参考資料

RM0433
リファレンスマニュアル

STM32H7x3 advanced ARM®-based 32-bit MCUs

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32H7x3 マイクロコントローラメモリとペリフェラルを使用する方法について、詳しく説明しています。

STM32H7x3 は、さまざまなメモリサイズ、パッケージ、およびペリフェラルを持つマイクロコントローラファミリ製品になります。

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

ARM® FPU 搭載 Cortex®-M7 コアについては、対応する ARM の *テクニカルリファレンスマニュアル* を参照してください。

関連ドキュメント

- ARM® Cortex®-M7 テクニカルリファレンスマニュアル（www.arm.com から入手可能）。
- Cortex®-M7 プログラミングマニュアル（PM0253）。

参考資料

目次

1 このマニュアルにおける表記の規則 96

1.1 レジスタに関する略記 96

1.2 用語 96

1.3 使用可能なペリフェラル 97

2 メモリとバスのアーキテクチャ 98

2.1 システムアーキテクチャ 98

2.1.1 バスマトリックス 100

2.1.2 バス間のブリッジ 100

2.1.3 ドメイン間バス 101

2.1.4 CPU バス 101

2.1.5 バスマスタペリフェラル 102

2.1.6 クロックと機能ブロック 103

2.2 メモリ構成 104

2.2.1 概要 104

2.2.2 メモリマップとレジスタ境界アドレス 104

2.3 内蔵 SRAM 108

2.4 フラッシュメモリの概要 109

2.5 ブート設定 109

3 内蔵フラッシュメモリ（フラッシュ）. 111

3.1 概要 111

3.2 FLASH の主な機能 111

3.3 FLASH の機能説明 112

3.3.1 ブロック図 112

3.3.2 ピンおよび内部信号 112

3.3.3 フラッシュメモリのアーキテクチャ 113

3.3.4 フラッシュ読み出し操作 115

3.3.5 エラーコード訂正（ECC） 118

3.3.6 巡回冗長検査モジュール 119

3.3.7 フラッシュのプログラムおよび消去操作 120

3.3.8 ユーザオプションバイトの変更 127

3.3.9 フラッシュインタフェースエラーフラグ 128

3.3.10 バンク 1 およびバンク 2 での同時読み出し／プログラム／消去 129

参考資料

3.3.11 フラッシュオプションバイト 130
3.3.12 保護メカニズム 132
3.3.13 フラッシュバンクスワッピング 137
3.4 FLASH 割り込み 139
3.5 フラッシュレジスタ 139
3.5.1 フラッシュアクセス制御レジスタ（FLASH_ACR）.....139
3.5.2 バンク 1 のフラッシュキーレジスタ（FLASH_KEYR1）.....140
3.5.3 フラッシュオプションキーレジスタ（FLASH_OPTKEYR）.....141
3.5.4 バンク 1 のフラッシュ制御レジスタ（FLASH_CR1）.....141
3.5.5 バンク 1 のフラッシュステータスレジスタ（FLASH_SR1）.....145
3.5.6 バンク 1 のフラッシュクリア制御レジスタ（FLASH_CCR1）.....148
3.5.7 フラッシュオプション制御レジスタ（FLASH_OPTCR）.....149
3.5.8 フラッシュオプションステータスレジスタ（現在値）（FLASH_OPTSR_CUR）.....151
3.5.9 フラッシュオプションステータスレジスタ（プログラムする値）（FLASH_OPTSR_PRG）153
3.5.10 フラッシュオプションクリア制御レジスタ（FLASH_OPTCCR）.....156
3.5.11 バンク 1 のフラッシュ保護アドレス（現在値）（FLASH_PRAR_CUR1）....157
3.5.12 バンク 1 のフラッシュ保護アドレス（プログラムする値）（FLASH_PRAR_PRG1）158
3.5.13 バンク 1 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR1）.....159
3.5.14 バンク 1 のフラッシュセキュアアドレス（プログラムする値）（FLASH_SCAR_PRG1）160
3.5.15 バンク 1 のフラッシュ書き込みセクタ保護（現在値）（FLASH_WPSN_CUR1R）.....161
3.5.16 バンク 1 のフラッシュ書き込みセクタ保護（プログラムする値）（FLASH_WPSN_PRG1R）161
3.5.17 ブートアドレスのフラッシュレジスタ（現在値）（FLASH_BOOT_CURR）.....162
3.5.18 ブートアドレスのフラッシュレジスタ（プログラムする値）（FLASH_BOOT_PRGR）162
3.5.19 バンク 1 のフラッシュ CRC 制御レジスタ（FLASH_CRCCR1）.....163
3.5.20 バンク 1 のフラッシュ CRC 開始アドレスレジスタ（FLASH_CRCSADD1R）.....164
3.5.21 バンク 1 のフラッシュ CRC 終了アドレスレジスタ（FLASH_CRCEADD1R）.....165
3.5.22 フラッシュ CRC データレジスタ（FLASH_CRCDATAR）.....165
3.5.23 バンク 1 のフラッシュ ECC 失敗アドレス（FLASH_ECC_FA1R）.....166
3.5.24 バンク 2 のフラッシュキーレジスタ（FLASH_KEYR2）.....166
3.5.25 バンク 2 のフラッシュ制御レジスタ（FLASH_CR2）.....167

参考資料

3.5.26 バンク 2 のフラッシュステータスレジスタ（FLASH_SR2）.....170
3.5.27 バンク 2 のフラッシュクリア制御レジスタ（FLASH_CCR2）.....173
3.5.28 バンク 2 のフラッシュ保護アドレス（現在値）（FLASH_PRAR_CUR2）....175
3.5.29 バンク 2 のフラッシュ保護アドレス（プログラムする値）.....（FLASH_PRAR_PRG2）176
3.5.30 バンク 2 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR2）.....176
3.5.31 バンク 2 のフラッシュセキュアアドレス（プログラムする値）.....（FLASH_SCAR_PRG2）177
3.5.32 バンク 2 のフラッシュ書き込みセクタ保護（現在値）（FLASH_WPSN_CUR2R）.....178
3.5.33 バンク 2 のフラッシュ書き込みセクタ保護（プログラムする値）.....（FLASH_WPSN_PRG2R）178
3.5.34 バンク 2 のフラッシュ CRC 制御レジスタ（FLASH_CRCCR2）.....179
3.5.35 バンク 2 のフラッシュ CRC 開始アドレスレジスタ（FLASH_CRCSADD2R）.....180
3.5.36 バンク 2 のフラッシュ CRC 終了アドレスレジスタ（FLASH_CRCEADD2R）.....181
3.5.37 バンク 2 のフラッシュ ECC 失敗アドレス（FLASH_ECC_FA2R）.....181
3.6 フラッシュレジスタマップとリセット値 182

4 セキュリティメモリ管理 187
4.1 概要 187
4.2 用語 187
4.3 フラッシュ保護 188
4.4 セキュアアクセスモード 189
4.4.1 関連機能 190
4.4.2 ブートステートマシン 190
4.4.3 セキュアアクセスモード設定 191
4.5 ルートセキュアサービス（RSS）..... 192
4.5.1 ルートセキュアサービスの呼び出し 192
4.5.2 ルートセキュアサービスの説明 193
4.6 セキュアユーザソフトウェア 193
4.6.1 アクセス規則 194
4.6.2 セキュアユーザメモリ領域の設定 194
4.6.3 セキュアユーザメモリ領域の削除 194
4.6.4 セキュアユーザソフトウェアの選択 195
4.7 フラッシュ保護メカニズムの概要 196

参考資料

5 AXI インタコネクト 197

5.1 AXI の概要 197

5.2 AXI インタコネクトの主な機能 197

5.3 AXI インタコネクトの機能説明 198

5.3.1 ブロック図 198

5.3.2 ASIB の設定 198

5.3.3 AMIB の設定 199

5.3.4 サービス品質（QoS） 199

5.3.5 Global programmer's view（GPV） 199

5.4 AXI インタコネクトレジスタ 200

5.4.1 AXI インタコネクト - ペリフェラル ID4 レジスタ（AXI_PERIPH_ID_4） 200

5.4.2 AXI インタコネクト - ペリフェラル ID0 レジスタ（AXI_PERIPH_ID_0） 200

5.4.3 AXI インタコネクト - ペリフェラル ID1 レジスタ（AXI_PERIPH_ID_1） 201

5.4.4 AXI インタコネクト - ペリフェラル ID2 レジスタ（AXI_PERIPH_ID_2） 201

5.4.5 AXI インタコネクト - ペリフェラル ID3 レジスタ（AXI_PERIPH_ID_3） 202

5.4.6 AXI インタコネクト - コンポーネント ID0 レジスタ（AXI_COMP_ID_0） 202

5.4.7 AXI インタコネクト - コンポーネント ID1 レジスタ（AXI_COMP_ID_1） 203

5.4.8 AXI インタコネクト - コンポーネント ID2 レジスタ（AXI_COMP_ID_2） 203

5.4.9 AXI インタコネクト - コンポーネント ID3 レジスタ（AXI_COMP_ID_3） 204

5.4.10 AXI インタコネクト - TARG x バスマトリックス発行機能レジスタ . . . . . （AXI_TARGx_FN_MOD_ISS_BM）204

5.4.11 AXI インタコネクト - TARG x バスマトリックス機能 2 レジスタ . . . . . （AXI_TARGx_FN_MOD2）205

5.4.12 AXI インタコネクト - TARG x ロングバースト機能変更レジスタ . . . . . （AXI_TARGx_FN_MOD_LB）205

5.4.13 AXI インタコネクト - TARG x 発行機能変更レジスタ （AXI_TARGx_FN_MOD） 206

5.4.14 AXI インタコネクト - INI x 機能変更 2 レジスタ（AXI_INIx_FN_MOD2） 206

5.4.15 AXI インタコネクト - INI x AHB 機能変更レジスタ （AXI_INIx_FN_MOD_AHB） 207

5.4.16 AXI インタコネクト - INI x 読み出し QoS レジスタ （AXI_INIx_READ_QOS） 207

5.4.17 AXI インタコネクト - INI x 書き込み QoS レジスタ （AXI_INIx_WRITE_QOS） 208

5.4.18 AXI インタコネクト - INI x 発行機能変更レジスタ （AXI_INIx_FN_MOD） 208

5.5 AXI インタコネクトレジスタマップ 209

参考資料

6 電源制御（PWR） 217

6.1 概要 217

6.2 PWR の主な機能 217

6.3 PWR ブロック図 218

6.3.1 PWR ピンおよび内部信号 219

6.4 電源 220

6.4.1 システム電源の起動 223

6.4.2 コアドメイン 225

6.4.3 PWR 外部電源 226

6.4.4 バックアップドメイン 226

6.4.5 VBAT バッテリの充電 228

6.4.6 アナログ電源 228

6.4.7 USB レギュレータ 229

6.5 電源供給の監視 229

6.5.1 パワーオンリセット（POR）／パワーダウンリセット（PDR） 230

6.5.2 ブラウンアウトリセット（BOR） 230

6.5.3 プログラム可能な電圧検出器（PVD） 231

6.5.4 アナログ電圧検出器（AVD） 233

6.5.5 バッテリ電圧の閾値 234

6.5.6 温度の閾値 235

6.6 電源管理 236

6.6.1 動作モード 237

6.6.2 電圧スケーリング 240

6.6.3 電源制御モード 241

6.6.4 電源管理の例 244

6.7 低電力モード 250

6.7.1 システムクロックの低速化 250

6.7.2 ペリフェラルクロックの制御 250

6.7.3 低電力モードへの移行 250

6.7.4 低電力モードの終了 251

6.7.5 CSleep モード 252

6.7.6 CStop モード 253

6.7.7 DStop モード 254

6.7.8 STOP モード 255

6.7.9 DStandby モード 257

6.7.10 STANDBY モード 259

参考資料

6.8 PWR レジスタの説明 261
6.8.1 PWR 制御レジスタ 1（PWR_CR1）. 261
6.8.2 PWR 制御ステータスレジスタ 1（PWR_CSR1）. 262
6.8.3 PWR 制御レジスタ 2（PWR_CR2）. 263
6.8.4 PWR 制御レジスタ 3（PWR_CR3）. 264
6.8.5 PWR CPU 制御レジスタ（PWR_CPUCR）. 266
6.8.6 PWR D3 ドメイン制御レジスタ（PWR_D3CR）. 267
6.8.7 PWR ウェイクアップクリアレジスタ（PWR_WKUPCR）. 268
6.8.8 PWR ウェイクアップフラグレジスタ（PWR_WKUPFR）. 268
6.8.9 PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）. 269
6.8.10 PWR レジスタマップ 270

7 低消費電力 D3 ドメイン 271
7.1 概要 271
7.2 EXTI、RCC、および PWR の相互接続 271
7.2.1 割り込みとウェイクアップ 273
7.2.2 ブロックの相互作用 273
7.2.3 D3 ドメイン DMAMUX2 の役割 274
7.3 LPUART1 送信に基づく低消費電力アプリケーションの例 275
7.3.1 メモリ保持 275
7.3.2 LPUART1 インタフェースを使用したメモリからペリフェラルへの転送 275
7.3.3 LPUART1 送信に基づく低消費電力アプリケーションの例の全体説明 280
7.3.4 代替実装 281
7.4 その他の低消費電力アプリケーション 282

8 リセットおよびクロック制御（RCC）. 283
8.1 RCC の主な機能 283
8.2 RCC ブロック図 284
8.3 RCC ピンおよび内部信号 284
8.4 RCC リセットブロックの機能説明 286
8.4.1 パワーオン／オフリセット 286
8.4.2 システムリセット 287
8.4.3 ローカルリセット 288
8.4.4 リセットソース識別 289
8.4.5 低電力モードセキュリティリセット（lpwr_rst）. 290
8.4.6 バックアップドメインリセット 291
8.4.7 パワーオンおよびウェイクアップのシーケンス 291

8.5 RCC クロックブロックの機能説明 293
8.5.1 クロックの命名規則 295
8.5.2 オシレータの説明 295
8.5.3 クロックセキュリティシステム（CSS） 300
8.5.4 クロック出力の生成（MCO1/MCO2） 301
8.5.5 PLL の説明 302
8.5.6 システムクロック（sys_ck） 306
8.5.7 STOP モードおよび STANDBY モードでのクロック生成の処理 308
8.5.8 カーネルクロックの選択 310
8.5.9 汎用クロックの概念の概要 322
8.5.10 ペリフェラルの割り当て 326
8.5.11 ペリフェラルクロックゲーティングの制御 328
8.5.12 CPU およびバスマトリックスクロックゲーティングの制御 333
8.6 RCC 割り込み 335
8.7 RCC レジスタの説明 336
8.7.1 レジスタ配置の概要 336
8.7.2 RCC ソース制御レジスタ（RCC_CR） 337
8.7.3 RCC 内部クロックソース較正レジスタ（RCC_ICSCR） 341
8.7.4 RCC クロックリカバリ RC レジスタ（RCC_CRRCR） 342
8.7.5 RCC クロック設定レジスタ（RCC_CFGR） 343
8.7.6 RCC ドメイン 1 クロック設定レジスタ（RCC_D1CFGR） 346
8.7.7 RCC ドメイン 2 クロック設定レジスタ（RCC_D2CFGR） 348
8.7.8 RCC ドメイン 3 クロック設定レジスタ（RCC_D3CFGR） 349
8.7.9 RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR） 350
8.7.10 RCC PLL 設定レジスタ（RCC_PLLCFGR） 352
8.7.11 RCC PLL1 分周器設定レジスタ（RCC_PLL1DIVR） 355
8.7.12 RCC PLL1 分数分周器レジスタ（RCC_PLL1FRACR） 357
8.7.13 RCC PLL2 分周器設定レジスタ（RCC_PLL2DIVR） 358
8.7.14 RCC PLL2 分数分周器レジスタ（RCC_PLL2FRACR） 360
8.7.15 RCC PLL3 分周器設定レジスタ（RCC_PLL3DIVR） 361
8.7.16 RCC PLL3 分数分周器レジスタ（RCC_PLL3FRACR） 363
8.7.17 RCC ドメイン 1 カーネルクロック設定レジスタ（RCC_D1CCIPR） 364
8.7.18 RCC ドメイン 2 カーネルクロック設定レジスタ（RCC_D2CCIP1R） 365
8.7.19 RCC ドメイン 2 カーネルクロック設定レジスタ（RCC_D2CCIP2R） 367
8.7.20 RCC ドメイン 3 カーネルクロック設定レジスタ（RCC_D3CCIPR） 369
8.7.21 RCC クロックソース割り込み有効レジスタ（RCC_CIER） 371
8.7.22 RCC クロックソース割り込みフラグレジスタ（RCC_CIFR） 373
8.7.23 RCC クロックソース割り込みクリアレジスタ（RCC_CICR） 375

参考資料

8.7.24 RCC バックアップドメイン制御レジスタ（RCC_BDCR）.....377
8.7.25 RCC クロック制御およびステータスレジスタ（RCC_CSR）.....379
8.7.26 RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）.....380
8.7.27 RCC AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）.....382
8.7.28 RCC AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）.....384
8.7.29 RCC AHB4 ペリフェラルリセットレジスタ（RCC_AHB4RSTR）.....385
8.7.30 RCC APB3 ペリフェラルリセットレジスタ（RCC_APB3RSTR）.....387
8.7.31 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1LRSTR）.....388
8.7.32 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1HRSTR）.....391
8.7.33 RCC APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）.....392
8.7.34 RCC APB4 ペリフェラルリセットレジスタ（RCC_APB4RSTR）.....394
8.7.35 RCC グローバル制御レジスタ（RCC_GCR）.....396
8.7.36 RCC D3 自律モードレジスタ（RCC_D3AMR）.....397
8.7.37 RCC リセットステータスレジスタ（RCC_RSR）.....400
8.7.38 RCC AHB3 クロックレジスタ（RCC_AHB3ENR）.....402
8.7.39 RCC AHB1 クロックレジスタ（RCC_AHB1ENR）.....404
8.7.40 RCC AHB2 クロックレジスタ（RCC_AHB2ENR）.....406
8.7.41 RCC AHB4 クロックレジスタ（RCC_AHB4ENR）.....408
8.7.42 RCC APB3 クロックレジスタ（RCC_APB3ENR）.....410
8.7.43 RCC APB1 クロックレジスタ（RCC_APB1LENR）.....411
8.7.44 RCC APB1 クロックレジスタ（RCC_APB1HENR）.....415
8.7.45 RCC APB2 クロックレジスタ（RCC_APB2ENR）.....416
8.7.46 RCC APB4 クロックレジスタ（RCC_APB4ENR）.....419
8.7.47 RCC AHB3 スリープクロックレジスタ（RCC_AHB3LPENR）.....422
8.7.48 RCC AHB1 スリープクロックレジスタ（RCC_AHB1LPENR）.....424
8.7.49 RCC AHB2 スリープクロックレジスタ（RCC_AHB2LPENR）.....426
8.7.50 RCC AHB4 スリープクロックレジスタ（RCC_AHB4LPENR）.....428
8.7.51 RCC APB3 スリープクロックレジスタ（RCC_APB3LPENR）.....431
8.7.52 RCC APB1 ロースリープクロックレジスタ（RCC_APB1LLPENR）.....432
8.7.53 RCC APB1 ハイスリープクロックレジスタ（RCC_APB1HLPENR）.....436
8.7.54 RCC APB2 スリープクロックレジスタ（RCC_APB2LPENR）.....437
8.7.55 RCC APB4 スリープクロックレジスタ（RCC_APB4LPENR）.....440
8.8 RCC レジスタマップ.....443

9 クロックリカバリシステム（CRS）.....452
9.1 概要.....452
9.2 CRS の主な機能.....452
9.3 CRS の機能詳細.....453

参考資料

9.3.1 CRS ブロック図 453
9.4 CRS 内部信号 454
9.4.1 同期入力 454
9.4.2 周波数誤差測定 454
9.4.3 周波数誤差評価および自動トリミング 456
9.4.4 CRS の初期化と設定 456
9.5 CRS 低電力モード 457
9.6 CRS 割り込み 457
9.7 CRS レジスタ 458
9.7.1 CRS 制御レジスタ（CRS_CR） 458
9.7.2 CRS 設定レジスタ（CRS_CFGR） 459
9.7.3 CRS 割り込みおよびステータスレジスタ（CRS_ISR） 460
9.7.4 CRS 割り込みフラグクリアレジスタ（CRS_ICR） 462
9.7.5 CRS レジスタマップ 463

10 ハードウェアセマフォ（HSEM） 464
10.1 ハードウェアセマフォの概要 464
10.2 ハードウェアセマフォの主な機能 464
10.3 HSEM の機能詳細 464
10.3.1 HSEM ブロック図 464
10.3.2 HSEM 内部信号 465
10.3.3 HSEM ロックの手順 465
10.3.4 HSEM 書き込み / 読み出し / 読み出しロックレジスタアドレス 467
10.3.5 HSEM クリアの手順 467
10.3.6 HSEM CoreID セマフォのクリア 467
10.3.7 HSEM 割り込み 468
10.3.8 AHB バスマスタ ID 照合 469
10.4 HSEM レジスタ 471
10.4.1 HSEM レジスタ（HSEM_R0 - HSEM_R31） 471
10.4.2 HSEM 読み出しロックレジスタ（HSEM_RLR0 - HSEM_RLR31） 472
10.4.3 HSEM 割り込み有効レジスタ（HSEM_CnIER） 473
10.4.4 HSEM 割り込みクリアレジスタ（HSEM_CnICR） 473
10.4.5 HSEM 割り込みステータスレジスタ（HSEM_CnISR） 474
10.4.6 HSEM マスク済み割り込みステータスレジスタ（HSEM_CnMISR） 474
10.4.7 HSEM クリアレジスタ（HSEM_CR） 475
10.4.8 HSEM 割り込みクリアレジスタ（HSEM_KEYR） 475
10.4.9 HSEM レジスタマップ 476

参考資料

11 汎用 I/O（GPIO）. 478

11.1 概要 478

11.2 GPIO の主な機能 478

11.3 GPIO の機能説明 478

11.3.1 汎用 I/O（GPIO）. 481

11.3.2 I/O ピンオルタネート機能マルチプレクサと配置 481

11.3.3 I/O ポート制御レジスタ 482

11.3.4 I/O ポートデータレジスタ 482

11.3.5 I/O データのビット単位の操作 482

11.3.6 GPIO ロック機構 483

11.3.7 I/O オルタネート機能の入力／出力 483

11.3.8 外部割り込み／ウェイクアップライン 483

11.3.9 入力設定 484

11.3.10 出力設定 484

11.3.11 I/O 補正セル 485

11.3.12 オルタネート機能設定 485

11.3.13 アナログ設定 486

11.3.14 HSE または LSE オシレータのピンを GPIO として使用 487

11.3.15 GPIO ピンをバックアップ供給ドメインで使用 487

11.4 GPIO レジスタ 488

11.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x =A ～ K）. 488

11.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPERR）（x = A ～ K） 488

11.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ～ K）. 489

11.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A ～ K）. 489

11.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A ～ K）. 490

11.4.6 GPIO ポート入力データレジスタ（GPIOx_ODR）（x = A ～ K）. 490

11.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A ～ K）. 491

11.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ～ K） 491

11.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A ～ K）. 492

11.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ～ J）. 493

11.4.11 GPIO レジスタマップ 494

参考資料

12 システム設定コントローラ（SYSCFG）. 496

12.1 概要 496

12.2 SYSCFG の主な機能 496

12.3 SYSCFG レジスタの説明 496

12.3.1 SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR） 496

12.3.2 SYSCFG 外部割り込み設定レジスタ 1（SYSCFG_EXTICR1） 498

12.3.3 SYSCFG 外部割り込み設定レジスタ 2（SYSCFG_EXTICR2） 499

12.3.4 SYSCFG 外部割り込み設定レジスタ 3（SYSCFG_EXTICR3） 500

12.3.5 SYSCFG 外部割り込み設定レジスタ 4（SYSCFG_EXTICR4） 501

12.3.6 SYSCFG 補正セル制御／ステータスレジスタ（SYSCFG_CCCSR） 502

12.3.7 SYSCFG 補正セル値レジスタ（SYSCFG_CCVR） 503

12.3.8 SYSCFG 補正セルコードレジスタ（SYSCFG_CCCR） 503

12.3.9 SYSCFG パッケージレジスタ（SYSCFG_PKGR） 504

12.3.10 SYSCFG ユーザレジスタ 0 （SYSCFG_UR0） 505

12.3.11 SYSCFG ユーザレジスタ 2 （SYSCFG_UR2） 505

12.3.12 SYSCFG ユーザレジスタ 3 （SYSCFG_UR3） 506

12.3.13 SYSCFG ユーザレジスタ 4 （SYSCFG_UR4） 506

12.3.14 SYSCFG ユーザレジスタ 5 （SYSCFG_UR5） 507

12.3.15 SYSCFG ユーザレジスタ 6 （SYSCFG_UR6） 507

12.3.16 SYSCFG ユーザレジスタ 7 （SYSCFG_UR7） 508

12.3.17 SYSCFG ユーザレジスタ 8 （SYSCFG_UR8） 508

12.3.18 SYSCFG ユーザレジスタ 9 （SYSCFG_UR9） 509

12.3.19 SYSCFG ユーザレジスタ 10 （SYSCFG_UR10） 509

12.3.20 SYSCFG ユーザレジスタ 11 （SYSCFG_UR11） 510

12.3.21 SYSCFG ユーザレジスタ 12 （SYSCFG_UR12） 510

12.3.22 SYSCFG ユーザレジスタ 13 （SYSCFG_UR13） 511

12.3.23 SYSCFG ユーザレジスタ 14 （SYSCFG_UR14） 512

12.3.24 SYSCFG ユーザレジスタ 15 （SYSCFG_UR15） 513

12.3.25 SYSCFG ユーザレジスタ 16 （SYSCFG_UR16） 514

12.3.26 SYSCFG ユーザレジスタ 17 （SYSCFG_UR17） 514

12.3.27 SYSCFG レジスタマップ 515

13 ブロック相互接続 518

13.1 ペリフェラル相互接続 518

13.1.1 概要 518

13.1.2 接続の概要 518

13.2 低電力モードからのウェイクアップ 537

13.3 DMA 542
13.3.1 MDMA （D1 ドメイン） 543
13.3.2 DMAMUX1、DMA1、および DMA2 （D2 ドメイン） 544
13.3.3 DMAMUX2、BDMA （D3 ドメイン） 549

14 MDMA コントローラ（MDMA） 552
14.1 MDMA の概要 552
14.2 MDMA の主な機能 552
14.3 MDMA の機能詳細 554
14.3.1 MDMA ブロック図 554
14.3.2 MDMA 内部信号 554
14.3.3 MDMA の概要 555
14.3.4 MDMA チャネル 556
14.3.5 転送元、転送先、および転送モード 556
14.3.6 ポインタの更新 556
14.3.7 MDMA バッファ転送 557
14.3.8 リクエストアービトレーション 558
14.3.9 FIFO 558
14.3.10 ブロック転送 559
14.3.11 ブロック繰り返しモード 559
14.3.12 リンクされたリストモード 559
14.3.13 MDMA 転送の完了 559
14.3.14 MDMA 転送の中断 560
14.3.15 エラー管理 560
14.4 MDMA 割り込み 560
14.5 MDMA レジスタ 561
14.5.1 MDMA グローバル割り込み／ステータスレジスタ（MDMA_GISR0） 561
14.5.2 MDMA チャネル x 割り込み／ステータスレジスタ（MDMA_CxISR）（x = 0 ～ 15） 562
14.5.3 MDMA チャネル x 割り込みフラグクリアレジスタ（MDMA_CxIFCR）（x = 0 ～ 15） 563
14.5.4 MDMA チャネル x エラーステータスレジスタ（MDMA_CxESR）（x = 0 ～ 15） 564
14.5.5 MDMA チャネル x 制御レジスタ（MDMA_CxCR）（x = 0 ～ 15） 566
14.5.6 MDMA チャネル x 転送設定レジスタ（MDMA_CxTCR）（x = 0 ～ 15） 568
14.5.7 MDMA チャネル x データレジスタのブロック数（MDMA_CxBNDTR）（x = 0 ～ 15） 571
14.5.8 MDMA チャネル x 転送元アドレスレジスタ（MDMA_CxSAR）（x = 0 ～ 15） 573

14.5.9 MDMA チャネル x 転送先アドレスレジスタ（MDMA_CxDAR）（x = 0 ～ 15）.....573
14.5.10 MDMA チャネル x ブロック繰り返しアドレス更新レジスタ MDMA_CxBRUR （x = 0 ～ 15）.....575
14.5.11 MDMA チャネル x リンクアドレスレジスタ（MDMA_CxLAR）（x = 0 ～ 15）.....576
14.5.12 MDMA チャネル x トリガおよびバス選択レジスタ（MDMA_CxTBR）（x = 0 ～ 15）.....577
14.5.13 MDMA チャネル x マスクアドレスレジスタ（MDMA_CxMAR）（x = 0 ～ 15）.....578
14.5.14 MDMA チャネル x マスクデータレジスタ（MDMA_CxMDR）（x = 0 ～ 15）.....578
14.5.15 MDMA レジスタマップ 579

15 ダイレクトメモリアクセスコントローラ（DMA1、DMA2）.....581

15.1 DMA の概要 581
15.2 DMA の主な機能 581
15.3 DMA の機能説明 583
15.3.1 DMA ブロック図 583
15.3.2 DMA 内部信号 583
15.3.3 DMA の概要 584
15.3.4 DMA トランザクション 584
15.3.5 DMA リクエストマッピング 584
15.3.6 アービタ 585
15.3.7 DMA ストリーム 585
15.3.8 転送元、転送先、および転送モード 585
15.3.9 ポインタのインクリメント 588
15.3.10 サーキュラモード 589
15.3.11 ダブルバッファモード 589
15.3.12 プログラム可能なデータ幅、パッキング／アンパッキング、エンディアン形式 590
15.3.13 シングル転送とバースト転送 592
15.3.14 FIFO 592
15.3.15 DMA 転送の完了 595
15.3.16 DMA 転送の中断 596
15.3.17 フローコントローラ 596
15.3.18 実現可能な DMA 設定の概要 597
15.3.19 ストリーム設定手順 598
15.3.20 エラー管理 599
15.4 DMA 割り込み 600

参考資料

15.5 DMA レジスタ 601
15.5.1 DMA ロー割り込みステータスレジスタ（DMA_LISR） 601
15.5.2 DMA ハイ割り込みステータスレジスタ（DMA_HISR） 602
15.5.3 DMA ロー割り込みフラグクリアレジスタ（DMA_LIFCR） 603
15.5.4 DMA ハイ割り込みフラグクリアレジスタ（DMA_HIFCR） 603
15.5.5 DMA ストリーム x 設定レジスタ（DMA_SxCR）（x=0 ～ 7） 604
15.5.6 DMA ストリーム x データ数レジスタ（DMA_HIFCR）（x=0 ～ 7） 607
15.5.7 DMA ストリーム x ペリフェラルアドレスレジスタ（DMA_SxPAR）（x=0 ～ 7） 608
15.5.8 DMA ストリーム x メモリ 0 アドレスレジスタ（DMA_SxM0AR）（x=0 ～ 7） 608
15.5.9 DMA ストリーム x メモリ 1 アドレスレジスタ（DMA_SxM1AR）（x=0 ～ 7） 609
15.5.10 DMA ストリーム x FIFO 制御レジスタ（DMA_SxFCR）（x=0 ～ 7） 609
15.5.11 DMA レジスタマップ 611

16 基本ダイレクトメモリアクセスコントローラ（BDMA） 614

16.1 概要 614
16.2 BDMA の主な機能 614
16.3 BDMA の機能説明 615
16.3.1 BDMA トランザクション 615
16.3.2 アービタ 616
16.3.3 BDMA チャネル 616
16.3.4 プログラム可能なデータ幅、データの整列、およびエンディアン 618
16.3.5 エラー管理 619
16.3.6 BDMA 割り込み 619
16.4 BDMA レジスタ 620
16.4.1 DMA 割り込みステータスレジスタ（BDMA_ISR） 620
16.4.2 DMA 割り込みフラグクリアレジスタ（BDMA_IFCR） 621
16.4.3 DMA チャネル x 設定レジスタ（BDMA_CCRx）（x = 1 ～ 8、ここで x = チャネル数） 622
16.4.4 DMA チャネル x データ数レジスタ（BDMA_CNDTRx）（x = 1 ～ 8、x = チャネル番号） 624
16.4.5 DMA チャネル x ペリフェラルアドレスレジスタ（BDMA_CPARx）（x = 1 ～ 8、x = チャネル番号） 624
16.4.6 DMA チャネル x メモリアドレスレジスタ（BDMA_CMARx）（x = 1 ～ 8、x = チャネル番号） 625
16.4.7 BDMA レジスタマップ 626

参考資料

17 DMA リクエストマルチプレクサ（DMAMUX） 629

17.1 概要 629

17.2 DMAMUX の主な機能 629

17.3 DMAMUX の実装 630

17.3.1 DMAMUX1 および DMAMUX2 のインスタンス化 630

17.3.2 DMAMUX1 の配置 630

17.3.3 DMAMUX2 の配置 632

17.4 DMAMUX の機能詳細 634

17.4.1 DMAMUX ブロック図 634

17.4.2 DMAMUX 信号 635

17.4.3 DMAMUX チャネル 635

17.4.4 DMAMUX リクエストラインマルチプレクサ 635

17.4.5 DMAMUX リクエストジェネレータ 638

17.5 DMAMUX 割り込み 639

17.6 DMAMUX レジスタ 640

17.6.1 DMAMUX1 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX1_CxCR） 640

17.6.2 DMAMUX2 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX2_CxCR） 641

17.6.3 DMAMUX1 リクエストラインマルチプレクサ割り込みチャネルステータスレジスタ（DMAMUX1_CSR） 642

17.6.4 DMAMUX2 リクエストラインマルチプレクサ割り込みチャネルステータスレジスタ（DMAMUX2_CSR） 642

17.6.5 DMAMUX1 リクエストラインマルチプレクサ割り込みクリアフラグレジスタ（DMAMUX1_CFR） 643

17.6.6 DMAMUX2 リクエストラインマルチプレクサ割り込みクリアフラグレジスタ（DMAMUX2_CFR） 643

17.6.7 DMAMUX1 リクエストジェネレータチャネル x 設定レジスタ（DMAMUX1_RGxCR） 644

17.6.8 DMAMUX2 リクエストジェネレータチャネル x 設定レジスタ . . . . . （DMAMUX2_RGxCR）644

17.6.9 DMAMUX1 リクエストジェネレータ割り込みステータスレジスタ（DMAMUX1_RGSR） 646

17.6.10 DMAMUX2 リクエストジェネレータ割り込みステータスレジスタ（DMAMUX2_RGSR） 646

17.6.11 DMAMUX1 リクエストジェネレータ割り込みクリアフラグレジスタ（DMAMUX1_RGCFR） 647

17.6.12 DMAMUX2 リクエストジェネレータ割り込みクリアフラグレジスタ（DMAMUX2_RGCFR） 647

17.6.13 DMAMUX レジスタマップ 648

参考資料

18 Chrom-Art Accelerator™ コントローラ（DMA2D）. 650

18.1 DMA2D の概要 650

18.2 DMA2D の主な機能 651

18.3 DMA2D の機能詳細 651

18.3.1 概要 651

18.4 DMA2D ピンおよび内部信号 652

18.4.1 DMA2D 制御 653

18.4.2 DMA2D フォアグラウンドおよびバックグラウンド FIFO 653

18.4.3 DMA2D フォアグラウンドおよびバックグラウンドのピクセルフォーマットコンバータ（PFC） 653

18.4.4 DMA2D フォアグラウンドおよびバックグラウンド CLUT インタフェース 656

18.4.5 DMA2D ブレンダ 657

18.4.6 DMA2D 出力 PFC 657

18.4.7 DMA2D 出力 FIFO 657

18.4.8 DMA2D AXI マスタポートタイマ 658

18.4.9 DMA2D トランザクション 658

18.4.10 DMA2D 設定 659

18.4.11 YCbCr サポート 661

18.4.12 DMA2D 転送制御（開始、サスペンド、アボート、完了） 662

18.4.13 ウォーターマーク 662

18.4.14 エラー管理 662

18.4.15 AXI デッドタイム 663

18.5 DMA2D 割り込み 663

18.6 DMA2D レジスタ 664

18.6.1 DMA2D 制御レジスタ（DMA2D_CR） 664

18.6.2 DMA2D 割り込みステータスレジスタ（DMA2D_ISR） 666

18.6.3 DMA2D 割り込みフラグクリアレジスタ（DMA2D_IFCR） 667

18.6.4 DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR） 668

18.6.5 DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR） 668

18.6.6 DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR） 669

18.6.7 DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR） 669

18.6.8 DMA2D フォアグラウンド PFC 制御レジスタ（DMA2D_FGPFCCR） 670

18.6.9 DMA2D フォアグラウンドカラーレジスタ（DMA2D_FGCOLR） 672

18.6.10 DMA2D バックグラウンド PFC 制御レジスタ（DMA2D_BGPFCCR） 673

18.6.11 DMA2D バックグラウンドカラーレジスタ（DMA2D_BGCOLR） 675

18.6.12 DMA2D フォアグラウンド CLUT メモリアドレスレジスタ（DMA2D_FGCMAR） 675

参考資料

18.6.13 DMA2D バックグラウンド CLUT メモリアドレスレジスタ（DMA2D_BGCMAR）. 676
18.6.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）. 676
18.6.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）. 677
18.6.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）. 678
18.6.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR）. 679
18.6.18 DMA2D ライン数レジスタ（DMA2D_NLR）. 679
18.6.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）. 680
18.6.20 DMA2D AXI マスタタイマ設定レジスタ（DMA2D_AMTCR）. 680
18.6.21 DMA2D レジスタマップ 681

19 ネスト化されたベクタ割り込みコントローラ 683

19.1 NVIC の機能 683
19.1.1 SysTick 較正値レジスタ 683
19.1.2 割り込みベクタと例外ベクタ 683

20 拡張割り込み／イベントコントローラ（EXTI）. 691

20.1 EXTI の主な機能 691
20.2 EXTI ブロック図 691
20.2.1 ペリフェラル、CPU、および D3 ドメイン間の EXTI 接続 692
20.3 EXTI の機能説明 693
20.3.1 EXTI の設定可能なイベント入力、CPU のウェイクアップ 694
20.3.2 EXTI の設定可能なイベント入力、任意のウェイクアップ 695
20.3.3 EXTI のダイレクトイベント入力、CPU のウェイクアップ 697
20.3.4 EXTI のダイレクトイベント入力、任意のウェイクアップ 698
20.3.5 EXTI D3 ペンディングリクエストクリア選択 699
20.4 EXTI イベント入力の配置 699
20.5 EXTI の機能的挙動 702
20.5.1 EXTI CPU 割り込み手順 703
20.5.2 EXTI CPU イベント手順 703
20.5.3 EXTI CPU ウェイクアップ手順 704
20.5.4 自律 RUN モードの D3 ドメインウェイクアップ手順 704
20.5.5 EXTI ソフトウェア割り込み／イベントトリガ手順 704
20.6 EXTI レジスタの説明 705
20.6.1 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）. 705
20.6.2 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）. 705
20.6.3 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER1）. 706
20.6.4 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR1）. 706

20.6.5 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR1L）.....707
20.6.6 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR1H）.....707
20.6.7 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）.....708
20.6.8 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）.....709
20.6.9 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER2）.....709
20.6.10 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR2）.....710
20.6.11 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR2L）.....711
20.6.12 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR2H）.....711
20.6.13 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）.....712
20.6.14 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）.....712
20.6.15 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER3）.....713
20.6.16 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR3）.....713
20.6.17 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR3L）.....714
20.6.18 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR3H）.....714
20.6.19 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR1）.....715
20.6.20 EXTI イベントマスクレジスタ（EXTI_CPUEMR1）.....715
20.6.21 EXTI ペンディングレジスタ（EXTI_CPUPR1）.....716
20.6.22 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR2）.....716
20.6.23 EXTI イベントマスクレジスタ（EXTI_CPUEMR2）.....717
20.6.24 EXTI ペンディングレジスタ（EXTI_CPUPR2）.....717
20.6.25 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR3）.....718
20.6.26 EXTI イベントマスクレジスタ（EXTI_CPUEMR3）.....719
20.6.27 EXTI ペンディングレジスタ（EXTI_CPUPR3）.....719
20.6.28 EXTI レジスタマップ 720

21 巡回冗長検査計算ユニット（CRC）.....723

21.1 概要 723
21.2 CRC の主な機能 723
21.3 CRC の機能説明 724
21.3.1 CRC のブロック図 724
21.3.2 CRC 内部信号 724
21.3.3 CRC 操作 724
21.4 CRC レジスタ 726
21.4.1 データレジスタ（CRC_DR）.....726
21.4.2 独立型データレジスタ（CRC_IDR）.....726
21.4.3 制御レジスタ（CRC_CR）.....727
21.4.4 CRC の初期値（CRC_INIT）.....728
21.4.5 CRC 多項式（CRC_POL）.....728

参考資料

21.4.6 CRC レジスタマップ 729

22 フレキシブルメモリコントローラ（FMC）. 730

22.1 FMC の主な機能 730
22.2 FMC ブロック図 731
22.3 FMC 内部信号 733
22.4 AHB インタフェース 733
22.5 AXI インタフェース 733
22.5.1 サポートされるメモリおよびトランザクション 734
22.6 外部デバイスアドレスマッピング 735
22.6.1 NOR/PSRAM アドレスマッピング 736
22.6.2 NAND フラッシュメモリアドレスマッピング 737
22.6.3 SDRAM アドレスマッピング 737
22.7 NOR 型フラッシュ／ PSRAM コントローラ 740
22.7.1 外部メモリインタフェース信号 742
22.7.2 サポートされるメモリおよびトランザクション 744
22.7.3 一般的なタイミング規則 745
22.7.4 NOR フラッシュ /PSRAM コントローラ非同期トランザクション 745
22.7.5 同期トランザクション 765
22.7.6 NOR/PSRAM コントローラレジスタ 771
22.8 NAND 型フラッシュコントローラ 780
22.8.1 外部メモリインタフェース信号 780
22.8.2 NAND 型フラッシュによってサポートされるメモリとトランザクション 781
22.8.3 NAND フラッシュメモリのタイミング図 782
22.8.4 NAND フラッシュ動作 783
22.8.5 NAND フラッシュのプリウェイト機能 784
22.8.6 NAND 型フラッシュメモリ内のエラー訂正コード（ECC）の計算 785
22.8.7 NAND フラッシュコントローラレジスタ 786
22.9 SDRAM コントローラ 792
22.9.1 SDRAM コントローラの主な機能 792
22.9.2 SDRAM 外部メモリインタフェース信号 792
22.9.3 SDRAM コントローラの機能説明 793
22.9.4 低電力モード 800
22.9.5 SDRAM コントローラレジスタ 803
22.10 FMC レジスタマップ 810

参考資料

23 Quad SPI インタフェース（QUADSPI） 812

23.1 概要 812

23.2 QUADSPI の主な機能 812

23.3 QUADSPI の機能説明 813

23.3.1 QUADSPI ブロック図 813

23.3.2 QUADSPI ピン および内部信号 814

23.3.3 QUADSPI コマンドシーケンス 814

23.3.4 QUADSPI シングルインタフェースプロトコルモード 817

23.3.5 QUADSPI インダイレクトモード 819

23.3.6 QUADSPI ステータスフラグポーリングモード 821

23.3.7 QUADSPI メモリマップドモード 821

23.3.8 QUADSPI フリーランニングクロックモード 822

23.3.9 QUADSPI フラッシュメモリの設定 822

23.3.10 QUADSPI 遅延データサンプリング 823

23.3.11 QUADSPI の設定 823

23.3.12 QUADSPI の使用 824

23.3.13 1 回限りの命令の送信 825

23.3.14 QUADSPI エラー管理 826

23.3.15 QUADSPI の BUSY ビットおよびアボート機能 826

23.3.16 nCS の動作 826

23.4 QUADSPI 割り込み 828

23.5 QUADSPI レジスタ 829

23.5.1 QUADSPI 制御レジスタ（QUADSPI_CR） 829

23.5.2 QUADSPI デバイス設定レジスタ（QUADSPI_DCR） 832

23.5.3 QUADSPI ステータスレジスタ（QUADSPI_SR） 833

23.5.4 QUADSPI フラグクリアレジスタ（QUADSPI_FCR） 834

23.5.5 QUADSPI データ長レジスタ（QUADSPI_DLR） 834

23.5.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR） 835

23.5.7 QUADSPI アドレスレジスタ（QUADSPI_AR） 837

23.5.8 QUADSPI オルタネートバイトレジスタ（QUADSPI_ABR） 838

23.5.9 QUADSPI データレジスタ（QUADSPI_DR） 838

23.5.10 QUADSPI ポーリングステータスマスクレジスタ（QUADSPI _PSMKR） 839

23.5.11 QUADSPI ポーリングステータス一致レジスタ（QUADSPI _PSMAR） 839

23.5.12 QUADSPI ポーリングインターバルレジスタ（QUADSPI _PIR） 840

23.5.13 QUADSPI 低電力タイムアウトレジスタ（QUADSPI_LPTR） 840

23.5.14 QUADSPI レジスタマップ 841

参考資料

24 遅延ブロック（DLYB） 842

24.1 概要 842

24.2 DLYB の主な機能 842

24.3 DLYB の機能詳細 842

24.3.1 DLYB 図 842

24.3.2 DLYB ピンおよび内部信号 843

24.3.3 概要 843

24.3.4 遅延ライン長さ設定手順 844

24.3.5 出力クロックフェーズ設定手順 844

24.4 DLYB レジスタ 845

24.4.1 DLYB 制御レジスタ（DLYB_CR） 845

24.4.2 DLYB 設定レジスタ（DLYB_CFGR） 845

24.4.3 DLYB レジスタのマップ 846

25 アナログデジタルコンバータ（ADC） 847

25.1 概要 847

25.2 ADC の主な機能 848

25.3 ADC の機能詳細 850

25.3.1 ADC ブロック図 850

25.3.2 ADC ピンと内部信号 851

25.3.3 クロック 852

25.3.4 ADC1/2/3 接続性 854

25.3.5 スレーブ AHB インタフェース 857

25.3.6 ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN） 857

25.3.7 シングルエンドおよび差動入力のチャネル 858

25.3.8 較正（ADCAL、ADCALDIF、ADCALLIN、ADCx_CALFACT） 858

25.3.9 ADC オン / オフ制御（ADEN、ADDIS、ADRDY） 864

25.3.10 ADC 制御ビット書き込み時の制約 865

25.3.11 チャネルの選択（SQRx、JSQRx） 865

25.3.12 プリチャネル選択レジスタ（ADCx_PCSEL） 866

25.3.13 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2） . . . 867

25.3.14 シングル変換モード（CONT=0） 868

25.3.15 連続変換モード（CONT=1） 868

25.3.16 変換の開始（ADSTART、JADSTART） 869

25.3.17 タイミング 870

25.3.18 実行中の変換の停止（ADSTP、JADSTP） 870

25.3.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）....872
25.3.20 インジェクトチャネルの管理 875
25.3.21 不連続モード（DISCEN、DISCNUM、JDISCEN）....877
25.3.22 インジェクト変換のコンテキストのキュー 878
25.3.23 プログラム可能な分解能（RES）- 高速変換モード 886
25.3.24 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）....886
25.3.25 変換シーケンスの終了（EOS、JEOS）....886
25.3.26 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）....887
25.3.27 データ管理 888
25.3.28 DFSDM を使用した変換の管理 895
25.3.29 動的低電力機能 895
25.3.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）....900
25.3.31 オーバーサンプリング回路 903
25.3.32 デュアル ADC モード 909
25.3.33 温度センサ 924
25.3.34 VBAT 電源監視 925
25.3.35 内部電圧基準の監視 926
25.4 ADC 割り込み 927
25.5 ADC レジスタ（ADC ごと）....929
25.5.1 ADC 割り込みおよびステータスレジスタ（ADCx_ISR、x=1..3）....929
25.5.2 ADC 割り込み有効レジスタ（ADCx_IER、x=1..3）....931
25.5.3 ADC 制御レジスタ（ADCx_CR、x=1..3）....933
25.5.4 ADC 設定レジスタ（ADCx_CFGR、x=1..3）....938
25.5.5 ADC 設定レジスタ 2（ADCx_CFGR2、x=1..3）....942
25.5.6 ADC サンプル時間レジスタ 1 （ADCx_SMPR1、x=1..3）....944
25.5.7 ADC サンプル時間レジスタ 2 （ADCx_SMPR2、x=1..3）....945
25.5.8 ADC チャネル事前選択レジスタ（ADCx_PCSEL、x=1..3）....946
25.5.9 ADC ウォッチドッグ閾値レジスタ 1 （ADCx_LTR1、x=1..3）....946
25.5.10 ADC ウォッチドッグ閾値レジスタ 1 （ADCx_LHTR1、x=1..3）....947
25.5.11 ADC レギュラシーケンスレジスタ 1 （ADCx_SQR1、x=1..3）....948
25.5.12 ADC レギュラシーケンスレジスタ 2 （ADCx_SQR2、x=1..3）....949
25.5.13 ADC レギュラシーケンスレジスタ 3 （ADCx_SQR3、x=1..3）....950
25.5.14 ADC レギュラシーケンスレジスタ 4 （ADCx_SQR4、x=1..3）....951
25.5.15 ADC レギュラデータレジスタ（ADCx_DR、x=1..3）....952
25.5.16 ADC インジェクトシーケンスレジスタ（ADCx_JSQR、x=1..3）....953
25.5.17 ADC オフセットレジスタ（ADCx_OFRy、x=1..3）（y=1..4）....955

25.5.18 ADC インジェクトデータレジスタ（ADCx_JDRy、x=1..3、y= 1..4）.....956
25.5.19 ADC アナログウォッチドッグ 2 設定レジスタ（ADCx_AWD2CR、x=1..3）.....956
25.5.20 ADC アナログウォッチドッグ 3 設定レジスタ（ADCx_AWD3CR、x=1..3）.....957
25.5.21 ADC ウォッチドッグ低閾値レジスタ 2 （ADCx_LTR2、x=1..3）.....957
25.5.22 ADC ウォッチドッグ高閾値レジスタ 2 （ADCx_HTR2、x=1..3）.....958
25.5.23 ADC ウォッチドッグ低閾値レジスタ 3 （ADCx_LTR3、x=1..3）.....958
25.5.24 ADC ウォッチドッグ高閾値レジスタ 3 （ADCx_HTR3、x=1..3）.....959
25.5.25 ADC 差動モード選択レジスタ（ADCx_DIFSEL、x=1..3）.....959
25.5.26 ADC 較正係数レジスタ（ADCx_CALFACT、x=1..3）.....960
25.5.27 ADC 較正係数レジスタ 2（ADCx_CALFACT2、x=1..3）.....961
25.5.28 ADC オプションレジスタ（ADCx_OR、x=1..3）.....961
25.6 ADC 共通レジスタ 962
25.6.1 ADC 共通ステータスレジスタ（ADCx_CSR、x=12 または 3）.....962
25.6.2 ADC 共通制御レジスタ（ADCx_CCR、x=12 または 3）.....964
25.6.3 デュアルモード用 ADC 共通レギュラデータレジスタ（ADCx_CDR、x=12 または 3）.....967
25.6.4 32 ビットデュアルモード用 ADC 共通レギュラデータレジスタ（ADCx_CDR2、x=12 または 3）.....967
25.6.5 ADC レジスタマップ 968

26 D/A コンバータ（DAC）.....972
26.1 概要 972
26.2 DAC の主な機能 972
26.3 DAC の機能詳細 973
26.3.1 DAC ブロック図 973
26.3.2 DAC ピンおよび内部信号 974
26.3.3 DAC チャネルイネーブル 975
26.3.4 DAC データフォーマット 975
26.3.5 DAC 変換 976
26.3.6 DAC 出力電圧 976
26.3.7 DAC トリガ選択 977
26.3.8 DMA リクエスト 978
26.3.9 ノイズ生成 978
26.3.10 三角波生成 979
26.3.11 DAC チャネルモード 980
26.3.12 DAC 出力バッファ較正 983
26.3.13 デュアル DAC チャネル変換（使用できる場合）.....984

参考資料

26.4 DAC 低電力モード 989
26.5 DAC レジスタ 989
26.5.1 DAC 制御レジスタ（DAC_CR） 989
26.5.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR） 993
26.5.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1） 993
26.5.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1） 994
26.5.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1） . . . 994
26.5.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2） 995
26.5.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2） 995
26.5.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2） . . . 996
26.5.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD） 996
26.5.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD） 997
26.5.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD） 997
26.5.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1） 998
26.5.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2） 998
26.5.14 DAC ステータスレジスタ（DAC_SR） 999
26.5.15 DAC 較正制御レジスタ（DAC_CCR） 1000
26.5.16 DAC モード制御レジスタ（DAC_MCR） 1000
26.5.17 DAC サンプルおよびホールドのサンプル時間レジスタ 1（DAC_SHSR1） 1001
26.5.18 DAC サンプルおよびホールドのサンプル時間レジスタ 2（DAC_SHSR2） 1002
26.5.19 DAC サンプルおよびホールドのホールド時間レジスタ（DAC_SHHR） 1002
26.5.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR） 1003
26.5.21 DAC オプションレジスタ（DAC_OR） 1004
26.5.22 DAC レジスタマップ 1005

27 電圧基準バッファ（VREFBUF） 1007

27.1 概要 1007
27.2 VREFBUF の機能説明 1007
27.3 VREFBUF レジスタ 1008
27.3.1 VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR） 1008
27.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR） 1009
27.3.3 VREFBUF レジスタマップ 1009

参考資料

28 コンパレータ（COMP）. 1010

28.1 概要 1010

28.2 COMP の主な機能 1010

28.3 COMP の機能説明 1011

28.3.1 COMP ブロック図 1011

28.3.2 COMP ピンおよび内部信号 1011

28.3.3 COMP のリセットおよびクロック 1013

28.3.4 コンパレータのロック機構 1013

28.3.5 ウィンドウコンパレータ 1013

28.3.6 ヒステリシス 1013

28.3.7 コンパレータの出力のブランキング機能 1014

28.3.8 GPIO へのコンパレータ出力 1015

28.3.9 コンパレータ出力先変更 1016

28.3.10 COMP 電力とスピードモード 1016

28.4 COMP 低電力モード 1017

28.5 COMP 割り込み 1017

28.5.1 EXTI ブロック経由の割り込み 1017

28.5.2 CPU の NVIC 経由の割り込み 1018

28.6 SCALER 機能 1018

28.7 COMP レジスタ 1019

28.7.1 コンパレータステータスレジスタ（COMP_SR）. 1019

28.7.2 コンパレータ割り込みクリアフラグレジスタ（COMP_ICFR）. 1019

28.7.3 コンパレータオプションレジスタ（COMP_OR）. 1020

28.7.4 コンパレータ設定レジスタ 1（COMP_CFGR1）. 1021

28.7.5 コンパレータ設定レジスタ 2（COMP_CFGR2）. 1023

28.7.6 COMP レジスタマップ 1026

29 オペアンプ（OPAMP）. 1027

29.1 概要 1027

29.2 OPAMP の主な機能 1027

29.3 OPAMP の機能説明 1027

29.3.1 OPAMP のリセットおよびクロック 1027

29.3.2 初期設定 1028

29.3.3 信号接続 1028

29.3.4 OPAMP モード 1029

29.3.5 較正 1035

29.4 OPAMP 低電力モード 1037
29.5 OPAMP PGA ゲイン 1037
29.6 OPAMP レジスタ 1037
29.6.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR） 1037
29.6.2 通常モードでの OPAMP1 トリミングレジスタ（OPAMP1_OTR） 1039
29.6.3 ハイスピードモードでの OPAMP1 トリミングレジスタ（OPAMP1_HSOTR） 1040
29.6.4 OPAMP オプションレジスタ（OPAMP_OR） 1040
29.6.5 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR） 1040
29.6.6 通常モードでの OPAMP2 トリミングレジスタ（OPAMP2_OTR） 1042
29.6.7 ハイスピードモードでの OPAMP2 トリミングレジスタ（OPAMP2_HSOTR） 1043
29.6.8 OPAMP レジスタマップ 1044

30 デルタシグマモジュレータのデジタルフィルタ（DFSDM） 1045
30.1 概要 1045
30.2 DFSDM の主な機能 1046
30.3 DFSDM の実装 1047
30.4 DFSDM の機能詳細 1048
30.4.1 DFSDM ブロック図 1048
30.4.2 DFSDM ピンおよび内部信号 1049
30.4.3 DFSDM のリセットおよびクロック 1050
30.4.4 シリアルチャネルトランシーバ 1051
30.4.5 入力シリアルインタフェースの設定 1060
30.4.6 並列データ入力 1060
30.4.7 チャネル選択 1062
30.4.8 デジタルフィルタ設定 1063
30.4.9 積分器ユニット 1064
30.4.10 アナログウォッチドッグ 1065
30.4.11 短絡検出回路 1067
30.4.12 極値検出回路 1067
30.4.13 データユニットブロック 1068
30.4.14 符号付きデータフォーマット 1069
30.4.15 変換の開始 1069
30.4.16 連続モードおよび高速連続モード 1070
30.4.17 リクエスト優先度 1071
30.4.18 RUN モードでの省電力 1071
30.5 DFSDM 割り込み 1071

参考資料

30.6 DFSDM の DMA 転送 1073
30.7 DFSDM チャネル y レジスタ（y=0 ～ 7）. 1074
30.7.1 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR1）（y=0 ～ 7）. 1074
30.7.2 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR2）（y=0 ～ 7）. 1076
30.7.3 DFSDM チャネルアナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）（y=0 ～ 7）. 1077
30.7.4 DFSDM チャネルウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）（y=0 ～ 7）1078
30.7.5 DFSDM チャネルデータ入力レジスタ（DFSDM_CHyDATINR）（y=0 ～ 7）. 1078
30.8 DFSDM フィルタ x モジュールレジスタ（x=0 ～ 3）. 1079
30.8.1 DFSDM 制御レジスタ 1（DFSDM_FLTxCR1）. 1079
30.8.2 DFSDM 制御レジスタ 2（DFSDM_FLTxCR2）. 1082
30.8.3 DFSDM 割り込みおよびステータスレジスタ（DFSDM_FLTxISR）. 1083
30.8.4 DFSDM 割り込みフラグクリアレジスタ（DFSDM_FLTxICR）. 1085
30.8.5 DFSDM インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR）. 1086
30.8.6 DFSDM フィルタ制御レジスタ（DFSDM_FLTxFCR）. 1087
30.8.7 インジェクトグループの DFSDM データレジスタ（DFSDM_FLTxJDATAR）. 1088
30.8.8 レギュラチャネルの DFSDM データレジスタ（DFSDM_FLTxRDATAR）. . . 1088
30.8.9 DFSDM アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）. 1089
30.8.10 DFSDM アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）. 1090
30.8.11 DFSDM アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）. 1090
30.8.12 DFSDM アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）. 1091
30.8.13 DFSDM 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）. 1091
30.8.14 DFSDM 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）. 1092
30.8.15 DFSDM 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）. 1092
30.8.16 DFSDM レジスタマップ 1093

31 デジタルカメラインタフェース（DCMI）. 1103
31.1 DCMI の概要 1103
31.2 DCMI の主な機能 1103
31.3 DCMI クロック 1103
31.4 DCMI 機能概要 1103
31.4.1 DCMI のブロック図 1104

参考資料

31.4.2 DCMI 内部信号 1105
31.4.3 DMA インタフェース 1105
31.4.4 DCMI 物理インタフェース 1105
31.4.5 同期 1107
31.4.6 キャプチャモード 1110
31.4.7 クロップ機能 1111
31.4.8 JPEG フォーマット 1112
31.4.9 FIFO 1112
31.5 データフォーマットの説明 1113
31.5.1 データフォーマット 1113
31.5.2 モノクロフォーマット 1113
31.5.3 RGB フォーマット 1114
31.5.4 YCbCr フォーマット 1114
31.5.5 YCbCr フォーマット - Y のみ 1115
31.5.6 ハーフ解像度画像抽出 1115
31.6 DCMI 割り込み 1115
31.7 DCMI レジスタの説明 1116
31.7.1 DCMI 制御レジスタ（DCMI_CR） 1116
31.7.2 DCMI ステータスレジスタ（DCMI_SR） 1119
31.7.3 DCMI raw 割り込みステータスレジスタ（DCMI_RIS） 1120
31.7.4 DCMI 割り込み有効レジスタ（DCMI_IER） 1121
31.7.5 DCMI マスク済み割り込みステータスレジスタ（DCMI_MIS） 1122
31.7.6 DCMI 割り込みクリアレジスタ（DCMI_ICR） 1123
31.7.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR） 1124
31.7.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR） 1125
31.7.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT） 1126
31.7.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE） 1126
31.7.11 DCMI データレジスタ（DCMI_DR） 1127
31.7.12 DCMI レジスタマップ 1128

32 LCD-TFT ディスプレイコントローラ（LTDC） 1129
32.1 概要 1129
32.2 LTDC の主な機能 1129
32.3 LTDC の機能詳細 1130
32.3.1 LTDC のブロック図 1130
32.3.2 LCD-TFT 内部信号 1130
32.3.3 LCD-TFT ピンと外部信号インタフェース 1131

参考資料

32.3.4 LTDC のリセットおよびクロック 1131
32.4 LTDC のプログラム可能なパラメータ 1133
32.4.1 LTDC グローバル設定パラメータ 1133
32.4.2 レイヤのプログラム可能なパラメータ 1135
32.5 LTDC 割り込み 1140
32.6 LTDC プログラミング手順 1141
32.7 LTDC レジスタ 1142
32.7.1 LTDC 同期サイズ設定レジスタ（LTDC_SSCR） 1142
32.7.2 LTDC バックポーチ設定レジスタ（LTDC_BPCR） 1142
32.7.3 LTDC アクティブ幅設定レジスタ（LTDC_AWCR） 1143
32.7.4 LTDC 全幅設定レジスタ（LTDC_TWCR） 1144
32.7.5 LTDC グローバル制御レジスタ（LTDC_GCR） 1144
32.7.6 LTDC シャドウ再ロード設定レジスタ（LTDC_SRCR） 1146
32.7.7 LTDC バックグラウンドカラー設定レジスタ（LTDC_BCCR） 1146
32.7.8 LTDC 割り込み有効レジスタ（LTDC_IER） 1147
32.7.9 LTDC 割り込みステータスレジスタ（LTDC_ISR） 1148
32.7.10 LTDC 割り込みクリアレジスタ（LTDC_ICR） 1148
32.7.11 LTDC ライン割り込み位置設定レジスタ（LTDC_LIPCR） 1149
32.7.12 LTDC 現在位置ステータスレジスタ（LTDC_CPSR） 1149
32.7.13 LTDC 現在表示ステータスレジスタ（LTDC_CDSR） 1150
32.7.14 LTDC レイヤ x 制御レジスタ（LTDC_LxCR）（x=1..2） 1151
32.7.15 LTDC レイヤ x ウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）（x=1..2） 1151
32.7.16 LTDC レイヤ x ウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）（x=1..2） 1152
32.7.17 LTDC レイヤ x カラーキーイング設定レジスタ（LTDC_LxCKCR）（x=1..2） 1153
32.7.18 LTDC レイヤ x ピクセルフォーマット設定レジスタ（LTDC_LxPFCR）（x=1..2） 1154
32.7.19 LTDC レイヤ x 定数アルファ設定レジスタ（LTDC_LxCACR）（x=1..2） 1155
32.7.20 LTDC レイヤ x デフォルト色設定レジスタ（LTDC_LxDCCR）（x=1..2） 1155
32.7.21 LTDC レイヤ x ブレンディング係数設定レジスタ（LTDC_LxBFCR）（x=1..2） 1156
32.7.22 LTDC レイヤ x カラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）（x=1..2） 1157
32.7.23 LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）（x=1..2） 1157
32.7.24 LTDC レイヤ x カラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）（x=1..2） 1158
32.7.25 LTDC レイヤ x CLUT 書き込みレジスタ（LTDC_LxCLUTWR）（x=1..2） . . . 1159

参考資料

32.7.26 LTDC レジスタマップ 1160

33 JPEG コーデック（JPEG）. 1163

33.1 概要 1163

33.2 JPEG コーデックの主な機能 1163

33.3 JPEG コーデックブロック の機能説明 1164

33.3.1 概要 1164

33.3.2 JPEG 内部信号 1164

33.3.3 JPEG デコード手順 1165

33.3.4 JPEG エンコード手順 1166

33.4 JPEG コーデック割り込み 1168

33.5 JPEG コーデックレジスタ 1168

33.5.1 JPEG コーデック制御レジスタ（JPEG_CONFR0）. 1168

33.5.2 JPEG コーデック設定レジスタ 1（JPEG_CONFR1）. 1169

33.5.3 JPEG コーデック設定レジスタ 2（JPEG_CONFR2）. 1170

33.5.4 JPEG コーデック設定レジスタ 3（JPEG_CONFR3）. 1170

33.5.5 JPEG コーデック設定レジスタ 4-7（JPEG_CONFR4-7）. 1171

33.5.6 JPEG 制御レジスタ（JPEG_CR）. 1172

33.5.7 JPEG ステータスレジスタ（JPEG_SR）. 1173

33.5.8 JPEG クリアフラグレジスタ（JPEG_CFR）. 1174

33.5.9 JPEG データ入力レジスタ（JPEG_DIR）. 1175

33.5.10 JPEG データ出力レジスタ（JPEG_DOR）. 1175

33.5.11 JPEG コーデックレジスタマップ 1176

34 真の乱数発生器（RNG）. 1178

34.1 概要 1178

34.2 RNG の主な機能 1178

34.3 RNG の機能説明 1179

34.3.1 RNG のブロック図 1179

34.3.2 RNG 内部信号 1179

34.3.3 乱数の生成 1180

34.3.4 RNG 初期化 1183

34.3.5 RNG 動作 1184

34.3.6 RNG クロック供給 1185

34.3.7 エラー管理 1185

34.4 RNG の低消費電力時の取り扱い 1186

34.5 RNG 割り込み 1186

参考資料

34.6 RNG 処理時間 1187
34.7 エントロピーソース検証 1187
34.7.1 概要 1187
34.7.2 検証条件 1187
34.7.3 データ収集 1187
34.8 RNG レジスタ 1189
34.8.1 RNG 制御レジスタ（RNG_CR） 1189
34.8.2 RNG ステータスレジスタ（RNG_SR） 1190
34.8.3 RNG データレジスタ（RNG_DR） 1191
34.8.4 RNG レジスタマップ 1191

35 暗号プロセッサ（CRYP） 1192
35.1 概要 1192
35.2 CRYP の主な機能 1192
35.3 CRYP の機能詳細 1194
35.3.1 CRYP ブロック図 1194
35.3.2 CRYP 内部信号 1194
35.3.3 CRYP DES ／トリプル DES 暗号コア 1195
35.3.4 CRYP AES 暗号コア 1196
35.3.5 暗号操作を行うための CRYP の手順 1202
35.3.6 CRYP ビジーステート 1205
35.3.7 CRYP AES キーを復号化用に準備 1206
35.3.8 CRYP スチーリングおよびデータパディング 1206
35.3.9 CRYP サスペンド／レジューム操作 1208
35.3.10 CRYP DES ／トリプル DES 基本連鎖モード（ECB、CBC） 1209
35.3.11 CRYP AES 基本連鎖モード（ECB、CBC） 1214
35.3.12 CRYP AES カウンタモード（AES-CTR） 1219
35.3.13 CRYP AES ガロア／カウンタモード（GCM） 1223
35.3.14 CRYP AES ガロアメッセージ認証コード（GMAC） 1228
35.3.15 CRYP AES CBC-MAC 付き カウンタ（CCM） 1229
35.3.16 CRYP データレジスタおよびデータスワッピング 1234
35.3.17 CRYP キーレジスタ 1238
35.3.18 CRYP 初期化ベクタレジスタ 1239
35.3.19 CRYP DMA インタフェース 1239
35.3.20 CRYP エラー管理 1241
35.4 CRYP 割り込み 1242
35.5 処理時間 1244

参考資料

35.6 CRYP レジスタ 1245
35.6.1 CRYP 制御レジスタ（CRYP_CR）. 1245
35.6.2 CRYP ステータスレジスタ（CRYP_SR）. 1247
35.6.3 CRYP データ入力レジスタ（CRYP_DIN）. 1247
35.6.4 CRYP データ出力レジスタ（CRYP_DOUT）. 1248
35.6.5 CRYP DMA 制御レジスタ（CRYP_DMACR）. 1249
35.6.6 CRYP 割り込みマスクセット／クリアレジスタ（CRYP_IMSCR）. 1249
35.6.7 CRYP マスク前割り込みステータスレジスタ（CRYP_RISR）. 1250
35.6.8 CRYP マスク済み割り込みステータスレジスタ（CRYP_MISR）. 1251
35.6.9 CRYP キーレジスタ 0L（CRYP_K0LR）. 1251
35.6.10 CRYP キーレジスタ 0R（CRYP_K0RR）. 1252
35.6.11 CRYP キーレジスタ 1L（CRYP_K1LR）. 1252
35.6.12 CRYP キーレジスタ 1R（CRYP_K1RR）. 1253
35.6.13 CRYP キーレジスタ 2L（CRYP_K2LR）. 1253
35.6.14 CRYP キーレジスタ 2R（CRYP_K2RR）. 1254
35.6.15 CRYP キーレジスタ 3L（CRYP_K3LR）. 1254
35.6.16 CRYP キーレジスタ 3R（CRYP_K3RR）. 1255
35.6.17 CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）. 1255
35.6.18 CRYP 初期化ベクタレジスタ 0R（CRYP_IV0RR）. 1256
35.6.19 CRYP 初期化ベクタレジスタ 1L（CRYP_IV1LR）. 1256
35.6.20 CRYP 初期化ベクタレジスタ 1R（CRYP_IV1RR）. 1257
35.6.21 CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）. 1257
35.6.22 CRYP コンテキストスワップ GCM レジスタ（CRYP_CSGCMxR）. 1258
35.6.23 CRYP レジスタマップ 1259

36 ハッシュプロセッサ（HASH）. 1262

36.1 概要 1262
36.2 HASH の主な機能 1262
36.3 HASH の機能詳細 1263
36.3.1 HASH ブロック図 1263
36.3.2 HASH 内部信号 1263
36.3.3 セキュアハッシュアルゴリズムについて 1264
36.3.4 メッセージデータの供給 1264
36.3.5 メッセージダイジェストの計算 1266
36.3.6 メッセージのパッディング 1267
36.3.7 HMAC 操作 1268
36.3.8 コンテキストスワッピング 1270

参考資料

36.3.9 HASH DMA インタフェース 1272
36.3.10 HASH エラー管理 1272
36.4 HASH 割り込み 1272
36.5 HASH 処理時間 1273
36.6 HASH レジスタ 1274
36.6.1 HASH 制御レジスタ（HASH_CR） 1274
36.6.2 HASH データ入力レジスタ（HASH_DIN） 1277
36.6.3 HASH スタートレジスタ（HASH_STR） 1278
36.6.4 HASH ダイジェストレジスタ（HASH_HR0..7） 1279
36.6.5 HASH 割り込み有効レジスタ（HASH_IMR） 1281
36.6.6 HASH ステータスレジスタ（HASH_SR） 1282
36.6.7 HASH コンテキストスワップレジスタ（HASH_CSRx） 1283
36.6.8 HASH レジスタマップ 1284

37 高分解能タイマ（HRTIM） 1285
37.1 概要 1285
37.2 主な特徴 1286
37.3 機能詳細 1287
37.3.1 概要 1287
37.3.2 HRTIM ピンおよび内部信号 1289
37.3.3 クロック 1290
37.3.4 タイマ A ～ E のタイミングユニット 1292
37.3.5 マスタタイマ 1309
37.3.6 セット／リセットイベントの優先順位と狭いパルスの管理 1310
37.3.7 外部イベントのグローバル調整 1311
37.3.8 タイミングユニットでの外部イベントのフィルタリング 1315
37.3.9 遅延保護 1320
37.3.10 レジスタのプリロードおよび更新管理 1326
37.3.11 複数タイマ内またはそれらをまたがったイベントの伝播 1329
37.3.12 出力管理 1333
37.3.13 バーストモードコントローラ 1335
37.3.14 チョッパ 1344
37.3.15 フォールト保護 1345
37.3.16 補助出力 1348
37.3.17 他のタイマまたは HRTIM インスタンスとの HRTIM の同期 1351
37.3.18 ADC トリガ 1354
37.3.19 DAC トリガ 1355

37.3.20 HRTIM 割り込み 1356
37.3.21 DMA 1357
37.3.22 HRTIM の初期化 1361
37.3.23 デバッグ 1362
37.4 アプリケーションの使用事例 1363
37.4.1 降圧コンバータ 1363
37.4.2 同期整流式の降圧コンバータ 1364
37.4.3 多相コンバータ 1365
37.4.4 遷移モード力率補正回路 1367
37.5 HRTIM レジスタ 1369
37.5.1 HRTIM マスタタイマ制御レジスタ（HRTIM_MCR） 1369
37.5.2 HRTIM マスタタイマ割り込みステータスレジスタ（HRTIM_MISR） 1372
37.5.3 HRTIM マスタタイマ割り込みクリアレジスタ（HRTIM_MICR） 1373
37.5.4 HRTIM マスタタイマ DMA ／割り込みイネーブルレジスタ（HRTIM_MDIER） 1374
37.5.5 HRTIM マスタタイマカウンタレジスタ（HRTIM_MCNTR） 1376
37.5.6 HRTIM マスタタイマ周期レジスタ（HRTIM_MPER） 1376
37.5.7 HRTIM マスタタイマ繰り返しレジスタ（HRTIM_MREP） 1377
37.5.8 HRTIM マスタタイマ比較 1 レジスタ（HRTIM_MCMP1R） 1377
37.5.9 HRTIM マスタタイマ比較 2 レジスタ（HRTIM_MCMP2R） 1378
37.5.10 HRTIM マスタタイマ比較 3 レジスタ（HRTIM_MCMP3R） 1378
37.5.11 HRTIM マスタタイマ比較 4 レジスタ（HRTIM_MCMP4R） 1379
37.5.12 HRTIM タイマ x 制御レジスタ（HRTIM_TIMxCR） 1380
37.5.13 HRTIM タイマ x 割り込みステータスレジスタ（HRTIM_TIMxISR） 1384
37.5.14 HRTIM タイマ x 割り込みクリアレジスタ（HRTIM_TIMxICR） 1386
37.5.15 HRTIM タイマ x DMA ／割り込みイネーブルレジスタ（HRTIM_TIMxDIER） 1387
37.5.16 HRTIM タイマ x カウンタレジスタ（HRTIM_CNTxR） 1390
37.5.17 HRTIM タイマ x 周期レジスタ（HRTIM_PERxR） 1390
37.5.18 HRTIM タイマ x 繰り返しレジスタ（HRTIM_REPxR） 1391
37.5.19 HRTIM タイマ x 比較 1 レジスタ（HRTIM_CMP1xR） 1391
37.5.20 HRTIM タイマ x 比較 1 合成レジスタ（HRTIM_CMP1CxR） 1392
37.5.21 HRTIM タイマ x 比較 2 レジスタ（HRTIM_CMP2xR） 1392
37.5.22 HRTIM タイマ x 比較 3 レジスタ（HRTIM_CMP3xR） 1393
37.5.23 HRTIM タイマ x 比較 4 レジスタ（HRTIM_CMP4xR） 1393
37.5.24 HRTIM タイマ x キャプチャ 1 レジスタ（HRTIM_CPT1xR） 1394
37.5.25 HRTIM タイマ x キャプチャ 2 レジスタ（HRTIM_CPT2xR） 1394
37.5.26 HRTIM タイマ x デッドタイムレジスタ（HRTIM_DTxR） 1395

37.5.27 HRTIM タイマ x 出力 1 セットレジスタ（HRTIM_SETx1R）.....1397
37.5.28 HRTIM タイマ x 出力 1 リセットレジスタ（HRTIM_RSTx1R）.....1399
37.5.29 HRTIM タイマ x 出力 2 セットレジスタ（HRTIM_SETx2R）.....1399
37.5.30 HRTIM タイマ x 出力 2 リセットレジスタ（HRTIM_RSTx2R）.....1400
37.5.31 HRTIM タイマ x 外部イベントフィルタリングレジスタ 1（HRTIM_EEFxR1）.....1401
37.5.32 HRTIM タイマ x 外部イベントフィルタリングレジスタ 2（HRTIM_EEFxR2）.....1403
37.5.33 HRTIM タイマ x リセットレジスタ（HRTIM_RSTxR）.....1404
37.5.34 HRTIM タイマ x チョッパレジスタ（HRTIM_CHPxR）.....1407
37.5.35 HRTIM タイマ x キャプチャ 1 制御レジスタ（HRTIM_CPT1xCR）.....1408
37.5.36 HRTIM タイマ x キャプチャ 2 制御レジスタ（HRTIM_CPT2xCR）.....1409
37.5.37 HRTIM タイマ x 出力レジスタ（HRTIM_OUTxR）.....1412
37.5.38 HRTIM タイマ x フォールトレジスタ（HRTIM_FLTxR）.....1415
37.5.39 HRTIM 制御レジスタ 1（HRTIM_CR1）.....1416
37.5.40 HRTIM 制御レジスタ 2（HRTIM_CR2）.....1418
37.5.41 HRTIM 割り込みステータスレジスタ（HRTIM_ISR）.....1419
37.5.42 HRTIM 割り込みクリアスレジスタ（HRTIM_ICR）.....1420
37.5.43 HRTIM 割り込み有効レジスタ（HRTIM_IER）.....1421
37.5.44 HRTIM 出力有効レジスタ（HRTIM_OENR）.....1422
37.5.45 HRTIM 出力無効レジスタ（HRTIM_ODISR）.....1423
37.5.46 HRTIM 出力無効ステータスレジスタ（HRTIM_ODSR）.....1424
37.5.47 HRTIM バーストモード制御レジスタ（HRTIM_BMCR）.....1425
37.5.48 HRTIM バーストモードトリガレジスタ（HRTIM_BMTRGR）.....1427
37.5.49 HRTIM バーストモード比較レジスタ（HRTIM_BMCMPR）.....1429
37.5.50 HRTIM バーストモード周期レジスタ（HRTIM_BMPER）.....1429
37.5.51 HRTIM タイマ外部イベント制御レジスタ 1（HRTIM_EECR1）.....1430
37.5.52 HRTIM タイマ外部イベント制御レジスタ 2（HRTIM_EECR2）.....1432
37.5.53 HRTIM タイマ外部イベント制御レジスタ 3（HRTIM_EECR3）.....1433
37.5.54 HRTIM ADC トリガ 1 レジスタ（HRTIM_ADC1R）.....1434
37.5.55 HRTIM ADC トリガ 2 レジスタ（HRTIM_ADC2R）.....1435
37.5.56 HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R）.....1436
37.5.57 HRTIM ADC トリガ 4 レジスタ（HRTIM_ADC4R）.....1438
37.5.58 HRTIM フォールト入力レジスタ 1（HRTIM_FLTINR1）.....1440
37.5.59 HRTIM フォールト入力レジスタ 2（HRTIM_FLTINR2）.....1442
37.5.60 HRTIM バースト DMA マスタタイマ更新レジスタ（HRTIM_BDMUPR）...1444
37.5.61 HRTIM バースト DMA タイマ x 更新レジスタ（HRTIM_BDTxUPR）.....1445
37.5.62 HRTIM バースト DMA データレジスタ（HRTIM_BDMADR）.....1446
37.5.63 HRTIM レジスタマップ.....1447

参考資料

38 高機能制御タイマ（TIM1/TIM8）. 1456

38.1 TIM1/TIM8 の概要 1456

38.2 TIM1/TIM8 の主な特長 1456

38.3 TIM1/TIM8 機能詳細 1458

38.3.1 タイムベースユニット 1458

38.3.2 カウンタモード 1460

38.3.3 繰り返しカウンタ 1471

38.3.4 外部トリガ入力 1473

38.3.5 クロック選択 1474

38.3.6 キャプチャ／比較チャネル 1478

38.3.7 入力キャプチャモード 1481

38.3.8 PWM 入力モード 1482

38.3.9 強制出力モード 1483

38.3.10 出力比較モード 1483

38.3.11 PWM モード 1484

38.3.12 非対称 PWM モード 1488

38.3.13 組み合わせ PWM モード 1488

38.3.14 組み合わせ 3 相 PWM モード 1490

38.3.15 相補出力とデッドタイム挿入 1491

38.3.16 ブレーク機能の使用 1493

38.3.17 双方向ブレーク入力 1499

38.3.18 外部イベントによる OCxREF 信号のクリア 1500

38.3.19 6 ステップ PWM 生成 1501

38.3.20 ワンパルスモード 1502

38.3.21 再トリガ可能なワンパルスモード（OPM） 1503

38.3.22 エンコーダインタフェースモード 1504

38.3.23 UIF ビットの再配置 1506

38.3.24 タイマ入力 XOR 機能 1507

38.3.25 ホールセンサとのインタフェース 1507

38.3.26 タイマの同期 1510

38.3.27 ADC の同期 1514

38.3.28 DMA バーストモード 1514

38.3.29 デバッグモード 1515

38.4 TIM1/TIM8 レジスタ 1516

38.4.1 TIM1/TIM8 制御レジスタ 1（TIMx_CR1） 1516

38.4.2 TIM1/TIM8 制御レジスタ 2（TIMx_CR2） 1518

38.4.3 TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR） 1520

38.4.4 TIM1/TIM8 DMA ／割り込み有効レジスタ（TIMx_DIER）.....1523
38.4.5 TIM1/TIM8 ステータスレジスタ（TIMx_SR）.....1524
38.4.6 TIM1/TIM8 イベント生成レジスタ（TIMx_EGR）.....1526
38.4.7 TIM1/TIM8 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）.....1527
38.4.8 TIM1/TIM8 キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）.....1531
38.4.9 TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）.....1533
38.4.10 TIM1/TIM8 カウンタ（TIMx_CNT）.....1537
38.4.11 TIM1/TIM8 プリスケーラ（TIMx_PSC）.....1537
38.4.12 TIM1/TIM8 自動再ロードレジスタ（TIMx_ARR）.....1538
38.4.13 TIM1/TIM8 繰り返しカウンタレジスタ（TIMx_RCR）.....1538
38.4.14 TIM1/TIM8 キャプチャ／比較レジスタ 1 （TIMx_CCR1）.....1539
38.4.15 TIM1/TIM8 キャプチャ／比較レジスタ 2 （TIMx_CCR2）.....1539
38.4.16 TIM1/TIM8 キャプチャ／比較レジスタ 3 （TIMx_CCR3）.....1540
38.4.17 TIM1/TIM8 キャプチャ／比較レジスタ 4 （TIMx_CCR4）.....1540
38.4.18 TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）.....1541
38.4.19 TIM1/TIM8 DMA 制御レジスタ（TIMx_DCR）.....1545
38.4.20 TIM1/TIM8 完全転送用の DMA アドレス（TIMx_DMAR）.....1546
38.4.21 TIM1/TIM8 キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）.....1546
38.4.22 TIM1/TIM8 キャプチャ／比較レジスタ 5 （TIMx_CCR5）.....1547
38.4.23 TIM1/TIM8 キャプチャ／比較レジスタ 6 （TIMx_CCR6）.....1548
38.4.24 TIM1 オルタネート機能オプションレジスタ 1 （TIM1_AF1）.....1548
38.4.25 TIM1 オルタネート機能レジスタ 2 （TIM1_AF2）.....1550
38.4.26 TIM8 オルタネート機能オプションレジスタ 1 （TIM8_AF1）.....1551
38.4.27 TIM8 オルタネート機能オプションレジスタ 2 （TIM8_AF2）.....1553
38.4.28 TIM1 タイマ入力選択レジスタ（TIM1_TISEL）.....1555
38.4.29 TIM8 タイマ入力選択レジスタ（TIM8_TISEL）.....1556
38.4.30 TIM1 レジスタマップ 1557
38.4.31 TIM8 レジスタマップ 1560

39 汎用タイマ（TIM2/TIM3/TIM4/TIM5）..... 1563

39.1 TIM2/TIM3/TIM4/TIM5 の概要 1563
39.2 TIM2/TIM3/TIM4/TIM5 の主な特長 1563
39.3 TIM2/TIM3/TIM4/TIM5 機能詳細 1565
39.3.1 タイムベースユニット 1565
39.3.2 カウンタモード 1567
39.3.3 クロック選択 1577
39.3.4 キャプチャ／比較チャネル 1581
39.3.5 入力キャプチャモード 1583

参考資料

39.3.6 PWM 入力モード 1584
39.3.7 強制出力モード 1585
39.3.8 出力比較モード 1585
39.3.9 PWM モード 1586
39.3.10 非対称 PWM モード 1590
39.3.11 組み合わせ PWM モード 1590
39.3.12 外部イベントによる OCxREF 信号のクリア 1591
39.3.13 ワンパルスモード 1593
39.3.14 再トリガ可能なワンパルスモード（OPM） 1594
39.3.15 エンコーダインタフェースモード 1595
39.3.16 UIF ビットの再配置 1597
39.3.17 タイマ入力 XOR 機能 1597
39.3.18 タイマと外部トリガの同期 1598
39.3.19 タイマの同期 1601
39.3.20 DMA バーストモード 1606
39.3.21 デバッグモード 1607
39.4 TIM2/TIM3/TIM4/TIM5 レジスタ 1608
39.4.1 TIMx 制御レジスタ 1（TIMx_CR1） 1608
39.4.2 TIMx 制御レジスタ 2（TIMx_CR2） 1610
39.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR） 1611
39.4.4 TIMx DMA ／割り込み有効レジスタ（TIMx_DIER） 1614
39.4.5 TIMx ステータスレジスタ（TIMx_SR） 1615
39.4.6 TIMx イベント生成レジスタ（TIMx_EGR） 1617
39.4.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1） 1618
39.4.8 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2） 1621
39.4.9 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER） 1623
39.4.10 TIMx カウンタ（TIMx_CNT） 1625
39.4.11 TIMx プリスケーラ（TIMx_PSC） 1625
39.4.12 TIMx 自動再ロードレジスタ（TIMx_ARR） 1626
39.4.13 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCR1） 1626
39.4.14 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCR2） 1627
39.4.15 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCR3） 1627
39.4.16 TIMx キャプチャ／比較モードレジスタ 4（TIMx_CCR4） 1628
39.4.17 TIMx DMA 制御レジスタ（TIMx_DCR） 1629
39.4.18 完全転送の TIMx DMA アドレス（TIMx_DMAR） 1629
39.4.19 TIM2 オルタネート機能オプションレジスタ 1 （TIM2_AF1） 1630
39.4.20 TIM3 オルタネート機能オプションレジスタ 1 （TIM3_AF1） 1630
39.4.21 TIM5 オルタネート機能オプションレジスタ 1 （TIM5_AF1） 1631

39.4.22 TIM2 タイマ入力選択レジスタ（TIM2_TISEL）.....1631
39.4.23 TIM3 タイマ入力選択レジスタ（TIM3_TISEL）.....1632
39.4.24 TIM5 タイマ入力選択レジスタ（TIM5_TISEL）.....1633
39.4.25 TIMx レジスタマップ 1634

40 汎用タイマ（TIM12/TIM13/TIM14）..... 1637

40.1 TIM12/TIM13/TIM14 の概要 1637
40.2 TIM12/TIM13/TIM14 の主な特長 1637
40.2.1 TIM12 の主な特長 1637
40.2.2 TIM13/TIM14 の主な特長 1638
40.3 TIM12/TIM13/TIM14 機能詳細 1640
40.3.1 タイムベースユニット 1640
40.3.2 カウンタモード 1642
40.3.3 クロック選択 1645
40.3.4 キャプチャ／比較チャネル 1647
40.3.5 入力キャプチャモード 1649
40.3.6 PWM 入力モード（TIM12 の場合のみ）.....1650
40.3.7 強制出力モード 1651
40.3.8 出力比較モード 1652
40.3.9 PWM モード 1653
40.3.10 組み合わせ PWM モード（TIM12 のみ）.....1654
40.3.11 ワンパルスモード 1655
40.3.12 再トリガ可能なワンパルスモード（OPM）（TIM12 のみ）.....1657
40.3.13 UIF ビットの再配置 1658
40.3.14 タイマ入力 XOR 機能 1658
40.3.15 TIM12 外部トリガ同期 1658
40.3.16 スレーブモード - リセットモードとトリガモードの組み合わせ 1661
40.3.17 タイマの同期（TIM12）.....1661
40.3.18 デバッグモード 1661
40.4 TIM12 レジスタ 1662
40.4.1 TIM12 制御レジスタ 1（TIMx_CR1）.....1662
40.4.2 TIM12 スレーブモード制御レジスタ（TIMx_SMCR）.....1663
40.4.3 TIM12 割り込み有効レジスタ（TIMx_DIER）.....1665
40.4.4 TIM12 ステータスレジスタ（TIMx_SR）.....1665
40.4.5 TIM12 イベント生成レジスタ（TIMx_EGR）.....1666
40.4.6 TIM12 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）.....1667
40.4.7 TIM12 キャプチャ／比較有効レジスタ（TIMx_CCER）.....1671
40.4.8 TIM12 カウンタ（TIMx_CNT）.....1673

参考資料

40.4.9 TIM12 プリスケーラ（TIMx_PSC）.....1673
40.4.10 TIM12 自動再ロードレジスタ（TIMx_ARR）.....1673
40.4.11 TIM12 キャプチャ／比較レジスタ 1 （TIMx_CCR1）.....1674
40.4.12 TIM12 キャプチャ／比較モードレジスタ 2（TIMx_CCR2）.....1674
40.4.13 TIM12 タイマ入力選択レジスタ（TIM12_TISEL）.....1675
40.4.14 TIM12 レジスタマップ 1676
40.5 TIM13/TIM14 レジスタ 1678
40.5.1 TIM13/TIM14 制御レジスタ 1（TIMx_CR1）.....1678
40.5.2 TIM13/TIM14 割り込み有効レジスタ（TIMx_DIER）.....1679
40.5.3 TIM13/TIM14 ステータスレジスタ（TIMx_SR）.....1680
40.5.4 TIM13/TIM14 イベント生成レジスタ（TIMx_EGR）.....1681
40.5.5 TIM13/TIM14 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）.....1681
40.5.6 TIM13/TIM14 キャプチャ／比較有効レジスタ（TIMx_CCER）.....1684
40.5.7 TIM13/TIM14 カウンタ（TIMx_CNT）.....1685
40.5.8 TIM13/TIM14 プリスケーラ（TIMx_PSC）.....1685
40.5.9 TIM13/TIM14 自動再ロードレジスタ（TIMx_ARR）.....1685
40.5.10 TIM13/TIM14 キャプチャ／比較レジスタ 1（TIMx_CCR1）.....1686
40.5.11 TIM13 タイマ入力選択レジスタ（TIM13_TISEL）.....1686
40.5.12 TIM14 タイマ入力選択レジスタ（TIM14_TISEL）.....1686
40.5.13 TIM13/TIM14 レジスタマップ 1687

41 汎用タイマ（TIM15/TIM16/TIM17） 1689
41.1 TIM15/TIM16/TIM17 の概要 1689
41.2 TIM15 の主な機能 1689
41.3 TIM16/TIM17 の主な特長 1690
41.4 TIM15/TIM16/TIM17 機能詳細 1693
41.4.1 タイムベースユニット 1693
41.4.2 カウンタモード 1695
41.4.3 繰り返しカウンタ 1699
41.4.4 クロック選択 1700
41.4.5 キャプチャ／比較チャネル 1702
41.4.6 入力キャプチャモード 1705
41.4.7 PWM 入力モード（TIM15 の場合のみ）.....1706
41.4.8 強制出力モード 1707
41.4.9 出力比較モード 1707
41.4.10 PWM モード 1708
41.4.11 組み合わせ PWM モード（TIM15 のみ）.....1710

41.4.12 相補出力とデッドタイム挿入 1711
41.4.13 ブレーク機能の使用 1713
41.4.14 ワンパルスモード 1718
41.4.15 UIF ビットの再配置 1719
41.4.16 タイマ入力 XOR 機能（TIM15 のみ） 1720
41.4.17 外部トリガ同期（TIM15 のみ） 1721
41.4.18 スレーブモード - リセットモードとトリガモードの組み合わせ 1723
41.4.19 DMA バーストモード 1723
41.4.20 タイマ同期（TIM15） 1725
41.4.21 デバッグモード 1725
41.5 TIM15 レジスタ 1726
41.5.1 TIM15 制御レジスタ 1 （TIM15_CR1） 1726
41.5.2 TIM15 制御レジスタ 2 （TIM15_CR2） 1727
41.5.3 TIM15 のスレーブモード制御レジスタ（TIM15_SMCR） 1729
41.5.4 TIM15 DMA ／割り込み有効レジスタ（TIM15_DIER） 1730
41.5.5 TIM15 ステータスレジスタ（TIM15_SR） 1731
41.5.6 TIM15 のイベント生成レジスタ（TIM15_EGR） 1733
41.5.7 TIM15 のキャプチャ／比較モードレジスタ 1（TIM15_CCMR1） 1734
41.5.8 TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER） 1737
41.5.9 TIM15 のカウンタ（TIM15_CNT） 1740
41.5.10 TIM15 のプリスケーラ（TIM15_PSC） 1740
41.5.11 TIM15 の自動再ロードレジスタ（TIM15_ARR） 1740
41.5.12 TIM15 繰り返しカウンタレジスタ（TIM15_RCR） 1741
41.5.13 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1） 1741
41.5.14 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2） 1742
41.5.15 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR） 1742
41.5.16 TIM15 DMA 制御レジスタ（TIM15_DCR） 1745
41.5.17 完全転送の TIM15 DMA アドレス（TIM15_DMAR） 1745
41.5.18 TIM15 オルタネートレジスタ 1 （TIM15_AF1） 1746
41.5.19 TIM15 入力選択レジスタ（TIM15_TISEL） 1747
41.5.20 TIM15 レジスタマップ 1748
41.6 TIM16/TIM17 レジスタ 1750
41.6.1 TIM16/TIM17 制御レジスタ 1（TIMx_CR1） 1750
41.6.2 TIM16/TIM17 制御レジスタ 2 （TIMx_CR2） 1751
41.6.3 TIM16/TIM17 DMA ／割り込み有効レジスタ（TIMx_DIER） 1752
41.6.4 TIM16/TIM17 ステータスレジスタ（TIMx_SR） 1753
41.6.5 TIM16/TIM17 イベント生成レジスタ（TIMx_EGR） 1754
41.6.6 TIM16/TIM17 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1） 1755

41.6.7 TIM16/TIM17 キャプチャ／比較有効レジスタ（TIMx_CCER）.....1758
41.6.8 TIM16/TIM17 カウンタ（TIMx_CNT）.....1761
41.6.9 TIM16/TIM17 プリスケーラ（TIMx_PSC）.....1761
41.6.10 TIM16/TIM17 自動再ロードレジスタ（TIMx_ARR）.....1761
41.6.11 TIM16/TIM17 繰り返しカウンタレジスタ（TIMx_RCR）.....1762
41.6.12 TIM16/TIM17 キャプチャ／比較レジスタ 1（TIMx_CCR1）.....1762
41.6.13 TIM16/TIM17 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）.....1763
41.6.14 TIM16/TIM17 DMA 制御レジスタ（TIMx_DCR）.....1766
41.6.15 完全転送の TIM16/TIM17 DMA アドレス（TIMx_DMAR）.....1766
41.6.16 TIM16 オルタネート機能レジスタ 1（TIM16_AF1）.....1767
41.6.17 TIM16 入力選択レジスタ（TIM16_TISEL）.....1768
41.6.18 TIM17 オルタネート機能レジスタ 1（TIM17_AF1）.....1769
41.6.19 TIM17 入力選択レジスタ（TIM17_TISEL）.....1770
41.6.20 TIM16/TIM17 レジスタマップ.....1771

42 基本タイマ（TIM6/TIM7）.....1773

42.1 TIM6/TIM7 の概要.....1773
42.2 TIM6/TIM7 の主な機能.....1773
42.3 TIM6/TIM7 の機能詳細.....1774
42.3.1 タイムベースユニット.....1774
42.3.2 カウントモード.....1776
42.3.3 UIF ビットの再配置.....1779
42.3.4 クロックソース.....1779
42.3.5 デバッグモード.....1780
42.4 TIM6/TIM7 レジスタ.....1780
42.4.1 TIM6/TIM7 制御レジスタ 1（TIMx_CR1）.....1781
42.4.2 TIM6/TIM7 制御レジスタ 2（TIMx_CR2）.....1782
42.4.3 TIM6/TIM7 DMA ／割り込み有効レジスタ（TIMx_DIER）.....1782
42.4.4 TIM6/TIM7 のステータスレジスタ（TIMx_SR）.....1783
42.4.5 TIM6/TIM7 のイベント生成レジスタ（TIMx_EGR）.....1783
42.4.6 TIM6/TIM7 のカウンタ（TIMx_CNT）.....1784
42.4.7 TIM6/TIM7 プリスケーラ（TIMx_PSC）.....1784
42.4.8 TIM6/TIM7 の自動再ロードレジスタ（TIMx_ARR）.....1784
42.4.9 TIM6/TIM7 レジスタマップ.....1785

参考資料

43 低電力タイマ（LPTIM）. 1786

43.1 概要 1786

43.2 LPTIM の主な機能 1786

43.3 LPTIM の実装 1786

43.4 LPTIM の機能詳細 1787

43.4.1 LPTIM ブロック図 1787

43.4.2 LPTIM ピンおよび内部信号 1789

43.4.3 LPTIM 入力およびトリガマッピング 1789

43.4.4 LPTIM のリセットとクロック 1792

43.4.5 グリッチフィルタ 1792

43.4.6 プリスケーラ 1793

43.4.7 トリガマルチプレクサ 1793

43.4.8 動作モード 1794

43.4.9 タイムアウト機能 1796

43.4.10 波形生成 1796

43.4.11 レジスタの更新 1797

43.4.12 カウンタモード 1798

43.4.13 タイマ有効 1798

43.4.14 タイマカウンタのリセット 1799

43.4.15 エンコーダモード 1799

43.5 LPTIM 割り込み 1801

43.6 LPTIM レジスタ 1802

43.6.1 LPTIM 割り込みおよびステータスレジスタ（LPTIM_ISR）. 1802

43.6.2 LPTIM 割り込みクリアレジスタ（LPTIM_ICR）. 1803

43.6.3 LPTIM 割り込み有効レジスタ（LPTIM_IER）. 1804

43.6.4 LPTIM 設定レジスタ（LPTIM_CFGR）. 1805

43.6.5 LPTIM 制御レジスタ（LPTIM_CR） 1808

43.6.6 LPTIM 比較レジスタ（LPTIM_CMP）. 1810

43.6.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）. 1810

43.6.8 LPTIM カウンタレジスタ（LPTIM_CNT）. 1811

43.6.9 LPTIM 設定レジスタ 2 （LPTIMx_CFGR2、x=1、2） 1811

43.6.10 LPTIM3 設定レジスタ 2（LPTIM3_CFGR2） 1812

43.6.11 LPTIM レジスタマップ 1813

参考資料

44 システムウィンドウ型ウォッチドッグ（WWDG）. 1814

44.1 概要 1814

44.2 WWDG の主な機能 1814

44.3 WWDG の機能説明 1814

44.3.1 WWDG ブロック図 1815

44.3.2 WWDG 内部信号 1815

44.3.3 ウォッチドッグの有効化 1815

44.3.4 ダウンカウンタの制御 1816

44.3.5 高度なウォッチドッグ割り込み機能 1816

44.3.6 ウォッチドッグタイムアウトをプログラムする方法 1816

44.3.7 デバッグモード 1817

44.4 .WWDG レジスタ 1818

44.4.1 制御レジスタ （WWDG_CR） 1818

44.4.2 設定レジスタ（WWDG_CFR） 1819

44.4.3 ステータスレジスタ（WWDG_SR） 1820

44.4.4 WWDG レジスタマップ 1820

45 独立型ウォッチドッグ（IWDG）. 1821

45.1 概要 1821

45.2 IWDG の主な機能 1821

45.3 IWDG の機能説明 1821

45.3.1 IWDG ブロック図 1821

45.3.2 IWDG 内部信号 1822

45.3.3 ウィンドウオプション 1822

45.3.4 ハードウェアウォッチドッグ 1823

45.3.5 低電力モードでの取り扱い 1823

45.3.6 STOP および STANDBY モードでの動作 1823

45.3.7 レジスタのアクセス保護 1823

45.3.8 デバッグモード 1823

45.4 IWDG レジスタ 1824

45.4.1 キーレジスタ（IWDG_KR） 1824

45.4.2 プリスケーラレジスタ（IWDG_PR） 1825

45.4.3 再ロードレジスタ（IWDG_RLR） 1826

45.4.4 ステータスレジスタ（IWDG_SR） 1827

45.4.5 ウィンドウレジスタ（IWDG_WINR） 1828

45.4.6 IWDG レジスタマップ 1829

参考資料

46 リアルタイムクロック（RTC）. 1830

46.1 概要 1830

46.2 RTC の主な機能 1830

46.3 RTC の機能説明 1831

46.3.1 RTC ブロック図 1831

46.3.2 RTC ピンおよび内部信号 1834

46.3.3 RTC によって制御される GPIO 1834

46.3.4 クロックとプリスケーラ 1836

46.3.5 リアルタイムクロックとカレンダ 1837

46.3.6 プログラム可能なアラーム 1837

46.3.7 周期的自動ウェイクアップ 1838

46.3.8 RTC の初期化と設定 1838

46.3.9 カレンダの読み出し 1840

46.3.10 RTC のリセット 1841

46.3.11 RTC の同期 1841

46.3.12 RTC リファレンスクロック検出 1842

46.3.13 RTC の高精度デジタル較正 1843

46.3.14 タイムスタンプ機能 1845

46.3.15 タンパ検出 1845

46.3.16 較正クロック出力 1847

46.3.17 アラーム出力 1848

46.4 RTC 低電力モード 1848

46.5 RTC 割り込み 1848

46.6 RTC レジスタ 1850

46.6.1 RTC 時刻レジスタ（RTC_TR）. 1850

46.6.2 RTC 日付レジスタ（RTC_DR）. 1851

46.6.3 RTC 制御レジスタ （RTC_CR）. 1852

46.6.4 RTC 初期化とステータスレジスタ（RTC_ISR）. 1855

46.6.5 RTC プリスケーラレジスタ（RTC_PRER）. 1858

46.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）. 1859

46.6.7 RTC アラーム A レジスタ（RTC_ALRMAR）. 1860

46.6.8 RTC アラーム B レジスタ（RTC_ALRMBR）. 1861

46.6.9 RTC 書き込み保護レジスタ（RTC_WPR）. 1862

46.6.10 RTC サブセカンドレジスタ（RTC_SSR）. 1862

46.6.11 RTC シフト制御レジスタ（RTC_SHIFTR）. 1863

46.6.12 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）. 1864

46.6.13 RTC タイムスタンプ日付レジスタ（RTC_TSDR）. 1865

参考資料

46.6.14 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）.....1866
46.6.15 RTC 較正レジスタ（RTC_CALR）.....1867
46.6.16 RTC タンパ設定レジスタ（RTC_TAMPCR）.....1868
46.6.17 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）.....1871
46.6.18 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）.....1872
46.6.19 RTC オプションレジスタ （RTC_OR）.....1873
46.6.20 RTC バックアップレジスタ（RTC_BKPxR）.....1873
46.6.21 RTC レジスタマップ.....1874

47 I2C（Inter-integrated circuit）インタフェース 1876

47.1 概要 1876
47.2 I2C の主な機能 1876
47.3 I2C の実装 1877
47.4 I2C の機能詳細 1877
47.4.1 I2C ブロック図 1878
47.4.2 I2C クロックの要件 1879
47.4.3 モード選択 1879
47.4.4 I2C の初期化 1881
47.4.5 ソフトウェアリセット 1885
47.4.6 データ転送 1886
47.4.7 I2C スレーブモード 1888
47.4.8 I2C マスタモード 1897
47.4.9 I2C_TIMINGR レジスタの設定例 1909
47.4.10 SMBus 固有の機能 1910
47.4.11 SMBus 初期化 1913
47.4.12 SMBus：I2C_TIMEOUTR レジスタの設定例 1915
47.4.13 SMBus スレーブモード 1916
47.4.14 アドレス一致時に STOP モードからウェイクアップ 1924
47.4.15 エラー条件 1924
47.4.16 DMA リクエスト 1926
47.4.17 デバッグモード 1927
47.5 I2C 低電力モード 1927
47.6 I2C 割り込み 1928
47.7 I2C レジスタ 1929
47.7.1 制御レジスタ 1（I2C_CR1）.....1929
47.7.2 制御レジスタ 2（I2C_CR2）.....1932
47.7.3 Own Address 1 レジスタ（I2C_OAR1）.....1935

参考資料

47.7.4 Own Address 2 レジスタ（I2C_OAR2）. 1936
47.7.5 タイミングレジスタ（I2C_TIMINGR）. 1937
47.7.6 タイムアウトレジスタ（I2C_TIMEOUTR）. 1938
47.7.7 割り込みおよびステータスレジスタ（I2C_ISR）. 1939
47.7.8 割り込みクリアレジスタ（I2C_ICR）. 1941
47.7.9 PEC レジスタ（I2C_PECR）. 1942
47.7.10 受信データレジスタ（I2C_RXDR）. 1943
47.7.11 送信データレジスタ（I2C_TXDR）. 1943
47.7.12 I2C レジスタマップ 1944

48 USART（Universal synchronous asynchronous receiver transmitter）. . . . 1946

48.1 USART の概要 1946
48.2 USART の主な機能 1947
48.3 USART の拡張機能 1948
48.4 USART の実装 1948
48.5 USART の機能詳細 1949
48.5.1 USART のブロック図 1949
48.5.2 USART 信号 1950
48.5.3 USART キャラクタの説明 1951
48.5.4 USART の FIFO と閾値 1953
48.5.5 USART トランスミッタ 1953
48.5.6 USART レシーバ 1957
48.5.7 USART ボーレート生成 1964
48.5.8 クロック偏差に対する USART レシーバの許容誤差 1965
48.5.9 USART 自動ボーレート検出 1966
48.5.10 USART マルチプロセッサ通信 1968
48.5.11 USART Modbus 通信 1970
48.5.12 USART パリティ制御 1970
48.5.13 USART LIN（Local Interconnection Network）モード 1971
48.5.14 USART 同期モード 1973
48.5.15 USART 単線半二重通信 1977
48.5.16 USART レシーバタイムアウト 1977
48.5.17 USART スマートカードモード 1978
48.5.18 USART IrDA SIR ENDEC ブロック 1982
48.5.19 USART および DMA を使用した連続通信 1984
48.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効 1986
48.5.21 USART 低電力管理 1988

参考資料

48.6 USART 割り込み 1991
48.7 USART レジスタ 1994
48.7.1 USART 制御レジスタ 1（USART_CR1） 1994
48.7.2 USART 制御レジスタ 2（USART_CR2） 1998
48.7.3 USART 制御レジスタ 3（USART_CR3） 2002
48.7.4 USART ボーレートレジスタ（USART_BRR） 2006
48.7.5 USART ガード時間およびプリスケーラレジスタ（USART_GTPR） 2007
48.7.6 USART レシーバタイムアウトレジスタ（USART_RTOR） 2008
48.7.7 USART リクエストレジスタ（USART_RQR） 2009
48.7.8 USART 割り込みおよびステータスレジスタ（USART_ISR） 2010
48.7.9 USART 割り込みフラグクリアレジスタ（USART_ICR） 2016
48.7.10 USART レシーバデータレジスタ（USART_RDR） 2017
48.7.11 USART トランスミッタデータレジスタ（USART_TDR） 2018
48.7.12 USART プリスケーラレジスタ（USART_PRESC） 2018
48.7.13 USART レジスタマップ 2020

49 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART） 2021
49.1 LPUART の概要 2021
49.2 LPUART の主な機能 2022
49.3 LPUART 機能の詳細 2023
49.3.1 LPUART ブロック図 2023
49.3.2 LPUART 信号 2024
49.3.3 LPUART キャラクタの説明 2024
49.3.4 LPUART の FIFO と閾値 2026
49.3.5 LPUART トランスミッタ 2026
49.3.6 LPUART レシーバ 2029
49.3.7 LPUART ボーレート生成 2033
49.3.8 クロック偏差に対する LPUART レシーバの許容誤差 2035
49.3.9 LPUART マルチプロセッサ通信 2035
49.3.10 LPUART パリティ制御 2038
49.3.11 LPUART 単線半二重通信 2039
49.3.12 DMA および LPUART を使用した連続通信 2039
49.3.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効 2042
49.3.14 LPUART 低電力管理 2044
49.4 LPUART 割り込み 2047
49.5 LPUART レジスタ 2049
49.5.1 制御レジスタ 1（LPUART_CR1） 2049

49.5.2 制御レジスタ 2（LPUART_CR2）. 2052
49.5.3 制御レジスタ 3（LPUART_CR3）. 2054
49.5.4 ボーレートレジスタ（LPUART_BRR）. 2057
49.5.5 リクエストレジスタ（LPUART_RQR）. 2058
49.5.6 割り込みおよびステータスレジスタ（LPUART_ISR）. 2058
49.5.7 割り込みフラグクリアレジスタ（LPUART_ICR）. 2063
49.5.8 受信データレジスタ（LPUART_RDR）. 2064
49.5.9 送信データレジスタ（LPUART_TDR）. 2064
49.5.10 プリスケーラレジスタ（LPUART_PRESC）. 2065
49.5.11 LPUART レジスタマップ 2066

50 シリアルペリフェラルインタフェース（SPI）. 2067

50.1 概要 2067
50.2 SPI の主な機能 2067
50.3 SPI の実装 2068
50.4 SPI の機能説明 2068
50.4.1 SPI ブロック図 2068
50.4.2 SPI 信号 2070
50.4.3 SPI 通信の一般的側面 2070
50.4.4 マスタとスレーブの 1 対 1 の通信 2070
50.4.5 標準マルチスレーブ通信 kusa 2073
50.4.6 マルチマスタ通信 2076
50.4.7 スレーブ選択（SS）ピンの管理 2076
50.4.8 通信フォーマット 2080
50.4.9 SPI の設定 2082
50.4.10 SPI を有効にする手順 2083
50.4.11 SPI のデータ送受信手順 2083
50.4.12 SPI を無効にする手順 2087
50.4.13 データパッキング 2088
50.4.14 DMA（ダイレクトメモリアクセス）を使用する通信 2089
50.5 SPI の特定モード及び制御 2091
50.5.1 TI モード 2091
50.5.2 SPI エラーフラグ 2091
50.5.3 CRC 計算 2093
50.6 低電力モード管理 2095
50.7 SPI ウェイクアップおよび割り込み 2097
50.8 I2S の主な機能 2098

参考資料

50.9 I2S の機能詳細 2099
50.9.1 I2S の概要 2099
50.9.2 SPI 機能とのピンの共用 2099
50.9.3 I2S/PCM モードで使用できるビットおよびフィールド 2100
50.9.4 スレーブおよびマスタモード 2101
50.9.5 サポートされるオーディオプロトコル 2101
50.9.6 シリアルインタフェースの付加的な柔軟性 2107
50.9.7 起動シーケンス 2109
50.9.8 停止シーケンス 2110
50.9.9 クロックジェネレータ 2111
50.9.10 内部 FIFO 2113
50.9.11 FIFO ステータスフラグ 2114
50.9.12 アンダーラン状況の取り扱い 2115
50.9.13 オーバーラン状況の取り扱い 2116
50.9.14 フレームエラー検出 2116
50.9.15 DMA インタフェース 2118
50.9.16 プログラミングの例 2119
50.9.17 フィリップス I2S 規格のスレーブ受信 2121
50.10 I2S ウェイクアップおよび割り込み 2122
50.11 SPI/I2S レジスタ 2123
50.11.1 SPI/I2S 制御レジスタ 1 （SPI2S_CR1） 2123
50.11.2 SPI 制御レジスタ 2（SPI_CR2） 2125
50.11.3 SPI 設定レジスタ 1 （SPI_CFG1） 2125
50.11.4 SPI 設定レジスタ 2 （SPI_CFG2） 2128
50.11.5 SPI/I2S 割り込み有効レジスタ（SPI2S_IER） 2130
50.11.6 SPI/I2S ステータスレジスタ（SPI2S_SR） 2131
50.11.7 SPI/I2S 割り込み／ステータスフラグクリアレジスタ（SPI2S_IFCR） 2134
50.11.8 SPI/I2S 送信データレジスタ（SPI2S_TXDR） 2135
50.11.9 SPI/I2S 受信データレジスタ（SPI2S_RXDR） 2135
50.11.10 SPI 多項式レジスタ（SPI_CRCPOLY） 2136
50.11.11 SPI トランスミッタ CRC レジスタ（SPI_TXCRC） 2136
50.11.12 SPI レシーバ CRC レジスタ（SPI_RXCRC） 2137
50.11.13 SPI アンダーランデータレジスタ（SPI_UDRDR） 2137
50.11.14 SPI/I2S 設定レジスタ（SPI_I2SCGFR） 2138
50.12 SPI レジスタマップとリセット値 2140

参考資料

51 シリアルオーディオインタフェース（SAI）. 2142

51.1 概要 2142

51.2 SAI の主な機能 2143

51.3 SAI の機能詳細 2144

51.3.1 SAI ブロック図 2144

51.3.2 SAI ピンおよび内部信号 2145

51.3.3 主要な SAI モード 2145

51.3.4 SAI 同期モード 2146

51.3.5 オーディオデータサイズ 2147

51.3.6 フレーム同期 2148

51.3.7 スロットの設定 2151

51.3.8 SAI クロックジェネレータ 2153

51.3.9 内部 FIFO 2155

51.3.10 PDM インタフェース 2157

51.3.11 AC'97 リンクコントローラ 2164

51.3.12 SPDIF 出力 2166

51.3.13 特有の機能 2168

51.3.14 エラーフラグ 2172

51.3.15 SAI の無効化 2176

51.3.16 SAI DMA インタフェース 2176

51.4 SAI 割り込み 2177

51.5 SAI レジスタ 2178

51.5.1 グローバル設定レジスタ（SAI_GCR） 2178

51.5.2 設定レジスタ 1（SAI_ACR1 / SAI_BCR1） 2179

51.5.3 設定レジスタ 2（SAI_ACR2 / SAI_BCR2） 2182

51.5.4 フレーム設定レジスタ（SAI_AFRCR / SAI_BFRCR） 2184

51.5.5 スロットレジスタ（SAI_ASLOTR / SAI_BSLOTR） 2185

51.5.6 割り込みマスクレジスタ 2（SAI_AIM / SAI_BIM） 2186

51.5.7 ステータスレジスタ（SAI_ASR / SAI_BSR） 2188

51.5.8 クリアフラグレジスタ（SAI_ACLRFR / SAI_BCLRFR） 2190

51.5.9 データレジスタ（SAI_ADR / SAI_BDR） 2192

51.5.10 PDM 制御レジスタ（SAI_PDMCR） 2192

51.5.11 PDM ディレイレジスタ（SAI_PDMDLY） 2193

51.5.12 SAI レジスタマップ 2196

参考資料

52 SPDIF レシーバインタフェース（SPDIFRX） 2198

52.1 SPDIFRX インタフェースの概要 2198

52.2 SPDIFRX の主な機能 2198

52.3 SPDIFRX の機能詳細 2198

52.3.1 SPDIFRX ピンおよび内部信号 2199

52.3.2 S/PDIF プロトコル（IEC-60958） 2200

52.3.3 SPDIFRX デコーダ（SPDIFRX_DC） 2202

52.3.4 クロック偏差に対する SPDIFRX の許容誤差 2206

52.3.5 SPDIFRX の同期 2206

52.3.6 SPDIFRX 処理 2208

52.3.7 データ受信管理 2210

52.3.8 専用制御フロー 2212

52.3.9 受信エラー 2212

52.3.10 クロック供給の方法 2215

52.3.11 シンボルクロック生成 2216

52.3.12 DMA インタフェース 2217

52.3.13 割り込みの生成 2218

52.3.14 レジスタの保護 2219

52.4 プログラミング手順 2219

52.4.1 初期化フェーズ 2220

52.4.2 SPDIFRX からの割り込みの処理 2221

52.4.3 DMA からの割り込みの処理 2221

52.5 SPDIFRX インタフェースレジスタ 2222

52.5.1 制御レジスタ（SPDIFRX_CR） 2222

52.5.2 割り込みマスクレジスタ（SPDIFRX_IMR） 2225

52.5.3 ステータスレジスタ（SPDIFRX_SR） 2226

52.5.4 割り込みフラグクリアレジスタ（SPDIFRX_IFCR） 2228

52.5.5 データ入力レジスタ（SPDIFRX_DR） 2229

52.5.6 データ入力レジスタ（SPDIFRX_DR） 2230

52.5.7 データ入力レジスタ（SPDIFRX_DR） 2231

52.5.8 チャネルステータスレジスタ（SPDIFRX_CSR） 2232

52.5.9 デバッグ情報レジスタ（SPDIFRX_DIR） 2233

52.5.10 SPDIFRX バージョンレジスタ（SPDIFRX_VERR） 2234

52.5.11 SPDIFRX 識別レジスタ（SPDIFRX_IDR） 2234

52.5.12 SPDIFRX サイズ識別レジスタ（SPDIFRX_SIDR） 2235

52.5.13 SPDIFRX インタフェースレジスタマップ 2236

参考資料

53 シングルワイヤプロトコルマスタインタフェース（SWPMI）. 2237

53.1 概要 2237

53.2 SWPMI の主な機能 2238

53.3 SWPMI の機能詳細 2239

53.3.1 SWPMI ブロック図 2239

53.3.2 SWPMI ピンおよび内部信号 2239

53.3.3 SWP の初期化と有効化 2240

53.3.4 SWP バス状態 2240

53.3.5 SWPMI_IO（内部トランシーバ）バイパス 2242

53.3.6 SWPMI ビットレート 2242

53.3.7 SWPMI フレーム処理 2243

53.3.8 送信手順 2243

53.3.9 受信手順 2248

53.3.10 エラー管理 2252

53.3.11 ループバックモード 2254

53.4 SWPMI 低電力モード 2254

53.5 SWPMI 割り込み 2255

53.6 SWPMI レジスタ 2256

53.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR） 2256

53.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR） 2257

53.6.3 SWPMI 割り込みおよびステータスレジスタ（SWPMI_ISR） 2258

53.6.4 SWPMI 割り込みフラグクリアレジスタ（SWPMI_ICR） 2260

53.6.5 SWPMI 割り込み有効レジスタ（SWMPMI_IER） 2261

53.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL） 2262

53.6.7 SWPMI 送信データレジスタ（SWPMI_TDR） 2262

53.6.8 SWPMI 受信データレジスタ（SWPMI_RDR） 2263

53.6.9 SWPMI オプションレジスタ（SWPMI_OR） 2263

53.6.10 SWPMI レジスタマップとリセット値の表 2264

54 マネジメントデータ インプット / アウトプット（MDIOS）. 2265

54.1 MDIOS の概要 2265

54.2 MDIOS の主な機能 2265

54.3 MDIOS の機能説明 2266

54.3.1 MDIOS ブロック図 2266

54.3.2 MDIOS ピンおよび内部信号 2266

54.3.3 MDIOS プロトコル 2267

54.3.4 MDIOS の有効化／無効化 2268

54.3.5 MDIOS データ 2268
54.3.6 MDIOS APB 周波数 2270
54.3.7 ライト／リードフラグおよび割り込み 2270
54.3.8 MDIOS エラー管理 2270
54.3.9 STOP モードでの MDIOS 2271
54.3.10 MDIOS 割り込み 2272
54.4 MDIOS レジスタ 2273
54.4.1 MDIOS 設定レジスタ（MDIOS_CR） 2273
54.4.2 MDIOS ライトフラグレジスタ（MDIOS_WRFR） 2274
54.4.3 MDIOS クリアライトフラグレジスタ（MDIOS_CWRFR） 2274
54.4.4 MDIOS リードフラグレジスタ（MDIOS_RDFR） 2275
54.4.5 MDIOS クリアリードフラグレジスタ（MDIOS_CRDFR） 2275
54.4.6 MDIOS ステータスレジスタ（MDIOS_SR） 2276
54.4.7 MDIOS クリアフラグレジスタ（MDIOS_CLRFR） 2277
54.4.8 MDIOS 入力データレジスタ（MDIOS_DINR0-MDIOS_DINR31） 2278
54.4.9 MDIOS 出力データレジスタ（MDIOS_DOUTR0-MDIOS_DOUTR31） 2278
54.4.10 MDIOS レジスタマップ 2279

55 セキュアデジタル入力／出力 MultiMediaCard インタフェース（SDMMC） 2280
55.1 SDMMC の主な機能 2280
55.2 SDMMC バストポロジ 2280
55.3 SDMMC 動作モード 2282
55.4 SDMMC の機能詳細 2283
55.4.1 SDMMC ブロック図 2283
55.4.2 SDMMC ピンおよび内部信号 2284
55.4.3 概要 2284
55.4.4 SDMMC アダプタ 2286
55.4.5 SDMMC AHB スレーブインタフェース 2306
55.4.6 SDMMC AHB マスタインタフェース 2306
55.4.7 MDMA リクエストの生成 2308
55.4.8 AHB と SDMMC_CK のクロック関係 2308
55.5 カード機能詳細 2309
55.5.1 SD I/O モード 2309
55.5.2 CMD12 送信タイミング 2317
55.5.3 SLEEP（CMD5） 2319
55.5.4 割り込みモード（Wait-IRQ） 2320

参考資料

55.5.5 ブート操作 2321
55.5.6 レスポンス R1b の処理 2325
55.5.7 リセットとカードサイクルパワー 2325
55.6 ハードウェアフロー制御 2327
55.7 超ハイスピードフェーズ I（UHS-I）の電圧スイッチ 2327
55.8 SDMMC レジスタ 2331
55.8.1 SDMMC 電源制御レジスタ（SDMMC_POWER）. 2331
55.8.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR）. 2332
55.8.3 SDMMC 引数レジスタ（SDMMC_ARGR）. 2335
55.8.4 SDMMC コマンドレジスタ（SDMMC_CMDR）. 2336
55.8.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMDR）. 2337
55.8.6 SDMMC レスポンス 1 ～ 4 レジスタ（SDMMC_RESPxR）（x = 1 ～ 4）. . . 2338
55.8.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）. 2338
55.8.8 SDMMC データ長レジスタ（SDMMC_DLENR）. 2339
55.8.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）. 2340
55.8.10 SDMMC データカウンタレジスタ（SDMMC_DCNTR）. 2342
55.8.11 SDMMC ステータスレジスタ（SDMMC_STAR）. 2343
55.8.12 SDMMC 割り込みクリアレジスタ（SDMMC_ICR）. 2345
55.8.13 SDMMC マスクレジスタ（SDMMC_MASKR）. 2348
55.8.14 SDMMC 確認応答タイマレジスタ（SDMMC_ACKTIMER）. 2351
55.8.15 SDMMC データ FIFO レジスタ（SDMMC_FIFOR）. 2352
55.8.16 SDMMC DMA 制御レジスタ（SDMMC_IDMACTRLR）. 2352
55.8.17 SDMMC IDMA バッファサイズレジスタ（SDMMC_IDMABSIZER）. 2353
55.8.18 SDMMC IDMA バッファ 0 ベースアドレスレジスタ（SDMMC_IDMABASE0R）. 2354
55.8.19 SDMMC IDMA バッファ 1 ベースアドレスレジスタ（SDMMC_IDMABASE1R）. 2354
55.8.20 SDMMC レジスタマップ 2355

56 FD コントローラエリアネットワーク（FDCAN）. 2357
56.1 概要 2357
56.2 FDCAN の主な機能 2360
56.3 FDCAN の機能詳細 2361
56.3.1 動作モード 2362
56.3.2 メッセージ RAM 2370
56.3.3 FIFO 確認応答の処理 2382
56.3.4 CAN のクロック較正 2383
56.3.5 TTCAN の動作（FDCAN1 のみ）. 2387

56.3.6 TTCAN の設定 2389
56.3.7 メッセージのスケジューリング 2391
56.3.8 TTCAN ギャップ制御 2397
56.3.9 ストップウォッチ 2398
56.3.10 ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期 2398
56.3.11 TTCAN のエラーレベル 2401
56.3.12 TTCAN のメッセージ処理 2402
56.3.13 TTCAN の割り込みとエラー処理 2405
56.3.14 レベル 0 2405
56.3.15 外部タイムスケジュールへの同期 2408
56.3.16 FDCAN の Rx バッファおよび FIFO 要素 2409
56.3.17 FDCAN の Tx バッファ要素 2411
56.3.18 FDCAN Tx イベント FIFO 要素 2413
56.3.19 FDCAN 標準メッセージ ID フィルタ要素 2414
56.3.20 FDCAN 拡張メッセージ ID フィルタ要素 2415
56.3.21 FDCAN のトリガメモリ要素 2417
56.4 FDCAN レジスタ 2419
56.4.1 FDCAN コア開放レジスタ（FDCAN_CREL） 2419
56.4.2 FDCAN コア開放レジスタ（FDCAN_ENDN） 2419
56.4.3 FDCAN データビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP） 2420
56.4.4 FDCAN テストレジスタ（FDCAN_TEST） 2421
56.4.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD） 2421
56.4.6 FDCAN CC 制御レジスタ（FDCAN_CCCR） 2422
56.4.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP） 2424
56.4.8 FDCAN タイムスタンプカウンタ設定レジスタ（FDCAN_TSCC） 2425
56.4.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV） 2426
56.4.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC） 2427
56.4.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV） 2427
56.4.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR） 2428
56.4.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR） 2429
56.4.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR） 2431
56.4.15 FDCAN 割り込みレジスタ（FDCAN_IR） 2431
56.4.16 FDCAN 割り込み有効レジスタ（FDCAN_IE） 2434
56.4.17 FDCAN 割り込みライン選択レジスタ（FDCAN_ILS） 2437
56.4.18 FDCAN 割り込みライン有効レジスタ（FDCAN_ILE） 2438
56.4.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_GFC） 2438

参考資料

56.4.20 FDCAN 標準 ID フィルタ設定レジスタ（FDCAN_SIDFC）.....2439
56.4.21 FDCAN 拡張 ID フィルタ設定レジスタ（FDCAN_XIDFC）.....2440
56.4.22 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）.....2441
56.4.23 FDCAN 高優先メッセージステータスレジスタ（FDCAN_HPMS）.....2441
56.4.24 FDCAN 新規データ 1 レジスタ（FDCAN_NDAT1）.....2442
56.4.25 FDCAN 新規データ 2 レジスタ（FDCAN_NDAT2）.....2442
56.4.26 FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）.....2443
56.4.27 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）.....2443
56.4.28 CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）.....2444
56.4.29 FDCAN Rx バッファ設定レジスタ（FDCAN_RXBC）.....2445
56.4.30 FDCAN Rx FIFO 1 設定レジスタ（FDCAN_RXF1C）.....2445
56.4.31 FDCAN Rx FIFO 1 ステータスレジスタ（FDCAN_RXF1S）.....2446
56.4.32 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）.....2447
56.4.33 FDCAN Rx バッファ要素サイズ設定レジスタ（FDCAN_RXESC）.....2447
56.4.34 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）.....2448
56.4.35 FDCAN Tx FIFO ／キューステータスレジスタ（FDCAN_TXFQS）.....2449
56.4.36 FDCAN Tx バッファ要素サイズ設定レジスタ（FDCAN_TXESC）.....2450
56.4.37 FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）.....2451
56.4.38 FDCAN Tx バッファ追加リクエストレジスタ（FDCAN_TXBAR）.....2451
56.4.39 FDCAN Tx バッファキャンセルリクエストレジスタ（FDCAN_TXBCR）...2452
56.4.40 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）.....2452
56.4.41 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）.....2453
56.4.42 FDCAN Tx バッファ送信割り込み有効レジスタ（FDCAN_TXBTIE）.....2453
56.4.43 FDCAN Tx バッファキャンセル終了割り込み有効レジスタ（FDCAN_ TXBCIE）.....2454
56.4.44 FDCAN Tx イベント FIFO 設定レジスタ（FDCAN_TXEFC）.....2454
56.4.45 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）.....2455
56.4.46 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）.....2456
56.4.47 FDCAN TT トリガメモリ設定レジスタ（FDCAN_TTTMC）.....2456
56.4.48 FDCAN TT リファレンスメッセージ設定レジスタ（FDCAN_TTRMC）.....2457
56.4.49 FDCAN TT 動作設定レジスタ（FDCAN_TTOCF）.....2458
56.4.50 FDCAN TT マトリックス制限レジスタ（FDCAN_TTMLM）.....2459
56.4.51 FDCAN TUR 設定レジスタ（FDCAN_TURCF）.....2460
56.4.52 FDCAN TT 動作制御レジスタ（FDCAN_TTOCN）.....2461
56.4.53 FDCAN TT グローバルタイムプリセットレジスタ（CAN_TTGTP）.....2463
56.4.54 FDCAN TT タイムマークレジスタ（FDCAN_TTTMK）.....2464
56.4.55 FDCAN TT 割り込みレジスタ（FDCAN_TTIR）.....2465

参考資料

56.4.56 FDCAN TT 割り込み有効レジスタ（FDCAN_TTIE）.....2467
56.4.57 FDCAN TT 割り込みライン選択レジスタ（FDCAN_TTILS）.....2468
56.4.58 FDCAN TT 動作ステータスレジスタ（FDCAN_TTOST）.....2470
56.4.59 FDCAN TUR Numerator 実測レジスタ（FDCAN_TURNA）.....2472
56.4.60 FDCAN TT ローカルおよびグローバルタイムレジスタ（FDCAN_TTLGT）.....2473
56.4.61 FDCAN TT サイクルタイムおよびカウントレジスタ（FDCAN_TTCTC）...2473
56.4.62 FDCAN TT キャプチャタイムレジスタ（FDCAN_TTCPT）.....2474
56.4.63 FDCAN TT サイクル同期マークレジスタ（FDCAN_TTCSM）.....2474
56.4.64 FDCAN TT トリガ選択レジスタ（FDCAN_TTTS）.....2475
56.4.65 FDCAN レジスタマップとリセット値の表 2476
56.5 CCU レジスタ 2482
56.5.1 クロック較正ユニットコア開放レジスタ（CCU_CREL）.....2482
56.5.2 較正設定レジスタ（CCU_CCFG）.....2482
56.5.3 較正ステータスレジスタ（CCU_CSTAT）.....2484
56.5.4 較正ウォッチドッグレジスタ（CCU_CWD）.....2485
56.5.5 クロック較正ユニット割り込みレジスタ（CCU_IR）.....2486
56.5.6 クロック較正ユニット割り込み有効レジスタ（CCU_IE）.....2486
56.5.7 CCU レジスタマップとリセット値の表 2487

57 USB On-The-Go ハイスピード（OTG_HS）..... 2488
57.1 概要 2488
57.2 USB_OTG の主な機能 2490
57.2.1 標準機能 2490
57.2.2 ホストモード機能 2491
57.2.3 ペリフェラルモード機能 2491
57.3 USB_OTG の実装 2492
57.4 USB OTG の機能詳細 2492
57.4.1 USB OTG ブロック図 2492
57.4.2 USB OTG ピンおよび内部信号 2494
57.4.3 OTG コア 2494
57.4.4 内蔵フルスピード OTG PHY 2495
57.4.5 ハイスピード OTG PHY 2495
57.4.6 I^2C インタフェースを使用する外部フルスピード OTG PHY 2495
57.5 OTG デュアルロールデバイス（DRD）.....2496
57.5.1 ID ラインの検出 2496
57.5.2 HNP デュアルロールデバイス 2496
57.5.3 SRP デュアルロールデバイス 2496

参考資料

57.6 USB ペリフェラル 2497
57.6.1 SRP 対応ペリフェラル 2497
57.6.2 ペリフェラルの状態 2497
57.6.3 ペリフェラルエンドポイント 2498
57.7 USB ホスト 2500
57.7.1 SRP 対応ホスト 2501
57.7.2 USB ホストの状態 2501
57.7.3 ホストチャネル 2502
57.7.4 ホストスケジューラ 2504
57.8 SOF トリガ 2505
57.8.1 ホストの SOF 2505
57.8.2 ペリフェラルの SOF 2505
57.9 電力オプション 2506
57.10 OTG_HFIR レジスタの動的更新 2506
57.11 USB データ FIFO 2507
57.11.1 ペリフェラル FIFO アーキテクチャ 2508
57.11.2 ホスト FIFO アーキテクチャ 2509
57.11.3 FIFO RAM の割り当て 2510
57.12 OTG_HS 割り込み 2512
57.13 OTG_HS 制御／ステータスレジスタ 2513
57.13.1 CSR メモリマップ 2513
57.14 OTG_HS レジスタ 2518
57.14.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL） 2518
57.14.2 OTG 割り込みレジスタ（OTG_GOTGINT） 2521
57.14.3 OTG AHB 設定レジスタ（OTG_GAHBCFG） 2522
57.14.4 OTG USB 設定レジスタ（OTG_GUSBCFG） 2523
57.14.5 OTG リセットレジスタ（OTG_GRSTCTL） 2527
57.14.6 OTG コア割り込みレジスタ（OTG_GINTSTS） 2529
57.14.7 OTG 割り込みマスクレジスタ（OTG_GINTMSK） 2534
57.14.8 OTG_FS 受信ステータスデバッグ読み出し／ OTG ステータス読み出しおよびポップレジスタ（OTG_GRXSTSR/OTG_GRXSTSP） 2537
57.14.9 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ） 2538
57.14.10 OTG ホスト非周期的送信 FIFO サイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ(OTG_DIEPTXF0) 2539
57.14.11 OTG 非周期的送信 FIFO ／キューステータスレジスタ（OTG_HNPTXSTS） 2540
57.14.12 OTG I^2C アクセスレジスタ（OTG_GI2CCTL） 2541
57.14.13 OTG 一般コア設定レジスタ（OTG_GCCFG） 2542

参考資料

57.14.14 OTG コア ID レジスタ（OTG_CID）.....2543
57.14.15 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）.....2544
57.14.16 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）.....2548
57.14.17 OTG デバイス IN エンドポイント送信 FIFO サイズレジスタ（OTG_DIEPTXFx）（x=1 ～ 8、ここで x は FIFO_number）.....2548
57.14.18 ホストモードレジスタ.....2548
57.14.19 OTG ホスト設定レジスタ（OTG_HCFG）.....2549
57.14.20 OTG ホストフレームインターバルレジスタ（OTG_HFIR）.....2549
57.14.21 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）.....2550
57.14.22 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）.....2551
57.14.23 OTG ホスト全チャネル割り込みレジスタ（OTG_HAINT）.....2552
57.14.24 OTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK）.....2552
57.14.25 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）.....2553
57.14.26 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0 ～ 15、ここで x = Channel_number）.....2555
57.14.27 OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）（x = 0..15、ここで x = Channel_number）.....2557
57.14.28 OTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0 ～ 15、ここで x = Channel_number）.....2557
57.14.29 OTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）（x = 0 ～ 15、ここで x = Channel_number）.....2558
57.14.30 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0 ～ 15、ここで x = Channel_number）.....2560
57.14.31 OTG ホストチャネル x DMA アドレスレジスタ（OTG_HCDMAx）（x = 0..15、ここで x = Channel_number）.....2561
57.14.32 デバイスモードレジスタ.....2561
57.14.33 OTG デバイス設定レジスタ（OTG_DCFG）.....2561
57.14.34 OTG デバイス制御レジスタ（OTG_DCTL）.....2563
57.14.35 OTG デバイスステータスレジスタ（OTG_DSTS）.....2566
57.14.36 OTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）.....2567
57.14.37 OTG デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK）.....2568
57.14.38 OTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT）.....2569
57.14.39 OTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK）.....2570
57.14.40 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）.....2570
57.14.41 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）.....2571
57.14.42 OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）.....2571
57.14.43 OTG デバイス各エンドポイント割り込みレジスタ（OTG_DEACHINT）.....2572

57.14.44 OTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ（OTG_DIEPEMPMSK）. 2573
57.14.45 OTG デバイス各エンドポイント割り込みレジスタマスク（OTG_DEACHINTMSK）. 2573
57.14.46 OTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 0 ～ 8、ここで x = Endpoint_number）. 2574
57.14.47 OTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）. 2576
57.14.48 OTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1 ～ 8、ここで x = Endpoint_number）. 2578
57.14.49 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）（x = 0 ～ 8、ここで x = Endpoint_number）. 2580
57.14.50 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）（x = 0 ～ 8、ここで x = Endpoint_number）. 2582
57.14.51 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）. 2583
57.14.52 OTG デバイスチャネル x DMA アドレスレジスタ（OTG_DIEPDMAx）（x = 0 ～ 15、ここで x = Channel_number）. 2583
57.14.53 OTG デバイスチャネル x DMA アドレスレジスタ（OTG_DOEPDMAx）（x = 0 ～ 15、ここで x = Channel_number）. 2584
57.14.54 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）. 2584
57.14.55 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1 ～ 8、ここで x = Endpoint_number）. 2585
57.14.56 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）（x = 0 ～ 8、ここでx = Endpoint_number）. 2586
57.14.57 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1 ～ 8、ここで x =Endpoint_number）. 2587
57.14.58 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）. 2588
57.14.59 OTG_HS レジスタマップ 2589
57.15 OTG_HS プログラミングモデル 2599
57.15.1 コアの初期化 2599
57.15.2 ホストの初期化 2600
57.15.3 デバイスの初期化 2600
57.15.4 DMA モード 2601
57.15.5 ホストプログラミングモデル 2601
57.15.6 デバイスプログラミングモデル 2632
57.15.7 最悪ケースの応答時間 2650
57.15.8 OTG プログラミングモデル 2652

参考資料

58 Ethernet（ETH）：メディアアクセス制御（MAC）、DMA コントローラ付き 2658

58.1 Ethernet の概要 2658

58.2 Ethernet の主な機能 2658

58.2.1 MAC コア機能 2658

58.2.2 DMA の機能 2660

58.2.3 バスインタフェースの機能 2660

58.3 Ethernet ピンおよび内部信号 2661

58.4 Ethernet アーキテクチャ 2663

58.4.1 DMA コントローラ 2663

58.4.2 MTL 2671

58.4.3 MAC 2671

58.5 Ethernet 機能の説明：MAC 2675

58.5.1 ダブル VLAN 処理 2675

58.5.2 ソースアドレスおよび VLAN の挿入、書き換え、または削除 2676

58.5.3 パケットフィルタ 2677

58.5.4 IEEE 1588 タイムスタンプ 2684

58.5.5 IPv4 ARP オフロード 2695

58.5.6 TCP セグメンテーションオフロード 2696

58.5.7 ループバック 2700

58.5.8 フロー制御 2700

58.5.9 チェックサムオフロードエンジン 2702

58.5.10 MAC 管理カウンタ 2707

58.5.11 MAC によって生成される割り込み 2707

58.5.12 MAC および MMC レジスタの説明 2707

58.6 Ethernet 機能の説明：PHY インタフェース 2708

58.6.1 ステーション管理エージェント（SMA） 2708

58.6.2 メディア独立インタフェース（MII） 2711

58.6.3 減少メディア独立インタフェース信号（RMII） 2713

58.7 Ethernet 低電力モード 2716

58.7.1 省電力型 Ethernet 2716

58.7.2 電源管理 2717

58.7.3 パワーダウンおよびウェイクアップシーケンス 2719

58.8 Ethernet 割り込み 2720

58.8.1 DMA 割り込み 2720

58.8.2 MTL 割り込み 2721

58.8.3 MAC 割り込み 2722

参考資料

58.9 Ethernet のプログラミングモデル 2723
58.9.1 DMA の初期化 2723
58.9.2 MTL の初期化 2724
58.9.3 MAC の初期化 2724
58.9.4 通常受信および送信動作の実行 2725
58.9.5 送信の停止および開始 2725
58.9.6 MII リンク状態遷移に関するプログラミングガイドライン 2726
58.9.7 IEEE 1588 タイムスタンプに関するプログラミングガイドライン 2727
58.9.8 省電力型 Ethernet に関するプログラミングガイドライン 2728
58.9.9 柔軟性の高い Pulse-Per-Second（PPS）出力に関するプログラミングガイドライン 2729
58.9.10 TSO に関するプログラミングガイドライン 2731
58.9.11 受信時の VLAN フィルタに関するプログラミングガイドライン 2731
58.10 ディスクリプタ 2732
58.10.1 ディスクリプタの概要 2732
58.10.2 ディスクリプタの構造 2732
58.10.3 送信ディスクリプタ 2734
58.10.4 受信ディスクリプタ 2745
58.11 Ethernet レジスタの説明 2757
58.11.1 レジスタマップ 2757
58.11.2 Ethernet DMA レジスタ 2757
58.11.3 Ethernet MTL レジスタ 2783
58.11.4 Ethernet MAC および MMC レジスタ 2795

59 HDMI-CEC コントローラ（HDMI-CEC） 2890
59.1 概要 2890
59.2 HDMI-CEC コントローラの主な機能 2890
59.3 HDMI-CEC の機能詳細 2891
59.3.1 HDMI-CEC ピン および内部信号 2891
59.3.2 HDMI-CEC ブロック図 2892
59.3.3 メッセージの説明 2893
59.3.4 ビットタイミング 2894
59.4 アービトレーション 2894
59.4.1 SFT オプションビット 2896
59.5 エラー処理 2896
59.5.1 ビットエラー 2896
59.5.2 メッセージエラー 2896

参考資料

59.5.3 ビット立ち上がりエラー（BRE）. 2897
59.5.4 ショートビット周期エラー（SBPE）. 2897
59.5.5 ロングビット周期エラー（LBPE）. 2897
59.5.6 送信エラー検出（TXERR）. 2899
59.6 HDMI-CEC 割り込み. 2900
59.7 HDMI-CEC レジスタ. 2901
59.7.1 CEC 制御レジスタ（CEC_CR）. 2901
59.7.2 CEC 設定レジスタ（CEC_CFGR）. 2902
59.7.3 CEC Tx データレジスタ（CEC_TXDR）. 2904
59.7.4 CEC Rx データレジスタ（CEC_RXDR）. 2904
59.7.5 CEC 割り込みおよびステータスレジスタ（CEC_ISR）. 2905
59.7.6 CEC 割り込み有効レジスタ（CEC_IER）. 2907
59.7.7 HDMI-CEC レジスタマップ. 2909

60 デバッグインフラストラクチャ. 2910
60.1 概要. 2910
60.2 デバッグインフラストラクチャ 機能. 2911
60.3 デバッグインフラストラクチャ機能詳細. 2911
60.3.1 デバッグインフラストラクチャ ブロック図. 2911
60.3.2 デバッグインフラストラクチャ ピンおよび内部信号. 2912
60.3.3 デバッグインフラストラクチャ 電力供給、クロック制御、および リセット. 2913
60.4 デバッグアクセスポートの機能詳細. 2915
60.4.1 シリアルワイヤ／JTAG デバッグポート（SWJ-DP）. 2915
60.4.2 アクセスポート. 2930
60.5 トレースおよびデバッグサブシステムの機能説明. 2936
60.5.1 システム ROM テーブル. 2936
60.5.2 グローバルタイムスタンプジェネレータ（TSG）. 2945
60.5.3 クロストリガインタフェース（CTI）／マトリックス（CTM）. 2954
60.5.4 トレースファネル（CSTF）. 2973
60.5.5 組み込みトレース FIFO （ETF）. 2984
60.5.6 トレースポートインタフェースユニット（TPIU）. 3006
60.5.7 シリアルワイヤ出力（SWO）と SWO トレースファネル（SWTF）. 3023
60.5.8 マイクロコントローラデバッグユニット（DBGMCU）. 3044
60.6 Cortex-M7 デバッグ機能詳細. 3052
60.6.1 Cortex-M7 の ROM テーブル. 3052
60.6.2 Cortex-M7 データウォッチポイントおよびトレースユニット（DWT）. 3064

参考資料

60.6.3 Cortex-M7 計装トレースマクロセル（ITM）. 3077
60.6.4 Cortex-M7 ブレークポイントユニット（FPB）. 3086
60.6.5 Cortex-M7 組み込みトレースマクロセル（ETM）. 3093
60.6.6 Cortex-M7 クロストリガインタフェース（CTI）. 3124
60.7 デバッグインフラストラクチャのリファレンスマニュアル 3125

61 デバイス電子署名 3126
61.1 ユニークデバイス ID レジスタ（96 ビット）. 3126
61.2 フラッシュサイズ 3127
61.3 パッケージデータレジスタ 3128

改版履歴 3129

参考資料

表の一覧

表 1. バスマスタとバススレーブの相互接続 98
表 2. レジスタ境界アドレス 104
表 3. ブートモード 109
表 4. パッケージのピンまたはボールに接続されたフラッシュ入力／出力信号 112
表 5. フラッシュ内部入力／出力信号 112
表 6. フラッシュモジュール - 1 MB のデュアルバンク構成 113
表 7. バス周波数（ACLK）および VCORE レンジによるウェイトステートの数 117
表 8. 並列度パラメータ 122
表 9. プログラミング速度 122
表 10. フラッシュユーザオプションバイトの一覧 130
表 11. RDP 値と読み出し保護レベルの比較 134
表 12. 許可されたアクセスと読み出し保護レベルの比較 134
表 13. AXI インタフェースのメモリマッピング SWAP_BANK = ‘0’ 137
表 14. AXI インタフェースのメモリマッピング SWAP_BANK = ‘1’ 137
表 15. バンクスワッピングシーケンス 138
表 16. レジスタマップとリセット値の表 182
表 17. 推奨する用語の一覧 187
表 18. フラッシュ保護メカニズム 189
表 19. セキュアユーザソフトウェア選択の使用例 195
表 20. フラッシュ保護領域のアクセス権限の一覧 196
表 21. ASIB の設定 198
表 22. AMIB の設定 199
表 23. AXI インタコネクトレジスタマップとリセット値 209
表 24. パッケージのピンまたはボールに接続された PWR 入力／出力信号 219
表 25. PWR 内部入力／出力信号 219
表 26. 電源設定の制御 222
表 27. 低電力モードの概要 239
表 28. PDDS_Dn 低電力モードの制御 241
表 29. 低電力終了モードフラグ 243
表 30. CSleep モード 252
表 31. CStop モード 253
表 32. DStop モードの概要 254
表 33. DStop モード 255
表 34. STOP モード動作 256
表 35. STOP モード 257
表 36. DStandby モード 258
表 37. STANDBY および STOP フラグ 260
表 38. STANDBY モード 260
表 39. 電源制御レジスタマップとリセット値 270
表 40. BDMA および DMAMUX2 の相互接続 278
表 41. BDMA および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT） 278
表 42. LPUART1 初期プログラミング（LPUART1_INIT） 280
表 43. LPUART1 初期プログラミング（LPUART1_Start） 280
表 44. パッケージのピンまたはボールに接続された RCC 入出力信号 284
表 45. RCC 内部入力／出力信号 285
表 46. リセット供給の概要 288
表 47. リセットソース識別（RCC_RSR） 290
表 48. クロックタイマと pclk の比率 307
表 49. STOPWUCK と STOPKERWUCK の説明 308

表 50. HSIKERON および CSIKERON の動作 309
表 51. カーネルクロック分配の概要 311
表 52. システムの状態の概要 325
表 53. D1 および D2 ペリフェラルでのペリフェラルクロックの有効化 330
表 54. D3 ペリフェラルでのペリフェラルクロックの有効化 331
表 55. 割り込みソースとその制御 335
表 56. RCC_RSR アドレスオフセットとリセット値 400
表 57. RCC_AHB3ENR アドレスオフセットとリセット値 402
表 58. RCC_AHB1ENR アドレスオフセットとリセット値 404
表 59. RCC_AHB2ENR アドレスオフセットとリセット値 406
表 60. RCC_AHB4ENR アドレスオフセットとリセット値 408
表 61. RCC_APB3ENR アドレスオフセットとリセット値 410
表 62. RCC_APB1ENR アドレスオフセットとリセット値 411
表 63. RCC_APB1ENR アドレスオフセットとリセット値 415
表 64. RCC_APB2ENR アドレスオフセットとリセット値 416
表 65. RCC_APB4ENR アドレスオフセットとリセット値 419
表 66. RCC_AHB3LPENR アドレスオフセットとリセット値 422
表 67. RCC_AHB1LPENR アドレスオフセットとリセット値 424
表 68. RCC_AHB2LPENR アドレスオフセットとリセット値 426
表 69. RCC_AHB4LPENR アドレスオフセットとリセット値 428
表 70. RCC_APB3LPENR アドレスオフセットとリセット値 431
表 71. RCC_APB1LLPENR アドレスオフセットとリセット値 432
表 72. RCC_APB1HLPENR アドレスオフセットとリセット値 436
表 73. RCC_APB2LPENR アドレスオフセットとリセット値 437
表 74. RCC_APB4LPENR アドレスオフセットとリセット値 440
表 75. RCC レジスタマップとリセット値 443
表 76. CRS 内部入力／出力信号 454
表 77. 低電力モードが CRS に与える影響 457
表 78. 割り込み制御ビット 457
表 79. CRS レジスタマップとリセット値 463
表 80. HSEM 内部入力／出力信号 465
表 81. 許可された AHB バスマスタ ID 470
表 82. HSEM レジスタマップとリセット値 476
表 83. ポートビット設定表 480
表 84. GPIO レジスタマップとリセット値 494
表 85. SYSCFG レジスタマップとリセット値 515
表 86. ペリフェラル相互接続マトリックス（D2 ドメイン） 519
表 87. ペリフェラル相互接続マトリックス（D3 ドメイン） 520
表 88. ペリフェラル相互接続マトリックスの詳細 521
表 89. EXTI ウェイクアップ入力 538
表 90. EXTI ペンディングリクエストクリア入力 541
表 91. MDMA 543
表 92. DMAMUX1、DMA1、および DMA2 接続 544
表 93. DMAMUX2 および BDMA 接続 549
表 94. MDMA 内部入力／出力信号 554
表 95. MDMA 割り込みリクエスト 561
表 96. MDMA レジスタマップとリセット値 579
表 97. DMA 内部入力／出力信号 583
表 98. 転送元および転送先アドレス 585
表 99. ダブルバッファモード（DBM=1）における転送元および転送先アドレスレジスタ 590
表 100. パッキング／アンパッキングおよびエンディアンの動作（ビット PINC=MINC=1） 591
表 101. PSIZE と MSIZE に対する NDT に関する制限事項 591

表 102. FIFO 閾値設定 593
表 103. 実現可能な DMA 設定 597
表 104. DMA 割り込みリクエスト 600
表 105. DMA レジスタマップとリセット値 611
表 106. プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合） 618
表 107. BDMA 割り込みリクエスト 619
表 108. BDMA レジスタマップとリセット値 626
表 109. DMAMUX1 および DMAMUX2 のインスタンス化 630
表 110. DMAMUX1：リソースへのマルチプレクサ入力の割り当て 630
表 111. DMAMUX1：リソースへのトリガ入力の割り当て 631
表 112. DMAMUX1：リソースへの同期入力の割り当て 632
表 113. DMAMUX2：リソースへのマルチプレクサ入力の割り当て 632
表 114. DMAMUX2：リソースへのトリガ入力の割り当て 632
表 115. DMAMUX2：リソースへの同期入力の割り当て 633
表 116. DMAMUX 信号 635
表 117. DMAMUX 割り込み 639
表 118. DMAMUX レジスタマップとリセット値 648
表 119. DMA2D 内部入力／出力信号 652
表 120. 入力でサポートされているカラーモード 653
表 121. メモリ内のデータ順序 655
表 122. アルファモード設定 655
表 123. サポートされている CLUT カラーモード 656
表 124. メモリ内の CLUT データ順序 656
表 125. 出力でサポートされているカラーモード 657
表 126. メモリ内のデータ順序 658
表 127. メモリ内の MCU 順序 662
表 128. DMA2D 割り込みリクエスト 663
表 129. DMA2D レジスタマップとリセット値 681
表 130. NVIC 683
表 131. EXTI イベント入力設定およびレジスタ制御 693
表 132. 設定可能なイベント入力、非同期エッジ検出回路リセット 695
表 133. EXTI イベント入力の配置 700
表 134. マスキングの機能 702
表 135. 非同期割り込み／イベントコントローラのレジスタマップとリセット値 720
表 136. CRC 内部入力／出力信号 724
表 137. CRC レジスタマップとリセット値 729
表 138. FMC ピン 733
表 139. FMC バンクのマッピングオプション 736
表 140. NOR/PSRAM バンク選択 736
表 141. NOR/PSRAM 外部メモリアドレス 736
表 142. NAND メモリマッピングおよびタイミングレジスタ 737
表 143. NAND バンク選択 737
表 144. SDRAM バンク選択 737
表 145. SDRAM アドレスマッピング 738
表 146. 8 ビットデータバス幅での SDRAM アドレスマッピング 738
表 147. 16 ビットデータバス幅での SDRAM アドレスマッピング 739
表 148. 32 ビットデータバス幅での SDRAM アドレスマッピング 740
表 149. プログラム可能な NOR/PSRAM のアクセスパラメータ 741
表 150. 非マルチプレクス I/O NOR フラッシュメモリ 742
表 151. 16 ビットマルチプレクス I/O NOR フラッシュメモリ 742
表 152. 非マルチプレクス I/O PSRAM/SRAM 743
表 153. 16 ビットマルチプレクス I/O PSRAM 743

参考資料

表 154. NOR フラッシュ /PSRAM：サポートされているメモリとトランザクションの例 744
表 155. FMC_BCRx ビットフィールド 747
表 156. FMC_BTRx ビットフィールド 748
表 157. FMC_BCRx ビットフィールド 750
表 158. FMC_BTRx ビットフィールド 751
表 159. FMC_BWTRx ビットフィールド 751
表 160. FMC_BCRx ビットフィールド 753
表 161. FMC_BTRx ビットフィールド 754
表 162. FMC_BWTRx ビットフィールド 754
表 163. FMC_BCRx ビットフィールド 756
表 164. FMC_BTRx ビットフィールド 757
表 165. FMC_BWTRx ビットフィールド 757
表 166. FMC_BCRx ビットフィールド 759
表 167. FMC_BTRx ビットフィールド 759
表 168. FMC_BWTRx ビットフィールド 760
表 169. FMC_BCRx ビットフィールド 762
表 170. FMC_BTRx ビットフィールド 762
表 171. FMC_BCRx ビットフィールド 768
表 172. FMC_BTRx ビットフィールド 768
表 173. FMC_BCRx ビットフィールド 769
表 174. FMC_BTRx ビットフィールド 770
表 175. プログラ可能な NAND フラッシュのアクセスパラメータ 780
表 176. 8 ビット NAND フラッシュメモリ 780
表 177. 16 ビット NAND フラッシュメモリ 781
表 178. サポートされるメモリおよびトランザクション 781
表 179. ECC 結果関連ビット 791
表 180. SDRAM 信号 792
表 181. FMC レジスタマップ 810
表 182. QUADSPI 内部信号 814
表 183. QUADSPI ピン 814
表 184. QUADSPI 割り込みリクエスト 828
表 185. QUADSPI レジスタマップとリセット値 841
表 186. DLYB 内部入力／出力信号 843
表 187. 遅延ブロック制御 843
表 188. DLYB レジスタのマップとリセット値 846
表 189. ADC 内部信号 851
表 190. ADC ピン 851
表 191. レギュラ外部トリガのトリガ極性の設定 872
表 192. インジェクト外部トリガのトリガ極性の設定 872
表 193. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ 873
表 194. ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ 874
表 195. TSAR タイミングは分解能に依存 886
表 196. オフセット計算対データ分解能 889
表 197. 16 ビットデータフォーマット 892
表 198. 16 ビットフォーマットの数値例（太字は飽和を示す） 892
表 199. アナログウォッチドッグチャネル選択 900
表 200. アナログウォッチドッグ 1、2、3 の比較 901
表 201. オーバーサンプリング動作モードの概要 909
表 202. ADC ごとの ADC 割り込み 928
表 203. DELAY ビット対 ADC 分解能 966
表 204. ADC グローバルレジスタマップ 968

表 205. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））.....968
表 206. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300）.....971
表 207. DAC ピン.....974
表 208. DAC 内部信号.....974
表 209. DAC トリガ選択.....977
表 210. サンプルおよびリフレッシュタイミング.....981
表 211. チャネル出力モードの概要.....983
表 212. 低電力モードが DAC に与える影響.....989
表 213. DAC レジスタマップとリセット値.....1005
表 214. VREF バッファモード.....1007
表 215. VREFBUF レジスタマップとリセット値.....1009
表 216. COMP 入出力内部信号.....1012
表 217. COMP 入出力ピン.....1012
表 218. GPIO に対する COMP1_OUT の割り当て.....1015
表 219. GPIO に対する COMP2_OUT の割り当て.....1015
表 220. 低電力モードでのコンパレータの動作.....1017
表 221. 割り込み制御ビット.....1017
表 222. 割り込み制御ビット.....1018
表 223. COMP レジスタマップとリセット値.....1026
表 224. オペアンプの可能な接続.....1028
表 225. 動作モードと較正.....1035
表 226. 低消費電力モードが OPAMP に与える影響.....1037
表 227. OPAMP レジスタマップとリセット値.....1044
表 228. DFSDM1 の実装.....1047
表 229. DFSDM 外部ピン.....1049
表 230. DFSDM 内部信号.....1049
表 231. DFSDM トリガ接続.....1049
表 232. DFSDM ブレーク接続.....1050
表 233. フィルタの最大出力分解能（フィルタ出力のピークデータ値）一部の FOSR 値.....1064
表 234. 積分器の最大出力分解能（積分器出力のピークデータ値）（一部の IOSR 値、FOSR = 256 および Sinc3 フィルタタイプ（最大データ）の場合）.....1065
表 235. DFSDM 割り込みリクエスト.....1073
表 236. DFSDM レジスタマップとリセット値.....1093
表 237. DCMI 内部信号.....1105
表 238. DCMI 外部信号.....1105
表 239. 32 ビットワード（8 ビット幅）でキャプチャされたデータバイトの配置.....1106
表 240. 32 ビットワード（10 ビット幅）でキャプチャされたデータバイトの配置.....1106
表 241. 32 ビットワード（12 ビット幅）でキャプチャされたデータバイトの配置.....1107
表 242. 32 ビットワード（14 ビット幅）でキャプチャされたデータバイトの配置.....1107
表 243. モノクロプログレッシブビデオフォーマットでのデータ保存.....1113
表 244. RGB プログレッシブビデオフォーマットでのデータ保存.....1114
表 245. YCbCr プログレッシブビデオフォーマットでのデータ保存.....1114
表 246. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード.....1115
表 247. DCMI 割り込み.....1115
表 248. DCMI レジスタマップとリセット値.....1128
表 249. LCD-TFT 内部信号.....1130
表 250. LCD-TFT ピンと信号インタフェース.....1131
表 251. 各レジスタのクロックドメイン.....1132
表 252. LTDC レジスタアクセスと更新時間.....1132
表 253. ピクセルデータマッピングとカラーフォーマット.....1136

表 254. LTDC 割り込みリクエスト 1140
表 255. LTDC レジスタマップとリセット値 1160
表 256. JPEG 内部信号 1164
表 257. JPEG コーデック割り込み 1168
表 258. JPEG コーデックレジスタマップとリセット値 1176
表 259. RNG 内部入力／出力信号 1179
表 260. RNG 割り込みリクエスト 1187
表 261. RNG レジスタマップとリセット値 1191
表 262. CRYP 内部入力／出力信号 1194
表 263. カウンタモード初期化ベクタ 1221
表 264. GCM 最終ブロックの定義 1224
表 265. GCM モード IV レジスタの初期化 1224
表 266. CCM モード IV レジスタの初期化 1231
表 267. DES ／トリプル DES データスワッピング機能 1236
表 268. AES データスワッピング機能 1238
表 269. キーレジスタ CRYP_KxR/LR エンディアン
（トリプル DES K1/2/3 および AES 128/192/256 ビットキー） 1238
表 270. 初期化ベクトルレジスタ CRYP_IVxR エンディアン 1239
表 271. メモリからペリフェラルへの
DMA 転送のための暗号プロセッサの設定 1240
表 272. ペリフェラルからメモリへの
DMA 転送のための暗号プロセッサの設定 1240
表 273. CRYP 割り込みリクエスト 1242
表 274. 128 ビットブロックあたりの ECB、CBC および CTR の処理時間
（クロックサイクル数） 1244
表 275. 128 ビットブロックあたりの GCM および CCM の処理時間（クロックサイクル数） 1244
表 276. CRYP レジスタマップとリセット値 1259
表 277. HASH 内部入力／出力信号 1263
表 278. ハッシュプロセッサ出力 1267
表 279. HASH 割り込みリクエスト 1273
表 280. 処理時間（クロックサイクル数） 1273
表 281. HASH レジスタマップとリセット値 1284
表 282. HRTIM の入出力の概要 1289
表 283. タイマ分解能および最小 PWM 周波数（f_{HRTIM} = 400 MHz の場合） 1290
表 284. 周期および比較レジスタの最小値および最大値 1293
表 285. タイマ動作モード 1293
表 286. タイマ A ～ E にわたるイベントマピング 1298
表 287. デッドタイム分解能と最大の絶対値 1306
表 288. 外部イベントのマッピングおよび関連機能 1312
表 289. 出力のセット／リセット遅延時間とジッタ 対 外部イベント動作モード 1313
表 290. タイマごとのフィルタリング信号のマッピング 1316
表 291. タイマごとのウィンドウイング信号のマッピング（EEFLTR[3:0] = 1111） 1318
表 292. HRTIM のプリロード可能な制御レジスタおよび関連する更新ソース 1327
表 293. 更新イネーブル入力とソース 1328
表 294. マスタタイマ更新イベントの伝播 1330
表 295. TIMx 更新イベントの伝播 1330
表 296. 更新を生成できるリセットイベント 1331
表 297. タイマのリセットのための更新イベントの伝播 1332
表 298. 出力状態のプログラミング（x = A ～ E、y = 1 または 2） 1333
表 299. バーストモードのためのタイマ出力のプログラミング 1336
表 300. 汎用タイマからのバーストモードのクロックソース 1338
表 301. フォールト入力 1346

表 302. サンプリングレートおよびフィルタ長 対 FLTFxF[3:0] およびクロック設定 1347
表 303. 同期イベントの効果 対 動作モード 1352
表 304. HRTIM 割り込みの概要 1356
表 305. HRTIM の DMA リクエストの概要 1358
表 306. RTIM グローバルレジスタマップ 1447
表 307. HRTIM レジスタマップとリセット値：マスタタイマ 1447
表 308. HRTIM レジスタマップとリセット値：TIMx（x= A ～ E） 1449
表 309. HRTIM レジスタマップとリセット値：共通機能 1452
表 310. タイマ出力と BRK/BRK2 入力の動作 1498
表 311. カウント方向とエンコーダ信号 1505
表 312. TIMx 内部トリガ接続 1522
表 313. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット 1536
表 314. TIM1 レジスタマップとリセット値 1557
表 315. TIM8 レジスタマップとリセット値 1560
表 316. カウント方向とエンコーダ信号 1596
表 317. TIMx 内部トリガ接続 1614
表 318. 標準 OCx チャネルの出力制御ビット 1624
表 319. TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値 1634
表 320. TIMx 内部トリガ接続 1664
表 321. 標準 OCx チャネルの出力制御ビット 1672
表 322. TIM12 レジスタマップとリセット値 1676
表 323. 標準 OCx チャネルの出力制御ビット 1684
表 324. TIM13/TIM14 レジスタマップとリセット値 1687
表 325. TIMx 内部トリガ接続 1730
表 326. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15） 1739
表 327. TIM15 レジスタマップとリセット値 1748
表 328. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17） 1760
表 329. TIM16/TIM17 レジスタマップとリセット値 1771
表 330. TIM6/TIM7 レジスタマップとリセット値 1785
表 331. STM32H7x3 LPTIM 機能 1786
表 332. LPTIM 内部信号 1789
表 333. HRTIM の入出力ピン 1789
表 334. LPTIM1 外部トリガ接続 1789
表 335. LPTIM2 外部トリガ接続 1790
表 336. LPTIM3 外部トリガ接続 1790
表 337. LPTIM4 外部トリガ接続 1790
表 338. LPTIM5 外部トリガ接続 1790
表 339. LPTIM1 入力 1 接続 1791
表 340. LPTIM1 入力 2 接続 1791
表 341. LPTIM2 入力 1 接続 1791
表 342. LPTIM2 入力 2 接続 1791
表 343. LPTIM3 入力 1 接続 1792
表 344. プリスケーラの分周比 1793
表 345. エンコーダのカウントシナリオ 1800
表 346. 割り込みイベント 1801
表 347. LPTIM レジスタマップとリセット値 1813
表 348. WWDG 内部入力／出力信号 1815
表 349. WWDG レジスタマップとリセット値 1820
表 350. IWDG 内部入力／出力信号 1822
表 351. IWDG レジスタマップとリセット値 1829
表 352. RTC ピンおよび内部信号 1834
表 353. RTC ピン PC13 の設定 1834

表 354. RTC_OUT の配置 1835
表 355. RTC 機能（モード共通） 1836
表 356. 低消費電力モードが RTC に与える影響 1848
表 357. 割り込み制御ビット 1849
表 358. RTC レジスタマップとリセット値 1874
表 359. STM32H7x3I2C の実装 1877
表 360. アナログフィルタとデジタルフィルタの比較 1881
表 361. I2C-SMBUS 仕様のデータのセットアップおよびホールド時間 1884
表 362. I2C 設定表 1888
表 363. I2C-SMBUS 仕様のクロックタイミング 1899
表 364. fI2CCLK = 8 MHz でのタイミング設定の例 1909
表 365. fI2CCLK = 16 MHz でのタイミング設定の例 1909
表 366. fI2CCLK = 48 MHz でのタイミング設定の例 1910
表 367. SMBus タイムアウト仕様 1912
表 368. SMBUS の PEC 設定 1914
表 369. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms） 1915
表 370. さまざまな i2c_ker_ck 周波数での TIMEOUTB の設定例 1916
表 371. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs） 1916
表 372. 低電力モード 1927
表 373. I2C 割り込みリクエスト 1928
表 374. I2C レジスタマップとリセット値 1944
表 375. USART/LPUART の機能 1948
表 376. サンプリングされたデータからのノイズ検出 1963
表 377. BRR [3:0] = 0000 のときの USART レシーバの許容誤差 1966
表 378. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差 1966
表 379. USART フレームのフォーマット 1970
表 380. USART 割り込みリクエスト 1991
表 381. USART レジスタマップとリセット値 2020
表 382. lpuart_ker_ck_pres = 32,768 KHz でプログラムされたボーレートのエラー計算 2034
表 383. fCK = 100 MHz でプログラムされたボーレートのエラー計算 2034
表 384. LPUART レシーバの許容誤差 2035
表 385. LPUART フレームのフォーマット 2038
表 386. LPUART 割り込みリクエスト 2047
表 387. LPUART レジスタマップとリセット値 2066
表 388. STM32H7xx SPI の機能 2068
表 389. SPI ウェイクアップおよび割り込みリクエスト 2097
表 390. PCM/I2S モードで使用できるビットフィールド 2100
表 391. AFCNTR = 1 のとき、SPI/I2S が有効化される前の WS および CK のレベル 2108
表 392. シリアルデータラインのスワッピング 2108
表 393. 通常の I2S 周波数での CLKGEN のプログラミング例 2113
表 394. I2S 割り込みリクエスト 2122
表 395. SPI レジスタマップとリセット値 2140
表 396. SAI 内部信号 2145
表 397. SAI ピン 2145
表 398. 外部同期の選択 2147
表 399. クロックジェネレータのプログラミング例 2155
表 400. TDM 設定 2161
表 401. 許容される TDM フレーム設定 2163
表 402. SOPD パターン 2167
表 403. パリティビットの計算 2167
表 404. オーディオサンプリング周波とシンボルレート 2168
表 405. SAI 割り込みソース 2177

表 406. SAI レジスタマップとリセット値 2196
表 407. SPDIFRX 内部入力／出力信号 2199
表 408. SPDIFRX ピン 2199
表 409. プリアンブルの遷移シーケンス 2205
表 410. spdifrx_ker_ck 最小周波数とオーディオサンプリングレート 2215
表 411. spdifrx_symb_ck 生成条件 2216
表 412. ビットフィールドの特性と SPDIFRX の状態 2219
表 413. SPDIFRX インタフェースレジスタマップとリセット値 2236
表 414. パッケージのピンまたはボールに接続された SWPMI スレーブ入力／出力信号 2239
表 415. SWPMI 内部入力／出力信号 2240
表 416. 低電力モードが SWPMI に与える影響 2254
表 417. 割り込み制御ビット 2255
表 418. 送信／受信に対するバッファモードの選択 2257
表 419. SWPMI レジスタマップとリセット値 2264
表 420. パッケージのピンまたはボールに接続された MDIOS スレーブ入力／出力信号 2266
表 421. MDIOS 内部入力／出力信号 2266
表 422. 割り込み制御ビット 2272
表 423. MDIOS レジスタマップとリセット値 2279
表 424. SDMMC 動作モード SD および SDIO 2282
表 425. SDMMC 動作モード eMMC 2283
表 426. SDMMC 内部入力／出力信号 2284
表 427. SDMMC ピン群 2284
表 428. SDMMC コマンドおよびデータの位相選択 2285
表 429. コマンドトークンフォーマット 2291
表 430. CRC ありショートレスポンスのトークンフォーマット 2292
表 431. CRC なしショートレスポンスのトークンフォーマット 2292
表 432. CRC ありロングレスポンスのトークンフォーマット 2292
表 433. 固有コマンドの概要 2293
表 434. コマンドパスステータスフラグ 2293
表 435. コマンドパスエラー処理 2294
表 436. データトークンフォーマット 2301
表 437. データパスステータスフラグとクリアビット 2301
表 438. データパスエラー処理 2302
表 439. データ FIFO アクセス 2303
表 440. 送信 FIFO ステータスフラグ 2304
表 441. 受信 FIFO ステータスフラグ 2305
表 442. MDMA との SDMMC 接続 2308
表 443. AHB と SDMMC_CK のクロック周波数関係 2308
表 444. SDIO 特殊動作制御 2309
表 445. 4 ビットモードのスタート、割り込み、CRC ステータスの信号検出 2313
表 446. 使用例 2317
表 447. レスポンスタイプと SDMMC_RESPxR レジスタ 2338
表 448. SDMMC レジスタマップ 2355
表 449. CAN サブシステム I/O 信号 2357
表 450. CAN サブシステム I/O ピン 2358
表 451. FDCAN での DLC コーディング 2364
表 452. Rx バッファのフィルタ設定の例 2377
表 453. デバッグメッセージのフィルタ設定の例 2378
表 454. フレーム送信の可能な設定 2378
表 455. Tx バッファ／ FIFO - キュー要素サイズ 2379
表 456. レベル 1 リファレンスメッセージの最初のバイト 2388
表 457. レベル 2 リファレンスメッセージの最初の 4 バイト 2388

表 458. レベル 0 リファレンスメッセージの最初の 4 バイト 2389
表 459. TUR 設定の例 2390
表 460. システムマトリックス、ノード A 2394
表 461. トリガリスト、ノード A 2395
表 462. リファレンスメッセージとともに送信されるデータバイト数 2402
表 463. Rx バッファおよび FIFO 要素 2409
表 464. Rx バッファおよび FIFO 要素の説明 2409
表 465. Tx バッファおよび FIFO 要素 2411
表 466. Tx バッファ要素の説明 2411
表 467. Tx イベント FIFO 要素 2413
表 468. Tx イベント FIFO 要素の説明 2413
表 469. 標準メッセージ ID フィルタ要素 2414
表 470. 標準メッセージ ID フィルタ要素フィールドの説明 2414
表 471. 拡張メッセージ ID フィルタ要素 2415
表 472. 拡張メッセージ ID フィルタ要素フィールドの説明 2416
表 473. トリガメモリ要素 2417
表 474. トリガメモリ要素の説明 2417
表 475. FDCAN レジスタマップとリセット値 2476
表 476. CCU レジスタマップとリセット値 2487
表 477. サポートされる OTG_HS 速度 2489
表 478. STM32H7 の USB_OTG 実装 2492
表 479. OTG_FS/OTG_HS 入力／出力信号 2494
表 480. OTG_FS/OTG_HS 入力／出力ピン 2494
表 481. コアグローバル制御およびステータスレジスタ（CSR） 2513
表 482. ホストモード制御およびステータスレジスタ （CSR） 2514
表 483. デバイスモード制御およびステータスレジスタ 2515
表 484. データ FIFO（DFIFO）アクセススレジスタマップ 2517
表 485. パワーおよびクロックゲーティング制御およびステータスレジスタ 2518
表 486. TRDT 値 2526
表 487. TRDT 値（HS） 2526
表 488. ソフト切断のための最短時間 2565
表 489. OTG_HS レジスタマップとリセット値 2589
表 490. Ethernet ペリフェラルピン 2661
表 491. Ethernet 内部入力／出力信号 2662
表 492. Tx パスでのダブル VLAN 処理機能 2675
表 493. Rx パスでのダブル VLAN 処理 2676
表 494. VLTI ビットに基づいた VLAN の挿入または書き換え 2677
表 495. デスティネーションアドレスフィルタリング 2680
表 496. ソースアドレスフィルタリング 2681
表 497. VLAN 一致ステータス 2682
表 498. 通常クロック：スナップショット用 PTP メッセージ 2687
表 499. エンドツーエンドトランスペアレントクロック：スナップショット用 PTP メッセージ 2688
表 500. ピア・ツー・ピアトランスペアレントクロック：スナップショット用 PTP メッセージ 2688
表 501. PTP メッセージ生成の基準 2694
表 502. TSO：TCP ヘッダおよび IP ヘッダフィールド 2698
表 503. ポーズパケットのフィールド 2700
表 504. Tx MAC フロー制御 2701
表 505. Rx MAC フロー制御 2702
表 506. 異なるパケットタイプの送信チェックサムオフロードエンジンの機能 2704
表 507. 異なるパケットタイプの受信チェックサムオフロードエンジンの機能 2705
表 508. MCD クロックの選択 2709
表 509. MDIO パケットフィールドの説明 2709

参考資料

表 510. RX インタフェース信号のエンコーディング 2712
表 511. 転送完了割り込みの動作 2721
表 512. TDES0 通常ディスクリプタ（読み出しフォーマット） 2734
表 513. TDES1 通常ディスクリプタ（読み出しフォーマット） 2735
表 514. TDES2 通常ディスクリプタ（読み出しフォーマット） 2735
表 515. TDES3 通常ディスクリプタ（読み出しフォーマット） 2736
表 516. DES0 通常ディスクリプタ（書き戻しフォーマット） 2738
表 517. TDES1 通常ディスクリプタ（書き戻しフォーマット） 2739
表 518. TDES2 通常ディスクリプタ（書き戻しフォーマット） 2739
表 519. TDES3 通常ディスクリプタ（書き戻しフォーマット） 2739
表 520. TDES0 コンテキストディスクリプタ 2742
表 521. TDES1 コンテキストディスクリプタ 2743
表 522. TDES2 コンテキストディスクリプタ 2743
表 523. TDES3 コンテキストディスクリプタ 2743
表 524. RDES0 通常ディスクリプタ（読み出しフォーマット） 2746
表 525. RDES1 通常ディスクリプタ（読み出しフォーマット） 2746
表 526. RDES2 通常ディスクリプタ（読み出しフォーマット） 2746
表 527. RDES3 通常ディスクリプタ（読み出しフォーマット） 2747
表 528. RDES0 通常ディスクリプタ（書き戻しフォーマット） 2748
表 529. RDES1 通常ディスクリプタ（書き戻しフォーマット） 2749
表 530. RDES2 通常ディスクリプタ（書き戻しフォーマット） 2751
表 531. RDES3 通常ディスクリプタ（書き戻しフォーマット） 2752
表 532. RDES0 コンテキストディスクリプタ 2755
表 533. RDES1 コンテキストディスクリプタ 2756
表 534. RDES2 コンテキストディスクリプタ 2756
表 535. RDES3 コンテキストディスクリプタ 2756
表 536. ETH_DMA レジスタマップとリセット値 2781
表 537. ETH_DMA_CH レジスタマップとリセット値 2781
表 538. ETH_MTL レジスタマップとリセット値 2793
表 539. S2KP および JE ビットに基づいたジャイアントパケットステータス 2800
表 540. CST および ACS ビットに基づいたパケット長 2800
表 541. リモートウェイクアップパケットフィルタレジスタ 2823
表 542. ETH_MACRWKPFR 2824
表 543. リモートウェイクアップパケットおよび PMT 割り込みの生成 2826
表 544. タイムスタンプスナップショットのレジスタビットに対する依存性 2862
表 545. Ethernet MAC レジスタマップとリセット値 2882
表 546. HDMI ピン 2891
表 547. HDMI-CEC 内部入力／出力信号 2891
表 548. エラー処理タイミングパラメータ 2898
表 549. TXERR タイミングパラメータ 2899
表 550. HDMI-CEC 割り込み 2900
表 551. HDMI-CEC レジスタマップとリセット値 2909
表 552. JTAG ／シリアルワイヤデバッグポートのピン 2912
表 553. トレースポートピン 2912
表 554. シリアルワイヤトレースポートのピン 2912
表 555. トリガピン 2912
表 556. パケットリクエスト 2916
表 557. ACK 応答 2916
表 558. データ転送 2917
表 559. JTAG-DP データレジスタ 2919
表 560. デバッグポートレジスタ 2921
表 561. MEM-AP レジスタ 2932

参考資料

表 562. システム ROM テーブル 1. 2936
表 563. システム ROM テーブル 2. 2937
表 564. システム ROM テーブルレジスタのマップとリセット値 2944
表 565. TSG レジスタのマップとリセット値 2953
表 566. システム CTI 入力 2955
表 567. システム CTI 出力 2955
表 568. Cortex-M7 CTI 入力 2955
表 569. Cortex-M7 CTI 出力 2956
表 570. CTI レジスタのマップとリセット値 2971
表 571. CSTF レジスタのマップとリセット値 2982
表 572. ETF レジスタのマップとリセット値. 3004
表 573. TPIU レジスタのマップとリセット値 3021
表 574. SWO レジスタのマップとリセット値. 3033
表 575. SWTF レジスタのマップとリセット値 3043
表 576. DBGMCU レジスタのマップとリセット値 3051
表 577. Cortex-M7 CPU の ROM テーブル 3052
表 578. Cortex-M7 PPB の ROM テーブル 3053
表 579. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値 3058
表 580. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値. 3063
表 581. Cortex-M7 DWT レジスタのマップとリセット値 3075
表 582. Cortex-M7 ITM レジスタのマップとリセット値 3084
表 583. Cortex-M7 FPB レジスタのマップとリセット値. 3092
表 584. Cortex-M7 ETM レジスタのマップとリセット値 3121
表 585. 文書改版履歴 3129
表 586. 日本語版文書改版履歴. 3135

参考資料

図の一覧

図 1. STM32H7x3 デバイスのシステムアーキテクチャ 99
図 2. フラッシュメモリのブロック図 112
図 3. フラッシュメモリの概要 114
図 4. バースト読み出しタイミング図 - 4x64 ビット、遅延 = 3 ws 116
図 5. バースト読み出しタイミング図 - 8x64 ビット、遅延 = 3 ws 116
図 6. 保護の遷移スキーム 134
図 7. 標準およびセキュアアクセスモードのフラッシュメモリ領域とサービス 190
図 8. セキュアブートローダステートマシン 191
図 9. ルートセキュアサービスコール 192
図 10. フラッシュメモリ領域へのコアアクセス 196
図 11. AXI インタコネクト 198
図 12. 電源制御のブロック図 218
図 13. 電源の概要 221
図 14. システム電源設定 222
図 15. 電圧レギュレータから供給される V_{CORE} によるデバイスの起動 224
図 16. バックアップドメイン 228
図 17. USB 電源設定 229
図 18. パワーオンリセット／パワーダウンリセット波形 230
図 19. BOR の閾値 231
図 20. PVD の閾値 232
図 21. AVD の閾値 233
図 22. VBAT の閾値 234
図 23. 温度の閾値 235
図 24. V_{CORE} 電圧スケーリングとシステム電力モード 240
図 25. 電源制御モードの詳細な状態図 242
図 26. RUN モードでのダイナミック電圧スケーリングの挙動 245
図 27. STOP モードでの D1、D2、およびシステムにおけるダイナミック電圧スケーリングの挙動 246
図 28. STANDBY モードでの D1、D2、およびシステムにおけるダイナミック電圧スケーリング 247
図 29. DStandby モードでの D1 および D2、および自律モードの D3 におけるダイナミック電圧スケーリングの挙動 249
図 30. EXTI、RCC、および PWR の相互接続 272
図 31. BDMA および D3 ドメインによる自律モードでの SRAM4 から LPUART1 への転送を示すタイミング図 276
図 32. D3 ドメインによる自律モードでの LPUART1 送信を示すタイミング図 281
図 33. RCC ブロック図 284
図 34. システムリセット回路 287
図 35. ブートシーケンスとシステムの状態 292
図 36. 最上位のクロックツリー 294
図 37. HSE/LSE クロックソース 295
図 38. PLL ブロック図 302
図 39. PLL 初期化フローチャート 305
図 40. コアおよびバスクロック生成 307
図 41. SAI および DFSDM のカーネルクロック分配 313
図 42. SPI および SPI/I2S のカーネルクロック分配 314
図 43. I2C のカーネルクロック分配 315
図 44. UART、USART、および LPUART1 のカーネルクロック分配 315
図 45. LTDC のカーネルクロック分配 316
図 46. SDMMC、QUADSPI、および FMC のカーネルクロック分配 316
図 47. USB のカーネルクロック分配 (2) 317

参考資料

図 48. Ethernet のカーネルクロック分配 . . . 318
図 49. ADC、SWPMI、RNG、および FDCAN のカーネルクロック分配 (2) . . . 319
図 50. LPTIM および HDMI-CEC のカーネルクロック分配 (2) . . . 320
図 51. ペリフェラルの割り当ての例 . . . 323
図 52. カーネルクロックの切り替え . . . 326
図 53. ペリフェラルカーネルクロック有効ロジックの詳細 . . . 329
図 54. バスクロック有効ロジック . . . 334
図 55. RCC 配置の概要 . . . 336
図 56. CRS ブロック図 . . . 453
図 57. CRS カウンタの動作 . . . 455
図 58. HSEM ブロック図 . . . 465
図 59. 手順の状態図 . . . 466
図 60. 割り込みの状態図 . . . 468
図 61. I/O ポートビットの基本構成 . . . 479
図 62. 5 V トレラント I/O ポートビットの基本構成 . . . 479
図 63. 入力フローティング／プルアップ／プルダウン設定 . . . 484
図 64. 出力設定 . . . 485
図 65. オルタネート機能設定 . . . 486
図 66. 高インピーダンスアナログ設定 . . . 486
図 67. ADC 入力に接続されたアナログ入力 . . . 487
図 68. MDMA ブロック図 . . . 554
図 69. DMA ブロック図 . . . 583
図 70. “ ペリフェラルからメモリ ” モード . . . 586
図 71. “ メモリからペリフェラル ” モード . . . 587
図 72. メモリ間モード . . . 588
図 73. FIFO の構造 . . . 593
図 74. BDMA ブロック図 . . . 615
図 75. DMAMUX ブロック図 . . . 634
図 76. DMAMUX リクエストラインマルチプレクサチャネルの同期モード . . . 637
図 77. DMA リクエストラインマルチプレクサチャネルのイベント生成 . . . 637
図 78. DMA2D ブロック図 . . . 652
図 79. EXTI ブロック図 . . . 692
図 80. 設定可能なイベントのトリガロジック、CPU ウェイクアップ . . . 694
図 81. 設定可能なイベントのトリガロジック、任意のウェイクアップ . . . 696
図 82. ダイレクトイベントのトリガロジック、CPU ウェイクアップ . . . 697
図 83. ダイレクトイベントのトリガロジック、任意のウェイクアップ . . . 698
図 84. D3 ドメインペンディングリクエストクリアロジック . . . 699
図 85. CRC 計算ユニットのブロック図 . . . 724
図 86. FMC ブロック図 . . . 732
図 87. FMC メモリバンク . . . 735
図 88. モード 1 読み出しアクセス波形 . . . 746
図 89. モード 1 書き込みアクセス波形 . . . 747
図 90. モード A 読み出しアクセス波形 . . . 749
図 91. モード A 書き込みアクセス波形 . . . 750
図 92. モード 2 およびモード B 読み出しアクセス波形 . . . 752
図 93. モード 2 書き込みアクセス波形 . . . 752
図 94. モード B 書き込みアクセス波形 . . . 753
図 95. モード C 読み出しアクセス波形 . . . 755
図 96. モード C 書き込みアクセス波形 . . . 755
図 97. モード D 読み出しアクセス波形 . . . 758
図 98. モード D 書き込みアクセス波形 . . . 758
図 99. マルチプレクス読み出しアクセス波形 . . . 761

図 100. マルチプレクス書き込みアクセス波形 761
図 101. 読み出しアクセス中の非同期ウェイト波形 764
図 102. 書き込みアクセス中の非同期ウェイト波形 764
図 103. ウェイト設定波形 766
図 104. 同期マルチプレクス読み出しモード波形 - NOR、PSRAM（CRAM） 767
図 105. 同期マルチプレクス書き込みモード波形 - PSRAM（CRAM） 769
図 106. 共通メモリアクセスの NAND フラッシュコントローラ波形 782
図 107. 'CE don't care' の NAND フラッシュへのアクセス 784
図 108. バースト書き込み SDRAM アクセス波形 794
図 109. バースト読み出し SDRAM アクセス 795
図 110. RBURST ビットセット時の読み出しアクセスの論理図（CAS=2、RPIPE=0） 796
図 111. Row 境界を跨ぐ読み出しアクセス 798
図 112. Row 境界を跨ぐ書き込みアクセス 798
図 113. セルフリフレッシュモード 801
図 114. パワーダウンモード 802
図 115. QUADSPI ブロック図（デュアルフラッシュモードが無効のとき） 813
図 116. QUADSPI ブロック図（デュアルフラッシュモードが有効のとき） 813
図 117. クワッドモードにおける読み出しコマンドの例 815
図 118. クワッドモードにおける DDR コマンドの例 818
図 119. nCS - CKMODE = 0（T = CLK 周期）のとき 827
図 120. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき 827
図 121. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき 827
図 122. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき 828
図 123. DLYB ブロック図 842
図 124. ADC ブロック図 850
図 125. ADC クロック構成 853
図 126. ADC1 接続性 854
図 127. ADC2 接続性 855
図 128. ADC3 接続性 856
図 129. ADC 較正 860
図 130. ADC オフセット較正係数の更新 860
図 131. シングルエンドチャネルと差動チャネルの混合 861
図 132. ADC の有効化／無効化 864
図 133. アナログ／デジタル変換時間 870
図 134. 実行中のレギュラ変換の停止 871
図 135. 実行中のレギュラ変換とインジェクト変換の停止 871
図 136. ADC マスタと ADC スレーブ間でトリガを共有 873
図 137. インジェクト変換の遅延時間 876
図 138. コンテキストの JSQR キューの例（シーケンスの変更） 879
図 139. コンテキストの JSQR キューの例（トリガの変更） 880
図 140. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例 880
図 141. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例 881
図 142. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合） 881
図 143. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合） 882
図 144. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に JADSTP が発生した場合 882
図 145. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に JADSTP が発生し、新しいトリガが発生した場合 883
図 146. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に外部で JADSTP が発生した場合 883
図 147. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1） 884
図 148. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0） 884

図 149. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1）.....885
図 150. シーケンスのシングル変換、ソフトウェアトリガ.....887
図 151. シーケンスの連続変換、ソフトウェアトリガ.....887
図 152. シーケンスのシングル変換、ハードウェアトリガ.....888
図 153. シーケンスの連続変換、ハードウェアトリガ.....888
図 154. 右詰め（オフセット無効、符号なしの値）.....890
図 155. 右詰め（オフセット有効、符号付きの値）.....890
図 156. 左詰め（オフセット無効、符号なしの値）.....891
図 157. 左詰め（オフセット有効、符号付きの値）.....891
図 158. オーバーラン（OVR）の例.....893
図 159. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ.....896
図 160. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=0、JDISCEN=0）.....897
図 161. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=1、JDISCEN=1）.....898
図 162. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換.....899
図 163. 自動インジェクトモードの AUTDLY=1 （JAUTO=1）.....899
図 164. アナログウォッチドッグによって保護される領域.....900
図 165. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）.....902
図 166. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合）.....902
図 167. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）.....903
図 168. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル）.....903
図 169. 10 ビット右シフトと丸めを行う 16 ビット結果のオーバーサンプリング.....904
図 170. トリガレギュラオーバーサンプリングモード（TROVS ビット =1）.....906
図 171. レギュラオーバーサンプリングモード（4x の比率）.....907
図 172. レギュラおよびインジェクトオーバーサンプリングモードの同時使用.....907
図 173. インジェクションでのトリガレギュラオーバーサンプリング.....908
図 174. 自動インジェクトモードでのオーバーサンプリング.....908
図 175. デュアル ADC ブロック図 (1).....911
図 176. 4 チャネルのインジェクト同時モード：デュアル ADC モード.....912
図 177. 16 チャネルのレギュラ同時モード：デュアル ADC モード.....914
図 178. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード.....915
図 179. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード.....916
図 180. インジェクションによるインタリーブ変換.....916
図 181. オルタネートトリガ：各 ADC のインジェクトグループ.....917
図 182. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）.....918
図 183. オルタネート + レギュラ同時.....919
図 184. インジェクト変換中にトリガが発生した場合.....919
図 185. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0.....920
図 186. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割り込み優先.....920
図 187. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割り込み優先.....920
図 188. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b00）.....921
図 189. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b10）.....922
図 190. インタリーブモードにおける DMA リクエスト（DAMDF=0b10）.....922
図 191. 温度センサチャネルのブロック図.....924
図 192. VBAT チャネルのブロック図.....926
図 193. VREFINT チャネルのブロック図.....926
図 194. DAC チャネルブロック図.....973
図 195. シングル DAC チャネルモードのデータレジスタ.....975
図 196. デュアル DAC チャネルモードのデータレジスタ.....976

図 197. トリガ無効（TEN = 0）時の変換タイミング図 976
図 198. DAC LFSR レジスタ計算アルゴリズム 978
図 199. LFSR 波形生成による DAC 変換（SW トリガ有効） 979
図 200. DAC 三角波生成 979
図 201. 三角波生成による DAC 変換（SW トリガ有効） 980
図 202. DAC サンプルおよびホールドモードフェーズの図 982
図 203. コンパレータの機能ブロック図 1011
図 204. コンパレータヒステリシス 1014
図 205. コンパレータ出力のブランキング 1014
図 206. 出力先変更 1016
図 207. スケーラブロック図 1018
図 208. スタンドアロンモード：外部ゲイン設定モード 1029
図 209. フォロア設定 1030
図 210. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力 1031
図 211. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力 1032
図 212. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）または反転ゲイン設定
（x-1/x-3/x-7/x-15） 1033
図 213. 設定例 1033
図 214. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）またはフィルタリング付き反転ゲイン設定
（x-1/x-3/x-7/x-15） 1034
図 215. 設定例 1034
図 216. 1 つの DFSDM ブロック図 1048
図 217. 入力チャネルピンの変更 1052
図 218. チャネルトランシーバのタイミング図 1054
図 219. SPI でのクロック無タイミング検出図 1055
図 220. マンチェスタ符号化でのクロック無タイミング検出図 1056
図 221. マンチェスタ符号化の最初の変換（マンチェスタ同期） 1058
図 222. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て 1062
図 223. 例：Sinc3 フィルタレスポンス： 1064
図 224. DCMI のブロック図 1104
図 225. 最上位ブロック図 1104
図 226. DCMI 信号の波形 1106
図 227. タイミング図 1108
図 228. スナップショットモードにおけるフレームキャプチャ波形 1110
図 229. 連続グラブモードにおけるフレームキャプチャ波形 1111
図 230. クロップ後のウィンドウの座標とサイズ 1111
図 231. データキャプチャ波形 1112
図 232. ピクセルラスタースキャンの順序 1113
図 233. LTDC のブロック図 1130
図 234. LCD-TFT 同期タイミング 1133
図 235. レイヤウィンドウのプログラム可能なパラメータ 1136
図 236. 2 つのレイヤとバックグラウンドとのブレンディング 1139
図 237. 割り込みイベント 1140
図 238. JPEG コーデックブロック図 1164
図 239. RNG のブロック図 1179
図 240. エントロピーソースモデル 1180
図 241. ランダムサンプル条件付けプロセス 1182
図 242. RNG 初期化の概要 1184
図 243. CRYP ブロック図 1194
図 244. AES-ECB モードの概要 1197
図 245. AES-CBC モードの概要 1198
図 246. AES-CTR モードの概要 1199

図 247. AES-GCM モードの概要 1200
図 248. AES-GMAC モードの概要 1200
図 249. AES-CCM モードの概要 1201
図 250. STM32 暗号ライブラリ DES/ トリプル DES フローチャート 1202
図 251. STM32 暗号ライブラリ AES フローチャート例 1203
図 252. サスペンドモードの管理例 1208
図 253. DES ／トリプル DES-ECB モードの暗号化 1209
図 254. DES ／トリプル DES-ECB モードの復号化 1210
図 255. DES ／トリプル DES-CBC モードの暗号化 1211
図 256. DES ／トリプル DES-CBC モードの復号化 1212
図 257. AES-ECB モードの暗号化 1214
図 258. AES-ECB モードの復号化 1215
図 259. AES-CBC モードの暗号化 1216
図 260. AES-CBC モードの復号化 1217
図 261. カウンタモードのメッセージ構造 1219
図 262. AES-CTR モードの暗号化 1220
図 263. AES-CTR モードの復号化 1221
図 264. ガロア／カウンタモードのメッセージ構造 1223
図 265. ガロアメッセージ認証コードモードのメッセージ構造 1228
図 266. CBC-MAC モードのメッセージ構造 1229
図 267. データ型に応じた 64 ビットブロックの構成（IN FIFO） 1235
図 268. データ型に応じた 128 ビットブロックの構成 1237
図 269. CRYP 割り込みマッピング図 1242
図 270. HASH ブロック図 1263
図 271. メッセージデータスワッピング機能 1265
図 272. HASH 保存／復元のメカニズム 1270
図 273. HASH 割り込みマッピング図 1272
図 274. 高分解能タイマのブロック図 1288
図 275. タイマ A ～ E の概要 1292
図 276. 連続タイマ動作 1293
図 277. シングルショットタイマ動作 1294
図 278. タイマリセットの再同期（32 を超えるプリスケーリング比） 1295
図 279. 連続モードでの繰り返しレート 対 HRTIM_REPxR の内容 1296
図 280. シングルショットモードでの繰り返しカウンタの動作 1297
図 281. 比較イベントの出力への作用：比較 1 でセット、比較 2 でリセット 1298
図 282. タイミングユニットのキャプチャ回路 1300
図 283. 自動遅延の概要（比較 2 のみ） 1301
図 284. 自動遅延比較 1302
図 285. プッシュプルモードのブロック図 1304
図 286. プッシュプルモードの例 1305
図 287. デッドタイム挿入のある相補出力 1305
図 288. デッドタイム挿入 対 デッドタイムの符号（1 は負のデッドタイムを示す） 1306
図 289. 小パルス幅に対する相補出力（SDTRx = SDTFx = 0） 1307
図 290. 小パルス幅に対する相補出力（SDTRx = SDTFx = 1） 1307
図 291. 小パルス幅に対する相補出力（SDTRx = 0、SDTFx = 1） 1307
図 292. 小パルス幅に対する相補出力（SDTRx = 1、SDTFx = 0） 1308
図 293. マスタタイマの概要 1309
図 294. 外部イベント調整の概要（代表 1 チャネル） 1311
図 295. 外部イベントの立ち下がりエッジの遅延時間（カウンタリセットおよび出力セット） 1314
図 296. 外部イベントの遅延時間（外部イベントによる出力リセット） 1314
図 297. イベントブランキングモード 1315
図 298. イベント延期モード 1315

図 299. エッジ対応トリガによる外部トリガブランキング 1317
図 300. レベル対応トリガによる外部トリガブランキング 1317
図 301. イベントウインドウイングモード 1318
図 302. エッジ対応トリガによる外部トリガウィンドウイング 1319
図 303. レベル対応トリガによる外部トリガウィンドウイング 1319
図 304. 遅延アイドルモードへの移行 1321
図 305. バーストモードおよび遅延保護優先順位（DIDL = 0） 1322
図 306. バーストモードおよび遅延保護優先順位（DIDL = 1） 1323
図 307. バランスアイドル保護の例 1324
図 308. 出力管理の概要 1334
図 309. HRTIM 出力状態と遷移 1334
図 310. バーストモードの動作例 1336
図 311. 外部イベントによるバーストモードのトリガ 1338
図 312. デッドタイムを有効にして IDLESx = 1 の場合の遅延バーストモードへの移行 1340
図 313. デッドタイム中の遅延バーストモードへの移行 1341
図 314. デッドタイムジェネレータが有効なときのバーストモードの終了 1342
図 315. バーストモードのエミュレーション例 1344
図 316. キャリア周波数信号の挿入 1344
図 317. チョッパモードが有効化された HRTIM 出力 1345
図 318. フォールト保護回路（FAULT1 は完全表示、FAULT2 ～ 5 は部分表示） 1346
図 319. フォールト信号のフィルタリング（FLTxF[3:0]= 0010：$f_{SAMPLING}$ = fHRTIM、N = 4） 1347
図 320. 補助出力 1349
図 321. バーストモード中の補助およびメイン出力（DIDLx = 0） 1350
図 322. バーストモードを出る時の補助出力上のデッドタイムの歪み 1350
図 323. 同期開始モードでのカウンタの動作 1353
図 324. ADC トリガ選択の概要 1354
図 325. 単一の hrtim_dac_trgx 出力への複数の更新の組み合わせ 1355
図 326. DMA バーストの概要 1359
図 327. バースト DMA の動作フローチャート 1360
図 328. DMA バースト転送後のレジスタ更新 1361
図 329. 降圧コンバータの接続形態 1363
図 330. デュアル降圧コンバータのマネージメント 1364
図 331. 出力電流に応じた同期整流 1364
図 332. 同期整流式の降圧 1365
図 333. 3 相インターリーブ降圧コンバータ 1365
図 334. 3 相インターリーブ降圧コンバータの制御 1366
図 335. 遷移モード PFC 1367
図 336. 遷移モード PFC の波形 1368
図 337. 高機能制御タイマのブロック図 1457
図 338. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1459
図 339. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1459
図 340. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1461
図 341. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1461
図 342. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1462
図 343. 内部クロック分周比が N の場合のカウンタのタイミング図 1462
図 344. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1463
図 345. ARPE = 1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1463
図 346. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1465
図 347. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1465
図 348. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1466

図 349. 内部クロック分周比が N の場合のカウンタのタイミング図 1466
図 350. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 1467
図 351. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図 1468
図 352. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1469
図 353. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図 1469
図 354. 内部クロック分周比が N の場合のカウンタのタイミング図 1470
図 355. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図 1470
図 356. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 1471
図 357. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1472
図 358. 外部トリガ入力ブロック 1473
図 359. TIM1/TIM8 ETR 入力回路 1473
図 360. 内部クロック分周比 1 の場合の、通常モードの制御回路 1474
図 361. TI2 外部クロックの接続例 1475
図 362. 外部クロックモード 1 の制御回路 1476
図 363. 外部トリガ入力ブロック 1476
図 364. 外部クロックモード 2 の制御回路 1477
図 365. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1478
図 366. キャプチャ／比較チャネル 1 メイン回路 1479
図 367. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ 1479
図 368. キャプチャ／比較チャネル（チャネル 4）の出力ステージ 1480
図 369. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ 1480
図 370. PWM 入力モードタイミング 1482
図 371. 出力比較モード、OC1 の反転 1484
図 372. エッジアライン PWM 波形（ARR=8） 1486
図 373. センターアライン PWM 波形（ARR=8） 1487
図 374. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 1488
図 375. チャネル 1 および 3 における組み合わせ PWM モード 1489
図 376. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号 1490
図 377. デッドタイム挿入のある相補出力 1491
図 378. 負のパルスより長い遅延があるときのデッドタイムの波形 1492
図 379. 正のパルスより長い遅延があるときのデッドタイムの波形 1492
図 380. ブレーク および ブレーク 2 回路の概要 1495
図 381. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作 1497
図 382. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1） 1498
図 383. BRK アサート後の PWM 出力状態（OSSI=0） 1499
図 384. 出力先変更 1499
図 385. TIMx OCxREF のクリア 1500
図 386. 6 ステップ生成 COM の例（OSSR=1） 1501
図 387. ワンパルスモードの例 1502
図 388. 再トリガ可能なワンパルスモード 1504
図 389. エンコーダインタフェースモードにおけるカウンタの動作例 1505
図 390. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 1506
図 391. 3 つの信号上のエッジ間の時間間隔の測定 1507
図 392. ホールセンサインタフェースの例 1509
図 393. リセットモードの制御回路 1510
図 394. ゲートモードの制御回路 1511
図 395. トリガモードの制御回路 1512
図 396. 外部クロックモード 2 ＋トリガモードの制御回路 1513
図 397. 汎用タイマのブロック図 1564
図 398. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1566
図 399. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1566
図 400. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1567

図 401. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1568
図 402. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1568
図 403. 内部クロック分周比が N の場合のカウンタのタイミング図 1569
図 404. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1569
図 405. ARPE=1（TIMx_ARR はプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1570
図 406. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1571
図 407. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1571
図 408. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1572
図 409. 内部クロック分周比が N の場合のカウンタのタイミング図 1572
図 410. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 1573
図 411. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 1574
図 412. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1575
図 413. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図 1575
図 414. 内部クロック分周比が N の場合のカウンタのタイミング図 1576
図 415. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図 1576
図 416. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 1577
図 417. 内部クロック分周比 1 の場合の、通常モードの制御回路 1578
図 418. TI2 外部クロックの接続例 1578
図 419. 外部クロックモード 1 の制御回路 1579
図 420. 外部トリガ入力ブロック 1580
図 421. 外部クロックモード 2 の制御回路 1580
図 422. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1581
図 423. キャプチャ／比較チャネル 1 メイン回路 1582
図 424. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1582
図 425. PWM 入力モードタイミング 1584
図 426. 出力比較モード、OC1 の反転 1586
図 427. エッジアライン PWM 波形（ARR=8） 1587
図 428. センターアライン PWM 波形（ARR=8） 1589
図 429. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 1590
図 430. チャネル 1 および 3 における組み合わせ PWM モード 1591
図 431. TIMx OCxREF のクリア 1592
図 432. ワンパルスモードの例 1593
図 433. 再トリガ可能なワンパルスモード 1595
図 434. エンコーダインタフェースモードにおけるカウンタの動作例 1596
図 435. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 1597
図 436. リセットモードの制御回路 1598
図 437. ゲートモードの制御回路 1599
図 438. トリガモードの制御回路 1600
図 439. 外部クロックモード 2 ＋トリガモードの制御回路 1601
図 440. マスタ／スレーブタイマの例 1602
図 441. TIM3 の OC1REF による TIM2 のゲート操作 1603
図 442. TIM3 の有効化による TIM2 のゲート操作 1604
図 443. TIM3 の更新による TIM2 のトリガ 1604
図 444. TIM3 の有効化による TIM2 のトリガ 1605
図 445. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。 1606
図 446. 汎用タイマのブロック図（TIM12） 1638
図 447. 汎用タイマのブロック図（TIM13/TIM14） 1639
図 448. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1641
図 449. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1641
図 450. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1642

図 451. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1643
図 452. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1643
図 453. 内部クロック分周比が N の場合のカウンタのタイミング図 1644
図 454. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1644
図 455. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1645
図 456. 内部クロック分周比 1 の場合の、通常モードの制御回路 1646
図 457. TI2 外部クロックの接続例 1646
図 458. 外部クロックモード 1 の制御回路 1647
図 459. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1648
図 460. キャプチャ／比較チャネル 1 メイン回路 1648
図 461. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1649
図 462. PWM 入力モードタイミング 1651
図 463. 出力比較モード、OC1 の反転 1653
図 464. エッジアライン PWM 波形（ARR=8） 1654
図 465. チャネル 1 および 2 における組み合わせ PWM モード 1655
図 466. ワンパルスモードの例 1656
図 467. 再トリガ可能なワンパルスモード 1657
図 468. 2 つの信号上のエッジ間の時間間隔の測定 1658
図 469. リセットモードの制御回路 1659
図 470. ゲートモードの制御回路 1660
図 471. トリガモードの制御回路 1660
図 472. TIM15 ブロック図 1691
図 473. TIM16/TIM17 ブロック図 1692
図 474. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1694
図 475. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1694
図 476. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1696
図 477. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1696
図 478. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1697
図 479. 内部クロック分周比が N の場合のカウンタのタイミング図 1697
図 480. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1698
図 481. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1698
図 482. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1700
図 483. 内部クロック分周比 1 の場合の、通常モードの制御回路 1701
図 484. TI2 外部クロックの接続例 1701
図 485. 外部クロックモード 1 の制御回路 1702
図 486. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1703
図 487. キャプチャ／比較チャネル 1 メイン回路 1703
図 488. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1704
図 489. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ 1704
図 490. PWM 入力モードタイミング 1706
図 491. 出力比較モード、OC1 の反転 1708
図 492. エッジアライン PWM 波形（ARR=8） 1709
図 493. チャネル 1 および 2 における組み合わせ PWM モード 1711
図 494. デッドタイム挿入のある相補出力 1712
図 495. 負のパルスより長い遅延があるときのデッドタイムの波形 1712
図 496. 正のパルスより長い遅延があるときのデッドタイムの波形 1713
図 497. ブレーク回路の概要 1715
図 498. ブレークに対する出力の動作 1717

図 499. ワンパルスモードの例 1718
図 500. 2 つの信号上のエッジ間の時間間隔の測定 1720
図 501. リセットモードの制御回路 1721
図 502. ゲートモードの制御回路 1722
図 503. トリガモードの制御回路 1723
図 504. 基本タイマブロック図 1773
図 505. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1775
図 506. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1775
図 507. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1776
図 508. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1777
図 509. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1777
図 510. 内部クロック分周比が N の場合のカウンタのタイミング図 1778
図 511. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1778
図 512. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1779
図 513. 内部クロック分周比 1 の場合の、通常モードの制御回路 1780
図 514. 低電力タイマのブロック図（LPTIM1 と LPTIM2） 1787
図 515. 低電力タイマのブロック図（LPTIM3） 1788
図 516. 低電力タイマのブロック図（LPTIM4 と LPTIM5） 1788
図 517. グリッチフィルタのタイミング図 1793
図 518. LPTIM 出力波形、シングルカウントモードの設定 1794
図 519. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット） 1795
図 520. LPTIM 出力波形、連続カウントモードの設定 1795
図 521. 波形生成 1797
図 522. エンコーダモードのカウントシーケンス 1800
図 523. ウォッチドッグのブロック図 1815
図 524. ウィンドウ型ウォッチドッグのタイミング図 1817
図 525. 独立型ウォッチドッグのブロック図 1821
図 526. RTC ブロックの概要 1831
図 527. 詳細な RTC ブロック図 1832
図 528. タンパ検出 1833
図 529. I2C ブロック図 1878
図 530. I2C バスプロトコル 1880
図 531. セットアップおよびホールドタイミング 1882
図 532. I2C 初期化フローチャート 1885
図 533. データ受信 1886
図 534. データ送信 1887
図 535. スレーブ初期化フローチャート 1890
図 536. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=0） 1892
図 537. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=1） 1893
図 538. I2C スレーブトランスミッタの転送バス図 1894
図 539. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0） 1895
図 540. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1） 1896
図 541. I2C スレーブレシーバの転送バス図 1896
図 542. マスタクロック生成 1898
図 543. マスタ初期化フローチャート 1900
図 544. HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス 1900
図 545. HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス 1901
図 546. N≤255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート 1902
図 547. N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート 1903

図 548. I2C マスタトランスミッタの転送バス図 1904
図 549. N≤255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート 1906
図 550. N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート 1907
図 551. I2C マスタレシーバの転送バス図 1908
図 552. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔 1913
図 553. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート 1917
図 554. SMBus スレーブトランスミッタの転送バス図（SBC=1） 1917
図 555. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート 1919
図 556. SMBus スレーブレシーバのバス転送図（SBC=1） 1920
図 557. SMBus マスタトランスミッタのバス転送図 1921
図 558. SMBus マスタレシーバのバス転送図 1923
図 559. I2C 割り込みマッピング図 1929
図 560. USART のブロック図 1949
図 561. ワード長のプログラミング 1952
図 562. 設定可能なストップビット 1954
図 563. 送信時の TC/TXE の動作 1956
図 564. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出 1957
図 565. usart_ker_ck クロック分周回路のブロック図 1961
図 566. データサンプリング（16 倍のオーバーサンプリング） 1962
図 567. データサンプリング（8 倍のオーバーサンプリング） 1962
図 568. アイドルライン検出を使用したミュートモード 1968
図 569. アドレスマーク検出を使用したミュートモード 1969
図 570. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1） 1972
図 571. LIN モードでのブレーク検出とフレーミングエラー検出 1973
図 572. USART の同期マスタ送信の例 1974
図 573. 同期マスタモードでの USART データクロックタイミング図（M ビット =“00”） 1974
図 574. 同期マスタモードでの USART データクロックタイミング図（M ビット =“01”） 1975
図 575. 同期スレーブモードでの USART データクロックタイミング図（M ビット =“00”） 1976
図 576. ISO 7816-3 非同期プロトコル 1978
図 577. ストップビット 1.5 個を使用したパリティエラー検出 1980
図 578. IrDA SIR ENDEC ブロック図 1983
図 579. IrDA データ変調（3/16）- 通常モード 1984
図 580. DMA を使用した送信 1985
図 581. DMA を使用した受信 1986
図 582. 2 つの USART 間のハードウェアフロー制御 1986
図 583. RS232 RTS フロー制御 1987
図 584. RS232 CTS フロー制御 1988
図 585. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効） 1990
図 586. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効） 1991
図 587. LPUART ブロック図 2023
図 588. LPUART ワード長のプログラミング 2025
図 589. 設定可能なストップビット 2027
図 590. 送信時の TC/TXE の動作 2029
図 591. lpuart_ker_ck クロック分周回路のブロック図 2032
図 592. アイドルライン検出を使用したミュートモード 2036
図 593. アドレスマーク検出を使用したミュートモード 2037
図 594. DMA を使用した送信 2040
図 595. DMA を使用した受信 2041
図 596. 2 つの LPUART 間のハードウェアフロー制御 2042

図 597. RS232 RTS フロー制御 2042
図 598. RS232 CTS フロー制御 2043
図 599. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効） 2046
図 600. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効） 2046
図 601. SPI2S ブロック図 2069
図 602. 全二重シングルマスタ／シングルスレーブアプリケーション 2071
図 603. 半二重シングルマスタ／シングルスレーブアプリケーション 2071
図 604. 単方向シングルマスタ／シングルスレーブアプリケーション（送信専用モードのマスタ／受信専用モードのスレーブ） 2072
図 605. スター接続形態のマスタと 3 つの独立したスレーブ 2073
図 606. サーキュラ（デイジーチェーン）接続形態のマスタと 3 つのスレーブ 2075
図 607. マルチマスタアプリケーション 2076
図 608. SS 制御ロジックの仕組み 2078
図 609. データフロータイミング制御（SSOE=1、SSOM=0、SSM=0） 2078
図 610. SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0） 2079
図 611. データクロックのタイミング図 2081
図 612. データサイズが 8 ビット、16 ビット、または 32 ビットと等しくない場合のデータ配置 2082
図 613. 送受信用 FIFO でのデータのパッキング 2089
図 614. TI モードでの転送 2091
図 615. 低電力モードでのアプリケーションの例 2096
図 616. 波形の例 2102
図 617. マスタ I2S フィリップスプロトコルの波形（16/32 ビットフル精度） 2103
図 618. フィリップス I2S 規格の波形 2103
図 619. マスタ MSB 詰め 16 ビットまたは 32 ビット長フル精度 2104
図 620. マスタ MSB 詰め 16 ビットまたは 24 ビットデータ長 2104
図 621. スレーブ MSB 詰め 2105
図 622. LSB 詰め 16 ビットまたは 24 ビットデータ長 2105
図 623. フレーム長がデータ長と等しいときのマスタ PCM 2106
図 624. マスタ PCM 規格の波形（16 ビットまたは 24 ビットデータ長） 2106
図 625. スレーブ PCM 波形 2107
図 626. フィリップス I2S 規格、マスタの起動シーケンス 2109
図 627. フィリップス I2S 規格、スレーブの起動シーケンス 2110
図 628. フィリップス I2S 規格、マスタの停止シーケンス 2111
図 629. I^2S クロックジェネレータのアーキテクチャ 2111
図 630. データフォーマット 2114
図 631. アンダーラン状況の取り扱い 2115
図 632. オーバーラン状況の取り扱い 2116
図 633. FIXCH=0 でのフレームエラー検出 2117
図 634. FIXCH=1 でのフレームエラー検出 2117
図 635. 機能ブロック図 2144
図 636. オーディオフレーム 2148
図 637. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1） 2150
図 638. FS の機能はフレーム開始（FSDEF = 0） 2151
図 639. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合） 2152
図 640. 先頭ビットオフセット 2152
図 641. オーディオブロッククロックジェネレータの概要 2153
図 642. PDM 標準接続とタイミング 2157
図 643. PDM インタフェースの詳細ブロック図 2158
図 644. 起動シーケンス 2159
図 645. TDM の SAI_ADR フォーマット（32 ビットスロット幅） 2160

参考資料

図 646. TDM の SAI_ADR フォーマット（16 ビットスロット幅）.....2160
図 647. TDM の SAI_ADR フォーマット（8 ビットスロット幅）.....2161
図 648. AC’97 オーディオフレーム.....2164
図 649. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC’97 設定の代表例.....2165
図 650. SPDIF フォーマット.....2166
図 651. SAI_xDR レジスタの順序.....2167
図 652. SAI オーディオブロックのデータコンパンディングハードウェア.....2170
図 653. 非アクティブスロットにおける SD 出力ラインのトライステート化.....2171
図 654. I2S のようなプロトコルにおける出力データラインのトライステート化.....2172
図 655. オーバーラン検出エラー.....2173
図 656. FIFO アンダーランイベント.....2174
図 657. SPDIFRX ブロック図.....2199
図 658. S/PDIF サブフレームフォーマット.....2200
図 659. S/PDIF ブロックフォーマット.....2200
図 660. S/PDIF プリアンブル.....2201
図 661. チャネルのコード化の例.....2202
図 662. SPDIFRX デコーダ.....2203
図 663. ノイズフィルタリングおよびエッジ検出.....2203
図 664. 閾値.....2205
図 665. 同期のフローチャート.....2207
図 666. 同期プロセスのスケジューリング.....2208
図 667. SPDIFRX の状態.....2209
図 668. SPDIFRX_DR レジスタ フォーマット.....2211
図 669. チャネル／ユーザデータフォーマット.....2212
図 670. S/PDIF オーバーランエラー（RXSTEO = 0 の場合）.....2214
図 671. S/PDIF オーバーランエラー（RXSTEO = 1 の場合）.....2215
図 672. SPDIFRX インタフェース割り込みマッピング図.....2218
図 673. S1 信号コーディング.....2237
図 674. S2 信号コーディング.....2237
図 675. SWPMI ブロック図.....2239
図 676. SWP バス状態.....2242
図 677. SWP フレーム構造.....2243
図 678. SWPMI ソフトウェアバッファモードなし送信.....2244
図 679. SWPMI ソフトウェアバッファなしモード送信、連続フレーム.....2245
図 680. SWPMI マルチソフトウェアバッファモード送信.....2247
図 681. SWPMI ソフトウェアバッファなしモード受信.....2249
図 682. SWPMI シングルソフトウェアバッファモード受信.....2250
図 683. SWPMI マルチソフトウェアバッファモード受信.....2252
図 684. SWPMI CRC エラーありのシングルバッファモード受信.....2253
図 685. MDIOS ブロック図.....2266
図 686. MDIO プロトコルのライトフレーム波形.....2267
図 687. MDIO プロトコルのリードフレーム波形.....2268
図 688. SDMMC「no response」と「no data」操作.....2281
図 689. SDMMC（複数）ブロック読み出し操作.....2281
図 690. SDMMC（複数）ブロック書き込み操作.....2281
図 691. SDMMC（シーケンシャル）ストリーム読み出し操作.....2282
図 692. SDMMC（シーケンシャル）ストリーム書き込み操作.....2282
図 693. SDMMC ブロック図.....2283
図 694. SDMMC コマンドおよびデータの位相関係.....2285
図 695. 制御ユニット.....2287
図 696. コマンド／レスポンスパス.....2288

図 697. コマンドパスステートマシン（CPSM）.....2289
図 698. データパス.....2294
図 699. DDR モードのデータパケットクロック供給.....2295
図 700. DDR モードの CRC ステータス／ブート確認応答クロック供給.....2295
図 701. データパスステートマシン（DPSM）.....2296
図 702. CLKMUX ユニット.....2306
図 703. 非同期割り込みの生成.....2310
図 704. 同期割り込み期間のデータ読み出し.....2311
図 705. 同期割り込み期間のデータ書き込み.....2311
図 706. 非同期割り込み期間のデータ読み出し.....2312
図 707. 非同期割り込み期間のデータ書き込み.....2312
図 708. DS、ハイスピード、SDR12、SDR25 向け SDMMC_CK でのクロック停止.....2315
図 709. DDR50、SDR50、SDR104 向け SDMMC_CK でのクロック停止.....2315
図 710. 50 MHz 未満の SDMMC_CK での ReadWait.....2316
図 711. 50 MHz 以上の SDMMC_CK での ReadWait.....2316
図 712. CMD12 ストリームタイミング.....2318
図 713. CMD5 SLEEP AWAKE 手順.....2320
図 714. 通常ブートモード動作.....2322
図 715. 代替ブートモード動作.....2324
図 716. コマンドレスポンス R1b ビジー信号.....2325
図 717. SDMMC の状態制御.....2326
図 718. カードサイクルパワー／パワーアップの図.....2326
図 719. ハードウェアフロータイミング.....2327
図 720. CMD11 信号電圧スイッチシーケンス開始.....2328
図 721. 電圧スイッチトランシーバの標準アプリケーション.....2329
図 722. CAN サブシステム.....2359
図 723. FDCAN ブロック図.....2361
図 724. トランシーバ遅延測定.....2366
図 725. バスモニタモードでのピン制御.....2367
図 726. ループバックモードでのピン制御.....2369
図 727. メッセージ RAM 設定.....2371
図 728. 標準メッセージ ID フィルタパス.....2374
図 729. 拡張メッセージ ID フィルタパス.....2375
図 730. 混合設定専用 Tx バッファ／Tx FIFO の例.....2381
図 731. 混合設定専用 Tx バッファ／Tx キューの例.....2381
図 732. バイパス操作.....2383
図 733. FSM 較正.....2385
図 734. サイクルタイムとグローバルタイムの同期.....2399
図 735. TTCAN レベル 0 およびレベル 2 のドリフト補正.....2400
図 736. レベル 0 スケジュール同期ステートマシン.....2407
図 737. レベル 0 スレーブに対するマスタの関係.....2408
図 738. OTG ハイスピードブロック図（OTG_HS1）.....2493
図 739. OTG ハイスピードブロック図（OTG_HS2）.....2493
図 740. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力）.....2505
図 741. OTG_HFIR の動的更新.....2507
図 742. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング.....2508
図 743. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング.....2509
図 744. 割り込みの階層.....2512
図 745. 送信 FIFO 書き込みタスク.....2603
図 746. 受信 FIFO 読み出しタスク.....2604
図 747. 通常のバルク／コントロール OUT/SETUP.....2605
図 748. バルク／コントロール IN トランザクション.....2609

図 749. 通常割り込み OUT 2611
図 750. 通常割り込み IN 2616
図 751. アイソクロナス OUT トランザクション 2618
図 752. アイソクロナス IN トランザクション 2621
図 753. 通常のバルク／制御 OUT/SETUP トランザクション - DMA 2623
図 754. 通常のバルク／制御 IN トランザクション - DMA 2625
図 755. 通常の割り込み OUT トランザクション - DMA モード 2626
図 756. 通常の割り込み IN トランザクション - DMA モード 2627
図 757. 通常のアイソクロナス OUT/IN トランザクション - DMA モード 2628
図 758. 通常のアイソクロナス IN トランザクション - DMA モード 2629
図 759. 受信 FIFO パケット読み出し 2635
図 760. SETUP パケットの処理 2637
図 761. バルク OUT トランザクション 2643
図 762. TRDT 最大タイミングのケース 2652
図 763. A デバイスの SRP 2653
図 764. B デバイスの SRP 2654
図 765. A デバイスの HNP 2655
図 766. B デバイスの HNP 2656
図 767. Ethernet ハイレベルブロック図 2663
図 768. DMA 送信フロー（標準モード） 2666
図 769. DMA 送信フロー（OSP モード） 2668
図 770. 受信 DMA フロー 2670
図 771. MAC 送信フローの概要 2672
図 772. MAC 受信フロー 2674
図 773. パケットフィルタシーケンス 2678
図 774. ネットワーク時間の同期 2685
図 775. ピアツーピアパスパス補正をサポートする伝搬遅延計算 2686
図 776. 密補正法を使用したシステム時間の更新 2690
図 777. TCP セグメンテーションオフロードの概要 2696
図 778. 分割されたパケットのヘッダおよびペイロードフィールド 2699
図 779. サポートされる PHY インタフェース 2708
図 780. SMA インタフェースブロック 2709
図 781. MDIO パケットの構造 2709
図 782. 書き込みデータパケット 2710
図 783. 読み出しデータパケット 2711
図 784. メディア独立インタフェース（MII）信号 2711
図 785. RMII ブロック図 2713
図 786. 送信ビットの順序 2714
図 787. 受信ビットの順序 2715
図 788. ディスクリプタのリング構造 2732
図 789. DMA ディスクリプタリング 2733
図 790. 送信ディスクリプタ（読み出しフォーマット） 2734
図 791. 送信ディスクリプタの書き戻しフォーマット 2738
図 792. 送信コンテキストディスクリプタのフォーマット 2742
図 793. 受信通常ディスクリプタ（読み出しフォーマット） 2745
図 794. 受信通常ディスクリプタ（書き戻しフォーマット） 2748
図 795. 受信コンテキストディスクリプタ 2755
図 796. ETH_DMAISR フラグの生成 2774
図 797. HDMI-CEC ブロック図 2892
図 798. メッセージの構造 2893
図 799. ブロック 2893
図 800. ビットタイミング 2894

参考資料

図 801. 信号フリータイム 2894
図 802. アービトレーションフェーズ 2895
図 803. 3 つの公称ビット周期の SFT 2895
図 804. エラービットタイミング 2896
図 805. エラー処理 2898
図 806. TXERR 検出 2899
図 807. デバッグインフラストラクチャのブロック図 2911
図 808. デバッグインフラストラクチャの電力ドメイン 2913
図 809. デバッグインフラストラクチャのクロックドメイン 2914
図 810. SWD 正常データ転送 2917
図 811. JTAG TAP ステートマシン 2918
図 812. デバッグポートとアクセスポートの接続 2930
図 813. APB-D CoreSight コンポーネントのトポロジ 2938
図 814. グローバルタイムスタンプの配布 2945
図 815. 組み込みクロストリガ 2954
図 816. トリガ入力から出力へのマッピング 2956
図 817. ETF ステート遷移図 2985
図 818. Cortex-M7 の CoreSight トポロジ 3054

参考資料

1 このマニュアルにおける表記の規則

1.1 レジスタに関する略記

レジスタの説明では、次の略記が使用されます。

読み出し／書き込み（rw）	このビットは、ソフトウェアによる読み出しと書き込みができます。
読み出し専用（r）	このビットは、ソフトウェアによる読み出しのみが可能です。
書き込み専用（w）	このビットは、ソフトウェアによる書き込みのみが可能です。このビットを読み出すと、リセット値が返されます。
読み出し／クリア（rc_w1）	このビットは、ソフトウェアによって読み出すことができ、“1”を書き込むことによってクリアできます。“0”を書き込んでも、ビットの値は変化しません。
読み出し／クリア（rc_w0）	このビットは、ソフトウェアによって読み出すことができ、“0”を書き込むことによってクリアできます。“1”を書き込んでも、ビットの値は変化しません。
読み出し／読み出しによるクリア（rc_r）	このビットは、ソフトウェアによって読み出すことができます。このビットを読み出すと、自動的に“0”にクリアされます。このビットを書き込んでも、ビットの値は変化しません。
読み出し／セット（rs）	このビットは、ソフトウェアによって読み出しとセットができます。“0”を書き込んでも、ビットの値は変化しません。
予約済み（Res.）	予約済みビットであり、リセット値に保持する必要があります。

1.2 用語

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

- **ワード**：32 ビット長のデータ。
- **ハーフワード**：16 ビット長のデータ。
- **バイト**：8 ビット長のデータ。
- **ダブルワード**：64 ビット長のデータ。
- フラッシュワード：256 ビット長のデータ。
- **IAP（アプリケーション内プログラミング）**：IAP は、ユーザプログラム実行中にマイクロコントローラのフラッシュメモリを再プログラムする機能です。
- **ICP（インサーキットプログラミング）**：ICP は、ユーザアプリケーションボードにデバイスが搭載された状態で、JTAG プロトコルや SWD プロトコルまたはブートローダを用いて、マイクロコントローラのフラッシュメモリをプログラムする機能です。
- **オプションバイト**：フラッシュメモリに格納された製品設定ビット。
- **AHB**：アドバンストハイパフォーマンスバス。
- AXI：Advanced eXtensible Interface プロトコル
- **PCROP**：独自仕様コード読み出し保護
- RDP：読み出し保護

参考資料

1.3 使用可能なペリフェラル

すべての販売タイプで使用可能なペリフェラルとその数については、該当するデバイスのデータシートを参照してください。

2 メモリとバスのアーキテクチャ

2.1 システムアーキテクチャ

AXI バスマトリックス（1 つ）および 2つのAHB バスマトリックスとそれらをつなぐバスブリッジにより、バスマスタをバススレーブに相互接続することができます。*表 1* および*図 1* を参照してください。

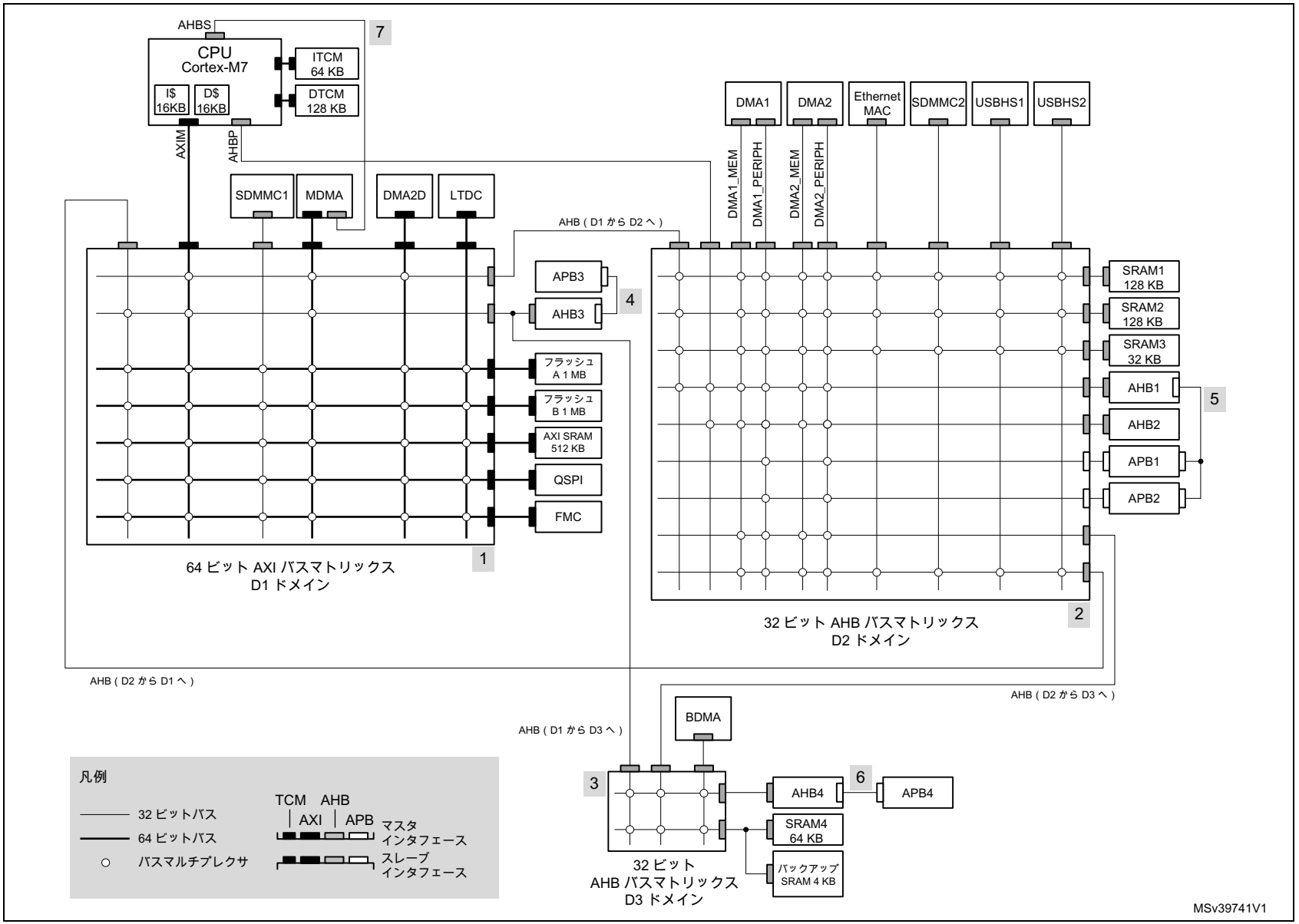
表 1. バスマスタとバススレーブの相互接続

	バスマスタ/タイプ(1)																			
	Cortex-M7 - AXIM	Cortex-M7 - AHBP	**Cortex-M7 - ITCM**	**Cortex-M7 -DTCM**	SDMMC1	**MDMA - AXI**	MDMA - AHBS	**DMA2D**	**LTDC**	DMA1 - MEM	DMA1 - PERIPH	DMA2 - MEM	DMA2 - PERIPH	Eth.MAC - AHB	SDMMC2 - AHB	USBHS1 - AHB	USBHS2 - AHB	BDMA - AHB
バススレーブ/タイプ (1)	**相互接続パスおよびタイプ(2)**																			
ITCM	-	-	**D**	-	-	-	**7**	-	-	-	-	-	-	-	-	-	-	-	-	-
DTCM	-	-	-	**D**	-	-	7	-	-	-	-	-	-	-	-	-	-	-	-	-
AHB3 ペリフェラル	1	-	-	-	-	1	-	-	-	21	21	-	21	21	-	21	21	21	21	-
APB3 ペリフェラル	14	-	-	-	-	14	-	-	-	214	214	-	214	214	-	214	214	214	214	-
フラッシュ A	**1**	-	-	-	*1*	**1**	-	**1**	**1**	*21*	*21*	-	*21*	*21*	-	*21*	*21*	*21*	*21*	-
フラッシュ B	**1**	-	-	-	*1*	**1**	-	**1**	**1**	*21*	*21*	-	*21*	*21*	-	*21*	*21*	*21*	*21*	-
AXI SRAM	**1**	-	-	-	*1*	**1**	-	**1**	**1**	*21*	*21*	-	*21*	*21*	-	*21*	*21*	*21*	*21*	-
QUADSPI	**1**	-	-	-	*1*	**1**	-	**1**	**1**	*21*	*21*	-	*21*	*21*	-	*21*	*21*	*21*	*21*	-
FMC	**1**	-	-	-	*1*	**1**	-	**1**	**1**	*21*	*21*	-	*21*	*21*	-	*21*	*21*	*21*	*21*	-
SRAM 1	12	-	-	-	-	12	-	12	-	2	2	-	2	2	-	2	2	2	2	-
SRAM 2	12	-	-	-	-	12	-	12	-	2	2	-	2	2	-	2	2	2	2	-
SRAM 3	12	-	-	-	-	12	-	12	-	2	2	-	2	2	-	2	2	2	2	-
AHB1 ペリフェラル	12	2	-	-	-	12	-	12	-	2	2	-	2	2	-	-	-	-	-	-
APB1 ペリフェラル	125	25	-	-	-	125	-	125	-	25	25	2	25	25	2	-	-	-	-	-
AHB2 ペリフェラル	-	2	-	-	-	-	-	-	-	2	2	-	2	2	-	-	-	-	-	-
APB2 ペリフェラル	125	25	-	-	-	125	-	125	-	25	25	2	25	25	2	-	-	-	-	-
AHB4 ペリフェラル	13	-	-	-	-	13	-	-	-	23	23	-	23	23	-	213	213	213	213	3
APB4 ペリフェラル	136	-	-	-	-	136	-	-	-	236	236	-	236	236	-	213	213	213	213	36
SRAM4	13	-	-	-	-	13	-	-	-	23	23	-	23	23	-	213	213	213	213	3
バックアップ RAM	13	-	-	-	-	13	-	-	-	23	23	-	23	23	-	213	213	213	213	3

1. **太字**は 64 ビットのバスを、通常字体は 32 ビットバスを表します。

2. 表本体のセルは、アクセスできる可能性、ユーティリティ、パス、およびタイプを示しています。
アクセスできる可能性およびユーティリティ：
任意の数値 = アクセス可能、「-」= アクセス不可、網掛け = アクセス有効／使用可能
アクセスパス：
D = 直接、1 = AXI バスマトリックス経由、2 = D2 の AHB バスマトリックス経由、3 = D3 の AHB バスマトリックス経由、4 = D1 の AHB/APB ブリッジ経由、5 = D2 の AHB/APB ブリッジ経由、6 = D3 の AHB/APB ブリッジ経由、7 = Cortex-M7 の AHBS バス経由、
複数桁の数字 = 相互接続パスは、桁順に複数のマトリックスやブリッジを通る
アクセスタイプ：
通常字体 = 32 ビット、*斜体* = バスマスタ終了時に 32 ビット／バススレーブ終了時に 64 ビット、**太字** = 64 ビット

図 1. STM32H7x3 デバイスのシステムアーキテクチャ



2.1.1 バスマトリックス

D1 ドメインの AXI バスマトリックス

D1 ドメインのマルチ AXI バスマトリックスは、複数のマスタから複数のスレーブへの同時アクセスを可能にし、また調停を行います。これにより、ハイスピードペリフェラルの同時動作を効率的に行うことができます。

調停には、QoS 機能を備えたラウンドロビン方式を使用します。

DTCM および ITCM （データおよび命令が密接に結合された RAM）は、専用の TCM バスで直接 Cortex-M7 コアに接続されています。MDMA コントローラは、CPU の特定の AHB スレーブである AHBS を通じて、DTCM および ITCM にアクセスできます。ITCM は Cortex-M7 によって 0 ウェイトステートの CPU クロックスピードでアクセスされます。

AXI 相互接続の詳細については、*セクション 5 : AXI インタコネクト*を参照してください。

D2 および D3 ドメインの AHB バスマトリックス

D2 および D3 ドメインの AHB バスマトリックスは、複数のマスタから複数のスレーブへの同時アクセスを可能にし、また調停を行います。これにより、ハイスピードペリフェラルの同時動作を効率的に行うことができます。

調停には、ラウンドロビン方式を使用します。

2.1.2 バス間のブリッジ

異なるタイプのバスを持つペリフェラルの通信を可能にするために、システムにはいくつかのバス間のブリッジがあります。

D1 および D3 ドメインの AHB/APB ブリッジでは、APB3 および APB4 のペリフェラルを、それぞれ AHB3 と AHB4 に接続することができます。D2 ドメインの AHB/APB ブリッジでは、APB1 および APB2 のペリフェラルを AHB1 に接続することができます。これらの AHB/APB ブリッジでは完全同期接続を提供しているため、APB ペリフェラルは接続している AHB に関係なくクロックとともに動作することができます。

AHB/APB ブリッジでは、APB1 および APB2 のペリフェラルを AHB1 を経由することなくそれぞれ DMA1 および DMA2 のペリフェラルバスに接続することもできます。

AHB/APB ブリッジは8ビット/16ビットAPBデータを、32ビットワードのうち8ビットは上位3バイト、16ビットは上位ハーフワードへ折り返す事によって、32ビットAHBデータに変換します。

AXI バスマトリックスでは AHB/AXI バスブリッジの機能をそのスレーブバスのインタフェースに組み込みます。*図 1* で 32 ビットと記されているマスタインタフェース上の AXI/AHB バスブリッジはマトリックスの外側にあります。

Cortex-M7 CPU では AHBS （AHB スレーブ）から AHB/TCM バス（ITCM および DTCM バス）への変換を提供しており、これによって MDMA コントローラは ITCM と DTCM にアクセスできるようになります。

2.1.3 ドメイン間バス

AHB（D2 から D1 へ）

この 32 ビットバスは、D2 ドメインを D1 ドメインの AXI バスマトリックスに接続します。これにより、D2 ドメインのバスマスタは、D1 ドメインならびに AHB（D1 から D3 へ）を介して間接的に D3 ドメインのリソース（バススレーブ）にアクセスできるようになります。

AHB（D1 から D2 へ）

この 32 ビットバスは、D1 ドメインを D2 ドメインの AHB バスマトリックスに接続します。これにより、D1 ドメインのバスマスタは、D2 ドメインのリソース（バススレーブ）にアクセスできるようになります。

