、ステータスステージがOUTの場合は、ゼロ以外のデータでステータストランザクションが実行された場合にエラーが生成されるように、STATUS_OUT（USB_CHEPnRレジスタのEPKIND）ビットをセットする必要があります。ステータストランザクションが処理されると、アプリケーションはSTATUS_OUTビットをクリアし、STATRXをVALIDにセット

し（新しいコマンドを受け入れるため）、STATTXをNAKにセットします（次のセットアップの直後のステータスステージを遅延するため）。

USB仕様では、SETUPパケットにはACK以外のハンドシェイクでは応答できず、結果として、以前許可されていないアボートして新しいコマンドを開始すると規定されているため、USBロジックでは、コントロールエンドポイントはホストから受信したSETUPトークンに対してNAKまたはSTALLパケットで応答することはできません。

STATRXビットが01（STALL）または10（NAK）にセットされ、SETUPトークンが受信されると、USBはデータを受け入れ、要求されたデータ転送を実行し、ACKハンドシェイクを返送します。そのエンドポイントに以前に発行されたVTRXリクエストがあり、アプリケーションによってまだ確認応答されていない場合（例：VTRXビットが以前に完了した受信からセットされたままになっている場合）、USBはSETUPトランザクションを破棄し、状態にかかわらずハンドシェイクパケットで応答せず、受信エラーをシミュレートして、ホストがSETUPトークンを再び送信しなければならないようにします。これは、VTRX割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定されたSETUPトランザクションの通知が失われるのを避けるために行われます。

ホストモードでの制御転送

制御転送は、SETUPトランザクション、すべて同じ方向のゼロ個以上のデータステージ、およびステータスステージ（反対方向のゼロバイトの転送）で構成されます。SETUPトランザクションは、制御エンドポイントによってのみ処理されます。制御エンドポイントでUSB_CHEPnRレジスタのSETUPビットをセットする必要があります。アドレス指定されたエンドポイントレジスタのDTOGTXおよびDTOGRXビットの値が0にセットされます。制御書込みか制御読出しかによって、ホストフレームスケジューラを通じて制御転送をトリガするため、STATTXまたはSTATRXが11（ACTIVE）に設定されます。

CTR割込みの受信時に、IDNおよびDIRビットを検査することにより、チャネル（デバイスアドレスとエンドポイント）を特定できます。パケットに確認が不能になるような破損がない場合、制御のたびにデバイスでNAKがセットされます。この状態はSTATTXの値に反映されます。

エラーが発生している場合、ERRビットがセットされます。たとえば、デバイスでCRCエラーが発生している場合、ホストにACKは返されません。ホストがACKを受信しない場合、適切な遅延の後に、ERR_TXがセットされて（割込みを生成可能）タイムアウトエラーが生成されます。

71.5.3 ダブルバッファエンドポイントとデバイスモードでの使用

USB標準によって定義されているすべてのエンドポイントタイプは、さまざまなトラフィックモデルを表し、各種のデータ転送操作の典型的な要件を記述しています。データの大部分をホストPCとUSB機能の間で転送する必要があるときには、バルクエンドポイントタイプが最適のモデルです。ホストはバルクトランザクションをスケジュールして、フレーム内の使用可能な帯域幅のすべてをいっぱいにして、アドレス指定されたUSB機能がバルクトランザクションを処理できる限り、実際の転送速度を最大化します。次のトランザクションが着信したときに、USB機能が前のトランザクションでビジーであった場合は、NAKハンドシェイクで応答し、ホストPCは、USB機能が処理できるようになるまで、同じトランザクションを再び発行するので、再送信によって帯域幅が占有されるため、実際の転送速度は低下します。このため、バルクエンドポイントでは、「ダブルバッファリング」と呼ばれる専用機能を使用できます。

「ダブルバッファリング」が有効なときには、データトグルシーケンシングが使用されて、USBペリフェラルが要求されたデータ転送を実行するために使用するバッファが選択され、送信と受信の両方のパケットメモリ領域を使用して、トランザクションの成功ごとのバッファスワッピングを管理するため、アプリケーションは常に完全なバッファを使用でき、USBペリフェラルは他方のバッファをいっぱいにします。たとえば、「受信」ダブルバッファバルクエンドポイント宛てのOUTトランザクション時、1つのバッファがUSBホストから着信する新しいデータで満たされている間、もう1つの

バッファはマイクロコントローラソフトウェアが使用できます（「送信」ダブルバッファバルクエンドポイントとINトランザクションでも同様です）。

スワップバッファ管理には、アドレスポインタと割り当て済みメモリバッファの長さをホストする4つすべてのバッファ記述テーブル位置を使用する必要があるため、ダブルバッファバルクエンドポイントの実装に使用されるUSB_CHEPnRレジスタは、強制的に単方向として使用されます。したがって、1つのSTATビットペアのみを00（無効）以外の値にセットする必要があります。ダブルバッファバルクエンドポイントの受信が有効化されている場合はSTAT_RX、ダブルバッファバルクエンドポイントの送信が有効化されている場合はSTAT_TXとします。ダブルバッファバルクエンドポイントを受信と送信の両方について有効にする必要がある場合は、2つのUSB_CHEPnRレジスタを使用する必要があります。

ダブルバッファリング機能を利用して、可能な限り高い転送速度を達成するには、前の章で説明したエンドポイントフロー制御構造を変更して、トランザクションの正常官僚の旅ではなく、USBペリフェラルとアプリケーションの間でバッファコンフリクトが発生したときのみ、エンドポイントのステータスをNAKに切り替える必要があります。USBペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関連するDTOGビットによって定義されます。すなわち、「受信」ダブルバッファバルクエンドポイントの場合はDTOGRX（USB_CHEPnRレジスタのビット14）、「送信」ダブルバッファバルクエンドポイントの場合はDTOGTX（USB_CHEPnRレジスタのビット6）です。新しいフロー制御方式を実装するには、USBペリフェラルはアプリケーションソフトウェアによって使用されているパケットバッファを知り、コンフリクトに注意する必要があります。USB_CHEPnRレジスタには2つのDTOGビットがありますが、データおよびバッファシーケンシングのためにUSBペリフェラルによって使用されるのは1つだけであり（ダブルバッファリング機能に必要な単方向成約のため）、もう1つはアプリケーションソフトウェアが、使用中のバッファを示すために使用できます。この新しいバッファフラグはSW_BUFと呼ばれます。次の表で、「送信」ダブルバッファバルクエンドポイントと「受信」ダブルバッファバルクエンドポイントについて、USB_CHEPnRレジスタのビットとDTOG/SW_BUF定義の対応を説明します。

表726.ダブルバッファリングバッファフラグの定義

バッファフラグ	「送信」エンドポイント	「受信」エンドポイント
DTOG	DTOGTX（USB_CHEPnRビット6）	DTOGRX（USB_CHEPnRビット14）
SW_BUF	USB_CHEPnRビット14	USB_CHEPnRビット6

USBペリフェラルによって現在使用されているメモリバッファは、DTOGバッファフラグによって定義され、アプリケーションソフトウェアによって現在使用されているバッファは、SW_BUFバッファフラグによって示されます。バッファフラグの値と使用されているパケットバッファの関係は、どちらの場合も同じであり、次の表に示されています。

表727.バルクダブルバッファリングメモリバッファの用途（デバイスモード）

エンドポイントタイプ	DTOG	SW_BUF	USBペリフェラルによって使用されるパケットバッファ	アプリケーションソフトウェアによって使用されるパケットバッファ
送信（IN）	0	1	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置
	1	0	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
	0	0	なし (1)	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
	1	1	なし (1)	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
受信（OUT）	0	1	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	1	0	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	0	0	なし (1)	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	1	1	なし (1)	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。

1. NAKステータスのエンドポイント。

バルクエンドポイントのダブルバッファリング機能は、次のようにして有効化されます。

- USB_CHEPnRレジスタUTYPEビットフィールドに00を書き込み、エンドポイントをバルクとして定義します。
- 同じレジスタのEPKINDビットを1（DBL_BUF）にセットします。

アプリケーションソフトウェアは、最初に使用されるバッファに応じて、DTOGおよびSW_BUFビットを初期化する責任があります。これは、この2つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。DBL_BUFのセット後に発生した初めてのトランザクションの終了により、ダブルバッファバルクエンドポイントの特殊なフロー制御がトリガされ、DBL_BUFがセットされたままである限り、このエンドポイントにアドレス指定された他のすべてのトランザクションについて、これが使用されます。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイントUSB_CHEPnRレジスタのVTRXまたはVTTXビットがセットされます。同時に、USB_CHEPnRレジスタの影響を受けたDTOGビットがハードウェアによってトグルされ、USBペリフェラルのバッファスワッピングを完全にソフトウェアから独立させます。通常のトランザクション、およびDBL_BUFのセット後の初めてのトランザクションと違って、STATビットペアはトランザクション終了の影響を受けず、値は11（VALID）のままです。ただし、新しいトランザクションのトークンパケットが受信されると、実際のエンドポイントステータスは、USBペリフェラルとアプ

リケーションソフトウェアの間のバッファコンフリクトが検出されたときには10（NAK）としてマスクされます（この条件はDTOGとSW_BUFが同じ値を持つことによって示されます。3063ページの表 727を参照）。アプリケーションソフトウェアは、CTRイベント通知に応答して、割込みフラグをクリアし、完了したトランザクションの要求された処理を開始します。アプリケーションパケットバッファの使用が終了すると、ソフトウェアはSW_BUFビットに1を書き込んでトグルし、そのバッファが使用可能になったことをUSBペリフェラルに通知します。このように、NAKされるトランザクションの数は、トランザクションデータのアプリケーションエラボレーション時間によってのみ制限されます。エラボレーション時間がUSBバスでのトランザクションの完了に必要な時間より短かった場合、フロー制御による再送信は行われず、実際の転送速度はホストPCによってのみ制限されます。

アプリケーションソフトウェアは、11（VALID）とは異なる明示的なステータスを、関連するUSB_CHEPnRレジスタのSTATビットペアに書き込むことによって、ダブルバッファバルクエンドポイントのために実装される特殊なフロー制御をオーバーライドできます。この場合、USBペリフェラルは、バッファ使用条件に関係なく、プログラムされたエンドポイントステータスを常に使用します。

71.5.4 ダブルバッファチャネル：ホストモードでの使用

ホストモードのダブルバッファチャネルで使用される送受信方法は、デバイスモードで説明した方法と同様です。

デバイスモードの表と同様に、以下の新しい表表 728：バルクダブルバッファリングメモリバッファの用途（ホストモード）にOUTおよびINトークンのプログラム設定を示します。

表728.バルクダブルバッファリングメモリバッファの用途（ホストモード）

エンドポイントタイプ	DTOG	SW_BUF	USBペリフェラルによって使用されるパケットバッファ	アプリケーションソフトウェアによって使用されるパケットバッファ
送信（OUT）	0	1	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置
	1	0	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
	0	0	なし (1)	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
	1	1	なし (1)	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。

表728.バルクダブルバッファリングメモリバッファの用途（ホストモード）（続き）

エンドポイントタイプ	DTOG	SW_BUF	USBペリフェラルによって使用されるパケットバッファ	アプリケーションソフトウェアによって使用されるパケットバッファ
受信（IN）	0	1	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	1	0	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	0	0	なし (1)	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	1	1	なし (1)	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。

1. NAKステータスのエンドポイント。

71.5.5 デバイスモードでのアイソクロナス転送

USB標準は、固定された正確なデータ生成/消費周波数を必要とするフルスピードペリフェラルをサポートし、この種類のトラフィックを「アイソクロナス」として定義しています。このデータの典型的な例は、オーディオサンプル、圧縮ビデオストリーム、および一般に、供給される周波数の正確さについて厳密な要件を持つ、あらゆる種類のサンプルデータです。エニュメレーションフェーズでエンドポイントが「アイソクロナス」として定義されると、ホストは必要な帯域幅をフレームに割り当て、エンドポイントの方向に応じて、各フレームに正確に1つのINまたはOUTパケットを供給します。帯域幅要件を制限するために、アイソクロナストラフィックについては、失敗したトランザクションの再送信はできません。このため、アイソクロナストランザクションにはハンドシェイクフェーズはなく、ACKパケットは予期されず、データパケット後に送信されません。同じ理由で、アイソクロナス転送はデータトグルシーケンシングをサポートせず、常にDATA0 PIDを使用してデータパケットを開始します。

エンドポイントのアイソクロナス動作は、USB_CHEPnRレジスタのUTYPEビットを10にセットすることによって選択されます。ハンドシェイクフェーズがないため、STATRX/STATTXビットの唯一の正当な値は00（DISABLED）と11（VALID）であり、他の値の場合は、USB標準に準拠しない結果になります。アイソクロナスエンドポイントはダブルバッファリングを実装して、アプリケーションソフトウェア開発を容易にし、「送信」と「受信」の両方のパケットメモリ領域を使用して、トランザクションの成功のたびにバッファスワッピングを管理します。これにより、アプリケーションは常に完全なバッファを使用でき、USBペリフェラルはもう1つのバッファを満たすことができます。

USBペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関するDTOGビットによって、表729に従って定義されます（「受信」アイソクロナスエンドポイントの場合はDTOGRX、「送信」アイソクロナスエンドポイントの場合はDTOGTXであり、両方とも、関連するUSB_CHEPnRレジスタ）。

表729.アイソクロナスメモリバッファの使用

エンドポイントタイプ	DTOGビットの値	USBペリフェラルによって使用されるパケットバッファ	アプリケーションソフトウェアによって使用されるパケットバッファ
送信（IN）	0	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置
	1	USB_CHEP_RXTXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置	USB_CHEP_TXRXBD_0（ADDR_TX / COUNT_TX）バッファ記述テーブル位置。
受信（OUT）	0	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。
	1	USB_CHEP_TXRXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置	USB_CHEP_RXTXBD_0（ADDR_RX / COUNT_RX）バッファ記述テーブル位置。

ダブルバッファアイソクロナスエンドポイントで発生すると、アイソクロナスエンドポイントの実装に使用されるUSB_CHEPnRレジスタは強制的に単方向として使用されます。アイソクロナスエンドポイントを受信と送信の両方について有効にする必要がある場合は、2つのUSB_CHEPnRレジスタを使用する必要があります。

アプリケーションソフトウェアは、最初に使用されるバッファに応じて、DTOGビットを初期化する責任があります。これは、この2つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイント USB_CHEPnR レジスタの VTRX または VTTX ビットがセットされます。同時に、USB_CHEPnRレジスタの影響を受けたDTOGビットがハードウェアによってトグルされ、バッファスワッピングを完全にソフトウェアから独立させます。STATビットペアはトランザクションの完了に影響を受けません。アイソクロナス転送ではハンドシェイクフェーズがないため、フロー制御は不可能であり、エンドポイントは常に11（VALID）のままです。CRCエラーまたはバッファオーバーラン条件がアイソクロナスOUT転送時に発生した場合、常に正しいトランザクションとみなされ、常にVTRXイベントをトリガします。ただし、CRCエラーが発生すると、USB_ISTRレジスタのERRビットがセットされて、データ破損の可能性をソフトウェアに通知します。

71.5.6 ホストモードでのアイソクロナス転送

ホストの観点から、ホストフレームスケジューラにより、フレームごとに1つずつアイソクロナスパケットが発行、またはリクエストされます。NAK/ACKプロトコルはなく、データやトークンの再送もありません。

メカニズムは、デバイスモードの表に示したものと非常に類似しています。DTOGビットバッファとバッファの使用との関係については、以下の表730を参照してください。

表730.アイソクロナスメモリバッファの使用

エンドポイントタイプ	DTOGビットの値	USBペリフェラルによって使用されるパケットバッファ	アプリケーションソフトウェアによって使用されるパケットバッファ
送信（OUT）	0	USB_CHEP_TXRXBD_0（ADDRn_TX / COUNTn_TX）バッファ記述テーブル位置。	USB_CHEP_RXTXBD_0（ADDRn_TX / COUNTn_TX）バッファ記述テーブル位置
	1	USB_CHEP_RXTXBD_0（ADDRn_TX / COUNTn_TX）バッファ記述テーブル位置	USB_CHEP_TXRXBD_0（ADDRn_TX / COUNTn_TX）バッファ記述テーブル位置。
受信（IN）	0	USB_CHEP_RXTXBD_0（ADDRn_RX / COUNTn_RX）バッファ記述テーブル位置。	USB_CHEP_TXRXBD_0（ADDRn_RX / COUNTn_RX）バッファ記述テーブル位置。
	1	USB_CHEP_TXRXBD_0（ADDRn_RX / COUNTn_RX）バッファ記述テーブル位置	USB_CHEP_RXTXBD_0（ADDRn_RX / COUNTn_RX）バッファ記述テーブル位置。

エンドポイントのアイソクロナス動作は、USB_EPnRレジスタのUTYPEビットを10にセットすることによって選択されます。ハンドシェイクフェーズがないため、STATRX/STATTXビットの唯一の正当な値は00（DISABLED）と11（VALID）となります。

デバイスモードの場合と同様、このメカニズムではDTOGビットの自動トグルが可能です。ただし、ホストモードではこのトグルと同時に、完了したバッファのSTATTXまたはSTATRXが自動的にDISABLEDにセットされるため、今後のバッファのアクセスは、11（VALID）にセットして最有効化しなくても行えます。

71.5.7 サスペンド/レジュームイベント

USB標準では、SUSPENDと呼ばれる特殊なペリフェラル状態が定義され、USBバスからの平均電流は2.5mAより大きくなってはなりません。この要件はバスパワードデバイスにとって基本的な重要性を持ちますが、セルフパワードデバイスは、この厳格な消費電力成約を順守する必要はありません。サスペンドモードでは、ホストPCは3msを超えてUSBバスにトラフィックを送信しないことによって通知を送信します。SOFパケットは、通常動作では1msごとに送信されなければならないので、USBペリフェラルは3回連続のSOFパケットの欠落をホストPCからのサスペンドリクエストとして検出し、USB_ISTRレジスタのSUSPビットを1にセットし、有効な場合は割込みが生成されます。デバイスがサスペンドされると、いわゆるRESUMEシーケンスによって通常操作が復元され、これはホストPCから、またはペリフェラルそのものから直接開始できますが、常にホストPCによって終了されます。サスペンドされたUSBペリフェラルは、常にRESETシーケンスを検出して、通常のUSBリセットイベントとして、このイベントに対応できなければなりません。

USBペリフェラルをサスペンドするために使用される実際の手順は、デバイスの構成によって合計消費量を削減するために必要なアクションが異なるため、デバイスに依存します。

USBペリフェラルのSUSP通知に応答するアプリケーションソフトウェアルーチンのUSB関連の局面を中心にして、典型的なサスペンド手順の簡単な説明を以下に示します。

1. USB_CNTRレジスタのSUSPENビットを1にセットします。このアクションによって、USBペリフェラル内のサスペンドモードが有効になります。サスペンドモードが有効になると、USBがサスペンドされている間にSUSP割込みが発行されるのを回避するために、SOF受信時のチェックが無効になります。
2. USBペリフェラルとは異なるブロックの静的消費電力を除去または削減します。

3. USB_CNTRレジスタのSUSPRDYビットを1にセットして、アナログUSBトランシーバの静的消費電力を除去しつつ、レジュームアクティビティを検出できるままにしておきます。
4. オプションで、外部オシレータとデバイスPLLをオフにして、デバイス内部のアクティビティを停止します。

デバイスがSUSPENDモード中にUSBイベントが発生したときには、RESUME手順が呼び出されて、通常のチェックと復元し、通常のUSB動作を復元する必要があります。ウェイクイベントがUSBリセットシーケンスである時には、このプロセスが10ms以内で終了するように特に注意が必要です（詳細については、「ユニバーサルシリアルバス仕様」を参照してください）。USBペリフェラルのサスペンド中にレジュームまたはリセットシーケンスが開始されると、USB_CNTRレジスタのSUSPRDYビットが非同期でクリアされます。このイベントによってWKUP割込みがトリガされた場合でも、システムクロックの再開始による長い遅延のため、割込み応答ルーチンの使用は慎重に評価する必要があります。公称クロックの再有効化の前の遅延を短くするには、サスペンド手順の終了直後にレジューム手順を行って、システムクロックが再開始したらすぐにコードが実行されるようにします。システムをウェイクアップする際のESD放電またはその他の種類のノイズを防止するために（サスペンドモードの終了は、非同期イベントです）、サスペンド中にデータラインステータスの適切なアナログフィルタを有効にします。フィルタ幅は約70nsです。

以下は、レジューム手順で行うべきアクションのリストです。

1. オプションで、外部オシレータおよびデバイスPLLをオンにします。
2. USB_CNTRレジスタのSUSPENビットをクリアします。
3. レジュームをトリガしたイベントを識別する必要がある場合は、USB_FNR レジスタのビットRXDPおよびRXDMを 表731に従って使用でき、この表には、あらゆる場合のソフトウェアのアクションもリストされています。必要な場合には、上記のビットのステータスを監視することによって、レジュームまたはリセットシーケンスの終了を検出でき、これらが「10」設定（アイドルバス状態を表します）に達したことをチェックします。さらに、リセットシーケンスの終了時には、USB_ISTRレジスタのRST_DCONビットが1にセットされ、有効な場合は割込みが発行され、通常通りに処理されます。

表731.レジュームイベント検出

[RXDP,RXDM] ステータス	ウェイクアップイベント	必要なレジュームソフトウェアアクション
『00』	ルートのリセット	なし
『10』	なし（バス上のノイズ）	サスペンドモードに戻る
『01』	ルートのレジューム	なし
『11』	不可（バス上のノイズ）	サスペンドモードに戻る

デバイスは、USBプロトコルに直接関係しない特定のイベントに対する応答として、サスペンドモードを終了しなければならない場合があります（たとえば、マウスの移動はシステム全体をウェイクアップします）。この場合、USB_CNTRレジスタのL2RESビットを1にセットし、1msから15msまでのインターバル後に0にリセットすることによって、レジュームシーケンスを開始できます（このインターバルは、システムクロックが公称周波数で動作しているときには1ms周期で発生するESOF割込みによって計測できます）。L2RESビットがクリアされると、レジュームシーケンスはホストPCによって完了され、その終了はUSB_FNRレジスタのRXDPおよびRXDMビットを使用して監視できます。

注： **L2RESビットは、USBペリフェラルがサスペンドモードにされた（USB_CNTRレジスタのSUSPENビットを1にセットした）後でのみ使用する必要があります。**

ホストモードでのサスペンドとレジューム

前のセクションでは、サスペンドメカニズムとレジュームメカニズムの基本について説明しました。

ホストから見ると、サスペンドはUSB_CNTRのSUSPENビットを書き込むことで有効化されます。サスペンドエントリが確認されると、（USB_CNTRの）SUSPRDYがセットされます。

サスペンドの状態のとき、アプリケーションでバスのレジュームを行うには、USB_CNTRのL2RESビットを1にセットします。

以下の表732には、ウェイクアップイベント後に推奨される別のアクションを示します。ウェイクアップイベント後の別のラインの状態に従って、イベントの解釈と提案される動作を示します。ただし、ホストはフルスピードデバイスとロースピードデバイスの両方に対応し、サスペンドとレジュームの両方に異なるラインの状態が使用されることがあるため、この表は前に示したデバイスの表と比べて多少長くなります。

表732.ホスト用のレジュームイベント検出

[RXDP,RXDM] ステータス	ウェイクアップイベント	必要なレジュームソフトウェアアクション
『00』	不可（バス上のノイズ）	サスペンドモードに戻る
『10』	フルスピード対応デバイス：不可（バス上のノイズ） ロースピードデバイス：デバイスのリモートウェイクアップレジューム	なし
『01』	フルスピード対応デバイス：デバイスのリモートウェイクアップレジューム ロースピードデバイス： 不可（バス上のノイズ）	なし
『11』	不可（バス上のノイズ）	サスペンドモードに戻る

71.6 USBおよびUSB SRAMレジスタ

USBペリフェラルレジスタは、以下のグループに分けることができます。

- 共通レジスタ：割込みおよび制御レジスタ
- エンドポイント/チャネルレジスタ：エンドポイント/チャネル設定およびステータス

USB SRAMレジスタは以下をカバーします。

- バッファディスクリプタテーブル：データバッファの配置に使用されたパケットメモリの位置（USB SRAMのベースアドレスは、セクション 2.3：メモリ構成を参照してください）。

すべてのレジスタアドレスは、USBペリフェラルレジスタのベースアドレスに対するオフセットとして表されますが、USB SRAMベースアドレスから始まるバッファディスクリプタテーブル位置は除外されます。

レジスタの説明で使用されている略語のリストについては、123ページのセクション 1.2を参照してください。

ペリフェラルレジスタには、ワード（32bit）単位でアクセスすることができます。

71.6.1 共通レジスタ

これらのレジスタはUSBペリフェラルの全般的動作に影響し、動作モード、割込み処理、デバイスアドレス、およびホストPCによって更新される現在のフレーム番号へのアクセス付与を定義します。

USB制御レジスタ（USB_CNTR）

アドレスオフセット：0x40

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HOST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DDISCM	THR 512M
rw														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTRM	PMA OVRM	ERRM	WKUP M	SUSP M	RST_D CONM	SOFM	ESOF M	L1REQ M	Res.	L1RE S	L2RE S	SUS PEN	SUSP RDY	PDWN	USB RST
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	r	rw	rw

ビット31 **HOST**：ホストモード

HOSTビットでは、ホストまたはデバイスUSB操作モードを選択します。このビットは、機能イネーブルビットによってUSBペリフェラルを有効化する前にセットする必要があります。

0：USBデバイス機能

1：USBホスト機能

ビット30:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DDISCM**：デバイス切断マスク

– ホストモード

0：デバイス切断割込みは無効です

1：デバイス切断割込みは有効です

ビット16 **THR512M**：512バイト閾値割込みマスク

0：512バイト閾値割込みは無効です

1：512バイト閾値割込みは有効です

ビット15 **CTRM**：正しい転送割込みマスク

0：正しい転送（CTR）割込みは無効です。

1：CTR割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット14 **PMAOVRM**：パケットメモリ領域のオーバー /アンダーラン割込みマスク

0：PMAOVR割込みは無効です。

1：PMAOVR割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット13 **ERRM**：エラー割込みマスク

0：ERR割込みは無効です。

1：ERR割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット12 **WKUPM**：ウェイクアップ割込みマスク

0：WKUP割込みは無効です。

1：WKUP割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット11 **SUSPM**：サスペンドモード割込みマスク

0：サスペンドモードリクエスト（SUSP）割込みは無効です。

1：SUSP割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット10 **RST_DCONM**：USBリセットリクエスト（デバイスモード）またはデバイス接続/切断（ホストモード）割込みマスク

0：RESET割込みは無効です。

1：RESET割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット9 **SOFM**：フレーム開始割込みマスク

0：SOF割込みは無効です。

1：SOF割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット8 **ESOFM**：予期されるフレーム開始割込みマスク

0：予期されるフレーム開始（ESOF）割込みは無効です。

1：ESOF割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット7 **L1REQM**：LPM L1状態リクエスト割込みマスク

0：LPM L1状態リクエスト（L1REQ）割込みは無効です。

1：L1REQ割込みは有効であり、USB_ISTRレジスタの対応するビットがセットされたときには割込みリクエストが生成されます。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **L1RES**：L1リモートウェイクアップ/レジュームドライバ

– デバイスモード

LPM L1 50µsリモートウェイクアップ信号をホストに送信するために、ソフトウェアによってこのビットがセットされます。このビットは、信号の送信終了後にハードウェアによってクリアされます。

0：影響なし。

1：50µsリモートウェイクアップ信号をホストに送信します。

ビット4 **L2RES**：L2リモートウェイクアップ/レジュームドライバ

– デバイスモード

マイクロコントローラはこのビットをセットして、ホストにリモートウェイクアップ信号を送信できます。USB仕様に従って、ホストPCがレジュームシーケンスを終了まで駆動する準備ができてから1ms以上15ms以下で有効にする必要があります。

– ホストモード

レジューム信号をデバイスに送信するために、ソフトウェアによってこのビットがセットされます。

レジューム終了をデバイスに送信してSOF生成を再開するために、ソフトウェアによってこのビットがクリアされます。

リモートウェイクアップのコンテキストでは、WAKEUP割込みに続いてこのビットをセットします。

0：影響なし。

1：L2レジューム信号をデバイスに送信します

ビット3 **SUSPEN**：サスペンド状態の有効化

– 条件：デバイスモード

このビットはSUSP割込みを受信したときにソフトウェアでセットすることができますが、SUSP割込みは、USBペリフェラルで3ms間、トラフィックの受信が行われなかったときに発行されます。このビットは、肯定応答が送信されてL1REQ割込みを受信した場合にもソフトウェアでセットできます。

サスペンド状態が内部的に伝播されるとすぐにデバイスのすべてのアクティビティが停止し、USBクロックがゲートされ、USBトランシーバが低電力モードにセットされ、SUSPRDYビットがハードウェアによってセットされます。USBクロックソースを停止し、マイクロコントローラをSTOPモードに移行することにより、デバイスアプリケーションの電力消費をより積極的に節約したい場合は、バスパワードデバイスアプリケーションの場合と同様、数サイクル待って、SUSPRDY = 1によりサスペンドリクエストに対して確認応答されることを確認する必要があります。

このビットは、WAKEUPフラグがセットされるのと同時にハードウェアによってクリアされます。

0：影響なし。

1：L1/L2サスペンドを開始します

– 条件：ホストモード

次のフレームに対してホストアプリケーションに何もスケジュールされていなくて、長期的な省電力を開始したい場合、ソフトウェアによってこのビットをセットすることができます。このビットをセットすると、ただちにSOF生成と他のホストのアクティビティが停止され、USBクロックがゲートされ、トランシーバが低電力モードにセットされます。USBトランザクションの実行中にSUSPENがセットされた場合、現在のトランザクションが終了した時点でサスペンドが開始されます。

サスペンド状態が内部で伝播されて有効になるとすぐに、SUSPRDYビットがセットされます。USBクロックソースを停止し、マイクロコントローラをSTOPモードに移行することにより、ホストアプリケーションの電力消費をより積極的に節約したい場合は、数サイクル待って、SUSPRDY = 1によりサスペンドリクエストに対して確認応答されることを確認する必要があります。

このビットは、WAKEUPフラグがセットされるのと同時にハードウェアによってクリアされます。

0：影響なし。

1：L1/L2サスペンドを開始します

ビット2 **SUSPRDY**：サスペンド状態が有効

SUSPEN制御が内部的に有効になることでサスペンド状態が開始されるとすぐに、ハードウェアによってこのビットがセットされます。この状態では、USBのアクティビティはサスペンドされ、USBクロックはゲートされ、トランシーバは作動レシーバを無効化することにより低電力モードにセットされます。リモートウェイクアップまたはレジュームイベントを検出するため、非同期ウェイクアップロジックとシングルエンドレシーバのみが動作状態を維持します。

STOPモードが開始される前にビットがセットされていることを確認するため、ソフトウェアによりこのビットをポーリングする必要があります。

このビットは、WAKEUPフラグがセットされるのと同時にハードウェアによってクリアされます。

0：通常動作

1：サスペンド状態

ビット1 **PDWN**：パワーダウン

このビットは、何らかの理由でUSBペリフェラルを完全に無効にする必要がある場合、USB関連のすべてのアナログ部品を完全にオフにするために使用されます。このビットがセットされると、USBペリフェラルはトランシーバから切断され、使用できなくなります。

0：パワーダウンを終了します

1：パワーダウンモードを開始します

ビット0 **USBRST**：USBリセット

– 条件：デバイスモード

USBでRESET信号を受信するときと同様に、USBコアをリセットするために、ソフトウェアによってこのビットをセットすることができます。USBペリフェラルは、RESETに対応して、内部プロトコル状態マシンをリセットします。RST_DCONビットがクリアされるまで、受信と送信は無効です。すべての設定レジスタがリセットされるわけではありません。マイクロコントローラはこれらのレジスタを明示的にクリアする必要があります（これは、RST_DCON割込みの安全な配信を確保し、RESETの直前のトランザクションを完了できるようにするためです）。機能アドレスおよびエンドポイントレジスタは、USBリセットイベントによってリセットされます。

0：影響なし。

1：USBコアがリセット中です。

– 条件：ホストモード

バスでUSBリセット状態を駆動し、デバイスを初期化するため、ソフトウェアによってこのビットがセットされます。ソフトウェアによってこのビットがクリアされるとすぐに、USBリセットが終了します。

0：影響なし。

1：USBリセットが駆動されます。

USB割込みステータスレジスタ（USB_ISTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

このレジスタはすべての割込みソースのステータスを含むためアプリケーションソフトウェアは割込みリクエストの原因になったイベントを判断できます。

このレジスタの上位部分はシングルビットを含み、そのそれぞれが特定のイベントを表します。これらのビットは、関連するイベントが発生したときにハードウェアによってセットされます。USB_CNTRレジスタの対応するビットがセットされている場合、汎用割込みリクエストが生成されます。各ビットを調べる割込みルーチンは、必要なすべてのアクションを実行し、最後に処理済みビットをクリアします。そのいずれかがクリアされていない場合、割込みはまだ保留中であるとみなされ、割込みラインは再びハイに保たれます。複数のビットが同時にセットされた場合、1つの割込みだけが生成されます。

エンドポイント/チャネルトランザクションの完了は、割込み応答遅延を軽減するために、さまざまな方法で処理できます。CTRビットは、エンドポイント/チャネルがトランザクションを正常に完了するとすぐにハードウェアによってセットされ、USB_CNTRの対応するビットがセットされていた場合は汎用割込みリクエストが生成されます。エンドポイント/チャネル専用割込み条件は、USB_CNTRレジスタのCTRMビットとは無関係に有効化されます。どちらの割込み条件も、ソフトウェアが対応するUSB_CHEPnRレジスタの保留中のビットをクリアするまでアクティブなままです（CTRビットは実際には読出し専用ビットです）。エンドポイント/チャネル関連の割込みの場合、ソフトウェアは、トランザクション方向（DIR）およびIDN読出し専用ビットを使用して、最後の割込みリクエストを発行して対応する割込みサービスルーチンを呼び出したエンドポイント/チャネルを特定できます。

ユーザは、ソフトウェアが割込みサービスルーチンでUSB_ISTRビットをチェックする順序を指定することによって、同時に保留中のUSB_ISTRビットの相対的優先順位を選ぶことができます。イベントに関連し、処理済みのビットだけがクリアされます。サービスルーチンの終了時、残りの条件を処理するために、別の割込みがリクエストされます。

ビットの誤ったクリアを避けるために、ロード命令付きでクリアすることが推奨され、その場合、変更してはならないすべてのビットに1が書き込まれ、クリアすべきすべてのビットに0が書き込まれます（これらのビットはソフトウェアによってのみクリアできます）。読出し-変更-書込みのサイクルは避けるべきです。読出し操作と書込み操作の間に、別のビットがハードウェアによってセットされる可能性があり、デバイスにイベントを処理する時間ができる前に、次の書込みによってクリアされる可能性があるためです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LS_DCON	DCON_STAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DDISC	THR512
	r	r												rc_w0	rc_w0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTR	PMAOVR	ERR	WKUP	SUSP	RST_DCON	SOF	ESOF	L1REQ	Res.	Res.	DIR	IDN[3:0]			
r	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0			r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **LS_DCON**：ロースピードデバイスが接続済み

– ホストモード：

このビットは、LSデバイス接続が検出されたときに、ハードウェアによってセットされます。デバイス接続は、未接続状態から連続で22サイクルのUSBクロック（48MHz）の間、LS J状態がサンプリングされた後で信号送信されます。

ビット29 **DCON_STAT**：デバイスの接続ステータス

– ホストモード：

このビットには、デバイス接続ステータスに関する情報が格納されます。このビットはLS/FSデバイスがホストに取り付けられるとハードウェアによってセットされ、デバイスが取り外されるとリセットされます。

0：デバイスは接続されていません

1：FSまたはLSデバイスがホストに接続されています

ビット28:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **DDISC**：デバイスの接続

– ホストモード

このビットは、デバイス接続が検出されたときにセットされます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット16 **THR512**：512バイト閾値割込み

このビットは、アイソクロナス転送中に512バイトの送受信が行われたときに、ハードウェアによって1にセットされます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。関連するチャネル/エンドポイントを示すための情報はありませんが、実際にはこのような大きなパケットを持つ1つのISOエンドポイント/チャネルのみがサポート可能です。

ビット15 **CTR**：ホストモードでの転送完了

このビットは、エンドポイント/チャネルがトランザクションを正常に完了したことを示すために、ハードウェアによってセットされます。DIRおよびIDNビットを使用して、ソフトウェアは割込みをリクエストしたエンドポイント/チャネルを判断できます。このビットは読出し専用です。

ビット14 **PMAOVR**：パケットメモリ領域のオーバー /アンダーラン

このビットは、マイクロコントローラにUSBメモリリクエストに対応する時間がなくなった場合にセットされます。USBペリフェラルは、このイベントを次のように処理します。受信時、ACKハンドシェイクパケットは送信されず、送信時、送信ストリームのビットスタッフエラーが強制的に発行されます。いずれの場合も、ホストはトランザクションを再試行します。通常操作では、PMAOVR割込みは発生しません。失敗したトランザクションはホストによって再試行されるので、アプリケーションソフトウェアには、この割込み処理中にデバイス操作をスピードアップして、次のトランザクションの差異施行に備える機会があります。ただし、アイソクロナス転送では（アイソクロナストランザクションは再試行されません）、データの喪失につながるため、これは行われません。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット13 **ERR**：エラー

このフラグは、下記にリストされているエラーのいずれかが発生したときにセットされます。

NANS：応答なしホスト応答がタイムアウトになりました。

CRC： 巡回冗長検査（Cyclic Redundancy Check）エラー。トークンまたはデータで受信されたCRCの1つが正しくありませんでした。

BST： ビットスタッフィングエラー。ビットスタッフィングエラーが、PID、データ、またはCRCで検出されました。

FVIO： フレーミングフォーマット違反。非標準のフレームが受信されました（正しい位置にないEOP、正しくないトークンシーケンスなど）。

USBペリフェラルとPCホストは、エラーの場合、完全にトランスペアレントな方法で再送信を管理するため、USBソフトウェアは、通常、エラーを無視できます。この割込みは、ソフトウェア開発フェーズで、またはUSBバスでの送信品質の監視に役立ち、問題の可能性をユーザに知らせることができます（コネクタの緩み、ノイズが多い環境、USBケーブルの導体の破損など）。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット12 **WKUP**：ウェイクアップ

このビットは、サスペンドモード中に、USBペリフェラルをウェイクアップするアクティビティが検出されたときにハードウェアによって1にセットされます。このイベントはCTLRレジスタのSUSPRDYビットを非同期でクリアし、USB_WAKEUPラインを有効にし、レジュームプロセスの開始に関して残りのデバイス（ウェイクアップユニットなど）に通知するために使用されます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット11 **SUSP**：サスペンドモードリクエスト

– デバイスモード

このビットは、USBバスからのサスペンドモードリクエストを含めて、トラフィックを3ms間受信していない場合に、ハードウェアによってセットされます。サスペンド条件検査はUSBリセットの直後に有効化され、サスペンドモードが有効化されたときに（SUSPEN=1）、レジュームシーエンスの終了までハードウェアによって無効化されます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット10 **RST_DCON**：USBリセットリクエスト（デバイスモード）またはデバイス接続/切断（ホストモード）

– デバイスモード

このビットは、ホストによってUSBリセットが解放され、バスがアイドル状態に戻るときに、ハードウェアによってセットされます。USBリセット状態は、連続する60回のSE0サイクルのサンプリング後に、内部で検出されます。

– ホストモード

このビットは、デバイス接続またはデバイス切断が検出されたときに、ハードウェアによってセットされます。デバイス接続の信号送信は、未接続状態から22サイクル連続でJ状態がサンプリングされた後で行われます。デバイス切断の信号送信は、接続状態から22bit時間連続でSE0状態が検出された後で行われます。

ビット9 **SOF**：フレーム開始（スタート･オブ･フレーム）

このビットは、新しいUSBフレームの開始を通知するもので、SOFパケットがUSBバス経由で着信したときにセットされます。割込みサービスルーチンによってSOFイベントを監視して、USBホストへの1 msの同期イベントを実行し、SOFパケット受信時に更新されるUSB_FNRレジスタを安全に読み出すことができます（これはアイソクロナスアプリケーションに便利です）。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット8 **ESOF**：予期されたフレーム開始

– デバイスモード

このビットは、予期されたSOFパケットが受信されなかったときにハードウェアによってセットされます。ホストから1msごとにSOFパケットが送信されますが、デバイスでこのパケットが正しく受信されない場合、サスペンドタイマによってこの割込みが発行されます。3回連続してESOF割込みが生成され（たとえば、3つのSOFパケットが失われ）、その間にトラフィックが発生しなかった場合、SUSP割込みが生成されます。このビットは、サスペンドタイマがまだロックされていないときにSOFパケットの喪失が発生したときにセットされます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット7 **L1REQ**：LPM L1状態リクエスト

– デバイスモード

このビットは、L1状態を入力するLPMコマンドが正常に受信され、確認応答された場合に、ハードウェアによってセットされます。このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット6:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **DIR：トランザクションの方向**

このビットは、割込みリクエストを生成した成功したトランザクションの方向に従って、ハードウェアによって書き込まれます。

DIRビット = 0の場合、割込みエンドポイントに関連してUSB_CHEPnRレジスタでVTTXビットがセットされます。割込みの原因となったトランザクションは、INタイプです（USBペリフェラルからホストPCへのデータ送信）。

DIRビット = 1の場合、割込みをかけたエンドポイントに関連するUSB_CHEPnRレジスタのVTRXビットまたはVTTXとVTRXの両方がセットされます。割込みの原因となったトランザクションはOUTタイプであるか（ホストPCからのデータをUSBペリフェラルが受信）、または2つの保留中のトランザクションが処理を待っています。

これは割込みを保留中にした方向を表すので、アプリケーションソフトウェアはこの情報を使用して、トリガしたトランザクションに関連するUSB_CHEPnRビットにアクセスできます。このビットは読出し専用です。

ビット3:0 **IDN[3:0]：デバイスエンドポイント/ホストチャネル識別番号**

これらのビットは、割込みリクエストを生成したホストチャネルまたはデバイスエンドポイントの番号に従って、ハードウェアによって書き込まれます。いくつかのエンドポイント/チャネルのトランザクションが保留中である場合、次のように定義された優先順位の高いエンドポイント/チャネルに関する識別番号がハードウェアによって書き込まれます。まず、優先順位に従って2つのレベルが定義され、イアイソクロナスおよびダブルバッファバルクチャネル/エンドポイントが最初に検討されてから、それ以外が検査されます。同じセットの複数のエンドポイント/チャネルが割込みをリクエストしている場合、最下位リクエストレジスタに従って、USB_ISTRレジスタのIDNビットが割り当てられ、CHEP0Rが最も高い優先順位を持ち、その後にCHEP1Rが続き、以下同様です。アプリケーションソフトウェアは、この優先順位方式するチャネルに従って、レジスタを各エンドポイント/チャネルに割り当てて、同時発生したエンドポイント/チャネルリクエストを適切に順序づけることができます。これらのビットは読出し専用です。

USBフレーム番号レジスタ（USB_FNR）

アドレスオフセット：0x48

リセット値：0x0000 0XXX（Xは未定義）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDP	RXDM	LCK	LSOF[1:0]		FN[10:0]										
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **RXDP**：受信データ + ラインステータス

このビットを使用して、受信データプラスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。

ビット14 **RXDM**：受信データ - ラインステータス

このビットを使用して、受信データマイナスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。

ビット13 **LCK**：ロック

– デバイスモード

このビットは、USBリセット条件の終了後、またはUSBレジュームシーケンスの終了後に、少なくとも2つ連続したSOFパケットが受信されたときに、ハードウェアによってセットされます。ロックされると、USBリセットまたはUSBサスペンドイベントが発生するまで、フレームタイマはこの状態に保たれます。

ビット12:11 **LSOF[1:0]**：SOFの喪失

– デバイスモード

これらのビットは、ESOF割込みが生成されたときに、ハードウェアによって書き込まれ、失われた連続SOFパケットの数をカウントします。SOFパケットの受信時、これらのビットはクリアされます。

ビット10:0 **FN[10:0]**：フレーム番号

このビットフィールドは、最後に受信されたSOFパケットに含まれる11bitのフレーム番号を含みます。フレーム番号は、ホストによって送信されたフレームごとにインクリメントされ、アイソクロナス転送の場合に便利です。このビットフィールドは、SOF割込みの生成時に更新されます。

USBデバイスアドレス（USB_DADDR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EF	ADD[6:0]						
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **EF**：機能の有効化

このビットは、USBデバイスを有効にするために、ソフトウェアによってセットされます。このデバイスのアドレスは、次のADD[6:0] ビットに含まれます。このビットが0の場合、USB_CHEPnRレジスタの設定にかかわらず、トランザクションは処理されません。

ビット6:0 **ADD[6:0]**：デバイスアドレス

– デバイスモード

これらのビットは、エニュメレーションプロセス時にホストPCによって割り当てられたUSB機能を含みます。必要なエンドポイントへのトランザクションを処理するためには、関連するUSB_CHEPnRレジスタのこのフィールドとエンドポイント/チャネルアドレス（EA）フィールドの両方とも、USBトークンに含まれる情報に一致する必要があります。

– ホストモード

これらのビットには、LPMトランザクションと一緒に送信されたアドレスが格納されます。

LPM制御およびステータスレジスタ（USB_LPMCSR）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BESL[3:0]				REM WAKE	Res.	LPM ACK	LPM EN
								r	r	r	r	r		rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **BESL[3:0]**：BESL値

– デバイスモード

これらのビットは、最後にACKされたLPMトークンとともに受信したBESL値を含みます。

ビット3 **REMWAKE**：bRemoteWake値

– デバイスモード

このビットは、最後にACKされたLPMトークンとともに受信したbRemoteWake値を含みます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **LPMACK**：LPMトークン確認応答の有効化

– デバイスモード：

0：有効なLPMトークンはNYETです。
1：有効なLPMトークンはACKです。
NYET/ACKはLPMトランザクションが成功した場合にのみ返されます。
EXTトークンとLPMトークンの両方にエラーがない場合（それ以外の場合はERROR）
有効なbLinkState = 0001B（L1）を受信した場合（それ以外の場合はSTALL）

ビット0 **LPMEN**：LPMサポートの有効化

– デバイスモード

このビットは、USBデバイス内のLPMサポートを有効にするために、ソフトウェアによってセットされます。このビットが0の場合は、LPMトランザクションは処理されません。

バッテリ充電検出回路(USB_BCDR)

アドレスオフセット：0x58

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPPU_DPD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET	SDEN	PDEN	DCDEN	BCDEN
rw								r	r	r	r	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **DPPU_DPD**：DPプルアップ/DPDMプルダウン

– デバイスモード

このビットは、DPライン上の組み込みプルアップを有効にするために、ソフトウェアによってセットされます。0にクリアすると、ユーザのソフトウェアが必要とするときに、ホストとの接続が切断されたことを通知します。

– ホストモード

このビットは、DPおよびDMライン上の組み込みプルダウンを有効にするために、ソフトウェアによってセットされます。

ビット14:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **PS2DET**：DMプルアップ検出ステータス

– デバイスモード

このビットは、PD中にのみアクティブになってDM電圧レベルとV_{LGC} 閾値の比較の結果を示します。通常の状況では、DMレベルは閾値を下回る必要があります。閾値を上回る場合は、DMが外部的にハイレベルにプルアップされることを意味します。これは、PS2ポート（DPおよびDMラインの両方をプルアップする）への接続、またはBCD仕様に準拠しない一部の独自の充電器への接続によって発生します。
0：通常のポートが検出されました（SDP、ACA、CDP、またはDCPに接続されたポート）。
1：PS2ポートまたは独自の充電器が検出されました。

ビット6 **SDET**：2次検出（SD）ステータス

– デバイスモード

このビットは、SDの結果を示します。
0：CDPが検出されました。
1：DCPが検出されました。

ビット5 **PDET**：1次検出（PD）ステータス

– デバイスモード

このビットは、PDの結果を示します。

0：BCDサポートは検出されませんでした（SDPまたは独自デバイスに接続）。

1：BCDのサポートが検出されました（ACA、CDP、またはDCPに接続）。

ビット4 **DCDET**：データ接触検出（DCD）ステータス

– デバイスモード

このビットは、DCDの結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

ビット3 **SDEN**：2次検出（SD）モードの有効化

– デバイスモード

このビットは、BCDをSDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット2 **PDEN**：1次検出（PD）モードの有効化

– デバイスモード

このビットは、BCDをPDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット1 **DCDEN**：データ接触検出（DCD）モードの有効化

– デバイスモード

このビットは、BCDをDCDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット0 **BCDEN**：バッテリ充電検出（BCD）の有効化

– デバイスモード

このビットは、USBデバイス内のBCDサポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHYはBCDによって完全に制御され、通常の通信では使用できなくなります。BCDの検出完了後は、通常のUSB動作を可能にするために、このビットを0にクリアしてBCDをOFFモードにします。

ホストチャネル固有/デバイスエンドポイント固有のレジスタ

これらのレジスタの数は、USBペリフェラルが処理する設計になっているエンドポイントまたはホストチャネルの数に応じて変化します。USBペリフェラルは、最大8つの双方向エンドポイントまたはホストチャネルをサポートします。各USBデバイスは、制御エンドポイント/チャネルをサポートしなければならず、そのアドレス（EAビット）は0にセットされなければなりません。複数のエンドポイントが有効であり、同じエンドポイント/チャネル番号値を持つ場合、USBペリフェラルの動作は未定義です。各エンドポイントについて、USB_CHEPnRレジスタはエンドポイント/チャネル固有の情報を格納できます。

USBエンドポイント/チャネルのnレジスタ（USB_CHEPnR）

アドレスオフセット：0x00 + 0x4*n、（n = 0 ~ 7）

リセット値：0x0000 0000

USBリセットがUSBバスから受信されたとき、またはCTLRレジスタのビットUSBRSTを通じて強制されたときにもリセットされますが、USBリセットイベントの直前の正しいパケット通知の喪失を回避するために、VTRX および VTTX ビットは変更されません。各エンドポイント / チャネルにUSB_CHEPnRレジスタがあり、*n*はエンドポイント/チャネル識別子です。

これらのレジスタでは読出し-変更-書込みのサイクルは避ける必要があります。読出し操作と書込み操作の間に、いくつかのビットがハードウェアによってセットされる可能性があり、CPUに変更を検

出する時間ができる前に、次の書込みによって変更される可能性があるためです。この目的のために、この問題の影響を受けるすべてのビットが「不変」の値を持ち、変更が不要なときに使用される必要があります。これらのレジスタはロード命令付きで変更することが推奨され、この場合、ハードウェアによって変更できるすべてのビットには「不変」値が書き込まれます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	THREE_ERR_RX[1:0]		THREE_ERR_TX[1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]						
	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rw	rc_w0	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VTRX	DTOG RX	STATRX[1:0]		SETUP	UTYPE[1:0]		EP KIND	VTTX	DTOG TX	STATTX[1:0]		EA[3:0]			
rc_w0	t	t	t	r	rw	rw	rw	rc_w0	t	t	t	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **THREE_ERR_RX[1:0]**：INトランザクションの3個のエラー

– ホストモード

このビットは、INトランザクションに対してUSBバスでトランザクションエラーが3回連続して発生した場合に、ハードウェアによってセットされます。アイソクロナストランザクションでは、THREE_ERR_RXは生成されません。このビットは、ソフトウェアのみがクリアできます。

受信したエラーのコーディング：

00：受信したエラーが3個未満。

01：受信したエラーが3個を超え、最後のエラーがタイムアウトエラー。

10：受信したエラーが3個を超え、最後のエラーがデータエラー（CRCエラー）。

11：受信したエラーが3個を超え、最後のエラーがプロトコルエラー（無効なPID、偽のEOP、ビットスタッフィングエラー、SYNCエラー）。

ビット28:27 **THREE_ERR_TX[1:0]**：OUTまたはSETUPトランザクションの3個のエラー

– ホストモード

このビットは、OUTトランザクションに対してUSBバスでトランザクションエラーが3回連続して発生した場合に、ハードウェアにってセットされます。アイソクロナストランザクションでは、THREE_ERR_TXは生成されません。このビットは、ソフトウェアのみがクリアできます。

受信したエラーのコーディング：

00：受信したエラーが3個未満。

01：受信したエラーが3個を超え、最後のエラーがタイムアウトエラー。

10：受信したエラーが3個を超え、最後のエラーがデータエラー（CRCエラー）。

11：受信したエラーが3個を超え、最後のエラーがプロトコルエラー（無効なPID、偽のEOP、ビットスタッフィングエラー、SYNCエラー）。

ビット26 **ERR_RX**：受信したINトランザクションのエラー

– ホストモード

このビットは、このチャネルでINトランザクション中にエラー（デバイスからの応答なし、CRCエラー、ビットスタッフィングエラー、フレーミングフォーマット違反など）が発生したときに、ハードウェアによってセットされます。このビットは、ソフトウェアのみがクリアできます。USB_CNTRレジスタのERRMビットがセットされた場合、チャネル関連のフラグ（常に有効化されます）とともに汎用割込み条件が生成されます。

ビット25 **ERR_TX**：受信したOUT/SETUPトランザクションのエラー

– ホストモード

このビットは、このチャネルでOUTまたはSETUPトランザクション中にエラー（デバイスからの応答なし、CRCエラー、ビットスタッフィングエラー、フレーミングフォーマット違反など）が発生したときに、ハードウェアによってセットされます。このビットは、ソフトウェアのみがクリアできます。USB_CNTRレジスタのERRMビットがセットされた場合、チャネル関連のフラグ（常に有効化されます）とともに汎用割込み条件が生成されます。

ビット24 **LS_EP**：ロースピードエンドポイント – ハブのみを持つホスト

– ホストモード

このビットは、対応するエンドポイントにLSトランザクションを送信するために、ソフトウェアによってセットされます。

0：フルスピードのエンドポイント

1：ロースピードのエンドポイント

ビット23 **NAK**：

– ホストモード

このビットは、デバイスがNAKにより応答したときに、ハードウェアによってセットされます。ソフトウェアでは、このビットを使用してデバイスから受信したNAKの数を監視することができます。

ビット22:16 **DEVADDR[6:0]**：

– ホストモード

エニュメレーションプロセス時にエンドポイントに割り当てられたデバイスアドレス。

ビット15 **VTRX**：USB有効トランザクションの受信

– デバイスモード

このビットは、このエンドポイントでOUT/SETUPトランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアのみがこのビットをクリアできます。USB_CNTRレジスタのCTRMビットがセットされた場合、エンドポイント関連の割込み条件（常に有効化されます）とともに汎用割込み条件が生成されます。発生したトランザクションのタイプ（OUTまたはSETUP）は、下記のSETUPビットから決めることができます。

NAKまたはSTALLハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。

このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

– ホストモード

このビットは、このチャネルでINトランザクションが正常に完了したときに、ハードウェアによってセットされます。このビットは、ソフトウェアのみがクリアできます。USB_CNTRレジスタのCTRMビットがセットされた場合、チャネル関連のフラグ（常に有効化されます）とともに汎用割込み条件が生成されます。

- トランザクションがNAKで終了した場合、このビットがセットされ、このレジスタのSTATRXフィールドからNAK状態が読み出されてアプリケーションにNAK応答がレポートされます。NAK応答されたトランザクションは保留中となり、次のフレームでホストによって自動的に再試行されます。または、STATRX状態をVALIDにリセットすることにより、ホストですぐに再試行することもできます。

- トランザクションがSTALLハンドシェイクで終了した場合、このビットがセットされ、このレジスタのSTATRXフィールドからSTALL状態が読み出されてアプリケーションにSTALL応答がレポートされます。最終的に、ホストアプリケーションによりチャネルが無効化され、再エニュメレーションされます。

- トランザクションがACKハンドシェイクで終了した場合、このビットがセットされます。

ダブルバッファリングが無効になっている場合、このレジスタのSTATRXフィールドからDISABLE状態が読み出され、アプリケーションによりACK応答がレポートされます。ホストアプリケーションにより、受信したデータがUSBRAMから読み出され、このレジスタのSTATRXフィールドにVALIDを書き込むことにより、チャネルが再設定されます。

ダブルバッファリングが有効になっている場合、このレジスタのSTATRXフィールドからVALID状態が読み出され、アプリケーションによりACK応答がレポートされます。ホストアプリケーションにより、受信したデータがUSBRAMから読み出され、このレジスタのDTOGTXビットがトグルされます。

- トランザクションがエラーで終了した場合、このビットがセットされます。

エラーはERR_RXビットを通じて確認できます（ホストモードのみ）。

このビットは読出し/書込みですが、0のみを書き込むことができ、1を書き込んでも効果はありません。

ビット14 **DTOGRX**：受信転送の場合のデータトグル

エンドポイント/チャネルがアイソクロナスでない場合、このビットには次の受信データパケットのデータトグルビット（0 = DATA0、1 = DATA1）の予期される値が格納されます。一致するデータPID値を持つデータパケット受信に続いてACKハンドシェイクが送信されると、ハードウェアによってこのビットがトグルされます。エンドポイントが制御エンドポイントとして定義される場合、ホストから受信したSETUP PIDの受信時にハードウェアによってこのビットがクリアされ（デバイスモード）、デバイスによってSETUPトランザクションが確認応答される場合はハードウェアによってこのビットが1にセットされます（ホストモード）。

エンドポイント/チャネルがダブルバッファリングを使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます（セクション 71.5.3：ダブルバッファエンドポイントとデバイスモードでの使用を参照）。

エンドポイント/チャネルがアイソクロナスの場合、この種類のチャネル/エンドポイントではデータトグルは使用されず、DATA0パケットのみが送信されるため、このビットはデータ転送用のパケットバッファスワッピングをサポートするためにのみ使用されます（セクション 71.5.5：デバイスモードでのアイソクロナス転送を参照）。アイソクロナス転送ではハンドシェイクは使用されないので、ハードウェアはデータパケット受信の終了直後に、このビットをトグルします。

このビットは、値を初期化するため（エンドポイントが制御エンドポイントでないときは必須）、または特定のデータトグル／パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが0を書き込むと、DTOGRXの値は変更されませんが、1を書き込むと、ビット値がトグルされます。このビットは読出し／書込みですが、1を書き込むことによってのみ、トグルが可能です。

ビット13:12 **STATRX[1:0]：受信転送の場合のステータスビット**

– デバイスモード

これらのビットには、表 733： 3087ページの受信ステータスエンコードにリストされているエンドポイントのステータスに関する情報が格納されます。これらのビットは、ソフトウェアによってトグルして、値を初期化できます。アプリケーションソフトウェアが0を書き込むと、値は変更されませんが、1を書き込むと、ビット値がトグルされます。ハードウェアは、このエンドポイントにアドレス指定されたOUTまたはSETUP（制御のみ）トランザクションに対応する正しい転送が発生したとき（VTRX = 1）、STATRXビットをNAKにセットするので、ソフトウェアは新しいトランザクションに確認応答する前に受信データを処理する時間があります。

ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します（セクション 71.5.3：ダブルバッファエンドポイントとデバイスモードでの使用を参照）。

エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみなので、ハードウェアはトランザクションの成功後にエンドポイントのステータスを変更できません。ソフトウェアがアイソクロナスエンドポイントのSTATRXビットをSTALLまたはNAKにセットした場合、USBペリフェラルの動作は未定義です。これらのビットは読出し/書込みですが、1を書き込むことによってのみ、トグルが可能です。

– ホストモード

これらのビットは、チャネルによって駆動されるホストトランザクションを開始、再試行、または中断するためのホストアプリケーション制御です。

これらのビットには、前回のINチャネルトランザクションに対するデバイス応答の情報も格納され、これらのビットでは次の状態のSTATRXテーブルに従ってチャネルの現在のステータスをレポートします。

- DISABLE

シングルバッファチャネルでACK確認応答を受信した場合、DISABLE値がレポートされます。DISABLE状態の場合、チャネルは使用されていないか、アクティブではなくてアプリケーションによってVALIDが書き込まれて再起動されるのを待っている状態です。アプリケーションでは、トランザクションを中止するためにVALIDのチャネルをDISABLEにリセットすることができます。この場合、トランザクションはすぐにホスト実行リストから削除されます。中断するトランザクションがすでに実行中になっている場合、USB上で定期的に終了しますが、相対VTRX割込みは生成されません。

- VALID

ホストチャネルは、VALID状態のときのみ、積極的にデバイスにUSBトランザクションを送信しようとします。VALID状態はソフトウェアにより、または新しいフレームの開始時にNAKEDチャネルでハードウェアにより自動的にセットできます。VALIDにセットされると、ホストチャネルはホスト実行キューに入り、設定されたトランザクションを送信するためにホストフレームスケジューラからの許可を待ちます。

ダブルバッファチャネルでACK確認応答を受信した場合、VALID値もレポートされます。この場合、チャネルは代替バッファでアクティブなままとなり、アプリケーションは現在のバッファを読み出してDTOGTXをトグルする必要があります。ソフトウェアの読出しが遅く、代替バッファがレディ状態でない場合、ホストチャネルはアプリケーションに対して透過的に、自動的にサスペンド状態になります。サスペンドされたダブルバッファチャネルは、遅延から復旧し、DTOGTXがトグルされるとすぐに再有効化されます。

- NAK

NAK値は、NAK確認応答を受信した場合にレポートされます。NAK状態の場合、チャネルはサスペンドされ、送信は行いません。NAK状態は、次のフレームの開始時にハードウェアによりVALIDに移行されます。あるいは、ソフトウェアによりVALIDとして書き込むと、すぐに送信を再試行するように変更でき、DISABLEとして書き込むと、無効化し、トランザクションを中断することができます。

- STALL

STALL値は、STALL確認応答を受信した場合にレポートされます。STALL状態の場合、チャネルは無効として動作します。アプリケーションにより送信の再試行は行われませんが、USBがリセットされ、再エニュメレーションされます。

ビット11 **SETUP**：セットアップトランザクション完了

– デバイスモード

このビットは読出し専用であり、最後に完了したトランザクションがSETUPのときにハードウェアによってセットされます。このビットはコントロールエンドポイントについてのみ値を変更します。成功した受信トランザクションの場合（VTRXイベント）、これを調べて、発生したトランザクションのタイプを判断する必要があります。割込みサービスルーチンを次の着信トークンによるSETUPビットの変更から保護するために、このビットはVTRXビットが1の間は停止され、VTRXが0のときに状態が変更されます。このビットは読出し専用です。

– ホストモード

このビットは、制御エンドポイントでSETUPトランザクションを送信するために、ソフトウェアによってセットされます。このビットはコントロールエンドポイントについてのみ値を変更します。SETUPトランザクションが確認応答され、VTTX割込みが生成されると、ハードウェアによってクリアされます。

ビット10:9 **UTYPE[1:0]**：トランザクションのUSBタイプ

これらのビットは、表 734：エンドポイント/チャネルタイプのエンコードに記載されているように、このエンドポイント/チャネルの動作を設定します。チャネル0/エンドポイント0は常に制御エンドポイント/チャネルでなければならず、各USB機能にはアドレス0を持つ制御エンドポイント/チャネルが少なくとも1つ必要ですが、必要な場合は、他の制御チャネル/エンドポイントを使用することもできます。制御チャネル/エンドポイントだけがSETUPトランザクションを処理し、他の種類のエンドポイントでは無視されます。SETUPトランザクションはNAKまたはSTALLで応答することはできません。制御エンドポイント/チャネルがNAKとして定義された場合、SETUPトランザクションが受信されたときには、受信方向ではUSBペリフェラルは応答せず、受信エラーをシミュレートします。制御エンドポイント/チャネルが受信方向でSTALLとして定義された場合、SETUPパケットは受け入れられ、データを送信し、CTR割込みを発行します。エンドポイント/チャネルが制御エンドポイントの場合でも、OUTトランザクションの受信は通常通りに処理されます。

バルクおよび割込みエンドポイントの動作はほぼ同様ですが、EPKIND設定ビットを使用して使用可能な特殊な機能のみが異なります。

アイソクロナスチャネル/エンドポイントの使用については、セクション 71.5.5：デバイスモードでのアイソクロナス転送で説明します。

ビット8 **EPKIND**：エンドポイント/チャネルの種類

このビットの意味は、UTYPEビットによって設定されたエンドポイント/チャネルのタイプに依存します。表735はさまざまな意味を要約します。

DBL_BUF：このビットは、このバルクエンドポイントのダブルバッファリング機能を有効にするために、ソフトウェアによってセットされます。ダブルバッファバルクエンドポイントの使用については、セクション 71.5.3：ダブルバッファエンドポイントとデバイスモードでの使用で説明します。

STATUS_OUT：このビットは、ステータスアウトトランザクションが予期されることを示すためにソフトウェアによってセットされます。この場合、1個以上のデータバイトを含んでいるすべてのOUTトランザクションは、ACKではなくSTALLで応答されます。このビットを使用して、制御転送時のプロトコルエラーに対するアプリケーションの堅牢性を向上させることができ、その使用はコントロールエンドポイントのみを対象としています。STATUS_OUTがリセットされると、OUTトランザクションは必要に応じて任意の数のバイトを持つことができます。

ビット7 **VTTX**：送信される有効なUSBトランザクション

– デバイスモード

このビットは、このエンドポイントでINトランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアのみがこのビットをクリアできます。USB_CNTRレジスタのCTRMビットがセットされた場合、エンドポイント関連の割込み条件（常に有効化されます）とともに汎用割込み条件が生成されます。

NAKまたはSTALLハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。このビットは読出し/書込みですが、0のみを書き込むことができます。

– ホストモード

VTRXの動作と同様ですが、USB OUTおよびSETUPトランザクション用です。

ビット6 **DTOGTX**：送信転送の場合のデータトグル

エンドポイント/チャネルがアイソクロナスでない場合、このビットには次の送信データパケットのデータトグルビット（0 = DATA0、1 = DATA1）の必要な値が格納されます。データパケットの送信後、USBホストからACKハンドシェイクが受信されると、ハードウェアはこのビットをトグルします。エンドポイント/チャネルが制御エンドポイント/チャネルとして定義されている場合、このエンドポイントにアドレス指定されたSETUP PIDの受信時（デバイスモード）、またはデバイスによってSETUPトランザクションが確認応答されたときに（ホストモード）、このビットはハードウェアによって1にセットされます。

エンドポイント/チャネルがダブルバッファ機能を使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます（セクション 71.5.3：ダブルバッファエンドポイントとデバイスモードでの使用を参照）。

エンドポイント/チャネルがアイソクロナスの場合、この種類のエンドポイントではデータトグルは使用されず、DATA0パケットのみが送信されるため、このビットはパケットバッファスワッピングをサポートするために使用されます（セクション 71.5.5：デバイスモードでのアイソクロナス転送を参照）。アイソクロナス転送ではハンドシェイクは使用されないので、ハードウェアはデータパケット送信の終了直後に、このビットをトグルします。

このビットは、値を初期化するため（エンドポイント/チャネルが制御エンドポイント/チャネルでないときは必須）、または特定のデータトグル/パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが0を書き込むと、DTOGTX値は変更されませんが、1を書き込むと、ビット値がトグルされます。このビットは読出し／書込みですが、1を書き込むことによってのみ、トグルが可能です。

ビット5:4 **STATTX[1:0]**：送信転送の場合のステータスビット

– デバイスモード

これらのビットは、表736にリストされているエンドポイントのステータスに関する情報を含みます。これらのビットは、ソフトウェアによってトグルして、値を初期化できます。アプリケーションソフトウェアが0を書き込むと、値は変更されませんが、1を書き込むと、ビット値がトグルされます。ハードウェアは、このチャネル/エンドポイントにアドレス指定されたINまたはSETUP（制御のみ）トランザクションに対応する正しい転送が発生したとき（VTTX = 1）、STATTXビットをNAKにセットします。次に、ソフトウェアが次の送信データセットを準備するのを待ちます。

ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します（セクション 71.5.3：ダブルバッファエンドポイントとデバイスモードでの使用を参照）。

エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみです。したがって、ハードウェアは、トランザクションの成功後にチャネル/エンドポイント/チャネルのステータスを変更することはできません。ソフトウェアがアイソクロナスチャネル/エンドポイントのSTATTXビットをSTALLまたはNAKにセットした場合、USBペリフェラルの動作は未定義です。これらのビットは読出し/書込みですが、1を書き込むことによってのみ、トグルが可能です。

– ホストモード

STATTXビットにはチャネルのステータスに関する情報が格納されます。詳細については、表736（「ホストモード」の説明）を参照してください。デバイスモードではこれらのビットには次のトランザクションで割り当てられるステータスが格納されますが、ホストモードではデバイスから前回受信したステータスが格納されます。NAKを受信した場合、STATTXにはNAKを示す値が格納されます。

ビット3:0 **EA[3:0]**：エンドポイント/チャネルアドレス

– デバイスモード

ソフトウェアは、このエンドポイント宛てのトランザクションを識別するために使用される4bitのアドレスをこのフィールドに書き込む必要があります。値は、対応するエンドポイントを有効にする前に書き込まれる必要があります。

– ホストモード

ソフトウェアは、ホストトランザクションによってアドレス指定されたチャネルを識別するために使用される4bitのアドレスをこのフィールドに書き込む必要があります。

表733.受信ステータスエンコード

STATRX[1:0]	意味
00	**DISABLED**：このエンドポイント/チャネルにアドレス指定されたすべての受信リクエストは無視されます。
01	**STALL**： デバイスモード：エンドポイントは停止していて、すべての受信リクエストはSTALLハンドシェイクで終了します。 ホストモード：デバイスでチャネルが停止していることを示します。
10	**NAK**： デバイスモード：エンドポイントはNAKされ、すべての受信リクエストはNAKハンドシェイクで終了します。 ホストモード：デバイスで受信リクエストがNAKしていることを示します。
11	**VALID**：このエンドポイント/チャネルは受信に有効です。

表734.エンドポイント/チャネルタイプのエンコード

UTYPE[1:0]	意味
00	BULK
01	CONTROL
10	ISO
11	INTERRUPT

表735.エンドポイント/チャネルの種類の意味

UTYPE[1:0]		EPKINDの意味
00	BULK	DBL_BUF
01	CONTROL	STATUS_OUT
10	ISO	SBUF_ISO：このビットは、アイソクロナスエンドポイントのシングルバッファリング機能を有効にするために、ソフトウェアによってセットされます
11	INTERRUPT	未使用

表736.送信ステータスエンコード

STATTX[1:0]	意味
00	**DISABLED**：このエンドポイント/チャネルにアドレス指定されたすべての送信リクエストは無視されます。
01	**STALL**： デバイスモード：エンドポイントは停止していて、すべての送信リクエストはSTALLハンドシェイクで終了します。 ホストモード：デバイスでチャネルが停止していることを示します。
10	**NAK**： デバイスモード：エンドポイントはNAKされ、すべての送信リクエストはNAKハンドシェイクで終了します。 ホストモード：デバイスで送信リクエストがNAKしていることを示します。
11	**VALID**：このエンドポイント/チャネルは送信に有効です。

71.6.2 バッファディスクリプタテーブル

注： バッファディスクリプタテーブルは、別の「USB SRAM」アドレス空間のパケットバッファメモリ内に位置しています。

バッファディスクリプタテーブルはパケットバッファメモリ（「USB SRAM」領域）内にありますが、そのエントリは、USBマクロセルとデバイスの間でデータを交換するためのパケットバッファの位置とサイズを設定する追加レジスタとみなすことができます。

最初のパケットメモリ位置は、USB SRAMベースアドレスにあります。USB_CHEPnRレジスタに関連するバッファディスクリプタテーブルのエントリについて、以下に説明します。メモリはワード（32bit）アクセスを使用してアドレス指定する必要があります。

パケットバッファとバッファディスクリプタテーブルの使用に関する詳細については、3056ページのパケットバッファの構造と用途を参照してください。

チャネル/エンドポイント送信バッファディスクリプタn（USB_CHEP_TXRXBD_n）

アドレスオフセット：n*8

このレジスタの説明が適用されるのは、該当するCHEPnRレジスタにより、受信モードで動作するダブルバッファリングの使用がプログラムされない場合です（それ以外の場合は、次のレジスタの説明を参照してください）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	COUNT_TX[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_TX[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **COUNT_TX[9:0]**：送信バイトカウント

これらのビットには、アドレス指定された次のINトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって送信されるバイト数が格納されます。

ビット15:0 **ADDR_TX[15:0]**：送信バッファアドレス

これらのビットは、アドレス指定された次のINトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって送信されるデータが格納されたパケットバッファの開始アドレスを指します。パケットメモリはワード長であり、すべてのパケットバッファがワードで揃えられている必要があるため、ビット1および0には常に00が書き込まれる必要があります。

チャネル/エンドポイント受信バッファディスクリプタn［オルタネート］（USB_CHEP_TXRXBD_n）

アドレスオフセット：n*8

このレジスタの説明が適用されるのは、該当するCHEPnRレジスタにより、ダブルバッファリングの使用がプログラムされ、受信バッファが有効化される場合です（それ以外の場合は、前のレジスタの説明を参照してください）。

このテーブル位置は、両方ともパケット受信時に必要とされる2つの値を格納するために使用されます。最上位ビットは割り当てられるバッファサイズの定義を含み、バッファオーバーフロー検出を可能にし、この位置の最下位部分には、受信終了時にUSBペリフェラルによって実際の受信バイト数が書き込まれます。使用可能なビット数が制限されているため、バッファサイズは割り当てられたメモリブロック数を使用して表され、細かい粒度／小さいバッファと粗い粒度／大きなバッファの間のトレードオフにより、ブロックサイズを選択できます。割り当てられたバッファのサイズは、エンドポイント / チャネルディスクリプタの一部であり、通常、エニュメレーションプロセス時にmaxPacketSizeパラメータ値に従って定義されます（「ユニバーサルシリアルバス仕様」を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLSIZE	NUM_BLOCK[4:0]					COUNT_RX[9:0]									
rw	rw	rw	rw	rw	rw	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_RX[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **BLSIZE**：ブロックサイズ

このビットは、割り当てられたバッファ領域を定義するために使用されるメモリブロックのサイズを選択します。

- BLSIZE =0の場合、メモリブロックは2バイト長であり、これは、ハーフワード長のメモリで可能な最小ブロックです。このブロックサイズでは、割り当てられるバッファサイズは2 ～ 62バイトです。
- BLSIZE =1の場合、メモリブロックは32バイト以上であり、USB仕様によって定義された最大パケット長に達することができます。このブロックサイズでは、割り当てられるバッファサイズは、理論上、32 ～ 1024バイトであり、USB標準仕様で可能な最長のパケットサイズです。ただし、適用可能なサイズは、使用可能なバッファメモリによって制限されます。

ビット30:26 **NUM_BLOCK[4:0]**：ブロック数

これらのビットは、このパケットバッファに割り当てられるメモリブロック数を定義します。実際の割り当てメモリの量は、表737に示されているように、BLSIZEの値に依存します。

ビット25:16 **COUNT_RX[9:0]**：受信バイトカウント

これらのビットには、USB_CHEPnRレジスタに関連するエンドポイント/チャネルによって、アドレス指定された最後のOUT/SETUPトランザクション中に受信されたバイト数が格納されます。

ビット15:0 **ADDR_RX[15:0]**：受信バッファアドレス

これらのビットは、アドレス指定された次のOUT/SETUPトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって受信されるデータが格納されたパケットバッファの開始アドレスを指します。パケットメモリはワード長であり、すべてのパケットバッファがワードで揃えられている必要があるため、ビット1および0には常に00が書き込まれる必要があります。

表737.割り当てられるバッファメモリの定義

NUM_BLOCK[4:0] の値	BLSIZE =0のとき割り当てられるメモリ	BLSIZE =1のとき割り当てられるメモリ
0(00000)	設定禁止	32バイト
1(00001)	2バイト	64バイト
2 (00010)	4バイト	96バイト
3(00011)	6バイト	128バイト
.....	.....	.....
14(01110)	28バイト	480バイト
15 (01111)	30バイト	
16 (10000)	32バイト	
.....	.....	.....
29 (11101)	58バイト	.....
30(11110)	60バイト	992バイト
31 (11111)	62バイト	1023バイト

チャネル/エンドポイント受信バッファディスクリプタn（USB_CHEP_RXTXBD_n）

アドレスオフセット：n*8+4

このレジスタの説明が適用されるのは、該当するCHEPnRレジスタにより、送信モードでダブルバッファリングの使用がプログラムされない場合です（それ以外の場合は、次のレジスタの説明を参照してください）。

このテーブル位置は、両方ともパケット受信時に必要とされる2つの値を格納するために使用されます。最上位ビットは割り当てられるバッファサイズの定義を含み、バッファオーバーフロー検出を可能にし、この位置の最下位部分には、受信終了時にUSBペリフェラルによって実際の受信バイト数が書き込まれます。使用可能なビット数が制限されているため、バッファサイズは割り当てられたメモリブロック数を使用して表され、細かい粒度／小さいバッファと粗い粒度／大きなバッファの間のトレードオフにより、ブロックサイズを選択できます。割り当てられたバッファのサイズは、エンドポイント / チャネルディスクリプタの一部であり、通常、エニュメレーションプロセス時にmaxPacketSizeパラメータ値に従って定義されます（「ユニバーサルシリアルバス仕様」を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLSIZE	NUM_BLOCK[4:0]					COUNT_RX[9:0]									
rw	rw	rw	rw	rw	rw	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_RX[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **BLSIZE**：ブロックサイズ

このビットは、割り当てられたバッファ領域を定義するために使用されるメモリブロックのサイズを選択します。

- BLSIZE =0の場合、メモリブロックは2バイト長であり、これは、ハーフワード長のメモリで可能な最小ブロックです。このブロックサイズでは、割り当てられるバッファサイズは2 ～ 62バイトです。
- BLSIZE =1の場合、メモリブロックは32バイト以上であり、USB仕様によって定義された最大パケット長に達することができます。このブロックサイズでは、割り当てられるバッファサイズは、理論上、32 ～ 1024バイトであり、USB標準仕様で可能な最長のパケットサイズです。ただし、適用可能なサイズは、使用可能なバッファメモリによって制限されます。

ビット30:26 **NUM_BLOCK[4:0]**：ブロック数

これらのビットは、このパケットバッファに割り当てられるメモリブロック数を定義します。実際の割り当てメモリの量は、表737に示されているように、BLSIZEの値に依存します。

ビット25:16 **COUNT_RX[9:0]**：受信バイトカウント

これらのビットには、USB_CHEPnRレジスタに関連するエンドポイント/チャネルによって、アドレス指定された最後のOUT/SETUPトランザクション中に受信されたバイト数が格納されます。

ビット15:0 **ADDR_RX[15:0]**：受信バッファアドレス

これらのビットは、アドレス指定された次のOUT/SETUPトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって受信されるデータが格納されたパケットバッファの開始アドレスを指します。パケットメモリはワード長であり、すべてのパケットバッファがワードで揃えられている必要があるため、ビット1および0には常に00が書き込まれる必要があります。

チャネル/エンドポイント送信バッファディスクリプタn［オルタネート］（USB_CHEP_RXTXBD_n）

アドレスオフセット：n*8+4

このレジスタの説明が適用されるのは、該当するCHEPnRレジスタにより、ダブルバッファリングの使用がプログラムされ、送信バッファが有効化される場合です（それ以外の場合は、前のレジスタの説明を参照してください）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	COUNT_TX[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_TX[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:16 **COUNT_TX[9:0]**：送信バイトカウント

これらのビットには、アドレス指定された次のINトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって送信されるバイト数が格納されます。

ビット15:0 **ADDR_TX[15:0]**：送信バッファアドレス

これらのビットは、アドレス指定された次のINトークンでUSB_CHEPnRレジスタに関連するエンドポイント/チャネルによって送信されるデータが格納されたパケットバッファの開始アドレスを指します。パケットメモリはワード長であり、すべてのパケットバッファがワードで揃えられている必要があるため、ビット1および0には常に00が書き込まれる必要があります。

71.6.3 USBレジスタマップ

次の表に、USBレジスタマップとリセット値を示します。

表738.USBレジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**USB_CHEP0R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**USB_CHEP1R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**USB_CHEP2R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**USB_CHEP3R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**USB_CHEP4R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**USB_CHEP5R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**USB_CHEP6R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**USB_CHEP7R**	Res.	THREE_ERR_RX [1:0]		THREE_ERR_TX [1:0]		ERR_RX	ERR_TX	LS_EP	NAK	DEVADDR[6:0]							VTRX	DTOGRX	STATRX [1:0]		SETUP	UTYPE [1:0]		EPKIND	VTTX	DTOGTX	STATTX [1:0]		EA[3:0].			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20～0x3F	予約済みです。																																
0x40	**USB_CNTR**	HOST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DDISCM	THR512M	CTRM	PMAOVRM	ERRM	WKUPM	SUSPM	RST_DCONM	SOFM	ESOFM	L1REQM	Res.	L1RES	L2RES	SUSPEN	SUSPRDY	PDWN	USBRST
	リセット値	0														0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	1	1

表738.USBレジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	**USB_ISTR**	Res.	LS_DCON	DCON_STAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DDISC	THR512	CTR	PMAOVR	ERR	WKUP	SUSP	RST_DCON	SOF	ESOF	L1REQ	Res.	Res.	DIR	IDN[3:0]			
	リセット値		0	0												0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0
0x48	**USB_FNR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDP	RXDM	LCK	LSOF [1:0]		FN[10:0].										
	リセット値																	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x
0x4C	**USB_DADDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EF	ADD[6:0]						
	リセット値																									0	0	0	0	0	0	0	0
0x54	USB_LPMCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BESL[3:0]				REMWAKE	Res.	LPMACK	LPMEN
	リセット値																									0	0	0	0	0		0	0
0x58	USB_BCDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DPPU_DPD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET	SDEN	PDEN	DCDEN	BCDEN
	リセット値																	0								0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

72 USB On-The-Go フルスピード （OTG_FS）

このセクションは、STM32U575/585デバイスにのみ適用されます。

72.1 概要

Portions Copyright (c) Synopsys, Inc. All rights reserved.掲載許可取得済み。

このセクションでは、OTG_FS コントローラのアーキテクチャとプログラミングモデルについて説明します。

このセクションでは、次の略記が使用されます。

FS	フルスピード
LS	ロースピード
MAC	メディアアクセスコントローラ
OTG	On-The-Go
PFC	パケットFIFOコントローラ
PHY	物理層
USB	ユニバーサルシリアル
UTMI	USB 2.0トランシーバマクロセルインタフェース（UTMI）
LPM	リンク電源管理
BCD	バッテリ充電検出回路
HNP	ホストネゴシエーションプロトコル
SRP	セッションリクエストプロトコル

このセクションでは、参照文献として以下の文書が使用されています。

- USB On-The-Go Supplement, Revision 2.0
- Universal Serial Bus Revision 2.0 Specification
- USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007
- USB 2.0 ECNの正誤表：リンク電源管理（LPM）- 2007年7月
- バッテリ充電（BC）仕様Revision 1.2

USB OTGは、デバイスとホストの両方の機能をサポートするデュアルロールデバイス（DRD）コントローラであり、On-The-Go Supplement to the USB 2.0 Specificationに完全に準拠しています。USB OTGは、USB 2.0 Specificationに完全に準拠したホスト専用またはデバイス専用のコントローラとして設定することも可能です。OTG_FSは、以下の表 739：サポートされるOTG_FS速度で定義されている速度に対応しています。USB OTGはHNPとSRPの両方をサポートしています。必要な外部デバイスは、OTGモードで使用されるV_{BUS} 用のチャージポンプのみです。

表739.サポートされるOTG_FS速度

-	HS（480Mb/s）	FS（12Mb/s）	LS（1.5Mb/s）
ホストモード	-	X	X
デバイスモード	-	X	-

72.2 OTG_FS の主な機能

主な機能は、標準機能、ホストモード機能、およびデバイスモード機能の3つのカテゴリに分けて考えることができます。

72.2.1 標準機能

OTG_FS インタフェースの機能概要を次に示します:

- Universal Serial Bus Specification Rev 2.0に準拠したUSB-IFです。
- OTG_FSは、以下のPHYインタフェースをサポートします。
  - フルスピードPHYを内蔵
- On-The-Go Supplement Rev 2.0 Specificationに詳細が規定されているオプションのOn-The-Go（OTG）プロトコルの完全サポート（PHY）を含みます。
  - A-Bデバイス認識（IDライン）をサポートしています。
  - ホストネゴシェーションプロトコル（HNP）およびセッションリクエストプロトコル（SRP）を統合的にサポートしています。
  - OTGアプリケーションでバッテリを節約するために、ホストでV_{BUS} をオフすることができます。
  - 内部コンパレータによってOTGにV_{BUS} のレベルを監視させることができます。
  - ホスト-ペリフェラルのダイナミックな役割切り替えをサポートします。
- 次の動作を行うようにソフトウェアでの設定が可能です。
  - SRP対応のUSB FS ペリフェラル（Bデバイス）
  - SRP対応のUSB FS/LSホスト（Aデバイス）
  - USB On-The-Goフルスピードデュアルロールデバイス
- 以下の要素により FS SOFおよびLS Keep-aliveをサポートします。
  - SOFパルスPAD接続性
  - タイマ（TIMx）に対するSOFパルス内部接続
  - 設定可能なフレーム周期
  - 設定可能なフレームの終わりの割込み
- USBサスペンド中のシステム停止、デジタルコア、PHY、およびDFIFO 電源管理回路内部のクロックドメインの電源オフなど、節電機能を内蔵しています。
- 以下のような高度なFIFO制御を備えた専用の 1.25KB RAMを備えています。
  - RAM空間を複数のFIFOに分割して、RAM を柔軟かつ効率的に使用できるように設定できます。
  - 各FIFOは複数のパケットを保持できます。
  - 動的なメモリ割り当てが可能です。
  - FIFOを2のべき乗でないサイズに設定して、連続的なメモリ位置の使用が可能です。
- システムの介入なしで、最大1フレーム（1ms）の間、最大USBバンド幅を保証します。
- バッテリ充電仕様Revision 1.2で説明されているとおり、充電ポート検出をサポートしています。

72.2.2 ホストモード機能

ホストモードでのOTG_FS インタフェースの主な機能と条件は次の通りです:

- V_{BUS} 電圧発生用の外部チャージポンプを持っています。
- 最大 12 個のホストチャネル（パイプ）を持っています。各チャネルはどんなタイプのUSB転送も割り当てられるように動的に設定可能です。
- 次を保持するハードウェアスケスケジューラを内蔵しています。
  - 最大12　個の割込みに加えて、周期的ハードウェアキュー内でのアイソクロナス転送リクエスト
  - 非周期的ハードウェアキュー内で、最大 12 つの制御転送およびバルク転送リクエスト
- USBデータRAMの効率的な使用のために、共有Rx FIFO、周期的Tx FIFO、および非周期的Tx FIFOを管理します。

72.2.3 ペリフェラルモード機能

ペリフェラルモードでの OTG_FS インタフェースの主要機能を次に示します:

- 1つの双方向制御エンドポイント0を持っています。
- バルク、割込み、またはアイソクロナス転送をサポートするために、設定可能な5 つのINエンドポイント（EP）を持っています。
- バルク、割込み、またはアイソクロナス転送をサポートするために、設定可能な5 つのOUTエンドポイント（EP）を持っています。
- USBデータRAMの効率的な使用のために、共有RX FIFOとTx-OUT FIFOを管理します。
- アプリケーションの負荷を軽減するために、最大 6 つの専用Tx-IN FIFO（有効なIN EPに対して1つずつ）を管理します。
- ソフト切断機能をサポートしています。

72.3 OTG_FS の実装

表740.OTG_FS の実装(1)

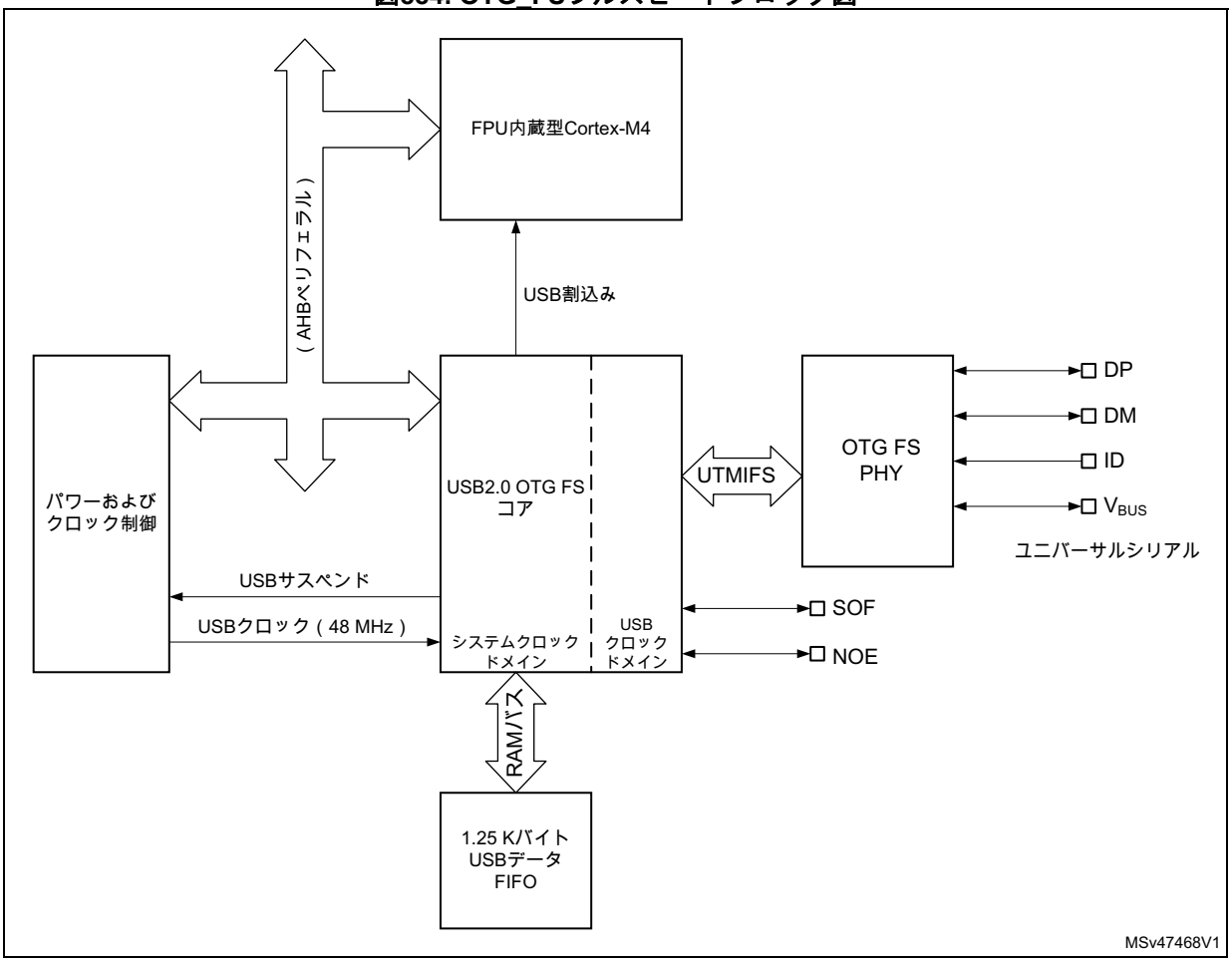
USBの機能	OTG_FS （STM32U575/585）
デバイスの双方向エンドポイント（EP0を含む）	6
ホストモードチャネル	12
専用SRAMのサイズ	1.2KB
USB 2.0リンク電源管理（LPM）のサポート	X
サポートされているOTGレビジョン	2.0
バッテリ充電検出（BCD）のサポート	X
統合PHY	FS

1. 「X」はサポート対象、「-」はサポート対象外です。「FS」はFSモードでサポートされ、「HS」はHSモードでサポートされます。

72.4 OTG_FS の機能説明

72.4.1 OTG_FS のブロック図

図884. OTG_FSフルスピードブロック図



72.4.2 OTG_FS ピン および内部信号

表741.OTG_FSの入出力ピン

信号名	信号タイプ	説明
OTG_FS_DP	デジタル入出力	USB OTG D+ ライン
OTG_FS_DM	デジタル入出力	USB OTG D- ライン
OTG_FS_ID	デジタル入力	USB OTG ID
OTG_FS_VBUS	アナログ入力	USB OTG VBUS
OTG_FS_SOF	デジタル出力	USB OTGフレーム開始（可視性）
OTG_FS_NOE	デジタル出力	USB OTG D+/D- 用出力の有効化（可視性）

表742.OTG_FS 入力／出力信号

信号名	信号タイプ	説明
usb_sof	デジタル出力	オンチップペリフェラルのUSB OTGフレーム開始イベント
usb_wkup	デジタル出力	USB OTG ウェイクアップイベント出力
usb_gbl_it	デジタル出力	USB OTGグローバル割込み

72.4.3 OTG_FS コア

The USB OTG_FS は、リセットおよびクロックコントローラ（RCC）から48MHzのクロックを受け取ります。このクロックは、フルスピード（12 Mbit/s）で48 MHzドメインを駆動するために使用され、OTGコアを設定する前に有効にされなければなりません。

CPUはAHBペリフェラルバスを通じてOTG Coreレジスタの読み書きを行います。これは、セクション 72.13 : OTG_FS 割込みで説明されている単一のUSB OTG割込みを介してUSBイベントとして通知されます。

CPUは、専用のOTGアドレス（プッシュレジスタ）に32bitのワードを書き込むことによって、USBを介してデータを提供します。データはUSBデータRAM内に設定されたTxデータFIFOに自動的に格納されます。各INエンドポイント（ペリフェラルモードの場合）またはOUTチャネル（ホストモードの場合）に対して1つのTx FIFOプッシュレジスタがあります。

CPUは、専用のOTGアドレス（ポップレジスタ）から32bitのワードを読み出すことによって、USBからデータを受け取ります。データは、1.25 KBのUSBデータRAM内に設定された共有Rx FIFOから自動的に取得されます。各OUTエンドポイントまたはINチャネルに対して1つのRx FIFOポップレジスタがあります。

USBプロトコル層は、シリアルインタフェースエンジン（SIE）によって駆動され、オンチップ物理層（PHY）内のトランシーバモジュールによってUSB上で直列化（シリアライズ）されます。

注意： **USB OTG FSペリフェラルの正常動作を保証するためには、AHB周波数を14.2 MHzより高くする必要があります。**

72.4.4 OTG-FSに接続された内蔵フルスピードOTG PHY

組み込みのフルスピードOTG PHYは、OTG FSコアによって制御され、UTMI+ バス（UTMIFS）のフルスピードサブセットを通じてUSB制御信号およびデータ信号を運びます。USB接続の物理サポートを提供します。
フルスピードOTG PHYには、以下のコンポーネントが含まれます。

- ホストとデバイスの両方で使用されるFS/LSトランシーバモジュール。シングルエンドUSBライン上で送受信を直接駆動します。
- デバイスの現在の役割に依存してOTG_FSコアによって制御される 組み込みのDP/DMプルアップおよびプルダウン抵抗。ペリフェラルとして、DPプルアップ抵抗を有効にし、フルスピードペリフェラル接続を知らせます。信号を送るタイミングは、V_{BUS} が有効なレベルであると判断されてすぐです（Bセッションが有効な場合）。ホストモードでは、DPとDMの両方でプルダウン抵抗が有効にされます。プルアップ抵抗とプルダウン抵抗は、デバイスの役割がホストネゴシエーションプロトコル（HNP）を通じて変更されたとき、動的に切り替えられます。
- プルアップ／プルダウン抵抗ECN回路。DPプルアップ回路は、USB Rev2.0に適用される抵抗の技術変更通知（Engineering Change Notice）に従って、OTG_FSから個別に制御される2つの抵抗で構成されます。DPプルアップの動的なトリミングは、ノイズの低減とTx/Rx信号品質の向上を可能にします。

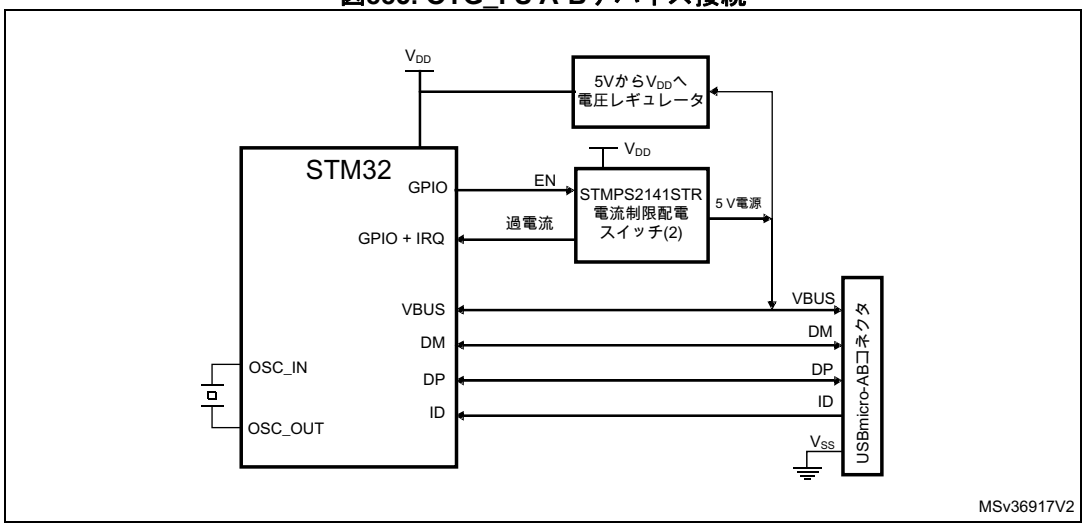
72.4.5 OTGの検出

また、OTG_FS では次の機能を使用します。

- A/B デバイス識別のための ID ラインをサンプリングするために使用される組み込みの ID プルアップ抵抗。
- V_{BUS} レベル検出用コンパレータ。V_{BUS} の有効性、A-Bセッションの有効性、およびセッション終了電圧閾値を検出するために使用されるヒステリシスを有します。これらは、セッションリクエストプロトコル（SRP）の駆動、有効な起動およびセッション終了条件の検出、およびUSB操作中のV_{BUS} 電源の定常的な監視のために使用されます。

72.5 OTG_FS デュアルロールデバイス（DRD）

図885. OTG_FS A-Bデバイス接続



1. VBUSで電源が供給されるデバイスを設定するときだけ必要な外部電圧レギュレータ
2. アプリケーションが VBUS で電源が供給されるデバイスをサポートしなければならない場合のみ必要な STMPS2141STR。アプリケーションボード上で5 Vが使用可能な場合、基本的なパワースイッチを使用することができます。

72.5.1 IDラインの検出

ホストかペリフェラル（デフォルト）かの役割は、ID入力ピンに応じて割り当てられます。IDラインのステータスは、USBケーブルでの挿入時に、MicroAまたはMicroBプラグがマイクロABレセプタクルに接続されたかによって決まります。

- USBケーブルのB側がフローティングIDワイヤに接続された場合、組み込みのプルアップ抵抗がIDラインのハイレベルを検出して、デフォルトのペリフェラルの役割が確立されます。この構成では、OTG_FS はセクション4.2.4で説明する標準FSM、すなわちOn-The-Go仕様Rev 2.0のIDピン（USB2.0への補完）に準拠しています。
- USBケーブルのA側がアース付きIDラインに接続された場合、OTG_FS は、ホストのソフトウェア初期化のためにIDラインステータス変化割込み（OTG_GINTSTSのCIDSCHGビット）を発行して、自動的にホストの役割に切り替わります。この構成では、OTG_FS はセクション4.2.4で説明する標準FSM、すなわちOn-The-Go仕様Rev 2.0のIDピン（USB2.0への補完）に準拠しています。

72.5.2 HNPデュアルロールデバイス

グローバルUSB設定レジスタのHNP対応ビット（OTG_ GUSBCFGのHNPCAPビット）は、OTG_FSコアがホストネゴシエーションプロトコル（HNP）に従って、役割をAホストからAペリフェラルに（またその逆）、またはBペリフェラルからBホスト（またその逆）に動的に変更できるようにします。現在のデバイスのステータスは、グローバルOTG制御およびステータスレジスタのコネクタIDステータスビット（OTG_GOTGCTLのCIDSTSビット）とグローバル割込みおよびステータスレジスタの現在の動作モードビット（OTG_GINTSTSのCMODビット）の値の組み合わせによって読み出すことができます。

HNPプログラミングモデルの詳細は、セクション 72.16 : OTG_FS プログラミングモデルに記載されています。

72.5.3 SRPデュアルロールデバイス

グローバルUSB設定レジスタのSRP対応ビット（OTG_GUSBCFGのSRPCAPビット）は、OTG_FSコアがAデバイスの節電のためにV_{BUS} の生成を停止できるようにします。Aデバイスは OTG_FS の役割がホストかペリフェラルかに関係なく、常に、V_{BUS} を駆動する役割を持っていることに注意してください。

SRP A/Bデバイスのプログラミングモデルの詳細は、セクション 72.16 : OTG_FS プログラミングモデルに記載されています。

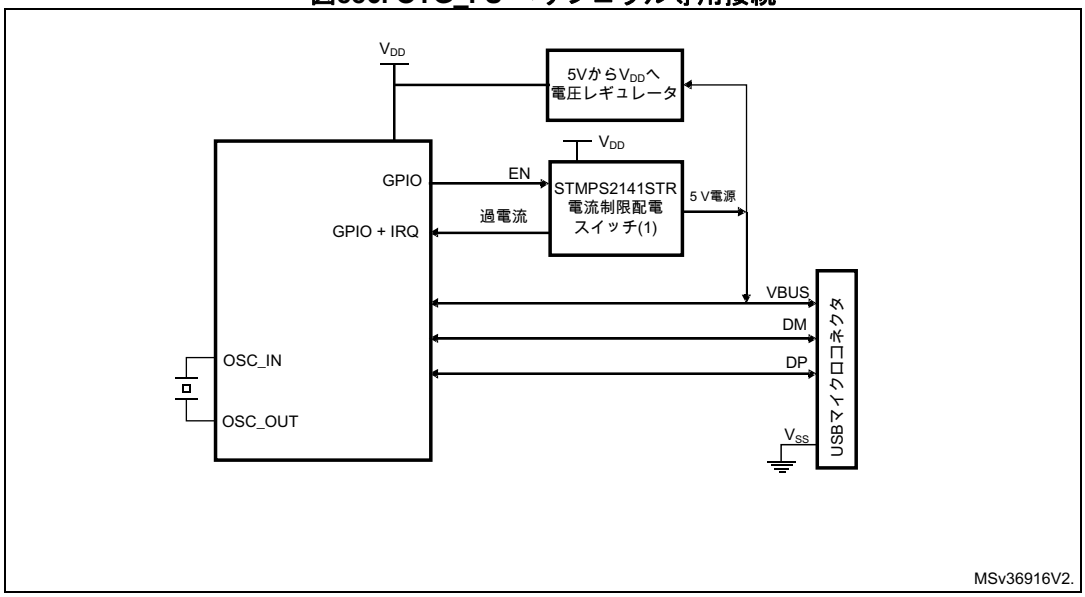
72.6 USBペリフェラルとしての OTG_FS

このセクションでは、USBペリフェラルモードでの OTG_FS の機能について説明します。OTG_FSは、以下の環境でUSBペリフェラルとして機能します。

- OTG Bペリフェラル
  - USBケーブルのB側が挿入された場合のOTG Bデバイスのデフォルトの状態
- OTG Aペリフェラル
  - HNPが OTG_FS の役割をペリフェラルに切り替えた後のOTG Aデバイスの状態
- Bデバイス
  - IDラインが存在し、機能していて、USBケーブルのB側に接続され、かつ、グローバルUSB設定レジスタのHNP対応ビット（OTG_GUSBCFGのHNPCAPビット）がクリアされている場合。
- ペリフェラルのみ （ 図 886 : OTG_FS ペリフェラル専用接続を参照）
  - セクション 72.15.4 : OTG USB設定レジスタ（OTG_GUSBCFG）の強制デバイスモードビット（FDMOD）が1にセットされると、OTG_FS コアはUSBペリフェラル専用として機能するよう強制されます。この場合、IDラインは、USBコネクタ上に存在していても無視されます。

注 : B デバイスまたはペリフェラル専用設定の場合に電源がバスから供給されるデバイス (bus-powered device)を実装するには、V_{BUS} から必要な電力を生成する外部レギュレータを追加する必要があります。

図886. OTG_FS ペリフェラル専用接続



1. バスから電源が供給されるデバイスを構築するには、レギュレータを使用します。

72.6.1 SRP対応ペリフェラル

グローバルUSB設定レジスタのSRP対応ビット（OTG_GUSBCFGのSRPCAPビット）は、OTG_FSコアがセッションリクエストプロトコル（SRP）をサポートできるようにします。この方法により、USBセッションがサスペンド状態にある間、V_{BUS} の供給を停止して、リモートAデバイスの消費電力を節約できます。

SRPペリフェラルモードのプログラミングモデルの詳細は、Bデバイスセッションリクエストプロトコル のセクションに記載されています。

72.6.2 ペリフェラルの状態

電源が供給された状態

V_{BUS} 入力は、USBペリフェラルが電源供給状態に入ることが許されるBセッション有効電圧を検出します（USB 2.0のセクション9.1を参照）。その場合、OTG_FS は、DPプルアップ抵抗を自動的に接続して、フルスピードデバイス接続をホストに知らせ、セッションリクエスト割込みを生成して（OTG_GINTSTSのSRQINTビット）、電源供給状態を通知します。

V_{BUS} 入力は、また、USB動作中にホストによって適切なV_{BUS} レベルが供給されることを保証します。V_{BUS} 電源がBセッションの有効値以下に低下したことが検出された場合（たとえば、電源の変動があった場合や、ホストポートの電源がオフとなった場合など）、OTG_FS は自動的に切断され、セッション終了検出割込み（OTG_GOTGINTのSEDETビット）が生成されて、OTG_FS が電源供給状態でなくなったことを通知します。

電源供給状態では、OTG_FS は、ホストからリセットシグナリングを受け取ることを期待しています。他のUSB操作を実行することはできません。リセットシグナリングを受け取ると、リセット検出割込み（OTG_GINTSTSのUSBRSTビット）が生成されます。リセットシグナリングが完了すると、エミュレーション終了割込み（OTG_GINTSTSのENUMDNEビット）が生成され、OTG_FS はデフォルト状態に入ります。

ソフト切断

電源供給状態は、ソフト切断機能によってソフトウェアで終了させることができます。DPプルアップ抵抗は、デバイス制御レジスタのソフト切断ビット（OTG_DCTLのSDISビット）をセットすることによって取り除かれ、USBケーブルが実際にはホストポートから取り除かれていなくても、ホスト側にデバイス切断検出割込みを生成します。

デフォルト状態

デフォルト状態では、OTG_FS はホストからSET_ADDRESSコマンドを受け取ることを期待しています。他のUSB操作を実行することはできません。有効なSET_ADDRESSコマンドがUSB上でデコードされると、アプリケーションは対応する番号をデバイス設定レジスタのデバイスアドレスフィールドに書き込みます（OTG_DCFGのDADビット）。その場合、OTG_FS アドレス状態に入り、設定されたUSBアドレスでホストトランザクションに応答する準備が整います。

サスペンド状態

OTG_FS ペリフェラルは、USBの活動状態を定常的に監視します。USBのアイドル状態が3 ms続くと、早期サスペンド割込み（OTG_GINTSTSのESUSPビット）が発行され、適切な場合は3 ms後にサスペンド割込み（OTG_GINTSTSのUSBSUSPビット）によって確認されます。デバイスステータスレジスタのデバイスサスペンドビット（OTG_DSTSのSUSPSTSビット）が自動的にセットされ、OTG_FS はサスペンド状態に入ります。

サスペンド状態は、デバイス自身によって終了することもできます。この場合、アプリケーションはデバイス制御レジスタのリモートウェイクアップシグナリングビット（OTG_DCTLのRWUSIGビット）をセットし、1 ~ 15 ms後にクリアします。

ホストからレジュームシグナリングが検出されると、レジューム割込み（OTG_GINTSTSのWKUPINTビット）が生成され、デバイスサスペンドビットは自動的にクリアされます。

72.6.3 ペリフェラルエンドポイント

OTG_FS コアは、以下のUSBエンドポイントをインスタンス化します。

- 制御エンドポイント0 :
  - 双方向であり、制御メッセージのみを取り扱います。
  - 入力および出力トランザクションを処理する個別のレジスタセットを備えています。
  - 固有の制御レジスタ（OTG_DIEPCTL0/OTG_DOEPCTL0）、転送設定レジスタ（OTG_DIEPTSIZ0/OTG_DOEPTSIZ0）、およびステータス割込みレジスタ（OTG_DIEPINT0/OTG_DOEPINT0）を備えています。制御および転送サイズレジスタ内で使用可能なビットのセットは、他のエンドポイントとは少し違います。
- 5 IN エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DIEPCTL*x*）、転送設定レジスタ（OTG_DIEPTSIZ*x*）、およびステータス割込みレジスタ（OTG_DIEPINT*x*）を備えています。
  - デバイスINエンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）を使用して、すべてのINエンドポイント（EP0も含む）上で1種類のエンドポイント割込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナスINエンドポイントが少なくとも1つあるときにアサートされる、不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFRビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTSのEOPF）の終わりとともにアサートされます。

- 5 OUT エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DOEPCTL*x*）、転送設定レジスタ（OTG_DOEPTSIZ*x*）、およびステータス割込みレジスタ（OTG_DOEPINT*x*）を備えています。
  - デバイスOUTエンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）を使用して、すべてのOUTエンドポイント（EP0も含む）上で1種類のエンドポイント割込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナスOUTエンドポイントが少なくとも1つあるときにアサートされる、不完全アイソクロナスOUT転送割込み（OTG_GINTSTSのINCOMPISOOUTビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTSのEOPF）の終わりとともにアサートされます。

エンドポイント制御

- 以下のエンドポイント制御は、デバイスエンドポイント-*x* IN/OUT制御レジスタ（OTG_DIEPCTL*x*/OTG_DOEPCTL*x*）を通じて、アプリケーションから使用できます。
  - エンドポイント有効／無効
  - 現在の設定でのエンドポイントのアクティブ化
  - USB転送タイプ（アイソクロナス、バルク、割込み）のプログラミング
  - サポートされるパケットサイズのプログラミング
  - INエンドポイントに関連するTx FIFO番号のプログラミング
  - 期待される、または送信されるデータ0／データ1 PID（バルク／割込みのみ）のプログラミング
  - トランザクションが送受信される偶数／奇数フレームのプログラミング（アイソクロナス転送のみ）
  - オプションで、FIFOのステータスに関係なく、常にホストに否定応答するNAKビットのプログラミング
  - オプションで、エンドポイントに対するホストトークンを常に停止するSTALLビットのプログラミング
  - オプションで、OUTエンドポイントが受信データのCRCフィールドをチェックしないようにするSNOOPモードのプログラミング

エンドポイント転送

アプリケーションは、デバイスエンドポイント-*x* 転送サイズレジスタ（OTG_DIEPTSIZ*x*/OTG_DOEPTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。エンドポイント制御レジスタのエンドポイントイネーブルビットをセットする前にプログラミングを行う必要があります。エンドポイントが有効になると、これらのフィールドは、OTG_FS コアが現在の転送ステータスで更新すると、読出し専用になります。

以下の転送パラメータがプログラム可能です。

- 転送サイズ（バイト単位）
- 全体の転送サイズを構成するパケットの数

エンドポイントのステータス／割込み

デバイスエンドポイント-*x* 割込みレジスタ（OTG_DIEPINT*x*/OTG_DOPEPINT*x*）は、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタのOUTエンドポイント割込みビットまたはINエンドポイント割込みビット（OTG_GINTSTSのOEPINTビット、またはOTG_GINTSTSのIEPINTビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、デバイスオールエンドポイント割込み（OTG_DAINT）レジスタを読み出して、デバイスエンドポイント-*x* 割込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

ペリフェラルコアは、以下のステータスチェックおよび割込み生成を行います。

- データ転送がアプリケーション側（AHB）とUSB側の両方で完了したことを示す転送完了割込み
- セットアップステージが終了しているかどうか（制御OUTのみ）
- 関連する送信FIFOが半分エンプティか完全エンプティか（INエンドポイント）
- NAK応答がホストに送信されたかどうか（アイソクロナスINのみ）
- Tx FIFOがエンプティのときINトークンが受け取られたかどうか（バルクIN/割込みINのみ）
- エンドポイントがまだ有効になっていないときにOUTトークンを受け取ったかどうか
- バブルエラー条件が検出されたかどうか
- アプリケーションによるエンドポイントの無効化が可能かどうか
- アプリケーションによるエンドポイントNAKが可能かどうか（アイソクロナスINのみ）
- 3つを超える連続したセットアップパケットが受信されたかどうか（制御OUTのみ）
- タイムアウト条件が検出されたかどうか（制御INのみ）
- アイソクロナスOUTパケットが割込みを生成せずにドロップされたかどうか

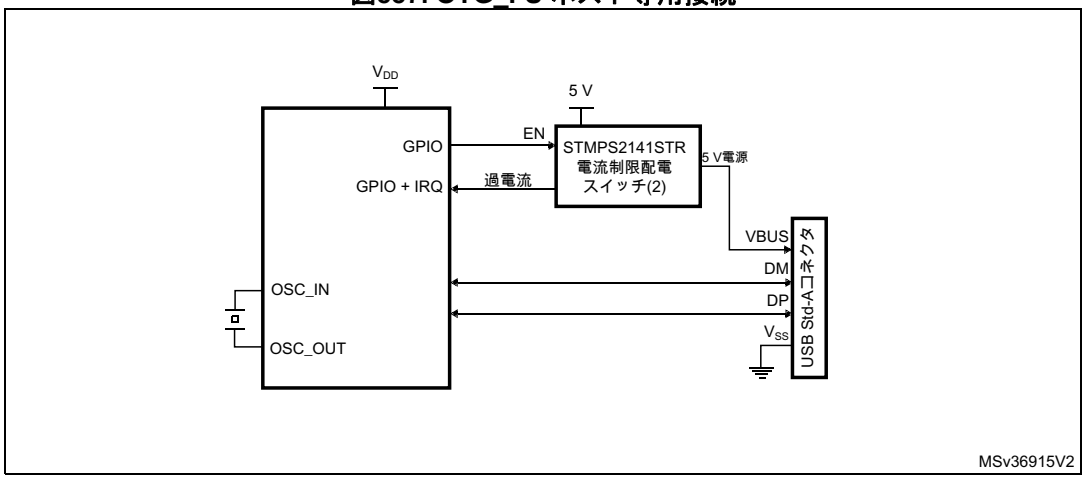
72.7 USBホストとしての OTG_FS

このセクションでは、USBホストモードでの OTG_FS の機能について説明します。OTG_FS は以下の環境でUSBホストとして機能します。

- OTG Aホスト
  - USBケーブルのA側が挿入されたときのOTG Aデバイスのデフォルト状態
- OTG Bホスト
  - HNPがホストの役割に切り替わった後のOTG Bデバイス
- Aデバイス
  - IDラインが存在し、機能していて、USBケーブルのA側に接続され、かつ、グローバルUSB設定レジスタのHNP対応ビット（OTG_GUSBCFGのHNPCAPビット）がクリアされている場合。組み込みのプルダウン抵抗が自動的にDP/DMライン上にセットされます。
- ホストのみ
  - OTG USB設定レジスタ（OTG_GUSBCFG） の強制ホストモードビット（FHMOD）は、OTG_FS コアをUSBホスト専用として機能するよう強制します。この場合、IDラインは、USBコネクタ上に存在していても無視されます。組み込みのプルダウン抵抗が自動的にDP/DMライン上にセットされます。

注： **オンチップの5V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または5 Vがアプリケーションボード上で使用可能な場合は基本的なパワースイッチを外部で追加して、5V V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意のGPIO出力からでも駆動できます。これは、OTG Aホスト、Aデバイス、およびホスト専用設定の場合に必要です。**

図887. OTG_FS ホスト専用接続



1. V_{DD} 範囲は2.0～3.6 Vです。

72.7.1 SRP対応ホスト

SRPのサポートはグローバルUSB設定レジスタのSRP対応ビット（OTG_GUSBCFGのSRPCAPビット）を通じて使用可能です。SRP機能が有効なときには、ホストは、USBセッションがサスペンド状態にある間、V_{BUS} の電力供給を停止することによって、消費電力を節約できます。

SRPホストモードのプログラミングモデルの詳細は、Aデバイスセッションリクエストプロトコルのセクションに記載されています。

72.7.2 USBホストの状態

ホストポートパワー

オンチップの5V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または5 Vがアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 VのV_{BUS}ラインを駆動しなければなりません。外付けチャージポンプは、任意のGPIO出力または外部PMIC（電源管理IC）に接続されたI^2Cインタフェースからでも駆動できます。アプリケーションがV_{BUS} に電源を供給することにした場合は、ホストポート制御およびステータスレジスタのポートパワービット（OTG_HPRTのPPWRビット）もセットする必要があります。

有効なV_{BUS}

HNPまたはSRPが有効化された場合VBUSセンシングピンはV_{BUS} に接続する必要があります。V_{BUS} 入力は、USB動作中にチャージポンプによって適切なV_{BUS} レベルが供給されることを保証します。予期せぬV_{BUS} 電圧の低下（ V_{BUS} の有効閾値4.4V以下へ）が発生すると、セッション終了検出ビット（OTG_GOTGINTのSEDETビット）によってOTG割込みがトリガされます。その場合、アプリケーションはV_{BUS} 電源供給を停止し、ポートパワービットをクリアする必要があります。

HNPとSRPが共に無効化された場合、VBUSセンシングピンはV_{BUS} に接続する必要はありません。

チャージポンプの過電流フラグは、電気的ダメージを防止するためにも使用できます。チャージポンプからの過電流フラグ出力を任意のGPIO入力に接続して、有効レベルになったときにポート割込みを生成するように設定します。過電流ISRは迅速にV_{BUS} の発生を無効にし、ポートパワービットをクリアする必要があります。

ペリフェラル接続のホスト検出

SRPまたはHNPを有効化した場合、USBペリフェラルまたはBデバイスを何時でも接続できる場合でも、V_{BUS} が有効レベル（5V）で検出できなくなるまで、OTG_FS はバス接続を検出しません。V_{BUS} が有効なレベルにあり、リモートBデバイスが接続されると、OTG_FS コアは、ホストポート制御およびステータスレジスタのデバイス接続ビット（OTG_HPRTのPCDETビット）によってトリガされるホストポート割込みを発行します。

HNPとSRPが共に切り離されたとき、USBペリフェラルまたはBデバイスが接続されると直ちに検出されます。OTG_FS コアは、ホストポート制御およびステータス内のデバイス接続ビット（OTG_HPRTのPCDETビット）から発生するホストポート割込みを発行します。

ペリフェラル切断のホスト検出

ペリフェラル切断イベントは、切断検出割込み（OTG_GINTSTSのDISCINTビット）をトリガします。

ホストのエニュメレーション

ペリフェラル接続を検出した後、ホストは、USBリセットおよび設定コマンドを新しいペリフェラルに送信することによって、エニュメレーションプロセスを開始しなければなりません。

USBリセットの駆動を開始する前に、アプリケーションは、デバウンス終了ビット（OTG_GOTGINTのDBCDNEビット）によってトリガされるOTG割込みを待ちます。これは、DP（FS）またはDM（LS）上のプルアップ抵抗の接続によって生じた電気的デバウンスの後、バスが再び安定状態に戻ったことを示します。

アプリケーションは、ホストポート制御およびステータスレジスタのポートリセットビット（OTG_HPRTのPRSTビット）を最短10ms、最長20msの間セット状態に保つことによって、USBを介したUSBリセットシグナリング（シングルエンド、ゼロ）を駆動します。アプリケーションは、タイミングカウントをチェックした後、ポートリセットビットがクリアされているかどうかを確認します。

USBリセットシーケンスが完了すると、ポートイネーブル／ディセーブル変更ビット（OTG_HPRTのPENCHNGビット）によってホストポート割込みがトリガされます。これにより、エニュメレーションされたペリフェラルの速度をホストポート制御およびステータスレジスタのポート速度フィールド（OTG_HPRTのPSPDビット）から読み出せることと、ホストがSOF（FS）またはキープアライブ（LS）の駆動を開始しようとしていることがアプリケーションに通知されます。これで、ホストは、ペリフェラル設定コマンドを送信することによってペリフェラルエニュメレーションを完了する準備ができたことになります。

ホストのサスペンド

アプリケーションは、ホストポート制御およびステータスレジスタのポートサスペンドビット（OTG_HPRTのPSUSPビット）をセットすることによって、USBの活動をサスペンド（一時停止）します。OTG_FS コアは、SOFの送信を停止し、サスペンド状態に入ります。

サスペンド状態は、リモートデバイスが主導して（リモートウェイクアップ）、終了することもできます。この場合、リモートウェイクアップシグナリング、ホストポート制御およびステータスレジスタのポートレジュームビット（OTG_HPRTのPRESビット）セルフセットが検出されると、リモートウェイクアップ割込み（OTG_GINTSTSのWKUPINTビット）が生成され、レジュームシグナリングがUSB上で自動的に駆動されます。アプリケーションは、レジュームウィンドウの時間を決めてから、ポートレジュームビットをクリアしてサスペンド状態を終了し、SOFをリスタートする必要があります。

サスペンド状態をホスト主導で終了した場合、アプリケーションは、ポートレジュームビットをセットして、ホストポートでのシグナリングのレジュームを開始し、レジュームウィンドウの時間を決め、最後に、ポートレジュームビットをクリアする必要があります。

72.7.3 ホストチャネル

OTG_FS コアは、12 つのホストチャネルをインスタンス化します。各ホストチャネルはUSBホスト転送をサポートします（USBパイプ）。ホストは、同時に 12つを超える転送リクエストをサポートすることはできません。アプリケーションからの 12 つを超える転送リクエストがペンディング中の場合、ホストコントローラドライバ（HCD）は、チャネルが以前のデューティから使用可能になったとき、すなわち、転送完了およびチャネルホールト割込みの受信後、チャネルの再割り当てを行う必要があります。

各ホストチャネルは、入力／出力および任意のタイプの周期的/非周期的トランザクションをサポートするように設定できます。各ホストチャネルは、適切な制御レジスタ（OTG_HCCHAR*x*）、転送設定レジスタ（OTG_HCTSIZ*x*）、およびステータス／割込みレジスタ（OTG_HCINT*x*）と関連するマスクレジスタ（OTG_HCINTMSK*x*）を使用します。

ホストチャネル制御

- アプリケーションは、ホストチャネル *x* 特性レジスタ（OTG_HCCHAR*x*）を通じて、以下のホストチャネル制御を使用できます。
  - チャネルの有効化／無効化
  - ターゲットUSBペリフェラルの FS/LS速度のプログラミング
  - ターゲットUSBペリフェラルのアドレスのプログラミング
  - ターゲットUSBペリフェラルのエンドポイント数のプログラミング
  - IN/OUT転送方向のプログラミング
  - USB転送タイプ（制御、バルク、割込み、アイソクロナス）のプログラミング
  - 最大パケットサイズ（MPS）のプログラミング
  - 奇数／偶数フレームで実行される周期的転送のプログラミング

ホストチャネル転送

アプリケーションは、ホストチャネル転送サイズレジスタ（OTG_HCTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。プログラミングは、ホストチャネル特性レジスタのチャネルイネーブルビットをセットする前に行う必要があります。エンドポイントが有効になると OTG_FS コアが現在の転送ステータスに従って更新するので、パケットカウントフィールドは読出し専用になります。

- 以下の転送パラメータがプログラム可能です。
  - 転送サイズ（バイト単位）
  - 全体の転送サイズを構成するパケットの数
  - 初期データPID

ホストチャネルのステータス／割込み

ホストチャネル *x* 割込みレジスタ（OTG_HCINT*x*）は、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、ホストオールチャネル割込み（OTG_HAINT）レジスタを読み出して、ホストチャネルx割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。各チャネルの各割込みソースのマスクビットもOTG_HCINTMSKxレジスタで使用可能です。

- ホストコアは、以下のステータスチェックおよび割込み生成を行います。
  - データ転送がアプリケーション側（AHB）とUSB側の両方で完了したことを示す転送完了割込み
  - 転送完了、USB トランザクションエラー、またはアプリケーションからの無効化コマンドのためにチャネルが停止したかどうか
  - 関連する送信FIFOが半分エンプティか完全エンプティか（INエンドポイント）
  - ACK応答を受け取った時
  - NAK応答を受け取った時
  - STALL応答を受け取った時
  - CRCエラー、タイムアウト、ビットスタッフエラー、偽EOPによるUSBトランザクションエラー
  - バブルエラー
  - フレームオーバーラン
  - データトグルエラー

72.7.4 ホストスケジューラ

ホストコアは、アプリケーションによってポストされたUSBトランザクションリクエストを自律的に並べ替えて、管理する組み込みハードウェアスケジューラを備えています。各フレームの開始時に、ホストは、周期的（アイソクロナスおよび割込み）トランザクションを最初に実行し、次に、非周期的（制御およびバルク）トランザクションを行うことによってUSBの仕様に従って、アイソクロナスおよび割込み転送タイプを優先します。

ホストは、リクエストキュー（1つは周期的転送のキュー、もう1つは非周期的転送のキュー）を通じてUSBトランザクションを処理します。各リクエストキューは、最大8個のエントリを保持できます。各エントリはアプリケーションからのペンディング中のトランザクションリクエストを表し、INまたはOUTチャネル番号とUSB上でトランザクションを実行するためのその他の情報を保持しています。リクエストがキューに書き込まれた順序によってUSBインタフェース上のトランザクションの順序が決まります。

各フレームの先頭で、ホストは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。現在のフレームに対してスケジュールされたアイソクロナスまたは割込みトランザクションが現在のフレームの終了時点でまだペンディング中である場合、ホストは、不完全周期転送割込み（OTG_GINTSTSのIPXFRビット）を発行します。OTG_FS コアは周期的と非周期的リクエストキューの管理を行います。周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）と非周期的送信FIFOおよびキューステータスレジスタ（OTG_HNPTXSTS）は読出し専用レジスタであり、アプリケーションから各リクエストキューのステータスを読み出すときに使うことができます。これらのレジスタには次が含まれます。

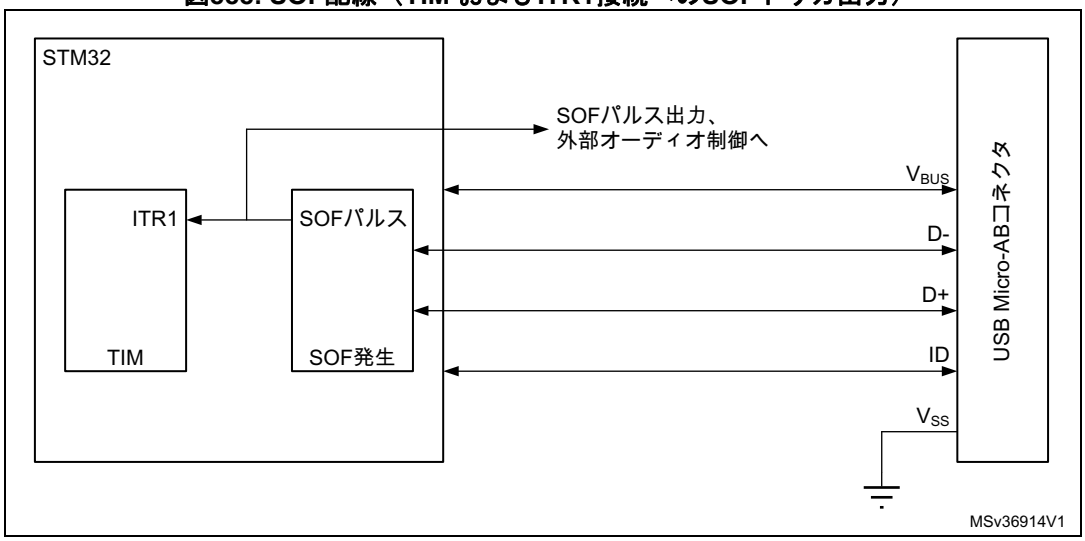
- 周期（非周期）リクエストキューで現在使用可能なフリーエントリの数（最大8エントリ）。
- 周期（非周期）Tx FIFO（OUTトランザクション）で現在使用可能な空きスペース
- IN/OUTトークン、ホストチャネル番号、およびその他のステータス情報

リクエストキューはそれぞれ最大8つのエントリを保持できるので、アプリケーションは、最大8つの保留中の周期トランザクションに加えて8つの保留中の非周期トランザクションについてSBに物理的に到達する前に、ホストトランザクションを事前にスケジュールできます。

ホストスケジューラ（キュー）へトランザクションリクエストをポストするときは、アプリケーションはOTG_HNPTXSTSレジスタのPTXQSAVビット、またはOTG_HNPTXSTSレジスタのNPTQXSAVビットを読み出して、周期的（非周期的）リクエストキュー内に少なくとも1つのエントリが存在することを確認する必要があります。

72.8 OTG_FS SOFトリガ

図888. SOF配線（TIM およびITR1接続へのSOFトリガ出力）



OTG_FS コアは、ホストとペリフェラルのSOFフレームを監視、追跡、設定する手段と、SOFパルス出力配線機能を提供します。

これらのユーティリティは、オーディオペリフェラルがPCによって提供されるアイソクロナスストリームに同期する必要がある場合や、ホストがオーディオペリフェラルの要件に従ってフレーム速度を微調整する必要がある場合など、適応型オーディオクロック生成技術に特に役立ちます。

72.8.1 ホストのSOF

ホストモードでは、2つの連続したSOF（FS）またはキープアライブ（LS）トークンの生成の間に発生するPHYクロックの数は、ホストフレームインターバルレジスタ（HFIR）でプログラム可能であり、SOFフレーム周期をアプリケーションが制御できます。割込みは、どのフレームの開始時点でも生成されます（OTG_GINTSTSのSOFビット）。現在のフレーム番号と次のSOFまでの残り時間が、ホストフレーム番号レジスタ（HFNUM）で追跡されます。

SOFパルス信号は、20 HCLKサイクルの幅の任意のSOF開始トークンで生成されます。SOFパルスは、タイマの入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマをSOFパルスによってトリガできます。

72.8.2 ペリフェラルのSOF

デバイスモードでは、フレーム割込みの開始は、SOFトークンがUSB上で受信されるたびに生成されます（OTG_GINTSTSのSOFビット）。対応するフレーム番号は、デバイスステータスレジスタ（OTG_DSTSのFNSOFビット）から読み出すことができます。20 HCLKサイクルの幅を持つSOFパルス信号も生成されます。SOFパルス信号は、TIM入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマをSOFパルスによってトリガできます。

周期的フレーム割込みの終わり（OTG_GINTSTSのEOPF）は、デバイス設定レジスタの周期的フレームインターバルフィールド（OTG_DCFGのPFIVLビット）に応じて、タイムフレームインターバルの80 %、85 %、90 % または95 % が経過した時点をアプリケーションに通知するために使用されます。この機能を使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

72.9 OTG_FS 低電力モード

下記の表743 に、STM32の低消費電力モードとOTGとの両立性を定義します。

表743.STM32の低消費電力モードのOTGとの両立性

モード	説明	USBとの両立性
RUN	MCUフルアクティブ	USBがサスペンド状態ではないときに必要です。
SLEEP	USBのサスペンド終了によって、デバイスはSLEEPモードから復帰します。ペリフェラルレジスタの内容は保たれます。	USBがサスペンド状態のときに使用可能です。
STOP	USBのサスペンド終了によって、デバイスはSTOPモードから復帰します。ペリフェラルレジスタの内容は保たれます(1)。	USBがサスペンド状態のときに使用可能です。
STANDBY	パワーダウン状態。ペリフェラルは、STANDBYモードを終了した後に再初期化する必要がある。	USBのアプリケーションとは両立しません。

1. STOP モード内では、さまざまな設定の可能性があります。いくつかの制約もあります。OTG を使用中にどのような制約があるか（ある場合）については、セクション 10：電源制御（PWR）を参照してください。

次のビットと手順によって電力消費が低減されます。

OTG PHYの消費電力は、サポートされるOTGリビジョンに応じて、一般コア設定レジスタの2つまたは3つのビットによって制御されます。

- PHYパワーダウン（OTG_GCCFGのPWRDWN）

  PHYのフルスピードトランシーバモジュールのオン／オフを切り替えます。事前にセットして、すべてのUSB動作を可能にする必要があります。

- V_{BUS} 検出イネーブル（OTG_GCCFGのVBDEN）

  OTGの動作に関連するV_{BUS} レベル検出用コンパレータのオン／オフを切り替えます。

USBセッションがまだ有効でないか、デバイスが切断されているときには、USBサスペンド状態で節電テクニックを使用できます。

- PHYクロックを停止します（OTG_PCGCCTLのSTPPCLKビット）。

  クロックゲーティング制御レジスタのストップPHYクロックビットをセットすると、OTGコア内部の48 MHzクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。USBクロック切り替え機能による動的な電力消費は、48 MHzクロック入力がアプリケーションによって動作状態を維持された場合でも節減されます。

  ほとんどのトランシーバも無効になり、非同期レジュームやリモートウェイクアップイベントの検出を担当する部分だけが動作状態を維持します。

- ゲートHCLK（OTG_PCGCCTLのGATEHCLKビット）

  クロックゲーティング制御レジスタのゲートHCLKビットをセットすると、OTG_FS コア内部のシステムクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。レジスタ読出しおよび書込みインタフェースのみが動作状態を維持します。USBクロック切り替え機能による動的な電力消費は、システムクロックが他の目的のためにアプリケーションによって動作状態を維持されている場合でも節減されます。

- USBシステムの停止

  OTG_FS がUSBサスペンド状態にある場合、アプリケーションはシステム内のすべてのクロックソースを完全にシャットダウンすることによって、電力消費全体を大幅に削減する方法を取ることができます。USBシステムの停止は、まず、ストップPHYクロックビットをセットし、次に、パワー制御システムモジュール（PWR）でシステムディープスリープモードを設定することによって有効になります。

OTG_FS コアは、USB上でのリモートウェイクアップ（ホストとして）またはリモートレジューム（デバイスとして）シグナリングの非同期検出によって、システムとUSBクロックの両方を自動的に再起動します。

動的消費電力を節減するために、USBデータFIFOは、OTG_FS コアによってアクセスされたときだけクロック駆動されます。

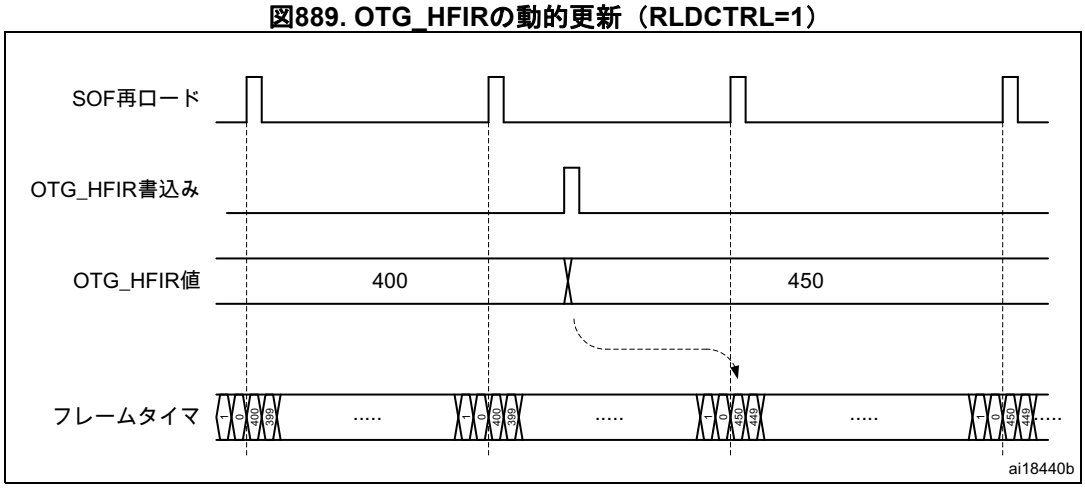
72.10 OTG_FS OTG_HFIRレジスタの動的更新

USBコアはホストモードで SOF フレーム周期を動的に調整する機能を内蔵しています。この機能を使うと、外部デバイスを SOF フレームと同期させることができます。