AHB（D1 から D3 へ）

この 32 ビットバスは、D1 ドメインを D3 ドメインの AHB バスマトリックスに接続します。これにより、D1 ドメインのバスマスタは、D3 ドメインのリソース（バススレーブ）にアクセスできるようになります。

AHB（D2 から D3 へ）

この 32 ビットバスは、D2 ドメインを D3 ドメインの AHB バスマトリックスに接続します。これにより、D2 ドメインのバスマスタは、D3 ドメインのリソース（バススレーブ）にアクセスできるようになります。

2.1.4 CPU バス

Cortex®-M7 AXIM バス

Cortex®-M7 CPU は、ITCM、DTCM、AHB2 ペリフェラル、およびアドレス不適合により AHB1、APB1、および APB2 の各ペリフェラルを除くすべてのメモリおよびペリフェラルとのアクセスに 64 ビットの AXIM バスを使用します。

この AXIM バスは、CPU を D1 ドメインの AXI バスマトリックスに接続します。

Cortex®-M7 ITCM バス

Cortex®-M7 CPU は、64 ビット ITCM バスを使用して ITCM から命令をフェッチし、そのデータにアクセスします。

Cortex®-M7 DTCM バス

Cortex®-M7 CPU は、64 ビット DTCM バスを使用して DTCM のデータにアクセスします。命令をフェッチすることもできます。

Cortex®-M7 AHBS バス

Cortex®-M7 CPU は、32 ビット AHBS スレーブバスを使用して MDMA コントローラが ITCM と DTCM にアクセスできるようにします。

Cortex®-M7 AHBP バス

Cortex®-M7 CPU は、32 ビット AHBP バスを使用して、D2 ドメインの AHB バスマトリックスを経由して AHB1、AHB2、APB1、および APB2 の各ペリフェラルにアクセスします。

2.1.5 バスマスタペリフェラル

SDMMC1

SDMMC1 は、AXI バスマトリックスに接続された 32 ビットバスを使用して内部 AXI SRAM およびフラッシュメモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

SDMMC2

SDMMC2 は、D2 ドメインの AHB バスマトリックスに接続された 32 ビットバスを使用します。システムのバスマトリックスを通じて、内部 AXI SRAM、SRAM1、SRAM2、SRAM3、およびフラッシュメモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

MDMA コントローラ

MDMA コントローラは、メモリ間およびペリフェラルとの DMA データ転送に AXI バスマトリックスに接続された 64 ビットバスを使用します。システムのバスマトリックスおよび Cortex-M7 AHBS スレーブバスを通じて、すべての内部メモリと AHB1 ペリフェラルにアクセスできます。また、Quad SPI コントローラおよび FMC を通じて外部メモリにもアクセスできます。

DMA1 および DMA2 コントローラ

DMA1 および DMA2 コントローラには、D2 ドメインの AHB バスマトリックスに接続された、メモリバスとペリフェラルバスという 2 つの 32 ビットバスがあります。

メモリバスではメモリ間の DMA データ転送を行うことができます。システムのバスマトリックスを通じて、メモリバスは ITCM と DTCM を除くすべての内部メモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

ペリフェラルバスでは、2 つのペリフェラル間、2 つのメモリ間、またはペリフェラルとメモリの間で DMA データ転送を行うことができます。システムのバスマトリックスを通じて、ペリフェラルバスは ITCM と DTCM を除くすべての内部メモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。AHB および APB のすべてのペリフェラルにもアクセスできます。AHB1 を介さずに、APB1 および APB2 に直接アクセスすることもできます。

BDMA コントローラ

BDMA コントローラは、2 つのペリフェラル間、2 つのメモリ間、およびペリフェラルとメモリの間の DMA データ転送に、D3 ドメインの AHB バスマトリックスに接続された 32 ビットバスを使用します。D3 ドメインの AHB バスマトリックスを通じて、内部 SRAM4、バックアップ RAM、および AHB4 と APB4 の各ペリフェラルにアクセスできます。

Chrom-ART アクセラレータ™ （DMA2D）

DMA2D グラフィックアクセラレータは、AXI バスマトリックスに接続された 64 ビットバスを使用します。システムのバスマトリックスを通じて、内部 AXI SRAM、SRAM1、SRAM2、SRAM3、およびフラッシュメモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

LCD-TFT コントローラ（LTDC）

LTDC （LCD-TFT ディスプレイコントローラ）は、AXI バスマトリックスに接続された 64 ビットバスを使用して内部 AXI SRAM およびフラッシュメモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

Ethernet MAC

Ethernet MAC は、D2 ドメインの AHB バスマトリックスに接続された 32 ビットバスを使用します。システムのバスマトリックスを通じて、ITCM と DTCM を除くすべての内部メモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

USBHS1 および USBHS2 ペリフェラル

USBHS1 および USBHS2 ペリフェラル は、D2 ドメインの AHB バスマトリックスに接続された 32 ビットバスを使用します。システムのバスマトリックスを通じて、ITCM と DTCM を除くすべての内部メモリにアクセスでき、また Quad SPI コントローラおよび FMC を通じて外部メモリにアクセスできます。

2.1.6 クロックと機能ブロック

リセットの際、ペリフェラルなどのクロック機能ブロックと一部のメモリは無効になります（SRAM、DTCM、ITCM、およびフラッシュメモリを除く）。リセット時にクロックなしでブロックを動作する場合、ソフトウェアはまずそれぞれの RCC_AHBxENR または RCC_APBxENR レジスタを通じてクロックを有効にする必要があります。

2.2 メモリ構成

2.2.1 概要

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

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

アドレス指定可能なメモリ空間は、それぞれ 512 MB の 8 つのメインブロックに分割されています。

オンチップメモリとペリフェラルに割り当てられていないメモリマップ領域はすべて、「予約済み」とみなされます。使用可能なメモリ領域とレジスタ領域の詳細なマッピングについては、*メモリマップとレジスタ境界アドレス*およびペリフェラルのセクションを参照してください。

2.2.2 メモリマップとレジスタ境界アドレス

包括的なメモリマッピング図については、お使いのデバイスに対応するデータシートを参照してください。

次の表に、デバイスで使用可能なペリフェラルの境界アドレスを示します。

表 2. レジスタ境界アドレス

境界アドレス	ペリフェラル	バス	レジスタマップ
0x58026400 - 0x580267FF	HSEM	AHB4（D3）	*セクション 10.4 : HSEM レジスタ*
0x58026000 - 0x580263FF	ADC3		*セクション 25.6 : ADC 共通レジスタ*
0x58025800 - 0x58025BFF	DMAMUX2		*セクション 17.6 : DMAMUX レジスタ*
0x58025400 - 0x580257FF	BDMA		*セクション 16.4 : BDMA レジスタ*
0x58024C00 - 0x58024FFF	CRC		*セクション 21.4 : CRC レジスタ*
0x58024800 - 0x58024BFF	PWR		*セクション 6.8 : PWR レジスタの説明*
0x58024400 - 0x580247FF	RCC		*セクション 8.7 : RCC レジスタの説明*
0x58022800 - 0x58022BFF	GPIOK		*セクション 11.4 : GPIO レジスタ*
0x58022400 - 0x580227FF	GPIOJ		*セクション 11.4 : GPIO レジスタ*
0x58022000 - 0x580223FF	GPIOI		*セクション 11.4 : GPIO レジスタ*
0x58021C00 - 0x58021FFF	GPIOH		*セクション 11.4 : GPIO レジスタ*
0x58021800 - 0x58021BFF	GPIOG		*セクション 11.4 : GPIO レジスタ*
0x58021400 - 0x580217FF	GPIOF		*セクション 11.4 : GPIO レジスタ*
0x58021000 - 0x580213FF	GPIOE		*セクション 11.4 : GPIO レジスタ*
0x58020C00 - 0x58020FFF	GPIOD		*セクション 11.4 : GPIO レジスタ*
0x58020800 - 0x58020BFF	GPIOC		*セクション 11.4 : GPIO レジスタ*
0x58020400 - 0x580207FF	GPIOB		*セクション 11.4 : GPIO レジスタ*
0x58020000 - 0x580203FF	GPIOA		*セクション 11.4 : GPIO レジスタ*

表 2. レジスタ境界アドレス（続き）

境界アドレス	ペリフェラル	バス	レジスタマップ
0x58006400 - 0x58006BFF	予約済み	APB4（D3）	予約済み
0x58005400 - 0x580057FF	SAI4		*セクション 51.5 : SAI レジスタ*
0x58004C00 - 0x58004FFF	予約済み		予約済み
0x58004800 - 0x58004BFF	IWDG1		*セクション 45.4 : IWDG レジスタ*
0x58004000 - 0x580043FF	RTC & BKP レジスタ		*セクション 46.6 : RTC レジスタ*
0x58003C00 - 0x58003FFF	VREF		*セクション 27.3 : VREFBUF レジスタ*
0x58003800 - 0x58003BFF	COMP1 - COMP2		*セクション 28.7 : COMP レジスタ*
0x58003000 - 0x580033FF	LPTIM5		*セクション 43.6 : LPTIM レジスタ*
0x58002C00 - 0x58002FFF	LPTIM4		*セクション 43.6 : LPTIM レジスタ*
0x58002800 - 0x58002BFF	LPTIM3		*セクション 43.6 : LPTIM レジスタ*
0x58002400 - 0x580027FF	LPTIM2		*セクション 43.6 : LPTIM レジスタ*
0x58001C00 - 0x58001FFF	I2C4		*セクション 47.7 : I2C レジスタ*
0x58001400 - 0x580017FF	SPI6		*セクション 50.11 : SPI/I2S レジスタ*
0x58000C00 - 0x58000FFF	LPUART1		*セクション 49.5 : LPUART レジスタ*
0x58000400 - 0x580007FF	SYSCFG		*セクション 12.3 : SYSCFG レジスタの説明*
0x58000000 - 0x580003FF	EXTI		*セクション 20.6 : EXTI レジスタの説明*
0x52008000 - 0x52008FFF	遅延ブロック SDMMC1	AHB3（D1）	*セクション 24.4 : DLYB レジスタ*
0x52007000 - 0x52007FFF	SDMMC1		*セクション 55.8 : SDMMC レジスタ*
0x52006000 - 0x52006FFF	遅延ブロック QUADSPI		*セクション 24.4 : DLYB レジスタ*
0x52005000 - 0x52005FFF	QUADSPI 制御レジスタ		*セクション 23.5 : QUADSPI レジスタ*
0x52004000 - 0x52004FFF	FMC 制御レジスタ		*セクション 22.7.6 : NOR/PSRAM コントローラレジスタ*、*セクション 22.8.7 : NAND フラッシュコントローラレジスタ*、*セクション 22.9.5 : SDRAM コントローラレジスタ*
0x52003000 - 0x52003FFF	JPEG		*セクション 33.5 : JPEG コーデックレジスタ*
0x52002000 - 0x52002FFF	フラッシュ インタフェースレジスタ		*セクション 3.5 : フラッシュレジスタ*
0x52001000 - 0x52001FFF	Chrom-ART（DMA2D）		*セクション 18.6 : DMA2D レジスタ*
0x52000000 - 0x52000FFF	MDMA		*セクション 14.5 : MDMA レジスタ*
0x51000000 - 0x510FFFFF	GPV		*セクション 5.4 : AXI インタコネクトレジスタ*
0x50003000 - 0x50003FFF	WWDG1	APB3（D1）	*セクション 44.4 : .WWDG レジスタ*
0x50001000 - 0x50001FFF	LTDC		*セクション 32.7 : LTDC レジスタ*
0x50000000 - 0x50000FFF	予約済み		
0x48022800 - 0x48022BFF	遅延ブロック SDMMC2	AHB2（D2）	*セクション 24.4 : DLYB レジスタ*

表 2. レジスタ境界アドレス（続き）

境界アドレス	ペリフェラル	バス	レジスタマップ
0x48022400 - 0x480227FF	SDMMC2	AHB2（D2）	*セクション 55.8 : SDMMC レジスタ*
0x48021800 - 0x48021BFF	RNG		*セクション 34.8 : RNG レジスタ*
0x48021400 - 0x480217FF	HASH		*セクション 36.6 : HASH レジスタ*
0x48021000 - 0x480213FF	CRYPTO		*セクション 35.6 : CRYP レジスタ*
0x48020000 - 0x480203FF	DCMI		*セクション 31.7 : DCMI レジスタの説明*
0x40080000 - 0x400BFFFF	USB2 OTG フルスピード	AHB1（D2）	*セクション 57.14 : OTG_HS レジスタ*
0x40040000 - 0x4007FFFF	USB1 OTG ハイスピード／フルスピード		*セクション 57.14 : OTG_HS レジスタ*
0x40028000 - 0x400293FF	ETHERNET MAC		*セクション 58.11 : Ethernet レジスタの説明*
0x40024400 - 0x400247FF	予約済み		予約済み
0x40022000 - 0x400223FF	ADC1 - ADC2		*セクション 25.6 : ADC 共通レジスタ*
0x40020800 - 0x40020BFF	DMAMUX1		*セクション 17.6 : DMAMUX レジスタ*
0x40020400 - 0x400207FF	DMA2		*セクション 15.5 : DMA レジスタ*
0x40020000 - 0x400203FF	DMA1		*セクション 15.5 : DMA レジスタ*
0x40017400 - 0x400177FF	HRTIM	APB2（D2）	*セクション 37.5 : HRTIM レジスタ*
0x40017000 - 0x400173FF	DFSDM1		*セクション 30.7 : DFSDM チャネル y レジスタ（y=0～7）*、*セクション 30.8 : DFSDM フィルタ x モジュールレジスタ（x=0～3）*を更新。
0x40016000 - 0x400163FF	SAI3		*セクション 51.5 : SAI レジスタ*
0x40015C00 - 0x40015FFF	SAI2		*セクション 51.5 : SAI レジスタ*
0x40015800 - 0x40015BFF	SAI1		*セクション 51.5 : SAI レジスタ*
0x40015000 - 0x400153FF	SPI5		*セクション 50.11 : SPI/I2S レジスタ*
0x40014800 - 0x40014BFF	TIM17		*セクション 41.6 : TIM16/TIM17 レジスタ*
0x40014400 - 0x400147FF	TIM16		*セクション 41.6 : TIM16/TIM17 レジスタ*
0x40014000 - 0x400143FF	TIM15		*セクション 41.5 : TIM15 レジスタ*
0x40013400 - 0x400137FF	SPI4		*セクション 50.11 : SPI/I2S レジスタ*
0x40013000 - 0x400133FF	SPI1/I2S1		*セクション 50.11 : SPI/I2S レジスタ*
0x40011400 - 0x400117FF	USART6		*セクション 48.7 : USART レジスタ*
0x40011000 - 0x400113FF	USART1		*セクション 48.7 : USART レジスタ*
0x40010400 - 0x400107FF	TIM8		*セクション 38.4 : TIM1/TIM8 レジスタ*
0x40010000 - 0x400103FF	TIM1		*セクション 38.4 : TIM1/TIM8 レジスタ*

表 2. レジスタ境界アドレス（続き）

境界アドレス	ペリフェラル	バス	レジスタマップ
0x4000AC00 - 0x4000D3FF	CAN メッセージ RAM	APB1（D2）	セクション 56.4 : FDCAN レジスタ
0x4000A800 - 0x4000ABFF	CAN CCU		セクション 56.4 : FDCAN レジスタ
0x4000A400 - 0x4000A7FF	FDCAN2		セクション 56.4 : FDCAN レジスタ
0x4000A000 - 0x4000A3FF	FDCAN1		セクション 56.4 : FDCAN レジスタ
0x40009400 - 0x400097FF	MDIOS		セクション 54.4 : MDIOS レジスタ
0x40009000 - 0x400093FF	OPAMP		セクション 29.6 : OPAMP レジスタ
0x40008800 - 0x40008BFF	SWPMI		セクション 53.6 : SWPMI レジスタ
0x40008400 - 0x400087FF	CRS		セクション 9.7 : CRS レジスタ
0x40007C00 - 0x40007FFF	UART8		セクション 48.7 : USART レジスタ
0x40007800 - 0x40007BFF	UART7		セクション 48.7 : USART レジスタ
0x40007400 - 0x400077FF	DAC1		セクション 26.5 : DAC レジスタ
0x40006C00 - 0x40006FFF	HDMI-CEC		セクション 59.7 : HDMI-CEC レジスタ
0x40005C00 - 0x40005FFF	I2C3		セクション 47.7 : I2C レジスタ
0x40005800 - 0x40005BFF	I2C2		セクション 47.7 : I2C レジスタ
0x40005400 - 0x400057FF	I2C1		セクション 47.7 : I2C レジスタ
0x40005000 - 0x400053FF	UART5		セクション 48.7 : USART レジスタ
0x40004C00 - 0x40004FFF	UART4		セクション 48.7 : USART レジスタ
0x40004800 - 0x40004BFF	USART3		セクション 48.7 : USART レジスタ
0x40004400 - 0x400047FF	USART2		セクション 48.7 : USART レジスタ
0x40004000 - 0x400043FF	SPDIFRX		セクション 52.5 : SPDIFRX インタフェースレジスタ
0x40003C00 - 0x40003FFF	SPI3/I2S3		セクション 50.11 : SPI/I2S レジスタ
0x40003800 - 0x40003BFF	SPI2/I2S2		セクション 50.11 : SPI/I2S レジスタ
0x40002C00 - 0x40002FFF	予約済み		予約済み
0x40002400 - 0x400027FF	LPTIM1		セクション 43.6 : LPTIM レジスタ
0x40002000 - 0x400023FF	TIM14		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001C00 - 0x40001FFF	TIM13		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001800 - 0x40001BFF	TIM12		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001400 - 0x400017FF	TIM7		セクション 42.4 : TIM6/TIM7 レジスタ
0x40001000 - 0x400013FF	TIM6		セクション 42.4 : TIM6/TIM7 レジスタ
0x40000C00 - 0x40000FFF	TIM5		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000800 - 0x40000BFF	TIM4		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000400 - 0x400007FF	TIM3		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000000 - 0x400003FF	TIM2		セクション 39.4 : TIM2/TIM3/TIM4/TIM5 レジスタ

2.3 内蔵 SRAM

STM32H7x3 デバイスの機能：

- 最大 864 KB のシステム SRAM
- 128 KB のデータ TCM RAM
- 64 KB の命令 TCM RAM
- 4 KB のバックアップ SRAM

内蔵システム SRAM は最大 5 つのブロックに分割されます。

- AXI SRAM （D1 ドメイン）:
  - アドレス 0x2400 0000 に配置された、BDMA を除くすべてのシステムマスタが D1 ドメインの AXI バスマトリックスを通じてアクセス可能な AXI SRAM
- AHB SRAM （D2 ドメイン）:
  - アドレス 0x3000 0000 に配置された、BDMA を除くすべてのシステムマスタが D2 ドメインの AHB マトリックスを通じてアクセス可能な AHB SRAM1
  - アドレス 0x3002 0000 に配置された、BDMA を除くすべてのシステムマスタが D2 ドメインの AHB マトリックスを通じてアクセス可能な AHB SRAM2
  - アドレス 0x3004 0000 に配置された、BDMA を除くすべてのシステムマスタが D2 ドメインの AHB マトリックスを通じてアクセス可能な AHB SRAM3
- AHB SRAM （D3 ドメイン）:
  - アドレス 0x3800 0000 に配置された、ほとんどのシステムマスタが D3 ドメインの AHB マトリックスを通じてアクセス可能な AHB SRAM4

AHB SRAM システムには、バイト、ハーフワード（16 ビット単位）、またはフルワード（32 ビット単位）によるアクセスが可能で、AXI SRAM システムには、バイト、ハーフワード、フルワード、またはダブルワード（64 ビット単位）によるアクセスが可能です。これらのメモリは、最大システムクロック周波数（ウェイトステートなし）でアドレス指定できます。

Ethernet MAC またはUSB OTG ハイスピードペリフェラルがSRAMセクションにアクセスしていても、AHBマスタは同時に 別のSRAM セクションに読み出し／書き込みアクセスできます。たとえばCPU が SRAM1 にアクセスしている間に、Ethernet MACは同時に SRAM2 にアクセスします。

TCM SRAM は Cortex®-M7 専用です。

- TCM インタフェース上の DTCM-RAM はアドレス 0x2000 0000 に配置され、Cortex®-M7 および Cortex®-M7 CPU の AHBS スレーブバスを通じて MDMA によるアクセスが可能です。
- TCM インタフェース上の ITCM-RAM はアドレス 0x0000 0000 に配置され、Cortex®-M7 および Cortex®-M7 CPU の AHBS スレーブバスを通じて MDMA によるアクセスが可能です。

バックアップ RAM はアドレス 0x3880 0000 に配置され、D3 ドメインの AHB マトリックスを通じてほとんどのシステムマスタによるアクセスが可能です。

2.4 フラッシュメモリの概要

フラッシュメモリインタフェースは、フラッシュメモリへの CPU AXI アクセスを管理します。消去を実行し、フラッシュメモリ操作そして読み出し／書き込み保護メカニズムをプログラムします。

フラッシュメモリは、次のように構成されています。

- セクタに分割された 2 つのメインメモリブロック
- 情報ブロック：
  - システムメモリブートモードでデバイスがブートするシステムメモリ
  - デバイスが STANDBY または STOP モードのときに読み出し／書き込み保護、BOR レベル、ウォッチドッグソフトウェア／ハードウェアおよびリセットを設定するオプションバイト

詳細については、*セクション 3：内蔵フラッシュメモリ（フラッシュ）*を参照してください。

2.5 ブート設定

STM32H7x3 では、*表 3*に示すように、2 つのブート領域が、BOOT ピンと BOOT_ADD0 および BOOT_ADD1 オプションバイトでプログラムされたブートのベースアドレスによって選択できます。

表 3. ブートモード

ブートモード選択		ブート領域
BOOT ピン	ブートアドレスのオプションバイト	
0	BOOT_ADD0[15:0]	ユーザオプションバイト BOOT_ADD0[15:0]で定義されたブートアドレスST のプログラム値： 0x0800 0000 番地のフラッシュメモリ
1	BOOT_ADD1[15:0]	ユーザオプションバイト BOOT_ADD1[15:0]で定義されたブートアドレスST のプログラム値： 0x1FF0 0000 番地のシステムブートローダ

BOOT ピンの値は、リセット解除後、SYSCLK の 4 番目の立ち上がりエッジでラッチされます。リセット後にBOOT ピンをセットするかどうかはユーザ次第です。

BOOT ピンは、デバイスが STANDBY モードを終了するときにも再度サンプリングされます。したがって、これらのピンはデバイスが STANDBY モードのときでも必要なブートモード設定に保たれる必要があります。

起動遅延時間後、プロセッサのリセットが解除される前に、ブート領域の選択が行われます。

BOOT_ADD0 および BOOT_ADD1 のアドレスオプションバイトを使用すると、下記を含む 0x0000 0000 から 0x3FFF 0000 までのすべてのブートメモリアドレスをプログラムすることができます。

- すべてのフラッシュアドレス空間
- すべての RAM アドレス空間：ITCM、DTCM RAM、および SRAM
- TCM 上の RAM

BOOT_ADD0/BOOT_ADD1 のオプションバイトをリセット後に変更して、次のリセット後に他のブートアドレスからブートできます。

プログラムされたブートメモリアドレスがメモリ配置範囲外、または予約済み領域である場合、デフォルトのブートフェッチアドレスが次のようにプログラムされます。

– ブートアドレス 0 : 0x0800 0000 のフラッシュ
– ブートアドレス 1 : アドレス 0x0000 0000 の ITCM 上の RAM

フラッシュのレベル 2 保護が有効の場合、フラッシュメモリからのブートのみが使用できます。すでに BOOT_ADD0/BOOT_ADD1 のオプションバイトにプログラムされたブートアドレスがメモリ範囲外または RAM アドレス範囲の一部である場合、デフォルトのフェッチはアドレス 0x0800 0000 のフラッシュメモリから強制的に行われます。

内蔵ブートローダ

内蔵ブートローダコードは、システムメモリに配置されます。これは生産時に ST によってプログラムされています。次のシリアルインタフェースのいずれかを使用して、フラッシュメモリを再プログラムするために使用します。

- ピン PA9/PA10 および PB14/PB15 の USART1、PA3/PA2 ピンの USART2、および PB10/PB11 ピンの USART3
- PB6/PB9 ピンの I2C1、PF0/PF1 ピンの I2C2、および　PA8/PC9 ピンの I2C3
- デバイスモード（DFU）での USB OTG フルスピード（PA11/PA12 ピン）
- PA7/PA6/PA5/PA4 ピンの SPI1、PI3/PI2/PI1/PI0 ピンの SPI2、PC12/PC11/PC10/PA15 ピンの SPI3、および PE14/PE13/PE12/PE11 ピンの SPI4

詳細については、アプリケーションノート AN2606 を参照してください。

参考資料

3 内蔵フラッシュメモリ（フラッシュ）

3.1 概要

フラッシュメモリインタフェースは、すべてのマスタのフラッシュメモリへの AXI アクセスを管理します。フラッシュメモリの読み出し、プログラム、および消去操作と、読み出しおよびプログラム／消去の保護メカニズムを実装しています。

このフラッシュインタフェースは、リセット時のオプションバイトローディングと動的なオプションバイトの変更も管理します。

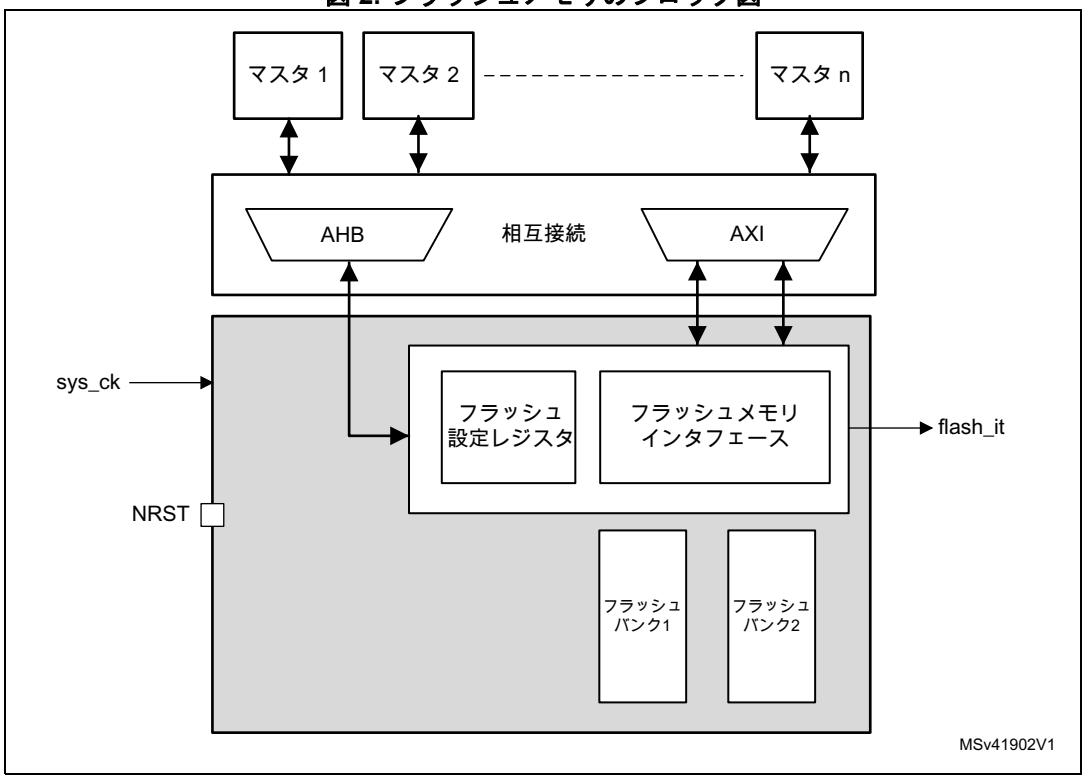
3.2 FLASH の主な機能

- 最大メモリ容量 2MB
- エラーコード訂正（ECC）: 256 ビットフラッシュワードで 10 ECC ビット
- ダブルワード、ワード、ハーフワード、およびバイト読み出し操作
- 256 ビットでのフラッシュプログラミング
- ダブルワード、ワード、ハーフワード、およびバイト書き込み操作
- セクタ消去、バンク消去、および全体消去
- 同時操作をサポートするデュアルバンク構成:2 つのバンクで 2 つの読み出し／プログラム／消去操作を並行して実行可能
- バンクスワッピング:各バンクのユーザフラッシュメモリのアドレスマッピングをスワップ可能
- 強化されたセキュリティ機能
  - 読み出し保護（RDP）
  - セクタ書き込み保護
  - 2 つの PCROP 保護領域（バンクに 1 つ）（実行のみ可能なメモリ）
- ユーザフラッシュメモリの 2 つのセキュア領域（バンクに 1 つ）

3.3 FLASH の機能説明

3.3.1 ブロック図

図 2. フラッシュメモリのブロック図



3.3.2 ピンおよび内部信号

表 4 にパッケージのピンまたはボールに接続されたフラッシュ入出力信号のリスト、*表 5* に内部フラッシュ信号のリストを示します。

表 4. パッケージのピンまたはボールに接続されたフラッシュ入力／出力信号

信号名	信号タイプ	説明
NRST	デジタル入力	外部リセット信号

表 5. フラッシュ内部入力／出力信号

信号名	信号タイプ	説明
sys_ck	デジタル入力	システムクロック
flash_it	デジタル出力	内蔵フラッシュメモリ割り込み

3.3.3 フラッシュメモリのアーキテクチャ

フラッシュメモリは、266 ビットフラッシュワードのメモリとして構成されており、コードとデータの両方の定数を格納するために使用できます。各ワードの構成は次のとおりです。

- 1 フラッシュワード（8 ワード、32 バイト、または 256 ビット）
- 10 ECC ビット

フラッシュメモリは、2 つの個別のバンクに分割されます。各バンクは、次のように構成されています。

- 128 KB のユーザセクタを 8 個含む 1 MB のユーザフラッシュメモリブロック（4 K フラッシュワード）
- デバイスがブート可能な 128 KB のシステムフラッシュメモリ

  この領域にはルートセキュアサービス（RSS）およびブートローダが含まれます。これらは、USART、USB (DFU)、I2C、SPI、または Ethernet のいずれかのインタフェースによるセキュアまたは非セキュアなフラッシュメモリのプログラミングに使用されます。システムフラッシュメモリは STMicroelectronicsの使用のために予約済みです。これはデバイスの製造時に STMicroelectronics によってプログラムされており、誤ったプログラム／消去操作から保護されています。詳細については、*www.st.com* から入手可能なアプリケーションノート AN2606 「STM32 マイクロコントローラのシステムフラッシュメモリブートモード」を参照してください。
- ユーザ設定用の 2 KB （64 フラッシュワード）のユーザオプションバイト

  この領域は、バンク 1 でのみ使用可能です。ユーザフラッシュメモリやシステムフラッシュメモリとは異なり、特定のメモリアドレスにマッピングされているわけではなく、またフラッシュレジスタインタフェースを通じてのみアクセスすることができます。

*図 2*にフラッシュメモリの全体のブロック図を示し、*表 6* ではフラッシュメモリ構成について説明します。

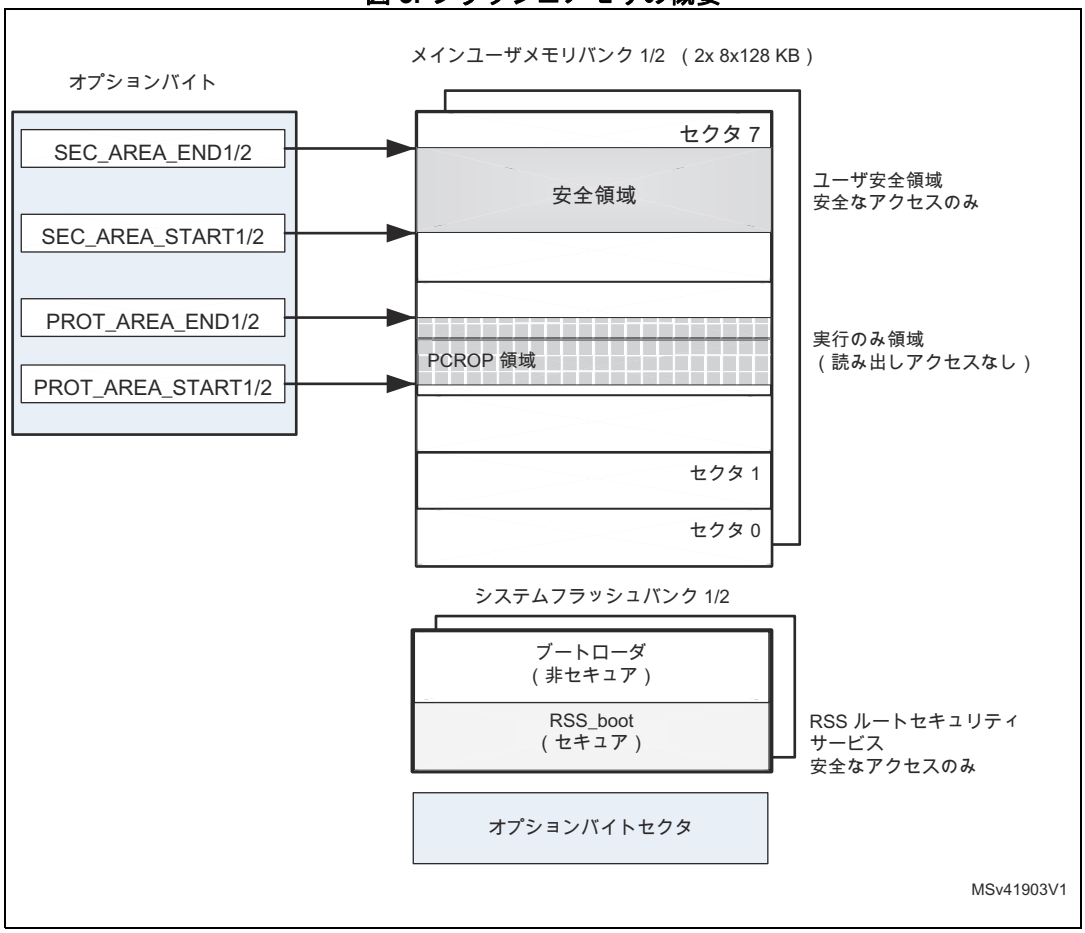
フラッシュメモリのマッピングと保護メカニズムの概要について、*図 3* に示します。PCROP とセキュア領域は、バンクごとに定義できます。*セクション 3.3.12 : 保護メカニズム* でこれらの保護された領域の特性について詳しく説明し、*セクション 4 : セキュリティメモリ管理* でシステムフラッシュメモリに保存されているセキュアサービスについて説明します。

表 6. フラッシュモジュール - 1 MB のデュアルバンク構成

フラッシュ領域		開始アドレス	終了アドレス	サイズ（バイト）	領域名
バンク 1	ユーザフラッシュメモリ	0x0800 0000	0x0801 FFFF	128 K	セクタ 0
		0x0802 0000	0x0803 FFFF	128 K	セクタ 1
		...	...	...	...
		0x080E 0000	0x080F FFFF	128 K	セクタ 7
	システムフラッシュメモリ	0x1FF0 0000	0x1FF1 FFFF	128 K	システムフラッシュメモリ
バンク 2	ユーザフラッシュメモリ	0x0810 0000	0x0811 FFFF	128 K	セクタ 0
		0x0812 0000	0x0813 FFFF	128 K	セクタ 1
		...	...	...	...
		0x081E 0000	0x081F FFFF	128 K	セクタ 7
	システムフラッシュメモリ	0x1FF4 0000	0x1FF5 FFFF	128 K	システムフラッシュメモリ

参考資料

図 3. フラッシュメモリの概要



3.3.4 フラッシュ読み出し操作

読み出しプロトコル

フラッシュインタフェースは、以下のようなアクセスタイプをサポートしています。

- ダブルワード（64 ビット）
- シングルワード（32 ビット）
- ハーフワード（16 ビット）
- バイト（8 ビット）

フラッシュメモリからデータを読み出す際は、フラッシュインタフェースクロックを有効にし、動作させる必要があります。フラッシュインタフェースの読み出し操作が正しく実行されるよう、フラッシュメモリインタフェースの周波数に従って、FLASH_ACR レジスタでウェイトステート（LATENCY）の数を正しく設定する必要があります（*セクション：読み出しアクセスの遅延* を参照）。

ウェイトステート数の設定を誤ると、誤った値が読み出されるか（十分なウェイトステートがプログラミングされていない場合など）、アクセスに時間がかかります（プログラミングされたウェイトステートが多すぎる場合など）。

操作はフラッシュインタフェースで受信した順番に実行されます。

フラッシュインタフェースは、特定のバンクで読み出し、プログラミング、または消去操作を一度に 1 回だけ実行する方法で構築されます。たとえば、ユーザフラッシュメモリがスワップされた場合について考えてみましょう。最初の AXI インタフェースではバンク 1 のシステムフラッシュメモリ（ICP）での読み出し操作を要求し、2 番目の AXI インタフェースでは同じバンクのユーザフラッシュメモリでの読み出しまたはフェッチ操作を要求します。この場合、ハードウェアアービトレーションが発生して両方の読み出しリクエストが受け付けられますが、処理は順番に行われます。

対象のバンクが異なる場合、同時に読み出し／プログラム／消去操作を行うことができます。プログラム／消去操作中に同じバンクで読み出し操作を要求すると、読み出しコマンドはバッファされ、プログラム／消去操作が完了してから実行されます。システムは、読み出しコマンドの FIFO がフル（3 つの読み出し操作がすでにペンディング）になるまでストールされません。

読み出しシーケンス

フラッシュインタフェースには、コード／データアクセス用のデュアル AXI バスインタフェースと、フラッシュインタフェース設定用の AHB インタフェースが実装されています。AXI インタフェースでは、読み出し／プログラム操作を同時に要求できます。

読み出しのメカニズムは、次のとおりです。

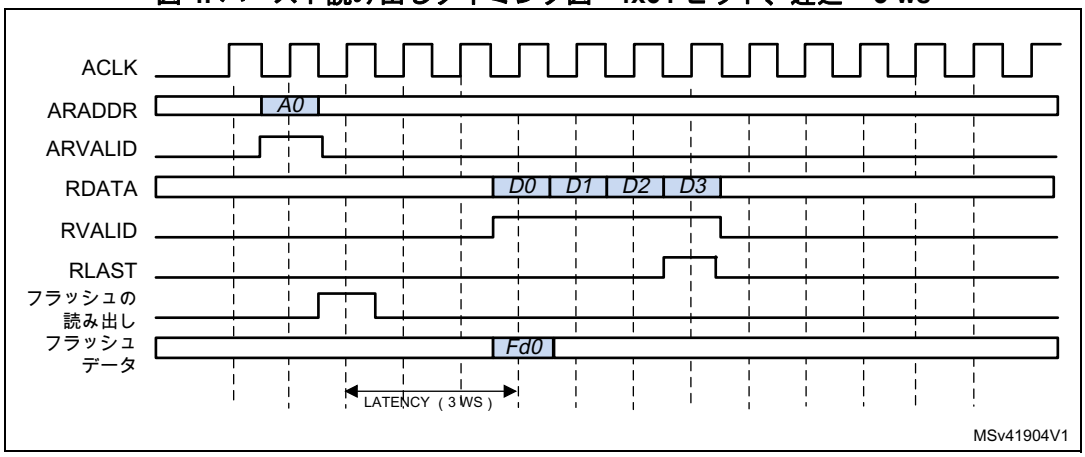
- 読み出しコマンドのバッファ段数は 3 段です。フル（3 個の読み出しリクエストがバッファのキューに入れられている）状態で新しい読み出しリクエストが発生すると、バスインタフェースはストールし、結果としてマスタもストールします。
- 現在の読み出しトランザクションの最後のデータがフラッシュメモリからフラッシュメモリインタフェースの読み出しデータバッファに転送されるとすぐに、フラッシュインタフェースでは読み出しコマンドキューバッファのリクエストを解放します。
- 読み出しバッファにデータがない任意のシステム読み出しリクエストでは、フラッシュ読み出し操作がトリガされます。これは、読み出しデータバッファ内でバッファリングされます。
- 連続するいくつかの読み出しアクセスが同じフラッシュデータワード（256 ビット）に属するデータを要求する場合、データは現在のデータ読み出しバッファから直接読み出され、さらなるフラッシュ読み出し操作をトリガすることはありません。

参考資料

図 4 および *図 5* に、それぞれ4 および 8 個のダブルワードデータのバースト読み出しトランザクションのタイミング図を示します。遅延は 3 ウェイトステートに設定されます。

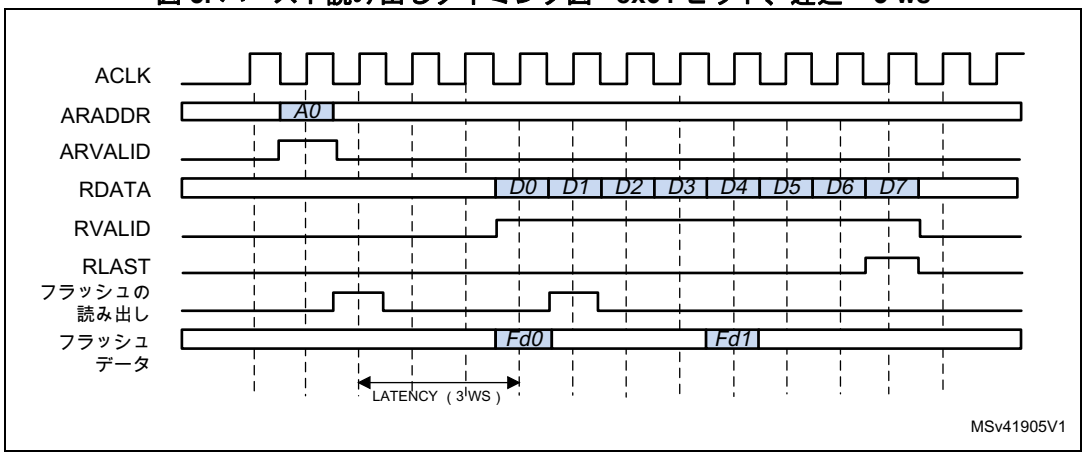
1. 最初のデータは最初の遅延（3 ウェイトステート）後に使用できます。
2. 現在の読み出しバッファでデータを使用できるようになると、フラッシュインタフェースは読み出しコマンドキューから次のコマンドを実行できるようになります。
3. 次のフラッシュワード読み出しアクセスに対応する 2 番目の遅延は、連続して 4 個のダブルワードの読み出しを処理できる現在のバッファの読み出し操作でマスクされます。これは、0 ウェイトステートの実行シーケンスと同等です。

図 4. バースト読み出しタイミング図 - 4x64 ビット、遅延 = 3 ws



1. ACLK、ARADDR、ARVALID、RDATA、RVALID、および RLAST は AXI バス信号です。
フラッシュ読み出しおよびフラッシュデータはフラッシュインタフェース信号です。

図 5. バースト読み出しタイミング図 - 8x64 ビット、遅延 = 3 ws



1. ACLK、ARADDR、ARVALID、RDATA、RVALID、および RLAST は AXI バス信号です。
フラッシュ読み出しおよびフラッシュデータはフラッシュインタフェース信号です。

読み出しアクセスの遅延

データをフラッシュメモリから正しく読み出すには、バスクロック（aclk）の周波数およびデバイス V_{CORE} の内部電圧範囲に従ってフラッシュアクセス制御レジスタ（FLASH_ACR）でウェイトステート（LATENCY）の数を正しくプログラムする必要があります。*表 7*には、ウェイトステートの数、バスクロック周波数、および V_{CORE} レンジとの対応を記載しています。

表 7. バス周波数（ACLK）および V_{CORE} レンジによるウェイトステートの数

V_{CORE} レンジ		ウェイトステート数	最大周波数
VOS1 レベル	1.15 V - 1.26 V	0	70 MHz
		1	140 MHz
		2	210 MHz
VOS2 レベル	1.05 V - 1.15 V	0	55 MHz
		1	110 MHz
		2	165 MHz
		3	220 MHz
VOS3 レベル	0.95 V - 1.05 V	0	45 MHz
		1	90 MHz
		2	135 MHz
		3	180 MHz
		4	225 MHz

電源投入後、使用されるクロックは HSI （64 MHz）であり、FLASH_ACR レジスタでは 0x7 ウェイトステートがデフォルトで設定されます。バス周波数の変更には、フラッシュメモリにアクセスするために必要なウェイトステート数を調整するために、*セクション：バス周波数の調整*で説明するソフトウェアシーケンスを適用する必要があります。

バス周波数の調整

- バス周波数の増加
  a) 必要に応じて、FLASH_ACR レジスタの LATENCY ビットに正しいウェイトステート数（*表 7* を参照）をプログラムします。
  b) FLASH_ACR レジスタを読み出してフラッシュメモリへのアクセスに新しいウェイトステート数が反映されていることを確認します。
  c) RCC_CFGR レジスタで、フラッシュインタフェースのクロックソースやバスクロックプリスケーラを修正します。
  d) RCC_CFGR レジスタで、クロックソースステータスやバスプリスケーラの値を読み出して、新しいフラッシュインタフェースのクロックソースや新しいフラッシュインタフェースのクロックプリスケーラの値が反映されていることを確認します。
- バス周波数の減少
  a) RCC_CFGR レジスタで、フラッシュインタフェースのクロックソースやバスクロックプリスケーラを修正します。
  b) RCC_CFGR レジスタで、フラッシュインタフェースのクロックソースステータスやバスプリスケーラの値を読み出して、新しいフラッシュインタフェースのクロックソースや新しいバスクロックプリスケーラの値が反映されていることを確認します。
  c) 必要に応じて、FLASH_ACR レジスタの LATENCY ビットに正しいウェイトステート数（*表 7* を参照）をプログラムします。
  d) FLASH_ACR レジスタを読み出し、新しいウェイトステート数が反映されていることを確認します。

3.3.5 エラーコード訂正（ECC）

フラッシュメモリのデータは 266 ビットワードです。フラッシュワード（256 ビット）ごとに 10 ECC ビットが追加されます。ECC のメカニズムは SECDED アルゴリズムを採用し、以下をサポートしています。

- シングルエラー訂正
- ダブルエラー検出

1 つのエラーを検出して訂正すると、FLASH_SR1/2 レジスタの SNECCERR1/2 フラグがセットされます。FLASH_CR1/2 レジスタの SNECCERRIE ビットがセットされている場合、割り込みが生成されます。

2 つのエラーを検出すると、FLASH_SR1/2 レジスタの DBECCERR1/2 フラグがセットされ、バスエラーが生成されます。この場合、受信データは訂正されません。FLASH_CR1/2 レジスタの DBECCERRIE1/2 ビットがセットされている場合、割り込みが生成されます。

ECC エラーが検出されると、エラーを含むフラッシュワードのアドレスが FLASH_ECC_FA1/2R レジスタに保存されます。連続してエラーが検出される場合、最初のエラーに対応するアドレスのみが保存されます。エラーを生成した関連フラグがリセットされると、このレジスタは自動的にクリアされます。

3.3.6 巡回冗長検査モジュール

フラッシュインタフェースに巡回冗長検査ハードウェアモジュールが組み込まれています。このモジュールにより、フラッシュ領域の内容の整合性を検査することができます。この領域は、セクタまたは開始／終了アドレスで定義できます。

バンク 1 または 2 では、一度に 1 つの CRC 検査操作のみ開始できます。

CRC 操作は、オプションバイトの変更操作と同時に行われます。これは、たとえばオプションバイトの変更中に CRC 操作が要求された場合、CRC 操作を処理する前にオプションバイトの変更操作を完了する必要があるということです（逆の場合も同様）。

有効にすると、CRC ハードウェアモジュールはフラッシュデータを 4、16、64、または 256 フラッシュワードの塊で処理します。CRC の計算では CRC-32 （Ethernet）多項式 0x4C11DB7 をします。

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

フラッシュインタフェースでは、内部的に 4、16、64、または 256 の連続したフラッシュワード読み出しアクセスを発行します。これらのトランザクションは、他の AXI 読み出しリクエストとともに読み出しコマンドキューに入れられます。これにより、AXI 読み出しコマンドを拒否しないようにしています。キューコマンドバッファには CRC コマンドを 1 つだけ含めることができます。

バンク 1/2 で CRC 操作を設定する際に推奨するシーケンスは次のとおりです。

1. FLASH_CR1/2 の CRC_EN ビットをセットして、CRC 機能を有効にします。
2. FLASH_CRCCR1/2 の CRC_BURST フィールドで希望するデータサイズをプログラムします。
3. CRC を計算する必要があるフラッシュ領域を定義します。これには、次の 2 つの方法があります。
   - FLASH_CRCSADD1/2R および FLASH_CRCEADD1/2R をプログラミングして、領域の開始および終了アドレスをそれぞれ定義します。
   - または、FLASH_CRCCR1/2 レジスタの CRC_BY_SECT ビットを設定し、続けて FLASH_CRCCR1/2 レジスタの CRC_SECT フィールドに対象のセクタ番号をプログラミングして、対象のセクタを選択します。各 CRC_SECT のプログラミング後に ADD_SECT ビットをセットします。
4. START_CRC ビットをセットして、CRC 操作を開始します。
5. CRC_BUSY フラグがリセットされるまで待ちます。
6. FLASH_CRCDATAR レジスタにて CRC の結果を取得します。

FLASH_CRCCR1/2 レジスタの ALL_BANK ビットをセットすると、バンク全体に対して CRC を計算することができます。

注： ***CRC をPCROP または安全に保護されたユーザフラッシュ領域で実行すると、CRC の期待値が変更される場合があります。***

3.3.7 フラッシュのプログラムおよび消去操作

プログラム／消去操作の概要

フラッシュインタフェースには、シングルエラー訂正とダブルエラー検出が実装されています（*セクション 3.3.5：エラーコード訂正（ECC）* を参照）。ECC アルゴリズムは、すべて「ゼロ」のデータがすべて「ゼロ」の ECC コードを生成し、すべて「1」のデータがすべて「1」の ECC コードを生成するように構築されています。

10 ビットの ECC コードは 256 ビットのフラッシュワードにそれぞれ関連付けられているため、256 ビットの書き込み操作のみがサポートされています。

未使用状態に復元するには、セクタ全体を消去する必要があります。

フラッシュメモリインタフェースでは、複数のプログラム操作をサポートしています。

- ユーザセクタへの書き込み
- ユーザセクタの消去
- バンク 1、バンク 2 または両方のバンクの消去
- ユーザオプションバイトの変更

AXI インタフェースを通じて発行された書き込みアクセスは、フラッシュインタフェースの書き込みバッファを読み戻すことができないことを除き、バッファ可能かつキャッシュ不可能であるとみなすことができます。

内蔵フラッシュメモリは、インサーキットプログラミングまたはアプリケーション内プログラミングを使用してプログラム可能です。

書き込みバッファの構造とヒント

フラッシュインタフェースの書き込みキューバッファには 2 つのリクエストを含めることができます。その結果、フラッシュインタフェースにいくつかの書き込みアクセスが要求されると、書き込みキューバッファがフルになるまで受け入れられます。バッファがフルになると、フラッシュインタフェースは AXI バスをストールします。

操作はフラッシュインタフェースで受信した順番に実行されます。システムは、書き込みコマンドのキューと書き込みバッファがフル（2 つの書き込み操作がすでに進行中）になるまでストールしません。

通常の書き込み操作中は、PG1/2 ビットを「1」にセットする必要があります（*セクション：書き込み操作の有効化* を参照）。次に、書き込みアクセスは 256 ビットの書き込みバッファに書き込む必要があります。フルになると、内容は自動的に書き込みキューバッファに移行されます。有効な書き込み操作は、フラッシュが使用可能になり、以前に要求した操作が処理されると、すぐに実行されます。

書き込みバッファが部分的に書き込まれている場合、書き込みバッファが完全に書き込まれる前に書き込み操作を強制的に実行できます。これには、FLASH_CR1/2 の FW1/2 ビットをセットします。この特殊な例では、まだ書き込まれていないビットが自動的に「ハイ」値に設定されます。書き込みバッファで「ロー」値に設定されたビットが無い場合、FW1/2 ビットは無効です。

未使用のフラッシュワード以外、上書きはお勧めしません。ECC コードが一致しない場合があります。フラッシュインタフェースでは ECC エラーを体系的に報告することができます。対応可能な有効なシナリオとしては、すべてゼロのデータで上書きを行う方法です。この場合、対応する ECC コードもすべてゼロになります。したがって、読み出し中に ECC エラーが検出されることはありません。

書き込みバッファは、インクリメンタル書き込みバーストアクセスもサポートしています。32 バイトの整列アドレスを超えない場合にのみ、ラップバースト書き込みアクセスを実行できます（フラッシュワード = 256 ビット）。

プログラミングパラメータの設定

ユーザアプリケーションでは、プログラム／消去操作を行う前にプログラミングパラメータを設定する必要があります。

1. フラッシュ設定レジスタ（FLASH_CR1/2）をアンロックおよびプログラムします。

   フラッシュインタフェースには 2 つのフラッシュ制御レジスタが実装されています：バンク 1 には FLASH_CR1、バンク 2 には FLASH_CR2。リセット後、これらのレジスタはフラッシュメモリ上での不要な操作（電気妨害などによる）を避けるため、書き込み保護されています。FLASH_CR1/2 レジスタのアンロックには、次のシーケンスを使用します。

   a) FLASH_KEYR1/2 レジスタに KEY1 = 0x45670123 を書き込みます。

   b) FLASH_KEYR1/2 レジスタに KEY2 = 0xCDEF89AB を書き込みます。

   FLASH_CR1/2 レジスタの LOCK1/2 ビットは自動的にクリアされます。シーケンスを誤ると、次のシステムリセットまで FLASH_CR1/2 がロックされます。この場合、バスエラーが生成されます。

   さらに、次の操作を実行すると、FLASH_CR1/2 はロックされたままとなりバスエラーが生成されます。

   – 3 番目のキー値をプログラミングする

   – FLASH_CR1/2 が完全にアンロックされる前に同じバンクに属する FLASH_KEYR1/2 とは異なるレジスタに書き込む（KEY1 はプログラミング済み、ただし KEY2 はプログラミング未了）

   – KEY1 または KEY2 に 1つ 未満のフラッシュワードを書き込む

   FLASH_CR1/2 レジスタを再度ロックするには、対応する LOCK1/2 ビットをソフトウェアで「1」にセットします。

注： ***FLASH_KEYR1/2 レジスタには、ワードレベルでのみアクセス可能です。***

2. フラッシュオプション設定レジスタ（FLASH_OPTCR）をアンロックします。

   オプションバイトを変更するには、FLASH_OPTCR レジスタをアンロックします。オプションバイトは両方のバンクに共通であるため、フラッシュインタフェースには FLASH_OPTCR レジスタが 1 つだけ存在し、バンクスワッピングによるFLASH_OPTCR への影響はありません。レジスタのエイリアスが作成されるため、2 つの異なるアドレスでアクセス可能です。

   – 0x018

   – 0x118

   リセット後、OPTLOCK ビットは「1」にセットされ、FLASH_OPTCR はロックされます。FLASH_OPTCR レジスタのアンロックには、次のシーケンスが必要です。

   a) FLASH_OPTKEYR レジスタの OPTKEY1 に 0x08192A3B を書き込みます。

   b) FLASH_OPTKEYR レジスタの OPTKEY2 に 0x4C5D6E7F を書き込みます。

   シーケンスの実行後、OPTLOCK ビットはクリアされ、FLASH_OPTCR はアンロックされます。キーシーケンスを誤ると、次のシステムリセットまで FLASH_OPTCR がロックされ、バスエラーが生成されます。

   さらに、次の操作を実行すると、FLASH_OPTCR はロックされたままバスエラーが生成されます。

   – 3 番目のキー値をプログラミングする

   – FLASH_OPTCR が完全にアンロックされる前に異なるレジスタに書き込む（OPTKEY1 はプログラミング済み、ただし OPTKEY2 はプログラミング未了）

   – OPTKEY1 または OPTKEY2 に 1 未満のフラッシュワードを書き込む

   FLASH_OPTCR レジスタを再度ロックするには、OPTLOCK ビットをソフトウェアで「1」にセットします。

注： ***FLASH_OPTCR レジスタには、ワードレベルでのみアクセス可能です。***

3. プログラミング並列度を設定します。

   並列度は、書き込み操作中 1 回で「0」を書き込めるビットの最大数です。プログラミング並列度はセクタおよびバンク消去時にも使用されます。

   プログラミング並列度にハードウェアの制限はありません。ユーザはアプリケーションの要件に応じて異なる並列度を選択できます。並列度が低いほどプログラム操作時のピーク消費量は低くなりますが、実行時間は長くなります。

   並列度は、FLASH_CR1/2 の PSIZE1/2 ビットを通じて設定されます。バンク 1 およびバンク 2 には 2 つの異なる値を定義できます（*表 8* を参照）。

表 8. 並列度パラメータ

PSIZE1/2	並列度
00	バイト（8 ビット）
01	ハーフワード（16 ビット）
10	ワード（32 ビット）
11	ダブルワード（64 ビット）

4. プログラミング遅延を設定します。

   プログラミング操作のタイミング制約は、パフォーマンスに直接的な影響を及ぼすフラッシュインタフェースの周波数に依存します。タイミング制約が厳しすぎるとフラッシュメモリが正しく動作せず、緩すぎるとプログラミング速度が最適ではなくなります。

   このため、ユーザは FLASH_ACR レジスタの WRHIGHFREQ パラメータを通じて最適なプログラミング遅延の調整を行う必要があります。この設定レジスタは、両方のバンクに共通です。フラッシュインタフェースの周波数に応じたプログラミング遅延の推奨値については、*表 9* を参照してください。

表 9. プログラミング速度

WRHIGHFREQの推奨値	フラッシュインタフェース周波数（MHz）
00	85
01	185

WRHIGHFREQ の変更後、読み直すことで新しい値が正しく反映されていることを確認します。

注： ***プログラム／消去操作中にプログラミング速度が変更されると、新しい値は現在の操作が完了した後で初めて反映されます。***

書き込み操作の有効化

プログラミングを行う前に、FLASH_CR1/2 レジスタの PG1/2 ビットをセットする必要があります。FLASH_CR1/2 レジスタを事前にアンロックする必要があります。PG1/2 ビットが「0」に設定されている間に要求された書き込みアクセスはすべて拒否されます。この場合、バス上でエラーは生成されませんが、PGSERR1/2 フラグは立てられます。

書き込み保護の確認

書き込みトランザクション対象の保護特性は、フラッシュメモリへの有効な書き込みの直前で、書き込みキューバッファから出力されるときに確認されます。フラッシュインタフェースが AXI 書き込みリクエストを受け付けると、確認は行われません。書き込み保護違反が検出された場合、書き込み操作はキャンセルされ、FLASH_SR1/2 レジスタの保護違反に対応するフラグが立てられます。書き込み操作が有効である場合、10 ビットの ECC コードは 256 ビットのデータに連結され、フラッシュメモリへの書き込みが有効に実行されます。

書き込み保護フラグは、新しいプログラム／消去操作を実行する前にクリアする必要があります。

書き込みステータスのビジーフラグ

FLASH_SR1/2 レジスタにある 3 つの異なるステータスフラグを、各バンクで使用できます。これらは進行中の書き込み操作のステータスを示します。

- **BSY1/2**：このビットは、フラッシュメモリに対する有効な書き込み、消去、またはオプションバイトの変更操作が進行中であることを示します。
- **QW1/2**：このビットは、プログラム、消去、またはオプションバイトの変更操作がペンディング中であることを示します。このビットは、書き込み操作が完了するまでハイを保持します。BSY1/2 ステータスビットより優先されるビットです。
- **WBNE1/2**：このビットは書き込みバッファが空ではないことを示します。書き込みコマンドがキューに入れられると、すぐにリセットされます。

注： ***書き込みバッファは 32 バイトのアドレスに整列された 256 ビットのフラッシュワードに対応するため、別の書き込み操作の進行中に新しいフラッシュワードの書き込みを開始することはできません。***

簡単な書き込みシーケンス（推奨）

1. 対象バンク（バンク 1/2）の FLASH_CR1/2 レジスタの PG1/2 ビットをセットします（*セクション：書き込み操作の有効化*を参照）。
2. 対象メモリ領域の保護を確認します（*セクション：書き込み保護の確認*を参照）。
3. 32 バイトの整列アドレスで始まる 32 バイトデータに対応するフラッシュワードを 1 つ書き込みます。
4. BSY1/2 が立てられたことを確認し、リセットされるまで待ちます（*セクション：書き込みステータスのビジーフラグ*を参照）。

最適なブロック書き込みシーケンス

このシーケンスは、フラッシュメモリにブロックをプログラムする際に使用できます。

1. 対応するバンク（バンク 1/2）の FLASH_CR1/2 レジスタの PG1/2 ビットをセットします（*セクション：書き込み操作の有効化*を参照）。
2. 対象メモリ領域の保護を確認します（*セクション：書き込み保護の確認*を参照）。
3. ブロック全体が移行されるまで、32 データバイト（フラッシュワード）を連続的に書き込みます。フラッシュワードは、それぞれ 32 バイトの整列アドレスで始める必要があります。

STOP または STANDBY モードへのシステム切り替えを可能にするためのフラッシュインタフェースの解放

いずれかのビジーフラグがアクティブなとき、マイクロコントローラでは D1 ドメインを STOP または STANDBY モードに切り替えることはできません（*セクション 6：電源制御（PWR）*を参照）。フラッシュインタフェースを解放するには、2 つの方法があります。

- 次のいずれかの操作で WBNE1/2 ビジーフラグをリセットします。
  a) データが欠落している書き込みバッファを完了します。
  b) FLASH_CR1/2レジスタ の FW1/2 ビットを有効化して、欠落しているデータを入力せずに書き込み操作を強制的に実行します。これにより、すべての欠落しているデータは強制的に「ハイ」値に設定されます。
  c) FLASH_CR1/2 レジスタの PG1/2 ビットがリセットされます。これにより、書き込みバッファが無効になり、結果的に内容が失われます。
- BSY1/2 および QW1/2 のビジービットを、クリアされるまでポーリングします。これは、記録されたすべての書き込み、消去、およびオプションの変更操作が完了したことを示します。

これにより、マイクロコントローラは STOP または STANDBY モードに切り替えることができます。

フラッシュセクタの消去

128 KB のユーザセクタを消去するには、次の手順に従います。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します（詳細については、*セクション 3.3.9：フラッシュインタフェースエラーフラグ*を参照）。
2. *セクション ：プログラミングパラメータの設定*を参照して FLASH_CR1/2 レジスタをアンロックします。
3. 対応する FLASH_CR1/2 レジスタの SER1/2 ビットおよび SNB1/2 ビットフィールドをセットします。SER1/2 はセクタの消去操作を示し、SNB1/2 には対象のセクタ番号が含まれます。
4. FLASH_CR1/2 レジスタの START1/2 ビットをセットします。
5. 対応する FLASH_SR1/2 レジスタの BSY1/2 ビットがクリアされるまで待ちます。

注： ***セクタ消去（BER1/2 ビットをセット）と同時にバンク消去が要求されると、セクタ消去操作よりバンク消去操作が優先されます。***

フラッシュバンク消去

安全かつ保護されたデータを含むセクタを除くすべてのバンクセクタを消去するには、次の手順に従います。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します（詳細については、*セクション 3.3.9：フラッシュインタフェースエラーフラグ*を参照）。
2. *セクション ：プログラミングパラメータの設定*を参照して FLASH_CR1/2 レジスタをアンロックします。
3. PCROP 保護領域が存在する場合は、FLASH_PRAR_CUR1/2 および FLASH_PRAR_PRG1/2 レジスタの DMEP1/2 ビットを両方とも 0 に設定します。
4. 安全な領域のみが存在する場合は、FLASH_SCAR_CUR1/2 および FLASH_SCAR_PRG1/2 レジスタの DMES1/2 ビットを両方とも 0 に設定します。
5. 対象バンクに対応する FLASH_CR1/2 レジスタの BER1/2 ビットをセットします。
6. バンク消去操作を開始するには、FLASH_CR1/2 レジスタの START1/2 ビットをセットします。対応する FLASH_SR1/2 レジスタの BSY1/2 ビットがクリアされるまで待ちます。

注： ***BER1/2 および START1/2 ビットは一緒にセットすることができ、手順 8 および 9 は統合することができます。***

バンク消去（SER1/2 ビットをセット）と同時にセクタ消去が要求されると、バンク消去操作がセクタ消去操作より優先されます。

自動保護解除を伴うフラッシュバンク消去

RDP 回帰を実行せずに、安全かつ保護されたデータを含むすべてのバンクセクタを消去するには、次の手順に従います。

1. FLASH_SR1/2 レジスタの BSY1/2 ビットを監視し、進行中のフラッシュメモリ操作がないことを確認します。
2. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します（詳細については、*セクション 3.3.9：フラッシュインタフェースエラーフラグ* を参照）。
3. *セクション ： プログラミングパラメータの設定* を参照して FLASH_OPTCR レジスタをアンロックします。
4. PCROP 保護領域が存在する場合は、FLASH_PRAR_CUR1/2 または FLASH_PRAR_PRG1/2 レジスタのどちらかで DMEP1/2 ビットをセットします。さらに、次のようにその差が負になるように PCROP 領域の終了および開始アドレスをプログラムします。

   PROT_AREA_END1/2 < PROT_AREA_START1/2
5. 安全のみの領域が存在する場合は、FLASH_SCAR_CUR1/2 または FLASH_SCAR_PRG1/2 レジスタのどちらかで DMES1/2 ビットをセットします。さらに、次のようにその差が負になるようにセキュア領域の終了および開始アドレスをプログラムします。

   SEC_AREA_END1/2 < SEC_AREA_START1/2

注： ***手順 5 は、ST セキュアライブラリまたはセキュアモードで実行中のアプリケーションのみ実行可能です。ただし、この手順は安全のみの領域が存在する場合に限り実行する必要があります。***

6. FLASH_WPSN_PRG1/2R の WRPSn1/2 ビットをすべて 1 にセットし、すべてのセクタ書き込み保護を無効化します。
7. FLASH_CR1/2 レジスタをアンロックします。
8. 対象バンクに対応する FLASH_CR1/2 レジスタの BER1/2 ビットをセットします。
9. 保護解除操作を伴うバンク消去を開始するには、FLASH_CR1/2 レジスタの START1/2 ビットをセットします。対応する FLASH_SR1/2 レジスタの BSY1/2 ビットがクリアされるまで待ちます。この時点で、PCROP 保護されたデータや安全なデータから成るセクタを含むバンク全体が、バンク消去操作によって消去され、またオプションバイト変更が自動的に実行され、すべての保護が無効化されます。

注： ***BER1/2 および START1/2 ビットは一緒にセットすることができ、手順 8 および 9 は統合することができます。***

警告： 上述のシーケンスに関する次の警告に注意してください。

- **別のバンクの保護パラメータを変更しながら、1 つのバンク上で上述のシーケンスを実行することはできません。**
- **上述のオプションバイト以外を変更することはできません。また、保護解除リクエストを伴うバンク消去の対象ではないバンクで保護を変更することはできません。**
- **上述のイベントのいずれかまたは両方が発生した場合、簡単なバンク消去が実行され、オプションバイトの変更は行われず、オプション変更エラーもセットされません。**

フラッシュ全体消去

安全および保護されたデータを含むセクタを除く両方のバンクのすべてのセクタを同時に消去するには、ユーザアプリケーションで次のように FLASH_OPTCR レジスタの MER ビットを 1 にセットします。

1. FLASH_SR1/2レジスタの BSY1/2 ビットを監視し、進行中のフラッシュメモリ操作がないことを確認します。
2. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します（詳細については、*セクション 3.3.9：フラッシュインタフェースエラーフラグ* を参照）。
3. *セクション ：プログラミングパラメータの設定* を参照して 2 つの FLASH_CR1/2 レジスタと FLASH_OPTCR レジスタをアンロックします。
4. PCROP 保護領域が存在する場合は、FLASH_PRAR_CUR1/2 および FLASH_PRAR_PRG1/2 レジスタの DMEP1/2 ビットを両方とも 0 に設定します。
5. 安全のみの領域が存在する場合は、FLASH_SCAR_CUR1/2 および FLASH_SCAR_PRG1/2 レジスタの DMES1/2 ビットを両方とも 0 に設定します。
6. FLASH_OPTCR レジスタの MER ビットに 1 をセットします。これにより、BER1、BER2、START1、および START2 が自動的に 1 にセットされ、両方のバンク上でバンク消去操作が開始されます。対応する FLASH_SR1/2 レジスタの BSY1/2 ビットがクリアされるまで待ちます。

自動保護解除を伴うフラッシュ全体消去

RDP 回帰を実行せずに、安全かつ保護されたデータから成るセクタを含む両方のバンクのセクタを同時にすべて消去するには、次の手順に従います。

1. FLASH_SR1/2 レジスタの BSY1/2 ビットを監視し、進行中のフラッシュメモリ操作がないことを確認します。
2. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。
3. 2 つの FLASH_CR1/2 レジスタと FLASH_OPTCR レジスタをアンロックします。
4. PCROP 保護領域が存在する場合は、FLASH_PRAR_CUR1/2 または FLASH_PRAR_PRG1/2 レジスタのどちらかで DMEP1/2 ビットをセットします。さらに、その差が負になるように PCROP 領域の終了および開始アドレスをプログラムします。
5. 安全のみの領域が存在する場合は、FLASH_SCAR_CUR1/2 または FLASH_SCAR_PRG1/2 レジスタのどちらかで DMES1/2 ビットをセットします。さらに、その差が負になるようにセキュア領域の終了および開始アドレスをプログラムします。

注： ***手順 5 は、ST セキュアライブラリまたはセキュアモードで実行中のアプリケーションのみ実行可能です。ただし、この手順は安全のみの領域が存在する場合に限り実行する必要があります。***

6. FLASH_WPSN_PRG1/2R の WRPSn1/2 ビットをすべて 1 にセットし、すべてのセクタ書き込み保護を無効化します。
7. FLASH_OPTCR レジスタの MER ビットに 1 をセットし、対応する FLASH_SR1/2 レジスタの BSY1/2 ビットがクリアされるまで待ちます。この時点で、自動保護解除を伴うフラッシュバンク消去が両方のバンク上で実行されます。

3.3.8 ユーザオプションバイトの変更

ユーザオプションバイトの変更操作は、フラッシュメモリオプションバイト領域として保存された設定や保護設定を変更するために使用できます（詳細なリストについては、*表 10：フラッシュユーザオプションバイトの一覧*を参照）。

フラッシュインタフェースには 2 セットのオプションバイトレジスタがあります。

- 最初のレジスタセットには現在のオプションバイトの値が含まれます。これらの名前には_CUR の拡張子が含まれます。すべての「_CUR」レジスタは読み出し専用です。値は電源投入後、またはオプションバイトの変更操作後に自動的にロードされます。
- 2つ目のレジスタセットでは、オプションバイトを変更することができます。これらの名前には _PRGの拡張子が 含まれます。すべての「_PRG」レジスタは読み出し／書き込みモードでアクセスすることができます。

FLASH_OPTCR レジスタの OPTLOCK ビットがセットされると、FLASH_OPTXX_PRG レジスタの変更は禁止されます。

OPTSTART ビットが「1」にセットされると、フラッシュインタフェースでは現在の値（_CUR）と新しい値（_PRG）を比較して、少なくとも 1 オプションバイトをプログラミングする必要があるかどうかを確認します。

オプションバイトの変更操作は、次の条件が満たされた場合にのみ許可されます。

- 現在の RDP がレベル 2 以外である場合（レベルにかかわらず変更可能な スワップビットを除く）。
- PCROP 保護領域が存在する場合で、新しくプログラミングされた領域が現在の領域と同じがそれよりも大きい必要があります（回帰レベルである場合を除く）。
- 安全保護領域が存在する場合、セキュアモードが設定されたときにこの領域を変更できます（*セクション 4：セキュリティメモリ管理*を参照）。
- レベルの回帰（レベル 1 からレベル 0 への変更）の場合、DMEP1/2 オプションビットを制限なくセットでき、それ以外の場合は「1」のみをセットできます。
- DMES1/2 オプションビットは、同時にレベルの回帰（レベル 1 からレベル 0 への変更）を実行した場合にのみ FLASH_SCAR_PRG1/2 で変更でき、それ以外の場合は「1」のみをセットできます。
- SECURITY オプションビットがセットされ、PCROP またはセキュア領域が存在する場合、レベルの回帰を実行して その差が負になるようにPCROP またはセキュア領域の終了および開始アドレスをプログラミングすることで、保護をリセット（SECURITY ビットをリセット）することができます。

これらの条件が 1 つでも満たされない場合、フラッシュインタフェースでは FLASH_OPTSR_CUR レジスタの OPTCHANGEERR フラグを「1」にセットし、オプションバイトの変更操作を停止します。

オプションバイトの変更シーケンス

ユーザオプションバイトを変更するには、次の手順に従います。

1. *セクション ：プログラミングパラメータの設定*で説明している特定のシーケンスに従って、OPTLOCK ビットをアンロックします。
2. 対応するオプションレジスタ（FLASH_XXX_PRG1/2）に希望するオプションバイトの値を書き込みます。
3. FLASH_OPTCR レジスタでオプションスタートビット（OPTSTART）をセットします。
4. OPT_BUSY ビットがクリアされるまで待ちます。

注： ***オプションバイトの変更中にリセットまたはパワーダウンが発生した場合は、古いオプションバイト値が保持されます。新しい値をプログラムするには、新しいオプションバイトの変更シーケンスが必要です。***

3.3.9 フラッシュインタフェースエラーフラグ

フラッシュインタフェースは、プログラム／消去操作中にエラーが発生した場合に報告を行います。ユーザアプリケーションでは、エラーごとに個別に割り込みを有効化することができます。新しい書き込みを開始する前に一部のエラーをクリアする必要があります。

各バンクには、どのバンクからエラーが生成されたかを識別するための専用のエラーフラグのセットがあります。これは、フラッシュステータスレジスタ 1 または 2（FLASH_SR1/2）で使用できます。

各エラーフラグは、FLASH_CR1/2レジスタの割り込みイネーブルビットに関連付けられています。割り込みイネーブルビットを「1」にセットすると、対応するエラーフラグを「1」に設定したときに割り込みが生成されます。エラーフラグをクリアするには、FLASH_CCR1/2 レジスタの対応するビットをセットします。

書き込み保護エラー（WRPERR1）

この WRPERR1/2 フラグはスティッキービットで、ユーザアプリケーションが保護された領域をプログラム／消去しようとした場合にハードウェアによってセットされます。このフラグが立てられると、操作は停止され、変更は行われません。このフラグは、FLASH_CCR1/2 レジスタの CLR_WRPERR1/2 ビットを「1」にセットするとクリアされます。FLASH_CR1/2 レジスタの WRPERRIE1/2 ビットが「1」にセットされている場合、WRPERR1/2 フラグが立てられたときに割り込みが生成されます。

新しい書き込み操作を開始する前に WRPERR1/2 フラグをクリアする必要があります。そうしないと、シーケンスエラー（PGSERR1/2 ビット）が生成され、次のプログラム操作が停止されます。

プログラミングシーケンスエラー（PGSERR1/2）

この PGSERR1/2 フラグはスティッキービットで、プログラミングシーケンスが正しくない場合にハードウェアによってセットされます。次のいずれかの条件が満たされた場合に、このビットは「1」にセットされます。

- PG1/2 ビットがセットされていない状態で、AXI バスから書き込み操作が要求された場合。
- 書き込み保護エラーフラグ（WRPERR1/2）がリセットされていない状態で、新しい書き込み操作が要求された場合。
- 不整合エラー（INCERR1/2）がリセットされていない状態で、新しい書き込み操作が要求された場合。
- フラッシュ操作エラー（OPERR1/2）がリセットされていない状態で、新しい書き込み操作が要求された場合。

PGSERR1/2 フラグが立てられると、プログラム操作は停止されます。このフラグをクリアするには、FLASH_CCR1/2 レジスタの CLR_PGSERR1/2 ビットを「1」にセットします。FLASH_CR1/2レジスタの PGSERRIE1/2 ビットがセットされている場合、PGSERR1/2 フラグが立てられたときに割り込みが生成されます。

新しい書き込み操作を開始する前に PGSERR1/2 をクリアする必要があります。

ストローブエラー（STRBERR1/2）

この STRBERR1/2 フラグは、ユーザアプリケーションによって書き込みバッファに同じバイトが数回書き込まれた場合に、ハードウェアによってセットされるスティッキービットです。書き込み操作は停止されず、アプリケーションではエラーを無視し、現在の書き込み操作を続行し、また新しい書き込み操作を要求することができます。このフラグは、FLASH_CCR1/2レジスタ の CLR_STRBERR1/2 ビットをセットするとクリアされます。FLASH_CR1/2レジスタ の STRBERRIE1/2 ビットがセットされている場合、STRBERR1/2 フラグが立てられたときに割り込みが生成されます。

新しい書き込み操作を開始する前に、STRBERR1/2フラグ を必ずクリアしなければいけないわけではありません。

不整合エラー（INCERR1/2）

この INCERR1/2 フラグはスティッキービットで、次の場合にハードウェアによってセットされます。

- ユーザアプリケーションが最初のバーストでフラッシュワードへの書き込み操作を開始し、この進行中の書き込み操作が完了する前に、別のフラッシュワードアドレスに新しいバースト書き込みを送信した場合。この場合、書き込みバッファは空になり、新しいデータは拒否されて、INCERR1/2 フラグが立てられます。
- マスタが最初のバーストでフラッシュワードへの書き込み操作を開始し、この最初のマスタが開始した操作が完了する前に、別のマスタが同じ、または新しいフラッシュワードアドレスに新しい書き込みバーストを送信した場合。この場合、書き込みバッファは空になり、新しいデータは拒否されて、INCERR1/2 フラグが立てられます。
- マスタが発行したラップバーストが 2 つ以上のフラッシュワードアドレスと重なる場合（ラップバーストの最大サイズはフラッシュワードサイズ、つまり 256 ビットに制限されています）

INCERR1/2 フラグが立てられると、操作は停止され、書き込み操作は実行されません。このフラグをクリアするには、FLASH_CCR1/2 レジスタの CLR_INCERR1/2 ビットをセットします。FLASH_CR レジスタの INCERRIE1/2 ビットがセットされている場合、INCERR1/2 フラグが立てられたときに割り込みが生成されます。

新しいプログラムまたは消去操作を開始する前に INCERR1/2 フラグをクリアする必要があります。そうしないと、シーケンスエラー（PGSERR1/2 ビット）が生成され、次のプログラム操作が停止されます。

操作エラー（OPERR1/2）

この OPERR1/2 フラグはスティッキービットで、フラッシュメモリが書き込みまたは消去操作中にエラーを検出した場合に、ハードウェアによってセットされます。このエラーの原因は、サイクル問題による誤ったフラッシュメモリの挙動である場合があります。

このフラグをクリアするには、FLASH_CCR1/2 レジスタの CLR_OPERR1/2 ビットを「1」にセットします。FLASH_CR1/2 レジスタの OPERRIE1/2 ビットが「1」の場合、OPERR1/2 フラグが立てられたときに割り込みが生成されます。

新しいプログラム操作を開始する前に OPERR1/2 をクリアする必要があります。そうしないと、シーケンスエラー（PGSERR1/2 ビット）が生成され、新しいプログラミングが停止されます。

3.3.10 バンク 1 およびバンク 2 での同時読み出し／プログラム／消去

フラッシュメモリは、2 つの個別のバンクに分割されます。フラッシュインタフェースでは、各バンクでで同時に異なる操作を実行することができます。バンク 1の読み出し、プログラムまたは消去操作を実行しながら、バンク 2 の読み出し、プログラムまたは消去操作を実行できます。

レベルの回帰が必要な場合、オプションバイトの変更操作には例外が存在します。この場合、両方のバンクが使用可能でなければなりません。

3.3.11 フラッシュオプションバイト

オプションバイトの説明

オプションバイトは、アプリケーションの要件に応じてエンドユーザが設定します。ユーザオプションバイトは、フラッシュインタフェースレジスタのインタフェースからアクセスできます。オプションバイトのプログラミングシーケンスについては、*セクション：オプションバイトの変更シーケンス*で説明されています。

フラッシュインタフェースのオプションバイトストレージの信頼性を高めるために、オプションバイトデータはそれぞれフラッシュメモリ内のエラーコード訂正（ECC）に関連付けられています。

表 10 では、フラッシュインタフェースレジスタで使用可能なすべてのユーザオプションバイトのリストについて説明しています。

表 10. フラッシュユーザオプションバイトの一覧

ユーザオプション	レジスタ(1)	サイズ（ビット）	出荷時にプログラムされたデフォルト値(2)
SWAP_BANK	FLASH_OPTSR_CUR	1	0x0
RDP	FLASH_OPTSR_CUR	8	0xAA
BOR_LEV	FLASH_OPTSR_CUR	2	0x1
BOOT_ADD0	FLASH_BOOT_CURR	16	0x0800
BOOT_ADD1	FLASH_BOOT_CURR	16	0x1FF0
DMEP1	FLASH_PRAR_CUR1	1	0x0
DMES1	FLASH_SCAR_CUR1	1	0x0
WRPSn1	FLASH_WRP_CUR1R	8	0xFF
PROT_AREA_START1	FLASH_PRAR_CUR1	12	0xFF
PROT_AREA_END1	FLASH_PRAR_CUR1	12	0x00
SEC_AREA_START1	FLASH_SCAR_CUR1	12	0xFF
SEC_AREA_END1	FLASH_SCAR_CUR1	12	0x00
DMEP2	FLASH_PRAR_CUR2	1	0x0
DMES2	FLASH_SCAR_CUR2	1	0x0
WRPSn2	FLASH_WPSN_CUR1R	8	0xFF
PROT_AREA_START2	FLASH_PRAR_CUR2	12	0xFF
PROT_AREA_END2	FLASH_PRAR_CUR2	12	0x00
SEC_AREA_START2	FLASH_SCAR_CUR2	12	0xFF
SEC_AREA_END2	FLASH_SCAR_CUR2	12	0x00
IWDG1_HW	FLASH_OPTSR_CUR	1	0x1
SECURITY	FLASH_OPTSR_CUR	1	0x0
ST_RAM_SIZE	FLASH_OPTSR_CUR	2	0x10
nRST_STBY_D1	FLASH_OPTSR_CUR	1	0x1
nRST_STOP_D1	FLASH_OPTSR_CUR	1	0x1
FZ_IWDG_SDBY	FLASH_OPTSR_CUR	1	0x1
FZ_IWDG_STOP	FLASH_OPTSR_CUR	1	0x1

表 10. フラッシュユーザオプションバイトの一覧（続き）

ユーザオプション	レジスタ(1)	サイズ（ビット）	出荷時にプログラムされたデフォルト値(2)
PERSO_OK	FLASH_OPTSR_CUR	1	0x1
IO_HSLV	FLASH_OPTSR_CUR	1	0x0

1. ビットマッピングと詳細なビットの説明については、*セクション 3.5：フラッシュレジスタ*を参照してください。
2. 出荷時のオプションバイトのデフォルト値は最初のオプションバイトの変更が行われる前のデフォルト値です。

オプションバイトローディング

フラッシュユーザオプションバイトをロードするには、3 つの方法があります。

1. パワーオン時：

   フラッシュインタフェースは、フラッシュメモリから自動的にユーザオプションバイトをロードします。ローディングシーケンス中、デバイスはリセット状態を維持し、フラッシュインタフェースにはアクセスできません。

2. フラッシュインタフェースを含む D1 電源ドメインが STANDBY モードから RUN モードに切り替えられた場合：

   この場合、フラッシュインタフェースはパワーオンシーケンス中と同様に挙動しますが、デバイスはリセット状態に維持されません。

3. ユーザアプリケーションが、フラッシュインタフェースレジスタを通じてオプションバイトの内容を変更した場合：

   この場合、フラッシュインタフェースがオプションバイトセクタを再プログラムした後で、フラッシュインタフェースのオプションレジスタを更新するためにオプションバイトの再ロードが実行されます。

特定のオプションビットを変更する際の規則

一部のオプションバイトフィールドは、新しい値で更新する前に、特定の規則に従う必要があります。これらのフィールドと関連する制約について、以下で説明します。

- セキュア領域のサイズ

  セキュア領域のサイズは FLASH_SCAR_CUR1/2 および FLASH_SCAR_PRG1/2 レジスタ内の SEC_AREA_START1/2、SEC_AREA_END1/2、および DMES1/2 によって制御されます。これらのオプションバイトは、セキュアモードで変更できます。SEC_AREA_END1/2 フィールドが SEC_AREA_START1/2 よりも低い場合は、セキュア領域のサイズは null です。
  SEC_AREA_START1/2 = SEC_AREA_END1/2 の場合、バンクはすべて安全に保護されています。詳細については、*セクション 4：セキュリティメモリ管理* を参照してください。

- DMES1/2

  このオプションビットをセットすると、レベルの回帰（レベル 1 からレベル 0）中、またはすべての保護が解除されるバンク消去中に、セキュア領域が消去されます。それ以外の場合は保持されます。このビットのリセットは、レベルの回帰（レベル 1 からレベル 0）、またはすべての保護が解除されるバンク消去が同時に要求された場合にのみ実行されます。詳細については、*セクション 4：セキュリティメモリ管理* を参照してください。

- PCROP 領域のサイズ（実行のみ）

  PCROP 領域のサイズは FLASH_PRAR_CUR1/2 および FLASH_PRAR_PRG1/2 レジスタの PROT_AREA_START1/2、PROT_AREA_END1/2、および DMEP1/2 によって制御されます。PCROP 領域は制限なく増やすことができます。PCROP 領域を減らしたり解除したりするには、同時にレベルの回帰を要求する必要があります。PCROP 領域を削除するには、すべての保護が解除されるバンク消去を実行します。PROT_AREA_END1/2 フィールドが PROT_AREA_START1/2 よりも低

い場合は、PCROP 領域のサイズは null です。PROT_AREA_START1/2 = PROT_AREA_END1/2 の場合、すべてのバンクはPCROP 保護されます。詳細については、*セクション 4：セキュリティメモリ管理* を参照してください。

- DMEP1/2

  このビットをセットすると、レベルの回帰中、またはすべての保護が解除されるバンク消去中に、PCROP 領域が消去されます。このビットがセットされないと、レベルの回帰（レベル 1 からレベル 0）中に PCROP 領域が保持されます。このビットの設定に制約はありません。ただし、このビットのリセットは、レベルの回帰またはバンク消去が同時に要求された場合にのみ実行されます。

- 読み出し保護（RDP）
  - RDP がレベル 2 にセットされると、下位のレベルに戻ることはできません。ユーザアプリケーションがそのような操作を試みると、エラーが発生して要求された変更は無視されます。
  - RDP がレベル 1 にセットされると、ユーザアプリケーションが RDP レベル 2 への切り替えを要求した場合に、RDP オプションバイトを制限なく変更できるようになります。
  - ただし、対応するバンクの FLASH_SCAR_CUR1/2 の DMES1/2 および FLASH_PRAR_CUR1/2 の DMEP1/2 ビットが「1」にセットされると、レベル 1 からレベル 0 に切り替えることでフラッシュ全体消去が発生します。
  - ユーザアプリケーションがレベルの回帰中に PCROP 領域またはセキュア領域を保持したい場合、レベルの回帰を実行する前に DMEP1/2 または DMES1/2 ビットを「0」に設定しておく必要があります。
  - RDP がレベル 0 に設定されると、制限なくレベル 1 またはレベル 2 に切り替えられるようになります。
- セクタ書き込み保護（WRPSn1/2）

  特定の規則はありません。
- セキュリティモード（SECURITY）

  このオプションビットをセットすることで、デバイスのセキュリティモードが有効になります。このビットはセキュア領域や PCROP 領域が存在しない場合はクリアすることができ、それ以外の場合はレベルの回帰またはセキュリティブートサービスを実行してビットをリセットする必要があります。詳細については、*セクション 4：セキュリティメモリ管理*を参照してください。
- ST_RAM_SIZE

  このオプションバイトの値は制限なく増やすことができます。この値の低下は、ルートセキュアサービスを通じてのみ許可されます。このオプションは、SECURITY オプションバイトがセットされたときのみ有効です。詳細については、*セクション 4：セキュリティメモリ管理*を参照してください。

3.3.12 保護メカニズム

フラッシュインタフェースには異なる保護メカニズムが実装されています。

- 読み出し保護（RDP）
- PCROP（独自仕様コード読み出し保護）：実行のみ可能な領域
- 安全保護
- セクタ書き込み保護

RDP（読み出し保護）

フラッシュユーザ領域は、信頼できないコードによる読み出し操作から保護できます。3 種類の読み出し保護レベルがあります（*表 11：RDP 値と読み出し保護レベルの比較*、*表 12：許可されたアクセスと読み出し保護レベルの比較*、および*図 6：保護の遷移スキーム*を参照）。

- レベル 0：読み出し保護なし

  読み出し保護オプションバイト（RDP）に 0xAA を書き込んで読み出し保護レベルをレベル 0 に設定すると、フラッシュメモリやバックアップ SRAM に対するすべての読み出し／プログラム操作（他の保護が設定されていない場合）がブート設定（フラッシュユーザブート、デバッグ、RAM からのブート）にかかわらず可能となります。

- レベル1

  読み出し保護レベル 1 は、RDPオプションバイトに何らかの値（それぞれレベル 0 およびレベル 2 を設定する 0xAA および0xCC を除く）を書き込むとアクティブになります。読み出し保護レベル 1 を設定すると次のようになります。

  – デバッガが接続されている場合やブート設定がユーザフラッシュと異なる場合は、フラッシュメモリやバックアップ SRAM に一切アクセス（読み出し、消去、プログラム）できなくなります。フラッシュメモリへの読み出しアクセスが要求されると、バスエラーが生成されます。

  – フラッシュメモリからブートする場合（デバッガが接続されていない場合）、ユーザコードからのフラッシュメモリおよびバックアップ SRAMの読み出し、消去、およびプログラムアクセスが許可されます。

  – レベル 1 がアクティブのときに保護オプションバイトをレベル 0 にプログラミングすると（RDP のレベルの回帰）、フラッシュメモリやバックアップ SRAM が全体消去されます。結果的に、ユーザフラッシュメモリ領域やバックアップ SRAM は、RDP がレベル 0 に設定される前にクリアされます。全体消去によって両方のバンクとバックアップ SRAM の内容が削除され、DMEP1/2 および DMES1/2 ビットが「0」に設定されている場合、フラッシュ PCROP とセキュア領域が保持されます。

- レベル 2：デバッグ無効、RDP 有効

  RDP オプションバイトに 0xCC を書き込むと、読み出し保護レベル 2 がアクティブになります。読み出し保護レベル 2 を設定すると次のようになります。

  – レベル 1 によるすべての保護がアクティブになります。

  – RAM からのブートは許可されなくなります。

  – セキュリティを有効にすると、セキュアシステムフラッシュメモリでのブートが強制的に設定されます（*セクション 4：セキュリティメモリ管理*を参照）。

  – セキュリティを有効にしないと、システムフラッシュメモリからのブートは許可されなくなります。セキュアシステムフラッシュでのブートが強制的に設定されます（*セクション 4：セキュリティメモリ管理*を参照）。

  – すべてのデバッグ機能が無効になります。

  – SWAPビットを除き、ユーザオプションバイトを変更できなくなります（*セクション 3.3.13：フラッシュバンクスワッピング*を参照）。

  – フラッシュメモリからブートする場合は、ユーザコードからのフラッシュメモリおよびバックアップ SRAM への読み出し、消去、プログラムアクセスが許可されます。メモリ読み出し保護レベル 2 は、不可逆操作です。レベル 2 がアクティブになると、保護レベルをレベル 0 やレベル 1 に下げることはできません。

注： ***レベル 2 がアクティブになると、JTAG ポートは恒久的に無効になります（JTAG ヒューズとして作用）。結果として、STMicroelectronics は、レベル 2 保護が設定されている不良部品には分析を実施できません。***

表 11. RDP 値と読み出し保護レベルの比較

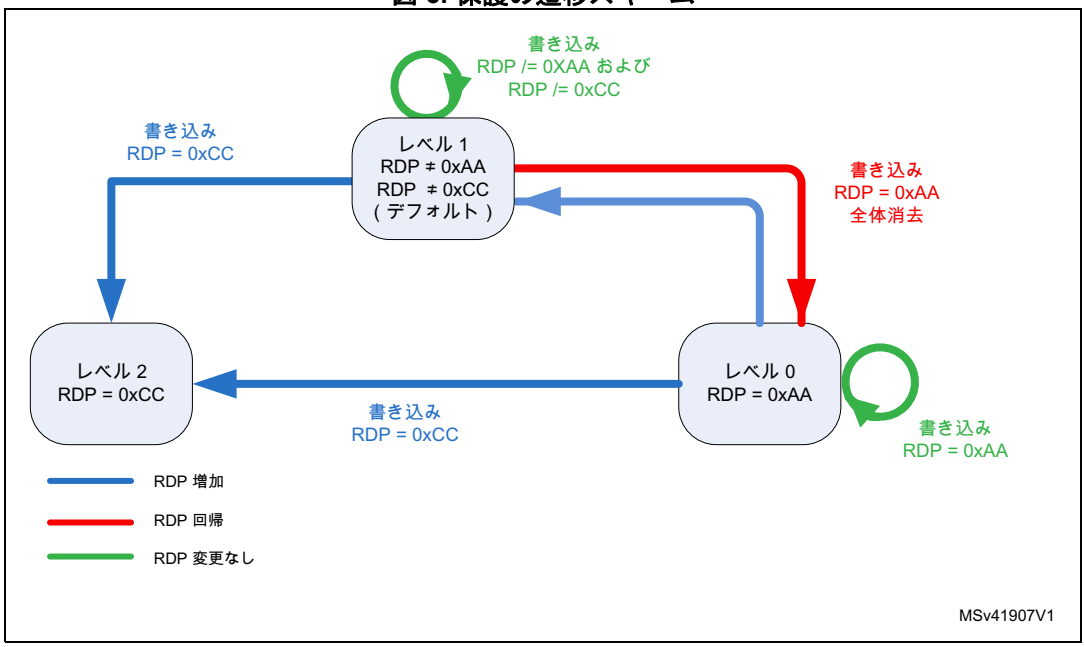
RDP オプションバイト値	読み出し保護レベル
0xAA	レベル 0
0xCC	レベル 2
その他の値	レベル 1

表 12. 許可されたアクセスと読み出し保護レベルの比較

領域	保護レベル（RDP）	ブート = ユーザフラッシュメモリ (1)	ブート /= ユーザフラッシュメモリまたは接続されているデバッグ
ユーザフラッシュメモリ	1	R/W/E	アクセスなし
	2	R/W/E	-(2)
システムフラッシュメモリ	1	R	R
	2	R	- (2)
オプションバイト	1	R/W/E	R/W/E
	2	R	- (2)
バックアップ SRAM	1	読み出し／書き込み	アクセスなし
	2	読み出し／書き込み	- (2)

1. W：書き込み、R：読み出し、E：消去
2. RDP = レベル 2 の場合、使用可能なデバッグ機能はなく、ユーザフラッシュ以外からブートすることはできず、また SWAPビットフィールドを除いてオプションバイトを変更することはできません。

図 6. 保護の遷移スキーム



PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）

フラッシュインタフェースでは、各フラッシュバンクの「実行のみ」領域を定義できます。この領域では、命令フェッチトランザクションのみ許可されます。データアクセス（データまたはリテラルプール）は許可されません。ネイティブコードは「実行のみ」オプションに従ってコンパイルする必要があります。実行のみ領域は、ソフトウェアの知的財産を保護するために役立ちます。

結果の差が正で null にならないよう（終了アドレスが開始アドレスより高くなるようにする）、PROT_AREA_END1/2 および PROT_AREA_START1/2 オプションバイトをセットすることで、2 つの異なる PCROP 領域を定義できます（フラッシュバンクごとに 1 つ）。

PCROP 領域の設定可能な最小サイズは 16 フラッシュワード、つまり 512 ビットです。実行のみ領域の設定可能な最大サイズはフラッシュバンク領域全体です。これには、PROT_AREA_END1/2 および PROT_AREA_START1/2 フィールドを同じ値に設定します。

PCROP 領域は、8 フラッシュワード単位で定義できます。つまり、1 ずつインクリメントされる終了および開始アドレスの差に8 を乗算した値は、PROT_AREA_END1/2 が PROT_AREA_START1/2 よりも厳密に高いことを前提とした場合の、PCROP で設定されたフラッシュワード数を表します。

任意の 2 つの PCROP 領域を無効にするには、PROT_AREA_END1/2 アドレスフィールドを PROT_AREA_START1/2 よりも低い値に設定する必要があります。つまり、結果の差は負になります（終了アドレスが開始アドレスより低くなる）。

たとえば、最初のバンクのバンク 1 ベースアドレスから始まる 4 KB に対するアドレス 0x08000000（始点を含む）からアドレス 0x08000FFF （終点を含む）の PCROP 領域を設定するには、PROT_AREA_START1 および PROT_AREA_END1 レジスタをそれぞれ次のようにプログラミングする必要があります。

- PROT_AREA_START1 = 0x000
- PROT_AREA_END1 = 0x00F

保護領域のサイズは、以下と等しくなります。

[(PROT_AREA_END - PROT_AREA_START) + 1] x 256 = 16 x 256 バイト = 4 KB

PCROP 領域の特性

- PCROP 領域から実行された読み出し（フェッチではない）トランザクションにより、FLASH_SR1/2 レジスタの RDPERR フラグが立てられます。
- PCROP 領域に対して実行された書き込みトランザクションにより、FLASH_SR1/2 レジスタの WRPERR フラグが立てられます。
- PCROP 領域がすでに定義されている場合、開始アドレスを減らすか、終了アドレスフィールドを増やすことで領域のサイズをより増加することができます。2 つのシーケンスを使用して、PCROP 領域を無効にすることができます。
  - RDP のレベルの回帰（レベル 1 からレベル 0）を実行し、DMEP1/2 ビットを有効にし、PROT_AREA_START1/2 を FLASH_PRAR_PRG1/2 レジスタの PROT_AREA_END1/2 よりも高い値に設定します。

    PCROP 保護は RDP 保護とは異なります。ただし、この場合、PCROP を解除する前に、まず RDP をレベル 1 に設定して RDP のレベルの回帰を有効にする必要があります。
  - バンク消去操作を要求し、同時に DMEP1/2、DMES1/2、およびすべての WRPSn1/2 ビットを「1」にセットし、PROT_AREA_START1/2 を FLASH_PRAR_PRG1/2 の PROT_AREA_END1/2 よりも高く設定し、また SEC_AREA_START1/2 を FLASH_SCAR_PRG1/2 の SEC_AREA_END1/2 よりも高く設定して有効にします。

セキュア領域

フラッシュインタフェースでは、各フラッシュバンクの「安全」領域を設定できます。この領域に保存されたデータとプログラムには、セキュアモードを設定するまでアクセスできません。セキュア領域は、セキュアユーザコードをアプリケーションの非セキュアコードから切り離す際に役立ちます。たとえば、セキュア領域はセキュアファームウェアアップグレードコードやセキュアブートコードを保護するために使用できます。

結果の差が正で null にならないよう（終了アドレスが開始アドレスより高くなるようにする）、SEC_AREA_END1/2 および SEC_AREA_START1/2 オプションバイトを設定することで、2 つの異なるセキュア領域を定義できます（フラッシュバンクごとに 1 つ）。これらのオプションバイトは、セキュアモードでのみ変更できる点に注意してください。（*セクション 4：セキュリティメモリ管理* を参照）。

設定できるセキュア領域の最小サイズは 512 バイトです。セキュア領域の最大サイズはフラッシュバンク全体です。これには、SEC_AREA_END1/2 および SEC_AREA_START1/2 フィールドを同じ値に設定します。

セキュア領域は、8 フラッシュワード単位で定義できます。つまり、1 ずつインクリメントされる終了および開始アドレスの差に8を乗算した値は、SEC_AREA_END1/2 が SEC_AREA_START1/2 よりも厳密に高いことを前提とした場合の、安全に保護されたフラッシュワード数を表します。

任意の 2 つのセキュア領域を無効にするには、SEC_AREA_END1/2 アドレスフィールドを SEC_AREA_START1/2 よりも低い値に設定する必要があります。つまり、結果の差は負になります（終了アドレスが開始アドレスより低くなる）。

たとえば、2 番目のバンクベースアドレスから始まる 8 KB に対するアドレス 0x08100000 （始点を含む）からアドレス 0x08101FFF（終点を含む）のセキュア領域を設定するには、SEC_AREA_START2 および SEC_AREA_END2 レジスタをそれぞれ次のようにプログラミングする必要があります。

- SEC_AREA_START2 = 0x000
- SEC_AREA_END2 = 0x01F

セキュア領域のサイズは、以下と等しくなります。

[(SEC_AREA_END2 - SEC_AREA_START2) + 1] x 256 = 32 x 256 バイト = 8 KB

セキュア領域の特性

- セキュア領域のサイズはセキュリティが有効な場合、および特定の条件下で変更できます。（*セクション 4：セキュリティメモリ管理*を参照）。セキュア領域を削除するには 2 つの方法があります。
  - RDP のレベルの回帰（レベル 1 からレベル 0）を実行し、DMES1/2 ビットを有効にし、SEC_AREA_START1/2 アドレスを FLASH_SCAR_PRG1/2 レジスタの SEC_AREA_END1/2 よりも高い値に設定します。

    RDP 保護は安全保護とは異なります。ただし、この場合、RDP を解除する前に、まず RDP をレベル 1 に設定して RDP のレベルの回帰を有効にする必要があります。
  - バンク消去操作を要求し、同時に DMEP1/2、DMES1/2、およびすべての WRPSn1/2 ビットを「1」にセットします。次に、SEC_AREA_START1/2 アドレスを FLASH_SCAR_PRG1/2 の SEC_AREA_END1/2 よりも高い値に設定し、また PROT_AREA_START1/2 を FLASH_PRAR_PRG1/2 の PROT_AREA_END1/2 よりも高い値に設定します。

デバイスのセキュリティ面の使用方法と、ユーザセキュアモードの有効化／無効化方法の詳細については、*セクション 4：セキュリティメモリ管理*を参照してください。

セクタ書き込み保護

セクタ書き込み保護の目的は、揮発性コードやデータの不要な変更からフラッシュメモリを保護することです。フラッシュセクタは、FLASH_WPSN_PRG1/2R レジスタの対応する WRPSn1/2 ビットをクリア／セットすることで、個別に書き込み保護または保護解除することができます。

書き込み保護されたセクタを消去またはプログラミングすることはできません。結果として、バンクセクタが 1 つ書き込み保護されている場合は、バンク消去がレベルの回帰の中で処理されない限り、バンク消去は実施できません。この場合、保護されていないセクタのみが消去されるため、バンク消去はセクタ消去操作のシーケンスに変換されます。

セクタ書き込み保護ビット（ユーザオプション）は、RDP のレベルがレベル 0 またはレベル 1 に設定されている場合、制限なく変更できます。レベル 2 にセットすると、それ以上オプションバイトの書き込み保護ビットフィールドを変更できなくなります。

注： ***デフォルトで、PCROP 領域は書き込み保護されています。結果的に、PCROP 領域に部分的または全体的に属しているセクタは、消去またはプログラミングすることはできません。***

3.3.13 フラッシュバンクスワッピング

フラッシュインタフェースでは、バンク 1 およびバンク 2 のメモリマッピングをスワッピングできます。この機能は、ファームウェアをアップグレードしてシステムリセットを行い、その後新しいファームウェアでデバイスを再起動する際に使用できます。バンクスワッピングは、FLASH_OPTCR レジスタの SWAP_BANK ビットによって制御されます。*表 13* に、SWAP_BANK オプションビットの設定に応じた両方の AXI スレーブフラッシュインタフェースからアクセス可能なメモリマップを示します。

表 13. AXI インタフェースのメモリマッピング SWAP_BANK = ‘0’

領域	AXI 1	AXI 2
ユーザセクタバンク 1	あり(1)	なし
ユーザセクタバンク 2	なし	あり (1)
システムフラッシュバンク 1	あり(2)	なし
システムフラッシュバンク 2	なし	あり (2)

1. ユーザバンク 1 と 2 はそれぞれ 0x0800 0000 から 0x080F FFFF と 0x0810 0000 から 0x081F FFFF にマッピングされます。
2. システムフラッシュバンク 1 と 2 はそれぞれ 0x1FF0 0000から0x1FF1 FFFFと0x1FF4 0000から0x1FF5 FFFF,にマッピングされます。

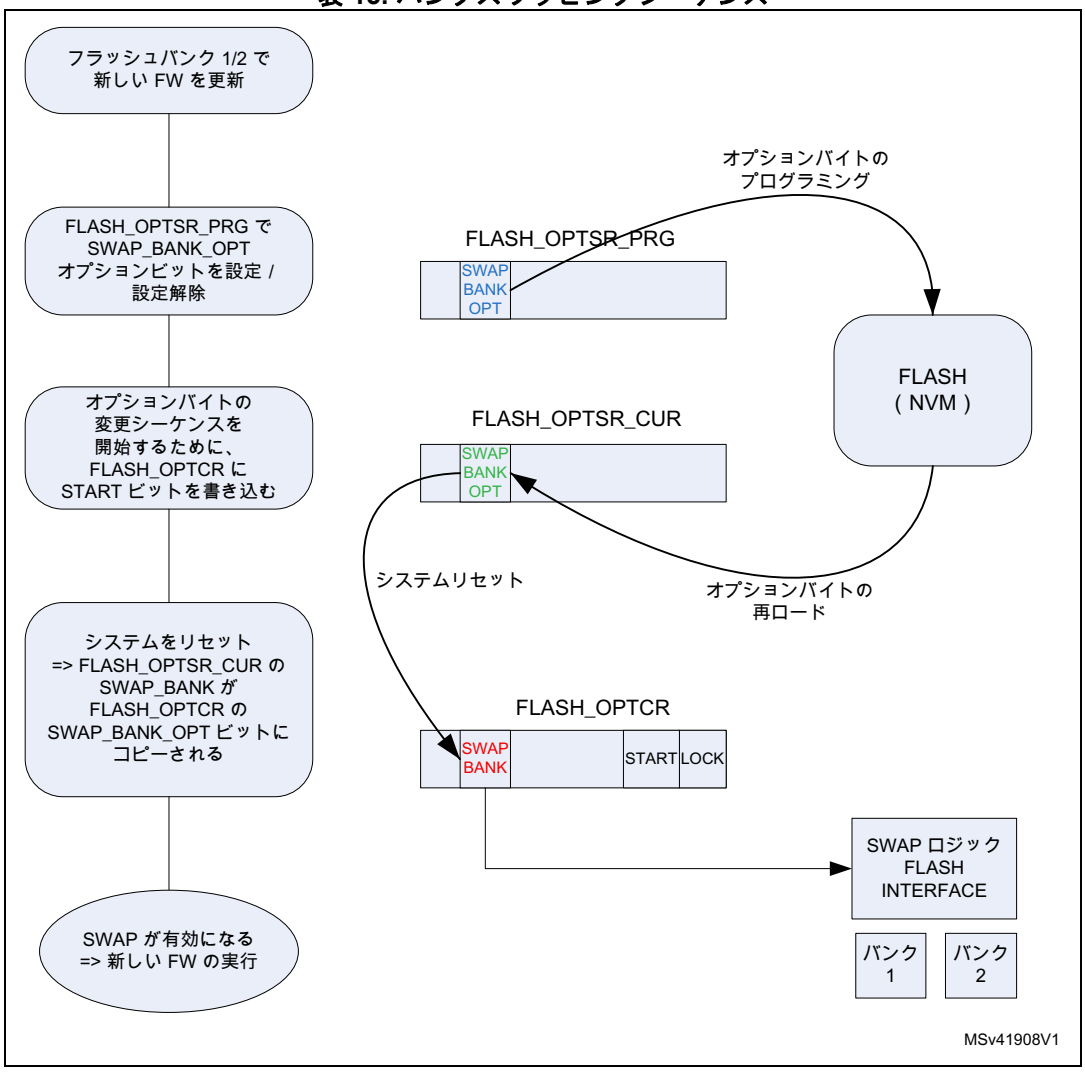
表 14. AXI インタフェースのメモリマッピング SWAP_BANK = ‘1’

領域	AXI 1	AXI 2
ユーザセクタバンク 1	なし	あり(1)
ユーザセクタバンク 2	あり (1)	なし
システムフラッシュバンク 1	あり(2)	なし
システムフラッシュバンク 2	なし	あり (2)

1. ユーザバンク 1 と 2 はそれぞれ 0x0810 0000 から 0x081F FFFF と 0x0800 0000 から 0x080F FFFF にマッピングされます。
2. システムフラッシュバンク 1 と 2 はそれぞれ 0x1FF0 0000から0x1FF1 FFFFと0x1FF4 0000から0x1FF5 FFFF,にマッピングされます。

FLASH_OPTSR_PRG レジスタの SWAP_BANK_OPT ビットを変更する場合、新しい値はオプションバイトが FLASH_OPTSR_CUR レジスタのオプションバイトセクタから再ロードされ、システムリセット後にその値が FLASH_OPTCR レジスタの SWAP_BANK ビットにコピーされた後で反映されます。

表 15. バンクスワッピングシーケンス



ファームウェアアップデート後にバンク 1 およびバンク 2 のメモリマッピングをスワップする際の推奨シーケンスは次のとおりです。

1. バンク 2 またはバンク 1 を新しいファームウェアで更新します。
2. FLASH_OPTSR_PRG レジスタで適宜 SWAP_BANK_OPT ビットを変更します。
3. FLASH_OPTCR レジスタの OPTSTART ビットをセットすることによりオプションバイトの変更シーケンスを開始します。プログラミングを行う前にレジスタがアンロックされていることを確認してください。
4. システムリセットを行います。オプションバイトのローディングシーケンス後に、バンクスワップが有効になり、新しいファームウェアが実行されます。

参考資料

注： ***FLASH_OPTSR_PRG の SWAP_BANK_OPT オプションビットは、RDP のレベルにかかわらず変更できます。レベル 2 でも更新できるため、どのレベルの読み出し保護でも高度なファームウェアのアップグレードを行うことができます。***

3.4 FLASH 割り込み

*セクション 3.3.9：フラッシュインタフェースエラーフラグ*で説明しているとおり、FLASH_CR1/2 レジスタにて対応する割り込みイネーブルビットがセットされている場合、エラーフラグごとに割り込みが生成されます。

FLASH_SR1/2 レジスタの操作終了（EOP1/2）ビットは、消去、プログラム、またはオプションの変更操作が正常に完了したときにセットされます。FLASH_CR1/2 レジスタの操作終了割り込みイネーブルビット（EOPIE1/2）をセットすると、消去、プログラム、またはオプションの変更操作が完了したときに割り込みを生成できます。

EOP1/2 フラグは、FLASH_FCCR1/2 レジスタの対応するビットに書き込むことでクリアすることができます。

3.5 フラッシュレジスタ

3.5.1 フラッシュアクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x000 または 0x100

リセット値：0x0000 0037

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRHIGH FREQ		Res.	LATENCY		
										rw	rw		rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **WRHIGHFREQ**：フラッシュ信号遅延

これらのビットは、プログラム操作中にフラッシュ信号間の遅延を制御するために使用されます。ユーザは、フラッシュインタフェースの周波数に応じて正しい値を書き込む必要があります（*表 7* を参照）。設定が正しいかどうかを検証するための確認は行われません。

00：≤85 MHz

01：≤185 MHz

10：≤285 MHz

11：≤385 MHz

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LATENCY**：読み出し遅延

これらのビットは、両方のフラッシュメモリバンク上で読み出し操作中に使用する、ウェイトステートの数を指定します。ユーザは、*表 7* で説明しているように、フラッシュメモリインタフェースの周波数およびパワー動作モードに応じて正しい値を書き込む必要があります。設定が正しいかどうかを検証するための確認は行われません。

000：フラッシュメモリからワードを読み出すために、0 ウェイトステートを使用します。

001：フラッシュメモリからワードを読み出すために、1 ウェイトステートを使用します。

...

111：フラッシュメモリからワードを読み出すために、7 ウェイトステートを使用します。

3.5.2 バンク 1 のフラッシュキーレジスタ（FLASH_KEYR1）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR1															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR1															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **KEYR1**：バンク 1 アクセス設定アンロックキー

FLASH_KEYR1 は書き込み専用レジスタです。FLASH_CR1 レジスタをアンロックし、プログラミングや消去を許可するには、次の値を連続してプログラムする必要があります。

a) 1 番目のキー = 0x4567 0123

b) 2 番目のキー = 0xCDEF 89AB

詳細については、*セクション ：プログラミングパラメータの設定*を参照してください。

3.5.3 フラッシュオプションキーレジスタ（FLASH_OPTKEYR）

アドレスオフセット：0x008 または 0x108

リセット値：0x0000 0000

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

ビット 31:0 **OPTKEYR**：キーオプションバイトのアンロック

FLASH_OPTKEYR は書き込み専用レジスタです。FLASH_OPTCR レジスタをアンロックし、このレジスタおよびすべての _PRG レジスタのプログラミングや消去を許可するには、次の値を連続してプログラムする必要があります。

a) 1 番目のキー = 0x0819 2A3B

b) 2 番目のキー = 0x4C5D 6E7F

詳細については、*セクション：プログラミングパラメータの設定* を参照してください。

3.5.4 バンク 1 のフラッシュ制御レジスタ（FLASH_CR1）

アドレスオフセット：0x00C

リセット値：0x0000 0031

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CRCENDIE1	DBECCERRIE1	SNECCERRIE1	RDSERRIE1	RDPERRIE1	OPERRIE1	INCERRIE1	Res.	STRBERRIE1	PGSERRIE1	WRPERRIE1	EOPIE1
				rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_EN	Res.	Res.	Res.	Res.	SNB1			START1	FW1	PSIZE1		BER1	SER1	PG1	LOCK1
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rs

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CRCENDIE**1：バンク 1 CRC 計算終了割り込みイネーブルビット

CRCENDIE1 ビットを「1」にセットすると、バンク 1 で CRC 計算が完了したときに割り込みが生成されます。CRCENDIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で CRC 計算が完了したときに、割り込みを生成しません。

1：バンク 1 で CRC 計算が完了したときに、割り込みを生成します。

ビット 26 **DBECCERRIE**1：バンク 1 ECC ダブル検出エラー割り込みイネーブルビット

DBECCERRIE1 ビットを「1」にセットすると、バンク 1 からの読み出し操作中に ECC ダブル検出エラーが発生したときに割り込みが生成されます。DBECCERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で ECC ダブル検出エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で ECC ダブル検出エラーが発生したときに、割り込みを生成します。

ビット 25 **SNECCERRIE**1：バンク 1 ECC シングル訂正エラー割り込みイネーブルビット

SNECCERRIE1 ビットを「1」にセットすると、バンク 1 からの読み出し操作中に ECC シングル訂正エラーが発生したときに割り込みが生成されます。SNECCERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で ECC シングル訂正エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で ECC シングル訂正エラーが発生したときに、割り込みを生成します。

ビット 24 **RDSERRIE1**：バンク 1 セキュアエラー割り込みイネーブルビット

RDSERRIE1 ビットを「1」にセットすると、バンク 1 からの読み出し操作中セキュアエラー（適切な権限なく安全に保護されたアドレスへのアクセス）が発生したときに割り込みが生成されます。RDSERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 でセキュアエラーが発生したときに、割り込みを生成しません。

1：バンク 1 でセキュアエラーが発生したときに、割り込みを生成します。

ビット 23 **RDPERRIE**1：バンク 1 読み出し保護エラー割り込みイネーブルビット

RDPERRIE1 ビットを「1」にセットすると、バンク 1 からの読み出し操作中に読み出し保護エラー（PCROP で保護されたアドレスへのアクセス）が発生したときに割り込みが生成されます。RDPERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で読み出し保護エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で読み出し保護エラーが発生したときに、割り込みを生成します。

ビット 22 **OPERRIE**1：バンク 1 書き込み／消去エラー割り込みイネーブルビット

OPERRIE1 ビットを「1」にセットすると、バンク 1 への書き込み／消去操作中にエラーを検出したときに割り込みが生成されます。OPERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で書き込み／消去エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で書き込み／消去エラーが発生したときに、割り込みを生成します。

ビット 21 **INCERRIE**1：バンク 1 不整合エラー割り込みイネーブルビット

INCERRIE1 ビットを「1」にセットすると、バンク 1 への書き込み操作中に不整合エラーが発生したときに割り込みが生成されます。INCERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で不整合エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で不整合エラーが発生したときに、割り込みを生成します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERRIE**1：バンク 1 ストローブエラー割り込みイネーブルビット

STRBERRIE1 ビットを「1」にセットすると、バンク 1 への書き込み操作中にストローブエラー（マスタにより書き込みバッファに同じバイトを数回プログラム）が発生したときに割り込みが生成されます。STRBERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 でストローブエラーが発生したときに、割り込みを生成しません。

1：バンク 1 でストローブエラーが発生したときに、割り込みを生成します。

ビット 18 **PGSERRIE**1：バンク 1 プログラミングシーケンスエラー割り込みイネーブルビット

PGSERRIE1 ビットを「1」にセットすると、バンク 1 へのプログラム操作中にシーケンスエラーが発生したときに割り込みが生成されます。PGSERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 でシーケンスエラーが発生したときに、割り込みを生成しません。

1：バンク 1 でシーケンスエラーが発生したときに、割り込みを生成します。

ビット 17 **WRPERRIE**1：バンク 1 書き込み保護エラー割り込みイネーブルビット

WRPERRIE1 ビットを「1」にセットすると、バンク 1 へのプログラム操作中に保護エラーが発生したときに割り込みが生成されます。WRPERRIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 で保護エラーが発生したときに、割り込みを生成しません。

1：バンク 1 で保護エラーが発生したときに、割り込みを生成します。

ビット 16 **EOPIE**1：バンク 1 プログラム終了割り込み制御ビット

EOPIE1 ビットを「1」にセットすると、バンク 1 へのプログラム操作の終了時に割り込み生成が有効になります。EOPIE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 1 へのプログラム操作の終了時に、割り込みを生成しません。

1：バンク 1 へのプログラム操作の終了時に、割り込みを有効にします。

ビット 15 **CRC_EN**：バンク 1 CRC 制御ビット

CRC_EN ビットを「1」にセットすると、バンク 1 で CRC 計算が有効になります。CRC_EN では CRC 計算を開始しませんが、FLASH_CRCCR1 レジスタを通じて CRC 設定を有効にします。

バンク 1 で CRC 計算を実行した場合は、CRC_EN ビットを「0」に設定することでのみ無効化することができます。CRC_EN をリセットすると、FLASH_CRCDATAR レジスタの内容もリセットされます。

CRC_EN は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **SNB1**：バンク 1 セクタ消去選択番号

これらのビットは、セクタ消去操作の対象セクタを選択するために使用します。SNB1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

000：バンク 1 のセクタ 0

001：バンク 1 のセクタ 1

...

111：バンク 1 のセクタ 7

ビット 7 **START1**：バンク 1 バンク消去またはセクタ消去開始制御ビット

START1 ビットは、バンク消去またはセクタ消去操作を開始するために使用します。START1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

該当する操作を確認すると、フラッシュメモリインタフェースは START1 をリセットします。該当する操作を確認するまで、ユーザアプリケーションはどのフラッシュレジスタにもアクセスできません。

ビット 6 **FW1**：バンク 1 書き込み強制制御ビット

FW1 は書き込みバッファがフルでなくても書き込み操作を強制します。FW1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

該当する操作を確認すると、フラッシュメモリインタフェースは FW1 をリセットします。該当する操作を確認するまで、ユーザアプリケーションはどのフラッシュレジスタにもアクセスできません。

書き込みの強制は、書き込みバッファが空でない場合にのみ有効です（特に書き込み操作が連続して行われている場合、FW1 ではいくつかの書き込み操作を開始しません）。

ビット 5:4 **PSIZE1 :** バンク 1 プログラムサイズ

PSIZE1 は、バンク 1 への書き込みおよび消去操作中に、フラッシュメモリが使用する並列度を選択します（詳細については、*セクション : プログラミングパラメータの設定* を参照）。PSIZE1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

00 : バイト並列処理でプログラミングを実行します。

01 : ハーフワード並列処理でプログラミングを実行します。

10 : ワード並列処理でプログラミングを実行します。

11 : ダブルワード並列処理でプログラミングを実行します。

ビット 3 **BER**1 : バンク 1 消去リクエスト

BER1 ビットを「1」にセットすると、バンク 1 でバンク消去操作が要求されます。BER1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

BER1 の優先順位は SER1 よりも高いため、両方がセットされている場合、フラッシュメモリインタフェースではバンク消去を実行します（詳細については、*セクション : フラッシュバンク消去* を参照）。

0 : バンク消去をバンク 1 で要求しません。

1 : バンク消去をバンク 1 で要求します。

ビット 2 **SER**1 : バンク 1 セクタ消去リクエスト

SER1 ビットを「1」にセットすると、バンク 1 でセクタ消去操作が要求されます。SER1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。

BER1 の優先順位は SER1 よりも高いため、両方がセットされている場合、フラッシュメモリインタフェースではバンク消去を実行します （詳細については、*セクション : フラッシュセクタの消去* を参照）。

0 : セクタ消去をバンク 1 で要求しません。

1 : セクタ消去をバンク 1 で要求します。

ビット 1 **PG**1 : バンク 1 プログラムイネーブルビット

PG1 ビットを「1」にセットすると、バンク 1 への書き込み操作が有効になります。これにより、セクタ消去やバンク消去の進行中でもプログラム操作を準備できるようになります。

PG1 は LOCK1 を「0」に設定している場合にのみプログラムすることができます。PG1 がリセットされると、バンク 1 への書き込み操作について内部バッファが無効化されます。また、バッファに保存された、プログラミングされていないデータはすべて失われます。

ビット 0 **LOCK1** : バンク 1 設定ロックビット

このビットでは、FLASH_CR1 レジスタをロックします。

FLASH_CR1 レジスタがアンロックされている場合、LOCK1 ビットは自動的にリセットされます（*セクション : プログラミングパラメータの設定* を参照）。誤ったシーケンスが実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

LOCK1 は「1」にプログラムすることでセットすることができます。「1」にセットすると、アンロックするには新しいアンロックシーケンスが必要になります。LOCK1 を「0」から「1」に変更するときに、FLASH_CR1 レジスタのその他のビットは変更されません。

0 : FLASH_CR1 レジスタをアンロックします。

1 : FLASH_CR1 レジスタをロックします。

3.5.5 バンク 1 のフラッシュステータスレジスタ（FLASH_SR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CRCEND1	DBECCERR1	SNECCERR1	RDSERR1	RDPERR1	OPERR1	INCERR1	Res.	STRBERR1	PGSERR1	WRPERR1	EOP1
				r	r	r	r	r	r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY1	QW1	WBNE1	BSY1
												r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CRCEND1**：バンク 1 CRC 完了フラグ

バンク 1 で CRC 計算が完了すると、CRCEND1 ビットがセットされます。CRCENDIE1 が「1」にセットされている場合、割り込みが生成されます。CRC 計算を再開する前に CRCEND1 をリセットする必要はありません。FLASH_CCR1 レジスタの CLR_CRCEND1 ビットに「1」を書き込むと CRCEND1 はクリアされます。

0：バンク 1 の CRC 計算が未完了。

1：バンク 1 の CRC 計算が完了。

ビット 26 **DBECCERR1**：バンク 1 ECC ダブル検出エラーフラグ

DBECCERR1 フラグは、バンク 1 からの読み出し操作中に ECC ダブル検出エラーが発生した場合に立てられます。DBECCERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_DBECCERR1 ビットに「1」を書き込むと DBECCERR1 はクリアされます。

0：ECC ダブル検出エラーがバンク 1 で発生していません。

1：ECC ダブル検出エラーがバンク 1 で発生しました。

ビット 25 **SNECCERR11**：バンク 1 シングル訂正エラーフラグ

SNECCERR1 フラグは、バンク 1 からの読み出し操作中に ECC シングル訂正エラーが発生した場合に立てられます。SNECCERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_SNECCERR1 ビットに「1」を書き込むと SNECCERR1 はクリアされます。

0：ECC シングル訂正エラーがバンク 1 で発生していません。

1：0：ECC シングル訂正エラーがバンク 1 で発生しました。

ビット 24 **RDSERR1**：バンク 1 セキュアエラーフラグ

RDSERR1 フラグは、バンク 1 でセキュアエラー（適切な権限なく安全に保護されたワードにアクセス）が発生した場合に立てられます。RDSERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_RDSERR1 ビットに「1」を書き込むと RDSERR1 はクリアされます。

0：バンク 1 でセキュアエラーが発生していません。

1：バンク 1 でセキュアエラーが発生しました。

ビット 23 **RDPERR1**：バンク 1 読み出し保護エラーフラグ

RDPERR1 フラグは、バンク 1 で読み出し保護エラー（PCROP で保護されたワードへのアクセス）が発生した場合に立てられます。RDPERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_RDPERR1 ビットに「1」を書き込むと RDPERR1 はクリアされます。

0：バンク 1 で読み出し保護エラーが発生していません。

1：バンク 1 で読み出し保護エラーが発生しました。

ビット 22 **OPERR1**：バンク 1 書き込み／消去エラーフラグ

OPERR1 フラグは、バンク 1 での書き込み／消去操作中にエラーが発生した場合に立てられます。OPERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_OPERR1 ビットに「1」を書き込むと OPERR1 はクリアされます。

0：バンク 1 で書き込み／消去エラーが発生していません。

1：バンク 1 で書き込み／消去エラーが発生しました。

ビット 21 **INCERR1**：バンク 1 不整合エラーフラグ

INCERR1 フラグは、バンク 1 で不整合エラーが発生した場合に立てられます。INCERRIE1 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_INCERR1 ビットに「1」を書き込むと INCERR1 はクリアされます。

（*セクション 3.3.9：フラッシュインタフェースエラーフラグ*を参照してください。）

0：バンク 1 で不整合エラーが発生していません。

1：バンク 1 で不整合エラーが発生しました。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERR1**：バンク 1 ストローブエラーフラグ

STRBERR1 フラグは、バンク 1 でストローブエラー（マスタにより書き込みバッファへ同じバイトを数回書き込み）が発生した場合に立てられます。STRBERRIE1 ビットが「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_STRBERR1 ビットに「1」を書き込むと STRBERR1 はクリアされます。

0：バンク 1 でストローブエラーが発生していません。

1：バンク 1 でストローブエラーが発生しました。

ビット 18 **PGSERR1**：バンク 1 プログラミングシーケンスエラーフラグ

PGSERR1 フラグは、バンク 1 でシーケンスエラーが発生した場合に立てられます。PGSERRIE1 ビットが「1」にセットされている場合、割り込みが生成されます。FLASH_CCR1 レジスタの CLR_PGSERR1 ビットに「1」を書き込むと PGSERR1 はクリアされます。

0：バンク 1 でシーケンスエラーが発生していません。

1：バンク 1 でシーケンスエラーが発生しました。

ビット 17 **WRPERR1**：バンク 1 書き込み保護エラーフラグ

WRPERR1 フラグは、バンク 1 へのプログラム操作中に保護エラーが発生した場合に立てられます。EOPIE1 が「1」にセットされている場合は、割り込みも生成されます。FLASH_CCR1 レジスタの CLR_EOP1 ビットに「1」を書き込むと WRPERR1 はクリアされます。

0：バンク 1 で保護エラーが発生していません。

1：バンク 1 で保護エラーが発生しました。

ビット 16 **EOP1**：バンク 1 プログラム終了フラグ

EOP1 フラグは、バンク 1 へのプログラム操作が完了した場合にセットされます。EOPIE1 が「1」にセットされている場合、割り込みが生成されます。新しい操作を開始する前に EOP1 をリセットする必要はありません。EOP1 ビットは、FLASH_CCR1 レジスタの CLR_EOP1 ビットに「1」を書き込むことによりクリアされます。

0：バンク 1 のプログラム操作が完了していません。

1：バンク 1 のプログラム操作が完了しました。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRC_BUSY1**：バンク 1 CRC ビジーフラグ

CRC_BUSY1 フラグは、バンク 1 での CRC 計算の進行中にセットされます。このビットを強制的に 0 にすることはできません。ユーザは、CRC 計算が完了するまで待つか、またはバンク 1 の CRC 計算を無効にする必要があります。

0：バンク 1 に進行中の CRC 計算がありません。

1：バンク 1 に進行中の CRC 計算があります。

ビット 2 **QW1**：バンク 1 ウェイトキューフラグ

QW1 フラグは、バンク 1 へのプログラム操作がウェイトキューにある場合にセットされます。キューに含まれるプログラム操作の種類を知ることはできません。すべてのプログラム操作を実行してウェイトキューから削除されると、このフラグはハードウェアによってリセットされます。このビットを強制的に 0 にすることはできません。他に要求された操作がない場合、決定論的な間隔を空けてリセットされます。

0：バンク 1 のウェイトキューにプログラム操作がありません。

1：バンク 1 のウェイトキューに少なくとも 1 つのプログラム操作があります。

ビット 1 **WBNE1**：バンク 1 書き込みバッファノットエンプティフラグ

WBNE1 フラグは、バンク 1 の書き込みバッファが空ではなく、フラッシュメモリインタフェースが新しいデータの書き込み完了を待っている場合にのみセットされます。

WBNE1 は書き込みバッファが空になるたびにハードウェアによってリセットされます。これは、次のいずれかのイベントが発生したときに生じます。

- 書き込みバッファがフルである場合
- ユーザが書き込み操作を強制した場合
- データの損失を伴うエラー
- 書き込み操作が無効化された場合

このビットを強制的に 0 にすることはできません。リセットするには、上記のいずれかのアクションを実行して、書き込みバッファをクリアします。

0：バンク 1の書き込みバッファが空です。

1：バンク 1 の書き込みバッファがデータの書き込み完了を待っています。

ビット 0 **BSY1**：バンク 1 プログラム進行中フラグ

BSY1 フラグは、バンク 1 へのプログラム操作が進行中である場合にセットされます。進行中のプログラム操作の種類を知ることはできません。

BSY1 を強制的に 0 にすることはできません。操作完了時に他のプログラム操作が開始しない場合に、ハードウェアによってリセットされます。

0：バンク 1 に実行中のプログラム操作がありません。

1：バンク 1 に実行中のプログラム操作があります。

3.5.6 バンク 1 のフラッシュクリア制御レジスタ（FLASH_CCR1）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CLR_CRCEND1	CLR_DBECCERR1	CLR_SNECCERR1	CLR_RDSERR1	CLR_RDPERR1	CLR_OPERR1	CLR_INCERR1	Res.	CLR_STRBERR1	CLR_PGSERR1	CLR_WRPERR1	CLR_EOP1
				w	w	w	w	w	w	w		w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CLR_CRCEND1**：バンク 1 CRCEND1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの CRCEND1 フラグが「0」にリセットされます。

ビット 26 **CLR_DBECCERR**1：バンク 1 DBECCERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの DBECCERR1 フラグが「0」にリセットされます。FLASH_SR1 レジスタの SNECCERR1 フラグが「0」に設定されている場合、FLASH_ECC_FA1R レジスタも「0」にリセットされます。

ビット 25 **CLR_SNECCERR1**：バンク 1 SNECCERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの SNECCERR1 フラグが「0」にリセットされます。FLASH_SR1 レジスタの DBECCERR1 フラグが「0」に設定されている場合、FLASH_ECC_FA1R レジスタも「0」にリセットされます。

ビット 24 **CLR_RDSERR1**：バンク 1 RDSERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの RDSERR1 フラグが「0」にリセットされます。

ビット 23 **CLR_RDPERR1**：バンク 1 RDPERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの RDPERR1 フラグが「0」にリセットされます。

ビット 22 **CLR_OPERR1**：バンク 1 OPERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの OPERR1 フラグが「0」にリセットされます。

ビット 21 **CLR_INCERR1**：バンク 1 INCERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの INCERR1 フラグが「0」にリセットされます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CLR_STRBERR1**：バンク 1 STRBERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの STRBERR1 フラグが「0」にリセットされます。

ビット 18 **CLR_PGSERR1**：バンク 1 PGSERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの PGSERR1 フラグが「0」にリセットされます。

ビット 17 **CLR_WRPERR1**：バンク 1 WRPERR1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの WRPERR1 フラグが「0」にリセットされます。

ビット 16 **CLR_EOP1**：バンク 1 EOP1 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR1 レジスタの EOP1 フラグが「0」にリセットされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

3.5.7 フラッシュオプション制御レジスタ（FLASH_OPTCR）

アドレスオフセット：0x018 または 0x118

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MER	Res.	OPTSTART	OPTLOCK
												w		rw	rs

ビット 31 **SWAP_BANK**：バンクスワッピング設定ビット

SWAP_BANK はバンク 1 およびバンク 2 をスワップするかどうかを制御します。オプションバイトローディング後、このビットは FLASH_OPTSR_CUR レジスタのSWAP_BANK_OPT ビットとともにロードされます。FLASH_OPTCR レジスタがアンロックされている場合（OPTLOCK = '0'）、マスタは SWAP_BANK を変更してユーザフラッシュメモリバンクをスワップ／スワップ解除することができます（*セクション 3.3.13：フラッシュバンクスワッピング*を参照）。

0：バンク 1 とバンク 2 をスワップしません。

1：バンク 1 とバンク 2 をスワップします。

ビット 30 **OPTCHANGEERRIE**：オプションバイト変更エラー割り込みイネーブルビット

OPTCHANGEERRIE ビットは、オプションバイトの変更中にエラーが発生したときに、割り込みを生成する必要があるかどうかを制御します。

0：オプションバイトの変更中にエラーが発生したときに、割り込みを生成しません。

1：オプションバイトの変更中にエラーが発生したときに、割り込みを生成します。

ビット 29:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MER**：フラッシュ全体消去イネーブルビット

MER ビットをプログラムするには、FLASH_CR1、FLASH_CR2、および FLASH_OPTCR の各レジスタを事前にアンロックしておく必要があります。

MER ビットを「1」にプログラムすると、BER1、BER2、START1、および START2 は自動的に「1」にセットされます。これにより、両方のバンクで同時に全体消去を実行できます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OPTSTART**：オプションバイト開始変更オプション設定ビット

OPTSTART は、オプションバイトの変更操作をトリガします。ユーザは、OPTLOCK ビットが「0」に設定されている場合にのみ OPTSTART をセットできます。オプションバイト変更操作を確認すると、フラッシュメモリインタフェースでは OPTSTART をリセットします。

該当する操作を確認するまで、ユーザアプリケーションはどのフラッシュレジスタにもアクセスできません。

このビットをセットする前に、ユーザは FLASH_XXX_PRG レジスタに必要な値を書き込む必要があります。FLASH_XXX_PRG レジスタは、フラッシュメモリでオプションバイトの変更操作が実行されるまでロックされます。

バンク 1 またはバンク 2 で CRC 計算が進行中の場合、オプションバイトの変更操作を開始することはできません。FLASH_SR1/2 レジスタの CRC_BUSY1/2 がセットされているときに OPTSTART のセットを試みても効果はありません。オプションバイトの変更は開始されず、エラーも生成されません。

ビット 0 **OPTLOCK**：FLASH_OPTCR ロックオプション設定ビット

OPTLOCK ビットは、FLASH_OPTCR レジスタをロックします。FLASH_OPTCR がアンロックされている場合、OPTLOCK は自動的にリセットされます（*セクション：プログラミングパラメータの設定* を参照）。誤ったシーケンスが実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

OPTLOCK は「1」にプログラムすることでセットすることができます。「1」にセットすると、アンロックするには新しいアンロックシーケンスが必要になります。OPTLOCK を「0」から「1」に変更するときに、FLASH_OPTCR レジスタのその他のビットは変更されません。

0：FLASH_OPTCR レジスタをアンロックします。

1：FLASH_OPTCR レジスタをロックします。

3.5.8 フラッシュオプションステータスレジスタ（現在値）（FLASH_OPTSR_CUR）

アドレスオフセット：0x01C または 0x11C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWAP_BANK_OPT	OPTCHANGEERR	IO_HSLV	PERSO_OK	RSS2	RSS1	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDP								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV		Res.	OPT_BUSY
r	r	r	r	r	r	r	r	r	r	r	r	r	r		r

ビット 31 **SWAP_BANK_OPT**：バンクスワッピングオプションステータスビット

SWAP_BANK_OPT には、バンク 1/2 スワッピングのデフォルト値を設定する、対応するオプションビットの値が反映されます。

0：ブートローディング後、ユーザセクタはスワップされません。

1：ブートローディング後、ユーザセクタをスワッピングします。

ビット 30 **OPTCHANGEERR**：オプションバイト変更エラーフラグ

OPTCHANGEERR フラグは、オプションバイトの変更操作中にエラーが発生したことを示します。OPTCHANGEERR が「1」にセットされた場合、オプションバイトの変更操作は正常に完了していません。FLASH_OPTCR レジスタの OPTCHANGEERRIE ビットが「1」にセットされている場合、このフラグが立てられると割り込みが生成されます。

FLASH_OPTCCR レジスタの CLR_OPTCHANGEERR に「1」を書き込むと OPTCHANGEERR はクリアされます。

0：オプションバイトの変更エラーは発生していません。

1：オプションバイトの変更操作中に 1 つ以上のエラーが発生しました。

ビット 29 **IO_HSLV**：低電圧ステータスビット（PRODUCT_BELOW_25V）での I/O ハイスピード

このビットは製品が 2.5 V 未満で動作することを示します。

0：全電圧範囲で動作する製品。低電圧での I/O スピードの最適化は無効です。

1：2.5 V 未満で動作する製品。低電圧での I/O スピードの最適化は有効です。

ビット 28 **PERSO_OK**：デバイスパーソナライズステータスビット

PERSO_OK はデバイスがパーソナライズされたことを示します。

ビット 27 **RSS2**：ユーザオプションビット 2

RSS2 は ST の開発コードに使用されます（RSS／ブートローダ）。

ビット 26 **RSS1**：ユーザオプションビット 1

RSS1 は ST の開発コードに使用されます（RSS／ブートローダ）。

ビット 25：22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SECURITY**：セキュリティイネーブルオプションステータスビット

0：セキュリティ機能は無効です。

1：セキュリティ機能は有効です。

ビット 20:19 **ST_RAM_SIZE**：DTCM RAM サイズオプションステータス

00：2 KB

01：4 KB

10：8 KB

11：16 KB

注： このビットフィールドは、セキュリティを有効（SECURITY = '1'）にした場合にのみ有効です。

ビット 18 **FZ_IWDG_SDBY**：IWDG STANDBY モード凍結オプションステータスビット

このビットは、IWDG_FZ_STOP オプションビットの凍結ステータスを反映しています。

0：独立型ウォッチドッグが STANDBY モードで凍結されています。

1：独立型ウォッチドッグが STANDBY モードで動作しています。

ビット 17 **FZ_IWDG_STOP**：IWDG STOP モード凍結オプションステータスビット

このビットは、IWDG_FZ_STANDBY オプションビットの凍結ステータスを反映しています。

0：独立型ウォッチドッグが STOP モードで凍結されています。

1：独立型ウォッチドッグが STOP モードで動作しています。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **RDP**：読み出し保護レベルオプションステータスバイト

読み出し保護レベルの詳細については、*セクション ：RDP（読み出し保護）*を参照してください。3 つの異なるレベルを使用できます。

0xAA：保護レベル 0

0xCC：保護レベル 2

その他の値：保護レベル 1

ビット 7 **nRST_STBY_D1**：D1 DStandby 移行リセットオプションステータスビット

0：D1 ドメインの DStandby モードに移行するときにリセットを生成します。

1：リセットは生成されません。

ビット 6 **nRST_STOP_D1**：D1 DStop 移行リセットオプションステータスビット

0：D1 ドメインの DStop モードに移行するときにリセットを生成します。

1：リセットは生成されません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IWDG1_HW**：IWDG1 制御オプションステータスビット

0：IWDG1 はソフトウェアによって制御されます。

1：IWDG1 ウォッチドッグはハードウェアによって制御されます。

ビット 3:2 **BOR_LEV**：ブラウンアウトレベルオプションステータスビット

これらのオプションビットは、システムリセットを生成するパワーレベルを定義するために使用されます。

00：リセットレベルを 2.1 V にセットします。

01：リセットレベルを 2.4 V にセットします。

10：リセットレベルを 2.7 V にセットします。

11：リセットレベルを 2.1 V にセットします（00 設定として）。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **OPT_BUSY**：オプションバイト変更進行中フラグ

OPT_BUSY は、オプションバイトの変更が進行中かどうかを示します。このビットを「1」にセットすると、フラッシュメモリインタフェースによってオプションの変更が実行され、フラッシュレジスタを変更できなくなります。

0：進行中のオプションバイトの変更はありません。

1：オプションバイトの変更が進行中で、オプションバイトの変更が完了するまで、フラッシュレジスタへのすべての書き込みアクセスがブロックされます。

3.5.9 フラッシュオプションステータスレジスタ（プログラムする値）（FLASH_OPTSR_PRG）

アドレスオフセット：0x020 または 0x120

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWAP_BANK_OPT	Res.	IO_HSLV	Res.	RSS2	RSS1	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.
rw		rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDP								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV		Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 **SWAP_BANK_OPT**：バンクスワッピングオプション設定ビット

SWAP_BANK_OPT オプションビットは、バンク 1/2 スワッピングのデフォルト値を設定するために使用します。

0：ブートローディング後、ユーザセクタはスワップされません。

1：ブートローディング後、ユーザセクタをスワップします。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **IO_HSLV**：低電圧（PRODUCT_BELOW_25V）での I/O ハイスピード

このビットは製品が 2.5 V 未満で動作することを示します。製品の電源電圧が 2.5 V 未満である場合にのみセットする必要があります。

0：全電圧範囲で動作する製品。低電圧での I/O スピードの最適化は無効です。

1：2.5 V 未満で動作する製品。低電圧での I/O スピードの最適化は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **RSS2**：ユーザオプション設定ビット 2

RSS2 は ST の開発コードに使用されます（RSS／ブートローダ）。このビットを変更しても影響はありません。

ビット 26 **RSS1**：ユーザオプション設定ビット 1

RSS1 は ST の開発コードに使用されます（RSS／ブートローダ）。このビットを変更しても影響はありません。

ビット 25：22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SECURITY**：セキュリティオプション設定ビット

SECURITY ビットは、オプションバイトの変更中にデバイスレベルでセキュリティ機能を有効にします。変更は、次のパワーオンリセットで有効になります。

一度有効にすると、セキュリティ機能は PCROP またはセキュアモードで保護された領域がない場合に無効化できます。保護されたセキュア領域または PCROP 領域がある場合は、レベルの回帰（レベル 1 から 0）を実行し、すべてのビットを保護されていないセキュア領域または PCROP 領域にセットします（*セクション 3.3.12：保護メカニズム*を参照）。

0：セキュリティ機能は無効です。

1：セキュリティ機能は有効です。

ビット 20:19 **ST_RAM_SIZE**：DTCM サイズ選択オプション設定ビット

ST_RAM_SIZE ビットは、オプションバイトの変更中に保護するDTCM RAM のサイズを設定するために使用します。

00：2 KB

01：4 KB

10：8 KB

11：16 KB

注：　このビットフィールドは、セキュリティを有効（SECURITY = '1'）にした場合にのみ有効です。

ビット 18 **FZ_IWDG_SDBY**：IWDG STANDBY モード凍結オプション設定ビット

FZ_IWDG_SDBY は、オプションバイトの変更中に、独立型ウォッチドッグを STANDBY モードで凍結するかどうかを選択するために使用します。

0：独立型ウォッチドッグが STANDBY モードで凍結されています。

1：独立型ウォッチドッグが STANDBY モードで動作しています。

ビット 17 **FZ_IWDG_STOP**：IWDG STOP モード凍結オプション設定ビット

FZ_IWDG_STOP は、オプションバイトの変更中に、独立型ウォッチドッグを STOP モードで凍結するかどうかを選択するために使用します。

0：独立型ウォッチドッグが STOP モードで凍結されています。

1：独立型ウォッチドッグが STOP モードで動作しています。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **RDP**：読み出し保護レベルオプション設定バイト

RDP ビットは、読み出し保護レベルを変更するために使用します。この変更は、現在の保護レベルがレベル 2 以外である場合にのみ行うことができます（*セクション ：RDP（読み出し保護）*を参照）。次のように設定できます。

0xAA：保護レベル 0 をセットします。

0xCC：保護レベル 2 をセットします。

その他の値：保護レベル 1 をセットします。

ビット 7 **nRST_STBY_D1**：D1 DStandby 後のオプションバイト消去のオプション設定ビット

nRST_STBY_D1 は、オプションバイトの変更中に使用されます。これを「1」にセットすると、D1 ドメインの DStandby モードに移行するときにオプションバイトは消去されます。

0：D1 ドメインの DStandby モードに移行するときにオプションバイトを消去します。

1：D1 ドメインの DStandby モードに移行するときにオプションバイトを消去しません。

ビット 6 **nRST_STOP_D1**：D1 DStop 後のオプションバイト消去のオプション設定ビット

nRST_STOP_D1 は、オプションバイトの変更中に使用されます。これを「1」にセットすると、D1 ドメインの DStop モードに移行するときにオプションバイトは消去されます。

0：D1 ドメインの DStop モードに移行するときにオプションバイトを消去します。

1：D1 ドメインの DStop モードに移行するときにオプションバイトを消去しません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IWDG1_HW**：IWDG1 オプション設定ビット

IWDG1_HW オプションビットは、IWDG1 独立型ウォッチドッグをハードウェアまたはソフトウェアで制御するかを選択するために使用されます。

0：IWDG1 をソフトウェアで制御します。

1：IWDG1 をハードウェアで制御します。

ビット 3:2 **BOR_LEV**：BOR リセットレベルオプション設定ビット

FLASH_OPTSR_PRG は、BOR_LEV オプションバイトを変更するために使用します。このオプションを変更するには、ユーザはオプションバイトの変更を開始する前に、BOR_LEV を必要とされる設定にプログラムする必要があります。次のように設定できます。

00 および 11：リセットレベルを 2.1 V にセットします。

01：リセットレベルを 2.4 V にセットします。

10：リセットレベルを 2.7 V にセットします。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

3.5.10 フラッシュオプションクリア制御レジスタ（FLASH_OPTCCR）

アドレスオフセット：0x024 または 0x124

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CLR_OPTCHANGEERR**：OPTCHANGEERR リセットビット

FLASH_OPTCCR は、FLASH_OPTSR_CUR レジスタの OPTCHANGEERR フラグをリセットするために使用します。FLASH_OPTCCR は書き込み専用です。

「1」をプログラムすることによってリセットされます。

ビット 29:0 予約済みであり、リセット値に保持する必要があります。

3.5.11 バンク 1 のフラッシュ保護アドレス（現在値）（FLASH_PRAR_CUR1）

アドレスオフセット：0x028

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP1	Res.	Res.	Res.	PROT_AREA_END1											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START1											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMEP1**：バンク 1 PCROP 保護消去イネーブルオプションステータスビット

DMEP1 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）が発生したときに、PCROP 保護領域が消去されます（*セクション ：PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）* および *セクション ：フラッシュバンク消去*）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END**1：バンク 1 PCROP 保護最高アドレス

これらのビットは、バンク 1 の PCROP で保護された最後のアドレスを含みます。
このアドレスが PROT_AREA_START1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_START1 より低い場合、バンク 1 には保護は設定されません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START**1：バンク 1 PCROP 保護最低アドレス

これらのビットは、バンク 1 の PCROP で保護された最初のアドレスを含みます。
このアドレスが PROT_AREA_END1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_END1 より高い場合、バンク 1 には保護は設定されません。

3.5.12 バンク 1 のフラッシュ保護アドレス（プログラムする値）（FLASH_PRAR_PRG1）

アドレスオフセット：0x02C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP1	Res.	Res.	Res.	PROT_AREA_END1											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START1											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMEP**1：バンク 1 PCROP 保護消去イネーブルオプション設定ビット

DMEP1 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）またはバンク 1 消去が発生したときに、PCROP 保護領域が消去されます（*セクション ：PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）*）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END**1：バンク 1 PCROP 保護最高アドレス設定

これらのビットにより、バンク 1 の最後の PCROP 保護アドレスを設定できます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START**1：バンク 1 PCROP 保護最低アドレス設定

これらのビットにより、バンク 1 の最初の PCROP 保護アドレスを設定できます。

3.5.13 バンク 1 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR1）

アドレスオフセット：0x030

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES1	Res.	Res.	Res.	SEC_AREA_END1											
r				r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START1											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMES**1：バンク 1 安全保護消去イネーブルオプションステータスビット

DMES1 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）が発生したときに、安全保護領域が消去されます（*セクション：セキュア領域* および *セクション：フラッシュバンク消去*）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END1**：バンク 1安全保護最高アドレス

これらのビットは、バンク 1 の最後の安全保護アドレスを含みます。
このアドレスが SEC_AREA_START1 と等しい場合、バンク 1 全体が安全に保護されます。
このアドレスが SEC_AREA_START1 より低い場合、バンク 1 には保護は設定されません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START1**：バンク 1安全保護最低アドレス

これらのビットは、バンク 1 の最初の安全保護アドレスを含みます。
このアドレスが SEC_AREA_END1 と等しい場合、バンク 1 全体が安全に保護されます。
このアドレスが SEC_AREA_END1 より高い場合、バンク 1 には保護は設定されません。

3.5.14 バンク 1 のフラッシュセキュアアドレス（プログラムする値）（FLASH_SCAR_PRG1）

アドレスオフセット：0x034

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES1	Res.	Res.	Res.	SEC_AREA_END1											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START1											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMES**1：バンク 1 安全保護消去イネーブルオプション設定ビット

DMES1 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）またはバンク 1 消去が発生したときに、安全保護領域が消去されます（*セクション：セキュア領域*）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END**1：バンク 1安全保護最高アドレス設定

これらのビットにより、バンク 1 の最後の安全保護アドレスを設定できます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START**1：バンク 1安全保護最低アドレス設定

これらのビットにより、バンク 1 の最初の安全保護アドレスを設定できます。

3.5.15 バンク 1 のフラッシュ書き込みセクタ保護（現在値）（FLASH_WPSN_CUR1R）

アドレスオフセット：0x038

リセット値：0x0000 00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn1							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRPSn**1：バンク 1 セクタ書き込み保護オプションステータスバイト

各 FLASH_WPSN_CUR1R ビットでは、対応するバンク 1 セクタの書き込み保護ステータスを反映します（*セクション：セクタ書き込み保護* を参照）。

3.5.16 バンク 1 のフラッシュ書き込みセクタ保護（プログラムする値）（FLASH_WPSN_PRG1R）

アドレスオフセット：0x03C

リセット値：0x0000 00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn1							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRPSn**1：バンク 1 セクタ書き込み保護設定バイト

WRPSn1 ビットを「0」に設定すると、対応するバンク 1 セクタを書き込み保護することができます（*セクション：セクタ書き込み保護* を参照）。

3.5.17 ブートアドレスのフラッシュレジスタ（現在値）（FLASH_BOOT_CURR）

アドレスオフセット：0x040

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BOOT_ADD1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT_ADD0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 BOOT_ADD1：ブートアドレス 1

これらのビットは、BOOT ピンがハイの場合に、ブートアドレスの MSB を反映します。

ビット 15:0 BOOT_ADD0：ブートアドレス 0

これらのビットは、BOOT ピンがローの場合に、ブートアドレスの MSB を反映します。

3.5.18 ブートアドレスのフラッシュレジスタ（プログラムする値）（FLASH_BOOT_PRGR）

アドレスオフセット：0x044

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BOOT_ADD1															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT_ADD0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 BOOT_ADD1：ブートアドレス 1 設定

これらのビットにより、BOOT ピンがハイの場合に、ブートアドレスの MSB を設定できます。

ビット 15:0 BOOT_ADD1：ブートアドレス 0 設定

これらのビットにより、BOOT ピンがローの場合に、ブートアドレスの MSB を設定できます。

3.5.19 バンク 1 のフラッシュ CRC 制御レジスタ（FLASH_CRCCR1）

アドレスオフセット：0x050

リセット値：0x001C 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC
										rw	rw			w	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	ALL_BANK	Res.	Res.	Res.	Res.	CRC_SECT		
					w	w	rw	w					rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **CRC_BURST : バンク 1** CRC バーストサイズ

CRC_BURST ビットは、CRC 計算ユニットで生成されたバーストのサイズをセットします。
00：バーストのサイズをすべて 4 フラッシュワードにします。
01：バーストのサイズをすべて 16 フラッシュワードにします。
10：バーストのサイズをすべて 64 フラッシュワードにします。
11：バーストのサイズをすべて 256 フラッシュワードにします。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CLEAN_CRC**：バンク 1 CRC クリアビット

CLEAN_CRC を「1」にセットすると、FLASH_CRCDATAR レジスタに保存された現在の CRC の結果がクリアされます。

ビット 16 **START_CRC**：バンク 1 CRC 開始ビット

START_CRC ビットは、現在の設定を使用して、バンク 1 の CRC 計算をトリガします。オプションバイトの変更操作が進行中である場合、CRC 計算を開始することはできません。これは、オプションバイトの変更操作が完了するまでフラッシュレジスタへのすべての書き込みアクセスがホールド状態になるためです。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CLEAN_SECT**：バンク 1 CRC セクタリストクリアビット

CLEAN_SECT を「1」にセットすると、CRC が計算されたセクタのリストがクリアされます。

ビット 9 **ADD_SECT**：バンク 1 CRC セクタ選択ビット

ADD_SECT を「1」にセットすると、CRC が計算されたセクタのリストに番号が CRC_SECT のセクタが追加されます。

ビット 8 **CRC_BY_SECT**：バンク 1 CRC セクタモード選択ビット

CRC_BY_SECT を「1」にセットすると、CRC の計算がセクタレベルで実行されます。これは、CRC_SECT によって選択されたセクタ、または ALL_BANK ビットがセットされている場合はすべてのバンクで行われます。

CRC_BY_SECT を「0」にリセットすると、CRC の計算が CRC_START_ADDR と CRC_END_ADDR の間のすべてのアドレスで実行されます。

ビット 7 **ALL_BANK**：バンク 1 CRC 選択ビット

ALL_BANK を「1」にセットすると、CRC が計算されたセクタのリストにすべてのバンク 1 ユーザセクタが追加されます。

ビット 6:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CRC_SECT**：バンク 1 CRC セクタ番号

CRC_SECT は、CRC 計算に追加する 1 つ以上のセクタを選択するために使用します。CRC は、2 つのアドレス間（開始アドレスと終了アドレス）またはセクタのリスト上で計算することができます。後者のオプションを選択した場合は、CRC_SECT にセクタ番号をプログラムし、ADD_SECT を「1」にセットすることで、セクタのリストにセクタを追加することができます。

セクタのリストは、CLEAN_SECT ビットをセットするか、CRC 計算を無効にすると、消去することができます。各セクタの番号を知るには、FLASH_CR レジスタの SNB1 ビットの説明を参照してください（*セクション 3.5.4：バンク 1 のフラッシュ制御レジスタ（FLASH_CR1）*を参照）。

CRC_SECT は、FLASH_CR レジスタの CRC_EN を「1」にセットしている場合にのみセットすることができます。

3.5.20 バンク 1 のフラッシュ CRC 開始アドレスレジスタ（FLASH_CRCSADD1R）

アドレスオフセット：0x054

リセット値：0x0000 0000

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

ビット 31:0 **CRC_START_ADDR**：バンク 1 の CRC 開始アドレス

CRC_START_ADDR は、CRC_BY_SECT が「0」のときに使用されます。これは、CRC 計算が実行されるバンク 1 メモリ領域の開始アドレスをプログラムする必要があります。

3.5.21 バンク 1 のフラッシュ CRC 終了アドレスレジスタ（FLASH_CRCEADD1R）

アドレスオフセット：0x058

リセット値：0x0000 0000

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

ビット 31:0 **CRC_END_ADDR**：バンク 1 の CRC 終了アドレス

CRC_END_ADDR は、CRC_BY_SECT が「0」のときに使用されます。これは、CRC 計算が実行されるバンク 1 メモリ領域の終了アドレスをプログラムする必要があります。

3.5.22 フラッシュ CRC データレジスタ（FLASH_CRCDATAR）

アドレスオフセット：0x05C または 0x15C

リセット値：0x0000 0000

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

ビット 31:0 **CRC_DATA**：CRC 結果

CRC_DATA は、CRC 計算の結果を含みます。

3.5.23 バンク 1 のフラッシュ ECC 失敗アドレス（FLASH_ECC_FA1R）

アドレスオフセット：0x060

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FAIL_ECC_ADDR1														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **FAIL_ECC_ADDR1**：バンク 1 ECC エラーアドレス

バンク 1 からの読み出し操作中に ECC エラー（シングル訂正またはダブル検出共に）が発生すると、FAIL_ECC_ADDR1 ビットフィールドは、エラーを生成したアドレスを含みます。

FAIL_ECC_ADDR1 は、FLASH_SR1 レジスタ（CLR_SNECCERR1 または CLR_DBECCERR1）のフラグエラーをリセットした場合に、リセットされます。

フラッシュメモリインタフェースでは、ECC エラーフラグがセットされていない場合にのみ、このレジスタのアドレスをプログラムします。つまり、ECC エラーを生成した最初のアドレスのみが保存されるということです。

3.5.24 バンク 2 のフラッシュキーレジスタ（FLASH_KEYR2）

アドレスオフセット：0x104

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR2															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR2															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **KEYR2**：バンク 2 アクセス設定アンロックキー

FLASH_KEYR2 は書き込み専用レジスタです。FLASH_CR2 レジスタのロックを解除し、プログラミングや消去を許可するには、次の値を連続してプログラムする必要があります。

a) 1 番目のキー = 0x4567 0123

b) 2 番目のキー = 0xCDEF 89AB

詳細については、*セクション：プログラミングパラメータの設定* を参照してください。

3.5.25 バンク 2 のフラッシュ制御レジスタ（FLASH_CR2）

アドレスオフセット：0x10C

リセット値：0x0000 0031

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CRCENDIE2	DBECCERRIE2	SNECCERRIE2	RDSERRIE2	RDPERRIE2	OPERRIE2	INCERRIE2	Res.	STRBERRIE2	PGSERRIE2	WRPERRIE2	EOPIE2
				rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_EN	Res.	Res.	Res.	Res.	SNB2			START2	FW2	PSIZE2		BER2	SER2	PG2	LOCK2
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rs

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CRCENDIE2**：バンク 2 CRC 計算終了割り込みイネーブルビット

CRCENDIE2 ビットを「1」にセットすると、バンク 2 で CRC 計算が完了したときに割り込みが生成されます。CRCENDIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で CRC 計算が完了したときに、割り込みを生成しません。

1：バンク 2 で CRC 計算が完了したときに、割り込みを生成します。

ビット 26 **DBECCERRIE2**：バンク 2 ECC ダブル検出エラー割り込みイネーブルビット

DBECCERRIE2 ビットを「1」にセットすると、バンク 2 からの読み出し操作中に ECC ダブル検出エラーが発生したときに割り込みが生成されます。DBECCERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で ECC ダブル検出エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で ECC ダブル検出エラーが発生したときに、割り込みを生成します。

ビット 25 **SNECCERRIE2**：バンク 2 ECC シングル訂正エラー割り込みイネーブルビット

SNECCERRIE2 ビットを「1」にセットすると、バンク 2 からの読み出し操作中に ECC シングル訂正エラーが発生したときに割り込みが生成されます。SNECCERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で ECC シングル訂正エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で ECC シングル訂正エラーが発生したときに、割り込みを生成します。

ビット 24 **RDSERRIE2**：バンク 2 セキュアエラー割り込みイネーブルビット

RDSERRIE2 ビットを「1」にセットすると、バンク 2 からの読み出し操作中セキュアエラー（適切な権限なく安全に保護されたアドレスにアクセス）が発生したときに割り込みが生成されます。RDSERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 でセキュアエラーが発生したときに、割り込みを生成しません。

1：バンク 2 でセキュアエラーが発生したときに、割り込みを生成します。

ビット 23 **RDPERRIE2**：バンク 2 読み出し保護エラー割り込みイネーブルビット

RDPERRIE2 ビットを「1」にセットすると、バンク 2 からの読み出し操作中に読み出し保護エラー（PCROP で保護されたアドレスへのアクセス）が発生したときに割り込みが生成されます。RDPERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で読み出し保護エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で読み出し保護エラーが発生したときに、割り込みを生成します。

ビット 22 **OPERRIE2**：バンク 2 書き込み／消去エラー割り込みイネーブルビット

OPERRIE2 ビットを「1」にセットすると、バンク 2 への書き込み／消去操作中にエラーを検出したときに割り込みが生成されます。OPERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で書き込み／消去エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で書き込み／消去エラーが発生したときに、割り込みを生成します。

ビット 21 **INCERRIE**2：バンク 2 不整合エラー割り込みイネーブルビット

INCERRIE2 ビットを「1」にセットすると、バンク 2 への書き込み操作中に不整合エラーが発生したときに割り込みが生成されます。INCERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で不整合エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で不整合エラーが発生したときに、割り込みを生成します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERRIE2**：バンク 2 ストローブエラー割り込みイネーブルビット

STRBERRIE2 ビットを「1」にセットすると、バンク 2 への書き込み操作中にストローブエラー（マスタによって書き込みバッファに数回同じバイトをプログラム）が発生したときに割り込みが生成されます。STRBERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 でストローブエラーが発生したときに、割り込みを生成しません。

1：バンク 2 でストローブエラーが発生したときに、割り込みを生成します。

ビット 18 **PGSERRIE2**：バンク 2 プログラミングシーケンスエラー割り込みイネーブルビット

PGSERRIE2 ビットを「1」にセットすると、バンク 2 へのプログラム操作中にシーケンスエラーが発生したときに割り込みが生成されます。PGSERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 でシーケンスエラーが発生したときに、割り込みを生成しません。

1：バンク 2 でシーケンスエラーが発生したときに、割り込みを生成します。

ビット 17 **WRPERRIE2**：バンク 2 書き込み保護エラー割り込みイネーブルビット

WRPERRIE2 ビットを「1」にセットすると、バンク 2 へのプログラム操作中に保護エラーが発生したときに割り込みが生成されます。WRPERRIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 で保護エラーが発生したときに、割り込みを生成しません。

1：バンク 2 で保護エラーが発生したときに、割り込みを生成します。

ビット 16 **EOPIE2**：バンク 2 プログラム終了割り込み制御ビット

EOPIE2 ビットを「1」にセットすると、バンク 2 へのプログラム操作の終了時に割り込み生成が有効になります。EOPIE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

0：バンク 2 へのプログラム操作の終了時に、割り込みを生成しません。

1：バンク 2 へのプログラム操作の終了時に、割り込みを生成します。

ビット 15 **CRC_EN**：バンク 2 CRC 制御ビット

CRC_EN ビットを「1」にセットすると、バンク 2 で CRC 計算が有効になります。CRC_EN では CRC 計算を開始しませんが、FLASH_CRCCR2 レジスタを通じて CRC 設定を有効にします。

バンク 2 で CRC 計算を実行した場合は、CRC_EN ビットを「0」に設定することでのみ無効化することができます。CRC_EN をリセットすると、FLASH_CRCDATAR レジスタの内容もリセットされます。

CRC_EN は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **SNB2**：バンク 2 セクタ消去選択番号

これらのビットは、セクタ消去操作の対象のセクタを選択するために使用します。SNB2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。最上位ビットを「1」にセットすると、3 番目のビットは ICP セクタとオプションセクタのどちらかを選択するために使用されます。000 0000：バンク 2 のセクタ 0

000 0001：バンク 2 のセクタ 1

...

000 0111：バンク 2 のセクタ 7

000 1000 から 011 1111：予約済み（この設定にすることはできません）

...

011 1111：予約済み（この設定にすることはできません）

100 0000：バンク 2 の ICP セクタ

他の設定：予約済み

ビット 7 **START2**：バンク 2 バンク消去またはセクタ消去開始制御ビット

START2 ビットは、バンク消去またはセクタ消去操作を開始するために使用します。START2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

該当する操作を確認すると、フラッシュメモリインタフェースでは START2 をリセットします。該当する操作を確認するまで、ユーザアプリケーションはどのフラッシュレジスタにもアクセスできません。

ビット 6 **FW2**：バンク 2 書き込み強制制御ビット

FW2 は書き込みバッファがフルでなくても書き込み操作を強制します。FW2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

該当する操作を確認すると、フラッシュメモリインタフェースでは FW2 をリセットします。該当する操作を確認するまで、ユーザアプリケーションはどのフラッシュレジスタにもアクセスできません。

書き込みの強制は、書き込みバッファが空でない場合にのみ有効です（特に書き込み操作が連続して行われている場合、FW2 ではいくつかの書き込み操作を開始しません）。

ビット 5:4 **PSIZE2**：バンク 2 プログラムサイズ

PSIZE2 は、バンク 2 への書き込みおよび消去操作中に、フラッシュメモリが使用する並列度を選択します（詳細については、*セクション：プログラミングパラメータの設定* を参照）。PSIZE2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

00：バイト並列処理でプログラミングを実行します。

01：ハーフワード並列処理でプログラミングを実行します。

10：ワード並列処理でプログラミングを実行します。

11：ダブルワード並列処理でプログラミングを実行します。

ビット 3 **BER2**：バンク 2 消去リクエスト

BER2 ビットを「1」にセットすると、バンク 2 でバンク消去操作が要求されます。BER2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

BER2 の優先順位は SER2 よりも高いため、両方がセットされている場合、フラッシュメモリインタフェースではバンク消去を実行します（詳細については、*セクション：フラッシュバンク消去* を参照）。

0：バンク消去をバンク 2 で要求しません。

1：バンク消去をバンク 2 で要求します。

ビット 2 **SER2**：バンク 2 セクタ消去リクエスト

SER2 ビットを「1」にセットすると、バンク 2 でセクタ消去操作が要求されます。SER2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。

BER2 の優先順位は SER2 よりも高いため、両方がセットされている場合、フラッシュメモリインタフェースではバンク消去を実行します （詳細については、*セクション：フラッシュセクタの消去* を参照）。

0：セクタ消去をバンク 2 で要求しません。

1：セクタ消去をバンク 2 で要求します。

ビット 1 **PG2**：バンク 2 プログラムイネーブルビット

PG2 ビットを「1」にセットすると、バンク 2 への書き込み操作が有効になります。これにより、セクタ消去やバンク消去の進行中でもプログラム操作を準備できるようになります。

PG2 は LOCK2 を「0」に設定している場合にのみプログラムすることができます。PG2 がリセットされると、バンク 2 への書き込み操作について内部バッファが無効化されます。また、バッファに保存された、プログラミングされていないデータはすべて失われます。

ビット 0 **LOCK2**：バンク 2 設定ロックビット

このビットでは、FLASH_CR2 レジスタをロックします。

FLASH_CR2 レジスタがアンロックされている場合、LOCK2 ビットは自動的にリセットされます（*セクション：プログラミングパラメータの設定* を参照）。誤ったシーケンスが実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

LOCK2 は「1」にプログラムすることでセットすることができます。「1」にセットすると、アンロックするには新しいアンロックシーケンスが必要になります。LOCK2 を「0」から「1」に変更するときに、FLASH_CR2 レジスタのその他のビットは変更されません。

3.5.26 バンク 2 のフラッシュステータスレジスタ（FLASH_SR2）

アドレスオフセット：0x110

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CRCEND2	DBECCERR2	SNECCERR2	RDSERR2	RDPERR2	OPERR2	INCERR2	Res.	STRBERR2	PGSERR2	WRPERR2	EOP2
				r	r	r	r	r	r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY2	QW2	WBNE2	BSY2
												r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CRCEND2**：バンク 2 CRC 完了フラグ

バンク 2 で CRC 計算が完了すると、CRCEND2 ビットが生成されます。CRCENDIE2 が「1」にセットされている場合、割り込みが生成されます。CRC 計算を再開する前に CRCEND2 をリセットする必要はありません。FLASH_CCR2 レジスタの CLR_CRCEND2 ビットに「1」を書き込むと CRCEND2 はクリアされます。

0：バンク 2 の CRC 計算が未完了の場合。

1：バンク 2 の CRC 計算が完了した場合。

ビット 26 **DBECCERR2**：バンク 2 ECC ダブル検出エラーフラグ

DBECCERR2 フラグは、バンク 2 からの読み出し操作中に ECC ダブル検出エラーが発生した場合に立てられます。DBECCERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_DBECCERR2 ビットに「1」を書き込むと DBECCERR2 はクリアされます。

0：ECC ダブル検出エラーがバンク 2 で発生しない場合。

1：ECC ダブル検出エラーがバンク 2 で発生する場合。

ビット 25 **SNECCERR2**：バンク 2 シングル訂正エラーフラグ

SNECCERR2 フラグは、バンク 2 からの読み出し操作中に ECC シングル訂正エラーが発生した場合に立てられます。SNECCERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_SNECCERR2 ビットに「1」を書き込むと SNECCERR2 はクリアされます。

0：ECC シングル訂正エラーがバンク 2 で発生しない場合。

1：0：ECC シングル訂正エラーがバンク 2 で発生する場合。

ビット 24 **RDSERR2**：バンク 2 セキュアエラーフラグ

RDSERR2 フラグは、バンク 2 でセキュアエラー（適切な権限なく安全に保護されたワードにアクセス）が発生した場合に立てられます。RDSERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_RDSERR2 ビットに「1」を書き込むと RDSERR2 はクリアされます。

0：バンク 2 でセキュアエラーが発生していません。

1：バンク 2 でセキュアエラーが発生しています。

ビット 23 **RDPERR2**：バンク 2 読み出し保護エラーフラグ

RDPERR2 フラグは、バンク 2 で読み出し保護エラー（PCROP で保護されたワードへのアクセス）が発生した場合に立てられます。RDPERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_RDPERR2 ビットに「1」を書き込むと RDPERR2 はクリアされます。

0：バンク 2 で読み出し保護エラーが発生していません。

1：バンク 2 で読み出し保護エラーが発生しています。

ビット 22 **OPERR2**：バンク 2 書き込み／消去エラーフラグ

OPERR2 フラグは、バンク 2 での書き込み／消去操作中にエラーが発生した場合に立てられます。OPERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_OPERR2 ビットに「1」を書き込むと OPERR2 はクリアされます。

0：バンク 2 で書き込み／消去エラーが発生していません。

1：バンク 2 で書き込み／消去エラーが発生しています。

ビット 21 **INCERR2**：バンク 2 不整合エラーフラグ

INCERR2 フラグは、バンク 2 で不整合エラーが発生した場合に立てられます。INCERRIE2 が「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_INCERR2 ビットに「1」を書き込むと INCERR2 はクリアされます。

0：バンク 2 で不整合エラーが発生していません。

1：バンク 2 で不整合エラーが発生しています。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERR2**：バンク 2 ストローブエラーフラグ

STRBERR2 フラグは、バンク 2 でストローブエラー（マスタによって書き込みバッファへの同じバイトの書き込みが数回試行されること）が発生した場合に立てられます。STRBERRIE2 ビットが「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_STRBERR2 ビットに「1」を書き込むと STRBERR2 はクリアされます。

0：バンク 2 でストローブエラーが発生していません。

1：バンク 2 でストローブエラーが発生しています。

ビット 18 **PGSERR2**：バンク 2 プログラミングシーケンスエラーフラグ

PGSERR2 フラグは、バンク 2 でシーケンスエラーが発生した場合に立てられます。PGSERRIE2 ビットが「1」にセットされている場合、割り込みが生成されます。FLASH_CCR2 レジスタの CLR_PGSERR2 ビットに「1」を書き込むと PGSERR2 はクリアされます。

0：バンク 2 でシーケンスエラーが発生していません。

1：バンク 2 でシーケンスエラーが発生しています。

ビット 17 **WRPERR2**：バンク 2 書き込み保護エラーフラグ

WRPERR2 フラグは、バンク 2 へのプログラム操作中に保護エラーが発生した場合に立てられます。EOPIE2 が「1」にセットされている場合は、割り込みも生成されます。FLASH_CCR2 レジスタの CLR_EOP2 ビットに「1」を書き込むと WRPERR2 はクリアされます。

0：バンク 2 で保護エラーが発生していません。

1：バンク 2 で保護エラーが発生しています。

ビット 16 **EOP2**：バンク 2 プログラム終了フラグ

EOP2 フラグは、バンク 2 へのプログラム操作が完了した場合にセットされます。EOPIE2 が「1」にセットされている場合、割り込みが生成されます。新しい操作を開始する前に EOP2 をリセットする必要はありません。EOP2 ビットは、FLASH_CCR2 レジスタの CLR_EOP2 ビットに「1」を書き込むことによりクリアされます。

0：バンク 2 に完了したプログラム操作がありません。

1：バンク 2 に完了したプログラム操作があります。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRC_BUSY2**：バンク 2 CRC ビジーフラグ

CRC_BUSY2 フラグは、バンク 2 での CRC 計算の進行中にセットされます。このビットを強制的に 0 にすることはできません。ユーザは、CRC 計算が完了するまで待つか、またはバンク 2 の CRC 計算を無効にする必要があります。

0：バンク 2 に進行中の CRC 計算がありません。

1：バンク 2 に進行中の CRC 計算があります。

ビット 2 **QW2**：バンク 2 ウェイトキューフラグ

QW2 フラグは、バンク 2 へのプログラム操作がウェイトキューにある場合にセットされます。キューに含まれるプログラム操作の種類を知ることはできません。すべてのプログラム操作を実行してウェイトキューから削除すると、このフラグはハードウェアによってリセットされます。このビットを強制的に 0 にすることはできません。リセットは、他に要求された操作がない場合は、決定論的な間隔を空けて実行されます。

0：バンク 2 のウェイトキューにプログラム操作がありません。

1：バンク 2 のウェイトキューに少なくとも 1 つのプログラム操作があります。

ビット 1 **WBNE2**：バンク 2 書き込みバッファノットエンプティフラグ

WBNE2 フラグは、バンク 2 の書き込みバッファが空ではなく、フラッシュメモリインタフェースが新しいデータの完了を待っている場合にのみセットされます。

WBNE2 は書き込みバッファが空になるたびにハードウェアによってリセットされます。これは、次のいずれかのイベントが発生したときに生じます。

- 書き込みバッファがフルである場合
- ユーザが書き込み操作を強制した場合
- データの損失を伴うエラー
- 書き込み操作が無効化された場合

このビットを強制的に 0 にすることはできません。リセットするには、上記のいずれかのアクションを実行して、書き込みバッファをクリアします。

0：バンク 2の書き込みバッファが空です。

1：バンク 2 の書き込みバッファがデータの書き込み完了を待っています。

ビット 0 **BSY2**：バンク 2 プログラム進行中フラグ

BSY2 フラグは、バンク 1 へのプログラム操作が進行中である場合にセットされます。進行中のプログラム操作の種類を知ることはできません。

BSY2 を強制的に 0 にすることはできません。操作完了時に他のプログラム操作が開始しない場合に、ハードウェアによってリセットされます。

0：バンク 2 に実行中のプログラム操作がありません。

1：バンク 2 に実行中のプログラム操作があります。

3.5.27 バンク 2 のフラッシュクリア制御レジスタ（FLASH_CCR2）

アドレスオフセット：0x114

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CLR_CRCEND2	CLR_DBECCERR2	CLR_SNECCERR2	CLR_RDSERR2	CLR_RDPERR2	CLR_OPERR2	CLR_INCERR2	Res.	CLR_STRBERR2	CLR_PGSERR2	CLR_WRPERR2	CLR_EOP2
				w	w	w	w	w	w	w		w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CLR_CRCEND**2：バンク 2 CRCEND2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの CRCEND2 フラグが「0」にリセットされます。

ビット 26 **CLR_DBECCERR**2：バンク 2 DBECCERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの DBECCERR2 フラグが「0」にリセットされます。FLASH_SR2 レジスタの SNECCERR2 フラグが「0」に設定されている場合、FLASH_ECC_FA2R レジスタも「0」にリセットされます。

ビット 25 **CLR_SNECCERR**2：バンク 2 SNECCERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの SNECCERR2 フラグが「0」にリセットされます。FLASH_SR2 レジスタの DBECCERR2 フラグが「0」に設定されている場合、FLASH_ECC_FA2R レジスタも「0」にリセットされます。

ビット 24 **CLR_RDSERR**2：バンク 2 RDSERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの RDSERR2 フラグが「0」にリセットされます。

ビット 23 **CLR_RDPERR**2：バンク 2 RDPERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの RDPERR2 フラグが「0」にリセットされます。

ビット 22 **CLR_OPERR**2：バンク 2 OPERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの **OPERR**2 フラグが「0」にリセットされます。

ビット 21 **CLR_INCERR**2：バンク 2 INCERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの INCERR2 フラグが「0」にリセットされます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CLR_STRBERR**2：バンク 2 STRBERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの STRBERR2 フラグが「0」にリセットされます。

ビット 18 **CLR_PGSERR**2：バンク 2 PGSERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの PGSERR2 フラグが「0」にリセットされます。

ビット 17 **CLR_WRPERR**2：バンク 2 WRPERR2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの WRPERR2 フラグが「0」にリセットされます。

ビット 16 **CLR_EOP**2：バンク 2 EOP2 フラグクリアビット

このビットを「1」にセットすると、FLASH_SR2 レジスタの EOP2 フラグが「0」にリセットされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

3.5.28 バンク 2 のフラッシュ保護アドレス（現在値）（FLASH_PRAR_CUR2）

アドレスオフセット：0x128

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP2	Res.	Res.	Res.	PROT_AREA_END2											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START2											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMEP2**：バンク 2 PCROP 保護消去イネーブルオプションステータスビット

DMEP2 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）が発生したときに、PCROP 保護領域が消去されます（*セクション ：PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）* および *セクション ：フラッシュバンク消去* を参照）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END2**：バンク 2 最高 PCROP 保護アドレス

これらのビットは、バンク 2 の PCROP で保護された最後のアドレスを含みます。

このアドレスが PROT_AREA_START2 と等しい場合、バンク 2 全体が PCROP 保護されます。

このアドレスが PROT_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START2**：バンク 2 最低 PCROP 保護アドレス

これらのビットは、バンク 2 の PCROP で保護された最初のアドレスを含みます。

このアドレスが PROT_AREA_END2 と等しい場合、バンク 2 全体が PCROP 保護されます。

このアドレスが PROT_AREA_END2 より高い場合、バンク 1 には保護は設定されません。

3.5.29 バンク 2 のフラッシュ保護アドレス（プログラムする値）（FLASH_PRAR_PRG2）

アドレスオフセット：0x12C

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP2	Res.	Res.	Res.	PROT_AREA_END2											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START2											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMEP2**：バンク 2 PCROP 保護消去イネーブルオプション保護ビット
DMEP2 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）またはバンク 2 消去が発生したときに、PCROP 保護領域が消去されます（*セクション：PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）*を参照）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END2**：バンク 2 最高 PCROP 保護アドレス設定
これらのビットにより、バンク 2 の最後の PCROP 保護アドレスを設定できます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START2**：バンク 2 最低 PCROP 保護アドレス設定
これらのビットにより、バンク 2 の最初の PCROP 保護アドレスを設定できます。

3.5.30 バンク 2 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR2）

アドレスオフセット：0x130

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES2	Res.	Res.	Res.	SEC_AREA_END2											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START2											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMES2**：バンク 2 セキュア保護消去イネーブルオプションステータスビット

DMES2 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）が発生したときに、安全保護領域が消去されます（*セクション：セキュア領域* および *セクション：フラッシュバンク消去* を参照）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END2**：バンク 2安全保護最高アドレス

これらのビットは、バンク 2 の最後の安全保護アドレスを含みます。
このアドレスが SEC_AREA_START2 と等しい場合、バンク 2 全体が安全に保護されます。
このアドレスが SEC_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START2**：バンク 2安全保護最低アドレス

これらのビットは、バンク 2 の最初の安全保護アドレスを含みます。
このアドレスが SEC_AREA_END2 と等しい場合、バンク 2 全体が安全に保護されます。
このアドレスが SEC_AREA_END2 より高い場合、バンク 2 には保護はセットされません。

3.5.31 バンク 2 のフラッシュセキュアアドレス（プログラムする値）（FLASH_SCAR_PRG2）

アドレスオフセット：0x134

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES2	Res.	Res.	Res.	SEC_AREA_END2											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START2											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMES2**：バンク 2 安全保護消去イネーブルオプション設定ビット

DMES2 が「1」にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）またはバンク 2 消去が発生したときに、安全保護領域が消去されます（*セクション：セキュア領域* を参照）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END2**：バンク 2安全保護最高アドレス設定

これらのビットにより、バンク 2 の最後の安全保護アドレスを設定できます。このビットは、セキュアモードでのみプログラムできます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START2**：バンク 2安全保護最低アドレス設定

これらのビットにより、バンク 2 の最初の安全保護アドレスを設定できます。このビットは、セキュアモードでのみプログラムできます。

3.5.32 バンク 2 のフラッシュ書き込みセクタ保護（現在値）（FLASH_WPSN_CUR2R）

アドレスオフセット：0x138

リセット値：0x0000 00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn2							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRPSn**2：バンク 2 セクタ書き込み保護ステータスバイト

各 FLASH_WPSN_CUR2R ビットは、対応するバンク 2 セクタの書き込み保護ステータスを反映します。

3.5.33 バンク 2 のフラッシュ書き込みセクタ保護（プログラムする値）（FLASH_WPSN_PRG2R）

アドレスオフセット：0x13C

リセット値：0x0000 00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn2							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRPSn**2：バンク 2 セクタ書き込み保護設定バイト

WRPSn2 ビットを「0」に設定すると、対応するバンク 2 セクタを書き込み保護することができます。

3.5.34 バンク 2 のフラッシュ CRC 制御レジスタ（FLASH_CRCCR2）

アドレスオフセット：0x150

リセット値：0x001C 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC
										rw	rw			w	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	ALL_BANK	Res.	Res.	Res.	Res.	CRC_SECT		
					w	w	rw	w					rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **CRC_BURST**：バンク 2 CRC バーストサイズ

CRC_BURST ビットは、CRC 計算ユニットで生成されたバーストのサイズをセットします。

00：バーストのサイズをすべて 4 ワードにします。

01：バーストのサイズをすべて 16 ワードにします。

10：バーストのサイズをすべて 64 ワードにします。

11：バーストのサイズをすべて 256 ワードにします。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CLEAN_CRC**：バンク 2 CRC クリアビット

CLEAN_CRC を「1」にセットすると、FLASH_CRCDATAR レジスタに保存された現在の CRC の結果がクリアされます。

ビット 16 **START_CRC**：バンク 2 CRC 開始ビット

START_CRC ビットは、現在の設定を使用して、バンク 2 の CRC 計算をトリガします。オプションバイトの変更操作が進行中である場合、CRC 計算を開始することはできません。これは、すべてのオプションバイトの変更操作が完了するまでフラッシュレジスタへのすべての書き込みアクセスがホールド状態になるためです。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CLEAN_SECT**：バンク 2 CRC セクタリストクリアビット

CLEAN_SECT を「1」にセットすると、CRC が計算されたセクタのリストがクリアされます。

ビット 9 **ADD_SECT**：バンク 2 CRC セクタ選択ビット

ADD_SECT を「1」にセットすると、CRC が計算されたセクタのリストに番号が CRC_SECT のセクタが追加されます。

ビット 8 **CRC_BY_SECT**：バンク 2 CRC セクタモード選択ビット

CRC_BY_SECT を「1」にセットすると、CRC の計算がセクタレベルで実行されます。これは、CRC_SECT によって選択されたセクタで行われます。

CRC_BY_SECT を「0」にリセットすると、CRC の計算が CRC_START_ADDR と CRC_END_ADDR の間のすべてのアドレスで実行されます。

ビット 7 **ALL_BANK**：バンク 2 CRC 選択ビット

ALL_BANK を「1」にセットすると、CRC が計算されるセクタのリストにすべてのバンク 2 ユーザセクタが追加されます。

ビット 6:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CRC_SECT**：バンク 2 CRC セクタ番号

CRC_SECT は、CRC 計算に追加する 1 つ以上のセクタを選択するために使用します。CRC は、2 つのアドレス間（開始アドレスと終了アドレス）またはセクタのリスト上で計算することができます。後者のオプションを選択した場合は、CRC_SECT でセクタ番号をプログラムし、ADD_SECT を「1」にセットすることで、セクタのリストにセクタを追加することができます。

セクタのリストは、CLEAN_SECT ビットをセットするか、CRC 計算を無効にすると、消去することができます。各セクタの番号を知るには、FLASH_CR レジスタの SNB2 ビットの説明を参照してください（*セクション 3.5.25：バンク 2 のフラッシュ制御レジスタ（FLASH_CR2）*を参照）。

CRC_SECT は、FLASH_CR レジスタの CRC_EN を「1」にセットしている場合にのみセットすることができます。

3.5.35 バンク 2 のフラッシュ CRC 開始アドレスレジスタ（FLASH_CRCSADD2R）

アドレスオフセット：0x154

リセット値：0x0000 0000

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

ビット 31:0 **CRC_START_ADDR**：バンク 2 の CRC 開始アドレス

CRC_START_ADDR は、CRC_BY_SECT が「0」のときに使用されます。これは、CRC 計算が実行されるバンク 2 メモリ領域の開始アドレスに対してプログラムする必要があります。

3.5.36 バンク 2 のフラッシュ CRC 終了アドレスレジスタ（FLASH_CRCEADD2R）

アドレスオフセット：0x158

リセット値：0x0000 0000

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

ビット 31:0 **CRC_END_ADDR**：バンク 2 の CRC 終了アドレス

CRC_END_ADDR は、CRC_BY_SECT が「0」のときに使用されます。これは、CRC 計算が実行されるバンク 2 メモリ領域の終了アドレスをプログラムする必要があります。

3.5.37 バンク 2 のフラッシュ ECC 失敗アドレス（FLASH_ECC_FA2R）

アドレスオフセット：0x160

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FAIL_ECC_ADDR2														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **FAIL_ECC_ADDR2**：バンク 2 ECC エラーアドレス

バンク 2 からの読み出し操作中に ECC エラー（シングルエラー訂正またはダブル検出ともに）が発生すると、FAIL_ECC_ADDR2 ビットフィールドは、エラーを生成したアドレスを含みます。

FAIL_ECC_ADDR2 は、FLASH_SR2 レジスタ（CLR_SNECCERR2 または CLR_DBECCERR2）のフラグエラーをリセットした場合に、リセットします。

フラッシュメモリインタフェースでは、ECC エラーフラグがセットされていない場合にのみ、このレジスタのアドレスをプログラムします。つまり、ECC エラーを生成した最初のアドレスのみが保存されるということです。

3.6 フラッシュレジスタマップとリセット値

表 16. レジスタマップとリセット値の表

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRHIGHFREQ [1:0]		Res.	LATENCY [3:0]		
	0x00000037																											1	1		1	1	1
0x004	FLASH_KEYR1	KEYR1 [31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	FLASH_ OPTKEYR	OPTKEYR [31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	FLASH_CR1	Res.	Res.	Res.	Res.	CRCENDIE1	DBECCERRIE1	SNECCERRIE1	RDSERRIE1	RDPERRIE1	OPERRIE1	INCERRIE1	Res.	STRBERRIE1	PGSERRIE1	WRPERRIE1	EOPIE1	CRC_EN	Res.	Res.	Res.	Res.	SNB1 [2:0]			START1	FW1	PSIZE1 [1:0]		BER1	SER1	PG1	LOCK1
	0x00000031					0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	1	1	0	0	0	1
0x010	FLASH_SR1	Res.	Res.	Res.	Res.	CRCEND1	DBECCERR1	SNECCERR1	RDSERR1	RDPERR1	OPERR1	INCERR1	Res.	STRBERR1	PGSERR1	WRPERR1	EOP1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY1	QW1	WBNE1	BSY1
	0x00000000					0	0	0	0	0	0	0		0	0	0	0													0	0	0	0
0x014	FLASH_CCR1	Res.	Res.	Res.	Res.	CLR_CRCEND1	CLR_DBECCERR1	CLR_SNECCERR1	CLR_RDSERR1	CLR_RDPERR1	CLR_OPERR1	CLR_INCERR1	Res.	CLR_STRBERR1	CLR_PGSERR1	CLR_WRPERR1	CLR_EOP1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000					0	0	0	0	0	0	0		0	0	0	0																
0x018	FLASH_OPTCR	SWAP_BANK	OPTCHANGEERRIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MER	Res.	Res.	OPTSTART	OPTLOCK
	0x00000001	0	0																										1			0	1

表 16. レジスタマップとリセット値の表（続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	FLASH_OPTSR_CUR	SWAP_BANK_OPT	OPTCHANGEERR	IO_HSLV	PERSO_OK	RSS2	RSS1	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE[1:0]		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.	RDP[7:0]								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV[1:0]		Res.	OPT_BUSY
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X		X
0x020	FLASH_OPTSR_PRG	SWAP_BANK_OPT	Res.	IO_HSLV	Res.	RSS2	RSS1	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE[1:0]		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.	RDP[7:0]								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV[1:0]		Res.	Res.
	0xXXXX XXXX	X		X		X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X		
0x024	FLASH_OPTCCR	Res.	CLR_OPTCHANGEERR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000		0																														
0x028	FLASH_PRAR_CUR1	DMEP1	Res.	Res.	Res.	PROT_AREA_END1[11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START1[11:0]											
	0xXXXX XXXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x02C	FLASH_PRAR_PRG1	DMEP1	Res.	Res.	Res.	PROT_AREA_END1[11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START1[11:0]											
	0xXXXX XXXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x030	FLASH_SCAR_CUR1	DMES1	Res.	Res.	Res.	SEC_AREA_END1[11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START1[11:0]											
	0xXXXX XXXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x034	FLASH_SCAR_PRG1	DMES1	Res.	Res.	Res.	SEC_AREA_END1[11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START1[11:0]											
	0xXXXX XXXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x038	FLASH_WPSN_CUR1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn1[7:0]							
	0x000000FF																									X	X	X	X	X	X	X	X
0x03C	FLASH_WPSN_PRG1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn1[7:0]							
	0x000000FF																									X	X	X	X	X	X	X	X

表 16. レジスタマップとリセット値の表（続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x040	FLASH_BOOT_CURR	BOOT_ADD1[15:0]																BOOT_ADD0[15:0]															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x044	FLASH_BOOT_PRGR	BOOT_ADD1[15:0]																BOOT_ADD0[15:0]															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x050	FLASH_CRCCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BURST[1:0]		Res.	Res.	CLEAN_CRC	START_CRC	Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	ALL_BANK	Res.	Res.	Res.	Res.	CRC_SECT[2:0]		
	0x001C0000											0	1			0	0						0	0	0	0					0	0	0
0x054	FLASH_CRCSADD1R	CRC_START_ADDR[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	FLASH_CRCEADD1R	CRC_END_ADDR[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	FLASH_CRCDATAR	CRC_DATA[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	FLASH_ECC_FA1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FAIL_ECC_ADDR1[14:0]														
	0x00000000																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRHIGHFREQ[1:0]		LATENCY[3:0]			
	0x00000037																											1	1	0	1	1	1
0x104	FLASH_KEYR2	KEYR2[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x108	FLASH_OPTKEYR	OPTKEYR[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10C	FLASH_CR2	Res.	Res.	Res.	Res.	CRCENDIE2	DBECCERRIE2	SNECCERRIE2	RDSERRIE2	RDPERRIE2	OPERRIE2	INCERRIE2	Res.	STRBERRIE2	PGSERRIE2	WRPERRIE2	EOPIE2	CRC_EN	Res.	Res.	Res.	Res.	SNB2[2:0]			START2	FW2	PSIZE2[1:0]		BER2	SER2	PG2	LOCK2
	0x00000031					0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	1	1	0	0	0	1

参考資料

表 16. レジスタマップとリセット値の表（続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x110	FLASH_SR2	Res.	Res.	Res.	Res.	CRCEND2	DBECCERR2	SNECCERR2	RDSERR2	RDPERR2	OPERR2	INCERR2	Res.	STRBERR2	PGSERR2	WRPERR2	EOP2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY2	QW2	WBNE2	BSY2
	0x00000000					0	0	0	0	0	0	0		0	0	0	0													0	0	0	0
0x114	FLASH_CCR2	Res.	Res.	Res.	Res.	CLR_CRCEND2	CLR_DBECCERR2	CLR_SNECCERR2	CLR_RDSERR2	CLR_RDPERR2	CLR_OPERR2	CLR_INCERR2	Res.	CLR_STRBERR2	CLR_PGSERR2	CLR_WRPERR2	CLR_EOP2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000					0	0	0	0	0	0	0		0	0	0	0																
0x118	FLASH_OPTCR	SWAP_BANK	OPTCHANGEERRIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OBL_LAUNCH	OPTSTART	OPTLOCK	
	0x00000001	0	0																											0	0	1	
0x11C	FLASH_OPTSR_CUR	SWAP_BANK_OPT	OPTCHANGEERR	Res.	PERSO_OK	RSS2	RSS1	nRST_STBY_D2	nRST_STOP_D2	Res.	Res.	SECURITY	ST_RAM_SIZE [1:0]		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.	RDP[7:0]								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV [1:0]		Res.	OPT_BUSY	
	0x13C600F0	0	0		1	0	0	1	1	1	1	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1	1	1	0	0		0
0x120	FLASH_OPTSR_PRG	SWAP_BANK_OPT	Res.	Res.	PERSO_OK	RSS2	RSS1	nRST_STBY_D2	nRST_STOP_D2	Res.	Res.	SECURITY	ST_RAM_SIZE [1:0]		FZ_IWDG_SDBY	FZ_IWDG_STOP	Res.	RDP[7:0]								nRST_STBY_D1	nRST_STOP_D1	Res.	IWDG1_HW	BOR_LEV [1:0]		Res.	Res.	
	0x13C600F0	0			1	0	0	1	1	1	1	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1	1	1	0	0		
0x124	FLASH_OPTCCR	Res.	CLR_OPTCHANGEERR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000		0																														
0x128	FLASH_PRAR_CUR2	DMEP2	Res.	Res.	Res.	PROT_AREA_END2[11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START2[11:0]											
	0x80000000	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X

表 16. レジスタマップとリセット値の表（続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x12C	FLASH_PRAR_PRG2	DMEP2	Res.	Res.	Res.	PROT_AREA_END2[11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START2[11:0]												
	0x80000000	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x130	FLASH_SCAR_CUR2	DMES2	Res.	Res.	Res.	SEC_AREA_END2[11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START2[11:0]												
	0x80000000	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x134	FLASH_SCAR_PRG2	DMES2	Res.	Res.	Res.	SEC_AREA_END2[11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START2[11:0]												
	0x80000000	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x138	FLASH_WPSN_CUR2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn2[7:0]							
	0x000000FF																									X	X	X	X	X	X	X	X
0x13C	FLASH_WPSN_PRG2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPSn2[7:0]							
	0x000000FF																									X	X	X	X	X	X	X	X
0x150	FLASH_CRCCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BURST[1:0]		Res.	Res.	CLEAN_CRC	START_CRC	Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	ALL_BANK	Res.	Res.	Res.	Res.	CRC_SECT[2:0]		
	0x001C0000											0	x			0	0						0	0	0	0					0	0	0
0x154	FLASH_CRCSADD2R	CRC_START_ADDR[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x158	FLASH_CRCEADD2R	CRC_END_ADDR[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x15C	FLASH_CRCDATAR	CRC_DATA[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x160	FLASH_ECC_FA2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FAIL_ECC_ADDR2[14:0]														
	0x00000000																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4 セキュリティメモリ管理

4.1 概要

STM32H7x3 マイクロコントローラは、最新の STM32 シリーズ（STM32F4/F7 および STM32L0/L4）と類似している保護メカニズムの最初のセットを提供します。

- グローバル読み出しデバイス保護
- 書き込み保護機能
- 独自仕様コード読み出し保護（PCROP）

これらの保護メカニズムの概要については、*セクション 4.3：フラッシュ保護*を参照してください。

STM32H7x3 ではセキュアアクセスモードという新しい拡張保護モードも提供しており、これによってユーザ定義のセキュアサービス（セキュアファームウェア更新）の開発が可能になり、また安全な実行、およびコードとデータの両方の保護が保証されます。この新しいメカニズムの詳細については、*セクション 4.4：セキュアアクセスモード*、*セクション 4.5：ルートセキュアサービス（RSS）*、および*セクション 4.6：セキュアユーザソフトウェア*を参照してください。

セキュリティメモリ管理ユニットは、D1 ドメインに含まれています。

4.2 用語

ここでは、次の用語を使用します。

表 17. 推奨する用語の一覧

用語	説明
デバイスセキュリティレベル	
標準モード	ユーザフラッシュメモリ、オプションバイト、およびブートローダ領域へのアクセスを可能にするデバイスの状態。
セキュアアクセスモード	デバイスのすべてのメモリ領域へのアクセスを可能にするデバイスの状態。
メモリ領域	
システムメモリ	ST ROM コードを格納するために使用する、ST の予約済みメモリ領域。
ユーザフラッシュメモリ	ユーザコードとデータを格納するために使用する、フラッシュメモリ領域。
セキュアユーザメモリ／領域	この領域は、リセット後に 1 回アクセスされるように設定でき、この領域に格納されたコードが実行された後で、ユーザフラッシュメモリに格納されたファームウェアから隠すことができます。

表 17. 推奨する用語の一覧（続き）

用語	説明
SW サービス	
ブートローダ	通常の通信ポートからファームウェアをダウンロードすることができる、リセット時に実行される STMicroelectronics ソフトウェア。
ルートセキュアサービス（RSS）	暗号化されたファームウェア／モジュールの復号化と、セキュアおよび非セキュアユーザメモリへのインストールを可能にする STMicroelectronics ソフトウェア。
セキュアブートローダ	STMicroelectronics のブートローダと ST のルーツセキュアサービスから成る相補サービス。
セキュアユーザソフトウェア	リセット後に 1 回実行される、SFU とファームウェアの初期化を格納するために使用できるユーザソフトウェア。セキュアユーザソフトウェアは、セキュアユーザメモリに配置されています。
セキュアファームウェア更新（SFU）	インストールされたファームウェアをユーザフラッシュメモリからダウンロードして更新するために使用するユーザソフトウェア

4.3 フラッシュ保護

エンドユーザアプリケーションで、または独自仕様コード（第三者のライブラリ）の開発中に、3 つの保護メカニズムを使用できます。

- 読み出しデバイス保護（RDP）

  最終製品に適したグローバル保護メカニズムです。デバッグポート、RAM からのブート、またはブートローダを介した外部からのデバイスへのアクセスを防ぎます。

- 独自仕様コード読み出し保護（PCROP）

  このメカニズムは特定のフラッシュメモリ領域にセットすることができ、実行のみ可能です。PCROP は、機密性の高いソフトウェアのダンプを避けるために使用します。

- 書き込み保護機能

  この保護は、フラッシュセクタレベルにセットすることができます。コードやデータが予期せず消去されないよう保護します。

表 18 に、STM32H7x3 とほとんどの STM32 デバイスで使用可能な保護メカニズムの一覧を示します。STM32H7x3は、標準とセキュアアクセスモードの両方で使用可能です。

詳細については、*セクション 3：内蔵フラッシュメモリ（フラッシュ）*を参照してください。

表 18. フラッシュ保護メカニズム

フラッシュ保護	範囲	説明
RDP	グローバル （すべてのバンクのすべてのセクタ + バックアップ SRAM）	保護レベル 1 ： – デバッガが接続されている場合やブート設定がユーザフラッシュメモリと異なる場合は、フラッシュメモリやバックアップ SRAM に一切アクセス（読み出し、消去、およびプログラム）できなくなります。 – フラッシュ全体消去を実行して保護を解除することができます。 保護レベル 2 ： – デバッグアクセス（SW または JTAG）無効 – システムブートローダアクセスなし(1) – RAM ブート不可 – 保護レベル 2 は永続
書き込み保護機能	セクタレベル（128 KB）	不要なユーザフラッシュメモリセクタへの書き込み／変更に対する保護。
PCROP／ 実行のみ領域	領域レベル （256 バイトの粒度）	実行アクセスのみ可能です。 この保護は、別のソフトウェアによるデバッグアクセスやコードのコピーを防ぐため、機密コードでセットされています。

1. ルートセキュアサービスにセキュアアクセスモードでアクセスすることができます。*セクション 4.4: セキュアアクセスモード*を参照してください。

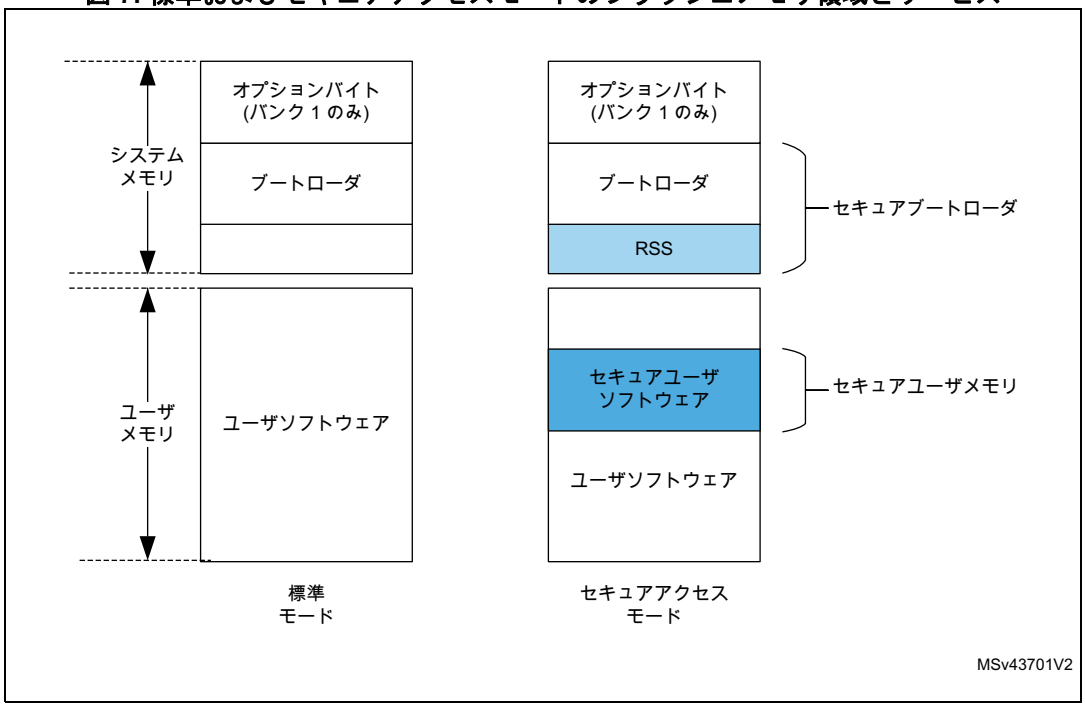
4.4 セキュアアクセスモード

一部の機密機能は、潜在的な悪意のあるソフトウェア攻撃から安全に実行する必要があります。セキュアファームウェア更新（SFU）ソフトウェアは、他のプロセスで取得されてはならない秘密データ（暗号化されたファームウェアまたは暗号キー）を扱っているため、高度な保護を必要とするコードの良い例です。

STM32H7x3 マイクロコントローラには、アクセスが制限されたセキュアメモリ領域があります。これにより、どのユーザアプリケーションよりも先に実行されるセキュアサービスを構築することができます。これらのセキュア領域と、これらに含まれるソフトウェアは、セキュアアクセスモードで新しいデバイスを設定する場合にのみアクセスすることができます。

*図 7*は、標準およびセキュアアクセスモードのフラッシュメモリ領域とサービスの概要を示します。

図 7. 標準およびセキュアアクセスモードのフラッシュメモリ領域とサービス



1. セキュアアクセスモードでのみアクセス可能な保護された領域を青で示します。

4.4.1 関連機能

セキュアアクセスモードはオプションバイトを介して設定できます。セットすると、次のアクセスが可能になります。

- セキュアブートローダへのアクセス。これには、ブートローダと、STMicroelectronics から提供される一部のセキュアサービスが組み込まれています（*セクション 4.5：ルートセキュアサービス（RSS）*を参照）。
- セキュアユーザメモリへのアクセス。これには、セキュアユーザコードやデータが組み込まれています。

各コアへのアクセス権限の概要については、*セクション 4.7：フラッシュ保護メカニズムの概要*を参照してください。

4.4.2 ブートステートマシン

セキュアアクセスモードでは、標準ブートローダがセキュアブートローダに置き換わります。RSS でのブートは、ブート設定（ブートピンおよびブートアドレス）から個別に強制的に行われます。これにより、ルートセキュアサービスおよびセキュアユーザソフトウェアは、デバイス（従来のブートローダを含む）のその他すべてのコードを優先します。

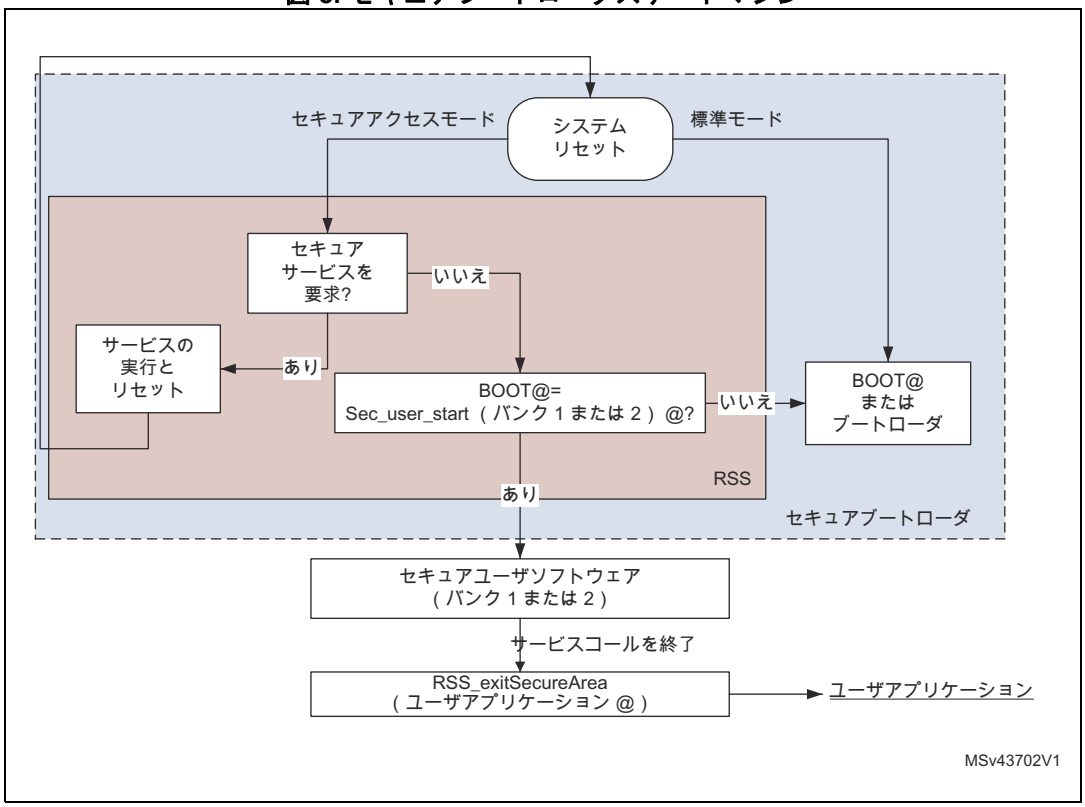
実行の優先順位

1. ルートセキュアサービスのシステムリセット後の優先順位は最も高くなります。コールされたサービスが実行され、直後に新しいシステムリセットがトリガされます。使用可能なセキュアサービスの詳細については、*セクション 4.5：ルートセキュアサービス（RSS）*を参照してください。
2. 要求しているサービスがない場合はセキュアユーザソフトウェアが実行されます。ただし、これは定義され、ブートアドレスが正しく設定されている場合に限ります。セキュアユーザソフトウェア設定については、*セクション 4.6：セキュアユーザソフトウェア*を参照してください。

3. それ以外の場合、従来のブートステートマシンがこれに続きます（ブートローダ、またはフラッシュメモリからのブート）。

*図 8*に STM32H7x3ブートステートマシンを示します。

図 8. セキュアブートローダステートマシン



4.4.3 セキュアアクセスモード設定

セキュアアクセスモードの有効化

デバイスのセキュアアクセスモードをアクティブにする方法に制限はありません。これは、FLASH_OPTSR_CUR レジスタの SECURITY オプションビットによって設定されます（*セクション 3.5.8：フラッシュオプションステータスレジスタ（現在値）（FLASH_OPTSR_CUR）*を参照）。

セキュアアクセスモードはシステムリセット後にアクティブになります。

セキュアアクセスモードの無効化

セキュアアクセスモードの無効化は、デバイスの保護されたコードがなくなった場合にのみ実行できるため、より慎重さが求められるタスクです。結果として、標準モードに戻るには、FLASH_OPTSR_CUR レジスタの SECURITY オプションビットをクリアする前に、セキュアユーザメモリおよび PCROP／実行のみ領域を削除しなければなりません。

保護された領域は、フラッシュ全体消去、バンク消去、または専用のセキュアサービスの呼び出し（PCROP 領域のみ）のいずれかを実行することで削除できます。

- PCROP 保護の解除については、*セクション：PCROP 領域（独自仕様コード読み出し保護、実行のみ領域）*および*RSS_resetAndDestroyPCROPArea*のセキュアサービスを参照してください。
- セキュア領域の削除方法の説明については、*セクション 3.3.12：保護メカニズム*を参照してください。

4.5 ルートセキュアサービス（RSS）

STMicroelectronics では、セキュア領域を設定するサービスを提供しています。これらのルートセキュアサービスは、セキュアアクセスモードでシステムリセット後、デバイスに格納されたその他のソフトウェアが実行される前に実行されます。

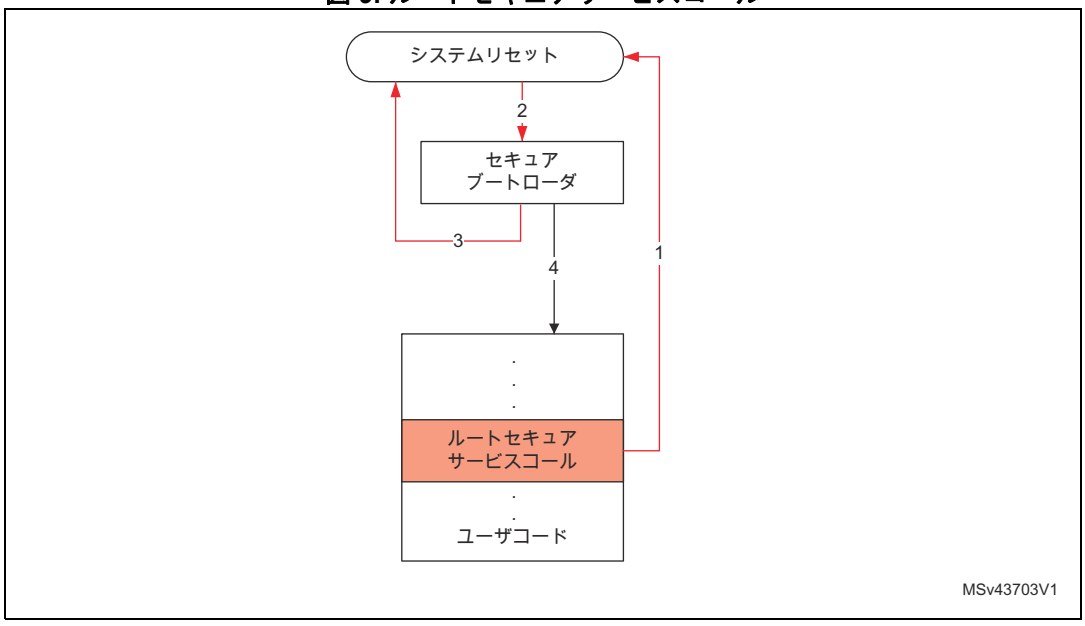
RSS ソフトウェアは、STM32H7x3 が標準モードで動作している場合はアクセス（読み出し、書き込み、実行、およびデバッグ）できません。

STM32H7x3 がセキュアアクセスモードで設定されている場合、RSS ソフトウェアはリセット時に 1 度だけ実行されます。RSS ソフトウェアの実行が完了すると、アクセスできなくなります（他のコードで直接 RSS にジャンプすることはできません）。リセットを実行すると、任意の RSS ルーチンが呼び出されます。

4.5.1 ルートセキュアサービスの呼び出し

ソフトウェアリセットをトリガするセキュアサービスがパブリック API を介して呼び出され（*図 9* のステップ 1 を参照）、これによりセキュアブートローダが使用可能な RSS コードを使用して実行されます（*図 9* のステップ 2 を参照）。実行されると、ほとんどのサービスでシステムリセットがトリガされます（*図 9* のステップ 3 を参照）。それ以上のサービスは不要になるため、セキュアブートローダはユーザソフトウェアにジャンプします（*図 9* のステップ 4 を参照）。

図 9. ルートセキュアサービスコール



4.5.2 ルートセキュアサービスの説明

以下のセキュアサービスは、PCROPの管理、セキュアユーザメモリ、およびセキュアモード設定で使用できます。

RSS_resetAndInitializeSecureAreas

プロトタイプ	void RSS_resetAndInitializeSecureAreas(RSS_SecureArea_t area)
引数	セキュアユーザ領域の開始アドレスと終了アドレス。1 つまたは 2 つのセキュアユーザ領域をセットできます。
説明	このサービスは、オプションバイトレジスタに格納された値に従って、セキュアユーザ領域の境界をセットします。 – SEC_AREA_START1 および SEC_AREA_END1 （バンク 1） – SEC_AREA_START2 および SEC_AREA_END2 （バンク 12） このサービスは、セキュア領域を初めてセットした場合にのみ使用できます。 サービス完了後にシステムリセットがトリガされます。

RSS_exitSecureArea

プロトタイプ	void RSS_exitSecureArea(unsigned int vectors)
引数	終了後のジャンプ先のアプリケーションベクタのアドレス。
説明	このサービスは、セキュアユーザソフトウェアから復帰し、ユーザのメインアプリケーションにジャンプする際に使用します。 このサービスによってトリガされるシステムリセットはありません。

RSS_resetAndDestroyPCROPArea

プロトタイプ	RSS_resetAndDestroyPCROPArea(RSS_FlashBank_et bank)
引数	対象のバンク番号
説明	このサービスは、PCROP 領域を上書きして PCROP 保護を解除します。 フラッシュ／バンク全体消去を回避します。 サービス完了後にシステムリセットがトリガされます。

4.6 セキュアユーザソフトウェア

セキュアユーザソフトウェアは、デバイスのパワーオン後、またはシステムリセット後に実行される信頼できるコード部分です。次のようなセキュアアプリケーションを構築することができます。

- コード署名または整合性チェック（ユーザセキュアブート）
- ソフトウェアライセンスチェック
- セキュアファームウェア更新

4.6.1 アクセス規則

セキュアユーザソフトウェアは、セキュアアクセスモードでのみアクセス可能で、セキュアユーザメモリに格納されています。このメモリは、設定可能な保護領域で、ユーザのメインメモリの一部です。

バンクごとに 1 つのユーザセキュア領域のみを設定できます。

ルートセキュアサービスの優先順位のみが高く、セキュアユーザソフトウェアの実行を優先できます（*セクション 4.4.2：ブートステートマシン*を参照）。

セキュアユーザソフトウェアの実行後、コードはユーザのメインアプリケーションにジャンプし、セキュアユーザ領域へのアクセスを防ぎます。これは、アプリケーションコードアドレスをパラメータとして指定した *RSS_exitSecureArea* のセキュアサービスを呼び出して実行されます。

アプリケーションコードでは、任意のセキュアユーザ領域にアクセスすると、フラッシュエラーがトリガされます。

4.6.2 セキュアユーザメモリ領域の設定

各バンクに設定可能なサイズのセキュア領域を 1 つセットできます。各領域のサイズは、512 バイトからフルバンクまで、256 バイトの単位でセットできます。

- バンク 1 のセキュア領域

  境界は、FLASH_SCAR_CUR1 の SEC_AREA_START1 および SEC_AREA_END1 オプションバイトを介して設定されます（*セクション 3.5.13：バンク 1 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR1）*を参照）。

- バンク 2 のセキュア領域

  境界は、FLASH_SCAR_CUR2 の SEC_AREA_START2 および SEC_AREA_END2 オプションバイトを介して設定されます（*セクション 3.5.30：バンク 2 のフラッシュセキュアアドレス（現在値）（FLASH_SCAR_CUR2）*を参照）。

注： ***セキュア領域の開始アドレスがセキュア領域の終了アドレスと等しい場合、フルバンクは安全に保護されているものとみなされます。***

アクティブな設定は、セキュアユーザ領域がデバイスで定義されているかいないかによって異なります。

- セキュアユーザ領域が存在していない場合（初回設定時）：

  専用のルートセキュアサービス *RSS_resetAndInitializeSecureAreas* を呼び出して、ST ブートステートマシンの空のアドレスを初期化します。同じ操作で両方のセキュア領域（バンクごとに 1 つ）を設定できます。

- セキュアユーザ領域がすでに存在している場合：

  この場合、セキュアユーザ領域のコードによって、独自のセキュアユーザ領域サイズを更新するか、別のバンクに新しく作成することができます。

  RSS_resetAndInitializeSecureAreas 機能を使用して、2 つ目のユーザセキュア領域を作成することはできない点に注意してください。

4.6.3 セキュアユーザメモリ領域の削除

セキュアユーザ領域は、フラッシュ全体消去またはバンク消去操作でのみ消去できます。

- フラッシュ全体消去は、RDP レベル 1 から RDP レベル 0 へのレベルの回帰によってトリガされます。
- バンク消去は、FLASH_CR1/2 レジスタの BER1/2 ビットをセットすることでトリガされます（*セクション 3.5.4：バンク 1 のフラッシュ制御レジスタ（FLASH_CR1）*および *セクション 3.5.25：バンク 2 のフラッシュ制御レジスタ（FLASH_CR2）*を参照）。

どちらの場合も、FLASH_PRAR_CUR1/2 レジスタの消去ポリシーを定義する DMES1/2 ビット（*セクション 3.5.11：バンク 1 のフラッシュ保護アドレス（現在値）(FLASH_PRAR_CUR1)* および *セクション 3.5.28：バンク 2 のフラッシュ保護アドレス（現在値）(FLASH_PRAR_CUR2)* を参照）は 1 にセットされ、また領域の開始アドレスは終了アドレスより大きくなります。

4.6.4 セキュアユーザソフトウェアの選択

現在のブートアドレスは、2 つのうちの 1 つのセキュアユーザソフトウェア（バンクごとに 1 つ）を選択します。2 つのブートアドレスを定義できるため、いくつかの使用例を考慮する必要があります。

FLASH_BOOT_CURR レジスタの BOOT_ADD0 および BOOT_ADD1 の間のブートアドレスの選択は、BOOT ピン（*セクション 2.5：ブート設定*を参照）で行うか、スワップバンクメカニズム（*セクション 3.3.13：フラッシュバンクスワッピング*を参照）で行います。

表 19 に、さまざまなブートアドレスのレジスタ設定と潜在的な関連シナリオを示します。

表 19. セキュアユーザソフトウェア選択の使用例(1)

BOOT_ADD0（ブートピン = 0）	BOOT_ADD1（ブートピン = 1）	使用例
SEC_AREA_START1	SEC_AREA_START2	BOOT ピンを使用して、2 つのセキュアユーザソフトウェアを切り替えます（今回とリカバリ、または今回と前回）。
SEC_AREA_START1	SEC_AREA_START1	バンクスワッピング機能を使用して、各バンクの同じオフセットで開始する 2 つのセキュアユーザソフトウェアを切り替えます（前回のセキュアブートから今回）。
SEC_AREA_START1	リカバリファームウェア	BOOT ピンを使用して、セキュアユーザソフトウェアと非セキュアリカバリブートを切り替えます。
SEC_AREA_START1	ブートローダ	BOOT ピンを（非セキュアモードで）ブートローダベースのリカバリに使用します。

1. バンク番号とブートアドレスのレジスタ番号は関連していません。BOOT_ADDx、またはバンク 1 とバンク 2 のどちらかにセットできます。

ブートアドレスが 2 つのセキュアユーザ領域と異なる場合、システムはセキュアユーザソフトウェアをバイパスして選択したアドレスに直接ジャンプします。

4.7 フラッシュ保護メカニズムの概要

図 10 および *表 20* に、セキュアアクセスモードと標準モードの両方について、異なるフラッシュメモリ領域のアクセス権限の一覧を示します。

図 10. フラッシュメモリ領域へのコアアクセス

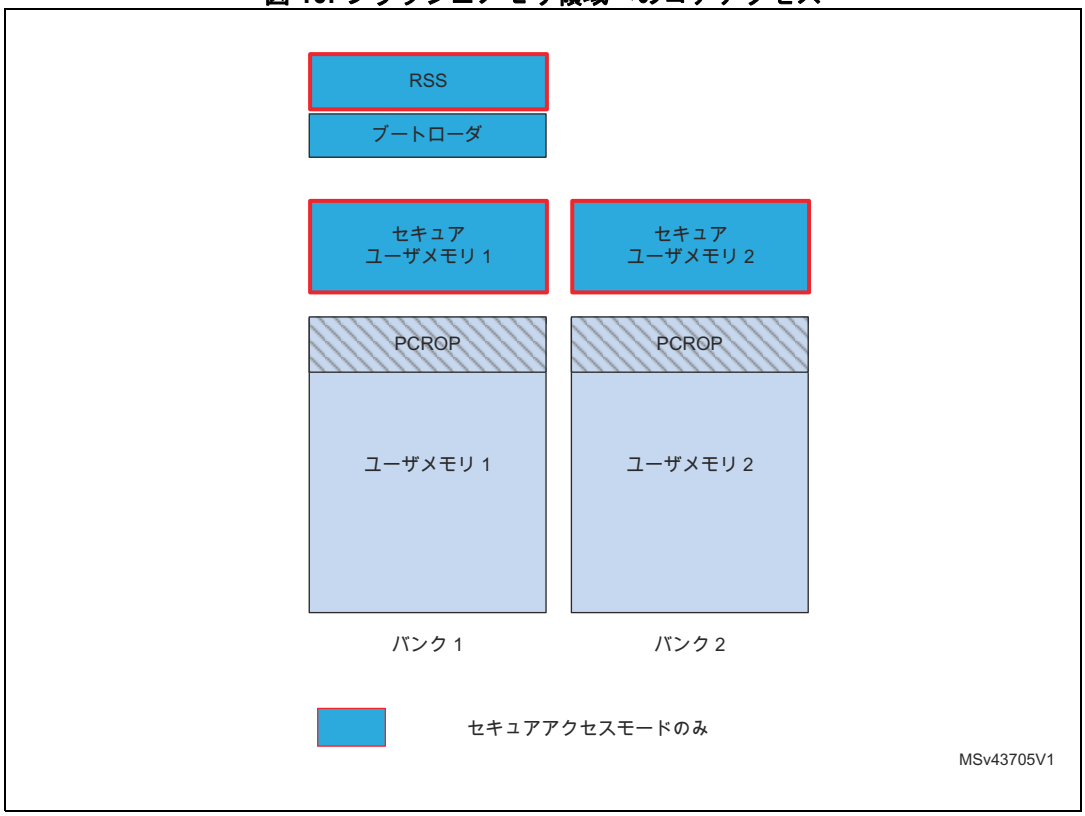


表 20. フラッシュ保護領域のアクセス権限の一覧

アクセスタイプ	SW 領域	セキュリティモード	アクセス
実行	PCROP	任意	✓
	セキュアユーザソフトウェア	セキュアアクセス	✓(1)
	ルートセキュアサービス	セキュアアクセス	✓(1)
読み出しアクセス	PCROP	任意	なし
	セキュアユーザソフトウェア	セキュアアクセス	✓(1)
	ルートセキュアサービス	セキュアアクセス	✓(1)
デバッグアクセス	PCROP	任意	なし
	セキュアユーザソフトウェア	セキュアアクセス	なし
	ルートセキュアサービス	セキュアアクセス	なし

1. リセット後にアクセス権限を許可（コードが完了するまでのみ）

参考資料

5 AXI インタコネクト

5.1 AXI の概要

AXI （Advanced eXtensible Interface）インタコネクトは ARM® CoreLink™ NIC-400 Network Interconnect に基づいています。インタコネクトには 6 つのイニシエータポートまたは ASIB（AMBA スレーブインタフェースブロック）があり、また 7 つのターゲットポートまたは AMIB （AMBA マスタインタフェースブロック）があります。ASIB は AXI スイッチマトリックスを通じて AMIB に接続されています。

各 ASIB は AXI バスまたは AHB （アドバンストハイパフォーマンスバス）上のスレーブです。同様に、各 AMIB は AXI または AHB バス上のマスタです。ASIB または AMIB が AHB に接続されている場合は、AHB と AXI プロトコルの間で変換されます。

AXI インタコネクトには、各 ASIB での QoS（サービス品質）レベルなどの特定のパラメータを設定するレジスタを含む GPV （Global Programmers View）が含まれます。

未割り当てアドレス空間へのアクセスはすべてデフォルトスレーブで処理され、リターン信号が生成されます。これにより、このようなトランザクションでも完結し、通信中のマスタおよび ASIB をブロックしません。

5.2 AXI インタコネクトの主な機能

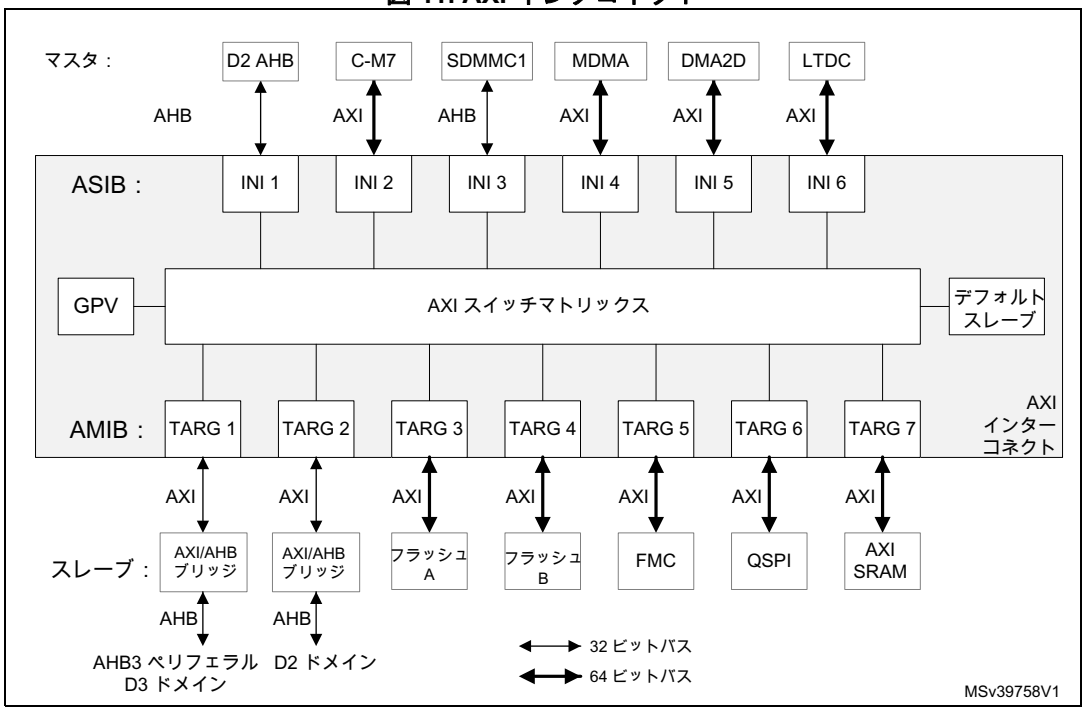
- 6 つの ASIB と 7 つの AMIB を持つ、D1 ドメインの 64 ビット AXI バスのスイッチマトリックス
- ASIB に組み込まれた AHB/AXI ブリッジの機能
- 複数の ASIB から複数の AMIB への同時接続性
- プログラム可能なトラフィック優先順位の管理（QoS - サービス品質）
- GPV を通じてソフトウェアでの設定が可能

5.3 AXI インタコネクトの機能説明

5.3.1 ブロック図

AXI インタコネクトを*図 11* に示します。

図 11. AXI インタコネクト



5.3.2 ASIB の設定

表 21 に ASIB の特性の一覧を示します。

表 21. ASIB の設定

ASIB	接続先のマスタ	プロトコル	バス幅	発行する R/W
INI 1	D2 ドメインからの AHB	AHB-Lite	32	1/4
INI 2	Cortex-M7	AXI4	64	7/32
INI 3	SDMMC1	AHB-Lite	32	1/4
INI 4	MDMA	AXI4	64	4/1
INI 5	DMA2D	AXI4	64	2/1
INI 6	LTDC	AXI4	64	1/1

5.3.3 AMIB の設定

表 22 に AMIB の特性の一覧を示します。

表 22. AMIB の設定

AMIB	接続先のスレーブ	プロトコル	バス幅	R／W／許容
TARG 1	ペリフェラル 3 および D3 AHB	AXI4(1)	32	1/1/1
TARG 2	D2 AHB	AXI4 (1)	32	1/1/1
TARG 3	フラッシュ A	AXI4	64	3/2/5
TARG 4	フラッシュ B	AXI4	64	3/2/5
TARG 5	FMC	AXI4	64	3/3/6
TARG 6	QUADSPI	AXI4	64	2/1/3
TARG 7	AXI SRAM	AXI3	64	2/2/2

1. AHB プロトコルへの変換は、AXI インタコネクトと接続先のスレーブとの間にある AXI/AHB ブリッジを通じて行われます。

5.3.4 サービス品質（QoS）

AXI スイッチマトリックスでは、2 つの ASIB が同じ AMIB に同時にアクセスしようとする場合に、優先順位に基づくアービトレーションを使用します。各 ASIB には QoS と呼ばれる 0 から 15 のプログラム可能な読み出しチャネルと書き込みチャネルの優先順位があり、この値が高いほど優先順位が高くなります。読み出しチャネルの QoS 値は *AXI インタコネクト - INI x 読み出し QoS レジスタ（AXI_INIx_READ_QOS）* でプログラムされ、書き込みチャネルは *AXI インタコネクト - INI x 書き込み QoS レジスタ（AXI_INIx_WRITE_QOS）* でプログラムされます。すべてのチャネルのデフォルトの QoS 値は 0 です（最低優先順位）。

同じ AMIB に 2 つのトランザクションが同時に着信した場合は、優先順位の高いトランザクションが優先順位の低いトランザクションより前に通過します。2 つのトランザクションの QoS 値が同じ場合は、最も長い時間使われていない（LRU）優先順位方式が採用されます。

QoS 値は、アプリケーションの遅延要件に従ってプログラムする必要があります。ASIB の優先順位を高くセットすることで、関連するバスマスタが開始したトランザクションの遅延を低減できます。これは、グラフィック処理（LTDC、DMA2D）などのリアルタイム制約タスクに役立ちます。マスタに高い優先順位を割り当てると、同じスレーブ（Cortex-M7 CPU など）に何度も頻繁にアクセスできるようになるため、他の優先順位の低いマスタがこのスレーブにアクセスするのをブロックする場合があります。

5.3.5 Global programmer's view（GPV）

GPV には AXI インタコネクトの設定レジスタが含まれます（*セクション 5.4* を参照）。これらのレジスタは Cortex-M7 CPU からのみアクセス可能です。

5.4 AXI インタコネクトレジスタ

5.4.1 AXI インタコネクト - ペリフェラル ID4 レジスタ（AXI_PERIPH_ID_4）

アドレスオフセット：0x1FD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：N/A

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM®

5.4.2 AXI インタコネクト - ペリフェラル ID0 レジスタ（AXI_PERIPH_ID_0）

アドレスオフセット：0x1FE0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：ペリフェラルパート番号のビット 0 から 7

0x00：パート番号 = 0x400

5.4.3 AXI インタコネクト - ペリフェラル ID1 レジスタ（AXI_PERIPH_ID_1）

アドレスオフセット：0x1FE4

リセット値：0x0000 00B4

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 識別のビット 0 から 3
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：ペリフェラルパート番号のビット 8 から 11
0x4：パート番号 = 0x400

5.4.4 AXI インタコネクト - ペリフェラル ID2 レジスタ（AXI_PERIPH_ID_2）

アドレスオフセット：0x1FE8

リセット値：0x0000 002B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 7:4 **REVISION[3:0]**：ペリフェラルリビジョン番号
0x2: r0p2

ビット 3 **JEDEC :** JEP106 コードフラグ
0x1：JEDEC 割り当てコード

ビット 2:0 **JEP106ID[6:4]**：JEP106 識別のビット 4 から 6
0x3：ARM® JEDEC コード

5.4.5 AXI インタコネクト - ペリフェラル ID3 レジスタ（AXI_PERIPH_ID_3）

アドレスオフセット：0x1FEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_AND[3:0]				CUST_MOD_NUM[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REV_AND[3:0]**：顧客バージョン

0：なし

ビット 3:0 **CUST_MOD_NUM[3:0]**：顧客による変更

0：なし

5.4.6 AXI インタコネクト - コンポーネント ID0 レジスタ（AXI_COMP_ID_0）

アドレスオフセット：0x1FF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE7:0]**：プリアンブルのビット 0 から 7

0xD：共通の ID 値

5.4.7 AXI インタコネクト - コンポーネント ID1 レジスタ（AXI_COMP_ID_1）

アドレスオフセット：0x1FF4

リセット値：0x0000 00F0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネントのクラス

0xF：汎用 IP コンポーネントのクラス

ビット 3:0 **PREAMBLE[11:8]**：プリアンブルのビット 8 から 11

0x0：共通の ID 値

5.4.8 AXI インタコネクト - コンポーネント ID2 レジスタ（AXI_COMP_ID_2）

アドレスオフセット：0x1FF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：プリアンブルのビット 12 から 19

0x05：共通の ID の値

5.4.9 AXI インタコネクト - コンポーネント ID3 レジスタ（AXI_COMP_ID_3）

アドレスオフセット：0x1FFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：プリアンブルのビット 20 から 27

0xB1：共通の ID の値

5.4.10 AXI インタコネクト - TARG x バスマトリックス発行機能レジスタ（AXI_TARGx_FN_MOD_ISS_BM）

アドレスオフセット：0x1008 + 0x1000 * x、ここで x = 1 から 7

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：ターゲットに対するスイッチマトリックスの書き込みトランザクション最大数の上書き

0：通常の発行可能なトランザクション最大数。

1：スイッチマトリックスの発行可能な書き込みトランザクションの最大数を 1 にセットします。

ビット 0 **READ_ISS_OVERRIDE**：ターゲットに対するスイッチマトリックスの読み出しトランザクション最大数の上書き

0：通常の発行可能なトランザクション最大数。

1：スイッチマトリックスの発行可能な読み出しトランザクションの最大数を 1 にセットします。

5.4.11 AXI インタコネクト - TARG x バスマトリックス機能 2 レジスタ（AXI_TARGx_FN_MOD2）

アドレスオフセット：0x1024 + 0x1000 * x、ここで x = 1、2、および 7

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
															rw

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BYPASS_MERGE**：出力データの幅に合わせて転送ビートの集積を無効化します。未整列のトランザクションは、入力データワードの境界に合わせて再整列されません。

0：通常動作

1：集積を無効化

5.4.12 AXI インタコネクト - TARG x ロングバースト機能変更レジスタ（AXI_TARGx_FN_MOD_LB）

アドレスオフセット：0x102C + 0x1000 * x、ここで x = 1 および 2

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
															rw

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FN_MOD_LB**：ロングバーストのバースト中断を制御します。

0：ASIB の出力時にロングバーストを生成することはできません。

1：ASIB の出力時にロングバーストを生成することができます。

5.4.13 AXI インタコネクト - TARG x 発行機能変更レジスタ（AXI_TARGx_FN_MOD）

アドレスオフセット：0x1108 + 0x1000 * x、ここで x = 1、2、および 7

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：発行可能なAMIB 書き込みトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

ビット 0 **READ_ISS_OVERRIDE**：発行可能なAMIB 読み出しトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

5.4.14 AXI インタコネクト - INI x 機能変更 2 レジスタ（AXI_INIx_FN_MOD2）

アドレスオフセット：0x41024 + 0x1000 * x、ここで x = 1 および 3

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BYPASS_MERGE**：プロトコルが必要としない限り、サイズ拡張によるトランザクションの変換を無効化します。

0：通常動作

1：許可されている場合、トランザクションは変換されないまま通過します。

5.4.15 AXI インタコネクト - INI x AHB 機能変更レジスタ（AXI_INIx_FN_MOD_AHB）

アドレスオフセット：0x41028 + 0x1000 * x、ここで x = 1 および 3

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WR_INC_OVERRIDE**: すべての AHB-Lite 読み出しトランザクションを一連のシングルビート AXI トランザクションに変換します。

0：上書きを無効化

1：上書きを有効化

ビット 0 **RD_INC_OVERRIDE**: すべての AHB-Lite 書き込みトランザクションを一連のシングルビート AXI トランザクションに変換します。また、各 AHB-Lite 書き込みビートは、AXI でバッファされた書き込み応答において承認されます。

0：上書きを無効化

1：上書きを有効化

5.4.16 AXI インタコネクト - INI x 読み出し QoS レジスタ（AXI_INIx_READ_QOS）

アドレスオフセット：0x41100 + 0x1000 * x、ここで x = 1 から 6

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **AR_QOS[3:0]**：読み出しチャネルの QoS 設定

0x0：最低優先順位

0xF：最高優先順位

5.4.17 AXI インタコネクト - INI x 書き込み QoS レジスタ（AXI_INIx_WRITE_QOS）

アドレスオフセット：0x41104 + 0x1000 * x、ここで x = 1 から 6

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **AW_QOS[3:0]**：書き込みチャネルの QoS 設定

0x0：最低優先順位

0xF：最高優先順位

5.4.18 AXI インタコネクト - INI x 発行機能変更レジスタ（AXI_INIx_FN_MOD）

アドレスオフセット：0x41108 + 0x1000 * x、ここで x = 1 から 6

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：発行可能なASIB 書き込みトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