現在の SOF フレーム内でOTG_HFIRレジスタを変更すると、図889で説明するように次のフレームでSOF周期補正が適用されます。

動的更新では、RLDCTRL=1にセットする必要があります。

図889. OTG_HFIRの動的更新（RLDCTRL=1）

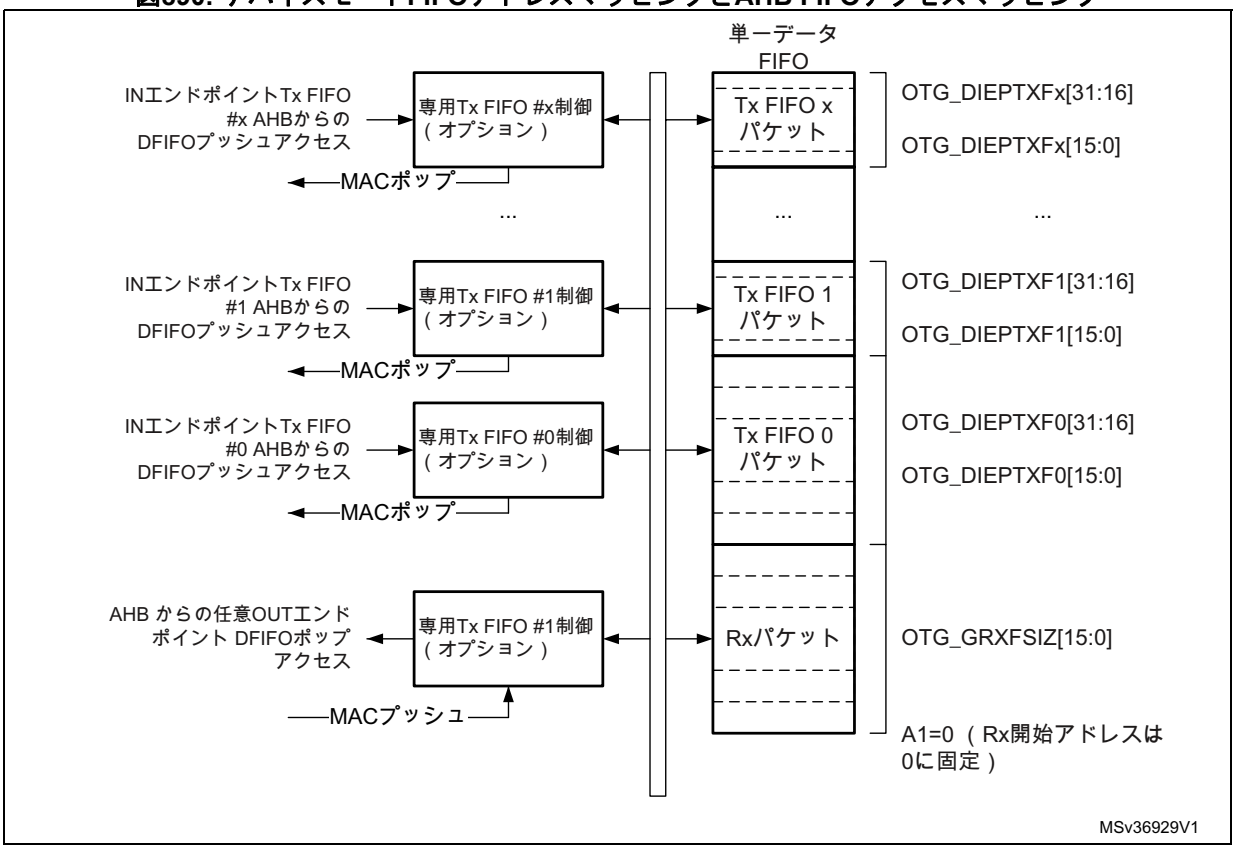


72.11 OTG_FS データFIFO

USBシステムは、高度なFIFO制御メカニズムを持つ 1.25 KBの専用RAMを備えています。OTG_FSコアのパケットFIFOコントローラモジュールは、RAM空間を、アプリケーションがUSB転送の前に一時的にデータを格納する（プッシュする）Tx FIFOと、USBから受信したデータをアプリケーションが取得する（ポップする）前に一時的に格納されるRx FIFOに編成します。割り当てられるFIFOの数と、これらがRAMの内部でどのように構成されるかは、デバイスの役割に依存します。ペリフェラルモードでは、追加のTx FIFOがアクティブなINエンドポイントのそれぞれに対して割り当てられます。FIFOのサイズは、アプリケーションの要件に合うように、ソフトウェアで設定できます。

72.11.1 ペリフェラルFIFOアーキテクチャ

図890. デバイスモードFIFOアドレスマッピングとAHB FIFOアクセスマッピング



ペリフェラルRx FIFO

OTGペリフェラルは、すべてのOUTエンドポイントに向けたデータを受信する単一の受信FIFOを使用します。受信パケットは、Rx FIFOの空きスペースが使用可能になるまで、連続的にスタックされます。受信パケットのステータス（OUTエンドポイントの宛先番号、バイト数、データPID、および受信データの有効性を含みます）も、コアによってデータペイロードの最上部に格納されます。使用可能なスペースがなくなると、ホストトランザクションはNACKされ、アドレス指定されたエンドポイントに割込みが受信されます。受信FIFOのサイズは、受信FIFOサイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信FIFOアーキテクチャは、USBペリフェラルによる受信RAMバッファへの書込みをより効率的にします。

- すべてのOUTエンドポイントは同じRAMバッファを共有します（共有FIFO）。
- OTG_FS コアは、OUTトークンのいかなるホストシーケンスについても、限度まで受信FIFOに書き込むことができます。

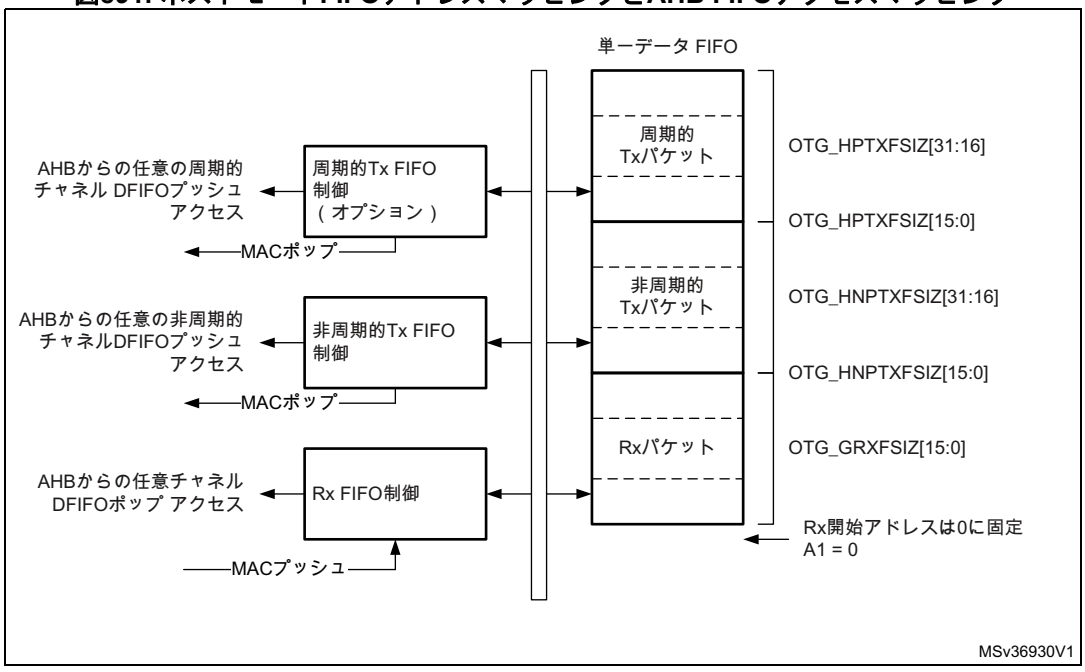
アプリケーションはRx FIFO非エンプティ割込み（OTG_GINTSTSのRXFLVLビット）の受信を、少なくとも1パケットがダウンロード可能である限り継続します。アプリケーションは、受信ステータス読出しおよびポップレジスタ（OTG_GRXSTSP）からパケット情報を読み出し、最後に、エンドポイントに関するポップアドレスから読み取ることによって受信FIFOからデータを取得します。

ペリフェラルTx FIFO

コアは、各INエンドポイント用の専用FIFOを備えています。アプリケーションは、INエンドポイント0についてはエンドポイント0送信FIFOサイズレジスタ（OTG_DIEPTXF0）に書き込み、INエンドポイントxについてはデバイスINエンドポイント送信FIFOxレジスタ（OTG_DIEPTXFx）に書き込むことによって、FIFOのサイズを設定します。

72.11.2 ホストFIFOアーキテクチャ

図891. ホストモードFIFOアドレスマッピングとAHB FIFOアクセスマッピング



ホストRx FIFO

ホストは、すべての周期的および非周期的トランザクションについて1つの受信FIFOを使用します。このFIFOは、USBから受信したデータ（受信パケットのペイロード）をシステムメモリに転送されるまで保持する受信バッファとして使用されます。任意のリモートINエンドポイントから受信されたパケットは、空きスペースがある限り連続的にスタックされます。各受信パケット（ホストチャネルデスティネーション、バイト数、データPID、および受信データの有効性を含みます）のステータスも、FIFOに格納されます。受信FIFOのサイズは、受信FIFOサイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信FIFOアーキテクチャは、USBホストによる受信データバッファへの書込みをより効率的にします。

- INに設定されたすべてのホストチャネルは、同じRAMバッファを共有します（共有FIFO）。
- OTG_FS コアは、ホストのソフトウェアによって駆動されたINトークンのいかなるシーケンスについても、限度まで受信FIFOに書き込むことができます。

アプリケーションは、少なくとも1つのパケットがダウンロード可能である限り、Rx FIFO非エンプティ割込みを受け取ります。アプリケーションは、受信ステータス読出しおよびポップレジスタからパケット情報を読み出し、最後に、受信FIFOからデータを吸い上げます。

ホストTx FIFO

ホストは、すべての非周期的（制御およびバルク）OUTトランザクションついて1つの送信FIFOを、すべての周期的（アイソクロナスおよび割込み）OUTトランザクションについて1つの送信FIFOを使用します。FIFOは、USB経由で送信されるデータ（送信パケットのペイロード）を保持する送信バッファとして使用されます。周期的（非周期的）Tx FIFOのサイズは、ホストの周期的（非周期的）送信FIFOサイズレジスタ（OTG_HPTXFSIZ/OTG_HNPTXFSIZ）で設定されます。

Tx FIFOが2つ実装されているのは、USBフレーム上の周期的型トラフィックに対して高い優先度が与えられていることによります。各フレームの先頭で、組み込みのホストスケジューラは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。

2つの送信FIFOアーキテクチャにより、USBホストは周期的送信データバッファと非周期的送信データバッファを個別に最適化できます。

- OUT 方向の周期的（非周期的）トランザクションをサポートするように設定されたすべてのホストチャネルは、同じRAMバッファを共有します（共有FIFO）。
- OTG_FS コアは、ホストのソフトウェアによって駆動されたOUTトークンのいかなるシーケンスについても、限度まで周期的（非周期的）送信FIFOに書き込むことができます。

OTG_FS コアは、AHB設定レジスタの周期的Tx FIFOエンプティレベルビット（OTG_GAHBCFGのPTXFELVLビット）の値に応じて、周期的Tx FIFOが半分または完全にエンプティである限り、周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのPTXFEビット）を発行します。アプリケーションは、周期的Tx FIFOと周期的リクエストキューの両方に空きスペースが存在する限り、前もって送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの周期的送信FIFOおよびキューステータスレジスタ（OTG_HPTXSTS）を読み出すことで知ることができます。

OTG_FS コアは、AHB設定レジスタの非周期的Tx FIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）の値に応じて、非周期的Tx FIFOが半分または完全にエンプティである限り、非周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのNPTXFEビット）を発行します。アプリケーションは、非周期的Tx FIFOと非周期的リクエストキューの両方に空きスペースが存在する限り、送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの非周期的送信FIFOおよびキューステータスレジスタ（OTG_HNPTXSTS）を読み出すことで知ることができます。

72.11.3 FIFO RAMの割り当て

デバイスモード

受信FIFO RAM割当て: アプリケーションはSETUPパケット用にRAMを割り当てる必要があります。

- 制御エンドポイントでSETUPパケットを受信するため受信FIFOに10ロケーションを確保する必要があります。コアは、SETUPパケット用に予約されている位置に他のデータを書き込むことはありません。
- 1つの位置は、グローバルOUT NAKのために割り当てられます。
- 各受信パケットとともに、ステータス情報がFIFOに書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1のスペースを割り当てる必要があります。複数のアイソクロナスエンドポイントが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ /4）+ 1の2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の2倍のスペースが推奨されます。前のパケットがCPUに転送されている間に、USBは次のパケットを受信できるからです。
- 各エンドポイントの最後のパケットとともに、送信完了ステータス情報がFIFOにプッシュされます。各OUTエンドポイントに対して1つのロケーションを予約することが推奨されます。

デバイスRxFIFO =

（5 * 制御エンドポイント数 + 8）+ （（最大USB使用パケット / 4）+ ステータス情報につき1）+ （2 * OUTエンドポイント数）+ グローバルNAKにつき1

例：MPSは周期的USBパケットでは1,024バイト、非周期的USBパケットでは512バイトです。3つのOUTエンドポイント、3つのINエンドポイント、1つの制御エンドポイント、3つのホストチャネルがあります。

デバイスRxFIFO = （5 * 1 + 8）+ （（1,024 / 4）+1）+ （2 * 4）+ 1 = 279

送信FIFO RAMの割り当て：各INエンドポイント送信FIFOに必要な最小RAMスペースは、その特定のINエンドポイントに対する最大パケットサイズを収容できるスペースとなります。

注： **送信INエンドポイントFIFOに割り当てるスペースが大きいほど、USBのパフォーマンスは高くなります。**

ホストモード

受信FIFO RAM割り当て：

各受信パケットとともに、ステータス情報がFIFOに書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1のスペースを割り当てる必要があります。複数のアイソクロナスチャネルが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1の2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の2倍のスペースが推奨されます。前のパケットがCPUに転送されている間に、USBは次のパケットを受信できるからです。

ホストチャネルの最後のパケットとともに、転送完了ステータス情報がFIFOにプッシュされます。このために1つのロケーションを割り当てる必要があります。

ホストRxFIFO = （最大USB使用パケット / 4）+ ステータス情報につき1 + 1転送完了

例：ホストRxFIFO = （（1,024 / 4）+ 1）+ 1 = 258

送信FIFO RAMの割り当て：

ホストの非周期的送信FIFOに必要な最小RAMスペースは、サポートされるすべての非周期的OUTチャネルの中で最大のパケットサイズを収容できるスペースとなります。

一般に、最大パケットサイズの2倍のスペースが推奨されます。現在のパケットがUSBに転送されているときに、CPUが次のパケットを取得できるからです。

非周期的TxFIFO = 最大非周期USB使用パケット / 4

例：非周期的TxFIFO = （512 / 4）= 128

ホストの周期的送信FIFOに必要な最小RAMスペースは、サポートされるすべての周期的OUTチャネルの中で最大のパケットサイズを収容できるスペースとなります。少なくとも1つのアイソクロナスOUTエンドポイントがある場合には、スペースはそのチャネルの最大パケットサイズの少なくとも2倍でなければなりません。

ホスト周期的TxFIFO = 最大周期USB使用パケット / 4

例：ホスト周期的TxFIFO = （1,024 / 4）= 256

注： **非周期的送信FIFOに割り当てるスペースが大きいほど、USBのパフォーマンスは高くなります。**

72.12 OTG_FSシステムのパフォーマンス

大きなRAMバッファ、柔軟に設定可能なFIFOサイズ、AHBプッシュ／ポップレジスタを介した迅速な32bit FIFOアクセス、そして、特に、高度なFIFO制御メカニズムにより、最高のUSBおよびシステムパフォーマンスが実現されます。このメカニズムにより、OTG_FSは、現在のUSBシーケンスに関係なく、使用可能なRAMスペースに効率的に書き込むことができます。これらの機能によって、以下のことが可能になります。

- アプリケーションは、CPU の帯域幅使用率を最適化するための介入を調整する十分な余裕ができます。
  - データを効率的にUSBに送信できるときまで、大量の送信データを事前に蓄積することができます。
  - 単一の受信 FIFO からデータをダウンロードするための時間の余裕が増えるという利点があります。
- USBコアは、最高動作速度を維持できるので、アプリケーションの介入がある場合に比べて、自律動作できる幅が広がり、最大のフルスピード帯域幅を実現できます。
  - 大量の送信データを自由に蓄積して、USB経由のデータ送信を自律管理できます。
  - 受信バッファの使用可能な空きスペースが大量にあるので、USBから入ってくるデータを自律的に書き込むことができます。

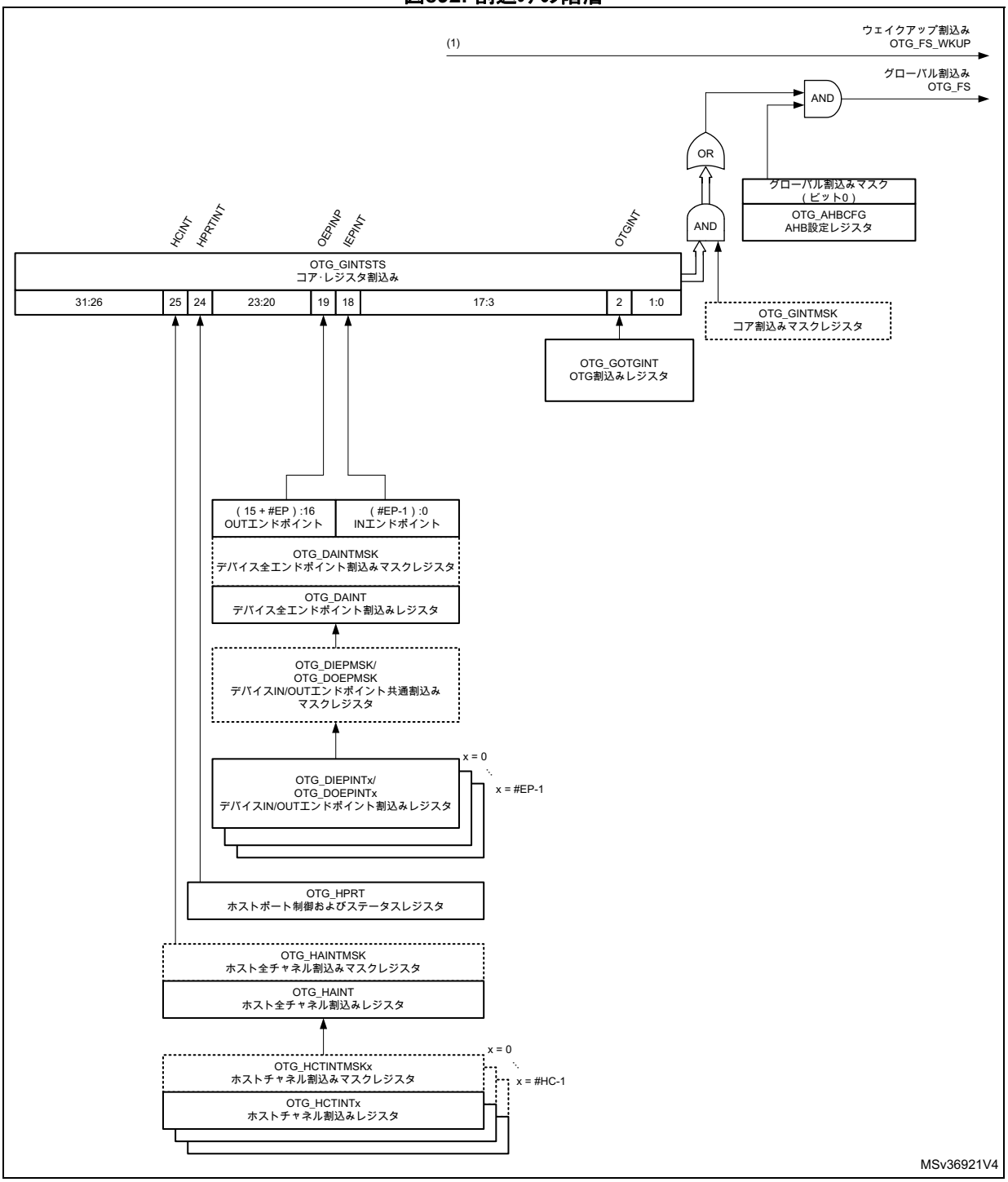
OTG_FSコアは1.25KBのRAMバッファを非常に効率的に使用でき、また、1.25KBの送信／受信データはフルスピードフレームをカバーするには十分以上なので、USBシステムはCPUの介入がなくても、最大1USBフレーム（1ms）に対応する最高のフルスピードデータ速度に耐えることができます。

72.13 OTG_FS 割込み

OTG_FS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTSレジスタのMMISビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

図892に、割込みの階層を示します。

図892. 割込みの階層



1. L1 SLEEPまたはL2 SUSPEND状態中にレジューム条件が発生すると、OTG_FS_WKUP がアクティブ（ハイ状態）になります。

72.14 OTG_FS 制御／ステータスレジスタ

アプリケーションは、AHBスレーブインタフェースを通じて制御／ステータスレジスタ（CSR）を読み書きすることによって、OTG_FS コントローラを制御します。これらのレジスタは32bit幅であり、アドレスは32bitブロックに配置されています。OTG_FS レジスタは、ワード（32bit）単位でアクセスする必要があります。

CSRは、次のように分類されます。

- コアグローバルレジスタ
- ホストモードレジスタ
- ホストグローバルレジスタ
- ホストポートCSR
- ホストチャネル固有レジスタ
- デバイスモードレジスタ
- デバイスグローバルレジスタ
- デバイスエンドポイント固有レジスタ
- パワーおよびクロックゲーティングレジスタ
- データFIFO（DFIFO）アクセスレジスタ

コアグローバル、パワーおよびクロックゲーティング、データFIFO（DFIFO）アクセス、およびホストポートCSRレジスタのみが、ホストモードおよびデバイスモードの両方でのアクセスが可能です。OTG_FS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTSレジスタのMMISビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

72.14.1 CSRメモリマップ

ホストモードおよびデバイスモードレジスタは、それぞれ異なるアドレスを占めます。すべてのレジスタはAHBクロックドメインで実装されます。

グローバルCSRマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能です。

表744.コアグローバル制御およびステータスレジスタ（CSR）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GOTGCTL	0x000	セクション 72.15.1：OTG制御およびステータスレジスタ（OTG_GOTGCTL）
OTG_GOTGINT	0x004	セクション 72.15.2：OTG割込みレジスタ（OTG_GOTGINT）
OTG_GAHBCFG	0x008	セクション 72.15.3：OTG AHB設定レジスタ（OTG_GAHBCFG）
OTG_GUSBCFG	0x00C	セクション 72.15.4：OTG USB設定レジスタ（OTG_GUSBCFG）
OTG_GRSTCTL	0x010	セクション 72.15.5：OTGリセットレジスタ（OTG_GRSTCTL）
OTG_GINTSTS	0x014	セクション 72.15.6：OTGコア割込みレジスタ（OTG_GINTSTS）
OTG_GINTMSK	0x018	セクション 72.15.7：OTG割込みマスクレジスタ（OTG_GINTMSK）

表744.コアグローバル制御およびステータスレジスタ（CSR）（続き）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GRXSTSR	0x01C	セクション 72.15.8：OTG受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR）
		セクション 72.15.9：OTG受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR）
OTG_GRXSTSP	0x020	セクション 72.15.10：OTGステータス読出しおよびポップレジスタ（OTG_GRXSTSP）
		セクション 72.15.11：OTGステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）
OTG_GRXFSIZ	0x024	セクション 72.15.12：OTG受信FIFOサイズレジスタ（OTG_GRXFSIZ）
OTG_HNPTXFSIZ/OTG_DIEPTXF0(1)	0x028	セクション 72.15.13：OTGホスト非周期的送信FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント0送信FIFOサイズ（OTG_DIEPTXF0）
OTG_HNPTXSTS	0x02C	セクション 72.15.14：OTG非周期的送信FIFO／キューステータスレジスタ（OTG_HNPTXSTS）
OTG_GCCFG	0x038	セクション 72.15.15：OTG一般コア設定レジスタ（OTG_GCCFG）
OTG_CID	0x03C	セクション 72.15.16：OTGコアIDレジスタ（OTG_CID）
OTG_GLPMCFG	0x54	セクション 72.15.17：OTGコアLPM設定レジスタ（OTG_GLPMCFG）
OTG_HPTXFSIZ	0x100	セクション 72.15.18：OTGホスト周期的送信FIFOサイズレジスタ（OTG_HPTXFSIZ）
OTG_DIEPTXFx	0x104 0x108 ..... 0x114	セクション 72.15.19：OTGデバイスINエンドポイント送信FIFO xサイズレジスタ（OTG_DIEPTXFx）

1. 一般的ルールとして、OTG_HNPTXFSIZはホストモードで、OTG_DIEPTXF0はデバイスモードで使用されます。

ホストモードCSRマップ

これらのレジスタは、コアがホストモードに変わるたびにプログラムされなければなりません。

表745.ホストモード制御およびステータスレジスタ （CSR）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCFG	0x400	セクション 72.15.21：OTGホスト設定レジスタ（OTG_HCFG）
OTG_HFIR	0x404	セクション 72.15.22：OTGホストフレームインターバルレジスタ（OTG_HFIR）
OTG_HFNUM	0x408	セクション 72.15.23：OTGホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）
OTG_HPTXSTS	0x410	セクション 72.15.24：OTGホスト周期的送信FIFO ／キューステータスレジスタ（OTG_HPTXSTS）
OTG_HAINT	0x414	セクション 72.15.25：OTGホスト全チャネル割込みレジスタ（OTG_HAINT）

表745.ホストモード制御およびステータスレジスタ （CSR）（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HAINTMSK	0x418	セクション 72.15.26 : OTGホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）
OTG_HPRT	0x440	セクション 72.15.27 : OTGホストポート制御およびステータスレジスタ（OTG_HPRT）
OTG_HCCHARx	0x500 0x520 ..... 0x660	セクション 72.15.28 : OTGホストチャネルx特性レジスタ（OTG_HCCHARx）
OTG_HCINTx	0x508 0x528 ... 0x668	セクション 72.15.29 : OTGホストチャネルx割込みレジスタ（OTG_HCINTx）
OTG_HCINTMSKx	0x50C 0x52C ... 0x66C	セクション 72.15.30 : OTGホストチャネルx割込みマスクレジスタ（OTG_HCINTMSKx）
OTG_HCTSIZx	0x510 0x530 ... 0x670	セクション 72.15.31 : OTGホストチャネルx転送サイズレジスタ（OTG_HCTSIZx）

デバイスモードCSRマップ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

表746.デバイスモード制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_DCFG	0x800	セクション 72.15.33 : OTGデバイス設定レジスタ（OTG_DCFG）
OTG_DCTL	0x804	セクション 72.15.34 : OTGデバイス制御レジスタ（OTG_DCTL）
OTG_DSTS	0x808	セクション 72.15.35 : OTGデバイスステータスレジスタ（OTG_DSTS）
OTG_DIEPMSK	0x810	セクション 72.15.36 : OTGデバイスINエンドポイント共通割込みマスクレジスタ （OTG_DIEPMSK）
OTG_DOEPMSK	0x814	セクション 72.15.37 : OTGデバイスOUTエンドポイント共通割込みマスクレジスタ （OTG_DOEPMSK）
OTG_DAINT	0x818	セクション 72.15.38 : OTGデバイス全エンドポイント割込みレジスタ（OTG_DAINT）
OTG_DAINTMSK	0x81C	セクション 72.15.39 : OTG全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）
OTG_DVBUSDIS	0x828	セクション 72.15.40 : OTGデバイスV_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）

表746.デバイスモード制御およびステータスレジスタ （続き）

項目（略称）	オフセット アドレス	レジスタ名
OTG_DVBUSPULSE	0x82C	セクション 72.15.41 : OTGデバイスV_{BUS} パルス時間レジスタ （OTG_DVBUSPULSE）
OTG_DIEPEMPMSK	0x834	セクション 72.15.42 : OTGデバイスINエンドポイントFIFOエンプティ割込みマスクレジスタ （OTG_DIEPEMPMSK）
OTG_DIEPCTL0	0x900	セクション 72.15.43 : OTGデバイス制御INエンドポイント0制御レジスタ （OTG_DIEPCTL0）
OTG_DIEPCTLx	0x920 0x940 ..... 0x9A0	セクション 72.15.44 : OTGデバイスINエンドポイントx制御レジスタ （OTG_DIEPCTLx）
OTG_DIEPINTx	0x908 0x928 ... 0x988	セクション 72.15.45 : OTGデバイスINエンドポイントx割込みレジスタ （OTG_DIEPINTx）
OTG_DIEPTSIZ0	0x910	セクション 72.15.46 : OTGデバイスINエンドポイント0転送サイズレジスタ （OTG_DIEPTSIZ0）
OTG_DTXFSTSx	0x918 0x938 ... 0x998	セクション 72.15.47 : OTGデバイスINエンドポイント送信FIFOステータスレジスタ （OTG_DTXFSTSx）
OTG_DIEPTSIZx	0x930 0x950 ..... 0x9B0	セクション 72.15.48 : OTGデバイスINエンドポイントx転送サイズレジスタ （OTG_DIEPTSIZx）
OTG_DOEPCTL0	0xB00	セクション 72.15.49 : OTGデバイス制御OUTエンドポイント0制御レジスタ （OTG_DOEPCTL0）
OTG_DOEPINTx	0xB08 0xB28 ..... 0xBA8	セクション 72.15.50 : OTGデバイスOUTエンドポイントx割込みレジスタ （OTG_DOEPINTx）
OTG_DOEPTSIZ0	0xB10	セクション 72.15.51 : OTGデバイスOUTエンドポイント0転送サイズレジスタ （OTG_DOEPTSIZ0）
OTG_DOEPCTLx	0xB20 0xB40 ..... 0xBA0	セクション 72.15.52 : OTGデバイスOUTエンドポイントx制御レジスタ （OTG_DOEPCTLx）
OTG_DOEPTSIZx	0xB30 0xB50 ..... 0xBB0	セクション 72.15.53 : OTGデバイスOUTエンドポイントx転送サイズレジスタ （OTG_DOEPTSIZx）

データFIFO（DFIFO）アクセスレジスタマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能であり、特定のエンドポイントまたはチャネルのFIFOスペースを指定された方向で読み書きするために使用されます。ホストチャネルがINタイプの場合FIFOはそのチャネル上では読出しのみが可能です。同様に、ホストチャネルがOUTタイプの場合FIFOはそのチャネル上では書込みのみが可能です。

表747.データFIFO（DFIFO）アクセスレジスタマップ

FIFOアクセスレジスタセクション	オフセットアドレス	アクセス
デバイスINエンドポイント0／ホストOUTチャネル0 : DFIFO書込みアクセス デバイスOUTエンドポイント0／ホストINチャネル0 : DFIFO読出しアクセス	0x1000 – 0x1FFC	w r
デバイスINエンドポイント1／ホストOUTチャネル1 : DFIFO書込みアクセス デバイスOUTエンドポイント1／ホストINチャネル1 : DFIFO読出しアクセス	0x2000～0x2FFC	w r
.....	.....	.....
デバイスINエンドポイントx(1)／ホストOUTチャネルx (1) : DFIFO書込みアクセス デバイスOUTエンドポイントx (1) ／ホストINチャネルx (1) : DFIFO読出しアクセス	0xX000～0xXFFC	w r

1. デバイスモードではx = 5、ホストモードではx = 11 です。

パワーおよびクロックゲーティングCSRマップ

パワーおよびクロックゲーティング用のレジスタが1つあります。このレジスタは、ホストモードとデバイスモードの両方で使用できます。

表748.パワーおよびクロックゲーティング制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_PCGCCTL	0xE00～0xE04	セクション 72.15.54 : OTGパワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

72.15 OTG_FS レジスタ

これらのレジスタは、ホストモードとデバイスモードの両方で使用でき、これらのモード間で切り替わっても再プログラムの必要はありません。

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

72.15.1 OTG制御およびステータスレジスタ（OTG_GOTGCTL）

アドレスオフセット：0x000

リセット値：0x0001 0000

OTG_GOTGCTLレジスタは、コアのOTG機能の動作を制御し、そのステータスを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CUR MOD	OTG VER	BSVLD	ASVLD	DBCT	CID STS
										r	rw	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EHEN	DHNP EN	HSHNP EN	HNP RQ	HNG SCS	BVALO VAL	BVALO EN	AVALO VAL	AVALO EN	VBVAL OVAL	VBVAL OEN	SRQ	SRQ SCS
			rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw	rw	r

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **CURMOD**：現在の動作モード

現在のモードを示します（ホストまたはデバイス）。

0：デバイスモード

1：ホストモード

ビット20 **OTGVER**：OTGバージョン

OTGバージョンを選択します。

0：OTGバージョン1.3です。OTG1.3は、新しい製品開発では廃止されています。

1：OTGバージョン2.0です。このバージョンでは、コアはSRPのデータラインパルスのみをサポートしています。

ビット19 **BSVLD**：Bセッション有効

デバイスモードでのトランシーバのステータスを示します。

0：Bセッションは無効です。

1：Bセッションは有効です。

OTGモードでは、ユーザはこのビットを使用して、デバイスが接続されているか、切断されているかを判断できます。

注： デバイスモードでのみアクセス可能です。

ビット18 **ASVLD**：Aセッションの有効性

ホストモードでのトランシーバのステータスを示します。

0：Aセッションは無効です。

1：Aセッションは有効です。

注： ホストモードでのみアクセス可能です。

ビット17 **DBCT**：ロング／ショートデバウンス時間

検出された接続のデバウンス時間を示します。

0：ロングデバウンス時間（100ms + 2.5µs）。物理的接続に使用されます。

1：ショートデバウンス時間（2.5 µs）。ソフト接続に使用されます。

注： ホストモードでのみアクセス可能です。

ビット16 **CIDSTS**：コネクタIDステータス

接続イベント時のコネクタIDのステータスを示します。

0：OTG_FS コントローラはAデバイスモードです。

1：OTG_FS コントローラはBデバイスモードです。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **EHEN**：組み込みホストの有効化

OTG Aデバイスのステートマシンと組み込みホストのステートマシンから選択するために使用します。

0：OTG Aデバイスのステートマシンを選択します。

1：組み込みホストのステートマシンを選択します。

ビット11 **DHNPEN**：デバイスHNPの有効化

アプリケーションは、接続されたUSBホストからSetFeature.SetHNPEnableコマンドを正常に受信したときに、このビットをセットします。

0：HNPは、アプリケーションでは無効です。

1：HNPは、アプリケーションで有効です。

注： デバイスモードでのみアクセス可能です。

ビット10 **HSHNPEN**：ホストセットHNP有効化

アプリケーションは、接続されたデバイスでHNPを正常に有効できたとき（SetFeature.SetHNPEnableコマンドを使用して）、このビットをセットします。

0：ホストセットHNPは無効です。

1：ホストセットHNPは有効です。

注： ホストモードでのみアクセス可能です。

ビット9 **HNPRQ**：HNPリクエスト

アプリケーションは、接続されたUSBホストへのHNPリクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINTレジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINTのHNSSCHGビット）がセットされたとき、0を書き込むことによって、このビットをクリアできます。コアはHNSSCHGビットがクリアされたとき、このビットをクリアします。

0：HNPリクエストなし

1：HNPリクエスト

注： デバイスモードでのみアクセス可能です。

ビット8 **HNGSCS**：ホストネゴシエーション成功

コアは、ホストネゴシエーションに成功したとき、このビットをセットします。コアは、このレジスタのHNPリクエスト（HNPRQ）ビットがセットされたとき、このビットをクリアします。

0：ホストネゴシエーション失敗

1：ホストネゴシエーション成功

注： デバイスモードでのみアクセス可能です。

ビット7 **BVALOVAL**：Bペリフェラルセッション有効上書き値

このビットは、BVALOENビットがセットされている場合に、Bvalid信号に上書き値をセットするために使用されます。

0：BVALOEN = 1のとき、Bvalid値は「0」です。

1：BVALOEN = 1のとき、Bvalid値は「1」です。

注： デバイスモードでのみアクセス可能です。

ビット6 **BVALOEN**：Bペリフェラルセッション有効値上書きの有効化

このビットは、BVALOVALビットを使用してBvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのBvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったBvalid信号はBVALOVALビット値で上書きされます。

注： デバイスモードでのみアクセス可能です。

ビット5 **AVALOVAL**：Aペリフェラルセッション有効上書き値

このビットは、AVALOENビットがセットされている場合に、Avalid信号に上書き値をセットするために使用されます。

0：AVALOEN = 1のとき、Avalid値は「0」です。

1：AVALOEN = 1のとき、Avalid値は「1」です。

注： ホストモードでのみアクセス可能です。

ビット4 **AVALOEN**：Aペリフェラルセッション有効値上書きの有効化

このビットは、AVALOVALビットを使用してAvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのAvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったAvalid信号はAVALOVALビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット3 **VBVALOVAL**：V_{BUS} 有効上書き値

このビットは、VBVALOENビットがセットされている場合に、vbusvalid信号に上書き値をセットするために使用されます。

0：VBVALOEN = 0のとき、vbusvalid値は”1”です。

1：VBVALOEN = 1のとき、vbusvalid値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット2 **VBVALOEN**：V_{BUS} 有効値上書きの有効化

このビットは、VBVALOVALビットを使用してvbusvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのvbusvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったvbusvalid信号はVBVALOVALビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット1 **SRQ**：セッションリクエスト

アプリケーションはUSB上でセッションリクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINTレジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINTのHNSSCHGビット）がセットされたとき、0を書き込むことによって、このビットをクリアできます。コアはHNSSCHGビットがクリアされたとき、このビットをクリアします。

ユーザがUSB 1.1フルスピードシリアルトランシーバインタフェースを使用してセッションリクエストを開始する場合、アプリケーションは、このレジスタのBセッション有効ビット（OTG_GOTGCTLのBSVLDビット）がクリアされた後V_{BUS} が0.2Vへ放電するまで待つ必要があります。

0：セッションリクエストなし

1：セッションリクエスト

注： デバイスモードでのみアクセス可能です。

ビット0 **SRQSCS**：セッションリクエスト成功

コアは、セッションリクエストの開始に成功したとき、このビットをセットします。

0：セッションリクエスト失敗

1：セッションリクエスト成功

注： デバイスモードでのみアクセス可能です。

72.15.2 OTG割込みレジスタ（OTG_GOTGINT）

アドレスオフセット：0x04

リセット値：0x0000 0000

アプリケーションはOTG割込みがあると、このレジスタを読み出し、このレジスタのビットをクリアすることによってOTG割込みをクリアします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBC DNE	ADTO CHG	HNG DET	Res.
												rc_w1	rc_w1	rc_w1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HNSS CHG	SRSS CHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
						rc_w1	rc_w1						rc_w1		

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **DBCDNE**：デバウンス完了

コアは、デバイスが接続された後、デバウンスが完了したときに、このビットをセットします。アプリケーションは、この割込みを確認した後にUSBリセットの駆動を開始できます。このビットは、OTG_GUSBCFGレジスタのHNP対応ビットまたはSRP対応ビット（OTG_GUSBCFGのHNPCAPビットとSRPCAPビット）がセットされたときのみ有効です。

注： ホストモードでのみアクセス可能です。

ビット18 **ADTOCHG**：Aデバイスタイムアウト変化

コアはBデバイスの接続を待っている間にAデバイスがタイムアウトしたことを示すために、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット17 **HNGDET**：ホストネゴシエーションの検出

コアはUSB上でホストネゴシエーションリクエストを検出したときに、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット16:10 予約済みであり、リセット値に保持する必要があります。

ビット9 **HNSSCHG**：ホストネゴシエーション成功ステータス変化

コアはUSBホストネゴシエーションリクエストの成功または失敗時に、このビットをセットします。アプリケーションは、OTG_GOTGCTLレジスタのホストネゴシエーション成功ビット（OTG_GOTGCTLのHNGSCSビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット7:3 予約済みであり、リセット値に保持する必要があります。

ビット8 **SRSSCHG**：セッションリクエスト成功ステータス変化

コアは、セッションリクエストの成功または失敗のときに、このビットをセットします。アプリケーションは、OTG_GOTGCTLレジスタのセッションリクエスト成功ビット（OTG_GOTGCTLのSRQSCSビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット2 **SEDET**：セッション終了検出

コアは、このビットをセットして、V_{BUS} の電圧レベルがBペリフェラルセッションにとって有効でなくなったことを示します（V_{BUS} < 0.8 Vのとき）。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

72.15.3 OTG AHB設定レジスタ（OTG_GAHBCFG）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタを使用して、電源投入後、またはモード変化後のコアを設定できます。このレジスタは、主にAHBシステム関連の設定パラメータを含んでいます。初期プログラミングの後は、このレジスタを変更しないでください。アプリケーションはAHBまたはUSBのいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFE LVL	TXFE LVL	Res.	Res.	Res.	Res.	Res.	Res.	GINT MSK
							rw	rw							rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PTXFELVL**：周期的Tx FIFOエンプティレベル

OTG_GINTSTSレジスタの周期的Tx FIFOエンプティ割込みビット（OTG_GINTSTSのPTXFEビット）がトリガされるタイミングを示します。

0：OTG_GINTSTSのPTXFE割込みビットは、周期的TxFIFOが半分エンプティであることを示します。

1：OTG_GINTSTSのPTXFE割込みビットは、周期的Tx FIFOが完全エンプティであることを示します。

注： ホストモードでのみアクセス可能です。

ビット7 **TXFELVL**：周期的Tx FIFOエンプティレベル

デバイスモードでは、このビットは、INエンドポイント送信FIFOエンプティ割込み（OTG_DIEPINTxのTXFE）がトリガされるタイミングを示します。

0：OTG_DIEPINTxのTXFE割込みは、INエンドポイントTx FIFOが半分エンプティであることを示します。

1：OTG_DIEPINTxのTXFE割込みは、INエンドポイントTx FIFOが完全エンプティであることを示します。

ホストモードでは、このビットは、非周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのNPTXFEビット）がトリガされるタイミングを示します。

0：OTG_GINTSTSのNPTXFE割込みは、非周期的Tx FIFOが半分エンプティであることを示します。

1：OTG_GINTSTSのNPTXFE割込みは、非周期的Tx FIFOが完全エンプティであることを示します。

ビット6:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **GINTMSK**：グローバル割込みマスク

アプリケーションは、このビットを使用して、アプリケーション自身に対する割込みラインのアサーションをマスクまたはマスク解除します。このビットの設定に関係なく、割込みステータスレジスタはコアによって更新されます。

0：アプリケーションへの割込みアサーションをマスクします。

1：アプリケーションへの割込みアサーションをマスク解除します。

注： デバイスモードとホストモードの両方でアクセス可能です。

72.15.4 OTG USB設定レジスタ（OTG_GUSBCFG）

アドレスオフセット：0x00C

リセット値：0x0000 1440

このレジスタを使用して、電源投入後、またはホストモードあるいはデバイスモードへの変化後に、コアを設定することができます。このレジスタはUSBおよびUSB-PHY関連の設定パラメータを含んでいます。アプリケーションはAHBまたはUSBのいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FD MOD	FH MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TRDT[3:0]				HNP CAP	SRP CAP	Res.	PHY SEL	Res.	Res.	Res.	TOCAL[2:0]		
		rw	rw	rw	rw	rw	rw		r				rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **FDMOD**：強制デバイスモード

このビットに1を書き込むと、コアはOTG_ID入力ピンに関係なく、強制的にデバイスモードになります。

0：通常モード

1：強制デバイスモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも25 ms待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット29 **FHMOD**：強制ホストモード

このビットに1を書き込むと、コアはOTG_ID入力ピンに関係なく、強制的にホストモードになります。

0：通常モード

1：強制ホストモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも25 ms待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット28:26 予約済みであり、リセット値に保持する必要があります。

ビット25:23 予約済みであり、リセット値に保持する必要があります。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21:16 予約済みであり、リセット値に保持する必要があります。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:10 **TRDT[3:0]**：USBターンアラウンド時間

これらのビットは、ターンアラウンド時間をPHYクロックでセットするために使用されます。これらは、表 749：TRDT値によってアプリケーションAHB周波数により設定する必要があります。TRDT値が高ければ、INトークンへのUSB応答時間を伸ばして、データFIFOへのより長いAHB読出しアクセス遅延を補正できます。

注： デバイスモードでのみアクセス可能です。

ビット9 **HNPCAP**：HNPの有効化

アプリケーションは、このビットを使用して OTG_FS コントローラのHNP機能を制御します。

0：HNP機能は無効です。

1：HNP機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット8 **SRPCAP**：SRPの有効化

アプリケーションは、このビットを使用して OTG_FS コントローラのSRP機能を制御します。コアが非SRP対応Bデバイスとして動作する場合、
デバイスは、接続されたAデバイス（ホスト）にV_{BUS} を起動し、セッションを開始させるように要求することはできません。

0：SRP機能は無効です。

1：SRP機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **PHYSEL**：フルスピードシリアルトランシーバモードの選択

このビットは常に1であり、アクセスは読出し専用です。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TOCAL[2:0]**：FSタイムアウトの較正

アプリケーションがこのフィールドでプログラムするPHYクロックの数は、PHYによって導入される追加遅延を考慮して、コアのフルスピードパケット間タイムアウト時間に追加されます。ライン状態条件の生成時にPHYによって導入される遅延はPHYごとに異なる場合があるので、較正が必要なことがあります。

フルスピード動作でのUSBの標準タイムアウト値は16～18bit時間です。アプリケーションは、エニュメレーションのスピードに基づいて、このフィールドをプログラムする必要があります。PHYクロックごとに追加されるビット時間の数は0.25bit時間です。

表749.TRDT値

AHB周波数範囲（MHz）		TRDT最小値
最小値	最大値	
14.2	15	0xF
15	16	0xE
16	17.2	0xD
17.2	18.5	0xC
18.5	20	0xB
20	21.8	0xA
21.8	24	0x9
24	27.5	0x8
27.5	32	0x7
32	-	0x6

72.15.5 OTGリセットレジスタ（OTG_GRSTCTL）

アドレスオフセット：0x10

リセット値：0x8000 0000

アプリケーションは、このレジスタを使用して、コア内のさまざまなハードウェア機能をリセットします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AHB IDL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXFNUM[4:0]					TXF FLSH	RXF FLSH	Res.	FCRST	PSRST	CSRST
					rw	rw	rw	rw	rw	rs	rs		rs	rs	r

ビット31 **AHBIDL**：AHBマスタアイドル

AHBマスタステートマシンがアイドル状態にあることを示します。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット30:11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **TXFNUM[4:0]**：Tx FIFO番号

これはTx FIFOフラッシュビットを使用して一掃しなければならないFIFO番号です。コアがTx FIFOフラッシュビットをクリアするまでは、このフィールドを変更しないでください。

00000：

– ホストモードでの非周期的Tx FIFOの一掃

– デバイスモードでのTx FIFO 0の一掃

00001：

– ホストモードでの周期的Tx FIFOの一掃

– デバイスモードでのTx FIFO 1の一掃

00010：デバイスモードでのTx FIFO 2の一掃

.....

01111：デバイスモードでのTx FIFO 15の一掃

10000：デバイスモードまたはホストモードのすべての送信FIFOを一掃します。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット5 **TXFFLSH**：Tx FIFOの一掃

このビットは、単一またはすべての送信FIFOを選択的に一掃しますが、コアがトランザクション中の場合は一掃できません。

アプリケーションは、コアがTx FIFOへの書込み中かTxFIFOからの読出し中のどちらでもないことを確認した後でなければ、このビットに書き込むことはできません。以下のレジスタを使用して確認してください。

読出し — NAK有効割込みを使用すると、コアがFIFOから読出しを行っていないことを確認できます。

書込み — OTG_GRSTCTLのAHBIDLビットで、コアがFIFOに書込みを行っていないことを保証します。

通常、FIFOが再設定される場合は一掃を実施することをお勧めします。デバイスエンドポイントが無効の間も、FIFO一掃の実施をお勧めします。アプリケーションは、コアがこのビットをクリアするまで待ってから、他の操作を実行しなければなりません。phy_clkまたはhclkのより低速なクロックを使用してこのビットをクリアするには、8クロックかかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット4 **RXFFLSH**：Rx FIFOの一掃

アプリケーションは、このビットを使用してRx FIFO全体を一掃できますが、まず、コアがトランザクション中でないことを確認する必要があります。

アプリケーションは、コアがRx FIFOからの読出し中、またはRx FIFOへの書込み中のどちらでもないことを確認した後でなければ、このビットに書き込んではなりません。

アプリケーションは、他の操作を実行する前に、このビットがクリアされるまで待つ必要があります。このビットがクリアされるまで、8クロック（PHYまたはAHBクロックの中で最も低速なクロック）かかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FCRST**：ホストフレームカウンタのリセット

アプリケーションは、このビットに書き込むことによって、コア内のフレーム数カウンタをリセットします。フレームカウンタがリセットされると、コアによって次に送出されるSOFのフレーム番号は0になります。

アプリケーションでこのビットに1が書き込まれた場合、このビットは数クロックサイクル内にコアによってクリアさるため、その値を読み戻すことはできない可能性があります。

注： ホストモードでのみアクセス可能です。

ビット1 **PSRST**：部分的なソフトリセット

内部ステートマシンをリセットしますが、エニュメレーション情報は保持されます。特定のPHYエラーからの復帰に使用できます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット0 **CSRST**：コアソフトのリセット

以下のようにHCLKおよびPHYクロックドメインをリセットします。

割込みと、以下のビットを除くすべてのCSRレジスタビットをクリアします。

–OTG_PCGCCTLレジスタ のGATEHCLKビット

–OTG_PCGCCTLレジスタ のSTPPCLKビット

–OTG_HCFGレジスタ のFSLSPCSビット

–OTG_DCFGレジスタ のDSPDビット

–OTG_DCTLレジスタ のSDISビット

–OTG_GCCFGレジスタ

すべてのモジュールのステートマシン（AHBスレーブユニットを除く）がアイドル状態にリセットされ、すべての送信FIFOおよび受信FIFOがフラッシュされます。

AHB転送の最後のデータフェーズの完了後AHBマスタ上のすべてのトランザクションは速やかに終了されます。USB上のすべてのトランザクションをただちに終了します。

アプリケーションは、いつでもこのビットに書き込むことによって、コアをリセットできます。これはセルフクリアビットであり、コア内のすべての必要なロジックがリセットされた後、コアがこのビットをクリアします。コアの現在の状態によって、クリアには数クロックかかります。このビットがクリアされると、ソフトウェアはPHYドメインにアクセスできるようになるまで少なくとも3 PHYクロック待つ必要があります（同期遅延）。また、ソフトウェアは、操作を開始する前に、このレジスタのビット31が1にセットされていること（AHBマスタがアイドル状態）を確認する必要があります。

一般に、ソフトウェアリセットは、ソフトウェア開発時に使用され、ユーザが上記のUSB設定レジスタのPHY選択ビットを動的に変更するときにも使用されます。PHYを変更するときには、PHYの対応クロックが選択されPHYドメインで使用されます。新しいクロックが選択されたら、適切な動作のためにはPHYドメインをリセットする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

72.15.6 OTGコア割込みレジスタ（OTG_GINTSTS）

アドレスオフセット：0x014

リセット値：0x0400 0020

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割込みをかけます。

このレジスタには、ホストモードでのみ有効なビットと、デバイスモードでのみ有効なビットがあります。このレジスタは、現在のモードも示します。rc_w1タイプの割込みステータスビットをクリアするには、アプリケーションは、そのビットに1を書き込む必要があります。

FIFOステータス割込みは読出し専用です。これらの割込みの処理中に、ソフトウェアがFIFOへの書込みまたはFIFOからの読出しを行うと、FIFO割込み条件は自動的にクリアされます。

初期化の前に割込みが発生するのを避けるためには、アプリケーションは、割込みビットのマスクを解除する前にOTG_GINTSTSレジスタをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	LPM INT	PTXFE	HCINT	HPRT INT	RST DET	Res.	IPXFR/ IN COMP ISO OUT	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	r	rc_w1		rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUMD NE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXFE	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット31 **WKUPINT**：レジューム／リモートウェイクアップ検出割込み

サスペンド（L2）またはLPM（L1）状態中のウェイクアップ割込み。

–サスペンド（L2）中

デバイスモードでは、USB上でレジュームが検出されたときにこの割込みがアサートされます。ホストモードでは、USB上でリモートウェイクアップが検出されたときにこの割込みがアサートされます。

–LPM（L1）中

この割込みは、USB上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット30 **SRQINT**：セッションリクエスト／新規セッション検出割込み

ホストモードでは、この割込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割込みは、V_{BUS} がBペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット29 **DISCINT**：切断検出割込み

デバイスの切断が検出されたときにアサートされます。

注： ホストモードでのみアクセス可能です。

ビット28 **CIDSCHG**：コネクタIDステータス変化

コアは コネクタIDステータスに変化があったとき、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット27 **LPMINT**：LPM割込み

デバイスモードでは、この割込みは、デバイスがLPMトランザクションを受信し、非エラー応答で応答したときにアサートされます。

ホストモードでは、この割込みは、デバイスがLPMトランザクションに非エラー応答で応答したとき、またはホストのコアがプログラムした回数（OTG_GLPMCFGのRETRYCNT）のLPMトランザクションを完了したときにアサートされます。

このフィールドは、OTG_GLPMCFGのLPMENビットが1にセットされたときにのみ有効です。

ビット26 **PTXFE**：周期的Tx FIFOエンプティ

周期的送信FIFOが半分エンプティか完全エンプティであり、周期的リクエストキューに少なくとも1つのエントリを書き込むためのスペースがあるときにアサートされます。半分エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFGのPTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット25 **HCINT**：ホストチャネル割込み

コアは、コアのチャネルの1つで割込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINTレジスタを読み出して、割込みが発生しているチャネルの正確な数を確認した後、対応するOTG_HCINTxレジスタを読み出して、割込みの正確な原因を確認する必要があります。アプリケーションはOTG_HCINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット24 **HPRTINT**：ホストポート割込み

コアは、ホストモードの OTG_FS コントローラポートの1つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRTレジスタを読み出して、この割込みの原因となった正確なイベントを特定する必要があります。アプリケーションはOTG_HPRTレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット23 **RSTDET**：リセット検出割込み

デバイスモードでは、この割込みは、デバイスがサスペンド状態にあるときに部分パワーダウンモードでUSB上にリセットが検出されたときにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 **IPXFR**：不完全な周期的転送

ホストモードでは、コアは、現在のフレームでスケジュールされていて、まだペンディング中の不完全な周期的トランザクションが存在するとき、この割込みビットをセットします。

INCOMPISOOUT：不完全アイソクロナスOUT転送

デバイスモードでは、コアは、現在のフレームで転送が完了していないアイソクロナスOUTエンドポイントが少なくとも1つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

ビット20 **IISOIXFR**：不完全アイソクロナスIN転送

コアは、現在のフレームで転送が完了していないアイソクロナスINエンドポイントが少なくとも1つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット19 **OEPINT**：OUTエンドポイント割込み

コアは、コアのOUTエンドポイントの1つで割込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したOUTエンドポイントの正確な数を確認した後、対応するOTG_DOEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DOEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット18 **IEPINT**：INエンドポイント割込み

コアは、コアのINエンドポイントの1つで割込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したINエンドポイントの正確な数を確認した後、対応するOTG_DIEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DIEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EOPF**：周期的フレーム終了割込み

現在のフレームで、OTG_DCFGレジスタの周期的フレームインターバルフィールド（OTG_DCFGのPFIVLビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット14 **ISOODRP**：アイソクロナスOUTパケットドロップ割込み

コアは、アイソクロナスOUTエンドポイントの最大サイズのパケットに対応できるだけの十分なスペースがRx FIFOになかったためにRx FIFOにアイソクロナスOUTパケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット13 **ENUMDNE**：エニュメレーションの完了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTSレジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット12 **USBRST**：USBのリセット

コアはUSB上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット11 **USBSUSP**：USBのサスペンド

コアはUSB上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが1つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット10 **ESUSP**：早期サスペンド

コアはUSB上で3 ms間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **GONAKEFF**：グローバルOUT NAKの有効性

アプリケーションによってセットされたOTG_DCTL レジスタのセットグローバルOUT NAKビット（OTG_DCTLのSGONAKビット）がコアで有効になったことを示します。このビットは、OTG_DCTLレジスタのクリアグローバルOUT NAKビット（OTG_DCTLのCGONAKビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット6 **GINAKEFF**：グローバルIN非周期的NAKの有効性

アプリケーションによってセットされたOTG_DCTLレジスタのセットグローバル 非周期的IN NAKビット（OTG_DCTLのSGINAKビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバルIN NAKビットをサンプリングしました。このビットは、OTG_DCTLレジスタのクリアグローバル非周期的IN NAKビット（OTG_DCTLのCGＩNAKビット）をクリアすることによってクリアできます。

この割込みは、必ずしもNAKハンドシェイクがUSBに送出されることを意味しません。STALLビットはNAKビットより優先されます。

注： デバイスモードでのみアクセス可能です。

ビット5 **NPTXFE**：非周期的Tx FIFOエンプティ

この割込みは、非周期的Tx FIFOが半分エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも1つのエントリを書き込むスペースがあるときにアサートされます。半分エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタの非周期的TxFIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット4 **RXFLVL**：Rx FIFO非エンプティ

Rx FIFOから読み出されるべきペンディング中のパケットが少なくとも1つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット3 **SOF**：フレーム開始（スタート･オブ･フレーム）

ホストモードでは、SOF（FS）またはキープアライブ（LS）がUSBで送信されることを示すために、コアによってこのビットがセットされます。アプリケーションは、このビットに1を書き込むことによって、割込みをクリアする必要があります。

デバイスモードでは、USB上でSOFトークンが受信されたことを示すために、コアでこのビットがセットされます。アプリケーションは、OTG_DSTSレジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割込みは、コアがFSで動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると”1”を返すことがあります。パワーオンリセット後すぐに”1”が読み出された場合、そのレジスタのビットが、SOFが送信された（ホストモードの場合）またはSOFが受信された（デバイスモードの場合）ことを示すわけではありません。この割込みの読出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット2 **OTGINT**：OTG割込み

コアは、OTGプロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションはOTG割込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割込みの原因となった正確なイベントを確認する必要があります。アプリケーションはOTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット1 **MMIS**：モード不一致割込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY応答によってAHB上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

72.15.7 OTG割込みマスクレジスタ（OTG_GINTMSK）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタは、コア割込みレジスタとともに働き、アプリケーションに割込みをかけます。割込みビットがマスクされると、そのビットに関連する割込みは生成されません。ただし、その割込みに対応するコア割込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	Res.	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	

ビット31 **WUIM**：レジューム／リモートウェイクアップ検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット30 **SRQIM**：セッションリクエスト／新規セッション検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット29 **DISCINT**：切断検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット28 **CIDSCHGM**：コネクタIDステータス変化マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット27 **LPMINTM**：LPM割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット26 **PTXFEM**：周期的Tx FIFOエンプティマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット25 **HCIM**：ホストチャネル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット24 **PRTIM**：ホストポート割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット23 **RSTDETM**：リセット検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット22 予約済みであり、リセット値に保持する必要があります。

ビット21 **IPXFRM**：不完全周期的転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。
IISOOXFRM：不完全アイソクロナスOUT転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット20 **IISOIXFRM**：不完全アイソクロナスIN転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット19 **OEPINT**：OUTエンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット18 **IEPINT**：INエンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EOPFM**：周期的フレーム終了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット14 **ISOODRPM**：アイソクロナスOUTパケットドロップ割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット13 **ENUMDNEM**：エニュメレーション完了マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット12 **USBRST**：USBリセットマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット11 **USBSUSPM**：USBサスペンドマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードでのみアクセス可能です。

ビット10 **ESUSPM**：早期サスペンドマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードでのみアクセス可能です。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **GONAKEFFM**：グローバルOUT NAK有効マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードでのみアクセス可能です。

ビット6 **GINAKEFFM**：グローバル非周期的IN NAK有効マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードでのみアクセス可能です。

ビット5 **NPTXFEM**：非周期的Tx FIFOエンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： ホストモードでのみアクセス可能です。

ビット4 **RXFLVLM**：受信FIFO非エンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット3 **SOFM**：フレーム開始マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット2 **OTGINT**：OTG割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット1 **MMISM**：モード不一致割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット0 予約済みであり、リセット値に保持する必要があります。

72.15.8 OTG受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR）

読出しアドレスオフセット：0x01C

リセット値：0x0000 0000

この説明は、デバイスモードのレジスタOTG_GRXSTSR用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信FIFOの最上部の内容が返されます。

コアは、受信FIFOがエンプティのときには、受信ステータスの読出しを無視して0x0000 0000の値を返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **STSPHST**：ステータスフェーズ開始

制御書込み転送のステータスフェーズの開始を示します。このビットは、OUT転送完了のPKTSTSパターンに合わせてセットされます。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24:21 **FRMNUM[3:0]**：フレーム番号

これはUSB上でパケットを受信するフレーム番号の下位4bitです。このフィールドは、アイソクロナスOUTエンドポイントがサポートされる場合にのみサポートされます。

ビット20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。
0001：グローバルOUT NAK（割込みをトリガします）。
0010：OUTデータパケットが受信されました。
0011：OUT転送が完了しました（割込みをトリガします）。
0100：SETUPトランザクションが完了しました（割込みをトリガします）。
0110：SETUPデータパケットが受信されました。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信したOUTデータパケットのデータPIDを示します。
00：DATA0
10:DATA1

ビット14:4 **BCNT[10:0]**：バイト数

受信したデータパケットのバイト数を示します。

ビット3:0 **EPNUM[3:0]**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

72.15.9 OTG受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR）

読出しアドレスオフセット：0x01C

リセット値：0x0000 0000

この説明は、ホストモードのレジスタOTG_GRXSTSR用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信FIFOの最上部の内容が返されます。

コアは、受信FIFOがエンプティのときには、受信ステータスの読出しを無視して0x0000 0000の値を返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。
0010：INデータパケットが受信されました。
0011：IN転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信パケットのデータPIDを示します。
00：DATA0
10:DATA1

ビット14:4 **BCNT[10:0]**：バイト数

受信したINデータパケットのバイト数を示します。

ビット3:0 **CHNUM[3:0]**：チャネル番号

現在の受信パケットが属するチャネル番号を示します。

72.15.10 OTGステータス読出しおよびポップレジスタ（OTG_GRXSTSP）

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

この説明は、デバイスモードのレジスタOTG_GRXSTSP用です。

読み出すと受信FIFOの最上部の内容が返されるOTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFOの最上部のデータエントリもポップアウトされます。

コアは、受信FIFOがエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000の値を返します。アプリケーションは、コア割込みレジスタの受信FIFO非エンプティビット（OTG_GINTSTSのRXFLVLビット）がアサートされたときにのみ、受信ステータスFIFOをポップしなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **STSPHST**：ステータスフェーズ開始

制御書込み転送のステータスフェーズの開始を示します。このビットは、OUT転送完了のPKTSTSパターンに合わせてセットされます。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24:21 **FRMNUM[3:0]**：フレーム番号

これはUSB上でパケットを受信するフレーム番号の下位4bitです。このフィールドは、アイソクロナスOUTエンドポイントがサポートされる場合にのみサポートされます。

ビット20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。
0001：グローバルOUT NAK（割込みをトリガします）。
0010：OUTデータパケットが受信されました。
0011：OUT転送が完了しました（割込みをトリガします）。
0100：SETUPトランザクションが完了しました（割込みをトリガします）。
0110：SETUPデータパケットが受信されました。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信したOUTデータパケットのデータPIDを示します。
00：DATA0
10:DATA1

ビット14:4 **BCNT[10:0]**：バイト数

受信したデータパケットのバイト数を示します。

ビット3:0 **EPNUM[3:0]**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

72.15.11 OTGステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

この説明は、ホストモードのレジスタOTG_GRXSTSP用です。

読み出すと受信FIFOの最上部の内容が返されるOTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFOの最上部のデータエントリもポップアウトされます。

コアは、受信FIFOがエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000の値を返します。アプリケーションは、コア割込みレジスタの受信FIFO非エンプティビット（OTG_GINTSTSのRXFLVLビット）がアサートされたときにのみ、受信ステータスFIFOをポップしなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。
0010：INデータパケットが受信されました。
0011：IN転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信パケットのデータPIDを示します。
00：DATA0
10:DATA1

ビット14:4 **BCNT[10:0]**：バイト数

受信したINデータパケットのバイト数を示します。

ビット3:0 **CHNUM[3:0]**：チャネル番号

現在の受信パケットが属するチャネル番号を示します。

72.15.12 OTG受信FIFOサイズレジスタ（OTG_GRXFSIZ）

アドレスオフセット：0x024

リセット値：0x0000 0200

アプリケーションはRxFIFOに割り当てられなければならないRAMサイズをプログラムできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RXFD[15:0]**：Rx FIFOの深さ

この値は32bitワード単位です。

最小値は16です。

プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

72.15.13 OTGホスト非周期的送信FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント0送信FIFOサイズ（OTG_DIEPTXF0）

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NPTXFD/TX0FD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSA/TX0FSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ホストモード

ビット31:16 **NPTXFD[15:0]**：非周期的Tx FIFOの深さ

この値は32bitワード単位です。

最小値は16です。

プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット15:0 **NPTXFSA[15:0]**：非周期的送信RAM開始アドレス

このフィールドは、非周期的送信FIFO RAMのメモリ開始アドレスを設定します。

デバイスモード

ビット31:16 **TX0FD**：エンドポイント0 TxFIFOの深さ

この値は32bitワード単位です。

最小値は16です。

プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット15:0 **TX0FSA**：エンドポイント0送信RAM開始アドレス

このフィールドは、エンドポイント0送信FIFO RAMのメモリ開始アドレスを設定します。

72.15.14 OTG非周期的送信FIFO／キューステータスレジスタ（OTG_HNPTXSTS）

アドレスオフセット：0x02C

リセット値：0x0008 0200

注： デバイスモードでは、このレジスタは無効です。

この読出し専用レジスタは、非周期的Tx FIFOおよび非周期的送信リクエストキューの空きスペース情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	NPTXQTOP[6:0]							NPTQXSAV[7:0]							
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSAV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:24 **NPTXQTOP[6:0]**：非周期的送信リクエストキューの先頭

MACによって処理中の非周期的Txリクエストキューのエントリ。

ビット30:27：チャネル／エンドポイント番号

ビット26:25：

00：IN/OUTトークン

01：ゼロ長送信パケット（デバイスIN／ホストOUT）

11：チャネル停止コマンド

ビット24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット23:16 **NPTQXSAV[7:0]**：非周期的送信リクエストキューの使用可能スペース

非周期的送信リクエストキューの使用可能な空きスペースの量を示します。このキューはINおよびOUTリクエストの両方を保持します。

0：非周期的送信リクエストキューはフルです。

1：1位置が使用可能です。

2：2位置が使用可能です。

n：n位置が使用可能です （0 ≤ n ≤ 8）。

その他：予約済み

ビット15:0 **NPTXFSAV[15:0]**：使用可能な非周期的Tx FIFOスペース

非周期的Tx FIFO内で使用できる空きスペースの量を示します。

値は32bitワード単位です。

0：非周期的Tx FIFOはフルです。

1：1ワードが使用可能です。

2：2ワードが使用可能です。

n：nワードが使用可能です （0 ≤ n ≤ 512）

その他：予約済み

72.15.15 OTG一般コア設定レジスタ（OTG_GCCFG）

アドレスオフセット：0x038

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCD EN	BCDEN	PWR DWN
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2 DET	SDET	PDET	DCDET
												r	r	r	r

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **VBDEN**：V_{BUS} 検出の有効化

V_{BUS} レベル検出用コンパレータを有効にして、V_{BUS} の有効レベルを検出します（USBホストおよびデバイス動作用のV_{BUS} パッド上で）。HNPまたはSRPサポートが有効であれば、V_{BUS} コンパレータはVBDEN値とは関係なく、自動的に有効化されます。

0 = V_{BUS} 検出無効

1 = V_{BUS} 検出有効

ビット20 **SDEN**：2次検出（SD）モードの有効化

このビットは、BCDをSDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット19 **PDEN**：1次検出（PD）モードの有効化

このビットは、BCDをPDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット18 **DCDEN**：データ接触検出（DCD）モードの有効化

このビットは、BCDをDCDモードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、またはOFF）を1つだけ選択する必要があります。

ビット17 **BCDEN**：バッテリ充電検出（BCD）の有効化

このビットは、USBデバイス内のBCDサポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHYはBCDによって完全に制御され、通常の通信では使用できなくなります。BCDの検出完了後は、通常のUSB動作を可能にするために、このビットを「0」にクリアしてBCDをOFFモードにします。

ビット16 **PWRDWN**：FS PHYのパワーダウン制御

送信／受信のためにFS PHYを起動するために使用されます。リセットされている場合、PHYはパワーダウン状態に保たれます。セットする場合、BCD機能をオフにする必要があります（BCDEN=0）。

0：USB FS PHYは無効です。

1：USB FS PHYは有効です。

ビット15:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **PS2DET**：DMプルアップ検出ステータス

このビットは、PD中にのみアクティブになってDM電圧レベルとVLGC閾値の比較の結果を示します。通常の状況では、DMレベルは閾値を下回る必要があります。閾値を上回る場合は、DMが外部的にハイレベルにプルアップされることを意味します。これは、PS2ポート（DPおよびDMラインの両方をプルアップする）への接続、またはBCD仕様に準拠しない一部の独自の充電器への接続によって発生します。

0：通常のポートが検出されました（SDP、CDP、またはDCPに接続）。

1：PS2ポートまたは独自の充電器が検出されました。

ビット2 **SDET**：2次検出（SD）ステータス

このビットは、SDの結果を示します。

0：CDPが検出されました。

1：DCPが検出されました。

ビット1 **PDET**：1次検出（PD）ステータス

このビットは、PDの結果を示します。

0：BCDサポートは検出されませんでした（SDPまたは独自デバイスに接続）。

1：BCDのサポートが検出されました（CDP、またはDCPに接続）。

ビット0 **DCDET**：データ接触検出（DCD）ステータス

このビットは、DCDの結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

72.15.16 OTGコアIDレジスタ（OTG_CID）

アドレスオフセット：0x03C

リセット値：0x00003000

これは、リセット値として製品IDを含むレジスタです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRODUCT_ID[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRODUCT_ID[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRODUCT_ID[31:0]**：製品IDフィールド

アプリケーションによってプログラム可能なIDフィールド。

72.15.17 OTGコアLPM設定レジスタ（OTG_GLPMCFG）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EN BESL	LPMRCNTSTS[2:0]			SND LPM	LPMRCNT[2:0]			LPMCHIDX[3:0]				L1RSMOK
			rw	r	r	r	rs	rw	rw	rw	rw	rw	rw	rw	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLP STS	LPMRSP[1:0]		L1DS EN	BESLTHRS[3:0]				L1SS EN	REM WAKE	BESL[3:0]				LPM ACK	LPM EN
r	r	r	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ENBESL**：ベストエフォート型サービスのレイテンシの有効化

このビットはLPM正誤表で定義されているようにBESL機能を有効にします。

0：コアは以下の文書に記載されているように機能します。
USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007

1：コアはLPM正誤表に記載されているように機能します。
USB 2.0 ECNの正誤表：リンク電源管理（LPM）- 2007年7月

注： この文書では更新された動作（LPM正誤表に記載）のみが考慮されているため、アプリケーションのソフトウェアによりENBESLビットに1をセットする必要があります。

ビット27:25 **LPMRCNTSTS[2:0]**：LPMリトライカウントステータス

現在のLPMシーケンスでは、送信すべきLPMホストの再試行回数がまだ残っています。

注： ホストモードでのみアクセス可能です。

ビット24 **SNDLPM**：送信LPMトランザクション

アプリケーションソフトウェアがこのビットをセットすると、2つのトークンEXTおよびLPMを含むLPMトランザクションが送信されます。デバイスから有効な応答（STALL、NYET、またはACK）を受信した時点、またはコアがプログラムされた回数のLPM再試行を送信し終えた時点で、このビットはハードウェアによってクリアされます。

注： このビットは、ホストがローカルポートに接続されている場合のみセットする必要があります。

注： ホストモードでのみアクセス可能です。

ビット23:21 **LPMRCNT[2:0]**：LPMリトライカウント

デバイスがERROR応答を返すと、このビットは追加のLPM再試行回数となります。追加の再試行は、有効なデバイスからの応答（STALL、NYET、またはACK）が受信されるまで、ホストによって実行されます。

注： ホストモードでのみアクセス可能です。

ビット20:17 **LPMCHIDX[3:0]**：LPMチャネルインデックス

ローカルデバイスへのLPMトランザクションの送信中にLPMトランザクションを適用するべきチャネルの番号を示します。LPMチャネルインデックスに基づき、コアは自動的に対応するチャネルでプログラムされたデバイスアドレスとエンドポイント番号をLPMトランザクションに挿入します。

注： ホストモードでのみアクセス可能です。

ビット16 **L1RSMOK**：スリープ状態レジュームOK

デバイスまたはホストがスリープ状態からレジュームを開始できることを示します。このビットはLPMスリープ状態（L1）のときに有効です。このビットは50 µs（$T_{L1Residency}$）の遅延後にスリープモードでセットされます。

このビットは、SLPSTS = 0のときにリセットされます。

1：アプリケーションまたはホストはスリープ状態からレジュームを開始できます。

0：アプリケーションまたはホストはスリープ状態からレジュームを開始できません。

ビット15 **SLPSTS**：ポートスリープステータス

デバイスモード：

このビットは、USBバス上で スリープ 状態が発生している限り、セットされます。このコアは、LPMトランザクションにACK応答が送信され、$T_{L1TokenRetry}$ タイマが時間切れになっている場合に、スリープ 状態に入ります。PHYクロックを停止させるには、アプリケーションでOTG_PCGCCTLのSTPPCLKビットをセットすることにより、PHYサスペンド入力信号をアサートする必要があります。

アプリケーションはスリープへの遷移を確認する際に、LPMRSPのACKではなく、SLPSTSを信頼する必要があります。

コアは、以下の場合にスリープ状態から抜け出します。

– USBライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL の RWUSIG ビットへ書き込むときやデバイスをリセットまたはソフト切断するとき。

ホストモード：

デバイスからACK応答を受け取ったローカルポートへのコアによるLPM トランザクション の成功の副作用として、ホストはスリープ状態（L1）へ遷移します。このビットの読出し値は、ポートの現在のスリープ状態を反映しています。

コアは、次の動作後にこのビットをクリアします。

– コアがリモートL1ウェイクアップ信号を検出

– アプリケーションで、OTG_HPRT レジスタの PRST ビットまたは PRES ビットをセットします、または

– アプリケーションで、コア割込みレジスタの L1 レジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（OTG_GINTSTSのWKUPINTまたはDISCINTビット）をセット

0：コアはL1状態ではありません。

1：コアはL1状態です。

ビット14:13 **LPMRSP[1:0]**：LPM応答

デバイスモード：

受信したLPMトランザクションへのコアの応答は、これら2つのビットに反映されます。

ホストモード：

ローカルデバイスから受け取ったLPMトランザクションへのハンドシェイク応答

11：ACK

10：NYET

01：STALL

00：ERROR（ハンドシェイク応答なし）

ビット12 **L1DSEN**：L1ディープスリープの有効化

L1スリープモードにおけるPHYのサスペンドを有効にします。L1スリープモード時に最大限の節電を行うために、いかなる場合もアプリケーションのソフトウェアによって、このビットに「1」をセットする必要があります。

ビット11:8 **BESLTHRS[3:0]**：BESL閾値

デバイスモード：

BESL値がこのBESL_Thres[3:0] フィールドの規定値以上である場合、コアはPHYをL1状態でディープ低消費電力モードに移行させます。

ホストモード：

コアはPHYをL1状態でディープ低消費電力モードに移行させます。BESLTHRS[3:0] は、デバイスから開始されたレジュームを検出した場合に、レジュームシグナリングに USB バス上のホスト（$T_{L1HubDrvResume2}$）が反映されるまでの時間を指定します。

ホストモードでは、BESLTHRS には 1100b より大きい値を設定してはなりません。この値が$T_{L1HubDrvResume2}$の最大値を超えてしまうからです。

Thres[3:0] ホストモードレジュームシグナリング時間（µs）：

0000：75

0001：100

0010：150

0011：250

0100：350

0101：450

0110：950

その他の値は予約済みです。

ビット7 **L1SSEN**：L1シャロースリープの有効化

L1 SLEEPモードにおけるPHYのサスペンドを有効にします。L1 SLEEPモード時に最大限の節電を行うために、いかなる場合もアプリケーションのソフトウェアによって、このビットに「1」をセットする必要があります。

ビット6 **REMWAKE**：bRemoteWake値

ホストモード：

LPMトランザクションのwIndexフィールドの送信されるリモートウェイクアップの値

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、またはSTALL応答がLPMトランザクションに送信される場合に、受信したLPMトークンbRemoteWake bmAttributeで更新されます。

ビット5:2 **BESL[3:0]**：ベストエフォート型サービスレイテンシ

ホストモード：

LPMトランザクションの送信されるBESLの値この値は、ホストから開始されたレジュームの処理時間（$T_{L1HubDrvResume1}$ ）の間にレジュームを開始するためにも使用されます。

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、またはSTALL応答がLPMトランザクションに送信される場合に、受信したLPMトークンBESL bmAttributeで更新されます。

BESL[3:0]T_{BESL} (µs)

0000：125
0001：150
0010：200
0011：300
0100：400
0101：500
0110：1000
0111：2000
1000：3000
1001：4000
1010：5000
1011：6000
1100：7000
1101：8000
1110：9000
1111：10000

ビット1 **LPMACK**：LPMトークン確認応答の有効化

デバイスのアプリケーションソフトウェアによって事前にプログラムされたLPMトークンへのハンドシェイク応答

1：ACK

ACKは事前にプログラムされているものの、コア･デバイスは成功したLPMトランザクションに対してのみACK応答を返します。LPMトランザクションは次の場合に成功します。

– EXT トークンまたは LPM トークンのどちらにも PID/CRC5 エラーがない（それ以外の場合はERROR）。
– LPMトランザクションで受信した有効なbLinkState = 0001B（L1）（それ以外の場合はSTALL）。
– 送信シーケンスにペンディング中のデータがない（それ以外の場合はNYET）。

0：NYET

事前にプログラムされたソフトウェアビットは、次の場合に、LPMトークンへの応答に上書きされます。

– 受信したbLinkStateがL1でない場合（STALL応答）、または
– いずれかのLPMトークンパケットで、破損によるエラーが検出された場合（エラー応答）

注： デバイスモードでのみアクセス可能です。

ビット0 **LPMEN**：LPMサポートの有効化

アプリケーションは、このビットを使用して、OTG_FS コアLPM機能を制御します。

コアが非LPM対応ホストとして動作する場合、ホストは接続されたデバイスまたはハブにLPMモードを有効にするように要求することはできません。

コアが非LPM対応デバイスとして動作する場合、デバイスはいかなるLPMトランザクションにも応答できません。

0：LPM機能は無効です。
1：LPM機能は有効です。

72.15.18 OTGホスト周期的送信FIFOサイズレジスタ （OTG_HPTXFSIZ）

アドレスオフセット：0x100

リセット値：0x0200 0400

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PTXFSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTXSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **PTXFSIZ[15:0]**：ホスト周期的Tx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。

ビット15:0 **PTXSA[15:0]**：ホスト周期的Tx FIFOの開始アドレス
このフィールドは、周期的送信FIFO RAMのメモリ開始アドレスを設定します。

72.15.19 OTGデバイスINエンドポイント送信FIFO xサイズレジスタ （OTG_DIEPTXFx）

アドレスオフセット：0x104 + 0x04 * (x-1)(x = 1から5)

リセット値:BLOCK 1: 0x0200 0400

リセット値:BLOCK 2: 0x0200 0600

リセット値:BLOCK 3: 0x0200 0800

リセット値:BLOCK 4: 0x0200 0A00

リセット値:BLOCK 5: 0x0200 0C00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INEPTXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTXSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **INEPTXFD[15:0]**：INエンドポイントTx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。

ビット15:0 **INEPTXSA[15:0]**：INエンドポイントFIFOx送信RAM開始アドレス
このフィールドはINエンドポイント送信FIFOxのメモリ開始アドレスを含みます。アドレスは32bitメモリロケーションに整列されている必要があります。

72.15.20 ホストモードレジスタ

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

ホストモードレジスタは、ホストモードでのコアの動作に影響を与えます。.ホストモードレジスタは、結果が定義されていないため、デバイスモードではアクセスしないでください。ホストモードレジスタは、次のように分類することができます。

72.15.21 OTGホスト設定レジスタ（OTG_HCFG）

アドレスオフセット：0x400

リセット値：0x0000 0000

このレジスタは、電源投入後のコアを設定します。ホストを初期化した後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLSPCS[1:0]	
													r	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FSLSS**：FSおよびLSのみのサポート

アプリケーションは、このビットを使用して、コアのエニュメレーション速度を制御します。このビットを使用して、アプリケーションは、接続されたデバイスがHSトラフィックをサポートする場合でも、FSホストとしてコアにエニュメレーションさせることができます。初期プログラミングの後は、このフィールドを変更しないでください。

1：接続されたデバイスがHSトラフィックをサポートしている場合でも、FS/LSのみ（読出し専用）。

ビット1:0 **FSLSPCS[1:0]**：FS/LS PHYクロックの選択

コアがFSホストモードのとき、

01：PHYクロックは48 MHzで動作

その他：予約済み

コアがLSホストモードのとき、

00：予約済み

01：PHYクロック周波数として48 MHzを選択

10：PHYクロック周波数として6 MHzを選択

11：予約済み

注： FSLSPCS は、接続されたデバイスの速度に応じて、接続イベント時にセットする必要があります（このビットを変更した後、ソフトウェアリセットを実行する必要があります）。

72.15.22 OTGホストフレームインターバルレジスタ（OTG_HFIR）

アドレスオフセット：0x404

リセット値：0x0000 EA60

このレジスタは、OTG_FS コントローラがエニュメレーションした現在の速度でのフレームインターバル情報を格納します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLD CTRL
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRIVL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RLDCTRL**：再ロード制御

このビットを使用して、ランタイム時にHFIRレジスタの動的再ロードが可能です。

0：HFIRは動的に再ロードすることができません。

1：HFIRはランタイム時に動的に再ロードすることができます。

このビットは、初期設定時にプログラムする必要があり、実行時にその値を変更してはなりません。

注意： RLDCTRL = 0は推奨しません。

ビット15:0 **FRIVL[15:0]**：フレームインターバル

アプリケーションがこのフィールドにプログラムする値は、2つの連続したSOF（FS）またはキープアライブトークン（LS）の間のインターバルを指定します。このフィールドは、必要なフレームインターバルを構成するPHYクロック数を含みます。アプリケーションは、ホストポート制御およびステータスレジスタのポートイネーブルビット（OTG_HPRTのPENAビット）がセットされた後のみ、このレジスタに値を書き込むことができます。値がプログラムされていない場合、コアは、ホスト設定レジスタのFS/LS PHYクロック選択フィールド（OTG_HCFGのFSLSPCS）で指定されたPHYクロックに基づいて値を計算します。初期設定の後は、RLDCTRLビットがセットされている場合を除いて、このフィールドの値を変更しないでください。そのような場合、FRIVLには 各SOFイベントが再ロードされます。

–フレームインターバル = 1ms × (FRIVL - 1)

72.15.23 OTGホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）

アドレスオフセット：0x408

リセット値：0x0000 3FFF

このレジスタは、現在のフレーム番号を示します。また、現在のフレームの残り時間（PHYクロック数）も示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FTREM[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRNUM[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **FTREM[15:0]：**フレームの残り時間

現在のフレームの残り時間をPHYクロック数で示します。このフィールドはPHYクロックごとにデクリメントされます。値がゼロになると、このフィールドにはフレームインターバルレジスタの値が再ロードされ、新しいSOFがUSBに送信されます。

ビット15:0 **FRNUM[15:0]：**フレーム番号

このフィールドはUSB上で新しいSOFが送信されるたびにインクリメントされ、0x3FFFに達すると0にクリアされます。

72.15.24 OTGホスト周期的送信FIFO ／キューステータスレジスタ（OTG_HPTXSTS）

アドレスオフセット：0x410

リセット値：0x0008 0100

この読出し専用レジスタは、周期的Tx FIFOおよび周期的送信リクエストキューの空きスペース情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PTXQTOP[7:0]								PTXQSAV[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTXFSAVL[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 **PTXQTOP[7:0]：**周期的送信リクエストキューの先頭

これはMACによって処理中の周期的Txリクエストキューのエントリを示します。
このレジスタは、デバッグに使用されます。
ビット31：奇数／偶数フレーム
0：偶数フレームで送信
1：奇数フレームで送信
ビット30:27：チャネル／エンドポイント番号
ビット26:25：タイプ
00：IN/OUT
01：長さゼロのパケット
11：チャネルコマンドの無効化
ビット24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット23:16 **PTXQSAV[7:0]：**周期的送信リクエストキューの使用可能スペース

周期的送信リクエストキューで書込みに使用できる空きロケーションの数を示します。このキューはINおよびOUTリクエストの両方を保持します。
00：周期的送信リクエストキューはフルです。
01：1位置が使用可能です。
10：2位置が使用可能です。
bxn：n位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット15:0 **PTXFSAVL[15:0]**：周期的送信データFIFOの使用可能スペース
周期的Tx FIFOで書込みに使用できる空きロケーションの数を示します。
値は32bitワード単位です。
0000：周期的Tx FIFOはフルです。
0001：1ワードが使用可能です。
0010：2ワードが使用可能です。
bxn：nワードが使用可能です （0 ≤ n ≤ PTXFD）
その他：予約済み

72.15.25 OTGホスト全チャネル割込みレジスタ（OTG_HAINT）

アドレスオフセット：0x414

リセット値：0x0000 0000

チャネル上で重大なイベントが発生すると、ホスト全チャネル割込みレジスタは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）を使用して、アプリケーションに割込みをかけます。これを 図892に示します。チャネルごとに1つの割込みビットがあります（最大16bit）。このレジスタのビットは、アプリケーションが対応するホストチャネルx割込みレジスタのビットをセット／クリアしたときにセット／クリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HAINT[15:0]**：チャネル割込み
チャネル0に対してビット0、チャネル15に対してビット15

72.15.26 OTGホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）

アドレスオフセット：0x418

リセット値：0x0000 0000

ホスト全チャネル割込みマスクレジスタは、ホスト全チャネル割込みレジスタとともに働き、チャネル上でイベントが発生したときに、アプリケーションに割込みをかけます。チャネルごとに1つの割込みマスクビットがあります（最大16bit）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINTM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HAINTM[15:0]**：チャネル割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

チャネル0に対してビット0、チャネル15に対してビット15

72.15.27 OTGホストポート制御およびステータスレジスタ（OTG_HPRT）

アドレスオフセット：0x440

リセット値：0x0000 0000

このレジスタは、ホストモードでのみ使用可能です。現時点ではOTGホストは1つのポートのみをサポートしています。

1つのレジスタは、ポートごとのUSBリセット、イネーブル、サスペンド、レジューム、接続ステータス、テストモードなどのUSBポート関連情報を保持します。図892に示されています。このレジスタのrc_w1bitは、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTSのHPRTINTビット）を使用して、アプリケーションへの割込みをトリガできます。ポート割込みの場合、アプリケーションは、このレジスタを読み出して、割込みの原因となったビットをクリアする必要があります。rc_w1bitの場合、アプリケーションは、このビットに1を書き込んで、割込みをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD[1:0]		PTCTL [3]
													r	r	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTCTL[2:0]			PPWR	PLSTS[1:0]		Res.	PRST	PSUSP	PRES	POC CHNG	POCA	PEN CHNG	PENA	PCDET	PCSTS
rw	rw	rw	rw	r	r		rw	rs	rw	rc_w1	r	rc_w1	rc_w1	rc_w1	r

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:17 **PSPD[1:0]**：ポートスピード

このポートに接続されたデバイスのスピードを示します。

01：フルスピード

10：ロースピード

11：予約済み

ビット16:13 **PTCTL[3:0]**：ポートテスト制御

アプリケーションがこのフィールドにゼロ以外の値を書き込んで、ポートをテストモードにすると、対応するパターンがポートにシグナリングされます。

0000：テストモードは無効

0001：Test_Jモード

0010：Test_Kモード

0011：Test_SE0_NAKモード

0100：Test_Packetモード

0101：Test_Force_Enable

その他：予約済み

ビット12 **PPWR**：ポートの電力供給

アプリケーションは、このフィールドを使用して、このポートへの電力供給を制御し、コアは、過電流条件が発生すると、このビットをクリアします。

0：パワーオフ

1：パワーオン

ビット11:10 **PLSTS[1:0]**：ポートラインステータス

USBデータラインの現在のロジックレベルを示します。

ビット10：OTG_DPのロジックレベル

ビット11：OTG_DMのロジックレベル

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PRST**：ポートのリセット

アプリケーションがこのビットをセットすると、このポートでリセットシーケンスが開始されます。アプリケーションはリセット時間を計時し、リセットシーケンスが完了したらこのビットをクリアする必要があります。

0：ポートはリセット中ではありません。

1：ポートはリセット中です。

アプリケーションは、ポートのリセットが開始されるまで、少なくとも10 msの間、このビットをセット状態に保つ必要があります。USB規格では最大限度が規定されていませんが、アプリケーションは、このビットをクリアする前に、必要最小時間に加えて、さらに10 msの間、このビットをセット状態に保つことができます。

ハイスピード：50ms

フルスピード／ロースピード：10ms

ビット7 **PSUSP**：ポートのサスペンド

アプリケーションは、このビットをセットして、このポートをサスペンドモードにします。コアは、このビットがセットされるとSOFの送信のみを停止します。PHYクロックを停止するには、アプリケーションはPHYのサスペンド入力ピンをアサートするポートクロック停止ビットをセットする必要があります。

このビットの読出し値は、ポートの現在のサスペンドステータスを反映しています。このビットは、リモートウェイクアップ信号の検出後、またはアプリケーションがこのレジスタのポートリセットビットまたはポートレジュームビット、またはコア割込みレジスタのレジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（それぞれOTG_GINTSTSのWKUPINTまたはDISCINTビット）をセットしたときに、コアによってクリアされます。

0：ポートはサスペンドモードではありません。

1：ポートはサスペンドモードです。

ビット6 **PRES**：ポートのレジューム

アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。コアは、アプリケーションがこのビットをクリアするまで、レジューム信号の駆動を続行します。

コアが、コア割込みレジスタのポートレジューム／リモートウェイクアップ検出割込みビット（OTG_GINTSTSのWKUPINTビット）によって示されるUSBリモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、切断条件を検出したときにこのビットをクリアします。このビットの読出し値は、コアが現在レジューム信号を駆動中であるかどうかを示します。

0：レジュームシグナリングを駆動しません。

1：レジュームシグナリングを駆動します。

LPMが有効で、コアがL1状態にあるとき、このビットは次のように動作します。

1。アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。

2。コアはOTG_GLPMCFGレジスタのBESLTHRS[3:0] フィールドに指定された所定時間まで、レジューム信号の駆動を続行します。

3。コアが、コア割込みレジスタのポートL1レジューム／リモートL1ウェイクアップ検出割込みビット（OTG_GINTSTSのWKUPINTビット）によって示されるUSBリモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、レジュームの最後にこのビットをクリアします。このビットは、コアとアプリケーション両方でセット／クリアできます。このビットは、ホストにデバイスが1つも接続されていない場合でもコアによってクリアされます。

ビット5 **POCCHNG**：ポート過電流の変化

コアは、このレジスタのポート過電流アクティブビット（ビット4）に変化があったとき、このビットをセットします。

ビット4 **POCA**：ポート過電流アクティブ

ポートの過電流条件を示します。

0：過電流条件がありません。

1：過電流条件があります。

ビット3 **PENCHNG**：ポートの有効／無効の変化

コアは、このレジスタのポートイネーブルビット2のステータスに変化があったとき、このビットをセットします。

ビット2 **PENA**：ポートの有効化

ポートは、リセットシーケンス後、コアによってのみ有効にされ、過電流条件、切断条件、またはアプリケーションがこのビットをクリアすることによって無効にされます。アプリケーションは、レジスタへの書込みによってこのビットをクリアすることはできません。クリアして、ポートを無効にすることだけができます。このビットは、アプリケーションへの割込みをトリガしません。

0：ポートは無効です。

1：ポートは有効です。

ビット1 **PCDET**：ポート接続の検出

コアは、デバイス接続が検出されると、このビットをセットして、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTSのHPRTINTビット）を使用して、アプリケーションへの割込みをトリガします。アプリケーションは、このビットに1を書き込むことによって、割込みをクリアする必要があります。

ビット0 **PCSTS**：ポート接続ステータス

0：ポートにデバイスが接続されていません。

1：ポートにデバイスが接続されています。

72.15.28 OTGホストチャネルx特性レジスタ（OTG_HCCHARx）

アドレスオフセット：0x500 + 0x20 * x, (x = 0から11)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHENA	CHDIS	ODD FRM	DAD[6:0]							MCNT[1:0]		EPTYP[1:0]		LSDEV	Res.
rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPDIR	EPNUM[3:0]				MPSIZ[10:0]										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **CHENA**：チャネルの有効化

このフィールドは、アプリケーションによってセットされOTGホストによってクリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

ビット30 **CHDIS**：チャネルの無効化

アプリケーションは、チャネルへの転送が完了する前でも、このビットをセットして、そのチャネル上での送信／受信を停止することができます。アプリケーションは、チャネルを無効として扱う前に、チャネル無効割込みを待つ必要があります。

ビット29 **ODDFRM**：奇数フレーム

このフィールドはOTGホストが奇数フレームで転送を実行しなければならないことを示すために、アプリケーションによってセット（リセット）されます。このフィールドは、周期的（アイソクロナスおよび割込み）トランザクションに対してのみ適用されます。

0：偶数フレーム

1：奇数フレーム

ビット28:22 **DAD[6:0]**：デバイスアドレス

このフィールドは、データソースまたはデータシンクとして機能する特定のデバイスを選択します。

ビット21:20 **MCNT[1:0]**：マルチカウント

このフィールドは、この周期的エンドポイントに対して、フレームごとに実行する必要があるトランザクションの数をホストに示します。非周期的転送の場合、このフィールドは使用されません。

00：予約済み。このフィールドは定義されていない結果をもたらします。

01：1つのトランザクション

10：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり2つです。

11：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり3つです。

注： このフィールドは、01以上にセットする必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

選択された転送タイプを示します。

00：制御

01：アイソクロナス

10：バルク

11：割込み

ビット17 **LSDEV**：ロースピードデバイス

このフィールドは、このチャネルがロースピードデバイスと通信中であることを示すために、アプリケーションによってセットされます。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EPDIR**：エンドポイントの方向
トランザクションがINかOUTかを示します。
0：OUT
1：IN

ビット14:11 **EPNUM[3:0]**：エンドポイント番号
データソースまたはデータシンクとして機能しているデバイスのエンドポイント番号を示します。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ
関連するエンドポイントの最大パケットサイズを示します。

72.15.29 OTGホストチャネルx割込みレジスタ（OTG_HCINTx）

アドレスオフセット：0x508 + 0x20 * x, (x = 0から11)

リセット値：0x0000 0000

このレジスタはUSBおよびAHB関連イベントに関するチャネルのステータスを示します。図892 に示されています。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）がセットされたとき、このレジスタを読み出す必要があります。アプリケーションがこれらのレジスタを読み出すためには、まず、ホスト全チャネル割込みレジスタ（OTG_HAINT）を読み出して、ホストチャネルx割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
					rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **DTERR**：データトグルエラー

ビット9 **FRMOR**：フレームのオーバーラン

ビット8 **BBERR**：バブルエラー

ビット7 **TXERR**：トランザクションエラー
USB上で以下のエラーの1つが発生したことを示します。
CRCチェック失敗
タイムアウト
ビットスタッフエラー
偽のEOP

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **ACK**：ACK応答受信／送信割込み

ビット4 **NAK**：NAK応答受信割込み

ビット3 **STALL**：STALL応答受信割込み

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CHH**：チャネルの停止

USBトランザクションエラーまたはアプリケーションによる無効化リクエストへの応答のいずれかの理由により、転送が異常終了したことを示します。

ビット0 **XFRC**：転送の完了

転送がエラーなく正常に完了しました。

72.15.30 OTGホストチャネルx割込みマスクレジスタ（OTG_HCINTMSKx）

アドレスオフセット：0x50C + 0x20 * x, (x = 0から11)

リセット値：0x0000 0000

このレジスタは、前のセクションで説明した各チャネルステータスのマスクを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	Res.	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
					rw	rw	rw	rw		rw	rw	rw		rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **DTERRM**：データトグルエラーマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット9 **FRMORM**：フレームオーバーランマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット8 **BBERRM**：バブルエラーマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット7 **TXERRM**：トランザクションエラーマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **ACKM**：ACK応答受信／送信割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット4 **NAKM**：NAK応答受信割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット3 **STALLM**：STALL応答受信割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **CHHM**：チャネル停止マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

72.15.31 OTGホストチャネルx転送サイズレジスタ（OTG_HCTSIZx）

アドレスオフセット：0x510 + 0x20 * x, (x = 0から11)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO PNG	DPID[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **DOPNG**：PING実行

このビット 転送のみに使用されます。このフィールドに1を設定すると、ホストにPINGプロトコルを実行させます。

注： IN転送ではこのビットをセットしないでください。このビットがIN転送に対してセットされると、そのチャネルが無効化されます。

ビット30:29 **DPID[1:0]**：データPID

アプリケーションは、このフィールドを最初のトランザクションで使用するPIDのタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：DATA0

10:DATA1

11：SETUP（制御）／予約済み（非制御）

ビット28:19 **PKTCNT [9:0]**：パケットカウント

このフィールドは、アプリケーションによって、送信（OUT）または受信（IN）されるパケットの予想数でプログラムされます。

ホストはOUT/INパケットの送信または受信が成功するたびに、このカウントをデクリメントします。このカウントがゼロに達すると、アプリケーションに割り込んで、正常完了を示します。

ビット18:0 **XFRSIZ[18:0]**：転送サイズ

OUTの場合、このフィールドは、ホストが転送時に送信するデータバイト数です。

INの場合、このフィールドは、アプリケーションが転送のために確保したバッファサイズです。アプリケーションは、INトランザクション（周期的および非周期的）の場合、このフィールドを最大パケットサイズの整数倍でプログラムすることが期待されます。

72.15.32 デバイスモードレジスタ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

72.15.33 OTGデバイス設定レジスタ（OTG_DCFG）

アドレスオフセット：0x800

リセット値：0x0220 0000

このレジスタは、電源投入後、または特定の制御コマンドあるいは列挙の後、コアをデバイスモードに設定します。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRATIM	Res.	Res.	PFIVL[1:0]		DAD[6:0]							Res.	NZLSOHSK	DSPD[1:0]	
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **ERRATIM**：不規則エラー割込みマスク

1：不規則エラー時の早期サスペンド割込みをマスクします。

0：不規則エラー時に早期サスペンド割込みが生成されます。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12:11 **PFIVL[1:0]**：周期的フレームインターバル

周期的フレーム終了割込みを使用してアプリケーションに通知しなければならないフレーム内の時間を示します。これを使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

00：フレームインターバルの80%

01：フレームインターバルの85%

10：フレームインターバルの90%

11：フレームインターバルの95%

ビット10:4 **DAD[6:0]**：デバイスアドレス

アプリケーションは、各SetAddress制御コマンドの後、このフィールドをプログラムする必要があります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **NZLSOHSK**：非ゼロ長ステータスOUTハンドシェイク

アプリケーションは、このフィールドを使用して、制御転送のステータスステージのOUTトランザクション中に、コアが非ゼロ長のデータパケットを受信したときに送信するハンドシェイクを選択できます。

1：非ゼロ長ステータスのOUTトランザクション時にはSTALLハンドシェイクを送信し、受信したOUTパケットをアプリケーションに送信しません。

0：受信したOUTパケット（ゼロ長または非ゼロ長）をアプリケーションに送信し、デバイスエンドポイント制御レジスタのNAKおよびSTALLビットに基づいてハンドシェイクを送信します。

ビット1:0 **DSPD[1:0]**：デバイスのスピード

アプリケーションがコアにエニュメレーションを要求するスピード、またはアプリケーションがサポートできる最高スピードを示します。ただし、実際のバススピードは、チャープシーケンスの完了後に決まり、コアが接続されるUSBホストのスピードに基づきます。

00：予約済み

01：予約済み

10：予約済み

11：フルスピード（USB 1.1トランシーバのクロックは48 MHzです）

72.15.34 OTGデバイス制御レジスタ（OTG_DCTL）

アドレスオフセット：0x804

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.
													rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL[2:0]			GONSTS	GINSTS	SDIS	RWUSIG
				rw	w	w	w	w	rw	rw	rw	r	r	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **DSBESLRJCT**：ディープスリープBESL拒否

コアは、プログラムされたBESL閾値より大きいBESL値を持つLPMリクエストを拒否します。NYET応答は、BESL閾値より大きいBESL値を持つLPMトークンに送信されます。デフォルトでは、ディープスリープBESL拒否機能は無効です。

ビット 17:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **POPRGDNE**：パワーオンプログラミング終了

アプリケーションは、このビットを使用して、パワーダウンモードからのウェイクアップ後に、レジスタのプログラミングが終了したことを示します。

ビット10 **CGONAK**：グローバルOUT NAKのクリア

このフィールドに1を書き込むと、グローバルOUT NAKがクリアされます。

ビット9 **SGONAK**：グローバルOUT NAKのセット

このフィールドに1を書き込むと、グローバルOUT NAKがセットされます。

アプリケーションは、このビットを使用して、すべてのOUTエンドポイントにNAKハンドシェイクを送信します。

アプリケーションがこのビットをセットするのは、コア割込みレジスタのグローバルOUT NAK有効ビット（OTG_GINTSTSのGONAKEFFビット）がクリアされていることを確認した後でなければなりません。

ビット8 **CGINAK**：グローバルIN NAKのクリア

このフィールドに1を書き込むと、グローバルIN NAKがクリアされます。

ビット7 **SGINAK**：グローバルIN NAKのセット

このフィールドに1を書き込むと、グローバル非周期的IN NAKがセットされます。アプリケーションは、このビットを使用して、すべての非周期的INエンドポイントにNAKハンドシェイクを送信します。

アプリケーションがこのビットをセットするのは、コア割込みレジスタのグローバルIN NAK有効ビット（OTG_GINTSTSのGINAKEFFビット）がクリアされていることを確認した後でなければなりません。

ビット6:4 **TCTL[2:0]**：テスト制御

000：テストモードは無効

001：Test_Jモード

010：Test_Kモード

011：Test_SE0_NAKモード

100：Test_Packetモード

101：Test_Force_Enable

その他：予約済み

ビット3 **GONSTS**：グローバルOUT NAKのステータス

0：ハンドシェイクは、FIFOステータスとNAKおよびSTALLビットの設定に基づいて送信されます。

1：使用可能なスペースに関係なくRx FIFOにはデータが書き込まれません。SETUPトランザクションを除き、すべてのパケットにNAKハンドシェイクを送信します。すべてのアイソクロナスOUTパケットがドロップされます。

ビット2 **GINSTS**：グローバルIN NAKのステータス

0：ハンドシェイクは、送信FIFO内の使用可能なデータに基づいて送信されます。

1：NAKハンドシェイクは、送信FIFO内の使用可能なデータに関係なく、非周期的INエンドポイントで送出されます。

ビット1 **SDIS**：ソフト切断

アプリケーションは、このビットを使用してUSB OTGコアにソフト切断を実行することを伝えます。このビットがセットされている限り、ホストではデバイスが接続されているとはみなされず、デバイスはUSB上で信号を受信しません。コアは、アプリケーションでこのビットがクリアされるまで、切断状態にとどまります。

0：通常動作。ソフト切断後にこのビットがクリアされると、コアはUSBホストに対してデバイス接続イベントを生成します。デバイスが再接続されると、USBホストはデバイスのエニュメレーションをリスタートします。

1：コアはUSBホストに対して、デバイス切断イベントを生成します。

ビット0 **RWUSIG**：リモートウェイクアップシグナリング

アプリケーションがこのビットをセットすると、コアはリモートシグナリングを実行してUSBホストをウェイクアップします。コアをサスペンド状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。USB 2.0仕様で規定されているように、アプリケーションは、このビットをセットしてから1 msから15 ms後にクリアする必要があります。

LPMが有効で、コアがL1（スリープ）状態にあるとき、アプリケーションがこのビットをセットすると、コアはL1リモートシグナリングを開始して、USBホストをウエイクアップします。コアをスリープ状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。LPM仕様で規定されているように、このビットはアプリケーションによってセットされてから50µs（$T_{L1DevDrvResume}$）後にハードウェアによって自動的クリアされます。アプリケーションは、前のLPMトランザクションからのbRemoteWakeがゼロのときこのビットをセットする必要があります（GLPMCFGレジスタのREMWAKEビットを参照）。

表750 はUSBホストがデバイスの切断を検出するためにソフト切断（SDIS）ビットがセットされなければならない最短時間（デバイスの状態による）を示します。クロックジッタに対応するには、アプリケーションは指定された最短時間まで遅延を追加することが推奨されます。

表750.ソフト切断のための最短時間

動作速度	デバイスの状態	最短時間
フルスピード	サスペンド状態	1ms + 2.5µs
フルスピード	アイドル	2.5µs
フルスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	2.5µs

72.15.35 OTGデバイスステータスレジスタ（OTG_DSTS）

アドレスオフセット：0x808

リセット値：0x0000 0010

このレジスタはUSB関連イベントに関するコアのステータスを示します。デバイス全割込みレジスタ（OTG_DAINT）からの割込み時に読み出されなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS[1:0]		FNSOF[13:8]					
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FNSOF[7:0]								Res.	Res.	Res.	Res.	EERR	ENUMSPD[1:0]		SUSP STS
r	r	r	r	r	r	r	r					r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:22 **DEVLNSTS[1:0]**：デバイスラインステータス

USBデータラインの現在のロジックレベルを示します。

ビット [23]：D+ のロジックレベル

ビット [22]：D- のロジックレベル

ビット21:8 **FNSOF[13:0]**：受信SOFのフレーム番号

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **EERR**：不規則エラー

コアは、このビットをセットして、不規則エラーを報告します。

不規則エラーが発生すると、OTG_FS コントローラはサスペンド状態に移行し、OTG_GINTSTSレジスタの早期サスペンドビット（OTG_GINTSTSのESUsPビット）でアプリケーションへの割込みが生成されます。不規則エラーによる早期サスペンドがアサートされた場合、アプリケーションは、ソフト切断回復を実行するしかありません。

ビット2:1 **ENUMSPD[1:0]**：エニュメレーションされた速度

チャープシーケンスによる速度検出後の OTG_FS コントローラの速度を示します。

11：組み込みFS PHYを使用するフルスピード

その他：予約済み

ビット0 **SUSPSTS**：サスペンドステータス

デバイスモードでは、USB上でサスペンド状態が検出されている限り、このビットがセットされます。USBデータライン上で3msの間アクティビティが1つもなかった場合、コアはサスペンド状態に入ります。コアは、以下の場合にサスペンド状態から抜け出します。

– USBデータライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL レジスタのリモートウェイクアップシグナリングビット（OTG_DCTLのRWUSIGビット）に書込みを行ったとき。

72.15.36 OTGデバイスINエンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）

アドレスオフセット：0x810

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各OTG_DIEPINTxレジスタとともに機能して、INエンドポイントごとに1つの割込みを生成します。このレジスタの対応するビットに書き込むことによってOTG_DIEPINTxレジスタの特定のステータスに対するINエンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	Res.	Res.	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
		rw							rw	rw	rw	rw		rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAKM**：NAK割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット12:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **INEPNEM**：INエンドポイントNAK有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット5 **INEPNMM**：EP不一致付きで受信されたINトークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット4 **ITTXFEMSK**：Tx FIFOエンプティ時に受信されたINトークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **EPDM**：エンドポイント無効割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

72.15.37 OTGデバイスOUTエンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）

アドレスオフセット：0x814

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各OTG_DOEPINTxレジスタとともに機能して、OUTエンドポイントごとに1つの割込みを生成します。このレジスタの対応するビットに書き込むことによってOTG_DOEPINTxレジスタの特定のステータスに対するOUTエンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKMSK	BERRM	Res.	Res.	Res.	OUTPKTERRM	Res.	Res.	STSPHSRXM	OTEPDM	STUPM	Res.	EPDM	XFRCM
		rw	rw				rw			rw	rw	rw		rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAKMSK**：NAK割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット12 **BERRM**：バブルエラー割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **OUTPKTERRM**：OUTパケットエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **STSPHSRXM**：制御書込みに対するステータスフェーズ受信マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット4 **OTEPDM**：エンドポイント無効時OUT トークン受信マスク制御OUTエンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット3 **STUPM**：STUPM：SETUPフェーズ終了マスク制御エンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **EPDM**：エンドポイント無効割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

72.15.38 OTGデバイス全エンドポイント割込みレジスタ（OTG_DAINT）

アドレスオフセット：0x818

リセット値：0x0000 0000

エンドポイント上で重要なイベントが発生すると、OTG_DAINTレジスタは、OTG_GINTSTSレジスタのデバイスOUTエンドポイント割込みビットまたはデバイスINエンドポイント割込みビット（それぞれ、OTG_GINTSTSのOEPINTまたはIEPINTビット）を使用して、アプリケーションに割込みをかけます。エンドポイントごとに1つの割込みビットがあります（OUTエンドポイントおよびINエンドポイントに対してそれぞれ最大16bit）。双方向エンドポイントの場合、対応するINおよびOUT割込みビットが使用されます。このレジスタのビットは、アプリケーションが対応するデバイスエンドポイント-x割込みレジスタのビット（OTG_DIEPINTx/OTG_DOEPINTx）をセット／クリアしたときに、セット／クリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEPINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IEPINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **OEPINT[15:0]**：OUTエンドポイント割込みビット
OUTエンドポイントチャネルあたり 1 ビット：
OUTエンドポイント0に対してビット16、OUTエンドポイント3に対してビット19

ビット15:0 **IEPINT[15:0]**：INエンドポイント割込みビット
INエンドポイントあたり 1 ビット：
INエンドポイント0に対してビット0、INエンドポイント3に対してビット3

72.15.39 OTG全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）

アドレスオフセット：0x81C

リセット値：0x0000 0000

OTG_DAINTMSKレジスタはデバイスエンドポイント割込みレジスタと連動し、デバイスエンドポイントでイベントが発生したときにアプリケーションに割込みを生成します。ただし、その割込みに対応するOTG_DAINTレジスタのビットはセットされたままです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEPM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IEPM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **OEPM[15:0]**：OUT EP割込みマスクビット
OUTエンドポイント（EP）あたり1bit：
OUT EP 0に対してビット16、OUT EP 3に対してビット19
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット15:0 **IEPM[15:0]**：IN EP割込みマスクビット
INエンドポイントあたり 1 ビット：
IN EP 0に対してビット0、IN EP 3に対してビット3
0：割込みはマスクされます。
1：割込みはマスクされません。

72.15.40 OTGデバイスV_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）

アドレスオフセット：0x0828

リセット値：0x0000 17D7

このレジスタはSRP中のV_{BUS} 放電時間（ V_{BUS} パルス後）を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VBUSDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **VBUSDT[15:0]**：デバイスV_{BUS} 放電時間
SRP中のV_{BUS} 放電時間（ V_{BUS} パルス後）を指定します。この値は以下のようになります。
V_{BUS} 放電時間（PHYクロック / 1024）
この値はV_{BUS} の負荷に応じて、調整が必要となる場合があります。

72.15.41 OTGデバイスV_{BUS} パルス時間レジスタ （OTG_DVBUSPULSE）

アドレスオフセット：0x082C

リセット値：0x0000 05B8

このレジスタはSRP期間中のV_{BUS} パルス時間を指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DVBUSP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **DVBUSP[15:0]**：デバイスV_{BUS} パルス時間。この機能はOTG1.3にのみ関係します。
SRP期間中のV_{BUS} パルス時間を指定します。この値は以下のようになります。
V_{BUS} パルス時間（PHYクロック / 1024）

72.15.42 OTGデバイスINエンドポイントFIFOエンプティ割込みマスクレジスタ （OTG_DIEPEMPMSK）

アドレスオフセット：0x834

リセット値：0x0000 0000

このレジスタは、INエンドポイントFIFOエンプティ割込みの生成（TXFE_OTG_DIEPINTx）を制御するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTXFEM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INEPTXFEM[15:0]**：IN EP Tx FIFOエンプティ割込みマスクビット
これらのビットはOTG_DIEPINTxに対するマスクビットとして機能します。
IN EPあたりTXFE割込み1bit :
IN EP 0に対してビット0、IN EP 3に対してビット3
0：割込みはマスクされます。
1：割込みはマスクされません。

72.15.43 OTGデバイス制御INエンドポイント0制御レジスタ（OTG_DIEPCTL0）

アドレスオフセット：0x900

リセット値：0x0000 0000

このセクションでは、USB_OTG FSのOTG_DIEPCTL0レジスタについて説明します。非ゼロ制御エンドポイントは、エンドポイント1～3のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	TXFNUM[3:0]				STALL	Res.	EPTYP[1:0]		NAK STS	Res.
rs	rs			w	w	rw	rw	rw	rw	rs		r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ[1:0]	
r														rw	rw

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイント0でデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

–エンドポイント無効

–転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、このビットをセットして、エンドポイントへの転送が完了する前でも、そのエンドポイントでのデータ送信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、エンドポイント上でSETUPパケットが受信された後、そのエンドポイントに対してこのビットをセットすることもできます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 **TXFNUM[3:0]**：Tx FIFO番号

この値は、INエンドポイント0に割り当てられるFIFO番号に設定されます。

ビット21 **STALL**：STALLハンドシェイク

アプリケーションは、このビットのセットのみが可能です。このエンドポイントのSETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともにNAKビット（グローバルINNAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

制御のために "00" にハードコード化されています。

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

このビットがアプリケーションまたはコアによってセットされると、Tx FIFOに使用可能なデータがある場合でも、コアによってデータの送信が停止されます。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **USBAEP**：USBアクティブエンドポイント

このビットは常に1にセットされ、制御エンドポイント0がすべての設定およびインタフェースにおいて常に有効であることを示します。

ビット14:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **MPSIZ[1:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。

00：64バイト

01：32バイト

10：16バイト

11：8バイト

72.15.44 OTGデバイスINエンドポイントx制御レジスタ（OTG_DIEPCTLx）

アドレスオフセット：0x900 + 0x20 * x（x = 1から5）

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	TXFNUM[3:0]				STALL	Res.	EPTYP[1:0]		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w	rw	rw	rw	rw	rw		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

–SETUPフェーズ終了

–エンドポイント無効

–転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SODDFRM**：奇数フレームの設定

アイソクロナスINおよびOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット28 **SD0PID**：DATA0 PIDの設定

割込み／バルクINエンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA0に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナスINエンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 **TXFNUM[3:0]**：Tx FIFO番号

これらのビットは、このエンドポイントに関連するFIFO番号を指定します。各有効INエンドポイントは、別々のFIFO番号にプログラムされなければなりません。

このフィールドは、INエンドポイントについてのみ有効です。

ビット21 **STALL**：STALLハンドシェイク

非制御の非アイソクロナスINエンドポイントにのみ適用されます（アクセスタイプはrwです）。

アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：制御

01：アイソクロナス

10：バルク

11：割込み

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりデータの送信が停止されます。

アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりゼロ長のデータパケットが送出されます。

このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **EONUM**：偶数／奇数フレーム

アイソクロナスINエンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタのSEVNFRMおよびSODDFRMフィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割込み／バルクINエンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットのPIDを含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットのPIDをプログラムする必要があります。アプリケーションはSC0PIDレジスタを使用して、DATA0またはDATA1 PIDのいずれかをプログラムします。

0：DATA0

1：DATA1

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

72.15.45 OTGデバイスINエンドポイントx割込みレジスタ（OTG_DIEPINTx）

アドレスオフセット：0x908 + 0x20 * x, (x = 0から5)

リセット値：0x0000 0080

このレジスタは、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。図892に示されています。コア割込みレジスタ（OTG_GINTSTSのIEPINT）のINエ ンドポイント割込みビットがセットされている場合、アプリケーションはこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出す前に、最初にデバイス･オール･エンドポイント割込み（OTG_DAINT）レジスタを読み込んで、デバイス･エンドポイント-x割込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAK	Res.	PKTD RPSTS	Res.	Res.	Res.	TXFE	IN EPNE	IN EPNM	ITTXFE	TOC	Res.	EP DISD	XFRC
		rc_w1		rc_w1				r	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAK**：NAK入力

コアは、デバイスがNAKを送受信したとき、この割込みを生成します。アイソクロナスINエンドポイントの場合、Tx FIFOにデータがないためにゼロ長パケットが送信されたとき、
この割込みが生成されます。

ビット12 予約済みであり、リセット値に保持する必要があります。

ビット11 **PKTDRPSTS**：パケットドロップステータス

このビットは、ISOC OUTパケットがドロップされたことをアプリケーションに示します。このビットには
対応するマスクビットがなく、割込みは生成されません。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TXFE**：送信FIFOエンプティ

この割込みは、このエンドポイントのTx FIFOが半分エンプティまたは完全エンプティのいずれかであるとき、アサートされます。半分エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタのTx FIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）によって判断されます。

ビット6 **INEPNE**：INエンドポイントNAKの有効性

このビットは、アプリケーションがOTG_DIEPCTLxレジスタのCNAKビットに書き込むことによってINエンドポイントNAKをクリアするとクリアできます。
この割込みは、コアが（アプリケーションまたはコアによって）設定されたNAKビットをサンプリングしたことを示します。この割込みは、アプリケーションによってセットされたINエンドポイントNAKビットがコアで有効になったことを示します。
この割込みは、NAKハンドシェイクがUSBで送出されることを保証するものではありません。STALLビットがNAKビットに優先します。

ビット5 **INEPNM**：EP不一致で受信されたINトークン

非周期的TxFIFOの最上位のデータが、INトークンが受信されたエンドポイント以外のエンドポイントに属していることを示します。この割込みは、INトークンが受信されたエンドポイント上でアサートされます。

ビット4 **ITTXFE**：Tx FIFOがエンプティ時にINトークンを受信

関連するTx FIFO（周期的/非周期的）が空のときにINトークンが受信されたことを示します。この割込みは、INトークンが受信されたエンドポイント上でアサートされます。

ビット3 **TOC**：タイムアウト条件

このエンドポイント上の最後のINトークンに対して、コアがUSB上でタイムアウト条件を検出したことを示します。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **EPDISD**：エンドポイント無効割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送がAHBに加えてUSB上でも完了したことを示します。

72.15.46 OTGデバイスINエンドポイント0転送サイズレジスタ（OTG_DIEPTSIZ0）

アドレスオフセット：0x910

リセット値：0x0000 0000

アプリケーションは、エンドポイント0を有効にする前に、このレジスタを変更する必要があります。デバイス制御エンドポイント0制御レジスタのエンドポイントイネーブルビット（OTG_DIEPCTL0のEPENA）を使用して、エンドポイント0が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント1～3のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT[1:0]		Res.	Res.	Res.
											rw	rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:19 **PKTCNT [1:0]**：パケットカウント

エンドポイント0のデータ転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がTxFIFOから読み出されるたびにデクリメントされます。

ビット18:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント0の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットがTxFIFOに書き込まれるたびに、このフィールドをデクリメントします。

72.15.47 OTGデバイスINエンドポイント送信FIFOステータスレジスタ（OTG_DTXFSTSx）

アドレスオフセット：0x918 + 0x20 * x, (x = 0から5)

リセット値：0x0000 0200

この読出し専用レジスタは、デバイスINエンドポイントTxFIFOの空きスペース情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTFSAV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INEPTFSAV[15:0]**：INエンドポイントTx FIFOの使用可能スペース

エンドポイントのTx FIFOで使用できる空きスペースの量を示します。

値は32bitワード単位です。

0x0：エンドポイントTx FIFOはフルです。

0x1：1ワードが使用可能です。

0X2：2ワードが使用可能です。

0xn：nワードが使用可能です。

その他：予約済み

72.15.48 OTGデバイスINエンドポイントx転送サイズレジスタ（OTG_DIEPTSIZx）

アドレスオフセット：0x910 + 0x20 * x, (x = 1から5)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DIEPCTLxレジスタのエンドポイントイネーブルビット（OTG_DIEPCTLxのEPENAビット）を使用してエンドポイントが有効になると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCNT[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **MCNT[1:0]**：マルチカウント

周期的INエンドポイントの場合、このフィールドは、USBでフレームごとに送信されなければならないパケット数を示します。コアは、このフィールドを使用して、アイソクロナスINエンドポイントのデータPIDを計算します。

01：1パケット

10：2パケット

11：3パケット

ビット28:19 **PKTCNT [9:0]**：パケットカウント

このエンドポイントのデータの転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がTxFIFOから読み出されるたびにデクリメントされます。

ビット18:0 **XFRSIZ[18:0]**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットがTxFIFOに書き込まれるたびに、このフィールドをデクリメントします。

72.15.49 OTGデバイス制御OUTエンドポイント0制御レジスタ（OTG_DOEPCTL0）

アドレスオフセット：0xB00

リセット値：0x0000 8000

このセクションでは、OTG_DOEPCTL0 レジスタについて説明します。非ゼロ制御エンドポイントは、エンドポイント1～3のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	Res.
w	r			w	w					rs	rw	r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ[1:0]	
r														r	r

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイント0上でデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

–SETUPフェーズ終了

–エンドポイント無効

–転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、制御OUTエンドポイント0を無効にできません。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、送信完了割込み時、またはエンドポイントでSETUPを受信した後に、このビットをセットすることもできます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **STALL**：STALLハンドシェイク

アプリケーションは、このビットのセットのみ可能です。このエンドポイントのSETUPトークンが受信されると、コアがこのビットをクリアします。NAKビット、すなわち、グローバルOUT NAKがこのビットとともにセットされた場合は、STALLビットが優先します。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアはOUTパケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

制御のために2'b00にハードコード化されています。

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、Rx FIFOに受信パケットを収容できるスペースがある場合でも、コアはデータの受信を停止します。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **USBAEP**：USBアクティブエンドポイント

このビットは常に1にセットされ、制御エンドポイント0がすべての設定およびインタフェースにおいて常にアクティブであることを示します。

ビット14:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **MPSIZ[1:0]**：最大パケットサイズ

制御OUTエンドポイント0の最大パケットサイズは、制御INエンドポイント0でプログラムされるサイズと同じです。

00：64バイト

01：32バイト

10：16バイト

11：8バイト

72.15.50 OTGデバイスOUTエンドポイントx割込みレジスタ（OTG_DOEPINTx）

アドレスオフセット：0xB08 + 0x20 * x, (x = 0から5)

リセット値：0x0000 0080

このレジスタは、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。図892に示されています。アプリケーションはOTG_GINTSTSレジスタのOUTエンドポイント割込みビット(OTG_GINTSTSのOEPINTビット)がセットされている場合にこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出すためには、まず、OTG_DAINTレジスタを読み出して、OTG_DOEPINTx レジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAK	BERR	Res.	Res.	Res.	Res.	Res.	Res.	STSPHSRX	OTEPDIS	STUP	Res.	EPDISD	XFRC
		rc_w1	rc_w1							rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAK**：NAK入力

コアは、デバイスがNAKを送受信したとき、この割込みを生成します。アイソクロナスINエンドポイントの場合、Tx FIFOにデータがないためにゼロ長パケットが送信されたとき、

この割込みが生成されます。

ビット12 **BERR**：バブルエラー割込み

コアは、エンドポイントに対してバブルが受信されたとき、この割込みを生成します。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **STSPHSRX**：制御書込みに対して受信されたステータスフェーズ

この割込みは、制御OUTエンドポイントについてのみ有効です。ホストが制御書込みのデータフェーズの転送中に送信したすべてのデータを OTG_FS がシステムメモリバッファに転送した後にのみ、この割込みが生成されます。この割込みによって、ホストが制御書込み転送のデータフェーズからステータスフェーズに切り換えたことがアプリケーションに知らされます。アプリケーションはこの割込みを使って、データフェーズをデコードした後、ステータスフェーズにACKまたはSTALLを返すことができます。

ビット4 **OTEPDIS**：エンドポイント無効時OUTトークン受信

制御OUTエンドポイントにのみ適用されます。

エンドポイントがまだ有効になっていないときにOUTトークンを受け取ったことを示します。この割込みは、OUTトークンが受信されたエンドポイントでアサートされます。

ビット3 **STUP**：SETUPフェーズの完了

制御OUTエンドポイントにのみ適用されます。制御エンドポイントのSETUPフェーズが完了し、現在の制御転送のために、これ以上、連続SETUPパケットは受信されなかったことを示します。この割込みにより、アプリケーションは、受信したSETUPデータパケットをデコードできます。

ビット2 予約済みであり、リセット値に保持する必要があります。

ビット1 **EPDISD**：エンドポイント無効割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送がAHBに加えてUSB上でも完了したことを示します。

72.15.51 OTGデバイスOUTエンドポイント0転送サイズレジスタ（OTG_DOEPTSIZ0）

アドレスオフセット：0xB10

リセット値：0x0000 0000

アプリケーションは、エンドポイント0を有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTL0レジスタのエンドポイントイネーブルビット（OTG_DOEPCTL0のEPENAビット）を使用してエンドポイント0が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント1～5 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	STUPCNT[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.
	rw	rw										rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **STUPCNT[1:0]**：SETUPパケットカウント

このフィールドは、エンドポイントが受信できる連続SETUPデータパケット数を指定します。

01：1パケット

10：2パケット

11：3パケット

ビット28:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PKTCNT**：パケットカウント

このフィールドは、パケットがRx FIFOに書き込まれた後、ゼロまでデクリメントされます。

ビット18:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント0の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFOからパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

72.15.52 OTGデバイスOUTエンドポイントx制御レジスタ（OTG_DOEPCTLx）

アドレスオフセット：0xB00 + 0x20 * x, (x = 1から5)

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID/ SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w					rw	rw	rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

INおよびOUTエンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

–SETUPフェーズ終了

–エンドポイント無効

–転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SD1PID**：DATA1 PIDの設定

割込み／バルクINおよびOUTエンドポイントにのみ適用されます。このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA1に設定されます。

SODDFRM：奇数フレームの設定

アイソクロナスINおよびOUTエンドポイントにのみ適用されます。このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット28 **SD0PID**：DATA0 PIDの設定

割込み／バルクOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA0に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナスOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **STALL**：STALLハンドシェイク

非制御の非アイソクロナスOUTエンドポイントにのみ適用されます（アクセスタイプはrwです）。

アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

制御エンドポイントにのみ適用されます（アクセスタイプはrsです）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントのSETUPトークンが受信されると、コアがこのビットをクリアします。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアはOUTパケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：制御

01：アイソクロナス

10：バルク

11：割込み

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

コアは、Rx FIFOに受信パケットを収容するためのスペースがある場合でも、OUTエンドポイントでのデータの受信を停止します。

このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **EONUM**：偶数／奇数フレーム

アイソクロナスINおよびOUTエンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタのSEVNFRMおよびSODDFRMフィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割込み／バルクOUTエンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットのPIDを含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットのPIDをプログラムする必要があります。アプリケーションはSC0PIDレジスタを使用して、DATA0またはDATA1 PIDのいずれかをプログラムします。

0：DATA0

1：DATA1

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

72.15.53 OTGデバイスOUTエンドポイントx転送サイズレジスタ（OTG_DOEPTSIZx）

アドレスオフセット：0xB10 + 0x20 * x, (x = 1から5)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTLxレジスタのエンドポイントイネーブルビット（OTG_DOEPCTLxのEPENAビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RXDPID/ STUPCNT[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **RXDPID[1:0]**：受信データPID

アイソクロナスOUTエンドポイントにのみ適用されます。

これは、このエンドポイントの最後のパケットで受信されたデータのPIDです。

00：DATA0

10：DATA1

STUPCNT[1:0]：SETUPパケットカウント

制御OUTエンドポイントにのみ適用されます。

このフィールドは、エンドポイントが受信できる連続SETUPデータパケット数を指定します。

01：1パケット

10：2パケット

11：3パケット

ビット28:19 **PKTCNT [9:0]**：パケットカウント

このエンドポイントのデータの転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がRx FIFOに書き込まれるたびにデクリメントされます。

ビット18:0 **XFRSIZ[18:0]**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFOからパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

72.15.54 OTGパワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

アドレスオフセット：0xE00

リセット値：0x200B 8000

このレジスタは、ホストモードとデバイスモードで使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHY SLEEP	ENL1 GTG	PHY SUSP	Res.	Res.	GATE HCLK	STPP CLK
								r	r	rw	r			rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **SUSP**：ディープスリープ

このビットは、L1状態時にPHYがディープスリープであることを示します。

ビット6 **PHYSLEEP**：PHYがスリープ状態

このビットは、PHYがスリープ状態であることを示します。

ビット5 **ENL1GTG**：スリープクロックゲーティングの有効化

このビットがセットされている場合、コアがutmi_l1_suspend_nをアサートできなければ、スリープ状態でコアの内部クロックゲーティングが有効化されます。このビットがセットされていない場合、PHYクロックはスリープ状態ではゲートされません。

ビット4 **PHYSUSP**：PHYのサスペンド

PHYがサスペンドされたことを示します。このビットは、アプリケーションがSTPPCLKビットをセットした後、PHYがサスペンドされると更新されます。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **GATEHCLK**：HCLKのゲーティング

USBがサスペンドされるか、セッションが有効でないとき、アプリケーションは、このビットをセットして、AHBスレーブとマスタおよびウェイクアップロジック以外のモジュールへのHCLKをゲーティングします。アプリケーションは、USBがレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

ビット0 **STPPCLK**：PHYクロックを停止

USBがサスペンドされるか、セッションが有効でないか、デバイスが切断されると、アプリケーションは、このビットをセットして、PHYクロックを停止します。アプリケーションは、USBがレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

72.15.55 OTG_FS レジスタマップ

次の表に、USB OTGのレジスタマップとリセット値を示します。

表751.OTG_FS レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	OTG_GOTGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CURMOD	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS	Res.	Res.	Res.	EHEN	DHNPEN	HSHNPEN	HNPRQ	HNGSCS	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	SRQ	SRQSCS
	リセット値											0	0	0	0	0	1				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	OTG_GOTGINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBCDNE	ADTOCHG	HNGDET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
	リセット値													0	0	0							-	0	0						0		
0x008	OTG_GAHBCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	Res.	Res.	Res.	Res.	Res.	GINTMSK
	リセット値																								0	0							0
0x00C	OTG_GUSBCFG	Res.	FDMOD	FHMOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRDT				HNPCAP	SRPCAP	Res.	PHYSEL	Res.	Res.	Res.	TOCAL			
	リセット値		0	0																0	1	0	1	0	0		1				0	0	0
0x010	OTG_GRSTCTL	AHBIDL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFNUM					TXFFLSH	RXFFLSH	Res.	FCRST	PSRST	CSRST
	リセット値	1																					0	0	0	0	0	0	0	0	0	0	0
0x014	OTG_GINTSTS	WKUPINT	SRQINT	DISCINT	CIDSCHG	LPMINT	PTXFE	HCINT	HPRTINT	RSTDET	Res.	IPXFR/INCOMPISOOUT	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	0	0	1	0	0	0		0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x018	OTG_ GINTMSK	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	Res.	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	
0x01C	OTG_ GRXSTSR （デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_ GRXSTSR （ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	OTG_ GRXSTSP （デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_ GRXSTSP （ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	OTG_ GRXFSIZ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFD															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x028	OTG_ HNPTXFSIZ/ OTG_ DIEPTXF0	NPTXFD/TX0FD																NPTXFSA/TX0FSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x02C	OTG_ HNPTXSTS	Res.	NPTXQTOP							NPTQXSAV									NPTXFSAV														
	リセット値		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0		0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x038	OTG_ GCCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCDEN	BCDEN	PWRDWN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET
	リセット値											0	0	0	0	0	0													X	X	X	X
0x03C	OTG_CID	PRODUCT_ID																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0x054	OTG_ GLPMCFG	Res.	Res.	Res.	ENBESL	LPMR CNTSTS			SNDLPM	LPM RCNT			LPMCHIDX				L1RSMOK	SLPSTS	LPM RSP		L1DSEN	BESLTHRS			L1SSEN	REMWAKE	BESL				LPMACK	LPMEN	
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	OTG_ HPTXFSIZ	PTXFSIZ																PTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x104	OTG_ DIEPTXF1	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x108	**OTG_ DIEPTXF2**	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
. . . .	. . . .	° ° ° ° °																															
0x114	**OTG_ DIEPTXF5**	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0x400	**OTG_ HCFG**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLS PCS	
	リセット値																														0	0	0
0x404	**OTG_ HFIR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLDCTRL	FRIVL															
	リセット値																0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0	0
0x408	**OTG_ HFNUM**	FTREM																FRNUM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x410	**OTG_ HPTXSTS**	PTXQTOP								PTXQSAV								PTXFSAVL															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x414	**OTG_ HAINT**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINT															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x418	**OTG_ HAINTMSK**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINTM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x440	**OTG_ HPRT**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD		PTCTL				PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS
	リセット値														0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x500	**OTG_ HCCHAR0**	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x508	**OTG_ HCINT0**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x508	**OTG_ HCINT0**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x50C	**OTG_ HCINTMSK0**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM		ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x510	OTG_ HCTSIZ0	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x520	OTG_ HCCHAR1	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.		EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x528	OTG_ HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x528	OTG_ HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x52C	OTG_ HCINTMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM		ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0
0x530	OTG_ HCTSIZ1	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . ..	. . . ..																															
0x660	OTG_ HCCHAR11	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.		EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x668	OTG_ HCINT11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x66C	OTG_ HCINTMSK11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM		ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0
0x670	OTG_ HCTSIZ11	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x800	OTG_ DCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM	XCVRDLY	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値																	0	0		0	0	0	0	0	0	0	0	0		0	0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x804	OTG_ DCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.	Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
	リセット値														0							0	0	0	0	0	0	0	0	0	0	1	0
0x808	OTG_ DSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEV LN STS		FNSOF														Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSPSTS
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x810	OTG_ DIEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	Res.	Res.	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
	リセット値																			0							0	0	0	0		0	0
0x814	OTG_ DOEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKMSK	BERRM	Res.	Res.	Res.	OUTPKTERRM	Res.	Res.	Res.	OTEPDM	STUPM	Res.	EPDM	XFRCM
	リセット値																			0	0				0				0	0		0	0
0x818	OTG_ DAINT	OEPINT																IEPINT															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x81C	OTG_ DAINTMSK	OEPM																IEPM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x828	OTG_ DVBUSDIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBUSDT															
	リセット値																	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1
0x82C	OTG_DVB USPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DVBUSP															
	リセット値																	0	0	0	0	0	1	0	1	1	0	1	1	1	0	0	0
0x834	OTG_DIE PEMPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTXFEM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x900	OTG_ DIEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0	0	0	0	0	0		0	0	0		1														0	0
0x908	OTG_ DIEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	Res.	TXFE	INEPNE	INEPNM	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																			0		0				1	0	0	0	0		0	0
0x910	OTG_ DIEPTSIZ0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値												0	0													0	0	0	0	0	0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x918	OTG_ DTXFSTS0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x920	OTG_ DIEPCTL1	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x928	OTG_ DIEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	Res.	TXFE	INEPNE	INEPNM	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																			0		0				1	0	0	0	0		0	0
0x930	OTG_ DIEPTSIZ1	Res.	MCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x938	OTG_ DTXFSTS1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x940	OTG_ DIEPCTL2	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . ..	. . . ..																															
0x9A0	OTG_ DIEPCTL5	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . ..	. . . ..																															
0x9A8	OTG_ DIEPINT5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	Res.	TXFE	INEPNE	INEPNM	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																			0		0				1	0	0	0	0		0	0
. . . .	. . . ..	. . . ..																															

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x9B8	OTG_ DTXFSTS5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
. . . .	. . . ..	. . . ..																															
0x9B0	OTG_ DIEPTSIZ5	Res.	MCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB00	OTG_ DOEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP	NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0					0	0	0	0	0		1														0	0
0xB08	OTG_ DOEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	BERR	Res.	Res.	Res.	Res.	Res.	Res.	STSPHSRX	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																			0	0							0	0	0		0	0
0xB10	OTG_ DOEPTSIZ0	Res.	STUP CNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ					
	リセット値		0	0										0													0	0	0	0	0	0	0
0xB20	OTG_ DOEPCTL1	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP	NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ											
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0xB28	OTG_ DOEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	BERR	Res.	Res.	Res.	Res.	Res.	Res.	STSPHSRX	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																			0	0							0	0	0		0	0
0xB30	OTG_ DOEPTSIZ1	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . ..	. . . ..																															
0xBA0	OTG_ DOEPCTL5	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP	NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ											
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0

表751.OTG_FS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xBA8	OTG_ DOEPINT5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	BERR	Res.	Res.	Res.	Res.	Res.	Res.	STSPHSRX	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																			0	0					-		0	0	0		0	0
0xBB0	OTG_ DOEPTSIZ5	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE00	OTG_ PCGCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHYSLEEP	ENL1GTG	PHYSUSP	Res.	Res.	GATEHCLK	STPPCLK
	リセット値																									0	0	0	0			0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

72.16 OTG_FS プログラミングモデル

72.16.1 コアの初期化

アプリケーションはコア初期化シーケンスを実行する必要があります。パワーアップ中にケーブルが接続された場合、OTG_GINTSTSレジスタの現在の動作モードビット（OTG_GINTSTSのCMODビット）はモードを反映します。OTG_FS コントローラは、Aプラグが接続されたときにはホストモードに、Bプラグが接続されたときにはデバイスモードに入ります。

このセクションでは、電源投入後の OTG_FS コントローラの初期化について説明します。アプリケーションは、ホストモードかデバイスモードかに関係なく、初期化シーケンスに従う必要があります。コアのすべてのグローバルレジスタは、コアの設定に従って以下の手順で初期化されます。

1. OTG_GAHBCFGレジスタの以下のフィールドをプログラムします。
   - グローバル割込みマスクビットGINTMSK = 1
   - Rx FIFO非エンプティ（OTG_GINTSTSのRXFLVLビット）
   - 周期的Tx FIFOエンプティレベル
2. OTG_GUSBCFGレジスタの以下のフィールドをプログラムします。
   - HNP対応ビット
   - SRP対応ビット
   - OTG_FS タイムアウト較正フィールド
   - USBターンアラウンド時間フィールド
3. ソフトウェアは、OTG_GINTMSKレジスタの以下のビットのマスクを解除する必要があります。

   OTG割込みマスク

   モード不一致割込みマスク
4. ソフトウェアは、OTG_GINTSTSレジスタのCMODビットを読み出すことによって、OTG_FS コントローラがホストモードで動作中か、デバイスモードで動作中かを判別することができます。

72.16.2 ホストの初期化

コアをホストとして初期化するには、アプリケーションは、以下の手順を実行しなければなりません。

1. OTG_GINTMSKレジスタのHPRTINTビットをプログラムして、マスクを解除します。
2. OTG_HCFGレジスタをプログラムして、フルスピードホストを選択します。
3. OTG_HPRTレジスタのPPWRビットを1にプログラムします。これによってUSB上でV_{BUS} が駆動されます。
4. OTG_HPRT0のPCDET割込みを待ちます。これは、デバイスがポートに接続されていることを示します。
5. OTG_HPRTレジスタのPRSTビットを1にプログラムします。これによってリセットプロセスが開始されます。
6. リセットプロセスが完了するまで、少なくとも10 ms待ちます。
7. OTG_HPRTレジスタのPRSTビットを0にプログラムします。
8. OTG_HPRTレジスタのPENCHNG割込みを待ちます。
9. OTG_HPRT レジスタのPSPD ビットを読み出して、エニュメレーションされたスピードを取得します。
10. HFIRレジスタを、選択されたPHYクロック1に対応する値でプログラムします。
11. OTG_HCFGレジスタのFSLSPCSフィールドを、手順9で検出したデバイスの速度に従ってプログラムします。FSLSPCSが変更されている場合は、ポートのリセットを行う必要があります。
12. OTG_GRXFSIZレジスタをプログラムして、受信FIFOのサイズを選択します。
13. OTG_HNPTXFSIZレジスタをプログラムして、非周期的トランザクションの非周期的送信FIFOのサイズと開始アドレスを選択します。
14. OTG_HPTXFSIZレジスタをプログラムして、周期的トランザクションの周期的送信FIFOのサイズと開始アドレスを選択します。

デバイスと通信するには、システムソフトウェアは、少なくとも1つのチャネルを初期化し、有効にする必要があります。

72.16.3 デバイスの初期化

アプリケーションは、パワーアップ時、またはホストモードからデバイスモードへの変更後、以下の手順を実行して、コアをデバイスとして初期化する必要があります。

1. OTG_DCFGレジスタの以下のフィールドをプログラムします。
   - デバイススピード
   - 非ゼロ長ステータスOUTハンドシェイク
   - 周期的フレームインターバル
2. DCTL.SDISビットをクリアします。コアはこのビットがクリアされた後に接続を発行します。
3. OTG_GINTMSKレジスタをプログラムして、次の割込みのマスクを解除します。
   - USBリセット
   - エニュメレーション終了
   - 早期サスペンド
   - USBサスペンド
   - SOF
4. OTG_GINTSTS の USBRST 割込みを待ちます。これは、この割込みの受信時、約 10ms 続くリセットがUSBで検出されたことを示します。
5. OTG_GINTSTSのENUMDNE割込みを待ちます。この割込みは、USB上でリセットが終了したことを示します。この割込みを受信したとき、アプリケーションは、OTG_DSTSレジスタを読み出して、エニュメレーション速度を判別し、にリストされている手順を実行する必要があります。3223ページのエニュメレーション完了時のエンドポイント初期化

この時点で、デバイスは、SOFパケットを受け入れて、制御エンドポイント0で制御転送を実行する準備ができたことになります。

72.16.4 ホストプログラミングモデル

チャネルの初期化

アプリケーションが接続されたデバイスと通信するためには、1つまたは複数のチャネルを初期化する必要があります。チャネルを初期化して、有効にするには、アプリケーションは、以下の手順を実行する必要があります。

1. OTG_GINTMSKレジスタをプログラムして、以下の割込みのマスクを解除します。
2. チャネル割込み
   - OUTトランザクションの非周期的送信FIFOエンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
   - OUT トランザクションの非周期的送信 FIFO 半分エンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
3. OTG_HAINTMSKレジスタをプログラムして、選択されたチャネルの割込みのマスクを解除します。
4. OTG_HCINTMSKレジスタをプログラムして、ホストチャネル割込みレジスタで指定された、所定のトランザクション関連割込みのマスクを解除します。
5. 選択されたチャネルのOTG_HCTSIZxレジスタを、合計転送サイズ（バイト数）およびショートパケットを含む予想されるパケット数でプログラムします。アプリケーションは、PIDフィールドを初期データPID（最初のOUTトランザクションで使用されるPID、または最初のINトランザクションから期待されるPID）でプログラムする必要があります。

6. 選択されたチャネルのOTG_HCCHARxレジスタを、タイプ、スピード、方向など、デバイスのエンドポイントの特性でプログラムします。（チャネルは、アプリケーションがパケットを送信または受信する準備ができている場合にのみ、チャネルイネーブルビットを1にセットすることで有効にできます）。

チャネルの停止

アプリケーションは、OTG_HCCHARxレジスタのCHDISおよびCHENAビットを1にセットすることによって、任意のチャネルを無効にできます。これにより、OTG_FS ホストは、ポストされたリクエスト（ある場合）をFlashして、チャネル停止割込みを生成できます。アプリケーションは、チャネルを他のトランザクションに再割り当てする前に、OTG_HCINTx内のCHH割込みを待つ必要があります。OTG_FS ホストは、USB上ですでに転送が開始しているトランザクションに対しては割込みを行いません。

チャネルを無効にする前に、アプリケーションは、非周期的リクエストキュー（非周期的チャネルを無効にするとき）、または周期的リクエストキュー（周期的チャネルを無効にするとき）に使用可能な空きスペースが少なくとも1つ以上存在することを確認する必要があります。アプリケーションは、（チャネルを無効化する前に）リクエストキューがフル状態のときには、OTG_HCCHARxレジスタのCHDISビットに1をセットし、CHENAビットを0にクリアすることで、ポストされたリクエストを一掃できます。

アプリケーションは、以下のいずれかの条件のとき、チャネルを無効にすることが期待されます。

1. INまたはOUTチャネルで、OTG_HCINTxレジスタのSTALL、TXERR、BBERR、またはDTERR割込みが受信されたとき。アプリケーションは、停止割込みを受信する前に、同じチャネルについての他の割込み（DTERR、NAK、データ、TxERR）を受信できなければなりません。
2. OTG_GINTSTSレジスタのDISCINT（デバイス切断）割込みを受信したとき。（アプリケーションは、すべての有効なチャネルを無効にすることが期待されます）。
3. アプリケーションが正常完了の前に転送を中断したとき。

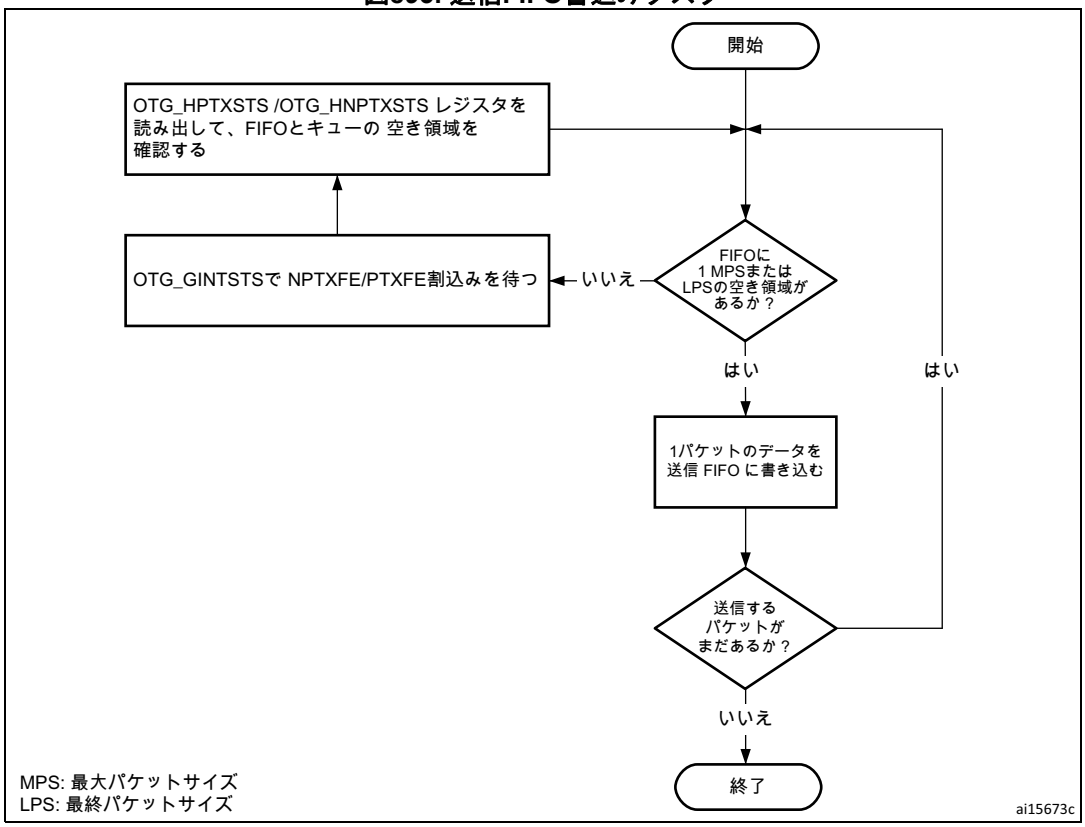
動作モデル

アプリケーションが接続されたデバイスと通信するためには、チャネルを初期化する必要があります。このセクションでは、さまざまなタイプのUSBトランザクションについて、実行しなければならない動作のシーケンスについて説明します。

- **送信FIFOへの書込み**

  OTG_FS ホストは、パケットの最後の32-bitワードの書込みとともに、周期的/非周期的リクエストキューにエントリ（OUTリクエスト）を自動的に書き込みます。アプリケーションは、送信FIFOへの書込みを開始する前に、周期的/非周期的リクエストキューに少なくとも1つ以上の空きスペースがあることを確認しなければなりません。アプリケーションは送信FIFOに必ず32-bitワード単位で書込みを行う必要があります。パケットサイズが32-bitワード単位で揃えられていない場合、アプリケーションはパディングを使用する必要があります。OTG_FS ホストは、プログラムされた最大パケットサイズと転送サイズに基づいて、実際のパケットサイズを決定します。

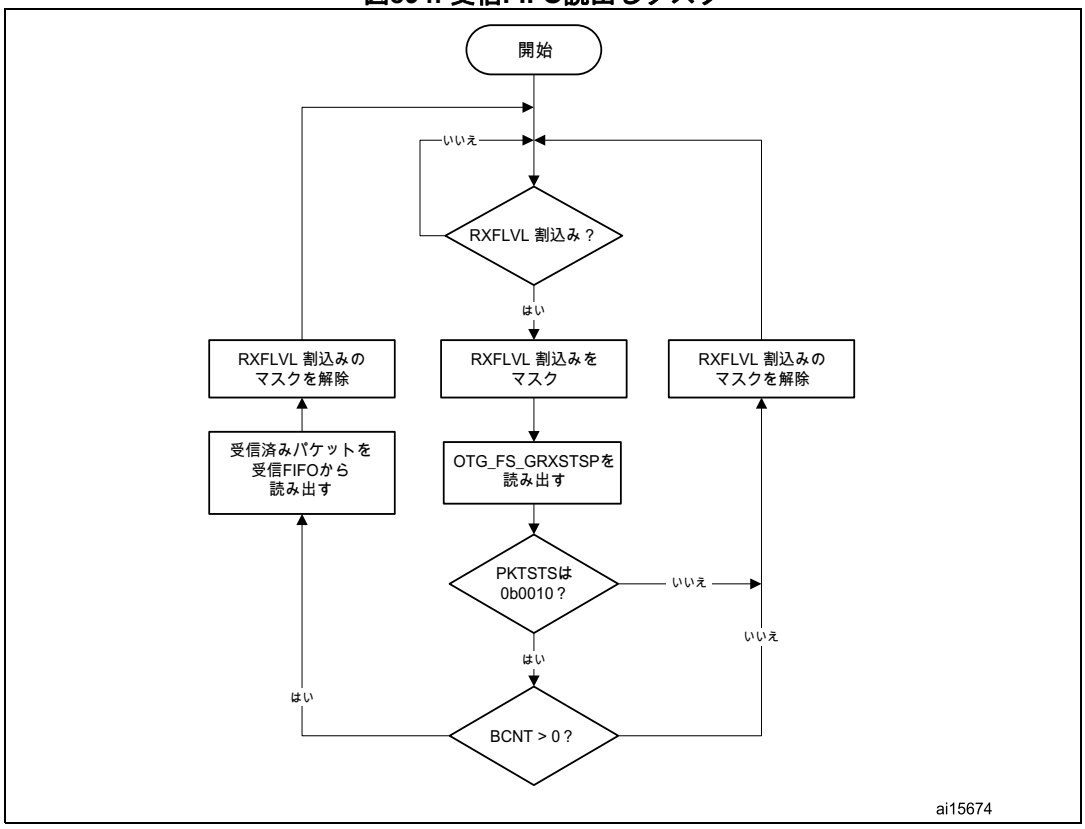
図893. 送信FIFO書込みタスク



- **受信FIFOの読出し**

  アプリケーションは、INデータパケット（bx0010）以外のすべてのパケットステータスを無視しなければなりません。

図894. 受信FIFO読出しタスク



- **バルクおよび制御OUT／SETUPトランザクション**

  典型的なバルクまたは制御OUT/SETUPパイプライントランザクションレベルの動作を図895に示します。チャネル1（ch_1）に注目してください。2つのバルクOUTパケットが転送されます。制御SETUPトランザクションは同じように動作しますが、1パケットのみです。ここでは、以下のように仮定されています。

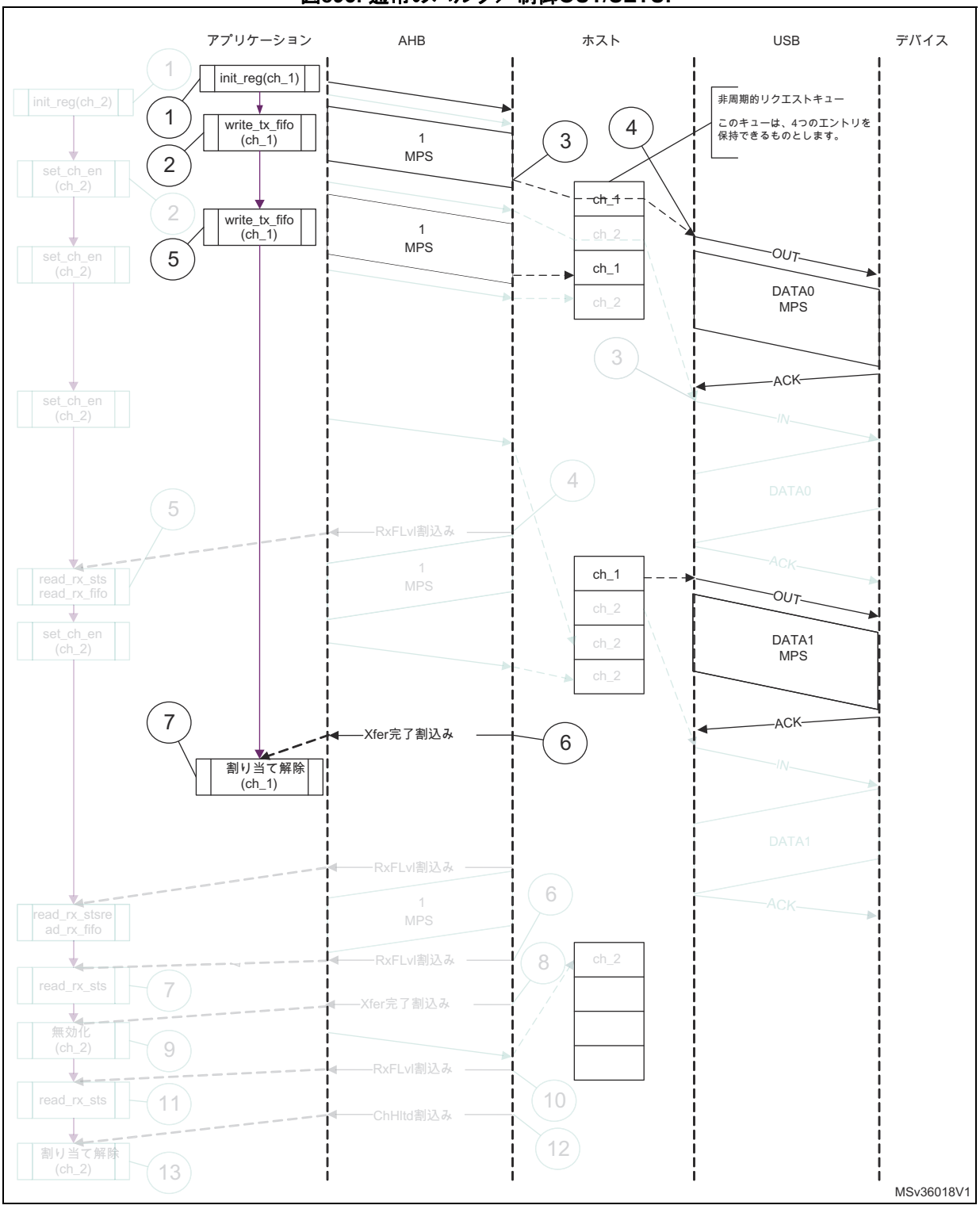
  – アプリケーションは、最大パケットサイズの2つのパケットを送信しようとしています（転送サイズ = 1,024バイト）。
  – 非周期的送信FIFOは2つのパケットを保持することができます（FSの場合128バイト）。
  – 非周期的リクエストキューの深さ = 4。

- **通常のバルクおよび制御OUT/SETUP動作**

  動作シーケンス（チャネル1）は、次のとおりです。

1. チャネル1を初期化します。
2. チャネル1の最初のパケットを書き込みます。
3. 最後のワードの書込みと同時に、コアは、非周期的リクエストキューにエントリを書き込みます。
4. 非周期的キューがエンプティでなくなるとすぐに、コアは、現在のフレームで OUT トークンの送信を試みます。
5. チャネル1の2番目（最後）のパケットを書き込みます。
6. コアは、最後のトランザクションが正常に完了すると、XFRC割込みを生成します。
7. XFRC割込みに応答して、他の転送のためにチャネル割り当てを解除します。
8. 非ACK応答の取り扱い

図895. 通常のバルク／制御OUT/SETUP



1. 灰色の要素は、この図のコンテキストでは無関係です。

バルクおよび制御OUT/SETUPトランザクションのためのチャネル固有の割込みサービスルーチンを、以下のコードサンプルで示します。

- **バルク／制御OUT/SETUPとバルクまたは制御INトランザクションの割込みサービスルーチン**

  a) バルク／制御OUT/SETUP

```
Unmask (NAK/TXERR/STALL/XFRC)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else if (STALL)
   {
   Transfer Done = 1
   Unmask CHH
   Disable Channel
   }
else if (NAK or TXERR )
   {
   Rewind Buffer Pointers
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   else
      {
      Reset Error Count
      }
      }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
   else if (ACK)
```

```
      {
      Reset Error Count
      Mask ACK
      }
```

アプリケーションは、送信FIFOおよびリクエストキューに空きがある場合、データパケットを送信FIFOに書き込むことが期待されます。アプリケーションは、OTG_GINTSTSレジスタのNPTXFE割込みを使用して、送信FIFOのスペースを見つけることができます。

b) バルク／制御IN

```
Unmask (TXERR/XFRC/BBERR/STALL/DTERR)
if (XFRC)
   {
   Reset Error Count
   Unmask CHH
   Disable Channel
   Reset Error Count
   Mask ACK
   }
else if (TXERR or BBERR or STALL)
   {
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
else if (ACK)
   {
   Reset Error Count
   Mask ACK
   }
else if (DTERR)
```

```
{
Reset Error Count
}
```

アプリケーションは、リクエストキューのスペースに空きがあるときは、XFRC割込みが受信されるまで、リクエストを書き込むことが期待されます。

- **バルクおよび制御INトランザクション**

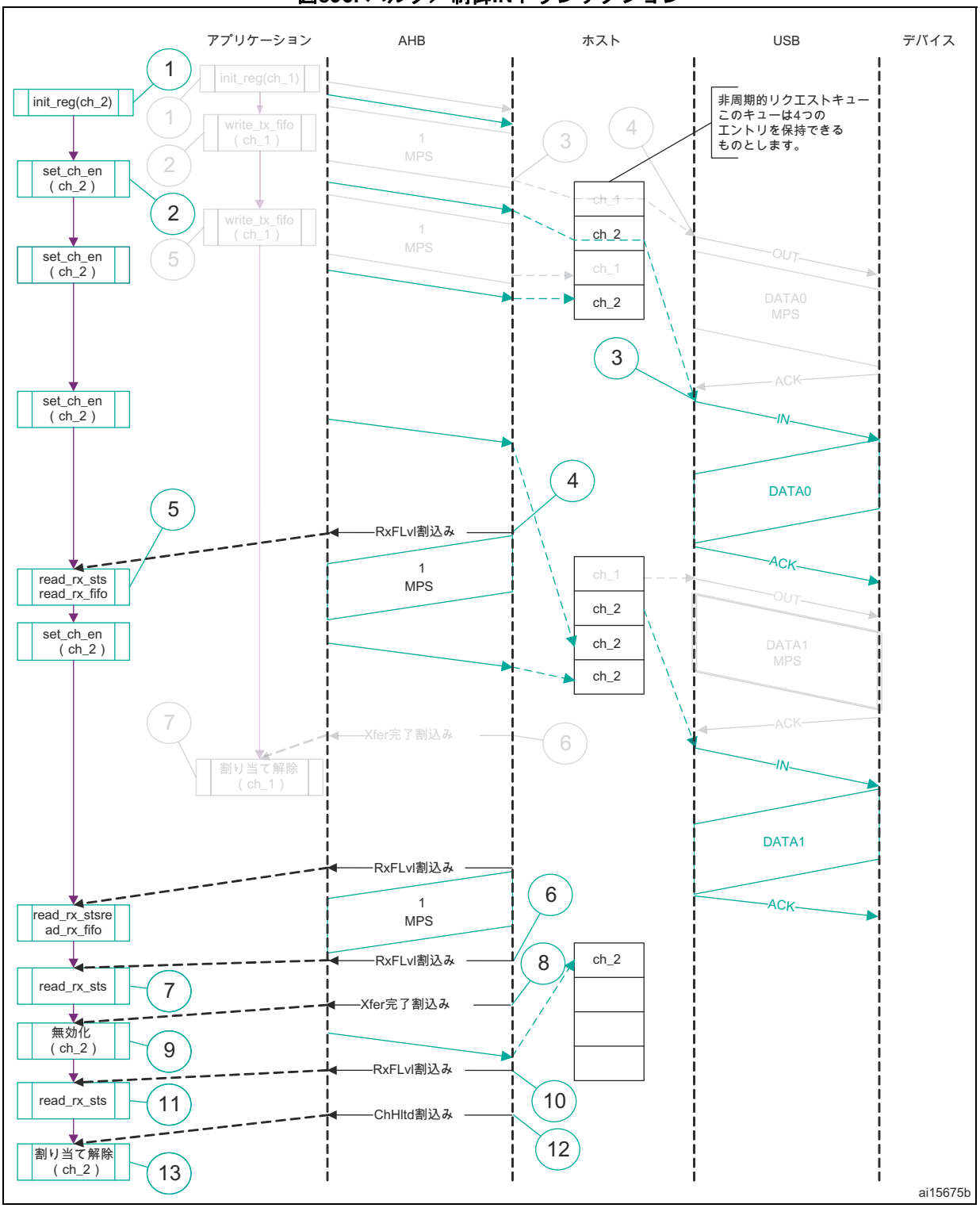
  典型的なバルクまたは制御INパイプライントランザクションレベルの動作を図896に示します。チャネル2（ch_2）に注目してください。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズ（転送サイズ = 1,024バイト）の2つのパケットを受信しようとしています。

  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットと、パケットあたり2つのステータスワード（FSの場合72バイト）を収容することができます。

  – 非周期的リクエストキューの深さ = 4。

図896. バルク／制御INトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。
2. OTG_HCCHAR2のCHENAビットをセットして、INリクエストを非周期的リクエストキューに書き込みます。
3. コアは、現在のOUTトランザクションの完了後、INトークンの送信を試みます。
4. コアは、受信したパケットが受信FIFOに書き込まれると、RXFLVL割込みを生成します。
5. RXFLVL割込みに応答して、RXFLVL割込みをマスクし、受信パケットステータスを読み出して受信バイト数を調べた後、それに応じて受信FIFOを読み出します。続いて、RXFLVL割込みのマスクを解除します。
6. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。
7. アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスがINデータパケットでないとき（OTG_GRXSTSR ≠ 0b0010のPKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると、XFRC割込みを生成します。
9. XFRC割込みに応答して、チャネルを無効にし、それ以降のリクエストについて、OTG_HCCHAR2レジスタへの書込みを停止します。コアは、OTG_HCCHAR2に書き込まれると、非周期的リクエストキューにチャネルディセーブルリクエストを書き込みます。
10. コアは、停止ステータスが受信FIFOに書き込まれると、RXFLVL割込みを生成します。
11. 受信パケットステータスを読み出して、それを無視します。
12. コアは、停止ステータスが受信FIFOからポップされるとすぐに、CHH割込みを生成します。
13. CHH割込みに応答して、他の転送のためにチャネル割り当てを解除します。
14. 非ACK応答の取り扱い

- **制御トランザクション**

  制御転送のセットアップ、データ、およびステータスステージは3つの個別の転送として実行されなければなりません。セットアップ、データ、およびステータスステージのOUTトランザクションは、すでに説明したバルクOUTトランザクションと同様に実行されます。データまたはステータスステージのINトランザクションは、すでに説明したバルクINトランザクションと同様に実行されます。3つのステージすべてについて、アプリケーションはOTG_HCCHAR1レジスタのEPTYPフィールドを制御に設定することが期待されます。セットアップステージでは、アプリケーションはOTG_HCTSIZ1レジスタのPIDフィールドをSETUPに設定することが期待されます。

- **割込みOUTトランザクション**

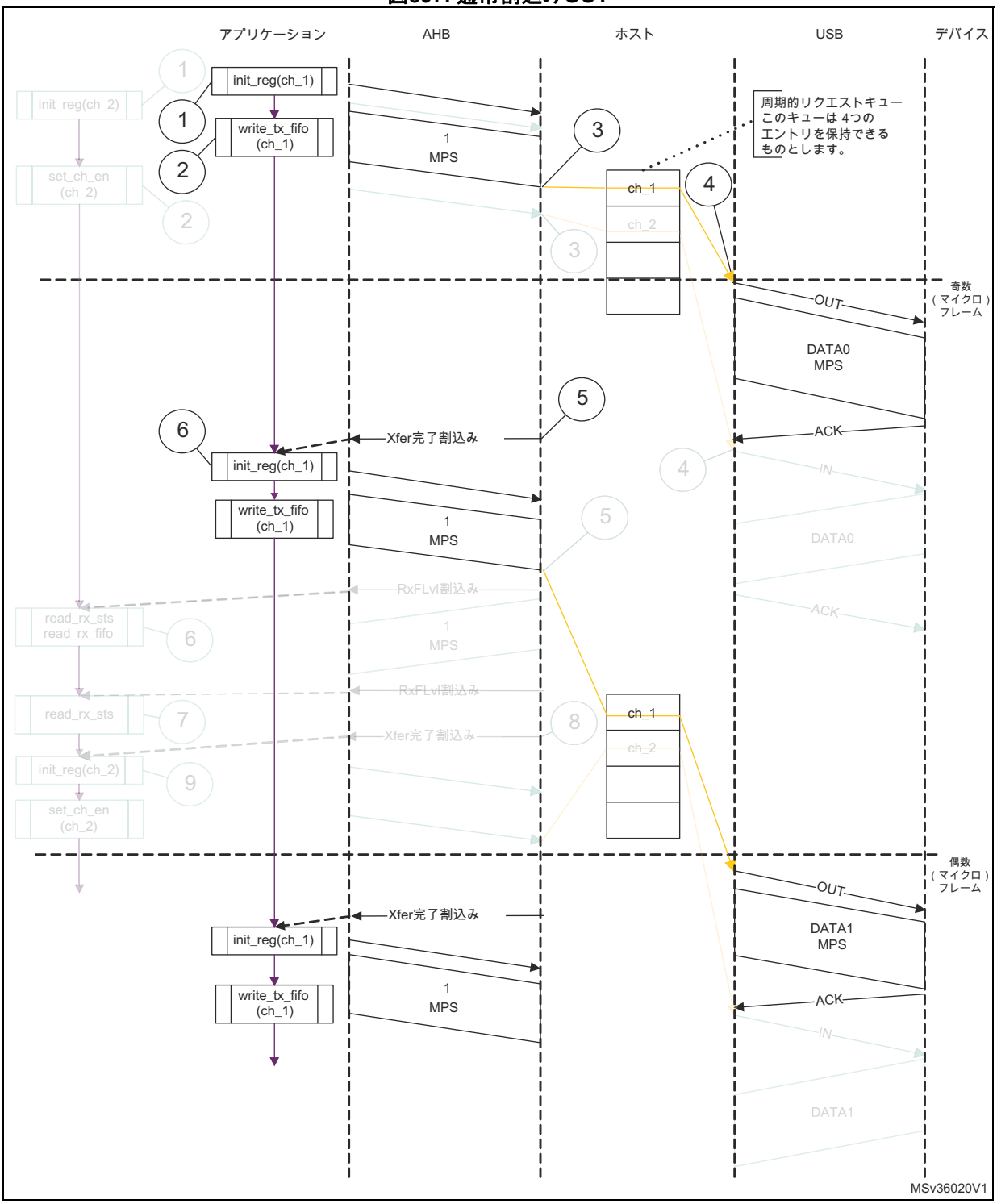
  典型的な割込みOUT動作を図897に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1 024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の送信を試みています。
  – 周期的送信FIFOは1パケット（1KB）を保持することができます。
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル1を初期化し、有効にします。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。
2. チャネル1の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_FS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_FS ホストは、次の（奇数）フレームでOUTトークンの送信を試みます。
5. OTG_FS ホストは、最後のパケットが正常に送信されると、XFRC割込みを生成します。
6. XFRC割込みに応答して、次の転送のためにチャネルを再初期化します。

図897. 通常割込みOUT



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **割込みOUT/INトランザクションのための割込みサービスルーチン**

  a) 割込みOUT

```
Unmask (NAK/TXERR/STALL/XFRC/FRMOR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else
   if (STALL or FRMOR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL)
         {
         Transfer Done = 1
         }
      }
   else
      if (NAK or TXERR)
         {
         Rewind Buffer Pointers
         Reset Error Count
         Mask ACK
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
         {
         Mask CHH
         if (Transfer Done or (Error_count == 3))
            {
            De-allocate Channel
            }
         else
            {
            Re-initialize Channel (in next b_interval - 1 Frame)
            }
         }
      else
         if (ACK)
         {
```

```
        Reset Error Count
        Mask ACK
        }
```

アプリケーションは OTG_GINTSTS レジスタの NPTXFE 割込みを使用して、送信 FIFO のスペースを見つけます。

割込み IN

```
Unmask (NAK/TXERR/XFRC/BBERR/STALL/FRMOR/DTERR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   if (OTG_HCTSIZx.PKTCNT == 0)
      {
      De-allocate Channel
      }
   else
      {
      Transfer Done = 1
      Unmask CHH
      Disable Channel
      }
   }
else
   if (STALL or FRMOR or NAK or DTERR or BBERR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL or BBERR)
         {
         Reset Error Count
         Transfer Done = 1
         }
      else
         if (!FRMOR)
         {
         Reset Error Count
         }
      }
else
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      Unmask CHH
```

```
      Disable Channel
      }
else
   if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
         {
         De-allocate Channel
         }
      else
         Re-initialize Channel (in next b_interval - 1 /Frame)
      }
   }
else
   if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

- **割込みINトランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の受信を試みています。

  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットとパケットあたり2つのステータスワード（1,031バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

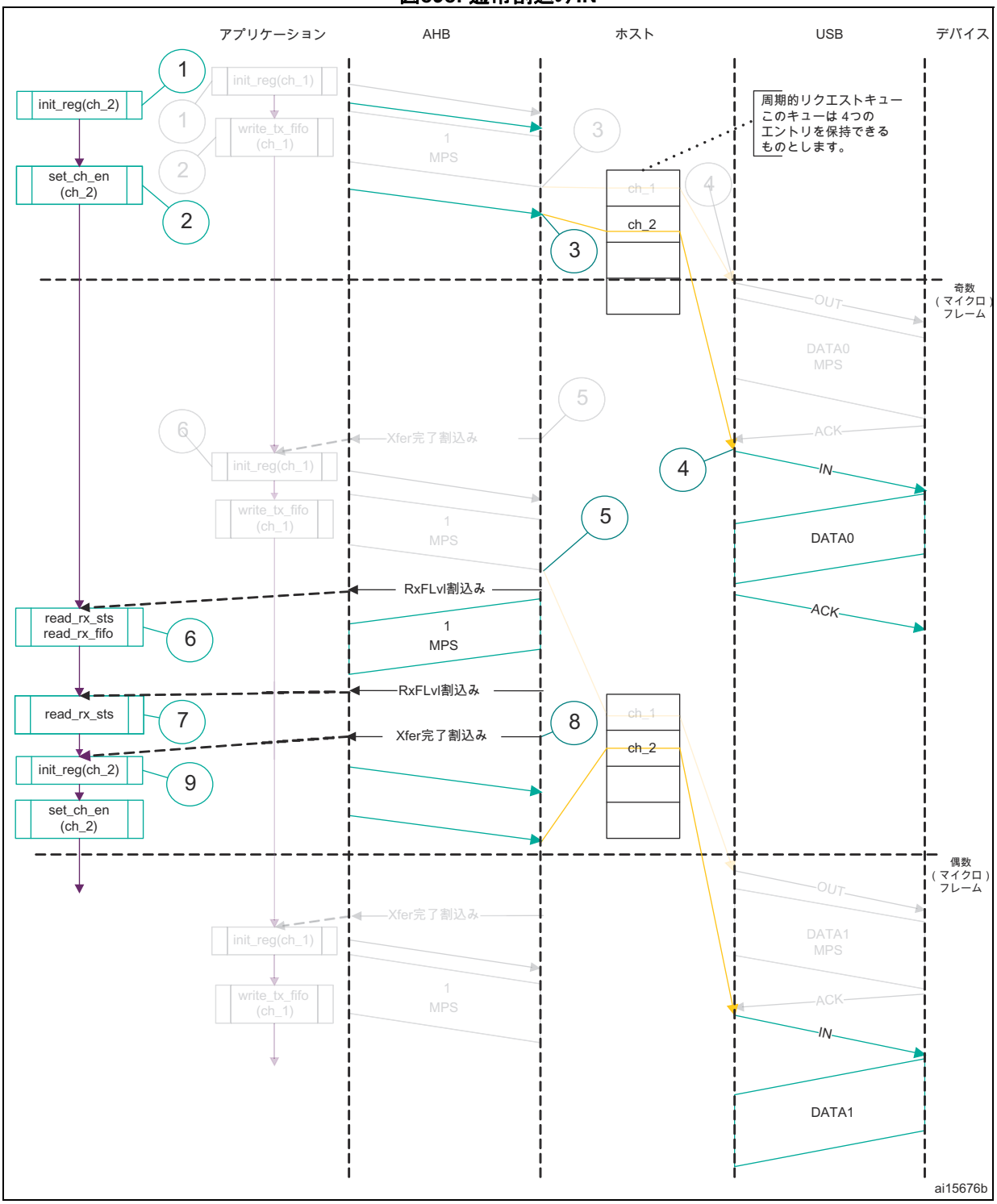
- **通常の割込みIN動作**

  動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをセットする必要があります。
2. OTG_HCCHAR2のCHENAビットをセットしてINリクエストを周期的リクエストキューを書き込みます。
3. OTG_FS ホストは、CHENAビットがセットされたOTG_HCCHAR2レジスタの書込みごとに、周期的リクエストキューにINリクエストを書き込みます。
4. OTG_FS ホストは、次の（奇数）フレームでINトークンの送信を試みます。
5. INパケットが受信され、受信FIFOに書き込まれると、OTG_FS ホストはRXFLVL割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信FIFOを読み出します。アプリケーションは、受信FIFOを読み出す前にRXFLVL割込みをマスクし、すべてのパケットを読み出した後にマスクを解除しなければなりません。
7. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスがINデータパケットでない（GRXSTSRのPKTSTSが0b0010でない）ときには無視する必要があります。

8. コアは、受信パケットステータスが読み出されるとXFRC割込みを生成します。
9. XFRC割込みに応答してOTG_HCTSIZ2のPKTCNTフィールドを読み出します。OTG_HCTSIZ2のPKTCNTビットが 0でない場合は、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2のPKTCNTビットが0の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをリセットする必要があります。

図898. 通常割込みIN



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナスOUTトランザクション**

  典型的なアイソクロナスOUT動作を図899に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレームから始めて、フレームごとに1パケット（最大1パケットサイズ）の送信を試みています。（転送サイズ = 1,024バイト）。

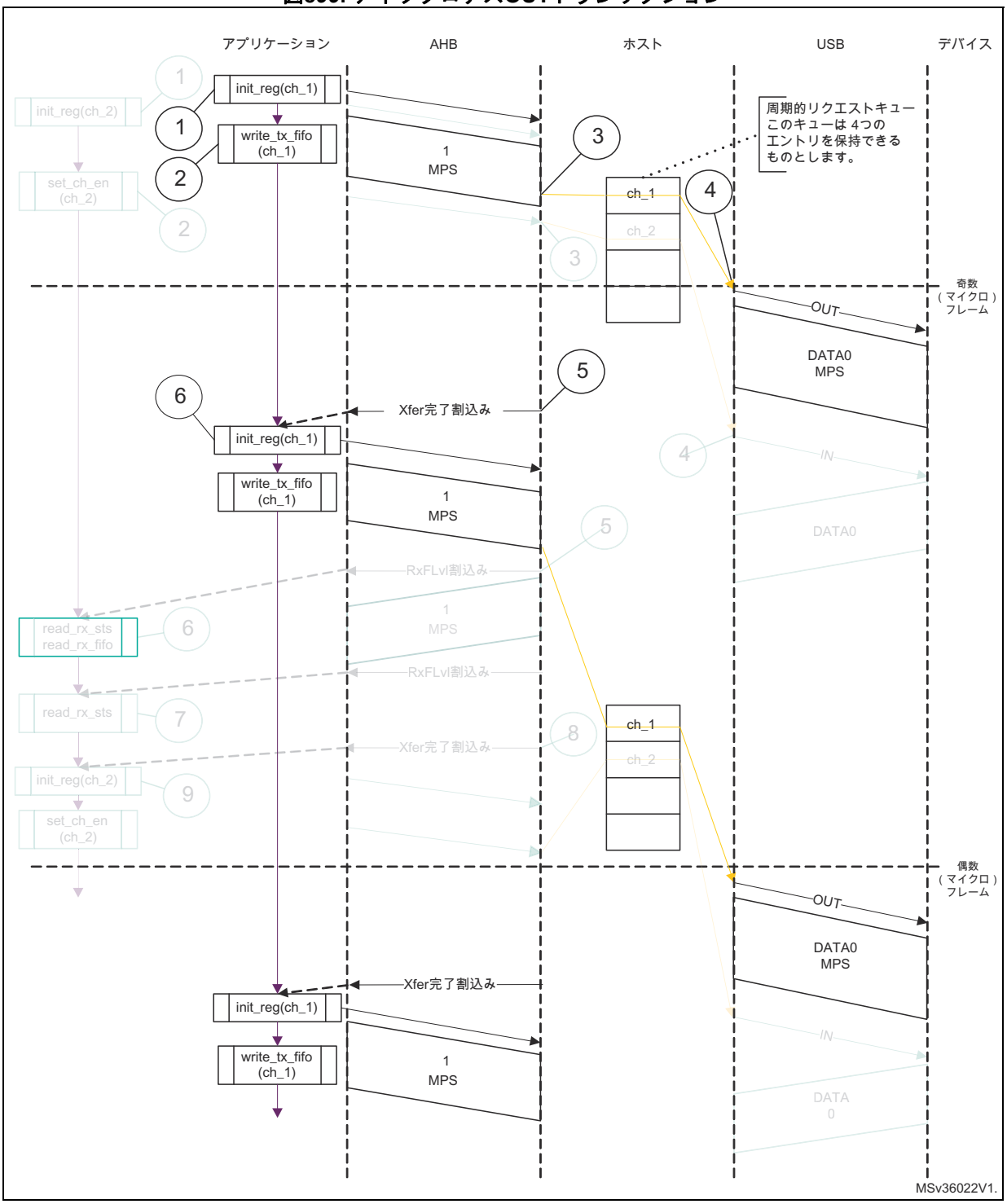
  – 周期的送信FIFOは1パケット（1KB）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル1を初期化し、有効にします。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。
2. チャネル1の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_FS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_FS ホストは、次のフレーム（奇数）でOUTトークンの送信を試みます。
5. OTG_FS ホストは、最後のパケットが正常に送信されると、XFRC割込みを生成します。
6. XFRC割込みに応答して、次の転送のためにチャネルを再初期化します。
7. 非ACK応答の取り扱い

図899. アイソクロナスOUTトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナスOUT/INトランザクションのための割込みサービスルーチン**

  コードサンプル：アイソクロナスOUT

```
Unmask (FRMOR/XFRC)
if (XFRC)
   {
   De-allocate Channel
   }
else
   if (FRMOR)
      {
      Unmask CHH
      Disable Channel
      }
   else
   if (CHH)
      {
      Mask CHH
      De-allocate Channel
      }
```

コードサンプル：アイソクロナス IN

```
Unmask (TXERR/XFRC/FRMOR/BBERR)
if (XFRC or FRMOR)
   {
   if (XFRC and (OTG_HCTSIZx.PKTCNT == 0))
      {
      Reset Error Count
      De-allocate Channel
      }
   else
      {
      Unmask CHH
      Disable Channel
      }
   }
   else
      if (TXERR or BBERR)
         {
         Increment Error Count
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
            {
            Mask CHH
```