ビット 0 **READ_ISS_OVERRIDE**：発行可能なASIB 読み出しトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

5.5 AXI インタコネクトレジスタマップ

表 23. AXI インタコネクトレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1FD0	AXI_PERIPH_ID_4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0x1FD4	AXI_PERIPH_ID_5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0x1FD8	AXI_PERIPH_ID_6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0x1FDC	AXI_PERIPH_ID_7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0x1FE0	AXI_PERIPH_ID_0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x1FE4	AXI_PERIPH_ID_1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	1	0	0
0x1FE8	AXI_PERIPH_ID_2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	0	1	0	1	1
0x1FEC	AXI_PERIPH_ID_3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_AND[3:0]				CUST_MOD_NUM [3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0x1FF0	AXI_COMP_ID_0																									PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0x1FF4	AXI_COMP_ID_1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	1	1	1	0	0	0	0

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1FF8	AXI_COMP_ID_2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0x1FFC	AXI_COMP_ID_3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1
0x2000 - 0x2004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2008	AXI_TARG1_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x200C - 0x2020	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2024	AXI_TARG1_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x2028	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x202C	AXI_TARG1_FN_MOD_LB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
	リセット値																																0
0x2030 - 0x2104	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2108	AXI_TARG1_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x210C - 0x3004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3008	AXI_TARG2_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x300C - 0x3020	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x3024	AXI_TARG2_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x3028	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x302C	AXI_TARG2_FN_MOD_LB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
	リセット値																																0
0x3030 - 0x3104	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x3108	AXI_TARG2_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x310C - 0x4004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x4008	AXI_TARG3_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x400C - 0x5004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5008	AXI_TARG4_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x500C - 0x6004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x6008	AXI_TARG5_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x600C - 0x7004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x7008	AXI_TARG6_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x700C - 0x8004	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x8008	AXI_TARG7_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x800C - 0x8020	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x8024	AXI_TARG7_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x8028 - 0x8104	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x8108	AXI_TARG7_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x810C - 0x42020	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x42024	AXI_INI1_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x42028	AXI_INI1_FN_MOD_AHB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
	リセット値																															0	0
0x4202C - 0x420FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x42100	AXI_INI1_READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x42104	AXI_INI1_WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x42108	AXI_INI1_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4210C - 0x430FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x43100	AXI_INI2_READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x43104	AXI_INI2_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x43108	AXI_INI2_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4310C - 0x44020	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44024	AXI_INI3_ FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x44028	AXI_INI3_ FN_MOD_AHB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
	リセット値																															0	0
0x4402C - 0x440FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44100	AXI_INI3_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x44104	AXI_INI3_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x44108	AXI_INI3_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4410C - 0x450FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 23. AXI インタコネクトレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x45100	AXI_INI4_ READ_QOS	Res.																												AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x45104	AXI_INI4_ WRITE_QOS	Res.																												AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x45108	AXI_INI4_ FN_MOD	Res.																												Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4510C - 0x460FC	予約済み	Res.																												Res.	Res.	Res.	Res.
0x46100	AXI_INI5_ READ_QOS	Res.																												AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x46104	AXI_INI5_ WRITE_QOS	Res.																												AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x46108	AXI_INI5_ FN_MOD	Res.																												Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4610C - 0x470FC	予約済み	Res.																												Res.	Res.	Res.	Res.
0x47100	AXI_INI6_ READ_QOS	Res.																												AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x47104	AXI_INI6_ WRITE_QOS	Res.																												AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x47108	AXI_INI6_ FN_MOD	Res.																												Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0

参考資料

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

参考資料

6 電源制御（PWR）

6.1 概要

電源制御のセクション（PWR）は、異なる電源ドメインに対する電源アーキテクチャと電源設定コントローラの概要を示します。

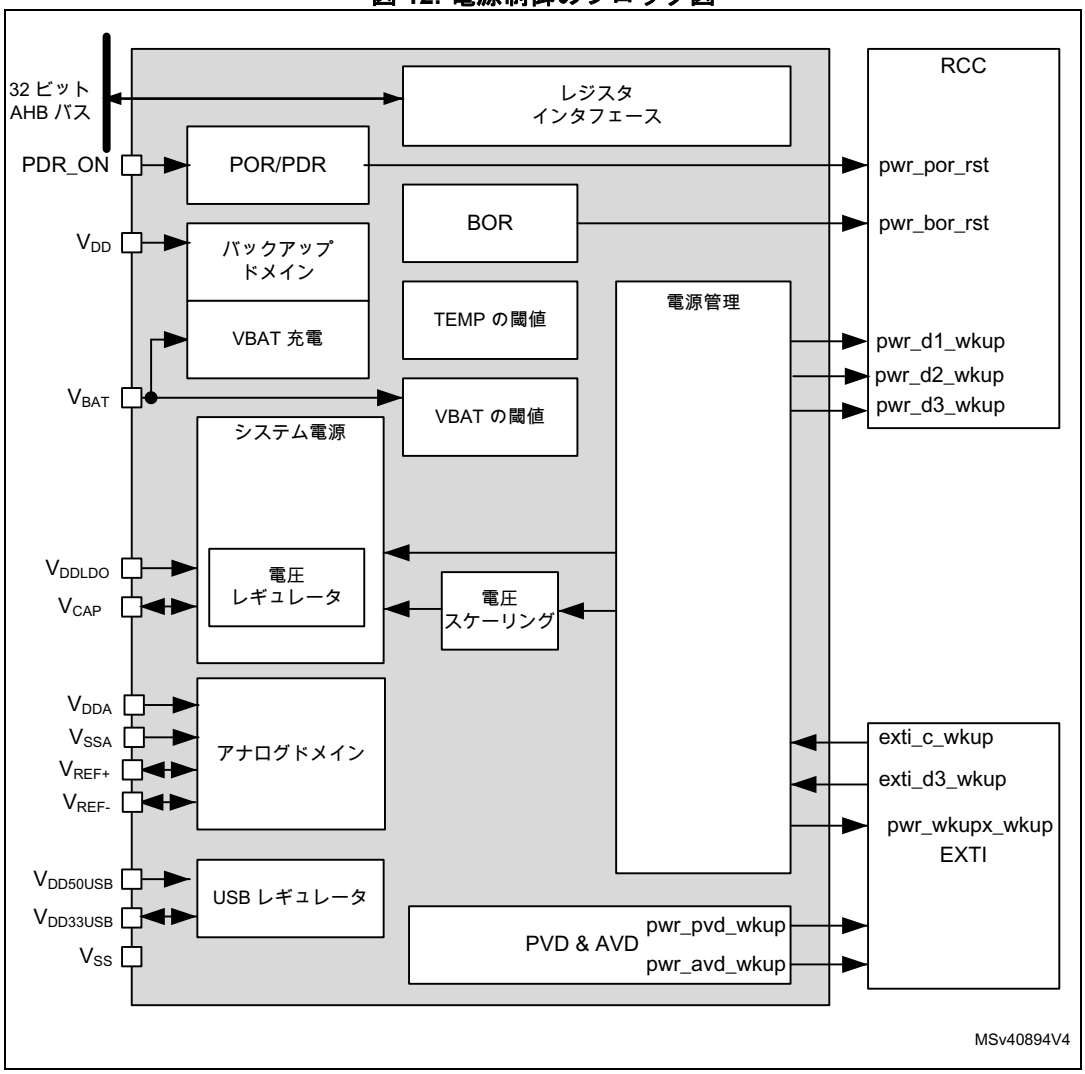
また、電源供給スーパーバイザの機能について説明し、V_{CORE} 電源ドメインが動作モード、選択された性能（クロック周波数）、および電圧スケーリングに応じてどのように設定されているかについて説明します。

6.2 PWR の主な機能

- 電源供給と電源ドメイン
  - コアドメイン（V_{CORE}）
  - V_{DD} ドメイン
  - バックアップドメイン（V_{SW}、V_{BKP}）
  - アナログドメイン（V_{DDA}）
- システム電源の電圧レギュレーション
  - 電圧レギュレータ（LDO）
- ペリフェラルの電源レギュレーション
  - USB レギュレータ
- 電源供給の監視
  - POR/PDR モニタ
  - BOR モニタ
  - PVD モニタ
  - AVD モニタ
  - V_{BAT} の閾値
  - 温度の閾値
- 電源管理
  - V_{BAT} バッテリの充電
  - 動作モード
  - 電圧スケーリングの制御
  - 低電力モード

6.3 PWR ブロック図

図 12. 電源制御のブロック図



参考資料

6.3.1 PWR ピンおよび内部信号

表 24 にパッケージのピンまたはボールに接続された PWR 入力および出力信号をリストし、*表 25* に PWR内部信号を示します。

表 24. パッケージのピンまたはボールに接続された PWR 入力／出力信号

ピン名	信号タイプ	説明
V_{DD}	電源入力	メイン I/O および V_{DD} ドメインの電源入力
V_{DDA}	電源入力	アナログペリフェラルの外部アナログ電源供給
V_{REF+}、V_{REF-}	電源入力／出力	ADC および DAC の外部基準電圧
V_{BAT}	電源入力	バックアップバッテリの電源入力
V_{DDLDO}	電源入力	電圧レギュレータの電源入力
V_{CAP}	電源入力／出力	デジタルコアドメインの電源
$V_{DD50USB}$	電源入力	USB レギュレータの電源入力
$V_{DD33USB}$	電源入力／出力	USB レギュレータの電源出力
V_{SS}	電源入力	メイングラウンド
AHB	デジタル入力／出力	AHB レジスタインタフェース
PDR_ON	デジタル入力	パワーダウンリセットイネーブル

表 25. PWR 内部入力／出力信号

信号名	信号タイプ	説明
pwr_pvd_wkup	デジタル出力	プログラム可能な電圧検出器の出力
pwr_avd_wkup	デジタル出力	アナログ電圧検出器の出力
pwr_wkupx_wkup	デジタル出力	CPU ウェイクアップ信号（x = 1～6）
pwr_por_rst	デジタル出力	パワーオンリセット
pwr_bor_rst	デジタル出力	ブラウンアウトリセット
exti_c_wkup	デジタル入力	CPU ウェイクアップリクエスト
exti_d3_wkup	デジタル入力	D3 ドメインウェイクアップリクエスト
pwr_d1_wkup	デジタル出力	D1 ドメインバスマトリックスクロックウェイクアップリクエスト
pwr_d2_wkup	デジタル出力	D2 ドメインバスマトリックスクロックウェイクアップリクエスト
pwr_d3_wkup	デジタル出力	D3 ドメインバスマトリックスクロックウェイクアップリクエスト

6.4 電源

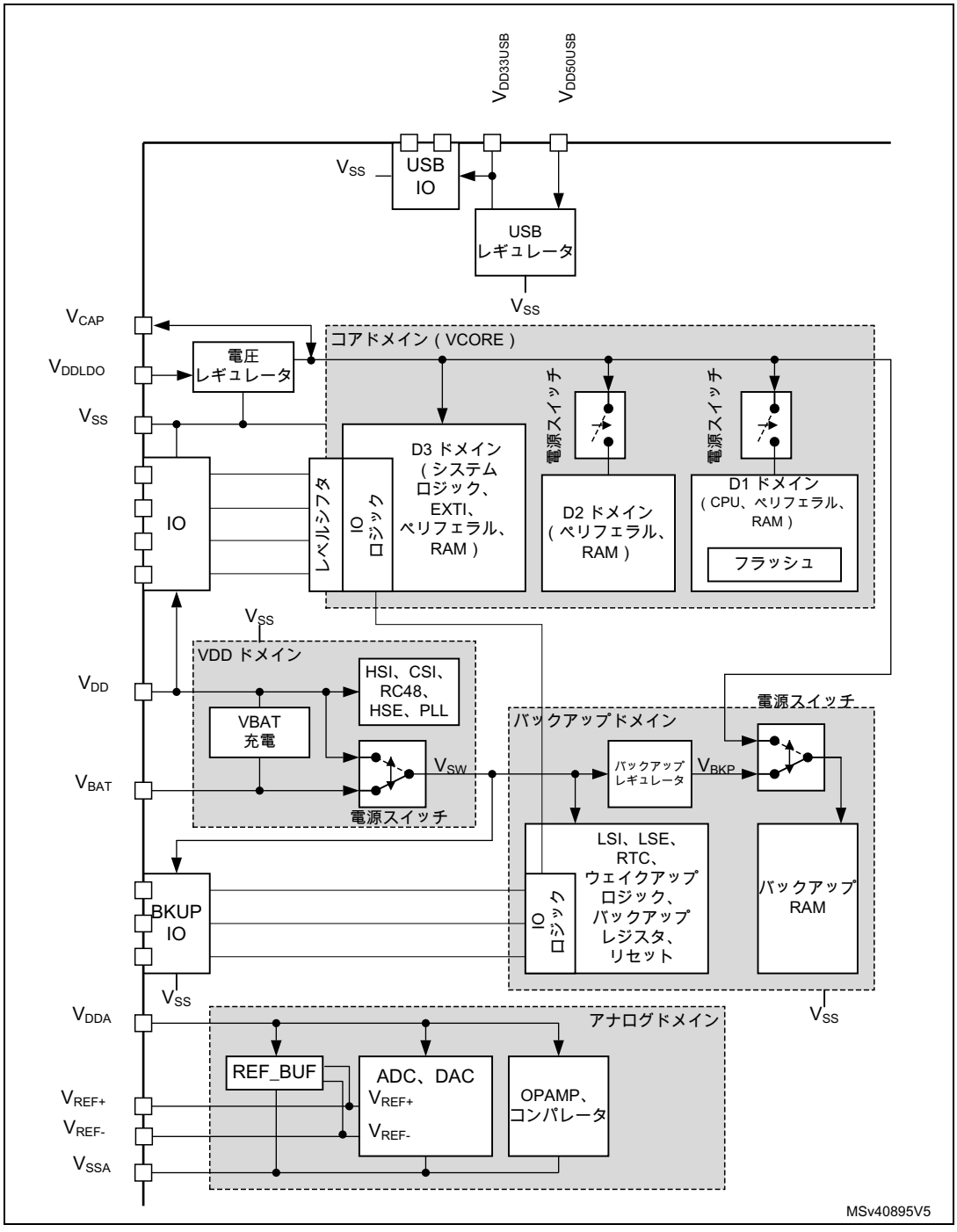
デバイスには、V_{DD} 電源に加え、V_{DDLDO}、V_{DDA}、V_{DDUSB}、および V_{CAP} 用に独立した電源が必要です。また、特定の機能（電圧レギュレータ、USB レギュレータ）用に制御された電源を提供することもできます。

- V_{DD} ：I/O およびリセット、電源管理、オシレータなどのシステムアナログブロック用の外部電源
- V_{BAT} ：V_{DD} が存在しない場合のバックアップドメイン用のオプション外部電源（V_{BAT} モード）
  バッテリを使用しない場合は、この電源を V_{DD} に接続してください。
- V_{DDLDO} ：電圧レギュレータ用の外部電源
- V_{CAP} ：デジタルコアドメインの電源
  この電源は、他のすべての電源から独立しています。
  - 電圧レギュレータを有効にすると、V_{CORE} は内部電圧レギュレータによって供給されます。
  - 電圧レギュレータを無効にすると、V_{CORE} は V_{CAP} ピンを通して外部電源によって供給されます。
- V_{DDA}：ADC、DAC、OPAMP、コンパレータ、および電圧基準バッファ用の外部アナログ電源
  この電源は、他のすべての電源から独立しています。
- V_{REF+}：ADC および DAC 用の外部基準電圧
  - 電圧基準バッファを有効にすると、V_{REF+} および V_{REF-} は内部電圧基準バッファによって供給されます。
  - 電圧基準バッファを無効にすると、V_{REF+} は独立した外部基準電源によって供給されます。
- V_{SSA} ：個別アナログおよび基準電圧グラウンド
- $V_{DD50USB}$ ：USB レギュレータ用の外部電源
- $V_{DD33USB}$：USB インタフェース用の USB レギュレータの電源出力
  - USB レギュレータを有効にすると、$V_{DD33USB}$ は内部 USB レギュレータによって供給されます。
  - USB レギュレータを無効にすると、$V_{DD33USB}$ は独立した外部電源入力によって供給されます。
- V_{SS}：すべての電源とアナログレギュレータの共通グラウンド

注： ***動作電源範囲に応じて、一部のペリフェラルは機能および性能が限定されることがあります。詳細については、デバイスのデータシートの「一般動作条件」のセクションを参照してください。***

参考資料

図 13. 電源の概要



電圧レギュレータを設定することで、V_{CORE} コアドメインと外部電源では *図 14* に示す電源設定をサポートします。

図 14. システム電源設定

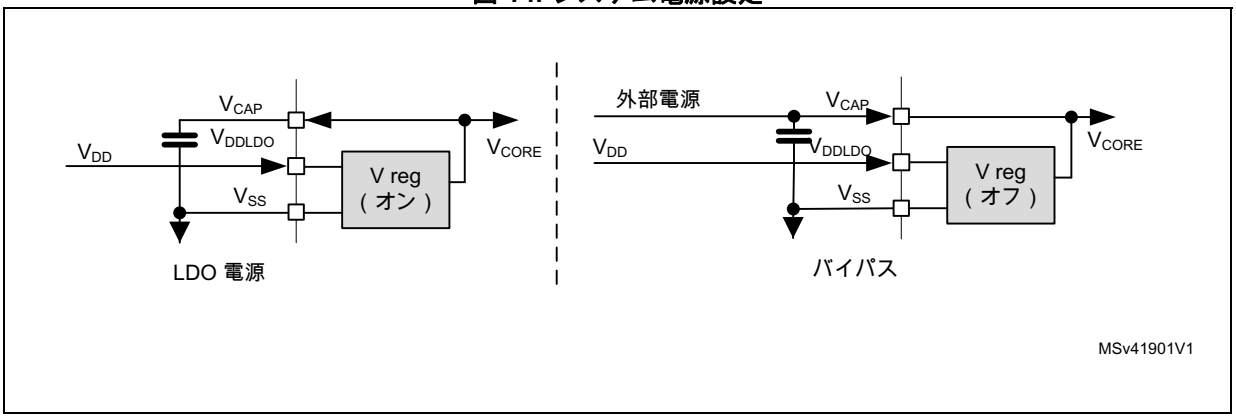


表 26 には、*PWR 制御レジスタ 3（PWR_CR3）* レジスタの LDOEN および BYPASS ビットによって異なる電源設定が制御されていることが示されています。

表 26. 電源設定の制御

電源構成	LDOEN	BYPASS	説明
デフォルト設定	1	0	– V_{CORE} 電源ドメインは VOS に従って LDO から供給されます。
LDO 電源	1	0	– V_{CORE} 電源ドメインは VOS に従って LDO から供給されます。 – LDO 電源モード（メイン、LP、オフ）はシステムの低電力モードに従います。
LDO バイパス	0	1	– V_{CORE}は外部ソースから供給されます。 – LDO はバイパスされ、電圧監視は有効のままです。
不正	0	0	– 不正な組み合わせです。デフォルトの設定が保持されます（書き込みデータは無視されます）。
	1	1	

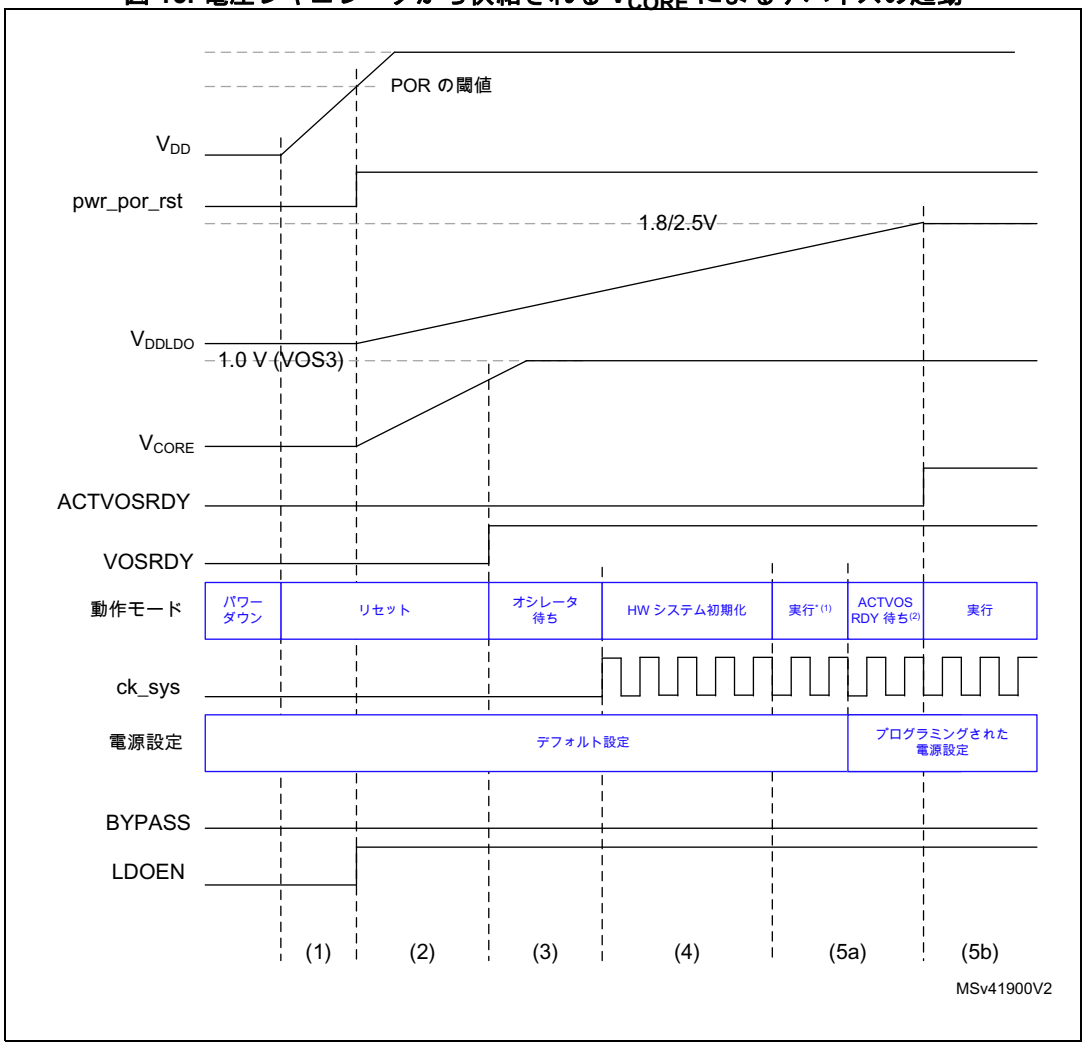
6.4.1 システム電源の起動

様々な電源設定での電源投入後のシステム起動シーケンスは次のとおりです（LDO 電源については *図 15* を参照）。

1. システムをパワーオンすると、POR では V_{DD} 電源を監視します。V_{DD} が POR 閾値レベルを上回ると、デフォルトの電源設定で電圧レギュレータが有効になります。
   – 電圧コンバータの出力レベルは*PWR D3 ドメイン制御レジスタ (PWR_D3CR)* で設定した VOS3 レベルに合わせて 1.0 V にセットされます。
2. V_{CORE} が正しくなければ、システムはリセットモードのまま保持されます。
3. 一度 V_{CORE} が正しくセットされると、システムはリセットモードから解除され、HSI オシレータが有効になります。
4. オシレータが安定すると、システムは初期化されます。フラッシュメモリおよびオプションバイトがロードされ、CPU は制限された RUN モード（RUN*）で起動します。
5. 次に、ソフトウェアで*PWR 制御レジスタ 3 (PWR_CR3)* の電源設定プログラムを含む、システムの初期化を行うことができます。電源設定が完了すると、*PWR 制御ステータスレジスタ 1 (PWR_CSR1)* の ACTVOSRDY ビットで電圧レベルが有効であるか確認されます。
   a) 電圧レベルが無効であることを ACTVOSRDY が示している場合、システムは RUN* モードで実行され、RAM への書き込みアクセスは許可されず、また VOS を変更することはできません。
   b) 電圧レベルが有効であることを ACTVOSRDY が示すと、システムは通常の RUN モードで実行され、RAM への書き込みアクセスが許可され、また VOS を変更できるようになります。

ソフトウェアで*PWR 制御レジスタ 3 (PWR_CR3)* の電源設定をプログラムする必要があります。

図 15. 電圧レギュレータから供給される V_{CORE} によるデバイスの起動



1. RUN* モードでは、RAM への書き込み操作はできません。
2. RAM への書き込み操作と VOS の変更は、ACTVOSRDY が有効な場合にのみ可能です。

STANDBY モードを終了すると、*PWR 制御レジスタ 3（PWR_CR3）*のレジスタの内容が保持されるため、システムは電源設定を把握しています。ただしソフトウェアは、V_{CORE} 電圧レベルが有効であることを示すために *PWR 制御ステータスレジスタ 1（PWR_CSR1）*の ACTVOSRDY ビットがセットされるのを待ってから、RAM への書き込みアクセスを実行したり、VOS を変更したりする必要があります。

6.4.2 コアドメイン

V_{CORE} コアドメインの電源は、電圧レギュレータまたは外部電源（V_{CAP}）から供給されます。V_{CORE} は、バックアップドメインと STANDBY 回路以外のすべてのデジタル回路に電力を供給しています。V_{CORE} ドメインは 3 のセクションに分割されます。

- D1 ドメイン：CPU （Cortex®-M7）、フラッシュメモリ、およびペリフェラルを含む
- D2 ドメイン：ペリフェラルを含む
- D3 ドメイン：システムコントロール、I/O ロジック、および低電力ペリフェラルを含む

システムリセットが発生すると、電圧レギュレータが有効になり、V_{CORE} が供給されます。これにより、システムは任意の電源設定で起動することができます（*図 14* を参照）。

システムリセット後、ソフトウェアは *PWR D3 ドメイン制御レジスタ（PWR_D3CR）* の VOS や RCC ck_sys 周波数を変更する前に、使用した電源設定を *PWR 制御レジスタ 3（PWR_CR3）* レジスタで設定する必要があります。様々なシステム電源設定の制御については、*表 26* を参照してください。

電圧レギュレータ

内蔵電圧レギュレータ（LDO）には、V_{CAP} ピンに外付けコンデンサを接続する必要があります。

電圧レギュレータには、メイン（MR）、低電力（LP）、またはオフの 3 つの異なる動作モードがあります。これらのモードは、システムの動作モード（RUN、STOP、および STANDBY）に応じて使用されます。

- RUN モード

  LDO レギュレータはメインモードで、V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。レギュレータの出力電圧は、ソフトウェアによって、*PWR D3 ドメイン制御レジスタ（PWR_D3CR）* の VOS ビットを使用して設定された異なる電圧レベル（VOS1、VOS2、および VOS3）にスケーリングできます。VOS 電圧スケーリングによって、システムが最大周波数より低速のクロックで動作しているときの消費電力を最適化することができます。デフォルトで、システムリセット後にVOS3 が選択されています。VOS は、システムに必要な性能に適応するように動作中に変更することができます。

- STOP モード

  電圧レギュレータは V_{CORE} ドメインに電源を供給して、レジスタと内部メモリの内容を保持します。

  レギュレータは、STOP モードからの高速復帰を可能にするためにメインモードのまま保持するか、LP モードにセットし、V_{CORE} 供給レベルを抑えて STOP からの復帰の遅延を延長することができます。レギュレータモードは、*PWR 制御レジスタ 1（PWR_CR1）* の SVOS および LPDS ビットによって選択されます。SVOS3 電圧スケーリングを選択するとメインモードと LP モードを使用でき、SVOS4 および SVOS5 スケーリングを選択すると LP モードのみを使用できます。SVOS4 および SVOS5 スケーリングの電圧レベルは低いため、STOP モードの消費電力をさらに低減できます。

- STANDBY モード

  電圧レギュレータはオフになり、V_{CORE} ドメインはパワーダウンします。STANDBY 回路とバックアップドメインを除き、レジスタとメモリの内容は失われます。

注： ***詳細については、データシートの電圧レギュレータのセクションを参照してください。***

6.4.3 PWR 外部電源

外部電源から V_{CORE} が供給されている場合、システムの動作モード（RUN、STOP、および STANDBY）に応じて、異なる動作モードを使用できます。

- RUN モード

  外部電源は V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。外部電源の出力電圧は、異なる電圧レベル（VOS1、VOS2、および VOS3）を使用してスケーリングできます。外部から印加された電圧レベルは、PWR_D3CR レジスタの VOS ビットに反映されます。RAM には、外部から印加された電圧レベルが VOS 設定と一致するときに、書き込み操作を行う場合にのみアクセスできます。

- STOP モード

  外部電源は V_{CORE} ドメインに電源を供給して、レジスタと内部メモリの内容を保持します。レギュレータはより低い V_{CORE} 供給レベルを選択して、STOP モードの消費電力を低減できます。

- STANDBY モード

  外部電源をオフにし、V_{CORE} ドメインをパワーダウンします。STANDBY 回路とバックアップドメインを除き、レジスタとメモリの内容は失われます。STANDBY モードから復帰する際は、外部電源をオンにする必要があります。

6.4.4 バックアップドメイン

V_{DD} がオフになった場合に、バックアップドメイン（RTC、バックアップレジスタ、およびバックアップ RAM）の内容を保持するために、V_{BAT} ピンをバッテリやその他の電源から供給されるオプションの予備電源に接続することができます。

V_{BAT} 電源への切り替えは、V_{DD} 供給を監視するリセットブロックに組み込まれているパワーダウンリセット回路によって制御されます。

警告： **$t_{RSTTEMPO}$（V_{DD} 起動時の過渡期間）中や PDR の検出後、V_{BAT} と V_{DD} の間の電源スイッチは V_{BAT} に接続されたままになります。**
起動フェーズ中、V_{DD} が $t_{RSTTEMPO}$ 以内に規定値に達し（$t_{RSTTEMPO}$ の値については、データシートを参照）、かつ $V_{DD} > V_{BAT} + 0.6$ V である場合、電流は V_{DD} と電源スイッチ（V_{BAT}）の間に接続された内部ダイオードを通して V_{BAT} に注入されます。
V_{BAT} ピンに接続された電源/バッテリがこの電流注入に対応できない場合は、この電源と V_{BAT} ピンの間に外部低電圧降下ダイオードを接続することを強く推奨します。

V_{DD} 電源が存在する場合、バックアップドメインは V_{DD} から供給を受けます。これにより、V_{BAT} 電源のバッテリ寿命を延ばすことができます。

外部バッテリを使用しないアプリケーションでは、100 nF の外部セラミックコンデンサを介して V_{BAT} を V_{DD} に外部接続することを推奨します。

V_{DD} 電源が存在し、これが PDR 閾値よりも高い場合、バックアップドメインは V_{DD} から供給され、また次の機能が使用できます。

- PC14 および PC15はGPIO または LSE ピンのいずれかとして使用可能
- PC13はGPIO、または RTC で設定されたと仮定し、RTC_AF1 または RTC_TAMP1 ピンとして使用可能
- PI8およびPC1はGPIO、またはRTC で設定されている場合、タンパピン（RTC_TAMP2 および RTC_TAMP3）として使用可能

注： ***スイッチに流すことができる電流は限られているので、GPIO の PC1、PC13 から PC15、および PI8 までの使用には制限があります。出力として使用できるのは同時に 1 つの I/O のみで、最大負荷 30 pF で最大速度 2 MHz に制限する必要があります。これらの I/O を電流ソースとして使用することはできません（たとえば、LED を駆動するなど）。***

V_{BAT} モードで、V_{DD} 電源が存在せず、V_{BAT} に電源が存在する場合、バックアップドメインは V_{BAT} から供給され、また次の機能が使用できます。

- PC14 および PC15はLSE ピンとしてのみ使用可能
- PC13はRTC で設定されたと仮定し、RTC_AF1 または RTC_TAMP1 ピンとして使用可能
- PI8およびPC1はGPIO、またはRTC で設定されている場合、タンパピン（RTC_TAMP2 および RTC_TAMP3）として使用可能

バックアップドメインへのアクセス

リセット後、バックアップドメイン（RTC レジスタおよび RTC バックアップレジスタ）は、予期しない書き込みアクセスから保護されます。バックアップドメインへのアクセスを有効にするには、*PWR 制御レジスタ 1（PWR_CR1）* の DBP ビットをセットします。

RTC とバックアップ RAM アクセスの詳細については、*セクション 8：リセットおよびクロック制御（RCC）* を参照してください。

バックアップ RAM

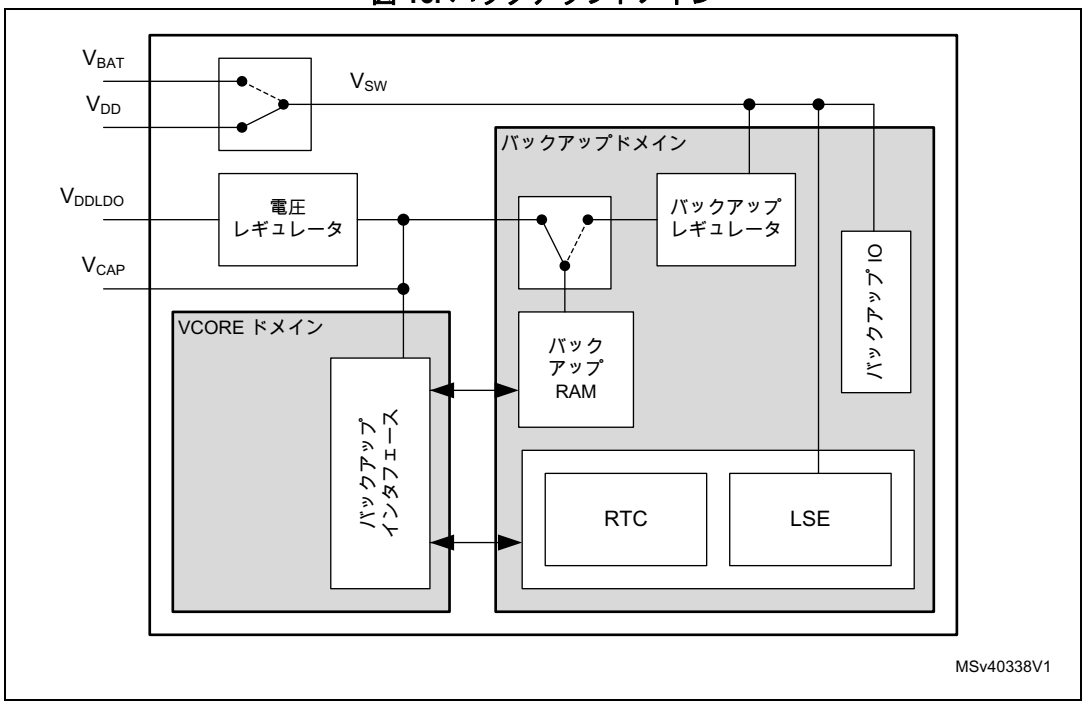
バックアップドメインには、32 ビット、16 ビット、または 8 ビットデータモードでアクセス可能な 4 KB のバックアップ RAM があります。バックアップ RAM は、バックアップドメインのバックアップレギュレータから供給されます。バックアップレギュレータが *PWR 制御レジスタ 2（PWR_CR2）* の BREN ビットによって有効化されると、STANDBY モードや V_{BAT} モードでもバックアップ RAM の内容は保持されます（V_{BAT} が常に存在している場合は、内部 EEPROM とみなすことができます）。

このバックアップレギュレータは、アプリケーションが STANDBY モードまたは V_{BAT} モードでバックアップ RAM の機能を必要とするか否かに応じてオン／オフを切り替えることができます。

バックアップ RAM はタンパイベントによって一括消去されることはありませんが、暗号の秘密鍵などの機密データへのアクセスを防ぐため、読み出し保護されています。タンパイベント後に再度バックアップ RAM へのアクセスを得るために、メモリ領域を最初に消去する必要があります。バックアップ RAM は、次の方法で消去することができます。

- 保護レベル 1から 0 への変更が要求された際に、フラッシュインタフェースを介して消去します（フラッシュプログラミングマニュアルの読み出し保護（RDP）の説明を参照）。
- タンパイベント後、データを 0 としてバックアップ RAM へのダミーの書き込みを実行して消去します。

図 16. バックアップドメイン



6.4.5 V_{BAT} バッテリの充電

V_{DD} が存在する場合、内部抵抗を通して V_{BAT} に接続された外部バッテリを充電できます。

V_{BAT} 充電は、*PWR 制御レジスタ 3（PWR_CR3）* の VBRS ビット値に応じて、5 kΩ レジスタまたは 1.5 kΩ レジスタを通して行われます。

バッテリの充電は、*PWR 制御レジスタ 3（PWR_CR3）* の VBE ビットをセットすると有効になります。V_{BAT} モードで自動的に無効になります。

6.4.6 アナログ電源

独立した V_{DDA} アナログ電源

アナログ電源ドメインは、専用の V_{DDA} および V_{SSA} パッドにより電源供給されます。これにより、電源から PCB のノイズをフィルタし、シールドすることができ、ADC および DAC 変換の精度を向上することができます。

- アナログ電源電圧入力は、独立した V_{DDA} ピンから供給されます。
- 電源のグランドは、独立した V_{SSA}ピンとして用意されています。

アナログ基準電圧 V_{REF+}/V_{REF-}

より高い精度の低電圧信号を得るため、ADC および DAC には V_{REF+} ピンに独立した基準電圧があります。独立した外部基準電圧を V_{REF+} に接続できます。

V_{REF+} は、V_{SSA} に接続された低い電圧基準（V_{REF-}）からのフルスケール値に相当する最高電圧を制御します。

VREFBUF 制御およびステータスレジスタの ENVR ビットによって有効化された場合（*セクション 27：電圧基準バッファ（VREFBUF）* を参照）、V_{REF+} は内部電圧基準バッファから供給されます。

内部電圧基準バッファは、V_{REF+}/V_{REF-} ピンから外部コンポーネントに基準電圧を供給することもできます。

内部電圧基準バッファが ENVR によって無効化されると、V_{REF+} は独立した外部の基準電源電圧によって供給されます。

6.4.7 USB レギュレータ

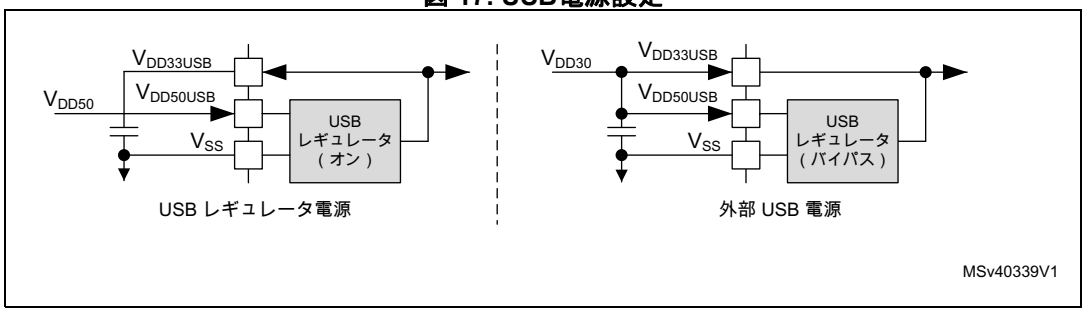
USB トランシーバは、統合された USB レギュレータまたは外部 USB 電源のどちらかから供給される専用の $V_{DD33USB}$ 電源から供給されます。

PWR 制御レジスタ 3（PWR_CR3） の USBREGEN ビットによって有効化された場合、$V_{DD33USB}$ は USB レギュレータから供給されます。$V_{DD33USB}$ を使用する前に、*PWR 制御レジスタ 3（PWR_CR3）* の USB33RDY ビットを監視して使用可能であることを確認します。$V_{DD33USB}$ 供給レベル検出器は、PWR_CR3 レジスタの USB33DEN ビットを使用して有効化します。

USBREGEN ビットを使用して USB レギュレータを無効化した場合は、外部電源から $V_{DD33USB}$ を供給することができます。この場合、$V_{DD33USB}$ と $V_{DD50USB}$ を接続します。

USB レギュレータの詳細については、*セクション 57：USB On-The-Go ハイスピード（OTG_HS）* を参照してください。

図 17. USB電源設定



6.5 電源供給の監視

電源供給レベルの監視は、以下の電源で使用きます。

- V_{DD} ：POR/PDR（*セクション 6.5.1* を参照）、BOR（*セクション 6.5.2* を参照）、および PVD モニタ（*セクション 6.5.3* を参照）を介して
- V_{DDA} ：AVD モニタ（*セクション 6.5.4* を参照）を介して
- V_{BAT} ：VBAT 閾値（*セクション 6.5.5* を参照）を介して
- V_{SW}：レベルが OK でない場合に V_{SW} ドメインをリセットモードで保持するrst_vsw を介して
- V_{BKP}：*PWR 制御レジスタ 2（PWR_CR2）* の BRRDY ビットを介して
- $V_{DD33USB}$：*PWR 制御レジスタ 3（PWR_CR3）* の USBRDY ビットを介して

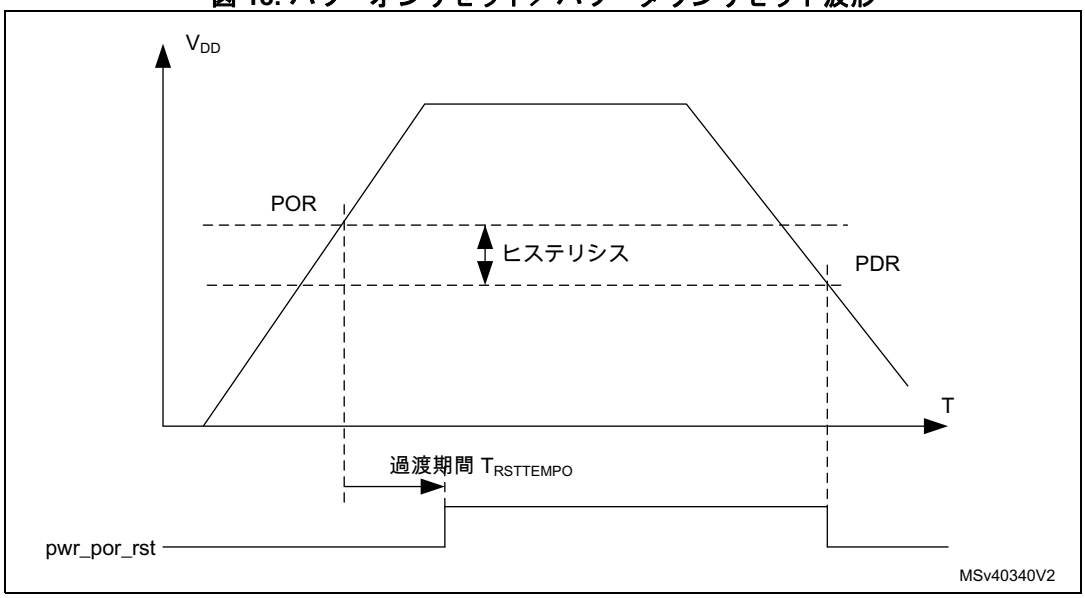
6.5.1 パワーオンリセット（POR）／パワーダウンリセット（PDR）

このシステムには、適切な起動操作を確実に実行する POR/PDR 統合回路が搭載されています。

システムは、V_{DD} が規定の V_{POR} の閾値を下回っている場合は、外部のリセット回路を必要とせずに、リセットモードを維持します。V_{DD} 供給レベルが V_{POR} の閾値を上回ると、システムはリセットモードから解除されます（*図 18* を参照）。パワーオンリセット／パワーダウンリセットの閾値の詳細については、データシートの電気的特性のセクションを参照してください。

PDR は、デバイスの PDR_ON 入力ピンで有効／無効にすることができます。

図 18. パワーオンリセット／パワーダウンリセット波形



1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.2 ブラウンアウトリセット（BOR）

パワーオン時、ブラウンアウトリセット（BOR）によってV_{DD} 電源電圧が規定の V_{BOR} の閾値に達するまで、システムはリセット状態に保持されます。

V_{BOR} の閾値は、システムオプションバイトを介して設定されます。デフォルトでは、BOR はオフです。次のプログラム可能な V_{BOR} の閾値を選択できます。

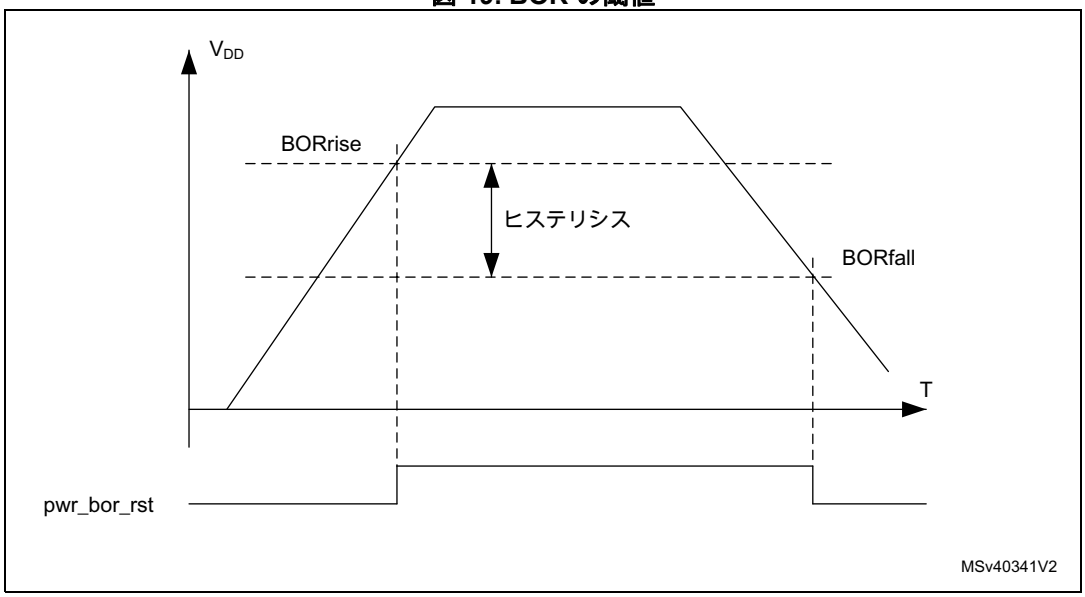
- BOR オフ（V_{BOR0}）
- BOR レベル 1（V_{BOR1}）
- BOR レベル 2（V_{BOR2}）
- BOR レベル 3（V_{BOR3}）

ブラウンアウトリセットの閾値の詳細については、製品データシートの「電気的特性」セクションを参照してください。

BOR を有効にし、V_{DD} 電源電圧が選択された V_{BOR} の閾値を下回ると、システムリセットが生成されます。

BOR は、システムオプションバイトをプログラミングすることによって、無効にすることができます。BOR 機能を無効にするには、システムオプションバイトのプログラミングシーケンスを開始するために、V_{DD} を V_{BOR0} よりも高く設定する必要があります。パワーダウンは PDR によって監視されます（*セクション 6.5.1* を参照）。

図 19. BOR の閾値



1. 閾値とヒステリシス値については、データシートを参照してください。

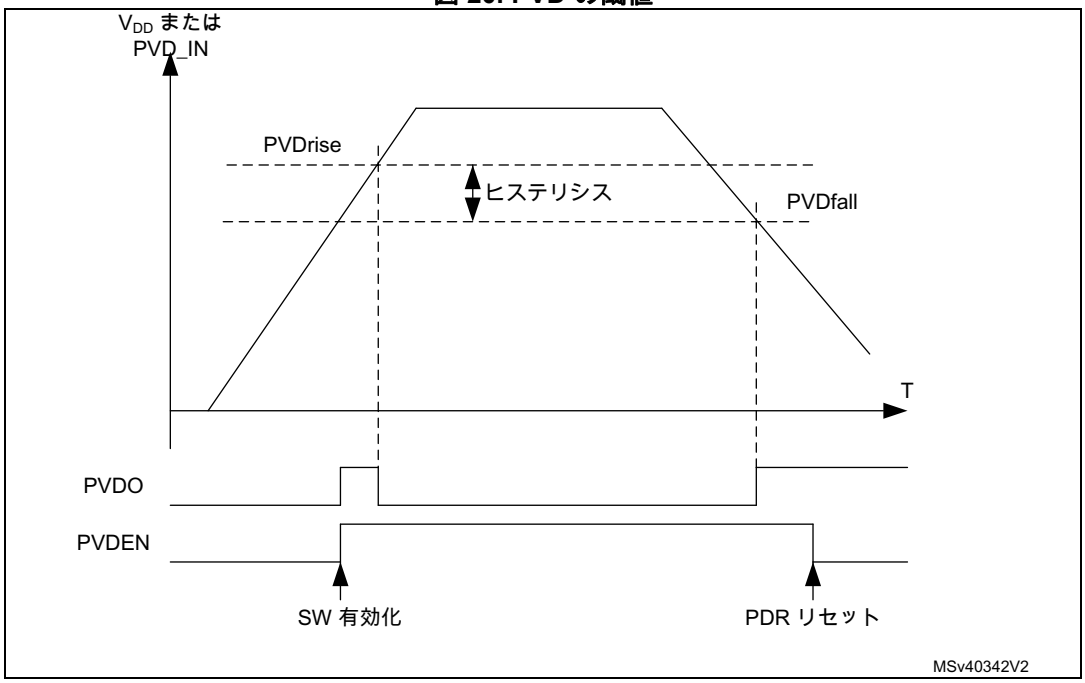
6.5.3 プログラム可能な電圧検出器（PVD）

PVD を使用して、*PWR 制御レジスタ 1（PWR_CR1）* の PLS[2:0] ビットで選択した閾値と比較することで、V_{DD} 電源を監視できます。PVD は、PVD_IN ピンの電圧レベルの監視にも使用できます。この場合、PVD_IN 電圧は内部 V_{REFINT} レベルと比較されます。

PVD は、*PWR 制御レジスタ 1（PWR_CR1）* の PVDE ビットをセットすることで有効になります。

PWR 制御ステータスレジスタ 1（PWR_CSR1） の PVDO フラグを使用して、V_{DD} または PVD_IN 電圧が PVD の閾値よりも高いか低いかを示すことができます。このイベントは EXTI に内部接続され、また EXTI レジスタで有効化されている場合は、割り込みを生成することができます。EXTI 立ち上がり／立ち下がりエッジの設定に応じて、V_{DD} または PVD_IN 電圧が PVD の閾値を下回るか、V_{DD} または PVD_IN 電圧が PVD の閾値を上回ったとき、あるいはその両方で、pwr_pvd_wkup 出力割り込みを生成することができます。たとえば、サービスルーチンで、緊急停止処理を実行することなどが可能です。

図 20. PVD の閾値



1. 閾値とヒステリシス値については、データシートを参照してください。

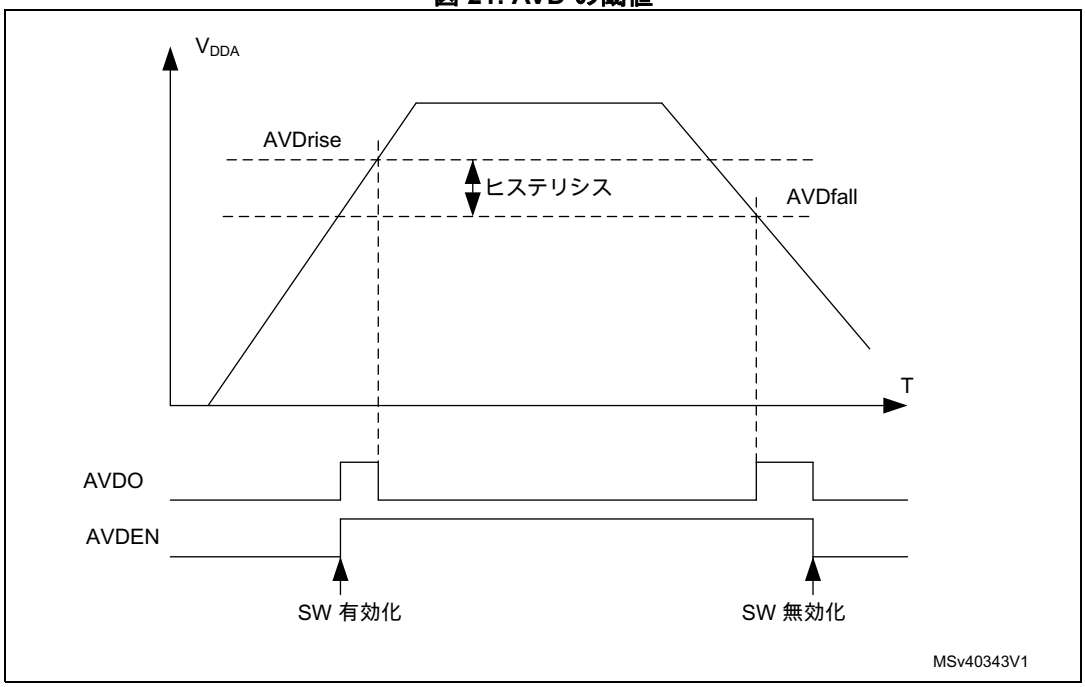
6.5.4 アナログ電圧検出器（AVD）

AVD を使用して、*PWR 制御レジスタ 1（PWR_CR1）* の ALS[1:0] ビットで選択した閾値と比較することで、V_{DDA} 電源を監視できます。

AVD は、*PWR 制御レジスタ 1（PWR_CR1）* の AVDEN ビットをセットすることで有効になります。

PWR 制御ステータスレジスタ 1（PWR_CSR1） の AVDO フラグを使用して、V_{DDA} が AVD の 閾値よりも高いか低いかを示すことができます。このイベントは EXTI に内部接続され、EXTI レジスタで有効化されている場合は、割り込みリクエストを生成させることができます。EXTI 立ち上がり／立ち下がりエッジの設定に応じて、V_{DDA} が AVD の閾値を下回るか上回ったとき、あるいはその両方で、pwr_avd_wkup 割り込みを生成することができます。たとえば、サービスルーチンでV_{DDA} 電源がいつ最小レベルを下回るかを示すことができます。

図 21. AVD の閾値



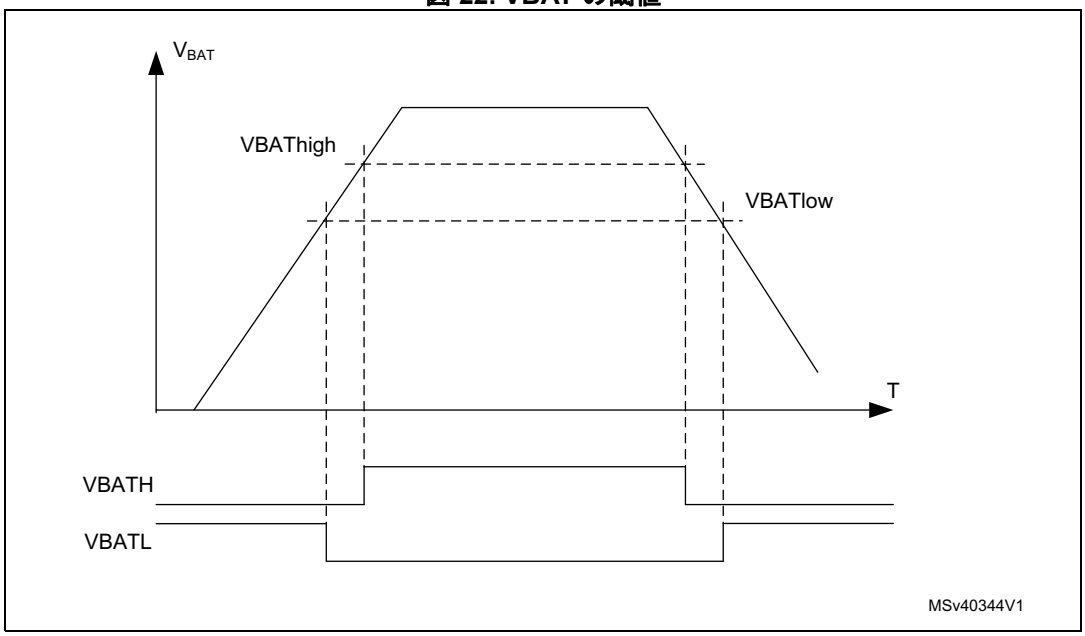
1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.5 バッテリ電圧の閾値

V_{BAT} バッテリ電源は、$V_{BAThigh}$ と V_{BATlow} の 2 つの閾値レベルと比較することで監視することができます。*PWR 制御レジスタ 2（PWR_CR2）* の VBATH および VBATL フラグは、V_{BAT} が閾値よりも高いか低いかを示しています。V_{BAT} 電源監視は、*PWR 制御レジスタ 2（PWR_CR2）* の MONEN ビットによって有効／無効にすることができます。有効にすると、バッテリ電圧の閾値によって消費電力が増加します。たとえば、このレベルを使用して緊急事態を回避するタスクを実行するルーチンをトリガすることなどが可能です。

RTC タンパ信号では、VBATH および VBATL ウェイクアップ割り込みを使用できます（*セクション 46：リアルタイムクロック（RTC）* を参照）。

図 22. VBAT の閾値



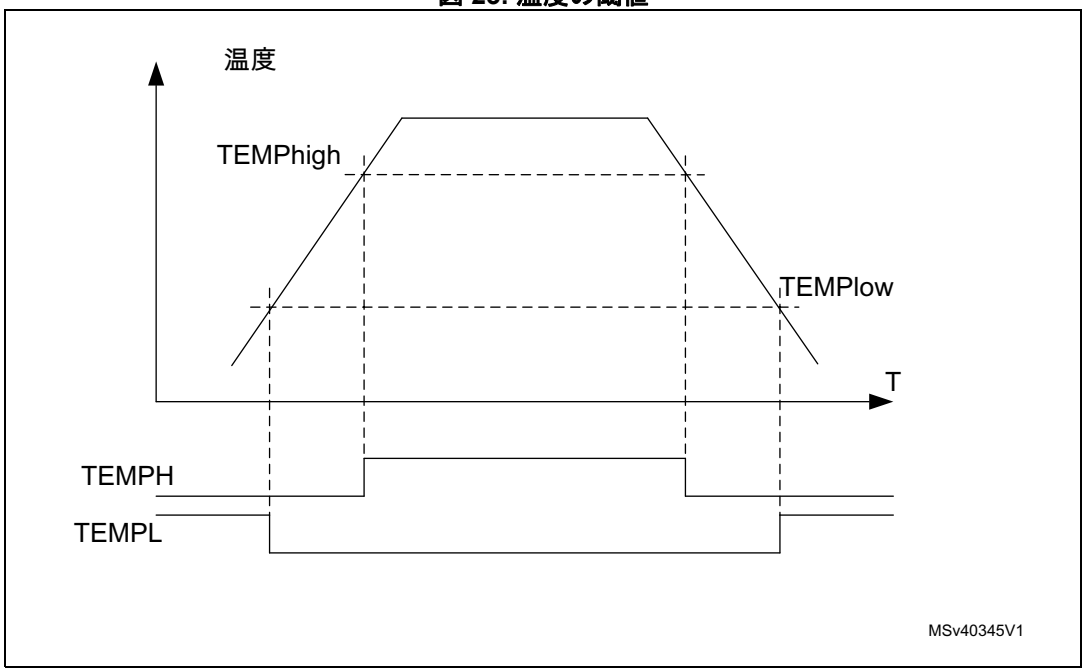
1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.6 温度の閾値

接合温度は、$TEMP_{high}$ と $TEMP_{low}$ の 2 つの閾値レベルと比較することで監視することができます。*PWR 制御レジスタ 2（PWR_CR2）* の TEMPH および TEMPL フラグは、デバイスの温度が閾値よりも高いか低いかを示しています。温度監視は、*PWR 制御レジスタ 2（PWR_CR2）* の MONEN ビットによって有効／無効にすることができます。有効にすると、温度の閾値によって消費電力が増加します。たとえば、このレベルを使用して温度制御タスクを実行するルーチンをトリガすることなどが可能です。

RTC タンパ信号では、TEMPH および TEMPL ウェイクアップ割り込みを使用できます（*セクション 46：リアルタイムクロック（RTC）* を参照）。

図 23. 温度の閾値



1. 閾値とヒステリシス値については、データシートを参照してください。

6.6 電源管理

電源管理ブロックでは、システムの動作モード（*セクション 6.6.1*を参照）に合わせて V_{CORE} 電源を制御します。

V_{CORE} ドメインは以下の電源ドメインに分割されます。

- D1 ドメイン：一部のペリフェラルと Cortex®-M7 コア（CPU）を含む
- D2 ドメイン：大部分のペリフェラルを含む
- D3 ドメイン：一部のペリフェラルとシステム制御を含む

D1、D2、およびシステム D3 電源ドメインは、次のいずれかの動作モードで動作できます。

- DRun/RUN/RUN* （パワーオン、クロックオン）
- DStop/STOP （パワーオン、クロックオフ）
- DStandby/STANDBY （パワーオフ、クロックオフ）

D1 ドメインおよび D2 ドメインの動作モードは独立しています。ただし、システム D3 ドメイン電源モードは D1 および D2 のドメインモードに依存します。

- システム D3 ドメインを STOP モードで動作するには、D1 および D2 の両方のドメインを DStop または DStandby モードにする必要があります。
- システム D3 ドメインを STANDBY モードで動作するには、同様に D1 および D2 の両方のドメインを DStandby モードにする必要があります。

D1、D2、およびシステム D3 ドメインは、共通の V_{CORE} レベルで単一のレギュレータから供給されます。V_{CORE} 供給レベルは、システムの動作モード（RUN、STOP、および STANDBY）に従います。D1 ドメインや D2 ドメイン電源は、これらのドメインが DStandby モードの場合に個別にパワーダウンすることができます。

以下の電圧スケーリング機能では、必要なシステム性能に応じて電源を制御することができます（*セクション 6.6.2：電圧スケーリング*を参照）。

- 所定のシステム性能を得るには、対応する電圧スケーリングをシステムクロック周波数に合わせて設定します。これには、VOS ビットを RUN モードの電圧スケーリングに設定します。
- 消費電力と STOP モードからの復帰の遅延の間で最高のトレードオフを実現するには、SVOS ビットを STOP モードの電圧スケーリングに設定します。

6.6.1 動作モード

必要な性能に合わせてシステムを調整するために、いくつかのシステム動作モードを使用できます（CPU がコードを実行する必要がない場合で、外部イベントを待っている場合）。消費電力の節減、起動時間の短縮、使用可能なウェイクアップソースを考慮した最適な動作モードの選択はユーザに委ねられています。

動作モードでは、異なるシステムブロックへのクロック分配を制御し、電源を供給します。システム動作モードは CPU サブシステム、D2 ドメイン、およびシステム D3 自律ウェイクアップで駆動されます。CPU サブシステムには、ペリフェラルの割り当てによって、複数のドメインを含めることができます（*セクション 8.5.11：ペリフェラルクロックゲーティングの制御* を参照）。

異なるシステムブロックには次の動作モードを使用できます（*表 27* を参照）。

- CPU サブシステムモード：
  - **CRun**

    RCC PERxEN ビットによって配置された CPU および CPU サブシステムペリフェラルがクロックされます。
  - **CSleep**：

    CPU クロックは停止し、CPU サブシステムに割り当てられたペリフェラルクロックは RCC PERxLPEN に従って動作します。
  - **CStop**：

    CPU および CPU サブシステムペリフェラルのクロックは停止されます。
- D1 ドメインモード：
  - DRun

    ドメインバスマトリックスにはクロックが供給されます。CPU サブシステムは CRun または CSleep モードで動作します。
  - DStop

    ドメインバスマトリックスのクロックは停止されます。

    CPU サブシステムは CStop モードで動作し、PDDS_D1[(1)] ビットは DStop モードを選択します。
  - DStandby

    ドメインがパワーダウンされます。

    CPU サブシステムは CStop モードで動作し、PDDS_D1 ビットは DStandby モードを選択します。
- D2 ドメインモード：
  - DRun

    ドメインバスマトリックスにはクロックが供給されます。

    CPU サブシステムでは D2 ドメインにペリフェラルが割り当てられおり、CRun または CSleep モードで動作します。

1. PDDS_Dn ビットは、*PWR CPU 制御レジスタ（PWR_CPUCR）* に属します。

– DStop

ドメインバスマトリックスのクロックは停止されます。

CPU サブシステムでは D2 ドメインにペリフェラルが割り当てられておらず、PDDS_D2（1）ビットは DStop モードを選択します。

または

CPU サブシステムでは D2 ドメインにペリフェラルが割り当てられおり、CStop モードで動作します。また、PDDS_D2 ビットは DStop モードを選択します。

– DStandby

ドメインがパワーダウンされます。

CPU サブシステムでは D2 ドメインにペリフェラルが割り当てられておらず、PDDS_D2 ビットは DStandby モードを選択します。

または

CPU サブシステムの D2 ドメインにはペリフェラルが割り当てられおり、CStop モードで動作します。また、PDDS_D2 ビットは DStandby モードを選択します。

- システム／D3 ドメインモード
  - **RUN/RUN***

    システムクロックおよび D3 ドメインバスマトリックスのクロックは動作します。
    - CPU サブシステムは CRun または CSleep モードで動作します。
    または
    - ウェイクアップ信号はアクティブです。（システム D3 自律モード）

    RUN* モードには POR リセットおよび STANDBY からのウェイクアップ後に移行します。RUN* モードでは性能は制限されており、システム電源設定は *PWR 制御レジスタ 3（PWR_CR3）* でプログラムされます。システムは、*PWR 制御ステータスレジスタ 1（PWR_CSR1）* の ACTVOSRDY ビットが 1 にセットされた場合にのみ RUN モードに移行します。
  - **STOP**

    システムクロックおよび D3 ドメインバスマトリックスのクロックは停止されます。
    - CPU サブシステムは CStop モードで動作します。
    および
    - ウェイクアップ信号はすべてインアクティブです。
    および
    - 任意のドメインの少なくとも 1 つの PDDS_Dn (1) ビットでSTOP モードを選択します。
  - **STANDBY**

    システムがパワーダウンされます。
    - CPU サブシステムは CStop モードで動作します。
    および
    - ウェイクアップ信号はすべてインアクティブです。
    および
    - すべてのドメインのすべての PDDS_Dn (1) ビットで STANDBY モードを選択します。

RUN モードでは、次の手段のいずれかによって消費電力を節減できます。

- システムクロックを低速化し、V_{CORE} 供給レベルを VOS 電圧スケーリングビットによって低減することにより、システム性能を下げます。
- PERxEN ビットによって、使用しない APBx および AHBx ペリフェラルへのクロックのゲーティングを行います。

表 27. 低電力モードの概要

システム	ドメイン	CPU	エントリ	ウェイクアップ	システムオシレータ	システムクロック	ドメインバスマトリックスクロック	ペリフェラルクロック	CPU クロック	電圧レギュレータ	ドメインの電源
RUN	DRun(1)	CRun	-	-	オン	オン	オン	オン	オン	オン	オン
		CSleep	WFI または ISR からの復帰、または WFE	任意の割り込みまたはイベント				オン／オフ(2)	オフ		
	DStop(3)	CStop	SLEEPDEEP ビット + WFI または ISR からの復帰または WFE	任意の EXTI 割り込みまたはイベント			オフ	オン／オフ(4)			
	DStandby (3)										オフ
STOP(5)	DStop (3)		SLEEPDEEP ビット + WFI または ISR からの復帰または WFE またはウェイクアップソースのクリア(6)		オン／オフ(7)	オフ		オフ			??
	DStandby (3)										オフ
STANDBY(8)	DStandby (3)		すべての PDDS_Dn ビット + SLEEPDEEP ビット + WFI または ISR からの復帰または WFE またはウェイクアップソースのクリア (6)	WKUP ピンの立ち上がり／立ち下がりエッジ、RTC アラーム（アラーム A またはアラーム B）、RTC ウェイクアップイベント、RTC タンパイベント、RTC タイムスタンプイベント、NRST ピンによる外部リセット、IWDG リセット	オフ					オフ	

1. CPU サブシステムは D2 ドメインにペリフェラルが割り当てられており、CRun または CSleep モードで動作します。
2. PERxLPEN ビットを持つ CPU サブシステムのペリフェラルは、これに従って動作します。
3. CPU サブシステムの D2 ドメインにペリフェラルが割り当てられている場合は、CStop モードで動作する必要があります。
4. PERxAMEN ビットを持つ CPU サブシステムのペリフェラルは、これに従って動作します。
5. すべてのドメインが DStop または DStandby である必要があります。
6. CPU が CStop で D3 ドメインが自律モードの場合、最後の EXTI ウェイクアップソースはクリアされます。
7. システムオシレータの HSI または CSI が使用されている場合、状態は HSIKERON および CSIKERON で制御され、それ以外の場合はシステムオシレータがオフになります。
8. ドメインはすべて DStandby モードです。

6.6.2 電圧スケーリング

D1、D2、および D3 ドメインは、次の機能を持つ電圧スケーリングをサポートする単一の電圧レギュレータから供給されます。

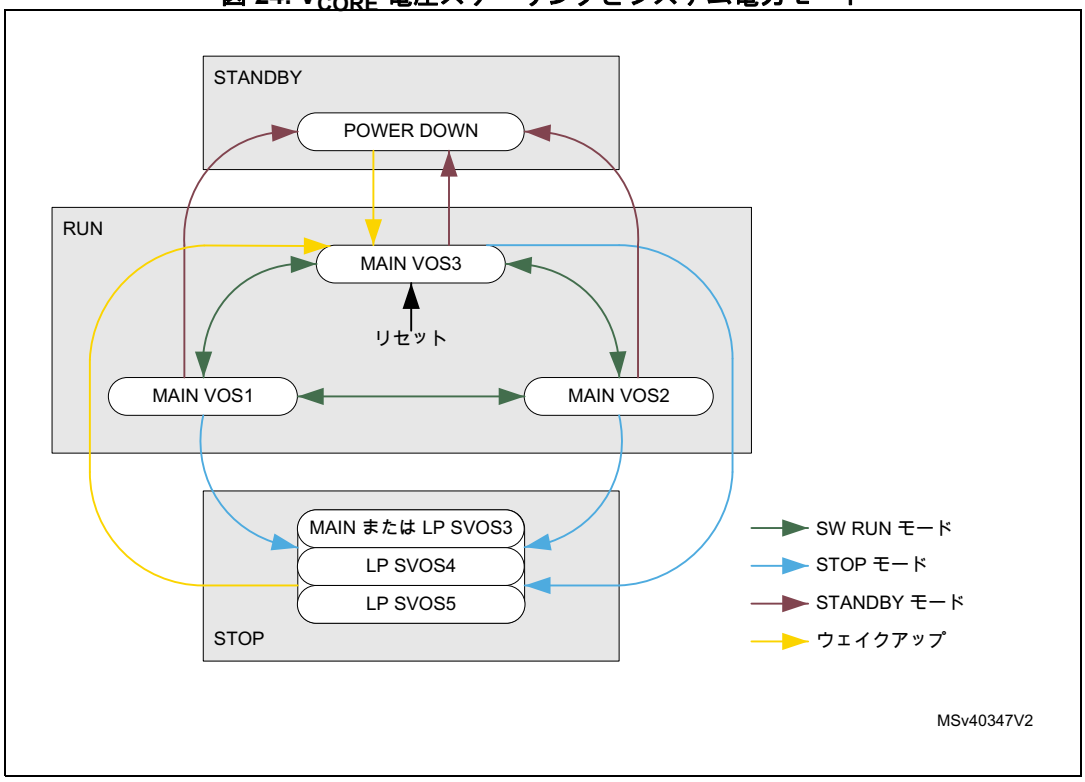
- RUN モードの電圧スケーリング
  - VOS1：スケール 1
  - VOS2：スケール 2
  - VOS3：スケール 3
- STOP モードの電圧スケーリング
  - SVOS3：スケール 3
  - LP-SVOS4：スケール 4
  - LP-SVOS5：スケール 5

電圧スケーリング値の詳細については、製品データシートを参照してください。

リセット後、システムはRUN モードの最低の電圧スケーリング（VOS3）で開始します。電圧スケーリングは、その後必要なシステム性能に合わせて *PWR D3 ドメイン制御レジスタ（PWR_D3CR）* の VOS ビットをプログラムすることで、ソフトウェアによって動作中に変更することができます。STOP モードまたは STANDBY モードを終了すると、RUN モードの電圧スケーリングはデフォルトの VOS3 値にリセットされます。

STOP モードに移行する前に、ソフトウェアは *PWR 制御レジスタ 1（PWR_CR1）* で SVOS レベルを事前選択することができます。SVOS4 および SVOS5 の STOP モードの電圧スケーリングでは、さらに消費電力を低減するために電圧レギュレータを低電力（LP）モードにセットします。SVOS3 を事前選択すると、LPDS レジスタビットで電圧レギュレータの低電力（LP）モードの使用を選択できます。

図 24. V_{CORE} 電圧スケーリングとシステム電力モード



6.6.3 電源制御モード

電源制御ブロックでは、システムの RUN、STOP、および STANDBY モードの V_{CORE} 電源を処理します。

システムの動作モードは CPU サブシステムモード（CRun、CSleep、CStop）、ドメインモード（DRun、DStop、DStandby）、およびシステム D3 自律ウェイクアップに依存します。

- RUN モードでは、V_{CORE} は VOS 電圧スケーリングによって定義されます。
  CPU サブシステムは CRun または CSleep で動作するか、EXTI ウェイクアップがアクティブになります。
- STOP モードでは、V_{CORE} は SVOS 電圧スケーリングによって定義されます。
  CPU サブシステムは CStop モードで動作し、EXTI ウェイクアップはすべてインアクティブになります。D1 ドメインおよび D2 ドメインは DStop または DStandby モードのどちらかです。
- STANDBY モードでは、V_{CORE} 電源がオフになります。
  CPU サブシステムは CStop モードで動作し、EXTI ウェイクアップはすべてインアクティブになります。D1 ドメインおよび D2 ドメインはどちらも DStandby モードです。

ペリフェラルが対応するドメインに割り当てられている場合、ドメインの動作モードは CPU サブシステムに依存します。DStop および DStandby 間のドメインモード選択は、*PWR CPU 制御レジスタ（PWR_CPUCR）* のドメイン専用の PDDS_Dn ビットで設定されます。CPU は、DStop のままドメインを保持するか、DStandby モードに移行するかを選択できます。

ドメインが DStandby モードの場合、対応する電源はオフになります。

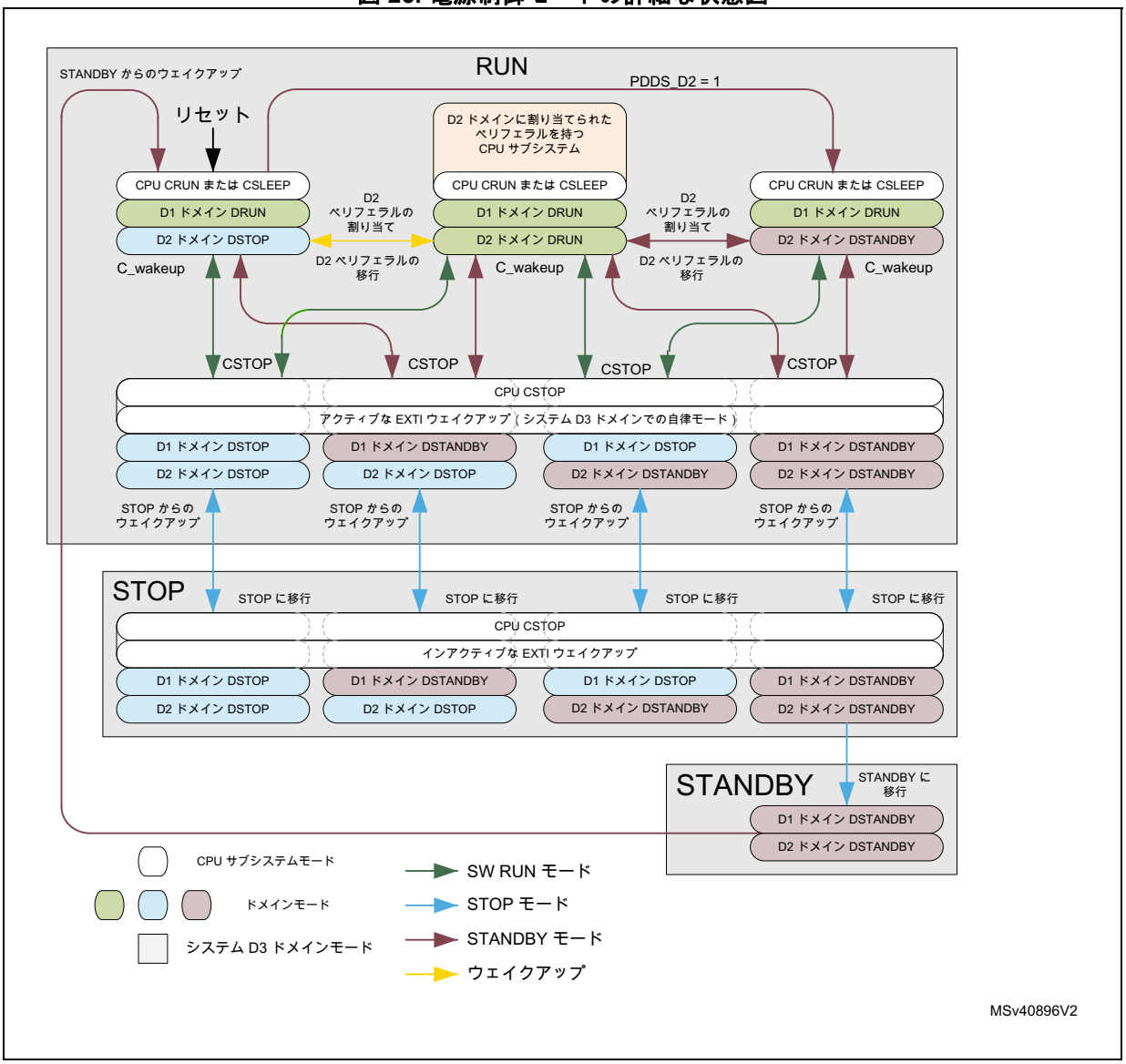
すべてのドメインのシステムモード（Stop または Standby）は、*PWR CPU 制御レジスタ（PWR_CPUCR）* の PDDS_Dn ビットを使用して設定できます。システムは、すべてのドメインのすべての PDDS_Dn ビットが許可している場合にのみ Standby に移行します。

表 28. PDDS_Dn 低電力モードの制御

PWR_CPUCR			D1 モード	D2 モード	D3 モード
PDDS_D1	PDDS_D2	PDDS_D3			
0	x	x	DStop	任意	RUN または STOP
1			DStandby	任意	任意
x	0		任意	DStop	RUN または STOP
	1		任意	DStandby	任意
少なくとも 1 つ = 0			DStop または DStandby	DStop または DStandby	STOP
1	1	1	DStandby	DStandby	STANDBY

参考資料

図 25. 電源制御モードの詳細な状態図



システムリセット後、CPU は CRun モードに移行します。

電源制御状態の遷移は次のイベントによって開始されます。

- CPUはCStop モードに移行（Run モードでの状態の遷移を緑と赤で示す）
  - 緑の遷移：CPU は CSleep からウェイクアップします。
  - 赤の遷移：CPU はドメインのリセットによってウェイクアップします。SBF_Dn がセットされます。
- ドメインへのペリフェラルの割り当てまたは移行（Run モードでの状態の遷移をオレンジと赤で示す）
  - オレンジの遷移：ドメインが DStop からウェイクアップします。
  - 赤の遷移：ドメインが DStandby からウェイクアップします。SBF_Dn がセットされます。
- システムの STOP モードへの移行または終了（状態の遷移を青で示す）
  - 青の遷移：システムが STOP モードからウェイクアップします。STOPF がセットされます。
- システムの STANDBY モードへの移行または終了（STOP および STANDBY モードでの状態の遷移を赤で示す）
  - STANDBY モードを終了するときに、SBF がセットされます。

ドメインが DStandby を終了するときに、ドメインの SBF_Dn ビットがセットされた状態でドメインのペリフェラルがリセットされます（ドメインのリセットを伴う状態の遷移を赤で示す）。

表 29 に、ドメイン／システム終了時のモードを示すフラグを表示します。CPU には、*PWR CPU 制御レジスタ（PWR_CPUCR）*から読み出し可能な一連のフラグがあります。

表 29. 低電力終了モードフラグ

システムモード	D1 ドメインモード	D2 ドメインモード	SBF_D1	SBF_D2	SBF	STOPF	コメント
RUN	DRun または DStop	DRun または DStop	0	0	0	0	D1、D2、およびシステムの内容を保持
RUN	DStandby	DStop	1	0	0	0	D1 の内容を失い、D2 およびシステムの内容を保持
RUN	DRun または DStop	DStandby	0	1	0	0	D2 の内容を失い、D1 およびシステムの内容を保持
RUN	DStandby	DStandby	1	1	0	0	D1 および D2 の内容を失い、システムの内容を保持
STOP	DStop	DStop	0	0	0	1	D1、D2、およびシステムの内容を保持し、クロックシステムをリセット
STOP	DStandby	DStop	1	0	0	1	D1 の内容を失い、D2 およびシステムの内容を保持し、クロックシステムをリセット
STOP	DStop	DStandby	0	1	0	1	D2 の内容を失い、D1 およびシステムの内容を保持し、クロックシステムをリセット
STOP	DStandby	DStandby	1	1	0	1	D1 および D2 の内容を失い、システムの内容を保持し、クロックシステムをリセット
STANDBY	DStandby	DStandby	0(1)	0(1)	1	0	D1、D2、およびシステムの内容を失う

1. STANDBY から復帰する際に、SBF_D1 と SBF_D2 はリセット値を反映します。

6.6.4 電源管理の例

- 図 26 は、RUN モードでの V_{CORE} 電圧スケーリングの挙動を示します。
- 図 27 は、STOP モードでの V_{CORE} 電圧スケーリングの挙動を示します。
- 図 28 は、STANDBY モードでの V_{CORE} 電圧レギュレータおよび電圧スケーリングの挙動を示します。
- 図 29 は、D1 および D2 ドメインが DStandby モードの場合の RUN モードでの V_{CORE} 電圧スケーリングの挙動を示します。

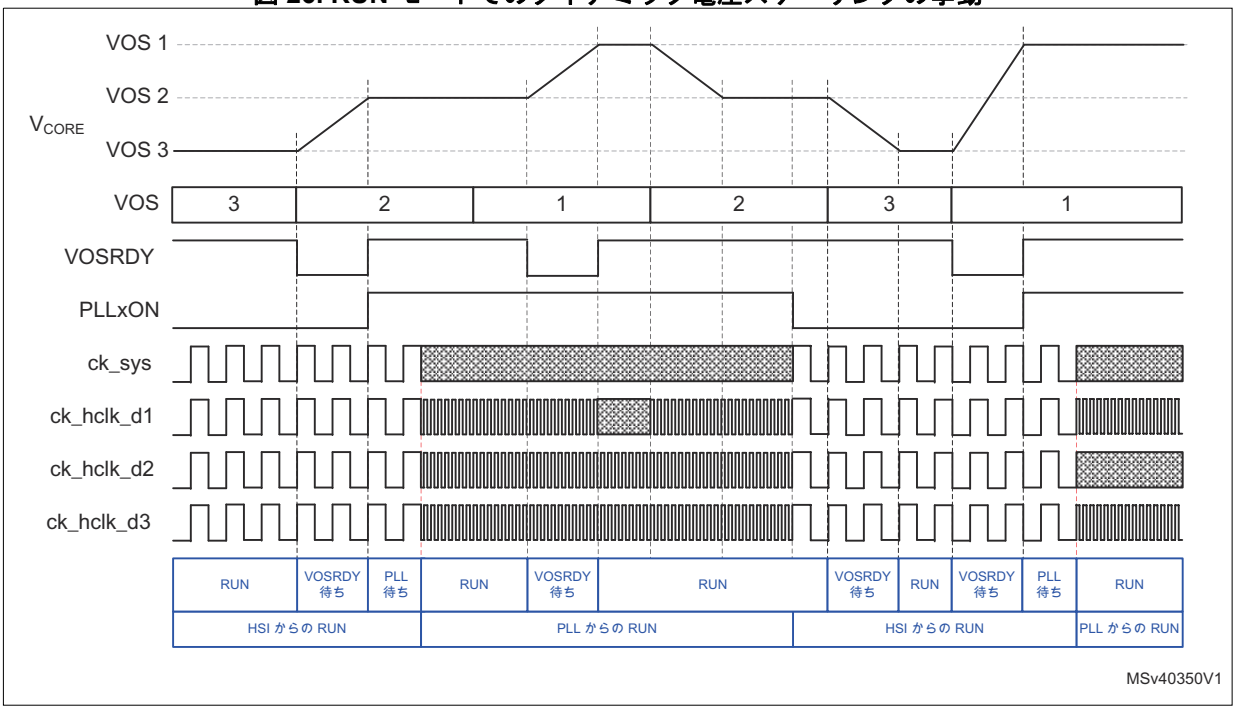
RUN モードでの V_{CORE} 電圧スケーリングの挙動の例

図 26 に、以下のシステム動作シーケンスの例を示します。

1. リセット後、システムは VOS3設定でHSI から開始します。
2. システム性能は、最初に電圧スケーリング VOS2 の設定で PLL からミディアムスピードクロックに上げられます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS2 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、PLL を有効にしてクロック周波数を増加します。
   c) PLL がロックされたら、システムクロックを切り替えます。
3. 次に、システム性能は、電圧スケーリング VOS1 の設定で PLL からハイスピードクロックに上げられます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS1 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、クロック周波数を増加します。
4. その後、システム性能は、電圧スケーリング VOS2 の設定でミディアムスピードクロックまで低下します。このためには、次の操作を行います。
   a) まず、システム周波数を下げます。
   b) 次に、電圧スケーリングを VOS2 に下げます。
5. 次のステップでは、システム性能を電圧スケーリング VOS3 の設定で HSI クロックまで低下します。このためには、次の操作を行います。
   a) クロックを HSI に切り替えます。
   b) PLL を無効にします。
   c) 電圧スケーリングを VOS3 に下げます。
6. その後、システム性能は、PLL からハイスピードクロックに上げることができます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS1 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、PLL を有効にしてクロック周波数を増加します。
   c) PLL がロックされたら、システムクロックを切り替えます。

システム性能（クロック周波数）が変わると、VOS は適切にセットされます。セットされない場合、システムは信頼できない場合があります。

図 26. RUN モードでのダイナミック電圧スケーリングの挙動



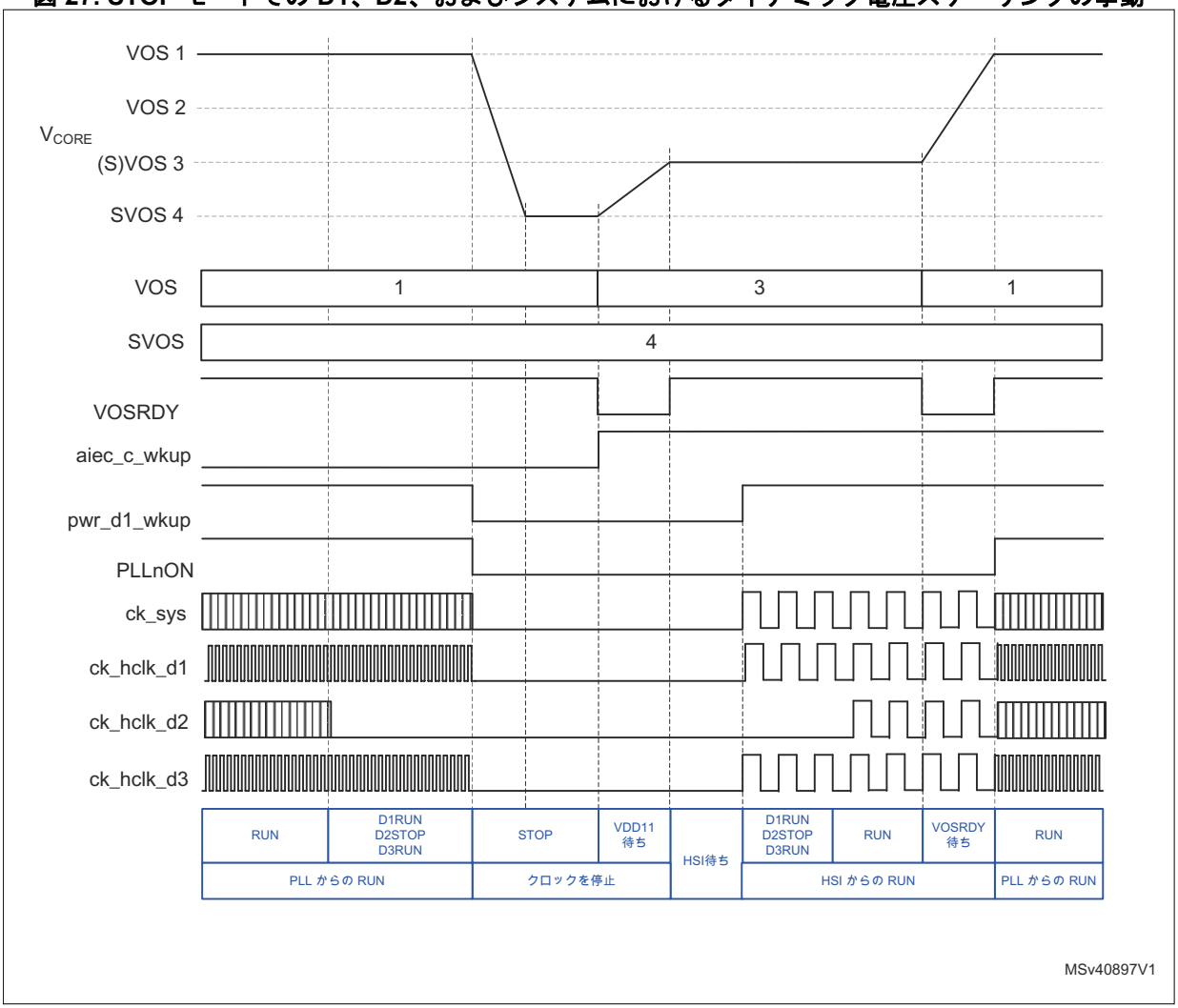
1. 各ステップでのレジスタビットのステータスを青で示します。

STOP モードでの V_{CORE} 電圧スケーリングの挙動の例

図 27 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスモードで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは、最初に DStop モードに移行する D2 ドメインのすべてのペリフェラルを割り当て解除します。D2 システムクロックは停止します。システムではハイパフォーマンスなシステムクロックを提供し続けるため、電圧スケーリングは VOS1 レベルのままにする必要があります。
3. 2 番目のステップで、CPU サブシステムは CStop モードに移行し、D1 ドメインは DStop モードに移行し、システムは STOP モードに移行します。システムクロックは停止され、ハードウェアでは電圧スケーリングをソフトウェアで事前選択された SVOS4 レベルに下げます。
4. これにより、CPU サブシステムがウェイクアップします。システムは STOP モードを終了し、D1 ドメインは DStop モードを終了し、CPU サブシステムは CStop モードを終了します。その後、ハードウェアは HSI を有効にする前に電圧スケーリングを VOS3 に設定して、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと D1 システムクロックが有効になります。
5. CPU サブシステムでは D2 ドメインにペリフェラルを割り当てます。D2 システムクロックが有効になります。
6. これにより、システム性能が向上します。このためには、次の操作を行います。
   a) ソフトウェアで最初に電圧スケーリングを VOS1 に設定します。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達すると、PLL を有効にしてクロック周波数を増加することができます。
   c) PLL がロックされると、システムクロックを切り替えることができます。

図 27. STOP モードでの D1、D2、およびシステムにおけるダイナミック電圧スケーリングの挙動



1. 各ステップでのレジスタビットのステータスを青で示します。

STANDBY モードの V_{CORE} 電圧レギュレータおよび電圧スケーリングの挙動の例

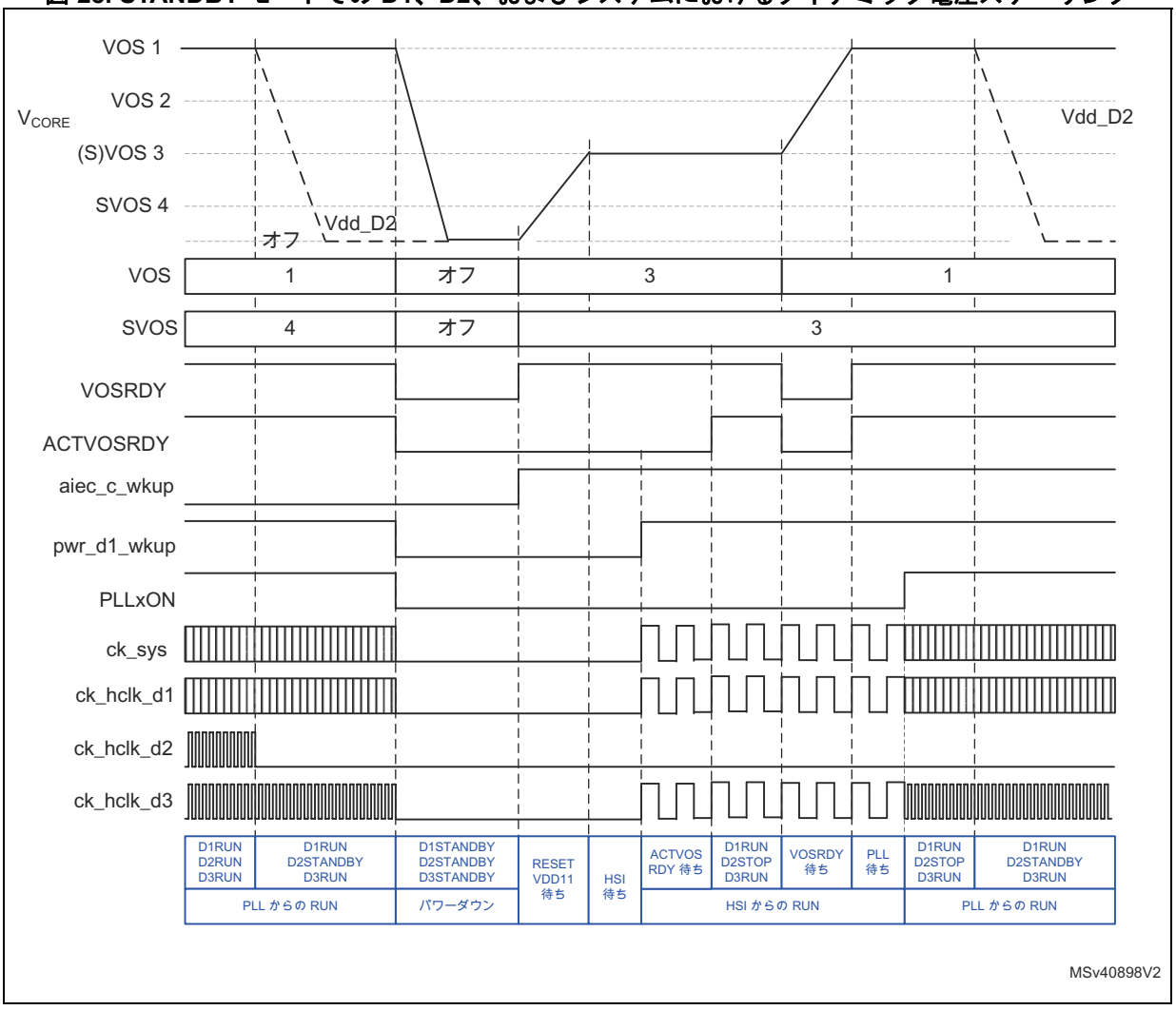
図 28 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスモードで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは、最初に DStandby モードに移行する D2 ドメインのすべてのペリフェラルを割り当て解除します。D2 ドメインバスマトリックスのクロックは停止され、電源はオフになります。電圧スケーリングが変わらないため、システム性能も変更されません。
3. その後、CPU サブシステムは CStop モードに移行し、D1 ドメインは DStandby モードに移行し、システムは STANDBY モードに移行します。システムクロックは停止され、電圧レギュレータはオフになります。
4. その後、システムはウェイクアップソースによってウェイクアップします。システムは STANDBY モードを終了します。ハードウェアはデフォルトの HSI オシレータを有効にする前に電圧スケーリングをデフォルトの VOS3 レベル にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと D1 サブシステムクロックが有効になります。D2 ドメインには割り当てられたペリフェラルはないため、このドメイン

は DStop モードを維持します。その後、ソフトウェアはシステム性能を変更する前に ACTVOSRDY をチェックして有効であることを確認します。

5. 次のステップではシステム性能を向上します。このためには、次の操作を行います。
   a) ソフトウェアでは最初に電圧スケーリングを VOS1 レベルに上げます。
   b) PLL を有効にする前に、VOSRDY ビットを監視して、リクエストした電源レベルに達するのを待ちます。
   c) PLL がロックされると、システムクロックを切り替えることができます。
6. CPU サブシステムは D2 ドメインを DStandby モードにします。

図 28. STANDBY モードでの D1、D2、およびシステムにおけるダイナミック電圧スケーリング



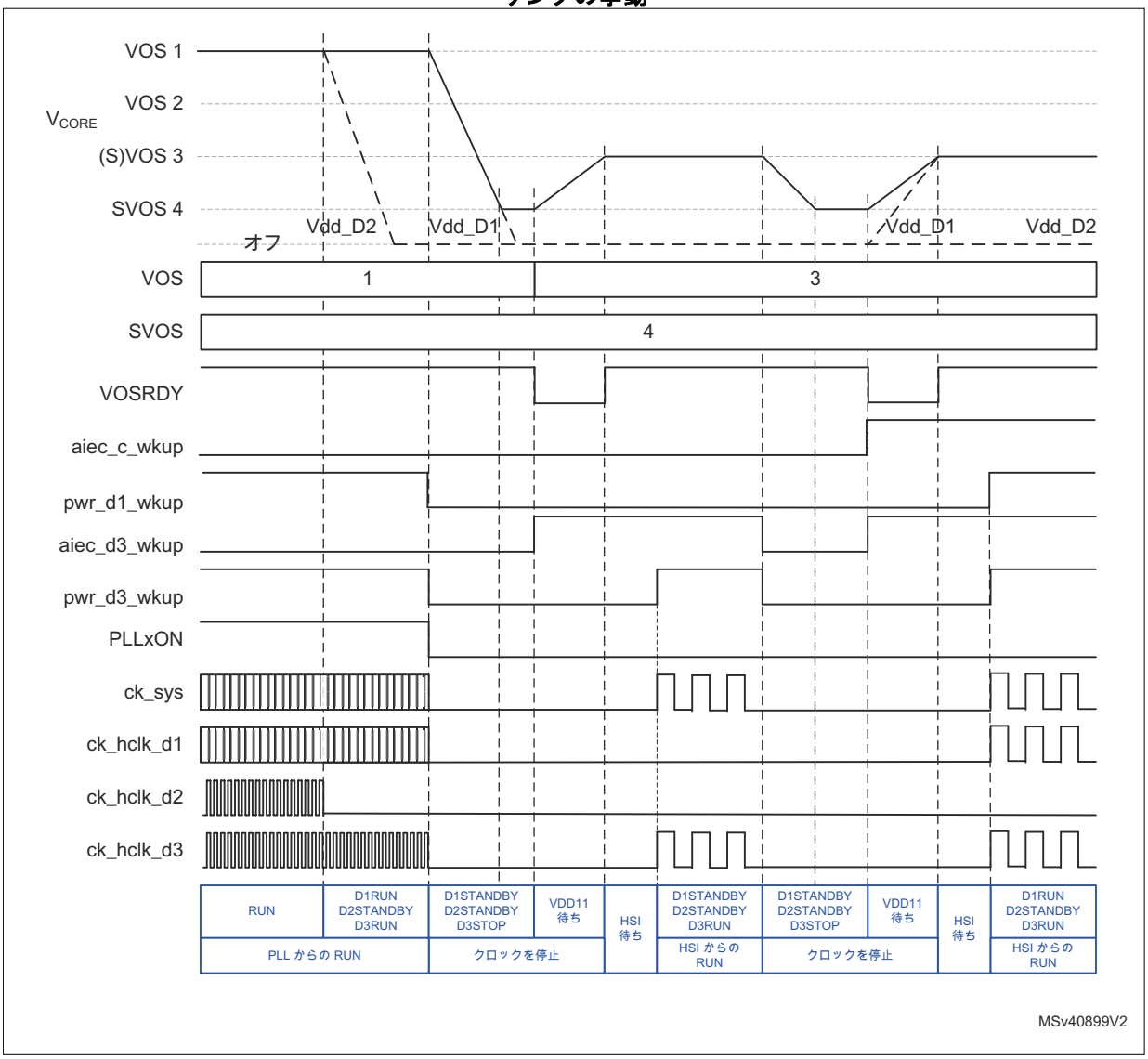
1. 各ステップでのレジスタビットのステータスを青で示します。

DStandby モードの D1 および D2 ドメインを持つ RUN モードでの V_{CORE} 電圧スケーリングの挙動の例

図 29 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは、最初に DStandby モードに移行する D2 ドメインのすべてのペリフェラルを割り当て解除します。D2 ドメインバスマトリックスのクロックは停止され、電源はオフになります。電圧スケーリングが変わらないため、システム性能も変更されません。
3. CPU サブシステムは CStop モードに移行し、D1 ドメインは DStandby モードに移行します。D1 ドメインバスマトリックスのクロックは停止され、電源はオフになります。同時に、システムは STOP モードに移行します。システムクロックは停止され、ハードウェアでは電圧スケーリングをソフトウェアで事前選択された SVOS4 レベルに下げます。
4. その後、システムは D3 自律モードのウェイクアップイベントによってウェイクアップします。システムは STOP モードを終了します。ハードウェアは HSI クロックを有効にする前に電圧スケーリングをデフォルトの VOS3 レベル にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックが有効になります。システムは D3 自律モードで動作します。
5. その後 D3 自律モードのウェイクアップソースはクリアされ、これによってシステムは STOP モードに移行します。システムクロックは停止され、電圧スケーリングはソフトウェアで事前選択された SVOS4 レベルに下げられます。
6. これにより、CPU サブシステムがウェイクアップします。システムは STOP モードを終了し、D1 ドメインは DStandby モードを終了し、CPU サブシステムは CStop モードを終了します。ハードウェアはデフォルトの HSI オシレータを有効にする前に電圧スケーリングをデフォルトの VOS3 にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと D1 サブシステムクロックが有効になります。D2 ドメインは、DStandby モードを維持します。

図 29. DStandby モードでの D1 および D2、および自律モードの D3 におけるダイナミック電圧スケーリングの挙動



1. 各ステップでのレジスタビットのステータスを青で示します。

6.7 低電力モード

CPU でコードの実行が不要なとき（外部イベント待ちなど）の節電のために、いくつかの低電力モードが用意されています。消費電力の節減、起動時間の短縮、使用可能なウェイクアップソースを考慮した最適なモード選択はユーザアプリケーションに委ねられています。

- システムクロックの低速化（*セクション 8.5.6：システムクロック（sys_ck）*を参照）
- 個々のペリフェラルクロックの制御（*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照）
- 低電力モード
  - CSleep （CPU クロックの停止）
  - CStop （CPU サブシステムクロックの停止）
  - DStop （ドメインバスマトリックスのクロックの停止）
  - STOP （システムクロックの停止）
  - DStandby （ドメインのパワーダウン）
  - STANDBY （システムのパワーダウン）

6.7.1 システムクロックの低速化

RUN モードでは、システムクロック ck_sys の速度を下げることができます。詳細については、*セクション 8.5.6：システムクロック（sys_ck）*を参照してください。

6.7.2 ペリフェラルクロックの制御

RUN モードでは、消費電力を低減するため、いつでも RCC_C1_xxxxENR または RCC_DnxxxxENR の PERxEN ビットを設定して個々のペリフェラルへの HCLKx および PCLKx を停止することができます。

CSleep モードで消費電力を低減するため、RCC_C1_xxxxLPENR または RCC_DnxxxxLPENR の PERxLPEN ビットを設定することで、個々のペリフェラルクロックを停止することができます。まだ CSleep モードでクロックを受信しているペリフェラルは、CSleep モードに移行する前にクロックを低速化することができます。

6.7.3 低電力モードへの移行

CPU サブシステムの CSleep および CStop 低電力モードへの移行は、MCU が WFI （Wait For Interrupt）または WFE （Wait For Event）命令を実行することにより、または Cortex®-M システム制御レジスタの SLEEPONEXIT ビットが割り込みサービスルーチン（ISR）終了時にセットされている時点で実行されます。

ドメインの DStop または DStandby 低電力モードへの移行は、CPU サブシステムのドメインにペリフェラルが割り当てられていて CStop モードに移行している場合、またはすべての D2 ドメインのペリフェラルが割り当て解除されている場合に実行可能です。

システムの STOP または STANDBY 低電力モードへの移行は、すべての EXTI ウェイクアップソースがクリアされ、その他のドメインが DStop または DStandby モードである場合に実行可能です。

6.7.4 低電力モードの終了

CPU サブシステムは、低電力モードへの移行方法に応じて、任意の割り込みやイベントにより CSleep モードを終了します。

- WFI 命令または 割り込みサービスルーチン（ISR）終了によって低電力モードに移行した場合、NVIC によって認識されたペリフェラル割り込みであればどれでもシステムをウェイクアップすることができます。
- WFE 命令によって低電力モードに移行した場合、CPU はイベントの発生直後に低電力モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。
  - NVIC IRQ 割り込み

    Cortex®-M7 システム制御レジスタの **SEVONPEND = 0 の場合**：ペリフェラル制御レジスタおよび NVIC にて割り込みを有効にする必要があります。

    MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと NVIC 割り込みクリアペンディングレジスタの NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。十分な優先度の NVIC 割り込みのみが MCU をウェイクアップし、割り込みを生成します。

    Cortex®-M7 システム制御レジスタの **SEVONPEND = 1 の場合**：ペリフェラル制御レジスタおよび任意で NVIC にて割り込みを有効にする必要があります。MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと、有効であれば NVIC 割り込みクリアペンディングレジスタの NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    すべての NVIC 割り込み（無効化されているものも含む）によって MCU をウェイクアップします。

    十分な優先度の有効化された NVIC 割り込みのみが MCU をウェイクアップし、割り込みを生成します。
  - イベント

    EXTI ラインはイベントモードに設定する必要があります。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、EXIT ペリフェラル割り込みペンディングビットや NVIC IRQ チャネルペンディングビットをクリアする必要はありません。ペリフェラルの割り込みフラグをクリアする必要があるかもしれません。

CPU サブシステムは、低電力モードへの移行方法に応じて、EXTI 割り込みやイベントを有効にすることで CStop、DStop、および STOP モードを終了します（上記参照）。

システムは、EXTI ウェイクアップを有効にすることで、CPU サブシステムをウェイクアップすることなく STOP モードからウェイクアップすることができます。この場合、システムは D3 自律モードで動作します。

CPU サブシステムは、DStandby モードの移行方法にかかわらず、EXTI 割り込みやイベントを有効にすることで DStandby モードを終了します。プログラム実行は、システム設定ブロック（SYSCFG）からフェッチされたリセットベクタなどの、CPU のローカルリセットからリスタートします。

D2 ドメインは、CPU がドメインに最初のペリフェラルを割り当てると、DStop または DStandby モードを終了できます。

CPU サブシステムは、外部リセット（NRST ピン）、IWDG リセット、有効な WKUPx ピンのうちの 1 つの立ち上がりエッジ、または RTC イベントのいずれかを有効にすることで、STANDBY モードを終了します。プログラム実行はシステムリセット後と同様にリスタートされます（ブートピン信号のサンプリング、オプションバイトローディング、リセットベクタのフェッチなど）。

6.7.5 CSleep モード

CSleep モードは、CPU サブシステムにのみ適用されます。CSleep モードでは CPU クロックを停止します。CPU サブシステムのペリフェラルクロックは、RCC_C1_xxxxENR または RCC_DnxxxxENR の PERxLPEN ビットの値に従って動作します。

CSleep モードへの移行

CSleep モードへの移行は、Cortex®-M システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、*セクション 6.7.3：低電力モードへの移行* に従って実行されます。

CSleep モードへの移行方法の詳細については、*表 30* を参照してください。

CSleep モードの終了

CSleep モードは、*セクション 6.7.4：低電力モードの終了*に従って終了します。

CSleep モードの終了方法の詳細については、*表 30* を参照してください。

表 30. CSleep モード

CSleep モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割り込みとイベントがクリアされます。
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割り込みとイベントがクリアされます。
モードの終了	WFI または ISR からの復帰を使用して移行した場合 – NVIC で有効化されている任意の割り込み：*表 130：NVIC* を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： – 任意のイベント：*セクション 20.5.3：EXTI CPU ウェイクアップ手順* を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： – 任意の割り込み（NVIC で無効化されている場合も含む）：*表 130：NVIC* または任意のイベント：*セクション 20.5.3：EXTI CPU ウェイクアップ手順*を参照してください。
ウェイクアップ遅延時間	なし

6.7.6 CStop モード

CStop モードは、CPU サブシステムにのみ適用されます。CStop モードでは CPU クロックを停止します。ほとんどの CPU サブシステムのペリフェラルクロックも停止され、CPU サブシステムの PERxAMEN ビットが設定されたペリフェラルのみが、これに従って動作します。

CStop モードでは、CPU サブシステムのペリフェラルがすでにカーネルクロックリクエストを持っていても、さらにカーネルクロックを要求できます。ペリフェラルが PERxAMEN ビットを持っている場合は、カーネルクロックを要求できるようにこのビットをセットします。

CStop モードへの移行

CStop モードへの移行は、Cortex®-M システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、*セクション 6.7.3：低電力モードへの移行* に従って実行されます。

CStop モードへの移行方法の詳細については、*表 31* を参照してください。

CStop モードの終了

CStop モードは、*セクション 6.7.4：低電力モードの終了* に従って終了します。

CStop モードの終了方法の詳細については、*表 31* を参照してください。

表 31. CStop モード

CStop モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割り込みとイベントがクリアされます。 – すべての CPU EXTI ウェイクアップソースがクリアされます。
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 1 および – SLEEPONEXIT = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割り込みとイベントがクリアされます。 – すべての CPU EXTI ウェイクアップソースがクリアされます。
モードの終了	WFI または ISR からの復帰を使用して移行した場合 – NVIC で有効化されている EXTI 割り込み：停止またはパワーダウンされていないペリフェラルについては、*表 130：NVIC* を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： – EXTI イベント：停止またはパワーダウンされていないペリフェラルについては、*セクション 20.5.3：EXTI CPU ウェイクアップ手順* を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： – EXTI 割り込み（NVIC で無効化されている場合も含む）：*表 130：NVIC* または EXTI イベント：停止またはパワーダウンされていないペリフェラルについては、*セクション 20.5.3：EXTI CPU ウェイクアップ手順*を参照してください。
ウェイクアップ遅延時間	EXTI および RCC ウェイクアップ同期（*セクション 8.4.7：パワーオンおよびウェイクアップのシーケンス* を参照）

6.7.7 DStop モード

D1 ドメインや D2 ドメインの DStop モードへの移行は、CPU サブシステムが CStop モードで動作し、ドメインに割り当てられたペリフェラルがある場合にのみ実行されます（*表 32* を参照）。DStop モードでは、ドメインバスマトリックスのクロックが停止します。

フラッシュメモリの低電力 STOP モードへの移行は、PWR_CR1 レジスタの FLPS を使用してこのモードを有効にすることで実行できます。これにより、ドメインの DStop 再起動時間と少ない消費電力の間でトレードオフを行うことができます。

表 32. DStop モードの概要

ペリフェラルの割り当て	CPU	D1 ドメイン	D2 ドメイン	コメント
D2 ドメインに割り当てられたペリフェラルなし	CRun または CSleep	DRun	DStop	
	CStop	DStop	DStop	
D2 ドメインに割り当てられたペリフェラルあり	CRun または CSleep	DRun	DRun	CPU サブシステム、D2 ドメインをアクティブなまま保持します。
	CStop	DStop	DStop	

DStop モードでは、LSI または LSE クロックを使用するドメインペリフェラル、およびカーネルクロックリクエストを持つペリフェラルを、まだ動作することができます。

DStop モードへの移行

DStop モードへの移行は、ドメインの少なくとも 1 つの *PWR CPU 制御レジスタ（PWR_CPUCR）* の PDDS_Dn ビットで STOP を選択している場合に、*セクション 6.7.3：低電力モードへの移行* に従って実行されます。

DStop モードへの移行方法の詳細については、*表 33* を参照してください。

フラッシュメモリがプログラミング中の場合、メモリアクセスが終了してから、DStop モードに移行します。

ドメインバスマトリックスにアクセス中の場合は、ドメインバスマトリックスへのアクセスが完了してから DStop モードに移行します。

DStop モードの終了

DStop モードは、*セクション 6.7.4：低電力モードの終了* に従って終了します。

DStop モードの終了方法の詳細については、*表 33* を参照してください。

DStop モードを終了する際、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングは、システムモードに依存します。

- システムが STOP モードに移行しない場合、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングの値は、DStop モードに移行する場合と同じです。
- システムが STOP モードに移行すると、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングはリセットされます。

表 33. DStop モード

DStop モード	説明
モードへの移行	– ドメインの CPU サブシステムが CStop モードに移行します。 – CPU サブシステムの D2 ドメインにはペリフェラルが割り当てられており、CStop に移行します。 – CPU サブシステムでは、D2 ドメインの最後のペリフェラルを割り当て解除します。 – ドメインの PDDS_Dn ビットでは、STOP モードを選択します。
モードの終了	– ドメインの CPU サブシステムが CStop モードを終了します（*表 31* を参照）。 – CPU サブシステムの D2 ドメインにはペリフェラルが割り当てられており、CStop を終了します（*表 31* を参照）。 – CPU サブシステムでは、D2 ドメインの最初のペリフェラルを割り当てます。
ウェイクアップ遅延時間	EXTI および RCC ウェイクアップ同期（*セクション 8.4.7：パワーオンおよびウェイクアップのシーケンス* を参照）。

6.7.8 STOP モード

システム D3 ドメインの STOP モードへの移行は、CPU サブシステムが CStop モードで、EXTI ウェイクアップソースがインアクティブで、任意のドメインの少なくとも 1 つの *PWR CPU 制御レジスタ（PWR_CPUCR）* の PDDS_Dn ビットで STOP を要求している場合にのみ実行されます。STOP モードでは、PLL を含むシステムクロックおよび D3 ドメインバスマトリックスのクロックが停止します。HSI または CSI が選択されていると、システムオシレータは RCC_CR レジスタの HSIKERON および CSIKERON ビットに従って動作します。その他のシステムオシレータのソースは停止します。

システム D3 ドメインの STOP モードで、D1 ドメインおよび D2 ドメインは DStop および／または DStandby モードです。

STOP モードでは、LSI または LSE クロックを使用するドメインペリフェラル、および HSI または CSI をカーネルクロックのソースとして選択しているペリフェラルを、まだ動作することができます。

システムの STOP モードでは、アクティブな状態を維持するために、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）

  IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません（*セクション 45：独立型ウォッチドッグ（IWDG）* の *セクション 45.3* を参照）。
- リアルタイムクロック（RTC）

  この設定は*RCC バックアップドメイン制御レジスタ（RCC_BDCR）* の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI RC）

  この設定は*RCC クロック制御およびステータスレジスタ（RCC_CSR）* の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE OSC）

  この設定は*RCC バックアップドメイン制御レジスタ（RCC_BDCR）* の LSEON ビットで行います。

- LSI または LSE クロックで動作可能なペリフェラル
- カーネルクロックリクエストを持つペリフェラル
- 内部 RC オシレータ（HSI および CSI）
- この設定は*RCC クロック制御およびステータスレジスタ（RCC_CSR）*の HSIKERON および CSIKERON ビットで行います。
- ADC や DAC は、STOP モードに移行する前に停止させない限り、STOP モード中にも電力を消費します。ADC および DAC を停止させるには、ADC_CR2 レジスタの ADON ビットと DAC_CR レジスタの ENx ビットの両方に 0 を書き込む必要があります。

選択した SVOS4 および SVOS5 レベルにより、システムの STOP モードを終了する際の起動時間が長くなります（*表 34*を参照）。

表 34. STOP モード動作

SVOS	LPDS	STOP モード 電圧レギュレータ動作	ウェイクアップ遅延時間
SVOS3	0	**メイン**	ウェイクアップ時間の遅延なし
	1	**LP**	LP モードからの電圧レギュレータのウェイクアップ時間
SVOS4 または SVOS5	x	**LP**	LP モードからの電圧レギュレータのウェイクアップ時間 + SVOS4 または SVOS5 レベルから VOS3 レベルへの電圧レベルのウェイクアップ時間

STOP モードへの移行

STOP モードへの移行は、任意のドメインの少なくとも 1 つの *PWR CPU 制御レジスタ（PWR_CPUCR）*の PDDS_Dn ビットで STOP を要求している場合に、*セクション 6.7.3：低電力モードへの移行*に従って実行されます。

STOP モードへの移行方法の詳細については、*表 35* を参照してください。

フラッシュメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP モードに移行します。

バスマトリックスにアクセス中の場合は、バスマトリックスへのアクセスが終了してから STOP モードに移行します。

カーネルクロックリクエストを持つペリフェラルを STOP モードで動作するには、システムで SVOS3 レベルを使用する必要があります。

STOP モードの終了

STOP モードは、*セクション 6.7.4：低電力モードの終了*に従って終了します。

STOP モードの終了方法の詳細については、*表 35* を参照してください。

STOP モードを終了する際、システムクロック、D3 ドメインバスマトリックスのクロック、および電圧スケーリングはリセットされます。

*PWR CPU 制御レジスタ（PWR_CPUCR）*の STOPF ステータスフラグは、システムがSTOP モードを終了したことを表します（*表 29* を参照）。

表 35. STOP モード

STOP モード	説明
モードへの移行	– CPU が CStop モードで動作している場合、アクティブな EXTI ウェイクアップソースはなく、Run_D3 = 0 です。 – 任意のドメインの少なくとも 1 つの PDDS_Dn ビットでSTOP モードが選択されます。
モードの終了	– EXTI ウェイクアップ時
ウェイクアップ遅延時間	システムオシレータの起動（無効である場合） + EXTI および RCC ウェイクアップ同期 + 電圧スケーリングについては*表 34*を参照（*セクション 6.6.2：電圧スケーリング*を参照）

STOP モードにおける I/O の状態

STOP モードでの I/O ピンの設定は変わりません。

6.7.9 DStandby モード

DStop モードと同様に、DStandby モードは CPU サブシステムの CStop モードをベースにしています。ただし、ドメインの V_{CORE} 電源はオフになります。ドメインの DStandby モードへの移行は、ドメインにペリフェラルが割り当てられている場合に、CPU サブシステムが CStop モードで動作する場合にのみ実行されます。

ドメインの DStandby モードへの移行は、ドメインにペリフェラルが割り当てられ、ドメインの *PWR CPU 制御レジスタ（PWR_CPUCR）* の PDDS_Dn ビットがそれに応じて設定されている場合に、CPU サブシステムが CStop モードで動作する場合にのみ実行されます。DStandby モードで、ドメインはパワーダウンされ、ドメインの RAM とレジスタの内容は失われます。

DStandby モードへの移行

DStandby モードへの移行は、Dn ドメインの *PWR CPU 制御レジスタ（PWR_CPUCR）* の PDDS_Dn ビットで STANDBY モードを選択している場合に、*セクション 6.7.3：低電力モードへの移行* に従って実行されます。

DStandby モードへの移行方法の詳細については、*表 36* を参照してください。

フラッシュメモリがプログラミング中の場合、メモリアクセスが終了してから、DStandby モードに移行します。

ドメインバスマトリックスにアクセス中の場合は、ドメインバスマトリックスへのアクセスが終了してから DStandby モードに移行します。

注： ***CPU が STANDBY モードを選択するために PDDS_D2 ビットをセットすると、D2 ドメインは DStandby モードに移行します（CPU には D2 ドメインに割り当てられたペリフェラルはありません）。***

DStandby モードの終了

DStandby モードは、*セクション 6.7.4：低電力モードの終了* に従って終了します。

DStandby モードの終了方法の詳細については、*表 36* を参照してください。

注： ***D2 ドメインが DStandby モードで、CPU で STOP モードを選択するようにドメインの PDDS_D2 ビットをセットすると、D2 ドメインは DStandby モードを維持します。D2 ドメインでは、CPU が D2 ドメインにペリフェラルを割り当てる場合のみ、DStandby を終了します。***

DStandby モードを終了する際、ドメイン CPU およびペリフェラルはリセットされます。ただし、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングの状態は、システムモードに依存します。

- システムが STOP モードに移行しない場合、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングは、DStandby モードに移行する場合と同じです。
- システムが STOP モードまたは STANDBY モードに移行すると、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングはリセットされます。

D2 ドメインは CPU サブシステムによって DStandby モードを終了すると（最初のペリフェラルを割り当てる際、または D2 ドメインにペリフェラルが割り当てられ、CPU サブシステムが CStop モードを終了する際）、CPU ではドメインが DStandby モードを終了したことを検証します。正しい動作のためには、以下の手順に従うことを推奨します。

1. 最初に、ドメインバスマトリックスのクロックが使用可能であることを確認します。ドメインバスマトリックスのクロックの状態は、RCC_CR レジスタで確認できます。
   - RCC DnCKRDY = 0 の場合、ドメインバスマトリックスのクロックは停止されます。
   - RCC DnCKRDY = 1 の場合、ドメインバスマトリックスのクロックは有効になります。
2. 次に、ドメインが DStandbyモードを終了するまで 待ちます。このためには、*PWR CPU 制御レジスタ（PWR_CPUCR）* の SBF_Dn フラグを確認します。SBF_Dn をクリアした場合にのみ、ドメインに電源が供給され、アクセスできるようになります。以下にコードの例を示します。

```
Loop
write PWR SBF_Dn = 0 ; try to clear bit.
read PWR SBF_Dn
While 1 ==> loop
```

表 36. DStandby モード

DStandby モード	説明
モードへの移行	– ドメインの CPU サブシステムが CStop モードに移行します。 – CPU サブシステムの D2 ドメインにはペリフェラルが割り当てられており、CStop に移行します。 – CPU サブシステムでは、D2 ドメインの最後のペリフェラルを割り当て解除します。 – ドメインの PDDS_Dn ビットでは、STANDBY モードを選択します。 – 電源制御／ステータスレジスタ（PWR_WKUPFR）のすべての WKUPF ビットをクリアします。
モードの終了	– CPU サブシステムが CStop モードを終了します（*表 31* を参照）。 – CPU サブシステムの D2 ドメインにはペリフェラルが割り当てられており、CStop を終了します（*表 31* を参照）。 – CPU サブシステムでは、D2 ドメインの最初のペリフェラルを割り当てます。
ウェイクアップ遅延時間	EXTI および RCC ウェイクアップ同期 + ドメインパワーアップおよびリセット （*セクション 8.4.7 : パワーオンおよびウェイクアップのシーケンス*を参照）

6.7.10 STANDBY モード

STANDBY モードでは、消費電力を最も少なくできます。STOP モードと同様に、CPU サブシステムの CStop モードをベースにしています。ただし、V_{CORE} 電源レギュレータはオフになります。

システム D3 ドメインの STANDBY モードへの移行は、D1 および D2 ドメインが DStandby で動作する場合にのみ実行されます。システム D3 ドメインが STANDBY モードに移行すると、電圧レギュレータが無効になります。結果として、V_{CORE} ドメインの電源がすべてオフになります。PLL、HSI オシレータ、CSI オシレータ、HSI48、および HSE オシレータもオフになります。バックアップドメインのレジスタ（RTC レジスタ、RTC バックアップレジスタ、およびバックアップ RAM）と STANDBY 回路を除いて、SRAM とレジスタの内容は失われます（*セクション 6.4.4：バックアップドメイン*を参照）。

システム STANDBY モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）

  IWDG は、キーレジスタへのプログラミングによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません（*セクション 45：独立型ウォッチドッグ（IWDG）* の *セクション 45.3* を参照）。
- リアルタイムクロック（RTC）

  この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI RC）

  この設定は制御／ステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE OSC）

  この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

STANDBY モードへの移行

STANDBY モードへの移行は、すべてのドメインで *PWR CPU 制御レジスタ（PWR_CPUCR）* のすべての PDDS_Dn ビットが STANDBY を要求している場合に、*セクション 6.7.3：低電力モードへの移行* に従って実行されます。

STANDBY モードへの移行方法の詳細については、*表 38* を参照してください。

STANDBY モードの終了

STANDBY モードは、*セクション 6.7.4：低電力モードの終了* に従って終了します。

STANDBY モードの終了方法の詳細については、*表 38* を参照してください。

システムは外部リセット（NRST ピン）、IWDG リセット、WKUP ピンイベント、RTC アラーム、タンパイベント、またはタイムスタンプイベントのいずれかが検出されると、STANDBY モードを終了します。STANDBY モードからのウェイクアップ後、電源制御／ステータスレジスタ（*PWR 制御レジスタ 2（PWR_CR2）*、*PWR 制御レジスタ 3（PWR_CR3）*）、*PWR CPU 制御レジスタ（PWR_CPUCR）* の SBF ビット、*PWR ウェイクアップフラグレジスタ（PWR_WKUPFR）*、および *PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）* を除くすべてのレジスタがリセットされます。

STANDBY モードからのウェイクアップ後、プログラム実行はシステムリセット後と同様にリスタートされます（ブートオプションのサンプリング、ブートベクタリセットのフェッチなど）。*PWR CPU 制御レジスタ（PWR_CPUCR）* レジスタの SBF ステータスフラグに、システム終了時のモードを示します（*表 37* を参照）。

表 37. STANDBY および STOP フラグ

SBF_D2	SBF_D1	SBF	STOPF	説明
0	1	0	0	システムが RUN モードで実行中の場合に D1 ドメインは DStandby を終了します。
0	1	0	1	システムが STOP モードの場合、またはその終了時に、D1 ドメインは DStandby を終了します。
1	0	0	0	システムが RUN モードで実行中の場合に D2 ドメインは DStandby を終了します。
1	0	0	1	システムが STOP モードの場合、またはその終了時に、D2 ドメインは DStandby を終了します。
1	1	0	0	システムが RUN モードを維持している間に、D1 および D2 ドメインは DStandby を終了します。
1	1	0	1	システムが STOP モードの場合またはその終了時に、D1 および D2 ドメインは DStandby を終了します。
0	0	0	1	システムが STOP モードにある、またはこのモードを終了します。
0 (1)	0(1)	1	0	システムは STANDBY を終了します。

1. STANDBY を終了する際に、SBF_D1 と SBF_D2 はリセット値を反映します。

表 38. STANDBY モード

STANDBY モード	説明
モードへの移行	– CPU サブシステムが CStop モードで動作していて、アクティブな EXTI ウェイクアップソースはなく、RUN_D3 = 0 です。 – すべてのドメインのすべての PDDS_Dn ビットで STANDBY を選択します。 – 電源制御／ステータスレジスタ（PWR_WKUPFR）のすべての WKUPF ビットをクリアします。
モードの終了	– WKUP ピンの立ち上がり／立ち下がりエッジ、RTC アラーム（アラーム A およびアラーム B）、RTC ウェイクアップ、タンパイベント、タイムスタンプイベント、NRST ピンによる外部リセット、IWDG リセット
ウェイクアップ遅延時間	システムリセットフェーズ（*セクション 8.4.2 : システムリセット*を参照）

STANDBY モードにおける I/O の状態

STANDBY モードでは、以下のピンを除き、すべての I/O ピンがプルのないハイインピーダンス状態となります。

- リセットパッド（有効）
- タンパ、タイムスタンプ、RTC アラーム出力、または RTC クロック較正出力として設定されている場合の RTC_AF1 ピン
- WKUP ピン（有効な場合）。WKUP ピンのプル設定は、*PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）*の WKUPPUPD レジスタビットによって定義できます。

6.8 PWR レジスタの説明

PWR レジスタには、特に指定がない限り、ワード、ハーフワード、およびバイトフォーマットでアクセスできます。

6.8.1 PWR 制御レジスタ 1（PWR_CR1）

アドレスオフセット：0x000

リセット値：0xF000 C000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SVOS		Res.	Res.	Res.	Res.	FLPS	DBP	PLS			PVDE	Res.	Res.	Res.	LPDS
rw	rw					rw	rw	rw	rw	rw	rw				rw

ビット 31:19　予約済みであり、リセット値に保持する必要があります。

ビット 18:17 **ALS**：アナログ電圧検出器のレベル選択

これらのビットは、AVD によって検出される電圧閾値を選択します。

00：1.7 V

01：2.1 V

10：2.5 V

11：2.8 V

ビット 16 **AVDEN**：V_{DDA} ペリフェラル電圧モニタイネーブル

0：V_{DDA} のペリフェラル電圧モニタを無効化します。

1：V_{DDA} のペリフェラル電圧モニタを有効化します。

ビット 15:14 **SVOS**：システム STOP モード電圧スケーリング選択

消費電力と性能の間で最高のトレードオフを実現するために、これらのビットは V_{CORE} 電圧レベルをシステム STOP モードで制御します。

00：予約済み

01：SVOS5 スケール 5

10：SVOS4 スケール 4

11：SVOS3 スケール 3 （デフォルト）

ビット 13:10　予約済みであり、リセット値に保持する必要があります。

ビット 9 **FLPS**：DStop モードにおけるフラッシュ低電力モード

このビットにより、DStop モードの終了時に低い消費電力と再起動時間の間で最高のトレードオフを実現できます。

セットすると、D1 ドメインが DStop モードのときに、フラッシュメモリが低電力モードに移行します。

0：D1 ドメインの DStop への移行時に、フラッシュメモリは通常モードに留まります（高速再起動時間）。

1：D1 ドメインの DStop への移行時に、フラッシュメモリが低電力モードに移行します（消費電力の削減）。

ビット 8 **DBP**：バックアップドメイン書き込み保護の無効化

リセット状態において、RCC_BDCR レジスタ、RTC レジスタ（バックアップレジスタを含む）、PWR_CR2 レジスタの BREN および MOEN ビットは、不要な書き込みアクセスから保護されています。これらのレジスタへの書き込みアクセスを可能にするには、このビットをセットする必要があります。

0：RTC、RTC バックアップレジスタ、およびバックアップ SRAM へのアクセスを無効化します。

1：RTC、RTC バックアップレジスタ、およびバックアップ SRAM へのアクセスを有効化します。

ビット 7:5 **PLS**：プログラム可能な電圧検出器のレベル選択

これらのビットは、PVD によって検出される電圧閾値を選択します。

000：1.95 V

001：2.1 V

010：2.25 V

011：2.4 V

100：2.55 V

101：2.7 V

110：2.85 V

111：この場合、PVD_IN ピンの外部電圧レベルは内部 V_{REFINT} レベルと比較されます。

注：　詳細については、製品データシートの「電気的特性」セクションを参照してください。

ビット 4 **PVDE**：プログラム可能な電圧検出器有効化

0：プログラム可能な電圧検出器を無効化します。

1：プログラム可能な電圧検出器を有効化します。

ビット 3:1　予約済みであり、リセット値に保持する必要があります。

ビット 0 **LPDS**：SVOS3 での低電力ディープスリープの使用（SVOS4 および SVOS5 は、このビットの設定にかかわらず、常に低電力を使用）

0：SVOS3 が STOP モードに選択されている場合、電圧レギュレータはメインモード（MR）になります。

1：SVOS3 が STOP モードに選択されている場合、電圧レギュレータは低電力モード（LPR）になります。

6.8.2 PWR 制御ステータスレジスタ 1（PWR_CSR1）

アドレスオフセット：0x004

リセット値：0x0000 4000

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

ビット 31:17　予約済みであり、リセット値に保持する必要があります。

ビット 16 **AVDO**：V_{DDA} のアナログ電圧検出器の出力

このビットは、ハードウェアによってセット／クリアされます。AVDEN ビットによって V_{DDA} の AVD が有効化されている場合のみ有効です。

0：V_{DDA} が ALS[2:0] ビットで選択された AVD 閾値以上です

1：V_{DDA} が ALS[2:0] ビットで選択された AVD 閾値より低いです

注：　AVD は STANDBY モードでは無効になるため、このビットは STANDBY またはリセット後、AVDEN ビットがセットされるまで 0 になります。

ビット 15:14 **ACTVOS**：V_{CORE} 電圧スケーリング選択に現在適用されている VOS

これらのビットは、電圧レギュレータに適用された最後の VOS 値を反映します。

ビット 13 **ACTVOSRDY**：現在使用している VOS の電圧レベルレディビット

このビットは、電圧レギュレータが無効化され、PWR 制御レジスタ 3 （PWR_CR3）でバイパスモードが選択された場合に、ハードウェアによって 1 にセットされます。

0：電圧レベルが無効で、現在選択されている VOS のレベルを上回っているか下回っている

1：現在選択されている VOS のレベルで電圧レベルが有効である

ビット 12:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **PVDO**：プログラム可能な電圧検出器の出力

このビットは、ハードウェアによってセット／クリアされます。PVDE ビットによって PVD が有効化されている場合のみ有効です。

0：V_{DD} または PVD_IN 電圧が PLS[2:0] ビットで選択された PVD 閾値以上です

1：V_{DD} または PVD_IN 電圧が PLS[2:0] ビットで選択された PVD 閾値より低いです

注： ***PVD は STANDBY モードでは無効になるため、このビットは STANDBY またはリセット後、PVDE ビットがセットされるまで 0 になります。***

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

6.8.3 PWR 制御レジスタ 2（PWR_CR2）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタは、STANDBY モード、RESET 信号、および V_{DD} POR からのウェイクアップによってリセットされません。V_{SW} POR および VSWRST リセットによってのみリセットされます。

RCC_BDCR レジスタの VSWRST ビットによって V_{SW} ドメインをリセットしたときに、このレジスタにアクセスしないでください。

リセット後、PWR_CR2 レジスタは書き込み保護されます。内容を変更する前に、PWR_CR1 レジスタの DBP ビットをセットして書き込み保護を無効にする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	VBATH	VBATL	Res.	Res.	Res.	BRRDY
								r	r	r	r				r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
											rw				rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TEMPH**：温度レベルの監視と高閾値

0：高閾値を下回る温度

1：高閾値以上の温度

ビット 22 **TEMPL**：温度レベルの監視と低閾値

0：低閾値を上回る温度

1：低閾値以下の温度

ビット 21 **VBATH**：V_{BAT} レベルの監視と高閾値

0：高閾値レベルを下回る V_{BAT} レベル

1：高閾値レベル以上の V_{BAT} レベル

ビット 20 **VBATL**：V_{BAT} レベルの監視と低閾値

0：低閾値レベルを上回る V_{BAT} レベル

1：低閾値レベル以下の V_{BAT} レベル

ビット 19:17　予約済みであり、リセット値に保持する必要があります。

ビット 16 **BRRDY**：バックアップレギュレータレディ

このビットは、バックアップレギュレータがレディ状態であることを示すために、ハードウェアによってセットされます。

0：バックアップレギュレータはレディ状態ではありません。

1：バックアップレギュレータはレディ状態です。

ビット 15:5　予約済みであり、リセット値に保持する必要があります。

ビット 4 **MONEN**：V_{BAT} および温度監視有効化

このビットがセットされると、V_{BAT} 電源および温度監視が有効になります。

0：V_{BAT} および温度監視が無効になります。

1：V_{BAT} および温度監視が有効になります。

ビット 3:1　予約済みであり、リセット値に保持する必要があります。

ビット 0 **BREN**：バックアップレギュレータ有効化

このビットがセットされると、バックアップレギュレータ（STANDBY モードおよび V_{BAT} モードでバックアップ RAM の内容を保持するために使用）が有効になります。

BREN がリセットされるとバックアップレギュレータはオフになります。RUN モードおよび STOP モードではバックアップ RAM をそのまま使用できます。ただし、STANDBY モードおよび V_{BAT} モードではその内容は失われることになります。

BREN をセットすると、アプリケーションは、バックアップレギュレータレディフラグ（BRRDY）がセットされ、SRAM に書き込まれたデータが STANDBY モードおよび V_{BAT} モードにおいて保持されることが示されるのを待つ必要があります。

0：バックアップレギュレータを無効化します。

1：バックアップレギュレータを有効化します。

6.8.4　PWR 制御レジスタ 3（PWR_CR3）

アドレスオフセット：0x00C

リセット値：0x0000 0006（POR によってのみリセットされ、STANDBY モードおよびリセットパッドからのウェイクアップによってリセットされない）

このレジスタの下位バイトは、POR 後に一度書き込まれ、VOS レベルまたは ck_sys クロック周波数の変更前にも書き込まれます。上位バイトに制約はありません。

LDOEN および BYPASS ビットの無効な組み合わせに対応するプログラミングデータ（*表 26* を参照）は無視されます。データは書き込まれず、1 回だけ書き込むメカニズムによってレジスタはロックされ、またその他の書き込みアクセスも無視されます。デフォルトの電源設定は保持され、*PWR 制御ステータスレジスタ 1（PWR_CSR1）* の ACTVOSRDY ビットには引き続き無効な電圧レベルが示されます。新しい値を書き込む前に、システムを再起動します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	USB33RDY	USBREGEN	USB33DEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					r	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	Res.	Res.	SCUEN	LDOEN	BYPASS
						rw	rw						rw	rw	rw

ビット 31:27　予約済みであり、リセット値に保持する必要があります。

ビット 26 **USB33RDY**：USB 電源レディ
0：USB33 電源はレディ状態ではありません。
1：USB33 電源はレディ状態です。

ビット 25 **USBREGEN**：USB レギュレータ有効化
0：USB レギュレータを無効化します。
1：USB レギュレータを有効化します。

ビット 24 **USB33DEN**：$V_{DD33USB}$ 電圧レベル検出器有効化
0：$V_{DD33USB}$ 電圧レベル検出器を無効化します。
1：$V_{DD33USB}$ 電圧レベル検出器を有効化します。

ビット 23:10　予約済みであり、リセット値に保持する必要があります。

ビット 9 **VBRS**：V_{BAT} 充電抵抗の選択
0：5 kΩ の抵抗で V_{BAT} を充電します。
1：1.5 kΩ の抵抗で V_{BAT} を充電します。

ビット 8 **VBE**：V_{BAT} 充電有効化
0：V_{BAT} バッテリ充電を無効化します。
1：V_{BAT} バッテリ充電を有効化します。

ビット 7:3　予約済みであり、リセット値に保持する必要があります。

ビット 2 **SCUEN**：電源設定更新有効化
このビットは読み出し専用です。
0：電源設定更新がロックされています
1：電源設定に 1 回の書き込みが有効化されています（LDOEN および BYPASS）

ビット 1 **LDOEN**：低損失レギュレータ有効化
0：低損失レギュレータを無効化します。
1：低損失レギュレータを有効化します（デフォルト）。

ビット 0 **BYPASS**：電源管理ユニットバイパス
0：電源管理ユニットは通常動作
1：電源管理ユニットはバイパスされ、電圧監視は有効のままです。

6.8.5 PWR CPU 制御レジスタ（PWR_CPUCR）

このレジスタを使用して、CPU 電源を制御できます。

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RUN_D3	Res.	CSSF	SBF_D2	SBF_D1	SBF	SOPFF	Res.	Res.	PDDS_D3	PDDS_D2	PDDS_D1
				rw		rw	r	r	r	r			rw	rw	rw

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11　**RUN_D3**：CPU サブシステムモードにかかわらず、システム D3 ドメインを RUN モードのまま保持します。
0：D3 ドメインは CPU サブシステムモードに従います。
1：CPU サブシステムモードにかかわらず、D3 ドメインを RUN モードのまま維持します。

ビット 10　予約済みであり、リセット値に保持する必要があります。

ビット 9　**CSSF**：STANDBY および STOP フラグのクリア（常に 0 として読み出される）
このビットはハードウェアによって 0 にクリアされます。
0：影響はありません。
1：STOPF、SBF、SBF_D1、および SBF_D2 フラグをクリアします。

ビット 8　**SBF_D2**：D2 ドメイン DStandby フラグ
このビットはハードウェアによってセットされ、すべてのシステムリセットまたは CSSF ビットをセットすることでクリアされます。一度セットすると、このビットは D2 ドメインが DStandby モードを終了している場合にのみクリアできます。
0：D2 ドメインが DStandby モードではありません
1：D2 ドメインが DStandby モードです

ビット 7　**SBF_D1**：D1 ドメイン DStandby フラグ
このビットはハードウェアによってセットされ、システムリセットまたは CSSF ビットをセットすることでクリアされます。一度セットすると、このビットは D1 ドメインが DStandby モードを終了している場合にのみクリアできます。
0：D1 ドメインが DStandby モードでない場合
1：D1 ドメインが DStandby モードの場合

ビット 6　**SBF**：システム STANDBY フラグ
このビットはハードウェアによってセットされ、POR （パワーオンリセット）または CSSF ビットをセットすることでのみクリアされます。
0：システムが STANDBY モードでない場合
1：システムが STANDBY モードの場合

ビット 5　**STOPF**：STOP フラグ
このビットはハードウェアによってセットされ、すべてのリセットまたは CSSF ビットをセットすることでのみクリアされます。
0：システムが STOP モードではありません
1：システムが STOP モードです

ビット 4:3　予約済みであり、リセット値に保持する必要があります。

ビット 2 **PDDS_D3**：システム D3 ドメインのパワーダウンディープスリープ
このビットにより、システム D3 ドメインのディープスリープモードを定義できます。
0：D3 ドメインのディープスリープへの移行時に、STOP モードを保持します。
1：D3 ドメインのディープスリープへの移行時に、STANDBY モードを保持できます。

ビット 1 **PDDS_D2**：D2 ドメインのパワーダウンディープスリープ
このビットにより、D2 ドメインのディープスリープモードを定義できます。
0：D2 ドメインのディープスリープへの移行時に、DStop モードを保持します。
1：D2 ドメインのディープスリープへの移行時に、DStandby モードを保持できます。

ビット 0 **PDDS_D1**：D1 ドメインのパワーダウンディープスリープ選択
このビットにより、D1 ドメインのディープスリープモードを定義できます。
0：D1 ドメインのディープスリープへの移行時に、DStop モードを保持します。
1：D1 ドメインのディープスリープへの移行時に、DStandby モードを保持できます。

6.8.6 PWR D3 ドメイン制御レジスタ（PWR_D3CR）

このレジスタを使用して、D3 ドメイン電源を制御できます。

アドレスオフセット：0x018

リセット値：0x0000 4000（ソフトウェアは、以下のVOSRDYのリセット値 を 1 として読み出します）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VOS		VOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw		r													

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **VOS**：性能に応じた電圧スケーリング選択
これらのビットは、V_{CORE} 電圧レベルを制御し、消費電力と性能の間で最高のトレードオフを実現します。
– 性能を向上するには、システム周波数を増加する前に電圧スケーリングを変更します。
– 性能を低下させるには、最初にシステム周波数を低減してから電圧スケーリングを変更します。
00：予約済み（スケール 3 が選択される）
01：スケール 3（デフォルト）
10：スケール 2
11：スケール 1

ビット 13 **VOSRDY**：V_{CORE} 電圧スケーリング出力選択の VOS レディビット
このビットは、PWR 制御レジスタ 3（PWR_CR3）でバイパスモードが選択された場合に、ハードウェアによって 1 にセットされます。
0：動作可能状態ではありません。電圧レベルが選択されている VOS のレベルを下回っています。
1：動作可能状態です。電圧レベルが選択されている VOS のレベル以上です。

ビット 12:0 予約済みであり、リセット値に保持する必要があります。

6.8.7 PWR ウェイクアップクリアレジスタ（PWR_WKUPCR）

アドレスオフセット：0x020

リセット値：0x0000 0000 （システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされません）

このレジスタを書き込むには、5 つのウェイトステートが必要です（PWR_WKUPFR の WKUPF ビットをクリアする場合、AHB 書き込みアクセスは WKUPF をクリアした後で完了します）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPC6	WKUPC5	WKUPC4	WKUPC3	WKUPC2	WKUPC1
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:6　予約済み、常に 0 として読み出されます。

ビット 5:0　**WKUPCn+1**：WKUPn+1 のウェイクアップピンフラグのクリア

これらのビットは常に 0 として読み出されます。

0：影響なし。

1：このビットに 1 を書き込むと WKUPFn+1 のウェイクアップピンフラグがクリアされます（このビットはハードウェアによって 0 にクリアされます）。

6.8.8 PWR ウェイクアップフラグレジスタ（PWR_WKUPFR）

アドレスオフセット：0x024

リセット値：0x0000 0000 （システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされない）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPF6	WKUPF5	WKUPF4	WKUPF3	WKUPF2	WKUPF1
										r	r	r	r	r	r

ビット 31:6　予約済みであり、リセット値に保持する必要があります。

ビット 5:0　**WKUPn+1**：ウェイクアップピン WKUPn+1 フラグ

このビットはハードウェアによってセットされ、リセットピンまたは*PWR ウェイクアップクリアレジスタ（PWR_WKUPCR）*の WKUPCn+1 ビットをセットすることでのみクリアされます。

0：ウェイクアップイベントは発生していません。

1：WKUPn+1 ピンからウェイクアップイベントを受信しました。

6.8.9 PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）

アドレスオフセット：0x028

リセット値：0x0000 0000 （システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされない）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	WKUPPUPD6		WKUPPUPD5		WKUPPUPD4		WKUPPUPD3		WKUPPUPD2		WKUPPUPD1	
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	WKUPP6	WKUPP5	WKUPP4	WKUPP3	WKUPP2	WKUPP1	Res.	Res.	WKUPEN6	WKUPEN5	WKUPEN4	WKUPEN3	WKUPEN2	WKUPEN1
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **WKUPPUPD[truncate(n/2)-7)**：WKUP(truncate(n/2)-7) のウェイクアップピンのプル設定

これらのビットは、WKUPEN(truncate(n/2)-7) = 1 の場合に使用される I/O パッドのプル設定を定義します。関連する GPIO ポートのプル設定は、同じ値にセットするか、「00」にセットします。

ウェイクアップピンのプル設定は、STANDBY モードのまま保持されます。

00：プルアップなし

01：プルアップ

10：プルダウン

11：予約済み

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **WKUPPn-7**：WKUPn-7 のウェイクアップピン極性ビット

これらのビットは、WKUPn-7 外部ウェイクアップピンのイベントの検出に使用される極性を定義します。

0：高レベルで検出します（立ち上がりエッジ）。

1：低レベルで検出します（立ち下がりエッジ）。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WKUPENn+1**：ウェイクアップピン WKUPn+1 有効化

各ビットは、ソフトウェアによってセット／クリアされます。

0：WKUPn+1 ピンでイベントが発生しても、システムは STANDBY モードからウェイクアップしません。

1：WKUPn+1 ピンで立ち上がり／立ち下がりエッジが発生し、システムを STANDBY モードからウェイクアップさせます。

注： ***WKUPPn+1 の立ち上がりエッジ選択時に WKUPn+1 ピンのレベルがすでに高い場合、または WKUPPn+1 の立ち下がりエッジ選択時にこのピンのレベルが低い場合、WKUPENn+1 ビットをセットして WKUPn+1 ピンが有効化されると、追加のウェイクアップイベントが検出されます。***

6.8.10 PWR レジスタマップ

表 39. 電源制御レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	PWR_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALS		AVDEN	SVOS		Res.	Res.	Res.	Res.	FLPS	DBP	PLS			PVDE	Res.	Res.	Res.	LPDS
	リセット値														0	0	0	1	1					0	0	0	0	0	0				0
0x004	PWR_CSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AVDO	ACTVOS		ACTVOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDO	Res.	Res.	Res.	Res.
	リセット値																0	0	1	0									0				
0x008	PWR_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	VBATH	VBATL	Res.	Res.	Res.	BRRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
	リセット値									0	0	0	0				0												0				0
0x00C	PWR_CR3	Res.	Res.	Res.	Res.	Res.	USB33RDY	USBREGEN	USB33DEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	Res.	Res.	SCUEN	LDOEN	BYPASS
	リセット値						0	0	0															0	0						1	1	0
0x010	PWR_CPUCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RUN_D3	Res.	CSSF	SBF_D2	SBF_D1	SBF	STOPF	Res.	Res.	PDDS_D3	PDDS_D2	PDDS_D1
	リセット値																					0		0	0	0	0	0			0	0	0
0x014	予約済み	予約済み																															
	リセット値																																
0x018	PWR_D3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VOS		VOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																	0	1	0													
0x020	PWR_WKUPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPC6	WKUPC5	WKUPC4	WKUPC3	WKUPC2	WKUPC1
	リセット値																											0	0	0	0	0	0
0x024	PWR_WKUPFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPF6	WKUPF5	WKUPF4	WKUPF3	WKUPF2	WKUPF1
	リセット値																											0	0	0	0	0	0
0x028	PWR_WKUPEPR	Res.	Res.	Res.	Res.	WKUPPUPD6		WKUPPUPD5		WKUPPUPD4		WKUPPUPD3		WKUPPUPD2		WKUPPUPD1		Res.	Res.	WKUPP6	WKUPP5	WKUPP4	WKUPP3	WKUPP2	WKUPP1	Res.	Res.	WKUPEN6	WKUPEN5	WKUPEN4	WKUPEN3	WKUPEN2	WKUPEN1
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x030	予約済み																																

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

7 低消費電力 D3 ドメイン

このセクションでは、D3 ドメインを使用して低消費電力アプリケーションを実装する方法を例示します。

7.1 概要

前半では、EXTI、RCC、および PWR のブロック間の、または他のシステムブロックとの相互作用について説明します。DMAMUX2 を使用して CPU 負荷を軽減する方法についても詳しく説明します。

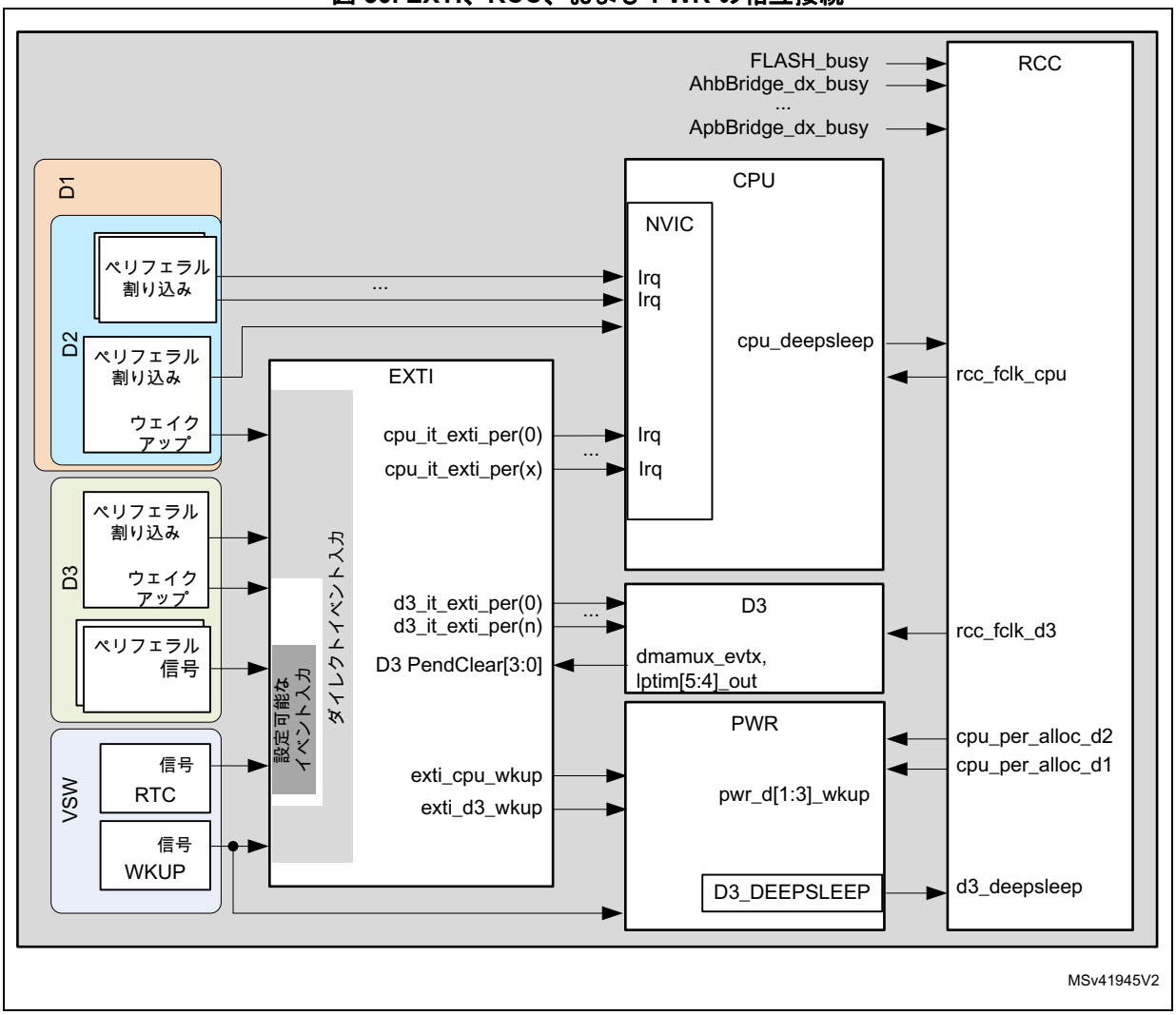
後半では、LINUART1 送信を例に、自律モードを使用した単純なデータ転送方法について説明します。

レジスタのプログラミングは、自律モードに関連するブロックについてのみ詳細に説明します。

7.2 EXTI、RCC、および PWR の相互接続

図 30 に主要な EXTI、RCC、および PWR の相互接続を示します。

図 30. EXTI、RCC、および PWR の相互接続



7.2.1 割り込みとウェイクアップ

3 種類の信号がペリフェラル間で交換されます。これらを使用して、システムを STOP モードからウェイクアップすることができます。

- **ウェイクアップイベント**（または非同期割り込み）

  一部のペリフェラルでは、バスインタフェースクロックが存在しなくても割り込みイベントを生成できます。これらの割り込みイベントは、ウェイクアップイベント（または非同期割り込み）と呼ばれます。

  例：i2c1_wkup、usart1_wkup、および lptim1_wkup

- 信号

  一部のペリフェラルでは、割り込み信号の代わりにパルスを生成します。これらのパルスは、信号と呼ばれます。

  例：lptim2_out および lptim3_out

- 割り込み

  信号と異なり、CPU またはその他のバスマスタで、割り込みをクリアする必要があります。これには、ペリフェラルレジスタの対応するイベントビットをクリアするか、FIFO 割り込みレベルを更新します。

  システムペリフェラルに関連付けられた割り込みは、すべて直接 NVIC に接続されています。ただし、システムを STOP モードから、または CPU を CStop からウェイクアップすることができるペリフェラルは除きます。後者の場合、割り込み、信号、またはウェイクアップイベントは EXTI を通じて NVIC に接続されています。

  例：spi1_it、tim1_brk_it、および tim1_upd_it

ペリフェラル自体からクリアする必要がある割り込みやウェイクアップソースは EXTI ダイレクトイベント入力に接続されています。EXTI では CPU ステータスペンディングビットを管理することはありません。

ペリフェラル信号は、EXTI の設定可能なイベント入力に接続されています。これらの EXTI 入力により、アプリケーションでクリアする必要がある CPU ステータスペンディングビットが提供されます。

7.2.2 ブロックの相互作用

EXTI および PWR ブロック間の相互作用

EXTI では、PWR コントローラにウェイクアップリクエスト信号（**exti_c_wkup**、**exti_d3_wkup**）を送信します。これらの信号は、EXTI に接続された割り込み、信号、またはウェイクアップイベントの状態に応じて有効化されます。これらのウェイクアップリクエストはPWRコントローラが使用し、ペリフェラルによって生成されたウェイクアップイベントを処理する必要があるドメインに電源を供給します。

PWR および RCC ブロック間の相互作用

PWR ブロックでは、システムの動作モード（CRun、CSleep、または CStop）に合わせて V_{DD11} 電源を制御します。PWR ブロックでは、V_{DD11} 電源を D1 および D2 ドメインに供給する電源スイッチ（ePOD）も制御します。

RCC ブロックでは、システムの動作モードに合わせてクロック生成を制御します。また、リセット生成も制御します。

システムモードの遷移を同期するため、RCC ブロックを PWR コントローラと密接に結合しています。

- CPU （**c_per_alloc_d2**、**c_per_alloc_d1**）によってDxドメインに配置されたペリフェラルが割り当てられると、RCCはPWR コントローラに通知します。
- RCC では、ドメインクロックが有効／無効になると、PWR ブロックに警告を発します。これらの信号は、DRun から DStop または DStandby にドメインが遷移する場合に使用されます。この場合、PWR コントローラはドメインクロックが停止されるまで待ってから、このドメインをオフにします。
- 同様に、PWR コントローラは各ドメイン（**pwr_d[1:3]_wkup**）の V_{DD11} 電源状態を RCC に通知します。この情報は、ひとつのドメインがDStop または DStandby から DRun へ遷移したとき、RCC によって使用されます。

EXTI および D3 ドメイン間の相互作用

D3 ドメインのペリフェラルからEXTIが 受信するすべてのウェイクアップイベント入力は、システムクロックの再同期後に D3 ドメインに転送されます。これらのイベントは、CPU を有効化せずに自律動作を実行するために、D3 ドメインによって使用されます。

D3 ドメインから受信された EXTI **D3_PenClear[3:0]** 入力は、D3 ドメインのペリフェラルによって生成された実行中のウェイクアップリクエストを確認するために使用されます。**D3_PenClear[3:0]** 入力で、システム D3 ドメインをRunからStopへ切り替えられます。

7.2.3 D3 ドメイン DMAMUX2 の役割

D3 ドメインに実装された DMAMUX2 では、BDMA 転送を連鎖させることができます。BDMA リクエストはトリガイベント（**dmamux2_evtx**）によって同期することができ、そのトリガイベントは期待データ量の転送が完了したときに生成されます。

これらのイベントによって DMAMUX2 リクエストジェネレータ（REQ_GEN[3:0]）もトリガすることができます。これにより、複数の BDMA 転送を連鎖させることができます。実際、REQ_GEN[3:0] は、すべての D3 ドメインペリフェラルによって生成されたすべてのウェイクアップイベントにより、間接的にトリガされます。

LPTIM5 および LPTIM4 出力のように、dmamux2_evt7 および dmamux2_evt6 イベントは EXTI に接続されています。これらを使用して、ウェイクアップイベントによって要求されたタスクが完了した後で、D3 ドメインを DRun から DStop モードに切り替えることができます。

7.3 LPUART1 送信に基づく低消費電力アプリケーションの例

このセクションでは、消費電力の観点で D3 ドメインを使用する利点を例示します。ユーザによるデバイスのプログラミングを支援するため、ここではキーレジスタ設定のみを示します。

詳細については、*セクション 8：リセットおよびクロック制御（RCC）*および*セクション 6：電源制御（PWR）*を参照してください。

7.3.1 メモリ保持

D3 ドメインには、64 KB の SRAM （SRAM4）があり、D1 および D2 ドメインが DStandby モードに移行する際にデータの保持に使えます。

この機能は、以下の場合にいくつかの使用例で使用されます。

- DStandby から適切に復帰するために、アプリケーションコードを保持するため
- D1またはD2ドメインがDstandby状態で、2 つの連続した処理と処理の間でCPU が CStop に移行した場合に、センサと送受信するデータを保持するため

注： ***SRAM4 は、システムが STANDBY モードに移行しない限り、使用可能な状態で維持されます。***

システムが STANDBY モードの場合は、引き続き BKUP_SRAM を使用できます。ただし、サイズは 4 KB に制限されます。

7.3.2 LPUART1 インタフェースを使用したメモリからペリフェラルへの転送

例

図 31 に、実装例を示します。LPTIM4 に設定した一定の時間間隔で、CPU が CStop モード（ドメインは DStandby）からウェイクアップしますCPU が RUN モードの場合、LPUART1 を介して送信するデータを準備し、これを SRAM4 に転送し、その後 CStop に戻ります。D3 ドメインは、LPUART1 を介してデータ転送を実行し、転送完了後に STOP モードに戻るよう設定されています。

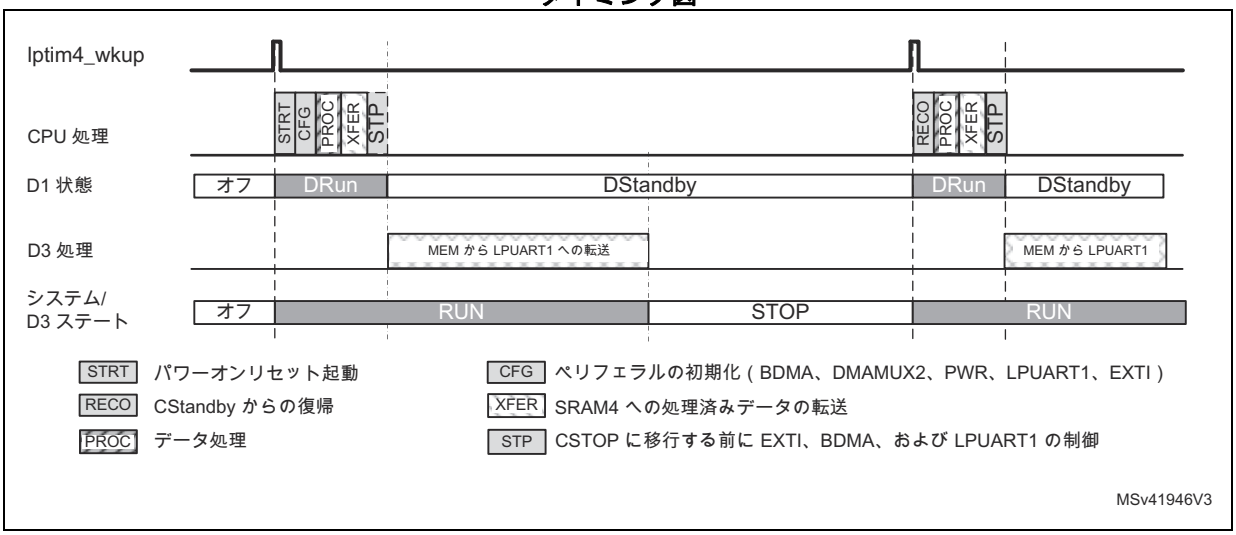
LPTIM4 インタフェースは、一定の時間間隔でシステムを STANDBY からウェイクアップするために使用します。
その後、CPU は次の操作を実行する必要があります。

1. システム STANDBY モードからアプリケーションを復帰します（RECO）。
2. 新しいデータを LPUART1 を介して送信できるように処理します（PROC）。
3. SRAM4 にデータを転送します（XFER）。
4. DMAMUX2、BDMA、LPUART1、および RCC を設定します（CFG）。
5. EXTI を設定します（CFG）。
6. D1 ドメインが DStandby モードに移行できるように PWR ブロックを設定します（STP）。
7. CPU を STOP モードにセットします。

D3 ドメインでは、次のタスクを自律モードで実行します。

1. BDMA を使用して、SRAM4 から LPUART1 にデータを転送します。
2. LPUART1 インタフェースに最後のバイトが転送されたことが通知されると、D3 ドメインは STOP モードに切り替わります。

図 31. BDMA および D3 ドメインによる自律モードでの SRAM4 から LPUART1 への転送を示すタイミング図



注： ***このセクションで説明する例では、PWR_CPUCR レジスタの RUN_D3 ビットを使用して D1 および D2 ドメインが DStop/DStandby 状態のとき、D3 ドメインを RUN モードのまま保持することはできません。RUN_D3 により D3 ドメインは強制的に RUN モードになりますが、独自に STOP に戻ることはできなくなります。***

D3 ドメインについてアプリケーションで STOP モードと RUN モードを切り替える必要がある場合は、RUN モードをウェイクアップイベントでトリガする必要があります。これは、D3 ドメインが必要に応じてこのイベントをクリアできるようにするためです。

RCC プログラミング

この例で、CPU サブシステムには、データ転送に使用される D3 ドメインのペリフェラル（BDMA、DMAMUX2、LPUART1、および LPTIM4）が含まれます。これらのペリフェラルは、CPU が CSTOP モードの場合でも動作するよう、自律モードでプログラムする必要があります。

LPUART1 では、独自の APB クロックをカーネルクロックとして使用できます。システムは LPUART1 がデータ転送を完了するまで STOP モードに移行できないため、PLLx を使用してペリフェラルにクロックを供給できます。

PWR プログラミング

この例では、以下のために PWR ブロックをプログラムする必要があります。

- データ転送の完了後にシステム D3 ドメインの STANDBY モードへの移行を回避するため
- D1 ドメインを DStandby モードに移行できるようにするため
- システムモードに応じて動作電圧を定義するため

注： ***D3 ドメインは STANDBY に移行することもできますが、その場合、LPTIM4 を使用してシステムをウェイクアップすることはできず、代わりに AWU を使用する必要があります。さらに、システムのウェイクアップ時にすべてを再プログラムする必要があります。***

EXTI プログラミング

次のサービスを提供するには、EXTI ブロックを設定する必要があります。

- D1 ドメインが DStandby の場合に、D3 ドメインを実行中のまま保持します。この設定はソフトウェアイベントで行われます。
- LPUART1 を介したデータ転送の完了後に、デバイスを STOP モードにセットします。
- LPTIM4 に設定した時間が経過した後で、マイコンを STOP からウェイクアップします。

最初のデータ転送を実行する前に、一度 EXTI ブロックが設定されます。着信データ転送向けにプログラムされた設定は変更されません。一部のイベントのみをトリガまたは確認する必要があります。

注： ***CPU ではソフトウェアイベントを生成するためにイベント入力番号 0 を使用します。LPTIM4 ウェイクアップ信号は、イベント入力番号 52 に接続されています（ダイレクトイベント入力）。***

他のすべてのイベント入力は無効でなければなりません。EXTI_RTSRx_TRy = '0'、EXTI_FTSRx_TRy = '0' にセットします。

D3 ドメインのウェイクアップイベントを生成するために、CPU では EXTI_SWIER1 の SWIER0 ビットに「1」を書き込む必要があります。誤ったウェイクアップを避けるため、このイベント入力に接続された GPIO をトグルしないでください。GPIO の影響を受けないようにするには、次のシーケンスを実行します。

BDMA および DMAMUX2 プログラミング

LPUART1 を介したデータ転送を実行するには、2 つの BDMA チャネルが必要です。

- チャネル 0 などの BDMA チャネルは、TXE フラグを使用して、SRAM4 から LPUART1 にデータを転送するために使用します。
- 2 つ目の BDMA チャネルの役割は、D3 ドメインを STOP モードに切り替えることです。このため、DMAMUX2 リクエストジェネレータのチャネル 0 （REQ_GEN0）および DMAMUX2 のチャネル 7 の同期ブロック（SYNC7）は、BDMA のチャネル 7 と組み合わせて使用されます。

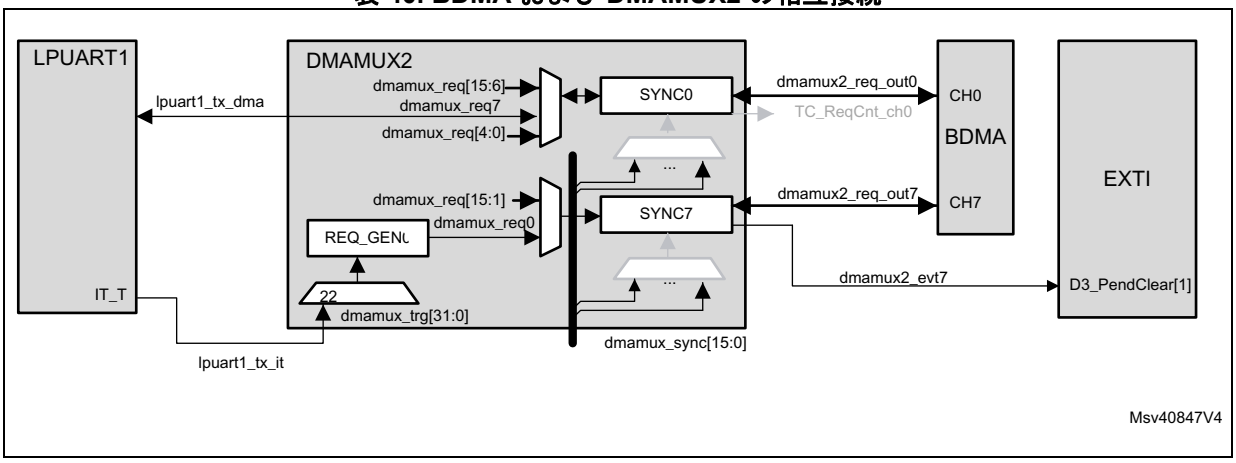
BDMA のチャネル 0 では DMAMUX2 のトリガ機能を使用しません。初期化の詳細については、*表 41* を参照してください。

BDMA のチャネル 7 では、BDMA リクエストを生成するために REQ_GEN0 を使用します。BDMA リクエストの生成は、LPUART1 送信割り込み（**lpuart1_tx_it**）によってトリガされます。LPUART1 インタフェースでは、送信完了イベントを検出すると、lpuart1_tx_it 割り込みを生成します。次に、BDMA では LPUART1 への書き込み動作を実行することで、ペンディング割り込みをクリアします。

SYNC7 ブロックは、フリーランニングモードでプログラムされます。これにより、REQ_GEN0 によって生成された BDMA リクエストが完了したときに、**dmamux2_evt7** 出力にパルスが生成されます。EXTI では **dmamux2_evt7** 信号を使用して、D3 ドメインを STOP モードに戻します。

図 40 に、DMAMUX2 を介した有効な信号パスを示します。灰色のブロックは使用されていないパスを表します。

表 40. BDMA および DMAMUX2 の相互接続



BDMA および DMAMUX2 の主要な機能のプログラム方法を、*表 41* で説明します。エラーの処理方法については説明していません。

表 41. BDMA および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT）

ペリフェラル	レジスタの内容	関連アクション
DMAMUX2 SYNC0	DMAMUX2_C0CR の DMAREQ_ID = ‘7’ DMAMUX2_C0CR の SE = ‘0’ DMAMUX2_C0CR の EGE = ‘0’ DMAMUX2_C0CR の NBREQ = ‘0’	BDMA リクエストとして LPUART_TX を選択します。 ブロック同期を無効化します。 イベント生成を無効化します。 BDMA 転送のたびにイベントを生成します（フリーランニングモード）。
DMAMUX2 SYNC7	DMAMUX2_C7CR の DMAREQ_ID = ‘0’ DMAMUX2_C7CR の SE = ‘0’ DMAMUX2_C7CR の EGE = ‘1’ DMAMUX2_C7CR の NBREQ = ‘0’	BDMA リクエストとして REQ_GEN0 を選択します。 ブロック同期を無効化します。 イベント生成を有効化します。 BDMA 転送のたびにイベントを生成します（フリーランニングモード）。
DMAMUX2 REQ_GEN0	DMAMUX2_RG0CR の SIG_ID = ‘0d24’ DMAMUX2_RG0CR の GPOL = ‘0b01’ DMAMUX2_RG0CR の GNBREQ = ‘0’ DMAMUX2_RG0CR の GE = ‘1’	トリガとして LPUART TX 割り込みを選択します。 エベントの立ち上がりエッジをトリガします。 BDMA リクエストを 1 つだけ生成します。 ジェネレータを有効化します。

表 41. BDMA および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT）（続き）

ペリフェラル	レジスタの内容	関連アクション
BDMA - CH0	BDMA_CNDTR0 の NDT ビット = DatNber BDMA_CPAR0 の PA = &LPUART1_TDR BDMA_CMAR0 の MA = &DatBuff BDMA_CCR0 の DIR = ‘1’ BDMA_CCR0 の CIRC = ‘0’ BDMA_CCR0 の PINC = ‘0’ BDMA_CCR0 の MINC = ‘1’ BDMA_CCR0 の PSIZE = ‘0’ BDMA_CCR0 の MSIZE = ‘1’ BDMA_CCR0 の MEM2MEM = ‘0’	転送するデータの数を設定します。 LPUART1_TDR のアドレスを設定します。 SRAM4 のメモリバッファのアドレスを設定します。 メモリから読み出します。 サーキュラモードは無効です。 ペリフェラルインクリメントは無効です。 メモリインクリメントは有効です。 ペリフェラルサイズは 8 ビットに設定します。 メモリサイズは 8 ビットに設定します。 メモリ間転送モードは無効です。
BDMA - CH7	BDMA_CNDTR7 の NDT ビット = ‘1’ BDMA_CPAR7 の PA = &LPUART1_ICR BDMA_CMAR7 の MA = &DatClrTC BDMA_CCR7 の DIR = ‘1’ BDMA_CCR7 の CIRC = ‘0’ BDMA_CCR7 の PINC = ‘0’ BDMA_CCR7 の MINC = ‘1’ BDMA_CCR7 の PSIZE = 2 BDMA_CCR7 の MSIZE = 2 BDMA_CCR7 の MEM2MEM = ‘0’	1 つのデータのみを転送します。 LPUART1_ICR のアドレス（割り込みフラグクリアレジスタ）を設定します。 SRAM4 に配置された変数のアドレスを設定します。TC フラグをクリアするために、この変数には 0x0040 が含まれている必要があります。 メモリから読み出します。 サーキュラモードは無効です。 ペリフェラルインクリメントは無効です。 メモリインクリメントは無効です。 ペリフェラルサイズは 32 ビットに設定します。 メモリサイズは 32 ビットに設定します。 メモリ間転送モードは無効です。

LPTIM4 プログラミング

LPTIM4 ウェイクアップイベントが発生した場合、CPU は再起動され、D3 ドメインモードも RUN モードにセットされます。

LPTIM4 によって発行された割り込みは CPU の NVIC でペンディング中です。LPTIM4 割り込みハンドラでは、LPTIM4_ICR レジスタの ARRMCF ビットに「1」を書き込んで、この LPTIM4 割り込みを承認する必要があります（LPTIM4_Ack）。

LPUART プログラミング

この使用例の説明では、一部のイベントに従ってカーネルクロックを要求する LPUART1 の機能は使用しません。

LPUART1 は、TX-FIFO がフルでない場合に BDMA リクエストを生成するようにプログラムされます。

また、LPUART1 では、TX-FIFO とその送信シフトレジスタが空の場合に割り込みを生成します。この割り込みは、D3 ドメインを STOP モードに切り替えるために使用されます。

表 42 に、LPUART1 の STOP モードの扱いに関する主要な設定を示します。

表 42. LPUART1 初期プログラミング（LPUART1_INIT）

レジスタの内容	関連アクション
LPUART1_CR1 の FIFOEN = ‘1’	FIFO を有効化します。次に、BDMA は、TXFNF （TXFIFO Not Full）フラグをBMDAリクエストを生成するために使います。
LPUART1_CR1 の TCIE = ‘0’	送信バッファが空のとき、割り込みを無効化します。
LPUART1_CR1 の UE = ‘1’	BDMA を有効化します。
LPUART1_CR1 の TE = ‘1’	LPUART1 を有効化します。
LPUART1_CR1 の TXE = ‘1’	転送を有効化します。
LPUART1_CR3 の DMAT = ‘1’	転送のために BDMA モードを有効化します。

LPUART1 を有効にするには、*表 43* に説明するシーケンスに従ってください。

表 43. LPUART1 初期プログラミング（LPUART1_Start）

レジスタの内容	関連アクション
LPUART1_ICR の TCCF = ‘1’	すぐに割り込みが生成されないよう TC フラグをクリアし、これによって EXTI の D3_PendClear[1] もクリアされます。
LPUART1_CR1 の TCIE = ‘1’	送信バッファが空のとき、割り込みを有効化します。

7.3.3 LPUART1 送信に基づく低消費電力アプリケーションの例の全体説明

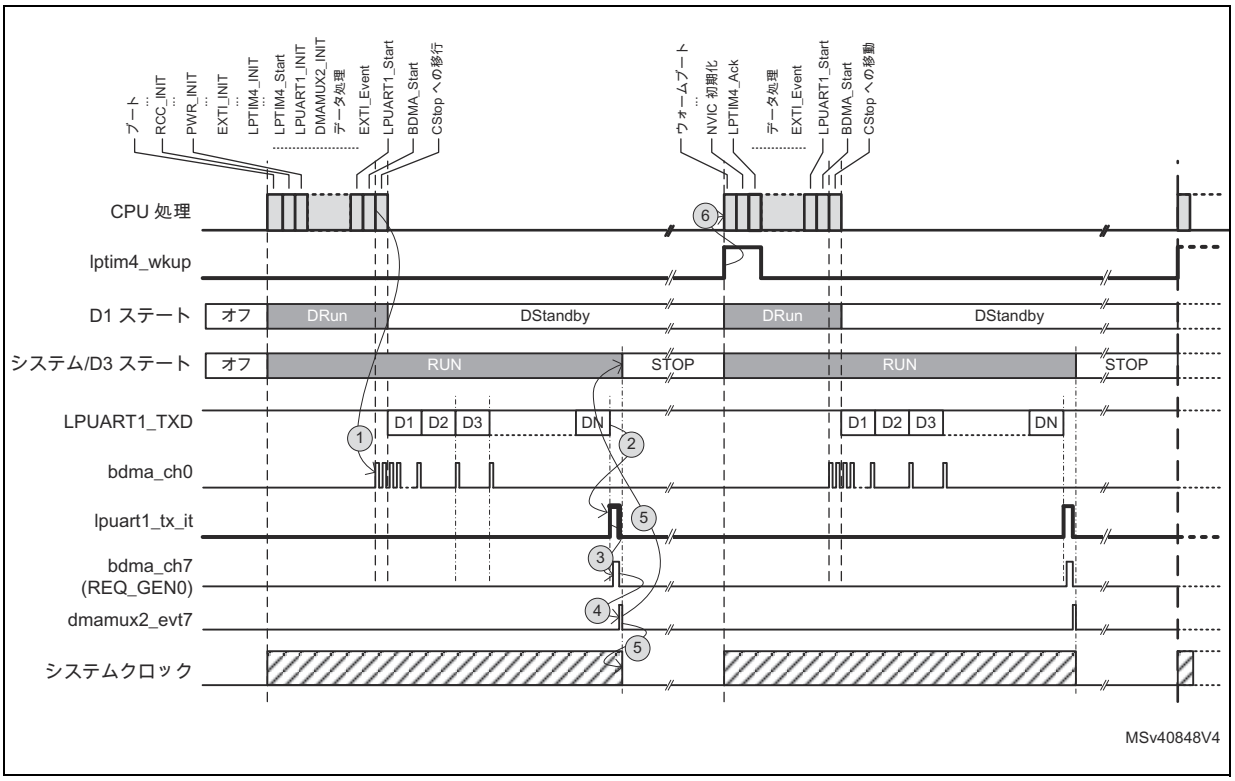
パワーオンリセット後、CPU では以下の動作を実行します。

1. ブートシーケンス（ここでは説明しません）。
2. RCC、PWR、EXTI、LPUART1、GPIOs、LPTIM4、DMAMUX2、BDMA、および NVIC の完全な初期化。
   ここでは、自律モードに関連する RCC、EXTI、PWR、LPUART1、BDMA、および DMAMUX2 の初期化の該当するステップについてのみ説明します。詳細については、前のセクションを参照してください。
3. CPU では転送するデータを処理し、SRAM4 にコピーします。
4. CPU では、D1 が DStandby に移行したときに D3 を RUN モードのまま保持するため、ウェイクアップイベント（EXTI_Event）を生成します。
5. CPU では LPUART 送信を開始するために BDMA を有効にし、STOP モードに移行します。許可されているように、D3 で RUN モードを維持しながら、D1 ドメインは DStandby に移行します。D1 ドメインが DStandby モードの間、SRAM4 に保管されたデータは維持されます。
6. BDMA が有効になると、TX-FIFO を満たすために LPUART1 からのリクエストを処理します。並行して、シリアルデータ転送を開始することができます。
7. 期待されるデータ量が送信されると（BDMA_CNDTR0 の NDT ビットを 0 にセット）、BDMA では LPUART1 へのデータ供給を停止します。LPUART1 では、TX-FIFO と送信バッファが空の場合に割り込みを生成します。
8. この割り込みにより DMAMUX2 の REQ_GEN0 がトリガされ、BDMA のチャネル 7（BDMA_Ch7）を介したデータ転送が有効になります。この転送により LPUART1 の TC フラグはクリアされ、**lpuart1_tx_it** は「0」にリセットされます。
9. この転送の終了によりdmamux2_evt7 信号がトリガされ、CPU が生成したウェイクアップリクエストをクリアします。
10. 結果として、D3 ドメイン（システム）は STOP モードに移行し、システムクロックは停止します。LPTIM4 は **ck_lsi** クロックを使用しているため、動作を続けます。

11. LPTIM4 の **lptim4_wkup** 割り込みにより、システムがウェイクアップします。デバイスは、HSI クロックによって STOP モードを終了します。CPU ではウォームリブートシーケンス中に正しいクロック構成を復元し、以下のタスクを実行する必要があります。
    d) LPTIM4 ウェイクアップ割り込みを確認します。
    e) 次のデータブロックを処理し、SRAM4 に転送します。
    f) 再度 D3 ドメインのウェイクアップイベントを生成します。
    g) BDMA を開始します。
    h) CStop モードに戻ります。

注： ***CPU では再度 BDMA、DMAMUX2、および LPUART1 を初期化する必要はありません。***

図 32. D3 ドメインによる自律モードでの LPUART1 送信を示すタイミング図



7.3.4 代替実装

より電力効率良く実装することも可能です。たとえば、LPUART1のTX-FIFOに送信データを入れた後すぐにシステムクロックを停止でき、上述の例のようにLPUART1の送信が完了するまでクロック供給を継続する必要はありません。この場合、LPUART1 ではシステムを RUN から STOP モードに切り替える際に、ck_hsi または ck_csi をカーネルクロックとして使用する必要があります。LPUART1 は、TX-FIFO がほぼ空になった場合に、D3 ドメインをウェイクアップするようにプログラムする必要があります。この非同期割り込みは、DMAMUX2 の REQ_GENx でトリガとして使用できます。これにより LPUART1_TDR へのデータ転送が一定数（例：14）行われ、その後 D3 ドメインが STOP モードに戻されます。TX-FIFO と送信バッファが空でない限り LPUART1 ではカーネルクロックを要求できるため、このように実装することができます。

参考資料

7.4 その他の低消費電力アプリケーション

D3 ドメインにあるその他のペリフェラル（I2C4、SPI6、SAI4、または ADC3 など）を使用して、低消費電力アプリケーションを実装することができます。

参考資料

8 リセットおよびクロック制御（RCC）

RCC ブロックでは、マイクロコントローラ全体のクロック生成とリセット生成を管理します。

RCC ブロックは D3 ドメインにあります（詳細については、*セクション 6：電源制御（PWR）*を参照）。

このセクションで参照する動作モードは、PWR ブロックの *セクション 6.6.1：動作モード*で定義されます。

8.1 RCC の主な機能

リセットブロック

- ローカルリセットとシステムリセットの生成
- マイクロコントローラまたは外部デバイスをリセットする双方向ピンのリセット
- ホールドブート機能
- WWDG リセットのサポート

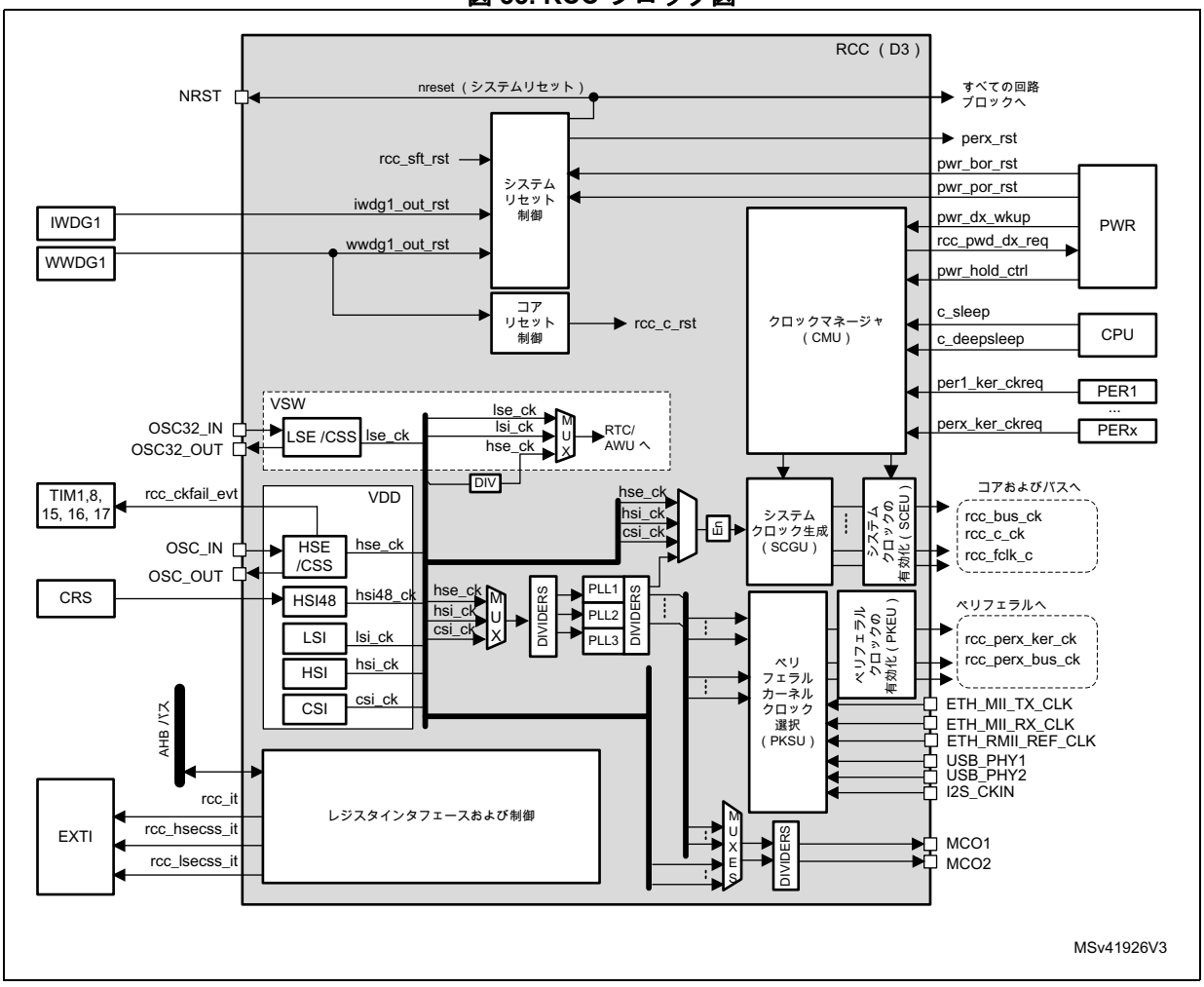
クロック生成ブロック

- デバイス全体に対するクロックの生成およびディスパッチ
- 整数比または分数比を用いた 3 つの異なる PLL
- 動作中の PLL 分数比変更可能性
- 消費電力を低減するためのスマートクロックゲーティング
- 2 つの外部オシレータ：
  - 4 から 48 MHz の周波数の幅広いクリスタルをサポートしているハイスピード外部オシレータ（HSE）
  - 32 kHz のクリスタルのロースピード外部オシレータ（LSE）
- 4 つの内部オシレータ
  - ハイスピード内部オシレータ（HSI）
  - 48 MHz の RC オシレータ（HSI48）
  - 低電力内部オシレータ（CSI）
  - ロースピード内部オシレータ（LSI）
- 外部デバイス用にバッファされたクロック出力
- 2 種類の割り込みラインの生成：
  - クロックセキュリティ管理専用の割り込みライン
  - その他のイベント用の 1 つの一般的な割り込みライン
- STOP モードおよび STANDBY モードでのクロック生成の処理
- D3 ドメインでの自律モード

8.2 RCC ブロック図

図 33 に RCC ブロック図を示します。

図 33. RCC ブロック図



8.3 RCC ピンおよび内部信号

表 44 にパッケージのピンまたはボールに接続された RCC 入出力信号を記載します。

表 44. パッケージのピンまたはボールに接続された RCC 入出力信号

信号名	信号タイプ	説明
NRST	I/O	システムリセット、外部デバイスにリセットを提供するために使用可能
OSC32_IN	I	32 kHz オシレータ入力
OSC32_OUT	O	32 kHz オシレータ出力
OSC_IN	I	システムオシレータ入力
OSC_OUT	O	システムオシレータ出力

参考資料

表 44. パッケージのピンまたはボールに接続された RCC 入出力信号（続き）

信号名	信号タイプ	説明
MCO1	O	外部デバイスの出力クロック 1
MCO2	O	外部デバイスの出力クロック 2
I2S_CKIN	I	デジタルオーディオインタフェースの外部カーネルクロック入力：SPI/I2S、SAI、および DFSDM
ETH_MII_TX_CLK	I	Ethernet MII インタフェースによって提供される外部 TX クロック
ETH_MII_RX_CLK	I	Ethernet MII インタフェースによって提供される外部 RX クロック
ETH_RMII_REF_CLK	I	Ethernet RMII インタフェースによって提供される外部リファレンスクロック
USB_PHY1	I	外部 USB PHY によって提供される USB クロック入力
USB_PHY2	I	外部 USB PHY によって提供される USB クロック入力

RCC では製品のすべてのコンポーネントと多くの内部信号をやり取りします。このため、*表 44* には最上位の内部信号のみを示します。

表 45. RCC 内部入力／出力信号

新しい信号名	信号タイプ	説明
rcc_it	O	一般的な割り込みリクエストライン
rcc_hsecss_it	O	HSE クロックセキュリティ障害割り込み
rcc_lsecss_it	O	LSE クロックセキュリティ障害割り込み
rcc_ckfail_evt	O	HSE クロックセキュリティ障害の検出を示すイベント。信号は TIMERS に接続されます。
nreset	I/O	システムリセット
iwdg1_out_rst	I	タイムアウトの発生を示す IWDG1 で駆動されるリセットライン
wwdg1_out_rst	I	タイムアウトの発生を示す WWDG1 で駆動されるリセットライン
pwr_bor_rst	I	PWR ブロックによって生成されるブラウンアウトリセット
pwr_por_rst	I	PWR ブロックによって生成されるパワーオンリセット
pwr_vsw_rst	I	PWR ブロックによって生成される VSW ドメインのパワーオンリセット
rcc_perx_rst	O	ペリフェラル用に RCC によって生成されるリセット
pwr_d[3:1]_wkup	I	PWR によって生成されるウェイクアップドメインリクエスト。一般に、ドメインが DStop を終了するときに、このドメインのクロックを復元するために使用します。
rcc_pwd_d[3:1]_req	O	RCC によって生成される低電力リクエスト。一般に、ドメインが DStop のときに、ドメインを低電力モードにセットするかどうかを PWR に確認するために使用します。
pwr_hold_ctrl	I	システム STOP モードを終了するときに、プロセッサを CStop にセットするために PWR によって生成される信号
c_sleep	I	CPU が CRun、CSleep、または CStop であることを示す、CPU によって生成される信号
c_deepsleep	I	
perx_ker_ckreq	I	カーネルクロックの有効化を要求するために一部のペリフェラルによって生成される信号
rcc_perx_ker_ck	O	一部のペリフェラル用に RCC によって生成されるカーネルクロック信号
rcc_perx_bus_ck	O	ペリフェラル用に RCC によって生成されるバスインタフェースクロック信号

表 45. RCC 内部入力／出力信号（続き）

新しい信号名	信号タイプ	説明
rcc_bus_ck	O	RCC によって生成される APB （rcc_apb_ck）、AHB （rcc_ahb_ck）、および AXI （rcc_axi_ck）ブリッジのクロック
rcc_c_ck	O	RCC によって生成される CPU のクロック
rcc_fclk_c	O	

8.4 RCC リセットブロックの機能説明

いくつかのソースでリセットを生成できます。

- NRST ピンを介して接続された外部デバイス
- V_{DD} に適用された電源電圧の異常
- ウォッチドッグタイムアウト
- ソフトウェアコマンド

リセットの範囲は、リセットを生成するソースに依存します。3 つのリセットカテゴリがあります。

- パワーオン／オフリセット
- システムリセット
- ローカルリセット

8.4.1 パワーオン／オフリセット

パワーオン／オフリセット（**pwr_por_rst**）は電源コントローラブロック（PWR）によって生成されます。これは、入力電圧（V_{DD}）が閾値レベルを下回る場合に有効化されます。これによりバックアップドメインを除く回路全体がリセットされるため、最も完全なリセットであると言えます。パワーオン／オフリセット機能は、PDR_ON ピンによって無効にすることができます（*セクション 6.5：電源供給の監視*を参照）。

詳細については、*表 46：リセット供給の概要*を参照してください。

8.4.2 システムリセット

システムリセット（**nreset**）は、RCC_RSR（または RCC_C1_RSR）レジスタのリセットステータスフラグ、デバッグ機能、フラッシュメモリ、およびバックアップドメインレジスタを除く、すべてのレジスタをデフォルト値にリセットします。

システムリセットは、次のいずれかのソースから生成できます。

- NRST ピンからのリセット（外部リセット）
- パワーオン／オフリセットブロックからのリセット（**pwr_por_rst**）
- ブラウンアウトリセットブロックからのリセット（**pwr_bor_rst**）

  BOR 機能の詳細な説明については、*セクション 6.5.2：ブラウンアウトリセット（BOR）*を参照してください。
- 独立型ウォッチドッグからのリセット（**iwdg1_out_rst**）
- Cortex®-M7 コアからのソフトウェアリセット

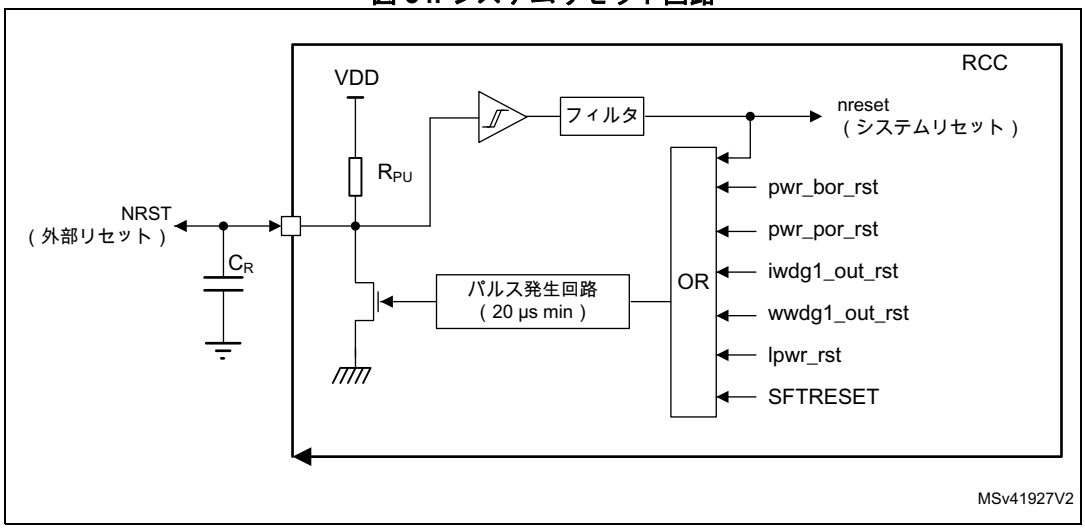
  Cortex®-M7 コアによって発行される SYSRESETREQ 信号から生成されます。本書では、この信号も SFTRESET と呼ばれています。
- WWDG 設定に応じたウィンドウ型ウォッチドッグからのリセット（**wwdg1_out_rst**）
- オプションバイト設定に応じた低電力モードセキュリティリセットからのリセット（**lpwr[2:1]_rst**）

注： ***デバイス上でソフトウェアリセットを実行するには、Cortex®-M7 のアプリケーション割り込みおよびリセット制御レジスタの SYSRESETREQ ビットをセットする必要があります。詳細については、FPU 搭載 Cortex®-M7 テクニカルリファレンスマニュアルを参照してください（http://infocenter.arm.com を参照）。***

図 34 に示すように、一部の内部ソース（**pwr_por_rst**、**pwr_bor_rst**、**iwdg1_out_rst** など）では回路のシステムリセットを行います。これは、接続された外部デバイスをリセットするために NRST ピンにも伝達されます。パルス発生回路は、各内部リセットソースについて 20µs の最小リセットパルス期間を保証します。外部リセットの場合、リセットパルスは NRST ピンがローにアサートされているときに生成されます。

注： ***NRST ピンを未接続にするのはお勧めしません。使用しないときは、10 から 100 nF のコンデンサを介してグランドに接続します（図 34 の C_R）。***

図 34. システムリセット回路



8.4.3 ローカルリセット

CPU リセット

CPU は、*RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）*の CPURST ビットを用いてリセットすることができます。

ドメインリセット

一部リセットは、ドメインの状態にも依存しています。たとえば、D1 ドメインは、DStandby を終了するときにリセットされます（**d1_rst**）。D2 にも同じ方法が適用されます。

システムが STANDBY モードを終了するときに、**stby_rst** リセットが適用されます。**stby_rst** 信号は、内部レギュレータから供給される V_{CORE} 電圧が無効である限り、V_{CORE} ドメイン全体のリセットを生成します。

表 46 にリセットソースとその適用範囲の詳細な概要を示します。

表 46. リセット供給の概要

リセットソース	リセット名	D1 CPU	D1 相互接続	D1 ペリフェラル	D1 デバッグ	WWDG1	D2 相互接続	D2 ペリフェラル	D3 ペリフェラル	IWDG1	FLASH	RTC ドメイン	バックアップ RAM	システム電源	NRST ピン	コメント
ピン	NRST	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– D1 および D2 ドメインと、そのすべてのペリフェラルをリセットします。 – D3 ドメインペリフェラルをリセットします。 – V_{DD} ドメインをリセットします（IWDG1 や LDO など）。 – デバッグ機能、フラッシュメモリ、RTC、およびバックアップ RAM はリセットされません。
PWR	pwr_bor_rst	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– ピンリセットと同じです。ピンもアサートされます。
	pwr_por_rst	x	x	x	x	x	x	x	x	x	x	-	-	x	x	– **pwr_bor_rst** リセットと同じです。加えて、次のリセットも実行されます。 フラッシュメモリデジタルブロック（オプションバイトローディングを含む）のリセット デバッグブロックのリセット
	lpwr_rst	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– 低電力モードセキュリティリセットの適用範囲は **pwr_por_rst** と同じです。詳細については、*セクション 8.4.5：低電力モードセキュリティリセット（lpwr_rst）*を参照してください。

表 46. リセット供給の概要（続き）

リセットソース	リセット名	D1 CPU	D1 相互接続	D1 ペリフェラル	D1 デバッグ	WWDG1	D2 相互接続	D2 ペリフェラル	D3 ペリフェラル	IWDG1	FLASH	RTC ドメイン	バックアップ RAM	システム電源	NRST ピン	コメント
RCC	BDRST	-	-	-	-	-	-	-	-	-	-	x	-	-	-	– ソフトウェアによって、バックアップドメインリセットをトリガできます。詳細については、*セクション 8.4.6 : バックアップドメインリセット*を参照してください。
	d1_rst	x	x	x	x	x	-	-	-	-	-	-	-	-	-	– D1 ドメインとそのすべてのペリフェラルを、DStandbyモード 終了時にリセットします。
	d2_rst	-	-	-	-	-	x	x	-	-	-	-	-	-	-	– D2 ドメインとそのすべてのペリフェラルを、DStandbyモード 終了時にリセットします。
	stby_rst	x	x	x	x	x	x	x	x	-	-	-	-	-	-	– デバイスが STANDBY モードを終了するときに、V_{CORE} 電圧が無効である限り、V_{CORE} ドメイン全体のリセットが実行されます。V_{CORE} は、内部レギュレータによって供給されます。 NRST 信号はアサートされません。
	CPURST	x	-	-	-	x	-	-	-	-	-	-	-	-	-	– このリセットは、*RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）*のビットで、ソフトウェアによって生成されます。 – CPU および WWDG1 ブロックをリセットします。
CPU	SFTRESET	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– このリセットは、Cortex®-M7 コアの AIRCR レジスタに SYSRESETREQ ビットを書き込む際に、ソフトウェアによって生成されます。 – 適用範囲は **pwr_bor_rst** リセットと同じです。
バックアップドメイン	pwr_vsw_rst	-	-	-	-	-	-	-	-	-	-	x	-	-	-	– このリセットは、V_{SW} 電源電圧が動作範囲外である場合に、バックアップドメインによって生成されます。
IWDG1	iwdg1_out_rst	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– **pwr_bor_rst** リセットと同じです。
WWDG1	wwdg1_out_rst	x	x	x	-	x	x	x	x	x	-	-	-	-	x	– **pwr_bor_rst** リセットと同じです。

8.4.4 リセットソース識別

CPU は、RCC_RSR （または RCC_C1_RSR）レジスタのリセットフラッグを確認することで、リセットソースを識別できます。

CPU は、RMVF ビットを設定することで、フラグをリセットすることができます。

表 47 に、RCC_RSR （または RCC_C1_RSR）レジスタのステータスビットが、リセット生成時の条件に応じてどのように挙動するかを示します。たとえば、IWDG1 タイムアウトの発生時に（ライン #10）、CPU がブートフェーズ中に RCC_RSR （または RCC_C1_RSR）レジスタを読み出している場合、PINRSTF と IWDG1RSTF の両方のビットがセットされます。これは、IWDG1 でピンリセットも生成されたことを示します。

表 47. リセットソース識別（RCC_RSR）(1)

#	リセット生成時の条件	LPWRRSTF	WWDG1RSTF	IWDG1RSTF	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	D2RSTF	D1RSTF	CPURSTF
1	パワーオンリセット（pwr_por_rst）	0	0	0	0	1	1	1	1	1	1
2	ピンリセット（NRST）	0	0	0	0	0	1	0	0	0	1
3	ブラウンアウトリセット（pwr_bor_rst）	0	0	0	0	0	1	1	0	0	1
4	CPU によって生成されたシステムリセット（SFTRESET）	0	0	0	1	0	1	0	0	0	1
5	CPU リセット（CPURST）	0	0	0	0	0	0	0	0	0	1
6	WWDG1 リセット（wwdg1_out_rst）	0	1	0	0	0	**1**	0	0	0	**1**
8	IWDG1 リセット（iwdg1_out_rst）	0	0	1	0	0	1	0	0	0	1
10	D1 が DStandby モードを終了した場合	0	0	0	0	0	0	0	0	1	0
11	D2 が DStandby モードを終了した場合	0	0	0	0	0	0	0	1	0	0
12	D1 が誤って DStandby モードに移行した場合、または CPU が誤って CStop モードに移行した場合	1	0	0	0	0	1	0	0	0	1

1. セットされているレジスタビットのセルは、灰色でハイライトされています。

8.4.5 低電力モードセキュリティリセット（lpwr_rst）

誤って重要なアプリケーションが低電力モードに移行しないように、2 つの低電力モードセキュリティリセットが使用できます。nRST_STOP_D1 オプションバイトで有効になっている場合、以下の条件を満たしていればシステムリセットが生成されます。

- CPU が予期せず CStop モードに移行した場合

  この種類のリセットは、ユーザオプションバイトの nRST_STOP_D1 をリセットすることで有効になります。この場合、CPU の CStop モードへの遷移シーケンスが正常に実行されるたびに、システムリセットが生成されます。

- D1 ドメインが予期せず DStandby モードに移行した場合

  この種類のリセットは、ユーザオプションバイトの nRST_STDBY_D1 ビットをリセットすることで有効になります。この場合、D1 ドメインの DStandby モードへの遷移シーケンスが正常に実行されるたびに、システムリセットが生成されます。

*RCC リセットステータスレジスタ（RCC_RSR）*の LPWRRSTF ビットは、低電力モードセキュリティリセットが発生したことを示します（*表 47* のライン #12 を参照）。

D1 または CPU による低電力モードセキュリティリセットが発生すると、**lpwr_rst** が有効化されます。

詳細については、*セクション 3.3.11：フラッシュオプションバイト*を参照してください。

8.4.6 バックアップドメインリセット

バックアップドメインのリセットは、次のいずれかのイベントが発生したときに生成されます。

- *RCC バックアップドメイン制御レジスタ（RCC_BDCR）* の BDRST ビットをセットすることでソフトウェアリセットがトリガされた場合。すべての RTC レジスタと RCC_BDCR レジスタはデフォルト値にリセットされます。バックアップ RAM は、影響を受けません。
- V_{SW} 電圧が動作範囲外である場合。すべての RTC レジスタと RCC_BDCR レジスタはデフォルト値にリセットされます。この場合、バックアップ RAM の内容は有効でなくなります。

バックアップ RAM をリセットするには、2 つの方法があります。

- フラッシュメモリインタフェースを使用して保護レベルの 1 から 0 への変更を要求する
- タンパイベント発生時

詳細については、PWR ブロックの *セクション 6.4.4：バックアップドメイン*セクションを参照してください。

8.4.7 パワーオンおよびウェイクアップのシーケンス

詳細な図については、PWR のセクションの *セクション 6.4.1：システム電源の起動* を参照してください。

製品が低電力を終了するイベントと CPU がコードを実行できるようになる時点とのあいだの時間間隔は、システムの状態と設定に依存します。*図 35* に最も一般的な例を示します。

パワーオンウェイクアップシーケンス

図 35 に示すパワーオンウェイクアップシーケンスは、パワーオンシーケンスの最上位のフェーズです。回路に電源が投入されていないため、一番長いシーケンスです。V_{BAT} が存在するかどうかにかかわらず、シーケンスが変化しないことに注意してください。

ピンリセットからのブート

ピンリセットの発生時に、V_{DD} はまだ存在しています。その結果：

- 基準電圧がすでに安定しているため、レギュレータの整定時間を短縮できます。
- HSI が NRST 発生時に有効化されなかった場合は、HSI の再起動遅延が必要になる場合がありますが、有効化された場合、この再起動遅延フェーズはスキップされます。
- フラッシュメモリが NRST 発生時に有効化された場合は、フラッシュメモリ電源リカバリ遅延もスキップできます。

注： ***ブートシーケンスは pwr_bor_rst、lpwr_rst、STFxRESET、iwdg1_out_rst、および wwdg1_out_rst（WW1RSC = '1' の場合）で同じです。***

システム STANDBY からのブート

システム STANDBY からウェイクアップする場合、V_{DD} は削除されていないため、基準電圧は安定しています。結果として、レギュレータの整定時間を短縮できます。V_{CORE} は存在しないため、HSI の再起動遅延、フラッシュメモリの電源リカバリ、およびオプションバイトの再ロードをスキップすることはできません。

システム STOP からの再起動

システム STOP からの再起動時に、V_{DD} はまだ存在しています。結果として、シーケンスは主に 2 つのステップで構成されます。

1. VOS3 （デフォルト電圧）に達するためのレギュレータの整定時間
2. HSI/CSI の再起動遅延。このステップは、*RCC ソース制御レジスタ（RCC_CR）* の HSIKERON または CSIKERON ビットが「1」にセットされている場合はスキップできます。

ドメイン DStandby からのブート

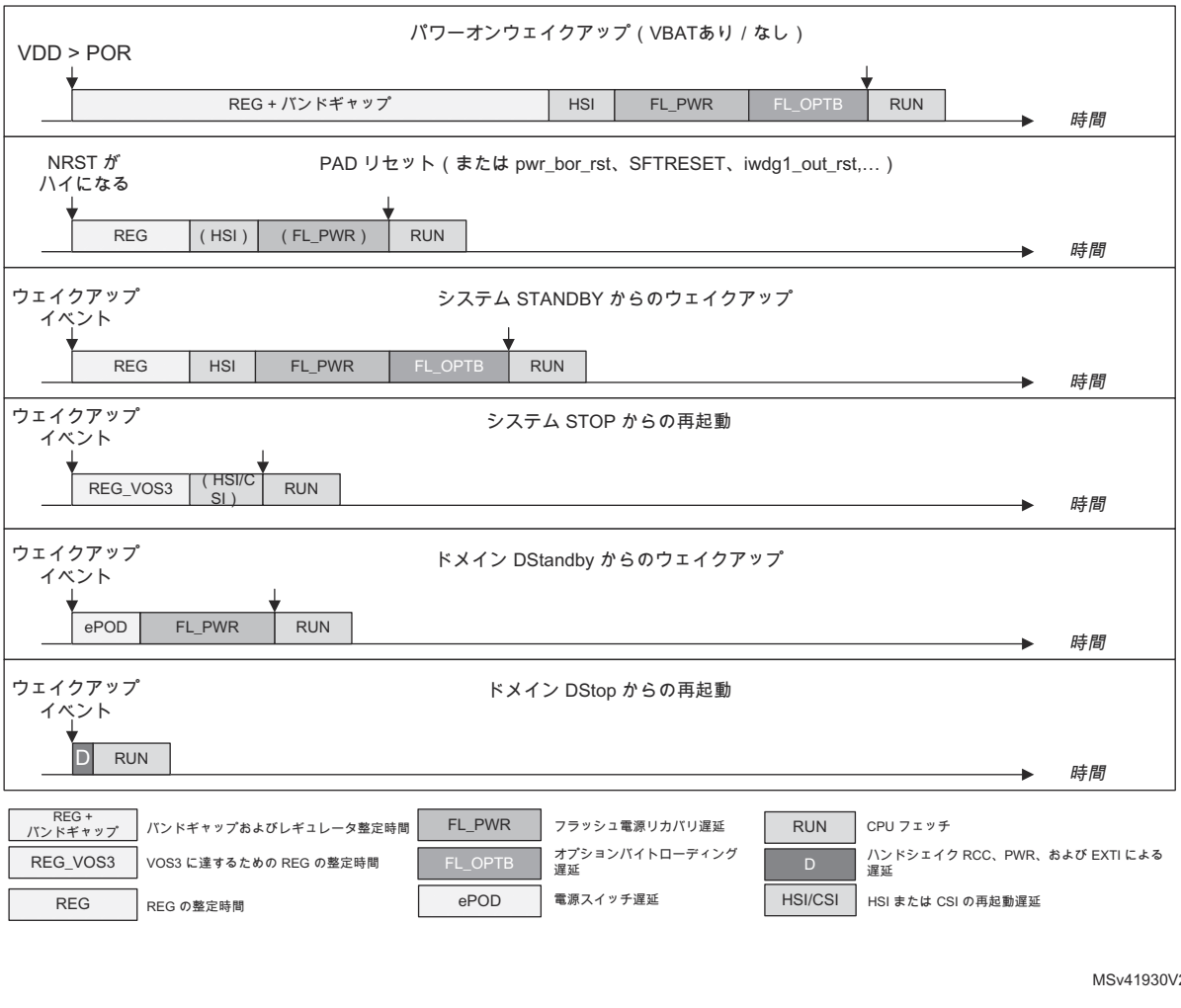
ドメイン DStandby からのドメインのブートシーケンスは、主に 2 つのステップで構成されます。

1. 電源スイッチの整定時間（レギュレータはすでにアクティブ）
2. フラッシュメモリの電源リカバリ

ドメイン DStop からの再起動

ドメイン DStop からのドメインの再起動シーケンスは、主に RCC、EXTI、および PWR ブロック間のハンドシェイクで構成されます。

図 35. ブートシーケンスとシステムの状態



8.5 RCC クロックブロックの機能説明

RCC では、幅広いクロックジェネレータの選択肢を提供しています。

- HSI （ハイスピード内部オシレータ）クロック：8、16、32、または 64 MHz まで
- HSE （ハイスピード外部オシレータ）クロック：4 から 48 MHz
- LSE （ロースピード外部オシレータ）クロック：32 kHz
- LSI （ロースピード内部オシレータ）クロック：32 kHz まで
- CSI （ロースピード内部オシレータ）クロック：4 MHz まで
- HSI48 （ハイスピード 48 MHz 内部オシレータ）クロック：48 MHz まで

このブロックでは、CPU とペリフェラル、特に Ethernet、USB OTG-FS ならびに HS、SPI/I2S、SAI、および SDMMC などの特定のクロックを必要とするペリフェラルに適切なクロックを選択するために、アプリケーションに高い柔軟性を与えます。

消費電力を最適化するために、各クロックソースのスイッチは個別にオン／オフを切り替えることができます。

RCC は、最大 3 つの PLLを提供します。これらはそれぞれ整数比または分数比を用いて設定できます。

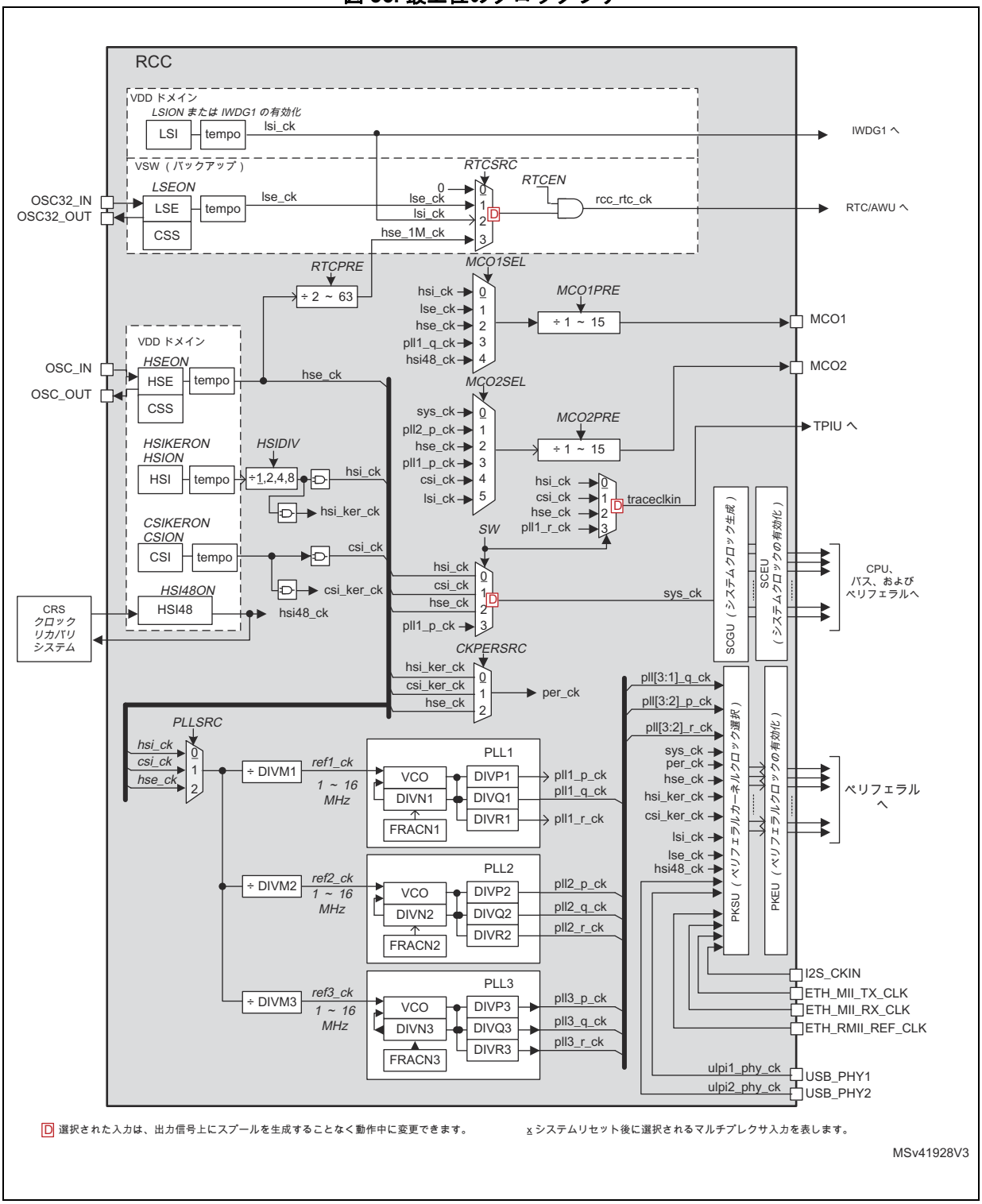
図 36 に示すように、RCC では非常に柔軟にクロックを選択し、周波数を調整できる 2 つのクロック出力（MCO1 および MCO2）を使用できます。

SCGU ブロック（System Clock Generation Unit、システムクロック生成ユニット）には、CPU とバスマトリックスのクロック周波数を設定するいくつかのプリスケーラが含まれます。

PKSU ブロック（Peripheral Kernel clock Selection Unit、ペリフェラルカーネルクロック選択ユニット）には、ペリフェラルに選択可能な多くのカーネルクロック分配を提供するいくつかの動的スイッチがあります。

PKEU （Peripheral Kernel clock Enable Unit、ペリフェラルカーネルクロックイネーブルユニット）および SCEU （System Clock Enable Unit、システムクロックイネーブルユニット）ブロックではペリフェラルカーネルクロックゲーティングおよびバスインタフェース／コア／バスマトリックスゲーティングをそれぞれ実行します。

図 36. 最上位のクロックツリー



8.5.1 クロックの命名規則

RCC では、回路全体にクロックを提供します。誤解を避けるため、本書では以下に定義するように各用語を使用します。

- ペリフェラルクロック

  ペリフェラルクロックは、RCC によってペリフェラルに提供されるクロックです。次の 2 種類のクロックを使用できます。

  – バスインタフェースクロック

  – カーネルクロック

  ペリフェラルはレジスタにアクセスするために RCC からバスインタフェースクロックを受信し、これによりペリフェラルの動作を制御します。このクロックは通常ペリフェラルの接続先のバスに応じて AHB、APB、または AXI クロックのいずれかです。一部のペリフェラルではバスインタフェースクロック（RNG、TIMx など）のみを使用します。

  一部のペリフェラルでは、インタフェース機能を処理するために、専用のクロックが必要です。このクロックは「カーネルクロック」と呼ばれます。たとえば、SAI などのペリフェラルでは特定の正確なマスタクロック周波数を生成する必要があり、これには専用のカーネルクロック周波数が必要です。特定のインタフェースのニーズからバスインタフェースクロックを切り離すもう一つの利点は、ペリフェラルを再プログラミングせずにバスクロックを変更できることです。

- CPU クロック

  CPU クロックは CPU に提供されるクロックです。システムクロック（**sys_ck**）から生成されます。

- バスマトリックスクロック

  バスマトリックスクロックは、異なるブリッジ（APB、AHB、または AXI）に提供されるクロックです。これらのクロックは、システムクロック（**sys_ck**）から生成されます。

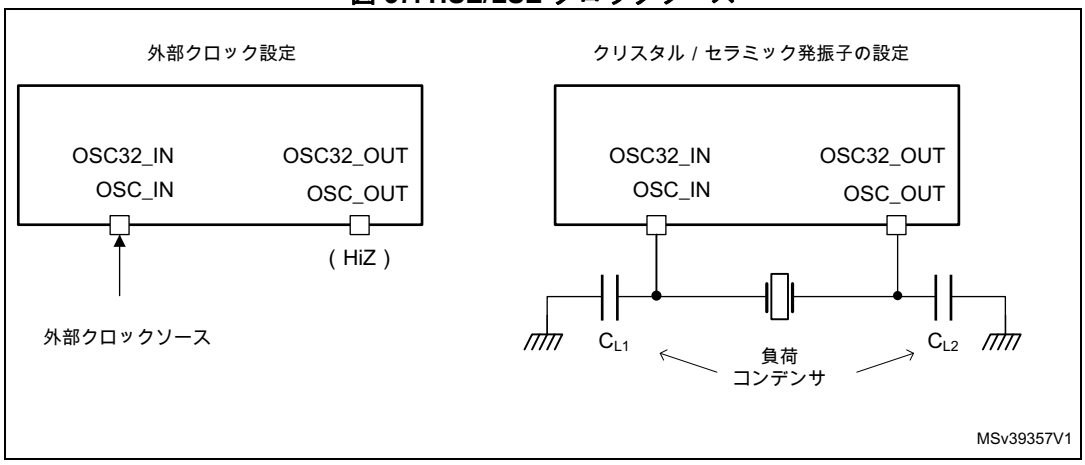
8.5.2 オシレータの説明

HSE オシレータ

HSE ブロックでは、2 つの使用可能なソースからクロックを生成できます。

- 外部クリスタル／セラミック発振子
- 外部クロックソース

図 37. HSE/LSE クロックソース



外部クロックソース（HSE バイパス）

このモードでは、OSC_IN ピンに外部クロックソースを提供する必要があります。このモードを選択するには、*RCC ソース制御レジスタ（RCC_CR）*の HSEBYP および HSEON ビットを「1」にセットします。約 50 % のデューティサイクルを持つ外部クロックソース（矩形波、正弦波、または三角波）で OSC_IN ピンを駆動する必要があり、その間、OSC_OUT ピンはハイインピーダンスに保つ必要があります（*図 37* を参照）。

外部クリスタル／セラミック発振子

オシレータは、HSEBYP ビットを「0」に、HSEON ビットを「1」にセットすることにより有効になります。

製品が非常に高精度のハイスピードクロックを必要とする場合は、HSE を使用できます。

関連するハードウェア構成を*図 37* に示します。波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したクリスタルまたはセラミック発振子に応じて調整する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。

*RCC ソース制御レジスタ（RCC_CR）*の HSERDY フラグは、HSE オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、**hse_ck** クロック信号は出力されません。*RCC クロックソース割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、割り込みを生成することができます。

HSE は、HSEON ビットを使用してオン／オフを切り替えることができます。次の 2 つの条件のうちの 1 つが満たされると、HSE をオフにすることはできません。

- HSE はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- HSE は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは HSEON ビットを「0」にプログラミングすることはできません。

HSE は、システムが STOP または STANDBY モードに移行すると、ハードウェアによって自動的に無効化されます（詳細については、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照）。

さらに、HSE クロックは MCO1 および MCO2 の出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

LSE オシレータ

LSE ブロックでは、2 つの使用可能なソースからクロックを生成できます。

- 外部クリスタル／セラミック発振子
- 外部ユーザクロック

外部クロックソース（LSE バイパス）

このモードでは、OSC32_IN ピンに外部クロックソースを提供する必要があります。入力クロックには最大 1 MHz までの周波数を使用できます。このモードを選択するには、*RCC バックアップドメイン制御レジスタ（RCC_BDCR）*の LSEBYP および LSEON ビットを「1」にセットします。約 50% のデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）で OSC32_IN ピンを駆動する必要があり、その間、OSC32_OUT ピンはハイインピーダンスに保つ必要があります（*図 37* を参照）。

外部クリスタル／セラミック発振子（LSE クリスタル）

LSE クロックは、32.768 kHz のクリスタルまたはセラミック発振子から生成されます。時計／カレンダ、その他のタイミング機能のためのリアルタイムクロック（RTC）に、低電力の高精度なクロックソースを供給できるという利点があります。

RCC バックアップドメイン制御レジスタ（RCC_BDCR） の LSERDY フラグは、LSE クリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSE クリスタル出力クロック信号はリリースされません。*RCC クロックソース割り込み有効レジスタ（RCC_CIER）* で有効になっていれば、割り込みを生成することができます。

LSE オシレータは、LSEON ビットを使用してオン／オフの切り替えができます。LSE は、システムの STOP または STANDBY モードへの移行時にも有効なままです。

さらに、LSE クロックは MCO1 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

LSE では、アンプの駆動能力を調節するために使用できるプログラム可能な駆動能力（LSEDRV[1:0]）も提供しています。駆動能力は、高駆動から中高駆動へ、次いで中低駆動へ動的に変更できます。

HSI オシレータ

HSI ブロックは、製品にデフォルトクロックを供給します。

HSI は、ハイスピード内部 RC オシレータで、システムクロック、ペリフェラルクロック、または PLL 入力として直接使用できます。前置分周器では、アプリケーションで 8、16、32、または 64 MHz の HSI 出力周波数を選択することができます。この前置分周器は、HSIDIV で制御されます。

HSI の利点は以下のとおりです。

- 外部クリスタルが不要であるため、クロックソースのコストを抑えることができる
- HSE よりも起動時間が短い（数ミリ秒）

HSI 周波数は、周波数の較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

HSI は、HSION ビットを使用してオン／オフを切り替えることができます。次の 2 つの条件のうちの 1 つが満たされると、HSI をオフにすることはできません。

- HSI はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- HSI は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは HSION ビットを「0」にプログラミングすることはできません。

最低 1 つの有効な PLL （PLLxON ビットを「1」にセット）で HSI をリファレンスクロックとして選択している場合は、HSIDIV を変更することはできません。この場合、ハードウェアでは HSIDIV を新しい値で更新しません。ただし、HSI を直接システムクロックとして使用している場合は、HSIDIV を変更することができます。

HSIRDY フラグは、HSI が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI 出力クロックは出力されません。

HSI クロックは、HSE に障害がある場合のバックアップソース（補助クロック）としても使用できます（*セクション：HSE の CSS*を参照）。HSI は、システムが STOP モードに移行する際に、無効にすることもできます（詳細については、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照）。

さらに、HSI クロックは MCO1 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

HSI を通信ペリフェラルのカーネルクロックとして使用している場合は注意が必要です。アプリケーションでは以下のパラメータを考慮する必要があります。

- ペリフェラルがカーネルクロックリクエストを生成する時点とクロックが実際に使用可能な時点とのあいだの時間間隔
- 周波数の精度

注： ***HSI は、システムが STOP モードのときに、有効のままにすることができます（詳細については、セクション 8.5.7 を参照）。***

HSION、HSIRDY、および HSIDIV ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。

HSI 較正

RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、ACC_{HSI} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

パワーオンリセット後、工場較正値が HSICAL[11:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの周波数に影響を与えることがあります。ユーザアプリケーションでは、HSITRIM[5:0] ビットを使用して、HSI 周波数をトリミングできます。

注： ***HSICAL[11:0] および HSITRIM[5:0] ビットは、RCC 内部クロックソース較正レジスタ（RCC_ICSCR）にあります。***

CSI オシレータ

CSI は、低電力 RC オシレータで、システムクロック、ペリフェラルクロック、または PLL 入力として直接使用できます。

CSI の利点は以下のとおりです。

- 外部クリスタルが不要であるため、クロックソースのコストを抑えることができる
- HSE よりも起動時間が短い（数ミリ秒）
- 非常に少ない消費電力

HSI では最大 64 MHz のクロック周波数を供給することができますが、CSI では約 4 MHz を供給します。

CSI 周波数は、周波数の較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

CSI は、CSION ビットを使用してオン／オフを切り替えることができます。CSIRDY フラグは、CSI が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、CSI 出力クロックは出力されません。

次の 2 つの条件のうちの 1 つが満たされると、CSI をオフにすることはできません。

- CSI はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- CSI は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは CSION ビットを「0」にプログラミングすることはできません。

CSI は、システムが STOP モードに移行する際に、無効にすることもできます（詳細については、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照）。

さらに、CSI クロックは MCO2 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

CSI の整定時間が HSI より速い場合でも、CSI を通信ペリフェラルのカーネルクロックとして使用している場合は注意が必要です。アプリケーションでは以下のパラメータを考慮する必要があります。

- ペリフェラルがカーネルクロックリクエストを生成する時点とクロックが実際に使用可能な時点とのあいだの時間間隔
- 周波数の精度

注： ***CSION および CSIRDY ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。***

CSI 較正

RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、ACC_{CSI} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

リセット後、工場較正値が CSICAL[7:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの周波数に影響を与えることがあります。ユーザアプリケーションでは、CSITRIM[4:0] ビットを使用して、CSI 周波数をトリミングできます。

注： ***CSICAL[7:0] および CSITRIM[4:0] ビットは、RCC 内部クロックソース較正レジスタ（RCC_ICSCR）にあります。***

HSI48 オシレータ

HSI48 は一部のペリフェラルでカーネルクロックとして直接使用できる 48 MHz クロックを供給する RC オシレータです。

HSI48 オシレータは、主に、特殊なクロックリカバリシステム（CRS）回路によって USB ペリフェラルに高精度のクロックを提供します。動作中のオシレータの周波数を自動的に最小単位で調整するために、USB SOF 信号、LSE、または外部信号を使用できます。

HSI48 オシレータは、システムが STOP または STANDBY モードに移行した直後に無効化されます。CRS が使用されない場合、このオシレータはフリーランニングするため製造工程でのばらつきの影響を受けます。このため、ACC_{HSI48} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

CRS の設定および使用方法の詳細については、*セクション 9：クロックリカバリシステム（CRS）*を参照してください。

HSI48RDY フラグは、HSI48 オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 出力クロックはリリースされません。

HSI48 は、HSI48ON ビットを使用してオン／オフを切り替えることができます。

HSI48 クロックは MCO1 マルチプレクサに駆動することもでき、他のアプリケーションコンポーネントのクロックソースとしても使用できます。

注： ***HSI48ON および HSI48RDY ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。***

LSI オシレータ

LSI は、独立型ウォッチドッグ（IWDG）や自動ウェイクアップユニット（AWU）で、システムがSTOP モードまたは STANDBY モードの場合に動作状態を維持できる低電力のクロックソースとして動作します。クロック周波数は約 32 kHz です。詳細については、データシートの電気的特性のセクションを参照してください。

LSI は、LSION ビットを使用してオン／オフを切り替えることができます。LSIRDY フラグは、LSI オシレータが安定しているかどうかを示します。独立型ウォッチドッグがハードウェアまたはソフトウェアとして開始された場合、LSI は強制的にオンになり、オフにすることはできません。

LSI は、システムの STOP または STANDBY モードへの移行時にも有効なままです（詳細については、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照）。

LSI の起動時に、LSIRDY ビットがハードウェアによってセットされるまで、クロックは供給されません。*RCC クロックソース割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、割り込みを生成することができます。

さらに、LSI クロックは MCO2 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

注： ***LSION および LSIRDY ビットは、RCC クロック制御およびステータスレジスタ（RCC_CSR）にあります。***

8.5.3 クロックセキュリティシステム（CSS）

HSE の CSS

クロックセキュリティシステムは、HSECSSON ビットを介してソフトウェアで有効にできます。HSECSSON ビット は HSEON が「0」に設定されている場合でも有効にすることができます。

HSE の CSS は、HSE が有効かつレディ状態であり、HSECSSON が「1」にセットされている場合に、ハードウェアによって有効にすることができます。

HSE の CSS は、HSE が無効にされたときに無効化されます。結果として、この機能はシステムが STOP モードのときは機能しません。

HSECSSON ビットをソフトウェアによって直接クリアすることはできません。

HSECSSON ビットは、システムリセットの発生時、またはシステムの STANDBY モードへの移行時に、ハードウェアによってクリアされます（*セクション 8.4.2：システムリセット*を参照）。

HSE クロックで障害を検出すると、システムはセーフクロックを供給するために自動的に HSI に切り替わります。その後 HSE は自動的に無効になり、クロック障害イベントが高機能制御タイマ（TIM1、TIM8、TIM15、TIM16、および TIM17）のブレーク入力に送られます。また、ソフトウェアに障害（CSS 割り込み：***rcc_hsecss_it***）があることを通知する割り込みが生成されるので、MCU は対応処理を行うことができます。障害発生時に HSE 出力が PLL のクロックソースとして使用されている場合は、PLL も無効になります。

CSS を有効にしたときに HSE クロック障害が発生する場合は、CSS が 割り込みを生成します。これにより、NMI の自動生成が実行されます。*RCC クロックソース割り込みフラグレジスタ（RCC_CIFR）*の HSECSSF フラグは、アプリケーションで障害の原因を見つけるために「1」にセットされます。NMI ルーチンは、HSECSSF ビットがクリアされるまで、無限に実行されます。つまり、アプリケーションでは *RCC クロックソース割り込みクリアレジスタ（RCC_CICR）*の HSECSSC ビットをセットして、NMI ISR で HSECSSF フラグをクリアする必要があります。

LSE の CSS

LSE オシレータのクロックセキュリティシステムは、*RCC バックアップドメイン制御レジスタ（RCC_BDCR）* の LSECSSON ビットをプログラムすることによって、ソフトウェアにより有効にできます。

このビットは、次の条件が満たされた場合に、ハードウェアによってのみ無効にできます。

- pwr_vsw_rst （V_{SW} ソフトウェアリセット）後
- または LSE の障害検出後

LSECSSON ビットは、LSE を有効（ソフトウェアで LSEON ビットをセット）かつレディ状態（ハードウェアで LSERDY ビットをセット）にし、RTCSEL ビットで RTC クロックを選択した後で書き込む必要があります。

LSE の CSS は、VBAT を除くすべてのモード（RUN、STOP、および STANDBY）で動作します。

LSE 障害が検出された場合、LSE クロックは RTC に供給されなくなりますが、RTCSEL、LSECSSON、および LSEON ビットの値がハードウェアによって変更されることはありません。

STANDBY モードでは、ウェイクアップが生成されます。その他のモードでは、ソフトウェアをウェイクアップさせるために割り込み（**rcc_lsecss_it**）を送信できます。その場合、ソフトウェアは、LSECSSON ビットを無効化し、障害のある LSE を停止し（LSEON ビットをクリア）、RTCSEL ビットを介して RTC クロックソースを変更し（クロックまたは LSI または HSE なし）、または必要なアクションを行ってアプリケーションを保護する必要があります。

8.5.4 クロック出力の生成（MCO1/MCO2）

2 つのマイクロコントローラクロック出力（MCO）ピン、MCO1、および MCO2 を利用できます。出力ごとにクロックソースを選択できます。選択したクロックは、設定可能なプリスケーラによって分周できます（信号選択の詳細については、*図 36* を参照）。

MCO1 および MCO2 出力は、*RCC クロック設定レジスタ（RCC_CFGR）* の MCO1PRE[3:0]、MCO1[2:0]、MCO2PRE[3:0]、および MCO2[2:0] を介して制御されます。

各 MCO ピンに対応する GPIO ポートを、オルタネート機能モードでプログラムする必要があります。

MCO 出力に供給されたクロックの最大ピン速度を超えないようにしてください（サポートされているピン速度の詳細については、製品データシートを参照）。

8.5.5 PLL の説明

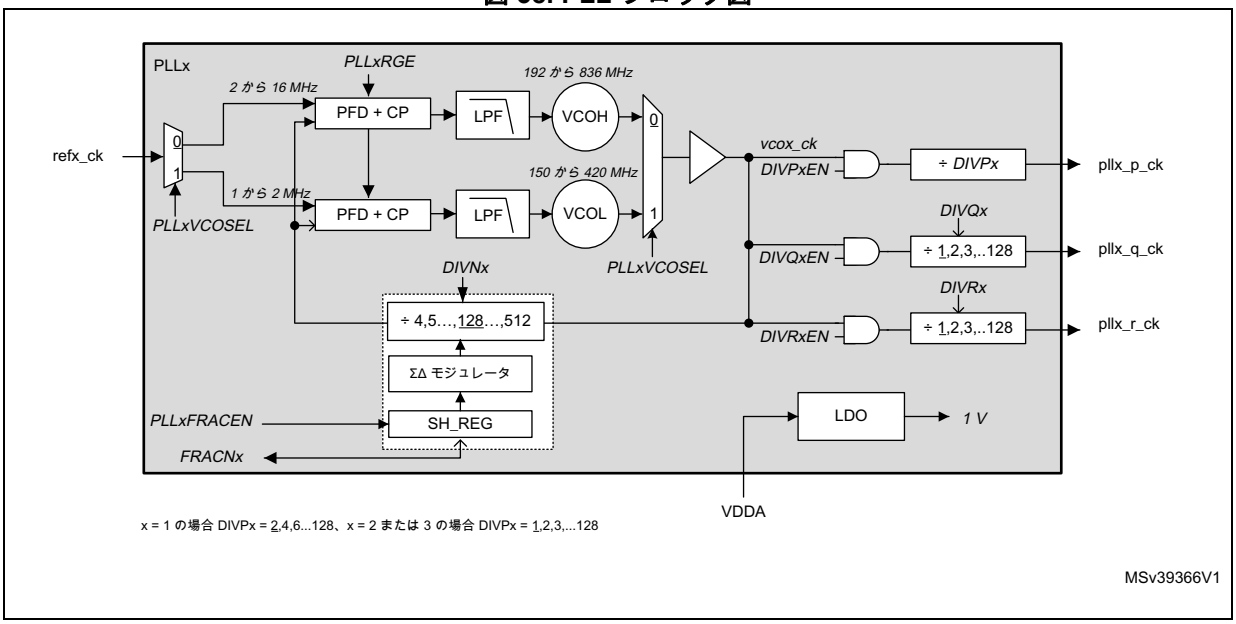
RCC には、次の 3 つの PLL が搭載されています。

- 通常 CPU および一部のペリフェラルにクロックを供給するために使用するメイン PLL（PLL1）
- ペリフェラルのカーネルクロックを生成するために使用する 2 つの専用 PLL（PLL2 および PLL3）

RCC に統合された PLL は、完全に独立しています。これらの PLL には以下の機能があります。

- 2 つの組み込み VCO：
  - 広帯域 VCO （VCOH）
  - 低周波数 VCO （VCOL）
- 入力周波数範囲：
  - 1 ～ 2 MHz （VCOL 使用時）
  - 2 ～ 16 MHz （VCOH 使用時）
- 整数モードまたは分数モードで動作可能
- 13 ビットのシグマデルタモジュレータ、11 ～ 0.3 ppm のステップで VCO 周波数を微調整可能
- シグマデルタモジュレータは、PLL 出力上で周波数のオーバーシュートを生成することなく、動作中に更新できます。
- 各 PLL では、ポスト分周器で 3 つの出力を使用できます。

図 38. PLL ブロック図



PLL は、RCC_PLLxDIVR、RCC_PLLxFRACR、RCC_PLLCFGR、および RCC_CR レジスタによって制御されます。

PLL（**refx_ck**）に供給されるリファレンスクロックの周波数範囲は、1 から 16 MHz の間でなければなりません。この条件を満たすよう、ユーザアプリケーションでは *RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR）* の DIVMx 分周器を適切にプログラムする必要があります。また、PLL を最適性能で動作するために、リファレンス入力周波数に合わせて *RCC PLL 設定レジスタ（RCC_PLLCFGR）* フィールドの PLLxRGE をセットする必要があります。