```
if (Transfer Done or (Error_count == 3))
   {
   De-allocate Channel
   }
else
   {
   Re-initialize Channel
   }
}
```

- **アイソクロナスINトランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、次の奇数フレーム（転送サイズ = 1,024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の受信を試みています。

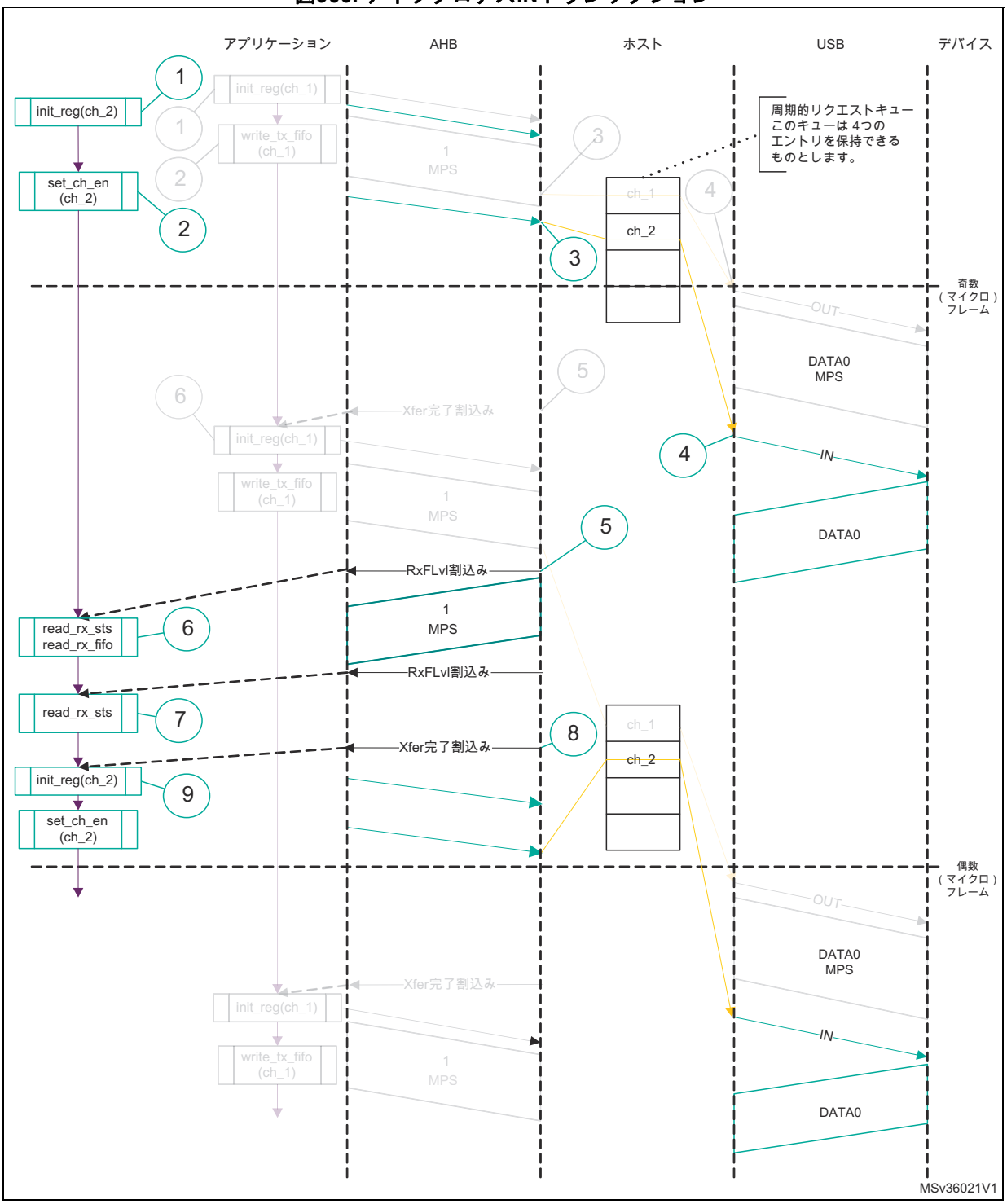
  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットとパケットあたり2つのステータスワード（1,031バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをセットする必要があります。
2. OTG_HCCHAR2のCHENAビットをセットしてINリクエストを周期的リクエストキューを書き込みます。
3. OTG_FS ホストは、CHENAビットがセットされたOTG_HCCHAR2レジスタの書込みごとに、周期的リクエストキューにINリクエストを書き込みます。
4. OTG_FS ホストは、次の奇数フレームでINトークンの送信を試みます。
5. INパケットが受信され、受信FIFOに書き込まれると、OTG_FS ホストはRXFLVL割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信FIFOを読み出します。アプリケーションは、受信FIFOを読み出す前にRXFVLV割込みをマスクし、すべてのパケットを読み出した後でマスクを解除しなければなりません。
7. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。今度は、アプリケーションは受信パケットステータスを読み出して、それがINデータパケットでない（OTG_GRXSTSRのPKTSTSビットが0b0010でない）ときには無視する必要があります。
8. コアは、受信パケットステータスが読み出されるとXFRC割込みを生成します。
9. XFRC割込みに応答してOTG_HCTSIZ2のPKTCNTフィールドを読み出します。OTG_HCTSIZ2のPKTCNT ≠ 0の場合、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2のPKTCNT = 0の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをリセットする必要があります。

図900. アイソクロナスINトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **キューの深さの選択**

  周期的および非周期的リクエストキューの深さは、アクセスされる周期的/非周期的エンドポイントの数に合うように注意して選択してください。

  非周期的リクエストキューの深さは、非周期的転送の性能に影響を与えます。キューが深いほど（FIFOのサイズが十分であれば）、コアは非周期的転送をより多くパイプライン化できます。キューのサイズが小さいと、コアはキューのスペースが空いたときしか新しいリクエストを入れることができません。

  コアの周期的リクエストキューの深さは、周期的転送をスケジュールどおりに実行するために不可欠です。周期的キューの深さは、マイクロフレーム内でスケジュールされた周期的転送の数に基づいて選択してください。周期的リクエストキューの深さがマイクロフレーム内でスケジュールされた周期的転送の数より小さい場合、フレームオーバーラン条件が発生します。

- **バブル条件の取り扱い**

  OTG_FS コントローラは、パケットバブルとポートバブルの2つのバブルを処理します。パケットバブルは、デバイスがチャネルの最大パケットサイズよりも多くのデータを送信した場合に発生します。ポートバブルは、コアがEOF2（フレーム2の終わりのSOFにきわめて近い）でデバイスからデータを受信し続けると発生します。

  OTG_FS コントローラがパケットバブルを検出すると、Rxバッファへのデータの書込みを停止して、パケットの終わり（EOP）を待ちます。EOPを検出すると コントローラはRxバッファにすでに書き込まれたデータを一掃して、アプリケーションに対するバブル割込みを生成します。

  OTG_FS コントローラがポートバブルを検出すると、Rx FIFOをFlashして、ポートを無効にします。コアは、ポートディセーブル割込み（OTG_GINTSTS の HPRTINT、OTG_HPRT の PENCHNG）を生成します。この割込みを受信すると、アプリケーションはOTG_HPRTのPOCAビットをチェックして、これが過電流条件（ポートディセーブル割込みのもう1つの原因）によるものではないことを確認してから、ソフトリセットを行う必要があります。コアは、ポートバブル条件を検出した後は、それ以上トークンを送信しません。

72.16.5 デバイスプログラミングモデル

USBリセット時のエンドポイントの初期化

1. すべてのOUTエンドポイントのNAKビットをセットします。
   - OTG_DOEPCTLxのSNAK = 1（すべてのOUTエンドポイントについて）
2. 以下の割込みビットのマスクを解除します。
   - OTG_DAINTMSKのINEP0 = 1（制御0 INエンドポイント）
   - OTG_DAINTMSKのOUTEP0 = 1（制御0 OUTエンドポイント）
   - OTG_DOEPMSKのSTUPM = 1
   - OTG_DOEPMSKのXFRCM = 1
   - OTG_DIEPMSKのXFRCM = 1
   - OTG_DIEPMSKのTOM = 1
3. 各FIFOのデータFIFO RAMをセットアップします。
   - OTG_GRXFSIZレジスタをプログラムして、制御OUTデータとセットアップデータを受信できるようにします。閾値の設定が無効の場合、これは、少なくとも、制御エンドポイント0の最大パケットサイズ1個分 + 2ワード（制御OUTデータパケットのステータス用）+ 10ワード（セットアップパケット用）でなければなりません。
   - OTG_DIEPTXF0レジスタを、制御INデータを送信できるようにプログラムします（選択されたFIFO番号に応じて）。これは、少なくとも、制御エンドポイント0の最大パケットサイズ1個分以上でなければなりません。

4. SETUPパケットを受信するために、制御OUTエンドポイント0のエンドポイント固有レジスタの以下のフィールドをプログラムします。
   – OTG_DOEPTSIZ0のSTUPCNT = 3（最大3つの連続SETUPパケットを受信するため）

この時点でSETUPパケットを受信するために必要なすべての初期化が終了したことになります。

エニュメレーション完了時のエンドポイント初期化

1. エニュメレーション終了割込み（OTG_GINTSTSのENUMDNEビット）時にはOTG_DSTSレジスタを読み出して、エニュメレーションスピードを決めます。
2. OTG_DIEPCTL0のMPSIZフィールドをプログラムして、最大パケットサイズを設定します。このステップでは、制御エンドポイント0を設定します。制御エンドポイントの最大パケットサイズは、エニュメレーションスピードに依存します。

この時点で、デバイスはSOFパケットを受信する準備ができ、制御エンドポイント0で制御転送を行うように設定されたことになります。

SetAddressコマンド受信時のエンドポイントの初期化

このセクションではSETUPパケットでSetAddressコマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. OTG_DCFGレジスタにSetAddressコマンドで受信したデバイスアドレスをプログラムします。
2. ステータスINパケットを送信するように、コアをプログラムします。

SetConfiguration/SetInterfaceコマンド受信時のエンドポイントの初期化

このセクションではSETUPパケットでSetConfigurationまたはSetInterfaceコマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. SetConfigurationコマンドを受信したとき、アプリケーションは、新しい設定で有効なエンドポイントの特性で、エンドポイントのレジスタをプログラムする必要があります。
2. SetInterfaceコマンドを受信したとき、アプリケーションは、このコマンドの影響を受けるエンドポイントのレジスタをプログラムする必要があります。
3. 前の設定または代替設定ではアクティブであったエンドポイントが、新しい設定または代替設定では無効なことがあります。これらの無効なエンドポイントは、機能を停止させる必要があります。
4. アクティブな各エンドポイントの割込みのマスクを解除し、OTG_DAINTMSKレジスタですべての非アクティブなエンドポイントに対する割込みをマスクします。
5. 各FIFOのデータFIFO RAMをセットアップします。
6. 必要なすべてのエンドポイントを設定した後、アプリケーションは、ステータスINパケットを送信するようにコアをプログラムする必要があります。

この時点で、デバイスのコアは、任意のタイプのデータパケットを送信／受信できるように設定されたことになります。

エンドポイントのアクティブ化

このセクションでは、デバイスエンドポイントをアクティブ化するか、既存のデバイスエンドポイントを新しいタイプに設定するために必要な手順について説明します。

1. 必要なエンドポイントの特性をOTG_DIEPCTLxレジスタ（INまたは双方向エンドポイントの場合）、またはOTG_DOEPCTLxレジスタ（OUTまたは双方向エンドポイントの場合）の以下のフィールドにプログラムします。
   – 最大パケットサイズ
   – USBアクティブエンドポイント = 1

– エンドポイント開始データトグル（割込みおよびバルクエンドポイントの場合）
– エンドポイントタイプ
– Tx FIFO番号

2. エンドポイントがアクティブ化されると、コアは、そのエンドポイントに宛てたトークンのデコードを開始し、そのエンドポイントで受信された有効な各トークンについて有効なハンドシェイクを送出します。

エンドポイントの機能停止

このセクションでは、既存のエンドポイントの機能を停止させるために必要な手順について説明します。

1. 非アクティブにすべきエンドポイントでは、OTG_DIEPCTLxレジスタ（INまたは双方向エンドポイントの場合）、またはOTG_DOEPCTLxレジスタ（OUTまたは双方向エンドポイントの場合）のUSBのアクティブエンドポイントビットをクリアします。
2. エンドポイントが非アクティブな状態になると、コアは、そのエンドポイントにアドレス指定されたトークンを無視するのでUSB上でタイムアウトが発生します。

注： **アプリケーションは、以下の条件を満たすように、デバイスコアがトラフィックを処理できるようにセットアップする必要があります。**
OTG_GINTMSKレジスタのNPTXFEMおよびRXFLVLMをクリアする必要があります。

動作モデル

SETUPおよびOUTデータ転送

このセクションでは、データOUT転送およびSETUPトランザクション時の内部データフローとアプリケーションレベルの動作について説明します。

- **パケットの読出し**

このセクションでは、受信FIFOからパケット（OUTデータおよびSETUPパケット）を読み出す方法について説明します。

1. RXFLVL割込み（OTG_GINTSTSレジスタ）を受信すると、アプリケーションは、受信ステータスポップレジスタ（OTG_GRXSTSP）を読み出す必要があります。
2. アプリケーションは、RXFLVL = 0（OTG_GINTMSKレジスタ）を書き込むことによって、受信FIFOからそのパケットが読み出されるまで、RXFLVL割込み（OTG_GINTSTSレジスタ）をマスクすることができます。
3. 受信パケットのバイトカウントが0でない場合、データのバイトカウントは受信データFIFOからポップされ、メモリに格納されます。受信パケットのバイトカウントが0の場合、受信データFIFOからデータはポップされません。
4. FIFOのパケット受信ステータスの読出し結果は、以下のいずれかを示します。
   a) グローバルOUT NAKのパターン
   PKTSTS = グローバルOUT NAK、BCNT = 0x000、EPNUM = （0x0）、DPID = （0b00）。
   これらのデータは、グローバルOUT NAKビットが有効になっていることを示します。
   b) SETUPパケットのパターン
   PKTSTS = SETUP、BCNT = 0x008、EPNUM = 制御EP番号、DPID = DATA0。これらのデータは、指定されたエンドポイントのSETUPパケットを受信FIFOから読み出せることを示します。
   c) セットアップステージ終了パターン
   PKTSTS = セットアップステージ終了、BCNT = 0x0、EPNUM = 制御EP番号、DPID = （0b00）。
   これらのデータは、指定されたエンドポイントのセットアップステージが完了し、データステージが開始されたことを示します。このエントリが受信FIFOからポップされた後、コアは、指定された制御OUTエンドポイントでセットアップ割込みをアサートします。

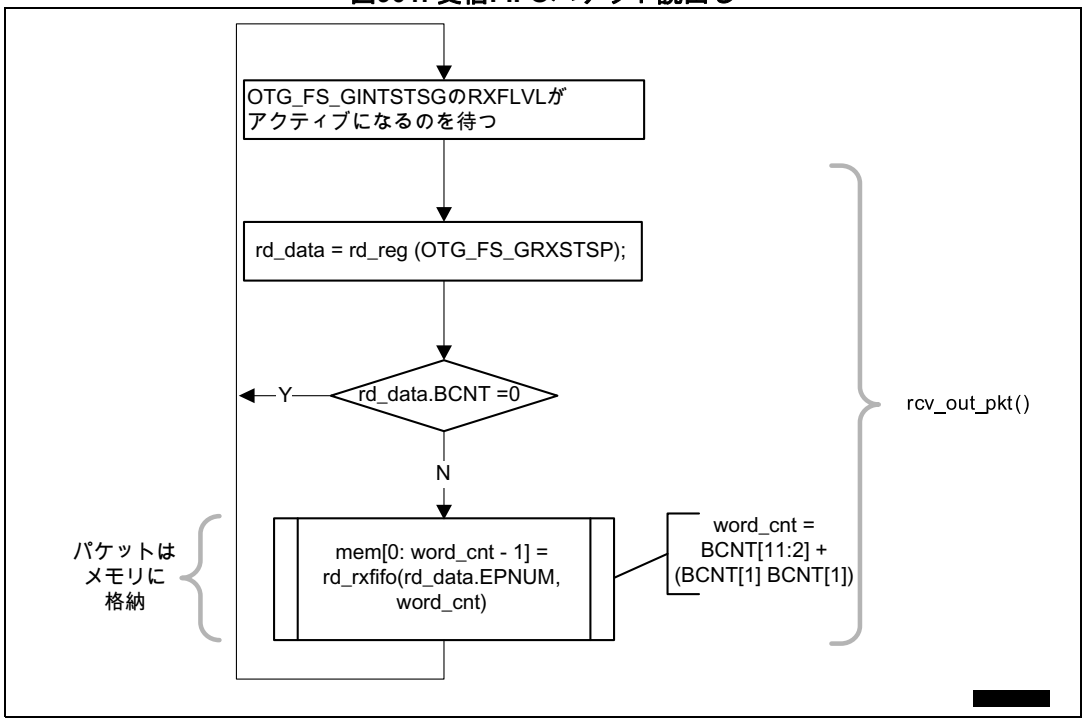
d) データOUTパケットのパターン
PKTSTS = データOUT、BCNT = 受信データOUTパケットのサイズ（0 ≤ BCNT ≤ 1 024）、EPNUM = パケットを受信したEPNUM、DPID = 実際のデータPID。

e) データ転送完了パターン
PKTSTS = データOUT転送終了、BCNT = 0x0、EPNUM = データ転送が完了したOUT EP番号、DPID = （0b00）。
これらのデータは、指定されたOUTエンドポイントのOUTデータ転送が完了したことを示します。このエントリが受信FIFOからポップされた後、コアは、指定されたOUTエンドポイントで転送完了割込みをアサートします。

5. データペイロードが受信 FIFO からポップされた後、RXFLVL 割込み（OTG_GINTSTS）がマスク解除されなければなりません。
6. 手順1～5は、アプリケーションがOTG_GINTSTS レジスタのRXFLVLによる割込みラインのアサーションを検出するたびに繰り返されます。空の受信FIFOを読み出すと、定義されていないコア動作を引き起こすことがあります。

図901上記の手順のフローチャートです。

図901. 受信FIFOパケット読出し



SETUPトランザクション

このセクションでは、コアがSETUPパケットを処理する方法と、SETUP トランザクションを処理するアプリケーションのシーケンスについて説明します。

- **アプリケーションの要件**

1. SETUP パケットを受信するには、制御 OUT エンドポイントの STUPCNT フィールド（OTG_DOEPTSIZxレジスタ）が、ゼロでない値にプログラムされなければなりません。アプリケーションがSTUPCNTフィールドをゼロでない値にプログラムすると、NAKのステータスとOTG_DOEPCTLxレジスタのEPENAビットの設定に関係なく、コアは、SETUPパケットを受信して、受信FIFOに書き込みます。STUPCNTフィールドは、制御エンドポイントがSETUPパケットを受信するたびにデクリメントされます。SETUPパケットを受信する前に、STUPCNTフィー

ルドが適切な値にプログラムされていなかった場合、コアはSETUP パケットを受信し、STUPCNTフィールドをデクリメントしますが、アプリケーションは制御転送のセットアップステージで受信したSETUPパケットの正しい数を判定できないことがあります。

– OTG_DOEPTSIZxレジスタのSTUPCNT = 3

2. 制御エンドポイントで3つまでのSETUPパケットを受信するためには、アプリケーションは、常に受信データFIFOに余分のスペースを割り当てておく必要があります。
   – 確保すべきスペースは10ワード分です。最初のSETUPパケット用に3ワードが必要であり、セットアップステージ終了ワードのために1ワード、すべての制御エンドポイントにわたって2つの余分なSETUPパケットを格納するために6ワードが必要です。
   – 8バイトのSETUPデータと4バイトのSETUPステータス（セットアップパケットパターン）を格納するには、SETUPパケットあたり3ワードが必要です。コアは受信データFIFOのこのスペースをSETUPデータを書き込むためにのみ確保し、このスペースをデータパケットのために使うことはありません。
3. アプリケーションは、受信FIFOからSETUPパケットの2ワードを読み出す必要があります。
4. アプリケーションは受信FIFOからセットアップステージ終了ワードを読み出して、破棄する必要があります。

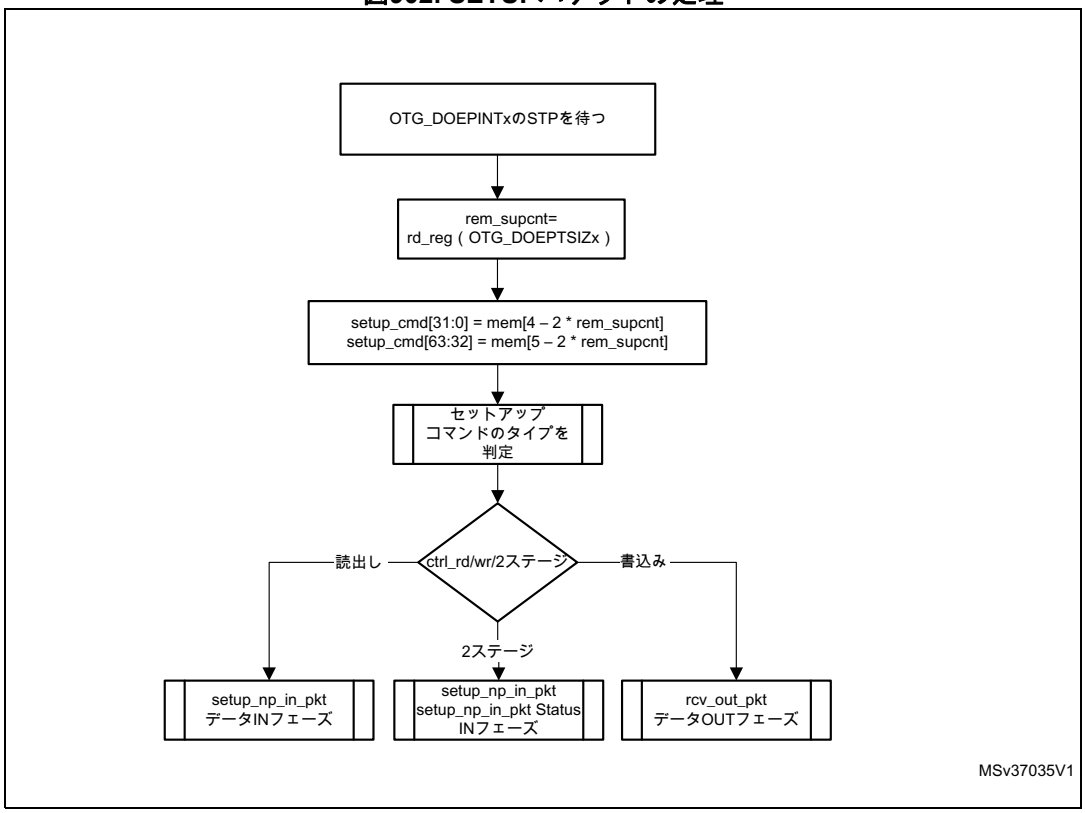
- **内部データフロー**

1. SETUPパケットが受信されると、コアは受信FIFO内の空きスペースをチェックせずに、また、エンドポイントのNAKおよびSTALLビットの設定に関係なく、受信したデータを受信FIFOに書き込みます。
   – コアは、SETUPパケットが受信された制御IN/OUTエンドポイントのIN　NAKおよびOUT NAKビットを内部でセットします。
2. USB上で受信された各SETUPパケットについて、3ワードのデータが受信FIFOに書き込まれ、STUPCNTフィールドが1ずつデクリメントされます。
   – 最初のワードは、コアが内部的に使用する制御情報を含みます。
   – 2番目のワードは、SETUPコマンドの最初の4バイトを含みます。
   – 3番目のワードは、SETUPコマンドの最後の4バイトを含みます。
3. セットアップステージがデータIN/OUTステージに変わると、コアは、エントリ（セットアップステージ終了ワード）を受信FIFOに書き込んで、セットアップステージの完了を示します。
4. AHB側では、SETUPパケットはアプリケーションによってエンプティにされます。
5. アプリケーションが受信 FIFO からセットアップステージ終了ワードをポップすると、コアは、SETUP割込み（OTG_DOEPINTx）でアプリケーションに割り込んで、受信したSETUPパケットを処理できることを示します。
6. コアは、制御OUTエンドポイントのエンドポイントイネーブルビットをクリアします。

- **アプリケーションのプログラミングシーケンス**：

1. OTG_DOEPTSIZxレジスタをプログラムします。
   – STUPCNT = 3
2. RXFLVL割込み（OTG_GINTSTS）を待ち、受信FIFOからデータパケットをエンプティにします。
3. STUP割込み（OTG_DOEPINTx）のアサートは、SETUPデータ転送が正常に完了したことを示します。
   – この割込み時、アプリケーションは、OTG_DOEPTSIZx　レジスタを読み出して、受信したSETUPパケットの数を確認し、最後に受信したSETUPパケットを処理する必要があります。

図902. SETUPパケットの処理



- **3つを超える連続SETUPパケットの処理**

USB 2.0仕様に従い、通常SETUPパケットエラー時にはホストは同じエンドポイントに対して3つを超える連続SETUPパケットを送信しません。ただしUSB 2.0仕様では、ホストが同じエンドポイントに送信できる連続SETUPパケットの数を制限してはいません。この条件が発生すると、OTG_FS コントローラは、割込み（OTG_DOEPINTxのB2BSTUP）を生成します。

- **グローバルOUT NAKの設定**

内部データフロー：

1. アプリケーションがグローバルOUT NAK（OTG_DCTLのSGONAKビット）をセットすると、コアはSETUPパケットを除き、受信FIFOへのデータの書込みを停止します。受信FIFOの使用可能なスペースの有無に関係なく、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信し、コアはアイソクロナスOUTデータパケットを無視します。
2. コアは、グローバルOUT　NAKパターンを受信FIFOに書き込みます。アプリケーションがこのデータパターンを書き込むには、受信FIFOに十分なスペースを確保する必要があります。
3. アプリケーションがグローバルOUT　NAKパターンワードを受信FIFOからポップすると、コアは、GONAKEFF割込み（OTG_GINTSTS）をセットします。
4. この割込みを検出したアプリケーションは、コアがグローバルOUT NAKモードにあるとみなすことができます。アプリケーションはOTG_DCTLのSGONAKビットをクリアすることによって、この割込みをクリアできます。

アプリケーションのプログラミングシーケンス：

1. 受信FIFOでのデータの受信も停止するには、アプリケーションは、以下のフィールドをプログラムすることによって、グローバルOUT NAKビットをセットする必要があります。
   – OTG_DCTLレジスタのSGONAK = 1
2. OTG_GINTSTSのGONAKEFF割込みのアサートを待ちます。アサートされた場合、この割込みは、コアがSETUPパケット以外のあらゆるタイプのデータの受信を停止したことを示します。
3. アプリケーションはOTG_DCTLのSGONAKビットをセットした後、コアがGONAKEFF割込み（OTG_GINTSTS）をアサートする前に、有効なOUTパケットを受信することができます。
4. アプリケーションは、OTG_GINTMSK レジスタの GONAKEFFM ビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_GINTMSKレジスタのGONAKEFFM = 0
5. アプリケーションがグローバルOUT　NAKモードを終了する準備ができたときにはOTG_DCTLレジスタのSGONAKビットをクリアする必要があります。GONAKEFF割込み（OTG_GINTSTS）もクリアされます。
   – OTG_DCTLレジスタのCGONAK = 1
6. アプリケーションがこの割込みをマスクするのが早すぎた場合は、以下のようにしてマスク解除を行う必要があります。
   – OTG_GINTMSKレジスタのGONAKEFFM = 1

- **OUTエンドポイントの無効化**

アプリケーションが有効にしたOUTエンドポイントを無効にするには、このシーケンスを使用する必要があります。

アプリケーションのプログラミングシーケンス：

1. OUTエンドポイントを無効にする前に、アプリケーションは、コアのグローバルOUT NAKモードを有効にする必要があります。
   – OTG_DCTLレジスタのSGONAK = 1
2. OTG_GINTSTSのGONAKEFF割込みを待ちます。
3. 以下のフィールドをプログラムすることによって、必要なOUTエンドポイントを無効にします。
   – OTG_DOEPCTLxレジスタのEPDIS = 1
   – OTG_DOEPCTLxレジスタのSNAK = 1
4. EPDISD割込み（OTG_DOEPINTx）を待ちます。これはOUTエンドポイントが完全に無効にされたことを示します。EPDISD割込みがアサートされると、コアは、以下のビットもクリアします。
   – OTG_DOEPCTLxレジスタのEPDIS = 0
   – OTG_DOEPCTLxレジスタのEPENA = 0
5. アプリケーションは、無効にされていない他のエンドポイントからのデータ受信を開始するために、グローバルOUT NAKビットをクリアする必要があります。
   – OTG_DCTLレジスタのSGONAK = 0

- **OUTエンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング･シーケンスを使用する必要があります。

動作シーケンス：

1. 以下をセットして、すべてのOUTエンドポイントを有効にします。
   – すべてのOTG_DOEPCTLxレジスタのEPENA = 1
2. 次のようにRxFIFOを排出します。
   – 1になるまでOTG_GRSTCTL.AHBIDLをポーリングします。これは、AHBマスタがアイドルであることを示します。
   – OTG_GRSTCTL.RXFFLSH = 1で読出し、変更、書込み操作を実行します。
   – 0になるまでOTG_GRSTCTL.RXFFLSHをポーリングしますが、（最小リセットシグナリング時間に応じて）10ミリ秒未満のタイムアウトも使用します。タイムアウト前に0が確認された場合、RxFIFO排出は完了です。その時点でタイムアウトが発生した場合、まだ1が残っており（ホストから来るEP0のパケットによる場合がある）、（1回だけ）前の手順（読出し変更、書込み動作の実行）に戻ります。
3. OUTエンドポイントを無効にする前に、アプリケーションは3227ページのグローバルOUT NAKの設定の指示に従ってコアのグローバルOUT NAKモードを有効にする必要があります。これによって、RxFIFOのデータがアプリケーションに正常に送信されたことを確認します。OTG_DCTLでSGONAK = 1をセットします。
4. OTG_GINTSTSのGONAKEFF割込みを待ちます。
5. 以下のレジスタビットをプログラムすることによって、すべてのアクティブなOUTエンドポイントを無効にします。
   – OTG_DOEPCTLxレジスタのEPDIS = 1
   – OTG_DOEPCTLxレジスタのSNAK = 1
6. 前の手順でプログラムされた各OUTエンドポイントについてOTG_DOEPINTxでEPDIS割込みを待ちます。OTG_DOEPINTxでのEPDIS割込みは対応するOUTエンドポイントが完全に無効にされたことを示します。EPDIS割込みがアサートされると、以下のビットがクリアされます。
   – OTG_DOEPCTLxレジスタのEPENA = 0
   – OTG_DOEPCTLxレジスタのEPDIS = 0
   – OTG_DOEPCTLxレジスタのSNAK = 0

- **一般の非アイソクロナスOUTデータ転送**

このセクションでは、通常の非アイソクロナスOUTデータ転送（制御、バルク、または割込み）について説明します。

アプリケーションの要件：

1. OUT転送をセットアップする前に、アプリケーションはOUT転送の一部として受信されるすべてのデータを収容できるバッファをメモリ内で割り当てる必要があります。
2. OUT 転送の場合、エンドポイントの転送サイズレジスタの転送サイズフィールドは、エンドポイントの最大パケットサイズの倍数でなければならず、ワードの境界に整列されていなければなりません。
   – 転送サイズ [EPNUM] = *n* * (MPSIZ[EPNUM] + 4 – (MPSIZ[EPNUM] mod 4))
   – パケットカウント [EPNUM] = *n*
   – *n* > 0
3. OUT エンドポイント割込み時には、アプリケーションは、エンドポイントの転送サイズレジスタを読み出して、メモリ内のペイロードのサイズを計算しなければなりません。受信したペイロードのサイズが、プログラムされた転送サイズより小さいこともあります。
   – メモリ内のペイロードのサイズ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ

– このペイロードが受信されたUSBパケットの数 = アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

内部データフロー：

1. アプリケーションがデータを受信するためには、エンドポイント固有レジスタの転送サイズおよびパケットカウントのフィールドを設定し、NAKビットをクリアし、エンドポイントを有効にする必要があります。
2. NAKビットがクリアされると、コアは、データの受信を開始し、受信FIFOにスペースがある限り、データを受信FIFOに書き込みます。USBで受信された各データパケットについて、データパケットとそのステータスが受信FIFOに書き込まれます。受信FIFOにパケット（最大パケットサイズまたはショートパケット）が書き込まれるたびに、そのエンドポイントのパケットカウントフィールドが1ずつデクリメントされます。
   – 受信されたOUTデータパケットのデータCRCが不良な場合、受信FIFOから自動的に一掃されます。
   – USB上のパケットに対してACKを送信した後、コアはACKを検出できないホストが再送信する非アイソクロナスOUTデータパケットを破棄します。アプリケーションは、同じエンドポイント上では、同じデータPIDを持つ複数の連続データOUTパケットを検出しません。この場合、パケットカウントはデクリメントされません。
   – 受信FIFOにスペースがない場合、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信FIFOには書き込まれません。さらに、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信します。
   – 上記の3つのケースのすべてにおいて、データは受信FIFOに書き込まれないので、パケットカウントはデクリメントされません。
3. パケットカウントが0になるか、エンドポイント上でショートパケットが受信されると、そのエンドポイントのNAKビットがセットされます。NAKビットがセットされると、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信FIFOには書き込まれず、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信します。
4. データが受信FIFOに書き込まれた後、アプリケーションは受信FIFOからデータを読み出して、エンドポイントあたり一度に1パケットずつ外部メモリに書き込みます。
5. AHB 上で外部メモリへのパケットの書込みが終わるたびに、書き込まれたパケットのサイズだけエンドポイントの転送サイズがデクリメントされます。
6. 以下の条件の1つでOUTエンドポイントのOUTデータ転送完了パターンが受信FIFOに書き込まれます。
   – 転送サイズが0、およびパケットカウントが0。
   – 受信FIFOに書き込まれた最後のOUTデータパケットがショートパケット。（0 ≤ パケットサイズ < 最大パケットサイズ）
7. アプリケーションがこのエントリ（OUT データ転送完了）をポップすると、エンドポイントの転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZxレジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLxレジスタをエンドポイントの特性でプログラムし、EPENAビットおよびCNAKビットをセットします。
   – OTG_DOEPCTLxレジスタのEPENA = 1
   – OTG_DOEPCTLxレジスタのCNAK = 1
3. RXFLVL割込み（OTG_GINTSTS）を待ち、受信FIFOからデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。

4. XFRC割込み（OTG_DOEPINTx）のアサートは、非アイソクロナスOUTデータ転送が正常に完了したことを示します。
5. OTG_DOEPTSIZxレジスタを読み出して、受信したデータペイロードのサイズを確認します。

- **一般のアイソクロナスOUTデータ転送**

このセクションでは、通常のアイソクロナスOUTデータ転送について説明します。

アプリケーションの要件：

1. 非アイソクロナス OUT データ転送のアプリケーションの要件はすべて、アイソクロナス OUT データ転送にも適用されます。
2. アイソクロナス OUT データ転送の場合、転送サイズおよびパケットカウントフィールドは、常に、単一フレームで受信できる最大パケットサイズのパケット数を設定しなければなりません。アイソクロナスOUTデータ転送は、複数のフレームにまたがってはいけません。
3. アプリケーションは、周期的フレームの終わり（OTG_GINTSTSのEOPF割込み）の前に、受信FIFOからすべてのアイソクロナスOUTデータパケット（データおよびステータス）を読み出す必要があります。
4. 次のフレームでデータを受信するにはEOPF（OTG_GINTSTS）の後とSOF（OTG_GINTSTS）の前に、アイソクロナスOUTエンドポイントを有効にしなければなりません。

内部データフロー：

1. アイソクロナスOUTエンドポイントの内部データフローは、非アイソクロナスOUTエンドポイントの内部データフローと基本的に同じですが、少し異なったところがあります。
2. エンドポイントイネーブルビットをセットし、NAK ビットをクリアすることによって、アイソクロナスOUTエンドポイントが有効にされたときには、偶数／奇数フレームビットも適切にセットされなければなりません。コアは、以下の条件が満たされた場合に限り、アイソクロナスOUTエンドポイント上で特定のフレームのデータを受信します。
   – EONUM（OTG_DOEPCTLxレジスタ）= FNSOF[0]（OTG_DSTSレジスタ）
3. アプリケーションが受信 FIFO からアイソクロナス OUT データパケット（データとステータス）を完全に読み出すと、コアはOTG_DOEPTSIZxレジスタのRXDPIDフィールドを、受信FIFOから読み出された最後のアイソクロナスOUTデータパケットのデータPIDで更新します。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx　レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、エンドポイントイネーブル、ClearNAK、および偶数／奇数フレームの各ビットをセットします。
   – EPENA = 1
   – CNAK = 1
   – EONUM =（0：偶数／1：奇数）
3. RXFLVL 割込み（OTG_GINTSTS）を待ち、受信 FIFO からのデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC割込み（OTG_DOEPINTx）のアサートは、アイソクロナスOUTデータ転送が正常に完了したことを示します。この割込みは、必ずしもメモリ内のデータが良好であることを意味しません。
5. この割込みは、アイソクロナス OUT 転送で必ず検出されるとは限りません。その代わり、アプリケーションはOTG_GINTSTSでINCOMPISOOUT割込みを検出することができます。

6. OTG_DOEPTSIZx レジスタを読み出して、受信した転送データのサイズを確認し、フレームで受信したデータの有効性を確認します。アプリケーションは、以下の条件の1つが満たされた場合のみ、メモリに受信されたデータを有効として扱う必要があります。
   – RXDPID = DATA0（OTG_DOEPTSIZxレジスタ）、およびこのペイロードが受信されたUSBパケットの数 = 1
   – RXDPID = DATA1（OTG_DOEPTSIZxレジスタ）、およびこのペイロードが受信されたUSBパケットの数 = 2

   このペイロードが受信されたUSBパケットの数 =
   アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

   アプリケーションは、無効なデータパケットを破棄できます。

- **不完全アイソクロナスOUTデータ転送**

このセクションでは、アイソクロナスOUTデータパケットがコアの内部でドロップされたときのアプリケーションのプログラミングシーケンスについて説明します。

内部データフロー：

1. アイソクロナスOUTエンドポイントの場合、XFRC割込み（OTG_DOEPINTx）は必ずアサートされるわけではありません。コアがアイソクロナスOUTデータパケットをドロップした場合、アプリケーションは以下の状況でXFRC割込み（OTG_DOEPINTx）の検出に失敗することがあります。
   – 受信FIFOが完全なISO　OUTデータパケットを収容できない場合、コアは、受信したISO OUTデータをドロップします。
   – アイソクロナスOUTデータパケットがCRCエラー付きで受信されたとき。
   – コアが受信したアイソクロナスOUTトークンが破損しているとき。
   – アプリケーションが受信FIFOからデータを読み出すのに長時間かかっているとき。
2. コアが、すべてのアイソクロナス OUT エンドポイントへの転送を完了する前に周期的フレームの終わりを検出すると、不完全アイソクロナスOUTデータ割込み（OTG_GINTSTSレジスタのINCOMPISOOUT）をアサートして、アイソクロナスOUTエンドポイントのうち少なくとも1つでXFRC割込み（OTG_DOEPINTx）がアサートされていないことを示します。この時点で、不完全転送のエンドポイントは有効なままですがUSB上のこのエンドポイントでは、アクティブな転送は進行していません。

アプリケーションのプログラミングシーケンス：

1. INCOMPISOOUT割込み（OTG_GINTSTS）のアサートは、現在のフレームで、少なくとも1つのアイソクロナスOUTエンドポイントが転送を完了していないことを示します。
2. アイソクロナス OUT データがエンドポイントから完全に出されていないためにこの割込みが発生した場合、アプリケーションは処理を進める前に、受信FIFOからすべてのアイソクロナスOUTデータ（データとステータス）を出力する必要があります。
   – すべてのデータが受信　FIFO　から出されると、アプリケーションは　XFRC　割込み（OTG_DOEPINTx）を検出することができます。この場合、アプリケーションは、次のフレームでアイソクロナスOUTデータを受信するには、エンドポイントを再び有効にする必要があります。
3. INCOMPISOOUT割込み（OTG_GINTSTS）を受信すると、アプリケーションは、すべてのアイソクロナスOUTエンドポイントの制御レジスタ（OTG_DOEPCTLx）を読み出して、現在のマイクロフレームにおいて転送を完了しなかったエンドポイントを確認する必要があります。次の両方の条件が満たされた場合、エンドポイント転送は完了しません。
   – EONUMビット（OTG_DOEPCTLxレジスタ）= FNSOF[0]（OTG_DSTSレジスタ）
   – OTG_DOEPCTLxレジスタのEPENA = 1

4. 現在のフレーム番号が変更されないように、上記の手順はSOF割込み（ OTG_GINTSTS）が検出される前に実行されなければなりません。
5. 不完全転送のアイソクロナスOUTエンドポイントの場合、アプリケーションは、メモリ内のデータを破棄しOTG_DOEPCTLxレジスタのEPDISビットをセットすることによってエンドポイントを無効にする必要があります。
6. EPDISD 割込み（OTG_DOEPINTx）を待ち、次のフレームで新しいデータを受信するために、エンドポイントを有効にします。
   – コアがエンドポイントを無効にするには若干の時間がかかるので、アプリケーションは不良なアイソクロナスデータを受信した後、次のフレームのデータを受信できないことがあります。

- **非アイソクロナスOUTエンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

1. コアをグローバルOUT NAKモードにします。
2. 必要なエンドポイントを無効にします。
   – エンドポイントを無効にするときには、OTG_DOEPCTLのSNAKビットをセットする代わりにSTALL = 1にセットしてください（OTG_DOEPCTL）。

   STALLビットは常にNAKビットより優先されます。
3. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときは、STALLビット （OTG_DOEPCTLx）をクリアしなければなりません。
4. アプリケーションがSetFeature.Endpoint HaltまたはClearFeature.Endpoint Haltコマンドによってエンドポイントの STALLビットをセットまたはクリアする場合、STALLビットは、アプリケーションが制御エンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

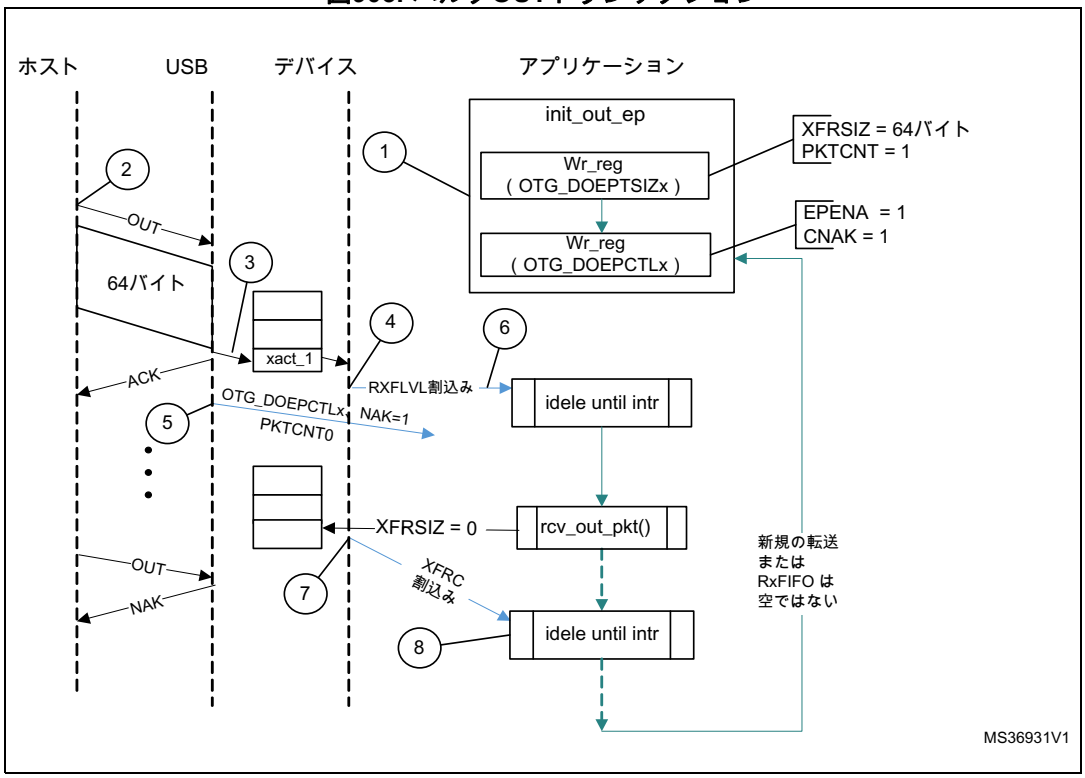
例

このセクションでは、いくつかの基本的な転送タイプとシナリオについて説明します。

- バルクOUTトランザクション

図903にUSBからAHBへの単一バルクOUTデータパケットの受信と、このプロセスに伴うイベントを示します。

図903. バルクOUTトランザクション



SetConfiguration/SetInterfaceコマンドの後、アプリケーションはCNAK = 1およびEPENA = 1（OTG_DOEPCTLx レジスタ）をセットし、OTG_DOEPTSIZx レジスタの適切なXFRSIZ およびPKTCNTビットをセットすることによって、すべてのOUTエンドポイントを初期化します。

1. ホストは、データ（OUTトークン）のエンドポイントへの送信を試みます。
2. コアは、USB上でOUTトークンを受信すると、スペースが使用可能なので、Rx FIFOにパケットを格納します。
3. Rx FIFOに完全なパケットを書き込んだ後、コアはRxFLVL割込み（OTG_GINTSTS）をアサートします。
4. USBパケットのPKTCNT番号を受信すると、コアは、それ以上パケットが受信されないように、このエンドポイントのNAKビットを内部でセットします。
5. アプリケーションは割込みを処理してRx FIFOからデータを読み出します。
6. アプリケーションがすべてのデータ（XFRSIZ に相当）を読み出すと、コアは、XFRC 割込み（OTG_DOEPINTx）を生成します。
7. アプリケーションは割込みを処理して、XFRC割込みビット（OTG_DOEPINTx）の設定を使用して、意図した転送が完了したかどうかを確認します。

INデータ転送

- **パケットの書込み**

このセクションでは、専用の送信FIFOが有効なときに、アプリケーションがデータパケットをエンドポイントFIFOに書き込む方法について説明します。

1. アプリケーションは、ポーリングまたは割込みモードのいずれかを選択できます。
   – ポーリングモードでは、アプリケーションは、OTG_DTXFSTSxレジスタを読み出すことによってエンドポイント送信データFIFOのステータスを監視して、データFIFOに十分なスペースがあるかどうかを確認します。
   – 割込みモードでは、アプリケーションは、TXFE割込み（OTG_DIEPINTx）を待ってから、OTG_DTXFSTSxレジスタを読み出して、データFIFO内に十分なスペースがあるかどうかを確認します。
   – 単一の非ゼロ長データパケットを書き込むには、パケット全体を書き込むためのスペースがデータFIFOになければなりません。
   – ゼロ長パケットを書き込むには、アプリケーションはFIFOのスペースを考慮してはなりません。
2. 上記の方法の1つを使用して、アプリケーションが送信パケットを書き込むのに十分なスペースがあることを確認するときには、アプリケーションは、データをデータFIFOに書き込む前に、まず、エンドポイント制御レジスタに書き込む必要があります。通常、アプリケーションは、エンドポイントイネーブルビットをセットする場合を除き、レジスタの内容が変更されないように、OTG_HS_DIEPCTLxレジスタに読出し／変更／書込みを行う必要があります。

アプリケーションは、使用可能なスペースがあれば、同じエンドポイントに対する複数のパケットを送信FIFOに書き込むことができます。周期的INエンドポイントの場合、アプリケーションは1つのマイクロフレームのパケットのみを書き込む必要があります。アプリケーションは、前のトランザクションの転送完了割込みを受信した後でのみ、次の周期的トランザクションのパケットを書き込むことができます。

- **INエンドポイントNAKの設定**

内部データフロー：

1. アプリケーションが特定のエンドポイントのIN NAKをセットすると、コアは、エンドポイントの送信FIFOにデータがあるかどうかに関係なく、そのエンドポイントでのデータ送信を停止します。
2. 非アイソクロナスINトークンはNAKハンドシェイク応答を受信します。
   – アイソクロナスINトークンはゼロデータ長のパケット応答を受信します。
3. コアは、OTG_DIEPCTLxのSNAKビットに応答して、OTG_DIEPINTxのINEPNE割込み（INエンドポイントNAK有効）をアサートします。
4. この割込みがアプリケーションによって検出されると、アプリケーションは、エンドポイントがIN NAKモードにあるとみなすことができます。この割込みは、アプリケーションがOTG_DIEPCTLxのCNAKビットをセットすることによってクリアできます。

アプリケーションのプログラミングシーケンス：

1. 特定のINエンドポイントでのデータ送信を停止するには、アプリケーションは、IN NAKビットをセットする必要があります。このビットをセットするには、以下のフィールドをプログラムする必要があります。
   – OTG_DIEPCTLxレジスタのSNAK = 1
2. OTG_DIEPINTxのINEPNE割込みのアサートを待ちます。この割込みは、コアがエンドポイントでのデータ送信を停止したことを示します。

3. コアは、アプリケーションがNAKビットをセットした後のNAK有効割込みがアサートされる前に、エンドポイントで有効なINデータを送信できます。
4. アプリケーションはOTG_DIEPMSKのINEPNEMビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_DIEPMSKレジスタのINEPNEM = 0
5. エンドポイントNAKモードを終了するには、アプリケーションはOTG_DIEPCTLxのNAKステータスビット（NAKSTS）をクリアする必要があります。これによってINEPNE割込み（OTG_DIEPINTx）もクリアされます。
   – OTG_DIEPCTLxレジスタのCNAK = 1
6. アプリケーションがこの割込みをマスクするのが早すぎた場合、以下のようにしてマスク解除を行う必要があります。
   – OTG_DIEPMSKレジスタのINEPNEM = 1

- **INエンドポイントディセーブル**

以前に有効にされた特定のINエンドポイントを無効にするには、以下のシーケンスを使用します。

アプリケーションのプログラミングシーケンス：

1. アプリケーションは、AHB でのデータ書込みを停止して、IN エンドポイントを無効にする必要があります。
2. アプリケーションは、エンドポイントをNAKモードに設定する必要があります。
   – OTG_DIEPCTLxレジスタのSNAK = 1
3. OTG_DIEPINTxのINEPNE割込みを待ちます。
4. 無効にしなければならないエンドポイントについてOTG_DIEPCTLxレジスタの以下のビットをセットします。
   – OTG_DIEPCTLxレジスタのEPDIS = 1
   – OTG_DIEPCTLxレジスタのSNAK = 1
5. OTG_DIEPINTxレジスタのEPDISD割込みのアサートは、コアが指定されたエンドポイントを完全に無効にしたことを示します。割込みのアサートとともに、コアは、以下のビットもクリアします。
   – OTG_DIEPCTLxレジスタのEPENA = 0
   – OTG_DIEPCTLxレジスタのEPDIS = 0
6. アプリケーションは、周期的IN EPのOTG_DIEPTSIZxレジスタを読み出して、エンドポイント上のどれだけのデータがUSBで送信されたかを計算する必要があります。
7. アプリケーションは、OTG_GRSTCTL レジスタの以下のフィールドを設定することによって、エンドポイント送信FIFO内のデータを排出する必要があります。
   – TXFNUM（OTG_GRSTCTL）= エンドポイント送信FIFO番号
   – TXFFLSH（OTG_GRSTCTL ）= 1

アプリケーションは、TXFFLSHビットがコアによってクリアされるまで（すなわち、一掃動作の終了まで）、OTG_GRSTCTLレジスタにポーリングする必要があります。このエンドポイントで新しいデータを送信するために、アプリケーションは後で、エンドポイントを再び有効にできます。

- **INエンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング･シーケンスを使用する必要があります。

動作シーケンス：

1. INエンドポイントを無効にするには、以下をセットします。
   – すべてのOTG_DIEPCTLxレジスタのEPDIS = 1
2. OTG_DIEPINTxのEPDIS割込みを待ちます。これはINエンドポイントが完全に無効にされたことを示します。EPDIS割込みがアサートされると、以下のビットがクリアされます。
   – OTG_DIEPCTLxレジスタのEPDIS = 0
   – OTG_DIEPCTLxレジスタのEPENA = 0
3. TxFIFOを一掃するために、次のビットをプログラムします。
   – OTG_GRSTCTLのTXFFLSH = 1
   – OTG_GRSTCTLのTXFNUM =「エンドポイントに固有のFIFO番号」
4. アプリケーションは、OTG_GRSTCTLのTXFFLSHがクリアされるまでポーリングを開始できます。このビットがクリアされると、Tx FIFOにデータが残っていないことを確認できます。

- **一般の非周期的INデータ転送**

アプリケーションの要件：

1. IN転送をセットアップする前に、アプリケーションは、IN転送の一部として送信されるすべてのデータが単一バッファの一部であることを確認する必要があります。
2. IN転送の場合、エンドポイント転送サイズレジスタの転送サイズフィールドは、最大パケットサイズの複数のパケットと単一のショートパケットから成るペイロードを表します。このショートパケットは、転送の最後に送信されます。
   – 転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには：
     転送サイズ [EPNUM] = *x* * MPSIZ[EPNUM] + sp
     （sp > 0）の場合、パケットカウント [EPNUM] = x + 1。
     そうでない場合、パケットカウント [EPNUM] = *x*
   – 単一のゼロ長データパケットを送信するには：
     転送サイズ [EPNUM] = 0
     パケットカウント [EPNUM] = 1
   – 転送の終わりに少数の最大パケットサイズのパケットとゼロ長データパケットを送信するには、アプリケーションは転送を2つの部分に分ける必要があります。最初の部分では最大パケットサイズのデータパケットを送信し、2番目の部分ではゼロ長データパケットのみを送信します。
     最初の転送：転送サイズ [EPNUM] = *x* * MPSIZ[epnum]；パケットカウント = *n*；
     2番目の転送：転送サイズ [EPNUM] = 0；パケットカウント = 1；
3. エンドポイントがデータ転送のために有効にされると、コアは、転送サイズレジスタを更新します。IN転送の終了時に、アプリケーションは、転送サイズレジスタを読み出して、送信FIFOにポストされたデータのうち、どれだけがUSBで送信されたかを確認する必要があります。
4. 送信FIFOにフェッチされたデータ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – USBで送信されたデータ = （アプリケーションがプログラムした初期パケットカウント – コアが更新した最終パケットカウント）* MPSIZ[EPNUM]
   – USB上でこれから送信されるデータ=（アプリケーションがプログラムした初期転送サイズ – USBで送信されたデータ）

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. アプリケーションは、要求されたデータをエンドポイントの送信FIFOに書き込む必要もあります。
3. アプリケーションによってパケットが送信FIFOに書き込まれるたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが0になるまで、データはアプリケーションによってメモリからフェッチされます。FIFOにデータを書き込んだ後、「FIFO内のパケット数」はインクリメントされます（これは3bitのカウントであり、各INエンドポイントの送信FIFOについて、コアによって内部で維持されます。INエンドポイントFIFO内にコアによって一度に維持されるパケットの最大数は8です）。ゼロ長のパケットの場合、各FIFOについて、FIFO内にデータがないことを示す個別のフラグがセットされます。
4. データが送信FIFOに書き込まれると、コアは、INトークンの受信時に読み出します。非アイソクロナスINデータパケットがACKハンドシェイクとともに送信されるたびに、エンドポイントのパケットカウントは、0になるまで1ずつデクリメントされます。パケットカウントは、タイムアウト時にはデクリメントされません。
5. ゼロ長パケットの場合（内部ゼロ長フラグで示されます）、コアはINトークンのゼロ長パケットを送出し、パケットカウントフィールドをデクリメントします。
6. 受信されたINトークンのデータがFIFOになく、そのエンドポイントのパケットカウントフィールドがゼロの場合、コアは、そのエンドポイントについて、「TxFIFOがエンプティのときINトークンが受信された」という割込み（ITTXFE割込み）を生成します。ただし、エンドポイントのNAKビットがセットされていない場合に限ります。コアは、USB上の非アイソクロナスエンドポイントに対してNAKハンドシェイクで応答します。
7. コアは、内部でFIFOポインタを巻き戻し、タイムアウト割込みは生成されません。
8. 転送サイズが0であり、パケットカウントが0のとき、エンドポイントの転送完了割込み（XFRC）が生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPTSIZxレジスタで転送サイズと対応するパケットカウントをプログラムします。
2. OTG_DIEPCTLxレジスタをエンドポイントの特性でプログラムし、CNAKビットおよびEPENA（エンドポイントイネーブル）ビットをセットします。
3. 非ゼロ長データパケットを送信するときには、アプリケーションは、OTG_ レジスタ（xはそのエンドポイントに関連するFIFO番号）をポーリングして、データFIFOに十分なスペースがあるかどうかを確認する必要があります。アプリケーションは、データを書き込む前に、オプションでTXFE（OTG_DIEPINTx）を使用することができます。

- **一般の周期的INデータ転送**

このセクションでは、典型的な周期的INデータ転送について説明します。

アプリケーションの要件：

1. 3237ページの一般の非周期的INデータ転送のアプリケーションの要件1、2、3、および4は、要件2が少し変更されていることを除いて、周期的INデータ転送にも適用されます。
   – アプリケーションは複数の最大パケットサイズのデータパケットの送信、または複数の最大パケットサイズのパケットに最後にショートパケットを加えた送信のみを行うことができます。転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには、以下の条件が満たされなければなりません。

   転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
   （*x* は整数 ≥ 0、および0 ≤ sp < MPSIZ[EPNUM]）

（sp > 0）の場合、パケットカウント [EPNUM] = *x* + 1
そうでない場合、パケットカウント [EPNUM] = *x* ;

MCNT[EPNUM] = パケットカウント [EPNUM]

– アプリケーションは転送の最後にゼロ長データパケットを送信することはできません。自分で単一のゼロ長パケットを送信することは可能です。単一のゼロ長データパケットを送信するには :
– 転送サイズ [EPNUM] = 0

パケットカウント [EPNUM] = 1

MCNT[EPNUM] = パケットカウント [EPNUM]

2. アプリケーションは一度に1フレームのデータ転送のみをスケジュールすることができます。
   – (MCNT – 1) × MPSIZ ≤ XFERSIZ ≤ MCNT × MPSIZ
   – PKTCNT = MCNT（OTG_DIEPTSIZxレジスタ）
   – XFERSIZ < MCNT * MPSIZの場合、転送の最後のデータパケットはショートパケットです。
   – 注 : MCNTはOTG_DIEPTSIZxレジスタに、MPSIZはOTG_DIEPCTLxレジスタに、PKTCNTはOTG_DIEPTSIZxレジスタに、またXFERSIZはOTG_DIEPTSIZxレジスタにあります。
3. フレームで送信される完全なデータは、INトークンが受信される前に、アプリケーションによって送信FIFOに書き込まれなければなりません。INトークンが受信されたときに、フレームあたりで送信されるデータの1ワードが送信FIFO内で欠落していても、コアはFIFOがエンプティである場合のように動作します。送信FIFOがエンプティのとき :
   – アイソクロナス IN エンドポイントに対して USB 上でゼロデータ長のパケットが送信されます。
   – INエンドポイントに割込みをかけるため、USB上でNAKハンドシェイクが送信されます。

内部データフロー :

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. また、アプリケーションは、要求されたデータをエンドポイントの関連する送信FIFOに書き込む必要があります。
3. アプリケーションがパケットを送信FIFOに書き込むたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが0になるまで、アプリケーションメモリからデータがフェッチされます。
4. 周期的エンドポイントのINトークンが受信されると、コアは、FIFOのデータ（ある場合）を送信します。フレームの完全なデータペイロード（専用FIFOモードでは完全なパケット）がFIFOに存在しない場合、コアは、エンドポイントに対するTx FIFOエンプティ割込み時にINトークンの受信を生成します。
   – アイソクロナスINエンドポイントに対して、USB上でゼロ長のデータパケットが送信されます。
   – INエンドポイントに割込みをかけるため、USB上でNAKハンドシェイクが送信されます。
5. エンドポイントのパケットカウントは、以下の条件のもとで1ずつデクリメントされます。
   – アイソクロナスエンドポイントに対して、ゼロ長または非ゼロ長データパケットが送信されたとき。
   – 割込みエンドポイントに対して、ACKハンドシェイクが送信されたとき。
   – 転送サイズとパケットカウントの両方が0のとき、エンドポイントに対して転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。
6. 「周期的フレームインターバル」（OTG_DCFGのPFIVLによって制御されます）で、コアが現在のフレームについてスケジュールされたアイソクロナスINエンドポイントFIFOのいずれかが非エンプティであることを検出すると、コアはOTG_GINTSTSのIISOIXFR割込みを生成します。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPCTLxレジスタをエンドポイントの特性でプログラムして、CNAKおよびEPENAビットをセットします。
2. 次のフレームで送信するデータを送信FIFOに書き込みます。
3. OTG_DIEPINTxレジスタのITTXFE割込みのアサートは、アプリケーションが送信するすべてのデータを送信FIFOにまだ書き込んでいないことを示します。
4. この割込みが検出されたときに、割込みエンドポイントがすでに有効だった場合、割込みは無視されます。まだ有効になっていなかった場合は、次のINトークン試行の際にデータを送信できるように、エンドポイントを有効にします。
5. OTG_DIEPINTxレジスタのITTXFE割込みがない状態でのXFRC割込み（OTG_DIEPINTx）のアサートは、アイソクロナスIN転送が正常に完了したことを示します。OTG_DIEPTSIZxレジスタの読出し結果は、転送サイズ = 0、かつパケットカウント = 0、すなわち、すべてのデータがUSBで送信されたことを示さなければなりません。
6. ITTXFE割込み（OTG_DIEPINTx）の有無にかかわらず、XFRC割込み（OTG_DIEPINTx）のアサートは、割込みIN転送が正常に完了したことを示します。OTG_DIEPTSIZxレジスタの読出し結果は、転送サイズ = 0、かつパケットカウント= 0、すなわち、すべてのデータがUSBで送信されたことを示さなければなりません。
7. 前述のどの割込みもない状態でのOTG_GINTSTSの不完全アイソクロナスIN転送（IISOIXFR）割込みのアサートは、コアが現在のフレームで少なくとも1つの周期的INトークンも受信しなかったことを示します。

- **不完全アイソクロナスINデータ転送**

このセクションでは、アプリケーションが不完全なアイソクロナスINデータ転送で行う必要があることについて説明します。

内部データフロー：

1. アイソクロナスIN転送は、以下の条件のいずれかがあてはまる場合に不完全として扱われます。
   a) コアが、少なくとも1つのアイソクロナスINエンドポイントで破損したアイソクロナスINトークンを受信した場合。この場合、アプリケーションは、不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFRビット）を検出します。
   b) アプリケーションが完全なデータペイロードを送信FIFOに書き込むのに時間がかかり、完全なデータペイロードがFIFOに書き込まれる前にINトークンが受信された場合。この場合、アプリケーションは、OTG_DIEPINTxのTx FIFOエンプティ割込み時にINトークンの受信を検出します。最終的には、周期的フレームの終わりでの不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFR）になるので、アプリケーションはこの割込みを無視することができます。

   コアは、受信したINトークンに応答して、USB上にゼロ長データパケットを送信します。
2. アプリケーションは、できるだけ速やかに、送信FIFOへのデータペイロードの書込みを停止する必要があります。
3. アプリケーションは、エンドポイントのNAKビットとディセーブルビットをセットする必要があります。
4. コアは、エンドポイントを無効にし、ディセーブルビットをクリアし、エンドポイントのエンドポイントディセーブル割込みをアサートします。

アプリケーションのプログラミングシーケンス：

1. 最終的には不完全アイソクロナスIN転送割込み（OTG_GINTSTS）になるので、アプリケーションは、アイソクロナスINエンドポイントでOTG_DIEPINTxのTx FIFOエンプティ割込み時に受信されたINトークンを無視することができます。
2. 不完全アイソクロナスIN転送割込み（OTG_GINTSTS）のアサートは、少なくとも1つのアイソクロナスINエンドポイントで不完全アイソクロナスIN転送があったことを示します。
3. アプリケーションは、すべてのアイソクロナスINエンドポイントのエンドポイント制御レジスタを読み出して、不完全INデータ転送があるエンドポイントを検出する必要があります。
4. アプリケーションは、AHB上でこれらのエンドポイントに関連する周期的送信FIFOへのデータの書込みを停止する必要があります。
5. OTG_DIEPCTLxレジスタの以下のフィールドをプログラムして、エンドポイントを無効にします。
   – OTG_DIEPCTLxレジスタのSNAK = 1
   – OTG_DIEPCTLxレジスタのEPDIS = 1
6. OTG_DIEPINTx のエンドポイントディセーブル割込みのアサートは、コアがエンドポイントを無効にしたことを示します。
   – この時点で、アプリケーションは、次のマイクロフレームで新しい転送を行うために、関連する送信FIFO内のデータを一掃するか、エンドポイントを有効にすることによって、FIFO内の既存のデータに上書きする必要があります。データを一掃するには、アプリケーションはOTG_GRSTCTLレジスタを使用する必要があります。

- **非アイソクロナスINエンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

アプリケーションのプログラミングシーケンス：

1. 停止するINエンドポイントを無効にします。STALLビットもセットします。
2. エンドポイントがすでに有効になっているときには、OTG_DIEPCTLxのEPDIS = 1にします。
   – OTG_DIEPCTLxレジスタのSTALL = 1にします。
   – STALLビットは常にNAKビットより優先されます。
3. エンドポイントディセーブル割込み（OTG_DIEPINTx）のアサートは、コアが指定されたエンドポイントを無効にしたことをアプリケーションに示します。
4. アプリケーションは、エンドポイントのタイプに応じて、非周期的または周期的FIFOを一掃する必要があります。非周期的エンドポイントの場合、アプリケーションは、データを送信するために、停止する必要がない他の非周期的エンドポイントを再び有効にする必要があります。
5. アプリケーションがエンドポイントのSTALLハンドシェイクを終了する準備ができたときには、OTG_DIEPCTLxのSTALLビットがクリアされなければなりません。
6. アプリケーションがSetFeature.Endpoint HaltコマンドまたはClearFeature.Endpoint Haltコマンドによってエンドポイントの STALLビットをセットまたはクリアする場合、STALLビットは、アプリケーションが制御エンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

特殊なケース：制御OUTエンドポイントの停止

制御転送のデータステージで、ホストがSETUPパケットで指定されたより多くのIN/OUTトークンを送信した場合、コアは、IN/OUTトークンを停止する必要があります。この場合、アプリケーションは、コアがSETUPパケットで指定されたデータ量を転送した後、制御転送のデータステージで、OTG_DIEPINTxのITTXFE割込みとOTG_DOEPINTxのOTEPDIS割込みを有効にしなければなりません。その後、アプリケーションがこの割込みを受信したとき、アプリケーションは、対応するエンドポイント制御レジスタのSTALLビットをセットし、この割込みをクリアする必要があります。

72.16.6 最悪ケースの応答時間

OTG_FS コントローラがデバイスとして機能するとき、アイソクロナスOUTに続く任意のトークンで、最悪ケースの応答時間があります。この最悪ケース応答時間は、AHBクロック周波数に依存します。

コアのレジスタはAHBドメインにあり、コアはこれらのレジスタを更新するまでは、別のトークンを受け入れません。アイソクロナストランザクションの場合、ハンドシェイクはなく、次のトークンがすぐに入ってくる可能性があるため、アイソクロナスOUTに続くトークンで最悪ケースが生じます。この最悪ケース値は、AHBクロックがPHYクロックと同じときには7 PHYクロックです。AHBクロックが高速なほど、この値は小さくなります。

この最悪ケース条件が発生した場合、コアは、バルク／割込みトークンにNAKで応答し、アイソクロナスおよびSETUP トークンをドロップします。ホストは、これをSETUPのタイムアウト条件と解釈して、SETUPパケットを再試行します。アイソクロナス転送の場合、不完全アイソクロナスIN転送割込み（IISOIXFR）と不完全アイソクロナスOUT転送割込み（IISOOXFR）は、アイソクロナスIN/OUTパケットがドロップされたことをアプリケーションに知らせます。

OTG_GUSBCFGのTRDTの値の選択

TRDT（OTG_GUSBCFG）の値は、MACがINトークンを受信した後、FIFOステータスとPFCブロックから最初のデータを取得するまでの時間（PHYクロック数）です。この時間には、PHYクロックとAHBクロック間の同期遅延も含まれます。この場合の最悪ケース遅延は、AHBクロックがPHYクロックと同じときに発生します。この場合、遅延は5クロックです。

MACがINトークンを受信すると、この情報（トークンの受信）はPFC（PFCはAHBクロックで動作します）によってAHBに同期されます。次に、PFCは、SPRAMからデータを読み出し、それらをデュアルクロックソースバッファに書き込みます。MACは、データをソースバッファ（深さ4）から読み出します。

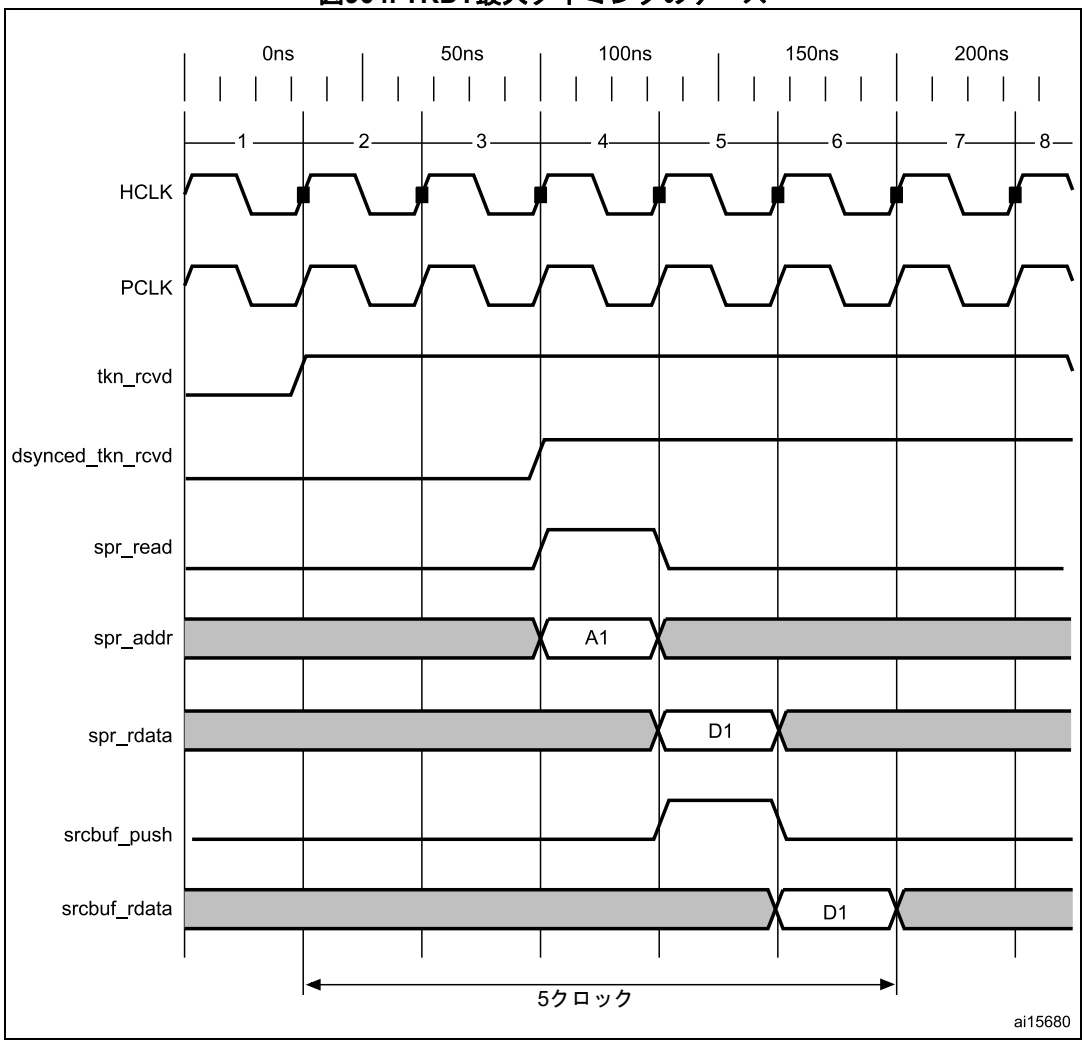
AHBがPHYより高い周波数で動作している場合、アプリケーションはTRDT（OTG_GUSBCFG）の値として、より小さな値を使用することができます。

図904に以下の信号を示します。

- tkn_rcvd：MACからPFCへの、トークンが受信されたという情報
- dynced_tkn_rcvd：PCLKからHCLKドメインへの二重同期tkn_rcvd
- spr_read：SPRAMへの読出し
- spr_addr：SPRAMのアドレス
- spr_rdata：SPRAMからのデータの読出し
- srcbuf_push：ソースバッファへのプッシュ
- srcbuf_rdata：ソースバッファからのデータの読出しMACによるデータの検出

TRDTの値を計算するには、表 749：TRDT値を参照してください。

図904. TRDT最大タイミングのケース



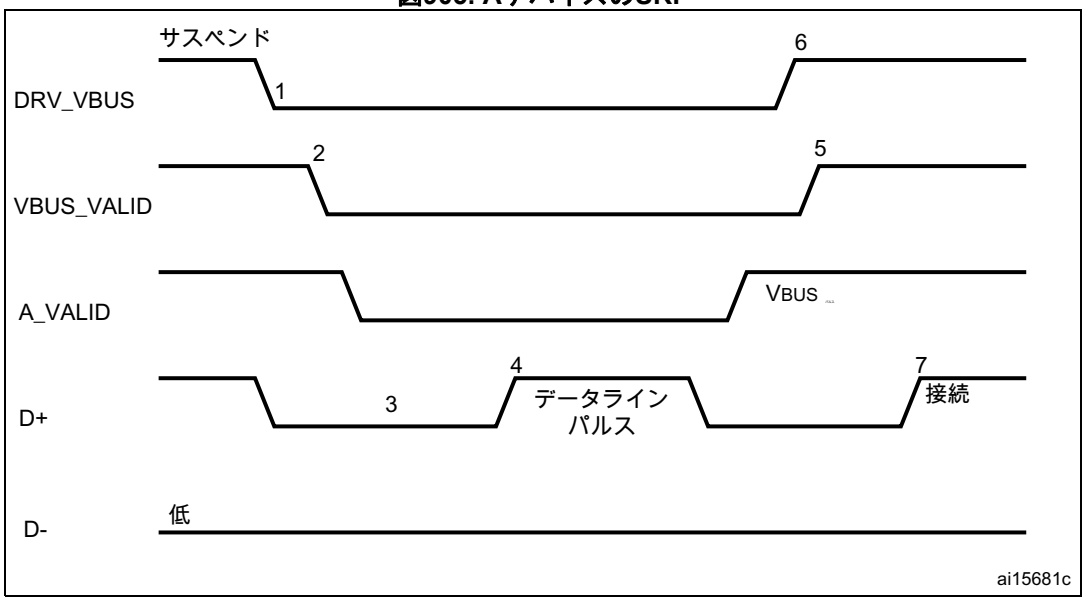
72.16.7 OTGプログラミングモデル

OTG_FS コントローラは、HNPとSRPをサポートするOTGデバイスです。コアがAプラグに接続されると、Aデバイスと呼ばれます。コアがBプラグに接続されると、Bデバイスと呼ばれます。ホストモードでは、OTG_FS コントローラは、電力を節約するためにV_{BUS} をオフにします。SPRは、BデバイスがAデバイスにV_{BUS} 電源をオンにするように信号を送る手段です。デバイスではデータラインパルスとV_{BUS} パルスの両方を実行する必要がありますが、ホストではSRPのデータラインパルスかV_{BUS} パルスのどちらかを検出することができます。HNPは、Bデバイスがネゴシエイトし、役割をホストに切り替える手段です。HNP後のネゴシエイトモードでは、Bデバイスはバスをサスペンドし、その役割をデバイスに戻します。

Aデバイスセッションリクエストプロトコル

アプリケーションは、コアUSB設定レジスタのSRP対応ビットをセットする必要があります。これによって、OTG_FS コントローラはSRPをAデバイスとして検出できるようになります。

図905. AデバイスのSRP



1. DRV_VBUS = PHYへのV_{BUS} 駆動信号
   VBUS_VALID = PHYからのV_{BUS} 有効信号
   A_VALID = PHYへのAペリフェラルV_{BUS} レベル信号
   D+ = データ正ライン
   D- = データ負ライン

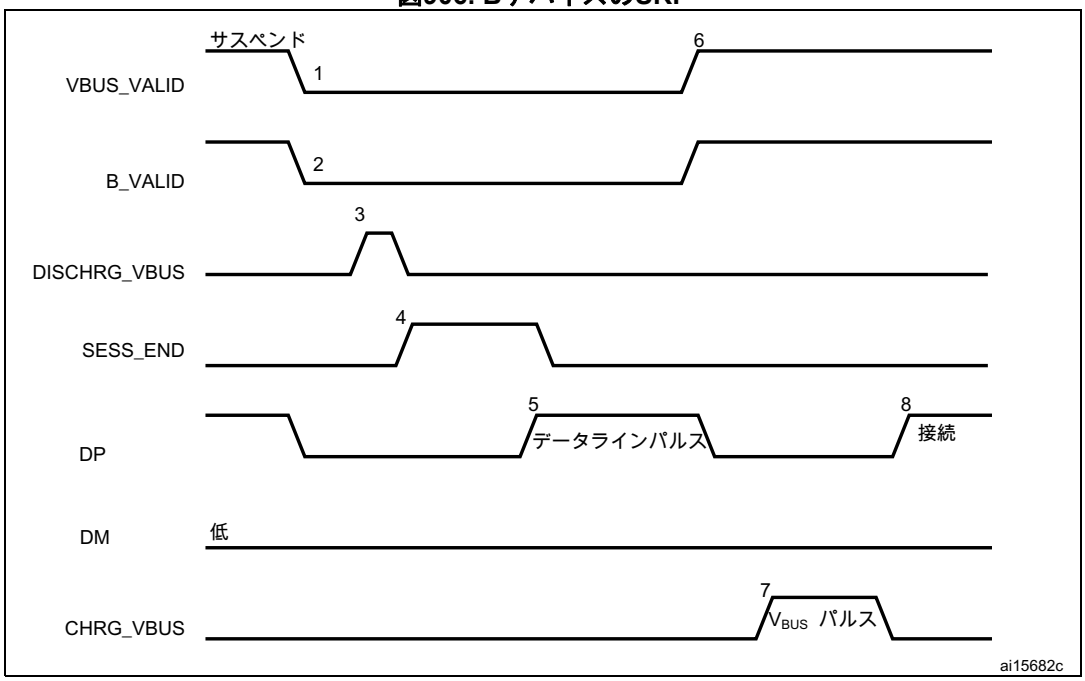
以下では、図905に示された信号計算法を記載しています。

1. 節電のためには、アプリケーションは、バスがアイドル状態のときに、ホストポート制御およびステータスレジスタのポートサスペンドビットおよびポートパワービットに書き込むことによって、ポート電源をサスペンドおよびオフにします。
2. PHYは、VBUS_VALID信号をネゲートすることによって、ポート電源のオフを示します。
3. デバイスは、V_{BUS} 電源がオフになると、SRPを開始するために少なくとも2 ms以上SE0を検出しなければなりません。
4. SRPを開始するには、デバイスはデータラインプルアップ抵抗を5 ~ 10 msの間オンにします。OTG_FS コントローラはデータラインのパルスを検出します。
5. デバイスは、V_{BUS} をAデバイスセッション有効レベル以上（最低2.0V）に駆動します（V_{BUS} パルスに対して）。
   OTG_FS コントローラは、SRPを検出すると、アプリケーションに割込みをかけます。グローバル割込みステータスレジスタのセッションリクエスト検出ビットがセットされます（OTG_GINTSTSのSRQINT）。
6. アプリケーションは、セッションリクエスト検出割込みを処理して、ホストポート制御およびステータスレジスタのポートパワービットに書き込むことによって、ポート電源をオンにする必要があります。PHYは、VBUS_VALID信号をアサートすることによって、ポート電源がオンになったことを示します。
7. USBに電源が投入されると、デバイスが接続され、SRPプロセスが完了します。

Bデバイスセッションリクエストプロトコル

アプリケーションは、コアUSB設定レジスタのSRP対応ビットをセットする必要があります。これによって、OTG_FS コントローラはSRPをBデバイスとして扱えるようになります。SRPは、OTG_FS コントローラがホストからの新しいセッションをリクエストする手段です。

図906. BデバイスのSRP



1. VBUS_VALID = PHYからのV_{BUS} 有効信号
   B_VALID = PHYへのBペリフェラル有効セッション信号
   DISCHRG_VBUS = PHYへの放電信号
   SESS_END = PHYへのセッション終了信号
   CHRG_VBUS = PHYへのV_{BUS} 充電信号
   DP = データ正ライン
   DM = データ負ライン

以下では、図906に示された信号計算法を記載しています。

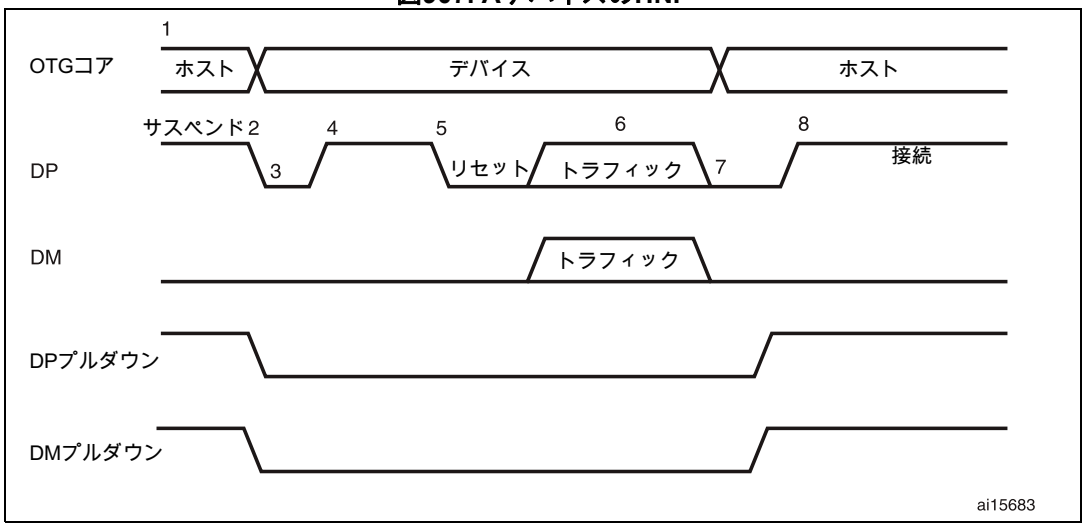
1. 節電のために、ホストはバスがアイドル状態のときはポート電源をサスペンドおよびオフにします。
   OTG_FS コントローラは、バスがアイドル状態になってから3 ms後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割込みレジスタのUSBサスペンドビットをセットします。
   OTG_FS コントローラは、PHYにV_{BUS} を放電するように通知します。
2. PHYはデバイスにセッションの終了を知らせます。これがSRPの初期条件です。OTG_FS コントローラは、SRPを開始する前に、2 msのSE0を必要とします。
   USB 1.1フルスピードシリアルトランシーバの場合、アプリケーションは、BSVLD（OTG_GOTGCTL）がネゲートされた後、V_{BUS} が0.2 Vまで放電するまで待つ必要があります。この放電時間の値は、トランシーバのベンダから入手でき、トランシーバごとに異なります。
3. OTG_FS コアはPHYにV_{BUS} の放電を加速させることを通知します。
4. アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエストビットに書き込むことによって、SRPを開始します。OTG_FS コントローラは、データラインパルスに続いてV_{BUS} パルスを実行します。
5. ホストは、データラインパルスまたはV_{BUS} パルスからSRPを検出して、V_{BUS} をオンにします。PHYは、デバイスにV_{BUS} 電源オンを知らせます。

6. OTG_FS コントローラは、V_{BUS} パルスを実行します。
ホストは、V_{BUS}をオンにして、SRPが成功したことを示すことによって、新しいセッションを開始します。OTG_FS コントローラは、OTG割込みステータスレジスタのセッションリクエスト成功ステータス変化ビットをセットすることによって、アプリケーションに割込みをかけます。アプリケーションは、OTG制御およびステータスレジスタのセッションリクエスト成功ビットを読み出します。
7. USBに電源が投入されると、OTG_FS コントローラが接続され、SRPプロセスが完了します。

Aデバイスホストネゴシエーションプロトコル

HNPは、USBホストの役割をAデバイスからBデバイスに切り替えます。アプリケーションは、OTG_FS コントローラがHNPをAデバイスとして実行できるように、コアUSB設定レジスタのHNP対応ビットをセットする必要があります。

図907. AデバイスのHNP



1. DPPULLDOWN = PHY内部のDPライン上でプルダウンを有効／無効にするためのコアからPHYへの信号
DMPULLDOWN = PHY内部のDMライン上でプルダウンを有効／無効にするためのコアからPHYへの信号

以下では、図907に示された信号計算法を記載しています。

1. OTG_FS コントローラは、SetFeature b_hnp_enable記述子をBデバイスに送信して、HNPサポートを有効にします。BデバイスのACK応答は、BデバイスがHNPをサポートしていることを示します。アプリケーションは、OTG制御およびステータスレジスタのホストセットHNPイネーブルビットをセットして、BデバイスがHNPをサポートしていることを OTG_FS コントローラに示す必要があります。
2. アプリケーションがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによって、サスペンドします。
3. BデバイスがUSBサスペンドを検出すると、切断して、HNPの初期条件を示します。Bデバイスは、ホストの役割に切り替える必要があるときにのみ、HNPを開始します。そうでないときには、バスはサスペンド状態を継続します。

   OTG_FS コントローラは、OTG割込みステータスレジスタのホストネゴシエーション検出割込みをセットして、HNPの開始を示します。

   OTG_FS コントローラは、PHYのDPプルダウンとDMプルダウンをネゲートして、デバイスの役割を示します。PHYは、OTG_DPプルアップ抵抗を有効にして、Bデバイスの接続を示します。

   アプリケーションは、OTG制御およびステータスレジスタの現在モードビットを読み出して、デバイスモードの動作を確認する必要があります。

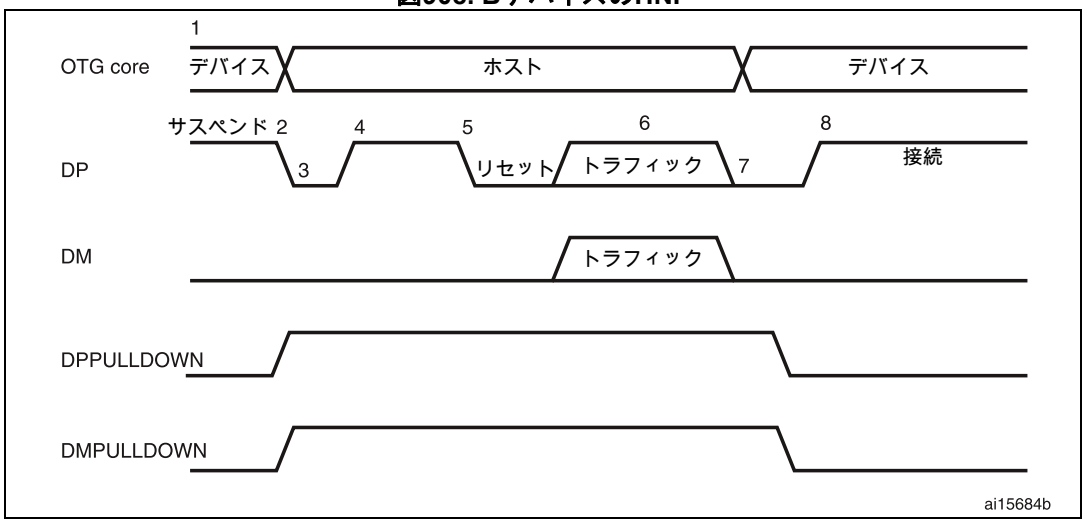
4. Bデバイスは接続を検出し、USBリセットを発行し、データトラフィックのために OTG_FS コントローラをエニュメレーションします。
5. Bデバイスはホストの役割を継続し、トラフィックを開始し、終了時にはバスをサスペンドします。

   OTG_FS コントローラは、バスがアイドル状態になってから3 ms後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割込みレジスタのUSBサスペンドビットをセットします。
6. ネゴシエートモードでは、OTG_FS コントローラは、サスペンドを検出して切断し、ホストの役割に戻ります。OTG_FS コントローラは、ホストの役割の前提を示すために、PHYのDPプルダウンとDMプルダウンをアサートします。
7. OTG_FS コントローラは、OTG割込みステータスレジスタのコネクタIDステータ変更割込みをセットします。アプリケーションは、OTG制御およびステータスレジスタのコネクタIDステータスを読み出して、OTG_FS コントローラの動作がAデバイスであることを確認する必要があります。これは、HNPの完了をアプリケーションに示します。アプリケーションは、OTG制御およびステータスレジスタの現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
8. Bデバイスが接続され、HNPプロセスが完了します。

Bデバイスホストネゴシエーションプロトコル

HNPは、USBホストの役割をBデバイスからAデバイスに切り替えます。アプリケーションは、OTG_FS コントローラがHNPをBデバイスとして実行できるように、コアUSB設定レジスタのHNP対応ビットをセットする必要があります。

図908. BデバイスのHNP



1. DPPULLDOWN = PHY内部のDPライン上でプルダウンを有効／無効にするためのコアからPHYへの信号
   DMPULLDOWN = PHY内部のDMライン上でプルダウンを有効／無効にするためのコアからPHYへの信号

以下では、図908に示された信号計算法を記載しています。

1. Aデバイスは、SetFeature　b_hnp_enable記述子を送信して、HNPサポートを有効にします。OTG_FS コントローラのACK応答は、それがHNPをサポートしていることを示します。アプリケーションは、OTG制御およびステータスレジスタのデバイスHNPイネーブルビットをセットしてHNPをサポートしていることを示す必要があります。

   アプリケーションは、OTG制御およびステータスレジスタのHNPリクエストビットをセットして、HNPを開始することを OTG_FS コントローラに示す必要があります。

2. Aデバイスがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってサスペンドします。

   OTG_FS コントローラは、バスがアイドル状態になってから3 ms後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割込みレジスタのUSBサスペンドビットをセットします。

   OTG_FS コントローラは切断され、AデバイスはバスでSE0を検出してHNPを示します。OTG_FS コントローラは、ホストの役割の前提を示すために、PHYのDPプルダウンとDPプルダウンをアサートします。

   Aデバイスは、SE0を検出してから3 ms以内にOTG_DPプルアップ抵抗を有効にすることによって応答します。OTG_FS コントローラはこれを接続として検出します。

   OTG_FS コントローラは、OTG割込みステータスレジスタのホストネゴシエーション成功ステータス変更割込みをセットして、HNPステータスを示します。アプリケーションは、OTG制御およびステータスレジスタのホストネゴシエーション成功ビットを読み出して、ホストネゴシエーションの成功を確認する必要があります。アプリケーションは、コア割込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。

3. アプリケーションはリセットビット（OTG_HPRTのPRST）をセットし、OTG_FS コントローラはUSBリセットを発行し、データトラフィックのためにAデバイスをエニュメレーションします。

4. OTG_FS コントローラは、トラフィックを開始するというホストの役割を継続し、これが終了すると、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってバスをサスペンドします。

5. ネゴシエートモードでは、Aデバイスは、サスペンドを検出すると、接続を切断し、ホストの役割に戻ります。OTG_FS コントローラは、PHYのDPプルダウンとDMプルダウンをネゲートして、デバイスの役割の前提を示します。

6. アプリケーションは、コア割込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。

7. OTG_FS コントローラが接続され、HNPプロセスが完了します。

73 USB On-The-Goハイスピード（OTG_HS）

このセクションは、STM32U59x/5Ax/5Fx/5Gxデバイスにのみ適用されます。

73.1 概要

Portions Copyright (c) Synopsys, Inc. All rights reserved.掲載許可取得済み。

このセクションでは OTG_HS コントローラのアーキテクチャとプログラミングモデルについて説明します。

このセクションでは、次の略記が使用されます。

FS	フルスピード
LS	ロースピード
HS	ハイスピード
MAC	メディアアクセスコントローラ
OTG	On-The-Go
PFC	パケットFIFOコントローラ
PHY	物理層
USB	ユニバーサルシリアル
UTMI	USB 2.0トランシーバマクロセルインタフェース（UTMI）
LPM	リンク電源管理
BCD	バッテリ充電検出回路
HNP	ホストネゴシエーションプロトコル
SRP	セッションリクエストプロトコル

このセクションでは、参照文献として以下の文書が使用されています。

- USB On-The-Go Supplement, Revision 2.0
- Universal Serial Bus Revision 2.0 Specification
- USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007
- USB 2.0 ECNの正誤表：リンク電源管理（LPM）- 2007年7月 (Errata for USB 2.0 ECN: Link Power Management (LPM) - 7/2007)
- バッテリ充電（BC）仕様Revision 1.2（Battery Charging Specification, Revision 1.2）

USB OTGは、デバイスとホストの両方の機能をサポートするデュアルロールデバイス（DRD）コントローラであり、On-The-Go Supplement to the USB 2.0 Specificationに完全に準拠しています。USB OTGは、USB 2.0 Specificationに完全に準拠したホスト専用またはデバイス専用のコントローラとして設定することも可能です。OTG_HS は、以下の表 752 : サポートされる OTG_HS の速度で定義されている速度に対応しています。必要な外部デバイスは、OTGモードで使用されるV_{BUS} 用のチャージポンプのみです。

表752.サポートされる OTG_HS の速度

-	HS（480 Mbit/s）	FS（12 Mbit/s）	LS（1.5 Mbit/s）
ホストモード	X	X	X
デバイスモード	X	X	-

73.2 OTG_HS の主な特徴

主な機能は、標準機能、ホストモード機能、およびデバイスモード機能の3つのカテゴリに分けて考えることができます。

73.2.1 標準機能

OTG_HS インタフェースの機能概要を次に示します。

- Universal Serial Bus Specification Rev 2.0に準拠したUSB-IFです。
- OTG_HS は、以下のPHYインタフェースをサポートします。
  - 内部HS PHY用のUTMI
- On-The-Go Supplement Rev 2.0 Specificationに詳細が規定されているオプションのOn-The-Go（OTG）プロトコルの完全サポート（PHY）を含みます。
  - A-Bデバイス認識（IDライン）をサポートしています。
  - OTGアプリケーションでバッテリを節約するために、ホストでV_{BUS} をオフすることができます。
  - 内部コンパレータによってOTGにV_{BUS} のレベルを監視させることができます。
  - ホスト-ペリフェラルのダイナミックな役割切り替えをサポートします。
- 次の動作を行うようにソフトウェアでの設定が可能です。
  - USB On-The-Goフルスピードデュアルロールデバイス
- 以下の要素によりHS SOFおよびLS Keep-aliveをサポートします。
  - SOFパルスPAD接続性
  - タイマ（TIMx）に対するSOFパルス内部接続
  - 設定可能なフレーム周期
  - 設定可能なフレームの終わりの割込み
- OTG_HS は、閾値の設定が可能な専用のDMAを内蔵し、DMAモードでAHBバーストタイプをソフトウェアで選択可能
- USBサスペンド中のシステム停止、デジタルコア、PHY、およびDFIFO電源管理回路内部のクロックドメインの電源オフなど、節電機能を内蔵しています。
- 以下のような高度なFIFO制御を備えた専用の 4KB RAMを備えています。
  - RAM空間を複数のFIFOに分割して、RAMを柔軟かつ効率的に使用できるように設定できます。
  - 各FIFOは複数のパケットを保持できます。
  - 動的なメモリ割り当てが可能です。
  - FIFOを2のべき乗でないサイズに設定して、連続的なメモリ位置の使用が可能です。
- システムの介入なしで、最大1フレーム（1ms）の間、最大USBバンド幅を保証します。
- バッテリ充電仕様Revision 1.2で説明されているとおり、充電ポート検出をサポートしています。

73.2.2 ホストモード機能

ホストモードでの OTG_HS インタフェースの主な機能と条件は次の通りです:

- V_{BUS} 電圧発生用の外部チャージポンプを持っています。
- 最大 16 個のホストチャネル（パイプ）を持っています。各チャネルはどんなタイプのUSB転送も割り当てられるように動的に設定可能です。
- 次を保持するハードウェアスケスケジューラを内蔵しています。
  - 周期的ハードウェアキュー内で、最大16のインタラプト及びアイソクロナス転送リクエスト
  - 非周期的ハードウェアキュー内で、最大16のコントロール転送およびバルク転送リクエスト
- USBデータRAMの効率的な使用のために、共有Rx FIFO、周期的Tx FIFO、および非周期的Tx FIFOを管理します。

73.2.3 ペリフェラルモード機能

ペリフェラルモードでの OTG_HS インタフェースの主な機能は次のとおりです。

- 1つの双方向制御エンドポイント0を持っています。
- バルク、インタラプト、またはアイソクロナス転送をサポートするために、設定可能な8つのINエンドポイント（EP）を持っています。
- バルク、インタラプト、またはアイソクロナス転送をサポートするために、設定可能な 8 つのOUTエンドポイント（EP）を持っています。
- USBデータRAMの効率的な使用のために、共有RX FIFOとTx-OUT FIFOを管理します。
- アプリケーションの負荷を軽減するために、最大 9 つの専用Tx-IN FIFO（有効なIN EPに対して1つずつ）を管理します。
- ソフト切断機能をサポートしています。

73.3 OTG_HS の実装

表753.OTG_HS の実装(1)

USBの機能	STM32U59x/5Ax/5Fx/5Gx用のOTG_HS
デバイスの双方向エンドポイント（EP0を含む）	9
ホストモードチャネル	16
専用SRAMのサイズ	4KB
USB 2.0リンク電源管理（LPM）のサポート	X
サポートされているOTGレビジョン	2.0
バッテリ充電検出（BCD）のサポート	X
統合PHY	HS
HNPおよびSRPサポート	-
SRP：OTG_DVBUSDISレジスタ	-
SRP：OTG_DVBUSPULSEレジスタ	-
HNP：OTG_GOTGCTLビット11:8	-
SRP：OTG_GOTGCTLビット1:0	-
HNP/SRP：OTG_GOTGINTビット19、17、9:8	-
HNP：OTG_GUSBCFGビット9	-

表753.OTG_HS の実装(1) （続き）

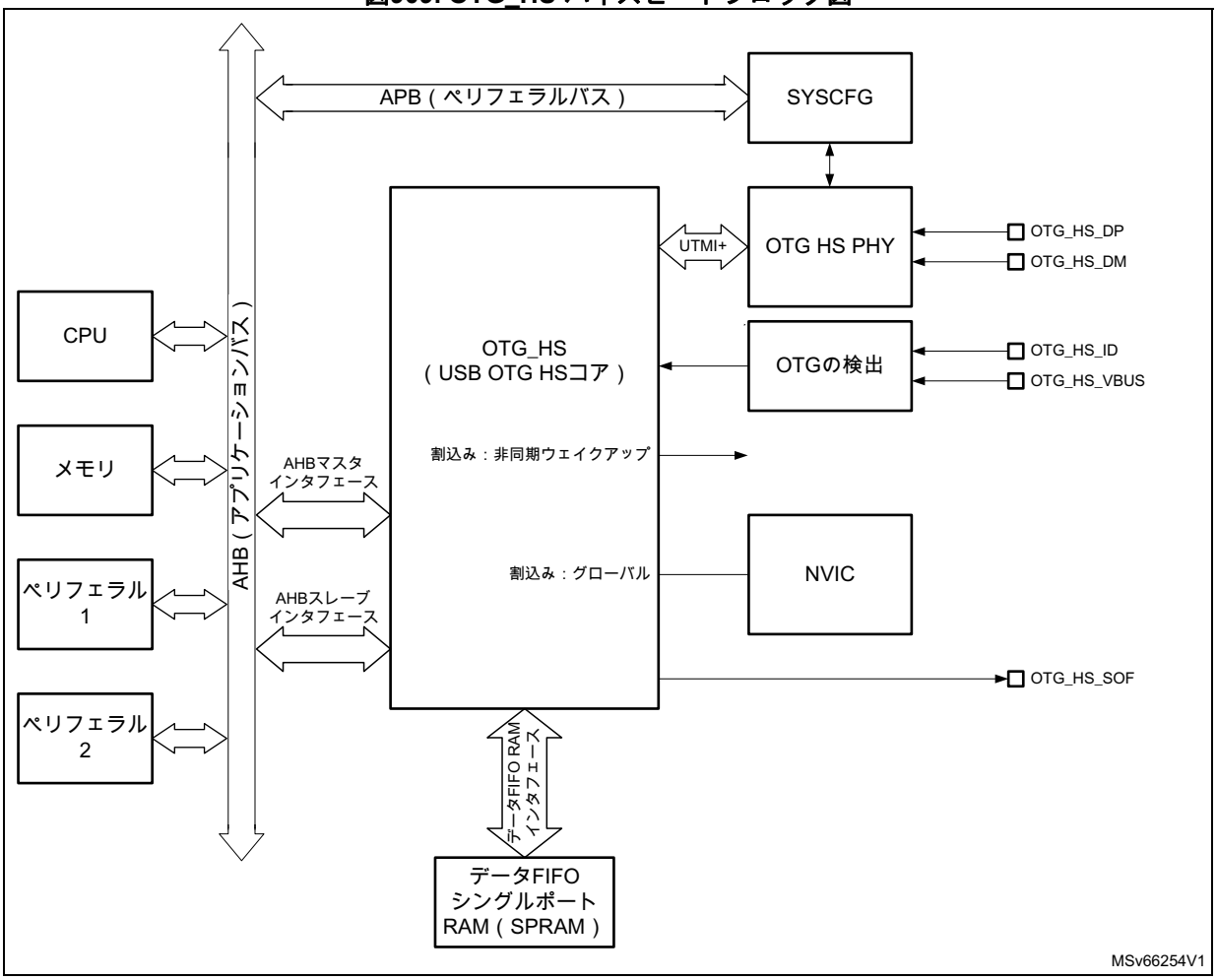
USBの機能	STM32U59x/5Ax/5Fx/5Gx用のOTG_HS
SRP：OTG_GUSBCFGビット8	-
OTG_PCGCCTL1レジスタ	X

1. 「X」はサポート対象、「-」はサポート対象外です。「FS」はFSモードでサポートされ、「HS」はHSモードでサポートされます。

73.4 OTG_HS 機能詳細

73.4.1 OTG_HS ブロック図

図909. OTG_HS ハイスピードブロック図



73.4.2 OTG_HS ピンおよび内部信号

表754.OTG_HS の入出力ピン

信号名	信号タイプ	説明
OTG_HS_DP	デジタル入出力	USB OTG D+ ライン
OTG_HS_DM	デジタル入出力	USB OTG D- ライン
OTG_HS_ID	デジタル入力	USB OTG ID
OTG_HS_VBUS	アナログ入力	USB OTG VBUS
OTG_HS_SOF	デジタル出力	USB OTGフレーム開始（可視性）

表755.OTG_HS 入出力信号

信号名	信号タイプ	説明
usb_sof	デジタル出力	オンチップペリフェラルのUSB OTGフレーム開始イベント
usb_wkup	デジタル出力	USB OTG ウェイクアップイベント出力
usb_gbl_it	デジタル出力	USB OTGグローバル割込み

73.4.3 OTG_HS コア

OTG_HS は、リセットおよびクロックコントローラ（RCC）から60 MHzのクロックを受け取ります。これは通常、HS PHYと関連するPLLで生成され、RCCで有効化されます。このクロックは、ハイスピード（480 Mbit/s）で60 MHzドメインを駆動するために使用され、OTGコアを設定する前に有効にされなければなりません。

CPUはAHBペリフェラルバスを通じてOTG Coreレジスタの読み書きを行います。これは、セクション 73.12：OTG_HS割込みで説明されている単一のUSB OTG割込みを介してUSBイベントとして通知されます。

CPUは、専用のOTG位置（プッシュレジスタ）に32bitのワードを書き込むことによって、USBを介してデータを提供します。データはUSBデータRAM内に設定されたTxデータFIFOに自動的に格納されます。各INエンドポイント（ペリフェラルモードの場合）またはOUTチャネル（ホストモードの場合）に対して1つのTx FIFOプッシュレジスタがあります。

CPUは、専用のOTGアドレス（ポップレジスタ）から32bitのワードを読み出すことによって、USBからデータを受け取ります。データは、4KBのUSBデータRAM内に設定された共有Rx FIFOから自動的に取得されます。各OUTエンドポイントまたはINチャネルに対して1つのRx FIFOポップレジスタがあります。

USBプロトコル層は、シリアルインタフェースエンジン（SIE）によって駆動され、オンチップ物理層（PHY）内のトランシーバモジュールによってUSB上で直列化（シリアライズ）されます。

注意： **USB OTG_HS ペリフェラルの正常動作を確保するためにはAHB周波数を30MHzより高くする必要があります。**

73.4.4 OTGの検出

また、OTG_HS では次の機能を使用します。

- A/B デバイス識別のための ID ラインをサンプリングするために使用される組み込みの ID プルアップ抵抗。
- V_{BUS} レベル検出用コンパレータ。V_{BUS} の有効性、A-Bセッションの有効性、およびセッション終了電圧閾値を検出するために使用されるヒステリシスを有します。これらは、有効な起動およびセッション終了条件の検出、およびUSB操作中のV_{BUS} 電源の定常的な監視のために使用されます。

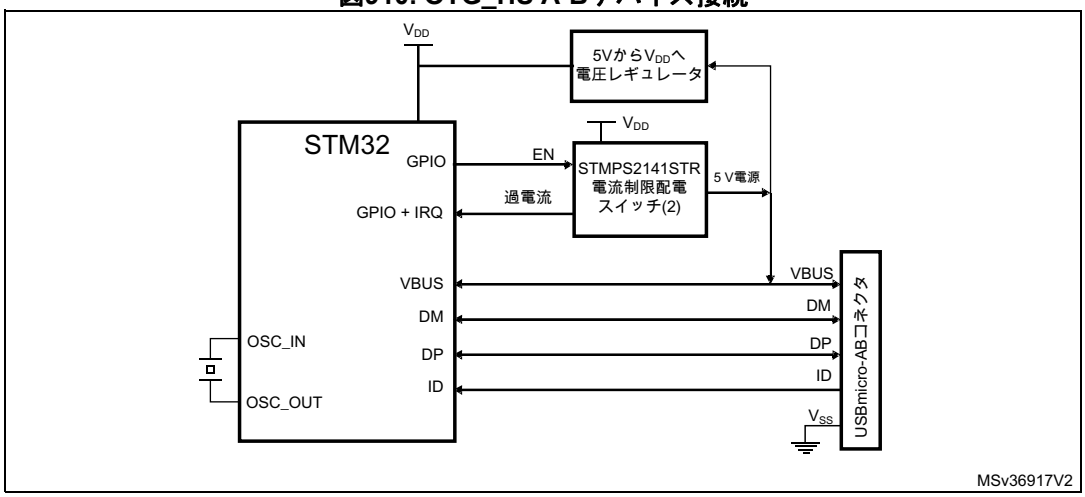
73.4.5 OTG-FSに接続されたハイスピードOTG PHY

注： **HS PHYが内蔵されているかどうかを確認するには、実装表を参照してください。**

USB OTGコアには、内蔵HS PHYに接続される内部UTMIインタフェースがあります（セクション 73.4.1：OTG_HS ブロック図を参照）。

73.5 OTG_HS デュアルロールデバイス（DRD）

図910. OTG_HS A-Bデバイス接続



1. VBUSで電源が供給されるデバイスを設定するときだけ必要な外部電圧レギュレータ
2. アプリケーションが VBUS で電源が供給されるデバイスをサポートしなければならない場合のみ必要な STMPS2141STR。アプリケーションボード上で5 Vが使用可能な場合、基本的なパワースイッチを使用することができます。

73.5.1 IDラインの検出

ホストかペリフェラル（デフォルト）かの役割は、ID入力ピンに応じて割り当てられます。IDラインのステータスは、USBケーブルでの挿入時に、MicroAまたはMicroBプラグがマイクロABレセプタクルに接続されたかによって決まります。

- USBケーブルのB側がフローティングIDワイヤに接続された場合、組み込みのプルアップ抵抗がIDラインのハイレベルを検出して、デフォルトのペリフェラルの役割が確立されます。この構成では、OTG_HS はセクション4.2.4で説明する標準FSM、すなわちOn-The-Go仕様Rev 2.0のIDピン（USB2.0への補完）に準拠しています。
- USBケーブルのA側のIDがグランドに接続された場合、OTG_HS は、ホストのソフトウェア初期化のためにIDラインステータス変化割込み（OTG_GINTSTSのCIDSCHGビット）を発行して、自動的にホストの役割に切り替わります。この構成では、OTG_HS はセクション4.2.4で説明する標準FSM、すなわちOn-The-Go仕様Rev 2.0のIDピン（USB2.0への補完）に準拠しています。

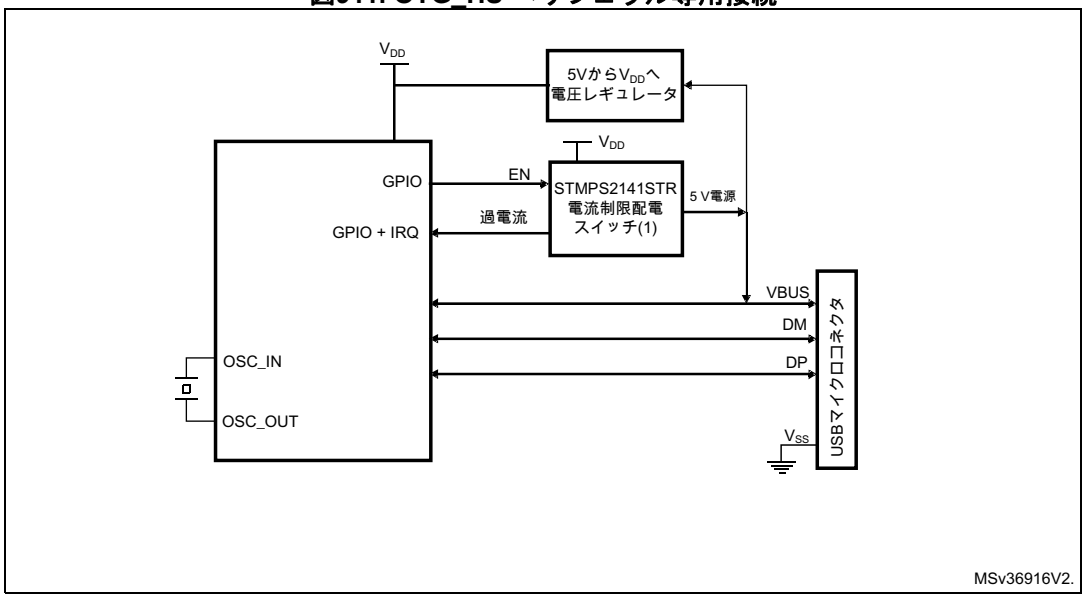
73.6 USBペリフェラルとしての OTG_HS

このセクションでは、USBペリフェラルモードでの OTG_HS の機能について説明します。OTG_HS は、以下の環境でUSBペリフェラルとして機能します。

- OTG Bペリフェラル
  - USBケーブルのB側が挿入された場合のOTG Bデバイスのデフォルトの状態
- Bデバイス
  - IDラインが存在し、機能していて、USBケーブルのB側に接続されている場合。
- ペリフェラルのみ
  - セクション 73.14.4：OTG USB設定レジスタ（OTG_GUSBCFG）の強制デバイスモードビット（FDMOD）に1がセットされると、OTG_HS コアはUSBペリフェラル専用として機能するよう強制されます。この場合、IDラインは、USBコネクタ上に存在していても無視されます。

注： **B デバイスまたはペリフェラル専用設定の場合に電源がバスから供給されるデバイス (bus-powered device)を実装するには、V_{BUS} から必要な電力を生成する外部レギュレータを追加する必要があります。**

図911. OTG_HS ペリフェラル専用接続



1. バスから電源が供給されるデバイスを構築するには、レギュレータを使用します。

73.6.1 ペリフェラルの状態

電源が供給された状態

V_{BUS} 入力は、USBペリフェラルが電源供給状態に入ることが許されるBセッション有効電圧を検出します（USB 2.0のセクション9.1を参照）。その場合、OTG_HS は、DPプルアップ抵抗を自動的に接続して、フルスピードデバイス接続をホストに知らせ、セッションリクエスト割込みを生成して（OTG_GINTSTSのSRQINTビット）、電源供給状態を通知します。

V_{BUS} 入力は、また、USB動作中にホストによって適切なV_{BUS} レベルが供給されることを保証します。V_{BUS} 電源がBセッションの有効値以下に低下したことが検出された場合（たとえば、電源の変動があった場合や、ホストポートの電源がオフとなった場合など）、OTG_HS は自動的に切断され、セッション終了検出割込み（OTG_GOTGINTのSEDETビット）が生成されて、OTG_HS が電源供給状態でなくなったことを通知します。

電源供給状態では、OTG_HS は、ホストからリセットシグナリングを受け取ることを期待しています。他のUSB操作を実行することはできません。リセットシグナリングを受け取ると、リセット検出割込み（OTG_GINTSTSのUSBRSTビット）が生成されます。リセットシグナリングが完了すると、エミュレーション終了割込み（OTG_GINTSTSのENUMDNEビット）が生成され、OTG_HS はデフォルト状態に入ります。

ソフト切断

電源供給状態は、ソフト切断機能によってソフトウェアで終了させることができます。DPプルアップ抵抗は、デバイス制御レジスタのソフト切断ビット（OTG_DCTLのSDISビット）をセットすることによって取り除かれ、USBケーブルが実際にはホストポートから取り除かれていなくても、ホスト側にデバイス切断検出割込みを生成します。

デフォルト状態

デフォルト状態では、OTG_HS はホストからSET_ADDRESSコマンドを受け取ることを期待しています。他のUSB操作を実行することはできません。有効なSET_ADDRESSコマンドがUSB上でデコードされると、アプリケーションは対応する番号をデバイス設定レジスタのデバイスアドレスフィールドに書き込みます（OTG_DCFGのDADビット）。その場合、OTG_HS はアドレス状態に入り、設定されたUSBアドレスでホストトランザクションに応答する準備が整います。

サスペンド状態

OTG_HS ペリフェラルは、USBの動作状態を定常的に監視します。USBのアイドル状態が3 ms続くと、アーリーサスペンド割込み（OTG_GINTSTSのESUSPビット）が発行され、適切な場合は3 ms後にサスペンド割込み（OTG_GINTSTSのUSBSUSPビット）によって確認されます。その場合、デバイスステータスレジスタのデバイスサスペンドビット（OTG_DSTSのSUSPSTSビット）が自動的にセットされ、OTG_HS はサスペンド状態に入ります。

サスペンド状態は、デバイス自身によって終了することもできます。この場合、アプリケーションはデバイス制御レジスタのリモートウェイクアップシグナリングビット（OTG_DCTLのRWUSIGビット）をセットし、1 ~ 15 ms後にクリアします。

ホストからレジュームシグナリングが検出されると、レジューム割込み（OTG_GINTSTSのWKUPINTビット）が生成され、デバイスサスペンドビットは自動的にクリアされます。

73.6.2 ペリフェラルエンドポイント

OTG_HS コアは、次のUSBエンドポイントをインスタンス化します。

- コントロールエンドポイント0 :
  - 双方向であり、制御メッセージのみを取り扱います。
  - 入力および出力トランザクションを処理する個別のレジスタセットを備えています。
  - 専用の制御レジスタ（OTG_DIEPCTL0/OTG_DOEPCTL0）、転送設定レジスタ（OTG_DIEPTSIZ0/OTG_DOEPTSIZ0）、およびステータス割込みレジスタ（OTG_DIEPINT0/OTG_DOEPINT0）を備えています。制御および転送サイズレジスタ内で使用可能なビットのセットは、他のエンドポイントとは少し違います。

- 8つのINエンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、またはインタラプト転送タイプをサポートするように設定できます。
  - それぞれ、専用の制御レジスタ（OTG_DIEPCTL*x*）、転送設定レジスタ（OTG_DIEPTSIZ*x*）、およびステータス割込みレジスタ（OTG_DIEPINT*x*）を備えています。
  - デバイスINエンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）を使用して、すべてのINエンドポイント（EP0も含む）上で1種類のエンドポイント割込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFRビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTSのEOPF）の終わりとともにアサートされます。
- 8つのOUTエンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、またはインタラプト転送タイプをサポートするように設定できます。
  - それぞれ、専用の制御レジスタ（OTG_DOEPCTL*x*）、転送設定レジスタ（OTG_DOEPTSIZ*x*）、およびステータス割込みレジスタ（OTG_DOEPINT*x*）を備えています。
  - デバイスOUTエンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）を使用して、すべてのOUTエンドポイント（EP0も含む）上で1種類のエンドポイント割込みソースを有効／無効にすることができます。
  - 現在のフレーム内で転送が完了していないアイソクロナスOUTエンドポイントが少なくとも1つあるときにアサートされる、不完全アイソクロナスOUT転送割込み（OTG_GINTSTSのINCOMPISOOUTビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTSのEOPF）の終わりとともにアサートされます。

エンドポイント制御

- 以下のエンドポイント制御は、デバイスエンドポイント-*x* IN/OUT制御レジスタ（OTG_DIEPCTL*x*/OTG_DOEPCTL*x*）を通じて、アプリケーションから使用できます。
  - エンドポイントの有効化／無効化
  - 現在の設定でのエンドポイントのアクティブ化
  - USB転送タイプ（アイソクロナス、バルク、インタラプト）のプログラミング
  - サポートされるパケットサイズのプログラミング
  - INエンドポイントに関連するTx FIFO番号のプログラミング
  - 期待される、または送信されるデータ0／データ1 PID（バルク／インタラプトのみ）のプログラミング
  - トランザクションが送受信される偶数／奇数フレームのプログラミング（アイソクロナス転送のみ）
  - オプションで、FIFOのステータスに関係なく、常にホストに否定応答するNAKビットのプログラミング
  - オプションで、そのエンドポイントに対するホストトークンを常に停止するSTALLビットのプログラミング
  - オプションで、OUTエンドポイントが受信データのCRCフィールドをチェックしないようにするSNOOPモードのプログラミング

エンドポイント転送

アプリケーションは、デバイスエンドポイント-*x* 転送サイズレジスタ（OTG_DIEPTSIZ*x*/OTG_DOEPTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。エンドポイント制御レジスタのエンドポイントイネーブルビットをセットする前にプログラミングを行う必要があります。エンドポイントが有効になると、これらのフィールドは、OTG_HS コアが現在の転送ステータスで更新すると、読出し専用になります。

以下の転送パラメータがプログラム可能です。

- 転送サイズ（バイト単位）
- 全体の転送サイズを構成するパケットの数

エンドポイントのステータス／割込み

デバイスエンドポイント-*x* 割込みレジスタ（OTG_DIEPINT*x*/OTG_DOPEPINT*x*）は、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタのOUTエンドポイント割込みビットまたはINエンドポイント割込みビット（OTG_GINTSTSのOEPINTビット、またはOTG_GINTSTSのIEPINTビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、デバイスオールエンドポイント割込み（OTG_DAINT）レジスタを読み出して、デバイスエンドポイント-*x* 割込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

ペリフェラルコアは、以下のステータスチェックおよび割込み生成を行います。

- データ転送がアプリケーション側（AHB）とUSB側の両方で完了したことを示す転送完了割込み
- セットアップステージが終了しているかどうか（コントロールOUTのみ）
- 関連する送信FIFOが半エンプティか完全エンプティか（INエンドポイント）
- NAK応答がホストに送信されたかどうか（アイソクロナスINのみ）
- Tx FIFOがエンプティのときINトークンが受け取られたかどうか（バルクIN/インタラプトINのみ）
- エンドポイントがまだ有効になっていないときにOUTトークンを受け取ったかどうか
- バブルエラー条件が検出されたかどうか
- アプリケーションによるエンドポイントの無効化が可能かどうか
- アプリケーションによるエンドポイントNAKが可能かどうか（アイソクロナスINのみ）
- 3つを超える連続したセットアップパケットが受信されたかどうか（コントロールOUTのみ）
- タイムアウト条件が検出されたかどうか（コントロールINのみ）
- アイソクロナスOUTパケットが割込みを生成せずにドロップされたかどうか

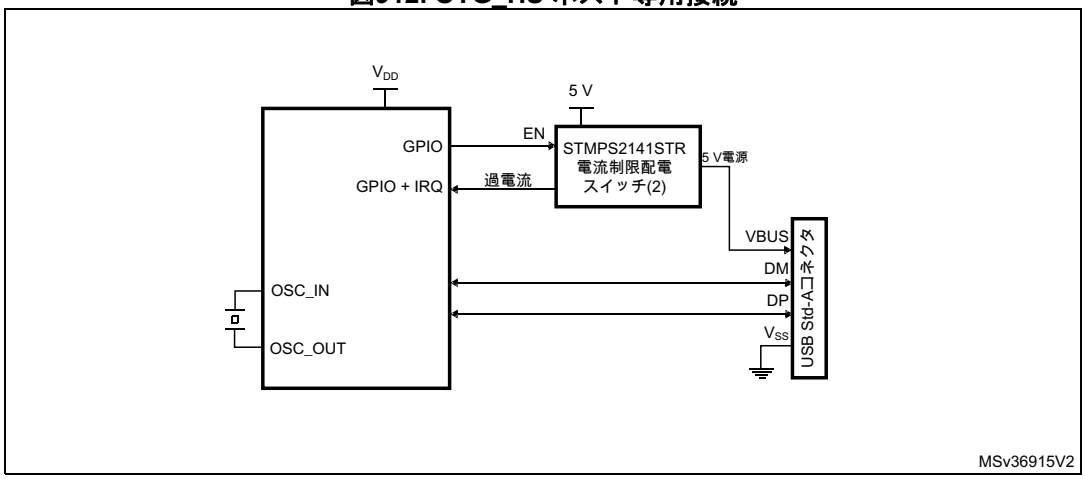
73.7 USBホストとしての OTG_HS

このセクションでは、USBホストモードでの OTG_HS の機能について説明します。OTG_HS は次の環境でUSBホストとして機能します。

- OTG Aホスト
  - USBケーブルのA側が挿入されたときのOTG Aデバイスのデフォルト状態
- Aデバイス
  - IDラインが存在し、機能していて、USBケーブルのA側に接続されている場合。組み込みのプルダウン抵抗が自動的にDP/DMライン上にセットされます。
- ホストのみ
  - OTG USB設定レジスタ（OTG_GUSBCFG）の強制ホストモードビット（FHMOD）は、OTG_HS コアをUSBホスト専用として機能するよう強制します。この場合、IDラインは、USBコネクタ上に存在していても無視されます。組み込みのプルダウン抵抗が自動的にDP/DMライン上にセットされます。

注： **オンチップの5V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または5 Vがアプリケーションボード上で使用可能な場合は基本的なパワースイッチを外部で追加して、5V V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意のGPIO出力からでも駆動できます。これは、OTG Aホスト、Aデバイス、およびホスト専用設定の場合に必要です。**

図912. OTG_HS ホスト専用接続



1. V_{DD} 範囲は2.0～3.6 Vです。

73.7.1 USBホストの状態

ホストポートパワー

オンチップの5V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または5 Vがアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 VのV_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意のGPIO出力または外部PMIC（電源管理IC）に接続されたI^2Cインタフェースからでも駆動できます。アプリケーションがV_{BUS} に電源を供給することにした場合は、ホストポート制御およびステータスレジスタのポートパワービット（OTG_HPRTのPPWRビット）もセットする必要があります。

有効なV_{BUS}

ホストモードでは、VBUSセンシングピンをV_{BUS} に接続する必要はありません。

チャージポンプの過電流フラグは、電気的ダメージを防止するためにも使用できます。チャージポンプからの過電流フラグ出力を任意のGPIO入力に接続して、有効レベルになったときにポート割込みを生成するように設定します。過電流ISRは迅速にV_{BUS} の発生を無効にし、ポートパワービットをクリアする必要があります。

ペリフェラル接続のホスト検出

USBペリフェラルまたはBデバイスは、接続されるとすぐに検出されます。OTG_HS コアは、ホストポート制御およびステータス内のデバイス接続ビット（OTG_HPRTのPCDETビット）から発生するホストポート割込みを発行します。

ペリフェラル切断のホスト検出

ペリフェラル切断イベントは、切断検出割込み（OTG_GINTSTSのDISCINTビット）をトリガします。

ホストのエニュメレーション

ペリフェラル接続を検出した後、ホストは、USBリセットおよび設定コマンドを新しいペリフェラルに送信することによって、エニュメレーションプロセスを開始しなければなりません。

アプリケーションは、ホストポート制御およびステータスレジスタのポートリセットビット（OTG_HPRTのPRSTビット）を最短10ms、最長20msの間セット状態に保つことによって、USBを介したUSBリセットシグナリング（シングルエンド、ゼロ）を駆動します。アプリケーションは、タイミングカウントをチェックした後、ポートリセットビットがクリアされているかどうかを確認します。

USBリセットシーケンスが完了すると、ポートイネーブル／ディセーブル変更ビット（OTG_HPRTのPENCHNGビット）によってホストポート割込みがトリガされます。これにより、エニュメレーションされたペリフェラルの速度をホストポート制御およびステータスレジスタのポート速度フィールド（OTG_HPRTのPSPDビット）から読み出せることと、ホストがSOF（FS）またはキープアライブ（LS）の駆動を開始しようとしていることがアプリケーションに通知されます。これで、ホストは、ペリフェラル設定コマンドを送信することによってペリフェラルエニュメレーションを完了する準備ができたことになります。

ホストのサスペンド

アプリケーションは、ホストポート制御およびステータスレジスタのポートサスペンドビット（OTG_HPRTのPSUSPビット）をセットすることによって、USBの活動をサスペンド（一時停止）します。OTG_HS コアはSOFの送信を停止し、サスペンド状態に入ります。

サスペンド状態は、リモートデバイスが主導して（リモートウェイクアップ）、終了することもできます。この場合、リモートウェイクアップシグナリング、ホストポート制御およびステータスレジスタのポートレジュームビット（OTG_HPRTのPRESビット）セルフセットが検出されると、リモートウェイクアップ割込み（OTG_GINTSTSのWKUPINTビット）が生成され、レジュームシグナリングがUSB上で自動的に駆動されます。アプリケーションは、レジュームウィンドウの時間を決めてから、ポートレジュームビットをクリアしてサスペンド状態を終了し、SOFをリスタートする必要があります。

サスペンド状態をホスト主導で終了した場合、アプリケーションは、ポートレジュームビットをセットして、ホストポートでのシグナリングのレジュームを開始し、レジュームウィンドウの時間を決め、最後に、ポートレジュームビットをクリアする必要があります。

73.7.2 ホストチャネル

OTG_HS コアは、16 のホストチャネルを設定可能です。各ホストチャネルはUSBホスト転送をサポートします（USBパイプ）。ホストは、同時に 16 つを超える転送リクエストをサポートすることはできません。アプリケーションからの 16 つを超える転送リクエストがペンディング中の場合、ホストコントローラドライバ（HCD）は、チャネルが以前の処理から使用可能になったとき、すなわち、転送完了およびチャネルホールト割込みの受信後、チャネルの再割り当てを行う必要があります。

各ホストチャネルは、入力／出力および任意のタイプの周期的/非周期的トランザクションをサポートするように設定できます。各ホストチャネルは、専用の制御レジスタ（OTG_HCCHAR*x*）、転送設定レジスタ（OTG_HCTSIZ*x*）、およびステータス／割込みレジスタ（OTG_HCINT*x*）と関連するマスクレジスタ（OTG_HCINTMSK*x*）を使用します。

ホストチャネル制御

- アプリケーションは、ホストチャネル *x* 特性レジスタ（OTG_HCCHAR*x*）を通じて、以下のホストチャネル制御を使用できます。
  - チャネルの有効化／無効化
  - ターゲットUSBペリフェラルのHS/FS/LS速度のプログラミング
  - ターゲットUSBペリフェラルのアドレスのプログラミング
  - ターゲットUSBペリフェラルのエンドポイント数のプログラミング
  - IN/OUT転送方向のプログラミング
  - USB転送タイプ（コントロール、バルク、インタラプト、アイソクロナス）のプログラミング
  - 最大パケットサイズ（MPS）のプログラミング
  - 奇数／偶数フレームで実行される周期的転送のプログラミング

ホストチャネル転送

アプリケーションは、ホストチャネル転送サイズレジスタ（OTG_HCTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。プログラミングは、ホストチャネル特性レジスタのチャネルイネーブルビットをセットする前に行う必要があります。エンドポイントが有効になると OTG_HS コアが現在の転送ステータスに従って更新するので、パケットカウントフィールドは読出し専用になります。

- 以下の転送パラメータがプログラム可能です。
  - 転送サイズ（バイト単位）
  - 全体の転送サイズを構成するパケットの数
  - 初期データPID

ホストチャネルのステータス／割込み

ホストチャネル *x* 割込みレジスタ（OTG_HCINT*x*）は、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、ホストオールチャネル割込み（OTG_HAINT）レジスタを読み出して、ホストチャネルx割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。各チャネルの各割込みソースのマスクビットもOTG_HCINTMSKxレジスタで使用可能です。

- ホストコアは、以下のステータスチェックおよび割込み生成を行います。
  - データ転送がアプリケーション側（AHB）とUSB側の両方で完了したことを示す転送完了割込み
  - 転送完了、USB トランザクションエラー、またはアプリケーションからの無効化コマンドのためにチャネルが停止したかどうか
  - 関連する送信FIFOが半エンプティか完全エンプティか（INエンドポイント）
  - ACK応答を受け取った時
  - NAK応答を受け取った時
  - STALL応答を受け取った時
  - CRCエラー、タイムアウト、ビットスタッフエラー、偽EOPによるUSBトランザクションエラー
  - バブルエラー
  - フレームオーバーラン
  - データトグルエラー

73.7.3 ホストスケジューラ

ホストコアは、アプリケーションによってポストされたUSBトランザクションリクエストを自律的に並べ替えて、管理する組み込みハードウェアスケジューラを備えています。各フレームの開始時に、ホストは、周期的（アイソクロナスおよびインタラプト）トランザクションを最初に実行し、次に、非周期的（コントロールおよびバルク）トランザクションを行うことによってUSBの仕様に従って、アイソクロナスおよびインタラプト転送タイプを優先します。

ホストは、リクエストキュー（1つは周期的転送のキュー、もう1つは非周期的転送のキュー）を通じてUSBトランザクションを処理します。各リクエストキューは、最大8個のエントリを保持できます。各エントリはアプリケーションからのペンディング中のトランザクションリクエストを表し、INまたはOUTチャネル番号とUSB上でトランザクションを実行するためのその他の情報を保持しています。リクエストがキューに書き込まれた順序によってUSBインタフェース上のトランザクションの順序が決まります。

各フレームの先頭で、ホストは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。現在のフレームに対してスケジュールされたアイソクロナスまたはインタラプトトランザクションが現在のフレームの終了時点でまだペンディング中である場合、ホストは、不完全周期転送割込み（OTG_GINTSTSのIPXFRビット）を発行します。OTG_HS コアは周期的と非周期的リクエストキューの管理を行います。周期的送信FIFOおよびキューステータスレジスタ（OTG_HPTXSTS）と非周期的送信FIFOおよびキューステータスレジスタ（OTG_HNPTXSTS）は読出し専用レジスタであり、アプリケーションから各リクエストキューのステータスを読み出すときに使うことができます。これらのレジスタには次が含まれます。

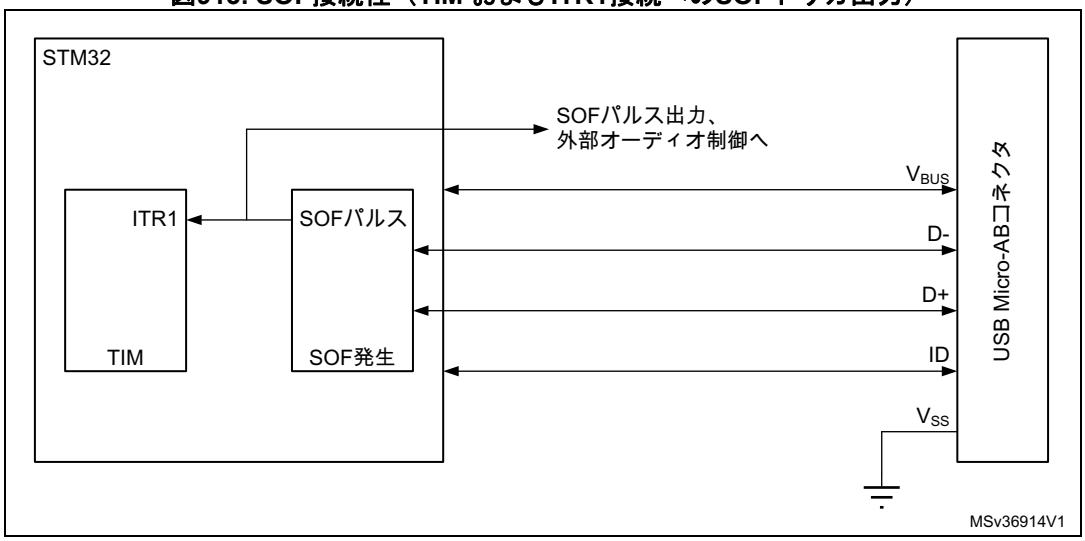
- 周期（非周期）リクエストキューで現在使用可能なフリーエントリの数（最大8エントリ）。
- 周期（非周期）Tx FIFO（OUTトランザクション）で現在使用可能な空きスペース
- IN/OUTトークン、ホストチャネル番号、およびその他のステータス情報

リクエストキューはそれぞれ最大8つのエントリを保持できるので、アプリケーションは、最大8つの保留中の周期トランザクションに加えて8つの保留中の非周期トランザクションについてSBに物理的に到達する前に、ホストトランザクションを事前にスケジュールできます。

ホストスケジューラ（キュー）へトランザクションリクエストをポストするときは、アプリケーションはOTG_HNPTXSTSレジスタのPTXQSAVビット、またはOTG_HNPTXSTSレジスタのNPTQXSAVビットを読み出して、周期的（非周期的）リクエストキュー内に少なくとも1つのエントリが存在することを確認する必要があります。

73.8 OTG_HS SOFトリガ

図913. SOF接続性（TIM およびITR1接続へのSOFトリガ出力）



OTG_HS コアは、ホスト、ペリフェラル、SOFパルス出力接続機能内でSOFフレーミングを監視、追跡、設定する手段を提供します。

これらのユーティリティは、オーディオペリフェラルがPCによって提供されるアイソクロナスストリームに同期する必要がある場合や、ホストがオーディオペリフェラルの要件に従ってフレーミング速度をトリム（微調整）する必要がある場合など、適応型オーディオクロック生成技術にとって特に役立ちます。

73.8.1 ホストのSOF

ホストモードでは、2つの連続したSOF（HS/FS）またはキープアライブ（LS）トークンの生成の間に発生するPHYクロックの数は、ホストフレームインターバルレジスタ（HFIR）でプログラム可能であり、それにより、SOFフレーミング周期中にアプリケーションによる制御が可能です。割込みは、どのフレームの開始時点でも生成されます（OTG_GINTSTSのSOFビット）。現在のフレーム番号と次のSOFまでの残り時間が、ホストフレーム番号レジスタ（HFNUM）で追跡されます。

SOFパルス信号は、20 HCLKサイクルの幅の任意のSOF開始トークンで生成されます。SOFパルスは、タイマの入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマをSOFパルスによってトリガできます。

73.8.2 ペリフェラルのSOF

デバイスモードでは、フレーム割込みの開始は、SOFトークンがUSB上で受信されるたびに生成されます（OTG_GINTSTSのSOFビット）。対応するフレーム番号は、デバイスステータスレジスタ（OTG_DSTSのFNSOFビット）から読み出すことができます。20 HCLKサイクルの幅を持つSOFパルス信号も生成されます。SOFパルス信号は、TIM入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマをSOFパルスによってトリガできます。

周期的フレーム割込みの終わり（OTG_GINTSTSのEOPF）は、デバイス設定レジスタの周期的フレームインターバルフィールド（OTG_DCFGのPFIVLビット）に応じて、タイムフレームインターバルの80 %、85 %、90 % または95 % が経過した時点をアプリケーションに通知するために使用されます。この機能を使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

73.9 OTG_HS 低消費電力モード

下記の表756 に、STM32の低消費電力モードとOTGとの両立性を定義します。

表756.STM32の低消費電力モードのOTGとの両立性

モード	説明	USBとの両立性
RUN	MCUフルアクティブ	USBがサスペンド状態ではないときに必要です。
SLEEP	USBのサスペンド終了によって、デバイスはSLEEPモードから復帰します。ペリフェラルレジスタの内容は保たれます。	USBがサスペンド状態のときに使用可能です。
STOP	USBのサスペンド終了によって、デバイスはSTOPモードから復帰します。ペリフェラルレジスタの内容は保たれます(1)。	USBがサスペンド状態のときに使用可能です。
STANDBY	パワーダウン状態。ペリフェラルは、STANDBYモードを終了した後に再初期化する必要がある。	USBのアプリケーションとは両立しません。

1. STOPモード内では、さまざまな設定の可能性があります。いくつかの制約もあります。OTGを使用中にどのような制約があるか（ある場合）については、セクション 10：電源制御（PWR）を参照してください。

次のビットと手順によって電力消費が低減されます。

OTG PHYの消費電力は、一般コア設定レジスタの 次のビットによって制御されます。

- V_{BUS} 検出イネーブル（OTG_GCCFGのVBDEN）
  OTGの動作に関連するV_{BUS} レベル検出用コンパレータのオン／オフを切り替えます。

USBセッションがまだ有効でないか、デバイスが切断されているときには、USBサスペンド状態で節電テクニックを使用できます。

- PHYクロックを停止します（OTG_PCGCCTLのSTPPCLKビット）。
  クロックゲーティング制御レジスタのストップPHYクロックビットをセットすると、ほとんどのトランシーバが無効になり、非同期レジュームやリモートウェイクアップイベントの検出を担当する部分だけが動作状態を維持します。
- ゲートHCLK（OTG_PCGCCTLのGATEHCLKビット）
  クロックゲーティング制御レジスタのゲートHCLKビットをセットすると OTG_HS コア内部のシステムクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。レジスタ読出しおよび書込みインタフェースのみが動作状態を維持します。USBクロック切り替え機能による動的な電力消費は、システムクロックが他の目的のためにアプリケーションによって動作状態を維持されている場合でも節減されます。
- USBシステムの停止
  OTG_HS がUSBサスペンド状態にある場合、アプリケーションはシステム内のすべてのクロックソースを完全にシャットダウンすることによって、電力消費全体を大幅に削減する方法を取ることができます。USBシステムの停止は、まず、ストップPHYクロックビットをセットし、次に、パワー制御システムモジュール（PWR）でシステムディープスリープモードを設定することによって有効になります。
  OTG_HS コアは、USB上でリモートウェイクアップ（ホストとして）またはリモートレジューム（デバイスとして）信号の非同期検出によって、システムとUSBクロックの両方を自動的に再起動します。

動的消費電力を節減するために、USBデータFIFOは、OTG_HS コアによってアクセスされたときだけクロック駆動されます。

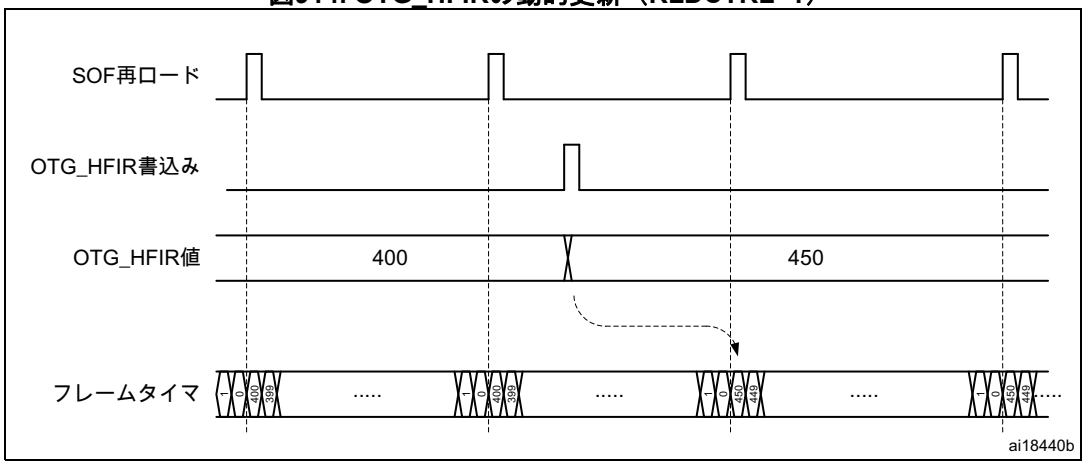
73.10 OTG_HSOTG_HFIRレジスタの動的更新

USBコアはホストモードでマイクロSOFフレーミング周期を動的にトリミングする機能を内蔵しています。この機能を使うと、外部デバイスをマイクロSOFフレームと同期させることができます。

現在のマイクロSOFフレーム内でOTG_HFIRレジスタを変更すると、図914で説明するように次のフレームSOF周期補正が適用されます。

動的更新では、RLDCTRL=1にセットする必要があります。

図914. OTG_HFIRの動的更新（RLDCTRL=1）

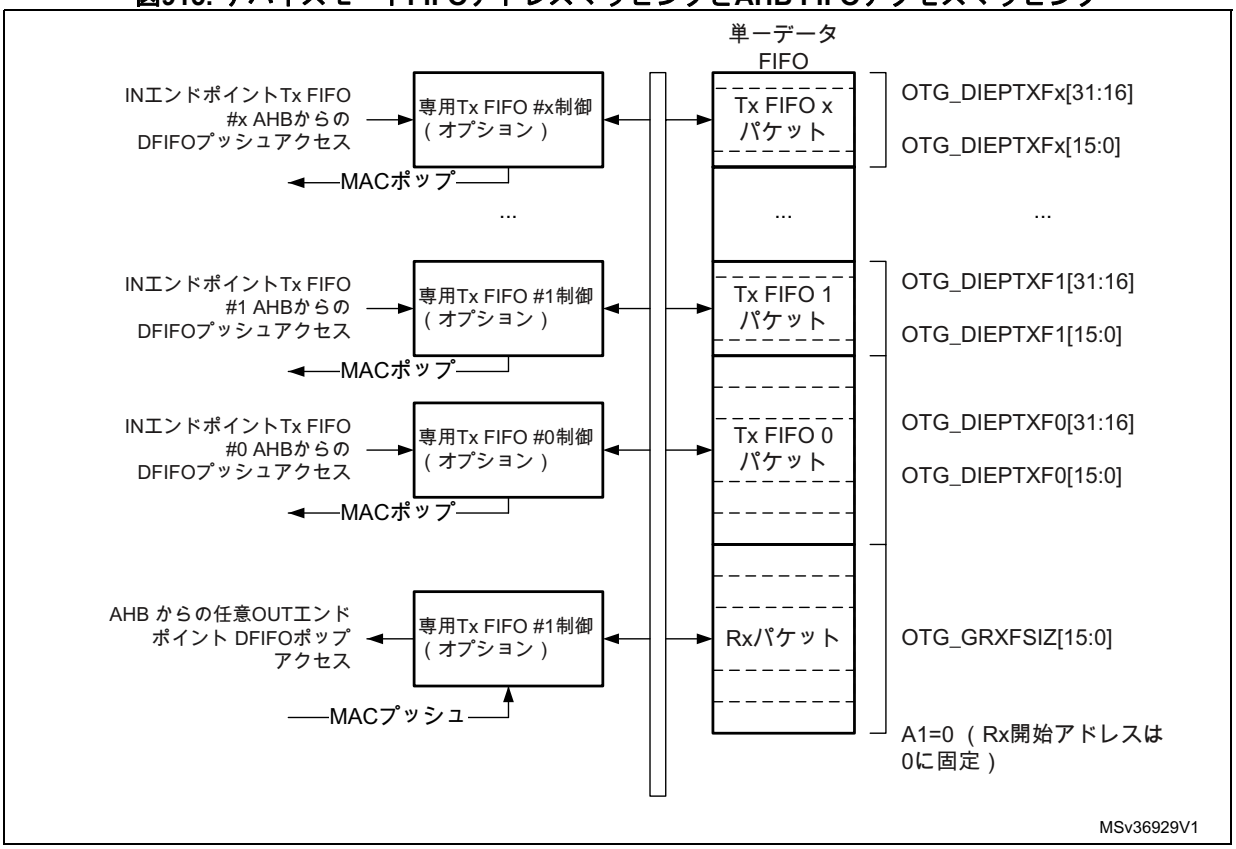


73.11 OTG_HS データFIFO

USBシステムは、高度なFIFO制御メカニズムを持つ 4 KBの専用RAMを備えています。OTG_HS コアのパケットFIFOコントローラモジュールは、RAM空間を、アプリケーションがUSB転送の前に一時的にデータを格納する（プッシュする）Tx FIFOと、USBから受信したデータをアプリケーションが取得する（ポップする）前に一時的に格納されるRx FIFOに編成します。割り当てられるFIFOの数と、これらがRAMの内部でどのように構成されるかは、デバイスの役割に依存します。ペリフェラルモードでは、追加のTx FIFOがアクティブなINエンドポイントのそれぞれに対して割り当てられます。FIFOのサイズは、アプリケーションの要件に合うように、ソフトウェアで設定できます。

73.11.1 ペリフェラルFIFOアーキテクチャ

図915. デバイスモードFIFOアドレスマッピングとAHB FIFOアクセスマッピング



ペリフェラルRx FIFO

OTGペリフェラルは、すべてのOUTエンドポイントに向けたデータを受信する単一の受信FIFOを使用します。受信パケットは、Rx FIFOの空きスペースが使用可能になるまで、連続的にスタックされます。受信パケットのステータス（OUTエンドポイントの宛先番号、バイト数、データPID、および受信データの有効性を含みます）も、コアによってデータペイロードの最上部に格納されます。使用可能なスペースがなくなると、ホストトランザクションはNACKされ、アドレス指定されたエンドポイントに割込みが受信されます。受信FIFOのサイズは、受信FIFOサイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信FIFOアーキテクチャは、USBペリフェラルによる受信RAMバッファへの書込みをより効率的にします。

- すべてのOUTエンドポイントは同じRAMバッファを共有します（共有FIFO）。
- OTG_HS コアは、OUTトークンのいかなるホストシーケンスについても、限度まで受信FIFOに書き込むことができます。

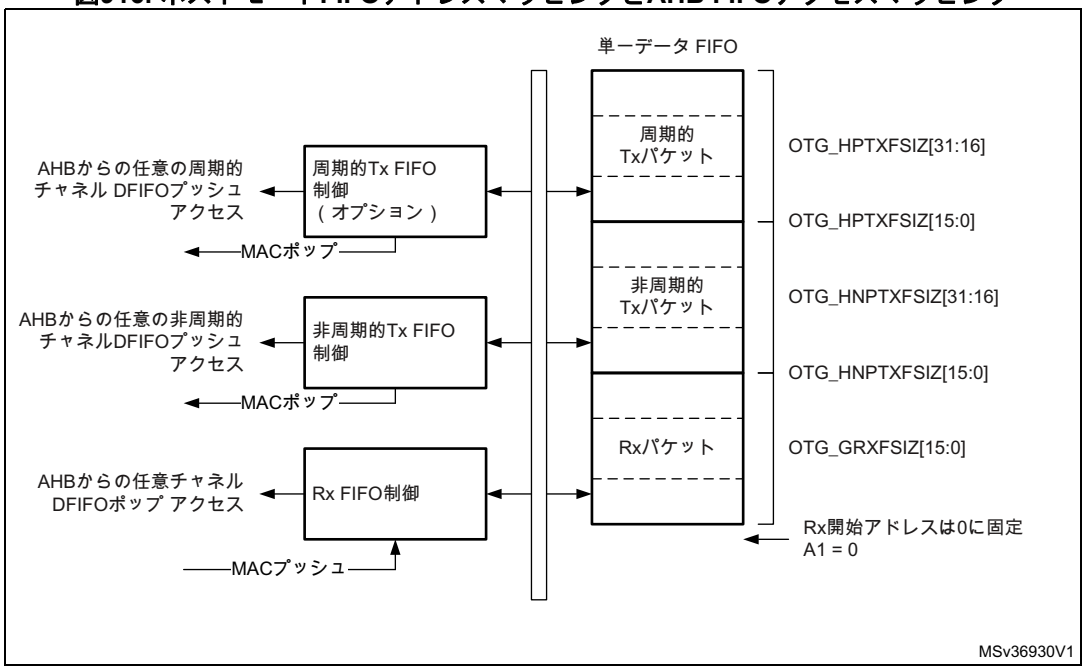
アプリケーションはRx FIFO非エンプティ割込み（OTG_GINTSTSのRXFLVLビット）の受信を、少なくとも1パケットがダウンロード可能である限り継続します。アプリケーションは、受信ステータス読出しおよびポップレジスタ（OTG_GRXSTSP）からパケット情報を読み出し、最後に、エンドポイントに関するポップアドレスから読み取ることによって受信FIFOからデータを取得します。

ペリフェラルTx FIFO

コアは、各INエンドポイント用の専用FIFOを備えています。アプリケーションは、INエンドポイント0についてはエンドポイント0送信FIFOサイズレジスタ（OTG_DIEPTXF0）に書き込み、INエンドポイントxについてはデバイスINエンドポイント送信FIFOxレジスタ（OTG_DIEPTXFx）に書き込むことによって、FIFOのサイズを設定します。

73.11.2 ホストFIFOアーキテクチャ

図916. ホストモードFIFOアドレスマッピングとAHB FIFOアクセスマッピング



ホストRx FIFO

ホストは、すべての周期的および非周期的トランザクションについて1つの受信FIFOを使用します。このFIFOは、USBから受信したデータ（受信パケットのペイロード）をシステムメモリに転送されるまで保持する受信バッファとして使用されます。任意のリモートINエンドポイントから受信されたパケットは、空きスペースがある限り連続的にスタックされます。各受信パケット（ホストチャネルデスティネーション、バイト数、データPID、および受信データの有効性を含みます）のステータスも、FIFOに格納されます。受信FIFOのサイズは、受信FIFOサイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信FIFOアーキテクチャは、USBホストによる受信データバッファへの書込みをより効率的にします。

- INに設定されたすべてのホストチャネルは、同じRAMバッファを共有します（共有FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動されたINトークンのいかなるシーケンスについても、限度まで受信FIFOに書き込むことができます。

アプリケーションは、少なくとも1つのパケットがダウンロード可能である限り、Rx FIFO非エンプティ割込みを受け取ります。アプリケーションは、受信ステータス読出しおよびポップレジスタからパケット情報を読み出し、最後に、受信FIFOからデータを吸い上げます。

ホストTx FIFO

ホストは、すべての非周期的（コントロールおよびバルク）OUTトランザクションついて1つの送信FIFOを、すべての周期的（アイソクロナスおよびインタラプト）OUTトランザクションについて1つの送信FIFOを使用します。FIFOは、USB経由で送信されるデータ（送信パケットのペイロード）を保持する送信バッファとして使用されます。周期的（非周期的）Tx FIFOのサイズは、ホストの周期的（非周期的）送信FIFOサイズレジスタ（OTG_HPTXFSIZ/OTG_HNPTXFSIZ）で設定されます。

Tx FIFOが2つ実装されているのは、USBフレームよりも周期的トラフィックタイプに対して高い優先度が与えられていることによります。各フレームの先頭で、組み込みのホストスケジューラは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。

2つの送信FIFOアーキテクチャにより、USBホストは周期的送信データバッファと非周期的送信データバッファを個別に最適化できます。

- OUT 方向の周期的（非周期的）トランザクションをサポートするように設定されたすべてのホストチャネルは、同じRAMバッファを共有します（共有FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動されたOUTトークンのいかなるシーケンスについても、限度まで周期的（非周期的）送信FIFOに書き込むことができます。

OTG_HS コアは、AHB設定レジスタの周期的Tx FIFOエンプティレベルビット（OTG_GAHBCFGのPTXFELVLビット）の値に応じて、周期的Tx FIFOが半分または完全にエンプティである限り、周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのPTXFEビット）を発行します。アプリケーションは、周期的Tx FIFOと周期的リクエストキューの両方に空きスペースが存在する限り、前もって送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの周期的送信FIFOおよびキューステータスレジスタ（OTG_HPTXSTS）を読み出すことで知ることができます。

OTG_HS コアは、AHB設定レジスタの非周期的Tx FIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）の値に応じて、非周期的Tx FIFOが半分または完全にエンプティである限り、非周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのNPTXFEビット）を発行します。アプリケーションは、非周期的Tx FIFOと非周期的リクエストキューの両方に空きスペースが存在する限り、送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの非周期的送信FIFOおよびキューステータスレジスタ（OTG_HNPTXSTS）を読み出すことで知ることができます。

73.11.3 FIFO RAMの割り当て

デバイスモード

受信FIFO RAM割当て: アプリケーションはSETUPパケット用にRAMを割り当てる必要があります。

- コントロールエンドポイントでSETUPパケットを受信するため受信FIFOに10ロケーションを確保する必要があります。コアは、SETUPパケット用に予約されている位置に他のデータを書き込むことはありません。
- 1つの位置は、グローバルOUT NAKのために割り当てられます。
- 各受信パケットとともに、ステータス情報がFIFOに書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1のスペースを割り当てる必要があります。複数のアイソクロナスエンドポイントが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ /4）+ 1の2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の2倍のスペースが推奨されます。前のパケットがCPUに転送されている間に、USBは次のパケットを受信できるからです。
- 各エンドポイントの最後のパケットとともに、送信完了ステータス情報がFIFOにプッシュされます。各OUTエンドポイントに対して1つのロケーションを予約することが推奨されます。

デバイスRxFIFO =

（5 * コントロールエンドポイント数 + 8）+ （（最大USB使用パケット / 4）+ ステータス情報につき1）+ （2 * OUTエンドポイント数）+ グローバルNAKにつき1

例：MPSは周期的USBパケットでは1,024バイト、非周期的USBパケットでは512バイトです。3つのOUTエンドポイント、3つのINエンドポイント、1つのコントロールエンドポイント、3つのホストチャネルがあります。

デバイスRxFIFO = （5 * 1 + 8）+ （（1,024 / 4）+1）+ （2 * 4）+ 1 = 279

送信FIFO RAMの割り当て：各INエンドポイント送信FIFOに必要な最小RAMスペースは、その特定のINエンドポイントに対する最大パケットサイズを収容できるスペースとなります。

注： **送信INエンドポイントFIFOに割り当てるスペースが大きいほど、USBのパフォーマンスは高くなります。**

ホストモード

受信FIFO RAM割り当て：

各受信パケットとともに、ステータス情報がFIFOに書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1のスペースを割り当てる必要があります。複数のアイソクロナスチャネルが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1の2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の2倍のスペースが推奨されます。前のパケットがCPUに転送されている間に、USBは次のパケットを受信できるからです。

ホストチャネルの最後のパケットとともに、転送完了ステータス情報がFIFOにプッシュされます。このために1つのロケーションを割り当てる必要があります。

ホストRxFIFO = （最大USB使用パケット / 4）+ ステータス情報につき1 + 1転送完了

例：ホストRxFIFO = （（1,024 / 4）+ 1）+ 1 = 258

送信FIFO RAMの割り当て：

ホストの非周期的送信FIFOに必要な最小RAMスペースは、サポートされるすべての非周期的OUTチャネルの中で最大のパケットサイズを収容できるスペースとなります。

一般に、最大パケットサイズの2倍のスペースが推奨されます。現在のパケットがUSBに転送されているときに、CPUが次のパケットを取得できるからです。

非周期的TxFIFO = 最大非周期USB使用パケット / 4

例：非周期的TxFIFO = （512 / 4）= 128

ホストの周期的送信FIFOに必要な最小RAMスペースは、サポートされるすべての周期的OUTチャネルの中で最大のパケットサイズを収容できるスペースとなります。少なくとも1つのアイソクロナスOUTエンドポイントがある場合には、スペースはそのチャネルの最大パケットサイズの少なくとも2倍でなければなりません。

ホスト周期的TxFIFO = 最大周期USB使用パケット / 4

例：ホスト周期的TxFIFO = （1,024 / 4）= 256

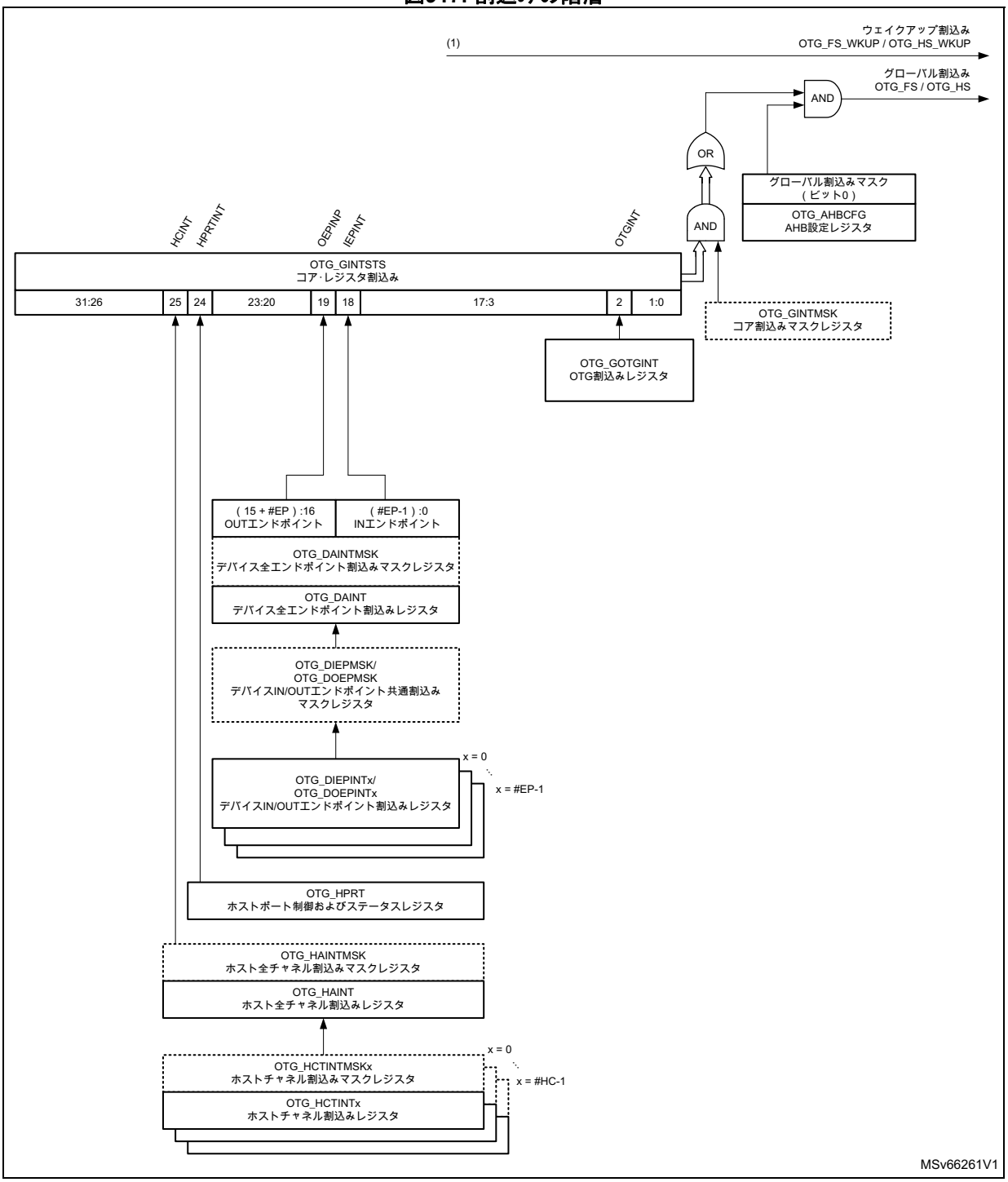
注： **非周期的送信FIFOに割り当てるスペースが大きいほど、USBのパフォーマンスは高くなります。**

73.12 OTG_HS割込み

OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTSレジスタのMMISビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

図917に、割込みの階層を示します。

図917. 割込みの階層



1. L1 SLEEPまたはL2 SUSPEND状態中にレジューム条件が発生すると、OTG_HS_WKUPがアクティブ（ハイ状態）になります。

73.13 OTG_HS 制御レジスタおよびステータスレジスタ

アプリケーションはAHBスレーブインタフェースを通じて制御／ステータスレジスタ（CSR）を読み書きすることによって OTG_HS コントローラを制御します。これらのレジスタは32bit幅であり、アドレスは32bitブロックに配置されています。OTG_HS レジスタは、ワード（32bit）単位でアクセスする必要があります。

CSRは、次のように分類されます。

- コアグローバルレジスタ
- ホストモードレジスタ
- ホストグローバルレジスタ
- ホストポートCSR
- ホストチャネル固有レジスタ
- デバイスモードレジスタ
- デバイスグローバルレジスタ
- デバイスエンドポイント固有レジスタ
- パワーおよびクロックゲーティングレジスタ
- データFIFO（DFIFO）アクセスレジスタ

コアグローバル、パワーおよびクロックゲーティング、データFIFO（DFIFO）アクセス、およびホストポートCSRレジスタのみが、ホストモードおよびデバイスモードの両方でのアクセスが可能です。OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTSレジスタのMMISビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

73.13.1 CSRメモリマップ

ホストモードおよびデバイスモードレジスタは、それぞれ異なるアドレスを占めます。すべてのレジスタはAHBクロックドメインで実装されます。

グローバルCSRマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能です。

表757.コアグローバル制御およびステータスレジスタ（CSR）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GOTGCTL	0x000	セクション 73.14.1：OTG制御およびステータスレジスタ（OTG_GOTGCTL）
OTG_GOTGINT	0x004	セクション 73.14.2：OTG割込みレジスタ（OTG_GOTGINT）
OTG_GAHBCFG	0x008	セクション 73.14.3：OTG AHB設定レジスタ（OTG_GAHBCFG）
OTG_GUSBCFG	0x00C	セクション 73.14.4：OTG USB設定レジスタ（OTG_GUSBCFG）
OTG_GRSTCTL	0x010	セクション 73.14.5：OTGリセットレジスタ（OTG_GRSTCTL）
OTG_GINTSTS	0x014	セクション 73.14.6：OTGコア割込みレジスタ[オルタネート]（OTG_GINTSTS） セクション 73.14.7：OTGコア割込みレジスタ[オルタネート]（OTG_GINTSTS）

表757.コアグローバル制御およびステータスレジスタ（CSR）（続き）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GINTMSK	0x018	セクション 73.14.8：OTG割込みマスクレジスタ[オルタネート]（OTG_GINTMSK） セクション 73.14.9：OTG割込みマスクレジスタ[オルタネート]（OTG_GINTMSK）
OTG_GRXSTSR	0x01C	セクション 73.14.10：OTG受信ステータスデバッグ読出しレジスタ[オルタネート]（OTG_GRXSTSR） セクション 73.14.11：OTG受信ステータスデバッグ読出しレジスタ[オルタネート]（OTG_GRXSTSR）
OTG_GRXSTSP	0x020	セクション 73.14.12：OTGステータス読出しおよびポップレジスタ（OTG_GRXSTSP） セクション 73.14.13：OTGステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）
OTG_GRXFSIZ	0x024	セクション 73.14.14：OTG受信FIFOサイズレジスタ（OTG_GRXFSIZ）
OTG_HNPTXFSIZ/OTG_DIEPTXF0(1)	0x028	セクション 73.14.15：OTGホスト非周期的送信FIFOサイズレジスタ[オルタネート]（OTG_HNPTXFSIZ） セクション 73.14.16：エンドポイント0送信FIFOサイズ[オルタネート]（OTG_DIEPTXF0）
OTG_HNPTXSTS	0x02C	セクション 73.14.17：OTG非周期的送信FIFO／キューステータスレジスタ（OTG_HNPTXSTS）
OTG_GCCFG	0x038	セクション 73.14.18：OTG一般コア設定レジスタ（OTG_GCCFG）
OTG_CID	0x03C	セクション 73.14.19：OTGコアIDレジスタ（OTG_CID）
OTG_GLPMCFG	0x54	セクション 73.14.20：OTGコアLPM設定レジスタ（OTG_GLPMCFG）
OTG_HPTXFSIZ	0x100	セクション 73.14.21：OTGホスト周期的送信FIFOサイズレジスタ（OTG_HPTXFSIZ）
OTG_DIEPTXFx	0x104 0x108 ..... 0x120	セクション 73.14.22：OTGデバイスINエンドポイント送信FIFO xサイズレジスタ（OTG_DIEPTXFx）

1. 一般的ルールとして、OTG_HNPTXFSIZはホストモードで、OTG_DIEPTXF0はデバイスモードで使用されます。

ホストモードCSRマップ

これらのレジスタは、コアがホストモードに変わるたびにプログラムされなければなりません。

表758.ホストモード制御およびステータスレジスタ （CSR）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCFG	0x400	セクション 73.14.24：OTGホスト設定レジスタ（OTG_HCFG）
OTG_HFIR	0x404	セクション 73.14.25：OTGホストフレームインターバルレジスタ（OTG_HFIR）
OTG_HFNUM	0x408	セクション 73.14.26：OTGホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）
OTG_HPTXSTS	0x410	セクション 73.14.27：OTGホスト周期的送信FIFO ／キューステータスレジスタ（OTG_HPTXSTS）
OTG_HAINT	0x414	セクション 73.14.28：OTGホスト全チャネル割込みレジスタ（OTG_HAINT）
OTG_HAINTMSK	0x418	セクション 73.14.29：OTGホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）
OTG_HPRT	0x440	セクション 73.14.30：OTGホストポート制御およびステータスレジスタ（OTG_HPRT）
OTG_HCCHARx	0x500 0x520 ..... 0x6E0	セクション 73.14.31：OTGホストチャネルx特性レジスタ（OTG_HCCHARx）
OTG_HCSPLTx	0x504 0x524 ... 0x6E4	セクション 73.14.32：OTGホストチャネルxスプリット制御レジスタ（OTG_HCSPLTx）
OTG_HCINTx	0x508 0x528 ... 0x6E8	セクション 73.14.33：OTGホストチャネルx割込みレジスタ（OTG_HCINTx）
OTG_HCINTMSKx	0x50C 0x52C ... 0x6EC	セクション 73.14.34：OTGホストチャネルx割込みマスクレジスタ（OTG_HCINTMSKx）
OTG_HCTSIZx	0x510 0x530 ... 0x6F0	セクション 73.14.35：OTGホストチャネルx転送サイズレジスタ（OTG_HCTSIZx）
OTG_HCDMAx	0x514 0x534 ... 0x6F4	セクション 73.14.36：OTGホストチャネルx DMAアドレスレジスタ（OTG_HCDMAx）

デバイスモードCSRマップ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

表759.デバイスモード制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_DCFG	0x800	セクション 73.14.38 : OTGデバイス設定レジスタ（OTG_DCFG）
OTG_DCTL	0x804	セクション 73.14.39 : OTGデバイス制御レジスタ（OTG_DCTL）
OTG_DSTS	0x808	セクション 73.14.40 : OTGデバイスステータスレジスタ（OTG_DSTS）
OTG_DIEPMSK	0x810	セクション 73.14.41 : OTGデバイスINエンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）
OTG_DOEPMSK	0x814	セクション 73.14.42 : OTGデバイスOUTエンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）
OTG_DAINT	0x818	セクション 73.14.43 : OTGデバイス全エンドポイント割込みレジスタ（OTG_DAINT）
OTG_DAINTMSK	0x81C	セクション 73.14.44 : OTG全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）
OTG_DTHRCTL	0x830	セクション 73.14.45 : OTGデバイス閾値制御レジスタ（OTG_DTHRCTL）
OTG_DIEPEMPMSK	0x834	セクション 73.14.46 : OTGデバイスINエンドポイントFIFOエンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）
OTG_DIEPCTLx	0x900 0x920 ..... 0xA00	セクション 73.14.47 : OTGデバイスINエンドポイントx制御レジスタ[オルタネート]（OTG_DIEPCTLx） セクション 73.14.48 : OTGデバイスINエンドポイントx制御レジスタ[オルタネート]（OTG_DIEPCTLx）
OTG_DIEPINTx	0x908 0x928 ..... 0x9E8	セクション 73.14.49 : OTGデバイスINエンドポイントx割込みレジスタ（OTG_DIEPINTx）
OTG_DIEPTSIZ0	0x910	セクション 73.14.50 : OTGデバイスINエンドポイント0転送サイズレジスタ（OTG_DIEPTSIZ0）
OTG_DIEPDMAx	0x914 0x934 ..... 0x9F4	セクション 73.14.51:OTGデバイスINエンドポイントx DMAアドレスレジスタ（OTG_DIEPDMAx）
OTG_DTXFSTSx	0x918 0x938 ..... 0x9F8	セクション 73.14.52 : OTGデバイスINエンドポイント送信FIFOステータスレジスタ（OTG_DTXFSTSx）
OTG_DIEPTSIZx	0x930 0x950 ..... 0x9F0	セクション 73.14.53 : OTGデバイスINエンドポイントx転送サイズレジスタ（OTG_DIEPTSIZx）

表759.デバイスモード制御およびステータスレジスタ （続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_DOEPCTL0	0xB00	セクション 73.14.54 : OTGデバイスコントロールOUTエンドポイント0制御レジスタ（OTG_DOEPCTL0）
OTG_DOEPINTx	0xB08 0XB28 ..... 0xC08	セクション 73.14.55 : OTGデバイスOUTエンドポイントx割込みレジスタ（OTG_DOEPINTx）
OTG_DOEPTSIZ0	0xB10	セクション 73.14.56 : OTGデバイスOUTエンドポイント0転送サイズレジスタ（OTG_DOEPTSIZ0）
OTG_DOEPDMAx	0xB14 0xB34 ..... 0xC14	セクション 73.14.57 : OTGデバイスOUTエンドポイントx DMAアドレスレジスタ （OTG_DOEPDMAx）
OTG_DOEPCTLx	0xB20 0xB40 ..... 0xC00	セクション 73.14.58 : OTGデバイスOUTエンドポイントx制御レジスタ[オルタネート] （OTG_DOEPCTLx） セクション 73.14.59 : OTGデバイスOUTエンドポイントx制御レジスタ[オルタネート] （OTG_DOEPCTLx）
OTG_DOEPTSIZx	0xB30 0xB50 .. 0xBF0	セクション 73.14.60 : OTGデバイスOUTエンドポイントx転送サイズレジスタ（OTG_DOEPTSIZx）

データFIFO（DFIFO）アクセスレジスタマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能であり、特定のエンドポイントまたはチャネルのFIFOスペースを指定された方向で読み書きするために使用されます。ホストチャネルがINタイプの場合FIFOはそのチャネル上では読出しのみが可能です。同様に、ホストチャネルがOUTタイプの場合FIFOはそのチャネル上では書込みのみが可能です。

表760.データFIFO（DFIFO）アクセスレジスタマップ

FIFOアクセスレジスタセクション	オフセットアドレス	アクセス
デバイスINエンドポイント0／ホストOUTチャネル0 : DFIFO書込みアクセス デバイスOUTエンドポイント0／ホストINチャネル0 : DFIFO読出しアクセス	0x1000 – 0x1FFC	w r
デバイスINエンドポイント1／ホストOUTチャネル1 : DFIFO書込みアクセス デバイスOUTエンドポイント1／ホストINチャネル1 : DFIFO読出しアクセス	0x2000～0x2FFC	w r
.....	.....	.....
デバイスINエンドポイントx(1)／ホストOUTチャネルx (1) : DFIFO書込みアクセス デバイスOUTエンドポイントx (1) ／ホストINチャネルx (1) : DFIFO読出しアクセス	0xX000～0xXFFC	w r

1. デバイスモードではx = 8、ホストモードではx = 15です。

パワーおよびクロックゲーティングCSRマップ

パワーおよびクロックゲーティングの場合、シングルレジスタです。このレジスタは、ホストモードとデバイスモードの両方で使用できます。

表761.パワーおよびクロックゲーティング制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_PCGCCTL	0xE00～0xE04	セクション 73.14.61：OTGパワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

73.14 OTG_HS レジスタ

これらのレジスタは、ホストモードとデバイスモードの両方で使用でき、これらのモード間で切り替わっても再プログラムの必要はありません。

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

73.14.1 OTG制御およびステータスレジスタ（OTG_GOTGCTL）

OTG_GOTGCTLレジスタは、コアのOTG機能の動作を制御し、そのステータスを反映します。

アドレスオフセット：0x000

リセット値：0x0001 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CURMOD	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS
										r	rw	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EHEN	Res.	Res.	Res.	Res.	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	Res.	Res.
			rw					rw	rw	rw	rw	rw	rw		

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **CURMOD**：現在の動作モード

現在のモードを示します（ホストまたはデバイス）。

0：デバイスモード

1：ホストモード

ビット20 **OTGVER**：OTGバージョン

OTGバージョンを選択します。

0：OTGバージョン1.3です。OTG1.3は、新しい製品開発では廃止されています。

1：OTGバージョン2.0です。このバージョンでは、コアはSRPのデータラインパルシングのみをサポートしています。

ビット19 **BSVLD**：Bセッション有効

デバイスモードでのトランシーバのステータスを示します。

0：Bセッションは無効。

1：Bセッションは有効。

OTGモードでは、ユーザはこのビットを使用して、デバイスが接続されているか、切断されているかを判断できます。

注： デバイスモードでのみアクセス可能です。

ビット18 **ASVLD**：Aセッションの有効性

ホストモードでのトランシーバのステータスを示します。

0：Aセッションは無効。

1：Aセッションは有効。

注： ホストモードでのみアクセス可能です。

ビット17 **DBCT**：ロング／ショートデバウンス時間

検出された接続のデバウンス時間を示します。

0：ロングデバウンス時間（100ms + 2.5µs）。物理的接続に使用されます。

1：ショートデバウンス時間（2.5 µs）。ソフト接続に使用されます。

注： ホストモードでのみアクセス可能です。

ビット16 **CIDSTS**：コネクタIDステータス

接続イベント時のコネクタIDのステータスを示します。

0：OTG_HS コントローラはAデバイスモードです。

1：OTG_HS コントローラはBデバイスモードです。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **EHEN**：組み込みホストの有効化

OTG Aデバイスのステートマシンと組み込みホストのステートマシンから選択するために使用します。

0：OTG Aデバイスのステートマシンを選択します。

1：組み込みホストのステートマシンを選択します。

ビット11:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **BVALOVAL**：Bペリフェラルセッション有効上書き値

このビットは、BVALOENビットがセットされている場合に、Bvalid信号に上書き値をセットするために使用されます。

0：BVALOEN = 1のとき、Bvalid値は「0」です。

1：BVALOEN = 1のとき、Bvalid値は「1」です。

注： デバイスモードでのみアクセス可能です。

ビット6 **BVALOEN**：Bペリフェラルセッション有効値上書きの有効化

このビットは、BVALOVALビットを使用してBvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのBvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったBvalid信号はBVALOVALビット値で上書きされます。

注： デバイスモードでのみアクセス可能です。

ビット5 **AVALOVAL**：Aペリフェラルセッション有効上書き値

このビットは、AVALOENビットがセットされている場合に、Avalid信号に上書き値をセットするために使用されます。

0：AVALOEN = 1のとき、Avalid値は「0」です。

1：AVALOEN = 1のとき、Avalid値は「1」です。

注： ホストモードでのみアクセス可能です。

ビット4 **AVALOEN**：Aペリフェラルセッション有効値上書きの有効化

このビットは、AVALOVALビットを使用してAvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのAvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったAvalid信号はAVALOVALビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット3 **VBVALOVAL**：V_{BUS} 有効上書き値

このビットは、VBVALOENビットがセットされている場合に、vbusvalid信号に上書き値をセットするために使用されます。

0：VBVALOEN = 0のとき、vbusvalid値は”1”です。

1：VBVALOEN = 1のとき、vbusvalid値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット2 **VBVALOEN**：V_{BUS} 有効値上書きの有効化

このビットは、VBVALOVALビットを使用してvbusvalid信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各PHYからのvbusvalid信号は、コアが内部的に使用します。

1：PHYから内部的に受け取ったvbusvalid信号はVBVALOVALビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

73.14.2 OTG割込みレジスタ（OTG_GOTGINT）

アプリケーションはOTG割込みがあると、このレジスタを読み出し、このレジスタのビットをクリアすることによってOTG割込みをクリアします。

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADTOCHG	Res.	Res.
													rc_w1		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
													rc_w1		

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **ADTOCHG**：Aデバイスタイムアウト変化

コアはBデバイスの接続を待っている間にAデバイスがタイムアウトしたことを示すために、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット17:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **SEDET**：セッション終了検出

コアは、このビットをセットして、V_{BUS} の電圧レベルがBペリフェラルセッションにとって有効でなくなったことを示します（V_{BUS} < 0.8 Vのとき）。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット1:0 予約済みであり、リセット値に保持する必要があります。

73.14.3 OTG AHB設定レジスタ（OTG_GAHBCFG）

このレジスタを使用して、電源投入後、またはモード変化後のコアを設定できます。このレジスタは、主にAHBシステム関連の設定パラメータを含んでいます。初期プログラミングの後は、このレジスタを変更しないでください。アプリケーションはAHBまたはUSBのいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	DMAEN	HBSTLEN[3:0]				GINTMSK
							rw	rw		rw	rw	rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PTXFELVL：**周期的Tx FIFOエンプティレベル

OTG_GINTSTSレジスタの周期的Tx FIFOエンプティ割込みビット（OTG_GINTSTSのPTXFEビット）がトリガされるタイミングを示します。

0：OTG_GINTSTSのPTXFE割込みビットは、周期的Tx FIFOがハーフエンプティであることを示します。

1：OTG_GINTSTSのPTXFE割込みビットは、周期的Tx FIFOが完全エンプティであることを示します。

注： ホストモードでのみアクセス可能です。

ビット7 **TXFELVL：**周期的Tx FIFOエンプティレベル

条件：デバイスモード

このビットは、INエンドポイント送信FIFOエンプティ割込み（OTG_DIEPINTxのTXFE）がトリガされるタイミングを示します。

0：OTG_DIEPINTxのTXFE割込みは、INエンドポイントTx FIFOがハーフエンプティであることを示します。

1：OTG_DIEPINTxのTXFE割込みは、INエンドポイントTx FIFOが完全エンプティであることを示します。

条件：ホストモード

このビットは、非周期的Tx FIFOエンプティ割込み（OTG_GINTSTSのNPTXFEビット）がトリガされるタイミングを示します。

0：OTG_GINTSTSのNPTXFE割込みは、非周期的Tx FIFOがハーフエンプティであることを示します。

1：OTG_GINTSTSのNPTXFE割込みは、非周期的Tx FIFOが完全エンプティであることを示します。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 **DMAEN**：DMAの有効化

0：コアはスレーブモードで動作します。

1：コアはDMAモードで動作します。

ビット4:1 **HBSTLEN[3:0]：**バースト長／タイプ

0000シングル：バストランザクションがシングル32bitアクセスを使用します（非推奨）。

0001 INCR：バストランザクションが指定されていない長さのアクセスを使用します（非推奨、INCR AHBバスコマンドを使用）。

0011 INCR4：バストランザクションが4x 32bitアクセスをターゲットします。

0101 INCR8：バストランザクションが8x 32bitアクセスをターゲットします。

0111 INCR16：バストランザクションが16x 32bitアクセスに基づきます。

その他：予約済み

ビット0 **GINTMSK**：グローバル割込みマスク

アプリケーションは、このビットを使用して、アプリケーション自身に対する割込みラインのアサーションをマスクまたはマスク解除します。このビットの設定に関係なく、割込みステータスレジスタはコアによって更新されます。

0：アプリケーションへの割込みアサーションをマスクします。

1：アプリケーションへの割込みアサーションをマスク解除します。

注： デバイスモードとホストモードの両方でアクセス可能です。

73.14.4 OTG USB設定レジスタ（OTG_GUSBCFG）

このレジスタを使用して、電源投入後、またはホストモードあるいはデバイスモードへの変化後に、コアを設定することができます。このレジスタはUSBおよびUSB-PHY関連の設定パラメータを含んでいます。アプリケーションはAHBまたはUSBのいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。初期プログラミングの後は、このレジスタを変更しないでください。

アドレスオフセット：0x00C

リセット値：0x0000 1400

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FD MOD	FH MOD	Res.	Res.	Res.	Res.	Res.	Res.	TSDPS	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw							rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PHYL PC	Res.	TRDT[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOCAL[2:0]		
rw		rw	rw	rw	rw								rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **FDMOD**：強制デバイスモード

このビットに1を書き込むと、コアはOTG_ID入力ピンに関係なく、強制的にデバイスモードになります。

0：通常モード

1：強制デバイスモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも25 ms待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット29 **FHMOD**：強制ホストモード

このビットに1を書き込むと、コアはOTG_ID入力ピンに関係なく、強制的にホストモードになります。

0：通常モード

1：強制ホストモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも25 ms待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット28:26 予約済みであり、リセット値に保持する必要があります。

ビット25:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **TSDPS**：TermSel DLineパルスの選択

このビットはSRP （セッションリクエストプロトコル）時にデータラインパルスを駆動するutmi_termselectを選択します。

0：utmi_txvalidを使用するデータラインパルシング（デフォルト）

1：utmi_termselを使用するデータラインパルシング

ビット21:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **PHYLPC**：PHY低消費電力クロック選択

このビットは480MHzまた48MHz （低消費電力）PHYモードを選択します。FSモードとLSモードではPHYは48 MHzクロックで動作して消費電力を削減できます。

0：480 MHz内部PLLクロック

1：48 MHz外部クロック

480 MHzモードではUTMIインタフェースは選択した8bitまた16bitのデータ幅に応じ60または30 MHzで動作します。48 MHzモードではUTMIインタフェースはFSモードとLSモードで48 MHz動作します。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:10 **TRDT[3:0]**：USBターンアラウンド時間

ターンアラウンド時間をPHYクロックで設定します。これらは、表 762：TRDT値によってアプリケーションAHB周波数により設定する必要があります。TRDT値が高ければ、INトークンへのUSB応答時間を伸ばして、データFIFOへのより長いAHB読出しアクセス遅延を補正できます。

注： デバイスモードでのみアクセス可能です。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 予約済みであり、リセット値に保持する必要があります。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **TOCAL[2:0]**：FSタイムアウトの較正

アプリケーションがこのフィールドでプログラムするPHYクロックの数は、PHYによって導入される追加遅延を考慮して、コアのフルスピードパケット間タイムアウト時間に追加されます。ライン状態条件の生成時にPHYによって導入される遅延はPHYごとに異なる場合があるので、較正が必要なことがあります。

フルスピード動作でのUSBの標準タイムアウト値は16～18bit時間です。アプリケーションは、エニュメレーションのスピードに基づいて、このフィールドをプログラムする必要があります。PHYクロックごとに追加されるビット時間の数は0.25bit時間です。

表762.TRDT値

AHB周波数範囲（MHz）		TRDT最小値
最小値	最大値	
30	-	0x9

73.14.5 OTGリセットレジスタ（OTG_GRSTCTL）

アプリケーションは、このレジスタを使用して、コア内のさまざまなハードウェア機能をリセットします。

アドレスオフセット：0x010

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AHB IDL	DMAR EQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXFNUM[4:0]					TXF FLSH	RXF FLSH	Res.	FCRST	PSRST	CSRST
					rw	rw	rw	rw	rw	rs	rs		rs	rs	rs

ビット31 **AHBIDL**：AHBマスタアイドル

AHBマスタステートマシンがアイドル状態にあることを示します。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット30 **DMAREQ**：DMAリクエスト信号が有効

このビットはDMAリクエストが進行中であることを表示します。デバッグには使用しません。

ビット29:11 予約済みであり、リセット値に保持する必要があります。

ビット10:6 **TXFNUM[4:0]**：Tx FIFO番号

これはTx FIFOフラッシュビットを使用して一掃しなければならないFIFO番号です。コアがTx FIFOフラッシュビットをクリアするまでは、このフィールドを変更しないでください。

条件：ホストモード

00000：非周期的Tx FIFO一掃

00001：周期的Tx FIFO一掃

10000：すべての送信FIFOの一掃

条件：デバイスモード

00000：Tx FIFO 0の一掃

00001：Tx FIFO 1の一掃

00010：Tx FIFO 2の一掃

.....

01111：Tx FIFO 15の一掃

10000：すべての送信FIFOの一掃

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット5 **TXFFLSH**：Tx FIFOの一掃

このビットは、単一またはすべての送信FIFOを選択的にフラッシュしますが、コアがトランザクション中の場合は一掃できません。

アプリケーションは、コアがTx FIFOへの書込み中かTxFIFOからの読出し中のどちらでもないことを確認した後でなければ、このビットに書き込むことはできません。以下のレジスタを使用して確認してください。

読出し — NAK有効割込みを使用すると、コアがFIFOから読出しを行っていないことを確認できます。

書込み — OTG_GRSTCTLのAHBIDLビットで、コアがFIFOに書込みを行っていないことを保証します。

FIFOが再設定される場合は通常一掃を実施することをお勧めします。デバイスエンドポイントが無効の間も、FIFO一掃の実施をお勧めします。アプリケーションは、コアがこのビットをクリアするまで待ってから、他の操作を実行しなければなりません。phy_clkまたはhclkのより低速なクロックを使用してこのビットをクリアするには、8クロックかかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット4 **RXFFLSH**：Rx FIFOの一掃

アプリケーションは、このビットを使用してRx FIFO全体を一掃できますが、まず、コアがトランザクション中でないことを確認する必要があります。

アプリケーションは、コアがRx FIFOからの読出し中、またはRx FIFOへの書込み中のどちらでもないことを確認した後でなければ、このビットに書き込んではなりません。

アプリケーションは、他の操作を実行する前に、このビットがクリアされるまで待つ必要があります。このビットがクリアされるまで、8クロック（PHYまたはAHBクロックの中で最も低速なクロック）かかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FCRST**：ホストフレームカウンタのリセット

アプリケーションは、このビットに書き込むことによって、コア内の（マイクロ）フレーム数カウンタをリセットします。（マイクロ）フレームカウンタがリセットされると、コアによって次に送出されるSOFのフレーム番号は0になります。

アプリケーションでこのビットに1が書き込まれた場合、このビットは数クロックサイクル内にコアによってクリアさるため、その値を読み戻すことはできない可能性があります。

注： ホストモードでのみアクセス可能です。

ビット1 **PSRST**：部分的なソフトリセット

内部ステートマシンをリセットしますが、エニュメレーション情報は保持されます。特定のPHYエラーからの復帰に使用できます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット0 **CSRST**：コアソフトのリセット

以下のようにHCLKおよびPHYクロックドメインをリセットします。

割込みと、以下のビットを除くすべてのCSRレジスタビットをクリアします。

–OTG_PCGCCTLレジスタ のGATEHCLKビット

–OTG_PCGCCTLレジスタ のSTPPCLKビット

–OTG_HCFGレジスタ のFSLSPCSビット

–OTG_DCFGレジスタ のDSPDビット

–OTG_DCTLレジスタ のSDISビット

–OTG_GCCFGレジスタ

すべてのモジュールのステートマシン（AHBスレーブユニットを除く）がアイドル状態にリセットされ、すべての送信FIFOおよび受信FIFOがフラッシュされます。

AHB転送の最後のデータフェーズの完了後AHBマスタ上のすべてのトランザクションは速やかに終了されます。USB上のすべてのトランザクションをただちに終了します。

アプリケーションは、いつでもこのビットに書き込むことによって、コアをリセットできます。これはセルフクリアビットであり、コア内のすべての必要なロジックがリセットされた後、コアがこのビットをクリアします。コアの現在の状態によって、クリアには数クロックかかります。このビットがクリアされると、ソフトウェアはPHYドメインにアクセスできるようになるまで少なくとも3 PHYクロック待つ必要があります（同期遅延）。また、ソフトウェアは、操作を開始する前に、このレジスタのビット31が1にセットされていること（AHBマスタがアイドル状態）を確認する必要があります。

一般に、ソフトウェアリセットは、ソフトウェア開発時に使用され、ユーザが上記のUSB設定レジスタのPHY選択ビットを動的に変更するときにも使用されます。PHYを変更するときには、PHYの対応クロックが選択されPHYドメインで使用されます。新しいクロックが選択されたら、適切な動作のためにはPHYドメインをリセットする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

73.14.6 OTGコア割込みレジスタ[オルタネート]（OTG_GINTSTS）

ホストモードで有効です。デバイスモードについては次のセクションを参照してください。

このレジスタは、現在のモードも示します。rc_w1タイプの割込みステータスビットをクリアするには、アプリケーションは、そのビットに1を書き込む必要があります。

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割込みをかけます。

FIFOステータス割込みは読出し専用です。これらの割込みの処理中に、ソフトウェアがFIFOへの書込みまたはFIFOからの読出しを行うと、FIFO割込み条件は自動的にクリアされます。

初期化の前に割込みが発生するのを避けるためには、アプリケーションは、割込みビットのマスクを解除する前にOTG_GINTSTSレジスタをクリアする必要があります。

アドレスオフセット：0x014

リセット値：0x0400 0020

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	LPM INT	PTXFE	HCINT	HPRT INT	RST DET	DATAF SUSP	IPXFR	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	r	rc_w1	rc_w1	rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUM DNE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXF E	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット31 **WKUPINT**：レジューム／リモートウェイクアップ検出割込み

サスペンド（L2）またはLPM（L1）状態中のウェイクアップ割込み。

– サスペンド（L2）中

デバイスモードでは、USB上でレジュームが検出されたときにこの割込みがアサートされます。ホストモードでは、USB上でリモートウェイクアップが検出されたときにこの割込みがアサートされます。

– LPM（L1）中

この割込みは、USB上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット30 **SRQINT**：セッションリクエスト／新規セッション検出割込み

ホストモードでは、この割込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割込みは、V_{BUS} がBペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット29 **DISCINT**：切断検出割込み

デバイスの切断が検出されたときにアサートされます。

注： ホストモードでのみアクセス可能です。

ビット28 **CIDSCHG**：コネクタIDステータス変化

コアは コネクタIDステータスに変化があったとき、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット27 **LPMINT**：LPM割込み

デバイスモードでは、この割込みは、デバイスがLPMトランザクションを受信し、非エラー応答で応答したときにアサートされます。

ホストモードでは、この割込みは、デバイスがLPMトランザクションに非エラー応答で応答したとき、またはホストのコアがプログラムした回数（OTG_GLPMCFGのRETRYCNT）のLPMトランザクションを完了したときにアサートされます。

このフィールドは、OTG_GLPMCFGのLPMENビットが1にセットされたときにのみ有効です。

ビット26 **PTXFE**：周期的Tx FIFOエンプティ

周期的送信FIFOが半分エンプティか完全エンプティであり、周期的リクエストキューに少なくとも1つのエントリを書き込むためのスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタの周期的TxFIFOエンプティレベルビット（OTG_GAHBCFGのPTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット25 **HCINT**：ホストチャネル割込み

コアは、コアのチャネルの1つで割込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINTレジスタを読み出して、割込みが発生しているチャネルの正確な数を確認した後、対応するOTG_HCINTxレジスタを読み出して、割込みの正確な原因を確認する必要があります。アプリケーションはOTG_HCINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット24 **HPRTINT**：ホストポート割込み

コアは、ホストモードの OTG_HS コントローラポートの1つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRTレジスタを読み出して、この割込みの原因となった正確なイベントを特定する必要があります。アプリケーションはOTG_HPRTレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット23 **RSTDET**：リセット検出割込み

デバイスモードでは、この割込みは、デバイスがサスペンド状態にあるときに部分パワーダウンモードでUSB上にリセットが検出されたときにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット22 **DATAFSUSP**：データフェッチサスペンド

この割込みはDMAモードでのみ有効です。この割込みはTxFIFOスペースまたはリクエストキュースペースがないためコアがINエンドポイントに対するデータフェッチを停止したことを表示します。アプリケーションでは、エンドポイント不一致アルゴリズムに対してこの割込みを使います。例えば、エンドポイント不一致を検出後、アプリケーションは以下を実行します。

- グローバルな非周期的IN NAKハンドシェイクを設定
- INエンドポイントを無効化
- FIFOを一掃
- INトークンシーケンス学習キューからトークンシーケンスを決定
- エンドポイントを再度有効化

グローバル非周期的IN NAKがクリアされ、コアはINエンドポイントのデータをフェッチしておらず、かつINトークンを受信した場合、すなわちコアが「FIFOエンプティ時INトークン受信」割込みを生成した場合、グローバル非周期的IN NAKハンドシェイクをクリアします。次にOTGはNAK応答をホストへ送信します。このシナリオを回避するため、アプリケーションはOTG_GINTSTS内でFetSusp割込みをチェックすることができます。これにより、グローバルNAKハンドシェイクをクリアする前にFIFOが満杯であることが保証されます。あるいは、グローバルNAKハンドシェイクをクリアするとき、アプリケーションが「FIFOエンプティ中にINトークンを受信」割込みはマスクされますすることができます。

ビット21 **IPXFR**：不完全な周期的転送

ホストモードでは、コアは、現在のフレームでスケジュールされていて、まだペンディング中の不完全な周期的トランザクションが存在するとき、この割込みビットをセットします。

ビット20 **IISOIXFR**：不完全アイソクロナスIN転送

コアは、現在のフレームで転送が完了していないアイソクロナスINエンドポイントが少なくとも1つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット19 **OEPINT**：OUTエンドポイント割込み

コアは、コアのOUTエンドポイントの1つで割込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したOUTエンドポイントの正確な数を確認した後、対応するOTG_DOEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DOEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット18 **IEPINT**：INエンドポイント割込み

コアは、コアのINエンドポイントの1つで割込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したINエンドポイントの正確な数を確認した後、対応するOTG_DIEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DIEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EOPF**：周期的フレーム終了割込み

現在のフレームで、OTG_DCFGレジスタの周期的フレームインターバルフィールド（OTG_DCFGのPFIVLビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット14 **ISOODRP**：アイソクロナスOUTパケットドロップ割込み

コアは、アイソクロナスOUTエンドポイントの最大サイズのパケットに対応できるだけの十分なスペースがRx FIFOになかったためにRx FIFOにアイソクロナスOUTパケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット13 **ENUMDNE**：エニュメレーションの完了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTSレジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット12 **USBRST**：USBのリセット

コアはUSB上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット11 **USBSUSP**：USBのサスペンド

コアはUSB上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが1つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット10 **ESUSP**：早期サスペンド

コアはUSB上で3 ms間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **GONAKEFF**：グローバルOUT NAKの有効性

アプリケーションによってセットされたOTG_DCTLレジスタのセットグローバルOUT NAKビット（OTG_DCTLのSGONAKビット）がコアで有効になったことを示します。このビットは、OTG_DCTLレジスタのクリアグローバルOUT NAKビット（OTG_DCTLのCGONAKビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット6 **GINAKEFF**：グローバルIN非周期的NAKの有効性

アプリケーションによってセットされたOTG_DCTLレジスタのセットグローバル 非周期的IN NAKビット（OTG_DCTLのSGINAKビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバルIN NAKビットをサンプリングしました。このビットは、OTG_DCTLレジスタのクリアグローバル非周期的IN NAKビット（OTG_DCTLのCGＩNAKビット）をクリアすることによってクリアできます。

この割込みは、必ずしもNAKハンドシェイクがUSBに送出されることを意味しません。STALLビットはNAKビットより優先されます。

注： デバイスモードでのみアクセス可能です。

ビット5 **NPTXFE**：非周期的Tx FIFOエンプティ

この割込みは、非周期的Tx FIFOが半分エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも1つのエントリを書き込むスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタの非周期的TxFIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット4 **RXFLVL**：Rx FIFO非エンプティ

Rx FIFOから読み出されるべきペンディング中のパケットが少なくとも1つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット3 **SOF**：フレーム開始（スタート･オブ･フレーム）

ホストモードでは、SOF（FS）またはキープアライブ（LS）がUSBで送信されることを示すために、コアによってこのビットがセットされます。アプリケーションは、このビットに1を書き込むことによって、割込みをクリアする必要があります。

デバイスモードでは、USB上でSOFトークンが受信されたことを示すために、コアでこのビットがセットされます。アプリケーションは、OTG_DSTSレジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割込みは、コアがFSで動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると”1”を返すことがあります。パワーオンリセット後すぐに”1”が読み出された場合、そのレジスタのビットが、SOFが送信された（ホストモードの場合）またはSOFが受信された（デバイスモードの場合）ことを示すわけではありません。この割込みの読出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット2 **OTGINT**：OTG割込み

コアは、OTGプロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションはOTG割込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割込みの原因となった正確なイベントを確認する必要があります。アプリケーションはOTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット1 **MMIS**：モード不一致割込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY応答によってAHB上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

73.14.7 OTGコア割込みレジスタ[オルタネート]（OTG_GINTSTS）

デバイスモードで有効です。ホストモードについては前のセクションを参照してください。

このレジスタは、現在のモードも示します。rc_w1タイプの割込みステータスビットをクリアするには、アプリケーションは、そのビットに1を書き込む必要があります。

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割込みをかけます。

FIFOステータス割込みは読出し専用です。これらの割込みの処理中に、ソフトウェアがFIFOへの書込みまたはFIFOからの読出しを行うと、FIFO割込み条件は自動的にクリアされます。

初期化の前に割込みが発生するのを避けるためには、アプリケーションは、割込みビットのマスクを解除する前にOTG_GINTSTSレジスタをクリアする必要があります。

アドレスオフセット：0x014

リセット値：0x0400 0020

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	LPM INT	PTXFE	HCINT	HPRT INT	RST DET	DATAF SUSP	IN COMP ISO OUT	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	r	rc_w1	rc_w1	rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUM DNE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXF E	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット31 **WKUPINT**：レジューム／リモートウェイクアップ検出割込み

サスペンド（L2）またはLPM（L1）状態中のウェイクアップ割込み。

– サスペンド（L2）中

デバイスモードでは、USB上でレジュームが検出されたときにこの割込みがアサートされます。ホストモードでは、USB上でリモートウェイクアップが検出されたときにこの割込みがアサートされます。

– LPM（L1）中

この割込みは、USB上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット30 **SRQINT**：セッションリクエスト／新規セッション検出割込み

ホストモードでは、この割込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割込みは、V_{BUS} がBペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット29 **DISCINT**：切断検出割込み

デバイスの切断が検出されたときにアサートされます。

注： ホストモードでのみアクセス可能です。

ビット28 **CIDSCHG**：コネクタIDステータス変化

コアは コネクタIDステータスに変化があったとき、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット27 **LPMINT**：LPM割込み

デバイスモードでは、この割込みは、デバイスがLPMトランザクションを受信し、非エラー応答で応答したときにアサートされます。

ホストモードでは、この割込みは、デバイスがLPMトランザクションに非エラー応答で応答したとき、またはホストのコアがプログラムした回数（OTG_GLPMCFGのRETRYCNT）のLPMトランザクションを完了したときにアサートされます。

このフィールドは、OTG_GLPMCFGのLPMENビットが1にセットされたときにのみ有効です。

ビット26 **PTXFE**：周期的Tx FIFOエンプティ

周期的送信FIFOが半分エンプティか完全エンプティであり、周期的リクエストキューに少なくとも1つのエントリを書き込むためのスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタの周期的TxFIFOエンプティレベルビット（OTG_GAHBCFGのPTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット25 **HCINT**：ホストチャネル割込み

コアは、コアのチャネルの1つで割込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINTレジスタを読み出して、割込みが発生しているチャネルの正確な数を確認した後、対応するOTG_HCINTxレジスタを読み出して、割込みの正確な原因を確認する必要があります。アプリケーションはOTG_HCINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット24 **HPRTINT**：ホストポート割込み

コアは、ホストモードの OTG_HS コントローラポートの1つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRTレジスタを読み出して、この割込みの原因となった正確なイベントを特定する必要があります。アプリケーションはOTG_HPRTレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット23 **RSTDET**：リセット検出割込み

デバイスモードでは、この割込みは、デバイスがサスペンド状態にあるときに部分パワーダウンモードでUSB上にリセットが検出されたときにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット22 **DATAFSUSP**：データフェッチサスペンド

この割込みはDMAモードでのみ有効です。この割込みはTxFIFOスペースまたはリクエストキュースペースがないためコアがINエンドポイントに対するデータフェッチを停止したことを表示します。アプリケーションでは、エンドポイント不一致アルゴリズムに対してこの割込みを使います。例えば、エンドポイント不一致を検出後、アプリケーションは以下を実行します。

– グローバルな非周期的IN NAKハンドシェイクを設定
– INエンドポイントを無効化
– FIFOを一掃
– INトークンシーケンス学習キューからトークンシーケンスを決定
– エンドポイントを再度有効化

グローバル非周期的IN NAKがクリアされ、コアはINエンドポイントのデータをフェッチしておらず、かつINトークンを受信した場合、すなわちコアが「FIFOエンプティ時INトークン受信」割込みを生成した場合、グローバル非周期的IN NAKハンドシェイクをクリアします。次にOTGはNAK応答をホストへ送信します。このシナリオを回避するため、アプリケーションはOTG_GINTSTS内でFetSusp割込みをチェックすることができます。これにより、グローバルNAKハンドシェイクをクリアする前にFIFOが満杯であることが保証されます。あるいは、グローバルNAKハンドシェイクをクリアするとき、アプリケーションが「FIFOエンプティ中にINトークンを受信」割込みはマスクされますすることができます。

ビット21 **INCOMPISOOUT**：不完全アイソクロナスOUT転送

デバイスモードでは、コアは、現在のフレームで転送が完了していないアイソクロナスOUTエンドポイントが少なくとも1つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

ビット20 **IISOIXFR**：不完全アイソクロナスIN転送

コアは、現在のフレームで転送が完了していないアイソクロナスINエンドポイントが少なくとも1つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット19 **OEPINT**：OUTエンドポイント割込み

コアは、コアのOUTエンドポイントの1つで割込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したOUTエンドポイントの正確な数を確認した後、対応するOTG_DOEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DOEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット18 **IEPINT**：INエンドポイント割込み

コアは、コアのINエンドポイントの1つで割込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINTレジスタを読み出して、割込みが発生したINエンドポイントの正確な数を確認した後、対応するOTG_DIEPINTxレジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応するOTG_DIEPINTxレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EOPF**：周期的フレーム終了割込み

現在のフレームで、OTG_DCFGレジスタの周期的フレームインターバルフィールド（OTG_DCFGのPFIVLビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット14 **ISOODRP**：アイソクロナスOUTパケットドロップ割込み

コアは、アイソクロナスOUTエンドポイントの最大サイズのパケットに対応できるだけの十分なスペースがRx FIFOになかったためにRx FIFOにアイソクロナスOUTパケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット13 **ENUMDNE**：エニュメレーションの完了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTSレジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット12 **USBRST**：USBのリセット

コアはUSB上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット11 **USBSUSP**：USBのサスペンド

コアはUSB上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが1つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット10 **ESUSP**：早期サスペンド

コアはUSB上で3 ms間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **GONAKEFF**：グローバルOUT NAKの有効性

アプリケーションによってセットされたOTG_DCTLレジスタのセットグローバルOUT NAKビット（OTG_DCTLのSGONAKビット）がコアで有効になったことを示します。このビットは、OTG_DCTLレジスタのクリアグローバルOUT NAKビット（OTG_DCTLのCGONAKビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット6 **GINAKEFF**：グローバルIN非周期的NAKの有効性

アプリケーションによってセットされたOTG_DCTLレジスタのセットグローバル 非周期的IN NAKビット（OTG_DCTLのSGINAKビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバルIN NAKビットをサンプリングしました。このビットは、OTG_DCTLレジスタのクリアグローバル非周期的IN NAKビット（OTG_DCTLのCG I NAKビット）をクリアすることによってクリアできます。

この割込みは、必ずしもNAKハンドシェイクがUSBに送出されることを意味しません。STALLビットはNAKビットより優先されます。

注： デバイスモードでのみアクセス可能です。

ビット5 **NPTXFE**：非周期的Tx FIFOエンプティ

この割込みは、非周期的Tx FIFOが半分エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも1つのエントリを書き込むスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタの非周期的TxFIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット4 **RXFLVL**：Rx FIFO非エンプティ

Rx FIFOから読み出されるべきペンディング中のパケットが少なくとも1つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット3 **SOF**：フレーム開始（スタート･オブ･フレーム）

ホストモードでは、SOF（FS）またはキープアライブ（LS）がUSBで送信されることを示すために、コアによってこのビットがセットされます。アプリケーションは、このビットに1を書き込むことによって、割込みをクリアする必要があります。

デバイスモードでは、USB上でSOFトークンが受信されたことを示すために、コアでこのビットがセットされます。アプリケーションは、OTG_DSTSレジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割込みは、コアがFSで動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると”1”を返すことがあります。パワーオンリセット後すぐに”1”が読み出された場合、そのレジスタのビットが、SOFが送信された（ホストモードの場合）またはSOFが受信された（デバイスモードの場合）ことを示すわけではありません。この割込みの読出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット2 **OTGINT**：OTG割込み

コアは、OTGプロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションはOTG割込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割込みの原因となった正確なイベントを確認する必要があります。アプリケーションはOTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット1 **MMIS**：モード不一致割込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY応答によってAHB上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

73.14.8 OTG割込みマスクレジスタ[オルタネート]（OTG_GINTMSK）

ホストモードで有効です。デバイスモードについては次のセクションを参照してください。

このレジスタは、コア割込みレジスタとともに働き、アプリケーションに割込みをかけます。割込みビットがマスクされると、そのビットに関連する割込みは生成されません。ただし、その割込みに対応するコア割込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

アドレスオフセット：0x018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	Res.	Res.	IPXFRM	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	r			rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
										rw	rw	rw	rw	rw	

ビット31 **WUIM**：レジューム／リモートウェイクアップ検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット30 **SRQIM**：セッションリクエスト／新規セッション検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット29 **DISCINT**：切断検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット28 **CIDSCHGM**：コネクタIDステータス変化マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット27 **LPMINTM**：LPM割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット26 **PTXFEM**：周期的Tx FIFOエンプティマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット25 **HCIM**：ホストチャネル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット24 **PRTIM**：ホストポート割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット23:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **IPXFRM**：不完全周期的転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット20:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **NPTXFEM**：非周期的Tx FIFOエンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット4 **RXFLVLM**：受信FIFO非エンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット3 **SOFM**：フレーム開始マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット2 **OTGINT**：OTG割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット1 **MMISM**：モード不一致割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット0 予約済みであり、リセット値に保持する必要があります。

73.14.9 OTG割込みマスクレジスタ[オルタネート]（OTG_GINTMSK）

デバイスモードで有効です。ホストモードについては前のセクションを参照してください。

このレジスタは、コア割込みレジスタとともに働き、アプリケーションに割込みをかけます。割込みビットがマスクされると、そのビットに関連する割込みは生成されません。ただし、その割込みに対応するコア割込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

アドレスオフセット：0x018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	Res.	CIDSCHGM	LPMINTM	Res.	Res.	Res.	RSTDETM	FSUSPM	IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.
rw	rw		rw	rw				rw	rw	rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	Res.	RXFLVLM	SOFM	OTGINT	MMISM	Res.
rw	rw	rw	rw	rw	rw			rw	rw		rw	rw	rw	rw	

ビット31 **WUIM**：レジューム／リモートウェイクアップ検出割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット30 **SRQIM**：セッションリクエスト／新規セッション検出割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット29 予約済みであり、リセット値に保持する必要があります。

ビット28 **CIDSCHGM**：コネクタIDステータス変化マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット27 **LPMINTM**：LPM割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット26:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **RSTDETM**：リセット検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット22 **FSUSPM**：データフェッチサスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット21 **IISOOXFRM**：不完全アイソクロナスOUT転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット20 **IISOIXFRM**：不完全アイソクロナスIN転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット19 **OEPINT**：OUTエンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット18 **IEPINT**：INエンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット17:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EOPFM**：周期的フレーム終了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット14 **ISOODRPM**：アイソクロナスOUTパケットドロップ割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット13 **ENUMDNEM**：エニュメレーション完了マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット12 **USBRST**：USBリセットマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット11 **USBSUSPM**：USBサスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット10 **ESUSPM**：早期サスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット9:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **GONAKEFFM**：グローバルOUT NAK有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット6 **GINAKEFFM**：グローバル非周期的IN NAK有効マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット5 予約済みであり、リセット値に保持する必要があります。

ビット4 **RXFLVLM**：受信FIFO非エンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット3 **SOFM**：フレーム開始マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット2 **OTGINT**：OTG割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット1 **MMISM**：モード不一致割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット0 予約済みであり、リセット値に保持する必要があります。

73.14.10 OTG受信ステータスデバッグ読出しレジスタ[オルタネート]（OTG_GRXSTSR）

この説明は、デバイスモードのレジスタOTG_GRXSTSR用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信FIFOの最上部の内容が返されます。

コアは、受信FIFOがエンプティのときには、受信ステータスの読出しを無視して0x0000 0000の値を返します。

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **STSPHST**：ステータスフェーズ開始

コントロール書込み転送のステータスフェーズの開始を示します。このビットは、OUT転送完了のPKTSTSパターンに合わせてセットされます。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24:21 **FRMNUM[3:0]**：フレーム番号

これはUSB上でパケットを受信するフレーム番号の下位4bitです。このフィールドは、アイソクロナスOUTエンドポイントがサポートされる場合にのみサポートされます。

ビット20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0001：グローバルOUT NAK（割込みをトリガします）。
0010：OUTデータパケットが受信されました。
0011：OUT転送が完了しました（割込みをトリガします）。
0100：SETUPトランザクションが完了しました（割込みをトリガします）。
0110：SETUPデータパケットが受信されました。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID
受信したOUTデータパケットのデータPIDを示します。
00：DATA0
10：DATA1
01：DATA2
11：MDATA

ビット14:4 **BCNT[10:0]**：バイト数
受信したデータパケットのバイト数を示します。

ビット3:0 **EPNUM[3:0]**：エンドポイント番号
現在の受信パケットが属するエンドポイント番号を示します。

73.14.11 OTG受信ステータスデバッグ読出しレジスタ[オルタネート]（OTG_GRXSTSR）

この説明は、ホストモードのレジスタOTG_GRXSTSR用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信FIFOの最上部の内容が返されます。

コアは、受信FIFOがエンプティのときには、受信ステータスの読出しを無視して0x0000 0000の値を返します。

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0010：INデータパケットが受信されました。
0011：IN転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信パケットのデータPIDを示します。

00：DATA0

10：DATA1

01：DATA2

11：MDATA

ビット14:4 **BCNT[10:0]**：バイト数

受信したINデータパケットのバイト数を示します。

ビット3:0 **CHNUM[3:0]**：チャネル番号

現在の受信パケットが属するチャネル番号を示します。

73.14.12 OTGステータス読出しおよびポップレジスタ（OTG_GRXSTSP）

この説明は、デバイスモードのレジスタOTG_GRXSTSP用です。

読み出すと受信FIFOの最上部の内容が返されるOTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFOの最上部のデータエントリもポップアウトされます。

コアは、受信FIFOがエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000の値を返します。アプリケーションは、コア割込みレジスタの受信FIFOノットエンプティビット（OTG_GINTSTSのRXFLVLビット）がアサートされたときにのみ、受信ステータスFIFOをポップしなければなりません。

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **STSPHST**：ステータスフェーズ開始

コントロール書込み転送のステータスフェーズの開始を示します。このビットは、OUT転送完了のPKTSTSパターンに合わせてセットされます。

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24:21 **FRMNUM[3:0]**：フレーム番号

これはUSB上でパケットを受信するフレーム番号の下位4bitです。このフィールドは、アイソクロナスOUTエンドポイントがサポートされる場合にのみサポートされます。

ビット20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。

0001：グローバルOUT NAK（割込みをトリガします）。

0010：OUTデータパケットが受信されました。

0011：OUT転送が完了しました（割込みをトリガします）。

0100：SETUPトランザクションが完了しました（割込みをトリガします）。

0110：SETUPデータパケットが受信されました。

その他：予約済み

ビット16:15 **DPID[1:0]**：データPID

受信したOUTデータパケットのデータPIDを示します。

00：DATA0

10：DATA1

01：DATA2

11：MDATA

ビット14:4 **BCNT[10:0]**：バイト数

受信したデータパケットのバイト数を示します。

ビット3:0 **EPNUM[3:0]**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

73.14.13 OTGステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）

この説明は、ホストモードのレジスタOTG_GRXSTSP用です。

読み出すと受信FIFOの最上部の内容が返されるOTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFOの最上部のデータエントリもポップアウトされます。

コアは、受信FIFOがエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000の値を返します。アプリケーションは、コア割込みレジスタの受信FIFOノットエンプティビット（OTG_GINTSTSのRXFLVLビット）がアサートされたときにのみ、受信ステータスFIFOをポップしなければなりません。

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0010：INデータパケットが受信されました。
0011：IN転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済み

ビット16:15 **DPID[1:0]**：データPID
受信パケットのデータPIDを示します。
00：DATA0
10：DATA1
01：DATA2
11：MDATA

ビット14:4 **BCNT[10:0]**：バイト数
受信したINデータパケットのバイト数を示します。

ビット3:0 **CHNUM[3:0]**：チャネル番号
現在の受信パケットが属するチャネル番号を示します。

73.14.14 OTG受信FIFOサイズレジスタ（OTG_GRXFSIZ）

アプリケーションはRxFIFOに割り当てられなければならないRAMサイズをプログラムできます。

アドレスオフセット：0x024

リセット値：0x0000 0400

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **RXFD[15:0]**：Rx FIFOの深さ
この値は32bitワード単位です。
最大値は1024です。
プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

73.14.15 OTGホスト非周期的送信FIFOサイズレジスタ[オルタネート]（OTG_HNPTXFSIZ）

ホストモードで有効です。デバイスモードについては次のセクションを参照してください。

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NPTXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **NPTXFD[15:0]**：非周期的Tx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。
プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット15:0 **NPTXFSA[15:0]**：非周期的送信RAM開始アドレス
このフィールドは、非周期的送信FIFO RAMのメモリ開始アドレスを設定します。

73.14.16 エンドポイント0送信FIFOサイズ[オルタネート]（OTG_DIEPTXF0）

デバイスモードで有効です。ホストモードについては前のセクションを参照してください。

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TX0FD [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TX0FSA [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **TX0FD[15:0]**：エンドポイント0 Tx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。
プログラムされた値は、使用可能なFIFOメモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット15:0 **TX0FSA[15:0]**：エンドポイント0送信RAM開始アドレス
このフィールドは、エンドポイント0送信FIFO RAMのメモリ開始アドレスを設定します。

73.14.17 OTG非周期的送信FIFO／キューステータスレジスタ（OTG_HNPTXSTS）

注： **デバイスモードでは、このレジスタは無効です。**

この読出し専用レジスタは、非周期的Tx FIFOおよび非周期的送信リクエストキューの空きスペース情報を含みます。

アドレスオフセット：0x02C

リセット値：0x0008 0400

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	NPTXQTOP[6:0]							NPTQXSAV[7:0]							
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSAV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:24 **NPTXQTOP[6:0]：**非周期的送信リクエストキューの先頭

MACによって処理中の非周期的Txリクエストキューのエントリ。

ビット30:27：チャネル／エンドポイント番号

ビット26:25：

XXXX00X：IN/OUTトークン

XXXX01X：ゼロ長送信パケット（デバイスIN／ホストOUT）

XXXX11X：チャネル停止コマンド

ビット24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット23:16 **NPTQXSAV[7:0]：**非周期的送信リクエストキューの使用可能スペース

非周期的送信リクエストキューの使用可能な空きスペースの量を示します。このキューはINおよびOUTリクエストの両方を保持します。

0：非周期的送信リクエストキューはフルです。

1：1位置が使用可能です。

2：2位置が使用可能です。

n：n位置が使用可能です （0 ≤ n ≤ 8）。

その他：予約済み

ビット15:0 **NPTXFSAV[15:0]：**使用可能な非周期的Tx FIFOスペース

非周期的Tx FIFO内で使用できる空きスペースの量を示します。

値は32bitワード単位です。

0：非周期的Tx FIFOはフルです。

1：1ワードが使用可能です。

2：2ワードが使用可能です。

n：nワードが使用可能です （0 ≤ n ≤ 512）

その他：予約済み

73.14.18 OTG一般コア設定レジスタ（OTG_GCCFG）

このレジスタは、ホストモードとデバイスモードで使用できます。

アドレスオフセット：0x038

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FORCEHOSTPD	VBVALOVEN	VBVALOVAL	SDEN	VBDEN	PDEN	DCDEN	HVDMSRCEN	HCDPDETEN	HCDPEN
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SESSVLD	FSVMINUS	FSVPLUS	CHGDET
												r	r	r	r

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 予約済みであり、リセット値に保持する必要があります。

ビット27 予約済みであり、リセット値に保持する必要があります。

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25 **FORCEHOSTPD**：ホストモードプルダウンの強制適用

IDピン機能が有効化されている場合、DPおよびDMのホストモードプルダウンが自動的にアクティブ化されます。ただし、自動的にアクティブ化されなくても、ホストモードが必要な場合は、このビットを使用してプルダウンを強制的にアクティブ状態にする必要があります。

0：ホストモードプルダウンを強制適用しません。

1：ホストモードプルダウンを強制適用します。

ビット24 **VBVALOVEN**：VBUS Bセッション検出のソフトウェア上書きの有効化

0：ハードウェアを使用します。

1：VBVALOVALを使用してBセッションのアクティブを指示します。

ビット23 **VBVALOVAL**：VBUS Bセッション検出のソフトウェア上書き値

0：Bセッションは非アクティブです。

1：Bセッションはアクティブです。

ビット22 **SDEN**：2次検出の有効化

0：2次検出は無効です。

1：2次検出は有効です。

ビット21 **VBDEN**：VBUS検出の有効化

VBUSレベル検出用コンパレータを有効にして、VBUSの存在を検出したり、OTG動作を実行したりします。

0：VBUS検出は無効です。

1：VBUS検出は有効です。

ビット20 **PDEN**：1次検出の有効化

0：1次検出は無効です。

1：1次検出は有効です。

ビット19 **DCDEN**：データ接触検出の有効化

0：データ接触検出は無効です。

1：データ接触検出は有効です。

ビット18 **HVDMSRCEN**：DMでのホストCDPポート電圧ソースの有効化
0：DM電圧ソースは無効です。
1：DM電圧ソースは有効です。

ビット17 **HCDPDETEN**：DPでのホストCDPポート電圧検出器の有効化
0：DP電圧検出は無効です。
1：DP電圧検出は有効です。

ビット16 **HCDPEN**：ホストCDP動作の有効化
0：CDP動作を無効化します。
1：CDP動作を有効化します。

ビット15:5 予約済みであり、リセット値に保持する必要があります。

ビット4 予約済みであり、リセット値に保持する必要があります。

ビット3 **SESSVLD**：VBUSセッションインジケータ
VBUSがVBUSセッション閾値を上回っているかどうかを示します。
0：VBUSがVBUSセッション閾値を下回っています。
1：VBUSがVBUSセッション閾値を上回っています。

ビット2 **FSVMINUS**：シングルエンドDMインジケータ
このビットでは、DMの電圧レベルを規定します（その他、BC v1.2基準での定義に従い、V_{LGC} 閾値との比較の結果も）。
0：DM電圧は低レベル
1：DM電圧は高レベル

ビット1 **FSVPLUS**：シングルエンドDPインジケータ
このビットでは、DPの電圧レベルを規定します（その他、BC v1.2基準での定義に従い、V_{LGC} 閾値との比較の結果も）。
0：DM電圧は低レベル
1：DM電圧は高レベル

ビット0 **CHGDET**：チャージャ検出、現在のモードの結果（1次または2次）
0：ピンでの低い値
1：ピンでの高い値

73.14.19 OTGコアIDレジスタ（OTG_CID）

これは、リセット値として製品IDを含むレジスタです。

アドレスオフセット：0x03C

リセット値：0x0000 5000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRODUCT_ID[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRODUCT_ID[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **PRODUCT_ID[31:0]**：製品IDフィールド
アプリケーションによってプログラム可能なIDフィールド。

73.14.20 OTGコアLPM設定レジスタ（OTG_GLPMCFG）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EN BESL	LPMRCNTSTS[2:0]			SND LPM	LPMRCNT[2:0]			LPMCHIDX[3:0]				L1RSM OK
			rw	r	r	r	rs	rw	rw	rw	rw	rw	rw	rw	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLP STS	LPMRSP[1:0]		L1DS EN	BESLTHRS[3:0]				L1SS EN	REM WAKE	BESL[3:0]				LPM ACK	LPM EN
r	r	r	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28 **ENBESL**：ベストエフォート型サービスのレイテンシの有効化

このビットはLPM正誤表で定義されているようにBESL機能を有効にします。

0： コアは以下の文書に記載されているように機能します。
USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007

1： コアはLPM正誤表に記載されているように機能します。
USB 2.0 ECNの正誤表：リンク電源管理（LPM）- 2007年7月

注： この文書では更新された動作（LPM正誤表に記載）のみが考慮されているため、アプリケーションのソフトウェアによりENBESLビットに1をセットする必要があります。

ビット27:25 **LPMRCNTSTS[2:0]**：LPMリトライカウントステータス

現在のLPMシーケンスでは、送信すべきLPMホストの再試行回数がまだ残っています。

注： ホストモードでのみアクセス可能です。

ビット24 **SNDLPM**：送信LPMトランザクション

アプリケーションソフトウェアがこのビットをセットすると、2つのトークンEXTおよびLPMを含むLPMトランザクションが送信されます。デバイスから有効な応答（STALL、NYET、またはACK）を受信した時点、またはコアがプログラムされた回数のLPM再試行を送信し終えた時点で、このビットはハードウェアによってクリアされます。

注： このビットは、ホストがローカルポートに接続されている場合のみセットする必要があります。

注： ホストモードでのみアクセス可能です。

ビット23:21 **LPMRCNT[2:0]**：LPMリトライカウント

デバイスがERROR応答を返すと、このビットは追加のLPM再試行回数となります。追加の再試行は、有効なデバイスからの応答（STALL、NYET、またはACK）が受信されるまで、ホストによって実行されます。

注： ホストモードでのみアクセス可能です。

ビット20:17 **LPMCHIDX[3:0]**：LPMチャネルインデックス

ローカルデバイスへのLPMトランザクションの送信中にLPMトランザクションを適用するべきチャネルの番号を示します。LPMチャネルインデックスに基づき、コアは自動的に対応するチャネルでプログラムされたデバイスアドレスとエンドポイント番号をLPMトランザクションに挿入します。

注： ホストモードでのみアクセス可能です。

ビット16 **L1RSMOK**：スリープ状態レジュームOK

デバイスまたはホストがスリープ状態からレジュームを開始できることを示します。このビットはLPMスリープ状態（L1）のときに有効です。このビットは50 µs（$T_{L1Residency}$）の遅延後にスリープモードでセットされます。

このビットは、SLPSTS = 0のときにリセットされます。

1：アプリケーションまたはホストはスリープ状態からレジュームを開始できます。

0：アプリケーションまたはホストはスリープ状態からレジュームを開始できません。

ビット15 **SLPSTS**：ポートスリープステータス

デバイスモード：

このビットは、USBバス上で スリープ 状態が発生している限り、セットされます。このコアは、LPMトランザクションにACK応答が送信され、$T_{L1TokenRetry}$ タイマが時間切れになっている場合に、スリープ 状態に入ります。PHYクロックを停止させるには、アプリケーションでOTG_PCGCCTLのSTPPCLKビットをセットすることにより、PHYサスペンド入力信号をアサートする必要があります。

アプリケーションはスリープへの遷移を確認する際に、LPMRSPのACKではなく、SLPSTSを信頼する必要があります。

コアは、以下の場合にスリープ状態から抜け出します。

– USBライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL の RWUSIG ビットへ書き込むときやデバイスをリセットまたはソフト切断するとき。

ホストモード：

デバイスからACK応答を受け取ったローカルポートへのコアによるLPM トランザクション の成功の副作用として、ホストはスリープ状態（L1）へ遷移します。このビットの読出し値は、ポートの現在のスリープ状態を反映しています。

コアは、次の動作後にこのビットをクリアします。

– コアがリモートL1ウェイクアップ信号を検出

– アプリケーションで、OTG_HPRT レジスタの PRST ビットまたは PRES ビットをセットします、または

– アプリケーションで、コア割込みレジスタの L1 レジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（OTG_GINTSTSのWKUPINTまたはDISCINTビット）をセット

0：コアはL1状態ではありません。

1：コアはL1状態です。

ビット14:13 **LPMRSP[1:0]**：LPM応答

デバイスモード：

受信したLPMトランザクションへのコアの応答は、これら2つのビットに反映されます。

ホストモード：

ローカルデバイスから受け取ったLPMトランザクションへのハンドシェイク応答

11：ACK

10：NYET

01：STALL

00：ERROR（ハンドシェイク応答なし）

ビット12 **L1DSEN**：L1ディープスリープの有効化

L1スリープモードにおけるPHYのサスペンドを有効にします。L1 SLEEPモード時に最大限の節電を行うために、いかなる場合もアプリケーションのソフトウェアによって、このビットに「1」をセットする必要があります。

ビット11:8 **BESLTHRS[3:0]**：BESL閾値

デバイスモード：

BESL値がこのBESL_Thres[3:0] フィールドの規定値以上である場合、コアはPHYをL1状態でディープ低消費電力モードに移行させます。

ホストモード：

コアはPHYをL1状態でディープ低消費電力モードに移行させます。BESLTHRS[3:0] は、デバイスから開始されたレジュームを検出した場合に、レジュームシグナリングに USB バス上のホスト（$T_{L1HubDrvResume2}$）が反映されるまでの時間を指定します。

ホストモードでは、BESLTHRS には 1100b より大きい値を設定してはなりません。この値が$T_{L1HubDrvResume2}$の最大値を超えてしまうからです。

Thres[3:0] ホストモードレジュームシグナリング時間（µs）：

0000：75

0001：100

0010：150

0011：250

0100：350

0101：450

0110：950

その他の値は予約済みです。

ビット7 **L1SSEN**：L1シャロースリープの有効化

L1 SLEEPモードにおけるPHYのサスペンドを有効にします。L1 SLEEPモード時に最大限の節電を行うために、いかなる場合もアプリケーションのソフトウェアによって、このビットに「1」をセットする必要があります。

ビット6 **REMWAKE**：bRemoteWake値

ホストモード：

LPMトランザクションのwIndexフィールドの送信されるリモートウェイクアップの値

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、またはSTALL応答がLPMトランザクションに送信される場合に、受信したLPMトークンbRemoteWake bmAttributeで更新されます。

ビット5:2 **BESL[3:0]**：ベストエフォート型サービスレイテンシ

ホストモード：

LPMトランザクションの送信されるBESLの値この値は、ホストから開始されたレジュームの処理時間（$T_{L1HubDrvResume1}$）の間にレジュームを開始するためにも使用されます。

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、またはSTALL応答がLPMトランザクションに送信される場合に、受信したLPMトークンBESL bmAttributeで更新されます。

BESL[3:0]T_{BESL} (µs)

0000：125
0001：150
0010：200
0011：300
0100：400
0101：500
0110：1000
0111：2000
1000：3000
1001：4000
1010：5000
1011：6000
1100：7000
1101：8000
1110：9000
1111：10000

ビット1 **LPMACK**：LPMトークン確認応答の有効化

デバイスのアプリケーションソフトウェアによって事前にプログラムされたLPMトークンへのハンドシェイク応答

1：ACK

ACKは事前にプログラムされているものの、コア･デバイスは成功したLPMトランザクションに対してのみACK応答を返します。LPMトランザクションは次の場合に成功します。

– EXT トークンまたは LPM トークンのどちらにも PID/CRC5 エラーがない（それ以外の場合はERROR）。
– LPMトランザクションで受信した有効なbLinkState = 0001B（L1）（それ以外の場合はSTALL）。
– 送信シーケンスにペンディング中のデータがない（それ以外の場合はNYET）。

0：NYET

事前にプログラムされたソフトウェアビットは、次の場合に、LPMトークンへの応答に上書きされます。

– 受信したbLinkStateがL1でない場合（STALL応答）、または
– いずれかのLPMトークンパケットで、破損によるエラーが検出された場合（エラー応答）

注： デバイスモードでのみアクセス可能です。

ビット0 **LPMEN**：LPMサポートの有効化

アプリケーションは、このビットを使用して、OTG_HS コアLPM機能を制御します。

コアが非LPM対応ホストとして動作する場合、ホストは接続されたデバイスまたはハブにLPMモードを有効にするように要求することはできません。

コアが非LPM対応デバイスとして動作する場合、デバイスはいかなるLPMトランザクションにも応答できません。

0：LPM機能は無効です。
1：LPM機能は有効です。

73.14.21 OTGホスト周期的送信FIFOサイズレジスタ（OTG_HPTXFSIZ）

アドレスオフセット：0x100

リセット値：0x0400 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PTXFSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTXSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **PTXFSIZ[15:0]**：ホスト周期的Tx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。

ビット15:0 **PTXSA[15:0]**：ホスト周期的Tx FIFOの開始アドレス
このフィールドは、周期的送信FIFO RAMのメモリ開始アドレスを設定します。

73.14.22 OTGデバイスINエンドポイント送信FIFO xサイズレジスタ（OTG_DIEPTXFx）

アドレスオフセット：0x104 + 0x04 * (x-1)(x = 1から8)

リセット値：0x0200 0400, 0x0200 0600, 0x0200 0800, 0x0200 0A00, 0x0200 0C00, 0x0200 0E00, 0x0200 1000, 0x0200 1200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INEPTXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTXSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **INEPTXFD[15:0]**：INエンドポイントTx FIFOの深さ
この値は32bitワード単位です。
最小値は16です。

ビット15:0 **INEPTXSA[15:0]**：INエンドポイントFIFOx送信RAM開始アドレス
このフィールドはINエンドポイント送信FIFOxのメモリ開始アドレスを含みます。アドレスは32bitメモリロケーションに整列されている必要があります。

73.14.23 ホストモードレジスタ

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

ホストモードレジスタは、ホストモードでのコアの動作に影響を与えます。.ホストモードレジスタは、結果が定義されていないため、デバイスモードではアクセスしないでください。ホストモードレジスタは、次のように分類することができます。

73.14.24 OTGホスト設定レジスタ（OTG_HCFG）

このレジスタは、電源投入後のコアを設定します。ホストを初期化した後は、このレジスタを変更しないでください。

アドレスオフセット：0x400

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLSPCS[1:0]	
													r	rw	rw

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **FSLSS**：FSおよびLSのみのサポート

アプリケーションは、このビットを使用して、コアのエニュメレーション速度を制御します。このビットを使用して、アプリケーションは、接続されたデバイスがHSトラフィックをサポートする場合でも、FSホストとしてコアにエニュメレーションさせることができます。初期プログラミングの後は、このフィールドを変更しないでください。

ビット1:0 **FSLSPCS[1:0]**：FS/LS PHYクロックの選択

条件：FSホストモード

01：PHYクロックは48 MHzで動作

その他：予約済み

条件：LSホストモード

00：予約済み

01：PHYクロック周波数として48 MHzを選択

10：PHYクロック周波数として6 MHzを選択

11：予約済み

注： FSLSPCS は、接続されたデバイスの速度に応じて、接続イベント時にセットする必要があります（このビットを変更した後、ソフトウェアリセットを実行する必要があります）。

73.14.25 OTGホストフレームインターバルレジスタ（OTG_HFIR）

このレジスタは OTG_HS コントローラがエニュメレーションした現在のスピードでのフレームインターバル情報を格納します。

アドレスオフセット：0x404

リセット値：0x0000 EA60

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLD CTRL
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRIVL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **RLDCTRL**：再ロード制御

このビットを使用して、ランタイム時にHFIRレジスタの動的再ロードが可能です。

0：HFIRは動的に再ロードすることができません。

1：HFIRはランタイム時に動的に再ロードすることができます。

このビットは、初期設定時にプログラムする必要があり、ランタイム時にそのを変更してはなりません。

注意： RLDCTRL = 0は推奨しません。

ビット15:0 **FRIVL[15:0]**：フレームインターバル

アプリケーションがこのフィールドにプログラムする値は、2つの連続したマイクロSOF（HS）またはキープアライブトークン（LS）の間のインターバルを指定します。このフィールドは、必要なフレームインターバルを構成するPHYクロック数を含みます。アプリケーションは、ホストポート制御およびステータスレジスタのポートイネーブルビット（OTG_HPRTのPENAビット）がセットされた後のみ、このレジスタに値を書き込むことができます。値がプログラムされていない場合、コアは、ホスト設定レジスタのFS/LS PHYクロック選択フィールド（OTG_HCFGのFSLSPCS）で指定されたPHYクロックに基づいて値を計算します。初期設定の後は、RLDCTRLビットがセットされている場合を除いて、このフィールドの値を変更しないでください。そのような場合、FRIVLには 各SOFイベントが再ロードされます。

–フレームインターバル = 125µs × (FRIVL - 1)ハイスピード動作

–フレームインターバル = 1ms × (FRIVL - 1)ロー／フルスピード動作

73.14.26 OTGホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）

このレジスタは、現在のフレーム番号を示します。また、現在のフレームの残り時間（PHYクロック数）も示します。

アドレスオフセット：0x408

リセット値：0x0000 3FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FTREM[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRNUM[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **FTREM[15:0]**：フレームの残り時間

現在のフレームの残り時間をPHYクロック数で示します。このフィールドはPHYクロックごとにデクリメントされます。値がゼロになると、このフィールドにはフレームインターバルレジスタの値が再ロードされ、新しいSOFがUSBに送信されます。

ビット15:0 **FRNUM[15:0]**：フレーム番号

このフィールドはUSB上で新しいSOFが送信されるたびにインクリメントされ、0x3FFFに達すると0にクリアされます。

73.14.27 OTGホスト周期的送信FIFO ／キューステータスレジスタ（OTG_HPTXSTS）

この読出し専用レジスタは、周期的Tx FIFOおよび周期的送信リクエストキューの空きスペース情報を含みます。

アドレスオフセット：0x410

リセット値：0x0008 0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PTXQTOP[7:0]								PTXQSAV[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTXFSAVL[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 **PTXQTOP[7:0]：**周期的送信リクエストキューの先頭

これはMACによって処理中の周期的Txリクエストキューのエントリを示します。
このレジスタは、デバッグに使用されます。
ビット31：奇数／偶数フレーム
0XXXXXXX：偶数フレームで送信
1XXXXXXX：奇数フレームで送信
ビット30:27：チャネル／エンドポイント番号
ビット26:25：タイプ
XXXXX00X：入力／出力
XXXXX01X：長さゼロのパケット
XXXXX11X：チャネルコマンドの無効化
ビット24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット23:16 **PTXQSAV[7:0]：**周期的送信リクエストキューの使用可能スペース

周期的送信リクエストキューで書込みに使用できる空きロケーションの数を示します。このキューはINおよびOUTリクエストの両方を保持します。
0：周期的送信リクエストキューはフルです。
1：1位置が使用可能です。
2：2位置が使用可能です。
n：n位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット15:0 **PTXFSAVL[15:0]：**周期的送信データFIFOの使用可能スペース

周期的Tx FIFOで書込みに使用できる空きロケーションの数を示します。
値は32bitワード単位です。
0：周期的Tx FIFOはフルです。
1：1ワードが使用可能です。
2：2ワードが使用可能です。
n：nワードが使用可能です（0 ≤ n ≤ PTXFD）。
その他：予約済み

73.14.28 OTGホスト全チャネル割込みレジスタ（OTG_HAINT）

チャネル上で重大なイベントが発生すると、ホスト全チャネル割込みレジスタは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）を使用して、アプリケーションに割込みをかけます。これを図917に示します。チャネルごとに1つの割込みビットがあります（最大16bit）。このレジスタのビットは、アプリケーションが対応するホストチャネルx割込みレジスタのビットをセット／クリアしたときにセット／クリアされます。

アドレスオフセット：0x414

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HAINT[15:0]**：チャネル割込み

チャネル0に対してビット0、チャネル15に対してビット15

73.14.29 OTGホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）

ホスト全チャネル割込みマスクレジスタは、ホスト全チャネル割込みレジスタとともに働き、チャネル上でイベントが発生したときに、アプリケーションに割込みをかけます。チャネルごとに1つの割込みマスクビットがあります（最大16bit）。

アドレスオフセット：0x418

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINTM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **HAINTM[15:0]**：チャネル割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

チャネル0に対してビット0、チャネル15に対してビット15

73.14.30 OTGホストポート制御およびステータスレジスタ（OTG_HPRT）

このレジスタは、ホストモードでのみ使用可能です。現時点ではOTGホストは1つのポートのみをサポートしています。

1つのレジスタは、ポートごとのUSBリセット、イネーブル、サスペンド、レジューム、接続ステータス、テストモードなどのUSBポート関連情報を保持します。図917に示されています。このレジスタのrc_w1bitは、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTSのHPRTINTビット）を使用して、アプリケーションへの割込みをトリガできます。ポート割込みの場合、アプリケーションは、このレジスタを読み出して、割込みの原因となったビットをクリアする必要があります。rc_w1bitの場合、アプリケーションは、このビットに1を書き込んで、割込みをクリアする必要があります。

アドレスオフセット：0x440

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD[1:0]		PTCTL[3]
													r	r	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTCTL[2:0]			PPWR	PLSTS[1:0]		Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS
rw	rw	rw	rw	r	r		rw	rs	rw	rc_w1	r	rc_w1	rc_w1	rc_w1	r

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:17 **PSPD[1:0]**：ポートスピード

このポートに接続されたデバイスのスピードを示します。

01：フルスピード

10：ロースピード

11：予約済み

00：ハイスピード

ビット16:13 **PTCTL[3:0]**：ポートテスト制御

アプリケーションがこのフィールドにゼロ以外の値を書き込んで、ポートをテストモードにすると、対応するパターンがポートにシグナリングされます。

0000：テストモードは無効

0001：Test_Jモード

0010：Test_Kモード

0011：Test_SE0_NAKモード

0100：Test_Packetモード

0101：Test_Force_Enable

その他：予約済み

ビット12 **PPWR**：ポートの電力供給

アプリケーションは、このフィールドを使用して、このポートへの電力供給を制御し、コアは、過電流条件が発生すると、このビットをクリアします。ただし、このビットにより、コネクタVBUSの電圧を直接アクティブ化することはできません。

0：パワーオフ

1：パワーオン

ビット11:10 **PLSTS[1:0]**：ポートラインステータス

USBデータラインの現在のロジックレベルを示します。

ビット10：OTG_DPのロジックレベル

ビット11：OTG_DMのロジックレベル

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **PRST**：ポートのリセット

アプリケーションがこのビットをセットすると、このポートでリセットシーケンスが開始されます。アプリケーションは、リセットシーケンスの完了後、リセット時間を決めて、このビットをクリアする必要があります。

0：ポートはリセット中ではありません。

1：ポートはリセット中です。

アプリケーションは、ポートのリセットが開始されるまで、少なくとも10 msの間、このビットをセット状態に保つ必要があります。USB規格では最大限度が規定されていませんが、アプリケーションは、このビットをクリアする前に、必要最小時間に加えて、さらに10 msの間、このビットをセット状態に保つことができます。

ハイスピード：50ms

フルスピード／ロースピード：10ms

ビット7 **PSUSP**：ポートのサスペンド

アプリケーションは、このビットをセットして、このポートをサスペンドモードにします。コアは、このビットがセットされるとSOFの送信のみを停止します。PHYクロックを停止するには、アプリケーションはPHYのサスペンド入力ピンをアサートするポートクロック停止ビットをセットする必要があります。

このビットの読出し値は、ポートの現在のサスペンドステータスを反映しています。このビットは、リモートウェイクアップ信号の検出後、またはアプリケーションがこのレジスタのポートリセットビットまたはポートレジュームビット、またはコア割込みレジスタのレジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（それぞれOTG_GINTSTSのWKUPINTまたはDISCINTビット）をセットしたときに、コアによってクリアされます。

0：ポートはサスペンドモードではありません。

1：ポートはサスペンドモードです。

ビット6 **PRES**：ポートのレジューム

アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。コアは、アプリケーションがこのビットをクリアするまで、レジューム信号の駆動を続行します。

コアが、コア割込みレジスタのポートレジューム／リモートウェイクアップ検出割込みビット（OTG_GINTSTSのWKUPINTビット）によって示されるUSBリモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、切断条件を検出したときにこのビットをクリアします。このビットの読出し値は、コアが現在レジューム信号を駆動中であるかどうかを示します。

0：レジュームシグナリングを駆動しません。

1：レジュームシグナリングを駆動します。

LPMが有効で、コアがL1状態にあるとき、このビットは次のように動作します。

1。アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。

2。コアはOTG_GLPMCFGレジスタのBESLTHRS[3:0] フィールドに指定された所定時間まで、レジューム信号の駆動を続行します。

3。コアが、コア割込みレジスタのポートL1レジューム／リモートL1ウェイクアップ検出割込みビット（OTG_GINTSTSのWKUPINTビット）によって示されるUSBリモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、レジュームの最後にこのビットをクリアします。このビットは、コアとアプリケーション両方でセット／クリアできます。このビットは、ホストにデバイスが1つも接続されていない場合でもコアによってクリアされます。

ビット5 **POCCHNG**：ポート過電流の変化

コアは、このレジスタのポート過電流アクティブビット（ビット4）に変化があったとき、このビットをセットします。

ビット4 **POCA**：ポート過電流アクティブ

ポートの過電流条件を示します。

0：過電流条件がありません。

1：過電流条件があります。

ビット3 **PENCHNG**：ポートの有効／無効の変化

コアは、このレジスタのポートイネーブルビット2のステータスに変化があったとき、このビットをセットします。

ビット2 **PENA**：ポートの有効化

ポートは、リセットシーケンス後、コアによってのみ有効にされ、過電流条件、切断条件、またはアプリケーションがこのビットをクリアすることによって無効にされます。アプリケーションは、レジスタへの書込みによってこのビットをクリアすることはできません。クリアして、ポートを無効にすることだけができます。このビットは、アプリケーションへの割込みをトリガしません。

0：ポートは無効です。

1：ポートは有効です。

ビット1 **PCDET**：ポート接続の検出

コアは、デバイス接続が検出されると、このビットをセットして、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTSのHPRTINTビット）を使用して、アプリケーションへの割込みをトリガします。アプリケーションは、このビットに1を書き込むことによって、割込みをクリアする必要があります。

ビット0 **PCSTS**：ポート接続ステータス

0：ポートにデバイスが接続されていません。

1：ポートにデバイスが接続されています。

73.14.31 OTGホストチャネルx特性レジスタ（OTG_HCCHARx）

アドレスオフセット：0x500 + 0x20 * x（x = 0から15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHENA	CHDIS	ODD FRM	DAD[6:0]							MCNT[1:0]		EPTYP[1:0]		LSDEV	Res.
rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPDIR	EPNUM[3:0]				MPSIZ[10:0]										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **CHENA**：チャネルの有効化

このフィールドは、アプリケーションによってセットされOTGホストによってクリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

ビット30 **CHDIS**：チャネルの無効化

アプリケーションは、チャネルへの転送が完了する前でも、このビットをセットして、そのチャネル上での送信／受信を停止することができます。アプリケーションは、チャネルを無効として扱う前に、チャネル無効割込みを待つ必要があります。

ビット29 **ODDFRM**：奇数フレーム

このフィールドはOTGホストが奇数フレームで転送を実行しなければならないことを示すために、アプリケーションによってセット（リセット）されます。このフィールドは、周期的（アイソクロナスおよびインタラプト）トランザクションに対してのみ適用されます。

0：偶数フレーム

1：奇数フレーム

ビット28:22 **DAD[6:0]**：デバイスアドレス

このフィールドは、データソースまたはデータシンクとして機能する特定のデバイスを選択します。

ビット21:20 **MCNT[1:0]**：マルチカウント

このフィールドは、この周期的エンドポイントについてフレームあたりで実行されなければならないトランザクションの数をホストに示します。非周期的転送の場合、このフィールドは使用されません。

00：予約済み。このフィールドは定義されていない結果をもたらします。

01：1つのトランザクション

10：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり2つです。

11：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり3つです。

注： このフィールドは、01以上にセットする必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

選択された転送タイプを示します。

00：コントロール

01：アイソクロナス

10：バルク

11：インタラプト

ビット17 **LSDEV**：ロースピードデバイス

このフィールドは、このチャネルがロースピードデバイスと通信中であることを示すために、アプリケーションによってセットされます。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **EPDIR**：エンドポイントの方向

トランザクションがINかOUTかを示します。

0：OUT

1：IN

ビット14:11 **EPNUM[3:0]**：エンドポイント番号

データソースまたはデータシンクとして機能しているデバイスのエンドポイント番号を示します。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

関連するエンドポイントの最大パケットサイズを示します。

73.14.32 OTGホストチャネルxスプリット制御レジスタ（OTG_HCSPLTx）

アドレスオフセット：0x504 + 0x20 * x（x = 0から15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT
rw															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XACTPOS[1:0]		HUBADDR[6:0]							PRTADDR[6:0]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **SPLITEN**：スプリットの有効化

アプリケーションはこのビットをセットして、このチャンネルがスプリットトランザクション処理用に有効化されていることを表示します。

ビット30:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **COMPLSPLT**：完全なスプリットを実行

アプリケーションはこのビットをセットしてOTGホストによる完全なスプリットトランザクション実行を要求します。

ビット15:14 **XACTPOS[1:0]**：トランザクションの配置

このフィールドを使って、各OUTトランザクションと一緒に全て、先頭、中央、または最終のいずれのペイロードを送信するか否かを決めます。

11：全て。このトランザクションの全データペイロード （188バイト以下）

10：先頭。このトランザクションの先頭データペイロード（188バイト超）

00：中央。このトランザクションの中央のペイロード （188バイト超）

01：最終。このトランザクションの最終ペイロード （188バイト超）

ビット13:7 **HUBADDR[6:0]**：ハブのアドレス

このフィールドは、トランザクショントランスレータのハブのデバイスアドレスを保持します。

ビット6:0 **PRTADDR[6:0]**：ポートアドレス

このフィールドは、受信トランザクショントランスレータのポート番号です。

73.14.33 OTGホストチャネルx割込みレジスタ（OTG_HCINTx）

このレジスタはUSBおよびAHB関連イベントに関するチャネルのステータスを示します。図917に示されています。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTSのHCINTビット）がセットされたとき、このレジスタを読み出す必要があります。アプリケーションがこれらのレジスタを読み出すためには、まず、ホスト全チャネル割込みレジスタ（OTG_HAINT）を読み出して、ホストチャネルx割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

アドレスオフセット：0x508 + 0x20 * x（x = 0から15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **DTERR**：データトグルエラー

ビット9 **FRMOR**：フレームのオーバーラン

ビット8 **BBERR**：バブルエラー

ビット7 **TXERR**：トランザクションエラー

USB上で以下のエラーの1つが発生したことを示します。
CRCチェック失敗
タイムアウト
ビットスタッフエラー
偽のEOP

ビット6 **NYET**：応答受信準備未完了割込み

ビット5 **ACK**：ACK応答受信／送信割込み

ビット4 **NAK**：NAK応答受信割込み

ビット3 **STALL**：STALL応答受信割込み

ビット2 **AHBERR**：AHBエラー

このエラーは、AHB読出し／書込み動作中にAHBエラーが発生したときに内部DMAモードでのみ生成されます。
読出し／書込み操作アプリケーションは対応するDMAチャネルアドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット1 **CHH**：チャネルの停止

USBトランザクションエラーまたはアプリケーションによる無効化リクエストへの応答のいずれかの理由により、転送が異常終了したことを示します。

ビット0 **XFRC**：転送の完了

転送がエラーなく正常に完了しました。

73.14.34 OTGホストチャネルx割込みマスクレジスタ（OTG_HCINTMSKx）

このレジスタは、前のセクションで説明した各チャネルステータスのマスクを反映します。

アドレスオフセット：0x50C + 0x20 * x、（x = 0 ~ 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	AHBERRM	CHHM	XFRCM
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **DTERRM**：データトグルエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット9 **FRMORM**：フレームオーバーランマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット8 **BBERRM**：バブルエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット7 **TXERRM**：トランザクションエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット6 **NYET**：応答受信割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット5 **ACKM**：ACK応答受信／送信割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット4 **NAKM**：NAK応答受信割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット3 **STALLM**：STALL応答受信割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット2 **AHBERRM**：AHBエラー
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット1 **CHHM**：チャネル停止マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

73.14.35 OTGホストチャネルx転送サイズレジスタ（OTG_HCTSIZx）

アドレスオフセット：0x510 + 0x20 * x（x = 0から15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO PNG	DPID[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **DOPNG**：PING実行

このビット 転送のみに使用されます。このフィールドに1を設定すると、ホストにPINGプロトコルを実行させます。

注： IN転送ではこのビットをセットしないでください。このビットがIN転送に対してセットされると、そのチャネルが無効化されます。

ビット30:29 **DPID[1:0]**：データPID

アプリケーションは、このフィールドを最初のトランザクションで使用するPIDのタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：DATA0

01：DATA2

10：DATA1

11：SETUP（コントロール）／MDATA（非コントロール）

ビット28:19 **PKTCNT [9:0]**：パケットカウント

このフィールドは、アプリケーションによって、送信（OUT）または受信（IN）されるパケットの予想数でプログラムされます。

ホストはOUT/INパケットの送信または受信が成功するたびに、このカウントをデクリメントします。このカウントがゼロに達すると、アプリケーションに割り込んで、正常完了を示します。

ビット18:0 **XFRSIZ[18:0]**：転送サイズ

OUTの場合、このフィールドは、ホストが転送時に送信するデータバイト数です。

INの場合、このフィールドは、アプリケーションが転送のために確保したバッファサイズです。アプリケーションは、INトランザクション（周期的および非周期的）の場合、このフィールドを最大パケットサイズの整数倍でプログラムすることが期待されます。

73.14.36 OTGホストチャネルx DMAアドレスレジスタ（OTG_HCDMAx）

アドレスオフセット：0x514 + 0x20 * x（x = 0から15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAADDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAADDR[31:0]**：DMAアドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる、または格納される開始アドレスを保持します。このレジスタは、各AHBトランザクションごとにインクリメントされます。

73.14.37 デバイスモードレジスタ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

73.14.38 OTGデバイス設定レジスタ（OTG_DCFG）

このレジスタは、電源投入後、または特定の制御コマンドあるいは列挙の後、コアをデバイスモードに設定します。初期プログラミングの後は、このレジスタを変更しないでください。

アドレスオフセット：0x800

リセット値：0x0220 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PERSCHIVL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRATIM	Res.	Res.	PFIVL[1:0]		DAD[6:0]							Res.	NZLSOHSK	DSPD[1:0]	
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット31:26 予約済みであり、リセット値に保持する必要があります。

ビット25:24 **PERSCHIVL[1:0]**：周期的スケジュールインターバル

このフィールドは、周期的INエンドポイントデータのフェッチ用に内部DMAエンジンが割り当てる必要のある時間を指定します。周期的エンドポイントの数に基づいて、この値を（マイクロ）フレームの25、50、75 % のいずれかに指定する必要があります。

– アクティブな周期的エンドポイントが存在するとき、内部DMAエンジンは、フェッチ中の周期的INエンドポイントデータに所定の時間を割り当てます。
– アクティブな周期的エンドポイントが存在しない場合、内部DMAエンジンは非周期的エンドポイントを処理し、このフィールドを無視します。
– （マイクロ）フレーム内で所定時間経過後 DMA は非周期的エンドポイントのフェッチへ切り替わります。

00：（マイクロ）フレームの25%
01：（マイクロ）フレームの50%
10：（マイクロ）フレームの75%
11：予約済み

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **ERRATIM**：不規則エラー割込みマスク

1：不規則エラー時の早期サスペンド割込みをマスクします。
0：不規則エラー時に早期サスペンド割込みが生成されます。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12:11 **PFIVL[1:0]**：周期的フレームインターバル

周期的フレーム終了割込みを使用してアプリケーションに通知しなければならないフレーム内の時間を示します。これを使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

00：フレームインターバルの80%
01：フレームインターバルの85%
10：フレームインターバルの90%
11：フレームインターバルの95%

ビット10:4 **DAD[6:0]**：デバイスアドレス

アプリケーションは、各SetAddressコントロールコマンドの後、このフィールドをプログラムする必要があります。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **NZLSOHSK**：非ゼロ長ステータスOUTハンドシェイク

アプリケーションは、このフィールドを使用して、コントロール転送のステータスステージのOUTトランザクション中に、コアが非ゼロ長のデータパケットを受信したときに送信するハンドシェイクを選択できます。

1：非ゼロ長ステータスのOUTトランザクション時にはSTALLハンドシェイクを送信し、受信したOUTパケットをアプリケーションに送信しません。
0：受信したOUTパケット（ゼロ長または非ゼロ長）をアプリケーションに送信し、デバイスエンドポイント制御レジスタのNAKおよびSTALLビットに基づいてハンドシェイクを送信します。

ビット1:0 **DSPD[1:0]**：デバイスのスピード

アプリケーションがコアにエニュメレーションを要求するスピード、またはアプリケーションがサポートできる最高スピードを示します。ただし、実際のバススピードは、チャープシーケンスの完了後に決まり、コアが接続されるUSBホストのスピードに基づきます。

00：ハイスピード
01：フルスピード
10：予約済み
11：予約済み

73.14.39 OTGデバイス制御レジスタ（OTG_DCTL）

アドレスオフセット：0x804

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.
													rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL[2:0]			GONSTS	GINSTS	SDIS	RWUSIG
				rw	w	w	w	w	rw	rw	rw	r	r	rw	rw

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18 **DSBESLRJCT**：ディープスリープBESL拒否

コアは、プログラムされたBESL閾値より大きいBESL値を持つLPMリクエストを拒否します。NYET応答は、BESL閾値より大きいBESL値を持つLPMトークンに送信されます。デフォルトでは、ディープスリープBESL拒否機能は無効です。

ビット 17:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **POPRGDNE**：パワーオンプログラミング終了

アプリケーションは、このビットを使用して、パワーダウンモードからのウェイクアップ後に、レジスタのプログラミングが終了したことを示します。

ビット10 **CGONAK**：グローバルOUT NAKのクリア

このフィールドに1を書き込むと、グローバルOUT NAKがクリアされます。

ビット9 **SGONAK**：グローバルOUT NAKのセット

このフィールドに1を書き込むと、グローバルOUT NAKがセットされます。
アプリケーションは、このビットを使用して、すべてのOUTエンドポイントにNAKハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割込みレジスタのグローバルOUT NAK有効ビット（OTG_GINTSTSのGONAKEFFビット）がクリアされていることを確認した後でなければなりません。

ビット8 **CGINAK**：グローバルIN NAKのクリア

このフィールドに1を書き込むと、グローバルIN NAKがクリアされます。

ビット7 **SGINAK**：グローバルIN NAKのセット

このフィールドに1を書き込むと、グローバル非周期的IN NAKがセットされます。アプリケーションは、このビットを使用して、すべての非周期的INエンドポイントにNAKハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割込みレジスタのグローバルIN NAK有効ビット（OTG_GINTSTSのGINAKEFFビット）がクリアされていることを確認した後でなければなりません。

ビット6:4 **TCTL[2:0]**：テスト制御

000：テストモードは無効
001：Test_Jモード
010：Test_Kモード
011：Test_SE0_NAKモード
100：Test_Packetモード
101：Test_Force_Enable
その他：予約済み

ビット3 **GONSTS**：グローバルOUT NAKのステータス

0：ハンドシェイクは、FIFOステータスとNAKおよびSTALLビットの設定に基づいて送信されます。

1：使用可能なスペースに関係なくRx FIFOにはデータが書き込まれません。SETUPトランザクションを除き、すべてのパケットにNAKハンドシェイクを送信します。すべてのアイソクロナスOUTパケットがドロップされます。

ビット2 **GINSTS**：グローバルIN NAKのステータス

0：ハンドシェイクは、送信FIFO内の使用可能なデータに基づいて送信されます。

1：NAKハンドシェイクは、送信FIFO内の使用可能なデータに関係なく、非周期的INエンドポイントで送出されます。

ビット1 **SDIS**：ソフト切断

アプリケーションは、このビットを使用してUSB OTGコアにソフト切断を実行することを伝えます。このビットがセットされている限り、ホストではデバイスが接続されているとはみなされず、デバイスはUSB上で信号を受信しません。コアは、アプリケーションでこのビットがクリアされるまで、切断状態にとどまります。

0：通常動作。ソフト切断後にこのビットがクリアされると、コアはUSBホストに対してデバイス接続イベントを生成します。デバイスが再接続されると、USBホストはデバイスのエニュメレーションをリスタートします。

1：コアはUSBホストに対して、デバイス切断イベントを生成します。

ビット0 **RWUSIG**：リモートウェイクアップシグナリング

アプリケーションがこのビットをセットすると、コアはリモートシグナリングを実行してUSBホストをウェイクアップします。コアをサスペンド状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。USB 2.0仕様で規定されているように、アプリケーションは、このビットをセットしてから1 msから15 ms後にクリアする必要があります。

LPMが有効で、コアがL1（スリープ）状態にあるとき、アプリケーションがこのビットをセットすると、コアはL1リモートシグナリングを開始して、USBホストをウエイクアップします。コアをスリープ状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。LPM仕様で規定されているように、このビットはアプリケーションによってセットされてから50µs（$T_{L1DevDrvResume}$）後にハードウェアによって自動的クリアされます。アプリケーションは、前のLPMトランザクションからのbRemoteWakeがゼロのときこのビットをセットする必要があります（GLPMCFGレジスタのREMWAKEビットを参照）。

表763はUSBホストがデバイスの切断を検出するためにソフト切断（SDIS）ビットがセットされなければならない最短時間（デバイスの状態による）を示します。クロックジッタに対応するには、アプリケーションは指定された最短時間まで遅延を追加することが推奨されます。

表763.ソフト切断のための最短時間

動作速度	デバイスの状態	最短時間
フルスピード	サスペンド状態	1ms + 2.5µs
フルスピード	アイドル	2.5µs
フルスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	2.5µs
ハイスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	125µs

73.14.40 OTGデバイスステータスレジスタ（OTG_DSTS）

このレジスタはUSB関連イベントに関するコアのステータスを示します。デバイス全割込みレジスタ（OTG_DAINT）からの割込み時に読み出されなければなりません。

アドレスオフセット：0x808

リセット値：0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS[1:0]		FNSOF[13:8]					
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FNSOF[7:0]								Res.	Res.	Res.	Res.	EERR	ENUMSPD[1:0]		SUSP STS
r	r	r	r	r	r	r	r					r	r	r	r

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23:22 **DEVLNSTS[1:0]**：デバイスラインステータス

USBデータラインの現在のロジックレベルを示します。

ビット [23]：D+ のロジックレベル

ビット [22]：D- のロジックレベル

ビット21:8 **FNSOF[13:0]**：受信SOFのフレーム番号

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **EERR**：不規則エラー

コアは、このビットをセットして、不規則エラーを報告します。

不規則エラーが発生すると、OTG_HS コントローラはサスペンド状態に移行し、OTG_GINTSTSレジスタのアーリーサスペンドビット（OTG_GINTSTSのESUSPビット）でアプリケーションへの割込みが生成されます。不規則エラーによるアーリーサスペンドがアサートされた場合、アプリケーションは、ソフト切断回復を実行するしかありません。

ビット2:1 **ENUMSPD[1:0]**：エニュメレーションされた速度

チャープシーケンスによるスピード検出後の OTG_HS コントローラのスピードを示します。

00：ハイスピード

01：フルスピード

11：予約済み

その他：予約済み

ビット0 **SUSPSTS**：サスペンドステータス

デバイスモードでは、USB上でサスペンド状態が検出されている限り、このビットがセットされます。USBデータライン上で3msの間アクティビティが1つもなかった場合、コアはサスペンド状態に入ります。コアは、以下の場合にサスペンド状態から抜け出します。

– USBデータライン上で何らかのアクティビティがあったとき。

– アプリケーションがOTG_DCTLレジスタのリモートウェイクアップシグナリングビット（OTG_DCTLのRWUSIGビット）に書込みを行ったとき。

73.14.41 OTGデバイスINエンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）

このレジスタは、すべてのエンドポイントの各OTG_DIEPINTxレジスタとともに機能して、INエンドポイントごとに1つの割込みを生成します。このレジスタの対応するビットに書き込むことによってOTG_DIEPINTxレジスタの特定のステータスに対するINエンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

アドレスオフセット：0x810

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	Res.	TXFURM	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
		rw					rw		rw	rw	rw	rw	rw	rw	rw

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAKM**：NAK割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット12:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **TXFURM**：FIFOアンダーランマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **INEPNEM**：INエンドポイントNAK有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット5 **INEPNMM**：EP不一致付きで受信されたINトークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット4 **ITTXFEMSK**：Tx FIFOエンプティ時に受信されたINトークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット2 **AHBERRM**：AHBエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット1 **EPDM**：エンドポイント無効割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

73.14.42 OTGデバイスOUTエンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）

このレジスタは、すべてのエンドポイントの各OTG_DOEPINTxレジスタとともに機能して、OUTエンドポイントごとに1つの割込みを生成します。このレジスタの対応するビットに書き込むことによってOTG_DOEPINTxレジスタの特定のステータスに対するOUTエンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

アドレスオフセット：0x814

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NYET MSK	NAK MSK	BERR M	Res.	Res.	Res.	OUT PKT ERRM	Res.	B2B STUPM	STS PHSR XM	OTEPD M	STUPM	AHB ERRM	EPDM	XFRC M
	rw	rw	rw				rw		rw	rw	rw	rw	rw	rw	rw

ビット31:15 予約済みであり、リセット値に保持する必要があります。

ビット14 **NYETMSK**：NYET割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット13 **NAKMSK**：NAK割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット12 **BERRM**：バブルエラー割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **OUTPKTERRM**：OUTパケットエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **B2BSTUPM**：連続SETUPパケット受信マスク
コントロールOUTエンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット5 **STSPHSRXM**：コントロール書込みに対するステータスフェーズ受信マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット4 **OTEPDM**：エンドポイント無効時OUT トークン受信マスクコントロールOUTエンドポイントにのみ適用されます。

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット3 **STUPM**：STUPM：SETUPフェーズ終了マスクコントロールエンドポイントにのみ適用されます。

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット2 **AHBERRM**：AHBエラーマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット1 **EPDM**：エンドポイント無効割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット0 **XFRCM**：転送完了割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

73.14.43 OTGデバイス全エンドポイント割込みレジスタ（OTG_DAINT）

エンドポイント上で重要なイベントが発生すると、OTG_DAINTレジスタは、OTG_GINTSTSレジスタのデバイスOUTエンドポイント割込みビットまたはデバイスINエンドポイント割込みビット（それぞれ、OTG_GINTSTSのOEPINTまたはIEPINTビット）を使用して、アプリケーションに割込みをかけます。エンドポイントごとに1つの割込みビットがあります（OUTエンドポイントおよびINエンドポイントに対してそれぞれ最大16bit）。双方向エンドポイントの場合、対応するINおよびOUT割込みビットが使用されます。このレジスタのビットは、アプリケーションが対応するデバイスエンドポイント-x割込みレジスタのビット（OTG_DIEPINTx/OTG_DOEPINTx）をセット／クリアしたときに、セット／クリアされます。

アドレスオフセット：0x818

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEPINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IEPINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **OEPINT[15:0]**：OUTエンドポイント割込みビット

OUTエンドポイントチャネルあたり 1 ビット：
OUTエンドポイント0に対してビット16、OUTエンドポイント3に対してビット19

ビット15:0 **IEPINT[15:0]**：INエンドポイント割込みビット

INエンドポイントあたり 1 ビット：
INエンドポイント0に対してビット0、INエンドポイント3に対してビット3

73.14.44 OTG全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）

OTG_DAINTMSKレジスタはデバイスエンドポイント割込みレジスタと連動し、デバイスエンドポイントでイベントが発生したときにアプリケーションに割込みをかけます。ただし、その割込みに対応するOTG_DAINTレジスタのビットはセットされたままです。

アドレスオフセット：0x81C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OEPM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IEPM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 **OEPM[15:0]：**OUT EP割込みマスクビット

OUTエンドポイント（EP）あたり1bit：
OUT EP 0に対してビット16、OUT EP 3に対してビット19
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット15:0 **IEPM[15:0]：**IN EP割込みマスクビット

INエンドポイントあたり 1 ビット：
IN EP 0に対してビット0、IN EP 3に対してビット3
0：割込みはマスクされます。
1：割込みはマスクされません。

73.14.45 OTGデバイス閾値制御レジスタ（OTG_DTHRCTL）

アドレスオフセット：0x0830

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	ARPEN:	Res.	RXTHRLEN[8:0]									RXTHREN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXTHRLEN[8:0]									ISOTHREN	NONISOTHREN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **ARPEN：**アービタパーキングの有効化

このビットは、INエンドポイントに対する内部DMAアービタパーキングを制御します。閾値が有効化され、かつこのビットに1が設定されると、アービタはUSBでトークンが受信されたINエンドポイントに待機します。これは、アンダーラン状態を回避するために実行されます。デフォルトではパーキングは有効化されています。

ビット26 予約済みであり、リセット値に保持する必要があります。

ビット25:17 **RXTHRLEN[8:0]**：受信閾値長

このフィールドは、受信閾値サイズを32-bitワード単位で指定します。また、このフィールドはコアがAHBで送信を開始できる前にUSBで受信するデータ量も指定します。閾値長は、少なくとも8個の32-bit ワードである必要があります。RXTHRLEN の推奨値は、設定された AHB バースト長（OTG_GAHBCFGのHBSTLENビット）と同じです。

ビット16 **RXTHREN**：受信閾値の有効化

このビットをセットすると、コアは受信方向で閾値を有効化します。

ビット15:11 予約済みであり、リセット値に保持する必要があります。

ビット10:2 **TXTHRLEN[8:0]**：送信閾値長

このフィールドは、送信閾値サイズを32-bitワード単位で指定します。このフィールドでは、コアがUSBで送信を開始できるようになるまでに、対応するエンドポイント送信FIFOで受信しているべきデータ量をバイト単位で指定します。閾値長は、少なくとも8個の32-bitワードである必要があります。このフィールドは、アイソクロナスと非アイソクロナスのINエンドポイント閾値を制御します。TXTHRLENの推奨値は、設定されたAHBバースト長（OTG_GAHBCFGのHBSTLENビット）と同じです。

ビット1 **ISOTHREN**：ISO INエンドポイント閾値の有効化

このビットをセットすると、コアはアイソクロナスINエンドポイントで閾値を有効化します。

ビット0 **NONISOTHREN**：非アイソクロナスINエンドポイント閾値の有効化

このビットをセットすると、コアは非アイソクロナスINエンドポイントで閾値を有効化します。

73.14.46 OTGデバイスINエンドポイントFIFOエンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）

このレジスタは、INエンドポイントFIFOエンプティ割込みの生成（TXFE_OTG_DIEPINTx）を制御するために使用されます。

アドレスオフセット：0x834

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTXFEM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INEPTXFEM[15:0]**：IN EP Tx FIFOエンプティ割込みマスクビット

これらのビットはOTG_DIEPINTxに対するマスクビットとして機能します。

IN EPあたりTXFE割込み1bit：

IN EP 0に対してビット0、IN EP 3に対してビット3

0：割込みはマスクされます。

1：割込みはマスクされません。

73.14.47 OTGデバイスINエンドポイントx制御レジスタ[オルタネート]（OTG_DIEPCTLx）

INT/BULKエンドポイントに対して有効です。ISOエンドポイントについては、次のセクションを参照してください。

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

アドレスオフセット：0x900 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID	SD0 PID	SNAK	CNAK	TXFNUM[3:0]				STALL	Res.	EPTYP[1:0]		NAK STS	DPID
rs	rs	w	w	w	w	rw	rw	rw	rw	rw		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUPフェーズ終了
– エンドポイント無効
– 転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SD1PID**：DATA1 PIDの設定

このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA1に設定されます。

ビット28 **SD0PID**：DATA0 PIDの設定

インタラプト／バルクINエンドポイントにのみ適用されます。
このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA0に設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 **TXFNUM[3:0]**：Tx FIFO番号

これらのビットは、このエンドポイントに関連するFIFO番号を指定します。各有効INエンドポイントは、別々のFIFO番号にプログラムされなければなりません。

このフィールドは、INエンドポイントについてのみ有効です。

ビット21 **STALL**：STALLハンドシェイク

非コントロール、非アイソクロナスINエンドポイントにのみ適用されます（アクセスタイプはrwです）。

アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：インタラプト

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりデータの送信が停止されます。

アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりゼロ長のデータパケットが送出されます。

このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **DPID**：エンドポイントデータPID

インタラプト／バルクINエンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットのPIDを含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットのPIDをプログラムする必要があります。アプリケーションはSC0PIDおよびSD1PIDレジスタフィールドを使用して、DATA0またはDATA1 PIDのいずれかをプログラムします。

0：DATA0

1：DATA1

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

73.14.48 OTGデバイスINエンドポイントx制御レジスタ[オルタネート]（OTG_DIEPCTLx）

ISOエンドポイントに対して有効です。INT/BULKエンドポイントについては、前のセクションを参照してください。

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

アドレスオフセット：0x900 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SEVN FRM	SNAK	CNAK	TXFNUM[3:0]				STALL	Res.	EPTYP[1:0]		NAK STS	EO NUM
rs	rs	w	w	w	w	rw	rw	rw	rw	rw		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUPフェーズ終了
– エンドポイント無効
– 転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SODDFRM**：奇数フレームの設定

アイソクロナスINおよびOUTエンドポイントにのみ適用されます。
このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット28 **SEVNFRM**：偶数フレームの設定

アイソクロナスINエンドポイントにのみ適用されます。
このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 **TXFNUM[3:0]**：Tx FIFO番号

これらのビットは、このエンドポイントに関連するFIFO番号を指定します。各有効INエンドポイントは、別々のFIFO番号にプログラムされなければなりません。

このフィールドは、INエンドポイントについてのみ有効です。

ビット21 **STALL**：STALLハンドシェイク

非コントロール、非アイソクロナスINエンドポイントにのみ適用されます（アクセスタイプはrwです）。

アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

ビット20 予約済みであり、リセット値に保持する必要があります。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：インタラプト

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりデータの送信が停止されます。

アイソクロナスINエンドポイントの場合：Tx FIFOに使用可能なデータがある場合でも、コアによりゼロ長のデータパケットが送出されます。

このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **EONUM**：偶数／奇数フレーム

アイソクロナスINエンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタのSEVNFRMおよびSODDFRMフィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

73.14.49 OTGデバイスINエンドポイントx割込みレジスタ（OTG_DIEPINTx）

このレジスタは、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。図917に示されています。コア割込みレジスタ（OTG_GINTSTSのIEPINT）のINエ ンドポイント割込みビットがセットされている場合、アプリケーションはこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出す前に、最初にデバイス･オール･エンドポイント割込み（OTG_DAINT）レジスタを読み込んで、デバイス･エンドポイント-x割込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

アドレスオフセット：0x908 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
		rc_w1		rc_w1			rc_w1	r	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **NAK**：NAK入力

コアは、デバイスがNAKを送受信したとき、この割込みを生成します。アイソクロナスINエンドポイントの場合、Tx FIFOにデータがないためにゼロ長パケットが送信されたとき、この割込みが生成されます。

ビット12 予約済みであり、リセット値に保持する必要があります。

ビット11 **PKTDRPSTS**：パケットドロップステータス

このビットは、ISOC OUTパケットがドロップされたことをアプリケーションに示します。このビットには対応するマスクビットがなく、割込みは生成されません。

ビット10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **TXFIFOUDRN**：送信FIFOアンダーラン（TxfifoUndrn）

コアはこのエンドポイントの送信FIFOアンダーラン状態を検出するとこの割込みを生成します。依存性：この割込みは、閾値が有効化されている場合にのみ有効です。

ビット7 **TXFE**：送信FIFOエンプティ

この割込みは、このエンドポイントのTx FIFOが半エンプティまたは完全エンプティのいずれかであるとき、アサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFGレジスタのTx FIFOエンプティレベルビット（OTG_GAHBCFGのTXFELVLビット）によって判断されます。

ビット6 **INEPNE**：INエンドポイントNAKの有効性

このビットは、アプリケーションがOTG_DIEPCTLxレジスタのCNAKビットに書き込むことによってINエンドポイントNAKをクリアするとクリアできます。

この割込みは、コアが（アプリケーションまたはコアによって）設定されたNAKビットをサンプリングしたことを示します。この割込みは、アプリケーションによってセットされたINエンドポイントNAKビットがコアで有効になったことを示します。

この割込みは、NAKハンドシェイクがUSBで送出されることを保証するものではありません。STALLビットがNAKビットに優先します。

ビット5 **INEPNM**：EP不一致で受信されたINトークン

非周期的TxFIFOの最上位のデータが、INトークンが受信されたエンドポイント以外のエンドポイントに属していることを示します。この割込みは、INトークンが受信されたエンドポイント上でアサートされます。

ビット4 **ITTXFE**：Tx FIFOがエンプティ時にINトークンを受信

関連するTx FIFO（周期的/非周期的）が空のときにINトークンが受信されたことを示します。この割込みは、INトークンが受信されたエンドポイント上でアサートされます。

ビット3 **TOC**：タイムアウト条件

このエンドポイント上の最後のINトークンに対して、コアがUSB上でタイムアウト条件を検出したことを示します。

ビット2 **AHBERR**：AHBエラー

このエラーはAHB読出し／書込み動作中にAHBエラーが発生したとき、内部DMAモードでのみ生成されます。アプリケーションは対応する エンドポイントDMAアドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット1 **EPDISD**：エンドポイント無効割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送がAHBに加えてUSB上でも完了したことを示します。

73.14.50 OTGデバイスINエンドポイント0転送サイズレジスタ（OTG_DIEPTSIZ0）

アプリケーションは、エンドポイント0を有効にする前に、このレジスタを変更する必要があります。デバイスコントロールエンドポイント 0 制御レジスタのエンドポイントイネーブルビット（OTG_DIEPCTL0のEPENA）を使用して、エンドポイント0が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント1～3のレジスタを使用します。

アドレスオフセット：0x910

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT[1:0]		Res.	Res.	Res.
											rw	rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20:19 **PKTCNT [1:0]**：パケットカウント

エンドポイント0のデータ転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がTxFIFOから読み出されるたびにデクリメントされます。

ビット18:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント0の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットがTxFIFOに書き込まれるたびに、このフィールドをデクリメントします。

73.14.51 OTGデバイスINエンドポイントx DMAアドレスレジスタ（OTG_DIEPDMAx）

アドレスオフセット：0x914 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAADDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAADDR[31:0]**：DMAアドレス

このフィールドは､外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各AHBトランザクションごとにインクリメントされます。

73.14.52 OTGデバイスINエンドポイント送信FIFOステータスレジスタ（OTG_DTXFSTSx）

この読出し専用レジスタは、デバイスINエンドポイントTxFIFOの空きスペース情報を含みます。

アドレスオフセット：0x918 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTFSAV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **INEPTFSAV[15:0]**：INエンドポイントTx FIFOの使用可能スペース

エンドポイントのTx FIFOで使用できる空きスペースの量を示します。

値は32bitワード単位です。

0x0：エンドポイントTx FIFOはフルです。

0x1：1ワードが使用可能です。

0X2：2ワードが使用可能です。

0xn：nワードが使用可能です。

その他：予約済み

73.14.53 OTGデバイスINエンドポイントx転送サイズレジスタ（OTG_DIEPTSIZx）

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DIEPCTLxレジスタのエンドポイントイネーブルビット（OTG_DIEPCTLxのEPENAビット）を使用してエンドポイントが有効になると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

アドレスオフセット：0x910 + 0x20 * x, (x = 1から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCNT[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **MCNT[1:0]**：マルチカウント

周期的INエンドポイントの場合、このフィールドは、USBでフレームごとに送信されなければならないパケット数を示します。コアは、このフィールドを使用して、アイソクロナスINエンドポイントのデータPIDを計算します。

01：1パケット

10：2パケット

11：3パケット

ビット28:19 **PKTCNT [9:0]**：パケットカウント

このエンドポイントのデータの転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がTxFIFOから読み出されるたびにデクリメントされます。

ビット18:0 **XFRSIZ[18:0]**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットがTxFIFOに書き込まれるたびに、このフィールドをデクリメントします。

73.14.54 OTGデバイスコントロールOUTエンドポイント0制御レジスタ（OTG_DOEPCTL0）

このセクションでは、OTG_DOEPCTL0 レジスタについて説明します。非ゼロコントロールエンドポイントは、エンドポイント1 ~ 8 のレジスタを使用します。

アドレスオフセット：0xB00

リセット値：0x0000 8000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	Res.
w	r			w	w					rs	rw	r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ[1:0]	
r														r	r

ビット31 **EPENA**：エンドポイントの有効化

アプリケーションは、このビットをセットして、エンドポイント0上でデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUPフェーズ終了
– エンドポイント無効
– 転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、コントロールOUTエンドポイント0を無効にできません。

ビット29:28 予約済みであり、リセット値に保持する必要があります。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、送信完了割込み時、またはエンドポイントでSETUPを受信した後に、このビットをセットすることもできます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **STALL**：STALLハンドシェイク

アプリケーションは、このビットのセットのみ可能です。このエンドポイントのSETUP トークンが受信されると、コアがこのビットをクリアします。NAKビット、すなわち、グローバルOUT NAKがこのビットとともにセットされた場合は、STALLビットが優先します。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアはOUTパケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

コントロール用に00にハードコード化されています。

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、Rx FIFOに受信パケットを収容できるスペースがある場合でも、コアはデータの受信を停止します。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15 **USBAEP**：USBアクティブエンドポイント

このビットは常に1にセットされ、コントロールエンドポイント0がすべての設定およびインタフェースにおいて常にアクティブであることを示します。

ビット14:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **MPSIZ[1:0]**：最大パケットサイズ

コントロールOUTエンドポイント0の最大パケットサイズは、コントロールINエンドポイント0でプログラムされるサイズと同じです。

00：64バイト

01：32バイト

10：16バイト

11：8バイト

73.14.55 OTGデバイスOUTエンドポイントx割込みレジスタ（OTG_DOEPINTx）

このレジスタは、USBおよびAHB関連イベントに関するエンドポイントのステータスを示します。図917に示されています。アプリケーションはOTG_GINTSTSレジスタのOUTエンドポイント割込みビット(OTG_GINTSTSのOEPINTビット)がセットされている場合にこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出すためには、まず、OTG_DAINTレジスタを読み出して、OTG_DOEPINTx レジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINTおよびOTG_GINTSTSレジスタの対応するビットをクリアする必要があります。

アドレスオフセット：0xB08 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STPK TRX	NYET	NAK	BERR	Res.	Res.	Res.	OUT PKT ERR	Res.	B2B STUP	STSPH SRX	OTEP DIS	STUP	AHB ERR	EP DISD	XFRC
rc_w1	rc_w1	rc_w1	rc_w1				rc_w1		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **STPKTRX**：セットアップパケット受信

バッファDMAモードでのみコントロールOUTエンドポイントに適用できます。OTG_HS でセットされた場合、このビットによってこのバッファが8バイトのセットアップデータを保持していることを示します。バッファごとにセットアップパケットは1つだけです。セットアップパケットを受信すると、OTG_HSはSETUP_COMPLETEステータスがRx FIFOで確認された後、バッファを閉じて、対応するエンドポイントを無効にします。OTG_HSは特定のエンドポイントのSETUPパケットの後の最初のINまたはOUTトークンを検出すると、SETUP_COMPLETEステータスをRx FIFOに格納します。その後、アプリケーションは、コントロール転送のためのOUTデータを受信するためにエンドポイントを再有効化し、バッファの開始アドレスを再プログラムする必要があります。上記の動作により、OTG_HS は連続セットアップパケットを受信でき、各セットアップパケットごとに1つのバッファが使用されます。

ビット14 **NYET**：NYET割込み

非アイソクロナスOUTエンドポイントに対してNYET応答が送信されたときこの割込みが生成されます。

ビット13 **NAK**：NAK入力

コアは、デバイスがNAKを送受信したとき、この割込みを生成します。アイソクロナスINエンドポイントの場合、Tx FIFOにデータがないためにゼロ長パケットが送信されたとき、
この割込みが生成されます。

ビット12 **BERR**：バブルエラー割込み

コアは、エンドポイントに対してバブルが受信されたとき、この割込みを生成します。

ビット11:10 予約済みであり、リセット値に保持する必要があります。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **OUTPKTERR**：OUTパケットエラー

この割込みは、コアがOUTパケットに対してオーバーフローまたはCRCエラーを検出したとき、アサートされます。この割込みは、閾値が有効化されている場合にのみ有効です。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **B2BSTUP**：連続SETUPパケットの受信

コントロールOUTエンドポイントにのみ適用されます。
このビットは、コアがこの特定のエンドポイントで3つを超える連続SETUPパケットを受信したことを示します。

ビット5 **STSPHSRX**：コントロール書込みに対して受信されたステータスフェーズ

この割込みは、コントロールOUTエンドポイントについてのみ有効です。ホストがコントロール書込みのデータフェーズの転送中に送信したすべてのデータを OTG_HS がシステムメモリバッファに転送した後にのみ、この割込みが生成されます。この割込みによって、ホストがコントロール書込み転送のデータフェーズからステータスフェーズに切り換えたことがアプリケーションに知らされます。アプリケーションはこの割込みを使って、データフェーズをデコードした後、ステータスフェーズにACKまたはSTALLを返すことができます。

ビット4 **OTEPDIS**：エンドポイント無効時OUTトークン受信

コントロールOUTエンドポイントにのみ適用されます。
エンドポイントがまだ有効になっていないときにOUTトークンを受け取ったことを示します。この割込みは、OUTトークンが受信されたエンドポイントでアサートされます。

ビット3 **STUP**：SETUPフェーズの完了

コントロールOUTエンドポイントにのみ適用されます。コントロールエンドポイントのSETUPフェーズが完了し、現在のコントロール転送のために、これ以上、連続SETUPパケットは受信されなかったことを示します。この割込みにより、アプリケーションは、受信したSETUPデータパケットをデコードできます。

ビット2 **AHBERR**：AHBエラー

このエラーはAHB読出し／書込み動作中にAHBエラーが発生したとき、内部DMAモードでのみ生成されます。アプリケーションは対応する エンドポイントDMAアドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット1 **EPDISD**：エンドポイント無効割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送がAHBに加えてUSB上でも完了したことを示します。

73.14.56 OTGデバイスOUTエンドポイント0転送サイズレジスタ（OTG_DOEPTSIZ0）

アプリケーションは、エンドポイント0を有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTL0レジスタのエンドポイントイネーブルビット（OTG_DOEPCTL0のEPENAビット）を使用してエンドポイント0が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント1～8 のレジスタを使用します。

アドレスオフセット：0xB10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	STUPCNT[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.
	rw	rw										rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **STUPCNT[1:0]**：SETUPパケットカウント

このフィールドは、エンドポイントが受信できる連続SETUPデータパケット数を指定します。

01：1パケット

10：2パケット

11：3パケット

ビット28:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **PKTCNT**：パケットカウント

このフィールドは、パケットがRx FIFOに書き込まれた後、ゼロまでデクリメントされます。

ビット18:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント0の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFOからパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

73.14.57 OTGデバイスOUTエンドポイントx DMAアドレスレジスタ（OTG_DOEPDMAx）

アドレスオフセット：0xB14 + 0x20 * x, (x = 0から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAADDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **DMAADDR[31:0]**：DMAアドレス

このフィールドは､外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各AHBトランザクションごとにインクリメントされます。

73.14.58 OTGデバイスOUTエンドポイントx制御レジスタ[オルタネート]（OTG_DOEPCTLx）

INT/BULKエンドポイントに対して有効です。ISOエンドポイントについては、次のセクションを参照してください。

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

アドレスオフセット：0xB00 + 0x20 * x, (x = 1から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID	SD0 PID	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	DPID
rs	rs	w	w	w	w					rw	rw	rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

INおよびOUTエンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUPフェーズ終了

– エンドポイント無効

– 転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SD1PID**：DATA1 PIDの設定

このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA1に設定されます。

ビット28 **SD0PID**：DATA0 PIDの設定

インタラプト／バルクOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータPID（DPID）フィールドがDATA0に設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **STALL**：STALLハンドシェイク

非コントロール、非アイソクロナスINエンドポイントにのみ適用されます（アクセスタイプはrwです）。

アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプはrsです）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントのSETUPトークンが受信されると、コアがこのビットをクリアします。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアはOUTパケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：インタラプト

ビット17 **NAKSTS**：NAKステータス

以下を示します。

0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。

1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

コアは、Rx FIFOに受信パケットを収容するためのスペースがある場合でも、OUTエンドポイントでのデータの受信を停止します。

このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **DPID**：エンドポイントデータPID

インタラプト／バルクOUTエンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットのPIDを含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットのPIDをプログラムする必要があります。アプリケーションはSC0PIDおよびSD1PIDレジスタフィールドを使用して、DATA0またはDATA1 PIDのいずれかをプログラムします。

0：DATA0

1：DATA1

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

73.14.59 OTGデバイスOUTエンドポイントx制御レジスタ[オルタネート]（OTG_DOEPCTLx）

ISOエンドポイントに対して有効です。INT/BULKエンドポイントについては、前のセクションを参照してください。

アプリケーションは、このレジスタを使用して、エンドポイント0以外の各論理エンドポイントの動作を制御します。

アドレスオフセット：0xB00 + 0x20 * x, (x = 1から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SEVN FRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	EO NUM
rs	rs	w	w	w	w					rw	rw	rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **EPENA**：エンドポイントの有効化

INおよびOUTエンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUPフェーズ終了

– エンドポイント無効

– 転送完了

ビット30 **EPDIS**：エンドポイントの無効化

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット29 **SODDFRM**：奇数フレームの設定

アイソクロナスINおよびOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット28 **SEVNFRM**：偶数フレームの設定

アイソクロナスOUTエンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット27 **SNAK**：NAKのセット

このビットに書き込むと、エンドポイントのNAKビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上でのNAKハンドシェイクの送信を制御できます。コアは、転送完了割込み時やエンドポイントでSETUPを受信した後のOUTエンドポイントにもこのビットを設定できます。

ビット26 **CNAK**：NAKのクリア

このビットに書き込むと、エンドポイントのNAKビットがクリアされます。

ビット25:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **STALL**：STALLハンドシェイク

非コントロール、非アイソクロナスINエンドポイントにのみ適用されます（アクセスタイプはrwです）。
アプリケーションは、このビットをセットして、USBホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。
制御エンドポイントにのみ適用されます（アクセスタイプはrsです）。
アプリケーションは、このビットのセットのみ可能です。このエンドポイントのSETUPトークンが受信されると、コアがこのビットをクリアします。このビットとともにNAKビット（グローバルIN NAKまたはグローバルOUT NAK）がセットされた場合は、STALLビットが優先されます。このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアはOUTパケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。
00：コントロール
01：アイソクロナス
10：バルク
11：インタラプト

ビット17 **NAKSTS**：NAKステータス

以下を示します。
0：コアは、FIFOのステータスに基づいて非NAKハンドシェイクを送信中です。
1：コアは、このエンドポイント上でNAKハンドシェイクを送信中です。
アプリケーションまたはコアがこのビットをセットすると、
コアは、Rx FIFOに受信パケットを収容するためのスペースがある場合でも、OUTエンドポイントでのデータの受信を停止します。
このビットの設定に関係なく、コアは、常にSETUPデータパケットにACKハンドシェイクで応答します。

ビット16 **EONUM**：偶数／奇数フレーム

アイソクロナスOUTエンドポイントにのみ適用されます。
コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタのSEVNFRMおよびSODDFRMフィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。
0：偶数フレーム
1：奇数フレーム

ビット15 **USBAEP**：USBアクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USBリセットを検出した後、すべてのエンドポイント（EP 0を除き）について、このビットをクリアします。SetConfigurationおよびSetInterfaceコマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット14:11 予約済みであり、リセット値に保持する必要があります。

ビット10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

73.14.60 OTGデバイスOUTエンドポイントx転送サイズレジスタ（OTG_DOEPTSIZx）

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTLxレジスタのエンドポイントイネーブルビット（OTG_DOEPCTLxのEPENAビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

アドレスオフセット：0xB10 + 0x20 * x, (x = 1から8)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RXDPID[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30:29 **RXDPID[1:0]：**

条件：アイソクロナスOUTエンドポイント

受信データPID

これは、このエンドポイントの最後のパケットで受信されたデータのPIDです。

00：DATA0

01：DATA2

10：DATA1

11：MDATA

条件：コントロールOUTエンドポイント

STUPCNT[1:0]：SETUPパケットカウント

このフィールドは、エンドポイントが受信できる連続SETUPデータパケット数を指定します。

01：1パケット

10：2パケット

11：3パケット

ビット28:19 **PKTCNT [9:0]：**パケットカウント

このエンドポイントのデータの転送サイズを構成するUSBパケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）がRx FIFOに書き込まれるたびにデクリメントされます。

ビット18:0 **XFRSIZ[18:0]：**転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFOからパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

73.14.61 OTGパワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

このレジスタは、ホストモードとデバイスモードで使用できます。

アドレスオフセット：0xE00

リセット値：0x200B 8000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHY SLEEP	ENL1 GTG	PHY SUSP	Res.	Res.	GATE HCLK	STPP CLK
								r	r	rw	r			rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **SUSP**：ディープスリープ

このビットは、L1状態時にPHYがディープスリープであることを示します。

ビット6 **PHYSLEEP**：PHYがスリープ状態

このビットは、PHYがスリープ状態であることを示します。

ビット5 **ENL1GTG**：スリープクロックゲーティングの有効化

このビットがセットされている場合、コアがutmi_l1_suspend_nをアサートできなければ、スリープ状態でコアの内部クロックゲーティングが有効化されます。このビットがセットされていない場合、PHYクロックはスリープ状態ではゲートされません。

ビット4 **PHYSUSP**：PHYのサスペンド

PHYがサスペンドされたことを示します。このビットは、アプリケーションがSTPPCLKビットをセットした後、PHYがサスペンドされると更新されます。

ビット3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **GATEHCLK**：HCLKのゲーティング

USBがサスペンドされるか、セッションが有効でないとき、アプリケーションは、このビットをセットして、AHBスレーブとマスタおよびウェイクアップロジック以外のモジュールへのHCLKをゲーティングします。アプリケーションは、USBがレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

ビット0 **STPPCLK**：PHYクロックを停止

USBがサスペンドされるか、セッションが有効でないか、デバイスが切断されると、アプリケーションは、このビットをセットして、PHYクロックを停止します。アプリケーションは、USBがレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

73.14.62 OTGパワーおよびクロックゲーティング制御レジスタ1（OTG_PCGCCTL1）

このレジスタは、ホストモードとデバイスモードで使用できます。

アドレスオフセット：0xE04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RAM_GATEEN	CNTGATECLK[1:0]		GATEEN
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **RAMGATEEN**：RAMクロックゲーティングの有効化

FIFO RAMのゲーティングを有効化します。

ビット2:1 **CNTGATECLK[1:0]**：クロックゲーティング用のカウンタ

コントローラ内部の各PHYおよびAHBクロックをゲーティングする前にコントローラが待ち合わせる、「IDLE」状態（アクティビティなし）のPHYクロックサイクルおよびAHBクロックサイクルの数をコントローラに通知します。

00：64クロック

01：128クロック

10：予約済み

11：予約済み

ビット0 **GATEEN**：アクティブなクロックゲーティングの有効化

アプリケーションによりGATEENをプログラムし、PHYおよびAHBクロックのアクティブクロックゲーティング機能を有効化します。

73.14.63 OTG_HS レジスタマップ

次の表に、USB OTGのレジスタマップとリセット値を示します。

表764.レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	OTG_ GOTGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CURMOD	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS	Res.	Res.	Res.	EHEN	Res.	Res.	Res.	Res.	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	Res.	Res.
	リセット値											0	0	0	0	0	1				0					0	0	0	0	0	0		
0x004	OTG_ GOTGINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADTOCHG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
	リセット値														0								-								0		
0x008	OTG_ GAHBCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	DMAEN	HBSTLEN				GINTMSK
	リセット値																								0	0		0	0	0	0	0	0
0x00C	OTG_ GUSBCFG	Res.	FDMOD	FHMOD	Res.	Res.	Res.	Res.	Res.	Res.	TSDPS	Res.	Res.	Res.	Res.	Res.	Res.	PHYLPC	Res.	TRDT				Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOCAL		
	リセット値		0	0														0		0	1	0	1								0	0	0
0x010	OTG_GRSTCTL	AHBIDL	DMAREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFNUM					TXFFLSH	RXFFLSH	Res.	FCRST	PSRST	CSRST
	リセット値	1	0																				0	0	0	0	0	0	0		0	0	0
0x014	OTG_GINTSTS （デバイスモード）	WKUPINT	SRQINT	DISCINT	CIDSCHG	LPMINT	PTXFE	HCINT	HPRTINT	RSTDET	DATAFSUSP	IPXFR	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	0	0	1	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0
	OTG_GINTSTS （ホストモード）	WKUPINT	SRQINT	DISCINT	CIDSCHG	LPMINT	PTXFE	HCINT	HPRTINT	RSTDET	DATAFSUSP	INCOMPISOOUT	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	0	0	1	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0
0x018	OTG_GINTMSK （デバイスモード）	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IPXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	
	OTG_GINTMSK （ホストモード）	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	OTG_GRXSTSR（デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	OTG_GRXSTSP（デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSP（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	OTG_GRXFSIZ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFD															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x028	OTG_HNPTXFSIZ（デバイスモード）	NPTXFD																NPTXFSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	OTG_DIEPTXF0（ホストモード）	TX0FD																TX0FSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x02C	OTG_HNPTXSTS	Res.	NPTXQTOP							NPTQXSAV								NPTXFSAV															
	リセット値		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x038	OTG_GCCFG	Res.	Res.	Res.	Res.	Res.	Res.	FORCEHOSTPD	VBVALOVEN	FORCEHOSTPD	SDEN	VBDEN	PDEN	DCDEN	HVDMSRCEN	HCDPDETEN	HCDPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SESSVLD	FSVMINUS	FSVPLUS	CHGDET	
	リセット値							0	0	0	0	0	0	0	0	0	0												X	X	X	X	
0x03C	**OTG_CID**	PRODUCT_ID																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0x054	OTG_GLPMCFG	Res.	Res.	Res.	ENBESL	LPMR CNTSTS			SNDLPM	LPM RCNT			LPMCHIDX				L1RSMOK	SLPSTS	LPM RSP		L1DSEN	BESLTHRS				L1SSEN	REMWAKE	BESL			LPMACK	LPMEN	
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	OTG_HPTXFSIZ	PTXFSIZ																PTXSA															
	リセット値	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0x104	OTG_DIEPTXF1	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x120	OTG_DIEPTXF7	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0x400	OTG_HCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLS PCS	
	リセット値																														0	0	0
0x404	OTG_HFIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLDCTRL	FRIVL															
	リセット値																0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0	0
0x408	OTG_HFNUM	FTREM																FRNUM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x410	OTG_HPTXSTS	PTXQTOP								PTXQSAV								PTXFSAVL															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x414	OTG_HAINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINT															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x418	OTG_ HAINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINTM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x440	OTG_HPRT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD		PTCTL				PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS
	リセット値														0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x500	OTG_ HCCHAR0	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x504	OTG_ HCSPLT0	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XAC TPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x508	OTG_HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x508	OTG_HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x50C	OTG_ HCINTMSK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x510	OTG_HCTSIZ0	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x514	OTG_HCDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	オフセット0x500から始まるレジスタの前のブロックを繰り返します																															
0x6E0	OTG_ HCCHAR15	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6E4	OTG_ HCSPLT15	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XAC TPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6E8	OTG_HCINT15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x6EC	OTG_ HCINTMSK15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x6F0	OTG_ HCTSIZ15	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6F4	OTG_HCDMA15	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x800	OTG_DCFG	Res.	Res.	Res.	Res.	Res.	Res.	PERS CHI VL		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM	XCVRDLY	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値							0	0									0	0		0	0	0	0	0	0	0	0	0		0	0	0
0x804	OTG_DCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.	Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
	リセット値														0							0	0	0	0	0	0	0	0	0	0	1	0
0x808	OTG_DSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEV LN STS		FNSOF															Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSPSTS
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x810	OTG_DIEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	Res.	TXFURM	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
	リセット値																			0					0		0	0	0	0	0	0	0
0x814	OTG_DOEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NYETMSK	NAKMSK	BERRM	Res.	Res.	Res.	OUTPKTERRM	Res.	B2BSTUPM	Res.	OTEPDM	STUPM	AHBERRM	EPDM	XFRCM
	リセット値																		0	0	0				0		0		0	0	0	0	0
0x818	OTG_DAINT	OEPINT																IEPINT															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x81C	OTG_ DAINTMSK	OEPM																IEPM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x830	OTG_ DTHRCTL	Res.	Res.	Res.	Res.	ARPEN :	Res.	RXTHRLEN									RXTHREN	Res.	Res.	Res.	Res.	TXTHRLEN										ISOTHREN	NONISOTHREN
	リセット値					0											0					0	0	0	0	0	0	0	0	0	0	0	0
0x834	OTG_DIE PEMPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTXFEM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x900	OTG_ DIEPCTL0 （INT/BULK）	EPENA	EPDIS	SD1PID	SD1P0D	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
	OTG_ DIEPCTL0 （ISO）	EPENA	EPDIS	SODDFRM	SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x908	OTG_ DIEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
	リセット値																			0		0			0	1	0	0	0	0	0	0	0
0x910	OTG_ DIEPTSIZ0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTC NT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値												0	0													0	0	0	0	0	0	0
0x914	OTG_ DIEPDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x918	OTG_ DTXFSTS0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
. . . . .	. . . .	オフセット0x900から始まるレジスタの前のブロックを繰り返します																															
0x9E0	OTG_ DIEPCTL7 （INT/BULK）	EPENA	EPDIS	SD1PID	SD1P0D	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
	OTG_ DIEPCTL7 （ISO）	EPENA	EPDIS	SODDFRM	SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x9E8	OTG_ DIEPINT7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	Res.	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
	リセット値																			0		0			0	1	0	0	0	0	0	0	0
0x9F0	OTG_ DIEPTSIZ7	Res.	MCNT		PKTCNT									XFRSIZ																			
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x9F8	OTG_ DTXFSTS7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0xB00	OTG_ DOEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0					0	0	0	0	0		1													0	0
0xB08	OTG_ DOEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	Res.	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0				0		0	0	0	0	0	0	0
0xB10	OTG_ DOEPTSIZ0	Res.	STUPCNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値		0	0										0													0	0	0	0	0	0	0
0xB14	OTG_ DOEPDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB20	OTG_ DOEPCTL1 （INT/BULK）	EPENA	EPDIS	SD1PID	SD0PID	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP		NAKSTS	DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
	OTG_ DOEPCTL1 （ISO）	EPENA	EPDIS	SODDFRM	SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP		NAKSTS	EONUM	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0

表764.レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB28	OTG_ DOEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	Res.	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0				0		0	0	0	0	0	0	0
0xB30	OTG_ DOEPTSIZ1	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB34	OTG_ DOEPDMA1	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. オフセット0xB20で始まるレジスタの反復先行ブロック .																															
0xC00	OTG_ DOEPCTL8 （INT/BULK）	EPENA	EPDIS	SD1PID	SD0PID	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP		NAKSTS	DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
	OTG_ DOEPCTL8 （ISO）	EPENA	EPDIS	SODDFRM	SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP		NAKSTS	EONUM	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0xC08	OTG_ DOEPINT8	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	Res.	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0		-		0		0	0	0	0	0	0	0
0xC10	OTG_ DOEPTSIZ8	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC14	OTG_ DOEPDMA8	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE00	OTG_ PCGCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHYSLEEP	ENL1GTG	PHYSUSP	Res.	Res.	GATEHCLK	STPPCLK
	リセット値																									0	0	0	0			0	0
0xE04	OTG_ PCGCCTL1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RAMGATEEN	CNTGATECLK		GATEEN
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

73.15 OTG_HS プログラミングモデル

73.15.1 コアの初期化

アプリケーションはコア初期化シーケンスを実行する必要があります。パワーアップ中にケーブルが接続された場合、OTG_GINTSTSレジスタの現在の動作モードビット（OTG_GINTSTSのCMODビット）はモードを反映します。OTG_HS コントローラは、Aプラグが接続されたときにはホストモードに、Bプラグが接続されたときにはデバイスモードに入ります。

このセクションでは、電源投入後の OTG_HS コントローラの初期化について説明します。アプリケーションは、ホストモードかデバイスモードかに関係なく、初期化シーケンスに従う必要があります。コアのすべてのグローバルレジスタは、コアの設定に従って以下の手順で初期化されます。

1. OTG_GAHBCFGレジスタの以下のフィールドをプログラムします。
   - グローバル割込みマスクビットGINTMSK = 1
   - Rx FIFOノットエンプティ（OTG_GINTSTSのRXFLVLビット）
   - 周期的Tx FIFOエンプティレベル
2. OTG_GUSBCFGレジスタの以下のフィールドをプログラムします。
   - OTG_HS タイムアウト較正フィールド
   - USBターンアラウンド時間フィールド
3. ソフトウェアは、OTG_GINTMSKレジスタの以下のビットのマスクを解除する必要があります。

   OTG割込みマスク

   モード不一致割込みマスク
4. ソフトウェアは、OTG_GINTSTSレジスタのCMODビットを読み出すことによって、OTG_HS コントローラがホストモードで動作中か、デバイスモードで動作中かを判別することができます。

73.15.2 ホストの初期化

コアをホストとして初期化するには、アプリケーションは、以下の手順を実行しなければなりません。

1. OTG_GINTMSKレジスタのHPRTINTビットをプログラムして、マスクを解除します。
2. OTG_HCFGレジスタをプログラムして、フルスピードホストを選択します。
3. OTG_HPRTレジスタのPPWRビットを1にプログラムします。ポートには電力供給されていると考えられますが、実際にUSBのV_{BUS} が駆動されていないため、システムの外部回路を制御してV_{BUS} が生成されるようにするには、別の対処が必要となります。
4. OTG_HPRT0のPCDET割込みを待ちます。これは、デバイスがポートに接続されていることを示します。
5. OTG_HPRTレジスタのPRSTビットを1にプログラムします。これによってリセットプロセスが開始されます。
6. リセットプロセスが完了するまで、少なくとも10 ms待ちます。
7. OTG_HPRTレジスタのPRSTビットを0にプログラムします。
8. OTG_HPRTレジスタのPENCHNG割込みを待ちます。
9. OTG_HPRT レジスタのPSPD ビットを読み出して、エニュメレーションされたスピードを取得します。
10. HFIRレジスタを、選択されたPHYクロック1に対応する値でプログラムします。
11. OTG_HCFGレジスタのFSLSPCSフィールドを、手順9で検出したデバイスの速度に従ってプログラムします。FSLSPCSが変更されている場合は、ポートのリセットを行う必要があります。
12. OTG_GRXFSIZレジスタをプログラムして、受信FIFOのサイズを選択します。
13. OTG_HNPTXFSIZレジスタをプログラムして、非周期的トランザクションの非周期的送信FIFOのサイズと開始アドレスを選択します。
14. OTG_HPTXFSIZレジスタをプログラムして、周期的トランザクションの周期的送信FIFOのサイズと開始アドレスを選択します。

デバイスと通信するには、システムソフトウェアは、少なくとも1つのチャネルを初期化し、有効にする必要があります。

73.15.3 デバイスの初期化

アプリケーションは、パワーアップ時、またはホストモードからデバイスモードへの変更後、以下の手順を実行して、コアをデバイスとして初期化する必要があります。

1. OTG_DCFGレジスタの以下のフィールドをプログラムします。
   - デバイススピード
   - 非ゼロ長ステータスOUTハンドシェイク
   - 周期的フレームインターバル
2. デバイス閾値制御レジスタをプログラムします。これは、DMA モードを使用しており、閾値を有効にする予定がある場合のみ必須です。
3. DCTL.SDISビットをクリアします。コアはこのビットがクリアされた後に接続を発行します。
4. OTG_GINTMSKレジスタをプログラムして、次の割込みのマスクを解除します。
   - USBリセット
   - エニュメレーション終了
   - アーリーサスペンド
   - USBサスペンド
   - SOF
5. OTG_GINTSTS の USBRST 割込みを待ちます。これは、この割込みの受信時、約 10ms 続くリセットがUSBで検出されたことを示します。
6. OTG_GINTSTSのENUMDNE割込みを待ちます。この割込みは、USB上でリセットが終了したことを示します。この割込みを受信したとき、アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーション速度を判別し、にリストされている手順を実行する必要があります。3396ページのエニュメレーション完了時のエンドポイント初期化

この時点で、デバイスは、SOFパケットを受け入れて、コントロールエンドポイント0で制御転送を実行する準備ができたことになります。

73.15.4 DMAモード

OTGホストはAHBマスタインタフェースを使って、送信パケットデータ （AHBからUSBへ）をフェッチし、データ更新 （USBからAHBへ）を受信します。AHBマスタはプログラムされたDMAアドレス（ホストモードでのOTG_HCDMAxレジスタとペリフェラルモードでのOTG_DIEPDMAx/OTG_DOEPDMAxレジスタ）を使って、データバッファにアクセスします。

73.15.5 ホストプログラミングモデル

チャネルの初期化

アプリケーションが接続されたデバイスと通信するためには、1つまたは複数のチャネルを初期化する必要があります。チャネルを初期化して、有効にするには、アプリケーションは、以下の手順を実行する必要があります。

1. OTG_GINTMSKレジスタをプログラムして、以下の割込みのマスクを解除します。
2. チャネル割込み
   - OUTトランザクションの非周期的送信FIFOエンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
   - OUTトランザクションの非周期的送信FIFO半エンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）

3. OTG_HAINTMSKレジスタをプログラムして、選択されたチャネルの割込みのマスクを解除します。
4. OTG_HCINTMSKレジスタをプログラムして、ホストチャネル割込みレジスタで指定された、所定のトランザクション関連割込みのマスクを解除します。
5. 選択されたチャネルのOTG_HCTSIZxレジスタを、合計転送サイズ（バイト数）およびショートパケットを含む予想されるパケット数でプログラムします。アプリケーションは、PIDフィールドを初期データPID（最初のOUTトランザクションで使用されるPID、または最初のINトランザクションから期待されるPID）でプログラムする必要があります。
6. 選択されたチャネルのOTG_HCCHARxレジスタを、タイプ、スピード、方向など、デバイスのエンドポイントの特性でプログラムします。（チャネルは、アプリケーションがパケットを送信または受信する準備ができている場合にのみ、チャネルイネーブルビットを1にセットすることで有効にできます）。
7. OTG_HCSPLTx レジスタ内の選択したチャネルにハブアドレスとポートアドレスをプログラムします （スプリットトランザクションの場合）。
8. OTG_HCDMAxレジスタ内の選択したチャネルにバッファ開始アドレスをプログラムします（DMAトランザクションのみ）。

チャネルの停止

アプリケーションは、OTG_HCCHARxレジスタのCHDISおよびCHENAビットを1にセットすることによって、任意のチャネルを無効にできます。これにより、OTG_HS ホストは、要求されたリクエスト（ある場合）を排出して、チャネル停止割込みを生成できます。アプリケーションは、チャネルを他のトランザクションに再割り当てする前に、OTG_HCINTx内のCHH割込みを待つ必要があります。OTG_HS ホストは、USB上ですでに転送が開始しているトランザクションに対しては割込みを行いません。

DMAモード動作でチャンネルを無効化するときは、アプリケーションはリクエストキュー内のスペースをチェックする必要はありません。OTG_HS ホストは、調停時の無効化されたチャネルの回転に関するディセーブル

リクエストを書き込むスペースがあるかをチェックします。その間、OTG_HCCHARxのCHDISビットに1が設定されたとき、要求された全リクエストはリクエストキューからドロップされます。

チャネルを無効にする前に、アプリケーションは、非周期的リクエストキュー（非周期的チャネルを無効にするとき）、または周期的リクエストキュー（周期的チャネルを無効にするとき）に使用可能な空きスペースが少なくとも1つ以上存在することを確認する必要があります。アプリケーションは、（チャネルを無効化する前に）リクエストキューがフル状態のときには、OTG_HCCHARxレジスタのCHDISビットに1をセットし、CHENAビットを0にクリアすることで、ポストされたリクエストを一掃できます。

アプリケーションは、以下のいずれかの条件のとき、チャネルを無効にすることが期待されます。

1. INまたはOUTチャネルで、OTG_HCINTxレジスタのSTALL、TXERR、BBERR、またはDTERR割込みが受信されたとき。アプリケーションは、停止割込みを受信する前に、同じチャネルについての他の割込み（DTERR、NAK、データ、TxERR）を受信できなければなりません。
2. 非周期的IN転送または広帯域割込みIN転送中にOTG_HCINTxのXFRC割込みが受信されたとき。
3. OTG_GINTSTSレジスタのDISCINT（デバイス切断）割込みを受信したとき。（アプリケーションは、すべての有効なチャネルを無効にすることが期待されます）。
4. アプリケーションが正常完了の前に転送を中断したとき。

ピンプロトコル

OTG_HS ホストが ハイスピードで動作する場合、ハイスピードバルクエンドポイントまたはコントロール（データおよびステータスステージ）OUTエンドポイントと交信中に、アプリケーションはピンプロトコルを開始する必要があります。アプリケーションはNAK/NYET/TXERR割込みを受信したとき、ピンプロトコルを開始する必要があります。OTG_HS ホストが上記応答の1つを受信した場合、特定のエンドポイントに対するトランザクションを継続しないで、（リクエストキューから）ポストまたはフェッチした全OUTリクエストをドロップさせ、（送信FIFOから）対応するデータを排出させます。これはスレーブモードでのみ有効です。スレーブモードでは、アプリケーションは、チャネルを有効化する前にOTG_HCTSIZx内のDOPINGビットをセットするか、またはチャネルが有効化済みの場合OTG_HCTSIZxレジスタのDOPINGビットにセットを書き込むことにより、pingトークンを送信することができます。これにより、OTG_HS ホストがpingリクエストエントリをリクエストキューに書き込めるようになります。アプリケーションは、ピントークンに対する応答 （NAK、ACK、またはTXERR割込み）を待った後に、トランザクションを継続するか、または別のピントークンを送信する必要があります。アプリケーションは、リクエストされたピンのOUTエンドポイントからのACKを受信した後にのみデータトランザクションを継続することができます。DMAモード動作では、アプリケーションは、バルク／コントロールOUTの場合NAK/NYET応答に対してOTG_HCTSIZx内のDOPINGビットをセットする必要はありません。OTG_HSホストはOTG_HCTSIZx内のDOPINGビットを自動的にセットし、バルク／コントロールOUTに対してpingトークンを発行します。OTG_HSホストは、ACKを受信するまでpingトークンの送信を続け、自動的にデータトランザクションに切り替わります。

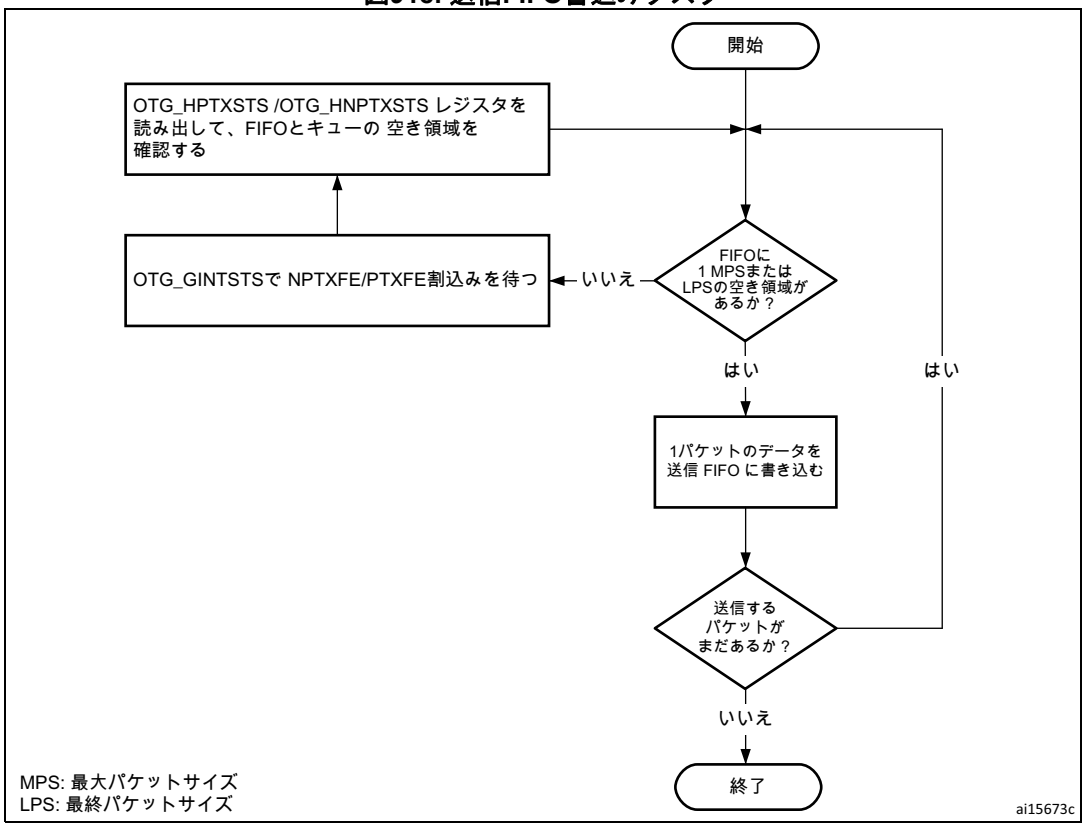
動作モデル

アプリケーションが接続されたデバイスと通信するためには、チャネルを初期化する必要があります。このセクションでは、さまざまなタイプのUSBトランザクションについて、実行しなければならない動作のシーケンスについて説明します。

- **送信FIFOへの書込み**

  OTG_HS ホストは、パケットの最後の32-bitワードの書込みとともに、周期的/非周期的リクエストキューにエントリ（OUTリクエスト）を自動的に書き込みます。アプリケーションは、送信FIFOへの書込みを開始する前に、周期的/非周期的リクエストキューに少なくとも1つ以上の空きスペースがあることを確認しなければなりません。アプリケーションは送信FIFOに必ず32-bitワード単位で書込みを行う必要があります。パケットサイズが32-bitワード単位で揃えられていない場合、アプリケーションはパディングを使用する必要があります。OTG_HS ホストは、プログラムされた最大パケットサイズと転送サイズに基づいて、実際のパケットサイズを決定します。

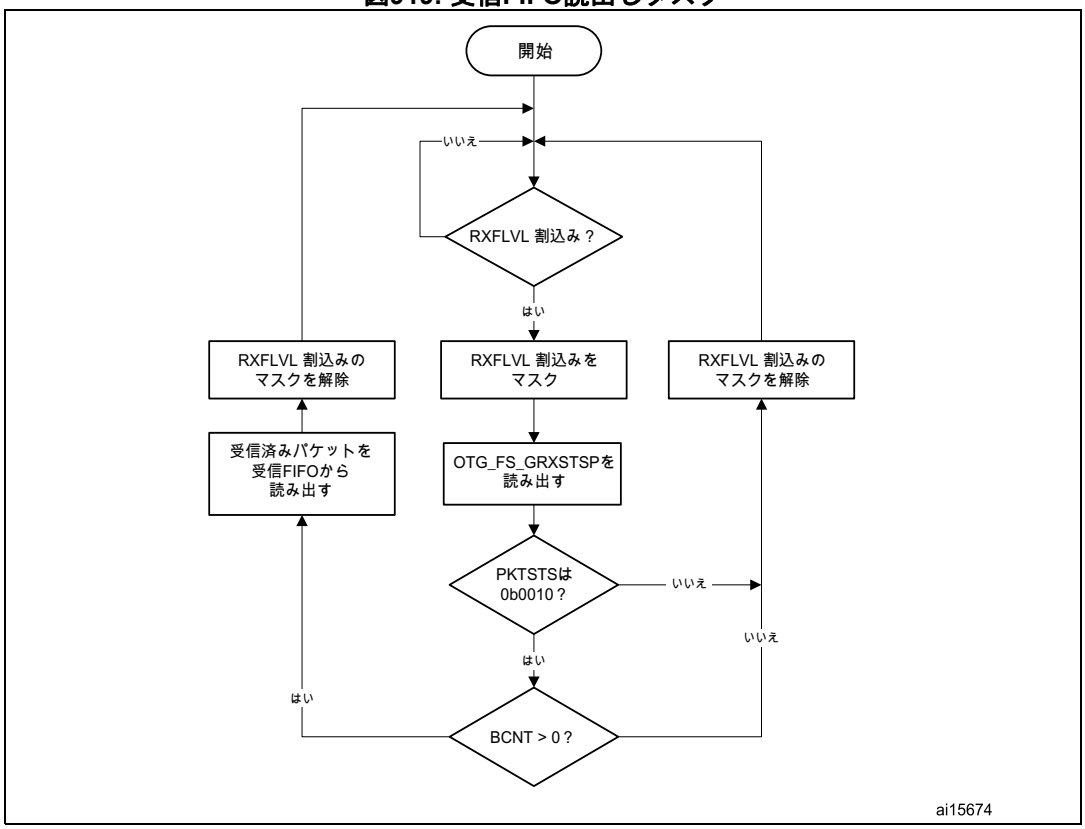
図918. 送信FIFO書込みタスク



- **受信FIFOの読出し**

  アプリケーションは、INデータパケット（bx0010）以外のすべてのパケットステータスを無視しなければなりません。

図919. 受信FIFO読出しタスク



- **バルクおよびコントロールOUT／SETUPトランザクション**

  典型的なバルクまたはコントロールOUT/SETUPパイプライントランザクションレベルの動作を図920に示します。チャネル1（ch_1）に注目してください。2つのバルクOUTパケットが転送されます。コントロールSETUPトランザクションは同じように動作しますが、1パケットのみです。ここでは、以下のように仮定されています。

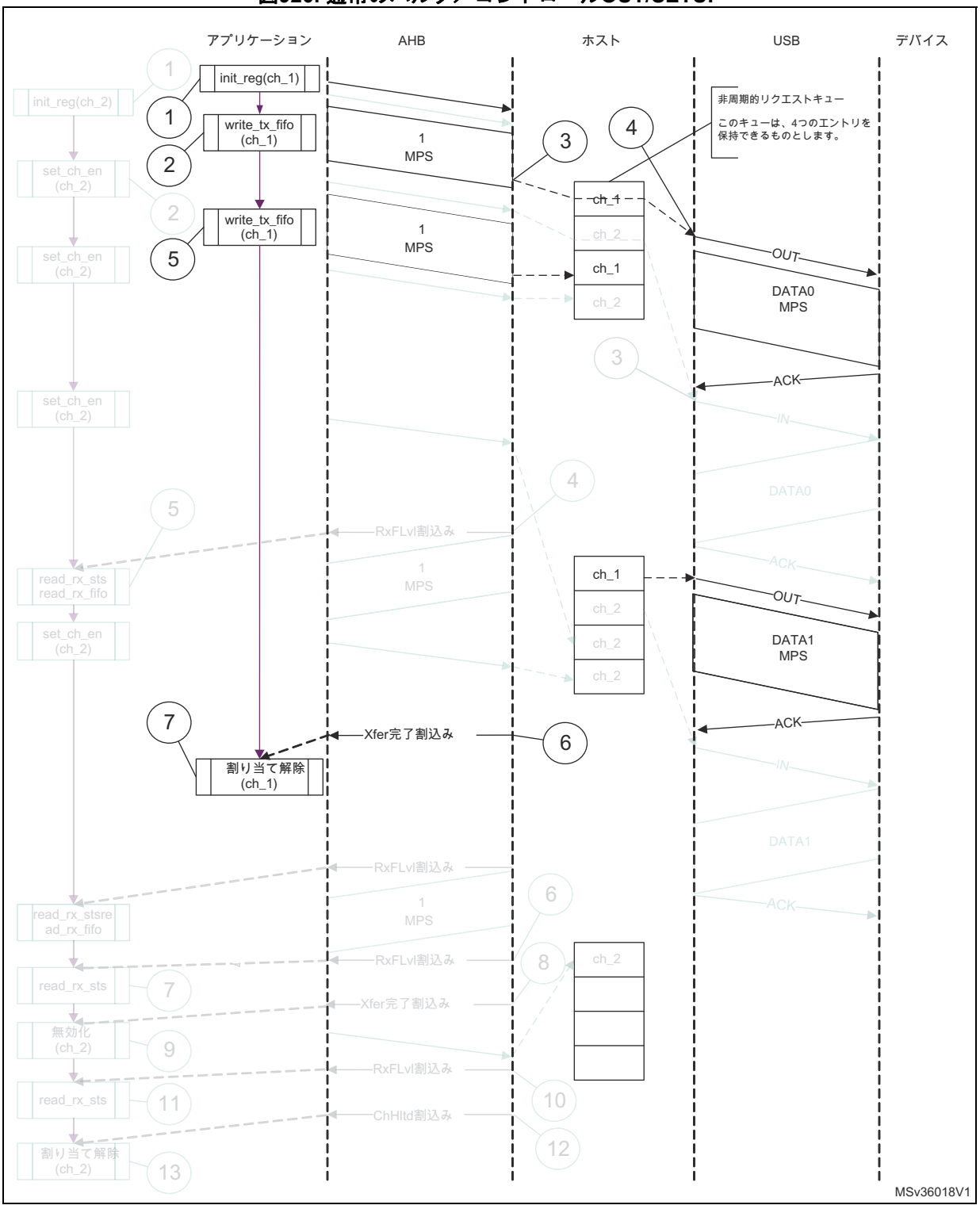
  – アプリケーションは、最大パケットサイズの2つのパケットを送信しようとしています（転送サイズ = 1,024バイト）。
  – 非周期的送信FIFOは2つのパケットを保持することができます（HSの場合1KB）。
  – 非周期的リクエストキューの深さ = 4。

- **通常のバルクおよびコントロールOUT/SETUP動作**

  動作シーケンス（チャネル1）は、次のとおりです。

1. チャネル1を初期化します。
2. チャネル1の最初のパケットを書き込みます。
3. 最後のワードの書込みと同時に、コアは、非周期的リクエストキューにエントリを書き込みます。
4. 非周期的キューがエンプティでなくなるとすぐに、コアは、現在のフレームで OUT トークンの送信を試みます。
5. チャネル1の2番目（最後）のパケットを書き込みます。
6. コアは、最後のトランザクションが正常に完了すると、XFRC割込みを生成します。
7. XFRC割込みに応答して、他の転送のためにチャネル割り当てを解除します。
8. 非ACK応答の取り扱い

図920. 通常のバルク／コントロールOUT/SETUP



1. 灰色の要素は、この図のコンテキストでは無関係です。

バルクおよびコントロールOUT/SETUPトランザクションのためのチャネル固有の割込みサービスルーチンを、以下のコードサンプルで示します。

- **バルク／コントロール OUT/SETUP とバルクまたはコントロール IN トランザクションの割込みサービスルーチン**

  a) バルク／コントロールOUT/SETUP