これにより、ユーザアプリケーションは適切な VCO を設定できるようになります。リファレンスクロックの周波数が 2 MHz 以下の場合は、VCOL を選択する必要がありますが、この値よりも大きい場合は VCOH を選択してください。消費電力を低減するには、VCO 出力を最低周波数に設定することをお勧めします。

VCO 出力で期待される周波数を得るには、DIVNx ループ分周器をプログラムする必要があります。また、VCO 出力範囲を考慮する必要があります。

PLL は、SH_REG （シャドウレジスタ FRACNx）の値を「0」に設定すると、整数モードで動作します。SH_REG は、PLLxFRACEN ビットが「0」から「1」になると FRACNx 値とともに更新されます。シグマデルタモジュレータは、最小周波数ステップを可能にしながら、ジッタによる影響を最小限に抑えるために設計されています。

PLL は、PLLxON を「1」にセットすることで有効化できます。PLLxRDY ビットは、PLL がレディ状態（ロック状態）であることを示します。

注： ***PLL を有効にする前に、PLL に供給されるリファレンス周波数（refx_ck）が安定していることを確認し、PLLx がオンのときにハードウェアによって DIVMx が変更されないようにしてください。また、PLL の 1 つがオンのときに、PLLSRC を変更することができません。***

ハードウェアは、PLL1 を使用してシステムクロックを供給している場合は、PLL1ON への「0」の書き込みを阻止します。クロックジェネレータには、その他のハードウェア保護機能があります（HSE オシレータ、HSI オシレータ、および CSI オシレータを参照）。
PLL パラメータ（DIVNx、PLLxRGE、PLLxVCOSEL、DIVPx、DIVQx、DIVRx、DIVPxEN、DIVQxEN、および DIVRxEN）は、一度 PLL を有効にすると変更することはできません。

PLL の最適挙動を実現するために、ポスト分周器（DIVP、DIVQ、または DIVR）のいずれかが使用されていない場合は、アプリケーションではイネーブルビット（DIVyEN）とともに、対応するポスト分周器ビット（DIVP、DIVQ、または DIVR）を「0」に設定する必要があります。

上記のルールに従わない場合、PLL 出力周波数の動作は保証されません。

出力周波数計算

PLL が整数モード（SH_REG = 「0」）で設定されている場合、VCO 周波数（F_{VCO}）は以下の式で求めることができます。

$$F_{VCO} = F_{REF_CK} \times DIVN$$

$$F_{PLL_y_CK} = (F_{VCO} / (DIVy + 1)) \text{ with } y = P, Q \text{ or } R$$

PLL が分数モードで設定されている場合（SH_REG が「0」以外の場合）、PLL を有効にする前に DIVN 分周器を初期化する必要があります。ただし、PLL 出力を損なうことなく FRACNx の値を動作中に変更することができます。

この機能は、精度の良い任意のクリスタル値から特定の周波数を生成するため、または動作中に周波数を微調整するために使用できます。

各 PLL の VCO 周波数は、次の式で与えられます。

$$F_{VCO} = F_{ref_ck} \times \left(DIVN + \frac{FRACN}{2^{(13)}} \right)$$

注： ***PLL1 の場合、DIVP は奇数値のみをとることができます。***

PLL は、以下の場合にハードウェアによって無効にされます。

- システムが STOP または STANDBY モードに移行したとき
- HSE または HSE でクロックされた PLL をシステムクロックとして使用して、HSE で障害が発生したとき

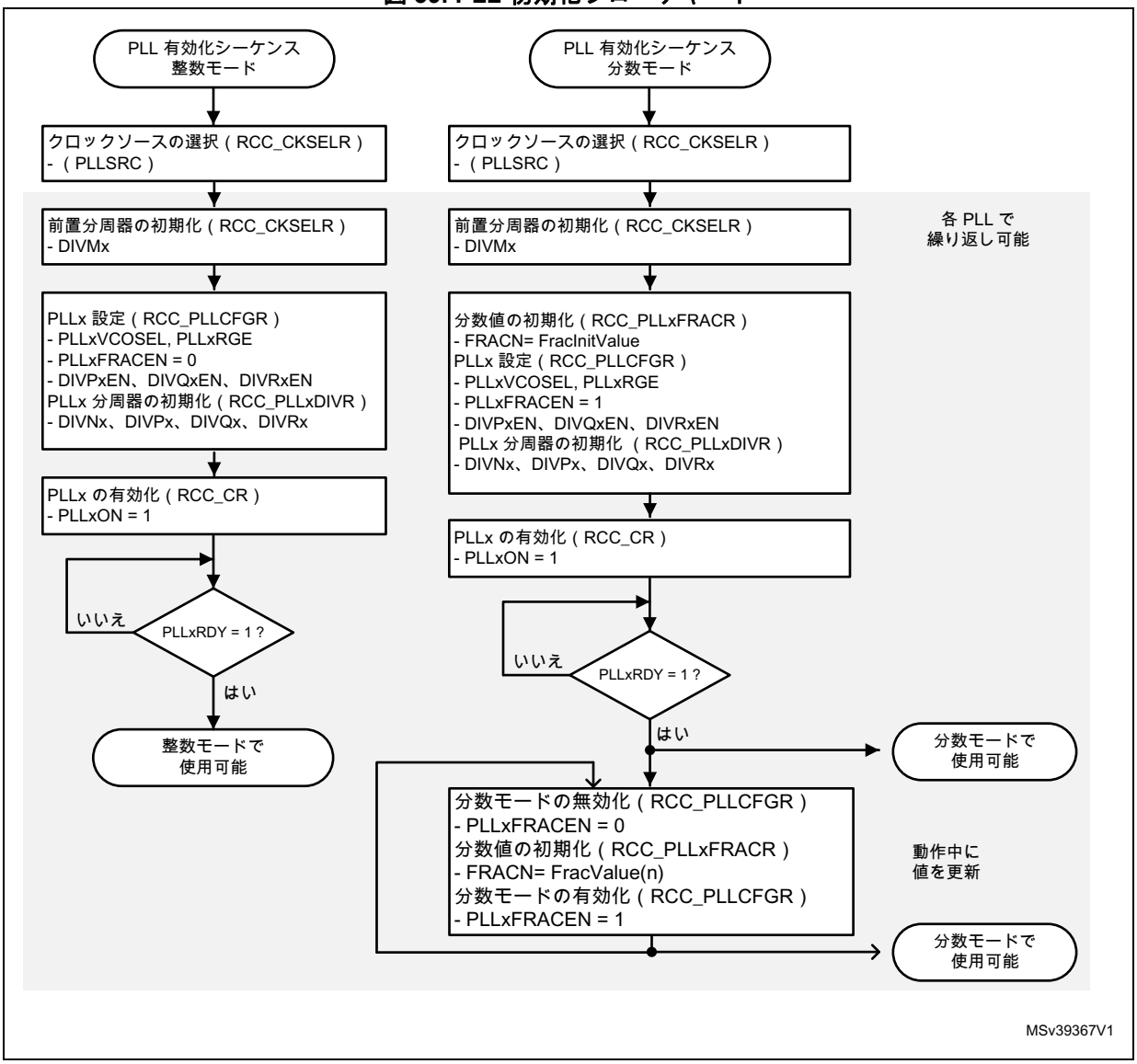
PLL 初期化フェーズ

図 39 に推奨する整数モードまたは分数モードの PLL 初期化シーケンスを示します。PLLx は、初期化シーケンスの始めに無効にします。

1. 必要な周波数に応じて PLL レジスタを初期化します。
   - 整数モードの場合は、*RCC PLL 設定レジスタ（RCC_PLLCFGR）* の PLLxFRACEN を「0」に設定します。
   - 分数モードの場合は、FRACN を必要な初期値（FracInitValue）にセットし、その後 PLLxFRACEN を「1」にセットします。
2. PLLxON ビットを「1」にセットすると、ユーザアプリケーションは PLLxRDY ビットが「1」にセットされるまで待つ必要があります。PLLx が分数モードの場合、PLLxRDY = 「0」である限り PLLxFRACEN ビットを「0」に設定し直すことはできません。
3. PLLxRDY ビットを「1」にセットすると、PLLx を使用できるようになります。
4. アプリケーションで PLLx 周波数を動作中に調整する場合には、以下の条件があります（分数モードのみ）。
   a) PLLxFRACEN を「0」に設定する必要があります。
   PLLxFRACEN = 「0」の場合、シグマデルタモジュレータはまだ SH_REG にラッチされた値で動作しています。
   b) PLLxFRACR （FracValue(n)）に新しい値をロードする必要があります。
   c) シャドウレジスタに PLLxFRACR の内容をラッチするには、PLLxFRACEN は「1」にセットする必要があります。

注： ***PLLxRDY が「1」になると、PLLx 出力周波数とターゲット値の差は±2% 未満になります。***

図 39. PLL 初期化フローチャート



8.5.6 システムクロック（sys_ck）

システムクロックの選択

システムリセット後、HSI はシステムクロックとして選択され、またすべての PLL はオフになります。システムクロックにクロックソースを使用した場合、ソフトウェアでは選択したソースを xxxON ビットによって無効にすることはできません。

もちろん、システムクロックはシステムが STOP または STANDBY モードに移行する際にハードウェアによって停止できます。

システムの実行中に、ユーザアプリケーションでは次の 4 つのソースからシステムクロック（**sys_ck**）を選択できます。

- HSE
- HSI
- CSI
- または pll1_p_ck

この機能は、*RCC クロック設定レジスタ（RCC_CFGR）*をプログラムすることで制御します。あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができている場合（起動遅延時間を経てクロックが安定している状態、または、PLL がロックされている状態）にのみ行われます。準備ができていないクロックソースが選択された場合は、クロックソースの準備ができたときに切り替えが行われます。
*RCC クロック設定レジスタ（RCC_CFGR）*の SWS ステータスビットは、どのクロックが現在システムクロックとして使用されているかを示します。RCC_CR レジスタのその他のステータスビットは、どのクロックがレディ状態かを示します。

システムクロック生成

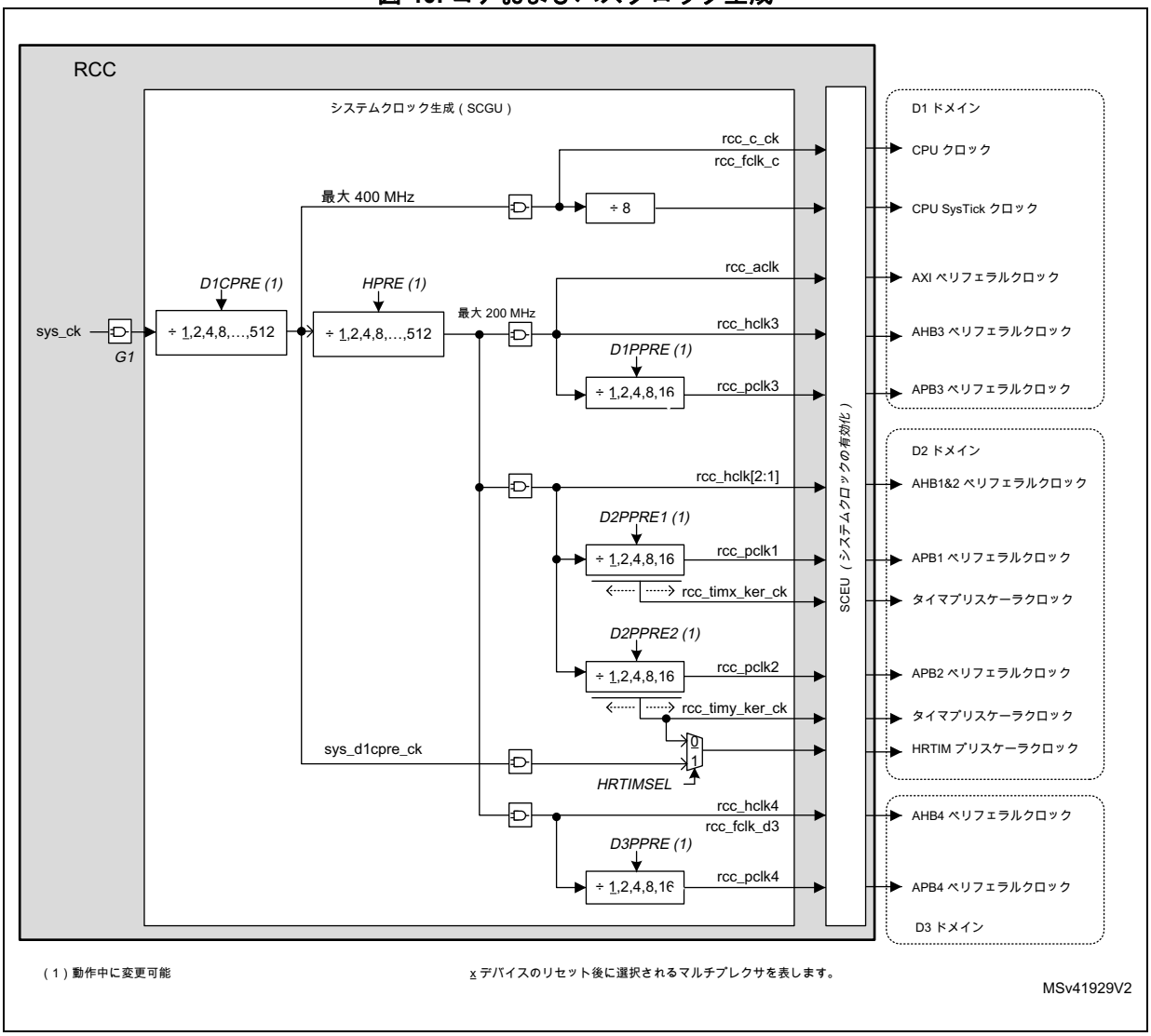
図 40 は、CPU およびバスのクロック分配の簡略化された図です。ブロック図に示された分周器はすべて、タイミング違反を生成することなく動作中に変更できます。この機能により、非常に簡単にバス周波数をアプリケーションのニーズに合わせることができ、これにより消費電力を最適化できます。

D1CPRE 分周器は、CPU クロックを調整するために使用できます。ただし、これはすべてのバスマトリックスおよび HRTIM のクロック周波数にも影響を及ぼします。

同様に、HPRE 分周器は D1 ドメインバスマトリックスのクロックを調整するために使用できますが、これは D2 および D3 ドメインのクロック周波数にも影響を及ぼします。

ほとんどのプリスケーラは、RCC_D1CFGR、RCC_D2CFGR、および RCC_D3CFGR レジスタによって制御されます。

図 40. コアおよびバスクロック生成



このブロックでは、タイマ（**rcc_timx_ker_ck および rcc_timy_ker_ck**）のクロックも供給します。タイマのクロックの周波数は、タイマが接続されたバスの対応する APB プリスケーラ、および TIMPRE ビットに依存します。*表 48* に、タイマクロック周波数の選択方法を示します。

表 48. クロックタイマと pclk の比率

D2PPRE1 (1) D2PPRE2	TIMPRE (2)		$F_{rcc_timx_ker_ck}$ $F_{rcc_timy_ker_ck}$	F_{rcc_pclk1} F_{rcc_pclk2}	コメント
0xx	0	→	F_{rcc_hclk1}	F_{rcc_hclk1}	このタイマクロックはバスクロックと同一です。
100	0	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 2	このタイマクロックの速さはバスクロックの 2 倍です。
101	0	→	F_{rcc_hclk1} / 2	F_{rcc_hclk1} / 4	
110	0	→	F_{rcc_hclk1} / 4	F_{rcc_hclk1} / 8	
111	0	→	F_{rcc_hclk1} / 8	F_{rcc_hclk1} / 16	
0xx	1	→	F_{rcc_hclk1}	F_{rcc_hclk1}	このタイマクロックはバスクロックと同一です。

表 48. クロックタイマと pclk の比率（続き）

D2PPRE1 (1) D2PPRE2	TIMPRE (2)		$F_{rcc_timx_ker_ck}$ $F_{rcc_timy_ker_ck}$	F_{rcc_pclk1} F_{rcc_pclk2}	コメント
100	1	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 2	このタイマクロックの速さはバスクロックの 2 倍です。
101	1	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 4	このタイマクロックの速さはバスクロックの 4 倍です。
110	1	→	F_{rcc_hclk1} / 2	F_{rcc_hclk1} / 8	
111	1	→	F_{rcc_hclk1} / 4	F_{rcc_hclk1} / 16	

1. D2PPRE1 および D2PPRE2 は *RCC ドメイン 2 クロック設定レジスタ（RCC_D2CFGR）* に属します。
2. TIMPRE は *RCC クロック設定レジスタ（RCC_CFGR）* に属します。

8.5.7 STOP モードおよび STANDBY モードでのクロック生成の処理

システム全体が STOP モードに移行すると、すべてのクロック（システムおよびカーネルクロック）と次のクロックソースが停止します。

- CSI、HSI （HSIKERON および CSIKERON ビットによる）
- HSE
- PLL1, PLL2 & PLL3
- HSI48

RCC レジスタの内容は、「0」に設定される PLL1ON、PLL2ON、PLL3ON、HSEON、および HSI48ON を除いて変更されません。

STOP モードの終了

マイクロコントローラがウェイクアップイベントによってシステム STOP モードを終了すると、アプリケーションでは再起動時に使用するオシレータ（HSI および／または CSI）を選択できます。STOPWUCK ビットがシステムクロックとして使用されるオシレータを選択します。STOPKERWUCK ビットがペリフェラルのカーネルクロックとして使用されるオシレータを選択します。STOPKERWUCK ビットは、システムクロックとは異なるオシレータによって生成されたカーネルクロックを、ペリフェラルがシステム STOP 後に必要とする場合に有用です。

ビットはすべて、*RCC クロック設定レジスタ（RCC_CFGR）* に属します。*表 49* には、その挙動の詳細な説明が記載されています。

表 49. STOPWUCK と STOPKERWUCK の説明

STOPWUCK	STOPKERWUCK		システム STOP モード終了時の有効化されたオシレータ	システム STOP モード終了時の分配されたクロック	
				システムクロック	カーネルクロック
0	0	→	HSI	HSI	HSI
	1	→	HSI および CSI		HSI および／または CSI
1	0	→		CSI	
	1	→	CSI		CSI

STOP モード時

システム STOP モード中に HSI または CSI を有効化できる 2 つの具体的な例があります。

- 専用のペリフェラルがカーネルクロックを要求した場合：

  この場合、ペリフェラルはそれ自体のために PERxSRC によって選択されたカーネルクロックソースに応じて HSI または CSI を受信します。

- HSIKERON または CSIKERON ビットがセットされている場合：

  この場合、HSI および CSI は STOP モード中に動作状態を維持しますが、出力はゲートされます。このように、クロックは STOP モードが終了した場合や、ペリフェラルがカーネルクロックを要求した場合に、ただちに使用できるようになります（詳細については、*表 50* を参照）。

HSIKERON および CSIKERON ビットは、*RCC ソース制御レジスタ（RCC_CR）*に属します。*表 50* には、その挙動の詳細な説明が記載されています。

表 50. HSIKERON および CSIKERON の動作

HSIKERON（CSIKERON）		STOP モード時の HSI （CSI）状態	HSI （CSI）の設定時間
0	→	オフ	$t_{su(HSI)}$ （$t_{su(CSI)}$）(1)
1	→	動作し、ゲートされる	即時

1. $t_{su(HSI)}$ および $t_{su(CSI)}$ は HSI および CSI オシレータの起動時間です（これらのパラメータの値については、製品データシートを参照）。

マイクロコントローラがシステム STANDBY モードを終了すると、HSI はシステムおよびカーネルクロックとして選択され、RCC レジスタは RCC_RSR （または RCC_C1_RSR）および RCC_BDCR レジスタを除いて初期値にリセットされます。

HSI および CSI 出力によって 2 つのクロックパスが供給される点にも注目してください（*図 36* を参照）。

- 1 つはシステムクロック用（**hsi_ck** または **csi_ck**）のパス
- 1 つはペリフェラルカーネルクロック用（**hsi_ker_ck** または **hsi_ker_ck**）のパス

ペリフェラルがシステム STOP モードでカーネルクロックを要求した場合、**hsi_ker_ck** または **csi_ker_ck** を提供するパスのみがアクティブになります。

注意： ***CStop モードから移行する際に CPU が自動的に同じクロック周波数を得るかどうかはわかりません。これは主にシステムの状態に依存します。たとえば、D3 ドメインが CRun モードで保持されている場合に CPU が CStop モードに移行すると、CPU が CStop モードを終了する際にクロック設定は変更されません。CPU も CStop モードである場合に D3 ドメインが CStop モードに移行すると、CPU は CStop モード終了後に HSI または CSI とともに動作します。***

8.5.8 カーネルクロックの選択

一部のペリフェラルは、非同期的に動作する 2 つの異なるクロックドメインと動作するよう設計されています。

- レジスタおよびバスインタフェースと同期するクロックドメイン（**ckg_bus_perx** クロック）
- 通常ペリフェラルと同期するクロックドメイン（カーネルクロック）

これらの 2 つのクロックドメインをサポートするペリフェラルを持つ利点は、ユーザアプリケーションで CPU、バスマトリックス、およびペリフェラルのカーネル部分の最適化されたクロック周波数をより自由に選択できるという点です。

結果として、ユーザアプリケーションでは、ペリフェラルを再プログラミングすることなくバス周波数を変更できます。たとえば、APB クロックを動作中に変更した場合、UART を使用した動作中の転送は妨害されません。

表 51 に、RCC がペリフェラルに供給可能なカーネルクロックを示します。*表 51* の各行は、出力に接続された MUX およびペリフェラルを表します。4 列目以降はクロックソースを表します。3 列目は MUX 出力ごとの最大許容周波数を表します。これらの要件に従うかどうかはユーザに委ねられています。

参考資料

表 51. カーネルクロック分配の概要

ペリフェラル	クロックマルチプレクサ制御ビット	最大許容周波数 [MHz]	ドメイン	クロックソース																		
				pll1_q_ck	pll2_p_ck	pll2_q_ck	pll2_r_ck	pll3_p_ck	pll3_q_ck	pll3_r_ck	sys_ck	バスクロック (1)	hse_ck	hsi_ker_ck	csi_ker_ck	hsi48_ck	lse_ck	lsi_ck	per_ck (2)	I2S_CKIN	USB_PHY1/2	無効
LTDC	-	66	D1							x												
FMC	FMCSEL	200		1			2					0							3			
QUADSPI	QSPISEL	200		1			2					0							3			
SDMMC1	SDMMCSEL	200 (4)		0			1															
SDMMC2		200	D2																			
DFSDM1 Aclk	SAI1SEL	133		0	1			2											4	3		
DFSDM1 clk	DFSDM1SEL	133									1	0										
FDCAN	FDCANSEL	100		1		2							0									
HDMI-CEC	CECSEL	66													2 (3)		0	1				
I2C1、2,3	I2C123SEL	100								1		0		2	3							
LPTIM1	LPTIM1SEL	100			1					2		0					3	4	5			
TIM[8:1]、TIM[17:12]	-	200										x										
HRTIM	-	400										x										
RNG	RNGSEL	66		1												0	2	3				
SAI1	SAI1SEL	133		0	1			2											4	3		
SAI2	SAI23SEL	133		0	1			2											4	3		
SAI3		133																				
SPDIFRX	SPDIFSEL	200		0			1			2				3								
SPI(I2S)1,2,3	SPI123SEL	200		0	1			2											4	3		
SPI4、5	SPI45SEL	100				1			2			0	5	3	4							
SWPMI	SWPSEL	100										0		1								
USART1、6	USART16SEL	100				1			2			0		3	4		5					
USART2、3 UART4,5,7,8	USART234578SEL	100				1			2			0		3	4		5					
USB1OTG USB2OTG	USBSEL	66		1					2							3						0
USB1ULPI	-	100																			x	
USB2ULPI	-	100																			x	
ADC1、2	ADCSEL	25 (4)			0					1									2			
ADC3			D3																			

表 51. カーネルクロック分配の概要（続き）

ペリフェラル	クロックマルチプレクサ制御ビット	最大許容周波数 [MHz]	ドメイン	クロックソース																	
				pll1_q_ck	pll2_p_ck	pll2_q_ck	pll2_r_ck	pll3_p_ck	pll3_q_ck	pll3_r_ck	sys_ck	バスクロック (1)	hse_ck	hsi_ker_ck	csi_ker_ck	hsi48_ck	lse_ck	lsi_ck	per_ck (2)	I2S_CKIN	USB_PHY1/2	無効
I2C4	I2C4SEL	100	D3							1		0		2	3						
LPTIM2	LPTIM2SEL	100			1					2		0					3	4	5			
LPTIM3、4,5	LPTIM345SEL	100			1					2		0					3	4	5			
LPUART1	LPUART1SEL	100				1			2			0		3	4		5					
SAI4_A	SAI4ASEL	133		0	1			2											4	3		
SAI4_B	SAI4BSEL	133		0	1			2											4	3		
SPI6	SPI6SEL	100				1			2			0	5	3	4							
RTC/AWU	RTCSEL	1	VSW										3 (5)				1	2				0

1. バスクロックは、ペリフェラルが接続されたバスインタフェースクロックで、APB、AHB、または AXI クロックのいずれかです。
2. per_ck クロックは、CKPERSEL の選択に従って、hse_ck、hsi_ker_ck、または csi_ker_ck です。
3. クロック CSI は 122 分周されます。
4. デューティサイクルが約 50% の場合、DIV[P/Q/R]x 値は偶数になります。SDMMCx では、DDR をサポートするデューティサイクルは 50% になります。
5. クロック HSE は RTCPRE で分周されます。

図 41 から *図 50* では、カーネルクロック分配のより詳細な説明を提供しています。図を簡略化するため、イネーブル信号によってゲートされている場合でも、バスインタフェースクロック（pclk、hclk）は表示されません。詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御* を参照してください。

スイッチ数を低減するために、一部のペリフェラルでは同じカーネルクロックソースを共用しています。ただし、すべてのペリフェラルは専用のイネーブル信号を持っています。

オーディオアプリケーション専用のペリフェラル

SPDIFRX 以外のオーディオペリフェラルには通常、特定の正確な周波数が必要です。*図 41* に示すように、SAI または SPI(I2S) のカーネルクロックは、以下によって生成できます。

- PLL1、アクティブな PLL の数を減らす必要がある場合
- PLL2 または 3、周波数生成時に最適な柔軟性を得るため
- HSE、HSI、または CSI、消費電流が重要な使用例の場合
- I2S_CKIN、外部クロックリファレンスを使用する必要がある場合

注： ***SPDIFRX は特定の周波数を必要としませんが、ペリフェラルが適切に動作するよう、カーネルクロック周波数だけは十分な高さにする必要があります。詳細については、SPDIFRX の説明を参照してください。***

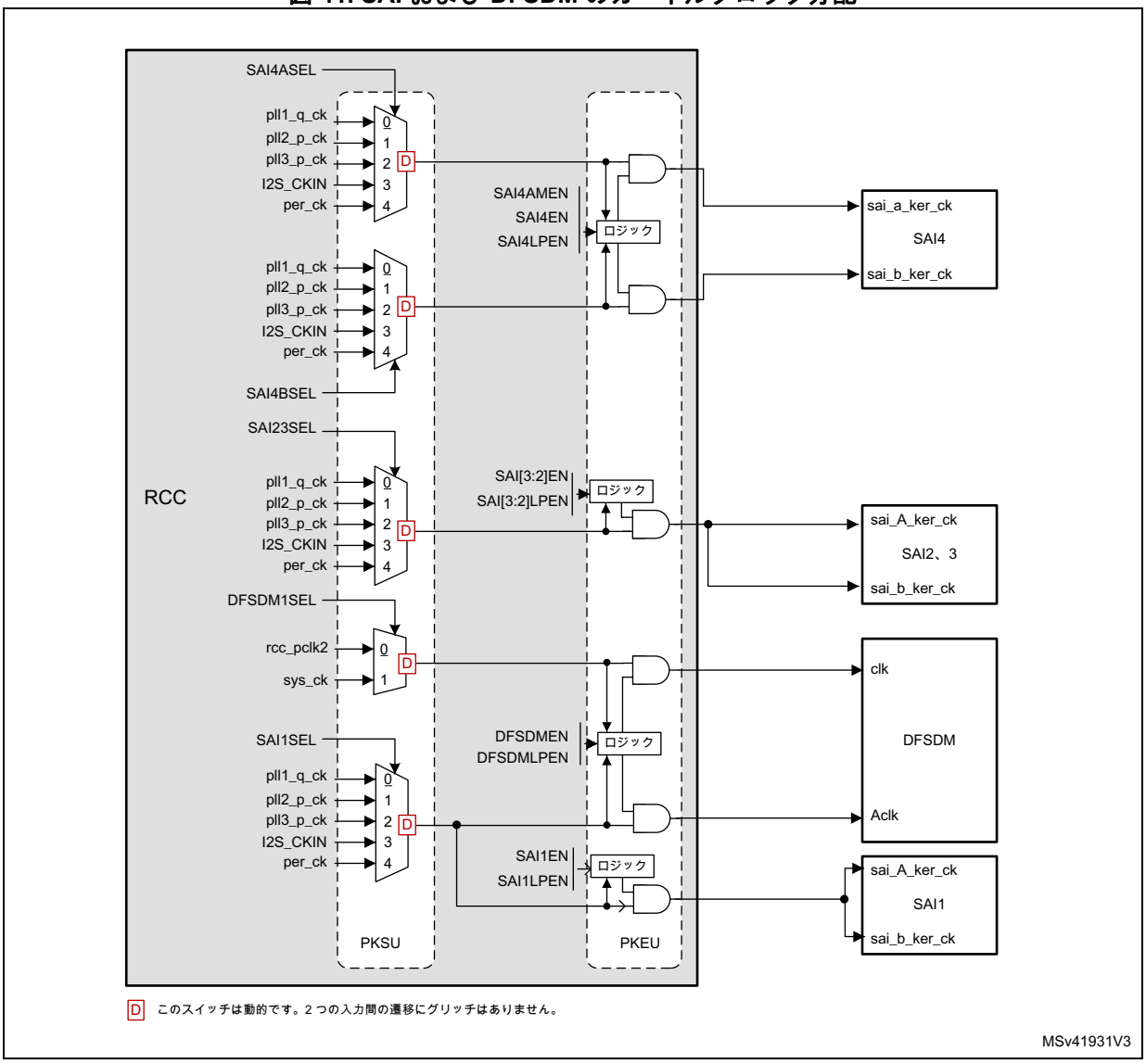
DFSDM1 は、SAI1A と同じクロックを使用できます。これは、DFSDM1 をオーディオアプリケーションで使用する場合に有用です。

柔軟性を向上するため、SAI4 では各サブブロックに異なるクロックを使用できます。

SPI/I2S1、2、および 3 では、同じカーネルクロックソースを共用しています（図 42 を参照）。

参考資料

図 41. SAI および DFSDM のカーネルクロック分配



1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

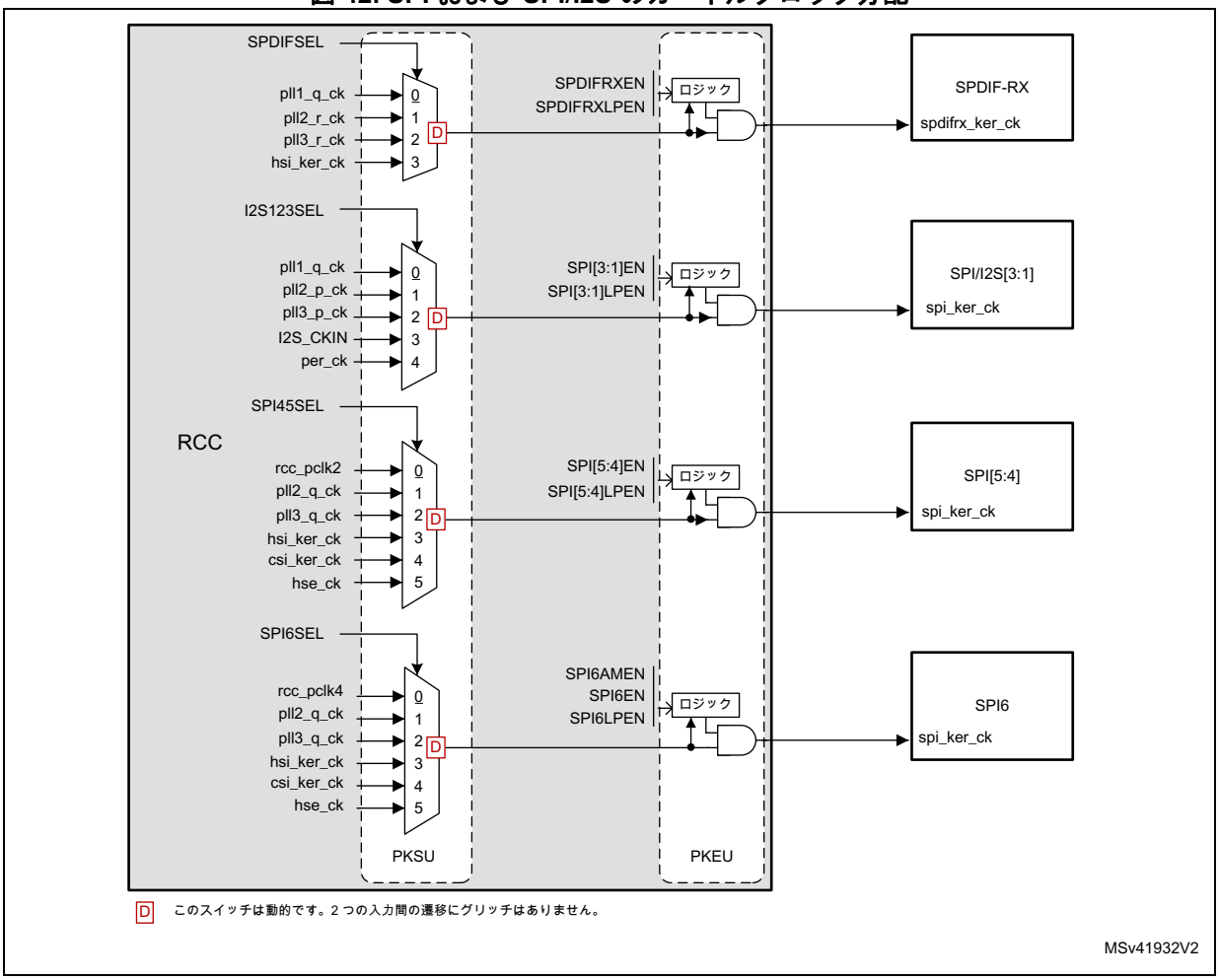
制御およびデータ転送専用のペリフェラル

SPI、I2C、および UART では、正しいボーレートを生成可能な速度のクロック、またはシリアルインタフェースで必要なビットクロックを除き、特定のカーネルクロック周波数を必要としません。このため、次のソースから選択できます。

- PLL1、アクティブな PLL の数を減らす必要がある場合
- PLL2 または PLL3、より高い柔軟性が必要な場合。たとえば、この 2 つの方法では、一部のシリアルインタフェースの速度に影響を与えずに、PLL1 によって周波数バスを変更できます。
- HSI または CSI、低電力の使用例、またはペリフェラルを迅速に STOP モードからウェイクアップする必要がある場合（UART、I2C など）

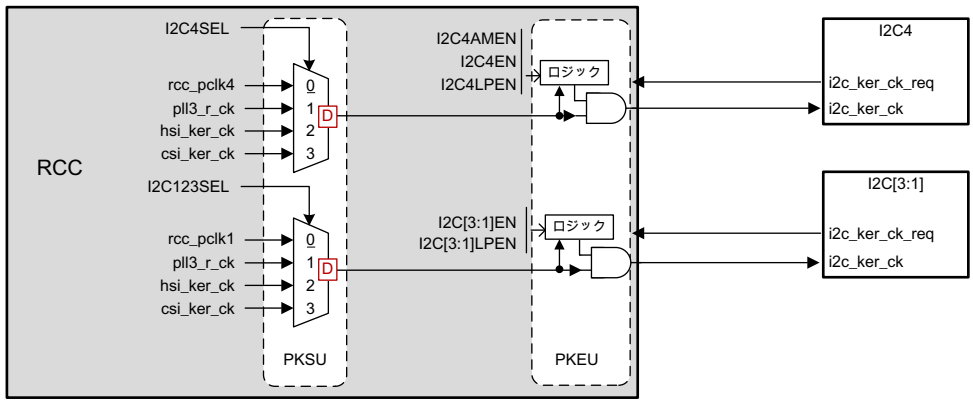
注： ***高いボーレートが必要ではない場合、UART では LSE クロックが必要です。***

図 42. SPI および SPI/I2S のカーネルクロック分配



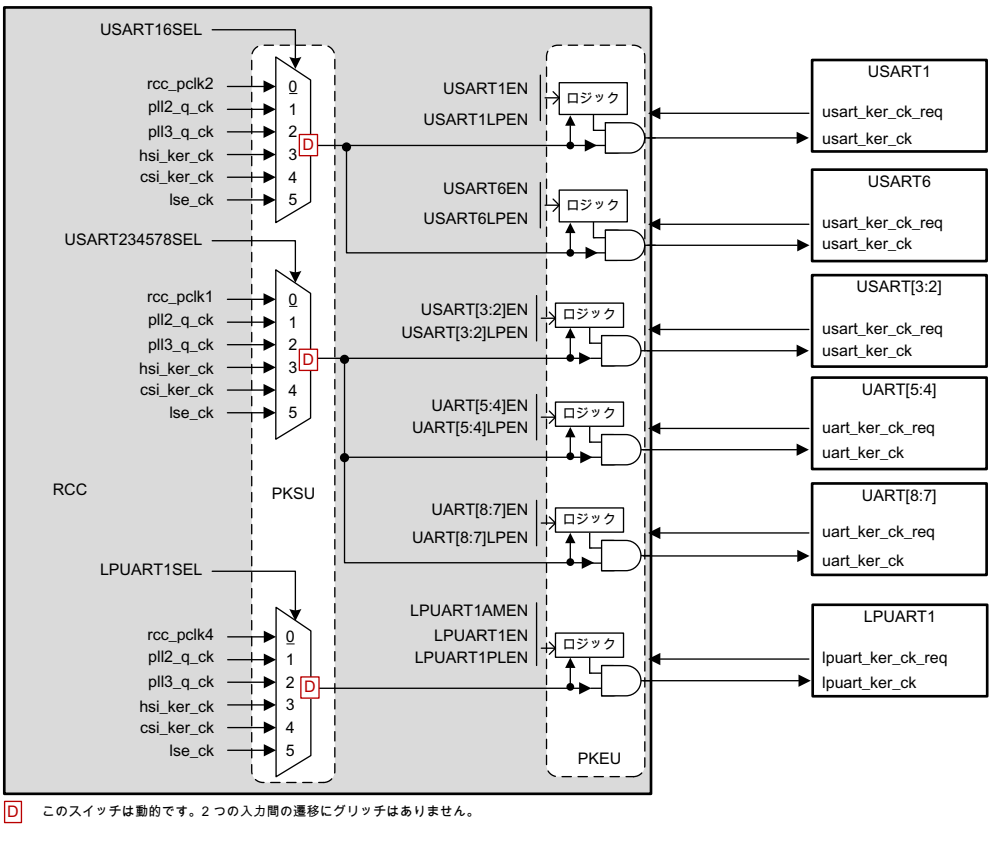
1. **X** は、デバイスのリセット後に選択される MUX を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

図 43. I2C のカーネルクロック分配



1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

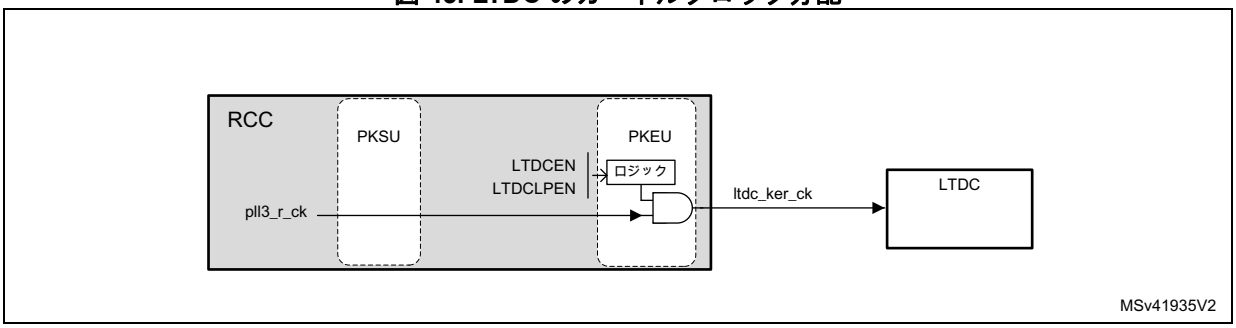
図 44. UART、USART、および LPUART1 のカーネルクロック分配



1. **X** は、デバイスのリセット後に選択される MUX を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

参考資料

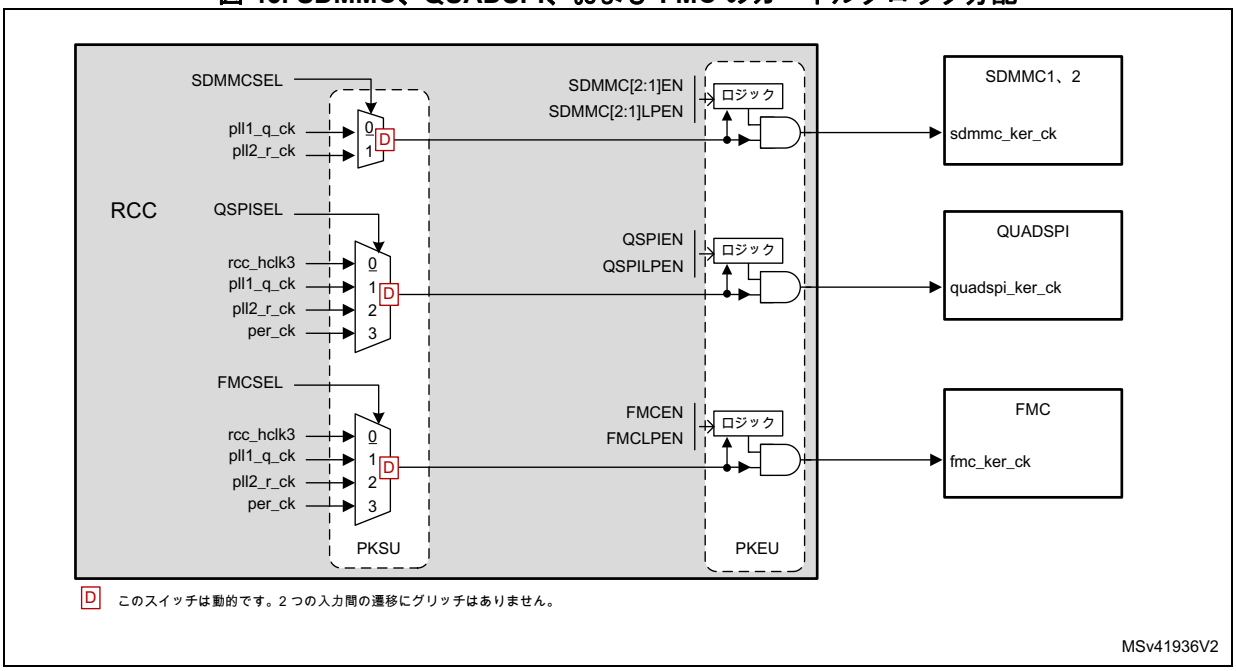
図 45. LTDC のカーネルクロック分配



1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

FMC、QUADSPI、および SDMMC1/2 では、より高い柔軟性を得るために、バスインタフェースクロックとは異なるクロックを使用することもできます。

図 46. SDMMC、QUADSPI、および FMC のカーネルクロック分配

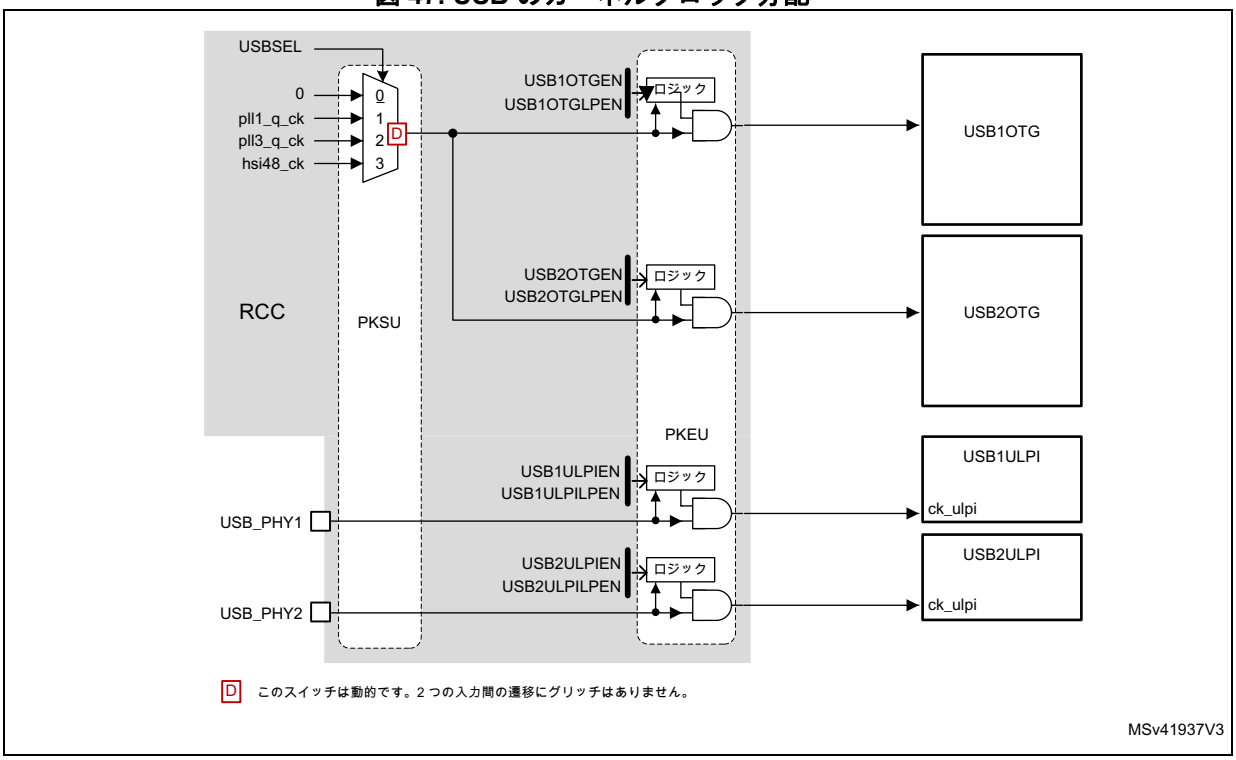


1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

図 47 に、USB ブロックのクロック分配を示します。USBxULPI ブロックは、外部 PHY からクロックを受信します。

USBxOTG ブロックは、MUX で制御される USBSEL により、さまざまなソースから USB 通信のクロックを受信できます。

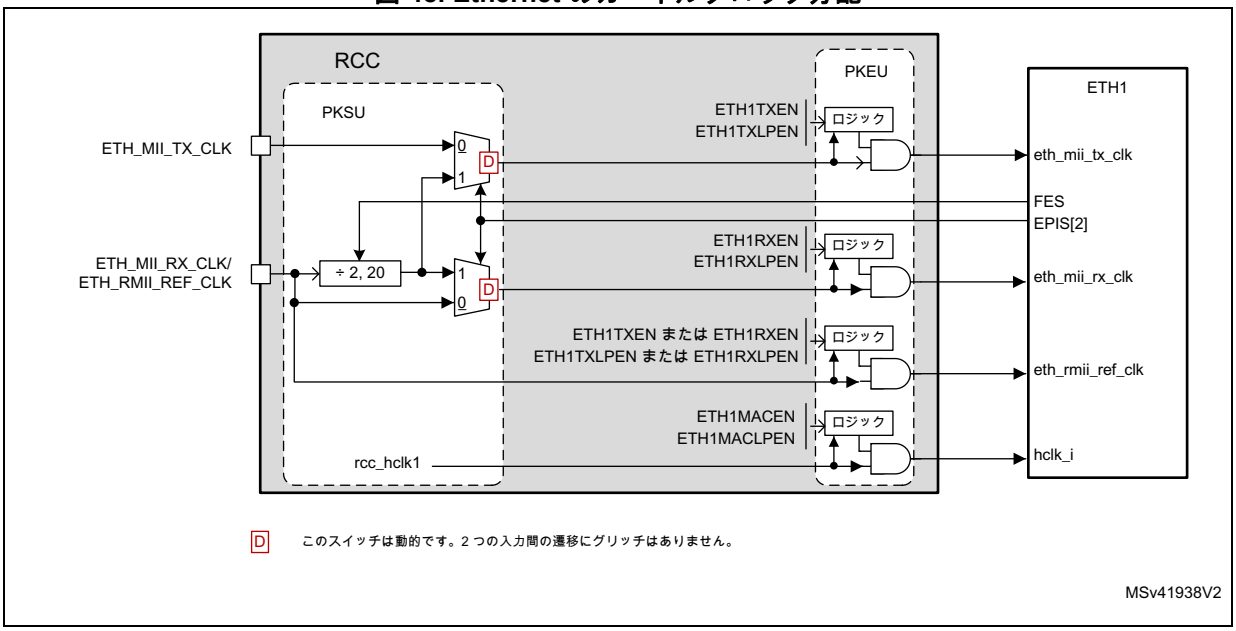
図 47. USB のカーネルクロック分配 (2)



1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

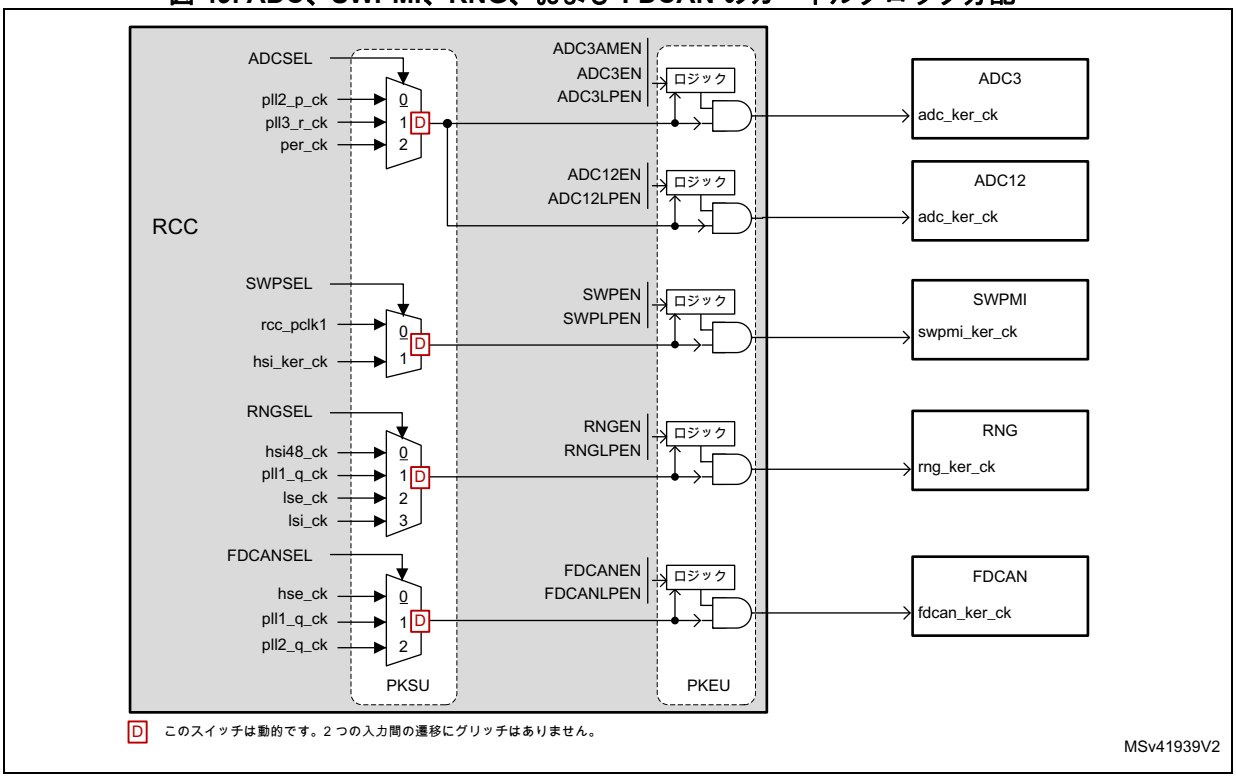
Ethernet で送受信されたクロックは、外部 Ethernet PHY から供給される必要があります。RX および TX パスのクロック選択は、SYSCFG ブロックによって制御されます。

図 48. Ethernet のカーネルクロック分配



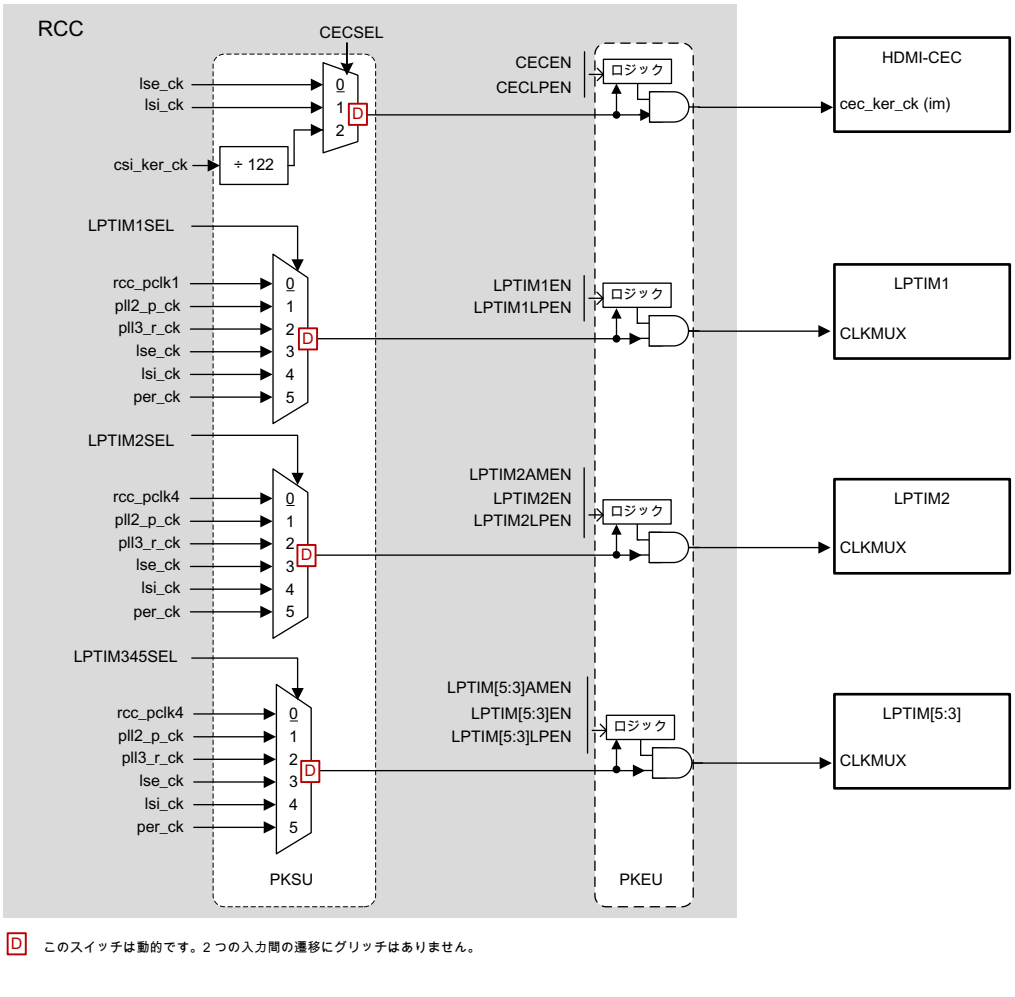
1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

図 49. ADC、SWPMI、RNG、および FDCAN のカーネルクロック分配 (2)



1. **X** は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

図 50. LPTIM および HDMI-CEC のカーネルクロック分配 (2)



1. X は、システムのリセット後に選択される MUX 入力を表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

RTC/AWU クロック を参照してください

rtc_ck クロックソースは次のいずれかです。

- **hse_1M_ck**（**hse_ck** をプログラム可能なプリスケーラで分周）
- **lse_ck**
- または、**lsi_ck** クロック

このソースクロックは、*RCC バックアップドメイン制御レジスタ（RCC_BDCR）*の RTCSEL[1:0] ビットおよび *RCC クロック設定レジスタ（RCC_CFGR）*の RTCPRE[5:0] ビットをプログラムすることによって、選択できます。

この選択は、バックアップドメインをリセットしない限り変更できません。

RTC クロックとして LSE が選択されている場合、バックアップまたは V_{DD} 電源が無くなっても RTC は正常に動作します。

LSE クロックはバックアップドメインに属しますが、その他のオシレータはそうではありません。その結果、以下のようになります。

- LSE が RTC クロックとして選択されている場合、V_{DD} 供給がオフになった場合でも、V_{BAT} の供給が保たれている限り、RTC は動作を続けます。
- LSI が RTC クロックとして選択されている場合、V_{DD} 供給がオフになると、AWU の状態は保証されません。
- HSE クロックが RTC クロックとして選択されている場合、V_{DD} 供給または V_{CORE} 供給がオフになると、RTC の状態は保証されません。

rtc_ck クロックは *RCC バックアップドメイン制御レジスタ（RCC_BDCR）* の RTCEN ビットを使用して有効化されます。

RTC バスインタフェースクロック（APB クロック）は、RCC_APB4ENR/LPENR レジスタの RTCAPBEN および RTCAPBLPEN ビットを使用して有効化されます。

注： ***APB クロック周波数が RTC クロック周波数の 7 倍 未満である場合（$F_{APB} < 7 \times F_{RTCLCK}$）に RTC カレンダレジスタを読み込むには、ソフトウェアはカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR への 2 回目の読み出しアクセスの結果が最初の読み出しアクセスの結果と同じであれば、データは正しいということになります。同じでない場合は、3 回目の読み出しアクセスを実行する必要があります。***

ウォッチドッグクロック

RCC では、回路で使用可能な 4 つのウォッチドッグブロックにクロックを供給します。独立型ウォッチドッグ（IWDG1）は、LSI に接続されます。ウィンドウ型ウォッチドッグ（WWDG1）は、APB クロックに接続されます。

独立型ウォッチドッグがハードウェアのオプションまたはソフトウェアアクセスによって開始された場合、LSI は強制的にオンになり、オフにすることはできません。LSI オシレータのセットアップ遅延後、このクロックが IWDG に供給されます。

注意： ***WWDG1 を有効にする前に、アプリケーションでは WW1RSC ビットを「1」にセットする必要があります。WW1RSC が「0」のままの場合、WWDG1 を有効にしたときの動作は保証されません。WW1RSC ビットは、RCC グローバル制御レジスタ（RCC_GCR）にあります。***

TIMx を使用したクロック周波数の測定

クロックソースジェネレータのほとんどの周波数は、TIMx の入力キャプチャで測定できます。

- LSE による HSI または CSI の較正

  LSE を TIMx の入力キャプチャに接続しておく主な目的は、HSI または CSI を正確に測定できるようにすることです。このとき、HSI または CSI を、システムクロックソースとして、直接または PLL1 を通じて使用する必要があります。LSE 信号の連続したエッジ間のシステムクロックカウント数により、内部クロック周期が測定されます。LSE クリスタルの高い精度（通常は数十 ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造工程、温度、電圧に関連した周波数偏差に対してクロックソースを調整して補正することができます。

  この基本概念は、相対測定（HSI/LSE 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接につながっています。この比が大きければ大きいほど、測定精度は向上します。

  HSI および CSI オシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります（*RCC 内部クロックソース較正レジスタ（RCC_ICSCR）* を参照）。PLLx によって HSI または CSI が使用されると、PLL の分数分周器を使用してシステムクロックを微調整することもできます。

- HSI による LSI の較正

  LSI 周波数も測定できます。クリスタルを使用しないアプリケーションで有用です。超低電力 LSI オシレータには、大きな製造工程偏差があります。これを HSI クロックソースで測定することにより、HSI の精度で LSI の周波数を判断することができます。測定値を使用して、より正確な RTC 時間基準のタイムアウト（LSI が RTC クロックソースとして使用されているとき）や許容精度の IWDG タイムアウトを得ることができます。

8.5.9 汎用クロックの概念の概要

RCC では、システムの CPU、バスインタフェース、およびペリフェラルクロック（D1、D2、および D3 ドメイン）の分配を各機能の動作モードに従って処理します（クロック定義の詳細については、*セクション 8.5.1：クロックの命名規則*を参照）。

アプリケーションは、ペリフェラルごとにカーネルとバスインタフェースクロックの有効化／無効化を制御できます。ペリフェラルを使用する前に、CPU ではこれを有効化（PERxEN を「1」にセット）する必要があり、またこのペリフェラルが CSleep モードでアクティブなままの場合は定義（PERxLPEN を「1」にセット）する必要があります。これは、CPU へのペリフェラルの「割り当て」と呼ばれます（詳細については、*セクション 8.5.10：ペリフェラルの割り当て*を参照）。

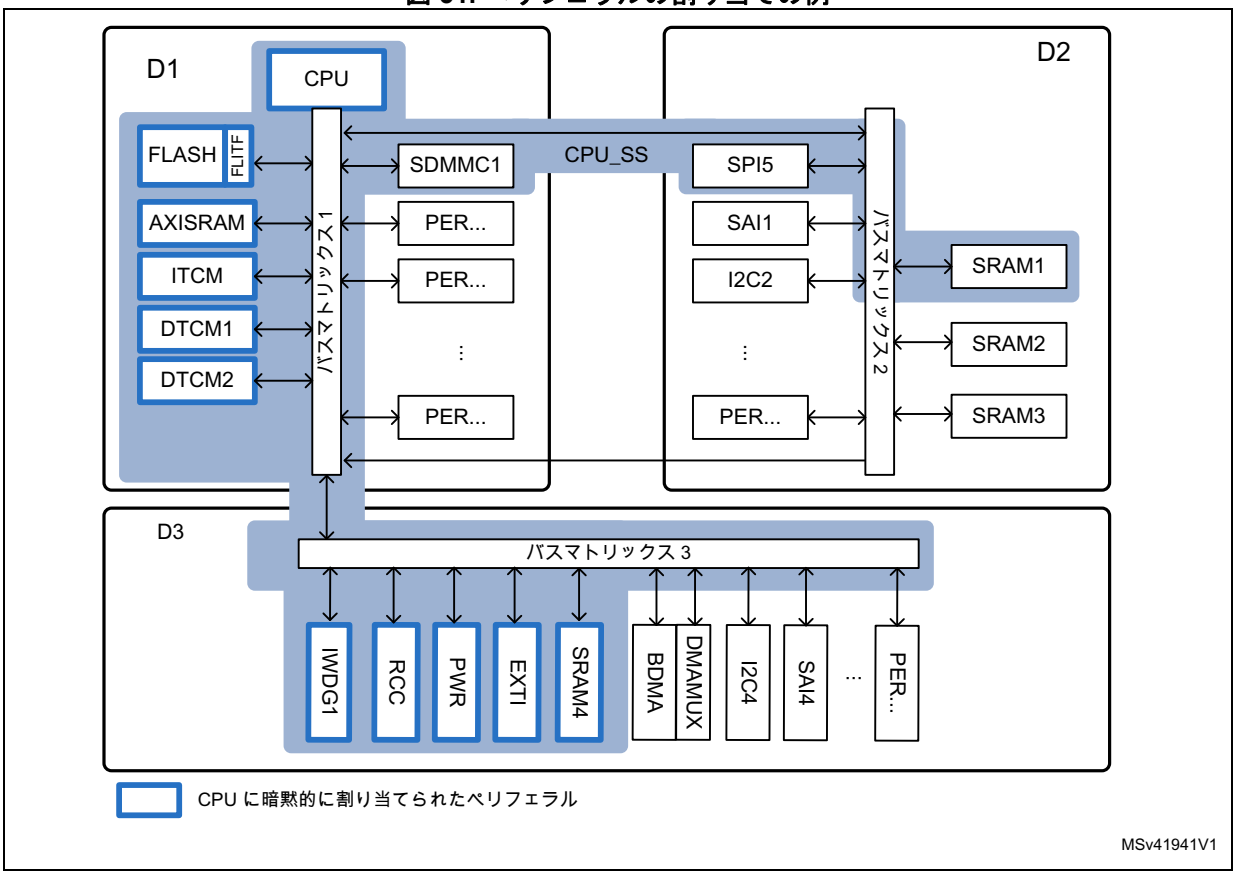
ペリフェラルの割り当ては、RCC が CPU およびドメインモードに従ってクロックゲーティングを自動的に制御する場合、さらに PWR が D1、D2、および D3 ドメインの電源電圧を制御する場合に使用されます。

図 51 に、ペリフェラルの割り当ての例を示します。

- CPU によって有効化された SDMMC1、SPI5、および SRAM1、AXISRAM、ITCM、DTCM1、DTCM2、および SRAM4 は、暗黙的に CPU に割り当てられます。CPU、バスマトリックス 1/2/3、および割り当てられたペリフェラルから成るグループにより、サブシステム（CPU_SS）が構成されます。

注： ***FLASH、AXISRAM、ITCM、DTCM1、DTCM2、SRAM4、IWGD1、IWGD2、PWR、EXTI、および RCC は共通のリソースで、CPU に暗黙的に割り当てられています。***

図 51. ペリフェラルの割り当ての例



CPU が CStop モードに移行すると、RCC では CPU_SS と CPU クロックのすべてのペリフェラルのバスインタフェースとカーネルクロックを自動的に無効化します。PLL が有効化されている場合は、D3 がまだ実行中であるため、RCC で無効化することはできません。

D3 ドメインは、CPU が CStop モードで、D1 および D2 ドメインが DStop または DStandby モードの場合、DRun モードを保持できます。これには、PWR_CPUCR レジスタの RUN_D3 ビットをセットします。

- RUN_D3 を「1」にセットすると、D3 は CPU モードとは別に DRun モードのまま保持されます（*PWR CPU 制御レジスタ（PWR_CPUCR）*を参照）。
- RUN_D3 を「0」に設定すると、CPU の CStop モードへの移行時に、D3 ドメインは DStop または DStandby モードに移行します（*表 52* を参照）。

条件を満たしている場合、CPU は、*PWR CPU 制御レジスタ（PWR_CPUCR）*の PDDS_D1、PDDS_D2、および PDDS_D3 ビットを使用して、D1、D2、または D3 ドメインが DStandby に移行可能かどうかを制御できる点に注意してください。

ウェイクアップイベントでは DStandby または DStop モードから D1、D2、または D3 ドメインを終了できるようになります。

また、D3 ドメインに割り当てられた一部のペリフェラルに、さらなる自律性を与えることができます（詳細については、*セクション：D3 ドメインでの自律モード*を参照）。

D3 ドメインでの自律モード

自律モードでは、CPU が CStop モードの場合でも、D3 に割り当てられたペリフェラルにペリフェラルクロックを供給できます。ペリフェラルの自律ビットが有効化されている場合、ペリフェラルは D3 ドメインの状態に従ってペリフェラルクロックを受信します（CPU が CStop モードの場合）。

- D3 ドメインが DRun モードの場合、自律モードが有効化されているペリフェラルではペリフェラルクロックを受信します。
- D3 ドメインが DStop モードの場合、ペリフェラルクロックは供給されません。

自律モードによって、D3 ドメインが DStop または DStandby モードに移行できなくなることはありません。

自律ビットは、*RCC D3 自律モードレジスタ（RCC_D3AMR）*に割り当てられています。

たとえば、BDMA を介して外部デバイスから受信したデータを SAI4 が SRAM4 に書き込んでいる間に、CPU は CStop モードに移行できます。受信するデータの量に達した場合、ウェイクアップイベントによって CPU を有効にできます。これには、D3 を DRun モード（RUN_D3 に「1」をセット）に保持したまま、SAI4、BDMA、および SRAM4 を自律モードに設定します。この例では、D3 ドメインは常に DRun モードであるため、RCC では PLL をオフにしません。

消費電源の低減においては、自律モードと一部のペリフェラル（UART、I2C）の機能を組み合わせて、CPU をウェイクアップすることなく独自にカーネルクロックを要求するなど、さらなる応用が可能です。たとえば、システムが I2C4 を介してメッセージを待っている場合は、システム全体を STOP モードにすることができます。I2C4 ペリフェラルクロックは START ビットを検出すると、「カーネルクロックリクエスト」を生成します。このリクエストによって HSI または CSI が有効化され、カーネルクロックはリクエスト元（この例では I2C4）にのみ供給されます。その後、I2C4 は受信メッセージをデコードします。次のいくつかのケースが可能です。

- メッセージのデバイスアドレスが一致しない場合、I2C4 では新しい START コンディションが検出されるまで「カーネルクロックリクエスト」を解除します。
- 受信メッセージのデバイスアドレスが一致しない場合は、D3 のローカルメモリに格納する必要があります。I2C4 では、アドレス一致時に D3 ドメインを DRun モードに切り替えるためにウェイクアップイベントを生成することができます。その後、メッセージは BDMA を介してメモリに転送され、D3 ドメインは CPU を有効化せずに DStop モードに戻ります。メモリに転送されたデータ量が転送回数に達した場合、BDMA では CPU をウェイクアップする割り込みも生成できます。
- 受信メッセージのデバイスアドレスが一致し、ペリフェラルが CPU をウェイクアップするために設定された場合、I2C4 では CPU を有効化するためのウェイクアップイベントを生成します。

どのペリフェラルがどのドメインに対してウェイクアップイベントを生成できるかについては、EXTI ブロックの説明を参照してください。

メモリの処理

CPU は、製品の使用可能なすべてのメモリ領域にアクセスできます。

- AXISRAM、ITCM、DTCM1、DTCM2、および FLASH
- SRAM1、SRAM2、および SRAM3
- SRAM4 および BKPRAM

図 51 に示すように、FLASH、AXISRAM、SRAM4、ITCM、DTCM1、および DTCM2 は、暗黙的に CPU に割り当てられます。このため、CPU がこれらのメモリを割り当てるためのイネーブルビットはありません。

CPU が D2 ドメインに割り当てられたメモリ（SRAM1、SRAM2、および SRAM3）を使用するには、これらを有効にする必要があります。

BKPRAM には、バスインタフェースクロックをゲートするための専用の有効ビットがあります。CPU では、これを使用する前に、BKPRAM を有効にする必要があります。

注： ***メモリインタフェースクロック（フラッシュおよびRAM インタフェース）は、ソフトウェアによってCSleep モード中に停止できます（DxSRAMyLPEN ビットを使用）。***

クロックの有効化の詳細については、*ペリフェラルクロックゲーティングの制御*および*CPU およびバスマトリックスクロックゲーティングの制御*の各セクションを参照してください。

システムの状態の概要

表 52 に、D1、D2、および D3 ドメインモードに対するシステムの状態の概要について示します。

- D3 が DRun モードである限り、システムは RUN モードを維持します。ここでは説明しませんが、システム RUN モードのいくつかのサブステートが存在します。（詳細については、*電源制御（PWR）*を参照）。
- D1 ドメインが DRun の場合、D2 ドメインは DRun、DStop、または DStandby です。D1 ドメインが DStop または DStandby の場合、D2 ドメインは DRun を維持できなくなり、PDDS_D2 ビットに従って DStop または DStandby に切り替わります。
- D3 ドメインは、PWR_CPUCR レジスタの RUN_D3 ビットによってD1 および D2 ドメインが DStop/DStandby モードである間、または自律モードである場合に、実行できます。
- D3 が DStop モードである限り、システムは STOP モードを維持します。これは、D1 および D2 が DStop または DStandby であることを暗黙的に意味しています。D1 または D2 が DStop または DStandby を終了すると、すぐに D3 は DRun モードに切り替わります。
- D1、D2、および D3 が DStandby モードである限り、システムは STANDBY モードを維持します。
- ドメインの状態と CPU の状態：
  - D1 ドメインが DRun モードの場合、バスマトリックスはクロックされ、CPU は CRun モードです。
  - D2 ドメインが DRun モードの場合、バスマトリックスはクロックされ、CPU は CRun モードで、D2 ドメインの少なくとも 1 つのペリフェラルが割り当てられています。
  - D1 ドメインが DStop モードの場合、バスマトリックスはクロックされなくなり、CPU は CStop モードです。
  - D2 ドメインが DStop モードの場合、バスマトリックスはクロックされなくなります。このような状況は次の場合に発生します。
    - CPU が D2 ドメインのペリフェラルを割り当てなかった場合
    - CPU は D2 ドメインのペリフェラルを割り当てたが、CPU が CStop または CStandby である場合
  - ドメインが DStandby モードである場合、CPU を含むドメインの電源がオフであることを意味しています。

表 52. システムの状態の概要

システムの状態	D1 状態	D2 状態	D3 状態
RUN	DRun	DRun/DStop/DStandby	DRun
	DStop/DStandby	DStop/DStandby	
STOP	DStop/DStandby	DStop/DStandby	DStop
STANDBY	DStandby	DStandby	DStandby

8.5.10 ペリフェラルの割り当て

CPU ではペリフェラルを割り当てることができ、これによりカーネルとバスインタフェースのクロックを制御できます。

CPU では、次のレジスタの専用の PERxEN ビットをセットして、ペリフェラルを割り当てることができます。

- RCC_xxxxENR レジスタ、または
- RCC_C1_xxxxENR レジスタ

CPU では、CSleep モードである場合、次のレジスタの PERxLPEN ビットによって、ペリフェラルクロックのゲートを制御することができます。

- RCC_xxxxLPENR レジスタ、または
- RCC_C1_xxxxLPENR レジスタ

詳細については、*セクション 8.7.1：レジスタ配置の概要*を参照してください。

ペリフェラルの割り当てビット（PERxEN ビット）は、ハードウェアがペリフェラルにカーネルおよびバスインタフェースのクロックを供給するために使用します。また、これらはペリフェラルを CPU（CPU サブシステム）に接続する際にも使用されます。これにより、ハードウェアは CPU の状態に従ってペリフェラルクロックとバスマトリックスクロックを安全にゲートできるようになります。PWR ブロックでも、ドメインの状態を正しく制御するために、この情報を使用します。

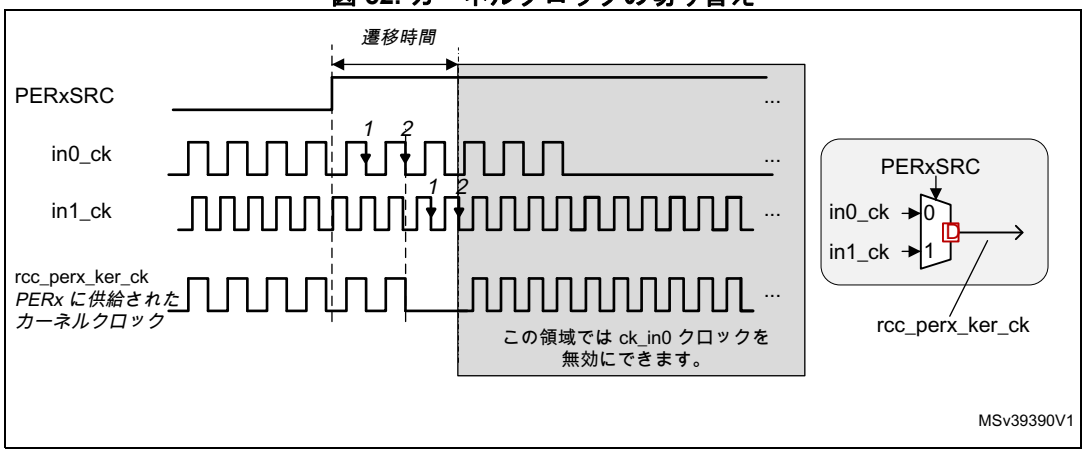
クロックスイッチとクロックゲーティング

- クロックスイッチ遅延

  カーネルクロックスイッチによって選択された入力は、スプールやタイミング違反を生成することなく動的に変更できます。つまり、両方の入力にクロックがある場合にのみ、元の入力から新しい入力に切り替えることができます。そうでない場合、ペリフェラルにクロックは供給されません。この状況を回復するには、両方の入力に有効なクロックを供給する必要があります。

  ペリフェラルに供給されたクロックは、1 つの入力から別の入力に遷移する際にゲートされます。最悪の場合、以前選択したクロックの 2 クロックサイクルと新しく選択したクロックの 2 クロックサイクルの間に、ゲートされます。*図 52* に示すように、両方の入力クロックが遷移時間の間、存在していなければなりません。

図 52. カーネルクロックの切り替え



- クロック有効化遅延

  同様に、クロックゲーティングロジックではスプールの生成を避けるため、有効化コマンド（通常はカーネルクロックリクエストまたはPERxENビットから）を選択したクロックと同期します。

  – 有効化コマンドとクロックの最初の立ち上がりエッジの間で、有効化されたクロックの 2 周期分の最大遅延が発生する可能性があります。有効化コマンドは、RCC_xxxxENR レジスタの PERxEN ビットの立ち上がりエッジ、またはペリフェラルによってアサートされたカーネルクロックリクエストである場合があります。

  – 無効化コマンドとクロックの最後の立ち下がりエッジの間で、無効化されたクロックの 1.5 周期分の最大遅延が発生する可能性があります。無効化コマンドは、RCC_xxxxENR レジスタの PERxEN ビットの立ち下がりエッジ、またはペリフェラルによって解除されたカーネルクロックリクエストである場合があります。

注： ***カーネルクロックとバスインタフェースクロックの両方が、この再同期遅延の影響を受けます。***

さらに、同じドメインに割り当てられていないペリフェラルを初めてアプリケーションで有効にする場合、クロック有効化遅延が大幅に増加する可能性があります。この原因は、ペリフェラルが割り当てられたドメインが DStop または DStandby モードである場合があるためです。アプリケーションがペリフェラルを使用する前に、ドメインを DRun モードに切り替える必要があります。

たとえば、D2 ドメインが DStop/DStandby モードのときに、CPU が D2 ドメインのペリフェラルを有効にした場合、電源コントローラ（PWR）は 、D2 ドメインのクロックを有効にする前に、まず D2 に電源電圧を供給し、次に RCC では PWR からの確認を待つ必要があります。この状況を適切に処理するために、RCC および PWR ブロックには次の 4 つのフラグがあります。

– *RCC ソース制御レジスタ（RCC_CR）* の D1CKRDY/D2CKRDY

– *PWR CPU 制御レジスタ（PWR_CPUCR）* の SBF_D1 および SBF_D2

次のシーケンスにより、この問題を回避することができます。

a) RCC_xxxxENR レジスタの対応する PERxEN ビットに「1」を書き込んで、ペリフェラルクロックを有効にします（ペリフェラルを割り当てます）。

b) 前の書き込み操作が書き込みバッファでペンディングにならないようにするため、RCC_xxxxENR レジスタを読み戻します。

c) ペリフェラルが別のドメインに割り当てられている場合は、次の 2 つのステップを実行します。

   DxCKRDY が「1」にリセットされるまで読み出します。

   ペリフェラルが割り当てられたドメインが DStandby のままであることを確認するために、SBF_Dx にゼロを書き込んで値を読み戻します。対応するビットが「1」として読み出された場合、ドメインはまだ DStandby です。SBF_Dx が「0」になるまでこの操作を繰り返してから、他のステップに進みます。

d) 有効化されたペリフェラルのレジスタへのダミーの読み出し操作を行います。この操作には、最低 2 クロックサイクル必要です。これは、有効化コマンドの最大遅延と同じです。

e) これにより、ペリフェラルを使用できるようになります。

注： ***バスインタフェースクロックがアクティブでない場合、ペリフェラルレジスタへの読み出しまたは書き込みアクセスはサポートされません。読み出しアクセスは無効なデータを返します。書き込みアクセスは無視され、バスエラーは生成されません。***

8.5.11 ペリフェラルクロックゲーティングの制御

前述のように、各ペリフェラルでは **rcc_perx_bus_ck**（ペリフェラル「x」の場合）というバスインタフェースクロックが必要です。このクロックは、ペリフェラルの接続先のバスに応じて APB、AHB、または AXI クロックのいずれかです。

D1 ドメインに割り当てられたペリフェラルのバスインタフェースとして使用されるクロックは、各ペリフェラルに接続されたバスに応じて **rcc_aclk**、**rcc_hclk3**、または **rcc_pclk3** です。単純にするため、これらのクロックを **rcc_bus_d1_ck** と呼びます。

同様に、**rcc_bus_d2_ck** と呼ばれる信号は、D2 ドメインの各ペリフェラルに接続されたバスによって、**rcc_hclk1**、**rcc_hclk2**、**rcc_pclk1**、または **rcc_pclk2** を表します。

また、**rcc_bus_d3_ck** 信号は、D3 に割り当てられたペリフェラルによって **rcc_hclk4** または **rcc_pclk4** を表します。

一部のペリフェラル（SAI、UART など）では、通信インタフェースとして専用のクロックも必要です。このクロックは通常、バスインタフェースクロックに対して非同期です。これはカーネルクロック（**perx_ker_ckreq**）と呼ばれます。両方のクロックを、以降に説明するいくつかの条件に従ってゲートすることができます。

図 53 に示すように、各ペリフェラルのカーネルクロックまたはバスインタフェースクロックの有効化は、いくつかの入力信号に依存します。

- PERxEN および PERxLPEN ビット

  PERxEN は、CPU のペリフェラル有効化（割り当て）ビットを表します。CPU では、RCC_C1_xxxxENR または RCC_xxxxENR レジスタによってこれらのビットに「1」を書き込むことができます。
- PERxAMEN ビット

  PERxAMEN ビットは、*RCC D3 自律モードレジスタ（RCC_D3AMR）* に属します。
- CPU の状態（**c_sleep** および **c_deepsleep** 信号）
- D3 ドメインの状態（**d3_deepsleep** 信号）
- ペリフェラル自体のカーネルクロックリクエスト（**perx_ker_ckreq**）（この機能を使用できる場合）

詳細については、*セクション 8.5.10：ペリフェラルの割り当て* を参照してください。

参考資料

図 53. ペリフェラルカーネルクロック有効ロジックの詳細

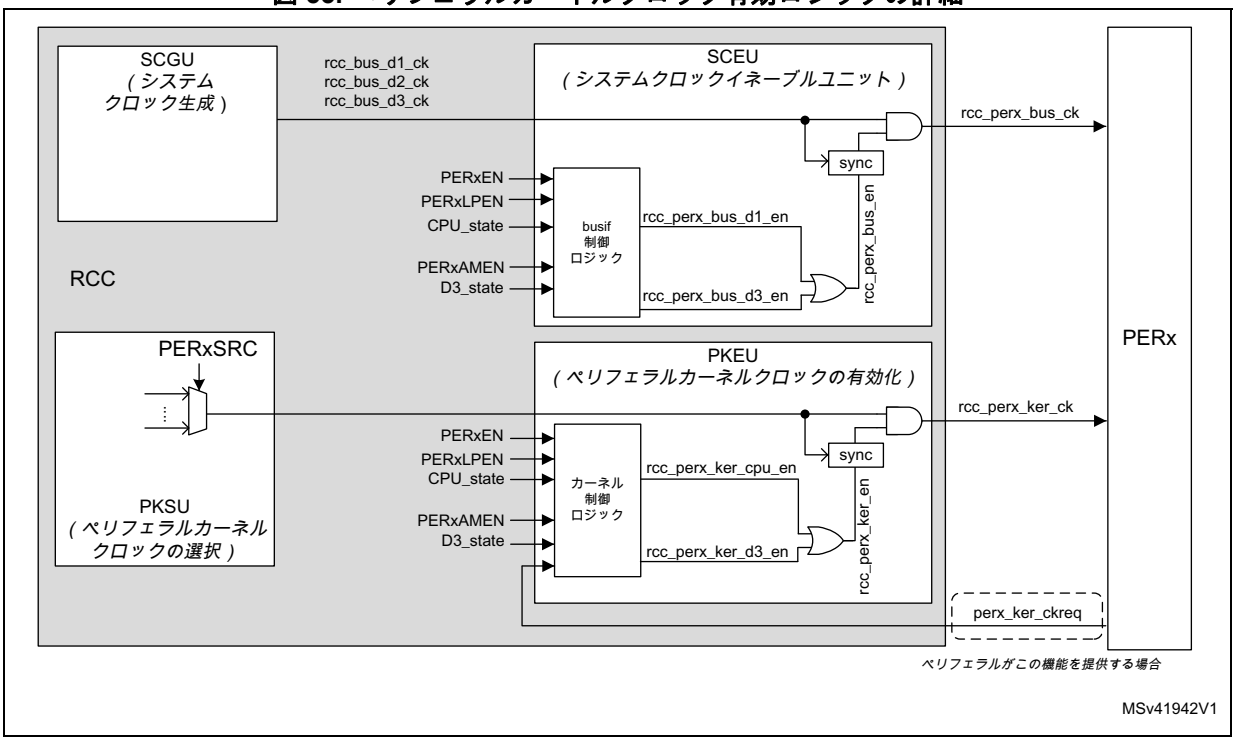


表 53 には、CPU によって D1 または D2 ドメインに割り当てられたペリフェラルのペリフェラルクロックの有効化ロジックについて、詳細な説明が記載されています。

表 53. D1 および D2 ペリフェラルでのペリフェラルクロックの有効化

PERxEN	PERxLPEN	PERxSRC	perx_ker_ckreq	CPU 状態	rcc_perx_ker_c_en	rcc_perx_bus_d1_en	コメント
0	X	X	X	X	0	0	PERxEN =「0」であるため、ペリフェラルへのクロックの供給なし
1	X	X	X	CRun	1	1	CPU が CRun で、PERxEN =「1」であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	CSleep	0	0	CPU が CSleep で、PERxLPEN =「0」であるため、ペリフェラルへのクロックの供給なし
1	1	X	X		1	1	CPU が CSleep で、PERxLPEN =「1」であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	CStop	0	0	PERxLPEN ビットが「0」に設定されているため、ペリフェラルへのクロックの供給なし
1	1	lsi_ck なし および lse_ck なし および hsi_ker_ck なし および csi_ker_ck なし	X		0	0	CPU が CStop で、**lse_ck** または **lsi_ck** または **hsi_ker_ck** または **csi_ker_ck** が選択されていないため、ペリフェラルへのクロックの供給なし
1	1	lsi_ck または lse_ck	X		1 (1)	0	PERxEN = PERxLPEN =「1」で、**lsi_ck** または **lse_ck** が選択されているため、ペリフェラルへのカーネルクロックの供給あり CPU が CStop であるため、バスインタフェースクロックは供給なし
1	1	hsi_ker_ck または csi_ker_ck	1		1	0	**req_ker_perx** =「1」および PERxEN = PERxLPEN =「1」で、**hsi_ker_ck** または **csi_ker_ck** が選択されているため、ペリフェラルへのカーネルクロックの供給あり CPU が CStop であるため、バスインタフェースクロックは供給なし
1	1	hsi_ker_ck または csi_ker_ck	0		0	0	CPU が CStop で、保留中のカーネルクロックリクエストがないため、ペリフェラルへのクロックの供給なし

1. RNG ブロックでは、選択しているクロックが lsi_ck または lse_ck であっても、カーネルクロックが割り当てられる CPU が CStop モードの場合、このカーネルクロックは供給されません。

まとめると、カーネルクロックは、次の条件を満たす場合にドメイン D1 および D2 に割り当てられたペリフェラルに供給される、と言えます。

1. CPU が CRun モードで、ペリフェラルが割り当てられている場合
2. CPU が CSleep モードで、PERxLPEN = 「1」でペリフェラルが割り当てられている場合
3. CPU が CStop モードで、PERxLPEN = 「1」でペリフェラルが割り当てられて、このペリフェラルによってカーネルクロックリクエストが生成され、選択されたクロックが **hsi_ker_ck** または **csi_ker_ck** である場合
4. CPU が CStop モードで、PERxLPEN = 「1」でペリフェラルが割り当てられ、このペリフェラルのカーネルソースクロックが **lse_ck** または **lsi_ck** である場合

バスインタフェースクロックは、条件 1 または 2 を満たしている場合にのみペリフェラルに供給されます。

表 54 には、D3 に割り当てられたすべてのペリフェラルのカーネルクロックの有効化ロジックについて、詳細な説明が記載されています。

表 54. D3 ペリフェラルでのペリフェラルクロックの有効化

PERxEN	PERxLPEN	PERxAMEN	PERxSRC	perx_ker_ckreq	CPU 状態	D3 ステート	rcc_perx_ker_d3_en	rcc_perx_bus_d3_en	コメント
0	X	X	X	X	任意	任意	0	0	PERxEN = 「0」であるため、ペリフェラルへのクロックの供給なし
1	X	X	X	X	CRun	DRun	1	1	CPU が CRun で、PERxEN = 「1」であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	X	CSleep		0	0	CPU が CSleep で、PERxLPEN = 「0」であるため、ペリフェラルへのクロックの供給なし
1	1	X	X	X			1	1	CPU が CSleep で、PERxLPEN = 「1」であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	X	0	X	X	CStop		0	0	CPU は CStop で、PERxEN = 「1」であるため、カーネルクロックゲーティングは D3 ステートおよび PERxAMEN ビットに依存する PERxAMEN = 「0」であるため、ペリフェラルへのクロックの供給なし
1	X	1	X	X			1	1	CPU が CStop モードでも、D3 は DRun モードで、PERxEN および PERxAMEN ビットは「1」にセットされるため、カーネルクロックおよびバスインタフェースクロックの供給あり
1	X	1	lse_ck 以外および lsi_ck 以外	0		DStop	0	0	D3 が DStop で、**req_ker_perx** = 「0」、また **lse_ck** または **lsi_ck** が選択されていないため、ペリフェラルへのクロックの供給なし

表 54. D3 ペリフェラルでのペリフェラルクロックの有効化（続き）

PERxEN	PERxLPEN	PERxAMEN	PERxSRC	perx_ker_ckreq	CPU 状態	D3 ステート	rcc_perx_ker_d3_en	rcc_perx_bus_d3_en	コメント
1	X	1	hsi_ker_ck 以外 および csi_ker_ck 以外 および lse_ck 以外 および lsi_ck 以外	1	CStop	DStop	0	0	**req_ker_perx** =「0」でも、**lse_ck** または **lsi_ck** または **hsi_ker_ck** または **csi_ker_ck** が選択されていないため、ペリフェラルへのクロックの供給なし
1	X	1	hsi_ker_ck または csi_ker_ck	1			1	0	**req_ker_perx** =「1」および PERxEN = PERxAMEN =「1」で、選択されたクロックが **hsi_ker_ck** または **csi_ker_ck** であるため、ペリフェラルへのカーネルクロックの供給あり D3 が DStop であるため、バスインタフェースクロックは供給なし
1	X	1	lse_ck または lsi_ck	X			1	0	D3 が STOP の間、PERxEN = PERxAMEN =「1」で、**lse_ck** または **lsi_ck** が選択されているため、ペリフェラルへのカーネルクロックの供給あり D3 が DStop であるため、バスインタフェースクロックは供給なし

まとめると、カーネルクロックは、次の条件を満たす場合にドメイン D3 のペリフェラルに供給される、と言えます。

1. CPU が CRun モードで、ペリフェラルが割り当てられている場合
2. CPU が CSleep モードで、PERxLPEN = 「1」でペリフェラルが割り当てられている場合
3. CPU が CStop モードで、PERxLPEN = 「1」でペリフェラルが割り当てられて D3 ドメインが DRun モードになる場合
4. CPU が CStop モードで、PERxAMEN = 「1」でペリフェラルが割り当てられて D3 ドメインが DStop モードになり、このペリフェラルによってカーネルクロックリクエストが生成され、カーネルクロックソースが **hsi_ker_ck** または **csi_ker_ck** である場合
5. CPU が CStop モードで、PERxAMEN = 「1」でペリフェラルが割り当てられて D3 ドメインが DStop モードになり、このペリフェラルのカーネルクロックソースが **lse_ck** または **lsi_ck** である場合

バスインタフェースクロックは、条件 1、2、または 3 を満たしている場合にのみペリフェラルに供給されます。

注： ***自律ビットが「1」にセットされると、関連付けられたペリフェラルが CPU のモードではなく D3 ステートに従ってカーネルクロックを受信することを示します。***

カーネルクロックを要求できるのは、I2C、U(S)ART、および LPUART ペリフェラルのみです。この機能により、ペリフェラルには省電力のデータ転送機能が与えられます。

D3 ドメインに割り当てられた一部のペリフェラル専用の自律ビットにより、CPU を有効化せずに外部デバイスとのデータ転送を行うことができます。

回路が STOP モードのときに lse_ck または lsi_ck とともに LPTIMER を動作するには、ユーザアプリケーションで LPTIMxSEL フィールドを介して lsi_ck または lse_ck 入力を選択し、LPTIMxAMEN および LPTIMxLPEN ビットを「1」にセットする必要があります。

8.5.12 CPU およびバスマトリックスクロックゲーティングの制御

CPU クロックとバスマトリックスクロックの有効化／無効化を、ドメインごとに制御できます。

従来の名称については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

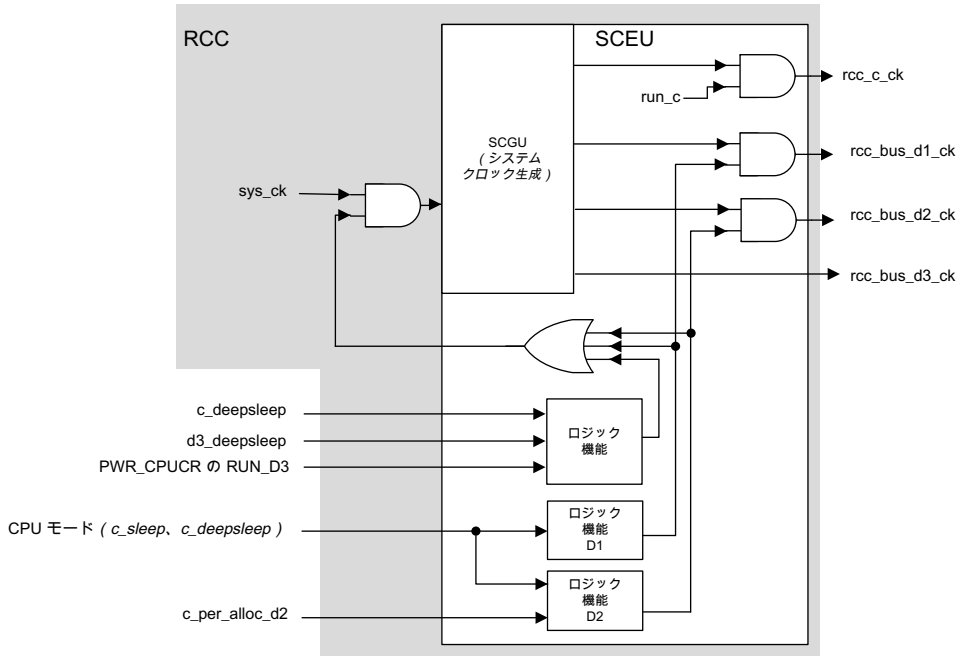
CPU、AHB & AXI ブリッジ、および APB バスのクロックは、ここで説明する規則に従って有効化されます。

- CPU クロックの **rcc_c_ck** は、CPU が CRun モードの場合に有効になります。
- AXI ブリッジクロックは、CPU が CRun モードの場合に有効になります。
- D2 ドメインの AHB ブリッジは、次の場合に有効になります。
  - CPU が CRun の場合、または
  - CPU が CSleep で、このバスに少なくともペリフェラル（マスタ）が 1 つ接続されており、PERxEN および PERxLPEN が両方とも「1」にセットされている場合、または
  - CPU が CSleep で、クロックが有効化された APB バスが少なくとも 1 つある場合
- D3 ドメインの AHB ブリッジクロックは、次の場合に有効になります。
  - CPU が CRun または CSleep モードの場合、または
  - RUN_D3 ビットが CPU のモードにかかわらず「1」にセットされている場合、または
  - **d3_deepsleep** 信号が CPU のモードにかかわらずインアクティブ（「0」）である場合
- APB1、2、3 のバスは、次の場合に有効になります。
  - CPU が CRun の場合、または
  - CPU が CSleep で、このバスに少なくともペリフェラルが 1 つ接続されており、PERxEN および PERxLPEN が両方とも「1」にセットされている場合
- APB4 バスは、D3 ドメインが DRun の場合に有効になります。

図 54 に示すように、各ドメインのコアおよびバスクロックの有効化は、いくつかの入力信号に依存します。

- CPU の **c_sleep** および **c_deepsleep** 信号
- **d3_sleepdeep** 信号
- D2 ドメインに割り当てられたペリフェラルの RCC_xxxxENR.PERxEN ビット

図 54. バスクロック有効ロジック



run_c は、NOT c_deepsleep および NOT c_sleep の組み合わせです。

rcc_bus_dx は、バスマトリックスのクロックとドメイン x のペリフェラルバスインタフェースを表します。

8.6 RCC 割り込み

RCC では、3 つの割り込みラインを提供します。

- **rcc_it** : PLL またはオシレータがレディ状態のときにイベントを提供する、一般的な割り込みライン
- **rcc_hsecss_it** : HSE クロックセキュリティシステムの障害検出専用の割り込みライン
- **rcc_lsecss_it** : LSE クロックセキュリティシステムの障害検出専用の割り込みライン

割り込みイネーブルは、HSE CSS 障害を除き、*RCC クロックソース割り込み有効レジスタ（RCC_CIER）*によって制御されます。HSE CSS 機能が有効なときには、割り込み生成をマスクすることはできません。

割り込みフラグは*RCC クロックソース割り込みフラグレジスタ（RCC_CIFR）*で確認でき、これらのフラグは*RCC クロックソース割り込みクリアレジスタ（RCC_CICR）*でクリアできます。

注： ***割り込みフラグは、対応する割り込みイネーブルビットがセットされていない場合は無関係です。***

割り込みソースの概要と、その制御方法については、*表 55* を参照してください。

表 55. 割り込みソースとその制御

割り込みソース	説明	割り込みイネーブル	割り込みをクリアする操作	割り込みライン
LSIRDYF	LSI レディ	LSIRDYIE	LSIRDYC を「1」にセット	rcc_it
LSERDYF	LSE レディ	LSERDYIE	LSERDYC を「1」にセット	
HSIDRYF	HSI レディ	HSIDRYIE	HSIRDYC を「1」にセット	
HSERDYF	HSE レディ	HSERDYIE	HSERDYC を「1」にセット	
CSIRDYF	CSI レディ	CSIRDYIE	CSIRDYC を「1」にセット	
HSI48RDYF	HSI48 レディ	HSI48RDYIE	HSI48RDYC を「1」にセット	
PLL1RDYF	PLL1 レディ	PLL1RDYIE	PLL1RDYC を「1」にセット	
PLL2RDYF	PLL2 レディ	PLL2RDYIE	PLL2RDYC を「1」にセット	
PLL3RDYF	PLL3 レディ	PLL3RDYIE	PLL3RDYC を「1」にセット	
LSECSSF	LSE クロックセキュリティシステム障害	LSECSSFIE (1)	LSECSSC を「1」にセット	rcc_lsecss_it
HSECSSF	HSE クロックセキュリティシステム障害	-(2)	HSECSSC を「1」にセット	rcc_hsecss_it

1. 割り込みを生成するには、セキュリティシステムの機能も有効にする必要があります（LSECSSON = 「1」）。
2. セキュリティシステムの機能が有効である場合（HSECSSON = 「1」）、この割り込みをマスクすることはできません。

8.7 RCC レジスタの説明

8.7.1 レジスタ配置の概要

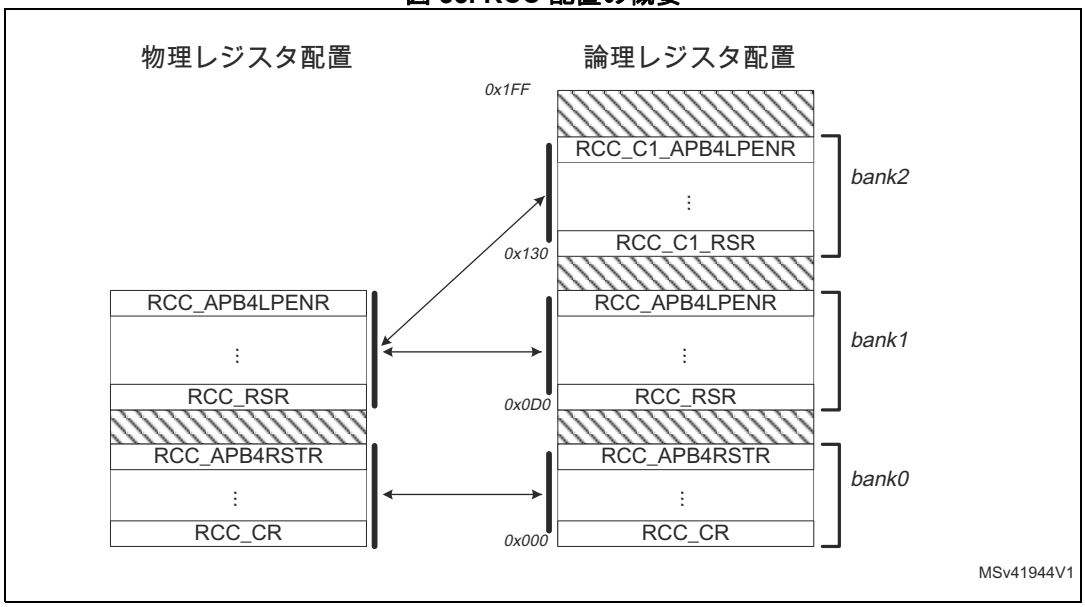
PERxEN および PERxLPEN ビットの制御は、2 つの異なるアドレスオフセット、0x0D0 および 0x130 で実行できます。これにより、アプリケーションでは次のレジスタを使用できます。

- PERxEN ビットを制御する RCC_xxxxENR または RCC_C1_xxxxENR
- PERxLPEN ビットを制御する RCC_xxxxLPENR または RCC_C1_xxxxLPENR
- リセットフラグステータスビットを制御する RCC_RSR または RCC_C1_RSR

この機能は、同じファミリの他の製品との互換性を保証するために提供されています。

図 55 に RCC 配置の概要を示します。

図 55. RCC 配置の概要



8.7.2 RCC ソース制御レジスタ（RCC_CR）

アドレスオフセット：0x000

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	Res.	HSECSSON	HSEBYP	HSERDY	HSEON
		r	rw	r	rw	r	rw					rs	rw	r	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D2CKRDY	D1CKRDY	HSI48RDY	HSI48ON	Res.	Res.	CSIKERON	CSIRDY	CSION	Res.	HSIDIVF	HSIDIV[1:0]		HSIRDY	HSIKERON	HSION
r	r	r	rw			rw	r	rw		r	rw	rw	r	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **PLL3RDY**：PLL3 クロックレディフラグ
ハードウェアによってセットされ、PLL3 がロック状態であることを示します。
0：PLL3 アンロック（リセット後のデフォルト）
1：PLL3 ロック

ビット 28 **PLL3ON**：PLL3 イネーブル
PLL3 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。
0：PLL3 オフ（リセット後のデフォルト）
1：PLL3 オン

ビット 27 **PLL2RDY**：PLL2 クロックレディフラグ
ハードウェアによってセットされ、PLL2 がロック状態であることを示します。
0：PLL2 アンロック（リセット後のデフォルト）
1：PLL2 ロック

ビット 26 **PLL2ON**：PLL2 イネーブル
PLL2 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。
0：PLL2 オフ（リセット後のデフォルト）
1：PLL2 オン

ビット 25 **PLL1RDY**：PLL1 クロックレディフラグ
ハードウェアによってセットされ、PLL1 がロック状態であることを示します。
0：PLL1 アンロック（リセット後のデフォルト）
1：PLL1 ロック

ビット 24 **PLL1ON**：PLL1 イネーブル
PLL1 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。PLL1 出力をシステムクロックとして使用している場合、ハードウェアではこのビットへの「0」の書き込みを阻止します。
0：PLL1 オフ（リセット後のデフォルト）
1：PLL1 オン

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **HSECSSON**：クロックセキュリティシステム有効化

HSE のクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。
このビットはセット専用です（システムリセット、またはシステムが STANDBY モードに移行すると無効になります）。
HSECSSON がセットされているとき、HSE がレディになるとクロック検出回路がハードウェアによって有効にされ、オシレータの障害が検出された場合ハードウェアによって無効にされます。
0：HSE のクロックセキュリティシステムオフ（クロック検出回路オフ）（リセット後のデフォルト）
1：HSE のクロックセキュリティシステムオン（HSE オシレータが安定していればクロック検出回路オン、そうでなければオフ）

ビット 18 **HSEBYP**：HSE クロックバイパス

外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。
デバイスで使用するには、HSEON ビットで外部クロックを有効にする必要があります。
HSEBYP ビットは、HSE オシレータが無効のときのみ、書き込みができます。
0：HSE オシレータはバイパスされません（リセット後のデフォルト）。
1：HSE オシレータはバイパスされ、外部クロックが使用されます。

ビット 17 **HSERDY**：HSE クロックレディフラグ

HSE オシレータが安定していることを示すために、ハードウェアによってセットされます。
0：HSE クロックはレディ状態ではありません（リセット後のデフォルト）
1：HSE クロックはレディ状態です。

ビット 16 **HSEON**：HSE クロック有効化

ソフトウェアでセット／クリアされます。
STOP または STANDBY モードに入るときに、HSE を停止するためにハードウェアによってクリアされます。
HSE がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で HSE が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを「1」にセット）は、このビットをクリアできません。
0：HSE がオフになります（リセット後のデフォルト）。
1：HSE がオンになります。

ビット 15 **D2CKRDY**：D2 ドメインクロックレディフラグ

ハードウェアによってセットされ、D2 ドメインクロックが使用可能であることを示します。
0：D2 ドメインクロックを使用できません（リセット後のデフォルト）。
1：D2 ドメインクロックを使用できます。

ビット 14 **D1CKRDY**：D1 ドメインクロックレディフラグ

ハードウェアによってセットされ、D1 ドメインクロック（CPU、バス、およびペリフェラル）が使用可能であることを示します。
0：D1 ドメインクロックを使用できません（リセット後のデフォルト）。
1：D1 ドメインクロックを使用できます。

ビット 13 **HSI48RDY**：HSI48 クロックレディフラグ

HSI48 オシレータが安定していることを示すために、ハードウェアによってセットされます。
0：HSI48 クロックはレディ状態ではありません（リセット後のデフォルト）
1：HSI48 クロックはレディ状態です。

ビット 12 **HSI48ON**：HSI48 クロック有効化

システムが STOP または STANDBY モードに移行する際に、ソフトウェアによってセットされ、ソフトウェアまたはハードウェアによってクリアされます。
0：HSI48 はオフになります（リセット後のデフォルト）。
1：HSI48 がオンになります。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CSIKERON**：STOP モードでの CSI クロック有効化

一部のペリフェラルのカーネルクロックとしてすばやく使用できるよう、STOP モードの場合でも強制的に CSI をオンにするために、ソフトウェアでセット／リセットされます。このビットは、CSION の値への影響はありません。

0：CSI への影響はありません（リセット後のデフォルト）。

1：CSI は強制的にオンになります（STOP モードの場合も含む）。

ビット 8 **CSIRDY**：CSI クロックレディフラグ

CSI オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは、RC が CSION によって有効化された場合にのみ有効になります（CSI が CSIKERON またはペリフェラルリクエストで有効化された場合は有効になりません）。

0：CSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：CSI クロックはレディ状態です。

ビット 7 **CSION**：CSI クロック有効化

システムおよび／またはペリフェラルの CSI クロックを有効化／無効化するために、ソフトウェアでセット／リセットされます。

システムが STOP モードを終了したときに、強制的に CSI をオンにするためにハードウェアによってセットされます（STOPWUCK = 「1」または STOPKERWUCK = 「1」の場合）。

CSI がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で CSI が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを「1」にセット）は、このビットをクリアできません。

0：CSI がオフになります（リセット後のデフォルト）。

1：CSI がオンになります。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **HSIDIVF**：HSI 分周器フラグ

ハードウェアでセット／リセットされます。

HSIDIV への書き込み操作は、周波数にすぐに影響を及ぼすことはないため、このフラグは HSI 分周器の現在のステータスを示します。HSIDIVF は、HSIDIV 値が変更されるとすぐに「0」になり、出力周波数が HSIDIV にプログラムされた値と一致する場合に「1」に戻されます。

0：**hsi(_ker)_ck** にはまだ伝搬されていない新しい分周比（リセット後のデフォルト）。

1：**hsi(_ker)_ck** クロック周波数によって新しい HSIDIV 値が反映されます。

ビット 4:3 **HSIDIV[1:0]**：HSI クロック分周器

ソフトウェアでセット／クリアされます。

これらのビットにより、必要な HSI クロック周波数を設定するための分周比を選択できます。最低 1 つの有効な PLL（PLLxON ビットを「1」にセット）で HSI をリファレンスクロックとして選択している場合は、HSIDIV を変更することはできません。この場合、新しい HSIDIV 値は無視されます。

00：分周比 1、**hsi(_ker)_ck** = 64 MHz （リセット後のデフォルト）

01：分周比 2、**hsi(_ker)_ck** = 32 MHz

10：分周比 4、**hsi(_ker)_ck** = 16 MHz

11：分周比 8、**hsi(_ker)_ck** = 8 MHz

ビット 2 **HSIRDY**：HSI クロックレディフラグ

HSI オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：HSI クロックはレディ状態です。

ビット 1 **HSIKERON**：STOP モードでのハイスピード内部クロックイネーブル

ペリフェラルのカーネルクロックとしてすばやく使用できるよう、STOP モードの場合でも強制的に HSI をオンにするために、ソフトウェアでセット／リセットされます。このビットは、HSION の値への影響はありません。

0：HSI への影響はありません（リセット後のデフォルト）。

1：HSI は強制的にオンになります（STOP モードの場合も含む）。

ビット 0 **HSION**：ハイスピード内部クロックイネーブル

ソフトウェアでセット／クリアされます。

製品が STOP モードを終了したときに、強制的に HSI をオンにするためにハードウェアによってセットされます（STOPWUCK = 「0」または STOPKERWUCK = 「0」の場合）。

製品が STANDBY モードを終了するとき、またはシステムクロックソースとして使用されている HSE に障害が発生した場合、HSI を強制的にオンにするために、ハードウェアによってセットされます。

HSI がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で HSI が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを「1」にセット）は、このビットをクリアできません。

0：HSI がオフになります。

1：HSI がオンになります（リセット後のデフォルト）。

8.7.3 RCC 内部クロックソース較正レジスタ（RCC_ICSCR）

アドレスオフセット：0x004

リセット値：0x4xxx 0xxx

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CSITRIM[4:0]					CSICAL[7:0]								HSITRIM[5:4]	
	rw					r								rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSITRIM[3:0]				HSICAL[11:0]											
rw				r											

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:26 **CSITRIM[4:0]**：CSI クロックトリミング
較正を調整するために、ソフトウェアでセットされます。
較正のトリミング値を形成するために、リセットフェーズ中にロードされたエンジニアリングオプションバイト（FLASH_CSI_opt）に CSITRIM フィールドが追加されます。
CSICAL = CSITRIM + FLASH_CSI_opt.

注： このフィールドのリセット値は 0x10 です。

ビット 25:18 **CSICAL[7:0]**：CSI クロック較正
ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。
ソフトウェアによって、トリミングビット CSITRIM を使用して調整されます。
このフィールドは、エンジニアリングオプションバイトの較正値と CSITRIM ビット値の合計を表します。

ビット 17:12 **HSITRIM[5:0]**：HSI クロックトリミング
較正を調整するために、ソフトウェアでセットされます。
較正のトリミング値を形成するために、リセットフェーズ中にロードされたエンジニアリングオプションバイト（FLASH_HSI_opt）に HSITRIM フィールドが追加されます。
HSICAL = HSITRIM + FLASH_HSI_opt.

注： このフィールドのリセット値は 0x20 です。

ビット 11:0 **HSICAL[11:0]**：HSI クロック較正
ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。
ソフトウェアによって、トリミングビット HSITRIM を使用して調整されます。
このフィールドは、エンジニアリングオプションバイトの較正値と HSITRIM ビット値の合計を表します。

8.7.4 RCC クロックリカバリ RC レジスタ（RCC_CRRCR）

アドレスオフセット：0x008

リセット値：0x0000 0xxx

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL[9:0]									
						r									

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **HSI48CAL[9:0]**：内部 RC 48 MHz クロック較正

ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。

読み出し専用。

8.7.5 RCC クロック設定レジスタ（RCC_CFGR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MCO2[2:0]			MCO2PRE[3:0]				MCO1[2:0]			MCO11PRE[3:0]				Res.	Res.
rw			rw				rw			rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMPRE	HRTIMSEL	RTCPRE[5:0]						STOPKERWUCK	STOPWUCK	SWS[2:0]			SW[2:0]		
rw	rw	rw						rw	rw	r			rw		

ビット 31:29 **MCO2[2:0]**：マイクロコントローラクロック出力 2

ソフトウェアでセット／クリアされます。クロックソースの選択によって、MCO2 でグリッチが生成することがあります。

これらのビットは、リセット後に、外部オシレータおよび PLL を有効にする前に設定することを強く推奨します。

000：システムクロックの選択（**sys_ck**）（リセット後のデフォルト）

001：PLL2 オシレータクロックの選択（**pll2_p_ck**）

010：HSE クロックの選択（**hse_ck**）

011：PLL1 クロックの選択（**pll1_p_ck**）

100：CSI クロックの選択（**csi_ck**）

101：LSI クロックの選択（**lsi_ck**）

その他：予約済み

ビット 28:25 **MCO2PRE[3:0]**：MCO2 プリスケーラ

MCO2 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。このプリスケーラの変更によって、MCO2 でグリッチが生成することがあります。このプリスケーラは、リセット後に、外部オシレータおよび PLL を有効にする前に変更することを強く推奨します。

0000：プリスケーラは無効（リセット後のデフォルト）

0001：分周比 1 （バイパス）

0010：分周比 2

0011：分周比 3

0100：分周比 4

...

1111：分周比 15

ビット 24:22 **MCO1[2:0]**：マイクロコントローラクロック出力 1

ソフトウェアでセット／クリアされます。クロックソースの選択によって、MCO1 でグリッチが生成することがあります。

これらのビットは、リセット後に外部オシレータおよび PLL を有効にする前に設定することを強く推奨します。

000：HSI クロックの選択（**hsi_ck**）（リセット後のデフォルト）

001：LSE オシレータクロックの選択（**lse_ck**）

010：HSE クロックの選択（**hse_ck**）

011：PLL1 クロックの選択（**pll1_q_ck**）

100：HSI48 クロックの選択（**hsi48_ck**）

その他：予約済み

ビット 21:18 **MCO1PRE[3:0]**：MCO1 プリスケーラ

MCO1 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。このプリスケーラの変更によって、MCO1 でグリッチが生成することがあります。このプリスケーラは、リセット後に外部オシレータおよび PLL を有効にする前に変更することを強く推奨します。

0000：プリスケーラは無効（リセット後のデフォルト）

0001：分周比 1 （バイパス）

0010：分周比 2

0011：分周比 3

0100：分周比 4

...

1111：分周比 15

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TIMPRE**：タイマクロックプリスケーラ選択

このビットは、APB1 ドメインおよび APB2 ドメインに接続されているすべてのタイマのクロック周波数を制御するために、ソフトウェアによってセット／クリアされます。

0：タイマカーネルクロックは D2PPREx が分周比 1 または分周比 2 に対応している場合は **rcc_hclk1** と等しくなり、それ以外の場合は 2 x $F_{rcc_pclkx_d2}$ と等しくなります（リセット後のデフォルト）。

1：タイマカーネルクロックは D2PPREx が分周比 1、2、または 4 に対応している場合は **rcc_hclk1** と等しくなり、それ以外の場合は 4 x $F_{rcc_pclkx_d2}$ と等しくなります。

*表 48：クロックタイマと pclk の比率*を参照してください。

ビット 14 **HRTIMSEL**：高分解能タイマクロックプリスケーラ選択

このビットは、高分解能タイマのクロック周波数を制御するために、ソフトウェアによってセット／クリアされます（HRTIM）。

0：HRTIM プリスケーラクロックソースは他のタイマと同じです（リセット後のデフォルト）。

1：HRTIM プリスケーラクロックソースは CPU クロックです（**rcc_c_ck**）。

ビット 13:8 **RTCPRE[5:0]**：RTC クロックのための HSE 分周比

HSE を分周して RTC にクロックを生成するために、ソフトウェアによってセット／クリアされます。

注：これらのビットは、RTC に供給されるクロックが 1 MHz 未満になるように、ソフトウェアで正しく設定する必要があります。これらのビットは、必要に応じて RTC クロックソースが選択される前に設定する必要があります。

000000：クロックなし（リセット後のデフォルト）

000001：クロックなし

000010：HSE/2

000011：HSE/3

000100：HSE/4

...

111110：HSE/62

111111：HSE/63

ビット 7 **STOPKERWUCK**：システム STOP からのウェイクアップ後のカーネルクロック選択

システム STOP からカーネルウェイクアップクロックを選択するために、ソフトウェアによってセット／リセットされます。

0：HSI がウェイクアップクロックとしてシステム STOP から選択されます（リセット後のデフォルト）。

1：CSI がウェイクアップクロックとしてシステム STOP から選択されます。

詳細は、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照してください。

ビット 6 **STOPWUCK**：システム STOP からのウェイクアップ後のシステムクロック選択

システム STOP からシステムウェイクアップクロックを選択するために、ソフトウェアによってセット／リセットされます。

選択されたクロックは、HSE のクロックセキュリティシステムの非常時クロックにも使用されます。

0：HSI がウェイクアップクロックとしてシステム STOP から選択されます（リセット後のデフォルト）。

1：CSI がウェイクアップクロックとしてシステム STOP から選択されます（リセット後のデフォルト）。

詳細は、*セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理*を参照してください。

注意： ***クロックセキュリティシステムが HSECSSON ビットによって有効化され、システムクロックが HSE（SWS =「10」）であるか HSE のスイッチ（SW =「10」）を要求している場合、STOPWUCK を変更しないでください。***

ビット 5:3 **SWS[2:0]**：システムクロックスイッチステータス

どのクロックソースがシステムクロックとして使用されているかを示すために、ハードウェアによってセット／リセットされます。

000：HSI がシステムクロックとして使用されます（**hsi_ck**）（リセット後のデフォルト）。

001：CSI がシステムクロックとして使用されます（**csi_ck**）。

010：HSE がシステムクロックとして使用されます（**hse_ck**）。

011：PLL1 がシステムクロックとして使用されます（**pll1_p_ck**）。

上記以外：予約済み

ビット 2:0 **SW[2:0]**：システムクロックスイッチ

システムクロックソース（**sys_ck**）を選択するために、ソフトウェアによってセット／リセットされます。

以下を実行するため、ハードウェアによってセットされます。

- システム STOP モード終了時に、HSI または CSI の選択（STOPWUCK 選択による）を強制
- 直接的または間接的にシステムクロックとして使用された HSE に障害が発生した場合に、HSI の選択を強制

000：HSI がシステムクロックとして選択されます（**hsi_ck**）（リセット後のデフォルト）。

001：CSI がシステムクロックとして選択されます（**csi_ck**）。

010：HSE がシステムクロックとして選択されます（**hse_ck**）。

011：PLL1 がシステムクロックとして選択されます（**pll1_p_ck**）。

上記以外：予約済み

8.7.6 RCC ドメイン 1 クロック設定レジスタ（RCC_D1CFGR）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	D1CPRE[3:0]				Res.	D1PPRE[2:0]			HPRE[3:0]			
				rw					rw			rw			

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **D1CPRE[3:0]**：D1 ドメインのコアプリスケーラ

D1 ドメインの CPU クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。分周比を変更すると、CPU クロックと、すべてのバスマトリックスクロックの周波数に影響します。クロックは、新しいプリスケーラ分周比で分周されます。この分周比の範囲は、D1CPRE　更新後の **rcc_pclk[4:1]** の最も低速な APB クロックの 1 から 16 周期です。このアプリケーションでは、このレジスタを読み戻して新しい分周比が考慮されているかどうかを確認できます。

0xxx：**sys_ck** は分周されません（リセット後のデフォルト）。
1000：**sys_ck** は 2 分周されます。
1001：**sys_ck** は 4 分周されます。
1010：**sys_ck** は 8 分周されます。
1011：**sys_ck** は 16 分周されます。
1100：**sys_ck** は 64 分周されます。
1101：**sys_ck** は 128 分周されます。
1110：**sys_ck** は 256 分周されます。
1111：**sys_ck** は 512 分周されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **D1PPRE[2:0]**：D1 ドメインの APB3 プリスケーラ

rcc_pclk3 の分周比を制御するために、ソフトウェアによってセット／リセットされます。
クロックは、D1PPRE に書き込んでから、**rcc_hclk3** の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk3** = **rcc_hclk3** （リセット後のデフォルト）
100：**rcc_pclk3** = **rcc_hclk3** / 2
101：**rcc_pclk3** = **rcc_hclk3** / 4
110：**rcc_pclk3** = **rcc_hclk3** / 8
111：**rcc_pclk3** = **rcc_hclk3** / 16

ビット 3:0 **HPRE[3:0]**：D1 ドメインの AHB プリスケーラ

rcc_hclk3 および **rcc_aclk** の分周比を制御するために、ソフトウェアによってセット／クリアされます。分周比を変更すると、すべてのバスマトリックスクロックの周波数に影響します。

0xxx：**rcc_hclk3** = **sys_d1cpre_ck** （リセット後のデフォルト）
1000：**rcc_hclk3** = **sys_d1cpre_ck** / 2
1001：**rcc_hclk3** = **sys_d1cpre_ck** / 4
1010：**rcc_hclk3** = **sys_d1cpre_ck** / 8
1011：**rcc_hclk3** = **sys_d1cpre_ck** / 16
1100：**rcc_hclk3** = **sys_d1cpre_ck** / 64
1101：**rcc_hclk3** = **sys_d1cpre_ck** / 128
1110：**rcc_hclk3** = **sys_d1cpre_ck** / 256
1111：**rcc_hclk3** = **sys_d1cpre_ck** / 512

注： ***クロックは、HPRE 更新後、rcc_pclk[4:1] の最も低速な APB クロックの 1 から 16 周期の新しいプリスケーラ分周比で分周されます。***

注： ***rcc_hclk3 = rcc_aclk である点にも注意してください。***

注意： ***電圧スケーリングを使用するときには注意が必要です。新しい分周比の伝搬遅延により、プリスケーラ分周比の変更後、V_{CORE} 電圧を下げる前に、このレジスタを読み出して新しいプリスケーラ値が正しく反映されていることを確認する必要があります。***

クロックソース周波数と電圧範囲に応じて、ソフトウェアアプリケーションでは HPRE に正しい値をプログラムし、システム周波数が最大周波数を超えないようにします。

8.7.7 RCC ドメイン 2 クロック設定レジスタ（RCC_D2CFGR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	D2PPRE2[2:0]			Res.	D2PPRE1[2:0]			Res.	Res.	Res.	Res.
					rw				rw						

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **D2PPRE2[2:0]**：D2 ドメインの APB2 プリスケーラ

D2 ドメインの APB2 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。クロックは、D2PPRE2 に書き込んでから、**rcc_hclk1** の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk2** = **rcc_hclk1** （リセット後のデフォルト）
100：**rcc_pclk2** = **rcc_hclk1** / 2
101：**rcc_pclk2** = **rcc_hclk1** / 4
110：**rcc_pclk2** = **rcc_hclk1** / 8
111：**rcc_pclk2** = **rcc_hclk1** / 16

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **D2PPRE1[2:0]**：D2 ドメインの APB1 プリスケーラ

D2 ドメインの APB1 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。クロックは、D2PPRE1 に書き込んでから、rcc_hclk1 の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk1** = **rcc_hclk1** （リセット後のデフォルト）
100：**rcc_pclk1** = **rcc_hclk1** / 2
101：**rcc_pclk1** = **rcc_hclk1** / 4
110：**rcc_pclk1** = **rcc_hclk1** / 8
111：**rcc_pclk1** = **rcc_hclk1** / 16

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

8.7.8 RCC ドメイン 3 クロック設定レジスタ（RCC_D3CFGR）

アドレスオフセット：0x020

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D3PPRE[2:0]			Res.			
									rw						

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **D3PPRE[2:0]**：D3 ドメインの APB4 プリスケーラ

D3 ドメインの APB4 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。クロックは、D3PPRE に書き込んでから、rcc_hclk4 の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk4** = **rcc_hclk4**（リセット後のデフォルト）

100：**rcc_pclk4** = **rcc_hclk4** / 2

101：**rcc_pclk4** = **rcc_hclk4** / 4

110：**rcc_pclk4** = **rcc_hclk4** / 8

111：**rcc_pclk4** = **rcc_hclk4** / 16

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

8.7.9 RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR）

アドレスオフセット：0x028

リセット値：0x0202 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	DIVM3[5:0]						Res.	Res.	DIVM2[5:4]	
						rw								rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVM2[3:0]				Res.	Res.	DIVM1[5:0]						Res.	Res.	PLLSRC[1:0]	
rw						rw								rw	

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:20 **DIVM3[5:0]**：PLL3 のプリスケーラ

PLL3 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL3 が有効である場合（PLL3ON =「1」）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL3 を使用しない場合は、DIVM3 の値を「0」にする必要があります。
000000：プリスケーラは無効（リセット後のデフォルト）
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
...
100000：分周比 32 （リセット後のデフォルト）
...
111111：分周比 63

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:12 **DIVM2[5:0]**：PLL2 のプリスケーラ

PLL2 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL2 が有効である場合（PLL2ON =「1」）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL2 を使用しない場合は、DIVM2 の値を「0」にする必要があります。
000000：プリスケーラは無効
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
...
100000：分周比 32 （リセット後のデフォルト）
...
111111：分周比 63

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:4 **DIVM1[5:0]**：PLL1 のプリスケーラ

PLL1 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL1 が有効である場合（PLL1ON =「1」）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL1 を使用しない場合は、DIVM1 の値を「0」にする必要があります。
000000：プリスケーラは無効
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
...
100000：分周比 32 （リセット後のデフォルト）
...
111111：分周比 63

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PLLSRC[1:0]**：DIVMx および PLL のクロックソース選択

PLL クロックソースを選択するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL がすべて無効なときだけです。
電力を削減するために、PLL を使用しない場合は、PLLSRC の値を「11」にセットする必要があります。
00：HSI が PLL クロックとして選択されます（**hsi_ck**）（リセット後のデフォルト）。
01：CSI が PLL クロックとして選択されます（**csi_ck**）。
10：HSE が PLL クロックとして選択されます（**hse_ck**）。
11：DIVMx 分周器および PLL にクロックが送信されません。

8.7.10 RCC PLL 設定レジスタ（RCC_PLLCFGR）

アドレスオフセット：0x02C

リセット値：0x01FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIVR3EN	DIVQ3EN	DIVP3EN	DIVR2EN	DIVQ2EN	DIVP2EN	DIVR1EN	DIVQ1EN	DIVP1EN
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PLL3RGE[1:0]		PLL3VCOSEL	PLL3FRACEN	PLL2RGE[1:0]		PLL2VCOSEL	PLL2FRACEN	PLL1RGE[1:0]		PLL1VCOSEL	PLL1FRACEN
				rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DIVR3EN**：PLL3 の DIVR 分周器出力有効化

PLL3 の **pll3_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。
0：**pll3_r_ck** 出力は無効です。
1：**pll3_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 23 **DIVQ3EN**：PLL3 の DIVQ 分周器出力有効化

PLL3 の **pll3_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。
0：**pll3_q_ck** 出力は無効です。
1：**pll3_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 22 **DIVP3EN**：PLL3 の DIVP 分周器出力有効化

PLL3 の **pll3_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
0：**pll3_p_ck** 出力は無効です。
1：**pll3_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 21 **DIVR2EN**：PLL2 の DIVR 分周器出力有効化

PLL2 の **pll2_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。
0：**pll2_r_ck** 出力は無効です。
1：**pll2_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 20 **DIVQ2EN**：PLL2 の DIVQ 分周器出力有効化

PLL2 の **pll2_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。
0：**pll2_q_ck** 出力は無効です。
1：**pll2_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 19 **DIVP2EN**：PLL2 の DIVP 分周器出力有効化

PLL2 の **pll2_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
0：**pll2_p_ck** 出力は無効です。
1：**pll2_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 18 **DIVR1EN**：PLL1 の DIVR 分周器出力有効化

PLL1 の **pll1_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 ビットを「0」に設定する必要があります。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
0：**pll1_r_ck** 出力は無効です。
1：**pll1_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 17 **DIVQ1EN**：PLL1 の DIVQ 分周器出力有効化

PLL1 の **pll1_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、PLL1 の **pll1_q_ck** 出力を使用しない場合は、**pll1_q_ck** の値を無効にする必要があります。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
0：**pll1_q_ck** 出力は無効です。
1：**pll1_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 16 **DIVP1EN**：PLL1 の DIVP 分周器出力有効化

PLL1 の **pll1_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
電力を削減するために、PLL1 の **pll1_p_ck** 出力を使用しない場合は、**pll1_p_ck** の値を無効にする必要があります。
0：**pll1_p_ck** 出力は無効です。
1：**pll1_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **PLL3RGE[1:0]**：PLL3 入力周波数範囲

PLL3 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
これらのビットは、PLL3 を有効にする前に書き込む必要があります。
00：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、1 から 2 MHz です（リセット後のデフォルト）。
01：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、2 から 4 MHz です。
10：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、4 から 8 MHz です。
11：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、8 から 16 MHz です。

ビット 9 **PLL3VCOSEL**：PLL3 VCO 選択

PLL3 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
このビットは、PLL3 を有効にする前に書き込む必要があります。
0：広帯域 VCOレンジ：192 から 836 MHz （リセット後のデフォルト）
1：中帯域 VCOレンジ：150 から 420 MHz

ビット 8 **PLL3FRACEN**：PLL3 分数ラッチイネーブル

FRACN3 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます

シグマデルタモジュレータに FRACN3 の値をラッチするには、PLL3FRACEN を「0」に設定し、次に「1」にセットする必要があります。0 から 1 への遷移により、モジュレータに FRACN3 の内容が転送されます。詳細については、*セクション ：PLL 初期化フェーズ*を参照してください。

ビット 7:6 **PLL2RGE[1:0]**：PLL2 入力周波数範囲

PLL2 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

これらのビットは、PLL2 を有効にする前に書き込む必要があります。

00：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、1 から 2 MHz です（リセット後のデフォルト）。

01：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、2 から 4 MHz です。

10：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、4 から 8 MHz です。

11：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、8 から 16 MHz です。

ビット 5 **PLL2VCOSEL**：PLL2 VCO 選択

PLL2 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

このビットは、PLL2 を有効にする前に書き込む必要があります。

0：広帯域 VCO：192 から 836 MHz （リセット後のデフォルト）

1：中帯域 VCO：150 から 420 MHz

ビット 4 **PLL2FRACEN**：PLL2 分数ラッチイネーブル

FRACN2 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます

シグマデルタモジュレータに FRACN2 の値をラッチするには、PLL2FRACEN を「0」に設定し、次に「1」にセットする必要があります。0 から 1 への遷移により、モジュレータに FRACN2 の内容が転送されます。詳細については、*セクション ：PLL 初期化フェーズ*を参照してください。

ビット 3:2 **PLL1RGE[1:0]**：PLL1 入力周波数範囲

PLL1 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

このビットは、PLL1 を有効にする前に書き込む必要があります。

00：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、1 から 2 MHz です（リセット後のデフォルト）。

01：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、2 から 4 MHz です。

10：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、4 から 8 MHz です。

11：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、8 から 16 MHz です。

ビット 1 **PLL1VCOSEL**：PLL1 VCO 選択

PLL1 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

これらのビットは、PLL1 を有効にする前に書き込む必要があります。

0：広帯域 VCOレンジ：192 から 836 MHz （リセット後のデフォルト）

1：中帯域 VCOレンジ：150 から 420 MHz

ビット 0 **PLL1FRACEN**：PLL1 分数ラッチイネーブル

FRACN1 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます

シグマデルタモジュレータに FRACN1 の値をラッチするには、PLL1FRACEN を「0」に設定し、次に「1」にセットする必要があります。0 から 1 への遷移により、モジュレータに FRACN1 の内容が転送されます。詳細については、*セクション ：PLL 初期化フェーズ*を参照してください。

8.7.11 RCC PLL1 分周器設定レジスタ（RCC_PLL1DIVR）

アドレスオフセット：0x030

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR1[6:0]							Res.	DIVQ1[6:0]						
	rw								rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP1[6:0]								DIVN1[8:0]							
rw								rw							

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR1[6:0]**：PLL1 DIVR の分周比

pll1_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
0000000：**pll1_r_ck** = **vco1_ck**
0000001：**pll1_r_ck** = **vco1_ck** / 2 （リセット後のデフォルト）
0000010：**pll1_r_ck** = **vco1_ck** / 3
0000011：**pll1_r_ck** = **vco1_ck** / 4
...
1111111：**pll1_r_ck** = **vco1_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ1[6:0]**：PLL1 DIVQ の分周比

pll1_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
0000000：**pll1_q_ck** = **vco1_ck**
0000001：**pll1_q_ck** = **vco1_ck** / 2 （リセット後のデフォルト）
0000010：**pll1_q_ck** = **vco1_ck** / 3
0000011：**pll1_q_ck** = **vco1_ck** / 4
...
1111111：**pll1_q_ck** = **vco1_ck** / 128

ビット 15:9 **DIVP1[6:0]**：PLL1 DIVP の分周比

pll1_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。

奇数の分周比は使用できません。

0000000：設定禁止
0000001：**pll1_p_ck** = **vco1_ck** / 2 （リセット後のデフォルト）
0000010：設定禁止
0000011：**pll1_p_ck** = **vco1_ck** / 4
...
1111111：**pll1_p_ck** = **vco1_ck** / 128

ビット 8:0 **DIVN1[8:0]**：PLL1 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL1ON = 「0」および PLL1RDY = 「0」）。
0x003：DIVN1 = 4
0x004：DIVN1 = 5
0x005：DIVN1 = 6
...
0x080：DIVN1 = 129 （リセット後のデフォルト）
...
0x1FF：DIVN1 = 512
その他：誤った設定

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL1VCOSEL = 「0」の場合、192 から 836 MHz
- PLL1VCOSEL = 「1」の場合、150 から 420 MHz
-

分数値 0 が次の条件で FRACN1 にロードされた場合、VCO 出力周波数 = F_{ref1_ck} x DIVN1 です。

- 4 から 512 の間の DIVN1
- 1 から 16 MHz の間の入力周波数 F_{ref1_ck}

8.7.12 RCC PLL1 分数分周器レジスタ（RCC_PLL1FRACR）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN1[12:0]													Res.	Res.	Res.
rw															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN1[12:0]**：PLL1 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL1 VCO を動的に微調整することができます。

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL1VCOSEL =「0」の場合、192 から 836 MHz
- PLL1VCOSEL =「1」の場合、150 から 420 MHz

VCO 出力周波数 = F_{ref1_ck} x $\left(DIVN1 + (FRACN1 / 2^{13})\right)$ で、次の要件を満たすものとします。

- DIVN1 が 4 から 512 の間であること
- FRACN1 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref1_ck} が 1 から 16 MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従います。

- PLL1FRACEN を「0」に設定します。
- FRACN1 に新しい分数値を書き込みます。
- PLL1FRACEN ビットを「1」にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.13 RCC PLL2 分周器設定レジスタ（RCC_PLL2DIVR）

アドレスオフセット：0x038

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR2[6:0]							Res.	DIVQ2[6:0]						
	rw								rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP2[6:0]								DIVN2[8:0]							
rw								rw							

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR2[6:0]**：PLL2 DIVR の分周比

pll2_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。
0000000：**pll2_r_ck** = **vco2_ck**
0000001：**pll2_r_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_r_ck** = **vco2_ck** / 3
0000011：**pll2_r_ck** = **vco2_ck** / 4
...
1111111：**pll2_r_ck** = **vco2_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ2[6:0]**：PLL2 DIVQ の分周比

pll2_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。
0000000：**pll2_q_ck** = **vco2_ck**
0000001：**pll2_q_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_q_ck** = **vco2_ck** / 3
0000011：**pll2_q_ck** = **vco2_ck** / 4
...
1111111：**pll2_q_ck** = **vco2_ck** / 128

ビット 15:9 **DIVP2[6:0]**：PLL2 DIVP の分周比

pll2_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。

0000000：**pll2_p_ck** = **vco2_ck**
0000001：**pll2_p_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_p_ck** = **vco2_ck** / 3
0000011：**pll2_p_ck** = **vco2_ck** / 4
...
1111111：**pll2_p_ck** = **vco2_ck** / 128

ビット 8:0 **DIVN2[8:0]**：PLL2 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL2ON = 「0」および PLL2RDY = 「0」）。

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL2VCOSEL = 「0」の場合、192 から 836 MHz
- PLL2VCOSEL = 「1」の場合、150 から 420 MHz

分数値 0 が次の条件で FRACN2 にロードされた場合、VCO 出力周波数 = F_{ref2_ck} x DIVN2 です。

- 4 から 512 の間の DIVN2
- 1 から 16MHz の間の入力周波数 F_{ref2_ck}

0x003：DIVN2 = 4
0x004：DIVN2 = 5
0x005：DIVN2 = 6
...
0x080：DIVN2 = 129 （リセット後のデフォルト）
...
0x1FF：DIVN2 = 512
その他：誤った設定

8.7.14 RCC PLL2 分数分周器レジスタ（RCC_PLL2FRACR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN2[12:0]													Res.	Res.	Res.
rw															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN2[12:0]**：PLL2 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL2 VCO を動的に微調整することができます。

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL2VCOSEL =「0」の場合、192 から 836 MHz
- PLL2VCOSEL =「1」の場合、150 から 420 MHz

VCO 出力周波数 = F_{ref2_ck} x $\left(DIVN2 + (FRACN2 / 2^{13})\right)$ で、次の要件を満たすものとします。

- DIVN2 が 4 から 512 の間であること
- FRACN2 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref2_ck} が 1 から 16 MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従います。

- PLL2FRACEN を「0」に設定します。
- FRACN2 に新しい分数値を書き込みます。
- PLL2FRACEN ビットを「1」にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.15 RCC PLL3 分周器設定レジスタ（RCC_PLL3DIVR）

アドレスオフセット：0x040

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR3[6:0]							Res.	DIVQ3[6:0]						
	rw								rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP3[6:0]								DIVN3[8:0]							
rw								rw							

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR3[6:0]**：PLL3 DIVR の分周比

pll3_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。
0000000：**pll3_r_ck** = **vco3_ck**
0000001：**pll3_r_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_r_ck** = **vco3_ck** / 3
0000011：**pll3_r_ck** = **vco3_ck** / 4
...
1111111：**pll3_r_ck** = **vco3_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ3[6:0]**：PLL3 DIVQ の分周比

pll3_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。
0000000：**pll3_q_ck** = **vco3_ck**
0000001：**pll3_q_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_q_ck** = **vco3_ck** / 3
0000011：**pll3_q_ck** = **vco3_ck** / 4
...
1111111：**pll3_q_ck** = **vco3_ck** / 128

ビット 15:9 **DIVP3[6:0]**：PLL3 DIVP の分周比

pll3_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。

0000000：**pll3_p_ck** = **vco3_ck**
0000001：**pll3_p_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_p_ck** = **vco3_ck** / 3
0000011：**pll3_p_ck** = **vco3_ck** / 4
...
1111111：pll3_p_ck = **vco3_ck** / 128

ビット 8:0 **DIVN3[7:0]**：PLL3 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL3ON = 「0」および PLL3RDY = 「0」）。

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL3VCOSEL = 「0」の場合、192 から 836 MHz
- PLL3VCOSEL = 「1」の場合、150 から 420 MHz

分数値 0 が次の要件で FRACN3 にロードされた場合、VCO 出力周波数 = F_{ref3_ck} x DIVN3 です。

- 4 から 512 の間の DIVN3
- 1 から 16MHz の間の入力周波数 F_{ref3_ck}

0x003：DIVN3 = 4
0x004：DIVN3 = 5
0x005：DIVN3 = 6
...
0x080：DIVN3 = 129 （リセット後のデフォルト）
...
0x1FF：DIVN3 = 512
その他：誤った設定

8.7.16 RCC PLL3 分数分周器レジスタ（RCC_PLL3FRACR）

アドレスオフセット：0x044

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN3[12:0]													Res.	Res.	Res.
rw															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN3[12:0]**：PLL3 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL3 VCO を動的に微調整することができます。

注意： ***これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。***

- PLL3VCOSEL =「0」の場合、192 から 836 MHz
- PLL3VCOSEL =「1」の場合、150 から 420 MHz

VCO 出力周波数 = F_{ref3_ck} x $\left(DIVN3 + (FRACN3 / 2^{13})\right)$ で、次の要件を満たすものとします。

- DIVN3 が 4 から 512 の間であること
- FRACN3 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref3_ck} が 1 から 16 MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従います。

- PLL1FRACEN を「0」に設定します。
- FRACN1 に新しい分数値を書き込みます。
- PLL1FRACEN ビットを「1」にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.17 RCC ドメイン 1 カーネルクロック設定レジスタ（RCC_D1CCIPR）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	CKPERSEL[1:0] (1)		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMCSEL (1)
		rw													rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPISEL[1:0] (1)		Res.	Res.	FMCSEL[1:0] (1)	
										rw				rw	

1. タイミング違反を生成することなく、動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。*セクション：クロックスイッチとクロックゲーティング*を参照してください。

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **CKPERSEL[1:0]**：**per_ck** クロックソース選択

00：**hsi_ker_ck** クロックが **per_ck** クロックとして選択されます（リセット後のデフォルト）。
01：**csi_ker_ck** クロックが **per_ck** クロックとして選択されます。
10：**hse_ck** クロックが **per_ck** クロックとして選択されます。
11：予約済み、**per_ck** クロックは無効です。

ビット 27:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMCSEL**：SDMMC カーネルクロックソースの選択

0：**pll1_q_ck** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
1：**pll2_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **QSPISEL[1:0]**：QUADSPI カーネルクロックソースの選択

00：**rcc_hclk3** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。
10：**pll2_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。
11：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **FMCSEL[1:0]**：FMC カーネルクロックソースの選択

00：**rcc_hclk3** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。
10：**pll2_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。
11：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

8.7.18 RCC ドメイン 2 カーネルクロック設定レジスタ（RCC_D2CCIP1R）

アドレスオフセット：0x050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWPSEL (1)	Res.	FDCANSEL[1:0] (1)		Res.	Res.	Res.	DFSDM1SEL (1)	Res.	Res.	SPDIFSEL[1:0] (1)		Res.	SPI45SEL[2:0] (1)		
rw		rw					rw			rw			rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI123SEL[2:0] (1)			Res.	Res.	Res.	SAI23SEL[2:0] (1)			Res.	Res.	Res.	SAI1SEL[2:0] (1)		
	rw						rw						rw		

1. タイミング違反を生成することなく、動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。*セクション：クロックスイッチとクロックゲーティング*を参照してください。

ビット 31 **SWPSEL**：SWPMI カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
0：**pclk** が SWPMI カーネルクロックとして選択されます（リセット後のデフォルト）。
1：**hsi_ker_ck** クロックが SWPMI カーネルクロックとして選択されます。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **FDCANSEL**：FDCAN カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**hse_ck** クロックが FDCAN カーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** クロックが FDCAN カーネルクロックとして選択されます。
10：**pll2_q_ck** クロックが FDCAN カーネルクロックとして選択されます。
11：予約済み、カーネルクロックは無効です。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DFSDM1SEL**：DFSDM1 カーネル **Clk** クロックソースの選択
ソフトウェアでセット／クリアされます。
注： DFSDM1 Aclk クロックソース選択は、SAI1SEL で実行されます。
0：**rcc_pclk2** が DFSDM1 Clk カーネルクロックとして選択されます（リセット後のデフォルト）。
1：**sys_ck** クロックが DFSDM1 Clk カーネルクロックとして選択されます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **SPDIFSEL[1:0]**：SPDIFRX カーネルクロックソースの選択
00：**pll1_q_ck** クロックが SPDIFRX カーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll2_r_ck** クロックが SPDIFRX カーネルクロックとして選択されます。
10：**pll3_r_ck** クロックが SPDIFRX カーネルクロックとして選択されます。
11：**hsi_ker_ck** クロックが SPDIFRX カーネルクロックとして選択されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **SPI45SEL[2:0]**：SPI4 および 5 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：APB クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** クロックがカーネルクロックとして選択されます。
010：**pll3_q_ck** クロックがカーネルクロックとして選択されます。
011：**hsi_ker_ck** クロックがカーネルクロックとして選択されます。
100：**csi_ker_ck** クロックがカーネルクロックとして選択されます。
101：**hse_ck** クロックがカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **SPI123SEL[2:0]**：SPI/I2S1、2、および 3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。

注意： ***選択されたクロックが外部クロックで、これが停止された場合、別のクロックへの切り替えはできません。詳細については、セクション：クロックスイッチとクロックゲーティングを参照してください。***

000：**pll1_q_ck** クロックが SPI/I2S1、2、および 3 カーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** クロックが SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
010：**pll3_p_ck** クロックが SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
011：I2S_CKIN クロックが SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
100：**per_ck** クロックが SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

注： ***I2S_CKIN は、ピンから取得した外部クロックです。***

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:6 **SAI23SEL[2:0]**：SAI2 および SAI3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。

注意： ***選択されたクロックが外部クロックで、これが停止された場合、別のクロックへの切り替えはできません。詳細については、セクション：クロックスイッチとクロックゲーティングを参照してください。***

000：**pll1_q_ck** クロックが SAI2 および SAI3 カーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** クロックが SAI2 および SAI3 カーネルクロックとして選択されます。
010：**pll3_p_ck** クロックが SAI2 および SAI3 カーネルクロックとして選択されます。
011：I2S_CKIN クロックが SAI2 および SAI3 カーネルクロックとして選択されます。
100：**per_ck** クロックが SAI2 および SAI3 カーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

注： ***I2S_CKIN は、ピンから取得した外部クロックです。***

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SAI1SEL[2:0]**：SAI1 および DFSDM1 カーネル **Aclk** クロックソースの選択
ソフトウェアでセット／クリアされます。

注意： ***選択されたクロックが外部クロックで、これが停止された場合、別のクロックへの切り替えはできません。詳細については、セクション：クロックスイッチとクロックゲーティングを参照してください。***

注： ***DFSDM1 クロックソース選択は、DFSDM1SEL で実行されます。***

000：**pll1_q_ck** クロックが SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** クロックが SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
010：**pll3_p_ck** クロックが SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
011：I2S_CKIN クロックが SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
100：**per_ck** クロックが SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

注： ***I2S_CKIN は、ピンから取得した外部クロックです。***

8.7.19 RCC ドメイン 2 カーネルクロック設定レジスタ（RCC_D2CCIP2R）

アドレスオフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPTIM1SEL[2:0] (1)			Res.	Res.	Res.	Res.	CECSEL[1:0] (1)		USBSEL[1:0] (1)		Res.	Res.	Res.	Res.
	rw							rw		rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	I2C123SEL[1:0] (1)		Res.	Res.	RNGSEL[1:0] (1)		Res.	Res.	USART16SEL[2:0] (1)			USART234578SEL[2:0] (1)		
		rw				rw				rw			rw		

1. タイミング違反を生成することなく、動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。*セクション：クロックスイッチとクロックゲーティング*を参照してください。

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **LPTIM1SEL[2:0]**：LPTIM1 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk1** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。
010：**pll3_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。
011：**lse_ck** クロックがカーネルペリフェラルクロックとして選択されます。
100：**lsi_ck** クロックがカーネルペリフェラルクロックとして選択されます。
101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **CECSEL[1:0]**：HDMI-CEC カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**lse_ck** クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**lsi_ck** クロックがカーネルクロックとして選択されます。
10：122 分周された **csi_ker_ck** がカーネルクロックとして選択されます。
11：予約済み、カーネルクロックは無効です。

ビット 21:20 **USBSEL[1:0]**：USBOTG 1 および 2 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：カーネルクロックを無効にします（リセット後のデフォルト）。
01：**pll1_q_ck** クロックがカーネルクロックとして選択されます。
10：**pll3_q_ck** クロックがカーネルクロックとして選択されます。
11：**hsi48_ck** クロックがカーネルクロックとして選択されます。

ビット 19:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **I2C123SEL[1:0]**：I2C1、2、および 3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**rcc_pclk1** クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll3_r_ck** クロックがカーネルクロックとして選択されます。
10：**hsi_ker_ck** クロックがカーネルクロックとして選択されます。
11：**csi_ker_ck** クロックがカーネルクロックとして選択されます。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RNGSEL[1:0]**：RNG カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**hsi8_ck** クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** クロックがカーネルクロックとして選択されます。
10：**lse_ck** クロックがカーネルクロックとして選択されます。
11：**lsi_ck** クロックがカーネルクロックとして選択されます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:3 **USART16SEL[2:0]**：USART1 および 6 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk2** クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** クロックがカーネルクロックとして選択されます。
010：**pll3_q_ck** クロックがカーネルクロックとして選択されます。
011：**hsi_ker_ck** クロックがカーネルクロックとして選択されます。
100：**csi_ker_ck** クロックがカーネルクロックとして選択されます。
101：**lse_ck** クロックがカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 2:0 **USART234578SEL[2:0]**：USART2/3、UART4、5、7/8 （APB1）カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk1** クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** クロックがカーネルクロックとして選択されます。
010：**pll3_q_ck** クロックがカーネルクロックとして選択されます。
011：**hsi_ker_ck** クロックがカーネルクロックとして選択されます。
100：**csi_ker_ck** クロックがカーネルクロックとして選択されます。
101：**lse_ck** クロックがカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

8.7.20 RCC ドメイン 3 カーネルクロック設定レジスタ（RCC_D3CCIPR）

アドレスオフセット：0x058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SPI6SEL[2:0] (1)			Res.	SAI4BSEL[2:0] (1)			SAI4ASEL[2:0] (1)			Res.	Res.	Res.	ADCSEL[1:0] (1)	
	rw				rw			rw						rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPTIM345SEL[2:0] (1)			LPTIM2SEL[2:0] (1)			I2C4SEL[1:0] (1)		Res.	Res.	Res.	Res.	Res.	LPUART1SEL[2:0] (1)		
rw			rw			rw							rw		

1. タイミング違反を生成することなく、動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。*セクション：クロックスイッチとクロックゲーティング*を参照してください。

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **SPI6SEL[2:0]**：SPI6 カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

000：**rcc_pclk4** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：**hsi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

100：**csi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

101：**hse_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SAI4BSEL[2:0]**：SAI4 カーネルクロックソースの選択のサブブロック B

ソフトウェアでセット／クリアされます。

注意： ***選択されたクロックが外部クロックで、これが停止された場合、別のクロックへの切り替えはできません。詳細については、セクション：クロックスイッチとクロックゲーティングを参照してください。***

000：**pll1_q_ck** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：I2S_CKIN クロックがカーネルペリフェラルクロックとして選択されます。

100：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

注： ***I2S_CKIN は、ピンから取得した外部クロックです。***

ビット 23:21 **SAI4ASEL[2:0]**：SAI4 カーネルクロックソースの選択のサブブロック A

ソフトウェアでセット／クリアされます。

注意： ***選択されたクロックが外部クロックで、これが停止された場合、別のクロックへの切り替えはできません。詳細については、セクション：クロックスイッチとクロックゲーティングを参照してください。***

000：**pll1_q_ck** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：I2S_CKIN クロックがカーネルペリフェラルクロックとして選択されます。

100：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

注： ***I2S_CKIN は、ピンから取得した外部クロックです。***

ビット 20:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **ADCSEL[1:0]**：SAR ADC カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

00：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

01：**pll3_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。

10：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

ビット 15:13 **LPTIM345SEL[2:0]**：LPTIM3、4、5 カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

000：**rcc_pclk4** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：**lse_ck** クロックがカーネルペリフェラルクロックとして選択されます。

100：**lsi_ck** クロックがカーネルペリフェラルクロックとして選択されます。

101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

ビット 12:10 **LPTIM2SEL[2:0]**：LPTIM2 カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

000：**rcc_pclk4** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_p_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：**lse_ck** クロックがカーネルペリフェラルクロックとして選択されます。

100：**lsi_ck** クロックがカーネルペリフェラルクロックとして選択されます。

101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

ビット 9:8 **I2C4SEL[1:0]**：I2C4 カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

00：**rcc_pclk4** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

01：**pll3_r_ck** クロックがカーネルペリフェラルクロックとして選択されます。

10：**hsi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

11：**csi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LPUART1SEL[2:0]**：LPUART1 カーネルクロックソースの選択

ソフトウェアでセット／クリアされます。

000：**rcc_pclk_d3** クロックがカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

001：**pll2_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。

010：**pll3_q_ck** クロックがカーネルペリフェラルクロックとして選択されます。

011：**hsi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

100：**csi_ker_ck** クロックがカーネルペリフェラルクロックとして選択されます。

101：**lse_ck** クロックがカーネルペリフェラルクロックとして選択されます。

その他：予約済み、カーネルクロックは無効です。

8.7.21 RCC クロックソース割り込み有効レジスタ（RCC_CIER）

アドレスオフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LSECSSIE	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	CSIRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LSECSSIE**：LSE クロックセキュリティシステム割り込みイネーブル

外部 32 kHz オシレータのクロックセキュリティシステムによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：LSE CSS 割り込みは無効です（リセット後のデフォルト）。

1：LSE CSS 割り込みは有効です。

ビット 8 **PLL3RDYIE**：PLL3 レディ割り込みイネーブル

PLL3 ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL3 ロック割り込み無効（リセット後のデフォルト）

1：PLL3 ロック割り込み有効

ビット 7 **PLL2RDYIE**：PLL2 レディ割り込みイネーブル

PLL2 ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL2 ロック割り込み無効（リセット後のデフォルト）

1：PLL2 ロック割り込み有効

ビット 6 **PLL1RDYIE**：PLL1 レディ割り込みイネーブル

PLL1 ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL1 ロック割り込み無効（リセット後のデフォルト）

1：PLL1 ロック割り込み有効

ビット 5 **HSI48RDYIE**：HSI48 レディ割り込みイネーブル

HSI48 オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：HSI48 レディ割り込みは無効です（リセット後のデフォルト）。

1：HSI48 レディ割り込みは有効です。

ビット 4 **CSIRDYIE**：CSI レディ割り込みイネーブル

CSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：CSI レディ割り込みは無効です（リセット後のデフォルト）。

1：CSI レディ割り込みは有効です。

ビット 3 **HSERDYIE** : HSE レディ割り込みイネーブル

HSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : HSE レディ割り込みは無効です（リセット後のデフォルト）。

1 : HSE レディ割り込みは有効です。

ビット 2 **HSIRDYIE** : HSI レディ割り込みイネーブル

HSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : HSI レディ割り込み無効（リセット後のデフォルト）

1 : HSI レディ割り込み有効

ビット 1 **LSERDYIE** : LSE レディ割り込みイネーブル

LSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : LSE レディ割り込みは無効です（リセット後のデフォルト）。

1 : LSE レディ割り込みは有効です。

ビット 0 **LSIRDYIE** : LSI レディ割り込みイネーブル

LSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : LSI レディ割り込みは無効です（リセット後のデフォルト）。

1 : LSI レディ割り込みは有効です。

8.7.22 RCC クロックソース割り込みフラグレジスタ（RCC_CIFR）

アドレスオフセット：0x64

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSECSSF	LSECSSF	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	CSIRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSECSSF**：HSE クロックセキュリティシステム割り込みフラグ
HSECSSC ビットに書き込むことによりソフトウェアによってリセットされます。
HSE クロックに障害が発生した場合に、ハードウェアによってセットされます。
0：HSE クロック障害によるクロックセキュリティ割り込みは発生していません（リセット後のデフォルト）。
1：HSE クロック障害によってクロックセキュリティ割り込みが発生しました。

ビット 9 **LSECSSF**：LSE クロックセキュリティシステム割り込みフラグ
LSECSSC ビットに書き込むことによりソフトウェアによってリセットされます。
外部 32 kHz オシレータで障害が検出され、LSECSSIE がセットされたときに、ハードウェアによってセットされます。
0：外部 32 kHz オシレータに障害が検出されません（リセット後のデフォルト）。
1：外部 32 kHz オシレータに障害が検出されました。

ビット 8 **PLL3RDYF**：PLL3 レディ割り込みフラグ
PLL3RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL3 がロックされ、PLL3RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL3 ロックによるクロックレディ割り込みは発生していません（リセット後のデフォルト）。
1：PLL3 ロックによるクロックレディ割り込みが発生しました。

ビット 7 **PLL2RDYF**：PLL2 レディ割り込みフラグ
PLL2RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL2 がロックされ、PLL2RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL2 ロックによるクロックレディ割り込みは発生していません（リセット後のデフォルト）。
1：PLL2 ロックによるクロックレディ割り込みが発生しました。

ビット 6 **PLL1RDYF**：PLL1 レディ割り込みフラグ
PLL1RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL1 がロックされ、PLL1RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL1 ロックによるクロックレディ割り込みは発生していません（リセット後のデフォルト）。
1：PLL1 ロックによるクロックレディ割り込みが発生しました。

ビット 5 **HSI48RDYF**：HSI48 レディ割り込みフラグ
HSI48RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
HSI48 クロックが安定し、HSI48RDYIE がセットされているときに、ハードウェアによってセットされます。
0：HSI48 オシレータによるクロックレディ割り込みは発生していません（リセット後のデフォルト）。
1：HSI48 オシレータによるクロックレディ割り込みが発生しました。

ビット 4 **CSIRDYF** : CSI レディ割り込みフラグ

CSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。

CSI クロックが安定し、CSIRDYIE がセットされているときに、ハードウェアによってセットされます。

0 : CSI によるクロックレディ割り込みは発生していません（リセット後のデフォルト）。

1 : CSI によるクロックレディ割り込みが発生しました。

ビット 3 **HSERDYF** : HSE レディ割り込みフラグ

HSERDYC ビットに書き込むことによりソフトウェアによってリセットされます。

HSE クロックが安定し、HSERDYIE がセットされているときに、ハードウェアによってセットされます。

0 : HSE によるクロックレディ割り込みは発生していません（リセット後のデフォルト）。

1 : HSE によるクロックレディ割り込みが発生しました。

ビット 2 **HSIRDYF** : HSI レディ割り込みフラグ

HSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。

HSI クロックが安定し、HSIRDYIE がセットされているときに、ハードウェアによってセットされます。

0 : HSI によるクロックレディ割り込みは発生していません（リセット後のデフォルト）。

1 : HSI によるクロックレディ割り込みが発生しました。

ビット 1 **LSERDYF** : LSE レディ割り込みフラグ

LSERDYC ビットに書き込むことによりソフトウェアによってリセットされます。

LSE クロックが安定し、LSERDYIE がセットされているときに、ハードウェアによってセットされます。

0 : LSE によるクロックレディ割り込みは発生していません（リセット後のデフォルト）。

1 : LSE によるクロックレディ割り込みが発生しました。

ビット 0 **LSIRDYF** : LSI レディ割り込みフラグ

LSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。

LSI クロックが安定し、LSIRDYIE がセットされているときに、ハードウェアによってセットされます。

0 : LSI によるクロックレディ割り込みは発生していません（リセット後のデフォルト）。

1 : LSI によるクロックレディ割り込みが発生しました。

8.7.23 RCC クロックソース割り込みクリアレジスタ（RCC_CICR）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSECSSC	LSECSSC	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	CSIRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC
					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSECSSC**：HSE クロックセキュリティシステム割り込みクリア
HSECSSF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSECSSF への影響はありません（リセット後のデフォルト）。
1：HSECSSF はクリアされます。

ビット 9 **LSECSSC**：LSE クロックセキュリティシステム割り込みクリア
LSECSSF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSECSSF への影響はありません（リセット後のデフォルト）。
1：LSECSSF はクリアされます。

ビット 8 **PLL3RDYC**：PLL3 レディ割り込みクリア
PLL3RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL3RDYF への影響はありません（リセット後のデフォルト）。
1：PLL3RDYF がクリアされます。

ビット 7 **PLL2RDYC**：PLL2 レディ割り込みクリア
PLL2RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL2RDYF への影響はありません（リセット後のデフォルト）。
1：PLL2RDYF がクリアされます。

ビット 6 **PLL1RDYC**：PLL1 レディ割り込みクリア
PLL1RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL1RDYF への影響はありません（リセット後のデフォルト）。
1：PLL1RDYF がクリアされます。

ビット 5 **HSI48RDYC**：HSI48 レディ割り込みクリア
HSI48RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSI48RDYF への影響はありません（リセット後のデフォルト）。
1：HSI48RDYF がクリアされます。

ビット 4 **CSIRDYC**：CSI レディ割り込みクリア
CSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：CSIRDYF への影響はありません（リセット後のデフォルト）。
1：CSIRDYF はクリアされます。

ビット 3 **HSERDYC**：HSE レディ割り込みクリア
HSERDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSERDYF への影響はありません（リセット後のデフォルト）。
1：HSERDYF がクリアされます。

ビット 2 **HSIRDYC**：HSI レディ割り込みクリア
HSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSIRDYF への影響はありません（リセット後のデフォルト）。
1：HSIRDYF がクリアされます。

ビット 1 **LSERDYC**：LSE レディ割り込みクリア
LSERDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSERDYF への影響はありません（リセット後のデフォルト）。
1：LSERDYF がクリアされます。

ビット 0 **LSIRDYC**：LSI レディ割り込みクリア
LSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSIRDYF への影響はありません（リセット後のデフォルト）。
1：LSIRDYF がクリアされます。

8.7.24 RCC バックアップドメイン制御レジスタ（RCC_BDCR）

アドレスオフセット：0x070

リセット値：0 x 0000 0000、バックアップドメインリセットによってリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス。このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

システムリセット後、RCC_BDCR レジスタは書き込み保護されます。このレジスタを変更するには、*PWR 制御レジスタ 1 (PWR_CR1)* の DBP ビットを「1」にセットする必要があります。RCC_BDCR ビットは、バックアップドメインのリセット後にのみリセットされます（*セクション 8.4.6：バックアップドメインリセット*を参照）。その他の内部または外部リセットは、これらのビットに影響しません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		Res.	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
rw						rwo			r	rs	rw		rw	r	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BDRST**：バックアップドメインソフトウェアリセット

ソフトウェアでセット／クリアされます。

0：リセットは有効ではありません（バックアップドメインリセット後のデフォルト）。

1：VSW ドメイン全体をリセットします。

ビット 15 **RTCEN**：RTC クロック有効化

ソフトウェアでセット／クリアされます。

0：**rtc_ck** クロックは無効です（バックアップドメインリセット後のデフォルト）。

1：**rtc_ck** クロックは有効です。

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RTCSEL[1:0]**：RTC クロックソース選択

RTC のクロックソースを選択するために、ソフトウェアによってセットされます。これらのビットは 1 度だけ書き込めます（LSE で障害が検出された場合を除く）。これらのビットは、LSECSSON を有効にする前に書き込む必要があります。これらのビットをリセットし、また 1 度だけ書き込めるようにするために、BDRST ビットを使用できます。

HSE が RTC クロックとして選択されている場合：このクロックは、システムが STOP モードの場合、またはピンリセット（NRST）時に失われます。

00：クロックなし（バックアップドメインリセット後のデフォルト）。

01：LSE クロックが RTC クロックとして使用されます。

10：LSI クロックが RTC クロックとして使用されます。

11：RTCPRE の値で分周された HSE クロックが RTC クロックとして使用されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LSECSSD** : LSE クロックセキュリティシステム障害の検出

外部 32 kHz オシレータのクロックセキュリティシステムによって障害が検出されたことを示すために、ハードウェアによってセットされます。

0 : 32 kHz オシレータに障害が検出されません（バックアップドメインリセット後のデフォルト）。

1 : 32 kHz オシレータに障害が検出されました。

ビット 5 **LSECSSON** : LSE クロックセキュリティシステム有効化

32 kHz オシレータのクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。LSECSSON は、LSE を有効（LSEON を有効化）かつレディ状態（ハードウェアで LSERDY をセット）にし、RTCSEL を選択した後で、有効にする必要があります。

このビットは、一度有効にすると、LSE の障害検出（LSECSSD = 「1」）の後を除き、無効にすることはできません。この場合、LSECSSON をソフトウェアで無効にする**必要があります**。

0 : 32 kHz オシレータのクロックセキュリティシステムオフ（バックアップドメインリセット後のデフォルト）

1 : 32 kHz オシレータのクロックセキュリティシステムオン

ビット 4:3 **LSEDRV[1:0]** : LSE オシレータの駆動能力

LSE オシレータの駆動能力を選択するために、ソフトウェアによってセットされます。

00 : 最低駆動（バックアップドメインリセット後のデフォルト）

01 : 中低駆動

10 : 中高駆動

11 : 最高駆動

ビット 2 **LSEBYP** : LSE オシレータバイパス

デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／リセットされます。LSE が有効（LSEON）またはレディ状態（LSERDY =「1」）のときは、このビットの書き込みを行わないでください。

0 : LSE オシレータはバイパスされません（バックアップドメインリセット後のデフォルト）。

1 : LSE オシレータはバイパスされます。

ビット 1 **LSERDY** : LSE オシレータはレディ状態です。

LSE が安定していることを示すために、ハードウェアによってセット／リセットされます。このビットでは、LSEON が「0」に設定された後で、立ち下げるために、**lse_ck** クロックが 6 サイクル必要です。

0 : LSE オシレータはレディ状態ではありません（バックアップドメインリセット後のデフォルト）。

1 : LSE オシレータはレディ状態です。

ビット 0 **LSEON** : LSE オシレータ有効

ソフトウェアでセット／クリアされます。

0 : LSE オシレータオフ（バックアップドメインリセット後のデフォルト）。

1 : LSE オシレータオン

8.7.25 RCC クロック制御およびステータスレジスタ（RCC_CSR）

アドレスオフセット：0x074

リセット値：0x0000 0000

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSIRDY**：LSI オシレータはレディ状態です。

ロースピード内部 RC オシレータが安定していることを示すために、ハードウェアによってセット／リセットされます。

このビットでは、LSION が「0」に設定された後で、立ち下げるために、**lsi_ck** クロックが 3 サイクル必要です。

このビットは、LSE のクロックセキュリティシステム、またはロースピードウォッチドッグ、または RTC による LSI クロックのリクエストがある場合は、LSION が有効でなくてもセットできます。

0：LSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：LSI クロックはレディ状態です。

ビット 0 **LSION**：LSI オシレータイネーブル

ソフトウェアでセット／クリアされます。

0：LSI がオフになります（リセット後のデフォルト）。

1：LSI がオンになります。

8.7.26 RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）

アドレスオフセット：0x07C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CPURST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1RST
rs															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	QSPIRST	Res.	FMCRST	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECRST	DMA2DRST	Res.	Res.	Res.	MDMARST
	rw		rw							rw	rw				rw

ビット 31 **CPURST**：CPU リセット
ソフトウェアでセットされ、ハードウェアでリセットされます。
0：CPU をリセットしません（リセット後のデフォルト）。
1：CPU をリセットし、ビットを「0」に自動的にクリアします。

ビット 30:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1RST**：SDMMC1 および SDMMC1 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：SDMMC1 および SDMMC1 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：SDMMC1 および SDMMC1 遅延ブロックをリセットします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **QSPIRST**：QUADSPI および QUADSPI 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：QUADSPI および QUADSPI 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：QUADSPI および QUADSPI 遅延ブロックをリセットします。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCRST**：FMC ブロックリセット
ソフトウェアでセット／クリアされます。
0：FMC ブロックをリセットしません（リセット後のデフォルト）。
1：FMC ブロックをリセットします。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECRST**：JPGDEC ブロックリセット
ソフトウェアでセット／クリアされます。
0：JPGDEC ブロックをリセットしません（リセット後のデフォルト）。
1：JPGDEC ブロックをリセットします。

ビット 4 **DMA2DRST**：DMA2D ブロックリセット
ソフトウェアでセット／クリアされます。
0：DMA2D ブロックをリセットしません（リセット後のデフォルト）。
1：DMA2D ブロックをリセットします。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMARST**：MDMA ブロックリセット
ソフトウェアでセット／クリアされます。
0：MDMA ブロックをリセットしません（リセット後のデフォルト）。
1：MDMA ブロックをリセットします。

8.7.27 RCC AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	USB2OTGRST	Res.	USB1OTGRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw		rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETH1MACRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12RST	Res.	Res.	Res.	DMA2RST	DMA1RST
rw										rw				rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **USB2OTGRST**：USB2OTG ブロックリセット
ソフトウェアでセット／クリアされます。
0：USB2OTG ブロックをリセットしません（リセット後のデフォルト）。
1：USB2OTG ブロックをリセットします。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **USB1OTGRST**：USB1OTG ブロックリセット
ソフトウェアでセット／クリアされます。
0：USB1OTG ブロックをリセットしません（リセット後のデフォルト）。
1：USB1OTG ブロックをリセットします。

ビット 24:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ETH1MACRST**：ETH1MAC ブロックリセット
ソフトウェアでセット／クリアされます。
0：ETH1MAC ブロックをリセットしません（リセット後のデフォルト）。
1：ETH1MAC ブロックをリセットします。

ビット 14:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12RST**：ADC1&2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：ADC1&2 ブロックをリセットしません（リセット後のデフォルト）。
1：ADC1&2 ブロックをリセットします。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2RST**：DMA2 ブロックリセット

ソフトウェアでセット／クリアされます。

0：DMA2 ブロックをリセットしません（リセット後のデフォルト）。

1：DMA2 ブロックをリセットします。

ビット 0 **DMA1RST**：DMA1 ブロックリセット

ソフトウェアでセット／クリアされます。

0：DMA1 ブロックをリセットしません（リセット後のデフォルト）。

1：DMA1 ブロックをリセットします。

8.7.28 RCC AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）

アドレスオフセット：0x084

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2RST	Res.	Res.	RNGRST	HASHRST	CRYPTRST	Res.	Res.	Res.	CAMITFRST
						rw			rw	rw	rw				rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2RST**：SDMMC2 および SDMMC2 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：SDMMC2 および SDMMC2 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：SDMMC2 および SDMMC2 遅延ブロックをリセットします。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGRST**：乱数発生回路ブロックリセット
ソフトウェアでセット／クリアされます。
0：RNG ブロックをリセットしません（リセット後のデフォルト）。
1：RNG ブロックをリセットします。

ビット 5 **HASHRST**：ハッシュブロックリセット
ソフトウェアでセット／クリアされます。
0：ハッシュブロックをリセットしません（リセット後のデフォルト）。
1：ハッシュブロックをリセットします。

ビット 4 **CRYPTRST**：暗号ブロックリセット
ソフトウェアでセット／クリアされます。
0：暗号ブロックをリセットしません（リセット後のデフォルト）。
1：暗号ブロックをリセットします。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CAMITFRST**：CAMITF ブロックリセット
ソフトウェアでセット／クリアされます。
0：CAMITF ブロックをリセットしません（リセット後のデフォルト）。
1：CAMITF ブロックをリセットします。

8.7.29 RCC AHB4 ペリフェラルリセットレジスタ（RCC_AHB4RSTR）

アドレスオフセット：0x088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HSEMRST	ADC3RST	Res.	Res.	BDMARST	Res.	CRCRST	Res.	Res.	Res.
						rw	rw			rw		rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKRST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **HSEMRST**：HSEM ブロックリセット
ソフトウェアでセット／クリアされます。
0：HSEM ブロックをリセットしません（リセット後のデフォルト）。
1：HSEM ブロックをリセットします。

ビット 24 **ADC3RST**：ADC3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：ADC3 ブロックをリセットしません（リセット後のデフォルト）。
1：ADC3 ブロックをリセットします。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMARST**：BDMA ブロックリセット
ソフトウェアでセット／クリアされます。
0：BDMA ブロックをリセットしません（リセット後のデフォルト）。
1：BDMA ブロックをリセットします。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CRCRST**：CRC ブロックリセット
ソフトウェアでセット／クリアされます。
0：CRC ブロックをリセットしません（リセット後のデフォルト）。
1：CRC ブロックをリセットします。

ビット 18:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKRST**：GPIOK ブロックリセット
ソフトウェアでセット／クリアされます。
0：GPIOK ブロックをリセットしません（リセット後のデフォルト）。
1：GPIOK ブロックをリセットします。

ビット 9 **GPIOJRST**：GPIOJ ブロックリセット
ソフトウェアでセット／クリアされます。
0：GPIOJ ブロックをリセットしません（リセット後のデフォルト）。
1：GPIOJ ブロックをリセットします。

ビット 8 **GPIOIRST** : GPIOI ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOI ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOI ブロックをリセットします。

ビット 7 **GPIOHRST** : GPIOH ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOH ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOH ブロックをリセットします。

ビット 6 **GPIOGRST** : GPIOG ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOG ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOG ブロックをリセットします。

ビット 5 **GPIOFRST** : GPIOF ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOF ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOF ブロックをリセットします。

ビット 4 **GPIOERST** : GPIOE ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOE ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOE ブロックをリセットします。

ビット 3 **GPIODRST** : GPIOD ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOD ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOD ブロックをリセットします。

ビット 2 **GPIOCRST** : GPIOC ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOC ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOC ブロックをリセットします。

ビット 1 **GPIOBRST** : GPIOB ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOB ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOB ブロックをリセットします。

ビット 0 **GPIOARST** : GPIOA ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOA ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOA ブロックをリセットします。

8.7.30 RCC APB3 ペリフェラルリセットレジスタ（RCC_APB3RSTR）

アドレスオフセット：0x08C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCRST**：LTDC ブロックリセット

ソフトウェアでセット／クリアされます。

0：LTDC ブロックをリセットしません（リセット後のデフォルト）。

1：LTDC ブロックをリセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.31 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1LRSTR）

アドレスオフセット：0x090

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
USART8RST	USART7RST	DAC12RST	Res.	HDMICECRST	Res.	Res.	Res.	I2C3RST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	SPDIFRXRST
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3RST	SPI2RST	Res.	Res.	Res.	Res.	LPTIM1RST	TIM14RST	TIM13RST	TIM12RST	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **USART8RST**：USART8 ブロックリセット
ソフトウェアでセット／クリアされます。
0：USART8 ブロックをリセットしません（リセット後のデフォルト）。
1：SWPMI USART8 をリセットします。

ビット 30 **USART7RST**：USART7 ブロックリセット
ソフトウェアでセット／クリアされます。
0：USART7 ブロックをリセットしません（リセット後のデフォルト）。
1：USART7 ブロックをリセットします。

ビット 29 **DAC12RST**：DAC1&2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DAC1&2 ブロックをリセットしません（リセット後のデフォルト）。
1：DAC1&2 ブロックをリセットします。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **HDMICECRST**：HDMI-CEC ブロックリセット
ソフトウェアでセット／クリアされます。
0：HDMI-CEC ブロックをリセットしません（リセット後のデフォルト）。
1：HDMI-CEC ブロックをリセットします。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3RST**：I2C3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C3 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C3 ブロックをリセットします。

ビット 22 **I2C2RST**：I2C2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C2 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C2 ブロックをリセットします。

ビット 21 **I2C1RST**：I2C1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C1 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C1 ブロックをリセットします。

ビット 20 **UART5RST** : UART5 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : UART5 ブロックをリセットしません（リセット後のデフォルト）。
1 : UART5 ブロックをリセットします。

ビット 19 **UART4RST** : UART4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : UART4 ブロックをリセットしません（リセット後のデフォルト）。
1 : UART4 ブロックをリセットします。

ビット 18 **USART3RST** : USART3 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART3 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART3 ブロックをリセットします。

ビット 17 **USART2RST** : USART2 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART2 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART2 ブロックをリセットします。

ビット 16 **SPDIFRXRST** : SPDIFRX ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPDIFRX ブロックをリセットしません（リセット後のデフォルト）。
1 : SPDIFRX ブロックをリセットします。

ビット 15 **SPI3RST** : SPI3 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI3 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI3 ブロックをリセットします。

ビット 14 **SPI2RST** : SPI2 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI2 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI2 ブロックをリセットします。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1RST** : LPTIM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : LPTIM1 ブロックをリセットしません（リセット後のデフォルト）。
1 : LPTIM1 ブロックをリセットします。

ビット 8 **TIM14RST** : TIM14 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM14 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM14 ブロックをリセットします。

ビット 7 **TIM13RST** : TIM13 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM13 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM13 ブロックをリセットします。

ビット 6 **TIM12RST** : TIM12 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM12 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM12 ブロックをリセットします。

ビット 5 **TIM7RST** : TIM7 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM7 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM7 ブロックをリセットします。

ビット 4 **TIM6RST** : TIM6 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM6 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM6 ブロックをリセットします。

ビット 3 **TIM5RST** : TIM5 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM5 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM5 ブロックをリセットします。

ビット 2 **TIM4RST** : TIM4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM4 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM4 ブロックをリセットします。

ビット 1 **TIM3RST** : TIM3 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM3 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM3 ブロックをリセットします。

ビット 0 **TIM2RST** : TIM2 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM2 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM2 ブロックをリセットします。

8.7.32 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1HRSTR）

アドレスオフセット：0x094

リセット値：0x0000 0000

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANRST**：FDCAN ブロックリセット
ソフトウェアでセット／クリアされます。
0：FDCAN ブロックをリセットしません（リセット後のデフォルト）。
1：FDCAN ブロックをリセットします。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSRST**：MDIOS ブロックリセット
ソフトウェアでセット／クリアされます。
0：MDIOS ブロックをリセットしません（リセット後のデフォルト）。
1：MDIOS ブロックをリセットします。

ビット 4 **OPAMPRST**：OPAMP ブロックリセット
ソフトウェアでセット／クリアされます。
0：OPAMP ブロックをリセットしません（リセット後のデフォルト）。
1：OPAMP ブロックをリセットします。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPRST**：SWPMI ブロックリセット
ソフトウェアでセット／クリアされます。
0：SWPMI ブロックをリセットしません（リセット後のデフォルト）。
1：SWPMI ブロックをリセットします。

ビット 1 **CRSRST**：クロックリカバリシステムリセット
ソフトウェアでセット／クリアされます。
0：CRS をリセットしません（リセット後のデフォルト）。
1：CRS をリセットします。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.33 RCC APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）

アドレスオフセット：0x098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	HRTIMRST	DFSDM1RST	Res.	Res.	Res.	SAI3RST	SAI2RST	SAI1RST	Res.	SPI5RST	Res.	TIM17RST	TIM16RST	TIM15RST
		rw	rw				rw	rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4RST	SPI1RST	Res.	Res.	Res.	Res.	Res.	Res.	USART6RST	USART1RST	Res.	Res.	TIM8RST	TIM1RST
		rw	rw							rw	rw			rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **HRTIMRST**：HRTIM ブロックリセット
ソフトウェアでセット／クリアされます。
0：HRTIM ブロックをリセットしません（リセット後のデフォルト）。
1：HRTIM ブロックをリセットします。

ビット 28 **DFSDM1RST**：DFSDM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DFSDM1 ブロックをリセットしません（リセット後のデフォルト）。
1：DFSDM1 ブロックをリセットします。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SAI3RST**：SAI3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI3 ブロックをリセットしません（リセット後のデフォルト）。
1：SAI3 ブロックをリセットします。

ビット 23 **SAI2RST**：SAI2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI2 ブロックをリセットしません（リセット後のデフォルト）。
1：SAI2 ブロックをリセットします。

ビット 22 **SAI1RST**：SAI1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI1 をリセットしません（リセット後のデフォルト）。
1：SAI1 をリセットします。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5RST**：SPI5 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SPI5 ブロックをリセットしません（リセット後のデフォルト）。
1：SPI5 ブロックをリセットします。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17RST** : TIM17 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM17 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM17 ブロックをリセットします。

ビット 17 **TIM16RST** : TIM16 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM16 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM16 ブロックをリセットします。

ビット 16 **TIM15RST** : TIM15 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM15 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM15 ブロックをリセットします。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4RST** : SPI4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI4 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI4 ブロックをリセットします。

ビット 12 **SPI1RST** : SPI1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI1 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI1 ブロックをリセットします。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **USART6RST** : USART6 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART6 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART6 ブロックをリセットします。

ビット 4 **USART1RST** : USART1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART1 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART1 ブロックをリセットします。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8RST** : TIM8 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM8 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM8 ブロックをリセットします。

ビット 0 **TIM1RST** : TIM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM1 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM1 ブロックをリセットします。

8.7.34 RCC APB4 ペリフェラルリセットレジスタ（RCC_APB4RSTR）

アドレスオフセット：0x09C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4RST	Res.	Res.	Res.	Res.	Res.
										rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFRST	COMP12RST	Res.	LPTIM5RST	LPTIM4RST	LPTIM3RST	LPTIM2RST	Res.	I2C4RST	Res.	SPI6RST	Res.	LPUART1RST	Res.	SYSCFGRST	Res.
rw	rw		rw	rw	rw	rw		rw		rw		rw		rw	

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI4RST**：SAI4 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI4 ブロックをリセットしません（リセット後のデフォルト）。
1：SAI4 ブロックをリセットします。

ビット 20:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **VREFRST**：VREF ブロックリセット
ソフトウェアでセット／クリアされます。
0：VREF ブロックをリセットしません（リセット後のデフォルト）。
1：VREF ブロックをリセットします。

ビット 14 **COMP12RST**：COMP12 ブロックリセット
ソフトウェアでセット／クリアされます。
0：COMP1&2 ブロックをリセットしません（リセット後のデフォルト）。
1：COMP1&2 ブロックをリセットします。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPTIM5RST**：LPTIM5 ブロックリセット
ソフトウェアでセット／クリアされます。
0：LPTIM5 ブロックをリセットしません（リセット後のデフォルト）。
1：LPTIM5 ブロックをリセットします。

ビット 11 **LPTIM4RST**：LPTIM4 ブロックリセット
ソフトウェアでセット／クリアされます。
0：LPTIM4 ブロックをリセットしません（リセット後のデフォルト）。
1：LPTIM4 ブロックをリセットします。

ビット 10 **LPTIM3RST**：LPTIM3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：LPTIM3 ブロックをリセットしません（リセット後のデフォルト）。
1：LPTIM3 ブロックをリセットします。

ビット 9 **LPTIM2RST** : LPTIM2 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : LPTIM2 ブロックをリセットしません（リセット後のデフォルト）。
1 : LPTIM2 ブロックをリセットします。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4RST** : I2C4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : I2C4 ブロックをリセットしません（リセット後のデフォルト）。
1 : I2C4 ブロックをリセットします。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6RST** : SPI6 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI6 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI6 ブロックをリセットします。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1RST** : LPUART1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : LPUART1 ブロックをリセットしません（リセット後のデフォルト）。
1 : LPUART1 ブロックをリセットします。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGRST** : SYSCFG ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SYSCFG ブロックをリセットしません（リセット後のデフォルト）。
1 : SYSCFG ブロックをリセットします。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.35 RCC グローバル制御レジスタ（RCC_GCR）

アドレスオフセット：0x0A0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WW1RSC
															rw1

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **WW1RSC**：WWDG1 リセット範囲制御

このビットは、ソフトウェアによってセットでき、ハードウェアによってシステムリセット中にクリアされます。適切に動作するよう、WWDG1 を有効にする前に、このビットを「1」にセットする必要があります。

8.7.36 RCC D3 自律モードレジスタ（RCC_D3AMR）

自律モードでは、CPU が CStop モードの場合でも、D3 に割り当てられたペリフェラルにペリフェラルクロックを供給できます。ペリフェラルが有効で、その自律ビットが有効化されている場合、ペリフェラルは D3 ドメインの状態に従ってペリフェラルクロックを受信します（CPU が CStop モードの場合）。

アドレスオフセット：0x0A8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SRAM4AMEN	BKPRAMAMEN	Res.	Res.	Res.	ADC3AMEN	Res.	Res.	SAI4AMEN	Res.	CRCAMEN	Res.	Res.	RTCAMEN
		rw	rw				rw			rw		rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFAMEN	COMP12AMEN	Res.	LPTIM5AMEN	LPTIM4AMEN	LPTIM3AMEN	LPTIM2AMEN	Res.	I2C4AMEN	Res.	SPI6AMEN	Res.	LPUART1AMEN	Res.	Res.	BDMAAMEN
rw	rw		rw	rw	rw	rw		rw		rw		rw			rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **SRAM4AMEN**：SRAM4 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、SRAM4 クロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、SRAM4 ペリフェラルバスクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 28 **BKPSRAMAMEN**：バックアップ RAM 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、バックアップ RAM クロックは無効です（リセット後のデフォルト）。

1：バックアップ RAM クロックの有効化は D3 ドメインの状態で制御されます。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ADC3AMEN**：ADC3 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、ADC3 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、ADC3 ペリフェラルクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI4AMEN**：SAI4 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、SAI4 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、SAI4 ペリフェラルクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CRCAMEN**：CRC 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、CRC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、CRC ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAMEN**：RTC 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、RTC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、RTC ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 15 **VREFAMEN**：VREF 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、VREF ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、VREF ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 14 **COMP12AMEN**：COMP12 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、COMP12 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、COMP12 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPTIM5AMEN**：LPTIM5 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、LPTIM5 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 11 **LPTIM4AMEN**：LPTIM4 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、LPTIM4 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 10 **LPTIM3AMEN**：LPTIM3 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、LPTIM3 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 9 **LPTIM2AMEN**：LPTIM2 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、LPTIM2 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4AMEN**：I2C4 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、I2C4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：D3 ドメインが DRun の場合、I2C4 ペリフェラルクロックは有効です。
詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6AMEN**：SPI6 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、SPI6 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、SPI6 ペリフェラルクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1AMEN**：LPUART1 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、LPUART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、LPUART1 ペリフェラルクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BDMAAMEN**：BDMA および DMAMUX 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、BDMA および DMAMUX ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：D3 ドメインが DRun の場合、BDMA および DMAMUX ペリフェラルクロックは有効です。

詳細については、*セクション 8.5.11：ペリフェラルクロックゲーティングの制御*を参照してください。

8.7.37 RCC リセットステータスレジスタ（RCC_RSR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 56. RCC_RSR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_RSR	0x0D0	0x00FE 0000 (1)
RCC_C1_RSR	0x130	

1. パワーオンリセットによってのみリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス。このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPWRRSTF	Res.	WWDG1RSTF	Res.	IWDG1RSTF	Res.	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	D2RSTF	D1RSTF	Res.	CPURSTF	RMVF
	r		r		r		r	r	r	r	r	r		r	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **LPWRRSTF**：不正な D1 DStandby または CPU CStop フラグによるリセット (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
D1 ドメインが誤って DStandby に移行した場合、または CPU が誤って CStop に移行した場合に、ハードウェアによってセットされます。
0：不正なリセットは発生していません（パワーオンリセット後のデフォルト）。
1：不正な D1 DStandby または CPU CStop リセットが発生しました。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **WWDG1RSTF**：ウィンドウ型ウォッチドッグリセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
0：WWDG1 からウィンドウ型ウォッチドッグリセットは発生していません（パワーオンリセット後のデフォルト）。
1：WWDG1 からウィンドウ型ウォッチドッグリセットが発生しました。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **IWDG1RSTF**：独立型ウォッチドッグリセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
0：独立型独立型ウォッチドッグリセットは発生していません（パワーオンリセット後のデフォルト）。
1：独立型ウォッチドッグリセットが発生しました。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SFTRSTF**：CPU リセットフラグからのシステムリセット (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

CPU によってシステムがリセットされた場合に、ハードウェアによってセットされます。CPU では CM7 の AIRCR レジスタの SYSRESETREQ ビットを書き込むことで、システムリセットを生成できます。

0：CPU のソフトウェアリセットは発生していません（パワーオンリセット後のデフォルト）。

1：CPU によってシステムリセットが生成されます。

ビット 23 **PORRSTF**：POR/PDR リセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

POR/PDR リセットが発生したときに、ハードウェアによってセットされます。

0：POR/PDR リセットは発生していません。

1：POR/PDR リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 22 **PINRSTF**：ピンリセットフラグ（NRST） (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

ピンからリセットが発生したときに、ハードウェアによってセットされます。

0：ピンからのリセットは発生していません。

1：ピンからのリセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 21 **BORRSTF**：BOR リセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

BOR リセットが発生したときに、ハードウェアによってセットされます（**pwr_bor_rst**）。

0：BOR リセットは発生していません。

1：BOR リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 20 **D2RSTF**：D2 ドメインの電源スイッチリセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

D2 ドメインが DStandby を終了するとき、またはパワーオンリセット後に、ハードウェアによってセットされます。詳細については、*表 47*を参照してください。

0：D2 ドメインの電源スイッチリセットは発生していません。

1：D2 ドメインの電源スイッチ（ePOD2）リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 19 **D1RSTF**：D1 ドメインの電源スイッチリセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

D1 ドメインが DStandby を終了するとき、またはパワーオンリセット後に、ハードウェアによってセットされます。詳細については、*表 47*を参照してください。

0：D1 ドメインの電源スイッチリセットは発生していません。

1：D1 ドメインの電源スイッチ（ePOD1）リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CPURSTF**：CPU リセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。

CPU リセットが発生するたびに、ハードウェアによってセットされます。

0：CPU リセットは発生していません。

1：CPU リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 16 **RMVF**：リセットフラグ解除

リセットフラグの値をリセットするために、ソフトウェアによってセット／リセットされます。

0：リセットフラグのリセットは有効になりません（パワーオンリセット後のデフォルト）。

1：リセットフラグの値をリセットします。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

1. フラグの動作の詳細については、*表 47：リセットソース識別（RCC_RSR）*を参照してください。

8.7.38 RCC AHB3 クロックレジスタ（RCC_AHB3ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 57. RCC_AHB3ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB3ENR	0x0D4	0x0000 0000
RCC_C1_AHB3ENR	0x134	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1EN
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	QSPIEN	Res.	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECEN	DMA2DEN	Res.	Res.	Res.	MDMAEN
	rw		rw							rw	rw				rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1EN**：SDMMC1 および SDMMC1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：SDMMC1 および SDMMC1 遅延クロックは無効です（リセット後のデフォルト）。
1：SDMMC1 および SDMMC1 遅延クロックは有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **QSPIEN**：QUADSPI および QUADSPI 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：QUADSPI および QUADSPI 遅延クロックは無効です（リセット後のデフォルト）。
1：QUADSPI および QUADSPI 遅延クロックは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCEN**：FMC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：FMC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：FMC ペリフェラルクロックは有効です。
FMC のペリフェラルクロックは、FMCSEL によって選択され、fmc_ker_ck 入力に供給されるカーネルクロックと、**rcc_hclk3** バスインタフェースクロックです。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECEN**：JPGDEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：JPGDEC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：JPGDEC ペリフェラルクロックは有効です。

ビット 4 **DMA2DEN**：DMA2D ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DMA2D ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DMA2D ペリフェラルクロックは有効です。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMAEN**：MDMA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：MDMA ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：MDMA ペリフェラルクロックは有効です。

8.7.39 RCC AHB1 クロックレジスタ（RCC_AHB1ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 58. RCC_AHB1ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB1ENR	0x0D8	0x0000 0000
RCC_C1_AHB1ENR	0x138	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	USB2ULPIEN	USB2OTGEN	USB1ULPIEN	USB1OTGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXEN	ETH1TXEN
			rw	rw	rw	rw								rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETH1MACEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12EN	Res.	Res.	Res.	DMA2EN	DMA1EN
rw										rw				rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **USB2ULPIEN**：USB_PHY2 クロック有効化
ソフトウェアでセット／クリアされます。
0：USB_PHY2 クロックは無効です（リセット後のデフォルト）。
1：USB_PHY2 クロックは有効です。

ビット 27 **USB2OTGEN**：USB2OTG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USB2OTG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USB2OTG ペリフェラルクロックは有効です。
USB2OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 26 **USB1ULPIEN**：USB_PHY1 クロック有効化
ソフトウェアでセット／クリアされます。
0：USB1ULPI PHY クロックは無効です（リセット後のデフォルト）。
1：USB1ULPI PHY クロックは有効です。

ビット 25 **USB1OTGEN**：USB1OTG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USB1OTG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USB1OTG ペリフェラルクロックは有効です。
USB1OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 24:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **ETH1RXEN**：Ethernet 受信クロック有効化
ソフトウェアでセット／クリアされます。
0：Ethernet 受信クロック無効（リセット後のデフォルト）
1：Ethernet 受信クロック有効

ビット 16 **ETH1TXEN**：Ethernet 送信クロック有効化
ソフトウェアでセット／クリアされます。
0：Ethernet 送信クロック無効（リセット後のデフォルト）
1：Ethernet 送信クロック有効

ビット 15 **ETH1MACEN**：Ethernet MAC バスインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：Ethernet MAC バスインタフェースクロック無効（リセット後のデフォルト）
1：Ethernet MAC バスインタフェースクロック有効

ビット 14:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12EN**：ADC1/2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：ADC1 および 2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：ADC1 および 2 ペリフェラルクロックは有効です。
ADC1 および 2 のペリフェラルクロックは、ADCSEL によって選択され、adc_ker_ck 入力に供給されるカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2EN**：DMA2 クロック有効化
ソフトウェアでセット／クリアされます。
0：DMA2 クロック無効（リセット後のデフォルト）。
1：DMA2 クロック有効

ビット 0 **DMA1EN**：DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：DMA1 クロック無効（リセット後のデフォルト）。
1：DMA1 クロック有効

8.7.40 RCC AHB2 クロックレジスタ（RCC_AHB2ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 59. RCC_AHB2ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB2ENR	0x0DC	0x0000 0000
RCC_C1_AHB2ENR	0x13C	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM3EN	SRAM2EN	SRAM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2EN	Res.	Res.	RNGEN	HASHEN	CRYPTEN	Res.	Res.	Res.	DCMIEN
						rw			rw	rw	rw				rw

ビット 31 **SRAM3EN**：SRAM3 ブロック有効化

ソフトウェアでセット／クリアされます。

セットされた場合、このビットは、SRAM3 が CPU によって割り当てられていることを示します。これにより、CPU が CRun のときに D2 ドメインを DRun のまま保持する CPU 動作モードについても D2 ドメインで考慮するようになります。

0：SRAM3 インタフェースクロックは無効です（リセット後のデフォルト）。

1：SRAM3 インタフェースクロックは有効です。

ビット 30 **SRAM2EN**：SRAM2 ブロック有効化

ソフトウェアでセット／クリアされます。

セットされた場合、このビットは、SRAM2 が CPU によって割り当てられていることを示します。これにより、CPU が CRun のときに D2 ドメインを DRun のまま保持する CPU 動作モードについても D2 ドメインで考慮するようになります。

0：SRAM2 インタフェースクロックは無効です（リセット後のデフォルト）。

1：SRAM2 インタフェースクロックは有効です。

ビット 29 **SRAM1EN**：SRAM1 ブロック有効化

ソフトウェアでセット／クリアされます。

セットされた場合、このビットは、SRAM1 が CPU によって割り当てられていることを示します。これにより、CPU が CRun のときに D2 ドメインを DRun のまま保持する CPU 動作モードについても D2 ドメインで考慮するようになります。

0：SRAM1 インタフェースクロックは無効です（リセット後のデフォルト）。

1：SRAM1 インタフェースクロックは有効です。

ビット 28:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2EN**：SDMMC2 および SDMMC2 遅延クロック有効化

ソフトウェアでセット／クリアされます。

0：SDMMC2 および SDMMC2 遅延クロックは無効です（リセット後のデフォルト）。

1：SDMMC2 および SDMMC2 遅延クロックは有効です。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGEN**：RNG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：RNG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：RNG ペリフェラルクロックは有効です。
RNG のペリフェラルクロックは、RNGSEL によって選択され、**rng_ker_ck** 入力に供給されるカーネルクロックと、**rcc_hclk2** バスインタフェースクロックです。

ビット 5 **HASHEN**：HASH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：HASH ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：HASH ペリフェラルクロックは有効です。

ビット 4 **CRYPTEN**：CRYPT ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CRYPT ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：CRYPT ペリフェラルクロックは有効です。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **DCMIEN**：DCMI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DCMI ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DCMI ペリフェラルクロックは有効です。

8.7.41 RCC AHB4 クロックレジスタ（RCC_AHB4ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 60. RCC_AHB4ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB4ENR	0x0E0	0x0000 0000
RCC_C1_AHB4ENR	0x140	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKPRAMEN	Res.	Res.	HSEMEN	ADC3EN	Res.	Res.	BDMAEN	Res.	CRCEN	Res.	Res.	Res.
			rw			rw	rw			rw		rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **BKPRAMEN**：バックアップ RAM クロック有効化
ソフトウェアでセット／クリアされます。
0：バックアップ RAM クロックは無効です（リセット後のデフォルト）。
1：バックアップ RAM クロックは有効です。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **HSEMEN**：HSEM ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：HSEM ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：HSEM ペリフェラルクロックは有効です。

ビット 24 **ADC3EN**：ADC3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：ADC3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：ADC3 ペリフェラルクロックは有効です。
ADC3 のペリフェラルクロックは、ADCSEL によって選択され、adc_ker_ck 入力に供給されるカーネルクロックと、**rcc_hclk4** バスインタフェースクロックです。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMAEN**：BDMA および DMAMUX2 クロック有効化
ソフトウェアでセット／クリアされます。
0：BDMA および DMAMUX2 クロックは無効です（リセット後のデフォルト）。
1：BDMA および DMAMUX2 クロックは有効です。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CRCEN**：CRC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CRC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：CRC ペリフェラルクロックは有効です。

ビット 18:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKEN**：GPIOK ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOK ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOK ペリフェラルクロックは有効です。

ビット 9 **GPIOJEN**：GPIOJ ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOJ ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOJ ペリフェラルクロックは有効です。

ビット 8 **GPIOIEN**：GPIOI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOI ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOI ペリフェラルクロックは有効です。

ビット 7 **GPIOHEN**：GPIOH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOH ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOH ペリフェラルクロックは有効です。

ビット 6 **GPIOGEN**：GPIOG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOG ペリフェラルクロックは有効です。

ビット 5 **GPIOFEN**：GPIOF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOF ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOF ペリフェラルクロックは有効です。

ビット 4 **GPIOEEN**：GPIOE ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOE ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOE ペリフェラルクロックは有効です。

ビット 3 **GPIODEN**：GPIOD ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOD ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOD ペリフェラルクロックは有効です。

ビット 2 **GPIOCEN**：GPIOC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOC ペリフェラルクロックは有効です。

ビット 1 **GPIOBEN**：GPIOB ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOB ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOB ペリフェラルクロックは有効です。

ビット 0 **GPIOAEN**：GPIOA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOA ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOA ペリフェラルクロックは有効です。

8.7.42 RCC APB3 クロックレジスタ（RCC_APB3ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 61. RCC_APB3ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB3ENR	0x0E4	0x0000 0000
RCC_C1_APB3ENR	0x144	

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1EN	Res.	Res.	LTDCEN	Res.	Res.	Res.
									rs			rw			

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDG1EN**：WWDG1 クロック有効化

システムリセットが発生したときに、ソフトウェアによってセットされ、ハードウェアによってリセットされます。

適切に動作するよう、WWDG1 を有効にする前に、WW1RSC ビットを「1」にセットする必要があります。

0：WWDG1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：WWDG1 ペリフェラルクロックは有効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCEN**：LTDC ペリフェラルクロック有効化

ピクセルクロック（**ltdc_ker_ck**）を LTDC ブロックに供給します。

ソフトウェアでセット／クリアされます。

0：LTDC ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：LTDC ペリフェラルクロックを LTDC ブロックに供給します。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.43 RCC APB1 クロックレジスタ（RCC_APB1LENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 62. RCC_APB1ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB1LENR	0x0E8	0x0000 0000
RCC_C1_APB1LENR	0x148	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
USART8EN	USART7EN	DAC12EN	Res.	HDMICECEN	Res.	Res.	Res.	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	SPDIFRXEN
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3EN	SPI2EN	Res.	Res.	Res.	Res.	LPTIM1EN	TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **USART8EN**：USART8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART8 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART8 ペリフェラルクロックは有効です。
USART8 のペリフェラルクロックは、USART234578SEL によって選択され、usart_ker_ck 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 30 **USART7EN**：USART7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART7 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART7 ペリフェラルクロックは有効です。
USART7 のペリフェラルクロックは、USART234578SEL によって選択され、usart_ker_ck 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 29 **DAC12EN**：DAC1&2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DAC1&2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DAC1&2 ペリフェラルクロックは有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **HDMICECEN**：HDMI-CEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：HDMI-CEC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：HDMI-CEC ペリフェラルクロックは有効です。
HDMI-CEC のペリフェラルクロックは、CECSEL によって選択され、**cec_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3EN**：I2C3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C3 ペリフェラルクロックは有効です。
I2C3 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 22 **I2C2EN**：I2C2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C2 ペリフェラルクロックは有効です。
I2C2 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 21 **I2C1EN**：I2C1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C1 ペリフェラルクロックは有効です。
I2C1 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 20 **UART5EN**：UART5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART5 ペリフェラルクロックは有効です。
UART5 のペリフェラルクロックは、USART234578SEL によって選択され、**uart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 19 **UART4EN**：UART4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART4 ペリフェラルクロックは有効です。
UART4 のペリフェラルクロックは、USART234578SEL によって選択され、**uart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 18 **USART3EN**：USART3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART3 ペリフェラルクロックは有効です。
USART3 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 17 **USART2EN**：USART2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART2 ペリフェラルクロックは有効です。
USART2 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 16 **SPDIFRXEN**：SPDIFRX ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPDIFRX ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPDIFRX ペリフェラルクロックは有効です。
SPDIFRX のペリフェラルクロックは、SPDIFSEL によって選択され、**spdifrx_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 15 **SPI3EN**：SPI3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI3 ペリフェラルクロックは有効です。
SPI3 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 14 **SPI2EN**：SPI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI2 ペリフェラルクロックは有効です。
SPI2 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1EN**：LPTIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM1 ペリフェラルクロックは有効です。
LPTIM1 のペリフェラルクロックは、LPTIM1SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 8 **TIM14EN**：TIM14 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM14 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM14 ペリフェラルクロックは有効です。

ビット 7 **TIM13EN**：TIM13 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM13 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM13 ペリフェラルクロックは有効です。

ビット 6 **TIM12EN**：TIM12 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM12 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM12 ペリフェラルクロックは有効です。

ビット 5 **TIM7EN**：TIM7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM7 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM7 ペリフェラルクロックは有効です。

ビット 4 **TIM6EN**：TIM6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM6 ペリフェラルクロックは有効です。

ビット 3 **TIM5EN**：TIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM5 ペリフェラルクロックは有効です。

ビット 2 **TIM4EN**：TIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM4 ペリフェラルクロックは有効です。

ビット 1 **TIM3EN**：TIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM3 ペリフェラルクロックは有効です。

ビット 0 **TIM2EN**：TIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM2 ペリフェラルクロックは有効です。

8.7.44 RCC APB1 クロックレジスタ（RCC_APB1HENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 63. RCC_APB1ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB1HENR	0x0EC	0x0000 0000
RCC_C1_APB1HENR	0x14C	

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANEN	Res.	Res.	MDIOSEN	OPAMPEN	Res.	SWPEN	CRSEN	Res.
							rw			rw	rw		rw	rw	

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANEN**：FDCAN ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：FDCAN ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：FDCAN ペリフェラルクロックは有効です。
FDCAN のペリフェラルクロックは、FDCANSEL によって選択され、**fdcan_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSEN**：MDIOS ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：MDIOS ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：MDIOS ペリフェラルクロックは有効です。

ビット 4 **OPAMPEN**：OPAMP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：OPAMP ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：OPAMP ペリフェラルクロックは有効です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPEN**：SWPMI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SWPMI ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SWPMI ペリフェラルクロックは有効です。
SWPMI のペリフェラルクロックは、SWPSEL によって選択され、**swpmi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 1 **CRSEN**：クロックリカバリシステムペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CRS ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：CRS ペリフェラルクロックは有効です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.45 RCC APB2 クロックレジスタ（RCC_APB2ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 64. RCC_APB2ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB2ENR	0x0F0	0x0000 0000
RCC_C1_APB2ENR	0x150	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	HRTIMEN	DFSDM1EN	Res.	Res.	Res.	SAI3EN	SAI2EN	SAI1EN	Res.	SPI5EN	Res.	TIM17EN	TIM16EN	TIM15EN
		rw	rw				rw	rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4EN	SPI1EN	Res.	Res.	Res.	Res.	Res.	Res.	USART6EN	USART1EN	Res.	Res.	TIM8EN	TIM1EN
		rw	rw							rw	rw			rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **HRTIMEN**：HRTIM ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：HRTIM ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：HRTIM ペリフェラルクロックは有効です。

ビット 28 **DFSDM1EN**：DFSDM1 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：DFSDM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：DFSDM1 ペリフェラルクロックは有効です。

DFSDM1 ペリフェラルクロックは、SAI1SEL および DFSDM1SEL によって選択され、**Aclk** および **clk** 入力にそれぞれ供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SAI3EN**：SAI3 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SAI3 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SAI3 ペリフェラルクロックは有効です。

SAI3 のペリフェラルクロックは、SAI23SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 23 **SAI2EN**：SAI2 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SAI2 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SAI2 ペリフェラルクロックは有効です。

SAI2 のペリフェラルクロックは、SAI23SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 22 **SAI1EN**：SAI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SAI1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SAI1 ペリフェラルクロックは有効です。
SAI1 のペリフェラルクロックは、SAI1SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5EN**：SPI5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI5 ペリフェラルクロックは有効です。
SPI5 のペリフェラルクロックは、SPI45SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17EN**：TIM17 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM17 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM17 ペリフェラルクロックは有効です。

ビット 17 **TIM16EN**：TIM16 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM16 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM16 ペリフェラルクロックは有効です。

ビット 16 **TIM15EN**：TIM15 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM15 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM15 ペリフェラルクロックは有効です。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4EN**：SPI4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI4 ペリフェラルクロックは有効です。
SPI4 のペリフェラルクロックは、SPI45SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 12 **SPI1EN**：SPI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI1 ペリフェラルクロックは有効です。
SPI1 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **USART6EN**：USART6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART6 ペリフェラルクロックは有効です。
USART6 のペリフェラルクロックは、USART16SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 4 **USART1EN**：USART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART1 ペリフェラルクロックは有効です。
USART1 のペリフェラルクロックは、USART16SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8EN**：TIM8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM8 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM8 ペリフェラルクロックは有効です。

ビット 0 **TIM1EN**：TIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM1 ペリフェラルクロックは有効です。

8.7.46 RCC APB4 クロックレジスタ（RCC_APB4ENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 65. RCC_APB4ENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB4ENR	0x0F4	0x0001 0000
RCC_C1_APB4ENR	0x154	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4EN	Res.	Res.	Res.	Res.	RTCAPBEN
										rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFEN	COMP12EN	Res.	LPTIM5EN	LPTIM4EN	LPTIM3EN	LPTIM2EN	Res.	I2C4EN	Res.	SPI6EN	Res.	LPUART1EN	Res.	SYSCFGEN	Res.
rw	rw		rw	rw	rw	rw		rw		rw		rw		rw	

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI4EN**：SAI4 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SAI4 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SAI4 ペリフェラルクロックは有効です。

SAI4 のペリフェラルクロックは、SAI4ASEL および SAI4BSEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力にそれぞれ供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 20:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAPBEN**：RTC APB クロック有効化

ソフトウェアでセット／クリアされます。

0：RTC （APB）のレジスタクロックインタフェースは無効です。

1：RTC （APB）のレジスタクロックインタフェースは有効です（リセット後のデフォルト）。

ビット 15 **VREFEN**：VREF ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：VREF ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：VREF ペリフェラルクロックは有効です。

ビット 14 **COMP12EN**：COMP1/2 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：COMP1/2 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：COMP1/2 ペリフェラルクロックは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPTIM5EN**：LPTIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM5 ペリフェラルクロックは有効です。
LPTIM5 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 11 **LPTIM4EN**：LPTIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM4 ペリフェラルクロックは有効です。
LPTIM4 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 10 **LPTIM3EN**：LPTIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM3 ペリフェラルクロックは有効です。
LPTIM3 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 9 **LPTIM2EN**：LPTIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM2 ペリフェラルクロックは有効です。
LPTIM2 のペリフェラルクロックは、LPTIM2SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4EN**：I2C4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C4 ペリフェラルクロックは有効です。
I2C4 のペリフェラルクロックは、I2C4SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6EN**：SPI6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI6 ペリフェラルクロックは有効です。
SPI6 のペリフェラルクロックは、SPI6SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1EN**：LPUART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPUART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPUART1 ペリフェラルクロックは有効です。
LPUART1 のペリフェラルクロックは、LPUART1SEL によって選択され、**lpuart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGEN**：SYSCFG ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SYSCFG ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SYSCFG ペリフェラルクロックは有効です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.47 RCC AHB3 スリープクロックレジスタ（RCC_AHB3LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 66. RCC_AHB3LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB3LPENR	0x0FC	0xF001 5131
RCC_C1_AHB3LPENR	0x15C	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AXISRAMLPEN	ITCMLPEN	DTCM2LPEN	DTCM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1LPEN
rw	rw	rw	rw												rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	QSPILPEN	Res.	FMCLPEN	Res.	Res.	Res.	FLASHLPEN	Res.	Res.	JPGDECLPEN	DMA2DLPEN	Res.	Res.	Res.	MDMALPEN
	rw		rw				rw			rw	rw				rw

ビット 31 **AXISRAMLPEN**：CSleep モード時 AXISRAM ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AXISRAM インタフェースクロックは無効です。
1：CSleep モード時の AXISRAM インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 30 **ITCMLPEN**：CSleep モード時 D1ITCM ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の D1 ITCM インタフェースクロックは無効です。
1：CSleep モード時の D1 ITCM インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 29 **DTCM2LPEN**：CSleep モード時 D1 DTCM2 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の D1 DTCM2 インタフェースクロックは無効です。
1：CSleep モード時の D1 DTCM2 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 28 **D1DTCM1LPEN**：CSleep モード時 D1DTCM1 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の D1DTCM1 インタフェースクロックは無効です。
1：CSleep モード時の D1DTCM1 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 27:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1LPEN**：CSleep モード時 SDMMC1 および SDMMC1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SDMMC1 および SDMMC1 遅延クロックは無効です。
1：CSleep モード時の SDMMC1 および SDMMC1 遅延クロックは有効です（リセット後のデフォルト）。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **QSPILPEN**：CSleep モード時 QUADSPI および QUADSPI 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の QUADSPI および QUADSPI 遅延クロックは無効です。
1：CSleep モード時の QUADSPI および QUADSPI 遅延クロックは有効です（リセット後のデフォルト）。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCLPEN**：CSleep モード時 FMC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の FMC ペリフェラルクロックは無効です。
1：CSleep モード時の FMC ペリフェラルクロックは有効です（リセット後のデフォルト）。
FMC のペリフェラルクロックは、FMCSEL によって選択され、**fmc_ker_ck** 入力に供給されるカーネルクロックと、**rcc_hclk3** バスインタフェースクロックです。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FLASHLPEN**：CSleep モード時フラッシュインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時のフラッシュインタフェースクロックは無効です。
1：CSleep モード時のフラッシュインタフェースクロックは有効です（リセット後のデフォルト）。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECLPEN**：CSleep モード時 JPGDEC クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の JPGDEC ペリフェラルクロックは無効です。
1：CSleep モード時の JPGDEC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **DMA2DLPEN**：CSleep モード時 DMA2D クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA2D ペリフェラルクロックは無効です。
1：CSleep モード時の DMA2D ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMALPEN**：CSleep モード時 MDMA クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の MDMA ペリフェラルクロックは無効です。
1：CSleep モード時の MDMA ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.48 RCC AHB1 スリープクロックレジスタ（RCC_AHB1LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 67. RCC_AHB1LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB1LPENR	0x100	0x1E03 C023
RCC_C1_AHB1LPENR	0x160	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	USB2ULPILPEN	USB2OTGLPEN	USB1ULPILPEN	USB1OTGLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXLPEN	ETH1TXLPEN
			rw	rw	rw	rw								rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETH1MACLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12LPEN	Res.	Res.	Res.	DMA2LPEN	DMA1LPEN
rw										rw				rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **USB2ULPILPEN**：CSleep モード時 USB_PHY2 クロック有効化

ソフトウェアでセット／クリアされます。

0：CSleep モード時の USB_PHY2 クロックは無効です。

1：CSleep モード時の USB_PHY2 クロックは有効です（リセット後のデフォルト）。

ビット 27 **USB2OTGLPEN**：CSleep モード時 USB2OTG ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：CSleep モード時の USB2OTG ペリフェラルクロックは無効です。

1：CSleep モード時の USB2OTG ペリフェラルクロックは有効です（リセット後のデフォルト）。

USB2OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 26 **USB1ULPILPEN**：CSleep モード時 USB_PHY1 クロック有効化

ソフトウェアでセット／クリアされます。

0：CSleep モード時の USB_PHY1 ペリフェラルクロックは無効です。

1：CSleep モード時の USB_PHY1 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 25 **USB1OTGLPEN**：CSleep モード時 USB1OTG ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：CSleep モード時の USB1OTG ペリフェラルクロックは無効です。

1：CSleep モード時の USB1OTG ペリフェラルクロックは有効です（リセット後のデフォルト）。

USB1OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 24:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **ETH1RXLPEN**：CSleep モード時 Ethernet 受信クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の Ethernet 受信クロックは無効です。
1：CSleep モード時の Ethernet 受信クロックは有効です（リセット後のデフォルト）。

ビット 16 **ETH1TXLPEN**：CSleep モード時 Ethernet 送信クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の Ethernet 送信クロックは無効です。
1：CSleep モード時の Ethernet 送信クロックは有効です（リセット後のデフォルト）。

ビット 15 **ETH1MACLPEN**：CSleep モード時 Ethernet MAC バスインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の Ethernet MAC バスインタフェースクロックは無効です。
1：CSleep モード時の Ethernet MAC バスインタフェースクロックは有効です（リセット後のデフォルト）。

ビット 14:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12LPEN**：CSleep モード時 ADC1/2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の ADC1/2 ペリフェラルクロックは無効です。
1：CSleep モード時の ADC1/2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
ADC1 および 2 のペリフェラルクロックは、ADCSEL によって選択され、adc_ker_ck 入力に供給されるカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2LPEN**：CSleep モード時 DMA2 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA2 クロックは無効です。
1：CSleep モード時の DMA2 クロックは有効です（リセット後のデフォルト）。

ビット 0 **DMA1LPEN**：CSleep モード時 DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA1 クロックは無効です。
1：CSleep モード時の DMA1 クロックは有効です（リセット後のデフォルト）。

8.7.49 RCC AHB2 スリープクロックレジスタ（RCC_AHB2LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 68. RCC_AHB2LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB2LPENR	0x104	0xE000 0271
RCC_C1_AHB2LPENR	0x164	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM3LPEN	SRAM2LPEN	SRAM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2LPEN	Res.	Res.	RNGLPEN	HASHLPEN	CRYPTLPEN	Res.	Res.	Res.	CAMITFLPEN
						rw			rw	rw	rw				rw

ビット 31 **SRAM3LPEN**：CSleep モード時 SRAM3 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SRAM3 クロックは無効です。
1：CSleep モード時の SRAM3 クロックは有効です（リセット後のデフォルト）。

ビット 30 **SRAM2LPEN**：CSleep モード時 SRAM2 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SRAM2 クロックは無効です。
1：CSleep モード時の SRAM2 クロックは有効です（リセット後のデフォルト）。

ビット 29 **SRAM1LPEN**：CSleep モード時 SRAM1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SRAM1 クロックは無効です。
1：CSleep モード時の SRAM1 クロックは有効です（リセット後のデフォルト）。

ビット 28:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2LPEN**：CSleep モード時 SDMMC2 および SDMMC2 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SDMMC2 および SDMMC2 遅延クロックは無効です。
1：CSleep モード時の SDMMC2 および SDMMC2 遅延クロックは有効です（リセット後のデフォルト）。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGLPEN**：CSleep モード時 RNG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の RNG ペリフェラルクロックは無効です。
1：CSleep モード時の RNG ペリフェラルクロックは有効です（リセット後のデフォルト）。
RNG のペリフェラルクロックは、RNGSEL によって選択され、**rng_ker_ck** 入力に供給されるカーネルクロックと、**rcc_hclk2** バスインタフェースクロックです。

ビット 5 **HASHLPEN**：CSleep モード時 HASH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の HASH ペリフェラルクロックは無効です。
1：CSleep モード時の HASH ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **CRYPTLPEN**：CSleep モード時 CRYPT ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRYPT ペリフェラルクロックは無効です。
1：CSleep モード時の CRYPT ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **DCMILPEN**：CSleep モード時 DCMI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DCMI ペリフェラルクロックは無効です。
1：CSleep モード時の DCMI ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.50 RCC AHB4 スリープクロックレジスタ（RCC_AHB4LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 69. RCC_AHB4LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_AHB4LPENR	0x108	0x3128 07FF
RCC_C1_AHB4LPENR	0x168	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SRAM4LPEN	BKPRAMLPEN	Res.	Res.	Res.	ADC3LPEN	Res.	Res.	BDMALPEN	Res.	CRCLPEN	Res.	Res.	Res.
		rw	rw				rw			rw		rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKLPEN	GPIOJLPEN	GPIOILPEN	GPIOHLPEN	GPIOGLPEN	GPIOFLPEN	GPIOELPEN	GPIODLPEN	GPIOCLPEN	GPIOBLPEN	GPIOALPEN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **SRAM4LPEN**：CSleep モード時 SRAM4 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SRAM4 クロックは無効です。
1：CSleep モード時の SRAM4 クロックは有効です（リセット後のデフォルト）。

ビット 28 **BKPRAMLPEN**：CSleep モード時バックアップ RAM クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時のバックアップ RAM クロックは無効です。
1：CSleep モード時のバックアップ RAM クロックは有効です（リセット後のデフォルト）。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ADC3LPEN**：CSleep モード時 ADC3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の ADC3 ペリフェラルクロックは無効です。
1：CSleep モード時の ADC3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
ADC3 のペリフェラルクロックは、ADCSEL によって選択され、adc_ker_ck 入力に供給されるカーネルクロックと、**rcc_hclk4** バスインタフェースクロックです。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMALPEN**：CSleep モード時 BDMA クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の BDMA クロックは無効です。
1：CSleep モード時の BDMA クロックは有効です（リセット後のデフォルト）。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CRCLPEN**：CSleep モード時 CRC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRC ペリフェラルクロックは無効です。
1：CSleep モード時の CRC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 18:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKLPEN**：CSleep モード時 GPIOK ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOK ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOK ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 9 **GPIOJLPEN**：CSleep モード時 GPIOJ ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOJ ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOJ ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 8 **GPIOILPEN**：CSleep モード時 GPIOI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOI ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOI ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 7 **GPIOHLPEN**：CSleep モード時 GPIOH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOH ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOH ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 6 **GPIOGLPEN**：CSleep モード時 GPIOG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOG ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOG ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 5 **GPIOFLPEN**：CSleep モード時 GPIOF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOF ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOF ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **GPIOELPEN**：CSleep モード時 GPIOE ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOE ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOE ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 **GPIODLPEN**：CSleep モード時 GPIOD ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOD ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOD ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 2 **GPIOCLPEN**：CSleep モード時 GPIOC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOC ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 1 **GPIOBLPEN**：CSleep モード時 GPIOB ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOB ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOB ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **GPIOALPEN**：CSleep モード時 GPIOA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOA ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOA ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.51 RCC APB3 スリープクロックレジスタ（RCC_APB3LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 70. RCC_APB3LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB3LPENR	0x10C	0x0000 0058
RCC_C1_APB3LPENR	0x16C	

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1LPEN	Res.	Res.	LTDCLPEN	Res.	Res.	Res.
									rw			rw			

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDG1LPEN**：CSleep モード時 WWDG1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の WWDG1 クロックは無効です。
1：CSleep モード時の WWDG1 クロックは有効です（リセット後のデフォルト）。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCLPEN**：CSleep モード時 LTDC ペリフェラルクロック有効化
ピクセルクロック（**ltdc_ker_ck**）を LTDC ブロックに供給します。
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LTDC クロックは無効です。
1：CSleep モード時に LTDC に供給された LTDC クロック（リセット後のデフォルト）。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.52 RCC APB1 ロースリープクロックレジスタ（RCC_APB1LLPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 71. RCC_APB1LLPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB1LLPENR	0x110	0xE8FF CBFF
RCC_C1_APB1LLPENR	0x170	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
USART8LPEN	USART7LPEN	DAC12LPEN	Res.	CECLPEN	Res.	Res.	Res.	I2C3LPEN	I2C2LPEN	I2C1LPEN	UART5LPEN	UART4LPEN	USART3LPEN	USART2LPEN	SPDIFRXLPEN
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3LPEN	SPI2LPEN	Res.	Res.	Res.	Res.	LPTIM1LPEN	TIM14LPEN	TIM13LPEN	TIM12LPEN	TIM7LPEN	TIM6LPEN	TIM5LPEN	TIM4LPEN	TIM3LPEN	TIM2LPEN
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **USART8LPEN**：CSleep モード時 USART8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART8 ペリフェラルクロックは無効です。
1：CSleep モード時の USART8 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART8 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 30 **USART7LPEN**：CSleep モード時 USART7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART7 ペリフェラルクロックは無効です。
1：CSleep モード時の USART7 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART7 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 29 **DAC12LPEN**：CSleep モード時 DAC1/2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DAC1/2 ペリフェラルクロックは無効です。
1：CSleep モード時の DAC1/2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **CECLPEN**：CSleep モード時 HDMI-CEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の HDMI-CEC ペリフェラルクロックは無効です。
1：CSleep モード時の HDMI-CEC ペリフェラルクロックは有効です（リセット後のデフォルト）。
HDMI-CEC のペリフェラルクロックは、CECSEL によって選択され、**cec_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3LPEN**：CSleep モード時 I2C3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C3 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C3 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 22 **I2C2LPEN**：CSleep モード時 I2C2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C2 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C2 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 21 **I2C1LPEN**：CSleep モード時 I2C1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C1 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C1 のペリフェラルクロックは、I2C123SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 20 **UART5LPEN**：CSleep モード時 UART5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART5 ペリフェラルクロックは無効です。
1：CSleep モード時の UART5 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART5 のペリフェラルクロックは、USART234578SEL によって選択され、**uart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 19 **UART4LPEN**：CSleep モード時 UART4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART4 ペリフェラルクロックは無効です。
1：CSleep モード時の UART4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART4 のペリフェラルクロックは、USART234578SEL によって選択され、**uart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 18 **USART3LPEN**：CSleep モード時 USART3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART3 ペリフェラルクロックは無効です。
1：CSleep モード時の USART3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART3 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 17 **USART2LPEN**：CSleep モード時 USART2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART2 ペリフェラルクロックは無効です。
1：CSleep モード時の USART2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART2 のペリフェラルクロックは、USART234578SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 16 **SPDIFRXLPEN**：CSleep モード時 SPDIFRX ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPDIFRX ペリフェラルクロックは無効です。
1：CSleep モード時の SPDIFRX ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPDIFRX のペリフェラルクロックは、SPDIFSEL によって選択され、**spdifrx_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 15 **SPI3LPEN**：CSleep モード時 SPI3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI3 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI3 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 14 **SPI2LPEN**：CSleep モード時 SPI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI2 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI2 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1LPEN**：CSleep モード時 LPTIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM1 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM1 のペリフェラルクロックは、LPTIM1SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 8 **TIM14LPEN**：CSleep モード時 TIM14 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM14 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM14 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 7 **TIM13LPEN**：CSleep モード時 TIM13 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM13 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM13 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 6 **TIM12LPEN**：CSleep モード時 TIM12 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM12 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM12 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 5 **TIM7LPEN**：CSleep モード時 TIM7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM7 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM7 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **TIM6LPEN**：CSleep モード時 TIM6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM6 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM6 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 **TIM5LPEN**：CSleep モード時 TIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM5 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM5 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 2 **TIM4LPEN**：CSleep モード時 TIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM4 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM4 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 1 **TIM3LPEN**：CSleep モード時 TIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM3 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM3 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **TIM2LPEN**：CSleep モード時 TIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM2 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.53 RCC APB1 ハイスリープクロックレジスタ（RCC_APB1HLPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 72. RCC_APB1HLPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB1HLPENR	0x114	0x0000 0136
RCC_C1_APB1HLPENR	0x174	

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANLPEN	Res.	Res.	MDIOSLPEN	OPAMPLPEN	Res.	SWPLPEN	CRSLPEN	Res.
							rw			rw	rw		rw	rw	

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANLPEN**：CSleep モード時 FDCAN ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の FDCAN ペリフェラルクロックは無効です。
1：CSleep モード時の FDCAN ペリフェラルクロックは有効です（リセット後のデフォルト）。
FDCAN のペリフェラルクロックは、FDCANSEL によって選択され、**fdcan_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSLPEN**：CSleep モード時 MDIOS ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の MDIOS ペリフェラルクロックは無効です。
1：CSleep モード時の MDIOS ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **OPAMPLPEN**：CSleep モード時 OPAMP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OPAMP ペリフェラルクロックは無効です。
1：CSleep モード時の OPAMP ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPLPEN**：CSleep モード時 SWPMI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SWPMI ペリフェラルクロックは無効です。
1：CSleep モード時の SWPMI ペリフェラルクロックは有効です（リセット後のデフォルト）。
SWPMI のペリフェラルクロックは、SWPSEL によって選択され、**swpmi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 1 **CRSLPEN**：CSleep モード時クロックリカバリシステムペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRS ペリフェラルクロックは無効です。
1：CSleep モード時の CRS ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.54 RCC APB2 スリープクロックレジスタ（RCC_APB2LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 73. RCC_APB2LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB2LPENR	0x118	0x31D7 3033
RCC_C1_APB2LPENR	0x178	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	HRTIMLPEN	DFSDM1LPEN	Res.	Res.	Res.	SAI3LPEN	SAI2LPEN	SAI1LPEN	Res.	SPI5LPEN	Res.	TIM17LPEN	TIM16LPEN	TIM15LPEN
		rw	rw				rw	rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4LPEN	SPI1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	USART6LPEN	USART1LPEN	Res.	Res.	TIM8LPEN	TIM1LPEN
		rw	rw							rw	rw			rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **HRTIMLPEN**：CSleep モード時 HRTIM ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の HRTIM ペリフェラルクロックは無効です。
1：CSleep モード時の HRTIM ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 28 **DFSDM1LPEN**：CSleep モード時 DFSDM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DFSDM1 ペリフェラルクロックは無効です。
1：CSleep モード時の DFSDM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
DFSDM1 ペリフェラルクロックは、SAI1SEL および DFSDM1SEL によって選択され、**Aclk** および **clk** 入力にそれぞれ供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SAI3LPEN**：CSleep モード時 SAI3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI3 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI3 のペリフェラルクロックは、SAI23SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 23 **SAI2LPEN**：CSleep モード時 SAI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI2 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI2 のペリフェラルクロックは、SAI23SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 22 **SAI1LPEN**：CSleep モード時 SAI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI1 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI1 のペリフェラルクロックは、SAI1SEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5LPEN**：CSleep モード時 SPI5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI5 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI5 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI5 のペリフェラルクロックは、SPI45SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17LPEN**：CSleep モード時 TIM17 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM17 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM17 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 17 **TIM16LPEN**：CSleep モード時 TIM16 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM16 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM16 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 16 **TIM15LPEN**：CSleep モード時 TIM15 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM15 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM15 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4LPEN**：CSleep モード時 SPI4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI4 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI4 のペリフェラルクロックは、SPI45SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 12 **SPI1LPEN**：CSleep モード時 SPI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI1 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI1 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **USART6LPEN**：CSleep モード時 USART6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART6 ペリフェラルクロックは無効です。
1：CSleep モード時の USART6 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART6 のペリフェラルクロックは、USART16SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 4 **USART1LPEN**：CSleep モード時 USART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART1 ペリフェラルクロックは無効です。
1：CSleep モード時の USART1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART1 のペリフェラルクロックは、USART16SEL によって選択され、**usart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8LPEN**：CSleep モード時 TIM8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM8 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM8 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **TIM1LPEN**：CSleep モード時 TIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM1 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.55 RCC APB4 スリープクロックレジスタ（RCC_APB4LPENR）

このレジスタには、2 つの異なるオフセットアドレスを使用してアクセスできます。

表 74. RCC_APB4LPENR アドレスオフセットとリセット値

レジスタ名	アドレスオフセット	リセット値
RCC_APB4LPENR	0x11C	0x0421 DEAA
RCC_C1_APB4LPENR	0x17C	

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4LPEN	Res.	Res.	Res.	Res.	RTCAPBLPEN
										rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFLPEN	COMP12LPEN	Res.	LPTIM5LPEN	LPTIM4LPEN	LPTIM3LPEN	LPTIM2LPEN	Res.	I2C4LPEN	Res.	SPI6LPEN	Res.	LPUART1LPEN	Res.	SYSCFGLPEN	Res.
rw	rw		rw	rw	rw	rw		rw		rw		rw		rw	

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI4LPEN**：CSleep モード時 SAI4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI4 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI4 のペリフェラルクロックは、SAI4ASEL および SAI4BSEL によって選択され、**sai_a_ker_ck** および **sai_b_ker_ck** 入力にそれぞれ供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 20:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAPBLPEN**：CSleep モード時 RTC APB クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の RTC （APB）のレジスタクロックインタフェースは無効です。
1：CSleep モード時の RTC（APB）のレジスタクロックインタフェースは有効です（リセット後のデフォルト）。

ビット 15 **VREFLPEN**：CSleep モード時 VREF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の VREF ペリフェラルクロックは無効です。
1：CSleep モード時の VREF ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 14 **COMP12LPEN**：CSleep モード時 COMP1/2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の COMP1/2 ペリフェラルクロックは無効です。
1：CSleep モード時の COMP1/2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPTIM5LPEN**：CSleep モード時 LPTIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM5 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM5 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM5 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 11 **LPTIM4LPEN**：CSleep モード時 LPTIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM4 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM4 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 10 **LPTIM3LPEN**：CSleep モード時 LPTIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM3 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM3 のペリフェラルクロックは、LPTIM345SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 9 **LPTIM2LPEN**：CSleep モード時 LPTIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM2 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM5 のペリフェラルクロックは、LPTIM2SEL によって選択され、**lptim_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4LPEN**：CSleep モード時 I2C4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C4 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C4 のペリフェラルクロックは、I2C4SEL によって選択され、**i2c_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6LPEN**：CSleep モード時 SPI6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI6 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI6 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI6 のペリフェラルクロックは、SPI6SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1LPEN**：CSleep モード時 LPUART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPUART1 ペリフェラルクロックは無効です。
1：CSleep モード時の LPUART1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPUART1 のペリフェラルクロックは、LPUART1SEL によって選択され、**lpuart_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGLPEN**：CSleep モード時 SYSCFG ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：CSleep モード時の SYSCFG ペリフェラルクロックは無効です。

1：CSleep モード時の SYSCFG ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 予約済みであり、リセット値に保持する必要があります。

参考資料

8.8 RCC レジスタマップ

表 75. RCC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RCC_CR	Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	Res.	HSECSSON	HSEBYP	HSERDY	HSEON	D2CKRDY	D1CKRDY	HSI48RDY	HSI48ON	Res.	Res.	CSIKERON	CSIRDY	CSION	Res.	HSIDIVF	HSIDIV[1:0]		HSIRDY	HSIKERON	HSION
	リセット値			0	0	0	0	0	0					0	0	0	0	0	0	0	0			0	0	0		0	0	0	0	0	1
0x004	RCC_ICSCR	Res.	CSITRIM[4:0]					CSICAL[7:0]								HSITRIM[5:0]						HSICAL[11:0]												
	リセット値		1	0	0	0	0	-	-	-	-	-	-	-	-	1	0	0	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-
0x008	RCC_CRRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL[9:0]									
	リセット値																							-	-	-	-	-	-	-	-	-	-
0x00C	予約済み	予約済み																															
0x010	RCC_CFGR	MCO2[2:0]			MCO2PRE[3:0]			MCO1[2:0]			MCO1PRE[3:0]					Res.	Res.	TIMPRE	HRTIMSEL	RTCPRE[5:0]						STOPKERWUCK	STOPWUCK	SWS[2:0]			SW[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	予約済み	予約済み																															
0x018	RCC_D1CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D1CPRE[3:0]			予約済み		D1PPRE[2:0]			HPRE[3:0]			
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x01C	RCC_D2CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D2PPRE2[2:0]			Res.	D2PPRE1[2:0]			Res.	Res.	Res.	Res.
	リセット値																						0	0	0		0	0	0				
0x020	RCC_D3CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D3PPRE[2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x024	予約済み	予約済み																															
0x028	RCC_ PLLCKSELR	Res.	Res.	Res.	Res.	Res.	Res.	DIVM3[5:0]						Res.	Res.	DIVM2[5:0]						Res.	Res.	DIVM1[5:0]						Res.	Res.	PLLSRC[1:0]	
	リセット値							1	0	0	0	0	0			1	0	0	0	0	0			1	0	0	0	0	0			0	0

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	**RCC_PLLCFGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIVR3EN	DIVQ3EN	DIVP3EN	DIVR2EN	DIVQ2EN	DIVP2EN	DIVR1EN	DIVQ1EN	DIVP1EN	Res.	Res.	Res.	Res.	PLL3RGE[1:0]		PLL3VCOSEL	PLL3FRACEN	PLL2RGE[1:0]		PLL2VCOSEL	PLL2FRACEN	PLL1RGE[1:0]		PLL1VCOSEL	PLL1FRACEN
	リセット値								1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x030	**RCC_PLL1DIVR**	Res.	DIVR1[6:0]							Res.	DIVQ1[6:0]							DIVP1[6:0]							DIVN1[8:0]								
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x034	**RCC_PLL1FRACR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN1[12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x038	**RCC_PLL2DIVR**	Res.	DIVR2[6:0]							Res.	DIVQ2[6:0]							DIVP2[6:0]							DIVN2[8:0]								
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x03C	**RCC_PLL2FRACR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN2[12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x040	**RCC_PLL3DIVR**	Res.	DIVR3[6:0]							Res.	DIVQ3[6:0]							DIVP3[6:0]							DIVN3[8:0]								
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x044	**RCC_PLL3FRACR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN3[12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x048	**予約済み**	予約済み																															
0x04C	**RCC_D1CCIPR**	Res.	Res.	CKPERSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMCSEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPISEL[1:0]		Res.	Res.	FMCSEL[1:0]	
	リセット値			0	0												0											0	0			0	0
0x050	**RCC_D2CCIP1R**	SWPSEL	Res.	FDCANSEL[1:0]		Res.	Res.	Res.	DFSDM1SEL	Res.	Res.	SPDIFSEL[1:0]		Res.	SPI45SEL[2:0]			Res.	SPI123SEL[2:0]			Res.	Res.	Res.	SAI23SEL[2:0]			Res.	Res.	Res.	SAI1SEL[2:0]		
	リセット値	0		0	0				0			0	0		0	0	0		0	0	0				0	0	0				0	0	0
0x054	**RCC_D2CCIP2R**	Res.	LPTIM1SEL[2:0]			Res.	Res.	Res.	Res.	CECSEL[1:0]		USBSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	I2C123SEL[1:0]		Res.	Res.	RNGSEL[1:0]		Res.	Res.	USART16SEL[2:0]			USART234578SEL[2:0]		
	リセット値		0	0	0					0	0	0	0							0	0			0	0			0	0	0	0	0	0

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x058	RCC_D3CCIPR	Res.	SPI6SEL[2:0]			Res.	SAI4BSEL[2:0]			SAI4ASEL[2:0]			Res.	Res.	Res.	ADCSEL[1:0]		LPTIM345SEL[2:0]			LPTIM2SEL[2:0]		I2C4SEL[1:0]		Res.	Res.	Res.	Res.	Res.	LPUART1SEL[2:0]		
	リセット値		0	0	0		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0						0	0	0
0x05C	予約済み	予約済み																															
0x060	RCC_CIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSECSSIE	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	CSIRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x064	RCC_CIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSECSSF	LSECSSF	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	CSIRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x068	RCC_CICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSECSSC	LSECSSC	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	CSIRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x06C	予約済み	予約済み																															
0x070	RCC_BDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDRST	RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		Res.	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
	リセット値																0	0						0	0		0	0	0	0	0	0	0
0x074	RCC_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSIRDY	LSION
	リセット値																															0	0
0x078	予約済み	予約済み																															
0x07C	RCC_ AHB3RSTR	CPURST															SDMMC1RST		QSPIRST		FMCRST							JPGDECRST	DMA2DRST					MDMARST
	リセット値	0															0		0		0							0	0					0
0x080	RCC_ AHB1RSTR	Res.	Res.	Res.	Res.	USB2OTGRST	Res.	USB1OTGRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1MACRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12RST	Res.	Res.	Res.	DMA2RST	DMA1RST
	リセット値					0		0										0										0				0	0