```
Unmask (NAK/TXERR/STALL/XFRC)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else if (STALL)
   {
   Transfer Done = 1
   Unmask CHH
   Disable Channel
   }
else if (NAK or TXERR )
   {
   Rewind Buffer Pointers
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   else
      {
      Reset Error Count
      }
      }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
```

```
   else if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

アプリケーションは、送信FIFOおよびリクエストキューに空きがある場合、データパケットを送信FIFOに書き込むことが期待されます。アプリケーションは、OTG_GINTSTSレジスタのNPTXFE割込みを使用して、送信FIFOのスペースを見つけることができます。

b) バルク／コントロールIN

```
Unmask (TXERR/XFRC/BBERR/STALL/DTERR)
if (XFRC)
   {
   Reset Error Count
   Unmask CHH
   Disable Channel
   Reset Error Count
   Mask ACK
   }
else if (TXERR or BBERR or STALL)
   {
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
else if (ACK)
   {
   Reset Error Count
   Mask ACK
   }
```

```
else if (DTERR)
   {
   Reset Error Count
   }
```

アプリケーションは、リクエストキューのスペースに空きがあるときは、XFRC割込みが受信されるまで、リクエストを書き込むことが期待されます。

- **バルクおよびコントロールINトランザクション**

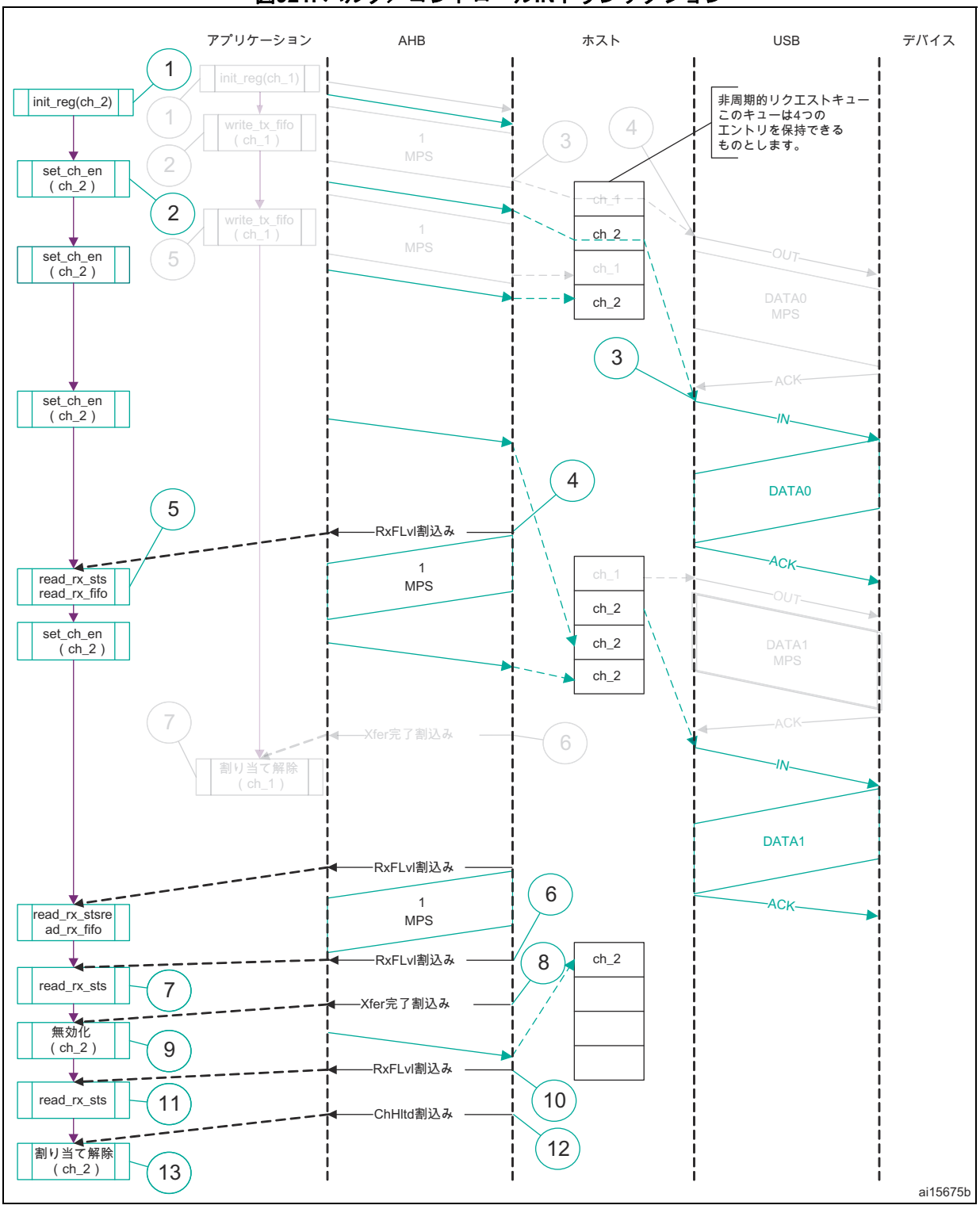
  典型的なバルクまたはコントロールINパイプライントランザクションレベルの動作を図921に示します。チャネル2（ch_2）に注目してください。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズ（転送サイズ = 1,024バイト）の2つのパケットを受信しようとしています。

  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットと、パケットあたり2つのステータスワード（HSの場合520バイト）を収容することができます。

  – 非周期的リクエストキューの深さ = 4。

図921. バルク／コントロールINトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。
2. OTG_HCCHAR2のCHENAビットをセットして、INリクエストを非周期的リクエストキューに書き込みます。
3. コアは、現在のOUTトランザクションの完了後、INトークンの送信を試みます。
4. コアは、受信したパケットが受信FIFOに書き込まれると、RXFLVL割込みを生成します。
5. RXFLVL割込みに応答して、RXFLVL割込みをマスクし、受信パケットステータスを読み出して受信バイト数を調べた後、それに応じて受信FIFOを読み出します。続いて、RXFLVL割込みのマスクを解除します。
6. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。
7. アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスがINデータパケットでないとき（OTG_GRXSTSR ≠ 0b0010のPKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると、XFRC割込みを生成します。
9. XFRC割込みに応答して、チャネルを無効にし、それ以降のリクエストについて、OTG_HCCHAR2レジスタへの書込みを停止します。コアは、OTG_HCCHAR2に書き込まれると、非周期的リクエストキューにチャネルディセーブルリクエストを書き込みます。
10. コアは、停止ステータスが受信FIFOに書き込まれると、RXFLVL割込みを生成します。
11. 受信パケットステータスを読み出して、それを無視します。
12. コアは、停止ステータスが受信FIFOからポップされるとすぐに、CHH割込みを生成します。
13. CHH割込みに応答して、他の転送のためにチャネル割り当てを解除します。
14. 非ACK応答の取り扱い

- **コントロールトランザクション**

  コントロール転送のセットアップ、データ、およびステータスステージは3つの個別の転送として実行されなければなりません。セットアップ、データ、およびステータスステージのOUTトランザクションは、すでに説明したバルクOUTトランザクションと同様に実行されます。データまたはステータスステージのINトランザクションは、すでに説明したバルクINトランザクションと同様に実行されます。3つのステージすべてについて、アプリケーションはOTG_HCCHAR1レジスタのEPTYPフィールドを制御に設定することが期待されます。セットアップステージでは、アプリケーションはOTG_HCTSIZ1レジスタのPIDフィールドをSETUPに設定することが期待されます。

- **インタラプトOUTトランザクション**

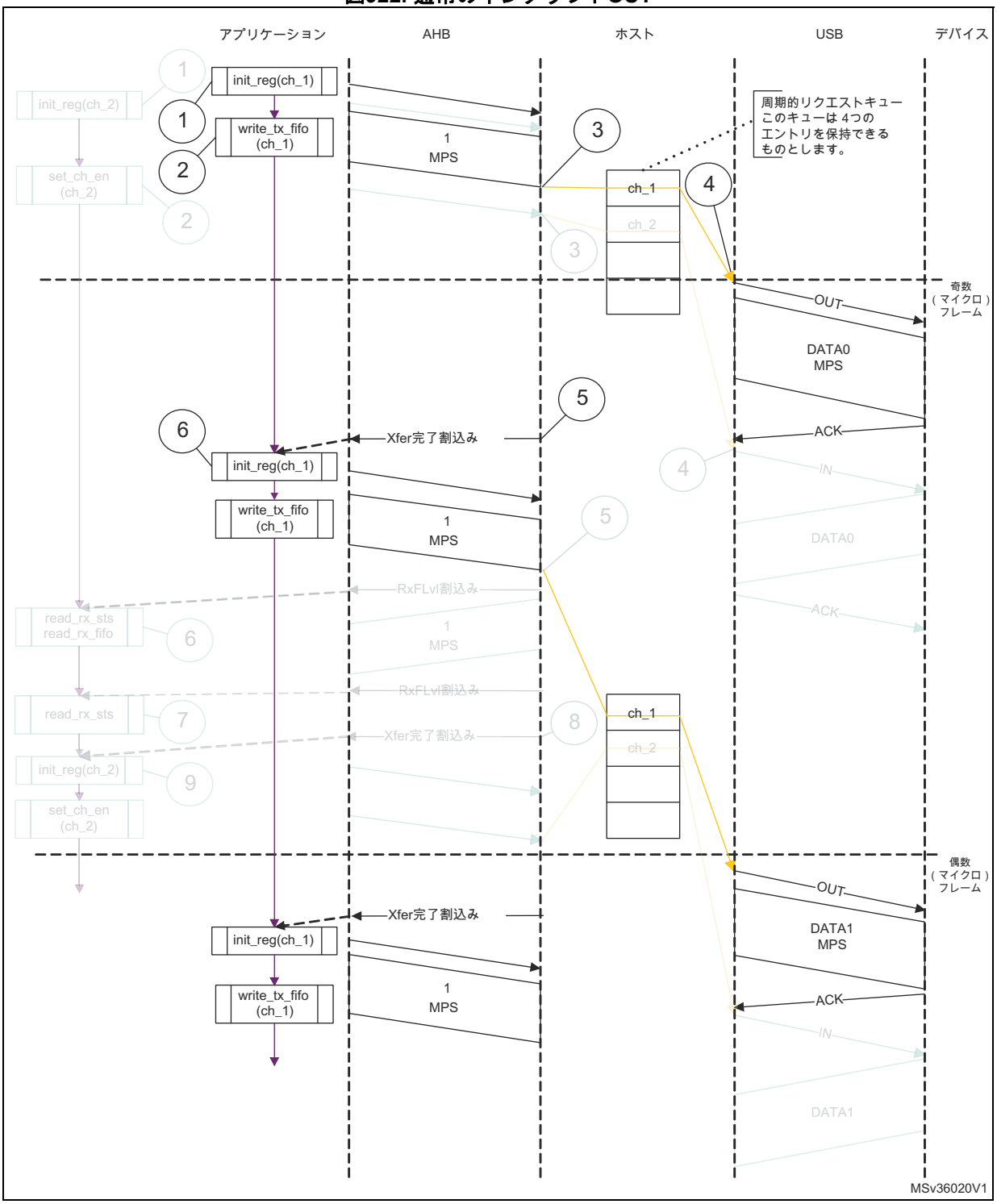
  典型的なインタラプトOUT動作を図922に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1 024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の送信を試みています。
  – 周期的送信FIFOは1パケット（1KB）を保持することができます。
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル1を初期化し、有効にします。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。
2. チャネル1の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次の（奇数）フレームでOUTトークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC割込みを生成します。
6. XFRC割込みに応答して、次の転送のためにチャネルを再初期化します。

図922. 通常のインタラプトOUT



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **インタラプトOUT/INトランザクションのための割込みサービスルーチン**

  a) インタラプトOUT

```
Unmask (NAK/TXERR/STALL/XFRC/FRMOR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else
   if (STALL or FRMOR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL)
         {
         Transfer Done = 1
         }
      }
   else
      if (NAK or TXERR)
         {
         Rewind Buffer Pointers
         Reset Error Count
         Mask ACK
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
         {
         Mask CHH
         if (Transfer Done or (Error_count == 3))
            {
            De-allocate Channel
            }
         else
            {
            Re-initialize Channel (in next b_interval - 1 Frame)
            }
         }
      else
         if (ACK)
         {
```

```
        Reset Error Count
        Mask ACK
        }
```

アプリケーションは OTG_GINTSTS レジスタの NPTXFE 割込みを使用して、送信 FIFO のスペースを見つけます。

インタラプト IN

```
Unmask (NAK/TXERR/XFRC/BBERR/STALL/FRMOR/DTERR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   if (OTG_HCTSIZx.PKTCNT == 0)
      {
      De-allocate Channel
      }
   else
      {
      Transfer Done = 1
      Unmask CHH
      Disable Channel
      }
   }
else
   if (STALL or FRMOR or NAK or DTERR or BBERR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL or BBERR)
         {
         Reset Error Count
         Transfer Done = 1
         }
      else
         if (!FRMOR)
         {
         Reset Error Count
         }
      }
else
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      Unmask CHH
```

```
      Disable Channel
      }
else
   if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
         {
         De-allocate Channel
         }
      else
         Re-initialize Channel (in next b_interval - 1 /Frame)
      }
   }
else
   if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

- **インタラプトINトランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の受信を試みています。

  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットとパケットあたり2つのステータスワード（1,031バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

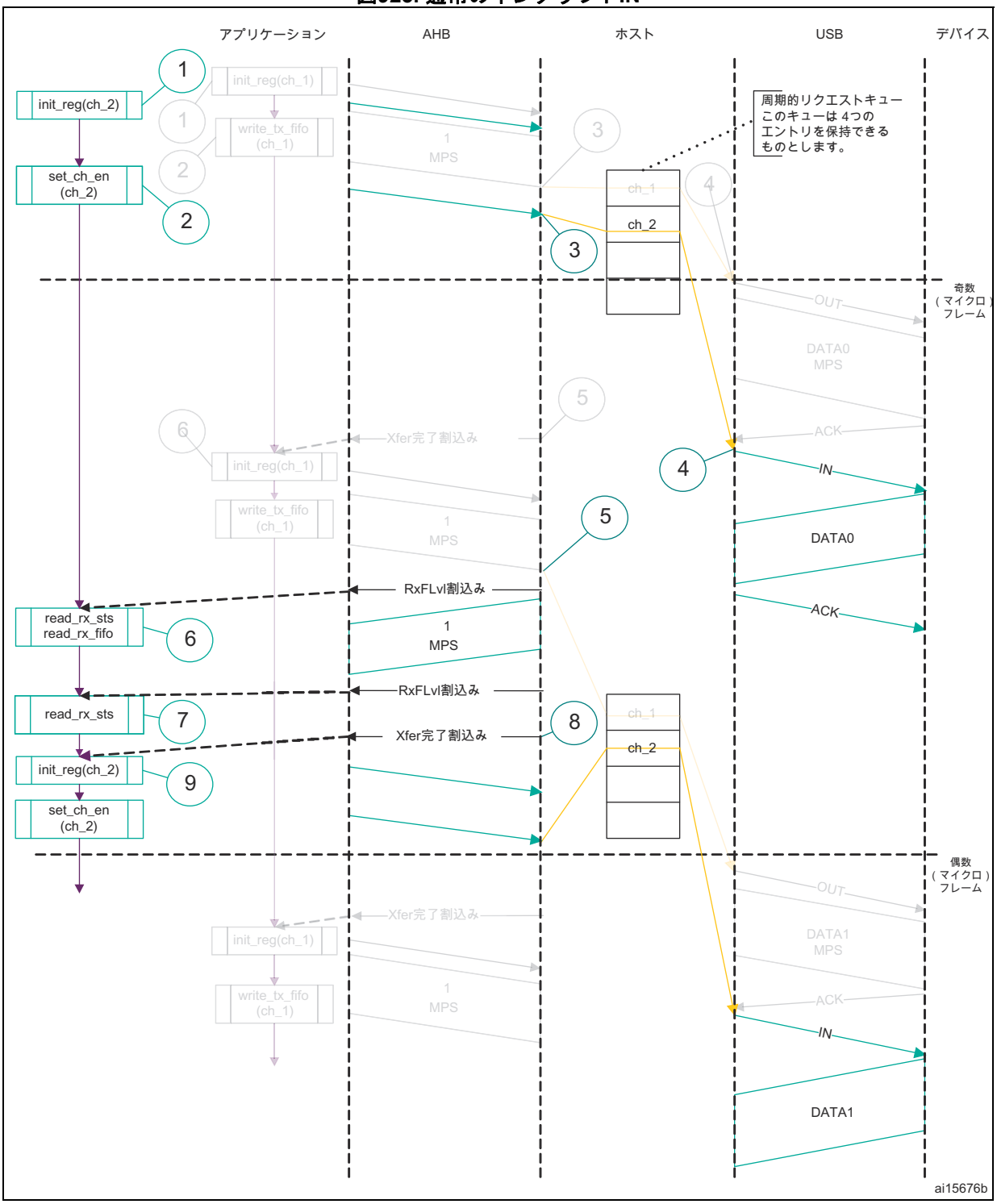
- **通常のインタラプトIN動作**

  動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをセットする必要があります。
2. OTG_HCCHAR2のCHENAビットをセットしてINリクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENAビットがセットされたOTG_HCCHAR2レジスタの書込みごとに、周期的リクエストキューにINリクエストを書き込みます。
4. OTG_HS ホストは、次の奇数フレームでINトークンの送信を試みます。
5. INパケットが受信され、受信FIFOに書き込まれると、OTG_HS ホストはRXFLVL割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信FIFOを読み出します。アプリケーションは、受信FIFOを読み出す前にRXFLVL割込みをマスクし、すべてのパケットを読み出した後にマスクを解除しなければなりません。
7. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスがINデータパケットでない（GRXSTSRのPKTSTSが0b0010でない）ときには無視する必要があります。

8. コアは、受信パケットステータスが読み出されるとXFRC割込みを生成します。
9. XFRC割込みに応答してOTG_HCTSIZ2のPKTCNTフィールドを読み出します。OTG_HCTSIZ2のPKTCNTビットが 0でない場合は、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2のPKTCNTビットが0の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをリセットする必要があります。

図923. 通常のインタラプトIN



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナスOUTトランザクション**

  典型的なアイソクロナスOUT動作を図924に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレームから始めて、フレームごとに1パケット（最大1パケットサイズ）の送信を試みています。（転送サイズ　= 1,024バイト）。

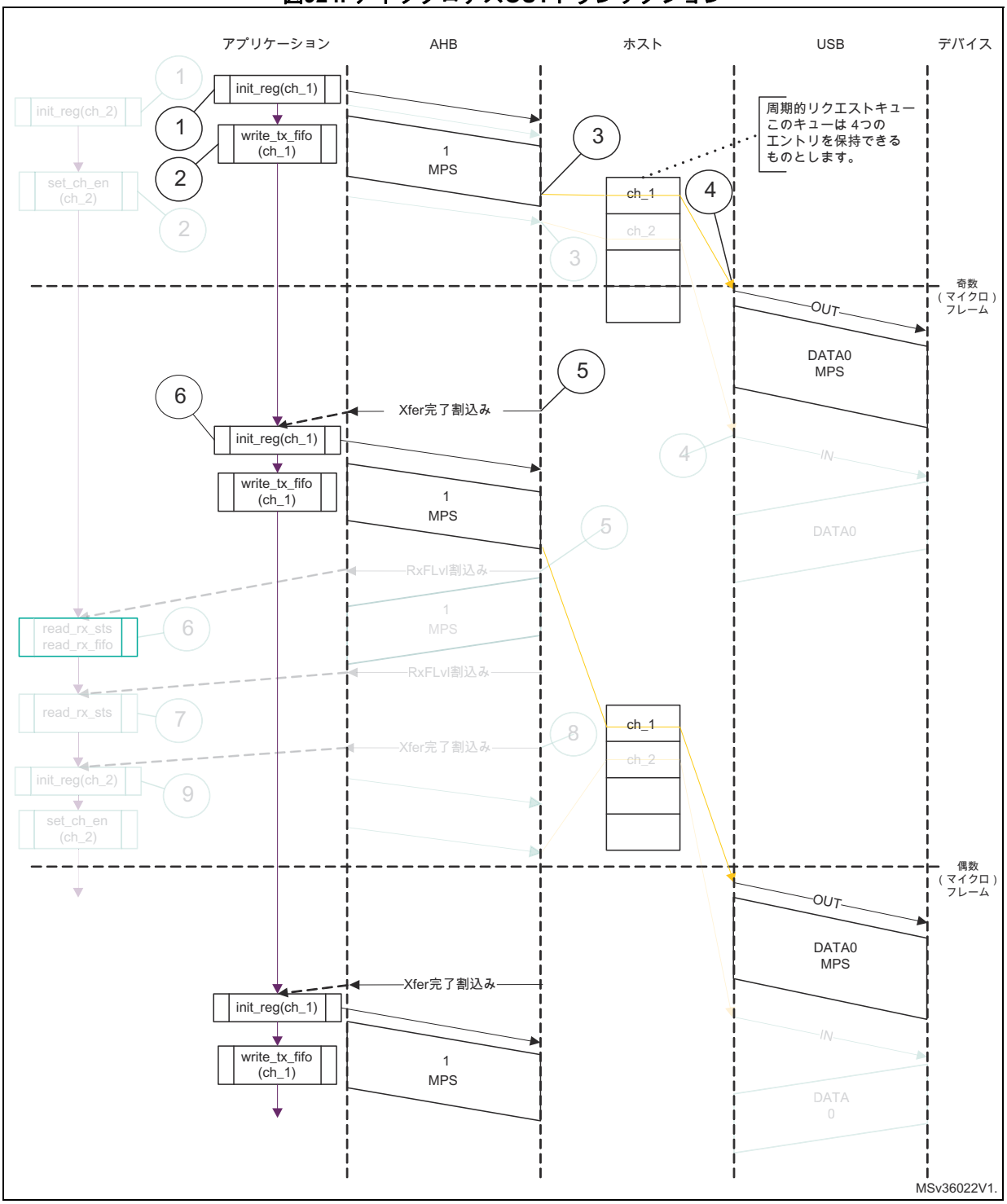
  – 周期的送信FIFOは1パケット（1KB）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル1を初期化し、有効にします。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。
2. チャネル1の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次のフレーム（奇数）でOUTトークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC割込みを生成します。
6. XFRC割込みに応答して、次の転送のためにチャネルを再初期化します。
7. 非ACK応答の取り扱い

図924. アイソクロナスOUTトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナスOUT/INトランザクションのための割込みサービスルーチン**

  コードサンプル：アイソクロナスOUT

```
Unmask (FRMOR/XFRC)
if (XFRC)
   {
   De-allocate Channel
   }
else
   if (FRMOR)
      {
      Unmask CHH
      Disable Channel
      }
   else
   if (CHH)
      {
      Mask CHH
      De-allocate Channel
      }
```

コードサンプル：アイソクロナス IN

```
Unmask (TXERR/XFRC/FRMOR/BBERR)
if (XFRC or FRMOR)
   {
   if (XFRC and (OTG_HCTSIZx.PKTCNT == 0))
      {
      Reset Error Count
      De-allocate Channel
      }
   else
      {
      Unmask CHH
      Disable Channel
      }
   }
   else
      if (TXERR or BBERR)
         {
         Increment Error Count
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
            {
            Mask CHH
```