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x084	RCC_ AHB2RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2RST	Res.	Res.	RNGRST	HASHRST	CRYPTRST	Res.	Res.	Res.	CAMITFRST
	リセット値																							0			0	0	0				0
0x088	RCC_ AHB4RSTR	Res.	Res.	Res.	Res.	Res.	Res.	HSEMRST	ADC3RST	Res.	Res.	BDMARST	Res.	CRCRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKRST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	リセット値							0	0			0		0									0	0	0	0	0	0	0	0	0	0	0
0x08C	RCC_ APB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTDCRST	Res.	Res.	Res.
	リセット値																													0			
0x090	RCC_ APB1LRSTR	USART8RST	USART7RST	DAC12RST	Res.	HDMICECRST	Res.	Res.	Res.	I2C3RST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	SPDIFRXRST	SPI3RST	SPI2RST	Res.	Res.	Res.	LPTIM1RST	TIM14RST	TIM13RST	TIM12RST	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST	
	リセット値	0	0	0		0				0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0
0x094	RCC_ APB1HRSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANRST	Res.	Res.	MDIOSRST	OPAMPRST	Res.	RST	CRSRST	Res.
	リセット値																								0			0	0		0	0	
0x098	RCC_ APB2RSTR	Res.	Res.	HRTIMRST	DFSDM1RST	Res.	Res.	Res.	SAI3RST	SAI2RST	SAI1RST	Res.	SPI5RST	Res.	TIM17RST	TIM16RST	TIM15RST	Res.	Res.	SPI4RST	SPI1RST	Res.	Res.	Res.	Res.	Res.	Res.	USART6RST	USART1RST	Res.	Res.	TIM8RST	TIM1RST
	リセット値			0	0				0	0	0		0		0	0	0			0	0							0	0			0	0
0x09C	RCC_ APB4RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4RST	Res.	Res.	Res.	Res.	Res.	VREFRST	COMP12RST	Res.	LPTIM5RST	LPTIM4RST	LPTIM3RST	LPTIM2RST	Res.	I2C4RST	Res.	SPI6RST	Res.	LPUART1RST	Res.	SYSCFGRST	Res.
	リセット値											0						0	0		0	0	0	0		0		0		0		0	
0x0A0	RCC_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WW1RSC
	リセット値																																0
0x0A4	予約済み	予約済み																															

参考資料

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0A8	RCC_D3AMR	Res.		SRAM4AMEN	BKPRAMAMEN	Res.	Res.	Res.	ADC3AMEN	Res.	Res.	SAI4AMEN	Res.	CRCAMEN	Res.	Res.	RTCAMEN	VREFAMEN	COMP12AMEN	Res.	LPTIM5AMEN	LPTIM4AMEN	LPTIM3AMEN	LPTIM2AMEN	Res.	I2C4AMEN	Res.	SPI6AMEN	Res.	LPUART1AMEN	Res.	Res.	BDMAAMEN
	リセット値			0	0				0			0		0			0	0	0		0	0	0	0		0		0		0			0
0x0AC ～ 0x0CC	予約済み	予約済み																															
0x0D0	RCC_RSR	Res.	LPWRRSTF	Res.	WWDG1RSTF	Res.	IWDG1RSTF	Res.	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	D2RSTF	D1RSTF	Res.	CPURSTF	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値		0		0		0		0	1	1	1	1	1		1	0																
0x0D4	RCC_ AHB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1EN	Res.	QSPIEN	Res.	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECEN	DMA2DEN	Res.	Res.	Res.	MDMAEN
	リセット値																0		0		0							0	0				0
0x0D8	RCC_ AHB1ENR	Res.	Res.	Res.	USB2ULPIEN	USB2OTGEN	USB1ULPIEN	USB1OTGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXEN	ETH1TXEN	ETH1MACEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12EN	Res.	Res.	DMA2EN	DMA1EN
	リセット値				0	0	0	0								0	0	0										0				0	0
0x0DC	RCC_ AHB2ENR	SRAM3EN	SRAM2EN	SRAM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2EN	Res.	Res.	RNGEN	HASHEN	CRYPTEN	Res.	Res.	Res.	Res.	DCMIEN
	リセット値	0	0	0																			0			0	0	0					0
0x0E0	RCC_ AHB4ENR	Res.	Res.	Res.	BKPRAMEN	Res.	Res.	HSEMEN	ADC3EN	Res.	Res.	BDMAEN	Res.	CRCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値				0			0	0			0		0									0	0	0	0	0	0	0	0	0	0	0
0x0E4	RCC_ APB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1EN	Res.	Res.	LTDCEN	Res.	Res.	Res.
	リセット値																										0			0			
0x0E8	RCC_ APB1LENR	USART8EN	USART7EN	DAC12EN	Res.	HDMICECEN	Res.	Res.	Res.	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	SPDIFRXEN	SPI3EN	SPI2EN	Res.	Res.	Res.	Res.	LPTIM1EN	TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	リセット値	0	0	0		0				0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0EC	RCC_ APB1HENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANEN	Res.	Res.	MDIOSEN	OPAMPEN	Res.	SWPEN	CRSEN	Res.
	リセット値																								0			0	0		0	0	
0x0F0	RCC_ APB2ENR	Res.	Res.	HRTIMEN	DFSDM1EN	Res.	Res.	Res.	SAI3EN	SAI2EN	SAI1EN	Res.	SPI5EN	Res.	TIM17EN	TIM16EN	TIM15EN	Res.	Res.	SPI4EN	SPI1EN	Res.	Res.	Res.	Res.	Res.	Res.	USART6EN	USART1EN	Res.	Res.	TIM8EN	TIM1EN
	リセット値			0	0				0	0	0		0		0	0	0			0	0							0	0			0	0
0x0F4	RCC_ APB4ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4EN	Res.	Res.	Res.	Res.	RTCAPBEN	VREFEN	COMP12EN	Res.	LPTIM5EN	LPTIM4EN	LPTIM3EN	LPTIM2EN	Res.	I2C4EN	予約済み	SPI6EN	Res.	LPUART1EN	Res.	SYSCFGEN	Res.
	リセット値											0					1	0	0		0	0	0	0		0	0	0		0		0	
0x0F8	予約済み	予約済み																															
0x0FC	RCC_ AHB3LPENR	AXISRAMLPEN	ITCMLPEN	DTCM2LPEN	DTCM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1LPEN	Res.	QSPILPEN	Res.	FMCLPEN	Res.	Res.	Res.	FLASHLPEN	Res.	Res.	JPGDECLPEN	DMA2DLPEN	Res.	Res.	Res.	MDMALPEN
	リセット値	1	1	1	1												1		1		1				1			1	1				1
0x100	RCC_ AHB1LPENR	Res.	Res.	Res.	USB2ULPILPEN	USB2OTGLPEN	USB1ULPILPEN	USB1OTGLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXLPEN	ETH1TXLPEN	ETH1MACLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12LPEN	Res.	Res.	Res.	DMA2LPEN	DMA1LPEN
	リセット値				1	1	1	1								1	1	1										1				1	1
0x104	RCC_ AHB2LPENR	SRAM3LPEN	SRAM2LPEN	SRAM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2LPEN	Res.	Res.	RNGLPEN	HASHLPEN	CRYPTLPEN	Res.	Res.	Res.	CAMITFLPEN
	リセット値	1	1	1																			1			1	1	1				1
0x108	RCC_ AHB4LPENR	Res.	Res.	SRAM4LPEN	BKPRAMLPEN	Res.	Res.	Res.	ADC3LPEN	Res.	Res.	BDMALPEN	Res.	CRCLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKLPEN	GPIOJLPEN	GPIOILPEN	GPIOHLPEN	GPIOGLPEN	GPIOFLPEN	GPIOELPEN	GPIODLPEN	GPIOCLPEN	GPIOBLPEN	GPIOALPEN
	リセット値			1	1				1			1		1									1	1	1	1	1	1	1	1	1	1	1
0x10C	RCC_ APB3LPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1LPEN	Res.	Res.	LTDCLPEN	Res.	Res.	Res.
	リセット値																										1			1			

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x110	RCC_ APB1LLPENR	USART8LPEN	USART7LPEN	DAC12LPEN	Res.	CECLPEN	Res.	Res.	Res.	I2C3LPEN	I2C2LPEN	I2C1LPEN	UART5LPEN	UART4LPEN	USART3LPEN	USART2LPEN	SPDIFRXLPEN	SPI3LPEN	SPI2LPEN	Res.	Res.	Res.	Res.	LPTIM1LPEN	TIM14LPEN	TIM13LPEN	TIM12LPEN	TIM7LPEN	TIM6LPEN	TIM5LPEN	TIM4LPEN	TIM3LPEN	TIM2LPEN
	リセット値	1	1	1		1				1	1	1	1	1	1	1	1	1	1					1	1	1	1	1	1	1	1	1	1
0x114	RCC_ APB1HLPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANLPEN	Res.	Res.	MDIOSLPEN	OPAMPLPEN	Res.	SWPLPEN	CRSLPEN	Res.
	リセット値																								1			1	1		1	1	
0x118	RCC_ APB2LPENR	Res.	Res.	HRTIMLPEN	DFSDM1LPEN	Res.	Res.	Res.	SAI3LPEN	SAI2LPEN	SAI1LPEN	Res.	SPI5LPEN	Res.	TIM17LPEN	TIM16LPEN	TIM15LPEN	Res.	Res.	SPI4LPEN	SPI1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	USART6LPEN	USART1LPEN	Res.	Res.	TIM8LPEN	TIM1LPEN
	リセット値			1	1				1	1	1		1		1	1	1			1	1							1	1			1	1
0x11C	RCC_ APB4LPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4LPEN	Res.	Res.	Res.	Res.	RTCAPBLPEN	VREFLPEN	COMP12LPEN	Res.	LPTIM5LPEN	LPTIM4LPEN	LPTIM3LPEN	LPTIM2LPEN	Res.	I2C4LPEN	Res.	SPI6LPEN	Res.	LPUART1LPEN	Res.	SYSCFGLPEN	Res.
	リセット値											1					1	1	1		1	1	1	1		1		1		1		1	
0x120 ～ 0x130	予約済み	予約済み																															
0x134	RCC_C1_ AHB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1EN	Res.	QSPIEN	Res.	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECEN	DMA2DEN	Res.	Res.	Res.	MDMAEN
	リセット値																0		0		0							0	0				0
0x138	RCC_C1_ AHB1ENR	Res.	Res.	Res.	USB2ULPIEN	USB2OTGEN	USB1ULPIEN	USB1OTGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXEN	ETH1TXEN	ETH1MACEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12EN	Res.	Res.	DMA2EN	DMA1EN
	リセット値				0	0	0	0								0	0	0										0				0	0
0x13C	RCC_C1_ AHB2ENR	SRAM3EN	SRAM2EN	SRAM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2EN	Res.	Res.	RNGEN	HASHEN	CRYPTEN	Res.	Res.	Res.	DCMIEN
	リセット値	0	0	0																			0			0	0	0				0

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x140	RCC_C1_AHB4ENR	Res.	Res.	Res.	BKPRAMEN	Res.	Res.	HSEMEN	ADC3EN	Res.	Res.	DMA1EN	Res.	CRCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値				0			0	0			0		0									0	0	0	0	0	0	0	0	0	0	0
0x144	RCC_C1_APB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1EN	Res.	Res.	LTDCEN	Res.	Res.	Res.
	リセット値																										0			0			
0x148	RCC_C1_APB1LENR	USART8EN	USART7EN	DAC12EN	Res.	HDMICECEN	Res.	Res.	Res.	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	SPDIFRXEN	SPI3EN	SPI2EN	Res.	Res.	Res.	Res.	LPTIM1EN	TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	リセット値	0	0	0		0				0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0
0x14C	RCC_C1_APB1HENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANEN	Res.	Res.	MDIOSEN	OPAMPEN	Res.	SWPEN	CRSEN	Res.
	リセット値																								0			0	0		0	0	
0x150	RCC_C1_APB2ENR	Res.	Res.	HRTIMEN	DFSDM1EN	Res.	Res.	Res.	SAI3EN	SAI2EN	SAI1EN	Res.	SPI5EN	Res.	TIM17EN	TIM16EN	TIM15EN	Res.	Res.	SPI4EN	SPI1EN	Res.	Res.	Res.	Res.	Res.	Res.	USART6EN	USART1EN	Res.	Res.	TIM8EN	TIM1EN
	リセット値			0	0				0	0	0		0		0	0	0			0	0							0	0			0	0
0x154	RCC_C1_APB4ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4EN	Res.	Res.	Res.	Res.	RTCAPBEN	VREFEN	COMP12EN	Res.	LPTIM5EN	LPTIM4EN	LPTIM3EN	LPTIM2EN	Res.	I2C4EN	Res.	SPI6EN	Res.	LPUART1EN	Res.	SYSCFGEN	Res.
	リセット値											0					1	0	0		0	0	0	0		0		0		0		0	
0x158	予約済み	予約済み																															
0x15C	RCC_C1_AHB3LPENR	AXISRAMLPEN	ITCMLPEN	DTCM2LPEN	DTCM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC1LPEN	Res.	QSPILPEN	Res.	FMCLPEN	Res.	Res.	Res.	FLASHLPEN	Res.	Res.	JPGDECLPEN	DMA2DLPEN	Res.	Res.	Res.	MDMALPEN
	リセット値	1	1	1	1												1		1		1				1			1	1				1
0x160	RCC_C1_AHB1LPENR	Res.	Res.	Res.	USB2ULPILPEN	USB2OTGLPEN	USB1ULPILPEN	USB1OTGLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETH1RXLPEN	ETH1TXLPEN	ETH1MACLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADC12LPEN	Res.	Res.	DMA2LPEN	DMA1LPEN
	リセット値				1	1	1	1								1	1	1										1				1	1

表 75. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x164	RCC_C1_AHB2LPENR	SRAM3LPEN	SRAM2LPEN	SRAM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMC2LPEN	Res.	Res.	RNGLPEN	HASHLPEN	CRYPTLPEN	Res.	Res.	Res.	CAMITFLPEN
	リセット値	1	1	1																				1			1	1	1				1
0x168	RCC_C1_AHB4LPENR	Res.	Res.	SRAM4LPEN	BKPRAMLPEN	Res.	Res.	Res.	ADC3LPEN	Res.	Res.	DMA1LPEN	Res.	CRCLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKLPEN	GPIOJLPEN	GPIOILPEN	GPIOHLPEN	GPIOGLPEN	GPIOFLPEN	GPIOELPEN	GPIODLPEN	GPIOCLPEN	GPIOBLPEN	GPIOALPEN
	リセット値			1	1				1			1		1									1	1	1	1	1	1	1	1	1	1	1
0x16C	RCC_C1_APB3LPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1LPEN	Res.	Res.	LTDCLPEN	Res.	Res.	Res.
	リセット値																										1			1			
0x170	RCC_C1_APB1LLPENR	USART8LPEN	USART7LPEN	DAC12LPEN	Res.	CECLPEN	Res.	Res.	Res.	I2C3LPEN	I2C2LPEN	I2C1LPEN	UART5LPEN	UART4LPEN	USART3LPEN	USART2LPEN	SPDIFRXLPEN	SPI3LPEN	SPI2LPEN	Res.	Res.	Res.	Res.	LPTIM1LPEN	TIM14LPEN	TIM13LPEN	TIM12LPEN	TIM7LPEN	TIM6LPEN	TIM5LPEN	TIM4LPEN	TIM3LPEN	TIM2LPEN
	リセット値	1	1	1		1				1	1	1	1	1	1	1	1	1	1					1	1	1	1	1	1	1	1	1	1
0x174	RCC_C1_APB1HLPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANLPEN	Res.	Res.	MDIOSLPEN	OPAMPLPEN	Res.	SWPLPEN	CRSLPEN	Res.
	リセット値																								1			1	1		1	1	
0x178	RCC_C1_APB2LPENR	Res.	Res.	HRTIMLPEN	DFSDM1LPEN	Res.	Res.	Res.	SA3LPEN	SAI2LPEN	SAI1LPEN	Res.	SPI5LPEN	Res.	TIM17LPEN	TIM16LPEN	TIM15LPEN	Res.	Res.	SPI4LPEN	SPI1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	USART6LPEN	USART1LPEN	Res.	Res.	Res.	TIM8LPEN	TIM1LPEN
	リセット値			1	1				1	1	1		1		1	1	1			1	1							1	1				1	1
0x17C	RCC_C1_APB4LPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAI4LPEN	Res.	Res.	Res.	Res.	RTCAPBLPEN	VREFLPEN	COMP12LPEN	Res.	LPTIM5LPEN	LPTIM4LPEN	LPTIM3LPEN	LPTIM2LPEN	Res.	I2C4LPEN	Res.	SPI6LPEN	Res.	LPUART1LPEN	Res.	SYSCFGLPEN	Res.
	リセット値											1					1	1	1		1	1	1	1		1		1		1		1	
0x180 ～ 0x1FC	予約済み	予約済み																															

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

9 クロックリカバリシステム（CRS）

9.1 概要

クロックリカバリシステム（CRS）は、高精度トリミング可能な内蔵 RC オシレータ HSI48 に作用する、高度なデジタルコントローラです。選択可能な同期信号との比較に基づいて、オシレータ出力周波数チェックを行うための強力な手段を提供しています。CRS では、測定した周波数の誤差値に基づいてオシレータのトリミングを自動調整することが可能です。また、手動でトリミングすることも可能です。

CRS は、USB ペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USB ホストによって正確に 1 ms 間隔で送信される USB バス上のフレーム開始（SOF）パケット信号から同期信号を取得できます。

同期信号は、LSE オシレータ出力からも取得することができ、またユーザソフトウェアから生成することもできます。

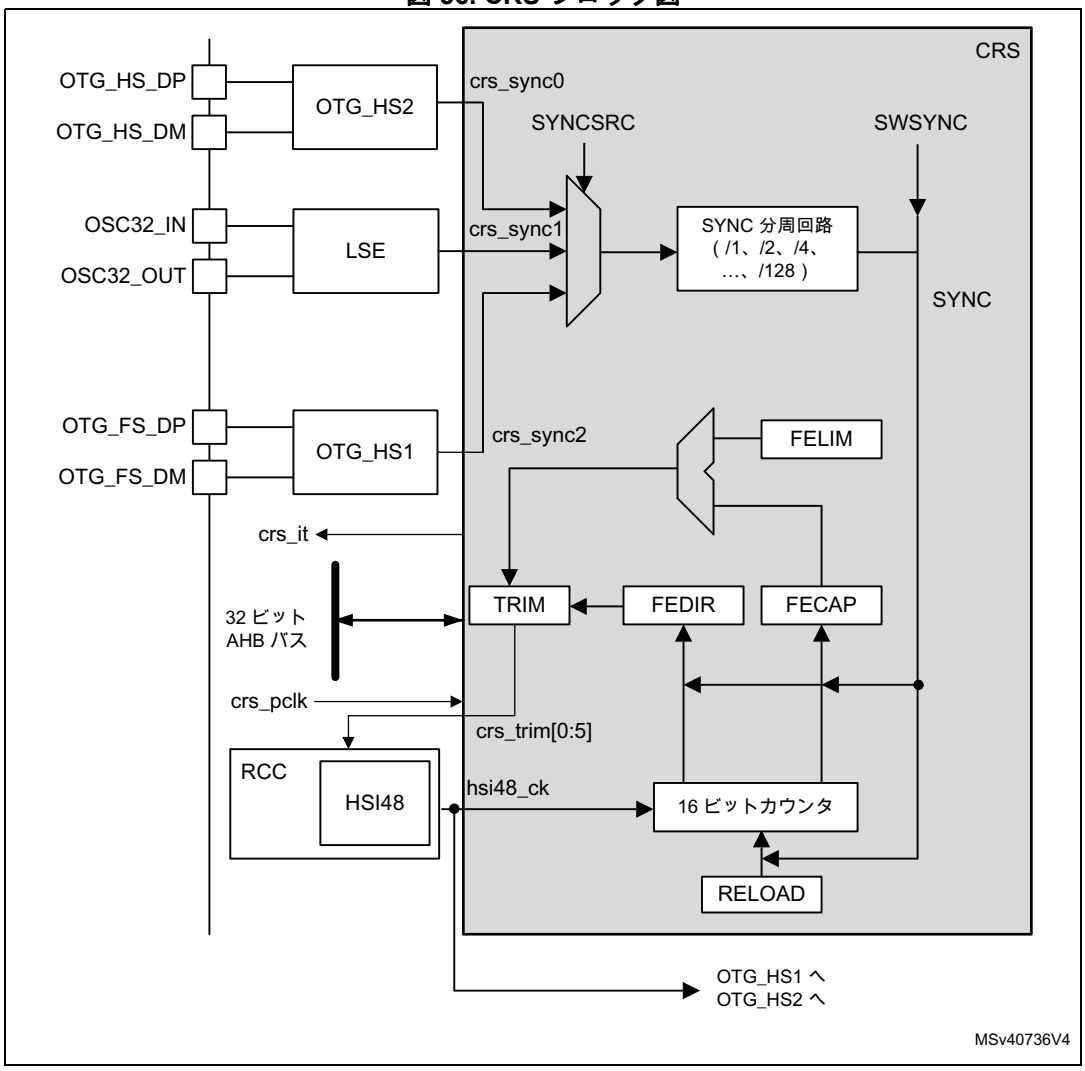
9.2 CRS の主な機能

- 次のプログラム可能なプリスケーラと極性を持つ、選択可能な同期ソース：
  - USB2 SOF パケット受信
  - LSE オシレータ出力
  - USB1 SOF パケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU 不要の自動オシレータトリミング機能
- より迅速なスタートアップのための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16 ビットの周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割り込み／イベント：
  - 期待される同期（ESYNC）
  - 同期 OK（SYNCOK）
  - 同期警告（SYNCWARN）
  - 同期またはトリミングエラー（ERR）

9.3 CRS の機能詳細

9.3.1 CRS ブロック図

図 56. CRS ブロック図



9.4 CRS 内部信号

表 76 に CRS 内部信号のリストを示します。

表 76. CRS 内部入力／出力信号

信号名	信号タイプ	説明
crs_it	デジタル出力	CRS 割り込み
crs_pclk	デジタル入力	AHB バスクロック
hsi48_ck	デジタル入力	HSI48 オシレータクロック
crs_trim[0:5]	デジタル出力	HSI48 オシレータ高精度トリミング値
crs_sync0、crs_sync1、crs_sync2	デジタル入力	SYNC 信号ソース選択（USB2、LSE、または USB1）

9.4.1 同期入力

CRS_CFGR レジスタから選択可能な CRS 同期（SYNC）ソースは、LSE クロックからの信号、USB1 SOF 信号、または USB2 SOF 信号です。このソース信号には設定可能な極性があり、適切な周波数範囲（通常 1 kHz 前後）の同期信号を取得するために、プログラム可能なバイナリプリスケーラによって分周することができます。

CRS 同期ソース設定の詳細については、*セクション 9.7.2：CRS 設定レジスタ（CRS_CFGR）*を参照してください。

CRS_CR レジスタの SWSYNC ビットをセットして、ソフトウェアで同期イベントを生成することもできます。

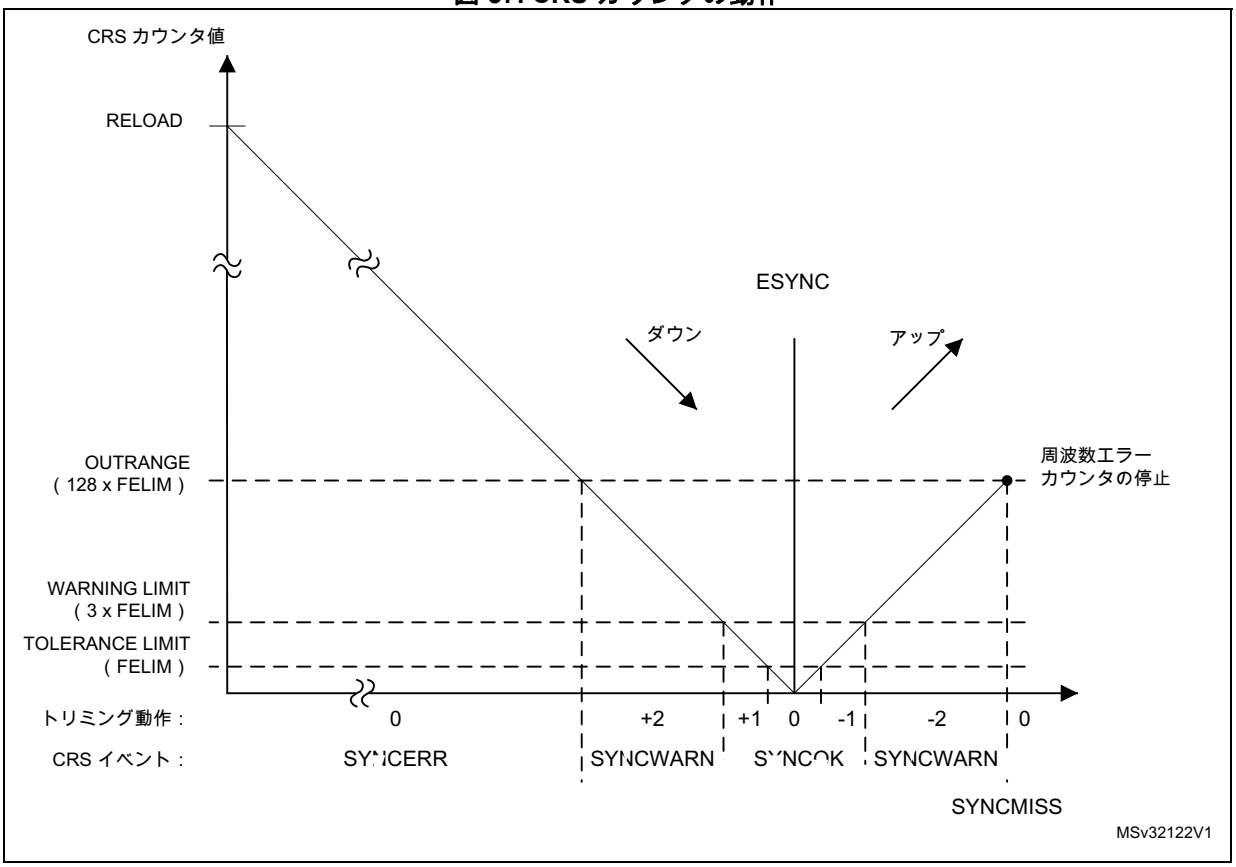
9.4.2 周波数誤差測定

この周波数誤差カウンタは、各 SYNC イベントで RELOAD 値によって再ロードされる16 ビットのダウン／アップカウンタです。ESYNC （期待される同期）イベントが生成されるゼロ値に達するまでカウントダウンします。次に、OUTRANGE 制限によって最終的に停止する（SYNC イベントが受信されない場合）までカウントアップし、SYNCMISS イベントを生成します。OUTRANGE 制限は、周波数誤差制限（CRS_CFGR レジスタの FELIM フィールド）に 128 を乗算した値として定義されます。

SYNC イベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向が FECAP （周波数誤差キャプチャ）フィールドと CRS_ISR レジスタの FEDIR （周波数誤差方向）ビットに保管されます。ダウンカウントフェーズで（ゼロ値に達する前に）SYNC イベントが検出されたときは、実際の周波数がターゲットよりも小さいことを意味します（そのため、TRIM 値をインクリメントする必要がある）。アップカウントフェーズ中に検出されると、実際の周波数は高くなります（そのため、TRIM 値をデクリメントする必要がある）。

参考資料

図 57. CRS カウンタの動作



9.4.3 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGR レジスタの FELIM フィールドに直接与えられる TOLERANCE LIMIT
- 3 * FELIM 値として定義される WARNING LIMIT
- 128 * FELIM 値として定義される OUTRANGE（エラー制限）

この比較の結果は、ステータスの表示を生成し、CRS_CR レジスタの AUTOTRIMEN ビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
  - SYNCOK ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1 回のトリミングステップによる調整で、十分適切な TRIM 値に達することができます。
  - SYNCOK ステータスが示されます。
  - AUTOTRIM モードで、1 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が警告制限を上回るか同じであり、エラー制限を下回る場合は、より強力なトリミング動作が必要であることを意味します。次の周期までに適切な TRIM 値に達することができない危険性もあります。
  - SYNCWARN ステータスが示されます。
  - AUTOTRIM モードで、2 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合は、周波数がトリミング範囲外であることを意味します。これは、SYNC 入力がクリーンではなく、SYNC パルスが一部欠落している場合に発生する可能性があります（たとえば、USB SOF が 1 つ破損している場合）。
  - SYNCERR または SYNCMISS ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。

注： ***TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM 値が誤差ちょうどに設定され、TRIMOVF ステータスが示されます。***

AUTOTRIM モード（CRS_CR レジスタにセットされた AUTOTRIMEN ビット）では、CRS_CR の TRIM フィールドがハードウェアによって調整され、読み出し専用になります。

9.4.4 CRS の初期化と設定

再ロード値

再ロード値は、ターゲット周波数とプリスケーリング後の同期ソースの周波数間の比率に従って選択します。ゼロ値で期待される同期に達するために、1 ずつ減算します。計算式は、次のとおりです。

再ロード = (f_{TARGET} / f_{SYNC}) -1

再ロードフィールドのリセット値は、48 MHz のターゲット周波数と 1 kHz の同期信号周波数（USB からの SOF 信号）に対応します。

FELIM 値

FELIM 値の選択は、HSI48 発振特性とその標準的なトリミングステップサイズに、深く関係しています。最適値は、HSI48 オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

FELIM = (f_{TARGET} / f_{SYNC}) * STEP[%] / 100% / 2

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられます。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM 値を少し上げることでトリミングのヒステリシスを増やすことができます。

FELIM フィールドのリセット値は、(f_{TARGET} / f_{SYNC}) = 48000 と標準的なトリミングステップサイズ（0.14%）に対応しています。

注意： ***再ロードおよび FELIM フィールドの誤った設定に対するハードウェアの保護はなく、不規則なトリム応答を引き起こす可能性があります。期待される動作モードでは、再ロード値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM 値よりも大きい値です（OUTRANGE 制限）。***

9.5 CRS 低電力モード

表 77. 低電力モードが CRS に与える影響

モード	説明
SLEEP	影響はありません。 CRS 割り込みによって、デバイスは SLEEP モードから復帰します。
STOP	CRS レジスタを停止します。 CRS によって、動作は STOP または STANDBY モードが終了し、HSI48 オシレータがリスタートするまで停止します。
STANDBY	

9.6 CRS 割り込み

表 78. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化 制御ビット	クリア フラッグビット
期待される同期	ESYNCF	ESYNCIE	ESYNCC
同期 OK	SYNCOKF	SYNCOKIE	SYNCOKC
同期警告	SYNCWARNF	SYNCWARNIE	SYNCWARNC
同期またはトリミングエラー （TRIMOVF、SYNCMISS、SYNCERR）	ERRF	ERRIE	ERRC

9.7 CRS レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*96 ページのセクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

9.7.1 CRS 制御レジスタ（CRS_CR）

アドレスオフセット：0x00

リセット値：0x0000 2000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
		rw	rw	rw	rw	rw	rw	rt_w	rw	rw		rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **TRIM[5:0]**：HSI48 オシレータ高精度トリミング

これらのビットにより、HSI48 オシレータにユーザプログラム可能なトリミング値が使用できます。HSI48 の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

デフォルト値は 32（トリミングインターバルの中間に相当）です。トリミングステップは、2 つの連続する TRIM ステップの間で、67 kHz 前後です。より高い TRIM 値は、より高い出力周波数に対応します。

AUTOTRIMEN ビットがセットされると、このフィールドはハードウェアによって調整され、読み出し専用になります。

ビット 7 **SWSYNC**：ソフトウェア SYNC イベントの生成

このビットは、ソフトウェア SYNC イベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ソフトウェア SYNC イベントが生成されます。

ビット 6 **AUTOTRIMEN**：自動トリミング有効

このビットは、2 つの SYNC イベント間で測定した周波数誤差に応じて、TRIM ビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIM ビットが読み出し専用になります。TRIM 値は、測定した周波数誤差値に応じて、ハードウェアで 1 度に 1 または 2 ステップずつ調整できます。詳細については、*セクション 9.4.3：周波数誤差評価および自動トリミング*を参照してください。

0：自動トリミングは無効です。ユーザは、TRIM ビットを調整できます。

1：自動トリミングは有効です。TRIM ビットは読み出し専用で、ハードウェアによって制御されます。

ビット 5 **CEN**：周波数誤差カウンタ有効

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。

0：周波数誤差カウンタは無効です。

1：周波数誤差カウンタは有効です。

このビットをセットすると、CRS_CFGR レジスタは書き込み保護され、変更できません。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCIE**：期待される SYNC 割り込み有効
0：期待される SYNC（ESYNCF）割り込みは無効です。
1：期待される SYNC（ESYNCF）割り込みは有効です。

ビット 2 **ERRIE**：同期またはトリミングエラー割り込み有効
0：同期またはトリミングエラー（ERRF）割り込みは無効です。
1：同期またはトリミングエラー（ERRF）割り込みは有効です。

ビット 1 **SYNCWARNIE**：SYNC 警告割り込み有効
0：SYNC 警告（SYNCWARNF）割り込みは無効です。
1：SYNC 警告（SYNCWARNF）割り込みは有効です。

ビット 0 **SYNCOKIE**：SYNC イベント OK 割り込み有効
0：SYNC イベント OK（SYNCOKF）割り込みは無効です。
1：SYNC イベント OK（SYNCOKF）割り込みは有効です。

9.7.2 CRS 設定レジスタ（CRS_CFGR）

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです（CEN ビットは CRS_CR でクリアされる）。このカウンタを有効にすると、このレジスタは書き込み保護されます。

アドレスオフセット：0x04

リセット値：0x2022 BB7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SYNCPOL	Res.	SYNCSRC[1:0]		Res.	SYNCDIV[2:0]			FELIM[7:0]							
rw		rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
再ロード[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SYNCPOL**：SYNC 極性選択
このビットは、SYNC 信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。
0：SYNC は立ち上がりエッジでアクティブです（デフォルト）。
1：SYNC は立ち下がりエッジでアクティブです。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **SYNCSRC[1:0]**：SYNC 信号ソース選択
これらのビットは、SYNC 信号ソースを選択するために、ソフトウェアによってセット／クリアされます。
00：SYNC 信号ソースとして USB2 SOF を選択します。
01：SYNC 信号ソースとして LSE を選択します。
10：SYNC 信号ソースとして USB1 SOF を選択します（デフォルト）。
11：予約済み

注： USB LPM（Link Power Management、リンク電源管理）の使用時にデバイスが SLEEP モードである場合、周期的 USB SOF はホストから生成されません。そのため、実行中に HSI48 を較正するために SYNC 信号が CRS に供給されることはありません。SLEEP モードからのウェイクアップ後に必要なクロック精度を確保するために、LSE を SYNC 信号として使用する必要があります。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SYNCDIV[2:0]**：SYNC 分周器

これらのビットは、SYNC 信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。

000：SYNC は分周されません（デフォルト）。

001：SYNC は 2 分周されます。

010：SYNC は 4 分周されます。

011：SYNC は 8 分周されます。

100：SYNC は 16 分周されます。

101：SYNC は 32 分周されます。

110：SYNC は 64 分周されます。

111：SYNC は 128 分周されます。

ビット 23:16 **FELIM[7:0]**：周波数誤差制限

FELIM には、CRS_ISR レジスタの FECAP[15:0] ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP 評価の詳細については、*セクション 9.4.3：周波数誤差評価および自動トリミング*を参照してください。

ビット 15:0 **再ロード[15:0]**：カウンタ再ロード値

再ロードは、SYNC イベントごとに周波数誤差カウンタにロードされる値です。

カウンタの動作の詳細については、*セクション 9.4.2：周波数誤差測定*を参照してください。

9.7.3 CRS 割り込みおよびステータスレジスタ（CRS_ISR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FECAP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
r					r	r	r					r	r	r	r

ビット 31:16 **FECAP[15:0]**：周波数誤差キャプチャ

FECAP は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタ値です。

FECAP の使用の詳細については、*セクション 9.4.3：周波数誤差評価および自動トリミング*を参照してください。

ビット 15 **FEDIR**：周波数誤差方向

FEDIR は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。

0：アップカウント方向。実際の周波数はターゲットを上回ります。

1：ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIMOVF**：トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによって TRIM 値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：トリミングエラーを信号化しません。

1：トリミングエラーを信号化します。

ビット 9 **SYNCMISS**：SYNC の欠落

このフラグは、周波数誤差カウンタが値 FELIM * 128 に達し、SYNC が検出されない場合にハードウェアによってセットされます。これは、SYNC パルスが失われているということ、または TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が高すぎる）、他の操作を行う必要があるということを意味します。この時点で、周波数誤差カウンタは停止し（次の SYNC を待つ）、CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC の欠落エラーを信号化しません。

1：SYNC の欠落エラーを信号化します。

ビット 8 **SYNCERR**：SYNC エラー

このフラグは、SYNC パルスが ESYNC イベントより前に発生し、測定した周波数誤差が FELIM * 128 以上である場合に、ハードウェアによってセットされます。これは、TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC エラーを信号化しません。

1：SYNC エラーを信号化します。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCF**：期待される SYNC フラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CR レジスタの ESYNCIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ESYNCC ビットをセットすることで、ソフトウェアによってクリアされます。

0：期待された SYNC を信号化しません。

1：期待された SYNC を信号化します。

ビット 2 **ERRF**：エラーフラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、および SYNCERR ビットの論理和です。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、および SYNCERR ビットをクリアします。

0：同期またはトリミングエラーを信号化しません。

1：同期またはトリミングエラーを信号化します。

ビット 1 **SYNCWARNF**：SYNC 警告フラグ

このフラグは、測定した周波数誤差が FELIM * 3 以上で FELIM * 128 未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM 値を複数のステップで調整する必要があります。CRS_CR レジスタの SYNCWARNIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCWARNC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC 警告を信号化しません。

1：SYNC 警告を信号化します。

ビット 0 **SYNCOKF**：SYNC イベント OK フラグ

このフラグは、測定した周波数誤差が FELIM * 3 未満である場合に、ハードウェアによってセットされます。これは、TRIM 値の補正が必要であること、または周波数誤差の補正は 1 つのトリミングステップで十分であることを意味します。CRS_CR レジスタの SYNCOKIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCOKC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC イベント OK フラグを信号化しません。

1：SYNC イベント OK フラグを信号化します。

9.7.4 CRS 割り込みフラグクリアレジスタ（CRS_ICR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCC**：期待 SYNC クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの ESYNCF フラグをクリアします。

ビット 2 **ERRC**：エラークリアフラグ

このビットに 1 を書き込むと、TRIMOVF、SYNCMISS、および SYNCERR ビットがクリアされ、結果として CRS_ISR レジスタの ERRF フラグもクリアされます。

ビット 1 **SYNCWARNC**：SYNC 警告クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCWARNF フラグをクリアします。

ビット 0 **SYNCOKC**：SYNC イベント OK クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCOKF フラグをクリアします。

9.7.5 CRS レジスタマップ

表 79. CRS レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
	リセット値																			1	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	CRS_CFGR	SYNCPOL	Res.	SYNC SRC [1:0]		Res.	SYNC DIV [2:0]			FELIM[7:0]								再ロード[15:0]															
	リセット値	0		1	0		0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1
0x08	CRS_ISR	FECAP[15:0]																FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0					0	0	0	0
0x0C	CRS_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

10 ハードウェアセマフォ（HSEM）

10.1 ハードウェアセマフォの概要

ハードウェアセマフォのブロックは、セマフォに基づいて 32 （32 ビット）レジスタを供給します。

セマフォは、異なるコア間で実行される個々のプロセスが必ず互いに同期されるように使用されます。HSEM は、微細操作でセマフォをロックする非ブロッキングメカニズムを供給します。次の機能が供給されます。

- セマフォのロック。次の 2 つの方法で実行できます。
  - 2 ステップロック：セマフォに CoreID と ProcessID、続いて読み出しチェックを書き込みます。
  - 1 ステップロック：セマフォに CoreID とセマフォを読み出します。
- セマフォ開放時の割り込み生成
  - 各セマフォは、割り込みラインの 1 つに割り込みを生成できます。
- セマフォクリア保護
  - セマフォは、CoreID と ProcessID が一致する場合にのみクリアされます。
- CoreID ごとのグローバルセマフォのクリア

10.2 ハードウェアセマフォの主な機能

HSEM には以下の機能があります。

- 32 （32 ビット）セマフォ
- 8 ビットの ProcessID
- 4 ビットの CoreID
- 2 本の割り込みライン
- ロック表示

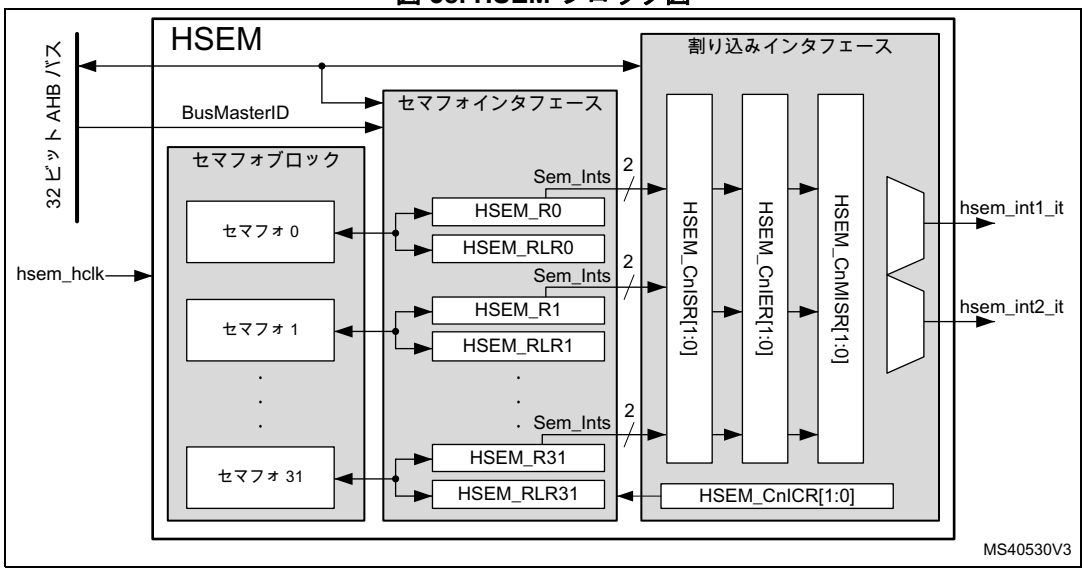
10.3 HSEM の機能詳細

10.3.1 HSEM ブロック図

図 58 に示すように、HSEM は 3 つのサブブロックをベースにしています。

- セマフォのステータスと ID を含むセマフォブロック
- HSEM_R および HSEM_RLR レジスタを介してセマフォに AHB アクセスを供給するセマフォインタフェースブロック
- HSEM_CnISR、HSEM_CnIER、HSEM_CnMISR、および HSEM_CnICR レジスタを介して割り込みの制御を行う割り込みインタフェースブロック

図 58. HSEM ブロック図



10.3.2 HSEM 内部信号

表 80. HSEM 内部入力／出力信号

信号名	信号タイプ	説明
hsem_hclk	デジタル入力	AHB クロック
hsem_int1_it	デジタル出力	割り込み 1 ライン
hsem_int2_it	デジタル出力	割り込み 2 ライン

10.3.3 HSEM ロックの手順

ロックの手順は 2 つあります。

- 2 ステップ（書き込み）のロック
- 1 ステップ（読み出し）のロック

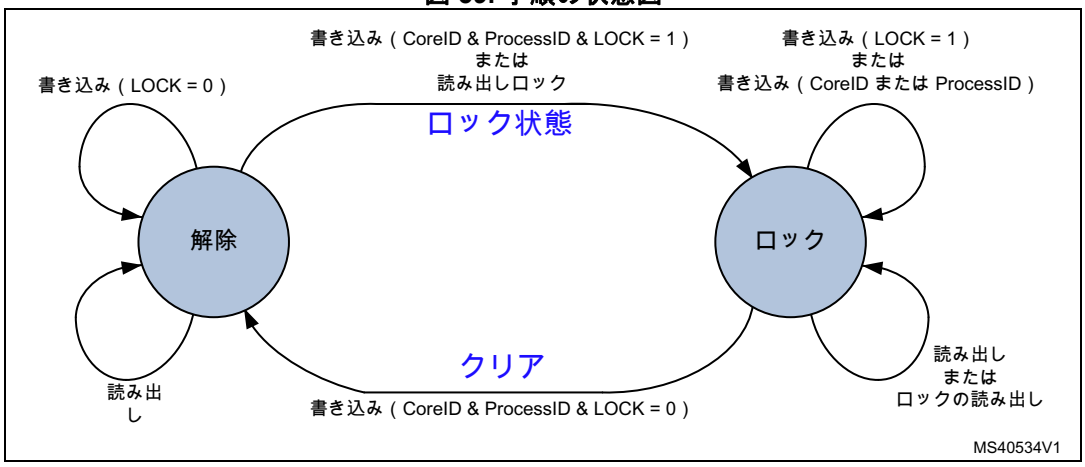
セマフォは、ロックビットが「0」のときに解除されます。この場合は、CoreID および ProcessID も「0」です。セマフォはロックビットが「1」の場合にロックされ、CoreID にはどの AHB バスマスタによってロックされたかが示されます。ProcessID には、当該 AHB バスマスタのどのプロセスがセマフォをロックしたかが示されます。

セマフォを書き込みロックすると、CoreID はマスタ ID から取得され、ProcessID は書き込みデータから取得されます。セマフォを読み出しロックすると、CoreID はマスタ ID から取得され、ProcessID はゼロになります。1 ステップ（読み出し）のロックで使用可能な ProcessID はありません。

CoreID は AHB バスマスタ ID から取得されます。ProcessID は当該 AHB バスマスタのファームウェアから書き込まれます。AHB バスマスタのプロセスには、それぞれ固有の ProcessID がなければなりません。ProcessID は 2 ステップロックの手順でのみ使用できます。

2 つの手順（1 ステップおよび 2 ステップ）は同時に使用できます。

図 59. 手順の状態図



2 ステップ（書き込み）のロック手順

2 ステップロックの手順は、セマフォをロックする書き込み操作と、その後ロックが正常に行われたかどうかをチェックする読み出し操作から成り、HSEM_Rn レジスタから実行されます。

- ProcessID と CoreID を持ち、LOCK ビット = 1 のセマフォを書き込みます。
（書き込み時にセマフォが開放されている場合はロックが設定されます。）
- セマフォを読み戻します。
（ProcessID と CoreID が一致する場合、FW ではロックステータスをチェックしてロックされていることを確認します。）
- そうでない場合、再試行します（セマフォは、別の AHB バスマスタまたはプロセスでロックされます）。

セマフォは、開放されている場合にのみロックできます。

セマフォは、ProcessID が「0」の場合にロックできます。

ロックビット = 1 でのロックされたセマフォへの連続書き込みは無視されます。

1 ステップ（読み出し）のロック手順

1 ステップの手順は、HSEM_RLRn レジスタからシングルステップでセマフォをロックしてチェックする読み出しから成ります。

- CoreID を持つロックセマフォの読み出し
- 読み出し CoreID が一致し、ProcessID = 0 の場合は、ロックが設定されます。
（CoreID が一致し、ProcessID が「0」でない場合は、同じ CoreID の別のプロセスが 2 ステップ（書き込み）手順でセマフォをロックしたことを意味します。）
- そうでない場合、再試行します（セマフォは、別の AHB バスマスタまたはプロセスでロックされます）。

セマフォは、開放されている場合にのみロックできます。開放されたセマフォを読み出しロックすると、ProcessID は「0」になります。ロックされたセマフォを読み出しロックすると、ロックに使用された CoreID と ProcessID が返されます。セマフォをロックした最初の読み出しロックを含むすべての読み出しロックから、セマフォをロックする、またはロックした CoreID が返されます。

同じ AHB バスマスタの複数のプロセスが 1 ステップの手順を使用する場合、同じセマフォを使用するすべてのプロセスで同じステータスが読み出されます。セマフォをロックするプロセスが 1 つのみである場合、その AHB バスマスタの各プロセスでは CoreID によってそれ自体でロックされたセマフォを読み出します。

10.3.4 HSEM 書き込み/読み出し/読み出しロックレジスタアドレス

各セマフォには、0x80 の 2 つの個別のバンクに、2 つの AHB レジスタアドレスが供給されます。

最初のレジスタアドレスバンクでは、HSEM_R レジスタを介してセマフォを書き込んだり（ロック／クリア）読み出したりすることができます。

2 つ目のレジスタアドレスバンクでは、HSEM_RLR レジスタを介してセマフォを読み出す（ロック）ことができます。

10.3.5 HSEM クリアの手順

セマフォのクリアは保護されたプロセスで、AHB バスマスタやセマフォロック権限を持たないプロセスによる予期しないクリアを防ぎます。セマフォクリアの手順は、セマフォへの対応する CoreID と ProcessID およびロックビット = 0 の書き込みから成ります。クリアすると、セマフォロックビット、CoreID、および ProcessID はすべて「0」になります。

クリアすると、イベントを通知するために割り込みが生成されることがあります。このために、セマフォの割り込みを有効にすることができます。

クリアの手順は、セマフォの HSEM_R レジスタへの書き込みから成ります。

- ProcessID と CoreID、ロックビット = 0 のセマフォを書き込みます。
- ProcessID と CoreID が一致する場合、セマフォが開放され、有効化すると割り込みが生成されることがあります。
- そうでない場合、書き込みは無視され、セマフォはロックされたまま、割り込みは生成されません（セマフォは別の AHB バスマスタまたはプロセスでロックされます）。

同じ AHB バスマスタの複数のプロセスが 1 ステップのロック手順（ProcessID = 0）を使用する場合、同じセマフォを使用するすべてのプロセスでセマフォがクリアされ、これは当該 AHB バスマスタのその他のプロセスについても同様です。

10.3.6 HSEM CoreID セマフォのクリア

AHB バスマスタによってロックされたすべてのセマフォを、HSEM_CR レジスタを使用して一度にすべてクリアすることができます。

AHB バスマスタによってロックされたすべてのセマフォをクリアする手順は次のとおりです。

- CoreID と正しい KEY 値を書き込みます。一致する CoreID を持つすべてのロックされたセマフォがクリア（開放）され、有効化すると割り込みが生成されることがあります。

この手順は AHB バスマスタが不正に動作する場合に使用することができます。たとえば不正に動作する CoreID を正しい KEY 値で HSEM_CR レジスタに書き込むことで、別の AHB バスマスタでロックされたセマフォを開放できる場合などです。これにより、一致する CoreID を持つすべてのロックされたセマフォがクリアされます。

開放されたセマフォに、割り込みが生成されることがあります。このために、HSEM_CnIER レジスタでセマフォの割り込みを有効にすることができます。

10.3.7 HSEM 割り込み

hsem_int1_it （割り込み 1 用）および hsem_int2_it （割り込み 2 用）の 2 本の割り込みラインがあり、32 個のセマフォのそれぞれから割り込みを生成できます。

これらの 2 本の割り込みラインから、それぞれ以下の機能を使用できます。

- セマフォごとの割り込みイネーブル
- セマフォごとの割り込みクリア
- セマフォごとの割り込みステータス
- セマフォごとのマスク済み割り込みステータス

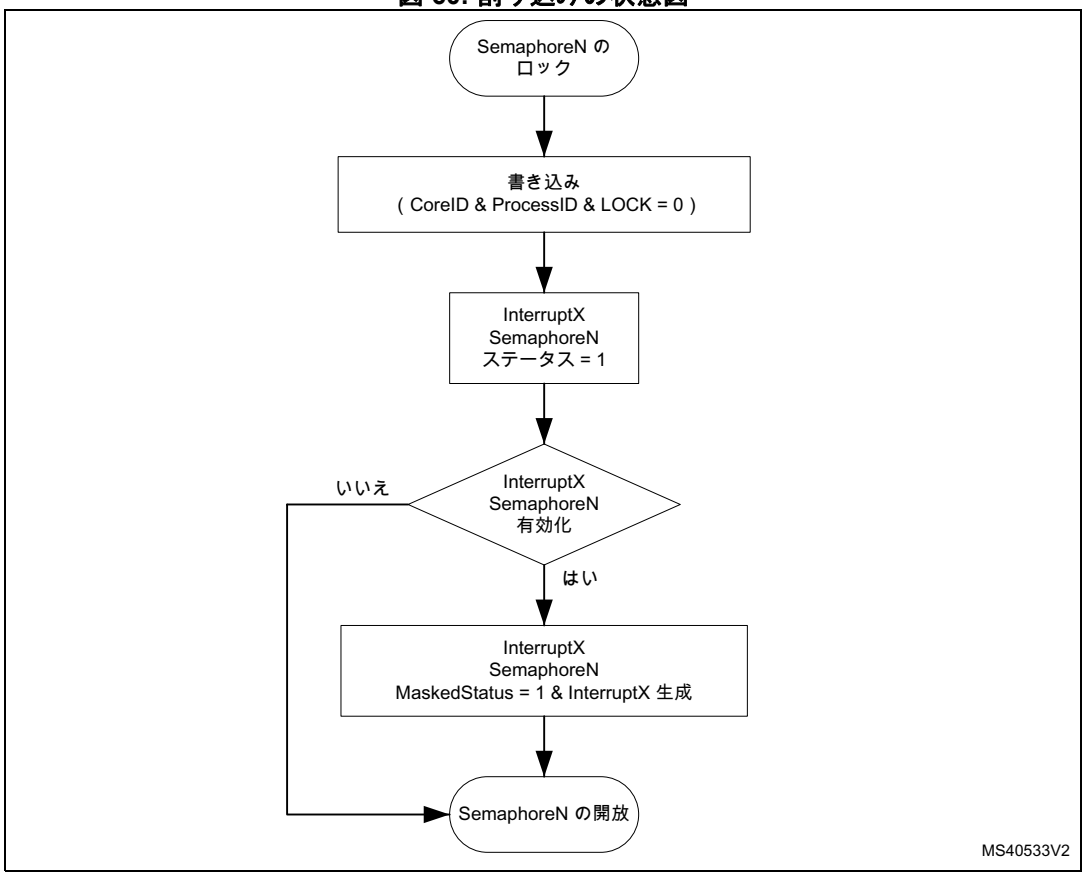
割り込みイネーブル（HSEM_CnIER）により、割り込みラインに影響を与えるセマフォを有効にすることができます。無効化された（マスク済み）セマフォの割り込みが当該セマフォにマスクされた割り込みステータスをセットすることはなく、また割り込みラインに割り込みを生成することもありません。

割り込みクリア（HSEM_CnICR）によって、割り込みラインの関連するセマフォの割り込みステータスやマスク済み割り込みステータスをクリアします。

割り込みステータス（HSEM_CnISR）には、有効化する前に割り込みラインのセマフォの割り込みステータスが反映されます。

マスク済み割り込みステータス（HSEM_CnMISR）には、割り込みライン上の有効化されたセマフォの割り込みの、セマフォの割り込みステータスのみが反映されます。割り込みラインをクリアするには、有効化されたセマフォのマスク済み割り込みステータスをすべてクリアする必要があります。

図 60. 割り込みの状態図



ここでは、セマフォが開放されたときに割り込みを取得する手順について説明します。

セマフォ N のロックを試みます。

- セマフォロックを取得できた場合は、割り込みは不要です。
- セマフォロックが失敗した場合は、次を実行してください。
  HSEM_CnICR の割り込みライン hsem_intn_it の、ペンディング中のセマフォ N 割り込みステータスをクリアします。
  セマフォ N のロックを再度試みます。
  - セマフォロックを取得できた場合は、割り込みは不要です（セマフォは最初にロックしようとしてから、セマフォの割り込み状態をクリアするまでに開放されています。）。
  - セマフォロックが失敗した場合は、HSEM_CnIER の割り込みライン hsem_intnn_it のセマフォ N 割り込みを有効化します。

セマフォ N の開放時の割り込みで、セマフォ N のロックを試みます。

- セマフォロックを取得できた場合は、次を実行してください。
  HSEM_CnIER の割り込みライン hsem_intn_it のセマフォ N 割り込みを無効化し、HSEM_CnICR の割り込みライン hsem_intn_it の、ペンディング中のセマフォ N 割り込みステータスをクリアします。
- セマフォ N ロックが失敗した場合は、次を実行してください。
  HSEM_CnICR の割り込みライン hsem_intn_it の、ペンディング中のセマフォ N 割り込みステータスをクリアします。
  セマフォ N のロックを再度試みます。
  - セマフォロックを取得できた場合は（セマフォは最初の試行時に開放されており、セマフォ割り込みステータスはクリアです）、次を実行してください。
    HSEM_CnIER の割り込みライン hsem_intn_it の、セマフォ割り込みを無効にします。
  - セマフォロックが失敗した場合は、セマフォの開放時の割り込みを待ちます。

注： ***割り込みによってセマフォがロックされることはありません。割り込み後、AHB バスマスタまたはプロセスのどちらかで、セマフォをロックするためのロック手順を実行しなければならない。***

セマフォの開放時の割り込みで通知される AHB バスマスタを、複数持つことができます。AHB バスマスタはそれぞれ割り込みを取得し、最初に反応したバスマスタによってセマフォはロックされます。

10.3.8 AHB バスマスタ ID 照合

セマフォをロック／アンロックする際、HSEM では許可された AHB バスマスタ ID のみを使用できます。

- セマフォの HSEM_Rn レジスタへの AHB バスマスタの 2 ステップロック書き込みアクセスが、有効なバスマスタ ID に対してチェックされます。
  - 許可されていない AHB バスマスタ ID からのアクセスは破棄され、セマフォがロックされることはありません。
- セマフォの HSEM_Rn レジスタへの AHB バスマスタの 1 ステップロック読み出しアクセスが、有効なバスマスタ ID に対してチェックされます。
  - HSEM_RLRn から読み出された許可されていない AHB バスマスタ ID は、セマフォのステータスに応じて、次の読み出しデータを返します。
    - セマフォが開放されている場合、すべてに「0」を返します。
    - セマフォがすでにロックされている場合、HSEM_RLRn データを返します。

- CoreID セマフォをクリアする HSEM_CR レジスタへの書き込みアクセスが、有効なバスマスタ ID に対してチェックされます。有効なバスマスタ ID のみが HSEM_CR レジスタに書き込んだり任意の CoreID セマフォをクリアできます。
  - 許可されていない AHB バスマスタ ID からのアクセスは破棄され、CoreID セマフォがクリアされることはありません。

表 81 に、バスマスタ／CPU と COREID の関係について示します。

表 81. 許可された AHB バスマスタ ID

バスマスタ 0 （CPU1）	バスマスタ 1 （CPU2）
COREID = 3	COREID = 1

注 : 許可されていない AHB バスマスタ ID からほかのレジスタへのアクセスが許可されています。

10.4 HSEM レジスタ

レジスタにはワードフォーマットでアクセスします。バイトおよびハーフワードによるアクセスは無視され、セマフォに影響することはありません。バイトおよびハーフワードによる読み出しアクセスは、常に 0 を返します。バイトおよびハーフワードによるアクセスがバスエラーを生成することはありません。

10.4.1 HSEM レジスタ（HSEM_R0 - HSEM_R31）

アドレスオフセット：0x000 + N x 4 （ここでは、N = セマフォ番号 0 - 31）

リセット値：0x0000 0000

2 ステップの書き込みロックおよび読み戻しを実行するには、HSEM_R0 - HSEM_R31 を使用します。許可された AHB バスマスタ ID を持つ書き込みアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ書き込みアクセスは破棄されます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	COREID				PROCID							
				rw	rw	rw	rw	rw							

ビット 31 **LOCK**：ロック表示

このビットの書き込み／読み出しは、ファームウェアでのみ行うことができます。

0：書き込み時にセマフォを開放し（COREID と PROCID が一致する場合のみ）、読み出し時にセマフォは開放されています。

1：書き込み時にセマフォのロックを試行し、読み出し時にセマフォはロックされています。

ビット 30:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 COREID：セマフォ CoreID

セマフォが開放されている場合で、同時に LOCK ビットに 1 が書き込まれている場合に、ファームウェアによって書き込まれます。COREID は、セマフォを書き込む AHB バスマスタのバス ID と一致する場合にのみ書き込まれます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれ、AHB バスマスタ ID が COREID と一致する場合）、CoreID は 0 にクリアされます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれ、AHB バスマスタ ID が COREID と一致しない場合）、CoreID はその影響を受けません。

LOCK ビットがすでに「1」の場合（セマフォがロックされている場合）に書き込むと、CoreID は影響を受けません。

読み出しにより、格納された CoreID 値が返されます。

ビット 7:0 **PROCID**：セマフォ ProcessID

セマフォが開放されている場合で、LOCK ビットに 1 が書き込まれている場合に、ファームウェアによって書き込まれます。ProcessID が書き込みデータにセットされます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれている場合）、ProcessID は 0 にクリアされます。

LOCK ビットがすでに「1」の場合（セマフォがロックされている場合）に書き込むと、ProcessID は影響を受けません。

読み出しにより、プログラムされた ProcessID 値が返されます。

10.4.2 HSEM 読み出しロックレジスタ（HSEM_RLR0 - HSEM_RLR31）

アドレスオフセット：0x080 + N x 4 （ここでは、N = セマフォ番号 0 - 31）

リセット値：0x0000 0000

HSEM_R0 - HSEM_R31 と同じ物理ビットにアクセスします。1 ステップの読み出しロックを実行するには、HSEM_RLR0 - HSEM_RLR31 を使用します。許可された AHB バスマスタ ID を持つ読み出しアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ読み出しアクセスは破棄されます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	COREID				PROCID							
				r	r	r	r	r							

ビット 31 **LOCK**：ロック表示

このビットは、このアドレスでファームウェアによってのみ読み出されます。有効なバスマスタ ID による読み出しは、常に 1 を返します。

セマフォが開放されている場合にファームウェアが読み出しを実行すると、ハードウェアによってセマフォがロックにセットされます。

セマフォがロックされている場合にファームウェアが読み出しを実行しても、LOCK ビットへの影響はありません。

0：セマフォは開放されます。

1：セマフォはロックされます。

ビット 30:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **COREID**：セマフォ CoreID

このフィールドは、このアドレスでファームウェアによってのみ読み出されます。

読み出し時、セマフォが開放されている場合、ハードウェアではセマフォを読み出す ID を AHB バスマスタ ID に、CoreID をセットします。セマフォをロックしている AHB バスマスタの CoreID が読み出されます。

読み出し時、セマフォがロックされている場合は、セマフォをロックした AHB バスマスタの CoreID が返されます。

ビット 7:0 **PROCID**：セマフォ ProcessID

このフィールドは、このアドレスでファームウェアによってのみ読み出されます。

読み出し時、セマフォが開放されている場合は「0」が返されます。

読み出し時、セマフォがロックされている場合は、セマフォをロックしたプロセスの ProcessID が返されます。

10.4.3 HSEM 割り込み有効レジスタ（HSEM_CnIER）

アドレスオフセット：0x100 + (N - 1) x 0x10 （ここで N = 割り込み番号 1 - 2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ISE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ISEn**：割り込み（N）セマフォ n イネーブルビット

このビットの書き込み／読み出しは、ファームウェアで行うことができます。

0：セマフォ n の割り込み（N）生成は無効（マスク済み）です。

1：セマフォ n の割り込み（N）生成は有効（未マスク）です。

10.4.4 HSEM 割り込みクリアレジスタ（HSEM_CnICR）

アドレスオフセット：0x104 + (N - 1) x 0x10 （ここで N = 割り込み番号 1 - 2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ISC[31:16]															
wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISC[15:0]															
wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0	wr0

ビット 31:0 **ISCn**：割り込み（N）セマフォ n クリアビット

このビットはファームウェアによって書き込まれ、常に 0 を読み出します。

0：割り込み（N）のセマフォ n ステータスおよびマスクされたステータスは影響を受けません。

1：割り込み（N）のセマフォ n ステータスおよびマスクされたステータスはクリアされます。

10.4.5 HSEM 割り込みステータスレジスタ（HSEM_CnISR）

アドレスオフセット：0x108 + (N - 1) x 0x10 （ここで N = 割り込み番号 1 - 2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ISF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **ISFn**：有効化（マスク）する前の割り込み（N）のセマフォ n ステータスビット
このビットはハードウェアでセットされ、ファームウェアによってのみ読み出されます。
ビットは、対応する HSEM_CnICR ビットを書き込むことで、ファームウェアによってクリアされます。
0：割り込み（N）のセマフォ n ステータス、ペンディング状態の割り込みなし
1：割り込み（N）のセマフォ n ステータス、ペンディング状態の割り込みあり

10.4.6 HSEM マスク済み割り込みステータスレジスタ（HSEM_CnMISR）

アドレスオフセット：0x10C + (N - 1) x 0x10 （ここで N = 割り込み番号 1 - 2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MISF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **MISFn**：有効化（マスク）後のマスクされた割り込み（N）のセマフォ n ステータスビット
このビットはハードウェアでセットされ、ファームウェアによってのみ読み出されます。
ビットは、対応する HSEM_CnICR ビットを書き込むことで、ファームウェアによってクリアされます。
0：ペンディング状態ではないマスク後の割り込み（N）のセマフォ n ステータス
1：ペンディング状態のマスク後の割り込み（N）のセマフォ n ステータス

10.4.7 HSEM クリアレジスタ（HSEM_CR）

アドレスオフセット：0x140

リセット値：0x0000 0000

許可された AHB バスマスタ ID を持つ書き込みアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ書き込みアクセスは破棄されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY															
wr0															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	COREID				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				wr0											

ビット 31:16 **KEY**：セマフォクリアキー

このビットの書き込みは、ファームウェアで行うことができます。常に 0 として読み出されます。

キー値が HSEM_KEYR.KEY と一致しない場合、セマフォへの影響はありません。

キー値が HSEM_KEYR.KEY と一致する場合、COREID と一致するすべてのセマフォが開放状態までクリアされます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **COREID**：クリアするセマフォの CoreID

このフィールドはファームウェアによって書き込まれ、常に 0 を読み出します。

HSEM_CR 書き込み時にセマフォをクリアする CoreID を示します。

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

10.4.8 HSEM 割り込みクリアレジスタ（HSEM_KEYR）

アドレスオフセット：0x144

リセット値：0x0000 0000

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

ビット 31:16 **KEY**：セマフォクリアキー

このビットの書き込み／読み出しは、ファームウェアでのみ行うことができます。

セマフォをクリアする際に合わせるキー値。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

10.4.9 HSEM レジスタマップ

表 82. HSEM レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**HSEM_R0**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID [3:0]				PROCID[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	**HSEM_R1**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID [3:0]				PROCID[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮																																	
0x07C	**HSEM_R31**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID[3:0]				PROCID[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x080	**HSEM_RLR0**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID [3:0]				PROCID							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	**HSEM_RLR1**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID [3:0]				PROCID[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮																																	
0x0FC	**HSEM_RLR31**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COREID [3:0]				PROCID[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	**HSEM_C1IER**	ISE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x104	**HSEM_C1ICR**	ISC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x108	**HSEM_C1ISR**	ISF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10C	**HSEM_C1MISR**	MISF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x110	**HSEM_C2IER**	ISE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x114	**HSEM_C2ICR**	ISC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x118	**HSEM_C2ISR**	ISF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 82. HSEM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x11C	**HSEM_C2MISR**	MISF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x140	**HSEM_CR**	KEY																Res.	Res.	Res.	Res.	COREID [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x144	**HSEM_KEYR**	KEY																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

参考資料

11 汎用 I/O（GPIO）

11.1 概要

各汎用 I/O ポートにはそれぞれ 4 つの 32 ビット設定レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）、2 つの 32 ビットデータレジスタ（GPIOx_IDR、GPIOx_ODR）、および 1 つの 32 ビットセット／リセットレジスタ（GPIOx_BSRR）があります。さらに、すべての GPIO には、それぞれ 32 ビットロックレジスタ（GPIOx_LCKR）と 2 つの 32 ビットオルタネート機能選択レジスタ（GPIOx_AFRH、GPIOx_AFRL）があります。

11.2 GPIO の主な機能

- 出力状態：プッシュプルまたはオープンドレイン + プルアップ／プルダウン
- 出力データレジスタ（GPIOx_ODR）またはペリフェラル（オルタネート機能出力）からの出力データ
- 各 I/O のスピード選択
- 入力状態：フローティング、プルアップ／プルダウン、アナログ
- 入力データレジスタ（GPIOx_IDR）またはペリフェラル（オルタネート機能入力）への入力データ
- GPIOx_ODR へのビット単位の書き込みアクセス用のビットセット／リセットレジスタ（GPIOx_BSRR）
- I/O ポート設定を固定するロック機構（GPIOx_LCKR）
- アナログ機能
- オルタネート機能選択レジスタ
- 2 クロックサイクルごとに変化可能な高速トグル
- 柔軟性の高いピンの多重化により、I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用可能

11.3 GPIO の機能説明

各 I/O ポートの特定のハードウェア特性については、データシートに記載されています。汎用 I/O（GPIO）ポートの各ポートビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各 I/O ポートビットは自由にプログラム可能ですが、I/O ポートレジスタには 32 ビットワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRR および GPIOx_BRR レジスタを使用すると、任意の GPIOx_ODR レジスタに不可分な読み出し／変更アクセスを行うことができます。これにより、読み出しと変更アクセスの間に IRQ が発生するリスクを回避できます。

図 61 および *図 62* は、標準 I/O ポートビットと 5 V トレラント I/O ポートビットの基本構成をそれぞれ示します。*表 83* には、実行可能なポートビットの設定を示します。

図 61. I/O ポートビットの基本構成

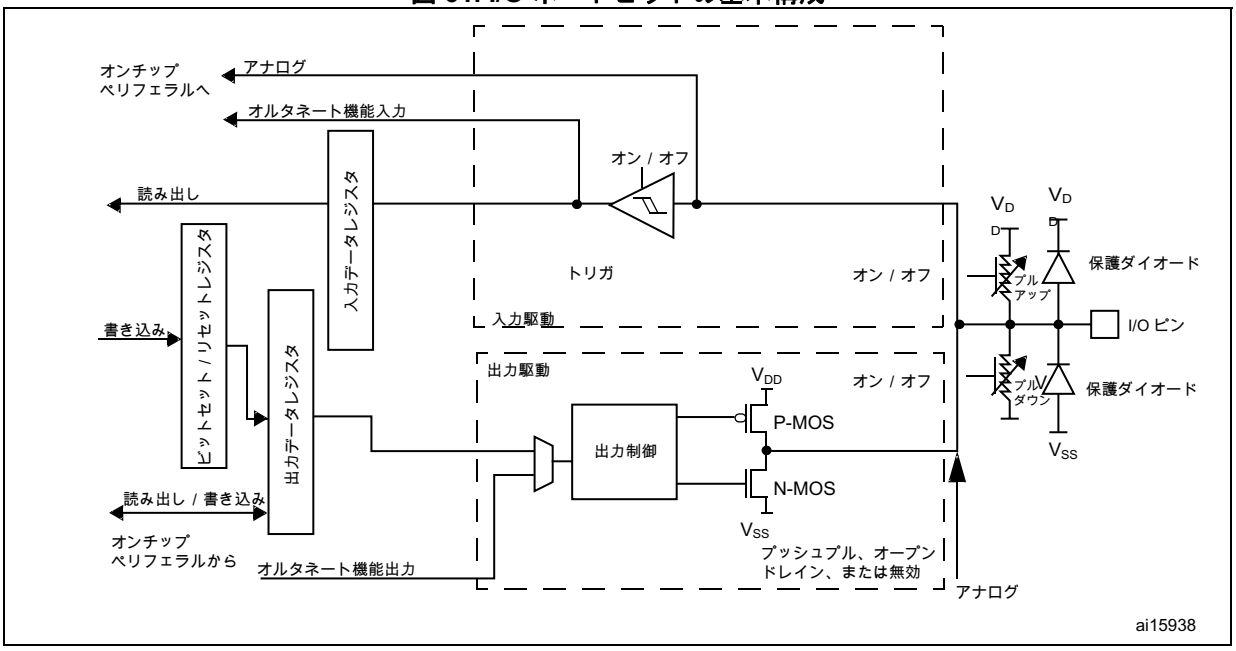
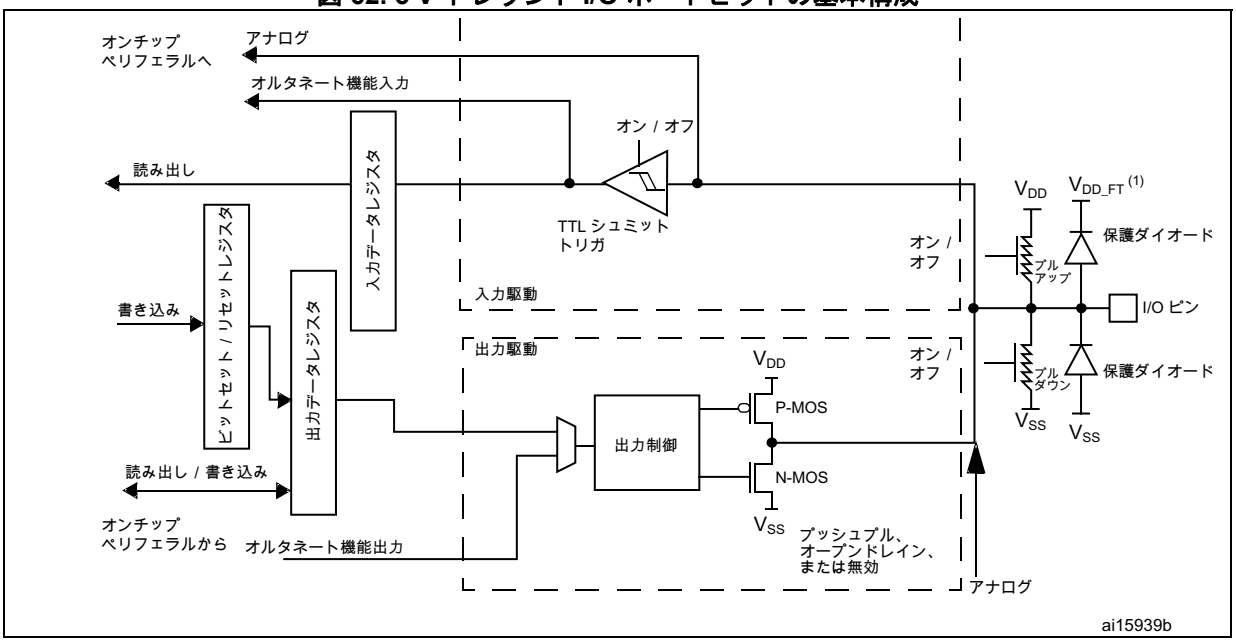


図 62. 5 V トレラント I/O ポートビットの基本構成



1. V_{DD_FT} は、5 V トレラント I/O に固有の電位であり、V_{DD} とは異なります。

表 83. ポートビット設定表(1)

MODE(i) [1:0]	OTYPER(i)	OSPEED(i) [1:0]		PUPD(i) [1:0]		I/O 設定	
01	0	SPEED [1:0]		0	0	GP 出力	PP
	0			0	1	GP 出力	PP + PU
	0			1	0	GP 出力	PP + PD
	0			1	1	予約済み	
	1			0	0	GP 出力	OD
	1			0	1	GP 出力	OD + PU
	1			1	0	GP 出力	OD + PD
	1			1	1	予約済み（GP 出力 OD）	
10	0	SPEED [1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	予約済み	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	予約済み	
00	x	x	x	0	0	入力	フローティング
	x	x	x	0	1	入力	PU
	x	x	x	1	0	入力	PD
	x	x	x	1	1	予約済み（入力フローティング）	
11	x	x	x	0	0	入力／出力	アナログ
	x	x	x	0	1	予約済み	
	x	x	x	1	0		
	x	x	x	1	1		

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

参考資料

11.3.1 汎用 I/O（GPIO）

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどの I/O ポートは入力フローティングモードに設定されています。

リセット後、デバッグピンはオルタネート機能のプルアップ／プルダウンに設定されています。

- PA15：JTDI プルアップ
- PA14：JTCK/SWCLK プルダウン
- PA13：JTMS/SWDAT プルアップ
- PB4：NJTRST プルアップ
- PB3：JTDO フローティング状態

ピンが出力として設定されている場合、出力データレジスタ（GPIOx_ODR）に書き込まれた値が I/O ピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード（ローレベルのみが駆動され、ハイレベルはハイインピーダンス）で使用することができます。

入力データレジスタ（GPIOx_IDR）は、AHB クロックサイクルごとに、I/O ピン上のデータをキャプチャします。

すべての GPIO ピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDR レジスタの値によってこれらを有効化/無効化できます。

11.3.2 I/O ピンオルタネート機能マルチプレクサと配置

デバイスの I/O ピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に 1 つのペリフェラルオルタネート機能（AF）のみが 1 つの I/O ピンに接続可能となっています。この方法により、同じ I/O ピンを共有するペリフェラル間での競合を無くすことができます。

各 I/O ピンは、最大 16 のオルタネート機能入力（AF0 ～ AF15）を持つマルチプレクサを内蔵しており、GPIOx_AFRL （ピン 0 ～ 7）と GPIOx_AFRH （ピン 8 ～ 15）レジスタを介して設定することができます。

- リセット後、マルチプレクサの選択はオルタネート機能 0（AF0）です。I/O は、GPIOx_MODER レジスタを通してオルタネート機能モードで設定されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。
- Cortex-M7with FPU EVENTOUT は AF15 に配置されています。

この柔軟性の高い I/O 多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれの I/O ピンに配置されており、さらに小型のパッケージで使用できるペリフェラルの数を最適化します。

I/O を任意の設定で使用するには、次の手順に従います。

- **デバッグ機能**：各デバイスのリセット後、これらのピンはデバッガホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **システム機能**：MCOx ピンはオルタネート機能モードに設定する必要があります。
- **GPIO**：必要とする I/O を、GPIOx_MODER レジスタで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**
  - I/O を GPIOx_AFRL または GPIOx_AFRH レジスタの どちらかで必要とする AFx に接続します。
  - タイプ、プルアップ／プルダウン、出力スピードをそれぞれ GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタで選択します。

  - 必要とする I/O を、GPIOx_MODER レジスタでオルタネート機能として設定します。
- **追加機能**：
  - ADC および DAC については、GPIOx_MODER レジスタのアナログモードで必要とする I/O を設定し、ADC および DAC レジスタで必要な機能を設定します。
  - RTC_OUT、RTC_TS、RTC_TAMPx、WKUPx、オシレータなどの追加機能については、関連する RTC、PWR、および RCC レジスタで必要な機能を設定します。これらの機能は、標準の GPIO レジスタの設定よりも優先されます。RTC による I/O 制御の詳細は、*セクション 46.3：1831 ページのRTC の機能説明*を参照してください。
- EVENTOUT
  - AF15 に接続することで、I/O ピンをコア EVENTOUT 信号を出力する用途に設定します。

オルタネート機能 I/O ピンの配置に関する詳細は、デバイスデータシートの「オルタネート機能配置」表を参照してください。

11.3.3 I/O ポート制御レジスタ

各 GPIO ポートには 32 ビットメモリマップド制御レジスタが 4 つ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）あり、最大 16 個の I/O を設定します。GPIOx_MODER レジスタは I/O モード（入力、出力、AF、アナログ）を選択するために使用されます。GPIOx_OTYPER および GPIOx_OSPEEDR レジスタは、出力タイプ（プッシュプルまたはオープンドレイン）およびスピードを選択するために使用されます。I/O の方向がどちらであっても、GPIOx_PUPDR レジスタは、プルアップ／プルダウンを選択するために使用されます。

11.3.4 I/O ポートデータレジスタ

各 GPIO には、16 ビットメモリマップドデータレジスタが 2 つあります。入力データレジスタ（GPIOx_IDR）と出力データレジスタ（GPIOx_ODR）です。GPIOx_ODR は出力されるデータを格納し、読み出し/書き込みアクセスが可能です。I/O から入力されるデータは読み出し専用の入力データレジスタ（GPIOx_IDR）に格納されます。

レジスタの説明は、*セクション 11.4.5：GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A～K）*および *セクション 11.4.6：GPIO ポート入力データレジスタ（GPIOx_ODR）（x = A～K）*を参照してください。

11.3.5 I/O データのビット単位の操作

ビットセット／リセットレジスタ（GPIOx_BSRR）は 32 ビットレジスタで、アプリケーションによる出力データレジスタ（GPIOx_ODR）のビット単位のセット／リセットを可能にします。ビットセットリセットレジスタは GPIOx_ODR の 2 倍のサイズです。

GPIOx_ODR の各ビットには GPIOx_BSRR の 2 つの制御ビット BS(i) と BR(i) が対応します。BS(i) および BR(i) です。ビットBS(i) に 1 を書き込むと、対応する ODR(i) ビットが**セット** されます。ビットBR(i) に 1 を書き込むと、対応する ODR(i) ビットが**リセット** されます。

GPIOx_BSRR のいかなるビットに 0 を書き込んでも GPIOx_ODR の対応するビットには影響しません。仮に、GPIOx_BSRR のビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRR レジスタを使用した GPIOx_ODR 内の個々のビットの変更には、1 回限りの効果しかなく、GPIOx_ODR ビットを固定するものではありません。GPIOx_ODR のビットは常に直接アクセスすることができます。GPIOx_BSRR レジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODR をビットレベルでプログラムする場合は、ソフトウェアで割り込みを無効にする必要はありません。1 回の不可分な AHB 書き込みアクセスで 1 ビットまたは複数ビットを変更することができます。

11.3.6 GPIO ロック機構

GPIOx_LCKR レジスタへ特定の書き込みシーケンスを行うことにより、GPIO 制御レジスタをロックすることができます。停止されるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH です。

GPIOx_LCKR レジスタに書き込むには、特定の書き込み／読み出しシーケンスを行う必要があります。このレジスタのビット 16 に適切な LOCK シーケンスを行う場合、LCKR[15:0] の値を使用して I/O の設定を固定します（この書き込みシーケンス中、LCKR[15:0] の値は同じである必要があります）。あるポートビットに LOCK シーケンスが行われると、次の MCU リセットまたはペリフェラルリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH）の対応するビットが停止されます。

GPIOx_LCKR ビット 16 を [15:0] ビットと同時に設定する必要があるため、この LOCK シーケンス（*セクション 11.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ～K）*を参照）は、GPIOx_LCKR レジスタへのワード（32 ビット長）アクセスを使用してのみ実行できます。

詳細については、*セクション 11.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ～K）*の LCKR レジスタの説明を参照してください。

11.3.7 I/O オルタネート機能の入力／出力

各 I/O が使用できるオルタネート機能入力／出力の 1 つを選択するため、2 つのレジスタが用意されています。これらのレジスタを使用し、必要に応じて、アプリケーションでオルタネート機能を他のピンに接続することができます。

つまり、GPIOx_AFRL および GPIOx_AFRH オルタネート機能レジスタを使用していくつかの使用可能なペリフェラル機能が、各 GPIO に多重化されることになります。こうして各 I/O に使用可能な機能のどれか 1 つをアプリケーションによって選択できます。AF 選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、任意の I/O が持つオルタネート機能入力／出力に対し 1 つのチャネルが選択されます。

どの機能が各 GPIO ピンに多重化されているかについてはデバイスデータシートを参照してください。

11.3.8 外部割り込み／ウェイクアップライン

すべてのポートに外部割り込み機能があります。外部割り込みラインを使用するには、ポートを入力モードに設定する必要があります。*セクション 20：拡張割り込み／イベントコントローラ（EXTI）*および*セクション 20.3：EXTI の機能説明*を参照してください。

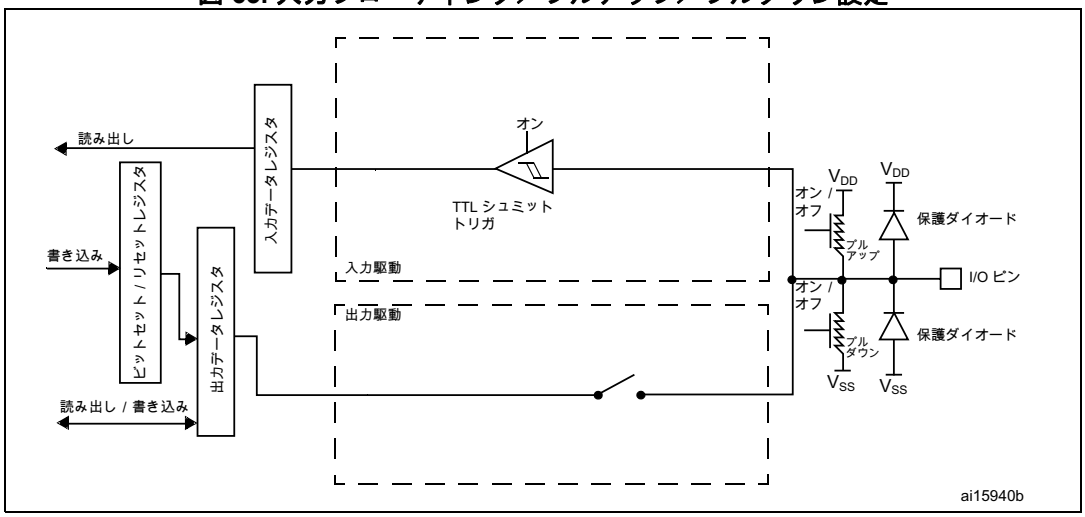
11.3.9 入力設定

I/O ポートが入力としてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。

図 63 は、I/O ポートビットの入力設定を示しています。

図 63. 入力フローティング／プルアップ／プルダウン設定



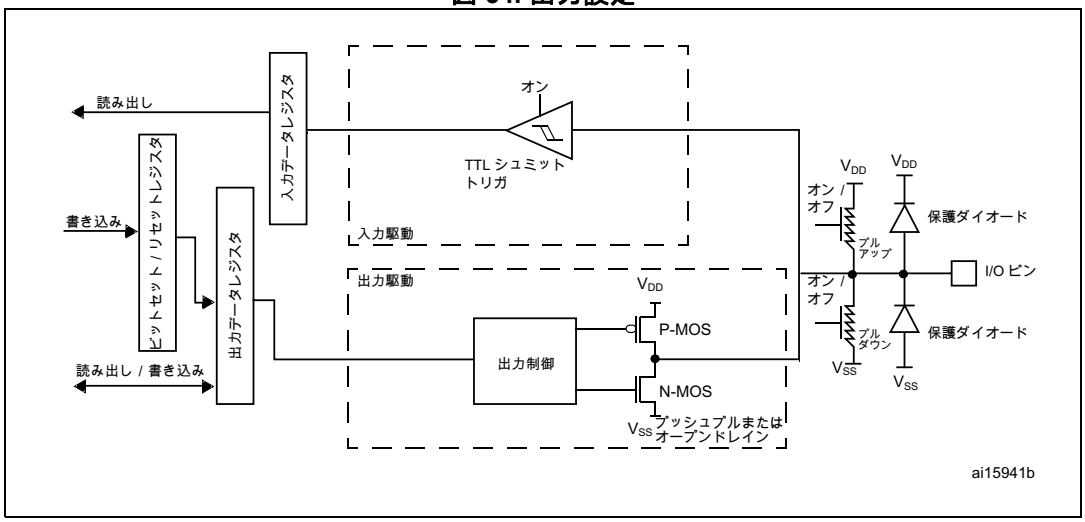
11.3.10 出力設定

I/O ポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
  - オープンドレインモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときはポートはハイインピーダンス状態のままです（P-MOS が有効になることはない）。
  - プッシュプルモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときは P-MOS が有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。
- 出力データレジスタの読み出しアクセスによって、最後に書き込まれたデータが得られます。

図 64 は、I/O ポートビットの出力設定を示しています。

図 64. 出力設定



11.3.11 I/O 補正セル

このセルは、I/O 遷移スルーレート（t_{fall} / t_{rise}）を制御して電源の I/O ノイズを低減するために使用します。

セルは、2 つのブロックに分割されます。

- 最初のブロックは、現在の PVT の最適なコードを供給します。このブロックに格納されているコードは、SYSCFG_CCSR の READY フラグがセットされたときに読み出すことができます。
- 2 つ目のブロックでは I/O スルーレートを制御します。ユーザは適用するコードを選択して、ソフトウェアによってプログラムします。

I/O 補正セルには、1.62 から 2.0 V および 2.7 から 3.6 V の 2 つの電圧範囲があります。

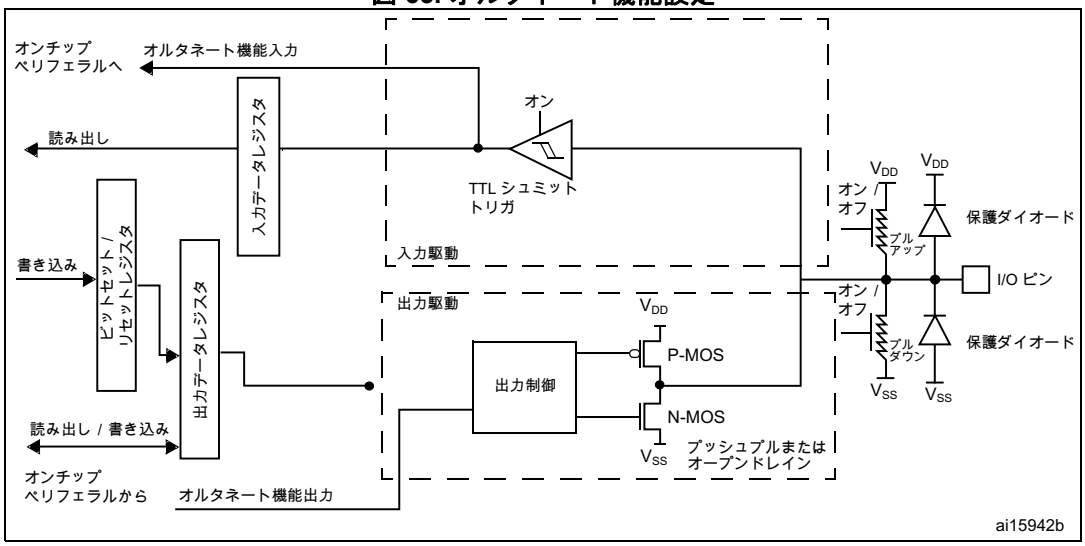
11.3.12 オルタネート機能設定

I/O ポートがオルタネート機能としてプログラムされた場合、

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル（トランスミッタイネーブルおよびデータ）からの信号で駆動されます。
- シュミットトリガ入力が有効になります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。

図 65 は、I/O ポートビットのオルタネート機能設定 を示しています。

図 65. オルタネート機能設定



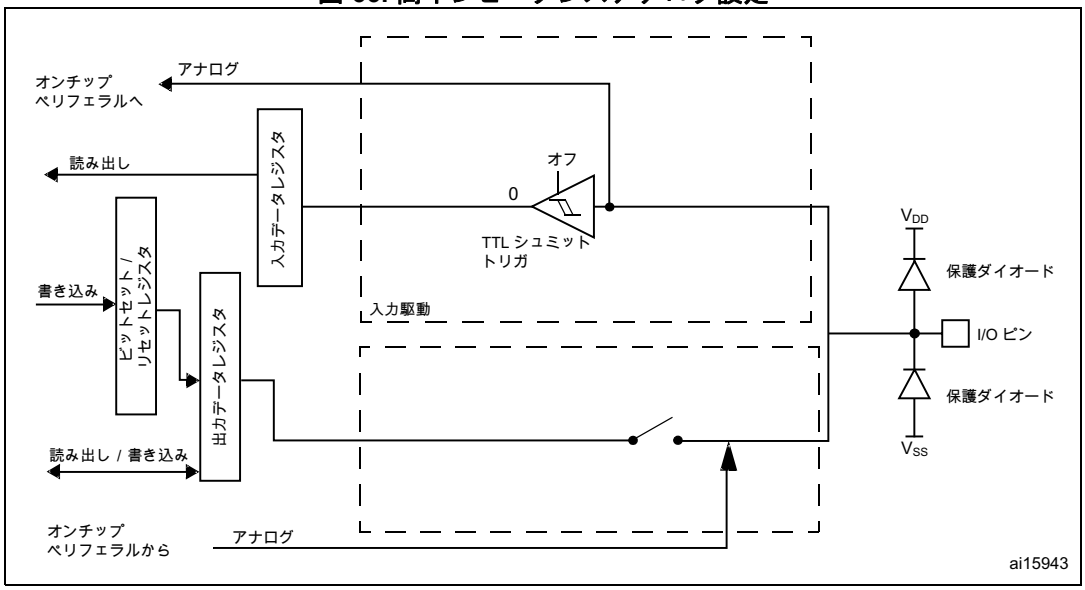
11.3.13 アナログ設定

I/O ポートがアナログとしてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップ抵抗およびプルダウン抵抗はハードウェアによって無効にされます。
- 入力データレジスタの読み出しアクセスを行うと、値“0”が得られます。

図 66 は、I/O ポートビットのハイインピーダンスアナログ入力設定を示しています。

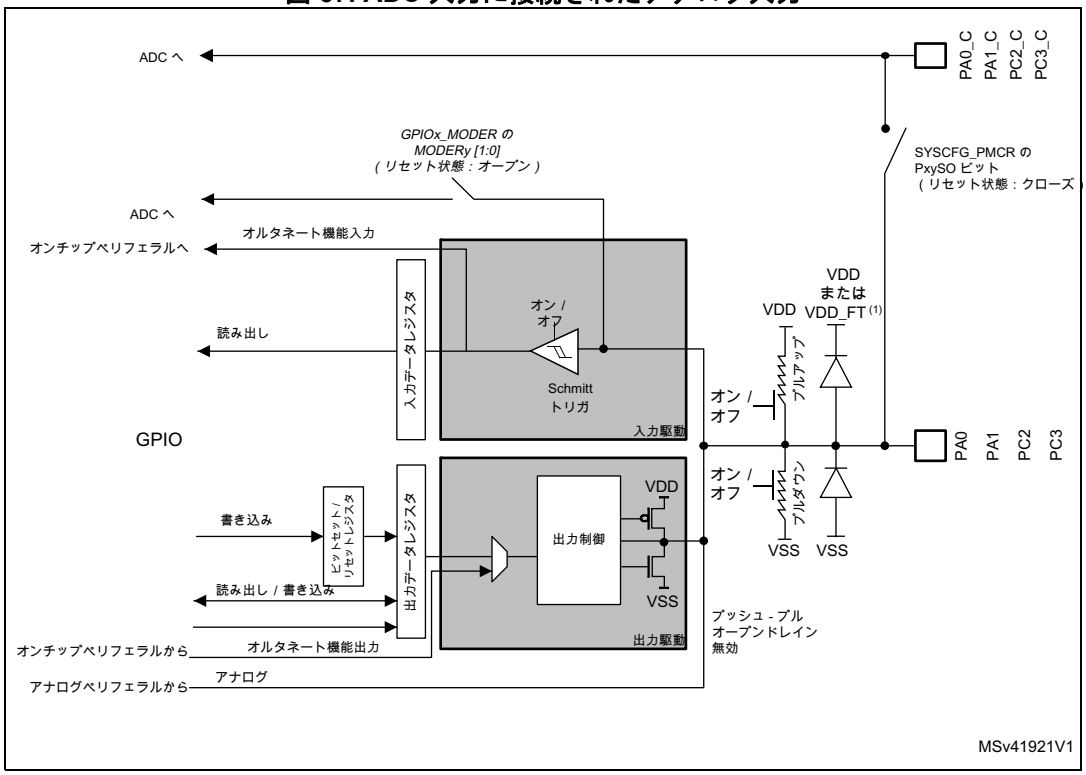
図 66. 高インピーダンスアナログ設定



一部のピン／ボールは直接 PA0_C、PA1_C、PC2_C、および PC3_Cの ADC アナログ入力（*図 67* を参照）に接続されています。Pxy_C と Pxy ピン／ボールの間には、アナログスイッチを経由する直

接パスがあります（アナログスイッチの設定方法の詳細については、*セクション 12.3.1 : SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR）*を参照）。

図 67. ADC 入力に接続されたアナログ入力



1. VDD_FT は 5 V トレラント I/O に固有の電位です。VDD とは異なります。

11.3.14 HSE または LSE オシレータのピンを GPIO として使用

HSE または LSE オシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータピンを 通常の GPIO として使用することができます。

HSE または LSE オシレータがスイッチオンされた場合（RCC_CSR レジスタの HSEON または LSEON ビットを設定することで）、オシレータは関連ピンを制御しますが、これらのピンの GPIO 設定は無効です。

オシレータがユーザ外部クロックモードに設定されている場合、OSC_IN または OSC32_IN ピンのみがクロック入力のために確保されますが、OSC_OUT または OSC32_OUT ピンは通常の GPIO として使用することができます。

11.3.15 GPIO ピンをバックアップ供給ドメインで使用

コア供給ドメインの電源がオフになったとき（デバイスが STANDBY モードに移行したとき）、PC13/PC14/PC15/PI8 の GPIO が機能しなくなります。この場合、それらの GPIO 設定が RTC 設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

セクション 46.3 : 1831 ページのRTC の機能説明

参考資料

11.4 GPIO レジスタ

このセクションには、GPIO レジスタの詳細な説明が記載されています。

レジスタビット、レジスタアドレスオフセット、リセット値の概要は、*表 84*を参照してください。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

11.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x =A～K）

アドレスオフセット：0x00

リセット値：

- ポート A 0xABFF FFFF
- ポート B 0xFFFF FEBF
- 他のポート 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **MODERy[1:0]**：ポート x 設定ビット（y = 0～15）
これらのビットは、I/O モードを設定するために、ソフトウェアによって書き込まれます。
00：入力モード（リセット状態）
01：汎用出力モード
10：オルタネート機能モード
11：アナログモード

11.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPERR）（x = A～K）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **OTy**：ポート x 設定ビット（y = 0～15）
これらのビットは、I/O 出力タイプを設定するために、ソフトウェアによって書き込まれます。
0：出力プッシュプル（リセット状態）
1：出力オープンドレイン

11.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A～K）

アドレスオフセット：0x08

リセット値：

- ポート A 0x0C00 0000
- ポート B 0x0000 00C0
- 他のポート 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEEDR15 [1:0]		OSPEEDR14 [1:0]		OSPEEDR13 [1:0]		OSPEEDR12 [1:0]		OSPEEDR11 [1:0]		OSPEEDR10 [1:0]		OSPEEDR9 [1:0]		OSPEEDR8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSPEEDR7 [1:0]		OSPEEDR6 [1:0]		OSPEEDR5 [1:0]		OSPEEDR4 [1:0]		OSPEEDR3 [1:0]		OSPEEDR2 [1:0]		OSPEEDR1 [1:0]		OSPEEDR0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **OSPEEDRy[1:0]**：ポート x 設定ビット（y = 0～15）

これらのビットは、I/O の出力スピードを設定するために、ソフトウェアによって書き込まれます。

00：ロースピード

01：ミディアムスピード

10：ハイスピード

11：超ハイスピード

注： ***OSPEEDRy ビット値対 V_{DD} 範囲および外部負荷については製品データシートを参照してください。***

11.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A～K）

アドレスオフセット：0x0C

リセット値：

- ポート A 0x6400 0000
- ポート B 0x0000 0100
- 他のポート 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **PUPDRy[1:0]**：ポート x 設定ビット（y = 0〜15）

これらのビットは、I/O のプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

00：プルアップ／プルダウンなし
01：プルアップ
10：プルダウン
11：予約済み

11.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A〜K）

アドレスオフセット：0x10

リセット値：0x0000 XXXX（X は未定義）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **IDRy**：ポート入力データビット（y = 0〜15）

これらのビットは読み出し専用です。これらのビットには、対応する I/O ポートの入力値が格納されています。

11.4.6 GPIO ポート入力データレジスタ（GPIOx_ODR）（x = A〜K）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ODRy**：ポート出入力データビット（y = 0〜15）

これらのビットは、ソフトウェアによって読み出し／書き込みができます。

注： 不可分なビット単位のセット／リセットのために、GPIOx_BSRR または GPIOx_BRR レジスタ（x = A〜F）への書き込みによって、ODR ビットを個々にセット／リセットすることができます。

11.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A～K）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 **BRy：**ポート x リセットビット y（y= 0～15）

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODRx ビットの値は変化しません。

1：対応する ODRx ビットをリセットします。

注： BSx ビットと BRx ビットの両方がセットされた場合、BSx ビットが優先されます。

ビット 15:0 **BSy：**ポート x セットビット y（y= 0～15）

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODRx ビットの値は変化しません。

1：対応する ODRx ビットをセットします。

11.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～K）

このレジスタは、ビット 16（LCKK）に正しい書き込みシーケンスが行われたときに、ポートビットの設定をロックするために使用されます。ビット [15:0] の値は、GPIO の設定をロックするために使用されます。書き込みシーケンスの間は、LCKR[15:0] の値を変更することはできません。あるポートビットに LOCK シーケンスが適用されると、次の MCU リセットまたはペリフェラルリセットまで、このポートビットの値を変更できなくなります。

注： GPIOx_LCKR レジスタへの書き込みには特定の書き込みシーケンスが使われます。このロックシーケンスではワードアクセス（32 ビット長）のみ可能です。

各ロックビットによって、特定の設定レジスタ（制御レジスタおよびオルタネート機能レジスタ）が固定されます。

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LCKK**：ロックキー

このビットは常時読み出しができます。ロックキー書き込みシーケンスを使用しなければ変更できません。

0：ポート設定ロックキーは無効です。

1：ポート設定ロックキーは有効です。GPIOx_LCKR レジスタは、次の MCU リセットまたはペリフェラルリセットまでロックされます。

ロックキー書き込みシーケンス：

書き込み LCKR[16] = “1” + LCKR[15:0]

書き込み LCKR[16] = “0” + LCKR[15:0]

書き込み LCKR[16] = “1” + LCKR[15:0]

読み出し LCKR

読み出し LCKR[16] = “1”（この読み出し操作は任意だが、ロックが有効であることを確認できる。）

注： ロックキー書き込みシーケンスの間は、LCK[15:0] の値を変更することはできません。

ロックシーケンス中にエラーが発生すると、ロックは中止されます。

ポートの任意のビットの最初のロックシーケンスの後、次の MCU リセットまたはペリフェラルリセットまでは、LCKK ビットのいかなる読み出しアクセスに対しても、“1”が返されます。

ビット 15:0 **LCKy**：ポート x ロックビット y（y= 0～15）

これらのビットは読み出し／書き込みができますが、書き込めるのは LCKK ビットが“0”のときだけです。

0：ポート設定はロックされません。

1：ポート設定はロックされます。

11.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A～K）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR7[3:0]				AFR6[3:0]				AFR5[3:0]				AFR4[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR3[3:0]				AFR2[3:0]				AFR1[3:0]				AFR0[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 AFR**y[3:0]**：ポート x ピン y（y = 0～7）のオルタネート機能選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択：

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

11.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ～ J）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR15[3:0]				AFR14[3:0]				AFR13[3:0]				AFR12[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR11[3:0]				AFR10[3:0]				AFR9[3:0]				AFR8[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 AFR**y[3:0]**：ポート x ピン y（y = 8～15）のオルタネート機能選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択：

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

11.4.11 GPIO レジスタマップ

次の表に、GPIO レジスタマップとリセット値を示します。

表 84. GPIO レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	GPIOA_MODER	MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]		MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
	リセット値	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x00	GPIOB_MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
0x00	GPIOx_MODER（x = **C ～ K**）	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODEv1[1:0]		MODER0[1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	GPIOx_OTYPER（x = **A ～ K**）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOA_OSPEEDR	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]	
	リセット値	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOB_OSPEEDR	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x08	GPIOx_OSPEEDR（x = **C ～ K**）	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOA_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
	リセット値	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOB_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

表 84. GPIO レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C	GPIOx_PUPDR（x = C ～ K）	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPIOx_IDR（x = A ～ I/J/K）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
	リセット値																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x14	GPIOx_ODR（x = A ～ K）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	GPIOx_BSRR（x = A ～ I/J/K）	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	GPIOx_LCKR（x = A ～ K）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK	LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	GPIOx_AFRL（x = A ～ K）	AFR7[3:0]				AFR6[3:0]				AFR5[3:0]				AFR4[3:0]				AFR3[3:0]				AFR2[3:0]				AFR1[3:0]				AFR0[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	GPIOx_AFRH（x = A ～ K）	AFR15[3:0]				AFR14[3:0]				AFR13[3:0]				AFR12[3:0]				AFR11[3:0]				AFR10[3:0]				AFR9[3:0]				AFR8[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

12 システム設定コントローラ（SYSCFG）

12.1 概要

デバイスは一連の設定レジスタを持っています。このセクションの目的は、システム設定コントローラについて詳細に説明することです。

12.2 SYSCFG の主な機能

システム設定コントローラの主な機能は以下のとおりです。

- アナログスイッチ設定管理
- I2C Fm+ 設定
- Ethernet PHY インタフェースの選択
- 外部割り込みラインの GPIO への接続の管理
- I/O 補正セルの管理
- 読み出し保護とフラッシュメモリバンクスワップ情報の取得
- ブートシーケンスとブートアドレスの管理
- BOR リセットレベルの管理
- セキュアおよび保護されているフラッシュメモリのセクタ状態の管理
- フラッシュメモリの書き込み保護ステータスの管理
- DTCM のセキュアなセクションステータスの管理
- 独立型ウォッチドッグ動作の管理（ハードウェアまたはソフトウェア／停止）
- STOP モードおよび STANDBY モードでのリセット生成
- セキュアモードの有効化／無効化

12.3 SYSCFG レジスタの説明

12.3.1 SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR）

アドレスオフセット：0x04

リセット値：0x0X00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	PC3SO	PC2SO	PA1SO	PA0SO	EPIS[2:0]			Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw	rw	rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOSTE	PB9 FMP	PB8 FMP	PB7 FMP	PB6 FMP	I2C4 FMP	I2C3 FMP	I2C2 FMP	I2C1 FMP
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **PC3SO**：PC3 スイッチ開

このビットによって、PC3 と PC3_C （デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2 つのパッド）

ビット 26 **PC2SO**：PC2 スイッチ開

このビットによって、PC2 と PC2_C （デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2 つのパッド）

ビット 25 **PA1SO**：PA1 スイッチ開

このビットによって、PA1 と PA1_C （デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2 つのパッド）

ビット 24 **PA0SO**：PA0 スイッチ開

このビットによって、PA0 と PA0_C （デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2 つのパッド）

ビット 23:21 **EPIS[2:0]**：Ethernet PHY インタフェース選択

これらのビットは Ethernet PHY インタフェースを選択します。

000：MII

001：予約済み

010：予約済み

011：予約済み

100：RMII

101：予約済み

110：予約済み

111：予約済み

ビット 20:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **BOOSTE**：ブースタ有効化

このビットは、供給電圧が 2.7 V 未満のときに、ブースタによるアナログスイッチの全高調波ひずみの低減を有効にします。

ブースタを有効にすると、供給電圧が 2.7 V 未満のときのアナログスイッチの AC 性能を保証できます。この場合、アナログスイッチ性能は全電圧範囲で同じです。

0：ブースタは無効です。

1：ブースタは有効です。

ビット 7 **PB9FMP**：PB(9) 高速モードプラス (Fm+)

このビットは、PB(9) の I2C Fm+ を有効にします。

0：Fm+ は無効です。

1：Fm+ は有効です。

ビット 6 **PB8FMP**：PB(8) 高速モードプラス (Fm+)

このビットは、PB(8) の I2C Fm+ を有効にします。

0：Fm+ は無効です。

1：Fm+ は有効です。

ビット 5 **PB7FMP**：PB(7) 高速モードプラス(Fm+)
このビットは、PB(7) の I2C Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 4 **PB6FMP**：PB(6) 高速モードプラス(Fm+)
このビットは、PB(6) の I2C Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 3 **I2C4FMP**：I2C4 高速モードプラス(Fm+)
このビットは、I2C4 の Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 2 **I2C3FMP**：I2C3 高速モードプラス(Fm+)
このビットは、I2C3 の Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 1 **I2C2FMP**：I2C2 高速モードプラス(Fm+)
このビットは、I2C2 の Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 0 **I2C1FMP**：I2C1 高速モードプラス(Fm+)
このビットは、I2C1 の Fm+ を有効にします。
0：Fm+ は無効です。
1：Fm+ は有効です。

12.3.2 SYSCFG 外部割り込み設定レジスタ 1（SYSCFG_EXTICR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 0～3）

これらのビットは、外部割り込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

12.3.3 SYSCFG 外部割り込み設定レジスタ 2（SYSCFG_EXTICR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7[3:0]				EXTI6[3:0]				EXTI5[3:0]				EXTI4[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 4～7）

これらのビットは、外部割り込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

12.3.4 SYSCFG 外部割り込み設定レジスタ 3（SYSCFG_EXTICR3）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11[3:0]				EXTI10[3:0]				EXTI9[3:0]				EXTI8[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 8～11）

これらのビットは、外部割り込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

注： PK[11:8] は使用されません。

12.3.5 SYSCFG 外部割り込み設定レジスタ 4（SYSCFG_EXTICR4）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15[3:0]				EXTI14[3:0]				EXTI13[3:0]				EXTI12[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 12～15）

これらのビットは、外部割り込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

注： PK[15:12] は使用されません。

12.3.6 SYSCFG 補正セル制御／ステータスレジスタ（SYSCFG_CCCSR）

アドレスオフセット：0x20

リセット値：0x0000 0000

I/O 補正メカニズムの詳細な説明については、*セクション 11.3.11 : I/O 補正セル*を参照してください。

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **HSLV**：低電圧でのハイスピード

このビットは、製品の電圧が低い場合に、I/O スピードを最適化するために、ソフトウェアによって書き込まれます。

このビットは、IO_HSLV ユーザオプションビットがセットされたときのみ有効です。製品の供給電圧が 2.5 V 未満の場合にのみ使用する必要があります。V_{DD} が 2.5 V よりも高い場合にこのビットをセットすると、デバイスにダメージを与えます。

0：I/O スピードは最適化されません。

1：I/O スピードを最適化します。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **READY**：補正セルレディフラグ

このビットは、補正セルのステータスを提供します。

0：I/O 補正セルはレディ状態ではありません。

1：I/O 補正セルはレディ状態です。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CS**：コード選択

このビットは、I/O 補正セルに適用するコードを選択します。

0：セルからのコード（SYSCFG_CCVR で使用可能）

1：SYSCFG 補正セルコードレジスタ（SYSCFG_CCCR）からのコード

ビット 0 **EN**：有効化

このビットは、I/O 補正セルを有効にします。

0：I/O 補正セルは無効です。

1：I/O 補正セルは有効です。

12.3.7 SYSCFG 補正セル値レジスタ（SYSCFG_CCVR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **PCV[3:0]**：PMOS 補正値

この値は、セルによって提供され、PMOS トランジスタの I/O 補正セルコードを計算するために CPU で使用できます。このコードは、SYSCFG_CMPCR の CS ビットがリセットされた場合に、I/O 補正セルに適用されます。

ビット 3:0 **NCV[3:0]**：NMOS 補正値

この値は、セルによって提供され、NMOS トランジスタの I/O 補正セルコードを計算するために CPU で使用できます。このコードは、SYSCFG_CMPCR の CS ビットがリセットされた場合に、I/O 補正セルに適用されます。

12.3.8 SYSCFG 補正セルコードレジスタ（SYSCFG_CCCR）

アドレスオフセット：0x28

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **PCC[3:0]**：PMOS 補正コード

これらのビットは、PMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CMPCR の CS ビットがセットされた場合に、I/O 補正セルに適用されます。

ビット 3:0 **NCC[3:0]**：NMOS 補正コード

これらのビットは、NMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。このコードは、SYSCFG_CCCSR の CS ビットがセットされた場合に、I/O 補正セルに適用されます。

12.3.9 SYSCFG パッケージレジスタ（SYSCFG_PKGR）

アドレスオフセット：0x124

リセット値：0x000X 000X

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PKG[3:0]：**パッケージ

このビットは、デバイスパッケージを示します。

0000：LQFP100（STM32H7x3）

0010：TQFP144（STM32H7x3）

0101：TQFP176/UFBGA176 （STM32H7x3）

1000：LQFP208/TFBGA240 （STM32H7x3）

その他の設定：すべてのパッドが有効です。

12.3.10 SYSCFG ユーザレジスタ 0 （SYSCFG_UR0）

アドレスオフセット：0x300

リセット値：0x00XX 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDP[7:0]							
								r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKS
															r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RDP[7:0]**：読み出し保護

このビットは、読み出し保護レベルを示します。

0xAA：レベル 0 （保護なし）

0xCC：レベル 2 （フラッシュメモリの読み出し保護、フルデバッグ機能、SRAM からのブート、および境界スキャンの無効化）

その他の設定：レベル 1 （フラッシュメモリの読み出し保護、限定デバッグ機能、および境界スキャンの有効化）

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BKS**：バンクスワップ

このビットは、フラッシュメモリのバンクマッピングを示します。

0：フラッシュメモリのバンクアドレスが反転されます。

1：フラッシュメモリのバンクが元のアドレスにマッピングされます。

12.3.11 SYSCFG ユーザレジスタ 2 （SYSCFG_UR2）

アドレスオフセット：0x308

リセット値：0xXXXX 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BOOT_ADD0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BORH[1:0]	
														r	r

ビット 31:16 BOOT_**ADD0[15:0]**：ブートアドレス 0
これらのビットは、BOOT ピンがローの場合に、コアブートアドレスの MSB を定義します。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **BORH[1:0]**：BOR_LVL ブラウンアウトリセット閾値レベル
これらのビットは、ハイレベルのブラウンアウトリセットを示します。
0x11：2.7 から 3.6 V 電圧範囲の BOR ハイリセットレベル閾値
0x10：2.4 から 2.7 V 電圧範囲の BOR ミディアムリセットレベル閾値
0x01：2.1 から 2.4 V 電圧範囲の BOR ローリセットレベル閾値
0x00：2.7 から 3.6 V 電圧範囲の BOR オフリセットレベル閾値

12.3.12 SYSCFG ユーザレジスタ 3 （SYSCFG_UR3）

アドレスオフセット：0x30C

リセット値：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT_ADD1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BOOT_ADD1[15:0]**：ブートアドレス 1
これらのビットは、BOOT ピンがハイの場合に、コアブートアドレスの MSB を定義します。

12.3.13 SYSCFG ユーザレジスタ 4 （SYSCFG_UR4）

アドレスオフセット：0x310

リセット値：0x000X XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEPAD_1
															r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **MEPAD_1**：バンク 1 の全体消去保護領域無効化
このビットは、フラッシュの保護領域（バンク 1）が全体消去の影響を受けることを示します。
0：全体消去が実施された場合に、保護領域を消去します。
1：全体消去が実施された場合に、保護領域を消去しません。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

12.3.14 SYSCFG ユーザレジスタ 5 （SYSCFG_UR5）

アドレスオフセット：0x314

リセット値：0x00XX 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPS_1[7:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MESAD_1
															r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **WRPN_1[7:0]**：フラッシュバンク 1 の書き込み保護

WRPN[i] ビットは、フラッシュメモリバンク 1 のセクタ i が保護されているかどうかを示します。

0：セクタ i で書き込み保護がアクティブになっています。

1：セクタ i で書き込み保護がアクティブになっていません。

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MESAD_1**：バンク 1 の全体消去セキュア領域無効化

このビットは、フラッシュのセキュア領域（バンク 1）が全体消去の影響を受けることを示します。

0：全体消去が実施された場合に、セキュア領域を消去します。

1：全体消去が実施された場合に、セキュア領域を消去しません。

12.3.15 SYSCFG ユーザレジスタ 6 （SYSCFG_UR6）

アドレスオフセット：0x318

リセット値：0x0XXX 0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	PA_END_1[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PA_BEG_1[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **PA_END_1[11:0]**：バンク 1 の保護領域終了アドレス

バンク 1 の保護領域の終了アドレスです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PA_BEG_1[11:0]**：バンク 1 の保護領域開始アドレス

バンク 1 の保護領域の開始アドレスです。

12.3.16 SYSCFG ユーザレジスタ 7 （SYSCFG_UR7）

アドレスオフセット：0x31C

リセット値：0x0XXX 0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SA_END_1[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SA_BEG_1[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **SA_END_1[11:0]**：バンク 1 のセキュア領域終了アドレス
バンク 1 のセキュア領域の終了アドレスです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SA_BEG_1[11:0]**：バンク 1 のセキュア領域開始アドレス
バンク 1 のセキュア領域の開始アドレスです。

12.3.17 SYSCFG ユーザレジスタ 8 （SYSCFG_UR8）

アドレスオフセット：0x320

リセット値：0x000X 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MESAD_2
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEPAD_2
															r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MESAD_2**：バンク 2 の全体消去セキュア領域無効化
このビットは、フラッシュメモリのセキュア領域（バンク 2）が全体消去の影響を受けることを示します。
0：全体消去が実施された場合に、セキュア領域を消去します。
1：全体消去が実施された場合に、セキュア領域を消去しません。

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MEPAD_2**：バンク 2 の全体消去保護領域無効化
このビットは、フラッシュメモリの保護領域（バンク 2）が全体消去の影響を受けることを示します。
0：全体消去が実施された場合に、保護領域を消去します。
1：全体消去が実施された場合に、保護領域を消去しません。

12.3.18 SYSCFG ユーザレジスタ 9 （SYSCFG_UR9）

アドレスオフセット：0x324

リセット値：0x0XXX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	PA_BEG_2[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPS_2[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PA_BEG_2[11:0]**：バンク 2 の保護領域開始アドレス
バンク 2 の保護領域の開始アドレスです。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRPN_2[7:0]**：フラッシュバンク 2 の書き込み保護
WRPN[i] ビットは、フラッシュメモリバンク 2 のセクタ i が保護されているかどうかを示します。
0：セクタ i で書き込み保護がアクティブになっています。
1：セクタ i で書き込み保護がアクティブになっていません。

12.3.19 SYSCFG ユーザレジスタ 10 （SYSCFG_UR10）

アドレスオフセット：0x328

リセット値：0x0XXX 0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SA_BEG_2[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PA_END_2[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SA_BEG_2[11:0]**：バンク 2 のセキュア領域開始アドレス
バンク 2 のセキュア領域の開始アドレスです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PA_END_2[11:0]**：バンク 2 の保護領域終了アドレス
バンク 2 の保護領域の終了アドレスです。

12.3.20 SYSCFG ユーザレジスタ 11 （SYSCFG_UR11）

アドレスオフセット：0x32C

リセット値：0x000X 0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IWDG1M
															r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SA_END_2[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **IWDG1M**：独立型ウォッチドッグ 1 モード

このビットは、独立型ウォッチドッグ 1 （IWDG1）の制御モードを示します。

0：IWDG1 をソフトウェアで制御します。

1：IWDG1 をハードウェアで制御します。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SA_END_2[11:0]**：バンク 2 のセキュア領域終了アドレス

バンク 2 のセキュア領域の終了アドレスです。

12.3.21 SYSCFG ユーザレジスタ 12 （SYSCFG_UR12）

アドレスオフセット：0x330

リセット値：0x000X 000X

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
															r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SECURE**：セキュアモード

このビットは、セキュアモードステータスを示します。

0：セキュアモードは無効です。

1：セキュアモードは有効です。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

12.3.22 SYSCFG ユーザレジスタ 13 （SYSCFG_UR13）

アドレスオフセット：0x334

リセット値：0x000X 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D1SBRST
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDRS[1:0]	
														r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **D1SBRST**：D1 STANDBY リセット

このビットは、D1 ドメインが DStandby モードに移行した場合にリセットが生成されるかどうかを示します。

0：D1 STANDBY モードに移行することでリセットを生成します。

1：リセットを生成せずに D1 STANDBY モードに移行します。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SDRS[1:0]**：セキュア DTCM RAM サイズ

このビットは、セキュア DTCM RAM のサイズを示します。

00：2 KB

01：4 KB

10：8 KB

11：16 KB

12.3.23 SYSCFG ユーザレジスタ 14 （SYSCFG_UR14）

アドレスオフセット：0x338

リセット値：0x000X 000X

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D1STPRST
															rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **D1STPRST**：D1 STOP リセット

このビットは、D1 ドメインが DStop モードに移行した場合にリセットが生成されるかどうかを示します。

0：D1 STOP モードに移行することでリセットを生成します。

1：リセットを生成せずに D1 STOP モードに移行します。

12.3.24 SYSCFG ユーザレジスタ 15 （SYSCFG_UR15）

アドレスオフセット：0x33C

リセット値：0x000X 000X

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **FZIWDGSTB**：独立型ウォッチドッグの STANDBY モードでの停止

このビットは、独立型ウォッチドッグが STANDBY モードで停止しているかどうかを示します。

0：独立型ウォッチドッグが STANDBY モードで停止しています。

1：独立型ウォッチドッグが STANDBY モードで動作しています。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

12.3.25 SYSCFG ユーザレジスタ 16 （SYSCFG_UR16）

アドレスオフセット：0x340

リセット値：0x000X 000X

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **PKP**：プログラムされた秘密鍵

このビットは、デバイスの秘密鍵がプログラムされているかどうかを示します。

0：秘密鍵がプログラムされていません。

1：秘密鍵がプログラムされています。

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FZIWDGSTP**：独立型ウォッチドッグの STOP モードでの停止

このビットは、独立型ウォッチドッグが STOP モードで停止しているかどうかを示します。

0：独立型ウォッチドッグが STOP モードで停止されている場合。

1：独立型ウォッチドッグが STOP モードで動作している場合。

12.3.26 SYSCFG ユーザレジスタ 17 （SYSCFG_UR17）

アドレスオフセット：0x344

リセット値：0x0000 000X

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **IO_HSLV**：I/O ハイスピード／低電圧

このビットは、IOHSLV オプションビットがセットされていることを示します。

0：製品は全電圧レンジで動作します。

1：製品は 2.5 V 未満で動作します。

12.3.27 SYSCFG レジスタマップ

次の表に、SYSCFG レジスタマップとリセット値を示します。

表 85. SYSCFG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	予約済み	予約済み																															
0x04	SYSCFG_PMCR	Res.	Res.	Res.	Res.	PC3SO	PC2SO	PA1SO	PA0SO	EPIS[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOSTE	PB9FMP	PB8FMP	PB7FMP	PB6FMP	I2C4FMP	I2C3FMP	I2C2FMP	I2C1FMP
	リセット値					X	X	X	0	0	0	0													0	0	0	0	0	0	0	0	0
0x08	SYSCFG_EXTICR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SYSCFG_EXTICR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI7[3:0]				EXTI6[3:0]				EXTI5[3:0]				EXTI4[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SYSCFG_EXTICR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI11[3:0]				EXTI10[3:0]				EXTI9[3:0]				EXTI8[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	SYSCFG_EXTICR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI15[3:0]				EXTI14[3:0]				EXTI13[3:0]				EXTI12[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SYSCFG_CCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	READY	Res.	Res.	Res.	Res.	Res.	CS	EN
	リセット値																0									0						0	0
0x24	SYSCFG_CCVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCV[3:0]				NCV[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0x28	SYSCFG_CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCC[3:0]				NCC[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0x2C - 0x120	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x124	SYSCFG_PKGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKG[3:0]			
	リセット値																													X	X	X	X
0x128 - 0x2FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x300	SYSCFG_UR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDP[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKS
	リセット値									x	x	x	x	x	x	x	x																x
0x304	予約済み	予約済み																															
0x308	SYSCFG_UR2	BOOT_ADD0[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BORH[1:0]	
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X															X	X
0x30C	SYSCFG_UR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_ADD1[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

表 85. SYSCFG レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x310	SYSCFG_UR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEPAD_1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																X																
0x314	SYSCFG_UR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPN_1[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MESAD_1
	リセット値									X	X	X	X	X	X	X	X																X
0x318	SYSCFG_UR6	Res.	Res.	Res.	Res.	PA_END_1[11:0]												Res.	Res.	Res.	Res.	PA_BEG_1[11:0]											
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x31C	SYSCFG_UR7	Res.	Res.	Res.	Res.	SA_END_1[11:0]												Res.	Res.	Res.	Res.	SA_BEG_1[11:0]											
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x320	SYSCFG_UR8	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MESAD_2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEPAD_2
	リセット値																x																x
0x324	SYSCFG_UR9	Res.	Res.	Res.	Res.	PA_END_2[11:0]												Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRPN_2[7:0]							
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x328	SYSCFG_UR10	Res.	Res.	Res.	Res.	SA_BEG_2[11:0]												Res.	Res.	Res.	Res.	PA_END_2[11:0]											
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x32C	SYSCFG_UR11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IWDG1M	Res.	Res.	Res.	Res.	SA_END_2[11:0]											
	リセット値																X					X	X	X	X	X	X	X	X	X	X	X	X
0x330	SYSCFG_UR12	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURE	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	Res.
	リセット値																X																X
0x334	SYSCFG_UR13	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	D1SBRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDRS[1:0]		
	リセット値																X														X	X	
0x338	SYSCFG_UR14	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	D1STPRST
	リセット値																X																X
0x33C	SYSCFG_UR15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FZIWDGSTB	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	Res.
	リセット値																X																X

表 85. SYSCFG レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x340	SYSCFG_UR16	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKP	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	FZIWDGSTP
	リセット値																X																X
0x344	SYSCFG_UR17	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	RES.	IO_HSLV
	リセット値																																X

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

13 ブロック相互接続

13.1 ペリフェラル相互接続

13.1.1 概要

いくつかのペリフェラルには相互に直接接続できるものがあります。

これにより、ペリフェラル間での自動的な通信や同期が可能になり、CPU リソースと消費電力を低減できます。

これらのハードウェア接続によってソフトウェア制御の遅延がなくなり、予測可能なシステム設計が可能になり、また結果としてピン数および GPIO 数が削減されます。

13.1.2 接続の概要

接続にはいくつかのタイプがあります。

- **非同期接続（A）**

  ソース出力信号はデスティネーションクロックによってサンプリングされます。これにより、ソース出力イベントとデスティネーションイベント検出の間の遅延にジッタが発生する可能性があります。
- **同期接続（S）**

  転送元と転送先は両方とも同期（同じクロックで実行）しています。転送元から転送先への遅延が予想可能です。ジッタは考慮されません。
- **即時接続（I）**

  転送元または転送先のどちらかがアナログ信号です。
- **TIM/HRTIM 出力のブレーク／フォールト接続（B）**

  ソース出力信号は、純粋な組み合わせ論理パスを介して、タイマ出力を遅延なく無効化します。

表 86. ペリフェラル相互接続マトリックス（D2 ドメイン）(1) (2)

転送元			転送先																										
			D2 ドメイン																				D3 ドメイン						
			APB1									APB2							AHB1			AHB4	APB4						
			TIM2	TIM3	TIM4	TIM5	TIM12	LPTIM1	DAC	CRS	CAN	TIM1	TIM8	TIM15	TIM16	TIM17	DFSDM1	HRTIM	ADC1	ADC2	ETHERNET	ADC3	LPTIM2	LPTIM3	LPTIM4	LPTIM5	COMP1	COMP2
D2 ドメイン	APB1	TIM2	-	S	S	-	-	-	S	-	A	S	S	S	-	-	-	S	S	S	A	S	-	-	-	-	I	I
		TIM3	S	-	S	S	-	-	-	-	A	S		S	-	-	S	S	S	S	A	S	-	-	-	-	I	I
		TIM4	S	S	-	S	S	-	S	-	-	S	S	S	-	-	S	-	S	S	-	S	-	-	-	-	-	-
		TIM5	-	-	-	-	S	-	S	-	-		S		-	-	-	-	-	-	-	S	-	-	-	-	-	-
		TIM6	-	-	-	-			S	-	-	-	-	-	-	-	S	S	S	S	-	S	-	-	-	-	-	-
		TIM7	-	-	-	-			S	-	-	-	-	-	-	-	S	S	-	-	-	-	-	-	-	-	-	-
		TIM13	-	-	-	-	S			-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		TIM14	-	-	-	-	S			-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		LPTIM1	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	A	A	A		A	-	-	-	-	-	-
		SPDIFRX	-	-	-	-	-	-		-	-	-	-	-	-	S	-	-	-	-	-	-	-	-	-	-	-	-
		OPAMP	-	-	-	-	-	-		-	-	-	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-
		CAN	-	-	-	A	-	-		-	-	-	-	-	-	-	-	A	-	-	A	-	-	-	-	-	-	-
	APB2	TIM1	S	S	S	S	-	-	S	-	-	-	S	S	-	-	S	S	S	S	-	S	-	-	-	-	I	I
		TIM8	S	-	S	S	-	-	S	-	-	-	-	-	-	-	S		S	S	-	S	-	-	-	-	I	I
		TIM15	-	S	-	-	-	-	S	-	-	S	-	-	-	-		S	S	S	-	S	-	-	-	-	I	I
		TIM16	-	-	-	-	-	-	-	-	-	-	-	S	-	-	S	S	-	-	-	-	-	-	-	-	-	-
		TIM17	-	-	-	-	-	-	-	-	-	-	-	S	-	-		S	-	-	-	-	-	-	-	-	-	-
		SAI1	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	-	-	-
		SAI2	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	-	-
		DFSDM1	-	-	-	-	-	-	-	-	-	B	B	B	B	B	-	-	-	-	-	-	-	-	-	-	-	-
		HRTIM	-	-	-	-	-	-	A	-	A	-	-	-	-	-	S	-	A	A	A	A	-	-	-	-	-	-
	AHB1	ADC1	-	-	-	-	-	-	-	-	-	A	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-
		ADC2	-	-	-	-	-	-	-	-	-	-	A	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-
		ETH	A	A	-	-	-	-	-	-	A	-	-	-	-	-	-		-	-	-	-	-	-	-	-	-	-
		USB1	A			A	-	-		A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		USB2	A			A	-	-		A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

1. 表内の文字は*セクション 13.1.2：接続の概要*で説明している接続タイプに対応しています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

表 87. ペリフェラル相互接続マトリックス（D3 ドメイン）(1) (2)

転送元			転送先																								
			D2 ドメイン																				D3 ドメイン AHB4 APB4				
			APB1									APB2							AHB1								
			TIM2	TIM3	TIM4	TIM5	TIM12	LPTIM1	DAC	CRS	CAN	TIM1	TIM8	TIM15	TIM16	TIM17	DFSDM1	HRTIM	ADC1	ADC2	ETHERNET	ADC3	LPTIM2	LPTIM3	LPTIM4	LPTIM5
D3 ドメイン	APB4	EXTI	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	-	A	A	-	-	-	-	-	-	-
		LPTIM2	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	A	A	A	-	A	-	A	A	A
		LPTIM3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	A	A	-	A	-	-	A	A
		LPTIM4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	A
		LPTIM5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	-
		COMP1	A	A	-	-	-	A	-	-	-	A/B	A/B	B	B	B		A/B	-	-	-	-	A	-	-	-
		COMP2	A	A	-	-	-	A	-	-	-	A/B	A/B	B	B	B		A/B	-	-	-	-	A	-	-	-
		SAI4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	A
		RTC	-	-	-	-	-	A	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	-	-	-
	AHB4	ADC3	-	-	-	-	-	-	-	-	-	A	A	-	-	-	-	-	-	-	-	-	-	-	-	-
		RCC	A	-	-	-	-	-	-	A	-	-	-	A	A	A	-	-	-	-	-	-	-	-	-	-

1. 表内の文字は *セクション 13.1.2：接続の概要* で説明している接続タイプに対応しています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

表 88. ペリフェラル相互接続マトリックスの詳細(1)

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	ITR0	TIM2	APB1	D2	S	-
		TIM8	TRGO	ITR1				S	-
	APB1	TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
	AHB1	ETH	PPS	ITR4				S	-
		USB1	SOF	ITR5				S	-
		USB2	SOF	ITR6				S	-
D3	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
		RCC	lse_ck	ETR3				A	-
D2	APB2	SAI1	SAI1_FS_A	ETR4				A	-
		SAI1	SAI1_FS_B	ETR5				A	-
D3	APB4	COMP1	comp1_out	TI4_1				I	-
		COMP2	comp2_out	TI4_2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	TI4_3				I	-
D2	APB2	TIM1	TRGO	ITR0	TIM3	APB1	D2	S	-
	APB1	TIM2	TRGO	ITR1				S	-
	APB2	TIM15	TRGO	ITR2				S	-
	APB1	TIM4	TRGO	ITR3				S	-
	AHB1	ETH	PPS	ITR4				S	-
D3	APB4	COMP1	comp1_out	ETR1				I	-
		COMP1	comp1_out	TI1_1				I	-
		COMP2	comp2_out	TI1_2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	TI1_3				I	-
D2	APB2	TIM1	TRGO	ITR0	TIM4	APB1	D2	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM3	TRGO	ITR2				S	-
	APB2	TIM8	TRGO	ITR3				S	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	ITR0	TIM5	APB1	D2	S	-
		TIM8	TRGO	ITR1				S	-
	APB1	TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
		CAN	SOC	ITR6				S	-
	AHB1	USB1	SOF	ITR7				S	-
		USB2	SOF	ITR8				S	-
	APB2	SAI2	SAI2_FS_A	ETR1				A	-
		SAI2	SAI2_FS_B	ETR2				A	-
	APB1	CAN	TMP	TI1_1				A	-
		CAN	RTP	TI1_2				A	-
D2	APB1	TIM4	TRGO	ITR0	TIM12	APB1	D2	S	-
		TIM5	TRGO	ITR1				S	-
		TIM13	OC1	ITR2				S	-
		TIM14	OC1	ITR3				S	-
	AHB1	USB1	SOF	crs_sync2	CRS	APB1	D2	A	-
		USB2	SOF	crs_sync0	CRS	APB1	D2	A	-
	AHB2	USB1	SOF	crs_sync2	CRS	APB1	D2	A	-
		USB2	SOF	crs_sync0	CRS	APB1	D2	A	-
D3	AHB4	RCC	lse_ck	crs_sync1	CRS	APB1	D2	A	

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM15	TRGO	ITR0	TIM1	APB2	D2	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
D3	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
D2	AHB1	ADC1	adc1_awd1	ETR3				A	-
		ADC1	adc1_awd2	ETR4				A	-
		ADC1	adc1_awd3	ETR5				A	-
D3	AHB4	ADC3	adc3_awd1	ETR6				A	-
		ADC3	adc3_awd2	ETR7				A	-
		ADC3	adc3_awd3	ETR8				A	-
	APB4	COMP1	comp1_out	TI1_1				I	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
D2	APB2	DFSDM1	dfsdm1_ break0	BRK_3				B	-
D3	APB4	COMP1	comp1_out	BRK2_1				B	-
		COMP2	comp2_out	BRK2_2				B	-
D2	APB2	DFSDM1	dfsdm1_ break1	BRK2_3				B	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	ITR0	TIM8	APB2	D2	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM4	TRGO	ITR2				S	-
		TIM5	TRGO	ITR3				S	-
D3	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
D2	AHB1	ADC2	adc2_awd1	ETR3				A	-
		ADC2	adc2_awd2	ETR4				A	-
		ADC2	adc2_awd3	ETR5				A	-
D3	AHB4	ADC3	adc3_awd1	ETR6				A	-
		ADC3	adc3_awd2	ETR7				A	-
		ADC3	adc3_awd3	ETR8				A	-
	APB4	COMP2	comp2_out	TI1_1				I	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
D2	APB2	DFSDM1	dfsdm1_ break2	BRK_3				B	-
D3	APB4	COMP1	comp1_out	BRK2_1				B	-
		COMP2	comp2_out	BRK2_2				B	-
D2	APB2	DFSDM1	dfsdm1_ break3	BRK2_3				B	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	ITR0	TIM15	APB2	D2	S	-
	APB1	TIM3	TRGO	ITR1				S	-
	APB2	TIM16	OC1	ITR2				S	-
		TIM17	OC1	ITR3				S	-
	APB1	TIM2	CH1	TI1_1				A	-
		TIM3	CH1	TI1_2				A	-
		TIM4	CH1	TI1_3				A	-
D3	AHB4	RCC	lse_ck	TI1_4				A	-
		RCC	csi_ck	TI1_5				A	-
		RCC	MCO2	TI1_6				A	-
D2	APB1	TIM2	CH2	TI2_1				A	-
		TIM3	CH2	TI2_2				A	-
		TIM4	CH2	TI2_3				A	-
D3	APB4	COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
D2	APB2	DFSDM1	dfsdm_ break0	BRK_3				B	-
D3	AHB4	RCC	lsi_ck	TI1_1	TIM16	APB2	D2	A	-
		RCC	lse_ck	TI1_2				A	-
	APB4	RTC	WKUP_IT	TI1_3				A	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
D2	APB2	DFSDM1	dfsdm_ break1	BRK_3				B	-
D2	APB1	SPDIFRX	spdifrx_frame_ sync	TI1_1	TIM17	APB2	D2	A	-
D3	AHB4	RCC	HSE_1MHZ	TI1_2				A	-
		RCC	MCO1	TI1_3				A	-
	APB4	COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
D2	APB2	DFSDM1	dfsdm_ break2	BRK_3				B	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D3	APB4	COMP1	comp1_out	hrtim_evt11	HRTIM	APB2	D2	B	-
D2	APB2	TIM1	TRGO	hrtim_evt12				B	-
	AHB1	ADC1	adc1_awd1	hrtim_evt13				B	-
D3	APB4	COMP2	OUT	hrtim_evt21				B	-
D2	APB2	TIM2	TRGO	hrtim_evt22				B	-
	AHB1	ADC1	adc1_awd2	hrtim_evt23				B	-
	NC	NC	NC	hrtim_evt31				B	-
	APB2	TIM3	TRGO	hrtim_evt32				B	-
	AHB1	ADC1	adc1_awd3	hrtim_evt33				B	-
	APB1	OPAMP1	opamp1_out	hrtim_evt41				B	-
		TIM7	TRGO	hrtim_evt42				B	-
	AHB1	ADC2	adc2_awd1	hrtim_evt43				B	-
	NC	NC	NC	hrtim_evt51				B	-
	APB1	LPTIM1	lptim1_out	hrtim_evt52				B	-
	AHB1	ADC2	adc2_awd2	hrtim_evt53				B	-
D3	APB4	COMP1	comp1_out	hrtim_evt61				I	-
D2	APB1	TIM6	TRGO	hrtim_evt62				S	-
	AHB1	ADC2	adc2_awd3	hrtim_evt63				A	-
D3	APB4	COMP2	comp2_out	hrtim_evt71				I	-
D2	APB1	TIM7	TRGO	hrtim_evt72				S	-
	NC	NC	NC	hrtim_evt73				-	-
				hrtim_evt81				-	-
	APB1	TIM6	TRGO	hrtim_evt82				S	-
	APB1	CAN	TTCAN_TMP	hrtim_evt83				A	-
		OPAMP1	opamp1_out	hrtim_evt91				I	-
	APB2	TIM15	TRGO	hrtim_evt92				S	-
	APB1	CAN	TTCAN_RTP	hrtim_evt93				A	-
	NC	NC	NC	hrtim_ evt101					-
D3	APB4	LPTIM2	lptim2_out	hrtim_ evt102				A	-
D2	APB1	CAN	TTCAN_SOC	hrtim_ evt103				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D3	APB4	COMP1	comp1_out	hrtim_in_flt1	HRTIM	APB2	D2	B	-
		COMP2	comp2_out	hrtim_in_flt2				B	-
D2	APB2	TIM16	OC	hrtim_upd_en1				S	-
		TIM17	OC	hrtim_upd_en2				S	-
	APB1	TIM6	TRGO	hrtim_upd_en3				S	-
		TIM7	TRGO	hrtim_bm_trg				S	-
	APB2	TIM16	OC	hrtim_bm_ck1				S	-
		TIM17	OC	hrtim_bm_ck2				S	-
	APB1	TIM7	TRGO	hrtim_bm_ck3				S	-
D3	APB4	RTC	rtc_alarm_a_evt	lptim1_ext_trg0	LPTIM1	APB1	D2	A	-
		RTC	rtc_alarm_b_evt	lptim1_ext_trg1				A	-
		RTC	rtc_tamp1_evt	lptim1_ext_trg2				A	-
		RTC	rtc_tamp2_evt	lptim1_ext_trg3				A	-
		RTC	rtc_tamp3_evt	lptim1_ext_trg4				A	-
		COMP1	comp1_out	lptim1_ext_trg5				I	-
		COMP2	comp2_out	lptim1_ext_trg6				I	-
		COMP1	comp1_out	lptim1_in1_mux1				I	-
		COMP2	comp2_out	lptim1_in2_mux2				I	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D3	APB4	RTC	rtc_alarm_a_evt	lptim2_ext_trg0	LPTIM2	APB4	D3	A	-
		RTC	rtc_alarm_b_evt	lptim2_ext_trg1				A	-
		RTC	rtc_tamp1_evt	lptim2_ext_trg2				A	-
		RTC	rtc_tamp2_evt	lptim2_ext_trg3				A	-
		RTC	rtc_tamp3_evt	lptim2_ext_trg4				A	-
		COMP1	comp1_out	lptim2_ext_trg5				I	-
		COMP2	comp2_out	lptim2_ext_trg6				I	-
		COMP1	comp1_out	lptim2_in1_mux1				I	-
		COMP2	comp2_out	lptim2_in1_mux2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	lptim2_in1_mux3				I	-
		COMP2	comp2_out	lptim2_in2_mux1				I	-
D3	APB4	LPTIM2	lptim2_out	lptim3_ext_trg0	LPTIM3	APB4	D3	S	カーネルクロックソースが同じである場合
		NC	NC	lptim3_ext_trg1				-	-
		LPTIM4	lptim4_out	lptim3_ext_trg2				S	カーネルクロックソースが同じである場合
		LPTIM5	lptim5_out	lptim3_ext_trg3				S	カーネルクロックソースが同じである場合
D2	APB2	SAI1	SAI1_FS_A	lptim3_ext_trg4				A	-
		SAI1	SAI1_FS_B	lptim3_ext_trg5				A	-
		SAI1	SAI1_FS_A	lptim3_in1_mux1				A	-
		SAI1	SAI1_FS_B	lptim3_in1_mux2				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D3	APB4	LPTIM2	lptim2_out	lptim4_ext_trg0	LPTIM4	APB4	D3	S	カーネルクロックソースが同じである場合
		LPTIM3	lptim3_out	lptim4_ext_trg1				S	カーネルクロックソースが同じである場合
		NC	NC	lptim4_ext_trg2					-
		LPTIM5	lptim5_out	lptim4_ext_trg3				S	カーネルクロックソースが同じである場合
D2	APB2	SAI2	SAI2_FS_A	lptim4_ext_trg4				A	-
		SAI2	SAI2_FS_B	lptim4_ext_trg5				A	-
D3	APB4	LPTIM2	lptim2_out	lptim5_ext_trg0	LPTIM5	APB4	D3	S	カーネルクロックソースが同じである場合
		LPTIM3	lptim3_out	lptim5_ext_trg1				S	カーネルクロックソースが同じである場合
		LPTIM4	lptim4_out	lptim5_ext_trg2				S	カーネルクロックソースが同じである場合
		SAI4	SAI4_FS_A	lptim5_ext_trg3				A	-
		SAI4	SAI4_FS_B	lptim5_ext_trg4				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	dac_ch1/2_trg0	DAC チャネル 1／チャネル 2	APB1	D2	S	-
	APB1	TIM2	TRGO	dac_ch1/2_trg1				S	-
		TIM4	TRGO	dac_ch1/2_trg02				S	-
		TIM5	TRGO	dac_ch1/2_trg3				S	-
		TIM6	TRGO	dac_ch1/2_trg4				S	-
		TIM7	TRGO	dac_ch1/2_trg5				S	-
	APB2	TIM8	TRGO	dac_ch1/2_trg6				S	-
		TIM15	TRGO	dac_ch1/2_trg7				S	-
		HRTIM1	hrtim_dac_trg1	dac_ch1/2_trg8				S	-
			hrtim_dac_trg2	dac_ch1/2_trg9				S	-
	APB1	LPTIM1	lptim1_out	dac_ch1/2_trg10				S	-
		LPTIM2	lptim2_out	dac_ch1/2_trg11				S	-
D3	APB4	SYSCFG	EXTI9	dac_ch1/2_trg12				S	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	TRG0	DFSDM1	APB2	D2	S	-
		TIM1	TRGO2	TRG1				S	-
		TIM8	TRGO	TRG2				S	-
		TIM8	TRGO2	TRG3				S	-
	APB1	TIM3	TRGO	TRG4				S	-
		TIM4	TRGO	TRG5				S	-
	APB2	TIM16	OC1	TRG6				S	-
	APB1	TIM6	TRGO	TRG7				S	-
		TIM7	TRGO	TRG8				S	-
	APB2	HRTIM1	hrtim_adc_trg1	TRG9				S	-
		HRTIM1	hrtim_adc_trg3	TRG10				S	-
D3	APB4	SYSCFG	EXTI11	TRG24				A	-
		SYSCFG	EXTI15	TRG25				A	-
D2	APB1	LPTIM1	lptim1_out	TRG26				A	-
D3	APB4	LPTIM2	lptim2_out	TRG27				A	-
		LPTIM3	lptim3_out	TRG28				A	-
D2	APB2	TIM1	CC1	adc_ext_trg0	ADC1 / ADC2	AHB1	D2	S	-
		TIM1	CC2	adc_ext_trg1				S	-
		TIM1	CC3	adc_ext_trg2				S	-
	APB1	TIM2	CC2	adc_ext_trg3				S	-
		TIM3	TRGO	adc_ext_trg4				S	-
		TIM4	CC4	adc_ext_trg5				S	-
D3	APB4	SYSCFG	EXTI11	adc_ext_trg6				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM8	TRGO	adc_ext_trg7	ADC1 / ADC2	AHB1	D2	S	-
		TIM8	TRGO2	adc_ext_trg8				S	-
		TIM1	TRGO	adc_ext_trg9				S	-
		TIM1	TRGO2	adc_ext_trg10				S	-
	APB1	TIM2	TRGO	adc_ext_trg11				S	-
		TIM4	TRGO	adc_ext_trg12				S	-
		TIM6	TRGO	adc_ext_trg13				S	-
	APB2	TIM15	TRGO	adc_ext_trg14				S	-
	APB1	TIM3	CC4	adc_ext_trg15				S	-
	APB2	HRTIM1	hrtim_adc_trg1	adc_ext_trg16				A	-
		HRTIM1	hrtim_adc_trg3	adc_ext_trg17				A	-
		LPTIM1	lptim1_out	adc_ext_trg18				A	-
D3	APB4	LPTIM2	lptim2_out	adc_ext_trg19				A	-
		LPTIM3	lptim3_out	adc_ext_trg20				A	-
D2	APB2	TIM1	TRGO	adc_jext_trg0	ADC1 / ADC2	AHB1	D2	S	-
		TIM1	CC4	adc_jext_trg1				S	-
	APB1	TIM2	TRGO	adc_jext_trg2				S	-
		TIM2	CC1	adc_jext_trg3				S	-
		TIM3	CC4	adc_jext_trg4				S	-
		TIM4	TRGO	adc_jext_trg5				S	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D3	APB4	SYSCFG	EXTI15	adc_jext_trg6	ADC1 / ADC2	AHB1	D2	A	-
D2	APB2	TIM8	CC4	adc_jext_trg7				S	-
		TIM1	TRGO2	adc_jext_trg8				S	-
		TIM8	TRGO	adc_jext_trg9				S	-
		TIM8	TRGO2	adc_jext_trg10				S	-
	APB1	TIM3	CC3	adc_jext_trg11				S	-
		TIM3	TRGO	adc_jext_trg12				S	-
		TIM3	CC1	adc_jext_trg13				S	-
		TIM6	TRGO	adc_jext_trg14				S	-
	APB2	TIM15	TRGO	adc_jext_trg15				S	-
		HRTIM1	hrtim_adc_trg2	adc_jext_trg16				A	-
		HRTIM1	hrtim_adc_trg4	adc_jext_trg17				A	-
	APB1	LPTIM1	lptim1_out	adc_jext_trg18				A	-
D3	APB4	LPTIM2	lptim2_out	adc_jext_trg19				A	-
		LPTIM3	lptim2_out	adc_jext_trg20				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	CC1	EXT0	ADC3	AHB4	D3	S	-
		TIM1	CC2	EXT1				S	-
		TIM1	CC3	EXT2				S	-
	APB1	TIM2	CC2	EXT3				S	-
		TIM3	TRGO	EXT4				S	-
		TIM4	CC4	EXT5				S	-
D3	APB4	SYSCFG	EXTI11	EXT6				A	-
D2	APB2	TIM8	TRGO	EXT7				S	-
		TIM8	TRGO2	EXT8				S	-
		TIM1	TRGO	EXT9				S	-
		TIM1	TRGO2	EXT10				S	-
	APB1	TIM2	TRGO	EXT11				S	-
		TIM4	TRGO	EXT12				S	-
		TIM6	TRGO	EXT13				S	-
	APB2	TIM15	TRGO	EXT14				S	-
	APB1	TIM3	CC4	EXT15				S	-
	APB2	HRTIM1	hrtim_adc_trg1	EXT16				A	-
		HRTIM1	hrtim_adc_trg3	EXT17				A	-
		LPTIM1	lptim1_out	EXT18				A	-
D3	APB4	LPTIM2	lptim2_out	EXT19				A	-
		LPTIM3	lptim3_out	EXT20				A	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB2	TIM1	TRGO	JEXT0	ADC3	AHB4	D3	SI	-
		TIM1	CC4	JEXT1				S	-
	APB1	TIM2	TRGO	JEXT2				S	-
		TIM2	CC1	JEXT3				S	-
		TIM3	CC4	JEXT4				S	-
		TIM4	TRGO	JEXT5				S	-
D3	APB4	SYSCFG	EXTI15	JEXT6				A	-
D2	APB2	TIM8	CC4	JEXT7				S	-
		TIM1	TRGO2	JEXT8				S	-
		TIM8	TRGO	JEXT9				S	-
		TIM8	TRGO2	JEXT10				S	-
	APB1	TIM3	CC3	JEXT11				S	-
		TIM3	TRGO	JEXT12				S	-
		TIM3	CC1	JEXT13				S	-
		TIM6	TRGO	JEXT14				S	-
	APB2	TIM15	TRGO	JEXT15				S	-
		HRTIM1	hrtim_adc_trg2	JEXT16				A	-
		HRTIM1	hrtim_adc_trg4	JEXT17				A	-
	APB1	LPTIM1	OUT	JEXT18				A	-
D3	APB4	LPTIM2	OUT	JEXT19				A	-
		LPTIM3	OUT	JEXT20				A	-
D2	APB2	TIM1	OC5	comp_blk1	COMP1 / COMP2	APB4	D3	I	-
		TIM1	OC3	comp_blk2				I	-
	APB1	TIM3	OC3	comp_blk3				I	-
		TIM3	OC4	comp_blk4				I	-
	APB2	TIM8	OC5	comp_blk5				I	-
		TIM15	OC1	comp_blk6				I	-

表 88. ペリフェラル相互接続マトリックスの詳細(1)（続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
D2	APB1	TIM2	TRGO	SWT0	FDCAN	APB1	D2	A	-
		TIM3	TRGO	SWT1				A	-
	AHB1	ETH	PPS	SWT2				A	-
	APB2	HRTIM1	hrtim_dac_trg1	SWT3				A	-
	APB1	TIM2	TRGO	EVT0				A	-
		TIM3	TRGO	EVT1				A	-
	AHB1	ETH	PPS	EVT2				A	-
	APB2	HRTIM1	hrtim_dac_trg1	EVT3				A	-
	APB1	TIM2	TRGO	PTP0	ETH	AHB1	D2	A	-
		TIM3	TRGO	PTP1				A	-
	APB2	HRTIM1	hrtim_dac_trg2	PTP2				A	-
	APB1	CAN	TMP	PTP3				A	-

1. 表内の文字は*セクション 13.1.2：接続の概要*で説明している接続タイプに対応しています。
2. comp1_out および comp2_out は OR ゲートの入力に接続されます。OR ゲートの出力は lptim2_in1_mux3 入力に接続されます。

13.2 低電力モードからのウェイクアップ

拡張割り込みとイベントコントローラモジュール（EXTI）により、システムを STOP モードから、および／または CPU を CStop モードからウェイクアップすることができます。ウェイクアップイベントはペリフェラルから受信します。

これらのイベントは、設定可能なイベント（**C**）またはダイレクトイベント（**D**）として EXTI によって処理されます。*表 89*の*タイプ*列を参照してください。詳細については、*セクション 20：拡張割り込み／イベントコントローラ（EXTI）*を参照してください。

3 つのタイプのペリフェラル出力信号が、EXTI 入力イベントに接続されています。

- ウェイクアップ信号。これらの信号は、バスインタフェースクロックなしでペリフェラルで生成できます。*表 89* では xxx_wkup と呼ばれます。一部のペリフェラルにはこの機能はありません。
- 割り込み信号。ペリフェラルバスインタフェースクロックが実行中の場合にのみ生成できます。これらの割り込み信号は、通常 CPU の NVIC に直接接続されています。これらは、xxx_it と呼ばれます。
- 信号。（例ペリフェラルによって生成されたパルス）ペリフェラルが信号を生成した後で、ペリフェラルレベルで必要になるアクション（フラグのクリア）はありません。

各 EXTI 入力イベントには、異なるウェイクアップ機能や使用可能なターゲットがあります（*表 89* の*ターゲット*列を参照）。

- CPU ウェイクアップ（**CPU**）：CPU をウェイクアップするために、入力イベントを有効にできます。
- 自律 RUN モード（**任意**）の CPU および D3 ドメインウェイクアップ：CPU または D3 ドメインをウェイクアップするために、入力イベントを自律 RUN モードフェーズでのみ有効にできます。

表 89. EXTI ウェイクアップ入力(1)

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
D3	APB4	SYSCFG	exti0_wkup	WKUP0	EXTI	C	任意	-
			exti1_wkup	WKUP1				-
			exti2_wkup	WKUP2				-
			exti3_wkup	WKUP3				-
			exti4_wkup	WKUP4				-
			exti5_wkup	WKUP5				-
			exti6_wkup	WKUP6				-
			exti7_wkup	WKUP7				-
			exti8_wkup	WKUP8				-
			exti9_wkup	WKUP9				-
			exti10_wkup	WKUP10				-
			exti11_wkup	WKUP11				-
			exti12_wkup	WKUP12				-
			exti13_wkup	WKUP13				-
			exti14_wkup	WKUP14				-
			exti15_wkup	WKUP15				-
D3	AHB4	PWR	pvd_avd_wkup	WKUP16		C	CPU	-
D3	APB4	RTC	ALARMS	WKUP17		D	CPU	-
D3	APB4	RTC	TAMPER TIMESTAMP	WKUP18		C	CPU	-
D3	AHB4	RCC	CSS_LSE					-
D3	APB4	RTC	WKUP	WKUP19		C	任意	-
D3	APB4	COMP1	comp1_out	WKUP20		C	任意	-
D3	APB4	COMP2	comp2_out	WKUP21		C	任意	-
D2	APB1	I2C1	i2c1_wkup	WKUP22		C	CPU	-
D2	APB1	I2C2	i2c2_wkup	WKUP23		D	CPU	-
D2	APB1	I2C3	i2c3_wkup	WKUP24		D	CPU	-
D2	APB1	I2C4	i2c4_wkup	WKUP25		D	任意	-
D2	APB2	USART1	usart1_wkup	WKUP26		D	CPU	-
D2	APB1	USART2	usart2_wkup	WKUP27		D	CPU	-

表 89. EXTI ウェイクアップ入力(1)（続き）

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
D2	APB1	USART3	usart3_wkup	WKUP28	EXTI	D	CPU	-
D2	APB2	USART6	usart6_wkup	WKUP29		D	CPU	-
D2	APB1	UART4	uart4_wkup	WKUP30		D	CPU	-
D2	APB1	UART5	uart5_wkup	WKUP31		D	CPU	-
D2	APB1	UART7	uart7_wkup	WKUP32		D	CPU	-
D2	APB1	UART8	uart8_wkup	WKUP33		D	CPU	-
D3	APB4	LPUART	lpuart_rx_wkup	WKUP34		D	任意	-
D3	APB4	LPUART	lpuart_tx_wkup	WKUP35		D	任意	-
D2	APB2	SPI1	spi1_wkup	WKUP36		D	CPU	-
D2	APB1	SPI2	spi2_wkup	WKUP37		D	CPU	-
D2	APB1	SPI3	spi3_wkup	WKUP38		D	CPU	-
D2	APB2	SPI4	spi4_wkup	WKUP39		D	CPU	-
D2	APB2	SPI5	spi5_wkup	WKUP40		D	CPU	-
D3	APB4	SPI6	spi6_wkup	WKUP41		D	任意	-
D2	APB1	MDIOS	mdios_wkup	WKUP42		D	CPU	-
D2	AHB1	USB1	usb1_wkup	WKUP43		D	CPU	-
D2	AHB1	USB2	usb2_wkup	WKUP44		D	CPU	-
-	-	NC	NC	WKUP45		-	-	-
D2	APB1	LPTIM1	lptim1_wkup	WKUP47		D	CPU	-
D3	APB4	LPTIM2	lptim2_wkup	WKUP48		D	任意	-
D3	APB4	LPTIM2	lptim2_out	WKUP49		C	任意	(2)
D3	APB4	LPTIM3	lptim3_wkup	WKUP50		D	任意	-
D3	APB4	LPTIM3	lptim3_out	WKUP51		C	任意	(2)
D3	APB4	LPTIM4	lptim4_wkup	WKUP52		D	任意	-
D3	APB4	LPTIM5	lptim5_wkup	WKUP53		D	任意	-
D2	APB1	SWPMI	swpmi_wkup	WKUP54		D	CPU	-

表 89. EXTI ウェイクアップ入力(1)（続き）

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
D3	AHB4	PWR	pwr_wkup1_wkup	WKUP55	EXTI	D	CPU	-
			pwr_wkup2_wkup	WKUP56				-
			pwr_wkup3_wkup	WKUP57				-
			pwr_wkup4_wkup	WKUP58				-
			pwr_wkup5_wkup	WKUP59				-
			pwr_wkup6_wkup	WKUP60				-
D3	AHB4	RCC	rcc_it	WKUP61		D	CPU	-
D3	APB4	I2C4	i2c4_ev_it	WKUP62		D	CPU	(1)
		I2C4	i2c4_err_it	WKUP63		D	CPU	(1)
D3	APB4	LPUART1	lpuart1_it	WKUP64		D	CPU	(1)
D3	APB4	SPI6	spi6_it	WKUP64		D	CPU	(1)
D3	AHB4	BDMA	bdma_ch0_it	WKUP66		D	CPU	(1)
			bdma_ch1_it	WKUP67		D	CPU	(1)
			bdma_ch2_it	WKUP68		D	CPU	(1)
			bdma_ch3_it	WKUP69		D	CPU	(1)
			bdma_ch4_it	WKUP70		D	CPU	(1)
			bdma_ch5_it	WKUP71		D	CPU	(1)
			bdma_ch6_it	WKUP72		D	CPU	(1)
			bdma_ch7_it	WKUP73		D	CPU	(1)
D3	AHB4	DMAMUX2	dmamux2_it	WKUP74			CPU	(1)
D3	AHB4	ADC3	adc3_it	WKUP75		D	CPU	(1)
D3	APB4	SAI4	sai4_gbl_it	WKUP76		D	CPU	(1)
D3	AHB4	HSEM	hsem_int_it	WKUP77		D	CPU	(1)
-	-	NC	NC	WKUP81		-	-	-
D1	APB3	WWDG1	wwdg1_out_rst	WKUP82		C	CPU	(1)
-	-	NC	NC	WKUP83		-	-	-
D1	APB1	CEC	cec_wkup	WKUP85		C	CPU	-
D2	AHB1	ETH	eth	WKUP86		C	CPU	-
D3	AHB4	RCC	hse_css_rcc_wkup	WKUP87		D	CPU	-

1. ソースペリフェラルでは、イベントを生成するためにバスクロックが必要です。これは、D3 ドメインの PCLK4 または HCLK4、または D1 ドメインの PCLK3 のいずれかです。

2. ソースペリフェラル信号は、NVIC に接続されていません。

自律 RUN モードの D3 ドメインをウェイクアップできる拡張割り込みとイベントコントローラモジュール（EXTI）のイベント入力には、4 つの異なる入力ソースでクリアできるペンディングリクエストロジックがあります（*表 90*）。詳細については、*セクション 20 : 拡張割り込み／イベントコントローラ（EXTI）*を参照してください。

表 90. EXTI ペンディングリクエストクリア入力

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D3	AHB4	DMAMUX2	dmamux2_evt6	PRC0	EXTI	APB4	D3	-
			dmamux2_evt7	PRC1				-
	APB4	LPTIM4	lptim4_out	PRC2				-
		LPTIM5	lptim5_out	PRC3				-

13.3 DMA

D1 ドメインでは、MDMA によってメモリによるデータ転送が可能になります。*セクション 13.3.1* で説明している接続に応じて、ソフトウェアまたはハードウェアによってトリガできます。

D2 ドメインの DMA マルチプレクサ（DMAMUX1）により、任意のペリフェラル DMA リクエストを、DMA1 または DMA2 の任意のストリームに配置することができます。加えて、DMAMUX ではさらに 2 つの機能を提供しています。

- ペリフェラル DMA リクエストを、タイマ、外部ピン、または別ストリームのDMA 転送の転送終了と同期できます。
- DMA リクエストは、DMAMUX1 自体によって、DMAストリーム上で生成できます。このイベントは、タイマ、外部ピンイベント、または別のストリームの完了している DMA 転送によってトリガできます。生成する DMA リクエスト数は設定可能です。

DMAMUX1 および DMA1/DMA2 の接続については、*セクション 17：DMA リクエストマルチプレクサ（DMAMUX）*、*セクション 15：ダイレクトメモリアクセスコントローラ（DMA1、DMA2）*、および*セクション 16：基本ダイレクトメモリアクセスコントローラ（BDMA）*を参照してください。

D3 ドメインの DMA マルチプレクサ（DMAMUX2）には DMAMUX1 と同じ機能があり、基本の DMA（BDMA）に接続されています。

DMAMUX2 および BDMA の接続については、*セクション 13.3.3：DMAMUX2、BDMA （D3 ドメイン）*を参照してください。詳細については、*セクション 13.3.3：DMAMUX2、BDMA （D3 ドメイン）*および *セクション 16：基本ダイレクトメモリアクセスコントローラ（BDMA）*を参照してください。

13.3.1 MDMA （D1 ドメイン）

表 91. MDMA

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D2	AHB1	DMA1	dma1_tcif0	mdma_str0	MDMA	AXI	D1	DMA1 ストリーム 0 転送完了
			dma1_tcif1	mdma_str1				DMA1 ストリーム 1 転送完了
			dma1_tcif2	mdma_str2				DMA1 ストリーム 2 転送完了
			dma1_tcif3	mdma_str3				DMA1 ストリーム 3 転送完了
			dma1_tcif4	mdma_str4				DMA1 ストリーム 4 転送完了
			dma1_tcif5	mdma_str5				DMA1 ストリーム 5 転送完了フラグ
			dma1_tcif6	mdma_str6				DMA1 ストリーム 6 転送完了
			dma1_tcif7	mdma_str7				DMA1 ストリーム 7 転送完了
D2	AHB1	DMA2	dma2_tcif0	mdma_str8				DMA2 ストリーム 0 転送完了
			dma2_tcif1	mdma_str9				DMA2 ストリーム 1 転送完了
			dma2_tcif2	mdma_str10				DMA2 ストリーム 2 転送完了
			dma2_tcif3	mdma_str11				DMA2 ストリーム 3 転送完了
			dma2_tcif4	mdma_str12				DMA2 ストリーム 4 転送完了
			dma2_tcif5	mdma_str13				DMA2 ストリーム 5 転送完了
			dma2_tcif6	mdma_str14				DMA2 ストリーム 6 転送完了
			dma2_tcif7	mdma_str15				DMA2 ストリーム 7 転送完了
D1	APB3	LTDC	ltdc_li_it	mdma_str16				LTDC ライン割り込み
D1	AHB3	JPEG	jpeg_ift_trg	mdma_str17	MDMA	AXI	D1	JPEG 入力 FIFO 閾値
			jpeg_ifnt_trg	mdma_str18				JPEG 入力 FIFO はフルではありません。
			jpeg_oft_trg	mdma_str19				JPEG 出力 FIFO 閾値
			jpeg_ofne_trg	mdma_str20				JPEG 出力 FIFO ノットエンプティ
			jpeg_oec_trg	mdma_str21				JPEG 変換の終了
D1	AHB3	QUADSPI	quadspi_ft_trg	mdma_str22				QUADSPI FIFO 閾値
			quadspi_tc_trg	mdma_str23				QUADSPI 転送完了
D1	AHB3	DMA2D	dma2d_clut_trg	mdma_str24				DMA2D CLUT 転送完了
			dma2d_tc_trg	mdma_str25				DMA2D 転送完了
			dma2d_tw_trg	mdma_str26				DMA2D 転送ウォーターマーク
D1	AHB3	SDMMC1	sdmmc1_dataend_trg	mdma_str29				データ終了

13.3.2 DMAMUX1、DMA1、および DMA2 （D2 ドメイン）

表 92. DMAMUX1、DMA1、および DMA2 接続(1)

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D3	AHB4	dmamux1 内部（リクエストジェネレータ）		dmamux1_req_in0	DMAMUX1	AHB1	D2	リクエスト
				dmamux1_req_in1				
				dmamux1_req_in2				
				dmamux1_req_in3				
				NC				
				NC				
				NC				
				NC				
D2	AHB1	ADC1	adc1_dma	dmamux1_req_in8				
D2	AHB1	ADC2	adc2_dma	dmamux1_req_in9				
D2	APB2	TIM1	tim1_ch1_dma	dmamux1_req_in10				
			tim1_ch2_dma	dmamux1_req_in11				
			tim1_ch3_dma	dmamux1_req_in12				
			tim1_ch4_dma	dmamux1_req_in13				
			tim1_up_dma	dmamux1_req_in14				
			tim1_trig_dma	dmamux1_req_in15				
			tim1_com_dma	dmamux1_req_in16				
D2	APB1	TIM2	tim2_ch1_dma	dmamux1_req_in17				
			tim2_ch2_dma	dmamux1_req_in18				
			tim2_ch3_dma	dmamux1_req_in19				
			tim2_ch4_dma	dmamux1_req_in20				
			tim2_up_dma	dmamux1_req_in21				
D2	APB1	TIM3	tim3_ch1_dma	dmamux1_req_in22				
			tim3_ch2_dma	dmamux1_req_in23				
			tim3_ch3_dma	dmamux1_req_in24				
			tim3_ch4_dma	dmamux1_req_in25				
			tim3_up_dma	dmamux1_req_in26				
			tim3_trig_dma	dmamux1_req_in27				
D2	APB1	TIM4	tim4_ch1_dma	dmamux1_req_in28				
			tim4_ch2_dma	dmamux1_req_in29				
			tim4_ch3_dma	dmamux1_req_in30				
			tim4_up_dma	dmamux1_req_in31				

表 92. DMAMUX1、DMA1、および DMA2 接続(1)（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D2	APB1	I2C1	i2c1_rx_dma	dmamux1_req_in32	DMAMUX1	AHB1	D2	リクエスト
			i2c1_tx_dma	dmamux1_req_in33				
D2	APB1	I2C2	i2c2_rx_dma	dmamux1_req_in34				
			i2c2_tx_dma	dmamux1_req_in35				
D2	APB2	SPI1	spi1_rx_dma	dmamux1_req_in36				
			spi1_tx_dma	dmamux1_req_in37				
D2	APB1	SPI2	spi2_rx_dma	dmamux1_req_in38				
			spi2_tx_dma	dmamux1_req_in39				
D2	APB2	USART1	usart1_rx_dma	dmamux1_req_in40				
			usart1_tx_dma	dmamux1_req_in41				
D2	APB1	USART2	usart2_rx_dma	dmamux1_req_in42				
			usart2_tx_dma	dmamux1_req_in43				
D2	APB1	USART3	usart3_rx_dma	dmamux1_req_in44				
			usart3_tx_dma	dmamux1_req_in45				
D2	APB2	TIM8	tim8_ch1_dma	dmamux1_req_in46				
			tim8_ch2_dma	dmamux1_req_in47				
			tim8_ch3_dma	dmamux1_req_in48				
			tim8_ch4_dma	dmamux1_req_in49				
			tim8_up_dma	dmamux1_req_in50				
			tim8_trig_dma	dmamux1_req_in51				
			tim8_com_dma	dmamux1_req_in52				
-	-	NC	NC	NC				
D1	APB1	TIM3	tim5_ch1_dma	dmamux1_req_in54				
			tim5_ch2_dma	dmamux1_req_in55				
			tim5_ch3_dma	dmamux1_req_in56				
			tim5_ch4_dma	dmamux1_req_in57				
			tim5_up_dma	dmamux1_req_in58				
			tim5_trig_dma	dmamux1_req_in59				
D2	APB1	SPI3	spi3_rx_dma	dmamux1_req_in60				
			spi3_tx_dma	dmamux1_req_in61				
D1	APB1	UART4	uart4_rx_dma	dmamux1_req_in62				
			usrt4_tx_dma	dmamux1_req_in63				

表 92. DMAMUX1、DMA1、および DMA2 接続(1)（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D1	APB1	UART5	uart5_rx_dma	dmamux1_req_in64	DMAMUX1	AHB1	D2	リクエスト
			uart5_tx_dma	dmamux1_req_in65				
D2	APB1	DAC1	dac_ch1_dma	dmamux1_req_in66				
D2	APB1	DAC2	dac_ch2_dma	dmamux1_req_in67				
D2	APB1	TIM6	tim6_up_dma	dmamux1_req_in68				
D2	APB1	TIM7	tim7_up_dma	dmamux1_req_in69				
D2	APB2	USART6	usart6_rx_dma	dmamux1_req_in70				
			usart6_tx_dma	dmamux1_req_in71				
D2	APB1	I2C3	i2c3_rx_dma	dmamux1_req_in72				
			i2c3_tx_dma	dmamux1_req_in73				
D2	AHB2	DCMI	dcmi_dma	dmamux1_req_in74				
D2	AHB2	CRYP	cryp_in_dma	dmamux1_req_in75				
			cryp_out_dma	dmamux1_req_in76				
D2	AHB2	HASH	hash_in_dma	dmamux1_req_in77				
D2	APB1	UART7	uart7_rx_dma	dmamux1_req_in78				
			usrt7_tx_dma	dmamux1_req_in79				
D2	APB1	UART8	uart8_rx_dma	dmamux1_req_in80				
			uart8_tx_dma	dmamux1_req_in81				
D2	APB2	SPI4	spi4_rx_dma	dmamux1_req_in82				
			spi4_tx_dma	dmamux1_req_in83				
D2	APB2	SPI5	spi5_rx_dma	dmamux1_req_in84				
			spi5_tx_dma	dmamux1_req_in85				
D2	APB2	SAI1	sai1_a_dma	dmamux1_req_in86				
			sai1_b_dma	dmamux1_req_in87				
D2	APB2	SAI2	sai2_a_dma	dmamux1_req_in88				
			sai2_b_dma	dmamux1_req_in89				
D2	APB1	SWPMI	swpmi_rx_dma	dmamux1_req_in90				
			swpmi_tx_dma	dmamux1_req_in91				
D2	APB1	SPDIFRX	spdifrx_dt_dma	dmamux1_req_in92				
			spdifrx_cs_dma	dmamux1_req_in93				

表 92. DMAMUX1、DMA1、および DMA2 接続(1)（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D2	APB2	HRTIM1	hrtim_dma1	dmamux1_req_in94	DMAMUX1	AHB1	D2	リクエスト
			hrtim_dma2	dmamux1_req_in95				
			hrtim_dma3	dmamux1_req_in96				
			hrtim_dma4	dmamux1_req_in97				
			hrtim_dma5	dmamux1_req_in98				
			hrtim_dma6	dmamux1_req_in99				
D2	APB2	DFSDM1	dfsdm1_dma0	dmamux1_req_in100				
			dfsdm1_dma1	dmamux1_req_in101				
			dfsdm1_dma2	dmamux1_req_in102				
			dfsdm1_dma3	dmamux1_req_in103				
D2	APB2	TIM15	tim15_ch1_dma	dmamux1_req_in104				
			tim15_up_dma	dmamux1_req_in105				
			tim15_trig_dma	dmamux1_req_in106				
			tim15_com_dma	dmamux1_req_in107				
D2	APB2	TIM16	tim16_ch1_dma	dmamux1_req_in108				
			tim16_up_dma	dmamux1_req_in109				
D2	APB2	TIM17	tim17_ch1_mda	dmamux1_req_in110				
			tim17_up_dma	dmamux1_req_in111				
D2	APB2	SAI3	sai3_a_dma	dmamux1_req_in112				
			sai3_b_dma	dmamux1_req_in113				
D3	AHB4	ADC3	adc3_dma	dmamux1_req_in114				
D2	AHB1	DMAMUX1	dmamux1_evt0	dmamux1_gen0	DMAMUX1	AHB1	D2	リクエスト生成
			dmamux1_evt1	dmamux1_gen1				
			dmamux1_evt2	dmamux1_gen2				
D2	APB1	LPTIM1	lptim1_out	dmamux1_gen3				
D2	APB1	LPTIM2	lptim2_out	dmamux1_gen4				
D2	APB1	LPTIM3	lptim3_out	dmamux1_gen5				
D3	APB4	EXTI	exti_exti0_it	dmamux1_gen6				
D2	APB1	TIM12	tim12_trgo	dmamux1_gen7				

表 92. DMAMUX1、DMA1、および DMA2 接続(1)（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D2	AHB1	DMAMUX1	dmamux1_evt0	dmamux1_trg0	DMAMUX1	AHB1	D2	トリガ
			dmamux1_evt1	dmamux1_trg1				
			dmamux1_evt2	dmamux1_trg2				
D2	APB1	LPTIM1	lptim1_out	dmamux1_trg3				
D2	APB1	LPTIM2	lptim2_out	dmamux1_trg4				
D2	APB1	LPTIM3	lptim3_out	dmamux1_trg5				
D3	APB4	EXTI	exti_exti0_it	dmamux1_trg6				
D2	APB1	TIM12	tim12_trgo	dmamux1_trg7				
D2	AHB1	DMAMUX1	dmamux1_req_out0	dma1_str0	DMA1	AHB1	D2	リクエスト出力
			dmamux1_req_out1	dma1_str1				
			dmamux1_req_out2	dma1_str2				
			dmamux1_req_out3	dma1_str3				
			dmamux1_req_out4	dma1_str4				
			dmamux1_req_out5	dma1_str5				
			dmamux1_req_out6	dma1_str6				
			dmamux1_req_out7	dma1_str7				
			dmamux1_req_out8	dma2_str0	DMA2	AHB1	D2	
			dmamux1_req_out9	dma2_str1				
			dmamux1_req_out10	dma2_str2				
			dmamux1_req_out11	dma2_str3				
			dmamux1_req_out12	dma2_str4				
			dmamux1_req_out13	dma2_str5				
			dmamux1_req_out14	dma2_str6				
			dmamux1_req_out15	dma2_str7				

1. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

13.3.3 DMAMUX2、BDMA （D3 ドメイン）

表 93. DMAMUX2 および BDMA 接続

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D3	AHB4	dmamux2 内部（リクエストジェネレータ）		dmamux2_req_in0	DMAMUX2	AHB4	D3	リクエスト
				dmamux2_req_in1				
				dmamux2_req_in2				
				dmamux2_req_in3				
				NC				
				NC				
				NC				
				NC				
D3	APB4	LPUART	dma_rx_lpuart	dmamux2_req_in8				
			dma_tx_lpuart	dmamux2_req_in9				
D3	APB4	SPI6	dma_rx_spi6	dmamux2_req_in10				
			dma_tx_spi6	dmamux2_req_in11				
D2	APB1	I2C4	dma_rx_i2c4	dmamux2_req_in12				
			dma_tx_i2c4	dmamux2_req_in13				
D3	APB4	SAI4	dma_a_sai4	dmamux2_req_in14				
			dma_b_sai4	dmamux2_req_in15				
D3	APB4	ADC3	dma_adc3	dmamux2_req_in16				

表 93. DMAMUX2 および BDMA 接続（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D3	AHB4	DMAMUX2	dmamux2_evt0	dmamux2_gen0	DMAMUX2	AHB4	D3	リクエスト生成
			dmamux2_evt1	dmamux2_gen1				
			dmamux2_evt2	dmamux2_gen2				
			dmamux2_evt3	dmamux2_gen3				
			dmamux2_evt4	dmamux2_gen4				
			dmamux2_evt5	dmamux2_gen5				
			dmamux2_evt6	dmamux2_gen6				
D3	APB4	EXTI	it_exti_rx_lpuart	dmamux2_gen7				
			it_exti_tx_lpuart	dmamux2_gen8				
			it_exti_wkup_lptim2	dmamux2_gen9				
			it_exti_out_lptim2	dmamux2_gen10				
			it_exti_wkup_lptim3	dmamux2_gen11				
			it_exti_out_lptim3	dmamux2_gen12				
			it_exti_wkup_lptim4	dmamux2_gen13				
			it_exti_wkup_lptim5	dmamux2_gen14				
			it_exti_wkup_i2c4	dmamux2_gen15				
			it_exti_wkup_spi6	dmamux2_gen16				
			it_exti_out_comp1	dmamux2_gen17				
			it_exti_out_comp2	dmamux2_gen18				
			it_exti_wkup_rtc	dmamux2_gen19				
			it_exti_exti0_syscfg	dmamux2_gen20				
			it_exti_exti2_syscfg	dmamux2_gen21				
D3	APB4	I2C4	it_evt_i2c4	dmamux2_gen22				
D3	APB4	SPI6	it_spi6	dmamux2_gen23				
D3	APB4	LPUART	it_tx_lpuart1	dmamux2_gen24				
			it_rx_lpuart1	dmamux2_gen25				
D3	AHB4	ADC3	it_adc3	dmamux2_gen26				
			out_awd1_adc3	dmamux2_gen27				
D3	AHB4	BDMA	it_ch0_bdma	dmamux2_gen28				
			it_ch1_bdma	dmamux2_gen29				

表 93. DMAMUX2 および BDMA 接続（続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
D3	AHB4	DMAMUX2	dmamux2_evt0	dmamux2_trg0	DMAMUX2	AHB4	D3	トリガ
			dmamux2_evt1	dmamux2_trg1				
			dmamux2_evt2	dmamux2_trg2				
			dmamux2_evt3	dmamux2_trg3				
			dmamux2_evt4	dmamux2_trg4				
			dmamux2_evt5	dmamux2_trg5				
D3	APB4	EXTI	it_exti_tx_lpuart1	dmamux2_trg6				
			it_exti_rx_lpuart1	dmamux2_trg7				
			it_exti_out_lptim2	dmamux2_trg8				
			it_exti_out_lptim3	dmamux2_trg9				
			it_exti_wkup_i2c4	dmamux2_trg10				
			it_exti_wkup_spi6	dmamux2_trg11				
			it_exti_out_comp1	dmamux2_trg12				
			it_exti_wkup_rtc	dmamux2_trg13				
			it_exti_exti0_syscfg	dmamux2_trg14				
			it_exti_exti2_syscfg	dmamux2_trg15				
D3	AHB4	DMAMUX2	dmamux1_req_out0	bdma_ch0	BDMA	AHB4	D3	リクエスト出力
			dmamux1_req_out1	bdma_ch1				
			dmamux1_req_out2	bdma_ch2				
			dmamux1_req_out3	bdma_ch3				
			dmamux1_req_out4	bdma_ch4				
			dmamux1_req_out5	bdma_ch5				
			dmamux1_req_out6	bdma_ch6				
			dmamux1_req_out7	bdma_ch7				

14 MDMA コントローラ（MDMA）

14.1 MDMA の概要

MDMA（Master Direct Memory Access、マスタダイレクトメモリアクセス）は、メモリ間またはペリフェラルとメモリ間でハイスピードデータ転送を行うために使用します。MDMA により、CPU を動作させずに、データを高速で移動することができます。これにより、CPU リソースを他の操作のためにあけておくことができます。

MDMA コントローラは、メインメモリとペリフェラルレジスタのアクセス（システムアクセスポート）用のマスタ AXI インタフェースと、Cortex-M7 TCM メモリアクセス（TCM アクセスポート）専用のマスタ AHB インタフェースを備えています。

MDMA は標準 DMA コントローラと連携して動作します（DMA1 および DMA2）。最大 16 本のチャネルがあり、それぞれが DMA ストリームメモリバッファの 1 つまたはその他のペリフェラル（内蔵 FIFO を含む）からのメモリアクセスリクエストを管理する役割を担っています。

14.2 MDMA の主な機能

- AXI/AHB マスタバスアーキテクチャ。1 つはメインメモリ／ペリフェラルアクセス専用、もう 1 つは Cortex-M7 AHBS ポート専用（TCM アクセス専用）
- 16 チャネル
- 最大 32 ハードウェアトリガソース
- 各チャネルリクエストは、任意のリクエストソースから選択することができます。この選択は、ソフトウェアによって設定可能であり、これによっていくつかのペリフェラルが DMA リクエストを発生することができるようになります。トリガの選択は 1 つのブロック転送の終了時に自動的に変更することができます。
- すべてのチャネルは同じであり、標準 DMA またはペリフェラルリクエスト（データの読み出し／書き込みで認識された）システムのどちらかに接続できます。
- 各チャネルで、ソフトウェアトリガもサポートします。
- 1 つまたは 2 つの連続するバッファについて、バーストまたはシングル転送モードで転送するデータを一時的に格納するために使用する、ファーストインファーストアウト（FIFO）で 2 つの 128 レベルに分けられた 1 つの 256 レベルメモリバッファ。FIFO には、現在のチャネルブロック転送中に転送されるデータも格納されます（ブロック転送サイズまで対応）。2 つ目の FIFO は、同じチャネルまたは次のチャネル転送のどちらかに対して、次に転送するバッファに使用できます。
- DMA チャネル間の優先順位は、ソフトウェアによってプログラム可能です（最高、高、中、低の 4 レベル）。レベルが等しい場合はハードウェアによって決定されます（チャネル 0 はチャネル 1 よりも優先、など）。
- 転送元および転送先の転送幅（バイト、ハーフワード、ワード、ダブルワード）に依存しない：転送元と転送先のデータ幅が異なる場合、MDMA ではバンド幅を最適化するために必要なデータのパック／アンパックを行うことができます。
- サイズとアドレスインクリメントは、転送元と転送先の両方について、個別に選択できます。

注： ***この選択に基づいて、ソフトウェアレベルでは、より高度なパック／アンパック操作を行うことができます。たとえば、2 x 16 ビットのデータブロックは、転送先メモリで 2 つの MDMA チャネルを使用することで、一緒にインターリーブすることができます。これには、単純に 4 バイトのインクリメントステップと 16 ビットのデータサイズを持つ 2 つのチャネル、および 2 つのチャネル間で 2 でシフトされた開始アドレスをプログラミングします。***

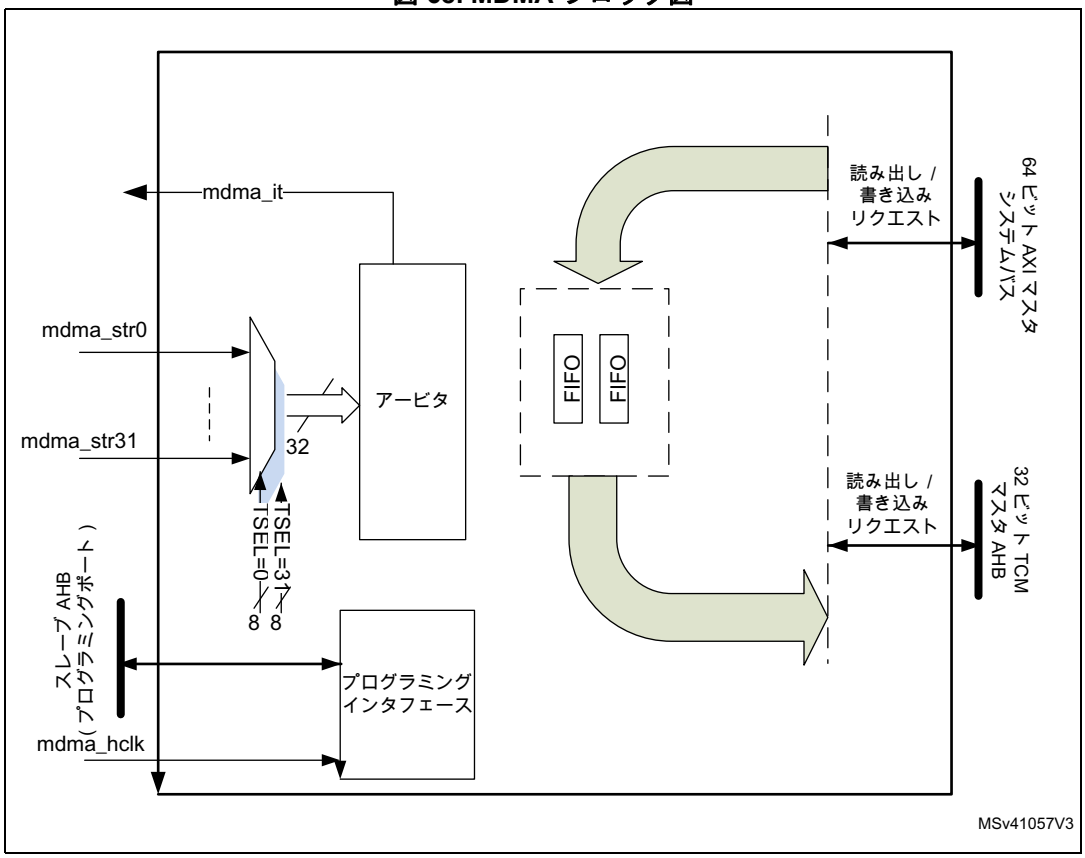
- 転送元および転送先に対するインクリメントアドレッシング、デクリメントアドレッシング、またはノンインクリメント／固定アドレッシング。
- データパッキング／アンパッキングは、常にリトルエンディアン規則に従って実行されます。データエンティティ（ダブルワード、ワード、またはハーフワード）の下位アドレスには、常に最下位バイトが含まれています。これは、転送元と転送先の両方のアドレスインクリメント／デクリメントモードとは無関係です。
- インクリメンタルバースト転送をサポートします。各バーストのサイズはソフトウェアで設定可能で、最大値は 128 MB です。これよりも大きなデータサイズの場合、最大 128 バイトのデータバーストサイズを考慮して、バースト長は制限されます（16x64 ビットまたは 32x32 ビットなど）。
- TCM メモリアクセスについては、インクリメントとデータサイズが同じで、32 ビット以下である場合にのみバーストアクセス可能です。
- 5 つのイベントフラグ（MDMA チャネル転送完了、MDMA ブロック転送完了、MDMA ブロック繰り返し転送完了、MDMA バッファ転送完了、MDMA 転送エラー）を使用可能で、割り込みを生成できます。

14.3 MDMA の機能詳細

14.3.1 MDMA ブロック図

図 68 に、MDMA のブロック図を示します。

図 68. MDMA ブロック図



14.3.2 MDMA 内部信号

*表 94*に、内部 MDMA 信号を示します。

表 94. MDMA 内部入力／出力信号

信号名	信号タイプ	説明
mdma_hclk	デジタル入力	MDMA AHB クロック
mdma_it	デジタル出力	MDMA 割り込み
mdma_str[0:31]	デジタル入力	MDMA ストリームリクエスト

14.3.3 MDMA の概要

MDMA コントローラはダイレクトメモリ転送を行います。AXI/AHB マスタとして、AXI/AHB トランザクションを開始するための AXI/AHB バスマトリックスの制御を行うことができます。

以下のトランザクションを実行することができます。

- メモリ間（ソフトウェアトリガ）
- ペリフェラルからメモリ
- メモリからペリフェラル

最後の 2 つのトランザクションタイプについて、メモリを MDMA フローが制御されないメモリマップドペリフェラルに置き換えることもできます。これらのタイプのトランザクションを使用する場合で、リクエストが標準 DMA （DMA1 または DMA2）から送信されている場合、ペリフェラルレジスタアクセスは、この DMA が使用するメモリバッファへのメモリアクセスに置き換えられます。

注： ***メモリアクセスにノンインクリメント／デクリメントモードは使用されません。***

転送元および転送先は、単純にアドレスによって定義されます（メモリマップドペリフェラルであるため）。

AHB スレーブポートは MDMA コントローラをプログラムするために用いられます（8/16/32 ビットアクセスのみサポート）。

シングルリクエストで転送されるデータ配列のサイズは、以下のいずれかです。

1. バッファ転送サイズ
2. ブロックサイズ
3. 繰り返しブロック
4. チャネルデータ全体（チャネルのリンクされたリストポインタが null になるまで）

サイズの選択は、TRGM[1:0] （トリガモード）選択フィールドを通じて行われます。

ユーザは、使用可能なデータ配列のサイズ（通常は DMA1/2 メモリバッファ）とその他の MDMA チャネルの「リアルタイム」要件に基づいて、いずれかを選択する必要があります（バッファ転送が、MDMA チャネルリクエスト間で新しいアービトレーションを実行せずに MDMA によって転送される、データの最小集合体であることを考慮すること）。

各チャネルには、3 つの主なデータ配列のサイズがあります。

1. バーストサイズ：バーストモードで実行可能なデータ転送長です。このバースト長では、バスアービトレーションレベルで割り込みできない最大転送長を定義し、その他のマスタがバスにアクセスしないようブロックすることができます。
2. バースト転送サイズ：その他のチャネルで MDMA リクエストを確認する前に、任意のチャネルで転送するデータ配列の長さです。MDMA レベルで他のチャネルリクエストによる割り込みができないデータ配列の転送長です。
3. ブロックサイズ：この値には 2 つの意味があり、一緒に使用できます。
   a) メイン：MDMA にリンクされたリストのブロック構造で説明されているデータブロック長（リンクされたリストの 1 つのエントリに対応）
   b) 選択可能：TRGM[1:0] が「01」のとき、シングル MDMA リクエストを有効化する際に転送されるデータ配列の長さ（それぞれのチャネルに対応）

14.3.4 MDMA チャネル

各 DMA コントローラチャネルは、転送元と転送先の間に単方向転送リンクを備えています。

各チャネルでは、以下を実行できます。

- 単一のブロック転送：1 つのブロックが転送されます。ブロックの最後で、DMA チャネルは無効化され、チャネル転送終了割り込みが生成されます。
- 繰り返しブロック転送：チャネルを無効化する前に、いくつかのブロックが転送されます。
- リンクされたリストの転送：現在のデータブロック（または繰り返しの最後のブロック）の転送が完了すると、メモリから新しいブロック制御構造がロードされ、新しいブロック転送が開始されます。

リクエスト（バッファサイズ、最大 128 バイト）ごとに転送される最小データ数はプログラム可能です。ブロックに含まれるデータの総量を、最大 64 KB にプログラム可能です。
この値は、各転送後にデクリメントされます。このカウンタが 0 に達すると、ブロックの終わりに到達し、繰り返しカウンタ（繰り返しブロック転送の場合）やリンクされたリストの構造の値に基づく操作が実行されます。

注： ***ブロック長がバッファ長の倍数でない場合、このブロックでの最後のバッファ転送は、現在のブロックで転送する残りのバイト数に応じて短くなります。***

リンク構造アドレスが有効なメモリアドレスを示している場合、MDMA ではこのアドレスのメモリからチャネルディスクリプタの構造レジスタの内容をすべて再ロードします。次に、この情報に基づいて、次の MDMA チャネルリクエストで新しいブロック転送が実行されます。

現在／繰り返しのブロック転送の終了時にリンク構造アドレスが 0x0 の場合、MDMA チャネルは無効化され、チャネル転送終了割り込みが生成されます。

14.3.5 転送元、転送先、および転送モード

転送元および転送先は両方とも、アドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4 GB メモリ領域内のペリフェラルおよびメモリを転送アドレスとして使用できます。

転送元／転送先アドレスは固定（FIFO／単一のデータレジスタのペリフェラルなど）するか、インクリメント／デクリメントできます。転送は、シングルアクセスまたはバーストモードで実行できます（プログラム可能）。

14.3.6 ポインタの更新

転送元および転送先のメモリのポインタは、MDMA_CxCR レジスタの SINC[1:0] および DINC[1:0] ビットに応じて、各転送後に任意で自動的にポストインクリメント／デクリメントまたは一定に保つことができます。

インクリメントモードの無効化は、転送元または転送先ペリフェラルデータを 1 つのレジスタを介して、または FIFO モードでアクセスする場合に便利です。

インクリメント／デクリメントモードが有効な場合、次のデータ転送アドレスは、MDMA_CxCR レジスタの SINCOS[1:0] または DINCOS[1:0] ビットにプログラムされたインクリメントサイズに応じて、前回のアドレスに 1、2、4、または 8 をインクリメント／デクリメントしたアドレスになります。

パッキング動作を最適化するため、インクリメントオフセットサイズとデータサイズは個別にプログラミング可能です。

14.3.7 MDMA バッファ転送

1 つのチャネルの MDMA リクエストイベントで転送される、論理的に最小のデータ量です（最大 128 バイト）。

MDMA バッファ転送は、一定数のデータ転送（シングルまたはバーストデータ転送）のシーケンスで構成されています。転送されるデータ項目の数とその幅（8 ビット、16 ビット、32 ビット、または 64 ビット）は、ソフトウェアでプログラム可能です。データ転送に使用するバースト長も個別にプログラム可能です。

データ配列を必要とするイベントの終了後、DMA／ペリフェラルは MDMA コントローラにリクエスト信号を送信します。MDMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。

これらのレジスタがセットされている場合に、アドレスが指定されたマスクアドレスにマスクデータ値を書き込むことで、リクエストが承認されます。

マスクアドレスレジスタがセットされていない場合（0x00 値）、リクエストは単純にデータをペリフェラルに読み出す／書き込むことで、リセットできます。この場合、リクエストが転送先のペリフェラルで実行されたときは、不正な新しい MDMA リクエストを避けるため、書き込みをバッファ不可にセットする必要があります。

現在のチャネルで、MDMA リクエストの後に転送するデータの総量は、TRGM[1:0] フィールドで決定されます。

TRGM[1:0] が 00 の場合、シングルバッファが転送され、その後 MDMA は同じチャネルで別のリクエストを待ちます。

注： ***この場合、現在アクティブなチャネルへのハードウェアのリクエスト（FIFO のデータ）は、このチャネルへの書き込みフェーズが終了するまで再び考慮されることはありません。この場合、読み出しフェーズ終了時にチャネルがまだアクティブであっても、別のチャネル（優先順位が低いものも含む）で読み出しフェーズを開始できます。これにより、優先順位の低いチャネルを現在のチャネル転送とインターリーブすることができます。***

TRGM[1:0] が 00 でない場合（複数のバッファを転送する必要がある）、現在のチャネルの mdma_strx は、TRGM （ブロック、繰り返しブロック、またはチャネル／リンクされたリスト全体）によって定義された転送がすべて完了するまで、アクティブなまま保持されます（内部的に記憶される）。ただし、個別のバッファの転送後、MDMA は新しいアービトレーションフェーズ（新しい外部リクエストと内部的に記憶されたリクエスト間）に移行します。他に優先順位の高いアクティブなチャネルリクエストがない場合は、新しいバッファ転送が同じチャネルで開始されます。

注： ***TRGM[1:0] が 00 でない場合、シングルリクエストでより大きなデータの配列が転送されます。ただし、チャネルアービトレーションが各バッファ転送後に実行されるため、どの優先順位が低いチャネルでも、上位の MDMA リクエストがバッファ転送周期を超えてブロックされることはありません。***

14.3.8 リクエストアービトレーション

アービタは、優先順位に基づいて MDMA チャネルリクエストを管理します。MDMA がアイドル状態の場合、各バッファの転送終了後に、すべての MDMA リクエスト（ハードウェアまたはソフトウェア）がすべての有効チャネルで確認されます。

優先順位は 2 段階で管理されます。

- ソフトウェア：各ストリームの優先順位は、MDMA_CxCR レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア：ハードウェアレベルでは、チャネルの優先順位は固定されています。2 つのリクエストのソフトウェア優先順位レベルが同じ場合､小さな番号のチャネルが大きな番号のチャネルよりも優先されます。たとえば、ソフトウェアの優先順位が同じ場合、チャネル 2 がチャネル 4 よりも優先されます。

14.3.9 FIFO

FIFO 構造は転送元からのデータを転送先へ書き込む前に一時的に格納するために使用します。すべてのチャネルに使用される集中型 FIFO 構造があります。

データの帯域幅とバスの使用率を最大化するために、複数の読み出し／書き込み動作を並列処理できる、以下のメカニズムが使用されます。

- バッファ転送中、FIFO に転送先へのバースト転送に十分なデータが蓄積されると、直ちに書き込み操作が開始されます。
- バッファ転送用の完全なデータを FIFO に読み出すと、アービトレーション手順が開始されます。続いて、次に転送するバッファデータを FIFO に読み出すことができます。

アクティブなチャネルがエラーによってバッファ転送中に無効化されると、内部 FIFO 内の残りのデータは破棄されます。

14.3.10 ブロック転送

ブロックは、最大 64 KB の「連続した」データの配列で、連続したバッファ転送によって転送されます。

データの各ブロックは、開始アドレスとブロック長で定義されます。ブロック転送が完了すると、以下の 3 つのアクションのいずれかが実行されます。

- ブロックが、繰り返しブロック転送の一部である場合 : ブロック長を再ロードし、新しいブロック開始アドレスを計算します（MDMA_CxBRUR レジスタの情報に基づく）。
- 単一のブロックである場合、または繰り返しブロック転送の最後のブロックである場合 : メモリから次のブロック情報をロードします（MDMA_CxLAR のリンクされたリストアドレス情報を使用）。
- 現在の MDMA チャネルに転送する必要がある最後のブロックである場合（MDMA_CxLAR = 0）: チャネルを無効にします。このチャネルに対するその後の MDMA リクエストは受け付けません。

14.3.11 ブロック繰り返しモード

ブロック繰り返しモードにより、転送先と転送元の開始アドレスが異なる場合でもブロック転送を繰り返すことができます。

繰り返しブロックモードがアクティブな場合（繰り返しカウンタがゼロでない場合）、現在のブロック転送の終了時にブロックパラメータが更新され（BRSUM/BRDUM の設定に従って BNDT 値が再ロードされ、SAR/DAR 値が更新されます）、繰り返しカウンタが 1 ずつデクリメントされます。

繰り返しブロックカウンタが 0 に達すると、この最後のブロックは、単一のブロック転送として処理されます。

14.3.12 リンクされたリストモード

リンクされたリストモードでは、CxLAR レジスタに指定されたアドレスから、新しい MDMA の設定（CxTCR、CxBNDTR、CxSAR、CxDAR、CxBRUR、CxLAR、CxTBR、CxMAR、および CxMDR レジスタ）をロードできます。このアドレスでは、AXI システムバス上に配置されたメモリを処理する必要があります。

この操作後、前述のブロック／繰り返しブロックモードで定義されたように、チャネルは新しいリクエストを受け付ける準備ができるか、TRGM[1:0] が 11 の場合は転送を続けます。

トリガソースは、CxTBR 値をロードする際に自動的に変更することができます。

TRGM および SWRM の値は、TRGM[1:0] が 11 の場合は変更しないでください。

14.3.13 MDMA 転送の完了

ステータスレジスタ（MDMA_CxISR）の CTCIFx ビットを設定することにより、さまざまなイベントによって転送を終了させることができます。

- MDMA_CxBNDTR カウンタがゼロに達すると、ブロック繰り返しカウンタもリンクされたリストポインタのアドレスも 0 になります。
- （MDMA_CxCR レジスタの EN ビットをクリアすることにより）チャネルは転送が終了する前に無効化され、残りのすべてのデータは FIFO から転送先に転送されます。

14.3.14 MDMA 転送の中断

MDMA 転送は、いつでも中断して再開することもできれば、MDMA 転送が終了する前に完全に無効化することもできます。

これには、2 つのケースがあります。

- チャネルが無効化され、停止した場所以降再開しない場合。チャネルを無効化するために MDMA_CxCR レジスタの EN ビットをクリアする以外、特に行うことはありません。チャネルを無効にするのに時間を要する可能性があります（まず、進行中のバッファ転送を完了します）。転送の終了を示すために転送完了割り込みフラグがセットされます。その時点で、MDMA_CxCR の EN ビットの値が 0 となって、チャネルが中断されます。ソフトウェアによって、チャネルが中断される前に転送されたデータ項目の数を判断できるように、MDMA_CxNDTR レジスタに、チャネルが停止した時点で残っているデータ項目の数を保持しています。
- チャネルは、MDMA_CxBNDTR レジスタ内の転送されるべき残りのバイト数が 0 に到達する前に転送を中断します。その目的は、チャネルを再度有効にすることで後から転送を再開することにあります。転送の終了を示すために CTCIF（Channel Transfer Complete Interrupt Flag、チャネル転送完了割り込みフラグ）がセットされます。MDMA_CxBNDTR、SAR、および DAR レジスタがソフトウェアによって変更されない場合、転送はチャネルを再度有効化したときに再開されます。チャネルを再開する前に、CTCIF をリセットする必要もあります。

注： ***完了したバッファが最後のブロックである場合、再起動に正しく備えるために、チャネルを無効にする前に設定レジスタも更新されます。***

注： ***チャネルを再プログラミングする前に、進行中のどの操作も完了していないことを保証するために、ソフトウェアでは CTCIF レジスタがセットされるのを待つ必要があります。***

14.3.15 エラー管理

MDMA コントローラによって、次のエラーを検出することができます。

転送エラー割り込みフラグ（TEIFx）は次の場合にセットされます。

- MDMA 読み出しまたは書き込みアクセス時にバスエラーが発生した場合
- アドレス割り当ては、データサイズと異なります。
- ブロックサイズはデータサイズ（転送元／転送先）の倍数ではありません。このエラーは最後の転送時に有効になり、エラーアドレスは最後の転送を示します（ただし、実現不可）。

14.4 MDMA 割り込み

各 MDMA チャネルに対し、次のイベントにて割り込みを生成することができます。

- チャネル転送完了
- ブロック転送完了
- ブロック繰り返し転送完了
- バッファ転送完了
- 転送エラー

柔軟性を持たせるため、*表 95* に示すよう個別の割り込みイネーブル制御ビットが使用できます。

表 95. MDMA 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
チャネル転送完了	CTCIF	CTCIE
ブロック転送繰り返し完了	BRTIF	BRTIE
ブロック転送完了	BTIF	BTIE
バッファ転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

注： *イネーブル制御ビットを '1' に設定する前に、対応するイベントフラグをクリアする必要があります。クリアしないと、ビットがすでにセットされている場合に、直ちに割り込みが生成される場合があります。*

注： *少なくとも 1 つの割り込みフラグと対応するイネーブル制御ビットがセットされると、GISR にチャネル割り込みビットがセットされます。また、割り込み出力も有効化されます。これにより、対応する割り込みチャネルが NVIC で有効化されると、割り込みが生成されます。*

14.5 MDMA レジスタ

MDMA レジスタは、ワード／ハーフワードまたはバイトフォーマットでアクセスできます。

14.5.1 MDMA グローバル割り込み／ステータスレジスタ（MDMA_GISR0）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GIF15	GIF14	GIF13	GIF12	GIF11	GIF10	GIF9	GIF8	GIF7	GIF6	GIF5	GIF4	GIF3	GIF2	GIF1	GIF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、すべての未使用チャネルで 0 として読み出されます。

ビット 15:0 **GIFx**：チャネル x グローバル割り込みフラグ（x = 0 ～ 15）

このビットは、ハードウェアによってセット／リセットされます。これは、割り込みマスクレジスタ（CTCIEx、BTIEx、BRTIEx、TEIEx）で有効化されているチャネル x 割り込みフラグ（CTCIFx、BTIFx、BRTIFx、TEIFx）の論理和です。

0：チャネル x で割り込みは生成されません。

1：チャネル x で割り込みが生成されます。

14.5.2 MDMA チャネル x 割り込み／ステータスレジスタ（MDMA_CxISR）（x = 0 ～ 15）

アドレスオフセット：0x40 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRQAx
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCIFx	BTIFx	BRTIFx	CTCIFx	TEIFx
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **CRQAx**：チャネル x リクエストアクティブ（Channel x ReQuest Active）フラグ

このビットは、MDMA 転送を要求するために、MDMA_CxCR レジスタの SWRQx ビットに 1 を書き込むことで、ソフトウェアによってセットされます。また、チャネル x が有効になります。

また、チャネルリクエストがアクティブになり、チャネルが有効化された場合に、ハードウェアによってセットされます。ハードウェアリクエストは、処理されるまで記憶されます。

これは、チャネル x リクエストが完了したときに、ハードウェアによってクリアされます（現在のリクエストの最後のバッファ転送のソース書き込みフェーズ後）。

0：MDMA 転送の mdma_strx はチャネル x でインアクティブです。

1：MDMA 転送の mdma_strx はチャネル x でアクティブです。

このビットは、チャネルが無効化された場合にハードウェアでリセットされます（転送エラーの場合、または繰り返しブロック = 0 でリンクされたリストのポインタが null になり、チャネルデータ転送の終わりに達した場合、またはそれ以前にソフトウェアでチャネルイネーブルビットを 0 にプログラミングした場合）。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TCIFx**：チャネル x バッファ転送完了割り込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にバッファ転送完了イベントは発生していません。

1：チャネル x にバッファ転送完了イベントが発生しました。

TC はシングルバッファが転送されたときにセットされます。チャネル転送リクエストごとに有効化されます。

デバッグ機能として使用できます（割り込みなし）。これは、最後のフラグリセット時に（少なくとも）MDMA バッファ転送が生成されたことを示します。

ビット 3 **BTIFx**：チャネル x ブロック転送完了割り込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にブロック転送完了イベントは発生していません。

1：チャネル x にブロック転送完了イベントが発生しました。

ビット 2 **BRTIFx**：チャネル x ブロック繰り返し転送完了割り込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にブロック繰り返し転送完了イベントは発生していません。

1：チャネル x にブロック繰り返し転送完了イベントが発生しました。

ビット 1 **CTCIFx**：チャネル x チャネル転送完了割り込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にチャネル転送完了イベントは発生していません。

1：チャネル x にチャネル転送完了イベントが発生しました。

CTC は最後のブロックが転送され、チャネルが自動的に無効化されたときセットされます。

また、CTC は EN ビットに 0 を書き込むことにより、チャネルが中断されたときにセットされます。

ビット 0 **TEIFx**：チャネル x 転送エラー割り込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

14.5.3 MDMA チャネル x 割り込みフラグクリアレジスタ（MDMA_CxIFCR）（x = 0 ～ 15）

アドレスオフセット：0x44 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLTCIFx	CBTIFx	CBRTIFx	CCTCIFx	CTEIFx
r	r	r	r	r	r	r	r	r	r	r	w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CLTCIFx**：チャネル x のクリアバッファ転送完了割り込みフラグ（CLear buffer Transfer Complete Interrupt Flag）

このビットに 1 を書き込むと、MDMA_ISRy レジスタの TCIFx がクリアされます。

ビット 3 **CBTIFx**：チャネル x クリアブロック転送完了割り込みフラグ

このビットに 1 を書き込むと、MDMA_ISRy レジスタの BTIFx がクリアされます。

ビット 2 **CBRTIFx**：チャネル x クリアブロック繰り返し転送完了割り込みフラグ

このビットに 1 を書き込むと、MDMA_ISRy レジスタの BRTIFx がクリアされます。

ビット 1 **CCTCIFx**：チャネル x のクリアチャネル転送完了割り込みフラグ

このビットに 1 を書き込むと、MDMA_ISRy レジスタの CTCIFx がクリアされます。

ビット 0 **CTEIFx**：チャネル x クリア転送エラー割り込みフラグ

このビットに 1 を書き込むと、MDMA_ISRy レジスタの TEIFx がクリアされます。

14.5.4 MDMA チャネル x エラーステータスレジスタ（MDMA_CxESR）（x = 0 ～ 15）

アドレスオフセット：0x48 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BSE	ASE	TEMD	TELD	TED	TEA[6:0]						
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BSE**：ブロックサイズエラー

これらのビットは、ブロックサイズが転送元または転送先どちらに対してもデータサイズの整数倍でない場合に、ハードウェアによってセットされます。問題が転送元にあるのか転送先にあるのかは、TED によって示されます。

MDMA_IFCRy レジスタの **CTEIFx** ビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ブロックサイズエラーはありません。

1：プログラムされたブロックサイズがデータサイズの整数倍ではありません。

ビット 10 **ASR**：アドレス／サイズエラー

これらのビットは、プログラムされたアドレスがデータサイズ境界に整列されていない場合に、ハードウェアによってセットされます。問題が転送元にあるのか転送先にあるのかは、TED によって示されます。

MDMA_IFCRy レジスタの **CTEIFx** ビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：アドレス／サイズエラーはありません。

1：プログラムされたアドレスがデータサイズと不整合です。

ビット 9 **TEMD**：転送エラーマスクデータ

これらのビットは、マスクデータの書き込み中に転送エラーが発生した場合に、ハードウェアによってセットされます。

MDMA_IFCRy レジスタの **CTEIFx** ビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：マスク書き込みアクセスエラーはありません。

1：チャネルの最後の転送エラーが、マスクデータの書き込みに関連している場合。

ビット 8 **TELD**：転送エラーリンクデータ

これらのビットは、ブロックリンクデータ構造の読み出し中に転送エラーが発生した場合に、ハードウェアによってセットされます。

MDMA_IFCRy レジスタの **CTEIFx** ビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：リンクデータの読み出しアクセスエラーはありません。

1：リンクデータ構造の読み出しに関連する、チャネルの最後の転送エラーが生成されます。

ビット 7 **TED**：転送誤差方向

このビットは、MDMA データ転送エラーが発生した場合に、ハードウェアによってセットおよびクリアされます。

0：読み出しアクセスに関連する、チャネルの最後の転送エラーが生成されます。

1：書き込みアクセスに関連する、チャネルの最後の転送エラーが生成されます。

ビット 6:0 **TEA[6:0]**：転送エラーアドレス

これらのビットは、MDMA データ転送エラーが発生した場合に、ハードウェアによってセットおよびクリアされます。これは TED と組み合わせて使用します。

このフィールドは、転送／アクセスエラーを生成したアドレスの 7 つの最下位ビットを示します。

失敗したアドレスを取得するために、ソフトウェアで使用できます。これには、この値（バッファ転送長サイズまで切り詰められた）を現在の SAR/DAR 値に追加します。

注： ***FIFO 管理システムにより、現在の SAR/DAR 値に最後のアドレスは反映されません。SAR/DAR は（バッファ）転送（TLEN+1 バイト）終了時に更新されます。***

注： ***リンクデータエラーの場合はセットされません。***

14.5.5 MDMA チャネル x 制御レジスタ（MDMA_CxCR）（x = 0 ～ 15）

このレジスタは当該チャネルを制御するために使用されます。

アドレスオフセット：0x4C + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWRQ
															w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WEX	HEX	BEX	Res.	Res.	Res.	Res.	PL[1:0]		TCIE	BTIE	BRTIE	CTCIE	TEIE	EN
	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SWRQ**：ソフトウェアリクエスト

このビットに 1 を書き込むと、MDMA_ISRy レジスタの CRQAx がセットされ、チャネル x でリクエストが有効になります。

注： ***CxCR レジスタ全体またはアドレスオフセットの 8 ビット／16 ビットレジスタのいずれか：0x4E + 0x40 × チャネル番号は、SWRQ 有効化に使用できます。***

ソフトウェアリクエストでは、確認応答は生成されません（ハードウェア信号も CxMAR 書き込みアクセスも生成されません）。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **WEX**：ワードエンディアンの入れ替え

このビットは、ソフトウェアによってセット／クリアされます。

0：ワード用に保持されたリトルエンディアン

1：ダブルワードでのワードの順序の入れ替え

このビットがセットされると、転送先のダブルワードのワードの順序は逆になります。上位のアドレスワードには転送元の下位のアドレスから読み出したデータが含まれます。

転送先がダブルワードでない場合は、このビットの値は無視してください。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 13 **HEX**：ハーフワードエンディアンの入れ替え

このビットは、ソフトウェアによってセット／クリアされます。

0：ハーフワード用に保持されたリトルエンディアン

1：各ワードでのハーフワードの順序の入れ替え

このビットがセットされると、転送先の各ワードのハーフワードの順序は逆になります。上位のアドレスのハーフワードには転送元の下位のアドレスから読み出したデータが含まれます。

転送先の長さがワードよりも短い場合は、このビットの値は無視してください。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 12 **BEX**：バイトエンディアンの入れ替え

このビットは、ソフトウェアによってセット／クリアされます。

0：バイト用に保持されたリトルエンディアン

1：各ハーフワードでのバイトの順序の入れ替え

このビットがセットされると、転送先の各ハーフワードのバイトの順序は逆になります。上位のアドレスのワードには転送元の下位のアドレスから読み出したデータが含まれます。

転送先がバイトでない場合は、このビットの値は無視してください。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **PL[1:0]**：優先順位レベル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：低

01：中

10：高

11：最優先

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 5 **TCIE**：バッファ転送完了割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割り込みは無効です。

1：TC 割り込みは有効です。

ビット 4 **BTIE**：ブロック転送割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：BT 完了割り込みは無効です。

1：BT 完了割り込みは有効です。

ビット 3 **BRTIE**：ブロック繰り返し転送割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：BT 割り込みは無効です。

1：BT 割り込みは有効です。

ビット 2 **CTCIE**：チャネル転送完了割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割り込みは無効です。

1：TC 割り込みは有効です。

ビット 1 **TEIE**：転送エラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割り込みは無効です。

1：TE 割り込みは有効です。

ビット 0 **EN**：チャネルイネーブル／ロー読み出し時のチャネルレディフラグ

このビットは、ソフトウェアによってセット／クリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

このビットは、次のときハードウェアによってクリアできます。

- MDMA 転送終了時（ストリームが設定準備完了）
- AHB/AXI マスタバス上で転送エラーが発生した場合（バスエラー／ハードフォールト）
- 他のエラー条件が発生した場合（データの配置、ブロック／データサイズ不適合）

このビットがソフトウェアによってリセットされると、進行中のバッファ転送（ある場合）は完了します。すべてのステータス／設定レジスタが現在値を保持します。これらのレジスタを書き込むことなくチャネルが再度有効化されると、チャネルは割り込まれたところから続行します。

このビットが 0 として読み出された場合、ソフトウェアによって設定レジスタをプログラムすることができます。EN ビットが 1 として読み出された場合、これらのレジスタの書き込みは禁止されています（書き込みは無視されます）。

注： ***このビットがソフトウェアによってリセットされると、チャネルを再プログラミングする前に進行中のすべてのバッファ転送が完了するのを確認するために、CTCIF = 1 になるのを待つことを推奨します。***

14.5.6 MDMA チャネル x 転送設定レジスタ（MDMA_CxTCR）（x = 0 ～ 15）

このレジスタは当該チャネルを設定するために使用されます。

アドレスオフセット：0x50 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWM	SWRM	TRGM[1:0]		PAM[1:0]		PKE	TLEN[6:0]							DBURST[2:1]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBURST[0]	SBURST[2:0]			DINCOS[1:0]		SINCOS[1:0]		DSIZE[1:0]		SSIZE[1:0]		DINC[1:0]		SINC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **BWM**：バッファ可能な書き込みモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送先書き込み操作はバッファ不可です。

1：転送先書き込み操作はバッファ可能です。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

注：　すべての MDMA 転送先アクセスはキャッシュ不可です。

ビット 30 **SWRM**：ソフトウェアリクエストモード

このビットは、ソフトウェアによってセット／クリアされます。現在アクティブなハードウェアまたはソフトウェアリクエストがある場合、ビットの変更は現在の転送が完了するまで遅延されます。

0：ハードウェアリクエストが考慮されます。転送は、TRGM 値で定義され、MDMA ACKx 信号で承認されたとおりに開始されます。

CxMAR に有効なアドレスが含まれる場合、CxMAR アドレスにも CxMDR 値が書き込まれます。

1：ハードウェアリクエストは無視されます。転送は、SWRQ ビットに 1 を書き込むことで、ソフトウェアによってトリガされます。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 29:28 **TRGM[1:0]**：トリガモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとにバッファ転送をトリガします。

01：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとにブロック転送をトリガします。

10：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとに繰り返しブロック転送をトリガします（ブロック繰り返しが 0 の場合、単一のブロックが転送されます）。

11：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとに、それぞれのチャネル（リンクされたリストなど）のデータ全体の転送を、チャネルの終わりに達して無効化されるまでトリガします。

注：　現在のブロックで TRGM が 11 の場合、リンクされたリストのメカニズムによって現在のブロックの最後にロードされたすべての値は同じ値（TRGM=11）および同じ SWRM 値を保持する必要があります。保持されない場合、結果は未定義です。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 27:26 **PAM[1:0]**：パディング／配置モード

これらのビットは、ソフトウェアによってセット／クリアされます。

ケース 1：転送元データサイズが転送先データサイズより小さい場合 - 3 つのオプションが有効です。
00：右詰め、0 でパディング（デフォルト）
01：右詰め、符号拡張
10：左詰め（0 でパディング）
11：予約済み

ケース 2：転送元データサイズが転送先データサイズより大きい場合。
00：右詰め - 転送元の最下位ビット部分のみが転送先アドレスに書き込まれます。
10：左詰め - 転送元の最上位ビット部分のみが転送先アドレスに書き込まれます。
残りの部分は破棄されます。

PKE = 1 または DSIZE = SSIZE のとき、これらのビットは無視されます。
これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 25 **PKE**：パック有効化

このビットは、ソフトウェアによってセット／クリアされます。
0：転送元データは単純に転送先に書き込まれます。
転送元のデータサイズが転送先より小さい場合は、PAM 値に従ってパディングされます。
転送元のデータサイズが転送先より大きい場合は、切り詰められます。配置は PAM[1:0] の値に従って行われます。
1：転送元データは、転送先のデータサイズにパック／アンパックされます。すべてのデータはリトルエンディアンモードで右詰めされます。
このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 24:18 **TLEN[6:0]**：転送長のバッファ（バイト数 - 1）

これらのビットは、ソフトウェアによってセット／クリアされます。
TLEN+1 の値はシングル転送で転送するバイト数を表します。
転送長はデータサイズの倍数である必要があります（転送元と転送先の両方）

注： ***転送元／転送先のサイズが異なる場合でパディング／切り詰めが使用されている場合、TLEN+1 は転送元の配列のサイズを参照します。***

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。
DBURST 値は、バーストサイズが転送サイズより低くなるようにプログラムする必要があります。

ビット 17:15 **DBURST[2:0]**：転送先バースト転送設定

これらのビットは、ソフトウェアによってセット／クリアされます。
000：シングル転送
N：2^N ビートのバースト
これらのビットは保護されており EN が 0 の場合のみ書き込みできます。
DBURST 値は、バーストサイズが転送長より低くなるようにプログラムする必要があります。これが確実でない場合は、予期しない結果になります。

注： ***転送先のバスが TCM/AHB（DBUS=1）と DINCOS=11 または DINC=00 または DINCOS/=DSIZE である場合、DBURST には 000 （シングル転送）をプログラムする必要があります。それ以外の場合は予期しない結果になります。***

注： ***転送先のバスがシステム／AXI バス（DBUS=0）と DINC=00 である場合、DBURST は最大 100（16 のバースト）である必要があります。それ以外の場合は予期しない結果になります。***

ビット 14:12 **SBURST[2:0]**：転送元バースト転送設定

これらのビットは、ソフトウェアによってセット／クリアされます。

000：シングル転送

N：2^N ビートのバースト

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

SBURST 値は、バーストサイズが転送長より低くなるようにプログラムする必要があります。これが確実でない場合は、予期しない結果になります。

注： ***転送元のバスが TCM （SBUS=1）と SINCOS=11 または SINC=00 または SINCOS/=SSIZE である場合、SBURST には 000 （シングル転送）をプログラムする必要があります。それ以外の場合は予期しない結果になります。***

注： ***転送元のバスがシステム／AXI バス（SBUS=0）と SINC=00 である場合、SBURST は最大 100（16 のバースト）である必要があります。それ以外の場合は予期しない結果になります。***

ビット 11:10 **DINCOS[1:0]**：転送先インクリメントオフセットサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：ダブルワード（64 ビット）-

ビット DINC[1:0] = '00' の場合、このビットは意味を持ちません。

これらのビットは保護されており EN = '0' の場合のみ書き込みできます。

DINCOS < DSIZE で DINC /= 00 の場合は、予期しない結果になります。

転送先が AHB で DBURST =/ 000 である場合、転送先アドレスは DINCOS サイズ境界に整列する必要があります。それ以外の場合は予期しない結果になります。

ビット 9:8 **SINCOS[1:0]**：転送元インクリメントオフセットサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：ダブルワード（64 ビット）-

ビット SINC[1:0] = '00' の場合、このビットは意味を持ちません。

これらのビットは保護されており EN = '0' の場合のみ書き込みできます。

SINCOS < SSIZE で SINC /= 00 の場合は、予期しない結果になります。

転送元が TCM/AHB で SBURST =/ 000 である場合、転送元アドレスは SINCOS サイズに合わせる必要があります。それ以外の場合は予期しない結果になります。

ビット 7:6 **DSIZE[1:0]**：転送先データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：ダブルワード（64 ビット）-

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

注： ***TCM アクセス／AHB ポートに値 11 をプログラムした場合、転送エラーが発生します（TEIF ビットがセットされる）。***

DINCOS < DSIZE で DINC /= 00 の場合は、予期しない結果になります。

注： ***DSIZE = 11 （ダブルワード）は、転送先が TCM/AHB バス（DBUS=1）の場合は禁止されます。***

ビット 5:4 **SSIZE[1:0]**：転送元データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：ダブルワード（64 ビット）

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

注： ***TCM アクセス／AHB ポートに値 11 をプログラムした場合、転送エラーが発生します(TEIF ビットがセットされる)。***

SINCOS < SSIZE で SINC /= 00 の場合は、予期しない結果になります。

注： ***SSIZE = 11 （ダブルワード）は、転送元が TCM/AHB バス（SBUS=1）の場合は禁止されます。***

ビット 3:2 **DINC[1:0]**：転送先インクリメントモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：転送先アドレスポインタは固定されています。

10：転送先アドレスポインタは、各データ転送後にインクリメントされます（DINCOS に応じてインクリメント）。

11：転送先アドレスポインタは、各データ転送後にデクリメントされます（DINCOS に応じてデクリメント）。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

注： ***転送先が AHB （DBUS=1）の場合、DINC = 00 は禁止されています。***

ビット 1:0 **SINC[1:0]**：転送元インクリメントモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：転送元アドレスポインタは固定されています。

10：転送元アドレスポインタは、各データ転送後にインクリメントされます（SINCOS に応じてインクリメント）。

11：転送元アドレスポインタは、各データ転送後にデクリメントされます（SINCOS に応じてデクリメント）。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

注： ***転送元が AHB （SBUS=1）の場合、SINC = 00 は禁止されています。***

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x00 で指定したアドレスから）ロードされます。

14.5.7 MDMA チャネル x データレジスタのブロック数（MDMA_CxBNDTR）（x = 0 ～ 15）

アドレスオフセット：0x54 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRC[11:0]												BRDUM	BRSUM	Res.	BNDT[16]
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BNDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 **BRC[11:0]**：ブロック繰り返しカウント

このフィールドには、現在のブロックの繰り返し数が含まれます（0 から 4095）。チャネルが有効になると、このレジスタは読み出し専用になり、現在のブロックを除いた残りのブロック数を示します。このレジスタは、各ブロック転送完了後にデクリメントされます。

最後のブロックの転送が完了すると、このレジスタは、ゼロのままにすることもできれば、メモリから自動的に再ロードすることもできます（リンクされたリストモード、つまりリンクアドレスは有効）。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 19 **BRDUM**：ブロック繰り返し転送先アドレス更新モード

0：ブロック転送の終了時に、DAR レジスタは現在の DAR 値に DUV を追加することで更新されます（現在の転送先アドレス）。

1：ブロック転送の終了時に、DAR レジスタは現在の DAR 値から DUV を差し引くことで更新されます（現在の転送先アドレス）。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 18 **BRSUM**：ブロック繰り返し転送元アドレス更新モード

0：ブロック転送の終了時に、SAR レジスタは現在の SAR 値に SUV を追加することで更新されます（現在の転送元アドレス）。

1：ブロック転送の終了時に、SAR レジスタは現在の SAR 値から SUV を削除することで更新されます（現在の転送元アドレス）。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **BNDT[16:0]**：転送するデータバイトのブロック数

現在のブロックで転送するバイト数（0 から最大 65536 まで）。チャネルが有効になると、このレジスタは読み出し専用になり、送信されるべき残りのデータ項目を示します。チャネル動作中、このレジスタはデクリメントし、現在のブロックの残りのデータ項目数を示します。

ブロック転送が完了すると、このレジスタは、ゼロのままにすることもできれば、チャネルがブロック繰り返しモードに設定されている場合は、事前にプログラムされた値を自動的に再ロードすることもできます。

このレジスタの値が 0 の場合、ストリームが有効であっても、トランザクションは処理できません。

これらのビットは保護されており EN が 0 の場合のみ書き込みできます。

注： 1：BNDT 値が TLEN+1 値の整数倍でない場合、最後の転送は短くなり、ブロックの残りのデータのみが含まれます。

注： 2：ブロックのサイズは、転送元と転送先のデータサイズの倍数である必要があります。この条件が満たされない場合、エラーがセットされ、データは書き込まれません。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x04 で指定したアドレスから）ロードされます。

14.5.8 MDMA チャネル x 転送元アドレスレジスタ（MDMA_CxSAR）（x = 0 ～ 15）

アドレスオフセット：0x58 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **SAR[31:0]**：ソースアドレス

これらのビットは、データの読み出し／書き込みが行われるペリフェラルデータレジスタのベースアドレスを表します。SSIZE に合わせる必要がありますが（SSIZE=10 の場合、SAR[1:0] = 00 など）、SINCOS と合わせる必要はありません。

転送元が TCM/AHB である場合、アドレスと SINCOS の整合が取れていない場合は、アクセスをシングル（SBURST=000）としてプログラムする必要があります。

これらのビットは書き込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書き込みできます。

チャネル動作中、このレジスタは更新され、次にデータが読み出される現在のアドレスが反映されます。

ブロック繰り返しモードがアクティブな場合、ブロック転送が完了すると、現在の値（ブロック内の最後の転送後に更新済み）への SAU 値の追加／削除によって転送元アドレスが更新されます。

リンクされたリストモードがアクティブな場合、ブロック（繰り返し）転送の最後に、SAR 値がメモリから（アドレス LSA + m から）ロードされます。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x08 で指定したアドレスから）ロードされます。

14.5.9 MDMA チャネル x 転送先アドレスレジスタ（MDMA_CxDAR）（x = 0 ～ 15）

アドレスオフセット：0x5C + 0x40 × *チャネル番号*

リセット値：0x0000 0000。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DAR[31:0]**：転送先アドレス

データの書き込みが行われる転送先アドレスのベースアドレス。

これらのビットは書き込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書き込みできます。

DSIZE に合わせる必要がありますが（DSIZE=01 の場合、DAR[0] = 0 など）、DINCOS と合わせる必要はありません。

転送先が AHB である場合、アドレスと DINCOS の整合が取れていない場合は、アクセスをシングル（DBURST=000）としてプログラムする必要があります。

チャネル動作中、このレジスタは更新され、次にデータが書き込まれる現在のアドレスが反映されます。

ブロック繰り返しモードがアクティブな場合、ブロック転送が完了すると、現在の値への DAU 値の追加／削除によって転送先アドレスが更新されます（ブロック内の最後の転送後）。

リンクされたリストモードがアクティブな場合、ブロック（繰り返し）転送の最後に、DAR 値がメモリから（アドレス LSA + m から）ロードされます。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x0C で指定したアドレスから）ロードされます。

14.5.10 MDMA チャネル x ブロック繰り返しアドレス更新レジスタ MDMA_CxBRUR （x = 0 ～ 15）

アドレスオフセット：0x60 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DUV[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUV[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **DUV[15:0]**：転送先アドレスの更新値

この値は、ブロック転送の終了時に現在の転送先アドレスを更新するために使用します（追加または削除）。DAR と DSIZE の整合性を保持するために（例：DSIZE=10 の場合、DAR[1:0] = 00）、DSIZE の整数倍である必要があります。

この値が 0 の場合、次のブロック転送の繰り返しは次のアドレスに続きます。

このブロック繰り返しモードがアクティブでない場合（BRC=0）、このフィールドは無視されます。

これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

注： DINC[1:0] = 00 のとき、このフィールドは 0 にセットする必要があります。

ビット 15:0 **SUV[15:0]**：転送元アドレスの更新値

この値は、ブロック転送の終了時に現在の転送元アドレスを更新するために使用します（追加または削除）。SAR と SSIZE の整合性を保持するために（例：SSIZE=10 の場合、SAR[1:0] = 00）、SSIZE の整数倍である必要があります。

この値が 0 の場合、次のブロック転送の繰り返しは次のアドレスから続行します。

このブロック繰り返しモードがアクティブでない場合（BRC=0）、このフィールドは無視されます。

これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

注： SINC[1:0] = 00 のとき、このフィールドは 0 にセットする必要があります。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x10 で指定したアドレスから）ロードされます。

14.5.11 MDMA チャネル x リンクアドレスレジスタ（MDMA_CxLAR）（x = 0 ～ 15）

アドレスオフセット：0x64 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r	r

ビット 31:0 **LAR[31:0]**：リンクアドレスレジスタ

（繰り返し）ブロック転送の最後に、現在のチャネル設定レジスタ（CxTCR、CxBNDTR、CxSAR、CxDAR、CxBRUR、CxMAR、CxMDR、および CxLAR レジスタ自体）はこのアドレスのデータ構成によってロードされます。

このレジスタの値が 0 の場合、レジスタの更新は行われず、チャネルは無効化され、

また CTCIF がセットされ、このチャネルの転送の終了を示します。これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

チャネル設定（LAR アドレス）は AXI アドレス空間になければなりません。

LAR 値は、LAR[2:0] = 0x0 などのダブルワードアドレス境界に整列する必要があります。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x14 で指定したアドレスから）ロードされます。

注：　新しい値は、次のブロック終了時のため、すべてのレジスタが更新された後にのみ考慮されます。

14.5.12 MDMA チャネル x トリガおよびバス選択レジスタ（MDMA_CxTBR）（x = 0 ～ 15）

アドレスオフセット：0x68 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSEL[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DBUS**：転送先バス選択

0：システム／AXI バスを、チャネル x で転送先（書き込み操作）として使用します。

1：AHB バス／TCM を、チャネル x で転送先（書き込み操作）として使用します。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 16 **SBUS**：転送元バス選択

0：システム／AXI バスを、チャネル x で転送元（読み出し操作）として使用します。

1：AHB バス／TCM を、チャネル x で転送元（読み出し操作）として使用します。

このビットは保護されており、EN が 0 のときのみ書き込みできます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TSEL[5:0]**：トリガ選択

このビットフィールドは、チャネル x でハードウェアトリガ（RQ）入力を選択します。ACK は、同じインデックス値を持つ ACK 出力で送信されます。

SWRM ビットをセットすると（ソフトウェアリクエストを選択）、このビットフィールドは無視されます。

これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

注： ***同じイベント（同じ TSEL 値を持つ）によって複数のチャネルがトリガされた場合、それらはすべて並列でトリガされます。ただし、最も低いインデックスを持つチャネルのみがリクエストを承認します。***

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x18 で指定したアドレスから）ロードされます。

14.5.13 MDMA チャネル x マスクアドレスレジスタ（MDMA_CxMAR）（x = 0 ～ 15）

アドレスオフセット：0x70 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MAR[31:0]**：マスクアドレス

MDR 値の書き込みも、このアドレスに対して行われます。これにより、その割り込みクリアレジスタに書き込むことで、DMA2 によって生成される RQ 信号をクリアできます。
このレジスタの値が 0 の場合、この機能は無効です。これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x20 で指定したアドレスから）ロードされます。

14.5.14 MDMA チャネル x マスクデータレジスタ（MDMA_CxMDR）（x = 0 ～ 15）

アドレスオフセット：0x74 + 0x40 × *チャネル番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MDR[31:0]**：マスクデータ

MDR 値の書き込みも、MAR レジスタによって定義されるこのアドレスに対して行われます。これにより、その割り込みクリアレジスタに書き込むことで、DMA2 によって生成される RQ 信号をクリアできます。
これらのビットは書き込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書き込みできます。

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x24 で指定したアドレスから）ロードされます。

14.5.15 MDMA レジスタマップ

表 96 に MDMA レジスタの要約を示します。

表 96. MDMA レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																																
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	MDMA_GISR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GIF15	GIF14	GIF13	GIF12	GIF11	GIF10	GIF9	GIF8	GIF7	GIF6	GIF5	GIF4	GIF3	GIF2	GIF1	GIF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04 - 0x3C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x40 + 0x40 × チャネル番号	MDMA_CxISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRQAx	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCIFx	BTIFx	BRTIFx	CTCIFx	TEIFx
	リセット値																0												0	0	0	0	0
0x44 + 0x40 × チャネル番号	MDMA_CxIFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTCIFx	CBTIFx	CBRTIFx	CCTCIFx	CTEIFx
	リセット値																												0	0	0	0	0
0x48 + 0x40 × チャネル番号	MDMA_CxESR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMD	TELD	TED	TEA[6:0]						
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x4C + 0x40 × チャネル番号	MDMA_CxCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWRQ	Res.	WEX	HEX	BEX	Res.	Res.	Res.	Res.	PL[1:0]		TCIE	BTIE	BRTIE	CTCIE	TEIE	EN
	リセット値																0		0	0	0					0	0	0	0	0	0	0	0
0x50 + 0x40 × チャネル番号	MDMA_CxTCR	BWM	SWRM	TRGM[1:0]		PAM[1:0]		PKE	TLEN[6:0]							DBURST[2:0]			SBURST[2:0]			DINCOS[1:0]		SINCOS[1:0]		DSIZE[1:0]		SSIZE[1:0]		DINC[1:0]		SINC[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54 + 0x40 × チャネル番号	MDMA_CxBNDTR	BRC[11:0]												BRDUM	BRSUM	Res.	BNDT[16:0]																
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x58 + 0x40 × チャネル番号	MDMA_CxSAR	SAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C + 0x40 × チャネル番号	MDMA_CxDAR	DAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60 + 0x40 × チャネル番号	MDMA_CxBRUR	DUV[15:0]																SUV[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64 + 0x40 × チャネル番号	MDMA_CxLAR	LAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68 + 0x40 × チャネル番号	MDMA_CxTBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBUS	SBUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSEL[5:0]					
	リセット値															0	0											0	0	0	0	0	0
0x6C + 0x40 × チャネル番号	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

表 96. MDMA レジスタマップとリセット値（続き）

オフセット	レジスタ名 リセット値	レジスタサイズ																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x70 + 0x40 × チャネル番号	MDMA_CxMAR	MAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 + 0x40 × チャネル番号	MDMA_CxMDR	MDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C + 0x40 × チャネル番号	予約済み	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
	リセット値																																

レジスタ境界アドレスについては、*セクション 2.2.2 : メモリマップとレジスタ境界アドレス* を参照してください。

参考資料

15 ダイレクトメモリアクセスコントローラ（DMA1、DMA2）

15.1 DMA の概要

DMA（Direct Memory Access）は、ペリフェラルとメモリ間、およびメモリ間でハイスピードデータ転送を行うために使用します。データは、CPU の動作を必要とすることなく、DMA によって高速に移動できます。これにより、CPU リソースを他の操作のためにあけておくことができます。

このDMA コントローラは、高機能なバスマトリックスアーキテクチャに基づいてシステムの帯域幅を最大限に利用するため、強力なデュアル AHB マスタバスアーキテクチャと、独立した FIFO を兼ね備えています。

2 つの DMA コントローラは合計 16 ストリーム（各コントローラごとに 8 つ）を搭載し、それぞれが 1 つ以上のペリフェラルからのメモリアセスリクエストを管理します。各ストリームは、合計で最大 8 チャネル（リクエスト）を処理することができます。また、それぞれ DMA リクエスト間の優先順位を操作するためのアービタを内蔵しています。

15.2 DMA の主な機能

DMA の主な機能は次のとおりです。

- デュアル AHB マスタバスアーキテクチャ（メモリアクセス専用、およびペリフェラルへのアクセス専用）
- 32 ビットアクセスのみをサポートする AHB スレーブプログラミングインタフェース
- DMA コントローラごとに 8 ストリーム、ストリームごとに最大 115 チャネル（リクエスト）
- ストリームごとに独立した 4 つの 32 ビットファーストインファーストアウトメモリバッファ（FIFO）。（FIFO モードまたはダイレクトモードで使用可能）
  - FIFO モード : FIFO サイズの 1/4、1/2 または 3/4 からソフトウェアで選択可能な閾値レベルを持つ
  - ダイレクトモード

    各 DMA リクエストによって、直ちにメモリとの転送を開始します。DMA がダイレクトダイレクトモード（FIFO が無効）に設定されている場合、“メモリからペリフェラル”モードでデータを転送するため、ペリフェラルからの DMA リクエストが発生した際、即座にかつ確実にデータ転送が行われるよう、DMA はデータをメモリから内部 FIFO へ 1 つだけプリロードします。
- 各ストリームはハードウェアによって次のように設定することができます。
  - “ペリフェラルからメモリ”、“メモリからペリフェラル”および“メモリ間”転送をサポートするレギュラチャネル
  - メモリ側でダブルバッファリングもサポートするダブルバッファチャネル
- 8 つのストリームのそれぞれは、専用のハードウェア DMA チャネル（リクエスト）に接続されます。
- DMA ストリームリクエスト間の優先順位は、ソフトウェアによってプログラミングできます（最高、高、中、低の 4 レベル）。レベルが等しい場合はハードウェアによって決まります（リクエスト 0 はリクエスト 1 よりも優先、など）です。

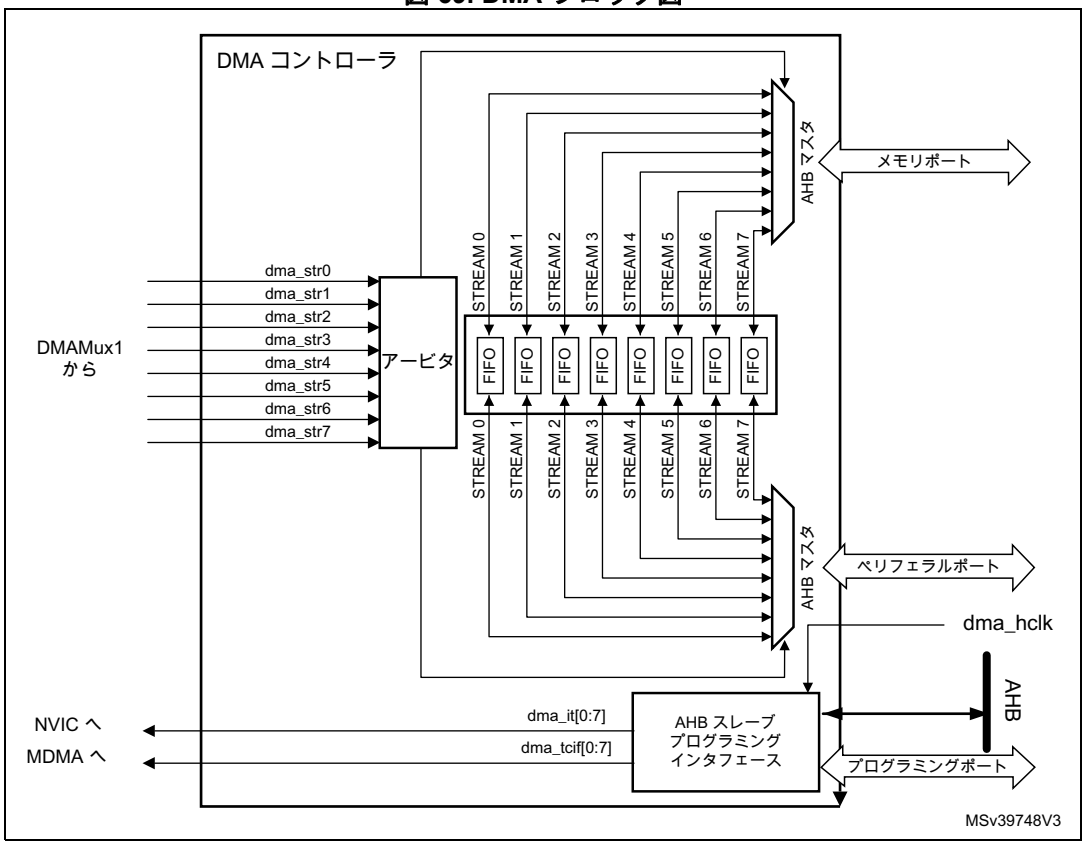
- 各ストリームは、メモリ間転送のためのソフトウェアトリガもサポートしています。
- 各ストリームリクエストは、発生しうる最大 115 のチャネルリクエストから DMAMux1 を介して選択することができます。この選択は、ソフトウェアによって設定可能であり、これによって多くのペリフェラルが DMA リクエストを発生することができるようになります。
- 転送するデータの数は、DMA コントローラまたはペリフェラルのいずれかによって管理することができます。
  - DMA フローコントローラ : 転送するデータ数は、1 から 65535 の間でソフトウェアによってプログラム可能です。
  - ペリフェラルフローコントローラ : 転送されるデータ項目の数は不明であり、ハードウェアによって転送の終了を通知する転送元または転送先ペリフェラルによって制御されます。
- 転送元および転送先の転送幅（バイト、ハーフワード、ワード）に依存しない : 転送元と転送先のデータ幅が異なる場合、DMA によって自動的に必要な転送のパック／アンパックが行われ、バンド幅が最適化されます。この機能は、FIFO モードでのみ使用できます。
- 転送元および転送先に対するインクリメントアドレッシングまたはノンインクリメントアドレッシング
- 4、8、または 16 ビートのインクリメンタルバースト転送をサポート。バーストのサイズはソフトウェアで設定可能であり、通常はペリフェラルの FIFO サイズの半分です。
- サーキュラバッファ管理は、各ストリームがサポートします。
- 各ストリームに対する同一の割り込みリクエストにおいて、5 つのイベントフラグ（DMA 1/2 転送、DMA 転送完了、DMA 転送エラー、DMA FIFO エラー、ダイレクトモードエラー）の論理和がとられます。

15.3 DMA の機能説明

15.3.1 DMA ブロック図

図 69 に、DMA のブロック図を示します。

図 69. DMA ブロック図



15.3.2 DMA 内部信号

*表 97*に、内部 DMA 信号を示します。

表 97. DMA 内部入力／出力信号

信号名	信号タイプ	説明
dma_hclk	デジタル入力	DMA AHB クロック
dma_it[0:7]	デジタル出力	DMA ストリーム [0:7] グローバル割り込み
dma_tcif[0:7]	デジタル出力	MDMA トリガ
dma_str[0:7]	デジタル入力	DMA ストリーム [0:7] リクエスト

15.3.3 DMA の概要

DMA コントローラはダイレクトメモリ転送を行います。AHB マスタとして、AHB トランザクションを開始するための AHB バスマトリックスの制御を行うことができます。

以下のトランザクションを実行することができます。

- ペリフェラルからメモリ
- メモリからペリフェラル
- メモリ間

この DMA コントローラは 2 つの AHB マスタポートを搭載しています。メモリに接続するための *AHB メモリポート*、およびペリフェラルに接続するための *AHB ペリフェラルポート*です。しかし、メモリ間転送を可能とするため、*AHB ペリフェラルポート*はメモリにもアクセスできる必要があります。

AHB スレーブポートは DMA コントローラをプログラムするために用いられます（32 ビットアクセスのみサポート）。

15.3.4 DMA トランザクション

DMA トランザクションは、一定数のデータ転送の列で構成されています。転送されるデータ項目の数とその幅（8 ビット、16 ビット、または 32 ビット）は、ソフトウェアでプログラム可能です。

各 DMA 転送は次の 3 つの操作で構成されます。

- DMA_SxPAR または DMA_SxM0AR レジスタを通じてアドレス指定されたメモリ上の場所またはペリフェラルデータレジスタからのデータのロード。
- DMA_SxPAR または DMA_SxM0AR レジスタを通じてアドレス指定されたメモリ内の場所またはペリフェラルデータレジスタにロードされたデータの格納。
- このあと予定される DMA トランザクションの数を保持している DMA_SxNDTR レジスタをカウントダウンします。

イベントの終了後、ペリフェラルは DMA コントローラにリクエスト信号を送信します。DMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。DMA コントローラがペリフェラルにアクセスするとすぐに、DMA コントローラはペリフェラルに確認応答信号を送信します。ペリフェラルは、DMA コントローラからの確認応答信号を受け取るとすくにそのリクエストを解除します。ペリフェラルによってリクエストがネゲートされると、DMA コントローラは確認応答信号を解除します。さらにリクエストがある場合、ペリフェラルは次のトランザクションを開始できます。

15.3.5 DMA リクエストマッピング

ペリフェラルおよび DMA チャネルへの DMA リクエストマッピングについては、*セクション 17.3.2：DMAMUX1 の配置*で説明しています。

15.3.6 アービタ

アービタは、2 つの AHB マスタポート（メモリポートおよびペリフェラルポート）のそれぞれについて、その優先順位に基づいて 8 つの DMA ストリームリクエストを管理し、ペリフェラル/メモリアクセスシーケンスを起動します。

優先順位は 2 段階で管理されます。

- ソフトウェア：各ストリームの優先順位は、DMA_SxCR レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア：2 つのリクエストのソフトウェア優先順位レベルが同じ場合、小さな番号のストリームが大きな番号のストリームよりも優先されます。たとえば、ストリーム 2 はストリーム4よりも優先されます。

15.3.7 DMA ストリーム

8 つの DMA コントローラストリームのそれぞれに、転送元と転送先の間の単方向転送リンクを備えています。

各ストリームは次のことを行うよう設定することができます。

- レギュラータイプトランザクション：メモリからペリフェラル、ペリフェラルからメモリまたはメモリ間の転送
- ダブルバッファタイプトランザクション：メモリ用に 2 つのメモリポインタを使用したダブルバッファ転送（DMA がバッファの読み書き込みを行っている間、アプリケーションは他のバッファの読み/書きが可能）

転送するデータの量（65535 以内）はプログラム可能で、ペリフェラル AHB ポートに接続されたDMA転送を要求するペリフェラルのデータ幅に関連しています。転送されるデータ数を格納しているレジスタは、トランザクションの終了ごとにデクリメントされます。

15.3.8 転送元、転送先、および転送モード

転送元および転送先は両方とも、アドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4 GB メモリ領域内のペリフェラルおよびメモリを転送アドレスとして使用できます。

方向は DMA_SxCR レジスタの DIR[1:0] ビットを使用して設定し、メモリからペリフェラル、ペリフェラルからメモリ、およびメモリ間の転送の 3 つの選択肢があります。*表 98* に対応する転送元と転送先アドレスを示します。

表 98. 転送元および転送先アドレス

DMA_SxCR レジスタの DIR [1:0] ビット	方向	ソースアドレス	転送先アドレス
00	ペリフェラルからメモリ	DMA_SxPAR	DMA_SxM0AR
01	メモリからペリフェラル	DMA_SxM0AR	DMA_SxPAR
10	メモリ間転送	DMA_SxPAR	DMA_SxM0AR
11	予約済み	-	-

DMA_SxCR レジスタの PSIZE または MSIZE ビットでプログラムされたデータ幅が、それぞれハーフワードまたはワードである場合、DMA_SxPAR または DMA_SxM0AR/M1AR レジスタに書き込まれたペリフェラルアドレスまたはメモリアドレスは、それぞれワードアドレス境界またはハーフワードアドレス境界で整列する必要があります。

“ペリフェラルからメモリ”モード

図 70 でこのモードを説明しています。

このモードが有効になっている場合（DMA_SxCR レジスタの EN ビットをセットすることにより）、ペリフェラルリクエストが発生するたびに、ストリームは転送元からの転送を開始し、FIFO を満たします。

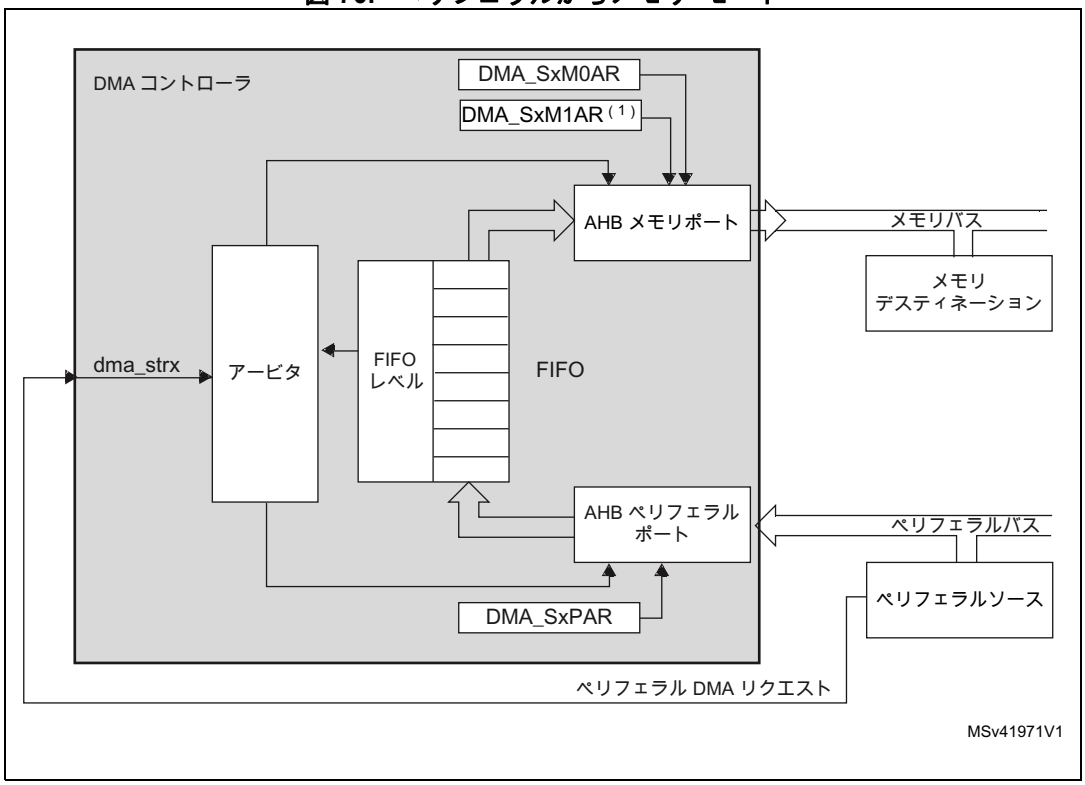
FIFO の閾値レベルに達すると、FIFO の内容が排出され、転送先に格納されます。

DMA_SxNDTR レジスタがゼロに達したか、ペリフェラルが転送の終了を要求したか（ペリフェラルフローコントローラの場合）、またはソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき、転送は停止します。

ダイレクトモード（DMA_SxFCR レジスタの DMDIS 値が“0”のとき）では、FIFO の閾値レベルは使用されません。ペリフェラルから FIFO への各データの転送後、対応するデータはすぐに排出されて転送先に格納されます。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

図 70. “ペリフェラルからメモリ”モード



1. ダブルバッファモード用。

“メモリからペリフェラル”モード

図 71 でこのモードを説明しています。

このモードが有効になっている場合（DMA_SxCR レジスタの EN ビットをセットしている場合）、ストリームはすぐに転送元から転送を開始し、FIFO を完全に満たします。

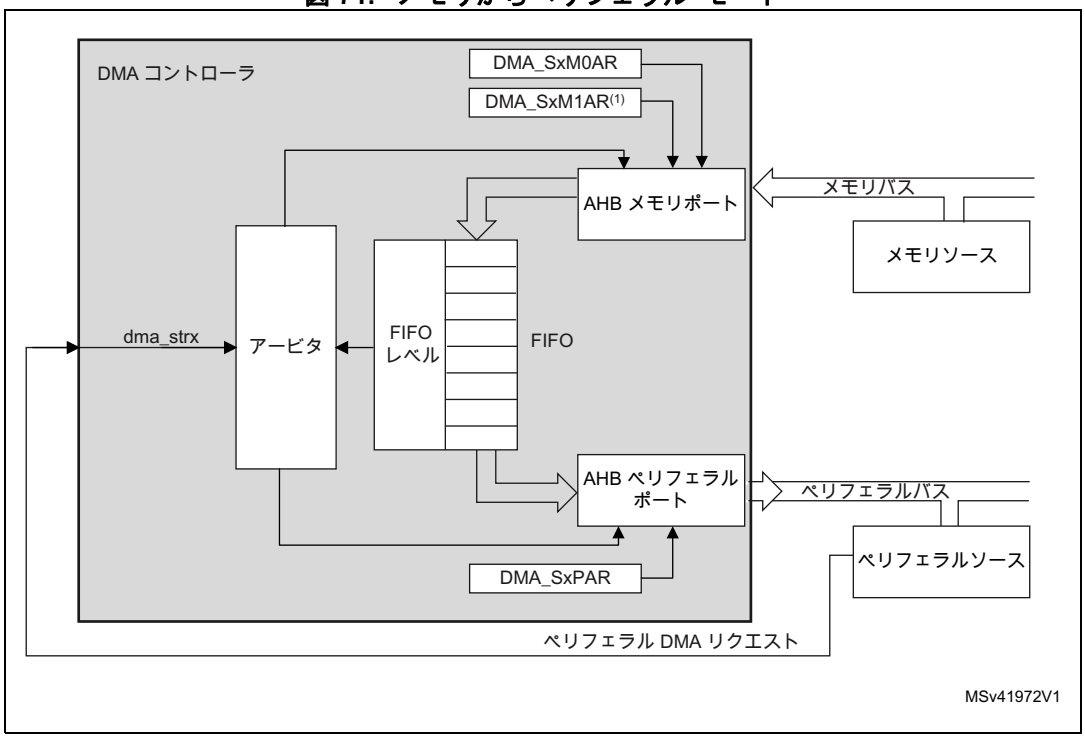
ペリフェラルリクエストが発生するたびに、FIFO のデータが転送されて転送先に格納されます。FIFO のレベルがあらかじめ設定された閾値レベルより低いか等しいとき、FIFO はメモリからのデータで完全に再ロードされます。

DMA_SxNDTR レジスタがゼロに達したか、ペリフェラルが転送の終了を要求したか（ペリフェラルフローコントローラの場合）、またはソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき、転送は停止します。

ダイレクトモード（DMA_SxFCR レジスタの DMDIS 値が“0”のとき）では、FIFO の閾値レベルは使用されません。ストリームが有効になると、DMA は内部 FIFO に転送するための最初のデータをプリロードします。ペリフェラルがデータ転送を要求すると、DMA はプリロードされたデータを設定された転送先へすぐに転送します。そしてまた転送すべき次のデータを空の内部 FIFO に再ロードします。プリロードデータサイズは、DMA_SxCR レジスタの PSIZE ビットフィールドの値に対応します。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

図 71. “メモリからペリフェラル”モード



1. ダブルバッファモード用。

メモリ間モード

DMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても機能します。これは *図 72* で説明しているメモリ間モードです。

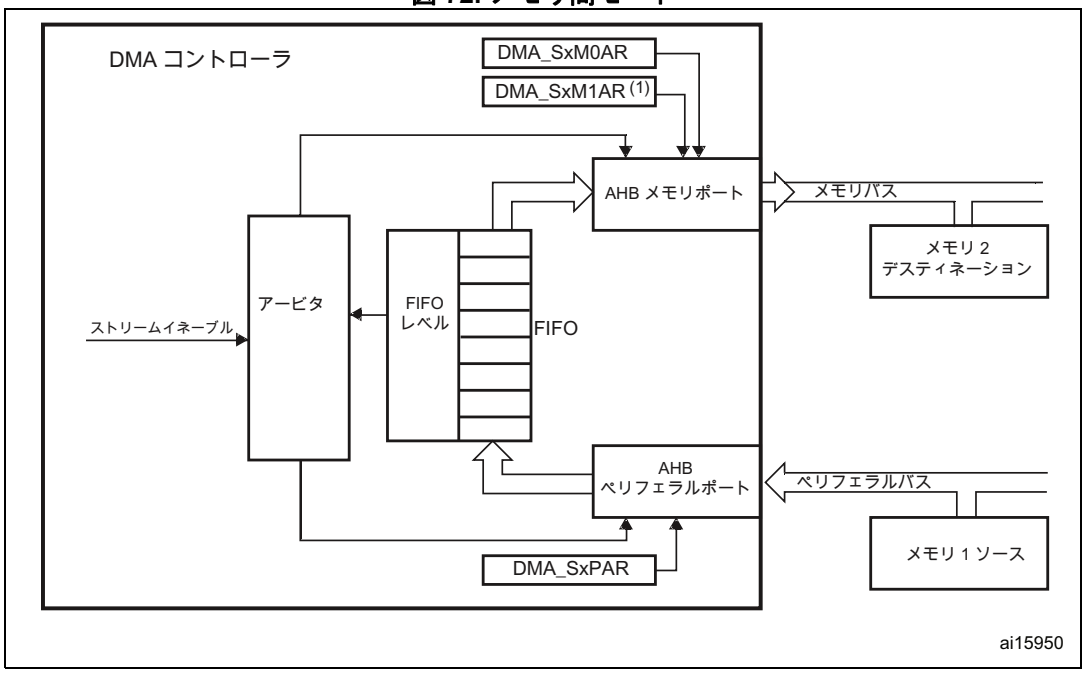
DMA_SxCR レジスタのイネーブルビット（EN）をセットすることによってストリームが有効になっている場合、ストリームはすぐに閾値レベルまで FIFO への書き込みを開始します。FIFO の閾値レベルに達すると、FIFO のデータが転送され、転送先に格納されます。

DMA_SxNDTR レジスタがゼロに達するかソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき転送を停止します。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

注： ***メモリ間モードを使用する場合、サーキュラモードおよびダイレクトモードは使用できません。***

図 72. メモリ間モード



1. ダブルバッファモード用。

15.3.9 ポインタのインクリメント

ペリフェラルやメモリのポインタは、DMA_SxCR レジスタの PINC および MINC ビットに応じて、各転送後に自動的にポストインクリメントまたは一定に保つかすることができます。

インクリメントモードの無効化は、転送元または転送先ペリフェラルデータを 1 つのレジスタを介してアクセスする場合に便利です。

インクリメントモードが有効な場合、次の転送アドレスは、DMA_SxCR レジスタの PSIZE または MSIZE ビットにプログラムされたデータ幅に応じて、前回のアドレスに 1（バイトの場合）、2（ハーフワードの場合）、または 4（ワードの場合）をインクリメントしたアドレスになります。

パッキング動作を最適化するため、AHB ペリフェラルポート上で転送されるデータのサイズに関わらずペリフェラルアドレスのインクリメントオフセットサイズを固定することができます。DMA_SxCR レジスタの PINCOS ビットは、インクリメントオフセットサイズをペリフェラル AHB ポート上の

データサイズまたは 32 ビットアドレス（この場合アドレスは 4 つインクリメントされます）に調整するために使用されます。PINCOS ビットは AHB ペリフェラルポートにのみ影響します。

PINCOS ビットがセットされている場合、次の転送のアドレスは、PSIZE 値にかかわらず前のアドレスに 4 をインクリメント（自動的に 32 ビットアドレスに調整）したアドレスです。しかしながら、この動作による AHB メモリポートへの影響はありません。

15.3.10 サーキュラモード

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（ADC スキャンモードなど）を処理できます。この機能は、DMA_SxCR レジスタの CIRC ビットを使用して有効にできます。

サーキュラモードが有効になると、転送されるデータ項目数にはストリーム設定段階でプログラミングされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。

注： ***サーキュラモードでは、メモリ用にバーストモードが設定されている場合、次のルールを守る必要があります。***

DMA_SxNDTR = ((Mburst ビート) の倍数× (Msize)/(Psize))、ここで

- *(Mburst ビート) = 4、8 または 16（DMA_SxCR レジスタの MBURST ビットによって異なる）*
- *((Msize)/(Psize)) = 1、2、4、1/2 または 1/4（Msize と Psize は DMA_SxCR レジスタの MSIZE と PSIZE ビットを表す。これらはバイト依存）*
- *DMA_SxNDTR = AHB ペリフェラルポート上で転送するデータ項目の数*

例：Mburst ビート=8（INCR8）、MSIZE=00（バイト）および PSIZE=01（ハーフワード）、この場合、DMA_SxNDTR は（8 × 1/2 = 4）の倍数でなければなりません。

この式を守らないと DMA 動作とデータの整合性は保証されません。

NDTR もペリフェラルバーストサイズの倍数にペリフェラルデータサイズを乗じた値でなければならず、さもないと不正な DMA 動作を引き起こす可能性があります。

15.3.11 ダブルバッファモード

このモードは、すべての DMA1 と DMA2 ストリームに対して使用可能です。

ダブルバッファモードは、DMA_SxCR レジスタの DBM ビットをセットすることで有効になります。

ダブルバッファストリームは通常の（シングルバッファ）ストリームとして動作しますが、2 つのメモリポインタを持っているところが異なります。ダブルバッファモードが有効になっていると、サーキュラモードが自動的に有効になり（DMA_SxCR の CIRC ビットは無視）、トランザクションが終わるごとにメモリポインタをスワップします。

このモードでは、DMA コントローラはトランザクションが終わるごとに、あるメモリターゲットから他へスワップします。これにより、DMA 転送によってある 1 つのメモリ領域がフルまたは使用中である間に、ソフトウェアは別のメモリ領域を処理することができます。ダブルバッファストリームは、*表 99：ダブルバッファモード (DBM=1) における転送元および転送先アドレスレジスタ*で説明しているように両方向に動作することができます（メモリは転送元または転送先のいずれにも設定可能）。

注： ***ダブルバッファモードでは、ストリームが有効なとき、以下の条件を守ることにより、いつでも AHB メモリポートのベースアドレス (DMA_SxM0AR または DMA_SxM1AR) を更新することができます。***

- *DMA_SxCR レジスタで CT ビットが"0"のとき DMA_SxM1AR レジスタに書き込むことができます。CT=1 のときに、このレジスタに書き込みを試みると、エラーフラグ（TEIF）がセットされ、ストリームが自動的に無効になります。*
- *DMA_SxCR レジスタで CT ビットが"1"のとき DMA_SxM0AR レジスタに書き込むことができます。CT=0 のときに、このレジスタに書き込みを試みると、エラーフラグ（TEIF）がセットされ、ストリームが自動的に無効になります。*

エラー条件を回避するためには、TCIF フラグがアサートされたらできるだけ早くベースアドレスを変更することを推奨します。なぜなら、この時点で、上記 2 つの内いずれかの条件にしたがって DMA_SxCR レジスタの CT の値に応じてターゲットメモリがメモリ 0 から 1（または 1 から 0）に変更されているためです。

他のすべてのモード（ダブルバッファモードを除く）に対し、ストリームが有効になるとすぐに、メモリアドレスレジスタは書き込み保護されます。

表 99. ダブルバッファモード（DBM=1）における転送元および転送先アドレスレジスタ

DMA_SxCR レジスタの DIR [1:0] ビット	方向	ソースアドレス	転送先アドレス
00	ペリフェラルからメモリ	DMA_SxPAR	DMA_SxM0AR/DMA_SxM1AR
01	メモリからペリフェラル	DMA_SxM0AR/DMA_SxM1AR	DMA_SxPAR
10	設定禁止(1)		
11	予約済み	-	-

1. ダブルバッファモードが有効になっている場合、サーキュラモードが自動的に有効になります。メモリ間モードはサーキュラモードと互換性がないため、ダブルバッファモードが有効になっているときは、メモリ間モードを設定することはできません。

15.3.12 プログラム可能なデータ幅、パッキング／アンパッキング、エンディアン形式

転送されるデータ項目の数は、ストリームを有効にする前に DMA_SxNDTR（転送データ項目数ビット、NDT）にプログラムする必要があります（フローコントローラがペリフェラルで、DMA_SxCR の PFCTRL ビットがセットされている場合を除く）。

内部 FIFO を使用する場合、転送元と転送先データのデータ幅は DMA_SxCR レジスタの PSIZE と MSIZEビットを介してプログラム可能です（8、16、または 32 ビットが可能）。

PSIZE と MSIZE が等しくない場合：

- DMA_SxNDTR レジスタに設定されている転送データ項目数のデータ幅は、ペリフェラルバスの幅（DMA_SxCR レジスタの PSIZE ビットで設定されている）と同じです。たとえば、ペリフェラルからメモリ、メモリからペリフェラル、またはメモリ間転送のケースで PSIZE[1:0] ビットがハーフワード用に設定されている場合、転送されるバイト数は 2×NDT です。
- DMA コントローラは、転送元と転送先の両方でリトルエンディアンアドレッシングのみ取り扱うことができます。これについては、*表 100：パッキング／アンパッキングおよびエンディアンの動作（ビット PINC=MINC=1）* で説明されています。

このパッキング／アンパッキング手順には、データが完全にパック／アンパックされる前に操作が中断された場合にデータが破損する危険性があります。したがって、データの一貫性を確保するため、ストリームはバースト転送を生成するよう設定することができます。この場合、バーストに属する各転送グループは分割できません（*セクション 15.3.13：シングル転送とバースト転送*を参照）。

ダイレクトモード（DMA_SxFCR レジスタで DMDIS=0 のとき）では、データのパッキング／アンパッキングはできません。この場合には、転送元と転送先の転送データ幅が異なってはなりません。両方が等しく、DMA_SxCR の PSIZE ビットで定義します（MSIZE ビットは無視されます）。

表 100. パッキング／アンパッキングおよびエンディアンの動作（ビット PINC=MINC=1）

AHB メモリポート幅	AHB ペリフェラルポート幅	転送するデータ項目の数（NDT）	メモリ転送番号	メモリポートアドレス／バイトレーン	ペリフェラル転送番号	ペリフェラルポートアドレス／バイトレーン	
						PINCOS = 1	PINCOS = 0
8	8	4	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
8	16	2	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1 2	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
8	32	1	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1	0x0 / B3|B2|B1|B0[31:0]	0x0 / B3|B2|B1|B0[31:0]
16	8	4	1 2	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
16	16	2	1 2	0x0 / B1|B0[15:0] 0x2 / B1|B0[15:0]	1 2	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
16	32	1	1 2	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]	1	0x0 / B3|B2|B1|B0[31:0]	0x0 / B3|B2|B1|B0[31:0]
32	8	4	1	0x0 / B3|B2|B1|B0[31:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
32	16	2	1	0x0 /B3|B2|B1|B0[31:0]	1 2	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
32	32	1	1	0x0 /B3|B2|B1|B0 [31:0]	1	0x0 /B3|B2|B1|B0 [31:0]	0x0 / B3|B2|B1|B0[31:0]

注： ***ペリフェラルポートは、転送元または転送先として設定可能です（メモリ間転送の場合はメモリ転送元としての設定も可能です）。***

PSIZE、MSIZE、および NDT[15:0] は、最後の転送が不完全にならないよう必ず設定する必要があります。これはペリフェラルポートのデータ幅（PSIZE ビット）がメモリポートのデータ幅（MSIZE ビット）より小さい場合に発生する可能性があります。この制約の一覧を*表 101* に示します。

表 101. PSIZE と MSIZE に対する NDT に関する制限事項

PSIZE[1:0] of DMA_SxCR	MSIZE[1:0] of DMA_SxCR	NDT[15:0] of DMA_SxNDTR
00 （8 ビット）	01 （16 ビット）	2 の倍数でなければならない
00 （8 ビット）	10 （32 ビット）	4 の倍数でなければならない
01 （16 ビット）	10 （32 ビット）	2 の倍数でなければならない

15.3.13 シングル転送とバースト転送

DMA コントローラは、シングル転送を生成することもできれば、4、8、または 16 ビートのインクリメンタルバースト転送を生成することもできます。

バーストのサイズは、DMA_SxCR レジスタの MBURST[1:0] と PBURST[1:0] ビットを使用して 2 つの AHB ポートに対して別々にソフトウェアで設定されます。

バーストサイズは、バースト内のビート数を示します。転送するバイト数ではありません。

データの一貫性を確保するため、バーストを形成する転送の各グループは分割できません。AHB 転送はロックされ、AHB バスマトリックスのアービタは、バースト転送シーケンス中 DMA マスタからバスの使用を奪取しません。

シングルまたはバーストの設定によって各 DMA リクエストによって AHB ペリフェラルポート上で開始される転送の数は異なります。

- AHB ペリフェラルポートがシングル転送用に設定されている場合、各 DMA リクエストによって、DMA_SxCR レジスタの PSIZE[1:0] ビットに応じた、バイト、ハーフワード、またはワードのデータ転送が生成されます。
- AHB ペリフェラルポートがバースト転送用に設定されている場合、各 DMA リクエストによって、DMA_SxCR レジスタの PBURST[1:0] ビットと PSIZE[1:0] ビットに応じた、4、8、または 16 ビートのバイト、ハーフワード、またはワードの転送が生成されます。

MBURST と MSIZE ビットに関しても、AHB メモリポートに対して上記と同じことを考慮する必要があります。

ダイレクトモードでは、ストリームはシングル転送のみ生成することができ、MBURST[1:0] と PBURST[1:0] ビットはハードウェアで決まってしまいます。

アドレスポインタ（DMA_SxPAR または DMA_SxM0AR レジスタ）は、バーストブロック内のすべての転送が転送サイズに等しいアドレス境界で整列するように選択する必要があります。

バースト設定は、AHB プロトコルを守るよう選択する必要があります。ここで、1 個のスレーブに割り当てることができる最小アドレス空間が 1 KB であるため、バーストが 1 KB アドレス境界を越えることは*できません*。つまり、バーストブロック転送が 1 KB アドレス境界を超えてはならず、超えた場合には AHB エラーが生成される可能性があります。このエラーは DMA レジスタで報告されません。

15.3.14 FIFO

FIFO の構造

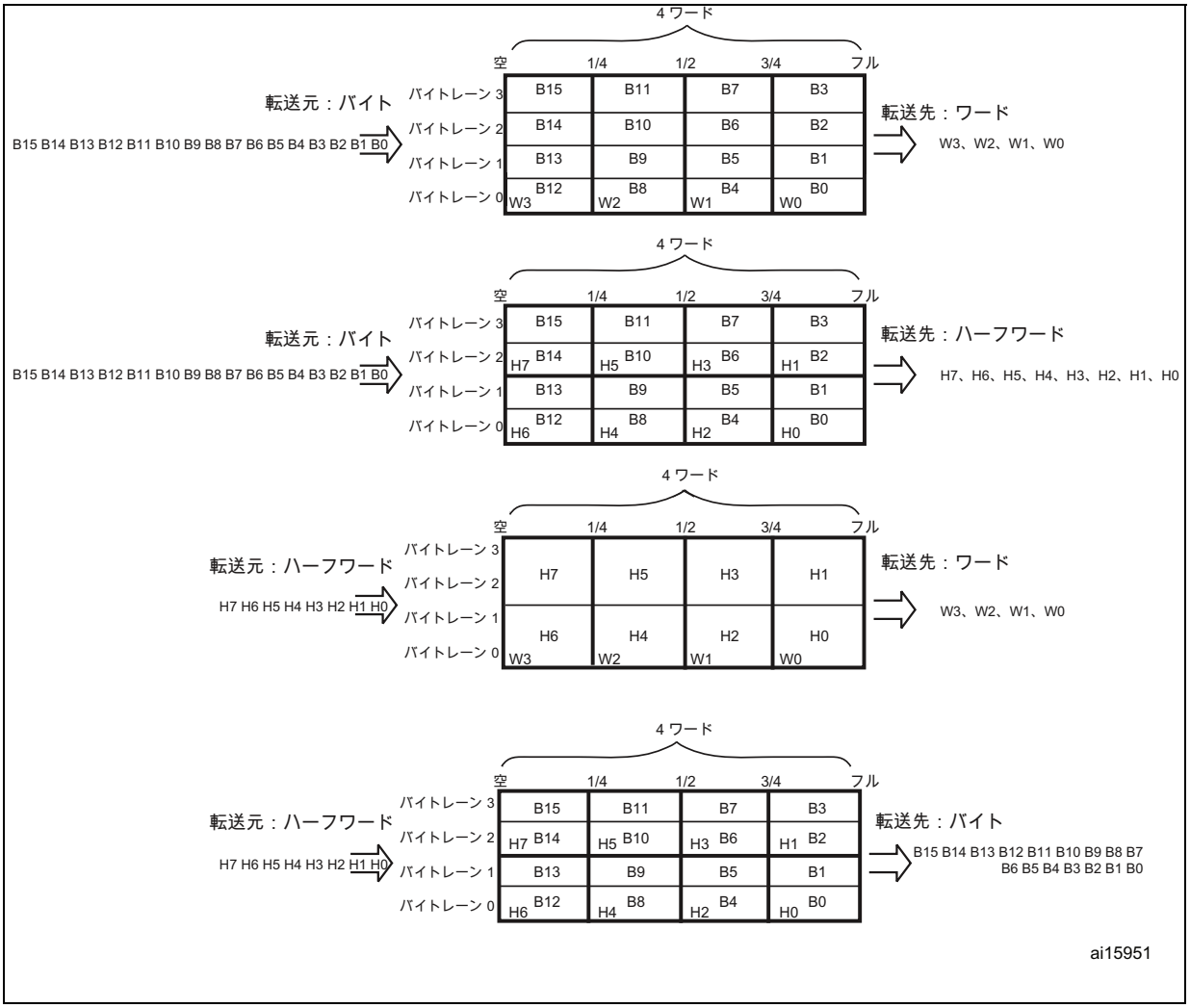
FIFO は転送元からのデータを転送先へ送信する前に一時的に格納するために使用します。

各ストリームには独立した 4 ワードの FIFOがあり、閾値レベルはソフトウェアで 1/4、1/2、3/4、またはフルに設定できます。

FIFO 閾値レベルを使用できるようにするには、DMA_SxFCR レジスタの DMDIS ビットをセットしてダイレクトモードを無効にする必要があります。

FIFO の構造は、転送元および転送先のデータ幅によって異なり、*図 73：FIFO の構造*で説明しています。

図 73. FIFO の構造



FIFO 閾値とバーストの設定

FIFO の閾値（DMA_SxFCR レジスタの FTH[1:0] ビット）とメモリバーストのサイズ（DMA_SxCR レジスタの MBURST[1:0]）を選択する場合は注意が必要です。FIFO 閾値が指す内容はメモリバースト転送の整数倍に一致する必要があります。一致しない場合、ストリームが有効なとき FIFO エラー（DMA_HISR または DMA_LISR レジスタの FEIFx フラグ）が発生し、ストリームが自動的に無効になります。可能な設定と禁止されている設定の説明は*表 102* を参照してください。禁止されている設定は、表内でグレイでハイライトされています。

表 102. FIFO 閾値設定

MSIZE	FIFO レベル	MBURST = INCR4	MBURST = INCR8	MBURST = INCR16
バイト	1/4	4 ビートの 1 バースト	設定禁止	設定禁止
	1/2	4 ビートの 2 バースト	8 ビートの 1 バースト	
	3/4	4 ビートの 3 バースト	設定禁止	
	フル	4 ビートの 4 バースト	8 ビートの 2 バースト	16 ビートの 1 バースト

表 102. FIFO 閾値設定（続き）

MSIZE	FIFO レベル	MBURST = INCR4	MBURST = INCR8	MBURST = INCR16
ハーフワード	1/4	設定禁止	設定禁止	設定禁止
	1/2	4 ビートの 1 バースト		
	3/4	設定禁止		
	フル	4 ビートの 2 バースト	8 ビートの 1 バースト	
ワード	1/4	設定禁止	設定禁止	
	1/2			
	3/4			
	フル	4 ビートの 1 バースト		

いかなる場合でも、バーストサイズにデータサイズを乗じた値が FIFO サイズを超えることはできません（可能なデータサイズは1（バイト）、2（ハーフワード）または 4（ワード））。

次のいずれかの条件が発生した場合、DMA 転送の終了時に、バースト転送が不完全となる可能性があります。

- AHB ペリフェラルポート設定において、データ項目の合計数（DMA_SxNDTR レジスタに設定）が、バーストサイズにデータサイズを乗じた値の倍数ではない
- AHB メモリポート設定において、メモリに転送する FIFO 内の残りのデータ項目数が、バーストサイズにデータサイズを乗じた値の倍数ではない

このような場合、DMA ストリーム設定時にバーストトランザクションが要求されていても、転送すべき残りのデータは、DMA によってシングルモードで管理されます。

注： ***ペリフェラル AHB ポートでバースト転送が要求されており、FIFO が使用されている場合（DMA_SxCR レジスタの DMDIS=1）、DMA ストリーム方向に応じて、恒久的なアンダーランまたはオーバーラン状態を回避するため、次のルールを守ることが必要です。***

（PBURST × PSIZE）= FIFO_SIZE（4 ワード）の場合、PSIZE = 1、2、または 4 であり、かつ、PBURST= 4、8、または 16 の時、FIFO_Threshold = 3/4 は 禁止。

このルールによって、ペリフェラルからの要求を一度に処理するのに十分な FIFO スペースを確実に確保できます。

FIFO フラッシュ

FIFO は DMA_SxCR レジスタの EN ビットをリセットすることによりストリームが無効化された場合、およびストリームが“ペリフェラルからメモリ”転送またはメモリ間転送を管理するよう設定された場合、一掃することができます。ストリームを無効化したときに FIFO にデータが残っている場合、DMA コントローラは引き続き残りのデータを転送先へ転送します（ストリームが事実上無効化されているとしても）。このフラッシュが完了すると、DMA_LISR または DMA_HISR レジスタの転送完了ステータスビット（TCIFx）がセットされます。

この場合、残りのデータカウンタ DMA_SxNDTR は値を保持し、現在いくつのデータ項目が転送先メモリに格納できるかを示します。

FIFO フラッシュ動作中に、FIFO 内のメモリに転送する残りのデータ項目数（バイト単位）がメモリデータ幅より少ない場合（たとえば、MSIZE がワードに設定されているとき FIFO 内に 2 バイト）、データは DMA_SxCR レジスタの MSIZE ビットに設定されたデータ幅で送信されます。つまり、メモリには不要な値が書き込まれることになります。ソフトウェアで DMA_SxNDTR レジスタを読み出すことにより、正しいデータが格納されたメモリ領域（開始アドレスと終了アドレス）を判断することができます。

FIFO 内の残りのデータ項目数がバーストサイズよりも小さい場合（AHB メモリポート上のバーストを管理するストリームを設定するため DMA_SxCR レジスタの MBURST ビットがセットされている場合）、FIFO フラッシュを完了するためシングルトランザクションが生成されます。

ダイレクトモード

デフォルトでは、FIFO はダイレクトモードで動作し（DMA_SxFCR の DMDIS ビットがリセットされています）、FIFO 閾値レベルは使用しません。このモードは、システムが各 DMA リクエスト後にメモリとの即時シングル転送を必要とする場合に便利です。

DMA がダイレクトモード（FIFOが無効）に設定されている場合、“メモリからペリフェラル”モードでデータを転送するため、ペリフェラルからの DMA リクエストが発生した際、即座にかつ確実にデータ転送が行われるように、DMA はデータをメモリから内部 FIFO へ 1 つプリロードします。

FIFO の飽和を避けるためには、対応するストリームを高い優先度で設定することを推奨します。

このモードは、次のような転送に制限されています。

- 転送元および転送先の転送幅が等しく、両方とも DMA_SxCR の PSIZE[1:0] ビットによって定義されている（MSIZE[1:0]ビットは無視される）
- バースト転送は不可（DMA_SxCR の PBURST[1:0] と MBURST[1:0] ビットは無視される）

メモリ間転送を実行する際はダイレクトモードを使用することはできません。

15.3.15 DMA 転送の完了

DMA_LISR または DMA_HISR ステータスレジスタの TCIFx ビットを設定することにより、さまざまなイベントによって転送を終了させることができます。

- DMA フローコントローラモード：
  - “メモリからペリフェラル”モードで DMA_SxNDTR カウンタがゼロに達した
  - （DMA_SxCR レジスタの EN ビットをクリアすることにより）ストリームが転送終了する前に無効化され、（転送がペリフェラルからメモリまたはメモリ間の場合）残りのすべてのデータが FIFO からメモリにフラッシュされた
- ペリフェラルフローコントローラモード：
  - 最後の外部バーストまたはシングルリクエストがペリフェラルから生成され、（DMA が“ペリフェラルからメモリ”モードで動作しているとき）残りのデータが FIFO からメモリへ転送された
  - ストリームがソフトウェアによって無効化され、（DMA が“ペリフェラルからメモリ”モードで動作しているとき）、残りのデータが FIFO からメモリへ転送された

注： ***転送完了は、“ペリフェラルからメモリ”モードの場合のみ、メモリに転送される FIFO 内の残りのデータによって異なります。この条件は、“メモリからペリフェラル”モードには適用されません。***

ストリームがノンサーキュラモードで設定されている場合、転送終了後（転送すべきデータの数がゼロに達したとき）、DMA は停止し（DMA_SxCR レジスタの EN ビットがハードウェアでクリアされます）、ソフトウェアによってストリームを再プログラムして再有効化（DMA_SxCR レジスタの EN ビットを設定）しない限り、DMA リクエストは処理されません。

15.3.16 DMA 転送の中断

DMA 転送は、いつでも中断して再開することもできれば、DMA 転送が終了する前に完全に無効化することもできます。

これには、2 つのケースがあります。

- ストリームは、転送を無効にし、停止した場所以降再開しません。ストリームを無効化するために DMA_SxCR レジスタの EN ビットをクリアする以外、特に行うことはありません。ストリームを無効にするのに時間を要する可能性があります（まず、進行中の転送を完了します）。転送の終了を示すために転送完了割り込みフラグ（DMA_LISR または DMA_HISR レジスタの TCIF）がセットされます。その時点で、DMA_SxCR の EN ビットの値が“0”となって、ストリームが中断されます。ソフトウェアによって、ストリームが中断される前に転送されたデータ項目の数を判断できるように、DMA_SxNDTR レジスタに、ストリームが停止した時点で残っているデータ項目の数を保持しています。
- ストリームは、DMA_SxNDTR レジスタ内の転送されるべき残りのデータ項目数が 0 に到達する前に転送を中断します。その目的は、ストリームを再度有効にすることで後から転送を再開することにあります。転送が停止した時点から再開するにあたり、ソフトウェアが既に収集したデータ項目数を知るためDMA_SxCR レジスタの EN ビットに書き込んで（その後それが“0”であることを確認します）ストリームを無効にした後で DMA_SxNDTR レジスタを読み出す必要があります。その後、
  - アドレスポインタを調整するために、ペリフェラルやメモリのアドレスを更新する必要があります。
  - SxNDTR レジスタは、転送すべきデータ項目の残数（ストリームを無効にしたとき読み出される値）で更新する必要があります。
  - ストリームは、その後再度有効化して転送が停止したところから再開することができます。

注： ***ストリームの中断による、転送の終了を示すため、転送完了割り込みフラグ（DMA_LISR または DMA_HISR の TCIF）がセットされることに注意してください。***

15.3.17 フローコントローラ

転送するデータの数を制御するものはフローコントローラとして知られています。このフローコントローラは、DMA_SxCR レジスタの PFCTRL ビットを使用して、各ストリームごとに個別に設定されます。

フローコントローラは次のように使用することができます。

- DMA コントローラ：この場合、DMA ストリームを有効にする前に、転送するデータ項目数を DMA_SxNDTR レジスタにソフトウェアでプログラムします。
- ペリフェラル転送元またはペリフェラル転送先：この場合、転送されるデータ項目数は不明です。最後のデータが転送されるとき、ペリフェラルがハードウェアによって DMA コントローラに知らせます。この機能は、転送の終了を通知できるペリフェラルでのみサポートされています。すなわち、

特定のストリームでペリフェラルフローコントローラを使用した場合、DMA_SxNDTR に書き込まれる値は DMA 転送に影響しません。実際に、いかなる値が書き込まれても、以下のスキームを守るためストリームが有効になるとすぐに、ハードウェアによって強制的に 0xFFFF になります。

- 想定されるストリームの中断：最終データを示すハードウェア信号（シングルまたはバースト）がペリフェラルによって送信される前に、DMA_SxCR レジスタの EN ビットをソフトウェアによって 0 にリセットし、ストリームを停止します。このようなケースでは、“ペリフェラルからメモリ”DMA 転送の場合、ストリームが停止し、FIFO フラッシュが作動します。対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、DMA の完了を示します。DMA

転送中に転送されたデータ項目数を知るには、DMA_SxNDTR レジスタを読み出し次の式を適用します。

– 転送されたデータ数 = 0xFFFF – DMA_SxNDTR

- 最終データを示すハードウェア信号の受信による通常ストリームの中断：ペリフェラルが最後の転送（シングルまたはバースト）を要求しこの転送が完了したときストリームは自動的に中断されます。対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、DMA 転送の完了を示します。転送されたデータ項目数を知るには、DMA_SxNDTR レジスタを読み出し上記と同じ式を適用します。
- DMA_SxNDTR レジスタが 0 に到達：対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、強制的な DMA 転送の完了を示します。最終データを示すハードウェア信号（シングルまたはバースト）がまだアサートされていない場合でも、ストリームは自動的に停止します。既に転送されたデータは失われません。つまり、ペリフェラルフローコントローラモードの場合でも、1 つの DMA のトランザクションにより最大 65535 のデータ項目が管理できることになります。

注： ***メモリ間モードに設定されているとき、DMA は常にフローコントローラとなり PFCTRL ビットはハードウェアによって強制的に 0 になります。***

ペリフェラルフロー制御モードでは、サーキュラモードは禁止されています。

15.3.18 実現可能な DMA 設定の概要

表 103 に、実現可能なさまざまな DMA 設定をまとめています。禁止されている設定は、表内でグレイでハイライトされています。

表 103. 実現可能な DMA 設定

DMA 転送モード	転送元	転送先	フローコントローラ	サーキュラモード	転送タイプ	ダイレクトモード	ダブルバッファモード
ペリフェラルからメモリ	AHB ペリフェラルポート	AHB メモリポート	DMA	設定可能	シングル	設定可能	設定可能
					バースト	設定禁止	
			ペリフェラル	設定禁止	シングル	設定可能	設定禁止
					バースト	設定禁止	
メモリからペリフェラル	AHB メモリポート	AHB ペリフェラルポート	DMA	設定可能	シングル	設定可能	設定可能
					バースト	設定禁止	
			ペリフェラル	設定禁止	シングル	設定可能	設定禁止
					バースト	設定禁止	
メモリ間転送	AHB ペリフェラルポート	AHB メモリポート	DMA のみ	設定禁止	シングル	設定禁止	設定禁止
					バースト		

15.3.19 ストリーム設定手順

DMA ストリーム x の設定は、次の手順で行います（x はストリーム番号です）。

1. ストリームが有効になっている場合、DMA_SxCR レジスタの EN ビットをリセットすることによって無効にし、その後、進行中のストリーム動作がないことを確認するため、このビットを読み出します。このビットに 0 を書き込んでもすぐに効果はありません。それは現在行われているすべての転送が完了した後、実際に 0 が書き込まれるからです。EN ビットが 0 として読み出される場合、ストリームが設定可能な状態であることを意味します。したがって、ストリームの設定を開始する前に EN ビットがクリアされるのを待つ必要があります。以前のデータブロックの DMA 転送からステータスレジスタ（DMA_LISR と DMA_HISR）に設定されたすべてのストリーム専用ビットは、ストリームを再度有効にする前にクリアする必要があります。
2. DMA_SxPAR レジスタにペリフェラルポートレジスタアドレスを設定します。ペリフェラルイベントの後、データはこのアドレスとペリフェラルポートの間で転送されます。
3. DMA_SxMA0R レジスタ（およびダブルバッファモードの場合は DMA_SxMA1R レジスタ）にメモリアドレスを設定します。ペリフェラルイベント後は、このメモリに対してデータを読み書きします。
4. 転送されるデータ項目の合計数を DMA_SxNDTR レジスタで設定します。各ペリフェラルイベントまたはバーストの各ビートの後、この値はデクリメントされます。
5. DMAMux1 を使用して、DMA チャネルに DMA リクエストラインを送ります。
6. ペリフェラルをフローコントローラとして使用することを意図し、かつこの機能をサポートしている場合、DMA_SxCR レジスタの PFCTRL ビットをセットします。
7. DMA_SxCRレジスタ の PL[1:0] ビットを使用して、ストリームの優先順位を設定します。
8. FIFO の使用方法を設定します（有効または無効にする、送信と受信における閾値）。
9. DMA_SxCR レジスタで以下の項目を設定します。データ転送方向、ペリフェラルとメモリのインクリメントモード/固定モード、シングルトランザクションまたはバーストトランザクション、ペリフェラルとメモリのデータ幅、サーキュラモード、ダブルバッファモード、ハーフ転送やフル転送後の割り込み、およびエラー。
10. DMA_SxCR レジスタの EN ビットをセットすることにより、ストリームを有効にします。

ストリームが有効になると、そのストリームに接続されているペリフェラルからの DMA リクエストを処理できます。

AHB 転送先ポート上でデータの半分が転送されると、1/2 転送フラグ（HTIF）がセットされ、1/2 転送割り込みイネーブルビット（HTIE）がセットされている場合には、割り込みが生成されます。転送が終了すると、転送完了フラグ（TCIF）がセットされ、転送完了割り込みイネーブルビット（TCIE）がセットされている場合には、割り込みが生成されます。

警告： **DMA ストリームリクエストに接続されているペリフェラルを切断するには、まず、ペリフェラルが接続されている DMA ストリームを切断し、次いで EN ビット=0 になるのを待つ必要があります。その後はじめてペリフェラルを安全に無効にすることができます。**

15.3.20 エラー管理

DMA コントローラによって、次のエラーを検出することができます。

- **転送エラー**：転送エラー割り込みフラグ（TEIFx）は次の場合にセットされます。
  - DMA 読み出しまたは書き込みアクセス時にバスエラーが発生した場合
  - ダブルバッファモードでは、メモリアドレスレジスタの書き込みアクセスはソフトウェアによって要求されますが、ストリームが有効になり、メモリアドレスレジスタへの書き込みによって影響を受けるのは現在のターゲットメモリです（*セクション 15.3.11：ダブルバッファモード*を参照）。
- **FIFO エラー**：FIFO エラー割り込みフラグ（FEIFx）は次の場合にセットされます。
  - FIFO アンダーラン状態が検出された場合
  - FIFO オーバーラン状態が検出された場合（リクエストと転送は DMA が内部で管理するため、このエラーはメモリ間モードでは検出されない）
  - FIFO　閾値レベルがメモリバーストのサイズに適合していないときにストリームが有効化された場合（*表 102：FIFO 閾値設定*を参照）
- **ダイレクトモードエラー**：ダイレクトモードで動作していて DMA_SxCR レジスタの MINC ビットがクリアされたときのみ、“ペリフェラルからメモリ”モードにおいてダイレクトモードエラー割り込みフラグ（DMEIFx）がセットされることがあります。このフラグは、メモリバスの使用権が与えられていなかったことにより、前回のデータがまだ完全にメモリに転送されていない状態で DMA リクエストが発生したときにセットされます。このケースでは、このフラグは 2 つのデータ項目が同じ転送先アドレスに連続して転送されたことを示し、転送先がこの状況を管理できない場合に、このことが問題となる可能性があります。

ダイレクトモードでは、FIFO エラーフラグが以下の条件でセットされることもあります。

- “ペリフェラルからメモリ”モードでは、複数のペリフェラルリクエストのためにメモリバスの使用が許可されない場合、FIFO が飽和（オーバーラン）することがあります。
- “メモリからペリフェラル”モードでは、ペリフェラルリクエストが発生する前に、メモリバスの使用が許可されていない場合、アンダーラン状態が発生する可能性があります。

バーストサイズと FIFO 閾値レベルが適合しないことによって TEIFx または FEIFx フラグがセットされた場合、欠陥のあるストリームは対応するストリーム設定レジスタ（DMA_SxCR）の EN ビットがハードウェアによってクリアされることにより、自動的に無効化されます。

オーバーランまたはアンダーラン状態によって DMEIFx または FEIFx フラグがセットされる場合、欠陥のあるストリームは自動的には無効化されず、DMA_SxCR レジスタの EN ビットをリセットすることによってストリームを無効化するかどうかはソフトウェアに任されています。なぜならば、この種のエラーが発生してもデータの損失がないためです。

DMA_LISR または DMA_HISR レジスタにあるストリームのエラー割り込みフラグ（TEIF、FEIF、DMEIF）がセットされると、DMA_SxCR または DMA_SxFCR レジスタの対応する割り込みイネーブルビット（TEIE、FEIE、DMIE）がセットされていれば、割り込みが発生します。

注： ***FIFO オーバーランまたはアンダーラン状態が発生した場合、オーバーランまたはアンダーラン状態がクリアされるまではペリフェラルリクエストがストリームによって承認されないため、データが失われることはありません。この承認に時間がかかりすぎた場合、ペリフェラル自体がその内部バッファのオーバーランまたはアンダーラン状態を検出し、データが失われる可能性があります。***

15.4 DMA 割り込み

各 DMA ストリームに対し、次のイベントにて割り込みを生成することができます。

- 1/2 転送終了
- 転送完了
- 転送エラー
- FIFO エラー（オーバーラン、アンダーラン、または FIFO レベルのエラー）
- ダイレクトモードエラー

柔軟性を持たせるため、*表 104* に示すよう個別の割り込みイネーブル制御ビットが使用できます。

表 104. DMA 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
1/2 転送	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE
FIFO オーバーラン／アンダーラン	FEIF	FEIE
ダイレクトモードエラー	DMEIF	DMEIE

注： ***イネーブル制御ビットを '1' に設定する前に、対応するイベントフラグをクリアする必要があります。さもないと直ちに割り込みが生成されてしまいます。***

15.5 DMA レジスタ

DMA レジスタには、ワード（32 ビット）単位でアクセスすることができます。

15.5.1 DMA ロー割り込みステータスレジスタ（DMA_LISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TCIF3	HTIF3	TEIF3	DMEIF3	Res.	FEIF3	TCIF2	HTIF2	TEIF2	DMEIF2	Res.	FEIF2
r	r	r	r	r	r	r	r		r	r	r	r	r		r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TCIF1	HTIF1	TEIF1	DMEIF1	Res.	FEIF1	TCIF0	HTIF0	TEIF0	DMEIF0	Res.	FEIF0
r	r	r	r	r	r	r	r		r	r	r	r	r		r

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **TCIFx**：ストリーム x 転送完了割り込みフラグ（x=3..0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送完了イベントは発生していません。

1：ストリーム x に転送完了イベントが発生しました。

ビット 26、20、10、4 **HTIFx**：ストリーム x 1/2 転送割り込みフラグ（x=3..0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に 1/2 転送イベントは発生していません。

1：ストリーム x に 1/2 転送イベントが発生しました。

ビット 25、19、9、3 **TEIFx**：ストリーム x 転送エラー割り込みフラグ（x=3..0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

ビット 24、18、8、2 **DMEIFx**：ストリーム x ダイレクトモードエラー割り込みフラグ（x=3..0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x にダイレクトモードエラーは発生していません。

1：ストリーム x にダイレクトモードエラーが発生しました。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **FEIFx**：ストリーム x FIFO エラー割り込みフラグ（x=3..0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に FIFO エラーイベントは発生していません。

1：ストリーム x に FIFO エラーイベントが発生しました。

15.5.2 DMA ハイ割り込みステータスレジスタ（DMA_HISR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TCIF7	HTIF7	TEIF7	DMEIF7	Res.	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Res.	FEIF6
				r	r	r	r		r	r	r	r	r		r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TCIF5	HTIF5	TEIF5	DMEIF5	Res.	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	Res.	FEIF4
				r	r	r	r		r	r	r	r	r		r

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **TCIFx**：ストリーム x 転送完了割り込みフラグ（x=7..4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送完了イベントは発生していません。

1：ストリーム x に転送完了イベントが発生しました。

ビット 26、20、10、4 **HTIFx**：ストリーム x 1/2 転送割り込みフラグ（x=7..4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に 1/2 転送イベントは発生していません。

1：ストリーム x に 1/2 転送イベントが発生しました。

ビット 25、19、9、3 **TEIFx**：ストリーム x 転送エラー割り込みフラグ（x=7..4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

ビット 24、18、8、2 **DMEIFx**：ストリーム x ダイレクトモードエラー割り込みフラグ（x=7..4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x にダイレクトモードエラーは発生していません。

1：ストリーム x にダイレクトモードエラーが発生しました。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **FEIFx**：ストリーム x FIFO エラー割り込みフラグ（x=7..4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に FIFO エラーイベントは発生していません。

1：ストリーム x に FIFO エラーイベントが発生しました。

15.5.3 DMA ロー割り込みフラグクリアレジスタ（DMA_LIFCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CTCIF3	CHTIF3	CTEIF3	CDMEIF3	Res.	CFEIF3	CTCIF2	CHTIF2	CTEIF2	CDMEIF2	Res.	CFEIF2
				w	w	w	w		w	w	w	w	w		w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CTCIF1	CHTIF1	CTEIF1	CDMEIF1	Res.	CFEIF1	CTCIF0	CHTIF0	CTEIF0	CDMEIF0	Res.	CFEIF0
				w	w	w	w		w	w	w	w	w		w

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **CTCIFx**：ストリーム x クリア転送完了割り込みフラグ（x=3..0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する TCIFx フラグをクリアします。

ビット 26、20、10、4 **CHTIFx**：ストリーム x クリア 1/2 転送割り込みフラグ（x=3..0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する HTIFx フラグをクリアします。

ビット 25、19、9、3 **CTEIFx**：ストリーム x クリア転送エラー割り込みフラグ（x=3..0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する TEIFx フラグをクリアします。

ビット 24、18、8、2 **CDMEIFx**：ストリーム x クリアダイレクトモードエラー割り込みフラグ（x=3..0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する DMEIFx フラグをクリアします。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **CFEIFx**：ストリーム x クリア FIFO エラー割り込みフラグ（x=3..0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する CFEIFx フラグをクリアします。

15.5.4 DMA ハイ割り込みフラグクリアレジスタ（DMA_HIFCR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CTCIF7	CHTIF7	CTEIF7	CDMEIF7	Res.	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	Res.	CFEIF6
				w	w	w	w		w	w	w	w	w		w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CTCIF5	CHTIF5	CTEIF5	CDMEIF5	Res.	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	Res.	CFEIF4
				w	w	w	w		w	w	w	w	w		w

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **CTCIFx**：ストリーム x クリア転送完了割り込みフラグ（x=7..4）
このビットに 1 を書き込むと DMA_HISR レジスタの対応する TCIFx フラグをクリアします。

ビット 26、20、10、4 **CHTIFx**：ストリーム x クリア 1/2 転送割り込みフラグ（x=7..4）
このビットに 1 を書き込むと DMA_HISR レジスタの対応する HTIFx フラグをクリアします。

ビット 25、19、9、3 **CTEIFx**：ストリーム x クリア転送エラー割り込みフラグ（x=7..4）
このビットに 1 を書き込むと DMA_HISR レジスタの対応する TEIFx フラグをクリアします。

ビット 24、18、8、2 **CDMEIFx**：ストリーム x クリアダイレクトモードエラー割り込みフラグ（x=7..4）
このビットに 1 を書き込むと DMA_HISR レジスタの対応する DMEIFx フラグをクリアします。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **CFEIFx**：ストリーム x クリア FIFO エラー割り込みフラグ（x=7..4）
このビットに 1 を書き込むと DMA_HISR レジスタの対応する CFEIFx フラグをクリアします。

15.5.5 DMA ストリーム x 設定レジスタ（DMA_SxCR）（x=0～7）

このレジスタは当該ストリームの設定に使用されます。

アドレスオフセット：0x10 + 0x18 ×*ストリーム番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MBURST [1:0]		PBURST[1:0]		Res.	CT	DBM	PL[1:0]	
							rw	rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:23 **MBURST**：メモリバースト転送設定
これらのビットは、ソフトウェアによってセット／クリアされます。
00：シングル転送
01：INCR4（4 ビートのインクリメンタルバースト）
10：INCR8（8 ビートのインクリメンタルバースト）
11：INCR16（16 ビートのインクリメンタルバースト）
これらのビットは保護されており EN が“0”の場合のみ書き込みできます。
ダイレクトモードでは、これらのビットは EN ビット=“1”になり次第ハードウェアによって強制的に 0x0 になります。

ビット 22:21 **PBURST[1:0]**：ペリフェラルバースト転送設定
これらのビットは、ソフトウェアによってセット／クリアされます。
00：シングル転送
01：INCR4（4 ビートのインクリメンタルバースト）
10：INCR8（8 ビートのインクリメンタルバースト）
11：INCR16（16 ビートのインクリメンタルバースト）
これらのビットは保護されており EN が“0”の場合のみ書き込みできます。
ダイレクトモードでは、これらのビットはハードウェアによって強制的に 0x0 になります。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CT**：現在のターゲット（ダブルバッファモード時のみ）
このビットは、ハードウェアによってセット／クリアされます。また、ソフトウェアによって書き込むこともできます。
0：現在のターゲットメモリはメモリ 0（DMA_SxM0AR ポインタによってアドレス指定）です。
1：現在のターゲットメモリはメモリ 1（DMA_SxM1AR ポインタによってアドレス指定）です。
このビットは、EN が '0' のときのみ書き込むことができ、最初の転送の対象メモリ領域を示します。
ストリームが有効になると、このビットはステータスフラグとして動作し、どのメモリ領域が現在のターゲットかを示します。

ビット 18 **DBM**：ダブルバッファモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送終了時にバッファを切り替えません

1：DMA 転送終了時にメモリターゲットを切り替えます

このビットは保護されており、EN が“0”のときのみ書き込みできます。

ビット 17:16 **PL[1:0]**：優先順位レベル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：低

01：中

10：高

11：最優先

これらのビットは保護されており EN が“0”のときのみ書き込みできます。

ビット 15 **PINCOS**：ペリフェラルインクリメントオフセットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

0：ペリフェラルアドレス計算のためのオフセットサイズは PSIZE と関連付けされています。

1：ペリフェラルアドレス計算のためのオフセットサイズは 4 に固定されています（32 ビットアライメント）。

ビット PINC=“0”の場合、このビットは意味を持ちません。

このビットは保護されており EN=“0”のときのみ書き込みできます。

ダイレクトモードが選択されているか、PBURSTが “00” ではない場合、ストリームが有効になったとき（EN ビット =“1”）このビットはハードウェアによって強制的にローになります。

ビット 14:13 **MSIZE[1:0]**：メモリデータサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：予約済み

これらのビットは保護されており EN が“0”のときのみ書き込みできます。

ダイレクトモードでは、EN ビット =“1” になり次第、MSIZE はハードウェアによって強制的に PSIZE と同じ値になります。

ビット 12:11 **PSIZE[1:0]**：ペリフェラルデータサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 ビット）

01：ハーフワード（16 ビット）

10：ワード（32 ビット）

11：予約済み

これらのビットは保護されており EN が“0”の場合のみ書き込みできます。

ビット 10 **MINC**：メモリインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：メモリアドレスポインタは固定されています。

1：メモリアドレスポインタは、各データ転送後にインクリメントされます（MSIZE に応じてインクリメント）

このビットは保護されており、EN が“0”のときのみ書き込みできます。

ビット 9 **PINC**：ペリフェラルインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：ペリフェラルアドレスポインタは固定されています

1：ペリフェラルアドレスポインタは、各データ転送後にインクリメントされます（PSIZE に応じてインクリメント）

このビットは保護されており、EN が“0”のときのみ書き込みできます。

ビット 8 **CIRC**：サーキュラモード

このビットは、ソフトウェアによってセット／クリアされ、かつハードウェアによってクリアできます。

0：サーキュラモードは無効です。

1：サーキュラモードは有効です。

ペリフェラルがフローコントローラ（PFCTRL ビット =1）となっていて、ストリームが有効なとき（ビット EN=1）、このビットは自動的にハードウェアによって 0 になります。

DBMビットがセットされている場合、ストリームが有効になり次第（EN ビット =“1”）ハードウェアにより強制的に 1 になります。

ビット 7:6 **DIR[1:0]**：データ転送方向

これらのビットは、ソフトウェアによってセット／クリアされます。

00：ペリフェラルからメモリ

01：メモリからペリフェラル

10：メモリ間転送

11：予約済み

これらのビットは保護されており EN が“0”のときのみ書き込みできます。

ビット 5 **PFCTRL**：ペリフェラルフローコントローラ

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA がフローコントローラ

1：ペリフェラルがフローコントローラ

このビットは保護されており、EN が“0”のときのみ書き込みできます。

メモリ間モードが選択されている場合（DIR[1:0] ビット =10）、このビットは自動的にハードウェアにより強制的に 0 になります。

ビット 4 **TCIE**：転送完了割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割り込みは無効です。

1：TC 割り込みは有効です。

ビット 3 **HTIE**：1/2 転送割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：HT 割り込みは無効です。

1：HT 割り込みは有効です。

ビット 2 **TEIE**：転送エラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割り込みは無効です。

1：TE 割り込みは有効です。

ビット 1 **DMEIE**：ダイレクトモードエラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DME 割り込みは無効です。

1：DME 割り込みは有効です。

ビット 0 **EN**：ストリームイネーブル／ロー読み出し時のストリームレディフラグ

このビットは、ソフトウェアによってセット／クリアされます。

0：ストリームは無効です。

1：ストリームは有効です。

このビットは、次のときハードウェアによってクリアできます。

– DMA 転送終了時（ストリームが設定準備完了）

– AHB マスタバス上で転送エラーが発生した場合

– メモリ AHB ポート上の FIFO 閾値がバーストの大きさと適合しない場合

このビットが 0 として読み出された場合、ソフトウェアによって設定レジスタと FIFO ビットレジスタをプログラムすることができます。EN ビットが 1 として読み出された場合、これらのレジスタの書き込みは禁止されています。

注： 新しい転送を開始するために EN ビットを“1”に設定する前に、DMA_LISR または DMA_HISR レジスタのストリームに対応するイベントフラグをクリアする必要があります。

15.5.6 DMA ストリーム x データ数レジスタ（DMA_HIFCR）（x=0～7）

アドレスオフセット：0x14 + 0x18 ×*ストリーム番号*

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送するデータ項目の数

転送されるデータ項目数（0～65535）。このレジスタに書き込めるのは、ストリームが無効のときだけです。ストリームが有効になると、このレジスタは読み出し専用になり、送信されるべき残りのデータ項目を示します。このレジスタは、各 DMA 転送後にデクリメントされます。

転送が完了すると、このレジスタは、ゼロのままにすることもできれば（ストリームが通常モードのとき）、次の場合には事前にプログラムされた値を自動的に再ロードすることもできます。

– ストリームがサーキュラモードに設定されている場合

– EN ビットを“1”に設定することにより、ストリームが再び有効になった場合

このレジスタの値が 0 の場合、ストリームが有効であっても、トランザクションは処理できません。

15.5.7 DMA ストリーム x ペリフェラルアドレスレジスタ（DMA_SxPAR）（x=0～7）

アドレスオフセット：0x18 + 0x18 ×*ストリーム番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PAR[31:0]**：ペリフェラルアドレス

データの読み出し／書き込みが行われるペリフェラルデータレジスタのベースアドレス。

これらのビットは書き込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書き込みできます。

15.5.8 DMA ストリーム x メモリ 0 アドレスレジスタ（DMA_SxM0AR）（x=0～7）

アドレスオフセット：0x1C + 0x18 ×*ストリーム番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M0A[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M0A[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **M0A[31:0]**：メモリ 0 アドレス

データの読み出し／書き込みが行われるメモリ領域 0 のベースアドレス。

これらのビットは書き込み保護されています。次の場合にのみ書き込みが可能です。

- ストリームが無効（DMA_SxCR レジスタの EN =“0”）の場合、または
- ストリームが有効（DMA_SxCR レジスタの EN =“1”）で DMA_SxCR レジスタのビット CT=“1”（ダブルバッファモード）の場合。

15.5.9 DMA ストリーム x メモリ 1 アドレスレジスタ（DMA_SxM1AR）（x=0～7）

アドレスオフセット：0x20 + 0x18 ×*ストリーム番号*

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M1A[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M1A[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **M1A[31:0]**：メモリ 1 アドレス（ダブルバッファモードの場合に使用）

データの読み出し／書き込みが行われるメモリ領域 1 のベースアドレス。

このレジスタは、ダブルバッファモードでのみ使用されます。

これらのビットは書き込み保護されています。次の場合にのみ書き込みが可能です。

– ストリームが無効（DMA_SxCR レジスタの EN =“0”）の場合、または

– ストリームが有効（DMA_SxCR レジスタの EN =“1”）で DMA_SxCR レジスタのビット CT=“0”の場合。

15.5.10 DMA ストリーム x FIFO 制御レジスタ（DMA_SxFCR）（x=0～7）

アドレスオフセット：0x24 + 0x24 ×*ストリーム番号*

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH[1:0]	
								rw		r	r	r	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **FEIE**：FIFO エラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：FE 割り込みは無効です。

1：FE 割り込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5:3 **FS[2:0]**: FIFO ステータス

これらのビットは読み出し専用です。

000：0 < FIFO レベル < 1/4

001：1/4 ≤ FIFO レベル < 1/2

010：1/2 ≤ FIFO レベル < 3/4

011：3/4 ≤ FIFO レベル < フル

100：FIFO はエンプティです。

101：FIFO はフルです。

その他：意味を持ちません。

これらのビットは、ダイレクトモードには無関係です（DMDIS ビットがゼロ）。

ビット 2 **DMDIS**：ダイレクトモードディセーブル

このビットは、ソフトウェアによってセット／クリアされます。ハードウェアによってセットすることができます。

0：ダイレクトモードは有効です。

1：ダイレクトモードは無効です。

このビットは保護されており、EN が“0”のときのみ書き込みできます。

このビットは、メモリ間モードが選択されており（DMA_SxCR の DIR ビットが “10”）、DMA_SxCR レジスタの EN ビットが“1”の場合、ハードウェアによってセットされます。その理由は、ダイレクトモードではメモリ間転送設定が使用できないからです。

ビット 1:0 **FTH[1:0]**：FIFO 閾値選択

これらのビットは、ソフトウェアによってセット／クリアされます。

00：FIFO フルの 1/4

01：FIFO フルの 1/2

10：FIFO フルの 3/4

11：FIFO フル

これらのビットは、DMIS 値がゼロのときダイレクトモードで使用されません。

これらのビットは保護されており EN が“0”のときのみ書き込みできます。

参考資料

15.5.11 DMA レジスタマップ

表 105 に DMA レジスタの要約を示します。

表 105. DMA レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA_LISR	Res.	Res.	Res.	Res.	TCIF3	HTIF3	TEIF3	DMEIF3	Res.	FEIF3	TCIF2	HTIF2	TEIF2	DMEIF2	Res.	FEIF2	Res.	Res.	Res.	Res.	TCIF1	HTIF1	TEIF1	DMEIF1	Res.	FEIF1	TCIF0	HTIF0	TEIF0	DMEIF0	予約済み	FEIF0
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0004	DMA_HISR	Res.	Res.	Res.	Res.	TCIF7	HTIF7	TEIF7	DMEIF7	Res.	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Res.	FEIF6	Res.	Res.	Res.	Res.	TCIF5	HTIF5	TEIF5	DMEIF5	Res.	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	予約済み	FEIF4
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0008	DMA_LIFCR	Res.	Res.	Res.	Res.	CTCIF3	CHTIF3	TEIF3	CDMEIF3	予約済み	CFEIF3	CTCIF2	CHTIF2	CTEIF2	CDMEIF2	Res.	CFEIF2	Res.	Res.	Res.	Res.	CTCIF1	CHTIF1	CTEIF1	CDMEIF1	Res.	CFEIF1	CTCIF0	CHTIF0	CTEIF0	CDMEIF0	予約済み	CFEIF0
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x000C	DMA_HIFCR	Res.	Res.	Res.	Res.	CTCIF7	CHTIF7	CTEIF7	CDMEIF7	予約済み	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	Res.	CFEIF6	Res.	Res.	Res.	Res.	CTCIF5	CHTIF5	CTEIF5	CDMEIF5	Res.	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	予約済み	CFEIF4
	リセット値	Res.	Res.	Res.	Res.	0	0	0	0		0	0	0	0	0	Res.	0					0	0	0	0		0	0	0	0	0		0
0x0010	DMA_S0CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MBURST[1:0]		PBURST[1:0]		Res.	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA_S0NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA_S0PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA_S0M0AR	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	DMA_S0M1AR	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	DMA_S0FCR	Res.	Res.	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x002C	DMA_S1NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	DMA_S1PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	DMA_S1M0AR	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 105. DMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0038	**DMA_S1M1AR**	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	**DMA_S1FCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0044	**DMA_S2NDTR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	**DMA_S2PAR**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	**DMA_S2M0AR**	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050	**DMA_S2M1AR**	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	**DMA_S2FCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x005C	**DMA_S3NDTR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	**DMA_S3PAR**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	**DMA_S3M0AR**	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0068	**DMA_S3M1AR**	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x006C	**DMA_S3FCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0074	**DMA_S4NDTR**	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0078	**DMA_S4PAR**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x007C	**DMA_S4M0AR**	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0080	**DMA_S4M1AR**	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	**DMA_S4FCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1

表 105. DMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x008C	DMA_S5NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0090	DMA_S5PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094	DMA_S5M0AR	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0098	DMA_S5M1AR	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x009C	DMA_S5FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x00A4	DMA_S6NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A8	DMA_S6PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00AC	DMA_S6M0AR	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	DMA_S6M1AR	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	DMA_S6FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1
0x00BC	DMA_S7NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:.]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C0	DMA_S7PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C4	DMA_S7M0AR	M0A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8	DMA_S7M1AR	M1A[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00CC	DMA_S7FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH [1:0]	
	リセット値																									0		1	0	0	0	0	1

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

参考資料

16 基本ダイレクトメモリアクセスコントローラ（BDMA）

16.1 概要

BDMA（Basic Direct Memory Access：基本ダイレクトメモリアクセス）コントローラは、ペリフェラルとメモリ間、およびメモリ間でハイスピードデータ転送を行うために使用します。DMA により、CPU を動作させずに、データを高速で移動することができます。これにより、CPU リソースを他の操作のためにあけておくことができます。

BDMA コントローラは、合計 8 本のチャネルを搭載し、それぞれが 1 つ以上のペリフェラルからのメモリアセスリクエストを管理する役割を担っています。また、DMA リクエスト間の優先順位を操作するためのアービタも内蔵しています。

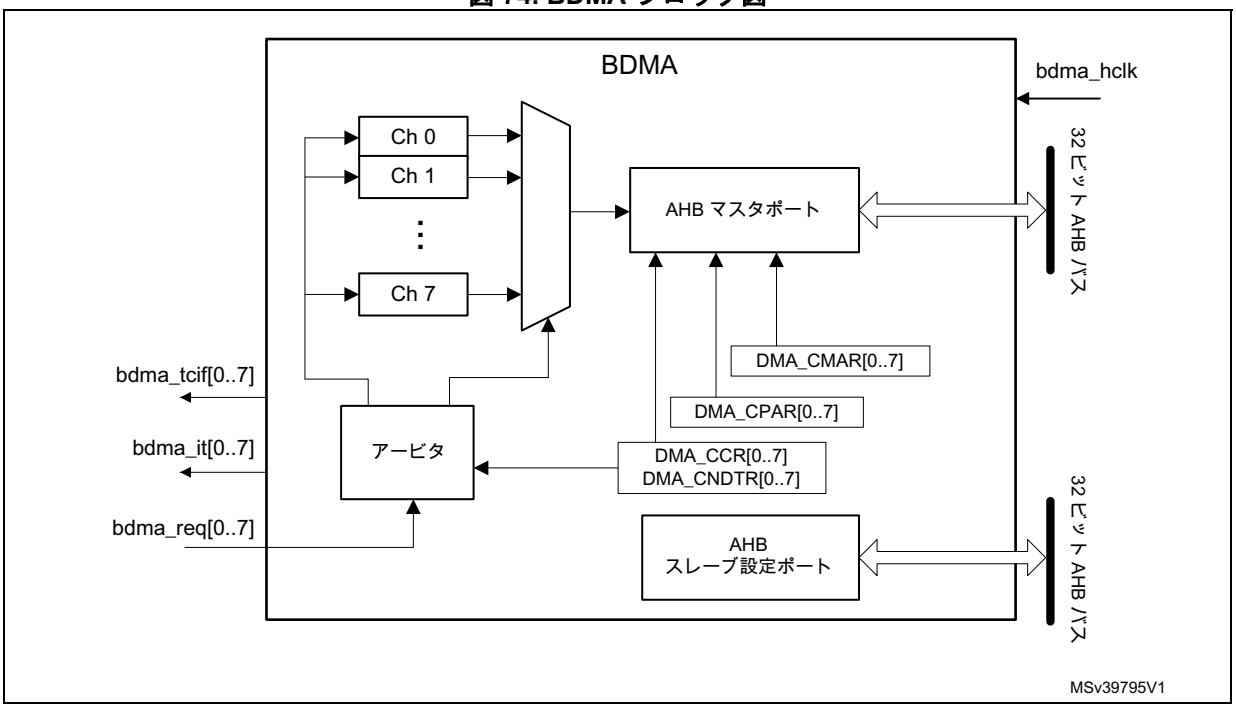
16.2 BDMA の主な機能

- 8 本の個別に設定可能なチャネル（リクエスト）を内蔵しています。
- 各チャネルは、専用のハードウェア DMA リクエストに接続されており、ソフトウェアトリガもサポートしています。この設定はソフトウェアで行われます。
- 1 つの BDMA コントローラのチャネルからのリクエスト間の優先順位はソフトウェアでプログラム可能です（最高、高、中、低の 4 レベル）。またはレベルが等しい場合はハードウェアによって決まります（リクエスト 1 はリクエスト 2 よりも優先、など）。
- 転送元および転送先の転送サイズ（バイト、ハーフワード、ワード）は個別に設定され、パッキング／アンパッキングをエミュレートします。転送元／転送先のアドレスは、データサイズに基づいて整列させてください。
- サーキュラバッファ管理をサポートします。
- 各チャネルに対する単一割り込みリクエストにおいて、3 つのイベントフラグ（DMA 1/2 転送、DMA 転送完了、DMA 転送エラー）の論理和がとられます。
- メモリ間の転送
- ペリフェラルからメモリ、メモリからペリフェラル、およびペリフェラル間の転送
- D3 ドメインメモリとペリフェラルに対して、転送元および転送先としてアクセス
- プログラム可能な転送データ数：最大65535

16.3 BDMA の機能説明

以下にブロック図を示します。

図 74. BDMA ブロック図



BDMA コントローラは、システムバスをその他のシステムマスタと共有することでダイレクトメモリ転送を行います。DMA リクエストは、CPU と DMA のターゲット転送先が同じである場合に（メモリまたはペリフェラル）、多少のバスサイクルの間、CPU のシステムバスへのアクセスを停止することがあります。バスマトリックスはラウンドロビンスケジューリングを実装します。これにより、CPU のシステムバス帯域幅（メモリとペリフェラルの両方への）の少なくとも半分を確保できます。

16.3.1 BDMA トランザクション

イベントの後、ペリフェラルは BDMA コントローラにリクエスト信号を送信します。BDMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。BDMA コントローラがペリフェラルにアクセスするとすぐに、BDMA コントローラはペリフェラルに確認応答を送信します。ペリフェラルは、BDMA コントローラからの確認応答を受け取るとすぐに、そのリクエストを解除します。ペリフェラルによってリクエストが無効にされると、DMA コントローラは確認応答を解除します。さらにリクエストがある場合、ペリフェラルは次のトランザクションを開始できます。

要約すると、各 BDMA 転送は次の 3 つの動作で構成されています。

- ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ位置からのデータのロード。最初の転送に使用される開始アドレスは、BDMA_CPARx または BDMA_CMARx レジスタでプログラムされたペリフェラル／メモリのベースアドレスです。

- ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ位置にロードされたデータの格納。最初の転送に使用される開始アドレスは、BDMA_CPARx または BDMA_CMARx レジスタでプログラムされたペリフェラル／メモリのベースアドレスです。
- まだ実行される予定のトランザクションの数を保持している BDMA_CNDTRx レジスタのポストデクリメント。

16.3.2 アービタ

アービタは、チャネルリクエストをその優先順位に基づいて管理し、ペリフェラル／メモリアクセスシーケンスを起動します。

優先順位は 2 段階で管理されます。

- ソフトウェア：各チャネルの優先順位は、BDMA_CCRx レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア : 2 つのリクエストのソフトウェア優先順位レベルが同じである場合、小さな番号のチャネルが大きな番号のチャネルよりも優先されます。たとえば、チャネル 2 はチャネル 4 よりも優先されます。

16.3.3 BDMA チャネル

各チャネルは、ある固定アドレスにあるペリフェラルレジスタとメモリアドレスの間の DMA 転送を処理します。転送データ数（最大 65535）はプログラム可能です。転送データ項目数を格納しているレジスタは、トランザクションの終了ごとにデクリメントされます。

プログラム可能なデータサイズ

ペリフェラルおよびメモリの転送データサイズは、BDMA_CCRx レジスタの PSIZE および MSIZE ビットを介して完全にプログラム可能です。

ポインタのインクリメント

ペリフェラルおよびメモリのポインタは、BDMA_CCRx レジスタの PINC および MINC ビットに応じて、各トランザクション後に任意で自動的にポストインクリメントすることができます。インクリメントモードを有効にした場合、次の転送アドレスは、選択されたデータサイズに応じて 1、2、または 4 だけインクリメントされた前回の転送アドレスとなります。最初の転送アドレスは、BDMA_CPARx/BDMA_CMARx レジスタでプログラムされたアドレスとなります。転送動作中は、これらのレジスタは最初にプログラムされた値を保持します。現在の転送アドレス（現在の内部ペリフェラル／メモリアドレスレジスタ内）に対するソフトウェアによるアクセスはできません。

チャネルがノンサーキュラモードに設定されている場合、転送終了後（転送すべきデータ項目数がゼロに達したとき）、DMA リクエストは処理されません。新たな転送すべきデータ項目数を BDMA_CNDTRx レジスタに再ロードするには、DMA チャネルを無効にする必要があります。

注： ***DMA チャネルが無効にされると、DMA レジスタはリセットされません。DMA チャネルレジスタ（BDMA_CCRx、BDMA_CPARx、BDMA_CMARx）は、チャネル設定フェーズでプログラムされた初期値のままです。***

サーキュラモードでは、転送終了後、BDMA_CNDTRx レジスタには最初にプログラムした値が自動的に再ロードされます。現在の内部アドレスレジスタには、BDMA_CPARx/BDMA_CMARx レジスタからのベースアドレス値が再ロードされます。

チャネル設定手順

DMA チャネル x（x はチャネル番号）の設定は、以下のシーケンスで行います。

1. BDMA_CPARx レジスタに、ペリフェラルレジスタアドレスをセットします。ペリフェラルイベント後は、メモリとの転送に使用されるこのアドレスとデータをやり取りします。
2. BDMA_CMARx レジスタにメモリアドレスをセットします。ペリフェラルイベント後は、このメモリに対してデータを読み書きします。
3. 転送すべきデータの総数を BDMA_CNDTRx レジスタに設定します。毎回のペリフェラルイベント後に、この値はデクリメントされます。
4. BDMA_CCRx レジスタの PL[1:0] ビットを使用して、チャネルの優先順位を設定します。
5. データ転送方向、サーキュラモード、ペリフェラルとメモリのインクリメントモード、ペリフェラルとメモリのデータサイズ、1/2 転送やフル転送後の割り込みを BDMA_CCRx レジスタに設定します。
6. BDMA_CCRx レジスタの ENABLE ビットをセットすることでチャネルを有効にします。

チャネルが有効になると、そのチャネルに接続されているペリフェラルからの DMA リクエストを処理することができます。

バイトの半分が転送されると、1/2 転送フラグ（HTIF）がセットされ、1/2 転送割り込みイネーブルビット（HTIE）がセットされている場合には、割り込みが生成されます。転送が終了すると、転送完了フラグ（TCIF）がセットされ、転送完了割り込みイネーブルビット（TCIE）がセットされている場合には、割り込みが生成されます。

サーキュラモード

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（ADC スキャンモードなど）を処理することができます。この機能は、BDMA_CCRx レジスタの CIRC ビットを使用して有効にできます。サーキュラモードが有効になると、転送されるデータ項目数はチャネル設定フェーズでプログラムされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。

メモリ間モード

DMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても機能します。このモードは、メモリ間モードと呼ばれます。

BDMA_CCRx レジスタの MEM2MEM ビットがセットされると、チャネルは、ソフトウェアが BDMA_CCRx レジスタの Enable ビット（EN）をセットすることによってそのチャネルを有効にするとすぐに、転送を開始します。BDMA_CNDTRx レジスタがゼロに達すると、転送は停止します。メモリ間モードとサーキュラモードを同時に使用することはできません。

16.3.4 プログラム可能なデータ幅、データの整列、およびエンディアン

PSIZE と MSIZE が等しくない場合、DMA は、*表 106：プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合）*に記載されているとおりに必要なデータの整列を行います。

表 106. プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合）

転送元ポートの幅	転送先ポートの幅	転送するデータ項目の数（NDT）	転送元の内容：アドレス／データ	転送動作	転送先の内容：アドレス／データ
8	8	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に B1[7:0] を 0x1 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に B2[7:0] を 0x2 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に B3[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3
8	16	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 00B0[15:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 00B1[15:0] を 0x2 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 00B2[15:0] を 0x4 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 00B3[15:0] を 0x6 に書き込みます。	@0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3
8	32	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 000000B0[31:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 000000B1[31:0] を 0x4 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 000000B2[31:0] を 0x8 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 000000B3[31:0] を 0xC に書き込みます。	@0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3
16	8	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B2[7:0] を 0x1 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B4[7:0] を 0x2 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B6[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6
16	16	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B3B2[15:0] を 0x2 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x4 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B7B6[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6
16	32	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に 0000B1B0[31:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に 0000B3B2[31:0] を 0x4 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に 0000B5B4[31:0] を 0x8 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に 0000B7B6[31:0] を 0xC に書き込みます。	@0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6
32	8	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B4[7:0] を 0x1 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B8[7:0] を 0x2 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BC[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC
32	16	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x2 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B9B8[15:0] を 0x4 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BDBC[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC
32	32	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B3B2B1B0[31:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B7B6B5B4[31:0] を 0x4 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に BBBAB9B8[31:0] を 0x8 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BFBEBDBC[31:0] を 0xC に書き込みます。	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC

バイトまたはハーフワード書き込み動作をサポートしていない AHB ペリフェラルへの対応

DMA が AHB のバイトまたはハーフワード書き込み動作を起動すると、データは HWDATA[31:0] バスの未使用のレーンに複製されます。そのため、使用中の AHB スレーブペリフェラルがバイト／ハーフワード書き込み動作をサポートしておらず（HSIZE がペリフェラルで使用されていない）、かつエラーが 1 件も出ていない場合は、以下の2つの例に示されるように、DMA は 32 HWDATA ビットを書き込みます。

- ハーフワード “0xABCD”を書き込むには、DMA は HWDATA バスを“0xABCDABCD”（HSIZE = ハーフワード）にセットします。
- バイト“0xAB”を書き込むには、DMA は HWDATA バスを“0xABABABAB”（HSIZE = バイト）にセットします。

AHB/APB ブリッジが AHB 32 ビットスレーブペリフェラルである（HSIZE の値は考慮せず）と仮定した場合、以下のようなやり方で、AHB バイトまたはハーフワードのあらゆる動作を 32 ビット APB 動作に変換します。

- データ「0xB0」を 0x0（または 0x1、0x2、0x3）に書き込む AHB バイト書き込み動作は、データ「0xB0B0B0B0」を 0x0 に書き込む APB ワード書き込み動作に変換されます。
- データ“0xB1B0 ”を 0x0（または 0x2）に書き込む AHB ハーフワード書き込み動作は、データ“0xB1B0B1B0”を 0x0 に書き込む APB ワード書き込み動作に変換されます。

たとえば、APB バックアップレジスタ（32 ビットアドレス境界に整列された 16 ビットレジスタ）に書き込むには、ソフトウェアで転送元のメモリサイズ（MSIZE）を「16 ビット」に、転送先のペリフェラルサイズ（PSIZE）を「32 ビット」に設定する必要があります。

16.3.5 エラー管理

DMA 転送エラーは、予約済みアドレス空間に対する読み出しや書き込みによって発生する可能性があります。DMA 読み出しまたは書き込みアクセス中に DMA 転送エラーが発生した場合、障害のあるチャネルは、ハードウェアが対応するチャネル設定レジスタ（BDMA_CCRx）の該当する EN ビットをクリアすることにより、自動的に無効化されます。BDMA_IFR レジスタのチャネルの転送エラー割り込みフラグ（TEIF）がセットされ、BDMA_CCRx レジスタの転送エラー割り込みイネーブルビット（TEIE）がセットされている場合には、割り込みが生成されます。

16.3.6 BDMA 割り込み

割り込みは、DMA チャネルごとの 1/2 転送、転送完了、転送エラーに対して生成されます。高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 107. BDMA 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
1/2 転送	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

BDMA リクエストマッピングを*セクション 17.3.3 : DMAMUX2 の配置*に示します。

16.4 BDMA レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）単位でアクセスできます。

16.4.1 DMA 割り込みステータスレジスタ（BDMA_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEIF8	HTIF8	TCIF8	GIF8	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31、27、23、19、15、11、7、3 **TEIFx**：チャネル x 転送エラーフラグ（x = 1 ～ 8）

このビットは、ハードウェアによってセットされます。BDMA_IFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x に転送エラーは発生していません。

1：チャネル x に転送エラー（TE）が発生しました。

ビット 30、26、22、18、14、10、6、2 **HTIFx**：チャネル x 1/2 転送フラグ（x = 1 ～ 8）

このビットは、ハードウェアによってセットされます。BDMA_IFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x に 1/2 転送（HT）イベントは発生していません。

1：チャネル x に 1/2 転送（HT）イベントが発生しました。

ビット 29、25、21、17、13、9、5、1 **TCIFx**：チャネル x 転送完了フラグ（x = 1 ～ 8）

このビットは、ハードウェアによってセットされます。BDMA_IFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x の転送完了（TC）イベントは発生していません。

1：チャネル x に転送完了（TC）イベントが発生しました。

ビット 28、24、20、16、12、8、4、0 **GIFx**：チャネル x グローバル割り込みフラグ（x = 1 ～ 8）

このビットは、ハードウェアによってセットされます。BDMA_IFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x に TE、HT、または TC イベントは発生していません。

1：チャネル x に TE、HT、または TC イベントが発生しました。

16.4.2 DMA 割り込みフラグクリアレジスタ（BDMA_IFCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTEIF8	CHTIF8	CTCIF8	CGIF8	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31、27、23、19、15、11、7、3 **CTEIFx**：チャネル x 転送エラークリア（x = 1 ～ 8）
このビットはソフトウェアでセットされます。
0：影響なし。
1：BDMA_ISR レジスタの対応する TEIF フラグをクリアします。

ビット 30、26、22、18、14、10、6、2 **CHTIFx**：チャネル x 1/2 転送クリア（x = 1 ～ 8）
このビットはソフトウェアでセットされます。
0：影響なし。
1：BDMA_ISR レジスタの対応する HTIF フラグをクリアします。

ビット 29、25、21、17、13、9、5、1 **CTCIFx**：チャネル x 転送完了クリア（x = 1 ～ 8）
このビットはソフトウェアでセットされます。
0：影響なし。
1：BDMA_ISR レジスタの対応する TCIF フラグをクリアします。

ビット 28、24、20、16、12、8、4、0 **CGIFx**：チャネル x グローバル割り込みクリア（x = 1 ～ 8）
このビットはソフトウェアでセットされます。
0：影響なし。
1：BDMA_ISR レジスタの GIF、TEIF、HTIF、および TCIF フラグをクリアします。

16.4.3 DMA チャネル x 設定レジスタ（BDMA_CCRx）（x = 1 ～ 8、ここで x = チャネル数）

アドレスオフセット：0x08 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **MEM2MEM**：メモリ間モード

このビットは、ソフトウェアによってセット／クリアされます。

0：メモリ間モードは無効です。
1：メモリ間モードは有効です。

ビット 13:12 **PL[1:0]**：チャネル優先順位レベル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：低
01：中
10：高
11：最優先

ビット 11:10 **MSIZE[1:0]**：メモリサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

ビット 9:8 **PSIZE[1:0]**：ペリフェラルサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

ビット 7 **MINC**：メモリインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：メモリインクリメントモードは無効です。
1：メモリインクリメントモードは有効です。

ビット 6 **PINC**：ペリフェラルインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：ペリフェラルインクリメントモードは無効です。
1：ペリフェラルインクリメントモードは有効です。

ビット 5 **CIRC**：サーキュラモード
このビットは、ソフトウェアによってセット／クリアされます。
0：サーキュラモードは無効です。
1：サーキュラモードは有効です。

ビット 4 **DIR**：データ転送方向
このビットは、ソフトウェアによってセット／クリアされます。
0：ペリフェラルから読み出します。
1：メモリから読み出します。

ビット 3 **TEIE**：転送エラー割り込み有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TE 割り込みは無効です。
1：TE 割り込みは有効です。

ビット 2 **HTIE**：1/2 転送割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：HT 割り込みは無効です。
1：HT 割り込みは有効です。

ビット 1 **TCIE**：転送完了割り込み有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TC 割り込みは無効です。
1：TC 割り込みは有効です。

ビット 0 **EN**：チャネルイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：チャネルは無効です。
1：チャネルは有効です。

16.4.4 DMA チャネル x データ数レジスタ（BDMA_CNDTRx）（x = 1〜8、x = チャネル番号）

アドレスオフセット：0x0C + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送データ項目の数

転送データ数：0〜65535このレジスタは、チャネルが無効な場合にのみ書き込みできます。チャネルが有効になると、このレジスタは読み出し専用になり、送信されるべき残りのバイトを指し示します。このレジスタは、各 DMA 転送後にデクリメントされます。

転送が完了すると、このレジスタは、ゼロのままにすることもできれば、チャネルがサーキュラモードモードに設定されている場合には、事前にプログラムされた値を自動的に再ロードすることもできます。

このレジスタの値が 0 の場合、チャネルが有効であるか否かにかかわらず、トランザクションは処理できません。

16.4.5 DMA チャネル x ペリフェラルアドレスレジスタ（BDMA_CPARx）（x = 1 〜 8、x = チャネル番号）

アドレスオフセット：0x10 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

チャネルが有効なときには、このレジスタへは書き込みを *行わないで* ください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PA[31:0]**：ペリフェラルアドレス

データの読み出し／書き込みが行われるペリフェラルデータレジスタのベースアドレス。

PSIZE が 01（16 ビット）のとき、PA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

PSIZE が 10（32ビット）のとき、PA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

16.4.6 DMA チャネル x メモリアドレスレジスタ（BDMA_CMARx）（x = 1 ～ 8、x = チャネル番号）

アドレスオフセット：0x14 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

チャネルが有効なときには、このレジスタへは書き込みを *行わない* でください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

データの読み出し／書き込みが行われるメモリ領域のベースアドレス。

MSIZE が 01（16 ビット）のとき、MA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

MSIZE が 10（32ビット）のとき、MA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

16.4.7 BDMA レジスタマップ

次の表に、DMA レジスタマップとリセット値を示します。

表 108. BDMA レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	BDMA_ISR	TEIF8	HTIF8	TCIF8	GIF8	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	BDMA_IFCR	CTEIF8	CHTIF8	CTCIF8	CGIF8	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	BDMA_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	BDMA_CNDTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	BDMA_CPAR1	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	BDMA_CMAR1	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x1C	BDMA_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	BDMA_CNDTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	BDMA_CPAR2	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	BDMA_CMAR2	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x30	BDMA_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	BDMA_CNDTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	BDMA_CPAR3	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	BDMA_CMAR3	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 108. BDMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x40	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44	**BDMA_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**BDMA_CNDTR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	**BDMA_CPAR4**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**BDMA_CMAR4**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x58	**BDMA_CCR5**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	**BDMA_CNDTR5**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	**BDMA_CPAR5**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	**BDMA_CMAR5**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x6C	**BDMA_CCR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**BDMA_CNDTR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	**BDMA_CPAR6**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	**BDMA_CMAR6**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	**BDMA_CCR7**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**BDMA_CNDTR7**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 108. BDMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	**BDMA_CPAR7**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**BDMA_CMAR7**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x94	**BDMA_CCR8**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x98	**BDMA_CNDTR8**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x9C	**BDMA_CPAR8**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA0	**BDMA_CMAR8**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

17 DMA リクエストマルチプレクサ（DMAMUX）

17.1 概要

ペリフェラルは DMA リクエスト信号の設定による DMA 転送のリクエストを示します。DMA リクエストは、DMA 確認応答信号を生成する DMA コントローラで処理されるまで保留され、対応する DMA リクエスト信号はデアサートされます。

本書では、DMA リクエスト／確認応答プロトコルに必要な一連の制御信号を明示的に示したり説明したりしていませんが、これは DMA リクエストラインとして参照します。

DMAMUX リクエストマルチプレクサにより、製品のペリフェラルと DMA コントローラの間で DMA リクエストラインをルーティングすることができます。ルーティング機能は、プログラム可能なマルチチャネル DMA リクエストラインマルチプレクサにより確保されます。各チャネルは、DMAMUX 同期入力のイベントで無条件に、または同期的に一意の DMA リクエストラインを選択します。DMAMUX は、入力トリガ信号のプログラム可能なイベントからの DMA リクエストジェネレータとしても使用できます。

DMAMUX インスタンスの数およびそれらの主な特性は、*セクション 17.3.1* で指定されています。

ペリフェラルからの DMA リクエストラインおよび DMAMUX リクエストジェネレータ出力への DMAMUX リクエストマルチプレクサ入力の割り当て、DMA コントローラチャネルへの DMAMUX リクエストマルチプレクサ出力の割り当て、および内部および外部信号への DMAMUX 同期およびトリガ入力の割り当ては、製品の実装に依存しています。詳細については、*セクション 17.6*を参照してください。

17.2 DMAMUX の主な機能

- 最大 16 チャネルのプログラム可能な DMA リクエストラインマルチプレクサ出力
- 最大 8 チャネルの DMA リクエストジェネレータ
- 最大 32 のDMA リクエストジェネレータへのトリガ入力
- 最大 16 の同期入力
- DMA リクエストジェネレータチャネル：
  - DMA リクエストトリガ入力セレクタ
  - DMA リクエストカウンタ
  - 選択された DMA リクエストトリガ入力のイベントオーバーランフラグ
- DMA リクエストラインマルチプレクサチャネル出力：
  - 最大 107 のペリフェラルからのDMA リクエストラインの入力
  - 1 つの DMA リクエストライン出力
  - 同期入力セレクタ
  - DMA リクエストカウンタ
  - 選択された同期入力のイベントオーバーランフラグ
  - DMA リクエスト連鎖のための 1 つのイベント出力

17.3 DMAMUX の実装

17.3.1 DMAMUX1 および DMAMUX2 のインスタンス化

この製品では、DMA リクエストマルチプレクサの 2 つのインスタンスを内蔵しています。

- DMA1 および DMA2 の DMAMUX1 （D2 ドメイン）
- BDMA の DMAMUX2 （D3 ドメイン）

DMAMUX1 および DMAMUX2 は、次の表に記載されているハードウェア設定パラメータでインスタンス化されています。

表 109. DMAMUX1 および DMAMUX2 のインスタンス化

機能	DMAMUX1	DMAMUX2
DMAMUX 出力リクエストチャネルの数	16	8
DMAMUX リクエストジェネレータチャネルの数	8	8
DMAMUX リクエストトリガ入力の数	8	32
DMAMUX 同期入力の数	8	16
DMAMUX ペリフェラルリクエスト入力の数	107	12

17.3.2 DMAMUX1 の配置

DMAMUX1 へのリソースの配置は配線接続されています。

DMAMUX1 は、D2 ドメインの DMA1 および DMA2 で使用されます。

- DMAMUX1 のチャネル 0 から 7 は、DMA1 のチャネル 0 から 7 に接続されます。
- DMAMUX1 のチャネル 8 から 15 は、DMA2 のチャネル 0 から 7 に接続されます。

表 110. DMAMUX1：リソースへのマルチプレクサ入力の割り当て

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
1	dmamux1_req_gen0	40	SPI2_TX	79	UART7_RX
2	dmamux1_req_gen1	41	USART1_RX	80	UART7_TX
3	dmamux1_req_gen2	42	USART1_TX	81	UART8_RX
4	dmamux1_req_gen3	43	USART2_RX	82	UART8_TX
5	dmamux1_req_gen4	44	USART2_TX	83	SPI4_RX
6	dmamux1_req_gen5	45	USART3_RX	84	SPI4_TX
7	dmamux1_req_gen6	46	USART3_TX	85	SPI5_RX
8	dmamux1_req_gen7	47	TIM8_CH1	86	SPI5_TX
9	ADC1	48	TIM8_CH2	87	SAI1_A
10	ADC2	49	TIM8_CH3	88	SAI1_B
11	TIM1_CH1	50	TIM8_CH4	89	SAI2_A
12	TIM1_CH2	51	TIM8_UP	90	SAI2_B
13	TIM1_CH3	52	TIM8_TRIG	91	SWPMI_RX

表 110. DMAMUX1：リソースへのマルチプレクサ入力の割り当て（続き）

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
14	TIM1_CH4	53	TIM8_COM	92	SWPMI_TX
15	TIM1_UP	54	予約済み	93	SPDIFRX_DT
16	TIM1_TRIG	55	TIM5_CH1	94	SPDIFRX_CS
17	TIM1_COM	56	TIM5_CH2	95	HR_REQ(1)
18	TIM2_CH1	57	TIM5_CH3	96	HR_REQ(2)
19	TIM2_CH2	58	TIM5_CH4	97	HR_REQ(3)
20	TIM2_CH3	59	TIM5_UP	98	HR_REQ(4)
21	TIM2_CH4	60	TIM5_TRIG	99	HR_REQ(5)
22	TIM2_UP	61	SPI3_RX	100	HR_REQ(6)
23	TIM3_CH1	62	SPI3_TX	101	dfsdm1_dma0
24	TIM3_CH2	63	UART4_RX	102	dfsdm1_dma1
25	TIM3_CH3	64	UART4_TX	103	dfsdm1_dma2
26	TIM3_CH4	65	USART5_RX	104	dfsdm1_dma3
27	TIM3_UP	66	UART5_TX	105	TIM15_CH1
28	TIM3_TRIG	67	DAC1	106	TIM15_UP
29	TIM4_CH1	68	DAC2	107	TIM15_TRIG
30	TIM4_CH2	69	TIM6_UP	108	TIM15_COM
31	TIM4_CH3	70	TIM7_UP	109	TIM16_CH1
32	TIM4_UP	71	USART6_RX	110	TIM16_UP
33	I2C1_RX	72	USART6_TX	111	TIM17_CH1
34	I2C1_TX	73	I2C3_RX	112	TIM17_UP
35	I2C2_RX	74	I2C3_TX	113	SAI3_A
36	I2C2_TX	75	DCMI	114	SAI3_B
37	SPI1_RX	76	CRYP_IN	115	ADC3
38	SPI1_TX	77	CRYP_OUT	-	-
39	SPI2_RX	78	HASH_IN	-	-

表 111. DMAMUX1：リソースへのトリガ入力の割り当て

トリガ入力	リソース	トリガ入力	リソース
0	dmamux1_evt0	4	LPTIMER2_out
1	dmamux1_evt1	5	LPTIMER3_out
2	dmamux1_evt2	6	extit0
3	LPTIMER1_out	7	TIM12_TRGO

表 112. DMAMUX1：リソースへの同期入力の割り当て

同期入力	リソース	同期入力	リソース
0	dmamux1_evt0	4	LPTIMER2_out
1	dmamux1_evt1	5	LPTIMER3_out
2	dmamux1_evt2	6	extit0
3	LPTIMER1_out	7	TIM12_TRGO

17.3.3 DMAMUX2 の配置

DMAMUX2 のチャネル 0 から 7 は、BDMA のチャネル 0 から 7 に接続されます。

表 113. DMAMUX2：リソースへのマルチプレクサ入力の割り当て

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
1	dmamux2_req_gen0	11	SPI6_RX
2	dmamux2_req_gen1	12	SPI6_TX
3	dmamux2_req_gen2	13	I2C4_RX
4	dmamux2_req_gen3	14	I2C4_TX
5	dmamux2_req_gen4	15	SAI4_A
6	dmamux2_req_gen5	16	SAI4_B
7	dmamux2_req_gen6	17	ADC3_REQ
8	dmamux2_req_gen7	18	予約済み
9	LP UART1_RX	19	予約済み
10	LP UART1_TX	20	予約済み

表 114. DMAMUX2：リソースへのトリガ入力の割り当て

トリガ入力	リソース	トリガ入力	リソース
0	dmamux2_evt0	16	Spi6_it_async
1	dmamux2_evt1	17	Comp1_out
2	dmamux2_evt2	18	Comp2_out
3	dmamux2_evt3	19	RTC_wkup
4	dmamux2_evt4	20	Syscfg_exti0_mux
5	dmamux2_evt5	21	Syscfg_exti2_mux
6	dmamux2_evt6	22	I2c4_it_event
7	Lpuart1_it_R_WUP_ASYNC	23	Spi6_it
8	Lpuart1_it_T_WUP_ASYNC	24	Lpuart1_it_T
9	Lptim2_ait	25	Lpuart1_it_R
10	Lptim2_out	26	ADC3_it
11	Lptim3_ait	27	ADC3_AWD1_out
12	Lptim3_out	28	DMA1_D3_ch0_it
13	Lptim4_ait	29	DMA1_D3_ch1_it

表 114. DMAMUX2：リソースへのトリガ入力の割り当て（続き）

トリガ入力	リソース	トリガ入力	リソース
14	Lptim5_ait	30	予約済み
15	I2c4_it_async	31	予約済み

表 115. DMAMUX2：リソースへの同期入力の割り当て

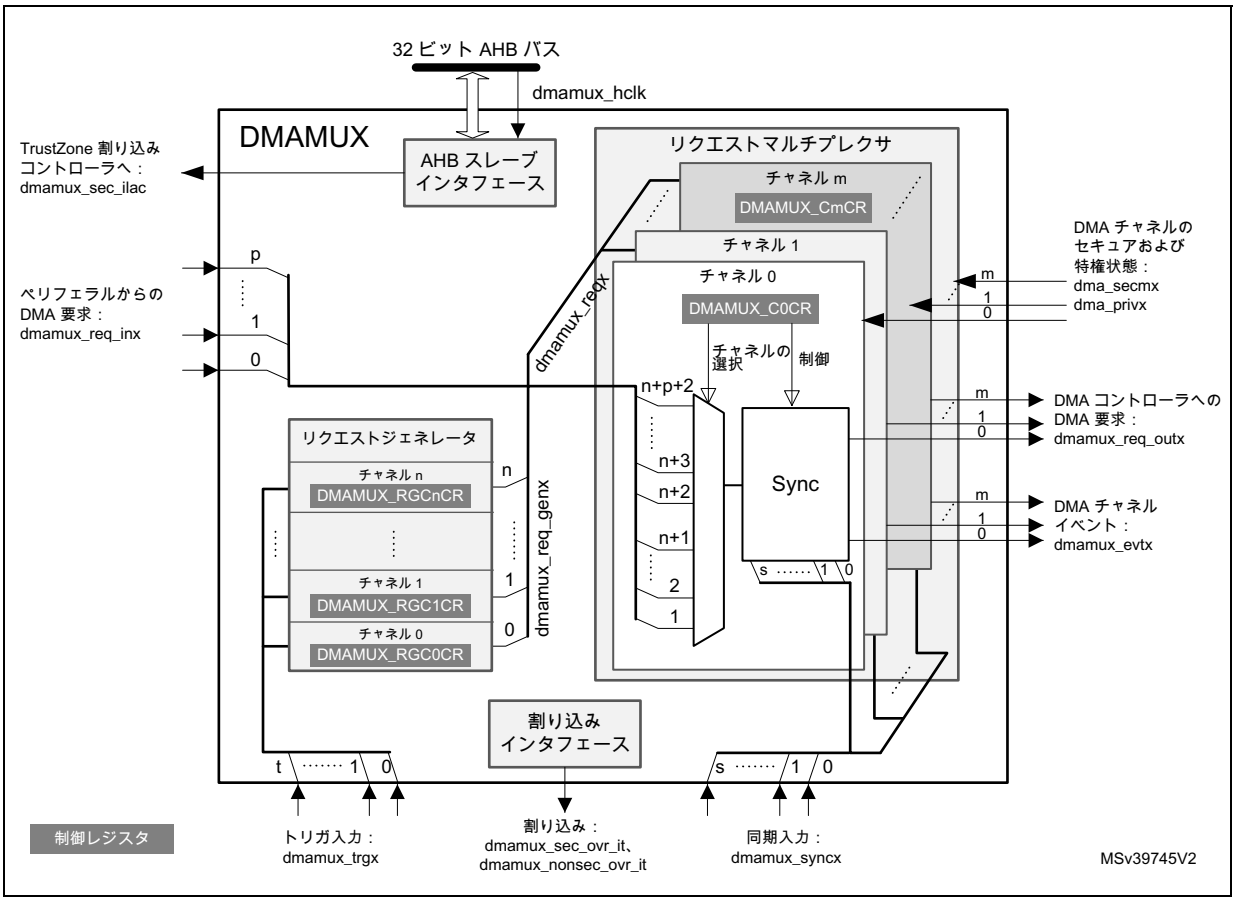
同期入力	リソース	同期入力	リソース
0	dmamux2_evt0	8	Lptim2_out
1	dmamux2_evt1	9	Lptim3_out
2	dmamux2_evt2	10	I2c4_it_async
3	dmamux2_evt3	11	Spi6_it_async
4	dmamux2_evt4	12	Comp1_out
5	dmamux2_evt5	13	RTC_wkup
6	Lpuart1_it_R_WUP_ASYNC	14	Syscfg_exti0_mux
7	Lpuart1_it_T_WUP_ASYNC	15	Syscfg_exti2_mux

17.4 DMAMUX の機能詳細

17.4.1 DMAMUX ブロック図

図 75 に DMAMUX ブロック図を示します。

図 75. DMAMUX ブロック図



DMAMUX には、リクエストラインマルチプレクサとリクエストラインジェネレータの 2 つの主なサブブロックがあります。

実装時の割り当ては次のとおりです。

- ペリフェラル（dmamux_req_inx）および DMAMUX リクエストジェネレータサブブロックのチャネル（dmamux_req_genx）からの DMAMUX リクエストマルチプレクササブブロック入力（dmamux_reqx）
- DMA コントローラのチャネルへの DMAMUX リクエスト出力（dmamux_req_outx）
- DMA リクエストトリガ入力への内部または外部信号（dmamux_trgx）
- 同期入力への内部または外部信号（dmamux_syncx）

17.4.2 DMAMUX 信号

表 116 に DMAMUX 信号をリストします。

表 116. DMAMUX 信号

信号名	説明
dmamux_hclk	DMAMUX AHB クロック
dmamux_req_inx	DMAMUX DMA リクエストライン入力（ペリフェラルから）
dmamux_trgx	DMAMUX DMA リクエストトリガ入力（リクエストジェネレータサブブロックへ）
dmamux_req_genx	DMAMUX リクエストジェネレータサブブロックチャネル出力
dmamux_reqx	DMAMUX リクエストマルチプレクササブブロック入力（ペリフェラルリクエストおよびリクエストジェネレータチャネルから）
dmamux_syncx	DMAMUX 同期入力（リクエストマルチプレクササブブロックへ）
dmamux_req_outx	DMAMUX リクエスト出力（DMA コントローラへ）
dmamux_evtx	DMAMUX イベント出力
dmamux_ovr_it	DMAMUX オーバーラン割り込み

17.4.3 DMAMUX チャネル

DMAMUX チャネルは、リクエストマルチプレクサで選択された入力に応じて、追加の DMAMUX リクエストジェネレータチャネルを含む、DMAMUX リクエストマルチプレクサチャネルです。

DMAMUX リクエストマルチプレクサチャネルは、DMA コントローラの単一チャネルへの接続専用です。

チャネル設定手順

DMAMUX の x チャネルと関連する DMA チャネル y の両方を設定するには、以下のシーケンスに従う必要があります。

1. DMA チャネル y をセットし、チャネル y の有効化を除くすべての設定を行います。
2. 関連する DMAMUX y チャネルをセットし、すべての設定を行います。
3. 最後に、DMA y チャネルレジスタの EN ビットをセットして DMA チャネル y を有効にします。

17.4.4 DMAMUX リクエストラインマルチプレクサ

DMAMUX リクエストマルチプレクサには複数のチャネルがあるため、DMA リクエストラインと呼ばれる DMA リクエスト／確認応答制御信号の実際のルーティングが確保されます。

各DMA リクエストラインは、DMAMUX リクエストラインマルチプレクサのすべてのチャネルにそれぞれ並列に接続されます。

DMA リクエストは、ペリフェラルまたは DMAMUX リクエストジェネレータのどちらかから供給されます。

DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの 8 ビット DMAREQ_ID フィールドの設定どおり、DMA リクエストライン数を選択します。

注： ***DMAREQ_ID フィールドが null 値の場合、いずれの DMA リクエストラインも選択されていません。***

同じ非 null の DMA_REQ_ID 値を、異なるDMAMUX リクエストマルチプレクサチャネルx および y（DMAMUX_CxCR および DMAMUX CyCR 経由）にプログラムすることはできません。

同じ非 null の DMAREQ_ID を、DMAMUX リクエストラインマルチプレクサの 2 つの異なるチャネルに設定することはできません。

DMA リクエスト選択の最上部では、必要に応じて同期モードおよび／またはイベント生成を設定および有効化できます。

同期モードおよびチャネルイベント生成

各 DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの同期有効化（SE）ビットを設定することで、個別に同期することができます。

DMAMUX には複数の同期入力があります。同期入力は、リクエストマルチプレクサのすべてのチャネルに並列に接続されます。

同期入力は、特定のチャネル x の DMAMUX_CxCR レジスタの 5 ビット SYNC_ID フィールドを介して選択されます。

チャネルが同期モードの場合、DMAMUX_CxCR レジスタの SPOL[1:0] フィールドを介して、選択された入力同期信号でプログラム可能な立ち上がり／立ち下がりエッジを検出すると、選択された DMA リクエストラインの入力はマルチプレクサチャネル出力に伝播されます。

さらに、内部的に DMAMUX リクエストマルチプレクサにはプログラム可能な DMA リクエストカウンタもあります。これは、チャネルリクエスト出力生成あるいはイベント生成にも使用されます。チャネル x 出力でのイベント生成は、DMAMUX_CxCR レジスタの EGE ビット（イベント生成イネーブル）を使用して有効化されます。

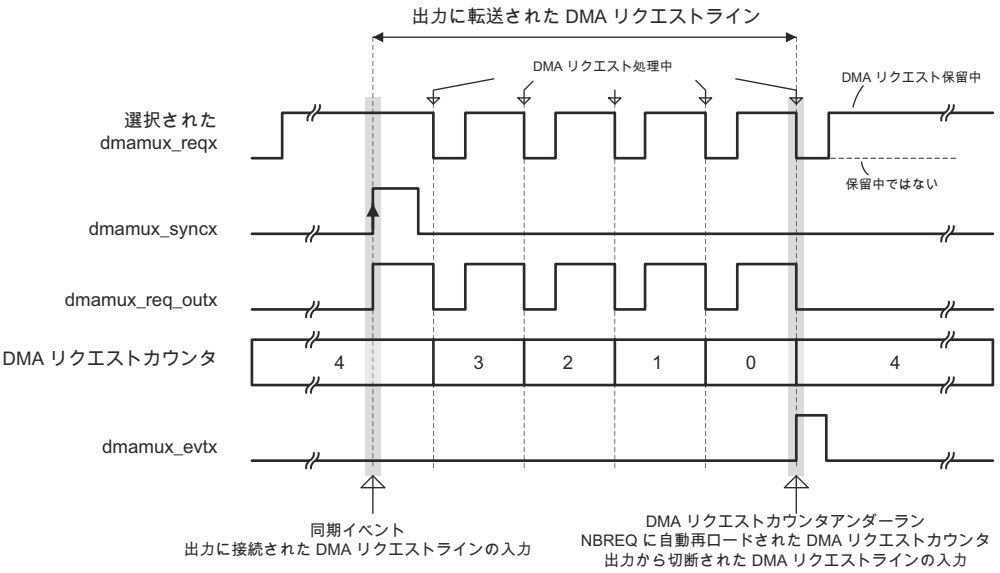
次の 2 つの図に示すように、同期入力の検出されたエッジで、選択された DMA リクエストラインの入力は DMAMUX マルチプレクサチャネル x 出力に接続されています。この点以降、選択された DMAMUX リクエストライン上の処理済みの各 DMA リクエスト（たとえば、リクエスト信号がデアサートされた場合）によって、DMA リクエストカウンタはデクリメントされます。アンダーランの際、DMA リクエストカウンタには DMAMUX_CxCR レジスタの NBREQ フィールドの値が自動的にロードされ、DMA リクエストラインの入力はマルチプレクサチャネル x 出力から切断されます。

このため、検出された同期イベントに続いてマルチプレクサチャネル x 出力に転送される DMA リクエストの数は、NBREQ フィールドの値に 1 を加えた数と等しくなります。

注： ***NBREQ フィールド値は、対応するマルチプレクサチャネル x の SE（同期イネーブル）および EGE（イベント生成イネーブル）ビットの両方が無効である場合にのみ、ソフトウェアによって書き込むことができます。***

EGE が有効である場合、DMA リクエストカウンタにプログラムされた NBREQ フィールドの値が自動的に再ロードされると、マルチプレクサチャネルは 1 つの AHB クロックサイクルのパルスとしてチャネルイベントを生成します。

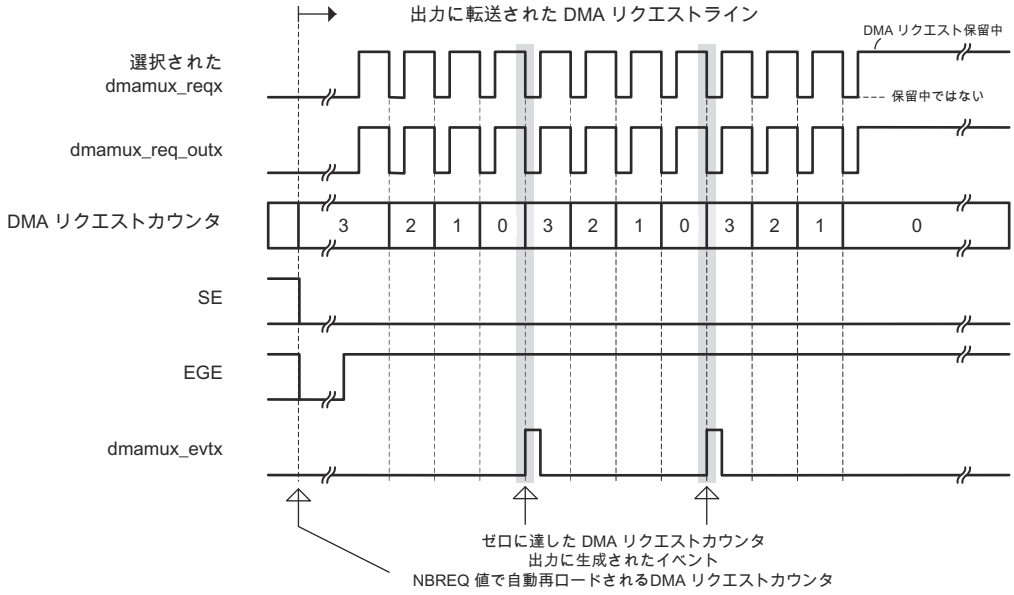
図 76. DMAMUX リクエストラインマルチプレクサチャネルの同期モード



例：次の条件で設定された DMAMUX_CCRxNBREQ=4、SE=1、EGE=1、SPOL=01（立ち上がりエッジ）

MSv41974V1

図 77. DMA リクエストラインマルチプレクサチャネルのイベント生成



例：次の条件で設定された DMAMUX_CCRxNBREQ=3、SE=0、EGE=1

MSv41975V1

注： ***同期イベント（エッジ）は、エッジ後の状態が 2 AHB クロックサイクルよりも長く安定している場合に検出されます。***

DMAMUX_CxCR レジスタに書き込む際、同期イベントは 3 AHB クロックサイクルの間、マスクされます。

同期オーバーランおよび割り込み

マルチプレクサチャネル x のプログラムされた NBREQ フィールド値よりも DMA リクエストカウンタの値が低い間、新しい同期イベントが発生すると、DMAMUX_CSR ステータスレジスタの同期オーバーランフラグビット SOFx がセットされます。

注： ***リクエストマルチプレクサチャネル x 同期は、DMA コントローラに関連するチャネルの使用完了時に無効化されます（DMAMUX_CxCR.SE=0）。そうでない場合、新しく同期イベントを検出したときに、DMA コントローラから受信する DMA 確認応答の欠落（未処理のリクエストとも呼ばれる）によって同期オーバーランが発生します。***

オーバーランフラグ SOFx は、DMAMUX_CFR レジスタの関連するクリア同期オーバーランフラグビット CSOFx をセットすることでリセットされます。

DMAMUX_CxCR レジスタの同期オーバーラン割り込みイネーブルビット SOIE がセットされている場合、同期オーバーランフラグをセットすると、割り込みが生成されます。

17.4.5 DMAMUX リクエストジェネレータ

DMAMUX リクエストジェネレータでは、DMA リクエストトリガ入力のトリガイベントに続いて、DMA リクエストを生成します。

DMAMUX リクエストジェネレータには複数のチャネルがあります。DMA リクエストトリガ入力は、すべてのチャネルに並列に接続されます。

DMAMUX リクエストジェネレータチャネルの出力は、DMAMUX リクエストラインマルチプレクサへの入力です。

各 DMAMUX リクエストジェネレータチャネル x には、対応する DMAMUX_RGxCR レジスタにイネーブルビット GE （ジェネレータイネーブル）があります。

DMAMUX リクエストジェネレータチャネル x の DMA リクエストトリガ入力は、対応する DMAMUX_RGxCR レジスタの SIG_ID （トリガ信号 ID）フィールドによって選択されます。

DMA リクエストトリガ入力のトリガイベントは立ち上がりエッジと立ち下がりエッジのいずれかです。アクティブなエッジは対応する DMAMUX_RGxCR レジスタの GPOL（ジェネレータ極性）フィールドによって選択されます。

トリガイベント時、対応するジェネレータチャネルでは出力時に DMA リクエストの生成を開始します。処理された各 DMA リクエストは（リクエスト信号がデアサートされた場合）、DMAMUX リクエストジェネレータに対して内部的に内蔵 DMA リクエストカウンタをデクリメントします。アンダーランの際、DMA リクエストカウンタには対応する DMAMUX_RGxCR レジスタの GNBREQ フィールドの値が自動的にロードされ、リクエストジェネレータチャネルは DMA リクエストの生成を停止します。

このため、トリガイベント後に生成される DMA リクエストの数は GNBREQ+1 です。

DMA リクエストカウンタは、対応するチャネルが無効である限りGNBREQ フィールド値に保持されます（つまり、DMAMUX_RGxCR.GE ビットはローになります）。

注： ***GNBREQ フィールド値は、対応するジェネレータチャネル x のイネーブル GE ビットが無効である場合にのみ、ソフトウェアによって書き込むことができます。***

トリガイベント（エッジ）は、エッジ後の状態が 2 AHB クロックサイクルよりも長く安定している場合に検出されます。

DMAMUX_RGxCR レジスタに書き込む際、トリガイベントは 3 AHB クロックサイクルの間、マスクされます。

トリガオーバーランおよび割り込み

新しい DMA リクエストトリガイベントが発生すると、DMAMUX リクエストジェネレータカウンタが対応するリクエストジェネレータチャネル x のプログラムされた GNBREQ フィールド値よりも低い場合、またリクエストジェネレータチャネル x が GE によって有効化されている場合、リクエストトリガイベントオーバーランフラグビット OFx は、ハードウェアによってステータス DMAMUX_RGSR レジスタにアサートされます。

注： ***リクエストジェネレータチャネル x は、DMA コントローラの関連するチャネルの使用完了時に無効化されます（DMAMUX_RGxCR.GE=0）。そうでない場合、新しくトリガイベントを検出したときに、DMA から受信する確認応答の欠落（未処理のリクエストとも呼ばれる）によってトリガオーバーランが発生します。***

オーバーランフラグ OFx は、DMAMUX_RGCFR レジスタの関連するクリアオーバーランフラグビット COFx をセットすることでリセットされます。

DMAMUX_RGxCR レジスタの DMA リクエストトリガイベントオーバーラン割り込みイネーブルビット OIE がセットされている場合、DMAMUX リクエストトリガオーバーランフラグによって、割り込みが生成されます。

17.5 DMAMUX 割り込み

次の場合に割り込みを生成することができます。

- 各 DMA リクエストラインマルチプレクサチャネルの同期イベントオーバーラン
- 各 DMA リクエストジェネレータチャネルのトリガイベントオーバーラン

それぞれのケースで、チャネルごとの個別の割り込みイネーブル、ステータス、およびクリアフラグレジスタビットを使用できます。

表 117. DMAMUX 割り込み

割り込み信号	割り込みイベント	イベントフラグ	クリアビット	イネーブルビット
dmamuxovr_it	DMAMUX リクエストラインマルチプレクサのチャネル x での同期イベントオーバーラン	SOFx	CSOFx	SOIE
	DMAMUX リクエストジェネレータのチャネル x でのトリガイベントオーバーラン	OFx	COFx	OIE

17.6 DMAMUX レジスタ

DMAMUX1 および DMAMUX2 ベースアドレスのレジスタ境界アドレスを含む表を参照してください。

DMAMUX レジスタには、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）でアクセスできます。アドレスはデータサイズに合わせて整列します。

17.6.1 DMAMUX1 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX1_CxCR）

アドレスオフセット：0x04 * x （x = 0 から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE
			rw					rw					rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
						rw	rw	rw							

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SYNC_ID[4:0]**：同期識別

同期入力を選択します（*表 112：DMAMUX1：リソースへの同期入力の割り当て*を参照）。

ビット 23:19 **NBREQ[4:0]**：転送する DMA リクエストの数から 1 を引いた数

同期イベントの後で DMA コントローラに転送する DMA リクエストの数、および／または出力イベントが生成される前の DMA リクエストの数を定義します。

このフィールドは、SE と EGE ビットの両方がローの場合にのみ書き込まれます。

ビット 18:17 **SPOL[1:0]**：同期極性

選択された同期入力のエッジ極性を定義します。

00：イベントなし、つまり同期または検出なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **SE**：同期有効化

0：同期は無効です。

1：同期は有効です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EGE**：イベント生成イネーブル

0：イベント生成は無効です。

1：イベント生成は有効です。

ビット 8 **SOIE**：同期オーバーラン割り込みイネーブル

0：割り込みは無効です。

1：割り込みは有効です。

ビット 7:0 **DMAREQ_ID[7:0]**：DMA リクエストID

DMA リクエストの入力を選択します。リソースへのマルチプレクサ入力の割り当てについては、DMAMUX の表を参照してください。

17.6.2 DMAMUX2 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX2_CxCR）

アドレスオフセット：0x04 * x、ここで x = 0 から 7

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE
			rw					rw					rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
						rw	rw	rw							

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SYNC_ID[4:0]**：同期識別

同期入力を選択します（リソースへの同期入力の割り当てについては、DMAMUX の表を参照）。

ビット 23:19 **NBREQ[4:0]**：転送する DMA リクエストの数から 1 を引いた数

同期イベントの後で DMA コントローラに転送する DMA リクエストの数、および／または出力イベントが生成される前の DMA リクエストの数を定義します。

このフィールドは、SE と EGE ビットの両方がローの場合にのみ書き込まれます。

ビット 18:17 **SPOL[1:0]**：同期極性

選択された同期入力のエッジ極性を定義します。

00：イベントなし。つまり同期または検出なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **SE**：同期有効化

0：同期は無効です。

1：同期は有効です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EGE**：イベント生成イネーブル

0：イベント生成は無効です。

1：イベント生成は有効です。

ビット 8 **SOIE**：同期オーバーラン割り込みイネーブル

0：割り込みは無効です。

1：割り込みは有効です。

ビット 7:0 **DMAREQ_ID[7:0]**：DMA リクエストID

DMA リクエストの入力を選択します。リソースへのマルチプレクサ入力の割り当てについては、DMAMUX の表を参照してください。

17.6.3 DMAMUX1 リクエストラインマルチプレクサ割り込みチャネルステータスレジスタ（DMAMUX1_CSR）

アドレスオフセット：0x080

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SOF[15:0]**：同期オーバーランイベントフラグ

このフラグは、DMA リクエストカウンタの値が NBREQ よりも低い場合に、DMA リクエストラインマルチプレクサチャネル x で同期イベントが発生したときにセットされます。

このフラグは、対応する DMAMUX_CFR レジスタの CSOFx ビットに 1 を書き込むことでクリアされます。DMAMUX2_CFR のビット 15:8 は予約済みです。

17.6.4 DMAMUX2 リクエストラインマルチプレクサ割り込みチャネルステータスレジスタ（DMAMUX2_CSR）

アドレスオフセット：0x080

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな読み出しにより、ビットレベルでアクセスします。このとき、接続された DMA コントローラチャネル y のセキュア制御ビットに応じて、対象の DMAMUX2 リクエストラインマルチプレクサチャネル x のセキュアモードに従い、また DMAMUX2 の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX2 の配置の実装セクションを参照）。

このレジスタには、非特権読み出しまたは特権読み出しにより、ビットレベルでアクセスします。このとき、接続された DMA コントローラチャネル y の特権制御ビットに応じて、対象の DMAMUX2 リクエストラインマルチプレクサチャネル x の特権モードに従い、また DMAMUX2 の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX2 の配置の実装セクションを参照）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SOF[7:0]**：同期オーバーランイベントフラグ

このフラグは、DMA リクエストカウンタの値が NBREQ よりも低い場合に、DMA リクエストラインマルチプレクサチャネル x で同期イベントが発生したときにセットされます。

このフラグは、対応する DMAMUX2_CFR レジスタの CSOFx ビットに 1 を書き込むことでクリアされます。

17.6.5 DMAMUX1 リクエストラインマルチプレクサ割り込みクリアフラグレジスタ（DMAMUX1_CFR）

アドレスオフセット：0x084

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSOF15	CSOF14	CSOF13	CSOF12	CSOF11	CSOF10	CSOF9	CSOF8	CSOF7	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CSOF[15:0]**：同期オーバーランイベントフラグをクリアします。

各ビットに 1 を書き込むと DMAMUX_CSR レジスタの対応するオーバーランフラグ SOFx をクリアします。

17.6.6 DMAMUX2 リクエストラインマルチプレクサ割り込みクリアフラグレジスタ（DMAMUX2_CFR）

アドレスオフセット：0x084

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな書き込みにより、ビットレベルで書き込みます。このとき、接続された DMA コントローラチャネル y のセキュア制御ビットに応じて、対象の DMAMUX リクエストラインマルチプレクサチャネル x のセキュアモードに従い、また DMAMUX の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX の配置の実装セクションを参照）。

このレジスタには、非特権書き込みまたは特権書き込みにより、ビットレベルで書き込みます。このとき、接続された DMA コントローラチャネル y の特権制御ビットに応じて、対象の DMAMUX リクエストラインマルチプレクサチャネル x の特権モードに従い、また DMAMUX の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX の配置の実装セクションを参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSOF6	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CSOF[7:0]**：同期オーバーランイベントフラグをクリアします。

各ビットに 1 を書き込むと DMAMUX2_CSR レジスタの対応するオーバーランフラグ SOFx をクリアします。

17.6.7 DMAMUX1 リクエストジェネレータチャネル x 設定レジスタ（DMAMUX1_RGxCR）

アドレスオフセット：0x100 + 0x04 * (x-0)（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE
								rw					rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
							rw				rw				

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 **GNBREQ[4:0]**：生成される DMA リクエストの数（1 を引く）

トリガイベント後に生成される DMA リクエストの数を定義します。実際に生成される DMA リクエストの数は GNBREQ+1 です。

注： このフィールドは、GE ビットが無効のときのみ書き込むことができます。

ビット 18:17 **GPOL[1:0]**：DMA リクエストジェネレータトリガ極性

選択されたトリガ入力のエッジ極性を定義します。
00：イベントなし。つまりトリガの検出または生成なし
01：立ち上がりエッジ
10：立ち下がりエッジ
11：立ち上がり／立ち下がりエッジ

ビット 16 **GE**：DMA リクエストジェネレータチャネル x イネーブル
0：DMA リクエストジェネレータチャネル x は無効です。
1：DMA リクエストジェネレータチャネル x は有効です。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OIE**：トリガオーバーラン割り込みイネーブル
0：トリガオーバーランイベント発生時の割り込みは無効です。
1：トリガオーバーランイベント発生時の割り込みは有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SIG_ID[4:0]**：信号ID

DMA リクエストジェネレータのチャネル x で使用する DMA リクエストトリガ入力を選択します。

17.6.8 DMAMUX2 リクエストジェネレータチャネル x 設定レジスタ（DMAMUX2_RGxCR）

アドレスオフセット：0x100 + 0x04 * (x-0)（x = 0 から 7）

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな書き込みによって書き込みを行います。このとき、このレジスタに割り当てられた対象の DMAMUX リクエストラインマルチプレクサチャネル y のセキュアモードに従い、また DMAMUX リクエストジェネレータの x チャネル出力が DMAMUX リクエストラインチャネルの y チャネルによって選択されていることを考慮します（DMAMUX2_CyCR.DMAREQ_ID[7:0] および DMAMUX の配置の実装セクションを参照）。

このレジスタには、非特権書き込みまたは特権書き込みによって書き込みを行います。このとき、このレジスタに割り当てられた対象の DMAMUX リクエストラインマルチプレクサチャネル y の特権モードに従い、また DMAMUX リクエストジェネレータの x チャネル出力が DMAMUX リクエストラインチャネルの y チャネルによって選択されていることを考慮します（DMAMUX2_CyCR.DMAREQ_ID[7:0] および DMAMUX の配置の実装セクションを参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE
								rw					rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
							rw				rw				

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 **GNBREQ[4:0]**：生成される DMA リクエストの数（1 を引く）

トリガイベント後に生成される DMA リクエストの数を定義します。実際に生成される DMA リクエストの数は GNBREQ+1 です。

注： このフィールドは、GE ビットが無効のときのみ書き込むことができます。

ビット 18:17 **GPOL[1:0]**：DMA リクエストジェネレータトリガ極性

選択されたトリガ入力のエッジ極性を定義します。

00：イベントなし。つまりトリガの検出または生成なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **GE**：DMA リクエストジェネレータチャネル x イネーブル

0：DMA リクエストジェネレータチャネル x は無効です。

1：DMA リクエストジェネレータチャネル x は有効です。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OIE**：トリガオーバーラン割り込みイネーブル

0：トリガオーバーランイベント発生時の割り込みは無効です。

1：トリガオーバーランイベント発生時の割り込みは有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SIG_ID[4:0]**：信号識別

DMA リクエストジェネレータのチャネル x で使用する DMA リクエストトリガ入力を選択します。

17.6.9 DMAMUX1 リクエストジェネレータ割り込みステータスレジスタ（DMAMUX1_RGSR）

アドレスオフセット：0x140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **OF[7:0]**：トリガオーバーランイベントフラグ

このフラグは、DMA リクエストジェネレータカウンタの値が GNBREQ よりも低い場合に、DMA リクエストジェネレータチャネル x でトリガイベントが発生したときにセットされます。

このフラグは、DMAMUX_RGCFR レジスタの対応する COFx ビットに 1 を書き込むことでクリアされます。

17.6.10 DMAMUX2 リクエストジェネレータ割り込みステータスレジスタ（DMAMUX2_RGSR）

アドレスオフセット：0x140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **OF[7:0]**：トリガオーバーランイベントフラグ

このフラグは、DMA リクエストジェネレータカウンタの値が GNBREQ よりも低い場合に、DMA リクエストジェネレータチャネル x でトリガイベントが発生したときにセットされます。

このフラグは、DMAMUX2_RGCFR レジスタの対応する COFx ビットに 1 を書き込むことでクリアされます。

17.6.11 DMAMUX1 リクエストジェネレータ割り込みクリアフラグレジスタ（DMAMUX1_RGCFR）

アドレスオフセット：0x144

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **COF[7:0]**：トリガオーバーランイベントフラグのクリア

各ビットに 1 を書き込むと DMAMUX_RGSR レジスタの対応するオーバーランフラグ OFx をクリアします。

17.6.12 DMAMUX2 リクエストジェネレータ割り込みクリアフラグレジスタ（DMAMUX2_RGCFR）

アドレスオフセット：0x144

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **COF[7:0]**：トリガオーバーランイベントフラグのクリア

各ビットに 1 を書き込むと DMAMUX2_RGSR レジスタの対応するオーバーランフラグ OFx をクリアします。

17.6.13 DMAMUX レジスタマップ

次の表に、DMAMUX レジスタとリセット値の一覧を示します。DMAMUX レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 118. DMAMUX レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMAMUX_C0CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x004	DMAMUX_C1CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x008	DMAMUX_C2CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x00C	DMAMUX_C3CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x010	DMAMUX_C4CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x014	DMAMUX_C5CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x018	DMAMUX_C6CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x01C	DMAMUX_C7CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x020(1)	DMAMUX_C8CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x024 (1)	DMAMUX_C9CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x028 (1)	DMAMUX_C10CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x02C (1)	DMAMUX_C11CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x030 (1)	DMAMUX_C12CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x034 (1)	DMAMUX_C13CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x038 (1)	DMAMUX_C14CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x03C (1)	DMAMUX_C15CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	DMAREQ_ID[7:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0	0	0	0	0	0	0	0	0
0x040 - 0x07C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

参考資料

表 118. DMAMUX レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080	DMAMUX_CSR(2)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	DMAMUX_CFR (2)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSOF15	CSOF14	CSOF13	CSOF12	CSOF11	CSOF10	CSOF9	CSOF8	CSOF7	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x088 - 0x0FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	DMAMUX_RG0CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL	GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x104	DMAMUX_RG1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL	GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x108	DMAMUX_RG2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL	GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x10C	DMAMUX_RG3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL	GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x140	DMAMUX_RGSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF15	OF14	OF13	OF12	OF11	OF10	OF9	OF8	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x144	DMAMUX_RGCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF15	COF14	COF13	COF12	COF11	COF10	COF9	COF8	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x140	DMAMUX_RGSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
	リセット値																									0	0	0	0	0	0	0	0
0x144	DMAMUX_RGCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
	リセット値																									0	0	0	0	0	0	0	0
0x148 - 0x3FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

1. DMAMUX1 にのみ適用されます。DMAMUX2 のワードは予約済みです。
2. DMAMUX2 のビット 15:8 は予約済みです。

参考資料

18 Chrom-Art Accelerator™ コントローラ（DMA2D）

18.1 DMA2D の概要

Chrom-Art Accelerator™（DMA2D）はイメージ操作専用の特殊な DMA です。以下の操作を実行できます。

- 転送先イメージの一部または全部を特定の色で塗りつぶす
- 転送元イメージの一部または全部を、転送先イメージの一部または全部にコピーする
- 転送元イメージの一部または全部を、ピクセルフォーマット変換を行って転送先イメージの一部または全部にコピーする
- 異なるピクセルフォーマットを持つ転送元イメージの全部または一部をブレンドし、その結果を別のカラーフォーマットで転送先イメージの一部または全体にコピーする

JPEG デコーダ出力を処理するためのブロックベースの YCbCr を含む、すべての標準的なピクセルフォーマットがサポートされており、4bit/pixel から 32bit/pixel のインデックスカラーモードまたはダイレクトカラーモードに対応しています。DMA2D は CLUT（color look-uptables：カラールックアップテーブル）用に専用メモリを搭載しています。

18.2 DMA2D の主な機能

DMA2D の主な機能は次のとおりです。

- シングル AXI マスタバスアーキテクチャ。
- 8/16/32 ビットアクセスをサポートする AHB スレーブプログラミングインタフェース（32 ビットの CLUT アクセスを除く）。
- ユーザが任意に設定可能なワーク領域サイズ
- ユーザが任意に設定可能な転送元および転送先領域のオフセット
- 全メモリ空間でユーザが任意に設定可能な転送元および転送先のアドレス
- 最大 2 つの転送元のブレンド操作
- アルファ値を調整可能（ソース値、固定値、または変調値）
- ユーザが任意に設定可能な転送元および転送先のカラーフォーマット
- インダイレクトカラーまたはダイレクトカラーで 4bit/pixel から 32bit/pixel まで、最大 11 のカラーフォーマットをサポート。
- 4:4:4、4:2:2、および 4:2:0 のクロマサブサンプリング係数のブロックベース（8x8）の YCbCr をサポート
- インダイレクトカラーモードにおける CLUT 格納用に 2 つの内部メモリを搭載
- 自動 CLUT ローディングまたは CPU を使用した CLUT プログラミング
- ユーザが任意にプログラム可能な CLUT サイズ
- AXI 帯域幅を制御する内部タイマ
- 4 つの動作モード : レジスタからメモリへの転送、メモリ間転送、ピクセルフォーマット変換を伴うメモリ間転送､およびピクセルフォーマット変換およびブレンディングを伴うメモリ間転送
- 固定カラーでの矩形領域塗りつぶし
- 領域間のコピー
- 転送元イメージと転送先イメージ間のピクセルフォーマット変換を伴うコピー
- カラーフォーマットの異なる 2 つの転送元のコピー、およびブレンディング
- DMA2D 動作のアボートおよびサスペンド
- ユーザが任意に設定可能な転送先ラインにおけるウォーターマーク割り込み
- バスエラーまたはアクセス競合時の割り込み生成
- プロセス完了時の割り込み生成

18.3 DMA2D の機能詳細

18.3.1 概要

DMA2D コントローラはダイレクトメモリ転送を行います。AXI マスタとして、AXI バスマトリックスを制御して AXI トランザクションを開始することができます。

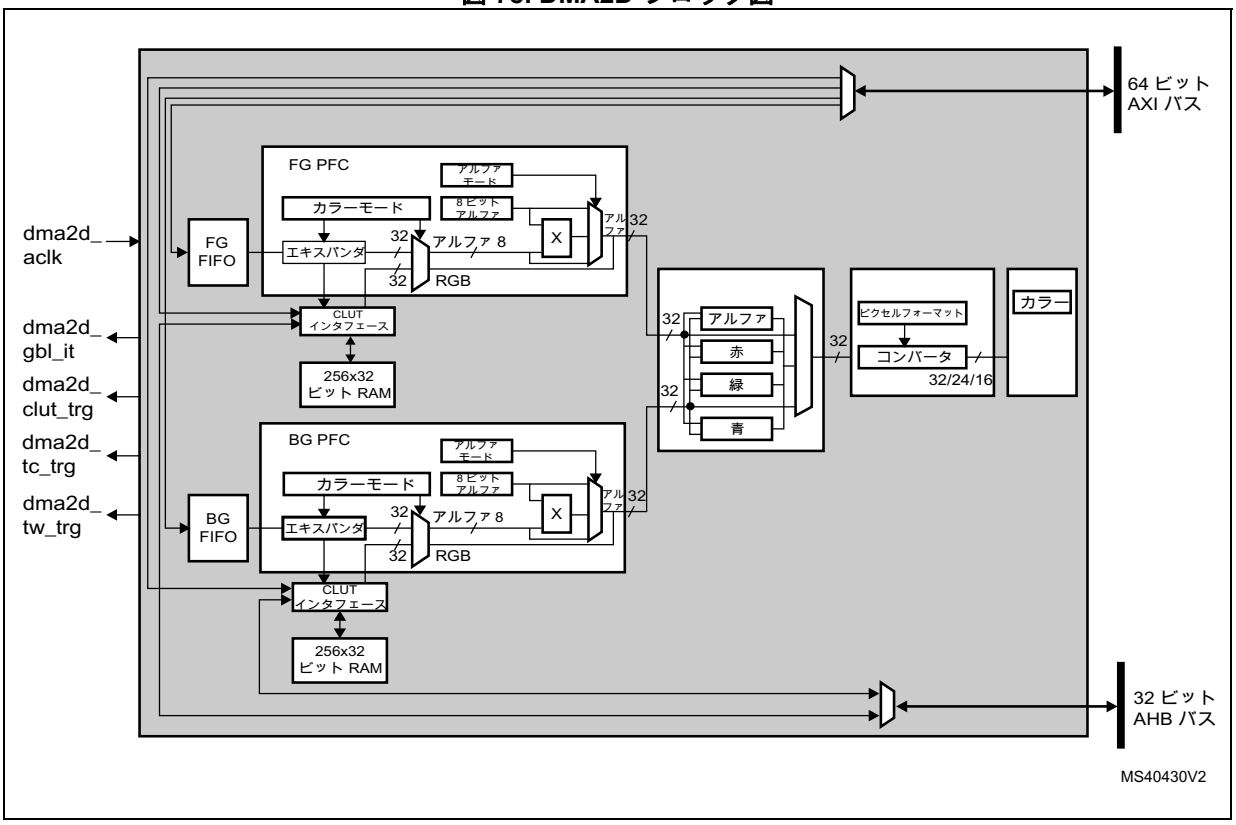
DMA2D は、次のモードで動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- ピクセルフォーマット変換を伴うメモリ間転送
- ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送

AHB スレーブポートは、DMA2D コントローラをプログラムするために使用されます。

DMA2D インタフェースのブロック図を*図 78 : DMA2D ブロック図*に示します。

図 78. DMA2D ブロック図



18.4 DMA2D ピンおよび内部信号

*表 119*に、DMA2D 内部信号のリストを示します。

表 119. DMA2D 内部入力／出力信号

信号名	信号タイプ	説明
dma2d_aclk	デジタル入力	32 ビット AXI バスクロック
dma2d_gbl_it	デジタル出力	DMA2D グローバル割り込み
dma2d_clut_trg	デジタル出力	CLUT 転送完了（MDMA へ）
dma2d_tc_trg	デジタル出力	転送完了（MDMA へ）
dma2d_tw_trg	デジタル出力	転送ウォーターマーク（MDMA へ）

18.4.1 DMA2D 制御

DMA2D コントローラは、DMA2D 制御レジスタ（DMA2D_CR）を使用して設定します。このレジスタによって次の選択を行うことができます。

ユーザアプリケーションで次の操作を実行できます。

- 動作モードの選択
- DMA2D 割り込みの有効化／無効化
- 進行中のデータ転送の開始／サスペンド／アボート

18.4.2 DMA2D フォアグラウンドおよびバックグラウンド FIFO

DMA2D フォアグラウンド（FG）FIFO およびバックグラウンド（BG）FIFO はコピーなどの処理をする入力データをフェッチします。

これらの FIFO は、それぞれのピクセルフォーマットコンバータ（PFC）に定義されているカラーフォーマットに従ってピクセルデータをフェッチします。

これらは次の制御レジスタにより設定可能です。

- DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）
- DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR）
- DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）
- DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリへの転送モードで動作する場合、どちらの FIFO もアクティブになりません。

DMA2D がメモリ間転送モードで動作する（ピクセルフォーマット変換、ブレンディング処理のいずれも行わない）場合は、FG FIFO のみアクティブになりバッファとして機能します。

DMA2D がピクセルフォーマット変換を伴うメモリ間転送モードで動作する（ブレンディング処理を行わない）場合は、BG FIFO はアクティブになりません。

18.4.3 DMA2D フォアグラウンドおよびバックグラウンドのピクセルフォーマットコンバータ（PFC）

DMA2D フォアグラウンドピクセルフォーマットコンバータ（PFC）およびバックグラウンドピクセルフォーマットコンバータはピクセルフォーマット変換を行って 32bit/pixel のデータを生成します。PFC はアルファ値に調整を加えることもできます。

コンバータの一段目ではカラーフォーマットを変換します。フォアグラウンドピクセルとバックグラウンドピクセルの元のカラーフォーマットは、それぞれ DMA2D_FGPFCCR および DMA2D_BGPFCCR の CM[3:0] ビットによって設定します。

サポートされている入力フォーマットを*表 120：入力でサポートされているカラーモード*に示します。

表 120. 入力でサポートされているカラーモード

CM[3:0]	カラーモード
0000	ARGB8888
0001	RGB888
0010	RGB565

表 120. 入力でサポートされているカラーモード

CM[3:0]	カラーモード
0011	ARGB1555
0100	ARGB4444
0101	L8
0110	AL44
0111	AL88
1000	L4
1001	A8
1010	A4
1011	YCbCr （フォアグラウンドの場合のみ）

カラーフォーマットは次のとおりコード化されています。

- アルファ値フィールド：透明度
  0xFF 値は不透明ピクセルに対応し 0x00 は透明ピクセルに対応します。
- R フィールドは赤
- G フィールドは緑
- B フィールドは青
- L フィールド：輝度
  このフィールドは RGB から 3 つの、ARGB から 4 つの成分を読み出すための CLUT のインデックスです。

元のフォーマットがダイレクトカラーモードの場合、MSB を LSB へコピーしてチャネルあたり 8 ビットへの拡張が行われます。これにより変換の線形性が完全に保証されます。

元のフォーマットにアルファ値が無い場合、アルファ値は自動的に 0xFF（不透明）にセットされます。

元のフォーマットがインダイレクトカラーモードの場合、CLUT が必要となり、各ピクセルフォーマットコンバータは、256 エントリの 32 ビット CLUT と連動します。

特定のアルファモード A4 および A8 では、カラー情報の格納、インデックスの利用のいずれも行われません。イメージ生成に使用されるカラーは固定となっており、フォアグラウンドピクセル用のカラーは DMA2D_FGCOLR レジスタで、バックグラウンドピクセル用のカラーは DMA2D_BGCOLR レジスタで定義されます。

システムメモリ内のフィールドの順序は*表 121：メモリ内のデータ順序*で定義しています。

表 121. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]
L8	L_3[7:0]	L_2[7:0]	L_1[7:0]	L_0[7:0]
AL44	A_3[3:0]L_3[3:0]	A_2[3:0]L_2[3:0]	A_1[3:0]L_1[3:0]	A_0[3:0]L_0[3:0]
AL88	A_1[7:0]	L_1[7:0]	A_0[7:0]	L_0[7:0]
L4	L_7[3:0]L_6[3:0]	L_5[3:0]L_4[3:0]	L_3[3:0]L_2[3:0]	L_1[3:0]L_0[3:0]
A8	A_3[7:0]	A_2[7:0]	A_1[7:0]	A_0[7:0]
A4	A_7[3:0]A_6[3:0]	A_5[3:0]A_4[3:0]	A_3[3:0]A_2[3:0]	A_1[3:0]A_0[3:0]

32 ビットに調整された 24 ビット RGB888 は ARGB8888 モードでサポートされます。

32 ビット値が生成されると、*表 122：アルファモード設定*に示すように DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AM[1:0] フィールドに従ってアルファチャネルを調整することができます。

この時アルファチャネルは以下が可能です。

- そのままを維持（調整なし）
- DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値で置換
- もしくは、DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値を 255 で除算した後、元のアルファ値に乗算した値で置換

表 122. アルファモード設定

AM[1:0]	アルファモード
00	調整なし
01	DMA2D_xxPFCCR の値で置換
10	DMA2D_xxPFCCR を 255 で除算した値を元の値に乗算して置換
11	予約済み

注： ***別のフォーマットをサポートするために、DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AI ビットをセットして、着信アルファ値を反転できます。これは、DMA2D_FGPFCCR/DMA2D_BGPFCCR および CLUT に格納されるアルファ値にも適用されます。***

DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドを入れ替えることもできます。これは、CLUT および DMA2D_FGCOLR/DMA2D_BGCOLR レジスタで使用される RGB 順序にも適用されます。

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

CLUT メモリアクセスおよび CLUT の自動ローディングは CLUT インタフェースによって管理されます。

次の 3 種類のアクセスが可能です。

- ピクセルフォーマット変換操作中の PFC による CLUT の読み出し
- CPU が CLUT に対してデータの読み出しまたは書き込みを行っているときの AHB スレーブポートを経由した CLUT へのアクセス
- CLUT の自動ローディングが行われているときの AXI マスタポートを経由した CLUT への書き込み

CLUT メモリローディングは次の 2 つの方法で行うことができます。

- オートローディング

  CLUT のローディングは、次の順序で行う必要があります。

  a) DMA2D_FGCMAR レジスタ（フォアグラウンド CLUT）または DMA2D_BGCMAR レジスタ（バックグラウンド CLUT）に CLUT アドレスを設定します。

  b) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CS[7:0] フィールドに CLUT サイズを設定します。

  c) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の START ビットをセットして転送を開始します。このオートローディング処理中、CLUT に対する CPU によるアクセスはできません。競合が発生した場合、CLUT アクセスエラー割り込みが発生します。ただし、DMA2D_CR 内の CAEIE ビットが“1”に設定されている必要があります。

- マニュアルローディング

  ローカル CLUT メモリがマップされている DMA2D AHB スレーブポートを使用して、アプリケーションがマニュアルで CLUT を設定する必要があります。フォアグラウンド CLUT はアドレスオフセット 0x0400 に配置され、バックグラウンド CLUT はアドレスオフセット 0x0800 に配置されます。

CLUT フォーマットは、24 ビットまたは 32 ビットが可能です。*表 123：サポートされている CLUT カラーモード*に示すように DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CCM ビットを介して CLUT フォーマットを設定します。

表 123. サポートされている CLUT カラーモード

CCM	CLUT カラーモード
0	32 ビット ARGB8888
1	24 ビット RGB888

システムメモリ内の CLUT データの構成方法は*表 124：メモリ内の CLUT データ順序*に規定しています。

表 124. メモリ内の CLUT データ順序

CLUT カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]

18.4.5 DMA2D ブレンダ

DMA2D ブレンダは、転送元ピクセルをペアでブレンドし、出来上がりのピクセルを計算します。

ブレンドは次の式に従って行われます。

$$\text{次の場合：}\alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}} \quad \text{ここで C = R または G または B}$$

除算では、最も近い整数に切り捨てられます。

ブレンダは設定レジスタを必要としません。ブレンダーの使用方法は DMA2D_CR レジスタの MODE[1:0] フィールドで定義される DMA2D 動作モードによって異なります。

18.4.6 DMA2D 出力 PFC

出力 PFC は、32 ビットから DMA2D 出力ピクセルフォーマット変換設定レジスタ（DMA2D_OPFCCR）の CM[2:0] フィールドで定義された出力フォーマットへのピクセルフォーマット変換を行います。

サポートされている出力フォーマットを*表 125：出力でサポートされているカラーモード*に示します。

表 125. 出力でサポートされているカラーモード

CM[2:0]	カラーモード
000	ARGB8888
001	RGB888
010	RGB565
011	ARGB1555
100	ARGB4444

注： ***別のフォーマットをサポートするために、DMA2D_OPFCCR レジスタの AI ビットをセットして、計算されたアルファ値を反転できます。これは、DMA2D_OCOLR で使用されるアルファ値にも適用されます。***

DMA2D_OPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドをスワップすることもできます。これは、DMA2D_OCOLR で使用される RGB 順序にも適用されます。

18.4.7 DMA2D 出力 FIFO

出力 FIFO は、出力 PFC に定義されているカラーフォーマットに従ってピクセルをプログラムします。

転送先領域は、次の制御レジスタによって定義します。

- DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）
- DMA2D 出力オフセットレジスタ（DMA2D_OOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリモードで動作する場合、設定された出力矩形領域が、32 ビット、24 ビットまたは 16 ビットの固定値が格納された DMA2D 出力カラーレジスタ（DMA2D_OCOLR）で指定された色で塗りつぶされます。フォーマットは、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで選択します。

データは*表 126：メモリ内のデータ順序*で定義されている順番でメモリに格納されます。

表 126. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]

32 ビットに調整された RGB888 は ARGB8888 モードでサポートされます。

18.4.8 DMA2D AXI マスタポートタイマ

AXI マスタポートには 8 ビットのタイマが組み込まれており、クロスバーの帯域幅を任意に制限することができます。

このタイマは AXI クロックによって駆動され、2 つの連続アクセス間のデッドタイムをカウントします。これによって帯域幅の使用を制限することができます。

タイマの有効化とデッドタイム値は AXI マスタポートタイマ設定レジスタ（DMA2D_AMPTCR）によって設定します。

18.4.9 DMA2D トランザクション

DMA2D トランザクションは、一定数のデータ転送のシーケンスで構成されています。データの数と幅はソフトウェアによりプログラムすることができます。

各 DMA2D データ転送は、最大 4 つのステップで構成されています。

1. DMA2D_FGMAR レジスタで示されるメモリ位置からのデータローディングおよび DMA2D_FGCR で定義されるピクセルフォーマット変換。
2. DMA2D_BGMAR レジスタで示されるメモリ位置からのデータローディングおよび DMA2D_BGCR で定義されるピクセルフォーマット変換。
3. アルファ値に関する PFC 操作の結果として生じるアルファチャネルに従ったすべての読み出しピクセルのブレンディング。
4. DMA2D_OCR レジスタに従って出来たピクセルのピクセルフォーマット変換および DMA2D_OMAR レジスタによってアドレス指定されたメモリ位置へのデータのプログラミング。

18.4.10 DMA2D 設定

データ転送元およびデータ転送先は両方ともアドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4 G バイトメモリ領域内のペリフェラルおよびメモリを対象にすることができます。

DMA2D は、DMA2D_CR レジスタの MODE[1:0] ビットによって次の 4 つのモードのいずれでも動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- PFC を伴うメモリ間転送
- PFC とブレンディングを伴うメモリ間転送

レジスタからメモリへの転送

“レジスタからメモリ”モードは、ユーザ定義領域を事前定義された色で塗りつぶすために使用します。

カラーフォーマットは DMA2D_OPFCCR で設定します。

DMA2D はどのソースからもデータのフェッチを行いません。DMA2D_OCOLR レジスタで定義された色を、DMA2D_OMAR で示されるアドレスの、DMA2D_NLR および DMA2D_OOR で定義される領域に書き込むだけです。

メモリ間転送

メモリ間モードでは、DMA2D はグラフィカルなデータの変換は何も行いません。フォアグラウンド入力 FIFO はバッファの役目を果たし、データは、DMA2D_FGMAR で定義される転送元のメモリ位置から DMA2D_OMAR が示す転送先のメモリ位置へ転送されます。

DMA2D_FGPFCCR レジスタの CM[3:0] ビットでプログラムされているカラーモードによって、入力および出力の両方のピクセルあたりのビット数を定義します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、転送先に対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

PFC を伴うメモリ間転送

このモードでは DMA2D は転送元データのピクセルフォーマット変換を行いそれを転送先メモリ位置に格納します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、転送先に対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

データは DMA2D_FGMAR レジスタで定義されている位置からフェッチされ、フォアグラウンドPFC によって処理されます。元のピクセルフォーマットは DMA2D_FGPFCCR レジスタで設定します。

元のピクセルフォーマットがダイレクトカラーモードである場合、カラーチャネルはすべて 8 ビットに拡張されます。

ピクセルフォーマットがインダイレクトカラーモードである場合、関連する CLUT を CLUT メモリにロードする必要があります。

CLUT のローディングは以下の流れに従って自動的に行うことができます。

1. CLUT アドレスを DMA2D_FGCMAR に設定します。
2. DMA2D_FGPFCCR レジスタの CS[7:0] ビットに CLUT サイズを設定します。
3. DMA2D_FGPFCCR レジスタの CCM ビットに CLUT フォーマット（24 ビットまたは 32 ビット）を設定します。
4. DMA2D_FGPFCCR レジスタの START ビットをセットすることにより CLUT ローディングを開始します。

CLUT のローディングが完了すると、DMA2D_IFR レジスタの CTCIF フラグが立ち、DMA2D_CR レジスタの CTCIE ビットがセットされていれば割り込みが生成されます。自動 CLUT ローディング処理は標準的な DMA2D 転送と並行して動作することはできません。

CLUT は CPU から書き込むこともできれば、AHB ポート経由で他のマスタによって書き込むこともできます。DMA2D 転送が進行中でかつ CLUT を使用しているとき（インダイレクトカラーフォーマット）、CLUT にはアクセスできません。

カラー変換処理と並行して、DMA2D_FGPFCCR レジスタにプログラムされている値によっては、アルファ値を追加または変更することができます。元のイメージにアルファチャネルが無い場合、デフォルトのアルファ値 0xFF が自動的に追加されて完全に不透明なピクセルになります。アルファ値は、DMA2D_FGPFCCR レジスタの AM[1:0] ビットに従って修正することができます。

- 変更しない設定ができます。
- DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値で定義された値で置き換えることができます。
- 元の値に DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値を掛けて 255 で割った値で置き換えることができます。

結果の 32 ビットデータは OUT PFC によって、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで指定されたフォーマットにエンコードされます。CLUT 生成処理がサポートされていないため、出力ピクセルフォーマットをインダイレクトモードにすることはできません。

処理されたデータは DMA2D_OMAR が指す転送先メモリ位置に書き込まれます。

PFC とブレンディングを伴うメモリ間転送

このモードでは、DMA2D_FGMAR および DMA2D_BGMAR で定義されるメモリ位置からフォアグラウンド FIFO およびバックグラウンド FIFO に 2 つの転送元データがフェッチされます。

メモリ間モードで説明しているように 2 つのピクセルフォーマットコンバータを、設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自の CLUT メモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれの PFC によって 32 ビットに変換された後、それらは次の式に従ってブレンドされます。

$$\text{次の場合}: \alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}} \quad \text{ここで C = R または G または B}$$

除算では、小数点以下は切り捨てられます。

結果の 32 ビットピクセル値は出力 PFC により指定出力フォーマットに従ってエンコードされ、データは DMA2D_OMAR が指す転送先のメモリ位置に書き込まれます。

設定エラー検出

DMA2D は転送を行う前に設定が正しいことを確認します。新規の転送／自動ローディングが開始されるとき、誤った設定が検出されると、ハードウェアによって設定エラー割り込みフラグがセットされます。DMA2D_CR レジスタの CEIE ビットがセットされている場合、割り込みが生成されます。

検出することができる誤った設定は次のとおりです。

- フォアグラウンド CLUT 自動ローディング：DMA2D_FGCMAR の MA ビットが DMA2D_FGPFCCR の CCM ビットと整合が取れていない
- バックグラウンド CLUT 自動ローディング：DMA2D_BGCMAR の MA ビットが DMA2D_BGPFCCR の CCM ビットと整合が取れていない
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGMAR の MA ビットが DMA2D_FGPFCCR の CM ビットと整合が取れていない
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが無効
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが A4 または L4 のとき DMA2D_FGOR の LO ビットが奇数
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGMAR の MA ビットがDMA2D_BGPFCCR の CM ビットと整合が取れていない
- メモリ転送：DMA2D_BGPFCCR の CM ビットが無効（ブレンディングモードのみ）
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数です。
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGPFCCR の CM ビットが A4 または L4のとき DMA2D_BGOR の LO ビットが奇数
- メモリ転送（メモリ間モードを除く）：DMA2D_OMAR の MA ビットが DMA2D_OPFCCR のCM ビットと整合が取れていない
- メモリ転送（メモリ間モードを除く）：DMA2D_OPFCCR の CM ビットが無効
- メモリ転送：DMA2D_NLR の NL ビット = 0
- メモリ転送：DMA2D_NLR の PL ビット = 0
- YCbCr フォーマット：DMA2D_FGPFCCR の START ビットをセットすることにより CLUT ローディングが開始された場合
- YCbCr フォーマット：メモリ間モードが選択されている場合
- YCbCr フォーマット：YCbCR4:4:4 が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が 8 ピクセルの倍数でない場合
- YCbCr フォーマット：YCbCr4:2:2 または YCbCr4:2:0 が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が 16 ピクセルの倍数でない場合

18.4.11 YCbCr サポート

DMA2D フォアグラウンド面では、8x8 のブロックベースの YCbCr を、異なるクロマサブサンプリング係数を持つ JPEG デコーダからの出力のようにサポートできます。

メモリ構成は標準 JFIF ルールに従います。

- 3 つの色のコンポーネントをそれぞれ 8 ビットでコード化する必要があります。
- 各コンポーネントは、MCU と呼ばれる 8x8（64 バイト）のブロックで整列する必要があります。

クロマサブサンプリング係数によっては、*表 127：メモリ内の MCU 順序* を参照して MCU をメモリに配置する必要があります。

表 127. メモリ内の MCU 順序

サブ サンプリング	@	@ + 64	@ + 128	@+192	@+256	@ + 320
4:4:4	Y_1	Cb_1	Cr_1	Y_2	Cb_2	Cr_2
4:2:2	Y_1	Y_2	Cb_{12}	Cr_{12}	Y_3	Y_4
4:2:0	Y_1	Y_2	Y_3	Y_4	Cb_{1234}	Cr_{1234}

クロマサブサンプリング係数は、DMA2D_FGPFCCR レジスタの CSS フィールドを介して設定されます。

YCbCr カラーモードで設定されたフォアグラウンドで DMA2D が開始されると、最初の 2 つのクロマ MCU がフォアグラウンドの CLUT にロードされます。クロマ MCU がロードされると、従来のカラーモードと同様に、DMA2D が Y MCU のローディングを実行します。

18.4.12 DMA2D 転送制御（開始、サスペンド、アボート、完了）

DMA2D が設定されると、DMA2D_CR レジスタの START ビットをセットすることにより転送を開始できます。転送が完了すると、START ビットが自動的にリセットされ、DMA2D_ISR レジスタの TCIF フラグが立ちます。DMA2D_CR レジスタの TCIE ビットがセットされている場合は、割り込みを生成できます。

DMA2D_CR レジスタの SUSP ビットをセットすることにより、ユーザアプリケーションからいつでも DMA2D をサスペンドすることができます。その後、DMA2D_CR レジスタの ABORT ビットをセットすることにより、トランザクションをアボートすることができます。また、DMA2D_CR レジスタの SUSP ビットをリセットすることによりリスタートすることができます。

DMA2D_CR レジスタの ABORT ビットをセットすることにより、ユーザアプリケーションからいつでも進行中のトランザクションをアボートすることができます。この場合、TCIF フラグは立ちません。

DMA2D_FGPFCCR および DMA2D_BGPFCCR レジスタのそれぞれの START ビットを使用することによって、自動 CLUT 転送もアボートしたりサスペンドしたりすることができます。

18.4.13 ウォーターマーク

所定のラインの最後のピクセルが転送先のメモリ領域に書き込まれたときに、割り込みを生成するようにウォーターマークをプログラムすることができます。

ライン番号は、DMA2D_LWR レジスタの LW[15:0] フィールドで定義します。

ラインの最後のピクセルが転送されると、DMA2D_CR レジスタの TWIE ビットがセットされていれば、DMA2D_ISR レジスタの TWIF フラグが立ち、割り込みが生成されます。

18.4.14 エラー管理

次の 2 種類のエラーをトリガすることができます。

- DMA2D_ISR レジスタの TEIF フラグによって通知される AXI マスタポートエラー。
- CLUT アクセス（CLUT ローディング中または DMA2D 転送中に CPU が CLUT にアクセスしようとする）によって引き起こされる競合。DMA2D_ISR レジスタの CAEIF フラグによって通知されます。

両方のフラグはそれぞれ DMA2D_CR レジスタの割り込みイネーブルフラグに関連付けられており、必要に応じて割り込みを生成します（TEIE と CAEIE）。

18.4.15 AXI デッドタイム

AXI 帯域幅の使用を制限するために、2 つの連続した AXI アクセス間のデッドタイムをプログラムすることができます。

この機能は、DMA2D_AMTCR レジスタの EN ビットをセットすることによって有効にできます。

デッドタイム値は、DMA2D_AMTCR レジスタの DT[7:0] フィールドに格納します。この値は AXI バス上の 2 つの連続するトランザクション間の最小保証サイクル数を表します。

DMA2D が実行中のデッドタイム値の更新は、次の AXI 転送に反映されます。

18.5 DMA2D 割り込み

割り込みは次のイベントによって生成できます。

- 設定エラー
- CLUT 転送完了
- CLUT アクセスエラー
- 転送ウォーターマークに到達
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 128. DMA2D 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
設定エラー	CEIF	CEIE
CLUT 転送完了	CTCIF	CTCIE
CLUT アクセスエラー	CAEIF	CAEIE
転送ウォーターマーク	TWF	TWIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

18.6 DMA2D レジスタ

18.6.1 DMA2D 制御レジスタ（DMA2D_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	Res.	Res.	Res.	Res.	ABORT	SUSP	START
		rw	rw	rw	rw	rw	rw						rs	rw	rs

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **MODE**：DMA2D モード

このビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

00：メモリ間転送（FG フェッチのみ）

01：PFCを伴うメモリ間転送（FG PFC アクティブで FG フェッチのみ）

10：ブレンディングを伴うメモリ間転送（PFC およびブレンディングを伴う FG フェッチおよび BG フェッチ）

11：レジスタからメモリ転送（FG および BG なし、出力ステージのみアクティブ）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CEIE**：設定エラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CE 割り込みは無効です。

1：CE 割り込みは有効です。

ビット 12 **CTCIE**：CLUT 転送完了割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CTC 割り込みは無効です。

1：CTC 割り込みは有効です。

ビット 11 **CAEIE**：CLUT アクセスエラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CAE 割り込みは無効です。

1：CAE 割り込みは有効です。

ビット 10 **TWIE**：転送ウォーターマーク割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：TW 割り込みは無効です。

1：TW 割り込みは有効です。

ビット 9 **TCIE**：転送完了割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割り込みは無効です。

1：TC 割り込みは有効です。

ビット 8 **TEIE**：転送エラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割り込みは無効です。

1：TE 割り込みは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **ABORT**：アボート

このビットを使用して現在の転送を中止できます。このビットはソフトウェアによってセットされ、START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送アボートは要求されていません。

1：転送アボートが要求されています。

ビット 1 **SUSP**：サスペンド

このビットを使用して現在の転送をサスペンドできます。このビットは、ソフトウェアによってセット／リセットされます。このビットは START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送は一時停止されません。

1：転送は一時停止されます。

ビット 0 **START**：スタート

このビットを使用し、各種設定レジスタにロードしたパラメータに従って DMA2D を開始することができます。このビットは以下のイベントにより自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションがデータ転送をアボートしたとき
- データ転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作を行っているためデータ転送が開始されないとき（自動 CLUT ローディング）。

18.6.2 DMA2D 割り込みステータスレジスタ（DMA2D_ISR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CEIF**：設定エラー割り込みフラグ

このビットは DMA2D_CR、DMA2DFGPFCCR、または DMA2D_BGPFCCR の START ビットがセットされ、誤った設定がプログラムされたときセットされます。

ビット 4 **CTCIF**：CLUT 転送完了割り込みフラグ

このビットは、システムメモリ領域から内部 DMA2D メモリへの CLUT コピーが完了したときセットされます。

ビット 3 **CAEIF**：CLUT アクセスエラー割り込みフラグ

このビットは、CLUT がシステムメモリから内部 DMA2D へ自動的にコピーされている間に CPU が CLUT にアクセスした場合にセットされます。

ビット 2 **TWIF**：転送ウォーターマーク割り込みフラグ

このビットはウォーターマークを付けたラインの最後のピクセルが転送されたときセットされます。

ビット 1 **TCIF**：転送完了割り込みフラグ

このビットは DMA2D 転送動作が完了したときセットされます（データ転送のみ）。

ビット 0 **TEIF**：転送エラー割り込みフラグ

このビットは DMA 転送中（データ転送または自動 CLUT ローディング）にエラーが発生したときセットされます。

18.6.3 DMA2D 割り込みフラグクリアレジスタ（DMA2D_IFCR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CCEIF**：設定エラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CEIF フラグがクリアされます。

ビット 4 **CCTCIF**：CLUT 転送完了割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CTCIF フラグがクリアされます。

ビット 3 **CAECIF**：CLUT アクセスエラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CAEIF フラグがクリアされます。

ビット 2 **CTWIF**：ウォーターマーク転送割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TWIF フラグがクリアされます。

ビット 1 **CTCIF**：転送完了割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TCIF フラグがクリアされます。

ビット 0 **CTEIF**：転送エラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TEIF フラグがクリアされます。

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

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

フォアグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこのレジスタは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、16 ビット／ピクセルフォーマットは 16 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

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

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13:0]**：ラインオフセット

ピクセルで表されるフォアグラウンドに使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこれらは読み出し専用になります。

イメージフォーマットが 4 ビット／ピクセル場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

バックグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこのレジスタは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、16 ビット／ピクセルフォーマットは 16 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

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

アドレスオフセット：0x0018

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13:0]**：ラインオフセット

ピクセルで表されるバックグラウンドに使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこれらは読み出し専用になります。

イメージフォーマットが 4 ビット／ピクセル場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x001C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7:0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、または AM[1:0] ビットによって選択されたアルファモードに従って元のアルファ値を乗算することができます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が

開始されると、このビットは読み出し専用になります。

0：レギュラモード（RGB または ARGB）

1：スワップモード（BGR または ABGR）

ビット 20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット 19:18 **CSS[1:0]**：クロマサブサンプリング

これらのビットによって、YCbCr カラーモードのクロマサブサンプリングモードを定義します。転送が

開始されると、これらのビットは読み出し専用になります。

00：4:4:4 （クロマサブサンプリングなし）

01：4:2:2

10：4:2:0

その他：意味を持ちません。

ビット 17:16 **AM[1:0]**：アルファモード

これらのビットによって、フォアグラウンドイメージに使用するアルファチャネル値を選択します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

00：フォアグラウンドイメージアルファチャネル値は修正されません。

01：元のフォアグラウンドアルファチャネル値を ALPHA[7: 0]によって置き換えます。0]

10：元のフォアグラウンドアルファチャネル値を ALPHA[7:0] に元のアルファチャネル値を掛けた値で置き換えます。

他の設定は意味を持ちません。

ビット 15:8 **CS[7:0]**：CLUT サイズ

これらのビットによって、フォアグラウンドイメージに使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読み出し専用になります。

CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：スタート

このビットをセットして CLUTの自動ローディングを開始することができます。これは次の場合自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

このビットによって、CLUT のカラーフォーマットを定義します。これは転送が無効のときにのみ書き込みできます。CLUT 転送が開始されるとこのビットは読み出し専用になります。

0：ARGB8888

1：RGB888

ビット 3:0 **CM[3:0]**：カラーモード

これらのビットによって、フォアグラウンドイメージのカラーフォーマットを定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

0000：ARGB8888
0001：RGB888
0010：RGB565
0011：ARGB1555
0100：ARGB4444
0101：L8
0110：AL44
0111：AL88
1000：L4
1001：A8
1010：A4
1011：YCbCr
その他：意味を持ちません。

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

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7:0]**：赤の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7:0]**：緑の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7:0]**：青の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

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

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7:0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、または AM[1:0] ビットによって選択されたアルファモードに従って元のアルファ値を乗算することができます。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が
開始されると、このビットは読み出し専用になります。
0：レギュラモード（RGB または ARGB）
1：スワップモード（BGR または ABGR）

ビット 20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。
0：レギュラアルファ
1：アルファ反転

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **AM[1:0]**：アルファモード

これらのビットによって、バックグラウンドイメージに使用するアルファチャネル値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。
00：フォアグラウンドイメージアルファチャネル値は修正されません。
01：元のバックグラウンドアルファチャネル値を ALPHA[7: 0]によって置き換えます。0]
10 : 元のバックグラウンドアルファチャネル値を ALPHA[7:0] に元のアルファチャネル値を掛けた値で置き換えます。
その他：意味を持ちません。

ビット 15:8 **CS[7:0]**：CLUT サイズ

これらのビットによって、BG に使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読み出し専用になります。
CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：スタート

このビットをセットして CLUTの自動ローディングを開始します。このビットは次のときに自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

これらのビットによって、CLUT のカラーフォーマットを定義します。このレジスタは、転送が無効のときにのみ書き込みできます。CLUT 転送が開始されるとこのビットは読み出し専用になります。

0：ARGB8888

1：RGB888

ビット 3:0 **CM[3:0]**：カラーモード

これらのビットによって、バックグラウンドイメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

0000：ARGB8888
0001：RGB888
0010：RGB565
0011：ARGB1555
0100：ARGB4444
0101：L8
0110：AL44
0111：AL88
1000：L4
1001：A8
1010：A4
その他：意味を持ちません。

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

アドレスオフセット：0x0028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7:0]**：赤の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7:0]**：緑の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7:0]**：青の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

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

アドレスオフセット：0x002C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

フォアグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書き込みできます。CLUT 転送が開始されるとこのレジスタは読み出し専用になります。

フォアグラウンド CLUT フォーマットが 32 ビットの場合、アドレスは 32 ビット境界に整列する必要があります。

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

アドレスオフセット：0x0030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

バックグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書き込みできます。CLUT 転送が開始されるとこのレジスタは読み出し専用になります。

バックグラウンド CLUT フォーマットが 32 ビットの場合、アドレスは 32 ビットに調整する必要があります。

18.6.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）

アドレスオフセット：0x0034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.
										rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
													rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が

開始されると、このビットは読み出し専用になります。

0：レギュラモード（RGB または ARGB）

1：スワップモード（BGR または ABGR）

ビット 20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット 19:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CM[2:0]**：カラーモード

これらのビットによって、出力イメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

その他：意味を持ちません。

18.6.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
RED[4:0]					GREEN[5:0]						BLUE[4:0]				
A	RED[4:0]				GREEN[4:0]						BLUE[4:0]				
ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7:0]**：アルファチャネル値

これらのビットによって、出力カラーのアルファチャネルを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:16 **RED[7:0]**：赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7:0]**：緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7:0]**：青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

18.6.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）

アドレスオフセット：0x003C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MA[31:0]**：メモリアドレス

出力 FIFO に使用されるデータのアドレスです。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。例えば、32ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

18.6.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13:0]**：ラインオフセット

ピクセルで表現される出力に使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

18.6.18 DMA2D ライン数レジスタ（DMA2D_NLR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **PL[13:0]**：ラインあたりのピクセル

転送する領域のラインあたりのピクセル数です。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

入力イメージフォーマットのいずれかがピクセルあたり 4 ビットの場合、ラインあたりのピクセル数は偶数である必要があります。

ビット 15:0 **NL[15:0]**：ライン数

転送する領域のライン数です。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

18.6.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LW[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LW[15:0]**：ラインウォーターマーク

これらのビットによって、割り込み生成のラインウォーターマークを設定することができます。ウォーターマークを付けたラインの最後のピクセルが転送されたとき割り込みが発生します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

18.6.20 DMA2D AXI マスタタイマ設定レジスタ（DMA2D_AMTCR）

アドレスオフセット：0x004C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット 31:16 予約済み

ビット 15:8 **DT[7:0]**：デッドタイム

AXI マスタポートに対する 2 つの連続アクセス間に挿入される AXI クロックサイクルのデッドタイム値です。これらのビットは 2 つの連続 AXI アクセス間の最小保証サイクル数を表します。

ビット 7:1 予約済み

ビット 0 **EN**：有効化

デットタイム機能を有効にします。

18.6.21 DMA2D レジスタマップ

次の表に DMA2D レジスタの一覧を示します。DMA2D レジスタベースアドレスについては、*104 ページのセクション 2.2.2* を参照してください。

表 129. DMA2D レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA2D_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[1:0]		Res.	Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	Res.	Res.	Res.	ABORT	SUSP	START
	リセット値															0	0				0	0	0	0	0	0					0	0	0
0x0004	DMA2D_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
	リセット値																											0	0	0	0	0	0
0x0008	DMA2D_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
	リセット値																											0	0	0	0	0	0
0x000C	DMA2D_FGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	DMA2D_FGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA2D_BGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA2D_BGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA2D_FGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0020	DMA2D_FGCOLR	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	DMA2D_BGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0028	DMA2D_BGCOLR	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	DMA2D_FGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	DMA2D_BGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	DMA2D_OPFCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
	リセット値											0	0																		0	0	0
0x0038	DMA2D_OCOLR	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]					
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[4:0]			GREEN[6:0]					BLUE[4:0]						
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	A	RED[4:0]			GREEN[4:0]				BLUE[4:0]						
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALPHA[3:0]			RED[3:0]			GREEN[3:0]			BLUE[3:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	DMA2D_OMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 129. DMA2D レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0040	DMA2D_OOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	DMA2D_NLR	Res.	Res.	PL[13:0]														NL[15:0]															
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	DMA2D_LWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LW[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	DMA2D_AMTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																	0	0	0	0	0	0	0	0								0
0x0050～Ox03FF		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0400～0x07FF	DMA2D_FGCLUT	ALPHA[7:0][255:0]								RED[7:0][255:0]								GREEN[7:0][255:0]								BLUE[7:0][255:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0800～0x0BFF	DMA2D_BGCLUT	ALPHA[7:0][255:0]								RED[7:0][255:0]								GREEN[7:0][255:0]								BLUE[7:0][255:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

19 ネスト化されたベクタ割り込みコントローラ

19.1 NVIC の機能

NVIC には以下の機能があります。

- 最大 150 本の STM32H7xxx 用のマスク可能な割り込みチャネル（FPU 搭載 Cortex®-M7 の 16 本の割り込みラインは数に含まれていません。）
- 16 のプログラム可能な優先レベル（4 ビットの割り込み優先順位を使用）
- 遅延時間の少ない例外および割り込み処理
- 電源管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインタフェースは密に結合され、割り込み処理の遅延時間を小さくし、遅れて到着した割り込みを効率的に処理できます。

コア例外を含むすべての割り込みは、NVIC によって管理されます。

例外と NVIC プログラミングの詳細については、Cortex®-M7 のプログラミングマニュアル PM0253 を参照してください。

19.1.1 SysTick 較正値レジスタ

SysTick 較正値は 18750 に固定され、SysTick クロックを 18.75 MHz（HCLK を 150 MHz に設定した状態で HCLK/8）に設定した状態で 1 ms という基準タイムベースを与えます。

19.1.2 割り込みベクタと例外ベクタ

NVIC に接続された例外ベクタは、Reset、NMI、HardFault、MemManage、Bus Fault、UsageFault、SVCall、DebugMonitor、PendSV、および SysTick です。

表 130. NVIC(1)

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
-	-	-	-	予約済み	0x0000 0000
-	-3	-	リセット	リセット	0x0000 0004
-	-2	-	NMI	ノンマスカブル割り込み。 RCC クロックセキュリティシステム（CSS）は NMI ベクタにリンクされます。	0x0000 0008
-	-1	-	HardFault	あらゆる種類の異常	0x0000 000C
-	0	-	MemManage	メモリ管理	0x0000 0010
-	1	-	BusFault	プリフェッチ異常、 メモリアクセス異常	0x0000 0014
-	2	-	UsageFault	未定義命令または無効状態	0x0000 0018
-	-	-	-	予約済み	0x0000 001C - 0x0000 002B
-	3	-	SVCall	SWI 命令によるシステムサービスコール	0x0000 002C
-	4	-	DebugMonitor	デバッグモニタ	0x0000 0030

表 130. NVIC[1]（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
-	-	-	-	予約済み	0x0000 0034
-	5	-	PendSV	ペンディング可能なシステムサービスリクエスト	0x0000 0038
-	6	-	SysTick	システムティックタイマ	0x0000 003C
wwdg1_it	7	0	WWDG1	ウィンドウ型ウォッチドッグ割り込み	0x0000 0040
exti_pwr_pvd_wkup	8	1	PVD_PVM	PVD 検出割り込み（EXTI ライン経由）	0x0000 0044
exti_tamp_rtc_wkup	9	2	RTC_TAMP_STAMP_CSS_LSE	RTC タンパ、タイムスタンプ	0x0000 0048
lsecss_rcc_it				CSS LSE	
exti_wkup_rtc_wkup	10	3	RTC_WKUP	RTC ウェイクアップ割り込み（EXTI ライン経由）	0x0000 004C
flash_it	11	4	FLASH	フラッシュメモリグローバル割り込み	0x0000 0050
rcc_it	12	5	RCC	RCC グローバル割り込み	0x0000 0054
exti_exti0_wkup	13	6	EXTI0	EXTI ライン 0 割り込み	0x0000 0058
exti_exti1_wkup	14	7	EXTI1	EXTI ライン 1 割り込み	0x0000 005C
exti_exti2_wkup	15	8	EXTI2	EXTI ライン 2 割り込み	0x0000 0060
exti_exti3_wkup	16	9	EXTI3	EXTI ライン 3 割り込み	0x0000 0064
exti_exti4_wkup	17	10	EXTI4	EXTI ライン 4 割り込み	0x0000 0068
dma1_it0	18	11	DMA_STR0	DMA1 Stream0グローバル割り込み	0x0000 006C
dma1_it1	19	12	DMA_STR1	DMA1 Stream1グローバル割り込み	0x0000 0070
dma1_it2	20	13	DMA_STR2	DMA1 Stream2グローバル割り込み	0x0000 0074
dma1_it3	21	14	DMA_STR3	DMA1 Stream3グローバル割り込み	0x0000 0078
dma1_it4	22	15	DMA_STR4	DMA1 Stream4グローバル割り込み	0x0000 007C
dma1_it5	23	16	DMA_STR5	DMA1 Stream5グローバル割り込み	0x0000 0080
dma1_it6	24	17	DMA_STR6	DMA1 Stream6グローバル割り込み	0x0000 0084
adc1_it	25	18	ADC1_2	ADC1 および ADC2グローバル割り込み	0x0000 0088
adc2_it					

表 130. NVIC(1)（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
ttfdcan_intr0_it	26	19	FDCAN1_IT0	FDCAN1 割り込み 0	0x0000 008C
fdcan_intr0_it	27	20	FDCAN2_IT0	FDCAN2 割り込み 0	0x0000 0090
ttfdcan_intr1_it	28	21	FDCAN1_IT1	FDCAN1 割り込み 1	0x0000 0094
fdcan_intr1_it	29	22	FDCAN2_IT1	FDCAN2 割り込み 1	0x0000 0098
exti_exti5_wkup	30	23	EXTI9_5	EXTI ライン [9:5] 割り込み	0x 0000 009C
exti_exti6_wkup					
exti_exti7_wkup					
exti_exti8_wkup					
exti_exti9_wkup					
tim1_brk_it	31	24	TIM1_BRK	TIM1 ブレーク割り込み	0x0000 00A0
tim1_upd_it	32	25	TIM1_UP	TIM1 更新割り込み	0x0000 00A4
tim1_trg_it	33	26	TIM1_TRG_COM	TIM1 トリガおよび遷移割り込み	0x0000 00A8
tim1_cc_it	34	27	TIM_CC	TIM1 キャプチャ／比較割り込み	0x0000 00AC
tim2_it	35	28	TIM2	TIM2 グローバル割り込み	0x0000 00B0
tim3_it	36	29	TIM3	TIM3 グローバル割り込み	0x0000 00B4
tim4_it	37	30	TIM4	TIM4 グローバル割り込み	0x0000 00B8
i2c1_ev_it	38	31	I2C1_EV	I2C1 イベント割り込み	0x0000 00BC
exti_i2c1_ev_wkup					
i2c1_err_it	39	32	I2C1_ER	I2C1 エラー割り込み	0x0000 00C0
i2c2_ev_it	40	33	I2C2_EV	I2C2 イベント割り込み	0x0000 00C4
exti_i2c2_ev_wkup					
i2c2_err_it	41	34	I2C2_ER	I2C2 エラー割り込み	0x0000 00C8
spi1_it	42	35	SPI1	SPI1 グローバル割り込み	0x0000 00CC
exti_spi1_it					
spi2_it	43	36	SPI2	SPI2 グローバル割り込み	0x0000 00D0
exti_spi2_it					
usart1_gbl_it	44	37	USART1	USART1 グローバル割り込み	0x0000 00D4
exti_usart1_wkup					
usart2_gbl_it	45	38	USART2	USART2 グローバル割り込み	0x0000 00D8
exti_usart2_wkup					
usart3_gbl_it	46	39	USART3	USART3 グローバル割り込み	0x0000 00DC
exti_usart3_wkup					

表 130. NVIC[1]（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
exti_exti10_it	47	40	EXTI15_10	EXTI ライン [15:10] 割り込み	0x0000 00E0
exti_exti11_wkup					
exti_exti12_wkup					
exti_exti13_wkup					
exti_exti14_wkup					
exti_exti15_wkup					
exti_rtc_al	48	41	RTC_ALARM	RTC アラーム（A および B）割り込み（EXTI ライン経由）	0x0000 00E4
-	49	42	-	-	0x0000 00E8
tim8_brk_it	50	43	TIM8_BRK_TIM12	TIM8 ブレークおよび TIM12 グローバル割り込み	0x0000 00EC
tim12_gbl_it					
tim8_upd_it	51	44	TIM8_UP_TIM13	TIM8 更新および TIM13 グローバル割り込み	0x0000 00F0
tim13_gbl_it					
tim8_trg_it	52	45	TIM8_TRG_COM_TIM14	TIM8 トリガ／遷移および TIM14 グローバル割り込み	0x0000 00F4
tim14_gbl_it					
tim8_cc_it	53	46	TIM8_CC	TIM8 キャプチャ／比較割り込み	0x0000 00F8
dma1_it7	54	47	DMA1_STR7	DMA1 Stream7 グローバル割り込み	0x0000 00FC
fmc_gbl_it	55	48	FMC	FMC グローバル割り込み	0x0000 0100
sdmmc_gbl_it	56	49	SDMMC1	SDMMC グローバル割り込み	0x0000 0104
tim5_gbl_it	57	50	TIM5	TIM5 グローバル割り込み	0x0000 0108
spi3_it	58	51	SPI3	SPI3 グローバル割り込み	0x0000 010C
exti_spi3_wkup					
usart4_gbl_it	59	52	UART4	UART4 グローバル割り込み	0x0000 0110
exti_uart4_wkup					
uart5_gbl_it	60	53	UART5	UART5 グローバル割り込み	0x0000 0114
exti_uart5_wkup					
tim6_gbl_it	61	54	TIM6_DAC	TIM6 グローバル割り込み	0x0000 0118
dac_unr_it				DAC アンダーランエラー割り込み	
tim7_gbl_it	62	55	TIM7	TIM7 グローバル割り込み	0x0000 011C
dma2_it0	63	56	DMA2_STR0	DMA2 Stream0 割り込み	0x0000 0120
dma2_it1	64	57	DMA2_STR1	DMA2 Stream1 割り込み	0x0000 0124
dma2_it2	65	58	DMA2_STR2	DMA2 Stream2 割り込み	0x0000 0128

表 130. NVIC(1)（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
dma2_it3	66	59	DMA2_STR3	DMA2 Stream3 割り込み	0x0000 012C
dma2_it4	67	60	DMA2_STR4	DMA2 Stream4 割り込み	0x0000 0130
eth_sbd_intr_it	68	61	ETH	Ethernetグローバル割り込み	0x0000 0134
exti_eth_wkup	69	62	ETH_WKUP	Ethernet ウェイクアップ割り込み（EXTI ライン経由）	0x0000 0138
can_cal_it	70	63	FDCAN_CAL	CAN2TX 割り込み	0x0000 013C
NC	71	64	-	-	0x0000 0140
NC	72	65	-	-	0x0000 0144
NC	73	66	-	-	0x0000 0148
NC	74	67	-	-	0x0000 014C
dma2_it5	75	68	DMA2_STR5	DMA2 Stream5 割り込み	0x0000 0150
dma2_it6	76	69	DMA2_STR6	DMA2 Stream6 割り込み	0x0000 0154
dma2_it7	77	70	DMA2_STR7	DMA2 Stream7 割り込み	0x0000 0158
usart6_gbl_it	78	71	USART6	USART6 グローバル割り込み	0x0000 015C
exti_usart6_wkup				USART6 ウェイクアップ割り込み	
i2c3_ev_it	79	72	I2C3_EV	I2C3 イベント割り込み	0x0000 0160
exti_i2c3_ev_wkup					
i2c3_err_it	80	73	I2C3_ER	I2C3 エラー割り込み	0x0000 0164
usb1_out_it	81	74	OTG_HS_EP1_OUT	OTG_HS 出力グローバル割り込み	0x0000 0168
usb1_in_it	82	75	OTG_HS_EP1_IN	OTG_HS 入力グローバル割り込み	0x0000 016C
exti_usb1_wkup	83	76	OTG_HS_WKUP	OTG_HS ウェイクアップ割り込み	0x0000 0170
usb1_gbl_it	84	77	OTG_HS	OTG_HS グローバル割り込み	0x0000 0174
dcmi_it	85	78	DCMI	DCMI グローバル割り込み	0x0000 0178
cryp_it	86	79	CRYP	CRYP グローバル割り込み	0x0000 017C
hash_rng_it	87	80	HASH_RNG	HASH および RNG グローバル割り込み	0x0000 0180
cpu_fpu_it	88	81	FPU	CPU FPU	0x0000 0184
uart7_gbl_it	89	82	UART7	UART7 グローバル割り込み	0x0000 0188
exti_uart7_wkup					
uart8_gbl_it	90	83	UART8	UART8 グローバル割り込み	0x0000 018C
exti_uart8_wkup					
spi4_it	91	84	SPI4	SPI4 グローバル割り込み	0x0000 0190
exti_spi4_wkup					

表 130. NVIC(1)（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
spi5_it	92	85	SPI5	SPI5 グローバル割り込み	0x0000 0194
exti_spi5_wkup					
spi6_it	93	86	SPI6	SPI6 グローバル割り込み	0x0000 0198
exti_spi6_wkup					
sai1_it	94	87	SAI1	SAI1 グローバル割り込み	0x0000 019C
ltdc_it	95	88	LTDC	LCD-TFT グローバル割り込み	0x0000 01A0
ltdc_err_it	96	89	LTDC_ER	LCD-TFT エラー割り込み	0x0000 01A4
dma2d_gbl_it	97	90	DMA2D	DMA2D グローバル割り込み	0x0000 01A8
-	98	91	SAI2	SAI2 グローバル割り込み	0x0000 01AC
-	99	92	QUADSPI	Quad SPI グローバル割り込み	0x0000 01B0
lptim1_it	100	93	LPTIM1	LPTIM1 グローバル割り込み	0x0000 01B4
exti_lptim_wkup					
cec_it	101	94	CEC	HDMI-CEC グローバル割り込み	0x0000 01B8
exti_cec_it					
i2c4_ev_it	102	95	I2C4_EV	I2C4 イベント割り込み	0x0000 01BC
exti_i2c4_ev_it					
i2c4_err_it	103	96	I2C4_ER	I2C4 エラー割り込み	0x0000 01C0
-	104	97	SPDIF	SPDIFRX グローバル割り込み	0x0000 01C4
usb2_out_it	105	98	OTG_FS_EP1_OUT	OTG_FS 出力グローバル割り込み	0x0000 01C8
usb2_in_it	106	99	OTG_FS_EP1_IN	OTG_FS 入力グローバル割り込み	0x0000 01CC
exti_usb2_wkup	107	100	OTG_FS_WKUP	OTG_FS ウェイクアップ	0x0000 01D0
usb2_gbl_it	108	101	OTG_FS	OTG_FS グローバル割り込み	0x0000 01D4
dmamux1_ovr_it	109	102	DMAMUX1_OV	DMAMUX1 オーバーラン割り込み	0x0000 01D8
hrtim1_mst_it	110	103	HRTIM1_MST	HRTIM1 マスタタイマ割り込み	0x0000 01DC
hrtim1_tima_it	111	104	HRTIM1_TIMA	HRTIM1 タイマ A 割り込み	0x0000 01E0
hrtim1_timb_it	112	105	HRTIM_TIMB	HRTIM1 タイマ B 割り込み	0x0000 01E4
hrtim1_timc_it	113	106	HRTIM1_TIMC	HRTIM1 タイマ C 割り込み	0x0000 01E8
hrtim1_timd_it	114	107	HRTIM1_TIMD	HRTIM1 タイマ D 割り込み	0x0000 01EC
hrtim1_time_it	115	108	HRTIM_TIME	HRTIM1 タイマ E 割り込み	0x0000 01F0
hrtim1_fault_it	116	109	HRTIM1_FLT	HRTIM1 フォールト割り込み	0x0000 01F4
dfsdm1_it0	117	110	DFSDM1_FLT0	DFSDM1 フィルタ 0 割り込み	0x0000 01F8
dfsdm1_it1	118	111	DFSDM1_FLT1	DFSDM1 フィルタ 1 割り込み	0x0000 01FC
dfsdm1_it2	119	112	DFSDM1_FLT2	DFSDM1 フィルタ 2 割り込み	0x0000 0200

表 130. NVIC(1)（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
dfsdm1_it3	120	113	DFSDM1_FLT3	DFSDM1 フィルタ 3 割り込み	0x0000 0204
sai3_gbl_it_it	121	114	SAI3	SAI3 グローバル割り込み	0x0000 0208
swpmi_gbl_it	122	115	SWPMI1	SWPMI グローバル割り込み	0x0000 020C
exti_swpmi_wup				SWPMI ウェイクアップ	
tim15_gbl_it	123	116	TIM15	TIM15 グローバル割り込み	0x0000 0210
tim16_gbl_it	124	117	TIM16	TIM16 グローバル割り込み	0x0000 0214
tim17_gbl_it	125	118	TIM17	TIM17 グローバル割り込み	0x0000 0218
-	126	119	MDIOS_WKUP	MDIOS ウェイクアップ	0x0000 021C
mdios_it	127	120	MDIOS	MDIOS グローバル割り込み	0x0000 0220
-	128	121	JPEG	JPEG グローバル割り込み	0x0000 0224
mdma_it	129	122	MDMA	MDMA	0x0000 0228
-	131	124	SDMMC	SDMMC グローバル割り込み	0x0000 0230
hsem_it	132	125	HSEM0	HSEM グローバル割り込み 1	0x0000 0234
-	133	-	-	-	0x0000 0238
-	134	127	ADC3	ADC3 グローバル割り込み	0x0000 023C
-	135	128	DMAMUX2_OVR	DMAMUX2 オーバーラン割り込み	0x0000 0240
bdma_ch0_it	136	129	BDMA_CH1	BDMA チャネル 1 割り込み	0x0000 0244
bdma_ch1_it	137	130	BDMA_CH2	BDMA チャネル 2 割り込み	0x0000 0248
bdma_ch2_it	138	131	BDMA_CH3	BDMA チャネル 3 割り込み	0x0000 024C
bdma_ch3_it	139	132	BDMA_CH4	BDMA チャネル 4 割り込み	0x0000 0250
bdma_ch4_it	140	133	BDMA_CH5	BDMA チャネル 5 割り込み	0x0000 0254
bdma_ch5_it	141	134	BDMA_CH6	BDMA チャネル 6 割り込み	0x0000 0258
bdma_ch6_it	142	135	BDMA_CH7	BDMA チャネル 7 割り込み	0x0000 025C
bdma_ch7_it	143	136	BDMA_CH8	BDMA チャネル 8 割り込み	0x0000 0260
comp_gbl_it	144	137	COMP	COMP1 および COMP2 グローバル割り込み	0x0000 0264
exti_comp1_wkup					
exti_comp2_wkup					
lptim2_it	145	138	LPTIM2	LPTIM2 タイマ割り込み	0x0000 0268
exti_lptim2_wkup					
lptim3_it	146	139	LPTIM3	LPTIM2 タイマ割り込み	0x0000 026C
exti_lptim3_wkup					
lptim4_it	147	140	LPTIM4	LPTIM2 タイマ割り込み	0x0000 0270
exti_lptim4_wkup					

表 130. NVIC[1]（続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
lptim5_it	148	141	LPTIM5	LPTIM2 タイマ割り込み	0x0000 0274
exti_lptim5_wkup					
lpuart_gbl_it	149	142	LPUART	LPUART グローバル割り込み	0x0000 0278
exti_lpuart_rx_it					
exti_lpuart_tx_it					
exti_d1_wwdg1_wkup	150	143	WWDG1_RST	ウィンドウ型ウォッチドッグ割り込み	0x0000 027C
crs_it	151	144	CRS	クロックリカバリシステムグローバル割り込み	0x0000 0280
-	152	145	-	-	0x0000 0284
-	153	146	SAI4	SAI4 グローバル割り込み	0x0000 0288
-	154	147	-	-	0x0000 028C
-	155	148	-	-	0x0000 0290
exti_wkup1_wkup	156	149	WKUP	WKUP1 から WKUP6 ピン	0x0000 0294
exti_wkup2_wkup					
exti_wkup3_wkup					
exti_wkup4_wkup					
exti_wkup5_wkup					
exti_wkup6_wkup					

1. 同じ NVIC 割り込みラインに異なる信号が接続される場合は、論理和がとられます。

20 拡張割り込み／イベントコントローラ（EXTI）

拡張割り込み／イベントコントローラ（EXTI）は、設定可能なダイレクトイベント入力によってウェイクアップを管理します。電源制御にウェイクアップリクエストを供給し、CPU の NVIC と D3 ドメイン DMAMUX2 に割り込みリクエストを生成し、CPU イベント入力へのイベントを生成します。

EXTI ウェイクアップリクエストにより、システムは STOP モードから、CPU は CStop モードからウェイクアップすることができます。

RUN モードでも、割り込みリクエストとイベントリクエスト生成の両方を使用できます。

20.1 EXTI の主な機能

EXTI の主な機能は以下のとおりです。

- すべてのイベント入力で、CPU をウェイクアップし、CPU 割り込みおよび／または CPU イベントを生成することができます。
- 一部のイベント入力で、ユーザは、自律 RUN モードの D3 ドメインをウェイクアップし、D3 ドメイン（DMAMUX2）に割り込みを生成することができます。

非同期イベント入力は 2 つのグループに分類されます。

- 設定可能なイベント（I/O またはペリフェラルからパルスを生成できる信号）には、以下の特徴があります。
  - 選択可能なアクティブトリガエッジ
  - 割り込みペンディングステータスレジスタビット
  - 個別割り込み／イベント生成マスク
  - SW トリガの可能性
  - 設定可能なシステム D3 ドメインのウェイクアップイベントには D3 ペンディングマスクとステータスレジスタがあり、D3 割り込み信号がある場合もあります。
- ダイレクトイベント（ペリフェラルでクリアする必要があるその他のペリフェラルからの割り込みおよびウェイクアップソース）には、以下の特徴があります。
  - 固定立ち上がりエッジアクティブトリガ
  - EXTI に割り込みペンディングステータスレジスタビットなし（割り込みペンディングステータスは、イベントを生成するペリフェラルから供給される）
  - 個別割り込み／イベント生成マスク
  - SW トリガの可能性なし
  - ダイレクトシステム D3 ドメインのウェイクアップイベントには D3 ペンディングマスクとステータスレジスタがあり、D3 割り込み信号がある場合もあります。

20.2 EXTI ブロック図

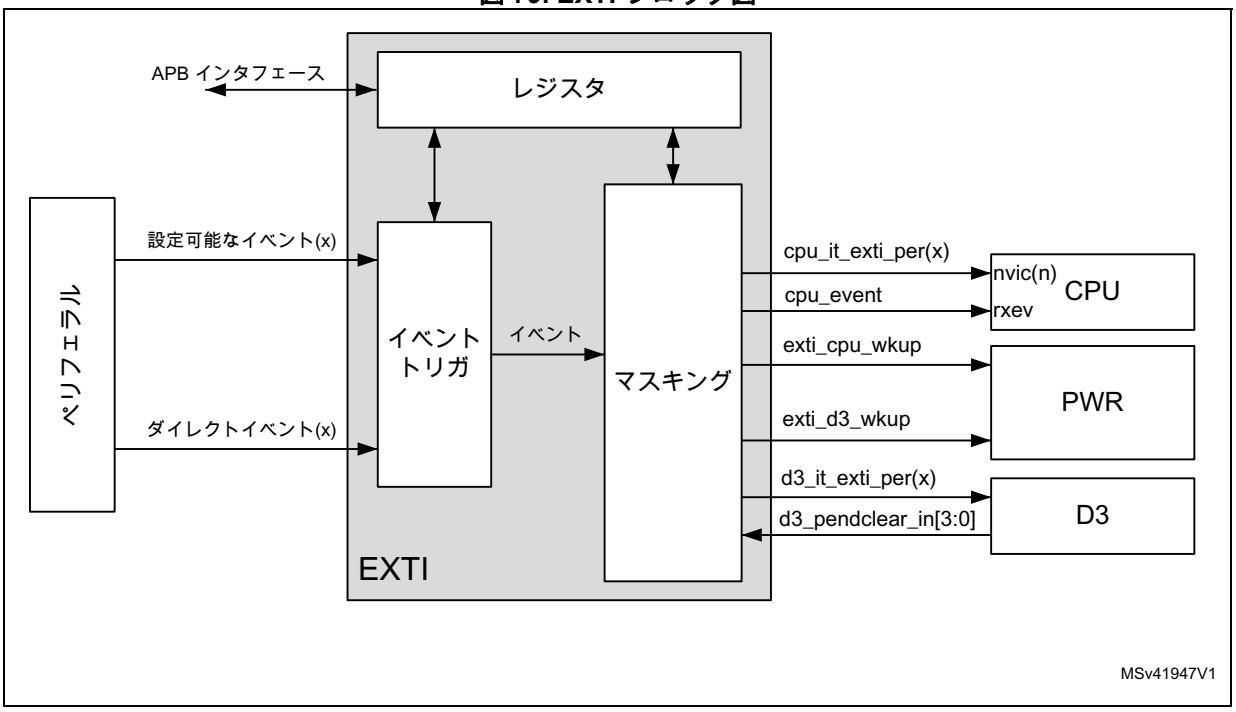
図 79 に示すように、EXTI は APB インタフェースを介してアクセスするレジスタブロック、イベント入力トリガブロック、およびマスキングブロックから成ります。

レジスタブロックは、すべての EXTI レジスタを含んでいます。

イベント入力トリガブロックは、イベント入力エッジのトリガロジックを供給します。

マスキングブロックは、異なるウェイクアップ、割り込み、およびイベントの出力とそのマスキングに、イベント入力を供給します。

図 79. EXTI ブロック図



20.2.1 ペリフェラル、CPU、および D3 ドメイン間の EXTI 接続

システムが STOP モードまたは CPU が CStop モードの場合にウェイクアップイベントを生成できるペリフェラルは、EXTI の設定可能なイベント入力またはダイレクトイベント入力に接続されます。

- パルスを生成するペリフェラル信号は、EXTI の設定可能なイベント入力に接続されます。これらのイベントに、EXTI はクリアする必要がある CPU ステータスペンディングビットを供給します。
- ペリフェラルからクリアする必要があるペリフェラル割り込みやウェイクアップソースは、EXTI のダイレクトイベント入力に接続されます。EXTI には CPU ステータスペンディングビットはありません。割り込みまたはウェイクアップは、ペリフェラルの CPU によってクリアされます。

自律 RUN モードの D3 をウェイクアップすることができるイベント入力には、クリアする必要がある D3 ドメインペンディングリクエスト機能が供給されます。このクリアする必要があるリクエストは、ペンディングクリアセレクタによって選択された信号によって処理されます。

CPU 割り込みは対応する CPU の NVIC に接続され、同様に CPU イベントは CPU rxev 入力に接続されます。

EXTI のウェイクアップ信号は PWR ブロックに接続され、D3 ドメインおよび／または CPU をウェイクアップするために使用されます。

D3 ドメイン割り込みにより、システムは D3 ドメインの自律 RUN モード操作のイベントをトリガすることができます。

20.3 EXTI の機能説明

EXTI のイベント入力タイプとウェイクアップターゲットに応じて、異なるロジックの実装が使用されます。適用可能な機能は、レジスタビットで制御されます。

- *EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）*、*EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）*、*EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）*、および *EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）*、*EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）*、*EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）*によるアクティブなトリガエッジの有効化
- *EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER1）*、*EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER2）*、*EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER3）*によるソフトウェアトリガ
- *EXTI 割り込みマスクレジスタ（EXTI_CPUIMR1）*、*EXTI 割り込みマスクレジスタ（EXTI_CPUIMR2）*、*EXTI 割り込みマスクレジスタ（EXTI_CPUIMR3）*による CPU 割り込みの有効化
- *EXTI イベントマスクレジスタ（EXTI_CPUEMR1）*、*EXTI イベントマスクレジスタ（EXTI_CPUEMR2）*、*EXTI イベントマスクレジスタ（EXTI_CPUEMR3）*による CPU イベントの有効化
- *EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR1）*、*EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR2）*、*EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR3）*による D3 ドメインウェイクアップペンディング

表 131. EXTI イベント入力設定およびレジスタ制御(1)

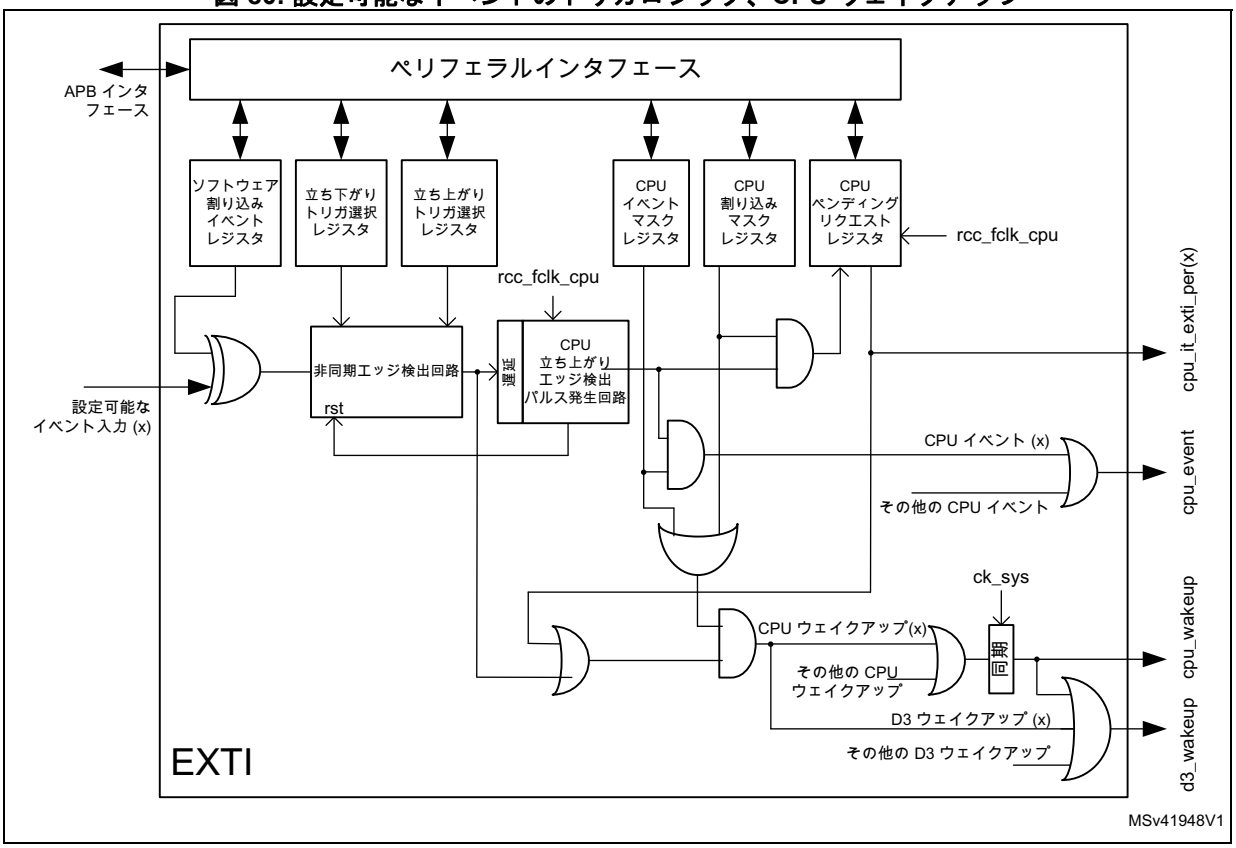
イベント入力タイプ	ウェイクアップターゲット	ロジックの実装	EXTI_RTSR	EXTI_FTSR	EXTI_SWIER	EXTI_CPUIMR	EXTI_CPUEMR	EXTI_D3PMR
設定可能	CPU	設定可能なイベント入力、CPU ウェイクアップロジック	X	X	X	X	X	-
	任意(2)	設定可能なイベント入力、任意のウェイクアップロジック						X
ダイレクト	CPU	ダイレクトイベント入力、CPU ウェイクアップロジック	-	-	-	X	X	-
	任意 (2)	ダイレクトイベント入力、任意のウェイクアップロジック						X

1. X は機能が使用可能であることを示します。
2. 自律 RUN モードの D3 ドメインおよび／または CPU をウェイクアップします。

20.3.1 EXTI の設定可能なイベント入力、CPU のウェイクアップ

図 81 は、必ず CPU をウェイクアップする、設定可能なイベント入力に関連付けられたロジックについて、詳しく表しています。

図 80. 設定可能なイベントのトリガロジック、CPU ウェイクアップ



システムは、ソフトウェア割り込みイベントレジスタにより、*EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER1）*、*EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER2）*、または *EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER3）*のレジスタビットを書き込むことで、ソフトウェアによる設定可能なイベントをトリガできます。

システムは、立ち上がりエッジ（*EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）*、*EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）*、*EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）*）および立ち下がりエッジ（*EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）*、*EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）*、*EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）*）の選択レジスタにより、設定可能なイベントのアクティブトリガエッジまたは両方のエッジを有効にし、選択することができます。

デバイスは、専用の割り込みマスクレジスタ（すなわち*EXTI 割り込みマスクレジスタ（EXTI_CPUIMR1）*と *EXTI 割り込みマスクレジスタ（EXTI_CPUIMR2）*、*EXTI 割り込みマスクレジスタ（EXTI_CPUIMR3）*）と、設定可能なイベントのペンディングリクエストレジスタ（*EXTI ペンディングレジスタ（EXTI_CPUPR1）*、*EXTI ペンディングレジスタ（EXTI_CPUPR2）*、*EXTI ペンディングレジスタ（EXTI_CPUPR3）*）を持っています。CPU のペンディングレジスタは、マスクされていない CPU 割り込みにのみセットされます。イベントごとに、個別の CPU 割り込みが CPU の NVIC に供給されます。設定可能なイベント割り込みは EXTI_CPUPR レジスタでソフトウェアによって確認されなければなりません。

デバイスは、専用のイベントマスクレジスタ（*EXTI イベントマスクレジスタ (EXTI_CPUEMR1)*、*EXTI イベントマスクレジスタ (EXTI_CPUEMR2)*、および*EXTI イベントマスクレジスタ (EXTI_CPUEMR3)*）を持っています。有効化されたイベントにより、その後 CPU にイベントが生成されます。CPU のすべてのイベントの論理和がとられ、ひとつの CPU イベント信号に集約されます。CPU のペンディングレジスタ（EXTI_CPUPR）は、マスクされていない CPU イベントにはセットされません。

CPU 割り込みまたは CPU イベントが有効化されると、非同期エッジ検出回路は、クロック遅延および立ち上がりエッジ検出パルス発生回路によってリセットされます。これにより、CPU クロックは、非同期エッジ検出回路がリセットされる前にウェイクアップすることが保証されます。

注： ***CPU によって有効化された、検出された設定可能なイベントは、CPU のウェイクアップ時にのみクリアされます。***

20.3.2 EXTI の設定可能なイベント入力、任意のウェイクアップ

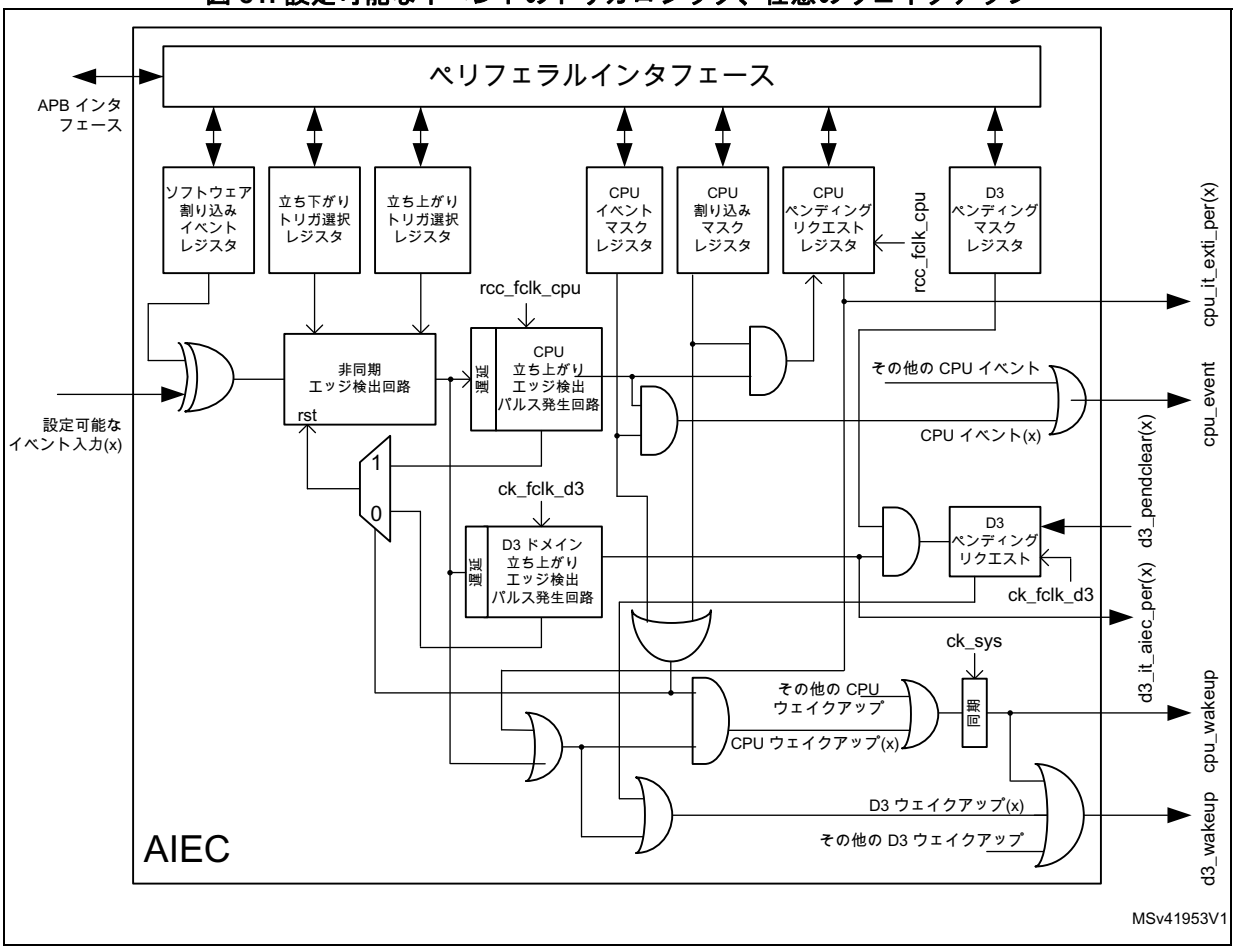
図 81 は、自律 RUN モードおよび／または CPU（「任意」のターゲット）の D3 ドメインをウェイクアップすることができる設定可能なイベント入力に関連付けられたロジックについて、詳しく表しています。設定可能なイベント入力の CPU ウェイクアップと同じ機能を提供し、さらに個別に D3 ドメインをウェイクアップする機能も備えています。

CPU 割り込みおよび CPU イベントがすべて無効化されると、非同期エッジ検出回路は、D3 ドメインのクロック遅延および立ち上がりエッジ検出パルス発生回路によってリセットされます。これにより、D3 ドメインクロックは、非同期エッジ検出回路がリセットされる前にウェイクアップすることが保証されます。

表 132. 設定可能なイベント入力、非同期エッジ検出回路リセット

EXTI_C1IMR	EXTI_C1EMR	非同期エッジ検出回路リセット
両方 = 0		D3 ドメインクロック立ち上がりエッジ検出パルス発生回路
少なくとも 1 つ = 1		CPU クロック立ち上がりエッジ検出パルス発生回路

図 81. 設定可能なイベントのトリガロジック、任意のウェイクアップ



「任意」のターゲットのイベントのトリガロジックには、追加の D3 ペンディングマスクレジスタ、*EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR2）*、*EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR3）*と D3 ペンディングリクエストロジックがあります。D3 ペンディングリクエストロジックは、マスクされていない D3 ペンディングイベントにのみセットされます。D3 ペンディングリクエストロジックでは、選択された D3 ドメインの pendclear ソースによって D3 ペンディングリクエストロジックがクリアされるまで、D3 ドメインを RUN モードで保持します。

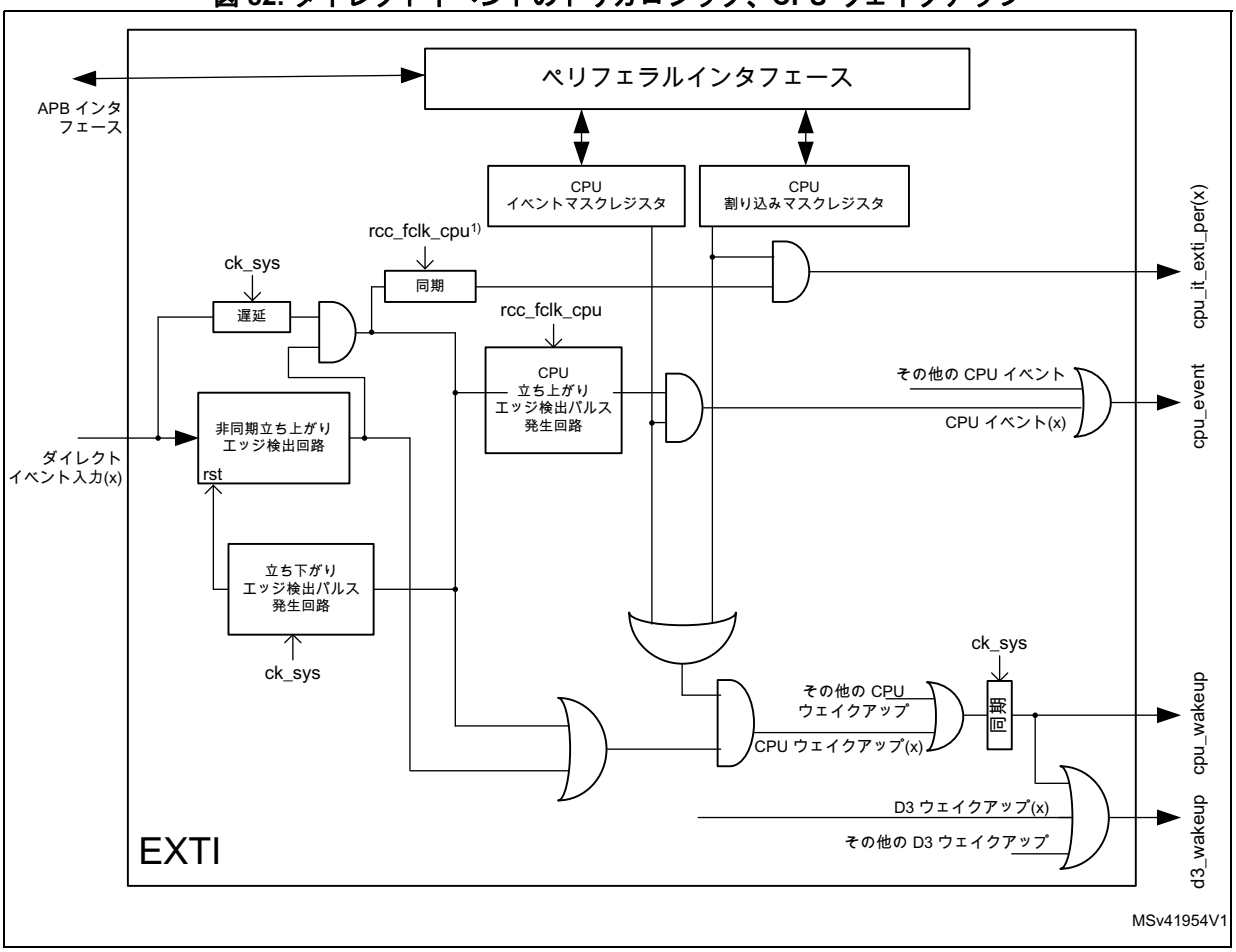
参考資料

20.3.3 EXTI のダイレクトイベント入力、CPU のウェイクアップ

図 82 は、CPU をウェイクアップするダイレクトイベント入力に関連付けられたロジックについて、詳しく表しています。

ダイレクトイベントでは、CPU 割り込みの有効化と CPU イベントの有効化機能のみを提供しています。

図 82. ダイレクトイベントのトリガロジック、CPU ウェイクアップ

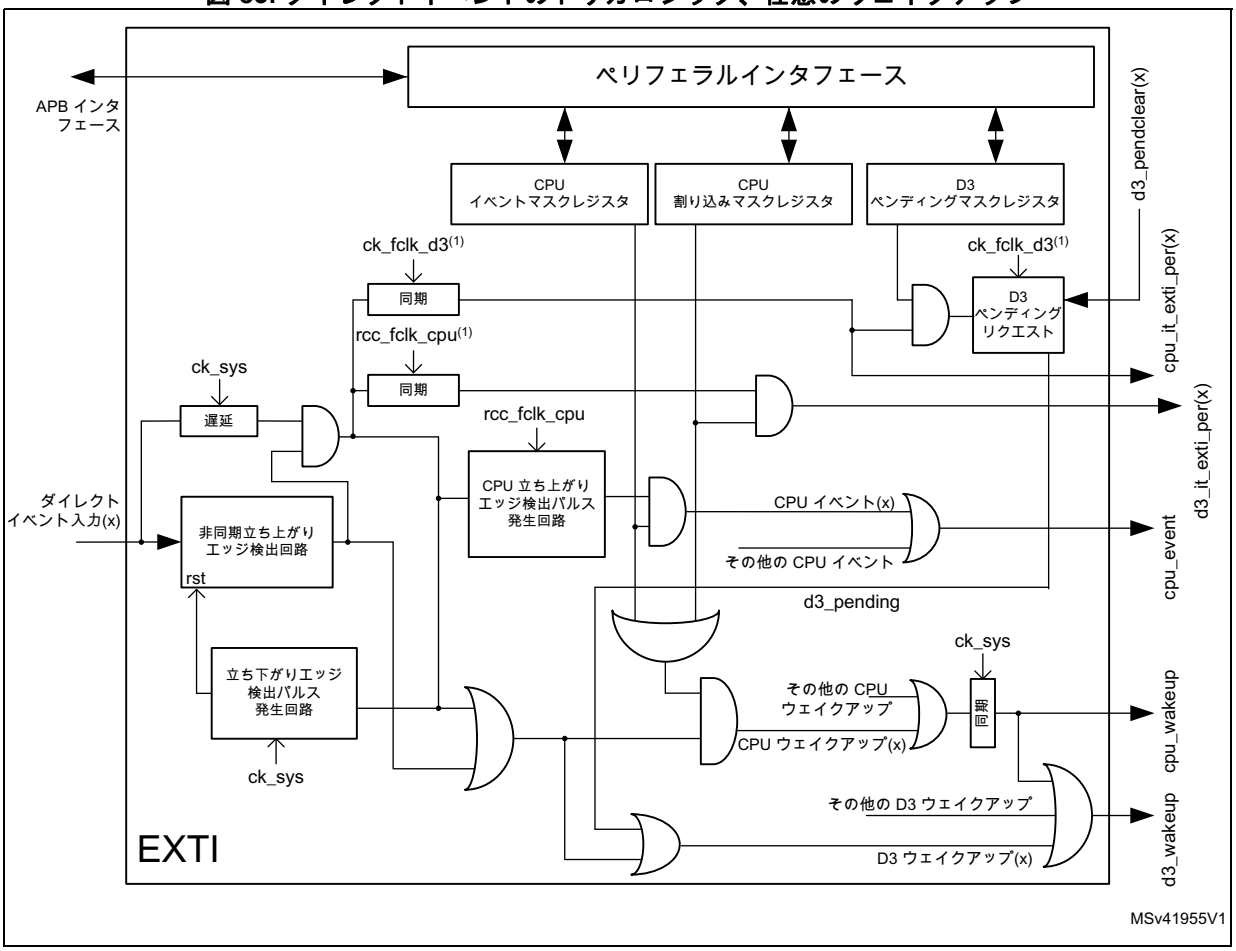


1. 非同期ダイレクトイベント入力の CPU 割り込み（ペリフェラルウェイクアップ信号）は、CPU クロックと同期します。同期ダイレクトイベント入力（ペリフェラル割り込み信号）は、非同期エッジの検出後、再同期することなく直接 CPU 割り込みに送信されます。

20.3.4 EXTI のダイレクトイベント入力、任意のウェイクアップ

図 83 は、自律 RUN モードおよび／または CPU（「任意」のターゲット）の D3 ドメインをウェイクアップするダイレクトイベント入力に関連付けられたロジックについて、詳しく表しています。ダイレクトイベント入力の CPU ウェイクアップと同じ機能を提供し、さらに個別に D3 ドメインをウェイクアップする機能も備えています。

図 83. ダイレクトイベントのトリガロジック、任意のウェイクアップ



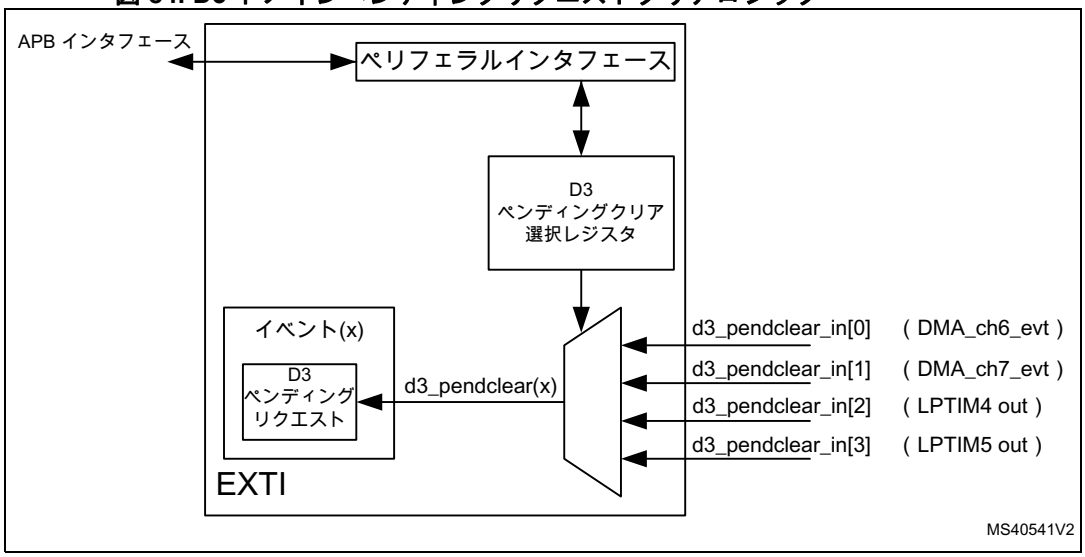
1. 非同期ダイレクトイベント入力の CPU 割り込みおよび D3 ドメイン割り込み（ペリフェラルウェイクアップ信号）は、CPU クロックおよび D3 ドメインクロックとそれぞれ同期します。同期ダイレクトイベント入力（ペリフェラル割り込み信号）は、非同期エッジの検出後、EXTI で再同期することなく直接 CPU 割り込みと D3 ドメイン割り込みに送信されます。

20.3.5 EXTI D3 ペンディングリクエストクリア選択

自律 RUN モードの D3 ドメインをウェイクアップできるイベント入力には、選択された D3 pendclear ソースでクリアできる D3 ペンディングリクエストロジックがあります。各 D3 ペンディングリクエストに、D3 ドメイン pendclear ソースを 4 つの異なる入力から選択できます。

図 84 は、D3 pendclear ソースを選択するロジックについて、詳しく表しています。

図 84. D3 ドメインペンディングリクエストクリアロジック



D3 ペンディングリクエストクリア選択レジスタ（*EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR1L）*、*EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR1H）*、*EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR2L）*、*EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR2H）*、*EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR3L）*および*EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR3H）*）により、システムは D3 ペンディングリクエストをリセットするためにソースを選択することができます。

20.4 EXTI イベント入力の配置

16 の GPIO イベント入力に対し、SYSCFG レジスタ（SYSCFG_EXTICRn）で、関連する IOPORT ピンを選択する必要があります。各 IOPORT の同じピンが、対応する EXTI イベント入力に割り当てられます。

各イベント入力のウェイクアップ機能の詳細については、*表 133* を参照してください。イベント入力により、CPU と自律 RUN モードの D3 ドメイン（「任意」の場合）のどちらかをウェイクアップすることができます。

CPU の NVIC に接続されている EXTI のイベント入力は、*「NVIC への接続」*列に示されています。NVIC に接続されていない EXTI イベントの場合、ペリフェラル割り込みは EXTI への接続と並列して NVIC に直接接続されます。

EXTI のすべてのイベント入力の論理和がとられ、CPU イベント入力（rxev）に接続されます。

表 133. EXTI イベント入力の配置

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
0～15	EXTI[15:0]	設定可能	任意	あり
16	PVD および AVD(1)	設定可能	CPU のみ	あり
17	RTC アラーム	設定可能	CPU のみ	あり
18	RTC タンパ、RTC タイムスタンプ、RCC LSECSS(2)	設定可能	CPU のみ	あり
19	RTC ウェイクアップタイマ	設定可能	任意	あり
20	COMP1	設定可能	任意	あり
21	COMP2	設定可能	任意	あり
22	I2C1 ウェイクアップ	ダイレクト	CPU のみ	あり
23	I2C2 ウェイクアップ	ダイレクト	CPU のみ	あり
24	I2C3 ウェイクアップ	ダイレクト	CPU のみ	あり
25	I2C4 ウェイクアップ	ダイレクト	任意	あり
26	USART1 ウェイクアップ	ダイレクト	CPU のみ	あり
27	USART2 ウェイクアップ	ダイレクト	CPU のみ	あり
28	USART3 ウェイクアップ	ダイレクト	CPU のみ	あり
29	USART6 ウェイクアップ	ダイレクト	CPU のみ	あり
30	UART4 ウェイクアップ	ダイレクト	CPU のみ	あり
31	UART5 ウェイクアップ	ダイレクト	CPU のみ	あり
32	UART7 ウェイクアップ	ダイレクト	CPU のみ	あり
33	UART8 ウェイクアップ	ダイレクト	CPU のみ	あり
34	LPUART1 RX ウェイクアップ	ダイレクト	任意	あり
35	LPUART1 TX ウェイクアップ	ダイレクト	任意	あり
36	SPI1 ウェイクアップ	ダイレクト	CPU のみ	あり
37	SPI2 ウェイクアップ	ダイレクト	CPU のみ	あり
38	SPI3 ウェイクアップ	ダイレクト	CPU のみ	あり
39	SPI4 ウェイクアップ	ダイレクト	CPU のみ	あり
40	SPI5 ウェイクアップ	ダイレクト	CPU のみ	あり
41	SPI6 ウェイクアップ	ダイレクト	任意	あり
42	MDIO ウェイクアップ	ダイレクト	CPU のみ	あり
43	USB1 ウェイクアップ	ダイレクト	CPU のみ	あり
44	USB2 ウェイクアップ	ダイレクト	CPU のみ	あり
45	予約済み	-	-	-
46	予約済み	-	-	-
47	LPTIM1 ウェイクアップ	ダイレクト	CPU のみ	あり
48	LPTIM2 ウェイクアップ	ダイレクト	任意	あり
49	LPTIM2 出力	設定可能	任意	なし(3)