```
if (Transfer Done or (Error_count == 3))
   {
   De-allocate Channel
   }
else
   {
   Re-initialize Channel
   }
}
```

- **アイソクロナスINトランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、次の奇数フレーム（転送サイズ = 1,024バイト）から始めて、フレームごとに1パケット（最大1パケットサイズ）の受信を試みています。

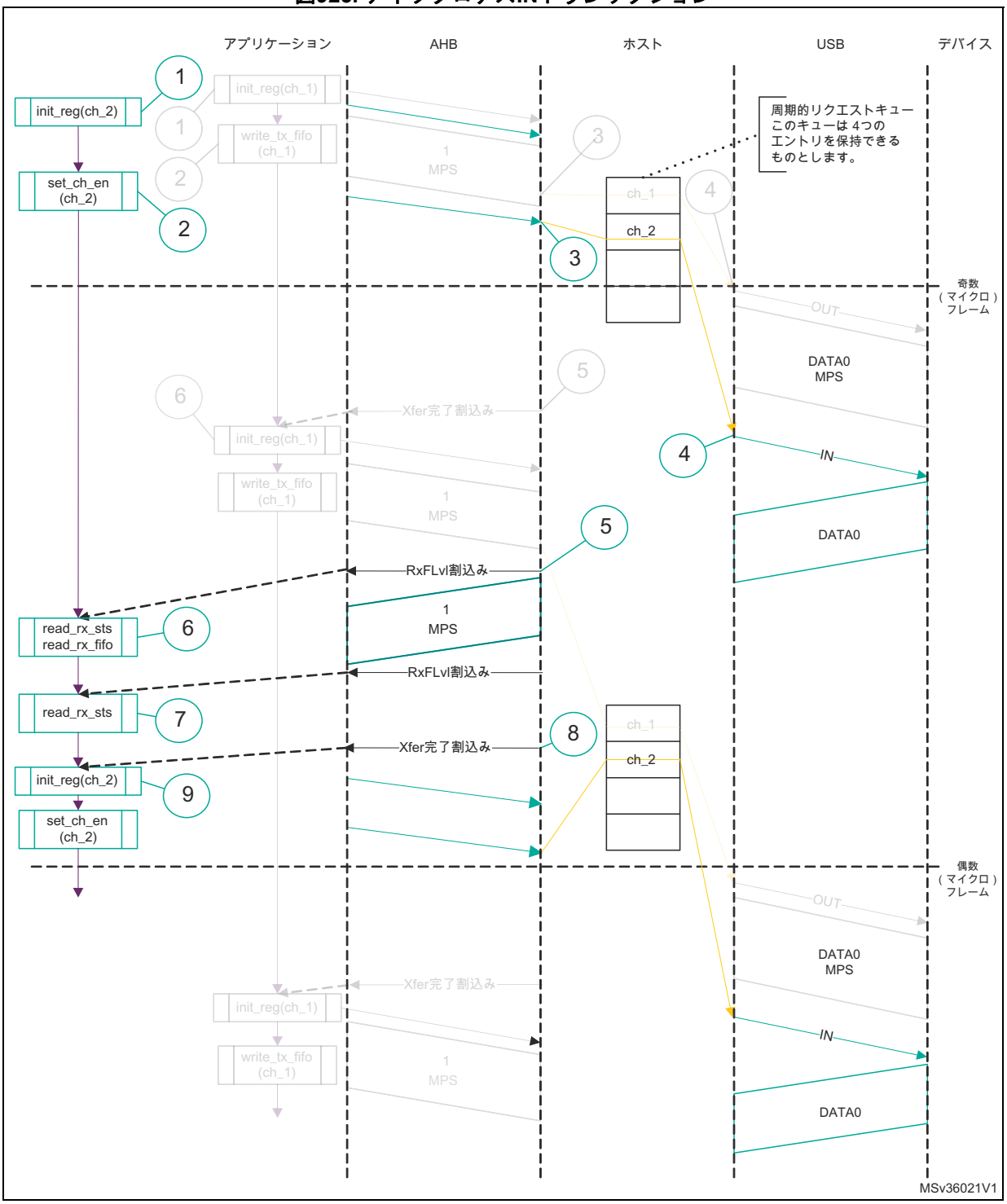
  – 受信FIFOは、少なくとも1つの最大パケットサイズのパケットとパケットあたり2つのステータスワード（1,031バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル2を初期化します。アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをセットする必要があります。
2. OTG_HCCHAR2のCHENAビットをセットしてINリクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENAビットがセットされたOTG_HCCHAR2レジスタの書込みごとに、周期的リクエストキューにINリクエストを書き込みます。
4. OTG_HS ホストは、次の奇数フレームでINトークンの送信を試みます。
5. INパケットが受信され、受信FIFOに書き込まれると、OTG_HS ホストはRXFLVL割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信FIFOを読み出します。アプリケーションは、受信FIFOを読み出す前にRXFVLV割込みをマスクし、すべてのパケットを読み出した後でマスクを解除しなければなりません。
7. コアは、受信FIFO内の転送完了ステータスエントリに対してRXFLVL割込みを生成します。今度は、アプリケーションは受信パケットステータスを読み出して、それがINデータパケットでない（OTG_GRXSTSRのPKTSTSビットが0b0010でない）ときには無視する必要があります。
8. コアは、受信パケットステータスが読み出されるとXFRC割込みを生成します。
9. XFRC割込みに応答してOTG_HCTSIZ2のPKTCNTフィールドを読み出します。OTG_HCTSIZ2のPKTCNT ≠ 0の場合、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2のPKTCNT = 0の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションはOTG_HCCHAR2レジスタのODDFRMビットをリセットする必要があります。

図925. アイソクロナスINトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **キューの深さの選択**

  周期的および非周期的リクエストキューの深さは、アクセスされる周期的/非周期的エンドポイントの数に合うように注意して選択してください。

  非周期的リクエストキューの深さは、非周期的転送の性能に影響を与えます。キューが深いほど（FIFOのサイズが十分であれば）、コアは非周期的転送をより多くパイプライン化できます。キューのサイズが小さいと、コアはキューのスペースが空いたときしか新しいリクエストを入れることができません。

  コアの周期的リクエストキューの深さは、周期的転送をスケジュールどおりに実行するために不可欠です。周期的キューの深さは、マイクロフレーム内でスケジュールされた周期的転送の数に基づいて選択してください。周期的リクエストキューの深さがマイクロフレーム内でスケジュールされた周期的転送の数より小さい場合、フレームオーバーラン条件が発生します。

- **バブル条件の取り扱い**

  OTG_HS コントローラは、パケットバブルとポートバブルの2つのバブルを処理します。パケットバブルは、デバイスがチャネルの最大パケットサイズよりも多くのデータを送信した場合に発生します。ポートバブルは、コアがEOF2（フレーム2の終わりのSOFにきわめて近い）でデバイスからデータを受信し続けると発生します。

  OTG_HS コントローラがパケットバブルを検出すると、Rxバッファへのデータの書込みを停止して、パケットの終了（EOP）を待ちます。EOPを検出すると コントローラはRxバッファにすでに書き込まれたデータを一掃して、アプリケーションに対するバブル割込みを生成します。

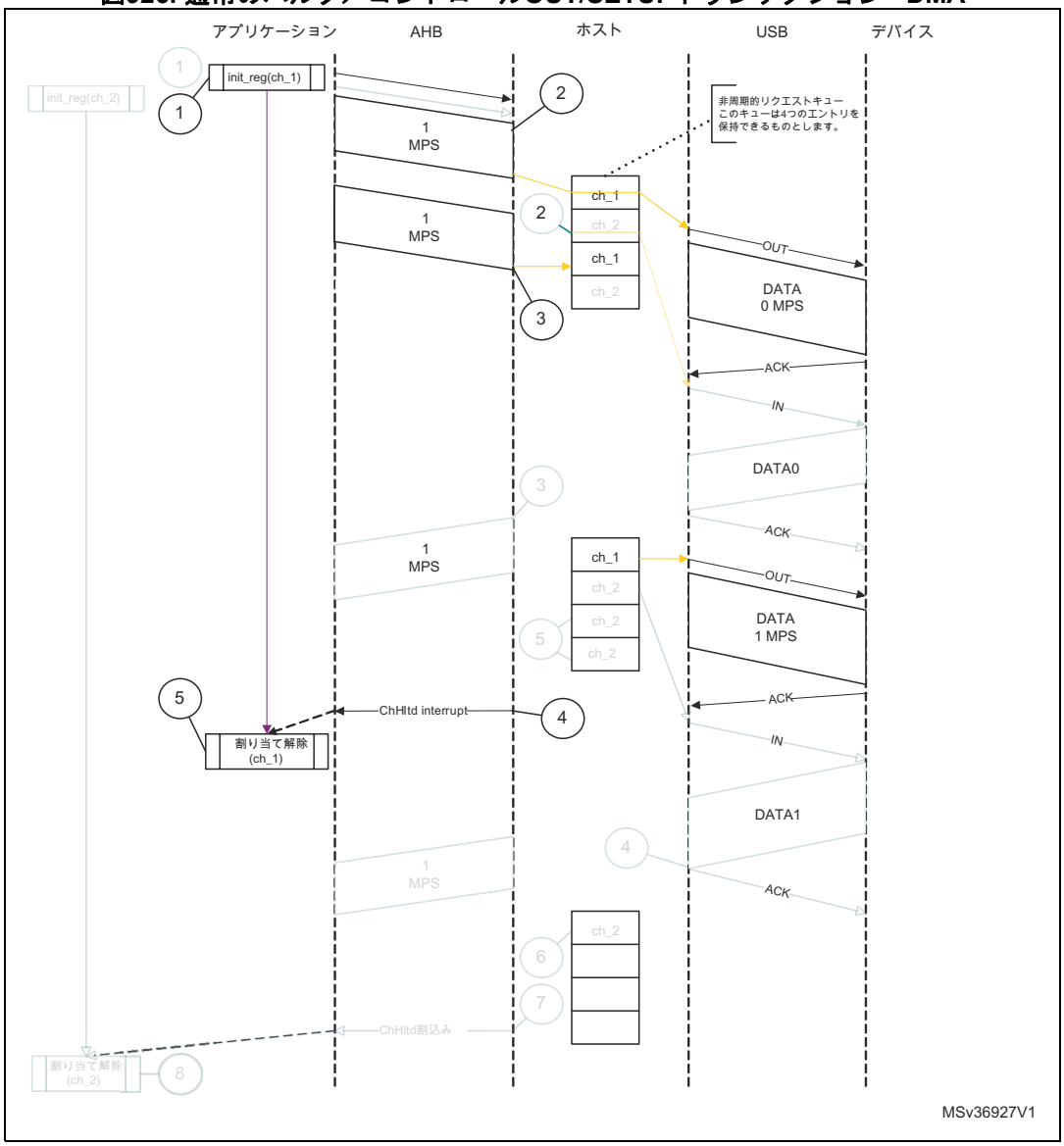
  OTG_HS コントローラがポートバブルを検出すると、Rx FIFOを排出して、ポートを無効にします。コアは、ポートディセーブル割込み（OTG_GINTSTS の HPRTINT、OTG_HPRT の PENCHNG）を生成します。この割込みを受信すると、アプリケーションはOTG_HPRTのPOCAビットをチェックして、これが過電流条件（ポートディセーブル割込みのもう1つの原因）によるものではないことを確認してから、ソフトリセットを行う必要があります。コアは、ポートバブル条件を検出した後は、それ以上トークンを送信しません。

- **バルクおよびコントロールOUT／SETUPトランザクション - DMAモード**

  動作シーケンスは、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネル1を初期化し、有効にします。
2. OTG_HS　ホストはチャネルが有効化されると直ちに最初のパケットのフェッチを開始します。内部DMAモードの場合、OTG_HS ホストはプログラムされたDMAアドレスを使ってパケットを取得します。
3. 2つ目（最後）のパケットの最後の32-bitワードを取得した後、OTG_HS ホストは以降の調停のために内部的にチャネル1をマスクします。
4. OTG_HS ホストは、最後のパケットが送信されると直ちにCHH割込みを生成します。
5. CHH割込みに応答して、他の転送のためにチャネル割り当てを解除します。

図926. 通常のバルク／コントロールOUT/SETUPトランザクション - DMA



- **内部DMAによるNAKとNYETの処理**

1. OTG_HS ホストはバルクOUTトランザクションを送信します。
2. デバイスはNAKまたはNYETで応答します。
3. アプリケーションでNAKまたはNYETのマスクを解除した場合、コアはアプリケーションへ対応する割込みを生成します。コアがバッファポインタの戻し処理を行っている場合およびアプリケーション介入なしでチャンネルを再初期化している場合、アプリケーションはこれらの割込みを生成する必要はありません。
4. コアは自動的にピントークンを発行します。
5. デバイスが ACK を返すと、コアは転送を続けます。オプションとして、アプリケーションはこれらの割込みを使用することができます。この場合NAKまたはNYET割込みはアプリケーションからマスクされます。

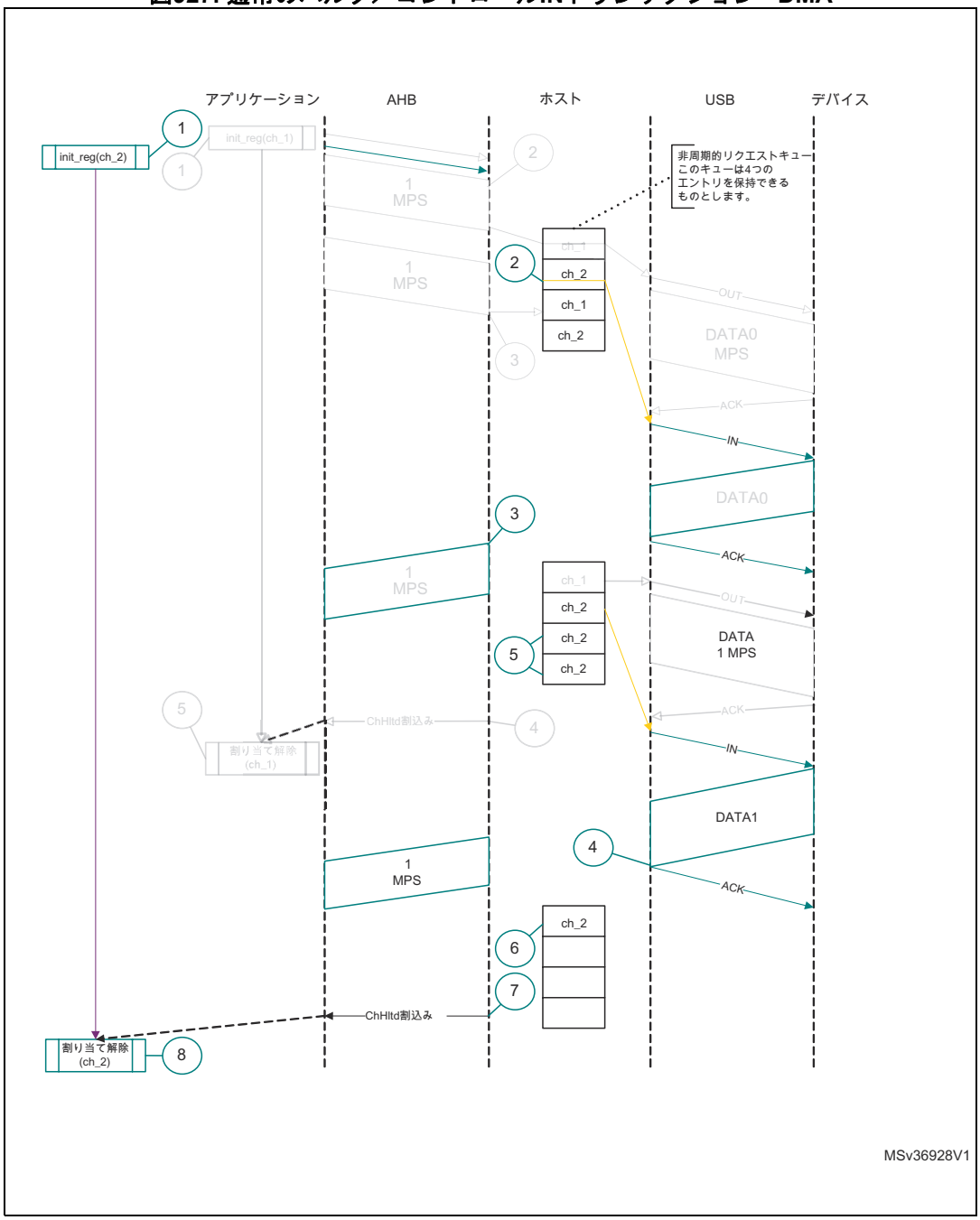
   ホスト機能がNAKまたはNYETを受信した場合、コアは個別の割込みを生成しません。

- **バルクおよびコントロールINトランザクション - DMAモード**

  動作シーケンスは、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、使用するチャネル（チャネルx）を初期化および有効化します。
2. チャネルがアービタから許可を受信すると直ちに、OTG_HS ホストはINリクエストをリクエストキューへ書き込みます（調停はラウンドロビン方式で行われます）。
3. OTG_HS ホストはエラーなしで最終バイトを受信すると、直ちにシステムメモリへの受信データの書込みを開始します。
4. OTG_HS ホストは最終パケットを受信すると、内部フラグをセットしてリクエストキューから余分なINリクエストを削除します。
5. OTG_HS ホストは余分なリクエストを排出します。
6. チャネルxを無効化する最終リクエストがリクエストキューに書き込まれます。この時点で、チャンネル2は以降の調停に対して内部でマスクされます。
7. OTG_HS ホストは、ディセーブルリクエストがキューのトップに来ると直ちにCHH割込みを生成します。
8. CHH割込みに応答して、他の転送のためにチャネル割り当てを解除します。

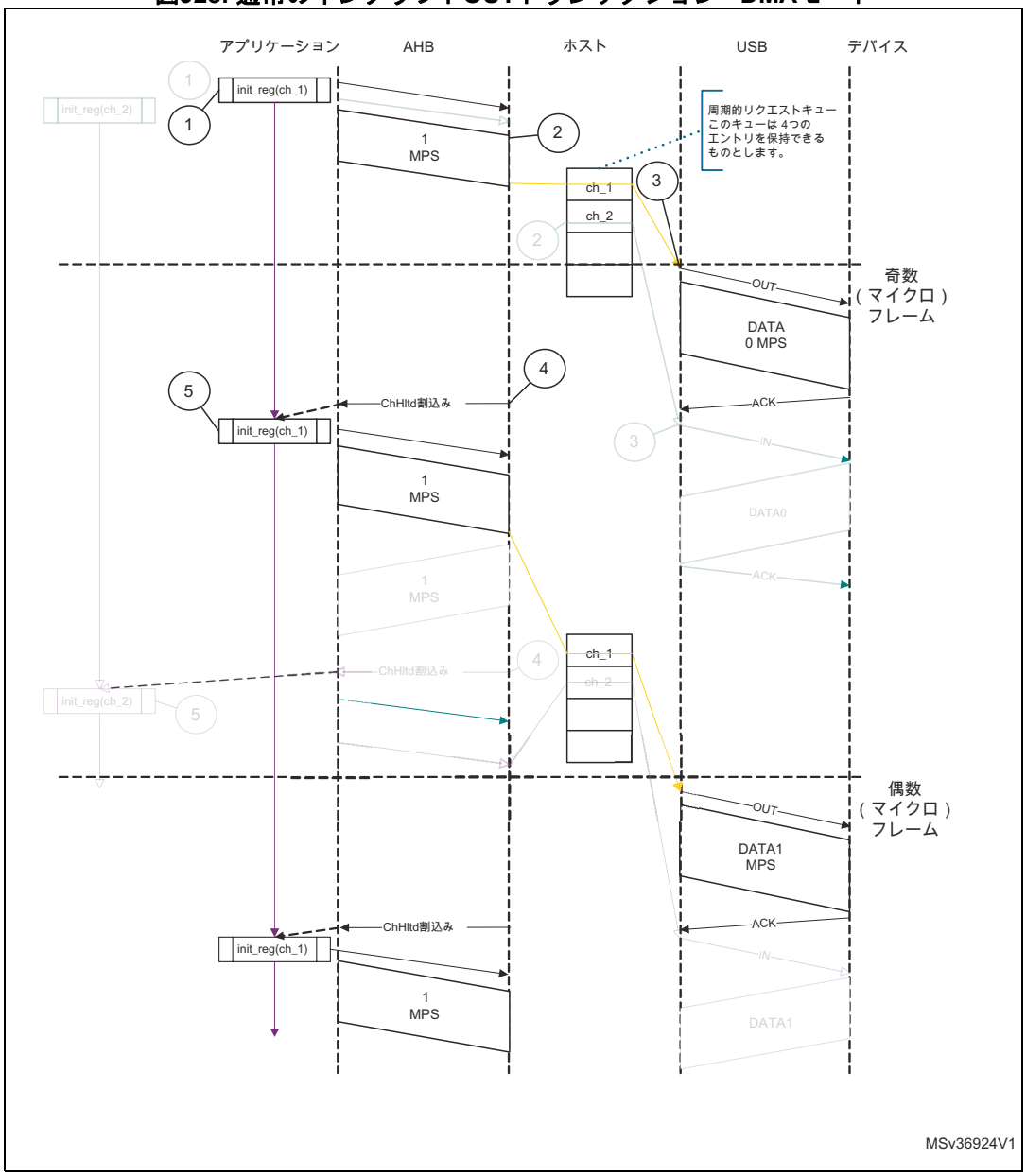
図927. 通常のバルク／コントロールINトランザクション - DMA



- **インタラプトOUTトランザクション - DMAモード**

1. セクション ：チャネルの初期化で説明しているように、チャネルxを初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始しOUTリクエストを最後の32-bitワードフェッチと一緒に書き込みます。広帯域転送では OTG_HS ホストは、次のパケットのフェッチを（MCフィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始でOUTトークンの送信を試みます。
4. パケットの正常送信の後、OTG_HS ホストはCHH割込みを生成します。
5. CHH割込みに応答して、次の転送のためにチャネルを再初期化します。

図928. 通常のインタラプトOUTトランザクション - DMAモード

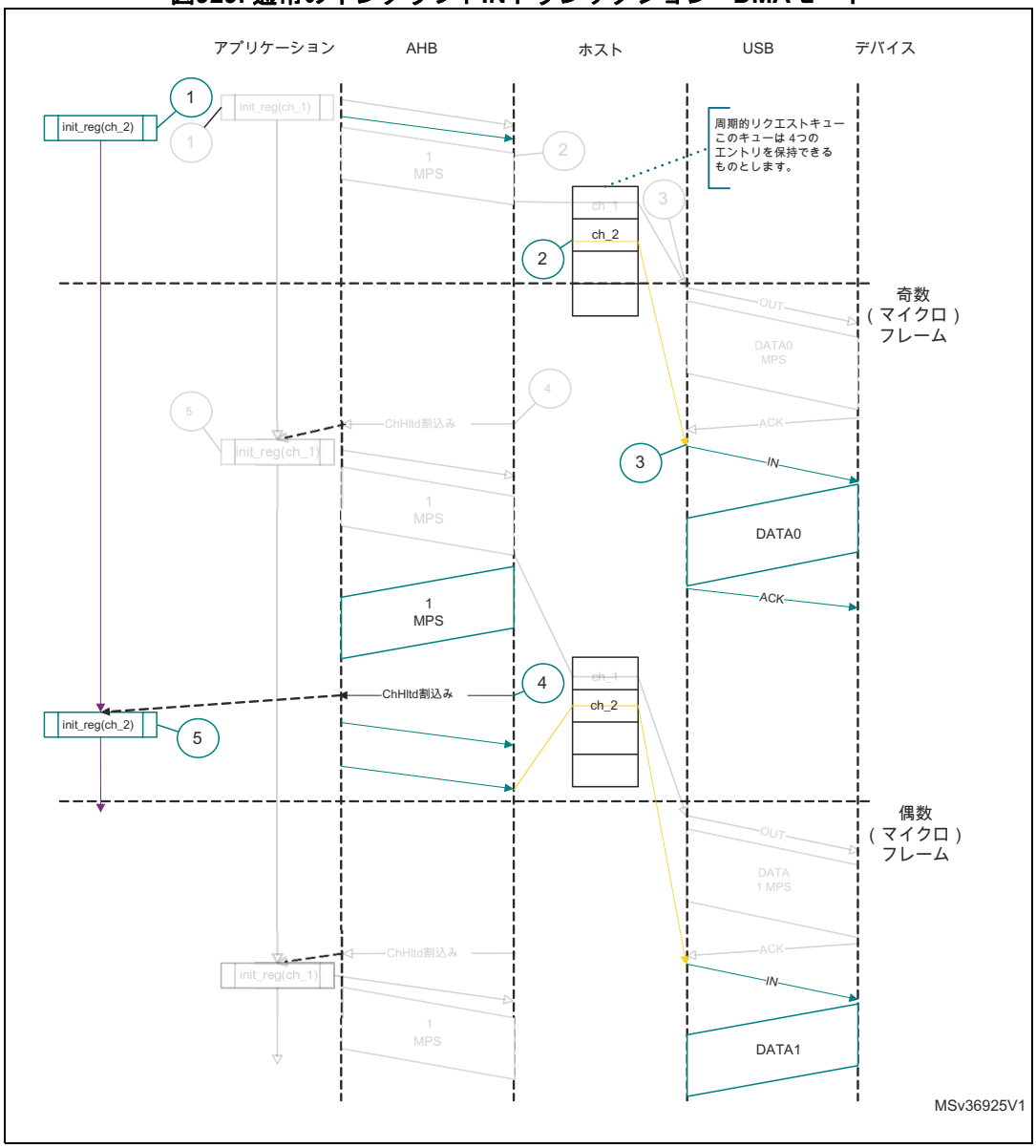


- **インタラプトINトランザクション - DMAモード**

  動作シーケンス （チャンネルx）は次の通りです。

1. セクション ：チャネルの初期化で説明しているように、チャネルxを初期化し、有効にします。
2. チャネルxがアービタから許可を受け取ると直ちに、OTG_HS ホストはINリクエストをリクエストキューへ書き込みます （公平なラウンドロビン方式を使用）。広帯域転送では、OTG_HS ホストはMC回までの連続書込みを行います。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始でINトークンの送信を試みます。
4. パケットが受信され、受信FIFOに書き込まれると、OTG_HS ホストはCHH割込みを生成します。
5. CHH割込みに応答して、次の転送のためにチャネルを再初期化します。

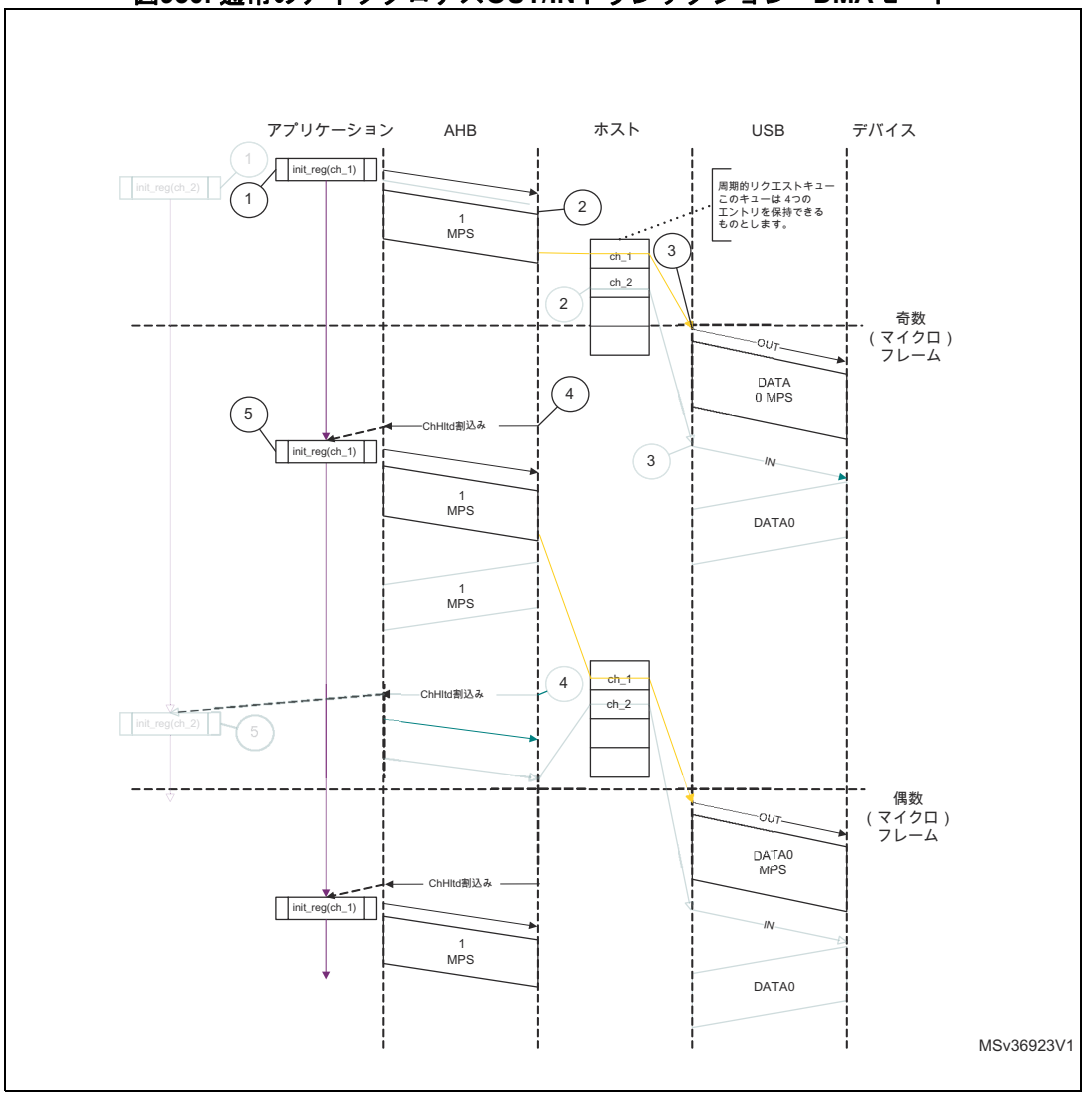
図929. 通常のインタラプトINトランザクション - DMAモード



- **アイソクロナスOUTトランザクション - DMAモード**

1. セクション ：チャネルの初期化で説明しているように、チャネルxを初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始しOUTリクエストを最後の32-bitワードフェッチと一緒に書き込みます。広帯域転送では OTG_HS ホストは、次のパケットのフェッチを（MCフィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始でOUTトークンの送信を試みます。
4. パケットの正常送信の後、OTG_HS ホストはCHH割込みを生成します。
5. CHH割込みに応答して、次の転送のためにチャネルを再初期化します。

図930. 通常のアイソクロナスOUT/INトランザクション - DMAモード

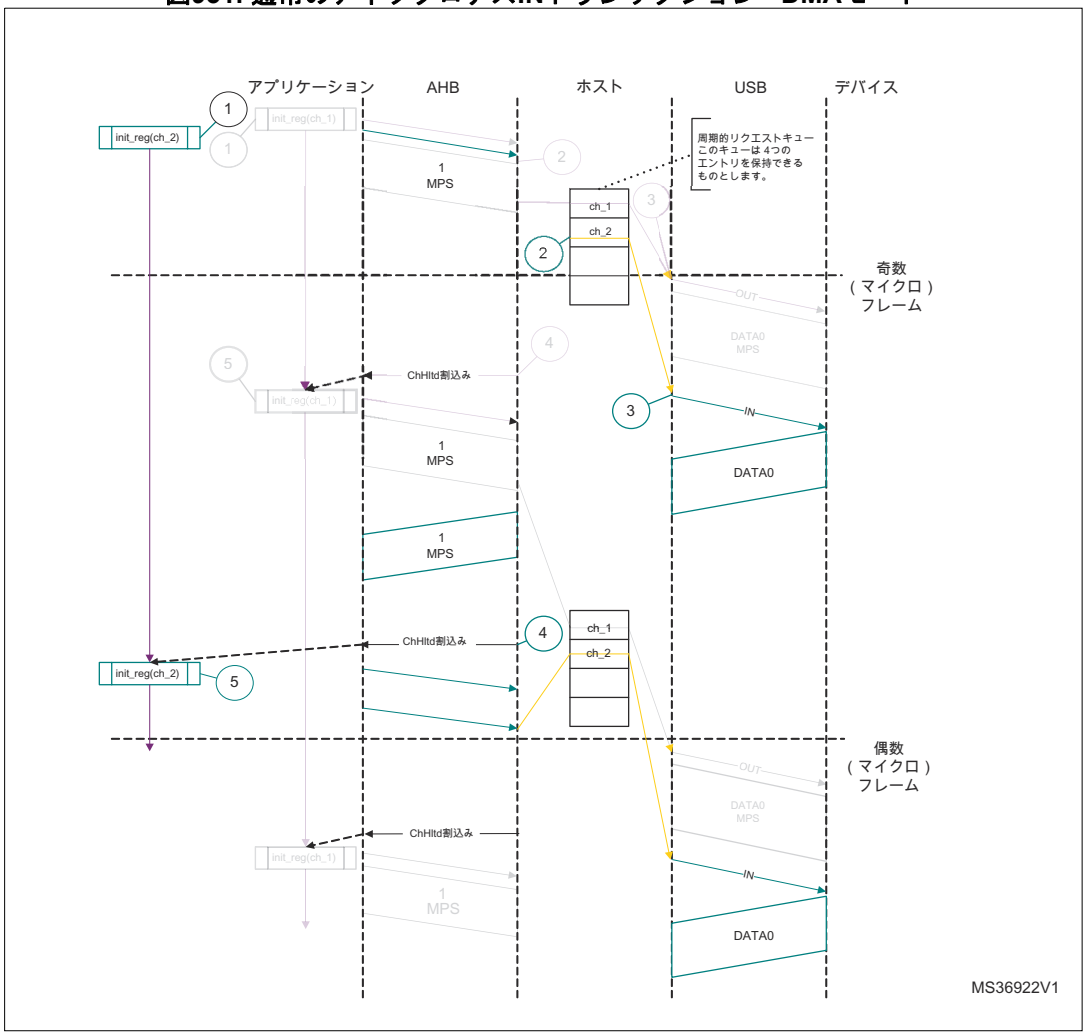


- **アイソクロナスINトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネルxを初期化し、有効にします。
2. チャネルxがアービタから許可を受け取ると直ちに、OTG_HS ホストはINリクエストをリクエストキューへ書き込みます （公平なラウンドロビン方式を使用）。広帯域転送では、OTG_HS ホストはMC回までの連続書込みを行います。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始でINトークンの送信を試みます。
4. パケットが受信され、受信FIFOに書き込まれると、OTG_HS ホストはCHH割込みを生成します。
5. CHH割込みに応答して、次の転送のためにチャネルを再初期化します。

図931. 通常のアイソクロナスINトランザクション - DMAモード



- **バルクおよびインタラプトOUT／SETUPスプリットトランザクション - DMAモード**

動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネルxを初期化および有効化します。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始しOUTリクエストを最後の32-bitワードフェッチと一緒に書き込みます。
3. スタートスプリットの正常送信の後、OTG_HS ホストはCHH割込みを生成します。
4. CHH割込みに応答してOTG_HCSPLT1のCOMPLSPLTビットをセットし、スプリット全体を送信します。
5. スプリット全体の正常送信の後、OTG_HS ホストはCHH割込みを生成します。
6. CHH割込みに応答して、チャネルの割り当てを解除します。

- **バルク／コントロールINスプリットトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネルxを初期化し、有効にします。
2. アービタから許可を得た後、OTG_HS ホストはスタートスプリットリクエストを非周期的リクエストへ書き込みます。OTG_HS ホストは、リクエストを書き込んだ後に、調停のために内部でチャネルxをマスクします。
3. INトークンが送信されると直ちに OTG_HS ホストはCHH割込みを生成します。
4. CHH割込みに応答してOTG_HCSPLT2のCOMPLSPLTビットをセットし、スプリットトークン全体を送信するためにチャネルを再度有効化します。これにより、調停用にチャンネルxのマスクが解除されます。
5. アービタから許可を得た後、OTG_HS ホストはスプリット全体リクエストを非周期的リクエストへ書き込みます。
6. OTG_HS ホストはパケットを正常に受信した後、パケットのシステムメモリへの書込みを開始します。
7. 受信パケットをシステムメモリへ書き込むと直ちに、OTG_HS ホストはCHH割込みを生成します。
8. CHH割込みに応答して、チャネルの割り当てを解除します。

- **インタラプトOUTスプリットトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネル1を初期化および有効化します。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。
2. OTG_HS ホストはパケットの読出しを開始します。
3. OTG_HS ホストは、スタートスプリットトランザクションの送信を試みます。
4. スタートスプリットの正常送信の後、OTG_HS ホストはCHH割込みを生成します。
5. CHH割込みに応答してOTG_HCSPLT1のCOMPLSPLTビットをセットし、スプリット全体を送信します。
6. スプリット全体トランザクションの正常送信の後､OTG_HS ホストはCHH割込みを生成します。
7. CHH割込みに応答して、チャネルの割り当てを解除します。

- **インタラプトINスプリットトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネルxを初期化および有効化します。
2. チャネルxがアービタから許可を受け取ると直ちに OTG_HS ホストはINリクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリットINトークンの送信を試みます。
4. OTG_HS ホストはスタートスプリットINトークンを正常に送信した後にCHH割込みを生成します。
5. CHH割込みに応答してOTG_HCSPLT2のCOMPLSPLTビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書込みを開始します。
7. OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH割込みを生成します。
8. CHH 割込みに応答して、次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

- **アイソクロナスOUTスプリットトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット（開始）用にチャネルxを初期化および有効化します。アプリケーションはOTG_HCCHAR1レジスタのODDFRMビットをセットする必要があります。MPSフィールドをプログラムします。
2. OTG_HS ホストはパケットの読出しを開始します。
3. スタートスプリット（開始）の正常送信の後、OTG_HS ホストはCHH割込みを生成します。
4. CHH 割込みに応答して、スタートスプリット（終わり）を送信するためレジスタを再初期化します。
5. スタートスプリット（終わり）の正常送信の後、OTG_HS ホストはCHH割込みを生成します。
6. CHH割込みに応答して、チャネルの割り当てを解除します。

- **アイソクロナスINスプリットトランザクション - DMAモード**

  動作シーケンス（チャネルx）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネルxを初期化および有効化します。
2. チャネルxがアービタから許可を受け取ると直ちに OTG_HS ホストはINリクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリットINトークンの送信を試みます。
4. OTG_HS ホストはスタートスプリットINトークンを正常に送信した後にCHH割込みを生成します。
5. CHH割込みに応答してOTG_HCSPLT2のCOMPLSPLTビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書込みを開始します。

   OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH割込みを生成します。CHH割込みに応答して､次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

73.15.6 デバイスプログラミングモデル

USBリセット時のエンドポイントの初期化

1. すべてのOUTエンドポイントのNAKビットをセットします。
   - OTG_DOEPCTLxのSNAK = 1（すべてのOUTエンドポイントについて）
2. 以下の割込みビットのマスクを解除します。
   - OTG_DAINTMSKのINEP0 = 1（コントロール0 INエンドポイント）
   - OTG_DAINTMSKのOUTEP0 = 1（コントロール0 OUTエンドポイント）
   - OTG_DOEPMSKのSTUPM = 1
   - OTG_DOEPMSKのXFRCM = 1
   - OTG_DIEPMSKのXFRCM = 1
   - OTG_DIEPMSKのTOM = 1

3. 各FIFOのデータFIFO RAMをセットアップします。
   – OTG_GRXFSIZレジスタをプログラムして、コントロールOUTデータとセットアップデータを受信できるようにします。閾値の設定が無効の場合、これは、少なくとも、コントロールエンドポイント0の最大パケットサイズ1個分 + 2ワード（コントロールOUTデータパケットのステータス用）+ 10ワード（セットアップパケット用）でなければなりません。
   – OTG_DIEPTXF0レジスタを、コントロールINデータを送信できるようにプログラムします（選択されたFIFO番号に応じて）。これは、少なくとも、コントロールエンドポイント0の最大パケットサイズ1個分以上でなければなりません。
4. SETUPパケットを受信するために、コントロールOUTエンドポイント0のエンドポイント固有レジスタの以下のフィールドをプログラムします。
   – OTG_DOEPTSIZ0のSTUPCNT = 3（最大3つの連続SETUPパケットを受信するため）
5. DMAモードのUSB OTG_HS では、OTG_DOEPDMA0レジスタは受信した全SETUPパケットを保存する有効なメモリアドレスを持つ必要があります。

この時点でSETUPパケットを受信するために必要なすべての初期化が終了したことになります。

エニュメレーション完了時のエンドポイント初期化

1. エニュメレーション終了割込み（OTG_GINTSTSのENUMDNEビット）時にはOTG_DSTSレジスタを読み出して、エニュメレーションスピードを決めます。
2. OTG_DIEPCTL0のMPSIZフィールドをプログラムして、最大パケットサイズを設定します。このステップでは、コントロールエンドポイント0を設定します。コントロールエンドポイントの最大パケットサイズは、エニュメレーションスピードに依存します。
3. DMAモードのUSB OTG_HS では、コントロールOUTエンドポイント0を有効化するOTG_DOEPCTL0レジスタをプログラムして、SETUPパケットを受信するようにします。

この時点で、デバイスはSOFパケットを受信する準備ができ、コントロールエンドポイント0でコントロール転送を行うように設定されたことになります。

SetAddressコマンド受信時のエンドポイントの初期化

このセクションではSETUPパケットでSetAddressコマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. OTG_DCFGレジスタにSetAddressコマンドで受信したデバイスアドレスをプログラムします。
2. ステータスINパケットを送信するように、コアをプログラムします。

SetConfiguration/SetInterfaceコマンド受信時のエンドポイントの初期化

このセクションではSETUPパケットでSetConfigurationまたはSetInterfaceコマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. SetConfigurationコマンドを受信したとき、アプリケーションは、新しい設定で有効なエンドポイントの特性で、エンドポイントのレジスタをプログラムする必要があります。
2. SetInterfaceコマンドを受信したとき、アプリケーションは、このコマンドの影響を受けるエンドポイントのレジスタをプログラムする必要があります。
3. 前の設定または代替設定ではアクティブであったエンドポイントが、新しい設定または代替設定では無効なことがあります。これらの無効なエンドポイントは、機能を停止させる必要があります。
4. アクティブな各エンドポイントの割込みのマスクを解除し、OTG_DAINTMSKレジスタですべての非アクティブなエンドポイントに対する割込みをマスクします。
5. 各FIFOのデータFIFO RAMをセットアップします。
6. 必要なすべてのエンドポイントを設定した後、アプリケーションは、ステータスINパケットを送信するようにコアをプログラムする必要があります。

この時点で、デバイスのコアは、任意のタイプのデータパケットを送信／受信できるように設定されたことになります。

エンドポイントのアクティブ化

このセクションでは、デバイスエンドポイントをアクティブ化するか、既存のデバイスエンドポイントを新しいタイプに設定するために必要な手順について説明します。

1. 必要なエンドポイントの特性をOTG_DIEPCTLxレジスタ（INまたは双方向エンドポイントの場合）、またはOTG_DOEPCTLxレジスタ（OUTまたは双方向エンドポイントの場合）の以下のフィールドにプログラムします。
   - 最大パケットサイズ
   - USBアクティブエンドポイント = 1
   - エンドポイント開始データトグル（インタラプトおよびバルクエンドポイントの場合）
   - エンドポイントタイプ
   - Tx FIFO番号
2. エンドポイントがアクティブ化されると、コアは、そのエンドポイントに宛てたトークンのデコードを開始し､そのエンドポイントで受信された有効な各トークンについて有効なハンドシェイクを送出します。

エンドポイントの機能停止

このセクションでは、既存のエンドポイントの機能を停止させるために必要な手順について説明します。

1. 非アクティブにすべきエンドポイントでは、OTG_DIEPCTLxレジスタ（INまたは双方向エンドポイントの場合）､またはOTG_DOEPCTLxレジスタ（OUTまたは双方向エンドポイントの場合）のUSBのアクティブエンドポイントビットをクリアします。
2. エンドポイントが非アクティブな状態になると､コアは､そのエンドポイントにアドレス指定されたトークンを無視するのでUSB上でタイムアウトが発生します。

注： **アプリケーションは、以下の条件を満たすように、デバイスコアがトラフィックを処理できるようにセットアップする必要があります。**
OTG_GINTMSKレジスタのNPTXFEMおよびRXFLVLMをクリアする必要があります。

動作モデル

SETUPおよびOUTデータ転送

このセクションでは、データOUT転送およびSETUPトランザクション時の内部データフローとアプリケーションレベルの動作について説明します。

- **パケットの読出し**

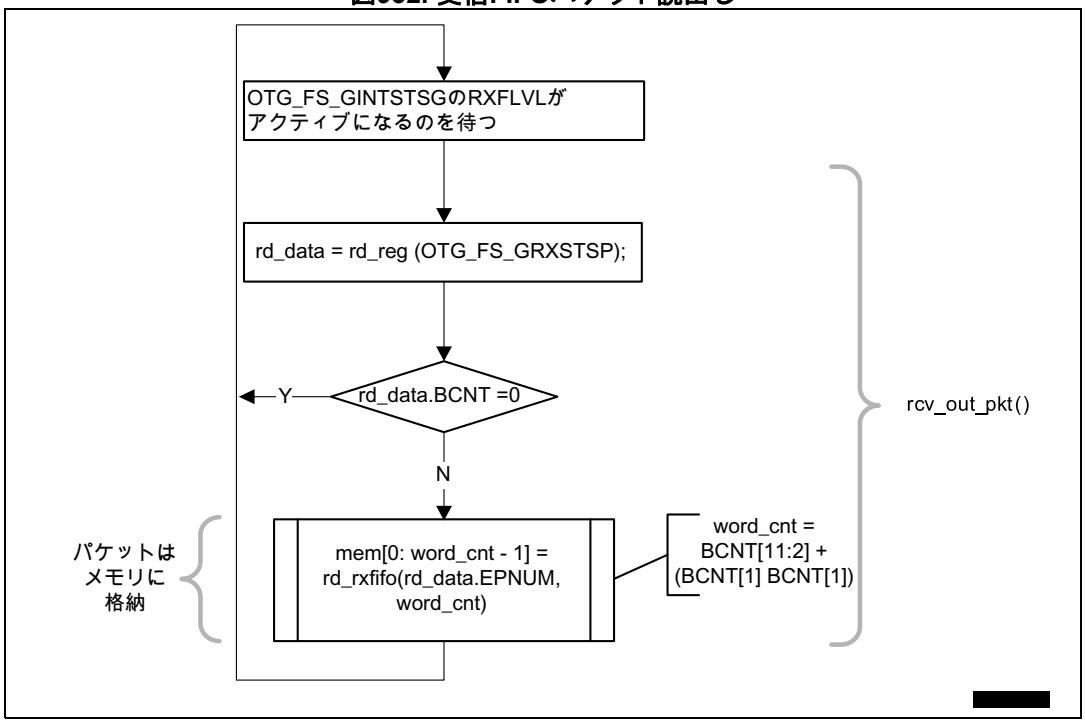
このセクションでは、受信FIFOからパケット（OUTデータおよびSETUPパケット）を読み出す方法について説明します。

1. RXFLVL割込み（OTG_GINTSTSレジスタ）を受信すると、アプリケーションは、受信ステータスポップレジスタ（OTG_GRXSTSP）を読み出す必要があります。
2. アプリケーションは、RXFLVL = 0（OTG_GINTMSKレジスタ）を書き込むことによって、受信FIFOからそのパケットが読み出されるまで、RXFLVL割込み（OTG_GINTSTSレジスタ）をマスクすることができます。
3. 受信パケットのバイトカウントが0でない場合､データのバイトカウントは受信データFIFOからポップされ、メモリに格納されます。受信パケットのバイトカウントが0の場合、受信データFIFOからデータはポップされません。

4. FIFOのパケット受信ステータスの読出し結果は、以下のいずれかを示します。
   a) グローバルOUT NAKのパターン
   PKTSTS = グローバルOUT NAK、BCNT = 0x000、EPNUM = （0x0）、DPID = （0b00）。
   これらのデータは、グローバルOUT NAKビットが有効になっていることを示します。
   b) SETUPパケットのパターン
   PKTSTS = SETUP、BCNT = 0x008、EPNUM = コントロールEP番号、DPID = DATA0。
   これらのデータは、指定されたエンドポイントのSETUPパケットを受信FIFOから読み出せることを示します。
   c) セットアップステージ終了パターン
   PKTSTS = セットアップステージ終了、BCNT = 0x0、EPNUM = コントロールEP番号、DPID = （0b00）。
   これらのデータは、指定されたエンドポイントのセットアップステージが完了し、データステージが開始されたことを示します。このエントリが受信FIFOからポップされた後、コアは、指定されたコントロールOUTエンドポイントでセットアップ割込みをアサートします。
   d) データOUTパケットのパターン
   PKTSTS = データOUT、BCNT = 受信データOUTパケットのサイズ（0 ≤ BCNT ≤ 1 024）、EPNUM = パケットを受信したEPNUM、DPID = 実際のデータPID。
   e) データ転送完了パターン
   PKTSTS = データOUT転送終了、BCNT = 0x0、EPNUM = データ転送が完了したOUT EP番号、DPID = （0b00）。
   これらのデータは、指定されたOUTエンドポイントのOUTデータ転送が完了したことを示します。このエントリが受信FIFOからポップされた後、コアは、指定されたOUTエンドポイントで転送完了割込みをアサートします。
5. データペイロードが受信FIFOからポップされた後、RXFLVL割込み（OTG_GINTSTS）がマスク解除されなければなりません。
6. 手順1～5は、アプリケーションがOTG_GINTSTSレジスタのRXFLVLによる割込みラインのアサーションを検出するたびに繰り返されます。空の受信FIFOを読み出すと、定義されていないコア動作を引き起こすことがあります。

図932上記の手順のフローチャートです。

図932. 受信FIFOパケット読出し



SETUPトランザクション

このセクションでは、コアがSETUPパケットを処理する方法と、SETUPトランザクションを処理するアプリケーションのシーケンスについて説明します。

- **アプリケーションの要件**

1. SETUP パケットを受信するには、コントロール OUT エンドポイントの STUPCNT フィールド（OTG_DOEPTSIZxレジスタ）が、ゼロでない値にプログラムされなければなりません。アプリケーションがSTUPCNTフィールドをゼロでない値にプログラムすると、NAKのステータスとOTG_DOEPCTLxレジスタのEPENAビットの設定に関係なく、コアは、SETUPパケットを受信して、受信FIFOに書き込みます。STUPCNTフィールドは、コントロールエンドポイントがSETUPパケットを受信するたびにデクリメントされます。SETUPパケットを受信する前に、STUPCNTフィールドが適切な値にプログラムされていなかった場合、コアはSETUPパケットを受信し、STUPCNTフィールドをデクリメントしますが、アプリケーションはコントロール転送のセットアップステージで受信したSETUPパケットの正しい数を判定できないことがあります。
   – OTG_DOEPTSIZxレジスタのSTUPCNT = 3
2. コントロールエンドポイントで3つまでのSETUPパケットを受信するためには、アプリケーションは、常に受信データFIFOに余分のスペースを割り当てておく必要があります。
   – 確保すべきスペースは10ワード分です。最初のSETUPパケット用に3ワードが必要であり、セットアップステージ終了ワードのために1ワード、すべてのコントロールエンドポイントにわたって2つの余分なSETUPパケットを格納するために6ワードが必要です。
   – 8バイトのSETUPデータと4バイトのSETUPステータス（セットアップパケットパターン）を格納するには、SETUPパケットあたり3ワードが必要です。コアは受信データFIFOのこのスペースをSETUPデータを書き込むためにのみ確保し、このスペースをデータパケットのために使うことはありません。
3. アプリケーションは、受信FIFOからSETUPパケットの2ワードを読み出す必要があります。
4. アプリケーションは受信FIFOからセットアップステージ終了ワードを読み出して、破棄する必要があります。

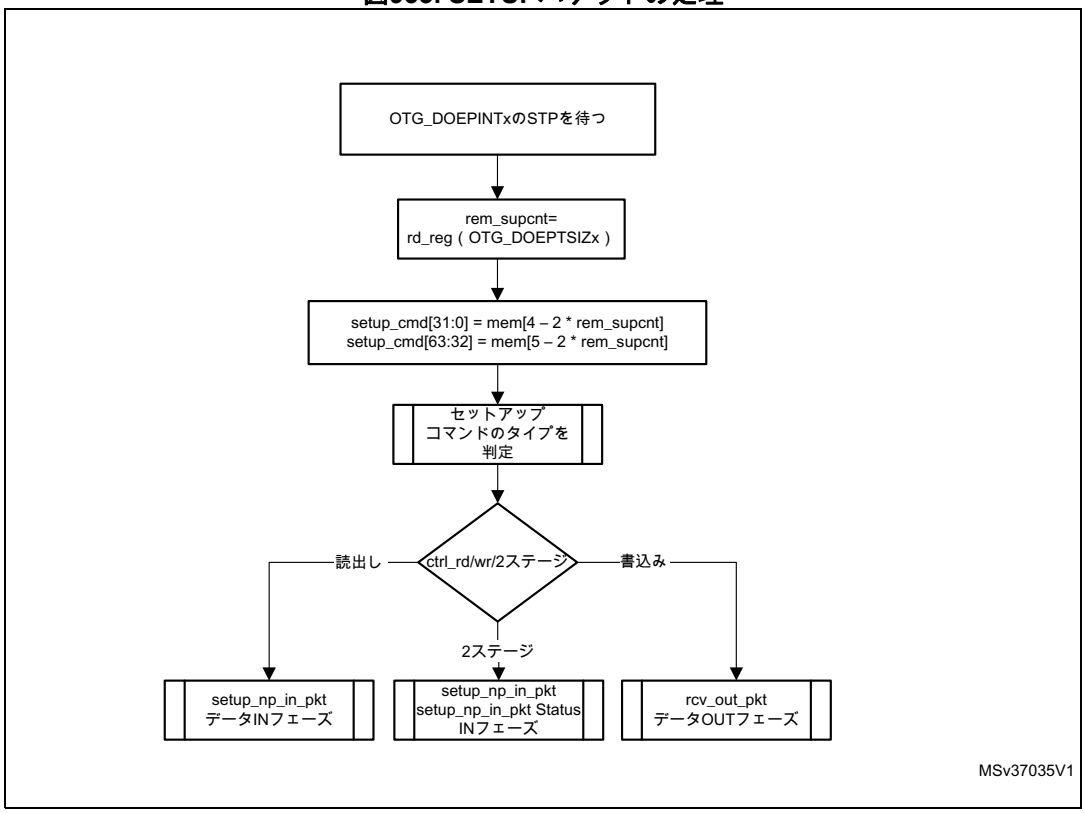
- **内部データフロー**

1. SETUPパケットが受信されると、コアは受信FIFO内の空きスペースをチェックせずに、また、エンドポイントのNAKおよびSTALLビットの設定に関係なく、受信したデータを受信FIFOに書き込みます。
   – コアは、SETUPパケットが受信されたコントロールIN/OUTエンドポイントのIN NAKおよびOUT NAKビットを内部でセットします。
2. USB上で受信された各SETUPパケットについて、3ワードのデータが受信FIFOに書き込まれ、STUPCNTフィールドが1ずつデクリメントされます。
   – 最初のワードは、コアが内部的に使用する制御情報を含みます。
   – 2番目のワードは、SETUPコマンドの最初の4バイトを含みます。
   – 3番目のワードは、SETUPコマンドの最後の4バイトを含みます。
3. セットアップステージがデータIN/OUTステージに変わると、コアは、エントリ（セットアップステージ終了ワード）を受信FIFOに書き込んで、セットアップステージの完了を示します。
4. AHB側では、SETUPパケットはアプリケーションによってエンプティにされます。
5. アプリケーションが受信 FIFO からセットアップステージ終了ワードをポップすると、コアは、SETUP割込み（OTG_DOEPINTx）でアプリケーションに割り込んで、受信したSETUPパケットを処理できることを示します。
6. コアは、コントロールOUTエンドポイントのエンドポイントイネーブルビットをクリアします。

- **アプリケーションのプログラミングシーケンス**：

1. OTG_DOEPTSIZxレジスタをプログラムします。
   – STUPCNT = 3
2. RXFLVL割込み（OTG_GINTSTS）を待ち、受信FIFOからデータパケットをエンプティにします。
3. STUP割込み（OTG_DOEPINTx）のアサートは、SETUPデータ転送が正常に完了したことを示します。
   – この割込み時、アプリケーションは、OTG_DOEPTSIZx レジスタを読み出して、受信したSETUPパケットの数を確認し、最後に受信したSETUPパケットを処理する必要があります。

図933. SETUPパケットの処理



- **3つを超える連続SETUPパケットの処理**

USB 2.0仕様に従い、通常SETUPパケットエラー時にはホストは同じエンドポイントに対して3つを超える連続SETUPパケットを送信しません。ただしUSB 2.0仕様では、ホストが同じエンドポイントに送信できる連続SETUPパケットの数を制限してはいません。この条件が発生すると、OTG_HS コントローラは、割込み（OTG_DOEPINTxのB2BSTUP）を生成します。

- **グローバルOUT NAKの設定**

内部データフロー：

1. アプリケーションがグローバルOUT NAK（OTG_DCTLのSGONAKビット）をセットすると、コアはSETUPパケットを除き、受信FIFOへのデータの書込みを停止します。受信FIFOの使用可能なスペースの有無に関係なく、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信し、コアはアイソクロナスOUTデータパケットを無視します。
2. コアは、グローバルOUT　NAKパターンを受信FIFOに書き込みます。アプリケーションがこのデータパターンを書き込むには、受信FIFOに十分なスペースを確保する必要があります。
3. アプリケーションがグローバルOUT　NAKパターンワードを受信FIFOからポップすると、コアは、GONAKEFF割込み（OTG_GINTSTS）をセットします。
4. この割込みを検出したアプリケーションは、コアがグローバルOUT NAKモードにあるとみなすことができます。アプリケーションはOTG_DCTLのSGONAKビットをクリアすることによって、この割込みをクリアできます。

アプリケーションのプログラミングシーケンス：

1. 受信FIFOでのデータの受信も停止するには、アプリケーションは、以下のフィールドをプログラムすることによって、グローバルOUT NAKビットをセットする必要があります。
   – OTG_DCTLレジスタのSGONAK = 1
2. OTG_GINTSTSのGONAKEFF割込みのアサートを待ちます。アサートされた場合、この割込みは、コアがSETUPパケット以外のあらゆるタイプのデータの受信を停止したことを示します。
3. アプリケーションはOTG_DCTLのSGONAKビットをセットした後、コアがGONAKEFF割込み（OTG_GINTSTS）をアサートする前に、有効なOUTパケットを受信することができます。
4. アプリケーションは、OTG_GINTMSK レジスタの GONAKEFFM ビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_GINTMSKレジスタのGONAKEFFM = 0
5. アプリケーションがグローバルOUT　NAKモードを終了する準備ができたときにはOTG_DCTLレジスタのSGONAKビットをクリアする必要があります。GONAKEFF割込み（OTG_GINTSTS）もクリアされます。
   – OTG_DCTLレジスタのCGONAK = 1
6. アプリケーションがこの割込みをマスクするのが早すぎた場合は、以下のようにしてマスク解除を行う必要があります。
   – OTG_GINTMSKレジスタのGONAKEFFM = 1

- **OUTエンドポイントの無効化**

アプリケーションが有効にしたOUTエンドポイントを無効にするには、このシーケンスを使用する必要があります。

アプリケーションのプログラミングシーケンス：

1. OUTエンドポイントを無効にする前に、アプリケーションは、コアのグローバルOUT NAKモードを有効にする必要があります。
   – OTG_DCTLレジスタのSGONAK = 1
2. OTG_GINTSTSのGONAKEFF割込みを待ちます。
3. 以下のフィールドをプログラムすることによって、必要なOUTエンドポイントを無効にします。
   – OTG_DOEPCTLxレジスタのEPDIS = 1
   – OTG_DOEPCTLxレジスタのSNAK = 1
4. EPDISD割込み（OTG_DOEPINTx）を待ちます。これはOUTエンドポイントが完全に無効にされたことを示します。EPDISD割込みがアサートされると、コアは、以下のビットもクリアします。
   – OTG_DOEPCTLxレジスタのEPDIS = 0
   – OTG_DOEPCTLxレジスタのEPENA = 0
5. アプリケーションは、無効にされていない他のエンドポイントからのデータ受信を開始するために、グローバルOUT NAKビットをクリアする必要があります。
   – OTG_DCTLレジスタのSGONAK = 0

- **OUTエンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング·シーケンスを使用する必要があります。

動作シーケンス：

1. 以下をセットして、すべてのOUTエンドポイントを有効にします。
   – すべてのOTG_DOEPCTLxレジスタのEPENA = 1
2. 次のようにRxFIFOを排出します。
   – 1になるまでOTG_GRSTCTL.AHBIDLをポーリングします。これは、AHBマスタがアイドルであることを示します。
   – OTG_GRSTCTL.RXFFLSH = 1で読出し、変更、書込み操作を実行します。
   – 0になるまでOTG_GRSTCTL.RXFFLSHをポーリングしますが、（最小リセットシグナリング時間に応じて）10ミリ秒未満のタイムアウトも使用します。タイムアウト前に0が確認された場合、RxFIFO排出は完了です。その時点でタイムアウトが発生した場合、まだ1が残っており（ホストから来るEP0のパケットによる場合がある）、（1回だけ）前の手順（読出し変更、書込み動作の実行）に戻ります。
3. OUTエンドポイントを無効にする前に、アプリケーションは3401ページのグローバルOUT NAKの設定の指示に従ってコアのグローバルOUT NAKモードを有効にする必要があります。これによって、RxFIFO　のデータがアプリケーションに正常に送信されたことを確認します。OTG_DCTLでSGONAK = 1をセットします。
4. OTG_GINTSTSのGONAKEFF割込みを待ちます。
5. 以下のレジスタビットをプログラムすることによって、すべてのアクティブなOUTエンドポイントを無効にします。
   – OTG_DOEPCTLxレジスタのEPDIS = 1
   – OTG_DOEPCTLxレジスタのSNAK = 1
6. 前の手順でプログラムされた各OUTエンドポイントについてOTG_DOEPINTxでEPDIS割込みを待ちます。OTG_DOEPINTxでのEPDIS割込みは対応するOUTエンドポイントが完全に無効にされたことを示します。EPDIS割込みがアサートされると、以下のビットがクリアされます。
   – OTG_DOEPCTLxレジスタのEPENA = 0
   – OTG_DOEPCTLxレジスタのEPDIS = 0
   – OTG_DOEPCTLxレジスタのSNAK = 0

- **一般の非アイソクロナスOUTデータ転送**

このセクションでは、通常の非アイソクロナスOUTデータ転送（コントロール、バルク、またはインタラプト）について説明します。

アプリケーションの要件：

1. OUT転送をセットアップする前に、アプリケーションはOUT転送の一部として受信されるすべてのデータを収容できるバッファをメモリ内で割り当てる必要があります。
2. OUT 転送の場合、エンドポイントの転送サイズレジスタの転送サイズフィールドは、エンドポイントの最大パケットサイズの倍数でなければならず、ワードの境界に揃えられていなければなりません。
   – 転送サイズ [EPNUM] = *n* * (MPSIZ[EPNUM] + 4 – (MPSIZ[EPNUM] mod 4))
   – パケットカウント [EPNUM] = *n*
   – *n* > 0

3. OUT エンドポイント割込み時には、アプリケーションは、エンドポイントの転送サイズレジスタを読み出して、メモリ内のペイロードのサイズを計算しなければなりません。受信したペイロードのサイズが、プログラムされた転送サイズより小さいこともあります。
   – メモリ内のペイロードのサイズ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – このペイロードが受信されたUSBパケットの数 = アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

内部データフロー：

1. アプリケーションがデータを受信するためには、エンドポイント固有レジスタの転送サイズおよびパケットカウントのフィールドを設定し、NAKビットをクリアし、エンドポイントを有効にする必要があります。
2. NAKビットがクリアされると、コアは、データの受信を開始し、受信FIFOにスペースがある限り、データを受信FIFOに書き込みます。USBで受信された各データパケットについて、データパケットとそのステータスが受信FIFOに書き込まれます。受信FIFOにパケット（最大パケットサイズまたはショートパケット）が書き込まれるたびに、そのエンドポイントのパケットカウントフィールドが1ずつデクリメントされます。
   – 受信されたOUTデータパケットのデータCRCが不良な場合、受信FIFOから自動的に一掃されます。
   – USB上のパケットに対してACKを送信した後、コアはACKを検出できないホストが再送信する非アイソクロナスOUTデータパケットを破棄します。アプリケーションは、同じエンドポイント上では、同じデータPIDを持つ複数の連続データOUTパケットを検出しません。この場合、パケットカウントはデクリメントされません。
   – 受信FIFOにスペースがない場合、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信FIFOには書き込まれません。さらに、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信します。
   – 上記の3つのケースのすべてにおいて、データは受信FIFOに書き込まれないので、パケットカウントはデクリメントされません。
3. パケットカウントが0になるか、エンドポイント上でショートパケットが受信されると、そのエンドポイントのNAKビットがセットされます。NAKビットがセットされると、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信FIFOには書き込まれず、非アイソクロナスOUTトークンはNAKハンドシェイク応答を受信します。
4. データが受信FIFOに書き込まれた後、アプリケーションは受信FIFOからデータを読み出して、エンドポイントあたり一度に1パケットずつ外部メモリに書き込みます。
5. AHB 上で外部メモリへのパケットの書込みが終わるたびに、書き込まれたパケットのサイズだけエンドポイントの転送サイズがデクリメントされます。
6. 以下の条件の1つでOUTエンドポイントのOUTデータ転送完了パターンが受信FIFOに書き込まれます。
   – 転送サイズが0、およびパケットカウントが0。
   – 受信FIFOに書き込まれた最後のOUTデータパケットがショートパケット。（0 ≤ パケットサイズ < 最大パケットサイズ）
7. アプリケーションがこのエントリ（OUT データ転送完了）をポップすると、エンドポイントの転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZxレジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx　レジスタをエンドポイントの特性でプログラムし、EPENA　ビットおよびCNAKビットをセットします。
   – OTG_DOEPCTLxレジスタのEPENA = 1
   – OTG_DOEPCTLxレジスタのCNAK = 1
3. RXFLVL割込み（OTG_GINTSTS）を待ち、受信FIFOからデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC割込み（OTG_DOEPINTx）のアサートは、非アイソクロナスOUTデータ転送が正常に完了したことを示します。
5. OTG_DOEPTSIZxレジスタを読み出して、受信したデータペイロードのサイズを確認します。

- **一般のアイソクロナスOUTデータ転送**

このセクションでは、通常のアイソクロナスOUTデータ転送について説明します。

アプリケーションの要件：

1. 非アイソクロナス OUT データ転送のアプリケーションの要件はすべて、アイソクロナス OUT データ転送にも適用されます。
2. アイソクロナス OUT データ転送の場合、転送サイズおよびパケットカウントフィールドは、常に、単一フレームで受信できる最大パケットサイズのパケット数を設定しなければなりません。アイソクロナスOUTデータ転送は、複数のフレームにまたがってはいけません。
3. アプリケーションは、周期的フレームの終わり（OTG_GINTSTSのEOPF割込み）の前に、受信FIFOからすべてのアイソクロナスOUTデータパケット（データおよびステータス）を読み出す必要があります。
4. 次のフレームでデータを受信するにはEOPF（OTG_GINTSTS）の後とSOF（OTG_GINTSTS）の前に、アイソクロナスOUTエンドポイントを有効にしなければなりません。

内部データフロー：

1. アイソクロナスOUTエンドポイントの内部データフローは、非アイソクロナスOUTエンドポイントの内部データフローと基本的に同じですが、少し異なったところがあります。
2. エンドポイントイネーブルビットをセットし、NAK ビットをクリアすることによって、アイソクロナスOUTエンドポイントが有効にされたときには、偶数／奇数フレームビットも適切にセットされなければなりません。コアは、以下の条件が満たされた場合に限り、アイソクロナスOUTエンドポイント上で特定のフレームのデータを受信します。
   – EONUM（OTG_DOEPCTLxレジスタ）= FNSOF[0]（OTG_DSTSレジスタ）
3. アプリケーションが受信 FIFO からアイソクロナス OUT データパケット（データとステータス）を完全に読み出すと、コアはOTG_DOEPTSIZxレジスタのRXDPIDフィールドを、受信FIFOから読み出された最後のアイソクロナスOUTデータパケットのデータPIDで更新します。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZxレジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx　レジスタをエンドポイントの特性でプログラムし、エンドポイントイネーブル、ClearNAK、および偶数／奇数フレームの各ビットをセットします。
   – EPENA = 1
   – CNAK = 1
   – EONUM =（0：偶数／1：奇数）

3. RXFLVL 割込み（OTG_GINTSTS）を待ち、受信 FIFO からのデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC割込み（OTG_DOEPINTx）のアサートは、アイソクロナスOUTデータ転送が正常に完了したことを示します。この割込みは、必ずしもメモリ内のデータが良好であることを意味しません。
5. この割込みは、アイソクロナス OUT 転送で必ず検出されるとは限りません。その代わり、アプリケーションはOTG_GINTSTSでINCOMPISOOUT割込みを検出することができます。
6. OTG_DOEPTSIZx レジスタを読み出して、受信した転送データのサイズを確認し、フレームで受信したデータの有効性を確認します。アプリケーションは、以下の条件の1つが満たされた場合のみ、メモリに受信されたデータを有効として扱う必要があります。
   – RXDPID = DATA0（OTG_DOEPTSIZxレジスタ）、およびこのペイロードが受信されたUSBパケットの数 = 1
   – RXDPID = DATA1（OTG_DOEPTSIZxレジスタ）、およびこのペイロードが受信されたUSBパケットの数 = 2

   RXDPID = D2（OTG_DOEPTSIZxレジスタ）、およびこのペイロードが受信されたUSBパケットの数 = 3このペイロードが受信されたUSBパケットの数 =
   アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

   アプリケーションは、無効なデータパケットを破棄できます。

- **不完全アイソクロナスOUTデータ転送**

このセクションでは、アイソクロナスOUTデータパケットがコアの内部でドロップされたときのアプリケーションのプログラミングシーケンスについて説明します。

内部データフロー：

1. アイソクロナスOUTエンドポイントの場合、XFRC割込み（OTG_DOEPINTx）は必ずアサートされるわけではありません。コアがアイソクロナスOUTデータパケットをドロップした場合、アプリケーションは以下の状況でXFRC割込み（OTG_DOEPINTx）の検出に失敗することがあります。
   – 受信FIFOが完全なISO　OUTデータパケットを収容できない場合、コアは、受信したISO OUTデータをドロップします。
   – アイソクロナスOUTデータパケットがCRCエラー付きで受信されたとき。
   – コアが受信したアイソクロナスOUTトークンが破損しているとき。
   – アプリケーションが受信FIFOからデータを読み出すのに長時間かかっているとき。
2. コアが、すべてのアイソクロナス OUT エンドポイントへの転送を完了する前に周期的フレームの終わりを検出すると、不完全アイソクロナスOUTデータ割込み（OTG_GINTSTSレジスタのINCOMPISOOUT）をアサートして、アイソクロナスOUTエンドポイントのうち少なくとも1つでXFRC割込み（OTG_DOEPINTx）がアサートされていないことを示します。この時点で、不完全転送のエンドポイントは有効なままですがUSB上のこのエンドポイントでは、アクティブな転送は進行していません。

アプリケーションのプログラミングシーケンス：

1. INCOMPISOOUT割込み（OTG_GINTSTS）のアサートは、現在のフレームで、少なくとも1つのアイソクロナスOUTエンドポイントが転送を完了していないことを示します。
2. アイソクロナス OUT データがエンドポイントから完全に出されていないためにこの割込みが発生した場合、アプリケーションは処理を進める前に、受信FIFOからすべてのアイソクロナスOUTデータ（データとステータス）を出力する必要があります。
   – すべてのデータが受信　FIFO　から出されると、アプリケーションは　XFRC　割込み（OTG_DOEPINTx）を検出することができます。この場合、アプリケーションは、次のフ

レームでアイソクロナスOUTデータを受信するには、エンドポイントを再び有効にする必要があります。

3. INCOMPISOOUT割込み（OTG_GINTSTS）を受信すると、アプリケーションは、すべてのアイソクロナスOUTエンドポイントの制御レジスタ（OTG_DOEPCTLx）を読み出して、現在のマイクロフレームにおいて転送を完了しなかったエンドポイントを確認する必要があります。次の両方の条件が満たされた場合、エンドポイント転送は完了しません。
   – EONUMビット（OTG_DOEPCTLxレジスタ）= FNSOF[0]（OTG_DSTSレジスタ）
   – OTG_DOEPCTLxレジスタのEPENA = 1
4. 現在のフレーム番号が変更されないように、上記の手順はSOF割込み（ OTG_GINTSTS）が検出される前に実行されなければなりません。
5. 不完全転送のアイソクロナスOUTエンドポイントの場合、アプリケーションは、メモリ内のデータを破棄しOTG_DOEPCTLxレジスタのEPDISビットをセットすることによってエンドポイントを無効にする必要があります。
6. EPDISD 割込み（OTG_DOEPINTx）を待ち、次のフレームで新しいデータを受信するために、エンドポイントを有効にします。
   – コアがエンドポイントを無効にするには若干の時間がかかるので、アプリケーションは不良なアイソクロナスデータを受信した後、次のフレームのデータを受信できないことがあります。

- **非アイソクロナスOUTエンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

1. コアをグローバルOUT NAKモードにします。
2. 必要なエンドポイントを無効にします。
   – エンドポイントを無効にするときには、OTG_DOEPCTLのSNAKビットをセットする代わりにSTALL = 1にセットしてください（OTG_DOEPCTL）。
   STALLビットは常にNAKビットより優先されます。
3. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときは、STALLビット （OTG_DOEPCTLx）をクリアしなければなりません。
4. アプリケーションがSetFeature.Endpoint HaltまたはClearFeature.Endpoint Haltコマンドによってエンドポイントの STALLビットをセットまたはクリアする場合、STALLビットは、アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

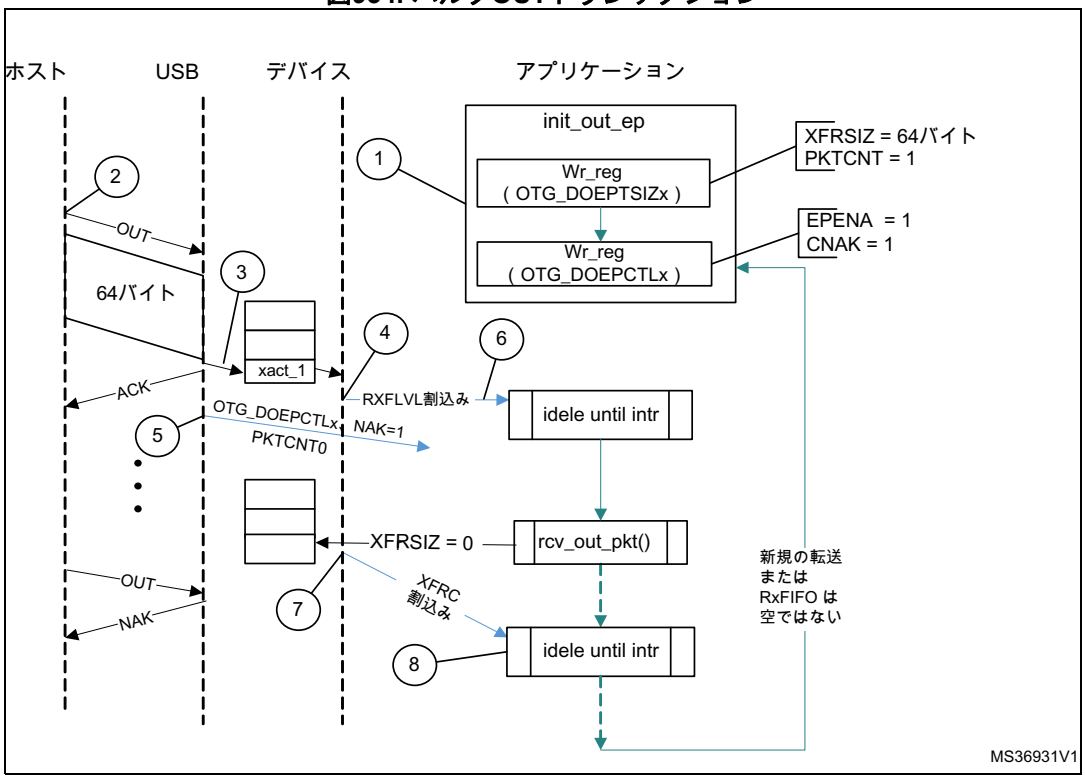
例

このセクションでは、いくつかの基本的な転送タイプとシナリオについて説明します。

- バルクOUTトランザクション

図934にUSBからAHBへの単一バルクOUTデータパケットの受信と、このプロセスに伴うイベントを示します。

図934. バルクOUTトランザクション



SetConfiguration/SetInterfaceコマンドの後、アプリケーションはCNAK = 1およびEPENA = 1（OTG_DOEPCTLx レジスタ）をセットし、OTG_DOEPTSIZx レジスタの適切なXFRSIZ およびPKTCNTビットをセットすることによって、すべてのOUTエンドポイントを初期化します。

1. ホストは、データ（OUTトークン）のエンドポイントへの送信を試みます。
2. コアは、USB上でOUTトークンを受信すると、スペースが使用可能なので、Rx FIFOにパケットを格納します。
3. Rx FIFOに完全なパケットを書き込んだ後、コアはRxFLVL割込み（OTG_GINTSTS）をアサートします。
4. USBパケットのPKTCNT番号を受信すると、コアは、それ以上パケットが受信されないように、このエンドポイントのNAKビットを内部でセットします。
5. アプリケーションは割込みを処理してRx FIFOからデータを読み出します。
6. アプリケーションがすべてのデータ（XFRSIZ に相当）を読み出すと、コアは、XFRC 割込み（OTG_DOEPINTx）を生成します。
7. アプリケーションは割込みを処理して、XFRC割込みビット（OTG_DOEPINTx）の設定を使用して、意図した転送が完了したかどうかを確認します。

INデータ転送

- **パケットの書込み**

このセクションでは、専用の送信FIFOが有効なときに、アプリケーションがデータパケットをエンドポイントFIFOに書き込む方法について説明します。

1. アプリケーションは、ポーリングまたは割込みモードのいずれかを選択できます。
   – ポーリングモードでは、アプリケーションは、OTG_DTXFSTSxレジスタを読み出すことによってエンドポイント送信データFIFOのステータスを監視して、データFIFOに十分なスペースがあるかどうかを確認します。
   – 割込みモードでは、アプリケーションは、TXFE割込み（OTG_DIEPINTx）を待ってから、OTG_DTXFSTSxレジスタを読み出して、データFIFO内に十分なスペースがあるかどうかを確認します。
   – 単一の非ゼロ長データパケットを書き込むには、パケット全体を書き込むためのスペースがデータFIFOになければなりません。
   – ゼロ長パケットを書き込むには、アプリケーションはFIFOのスペースを考慮してはなりません。
2. 上記の方法の1つを使用して、アプリケーションが送信パケットを書き込むのに十分なスペースがあることを確認するときには、アプリケーションは、データをデータFIFOに書き込む前に、まず、エンドポイント制御レジスタに書き込む必要があります。通常、アプリケーションは、エンドポイントイネーブルビットをセットする場合を除き、レジスタの内容が変更されないように、OTG_HS_DIEPCTLxレジスタに読出し／変更／書込みを行う必要があります。

アプリケーションは、使用可能なスペースがあれば、同じエンドポイントに対する複数のパケットを送信FIFOに書き込むことができます。周期的INエンドポイントの場合、アプリケーションは1つのマイクロフレームのパケットのみを書き込む必要があります。アプリケーションは、前のトランザクションの転送完了割込みを受信した後でのみ、次の周期的トランザクションのパケットを書き込むことができます。

- **INエンドポイントNAKの設定**

内部データフロー：

1. アプリケーションが特定のエンドポイントのIN NAKをセットすると、コアは、エンドポイントの送信FIFOにデータがあるかどうかに関係なく、そのエンドポイントでのデータ送信を停止します。
2. 非アイソクロナスINトークンはNAKハンドシェイク応答を受信します。
   – アイソクロナスINトークンはゼロデータ長のパケット応答を受信します。
3. コアは、OTG_DIEPCTLxのSNAKビットに応答して、OTG_DIEPINTxのINEPNE割込み（INエンドポイントNAK有効）をアサートします。
4. この割込みがアプリケーションによって検出されると、アプリケーションは、エンドポイントがIN NAKモードにあるとみなすことができます。この割込みは、アプリケーションがOTG_DIEPCTLxのCNAKビットをセットすることによってクリアできます。

アプリケーションのプログラミングシーケンス：

1. 特定のINエンドポイントでのデータ送信を停止するには、アプリケーションは、IN NAKビットをセットする必要があります。このビットをセットするには、以下のフィールドをプログラムする必要があります。
   – OTG_DIEPCTLxレジスタのSNAK = 1
2. OTG_DIEPINTxのINEPNE割込みのアサートを待ちます。この割込みは、コアがエンドポイントでのデータ送信を停止したことを示します。

3. コアは、アプリケーションがNAKビットをセットした後のNAK有効割込みがアサートされる前に、エンドポイントで有効なINデータを送信できます。
4. アプリケーションはOTG_DIEPMSKのINEPNEMビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_DIEPMSKレジスタのINEPNEM = 0
5. エンドポイントNAKモードを終了するには、アプリケーションはOTG_DIEPCTLxのNAKステータスビット（NAKSTS）をクリアする必要があります。これによって INEPNE 割込み（OTG_DIEPINTx）もクリアされます。
   – OTG_DIEPCTLxレジスタのCNAK = 1
6. アプリケーションがこの割込みをマスクするのが早すぎた場合、以下のようにしてマスク解除を行う必要があります。
   – OTG_DIEPMSKレジスタのINEPNEM = 1

- **INエンドポイントディセーブル**

以前に有効にされた特定のINエンドポイントを無効にするには、以下のシーケンスを使用します。

アプリケーションのプログラミングシーケンス：

1. アプリケーションは、AHB でのデータ書込みを停止して、IN エンドポイントを無効にする必要があります。
2. アプリケーションは、エンドポイントをNAKモードに設定する必要があります。
   – OTG_DIEPCTLxレジスタのSNAK = 1
3. OTG_DIEPINTxのINEPNE割込みを待ちます。
4. 無効にしなければならないエンドポイントについてOTG_DIEPCTLxレジスタの以下のビットをセットします。
   – OTG_DIEPCTLxレジスタのEPDIS = 1
   – OTG_DIEPCTLxレジスタのSNAK = 1
5. OTG_DIEPINTxレジスタのEPDISD割込みのアサートは、コアが指定されたエンドポイントを完全に無効にしたことを示します。割込みのアサートとともに、コアは、以下のビットもクリアします。
   – OTG_DIEPCTLxレジスタのEPENA = 0
   – OTG_DIEPCTLxレジスタのEPDIS = 0
6. アプリケーションは、周期的IN EPのOTG_DIEPTSIZxレジスタを読み出して、エンドポイント上のどれだけのデータがUSBで送信されたかを計算する必要があります。
7. アプリケーションは、OTG_GRSTCTL レジスタの以下のフィールドを設定することによって、エンドポイント送信FIFO内のデータを排出する必要があります。
   – TXFNUM（OTG_GRSTCTL）= エンドポイント送信FIFO番号
   – TXFFLSH（OTG_GRSTCTL ）= 1

アプリケーションは、TXFFLSHビットがコアによってクリアされるまで（すなわち、一掃動作の終了まで）、OTG_GRSTCTLレジスタにポーリングする必要があります。このエンドポイントで新しいデータを送信するために、アプリケーションは後で、エンドポイントを再び有効にできます。

- **INエンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング·シーケンスを使用する必要があります。

動作シーケンス：

1. INエンドポイントを無効にするには、以下をセットします。
   – すべてのOTG_DIEPCTLxレジスタのEPDIS = 1
2. OTG_DIEPINTxのEPDIS割込みを待ちます。これはINエンドポイントが完全に無効にされたことを示します。EPDIS割込みがアサートされると、以下のビットがクリアされます。
   – OTG_DIEPCTLxレジスタのEPDIS = 0
   – OTG_DIEPCTLxレジスタのEPENA = 0
3. TxFIFOを一掃するために、次のビットをプログラムします。
   – OTG_GRSTCTLのTXFFLSH = 1
   – OTG_GRSTCTLのTXFNUM =「エンドポイントに固有のFIFO番号」
4. アプリケーションは、OTG_GRSTCTLのTXFFLSHがクリアされるまでポーリングを開始できます。このビットがクリアされると、Tx FIFOにデータが残っていないことを確認できます。

- **一般の非周期的INデータ転送**

アプリケーションの要件：

1. IN転送をセットアップする前に、アプリケーションは、IN転送の一部として送信されるすべてのデータが単一バッファの一部であることを確認する必要があります。
2. IN転送の場合、エンドポイント転送サイズレジスタの転送サイズフィールドは、最大パケットサイズの複数のパケットと単一のショートパケットから成るペイロードを表します。このショートパケットは、転送の最後に送信されます。
   – 転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには：
     転送サイズ [EPNUM] = *x* * MPSIZ[EPNUM] + sp
     （sp > 0）の場合、パケットカウント [EPNUM] = x + 1。
     そうでない場合、パケットカウント [EPNUM] = *x*
   – 単一のゼロ長データパケットを送信するには：
     転送サイズ [EPNUM] = 0
     パケットカウント [EPNUM] = 1
   – 転送の終わりに少数の最大パケットサイズのパケットとゼロ長データパケットを送信するには、アプリケーションは転送を2つの部分に分ける必要があります。最初の部分では最大パケットサイズのデータパケットを送信し、2番目の部分ではゼロ長データパケットのみを送信します。
     最初の転送：転送サイズ [EPNUM] = *x* * MPSIZ[epnum]；パケットカウント = *n*；
     2番目の転送：転送サイズ [EPNUM] = 0；パケットカウント = 1；
3. エンドポイントがデータ転送のために有効にされると、コアは、転送サイズレジスタを更新します。IN転送の終了時に、アプリケーションは、転送サイズレジスタを読み出して、送信FIFOにポストされたデータのうち、どれだけがUSBで送信されたかを確認する必要があります。
4. 送信FIFOにフェッチされたデータ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – USBで送信されたデータ = （アプリケーションがプログラムした初期パケットカウント – コアが更新した最終パケットカウント）* MPSIZ[EPNUM]
   – USB上でこれから送信されるデータ=（アプリケーションがプログラムした初期転送サイズ – USBで送信されたデータ）

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. アプリケーションは、要求されたデータをエンドポイントの送信FIFOに書き込む必要もあります。
3. アプリケーションによってパケットが送信FIFOに書き込まれるたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが0になるまで、データはアプリケーションによってメモリからフェッチされます。FIFOにデータを書き込んだ後、「FIFO内のパケット数」はインクリメントされます（これは3bitのカウントであり、各INエンドポイントの送信FIFOについて、コアによって内部で維持されます。INエンドポイントFIFO内にコアによって一度に維持されるパケットの最大数は8です）。ゼロ長のパケットの場合、各FIFOについて、FIFO内にデータがないことを示す個別のフラグがセットされます。
4. データが送信FIFOに書き込まれると、コアは、INトークンの受信時に読み出します。非アイソクロナスINデータパケットがACKハンドシェイクとともに送信されるたびに、エンドポイントのパケットカウントは、0になるまで1ずつデクリメントされます。パケットカウントは、タイムアウト時にはデクリメントされません。
5. ゼロ長パケットの場合（内部ゼロ長フラグで示されます）、コアはINトークンのゼロ長パケットを送出し、パケットカウントフィールドをデクリメントします。
6. 受信されたINトークンのデータがFIFOになく、そのエンドポイントのパケットカウントフィールドがゼロの場合、コアは、そのエンドポイントについて、「TxFIFOがエンプティのときINトークンが受信された」という割込み（ITTXFE割込み）を生成します。ただし、エンドポイントのNAKビットがセットされていない場合に限ります。コアは、USB上の非アイソクロナスエンドポイントに対してNAKハンドシェイクで応答します。
7. コアは、内部でFIFOポインタを巻き戻し、タイムアウト割込みは生成されません。
8. 転送サイズが0であり、パケットカウントが0のとき、エンドポイントの転送完了割込み（XFRC）が生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPTSIZxレジスタで転送サイズと対応するパケットカウントをプログラムします。
2. OTG_DIEPCTLxレジスタをエンドポイントの特性でプログラムし、CNAKビットおよびEPENA（エンドポイントイネーブル）ビットをセットします。
3. 非ゼロ長データパケットを送信するときには、アプリケーションは、OTG_ レジスタ（xはそのエンドポイントに関連するFIFO番号）をポーリングして、データFIFOに十分なスペースがあるかどうかを確認する必要があります。アプリケーションは、データを書き込む前に、オプションでTXFE（OTG_DIEPINTx）を使用することができます。

- **一般の周期的INデータ転送**

このセクションでは、典型的な周期的INデータ転送について説明します。

アプリケーションの要件：

1. 3411ページの一般の非周期的INデータ転送 のアプリケーションの要件1、2、3、および4は、要件2が少し変更されていることを除いて、周期的INデータ転送にも適用されます。
   – アプリケーションは複数の最大パケットサイズのデータパケットの送信、または複数の最大パケットサイズのパケットに最後にショートパケットを加えた送信のみを行うことができます。転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには、以下の条件が満たされなければなりません。

   転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
   （*x* は整数 ≥ 0、および0 ≤ sp < MPSIZ[EPNUM]）

（sp > 0）の場合、パケットカウント [EPNUM] = *x* + 1
そうでない場合、パケットカウント [EPNUM] = *x* ;

MCNT[EPNUM] = パケットカウント [EPNUM]

– アプリケーションは転送の最後にゼロ長データパケットを送信することはできません。自分で単一のゼロ長パケットを送信することは可能です。単一のゼロ長データパケットを送信するには :

– 転送サイズ [EPNUM] = 0

パケットカウント [EPNUM] = 1

MCNT[EPNUM] = パケットカウント [EPNUM]

2. アプリケーションは一度に1フレームのデータ転送のみをスケジュールすることができます。
   – (MCNT – 1) × MPSIZ ≤ XFERSIZ ≤ MCNT × MPSIZ
   – PKTCNT = MCNT（OTG_DIEPTSIZxレジスタ）
   – XFERSIZ < MCNT * MPSIZの場合、転送の最後のデータパケットはショートパケットです。
   – 注 : MCNTはOTG_DIEPTSIZxレジスタに、MPSIZはOTG_DIEPCTLxレジスタに、PKTCNTはOTG_DIEPTSIZxレジスタに、またXFERSIZはOTG_DIEPTSIZxレジスタにあります。
3. フレームで送信される完全なデータは、INトークンが受信される前に、アプリケーションによって送信FIFOに書き込まれなければなりません。INトークンが受信されたときに、フレームあたりで送信されるデータの1ワードが送信FIFO内で欠落していても、コアはFIFOがエンプティである場合のように動作します。送信FIFOがエンプティのとき :
   – アイソクロナスINエンドポイントに対してUSB上でゼロデータ長のパケットが送信されます。
   – INエンドポイントに割込みをかけるため、USB上でNAKハンドシェイクが送信されます。

内部データフロー :

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. また、アプリケーションは、要求されたデータをエンドポイントの関連する送信FIFOに書き込む必要があります。
3. アプリケーションがパケットを送信FIFOに書き込むたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが0になるまで、アプリケーションメモリからデータがフェッチされます。
4. 周期的エンドポイントのINトークンが受信されると、コアは、FIFOのデータ（ある場合）を送信します。フレームの完全なデータペイロード（専用FIFOモードでは完全なパケット）がFIFOに存在しない場合、コアは、エンドポイントに対するTx FIFOエンプティ割込み時にINトークンの受信を生成します。
   – アイソクロナスINエンドポイントに対して、USB上でゼロ長のデータパケットが送信されます。
   – INエンドポイントに割込みをかけるため、USB上でNAKハンドシェイクが送信されます。
5. エンドポイントのパケットカウントは、以下の条件のもとで1ずつデクリメントされます。
   – アイソクロナスエンドポイントに対して、ゼロ長または非ゼロ長データパケットが送信されたとき。
   – 割込みエンドポイントに対して、ACKハンドシェイクが送信されたとき。
   – 転送サイズとパケットカウントの両方が0のとき、エンドポイントに対して転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。
6. 「周期的フレームインターバル」（OTG_DCFGのPFIVLによって制御されます）で、コアが現在のフレームについてスケジュールされたアイソクロナスINエンドポイントFIFOのいずれかが非エンプティであることを検出すると、コアはOTG_GINTSTSのIISOIXFR割込みを生成します。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPCTLxレジスタをエンドポイントの特性でプログラムして、CNAKおよびEPENAビットをセットします。
2. 次のフレームで送信するデータを送信FIFOに書き込みます。
3. OTG_DIEPINTxレジスタのITTXFE割込みのアサートは、アプリケーションが送信するすべてのデータを送信FIFOにまだ書き込んでいないことを示します。
4. この割込みが検出されたときに、割込みエンドポイントがすでに有効だった場合、割込みは無視されます。まだ有効になっていなかった場合は、次のINトークン試行の際にデータを送信できるように、エンドポイントを有効にします。
5. OTG_DIEPINTxレジスタのITTXFE割込みがない状態でのXFRC割込み（OTG_DIEPINTx）のアサートは、アイソクロナスIN転送が正常に完了したことを示します。OTG_DIEPTSIZxレジスタの読出し結果は、転送サイズ = 0、かつパケットカウント = 0、すなわち、すべてのデータがUSBで送信されたことを示さなければなりません。
6. ITTXFE割込み（OTG_DIEPINTx）の有無にかかわらず、XFRC割込み（OTG_DIEPINTx）のアサートは、インタラプトIN転送が正常に完了したことを示します。OTG_DIEPTSIZxレジスタの読出し結果は、転送サイズ = 0、かつパケットカウント= 0、すなわち、すべてのデータがUSBで送信されたことを示さなければなりません。
7. 前述のどの割込みもない状態でのOTG_GINTSTSの不完全アイソクロナスIN転送（IISOIXFR）割込みのアサートは、コアが現在のフレームで少なくとも1つの周期的INトークンも受信しなかったことを示します。

- **不完全アイソクロナスINデータ転送**

このセクションでは、アプリケーションが不完全なアイソクロナスINデータ転送で行う必要があることについて説明します。

内部データフロー：

1. アイソクロナスIN転送は、以下の条件のいずれかがあてはまる場合に不完全として扱われます。
   a) コアが、少なくとも1つのアイソクロナスINエンドポイントで破損したアイソクロナスINトークンを受信した場合。この場合、アプリケーションは、不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFRビット）を検出します。
   b) アプリケーションが完全なデータペイロードを送信FIFOに書き込むのに時間がかかり、完全なデータペイロードがFIFOに書き込まれる前にINトークンが受信された場合。この場合、アプリケーションは、OTG_DIEPINTxのTx FIFOエンプティ割込み時にINトークンの受信を検出します。最終的には、周期的フレームの終わりでの不完全アイソクロナスIN転送割込み（OTG_GINTSTSのIISOIXFR）になるので、アプリケーションはこの割込みを無視することができます。

   コアは、受信したINトークンに応答して、USB上にゼロ長データパケットを送信します。
2. アプリケーションは、できるだけ速やかに、送信FIFOへのデータペイロードの書込みを停止する必要があります。
3. アプリケーションは、エンドポイントのNAKビットとディセーブルビットをセットする必要があります。
4. コアは、エンドポイントを無効にし、ディセーブルビットをクリアし、エンドポイントのエンドポイントディセーブル割込みをアサートします。

アプリケーションのプログラミングシーケンス：

1. 最終的には不完全アイソクロナスIN転送割込み（OTG_GINTSTS）になるので、アプリケーションは、アイソクロナスINエンドポイントでOTG_DIEPINTxのTx FIFOエンプティ割込み時に受信されたINトークンを無視することができます。
2. 不完全アイソクロナスIN転送割込み（OTG_GINTSTS）のアサートは、少なくとも1つのアイソクロナスINエンドポイントで不完全アイソクロナスIN転送があったことを示します。
3. アプリケーションは、すべてのアイソクロナスINエンドポイントのエンドポイント制御レジスタを読み出して、不完全INデータ転送があるエンドポイントを検出する必要があります。
4. アプリケーションは、AHB上でこれらのエンドポイントに関連する周期的送信FIFOへのデータの書込みを停止する必要があります。
5. OTG_DIEPCTLxレジスタの以下のフィールドをプログラムして、エンドポイントを無効にします。
   – OTG_DIEPCTLxレジスタのSNAK = 1
   – OTG_DIEPCTLxレジスタのEPDIS = 1
6. OTG_DIEPINTx のエンドポイントディセーブル割込みのアサートは、コアがエンドポイントを無効にしたことを示します。
   – この時点で、アプリケーションは、次のマイクロフレームで新しい転送を行うために、関連する送信FIFO内のデータを一掃するか、エンドポイントを有効にすることによって、FIFO内の既存のデータに上書きする必要があります。データを一掃するには、アプリケーションはOTG_GRSTCTLレジスタを使用する必要があります。

- **非アイソクロナスINエンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

アプリケーションのプログラミングシーケンス：

1. 停止するINエンドポイントを無効にします。STALLビットもセットします。
2. エンドポイントがすでに有効になっているときには、OTG_DIEPCTLxのEPDIS = 1にします。
   – OTG_DIEPCTLxレジスタのSTALL = 1にします。
   – STALLビットは常にNAKビットより優先されます。
3. エンドポイントディセーブル割込み（OTG_DIEPINTx）のアサートは、コアが指定されたエンドポイントを無効にしたことをアプリケーションに示します。
4. アプリケーションは、エンドポイントのタイプに応じて、非周期的または周期的FIFOをフラッシュする必要があります。非周期的エンドポイントの場合、アプリケーションは、データを送信するために、停止する必要がない他の非周期的エンドポイントを再び有効にする必要があります。
5. アプリケーションがエンドポイントのSTALLハンドシェイクを終了する準備ができたときには、OTG_DIEPCTLxのSTALLビットがクリアされなければなりません。
6. アプリケーションがSetFeature.Endpoint HaltコマンドまたはClearFeature.Endpoint Haltコマンドによってエンドポイントの STALLビットをセットまたはクリアする場合、STALLビットは、アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

特殊なケース：コントロールOUTエンドポイントの停止

コントロール転送のデータステージで、ホストがSETUPパケットで指定されたより多くのIN/OUTトークンを送信した場合、コアは、IN/OUT トークンを停止する必要があります。この場合、アプリケーションは、コアがSETUPパケットで指定されたデータ量を転送した後、コントロール転送のデータステージで、OTG_DIEPINTxのITTXFE割込みとOTG_DOEPINTxのOTEPDIS割込みを有効にしなければなりません。その後、アプリケーションがこの割込みを受信したとき、アプリケーションは、対応するエンドポイント制御レジスタのSTALLビットをセットし、この割込みをクリアする必要があります。

73.15.7 最悪ケースの応答時間

OTG_HSコントローラがデバイスとして機能するとき、アイソクロナスOUTに続く任意のトークンで、最悪ケースの応答時間があります。この最悪ケース応答時間は、AHBクロック周波数に依存します。

コアのレジスタはAHBドメインにあり、コアはこれらのレジスタを更新するまでは、別のトークンを受け入れません。アイソクロナストランザクションの場合、ハンドシェイクはなく、次のトークンがすぐに入ってくる可能性があるため、アイソクロナスOUTに続くトークンで最悪ケースが生じます。この最悪ケース値は、AHBクロックがPHYクロックと同じときには7 PHYクロックです。AHBクロックが高速なほど、この値は小さくなります。

この最悪ケース条件が発生した場合、コアは、バルク／インタラプトトークンにNAKで応答し、アイソクロナスおよびSETUPトークンをドロップします。ホストは、これをSETUPのタイムアウト条件と解釈して、SETUPパケットを再試行します。アイソクロナス転送の場合、不完全アイソクロナスIN転送割込み（IISOIXFR）と不完全アイソクロナスOUT転送割込み（IISOOXFR）は、アイソクロナスIN/OUTパケットがドロップされたことをアプリケーションに知らせます。

OTG_GUSBCFGのTRDTの値の選択

TRDT（OTG_GUSBCFG）の値は、MACがINトークンを受信した後、FIFOステータスとPFCブロックから最初のデータを取得するまでの時間（PHYクロック数）です。この時間には、PHYクロックとAHBクロック間の同期遅延も含まれます。この場合の最悪ケース遅延は、AHBクロックがPHYクロックと同じときに発生します。この場合、遅延は5クロックです。

MACがINトークンを受信すると、この情報（トークンの受信）はPFC（PFCはAHBクロックで動作します）によってAHBに同期されます。次に、PFCは、SPRAMからデータを読み出し、それらをデュアルクロックソースバッファに書き込みます。MACは、データをソースバッファ（深さ4）から読み出します。

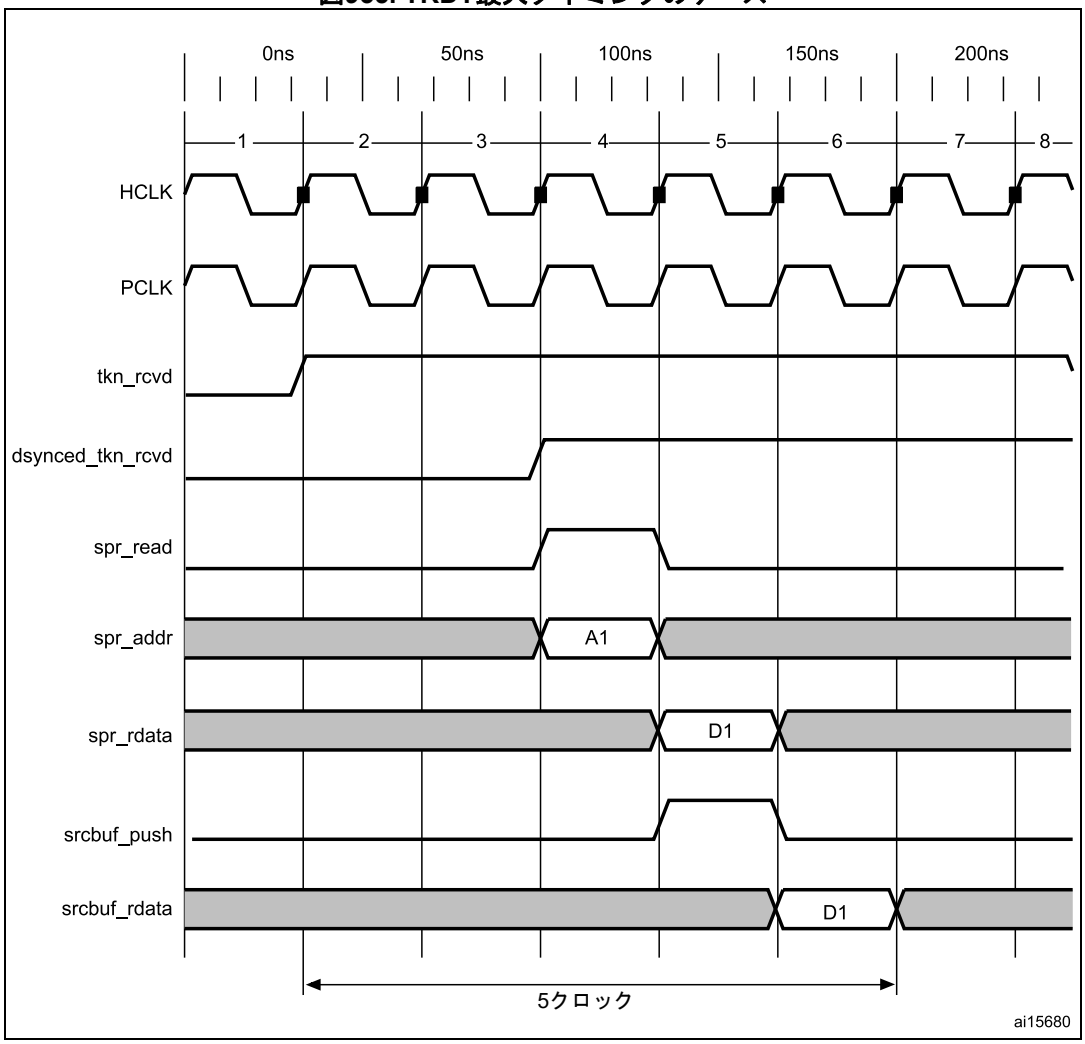
AHBがPHYより高い周波数で動作している場合、アプリケーションはTRDT（OTG_GUSBCFG）の値として、より小さな値を使用することができます。

図935に以下の信号を示します。

- tkn_rcvd：MACからPFCへの、トークンが受信されたという情報
- dynced_tkn_rcvd：PCLKからHCLKドメインへの二重同期tkn_rcvd
- spr_read：SPRAMへの読出し
- spr_addr：SPRAMのアドレス
- spr_rdata：SPRAMからのデータの読出し
- srcbuf_push：ソースバッファへのプッシュ
- srcbuf_rdata：ソースバッファからのデータの読出しMACによるデータの検出

TRDTの値を計算するには、表 762：TRDT値を参照してください。

図935. TRDT最大タイミングのケース



73.15.8 OTGプログラミングモデル

OTG_HS コントローラはOTGデバイスです。コアがAプラグに接続されると、Aデバイスと呼ばれます。コアがBプラグに接続されると、Bデバイスと呼ばれます。

74 USB Type-C®/USB Power Deliveryインタフェース（UCPD）

74.1 概要

USB Type-C/USB Power Deliveryインタフェースの構成要素：

- Universal Serial Bus Type-Cケーブルおよびコネクタ仕様：リリース2.2、2022年10月
- Universal Serial Bus Power Delivery仕様：
  - リビジョン2.0、バージョン1.3、2017年1月12日
  - リビジョン3.1、バージョン1.6、2022年10月

Type-Cケーブルでの動作用に、Power Delivery（PD）仕様の物理層と、CC信号手法（VBUSなし）を統合します。

74.2 UCPDの主な機能

- USB Type-C仕様リリース2.2に準拠
- USB Power Delivery仕様リビジョン2.0および3.1に準拠
  - PPSなどの高度なアプリケーションを有効化（プログラム可能な電源機器）
- STOPモードの低電力動作のサポート
- 内蔵アナログPHY
  - USB Type-Cのプルアップ（Rp、すべての値）およびプルダウン（Rd）レジスタ
  - 『デッドバッテリ』Rdサポート
  - USB Power Deliveryのメッセージの送受信
  - FRS（ファストロールスワップ）Rxのサポート
- デジタル･コントローラ
  - BMC（バイフェーズマークコーディング）のエンコードおよびデコード
  - 4b5bでのエンコードおよびデコード
  - 割込みが生成される デバウンスによるUSB Type-Cレベル検出
  - FRS信号
  - 割込みが生成されるFRS検出
  - 割込みが生成されるUSB Power DeliveryペイロードのDMA互換のバイトレベルインタフェース
  - USB Power Deliveryクロックプリスケーラ/分周回路
  - CRC生成／チェック
  - 順序集合（受信時のプログラム可能な順序集合のマスクを含む）のサポート
  - 受信Rxストリームからのクロックリカバリ

74.3 UCPDの実装

デバイスには、1つのUSBタイプCポートに対応する1つのUCPDコントローラが装備されています。

表 765. UCPDの実装(1)

UCPDの機能	STM32U575/585 Rev.X	STM32U535/545を除くその他のSTM32U5xxデバイスモデル
UCPDx_DBCC1およびUCPDx_DBCC2外部信号によるデッドバッテリのサポート	X	X
オルタネート機能ピンとして使用できるUCPDx_FRSTX1/2	X	X
全自動のトリミング	X(2)	-(3)
USB PDレシーバのハードウェアフィルタ制御	X	X
ディスクリート・コンポーネントPHYのサポート	-	-

1. “X” はサポートされています。“-” はサポートされていません。
2. ソフトウェアトリミングは不要です。
3. セクション 74.5.5 : UCPDソフトウェアのトリミングで説明するソフトウェアのトリミングを適用します。

以下の表に、不揮発性メモリに格納されたトリムデータのメモリ位置を示します。これは、セクション 74.5.5 : UCPDソフトウェアのトリミングで説明するソフトウェアのタイミング手順で使用するものです。

表 766. UCPDソフトウェアのtrimデータ

名前	不揮発性メモリ位置	
	アドレス	ビット
3A0_CC1 [3:0]	0x0BFA 0545	3:0
3A0_CC2 [3:0]	0x0BFA 0547	3:0
1A5_CC1 [3:0]	0x0BFA 07A7	3:0
1A5_CC2 [3:0]	0x0BFA 07A8	3:0
Rd_CC1 [3:0]	0x0BFA 0544	3:0
Rd_CC2 [3:0]	0x0BFA 0546	3:0

74.4 UCPDの機能詳細

CUPDペリフェラルでは、USB Power Delivery制御インタフェース仕様のためのハードウェアサポートを行うため、その目的で特別に設計されたI/Oを使用します。

内蔵PHYでは、Type-C電圧レベルを直接検出し、Power Delivery BISTキャリアモード2（Txのみ）、BISTテストデータ（TxおよびRx）、Power Delivery Rx FRS信号をサポートします。

Power DeliveryのFRS Tx信号では、UCPD_FRSTX1/2 ピン（オルタネート機能）の設定によって外付けのNMOSトランジスタを制御するようにデバイスを設定できるので、CCラインを低抵抗でプルダウンできます。

UCPDトランスミッタBMC（バイフェーズマーク）はデータ、たとえばプリアンブル、SOP、プロトコル層からのペイロードデータ（4b5bエンコーディング後）、CRC、およびType-Cコネクタ CCライ

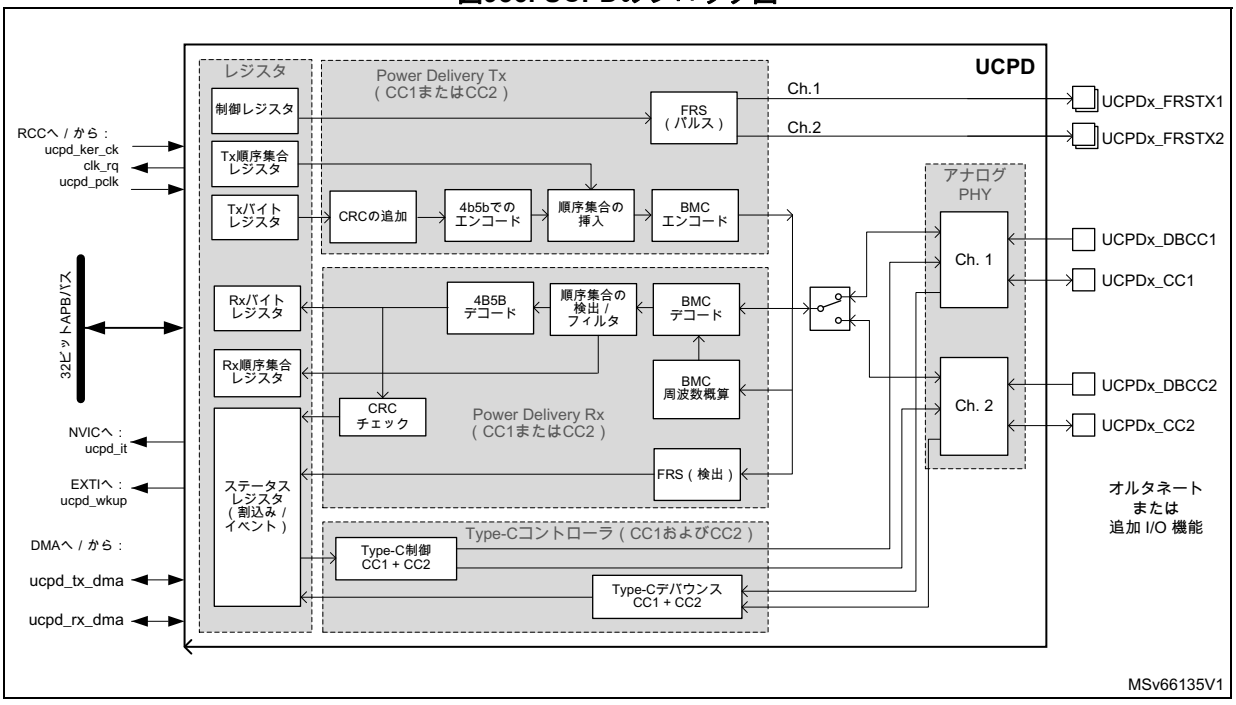
ン上のEOPをエンコードし、送信します。自動的にフレーム間ギャップを挿入し、「Hard Reset」を実行します。

UCPDレシーバはSOPを検出し、受信ストリームをBMCデコードし、プリアンブルを回復させ、ペイロードデータを4b5bでデコードし、EOPを検出し、CRCをチェックします。5つのKコードSOPと2つのリセットの順序集合、2つのソフトウェア定義のパターン（標準に定義されているとおり、4つのKコードのうち3つのみを正しく受信することが可能）が自動的に検出されます。

STOPモードでは、ペリフェラルは受信USB Power Deliveryメッセージと、低電力動作を可能にするFRS信号を検出する能力を維持します。

74.4.1 UCPDブロック図

図936. UCPDのブロック図



以下の表に、外部信号（オルタネートまたは追加のI/O機能）を示します。

表 767. ピンのUCPD信号

ピン名	信号タイプ	説明
UCPDx_FRSTX1	出力	USB Type-Cのファストロールスワップ（FRS）信号（DRPのみに適用）。この信号（アクティブ·ハイ）は、CC1ラインをプルダウンする外付けのNMOSトランジスタを駆動します。
UCPDx_FRSTX2	出力	USB Type-Cのファストロールスワップ（FRS）信号（DRPのみに適用）。この信号（アクティブ·ハイ）は、CC2ラインをプルダウンする外付けのNMOSトランジスタを駆動します。
UCPDx_CC1	入力／出力	USB Type-C設定制御ライン1。USB Type-CコネクタのCC1端子に接続します。
UCPDx_CC2	入力／出力	USB Type-C設定制御ライン2。USB Type-CコネクタのCC2端子に接続します。

表 767. ピンのUCPD信号 （続き）

ピン名	信号タイプ	説明
UCPDx_DBCC1	入力	USB Type-C設定制御ライン1のデッドバッテリ信号。デッドバッテリのサポートが必要な場合はUSB Type-CコネクタのCC1端子に接続します。
UCPDx_DBCC2	入力	USB Type-C設定制御ライン2のデッドバッテリ信号。デッドバッテリのサポートが必要な場合はUSB Type-CコネクタのCC2端子に接続します。

次の表に、主要な内部信号を示します。

表 768. UCPD内部信号

内部信号名	信号タイプ	説明
ucpd_pclk	入力	レジスタ用のAPBクロック
ucpd_ker_ck	入力	カーネルクロック
ucpd_tx_dma	入力／出力	Rx DMA確認応答/リクエスト
ucpd_rx_dma	入力／出力	Tx DMA確認応答/リクエスト
ucpd_it	出力	NVICに接続された割込みリクエスト（すべての割込みの論理和）
ucpd_wkup	出力	EXTIに接続されたウェイクアップリクエスト
clk_rq	出力	RCCに接続されたクロックリクエスト

74.4.2 UCPDのリセットおよびクロック

ペリフェラルはシングルリセット信号（APBバスリセット）を備えています。

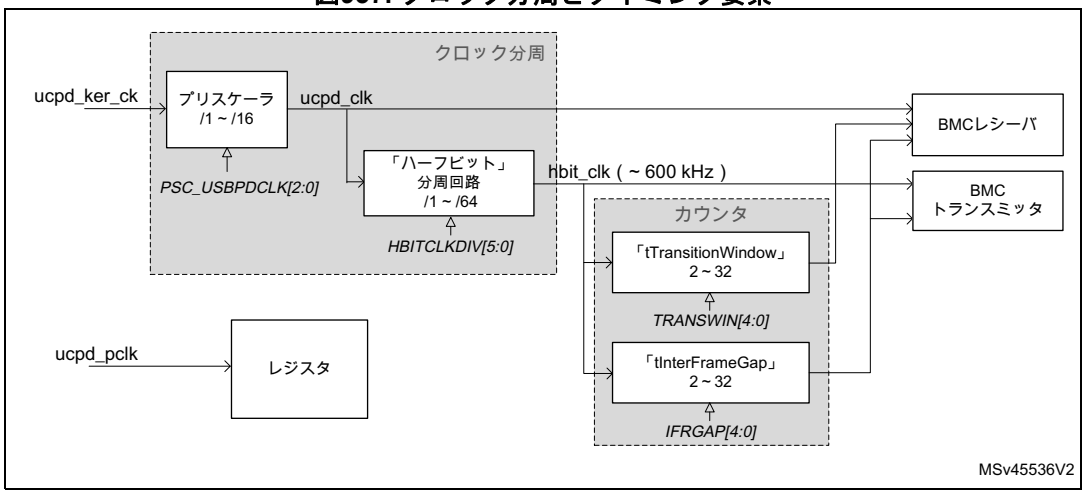
レジスタセクションは、APBクロック（ucpd_pclk）によりクロック供給されます。

トランスミッタのメイン機能部分は、ucpd_clkクロックによりクロック供給され、UCPD_CFGR1レジスタのPSC_USBPDCLK[2:0] ビットフィールドに従ってucpd_ker_ck（HSI16）クロックからプリスケールされます。レシーバのメイン機能部分は、受信ビットストリームから復元されたucpd_rx_clkによりクロック供給されます。

レシーバは、6 ~ 18 MHzのクロック周波数範囲で動作するように設計されています。ただし、最適なパフォーマンスが得られるのは、9 ~ 18 MHzの範囲内です。

下図に、UCPDペリフェラルのクロック供給およびタイミングの要素を示します。

図937. クロック分周とタイミング要素



適切な遅延をセットするには、USB PD仕様を参照してください。*tTransitionWindow* と、特に *tInterFrameGap* では、どのような場合でも指定したタイミングを満たすことができるよう、クロック周波数の不確定性を考慮する必要があります。

74.4.3 物理層のプロトコル

物理層は、USB Power Delivery仕様の基礎となる信号送信をカバーします。

トランスミッタ側では、主な機能は、定義されたパケットフォーマットに従ってパケットを形成することです。一般的に、以下の内容が含まれます。

- プリアンブル
- パケットの開始（SOP、順序集合）
- ペイロード･ヘッダ
- ペイロード･データ
- 巡回冗長検査（CRC）情報
- パケットの終了（EOP）

CCラインに送られる前に、指定されたタイミング制限について、データストリームがBMCエンコードされます。

受信側の基本タスクは、以下の内容を実行することです。

- パケットの開始（SOP、順序集合）情報の抽出
- ペイロード･ヘッダの抽出
- ペイロード･データの抽出
- CRCの受信と確認
- パケットの終了（EOP）の受信

受信は基本的に送信プロセスの逆であるため、BMCデータストリームのデコードから開始されます。

記号のエンコーディング

プリアンブルを除いて、記号はすべて以下の表に示す仕様に従って、4b5bスキームでエンコードされます。

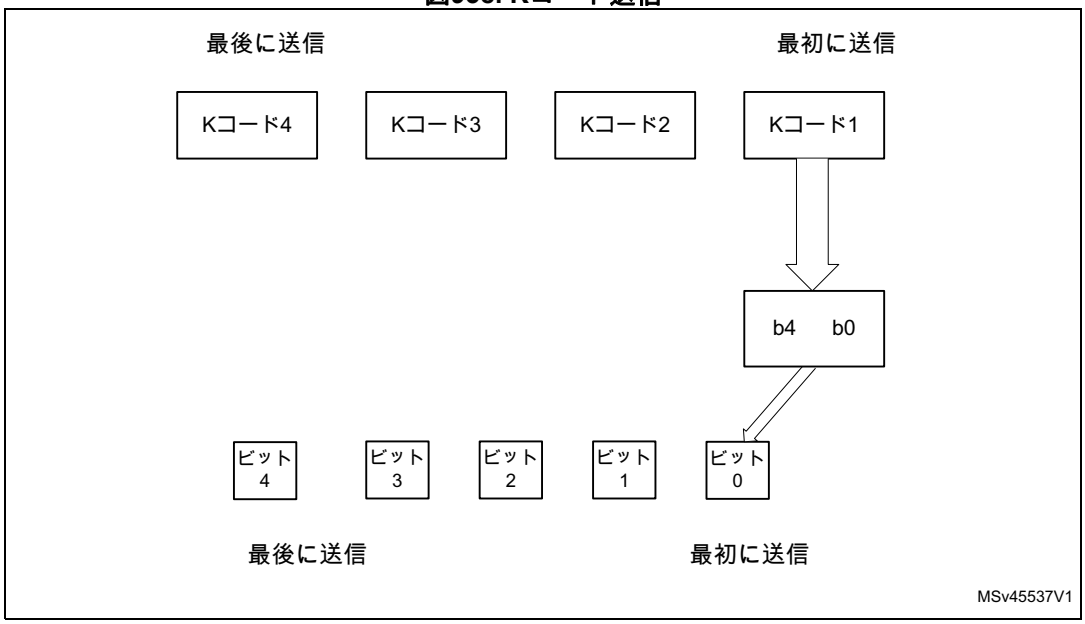
表 769. 4b5b記号エンコード表

名前	4b	5b	記号の説明
0	0000	11110	16進数データ0
1	0001	01001	16進数データ1
2	0010	10100	16進数データ2
3	0011	10101	16進数データ3
4	0100	01010	16進数データ4
5	0101	01011	16進数データ5
6	0110	01110	16進数データ6
7	0111	01111	16進数データ7
8	1000	10010	16進数データ8
9	1001	10011	16進数データ9
A	1010	10110	16進数データA
B	1011	10111	16進数データB
C	1100	11010	16進数データC
D	1101	11011	16進数データD
E	1110	11100	16進数データE
F	1111	11101	16進数データF
Sync-1	Kコード	11000	Startsynch #1
Sync-2	Kコード	10001	Startsynch #2
RST-1	Kコード	00111	Hard Reset #1
RST-2	Kコード	11001	Hard Reset #2
EOP	Kコード	01101	EOP
予約済みです。	誤差	00000	使用不可
予約済みです。	誤差	00001	使用不可
予約済みです。	誤差	00010	使用不可
予約済みです。	誤差	00011	使用不可
予約済みです。	誤差	00100	使用不可
予約済みです。	誤差	00101	使用不可
Sync-3	Kコード	00110	Startsynch #3
予約済みです。	誤差	01000	使用不可
予約済みです。	誤差	01100	使用不可
予約済みです。	誤差	10000	使用不可
予約済みです。	誤差	11111	使用不可

順序集合

順序集合は、次の図に示す4つのKコードから成ります。

図938. Kコード送信



以下の表に、発生しうるSOP* シーケンスを含め、定義済みの順序集合を示します。

物理層では、Hard Resetは他の順序集合よりも優先順位が高くなるため、進行中のTxメッセージに割込みが発生することがあります。

表 770. 順序集合

順序集合名	Kコード #1	Kコード #2	Kコード #3	Kコード #4
SOP	Sync-1	Sync-1	Sync-1	Sync-2
SOP'	Sync-1	Sync-1	Sync-3	Sync-3
SOP"	Sync-1	Sync-3	Sync-1	Sync-3
Hard Reset	RST-1	RST-1	RST-1	RST-2
Cable Reset	RST-1	Sync-1	RST-1	Sync-3
SOP'_Debug	Sync-1	RST-2	RST-2	Sync-3
SOP"_Debug	Sync-1	RST-2	Sync-3	Sync-2

受信時、物理層は、以下の表に示すように、4つのうちの3つの正しいKコードの任意の組み合わせにより順序集合を受け入れる必要があります。

表 771. 順序集合の検証

ステータス	1番目のコード	2番目のコード	3番目のコード	4番目のコード
有効	破壊	Kコード	Kコード	Kコード
有効	Kコード	破壊	Kコード	Kコード
有効	Kコード	Kコード	破壊	Kコード
有効	Kコード	Kコード	Kコード	破壊
有効（完全）	Kコード	Kコード	Kコード	Kコード
無効（例）	Kコード	破壊	Kコード	破壊

送信時のビット順

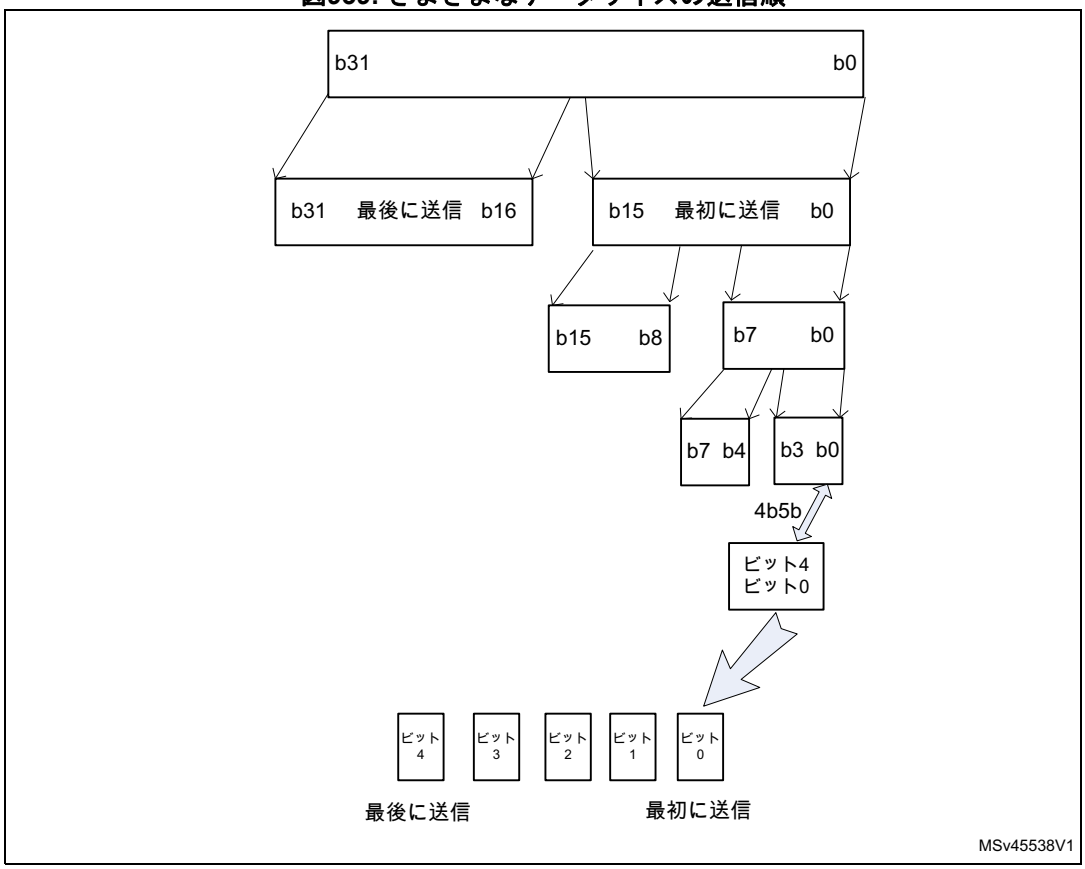
許容される送信データ単位/データサイズを以下の表に示します。

表 772. データサイズ

データユニット	エンコードされない場合	エンコードされる場合
バイト	8bit	10bit
ワード	16bit	20bit
ダブルワード	32bit	40bit

下図にビット送信順序を示します。

図939. さまざまなデータサイズの送信順

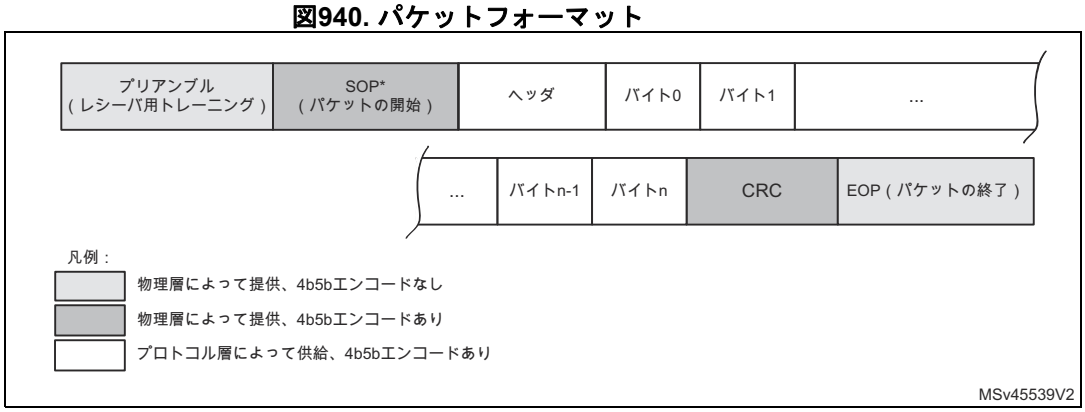


パケットフォーマット

Hard ResetとCable Reset以外のメッセージ

パケットフォーマットを、4b5bエンコードとデータソースに関する情報と一緒に下図に示します。

図940. パケットフォーマット



Hard Reset

物理層では、Hard Reset信号は、進行中の転送への割込みが行えるように高い優先順位を持っているため、他のタイプのメッセージとは違う方法で処理されます。

Txメッセージの送信中の場合、物理層の仕様は次のようなシーケンスとなります。

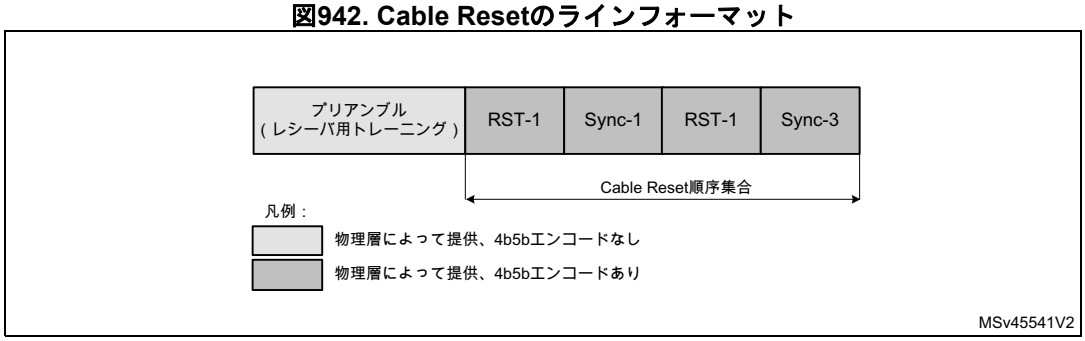
1. EOP Kコードを送信してメッセージを終了させ、メッセージの残りを破棄します。
2. *tInterFrameGap* 時間の間待ちます。
3. CCラインがアイドル状態でない場合、アイドル状態になるまで待ちます。
4. プリアンブル、Hard Reset信号の4個のKコードの順に送信します。
5. CCチャネルを無効にし（送受信を停止）、物理層をリセットし、プロトコル層に物理層がリセットされたことを通知します。
6. プロトコル層にリクエストされたときは、チャネルを再度有効にします。

図941. Hard Resetのラインフォーマット

Cable Reset

下図に示すCable ResetはHard Resetとフォーマットが似ていますが、Hard Resetとは違い、特に優先して処理する必要はありません。

図942. Cable Resetのラインフォーマット



衝突の回避

物理層では、Txメッセージの最後に送信されたビットの終了時から次のメッセージの最初のビットの間に発生する *tInterFrameGap* 遅延が考慮されます。

また、送信開始前にCCラインのアイドル状態についても確認されます。CCラインは、*tTransitionWindow*（12 ~ 20 µs）内での遷移が3回（*nTransitionCount*）未満の場合、アイドル状態とみなされます。Power Delivery仕様リビジョン3.1では、Rp値（ソース）を管理し、（シンクでの）これらのRp変更のType-C電圧レベルを監視する必要もあります。

物理層の信号方式

ビットはバイフェーズマークコーディング（BMC）により信号送信されます。

BIST

プロトコル層が要求するBISTアクションに応じて、以下のいずれかを実行できます。

- Tx BISTパターンテスト。TXMODEおよびTXSENDを書き込むことで達成されます。
- Rx BISTパターンテスト。RXBISTの正しい値にRXMODEを書き込むことで達成されます。

UCPDでは、（「BMC」モードに対応する）次の2つのパターンをサポートできます。

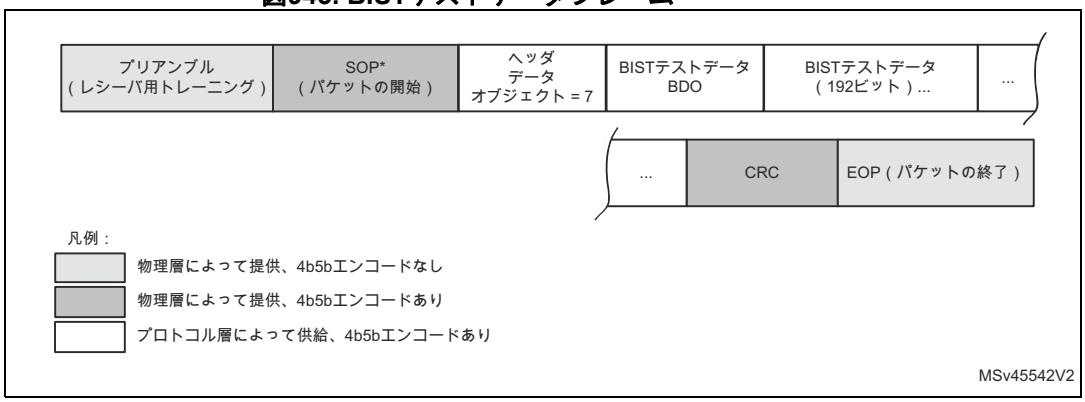
- BISTテストデータ（192bitパターン）。TxおよびRxに適用されます。Rxの場合は、メッセージを受信します（ただし、プロトコル層には渡されず破棄されます。その場合でも、確認応答としてGoodCRC Txメッセージが生成されます）。
- BISTキャリアモード2（単一パターン、無限長のメッセージ）。Txのみに適用されます。Txとは異なり、この状態のとき、このモードのレシーバは単にCCラインを無視します。

BISTテストデータパターン

テストデータパターンは、UCPDでは特殊なケースとみなされません。

BISTテストデータのパケットフレームフォーマットを下図に示します。

図943. BISTテストデータフレーム



固定長のテストデータパターンです。実際、図 940：パケットフォーマットに示されている通常のパケットフォーマットとの違いを表す点はヘッダの内容のみです。UCPDはTxヘッダの内容をプログラミングによって受信するため（単にペイロードの一部とみなされている）、BISTテストデータパケットから通常のパケットを区別するのはこのプログラミングのみ（ブロックの動作ではない）です。

BISTキャリアモード2

必要な場合は、このBISTテストモードから反復する1010の代替パターンが送信されます。このモードは信号分析用であるため、一定の長さを持たない安定した状態です（USB PD仕様のV1.0）。USB PD仕様のV1.1以降では、このモードを終了するタイミングを示すカウンタをプロトコル層で定義します。

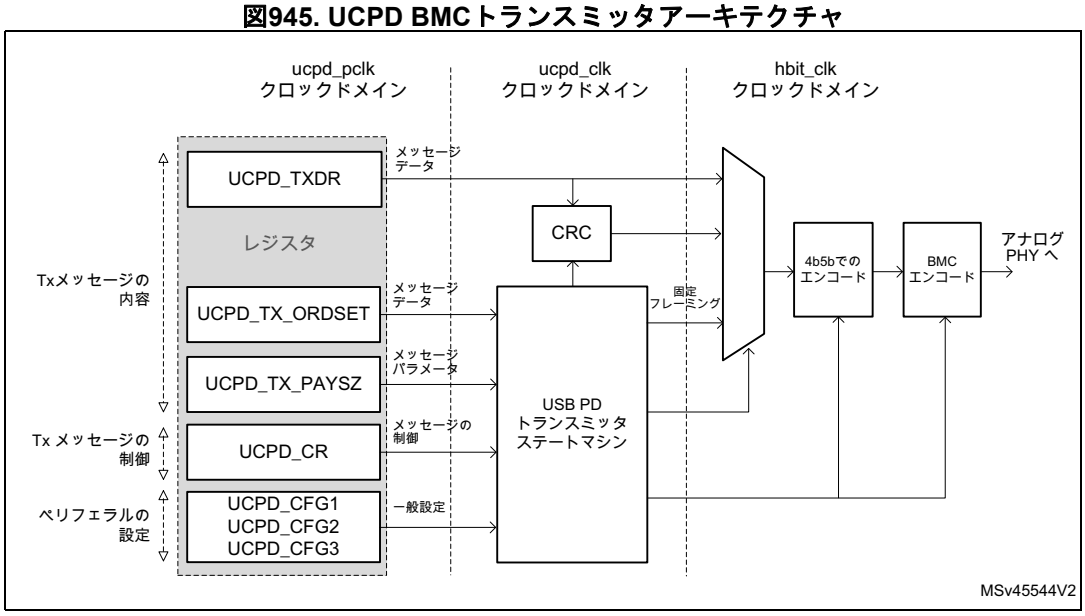
無限の1010シーケンスを停止するには（USB PD仕様の要件に従って）、UCPDENビットによりUCPDペリフェラルを無効化します。

図944. BISTキャリアモード2フレーム

74.4.4 UCPD BMCトランスミッタ

BMCトランスミッタでは、下図に示すように4b5bエンコード、CRC生成、BMCエンコードが行われます。その出力はチャネルスイッチを経由してアナログPHYに送られます。

図945. UCPD BMCトランスミッタアーキテクチャ



BMCエンコーダ

バイフェーズマークコーディングメソッドは、「IEC 60958-1デジタルオーディオインタフェースパート：1総合版3.0 2008-09 www.iec.ch」仕様で定義されています。

ハーフビットクロックhbit_clkは、UCPD_CFGR1レジスタのHBITCLKDIV[5:0] ビットフィールドによって制御されるシンプルな分周回路を通してucpd_clkから生成されます。これにより、ハイとローのハーフビット周期の同じ期間（立上りエッジと立下りエッジの異なる期間による、またはジッタによる小さな差を無視できる場合）、および同じビット期間（ジッタを無視できる場合）が保証されます。

トランスミッタのタイミングと衝突の回避

衝突の回避に対するハードウェアのサポートは、トランスミッタのハーフビット時間の機能の1つです。2つのカウンタが実装されています。

- *tInterFrameGap*：IFRGAP経由（事前定義値、変更可能）
- *tTransitionWindow*：TRANSWIN経由（事前定義値、変更可能）

この2つのカウンタを一度正しくセットすると、フレーム間ギャップを生成します。

トランスミッタにおけるHard Reset

Hard Resetの生成を容易にするために、TXMODEフィールドの特殊コードが使用されます。他のフィールドに書き込む必要はありません。

正しいコードを書き込むと、ハードウェアは正しい（最適な）タイミングで実行中のTxメッセージに強制的にHard Reset Txを適用します。このメッセージは（まだ進行中である場合）、現行のシーケンスを切り詰め、EOP Kコードシーケンスを直接付加することで、クリーンに終了します。特定の割込みがこの切詰めイベントに関連して発生することはありません。

エラー時のトランスミッタの動作

アンダーラン状態（TXUND割込み）が偶発的に発生する可能性があり、その場合、UCPDでは（正しい）Txペイロードが足りなくなってTxメッセージを正しく完成することができなくなる可能性があります。これは重大なエラーです（これが生じるということは、ソフトウェアが適時応答できていないということ）。結果として、ハードウェアはメッセージの最後でCRCが不正であることを確認し、これによりレシーバ側でメッセージを破棄することを保証します。

74.4.5 UCPD BMCレシーバ

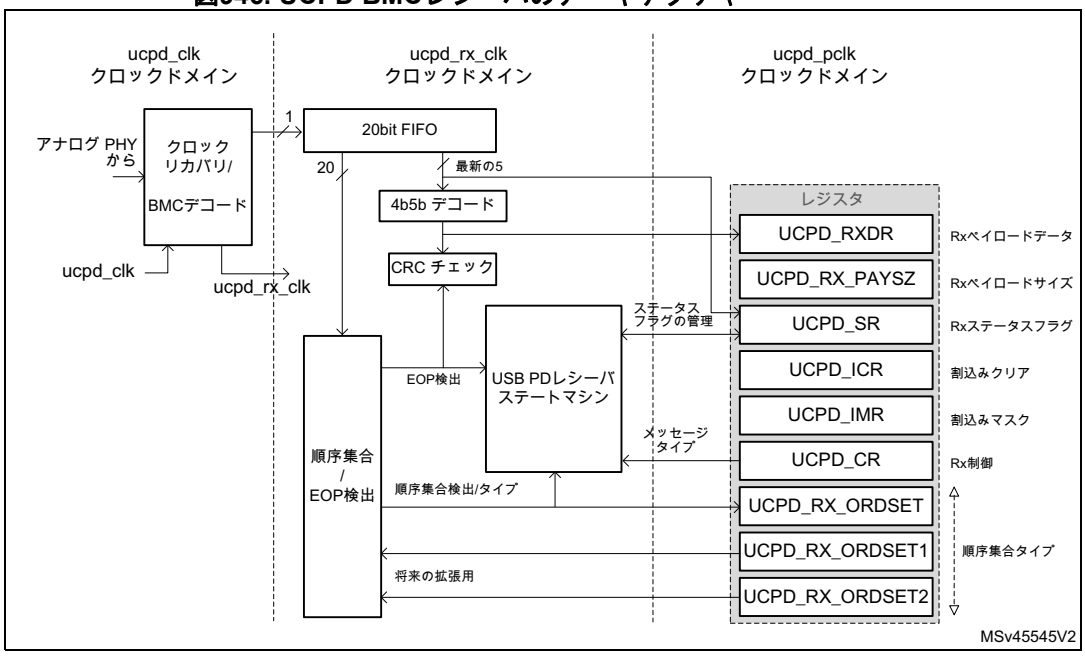
UCPD BMCレシーバの処理内容：

- クロックリカバリ
- プリアンブル検出／タイミング導出
- BMCデコード
- 4b5bでのデコード
- Kコードの順序集合の識別
- CRCチェック
- SOP検出
- EOP検出

レシーバはUCPDペリフェラルが（UCPDENを通じて）有効化されるとすぐに有効化されますが、メッセージの受信を開始する前にCCラインのアイドル状態を待ちます。

下図に、UCPD BMCレシーバのハイレベルアーキテクチャを示します。

図946. UCPD BMCレシーバのアーキテクチャ



CRCチェッカー

受信したビットはCRCチェッカーに供給され、受信したペイロードビットストリーム中に32bitの状態が進展します。最後には、CRCの32bitもロジックに供給されます。

EOP検出（5bit）はプロセスを停止し、固定残差状態のチェックを行ってペイロードを正しく受信したことを確認します（実際には残差は0xC704DD78です）。

この時点でUCPDによって割込み RXMSGEND が発生します。CRCが不正な場合は RXERR が真にセットされ、受信データは破棄する必要があります。

通常動作時には、この割込みは以前に認識され、クリアされています。そうでない場合は、別の割込み **RXOVR** が **RXMSGEND** の代わりに生成されます。

順序集合の検出

この機能では、それぞれ4つの5bit Kコードから成るさまざまな順序集合を検出します。

プリアンブルに入ったら、順序集合のスライディングウィンドウ検出を開始します（5bitの4つのワード）。

検出された順序集合にはすべてのSOP* コード（SOP、SOP'、およびSOP''）が含まれますが、Hard Reset、Cable Reset、SOP'_Debug、SOP''_Debugのほか、UCPD_RX_ORDEXT1およびUCPD_RX_ORDEXT2レジスタで定義された2つの拡張機能も含まれます。

EOP検出およびHard Resetの例外処理

EOPはメッセージの終了を表す固定の5bit Kコードです。

トランスミッタでHard Resetの送信が必要となるのは（前回のメッセージの送信がまだ進行中の場合に）、この前回のメッセージがEOPによって早めに切捨てられているためです。

Hard Resetが無視された場合、EOP検出は予想された時間でのみ実行できます。ただし、このHard Reset問題により、EOP検出はRxメッセージの着信中はアクティブでなければなりません。「早期EOP」が検出されると、切捨てられたRxメッセージはすぐに破棄されます。

切捨てられたメッセージまたは破損したメッセージの例外

順序集合が検出されると、メッセージによっては、CRCとEOPで完成するデータバイトを受信する必要がある場合があります。これらのフェーズのどこかで、次のようなエラー条件が発生します。

- 「1 UI」を十分上回る時間の間、ラインが静的になります（この条件の正確な閾値は重要ではありませんが、例外は3 UIより前に発生する必要があります）。または
- メッセージは終了しますが、認識されません（例：EOPの破損）。

どちらの場合でも、レシーバはRXMSGENDおよびRXERRフラグを立ち上げ、現在のメッセージを停止します。

短いプリアンブルまたは未完成の順序集合の例外

レシーバが確認する予期したプリアンブルが半分に満たない例外的な事例では、正しいBMCデコードを可能にする周波数の概算を算出できません。プリアンブル全体を確認している場合でも、周波数の概算は可能ですが、ラインが静的になる前にすべての順序集合を受信できず、レシーバステートマシンは開始されません。

どちらの場合でもクロックリカバリ/BMCデコーダは再開し、最初にIDLE条件、次にプリアンブルを確認し、続いて周波数を概算します。

74.4.6 UCPD Type-Cのプルアップ（Rp）およびプルダウン（Rd）

UCPDではこれらのレジスタをANAMODEおよびANASUBMODE[1:0] によって簡単に制御できます。使用するCCラインが1つのみである場合、CCENABLE[1:0] ビットフィールドを使用して他のラインの制御を無効にすることで、消費電力を最適化することができます。

MCUが電源オフの場合でも、UCPDx_DBCC1ピンとUCPDx_DBCC2ピンが、それぞれUCPDx_CC1ピンとUCPDx_CC2ピンに接続されている場合は、「デッドバッテリ」Rdが示されます。

デッドバッテリ動作が必要ない場合は（ソースのみを対象とした製品など）、UCPDx_DBCC1ピンとUCPDx_DBCC2ピンの両方をアース接続してください。

電力が供給され、MCUが起動されたら、ANAMODEとANASUBMODE[1:0] に望ましい動作（ソースなど）をプログラムする必要があります。電源が投入されMCUがブートした後に期待する動作（例えばソースなど）にするには、PWR_CR3レジスタのUCPD_DBDISビットをセットしてデッドバッテリ用プルダウン抵抗を無効にするより前に、ANAMODEおよびANASUBMODE[1:0]をプログラムしてください。

切断されている状態のシンクの場合は、STANDBY低電力モードを使用できます。

74.4.7 UCPD Type-C電圧監視とデバウンス

Type-Cステートマシンを正しく動作させ、ケーブル方向を検出するには、ピークなどの高速イベントを無視しながら、CC1/2ラインの電圧レベルを監視する必要があります。

CC1/2ラインの電圧レベル間の閾値は、PHY閾値検出設定に基づいて決まります。

TYPEC_VSTATE_CC1/2[1:0] ビットフィールドは、ピークなどの高速ラインイベントを抑制するハードウェアデバウンスフィルタにより処理されるCC1/2ラインレベルを反映します。PHYCCSELビットでは、Power Delivery信号に使用されるCC1またはCC2ラインを選択します。

消費電力を最小限に抑えるため、Type-C検出器を常にオンにしておくのではなく、ポーリングの瞬間だけType-C検出器をオンにするポーリング方法を使用し、CC1/2ラインイベント時にデバイスをSTOPモードからウェイクアップさせることを推奨します。

74.4.8 UCPDファストロールスワップ（FRS）

FRS信号

FRS接続（特定の長さのパルス）は、FRSTXビットのセット時に生成されます。

FRS接続の期間、UCPD_FRSTX1（または2）として設定された現在アクティブな I/O（オルタネート機能）は、アクティブCCラインをプルダウンする外部NMOSトランジスタのゲートをハイレベルで制御します。

FRS検出

FRS監視を有効にするには、検出されたケーブル方向に基づいてアクティブCCラインを選択するPHYCCSELを書き込んだ後、ビットFRSRXENをセットします。

74.4.9 UCPD DMAインタフェース

DMAはUCPDに実装され、有効にすると、UCPD1_TXDRおよびUCPD1_RXDRレジスタ（TxとRxのデータレジスタ、それぞれ1バイト）を処理するバイトレベルの割込みは不要になります。

TXDMAENやRXDMAENビットを有効にすることで、TxやRxの機能に対して個別にDMAを有効にできます。

74.4.10 STOPモードからのウェイクアップ

消費電力を最適化するには、STOPモードを使用して、CCラインでのイベントを待ってMCUをウェイクアップできると便利です。

これが機能するためには、最初にWUPENに1を書き込んで有効にする必要があります。

ウェイクアップを引き起こすイベントは次のとおりです。

- BMCレシーバのイベント（RXORDDET、RXHRSTDET）、ハードウェアによるPHYRXENの有効化
- FRS検出器のイベント（FRSEVT）、ハードウェアによるFRSRXENの有効化
- Type-C 検出器のイベント（TYPECEVT1、TYPECEVT2）、ハードウェアによる CC1TCDIS、CC2TCDISの有効化

74.5 UCPDプログラミングシーケンス

UCPDユニットを使用する正常なシーケンスは次のとおりです。

1. UCPDを設定します。
2. UCPDを有効化します。
3. 同時に
   - プロトコル層からの要求に応じて、Txメッセージを送信します。
   - 関連するRxメッセージを取得し（ポーリングまたは割込みを待ち）、プロトコル層に引き渡す詳細を復元します。

最後の項目を無限に繰り返します。

74.5.1 初期化フェーズ

次のシーケンスを使用して、クリーンに起動します。

1. UCPD_CFGレジスタを書き込んで、すべての初期クロック分周値を準備します。
2. UCPDENビットをセットすることによって、ユニットを有効にします。
3. UCPD_CFGR2レジスタのRXAFILTENビットを使用して、各CCラインのアナログRxフィルタを有効にします。

74.5.2 Type-Cステートマシンの処理

ソース、シンク、デュアルロールポート（ソースとシンクの代替となる最後のもの）の一般的な用途では、ソフトウェアにより該当するUSB Type-Cステートマシンを実装する必要があります。基本的なコードを以下の表に示します。

表 773. ANAMODEとANASUBMODEのコーディングおよびTYPEC_VSTATE_CCxとのリンク

ANAMODE	ANASUBMODE[1:0]	注	TYPEC_VSTATE_CCx[1:0]			
			00	01	10	11
0：ソース	00：無効	無効	N/A			
	01：デフォルトUSB Rp	-	vRa[Def]	vRd[Def]	vOPEN[Def]	N/A
	10：1.5A Rp	-	vRa[1.5]	vRd[1.5]	vOPEN[1.5]	
	11：3.0A Rp	-	vRa[3.0]	vRd[3.0]	vOPEN[3.0]	
1：シンク	xx	-	vRa	vRd-USB	vRd-1.5	vRd-3.0

CCENABLE[1:0] ビットフィールドにより、CCラインの1つでプルアップ/プルダウンを無効にできます。

注： **Type-Cステートマシンは CC ピンのレベルだけでなく、VBUSの存在検出（シンクモード）にも依存します。ソースモードの場合は、VCONN生成とVBUS状態（オン／オフ／電圧レベル、放電）を決定します。UCPDはVBUSの生成回路やVCONN負荷スイッチ（CCラインに接続するVCONN供給ジェネレータを有効化）を直接制御しませんが、アプリケーションでは正常に機能するためにこれらの入力や制御を必要とします。**

一般的なプログラミングシーケンス（UCPDを設定し、有効化）

1. ANAMODEとANASUBMODE[1:0] をUSB Type-Cステートマシンの現在の位置に（ソースの場合は現在の通知にも）基づいてセットします。これにより、CCラインの適切なプルアップ/プルダウンがオンになり、TYPEC_VSTATEフィールドが表す電圧レベルが定義されます。プログラミングの前には、PHYが事実上オフである点に注意してください。
2. TYPEC_VSTATE_CC1/2 を読み出して、Type-C の初期状態を決定します（ローカルソースがリモートシンクに接続されているかどうかなど）。
3. 接続されていない場合は、接続イベントを待ちます。
4. 接続が検出され、ローカルのPower Delivery機能が実装されたと仮定して、Power Deliveryのメッセージの送受信を開始します。
5. 安定した電圧の変更を示すPHYEVT1/2に対して新しい割込み／イベントが発生した場合は、影響を再評価して、この入力内容をType-Cステートマシンに通知します。

設定可能な3つのRp値（Default-USB / 1.5A / 3.0A）のいずれかに変更する必要のあるソースと、これに接続されたシンクの例

- [ソース] 単純にANASUBMODE[1:0] を再プログラムします。
- [その時点からのシンクの動作] PHYEVT1/2が発生し、TYPEC_VSTATE1/2がこれに合わせて変化します。

ソースからシンクにトグルするデュアルロールポート（DRP）のプログラミング：

- 新しい動作を開始するために、単純にANAMODEとANASUBMODE[1:0] を再プログラムします。

詳細なプログラミングシーケンス（例）：

表 774. Type-Cシーケンス（転送元：3A）、ケーブル／シンク接続済み（CC1ではRd、CC2ではRa）

<table>
<tr><th>Type-C ステート</th><th>ANAMODE、ANASUBMODE[1:0]</th><th>CCENABLE</th><th>PHYCCSEL</th><th>RDCH</th><th>CC[x] VCONNEN (1)</th><th>イベント => 次の行に移動</th><th>コメント</th></tr>
<tr><td>切断。
SRC</td><td rowspan="2">0：転送元、
11：Rp3A0</td><td rowspan="2">11：両方有効</td><td>0（無視）</td><td rowspan="8">0：
[通常]</td><td rowspan="2">00：
[どちらでもない]</td><td>PHYEVT1：
[VRd-3A0]</td><td>シンク接続検出待ち、CC1 [EVT1] に出現</td></tr>
<tr><td>接続待ち。
SRC</td><td rowspan="6">0
[CC1はRd]</td><td>PHYEVT2：
[VRa]</td><td>接続待ち開始（100～200ms）、Ra => VCONNのリクエストも参照</td></tr>
<tr><td rowspan="4">接続。
SRC
[VCONN => CC2]</td><td>0：転送元、
11：Rp3A0
[SinkTxOK]</td><td rowspan="4">01：CC2は無効です。
（外部VCONNスイッチにより可能／推奨）</td><td rowspan="4">10：
[CC2アクティブ]</td><td>タイマ（100ms）、PHYEVTxなし</td><td>PHYから切断されたローカルのCC2（VCONNスイッチはVCONNソースを外部的にCC2に接続）
PHYEVT1の監視を継続</td></tr>
<tr><td>0：転送元、
10：Rp1A5
[SinkTxNG]</td><td rowspan="2">SWタイマ（SinkTxNG）</td><td>ソースがメッセージシーケンスの初期化を希望（SinkTxNG条件を最初にセット）</td></tr>
<tr><td rowspan="3">0：転送元、
11：Rp3A0
[SinkTxOK]</td><td>ソースがメッセージシーケンスを終了（その後SinkTxOK条件を設定）</td></tr>
<tr><td>PHYEVT1：
[VOpen-3A0]</td><td>シンクの切断待ち（CC1はVopen）</td></tr>
<tr><td>切断待ち。
SRC</td><td rowspan="2">11：両方有効</td><td rowspan="2">0：（無視）</td><td>1：
[放電]</td><td rowspan="2">00：
[どちらでもない]</td><td>> 0.8V検出</td><td>能動的にVCONN [CC2] を放電 [Rdch]、< 0.8Vまで</td></tr>
<tr><td>切断。
SRC</td><td>0：転送元、
11：Rp3A0</td><td>0：
[通常]</td><td></td><td>[詳細は表の最初の行を参照]</td></tr>
</table>

1. 外部負荷スイッチコンポーネントを通してVCONNを有効化する2つのGPIO

74.5.3 USB PDの送信

プロトコル層のメッセージの受信時（つまり、送信）、UCPD_TX_ORDSETおよびUCPD_TX_PAYSZレジスタを書き込むことによって、Txメッセージコンテンツを準備します。

メッセージ送信をトリガするには、TXMODEビットフィールドの適切な値でTXSENDビットをセットします。

データバイトの送信時は、UCPD_TXDRレジスタへの新しいデータ書込みをリクエストするため、TXISフラグが立てられます。

データのペイロード全体が送信されるまで、これが繰り返されます。

CRCパケットの送信時、TXMSGSENTフラグがセットされ、メッセージ送信の完了が示されます。

Hard Resetの送信

Hard Resetを送信する必要があるとわかったとき、すぐにUCPD_CRレジスタのTXHRSTビットをセットすると、内部ステートマシンで強制的に正しいシーケンスを生成できます。UCPD_TX_ORDSETの値は、この特定のケースでは更新は不要です（Hard Resetの正しいコードはUCPDによって送信されます）。

USB Power Delivery仕様では、メッセージ送信中にHard Resetを優先することを規定しています。この場合、UCPDでは現在のメッセージのペイロードを切捨てるなどして、最後にEOPを付加します。レジスタによる通知は利用できません（TXMSGSENDフラグによる通知など）。これは、Hard Resetが以前のすべてのアクティビティに優先されるという事実によって裏付けられています（このため、完了したかどうかを知ることは重要ではなくなります）。

送信用のDMAの使用

送信についてDMA（Direct Memory Access）を有効にするには、UCPD_CRレジスタのTXDMAENビットをセットします。

各メッセージに対して：

- メッセージ全体をメモリ内で準備します（2つのヘッダバイトから始まります）。
- 2つのヘッダバイトに対応する長さと、データワードの数の4倍に対応する多数のデータバイトにより、DMAの動作をプログラムします。
- メッセージ転送を開始するには、TXSENDを書き込みます。
- TXMSGDISCの場合は、前のラインに戻ります（TXSEND）。
- DMA転送完了割込みを待ちます（つまり、UCPDへの最後のTxバイトの書込み）。
- 後続のTXMSGSENT割込みが発生していないかダブルチェックします。

74.5.4 USB PDの受信

受信メッセージシーケンスの開始の通知は、UCPD_SR（ビットRXORDDET）の割込みによってトリガされます。

情報は、以下の読出しによって復旧されます。

- UCPD_RX_SOP（割込みRXORDDET）
- UCPD_RXDR（割込みRXNE、バイトごとに繰り返される）
- UCPD_RXPAYSZ（割込みRXMSGEND）

上記のUCPD_RXDRから前に読み出したデータは、RXERRフラグがセットされている場合、この時点で破棄する必要があります。

CRCが有効な場合、受信データはプロトコル層に転送されます。

デバッグのため、受信した不正なKコードの数に関する統計を追跡することが望ましいことがあります（仕様で定義されているとおり、Kコードの3/4が有効である場合にのみ実行します）。これは、以下を使用して容易に行えます。

- 少なくとも1つの無効なKコードがあることを示すRXSOP3OF4bit
- 順序集合内の無効なKコードの順序を特定するRXSOPKINVALIDビットフィールド

受信用のDMAの使用

受信についてDMA（Direct Memory Access）を有効にするには、UCPD_CRレジスタのRXDMAENビットをセットします。

Rxメッセージを受信する可能性があるときには、以下を実行します。

- DMA 受信動作をプログラムします（また、最大有効メッセージより少し長くしてバッファを割きます）。長さは拡張メッセージサポートに依存します。
- RXORDDETを受信した後、DMAは、バックグラウンドで動作を開始します。
- RXMSGEND割込みの受信時にRXPAYSZを読み出します。
- RXPAYSZをDMA Rxのバイト数に対してダブルチェックします（一致するはずですが、RXDRのDMA読出しはRXDRが最後のバイトを取得した少しあとになります）。
- DMA Rxバッファを処理します。
- できるだけ速やかに次のRx DMAバッファを用意して備えます。

74.5.5 UCPDソフトウェアのトリミング

CCプルアップ（Rp）およびプルダウン（Rd）デバイスは、必要な精度を得るために各部をトリミングする必要があります。トリミング値は不揮発性メモリに保存されます。

ソフトウェアによってCCプルアップおよびプルダウンデバイスをトリミングするには、次の手順を実行します。

1. 不揮発性メモリからTRIM値を取得します（表 766：UCPDソフトウェアのtrimデータを参照）。
2. 初期化時に、次のようにUCPD_CFGR3レジスタのビットフィールドにTRIM値を書き込みます。
   - TRIM_CC1_RP[3:0] に3A0_CC1[3:0]
   - TRIM_CC2_RP[3:0] に3A0_CC2[3:0]
   - TRIM_CC1_RD[3:0] にRd_CC1[3:0]
   - TRIM_CC2_RD[3:0] にRd_CC2[3:0]
3. 各ANASUBMODEを1A5または3A0にセットし、UCPD_CFGR3レジスタビットフィールドに次のようにトリミング値を書き込みます。
   - TRIM_CC1_RP[3:0] にそれぞれ1A5_CC1[3:0] または3A0_CC1[3:0]
   - TRIM_CC2_RP[3:0] にそれぞれ1A5_CC2[3:0] または3A0_CC2[3:0]

74.6 UCPD低消費電力モード

低電力モードの概要を以下の表 775：低電力モードがUCPDに与える影響に示します。

表 775. 低電力モードがUCPDに与える影響

モード	説明
SLEEP	影響なし。
STOP	イベントの検出（Type-C、BMC Rx、FRS検出）は動作を続行し、 MCUをウェイクアップすることができます。
STANDBY	UCPDは動作しておらず、MCUをウェイクアップすることはできません。 プルダウンは、設定されている場合はアクティブなままです。
電源オフ	デッドバッテリプルダウンはアクティブなままです。

UCPDは、次のいずれかの対応するイベントを認識したときに、MCUをSTOPモードからウェイクアップできます。

- TYPEC_VSTATE_CCxに表示される、CCラインの電圧範囲の変化に関連するType-Cイベント
- RXORDSETを読み出すことで表示される、順序集合がRXORDSETEN[8:0] に応じてフィルタリングされた順序集合と一致するPower Delivery受信メッセージ

STOPモードからのウェイクアップを有効にするには、UCPD_CFG2レジスタのWUPENビットをセットします。

UCPDレベルでは、STOPモード中にカーネルクロックアクティビティを必要とする次の3つのタイプのイベントが発生する場合があります。

- 後でType-C仕様で定義された電圧範囲間の一定の変化であると確認できるアナログPHY電圧閾値検出器のアクティビティ
- 後でRxメッセージイベント（RXORDSET）を生成可能なPower Delivery BMCレシーバのアクティビティ（選択したCCラインから入力される）
- 後でFRS信号検出イベント（FRSEVT）を生成可能なPower Delivery FRS検出器のアクティビティ

以下の非同期アクティビティがある場合、RCCとともに正常に機能するために、クロックリクエスト信号が有効化されます（WUPENの条件付き）。

- Type-C電圧閾値検出器（いずれかのCCラインから入力される）
- Power Deliveryレシーバ信号（選択したCCラインから入力される）
- FRS検出信号（選択したCCラインから入力される）

74.7 UCPD割込み

以下の表に、UCPDイベントフラグのほか、関連するフラグクリアビットと割込みイネーブルビットを示します。

表 776. UCPD割込みリクエスト

割込みイベント	イベントフラグ	イベントフラグ／割込みのクリア方法	割込み有効制御ビット
FRS検出	FRSEVT	FRSEVTCFをセット	FRSEVTIE
CC2でのType-Cの電圧レベル変化	TYPECEVT2	TYPECEVT2CFをセット	TYPECEVT2IE
CC1でのType-Cの電圧レベル変化	TYPECEVT1	TYPECEVT1CFをセット	TYPECEVT1IE
Rxメッセージ受信	RXMSGEND	RXMSGENDCFをセット	RXMSGENDIE
Rxデータオーバーフロー	RXOVR	RXOVRCFをセット	RXOVR
Rx Hard Reset検出済み	RXHRSTDET	RXHRSTDETCFをセット	RXHRSTDETIE
Rx順序集合（4つのKコード）検出済み	RXORDDET	RXORDDETCFをセット	RXORDDETIE
受信データレジスタノットエンプティ	RXNE	UCPD_RXDRのデータを読み出す	RXNEIE
Txデータアンダーラン	TXUND	TXUNDCFをセット	TXUNDIE
Hard Reset送信済み	HRSTSENT	HRSTSENTCFをセット	HRSTSENTIE
Hard Reset破棄済み	HRSTDISC	HRSTDISCCFをセット	HRSTDISCIE
送信メッセージがアボート	TXMSGABT	TXMSGABTCFをセット	TXMSGABTIE
送信メッセージ送信済み	TXMSGSENT	TXMSGSENTCFをセット	TXMSGSENTIE
送信メッセージ破棄済み	TXMSGDISC	TXMSGDISCCFをセット	TXMSGDISCIE
送信データが必要	TXIS	UCPD_TXDRレジスタにデータを書き込む	TXISIE

UCPDからの割込みが受信された場合、ソフトウェアによって、UCPD_SRレジスタを読み出すことで、割込みの転送元を確認する必要があります。

どのビットが1であるかによって、ISRはこの条件を処理し、UCPD_ICRレジスタの適切なビットに書き込んでこのビットをクリアする必要があります。

74.8 UCPD レジスタ

74.8.1 UCPD設定レジスタ1（UCPD_CFGR1）

アドレスオフセット：0x000

リセット値：0x0000 0000

ペリフェラルの一般設定このレジスタへの書込みが可能なのは、UCPDが無効（UCPDEN = 0）の場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UCPDEN	RXDMAEN	TXDMAEN	RXORDSETEN[8:0]									PSC_USBPDCLK[2:0]			Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRANSWIN[4:0]					IFRGAP[4:0]					HBITCLKDIV[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31 **UCPDEN**：UCPDペリフェラルの有効化

UCPDペリフェラルの全般的有効化

0：無効

1：有効

無効化すると、ペリフェラルではすぐに実行中のアクティビティが停止し、制御ビットとビットフィールドはすべてデフォルトでリセット値になります。ペリフェラルが無効状態から有効状態に遷移するたびに、望ましい値をセットする必要があります。

ビット30 **RXDMAEN**：受信DMAモードの有効化

このビットをセットすると、DMAモードの受信が有効になります。

0：無効

1：有効

ビット29 **TXDMAEN**：送信DMAモードの有効化

このビットをセットすると、DMAモードの送信が有効になります。

0：無効

1：有効

ビット28:20 **RXORDSETEN[8:0]**：レシーバの順序集合の有効化

このビットフィールドで、レシーバが検出する順序集合のタイプを規定します。セット/クリアすると、各ビットにより特定の機能が有効化/無効化されます。

0bXXXXXXXX1：SOP検出は有効です。

0bXXXXXXX1X：SOP' 検出は有効です。

0bXXXXXX1XX：SOP" 検出は有効です。

0bXXXXX1XXX：Hard Reset検出は有効です。

0bXXXX1XXXX：Cable Detectリセットは有効です。

0bXXX1XXXXX：SOP'_Debugは有効です。

0bXX1XXXXXX：SOP"_Debugは有効です。

0bX1XXXXXXX：SOP拡張 #1は有効です。

0b1XXXXXXXX：SOP拡張 #2は有効です。

ビット19:17 **PSC_USBPDCLK[2:0]**：ucpd_clk生成のためのプリスケーラの分周比

このビットフィールドで、UCPDペリフェラルクロックを生成するカーネルクロックのプリスケーラの分周比を規定します（ucpd_clk）。

0x0：1（バイパス）
0x1：2
0x2：4
0x3：8
0x4：16

プリスケーラの使用では、ucpd_clk周波数を6 ~ 9 MHzの範囲でセットできるようにすることを推奨します。

ビット16 予約済みであり、リセット値に保持する必要があります。

ビット15:11 **TRANSWIN[4:0]**：トランザクションウィンドウ継続時間

このビットフィールドで、*tTransitionWindow* 間隔を生成するhbit_clk分周回路の分周比（ビットフィールド値 - 1）を規定します。

0x00：サポートされません。
0x01：2
0x09：10（推奨）
0x1F：32

ucpd_clk周波数とHBITCLKDIV[5:0] ビットフィールドの設定を考慮して、12 ~ 20 usの間隔を生成する値をセットします。

ビット10:6 **IFRGAP[4:0]**：フレーム間ギャップタイマクロック生成のための分周比

このビットフィールドで、フレーム間ギャップタイマクロック（*tInterFrameGap*）を生成するucpd_clk分周回路の分周比（ビットフィールド値 - 1）を規定します。

0x00：サポートされません。
0x01：2
0x0D：14
0x0E：15
0x0F：16
0x1F：32

分周比15はUSB PD 2.0仕様の正常値のTxクロックが対象となります。15未満の分周比は正常未満のTxクロックが対象となり、15を超える分周比は正常範囲を超えるTxクロックが対象となります。

ビット5:0 **HBITCLKDIV[5:0]**：ハーフビットクロック生成のための分周比

このビットフィールドで、ハーフビットクロック（hbit_clk）を生成するucpd_clk分周回路の分周比（ビットフィールド値 - 1）を規定します。

0x00：1（バイパス）
0x1A：27
0x3F：64

74.8.2 UCPD設定レジスタ2（UCPD_CFGR2）

アドレスオフセット：0x004

リセット値：0x0000 0000

UCPD Rx信号のフィルタリングの設定。このレジスタへの書込みが可能なのは、UCPDが無効（UCPDEN = 0）の場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXAFILTEN	Res.	Res.	Res.	Res.	WUPEN	FORCECLK	RXFILT2N3	RXFILTDIS
							rw					rw	rw	rw	rw

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **RXAFILTEN**：Rxアナログフィルタの有効化

このビットをセットすると、最適なPower Deliveryの受信に必要なRxアナログフィルタが有効になります。

0：無効

1：有効

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **WUPEN**：STOPモードからのウェイクアップの有効化

このビットをセットすると、UCPD_ASYNC_INT信号が有効になります。

0：無効

1：有効

ビット2 **FORCECLK**：ClkReqクロックリクエストの強制適用

0：クロックリクエストを強制適用しません。

1：クロックリクエストを強制適用します。

ビット1 **RXFILT2N3**：BMCデコーダのRxプレフィルタのサンプリング方法

新しい値を確認応答する前の適合連続サンプルの数

0：3サンプル

1：2サンプル

ビット0 **RXFILTDIS**：BMCデコーダのRxプレフィルタの有効化

0：有効

1：無効

このサンプリングクロックはレシーバのものです（つまり、プリスケーラ後）。

74.8.3 UCPD設定レジスタ3（UCPD_CFGR3）

アドレスオフセット：0x008

リセット値：0x0000 0000

CCプルアップおよびプルダウンデバイスのUCPDトリミングの設定。トリミングは、ソフトウェアにより最初にこのレジスタに書き込まれるまで、ハードウェアによって管理されます。

レジスタは全自動のトリミングをサポートするデバイス用に予約されています（書き込むこととはできません）。表 765：UCPDの実装を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	TRIM_CC2_RP [3:0]				Res.	Res.	Res.	Res.	Res.	TRIM_CC2_RD [3:0]			
			rw	rw	rw	rw						rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIM_CC1_RP [3:0]				Res.	Res.	Res.	Res.	Res.	TRIM_CC1_RD [3:0]			
			rw	rw	rw	rw						rw	rw	rw	rw

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:25 **TRIM_CC2_RP[3:0]**：CC2ライン上のRp電流ソース用のSW TRIM値

ビット24:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **TRIM_CC2_RD[3:0]**：CC2ライン上のRd抵抗器用のSW TRIM値

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12:9 **TRIM_CC1_RP[3:0]**：CC1ライン上のRp電流ソース用のSW TRIM値

ビット8:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **TRIM_CC1_RD[3:0]**：CC1ライン上のRd抵抗器用のSW TRIM値

74.8.4 UCPD制御レジスタ（UCPD_CR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2TCDIS	CC1TCDIS	Res.	RDCH	FRSTX	FRSRXEN
										rw	rw		rw	rs	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CCENABLE[1:0]		ANAMODE	ANASUBMODE[1:0]		PHYCCSEL	PHYRXEN	RXMODE	TXHRST	TXSEND	TXMODE[1:0]	
				rw	rw	rw	rw	rw	rw	rw	rw	rs	rs	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **CC2TCDIS**：CC2 Type-C検出器の無効化

このビットで、CC2ライン上でType-C検出器を無効化します。

0：有効

1：無効

有効な場合、CC2用のType-C検出器はANAMODEとANASUBMODE[1:0] を使用して設定されます。

ビット20 **CC1TCDIS**：CC1 Type-C検出器の無効化

このビットで、CC1ライン上でType-C検出器を無効化します。

0：有効

1：無効

有効な場合、CC1用のType-C検出器はANAMODEとANASUBMODE[1:0] を使用して設定されます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18 **RDCH**:：Rdch条件の実行

このビットは、ソース専用の UnattachedWait.SRC 状態中に、Type-Cステートを考慮して、残りのセットにより、PHYCCSELビット（そのため、VCONNに関連）を通して選択されたCCラインのRdch条件を実行します。「USB Type-C ECN for Source VCONN Discharge」を参照してください。CCENABLE[1:0]ビットフィールドもそれに合わせてセットする必要があります。

0：影響なし

1：Rdch条件の実行

ビット17 **FRSTX**：FRS Tx信号の有効化

このビットをセットすると、FRS Tx信号が有効になります。

0：影響なし

1：有効

このビットは、USB Power Delivery仕様リビジョン3.1を考慮した遅延後、ハードウェアによってクリアされます。

ビット16 **FRSRXEN**：FRSイベント検出の有効化

このビットをセットすると、PHYCCSELビットを使用して選択されたCCラインでのFRS Rxイベント（FRSEVT）検出が有効になります。0：無効

1：有効

デバイスがFRS不対応ソース/シンクに設置されている場合は、このビットをクリアします。

ビット15 予約済みであり、リセット値に保持する必要があります。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **CCENABLE[1:0]**：CCラインの有効化

このビットフィールドでは、ANAMODEとANASUBMODE[1:0] の設定に従ってCC1およびCC2ラインのアナログPHY（プルアップおよびプルダウン）を有効化します。

0x0：両方のPHYを無効化

0x1：CC1 PHYを有効化

0x2：CC2 PHYを有効化

0x3：CC1とCC2のPHYを有効化

単一ラインPHYを有効化できるのは、外部VCONNスイッチを使用して他のラインがVCONNによって駆動されている場合などです。両方のPHYを有効化するのは、シンク/ソースの通常の使用法です。

ビット9 **ANAMODE**：アナログPHYの動作モード

0：ソース

1：シンク

CC1とCC2の動作は、CCENABLEによって異なります。ANASUBMODE[1:0] とこのビットフィールドを一緒に使用した場合の影響については、表 773 : ANAMODEとANASUBMODEのコーディングおよびTYPEC_VSTATE_CCxとのリンクを参照してください。

ビット8:7 **ANASUBMODE[1:0]**：アナログPHYのサブモード

このビットフィールドの効果については、表 773 : ANAMODEとANASUBMODEのコーディングおよびTYPEC_VSTATE_CCxとのリンクを参照してください。

ビット6 **PHYCCSEL**：USB Power Delivery信号に使用するCC1/CC2ラインの選択

0：Power Delivery通信でCC1 IOを使用します。
1：Power Delivery通信でCC2 IOを使用します。
選択は、設置時に検知されるケーブルの方向によって決まります。

ビット5 **PHYRXEN**：USB Power Deliveryレシーバの有効化

0：無効
1：有効
このビットをクリアすると、CC1とCC2の両方のレシーバが無効になります。このビットをセットすると、PHYCCSELビットを使用して選択したCCレシーバのみが有効になります。

ビット4 **RXMODE**：レシーバのモード

レシーバのモードを決定します。
0：通常レシーバモード
1：BIST受信モード（BISTテストデータモード）
このビットがセットされると、RXORDSETは通常どおりに動作します。RXDRはバイトを受信しなくなりますが、CRCチェックは通常のメッセージとして続行します。このモードではヘッダ（MessageIDを含む）が受信されないため、このテスト中にまだ送信する必要があるGoodCRC確認応答に含めるために、受信したMessageIDのカウンタをソフトウェアにより自動でインクリメントする必要があります。

ビット3 **TXHRST**：Tx Hard Resetの送信コマンド

0：影響なし
1：Tx Hard Resetメッセージを開始します
このビットは、メッセージの送信が開始されるか、破棄されるとすぐに、ハードウェアによりクリアされます。

ビット2 **TXSEND**：Txパケットの送信コマンド

0：影響なし
1：Txパケット送信を開始します
このビットは、パケットの送信が開始されるか、破棄されるとすぐに、ハードウェアによりクリアされます。

ビット1:0 **TXMODE[1:0]**：Txパケットのタイプ

このビットフィールドに書き込むと、値に応じて、次のようなアクションがトリガされます。
0x0：他のレジスタであらかじめ定義されたTxパケットの送信
0x1：Cable Resetシーケンス
0x2：BISTテストシーケンス（BISTキャリアモード2）
その他：無効
USB PD仕様のV1.1以降は、BISTキャリアモード2の期間に定義されたカウンタが設けられています。このモードを（「tBISTContMode」遅延後に）正しく終了するには、ペリフェラルを無効化します（UCPDEN = 0）。

74.8.5 UCPD割込みマスクレジスタ（UCPD_IMR）

アドレスオフセット：0x010

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTIE	Res.	Res.	Res.	Res.
											r				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2IE	TYPECEVT1IE	Res.	RXMSGENDIE	RXOVRIE	RXHRSTDETIE	RXORDDETIE	RXNEIE	Res.	TXUNDIE	HRSTSENTIE	HRSTDISCIE	TXMSGABTIE	TXMSGSENTIE	TXMSGDISCIE	TXISIE
rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **FRSEVTIE**：FRSEVT割込みの有効化
0：無効
1：有効

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **TYPECEVT2IE**：TYPECEVT2割込みの有効化
0：無効
1：有効

ビット14 **TYPECEVT1IE**：TYPECEVT1割込みの有効化

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **RXMSGENDIE**：RXMSGEND割込みの有効化
0：無効
1：有効

ビット11 **RXOVRIE**：RXOVR割込みの有効化
0：無効
1：有効

ビット10 **RXHRSTDETIE**：RXHRSTDET割込みの有効化
0：無効
1：有効

ビット9 **RXORDDETIE**：RXORDDET割込みの有効化
0：無効
1：有効

ビット8 **RXNEIE**：RXNE割込みの有効化
0：無効
1：有効

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TXUNDIE**：TXUND割込みの有効化
0：無効
1：有効

ビット5 **HRSTSENTIE**：HRSTSENT割込みの有効化
0：無効
1：有効

ビット4 **HRSTDISCIE**：HRSTDISC割込みの有効化
0：無効
1：有効

ビット3 **TXMSGABTIE**：TXMSGABT割込みの有効化
0：無効
1：有効

ビット2 **TXMSGSENTIE**：TXMSGSENT割込みの有効化
0：無効
1：有効

ビット1 **TXMSGDISCIE**：TXMSGDISC割込みの有効化
0：無効
1：有効

ビット0 **TXISIE**：イベントTXIS割込み有効
0：無効
1：有効

74.8.6 UCPDステータスレジスタ（UCPD_SR）

アドレスオフセット：0x014

リセット値：0x0000 0000

フラグ（単一ビットのステータスビットフィールド）は割込みリクエストと関連付けられます。UCPD_IMRレジスタの対応するビットによって有効化された場合に、割込みが生成されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVT	TYPEC_VSTATE_CC2 [1:0]		TYPEC_VSTATE_CC1 [1:0]	
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2	TYPECEVT1	RXERR	RXMSGEND	RXOVR	RXHRSTDET	RXORDDET	RXNE	Res.	TXUND	HRSTSENT	HRSTDISC	TXMSGABT	TXMSGSENT	TXMSGDISC	TXIS
r	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **FRSEVT**：FRS検出イベント

フラグをクリアするには、FRSEVTCFビットをセットします。

0：新規イベントはありません。

1：新規のFRS受信イベントが発生しました。

ビット19:18 **TYPEC_VSTATE_CC2[1:0]**：CC2ラインの電圧レベル

このステータスビットフィールドは、定常状態のCC2ラインの電圧レベルを示します。

0x0：最低

0x1：低

0x2：高

0x3：最高

USB PDメッセージ中のBMC PHY変調によるCC2ラインの電圧の変化は、ビットフィールド値に影響しません。

ビット17:16 **TYPEC_VSTATE_CC1[1:0]**：

このステータスビットフィールドは、定常状態のCC1ラインの電圧レベルを示します。

0x0：最低

0x1：低

0x2：高

0x3：最高

USB PDメッセージ中のBMC PHY変調によるCC1ラインの電圧の変化は、ビットフィールド値に影響しません。

ビット15 **TYPECEVT2**：CC2ラインでのType-C電圧レベルイベント

このフラグは、新しいType-Cイベントに対応するTYPEC_VSTATE_CC2[1:0] ビットフィールド値の変化を示します。クリアするには、TYPECEVT2CFビットをセットします。

0：新規イベントはありません。

1：新規のType-Cイベントです。

ビット14 **TYPECEVT1**：CC1ラインでのType-C電圧レベルイベント

このフラグは、新しいType-Cイベントに対応するTYPEC_VSTATE_CC1[1:0] ビットフィールド値の変化を示します。クリアするには、TYPECEVT2CFビットをセットします。

0：新規イベントはありません。

1：新規のType-Cイベントです。

ビット13 **RXERR**：受信メッセージエラー

このフラグは、不適切なCRCや切捨てられたメッセージ（EOPが満たされる前に静的になるライン）など、（RXMSGENDを使用して）宣言された最後のRxメッセージのエラーを示します。RXMSGENDフラグがセットされると必ずアサートされます。

0：エラーは検出されていません。

1：エラーが検出されました。

ビット12 **RXMSGEND**：Rxメッセージの受信

このフラグは、CRC値に関係なく、メッセージ（Hard Resetメッセージを除く）が受信されたかどうかを示します。このフラグをクリアするには、RXMSGENDCFビットをセットします。

0：新規Rxメッセージを受信していません。

1：新規Rxメッセージを受信しました。

RXMSGENDフラグがハイになったときにセットされるRXERRフラグは、最後に受信したメッセージのエラーを示します。

ビット11 **RXOVR**：Rxデータオーバーフローの検出

このフラグはRxデータバッファのオーバーフローを示します。クリアするには、RXOVRCFビットをセットします。

0：オーバーフローなし

1：オーバーフロー

バッファのオーバーフローは、受信データの読出し速度が十分でない場合に発生する可能性があります。

ビット10 **RXHRSTDET**：Rx Hard Reset受信の検出

このフラグは、有効なHard Resetメッセージを受信したことを示します。クリアするには、RXHRSTDETCFビットをセットします。

0：Hard Resetは受信していません。

1：Hard Resetを受信しました。

ビット9 **RXORDDET**：Rx順序集合（4つのKコード）の検出

このフラグは、順序集合が検出されたことを示します。関連情報は、UCPD_RX_ORDSETレジスタのRXORDSET[2:0] ビットフィールドに保存されます。クリアするには、RXORDDETCFビットをセットします。

0：順序集合は検出されていません。

1：新規の順序集合を検出しました。

ビット8 **RXNE**：受信データレジスタのノットエンプティの検出

このフラグは、UCPD_RXDRレジスタが空でないことを示します。UCPD_RXDRの読出し時に自動的にクリアされます。

0：Rxデータレジスタは空です。

1：Rxデータレジスタは空ではありません。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TXUND**：Txデータアンダーランの検出

このフラグは、送信メッセージが正常に実行されるように時間内にTxデータレジスタ（UCPD_TXDR）が書き込まれなかったことを示します。クリアするには、TXUNDCFビットをセットします。

0：Txデータ･アンダーランは検出されていません。

1：Txデータ･アンダーランが検出されました。

ビット5 **HRSTSENT**：Hard Resetメッセージの送信

このフラグは、Hard Resetメッセージが送信されることを示します。このフラグをクリアするには、HRSTSENTCFビットをセットします。

0：Hard Resetメッセージが送信されていません。

1：Hard Resetメッセージが送信されました。

ビット4 **HRSTDISC**：Hard Resetの破棄

このフラグは、Hard Resetメッセージが破棄されることを示します。このフラグをクリアするには、HRSTDISCCFビットをセットします。

0：Hard Resetは破棄されません。

1：Hard Resetは破棄されます。

ビット3 **TXMSGABT**：送信メッセージのアボート

このフラグは、後続のHard Resetメッセージ送信リクエストが送信中に優先されるために、Txメッセージがアボートされることを示します。クリアするには、TXMSGABTCFビットをセットします。

0：送信メッセージはアボートされません。

1：送信メッセージはアボートされます。

ビット2 **TXMSGSENT**：メッセージ送信の完了

このフラグは、パケット送信が完了したことを示します。クリアするには、TXMSGSENTCFビットをセットします。

0：Txメッセージは完了していません。

1：Txメッセージは完了しました。

Hard Resetによってメッセージ送信の割込みが発生した場合、このフラグは立てられません。

ビット1 **TXMSGDISC**：メッセージ送信の破棄

このフラグは、メッセージ送信が破棄されたことを示します。このフラグをクリアするには、TXMSGDISCCFビットをセットします。

0：Txメッセージは破棄されていません。

1：Txメッセージは破棄されました。

同時に受信が行われている場合や、ラインに過剰なノイズがある場合は、メッセージの送信が破棄されることがあります。Txメッセージの破棄後、CCラインがアイドル状態になった場合のみ、フラグが立てられます。

ビット0 **TXIS**：送信割込みステータス

このフラグは、UCPD_TXDRレジスタが空で、新しいデータの書込みが必要であることを示します（送信されたデータの量が、TXPAYSZビットフィールドで定義されたペイロードサイズに達していないため）。UCPD_TXDRレジスタにデータが書き込まれると、このフラグはクリアされます。

0：新規のTxデータ書込みは不要です。

1：新規のTxデータ書込みが必要です。

74.8.7 UCPD割込みクリアレジスタ（UCPD_ICR）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTCF	Res.	Res.	Res.	Res.
											w				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2CF	TYPECEVT1CF	Res.	RXMSGENDCF	RXOVRCF	RXHRSTDETCF	RXORDDETCF	Res.	Res.	TXUNDCF	HRSTSENTCF	HRSTDISCCF	TXMSGABTCF	TXMSGSENTCF	TXMSGDISCCF	Res.
w	w		w	w	w	w			w	w	w	w	w	w	

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **FRSEVTCF**：FRSイベントフラグ（FRSEVT）のクリア

このビットをセットするとUCPD_SRレジスタのFRSEVTフラグがクリアされます。

ビット19:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **TYPECEVT2CF**：Type C CC2ラインイベントフラグ（TYPECEVT2）のクリア

このビットをセットするとUCPD_SRレジスタのTYPECEVT2フラグがクリアされます

ビット14 **TYPECEVT1CF**：Type C CC1イベントフラグ（TYPECEVT1）のクリア

このビットをセットするとUCPD_SRレジスタのTYPECEVT1フラグがクリアされます

ビット13 予約済みであり、リセット値に保持する必要があります。

ビット12 **RXMSGENDCF**：Rxメッセージ受信フラグ（RXMSGEND）のクリア

このビットをセットするとUCPD_SRレジスタのRXMSGENDフラグがクリアされます。

ビット11 **RXOVRCF**：Rxオーバーフローフラグ（RXOVR）のクリア

このビットをセットするとUCPD_SRレジスタのRXOVRフラグがクリアされます。

ビット10 **RXHRSTDETCF**：Rx Hard Reset検出フラグ（RXHRSTDET）のクリア
このビットをセットするとUCPD_SRレジスタのRXHRSTDETフラグがクリアされます。

ビット9 **RXORDDETCF**：Rx順序集合検出フラグ（RXORDDET）のクリア
このビットをセットするとUCPD_SRレジスタのRXORDDETフラグがクリアされます。

ビット8:7 予約済みであり、リセット値に保持する必要があります。

ビット6 **TXUNDCF**：Txアンダーフローフラグ（TXUND）のクリア
このビットをセットするとUCPD_SRレジスタのTXUNDフラグがクリアされます。

ビット5 **HRSTSENTCF**：Hard Reset送信フラグ（HRSTSENT）のクリア
このビットをセットするとUCPD_SRレジスタのHRSTSENTフラグがクリアされます。

ビット4 **HRSTDISCCF**：Hard Reset破棄フラグ（HRSTDISC）のクリア
このビットをセットするとUCPD_SRレジスタのHRSTDISCフラグがクリアされます。

ビット3 **TXMSGABTCF**：Txメッセージアボートフラグ（TXMSGABT）のクリア
このビットをセットするとUCPD_SRレジスタのTXMSGABTフラグがクリアされます。

ビット2 **TXMSGSENTCF**：Txメッセージ送信フラグ（TXMSGSENT）のクリア
このビットをセットするとUCPD_SRレジスタのTXMSGSENTフラグがクリアされます。

ビット1 **TXMSGDISCCF**：Txメッセージ破棄フラグ（TXMSGDISC）のクリア
このビットをセットするとUCPD_SRレジスタのTXMSGDISCフラグがクリアされます。

ビット0 予約済みであり、リセット値に保持する必要があります。

74.8.8 UCPD Tx順序集合タイプレジスタ（UCPD_TX_ORDSETR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっていて、パケット送信が実行中でない（TXSENDおよびTXHRSTビットが両方ともロー）場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXORDSET[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXORDSET[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **TXORDSET[19:0]**：送信する順序集合
このビットフィールドで、それぞれ5bitの4つのKコードからなり、送信パケットを定義する、送信対象のフル20bitシーケンスを規定します。ビット0（Kコード1のビット0）を先頭、ビット19（Kコード4のビット4）を末尾とします。

74.8.9 UCPD Txペイロードサイズレジスタ（UCPD_TX_PAYSZR）

アドレスオフセット：0x020

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TXPAYSZ[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **TXPAYSZ[9:0]**：未送信のペイロードサイズ

このビットフィールドは、ソフトウェアとハードウェアによって変更されます。これから送信するペイロード（ヘッダは含まれますが、CRCは含まれません）のバイト数で構成され、データバイトがUCPD_TXDRレジスタに書き込まれるたびに、ビットフィールド値が0に達した場合を除いて、ビットフィールド値がデクリメントされ、TXISビットがセットされます。エニュメレーションした値は、送信開始前の標準ペイロードサイズです。

0x2：2バイト - プロトコル層からの制御メッセージのサイズ
0x6：6バイト - プロトコル層からのデータメッセージの最小許容サイズ
0x1E：30バイト - プロトコル層からのデータメッセージの拡張なしの最大許容サイズ
0x106：262バイト - 拡張メッセージの最大許容サイズ
0x3FF：1024バイト - ペイロードの最大許容サイズ（今後の拡張のために用意）

74.8.10 UCPD Txデータレジスタ（UCPD_TXDR）

アドレスオフセット：0x024

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが有効（UCPDEN = 1）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **TXDATA[7:0]**：送信するデータバイト

74.8.11 UCPD Rx順序集合レジスタ（UCPD_RX_ORDSETR）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPKINVALID[2:0]			RXSOP3OF4	RXORDSET[2:0]		
									r	r	r	r	r	r	r

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6:4 **RXSOPKINVALID[2:0]**：

このビットフィールドは、デバッグ目的でのみ使用されます。
0x0：Kコードは壊れていません。
0x1：最初のKコードが壊れています。
0x2：2番目のKコードが壊れています。
0x3：3番目のKコードが壊れています。
0x4：4番目のKコードが壊れています。
その他：無効

ビット3 **RXSOP3OF4**：

このビットは、正常なKコードの数を示します。デバッグ専用です。
0：4つのKコードのうち、4つが正常です。
1：4つのKコードのうち、3つが正常です。

ビット2:0 **RXORDSET[2:0]**：Rx順序集合コードの検出

0x0：レシーバでSOPコードを検出しました。
0x1：レシーバでSOP' コードを検出しました。
0x2：レシーバでSOP" コードを検出しました。
0x3：レシーバでSOP'_Debugを検出しました。
0x4：レシーバでSOP"_Debugを検出しました。
0x5：レシーバでCable Resetを検出しました。
0x6：レシーバでSOP拡張 #1を検出しました。
0x7：レシーバでSOP拡張 #2を検出しました。

74.8.12 UCPD Rxペイロードサイズレジスタ（UCPD_RX_PAYSZR）

アドレスオフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	RXPAYSZ[9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:0 **RXPAYSZ[9:0]**：受信したRxペイロードサイズ

このビットフィールドには、受信したペイロード（ヘッダは含まれますが、CRCは含まれません）のバイト数が格納されます。UCPD_RXDRレジスタで新しいデータバイトを受信するたびに、ビットフィールド値がインクリメントされ、RXMSGENDフラグがセットされます（そして、有効な場合は割込みが生成されます）。

0x2：2バイト - プロトコル層からの制御メッセージのサイズ

0x6：6バイト - プロトコル層からのデータメッセージの最小許容サイズ

0x1E：30バイト - プロトコル層からのデータメッセージの拡張なしの最大許容サイズ

0x106：262バイト - 拡張メッセージの最大許容サイズ

0x3FF：1024バイト - ペイロードの最大許容サイズ（今後の拡張のために用意）

このビットフィールドでは、バイト受信の実行中の場合（RXMSGENDフラグがロー）、不正な値が返されることがあります。

74.8.13 UCPD受信データレジスタ（UCPD_RXDR）

アドレスオフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **RXDATA[7:0]**：受信したデータバイト

74.8.14 UCPD Rxオーダード･セット拡張レジスタ1（UCPD_RX_ORDEXTR1）

アドレスオフセット：0x034

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが無効（UCPDEN = 0）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXSOPX1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **RXSOPX1[19:0]**：受信した順序集合1

このビットフィールドには、受信した20bitの全シーケンスが格納されます。それぞれ5bitのKコード4つで構成されます。ビット0（Kコード1のビット0）を先頭、ビット19（Kコード4のビット4）を末尾として順番に受信します。

74.8.15 UCPD Rx順序集合拡張レジスタ2（UCPD_RX_ORDEXTR2）

アドレスオフセット：0x038

リセット値：0x0000 0000

このレジスタへの書込みが可能なのは、ペリフェラルが無効（UCPDEN = 0）になっている場合のみです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXSOPX2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:0 **RXSOPX2[19:0]**：受信した順序集合2

このビットフィールドには、受信した20bitの全シーケンスが格納されます。それぞれ5bitのKコード4つで構成されます。ビット0（Kコード1のビット0）を先頭、ビット19（Kコード4のビット4）を末尾として順番に受信します。

74.8.16 UCPDレジスタ・マップ

表 777. UCPDレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	UCPD_CFGR1	UCPDEN	RXDMAEN	TXDMAEN	RXORDSETEN[8:0]									PSC_USBPDCLK[2:0]			Res.	TRANSWIN[4:0]					IFRGAP[4:0]					HBITCLKDIV[5:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	UCPD_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXAFILTEN	Res.	Res.	Res.	Res.	WUPEN	FORCECLK	RXFILT2N3	RXFILTDIS
	リセット値																								0					0	0	0	0
0x008	UCPD_CFGR3	Res.	Res.	Res.	TRIM_CC2_RP [3:0]				Res.	Res.	Res.	Res.	Res.	TRIM_CC2_RD [3:0]				Res.	Res.	Res.	TRIM_CC1_RP [3:0]			Res.	Res.	Res.	Res.	Res.	TRIM_CC1_RD [3:0]			
	リセット値				0	0	0	0						0	0	0	0				0	0	0	0						0	0	0	0
0x00C	UCPD_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2TCDIS	CC1TCDIS	Res.	RDCH	FRSTX	FRSRXEN	Res.	Res.	Res.	Res.	CCENABLE[1:0]		ANAMODE	ANASUBMODE[1:0]		PHYCCSEL	PHYRXEN	RXMODE	TXHRST	TXSEND	TXMODE[1:0]	
	リセット値											0	0		0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x010	UCPD_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTIE	Res.	Res.	Res.	Res.	TYPECEVT2IE	TYPECEVT1IE	Res.	RXMSGENDIE	RXOVRIE	RXHRSTDETIE	RXORDDETIE	RXNEIE	Res.	TXUNDIE	HRSTSENTIE	HRSTDISCIE	TXMSGABTIE	TXMSGSENTIE	TXMSGDISCIE	TXISIE
	リセット値												0					0	0		0	0	0	0	0		0	0	0	0	0	0	0
0x014	UCPD_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVT	TYPEC_VSTATE_CC2 [1:0]		TYPEC_VSTATE_CC1 [1:0]		TYPECEVT2	TYPECEVT1	RXERR	RXMSGEND	RXOVR	RXHRSTDET	RXORDDET	RXNE	Res.	TXUND	HRSTSENT	HRSTDISC	TXMSGABT	TXMSGSENT	TXMSGDISC	TXIS
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x018	UCPD_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTCF	Res.	Res.	Res.	Res.	TYPECEVT2CF	TYPECEVT1CF	Res.	RXMSGENDCF	RXOVRCF	RXHRSTDETCF	RXORDDETCF	Res.	Res.	TXUNDCF	HRSTSENTCF	HRSTDISCCF	TXMSGABTCF	TXMSGSENTCF	TXMSGDISCCF	Res.
	リセット値												0					0	0		0	0	0	0			0	0	0	0	0	0	
0x01C	UCPD_TX_ORDSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXORDSET[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 777. UCPDレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x020	UCPD_TX_PAYSZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXPAYSZ[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x024	UCPD_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x028	UCPD_RX_ORDSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPKINVALID[2:0]			RXSOP3OF4	RXORDSET[2:0]		
	リセット値																										0	0	0	0	0	0	0
0x02C	UCPD_RX_PAYSZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXPAYSZ[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x030	UCPD_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x034	UCPD_RX_ORDEXTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	UCPD_RX_ORDEXTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C - 0x3FF	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、136ページのセクション 2.3を参照してください。

75 デバッグサポート（DBG）

75.1 DBGの概要

以下に示すデバッグの機能を統合させたもので、ソフトウェア開発およびシステムインテグレーションをサポートするために提供されます。

- CPUコアのブレークポイントのデバッグ
- コード実行トレース
- ソフトウェア計測
- クロストリガ

デバッグ機能はJTAG／シリアルワイヤデバッグ用のアクセスポートを介し、業界標準のデバッグツールを使用して制御することができます。トレース用のポートを使用すると、ログおよび分析のためにデータをキャプチャすることができます。

デバッグ機能はArm CoreSightコンポーネントに基づいています。

- SWJ-DP：JTAG／シリアルワイヤデバッグポート
- AHB-AP：AHBアクセスポート
- ROMテーブル
- システム制御空間（SCS）
- ブレークポイントユニット（BPU）
- データウォッチポイントおよびトレースユニット（DWT）
- 計装トレースマクロセル（ITM）
- Embedded Trace Macrocell™（ETM）
- クロストリガインタフェース（CTI）
- トレースポートインタフェースユニット（TPU）

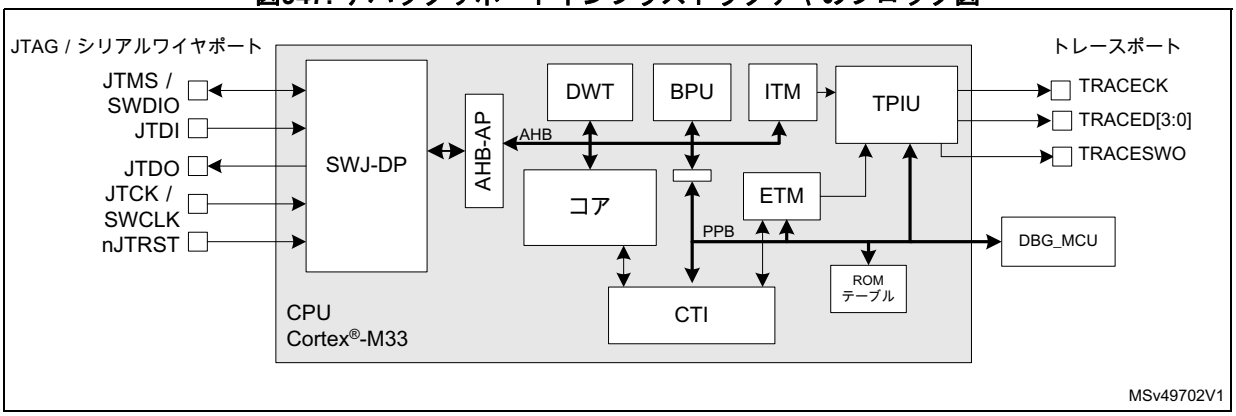
デバッグ機能には、デバッガによってAHB-AP経由でアクセスできます。

追加情報はセクション75.13に掲載されているArm の資料をご覧ください。

75.2 DBGの機能説明

75.2.1 DBGブロック図

図947. デバッグサポートインフラストラクチャのブロック図



75.2.2 DBGピンおよび内部信号

表778.JTAG／シリアルワイヤデバッグポートのピン

ピン名	JTAGデバッグポート		SWデバッグポート		ピン割り当て
	タイプ	説明	タイプ	説明	
JTMS／SWDIO	I	JTAGテストモード選択	IO	シリアルワイヤデータ入出力	PA13
JTCK／SWCLK	I	JTAGテストクロック	I	シリアルワイヤクロック	PA14
JTDI(1)	I	JTAGテストデータ入力	-	-	PA15
JTDO	O	JTAGテストデータ出力	-	-	PB3
nJTRST	I	JTAGテストリセット	-	-	PB4

1. TDIは、同じIO上にUSBPD-CCラインとしてホストされています。プルアップ/プルダウンの競合を避けるため、パッドをTDIとして使用するかCCとして使用するかをユーザオプションで決定できます。

表779.トレースポートピン

ピン名	タイプ	説明	ピン割り当て
TRACED0	O	トレース同期データ出力0	データシートを参照してください
TRACED1		トレース同期データ出力1	
TRACED2		トレース同期データ出力2	
TRACED3		トレース同期データ出力3	
TRACECK		トレースクロック	

表780.シングルワイヤトレースポートのピン

ピン名	タイプ	説明	ピン割り当て
TRACESWO	O	シングルワイヤトレース非同期データ出力	PB3(1)

1. TRACESWOがJTDOと多重化されます。つまり、シングルワイヤトレースはJTAGではなく、シリアルワイヤデバッグ用のインタフェースを使用する場合にしか使用できません。

75.2.3 DBGのリセットおよびクロック

デバッグポート（SWJ-DP）は、STANDBYモードからウェイクアップするときに、パワーオンリセットによってリセットされます。

デバッガは、デバッグインタフェースのピンJTCK/SWCLKを介して、デバッガポート用のクロックを供給します。このクロックを使用して、シリアルワイヤおよびJTAGモード でシリアル入力データを登録し、同時にデバッグポートのステートマシンや内部ロジックを運用します。したがって、デバッグポートを確実にアイドルステートに戻すためには、このクロックはアクセス終了後に数回のサイクルの間トグルし続ける必要があります。

SWJ-DPはDCLKドメインとの非同期インタフェースを内蔵しており、このインタフェースは残りのSWJ-DPおよびアクセスポートを範囲に含みます。

DCLKは、システムクロックのゲート付きバージョンです。

DCLKドメインは、DP_CTRL/STATRのCDBGPWRUPREQを使用して、デバッガによって有効にできます。デバッガがデバイスのデバッグ機能のいずれかにアクセスするには、先にクロックを有効にする必要があります。クロックの有効性は、DP_CTRL/STATRのCDBGPWRUPACKによって反映されます。エネルギー節約のため、DCLKはパワーアップ時に無効化され、またデバッガが切断されたときに無効化する必要があります。

プロセッサに内蔵されているデバッグおよびトレースコンポーネントは、プロセッサクロックでクロック供給されます。

75.2.4 DBG電力ドメイン

デバッグコンポーネントは、コア電力ドメインに配置されています。これは、SHUTDOWNあるいはSTANDBY低電力モードではデバッガ接続を使用できないことを意味しています。デバイスがSTANDBYモードになったときに接続が切断されないよう、DBGMCU_CRにビットをセットしてコアへの電源供給を維持することができます。これにより、プロセッサクロックをアクティブな状態で維持しながらリセットを保留して、デバッグセッションを維持することができます。

75.2.5 デバッグおよび低電力モード

デバイスには省電力機能が内蔵されており、必要ないときにはコア電力ドメインを電源オフまたは停止することができます。電源オフ、またはコアがクロック供給されていない場合、デバッグコンポーネントはすべてデバッガにアクセスできません。このような事態を防ぐために、省電力モードのエミュレーションが実装されています。あるドメインでエミュレーションが有効な場合、そのドメインは省電力モードに移行しますが、そのクロックと電力は維持されます。言い換えれば、ドメインは省電力モードと同様に動作しながら、デバッガとの接続は維持します。

エミュレーションモードは、マイクロコントローラデバッグ（DBGMCU）ユニットでプログラムされます。詳細については、セクション75.12を参照してください。

75.2.6 セキュリティ

トレースおよびデバッグのコンポーネントにより、製品開発中のプロセッサやシステムへの高度なアクセスが可能です。ユーザコードを保護するため、またデバッグ機能を使用して完成した製品の通常動作が変更されたり危険にさらされることがないよう、これらの機能は無効にするか、範囲を制限することができます。たとえば、セキュアソフトウェアのデバッグおよびトレースの無効化は、非セキュアコードのデバッグを妨げることなく実現できます。

システムは次の認証信号を使用して、有効化/無効化するデバッグ機能を判断します。

- **dbgen**：すべてのデバッグ機能のグローバル有効

  0：すべてのデバッグ機能が無効になります。

1：非セキュアステートのデバッグ機能が有効になります。セキュアステートのデバッグ機能は **spiden** 信号の状態に依存します。

- **spiden**：**dbgen** = 1の場合にセキュアステートでのデバッグを有効にします。
  0：デバッグ機能がセキュアステートで無効になります。
  1：デバッグ機能がセキュアステートで有効になります。
- **niden**：トレースと性能の監視を有効にします（非侵入型デバッグ）。
  0：トレース生成は無効です。
  1：非セキュアステートのトレース生成は有効です。セキュアステートのトレース生成は **spniden** 信号の状態に依存します。
- **spniden**：**niden** = 1の場合にセキュアステートでのトレースと性能の監視を有効にします。
  0：トレース生成がセキュアステートで無効になります。
  1：トレース生成がセキュアステートで有効になります。

認証信号の状態に応じた各コンポーネントの動作の詳細については、関連するコンポーネントの章または関連するArmのテクニカルドキュメントを参照してください。

信号の状態は、次の図に示すように、読出しデータ保護（RDP）レベル（セクション 7.6.2：読出し保護（RDP）を参照）に応じてセットされます。

表781.認証信号の状態

RDPレベル	認証信号の状態	説明
0	dbgen = 1、spiden = 1 niden = 1、spniden = 1	デバッグおよびトレースは、プロセッサの状態にかかわらず有効になります。セキュアメモリへのデバッガアクセスが許可されます。
0.5	dbgen = 1、spiden = 0 niden = 1、spniden = 0	デバッガおよびトレースは、プロセッサが非セキュアステートである場合に有効になります。セキュアメモリへのデバッガアクセスが無効になります。
1	dbgen = 1、spiden = 0 niden = 1、spniden = 0	デバッガおよびトレースは、プロセッサが非セキュアステートである場合に有効になります。セキュアメモリおよび次の領域へのデバッガアクセスが無効になります（Flashメモリ、SRAM2、バックアップレジスタ、ICACHE、動作中に復号化される領域（OCTOSPI））。
2	dbgen = 0、spiden = 0 niden = 0、spniden = 0	デバッグおよびトレースは無効になります。

注： **セキュリティ機能は、オプションビットTZEN = 1の場合にのみ適用されます。セキュリティ機能が無効である場合、認証信号はそのままRDPレベルに応じてセットされますが、プロセッサとすべてのメモリが非セキュアであるため、SPNIDENとSPIDENは冗長です。**

認証信号の状態は、Cortex-M33のシステム制御空間（SCS）のDAUTHSTATUSレジスタから読み出すことができます。

デバッガのセキュアメモリへのアクセスは（許可されている場合）、デバッグAHBのセキュアなトランザクションを使用して（つまりAP_CSWRでPROT[6] をセットして）実行する必要があります。

デバッガアクセスは、セキュリティ機能が有効（TZEN = 1）である場合、RDPレベルを問わず、プロセッサがシステムFlashメモリ（RSS）からブートする間は無効になります。

75.3 シリアルワイヤ／JTAGデバッグポート（SWJ-DP）

SWJ-DPはCoreSightコンポーネントの1つで、デバッグ装置との接続用の外部アクセスポートを実装しています。

2種類のインタフェースを設定することができます。

- 5ピンの標準JTAGインタフェース（JTAG-DP）
- 2ピン（クロック + データ）のシリアルワイヤデバッグポート（SW-DP）

これらの2つのモードは同じIOピンを共有しているため、相互に排他的です。

デフォルトでは、JTAG-DPはシステムリセットまたはパワーオンリセット後に選択されます。5つのIO ピンはデバッグ代替機能モードでハードウェアによって設定されます。SWJ-DP は、JTDI、JTMS/SWDIO、およびnJTRSTにプルアップレジスタ、JTCK/SWCLKにプルダウンレジスタを搭載しています。

デバッガはJTMS/SWDIOに次のようなシリアルデータシーケンスを送信することによって、SW-DPを選択することができます。

...,（1が50個以上）,...,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,...,（1が50個以上）,...

JTCK/SWCLKはデータビットごとに循環させる必要があります。

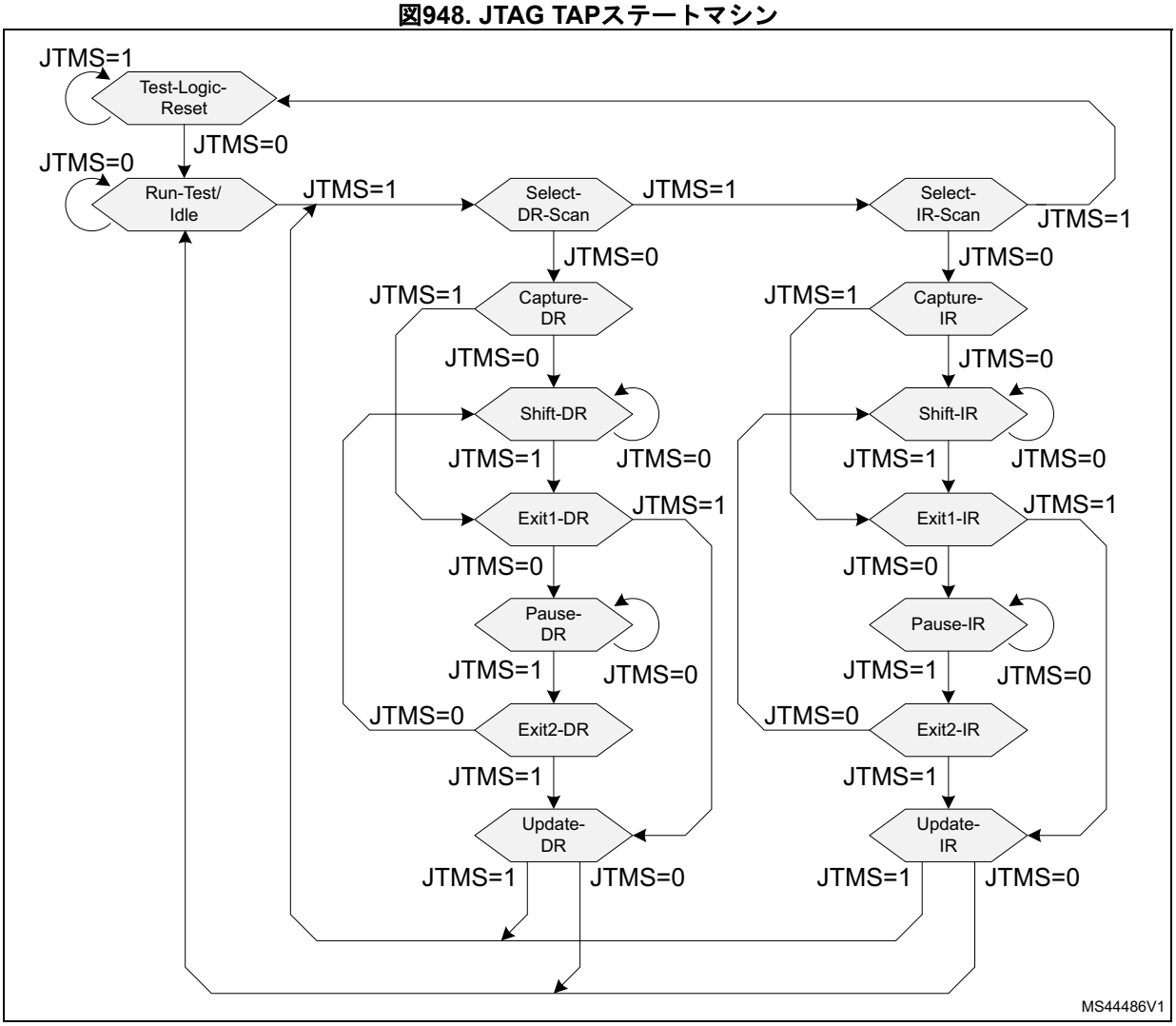
SW-DPモードでは、使用していないJTAGピン、JTDI、JTDO、およびnJTRSTを他の機能に使用することができます。

注： **SWJポートのすべてのIOはソフトウェアで他の機能に合わせて再設定できますが、デバッグはできなくなります。**

75.3.1 JTAGデバッグポート

JTAG-DPはIEEE 1149.1-1990に基づく、下図に示すTAPステートマシン（TAPSM）を実装しています。ステートマシンは2つのスキャンチェインを制御します。一方は1つの命令レジスタ（IR）に、もう一方は複数のデータレジスタ（DR）に関連付けられています。

図948. JTAG TAPステートマシン



JTAG-DPの動作は次のとおりです。

1. TAPSMがCapture-IRステートに入ると、0b0001が命令レジスタ（IR）のスキャンチェインに転送されます。IRスキャンチェインはJTDIとJTDOの間に接続されます。
2. TAPSMがShift-IRステートにあるとき、IRスキャンチェインはJTCKの各立ち上がりエッジで1bitだけシフトします。つまり、最初の単位時間で以下のアクションが起こります。
   - IRスキャンチェインのLSBがJTDOに出力されます。
   - IRスキャンチェインのビット [n] がビット [n-1] に転送されます。
   - JTDIの値がIRスキャンチェインのMSBに転送されます。

3. TAPSMがUpdate-IRステートに入ると、IRスキャンチェインにスキャンされた値が命令レジスタに転送されます。
4. TAPSMがCapture-DRステートに入ると、ある値がデータレジスタの1つから、JTDIとJTDOの間に接続されたDRスキャンチェインの1つに転送されます。
5. 命令レジスタに保持された値によって、どのデータレジスタ、および該当するDRスキャンチェインを選択するかが決まります。
6. 次に、このデータは、TAPSMがShift-DRステートにある間に、Shift-IRステートでIRがシフトしたのと同じようにシフトされます。
7. TAPSMがUpdate-DRステートに入ると、DRスキャンチェインにスキャンされた値が選択されたデータレジスタに転送されます。
8. TAPSMがRun-Test/Idleステートにあるときは、特別なアクションは起こりません。IDCODE命令はIRにロードされます。

アクティブな場合、nJTRST信号はTest-Logic-Resetステートとは非同期にステートマシンをリセットします。

以下の表に、4bitのIR命令に対応するデータレジスタを記載します。

表782.JTAG-DPデータレジスタ

IR命令	DRレジスタ	スキャンチェイン長	説明
0000～0111	（BYPASS）	1	実装なし：BYPASSが選択されます。
1000	ABORT	35	アボートレジスタ – ビット31:1 = 予約済み – ビット0 = APABORT：APアボートを生成するには1を書き込みます。
1001	（BYPASS）	1	予約済み：BYPASSが選択されます。
1010	DPACC	35	デバッグポートアクセスレジスタ デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット34:3 = DATA[31:0] = 書込みリクエスト用に転送する32bitデータ ビット2:1 = A[3:2] = デバッグポートレジスタの2bitアドレス ビット0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時： ビット34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される32bitデータ ビット2:0 = ACK[2:0] = 3bit確認応答： – 010 = OK/FAULT – 001 = WAIT – その他：予約済み

表782.JTAG-DPデータレジスタ （続き）

IR命令	DRレジスタ	スキャンチェイン長	説明
1011	APACC	35	アクセスポートアクセスレジスタ アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット34:3 = DATA[31:0] = 書込みリクエスト用にシフトインする32bitデータ ビット2:1 = A[3:2] = アクセスポートレジスタの2bitサブアドレス ビット0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時： ビット34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される32bitデータ ビット2:0 = ACK[2:0] = 3bit確認応答： – 010 = OK/FAULT – 001 = WAIT – その他：予約済み
1100	（BYPASS）	1	予約済み：BYPASSが選択されます。
1101	（BYPASS）	1	予約済み：BYPASSが選択されます。
1110	IDCODE	32	識別コード 0x0BA0 4477：Cortex-M33 JTAGデバッグポートIDコード
1111	BYPASS	1	バイパス：1 JTCKサイクル遅延がJTDIとJTDOの間に挿入されます。

DRレジスタの詳細は、Arm Debug Interface Architecture Specificationに記載されています（セクション75.13を参照）。

75.3.2 シリアルワイヤデバッグポート

シリアルワイヤデバッグプロトコルでは、次のピンを使用します。

- SWCLK：ホストからターゲットへのクロック
- SWDIO：双方向シリアルデータ

シリアルデータはクロックと同期してLSBが先に転送されます。

転送は以下の3フェーズで構成されます。

1. ホストによって送信されるパケットリクエスト（8bit）（表783を参照）
2. ターゲットによって送信される確認応答（3bit）（表784を参照）
3. ホスト（書込みの場合）またはターゲット（読出しの場合）によって送信されるデータ転送（33bit）（表785を参照）

確認応答がOKである場合にのみ、データ転送が実行されます。

データ方向が逆転した場合、各フェーズの間に1クロックサイクルのターンアラウンド時間が挿入されます。

表783.パケットリクエスト

ビットフィールド	名前	説明
0	開始	1である必要があります。
1	APnDP	– 0：DPレジスタアクセス - セクション 75.3.3：デバッグポートレジスタを参照のこと。 – 1：APレジスタアクセス - セクション 75.4：アクセスポートを参照のこと。
2	RnW	– 0：書込みリクエスト – 1：読出しリクエスト
4:3	A[3:2]	DP/APレジスタのアドレスフィールド（表786 または 表787 を参照）
5	Parity	先行するビットの1bitパリティ
6	STOP	0
7	Park	ホストによって駆動されません。ターゲットによって1と読み出される必要があります。

表784.ACK応答

ビットフィールド	名前	説明
2:0	ACK	– 000：FAULT – 010：WAIT – 100：OK

表785.データ転送

ビットフィールド	名前	説明
31:0	WDATAまたはRDATA	書込み／読出しデータ
32	Parity	32データビットのうちの1bitパリティ

ターゲットからフォールトまたはウェイトのACK応答があった場合、オーバーラン検出が有効でなければ、データ転送フェーズは取り消されます。この場合、データはターゲットから無視されるか（書込みの場合）、駆動されません（読出しの場合）。

初回接続時またはプロトコルエラー発生時には、ホストはラインリセットを生成する必要があります。ラインリセットは、SWDIOがハイの状態の50回以上のSWCLKサイクルと、これに続くSWDIOがローの状態の2回のSWCLKサイクルで構成されています。

シリアルワイヤデバッグプロトコルの詳細については、Arm Debug Interface Architecture Specification[1]を参照してください。

注： **SWJ-DPはSWDプロトコルのバージョン2を実装しています。**

75.3.3 デバッグポートレジスタ

SW-DPとJTAG-DPは両方とも表786に示されたデバッグポート（DP）レジスタにアクセスします。

デバッガは次のようにDPレジスタへアクセスすることができます。

1. JTAGを使用する場合は、バンク内にあるレジスタアドレスをDPACCレジスタのA(3:2) フィールドに書き込みます。読出しか書込みかを選択するために、RnWビットをプログラムします。書込みの場合は、データフィールドに書込みデータをプログラムします。SWDを使用する場合、A(3:2) およびRnWフィールドは、APnDPビットをリセットした状態でSW-DPに送信されるパケットリクエストワードの一部となります（表783を参照）。書込みデータはデータフェーズで送信されます。
2. アドレス0x4のDPバンクレジスタの1つにアクセスするには、まずアドレス0x8のDP_SELECTRレジスタにレジスタ番号を書き込む必要があります。アドレス0x4への後続の読出しまたは書込みはすべて、DP_SELECTRの内容に対応するレジスタにアクセスします。

DPデバッグポートの識別レジスタ（DP_DPIDR）

アドレスオフセット：0x0

リセット値：0x0BE0 2477 (SW-DP)、0x0BE0 1477 (JTAG-DP)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				PARTNO[7:0]								Res.	Res.	Res.	MIN
r	r	r	r	r	r	r	r	r	r	r	r				r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VERSION[3:0]				DESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット31:28 **REVISION[3:0]**：リビジョンコード
0x0（JTAG-DP）：r0p0
0x0（SW-DP）：r0p0

ビット27:20 **PARTNO[7:0]**：デバッグポートの部品番号
0xBE

ビット19:17 予約済みであり、リセット値に保持する必要があります。

ビット16 **MIN**：最小デバッグポート（MINDP）の実装
0x1：MINDPが実装されます（トランザクションカウンタとプッシュ動作はサポートされません）。

ビット15:12 **VERSION[3:0]**：デバッグポートアーキテクチャバージョン
0x1（JTAG-DP）：DPv1
0x2（SW-DP）：DPv2

ビット11:1 **DESIGNER[10:0]**：JEDEC設計者IDコード
0x23B：Arm JEDECコード

ビット0 予約済みであり、リセット値に保持する必要があります。

DPアボートレジスタ（DP_ABORTR）

アドレスオフセット：0x0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ORUNERRCLR	WDERRCLR	STKERRCLR	Res.	DAPABORT
											w	w	w		w

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4 **ORUNERRCLR**：オーバーランエラーのクリア

0：影響なし。

1：DP_CTRL/STATRレジスタのSTICKYORUNビットがクリアされます。

ビット3 **WDERRCLR**：書込みデータエラーのクリア

0：影響なし。

1：DP_CTRL/STATRレジスタのWDATAERRビットがクリアされます。

ビット2 **STKERRCLR**：スティッキーエラーのクリア

0：影響なし。

1：DP_CTRL/STATRレジスタのSTICKYERRビットがクリアされます。

ビット1 予約済みであり、リセット値に保持する必要があります。

ビット0 **DAPABORT**：過剰な数のウェイト応答が返された場合の現在のAPトランザクションのアボート

このビットはトランザクションがストールしていることを示します。

0：影響なし。

1：トランザクションがアボートされます。

DP制御およびステータスレジスタ（DP_CTRL/STATR）

アドレスオフセット：0x4

リセット値：0x0000 0000

このレジスタはDP_SELECTR.DPBANKSEL[3:0] = 0x0の場合にアクセス可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	CDBGPWRUPACK	CDBGPWRUPREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		r	r												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATAERR	READOK	STICKYERR	Res.	Res.	Res.	STICKYORUN	ORUNDETECT
								r	r	r				r	r

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **CDBGPWRUPACK**：デバッグパワーアップの確認応答（セクション75.2.5の説明を参照）

0：DCLKがゲートされます。

1：DCLKは有効です。

ビット28 **CDBGPWRUPREQ**：デバッグパワーアップリクエスト

このビットはDCLKイネーブルリクエスト信号を制御します。

0：DCLKのゲート操作をリクエストします。

1：DCLKの有効化をリクエストします。

ビット27:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **WDATAERR**：SW-DPの書込みデータエラー（読出し専用）

このビットは、書込みのデータフェーズでパリティエラーまたはフレーミングエラーが発生しているか、DPによって受け付けられた書込みが、その後APに提供されることなく破棄されたことを示します。このビットは、ABORT.WDERRCLRビットに1を書き込むことによってリセットされます。

0：エラーはありません。

1：エラーが発生しました。

注： このビットはJTAG-DPに予約済みです。

ビット6 **READOK**：SW-DPのAP読出し応答（読出し専用）

このビットは、最後のAP読出しアクセスへの応答を示します。

0：読出しはNGです。

1：読出しはOKです。

注： このビットはJTAG-DPに予約済みです。

ビット5 **STICKYERR**：トランザクションエラー（SW-DPでは読出し専用、JTAG-DPでは読み書き）

このビットは、AP トランザクション中にエラーが発生したことを示します。このビットは、DP_ABORTR.STKERRCLRビット（SW-DPおよびJTAG-DP）に1を書き込むことによってリセットされます。

0：エラーはありません。

1：エラーが発生しました。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **STICKYORUN**：オーバーラン（SW-DPでは読出し専用、JTAG-DPでは読み書き）

このビットは、オーバーラン（前のトランザクションの完了前に新たなトランザクションを受信）が発生したことを示します。このビットは、ORUNDETECTビットがセットされた場合にのみセットされます。このビットは、ABORTR.ORUNERRCLRビット（SW-DPおよびJTAG-DP）に1を書き込むことによってリセットされます。

0：オーバーランはありません。

1：オーバーランが発生しました。

ビット0 **ORUNDETECT**：オーバーラン検出モード有効

0：無効

1：有効オーバーランが発生した場合、STICKYORUNビットがセットされ、このビットがクリアされるまで次のトランザクションはブロックされます。

DPのデータリンク制御レジスタ（DP_DLCR）

アドレスオフセット：0x4

リセット値：0x0000 0000

このレジスタはDP_SELECTR.DPBANKSEL[3:0] = 0x1の場合にアクセス可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TURNROUND[1:0]		WIREMODE[1:0]		Res.	Res.	Res.	Res.	Res.	Res.
						r	r	r	r						

ビット31:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **TURNROUND[1:0]**：SWDIOのトライステート周期

0x0：1データビット周期

ビット7:6 **WIREMODE[1:0]**：SW-DPモード

0x0：同期モード

ビット5:0 予約済みであり、リセット値に保持する必要があります。

DPターゲット識別レジスタ（DP_TARGETIDR）

アドレスオフセット：0x4

リセット値：0xXXXX 0041

このレジスタはDP_SELECTR.DPBANKSEL[3:0] = 0x2の場合にアクセス可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TREVISION[3:0]				TPARTNO[15:4]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPARTNO[3:0]				TDESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット31:28 **TREVISION[3:0]**：ターゲットのリビジョン

STM32U5Fx/5Gxの場合
0x1：リビジョンA、Z

STM32U59x/5Axの場合
0x3：リビジョンX

STM32U575/585の場合
0x2：リビジョンX、W

STM32U535/545の場合
0x1：リビジョンZ

ビット27:12 **TPARTNO[15:0]**：ターゲットの部品番号
0x4550：STM32U535/545
0x4760：STM32U5Fx/5Gx
0x4810：STM32U59x/5Ax
0x4820：STM32U575/585

ビット11:1 **TDESIGNER[10:0]**：ターゲットの設計者JEDECコード
0x020：STMicroelectronics

ビット0 予約済みであり、リセット値に保持する必要があります。

DPデータリンクプロトコル識別レジスタ（DP_DLPIDR）

アドレスオフセット：0x4

リセット値：0x0000 0001

このレジスタはDP_SELECTR.DPBANKSEL[3:0] = 0x3の場合にアクセス可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TINSTANCE[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PROTSVN[3:0]			
												r	r	r	r

ビット31:28 **TINSTANCE[3:0]**：ターゲットのインスタンス番号
このフィールドは、マルチドロップシステムにおけるデバイスのインスタンス番号を定義します。
0x0：インスタンス番号0

ビット27:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PROTSVN[3:0]**：シリアルワイヤデバッグプロトコルのバージョン
0x1：バージョン2

DPイベントステータス（DP_EVENTSTATR）

アドレスオフセット：0x4

リセット値：0x0000 0001

このレジスタはDP_SELECTR.DPBANKSEL[3:0] = 0x4の場合にアクセス可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EA
															r

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EA**：イベントステータスフラグ
0：Cortex-M33プロセッサは停止しています。
1：Cortex-M33プロセッサは停止していません。

DPイベントステータスレジスタ（DP_RESENDR）

アドレスオフセット：0x8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESEND[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESEND[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RESEND[31:0]**：最後のAP読出しまたはDP_RDBUFF読出しによって返された値
このレジスタは、読出し転送の破損時に使用されます。

DPアクセスポート選択レジスタ（DP_SELECTR）

アドレスオフセット：0x8

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
APSEL[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w	w	w	w	w	w	w	w								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBANKSEL[3:0]				DPBANKSEL[3:0]			
								w	w	w	w	w	w	w	w

ビット31:24 **APSEL[7:0]**：アクセスポートの選択
このフィールドによって次のトランザクションのアクセスポートを選択します。
0x0：AP0 - Cortex-M33デバッグアクセスポート（AHB-AP）
その他：予約済み

ビット23:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **APBANKSEL[3:0]**：APレジスタバンクの選択
このフィールドによって、次のトランザクションのアクティブなAPで4ワードレジスタバンクを選択します。

ビット3:0 **DPBANKSEL[3:0]**：DPレジスタバンクの選択
このフィールドによって、デバッグポートのアドレス0x4のレジスタを選択します。
0x0：DP_CTRL/STATレジスタ
0x1：DP_DLCRレジスタ
0x2：DP_TARGETIDレジスタ
0x3：DP_DLPIDRレジスタ
0x4：DP_EVENTSTATレジスタ
その他：予約済み

DP読出しバッファレジスタ（DP_RDBUFFR）

アドレスオフセット：0xC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDBUFF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDBUFF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **RDBUFF[31:0]**：最後のAP読出しアクセスによって返された値

AP読出しアクセスによって返された値は、同じアドレスに対する2回目の読出しアクセスで取得でき、これによって対応するバス上で新しいトランザクションを開始することができます。値はこのレジスタから読み出すこともできますが、その場合、新しいAPトランザクションは発生しません。

75.3.4 デバッグポートレジスタマップ

これらのレジスタはCPUメモリバス上にはありません。SW-DPおよびJTAG-DPのデバッグインタフェースからのみアクセスできます。

デバッグポートアドレスは2bit幅で、JTAG-DPレジスタのDPACCまたはSW-DPパケットリクエストのA[3:2] フィールドで定義されています。

表786.デバッグポートのレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	DP_DPIDR	REVISION [3:0]				PARTNO[7:0]								Res.	Res.	Res.	MIN	VERSION [3:0]				DESIGNER[10:0]											Res.
	リセット値	0	0	0	0	1	0	1	1	1	1	1	0				0	0	0	1	0	0	1	0	0	0	1	1	1	0	1	1	1
0x0	DP_ABORTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ORUNERRCLR	WDERRCLR	STKERRCLR	Res.	DAPABORT
	リセット値																												0	0	0		0
0x4(1)	DP_CTRL/STATR	Res.	Res.	CDBGPWRUPACK	CDBGPWRUPREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATAERR	READOK	STICKYERR	Res.	Res.	Res.	STICKYORUN	ORUNDETECT
	リセット値			0	0																					0	0	0				0	0
0x4(2)	DP_DLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TURNROUND [1:0]		WIREMODE [1:0]		Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																							0	0	0	0						
0x4(3)	DP_TARGETIDR	TREVISION [3:0]				TPARTNO[15:0]																TDESIGNER[10:0]											Res.
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	

表786.デバッグポートのレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4[4]	DP_DLPIDR	TINSTANCE[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PROTSVN[3:0]			
	リセット値	0	0	0	0																									0	0	0	1
0x4[5]	DP_EVENTSTATR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EA
	リセット値																																1
0x8	DP_RESENDR	RESEND[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	DP_SELECTR	APSEL[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBANKSEL[3:0]				DPBANKSEL[3:0]			
	リセット値	x	x	x	x	x	x	x	x																	x	x	x	x	x	x	x	x
0xC	DP_RDBUFFR	RDBUFF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. DP_SELECTR.DPBANKSEL[3:0] = 0x0
2. DP_SELECTR.DPBANKSEL[3:0] = 0x1
3. DP_SELECTR.DPBANKSEL[3:0] = 0x2
4. DP_SELECTR.DPBANKSEL[3:0] = 0x3
5. DP_SELECTR.DPBANKSEL[3:0] = 0x4

75.4 アクセスポート

DPにはアクセスポート（AP）が1つ接続されています。内部AHBバスを介して、Cortex-M33プロセッサコアに組み込まれているデバッグおよびトレース機能へのアクセスを可能にします。

75.4.1 アクセスポートレジスタ

アクセスポートはMEM-APタイプです。つまり、デバッグおよびトレースコンポーネントのレジスタはAHBのアドレス空間にマップされます。APは、デバッガによって、それぞれ4つのレジスタからなるバンクとして構成された一連の32bitレジスタと見なされます。これらのレジスタの一部を使用してAP自体を設定または監視し、その他のレジスタはバスで転送を行うために使用します。APレジスタを 表787に記載します。

APレジスタのアドレスは、以下のフィールドによって構成されます。

- ビット [7:4] : DP_SELECTR内のAPBANKSEL[3:0] の内容。
- ビット [3:2] : 使用するデバッグインタフェースに応じて、JTAG-DPのAPACCデータレジスタのA(3:2) フィールドの内容（表782を参照）、またはSW-DPパケットリクエストの内容（表783を参照）のいずれか。
- ビット [1:0] = 常に0にセットされます。

DP_SELECTR.APSEL[3:0] の内容は、アクセスされるMEM-APを定義します。

デバッガは次のようにAPレジスタへアクセスすることができます。

1. APを選択するためにDP_SELECTRのAPSEL[3:0] をプログラムし、アクセスするレジスタバンクを選択するためにDP_SELECTRのAPBANKSEL[3:0] をプログラムします。
2. JTAG を使用する場合は、バンク内にあるレジスタアドレスを APACC データレジスタの A(3:2) フィールドに書き込みます。読出しか書込みかを選択するために、RnWビットをプログラムし

ます。書込みの場合は、DATAフィールドに書込みデータをプログラムします。SWDを使用する場合、A(3:2) およびRnWフィールドは、APnDPビットをセットした状態でSW-DPに送信されるパケットリクエストワードの一部となります（表783を参照）。書込みデータはデータフェーズで送信されます。

デバッガは、APレジスタを使用して、以下の手順でメモリマップドデバッグコンポーネントレジスタにアクセス可能です（上記のAPレジスタアクセス手順を使用）。

1. AP_TARのトランザクションターゲットアドレスをプログラムします。
2. 必要であれば、転送パラメータ（AddrIncなど）を使用してAP_CSWRをプログラムします。
3. AP_TARに保持されているアドレスのバストランザクションを開始するには、AP_DRWRに対する書込みまたは読出しを行います。あるいは、AP_BDnRに対する読出し／書込みを行うことで、アドレスTAR[31:4] + nへのアクセスを開始します（AP_TARレジスタのアドレスを変更せずに、最大4つの連続したアドレスへのアクセスを可能にします）。

MEM-APの詳細については、ARM® Debug Interface Architecture Specification[1]を参照してください。

AP制御／ステータスワードレジスタ（AP_CSWR）

アドレスオフセット：0x0

リセット値：0x0100：00X0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	PROT[6]	Res.	Res.	PROT[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw			rw	rw	rw	r								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTATUS	ADDRINC[1:0]		Res.	SIZE[2:0]		
									r	rw	rw		rw	rw	rw

ビット31　予約済みであり、リセット値に保持する必要があります。

ビット30　**PROT[6]**：セキュア転送リクエスト

このフィールドにはバス転送の保護属性HPROT[6] をセットします。

0：セキュア転送

1：非セキュア転送

ビット29:28　予約済みであり、リセット値に保持する必要があります。

ビット27:24　**PROT[3:0]**：バス転送保護

このフィールドにはバス転送の保護属性HPROT[3:0] をセットします。

0bXXX1：データアクセス（ビット24は読出し専用）

0bXX0X：非特権モード

0bXX1X：特権モード

0bX0XX：バッファ格納不可

0bX1XX：バッファ格納可

0b0XXX：共有不可、ルックアップ不可、変更不可

0b1XXX：共有可、ルックアップ可、変更可

ビット23:7　予約済みであり、リセット値に保持する必要があります。

ビット6　**DBGSTATUS**：デバイス有効（DEVICEEN）ステータス

0：AHB転送はブロックされます。

1：AHB転送は有効です。

ビット5:4 **ADDRINC[1:0]**：自動インクリメントモード

TARアドレスがトランザクション後に自動的にインクリメントされるかどうかを定義します。

0x0：自動インクリメントされません。

0x1：アドレスがトランザクションのバイトサイズ（SIZEフィールド）でインクリメントされます。

その他：予約済み

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2:0 **SIZE[2:0]**：次のメモリアクセストランザクションのサイズ

0x0：バイト（8bit）

0x1：ハーフワード（16bit）

0x2：ワード（32bit）

その他：予約済み

AP転送アドレスレジスタ（AP_TAR）

アドレスオフセット：0x04

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **TA[31:0]**：現在の転送のアドレス

APデータ読出し／書込みレジスタ（AP_DRWR）

アドレスオフセット：0x0C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TD[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **TD[31:0]**：現在の転送のデータ

APバンクデータnレジスタ（AP_BDnR）

アドレスオフセット：0x10 + 0x4 * n、（n = 0〜3）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TBD[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TBD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **TBD[31:0]**：アドレスTARへの現在の転送のバンクデータ

TA + AP_BDnRアドレス [3:2] + 0b00。

自動アドレスのインクリメントはAP_BD0-3Rでは行われません。バンク転送は、ワード転送の場合にのみサポートされます。

AP設定レジスタ（AP_CFGR）

アドレスオフセット：0xF4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LD	LA	BE
													r	r	r

ビット31:3 予約済みであり、リセット値に保持する必要があります。

ビット2 **LD**：大きなデータ

0：32bit以下のデータをサポートします。

ビット1 **LA**：長いアドレス

0：32bit以下の物理アドレスをサポートします。

ビット0 **BE**：ビッグエンディアン

0：リトルエンディアンのみをサポートします。

APベースアドレスレジスタ（AP_BASER）

アドレスオフセット：0xF8

リセット値：0xE00F E003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BASEADDR[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASEADDR[15:12]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FORMAT	ENTRYPRESENT
r	r	r	r											r	r

ビット31:12 **BASEADDR[31:12]**：最初のROMテーブルのベースアドレス（ビット31 ~ 12）
ROMテーブルは4KBの境界上に整列させる必要があるため、12 LSBはゼロです。
0xE00FE

ビット11:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **FORMAT**：ベースアドレスレジスタフォーマット
1：ARMデバッグインタフェースv5

ビット0 **ENTRYPRESENT**：デバッグコンポーネントの有無
アクセスポートバス上のデバッグコンポーネントの有無を示します。
1：デバッグコンポーネントあり

AP識別レジスタ（AP_IDR）

アドレスオフセット：0xFC

リセット値：0x1477 0015

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				JEDECBANK[3:0]				JEDECCODE[6:0]							CLASS[3]
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLASS[2:0]			Res.	Res.	Res.	Res.	Res.	VARIANT[3:0]				TYPE[3:0]			
r	r	r						r	r	r	r	r	r	r	r

ビット31:28 **REVISION[3:0]**：リビジョン番号
0x1：r0p1

ビット27:24 **JEDECBANK[3:0]**：JEDECバンク
0x4：Arm

ビット23:17 **JEDECCODE[6:0]**：JEDECコード
0x3B：Arm

ビット16:13 **CLASS[3:0]**：
0x8：MEM-AP

ビット12:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **VARIANT[3:0]**：

0x1：Cortex-M33

ビット3:0 **TYPE[3:0]**：

0x5：AHB5

75.4.2 アクセスポートレジスタマップ

これらのレジスタはCPUメモリバス上にはありません。SW-DPおよびJTAG-DPのデバッグインタフェースからのみアクセスできます。

アクセスポートアドレスは8bit幅で、DP_SELECTR.APBANKSEL[3:0] フィールドならびにJTAG-DPレジスタのDPACCまたはSW-DPパケットリクエストのA[3:2] フィールドで定義されます。

表787.アクセスポートのレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	AP_CSWR	Res.	PROT[6]	Res.	Res.	PROT[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTATUS	ADDRINC[1:0]		Res.	SIZE[2:0]		
	リセット値		0			0	0	0	1																		X	X	X		0	0	0
0x04	AP_TAR	TA[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x08	予約済み	予約済み																															
0x0C	AP_DRWR	TD[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x10	AP_BD0R	TBD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	AP_BD1R	TBD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	AP_BD2R	TBD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	AP_BD3R	TBD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20から0xF0	予約済み	予約済み																															
0xF4	AP_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LD	LA	BE
	リセット値																														0	0	0
0xF8	AP_BASER	BASEADDR[31:12]																				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FORMAT	ENTRYPRESENT
	リセット値	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0										1	1
0xFC	AP_IDR	REVISION[3:0]				JEDECBANK[3:0]				JEDECCODE[6:0]							CLASS[3:0]				Res.	Res.	Res.	Res.	Res.	VARIANT[3:0]				TYPE[3:0]			
	リセット値	0	0	0	1	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0						0	0	0	1	0	1	0	1

75.5 ROMテーブル

ROMテーブルはCoreSightコンポーネントの一つで、AHB-APを介してアクセス可能なすべてのCoreSightデバッグコンポーネントのベースアドレスを格納します。このようなテーブルを使用することにより、デバッガはCoreSightシステムのトポロジを自動検出することができます。

CPUサブシステムには2つのROMテーブルがあります。MCU ROMテーブルはAHB-APのベースレジスタによって指し示されます。これには、プロセッサROMテーブルとTPIUレジスタ、またMCUデバッグユニットのベースアドレスポインタが格納されます。

MCU ROMテーブル（以下の表を参照）は4KB、すなわちアドレス空間の32bit幅の領域（0xE00F E000 ~ 0xE00F EFFC）を占有します。

表788.MCU ROMテーブル

ROMテーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ（KB）	エントリ
0xE00F E000	プロセッサROMテーブル	0xE00F F000	0x0000 1000	4	0x0000 1003
0xE00F E004	TPIU	0xE004 0000	0xFFF4 2000	4	0xFFF4 2003
0xE00F E008	DBGMCU	0xE004 4000	0xFFF4 6000	4	0xFFF4 6003
0xE00F E00C	予約済みです。	-	-	-	0x1FF0 2002
0xE00F E010	テーブルの先頭	-	-	-	0x0000 0000
0xE00F E014 ～ 0xE00F EFC8	予約済みです。	-	-	-	0x0000 0000
0xE00F EFCC ~ 0xE00F EFFC	ROMテーブルレジスタ	-	-	-	表790を参照

プロセッサROMテーブルには、ベースアドレスポインタが、システム制御空間（SCS）レジスタ用に加え、BPU、DWT、ITM、ETM、およびCTI用に用意されています。これにより、デバッガはCPUコアを識別可能になります。

プロセッサROMテーブル（以下の表を参照）は4KB、すなわちアドレス空間の32bit幅の領域（0xE00F F000 ~ 0xE00F FFFC）を占有します。

表789.プロセッサROMテーブル

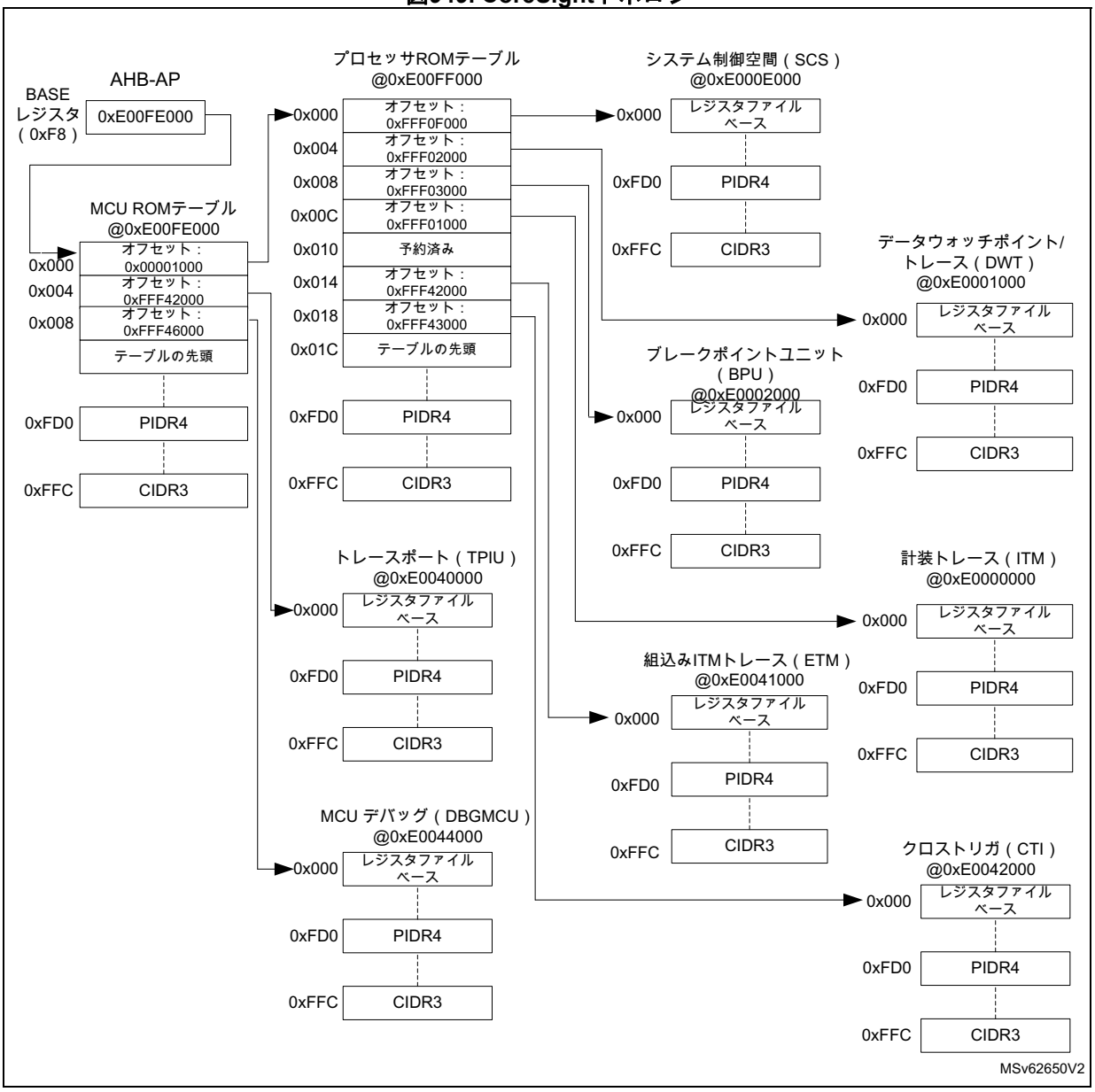
ROMテーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ（KB）	エントリ
0xE00F F000	SCS	0xE000 E000	0xFFF0 F000	4	0xFFF0 F003
0xE00F F004	DWT	0xE000 1000	0xFFF0 2000	4	0xFFF0 2003
0xE00F F008	BPU	0xE000 2000	0xFFF0 3000	4	0xFFF0 3003
0xE00F F00C	ITM	0xE000 0000	0xFFF0 1000	4	0xFFF0 1003
0xE00F F010	予約済みです。	-	-	-	0xFFF4 1002
0xE00F F014	ETM	0xE004 1000	0xFFF4 2000	4	0xFFF4 2003
0xE00F F018	CTI	0xE004 2000	0xFFF4 3000	4	0xFFF4 3003
0xE00F F01C	予約済みです。	-	-	-	0xFFF4 4002
0xE00F F020	テーブルの先頭	-	-	-	0x0000 0000

表789.プロセッサROMテーブル （続き）

ROMテーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ（KB）	エントリ
0xE00F F024 ～ 0xE00F FFC8	予約済みです。	-	-	-	0x0000 0000
0xE00F FFCC ～ 0xE00F FFFC	ROMテーブルレジスタ	-	-	-	表791 を参照

Cortex-M33内のCoreSightコンポーネントのトポロジを次の図に示します。

図949. CoreSightトポロジ



75.5.1 MCU ROMテーブルレジスタ

MCU ROMメモリタイプレジスタ（MCUROM_MEMTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
															r

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SYSMEM**：システムメモリ

0x1：このバス上にシステムメモリがあります。

MCU ROM CoreSightペリフェラルIDレジスタ4（MCUROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x0：STMicroelectronics JEDEC継続コード

MCU ROM CoreSightペリフェラルIDレジスタ0（MCUROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8　予約済みであり、リセット値に保持する必要があります。

ビット7:0　**PARTNUM[7:0]**：部品番号のビット [7:0]

0x55：STM32U535/545
0x76：STM32U5Fx/5Gx
0x81：STM32U59x/5Ax
0x82：STM32U575/585

MCU ROM CoreSightペリフェラルIDレジスタ1（MCUROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8　予約済みであり、リセット値に保持する必要があります。

ビット7:4　**JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0x0：STMicroelectronics JEDECコード

ビット3:0　**PARTNUM[11:8]**：部品番号のビット [11:8]

0x4：STM32U5シリーズ

MCU ROM CoreSightペリフェラルIDレジスタ2（MCUROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョンr0p0

ビット3 **JEDEC**：JEDEC割当て値
1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x2：STMicroelectronics JEDECコード

MCU ROM CoreSightペリフェラルIDレジスタ3（MCUROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスはありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

MCU ROM CoreSightコンポーネントIDレジスタ0（MCUROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

MCU ROM CoreSightペリフェラルIDレジスタ1（MCUROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス

0x1：ROMテーブルコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

MCU ROM CoreSightコンポーネントIDレジスタ2（MCUROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

MCU ROM CoreSightコンポーネントIDレジスタ3（MCUROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.5.2 MCU ROMテーブルレジスタマップ

表790.MCU ROMテーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	MCUROM_MEMTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD0	MCUROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD4-FDC	予約済み	予約済み																															
0xFE0	MCUROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0xFE4	MCUROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
	リセット値																									0	0	0	0	0	1	0	0
0xFE8	MCUROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID[6:4]		
	リセット値																									0	0	0	0	1	0	1	0
0xFEC	MCUROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	MCUROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	MCUROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	MCUROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	MCUROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 788：MCU ROMテーブルを参照してください。

75.5.3 プロセッサROMテーブルレジスタ

CPU ROMメモリタイプレジスタ（CPUROM_MEMTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
															r

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **SYSMEM**：システムメモリ

1：このバス上にシステムメモリがあります。

CPU ROM CoreSightペリフェラルIDレジスタ4（CPUROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDEC継続コード

CPU ROM CoreSightペリフェラルIDレジスタ0（CPUROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 00C9

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0xC9：Cortex-M33

CPU ROM CoreSightペリフェラルIDレジスタ1（CPUROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]

0x4：Cortex-M33

CPU ROM CoreSightペリフェラルIDレジスタ2（CPUROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョンr0p0

ビット3 **JEDEC**：JEDEC割当て値
1：JEDECによって指定された設計者ID

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

CPU ROM CoreSightペリフェラルIDレジスタ3（CPUROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスはありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CPU ROM CoreSightコンポーネントIDレジスタ0（CPUROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

CPU ROM CoreSightペリフェラルIDレジスタ1（CPUROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス

0x1：ROMテーブルコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

CPU ROM CoreSightコンポーネントIDレジスタ2（CPUROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

CPU ROM CoreSightコンポーネントIDレジスタ3（CPUROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.5.4 プロセッサROMテーブルレジスタマップ

表791.CPU ROMテーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	CPUROM_ MEMTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD4-FDC	予約済み	予約済み																															
0xFD0	CPUROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFE0	CPUROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									1	1	0	0	1	0	0	1
0xFE4	CPUROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	1	0	0
0xFE8	CPUROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]					JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	CPUROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CPUROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	CPUROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	CPUROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CPUROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブルを参照してください。

75.6 データウォッチポイントおよびトレースユニット（DWT）

DWTには4つのコンパレータがあり、それぞれ以下の役割に使用できます。

- ウォッチポイント
- ETMトリガ
- PCサンプリングトリガ
- データアドレスサンプリングトリガ
- データコンパレータ（COMP 1のみ）
- クロックサイクルカウンタコンパレータ（COMP 0のみ）

DWTは以下をカウントするためのカウンタも内蔵しています。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット（LSU）の動作数
- スリープサイクル数
- 命令当たりのサイクル数
- 割込みオーバーヘッドの回数

DWTコンパレータは、DWT_COMPxRに保持されている値と次のいずれかを比較します。

- データアドレス
- 命令アドレス
- データ値
- サイクルカウント値（COMP 0のみ）

アドレス照合の場合、コンパレータはマスクを使用することができるため、あるレンジのアドレスと一致します。

照合が成立すると、コンパレータは以下のうちの1つを生成します。

- 1つ以上のDWTデータトレースパケットで、以下を1つ以上含むもの。
  - データアクセスを伴う命令のアドレス
  - アドレスオフセット（データアクセスアドレスのビット [15:0] ）
  - 一致したデータ値
- PC値またはアクセスしたデータアドレスのいずれかで発生するウォッチポイントデバッグイベント
- DWTユニット外での一致を信号で伝えるCMPMATCH[N] イベント

ウォッチポイントデバッグイベントが発生すると、DebugMonitor例外が生成されるか、またはプロセッサが停止してデバッグステートに入ります。

DWTの使用方法の詳細については、Armv8-M Architecture Reference Manual[3]を参照してください。

75.6.1 DWTレジスタ

DWTレジスタはアドレスレンジ0xE000 1000 ～ 0xE000 1FFCに配置されています。

DWT制御レジスタ（DWT_CTRLR）

アドレスオフセット：0x000

リセット値：0x4000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCOMP[3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	CYCDISS	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA
r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PCSAMPLENA	SYNCTAP[1:0]		CYCTAP	POSTINIT[3:0]				POSTRESET[3:0]				CYCCNTENA
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:28 **NUMCOMP[3:0]**：実装されるコンパレータの数（読出し専用）
0x4：4つのコンパレータ

ビット27 **NOTRCPKT**：トレースサンプリングおよび例外トレースのサポート（読出し専用）
0：サポートされます。

ビット26 **NOEXTTRIG**：外部一致信号、CMPMATCHのサポート（読出し専用）
0：サポートされます。

ビット25 **NOCYCCNT**：サイクルカウンタのサポート（読出し専用）
0：サポートされます。

ビット24 **NOPRFCNT**：プロファイリングカウンタのサポート（読出し専用）
0：サポートされます。

ビット23 **CYCDISS**：セキュアなサイクルカウンタの無効化
サイクルカウンタをセキュアモードで無効化するかどうかを制御します。
0：影響なし。
1：プロセッサがセキュアステートである場合にサイクルカウンタのインクリメントを無効にします。

ビット22 **CYCEVTENA**：POSTCNTアンダーフローイベントカウンタのパケット生成の有効化
0：無効
1：有効

ビット21 **FOLDEVTENA**：フォールドされた命令数カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット20 **LSUEVTENA**：LSUカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット19 **SLEEPEVTENA**：スリープカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット18 **EXCEVTENA**：例外オーバーヘッドカウンタのオーバーフローイベント生成の有効化

0：無効

1：有効

ビット17 **CPIEVTENA**：CPIカウンタのオーバーフローイベント生成の有効化

0：無効

1：有効

ビット16 **EXCTRCENA**：例外トレース生成の有効化

0：無効

1：有効

ビット15:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **PCSAMPLENA**：周期的PCサンプルパケット生成用のタイマとして使用するためのPOSTCNTカウンタの有効化

0：無効

1：有効

ビット11:10 **SYNCTAP[1:0]**：CYCCNTカウンタ上の同期パケットカウンタのタップ位置

ここでのフィールドが、同期パケットの速度を決定します。

00：無効、同期パケットはありません。

01：CYCCNT[24] でタップします。

10：CYCCNT[26] でタップします。

11：CYCCNT[28] でタップします。

ビット9 **CYCTAP**：**CYCCNTカウンタ上のPOSTCNTのタップ位置の選択**

0：CYCCNT[6] でタップします。

1：CYCCNT[10] でタップします。

ビット8:5 **POSTINIT[3:0]**：POSTCNTカウンタの初期値

このフィールドへの書込みはPOSTCNTカウンタが有効であれば、無視されます。POSTINITを書き込む前にCYCEVTENAまたはPCSAMPLENAをリセットする必要があります。

ビット4:1 **POSTRESET[3:0]**：POSTCNTカウンタの値の再ロード

ビット0 **CYCCNTENA**：CYCCNTカウンタの有効化

0：無効

1：有効

DWTサイクルカウントレジスタ（DWT_CYCCNTR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CYCCNT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CYCCNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **CYCCNT[31:0]**：プロセッサのクロックサイクルカウンタ

DWT CPIカウントレジスタ（DWT_CPICNTR）

アドレスオフセット：0x008

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPICNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **CPICNT[7:0]**：CPIカウンタ

複数サイクルの命令を実行するために必要な追加サイクル数（DWT_LSUCNTRによって記録されるものを除く）をカウントし、命令フェッチストールがあればカウントします。

DWT例外カウントレジスタ（DWT_EXCCNTR）

アドレスオフセット：0x00C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXCCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **EXCCNT[7:0]**：例外オーバーヘッドサイクルカウンタ