表 133. EXTI イベント入力の配置（続き）

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
50	LPTIM3 ウェイクアップ	ダイレクト	任意	あり
51	LPTIM3 出力	設定可能	任意	なし (3)
52	LPTIM4 ウェイクアップ	ダイレクト	任意	あり
53	LPTIM5 ウェイクアップ	ダイレクト	任意	あり
54	SWPMI ウェイクアップ	ダイレクト	CPU のみ	あり
55	WKUP1	ダイレクト	CPU のみ	あり
56	WKUP2	ダイレクト	CPU のみ	あり
57	WKUP3	ダイレクト	CPU のみ	あり
58	WKUP4	ダイレクト	CPU のみ	あり
59	WKUP5	ダイレクト	CPU のみ	あり
60	WKUP6	ダイレクト	CPU のみ	あり
61	RCC 割り込み	ダイレクト	CPU のみ	なし(4)
62	I2C4 イベント割り込み	ダイレクト	CPU のみ	なし (4)
63	I2C4 エラー割り込み	ダイレクト	CPU のみ	なし (4)
64	LPUART1 グローバル割り込み	ダイレクト	CPU のみ	なし (4)
65	SPI6 割り込み	ダイレクト	CPU のみ	なし (4)
66	BDMA CH0 割り込み	ダイレクト	CPU のみ	なし (4)
67	BDMA CH1 割り込み	ダイレクト	CPU のみ	なし (4)
68	BDMA CH2 割り込み	ダイレクト	CPU のみ	なし (4)
69	BDMA CH3 割り込み	ダイレクト	CPU のみ	なし (4)
70	BDMA CH4 割り込み	ダイレクト	CPU のみ	なし (4)
71	BDMA CH5 割り込み	ダイレクト	CPU のみ	なし (4)
72	BDMA CH6 割り込み	ダイレクト	CPU のみ	なし (4)
73	BDMA CH7 割り込み	ダイレクト	CPU のみ	なし (4)
74	DMAMUX2 割り込み	ダイレクト	CPU のみ	なし (4)
75	ADC3 割り込み	ダイレクト	CPU のみ	なし (4)
76	SAI4 割り込み	ダイレクト	CPU のみ	なし (4)
77	予約済み	-	-	-
78	予約済み	-	-	-
79	予約済み	-	-	-
80	予約済み	-	-	-
81	予約済み	-	-	-
82	予約済み	-	-	-
83	予約済み	-	-	-
84	予約済み	-	-	-
85	HDMI-CEC ウェイクアップ	設定可能	CPU のみ	あり

表 133. EXTI イベント入力の配置（続き）

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
86	ETHERNET ウェイクアップ	設定可能	CPU のみ	あり
87	HSECSS 割り込み	ダイレクト	CPU のみ	なし (4)
88	予約済み	-	-	-

1. PVD と AVD 信号の論理和がとられ、同じEXTIイベント入力に集約されます。
2. RTC タンパ、RTC タイムスタンプ、および RCC LSECSS 信号の論理和がとられ、同じEXTIイベント入力に集約されます。
3. CPU の NVIC では使用できません。システムウェイクアップのみ、または CPU イベント入力（rxev）に使用することができます。
4. CPU の NVIC でペリフェラルから直接使用できます。

20.5 EXTI の機能的挙動

ダイレクトイベント入力は、イベントを生成する各ペリフェラルで有効になります。設定可能なイベントは、少なくとも 1 つのトリガエッジを有効にすることで有効になります。

STOP モードの場合、D3 ドメインは必ずイベントによってウェイクアップします。システム RUN および STOP モードの場合、関連する D3 ドメイン割り込みが必ずイベントによって生成されます。イベントは、イベントに関連する CPU 割り込みおよび／または CPU イベントがマスクされていない場合にのみ CPU をウェイクアップします。

表 134. マスキングの機能

CPU		設定可能なイベント入力 EXTI_CPUPR の PRx ビット	CPU			D3 ドメインウェイクアップ
EXTI_CPUIMR の割り込みイネーブル MRx ビット	EXTI_CPUEMR のイベントイネーブル MRx ビット		割り込み	イベント	ウェイクアップ	
0	0	いいえ	マスク	マスク	マスク	はい(1) ／マスク(2)
0	1	いいえ	マスク	はい	はい	はい
1	0	ステータスラッチ	あり	マスク	あり	はい
1	1	ステータスラッチ	あり	はい	はい	はい

1. システムが自律 RUN モード（任意のターゲット）の D3 ドメインをウェイクアップできるようにするイベント入力のみ。
2. 必ず CPU をウェイクアップするイベント入力のみ。

設定可能なイベント入力の場合、イベント入力時にイネーブルエッジが発生すると、イベントリクエストが生成されます。関連する CPU 割り込みがマスクされていない場合、対応する EXTI_CPUPR の PRx ペンディングビットがセットされ、CPU 割り込み信号がアクティブになります。EXTI_CPUPR の PRx ペンディングビットに「1」を書き込んで、ソフトウェアでクリアします。これによって CPU 割り込みもクリアされます。

ダイレクトイベント入力の場合、関連するペリフェラルで有効にすると、イベントリクエストが立ち上がりエッジでのみ生成されます。対応する CPU ペンディングビットはありません。関連する CPU 割り込みがマスクされていない場合、対応する CPU 割り込み信号がアクティブになります。

CPU イベントでイベントを生成するには、マスクを解除しなければなりません。イベント入力時にイネーブルエッジが発生すると、CPU イベントパルスが生成されます。CPU イベントペンディングビットはありません。

CPU 割り込みおよび CPU イベントの両方を、同じイベント入力で有効にすることができます。両方とも同じイベント入力条件をトリガします。

設定可能なイベント入力の場合、ソフトウェア割り込み／イベントレジスタ（EXTI_SWIER）に「1」を書き込む際に、ソフトウェアによってイベント入力リクエストを生成することができます。

イベント入力が有効になると、CPU 割り込みおよび／または CPU イベントのマスクが解除され、イベント入力によって CPU ウェイクアップの次に D3 ドメインウェイクアップも生成されます。

一部のイベント入力では自律 RUN モードの D3 ドメインをウェイクアップすることができ、この場合、CPU 割り込みおよび CPU イベントはマスクされ、CPU のウェイクアップが阻止されます。2 つの自律 RUN モードの D3 ドメインのウェイクアップメカニズムをサポートしています。

- ペンディングなしの D3 ドメインウェイクアップ（EXTI_D3PMR = 0）
  - 設定可能なイベント入力では、遅延および立ち上がりエッジ検出パルス発生回路の後に、D3 ドメインをウェイクアップし、また D3 ドメインウェイクアップ信号を自動的にクリアします。
  - ダイレクトイベント入力では、ダイレクトイベント入力信号をクリアした後で、D3 ドメインをウェイクアップし、また D3 ドメインウェイクアップ信号をクリアします。
- ペンディングありの D3 ドメインウェイクアップ（EXTI_D3PMR = 1）
  - 設定可能なイベント入力では、遅延および立ち上がりエッジ検出パルス発生回路の後に、D3 ペンディングリクエストがクリアされた場合に、D3 ドメインをウェイクアップし、また D3 ドメインウェイクアップ信号をクリアします。
  - ダイレクトイベント入力では、ダイレクトイベント入力信号をクリアした後で、D3 ペンディングリクエストがクリアされた場合に、D3 ドメインをウェイクアップし、また D3 ドメインウェイクアップ信号をクリアします。

20.5.1 EXTI CPU 割り込み手順

- EXTI_CPUIMR レジスタに対応するマスクビットをセットすることでイベント入力割り込みのマスクを解除します。
- 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
- CPU の NVIC の関連する割り込みソースを有効にするか、SEVONPEND を使用して、WFI/WFE 命令の後で CPU 割り込み信号からの割り込みを CPU で検出できるようにします。
  - 設定可能なイベント入力の場合、関連する EXTI ペンディングビットをクリアする必要があります。

20.5.2 EXTI CPU イベント手順

- EXTI_CPUEMR レジスタの対応するマスクビットをセットすることでイベント入力のマスクを解除します。
- 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
- CPU イベント信号は、WFE 命令の後で CPU によって検出されます。
  - 設定可能なイベント入力の場合、クリアする EXTI ペンディングビットはありません。

20.5.3 EXTI CPU ウェイクアップ手順

- EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを少なくとも 1 つセットすることで、イベント入力のマスクを解除します。CPU ウェイクアップは、マスクされていない CPU 割り込みおよび／または CPU イベントと同時に生成されます。
- 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
- ダイレクトイベントでは自動的に CPU ウェイクアップを生成します。

20.5.4 自律 RUN モードの D3 ドメインウェイクアップ手順

- EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを両方ともクリアすることで、CPU をウェイクアップするイベント入力をマスクします。
- 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
- ダイレクトイベントでは自動的に D3 ドメインウェイクアップを生成します。
- EXTI_D3PMR で D3 ドメインのウェイクアップメカニズムを選択します。
  - ペンディングなしの D3 ドメインウェイクアップ（EXTI_PMR = 0）を選択した場合、ウェイクアップはイベント入力をクリアした後で自動的にクリアされます。
  - ペンディングありの D3 ドメインウェイクアップ（EXTI_PMR = 1）を選択した場合、ウェイクアップは選択された D3 ドメインの pendclear ソースでクリアする必要があります。保留中の D3 ドメインウェイクアップ信号は、関連する EXTI_D3PMR レジスタビットをクリアする FW でクリアすることもできます。
- D3 ドメインのウェイクアップ後に、D3 ドメイン割り込みが生成されます。
  - 設定可能なイベント入力により、D3 ドメイン割り込みにパルスが生成されます。
  - ダイレクトイベント入力は、ペリフェラルでイベント入力がクリアされるまで、D3 ドメイン割り込みをアクティブにします。

20.5.5 EXTI ソフトウェア割り込み／イベントトリガ手順

設定可能なイベント入力のいずれかを、ソフトウェア割り込み／イベントレジスタからトリガすることができます（関連する CPU 割り込みおよび／または CPU イベントを、それぞれに対応する手順で有効にします）。

- EXTI_RTSR および／または EXTI_FTSR レジスタの対応するエッジトリガビットを少なくとも 1 つセットすることで、イベント入力を有効にします。
- EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを少なくとも 1 つセットすることで、ソフトウェア割り込み／イベントトリガのマスクを解除します。
- EXTI_SWIER レジスタの対応するビットに「1」を書き込むことで、ソフトウェア割り込み／イベントをトリガします。
- EXTI_RTSR および EXTI_FTSR レジスタビットをクリアすることで、イベント入力を無効にすることができます。

注： ***設定可能なイベント入力のエッジでも、割り込み／イベントをトリガできます。***

D3 ペンディングリクエストロジックをセットするためにソフトウェアトリガを使用できます。D3 ペンディングリクエストロジックがクリアされるまで、D3 ドメインを RUN モードで保持します。

20.6 EXTI レジスタの説明

すべてのレジスタに 32 ビット（ワード）でのみアクセスできます。バイトまたはハーフワードの読み出しまたは書き込みはできません。

20.6.1 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）

アドレスオフセット：0x00
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR21	TR20	TR19	TR18	TR17	TR16
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:0 **TRx**：設定可能なイベント入力 x の立ち上がりトリガイベント設定ビット(1)

0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.2 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）

アドレスオフセット：0x04
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR21	TR20	TR19	TR18	TR17	TR16
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:0 **TRx**：設定可能なイベント入力 x の立ち下がりトリガイベント設定ビット(1)

0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち下がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.3 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER1）

アドレスオフセット：0x08
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWIER 15	SWIER 14	SWIER 13	SWIER 12	SWIER 11	SWIER 10	SWIER 9	SWIER 8	SWIER 7	SWIER 6	SWIER 5	SWIER 4	SWIER 3	SWIER 2	SWIER 1	SWIER 0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:0 **SWIERx**：ライン x のソフトウェア割り込み
読み出し時に必ず 0 を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

20.6.4 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR1）

アドレスオフセット：0x0C
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MR25	Res.	Res.	Res.	MR21	MR20	MR19	Res.	Res.	Res.
						rw				rw	rw	rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値（0）に保持する必要があります。

ビット 25 **MRx**：イベント入力 x の D3 ペンディングマスク
0：ライン x からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。
1：ライン x からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 24:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:19 **MRx**：イベント入力 x の D3 ペンディングマスク

0：ライン x からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。

1：ライン x からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 18:16 予約済みであり、リセット値（0）に保持する必要があります。

ビット 15:0 **MRx**：イベント入力 x の D3 ペンディングマスク

0：ライン x からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。

1：ライン x からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

20.6.5 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR1L）

アドレスオフセット：0x10
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCS15		PCS14		PCS13		PCS12		PCS11		PCS10		PCS9		PCS8	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCS7		PCS6		PCS5		PCS4		PCS3		PCS2		PCS1		PCS0	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PCSx**：イベント入力 x = truncate (n/2) の D3 ペンディングリクエストクリア入力信号の選択

00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。

01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。

10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。

11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

20.6.6 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR1H）

アドレスオフセット：0x14
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS25		Res.	Res.
												rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PCS21		PCS20		PCS19		Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw	rw	rw						

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19:18 **PCSx**：イベント入力 x = truncate ((n+32)/2) の D3 ペンディングリクエストクリア入力信号の選択
00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。
01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。
10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。
11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

ビット 17:12 予約済みであり、リセット値（0）に保持する必要があります。

ビット 11:6 **PCSx**：イベント入力 x = truncate ((n+32)/2) の D3 ペンディングリクエストクリア入力信号の選択
00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。
01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。
10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。
11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

ビット 5:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.7 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）

アドレスオフセット：0x20
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR51	Res.	TR49	Res.
												rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19 **TRx**：設定可能なイベント入力 x+32 の立ち上がりトリガイベント設定ビット(1)
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 18 予約済みであり、リセット値（0）に保持する必要があります。

ビット 17 **TRx**：設定可能なイベント入力 x+32 の立ち上がりトリガイベント設定ビット (1)
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 16:0 予約済みであり、リセット値（0）に保持する必要があります。

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.8 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）

アドレスオフセット：0x24
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR51	Res.	TR49	Res.
												rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19 **TRx**：設定可能なイベント入力 x+32 の立ち下がりトリガイベント設定ビット(1)
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 18 予約済みであり、リセット値（0）に保持する必要があります。

ビット 17 **TRx**：設定可能なイベント入力 x+32 の立ち下がりトリガイベント設定ビット (1)
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 16:0 予約済みであり、リセット値（0）に保持する必要があります。

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち下がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.9 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER2）

アドレスオフセット：0x28
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER 51	Res.	SWIER 49	Res.
												rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19 **SWIERx**：ライン x+32 のソフトウェア割り込み
読み出し時に必ず 0 を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 18 予約済みであり、リセット値（0）に保持する必要があります。

ビット 17 **SWIERx**：ライン x+32 のソフトウェア割り込み
読み出し時に必ず 0 を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 16:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.10 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR2）

アドレスオフセット：0x2C
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR53	MR52	MR51	MR50	MR49	MR48
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MR41	Res.	Res.	Res.	Res.	Res.	MR35	MR34	Res.	Res.
						rw						rw	rw		

ビット 31:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:16 **MRx**：イベント入力 x+32 の D3 ペンディングマスク
0：ライン x+32 からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。
1：ライン x+32 からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 15:10 予約済みであり、リセット値（0）に保持する必要があります。

ビット 9 **MRx**：イベント入力 x+32 の D3 ペンディングマスク
0：ライン x+32 からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。
1：ライン x+32 からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 8:4 予約済みであり、リセット値（0）に保持する必要があります。

ビット 3:2 **MRx**：イベント入力 x+32 の D3 ペンディングマスク
0：ライン x+32 からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。
1：ライン x+32 からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 1:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.11 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR2L）

アドレスオフセット：0x30
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS41		Res.	Res.
												rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS35		PCS34		Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19:18 **PCSx**：イベント入力 x = truncate ((n+64)/2) の D3 ペンディングリクエストクリア入力信号の選択
00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。
01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。
10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。
11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

ビット 17:8 予約済みであり、リセット値（0）に保持する必要があります。

ビット 7:4 **PCSx**：イベント入力 x = truncate ((n+64)/2) の D3 ペンディングリクエストクリア入力信号の選択
00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。
01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。
10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。
11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

ビット 3:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.12 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR2H）

アドレスオフセット：0x34
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PCS53		PCS52		PCS51		PCS50		PCS49		PCS48	
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値（0）に保持する必要があります。

ビット 11:0 **PCSx**：イベント入力 x = truncate ((n+96)/2) の D3 ペンディングリクエストクリア入力信号の選択
00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。
01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。
10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。
11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

20.6.13 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）

アドレスオフセット：0x40
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR86	TR85	TR84	Res.	TR82	Res.	Res.
									rw	rw	rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:23 予約済みであり、リセット値（0）に保持する必要があります。

ビット 22:20 **TRx**：設定可能なイベント入力 x+64 の立ち上がりトリガイベント設定ビット(1)
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 19 予約済みであり、リセット値（0）に保持する必要があります。

ビット 18 **TRx**：設定可能なイベント入力 x+64 の立ち上がりトリガイベント設定ビット (1)
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 17:0 予約済みであり、リセット値（0）に保持する必要があります。

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.14 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）

アドレスオフセット：0x44
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR86	TR85	TR84	Res.	TR82	Res.	Res.
									rw	rw	rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:23 予約済みであり、リセット値（0）に保持する必要があります。

ビット 22:20 **TRx**：設定可能なイベント入力 x+64 の立ち下がりトリガイベント設定ビット(1)
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 19 予約済みであり、リセット値（0）に保持する必要があります。

ビット 18 **TRx**：設定可能なイベント入力 x+64 の立ち下がりトリガイベント設定ビット (1)
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 17:0 予約済みであり、リセット値（0）に保持する必要があります。

1. 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書き込み中に設定可能なイベント入力で立ち下がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.15 EXTI ソフトウェア割り込みイベントレジスタ（EXTI_SWIER3）

アドレスオフセット：0x48
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER 86	SWIER 85	SWIER 84	Res.	SWIER 82	Res.	Res.
									rw	rw	rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:23 予約済みであり、リセット値（0）に保持する必要があります。

ビット 22:20 **SWIERx**：ライン x+64 のソフトウェア割り込み
読み出し時に必ず 0 を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 19 予約済みであり、リセット値（0）に保持する必要があります。

ビット 18 **SWIERx**：ライン x+64 のソフトウェア割り込み
読み出し時に必ず 0 を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 17:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.16 EXTI D3 ペンディングマスクレジスタ（EXTI_D3PMR3）

アドレスオフセット：0x4C
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:25 予約済みであり、リセット値（0）に保持する必要があります。

ビット 24 **MRx**：イベント入力 x+64 の D3 ペンディングマスク

0：ライン x+64 からの D3 ペンディングリクエストはマスクされます。このビットに 0 を書き込むことで、D3 ペンディングリクエストもクリアされます。

1：ライン x+64 からの D3 ペンディングリクエストはマスクされません。トリガ時の D3 ドメインのペンディング信号は、クリアされるまで D3 ドメインのウェイクアップをアクティブな状態で保持します。

ビット 23:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.17 EXTI D3 ペンディングクリア選択レジスタ、ロー（EXTI_D3PCR3L）

アドレスオフセット：0x50
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.18 EXTI D3 ペンディングクリア選択レジスタ、ハイ（EXTI_D3PCR3H）

アドレスオフセット：0x54
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS88	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:18 予約済みであり、リセット値（0）に保持する必要があります。

ビット 17:16 **PCSx**：イベント入力 x = truncate ((n+160)/2) の D3 ペンディングリクエストクリア入力信号の選択

00：DMA ch6 イベントが D3 ドメイン pendclear ソースとして選択されます。

01：DMA ch7 イベントが D3 ドメイン pendclear ソースとして選択されます。

10：LPTIM4 出力が D3 ドメイン pendclear ソースとして選択されます。

11：LPTIM5 出力が D3 ドメイン pendclear ソースとして選択されます。

ビット 15:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.19 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR1）

アドレスオフセット：0x80

リセット値：0xFFC0 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR31	MR30	MR29	MR28	MR27	MR26	MR25	MR24	MR23	MR22	MR21	MR20	MR19	MR18	MR17	MR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 **MRx**：ダイレクトイベント入力 x の CPU 割り込みマスク(1)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

ビット 21:0 **MRx**：設定可能なイベント入力 x の CPU 割り込みマスク (2)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

1. デフォルトで割り込みを有効にするために、ダイレクトイベント入力のリセット値は「1」にセットされます。
2. デフォルトで割り込みを無効にするために、設定可能なイベント入力のリセット値は「0」に設定されます。

20.6.20 EXTI イベントマスクレジスタ（EXTI_CPUEMR1）

アドレスオフセット：0x84

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR31	MR30	MR29	MR28	MR27	MR26	MR25	MR24	MR23	MR22	MR21	MR20	MR19	MR18	MR17	MR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MRx**：イベント入力 x の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

20.6.21 EXTI ペンディングレジスタ（EXTI_CPUPR1）

アドレスオフセット：0x88

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR21	PR21	PR19	PR18	PR17	PR16
										rc1	rc1	rc1	rc1	rc1	rc1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PR15	PR14	PR13	PR12	PR11	PR10	PR9	PR8	PR7	PR6	PR5	PR4	PR3	PR2	PR1	PR0
rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1

ビット 31:22 予約済みであり、リセット値（0）に保持する必要があります。

ビット 21:0 **PRx**：設定可能なイベント入力 x のペンディングビット

0：トリガリクエストは発生していません。

1：選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが外部割り込みラインで発生したときにセットされます。このビットに「1」を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

20.6.22 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR2）

アドレスオフセット：0x90

リセット値：0xFFF5 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR63	MR62	MR61	MR60	MR59	MR58	MR57	MR56	MR55	MR54	MR53	MR52	MR51	MR50	MR49	MR48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR47	MR46	Res.	MR44	MR43	MR42	MR41	MR40	MR39	MR38	MR37	MR36	MR35	MR34	MR33	MR32
rw	rw	1	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 **MRx**：ダイレクトイベント入力 x+32 の CPU 割り込みマスク(1)

0：ライン x からの割り込みリクエストはマスクされます。

1：ライン x からの割り込みリクエストはマスクされません。

ビット 19 **MRx**：設定可能なイベント入力 x+32 の CPU 割り込みマスク (2)

0：ライン x からの割り込みリクエストはマスクされます。

1：ライン x からの割り込みリクエストはマスクされません。

ビット 18 **MRx**：ダイレクトイベント入力 x+32 の CPU 割り込みマスク (1)

0：ライン x からの割り込みリクエストはマスクされます。

1：ライン x からの割り込みリクエストはマスクされません。

ビット 17 **MRx**：設定可能なイベント入力 x+32 の CPU 割り込みマスク (2)

0：ライン x からの割り込みリクエストはマスクされます。

1：ライン x からの割り込みリクエストはマスクされません。

ビット 16:14 **MRx**：ダイレクトイベント入力 x+32 の CPU 割り込みマスク (1)
0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

ビット 13 予約済みであり、リセット値（1）に保持する必要があります。

ビット 12:0 **MRx**：ダイレクトイベント入力 x+32 の CPU 割り込みマスク (1)
0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

1. デフォルトで割り込みを有効にするために、ダイレクトイベント入力のリセット値は「1」にセットされます。
2. デフォルトで割り込みを無効にするために、設定可能なイベント入力のリセット値は「0」に設定されます。

20.6.23 EXTI イベントマスクレジスタ（EXTI_CPUEMR2）

アドレスオフセット：0x94

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR63	MR62	MR61	MR60	MR59	MR58	MR57	MR56	MR55	MR54	MR53	MR52	MR51	MR50	MR49	MR48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR47	MR46	Res.	MR44	MR43	MR42	MR41	MR40	MR39	MR38	MR37	MR36	MR35	MR34	MR33	MR32
rw	rw	0	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:14 **MRx**：イベント入力 x+32 の CPU イベントマスク
0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

ビット 13 予約済みであり、リセット値（0）に保持する必要があります。

ビット 12:0 **MRx**：イベント入力 x+32 の CPU イベントマスク
0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

20.6.24 EXTI ペンディングレジスタ（EXTI_CPUPR2）

アドレスオフセット：0x98

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR51	Res.	PR49	Res.
												rc1		rc1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:20 予約済みであり、リセット値（0）に保持する必要があります。

ビット 19 **PRx**：設定可能なイベント入力 x+32 のペンディングビット

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが外部割り込みラインで発生したときにセットされます。このビットに「1」を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 18 予約済みであり、リセット値（0）に保持する必要があります。

ビット 17 **PRx**：設定可能なイベント入力 x+32 のペンディングビット

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが外部割り込みラインで発生したときにセットされます。このビットに「1」を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 16:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.25 EXTI 割り込みマスクレジスタ（EXTI_CPUIMR3）

アドレスオフセット：0xA0

リセット値：0x018B FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	MR87	MR86	MR85	MR84	Res.	MR82	Res.	MR80
							rw	rw	rw	rw	rw		rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR79	MR78	MR77	MR76	MR75	MR74	MR73	MR72	MR71	MR70	MR69	MR68	MR67	MR66	MR65	MR64
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値（0）に保持する必要があります。

ビット 24:23 **MRx**：ダイレクトイベント入力 x+64 の CPU 割り込みマスク (1)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

ビット 22:20 **MRx**：設定可能なイベント入力 x+64 の CPU 割り込みマスク (2)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

ビット 19 予約済みであり、リセット値（1）に保持する必要があります。

ビット 18 **MRx**：設定可能なイベント入力 x+64 の CPU 割り込みマスク (2)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

ビット 17 予約済みであり、リセット値（1）に保持する必要があります。

ビット 16:0 **MRx**：ダイレクトイベント入力 x+64 の CPU 割り込みマスク (1)

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

1. デフォルトで割り込みを有効にするために、ダイレクトイベント入力のリセット値は「1」にセットされます。
2. デフォルトで割り込みを無効にするために、設定可能なイベント入力のリセット値は「0」に設定されます。

20.6.26 EXTI イベントマスクレジスタ（EXTI_CPUEMR3）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	MR87	MR86	MR85	MR84	Res.	MR82	Res.	MR80
							rw	rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR79	MR78	MR77	MR76	MR75	MR74	MR73	MR72	MR71	MR70	MR69	MR68	MR67	MR66	MR65	MR64
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値（0）に保持する必要があります。

ビット 24:0 **MRx**：イベント入力 x+64 の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

20.6.27 EXTI ペンディングレジスタ（EXTI_CPUPR3）

アドレスオフセット：0xA8

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR86	PR85	PR84	Res.	PR82	Res.	Res.
									rc1	rc1	rc1		rc1		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:23 予約済みであり、リセット値（0）に保持する必要があります。

ビット 22:20 **PRx**：設定可能なイベント入力 x+64 のペンディングビット

0：トリガリクエストは発生していません。

1：選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが外部割り込みラインで発生したときにセットされます。このビットに「1」を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 19 予約済みであり、リセット値（0）に保持する必要があります。

ビット 18 **PRx**：設定可能なイベント入力 x+64 のペンディングビット

0：トリガリクエストは発生していません。

1：選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが外部割り込みラインで発生したときにセットされます。このビットに「1」を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 17:0 予約済みであり、リセット値（0）に保持する必要があります。

20.6.28 EXTI レジスタマップ

次の表に、EXTI レジスタマップとリセット値を示します。

表 135. 非同期割り込み／イベントコントローラのレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	EXTI_RTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	EXTI_FTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	EXTI_SWIER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	EXTI_D3PMR1	Res.	Res.	Res.	Res.	Res.	Res.	MR[25]	Res.	Res.	Res.	MR[21:19]			Res.	Res.	Res.	MR[15:0]															
	リセット値							0				0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	EXTI_D3PCR1L	PCS[15]		PCS[14]		PCS[13]		PCS[12]		PCS[11]		PCS[10]		PCS[9]		PCS[8]		PCS[7]		PCS[6]		PCS[5]		PCS[4]		PCS[3]		PCS[2]		PCS[1]		PCS[0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	EXTI_D3PCR1H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[25]		Res.	Res.	Res.	Res.	Res.	Res.	PCS[21]		PCS[20]		PCS[19]		Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0	0							0	0	0	0	0	0						
0x20	EXTI_RTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[51]	Res.	TR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x24	EXTI_FTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[51]	Res.	TR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x28	EXTI_SWIER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[51]	Res.	SWIER[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x2C	EXTI_D3PMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[53:48]						Res.	Res.	Res.	Res.	Res.	Res.	MR[41]	Res.	Res.	Res.	Res.	Res.	MR[35]	MR[34]	Res.	Res.
	リセット値											0	0	0	0	0	0							0						0	0		
0x30	EXTI_D3PCR2L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[41]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[35]		PCS[34]		Res.	Res.	Res.	Res.
	リセット値													0	0											0	0	0	0				
0x34	EXTI_D3PCR2H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[53]		PCS[52]		PCS[51]		PCS[50]		PCS[49]		PCS[48]	
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x40	EXTI_RTSR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[86]	TR[85]	TR[84]	Res.	TR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値										0	0	0		0																		

参考資料

表 135. 非同期割り込み／イベントコントローラのレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	EXTI_FTSR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[86]	TR[85]	TR[84]	Res.	TR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値										0	0	0		0																		
0x48	EXTI_SWIER3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[86]	SWIER[85]	SWIER[84]	Res.	SWIER[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値										0	0	0		0																		
0x4C	EXTI_D3PMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0																								
0x50	EXTI_D3PCR3L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x54	EXTI_D3PCR3H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[88]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0																
0x58～0x7C	予約済み																																
0x80	EXTI_CPUIMR1	MR[31:22]										MR[21:0]																					
	リセット値	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	EXTI_CPUEMR1	MR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	EXTI_CPUPR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	EXTI_CPUIMR2	MR[63:52]												MR[51]	MR[50]	MR[49]	MR[48:46]			Res.	MR[44:32]												
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1		1	1	1	1	1	1	1	1	1	1	1	1	1
0x94	EXTI_CPUEMR2	MR[63:46]																		Res.	MR[44:32]												
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x98	EXTI_CPUPR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[51]	Res.	PR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0xA0	EXTI_CPUIMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88]	MR[87]	MR[86]	MR[85]	MR[84]	Res.	MR[82]	Res.	MR[80:64]																
	リセット値								1	1	0	0	0		0		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0xA4	EXTI_CPUEMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88:84]					Res.	MR[82]	Res.	MR[80:64]																
	リセット値								0	0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 135. 非同期割り込み／イベントコントローラのレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xA8	EXTI_CPUPR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[86]	PR[85]	PR[84]	Res.	PR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値										0	0	0		0																		
0xAC～0xBC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

21 巡回冗長検査計算ユニット（CRC）

21.1 概要

CRC（Cyclic Redundancy Check：巡回冗長検査）計算ユニットは、8、16、または 32 ビットデータワードと、ある生成多項式から、CRC コードを得るために使用されます。

他のアプリケーションの中でも、CRC ベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがフラッシュメモリの整合性を確認するひとつの手段となっています。CRC 計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたリファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

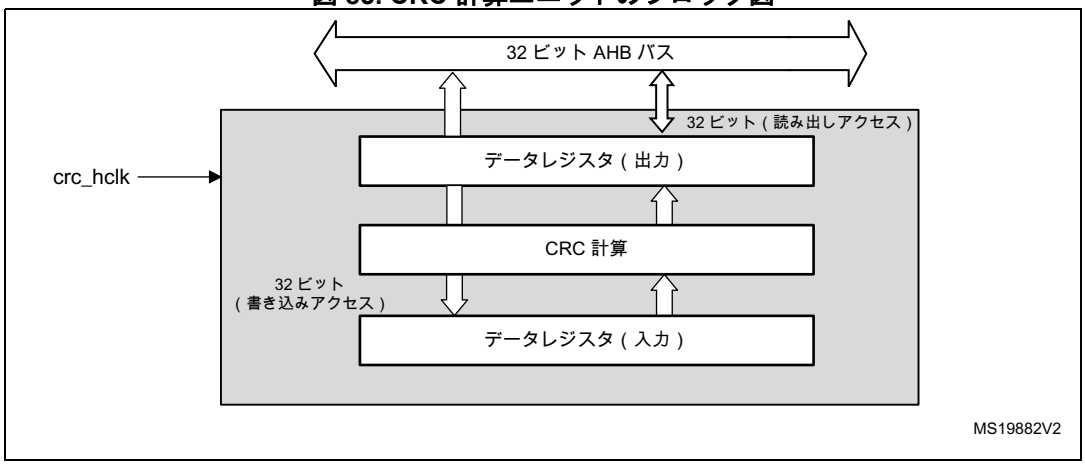
21.2 CRC の主な機能

- 以下のCRC-32 （Ethernet）多項式を使用します。0x4C11DB7
  $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- あるいは、完全にプログラム可能な多項式をプログラム可能なサイズ（7、8、16、32 ビット）で使用します。
- サイズが 8、16、32 ビットのデータを取り扱います。
- プログラム可能な CRC の初期値
- シングル入力／出力 32 ビットデータレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが 32 ビットの場合、CRC の計算は AHB クロック 4 サイクル（HCLK）以内に行われます。
- 汎用 8 ビットレジスタ（一時ストレージとして使用可能）
- I/O データの可逆性のオプション

21.3 CRC の機能説明

21.3.1 CRC のブロック図

図 85. CRC 計算ユニットのブロック図



21.3.2 CRC 内部信号

表 136. CRC 内部入力／出力信号

信号名	信号タイプ	説明
crc_hclk	デジタル入力	AHB クロック

21.3.3 CRC 操作

CRC 計算ユニットは、1 つの 32 ビット読み出し／書き込みデータレジスタ（CRC_DR）を持っています。このレジスタを使用して、新しいデータを入力し（書き込みアクセス）、前回の CRC 計算結果を保持します（読み出しアクセス）。

データレジスタへの書き込み操作のたびに、前回の CRC 値（CRC_DR に格納）と新しい値の組み合わせが作成されます。CRC 計算は、書き込まれるデータのフォーマットに応じ、32 ビットデータワード全体に対して、またはバイト単位で行われます。

CRC_DR レジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32 ビットアクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32 ビットの場合、4 AHB クロックサイクル
- 16 ビットの場合、2 AHB クロックサイクル
- 8 ビットの場合、1 AHB クロックサイクル

入力バッファを使うと、前回の CRC 計算によるウェイトステートを待つことなく、すぐに第 2 のデータを書き込むことができます。

与えられたバイト数に対する書き込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5 バイトの CRC は、1 ワードの書き込みと、それに続く 1 バイトの書き込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CR レジスタの REV_IN[1:0] ビットに応じて、8、16、および 32 ビットで行うことができます。

たとえば、入力データ 0x1A2B3C4D は、CRC 計算では以下のように使用されます。

0x58D43CB2（バイト単位でビットが逆転）

0xD458B23C（ハーフワード単位でビットが逆転）

0xB23CD458（フルワードでビットが逆転）

また、出力データも、CRC_CR レジスタの REV_OUT ビットをセットすることによって逆にすることができます。

操作はビットレベルで行われます。たとえば、出力データ 0x11223344 は 0x22CC4488 に変換されます。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は 0xFFFFFFFF）。

CRC の初期値は、CRC_INIT レジスタでプログラムすることができます。CRC_DR レジスタは、CRC_INIT レジスタの書き込みアクセス時に自動的に初期化されます。

CRC_IDR レジスタを使用して、CRC 計算に関する一時的な値を保持することができます。このレジスタは、CRC_CR レジスタの RESET ビットによる影響を受けません。

多項式のプログラミング可能性

多項式係数は CRC_POL レジスタを介して完全にプログラム可能であり、多項式のサイズは CRC_CR レジスタの POLYSIZE[1:0] ビットをプログラムすることにより、7、8、16、または 32 ビットに設定できます。偶多項式はサポートされていません。

CRC データが 32 ビット未満であれば、その値は CRC_DR レジスタの下位ビットから読み出すことができます。

信頼できる CRC 計算結果を得るために、CRC 計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC 計算が進行中である場合、アプリケーションは計算をリセットするか、または CRC_DR の読み出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32 （Ethernet）多項式、0x4C11DB7

21.4 CRC レジスタ

21.4.1 データレジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															
rw															

ビット 31:0 **DR[31:0]**：データレジスタビット

このレジスタを使用して、CRC 計算機に新しいデータを書き込みます。
読み出し時には、前回の CRC 計算結果を保持します。
データサイズが 32 ビット未満であれば、下位ビットを使用して正しい値の書き込み／読み出しを行います。

21.4.2 独立型データレジスタ（CRC_IDR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IDR[31:0]**：汎用 32 ビットデータレジスタビット

これらのビットは、4 バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CR レジスタの RESET ビットによって生成される CRC リセットの影響を受けません。

21.4.3 制御レジスタ（CRC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
								rw	rw	rw	rw	rw			rs

ビット 31:8　予約済み、クリア状態を保つ必要があります。

ビット 7　**REV_OUT**：出力データを逆にします。

このビットは、出力データビット順序の反転を制御します。

0：ビットの順序は変わりません。

1：出力フォーマットのビットが反転しています。

ビット 6:5　**REV_IN[1:0]**：入力データを逆にします。

これらのビットは、入力データビット順序の反転を制御します。

00：ビットの順序は変わりません。

01：バイト単位でビットが反転しています。

10：ハーフワード単位でビットが反転しています。

11：ワード単位でビットが反転しています。

ビット 4:3　**POLYSIZE[1:0]**：多項式のサイズ

これらのビットは、多項式のサイズを制御します。

00：32 ビットの多項式

01：16 ビットの多項式

10：8 ビットの多項式

11：7 ビットの多項式

ビット 2:1　予約済み、クリア状態を保つ必要があります。

ビット 0　**RESET**：RESET ビット

このビットは、CRC 計算ユニットをリセットし、CRC_INIT レジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

21.4.4 CRC の初期値（CRC_INIT）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRC_INIT[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_INIT[15:0]															
rw															

ビット 31:0 **CRC_INIT**：*プログラム可能な CRC の初期値*

このレジスタを使用して、CRC の初期値を書き込みます。

21.4.5 CRC 多項式（CRC_POL）

アドレスオフセット：0x14

リセット値：0x04C11DB7

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
POL[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL[15:0]															
rw															

ビット 31:0 **POL[31:0]**：プログラム可能な多項式

このレジスタを使用して、CRC 計算に使用される多項式の係数を書き込みます。

多項式のサイズが 32 ビット未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

21.4.6 CRC レジスタマップ

表 137. CRC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRC_DR	DR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	CRC_IDR	IDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	CRC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
	リセット値																									0	0	0	0	0			0
0x10	CRC_INIT	CRC_INIT[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	CRC_POL	多項式係数																															
	リセット値	0x04C11DB7																															

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

22 フレキシブルメモリコントローラ（FMC）

フレキシブルメモリコントローラ（FMC）には、次の 3 つの メモリコントローラが含まれます。

- NOR/PSRAM メモリコントローラ
- NAND メモリコントローラ
- 同期 DRAM（SDRAM／モバイル LPSDR SDRAM）コントローラ

22.1 FMC の主な機能

FMC 機能ブロックは、同期および非同期のスタティックメモリ、SDRAM メモリ、NAND フラッシュメモリとのインタフェースを構成します。主要目的は、

- AXI トランザクションを該当する外部デバイスプロトコルに変換すること
- 外部メモリデバイスのアクセス時間条件を満たすようにすること

すべての外部メモリが、アドレス信号、データ信号、制御信号をコントローラと共用します。各外部デバイスを固有のチップセレクトを使ってアクセスします。FMC は外部デバイスへ一度に 1 回だけアクセスします。

FMC コントローラの主要機能は次の通りです。

- 次のスタティックメモリにマップされたデバイスとのインタフェース
  - スタティックランダムアクセスメモリ（SRAM）
  - NOR 型フラッシュメモリ/OneNAND フラッシュメモリ
  - PSRAM （4 メモリバンク）
  - 最大 8 KB のデータをチェックできる ECC ハードウェア付き NAND フラッシュメモリ
- 同期 DRAM（SDRAM／モバイル LPSDR SDRAM）メモリとのインタフェース
- NOR フラッシュメモリ、PSRAM 、SDRAMなどの同期デバイスに対するアクセスを高速化するバーストモードのサポート
- 非同期アクセスと同期アクセス用のプログラム可能な連続クロック出力
- 8 ビット、16 ビット 、または 32ビット幅のデータバス
- 各メモリバンクに対する独立したチップセレクト制御
- メモリバンクごとに独立した設定
- PSRAM デバイス、SRAM デバイス 、SDRAM デバイスで使用する書き込みイネーブルとバイトレーン選択出力
- 外部非同期ウェイト制御
- 16x32 ビットの深さの書き込み FIFO

  書き込み FIFO はすべてのメモリコントローラに共通であり、以下の構成となっています。
  - メモリへ書き込むデータを格納する書き込みデータ FIFO
  - アドレス（最大 28 ビット）とデータサイズ（最大 2 ビット）を格納する書き込みアドレス FIFO。バーストモードで動作する場合、ページ境界（PSRAM と SDRAM）を跨ぐとき以外は、開始アドレスのみを格納します。この場合、バーストは 2 つの FIFO エントリに分割されます。
- SDRAM コントローラ用の 6x64 ビット深さ（6x14 ビットアドレスタグ）のキャシュ可能読み出し FIFO

開始時に、ユーザーアプリケーションから FMC ピンを設定する必要があります。アプリケーションで使用しない FMC I/O ピンは、他の用途に使用することができます。

外部デバイスタイプおよびその特性を定義する FMC レジスタは、起動時に設定され、次のリセットまたはパワーアップまで変更されません。ただし、以下のいくつかのビットだけなら動作中に変更可能です。

- FMC_PCR レジスタの ECCEN および PBEN ビット
- FMC_SR レジスタの IFS、IRS、および ILS ビット
- FMC_SDCMR レジスタの MODE[2:0]、CTB1/CTB2、NRFS、および MRD ビット
- FMC_SDRTR レジスタの REIE および CRE ビット

以下のシーケンスに従って、FMC の有効時にいくつかのパラメータを変更します。

1. 最初に、FMC コントローラを無効にすることで、レジスタの変更中はそれ以上メモリコントローラへアクセスできないようにします。
2. すべての必要な設定を更新します。
3. FMC コントローラを再び有効にします。

SDRAM コントローラを使用しており、初期化フェーズ後に SDCLK クロック比またはリフレッシュレートを変更する必要がある場合は、以下の手順に従ってください。

1. SDRAM デバイスをセルフリフレッシュモードにします。
2. FMC_BCR1 レジスタの FMCEN ビットをリセットして、FMC コントローラを無効にします。
3. 必要なパラメータを更新します。
4. すべてのパラメータが更新されたら、FMC コントローラを有効にします。
5. 次に、セルフリフレッシュモードを終了するために、クロック設定有効コマンドを送信します。

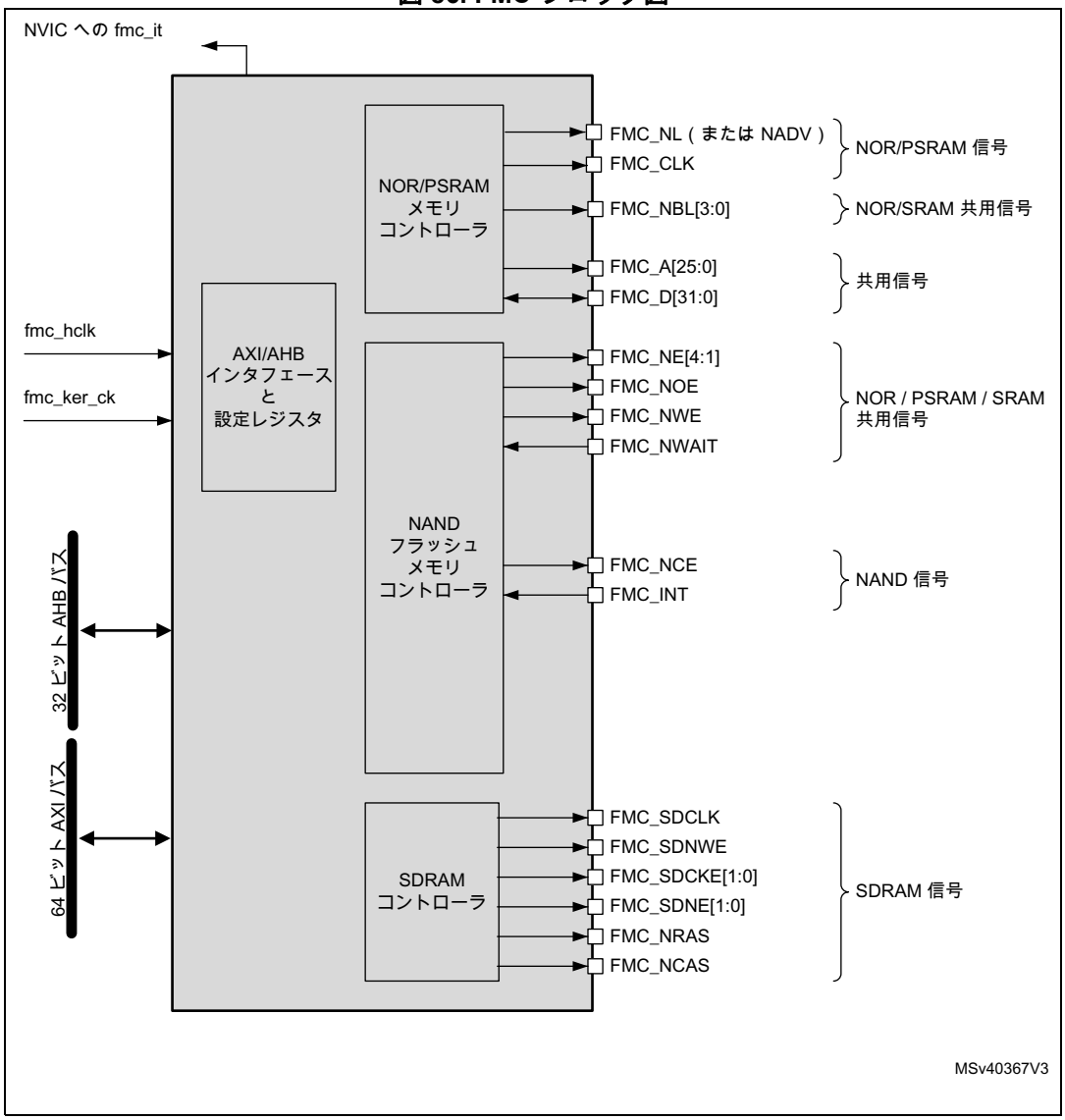
22.2 FMC ブロック図

FMC は 次のメインブロックで構成されます。

- NOR フラッシュ /PSRAM/SRAM コントローラ
- NAND コントローラ
- SDRAM コントローラ
- AXI インタフェース
- AHB インタフェース（FMC 設定レジスタを含む）

以下にブロック図を示します。

図 86. FMC ブロック図



参考資料

22.3 FMC 内部信号

表 138 に FMC 内部信号のリストを示します。FMC ピン（または外部信号）の説明は *セクション 22.7.1：外部メモリインタフェース信号*を参照してください。

表 138. FMC ピン

名称	信号タイプ	説明
fmc_it	デジタル出力	FMC 割り込み
fmc_ker_ck	デジタル入力	FMC カーネルクロック
fmc_hclk	デジタル入力	FMC インタフェースクロック

22.4 AHB インタフェース

AHB スレーブインタフェースを使用することで、内部 CPU による FMC レジスタの設定が可能になります。

AHB クロック（fmc_hclk）は、FMC レジスタアクセス用のリファレンスクロックです。

22.5 AXI インタフェース

AXI スレーブインタフェースを使用することで、内部 CPU やその他のバスマスタペリフェラルから外部メモリへアクセスすることができます。

AXI トランザクションは、外部デバイスプロトコルに変換されます。AXI データバスは 64 ビット幅なので、AXI トランザクションは、アクセスするデータサイズにより、幾つかの連続する 32、16、または 8 ビットアクセスに分割されるかもしれません。FMC チップセレクト（FMC_NEx）は、拡張モードが有効な場合、モード D でのアクセスを除き、連続したアクセスのトグルを行いません。

FMC は、次の条件のうちの一つが満たされると、AXI スレーブエラーが発生します。

- 有効化されていない FMC バンク（バンク 1～4）に対する読み出し／書き込み
- FMC_BCRx レジスタの FACCEN ビットがリセットされている状態での NOR フラッシュバンクに対する読み出し／書き込み
- 書き込み保護された SDRAM バンク（SDRAM_SDCRx レジスタの WP ビットがセット中）に対する書き込み
- SDRAM アドレスレンジの違反（予約済みアドレスレンジへのアクセス）
- まだ初期化されていない SDRAM バンクに対する読み出し／書き込みアクセスの試行

BMAP[1:0]ビット設定で指定されるFMCバンクベースアドレスでサポートされないADDR[31:28] アドレスビット設定の場合、FMC は AXI デコーダエラーを引き起こします。

FMC コントローラのカーネルクロックは非同期の fmc_ker_ck クロックです（fmc_ker_ck クロックソース選択については、*セクション 8：リセットおよびクロック制御（RCC）*を参照のこと）。

22.5.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

アクセスされる外部デバイスのデータ幅は固定ですが、リクエストされる AXI トランザクションのデータ幅は 8、16、32、または 64 ビットのいずれかです。このため、転送に一貫性が無くなる可能性があります。

そこで、AXI トランザクションサイズとメモリデータサイズに応じて、以下に示す単純なトランザクション規則が適用できます。

- 問題が起こらないようにするには、AXI トランザクションサイズとメモリデータサイズを等しくします。
- AXI トランザクションサイズがメモリサイズより大きい場合：

  この場合、FMC は、AXI トランザクションを外部データ幅に合わせて、より小さな連続したメモリアクセスに分割します。
- AXI トランザクションサイズがメモリサイズより小さい場合：

  外部デバイスのタイプによっては、転送に一貫性がなくなる場合があります：
  - バイト選択機能を持つデバイス（SRAM、ROM、PSRAM、SDRAM）に対するアクセス

    この場合、FMC は、読み出し／書き込みトランザクションを許可して、バイトレーン NBL[3:0] を使用して正しいデータにアクセスします。

    書き込み対象バイトは NBL[3:0] によりアドレス指定されます。

    すべてのメモリバイトが読み出され（読み出しトランザクション中 NBL[3:0] はローに駆動されます）、不要なバイトは無視されます。
  - バイト選択機能を持たないデバイス（NOR および NAND フラッシュメモリ）に対するアクセス

    この状況は、16 ビット幅のフラッシュメモリへのバイトアクセスが要求されたときに発生します。バイトモードではデバイスにアクセスできないため（フラッシュメモリに対しては 16 ビットワードの読み出し／書き込みのみが可能）、読み出しトランザクションは許可されますが、書き込みトランザクションは許可されません（コントローラは 16 ビットメモリワードを全部読み出して、必要なバイトのみを使用）。

NOR 型フラッシュ／PSRAM および SDRAMのラップサポート

同期メモリは、すべてのマスタがラップトランザクションを発行できるとは限らないため、不定長のリニアバーストモードに設定する必要があります。

マスタがラップトランザクションを生成すると、

- 読み出しは、2 つのリニアバーストトランザクションに分割されます。
- 書き込みは、書き込み FIFO が有効な場合は 2 つのリニアバーストトランザクションに、無効な場合はいくつかのリニアバーストトランザクションに分割されます。

設定レジスタ

FMCは使用するメモリにあわせて設定を行います。NOR フラッシュ／PSRAM コントローラレジスタの詳細については、*セクション 22.7.6*を参照してください。また、*セクション 22.8.7*は、NAND フラッシュレジスタの詳細説明 、*セクション 22.9.5* は SDRAM コントローラレジスタの詳細を説明しています。

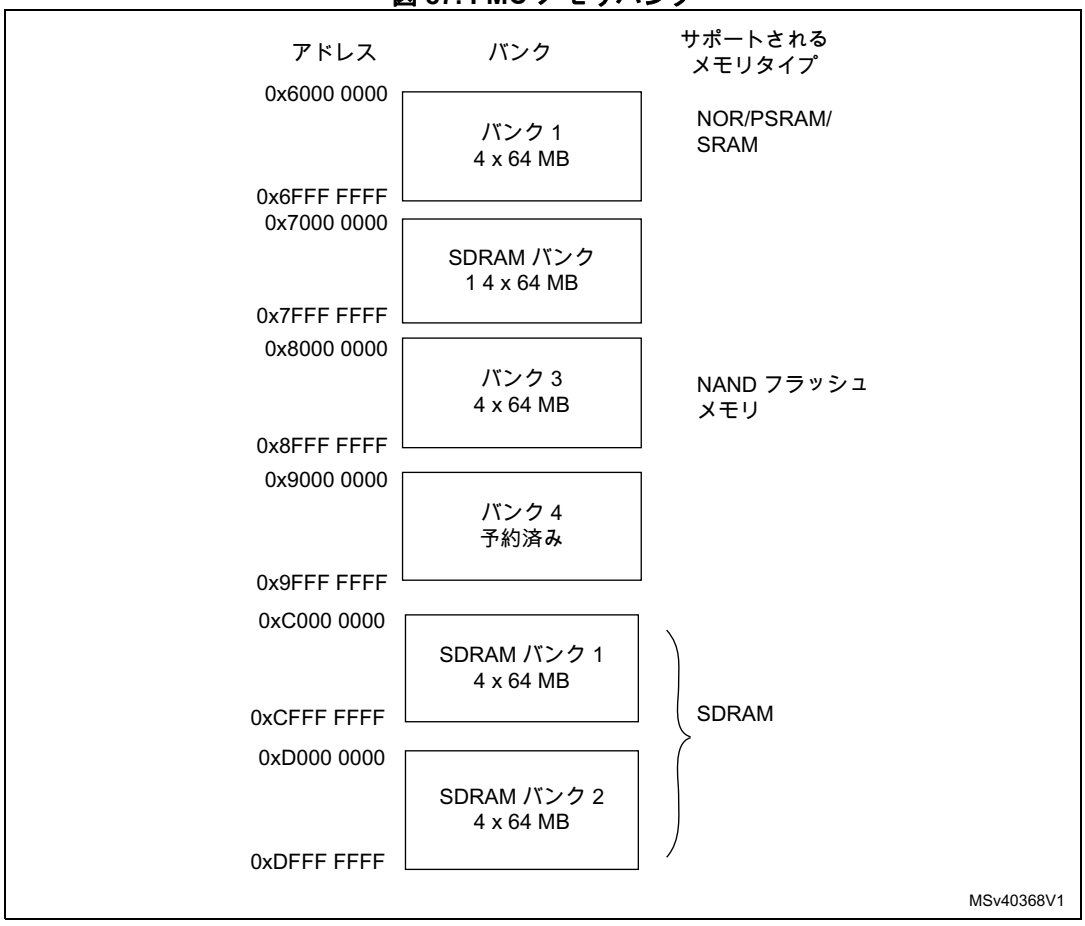
22.6 外部デバイスアドレスマッピング

FMC から見ると、外部メモリは、それぞれ 256 MB の固定サイズのバンクに分けられています（*図 87* を参照）。

- バンク 1 は、最大 4 つの NOR フラッシュメモリまたは PSRAM デバイスのアドレス指定に使用されます。このバンクは、4 つの専用チップセレクトを持つ 4 つの NOR/PSRAM サブバンクに分割されます。
  - バンク 1 - NOR/PSRAM 1
  - バンク 1 - NOR/PSRAM 2
  - バンク 1 - NOR/PSRAM 3
  - バンク 1 - NOR/PSRAM 4
- バンク 2 は、BMAP ビットの設定に応じて、SDRAM デバイス、SDRAM バンク 1、または SDRAM バンク 2 用として使用されます。
- バンク 3 は、NAND フラッシュメモリデバイスのアドレス指定に使用されます。この空間の MPU メモリ属性をソフトウェアによってデバイスに再設定する必要があります。
- バンク 4 と 5 は、SDRAM デバイスのアドレス指定に使用されます（バンクごとに 1 つのデバイス）。

各バンクで使用されるメモリのタイプは、ユーザアプリケーションから設定レジスタを使って設定することができます。

図 87. FMC メモリバンク



FMC バンクのマッピングは、FMC_BCR1 レジスタの BMAP[1:0] ビットで変更できます。*表 139* は、NOR/PSRAM バンクを SDRAM バンクとスワップする、または SDRAM バンク 2 を再マップするための設定を示します。このような設定により、異なる 2 つのアドレスマッピングで SDRAM バンクにアクセスできるようになります。

表 139. FMC バンクのマッピングオプション

開始 - 終了アドレス	BMAP[1:0] = 00（デフォルトマッピング）	BMAP[1:0] = 01 スワップされた NOR/PSRAM および SDRAM バンク	BMAP[1:0] = 10 再マップされた SDRAM バンク 2
0x6000 0000 - 0x6FFF FFFF	NOR/PSRAM バンク	SDRAM バンク 1	NOR/PSRAM バンク
0x7000 0000 - 0x7FFF FFFF	SDRAM バンク 1	SDRAM バンク 2	SDRAM バンク 2
0x8000 0000 - 0x8FFF FFFF	NAND バンク	NAND バンク	NAND バンク
0x9000 0000 - 0x9FFF FFFF	予約済み	予約済み	予約済み
0xC000 0000 - 0xCFFF FFFF	SDRAM バンク 1	NOR/PSRAM バンク	SDRAM バンク 1
0xD000 0000 - 0xDFFF FFFF	SDRAM バンク 2	SDRAM バンク 2	SDRAM バンク 2

22.6.1 NOR/PSRAM アドレスマッピング

ADDR[27:26] ビットは、*表 140*に示される 4 つのメモリバンクの 1 つを選択するために使用されます。

表 140. NOR/PSRAM バンク選択

ADDR[27:26](1)	選択されるバンク
00	バンク 1 - NOR/PSRAM 1
01	バンク 1 - NOR/PSRAM 2
10	バンク 1 - NOR/PSRAM 3
11	バンク 1 - NOR/PSRAM 4

1. ADDR は、外部メモリに変換される内部 AHB アドレスラインです。

ADDR[25:0] ビットは、外部メモリアドレスは相当します。ADDR はバイトアドレスですが、メモリはワードレベルでアドレス指定されるので、メモリに対して実際に発行されるアドレスは、次の表に示されるように、メモリのデータ幅に応じて変わります。

表 141. NOR/PSRAM 外部メモリアドレス

メモリ幅(1)	メモリに発行されるデータアドレス	最大メモリ容量（ビット）
8 ビット	ADDR[25:0]	64 Mバイト * 8 = 512 M ビット
16 ビット	ADDR[25:1] >> 1	64 Mバイト/2 * 16 = 512 M ビット
32 ビット	ADDR[25:2] >> 2	64 Mバイト/4 * 32 = 512 M ビット

1. 外部メモリが 16 ビット幅の場合、FMC は、内部で ADDR[25:1] を使用して外部メモリ FMC_A[24:0] のアドレスを生成します。メモリが 32 ビット幅の場合、FMC は、内部で ADDR[25:2] を使用して外部アドレスを生成します。
外部メモリのバス幅にかかわらず、FMC_A[0] は外部メモリアドレス A[0] に接続されます。

22.6.2 NAND フラッシュメモリアドレスマッピング

NAND バンクは、*表 142* に示されるメモリ領域に分割されます。

表 142. NAND メモリマッピングおよびタイミングレジスタ

開始アドレス	終了アドレス	FMC バンク	メモリ空間	タイミングレジスタ
0x8800 0000	0x8BFF FFFF	バンク 3 - NAND フラッシュ	属性	FMC_PATT (0x8C)
0x8000 0000	0x83FF FFFF		共通	FMC_PMEM (0x88)

NAND 型フラッシュメモリでは、共通および属性メモリ空間は、下位 256 KB に位置する 3 つのセクションに分割されます（*表 143* を参照）。

- データセクション（共通/属性メモリ空間の最初の 64 KB）
- コマンドセクション（共通/属性メモリ空間の 2 番目の 64 KB）
- アドレスセクション（共通/属性メモリ空間の次の 128 KB）

表 143. NAND バンク選択

セクション名	ADDR[17:16]	アドレス範囲
アドレスセクション	1X	0x020000～0x03FFFF
コマンドセクション	01	0x010000～0x01FFFF
データセクション	00	0x000000～0x0FFFF

アプリケーションソフトウェアは、3 つのセクションを使用して、NAND 型フラッシュメモリにアクセスします。

- **NAND フラッシュメモリにコマンドを送信するには、**ソフトウェアからコマンドセクションの任意のメモリ位置にコマンドの値を書き込む必要があります。
- **読み出しまたは書き込みを行う必要がある NAND フラッシュのアドレスを指定するには、**ソフトウェアは、アドレスセクションの任意のメモリ位置にアドレス値を書き込む必要があります。アドレスは 4 または 5 バイト長なので（実際のメモリサイズに依存）、アドレス全体を指定するには、アドレスセクションへのいくつかの連続した書き込みが必要です。
- **データの読み出しまたは書き込みを行うときは、**ソフトウェアは、データセクションの任意のメモリ位置からデータを読み出すか、書き込む必要があります。

NAND 型フラッシュメモリはアドレスを自動的にインクリメントするので、連続したメモリ位置にアクセスするには、データセクションのアドレスをインクリメントする必要はありません。

22.6.3 SDRAM アドレスマッピング

*表 144*に示すように、2 つ SDRAM バンクを使用できます。

表 144. SDRAM バンク選択

選択されるバンク	制御レジスタ	タイミングレジスタ
SDRAM バンク 1	FMC_SDCR1	FMC_SDTR1
SDRAM バンク 2	FMC_SDCR2	FMC_SDTR2

表 145 に、13 ビットRowおよび 11 ビットColumn構成に対する SDRAM マッピングを示します。

表 145. SDRAM アドレスマッピング

メモリ幅(1)	内部バンク	Rowアドレス	Columnアドレス(2)	最大メモリ容量（M バイト）
8 ビット	ADDR[25:24]	ADDR[23:11]	ADDR[10:0]	64 MB： 4 x 8K x 2K
16 ビット	ADDR[26:25]	ADDR[24:12]	ADDR[11:1]	128 MB： 4 x 8K x 2K x 2
32 ビット	ADDR[27:26]	ADDR[25:13]	ADDR[12:2]	256 MB： 4 x 8K x 2K x 4

1. 16 ビットメモリとインタフェースする場合、FMC は内部で ADDR[11:1] 内部アドレスラインを使用して外部アドレスを生成します。32 ビットメモリとインタフェースする場合、FMC は内部で ADDR[12:2] ラインを使用して外部アドレスを生成します。メモリの幅にかかわらず、FMC_A[0] は外部メモリアドレス A[0] に接続する必要があります。
2. 自動プリチャージはサポートされていません。FMC_A[10] は外部メモリアドレス A[10] に接続する必要がありますが、常にローレベルに駆動されます。

ADDR[27:0] ビットは、SDRAM コントローラの設定に応じて外部 SDRAM アドレスに変換されます。

- データサイズ：8、16 または 32 ビット
- Rowサイズ：11、12 または 13 ビット
- Columnサイズ：8、9、10 または 11 ビット
- 内部バンク数：2 つまたは 4 つの内部バンク

次の表に、SDRAM アドレスマッピングと SDRAM コントローラ設定の関係を示します。

表 146. 8 ビットデータバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（内部アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 ビット Rowサイズ設定	Res.							バンク [1:0]		ROW [10:0]											COLUMN [7:0]							
	Res.							バンク [1:0]		ROW [10:0]											COLUMN [8:0]							
	Res.							バンク [1:0]		ROW [10:0]											COLUMN [9:0]							
	Res.							バンク [1:0]		ROW [10:0]											COLUMN [10:0]							
12 ビット行サイズ設定	Res.							バンク [1:0]		ROW [11:0]											COLUMN [7:0]							
	Res.							バンク [1:0]		ROW [11:0]											COLUMN [8:0]							
	Res.							バンク [1:0]		ROW [11:0]											COLUMN [9:0]							
	Res.							バンク [1:0]		ROW [11:0]											COLUMN [10:0]							

表 146. 8 ビットデータバス幅での SDRAM アドレスマッピング(1)(2)（続き）

Rowサイズ設定	ADDR（内部アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
13 ビット行サイズ設定	Res.					バンク[1:0]		ROW [12:0]													COLUMN [7:0]						
	Res.					バンク[1:0]		ROW [12:0]													COLUMN [8:0]						
	Res.				バンク[1:0]			ROW [12:0]													COLUMN [9:0]						
	Res.			バンク[1:0]				ROW [12:0]													COLUMN [10:0]						

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済み（Res.）アドレスレンジにアクセスすると、AXI スレーブエラーが発生します。

表 147. 16 ビットデータバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 ビットRowサイズ設定	Res.						バンク[1:0]		ROW [10:0]											COLUMN [7:0]								BM0(3)
	Res.					バンク[1:0]			ROW [10:0]											COLUMN [8:0]								BM0
	Res.				バンク[1:0]				ROW [10:0]											COLUMN [9:0]								BM0
	Res.			バンク[1:0]					ROW [10:0]											COLUMN [10:0]								BM0
12 ビットRowサイズ設定	Res.					バンク[1:0]			ROW [11:0]											COLUMN [7:0]								BM0
	Res.				バンク[1:0]				ROW [11:0]											COLUMN [8:0]								BM0
	Res.			バンク[1:0]					ROW [11:0]											COLUMN [9:0]								BM0
	Res.		バンク[1:0]						ROW [11:0]											COLUMN [10:0]								BM0
13 ビットRowサイズ設定	Res.				バンク[1:0]				ROW [12:0]											COLUMN [7:0]								BM0
	Res.			バンク[1:0]					ROW [12:0]											COLUMN [8:0]								BM0
	Res.		バンク[1:0]						ROW [12:0]											COLUMN [9:0]								BM0
	Res.	バンク[1:0]							ROW [12:0]											COLUMN [10:0]								BM0

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済みの空間（Res.）にアクセスすると、AXI スレーブエラーが発生します。
3. BM0 は、16 ビットアクセスに対するバイトマスクです。

表 148. 32 ビットデータバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 ビットRow サイズ設定	Res.					バンク [1:0]		ROW [10:0]											COLUMN [7:0]								BM[1:0] (3)
	Res.					バンク [1:0]		ROW [10:0]											COLUMN [8:0]								BM[1:0
	Res.				バンク [1:0]			ROW [10:0]											COLUMN [9:0]								BM[1:0
	Res.			バンク [1:0]				ROW [10:0]											COLUMN [10:0]								BM[1:0
12 ビットRow サイズ設定	Res.					バンク [1:0]		ROW [11:0]											COLUMN [7:0]								BM[1:0
	Res.				バンク [1:0]			ROW [11:0]											COLUMN [8:0]								BM[1:0
	Res.			バンク [1:0]				ROW [11:0]											COLUMN [9:0]								BM[1:0
	Res.		バンク [1:0]					ROW [11:0]											COLUMN [10:0]								BM[1:0
13 ビットRow サイズ設定	Res.				バンク [1:0]			ROW [12:0]											COLUMN [7:0]								BM[1:0
	Res.			バンク [1:0]				ROW [12:0]											COLUMN [8:0]								BM[1:0
	Res.		バンク [1:0]					ROW [12:0]											COLUMN [9:0]								BM[1:0
	バンク [1:0]							ROW [12:0]											COLUMN [10:0]								BM[1:0

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済みの空間（Res.）にアクセスすると、AXI スレーブエラーが発生します。
3. BM[1:0] は、32 ビットアクセスに対するバイトマスクです。

22.7 NOR 型フラッシュ／PSRAM コントローラ

FMC は、以下のタイプのメモリを駆動するのに適した信号タイミングを発生します。

- 非同期 SRAM および ROM
  - 8 ビット
  - 16 ビット
  - 32 ビット
- PSRAM（Cellular RAM）
  - 非同期モード
  - CRAM 1.5. のページ境界を跨ぐ場合にバーストアクセスを分割するための設定可能なオプション付きの同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス
- NOR フラッシュメモリ
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス

FMC は、バンクごとに 1 つのチップセレクト信号 NE[4:1] を出力します。その他の信号（アドレス、データ、および制御）はすべて、共有されます。

FMC は、次のようなプログラム可能なタイミングにより、広範囲なデバイスをサポートしています。

- プログラム可能なウェイトステート（15 まで）
- プログラム可能なバスターンアラウンドサイクル（15 まで）
- プログラム可能な出力イネーブルおよび書き込みイネーブル遅延（15 まで）
- 独立した読み出しおよび書き込みタイミングとプロトコルにより、広範囲なメモリおよびタイミングをサポート
- プログラム可能な連続クロック（FMC_CLK）出力

FMC 出力クロック（FMC_CLK）は fmc_ker_ck クロックの約数です。このクロックは、同期アクセス中にのみ、または FMC_BCR1 レジスタの CCKEN ビット設定に応じて非同期アクセスおよび同期アクセス中に、選択した外部デバイスへ供給することができます。

- CCLKEN ビットがリセットされている場合、FMC は同期アクセス（読み出し／書き込みトランザクション）中にのみクロック（FMC_CLK）を生成します。
- CCLKEN ビットがセットされている場合、FMC は非同期アクセス中および同期アクセス中に連続クロックを生成します。FMC_CLK 連続クロックを生成するためには、バンク 1 を同期モードに設定する必要があります（*セクション 22.7.6 : NOR/PSRAM コントローラレジスタ*を参照）。すべての同期メモリに対して同じクロックが使用されるため、連続出力クロックを生成して同期アクセスを実行する場合、AXI データサイズはメモリデータ幅（MWID）と同じにする必要があります。そうしないと、AXI データトランザクションに応じて、FMC_CLK 周波数が変化します（FMC_CLK 分周比の式については、*セクション 22.7.5: 同期トランザクション* を参照）。

各バンクのサイズは 64 MB に固定されます。各バンクは、専用レジスタによって設定されます（*セクション 22.7.6 : NOR/PSRAM コントローラレジスタ*を参照）。

プログラム可能なメモリパラメータは、アクセスタイミング（*表 149*を参照）を含み、ウェイト管理をサポートします（バーストモードでの PSRAM および NOR フラッシュメモリへのアクセス）。

表 149. プログラム可能な NOR/PSRAM のアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	FMC クロックサイクル（fmc_ker_ck）	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクス I/O	FMC クロックサイクル（fmc_ker_ck）	1	15
データセットアップ	データセットアップフェーズ時間	非同期	FMC クロックサイクル（fmc_ker_ck）	1	256
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読み出し	FMC クロックサイクル（fmc_ker_ck）	0	15
クロック分周比	1 メモリクロックサイクル（CLK）を構築するために必要な FMC クロックサイクル（fmc_ker_ck）数	同期	FMC クロックサイクル（fmc_ker_ck）	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル（fmc_ker_ck）	2	17

22.7.1 外部メモリインタフェース信号

表 150、*表 151* および *表 152* に、NOR フラッシュメモリ、SRAM、および PSRAM へのインタフェースで一般に使用される信号を示します。

注： ***接頭辞「N」は、アクティブローの信号を表します。***

NOR フラッシュメモリ、非マルチプレクス I/O

表 150. 非マルチプレクス I/O NOR フラッシュメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:0]	O	アドレスバス
D[31:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR フラッシュウェイト入力信号

最大容量は、512 M ビット（26 アドレスライン）です。

NOR フラッシュメモリ、16 ビットマルチプレクス I/O

表 151. 16 ビットマルチプレクス I/O NOR フラッシュメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス （16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスで マルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR フラッシュウェイト入力信号

最大容量は 512 M ビットです。

PSRAM/SRAM、非マルチプレクス I/O

表 152. 非マルチプレクス I/O PSRAM/SRAM

FMC 信号名	I/O	機能
CLK	O	クロック （PSRAM 同期アクセス専用）
A[25:0]	O	アドレスバス
D[31:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	PSRAM 入力の場合のみアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[3:0]	O	バイトレーン出力。バイト 0～3 制御（上位および下位バイトイネーブル）

最大容量は 512 M ビットです。

PSRAM、16 ビットマルチプレクス I/O

表 153. 16 ビットマルチプレクス I/O PSRAM

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス（16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	アドレス有効 PSRAM 入力（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は、512 M ビット（26 アドレスライン）です。

22.7.2 サポートされるメモリおよびトランザクション

以下の*表 154* に、NOR フラッシュメモリ、PSRAM、SRAM に対してメモリデータバスが 16 ビット幅の場合にサポートされているデバイス、アクセスモード、トランザクションの例を示します。FMC では許容されない（すなわち非サポート）トランザクションは、この例で灰色表示しています。

表 154. NOR フラッシュ /PSRAM：サポートされているメモリとトランザクションの例(1)

デバイス	モード	読み出し／書き込み	AXI データサイズ	メモリデータサイズ	許可/禁止	コメント
NOR 型フラッシュ（マルチプレクス I/O と非マルチプレクス I/O）	非同期	R	8	16	Y	
	非同期	W	8	16	N	
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32/64	16	Y	
PSRAM（マルチプレクス I/O および非マルチプレクス I/O）	非同期	R	8	16	Y	
	非同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32/64	16	Y	
	同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	同期	W	16/32/64	16	Y	

表 154. NOR フラッシュ /PSRAM：サポートされているメモリとトランザクションの例(1)

デバイス	モード	読み出し／書き込み	AXI データサイズ	メモリデータサイズ	許可/禁止	コメント
SRAM と ROM	非同期	R	8/16	16	Y	
	非同期	W	8/16	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用

1. NBL[1:0] は AXI 書き込みストローブによっても駆動されます。

22.7.3 一般的なタイミング規則

信号同期は次のように行われます。

- すべてのコントローラ出力信号は、fmc_ker_ck クロックの立ち上がりエッジで変化します。
- 同期読み出し／書き込みモードでは、すべての出力信号が fmc_ker_ck クロックの立ち上がりエッジで変化します。CLKDIV の値によらず、すべての信号は次のように変化します。
  - NOEL/NWEL/ NEL/NADVL/ NADVH /NBLL/ アドレスの有効出力は、FMC_CLK クロックの立ち下がりエッジで変化します。
  - NOEH/ NWEH / NEH/ NOEH/NBLH/ アドレスの無効出力は、FMC_CLK クロックの立ち上がりエッジで変化します。

22.7.4 NOR フラッシュ /PSRAM コントローラ非同期トランザクション

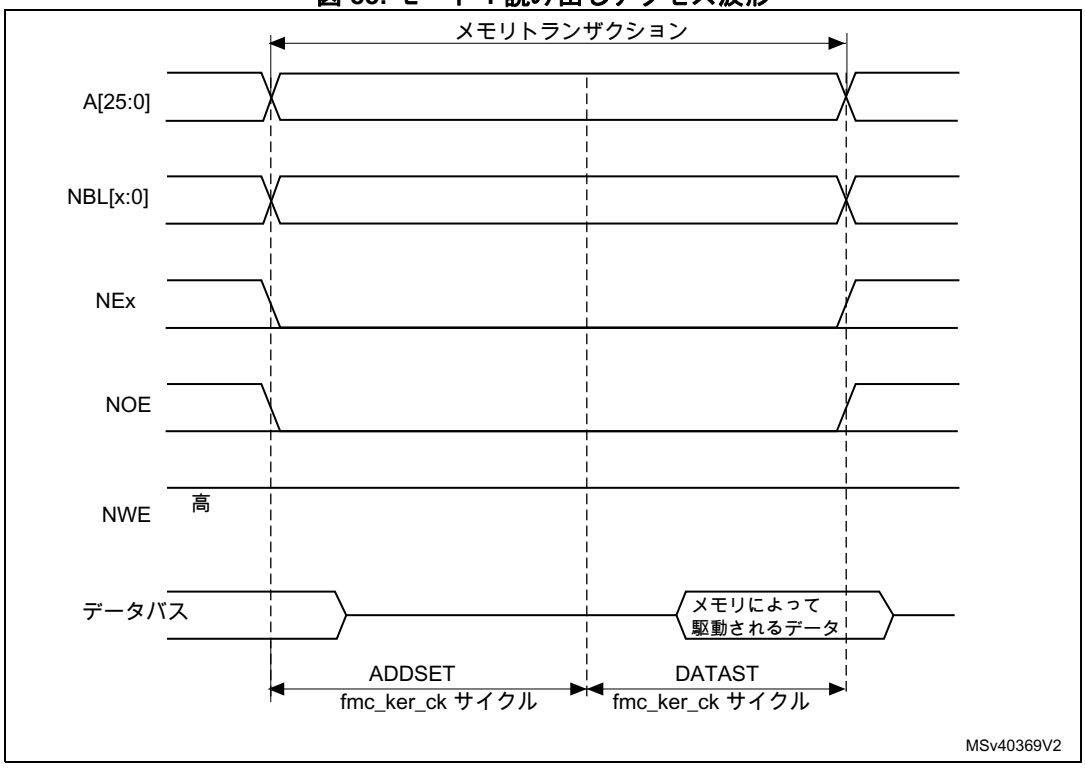
スタティックメモリの非同期トランザクション（NOR 型フラッシュメモリ、PSRAM、SRAM）は以下のように行われます。

- 信号は内部クロックによって同期されます。このクロックはメモリには発行されません。
- FMC は、チップセレクト信号をデアサートする前に、常にデータをサンプリングします。これにより、メモリのデータホールドタイミングに関する制約が満たされます（チップイネーブルハイからデータ遷移までの最小時間は通常 0 ns です）。
- 拡張モードを有効にした場合（FMC_BCRx レジスタの EXTMOD ビットをセット）、最大 4 つの拡張モード（A、B、C、D）が使用できます。書き込み動作および読み出し動作で、A、B、C、D モードをミックスすることができます。例えば、読み出し動作をモード A で実行し、書き込み動作をモード B で実行することができます。
- 拡張モードを無効にした場合（FMC_BCRx レジスタで EXTMOD ビットをリセット）、FMC は次のようにモード 1 またはモード 2 で動作することができます。
  - SRAM/PSRAM メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x0 または 0x01）、モード 1 がデフォルトモードです。
  - NOR メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x10）、モード 2 がデフォルトモードです。

モード 1 - SRAM/PSRAM （CRAM）

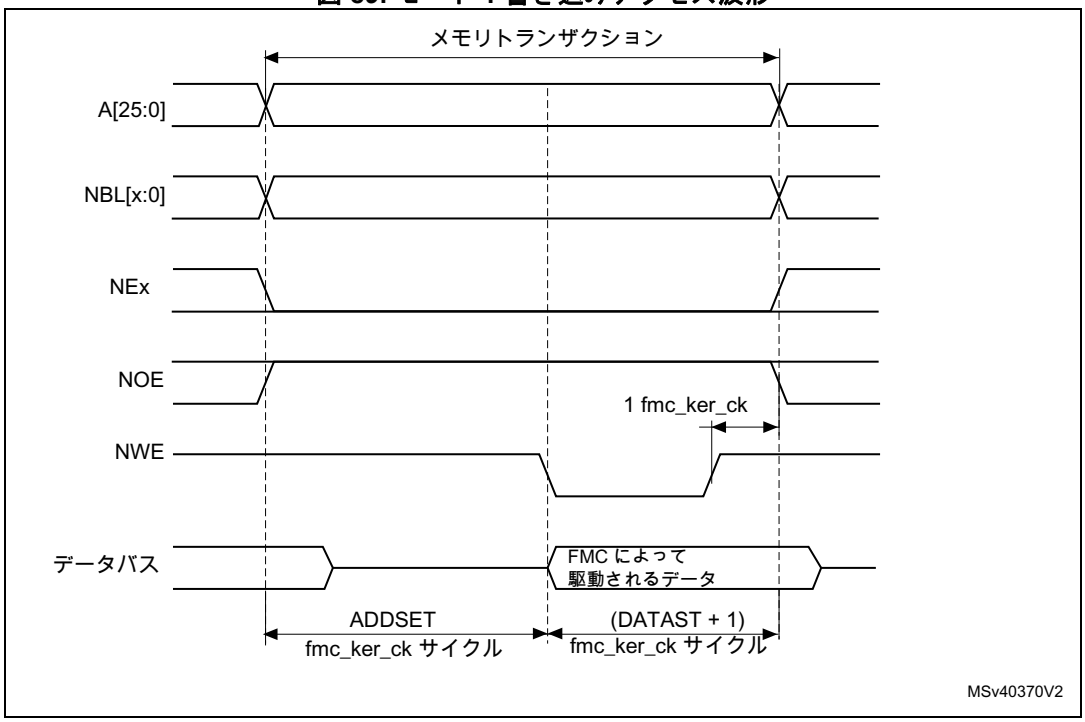
次の図に、サポートされているモードに対する読み出しトランザクションと書き込みトランザクションを示します。それに続いて、FMC _BCRx レジスタ、および FMC_BTRx/FMC_BWTRx レジスタの必要とされる設定を示します。

図 88. モード 1 読み出しアクセス波形



参考資料

図 89. モード 1 書き込みアクセス波形



書き込みトランザクションの最後の fmc_ker_ck サイクルは、NWE の立ち上がりエッジ後のアドレスおよびデータホールド時間の保証に役立ちます。この fmc_ker_ck サイクルがあるため、DATAST の値はゼロより大きくなければなりません（DATAST ＞ 0）。

表 155. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 155. FMC_BCRx ビットフィールド（続き）

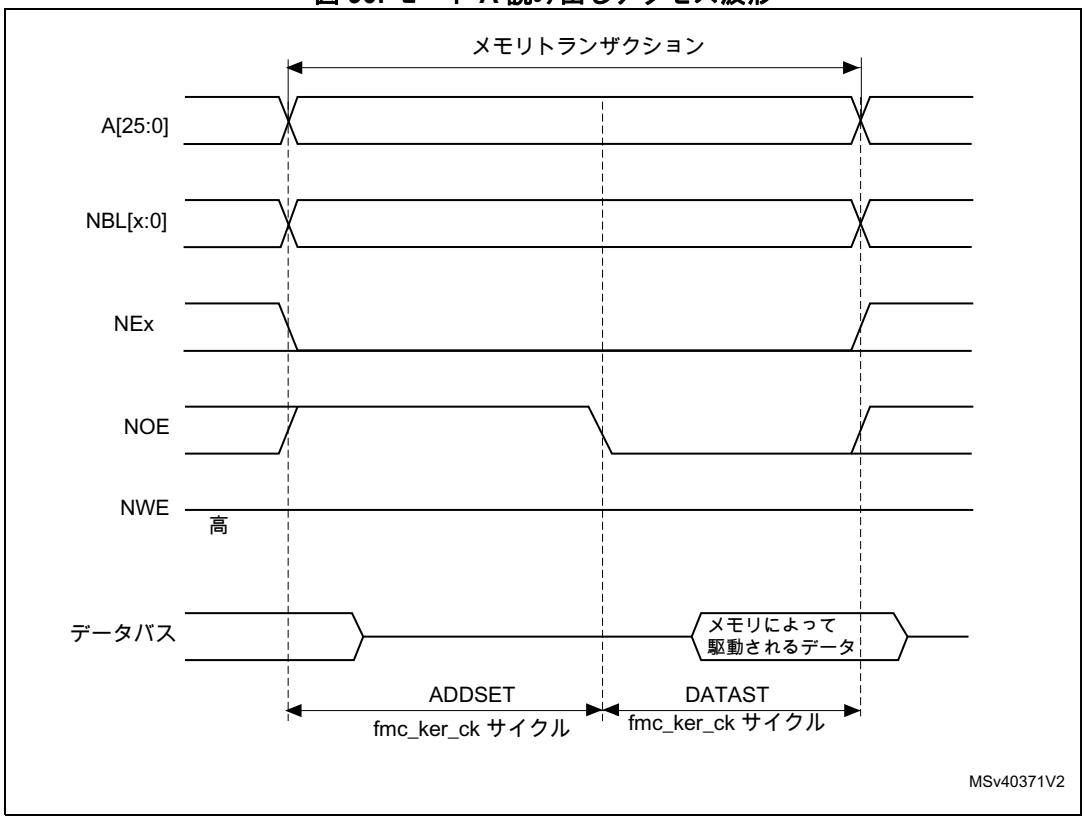
ビット番号	ビット名	設定値
7	予約済み	0x1
6	FACCEN	無視
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて、0x2（NOR フラッシュメモリ）を除きます。
1	MUXE	0x0
0	MBKEN	0x1

表 156. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	無視
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	2 番目のアクセスフェーズの時間（書き込みアクセスの場合は DATAST + 1 fmc_ker_ck サイクル、読み出しアクセスの場合は DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）。 ADDSET の最小値は 0。

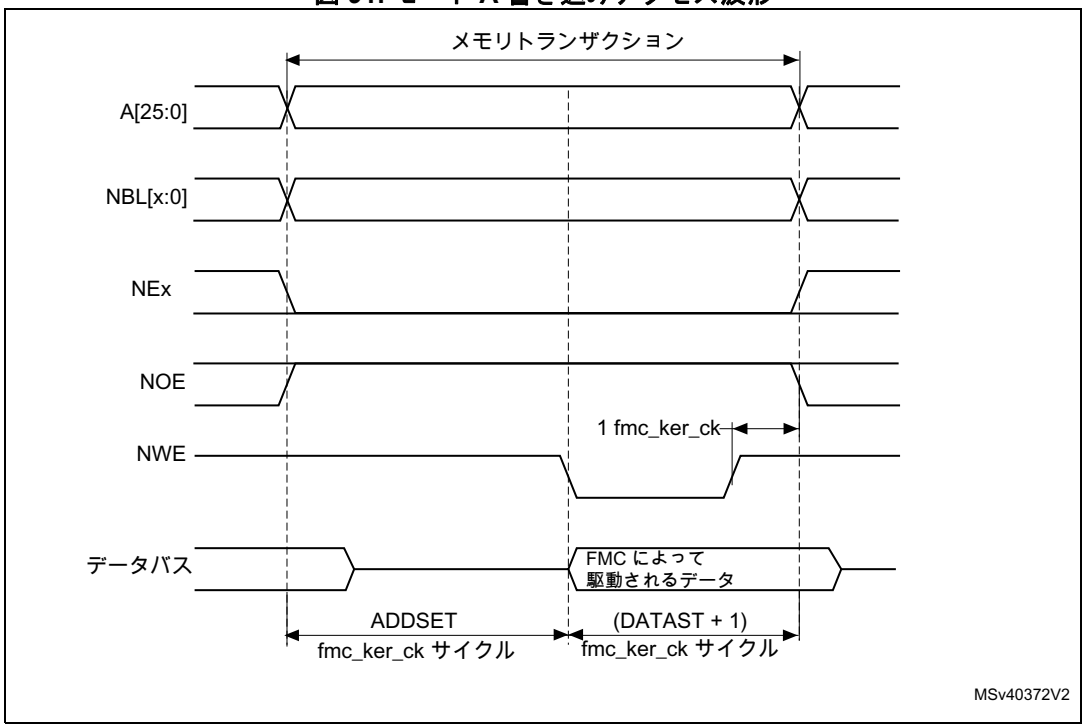
モード A - SRAM/PSRAM（CRAM）OE トグル

図 90. モード A 読み出しアクセス波形



1. NBL[3:0] は読み出しアクセス中ローレベルに駆動されます。

図 91. モード A 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読み出しおよび書き込みのタイミングです。

表 157. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 157. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
7	予約済み	0x1
6	FACCEN	無視
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて、0x2（NOR フラッシュメモリ）を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

表 158. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル） ADDSET の最小値は 0。

表 159. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル） ADDSET の最小値は 0。

モード 2/B - NOR 型フラッシュ

図 92. モード 2 およびモード B 読み出しアクセス波形

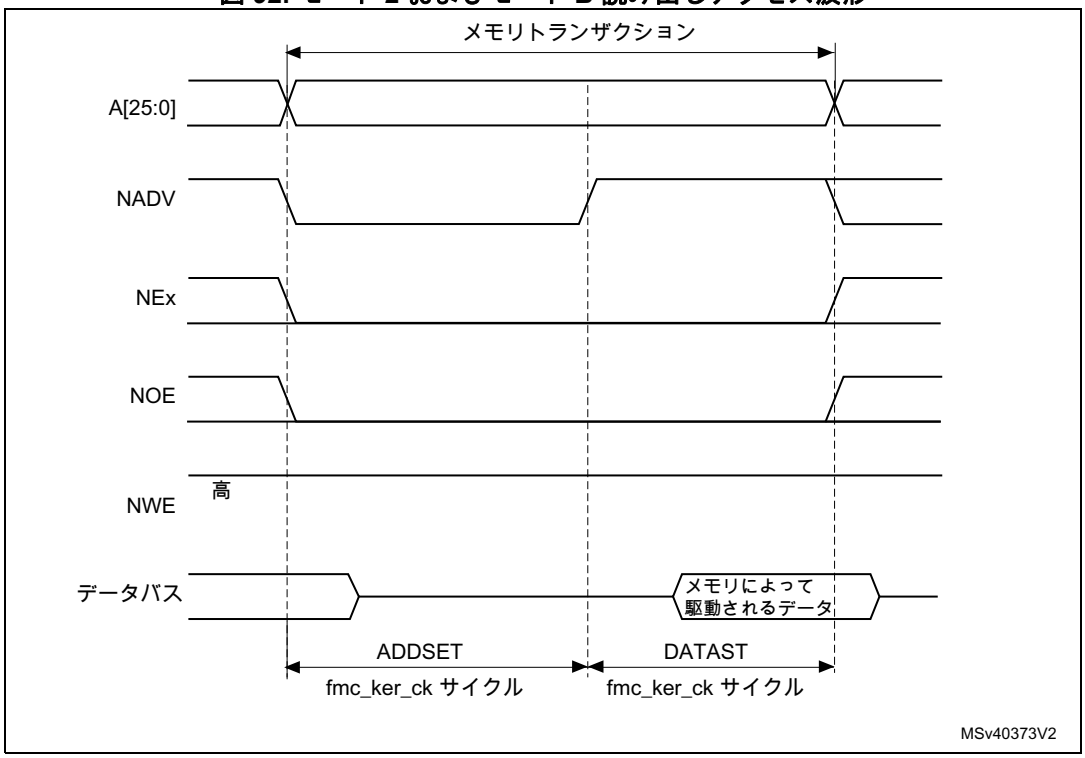


図 93. モード 2 書き込みアクセス波形

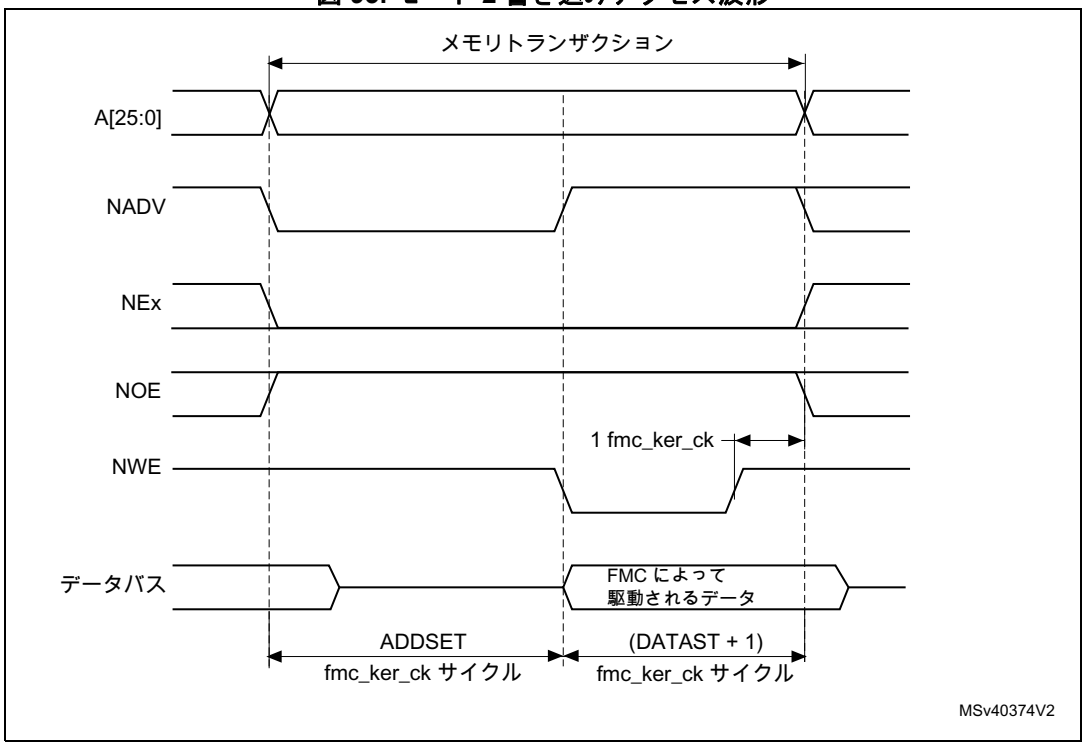
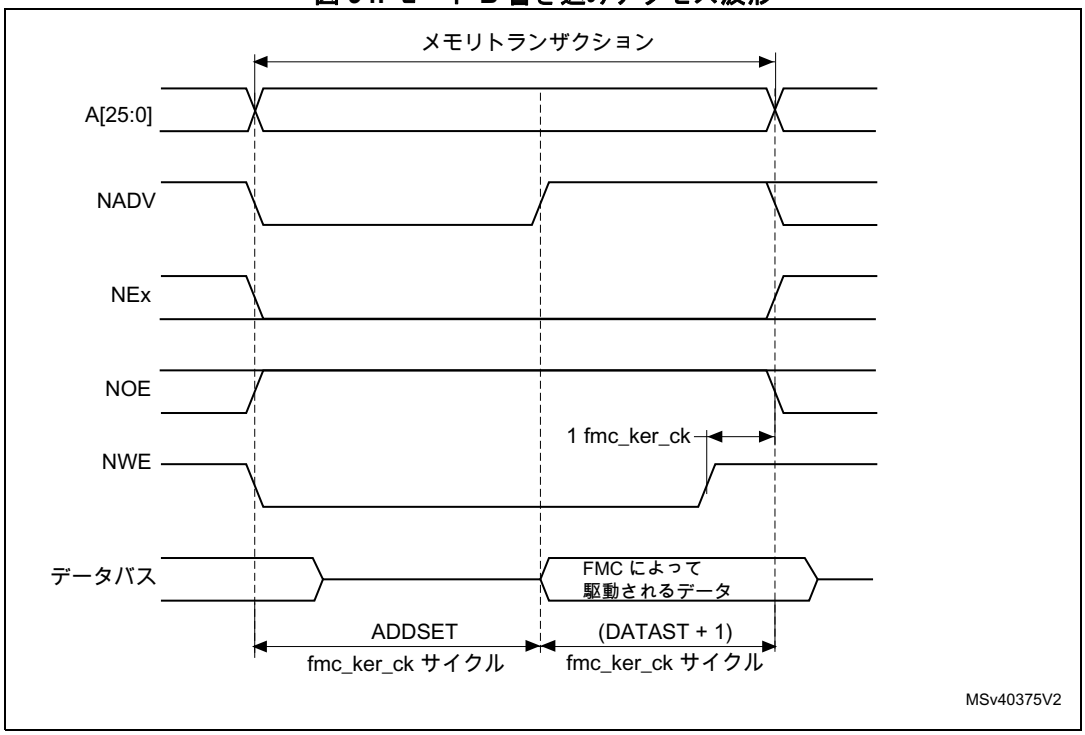


図 94. モード B 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NWE のトグリングと拡張モードがセットされたとき（モード B）の独立した読み出しおよび書き込みのタイミングです。

表 160. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 160. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x2 （NOR 型フラッシュメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 161. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31-30	予約済み	0x0
29-28	ACCMOD	拡張モードがセットされている場合は 0x1
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

表 162. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31-30	予約済み	0x0
29-28	ACCMOD	拡張モードがセットされている場合は 0x1
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

注： ***FMC_BWTRx レジスタは、拡張モードがセットされている場合（モード B）のみ有効です。それ以外の場合には、その内容は意味を持ちません。***

モード C - NOR 型フラッシュ - OE トグリング

図 95. モード C 読み出しアクセス波形

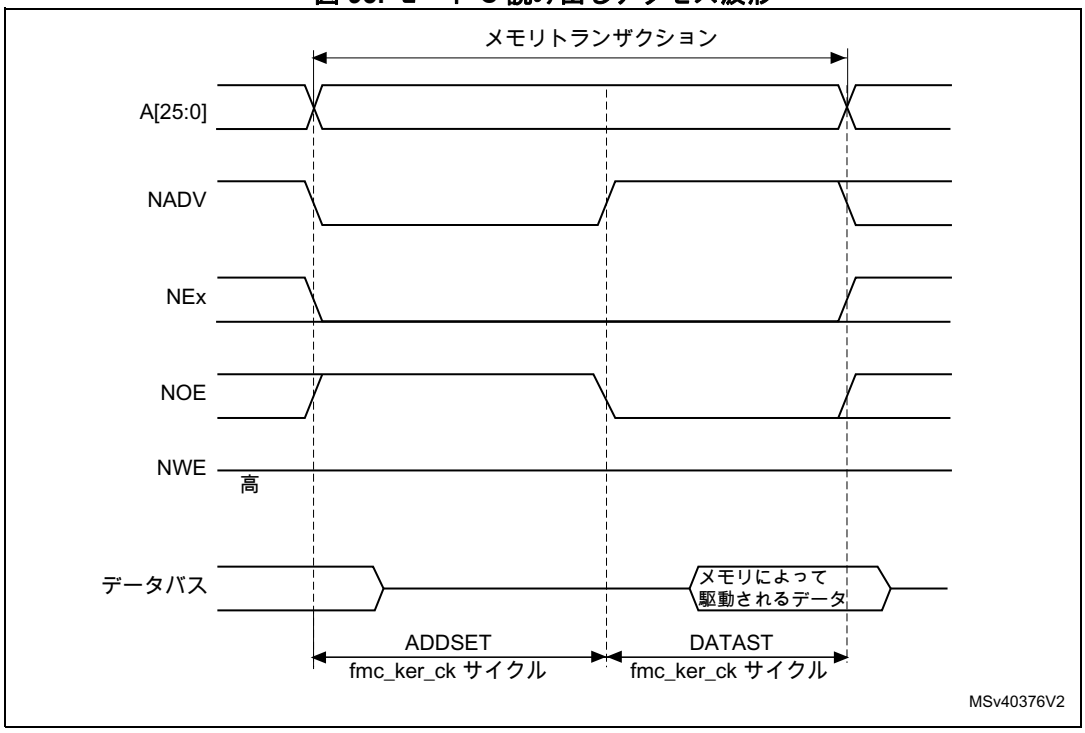
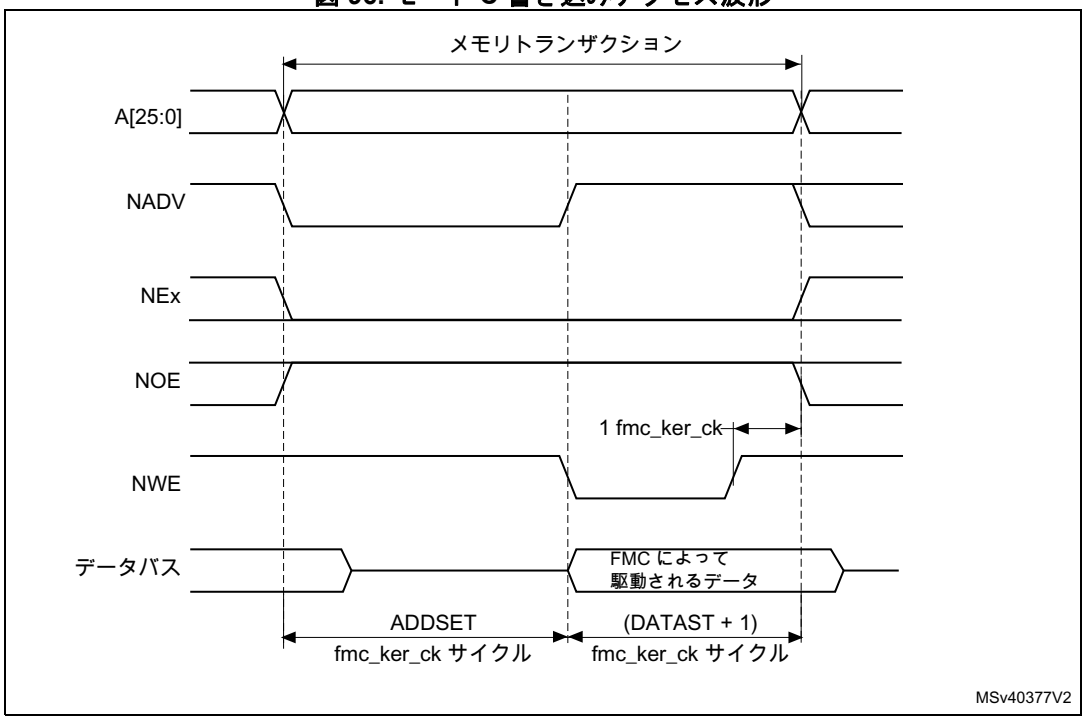


図 96. モード C 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読み出しおよび書き込みのタイミングです。

表 163. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x02 （NOR 型フラッシュメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

参考資料

表 164. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x2
27-24	DATLAT	0x0
23-20	CLKDIV	0x0
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

表 165. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x2
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7-4	ADDHLD	無視
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

モード D - 拡張アドレスによる非同期アクセス

図 97. モード D 読み出しアクセス波形

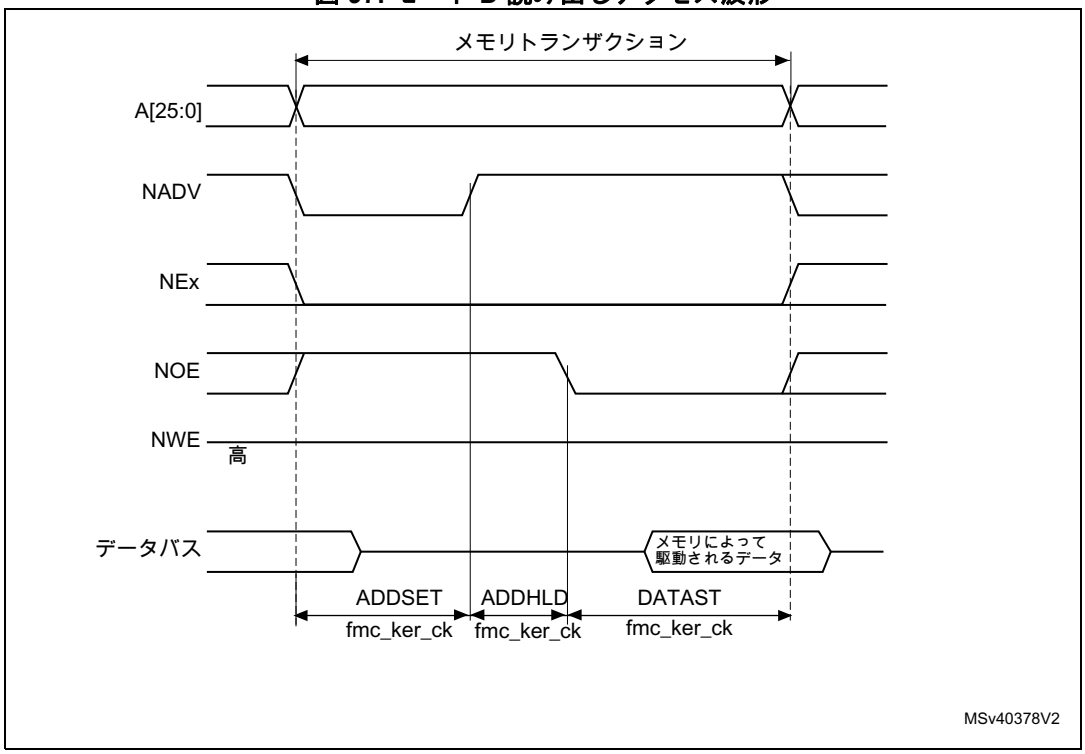
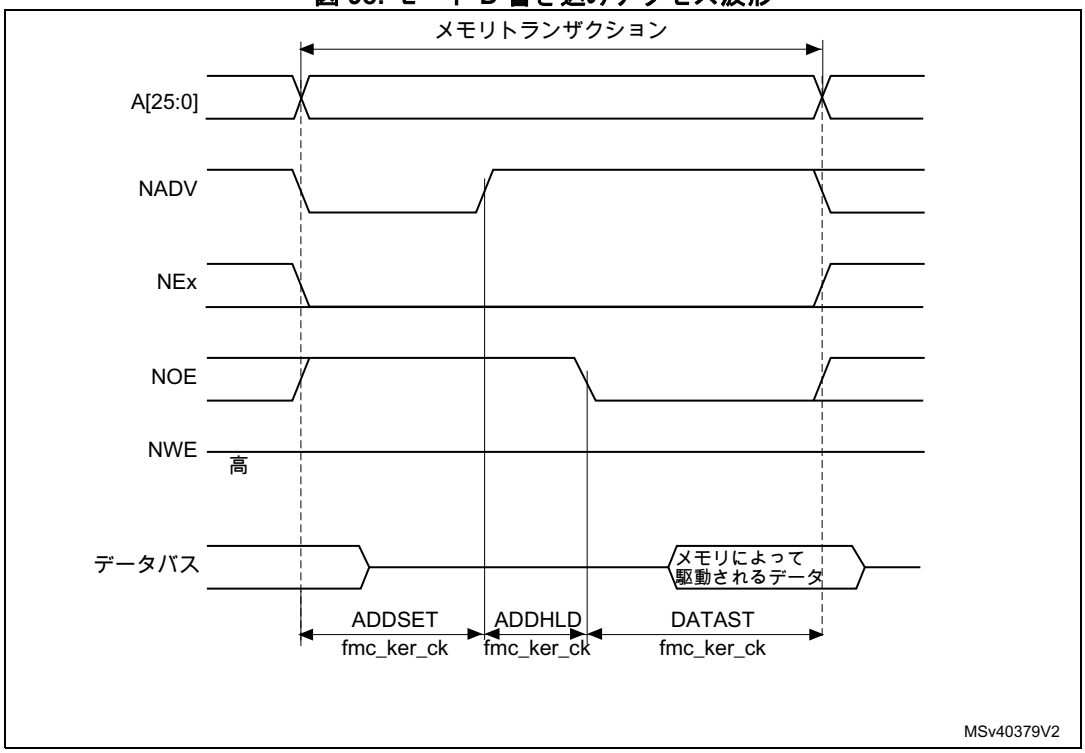


図 98. モード D 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NADV が変化した後にトグルを続ける NOE のトグル と独立した読み出し書き込みタイミングです。

表 166. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて設定します。
1	MUXEN	0x0
0	MBKEN	0x1

表 167. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x3
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）

表 167. FMC_BTRx ビットフィールド（続き）

ビット番号	ビット名	設定値
7-4	ADDHLD	読み出しアクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 1。

表 168. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x3
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST + 1 fmc_ker_ck サイクル）
7-4	ADDHLD	書き込みアクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 1。

マルチプレクスモード - NOR 型フラッシュメモリに対するマルチプレクス非同期アクセス

図 99. マルチプレクス読み出しアクセス波形

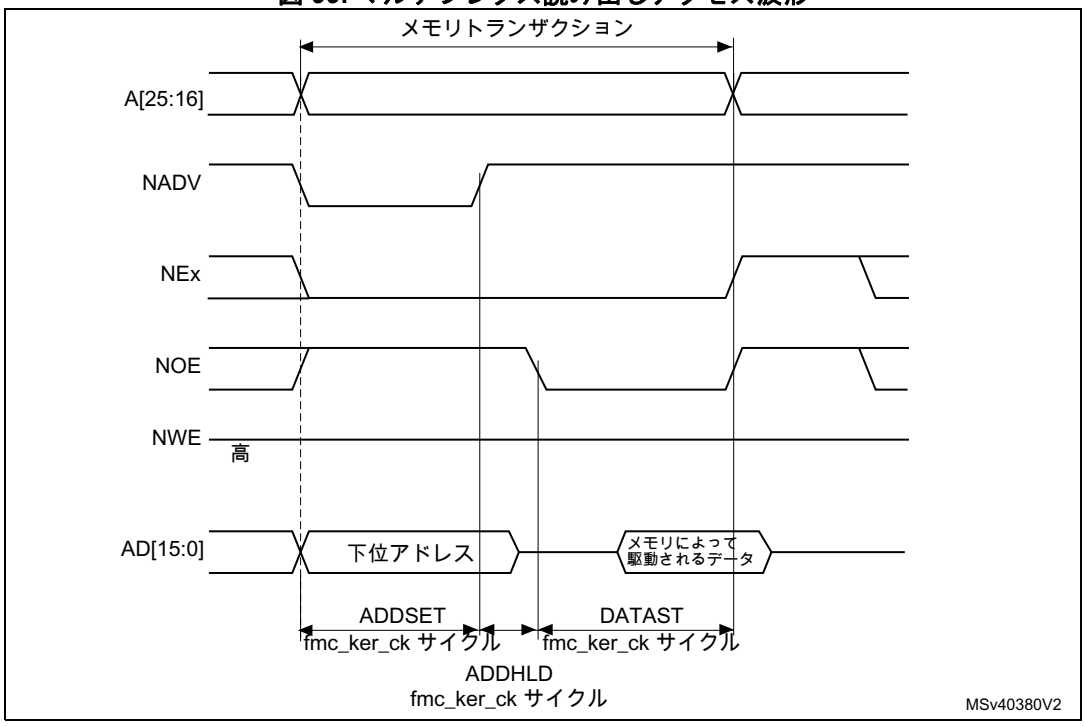
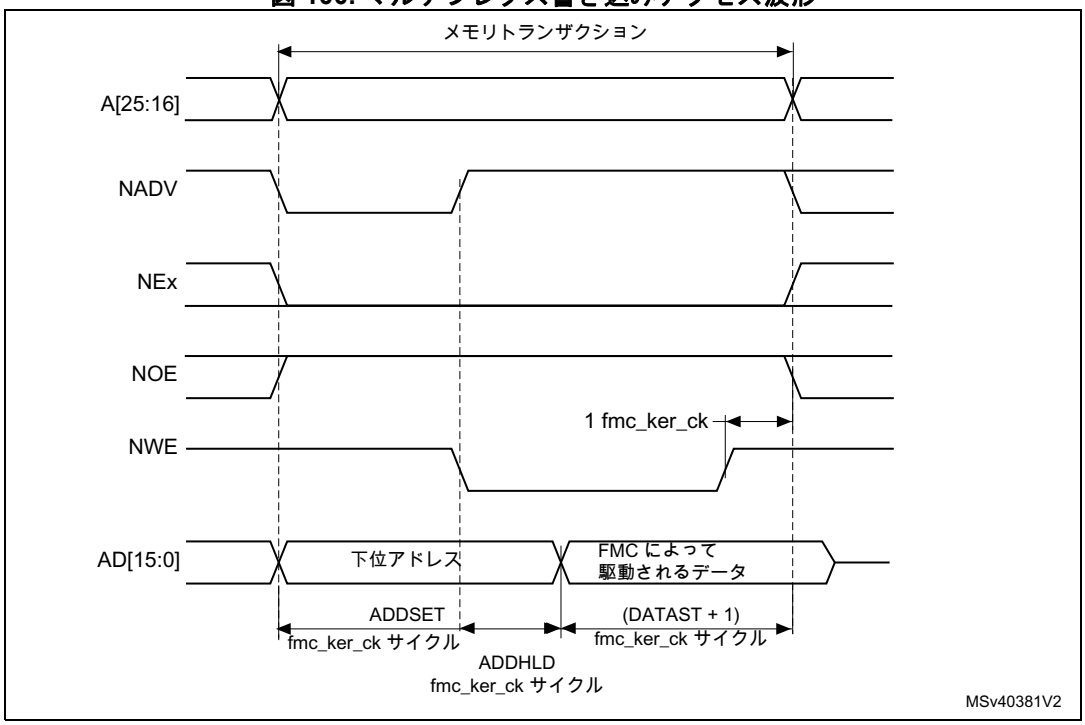


図 100. マルチプレクス書き込みアクセス波形



モード D との相違点は、データバス上の下位アドレスバイトのドライブです。

参考資料

表 169. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。 そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x2 （NOR 型フラッシュメモリ）
1	MUXEN	0x1
0	MBKEN	0x1

表 170. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	2 番目のアクセスフェーズの時間（読み出しアクセスの場合は DATAST fmc_ker_ck サイクル、書き込みアクセスの場合は DATAST+1 fmc_ker_ck サイクル）
7-4	ADDHLD	アクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3-0	ADDSET	最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）。ADDSET の最小値は 1。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを知らせるために、WAIT 信号をアサートした場合、FMC_BCRx レジスタの ASYNCWAIT ビットをセットする必要があります。

WAIT 信号がアクティブな場合（WAITPOL ビットに依存して、ハイまたはロー）、DATAST ビットによってプログラムされる 2 番目のアクセスフェーズ（データセットアップフェーズ）は、WAIT ビットが非アクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSET およびADDHLD ビットによってプログラムされる最初のアクセスフェーズ（アドレスセットアップおよびアドレスホールドフェーズ）は、WAIT に対応しないので、延長されません。

データセットアップフェーズは、メモリトランザクションが終了する 4 fmc_ker_ck サイクル前に WAIT が検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルする NOE/NWE に合わせた WAIT 信号をアサートする場合。

$$\text{DATAST} \geq (4 \times \text{FMC_CLK}) + \text{max_wait_assertion_time}$$

2. メモリが、NEx（またはトグルしない NOE/NWE）に合わせた WAIT 信号をアサートする場合。
   次の場合

$$\text{max_wait_assertion_time} > \text{address_phase} + \text{hold_phase}$$

   以下の通りです。

$$\text{DATAST} \geq (4 \times \text{FMC_CLK}) + (\text{max_wait_assertion_time} - \text{address_phase} - \text{hold_phase})$$

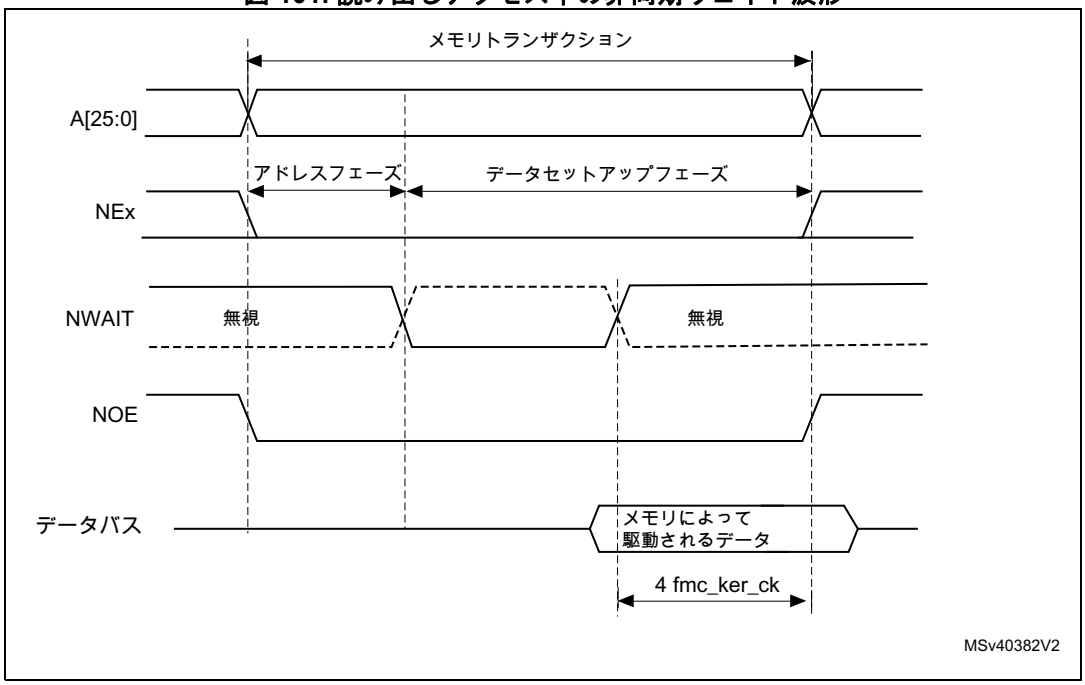
   それ以外の場合は、

$$\text{DATAST} \geq (4 \times \text{FMC_CLK})$$

   ここで、max_wait_assertion_time は、NEx/NOE/NWE がローになったときにメモリが WAIT 信号をアサートするために必要な最大時間です。

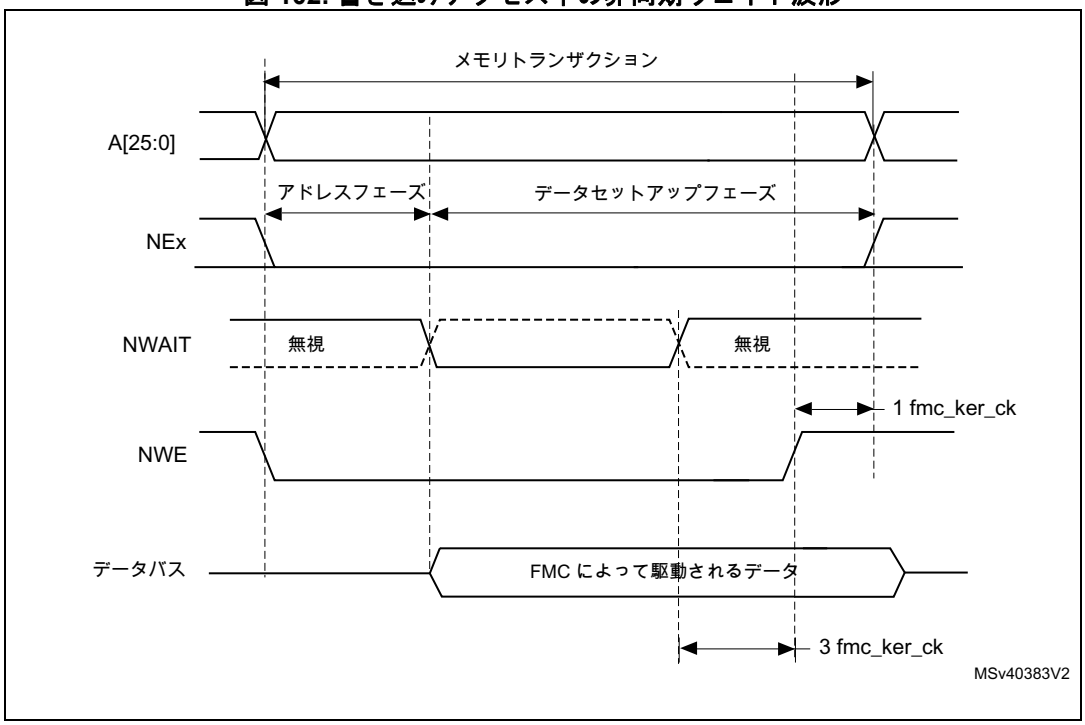
図 101 また、*図 102* に、WAIT が非同期メモリによって解除されたあとにメモリアクセスフェーズに追加される fmc_ker_ck クロックのサイクル数を示します（上記のケースに無関係）。

図 101. 読み出しアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

図 102. 書き込みアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

22.7.5 同期トランザクション

メモリクロック FMC_CLK は fmc_ker_ck の約数です。この値は CLKDIV 値と MWID/ AXI データサイズに依存し、次式で与えられます。

$$\text{FMC_CLK divider ratio} = \max(\text{CLKDIV} + 1, \text{MWID}(\text{AXI data size}))$$

MWID が 16 ビットまたは 8 ビットの場合、FMC_CLK の分周比は常にプログラムされた CLKDIV 値で決まります。

MWID が 32 ビットの場合、FMC_CLK の分周比は AXI データサイズにも依存します。

例：

- CLKDIV = 1、MWID = 32 ビット、AXI データサイズ = 8 ビットの場合、FMC_CLK = fmc_ker_ck / 4
- CLKDIV = 1、MWID = 16 ビット、AXI データサイズ = 8 ビットの場合、FMC_CLK = fmc_ker_ck / 2

NOR 型フラッシュメモリは、NADV のアサーションから FMC_CLK がハイになるまでの最小時間を指定します。この制約を満たすために、FMC は、同期アクセスの最初の内部クロックサイクルでは（NADV アサーションの前）、クロックをメモリに供給しません。これにより、メモリクロックの立ち上がりエッジは、NADV ローパルスの中間で発生します。

BCR レジスタの書き込み中に、同期モードに設定しなければならない一部の PSRAM メモリについては、メモリの属性空間を Device または Strongly-ordered に設定する必要があります。一度 PSRAM BCR レジスタを設定すると、PSRAM アドレス空間のメモリ属性をキャッシュ可能なようにプログラムすることができます。

データ遅延と NOR メモリ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLAT の値は、NOR 型フラッシュ設定レジスタで指定された遅延の値と一致しなければなりません。FMC は、NADV がローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： ***一部の NOR フラッシュメモリは、データ遅延カウントに NADV ローサイクルを含めるので、NOR フラッシュの遅延と FMC DATALAT パラメータの厳密な関係は、次のいずれかです。***

- NOR 型フラッシュ遅延 = (DATLAT + 2) FMC_CLK クロックサイクル、または
- NOR 型フラッシュ遅延 = (DATLAT + 3) FMC_CLK クロックサイクル

最近のメモリの中には、遅延フェーズにおいて NWAIT をアサートするものがあります。そのような場合には、DATLAT を最小値に設定することができます。結果として、FMC はデータをサンプリングして、データが有効であるかどうか評価するのに十分な時間、ウェイトします。このように、FMC は、メモリが遅延から抜けたことを検出し、実際のデータが処理されます。

その他のメモリは、遅延中には NWAIT をアサートしません。この場合、FMC とメモリの両方について、遅延が正しく設定されなければなりません。そうしないと、正しいデータの代わりに無効なデータが取られたり、有効なデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードで設定される場合、例えば、16 ビットメモリでシングルバーストトランザクションがリクエストされると、FMC は長さ 1（AXI 転送が16 ビットの場合）または 2（AXI 転送が 32 ビットの場合）のバーストトランザクションを実行して、最後のデータがストローブされると、チップセレクト信号をデアサートします。

このような転送はサイクル数の点で、非同期読み出し動作に比較して最も効率的ではありません。しかし、時間がかかるメモリアクセスモードの再プログラムには、まず、ランダム非同期アクセスが必要です。

Cellular RAM 1.5 ページ境界跨ぎ

Cellular RAM 1.5 では、ページ境界を跨ぐバーストアクセスは許可されていません。メモリページサイズに従って FMC_BCR1 レジスタの CPSIZE ビットを設定することで、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割できます。

ウェイト管理

同期 NOR フラッシュメモリでは、NWAIT がプログラムされた遅延周期、すなわち（DATLAT+2）FMC_CLK サイクルの後で評価されます。

NWAIT がアクティブの場合（WAITPOL = 0 ではローレベル、WAITPOL = 1 ではハイレベル）、ウェイトステートはNWAIT が非アクティブ（WAITPOL = 0 ではハイレベル、WAITPOL = 1ではローレベル）になるまでが挿入されます。

NWAIT が非アクティブのときには、データはただちに（ビット WAITCFG=1）または次のクロックエッジで（ビット WAITCFG=0）、有効とみなされます。

NWAIT 信号によるウェイトステートの挿入時には、コントローラはメモリへのクロックパルスの送信を続け、チップセレクトと出力イネーブル信号を維持します。データは有効と見なしません。

バーストモードでの NOR フラッシュの NWAIT 信号には、2 つのタイミング設定があります。

- フラッシュメモリは、ウェイトステートの 1 データサイクル前に NWAIT 信号をアサートします（リセット後のデフォルト）。
- フラッシュメモリは、ウェイトステートの間に NWAIT 信号をアサートします。

FMC は FMC_BCRx レジスタ（x = 0..3）の WAITCFG ビットを使って、各チップセレクトに対する両 NOR フラッシュウェイトステート設定をサポートします。

図 103. ウェイト設定波形

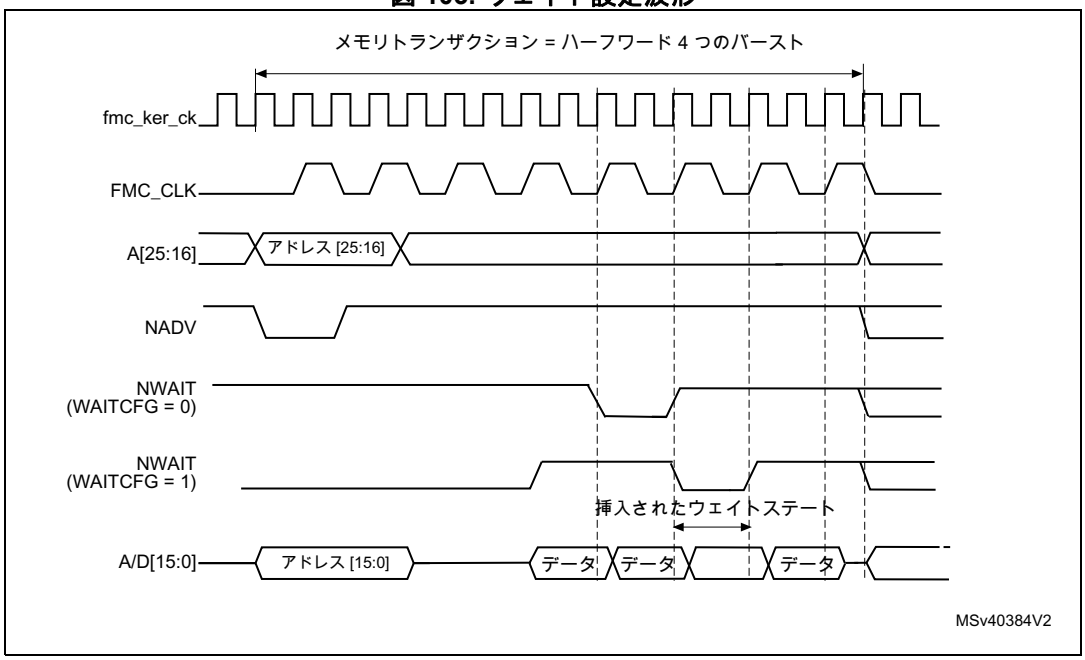
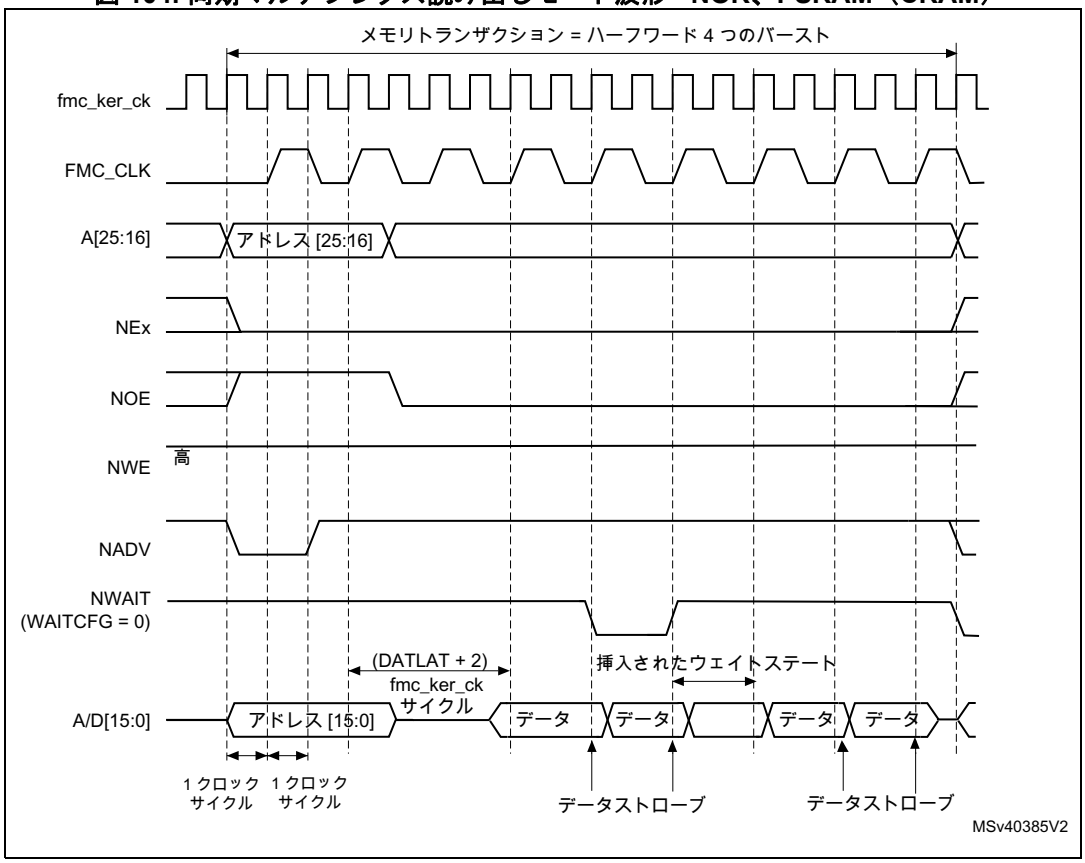


図 104. 同期マルチプレクス読み出しモード波形 - NOR、PSRAM（CRAM）



1. バイトレーン出力（NBL は示されていません。NOR アクセス時にはハイに保たれ、PSRAM（CRAM）アクセス時にはローに保たれます。）

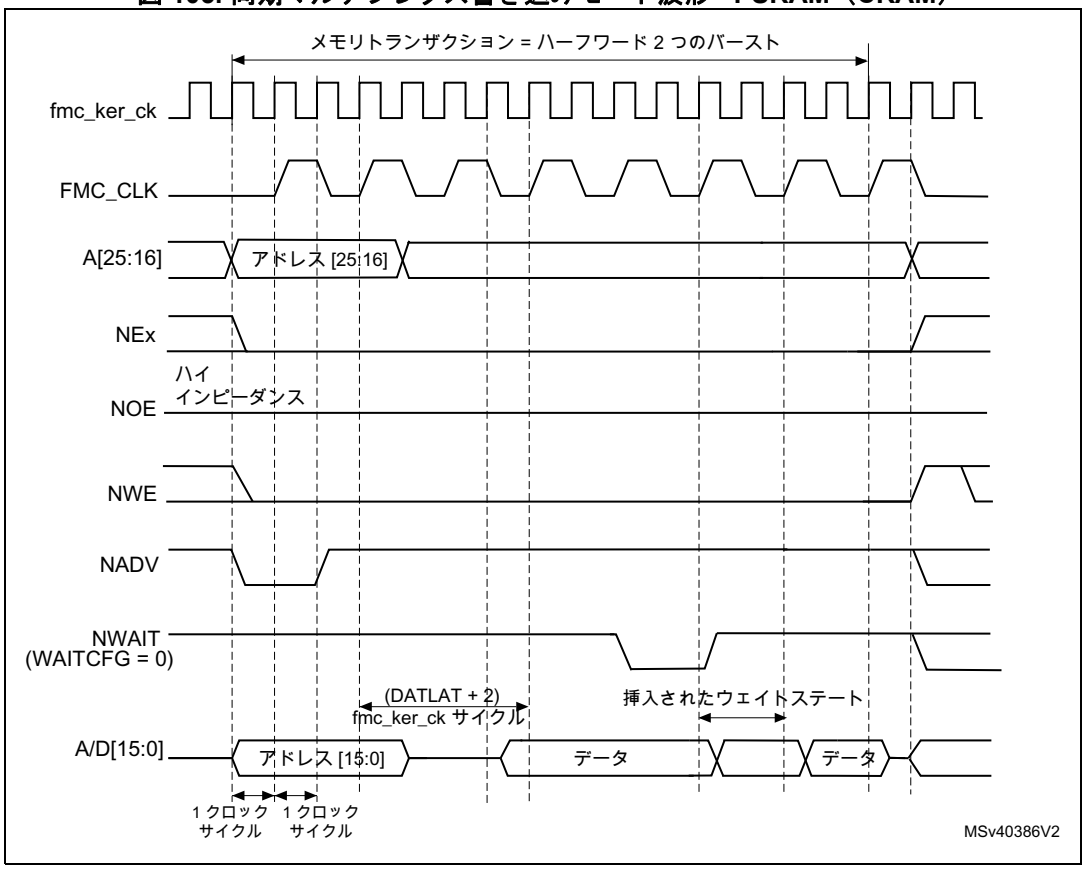
表 171. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	MC	0x1
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読み出し時は影響しません。
18:16	CPSIZE	必要に応じて設定します。（CRAM 1.5 使用時は 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	同期読み出し時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	予約済み	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済み	0x1
6	FACCEN	メモリサポートに従って設定します （NOR 型フラッシュメモリ）
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1 または 0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 172. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
27-24	DATLAT	データ遅延
23-20	CLKDIV	CLK = fmc_ker_ck とするには、0x0 （サポートされていません） CLK = 2 × fmc_ker_ck とするには 0x1 ..
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

図 105. 同期マルチプレクス書き込みモード波形 - PSRAM（CRAM）



1. メモリは、NWAIT 信号を 1 サイクル前に発行しなければならないので、WAITCFG を 0 にプログラムする必要があります。
2. バイトレーン（NBL）出力は、図に示されておらず、NEx がアクティブな間はローに保たれます。

表 173. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31	FMCEN	0x0
30-26	予約済み	0x000
25-24	BMAP	必要に応じて設定します。
23-22	予約済み	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読み出し時は影響しません。
18:16	CPSIZE	必要に応じて設定します。（CRAM 1.5 使用時は 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	0x1
11	WAITCFG	0x0

表 173. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
10	予約済み	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期読み出し時は影響しません。
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 174. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31-30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
23-20	CLKDIV	CLK = fmc_ker_ck とするには、0x0 （サポートされていません） CLK = 2 × fmc_ker_ck とするには 0x1
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

22.7.6 NOR/PSRAM コントローラレジスタ

SRAM/NOR 型フラッシュチップセレクト制御レジスタ 1～4（FMC_BCR1～4）

アドレスオフセット：8 * (x – 1), x = 1...4

リセット値：バンク 1 は 0x0000 30DB、バンク 2～4 は 0x0000 30D2

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FMCEN	Res.	Res.	Res.	Res.	Res.	BMAP[1:0]		Res.	Res.	WFDIS	CCLKEN	CBURSTRW	CPSIZE[2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID		MTYP		MUXEN	MBKEN
rw						rw				rw	rw	rw	rw			rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31 **FMCEN**：FMC コントローラ有効化

このビットは、FMC コントローラを有効または無効にします。

0：FMC コントローラを無効にします。

1：FMC コントローラを有効にします。

注： FMC_BCR2～4 レジスタの FMCEN ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **BMAP[1:0]**：FMC バンクのマッピング

これらのビットは、SDRAM バンク 2 の再マップ、または FMC NOR/PSRAM バンクと SDRAM バンクのスワップを可能にします。表 10 を参照してください。

01：デフォルトのマッピングです（図2 と表10 を参照）。

01：NOR/PSRAM バンクと SDRAM バンク 1／バンク 2 がスワップされます。

10：SDRAM バンク 2 は FMC バンク 2 上に再マップされ、引き続きデフォルトのマッピングでアクセス可能です。

11：予約済み

注： FMC_BCR2～4 レジスタの BMAP ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **WFDIS**：書き込み FIFO 無効化

このビットは、FMC コントローラで使用される書き込み FIFO を無効にします。

0：書き込み FIFO 有効（リセット後のデフォルト）

1：書き込み FIFO 無効

注： FMC_BCR2..4 レジスタの WFDIS ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 20 **CCLKEN**：連続クロック有効化

このビットは、外部メモリデバイスに対する FMC_CLK クロック出力を有効化します。

0：FMC_CLK は同期メモリアクセス（読み出し／書き込みトランザクション）時にのみ発生されます。FMC_CLK クロック比は、FMC_BCRx レジスタに設定した CLKDIV 値により指定されます （リセット後のデフォルト）。

1：FMC_CLK は、非同期アクセスと同期アクセス時に連続的に発生されます。CCLKEN がセットされると、FMC_CLK クロックがアクティブになります。

*注： **FMC_BCR2..4 レジスタの CCLKEN ビットは 無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。FMC_CLK連続クロックを発生するときは、バンク 1 を同期モードに設定する必要があります。***

CCLKEN ビットがセットされている場合、FMC_CLK クロック比は FMC_BTR1 レジスタの CLKDIV 値で指定されます。FMC_BWTR1 の CLKDIV は、無視されます。

同期モードを使用し、かつ CCLKEN ビットがセットされる場合、バンク 1 以外のバンクに接続される同期メモリはすべて、同じクロックで駆動されます（他のバンクに対する FMC_BTR2..4 レジスタとFMC_BWTR2..4 レジスタの CLKDIV 値は無効です）。

ビット 19 **CBURSTRW**：書き込みバースト有効化

バーストモードで動作する PSRAM（CRAM）の場合、このビットは書き込み動作時に同期アクセスを有効にします。同期読み出しアクセスに対するイネーブルビットは、FMC_BCRx レジスタの BURSTEN ビットです。

0：書き込み動作は、常に非同期モードで実行されます。

1：書き込み動作は、同期モードで実行されます。

ビット 18:16 **CPSIZE[2:**0]：CRAM ページサイズ

これらのビットは、ページ間でアドレス境界を跨ぐバーストアクセスができないCellular RAM 1.5 に使用されます。これらのビットを設定すると、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割します（ページサイズについてはメモリのデーターシートを参照）。

000：ページ境界を跨ぐ場合、バーストは分割されません（リセット後のデフォルト）。

001：128 バイト

010：256 バイト

100：1024 バイト

他の設定：予約済み

ビット 15 **ASYNCWAIT**：非同期転送中のウエイト信号

このビットは、非同期プロトコル時でも、FMC が NWAIT 信号を有効化／無効化できるようにします。

0：非同期プロトコル動作中は、NWAIT 信号は考慮されません（リセット後のデフォルト）。

1：非同期プロトコル動作中に NWAIT 信号が考慮されます。

ビット 14 **EXTMOD：拡張モード有効化**

このビットを使うと、FMC が FMC_BWTR レジスタ内で非同期アクセスの書き込みタイミングをプログラムできるため、読み出しと書き込みで異なるタイミングが可能になります。

0：FMC_BWTR レジスタ内部の値は考慮されません（リセット後のデフォルト）。

1：FMC_BWTR レジスタ内部の値が考慮されます。

*注： **拡張モードを無効化した場合、FMC は次のようにモード 1 またはモード 2 で動作することができます。***

- *モード1は、SRAM/PSRAM メモリタイプを選択した場合 （MTYP = 0x0 または 0x01）のデフォルトモードです。*
- *モード2は、NOR メモリタイプを選択した場合 （MTYP = 0x10）のデフォルトモードです。*

ビット 13 **WAITEN**：ウェイト有効ビット

このビットは、同期モードでフラッシュメモリをアクセスするとき NWAIT 信号を使用したウェイトステートの挿入を有効化／無効化します。

0：NWAIT 信号は無効です（レベルは考慮されず、プログラムされたフラッシュ遅延時間後にウェイトステートは挿入されません）。

1：NWAIT 信号は有効です（プログラムされた遅延時間後にレベルが考慮され、アサートされた場合に、ウェイトステートが挿入されます）（リセット後のデフォルト）。

ビット 12 **WREN**：書き込み有効ビット

このビットは、FMC によるバンクへの書き込み動作の有効／無効を示します。

0：FMC によるバンクへの書き込み動作は無効にされています。AXI スレーブエラーが報告されます。

1：FMC によるバンクへの書き込み動作は有効にされています（リセット後のデフォルト）。

ビット 11 **WAITCFG**：ウェイトタイミング設定

NWAIT 信号は、メモリからのデータが有効か、または同期モードでフラッシュメモリをアクセスするときウェイトステートを挿入すべきかを表示します。この設定ビットは、ウェイトステートの 1 クロック前、またはウェイトステート中に、メモリによって NWAIT がアサートされるかどうかを定義します。

0：NWAIT 信号は、ウェイトステートの 1 データサイクル前にアクティブです（リセット後のデフォルト）。

1：NWAIT 信号は、ウェイトステートの間アクティブです（PSRAM には不使用）。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **WAITPOL**：ウェイト信号極性ビット

このビットは、同期モードまたは非同期モードで使用されるメモリからのウェイト信号の極性を定義します。

0：NWAIT はアクティブローです（リセット後のデフォルト）。

1：NWAIT はアクティブハイです。

ビット 8 **BURSTEN**：バースト有効ビット

このビットは、読み出し動作での同期アクセスを有効化/無効化します。これは、バーストモードでの同期メモリ動作に対してのみ有効です。

0：バーストアクセスモードは無効です（リセット後のデフォルト）。読み出しアクセスは非同期モードで実行されます。

1：バーストモードを有効化。読み出しアクセスは同期モードで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FACCEN**：フラッシュアクセス有効化

NOR 型フラッシュメモリのアクセス動作を有効にします。

0：対応する NOR 型フラッシュメモリアクセスは無効です。

1：対応する NOR フラッシュメモリアクセスは有効です（リセット後のデフォルト）。

ビット 5:4 **MWID[1:0]：メモリのデータバス幅**

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。

00：8 ビット

01：16 ビット（リセット後のデフォルト）

10：32 ビット

11：予約済み

ビット 3:2 **MTYP[1:0]：メモリタイプ**

対応するメモリバンクに接続される外部メモリのタイプを定義します。
00：SRAM（バンク 2...4 のリセット後のデフォルト）
01：PSRAM（CRAM）
10：NOR 型フラッシュ /OneNAND フラッシュ（バンク 1 のリセット後のデフォルト）
11：予約済み

ビット 1 **MUXEN：アドレス / データマルチプレクス有効ビット**

このビットがセットされているときには、データバス上でアドレスとデータがマルチプレクスされます。NOR および PSRAM メモリでのみ有効です。
0：アドレス/データはマルチプレクスされません。
1：アドレス/データはデータバス上でマルチプレクスされます（リセット後のデフォルト）。

ビット 0 **MBKEN：メモリバンク有効ビット**

このビットはメモリバンクを有効にします。リセット後は、バンク 1 は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AXI バスでエラーが発生します。
0：対応するメモリバンクは無効です
1：対応するメモリバンクは有効です。

SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）

アドレスオフセット：0x04 + 8 * (x – 1), x = 1..4

リセット値：0x0FFF FFFF

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。FMC_BCRx レジスタの EXTMOD ビットがセットされている場合、このレジスタは、書き込みと読み出しアクセスに分割されます。すなわち、読み出しアクセスを設定するためのレジスタ（このレジスタ）と、書き込みアクセスを設定するためのレジスタ（FMC_BWTR x レジスタ）の 2 つのレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ACCMOD		DATLAT				CLKDIV				BUSTURN			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD[1:0]**：アクセスモード

タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 のときのみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:24 **DATLAT[3:0]**：（下のビット説明参照）同期メモリのデータ遅延

読み出し／書き込みバーストモードを有効にした同期 アクセスの場合（BURSTEN/CBURSTRW ビットをセット）、これらのビットは最初のデータの読み書きの前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、fmc_ker_ck 周期ではなく、FMC_CLK 周期で表されます。

非同期アクセスでは、この値は無視されます。

0000：最初のバーストアクセスの 2 FMC_CLK クロックサイクルのデータ遅延

1111：最初のバーストアクセスの 17 FMC_CLK クロックサイクルのデータ遅延（リセット後のデフォルト）

ビット 23:20 **CLKDIV[3:0]**：クロック分周比（FMC_CLK 信号）

FMC_CLK クロック出力信号の周期を定義します。fmc_ker_ck サイクル数で表されます。

0000：予約済み

0001：FMC_CLK 周期 = 2 x fmc_ker_ck 周期

0010：FMC_CLK 周期 = 3 x fmc_ker_ck 周期

1111：FMC_CLK 周期 = 16 x fmc_ker_ck 周期（リセット後のデフォルト値）

非同期 NOR 型フラッシュ、SRAM、または PSRAM アクセスでは、この値は無視されます。

注： FMC_CLK 分周比の式については、セクション 22.7.5: 同期トランザクション を参照してください。

ビット 19:16 **BUSTURN**：バスターンアラウンドフェーズ時間

これらのビットは、書き込みから読み出しまで（および読み出しから書き込みまで）のトランザクションの終わりに遅延を追加するためにソフトウェアで書き込まれます。この遅延を使うと、連続したトランザクションの間の最小時間（NEx ハイから NEx ローまでの t_{EHEL}）と、読み出しアクセス後にメモリがデータバスを解放するために要する最大時間（t_{EHQZ}）を一致させることができます。プログラムされたバスターンアラウンド遅延は、非同期読み出しトランザクション（マルチプレクスまたはモード D）や書き込みトランザクションと、その他のスタティックバンクに対する非同期／同期読み書きの間に挿入されます。読み出し時、バンクを同じものまたは別のものにできますが、書き込み時は、マルチプレクスまたはモード D を除いて別にできます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。

次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスまたはモード D 以外の同じスタティックメモリバンクへの 2 つの連続した非同期書き込み転送の間には挿入されません。
- 次の動作の間で 1 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - マルチプレクスとモード D 以外の同じスタティックメモリバンクへの 2 回の連続した非同期読み出し転送
  - マルチプレクスまたはモード D 以外のすべてのスタティックバンクまたはダイナミックバンクへの非同期／同期書き込みに対する非同期読み出し
  - 非同期（モード 1、2、A、B、C）読み出しと別のスタティックバンクからの読み出し
- 次の動作の間で 2 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 回の連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）アクセスと、スタティックメモリバンクへの非同期書き込み／読み出し転送（読み出しの場合、バンクを同じものまたは異なるものに設定できます）
  - 2 つの連続した同期読み出し（バーストまたはシングル）、それに続く別のスタティックメモリバンクへのすべての同期／非同期読み出しまたは書き込み
- 次の動作の間で 3 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）アクセスと、同じまたは異なるバンクからの同期読み出し

0000：BUSTURN フェーズ時間 = 0 fmc_ker_ck クロックサイクル追加

...

1111：BUSTURN フェーズ時間 = 15 x fmc_ker_ck クロックサイクル追加（リセット後のデフォルト値）

ビット 15:8 **DATAST**：データフェーズ時間

これらのビットは、非同期アクセスで使用されるデータフェーズの時間を定義するためにソフトウェアにより書き込まれます（*図 88* ～ *図 100*を参照）。

0000 0000：予約済み

0000 0001：DATAST フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

...

1111 1111：DATAST フェーズ時間 = 255 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図（*図 88* ～ *図 100*）を参照してください。

例：モード 1、書き込みアクセス、DATAST = 1：データフェーズ時間 = DATAST + 1 = 1 fmc_ker_ck クロックサイクル

注：　同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD**：アドレスホールドフェーズ時間

これらのビットは 、*アドレスホールド*フェーズの時間を定義するためにソフトウェアから書き込まれ（*図 88* ～ *図 100*を参照）、モード D またはマルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLD フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0010：ADDHLD フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

...

1111：ADDHLD フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図（*図 88* ～ *図 100*）を参照してください。

注： ***同期アクセスでは、この値は使用されず、アドレスホールドフェーズは、常に 1 メモリクロック周期の長さです。***

ビット 3:0 **ADDSET**：アドレスセットアップフェーズ時間

これらのビットは、*アドレスセットアップ*フェーズの時間を定義するために、ソフトウェアによって書き込まれ（*図 88* ～ *図 100*を参照）、SRAM、ROM、非同期 NOR フラッシュで使用されます。

0000：ADDSET フェーズ時間 = 0 x fmc_ker_ck クロックサイクル

...

1111：ADDSET フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図（*図 88* ～ *図 100*）を参照してください。

注： ***同期アクセスでは、この値は無視されます。***

マルチプレクスモードとモード D では、ADDSET の最小値は 1 です。

注： ***PSRAM（CRAM）は内部リフレッシュがあるため遅延（レイテンシ）が可変になっています。したがって、これらのメモリは必要に応じて遅延時間を延長するために、遅延フェーズ中ずっと NWAIT 信号を発行します。***

SPRAM（CRAM）では、DATLAT を 0 に設定してください。こうすると、FMCは、すぐに遅延フェーズを終了して、メモリからの NWAIT 信号のサンプリングを開始し、メモリがレディ状態になると、読み出しまたは書き込みを開始します。

この方法は、古い世代のフラッシュメモリと異なり、NWAIT 信号を発行する最新世代の同期フラッシュメモリでも使用できます（使用中の特定のフラッシュメモリのデータシートを参照してください）。

SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1～4（FMC_BWTR1～4）

アドレスオフセット：0x104 + 8 * (x – 1), x = 1...4

リセット値：0x0FFF FFFF

このレジスタには各メモリバンクの制御情報が格納されます。SRAM、PSRAM、NOR フラッシュメモリに使用されます。FMC_BCRx レジスタの EXTMOD ビットがセットされているときには、このレジスタは書き込みアクセスについてアクティブになります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD**：アクセスモード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 にセットされている場合のみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 BUSTURN：バスターンアラウンドフェーズ時間

これらのビットは、書き込みトランザクションの終わりに遅延を追加して、次のように連続トランザクション間の最小時間（ENx ハイから ENx ローまでの t_{EHEL}）を一致させるために、ソフトウェアによって書き込まれます。

(BUSTRUN + 1) fmc_ker_ck 周期 ≥ $t_{EHELmin}$

プログラムされたバスターンアラウンド遅延は、非同期書き込み転送と、その他のスタティックバンクに対する非同期／同期読み書き転送の間に挿入されます。読み出し時、バンクを同じものまたは別のものにできますが、書き込み時は、マルチプレクスまたはモード D を除いて別にできます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスモードおよびモード D を除いて同じスタティックメモリバンクに向けた 2 つの連続した非同期書き込み転送の間に挿入されません。
- 次の動作の間で 2 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）転送と、スタティックメモリバンクへの非同期書き込み／読み出し転送
- 次の動作の間で 3 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同じまたは別のバンクへの同期書き込み（バーストまたはシングル）転送と同期読み出し

0000：BUSTURN フェーズ時間 = 0 fmc_ker_ck クロックサイクル追加

...

1111：BUSTURN フェーズ時間 = 15 fmc_ker_ck クロックサイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST**：データフェーズ時間

これらのビットは、非同期 SRAM、PSRAM、NOR フラッシュメモリアクセスで使用されるデータフェーズの時間を定義するために、ソフトウェアによって書き込まれます（*図 88* ～ *図 100*を参照）。

0000 0000：予約済み

0000 0001：DATAST フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

...

1111 1111：DATAST フェーズ時間 = 255 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

ビット 7:4 **ADDHLD**：アドレスホールドフェーズ時間

これらのビットは 、*アドレスホールド* フェーズの時間を定義するためにソフトウェアによって書き込まれ（*図 88* ～ *図 100*を参照）、非同期マルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLD フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0010：ADDHLD フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

...

1111：ADDHLD フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

注： ***同期 NOR 型フラッシュアクセスでは、この値は使用されず、アドレスホールドフェーズは常に 1 フラッシュクロック周期の長さです。***

ビット 3:0 **ADDSET**：アドレスセットアップフェーズ時間。

これらのビットは、*アドレスセットアップ* フェーズの時間を fmc_ker_ck サイクル数で定義するためにソフトウェアによって書き込まれ（*図 88* ～ *図 100*を参照）、非同期アクセスで使用されます。

0000：ADDSET フェーズ時間 = 0 x fmc_ker_ck クロックサイクル

...

1111：ADDSET フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

注： ***同期アクセスでは、この値は使用されず、アドレスセットアップフェーズは常に 1 フラッシュクロック周期の長さです。マルチプレクスモードでは、ADDSET の最小値は 1 です。***

22.8 NAND 型フラッシュコントローラ

FMC は、8 ビットおよび 16 ビットの NAND フラッシュメモリを駆動するのに適した信号タイミングを発生させます。

NAND バンクは、専用レジスタによって設定されます（*セクション 22.8.7*を参照）。プログラ可能なメモリパラメータは、アクセスタイミング（*表 175*を参照）と ECC 構成を含みます。

表 175. プログラ可能なNAND フラッシュのアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
メモリセットアップ時間	コマンドアサーション前にアドレスをセットアップするために必要なクロックサイクル数（fmc_ker_ck）	読み出し／書き込み	AHB クロックサイクル（fmc_ker_ck）	1	255
メモリウェイト	コマンドアサーションの最小時間（fmc_ker_ck クロックサイクル）	読み出し／書き込み	AHB クロックサイクル（fmc_ker_ck）	2	255
メモリホールド	コマンドをデアサートした後にアドレス（書き込みアクセス時はデータも）を保持する必要があり、その間のクロックサイクル（fmc_ker_ck）数	読み出し／書き込み	AHB クロックサイクル（fmc_ker_ck）	1	254
メモリデータバスハイインピーダンス	書き込みアクセス開始後、データバスがハイインピーダンス状態に保たれている間のクロックサイクル（fmc_ker_ck）数	書き込み	AHB クロックサイクル（fmc_ker_ck）	0	254

22.8.1 外部メモリインタフェース信号

次の表に、通常、NAND フラッシュメモリとのインタフェースに使用される代表的な信号を示します。

注： ***接頭辞「N」は、アクティブローの信号を表します。***

8 ビット NAND フラッシュメモリ

表 176. 8 ビット NAND フラッシュメモリ

FMC 信号名	I/O	機能
A[17]	O	NAND 型フラッシュアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型フラッシュコマンドラッチイネーブル（CLE）信号
D[7:0]	I/O	8 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読み出しイネーブル、NRE）
NWE	O	書き込みイネーブル
NWAIT/INT	I	FMC への NAND フラッシュレディ／ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16 ビット NAND フラッシュメモリ

表 177. 16 ビット NAND フラッシュメモリ

FMC 信号名	I/O	機能
A[17]	O	NAND 型フラッシュアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型フラッシュコマンドラッチイネーブル（CLE）信号
D[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読み出しイネーブル、NRE）
NWE	O	書き込みイネーブル
NWAIT/INT	I	FMC への NAND フラッシュレディ／ビジー入力信号

注： FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

22.8.2 NAND 型フラッシュによってサポートされるメモリとトランザクション

表 178 に、サポートされるデバイス、アクセスモード、およびトランザクションを示します。NAND フラッシュコントローラによって許可されない（またはサポートされない）トランザクションは、グレイで示されています。

表 178. サポートされるメモリおよびトランザクション

デバイス	モード	読み出し／書き込み	AXI データサイズ	メモリデータサイズ	設定許可／設定禁止	コメント
NAND 8 ビット	非同期	R	8	8	Y	-
	非同期	W	8	8	Y	-
	非同期	R	16	8	Y	2 つの FMC アクセスに分割
	非同期	W	16	8	Y	2 つの FMC アクセスに分割
	非同期	R	32	8	Y	4 つの FMC アクセスに分割
	非同期	W	32	8	Y	4 つの FMC アクセスに分割
	非同期	R	32	8	Y	8 つの FMC アクセスに分割
	非同期	W	32	8	Y	8 つの FMC アクセスに分割
NAND 16 ビット	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	32	16	Y	4 つの FMC アクセスに分割
	非同期	W	32	16	Y	4 つの FMC アクセスに分割

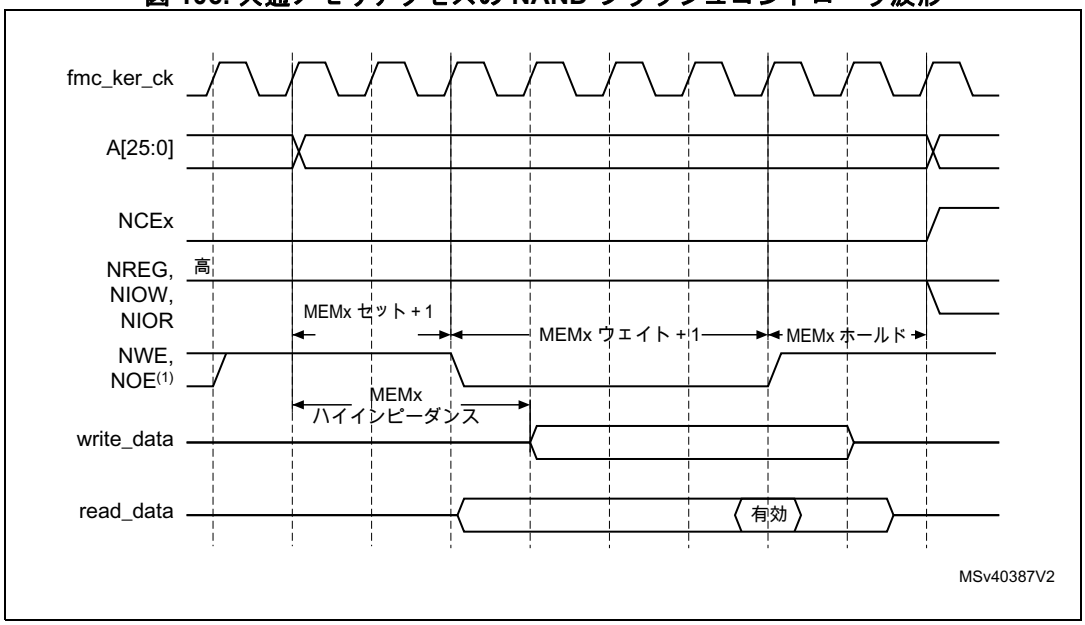
22.8.3 NAND フラッシュメモリのタイミング図

NAND 型フラッシュメモリのバンクは、以下のレジスタを通じて管理されます。

- 制御レジスタ：FMC_PCR
- 割り込みステータスレジスタ：FMC_SR
- ECC レジスタ：FMC_ECCR
- 共通メモリ空間用タイミングレジスタ：FMC_PMEM
- 属性メモリ空間用タイミングレジスタ：FMC_PATT

各タイミング設定レジスタは、NAND フラッシュアクセスの 3 つのフェーズの fmc_ker_ck サイクル数を定義するための 3 つのパラメータ と、書き込みアクセス時にデータバスの駆動を開始するタイミングを定義する 1 つのパラメータを格納します。*図 106* に共通メモリアクセス用のタイミングパラメータの定義を示します。属性メモリ空間のアクセスタイミングが同様であることがわかります。

図 106. 共通メモリアクセスの NAND フラッシュコントローラ波形



1. 書き込みアクセス時、NOE はハイ（非アクティブ）に保たれます。読み出しアクセス時、NWE はハイ（非アクティブ）に保たれます。
2. 書き込みアクセスの場合、ホールドフェーズ遅延は（MEMHOLD）fmc_ker_ck サイクルとなり、読み出しアクセスの場合は (MEMHOLD + 1) fmc_ker_ck サイクルとなります。

22.8.4 NAND フラッシュ動作

NAND フラッシュデバイスのコマンドラッチイネーブル（CLE）およびアドレスラッチイネーブル（ALE）信号は、FMC コントローラのアドレス信号によって駆動されます。これは、NAND フラッシュメモリにコマンドまたはアドレスを送信するには、CPU は、CPU のメモリ空間内の特定のアドレスに書き込みを行う必要があることを意味します。

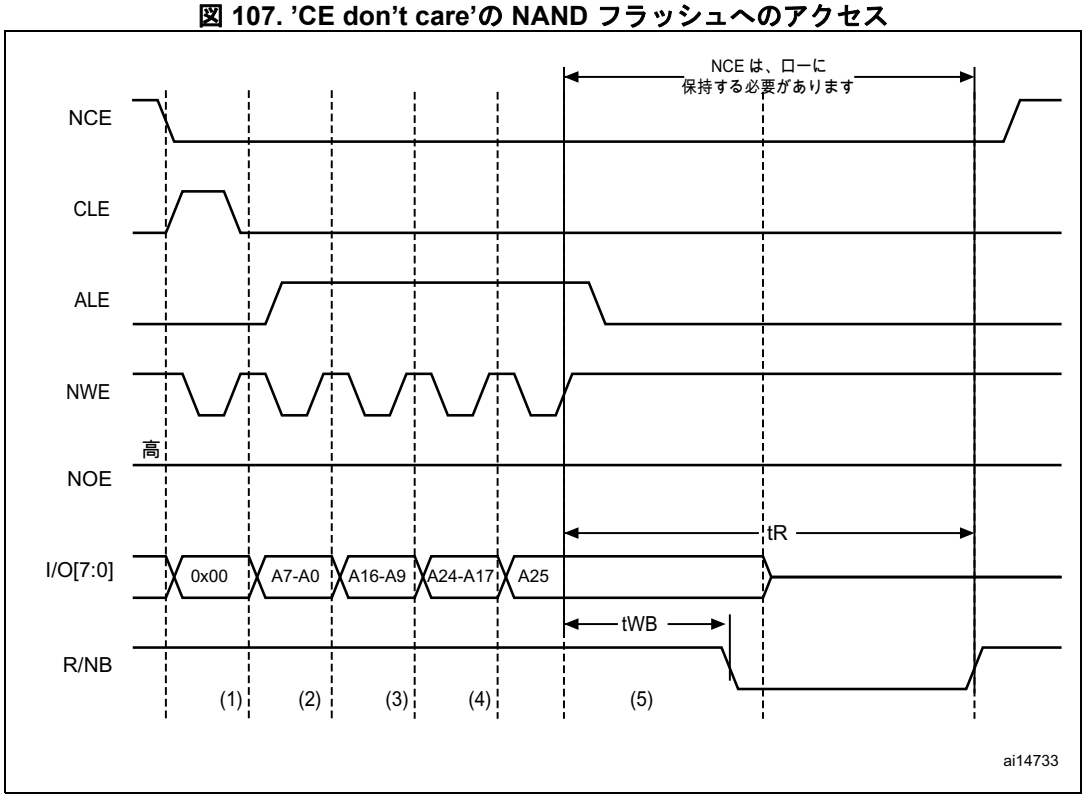
NAND フラッシュデバイスからの一般的なページ読み出し動作には、次のステップが必要です。

1. FMC_PCR レジスタと FMC_PMEM レジスタ（デバイスによっては、FMC_PATTも。*セクション 22.8.5 : NAND フラッシュのプリウェイト機能*を参照）を、NAND フラッシュメモリの特性（NAND フラッシュのデータバス幅に対する PWID ビット 、必要に応じて PWAITEN = 0 または 1、タイミング設定については *セクション 22.6.2 : NAND フラッシュメモリアドレスマッピング* を参照）に従って設定することで、対応するメモリバンクのプログラムおよび有効化を行います。
2. CPU は、1 つのフラッシュコマンドバイトと等しいデータバイト（たとえば、Samsung の NAND フラッシュデバイスの場合は 0x00）で、共通メモリ空間にバイト書き込みを行います。NAND フラッシュの LE 入力は、書き込みストローブ（NWE 上のローパルス）の間はアクティブであり、書き込まれたバイトは、NAND フラッシュによってコマンドとして認識されます。コマンドがメモリデバイスにラッチされたら、次のページ読み出し動作のために書き込む必要はありません。
3. CPU は、4 バイト（より小容量のデバイスの場合は 3 バイト）を共通メモリまたは属性空間に書き込むことによって、読み出し操作の開始アドレス（STARTAD）を送信できます（STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、および 64 M ビット x 8 ビット NAND フラッシュの場合は最後に STARTAD[25]）。NAND 型フラッシュデバイスの ALE 入力は、書き込みストローブの間アクティブであり（NWE 上のローパルス）、書き込まれたバイトは、読み出し動作の開始アドレスとして解釈されます。属性メモリ空間を使用すると、FMC の異なるタイミング設定を使用でき、一部の NAND フラッシュメモリで必要とされるプリウェイト機能を実装できます（詳細については、*セクション 22.8.5 : NAND フラッシュのプリウェイト機能*を参照してください）。
4. コントローラは、NAND フラッシュメモリがレディ（R/NB 信号がハイ）になるのを待ってから、同じメモリバンクまたは別のメモリバンクへの新しいアクセスを開始します。待っている間、コントローラは NCE 信号をアクティブ（ロー）に保ちます。
5. CPU は、NAND フラッシュページ（データフィールド＋スペアフィールド）をバイト単位で読み出すために、共通メモリ空間からバイト読み出し動作を行います。
6. 次の NAND フラッシュページは、CPU コマンドやアドレス書き込み動作なしに読み出すことができます。これは次の 3 つの方法で行うことができます。
   - ステップ 5 に記述された動作を実行します。
   - ステップ 3 から動作をリスタートすることによって、新しいランダムアドレスにアクセスできます。
   - ステップ 2 からリスタートすることによって、NAND 型フラッシュデバイスに新しいコマンドを送信できます。

22.8.5 NAND フラッシュのプリウェイト機能

一部の NAND フラッシュデバイスでは、アドレスの最後の部分を書き込んだ後、コントローラは R/NB 信号がLowになるまで待つ必要があります。（*図 107*を参照。）

図 107. 'CE don't care'の NAND フラッシュへのアクセス



1. CPU はアドレス 0x7001 0000 にバイト 0x00 を書き込みました。
2. CPU はアドレス 0x7002 0000 にバイト A7～A0 を書き込みました。
3. CPU はアドレス 0x7002 0000 にバイト A16～A9 を書き込みました。
4. CPU はアドレス 0x7002 0000 にバイト A24～A17 を書き込みました。
5. CPU はアドレス 0x8802 0000 にバイト A25 を書き込みました : FMC は FMC_PATT2 のタイミング定義を使用して書き込みアクセスを実行します。ここで ATTHOLD ≥ 7（ただし、(7+1) * fmc_ker_ck = 112 ns > t_{WB} max)。これにより、NCE は R/NB がローから再びハイになるまでローを維持します（NCE が無視されない NAND 型フラッシュメモリの場合のみ必要）。

この機能が必要な場合、MEMHOLD 値をプログラムすることにより、t_{WB} タイミングを満たすことができます。ただし、NAND フラッシュメモリに対するすべての CPU 読み出しアクセスでは (MEMHOLD + 1) fmc_ker_ck サイクル、CPU 書き込みアクセスでは (MEMHOLD) fmc_ker_ck サイクルのホールド遅延が NWE 信号の立ち上がりエッジと次のアクセスとの間に挿入されます。

このタイミング制約に対処するためには、t_{WB} タイミングに応じた ATTHOLD 値でタイミングレジスタをプログラムし、MEMHOLD 値を最小値のままにしておくことで、属性メモリ空間を使用できます。この場合、CPU は NAND フラッシュのすべての読み出しおよび書き込みアクセスには共通メモリ空間を使用する必要があります。ただし、NAND フラッシュデバイスに最後のアドレスバイトを書き込むときには、属性メモリ空間に書き込む必要があります。

22.8.6 NAND 型フラッシュメモリ内のエラー訂正コード（ECC）の計算

FMC コントローラにはエラー訂正コード計算用のハードウェアブロックが 1 つあります。このハードウェアブロックは、ソフトウェアによる ECC 処理時にホスト CPU の負荷を軽減します。ECC ブロックは NAND バンクに関連付けられます。

FMC に実装されたエラー訂正コード（ECC）のアルゴリズムでは、NAND フラッシュメモリに対する 256、512、1024、2048、4096、または 8192 バイトの読み出しまたは書き込みごとに、1 ビットのエラー訂正と 2 ビットのエラー検出を実行できます。BCH8 コーディングアルゴリズムを採用し、RowとColumのパリティ計算を行います。

ECC モジュールは、NAND フラッシュメモリバンクがアクティブになるたびに、NAND フラッシュデータバスおよび読み出し／書き込み信号（NCE と NWE）を監視します。

ECC は次のように動作します。

- NAND フラッシュバンクにアクセスするときは、D[15:0] バス上にあるデータがラッチされて、ECC 計算に使用されます。
- NAND フラッシュメモリの他のアドレスをアクセスするときは、ECC ロジックはアイドル状態となり、動作しません。このため、NAND フラッシュメモリに対するコマンドまたはアドレスを定義する書き込み動作は、ECC計算で考慮されません。

目的のバイト数がホスト CPU によって NAND フラッシュメモリから読み出されるか、NAND フラッシュメモリに書き込まれたら、FMC_ECCR レジスタを読み出して、計算値を取得する必要があります。読み出し後は、ECCEN ビットを"0"にリセットすることによってクリアされます。新しいデータブロックを計算するには、FMC_PCR レジスタの ECCEN ビットを 1 にセットする必要があります。

ECC計算を行うには以下のシーケンスを実行します。

1. FMC_PCR レジスタで ECCEN ビットを有効にします。
2. NAND 型フラッシュメモリページへデータを書き込みます。NAND ページへ書き込む際、ECC ブロックは ECC 値を計算します。
3. ECC コードがレディ状態（FIFO エンプテイ）になるまで待ちます。
4. FMC_ECCR レジスタにある ECC 値を読み出して、変数内に格納します。
5. ECCEN ビットをクリアし、FMC_PCR レジスタで有効にした後、書き込んだデータを NAND ページから読み出します。NAND ページを読み出す際、ECC ブロックは ECC 値を計算します。
6. FMC_ECCR レジスタにある新しい ECC 値を読み出します。
7. 2 つの ECC 値が同じ場合、訂正は不要です。その他の場合、ECC エラーが存在するためソフトウェア訂正ルーチンがエラーが訂正できるか否かの情報を返します。

22.8.7 NAND フラッシュコントローラレジスタ

NAND フラッシュ制御レジスタ（FMC_PCR）

アドレスオフセット：0x80

リセット値：0x0000 0018

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS			TAR				TCLR				Res.	Res.	ECCEN	PWID		Res.	PBKEN	PWAITEN	Res.
												rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw		rw	rw	

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 **ECCPS**：ECC ページサイズ

これらのビットによって、拡張 ECC のページサイズを定義します。

000：256 バイト

001：512 バイト

010：1024 バイト

011：2048 バイト

100：4096 バイト

101：8192 バイト

ビット 16:13 **TAR**：ALE から RE 間の遅延

これらのビットによって、ALE ローから RE ローまでの時間を fmc_ker_ck クロックサイクル数で設定します。

時間の計算式は、$t_ar = (TAR + SET + 2) \times t_{fmc_ker_ck}$（ここで、$t_{fmc_ker_ck}$ は FMC クロック周期）です。

0000：1 x fmc_ker_ck サイクル （デフォルト）

1111：16 x fmc_ker_ck サイクル

注： アドレス指定される空間に応じて、MEMSET または ATTSETにセットされます。

ビット 12:9 **TCLR**：CLE から RE 間の遅延

CLE ローから RE ローまでの時間を fmc_ker_ck クロックサイクル数で設定します。時間の計算式は、$t_clr = (TCLR + SET + 2) \times t_{fmc_ker_ck}$ （ここで、$t_{fmc_ker_ck}$ は fmc_ker_ck クロック周期）です。

0000：1 x fmc_ker_ck サイクル （デフォルト）

1111：16 x fmc_ker_ck サイクル

注： アドレス指定される空間に応じて、MEMSET または ATTSETにセットされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ECCEN**：ECC 計算ロジックイネーブルビット

0：ECC ロジックは無効であり、リセットされています（リセット後のデフォルト）。

1：ECC ロジックは有効です。

ビット 5:4 **PWID**：データバス幅。

これらのビットは外部メモリデバイスの幅を定義します。

00：8 ビット

01：16 ビット（リセット後のデフォルト）

10：予約済み

11：予約済み

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **PBKEN**：NAND フラッシュメモリバンクイネーブルビット

このビットはメモリバンクを有効にします。無効なメモリバンクにアクセスすると、AXI バスでエラーが発生します。

0：対応するメモリバンクは無効です（リセット後のデフォルト）。

1：対応するメモリバンクは有効です。

ビット 1 **PWAITEN**：ウェイト機能イネーブルビット

NAND フラッシュメモリバンクのウェイト機能を有効にします。

0：無効

1：有効

ビット 0 予約済みであり、リセット値に保持する必要があります。

FIFO ステータスおよび割り込みレジスタ（FMC_SR）

アドレスオフセット：0x84

リセット値：0x0000 0040

このレジスタは、FIFO ステータスと割り込みに関する情報を格納します。FMC の特徴的な機能は、最大 16 ワードのデータをメモリに書き込んで転送する際に FIFO を使用することです。

この機能を使用することで、FMC は FIFO をメモリにドレインしながら、迅速に FIFO への書き込みを行い、FMC 以外のペリフェラルへのトランザクションのために AXI バスを解放します。これらのレジスタビットの1 つは、ECC のために、FIFO のステータスを示します。

ECC は、データのメモリへの書き込み中に計算されます。正しい ECC を読み出すため、ソフトウェアは FIFO がエンプティになるまで待つ必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
									r	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FEMPT**：FIFO エンプティ

FIFO のステータスを表示する読み取り専用ビット

0：FIFO ノットエンプティ

1：FIFO エンプティ

ビット 5 **IFEN**：割り込み立ち下がりエッジ検出イネーブルビット

0：割り込み立ち下がりエッジ検出リクエストは無効

1：割り込み立ち下がりエッジ検出リクエストは有効

ビット 4 **ILEN**：割り込みハイレベル検出イネーブルビット

0：割り込みハイレベル検出リクエストは無効

1：割り込みハイレベル検出リクエストは有効

ビット 3 **IREN**：割り込み立ち上がりエッジ検出イネーブルビット

0：割り込み立ち上がりエッジ検出リクエストは無効

1：割り込み立ち上がりエッジ検出リクエストは有効

ビット 2 **IFS**：割り込み立ち下がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込み立ち下がりエッジは発生しませんでした。

1：割り込み立ち下がりエッジが発生しました。

注：　ソフトウェアがこのビットに 1 を書き込むとセットされます。

ビット 1 **ILS**：割り込みハイレベルステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込みハイレベルは発生しませんでした。

1：割り込みハイレベルが発生しました。

ビット 0 **IRS**：割り込み立ち上がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込み立ち上がりエッジは発生しませんでした。

1：割り込み立ち上がりエッジが発生しました。

注：　ソフトウェアがこのビットに 1 を書き込むとセットされます。

共通メモリ空間タイミングレジスタ 2..4（FMC_PMEM）

アドレスオフセット：アドレス：0x88

リセット値：0xFCFC FCFC

FMC_PMEMx 読み出し／書き込みレジスタは、NAND フラッシュメモリバンク のタイミング情報を格納します。この情報は、NAND フラッシュの共通メモリ空間に、コマンド、アドレスの書き込みアクセス、およびデータ読み出し／書き込みアクセスするために使われます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MEMHIZx								MEMHOLDx							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMWAITx								MEMSETx							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **MEMHIZ**：共通メモリ x データバスハイインピーダンス時間

共通メモリ空間への NAND フラッシュ書き込みアクセスの開始後、データバスをハイインピーダンスに維持する間の fmc_ker_ck クロックサイクル数を定義します。これは、書き込み トランザクションに対してのみ有効です。

0000 0000：0 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 23:16 **MEMHOLD**：共通メモリホールド時間

共通メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NWE、NOE）のデアサート後にアドレスを（書き込みアクセス時はデータも）保持する間の fmc_ker_ck クロックサイクル数を定義します。書き込みアクセス時は fmc_ker_ck クロックサイクル数、読み出しアクセス時は fmc_ker_ck + 1 クロックサイクル数とします。

0000 0000：予約済み

0000 0001：書き込みアクセスの場合 1 fmc_ker_ck サイクル／読み出しアクセスの場合 3 fmc_ker_ck サイクル

1111 1110：書き込みアクセスの場合 254 fmc_ker_ck サイクル／読み出しアクセスの場合 257 fmc_ker_ck サイクル

1111 1111：予約済み

ビット 15:8 **MEMWAIT**：共通メモリウェイト時間

共通メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NEW、NOE）をアサートするための fmc_ker_ck (+1) クロックサイクルの最小数を定義します。プログラムされた fmc_ker_ck 値の最後にあるウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1110：255 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **MEMSET**：共通メモリ x セットアップ時間

共通メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NWE、NOE）のアサーション前にアドレスをセットアップするための fmc_ker_ck (+1) クロックサイクル数を定義します。

0000 0000：fmc_ker_ck サイクル

1111 1110：255 x fmc_ker_ck サイクル

1111 1111：予約済み

属性メモリ空間タイミングレジスタ（FMC_PATT）

アドレスオフセット：0x8C

リセット値：0xFCFC FCFC

FMC_PATT 読み出し／書き込みレジスタは、NAND フラッシュメモリバンクのタイミング情報を格納します。この情報は、最後のアドレス書き込みアクセスの際に、タイミングが以前のアクセスのタイミングと異なる必要がある場合に、NAND フラッシュの属性メモリ空間へ 8 ビットアクセスするために使用されます（レディ／ビジー管理については、*セクション 22.8.5：NAND フラッシュのプリウェイト機能*を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ATTHIZ								ATTHOLD							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATTWAIT								ATTSET							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **ATTHIZx**：属性メモリデータバスハイインピーダンス時間

ソケット上の属性メモリ空間への NAND フラッシュ書き込みアクセスの開始後、データバスをハイインピーダンスに維持する間の fmc_ker_ck クロックサイクル数を定義します。これは、書き込みトランザクションに限り有効です。

0000 0000：0 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 23:16 **ATTHOLD**：属性メモリホールド時間

これらのビットは、属性メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NWE、NOE）をデアサートした後にアドレスを（書き込みアクセス時はデータも）保持する間の fmc_ker_ck クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：1 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 15:8 **ATTWAIT**：属性メモリウェイト時間

これらのビットは、属性メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NEW、NOE）をアサートするための x fmc_ker_ck (+1) クロックサイクルの最小数を定義します。プログラムされた fmc_ker_ck 値の最後にあるウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1110：255 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **ATTSET**：属性メモリセットアップ時間

属性メモリ空間への NAND フラッシュ読み出しまたは書き込みアクセスの際、コマンド（NWE、NOE）のアサーション前にアドレスをセットアップするための fmc_ker_ck (+1) クロックサイクル数を定義します。

0000 0000：1 x fmc_ker_ck サイクル

1111 1110：255 x fmc_ker_ck サイクル

1111 1111：予約済み

ECC 結果レジスタ（FMC_ECCR）

アドレスオフセット：0x94

リセット値：0x0000 0000

このレジスタは、FMC NAND コントローラの ECC 計算モジュールによって計算された現在のエラー訂正コード値を格納します。CPU が、NAND フラッシュメモリページ上の正しいアドレスに対してデータの書き込み／読み出しを行うときには（*セクション 22.8.6 : NAND 型フラッシュメモリ内のエラー訂正コード（ECC）の計算*を参照）、NAND フラッシュメモリに対して読み書きされたデータは、ECC 計算モジュールによって自動的に処理されます。X バイトを読み出した後（FMC_PCR レジスタの ECCPS フィールドに従って）、CPU は FMC_ECC レジスタから計算された ECC 値を読み出す必要があります。次に、これらの計算されたパリティデータがスペア領域に記録されているパリティ値と一致するか検証して、ページが有効か否か、訂正が必要か否かを確認します。FMC_ECCR レジスタは、読み出し後、ECCEN ビットを "0" に設定することによってクリアする必要があります。新しいデータブロックを計算するときは、ECCEN ビットを "1" にセットする必要があります。

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

ビット 31:0 **ECC**：ECC 結果

このフィールドは、ECC 計算ロジックによって計算された値を格納します。*表 179* に、これらのビットフィールドの値を示します。

表 179. ECC 結果関連ビット

ECCPS[2:0]	ページサイズ（バイト数）	ECC ビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]

参考資料

22.9 SDRAM コントローラ

22.9.1 SDRAM コントローラの主な機能

SDRAM コントローラの主な機能は次の通りです。

- 独立した設定を持つ 2 つの SDRAM バンク
- 8 ビット、16 ビット、32 ビット のデータバス幅
- 13 ビットのアドレス Row、11 ビットのアドレス Column、4 つの内部バンク：4x16Mx32 ビット（256 MB）、4x16Mx16ビット（128 MB）、4x16Mx8ビット（64 MB）
- ワード、ハーフワード、バイトアクセス
- 供給可能な SDRAM クロックは fmc_ker_ck/2 または fmc_ker_ck/3
- Rowおよびバンクの自動境界管理
- マルチバンクピンポンアクセス
- プログラム可能なタイミングパラメータ
- プログラム可能なリフレッシュレートでのオートリフレッシュ動作
- セルフリフレッシュモード
- パワーダウンモード
- ソフトウェアによる SDRAM パワーアップ初期化
- CAS 遅延 = 1、2、3
- 深さ 6 ライン * 32 ビット（6x14 ビットアドレスタグ）のキャッシュ可能読み出し FIFO

22.9.2 SDRAM 外部メモリインタフェース信号

FMC SDRAM コントローラと外部SDRAM デバイスとのインタフェースに使用される SDRAM I/O ピンは、起動時にユーザーアプリケーションから設定する必要があります。アプリケーションで使用しない SDRAM コントローラ I/O ピンは、他の用途に使用することができます。

表 180. SDRAM 信号

SDRAM 信号	I/O タイプ	説明	オルタネート機能
SDCLK	O	SDRAM クロック	-
SDCKE[1:0]	O	SDCKE0：SDRAM バンク 1 クロック有効化 SDCKE1：SDRAM バンク 2 クロック有効化	-
SDNE[1:0]	O	SDNE0：SDRAM バンク 1 チップ有効化 SDNE1：SDRAM バンク 2 チップ有効化	-
A[12:0]	O	アドレス	FMC_A[12:0]
D[31:0]	I/O	双方向データバス	FMC_D[31:0]
BA[1:0]	O	バンクアドレス	FMC_A[15:14]
NRAS	O	Rowアドレスストローブ	-
NCAS	O	Columnアドレスストローブ	-
SDNWE	O	書き込みイネーブル	-
NBL[3:0]	O	書き込みアクセス用の出力バイトマスク （メモリ信号名：DQM[3:0]	FMC_NBL[3:0]

22.9.3 SDRAM コントローラの機能説明

すべての SDRAM コントローラ出力（信号、アドレス、データ）は、メモリクロック（FMC_SDCLK）の立ち下がりエッジで変化します。

SDRAM 初期化

初期化シーケンスはソフトウェアから管理されます。2 つのバンクを使用する場合、FMC_SDCMR レジスタの対象バンクビット CTB1 と CTB2 をセットして、初期化シーケンスをバンク 1 とバンク 2 に対して同時に発生させる必要があります。

1. FMC_SDCRx レジスタにメモリデバイス機能を設定します。SDRAM クロック周波数、RBURST、RPIPEは、FMC_SDCR1 レジスタに設定する必要があります。
2. メモリデバイスタイミングは FMC_SDTRx レジスタに設定します。TRP タイミングと TRC タイミングは、FMC_SDTR1 レジスタに設定します。
3. MODE ビットに"001"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）を設定して、メモリに対するクロックの供給を開始させます（SDCKE はハイに駆動されます）。
4. 所定の遅延時間を待ちます。代表的遅延時間は約 100 ?s です（パワーアップ後に必要とされる遅延については、SDRAM データシート参照）。
5. MODE ビットに"010"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）を設定して、「Precharge All」コマンドを発行させます。
6. MODE ビットに"011"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）、さらに連続オートリフレッシュコマンド数（NRFS）を設定します。発行が必要なオートリフレッシュコマンド数については SDRAM データシートを参照してください。代表値は 8 です。
7. MRDフィールドを設定し、MODEビットに 100 をセットし、FMC_SDCMR レジスタのターゲットバンクビット（CTB1 および／または CTB2）を設定することで、ロードモードレジスタコマンドを発行させ、SDRAM デバイスをプログラムします。特に、バースト長（BL）を 1 に設定することと、CAS 遅延を選択することは必須です。2 つの SDRAM バンクのモードレジスタが一致しない場合、このステップを 2 回繰り返し（各バンクに 1 回）、その結果に応じてターゲットバンクビットをセットします。モバイル SDRAM デバイスの場合、MRDフィールドは、ロードモードレジスタコマンドを発行すると同時に、拡張モードレジスタを設定するためにも使用されます。
8. リフレッシュレートを FMC_SDRTR レジスタに設定します。

   リフレッシュレートは、リフレッシュサイクル間の遅延に対応します。この値を SDRAM デバイスに対して使用する必要があります。

この段階で、SDRAM デバイスはをコマンドを受け付ける準備が整います。SDRAM アクセスの実行中にシステムリセットが発生した場合、データバスはそのまま SDRAM デバイスによって駆動される可能性があります。従って、SDRAM デバイスはリセット後、NOR フラッシュ、PSRAM、SRAM、NAND フラッシュコントローラのいずれかで新しいアクセスを発行する前に、再初期化をする必要があります。

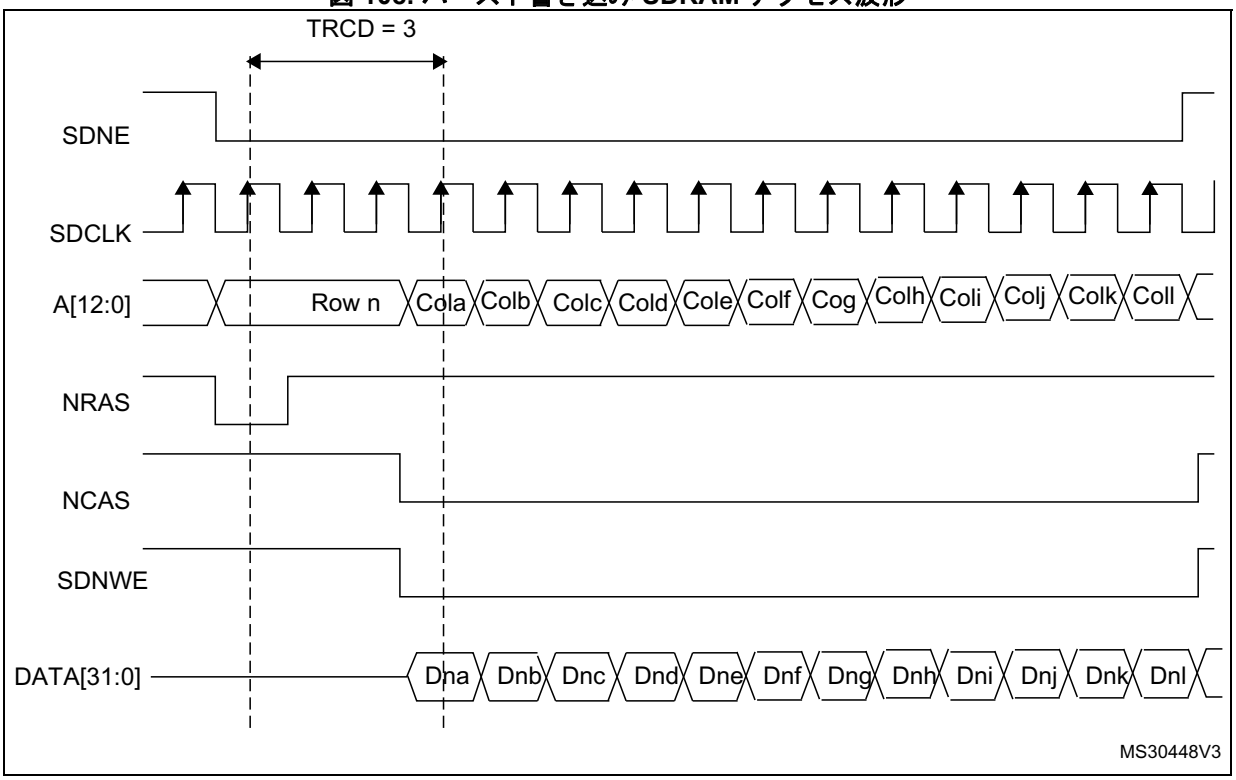
注： ***2 つの SDRAM デバイスを FMC に接続する場合は、コマンドモードレジスタにより両デバイスに同時に実行されるすべてのアクセス（ロードモードレジスタコマンド）は、FMC_SDTR1 レジスタで SDRAM バンク 1 に対して設定されたタイミングパラメータ（TMRD タイミング、TRAS タイミング）を使って発行されます。***

SDRAM コントローラ書き込みサイクル

SDRAM コントローラはシングルおよびバースト書き込みリクエストを受け付けて、これらをシングルメモリアクセスに変換します。両ケースとも、SDRAM コントローラはアクティブRowを記録して、各バンクが異なるバンクへ連続書き込みアクセスを行えるようにします（マルチバンクピンポンアクセス）。

書き込みアクセスを行う前に、FMC_SDCRx レジスタの WP ビットをクリアして、SDRAM バンク書き込み保護機能を無効化しておく必要があります。

図 108. バースト書き込み SDRAM アクセス波形



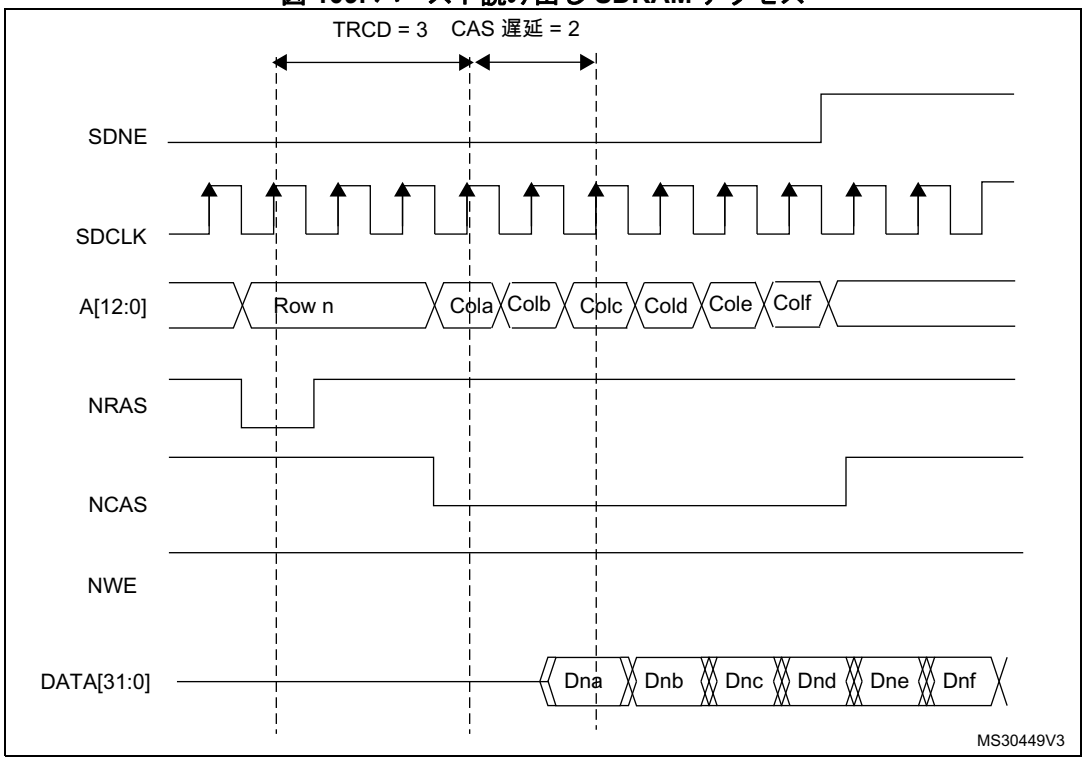
SDRAM コントローラは常に次のアクセスをチェックしています。

- 次のアクセスが同じRowまたは別のアクティブなRowである場合、書き込み動作が実行されます。
- 次のアクセスが別のRow（非アクティブ）を対象とする場合、SDRAM コントローラはプリチャージコマンドを発生し、新しいRowをアクティブにして、書き込みコマンドを開始します。

SDRAM コントローラ読み出しサイクル

SDRAM コントローラはシングルおよびバースト読み出しリクエストを受け付けて、これらをシングルメモリアクセスに変換します。両ケースとも、SDRAM コントローラはアクティブRowを記録して、各バンクが異なるバンクへ連続読み出しアクセスを行えるようにします（マルチバンクピンポンアクセス）。

図 109. バースト読み出し SDRAM アクセス



FMC SDRAM コントローラは、キャシュ可能読み出し FIFO（6 ライン x 32 ビット）を内蔵しています。これは、CAS レイテンシ中（最大 3 メモリクロックサイクル、FMC_SDCRx に設定）および RPIPE 遅延中（最大 2xfmc_ker_ck クロックサイクル、FMC_SDCR1 に設定）に、あらかじめ読み出しておいたデータを保存するために使われます。CAS 遅延 + 1 + (RPIPE DIV2) という式を使用します。次の読み出しアクセスを予測するときは、FMC_SDCR1 レジスタのRBURST ビットをセットする必要があります。

例：

- CAS = 3、RPIPE = 2 x fmc_ker_ck。この場合、5 つのデータ（コミットされていない）が FIFO に保存されます（CASレイテンシ時に 4 データと RPIPE遅延時に 1 データ）。
- CAS = 3、RPIPE = 1 x fmc_ker_ck。この場合、4 つのデータ（コミットされていない）が FIFO に保存されます（CAS レイテンシ時の 4 データ）。

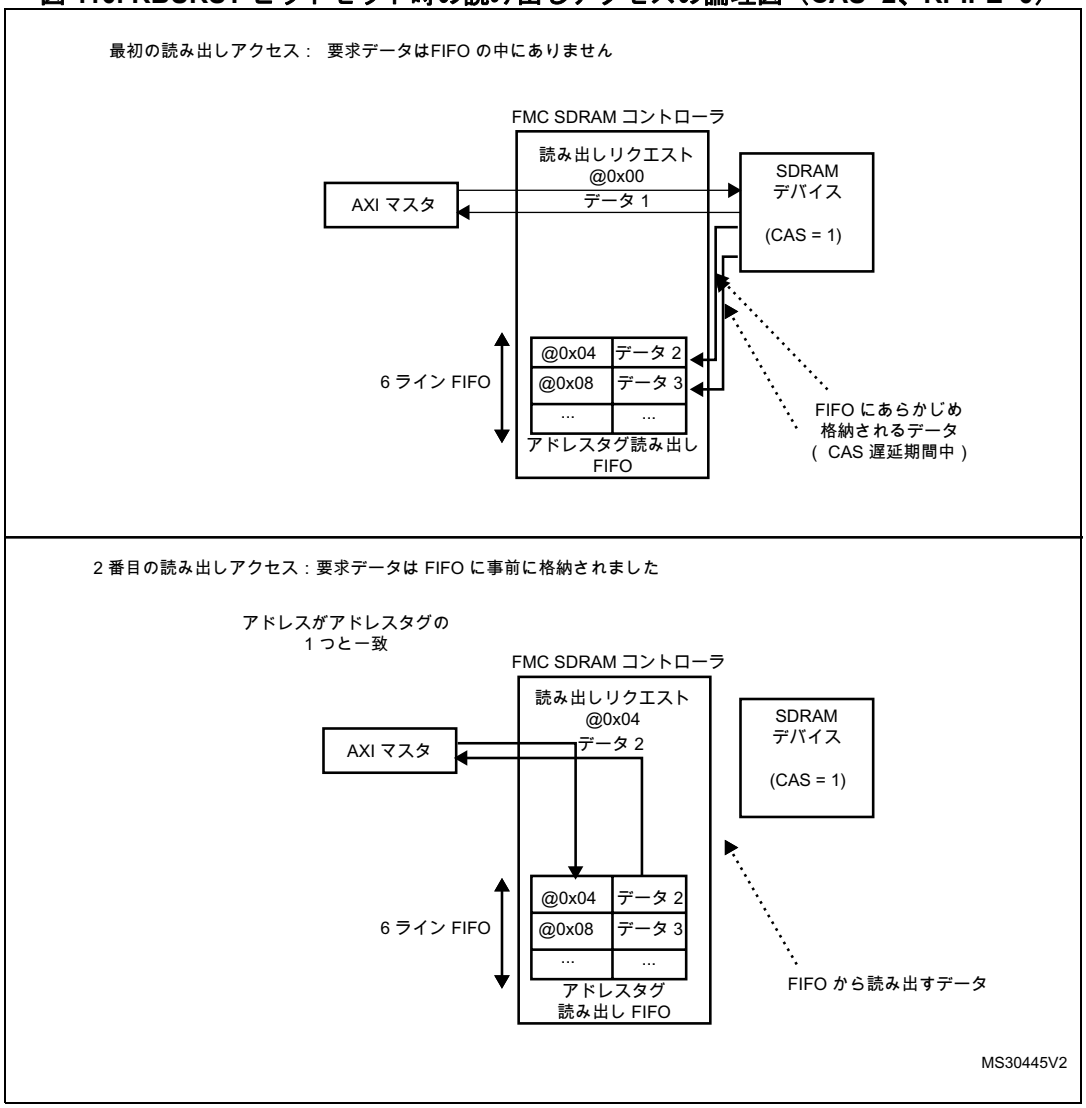
読み出し FIFO は、各ラインに対して値を識別する 14 ビットのアドレスタグを持っています。11ビットはColumnアドレスに、2 ビットは内部バンクとアクティブRowの選択に、1 ビットは SDRAM デバイスの選択に、それぞれ使用します。

バースト読み出しトランザクション時にすでにRowの終わりに到達している場合、あらかじめ読み出されたデータ（コミットされていない）は読み出し FIFO には保存されません。シングル読み出しアクセスの場合、データは正常に FIFOに保存されます。

読み出しリクエストが発生するたびに、SDRAM コントローラは次をチェックします。

- アドレスがアドレスタグの 1 つに一致すると、データが直接 FIFO から読み出され、対応するアドレスタグ／ライン値がクリアされ、FIFO 内の残りのデータはエンプティラインを回避するためコンパクト化されます。
- その他の場合、新しい読み出しコマンドがメモリに対して発行され、FIFO は新しいデータで更新されます。FIFO がフルの場合、古いデータは失われます。

図 110. RBURST ビットセット時の読み出しアクセスの論理図（CAS=2、RPIPE=0）



書き込みアクセス時またはプリチャージコマンド時、読み出し FIFO が一掃されて、新しいデータ書き込みの準備が整います。

最初の読み出しリクエストの後、現在のアクセスがRow境界に対して実行されなかった場合、SDRAM コントローラは、CAS遅延中および RPIPE遅延中（設定した場合）に次の読み出しアクセスがあるものと見なします。これは、メモリアドレスをインクリメントして行います。次の条件を満たす必要があります。

- FMC_SDCR1 レジスタの RBURST 制御ビットを”1”にセットする必要があります。

アドレスは以下に示す AXI リクエストに応じて管理されます。

- 次のリクエストがシーケンシャル（バーストアクセス）である。
  この場合、SDRAM コントローラはアドレスをインクリメントさせます。
- 次のリクエストはシーケンシャルでない。
  - 新しい読み出しリクエストが同じRowまたは別のアクティブなRowを対象とする場合、新しいアドレスがメモリへ渡され、マスタは CAS 遅延の間停止され、メモリからの新しいデータを待ちます。
  - 新しい読み出しリクエストがアクティブなRowを対象としない場合、SDRAM コントローラはプリチャージコマンドを発生し、新しいRowをアクティブにして、読み出しコマンドを開始します。

RBURST がリセットされると、読み出し FIFO は使用されません。

Rowおよびバンク境界管理

読み出しアクセスまたは書き込みアクセスがRow境界を跨ぐとき、次の読み出しアクセスまたは書き込みアクセスがシーケンシャルで、かつ現在のアクセスがRow境界に対して実行された場合、SDRAM コントローラは次の動作を行います。

1. アクティブRowのプリチャージ
2. 新しいRowのアクティブ化
3. 読み出し／書き込みコマンドの開始

Row境界で、すべてのColumnとデータバス幅設定に対して次のRowの自動アクティブ化をサポート

必要に応じて、SDRAM コントローラは次のようにコマンドの間に追加クロックサイクルを挿入します。

- プリチャージコマンドとアクティブコマンドの間に挿入して TRP パラメータを一致させます（次のアクセスが同じバンクの異なるRowの場合のみ）
- アクティブコマンドと読み出しコマンドの間に挿入して TRCD パラメータを一致させます。

これらのパラメータは FMC_SDTRx レジスタで定義されます。

Row境界を跨ぐ読み出しアクセスとバースト書き込みアクセスについては、*図 108* と *図 109* を参照してください。

参考資料

図 111. Row境界を跨ぐ読み出しアクセス

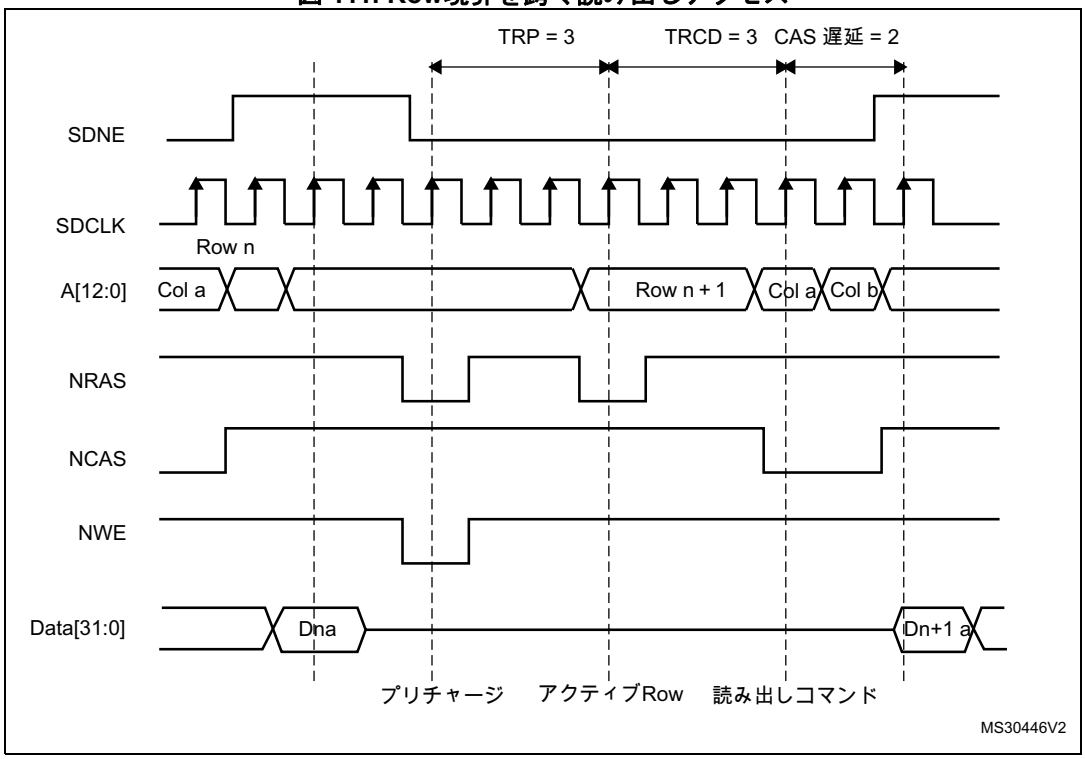
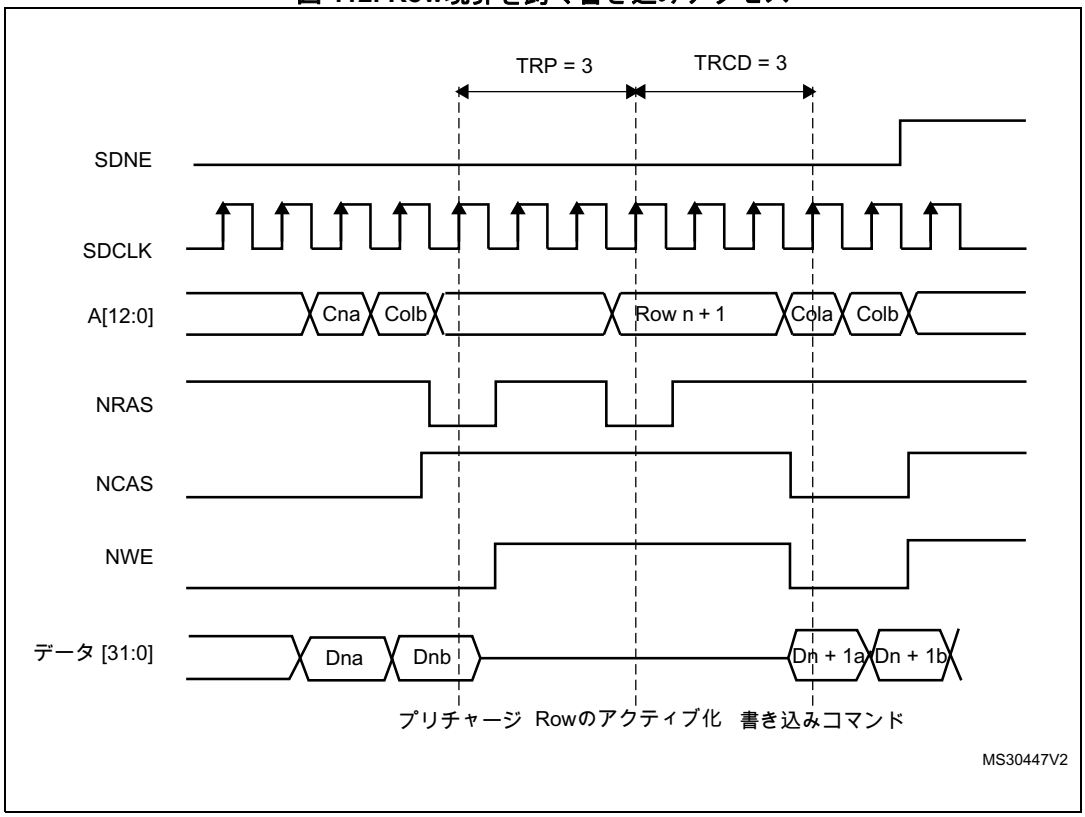


図 112. Row境界を跨ぐ書き込みアクセス



次のアクセスがシーケンシャルで、かつ現在のアクセスがバンク境界を跨ぐ場合、SDRAM コントローラは次のバンクの最初のRowをアクティブ化して、新しい読み出し／書き込みコマンドを開始します。次の 2 つのケースが可能です。

- 現在のバンクが直前のバンクでない場合、新しいバンク内のアクティブなRowをプリチャージする必要があります。バンク境界で、すべてのRow／Columnとデータバス幅設定に対して次のRowの自動アクティブ化をサポートしています。
- 現在のバンクが直前のバンクである場合、次のRowの自動アクティブ化は、13 ビットRow、11 ビットColumn、4 つの内部バンク、32 ビットデータバスの SDRAM デバイスをアドレス指定する場合にのみサポートされます。そうしないと、SDRAM アドレスレンジに違反するため、AXI スレーブエラーが発生します。
- 13 ビットRowアドレス、11 ビットColumnアドレス、4 つの内部バンク、バス幅 32 ビットの SDRAM メモリでは、バンク境界で、SDRAM コントローラは 2 番目の SDRAM デバイスに対する読み出し／書き込みを続けます（ただし初期化済みとします）。
  a) 最初の内部バンク内にアクティブなRowが既に存在している場合、SDRAM コントローラはアクティブなRowをプリチャージした後に最初のRowをアクティブ化し、新しい読み出し／書き込みコマンドを開始します。
  b) 最初のRowが既にアクティブ化されている場合、SDRAM コントローラは読み出し／書き込みコマンドを開始するのみです。

SDRAM コントローラリフレッシュサイクル

オートリフレッシュコマンドは、SDRAM デバイスの値をリフレッシュするときに使います。SDRAM コントローラは、オートリフレッシュコマンドを周期的に発行します。内部カウンタに、FMC_SDRTR レジスタの COUNT 値がロードされます。この値は、リフレッシュサイクル間のメモリクロックサイクル数を決定します（リフレッシュレート）。このカウンタがゼロになると、内部パルスが発生されます。

メモリアクセスが進行中である場合、オートリフレッシュリクエストは遅延されます。ただし、メモリアクセスとオートリフレッシュリクエストが同時に発生した場合は、オートリフレッシュリクエストが優先されます。

メモリアクセスがオートリフレッシュ動作中に発生すると、リクエストはバッファされて、オートリフレッシュが完了した後に処理されます。

前のオートリフレッシュリクエストの処理中に、新しいオートリフレッシュリクエストが発生すると、ステータスレジスタの RE（リフレッシュエラー）ビットがセットされます。有効化されている場合（REIE = "1"）、割り込みが発生します。

SDRAM ラインがアイドル状態でない（すべてのRowが終了していない）場合、SDRAM コントローラはオートリフレッシュの前に PALL（プリチャージ ALL）コマンドを発生します。

FMC_SDCMR コマンドモードレジスタからオートリフレッシュコマンドが発生されると（MODE ビット = "011"）、最初に PALL コマンド（MODE ビット = "010"）を発行する必要があります。

22.9.4 低電力モード

次の 2 つの低消費電力モードがあります。

- セルフリフレッシュモード

  SDRAM デバイス自体によりオートリフレッシュサイクルが実行されて、外部クロックなしでデータを保持します。
- パワーダウンモード

  オートリフレッシュサイクルは SDRAM コントローラにより実行されます。

セルフリフレッシュモード

MODE ビットに"101"を設定し、かつ FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定すると、このモードが選択されます。

SDRAM クロックは TRAS遅延後に停止し、内部リフレッシュタイマは次の条件をみたすときにカウントを停止します。

- セルフリフレッシュコマンドが両デバイスに発行される。
- 一方のデバイスは非アクティブである（SDRAM バンクが初期化されていない）。

セルフリフレッシュモードになる前に、SDRAM コントローラは PALL コマンドを自動的に発行します。

書き込みデータ FIFO がノットエンプティの場合、すべてのデータがメモリに送られた後にセルフリフレッシュモードが開始され、BUSY ステータスフラグはセットされたままになります。

セルフリフレッシュモードでは、ローを維持する SDCKE 以外のすべての SDRAM デバイス入力が無視されます。

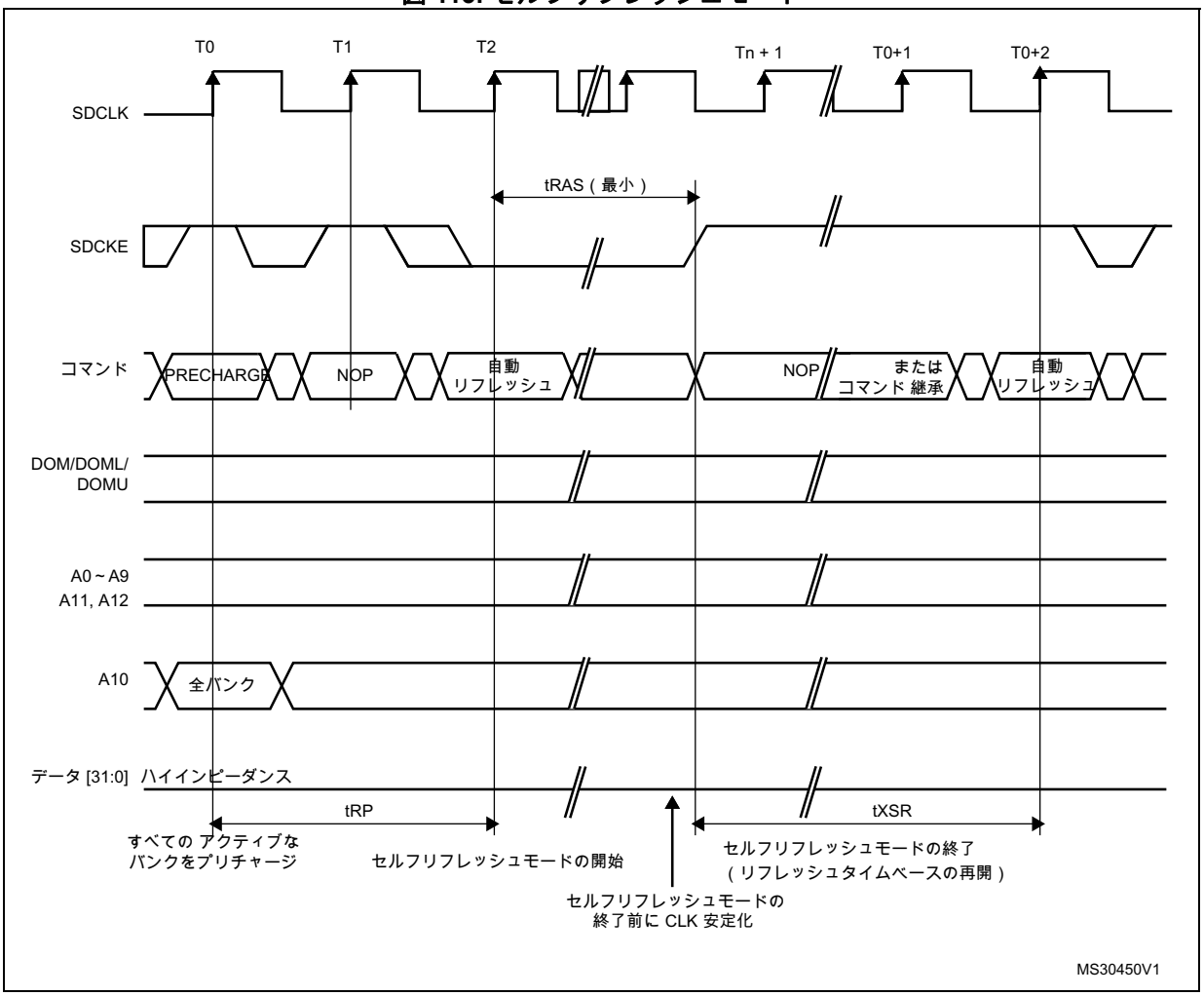
SDRAM デバイスは最小時間TRAS の間セルフリフレッシュモードに留まる必要があり、無限時間セルフリフレッシュモードを維持することもできます。この最小時間を保証するため、TRAS遅延中にセルフリフレッシュを開始した後、BUSY ステータスフラグはハイを維持します。

SDRAM デバイスを選択すると直ちに、SDRAM コントローラはセルフリフレッシュモードを終わらせる一連のコマンドを発生します。メモリアクセスの後、選択されたデバイスは通常モードに留まります。

セルフリフレッシュを終了するときは、MODE ビットに"000"（通常モード）を設定し、FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定する必要があります。

参考資料

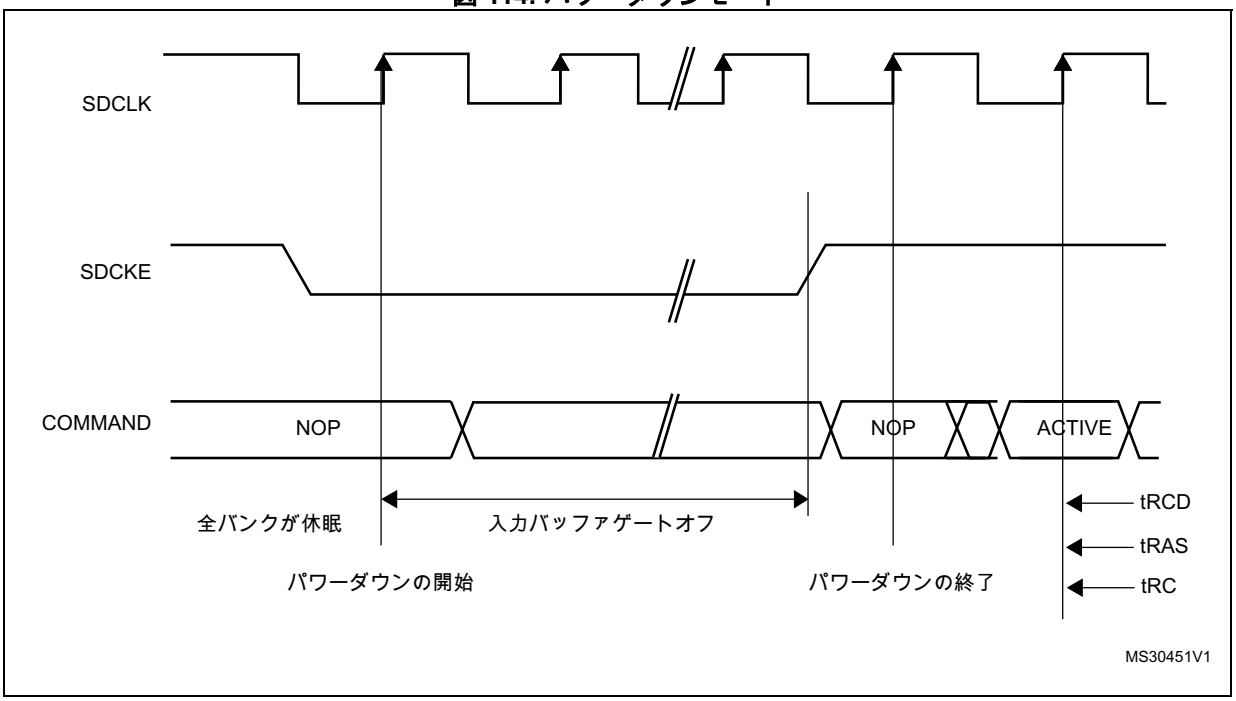
図 113. セルフリフレッシュモード



パワーダウンモード

MODE ビットに"110"を設定し、かつ FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定すると、このモードが選択されます。

図 114. パワーダウンモード



書き込みデータ FIFO がノットエンプティの場合、すべてのデータがメモリに送られた後にパワーダウンモードが開始されます。

SDRAM デバイスが選択されると直ちに、SDRAM コントローラはパワーダウンモードを終了します。メモリアクセスの後、選択されたデバイスは通常モードに留まります。

パワーダウンモード時、ローを維持する SDCKE 以外のすべての SDRAM デバイス入力と出力バッファは非アクティブになります。

SDRAM デバイスはリフレッシュ周期より長い間パワーダウンモードに留まることができないため、自分自身でオートリフレッシュサイクルを実行できません。したがって、SDRAM コントローラは次の動作によりリフレッシュ動作を実行します。

1. パワーダウンモードを終了し、SDCKE をハイに駆動します。
2. パワーダウンモード中にRowがアクティブであった場合にのみ PALL コマンドを発生します。
3. オートリフレッシュコマンドを発生します。
4. 再度 SDCKE をローに駆動し、パワーダウンモードに戻ります。

パワーダウンモードを終了するときは、MODE ビットに"000"（通常モード）を設定し、FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定する必要があります。

22.9.5 SDRAM コントローラレジスタ

SDRAM 制御レジスタ 1、2（FMC_SDCR1、2）

アドレスオフセット：0x140+ 4* (x – 1), x = 1,2

リセット値：0x0000 02D0

このレジスタは各 SDRAM メモリバンクの制御パラメータを格納します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15
Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC51:0]		Res.
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **RPIPE[1:0]：読み出しパイプ**

これらのビットは、CAS レイテンシ後のデータ読み出しに対して、fmc_ker_ck クロックサイクル数で遅延を定義します。

00：fmc_ker_ck クロックサイクルの遅延はありません。

01：1 fmc_ker_ck クロックサイクルの遅延があります。

10：2 fmc_ker_ck クロックサイクルの遅延があります。

11：予約済み

注： FMC_SDCR2 レジスタの対応するビットは読み出し専用です。

ビット 12 **RBURST：バースト読み出し**

このビットはバースト読み出しモードを有効化します。SDRAM コントローラは CAS遅延中に次の読み出しコマンドを予測し、データを読み出し FIFO に保存します。

0：シングル読み出しリクエストをバーストとして管理しません。

1：シングル読み出しリクエストを常にバーストとして管理します。

注： FMC_SDCR2 レジスタの対応するビットは読み出し専用です。

ビット 11:10 **SDCLK[1:0]：**SDRAM クロック設定

これらのビットは両 SDRAM バンクの SDRAM クロック周期を指定し、周波数の変更前にクロックの無効化を可能にします。この場合、SDRAM の再初期化が必要です。

00：SDCLK クロックを無効化

01：予約済み

10：SDCLK 周期 = 2 x fmc_ker_ck 周期

11：SDCLK 周期 = 3 x fmc_ker_ck 周期

注： FMC_SDCR2 レジスタの対応するビットは読み出し専用です。

ビット 9 **WP：**書き込み保護機能

このビットは、SDRAM バンクに対する書き込みモードアクセスを有効化します。

0：書き込みアクセスを許可

1：書き込みアクセスを無視

ビット 8:7 **CAS[1:0]：CAS 遅延**

このビットは、メモリクロックサイクル数で SDRAM CAS遅延を設定します。
00：予約済み
01：1 サイクル
10：2 サイクル
11：3 サイクル

ビット 6 **NB：内部バンク数**

このビットは内部バンク数を設定します。
0：2 内部バンク
1：4 内部バンク

ビット 5:4 **MWID[1:0]：メモリデータバス幅。**

これらのビットはメモリデバイス幅を指定します。
00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

ビット 3:2 **NR[1:0]：Row アドレスビット数**

これらのビットは、Rowアドレスのビット数を指定します。
00：11 ビット
01：12 ビット
10：13 ビット
11：予約済み

ビット 1:0 **NC[1:0]：Column アドレスビット数**

これらのビットは、Columnアドレスのビット数を指定します。
00：8 ビット
01：9 ビット
10：10 ビット
11：11 ビット。

注： ***RBURST または RPIPE の設定値を変更する前、または SDCLK クロックを無効化する前に、先に PALL コマンドを送信して実行中の動作の完了を確認する必要があります。***

SDRAM タイミングレジスタ 1、2（FMC_SDTR1、2）

アドレスオフセット：0x148 + 4 * (x – 1), x = 1,2

リセット値：0x0FFF FFFF

このレジスタは各 SDRAM バンクのタイミングパラメータを格納します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TRCD				TRP				TWR			
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRC				TRAS				TXSR				TMRD			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TRCD[3:0]：Row から Column までの遅延**

これらのビットは、アクティブコマンドと読み出し／書き込みコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

ビット 23:20 **TRP[3:0]：Row プリチャージ遅延**

これらのビットは、プリチャージコマンドと別のコマンドとの間の遅延をメモリクロックサイクル数で指定します。TRP タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRP には低速側デバイスのタイミングを設定する必要があります。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

注： FMC_SDTR2 レジスタの対応するビットは 無視されます。

ビット 19:16 **TWR[3:0]：復帰遅延**

これらのビットは、書き込みコマンドとプリチャージコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

注： TWR は SDRAM のデータシートで規定される書き込み復帰時間(t_{WR})に一致するようにプログラムし、次を保証する必要があります。

TWR ³ TRAS - TRCD、かつ TWR ³TRC - TRCD - TRP

例 : TRAS= 4 サイクル、TRCD= 2 サイクル。したがって、TWR >= 2 サイクル。TWR には 0x1 を設定する必要があります。

2 つの SDRAM デバイスを使用する場合、FMC_SDTR1 および FMC_SDTR2 には、より低速な SDRAM デバイスと同じ TWR タイミングを設定する必要があります。

ビット 15:12 **TRC[3:0]：Row サイクル遅延**

これらのビットは、リフレッシュコマンドとアクティブコマンドとの間の遅延、および連続する 2 つのリフレッシュコマンドの間の遅延をそれぞれ定義します。メモリクロックサイクル数で表されます。TRC タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRC には低速側デバイスのタイミングを設定する必要があります。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

注： TRC は SDRAM デバイスデータシートで規定される TRC と TRFC（オートリフレッシュ周期）タイミングに一致する必要があります。

注： FMC_SDTR2 レジスタの対応するビットは 無視されます。

ビット 11:8 **TRAS[3:0]：セルフリフレッシュ時間**

これらのビットは、メモリクロックサイクルで最小セルフリフレッシュ周期を指定します。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

ビット 7:4 **TXSR[3:0]：セルフリフレッシュ遅延の終了**

これらのビットは、セルフリフレッシュコマンドの解除からアクティブコマンド発行までの遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

注： 2 つの SDRAM デバイスを使用する場合、FMC_SDTR1 および FMC_SDTR2 には、より低速な SDRAM デバイスと同じ TXSR タイミングを設定する必要があります。

ビット 3:0 **TMRD[3:0]：ロードモードレジスタのアクティブまでの時間**

これらのビットは、ロードモードレジスタコマンドとアクティブまたはリフレッシュコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

....

1111：16 サイクル

注： 2 つの SDRAM デバイスを接続する場合は、コマンドモードレジスタにより両デバイスに同時に実行されるすべてのアクセス（ロードモードレジスタコマンド）は、FMC_SDTR1 レジスタでバンク 1 に対して設定されたタイミングパラメータ（TMRD タイミング、TRAS タイミング）を使って発行されます。

TRP タイミングと TRC タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRP と TRC には低速側デバイスのタイミングを設定する必要があります。

SDRAM コマンドモードレジスタ（FMC_SDCMR）

アドレスオフセット：0x150

リセット値：0x0000 0000

このレジスタは、SDRAM デバイスをアクセスするときに発行されたコマンドを格納します。このレジスタを使って、SDRAM デバイスの初期化、セルフリフレッシュの開始、パワーダウンモードの開始を行います。MODE フィールドに書き込みを行うと直ちに、CTB1 コマンドビットと CTB2 コマンドビットに応じて一方または両方の SDRAM バンクへコマンドが発行されます。このレジスタは、両 SDRAM バンクに対して同じです。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MRD							NRFS				CTB1	CTB2	MODE		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 **予約済みであり、リセット値に保持する必要があります。**

ビット 22:9 **MRD[13:0]：モードレジスタの定義**

この 14 ビットフィールドは、SDRAM モードレジスタ値を指定します。モードレジスタは、ロードモードレジスタコマンドを使って設定します。MRD[13:0] ビットは、モバイル SDRAM 用に拡張モードレジスタをプログラムするためにも使用されます。

ビット 8:5 **NRFS[3:0]：オートリフレッシュ数**

これらのビットは、MODE = "011"のときに発行される連続オートリフレッシュコマンド数を指定します。

0000：1オートリフレッシュサイクル

0001：2オートリフレッシュサイクル

....