例外処理に費やされるサイクル数をカウントします。

DWTスリープカウントレジスタ（DWT_SLPCNTR）

アドレスオフセット：0x010

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEPCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SLEEPCNT[7:0]**：スリープサイクルカウンタ

SLEEPモードに費やされるサイクル数をカウントします（WFI、WFE、Sleep-on-exit）。

DWT LSUカウントレジスタ（DWT_LSUCNTR）

アドレスオフセット：0x014

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **LSUCNT[7:0]**：ロードストアカウンタ
ロードおよびストア命令を実行するために必要な追加サイクル数をカウントします。

DWTフォールドカウントレジスタ（DWT_FOLDCNTR）

アドレスオフセット：0x018

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **FOLDCNT[7:0]**：フォールドされた命令数のカウンタ
サイクル数0の命令のたびにインクリメントします。

DWTプログラムカウンタサンプルレジスタ（DWT_PCSR）

アドレスオフセット：0x01C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIASAMPLE[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIASAMPLE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **EIASAMPLE[31:0]**：実行された命令アドレスのサンプル値
プログラムカウンタの現在値をサンプリングします。

DWTコンパレータxレジスタ（DWT_COMPxR）

アドレスオフセット：0x020 + 0x010 * x、（x = 0から3）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
COMP[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **COMP[31:0]**：比較のための基準値

DWT機能レジスタ0（DWT_FUNCTR0）

アドレスオフセット：0x028

リセット値：0x5800 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[4:0]					Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r			r								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DATAVSIZE[1:0]		Res.	Res.	Res.	Res.	ACTION[1:0]		MATCH[3:0]			
				rw	rw					rw	rw	rw	rw	rw	rw

ビット31:27 **ID[4:0]**：機能識別

コンパレータ0に適した機能を識別します。

0b01011：サイクルカウンタ、命令アドレス、データアドレス、および値が入力されたデータアドレス

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **MATCHED**：コンパレーター致

レジスタが最後に読み出されたあと、コンパレータの一致が発生したかどうかを示します。

0：一致していません。

1：一致が発生しました。

ビット23:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **DATAVSIZE[1:0]**：データ値のサイズ

データ値およびデータアドレスのコンパレータによって監視されているオブジェクトのサイズを定義します。

0x0：1バイト

0x1：2バイト

0x2：4バイト

0x3：予約済み

ビット9:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ACTION[1:0]**：一致時のアクション

0x0：トリガのみ。

0x1：デバッグイベントを生成します。

0x2：サイクルカウンタ、命令アドレス、データアドレス、データ値、またはリンクされたデータ値のコンパレータ用に、データトレース一致パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースデータ値パケットを生成します。

0x3：データアドレス制限コンパレータ用に、データトレースデータアドレスパケットを生成します。サイクルカウンタ、命令アドレス制限、またはデータアドレスのコンパレータ用に、データトレースPC値パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースPC値パケットとデータトレースデータ値パケットの両方を生成します。

ビット3:0 **MATCH[3:0]**：一致タイプ

コンパレータ0によって生成された一致タイプを制御します。

このフィールドに設定可能な値については、[3]を参照してください。

DWT機能レジスタ1（DWT_FUNCTR1）

アドレスオフセット：0x038

リセット値：0xD000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[4:0]					Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r			r								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DATAVSIZE[1:0]		Res.	Res.	Res.	Res.	ACTION[1:0]		MATCH[3:0]			
				rw	rw					rw	rw	rw	rw	rw	rw

ビット31:27 **ID[4:0]**：機能識別

コンパレータ1に適した機能を識別します。

0b11010：命令アドレス、命令アドレス制限、データアドレス、データアドレス制限、および値が入力されたデータアドレス

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **MATCHED**：コンパレーター一致

レジスタが最後に読み出されたあと、コンパレータの一致が発生したかどうかを示します。

0：一致していません。

1：一致が発生しました。

ビット23:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **DATAVSIZE[1:0]**：データ値のサイズ

データ値およびデータアドレスのコンパレータによって監視されているオブジェクトのサイズを定義します。

0x0：1バイト

0x1：2バイト

0x2：4バイト

0x3：予約済み

ビット9:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ACTION[1:0]**：一致時のアクション

0x0：トリガのみ。

0x1：デバッグイベントを生成します。

0x2：サイクルカウンタ、命令アドレス、データアドレス、データ値、またはリンクされたデータ値のコンパレータ用に、データトレース一致パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースデータ値パケットを生成します。

0x3：データアドレス制限コンパレータ用に、データトレースデータアドレスパケットを生成します。サイクルカウンタ、命令アドレス制限、またはデータアドレスのコンパレータ用に、データトレースPC値パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースPC値パケットとデータトレースデータ値パケットの両方を生成します。

ビット3:0 **MATCH[3:0]**：一致タイプ

コンパレータ1によって生成された一致タイプを制御します。

このフィールドに設定可能な値については、[3]を参照してください。

DWT機能レジスタ2（DWT_FUNCTR2）

アドレスオフセット：0x048

リセット値：0x5000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[4:0]					Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r			r								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DATAVSIZE[1:0]		Res.	Res.	Res.	Res.	ACTION[1:0]		MATCH[3:0]			
				rw	rw					rw	rw	rw	rw	rw	rw

ビット31:27 **ID[4:0]**：機能識別

コンパレータ2に適した機能を識別します。

0b01010：命令アドレス、データアドレス、および値が入力されたデータアドレス

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **MATCHED**：コンパレーター一致

レジスタが最後に読み出されたあと、コンパレータの一致が発生したかどうかを示します。

0：一致していません。

1：一致が発生しました。

ビット23:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **DATAVSIZE[1:0]**：データ値のサイズ

データ値およびデータアドレスのコンパレータによって監視されているオブジェクトのサイズを定義します。

0x0：1バイト

0x1：2バイト

0x2：4バイト

0x3：予約済み

ビット9:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ACTION[1:0]**：一致時のアクション

0x0：トリガのみ。

0x1：デバッグイベントを生成します。

0x2：サイクルカウンタ、命令アドレス、データアドレス、データ値、またはリンクされたデータ値のコンパレータ用に、データトレース一致パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースデータ値パケットを生成します。

0x3：データアドレス制限コンパレータ用に、データトレースデータアドレスパケットを生成します。サイクルカウンタ、命令アドレス制限、またはデータアドレスのコンパレータ用に、データトレースPC値パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースPC値パケットとデータトレースデータ値パケットの両方を生成します。

ビット3:0 **MATCH[3:0]**：一致タイプ

コンパレータ2によって生成された一致タイプを制御します。

このフィールドに設定可能な値については、[3]を参照してください。

DWT機能レジスタ3（DWT_FUNCTR3）

アドレスオフセット：0x058

リセット値：0xF000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[4:0]					Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r			r								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DATAVSIZE[1:0]		Res.	Res.	Res.	Res.	ACTION[1:0]		MATCH[3:0]			
				rw	rw					rw	rw	rw	rw	rw	rw

ビット31:27 **ID[4:0]**：機能識別

コンパレータ2に適した機能を識別します。

0b11110：命令アドレス、命令アドレス制限、データアドレス、データアドレス制限、データ値、リンクされたデータ値、および値が入力されたデータアドレス

ビット26:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **MATCHED**：コンパレーター一致

レジスタが最後に読み出されたあと、コンパレータの一致が発生したかどうかを示します。

0：一致していません。

1：一致が発生しました。

ビット23:12 予約済みであり、リセット値に保持する必要があります。

ビット11:10 **DATAVSIZE[1:0]**：データ値のサイズ

データ値およびデータアドレスのコンパレータによって監視されているオブジェクトのサイズを定義します。

0x0：1バイト

0x1：2バイト

0x2：4バイト

0x3：予約済み

ビット9:6 予約済みであり、リセット値に保持する必要があります。

ビット5:4 **ACTION[1:0]**：一致時のアクション

0x0：トリガのみ。

0x1：デバッグイベントを生成します。

0x2：サイクルカウンタ、命令アドレス、データアドレス、データ値、またはリンクされたデータ値のコンパレータ用に、データトレース一致パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースデータ値パケットを生成します。

0x3：データアドレス制限コンパレータ用に、データトレースデータアドレスパケットを生成します。サイクルカウンタ、命令アドレス制限、またはデータアドレスのコンパレータ用に、データトレースPC値パケットを生成します。値が入力されたデータアドレスのコンパレータ用に、データトレースPC値パケットとデータトレースデータ値パケットの両方を生成します。

ビット3:0 **MATCH[3:0]**：一致タイプ

コンパレータ2によって生成された一致タイプを制御します。

このフィールドに設定可能な値については、[3]を参照してください。

DWTデバイスタイプアーキテクチャレジスタ（DWT_DEVARCHR）

アドレスオフセット：0xFC8

リセット値：0x4770：1A02

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARCHITECT[10:0]											PRESENT	REVISION[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARCHVER[3:0]				ARCHPART[11:0]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 **ARCHITECT[10:0]**：アーキテクチャ JEP106コード

0x23B：JEP106継続コード0x4、JEP106 IDコード0x3B。Arm限定。

ビット20 **PRESENT**：DWT_DEVARCHレジスタの有無

0x1：あり

ビット19:16 **REVISION[3:0]**：アーキテクチャのリビジョン

0x0：DWTアーキテクチャ v2.0

ビット15:12 **ARCHVER[3:0]**：アーキテクチャのバージョン

0x1：DWTアーキテクチャ v2.0

ビット11:0 **ARCHPART[11:0]**：アーキテクチャパート

0xA02：DWTアーキテクチャ

DWTデバイスタイプレジスタ（DWT_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUB[3:0]**：サブタイプ
0x0：その他

ビット3:0 **MAJOR[3:0]**：メジャータイプ
0x0：その他

DWT CoreSightペリフェラルIDレジスタ4（DWT_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ
0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード
0x4：Arm JEDECコード

DWT CoreSightペリフェラルIDレジスタ0（DWT_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：Cortex-M33 DWT部品番号

DWT CoreSightペリフェラルIDレジスタ1（DWT_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]

0xD：Cortex-M33 DWT部品番号

DWT CoreSightペリフェラルIDレジスタ2（DWT_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

DWT CoreSightペリフェラルIDレジスタ3（DWT_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

DWT CoreSightコンポーネントIDレジスタ0（DWT_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

DWT CoreSightペリフェラルIDレジスタ1（DWT_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス

0x9：デバッグコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

DWT CoreSightコンポーネントIDレジスタ2（DWT_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

DWT CoreSightコンポーネントIDレジスタ3（DWT_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.6.2 DWTレジスタマップ

DWTレジスタはアドレスレンジ0xE000 1000 ～ 0xE000 1FFCに配置されています。

表792.DWTレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DWT_CTRLR	NUMCOMP [3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	CYCDISS	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA	Res.	Res.	Res.	PCSAMPLENA	SYNCTAP[1:0]		CYCTAP	POSTINIT [3:0]				POSTPRESET [3:0]				CYCCNTENA
	リセット値	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	DWT_CYCCNTR	CYCCNT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	DWT_CPICNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPICNT[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x00C	DWT_EXCCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXCCNT[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x010	DWT_SLPCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEPCNT[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x014	DWT_LSUCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x018	DWT_FOLDCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x01C	DWT_PCSR	EIASAMPLE[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x020	DWT_COMP0R	COMP[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x024	予約済み	予約済み																															
0x028	DWT_FUNCTR0	ID[4:0]				Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATAVSIZE [1:0]		Res.	Res.	Res.	Res.	ACTION [1:0]		MATCH[3:0].			
	リセット値	0	1	0	1	1			0													0	0					0	0	0	0	0	0
0x02C	予約済み	予約済み																															
0x030	DWT_COMP1R	COMP[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x034	予約済み	予約済み																															
0x038	DWT_FUNCTR1	ID[4:0]				Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATAVSIZE [1:0]		Res.	Res.	Res.	Res.	ACTION [1:0]		MATCH[3:0].			
	リセット値	1	1	0	1	0			0													0	0					0	0	0	0	0	0
0x03C	予約済み	予約済み																															
0x040	DWT_COMP2R	COMP[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x044	予約済み	予約済み																															
0x048	DWT_FUNCTR2	ID[4:0]				Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATAVSIZE [1:0]		Res.	Res.	Res.	Res.	ACTION [1:0]		MATCH[3:0].			
	リセット値	0	1	0	1	0			0													0	0					0	0	0	0	0	0
0x04C	予約済み	予約済み																															

表792.DWTレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x050	DWT_COMP3R	COMP[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x054	予約済み	予約済み																															
0x058	DWT_FUNCTR3	ID[4:0]					Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATAVSIZE [1:0]		Res.	Res.	Res.	Res.	ACTION [1:0]		MATCH[3:0].			
	リセット値	1	1	1	1	0			0													0	0					0	0	0	0	0	0
0x05Cから0xFC4	予約済み	予約済み																															
0xFC8	DWT_DEVARCHR	ARCHITECT[10:0]											PRESENT	REVISION [3:0]				ARCHVER [3:0]				ARCHPART[11:0]											
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0	
0xFCC	DWT_DEVTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD0	DWT_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4から0xFDC	予約済み	予約済み																															
0xFE0	DWT_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	1	0	0	0	0	1
0xFE4	DWT_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	1	0	1
0xFE8	DWT_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]					JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	DWT_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	DWT_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	DWT_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	DWT_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	DWT_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブルを参照してください。

75.7 計装トレースマクロセル（ITM）

ITMはトレース情報をパケットとして生成します。3つのソースでパケットを生成できます。パケットが複数のソースから同時に生成される場合、ITMはパケットの出力順番についてアービトレーションを行います。3つのソースを優先度の高いものから順に挙げます。

- ソフトウェアトレース

  ソフトウェアは、32 x 32bitのITMスティムラスレジスタのいずれかに直接書込みを行って、パケットを生成します。各ポートの許可レベルはプログラムで設定できます。ソフトウェアが有効なスティムラスポートに書き込むと、ITMはFIFOに書き込むパケットの中に、ポートのID、書込みアクセスのサイズ、および書き込まれたデータを統合します。ITMはFIFOからトレースバスにパケットを出力します。スティムラスポートレジスタを読み出すと、ビット0のスティムラスレジスタのステータス（エンプティまたはペンディング）を返します。

- ハードウェアトレース

  DWTはデータトレースイベント、すなわちPCサンプルやパフォーマンスプロファイリングカウンタの折り返しに応じて、トレースパケットを生成します。ITMがこれらのパケットをトレースバスに出力します。

- ローカルタイムスタンプ

  ITMは（分周済みの）プロセッサクロックが供給される21bitカウンタを内蔵しています。カウンタの値はトレースバスのタイムスタンプパケットに出力されます。タイムスタンプパケットが生成されるたびに、カウンタはゼロにリセットされます。よって、タイムスタンプは前のタイムスタンプパケットからの経過時間となります。

ITMとその使用方法の詳細については、[3]を参照してください。

75.7.1 ITMレジスタ

ITMレジスタはアドレスレンジ0xE000 0000 ～ 0xE000 0FFCに配置されています。

ITMスティムラスレジスタx （ITM_STIMRx）

アドレスオフセット：0x000 + 0x4 * x、（x = 0から31）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STIMULUS[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STIMULUS[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	rw	rw

ビット31:0 **STIMULUS[31:0]**：トレース出力データ

書込み時、書込みデータは、ソフトウェアイベントパケットとしてトレースバスに出力されます。

読出し時：

- ビット1は無効化フラグです。
  - 0：スティムラスポートおよびITMは有効です。
  - 1：スティムラスポートおよびITMは無効です。
- ビット0はFIFOレディインジケータです。
  - 0：スティムラスポートのバッファはフルです（またはポートが無効）。
  - 1：スティムラスポートは新しい書込みデータを受け付けることができます。

ITMトレース有効レジスタ（ITM_TER）

アドレスオフセット：0xE00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STIMENA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STIMENA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:0 **STIMENA[31:0]**：スティムラスポートの有効化

各ビットx（0 ～ 31）は、ITM_STIMRxレジスタと関連のあるスティムラスポートを有効にします。

0：ポートを無効化

1：ポートを有効化

ITMトレース特権レジスタ（ITM_TPR）

アドレスオフセット：0xE40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIVMASK[3:0]			
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PRIVMASK[3:0]**：ITMスティムラスポートへの非特権のアクセスを無効にします。

各ビットは8つのスティムラスポートを制御します。

XXX0：ポート0～7では非特権のアクセスが許可されます。

XXX1：ポート0～7では特権アクセスのみが許可されます。

XX0X：ポート8～15では非特権のアクセスが許可されます。

XX1X：ポート8～15では特権アクセスのみが許可されます。

X0XX：ポート16～23では非特権のアクセスが許可されます。

X1XX：ポート16～23では特権アクセスのみが許可されます。

0XXX：ポート24～31では非特権のアクセスが許可されます。

1XXX：ポート24～31では特権アクセスのみが許可されます。

ITMトレース制御レジスタ（ITM_TCR）

アドレスオフセット：0xE80

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]						
								r	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSPRESCALE[1:0]		Res.	Res.	STALLENA	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
						rw	rw			rw	r	rw	rw	rw	rw

ビット31:24 予約済みであり、リセット値に保持する必要があります。

ビット23 **BUSY**：ITMが現在イベント処理中かどうかを示します。

0：非ビジー

1：ビジー

ビット22:16 **TRACEBUSID[6:0]**：マルチソーストレースストリームのフォーマット識別子

マルチソーストレースを使用している場合、デバッガはこのフィールドにゼロ以外の値を書き込まなければなりません。

注： システム内のトレースソースにはそれぞれ異なるIDを使用してください。

ビット15:10 予約済みであり、リセット値に保持する必要があります。

ビット9:8 **TSPRESCALE[1:0]**：トレースパケットのリファレンスクロックとともに使用するローカルタイムスタンププリスケーラ

0x0：プリスケールなし

0x1：4分周されます。

0x2：16分周されます。

0x3：64分周されます。

ビット7:6 予約済みであり、リセット値に保持する必要があります。

ビット5 **STALLENA**：ストールの有効化

0：ITM出力がストールしている場合に、ハードウェアソースパケットをドロップし、オーバーフローを生成します。

1：プロセッサをストールし、データトレースパケットの供給を保証します。

ビット4 **SWOENA**：SWOの有効化

タイムスタンプカウンタの非同期クロックを有効にします（読出し専用）。

0：タイムスタンプカウンタはプロセッサクロックを使用します。

ビット3 **TXENA**：送信の有効化

DWTユニットからトレースポートへのハードウェアイベントパケットの転送を有効にします。

0：無効

1：有効

ビット2 **SYNCENA**：同期パケット送信の有効化

このビットをセットするデバッガは、正しい同期速度にするために、DWT_CTRLR.SYNCTAPフィールドも設定する必要があります。

0：無効

1：有効

ビット1 **TSENA**：ローカルタイムスタンプ生成の有効化

0：無効

1：有効

ビット0 **ITMENA**：ITMの有効化
0：無効
1：有効

ITMデバイスタイプアーキテクチャレジスタ（ITM_DEVARCHR）

アドレスオフセット：0xFBC

リセット値：0x4770 1A01

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARCHITECT[10:0]											PRESENT	REVISION[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARCHVER[3:0]				ARCHPART[11:0]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 **ARCHITECT[10:0]**：アーキテクチャ JEP106コード
0x23B：JEP106継続コード0x4、JEP106 IDコード0x3B。Arm限定。

ビット20 **PRESENT**：DEVARCHレジスタの有無
0x1：あり

ビット19:16 **REVISION[3:0]**：アーキテクチャのリビジョン
0x0：ITMアーキテクチャ v2.0

ビット15:12 **ARCHVER[3:0]**：アーキテクチャのバージョン
0x1：ITMアーキテクチャ v2.0

ビット11:0 **ARCHPART[11:0]**：アーキテクチャパート
0xA01：ITMアーキテクチャ

ITMデバイスタイプレジスタ（ITM_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0043

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUB[3:0]**：サブタイプ
0x4：バスアクティビティから生成されたスティムラスであるバスに関連します。

ビット3:0 **MAJOR[3:0]**：メジャータイプ
0x3：トレースソース

ITM CoreSightペリフェラルIDレジスタ4（ITM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDECコード

ITM CoreSightペリフェラルIDレジスタ0（ITM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：ITM部品番号

ITM CoreSightペリフェラルIDレジスタ1（ITM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]
0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]
0xD：ITM部品番号

ITM CoreSightペリフェラルIDレジスタ2（ITM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

ITM CoreSightペリフェラルIDレジスタ3（ITM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

ITM CoreSightコンポーネントIDレジスタ0（ITM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]
0x0D：共通の識別値

ITM CoreSightペリフェラルIDレジスタ1（ITM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス
0xE：トレースジェネレータのコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]
0x0：共通の識別値

ITM CoreSightコンポーネントIDレジスタ2（ITM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]
0x05：共通の識別値

ITM CoreSightコンポーネントIDレジスタ3（ITM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.7.2 ITMレジスタマップ

ITMレジスタはアドレスレンジ0xE000 0000 ～ 0xE000 0FFCに配置されています。

表793. ITMレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000から0x07C	ITM_STIMR0からITM_STIMR31	STIMULUS[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x07C - 0xDFC	予約済み	予約済み																															
0xE00	ITM_TER	STIMENA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE04 ~0xE3C	予約済み	予約済み																															
0xE40	ITM_TPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIVMASK[3:0]			
	リセット値																													0	0	0	0
0xE44 ~0xE7C	予約済み	予約済み																															
0xE80	ITM_TCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	TSPRESCALE[1:0]		Res.	Res.	STALLENA	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
	リセット値									0	0	0	0	0	0	0	0							0	0			0	0	0	0	0	0
0xE84 - 0xFB8	予約済み	予約済み																															
0xFBC	ITM_DEVARCHR	ARCHITECT[10:0]											PRESENT	REVISION[3:0]				ARCHVER[3:0]				ARCHPART[3:0]											
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	1
0xFC0 ~0xFC8	予約済み	予約済み																															
0xFCC	ITM_DEVTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
	リセット値																									0	1	0	0	0	0	1	1
0xFD0	ITM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 - 0xFDC	予約済み	予約済み																															
0xFE0	ITM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	1	0	0	0	0	1
0xFE4	ITM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	1	0	1
0xFE8	ITM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID[6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	ITM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	ITM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	ITM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	1	1	0	0	0	0	0

表793. ITMレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF8	ITM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	ITM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブルを参照してください。

75.8 ブレークポイントユニット（BPU）

BPUによって、ハードウェアブレークポイントをセットできます。命令フェッチアドレスを監視する8つのコンパレータが含まれます。一致の発生時に、命令コンパレータはブレークポイント命令を生成するよう設定できます。

ブレークポイントユニットとその使用方法の詳細については、[3]を参照してください。

75.8.1 BPUレジスタ

BPUレジスタはアドレスレンジ0xE0002000～0xE0002FFCに配置されています。

BPU制御レジスタ（BPU_CTRLR）

アドレスオフセット：0x000

リセット値：0x1000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REV[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NUM_CODE[6:4]			Res.	Res.	Res.	Res.	NUM_CODE[3:0]				Res.	Res.	KEY	ENABLE
	r	r	r					r	r	r	r			rw	rw

ビット31:28 **REV[3:0]**：リビジョン番号
0x1：BPUバージョン2

ビット27:15 予約済みであり、リセット値に保持する必要があります。

ビット14:12、7:4 **NUM_CODE[6:0]**：サポートされる命令アドレスコンパレータの数
0x08：8つの命令コンパレータがサポートされます。

ビット11:8、3:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **KEY**：書込み保護キー
FPB_CTRLRレジスタへの書込みは、このビットが「1」にセットされていない場合は無視されます。

ビット0 **ENABLE**：FPBの有効化
0：無効
1：有効

BPUコンパレータxレジスタ（BPU_COMPxR）

アドレスオフセット：0x008 + 0x4 * x、（x = 0から7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BPADDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BPADDR[15:1]															BE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:1 **BPADDR[31:1]**：ブレークポイントアドレス

ビット0 **BE**：ブレークポイントの有効化
0：無効
1：有効

BPUデバイスタイプアーキテクチャレジスタ（BPU_DEVARCHR）

アドレスオフセット：0xFBC

リセット値：0x4770 1A03

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARCHITECT[10:0]											PRESENT	REVISION[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARCHVER[3:0]				ARCHPART[11:0]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 **ARCHITECT[10:0]**：アーキテクチャ JEP106コード
0x23B：JEP106継続コード0x4、JEP106 IDコード0x3B。Arm限定。

ビット20 **PRESENT**：DEVARCHレジスタの有無
0x1：あり

ビット19:16 **REVISION[3:0]**：アーキテクチャのリビジョン
0x0：BPUアーキテクチャ v2.0

ビット15:12 **ARCHVER[3:0]**：アーキテクチャのバージョン
0x1：BPUアーキテクチャ v2.0

ビット11:0 **ARCHPART[11:0]**：アーキテクチャパート
0xA03：BPUアーキテクチャ

BPUデバイスタイプレジスタ（BPU_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUB[3:0]**：サブタイプ
0x0：その他

ビット3:0 **MAJOR[3:0]**：メジャータイプ
0x0：その他

BPU CoreSightペリフェラルIDレジスタ4（BPU_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDECコード

BPU CoreSightペリフェラルIDレジスタ0（BPU_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：BPU部品番号

BPU CoreSightペリフェラルIDレジスタ1（BPU_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]
0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]
0xD：BPU部品番号

BPU CoreSightペリフェラルIDレジスタ2（BPU_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

BPU CoreSightペリフェラルIDレジスタ3（BPU_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

BPU CoreSightコンポーネントIDレジスタ0（BPU_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]
0x0D：共通の識別値

BPU CoreSightペリフェラルIDレジスタ1（BPU_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス
0x9：デバッグコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]
0x0：共通の識別値

BPU CoreSightコンポーネントIDレジスタ2（BPU_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]
0x05：共通の識別値

BPU CoreSightコンポーネントIDレジスタ3（BPU_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.8.2 BPUレジスタマップ

BPUレジスタはアドレスレンジ0xE000 2000 ～ 0xE000 2FFCに配置されています。

表794.BPUレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	BPU_CTRLR	REV[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUM_CODE [6:4]			Res.	Res.	Res.	Res.	NUM_CODE [3:0]				Res.	Res.	KEY	ENABLE
	リセット値	0	0	0	1														0	0	0					1	0	0	0			0	0
0x004	予約済み	予約済み																															
0x008から 0x024	BPU_COMP0Rから BPU_COMP7R	BPADDR[31:1]																															BE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028 - 0xFB8	予約済み	予約済み																															
0xFBC	BPU_DEVARCHR	ARCHITECT[10:0]											PRESENT	REVISION [3:0]				ARCHVER [3:0]				ARCHPART[11:0]											
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	1
0xFC0 - 0xFC8	予約済み	予約済み																															
0xFCC	BPU_DEVTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD0	BPU_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 - 0xFDC	予約済み	予約済み																															
0xFE0	BPU_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	1	0	0	0	0	1
0xFE4	BPU_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	1	0	1
0xFE8	BPU_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]					JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	BPU_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	BPU_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	BPU_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	BPU_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	BPU_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブルを参照してください。

75.9 組み込みトレースマクロセル（ETM）

ETMはCPUと密接に連結されたCoreSightコンポーネントです。ETMはCortex-M33コアの実行をトレースすることができるトレースパケットを生成します。STM32U5シリーズでは、ETMは命令トレース専用に設定されています。トレース情報にデータアクセスは含まれません。

ETMはプロセッサのトレースインタフェース経由でCPUから以下の情報を受信します。

- 同一サイクルで実行される命令の数
- プログラムフローの変更
- プロセッサの現在の命令ステート
- ロードおよびストア命令によってアクセスされるメモリ位置のアドレス
- 転送のタイプ、方向、およびサイズ
- 条件コードに関する情報
- 例外情報
- 割込み待ちステートに関する情報

詳細については、Arm CoreSight ETM-M33 Technical Reference Manual[5]を参照してください。

75.9.1 ETMレジスタ

ETMレジスタはアドレスレンジ0xE004 1000 ～ 0xE004 1FFCに配置されています。

ETMプログラミング制御レジスタ（ETM_PRGCTLR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **EN**：トレースユニットの有効化

0：無効

1：有効

ETMステータスレジスタ（ETM_STATR）

アドレスオフセット：0x00C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PMSTABLE	IDLE
														r	r

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **PMSTABLE**：安定性ステータス

ETM-M33レジスタが安定しており、読出し可能であることを示します。

0：安定していません。

1：安定しています。

ビット0 **IDLE**：トレースユニットステータス

トレースユニットが非アクティブであることを示します。

0：アイドル状態ではありません。

1：アイドル状態です。

ETM設定レジスタ（ETM_CONFIGR）

アドレスオフセット：0x010

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RS	Res.	COND[5:0]						CCI	BB	Res.	Res.	Res.
			rw		rw	rw	rw	rw	rw	rw	rw	rw			

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **RS**：リターンスタックの有効化

0：無効

1：有効

ビット11 予約済みであり、リセット値に保持する必要があります。

ビット10:5 **COND[5:0]**：条件付き命令のトレース

0x0：条件付き命令のトレースは無効です。

0x1：条件付きロード命令をトレースします。

0x2：条件付きストア命令をトレースします。

0x3：条件付きロードおよびストア命令をトレースします。

0x7：すべての条件付き命令をトレースします。

ビット4 **CCI**：命令トレース時のサイクルカウント
0：無効
1：有効

ビット3 **BB**：分岐ブロードキャストモード
0：無効
1：有効

ビット2:0 予約済みであり、リセット値に保持する必要があります。

ETMイベント制御0レジスタ（ETM_EVENTCTL0R）

アドレスオフセット：0x020

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPE1	Res.	Res.	Res.	SEL1 [3:0]				TYPE0	Res.	Res.	Res.	SEL0 [3:0]			
rw				rw	rw	rw	rw	rw				rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15 **TYPE1**：event1のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット14:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **SEL1[3:0]**：TYPE1に基づくリソース番号
TYPE1の値に基づいて、リソース番号を選択します。
TYPE1 = 0のときは、SEL1[3:0] で定義された0 ～ 15の中から単一リソースが選択されます。
TYPE1 = 1のときは、SEL1[2:0] で定義されたブール組み合わせリソースペアが選択されます。

ビット7 **TYPE0**：event0のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット6:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **SEL0[3:0]**：TYPE0に基づくリソース番号
TYPE0の値に基づいて、リソース番号を選択します。
TYPE0 = 0のときは、SEL0[3:0] で定義された0 ～ 15の中から単一リソースが選択されます。
TYPE0 = 1のときは、SEL0[2:0] で定義されたブール組み合わせリソースペアが選択されます。

ETMイベント制御1レジスタ（ETM_EVENTCTL1R）

アドレスオフセット：0x024

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[1:0]	
			rw	rw										rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **LPOVERRIDE**：低電力ステートでの動作のオーバーライド

0：通常の低電力ステートの動作を行います。

1：リソースやイベントトレースの生成は、低電力ステートへの移行による影響を受けません。

ビット11 **ATB**：ATBトリガの有効化

0：無効

1：有効

ビット10:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **INSTEN[1:0]**：命令イベント生成

命令ストリームでのイベント要素の生成を有効にします。

0bX0：Event0はイベント要素の原因とはなりません。

0bX1：Event0は、その発生時にイベント要素の原因となります。

0b0X：Event1はイベント要素の原因とはなりません。

0b1X：Event1は、その発生時にイベント要素の原因となります。

ETMストール制御レジスタ（ETM_STALLCTLR）

アドレスオフセット：0x02C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	INSTPRIORITY	Res.	ISTALL	Res.	Res.	Res.	Res.	LEVEL[3:0]			
					rw		rw					rw	rw	rw	rw

ビット31:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **INSTPRIORITY**：命令トレースの優先順位

命令トレースのバッファスペースがLEVEL[3:0] を下回る場合に、命令トレースを優先します。

0：ETMは命令トレースを優先してはなりません。

1：ETMは命令トレースを優先することができます。

ビット9 予約済みであり、リセット値に保持する必要があります。

ビット8 **ISTALL**：プロセッサのストール

命令トレースのバッファスペースに基づいてプロセッサをストールします。

0：ETMはプロセッサをストールしてはなりません。

1：ETMはプロセッサをストールすることができます。

ビット7:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **LEVEL[3:0]**：ストールがアクティブになる閾値

このフィールドには4つのレベルがあります。このレベルは、ストールによって生じた侵入のレベルを最適化してFIFOオーバーフローのリスクとのバランスを取るために変更できます。

0x0：侵入はありませんが、FIFOオーバーフローのリスクが大きくなります。

.....

0xF：侵入レベルは最大ですが、FIFOオーバーフローのリスクは小さくなります。

ETM同期周期レジスタ（ETM_SYNCPR）

アドレスオフセット：0x034

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
											r	r	r	r	r

ビット31:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **PERIOD[4:0]**：同期周期

トレース同期リクエスト間のトレースのバイト数を、命令ストリームによって生成されたバイト数の合計として定義します。

0xA：1024バイト

ETMサイクルカウント制御レジスタ（ETM_CCCTLR）

アドレスオフセット：0x038

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	THRESHOLD[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **THRESHOLD[11:0]**：命令トレースサイクルカウントの閾値

命令トレースサイクルカウントの閾値をセットします。この閾値は、サイクルカウントトレースパケットの最小間隔を表します。

ETMトレース識別レジスタ（ETM_TRACEIDR）

アドレスオフセット：0x040

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACEID[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット31:7 予約済みであり、リセット値に保持する必要があります。

ビット6:0 **TRACEID[6:0]**：トレースバスに出力するトレース識別

このフィールドは、一意の値でプログラムすることで、システム内の他のトレースソースと区別する必要があります。

値0x00および0x70 ～ 0x7Fは予約済みです。

ETM ViewInstメイン制御レジスタ（ETM_VICTLR）

アドレスオフセット：0x080

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSSTATUS	Res.	EVENT[7:0]							
				rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **EXLEVEL_S[3:0]**：セキュアステートの例外レベル

セキュアステートで、対応する例外レベルについて命令トレースを有効にするかどうかを制御します。

0bXXX0：例外レベル0のセキュアステートにおいて、命令トレースは生成されません。

0bXXX1：例外レベル0のセキュアステートにおいて、命令トレースが生成されます。

0b0XXX：例外レベル3のセキュアステートにおいて、命令トレースは生成されません。

0b1XXX：例外レベル3のセキュアステートにおいて、命令トレースが生成されます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **TRCERR**：システムエラー例外のトレース

0：システムエラー例外の直前の命令または例外がトレースされる場合のみ、システムエラー例外はトレースされます。

1：システムエラー例外は常にトレースされます。

ビット10 **TRCRESET**：リセット例外のトレース

0：リセット例外の直前の命令または例外がトレースされる場合のみ、リセット例外はトレースされます。

1：リセット例外は常にトレースされます。

ビット9 **SSSTATUS**：開始／停止ロジックのステータス
0：停止されます。
1：開始されます。

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **EVENT[7:0]**：イベントセレクタ

ETMカウンタ再ロード値レジスタ0（ETM_CNTRLDVR0）

アドレスオフセット：0x140

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VALUE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **VALUE[15:0]**：カウンタ再ロード値
この値は、再ロードイベント発生のたびにカウンタにロードされます。

ETM識別レジスタ8（ETM_IDR8）

アドレスオフセット：0x180

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MAXSPEC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAXSPEC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **MAXSPEC[31:0]**：最大スペキュレーション深さ
命令トレースストリームの最大スペキュレーション深さを示します。これは、トレースストリームで同時にコミットされることがないP0要素の最大数です。
0x0：トレーススペキュレーションの最大深さは0です。

ETM識別レジスタ9（ETM_IDR9）

アドレスオフセット：0x184

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP0KEY [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP0KEY [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **NUMP0KEY[31:0]**：使用するP0右側キーの数

0x0：命令トレースで使用するP0右側キーはありません。

ETM識別レジスタ10（ETM_IDR10）

アドレスオフセット：0x188

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1KEY [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1KEY [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **NUMP1KEY[31:0]**：使用するP1右側キーの数（通常キーと特殊キーを含む）

0x0：命令トレースで使用するP1右側キーはありません。

ETM識別レジスタ11（ETM_IDR11）

アドレスオフセット：0x18C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1SPC [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1SPC [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **NUMP1SPC[31:0]**：使用する特殊なP1右側キーの数

0x0：設定で使用されている特殊なP1右側キーはありません。

ETM識別レジスタ12（ETM_IDR12）

アドレスオフセット：0x190

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDKEY[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDKEY[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **NUMCONDKEY[31:0]**：使用する条件付き命令右側キーの数（通常キーと特殊キーを含む）

0x1：条件付き命令右側キーが1つ実装されます。

ETM識別レジスタ13（ETM_IDR13）

アドレスオフセット：0x194

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDSPC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDSPC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **NUMCONDSPC[31:0]**：使用する特殊な条件付き命令右側キーの数

0x0：特殊な条件付き命令右側キーは実装されません。

ETM実装固有レジスタ0（ETM_IMSPECR0）

アドレスオフセット：0x1C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUPPORT[3:0]			
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **SUPPORT[3:0]**：実装固有の拡張機能のサポート

0x0：実装固有の拡張機能はサポートされません。

ETM識別レジスタ0（ETM_IDR0）

アドレスオフセット：0x1E0

リセット値：0x2800 06E1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	COMMOPT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRCEXDATA	QSUPP[1]
		r												r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
QSUPP[0]	Res.	CONDTYPE[1:0]		NUMEVENT[1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	TRCDATA[1:0]		INSTP0 [1:0]		Res.
r		r	r	r	r	r		r	r	r	r	r	r	r	

ビット31:30 予約済みであり、リセット値に保持する必要があります。

ビット29 **COMMOPT**：コミットフィールドの意味
一部のパケットにおけるコミットフィールドの意味を示します。
1：コミットモード1

ビット28:18 予約済みであり、リセット値に保持する必要があります。

ビット17 **TRCEXDATA**：例外のデータ転送のトレース
例外および返された例外のデータ転送のトレースに対するサポートを示します。
0：実装されません。

ビット16:15 **QSUPP[1:0]**：Q要素のサポート
0：サポートされません。

ビット14 予約済みであり、リセット値に保持する必要があります。

ビット13:12 **CONDTYPE[1:0]**：条件付き結果のトレース
条件付き結果のトレース方法を示します。
0：トレースユニットは、条件付き命令が条件コードチェックを通過または失敗したときにのみ示されます。

ビット11:10 NUMEVENT[1:0]：サポートされるイベントの数
0x1：2つのイベント

ビット9 **RETSTACK**：リターンスタックのサポート
1：2つのエントリリターンスタック

ビット8 予約済みであり、リセット値に保持する必要があります。

ビット7 **TRCCCI**：サイクルカウントのサポート
1：サイクルカウントが実装されます。

ビット6 **TRCCOND**：条件付き命令のサポート
1：条件付き命令のトレースが実装されます。

ビット5 **TRCBB**：分岐ブロードキャストのサポート
1：分岐ブロードキャストのトレースが実装されます。

ビット4:3 **TRCDATA[1:0]**：データトレースのサポート
0x0：データトレースはサポートされません。

ビット2:1 **INSTP0[1:0]**：P0要素としてのロードおよびストア命令のトレースのサポート
0x0：サポートされません。

ビット0 予約済みであり、リセット値に保持する必要があります。

ETM識別レジスタ1（ETM_IDR1）

アドレスオフセット：0x1E4

リセット値：0x4100 F421

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCARCHMAJ[3:0]				TRCARCHMIN[3:0]				REVISION[3:0]			
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:24 **DESIGNER[7:0]**：トレースユニット設計者
0x41：Arm

ビット23:12 予約済みであり、リセット値に保持する必要があります。

ビット11:8 **TRCARCHMAJ[3:0]**：メジャーなトレースユニットアーキテクチャのバージョン番号
0x4：ETMv4

ビット7:4 **TRCARCHMIN[3:0]**：マイナーなトレースユニットアーキテクチャのバージョン番号
0x2：マイナーリビジョン2

ビット3:0 **REVISION[3:0]**：実装のリビジョン番号
0x1：実装のリビジョン1

ETM識別レジスタ2（ETM_IDR2）

アドレスオフセット：0x1E8

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:1]			
			r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DASIZE[0]	VMIDSIZE[4:0]					CIDSIZE[4:0]					IASIZE[4:0]				
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:29 予約済みであり、リセット値に保持する必要があります。

ビット28:25 **CCSIZE[3:0]**：サイクルカウンタのサイズ
0x0：12bit

ビット24:20 **DVSIZE[4:0]**：データ値のサイズ
0x0：データ値のサイズはサポートされません。

ビット19:15 **DASIZE[4:0]**：データアドレスのサイズ
0x0：データアドレスのサイズはサポートされません。

ビット14:10 **VMIDSIZE[4:0]**：仮想マシンIDサイズ
0x0：仮想マシンIDのトレースは実装されません。

ビット9:5 **CIDSIZE[4:0]**：コンテキストIDサイズ
0x0：コンテキストIDのトレースは実装されません。

ビット4:0 **IASIZE[4:0]**：命令アドレスのサイズ
0x4：最大32bitのアドレスサイズ

ETM識別レジスタ3（ETM_IDR3）

アドレスオフセット：0x1EC

リセット値：0x0F09 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NOOVERFLOW	NUMPROC[2:0]			SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]			
r	r	r	r	r	r	r	r					r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CCITMIN[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31 **NOOVERFLOW**：ETM_STALLCTLR.NOOVERFLOWの実装
0：実装されません。

ビット30:28 **NUMPROC[2:0]**：トレースに利用できるプロセッサの数
0x0：1つのプロセッサ

ビット27 **SYSSTALL**：プロセッサのストール制御のシステムサポート
1：システムでストール制御をサポートします。

ビット26 **STALLCTL**：ストール制御のサポート
1：ETM_STALLCTLRが実装されます。

ビット25 **SYNCPR**：トレース同期周期のサポート
1：命令トレースのみの設定ではETM_SYNCPRは読出し専用です。トレース同期周期は固定です。

ビット24 **TRCERR**：ETM_VICTLR.TRCERRの実装
0x1：実装されます。

ビット23:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **EXLEVEL_S[3:0]**：特権レベルの実装
0x9：特権レベル、スレッドとハンドラーが実装されます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **CCITMIN[11:0]**：TRCCCCTLR.THRESHOLDにプログラム可能な最小値
最小サイクルカウントの閾値を定義します。
0x4：最小値は4命令トレースサイクルです。

ETM識別レジスタ4（ETM_IDR4）

アドレスオフセット：0x1F0

リセット値：0x0011 4000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMVMIDC[3:0]				NUMCIDC[3:0]				NUMSSCC[3:0]				NUMRSPAIR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMPC[3:0]				Res.	Res.	Res.	SUPPDAC	NUMDVC[3:0]				NUMACPAIRS[3:0]			
r	r	r	r				r	r	r	r	r	r	r	r	r

ビット31:28 **NUMVMIDC[3:0]**：仮想マシンID（VMID）コンパレータの数
0x0：VMIDコンパレータは実装されません。

ビット27:24 **NUMCIDC[3:0]**：コンテキストIDコンパレータの数
0x0：コンテキストIDコンパレータはサポートされません。

ビット23:20 **NUMSSCC[3:0]**：シングルショットコンパレータ制御の数
0x1：シングルショットコンパレータ制御が1つ実装されます。

ビット19:16 **NUMRSPAIR[3:0]**：リソース選択ペアの数
0x1：リソース選択ペアが2つ実装されます。

ビット15:12 **NUMPC[3:0]**：DWT用のプロセッサコンパレータ入力の数
0x4：プロセッサコンパレータ入力が4つ実装されます。

ビット11:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **SUPPDAC**：データアドレスの比較
0：データアドレスの比較はサポートされません。

ビット7:4 **NUMDVC[3:0]**：データ値コンパレータの数
0x0：データ値コンパレータは実装されません。

ビット3:0 **NUMACPAIRS[3:0]**：アドレスコンパレータペアの数
0x0：アドレスコンパレータペアは実装されません。

ETM識別レジスタ5（ETM_IDR5）

アドレスオフセット：0x1F4

リセット値：0x90C7 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE[2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE[5:0]					
r	r	r	r	r	r	r		r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NUMEXTINSEL[2:0]			NUMEXTIN[8:0]								
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31 **REDFUNCNTR**：低減機能カウンタ
1：低減機能カウンタとしてカウンタ0が実装されます。

ビット30:28 **NUMCNTR[2:0]：カウンタの数**
0x1：カウンタが1つ実装されます。

ビット27:25 **NUMSEQSTATE[2:0]**：シーケンサのステート数
0x0：シーケンサのステートは実装されません。

ビット24 予約済みであり、リセット値に保持する必要があります。

ビット23 **LPOVERRIDE**：低電力ステートのオーバーライドのサポート
1：低電力ステートのオーバーライドのサポートが実装されます。

ビット22 **ATBTRIG**：ATBトリガのサポート
1：ATBトリガのサポートが実装されます。

ビット21:16 **TRACEIDSIZE[5:0]**：トレース識別のビット数
0x7：7bitトレース識別が実装されます。

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:9 **NUMEXTINSEL[2:0]**：外部入力セレクタの数
0x0：外部入力セレクタは実装されません。

ビット8:0 **NUMEXTIN[8:0]**：外部入力の数
0x004：4つの外部入力が実装されます。

ETMリソースレジスタ2（ETM_RSCTLR2）

アドレスオフセット：0x208

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP[2:0]		
										rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:22 予約済みであり、リセット値に保持する必要があります。

ビット21 **PAIRINV**：組み合わせリソースペアの結果の反転
0：反転されません。
1：反転されます。

ビット20 **INV**：選択されたリソースの反転
0：反転されません。
1：反転されます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **GROUP[2:0]**：リソースグループの選択
0x0：外部入力セレクタ（0 ～ 3を選択）
0x1：プロセッサDWTコンパレータ要素からの入力（0 ～ 3を選択）
0x2：ゼロのカウンタ（0を選択）
0x3：シングルショットコンパレータ（0を選択）
その他：予約済み

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SELECT[7:0]**：さらなるリソースの選択
GROUP[2:0] で選択したグループから1つ以上のリソースを選択します。

ETMリソースレジスタ3（ETM_RSCTLR3）

アドレスオフセット：0x20C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INV	Res.	GROUP[2:0]		
											rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット31:21 予約済みであり、リセット値に保持する必要があります。

ビット20 **INV**：選択されたリソースの反転
0：反転されません。
1：反転されます。

ビット19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **GROUP[2:0]**：リソースグループの選択
0x0：外部入力セレクタ（0 ～ 3を選択）
0x1：プロセッサDWTコンパレータ要素からの入力（0 ～ 3を選択）
0x2：ゼロのカウンタ（0を選択）
0x3：シングルショットコンパレータ（0を選択）
その他：予約済み

ビット15:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **SELECT[7:0]**：さらなるリソースの選択
GROUP[2:0] で選択したグループから1つ以上のリソースを選択します。

ETMシングルショットコンパレータ制御レジスタ0（ETM_SSCCR0）

アドレスオフセット：0x280

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット31:25 予約済みであり、リセット値に保持する必要があります。

ビット24 **RST**：シングルショットコンパレータのリセット
シングルショットコンパレータリソースの発生時にそのリセットを有効にし、別のコンパレータの一致を検出できるようにします。
1：リセットを有効化

ビット23:0 予約済みであり、リセット値に保持する必要があります。

ETMシングルショットコンパレータのステータスレジスタ0（ETM_SSCSR0）

アドレスオフセット：0x2A0

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC	DV	DA	INST
												r	r	r	r

ビット31 **STATUS**：シングルショットコンパレータのステータス

選択されたコンパレータのいずれかで一致が発生したかどうかを示します。

0：一致は発生しませんでした。

1：最低でも1回の一致が発生しました。

ビット30:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **PC**：プロセッサコンパレータ入力の感度

1：プロセッサコンパレータ入力に反応するシングルショットコンパレータ

ビット2 **DV**：データ値コンパレータのサポート

0：シングルショットデータ値の比較はサポートされません。

ビット1 **DA**：データアドレスコンパレータのサポート

0：シングルショットデータアドレスの比較はサポートされません。

ビット0 **INST**：命令アドレスコンパレータのサポート

0：シングルショット命令アドレスの比較はサポートされません。

ETMシングルショットプロセッサコンパレータの入力制御レジスタ0（ETM_SSPCICR0）

アドレスオフセット：0x2C0

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **PC[3:0]**：シングルショット制御のプロセッサコンパレータ入力の選択
0xXXX0：プロセッサコンパレータ入力0は選択されません。
0xXXX1：プロセッサコンパレータ入力0が選択されます。
0xXX0X：プロセッサコンパレータ入力1は選択されません。
0xXX1X：プロセッサコンパレータ入力1が選択されます。
0xX0XX：プロセッサコンパレータ入力2は選択されません。
0xX1XX：プロセッサコンパレータ入力2が選択されます。
0x0XXX：プロセッサコンパレータ入力3は選択されません。
0x1XXX：プロセッサコンパレータ入力3が選択されます。

ETMパワーダウン制御レジスタ（ETM_PDCR）

アドレスオフセット：0x310

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU	Res.	Res.	Res.
												rw			

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **PU**：パワーアップリクエスト
0：パワーアップはリクエストされません。
1：パワーアップがリクエストされます。

ビット2:0 予約済みであり、リセット値に保持する必要があります。

ETMパワーダウンステータスレジスタ（ETM_PDSR）

アドレスオフセット：0x314

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
														r	r

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **STICKYPD**：スティッキーなパワーダウンステート

0：トレースレジスタの電源は、ETM_PDSRが最後に読み出されたあとで切断されていません。

1：トレースレジスタの電源は、ETM_PDSRが最後に読み出されたあとで切断されています。

ビット0 **POWER**：ETMパワーアップステータス

1：ETMはパワーアップ状態です。

ETM要求タグ設定レジスタ（ETM_CLAIMSETR）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書込み：

0000：影響なし。

xxx1：ビット0をセットします。

xx1x：ビット1をセットします。

x1xx：ビット2をセットします。

1xxx：ビット3をセットします。

読出し：

0xF：要求タグには4つのビットがあることを示します。

ETM要求タグクリアレジスタ（ETM_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：

0000：影響なし。

xxx1：ビット0をクリアします。

xx1x：ビット1をクリアします。

x1xx：ビット2をクリアします。

1xxx：ビット3をクリアします。

読出し：要求タグの現在値を返します。

ETM認証ステータスレジスタ（ETM_AUTHSTATR）

アドレスオフセット：0xFB8

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x2：保護された非侵入型デバッグは無効です。

0x3：保護された非侵入型デバッグは有効です。

ビット5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル

0x2：無保護の非侵入型デバッグは無効です。

0x3：無保護の非侵入型デバッグは有効です。

ビット1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル

0x0：実装されません。

ETMデバイスタイプアーキテクチャレジスタ（ETM_DEVARCHR）

アドレスオフセット：0xFBC

リセット値：0x4772 4A13

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARCHITECT[10:0]											PRESENT	REVISION[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARCHVER[3:0]				ARCHPART[11:0]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:21 **ARCHITECT[10:0]**：アーキテクチャ JEP106コード
0x23B：JEP106継続コード0x4、JEP106 IDコード0x3B。Arm限定。

ビット20 **PRESENT**：DEVARCHレジスタの有無
0x1：あり

ビット19:16 **REVISION[3:0]**：アーキテクチャのリビジョン
0x2：ETMアーキテクチャ v4.2

ビット15:12 **ARCHVER[3:0]**：アーキテクチャのバージョン
0x4：ETMアーキテクチャ v4.2

ビット11:0 **ARCHPART[11:0]**：アーキテクチャパート
0xA13：ETMアーキテクチャ

ETM CoreSightデバイスタイプレジスタ（ETM_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0013

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUBTYPE[3:0]**：デバイスのサブタイプ識別子
0x1：プロセッサのトレース

ビット3:0 **MAJORTYPE[3:0]**：デバイスのメインタイプ識別子
0x3：トレースソース

ETM CoreSightペリフェラルIDレジスタ4（ETM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8　予約済みであり、リセット値に保持する必要があります。

ビット7:4　**SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0　**JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDECコード

ETM CoreSightペリフェラルIDレジスタ0（ETM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8　予約済みであり、リセット値に保持する必要があります。

ビット7:0　**PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：ETM部品番号

ETM CoreSightペリフェラルIDレジスタ1（ETM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]
0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]
0xD：ETM部品番号

ETM CoreSightペリフェラルIDレジスタ2（ETM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

ETM CoreSightペリフェラルIDレジスタ3（ETM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

ETM CoreSightコンポーネントIDレジスタ0（ETM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]
0x0D：共通の識別値

ETM CoreSightペリフェラルIDレジスタ1（ETM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス
0x9：トレースジェネレータのコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]
0x0：共通の識別値

ETM CoreSightコンポーネントIDレジスタ2（ETM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]
0x05：共通の識別値

ETM CoreSightコンポーネントIDレジスタ3（ETM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]
0xB1：共通の識別値

75.9.2 ETMレジスタマップ

ETMレジスタへはデバッガからアクセスでき、アドレスレンジは0xE0041000 ～ 0xE0041FFCです。

表795.ETMレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004	ETM_PRGCTLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																																0
0x008	予約済み	予約済み																															
0x00C	ETM_STATR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PMSTABLE	IDLE
	リセット値																															X	X
0x010	ETM_CONFIGR	Res.		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RS	Res.	COND[5:0]						CCI	BB		Res.	Res.
	リセット値																				X		X	X	X	X	X	X	X	X			
0x014 ~0x01C	予約済み	予約済み																															
0x020	ETM_ EVENTCTL0R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TYPE1	Res.	Res.	Res.	SEL1 [3:0]				TYPE0	Res.	Res.	Res.	SEL0 [3:0]			
	リセット値																	X				X	X	X	X	X				X	X	X	X
0x024	ETM_ EVENTCTL1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[1:0]	
	リセット値																				X	X										X	X
0x028	予約済み	予約済み																															
0x02C	ETM_STALLCTLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTPRIORITY	Res.	ISTALL	Res.	Res.	Res.	Res.	LEVEL[3:0]			
	リセット値																						X		X					X	X	X	X
0x030	予約済み	予約済み																															
0x034	ETM_SYNCPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
	リセット値																												0	1	0	1	0
0x038	ETM_CCCTLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	THRESHOLD[11:0]											
	リセット値																					X	X	X	X	X	X	X	X	X	X	X	X
0x03C	予約済み	予約済み																															
0x040	ETM_TRACEIDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACEID[6:0]						
	リセット値																										X	X	X	X	X	X	X
0x044 - 0x07C	予約済み	予約済み																															
0x080	ETM_VICTLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S [3:0]				Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSTATUS	Res.	EVENT[7:0]							
	リセット値													X	X	X	X					X	X	X		X	X	X	X	X	X	X	X
0x084 - 0x13C	予約済み	予約済み																															
0x140	ETM_CNTRLDVR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALUE[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x144から 0x17C	予約済み	予約済み																															

表795.ETMレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x180	ETM_IDR8	MAXSPEC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x184	ETM_IDR9	NUMP0KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x188	ETM_IDR10	NUMP1KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18C	ETM_IDR11	NUMP1SPC [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x190	ETM_IDR12	NUMCONDKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x194	ETM_IDR13	NUMCONDSPC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x198 - 0x1BC	予約済み	予約済み																															
0x1C0	ETM_IMSPECR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUPPORT [3:0]			
	リセット値																													0	0	0	0
0x1C4 ~0x1DC	予約済み	予約済み																															
0x1E0	ETM_IDR0	Res.	Res.	COMMOPT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRCEXDATA	QSUPP[1:0]		Res.	CONDTYPE [1:0]		NUMEVENT [1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	TRCDATA[1:0]		INSTP0 [1:0]		Res.
	リセット値			1												0	0	0		0	0	0	1	1		1	1	1	0	0	0	0	
0x1E4	ETM_IDR1	DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRCARCHMAJ [3:0]				TRCARCHMIN [3:0]				REVISION [3:0]			
	リセット値	0	1	0	0	0	0	0	1													0	1	0	0	0	0	1	0	0	0	0	1
0x1E8	ETM_IDR2	Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:0]					VMIDSIZE[4:0]					CIDSIZE[4:0]					IASIZE[4:0]			
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0x1EC	ETM_IDR3	NOOVERFLOW	NUMPROC[2:0]		SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S [3:0]			Res.	Res.	Res.	Res.	CCITMIN[11:0]															
	リセット値	0	0	0	0	1	1	1	1					1	0	0	1					0	0	0	0	0	0	0	0	0	1	0	0
0x1F0	ETM_IDR4	NUMVMIDC [3:0]				NUMCIDC [3:0]				NUMSSCC [3:0]				NUMRSPAIR [3:0]				NUMPC [3:0]				Res.	Res.	Res.	SUPPDAC	NUMDVC [3:0]				NUMACPAIRS [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0				0	0	0	0	0	0	0	0	0
0x1F4	ETM_IDR5	REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE [2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE [5:0]							Res.	Res.	Res.	Res.	NUMEXTINSEL [2:0]			NUMEXTIN[8:0]								
	リセット値	1	0	0	1	0	0	0		1	1	0	0	0	1	1	1					0	0	0	0	0	0	0	0	0	1	0	0
0x1F8 ~0x204	予約済み	予約済み																															
0x208	ETM_RSCTLR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値											X	X		X	X	X									X	X	X	X	X	X	X	X
0x20C	ETM_RSCTLR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INV	Res.	GROUP [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値												X		X	X	X									X	X	X	X	X	X	X	X

表795.ETMレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x210 - 0x27C	予約済み	予約済み																															
0x280	ETM_SSCCR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								X																								
0x284 - 0x29C	予約済み	予約済み																															
0x2A0	ETM_SSCSR0	STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC	DV	DA	INST
	リセット値	X																												X	X	X	X
0x2A4 ~0x2BC	予約済み	予約済み																															
0x2C0	ETM_SSPCICR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[3:0]			
	リセット値																													X	X	X	X
0x2C4 ~0x30C	予約済み	予約済み																															
0x310	ETM_PDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU	Res.	Res.	Res.
	リセット値																													0			
0x314	ETM_PDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
	リセット値																															1	1
0x318 - 0xF9C	予約済み	予約済み																															
0xFA0	ETM_CLAIMSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1
0xFA4	ETM_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFA8 - 0xFB4	予約済み	予約済み																															
0xFB8	ETM_AUTHSTATR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									X	X	X	X	X	X	X	X
0xFBC	ETM_DEVARCHR	ARCHITECT[10:0]											PRESENT	REVISION [3:0]				ARCHVER [3:0]				ARCHPART[11:0]											
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	1	0	0	1	0	0	1	0	1	0	0	0	0	1	0	0	1	1
0xFC0 - 0xFC8	予約済み	予約済み																															
0xFCC	ETM_DEVTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYP [3:0]			
	リセット値																									0	0	0	1	0	0	1	1
0xFD0	ETM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 - 0xFDC	予約済み	予約済み																															
0xFE0	ETM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	1	0	0	0	0	1
0xFE4	ETM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	1	0	1

表795.ETMレジスタのマップとリセット値 （続き）

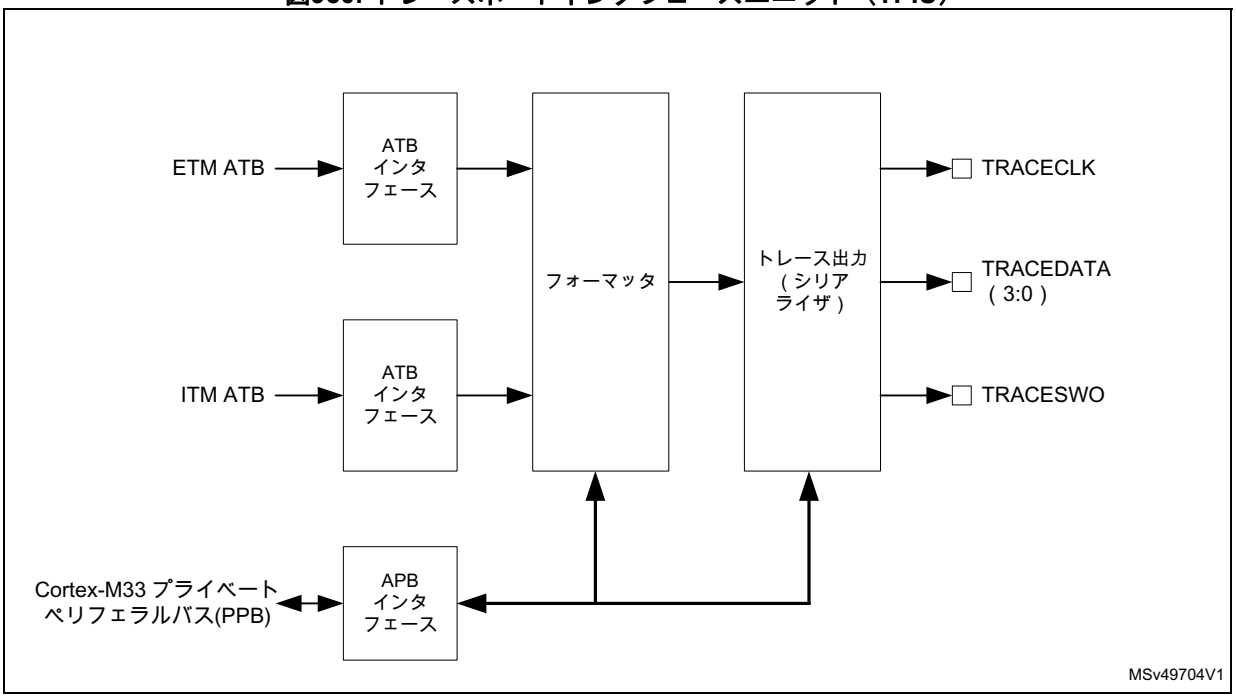
オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE8	ETM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]					JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	ETM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	ETM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	ETM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	ETM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	ETM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブルを参照してください。

75.10 トレースポートインタフェースユニット（TPIU）

TPIUはトレースストリームをフォーマットし、それを外部トレースポート信号に出力します。下図に示すように、TPIUにはETMとITMの受信トレースデータ用に2つのATBスレーブポートが搭載されています。トレースポートは同期パラレルポートで、クロック出力TRACECLKと4つのデータ出力TRACEDATA(3:0) から成ります。トレースポートの幅はレンジ1～4でプログラム可能です。より幅の狭いポート設定を使用すると、必要なテスト点／コネクタピンの数が減り、他の目的のために出力ポートを解放することができます。しかし、それによりトレースポートの帯域幅が制限され、リアルタイムで出力できるトレース情報量が減少します。

図950. トレースポートインタフェースユニット（TPIU）



トレースデータはシリアルワイヤ出力TRACESWOに出力することもできます。

Cortex-M33のトレースポートインタフェースの詳細については、Arm Cortex-M33 Technical Reference Manual[4]を参照してください。

75.10.1 TPIUレジスタ

TPIUでサポートされるポートサイズのレジスタ（TPIU_SSPSR）

アドレスオフセット：0x000

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PORTSIZE[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PORTSIZE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **PORTSIZE[31:0]**：1 ～ 32ピンのトレースポートのサイズ

ビットn-1がセットされている場合、ポートサイズnがサポートされることを示します。

0x0000 000F：ポートサイズ1～4がサポートされます。

TPIUの現在のポートサイズのレジスタ（TPIU_CSPSR）

アドレスオフセット：0x004

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PORTSIZE[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PORTSIZE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **PORTSIZE[31:0]**：現在のトレースポートサイズ

ビットn-1がセットされている場合、現在のポートサイズがnピンであることを示します。nの値はサポートされるポートサイズレンジ内（1～4）でなければなりません。1bitのみをセットしてください。そうしないと、予期せぬ動作を引き起こす場合があります。

このレジスタの変更は必ずフォーマッタの停止時に行ってください。

TPIU非同期クロックプリスケーラレジスタ（TPIU_ACPR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PRESCALER[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **PRESCALER[12:0]**：非同期出力TRACESWOのボーレート

ボーレートは（PRESCALER +1）で割ったTRACELKIN周波数で与えられます。

TPIU選択ピンプロトコルレジスタ（TPIU_SPPR）

アドレスオフセット：0x0F0

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXMODE[1:0]	
														rw	rw

ビット31:2 予約済みであり、リセット値に保持する必要があります。

ビット1:0 **TXMODE[1:0]**：トレース出力に使用されるプロトコル

0x0：パラレルトレースポートモード

0x1：マンチェスタ符号化を使用した非同期SWO

0x2：NRZ符号化を使用した非同期SWO

0x3：予約済み

TPIUフォーマッタおよび一掃のステータスレジスタ（TPIU_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTNONSTOP	TCPRESENT	FTSTOPPED	FLINPROG
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3 **FTNONSTOP**：フォーマッタ停止

フォーマッタを停止できるかどうかを示します。

1：フォーマッタは停止できません。

ビット2 **TCPRESENT**：TRACECTL出力ピンの使用可能性

オプションのTRACECTL出力ピンを使用可能かどうかを示します。

0：TRACECTLピンはこのデバイスには存在しません。

ビット1 **FTSTOPPED**：フォーマッタ停止

フォーマッタは停止リクエスト信号を受信しており、すべてのトレースデータおよびポストアンブルが送信されます。ATBインタフェースでの追加のトレースデータは無視されます。

0：フォーマッタは停止していません。

ビット0 **FLINPROG**：一掃の実行中

ATBスレーブポートで一掃を実行中かどうかを示します。このビットは、AFVALIDS出力の状態を反映します。一掃はTPIU_FFCRレジスタの一掃制御ビットによって開始することができます。

0：実行中の一掃はありません。

1：一掃の実行中です。

TPIUフォーマッタおよび一掃の制御レジスタ（TPIU_FFCR）

アドレスオフセット：0x304

リセット値：0x0000 0102

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGIN	Res.	FONMAN	Res.	Res.	Res.	Res.	ENFCONT	Res.
							r		rw					rw	

ビット31:9 予約済みであり、リセット値に保持する必要があります。

ビット8 **TRIGIN**：トリガ入力時のトリガ

1：TRIGIN入力がアサートされたときにトレースストリームにあるトリガを示します。

ビット7 予約済みであり、リセット値に保持する必要があります。

ビット6 **FONMAN**：マニュアル一掃

0：一掃が完了しました。

1：一掃を生成します。

ビット5:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **ENFCONT**：連続フォーマットの有効化

SWOモードでこのビットを0にセットすることでフォーマッタはバイパスされ、ITM/DWTトレースのみが出力されて、ETMトレースは破棄されます。

0：連続フォーマットは無効です。

1：連続フォーマットは有効です。

ビット0 予約済みであり、リセット値に保持する必要があります。

TPIU周期的同期カウンタのレジスタ（TPIU_PSCR）

アドレスオフセット：0x308

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PSCOUNT[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:13 予約済みであり、リセット値に保持する必要があります。

ビット12:0 **PSCOUNT[12:0]**：フォーマッタフレームカウンタ

キャプチャデバイスの大容量のストレージを無駄にすることなく、サイズの異なる複数のTPAを効果的に使用できるようにします。このカウンタには、最後の128bit同期パケット以降のフォーマッタフレームの数が格納されます。12bitのカウンタで、最大カウント値は4096です。この値は65536バイトごとの同期、すなわち4096パケットxパケット当たり16バイトに等しい数です。デフォルトは1024バイトごと、すなわち64フォーマッタフレームごとの同期パケットに対して設定されます。フォーマッタが連続モード用に設定されている場合は、フルワードとハーフワードの同期フレームが通常動作中に挿入されます。このような状況では、カウント値はフル同期パケット間の完了フレームの最大数となります。

TPIU要求タグ設定レジスタ（TPIU_CLAIMSETR）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書込み：

0000：影響なし。

xxx1：ビット0をセットします。

xx1x：ビット1をセットします。

x1xx：ビット2をセットします。

1xxx：ビット3をセットします。

読出し：

0xF：要求タグには4つのビットがあることを示します。

TPIU要求タグクリアレジスタ（TPIU_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：

0000：影響なし。

xxx1：ビット0をクリアします。

xx1x：ビット1をクリアします。

x1xx：ビット2をクリアします。

1xxx：ビット3をクリアします。

読出し：要求タグの現在値を返します。

TPIUデバイス設定レジスタ（TPIU_DEVIDR）

アドレスオフセット：0xFC8

リセット値：0x0000 0CA1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MAXNUM[4:0]				
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:12 予約済みであり、リセット値に保持する必要があります。

ビット11 **SWOUARTNRZ**：シリアルワイヤ出力、NRZのサポート

0x1：サポートされます。

ビット10 **SWOMAN**：シリアルワイヤ出力、マンチェスタ符号化フォーマットのサポート

0x1：サポートされます。

ビット9 **TCLKDATA**：トレースクロックおよびデータのサポート

0x0：サポートされます。

ビット8:6 **FIFOSIZE[2:0]**：FIFOのサイズ（2の累乗値）

0x2：FIFOのサイズ = 4バイト

ビット5 **CLKRELAT**：ATBクロックとTRACECLKINの関係（同期または非同期）

0x1：非同期

ビット4:0 **MAXNUM[4:0]**：ATB入力ポートマルチプレクスの数／タイプ

0x1：2つの入力ポート

TPIUデバイスタイプ識別子レジスタ（TPIU_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0011

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUBTYPE[3:0]**：下位分類

0x1：トレースポートコンポーネント

ビット3:0 **MAJORTYPE[3:0]**：大分類

0x1：トレースシンクコンポーネント

TPIU CoreSightペリフェラルIDレジスタ4（TPIU_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDECコード

TPIU CoreSightペリフェラルIDレジスタ0（TPIU_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：TPIU部品番号

TPIU CoreSightペリフェラルIDレジスタ1（TPIU_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]

0xD：TPIU部品番号

TPIU CoreSightペリフェラルIDレジスタ2（TPIU_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

TPIU CoreSightペリフェラルIDレジスタ3（TPIU_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

TPIU CoreSightコンポーネントIDレジスタ0（TPIU_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

TPIU CoreSightペリフェラルIDレジスタ1（TPIU_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]**：コンポーネントIDビット [15:12] - コンポーネントクラス

0x9：CoreSightコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

TPIU CoreSightコンポーネントIDレジスタ2（TPIU_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

TPIU CoreSightコンポーネントIDレジスタ3（TPIU_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.10.2 TPIUレジスタマップ

表796.TPIUレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	TPIU_SSPSR	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0x004	TPIU_CSPSR	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x008	予約済み	予約済み																															
0x010	TPIU_ACPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x014から0x0EC	予約済み	予約済み																															
0x0F0	TPIU_SPPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXMODE[1:0]	
	リセット値																															0	1
0x0F4から0x2FC	予約済み	予約済み																															
0x300	TPIU_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTNONSTOP	TCPRESENT	FTSTOPPED	FLINPROG
	リセット値																													1	0	0	0
0x304	TPIU_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGIN	Res.	FONMAN	Res.	Res.	Res.	Res.	ENFCONT	Res.
	リセット値																								1		0					1	
0x308	TPIU_PSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSCOUNT[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
030Cから0xF9C	予約済み	予約済み																															
0xFA0	TPIU_CLAIMSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1

表796.TPIUレジスタのマップとリセット値 （続き）

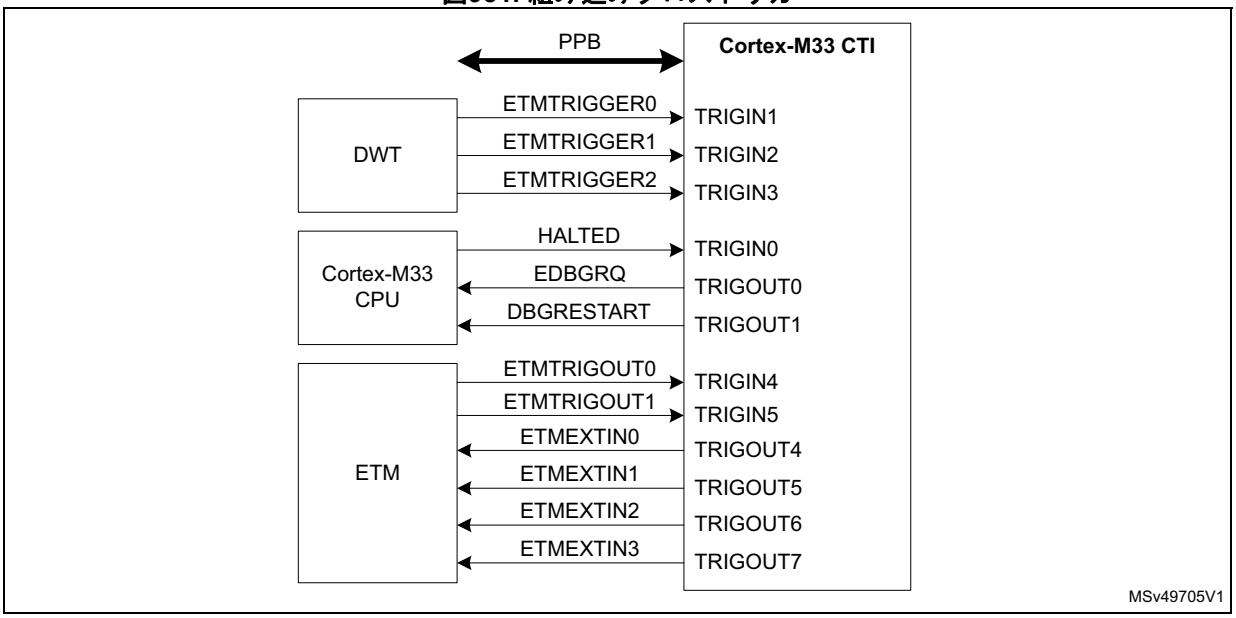
オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFA4	TPIU_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0FA8から0xFC4	予約済み	予約済み																															
0xFC8	TPIU_DEVIDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MaXNUM[4:0]				
	リセット値																					1	1	0	0	1	0	1	0	0	0	0	1
0xFCC	TPIU_DEVTYPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYPE [3:0]			
	リセット値																									0	0	0	1	0	0	0	1
0xFD0	TPIU_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFE0	TPIU_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	1	0	0	0	0	1
0xFE4	TPIU_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	1	0	1
0xFE8	TPIU_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]					JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	TPIU_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	TPIU_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	TPIU_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	TPIU_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	TPIU_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 788：MCU ROMテーブルを参照してください。

75.11 クロストリガインタフェース（CTI）

CTIは、プロセッサとETMの間のクロストリガを可能にします（下図を参照）。

図951. 組み込みクロストリガ



CTIは、さまざまな転送元からのイベントによるデバッグ／トレース動作、またはその両方のトリガを有効にします。たとえば、プロセッサで到達したウォッチポイントがコードトレースを開始または停止するか、トレースコンパレータがプロセッサを停止することができます。

CTIのトリガ入力および出力信号は下図に記載されています。

表797.CTI入力

番号	転送元信号	転送元コンポーネント	コメント
0	HALTED	CPU	プロセッサ停止 - CPUはデバッグモードです。
1	ETMTRIGGER0	DWT	DWTコンパレータ出力0
2	ETMTRIGGER1	DWT	DWTコンパレータ出力1
3	ETMTRIGGER2	DWT	DWTコンパレータ出力2
4	ETMTRIGOUT0	ETM	ETMイベント出力0
5	ETMTRIGOUT1	ETM	ETMイベント出力1
6	-	-	未使用
7	-	-	未使用

表798.CTI出力

番号	転送元信号	転送先コンポーネント	コメント
0	EDBGRQ	CPU	CPU停止リクエスト - CPUをデバッグモードにします。
1	DBGRESTART	CPU	CPU再開リクエスト - CPUはデバッグモードを終了します。
2	-	-	未使用
3	-	-	未使用
4	ETMEXTIN0	ETM	ETMイベント入力0
5	ETMEXTIN1	ETM	ETMイベント入力1
6	ETMEXTIN2	ETM	ETMイベント入力2
7	ETMEXTIN3	ETM	ETMイベント入力3

クロストリガインタフェースCoreSightコンポーネントの詳細情報は、Arm CoreSight SoC-400 Technical Reference Manualを参照してください[2]。

75.11.1 CTIレジスタ

CTIのレジスタファイルのベースアドレスは0xE004 2000です。

CTI制御レジスタ（CTI_CONTROLR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLBEN
															rw

ビット31:1 予約済みであり、リセット値に保持する必要があります。

ビット0 **GLBEN：グローバルCTI有効化**

0：無効

1：有効

CTIトリガ確認応答レジスタ（CTI_INTACKR）

アドレスオフセット：0x010

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTACK[7:0]							
								w	w	w	w	w	w	w	w

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **INTACK[7:0]**：トリガ確認応答

CTITRIGOUT出力ごとに レジスタの1bitが割り当てられます。このレジスタの1bitに1が書き込まれると、対応するCTITRIGOUT出力が確認応答され、クリアされます。

CTIアプリケーショントリガ設定レジスタ（CTI_APPSETR）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPSET[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **APPSET[3:0]**：チャネルイベント設定

読出し：

XXX0：チャネル0イベントはインアクティブです。

XXX0：チャネル0イベントはアクティブです。

XX0X：チャネル1イベントはインアクティブです。

XX1X：チャネル1イベントはアクティブです。

X0XX：チャネル2イベントはインアクティブです。

X1XX：チャネル2イベントはアクティブです。

0XXX：チャネル3イベントはインアクティブです。

1XXX：チャネル3イベントはアクティブです。

書込み：

XXX0：影響はありません。

XXX0：チャネル0にイベントを設定します。

XX0X：影響はありません。

XX1X：チャネル1にイベントを設定します。

X0XX：影響はありません。

X1XX：チャネル2にイベントを設定します。

0XXX：影響はありません。

1XXX：チャネル3にイベントを設定します。

CTIアプリケーショントリガクリアレジスタ（CTI_APPCLEAR）

アドレスオフセット：0x018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **APPCLEAR[3:0]**：チャネルイベントのクリア

0000：影響なし。

XXX1：チャネル0のイベントをクリアします。

XX1X：チャネル1のイベントをクリアします。

X1XX：チャネル2のイベントをクリアします。

1XXX：チャネル3のイベントをクリアします。

CTIアプリケーションパルスレジスタ（CTI_APPPULSER）

アドレスオフセット：0x01C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE[3:0]			
												w	w	w	w

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **APPPULSE[3:0]**：チャネルイベントのパルス出力

このレジスタ自体がただちにクリアされます。

0000：影響なし。

XXX1：チャネル0のパルスを生成します。

XX1X：チャネル1のパルスを生成します。

X1XX：チャネル2のパルスを生成します。

1XXX：チャネル3のパルスを生成します。

CTIトリガ入力x有効レジスタ（CTI_INENxR）

アドレスオフセット：0x020 + 0x004 * x（x = 0から7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN[3:0]			
												rw	rw	rw	rw

ビット31:4　予約済みであり、リセット値に保持する必要があります。

ビット3:0　**TRIGINEN[3:0]**：トリガ入力イベントの有効化

CTITRIGINxがアクティブ化された時点で（x = 0 ～ 7）、4つのチャネルそれぞれのクロストリガイベントを有効または無効にします。

0000：トリガはチャネルのイベントを生成しません。

XXX1：トリガxはチャネル0のイベントを生成します。

XX1X：トリガxはチャネル1のイベントを生成します。

X1XX：トリガxはチャネル2のイベントを生成します。

1XXX：トリガxはチャネル3のイベントを生成します。

CTIトリガ出力x有効レジスタ（CTI_OUTENxR）

アドレスオフセット：0x0A0 + 0x4 * x（x = 0から7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN[3:0]			
												rw	rw	rw	rw

ビット31:4　予約済みであり、リセット値に保持する必要があります。

ビット3:0　**TRIGOUTEN[3:0]**：トリガ出力イベントの有効化

チャネルごとに、そのチャネルのイベントがCTITRIGOUTx （x = 0～7）のトリガを生成するかどうかを定義します。

0000：チャネルイベントはトリガ出力のトリガを生成しません。

XXX1：チャネル0イベントはトリガ出力xのトリガを生成します。

XX1X：チャネル1イベントはトリガ出力xのトリガを生成します。

X1XX：チャネル2イベントはトリガ出力xのトリガを生成します。

1XXX：チャネル3イベントはトリガ出力xのトリガを生成します。

CTIトリガ入力ステータスレジスタ（CTI_TRGISTSR）

アドレスオフセット：0x130

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **TRIGINSTATUS[7:0]**：トリガ入力ステータス

CTITRIGINx入力ごとにレジスタの1bitが割り当てられます。このビットが1にセットされている場合は、対応するトリガ入力がアクティブであることを示します。このビットが0に設定されている場合は、対応するトリガ入力はインアクティブです。

CTIトリガ出力ステータスレジスタ（CTI_TRGOSTSR）

アドレスオフセット：0x134

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **TRIGOUTSTATUS[7:0]**：トリガ出力ステータス

CTITRIGOUT出力ごとに レジスタの1bitが割り当てられます。このビットが1にセットされている場合は、対応するトリガ出力がアクティブであることを示します。このビットが0に設定されている場合は、対応するトリガ出力はインアクティブです。

CTIチャネル入力ステータスレジスタ（CTI_CHINSTSR）

アドレスオフセット：0x138

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHINSTATUS[3:0]			
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CHINSTATUS[3:0]**：チャネル入力ステータス

チャネル入力ごとに レジスタの1bitが割り当てられます。このビットが1にセットされている場合は、対応するチャネル入力がアクティブであることを示します。このビットが0に設定されている場合は、対応するチャネル入力はインアクティブです。

CTIチャネル出力ステータスレジスタ（CTI_CHOUTSTSR）

アドレスオフセット：0x13C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHOUTSTATUS[3:0]			
												r	r	r	r

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **CHOUTSTATUS[3:0]**：チャネル出力ステータス

チャネル出力ごとに レジスタの1bitが割り当てられます。このビットが1にセットされている場合は、対応するチャネル出力がアクティブであることを示します。このビットが0に設定されている場合は、対応するチャネル出力はインアクティブです。

CTIチャネルゲートレジスタ（CTI_GATER）

アドレスオフセット：0x140

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GATEEN[3:0]			
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **GATEEN[3:0]**：チャネル出力の有効化

チャネルごとに、そのチャネルのイベントがCTMを通って 他のCTIへ伝わり得るかどうかを定義します。

0000：チャネルイベントは伝わりません。

XXX1：チャネル0イベントは伝わります。

XX1X：チャネル1イベントは伝わります。

X1XX：チャネル2イベントは伝わります。

1XXX：チャネル3イベントは伝わります。

CTIデバイス設定レジスタ（CTI_DEVIDR）

アドレスオフセット：0xFC8

リセット値：0x0004 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMCH[3:0]			
												r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMTRIG[7:0]								Res.	Res.	Res.	EXTMUXNUM[4:0]				
r	r	r	r	r	r	r	r				r	r	r	r	r

ビット31:20 予約済みであり、リセット値に保持する必要があります。

ビット19:16 **NUMCH[3:0]**：使用できるECTチャネルの数

0x4：4つのチャネル

ビット15:8 **NUMTRIG[7:0]**：使用できるECTトリガの数

0x8：8つのトリガ入力と8つのトリガ出力

ビット7:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **EXTMUXNUM[4:0]**：トリガ入力／出力マルチプレクサの数

0x0：なし

CTIデバイスタイプ識別子レジスタ（CTI_DEVTYPER）

アドレスオフセット：0xFCC

リセット値：0x0000 0014

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SUBTYPE[3:0]**：下位分類

0x1：クロストリガコンポーネント

ビット3:0 **MAJORTYPE[3:0]**：大分類

0x4 : このコンポーネントによりデバッガがCoreSight SoC-400システムの他のコンポーネントを制御できることを示します。

CTI CoreSightペリフェラルIDレジスタ4（CTI_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x4：Arm JEDECコード

CTI CoreSightペリフェラルIDレジスタ0（CTI_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x21：CTI部品番号

CTI CoreSightペリフェラルIDレジスタ1（CTI_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00BD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0xB：Arm JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]

0xD：CTI部品番号

CTI CoreSightペリフェラルIDレジスタ2（CTI_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値
0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]
0x3：Arm JEDECコード

CTI CoreSightペリフェラルIDレジスタ3（CTI_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CTI CoreSightコンポーネントIDレジスタ0（CTI_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

CTI CoreSightペリフェラルIDレジスタ1（CTI_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス

0x9：CoreSightコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

CTI CoreSightコンポーネントIDレジスタ2（CTI_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

CTI CoreSightコンポーネントIDレジスタ3（CTI_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.11.2 CTIレジスタマップ

表799.CTIレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	CTI_CONTROLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLBEN
	リセット値																																0
0x004：0x00C	予約済みです。	予約済みです。																															
0x010	CTI_INTACKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTACK[7:0]							
	リセット値																									X	X	X	X	X	X	X	X
0x014	CTI_APPSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPSET[3:0]			
	リセット値																													0	0	0	0
0x018	CTI_APPCLEAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR [3:0]			
	リセット値																													0	0	0	0
0x01C	CTI_APPPULSER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE [3:0]			
	リセット値																													X	X	X	X
0x020から0x03C	CTI_INEN0Rから CTI_INEN7R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x040 - 0x09C	予約済み	予約済み																															
0x0A0から0x0BC	CTI_OUTEN0Rから CTI_OUTEN7R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0C0 - 0x12C	予約済み	予約済み																															
0x130	CTI_TRGISTSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x134	CTI_TRGOSTSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表799.CTIレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x138	CTI_CHINSTSR	Res.																												CHINSTATUS[3:0]			
	リセット値																													0	0	0	0
0x13C	CTI_CHOUTSTSR	Res.																												CHOUTSTATUS[3:0]			
	リセット値																													0	0	0	0
0x140	CTI_GATER	Res.																												GATEEN[3:0]			
	リセット値																													1	1	1	1
0x144 - 0xFC4	予約済み	予約済み																															
0xFC8	CTI_DEVIDR	Res.												NUMCH[3:0]				NUMTRIG[7:0]								Res.			EXTMUXNUM[4:0]				
	リセット値													0	1	0	0	0	0	0	0	1	0	0	0				0	0	0	0	0
0xFCC	CTI_DEVTYPER	Res.												Res.												SUB[3:0]				MAJOR[3:0]			
	リセット値																									0	0	0	1	0	1	0	0
0xFD0	CTI_PIDR4	Res.												Res.												SIZE[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 - 0xFDC	予約済み	予約済み																															
0xFE0	CTI_PIDR0	Res.												Res.												PARTNUM[7:0]							
	リセット値																									0	0	0	1	0	0	0	1
0xFE4	CTI_PIDR1	Res.												Res.												JEP106ID[3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	1	0	1
0xFE8	CTI_PIDR2	Res.												Res.												REVISION[3:0]					JEP106ID[6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	CTI_PIDR3	Res.												Res.												REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CTI_CIDR0	Res.												Res.												PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	CTI_CIDR1	Res.												Res.												CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	CTI_CIDR2	Res.												Res.												PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CTI_CIDR3	Res.												Res.												PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについては表 789：プロセッサROMテーブル を参照してください。

75.12 マイクロコントローラデバッグユニット（DBGMCU）

DBGMCUは、デバッグモードで電源やクロックの動作を制御するレジスタが複数内蔵されているコンポーネントです。デバッガ（またはソフトウェア）の以下の動作を可能にします。

- 低電力モード（SLEEP、STOP、STANDBY）時に、プロセッサコアへのクロックや電源の供給を維持します。
- 低電力モード時に、システムデバッグおよびトレース用のコンポーネントへのクロックや電源の供給を維持します。
- 一方のプロセッサコアがデバッグモードで停止した場合、特定のペリフェラルへのクロック（SMBUSタイムアウト、ウォッチドッグ、タイマ、RTC）の供給を停止します。

75.12.1 デバイスID

DBGMCUには、IDコードレジスタDBGMCU_IDCODEが含まれます。このレジスタには、デバイス用のIDコードが格納されます。デバッグツールでは、表75.5で説明するCoreSight検出手順によりこのレジスタの位置を特定できます。

75.12.2 低電力モードエミュレーション

デバイスがSTOPモード（クロックが停止）またはSTANDBYモード（コア電源がオフ）に移行すると、デバッガではデバッグアクセスポートにアクセスできなくなり、デバイスとの接続が切れます。これを避けるため、デバッガ（またはソフトウェア）によりDBGMCU_CRのDBG_STANDBYやDBG_STOPをセットすることができます。これらのビットをセットすると、プロセッサへのクロックや電力の供給を維持しながら、デバイスを対応する低電力モードで動作させることができます。プロセッサはSLEEPモードのまま、正常に低電力モードを終了します。ただし、ペリフェラルデバイスは動作を継続するため、デバイスの動作は実際の低電力モードの動作とは異なることがあります。

75.12.3 ペリフェラルクロックフリーズ

DBGMCUペリフェラルクロックフリーズレジスタを使用すると、特定のペリフェラルの動作をデバッグモードでサスペンドすることができます。この機能に対応しているペリフェラルユニットを、次の表に示します。

表800.ペリフェラルクロックフリーズ制御ビット

バス	制御レジスタ	ペリフェラル	説明
APB1L	DBGMCU_APB1LFZR	I2C2	I2C2 SMBUSタイムアウト
		I2C1	I2C1 SMBUSタイムアウト
		IWDG	独立型ウォッチドッグ
		WWDG	ウィンドウ型ウォッチドッグ
		TIM7	汎用タイマ7
		TIM6	汎用タイマ6
		TIM5	汎用タイマ5
		TIM4	汎用タイマ4
		TIM3	汎用タイマ3
		TIM2	汎用タイマ2

表800.ペリフェラルクロックフリーズ制御ビット （続き）

バス	制御レジスタ	ペリフェラル	説明
APB1H	DGBMCU_APB1HFZR	LPTIM2	低電力タイマ2
		I2C4	I2C4 SMBUSタイムアウト
		I2C5	I2C5 SMBUSタイムアウト
		I2C6	I2C6 SMBUSタイムアウト
APB2	DBGMCU_APB2FZR	TIM17	汎用タイマ17
		TIM16	汎用タイマ16
		TIM15	汎用タイマ15
		TIM8	汎用タイマ8
		TIM1	汎用タイマ1
APB3	DBGMCU_APB3FZR	RTC	リアルタイムクロック
		LPTIM4	低電力タイマ4
		LPTIM3	低電力タイマ3
		LPTIM1	低電力タイマ1
		I2C3	I2C3 SMBUSタイムアウト
AHB1	DBGMCU_AHB1FZR	GPDMA 0 ~ 15	汎用DMAチャネル0 ~ 15
AHB3	DBGMCU_AHB3FZR	GPDMA 0 ~ 3	低電力DMAチャネル0 ~ 3

各ペリフェラルユニットまたはDMAチャネルには対応する制御ビットDBG_xxx_STOPがあります。ここで、xxxはペリフェラル（またはDMAチャネル）の略称です。制御ビットはDBGMCU_zzzFZRレジスタで構成されます。ここで、zzzはバスの名前に対応します（AHBまたはAPB）。たとえば、DBGMCU_APB1LFZRにはAPB1Lバスのペリフェラルの制御ビットが格納されます。

制御ビットをセットすると、以下の表に従って、デバッグでCPUが停止したときに（HALTED = 1）、対応するペリフェラルの動作をサスペンドできます。

表801.デバッグモードでのペリフェラルの動作

HALTED	DBG_xxx_STOP	ペリフェラルの動作
0	X	動作が継続します。
1	0	動作が継続します。
1	1	動作がサスペンドします。

デバッガからDBG_xxx_STOPビットへのアクセス可能性は、認証信号spidenの状態によって変わります。

spiden = 1の場合（セキュアな特権デバッグが有効）、セキュアアクセスによりすべてのビットが変更可能です。非セキュアアクセスでは、非セキュアペリフェラル（またはDMAチャネル）に対応するビットのみを変更できます。ビットの読出しは、セキュアアクセスでも非セキュアアクセスでもできます。

spiden = 0の場合（セキュアな特権デバッグが無効）、非セキュアアクセスのみが可能です（デバッガによるセキュアアクセスリクエストはCPUによる非セキュアアクセスリクエストに変換されます）。

非セキュアペリフェラル（またはDMAチャネル）に対応するビットのみが変更できます。すべてのビットの読出しもできます。これについて、次の表にまとめます。

表802.フリーズレジスタビットへのデバッガのアクセス

spiden	ペリフェラルxxxステータス	アクセスのセキュリティ属性	DBG_xxx_STOPの変更の可否	DBG_xxx_STOPの読出しの可否
0	非セキュア	非セキュア	あり	あり
		セキュア	あり(1)	あり(1)
	セキュア	非セキュア	なし	あり
		セキュア	不可(1)	あり(1)
1	非セキュア	非セキュア	あり	あり
		セキュア	あり	あり
	セキュア	非セキュア	なし	あり
		セキュア	あり	あり

1. spiden = 0の場合、デバッガによるセキュアアクセスリクエストは非セキュアアクセスリクエストに変換されます。

TrustZone対応ペリフェラルまたはDMAチャネルのステータス（セキュアまたは非セキュア）は、ペリフェラルによりDBGMCUに通知されます。

DBG_xxx_STOPビットへのCPUのアクセスは、spidenに関係なく行えます。このアクセスに関係するのは、ペリフェラルのセキュリティステータスのみです。セキュアペリフェラル（またはDMAチャネル）に対応するビットは、セキュアアクセスによってのみ変更できます（CPUがセキュア状態の場合）。

75.12.4 DBGMCUレジスタ

DBGMCUレジスタは、システムリセットではリセットされず、パワーオンリセットによってのみリセットされます。それらはAHBアクセスポート経由でデバッガおよびソフトウェアのベースアドレス0xE004 4000にアクセス可能です。

DBGMCU IDコードレジスタ（DBGMCU_IDCODE）

アドレスオフセット：0x00

リセット値：0xXXXX 6XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REV_ID[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DEV_ID[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **REV_ID[15:0]**：リビジョン

STM32U5Fx/5Gxの場合
0x1000：リビジョンA
0x1001：リビジョンZ

STM32U59x/5Axの場合
0x3001：リビジョンX

STM32U575/585の場合
0x2001：リビジョンX
0x3001：リビジョンW

STM32U535/545の場合
0x1001：リビジョンZ

ビット15:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **DEV_ID[11:0]**：デバイス識別

0x455：STM32U535/545
0x476：STM32U5Fx/5Gx
0x481：STM32U59x/5Ax
0x482：STM32U575/585

DBGMCU設定レジスタ（DBGMCU_CR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE [1:0]		TRACE _EN	TRACE _IOEN	Res.	DBG_S TANDB Y	DBG_S TOP	Res.
								rw	rw	rw	rw		rw	rw	

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:6 **TRACE_MODE[1:0]**：TRACEピンの割当て

0x0：非同期モードに割り当てられたTRACEピン（TRACESWO）

0x1：ポート幅が1の同期モードに割り当てられたTRACEピン（TRACECK、TRACED0）

0x2：ポート幅が2の同期モードに割り当てられたTRACEピン（TRACECK、TRACED0-1）

0X3：ポート幅が4の同期モードに割り当てられたTRACEピン（TRACECK、TRACED0-3）

ビット5 **TRACE_EN**：トレースポートおよびクロックの有効化

このビットは、トレースポートクロックTRACECKを有効にします。

0：無効

1：有効

ビット4 **TRACE_IOEN**：TRACEピンの有効化

0：無効。TRACEピンは割り当てられません。

1：有効。TRACEピンはTRACE_MODEフィールドの値に応じて割り当てられます。

ビット3 予約済みであり、リセット値に保持する必要があります。

ビット2 **DBG_STANDBY**：STANDBYモードでデバッグ許可

0：通常動作

STANDBYモードでは、すべてのクロックが無効化され、コアの電源は自動的に切断されます。

1：自動クロック停止／パワーダウンは無効。

すべてのアクティブなクロックおよびオシレータはSTANDBYモード中も動作し続け、コア供給は維持され、すべてのデバッグ機能を許可します。STANDBYモード を終了すると、システムリセットが行われます。

ビット1 **DBG_STOP**：STOPモードでデバッグ許可

0：通常動作

STOPモードでは、すべてのクロックが自動的に無効化されます。

1：自動クロック停止は無効。

すべてのアクティブなクロックおよびオシレータはSTOPモード中も動作し続け、すべてのデバッグ機能を許可します。STOPモードから抜けると、クロック設定はSTOPモード終了ステートに設定されます。

ビット0 予約済みであり、リセット値に保持する必要があります。

DBGMCU APB1Lペリフェラルフリーズレジスタ（DBGMCU_APB1LFZR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.
									rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	Res.	Res.	Res.	Res.	Res.	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
			rw	rw						rw	rw	rw	rw	rw	rw

ビット31:23 予約済みであり、リセット値に保持する必要があります。

ビット22 **DBG_I2C2_STOP**：デバッグ時のI2C2 SMBUSタイムアウトの停止

0：通常動作。I2C2 SMBUSタイムアウトは、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C2 SMBUSタイムアウトは、CPUがデバッグモードの間は停止します。

ビット21 **DBG_I2C1_STOP**：デバッグ時のI2C1 SMBUSタイムアウトの停止

0：通常動作。I2C1 SMBUSタイムアウトは、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C1 SMBUSタイムアウトは、CPUがデバッグモードの間は停止します。

ビット20:13 予約済みであり、リセット値に保持する必要があります。

ビット12 **DBG_IWDG_STOP**：デバッグ時のIWDGの停止

0：通常動作。IWDG は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。IWDGは、CPUがデバッグモードの間は停止します。

ビット11 **DBG_WWDG_STOP**デバッグ時のWWDGの停止

0：通常動作。WWDG は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。WWDGは、CPUがデバッグモードの間は停止します。

ビット10:6 予約済みであり、リセット値に保持する必要があります。

ビット5:0 **DBG_TIMy_STOP**：デバッグ時のTIMyの停止（Y = 7 ~ 2）

0：通常動作。TIMy は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。TIMyは、CPUがデバッグモードの間は停止します。

DBGMCU APB1Hペリフェラルフリーズレジスタ（DBGMCU_APB1HFZR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C6_STOP	DBG_I2C5_STOP	DBG_LPTIM2_STOP	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
								rw	rw	rw				rw	

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7 **DBG_I2C6_STOP**：デバッグ時のI2C6の停止

0：通常動作。I2C6は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C6は、CPUがデバッグモードの間は停止します。

注： このビットはSTM32U535/545/575/585デバイスでは予約済みです。

ビット6 **DBG_I2C5_STOP**：デバッグ時のI2C5の停止

0：通常動作。I2C5は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C5は、CPUがデバッグモードの間は停止します。

注： このビットはSTM32U535/545/575/585デバイスでは予約済みです。

ビット5 **DBG_LPTIM2_STOP**：デバッグ時のLPTIM2の停止

0：通常動作。LPTIM2は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。LPTIM2は、CPUがデバッグモードの間は停止します。

ビット4:2 予約済みであり、リセット値に保持する必要があります。

ビット1 **DBG_I2C4_STOP**：デバッグ時のI2C4の停止

0：通常動作。I2C4は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C4は、CPUがデバッグモードの間は停止します。

ビット0 予約済みであり、リセット値に保持する必要があります。

DBGMCU APB2ペリフェラルフリーズレジスタ（DBGMCU_APB2FZR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		rw		rw											

ビット31:19 予約済みであり、リセット値に保持する必要があります。

ビット18:16 **DBG_TIMy_STOP**：デバッグ時のTIMyの停止（Y = 17 ~ 15）

0：通常動作。TIMy は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。TIMyは、CPUがデバッグモードの間は停止します。

ビット15:14 予約済みであり、リセット値に保持する必要があります。

ビット13 **DBG_TIM8_STOP**：デバッグ時のTIM8の停止

0：通常動作。TIM8は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。TIM8は、CPUがデバッグモードの間は停止します。

ビット12 予約済みであり、リセット値に保持する必要があります。

ビット11 **DBG_TIM1_STOP**：デバッグ時のTIM1の停止

0：通常動作。TIM1は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。TIM1は、CPUがデバッグモードの間は停止します。

ビット10:0 予約済みであり、リセット値に保持する必要があります。

DBGMCU APB3ペリフェラルフリーズレジスタ（DBGMCU_APB3FZR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DBG_RTC_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPTIM4_STOP	DBG_LPTIM3_STOP	DBG_LPTIM1_STOP	Res.
	rw											rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DBG_I2C3_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw										

ビット31 予約済みであり、リセット値に保持する必要があります。

ビット30 **DBG_RTC_STOP**：デバッグ時のRTC停止

0：通常動作。RTCは、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。RTCは、CPUがデバッグモードの間は停止します。

ビット29:20 予約済みであり、リセット値に保持する必要があります。

ビット19 **DBG_LPTIM4_STOP**：デバッグ時のLPTIM4の停止

0：通常動作。LPTIM4は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。LPTIM4は、CPUがデバッグモードの間は停止します。

ビット18 **DBG_LPTIM3_STOP**：デバッグ時のLPTIM3の停止

0：通常動作。LPTIM3は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。LPTIM3は、CPUがデバッグモードの間は停止します。

ビット17 **DBG_LPTIM1_STOP**：デバッグ時のLPTIM1の停止

0：通常動作。LPTIM1は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。LPTIM1は、CPUがデバッグモードの間は停止します。

ビット16:11 予約済みであり、リセット値に保持する必要があります。

ビット10 **DBG_I2C3_STOP**：デバッグ時のI2C3の停止

0：通常動作。I2C3は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。I2C3は、CPUがデバッグモードの間は停止します。

ビット9:0 予約済みであり、リセット値に保持する必要があります。

DBGMCU AHB1ペリフェラルフリーズレジスタ（DBGMCU_AHB1FZR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBG_GPDMA15_STOP	DBG_GPDMA14_STOP	DBG_GPDMA13_STOP	DBG_GPDMA12_STOP	DBG_GPDMA11_STOP	DBG_GPDMA10_STOP	DBG_GPDMA9_STOP	DBG_GPDMA8_STOP	DBG_GPDMA7_STOP	DBG_GPDMA6_STOP	DBG_GPDMA5_STOP	DBG_GPDMA4_STOP	DBG_GPDMA3_STOP	DBG_GPDMA2_STOP	DBG_GPDMA1_STOP	DBG_GPDMA0_STOP
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット31:16 予約済みであり、リセット値に保持する必要があります。

ビット15:0 **DBG_GPDMAy_STOP**：デバッグ時のGPDMAチャネルyの停止（y = 15 ~ 0）

0：通常動作。GPDMAチャネルyは、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。GPDMAチャネルyは、CPUがデバッグモードの間は停止します。

DBGMCU AHB3ペリフェラルフリーズレジスタ（DBGMCU_AHB3FZR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPDMA3_STOP	DBG_LPDMA2_STOP	DBG_LPDMA1_STOP	DBG_LPDMA0_STOP
												rw	rw	rw	rw

ビット31:4 予約済みであり、リセット値に保持する必要があります。

ビット3:0 **DBG_LPDMAy_STOP**：デバッグ時のLPDMAチャネルyの停止（y = 3 ~ 0）

0：通常動作。LPDMAチャネル3は、CPUがデバッグモードの間は動作し続けます。

1：デバッグ時の停止。LPDMAチャネル3は、CPUがデバッグモードの間は停止します。

DBGMCUステータスレジスタ（DBGMCU_SR）

アドレスオフセット：0xFC

リセット値：0xXXXX 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AP_ENABLED[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AP_PRESENT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:16 **AP_ENABLED[15:0]**：ビットnは、アクセスポートAP nがオープンしている（デバッグポートからアクセス可能）かロックされている（APへのデバッグアクセスがブロックされる）かを示します。

ビットn = 0：APnはロックされています。

ビットn = 1：APnは有効です。

ビット15:0 **AP_PRESENT[15:0]**：ビットnは、アクセスポートAP nがデバイスにあるかどうかを示します。

ビットn = 0：APnは存在しません。

ビットn = 1：APnは存在します。

DBGMCUデバッグホスト認証レジスタ（DBGMCU_DBG_AUTH_HOST）

アドレスオフセット：0x100

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AUTH_KEY[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUTH_KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット31:0 **AUTH_KEY[31:0]**：デバイス認証キー

デバイス固有の64bitの認証キー（OEMキー）をこのレジスタに書き込み（連続する2つの32bitの書込みで、最下位ワードが最初）、RDPの回帰を許可する必要があります。間違ったキーに書き込むと、デバイスへのアクセスがロックされ、Flashメモリからのコード実行は行われません。

DBGMCUデバッグデバイス認証レジスタ（DBGMCU_DBG_AUTH_DEVICE）

アドレスオフセット：0x104

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AUTH_ID[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUTH_ID[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **AUTH_ID[31:0]**：デバイス固有ID

RDP回帰のために使用されるデバイス固有ID。

DBGMCU CoreSightペリフェラルIDレジスタ4（DBGMCU_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **SIZE[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは4KBの領域1つ分を占有します。

ビット3:0 **JEP106CON[3:0]**：JEP106継続コード

0x0：STMicroelectronics JEDECコード

DBGMCU CoreSightペリフェラルIDレジスタ0（DBGMCU_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PARTNUM[7:0]**：部品番号のビット [7:0]

0x00：DBGMCU部品番号

DBGMCU CoreSightペリフェラルIDレジスタ1（DBGMCU_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **JEP106ID[3:0]**：JEP106 IDコードビット [3:0]

0x0：STMicroelectronics JEDECコード

ビット3:0 **PARTNUM[11:8]**：部品番号のビット [11:8]

0x0：DBGMCU部品番号

DBGMCU CoreSightペリフェラルIDレジスタ2（DBGMCU_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号

0x0：r0p0

ビット3 **JEDEC**：JEDEC割当て値

0x1：JEDECによって指定される設計者識別

ビット2:0 **JEP106ID[6:4]**：JEP106 IDコードビット [6:4]

0x2：STMicroelectronics JEDECコード

DBGMCU CoreSightペリフェラルIDレジスタ3（DBGMCU_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **REVAND[3:0]**：メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

DBGMCU CoreSightコンポーネントIDレジスタ0（DBGMCU_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[7:0]**：コンポーネント識別ビット [7:0]

0x0D：共通の識別値

DBGMCU CoreSightコンポーネントIDレジスタ1（DBGMCU_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00F0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:4 **CLASS[3:0]：コンポーネント識別**ビット [15:12] - コンポーネントクラス

0xF：非CoreSightコンポーネント

ビット3:0 **PREAMBLE[11:8]**：コンポーネント識別ビット [11:8]

0x0：共通の識別値

DBGMCU CoreSightコンポーネントIDレジスタ2（DBGMCU_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[19:12]**：コンポーネント識別ビット [23:16]

0x05：共通の識別値

DBGMCU CoreSightコンポーネントIDレジスタ3（DBGMCU_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット31:8 予約済みであり、リセット値に保持する必要があります。

ビット7:0 **PREAMBLE[27:20]**：コンポーネント識別ビット [31:24]

0xB1：共通の識別値

75.12.5 DBGMCUレジスタマップ

表803.DBGMCUレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DBGMCU_IDCODE	REV_ID[15:0]																Res.	Res.	Res.	Res.	DEV_ID[11:0]											
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x004	DBGMCU_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE [1:0]		TRACE_EN	TRACE_IOEN	Res.	DBG_STANDBY	DBG_STOP	Res.
	リセット値																									0	0	0	0		0	0	
0x008	DBGMCU_ APB1LFZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	Res.	Res.	Res.	Res.	Res.	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
	リセット値										0	0									0	0					0	0	0	0	0	0	0
0x0C0	DBGMCU_ APB1HFZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C6_STOP	DBG_I2C5_STOP	DBG_LPTIM2_STOP	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
	リセット値																									0	0	0				0	
0x010	DBGMCU_ APB2FZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP	Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0			0		0											

表803.DBGMCUレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x04	DBGMCU_APB3FZR	Res.	DBG_RTC_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPTIM4_STOP	DBG_LPTIM3_STOP	DBG_LPTIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C3_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値		0											0	0	0							0										
0x018から0x01C	予約済み	予約済み																															
0x020	DBGMCU_AHB1FZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_GPDMA15_STOP	DBG_GPDMA14_STOP	DBG_GPDMA13_STOP	DBG_GPDMA12_STOP	DBG_GPDMA11_STOP	DBG_GPDMA10_STOP	DBG_GPDMA9_STOP	DBG_GPDMA8_STOP	DBG_GPDMA7_STOP	DBG_GPDMA6_STOP	DBG_GPDMA5_STOP	DBG_GPDMA4_STOP	DBG_GPDMA3_STOP	DBG_GPDMA2_STOP	DBG_GPDMA1_STOP	DBG_GPDMA0_STOP
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	予約済み	予約済み																															
0x028	DBGMCU_AHB3FZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPDMA3_STOP	DBG_LPDMA2_STOP	DBG_LPDMA1_STOP	DBG_LPDMA0_STOP
	リセット値																													0	0	0	0
0x02C - 0x0F8	予約済み	予約済み																															
0x0FC	DBGMCU_SR	AP_ENABLED[15:0]																AP_PRESENT[15:0]															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x100	DBGMCU_DBG_AUTH_HOST	AUTH_KEY[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x104	DBGMCU_DBG_AUTH_DEVICE	AUTH_ID[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x108 - 0xFBC	予約済み	予約済み																															
0xFD0	DBGMCU_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SIZE[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD4 - 0xFDC	予約済み	予約済み																															
0xFE0	DBGMCU_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0xFE4	DBGMCU_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
	リセット値																									0	0	0	0	0	0	0	0
0xFE8	DBGMCU_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]					JEP106ID[6:4]		
	リセット値																									0	0	0	0	1	0	1	0
0xFEC	DBGMCU_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	DBGMCU_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1

表803.DBGMCUレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF4	DBGMCU_CIDR1	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	1	1	1	0	0	0	0
0xFF8	DBGMCU_CIDR2	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	DBGMCU_CIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

レジスタ境界アドレスについてはセクション2.3を参照してください。

75.13 参照

1. IHI 0031C (ID080813) - Arm Debug Interface Architecture Specification ADIv5.0 ~ ADIv5.2、C版、2013年8月8日
2. DDI 0480F（ID100313）- Arm CoreSight SoC-400 r3p2 Technical Reference Manual、G版, 2015年3月16日DDI 0314H - Arm CoreSight Components Technical Reference Manual、H版、2009年7月10日
3. DDI 0553A (ID092917) - Arm v8-M Architecture Reference Manual、A.f版、2017年9月29日
4. 100230_0002_00_en - Arm Cortex-M33 Processor r0p2 Technical Reference Manual、0002-00版、2017年5月10日
5. 100232_0001_00_en - Arm CoreSight ETM-M33 r0p1 Technical Reference Manual、0001-00版、2017年2月3日

76 デバイス電子署名

デバイス電子署名は、Flashメモリモジュールのシステムメモリ領域に格納され、デバッグインタフェースまたはCPUを使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データおよび較正データを使用すれば、ユーザファームウェアやその他の外部デバイスは、デバイスの特性に自動的に整合させることができます。

76.1 ユニークデバイスIDレジスタ（96bit）

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号（例：USB文字列シリアル番号やその他のエンドアプリケーション）として使用
- 内部Flashメモリをプログラムする前に、このユニークIDをソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、Flashメモリ内のコードのセキュリティを高めるためのセキュリティキーの一部として使用
- セキュアなブートプロセスの起動に使用

96bitのユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

ベースアドレス：0x0BFA 0700

アドレスオフセット：0x00

読出し専用 = 0xXXXX XXXX（Xは出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **UID[31:0]**：ウェハ上のXおよびY座標

アドレスオフセット：0x04

読出し専用 = 0xXXXX XXXX（Xは出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[63:48]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[47:32]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:8 **UID[63:40]**：LOT_NUM[23:0]
ロット番号（ASCIIで符号化）

ビット7:0 **UID[39:32]**：WAF_NUM[7:0]
ウェハー番号（8bitの符号なし番号）

アドレスオフセット：0x08

読出し専用 = 0xXXXX XXXX（Xは出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[95:80]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[79:64]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット31:0 **UID[95:64]**：LOT_NUM[55:24]
ロット番号（ASCIIで符号化）

76.2 Flashサイズデータレジスタ

ベースアドレス：0x0BFA 07A0

アドレスオフセット：0x00

読出し専用 = 0xXXXX（Xは出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Flash_SIZE															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット15:0 **Flash_SIZE[15:0]**：Flashメモリサイズ
このフィールドは、デバイスのFlashメモリサイズをKB単位で示します。
たとえば、0x800は2048 KBに対応します。

76.3 パッケージデータレジスタ

ベースアドレス：0x0BFA 0500

アドレスオフセット：0x00

読出し専用 = 0xXXXX（Xは出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKG[4:0]				
											r	r	r	r	r

ビット15:5 予約済みであり、リセット値に保持する必要があります。

ビット4:0 **PKG[4:0]**：パッケージタイプ

00000：LQFP64
00001：WLCSP72 SMPS
00010：LQFP100
00011：UFBGA132
00100：LQFP144
00101：LQFP48
00111：UFBGA169またはTFBGA169
01000：LQFP64 SMPS
01001：WLSCP90 SMPS
01010：LQFP100 SMPS
01011：UFBGA132 SMPS
01100：LQFP144 SMPS
01101：LQFP48 SMPS
01111：UFBGA169 SMPSまたはTFBGA169 SMPS
10010：UFBGA64
10011：UFBGA100
10100：LQFP100 DSI SMPS
10101：LQFP144 DSI SMPS
11001：UFBGA144 DSI SMPS
11011：WLCSP208 DSI SMPS
11100：TFBGA216 DSI SMPS
11101：UFBGA100 SMPS
11110：WLCSP56 SMPS
11111：WLCSP150 SMPSまたはWLCSP150 DSI SMPS

注： デバイスで利用できるパッケージについては、製品のデータシートを参照してください。

77 セキュリティ上の重要通知

STマイクロエレクトロニクス（ST）では製品のセキュリティを重視しているため、本書に示すSTの製品は各種セキュリティ認証機関の認証を受け、ここに記載する独自のセキュリティ対策を実施しています。ただし、どのセキュリティ認証も内部のセキュリティ対策も、あらゆる形態の攻撃に対するST製品の耐性を保証するものではありません。そのため、ST製品を単体で使用する場合も、他のコンポーネントやソフトウェアと組み合わせてお客様の最終製品やアプリケーションを構築する場合も、ST製品が提供するセキュリティのレベルがお客様のニーズを満たしているか、お客様の責任でご判断ください。特に、次の点に注意してください。

- ST製品は、Platform Security Architecture（www.psacertified.org）、Security Evaluation standard for IoT Platforms（www.trustcb.com）をはじめとする1つまたは複数のセキュリティ認証機関の認証を受けている場合があります。ここに示すST製品がこのような認証のレベルや現在のステータスと一緒にセキュリティ認証を受けているかどうかについてお知りになりたい場合は、関連する認証基準のウェブサイトまたは関連する製品ページ（www.st.com）にアクセスして、最新情報をご確認ください。ST製品のセキュリティ認証のステータスやレベルは変更されることがあるため、必要に応じてセキュリティ認証のステータス/レベルをその都度ご確認ください。特定のセキュリティ基準への適合性が証明されない場合、認証を受けているとみなさないでください。
- 認証機関はST製品に関して、セキュリティ認証を評価し、付与し、取り消す権利を有しています。そのため、ST製品に対するセキュリティ認証の付与または取り消しに関する責任はこのような認証機関のみにあり、STはST製品に関して認証機関で生じた誤り、実施した評価、査定、テスト、その他の活動についていかなる責任も負いません。
- ST製品と一緒に使用されることのある産業ベースの暗号化アルゴリズム（AES、DES、MD5など）やその他のオープンスタンダードの技術は、STが規定したもの以外の基準に基づいています。STは、このような暗号化アルゴリズムやオープンテクノロジの不具合について、またこのようなアルゴリズムやテクノロジの迂回、復号、解読のために開発された、または開発される可能性のあるいかなる手法についても、責任を負いません。
- 堅固なセキュリティテストは実施されていると考えられますが、いかなるレベルの認証でも、まだテストがされていない高度な攻撃、新しく未定義の攻撃の形態、仕様や想定された使用法に従わずにST製品を使用した場合や、最終製品またはアプリケーションを構築するためにお客様が他のコンポーネントやソフトウェアと一緒に使用した場合の攻撃の形態など、あらゆる攻撃に対する保護について絶対的な保証をすることはできません。STはこのような攻撃に対する耐性について、責任を負いません。そのため、組み込まれたセキュリティ機能、STが提供する情報やサポートに関係なく、ST製品を単体で使用する場合も、お客様の最終製品またはアプリケーションに組み込んで使用する場合も、テストする攻撃のレベルが実際のニーズに適合しているかどうかを判断するのは、お客様の責任です。
- ST製品のセキュリティ機能（ハードウェア、ソフトウェア、ドキュメントなど）は、STによりセキュリティが強化されたものも含めてすべて、「現状有姿」で提供されます。そのため、明示的または黙示的なあらゆる保証（商品性、特定目的への適合性の黙示的な保証を含みますがこれらに限定されません）は、書面による署名付きの契約条件で別に定められている場合を除いて、適用される法令により認められる範囲内で免責されるものとします。

78 改版履歴

表804.文書改版履歴

日付	版	変更内容
2021年6月22日	1	初版発行
2021年9月20日	2	更新： – 図1：システムアーキテクチャ – 図3：STM32U575/585用のIDAUマッピングに基づくメモリマップ – セクション6.3.2：エラーコードの修正（SRAM2、SRAM3、BKPSRAM）の最後 – セクション7.6.2：読出し保護（RDP）の導入部に文を追加 – セクション7.9.14：FLASHオプションレジスタ（FLASH_OPTR）のビット21の説明 – セクション10.5.3：LDOおよびSMPSステップダウンコンバータの高速起動の最初の文 – STOP 0モードの終了、STOP 2モードの終了、およびSTOP 3モードの終了から1つの文を削除 – セクション10.10.9：PWRバックアップドメイン制御レジスタ1（PWR_BDCR1）のBRENビットに注を追加 – セクション 11.8.50：RCC 制御／ステータスレジスタ（RCC_CSR）の MSISSRANGE と MSIKSRANGEの説明 – 表125：ペリフェラル相互接続マトリックス – 表124：プログラムされたGPDMA1リクエスト – セクション25.7.1：OCTOSPI制御レジスタ（OCTOSPI_CR）のビット7 – セクション37.2：PSSIの主な機能 – 表212：RNGの設定 – セクション34.4.13：AESデータレジスタとデータスワッピング – セクション42.4.14：共有キーを使用したSAESの操作 – セクション23.6.19:TIMxオプションレジスタ1(TIMx_OR1)(x = 16 ~ 17)のアドレスオフセット – 表121：LPTIM1/2/3入出力ピン、表122：LPTIM4入出力ピン、および表146：LPTIM1/2/3/4外部トリガ接続 – 表533：I2C1、I2C2、I2C4の相互接続および表534：I2C3の相互接続 – 表294：アナログフィルタとデジタルフィルタの比較 – 表553：USARTの相互接続（USART1/2/3およびUART4/5）のusart/lpuart_trg6/7 – セクション31.5.20：USARTおよびDMAを使用した連続通信の新しい注 – 新規 マイクロコントローラの低消費電力モードからの正常なウェイクアップを可能にする最大USARTボーレートの決定 – 表322：LPUART入出力ピン – 表565：LPUARTの相互接続（LPUART1）のusart/lpuart_trg6/7 – NEW – 表576：SPIの相互接続（SPI1およびSPI2）および 表577：SPIの相互接続（SPI3） – 新規 I/Oの制御 – セクション65.3：シリアルワイヤとJTAGデバッグポート（SWJ-DP）の構成 – DPアクセスポート選択レジスタ（DP_SELECTR）のAPSEL範囲 – DBGMCU識別コードレジスタ（DBGMCU_IDCODE）のREV_ID[15:0] – DPターゲット識別レジスタ（DP_TARGETIDR）のTREVISION[3:0]

表804.文書改版履歴 （続き）

日付	版	変更内容
2022年3月16日	3	更新： – 関連ドキュメント – セクション2.1.9：SmartRunドメイン（SRD）の新しい注 – セクション3.1：主なセキュリティ機能 – セクション3.2：セキュアインストール – セクション3.3.2：システムFlashメモリにおける不変の信頼の起点 – セクション3.4：セキュア更新 – 表10：セキュアI/Oに接続されていない非セキュアペリフェラル機能 – セクション3.5.7：TrustZoneセキュリティの無効化 – セクション3.8.1：ハードウェアのセキュアな鍵管理の新しい注 – セクション3.11：アクセス制御されたデバッグ – セクション5.6.4：GTZC1　TZSCセキュア設定レジスタ3（GTZC1_TZSC_SECCFGR3）のDCMISECの説明 – セクション5.6.7：GTZC1 TZSC特権設定レジスタ3（GTZC1_TZSC_PRIVCFGR3）のDCMIPRIVの説明 – セクション5.7.3：GTZC1 TZIC割込みイネーブルレジスタ3（GTZC1_TZIC_IER3）のDCMIIEの説明 – セクション5.7.7：GTZC1 TZICステータスレジスタ3（GTZC1_TZIC_SR3）のDCMIFの説明 – セクション5.7.11：GTZC1 TZICフラグクリアレジスタ3（GTZC1_TZIC_FCR3）のCDCMIFの説明 – セクション6.3.2：エラーコードの修正（SRAM2、SRAM3、BKPSRAM）の新しい警告 – セクション7.3.1：Flashメモリ構成内の文 – セクション7.9.13：FLASHオプションレジスタ（FLASH_OPTR）からセクション7.9.24：FLASH WPR2領域Bアドレスレジスタ（FLASH_WRP2BR）の新しいST出荷時の値 – 図29：電源の概要 – セクション10.4.1：外部電源のV_{BAT} – セクション10.4.7：バッテリバックアップドメイン – 表89：動作モードに応じた機能にPSSIを追加 – セクション10.10.1：PWR制御レジスタ1（PWR_CR1）の ULPMENの説明 – セクション10.6.1：ブラウンアウトリセット（BOR） – セクション10.6.2：プログラム可能な電圧検出器（PVD） – セクション10.7.8：STOP 2モードおよびセクション10.7.9：STOP 3モード – 表99：PWR割込みリクエストの注 – セクション11.4.6：PLL – セクション11.4.7：LSEクロック – LSEによるハードウェア自動較正（PLLモード） – セクション11.4.12：LSEのクロックセキュリティシステム – セクション11.4.18 :クロック信号出力 – セクション11.5.1：RCC TrustZoneセキュリティ保護モード – 表107：割込みソースと制御のLSECSS、MSI_PLL_UNLOCK、および注 – セクション11.8.2：RCC内部クロックソース較正レジスタ1（RCC_ICSCR1）のMSIBIASの説明 – セクション11.8.12：RCC PLL1分周回路レジスタ（RCC_PLL1DIVR）のPLL1P/Rの説明 – セクション11.8.49：RCCバックアップドメイン制御レジスタ（RCC_BDCR）のLSESYSENの説明 – 表153：STM32U575/585ベクタテーブルのLSECSS、MSI_PLL_UNLOCK

表804.文書改版履歴 （続き）

日付	版	変更内容
2022年3月16日	3（続き）	– 表156：EXTIラインの接続のライン23 – セクション21.6.1 ~ セクション21.6.7、セクション21.6.10、および21.6.11 のビット23 – セクション18.6.8：EXTI外部割込み選択レジスタ（EXTI_EXTICRm） – バルブサンプリングモード – ADC_CRのCALINDEX[3:0] – 新規 拡張較正モード – 表249：温度センサ較正値のメモリ位置 – セクション21.7.13：ADC電源レジスタ（ADC_PWRR） – 新規 セクション55.3：TAMPの実装 – アクティブタンパの検出 – セクション50.6.5：TAMPアクティブタンパ制御レジスタ1（TAMP_ATCR1）のATCKSELの説明 – 表514：STM32U575/585 IWDG機能の注 – ウィンドウオプションが有効な場合のIWDGの設定 – 新規 ウィンドウコンパレータの更新 – セクション48.4.6：レジスタのアクセス保護 – セクション48.7.1：IWDGキーレジスタ （IWDG_KR）のKEY[15:0] の説明 – 表653：UCPDの実装および表654：UCPDソフトウェアトリムデータ – 新規 セクション64.5.5：UCPDソフトウェアトリミング – セクション64.8.3：UCPD設定レジスタ3（UCPD_CFGR3） – DPターゲット識別レジスタ（DP_TARGETIDR）のTREVISIONの説明 – DBGMCU識別コードレジスタ（DBGMCU_IDCODE）のREV_IDの説明
2023年2月16日	4	STM32U5シリーズの全製品に対応するよう、全範囲を更新 – STM32U535/545 – STM32U575/585 rev X以上 – STM32U59x/5Ax, – STM32U5Fx/5Gx 更新： – セクション5.6.9：GTZC1 TZSCメモリxサブ領域Aウォーターマークレジスタ（GTZC1_TZSC_MPCWMxAR）およびセクション5.6.10：GTZC1 TZSCメモリxサブ領域Bウォーターマークレジスタ（GTZC1_TZSC_MPCWMxBR）のリセット値 – SRAM3 ECC固有の管理 – セクション7.3.8：Flashメモリの書換え耐性 – セクション10：電源制御（PWR） – セクション11.4.6：PLL – セクション11.8.12：RCC PLL1分周回路レジスタ（RCC_PLL1DIVR） – セクション11.8.30：RCC AHB2ペリフェラルクロック有効化レジスタ1（RCC_AHB2ENR1）およびセクション11.8.31：RCC AHB2ペリフェラルクロック有効化レジスタ2（RCC_AHB2ENR2）のリセット値 – 表682：LPUARTの相互接続（LPUART1） – セクション33：A/Dコンバータ（ADC12） – DPターゲット識別レジスタ（DP_TARGETIDR） – DBGMCU IDコードレジスタ（DBGMCU_IDCODE） – DBGMCUステータスレジスタ（DBGMCU_SR）
2023年10月19日	5	カバーページにSTM32U5FxxxおよびSTM32U5Gxxxドキュメントの記載を追加

索引

A

ADC_AWD1TR 1409
ADC_AWD2CR 1338, 1415
ADC_AWD2TR 1409
ADC_AWD3CR 1339, 1415
ADC_AWD3TR 1413
ADC_CALFACT 1343, 1416
ADC_CALFACT2 1344
ADC_CCR 1417
ADC_CFGR1 . . . . . 1322, 1403
ADC_CFGR2 . . . . . 1326, 1406
ADC_CHSELR 1410, 1411
ADC_CR 1319, 1401
ADC_DIFSEL 1342
ADC_DR 1334, 1413
ADC_GCOMP 1337
ADC_HTR1 1340
ADC_HTR2 1341
ADC_HTR3 1342
ADC_IER 1317, 1399
ADC_ISR 1315, 1397
ADC_JDRy 1338
ADC_JSQR 1335
ADC_LTR1 1339
ADC_LTR2 1340
ADC_LTR3 1341
ADC_OFRy 1336
ADC_OR 1416
ADC_PCSEL 1330
ADC_PWRR 1414
ADC_SMPR 1408
ADC_SMPR1 1329
ADC_SMPR2 1329
ADC_SQR1 1331
ADC_SQR2 1332
ADC_SQR3 1333
ADC_SQR4 1334
ADC12_CCR 1347
ADC12_CDR 1350
ADC12_CDR2 1351
ADC12_CSR 1345
ADF_BSMX0CR 1638
ADF_CKGCR 1634
ADF_DFLT0CICR 1640
ADF_DFLT0CR 1638
ADF_DFLT0DR 1650
ADF_DFLT0IER 1643
ADF_DFLT0ISR 1644
ADF_DFLT0RSFR 1642
ADF_DLY0CR 1643
ADF_GCR 1633
ADF_SADANLVR 1649
ADF_SADCFGR 1648
ADF_SADCR 1646
ADF_SADSDLVR 1649
ADF_SITF0CR 1636
AES_CR 1938
AES_DINR 1942
AES_DOUTR 1943
AES_ICR 1950
AES_IER 1948
AES_ISR 1949
AES_IVR0 1945
AES_IVR1 1945
AES_IVR2 1946
AES_IVR3 1946
AES_KEYR0 1943
AES_KEYR1 1944
AES_KEYR2 1944
AES_KEYR3 1944
AES_KEYR4 1946
AES_KEYR5 1947
AES_KEYR6 1947
AES_KEYR7 1947
AES_SR 1940
AES_SUSPxR 1948
AP_BASER 3478
AP_BDnR 3477
AP_CFGR 3477
AP_CSWR 3475
AP_DRWR 3476
AP_IDR 3478
AP_TAR 3476

B

BPU_CIDR0 3526
BPU_CIDR1 3527
BPU_CIDR2 3527
BPU_CIDR3 3528
BPU_COMPxR 3522
BPU_CTRLR 3522
BPU_DEVARCHR 3523
BPU_DEVTYPER 3523
BPU_PIDR0 3524
BPU_PIDR1 3525
BPU_PIDR2 3525

BPU_PIDR3 3526
BPU_PIDR4 3524

C

C1ROM_CIDR3 . . . . . .3487, 3493
COMP1_CSR 1471
COMP2_CSR 1473
CORDIC_CSR 947
CORDIC_RDATA 949
CORDIC_WDATA 949
CPUROM_CIDR0 3491
CPUROM_CIDR1 3491
CPUROM_CIDR2 3492
CPUROM_CIDR3 3492
CPUROM_MEMTYPER 3488
CPUROM_PIDR0 3489
CPUROM_PIDR1 3489
CPUROM_PIDR2 3490
CPUROM_PIDR3 3490
CPUROM_PIDR4 3488
CRC_CR 930
CRC_DR 929
CRC_IDR 929
CRC_INIT 931
CRC_POL 931
CRS_CFGR 612
CRS_CR 611
CRS_ICR 615
CRS_ISR 613
CTI_APPCLEAR 3574
CTI_APPPULSER 3574
CTI_APPSETR 3573
CTI_CHINSTSR 3577
CTI_CHOUTSTSR 3577
CTI_CIDR0 3582
CTI_CIDR1 3582
CTI_CIDR2 3582
CTI_CIDR3 3583
CTI_CONTROLR 3572
CTI_DEVIDR 3578
CTI_DEVTYPER 3579
CTI_GATER 3578
CTI_INENxR 3575
CTI_INTACKR 3573
CTI_OUTENxR 3575
CTI_PIDR0 3580
CTI_PIDR1 3580
CTI_PIDR2 3581
CTI_PIDR3 3581
CTI_PIDR4 3579
CTI_TRGISTSR 3576
CTI_TRGOSTSR 3576

D

DAC_AUTOCR 1457
DAC_CCR 1453
DAC_CR 1442
DAC_DHR12L1 1446
DAC_DHR12L2 1448
DAC_DHR12LD 1449
DAC_DHR12R1 1446
DAC_DHR12R2 1447
DAC_DHR12RD 1449
DAC_DHR8R1 1447
DAC_DHR8R2 1448
DAC_DHR8RD 1450
DAC_DOR1 1450
DAC_DOR2 1451
DAC_MCR 1453
DAC_SHHR 1456
DAC_SHRR 1456
DAC_SHSR1 1455
DAC_SHSR2 1455
DAC_SR 1451
DAC_SWTRGR 1445
DBGMCU_AHB1FZR 3593
DBGMCU_AHB3FZR 3594
DBGMCU_APB1HFZR 3591
DBGMCU_APB1LFZR 3590
DBGMCU_APB2FZR 3592
DBGMCU_APB3FZR 3592
DBGMCU_CIDR0 3598
DBGMCU_CIDR1 3598
DBGMCU_CIDR2 3598
DBGMCU_CIDR3 3599
DBGMCU_CR 3589
DBGMCU_DBG_AUTH_DEVICE 3595
DBGMCU_DBG_AUTH_HOST 3595
DBGMCU_IDCODE 3588
DBGMCU_PIDR0 3596
DBGMCU_PIDR1 3596
DBGMCU_PIDR2 3597
DBGMCU_PIDR3 3597
DBGMCU_PIDR4 3595
DBGMCU_SR 3594
DCACHE_CMDREADDRR 393
DCACHE_CMDRSADDRR 392
DCACHE_CR 387
DCACHE_FCR 390
DCACHE_IER 390
DCACHE_RHMONR 391
DCACHE_RMMONR 391

DCACHE_SR 389
DCACHE_WHMONR 392
DCACHE_WMMONR 392
DCMI_CR 1664
DCMI_CWSIZE 1674
DCMI_CWSTRT 1674
DCMI_DR 1675
DCMI_ESCR 1672
DCMI_ESUR 1673
DCMI_ICR 1671
DCMI_IER 1669
DCMI_MIS 1670
DCMI_RIS 1668
DCMI_SR 1667
DLYB_CFGR 1242
DLYB_CR 1241
DMA2D_AMTCR 844, 879
DMA2D_BGCLUTx 845, 880
DMA2D_BGCMAR 838, 874
DMA2D_BGCOLR 837, 873
DMA2D_BGMAR 832, 868
DMA2D_BGOR 832, 868
DMA2D_BGPFCCR 835, 871
DMA2D_CR 828, 864
DMA2D_FGCLUTx 844, 880
DMA2D_FGCMAR 838, 873
DMA2D_FGCOLR 834, 871
DMA2D_FGMAR 831, 867
DMA2D_FGOR 831, 867
DMA2D_FGPFCCR 833, 869
DMA2D_IFCR 830, 866
DMA2D_ISR 830, 866
DMA2D_LWR 843, 879
DMA2D_NLR 843, 878
DMA2D_OCOLR 840, 841, 875, 876, 877
DMA2D_OMAR 842, 877
DMA2D_OOR 842, 878
DMA2D_OPFCCR 839, 874
DP_ABORTR 3468
DP_CTRL/STATR 3468
DP_DLCR 3470
DP_DLPIDR 3471
DP_DPIDR 3467
DP_EVENTSTATR 3471
DP_RDBUFFR 3473
DP_RESENDR 3472
DP_SELECTR 3472
DP_TARGETIDR 3470
DSI_BCFGR 1829
DSI_CCR 1781
DSI_CLCR 1799
DSI_CLTCR 1799
DSI_CMCR 1792
DSI_CR 1781
DSI_DLTCR 1800
DSI_DLTRCR 1812
DSI_DPCBCR 1829
DSI_DPCSRCR 1830
DSI_DPDL0BCR 1832
DSI_DPDL0HSOCR 1830
DSI_DPDL0LPXOCR 1831
DSI_DPDL0SRCR 1832
DSI_DPDL1BCR 1834
DSI_DPDL1HSOCR 1833
DSI_DPDL1LPXOCR 1833
DSI_DPDL1SRCR 1834
DSI_FBSR 1819
DSI_FIR0 1810
DSI_FIR1 1811
DSI_GHCR 1793
DSI_GPDR 1794
DSI_GPSR 1794
DSI_GVCIDR 1785
DSI_IER0 1806
DSI_IER1 1808
DSI_ISR0 1803
DSI_ISR1 1804
DSI_LCCCR 1813
DSI_LCCR 1791
DSI_LCOLCR 1782
DSI_LCVCIDR 1813
DSI_LPCR 1783
DSI_LPMCCR 1814
DSI_LPMCR 1784
DSI_LVCIDR 1782
DSI_MCR 1786
DSI_PCONFR 1801
DSI_PCR 1784
DSI_PCTLR 1800
DSI_PSR 1802
DSI_PTTCR 1802
DSI_PUCR 1801
DSI_TCCR0 1796
DSI_TCCR1 1796
DSI_TCCR2 1797
DSI_TCCR3 1797
DSI_TCCR4 1798
DSI_TCCR5 1798
DSI_VCCCR 1816
DSI_VCCR 1788
DSI_VHBPCCR 1817
DSI_VHBPCR 1789
DSI_VHSACCR 1817
DSI_VHSACR 1789

DSI_VLCCR 1818
DSI_VLCR 1790
DSI_VMCCR 1815
DSI_VMCR 1786
DSI_VNPCCR 1817
DSI_VNPCR 1789
DSI_VPCCR 1816
DSI_VPCR 1788
DSI_VR 1781
DSI_VSCR 1812
DSI_VVACCR 1819
DSI_VVACR 1791
DSI_VVBPCCR 1818
DSI_VVBPCR 1790
DSI_VVFPCCR 1819
DSI_VVFPCR 1791
DSI_VVSACCR 1818
DSI_VVSACR 1790
DSI_WCFGR 1822
DSI_WCR 1823
DSI_WIER 1824
DSI_WIFCR 1825
DSI_WISR 1824
DSI_WPCR0 1826
DSI_WPTR 1828
DSI_WRPCR 1827
DWT_CIDR0 3507
DWT_CIDR1 3507
DWT_CIDR2 3508
DWT_CIDR3 3508
DWT_COMPxR 3499
DWT_CPICNTR 3497
DWT_CTRLR 3495
DWT_CYCCNTR 3496
DWT_DEVARCHR 3503
DWT_DEVTYPER 3504
DWT_EXCCNTR 3497
DWT_FOLDCNTR 3498
DWT_FUNCTR0 3499
DWT_FUNCTR1 3500
DWT_FUNCTR2 3501
DWT_FUNCTR3 3502
DWT_LSUCNTR 3498
DWT_PCSR 3498
DWT_PIDR0 3505
DWT_PIDR1 3505
DWT_PIDR2 3506
DWT_PIDR3 3506
DWT_PIDR4 3504
DWT_SLPCNTR 3497

E

ETM_AUTHSTATR 3549
ETM_CCCTLR 3534
ETM_CIDR0 3553
ETM_CIDR1 3554
ETM_CIDR2 3554
ETM_CIDR3 3554
ETM_CLAIMCLR 3549
ETM_CLAIMSETR 3548
ETM_CNTRLDVR0 3536
ETM_CONFIGR 3531
ETM_DEVARCHR 3550
ETM_DEVTYPER 3550
ETM_EVENTCTL0R 3532
ETM_EVENTCTL1R 3533
ETM_IDR0 3539
ETM_IDR1 3540
ETM_IDR10 3537
ETM_IDR11 3537
ETM_IDR12 3538
ETM_IDR13 3538
ETM_IDR2 3540
ETM_IDR3 3541
ETM_IDR4 3542
ETM_IDR5 3543
ETM_IDR8 3536
ETM_IDR9 3537
ETM_IMSPECR0 3538
ETM_PDCR 3547
ETM_PDSR 3548
ETM_PIDR0 3551
ETM_PIDR1 3552
ETM_PIDR2 3552
ETM_PIDR3 3553
ETM_PIDR4 3551
ETM_PRGCTLR 3530
ETM_RSCTLR2 3544
ETM_RSCTLR3 3545
ETM_SSCCR0 3545
ETM_SSCSR0 3546
ETM_SSPCICR0 3547
ETM_STALLCTLR 3533
ETM_STATR 3531
ETM_SYNCPR 3534
ETM_TRACEIDR 3535
ETM_VICTLR 3535
EXTI_EMR1 923
EXTI_EXTICRm 920
EXTI_FPR1 918
EXTI_FTSR1 916
EXTI_IMR1 923
EXTI_LOCKR 922

EXTI_PRIVCFGR1 919
EXTI_RPR1 917
EXTI_RTSR1 915
EXTI_SECCFGR1 919
EXTI_SWIER1 917

F

FDCAN_CCCR 3019
FDCAN_CKDIV 3046
FDCAN_CREL 3016
FDCAN_DBTP 3017
FDCAN_ECR 3025
FDCAN_ENDN 3016
FDCAN_HPMS 3036
FDCAN_IE 3031
FDCAN_ILE 3034
FDCAN_ILS 3033
FDCAN_IR 3028
FDCAN_NBTP 3021
FDCAN_PSR 3025
FDCAN_RWD 3019
FDCAN_RXF0A 3038
FDCAN_RXF0S 3037
FDCAN_RXF1A 3039
FDCAN_RXF1S 3038
FDCAN_RXGFC 3034
FDCAN_TDCR 3028
FDCAN_TEST 3018
FDCAN_TOCC 3023
FDCAN_TOCV 3024
FDCAN_TSCC 3022
FDCAN_TSCV 3023
FDCAN_TXBAR 3041
FDCAN_TXBC 3039
FDCAN_TXBCF 3043
FDCAN_TXBCIE 3044
FDCAN_TXBCR 3042
FDCAN_TXBRP 3040
FDCAN_TXBTIE 3043
FDCAN_TXBTO 3042
FDCAN_TXEFA 3045
FDCAN_TXEFS 3044
FDCAN_TXFQS 3040
FDCAN_XIDAM 3036
FLASH_ACR 322
FLASH_ECCR 334
FLASH_NSBOOTADD0R 338
FLASH_NSBOOTADD1R 339
Flash_NSCR 330
Flash_NSKEYR 324
Flash_NSSR 326
FLASH_OEM1KEYR1 349
FLASH_OEM1KEYR2 349
FLASH_OEM2KEYR1 350
FLASH_OEM2KEYR2 350
FLASH_OPSR 335
FLASH_OPTKEYR 325
FLASH_OPTR 336
FLASH_PDKEY1R 325
FLASH_PDKEY2R 326
FLASH_PRIVBB1Rx 353
FLASH_PRIVBB2Rx 354
Flash_PRIVCFGR 352
FLASH_SECBB1Rx 351
FLASH_SECBB2Rx 351
FLASH_SECBOOTADD0R 340
Flash_SECCR 332
Flash_SECHDPCR 352
Flash_SECKEYR 324
Flash_SECSR 329
FLASH_SECWM1R1 341
FLASH_SECWM1R2 342
FLASH_SECWM2R1 345
FLASH_SECWM2R2 346
FLASH_WRP1AR 343
FLASH_WRP1BR 344
FLASH_WRP2AR 347
FLASH_WRP2BR 348
FMAC_CR 974
FMAC_PARAM 973
FMAC_RDATA 977
FMAC_SR 975
FMAC_WDATA 977
FMAC_X1BUFCFG 971
FMAC_X2BUFCFG 971
FMAC_YBUFCFG 972
FMC_BCRx 1012
FMC_BTRx 1015
FMC_BWTRx 1017
FMC_ECCR 1030
FMC_PATT 1029
FMC_PCR 1026
FMC_PCSCNTR 1019
FMC_PMEM 1028
FMC_SR 1027

G

GFXMMU_B0CR 897
GFXMMU_B1CR 898
GFXMMU_B2CR 898
GFXMMU_B3CR 899
GFXMMU_CCR 896

GFXMMU_CR 893
GFXMMU_DVR 897
GFXMMU_FCR 895
GFXMMU_LUTxH 900
GFXMMU_LUTxL 899
GFXMMU_SR 895
GFXTIM_AFCC1R 2532
GFXTIM_AFCR 2531
GFXTIM_ALCC1R 2533
GFXTIM_ALCC2R 2533
GFXTIM_ALCR 2532
GFXTIM_ATR 2531
GFXTIM_CGCR 2515
GFXTIM_CR 2514
GFXTIM_EVCR 2519
GFXTIM_EVSR 2520
GFXTIM_FCCRR 2530
GFXTIM_ICR 2525
GFXTIM_IER 2527
GFXTIM_ISR 2523
GFXTIM_LCCRR 2530
GFXTIM_RFC1R 2533
GFXTIM_RFC1RR 2534
GFXTIM_RFC2R 2534
GFXTIM_RFC2RR 2535
GFXTIM_TCR 2517
GFXTIM_TDR 2518
GFXTIM_TSR 2529
GFXTIM_WDGCR 2535
GFXTIM_WDGPAR 2536
GFXTIM_WDGRR 2536
GFXTIM_WDGTCR 2522
GPDMA_CxBR1 . . . . . 743, 745
GPDMA_CxBR2 750
GPDMA_CxCR 734
GPDMA_CxDAR 748
GPDMA_CxFCR 731
GPDMA_CxLBAR 731
GPDMA_CxLLR 751, 753
GPDMA_CxSAR 747
GPDMA_CxSR 732
GPDMA_CxTR1 736
GPDMA_CxTR2 739
GPDMA_CxTR3 749
GPDMA_MISR 729
GPDMA_PRIVCFGR 728
GPDMA_RCFGLOCKR 729
GPDMA_SECCFGR 727
GPDMA_SMISR 730
GPIOx_AFRH 634
GPIOx_AFRL 633
GPIOx_BRR 635
GPIOx_BSRR 631
GPIOx_HSLVR 635
GPIOx_IDR 630
GPIOx_LCKR 632
GPIOx_MODER 628
GPIOx_ODR 631
GPIOx_OSPEEDR 629
GPIOx_OTYPER 629
GPIOx_PUPDR 630
GPIOx_SECCFGR 636
GTZC1_MPCBBz_CFGLOCKR1 251
GTZC1_MPCBBz_CFGLOCKR2 251
GTZC1_MPCBBz_CR 250
GTZC1_MPCBBz_PRIVCFGRx 252
GTZC1_MPCBBz_SECCFGRx 252
GTZC1_TZIC_FCR1 239
GTZC1_TZIC_FCR2 241
GTZC1_TZIC_FCR3 242
GTZC1_TZIC_FCR4 246
GTZC1_TZIC_IER1 220
GTZC1_TZIC_IER2 222
GTZC1_TZIC_IER3 224
GTZC1_TZIC_IER4 227
GTZC1_TZIC_SR1 229
GTZC1_TZIC_SR2 232
GTZC1_TZIC_SR3 233
GTZC1_TZIC_SR4 236
GTZC1_TZSC_CR 200
GTZC1_TZSC_MPCWMxAR 216
GTZC1_TZSC_MPCWMxBR 217
GTZC1_TZSC_MPCWMxzCFGR 215
GTZC1_TZSC_PRIVCFGR1 208
GTZC1_TZSC_PRIVCFGR2 210
GTZC1_TZSC_PRIVCFGR3 212
GTZC1_TZSC_SECCFGR1 200
GTZC1_TZSC_SECCFGR2 203
GTZC1_TZSC_SECCFGR3 204
GTZC2_MPCBB4_CFGLOCKR1 268
GTZC2_MPCBB4_CR 267
GTZC2_MPCBB4_PRIVCFGR0 269
GTZC2_MPCBB4_SECCFGR0 268
GTZC2_TZIC_FCR1 263
GTZC2_TZIC_FCR2 264
GTZC2_TZIC_IER1 258
GTZC2_TZIC_IER2 259
GTZC2_TZIC_SR1 260
GTZC2_TZIC_SR2 262
GTZC2_TZSC_CR 254
GTZC2_TZSC_PRIVCFGR1 256
GTZC2_TZSC_SECCFGR1 254

H

HASH_CR 2008
HASH_CSRx 2016
HASH_DIN 2010
HASH_HRAx 2012
HASH_HRx 2013
HASH_IMR 2014
HASH_SR 2014
HASH_STR 2011
HSPI_ABR 1144
HSPI_AR 1139
HSPI_CALFCR 1153
HSPI_CALMR 1153
HSPI_CALSIR 1155
HSPI_CALSOR 1154
HSPI_CCR 1141
HSPI_CR 1130
HSPI_DCR1 1133
HSPI_DCR2 1134
HSPI_DCR3 1135
HSPI_DCR4 1136
HSPI_DLR 1138
HSPI_DR 1139
HSPI_FCR 1137
HSPI_HLCR 1152
HSPI_IR 1144
HSPI_LPTR 1144
HSPI_PIR 1141
HSPI_PSMAR 1140
HSPI_PSMKR 1140
HSPI_SR 1136
HSPI_TCR 1143
HSPI_WABR 1152
HSPI_WCCR 1149
HSPI_WIR 1151
HSPI_WPABR 1148
HSPI_WPCCR 1145
HSPI_WPIR 1148
HSPI_WPTCR 1147
HSPI_WTCR 1151

I

I2C_AUTOCR 2727
I2C_CR1 2713
I2C_CR2 2716
I2C_ICR 2724
I2C_ISR 2722
I2C_OAR1 2718
I2C_OAR2 2719
I2C_PECR 2725
I2C_RXDR 2726
I2C_TIMEOUTR 2721
I2C_TIMINGR 2720
I2C_TXDR 2726
ICACHE_CR 370
ICACHE_CRRx 373
ICACHE_FCR 372
ICACHE_HMONR 372
ICACHE_IER 371
ICACHE_MMONR 373
ICACHE_SR 371
ITM_CIDR0 3517
ITM_CIDR1 3518
ITM_CIDR2 3518
ITM_CIDR3 3519
ITM_DEVARCHR 3514
ITM_DEVTYPER 3514
ITM_PIDR0 3515
ITM_PIDR1 3516
ITM_PIDR2 3516
ITM_PIDR3 3517
ITM_PIDR4 3515
ITM_STIMRx 3511
ITM_TCR 3513
ITM_TER 3512
ITM_TPR 3512
IWDG_EWCR 2552
IWDG_KR 2548
IWDG_PR 2549
IWDG_RLR 2550
IWDG_SR 2550
IWDG_WINR 2551

J

JPEG_CFR 1857
JPEG_CONFR0 1851
JPEG_CONFR1 1852
JPEG_CONFR2 1853
JPEG_CONFR3 1853
JPEG_CONFRx 1854
JPEG_CR 1855
JPEG_DHTMEMx 1862
JPEG_DIR 1858
JPEG_DOR 1858
JPEG_HUFFBASEx 1861
JPEG_HUFFENC_ACx_y 1863
JPEG_HUFFENC_DCx_y 1863
JPEG_HUFFMINx_y 1860
JPEG_HUFFSYMBx 1861
JPEG_QMEMx_y 1859
JPEG_SR 1856

L

LPDMA_CxBR1 806
LPDMA_CxCR 800
LPDMA_CxDAR 808
LPDMA_CxFCR 797
LPDMA_CxLBAR 797
LPDMA_CxLLR 809
LPDMA_CxSAR 807
LPDMA_CxSR 798
LPDMA_CxTR1 802
LPDMA_CxTR2 804
LPDMA_MISR 795
LPDMA_PRIVCFGR 794
LPDMA_RCFGLOCKR 795
LPDMA_SECCFGR 793
LPDMA_SMISR 796
LPGPIO_BRR 643
LPGPIO_BSRR 642
LPGPIO_IDR 641
LPGPIO_MODER 641
LPGPIO_ODR 642
LPTIM_ARR 2493
LPTIM_CCMR1 2495
LPTIM_CCR1 2492
LPTIM_CCR2 2498
LPTIM_CFGR 2488
LPTIM_CFGR2 2494
LPTIM_CNT 2493
LPTIM_CR 2491
LPTIM_RCR 2495
LPTIM4_DIER 2483
LPTIM4_ICR 2479
LPTIM4_ISR 2474
LPTIMx_DIER 2484, 2486
LPTIMx_ICR 2480, 2482
LPTIMx_ISR 2475, 2477
LPUART_AUTOCR 2865
LPUART_BRR 2853
LPUART_CR1 . . . . . 2843, 2846
LPUART_CR2 2849
LPUART_CR3 2851
LPUART_ICR 2862
LPUART_ISR 2855, 2859
LPUART_PRESC 2864
LPUART_RDR 2863
LPUART_RQR 2854
LPUART_TDR 2863
LTDC_AWCR 1705
LTDC_BCCR 1708
LTDC_BPCR 1704
LTDC_CDSR 1712
LTDC_CPSR 1711
LTDC_GCR 1706
LTDC_ICR 1710
LTDC_IER 1709
LTDC_ISR 1710
LTDC_LIPCR 1711
LTDC_LxBFCR 1717
LTDC_LxCACR 1716
LTDC_LxCFBAR 1718
LTDC_LxCFBLNR 1719
LTDC_LxCFBLR 1718
LTDC_LxCKCR 1715
LTDC_LxCLUTWR 1719
LTDC_LxCR 1712
LTDC_LxDCCR 1716
LTDC_LxPFCR 1715
LTDC_LxWHPCR 1713
LTDC_LxWVPCR 1714
LTDC_SRCR 1708
LTDC_SSCR 1704
LTDC_TWCR 1706

M

MCUROM_CIDR0 3485
MCUROM_CIDR1 3485
MCUROM_CIDR2 3486
MCUROM_CIDR3 3486
MCUROM_MEMTYPER 3482
MCUROM_PIDR0 3483
MCUROM_PIDR1 3483
MCUROM_PIDR2 3484
MCUROM_PIDR3 3484
MCUROM_PIDR4 3482
MDF_BSMXxCR 1554
MDF_CKGCR 1550
MDF_DFLT0IER 1565
MDF_DFLT0ISR 1567
MDF_DFLTxCICR 1557
MDF_DFLTxCR 1555
MDF_DFLTxDR 1572
MDF_DFLTxIER 1566
MDF_DFLTxINTR 1560
MDF_DFLTxISR 1569
MDF_DFLTxRSFR 1559
MDF_DLYxCR 1563
MDF_GCR 1549
MDF_OECxCR 1571
MDF_OLDxCR 1561
MDF_OLDxTHHR 1563
MDF_OLDxTHLR 1562
MDF_SCDxCR 1564
MDF_SITFxCR 1553

MDF_SNPSxDR 1572

O

OCTOSPI_ABR 1078
OCTOSPI_AR 1072
OCTOSPI_CCR 1075
OCTOSPI_CR 1064
OCTOSPI_DCR1 1066
OCTOSPI_DCR2 1068
OCTOSPI_DCR3 1069
OCTOSPI_DCR4 1070
OCTOSPI_DLR 1072
OCTOSPI_DR 1073
OCTOSPI_FCR 1071
OCTOSPI_HLCR 1086
OCTOSPI_IR 1078
OCTOSPI_LPTR 1079
OCTOSPI_PIR 1075
OCTOSPI_PSMAR 1074
OCTOSPI_PSMKR 1074
OCTOSPI_SR 1070
OCTOSPI_TCR 1077
OCTOSPI_WABR 1086
OCTOSPI_WCCR 1083
OCTOSPI_WIR 1085
OCTOSPI_WPABR 1082
OCTOSPI_WPCCR 1079
OCTOSPI_WPIR 1082
OCTOSPI_WPTCR 1081
OCTOSPI_WTCR 1085
OCTOSPIM_CR 1094
OCTOSPIM_PnCR 1094
OPAMP1_CSR 1482
OPAMP1_LPOTR 1484
OPAMP1_OTR 1484
OPAMP2_CRS 1485
OPAMP2_LPOTR 1487
OPAMP2_OTR 1486
OTFDEC_CR 2029
OTFDEC_ICR 2037
OTFDEC_IER 2038
OTFDEC_ISR 2036
OTFDEC_PRIVCFGR 2030
OTFDEC_RxCFGR 2030
OTFDEC_RxENDADDR 2033
OTFDEC_RxKEYR0 2034
OTFDEC_RxKEYR1 2035
OTFDEC_RxKEYR2 2035
OTFDEC_RxKEYR3 2036
OTFDEC_RxNONCER0 2033
OTFDEC_RxNONCER1 2034
OTFDEC_RxSTARTADDR 2032
OTG_CID 3149, 3305
OTG_DAINT 3172, 3330
OTG_DAINTMSK 3173, 3331
OTG_DCFG 3166, 3323
OTG_DCTL 3167, 3325
OTG_DIEPCTL0 3175
OTG_DIEPCTLx 3176, 3333, 3335
OTG_DIEPDMAx 3339
OTG_DIEPEMPMSK 3174, 3332
OTG_DIEPINTx 3179, 3337
OTG_DIEPMSK 3170, 3328
OTG_DIEPTSIZ0 . . . . . 3180, 3338
OTG_DIEPTSIZx 3182, 3340
OTG_DIEPTXF0 . . . . . 3146, 3302
OTG_DIEPTXFx 3154, 3310
OTG_DOEPCTL0 . . . . . 3183, 3341
OTG_DOEPCTLx 3187, 3345, 3348
OTG_DOEPDMAx 3345
OTG_DOEPINTx 3184, 3342
OTG_DOEPMSK 3171, 3329
OTG_DOEPTSIZ0 . . . . . 3186, 3344
OTG_DOEPTSIZx 3189, 3350
OTG_DSTS 3169, 3327
OTG_DTHRCTL 3331
OTG_DTXFSTSx 3181, 3339
OTG_DVBUSDIS 3173
OTG_DVBUSPULSE 3174
OTG_GAHBCFG 3130, 3280
OTG_GCCFG 3148, 3304
OTG_GINTMSK 3139, 3294, 3295
OTG_GINTSTS 3135, 3285, 3289
OTG_GLPMCFG 3150, 3306
OTG_GOTGCTL 3126, 3277
OTG_GOTGINT 3129, 3279
OTG_GRSTCTL 3133, 3283
OTG_GRXFSIZ 3146, 3301
OTG_GRXSTSP 3144, 3145, 3299, 3300
OTG_GRXSTSR 3142, 3143, 3297, 3298
OTG_GUSBCFG 3131, 3281
OTG_HAINT 3158, 3314
OTG_HAINTMSK 3159, 3314
OTG_HCCHARx 3162, 3317
OTG_HCDMAx 3323
OTG_HCFG 3155, 3311
OTG_HCINTMSKx 3164, 3321
OTG_HCINTx 3163, 3320
OTG_HCSPLTx 3319
OTG_HCTSIZx 3165, 3322
OTG_HFIR 3156, 3311
OTG_HFNUM 3156, 3312
OTG_HNPTXFSIZ 3146, 3302

OTG_HNPTXSTS 3147, 3303
OTG_HPRT 3159, 3315
OTG_HPTXFSIZ 3154, 3310
OTG_HPTXSTS 3157, 3313
OTG_PCGCCTL 3190, 3351
OTG_PCGCCTL1 3352

P

PKA_CLRFR 2070
PKA_CR 2067
PKA_SR 2069
PSSI_CR 1685
PSSI_DR 1690
PSSI_ICR 1689
PSSI_IER 1688
PSSI_MIS 1688
PSSI_RIS 1687
PSSI_SR 1687
PWR_APCR 459
PWR_BDCR1 449
PWR_BDCR2 450
PWR_BDSR 456
PWR_CR1 436
PWR_CR2 438
PWR_CR3 441
PWR_CR4 470
PWR_CR5 471
PWR_DBPR 450
PWR_PDCRA 460
PWR_PDCRB 461
PWR_PDCRC 462
PWR_PDCRD 463
PWR_PDCRE 464
PWR_PDCRF 465
PWR_PDCRG 466
PWR_PDCRH 467
PWR_PDCRI 468
PWR_PDCRJ 469
PWR_PRIVCFGR 453
PWR_PUCRA 459
PWR_PUCRB 460
PWR_PUCRC 461
PWR_PUCRD 462
PWR_PUCRE 463
PWR_PUCRF 464
PWR_PUCRG 465
PWR_PUCRH 466
PWR_PUCRI 467
PWR_PUCRJ 468
PWR_SECCFGR 452
PWR_SR 454
PWR_SVMCR 444
PWR_SVMSR 455
PWR_UCPDR 451
PWR_VOSR 442
PWR_WUCR1 445
PWR_WUCR2 446
PWR_WUCR3 447
PWR_WUSCR 458
PWR_WUSR 456

R

RAMCFG_M2WPR1 280
RAMCFG_M2WPR2 280
RAMCFG_MxCR 276
RAMCFG_MxDEAR 279
RAMCFG_MxECCKEYR 281
RAMCFG_MxERKEYR 281
RAMCFG_MxICR 279
RAMCFG_MxIER 277
RAMCFG_MxISR 278
RAMCFG_MxSEAR 278
RCC_AHB1ENR 544
RCC_AHB1RSTR 531
RCC_AHB1SMENR 561
RCC_AHB2ENR1 547
RCC_AHB2ENR2 551
RCC_AHB2RSTR1 533
RCC_AHB2RSTR2 536
RCC_AHB2SMENR1 564
RCC_AHB2SMENR2 568
RCC_AHB3ENR 552
RCC_AHB3RSTR 537
RCC_AHB3SMENR 570
RCC_APB1ENR1 554
RCC_APB1ENR2 556
RCC_APB1RSTR1 538
RCC_APB1RSTR2 540
RCC_APB1SMENR1 571
RCC_APB1SMENR2 574
RCC_APB2ENR 557
RCC_APB2RSTR 541
RCC_APB2SMENR 576
RCC_APB3ENR 559
RCC_APB3RSTR 543
RCC_APB3SMENR 578
RCC_BDCR 591
RCC_CCIPR1 583
RCC_CCIPR2 586
RCC_CCIPR3 589
RCC_CFGR1 511
RCC_CFGR2 513

RCC_CFGR3 515
RCC_CICR 530
RCC_CIER 527
RCC_CIFR 528
RCC_CR 503
RCC_CRRCR 510
RCC_CSR 594
RCC_ICSCR1 507
RCC_ICSCR2 509
RCC_ICSCR3 510
RCC_PLL1CFGR 516
RCC_PLL1DIVR 521
RCC_PLL1FRACR 522
RCC_PLL2CFGR 518
RCC_PLL2DIVR 523
RCC_PLL2FRACR 524
RCC_PLL3CFGR 519
RCC_PLL3DIVR 525
RCC_PLL3FRACR 526
RCC_PRIVCFGR 598
RCC_SECCFGR 596
RCC_SRDAMR 580
RNG_CR 1895
RNG_DR 1898
RNG_HTCR 1898
RNG_SR 1897
RTC_ALRABINR 2610
RTC_ALRBBINR 2611
RTC_ALRMAR 2601
RTC_ALRMASSR 2603
RTC_ALRMBR 2604
RTC_ALRMBSSR 2605
RTC_CALR 2597
RTC_CR 2591
RTC_DR 2585
RTC_ICSR 2587
RTC_MISR 2607
RTC_PRER 2589
RTC_PRIVCFGR 2594
RTC_SCR 2609
RTC_SECCFGR 2595
RTC_SHIFTR 2598
RTC_SMISR 2608
RTC_SR 2606
RTC_SSR 2586
RTC_TR 2584
RTC_TSDR 2600
RTC_TSSSR 2601
RTC_TSTR 2599
RTC_WPR 2596
RTC_WUTR 2590

S

SAES_CR 1980
SAES_DINR 1985
SAES_DOUTR 1986
SAES_ICR 1993
SAES_IER 1991
SAES_ISR 1992
SAES_IVR0 1988
SAES_IVR1 1989
SAES_IVR2 1989
SAES_IVR3 1989
SAES_KEYR0 1987
SAES_KEYR1 1987
SAES_KEYR2 1988
SAES_KEYR3 1988
SAES_KEYR4 1990
SAES_KEYR5 1990
SAES_KEYR6 1990
SAES_KEYR7 1991
SAES_SR 1983
SAI_ACLRFR 2978
SAI_ACR1 2957
SAI_ACR2 2963
SAI_ADR 2980
SAI_AFRCR 2967
SAI_AIM 2971
SAI_ASLOTR 2969
SAI_ASR 2974
SAI_BCLRFR 2979
SAI_BCR1 2960
SAI_BCR2 2965
SAI_BDR 2980
SAI_BFRCR 2968
SAI_BIM 2973
SAI_BSLOTR 2970
SAI_BSR 2976
SAI_GCR 2956
SAI_PDMCR 2981
SAI_PDMDLY 2982
SDMMC_ACKTIMER 1231
SDMMC_ARGR 1216
SDMMC_CLKCR 1214
SDMMC_CMDR 1216
SDMMC_DCNTR 1222
SDMMC_DCTRL 1220
SDMMC_DLENR 1220
SDMMC_DTIMER 1219
SDMMC_FIFORx 1231
SDMMC_ICR 1225
SDMMC_IDMABAR 1234
SDMMC_IDMABASER 1233
SDMMC_IDMABSIZER 1232

SDMMC_IDMACTRLR 1232
SDMMC_IDMALAR 1233
SDMMC_MASKR 1228
SDMMC_POWER 1213
SDMMC_RESPCMDR 1218
SDMMC_RESPxR 1218
SDMMC_STAR 1223
SPI_AUTOCR 2913
SPI_CFG1 2903
SPI_CFG2 2906
SPI_CR1 2901
SPI_CR2 2903
SPI_CRCPOLY 2915
SPI_IER 2908
SPI_IFCR 2912
SPI_RXCRC 2917
SPI_RXDR 2914
SPI_SR 2909
SPI_TXCRC 2916
SPI_TXDR 2914
SPI_UDRDR 2917
SYSCFG_CCCR 658
SYSCFG_CCCSR 655
SYSCFG_CCVR 657
SYSCFG_CFGR1 649
SYSCFG_CFGR2 653
SYSCFG_CNSLCKR 651
SYSCFG_CSLCKR 652
SYSCFG_FPUIMR 651
SYSCFG_MESR 654
SYSCFG_OTGHSPHYCR 659
SYSCFG_OTGHSPHYTUNER2 660
SYSCFG_RSSCMDR 659
SYSCFG_SECCFGR 648

T

TAMP_ATCR1 2637
TAMP_ATCR2 2641
TAMP_ATOR 2640
TAMP_ATSEEDR 2639
TAMP_BKPxR 2657
TAMP_COUNT1R 2656
TAMP_CR1 2630
TAMP_CR2 2632
TAMP_CR3 2634
TAMP_ERCFGR 2656
TAMP_FLTCR 2636
TAMP_IER 2647
TAMP_MISR 2651
TAMP_PRIVCFGR 2646
TAMP_SCR 2654
TAMP_SECCFGR 2644
TAMP_SMISR 2652
TAMP_SR 2649
TIM15_AF1 2396
TIM15_AF2 2398
TIM15_ARR 2388
TIM15_BDTR 2391
TIM15_CCER 2384
TIM15_CCMR1 . . . . . 2380, 2382
TIM15_CCR1 2389
TIM15_CCR2 2390
TIM15_CNT 2387
TIM15_CR1 2371
TIM15_CR2 2372
TIM15_DCR 2399
TIM15_DIER 2376
TIM15_DMAR 2400
TIM15_DTR2 2394
TIM15_EGR 2379
TIM15_PSC 2387
TIM15_RCR 2388
TIM15_SMCR 2374
TIM15_SR 2377
TIM15_TISEL 2395
TIMx_AF1 . . . . . 2205, 2317, 2421
TIMx_AF2 . . . . . 2208, 2318, 2424
TIMx_ARR 2190, 2310, 2414, 2447
TIMx_BDTR 2195, 2417
TIMx_CCER 2185, 2307, 2411
TIMx_CCMR1 . . 2177, 2179, 2301, 2303, 2408, 2409
TIMx_CCMR2 . . . . . 2181, 2182, 2305, 2306
TIMx_CCMR3 2201
TIMx_CCR1 . . . . . 2191, 2311, 2415
TIMx_CCR2 . . . . . 2192, 2312
TIMx_CCR3 . . . . . 2193, 2313
TIMx_CCR4 . . . . . 2194, 2314
TIMx_CCR5 2199
TIMx_CCR6 2200
TIMx_CNT 2189, 2309, 2413, 2446
TIMx_CR1 . . . . . 2163, 2291, 2403, 2442
TIMx_CR2 . . . . . 2165, 2293, 2404, 2444
TIMx_DCR 2211, 2319, 2425
TIMx_DIER 2172, 2297, 2405, 2444
TIMx_DMAR 2212, 2320, 2426
TIMx_DTR2 . . . . . 2202, 2420
TIMx_ECR 2203, 2315
TIMx_EGR 2176, 2300, 2407, 2445
TIMx_OR1 2424
TIMx_PSC 2189, 2309, 2414, 2446
TIMx_RCR 2190, 2415
TIMx_SMCR 2168, 2294

TIMx_SR 2173, 2298, 2406, 2445
TIMx_TISEL 2204, 2316, 2421
TPIU_ACPR 3561
TPIU_CIDR0 3568
TPIU_CIDR1 3568
TPIU_CIDR2 3568
TPIU_CIDR3 3569
TPIU_CLAIMCLR 3564
TPIU_CLAIMSETR 3563
TPIU_CSPSR 3560
TPIU_DEVIDR 3564
TPIU_DEVTYPER 3565
TPIU_FFCR 3562
TPIU_FFSR 3561
TPIU_PIDR0 3566
TPIU_PIDR1 3566
TPIU_PIDR2 3567
TPIU_PIDR3 3567
TPIU_PIDR4 3565
TPIU_PSCR 3563
TPIU_SPPR 3561
TPIU_SSPSR 3560
TSC_CR 1875
TSC_ICR 1878
TSC_IER 1877
TSC_IOASCR 1879
TSC_IOCCR 1880
TSC_IOGCSR 1881
TSC_IOGxCR 1881
TSC_IOHCR 1879
TSC_IOSCR 1880
TSC_ISR 1878

U

UCPD_CFGR1 3440
UCPD_CFGR2 3442
UCPD_CFGR3 3443
UCPD_CR 3443
UCPD_ICR 3450
UCPD_IMR 3446
UCPD_RX_ORDEXTR1 3455
UCPD_RX_ORDEXTR2 3455
UCPD_RX_ORDSETR 3453
UCPD_RX_PAYSZR 3454
UCPD_RXDR 3454
UCPD_SR 3447
UCPD_TX_ORDSETR 3451
UCPD_TX_PAYSZR 3452
UCPD_TXDR 3452
USART_AUTOCR 2811
USART_BRR 2794
USART_CR1 . . . . . .2778, 2782
USART_CR2 2786
USART_CR3 2790
USART_GTPR 2794
USART_ICR 2808
USART_ISR 2797, 2803
USART_PRESC 2810
USART_RDR 2809
USART_RQR 2796
USART_RTOR 2795
USART_TDR 2809
USB_BCDR 3079
USB_CHEP_RXTXBD_n 3092
USB_CHEP_TXRXBD_n 3088, 3089, 3091
USB_CHEPnR 3080
USB_CNTR 3070
USB_DADDR 3078
USB_FNR 3077
USB_ISTR 3073
USB_LPMCSR 3078

V

VREFBUF_CCR 1463
VREFBUF_CSR 1462

W

WWDG_CFR 2560
WWDG_CR 2559
WWDG_SR 2560