1110：15オートリフレッシュサイクル

1111：16オートリフレッシュサイクル

ビット 4 **CTB1：コマンドターゲットバンク 1**

このビットは、SDRAM バンク 1 へコマンドを発行するか否かを指定します。

0：SDRAM バンク 1 へコマンドを発行しない

1：SDRAM バンク 1 へコマンドを発行する

ビット 3 **CTB2：コマンドターゲットバンク 2**

このビットは、SDRAM バンク 2 へコマンドを発行するか否かを指定します。

0：SDRAM バンク 2 へコマンドを発行しない

1：SDRAM バンク 2 へコマンドを発行する

ビット 2:0 **MODE[2:0]**：コマンドモード

これらのビットは、SDRAM デバイスへ発行するコマンドを指定します。

000：通常モード

001：クロック設定イネーブル

010：PALL（「全バンクプリチャージ」）コマンド

011：オートリフレッシュコマンド

100：ロードモードレジスタ

101：セルフリフレッシュコマンド

110：パワーダウンコマンド

111：予約済み

注： ***コマンドを発行する場合、少なくとも 1 つのコマンドターゲットバンクビット（CTB1 または CTB2）をセットする必要があります。そうしないと、そのコマンドは無視されます。***

注： ***2 つの SDRAM バンクを使用する場合、オートリフレッシュコマンドおよび PALL コマンドを CTB1 および CTB2 ビットがセットされた 2 つにデバイスに対して同時に発行する必要があります。そうしないと、そのコマンドは無視されます。***

注： ***1 つの SDRAM バンクのみを使用し、該当する CTB ビットをセットした状態でコマンドを発行する場合、もう一方の使用されていないバンクの CTB ビットを 0 に保持する必要があります。***

SDRAM リフレッシュタイマレジスタ（FMC_SDRTR）

アドレスオフセット：0x154

リセット値：0x0000 0000

このレジスタは、リフレッシュタイマカウント値を設定することにより、リフレッシュサイクル間の SDCLK クロックサイクル数でリフレッシュレートを設定します。

$$\text{Refresh rate} = (\text{COUNT} + 1) \times \text{SDRAM clock frequency}$$

$$\text{COUNT} = (\text{SDRAM refresh period} / \text{Number of rows}) - 20$$

例

$$\text{Refresh rate} = 64\ \text{ms} / (8196\text{rows}) = 7.81\mu\text{s}$$

ここで、64 ms は SDRAM リフレッシュ周期です。

$$7.81\mu\text{s} \times 60\text{MHz} = 468.6$$

読み出しリクエストが受理されているときに内部リフレッシュリクエストが発生した場合には、安全マージンを確保するためリフレッシュレートを 20 SDRAM クロックサイクルだけ大きくする必要があります（上の例の場合）。これは、COUNT 値"0000111000000"（448）に対応します。

この 13 ビットフィールドは、SDRAM クロックを使ってデクリメントされるタイマにロードされます。このタイマはゼロになると、リフレッシュパルスを発生します。COUNT 値は、少なくとも 41SDRAM クロックサイクルである必要があります。

FMC_SDRTR レジスタが設定されると直ちに、タイマはカウントを開始します。レジスタに設定された値が "0"の場合、リフレッシュは実行されません。初期化の後にこのレジスタを再設定して、リフレッシュレートが変更されるのを回避する必要があります。

リフレッシュパルスが発生されるたびに、この 13 ビット COUNT フィールドがカウンタへ再ロードされます。

メモリアクセスが進行中の場合、オートリフレッシュリクエストは遅延させられます。ただし、メモリアクセスとオートリフレッシュリクエストが同時に発生した場合は、オートリフレッシュリクエストが優先されます。メモリアクセスがリフレッシュ動作中に発生すると、リクエストはバッファされて、リフレッシュが完了した後に処理されます。

このレジスタは、SDRAM バンク 1 と SDRAM バンク 2 に共通です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	REIE	COUNT													CRE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **REIE**：RES 割り込み有効化

0：割り込みは無効です

1：RE = 1 の場合に割り込みが発生します

ビット 13:1 **COUNT[12:0]**：リフレッシュタイマカウント

この 13 ビットフィールドは、SDRAM デバイスのリフレッシュレートを指定します。メモリクロックサイクル数で表されます。この値は、少なくとも 41SDRAM クロックサイクル (0x29) である必要があります。

リフレッシュレート = (COUNT + 1) x SDRAM クロック周波数

COUNT = (SDRAM リフレッシュ周期 / Row数) - 20

ビット 0 **CRE**：リフレッシュエラーフラグのクリア

このビットを使って、ステータスレジスタのリフレッシュエラーフラグ（RE）をクリアします。

0：影響なし。

1：リフレッシュエラーフラグをクリア

注： ***プログラムされた COUNT 値は、以下で求めるタイミングの合計とイコールであってはなりません。TWR + TRP + TRC + TRCD + 4 メモリクロックサイクル***

SDRAM ステータスレジスタ（FMC_SDSR）

アドレスオフセット：0x158

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODES2		MODES1		RE
											r	r	r	r	r

ビット 31:5 **予約済みであり、リセット値に保持する必要があります。**

ビット 4:3 **MODES2：バンク 2 のステータスモード**

これらのビットは、SDRAM バンク 2 のステータスモードを定義します。
00：通常モード
01：セルフリフレッシュモード
10：パワーダウンモード

ビット 2:1 **MODES1：バンク 1 のステータスモード**

これらのビットは、SDRAM バンク 1 のステータスモードを定義します。
00：通常モード
01：セルフリフレッシュモード
10：パワーダウンモード

ビット 0 **RE：リフレッシュエラーフラグ**

0：リフレッシュエラーは検出されない
1：リフレッシュエラーが検出された
REIE = 1 かつ RE = 1 の場合割り込みが発生します。

22.10 FMC レジスタマップ

次の表に FMC レジスタの一覧を示します。

表 181. FMC レジスタマップ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMC_BCR1	FMCEN						BMAP[1:0]				WFDIS	CCLKEN	CBURSTRW	CPSIZE[2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG		WAITPOL	BURSTEN		FACCEN	MWID		MTYP		MUXEN	MBKEN
0x08	**FMC_BCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x10	**FMC_BCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x18	**FMC_BCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x04	**FMC_BTR1**	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0C	**FMC_BTR2**	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	**FMC_BTR3**	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	**FMC_BTR4**	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x104	**FMC_BWTR1**	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x10C	**FMC_BWTR2**	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x114	**FMC_BWTR3**	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x11C	**FMC_BWTR4**	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x80	**FMC_PCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS [2:0]				TAR[3:0]				TCLR[3:0]				Res.	Res.	ECCEN		PWID [1:0]		Res.	PBKEN	PWAITEN	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0			0		0	1	1	0	0	
0x84	**FMC_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
	リセット値																										1	0	0	0	0	0	0

参考資料

表 181. FMC レジスタマップ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	FMC_PMEM	MEMHIZx[7:0]								MEMHOLDx[7:0]								MEMWAITx[7:0]								MEMSETx[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x8C	FMC_PATT	ATTHIZ[7:0]								ATTHOLD[7:0]								ATTWAIT[7:0]								ATTSET[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x94	FMC_ECCR	ECCx[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x140	FMC_SDCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC	
	リセット値																		0	0	0	1	1	0	1	0	0	1	0	0	0	0	0
0x144	FMC_SDCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC	
	リセット値																		0	0	0	1	1	0	1	0	0	1	0	0	0	0	0
0x148	FMC_SDTR1	Res.	Res.	Res.	Res.	TRCD[3:0]				TRP[3:0]				TWR[3:0]				TRC[3:0]				TRAS[3:0]				TXSR[3:0]				TMRD[3:0]			
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14C	FMC_SDTR2	Res.	Res.	Res.	Res.	TRCD[3:0]				TRP[3:0]				TWR[3:0]				TRC[3:0]				TRAS[3:0]				TXSR[3:0]				TMRD[3:0]			
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x150	FMC_SDCMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MRD													NRFS[3:0]				CTB1	CTB2	MODE[2:0]			
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x154	FMC_SDRTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REIE	COUNT[12:0]													CRE	
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x158	FMC_SDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODES2[1:0]		MODES1[1:0]		RE	
	リセット値																												0	0	0	0	0

レジスタ境界アドレスについては *104 ページのセクション 2.2.2* を参照してください。

23 Quad SPI インタフェース（QUADSPI）

23.1 概要

QUADSPI は、シングル、デュアル、またはQuad SPI フラッシュメモリを対象とする特殊な通信インタフェースです。このインタフェースは、次の 3 つのモードのいずれかで動作できます。

- インダイレクトモード：すべての動作は QUADSPI レジスタを使用して実行されます。
- ステータスポーリングモード：外部フラッシュメモリステータスレジスタが定期的に読み出され、フラグのセット時に割り込みを生成することができます。
- メモリマップドモード:外部フラッシュメモリはマイクロコントローラのアドレス空間に配置され、システムによって内部メモリであるかのようにみなされます。

デュアルフラッシュモードを使用すると、スループットと容量の両方を 2 倍に増やすことができます。このモードでは、2 つのQuad SPI フラッシュメモリに同時にアクセスします。

23.2 QUADSPI の主な機能

- 3 つの機能モード：インダイレクト、ステータスポーリング、メモリマップド
- 2 つのフラッシュメモリに並列にアクセスすることで 8 ビットの同時送受信を可能にするデュアルフラッシュモード
- SDR および DDR サポート
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能な OP コード
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能なフレームフォーマット
- 受信および送信用の内蔵 FIFO
- 8、16、32 ビットのデータアクセスを許可
- MDMA の FIFO 閾値および転送完了のためのトリガ生成
- FIFO 閾値、タイムアウト、動作完了、アクセスエラーによる割り込み生成

参考資料

23.3 QUADSPI の機能説明

23.3.1 QUADSPI ブロック図

図 115. QUADSPI ブロック図（デュアルフラッシュモードが無効のとき）

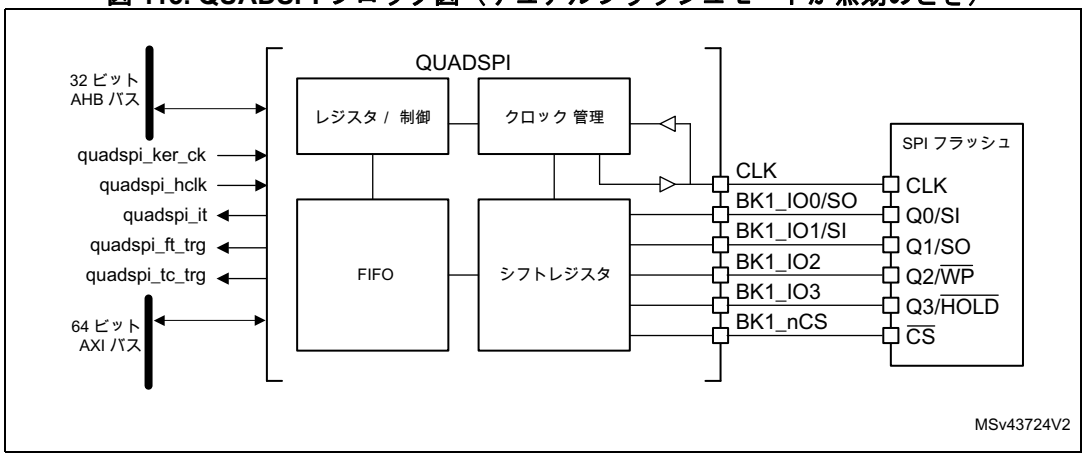
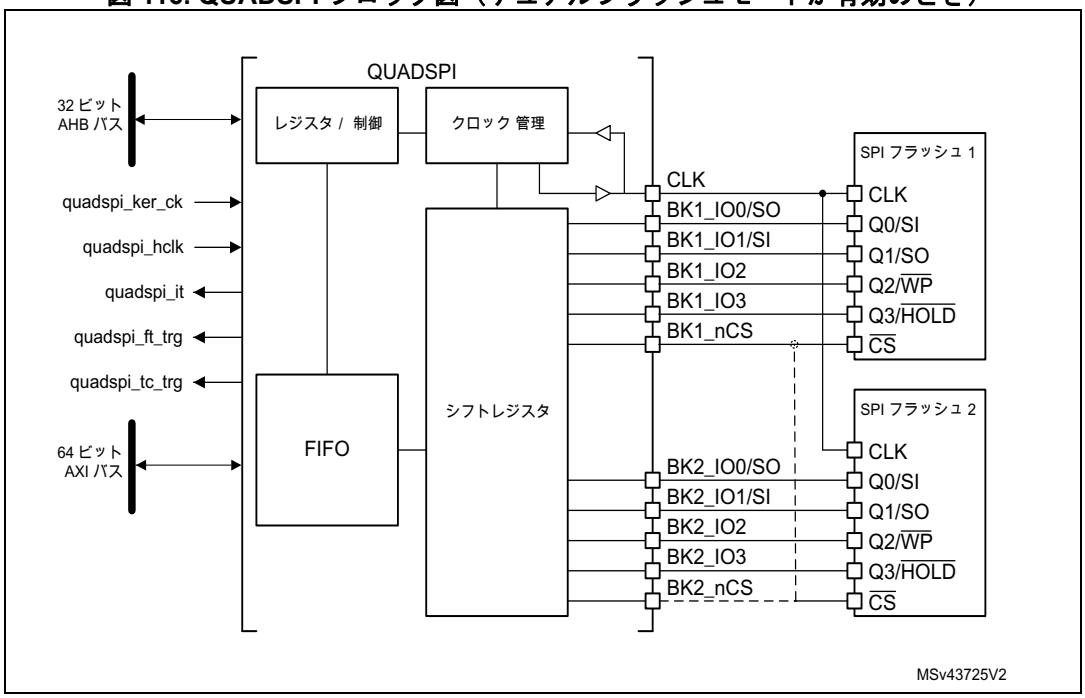


図 116. QUADSPI ブロック図（デュアルフラッシュモードが有効のとき）



23.3.2 QUADSPI ピン および内部信号

表 182 は QUADSPI 内部信号のリストです。

表 182. QUADSPI 内部信号

信号名	信号タイプ	説明
quadspi_ker_ck	デジタル入力	QUADSPI カーネルクロック
quadspi_hclk	デジタル入力	QUADSPI レジスタのインタフェースクロック
quadspi_it	デジタル出力	QUADSPI グローバル割り込み
quadspi_ft_trg	デジタル出力	MDMA の QUADSPI FIFO 閾値トリガ
quadspi_tc_trg	デジタル出力	MDMA の QUADSPI 転送完了トリガ

表 183 では QUADSPI ピンを一覧化しており、そのうち 6 本 は シングル フラッシュメモリとのインタフェース用、10 または 11 本はデュアルフラッシュモードの 2 つのフラッシュメモリ（フラッシュ 1 およびフラッシュ 2）とのインタフェース用です。

表 183. QUADSPI ピン

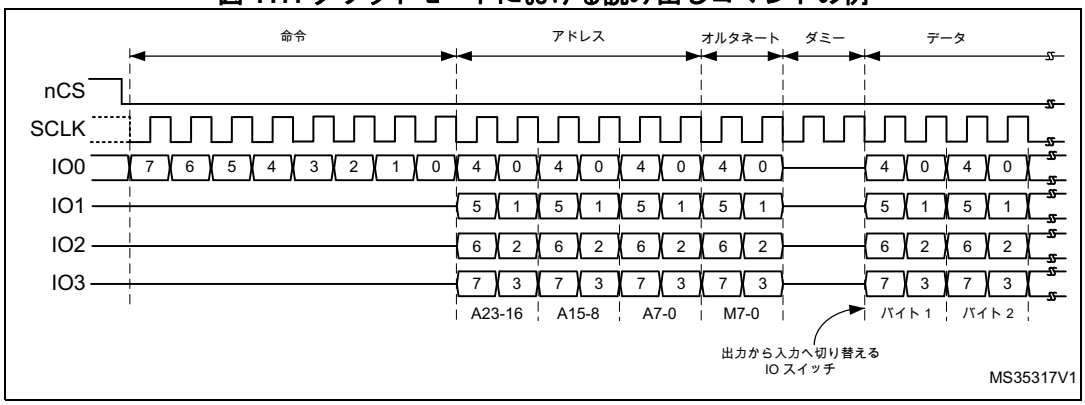
信号名	信号タイプ	説明
CLK	デジタル出力	FLASH 1 および FLASH 2 へのクロック
BK1_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（フラッシュ 1 のみ）
BK1_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（フラッシュ 1 のみ）
BK1_IO2	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 1 のみ）
BK1_IO3	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 1 のみ）
BK2_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（フラッシュ 2 のみ）
BK2_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（フラッシュ 2 のみ）
BK2_IO2	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 2 のみ）
BK2_IO3	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 2 のみ）
BK1_nCS	デジタル出力	チップセレクト出力（アクティブロー）（フラッシュ 1 のみ）。デュアルフラッシュモードで QUADSPI が常時使用される場合は、フラッシュ 2 にも使用可能
BK2_nCS	デジタル出力	チップセレクト出力（アクティブロー）（フラッシュ 2 のみ）。デュアルフラッシュモードで QUADSPI が常時使用される場合は、フラッシュ 1 にも使用可能

23.3.3 QUADSPI コマンドシーケンス

QUADSPI はコマンドを使用してフラッシュメモリと通信します。各コマンドには、命令、アドレス、オルタネートバイト、ダミー、データの 5 つのフェーズが含まれます。これらのフェーズのいずれかをスキップするように設定することができますが、命令、アドレス、オルタネートバイト、データの少なくとも 1 つのフェーズが必要です。

nCS は各コマンドの開始前に立ち下がり、各コマンドの終了後に再び立ち上がります。

図 117. クワッドモードにおける読み出しコマンドの例



命令フェーズ

このフェーズでは、QUADSPI_CCR[7:0] レジスタの INSTRUCTION フィールドに設定された 8 ビットの命令がフラッシュメモリに送信され、実行すべき動作のタイプが指定されます。

ほとんどのフラッシュメモリは IO0/SO 信号（シングル SPI モード）から同時に 1 ビットの命令しか受信できませんが、命令フェーズでは、オプションで、同時に 2 ビット（デュアル SPI モードの IO0/IO1 経由）または 4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）を送信することができます。この設定には、QUADSPI_CCR[9:8] レジスタの IMODE[1:0] フィールドを使用することができます。

IMODE = 00 の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す 1～4 バイトがフラッシュメモリに送信されます。送信されるアドレスバイト数は、QUADSPI_CCR[13:12] レジスタの ADSIZE[1:0] フィールドに設定されます。インダイレクトモードと自動ポーリングモードでは、送信されるアドレスバイトは QUADSPI_AR レジスタの ADDRESS[31:0] フィールドに指定されます。一方、メモリマップドモードの場合は、アドレスはAXI（Cortex® から、または DMA から）経由で直接与えられます。

アドレスフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[11:10] レジスタの ADMODE[1:0] フィールドを使用することができます。

ADMODE = 00 の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作モードを制御する 1～4 バイトがフラッシュメモリに送信されます。送信されるオルタネートバイト数は、QUADSPI_CCR[17:16] レジスタの ABSIZE[1:0] フィールドに設定されます。送信されるバイトは QUADSPI_ABR レジスタで指定されます。

オルタネートバイトフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

ADMODE = 00 の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

場合によっては、オルタネートバイトフェーズ中にフルバイトではなく 2 分の 1 バイト（4 ビット）を送信する必要があるかもしれません。たとえば、デュアルモード使用時で、オルタネートバイトに 2 サイクルだけが使われる場合などです。この場合、ファームウェアによって Quad モード（ABMODE=11）を使用し、ALTERNATE のビット 7 および 3 を“1”（IO3 ラインをハイで保持）に、ビット 6 および 2 を“0”（IO2 ラインをローで保持）にセットしてバイトを送信できます。送信される 4 ビットの上位 2 ビットは ALTERNATE のビット 4:3 にセットされ、下位 2 ビットはビット 1 および 0 にセットされます。例えば、4bitの値2（0010）が IO0/IO1 で送信される場合、ALTERNATE は 0x8A（1000_1010）にセットされている必要があります。

ダミーサイクルフェーズ

ダミーサイクルフェーズでは、データの送受信は一切なく、1～31 サイクルが実行されます。これは、より高いクロック周波数が使われいてる場合に、データフェーズの準備をする時間をフラッシュメモリに与えるためです。このフェーズ中に実行されるサイクル数は、QUADSPI_CCR[22:18] レジスタの DCYC[4:0] フィールドで指定されます。SDR モードと DDR モードのどちらでも、処理時間にはフル CLK サイクルの数が指定されます。

DCYC = 00 の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

ダミーサイクルフェーズの動作モードは DMODE によって決まります。

データ信号を出力モードから入力モードへ変更するのに十分な「ターンアラウンド」時間を確保するために、デュアルまたはクワッドモードを使用してフラッシュメモリからデータを受信する場合、少なくとも 1 ダミーサイクルが必要です。

データフェーズ

データフェーズ中は、フラッシュメモリに対して何バイトでも送受信することができます。

インダイレクトモードと自動ポーリングモードでは、送受信するバイト数は QUADSPI_DLR レジスタで指定されます。

インダイレクト書き込みモードでは、フラッシュメモリに送信されるデータを QUADSPI_DR レジスタに書き込む必要があります。一方、インダイレクト読み出しモードでは、フラッシュメモリから受信するデータは QUADSPI_DR レジスタから読み出して取得します。

メモリマップドモードでは、読み出されるデータは、直接 AXI を経由して Cortex または DMA へ送り戻されます。

データフェーズで同時に送受信できるのは、1 ビット（シングル SPI モードの SO/SI 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

DMODE = 00 の場合、データフェーズはスキップされますので、コマンドシーケンスは nCS を立ち上げることですぐに終了します。この設定はインダイレクト書き込みモードのみで使用してください。

23.3.4 QUADSPI シングルインタフェースプロトコルモード

シングル SPI モード

レガシー SPI モードでは、1 ビットの順次送受信のみが可能です。このモードでは、データは SO 信号（I/O を IO0 と共有）を経由してフラッシュメモリへ送信されます。フラッシュメモリからのデータは、SI 信号（I/O を IO1 と共有）を経由して受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を”01”にセットすることで、このシングルビットモードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをシングルモードに設定する場合

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2 は出力モードで、強制的に“0”に設定されます（書き込み保護機能を無効にするため）。
- IO3 は出力モードで、強制的に“1”に設定されます（ホールド機能を無効にするため）。

これは、DMODE = 01 であれば、ダミーフェーズにも該当します。

デュアル SPI モード

デュアル SPI モードでは、IO0/IO1 信号を経由して 2 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“10”にセットすることで、このデュアル SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをデュアルモードに設定する場合

- IO0/IO1 は、データフェーズの読み出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2 は出力モードで、強制的に“0”に設定されます。
- IO3 は出力モードで、強制的に“1”に設定されます。

ダミーフェーズでは、DMODE = 01 の場合、IO0/IO1 は常にハイインピーダンスです。

Quad SPI モード

Quad SPI モードでは、IO0/IO1/IO2/IO3 信号を経由して 4 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を”11”にセットすることで、このQuad SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

クワッドモードに設定されている各フェーズでは、IO0/IO1/IO2/IO3 はすべてデータフェーズの読み出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE = 11 の場合、IO0/IO1/IO2/IO3 はすべて常にハイインピーダンスです。

IO2 と IO3 はQuad SPI モードでのみ使用されます。Quad SPI モードを使用できるように設定されたフェーズが 1 つもない場合、IO2 および IO3 に対応するピンは QUADSPI がアクティブなときでも、他の機能に使用することができます。

SDR モード

DDRM ビット（QUADSPI_CCR[31]）は、デフォルトで 0 に設定されており、QUADSPI はシングルデータレート（SDR）モードで動作します。

SDR モードでは、QUADSPI が IO0/SO、IO1、IO2、IO3 信号を駆動している場合、これらの信号は CLK の立ち下がりエッジでのみ遷移します。

SDR モードでデータを受信する場合、QUADSPI はフラッシュメモリも CLK の立ち下がりエッジを使用してデータを送信するとみなします。デフォルトでは（SSHIFT = 0 の場合）、信号は次の CLK の（立ち上がり）エッジを使用してサンプリングされます。

DDR モード

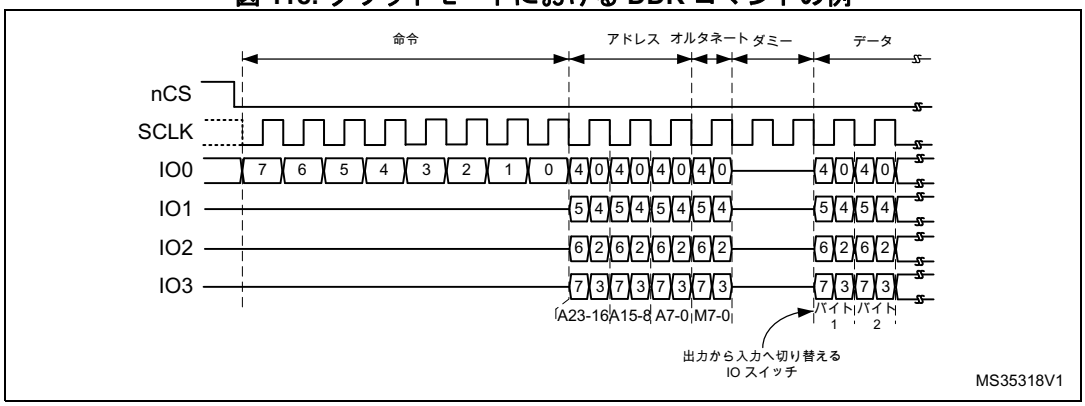
DDRM ビット（QUADSPI_CCR[31]）が 1 にセットされている場合、QUADSPI はダブルデータレート（DDR）モードで動作します。

DDR モードでは、QUADSPI がアドレス／オルタネートバイト／データフェーズで IO0/SO、IO1、IO2、IO3 信号を駆動している場合、CLK の立ち下がりエッジおよび立ち上がりエッジのそれぞれでビットを送信します。

命令フェーズは DDRM の影響は受けません。命令は常に CLK の立ち下がりエッジを使用して送信されます。

DDR モードでデータを受信する場合、QUADSPI はフラッシュメモリも CLK の立ち上がりエッジと立ち下がりエッジの両方を使用してデータを送信するとみなします。DDRM = 1 の場合、ファームウェアは SSHIFT ビット（QUADSPI_CR のビット 4）をクリアする必要があります。したがって、信号は 1/2 CLK サイクル後にサンプリングされます（下図の逆エッジで）。

図 118. クワッドモードにおける DDR コマンドの例



デュアルフラッシュモード

DFM ビット（QUADSPI_CR のビット 6）が 1 の場合、QUADSPI はデュアルフラッシュモードです。このモードでは、サイクルごとに 8ビットの送受信（DDR モードでは 16 ビット）を行うために 2 つの外部 Quad SPI フラッシュメモリ（フラッシュ 1 と フラッシュ 2）が使用され、スループットおよび容量を効果的に倍にします。

どちらのフラッシュメモリも同じ CLK を使用し、オプションで同じ nCS 信号を使用しますが、それぞれが個別の IO0、IO1、IO2、IO3 信号を有しています。

デュアルフラッシュモードは、SDR　モードまたは DDR モードと組み合わせて使用できるだけでなく、シングルビットモード、デュアルビットモード、クワッドビットモードとも組み合わせて使用できます。

FSIZE[4:0]（QUADSPI_DCR[20:16]）で指定されているフラッシュメモリのサイズは、フラッシュメモリの総容量（コンポーネント 1 個のサイズの 2 倍）を反映していなければなりません。

アドレス X が偶数の場合、QUADSPI はアドレス X に対してフラッシュ 1 のアドレス X/2 にあるバイトを与え、アドレス X+1 に対してフラッシュ 2 のアドレス X/2 にあるバイトを与えます。言い換えれば、偶数アドレスにあるバイトはすべてフラッシュ 1 に、奇数アドレスにあるバイトはすべてフラッシュ 2 に格納されます。

デュアルフラッシュモードでフラッシュメモリステータスレジスタを読み出す場合、シングルフラッシュモードで読み出すバイト数の2 倍のバイトを読み出す必要があります。つまり、各フラッシュメモリがステータスレジスタのフェッチ命令後に有効な 8 ビットを提供する場合、QUADSPI はデータ長 2 バイト（16 ビット）で設定する必要があり、それにより QUADSPI は各フラッシュメモリから 1 バイトを受信します。各フラッシュメモリが 16 ビットのステータスを提供する場合、QUADSPI は、4 バイトを読み出して、デュアルフラッシュモードの両方のフラッシュメモリのステータスビットをすべて取得できるよう設定する必要があります。リザルト（データレジスタ内）の最下位バイトはフラッシュ 1 ステータスレジスタの最下位バイト、次のバイトはフラッシュ 2 ステータスレジスタの最下位バイトとなります。さらに、データレジスタの 3 番目のバイトはフラッシュ 1 の 2 番目のバイト、4 番目のバイトはフラッシュ 2 の 2 番目のバイトとなります（フラッシュメモリに 16 ビットステータスレジスタがある場合）。

偶数バイトは常にデュアルフラッシュモードでアクセスする必要があります。このため、DRM = 1 のとき、データ長フィールドのビット 0 （QUADSPI_DLR[0]）は 1 のままとなります。

デュアルフラッシュモードでは、フラッシュ 1 インタフェース信号の動作は基本的に通常モードの動作と同じです。フラッシュ 2 インタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、フラッシュ 1 とまったく同じです。つまり、どちらのフラッシュメモリも常に同じ命令や同じアドレスを受信するということです。データフェーズ中は、2 つのバス BK1_IOx と BK2_IOx は並列してデータ転送を行いますが、フラッシュ 1 に対する送信（または受信）データはフラッシュ 2 のものとは異なります。

23.3.5 QUADSPI インダイレクトモード

インダイレクトモードでは、QUADSPI レジスタに書き込むことでコマンドが開始され、データレジスタへの書き込み／読み出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

FMODE = 00（QUADSPI_CCR[27:26]）の場合、QUADSPI はインダイレクト書き込みモードです。このモードでは、データフェーズ中にフラッシュメモリへのバイトの送信が行われます。データは、データレジスタ（QUADSPI_DR）への書き込みによって提供されます。

FMODE = 01 の場合、QUADSPI はインダイレクト読み出しモードです。このモードでは、データフェーズ中にフラッシュメモリからのバイトの受信が行われます。データは、QUADSPI_DR の読み出しによって復旧されます。

送受信するバイト数はデータ長レジスタ（QUADSPI_DLR）で指定されます。
QUADSPI_DLR = 0xFFFF_FFFF（すべて“1”）の場合、データ長は不定長と見なされ、QUADSPI はフラッシュメモリの終了（FSIZE によって定義される）までデータを転送し続けるのみです。転送されるバイトが無い場合は、DMODE（QUADSPI_CCR[25:24]）を”00”にセットする必要があります。

QUADSPI_DLR = 0xFFFF_FFFF かつ FSIZE = 0x1F（4 GB のフラッシュメモリを示す最大値）の場合、このような特殊なケースでは、転送は無限に続けられ、アボートリクエスト後または QUADSPI の無効化後にやっと停止します。最後のメモリアドレスが読み出された後も（アドレス 0xFFFF_FFFF で）、アドレス = 0x0000_0000 で読み出しが続けられます。

プログラムされた送信または受信バイト数に達すると、TCF がセットされ、TCIE = 1 であれば割り込みが生成されます。データ数が定義されていない場合、外部 SPI メモリの限界に達した時点で、QUADSPI_CR に定義されているフラッシュメモリのサイズに応じて TCF がセットされます。

コマンド開始のトリガ

コマンドは、基本的に、ファームウェアがそのコマンドに必要な最期の情報を提供するとすぐに開始されます。インダイレクトモードでコマンドを開始する方法は、QUADSPI の設定に応じて、3 通りあります。コマンドは以下の動作の直後に開始されます。

1. アドレスを必要としない場合（ADMODE = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、INSTRUCTION[7:0]（QUADSPI_CCR）への書き込み動作直後
2. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、ADDRESS[31:0]（QUADSPI_AR）への書き込み動作直後
3. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要とする場合（FMODE = 00 および DMODE ! = 00）は、DATA[31:0]（QUADSPI_DR）への書き込み動作直後

オルタネートバイトレジスタ（QUADSPI_ABR）への書き込みが通信開始をトリガすることはありません。オルタネートバイトが必要である場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSY ビット（QUADSPI_SR の ビット5）が自動的にセットされます。

FIFO およびデータの管理

インダイレクトモードでは、データは QUADSPI に内蔵されている 32バイト FIFO を通ります。FLEVEL[5:0] （QUADSPI_SR[13:8]）は、現在 FIFO に保持されているバイト数を示します。

インダイレクト書き込みモード（FMODE = 00）では、ファームウェアは QUADSPI_DR に書き込む時点で FIFO にデータを追加します。書き込みにより FIFO に追加されるバイト数は、ワード書き込みで 4 バイト、ハーフワード書き込みで 2 バイト、バイト書き込みで 1 バイトです。ファームウェアが FIFO に追加するバイト数が多すぎる（DL[31:0] によって示される数より多い）場合、余分なバイトは、書き込み動作の終了時（TCF がセットされるとき）に FIFO から一掃されます。

QUADSPI_DR へのバイト／ハーフワードアクセスは、32 ビットレジスタの最下位バイト／ハーフワードにのみ実行されます。

FTHRES[3:0] を使用して FIFO 閾値が定義されます。閾値に達すると、FTF（FIFO 閾値フラグ）がセットされます。インダイレクト読み出しモードでは、FIFO から読み出される有効バイト数が閾値を超えると FTF がセットされます。また、FTHRES の設定に関わらず、フラッシュメモリから最後のバイトが読み出された後に FIFO 内にデータが残っている場合も FTF がセットされます。インダイレクト書き込みモードでは、FIFO 内の空のバイト数が閾値を超えると FTF がセットされます。

FTIE = 1 の場合、FTF がセットされた時点で割り込みが発生します。閾値条件が真でなくなる（CPU または DMA によって十分なデータが転送された後）とすぐに、HW によって FTF がクリアされます。

インダイレクト読み出しモードでは、FIFO がフルになると、QUADSPI は一時的にフラッシュメモリからのバイトの読み出しを停止して、オーバーランを回避します。フラッシュメモリの読み出しは、FIFO 内の 4 バイトが空になるまで再開されませんのでご注意ください（FLEVEL ≤ 11 の場合）。したがって、FTHRES ≥ 13 の場合、QUADSPI は確実にフラッシュメモリからのデータ取得を再開できるものとして、アプリケーションによる十分なバイトの読み出しが行われる必要があります。そうしないと、11 < FLEVEL < FTHRES である限り、FTF フラグは 0 のままとなります。

23.3.6 QUADSPI ステータスフラグポーリングモード

自動ポーリングモードでは、QUADSPI は定期的にコマンドを開始させて、定義された数のステータスバイト（最大 4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割り込みを生成することができます。

フラッシュメモリへのアクセス方法はインダイレクト読み出しモードの場合と同様です。アドレスを必要としない場合（ADMODE = 00）は、QUADSPI_CCR が書き込まれるとすぐにアクセスが開始されます。反対にアドレスを必要とする場合は、QUADSPI_AR が書き込まれた時点で最初のアクセスが始まります。この時点で BUSY がハイになり、周期的なアクセスが実行される間もハイのままです。

MASK[31:0]（QUADSPI_PSMAR）の内容を使用して、自動ポーリングモードにおけるフラッシュメモリからのデータがマスクされます。MASK[n] = 0 の場合、リザルトのビット n はマスクされ、考慮されません。MASK[n] = 1 かつ　bit[n] の値が MATCH[n]（QUADSPI_PSMAR）の値と同じである場合、bit n との一致が見られます。

ポーリング一致モードビット（PMM、QUADSPI_CR のビット 23）が 0 の場合、「AND」一致モードがアクティブになります。つまり、ステータス一致フラグ（SMF）は、マスクされていないすべてのビットで一致が検出された場合にのみセットされます。

PMM = 1 の場合、「OR」一致モードがアクティブになります。つまり、SMF は、マスクされていない任意のビットで一致が検出された場合にセットされます。

SMIE = 1 の場合に SMF がセットされると、割り込みが呼び出されます。

自動ポーリングモード停止（APMS）ビットがセットされている場合、一致が検出されるとすぐに動作は停止し、BUSY フラグが 0 になります。あるいは、BUSY は 1 のままで、アボートが検出される、または QUADSPI が無効化される（EN = 0）まで周期的なアクセスが継続します。

データレジスタ（QUADSPI_DR）には、最後に受信したステータスバイト（FIFO の無効化）が含まれます。データレジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。FTF ステータスビットは新たなステータスの読み出しが完了するとすぐにセットされ、データが読み出された直後にクリアされます。

23.3.7 QUADSPI メモリマップドモード

メモリマップドモードに設定された場合、外部 SPI デバイスは内部メモリとみなされます。

QUADSPI ペリフェラルを正しく設定し、有効にする前に、QUADSPI フラッシュバンク領域にアクセスすることは禁じられています。

フラッシュメモリの容量がもっと大きい場合でも、アドレス指定できるのは 256 MB までです。

FSIZE で規定されている範囲外のアドレスにアクセスすると、256 MB の範囲内であっても、バスエラーが発生します。このエラーの影響は、アクセスを試みたバスマスタに依存します。

- Cortex® CPU の場合は、有効時バスフォールト例外（またはバスフォールト無効時にハードフォールト例外）が発生します。
- DMA の場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。この場合、QUADSPI は次のマイクロコントローラのアクセスを予測し、あらかじめ次のアドレスにバイトをロードします。その後のアクセスが実際に連続したアドレスに対して行われる場合、値がすでにプリフェッチされているので、アクセスは高速で完了します。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、フラッシュメモリへのアクセスが長時間なくても、前の読み出し動作はアクティブな状態を保ち、nCs もローに保持されます。nCS がローのときにフラッシュメモリはより多く電力を消費する傾向があるため、FIFO がプリフェッチデータでフルになっている場合は、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過したあとで nCS が開放されるように、アプリケーションはタイムアウトカウンタ（TCEN=1、QUADSPI_CR のビット 3）の起動を要求する場合があります。

最初のメモリマップドアクセスが発生するとすぐに、BUSY はハイになります。プリフェッチ動作の影響で、BUSY の立ち下がりは、タイムアウトが起こる、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

23.3.8 QUADSPI フリーランニングクロックモード

フリーランニングクロックモードに設定される場合、QUADSPI ペリフェラルはテストおよび較正用のクロックを出力し続けます。

QUADSPI 通信設定レジスタ（QUADSPI_CCR）のフリーランニングクロックモードビット（FRCM）がセットされるとすぐに、フリーランニングクロックモードに入ります。モードを終了するには、QUADSPI 制御レジスタ（QUADSPI_CR）の ABORT ビットをセットします。

QUADSPI がフリーランニングクロックモードで動作する場合

- クロックは連続的に動作します。
- nCS はハイのままです（外部デバイスは非選択）
- データラインは開放されます（ハイインピーダンス）。
- QUADSPI ステータスレジスタ（QUADSPI_SR）の BUSY フラグがセットされます。

23.3.9 QUADSPI フラッシュメモリの設定

デバイス設定レジスタ（QUADSPI_DCR）を使用して、外部 SPI フラッシュメモリの特性を指定することができます。

FSIZE[4:0] フィールドは、次の式を使用して外部メモリのサイズを定義します。

$$\text{フラッシュメモリ内のバイト数} = 2^{[FSIZE+1]}$$

FSIZE+1 はフラッシュメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。フラッシュメモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE は 2 つのフラッシュメモリの総容量を示します。

QUADSPI が 2 つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは1 CLK サイクル中の 2 つのコマンドの間にハイレベルのチップセレクト信号（nCS）を立ち上げます。フラッシュメモリがコマンド間により長い時間を必要とする場合、チップセレクトハイ時間（CSHT）フィールドを使用して、nCS がハイに保たれなければならない CLK サイクルの回数（最大 8）を指定することができます。

クロックモード（CKMODE）ビットは、コマンド間の CLK 信号ロジックレベルを示します（nCS = 1 の場合）。

23.3.10 QUADSPI 遅延データサンプリング

デフォルトでは、QUADSPI がフラッシュメモリに駆動されているデータのサンプリングを行うタイミングは、フラッシュメモリが信号を駆動してから 1/2 CLK サイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。SSHIFT ビット（QUADSPI_CR のビット 4）を使用することで、データのサンプリングを 1/2 CLK サイクルだけシフトすることができます。

クロックのシフトは DDR モードではサポートされていませんので、DDRM ビットがセットされた時点で SSHIFT ビットをクリアする必要があります。

23.3.11 QUADSPI の設定

QUADSPI の設定は次の 2 つのフェーズで行われます。

- QUADSPI IP の設定
- QUADSPI フラッシュメモリの設定

QUADSPI は、設定および有効化が行われると、インダイレクトモード、ステータスポーリングモード、メモリマップドモードの 3 つの動作モードのいずれかで使用することができます。

QUADSPI IP の設定

QUADSPI IP は QUADSPI_CR を使用して設定します。ユーザは、受信データに対してクロックプリスケーラ分周比とサンプリングシフトを設定します。

DDR モードは DDRM ビットによって設定することができます。このモードを有効にすると、アドレスおよびオルタネートバイトが両方のクロックエッジに送信され、両方のクロックエッジでデータの送受信が行われます。DDRM ビットの設定にかかわらず、命令は常に SDR モードで送信されます。

MDMA トリガ生成 または割り込み生成のための FIFO レベルは FTHRES ビットでプログラムされます。

タイムアウトの発生が必要な場合、TCEN ビットをセットして、QUADSPI_LPTR レジスタにタイムアウト値を設定することができます。

デュアルフラッシュモードは DFM を “1”にセットすることで有効にできます。

QUADSPI フラッシュメモリの設定

対象の外部フラッシュメモリに関連するパラメータは QUADSPI_DCR レジスタによって設定されます。ユーザは FSIZE ビットでフラッシュメモリのサイズを、CSHT ビットでチップセレクトハイ時間の最小値を、MODE ビットで機能モード（モード 0 またはモード 3）を設定します。

23.3.12 QUADSPI の使用

動作モードは FMODE[1:0]（QUADSPI_CCR[27:26])を使用して選択されます。

インダイレクトモードの手順

FMODE が 00 に設定されている場合、インダイレクト書き込みモードが選択され、データをフラッシュメモリに送信することができます。FMODE が 01 に設定されている場合、インダイレクト読み出しモードが選択され、データをフラッシュメモリから読み出すことができます。

QUADSPI がインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. QUADSPI_DLR に、読み出される／書き込まれるデータのバイト数を指定します。
2. QUADSPI_CCR に、フレームフォーマット、モード、および命令コードを指定します。
3. QUADSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
4. QUADSPI_CR に、動作モードを指定します。
5. QUADSPI_AR に、ターゲットアドレスを指定します。
6. QUADSPI_DR により FIFO に対してデータの読み出し／書き込みを行います。

制御レジスタ（QUADSPI_CR）に書き込む場合は次の設定値を指定してください。

- イネーブルビット（EN）を“1”にセット
- タイムアウトカウンタイネーブルビット（TCEN）
- サンプリングシフト設定（SSHIFT）
- FTF フラグをセットするタイミングを示す FIFO 閾値レベル（FTRHES）
- 割り込みイネーブル
- 自動ポーリングモードのパラメータ : 一致モードおよび STOP モード（FMODE=11 のとき有効）
- クロックプリスケーラ

通信設定レジスタ（QUADSPI_CCR）に書き込む場合は次のパラメータを指定してください。

- INSTRUCTION ビットによる命令バイト
- IMODE ビット（1、2、または4 ライン）により命令を送信する方法
- ADMODE ビット（なし、または1、2、4 ライン）によりアドレスを送信する方法
- ADSIZE ビットによるアドレスサイズ（8、16、24、または32 ビット）
- ADMODE ビット（なし、または1、2、4 ライン）によりオルタネートバイトを送信する方法
- ABSIZE ビットによるオルタネートバイト数（1、2、3、または4）
- DBMODE ビットによるダミーバイトの有無
- DCYC ビットによるダミーバイト数
- DMODE ビットによるデータの送受信方法（なし、または1、2、4 ライン）

アドレスレジスタ（QUADSPI_AR）とデータレジスタ（QUADSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、QUADSPI_CCR が書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE と DMODE がどちらも 00 である場合、またはインダイレクト読み出しモード（FMODE=01）で ADMODEが 00 の場合に該当します。

アドレスを必要とする場合（ADMODE が 00 以外）で、データレジスタに書き込む必要がない場合（FMODE = 01 または DMODE = 00）、QUADSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE = 00 かつ DMODE! = 00）、QUADSPI_DR による FIFO への書き込みによって通信の開始がトリガされます。

ステータスフラグポーリングモード

ステータスフラグポーリングモードは、FMODE フィールド（QUADSPI_CCR[27:26]）を 10 にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は 4 バイトです。QUADSPI_DLR で もっと大量のデータが要求されても、そのリクエストは無視され、4 バイトだけが読み出されます。

QUADSPI_PISR レジスタには周期性が指定されます。

ステータスデータが取得されると、そのデータを次の目的のために内部で処理することができます。

- ステータス一致フラグをセットし、有効になっていれば割り込みを生成する。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値は QUADSPI_PSMKR に格納されている値でマスクし、QUADSPI_PSMAR に格納されている値で論理和または論理積をとることができます。

一致する場合、ステータス一致フラグがセットされ、有効になっていれば割り込みが生成されます。また、AMPS ビットがセットされている場合、QUADSPI は自動的に停止する可能性があります。

いずれにしろ、一番最後に取得した値は QUADSPI_DR で使用可能です。

メモリマップドモード

メモリマップドモードでは、外部フラッシュメモリは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部フラッシュメモリに対して読み出し動作のみが許可されます。

メモリマップドモードに移行するには、QUADSPI_CCR レジスタの FMODE を 11 にセットします。

プログラムされた命令およびフレームは、マスタがメモリマップド空間にアクセスしているときに送信されます。

FIFO は、リニア読み出しを予測するプリフェッチバッファとして使用されます。このモードでの QUADSPI_DR へのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（QUADSPI_DLR）は意味を持ちません。

23.3.13 1 回限りの命令の送信

一部のフラッシュメモリ（Winbound など）が提供するモードは、最初のコマンドシーケンスでのみ命令を送信し、その後のコマンドは直接アドレスから開始されるというものです。SIOO ビット（QUADSPI_CCR[28]）を使用して、そのような機能を活用することができます。

SIOO はすべての機能モードで有効です（インダイレクト、自動ポーリング、メモリマップド）。SIOO ビットがセットされている場合、QUADSPI_CCR への書き込み後の最初のコマンドに対してのみ命令が送信されます。その後のコマンドシーケンスは、命令フェーズでは、QUADSPI_CCR への書き込みがあるまでスキップされます。

IMODE = 00（命令なし）の場合、SIOO は効果がありません。

23.3.14 QUADSPI エラー管理

次の場合に、エラーが生成される可能性があります。

- インダイレクトモードまたはステータスフラグポーリングモードでは、QUADSPI_AR に間違ったアドレスがプログラムされている場合（QUADSPI_DCR の FSIZE[4:0] で定義されているフラッシュメモリのサイズによる）、TEF がセットされ、有効になっていれば割り込みが生成されます。
- また、インダイレクトモードでは、アドレスおよびデータ長がフラッシュメモリのサイズを超えている場合は、アクセスがトリガされるとすぐに TEF がセットされます。
- メモリマップドモードでは、マスタによる範囲外のアクセスが行われた場合または QUADSPI が無効な場合は、誤った バスマスタリクエストへの応答としてバスエラーが生成されます。
- メモリマップドモードが無効なときにマスタがメモリマップド空間にアクセスすると、誤ったバスマスタリクエストへの応答としてバスエラーが生成されます。

23.3.15 QUADSPI の BUSY ビットおよびアボート機能

QUADSPI がフラッシュメモリでの動作を開始すると、BUSY ビットが QUADSPI_SR に自動的にセットされます。

インダイレクトモードでは、QUADSPI が要求されたコマンドシーケンスを完了し、FIFO が空になると、BUSY ビットがリセットされます。

自動ポーリングモードでは、一致（APMS = 1 のとき）またはアボートの検出により、最後の周期的なアクセスが完了して初めて BUSY ビットはローになります。

メモリマップドモードでの最初のアクセス後、タイムアウトイベントまたはアボートの検出により、BUSY ビットはローになります。

QUADSPI_CR レジスタの ABORT ビットをセットすることによって、動作を中止することができます。アボートが完了すると、BUSY ビットおよび ABORT ビットは自動的にリセットされ、FIFO が一掃されます。

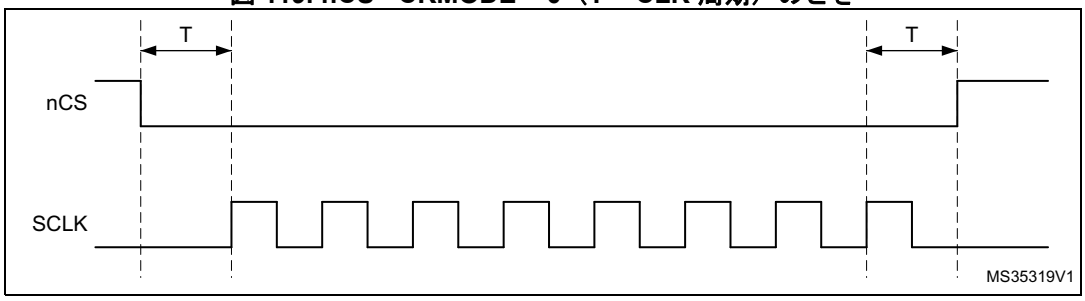
注： ***フラッシュメモリには、ステータスレジスタへの書き込み動作が中止された場合に誤動作するものもあります。***

23.3.16 nCS の動作

デフォルトでは nCS はハイで、外部フラッシュメモリは選択解除されています。nCS の立ち下がりは動作が始まる前に、立ち上がりは動作終了の直後に起こります。

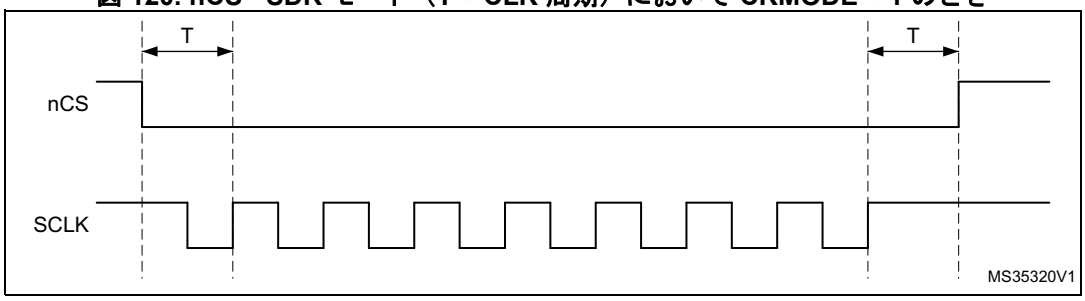
CKMODE = 0（「モード0」で、この状態では進行中の動作がない場合は CLK がローに保持されいてる）のとき、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（*図 119*を参照）。

図 119. nCS - CKMODE = 0（T = CLK 周期）のとき



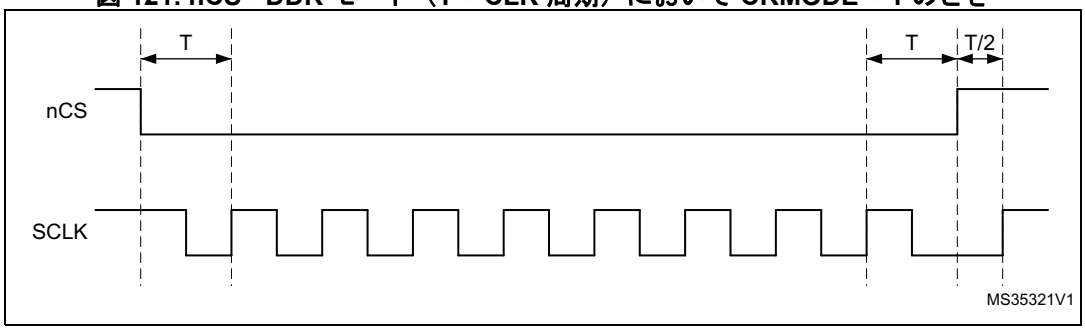
CKMODE = 1（「モード3」で、この状態では進行中の動作がない場合に CLK がハイになる）、かつ DDRM = 0（SDR モード）のとき、nCS は動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前でさらに立ち下がり、動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に立ち上がります（*図 120*を参照）。

図 120. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき



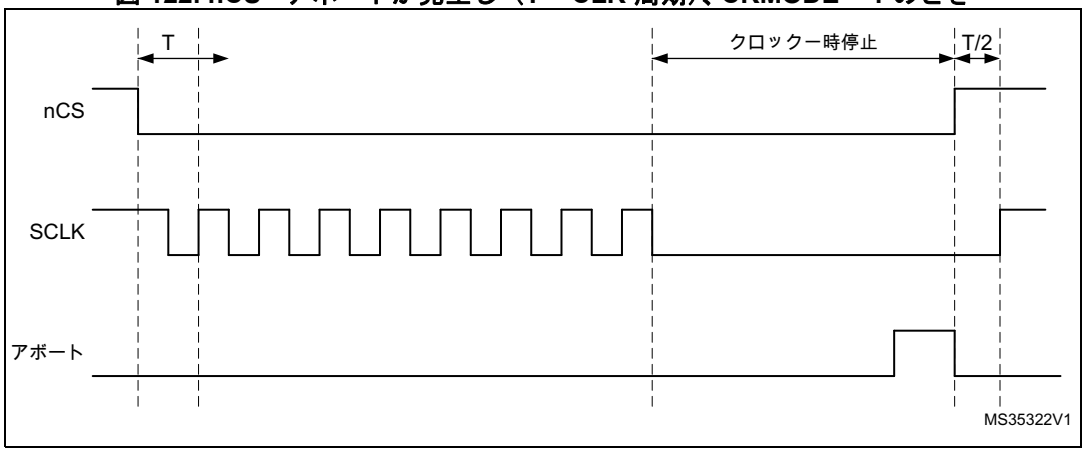
CKMODE = 1（モード3）かつ DDRM = 1（DDR モード）の場合、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の アクティブ CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（*図 121*を参照）。DDR 動作は立ち下がりエッジで終わらなければならないため、nCS が立ち上がるとき CLK はローとなり、その後 CLK は 1/2 CLK サイクルだけ後ろ向きに立ちあがります。

図 121. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき



読み出し動作で FIFO がフルのまま、または書き込み動作で FIFO が空のままであれば、ファームウェアが FIFO を処理するまでその動作は停止し、CLK はローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後に nCS の立ち上がりが起こり、CLK は 1/2 CLK サイクル後に立ち上がります（*図 122*を参照）。

図 122. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき



デュアルフラッシュモード以外のとき（DFM = 0）はフラッシュ 1 のみがアクセスされるので、BK2_nCS はハイのままです。デュアルフラッシュモードでは、BK2_nCS は BK1_nCS とまったく同じように動作します。したがって、フラッシュ 2 が存在し、アプリケーションがずっとデュアルフラッシュモードのままであれば、フラッシュ 2 は BK1_nCS を使用することができ、BK2_nCS を出力するピンを他の機能に使用することができます。

23.4 QUADSPI 割り込み

割り込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO 閾値
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 184. QUADSPI 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO 閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

参考資料

23.5 QUADSPI レジスタ

23.5.1 QUADSPI 制御レジスタ（QUADSPI_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESCALER								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FTHRES					FSEL	DFM	Res.	SSHIFT	TCEN	Res.	ABORT	EN
			rw	rw	rw	rw	rw	rw	rw		rw	rw		rw	w1s

ビット 31:24 **PRESCALER[7:0]**：クロックプリスケーラ

このフィールドは、quadspi_ker_ck クロック（値 + 1）に基づいて CLK を生成するスケーラ係数を定義します。

0：$F_{CLK} = F_{quadspi_ker_ck}$、すなわち QUADSPI CLK（バイパスされたプリスケーラ）として直接的に使用される quadspi_ker_ck クロック

1：$F_{CLK} = F_{quadspi_ker_ck}/2$

2：$F_{CLK} = F_{quadspi_ker_ck}/3$

...

255：$F_{CLK} = F_{quadspi_ker_ck}/256$

クロック分周比が奇数の場合、CLK のデューティサイクルは 50% ではありません。クロック信号は、ハイに保持された時間より 1 サイクル分長く、ローに保持されます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 23 **PMM**：ポーリング一致モード

このビットは、自動ポーリングモード時に「一致」を決定するために使うべき方法を示します。

0：AND 一致モードSMF は、フラッシュメモリから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1：OR 一致モードSMF は、フラッシュメモリから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 22 **APMS**：自動ポーリングモード停止

このビットは、一致後に自動ポーリングを停止するかどうかを決定します。

0：自動ポーリングモードは、アボートまたは QUADSPI の無効化によってのみ停止されます。

1：自動ポーリングモードは、一致が検出されるとすぐに停止されます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TOIE**：タイムアウト割り込み有効化

このビットは、タイムアウト割り込みを有効にします。

0：割り込みは無効です。

1：割り込みは有効です。

ビット 19 **SMIE**：ステータス一致割り込み有効化

このビットは、ステータス一致割り込みを有効にします。

0：割り込みは無効です。

1：割り込みは有効です。

ビット 18 **FTIE**：FIFO 閾値割り込み有効化
このビットは、FIFO 閾値割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 17 **TCIE**：転送完了割り込み有効化
このビットは、転送完了割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 16 **TEIE**：転送エラー割り込み有効化
このビットは、転送エラー割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **FTHRES[4:0]** FIFO 閾値レベル
インダイレクトモードでは、FIFO 閾値フラグ（FTF、QUADSPI_SR[2]）がセットされる FIFO 内のバイトの閾値数を定義します。
インダイレクト書き込みモード（FMODE = 00)の場合
0 : FTF は、FIFO 内に書き込みのできる空きバイトが 1 バイト以上ある場合にセットされます。
1 : FTF は、FIFO 内に書き込みのできる空きバイトが 2 バイト以上ある場合にセットされます。
...
31：FTF は、FIFO 内に書き込みのできる空きバイトが 32 バイト以上ある場合にセットされます。
インダイレクト読み出しモード（FMODE = 01)の場合
0：FTF は、FIFO から読み出せる有効なバイトが 1 バイト以上ある場合にセットされます。
1：FTF は、FIFO から読み出せる有効なバイトが 2 バイト以上ある場合にセットされます。
...
31：FTF は、FIFO から読み出せる有効なバイトが 32 バイト以上ある場合にセットされます。

ビット 7 **FSEL**：フラッシュメモリの選択
このビットは、シングルフラッシュモード（DFM = 0）でアドレス指定するフラッシュメモリを選択します。
0：フラッシュ 1 が選択されます。
1：フラッシュ 2 が選択されます。
このビットは、BUSY が 0 のときにのみ修正できます。
このビットは、DFM = 1 のときは無視されます。

ビット 6 **DFM**：デュアルフラッシュモード
このビットはデュアルフラッシュモードをアクティブにします。このモードでは、2 つの外部フラッシュメモリを同時に使用して、スループットおよび容量を倍にします。
0：デュアルフラッシュモードは無効です。
1：デュアルフラッシュモードは有効です。
このビットは、BUSY が 0 のときにのみ修正できます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SSHIFT**：サンプリングシフト

デフォルトでは、QUADSPI はフラッシュメモリによるデータ駆動の 1/2 CLK サイクル後にデータをサンプリングします。このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2 サイクルのシフト

ファームウェアは、DDR モード（DDRM = 1）では SSHIFT = 0 であることを予測する必要があります。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 3 **TCEN**：タイムアウトカウンタ有効化

このビットは、メモリマップドモードが選択されている（FMODE = 11)場合にのみ有効です。ある一定の時間アクセスがない場合（この時間は TIMEOUT[15:0]（QUADSPI_LPTR）によって定義されます)、このビットをアクティブにすることでチップセレクト（nCS）がリリースされます（これにより消費電力が削減されます)。

タイムアウトカウンタを有効にします。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、フラッシュメモリへのアクセスが長時間なくても、前の読み出し動作はアクティブな状態を保ち、nCs もローに保持されます。nCS がローのときにフラッシュメモリはより多く電力を消費する傾向があるため、FIFO がプリフェッチデータでフルになっている場合は、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過したあとで nCS が開放されるように、アプリケーションはタイムアウトカウンタ（TCEN=1、QUADSPI_CR のビット 3）の起動を要求する場合があります。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（nCS）はメモリマップドモードにおける 1 回のアクセス後は無限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、インアクティブなフラッシュメモリの TIMEOUT[15:0] サイクル後にメモリマップドモードでリリースされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 2 予約済み

ビット 1 **ABORT**：アボートリクエスト

このビットは進行中のコマンドシーケンスを停止します。すべてのアボートが完了すると自動的にリセットされます。

このビットは現在の転送を中止させます。

ポーリングモードまたはメモリマップドモードでは、このビットは APM ビットまたは DM ビットもリセットします。

0：アボートは要求されていません。

1：アボートが要求されています。

ビット 0 **EN**：有効化

QUADSPI. を有効にします。

0：QUADSPI. は無効です。

1：QUADSPI. は有効です。

23.5.2 QUADSPI デバイス設定レジスタ（QUADSPI_DCR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

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

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **FSIZE[4:0]**：フラッシュメモリサイズ

このフィールドは、次の式を使用して外部メモリのサイズを定義します。

フラッシュメモリ内のバイト数 = $2^{[FSIZE+1]}$

FSIZE+1 はフラッシュメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。フラッシュメモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE は 2 つのフラッシュメモリの総容量を示します。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **CSHT[2:0]**：チップセレクトハイ時間

CSHT+1 は、フラッシュメモリに対して発行されるコマンド間でチップセレクト（nCS）がハイに保たれなければならない CLK サイクルの最小数を定義します。

0：nCS は、フラッシュメモリコマンド間の少なくとも 1 サイクルの間はハイに保持されます。

1：nCS は、フラッシュメモリコマンド間の少なくとも 2 サイクルの間はハイに保持されます。

...

7：nCS は、フラッシュメモリコマンド間の少なくとも 8 サイクルの間はハイに保持されます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CKMODE**：モード 0／モード 3

このビットは、コマンド間での CLK のレベルを示します（nCS = 1のとき）。

0 : CLK は nCS がハイである間はローに保持される必要があります（チップセレクトがリリースされる）。これは、モード 0　と呼ばれます。

1 : CLK は nCS がハイである間はハイに保持される必要があります（チップセレクトがリリースされる）。これは、モード 3　と呼ばれます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

23.5.3 QUADSPI ステータスレジスタ（QUADSPI_SR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FLEVEL[5:0]						Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
		r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **FLEVEL[5:0]**：FIFO レベル

このフィールドは FIFO 内に保持されている有効バイト数を示します。FIFO が空の場合、FLEVEL = 0、フルになっている場合、32です。メモリマップドモードおよび自動ステータスポーリングモードでは、FLEVEL はゼロです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。このビットは、フラッシュメモリの動作が終わり、FIFO が空になると自動的にクリアされます。

ビット 4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOF に 1 を書き込むことによってクリアされます。

ビット 3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（QUADSPI_PSMAR）の対応するビットと一致している場合に、自動ポーリングモードでセットされます。CSMF に 1 を書き込むことによってクリアされます。

ビット 2 **FTF**：FIFO 閾値フラグ

インダイレクトモードでは、このビットは FIFO 閾値に達した時点で、またはフラッシュメモリからの読み出し完了後に FIFO 内にデータが残っている場合にセットされます。閾値条件が真でなくなるとすぐに自動的にクリアされます。

自動ポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット 1 **TCF**：転送完了フラグ

このビットは、プログラムされたデータ数が転送された時点でインダイレクトモードでセットされるか、または転送が中止された時点でいずれかのモードでセットされます。また、CTCF に 1 を書き込むことによってクリアされます。

ビット 0 **TEF**：転送エラーフラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときに、インダイレクトモードでセットされます。CTEF に 1 を書き込むことによってクリアされます。

23.5.4 QUADSPI フラグクリアレジスタ（QUADSPI_FCR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
											w1o	w1o		w1o	w1o

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTOF**：タイムアウトフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TOF フラグがクリアされます。

ビット 3 **CSMF**：ステータス一致フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの SMF フラグがクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TCF フラグがクリアされます。

ビット 0 **CTEF**：転送エラーフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TEF フラグがクリアされます。

23.5.5 QUADSPI データ長レジスタ（QUADSPI_DLR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DL[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DL[31: 0]**：データ長

インダイレクトモードおよびステータスポーリングモードで取得されるデータ数（値 + 1）ステータスポーリングモードの場合、3 より大きい値（4 バイトを示す）を使用する必要があります。

インダイレクトモードの すべて1は、不定長を意味します。この場合、FSIZE によって定義されているメモリの終了まで QUADSPI が継続します。

0x0000_0000：1 バイトが転送されます。

0x0000_0001：2 バイトが転送されます。

0x0000_0002：3 バイトが転送されます。

0x0000_0003：4 バイトが転送されます。

...

0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。

0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。

0xFFFF_FFFF：不定長 -- すべてのバイトがフラッシュメモリの終了（FSIZE によって定義）まで転送されます。FSIZE = 0x1F の場合、読み出しが無限に続けられます。

デュアルフラッシュモード（DFM=1）では、DL[0] は、たとえこのビットに 0 が書き込まれても 1 のままとなります。したがって、アクセスのたびに必ず偶数バイトが転送されます。

このフィールドはメモリマップドモード（FMODE = 10）では効果がありません。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR）

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DDRM	DHHC	FRCM	SIOO	FMODE[1:0]		DMODE		Res.	DCYC[4:0]					ABSIZE	
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABMODE		ADSIZE		ADMODE		IMODE		INSTRUCTION[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DDRM**：データレートモードを倍にします。

このビットは、アドレス、オルタネートバイト、およびデータフェーズの DDR モードをセットします。

0：DDR モードは無効です。

1：DDR モードは有効です。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 30 **DHHC**：DDR ホールド

DDR モードで、データの出力を 1/4 QUADSPI 出力クロックサイクルだけ遅延させます。

0：アナログ遅延を使用してデータ出力を遅延させます。

1：データの出力を 1/4 QUADSPI 出力クロックサイクルだけ遅延させます。

この機能は、DDR モードでのみアクティブです。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 29 **FRCM**：フリーランニングクロックモード

このビットがセットされている場合、QUADSPI ペリフェラルは FMODE ビットに関係なく、フリーランニングクロックモードに入ります。

0：通常モード

1：フリーランニングクロックモード

このビットは、BUSY が 0 のときにのみ書き込みができます。

ビット 28 **SIOO**：モードにつき 1 回のみ命令を送信します。
*セクション 23.3.13： 825 ページの1 回限りの命令の送信*を参照してください。このビットは、IMODE = 00 のときには効果がありません。
0：トランザクションごとに命令を送信します。
1：最初のコマンドに対してのみ命令を送信します。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 27:26 **FMODE[1:0]**：機能モード
このフィールドは、動作の QUADSPI 機能モードを定義します。
00：インダイレクト書き込みモード
01：インダイレクト読み出しモード
10：自動ポーリングモード
11：メモリマップドモード
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 25:24 **DMODE[1:0]**：データモード
このフィールドは、動作のデータフェーズのモードを次のように定義します。
00：データはありません。
01：データは 1 本のライン上にあります。
10：データは 2 本のライン上にあります。
11：データは 4 本のライン上にあります。
また、このフィールドは動作のダミーフェーズモードも決定します。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **DCYC[4:0]**：ダミーサイクル数
このフィールドはダミーフェーズの処理時間を定義します。SDR モードと DDR モードの両方で、CLK サイクルの数を指定します（0～31）。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 17:16 **ABSIZE[1:0]**：オルタネートバイトサイズ
このビットはオルタネートバイトのサイズを定義します。
00：8 ビットオルタネートバイト
01：16 ビットオルタネートバイト
10：24 ビットオルタネートバイト
11：32 ビットオルタネートバイト
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 15:14 **ABMODE[1:0]**：オルタネートバイトモード
このフィールドは、動作のオルタネートバイトフェーズのモードを次のように定義します。
00：オルタネートバイトはありません。
01：オルタネートバイトは 1 本のライン上にあります。
10：オルタネートバイトは 2 本のライン上にあります。
11：オルタネートバイトは 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ
このビットはアドレスサイズを定義します。
00：8 ビットアドレス
01：16 ビットアドレス
10：24 ビットアドレス
11：32 ビットアドレス
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 11:10 **ADMODE[1:0]**：アドレスモード
このフィールドは、動作のアドレスフェーズのモードを次のように定義します。
00：アドレスはありません。
01：アドレスは 1 本のライン上にあります。
10：アドレスは 2 本のライン上にあります。
11：アドレスは 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 9:8 **IMODE[1:0]**：命令モード
このフィールドは、動作の命令フェーズのモードを次のように定義します。
00：命令はありません。
01：命令は 1 本のライン上にあります。
10：命令は 2 本のライン上にあります。
11：命令は 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 7：0 **INSTRUCTION[7: 0]**：命令
外部 SPI デバイスに送信される命令
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.7 QUADSPI アドレスレジスタ（QUADSPI_AR）

アドレスオフセット：0x0018

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDRESS[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRESS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ADDRESS[31:0]**：アドレス
外部フラッシュメモリに送信されるアドレス
このフィールドへの書き込みは、BUSY = 0 または FMODE = 11 （メモリマップドモード）の場合は無視されます。
デュアルフラッシュモードでは、アドレスは常に偶数でなければならないので、ADDRESS[0] は自動的に 0 に固定されます。

23.5.8 QUADSPI オルタネートバイトレジスタ（QUADSPI_ABR）

アドレスオフセット：0x001C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALTERNATE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTERNATE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ALTERNATE[31: 0]**：オルタネートバイト

アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.9 QUADSPI データレジスタ（QUADSPI_DR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATA[31: 0]**：データ

外部 SPI デバイスに対して送受信されるデータ
インダイレクト書き込みモードでは、このレジスタに書き込まれたデータはいったん FIFO に格納されてから、データフェーズ時にフラッシュメモリに送信されます。FIFO がフルの場合、書き込まれるデータ量を受け入れるだけの十分な空間が FIFO にできるまで、書き込み動作は停止されます。
インダイレクト読み出しモードでは、このレジスタを読み出すことで（FIFO 経由）、フラッシュメモリから受信したデータが提供されます。FIFO のバイト数が読み出し動作で要求したバイト数ほど多くない場合で、BUSY = 1 のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読み出し動作は停止されます。
自動ポーリングモードでは、このレジスタはフラッシュメモリから読み出された最後のデータを格納します（マスキングなし）。
このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書き込みモードでの書き込みにより FIFO に追加されるバイト数は、バイト書き込みで 1 バイト、ハーフワード書き込みで 2 バイト、ワード書き込みで 4 バイトです。同様に、インダイレクト読み出しモードでの読み出しにより FIFO から削除されるバイト数は、バイト読み出しで 1 バイト、ハーフワード読み出しで 2 バイト、ワード読み出しで 4 バイトです。インダイレクトモードでのアクセスは、このレジスタのボトムに整列されなければなりません。つまり、バイト読み出しでは DATA[7:0] が、ハーフワード読み出しでは DATA[15:0] が読み出されます。

23.5.10 QUADSPI ポーリングステータスマスクレジスタ（QUADSPI _PSMKR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MASK[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MASK[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MASK[31: 0]**：ステータスマスク

ポーリングモードで受信したステータスバイトに適用されるマスク

ビット n の場合

0：自動ポーリングモードで受信したデータのビット n はマスクされ、その値は一致ロジックでは考慮されません。

1：自動ポーリングモードで受信したデータのビット n はマスクされず、その値は一致ロジックで考慮されます。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.11 QUADSPI ポーリングステータス一致レジスタ（QUADSPI _PSMAR）

アドレスオフセット：0x0028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MATCH[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MATCH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MATCH[31: 0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.12 QUADSPI ポーリングインターバルレジスタ（QUADSPI _PIR）

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTERVAL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INTERVAL[15: 0]**：ポーリングインターバル

自動ポーリングフェーズ時の読み出し間の CLK サイクル数

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.13 QUADSPI 低電力タイムアウトレジスタ（QUADSPI_LPTR）

アドレスオフセット：0x0030

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEOUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TIMEOUT[15: 0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、QUADSPI はその後のバイトをプリフェッチし、それらを FIFO に保持します。このフィールドは、FIFO がフルになってから nCS を立ち上げるまでに QUADSPI が待つ CLK サイクルの数を示し、フラッシュメモリを消費電力がより少ない状態にします。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

23.5.14 QUADSPI レジスタマップ

表 185. QUADSPI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	QUADSPI_CR	PRESCALER[7:0]								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	Res.	FTHRES [4:0]					FSEL	DFM	Res.	SSHIFT	TCEN	Res.	ABORT	EN
	リセット値	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0				0	0	0	0	0	0	0		0	0		0	0
0x0004	QUADSPI_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	CSHT			Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKMODE
	リセット値												0	0	0	0	0						0	0	0								0
0x0008	QUADSPI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[6:0]							Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																		0	0	0	0	0	0	0			0	0	0	0	0	0
0x000C	QUADSPI_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x0010	QUADSPI_DLR	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	QUADSPI_CCR	DDRM	DHHC	FRCM	SIOO	FMODE[1:0]		DMODE[1:0]		Res.	DCYC[4:0]					ABSIZE[1:0]		ABMODE[1:0]		ADSIZE[1:0]		ADMODE[1:0]		IMODE[1:0]		INSTRUCTION[7:0]							
	リセット値	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	QUADSPI_AR	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	QUADSPI_ABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	QUADSPI_DR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	QUADSPI_ PSMKR	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	QUADSPI_ PSMAR	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	QUADSPI_PIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	QUADSPI_ LPTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては *セクション 2.2.2* を参照してください。

24 遅延ブロック（DLYB）

24.1 概要

遅延ブロック（DLYB）は、入力クロックからディフェーズ(位相をずら)した出力クロックを生成するために使用します。出力クロックのフェーズは、ユーザアプリケーションでプログラムする必要があります。次に、出力クロックを使用して、SDMMC や QUADSPI インタフェースなどの別のペリフェラルで受信したデータをクロックします。

遅延は電圧や温度に依存するため、場合によってはその受信データを用いて出力クロックフェーズの設定と中央揃えを再度行うためのアプリケーションが必要かもしれません。

24.2 DLYB の主な機能

遅延ブロックは、次のような機能を備えています。

- 25～208 MHz の入力クロック周波数
- 最大 12 のオーバーサンプリングフェーズ

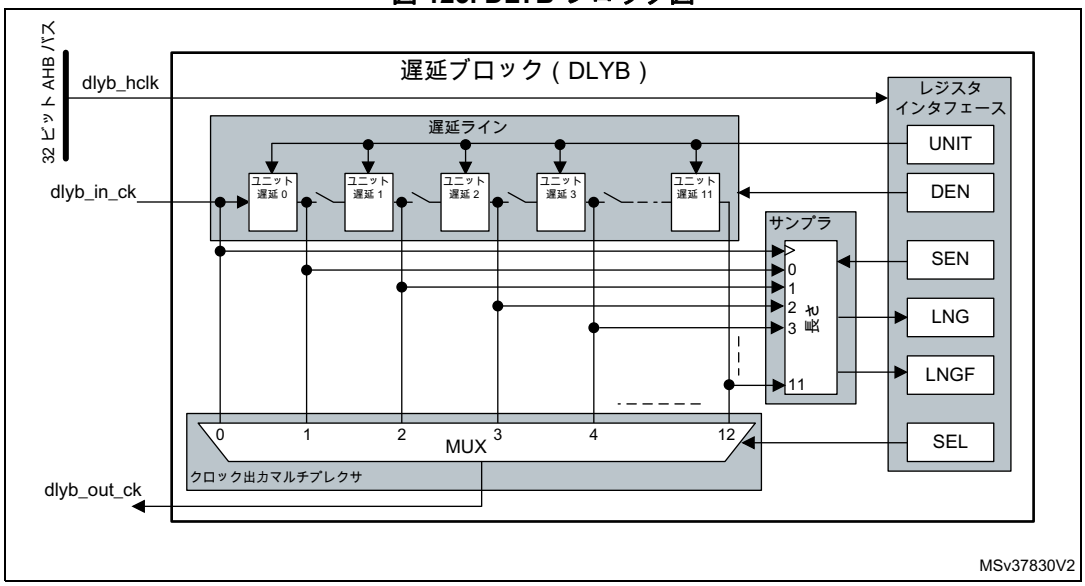
24.3 DLYB の機能詳細

24.3.1 DLYB 図

遅延ブロックは、次のようなサブブロックから成ります。

- 遅延ブロックレジスタへの AHB アクセスを可能にするレジスタインタフェースブロック
- ユニット遅延に対応する遅延ライン
- 遅延ライン長さサンプラ
- 出力クロック選択マルチプレクス

図 123. DLYB ブロック図



24.3.2 DLYB ピンおよび内部信号

表 186 は DLYB 内部信号のリストです。

表 186. DLYB 内部入力／出力信号

信号名	信号タイプ	説明
dlyb_hclk	デジタル入力	遅延ブロックレジスタのインタフェースクロック
dlyb_in_ck	デジタル入力	遅延ブロック入力クロック
dlyb_out_ck	デジタル出力	遅延ブロック出力クロック

24.3.3 概要

遅延ブロックは、DLYB_CR レジスタの DEN ビットを設定することによって有効にします（*セクション 24.4.1 : DLYB 制御レジスタ（DLYB_CR）* を参照）。長さサンプラは、DLYB_CR レジスタの SEN ビットによって有効にします。

遅延ブロックが有効なとき、ユニット遅延によって追加された遅延は DLYB_CFGR レジスタの UNIT ビットで定義します（*セクション 24.4.2 : DLYB 設定レジスタ（DLYB_CFGR）* を参照）。なお、UNIT ビットは出力クロックが無効（SEN = 1）のときのみプログラム可能です。

遅延ブロックが有効なとき、出力クロックフェーズは DLYB_CFGR レジスタの SEL ビットによって選択します。なお、SEL は出力クロックが無効（SEN = 1）のときのみプログラム可能です。

出力クロックをディフェーズする前に、遅延ライン長さを入力クロックの 1 周期に対して設定する必要があります。遅延ライン長さは SEN ビットによって長さサンプラを有効にすることで設定でき、それにより DLYB_CFGR の遅延ライン長さ（LNG ビット）および長さ有効フラグ（LNGF）へのアクセスが可能になります。

遅延ライン長さが設定されると、出力クロックマルチプレクスによって、ディフェーズされた出力クロックを選択することができます。これは SEL ビットを使って行います。出力クロックは SEN が 0 に設定されている場合に選択されたフェーズでのみ使用できます。

表 187 に、遅延ブロック制御の概要を示します。

表 187. 遅延ブロック制御

DEN	SEN	UNIT	SEL	LNG	LNGF	出力クロック
0	0	無視	無視	無視	無視	有効（= 入力クロック）
x	1	ユニット遅延	出力クロックフェーズ	長さ	長さフラグ	無効
1	0	ユニット遅延 (1)	出力クロックフェーズ (2)	無視	無視	有効（= 選択されたフェーズ）

1. ユニット遅延は SEN = 1 の場合にのみ変更できます。
2. 出力クロックフェーズは SEN = 1 の場合にのみ変更できます。

24.3.4 遅延ライン長さ設定手順

LNG ビットを使用して、入力クロック周期に対する遅延ライン長さを決定します。遅延ライン長さは、入力クロックの 1 周期全体をカバーできるような長さに設定しなければなりません。

入力クロック 1 周期に対する遅延ライン長さを設定するには、以下のシーケンスに従います。

1. DEN ビットを 1 に設定して遅延ブロックを有効にします。
2. SEN ビットを 1 に設定して長さサンプリングを有効にします。
3. SEL ビットを 12 に設定してすべての遅延セルを有効にします。
4. UNIT = 0〜127 の場合（遅延ライン長さが設定されるまで、このステップを繰り返すこと）:
   a) UNIT の値を更新し、長さフラグ LNGF が 1 に設定されるのを待ちます。
   b) LNG ビットを読み出します。
   LNG[10:0] > 0、かつ LNG[11] または LNG[10] = 0 の場合、遅延ライン長さは入力クロック 1 周期に対して設定されます。
5. 入力クロック 1 周期の間に発生するユニット遅延の数（N）を決定します。
   – N = 10〜0 の場合：
   LNG[N] = 1 の場合、入力クロック周期中のユニット遅延の数 = N
6. SEN ビットを 0 クリアして長さサンプリングを無効にします。

24.3.5 出力クロックフェーズ設定手順

遅延ライン長さを入力クロック 1 周期に対して設定する場合、入力クロック 1 周期の期間中に発生するユニット遅延の間で出力クロックフェーズを選択することができます。

出力クロックフェーズを選択するには以下の手順に従います。

1. 出力クロックを無効にし、SEN ビットを 1 に設定することでフェーズ選択 SEL ビットへのアクセスを有効にします。
2. 任意の出力クロックフェーズ値を SEL ビットに書き込みます。
3. SEN を 0 クリアすることにより、選択されたフェーズで出力クロックを有効にします。

24.4 DLYB レジスタ

すべてのレジスタはワード、ハーフワード、バイトでアクセスすることができます。

24.4.1 DLYB 制御レジスタ（DLYB_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SEN**：サンプラ長さ有効ビット

0：サンプラ長と UNIT にアクセスするレジスタと SELを無効化、出力クロックを有効化

1：サンプラ長と UNIT にアクセスするレジスタと SELを有効化、出力クロックを無効化

ビット 0 **DEN**：遅延ブロック有効ビット

0：遅延ブロックは無効です。

1：遅延ブロックは有効です。

24.4.2 DLYB 設定レジスタ（DLYB_CFGR）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

ビット 31 **LNGF**：長さ有効フラグ

このフラグは UNIT ビットの変更後、LNG ビットに含まれる遅延ライン長さの値が有効なときを知らせます。

0：LNG の長さの値は無効です。

1：LNG の長さの値は有効です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **LNG**：遅延ライン長さの値

これらのビットには、入力クロックの立ち上がりエッジでサンプリングされた 12 個のユニット遅延の値が反映されます。

値は LNGF = 1 のときにのみ有効です。

ビット 15　予約済みであり、リセット値に保持する必要があります。

ビット 14:8　**UNIT**：ユニット遅延セルの遅延の定義
これらのビットは、SEN = 1 のときにのみ書き込むことができます。
ユニット遅延 = 初期遅延 + UNIT x 遅延ステップ

ビット 7:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**SEL**：出力クロックのフェーズ選択
これらのビットは、SEN = 1 のときにのみ書き込むことができます。
出力クロックフェーズ = 入力クロック + SEL x ユニット遅延

24.4.3　DLYB レジスタのマップ

表 188. DLYB レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**DLYB_CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEN	DEN
	リセット値																															0	0
0x004	**DLYB_CFGR**	LNGF	Res.	Res.	Res.	LNG												Res.	UNIT							Res.	Res.	Res.	Res.	SEL			
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0					0	0	0	0

レジスタ境界アドレスについては *104 ページのセクション 2.2.2* を参照してください。

25 アナログデジタルコンバータ（ADC）

25.1 概要

このセクションでは、最大 3 つ実装されている ADC について説明します。

- ADC1 および ADC2 は密接に結合されており、デュアルモードで動作できます（ADC1 はマスタです）。
- ADC3 は独立してインスタンス化されます。

各 ADC は、16 ビットの逐次比較型アナログデジタルコンバータで構成されています。

各 ADC には、最大 20 の多重化チャネルがあります。さまざまなチャネルの A/D 変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADC の結果は、左詰めまたは右詰めで 32 ビットのデータレジスタに格納されます。

ADC は AHB バスに配置され、データを高速で処理できます。

アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

組み込みのオーバーサンプリング回路により、CPU の計算負荷を軽減しながらアナログ性能を高めることができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードが実装されています。

25.2 ADC の主な機能

- ハイパフォーマンス機能
  - 最大 2つの ADC、デュアルモードで動作可能
  - 16、14、12、10、または 8 ビットに設定可能な分解能
  - ADC 変換時間は AHB バスクロック周波数から独立
  - 分解能を下げることで変換時間をより短縮できる
  - シングルエンドまたは差動入力（チャネルごとにプログラム可能）を管理可能
  - 高速データ処理を可能にする AHB スレーブバスインタフェース
  - 自己較正（オフセットと直線性とも）
  - チャネル単位でプログラム可能なサンプリング時間
  - 最大 4 つのインジェクトチャネル（アナログ入力をレギュラまたはインジェクトチャネルへ完全に割り当て設定可能）
  - 高速コンテキスト切り替えを可能にするためにインジェクトチャネルの内容を準備するハードウェアによる支援
  - 組み込みのデータコヒーレンシによるデータ配置
  - データは、FIFO を用いたレギュラチャネル変換において、GP-DMA で管理可能
  - データは、事後処理のために DFSDM に送信可能
  - 4 つのインジェクトチャネル専用のデータレジスタ
- オーバーサンプリング回路
  - 32 ビットデータレジスタ
  - 2～1024 倍までオーバーサンプリング比を調整可能
  - プログラム可能なデータの右左シフト
- 低電力機能
  - 低周波数での動作時に ADC 消費電流を低減するための速度適応型低電力モード
  - 最適な ADC 性能を維持しながら、低いバス周波数を適用可能
  - 低 AHB バスクロック周波数アプリケーション（自動遅延モード）で ADC オーバーランを避けるための自動制御を提供
- 各 ADC には外部アナログ入力チャネルがあります。
  - 専用の GPIO パッドから最大 6 つの高速チャネル
  - 専用の GPIO パッドから最大 14 個の低速チャネル
- さらに、専用の内部チャネルが 5 個
  - 内部基準電圧（V_{REFINT}）は ADC3 に接続
  - 内部温度センサ（V_{TS}）は ADC3 に接続
  - V_{BAT} 監視チャネル（$V_{BAT}/4$）は ADC3 に接続
  - 内部 DAC チャネル 1 とチャネル 2 は ADC2 に接続
- 変換開始は、次のように開始できます。
  - レギュラ変換とインジェクト変換の両方について、ソフトウェアによって開始
  - レギュラ変換とインジェクト変換の両方について、極性が設定可能なハードウェアトリガによって開始（内部タイマイベントまたは GPIO 入力イベント）
- 変換モード
  - 各 ADC では単一チャネルの変換、または一連のチャネルをスキャンできます。
  - シングルモードは、選択された入力をトリガごとに1回変換します。

  - 連続モードは、選択された入力を連続的に変換します。
  - 不連続モード
- ADC1 および 2 のデュアル ADC モード
- ADC レディ時、サンプリング終了時、（レギュラまたはインジェクト）変換終了時、（レギュラまたはインジェクト）シーケンス変換終了時、アナログウォッチドッグ 1、2、3、またはオーバーランイベント時に割り込みを生成
- ADC ごとの 3 つのアナログウォッチドッグ
- ADC 電源仕様：1.62～3.6 V
- ADC 入力電圧範囲：$V_{REF-} \leq V_{IN} \leq V_{REF+}$

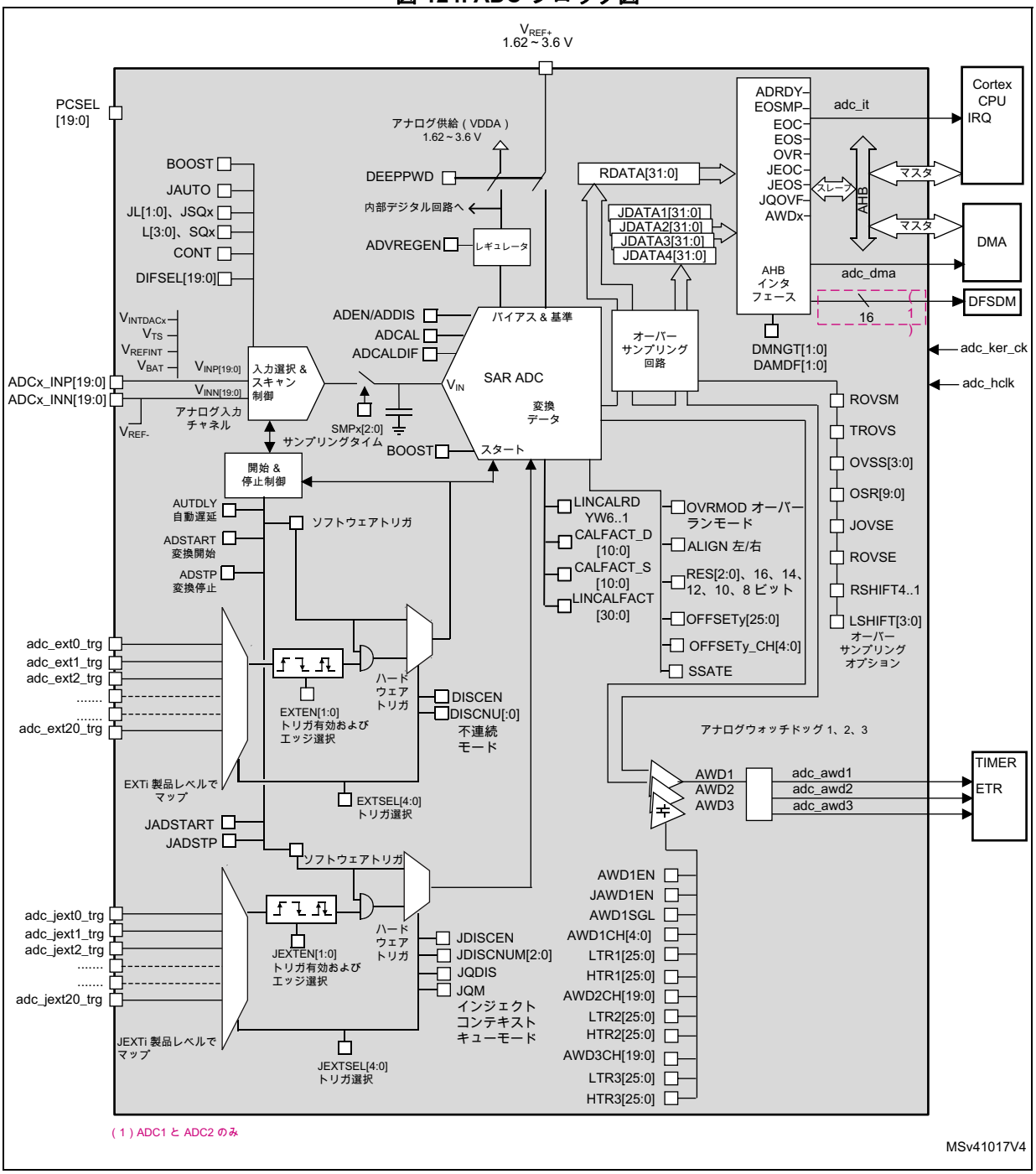
図 124 に、1 つの ADC のブロック図を示します。

25.3 ADC の機能詳細

25.3.1 ADC ブロック図

図 124 に ADC ブロック図を、*表 190* に ADC ピンの説明を示します。

図 124. ADC ブロック図



25.3.2 ADC ピンと内部信号

表 189. ADC 内部信号

内部信号名	信号タイプ	説明
adc_ext_trg[20:0]	デジタル入力	レギュラ変換の最大 21 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_jext_trg[20:0]	デジタル入力	インジェクト変換の最大 21 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_awd1 adc_awd2 adc_awd3	デジタル出力	オンチップタイマに接続された内部アナログウォッチドッグの出力信号（x = アナログウォッチドッグの数 1、2、3）
V_{TS}	アナログ入力	内部温度センサからの出力電圧
V_{REFINT}	アナログ入力	内部基準電圧からの出力電圧
V_{BAT}	アナログ入力	外部バッテリ電圧供給
adc_it	デジタル出力	ADC 割り込み
adc_hclk	デジタル入力	AHB クロック
adc_ker_ck	デジタル入力	ADC カーネルクロック
adc_dma	デジタル出力	ADC DMA リクエスト

表 190. ADC ピン

名前	信号タイプ	コメント
V_{REF+}	入力、アナログ基準電圧正	ADC のハイレベル／正基準電圧、 1.62 V ≤ V_{REF+} ≤ V_{DDA}
V_{DDA}	入力、アナログ電源供給	V_{DDA} に等しいアナログ電源供給： 1.62 V ≤ V_{DDA} ≤ 3.6 V
V_{REF-}	入力、アナログ基準電圧負	ADC のローレベル／負基準電圧、 V_{REF-} = V_{SSA}
V_{SSA}	入力、アナログ供給グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド
V_{INP}[19:0]	各 ADC の正のアナログ入力チャネル	外部チャネルADC_INP*i* または内部チャネルのいずれかに接続
V_{INN}[19:0]	各 ADC の負のアナログ入力チャネル	V_{REF-} または外部チャネルADC_INN*i-1* に接続
ADCx_INP[19:0]	外部アナログ入力信号	最大 20 アナログ入力チャネル（x = ADC 数 = 1、2、または 3）： – ADCx_INP[0:5] 高速チャネル – ADCx_INP[6:19] 低速チャネル
ADCx_INN[19:0]		最大 20 アナログ入力チャネル（x = ADC 数 = 1、2、または 3）： – ADCx_INN[0:5] 高速チャネル – ADCx_INN[6:19] 低速チャネル
PCSEL[19:0]	出力、プリチャネル選択制御信号	チャネルの事前選択のために GPIO に接続

25.3.3 クロック

デュアルクロックドメインアーキテクチャ

デュアルクロックドメインアーキテクチャは、ADC クロックが AHB バスクロックから独立していることを意味します。

入力クロックは、3 つの ADC に対して同じで、2 つの異なるクロックソースから選択できます（*図 125 : ADC クロック構成*を参照）。

1. ADC クロックは、AHB クロックから独立し、非同期の adc_ker_ck という名前の特定のクロックソースにできます。

   このクロックは RCC に設定できます（ADC クロック（adc_ker_ck）専用クロックの生成方法の詳細については、RCC のセクションを参照してください）。

   この構成を選択するには、ADCx_CCR レジスタの CKMODE[1:0] ビットをリセットする必要があります。

2. ADC クロックは、ADC バスインタフェースの AHB クロックから生成し、プログラム可能な係数（1、2、または 4）で分周することができます。このモードでは、プログラム可能な分周比を選択できます（ビット CKMODE[1:0] に従って /1、2、または 4）。

   この構成を選択するには、ADCx_CCR レジスタの CKMODE[1:0] ビットが「00」以外の値である必要があります。

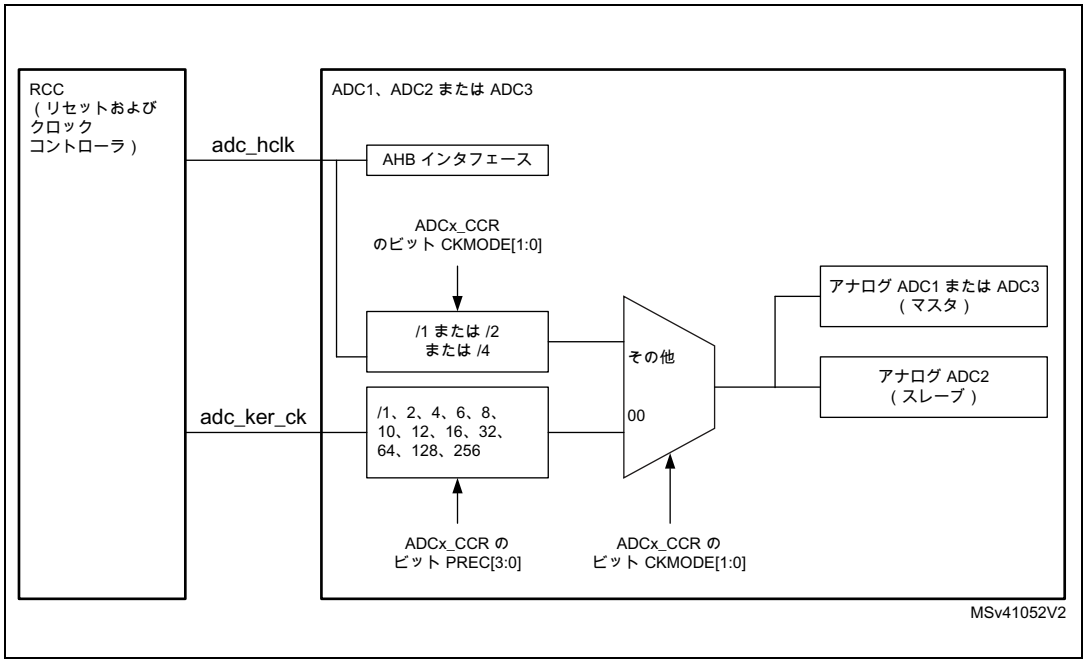
注： ***オプション b）では、AHB プリスケーラが 1（RCC_CFGR レジスタで HPRE[3:0] = 0xxx）の場合のみ、プリスケーラ係数を 1（CKMODE[1:0]=01）にできます。***

オプション a）には、選択された AHB クロック構成にかかわらず、最大の ADC クロック周波数に到達できるという利点があります。ADC クロックは、最終的には1、2、4、6、8、10、12、16、32、64、128、256 の比率で分周できます。このとき、ADCx_CCR レジスタの PRESC[3:0] ビットで設定されたプリスケーラを使用します。

オプション b) には、クロックドメインの再同期を迂回するという利点があります。これは、ADC がタイマによってトリガされるときと、アプリケーションが ADC の確実で精密なトリガを必要とする場合に便利です（そうしないと、トリガインスタンスの不確実性は、2 つのクロックドメイン間の再同期によって高まります）。

CKMODE[1:0] ビットで設定されたクロックは、製品データシートに指定された動作周波数と一致している必要があります。

図 125. ADC クロック構成



1. adc_hclk および adc_ker_ck の生成の詳細については、RCC のセクションを参照してください。

ADC クロックと AHB クロックの間におけるクロック比の制約

通常、ADC クロックと AHB クロックの間で注意すべき制約はありませんが、インジェクトチャネルがプログラムされている場合は例外です。この場合、次の比率に注意する必要があります。

- すべてのチャネルの分解能が 16 ビット、14 ビット、12 ビットまたは 10 ビットである場合、$F_{HCLK} >= F_{ADC} / 4$
- いくつかのチャネルの分解能が 8 ビットである場合（それを下回る分解能のチャネルがない場合）、$F_{HCLK} >= F_{ADC} / 3$

ブーストビット制御

ADCx_CR レジスタには、ADC ブースト制御ビット BOOST があります。

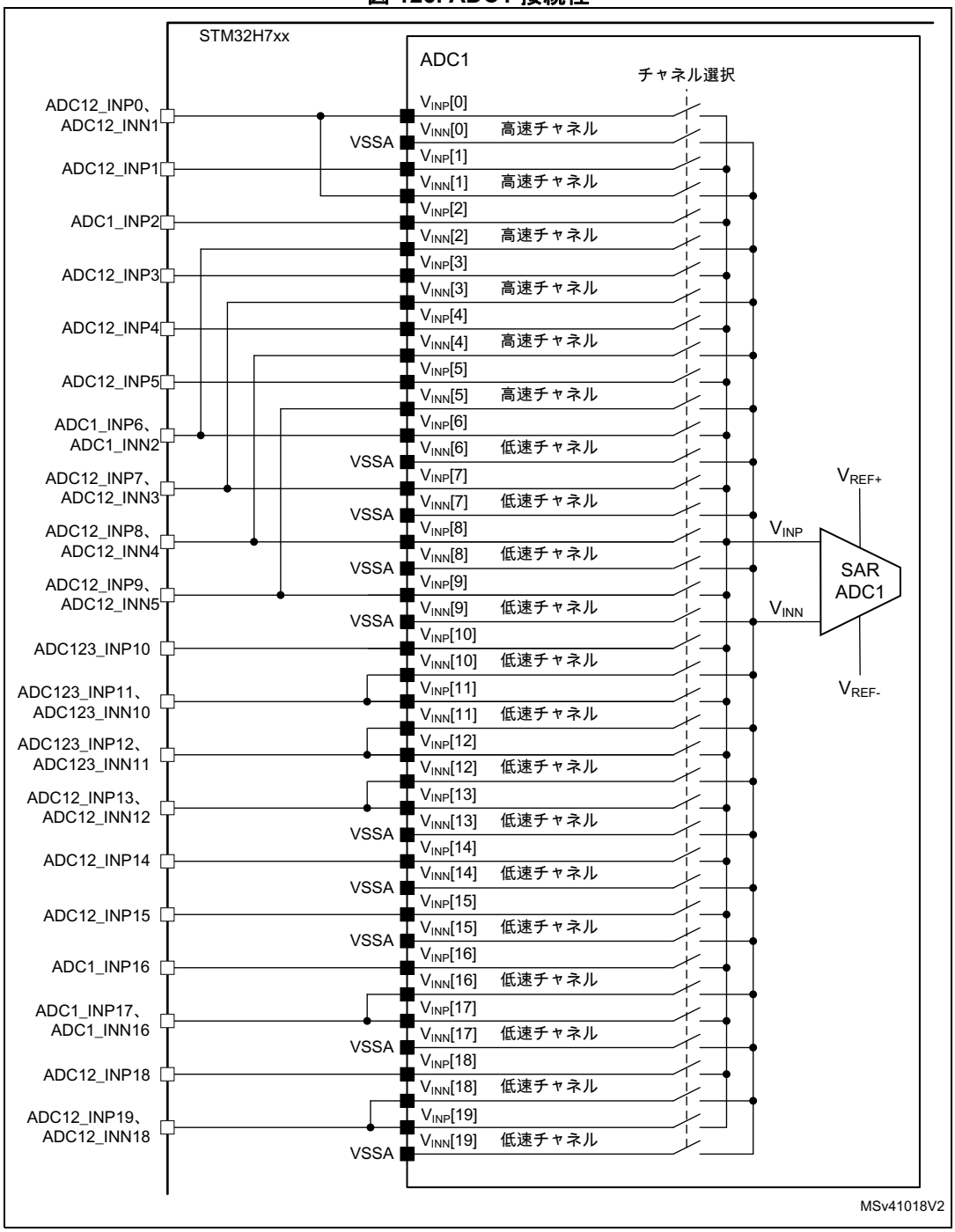
ADC クロックが 20 MHz を超える場合に、このビットをセットする必要があります。ADC クロックが 20 MHz 未満の場合は、このビットをクリアして電力を低減することができます。

25.3.4 ADC1/2/3 接続性

ADC1 および ADC2 は密接に結合されており、以下の図に示すいくつかの外部チャネルを共有しています。

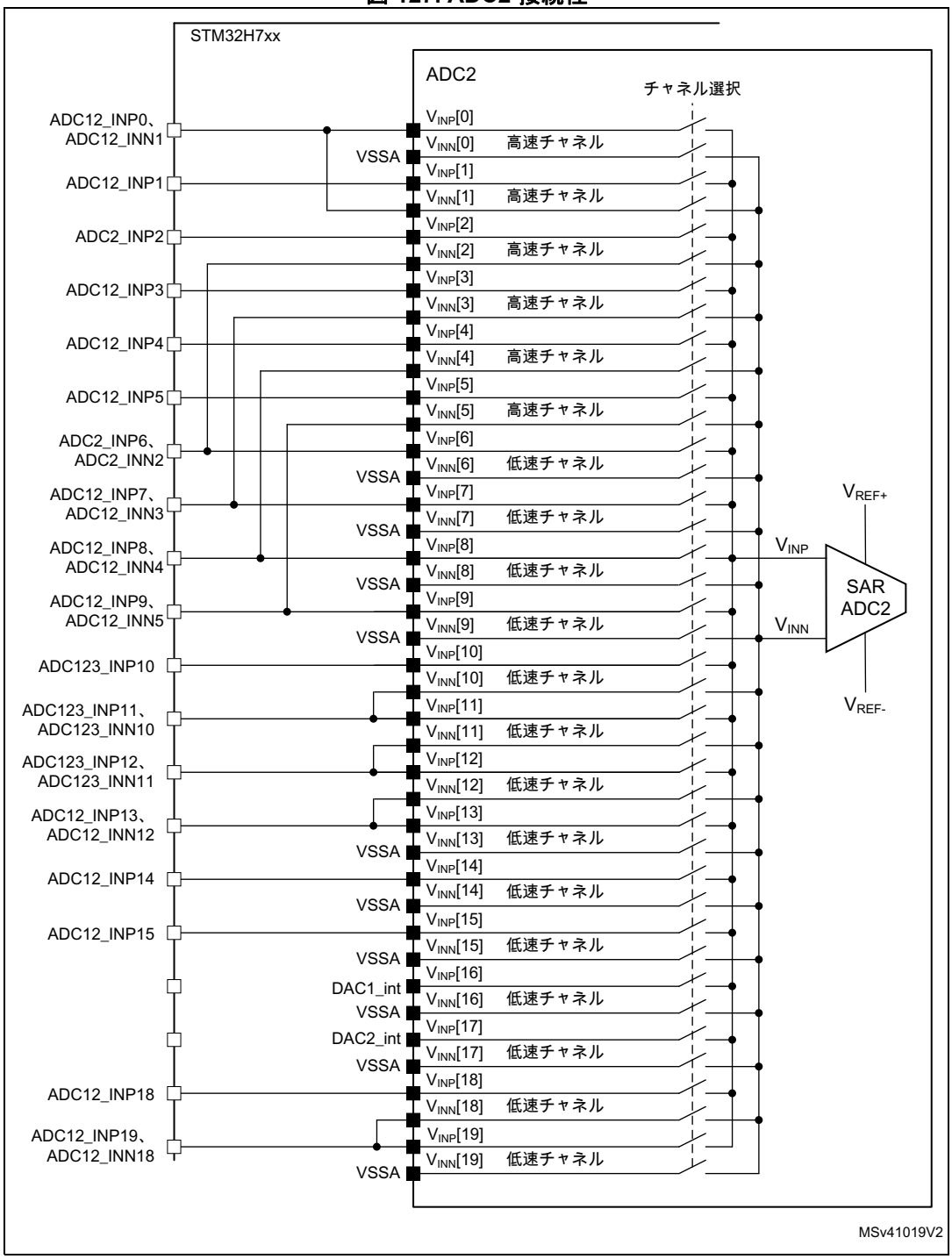
ADC3 は独立でインスタンス化されますが、入力の一部は ADC1 および ADC2 と共有となっています。

図 126. ADC1 接続性



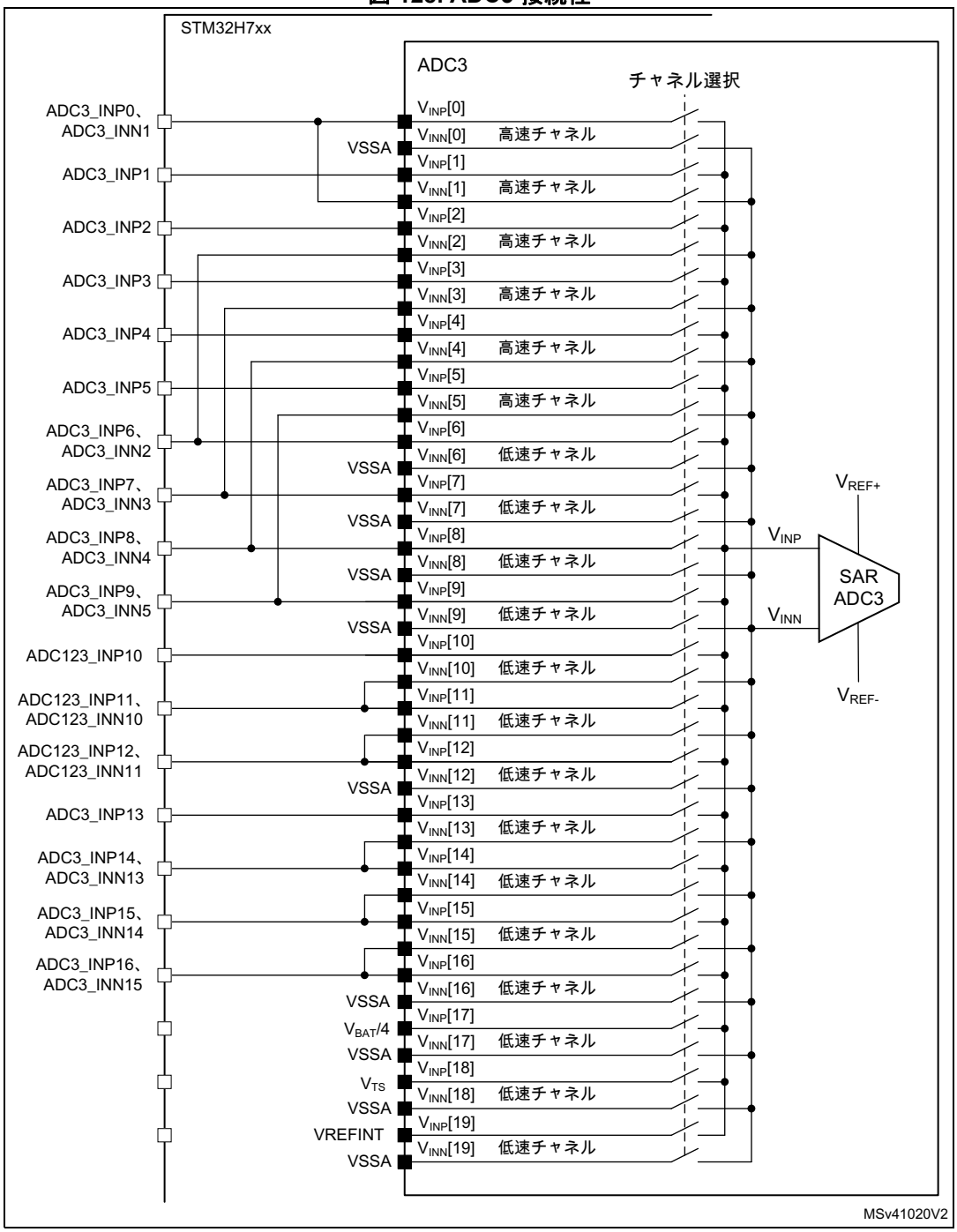
1. 対応する ADC 入力チャネルが差動モードとして設定された場合に限り、ADCx_INNy 信号を使用できます。

図 127. ADC2 接続性



1. 対応する ADC 入力チャネルが差動モードとして設定された場合に限り、ADCx_INNy 信号を使用できます。

図 128. ADC3 接続性



1. 対応する ADC 入力チャネルが差動モードとして設定された場合に限り、ADCx_INNy 信号を使用できます。

25.3.5 スレーブ AHB インタフェース

ADC は、制御／ステータスレジスタおよびデータアクセスのために AHB スレーブポートを実装しています。AHB インタフェースの機能を次に示します。

- ワード（32 ビット）アクセス
- シングルサイクルレスポンス
- レジスタへのすべての読み出し／書き込みアクセスに対してゼロウェイトステートの応答。

AHB スレーブインタフェースでは分割／再試行リクエストをサポートしておらず、また AHB エラーも発生しません。

25.3.6 ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）

デフォルトでは、リーク電流を低減するために電源が内部的にオフになっている（ADCx_CR レジスタのビット DEEPPWD のリセット状態が 1 である）場合、ADC はディープパワーダウンモードになります。

ADC の動作を開始するには、まずビット DEEPPWD=0 にクリアしてディープパワーダウンモードを終了する必要があります。

そして、ADCx_CR レジスタでビット ADVREGEN=1 にセットすることによって、ADC 内部電圧レギュレータを有効にする必要があります。ソフトウェアは、較正を起動するか ADC を有効化する前に、ADC 電圧レギュレータの起動時間の間待つ必要があります。（$T_{ADCVREG_STUP}$）この遅延は、ソフトウェアによって実装する必要があります。

ADC 電圧レギュレータの起動時間については、デバイスデータシートで $T_{ADCVREG_STUP}$ パラメータを参照してください。

ADC の動作が完了すると、ADC を無効化できます（ADEN=0）。ADC 電圧レギュレータを無効にすることでも節電できます。これは、ビット ADVREGENに 0 を書き込んで実行します。

そして、リーク電流を削減してさらに節電するには、ADCx_CR レジスタでビット DEEPPWD=1 をセットして ADC ディープパワーダウンモードに再移行することもできます。これは特に STOP モードに移行する前に有益です。

注： ***DEEPPWD=1 を書き込むと、ADC 電圧レギュレータは自動的に無効になり、ビット ADVREGEN が自動的にクリアされます。***

注： ***内部電圧レギュレータが無効なとき（ADVREGEN=0）には、内部アナログ較正が保持されます。***

ADC ディープパワーダウンモード（DEEPPWD=1）では、内部アナログ較正が失われ、較正を再起動するか、以前に保存した較正係数を再適用する必要があります（*セクション 25.3.8：較正（ADCAL、ADCALDIF、ADCALLIN、ADCx_CALFACT）*を参照してください）。

25.3.7 シングルエンドおよび差動入力のチャネル

ADCx_DIFSEL レジスタのビット DIFSEL[19:0] に書き込むことで、チャネルを、シングルエンド入力または差動入力のいずれかに設定できます。この設定は、ADC が無効（ADEN=0）のときに書き込む必要があります。

シングルエンド入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 $V_{INP[i]}$（正の入力）と V_{REF-}（負の入力）の差です。

差動入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 $V_{INP[i]}$（正の入力）と $V_{INN[i]}$（負の入力）の差です。

差動モードの出力データは、符号なしデータです。$V_{INP[i]}$ が VREF- で $V_{INN[i]}$ が VREF+ のときには出力データは 0x0000（16 ビット分解能モード）、$V_{INP[i]}$ が VREF+ で $V_{INN[i]}$ が VREF- のときには出力データは 0xFFFF となります。

$$\text{Converted value} = \frac{\text{ADC_Full_Scale}}{2} \times \left[1 + \frac{V_{INP} - V_{INN}}{V_{REF+}}\right]$$

ADC が差動モードに設定されている場合、両方の入力を V_{REF+} / 2 電圧でバイアスする必要があります。

入力信号は区別します（モード共通の電圧は固定する必要があります）。

各 ADC への入力チャネルの接続方法の詳細については、*図 126 : ADC1 接続性*～*図 128 : ADC3 接続性*を参照してください。

注意： ***差動入力モードでチャネル「i」を設定する場合、負の入力電圧は $V_{INN[i]}$ に接続されます。その結果、$V_{INN[i]}$ に接続されているチャネル「i+n」は、異なるADC で同時に変換しないようにする必要があります。ADC1/ADC2 では一部のチャネルを共有しています。これにより、ほかの ADC のチャネルが使用不可になる場合があります。***

25.3.8 較正（ADCAL、ADCALDIF、ADCALLIN、ADCx_CALFACT）

各 ADC は、ADC のパワーオン／オフシーケンスを含むすべての較正シーケンスを駆動する、自動較正手順を提供しています。この較正処理時、ADC は、次の ADC のパワーオフまで ADC に内部で適用される、オフセットが 11 ビットまたは直線性が 160 ビットの較正係数を計算します。較正処理時、アプリケーションでは ADC を使用してはならず、較正が完了するまで待つ必要があります。

較正は、すべての ADC 動作に先立って行われます。較正によって、チップごとに異なる系統的エラーが除去され、オフセットと直線性の偏差を補償することができます。

シングルエンド入力変換に適用されるオフセット較正係数は、差動入力変換に適用される係数とは異なります。

- シングルエンド入力変換に適用される較正を起動する前に、ADCALDIF=0 を書き込みます。
- 差動入力変換に適用される較正を起動する前に、ADCALDIF=1 を書き込みます。

シングル構成と差動構成のいずれであっても、直線性の補正は一度だけとする必要があります。

- 直線性較正とオフセット較正を同時に実行する較正を起動する前に、ADCALLIN=1 を書き込みます。
- 直線性較正は実行せずにオフセット較正のみを実行する較正を起動する前に、ADCALLIN=0 を書き込みます。

較正は、ビット ADCAL=1 をセットすることによって、ソフトウェアによって開始されます。較正は、ADC が無効のとき（ADEN=0 のとき）だけ開始できます。すべての較正シーケンス時、ADCAL ビットは 1 のままです。較正が完了すると、ハードウェアによってクリアされます。このとき、関連する較正係数は、アナログ ADC と、ADCx_CALFACT レジスタのビット CALFACT_S[10:0] または CALFACT_D[10:0] に内部的に格納されます（シングルエンドであるか差動入力較正であるかによって異なります）。160 ビット直線性較正係数は、ADEN が 1 にセットされた ADCx_CALFACT2 レジスタを用いてアクセスできます。

ADC が無効の場合（ADEN=0）、内部アナログ較正が保持されます。ただし、ADC が長時間無効である場合は、ADC を再度有効にする前に、新しい較正サイクルを実行することを推奨します。

内部アナログ較正は ADC の電源がオフになるたびに失われます（たとえば、製品が STANDBY または VBAT モードになった場合）。この場合、ADC の再較正に時間をかけるのを避けるためには、ソフトウェアが前回の較正時に較正係数を保存していた場合にかぎり、再較正せずに ADCx_CALFACT レジスタおよび ADCx_CALFACT2 レジスタに較正係数を再度書き込むことができます。

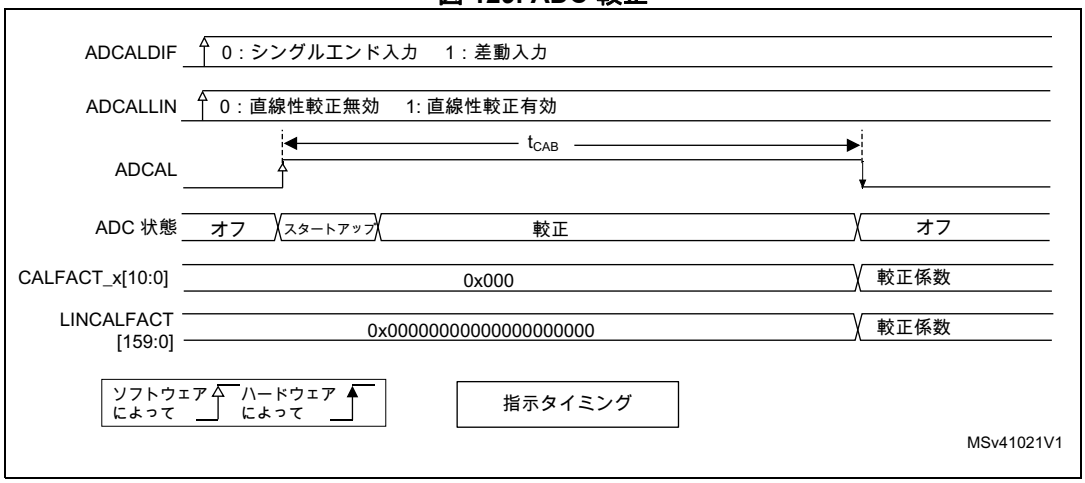
ADC が有効であり、変換中でない場合（ADEN=1 かつ ADSTART=0 かつ JADSTART=0）、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログ ADC に自動的にインジェクトされます。このローディングは透過的であり、変換開始のサイクル遅延は増加しません。V_{REF+} 電圧が 10% を超えて変化した場合、再較正することを推奨します。

較正には、直線性較正に 131,072 ADC クロックサイクル、オフセット較正に 520 ADCクロックサイクルを必要とします。

ソフトウェアでの ADC の較正手順

1. DEEPPWD=0、ADVREGEN=1 であることを確認し、ADC 電圧レギュレータの起動時間が経過していることを確かめます。
2. ADEN=0 であることを確認します。
3. ADCALDIF=0 （シングルエンド入力）または ADCALDIF=1 （差動入力）にセットして、この較正の入力モードを選択します。ADCALLIN=1（有効）であるか ADCALLIN=0（無効）であるかによって、直線性較正をイネーブルするかどうかを選択します。
4. ADCAL=1 にセットします。
5. ADCAL=0 になるまで待ちます。
6. オフセット較正係数を ADCx_CALFACT レジスタから読み出すことができます。
7. *セクション ：直線性較正読出し手順*に記載された手順（ADCx_CALFACT2 レジスタにアクセスする前に ADEN を 1 にセットする必要があります）に従って、直線性較正係数を ADCx_CALFACT2 レジスタから読み出すことができます。

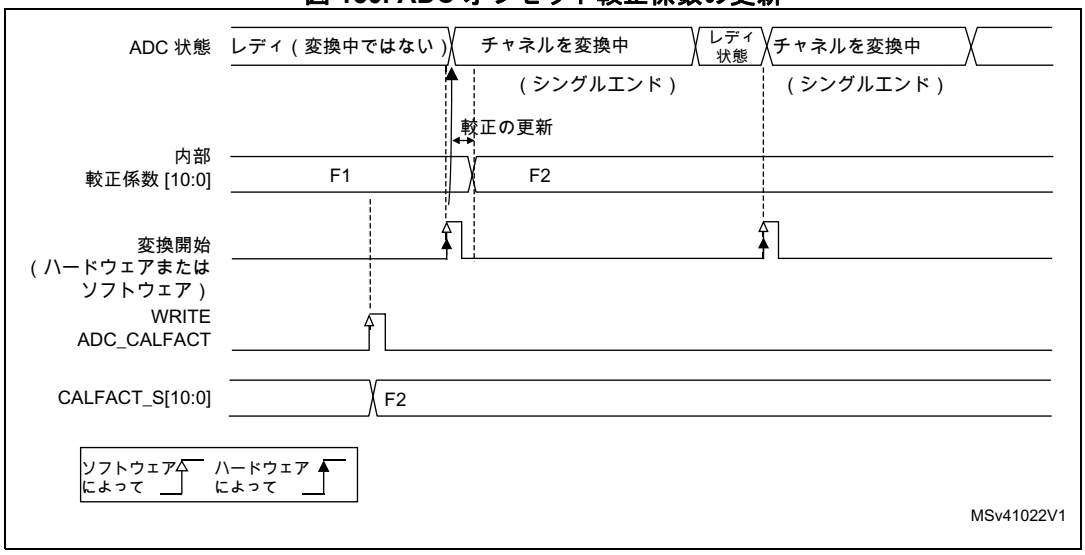
図 129. ADC 較正



ソフトウェアでの ADC への較正係数の再インジェクト手順

1. ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効で変換中でない）ことを確認します。
2. 新しいオフセット較正係数で CALFACT_S および CALFACT_D を書き込みます。
3. *セクション：直線性較正書込み手順*に記載された手順に従って、新しい直線性較正係数で LINCALFACT ビットを書き込みます。
4. 変換を起動すると、較正係数がアナログ ADC にインジェクトされます。これは、内部アナログ較正係数が、シングルエンド入力チャネルの場合はビット CALFACT_S、差動入力チャネルの場合はビット CALFACT_D に格納された較正係数と異なる場合にのみ有効です。

図 130. ADC オフセット較正係数の更新



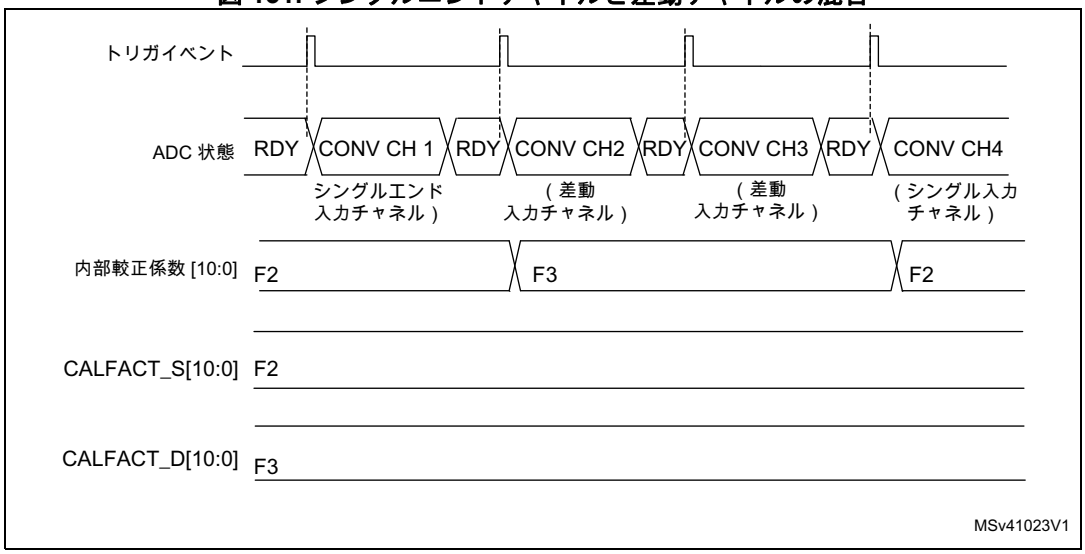
参考資料

シングル ADC によるシングルエンドおよび差動アナログ入力の変換

ADC で、差動入力とシングルエンド入力の両方を変換することを想定している場合、ADCALDIF=0 と ADCALDIF=1 の 2 つの較正を実行する必要があります。手順は、次のとおりです。

1. ADC を無効にします。
2. シングルエンド入力モード（ADCALDIF=0）で直線性較正を有効として（ADCALLIN=1）ADC を較正します。これにより、レジスタ CALFACT_S[10:0] および LINCALFACT[159:0] が更新されます。
3. 差動入力モード（ADCALDIF=1）で直線性較正を無効として（ADCALLIN=0）ADC を較正します。これにより、レジスタ CALFACT_D[10:0] が更新されます。
4. ADC を有効化し、チャネルを設定して変換を起動します。シングルエンド入力チャネルから差動入力チャネル（またはその逆）に切り替えられるたびに、アナログ ADC では自動的に較正が行われます。

図 131. シングルエンドチャネルと差動チャネルの混合



直線性較正読出し手順

ADCALLIN=1で 較正が終了すると（ハードウェアによって ADCAL ビットがクリア）、ADCx_CALFACT2 30 ビットレジスタを使って 160 ビット直線性補正係数を読み出しできます（6 回の読込みアクセスが必要です）。

較正が完了すると、ADCx_CR の LINCALRDYW1..6 コントロール／ステータスビットの 6 ビットがセットされます。ADEN が 1 にセットされたときに、これらのビットの 1 つをクリアすると、ADCx_CALFACT2 レジスタの LINCALFACT[29:0] に対して、直線性係数の一部の転送が開始されます。ADCx_CALFACT2 レジスタの読出しが可能となると、そのビットはハードウェアによってリセットされます（ソフトウェアはクリアされるまでそのビットをポーリングする必要があります）。全体の手順は、次のとおりです。

1. DEEPPWD=0、ADVREGEN=1 であり、ADC 電圧レギュレータの起動時間が経過していることを確認します。
2. ADEN = 1 にセットして、ADRDY=1 になるまで待ちます。
3. LINCALRDYW6 ビット（直線性較正レディワード 6）をクリアします。
4. 戻り値がゼロになって、直線性補正ビット [159:150] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW6 ビットをポーリングします。
5. ADCx_CALFACT2[29:0] を読み出します。
6. LINCALRDYW5 ビットをクリアします。
7. 戻り値がゼロになって、直線性補正ビット [149:120] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW5 ビットをポーリングします。
8. ADCx_CALFACT2[29:0] を読み出します。
9. LINCALRDYW4 ビットをクリアします。
10. 戻り値がゼロになって、直線性補正ビット [119:90] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW4 ビットをポーリングします。
11. ADCx_CALFACT2[29:0] を読み出します。
12. LINCALRDYW3 ビットをクリアします。
13. 戻り値がゼロになって、直線性補正ビット [89:60] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW3 ビットをポーリングします。
14. ADCx_CALFACT2[29:0] を読み出します。
15. LINCALRDYW2 ビットをクリアします。
16. 戻り値がゼロになって、直線性補正ビット [59:30] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW2 ビットをポーリングします。
17. ADCx_CALFACT2[29:0] を読み出します。
18. LINCALRDYW1 ビットをクリアします。
19. 戻り値がゼロになって、直線性補正ビット [29:0] が ADCx_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW1 ビットをポーリングします。
20. ADCx_CALFACT2[29:0] を読み出します。

注： ***ソフトウェア は、LINCALRDYWx ビットを一度に 1 ビットだけトグルできます（他のビットは変化しません）。そうしないと、予期しない動作となります。***

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、LINCALRDYW1..6 ビット を書き込むことによって直線性較正係数にアクセスできます。

直線性較正書込み手順

較正が未完了であるか、新しい直線性較正係数が再書込みされると、ADCx_CR の LINCALRDYW1..6 コントロール／ステータスビットの 6 ビットがリセットされます。直線性較正係数を直接強制することもできますし、次の手順を使って再注入することもできます。

1. DEEPPWD=0、ADVREGEN=1 であり、ADC 電圧レギュレータの起動時間が経過していることを確認します。
2. ADEN = 1 にセットして、ADRDY=1 になるまで待ちます。
3. 前もって保存していた直線性較正係数 [159:150] を ADCx_CALFACT2[9:0] に書き込みます。
4. LINCALRDYW6 ビットをセットします。
5. 戻り値が 1 になって、直線性補正ビット [159:150] が有効に書き込まれたことが示されるまで LINCALRDYW6 ビットをポーリングします。
6. 前もって保存していた直線性較正係数 [149:120] を ADCx_CALFACT2[29:0] に書き込みます。
7. LINCALRDYW5 ビットをセットします。
8. 戻り値が 1 になって、直線性補正ビット [149:120] が有効に書き込まれたことが示されるまで LINCALRDYW5 ビットをポーリングします。
9. 前もって保存していた直線性較正係数 [119:90] を ADCx_CALFACT2[29:0] に書き込みます。
10. LINCALRDYW4 ビットをセットします。
11. 戻り値が 1 になって、直線性補正ビット [119:90] が有効に書き込まれたことが示されるまで LINCALRDYW4 ビットをポーリングします。
12. 前もって保存していた直線性較正係数 [89:60] を ADCx_CALFACT2[29:0] に書き込みます。
13. LINCALRDYW3 ビットをセットします。
14. 戻り値が 1 になって、直線性補正ビット [89:60] が有効に書き込まれたことが示されるまで LINCALRDYW3 ビットをポーリングします。
15. 前もって保存していた直線性較正係数 [59:30] を ADCx_CALFACT2[29:0] に書き込みます。
16. LINCALRDYW2 ビットをセットします。
17. 戻り値が 1 になって、直線性補正ビット [59:30] が有効に書き込まれたことが示されるまで LINCALRDYW2 ビットをポーリングします。
18. 前もって保存していた直線性較正係数 [29:0] を ADCx_CALFACT2[29:0] に書き込みます。
19. LINCALRDYW1 ビットをセットします。
20. 戻り値が 1 になって、直線性補正ビット [29:0] が有効に書き込まれたことが示されるまで LINCALRDYW1 ビットをポーリングします。

注： ***ソフトウェアは、LINCALRDYWx ビットを一度に 1 ビットだけトグルできます（他のビットは変化しません）。そうしないと、予期しない動作となります。***

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、LINCALRDYW1..6 ビット を書き込むことによって直線性較正係数を更新できます。

25.3.9 ADC オン / オフ制御（ADEN、ADDIS、ADRDY）

最初に、*セクション 25.3.6：ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）*で説明している手順に従ってください。

一度 DEEPPWD=0 および ADVREGEN=1 になると ADC を有効化することができ、*図 132*にあるように、ADC は正確な変換を開始する前に、安定時間 t_{STAB} を必要とします。ADC は、次の 2 つの制御ビットによって有効化または無効化されます。

- ADC は、ADEN=1 によって有効化されます。ADC の動作準備ができると、ADRDY フラグがセットされます。
- ADC は、ADDIS=1 によって無効化されます。アナログ ADC が実質的に無効になると、ADEN および ADDIS はハードウェアによって自動的にクリアされます。

その後、ADSTART=1 をセットすることによって（*セクション 25.3.19：外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）*を参照）、またはトリガが有効な場合に外部トリガイベントが発生したときに、レギュラ変換を開始できます。

JADSTART=1 をセットすることによって、またはインジェクトトリガが有効な場合に外部インジェクトトリガイベントが発生したときに、インジェクト変換を開始できます。

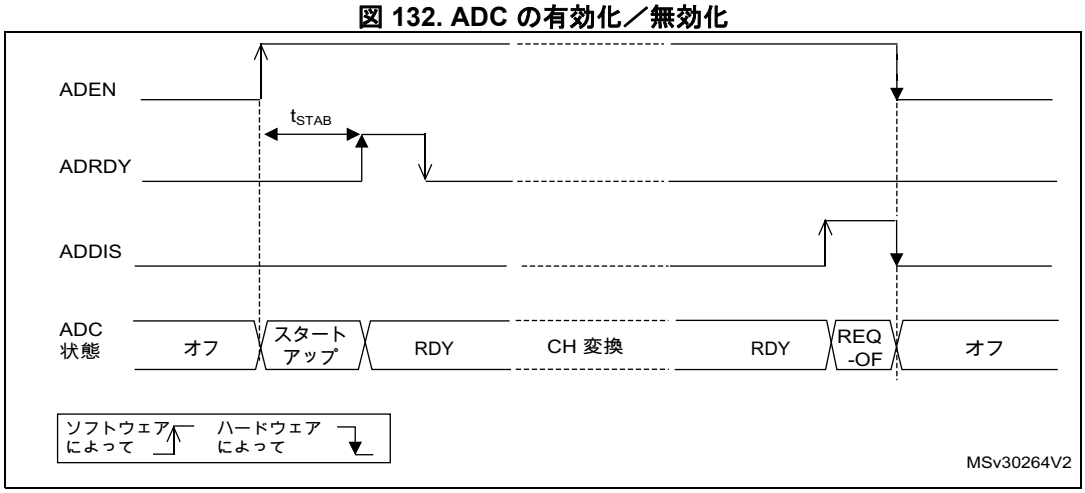
ソフトウェアでの ADC の有効化手順

1. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします。
2. ADEN=1 にセットします。
3. ADRDY=1 になるまで待ちます（ADRDY は ADC 起動時間後にセットされます）。これは、対応する割り込みを使用することで実行できます（ADRDYIE=1 をセットします）。
4. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします（オプション）。

ソフトウェアでの ADC の無効化手順

1. ADSTART=0 かつ JADSTART=0 であることを確認し、変換が実行中でないことを確認します。必要な場合は、ADSTP=1 および JADSTP=1 にセットして、実行中のレギュラ変換とインジェクト変換を停止し、ADSTP=0 および JADSTP=0 になるまで待ちます。
2. ADDIS=1 にセットします。
3. アプリケーションによって必要とされる場合、ADEN=0 になり、アナログ ADC が実質的に無効になるまで待ちます（ADEN=0 になると、ADDIS は自動的にリセットされます）。

図 132. ADC の有効化／無効化



25.3.10 ADC 制御ビット書き込み時の制約

ソフトウェアでは、ADC が無効である場合のみ（ADEN は 0 である必要があります）、ADC クロック（RCC セクションを参照）、ADCx_DIFSEL レジスタの制御ビット DIFSEL、ADCx_CCR レジスタ、および ADCx_CR レジスタの制御ビット ADCAL と ADEN を設定して有効化するために、RCC 制御ビットを書き込むことができます。

ADC が有効であり、ADC を無効にするための保留中のリクエストがない場合のみ（ADEN=1 かつ ADDIS=0 である必要があります）、ソフトウェアは ADCx_CR レジスタの制御ビット ADSTART、JADSTART、および ADDIS を書き込むことができます。

ADCx_CFGR、ADCx_SMPRy、ADCx_TRy、ADCx_SQRy、ADCx_JDRy、ADCx_OFRy、ADCx_OFCHRy、および ADCx_IER レジスタのその他の制御ビットについては、次のとおりです。

- レギュラ変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のレギュラ変換がない（ADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。
- インジェクト変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のインジェクト変換がない（JADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。

ADC が有効であり、最終的に変換される場合、および ADC を無効にするための保留中のリクエストがない場合のみ（ADSTART = 1 または JADSTART = 1、かつ ADDIS = 0）、ソフトウェアは ADCx_CR レジスタの ADSTP 制御ビットまたは JADSTP 制御ビットを書き込むことができます。

ソフトウェアは、ADCが有効である場合（ADEN=1）、いつでもレジスタ ADCx_JSQR を書き込むことができます。

注： ***これらの禁止された書き込みアクセスを防ぐハードウェア保護はありません。ADCの挙動は不明な状態になる場合があります。この状況を回復するには、ADC を無効にする必要があります（ADEN=0 と ADCx_CR レジスタのすべてのビットをクリアします）。***

25.3.11 チャネルの選択（SQRx、JSQRx）

ADC ごとに最大 20 の多重化チャネルがあります。

- アナログパッドおよび GPIO パッドからの 6 つの高速アナログ入力（ADCx_INP/INN[0..5]）
- GPIO パッドからの最大 14 個の低速アナログ入力（ADCx_INP/INN[6..19]）
- ADC は、5 つの内部アナログ入力に接続されています。
  - 内部基準電圧（V_{REFINT}）は ADC3_INP/INN19 に接続されています。
  - 内部温度センサ（V_{TS}）は ADC3_INP/INN18 に接続されています。
  - V_{BAT} 監視チャネル（$V_{BAT}/4$）は ADC3_INP/INN17 に接続されています。
  - DAC 内部チャネル 1 は ADC2_INP/INN16 に接続されています。
  - DAC 内部チャネル 2 は ADC2_INP/INN17 に接続されています。

変換は、レギュラとインジェクトの 2 つのグループに構成することができます。各グループは、任意のチャネルに対して任意の順序で行うことができる一連の変換で構成されます。たとえば、次のような順で変換順を設定することができます：ADC_INP/INN3、ADC_INP/INN8、ADC_INP/INN2、ADC_INP/INN2、ADC_INP/INN0、ADC_INP/INN2、ADC_INP/INN2、ADC_INP/INN15

- **レギュラグループ**は、最大 16 の変換で構成されます。レギュラチャネルと変換シーケンス内での順序は、ADCx_SQR レジスタで選択する必要があります。レギュラグループの合計変換数は、ADCx_SQR1 レジスタの L[3:0] ビットに書き込む必要があります。

- **インジェクトグループ**は、最大 4 つの変換で構成されます。インジェクトチャネルと変換シーケンス内での順序は、ADCx_JSQR レジスタで選択する必要があります。インジェクトグループの合計変換数は、ADCx_JSQR レジスタの L[1:0] ビットに書き込む必要があります。

レギュラ変換が実行される可能性がある場合は、ADCx_SQR レジスタを変更しないでください。この場合、最初に ADSTP=1 を書き込んで、ADC レギュラ変換を停止する必要があります（*セクション 25.3.18：実行中の変換の停止（ADSTP、JADSTP）*を参照）。

インジェクト変換の実行中は、動作中に ADCx_JSQR レジスタを変更することができます。*セクション 25.3.22：インジェクト変換のコンテキストのキュー*を参照してください。

温度センサ、V_{REFINT}、および V_{BAT} 内部チャネル

温度センサ V_{TS} は、チャネル ADC3 V_{INP}[18 に接続されます。

V_{BAT} チャネルは、チャネル ADC3 V_{INP}[17] に接続されます。

内部基準電圧 V_{REFINT} は ADC3 V_{INP}[19]に接続されています。

注： ***内部アナログチャネルのうちの 1 つを変換するには、最初に ADCx_CCR レジスタのビット VREFEN、TSEN、または VBATEN をプログラムして、対応するアナログソースを有効化する必要があります。***

25.3.12 プリチャネル選択レジスタ（ADCx_PCSEL）

SQRx または JSQRx で選択された各チャネルについて、対応する ADCx_PCSEL ビットを事前に設定する必要があります。

この ADCx_PCSEL ビットは、IO レベルに組み込まれている送信ゲートを制御します。ADC 入力 MUX は、SQRx および JSQRx に従って ADC 入力を超高速で選択しますが、IOに組み込まれている送信ゲートは、ADC MUX ほどの高速では反応できません。IO での送信ゲート制御による遅延を回避するためには、SQRx と JSQRx で選択される予定の入力チャネルを事前に選択しておくことが必要です。

この選択は、各 ADC 入力の $V_{INP[i]}$ に基づいています。ADC1 が 差動モードで ADC123_INP2（$V_{INP[2]}$）を変換するのであれば、ADC123_INP6（$V_{INP[6]}$）についても ADCx_PCSEL で選択される必要があります。

ADC 入力の中には、ADCx の複数の $V_{INP[i]}$ に接続されているものもあります。これらの入力は、ADCx_PCSEL レジスタのビットによって OR を取られます。

25.3.13 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）

変換を開始する前に、ADC は測定する電圧ソースと ADC の内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースが内蔵コンデンサを入力電圧レベルまでチャージできるだけの十分な長さが必要です。

各チャネルは、ADCx_SMPR1 および ADCx_SMPR2 レジスタの SMP[2:0] ビットをプログラムすることでな異なるサンプリング時間でサンプリングできます。したがって、次のサンプリング時間の値から選択することができます。

- SMP = 000：1.5 ADC クロックサイクル
- SMP = 001：2.5 ADC クロックサイクル
- SMP = 010：8.5 ADC クロックサイクル
- SMP = 011：16.5 ADC クロックサイクル
- SMP = 100：32.5 ADC クロックサイクル
- SMP = 101：64.5 ADC クロックサイクル
- SMP = 110：387.5 ADC クロックサイクル
- SMP = 111：810.5 ADC クロックサイクル

合計変換時間は、次のように計算されます。

T_{CONV} = サンプリング時間 + 7.5 ADC クロックサイクル

例：

F_{ADC_CLK} = 24 MHz、サンプリング時間 = 1.5 ADC クロックサイクル：

T_{CONV} =（1.5 + 7.5）ADC クロックサイクル = 9 ADC クロックサイクル = 0.375 µs（高速チャネルの 14 ビットモード）

ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。

高速および低速チャネルのサンプリング時間の制約

各チャネルで、SMP[2:0] ビットはデータシートの ADC 特性セクションで指定された最小サンプリング時間を考慮してプログラムする必要があります。

I/O アナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/O アナログスイッチ抵抗が増加します。これには、状況に応じて適用されるサンプリング時間が必要になります（電気的特性に関するデータシート参照）。SYSCFG_CFGR1 レジスタの BOOSTEN ビットで内部電圧ブースタを有効にすることによって、この抵抗を低い V_{DDA} で最小化できます。

25.3.14 シングル変換モード（CONT=0）

シングル変換モードでは、ADC はすべてのチャネルを一度変換します。CONT ビットが 0 の時、次のいずれかによってこのモードが開始します。

- ADCx_CR レジスタの ADSTART ビットをセットします（レギュラチャネル、ソフトウェアトリガを選択）。
- ADCx_CR レジスタの JADSTART ビットをセットします（インジェクトチャネル、ソフトウェアトリガを選択）。
- 外部ハードウェアトリガイベント（レギュラまたはインジェクトチャネル）

  外部イベントをトリガする前に ADSTART ビットまたは JADSTART ビットをセットする必要があります。

レギュラシーケンス内で、各変換の完了後、

- 変換データは 32 ビットの ADCx_DR レジスタに格納されます。
- EOC （レギュラ変換の完了）フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

インジェクトシーケンス内で、各変換の完了後、

- 変換データは 4 つの 32 ビット ADCx_JDRy レジスタのうちの 1 つに格納されます。
- JEOC （インジェクト変換の完了）フラグがセットされます。
- JEOCIE ビットがセットされている場合、割り込みが生成されます。

レギュラシーケンスの完了後、

- EOS （レギュラシーケンスの完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割り込みが生成されます。

インジェクトシーケンスの完了後、

- JEOS （インジェクトシーケンスの完了）フラグがセットされます。
- JEOSIE ビットがセットされている場合、割り込みが生成されます。

次に、ADC は、新しい外部レギュラトリガまたは外部インジェクトトリガが発生するか、ADSTART または JADSTART ビットが再びセットされるまで停止します。

注： ***単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。***

25.3.15 連続変換モード（CONT=1）

このモードはレギュラチャネルにのみ適用されます。

連続変換モードでは、ソフトウェアまたはハードウェアのレギュラトリガイベントが発生すると、ADC はチャネルのレギュラ変換を一度すべて実行し、自動的に再起動して、シーケンスの各変換を連続的に実行します。CONT ビットが 1 の時、外部トリガまたは ADCx_CR レジスタの ADSTART ビットをセットすることによって、このモードが開始されます。

レギュラシーケンス内で、各変換の完了後、

- 変換データは 32 ビットの ADCx_DR レジスタに格納されます。
- EOC（変換完了）フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割り込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADC は変換シーケンスを連続的に繰り返します。

注： ***単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。***

不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。

インジェクトチャネルを連続して変換することはできません。唯一の例外は、連続モードに設定されたレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTO ビットを使用）場合です。自動インジェクションモードのセクションを参照してください。

25.3.16 変換の開始（ADSTART、JADSTART）

ソフトウェアは、ADSTART=1 をセットすることによって ADC レギュラ変換を開始します。

ADSTART がセットされると、変換は、

- EXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- EXTEN ≠ 0x0 の場合、選択されたレギュラハードウェアトリガの次のアクティブエッジで開始します。

ソフトウェアは、JADSTART = 1 をセットすることによって ADC インジェクト変換を開始します。

JADSTART がセットされると、変換は、

- JEXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- JEXTEN ≠ 0x0 の場合、選択されたインジェクトハードウェアトリガの次のアクティブエッジで開始します。

注： ***自動インジェクションモード（JAUTO=1）では、ADSTART ビットを使用してレギュラ変換を開始し、続いて自動インジェクト変換を行います（JADSTART はクリアされたままでなければなりません）。***

ADSTART および JADSTART では、ADC の動作が実行中かどうかを示す情報も提供しています。ADSTART=0 および JADSTART=0 が両方とも真であり、ADC がアイドルであることを示しているときには、ADC を再設定できます。

次のときに、ADSTART はハードウェアによってクリアされます。

- ソフトウェアトリガによるシングルモードのとき（CONT=0、EXTEN=0x0）
  - 変換シーケンスの終了時に（EOS=1）
- ソフトウェアトリガによる不連続モードのとき（CONT=0、DISCEN=1、EXTEN=0x0）
  - 変換の終了時に（EOC=1）
- それ以外のすべての場合に（CONT=x、EXTEN=x）
  - ソフトウェアによってアサートされた ADSTP 手順の実行後

注： ***連続モード（CONT=1）では、シーケンスは自動的に再起動されるので、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。***

シングルモードでハードウェアトリガが選択されたとき（CONT=0 かつ EXTEN ≠ 0x00）、次のハードウェアトリガイベントで ADSTART の再リセットを必要としないソフトウェアを支援するために、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。これにより、さらなるハードウェアトリガの見落としを防ぐことができます。

次のときに、JADSTART はハードウェアによってクリアされます。

- シングルモード時のソフトウェアのインジェクトトリガ（JEXTEN=0x0）
  - インジェクト変換シーケンスの終了時に（JEOS のアサート）、または JDISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（JEXTEN=x）
  - ソフトウェアによってアサートされた JADSTP 手順の実行後

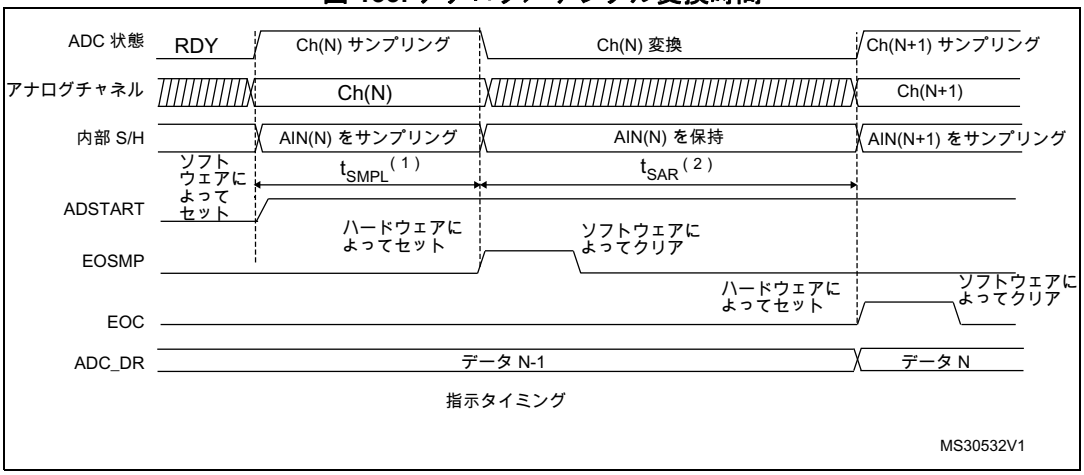
25.3.17 タイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計です。

$T_{CONV} = T_{SMPL} + T_{SAR} = [1.5\ _{|min} + 7.5\ _{|14bit}] \times T_{adc_ker_ck}$

$T_{CONV} = T_{SMPL} + T_{SAR} = 62.5\ ns\ _{|min} + 312.5\ ns\ _{|14bit} = 375.0\ ns$（$F_{adc_ker_ck}$ = 24 MHz の場合）

図 133. アナログ／デジタル変換時間



1. T_{SMPL} は SMP[2:0] に依存
2. T_{SAR} は RES[2:0] に依存

25.3.18 実行中の変換の停止（ADSTP、JADSTP）

ソフトウェアは、ADSTP=1 をセットすることによって実行中のレギュラ変換を、また JADSTP=1 をセットすることによって実行中のインジェクト変換を停止することができます。

変換を停止すると、実行中の ADC 動作がリセットされます。これによって ADC 動作が再設定され（チャネル選択やトリガの変更など）、新しい動作の準備ができます。

レギュラ変換の実行中に、インジェクト変換を停止することもできる点に注意してください（その逆も可能）。これにより、たとえば、レギュラ変換の実行中にインジェクト変換シーケンスやトリガを再設定することができます（その逆も可能）。

ADSTP ビットがソフトウェアによってセットされると、実行中のレギュラ変換はアボートされ、部分的な結果は破棄されます（ADCx_DR レジスタは現在の変換では更新されません）。

JADSTP ビットがソフトウェアによってセットされると、実行中のインジェクト変換はアボートされ、部分的な結果は破棄されます（ADCx_JDRy レジスタは現在の変換では更新されません）。スキャンシーケンスも中止され、リセットされます（ADC を再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTP/ADSTART ビット（レギュラ変換の場合）、または JADSTP/JADSTART ビット（インジェクト変換の場合）がハードウェアによってクリアされ、ソフトウェアは ADC が完全に停止する前にビットがリセットされるまで ADSTART（または JADSTART）をポーリングする必要があります。

注： ***自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP を使用してはいけません）。***

図 134. 実行中のレギュラ変換の停止

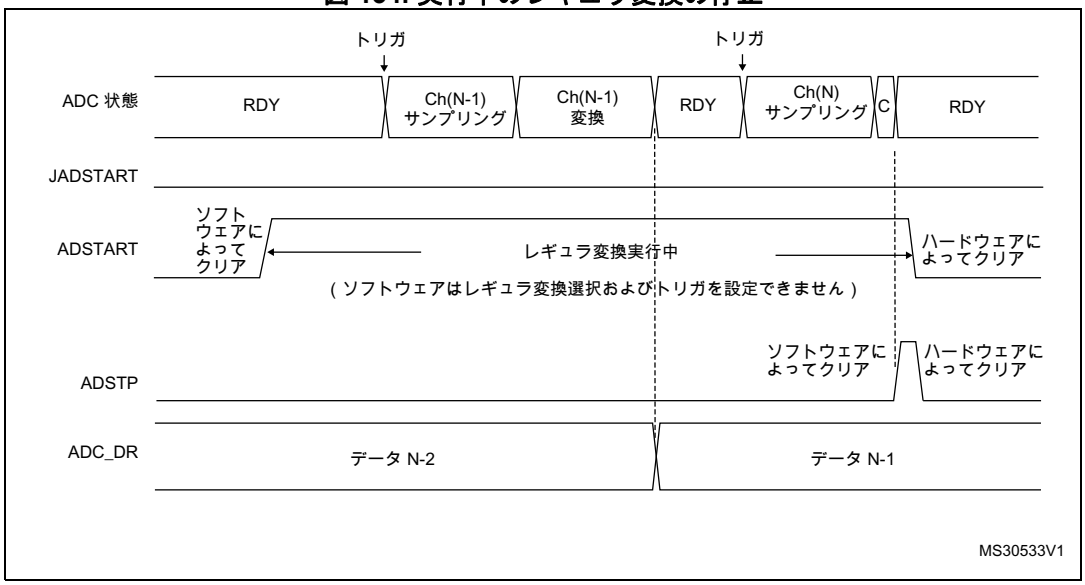
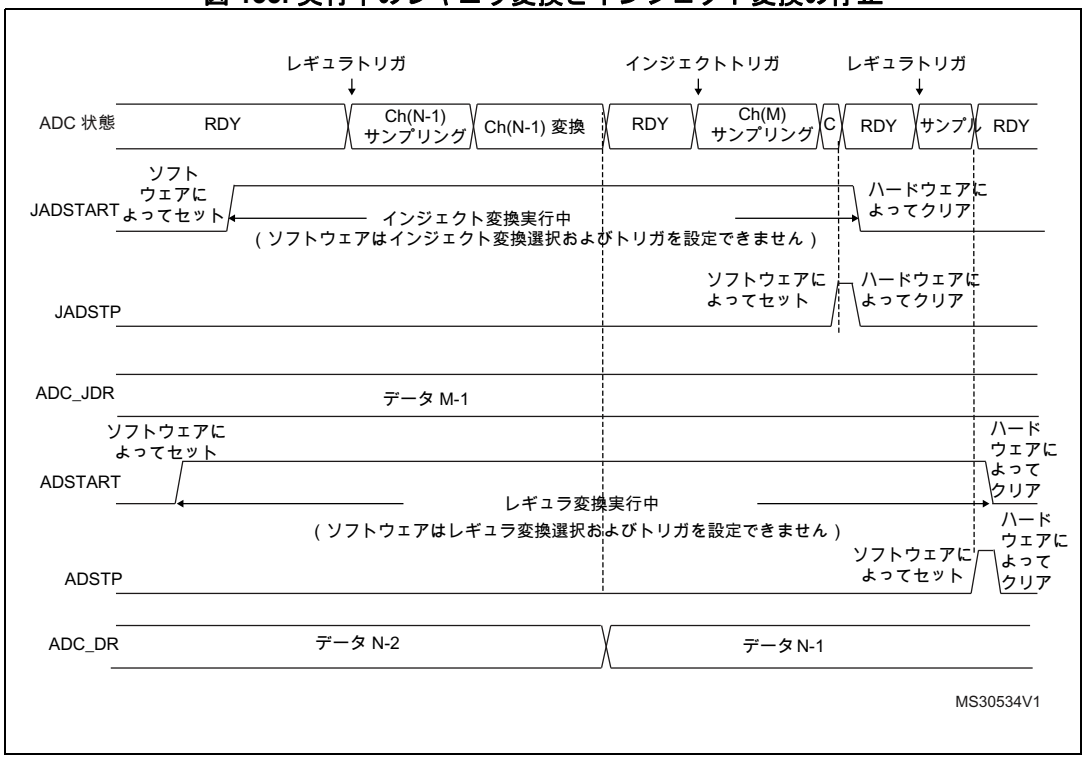


図 135. 実行中のレギュラ変換とインジェクト変換の停止



25.3.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャや入力ピンなど）によってトリガできます。EXTEN[1:0]制御ビット（レギュラ変換の場合）またはJEXTEN[1:0] ビット（インジェクト変換の場合）が 0b00 以外の場合、外部イベントは選択した極性で変換をトリガすることができます。

インジェクトキューが有効である場合（ビット JQDIS=0）、インジェクトされたソフトウェアのトリガはできません。

レギュラトリガ選択は、一度ソフトウェアでビット ADSTART=1 をセットすると有効になります。インジェクトトリガ選択は、一度ソフトウェアでビット JADSTART=1 をセットすると有効になります。

変換中に発生したハードウェアトリガは無視されます。

- ビット ADSTART=0 の場合、発生したレギュラハードウェアトリガは無視されます。
- ビット JADSTART=0 の場合、発生したインジェクトハードウェアトリガは無視されます。

表 191 に EXTEN[1:0] と JEXTEN[1:0] の値とトリガ極性の対応を示します。

表 191. レギュラ外部トリガのトリガ極性の設定

EXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***レギュラトリガの極性は動作中に変更することはできません。***

表 192. インジェクト外部トリガのトリガ極性の設定

JEXTEN[1:0]	転送元
00	– JQDIS=1（キュー無効）の場合：ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。 – JQDIS=0（キュー有効）の場合：ハードウェアおよびソフトウェアトリガ検出は無効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***キューが有効（JQDIS=0）になっている場合、インジェクトトリガの極性は動作中に予想して変更できます。セクション 25.3.22：インジェクト変換のコンテキストのキューを参照してください。***

EXTSEL[4:0] および JEXTSEL[4:0] 制御ビットでは、レギュラおよびインジェクトグループの変換をトリガできるイベントを 21 のイベントから選択します。

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： ***レギュラトリガの選択は動作中に変更することはできません。***
インジェクトトリガの選択は動作中に予想して変更できます。セクション 25.3.22： 878 ページのインジェクト変換のコンテキストのキュー を参照してください。

参考資料

図 136 に示すように、各 ADC マスタは ADC スレーブと同じ入力トリガを共有しています。

図 136. ADC マスタと ADC スレーブ間でトリガを共有

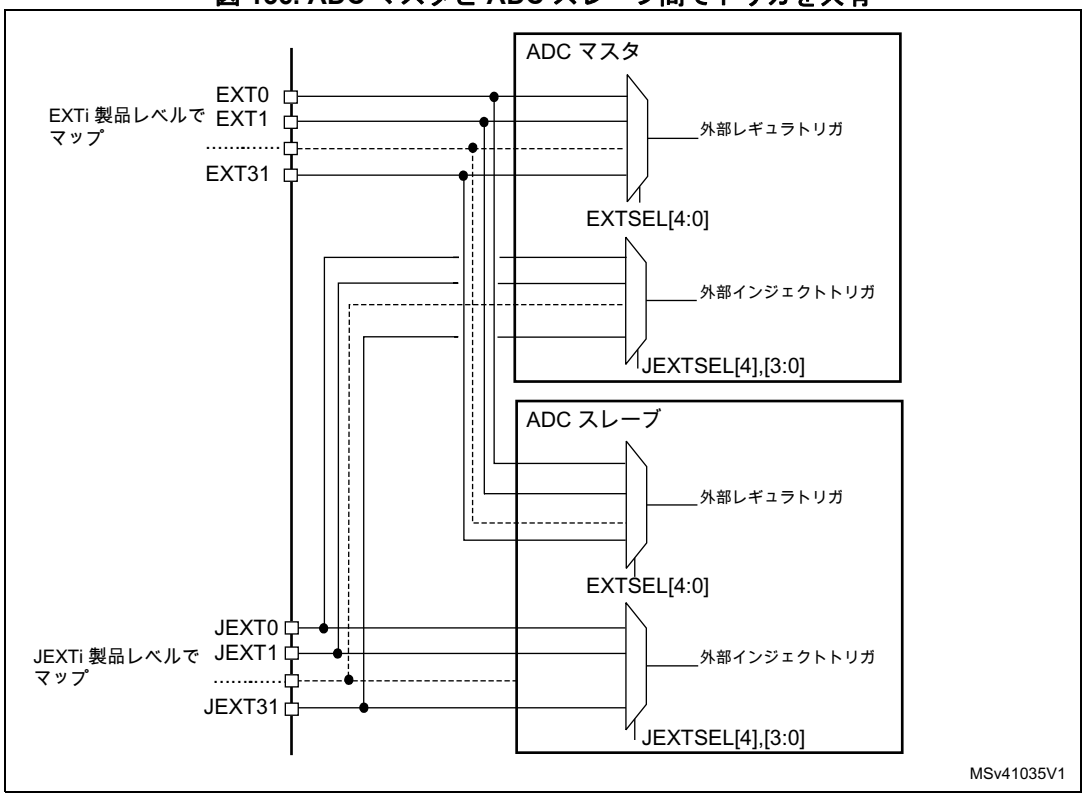


表 193 および*表 194* では、レギュラ変換とインジェクト変換に使用できる 3 つの ADC の全ての外部トリガを示します。

表 193. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg0	TIM1_CC1 イベント	オンチップタイマからの内部信号	00000
adc_ext_trg1	TIM1_CC2 イベント	オンチップタイマからの内部信号	00001
adc_ext_trg2	TIM1_CC3 イベント	オンチップタイマからの内部信号	00010
adc_ext_trg3	TIM2_CC2 イベント	オンチップタイマからの内部信号	00011
adc_ext_trg4	TIM3_TRGO イベント	オンチップタイマからの内部信号	00100
adc_ext_trg5	TIM4_CC4 イベント	オンチップタイマからの内部信号	00101
adc_ext_trg6	EXTI ライン 11	外部ピン	00110
adc_ext_trg7	TIM8_TRGO イベント	オンチップタイマからの内部信号	00111
adc_ext_trg8	TIM8_TRGO2 イベント	オンチップタイマからの内部信号	01000
adc_ext_trg9	TIM1_TRGO イベント	オンチップタイマからの内部信号	01001
adc_ext_trg10	TIM1_TRGO2 イベント	オンチップタイマからの内部信号	01010
adc_ext_trg11	TIM2_TRGO イベント	オンチップタイマからの内部信号	01011
adc_ext_trg12	TIM4_TRGO イベント	オンチップタイマからの内部信号	01100
adc_ext_trg13	TIM6_TRGO イベント	オンチップタイマからの内部信号	01101

表 193. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ（続き）

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg14	TIM15_TRGO イベント	オンチップタイマからの内部信号	01110
adc_ext_trg15	TIM3_CC4 イベント	オンチップタイマからの内部信号	01111
adc_ext_trg16	HRTIM1_ADCTRG1 イベント	オンチップタイマからの内部信号	10000
adc_ext_trg17	HRTIM1_ADCTRG3 イベント	オンチップタイマからの内部信号	10001
adc_ext_trg18	LPTIM1_OUT イベント	オンチップタイマからの内部信号	10010
adc_ext_trg19	LPTIM2_OUT イベント	オンチップタイマからの内部信号	10011
adc_ext_trg20	LPTIM3_OUT イベント	オンチップタイマからの内部信号	10100
adc_ext_trg21	予約済み	-	10101
adc_ext_trg22	予約済み	-	10110
adc_ext_trg23	予約済み	-	10111
adc_ext_trg24	予約済み	-	11000
adc_ext_trg25	予約済み	-	11001
adc_ext_trg26	予約済み	-	11010
adc_ext_trg27	予約済み	-	11011
adc_ext_trg28	予約済み	-	11100
adc_ext_trg29	予約済み	-	11101
adc_ext_trg30	予約済み	-	11110
adc_ext_trg31	予約済み	-	11111

表 194. ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg0	TIM1_TRGO イベント	オンチップタイマからの内部信号	00000
adc_ext_trg1	TIM1_CC4 イベント	オンチップタイマからの内部信号	00001
adc_jext_trg2	TIM2_TRGO イベント	オンチップタイマからの内部信号	00010
adc_jext_trg3	TIM2_CC1 イベント	オンチップタイマからの内部信号	00011
adc_jext_trg4	TIM3_CC4 イベント	オンチップタイマからの内部信号	00100
adc_jext_trg5	TIM4_TRGO イベント	オンチップタイマからの内部信号	00101
adc_jext_trg6	EXTI ライン 15	外部ピン	00110
adc_jext_trg7	TIM8_CC4 イベント	オンチップタイマからの内部信号	00111
adc_jext_trg8	TIM1_TRGO2 イベント	オンチップタイマからの内部信号	01000
adc_jext_trg9	TIM8_TRGO イベント	オンチップタイマからの内部信号	01001
adc_jext_trg10	TIM8_TRGO2 イベント	オンチップタイマからの内部信号	01010
adc_jext_trg11	TIM3_CC3 イベント	オンチップタイマからの内部信号	01011
adc_jext_trg12	TIM3_TRGO イベント	オンチップタイマからの内部信号	01100
adc_jext_trg13	TIM3_CC1 イベント	オンチップタイマからの内部信号	01101
adc_jext_trg14	TIM6_TRGO イベント	オンチップタイマからの内部信号	01110
adc_jext_trg15	TIM15_TRGO イベント	オンチップタイマからの内部信号	01111

表 194. ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ（続き）

名前	転送元	タイプ	EXTSEL[4:0]
adc_jext_trg16	HRTIM1_ADCTRG2 イベント	オンチップタイマからの内部信号	10000
adc_jext_trg17	HRTIM1_ADCTRG4 イベント	オンチップタイマからの内部信号	10001
adc_jext_trg18	LPTIM1_OUT イベント	オンチップタイマからの内部信号	10010
adc_jext_trg19	LPTIM2_OUT イベント	オンチップタイマからの内部信号	10011
adc_jext_trg20	LPTIM3_OUT イベント	オンチップタイマからの内部信号	10100
adc_jext_trg21	予約済み	-	10101
adc_jext_trg22	予約済み	-	10110
adc_jext_trg23	予約済み	-	10111
adc_jext_trg24	予約済み	-	11000
adc_jext_trg25	予約済み	-	11001
adc_jext_trg26	予約済み	-	11010
adc_jext_trg27	予約済み	-	11011
adc_jext_trg28	予約済み	-	11100
adc_jext_trg29	予約済み	-	11101
adc_jext_trg30	予約済み	-	11110
adc_jext_trg31	予約済み	-	11111

25.3.20 インジェクトチャネルの管理

トリガインジェクションモード

トリガインジェクションを使用するには、ADCx_CFGR レジスタの JAUTO ビットがクリアされていなければなりません。

1. 外部トリガによって、または、ADCx_CR レジスタの ADSTART ビットをセットすることによって、レギュラチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合、または ADCx_CR レジスタの JADSTART ビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンススイッチが起動されます（すべてのインジェクトチャネルが一度変換されます）。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。
4. インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。*図 137* に対応するタイミング図を示します。

注： ***トリガインジェクションを使用する場合は、トリガイベントの間隔がインジェクションシーケンスより長くなるようにしなければなりません。たとえば、シーケンスの長さが 20 ADC クロックサイクルの場合（1.5 クロック周期のサンプリング時間を持つ 2 つの変換）、トリガの最小間隔は 21 ADC クロックサイクルでなければなりません。***

自動インジェクションモード

ADCx_CFGR レジスタの JAUTO ビットがセットされている場合、レギュラグループチャネルの変換後、インジェクトグループチャネルが自動的に変換されます。これを使用して、ADCx_SQR レジスタおよび ADCx_JSQR レジスタにプログラミングされた最大 20 回の変換シーケンスを実行することができます。

このモードでは、レギュラ変換を開始し、続いてインジェクト変換を開始するために ADCx_CR レジスタの ADSTART ビットをセットする必要があります（JADSTART はクリアされたままでなければなりません）。ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP ビットを使用することはできません）。

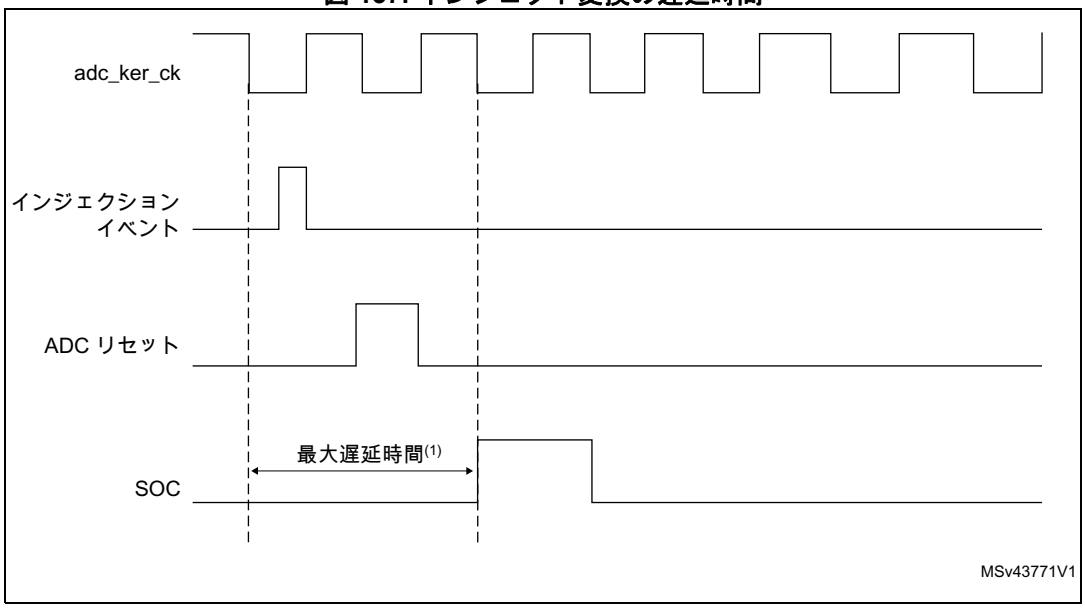
このモードでは、インジェクトチャネルの外部トリガは無効でなければなりません。

JAUTO ビットに加えて CONT ビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： ***自動インジェクトモードと不連続モードを同時に使用することはできません。***

JAUTO モードで、レギュラシーケンスのデータをエクスポートする際に DMA を使用する場合は、サーキュラモードでプログラムする必要があります（DMA_CCRx レジスタの CIRC ビットをセット）。CIRC ビットがリセットされると（シングルショットモード）、DMA 転送完了イベントによって JAUTO シーケンスが停止されます。

図 137. インジェクト変換の遅延時間



1. 最大遅延時間の値は、デバイスのデータシートの電気特性に記載されています。

25.3.21 不連続モード（DISCEN、DISCNUM、JDISCEN）

レギュラグループモード

このモードは、ADCx_CFGR レジスタの DISCEN ビットをセットすることによって有効になります。

これを使用して、ADCx_SQRy レジスタで選択された変換シーケンスの一部である短い n 回（n ≤ 8）の変換シーケンス（サブグループ）を変換できます。n の値は、ADCx_CFGR レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADCx_SQR レジスタで選択された次の n 回の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADCx_SQR1 レジスタの L[3:0] ビットによって定義されます。

例：

- DISCEN=1、n=3、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - 2 番目のトリガ：変換されるチャネルは 6、7、8 です（各変換で EOC イベントが生成されます）。
  - 3 番目のトリガ：変換されるチャネルは 9、10、11 です（各変換で EOC イベントが生成されます）。EOS イベントはチャネル 11 の変換後に生成されます。
  - 4 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - ...
- DISCEN=0、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：シーケンス全体、すなわち、チャネル 1、次いで 2、3、6、7、8、9、10、および 11 が変換されます。各変換後に EOC イベントが生成され、最後の変換後には EOS イベントも生成されます。
  - 後続のすべてのトリガイベントによって、シーケンス全体が再起動されます。

注： ***レギュラグループが不連続モードで変換されるときには、ロールオーバーは発生しません（シーケンスの最後のサブグループの変換は n 回以下です）。***

すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。上の例では、4 番目のトリガで最初のサブグループのチャネル 1、2、および 3 が再変換されます。

不連続モードと連続モードの両方を有効にすることはできません。この場合（DISCEN=1、CONT=1 の場合）、ADC は連続モードが無効化されたかのように挙動します。

インジェクトグループモード

このモードは、ADCx_CFGR レジスタの JDISCEN ビットをセットすることによって有効になります。ADCx_JSQR レジスタで選択されたシーケンスを、外部インジェクトトリガイベント後に 1 チャネルずつ変換します。これは、「n」が 1 に固定されたレギュラチャネルの不連続モードに相当します。

外部トリガが発生すると、ADCx_JSQR レジスタで選択された次のチャネル変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADCx_JSQR レジスタの JL[1:0]ビットによって定義されます。

例：

- JDISCEN=1、変換されるチャネル = 1、2、3
  - 1 番目のトリガ：チャネル 1 が変換されます（JEOC イベントが生成されます）。
  - 2 番目のトリガ：チャネル 2 が変換されます（JEOC イベントが生成されます）。
  - 3 番目のトリガ：チャネル 3 が変換され、JEOC イベントと JEOS イベントが生成されます。
  - ...

注： ***すべてのインジェクトチャネルが変換されると、次のトリガでは、最初のインジェクトチャネルの変換が開始されます。上の例では、4 番目のトリガで最初のインジェクトチャネル 1 が再変換されます。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO がセットされている場合には、DISCEN ビット と JDISCEN ビット はクリアされている必要があります。

25.3.22 インジェクト変換のコンテキストのキュー

コンテキストのキューは、次のインジェクト変換シーケンスに最大 2 つのコンテキストを予測して実装されます。この機能を有効にするには、ADCx_CFGR レジスタの JQDIS ビットをリセットする必要があります。コンテキストキューが有効になっている場合、ハードウェアトリガによる変換のみ可能です。

このコンテキストの構成は次のとおりです。

- インジェクトトリガの設定（ADCx_JSQR レジスタのJEXTEN[1:0]ビット および JEXTSEL[4:0]ビット）
- インジェクトシーケンスの定義（ADCx_JSQR レジスタのJSQx[4:0]ビット および JL[1:0]ビット）

コンテキストのすべてのパラメータは 1 つのADCx_JSQRレジスタ 内に定義され、このレジスタには最大 2 セットのパラメータのバッファ化を可能にする 2 つのバッファのキューが実装されます。

- JSQR レジスタは、インジェクト変換の実行中など、いつでも書き込むことができます。
- JSQR レジスタに書き込まれる各データは、コンテキストのキューに格納されます。
- 最初に、キューは空であり、JSQR レジスタへの最初の書き込みアクセスによってコンテキストがすぐに変更され、ADC ではインジェクトトリガを受信する準備ができます。
- 一度インジェクトシーケンスが完了すると、キューは消費され、キューに格納された次の JSQR パラメータに従ってコンテキストが変化します。この新しいコンテキストは、次のインジェクト変換シーケンスに適用されます。
- キューのオーバーフローは、キューがフル状態のときにJSQR レジスタに書き込むと発生します。このオーバーフローは、フラグ JQOVF のアサーションによって示されます。オーバーフローが発生すると、オーバーフローを作成した JSQR レジスタの書き込みアクセスは無視され、コンテキストのキューは変更されません。JQOVFIE ビットがセットされている場合は、割り込みを生成することができます。
- キューが空になると、ADCx_CFGR レジスタの制御ビット JQM の値に応じて、2 つの挙動が考えられます。
  - JQM=0 の場合、キューは ADC を有効化した直後に空になりますが、実行動作中に空になることはありません。キューは常に最後のアクティブコンテキストを維持し、さらに有効なインジェクトシーケンスの開始を最後のアクティブコンテキストに応じて処理します。
  - JQM=1 の場合、キューはインジェクトシーケンス終了後、またはキューのフラッシュ時に空になります。このとき、キュー内にコンテキストはなく、ハードウェアトリガは無効になります。そのため、ソフトウェアが新しいインジェクトコンテキストを JSQR レジスタに再度書き込むまで、さらなるハードウェアインジェクトトリガは無視されます。

- JSQR レジスタを読み出すと、その時点でアクティブな現在の JSQR コンテキストが返されます。JSQR コンテキストが空である場合、JSQR は 0x0000 として読み出されます。
- キューは、JADSTP=1 をセットしてインジェクト変換を停止した場合、または ADDIS=1 をセットして ADC を無効化した場合にフラッシュされます。
  - JQM=0 の場合、キューは最後のアクティブコンテキストによって維持されます。
  - JQM=1 の場合、キューは空になり、トリガは無視されます。

注： ***不連続モードで設定された場合（ビット JDISCEN=1）、コンテキストを変更し、キューを消費するのはインジェクトシーケンスの最後のトリガのみです。1 番目のトリガではキューを消費するだけですが、次の不連続モードの例に示すように、その他のトリガは引き続き有効なトリガとなります（両方のコンテキストにおいて長さは 3）。***

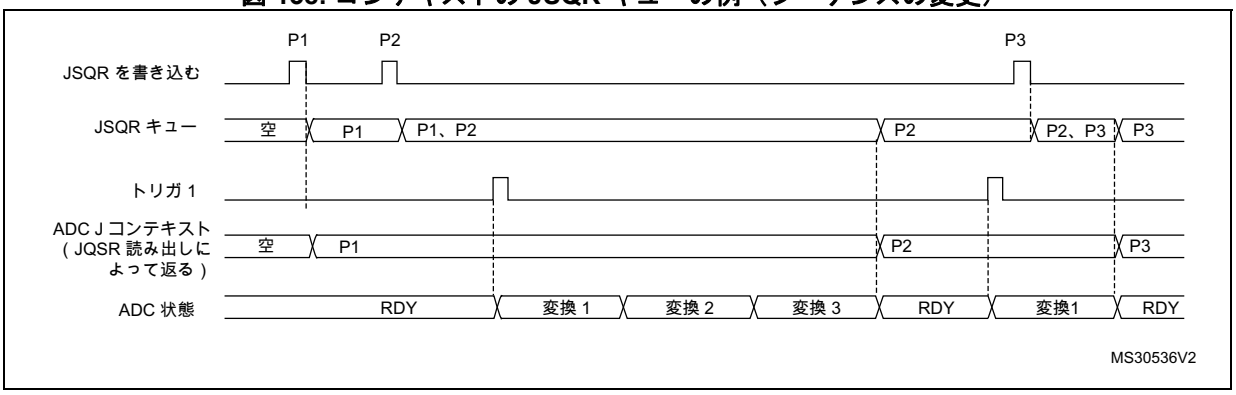
- *1 番目のトリガ、不連続。シーケンス 1：コンテキスト 1 を消費、1 番目の変換を実行*
- *2 番目のトリガ、不連続。シーケンス 1：2 番目の変換*
- *3 番目のトリガ、不連続。シーケンス 1：3 番目の変換*
- *4 番目のトリガ、不連続。シーケンス 2：コンテキスト 2 を消費、1 番目の変換を実行*
- *5 番目のトリガ、不連続。シーケンス 2：2 番目の変換*
- *6 番目のトリガ、不連続。シーケンス 2：3 番目の変換*

注： ***コンテクストのキューが有効である場合（ビット JQDIS=0）、ハードウェアトリガのみ使用できます。***

トリガまたはシーケンスのコンテキストを変更する場合の挙動

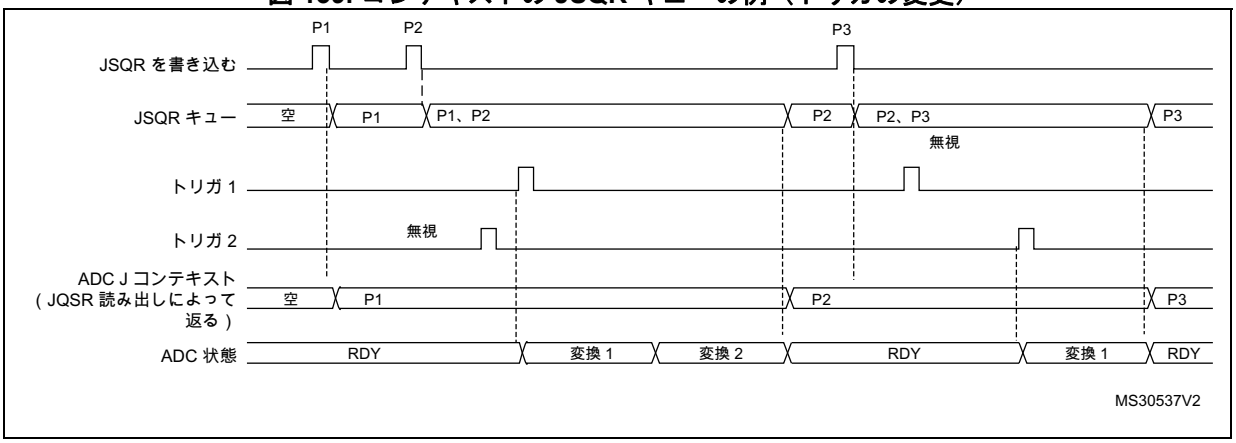
図 138 と *図 139* に、シーケンスまたはトリガを変更する際のコンテキストのキューの挙動を示します。

図 138. コンテキストの JSQR キューの例（シーケンスの変更）



1. パラメータ：
   P1：3 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

図 139. コンテキストの JSQR キューの例（トリガの変更）

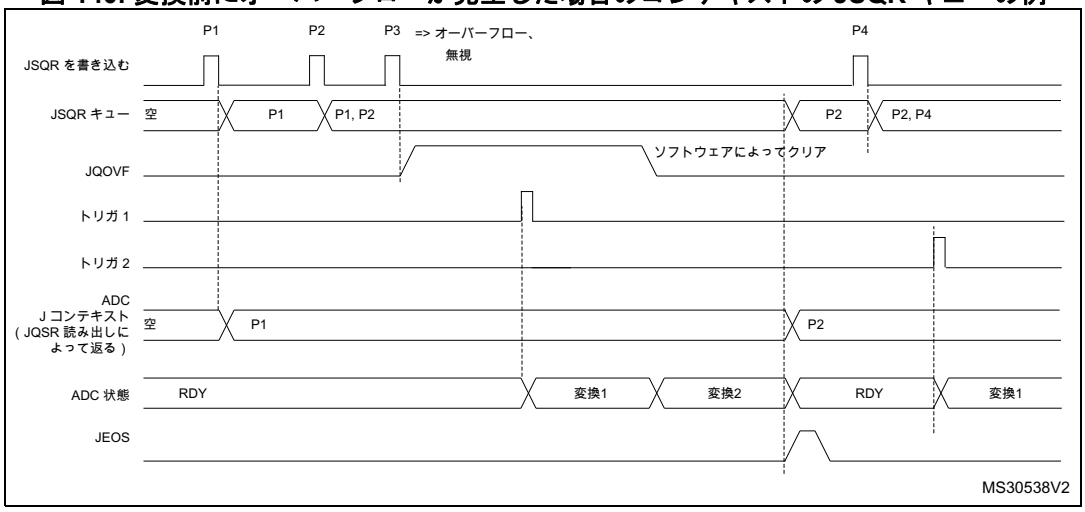


1. パラメータ：
P1：2 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 2
P3：4 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：キューのオーバーフローが発生した場合の挙動

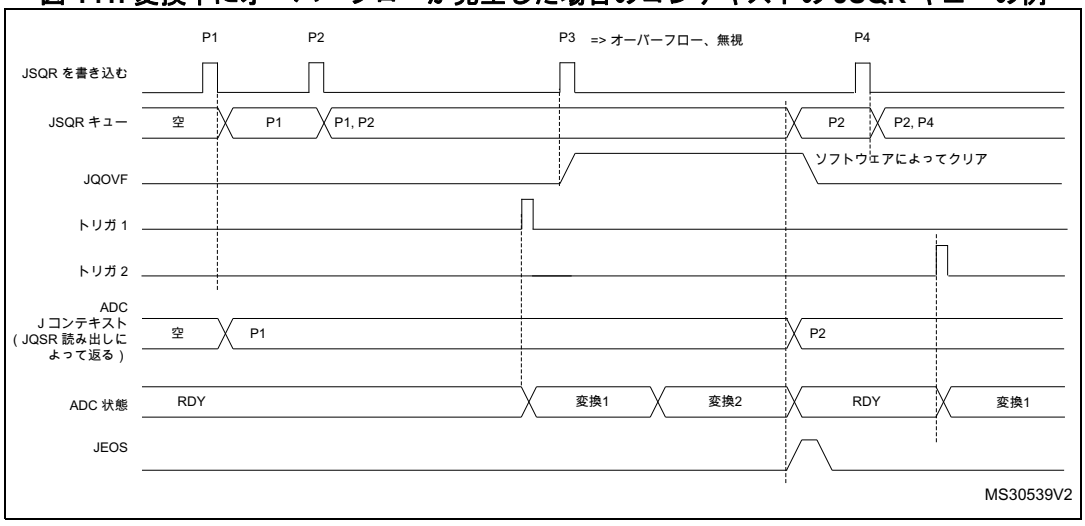
図 140 と *図 141* に、変換前または変換中にオーバーフローが発生した場合のコンテキストのキューの挙動を示します。

図 140. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
P1：2 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 2
P3：3 つの変換のシーケンス、ハードウェアトリガ 1
P4：4 つの変換のシーケンス、ハードウェアトリガ 1

図 141. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

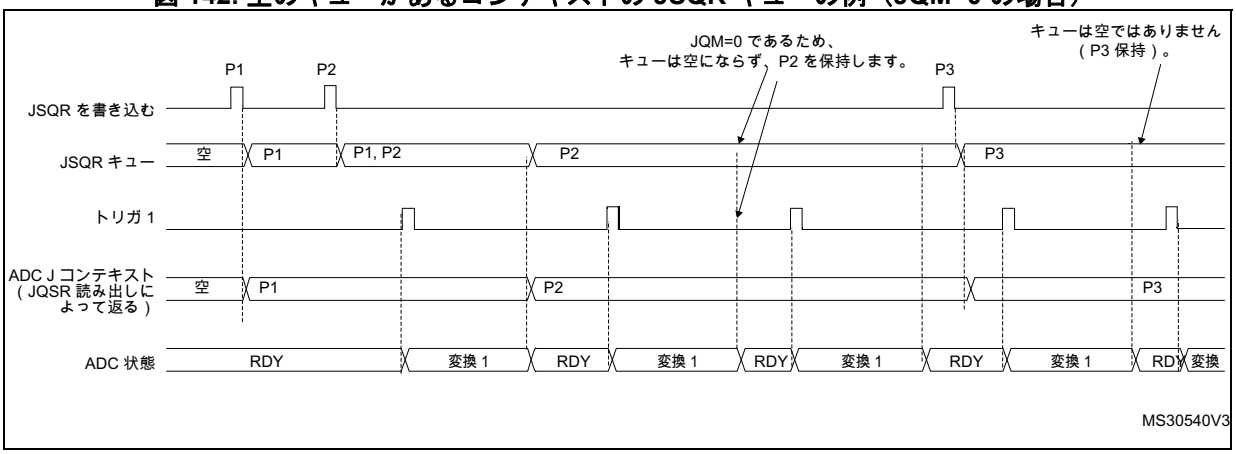
次のようにキューのオーバーフローを管理することを推奨します。

- JSQR レジスタへの各 P コンテキストの書き込み後、JQOVFフラグ は書き込みが無視されたかどうかを示します（割り込みを生成することができます）。
- 一度前のコンテキスト P2 のフラグ JEOS がセットされた場合のみ、3 番目のコンテキスト（P3）を書き込んでキューのオーバーフローを避けます。これにより、前のコンテキストが消費され、キューがフル状態ではないことを確認できます。

コンテキストのキュー：キューが空になった場合の挙動

図 142 と *図 143* に、JQM=0 または 1 の場合にキューが空になる際のコンテキストのキューの挙動を示します。

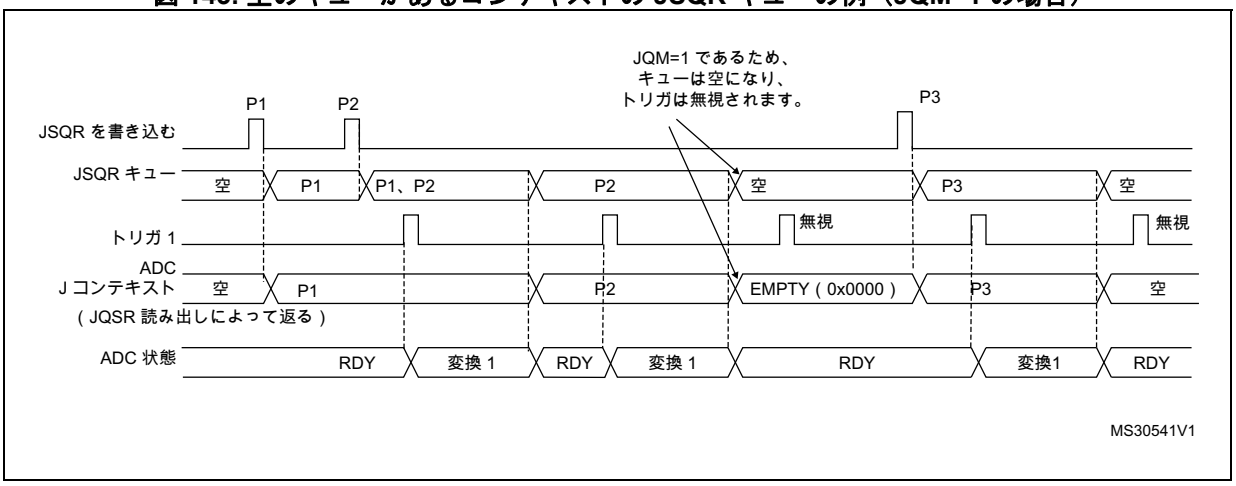
図 142. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

注： ***P3 を書き込む際、コンテキストはすぐに変更されます。ただし、内部での再同期のため、遅延が発生します。また、P3 を書き込んだ後または前にトリガが発生すると、コンテキスト P2 を考慮して変換が発生する可能性があります。この状況を避けるために、ユーザはすぐに適用される新しいコンテキストを書き込む際に ADC トリガが発生していないことを確認する必要があります。***

図 143. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合）

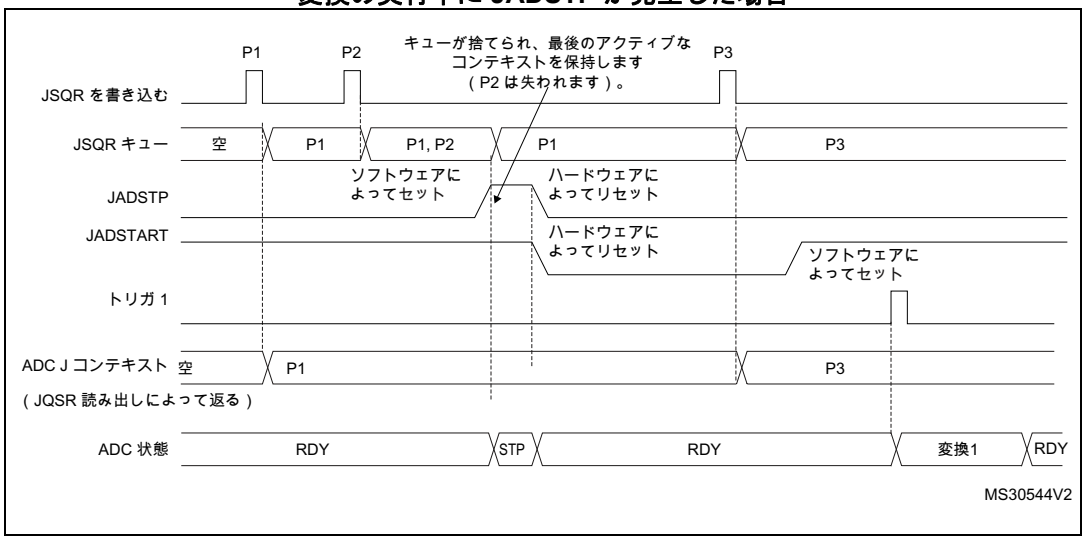


1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキューのフラッシュ

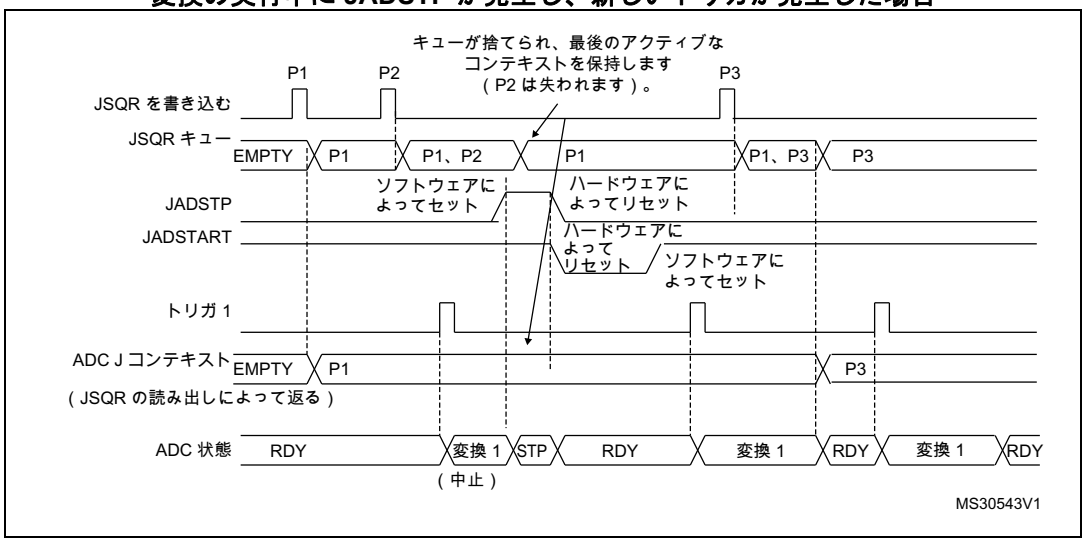
次の図に、キューがフラッシュされた場合のさまざまな状況におけるコンテキストのキューの挙動を示します。

図 144. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に JADSTP が発生した場合



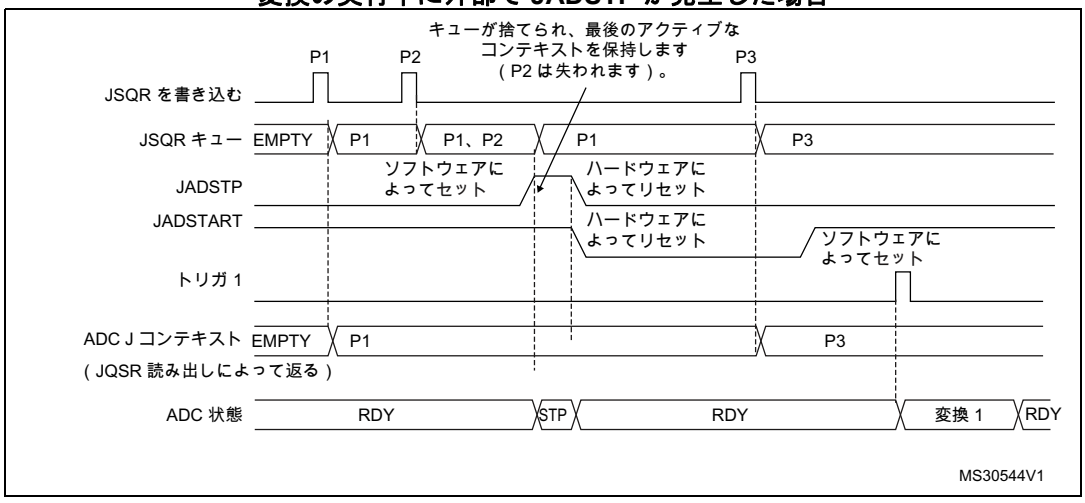
1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 145. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に JADSTP が発生し、新しいトリガが発生した場合



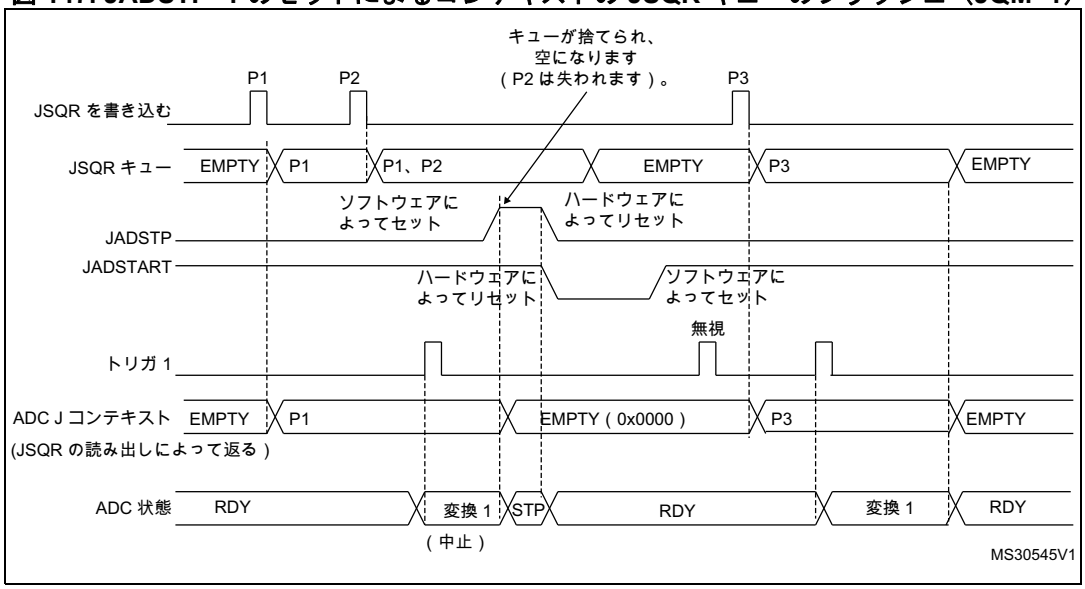
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 146. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に外部で JADSTP が発生した場合



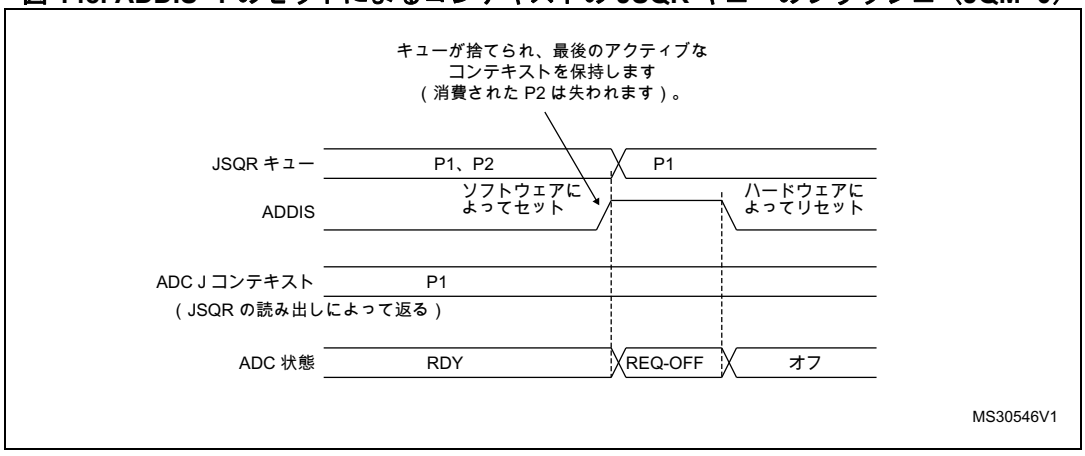
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 147. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1）



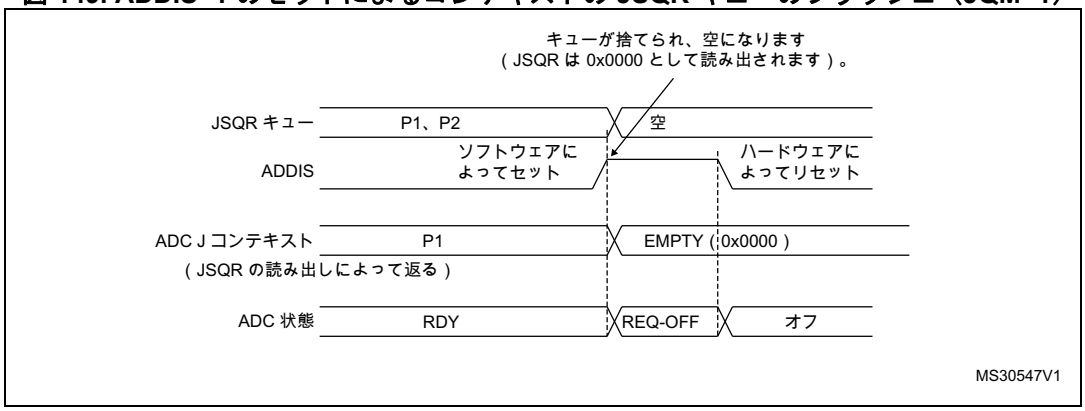
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 148. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 149. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：空のキューでの ADC 開始

最初のコンテキストが ADC 初期化の時点で不明である場合、空のキューで ADC 操作を開始するには次の手順に従う必要があります。この手順は、JQM ビットがリセットされているときのみ適用可能です。

5. 0 ではない JEXTEN でダミーの JSQR を書き込みます（ソフトウェア変換のトリガを除く）。
6. JADSTART をセットします。
7. JADSTP をセットします。
8. JADSTART がリセットされるまで待ちます。
9. JADSTART をセットします。

キューの無効化

ADCx_CFGR レジスタでビット JQDIS=1 にセットすることによって、キューを無効化できます。

コンテキストのキュー：レジスタ ADCx_JSQR のプログラミング

コンテキストのインジェクト変換キューが有効（JQDIS=0）であるときには、ADCx_JSQR は 1 回のレジスタ書込みアクセスでプログラムされる必要があります。JL[1:0] レジスタによってインジェクトシーケンスの回数が設定されるため、対応する JSQ1～JSQ4 は同時に書き込まれる必要があります。インジェクト変換が始まる前に ADCx_JSQR が再プログラムされると、再プログラムされたデータはキューに置かれます。コンテキストのキューが空である場合、ADCx_JSQR は 0x0000 として読み出されます。レジスタへのアクセスには、リードモディファイライトシーケンスを使用しないでください。

すでに 2 つのコンテキストがキューに入っているときに ADCx_JSQR がプログラムされると、JQOVF フラグが立って、割込みが生成されます。

25.3.23 プログラム可能な分解能（RES）- 高速変換モード

ADC の分解能を下げることによって、高速変換を行うことができます。

分解能は、制御ビット RES[1:0] をプログラムすることによって、16、14、12、10、または 8 ビットに設定できます。*図 154*、*図 155*、*図 156*、および *図 157* に、分解能とデータの整列に対する変換結果のフォーマットを示します。

分解能を下げることによって変換時間を高速にでき、高いデータ精度を必要としないアプリケーションに有効です。*表 195* のように、これによって逐次比較ステップの間に変換時間は短縮されます。

表 195. T_{SAR} タイミングは分解能に依存

RES	T_{SAR}（ADC クロックサイクル）	T_{SAR}（ns）（F_{ADC}=24 MHz 時）	T_{ADC}（ADC クロックサイクル）（サンプリング時間 = 1.5 ADC クロックサイクル）	T_{ADC}（ns）（F_{ADC}=24 MHz 時）
16	8.5 ADC クロックサイクル	354.2	10 ADC クロックサイクル	416.7
14	7.5 ADC クロックサイクル	312.5	9 ADC クロックサイクル	375
12	6.5 ADC クロックサイクル	270.8	8 ADC クロックサイクル	333.3
10	5.5 ADC クロックサイクル	229.2	7 ADC クロックサイクル	291.7
8	4.5 ADC クロックサイクル	187.5	6 ADC クロックサイクル	250.0

25.3.24 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）

ADC は各レギュラ変換の終了（EOC）イベントと各インジェクト変換の終了（JEOC）イベントをアプリケーションに通知します。

ADC は、新しいレギュラ変換データが ADCx_DR レジスタで使用可能になるとすぐに EOC フラグをセットします。EOCIE ビットがセットされている場合は、割り込みを生成することができます。EOC フラグは、ソフトウェアによって 1 を書き込むことによって、または ADCx_DR を読み出すことによってクリアされます。

ADC は、新しいインジェクト変換データが ADCx_JDRy レジスタの 1 つで使用可能になるとすぐに JEOC フラグをセットします。JEOCIE ビットがセットされている場合は、割り込みを生成することができます。JEOC フラグは、ソフトウェアによって 1 を書き込むことによって、または対応する ADCx_JDRy レジスタを読み出すことによってクリアされます。

さらに、ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。EOSMP フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。EOSMPIE ビットがセットされている場合は、割り込みを生成することができます。

25.3.25 変換シーケンスの終了（EOS、JEOS）

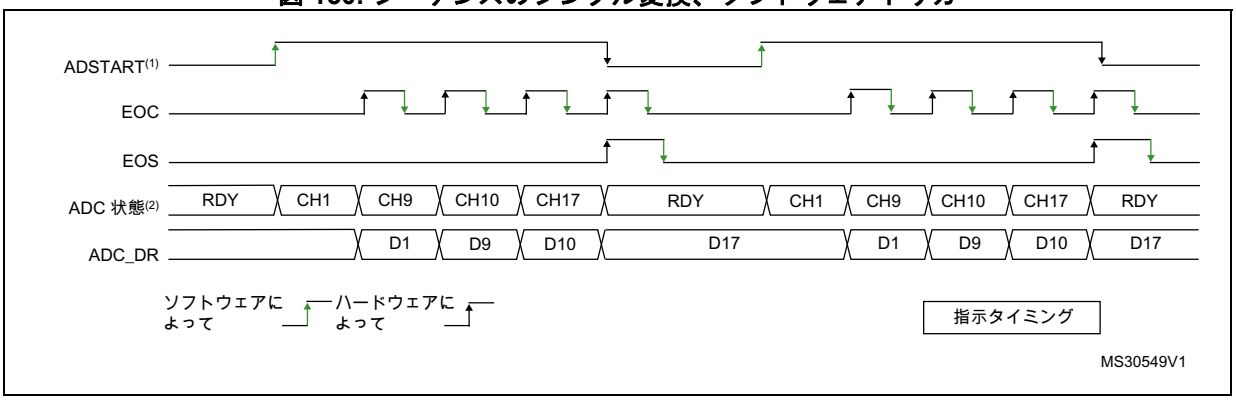
ADC は各レギュラシーケンスの終了（EOC）と各インジェクトシーケンスの終了（JEOC）イベントをアプリケーションに通知します。

ADC は、レギュラ変換シーケンスの最後のデータが ADCx_DR レジスタで使用可能になるとすぐに、EOS フラグをセットします。EOSIE ビットがセットされている場合は、割り込みを生成することができます。EOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC は、インジェクト変換シーケンスの最後のデータが完了するとすぐに、JEOS フラグをセットします。JEOSIE ビットがセットされている場合は、割り込みを生成することができます。JEOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

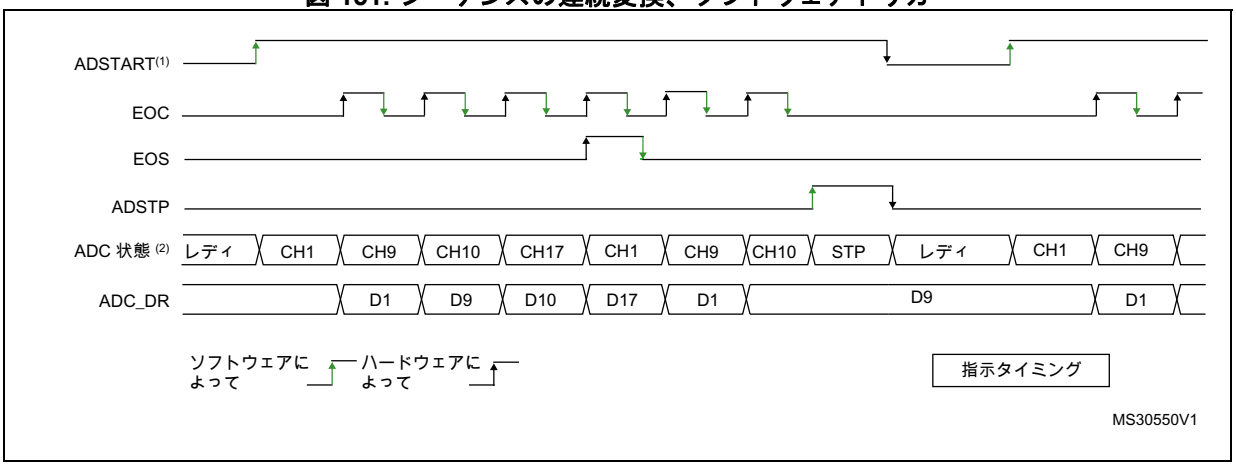
25.3.26 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

図 150. シーケンスのシングル変換、ソフトウェアトリガ



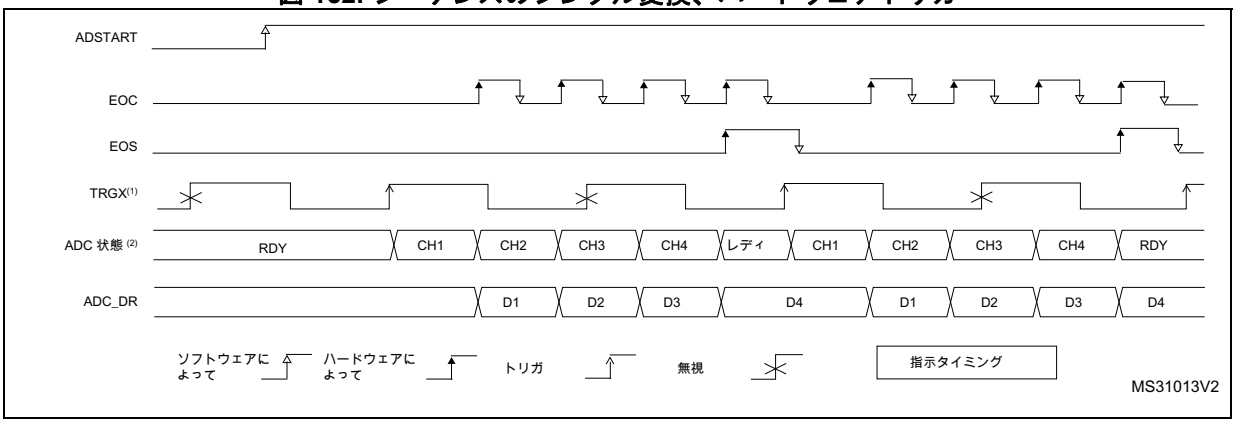
1. EXTEN=0x0、CONT=0
2. 選択されたチャネル = 1、9、10、17；AUTDLY=0

図 151. シーケンスの連続変換、ソフトウェアトリガ



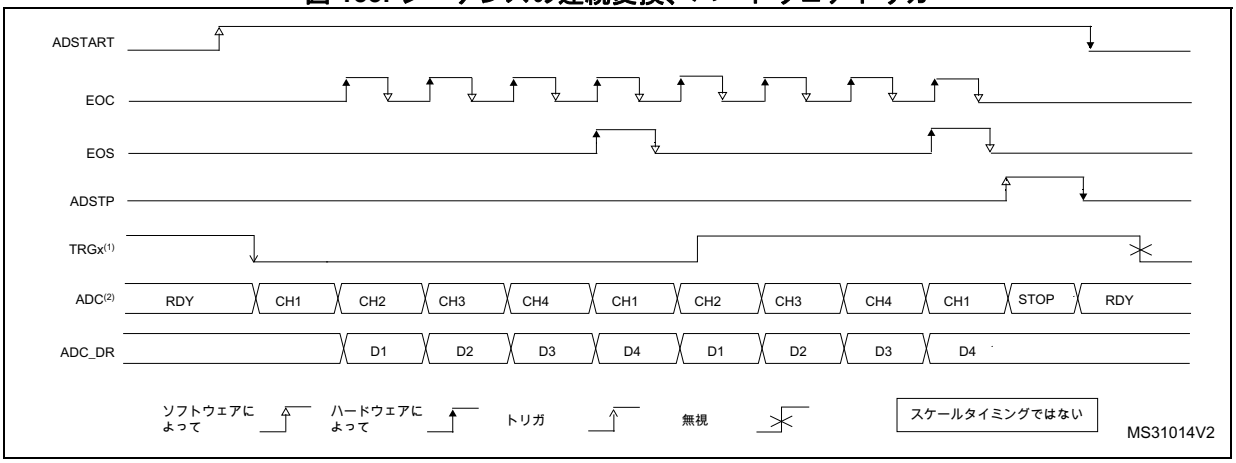
1. EXTEN=0x0、CONT=1
2. 選択されたチャネル = 1、9、10、17；AUTDLY=0

図 152. シーケンスのシングル変換、ハードウェアトリガ



1. TRGx（オーバー周波数）がトリガソースとして選択されます（EXTEN = 01、CONT = 0）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

図 153. シーケンスの連続変換、ハードウェアトリガ



1. TRGx がトリガソースとして選択されます（EXTEN = 10、CONT = 1）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

25.3.27 データ管理

データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）

データおよび配置

各レギュラ変換チャネルの終了時（EOC イベントの発生時）、変換データの結果は 32 ビット幅の ADCx_DR データレジスタに格納されます。

各インジェクト変換チャネルの終了時（JEOC イベントの発生時）、変換データの結果は対応する 32 ビット幅の ADCx_JDRy データレジスタに格納されます。

ADCx_CFGR レジスタの OVSS[3:0] と LSHIFT[3:0] のビットフィールドは、変換後に格納されるデータの配置を選択します。データは、*図 154*、*図 155*、*図 156*、および *図 157* に示すように、右詰めまたは左詰めに配置できます。

注： ***データは、通常モードとオーバーサンプリングモードで再配置できます。***

オフセット

オフセット y （y=1、2、3、4）は、ADCx_OFRy レジスタの OFFSETy[25:0] ビットフィールドに 0 とは異なる値をプログラムすることでチャネルに適用されます。オフセットを適用するチャネルは、ADCx_OFRy レジスタのOFFSETy_CH[4:0] ビットにプログラムされます。この場合、変換された値はOFFSETy[25:0] ビットに書き込まれたユーザ定義のオフセットによって減算されます。結果は負の値になることがあり、そのため読み出しデータは符号付きで、SEXT ビットは拡張符号値を表します。

オフセット値は、最大変換値（16 ビットモードであれば、最大オフセット値は 0xFFFF）未満としてください。

オフセット補正は、オーバーサンプリングモードでもサポートされます。オーバーサンプリングモードでは、OVSS 右シフトが適用される前にオフセットが減算されます。

表 196 に、アナログウォッチドッグ 1、2、3 に可能なすべての分解能での比較方法を示します。

表 196. オフセット計算対データ分解能

分解能（ビット RES[2:0]）	元の変換データとオフセット間での減算：		結果	コメント
	元の変換データ、左詰め	オフセット		
000 : 16 ビット	DATA[15:0]	OFFSET[25:0]	符号付き 27 ビットデータ	-
001 : 14 ビット	DATA[15:2],00	OFFSET[25:0]	符号付き 27 ビットデータ	ユーザは OFFSET[1:0] を 00 に設定する必要があります。
010 : 12 ビット	DATA[15:4],0000	OFFSET[25:0]	符号付き 27 ビットデータ	ユーザは OFFSET[3:0] を 0000 に設定する必要があります。
011 : 10 ビット	DATA[15:6],000000	OFFSET[25:0]	符号付き 27 ビットデータ	ユーザは OFFSET[5:0] を 000000 に設定する必要があります。
100 : 8 ビット	DATA[15:8],000000	OFFSET[25:0]	符号付き 27 ビットデータ	ユーザは OFFSET[7:0] を 00000000 に設定する必要があります。

チャネル “i” に対して ADCx_DR （レギュラチャネル）または ADCx_JDRy （インジェクトチャネル、y=1、2、3、4）からデータを読み出す場合：

- 対応するチャネルのオフセットの 1 つが有効（ビット OFFSETy_EN=1）である場合、読み出しデータは符号付きです。
- このチャネルの 4 つのオフセットに有効なものがない場合、読み出しデータは符号なしです。

図 154、*図 155*、*図 156*、および*図 157*に、符号付きおよび符号なしのデータの配置を、対応する OVSS 値および LSHIFT 値とともに示します。

参考資料

図 154. 右詰め（オフセット無効、符号なしの値）

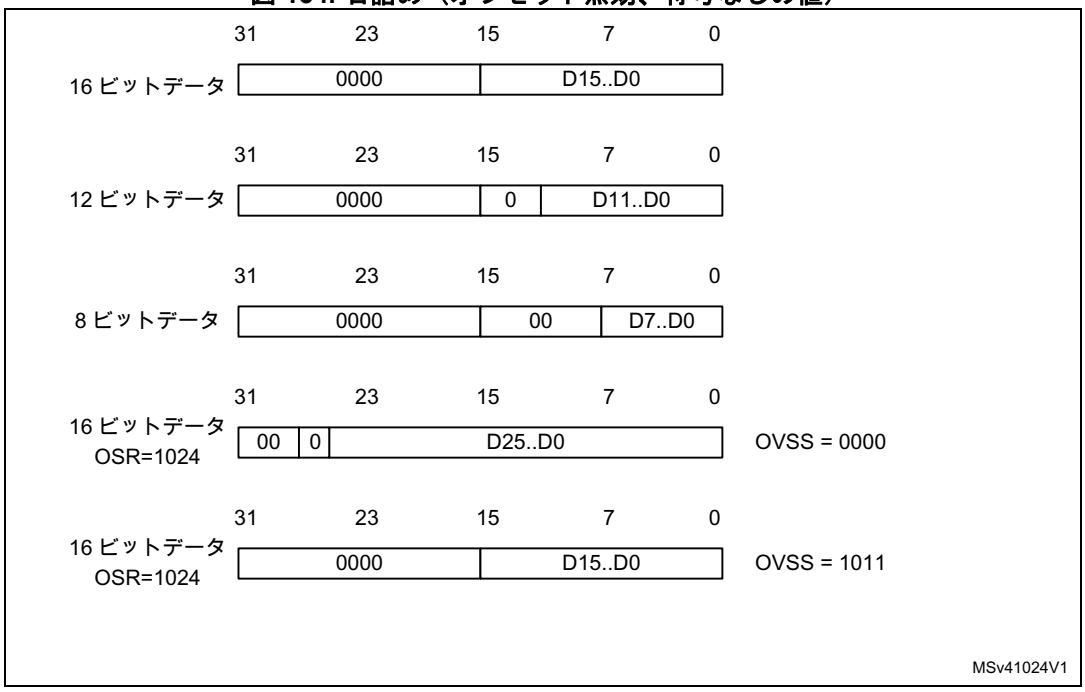


図 155. 右詰め（オフセット有効、符号付きの値）

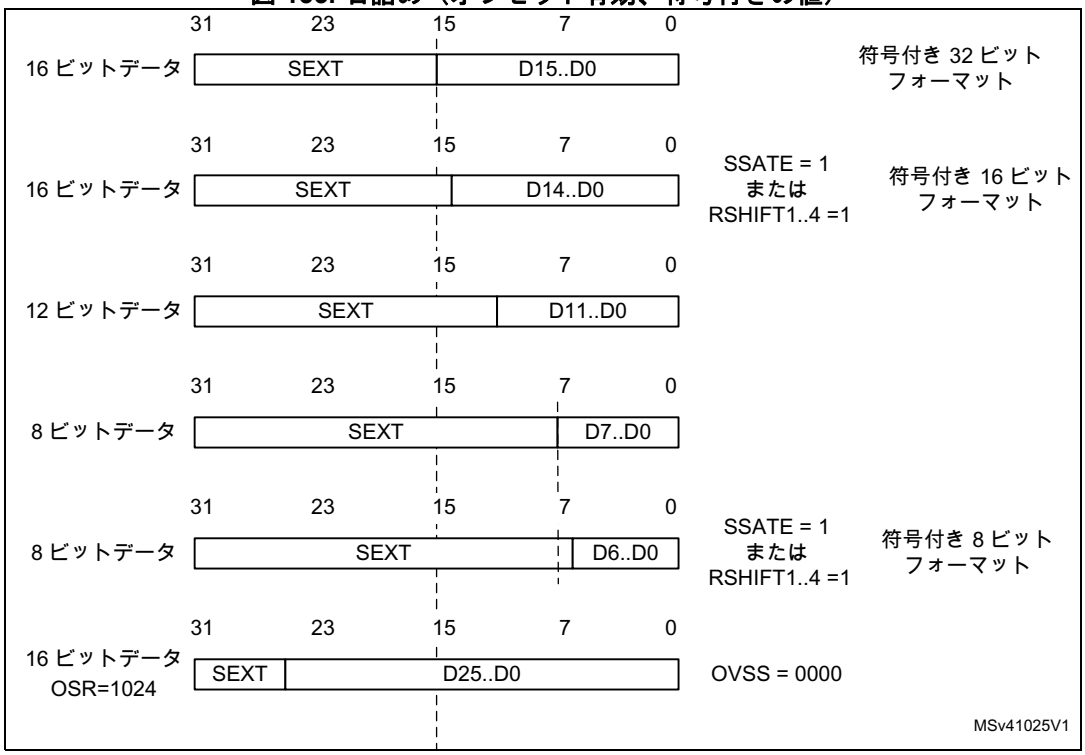


図 156. 左詰め（オフセット無効、符号なしの値）

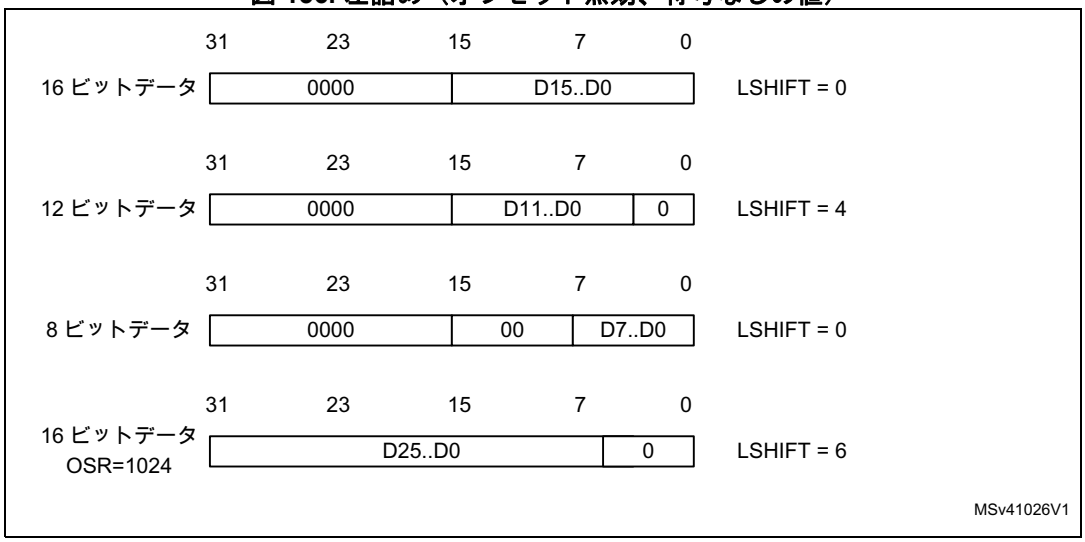


図 157. 左詰め（オフセット有効、符号付きの値）



16 ビットおよび 8 ビット符号付きフォーマットマネージメント：RSHIFTx、SSATE

オフセット補正によってデータフォーマットの符号拡張が行われ、たとえば符号なし 16 ビット変換は、17 ビット符号付きフォーマットに拡張されます。

8 ビットと 16 ビットの変換結果のフォーマットには、3 つのオプションがあります。

オフセット補正チャネル 1～4 それぞれに対して、ADCx_CFGR2 レジスタの RSHIFT1..4 ビットによって、結果を 1 ビット右シフトし、標準の 8 ビットまたは 16 ビットフォーマットに収まるようにすることが可能です。

もう 1 つのオプションは、結果を 16 ビットまたは 8 ビット符号付きフォーマットで飽和させるというものですが、これは、RES[2:0] = 000（16 ビットフォーマット）および RES[2:0] = 100（8 ビットフォーマット）の場合に限られます。

このモードは、ADCx_OFRy レジスタの SSATE ビットによって有効になります。

次の表に、16 ビットフォーマットで利用可能な 3 つのユースケースを示します。

表 197. 16 ビットデータフォーマット

SSATE	RSHIFTx	フォーマット	データ範囲 （オフセット = 0x8000）
0	0	符号拡張された有効桁 17 ビットのデータ SEXT[31:16] DATA[15:0]	0x00007FFF～0xFFFF8000
0	1	符号拡張された有効桁 16 ビットの右シフトデータ SEXT[31:15] DATA[14:0]	0x3FFF～0xC000
1	0	符号拡張された有効桁 16 ビットの飽和データ SEXT[31:15] DATA[14:0]	7FFF～0x8000
1	1	予約済み	-

数値の例を、3 つの異なるオフセット値で*表 198*に示します。

表 198. 16 ビットフォーマットの数値例（太字は飽和を示す）

生の変換結果	オフセット値	結果 SSATE = 0 RSHIFT = 0	結果 SSATE = 0 RSHIFT = 1	結果 SSATE = 1 RSHIFT = 0
0xFFFF	0x8000	0x0000 7FFF	3FFF	7FFF
0x8000		0x0000 0000	0	0
0x0000		0xFFFF 8000	C000	8000
0xFFFF	0x8020	0x0000 7FDF	3FEF	7FDF
0x8000		0xFFFF FFE0	FFF0	FFE0
0x0000		0xFFFF 7FE0	BFF0	8000
0xFFFF	0x7FF0	0x0000 800F	4007	7FFF
0x8000		0x0000 0010	8	0010
0x0000		0xFFFF 8010	C008	8010

オーバーサンプリングモードが有効な場合には、SSATE ビットと RSHIFT1..4 ビットはサポートされません。

ADC オーバーラン（OVR、OVRMOD）

オーバーランフラグ（OVR）は、新しい変換データが使用可能になる前に、（CPU または DMA によって）レギュラ変換データが読み出されなかったときに、バッファオーバーランイベントを通知します。

OVR フラグは、新しい変換が完了した時点で EOC フラグが 1 のままであった場合にセットされます。ビット OVRIE=1 の場合、割り込みを生成することができます。

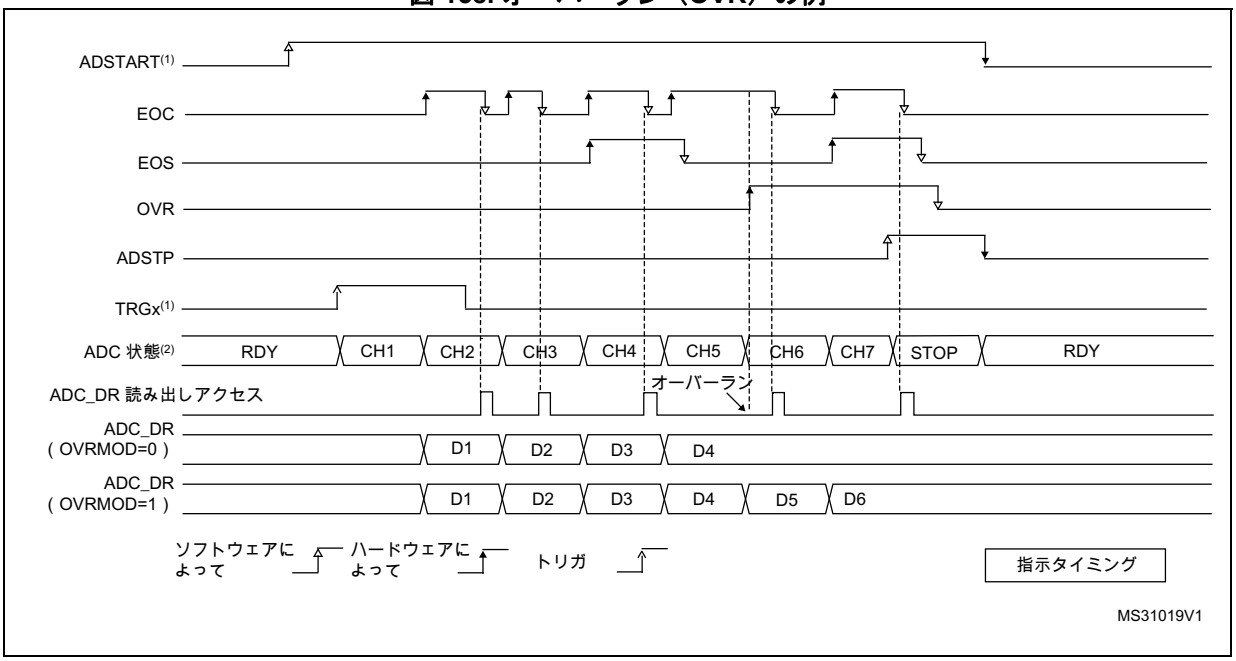
オーバーラン条件が発生すると、ADC は動作し続け、ソフトウェアがビット ADSTP=1 をセットすることによってシーケンスの停止およびリセットを決めるまで、変換を続行できます。

OVR フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

制御ビット OVRMOD をプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD=0 : オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータは保持され、新しい変換は破棄され、失われます。OVR が 1 のままの場合、その後の変換は行われますが、結果データは破棄されます。
- OVRMOD=1 : データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。OVR が 1 のままの場合、その後の変換は通常どおり動作し、ADCx_DR レジスタは常に最新の変換データを含みます。

図 158. オーバーラン（OVR）の例



注： ***インジェクトチャネルでは、4 つのインジェクトチャネルにそれぞれ専用のデータレジスタがあるため、オーバーランを検出する機能はありません。***

DMA を使用しない変換シーケンスの管理

変換が十分遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアは EOC フラグと関連の割り込みを使用して、各データを処理する必要があります。変換が終了するごとに、EOC はセットされ、ADCx_DR レジスタを読み出すことができます。オーバーランイベントをエラーとして管理するには、OVRMOD を 0 に設定する必要があります。

オーバーランなしでの DMA を使用しない変換の管理

毎回データの読み出しをせずに ADC に 1 つまたは複数のチャネルを変換させると便利な場合があります（たとえば、アナログウォッチドッグがある場合）。この場合、OVRMOD ビットを 1 に設定する必要があり、ソフトウェアは OVR フラグを無視する必要があります。オーバーランイベントが発生しても ADC は変換を続行し、ADCx_DR レジスタは常に最新の変換を含みます。

DMA を使用した変換の管理

変換されたチャネルの値は特定のデータレジスタに格納されるので、複数のチャネルの変換には DMA の使用が便利です。これによって、ADCx_DR レジスタにすでに格納されているデータの損失を防ぐことができます。

DMA モードが有効なとき（ADCx_CFGR レジスタの DMNGT ビットがシングル ADC モードで 01 または 11 にセットされている場合、またはデュアル ADC モードで MDMA が 0b00 以外に設定されている場合）、各チャネルの変換後、DMA リクエストが生成されます。これにより、変換データを ADCx_DR レジスタからソフトウェアで選択した場所へ転送することができます。

これにもかかわらず、DMA が DMA 転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR=1）、ADC は DMA リクエストの生成を停止し、新しい変換に対応するデータは DMA によって転送されません。これは、RAM に転送されるすべてのデータを有効とみなすことができることを意味します。

OVRMOD ビットの設定に応じて、データは保存または上書きされます（*セクション：ADC オーバーラン（OVR、OVRMOD）*を参照してください）。

DMA 転送リクエストは、ソフトウェアが OVR ビットをクリアするまでブロックされます。

アプリケーションの用途に応じて 2 つの DMA モードがあり、シングル ADC モードでは ADCx_CFGR レジスタの DMNGT ビットで設定され、デュアル ADC モードでは ADCx_CCR レジスタの DAMDF ビットで設定されます。

- DMA ワンショットモード（DMNGT ビット = 01）
  このモードは、DMA が固定数のデータを転送するようにプログラムされた場合に適しています。
- DMA サーキュラモード（DMNGT ビット = 11）
  このモードは、DMA をサーキュラモードでプログラムする場合に適しています。

DMA ワンショットモード（DMNGT=01）

このモードでは、ADC は新しい変換データが使用可能になるたびに DMA 転送リクエストを生成し、変換が再び開始された場合でも、DMA が最後の DMA 転送に達すると（DMA_EOT 割り込みが発生すると）（DMA の段落を参照）、DMA リクエストの生成を停止します。

DMA 転送完了後（DMA コントローラで設定されたすべての転送後）：

- ADC データレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMA コントローラに対する新しいDMA リクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMA は停止します。

DMA サーキュラモード（DMNGT=11）

このモードでは、DMA が最後の DMA 転送に達した場合でも、ADC は新しい変換データがデータレジスタで使用可能になるたびに DMA 転送リクエストを生成します。これにより、DMA を連続的なアナログ入力データストリームを処理するようにサーキュラモードに設定できます。

FIFO 付き DMA

出力データレジスタには 8 段 FIFO が備わっています。2 つの異なる DMA リクエストが並列で生成されます。データが 1 つの場合には「SREQ シングルリクエスト」が生成され、データが 4 つの場合には「BREQ バーストリクエスト」が生成されます。このモードに応じて、DMA2 は、シングル転送モードとインクリメンタルバーストモード（4 ビート）のどちらでもプログラム可能です。正しいリクエストラインが DMA2 によって選択されます。詳細については、DMA2 の章を参照してください。

25.3.28 DFSDM を使用した変換の管理

ADC 変換結果は、デルタシグマモジュレータのデジタルフィルタ（DFSDM）に直接転送できます。

この場合、DMNGT[1:0] ビットは 10 にセットしてください。

ADC は、レギュレータデータレジスタデータの最下位 16 ビットを DFSDM に転送し、転送が有効になると EOC フラグをリセットします。

データフォーマットは、16 ビット符号である必要があります。

ADCx_DR[31:16] = 無視

ADCx_DR[15] = 符号

ADCx_DR[14:0] = データ

16 ビット符号付きフォーマットを超えるあらゆるデータは切り捨てられます。

25.3.29 動的低電力機能

自動遅延変換モード（AUTDLY）

ADC は、AUTDLY 設定ビットによって制御されている自動遅延変換モードを実装しています。自動遅延変換は、ソフトウェアを単純化する場合だけでなく、ADC オーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化する場合に便利です。

AUTDLY=1 の場合、新しい変換は同じグループの前のデータがすべて処理された場合にのみ開始できます。

- レギュラ変換の場合：ADCx_DR レジスタの読み出し後、または EOC ビットのクリア後（*図 159* を参照）
- インジェクト変換の場合：JEOS ビットのクリア後（*図 160* を参照）

これは、データを読み出すシステムの速度に ADC の速度を自動的に適応させる方法です。

遅延は、各レギュラ変換後（DISCEN=0 か 1 かにかかわらず）、およびインジェクト変換の各シーケンス後（JDISCEN=0 か 1 かにかかわらず）に挿入されます。

注： ***遅延は、インジェクトシーケンスの各変換の間には挿入されません（最後の変換を除く）。***

変換時、この遅延中に発生した（同じ変換グループに対する）ハードウェアトリガイベントは無視されます。

注： ***ソフトウェアトリガは、変換を再開するために、この遅延中にADSTART ビットまたは JADSTART ビットをセットできるようにしている点で、ハードウェアトリガとは異なります。新しい変換を起動する前にデータを読み出すかどうかは、ソフトウェアに依存します。***

遅延は、異なるグループの変換（レギュラ変換に続いてインジェクト変換、またはその逆）の間には挿入されません。

- レギュラ変換の自動遅延中にインジェクトトリガが発生すると、すぐにインジェクト変換が開始されます（*図 160* を参照）。
- インジェクトシーケンスが完了すると、新しいレギュラ変換を開始する前に、ADC は前のレギュラ変換の遅延を待ちます（終了していない場合）（*図 162* を参照）。

自動インジェクトモード（JAUTO=1）では挙動が少し異なり、前のインジェクト変換シーケンスの自動遅延が終了した場合（JEOS がクリアされた場合）にのみ新しいレギュラ変換を開始することができます。これは、新しいシーケンスを開始する前に、ソフトウェアが特定のシーケンスのデータをすべて読み出すことができるようにするためです（*図 163* を参照）。

自動遅延モード（JAUTO=1、CONT=1、および AUTDLY=1）と組み合わされた連続自動インジェクションモードの変換を停止するには、次の手順に従います。

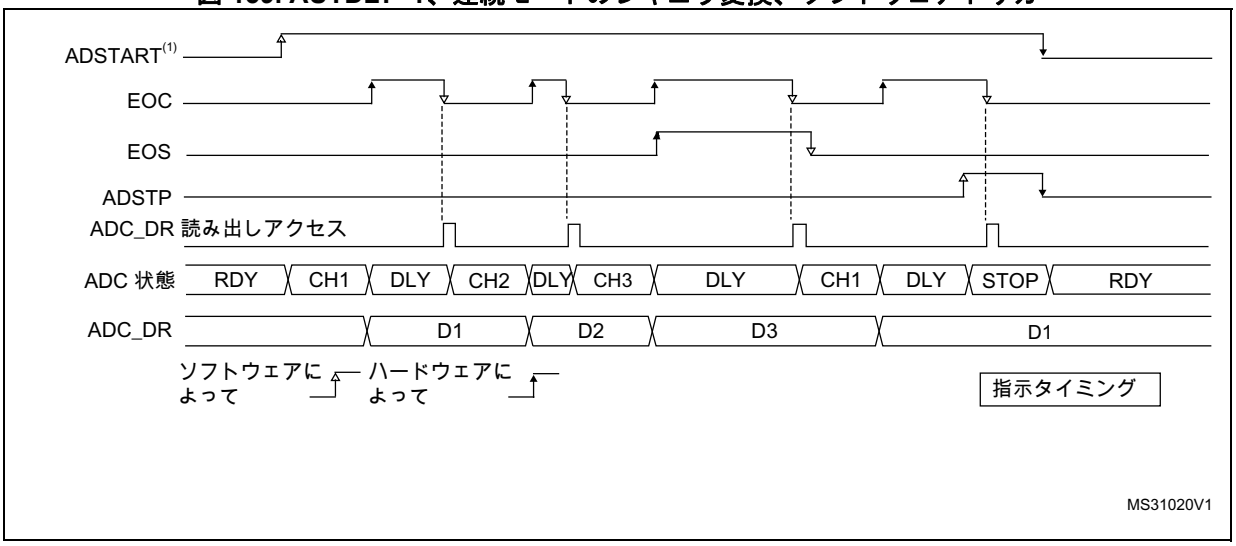
1. JEOS=1 になるまで（リスタートされる変換がなくなるまで）待ちます。
2. JEOS をクリアします。
3. ADSTP=1 にセットします。
4. レギュラデータを読み出します。

この手順に従わない場合、ADSTP がセットされた後で JEOS がクリアされると、新しいレギュラシーケンスが再開されます。

AUTDLY モードでは、すでに実行中のレギュラシーケンス中、またはシーケンスの最後のレギュラ変換後の遅延中に発生した場合、ハードウェアレギュラトリガイベントは無視されます。ただし、この遅延後に発生した場合は、後続の遅延のインジェクトシーケンス中に発生した場合でも、保留とみなされます。その後、インジェクトシーケンスの遅延終了時に変換が開始されます。

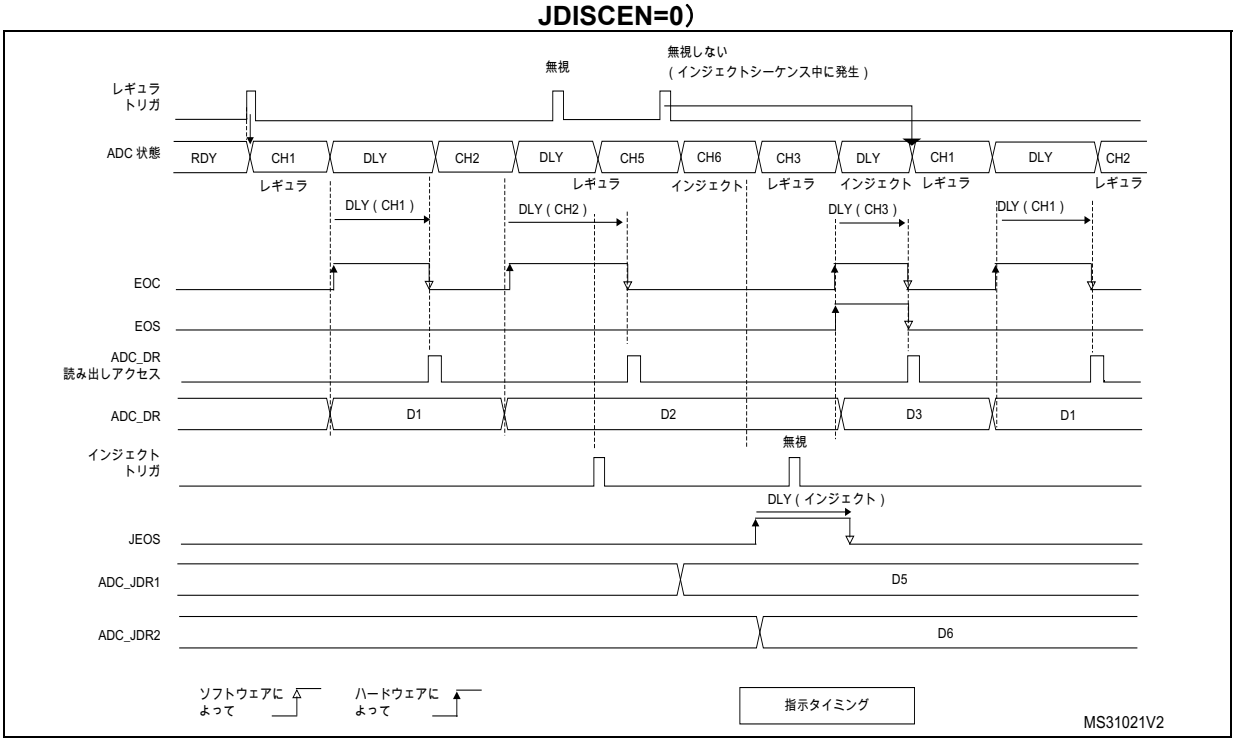
AUTDLY モードでは、すでに実行中のインジェクトシーケンス中、またはシーケンスの最後のインジェクト変換後の遅延中に発生した場合、ハードウェアインジェクトトリガイベントは無視されます。

図 159. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、CHANNELS = 1、2、3
3. インジェクト設定：無効

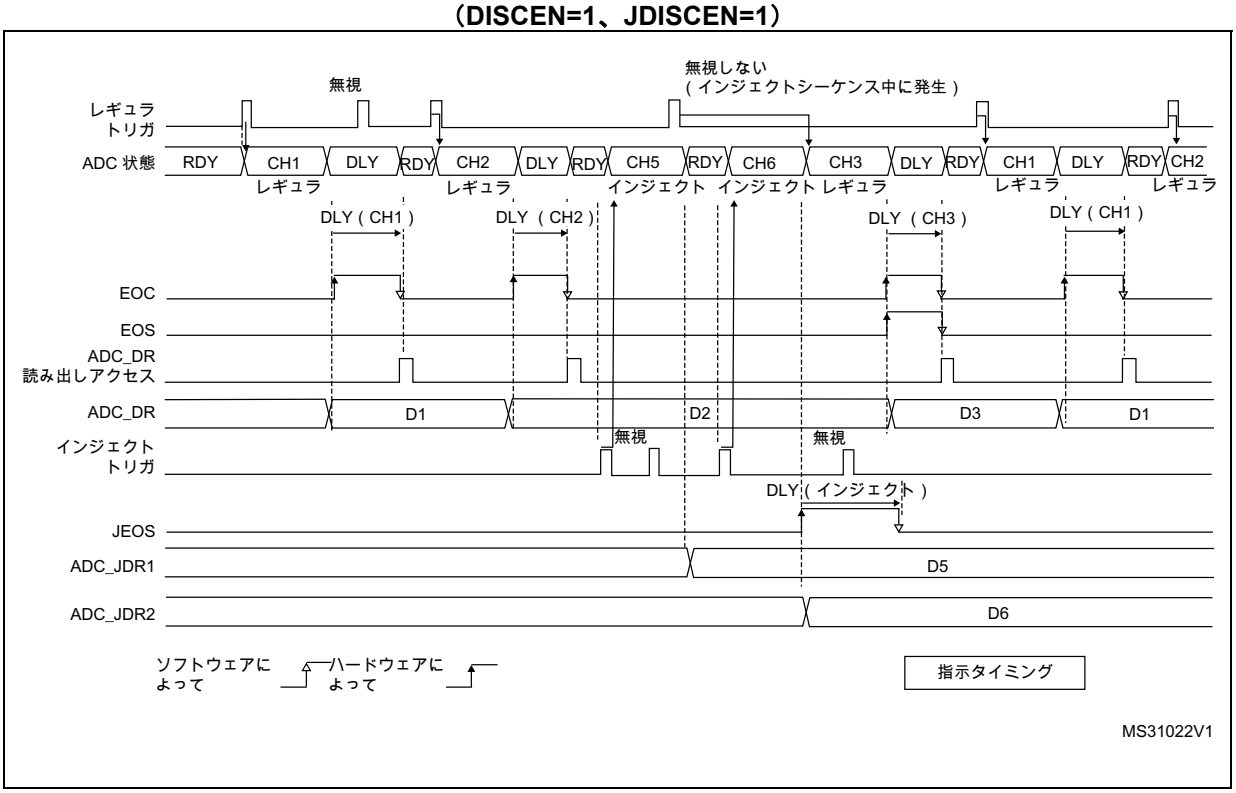
図 160. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=0、JDISCEN=0）



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

参考資料

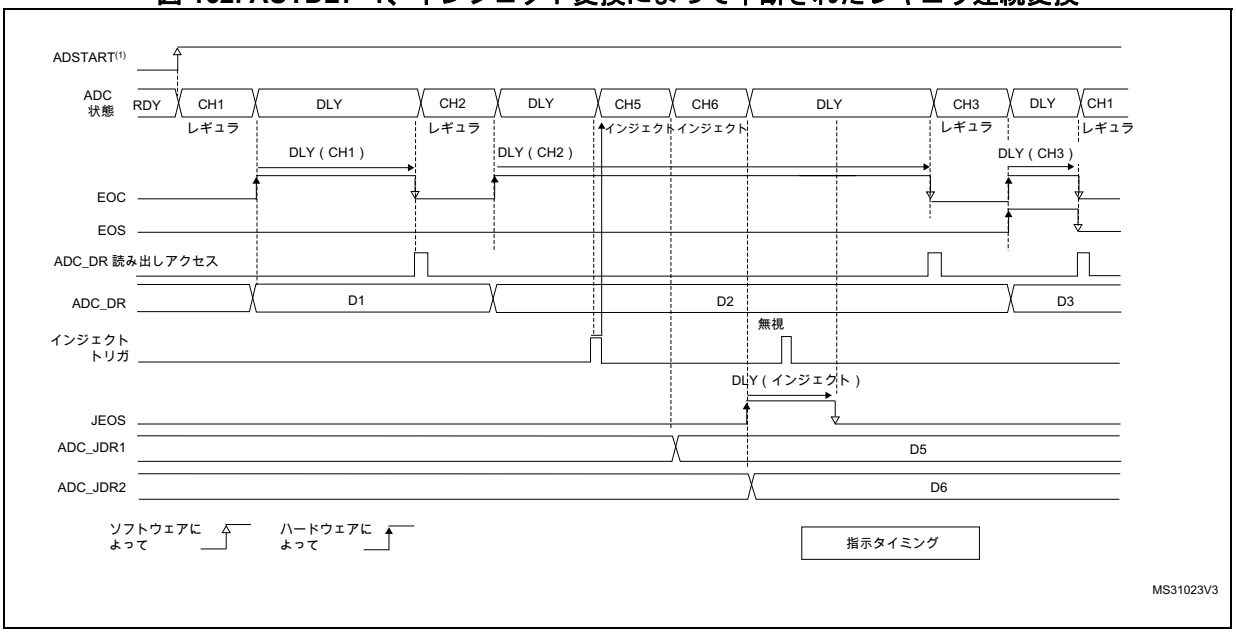
図 161. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=1、JDISCEN=1）



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=1、DISCNUM=1、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=1、CHANNELS = 5、6

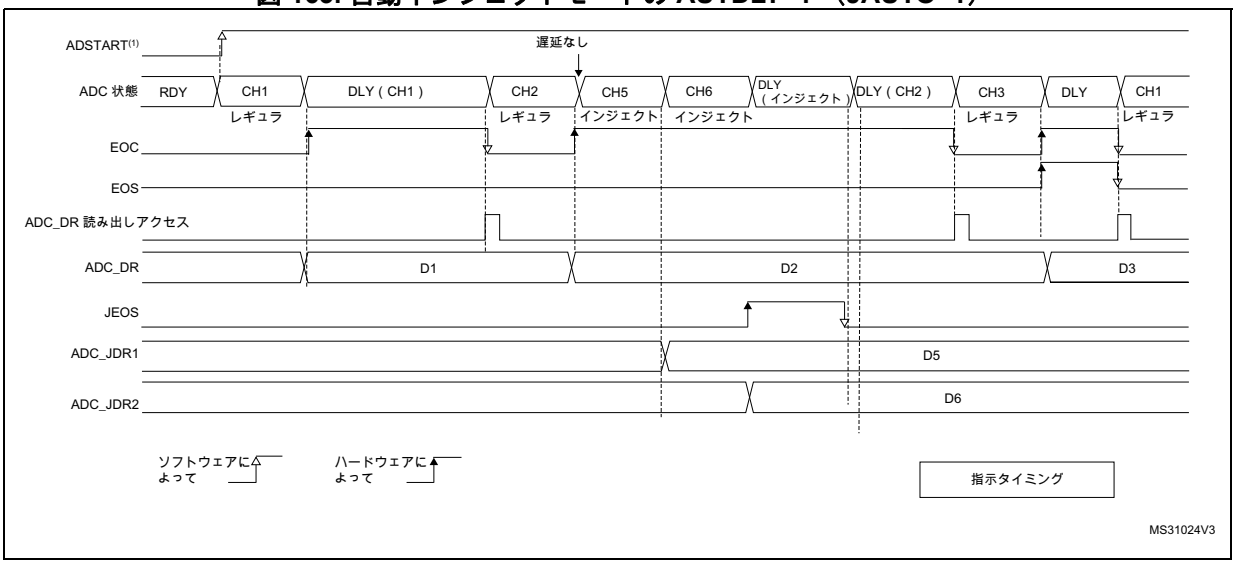
参考資料

図 162. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

図 163. 自動インジェクトモードの AUTDLY=1 （JAUTO=1）

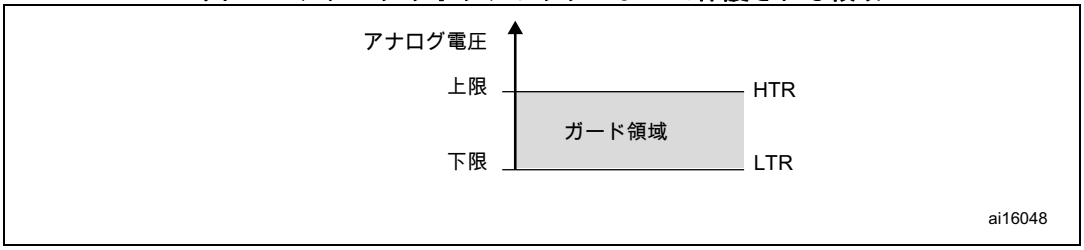


1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2
3. インジェクト設定：JAUTO=1、CHANNELS = 5、6

25.3.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）

3 つの AWD アナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

図 164. アナログウォッチドッグによって保護される領域



AWDx フラグと割り込み

ADCx_IER レジスタの AWDyIE （x=1、2、3）をセットすることによって、3 つのアナログウォッチドッグでそれぞれ割り込みを有効にできます。

AWDy （y=1、2、3）フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC 変換結果は配置される前に低閾値および高閾値と比較されます。

アナログウォッチドッグ 1 の説明

AWD アナログウォッチドッグ 1 は、ADCx_CFGR レジスタの AWD1EN ビットをセットすることによって有効になります。ウォッチドッグは、選択された 1 つのチャネルまたはすべての有効チャネル (1) が設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視します。

表 199 に、1 つまたは複数のチャネル上でアナログウォッチドッグを有効にするためにどのようにして ADCx_CFGRy レジスタを設定するかを示します。

表 199. アナログウォッチドッグチャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGL ビット	AWD1EN ビット	JAWD1EN ビット
なし	x	0	0
すべてのインジェクトチャネル	0	0	1
すべてのレギュラチャネル	0	1	0
すべてのレギュラおよびインジェクトチャネル	0	1	1
1 つの(1) インジェクトチャネル	1	0	1
1 つの (1) レギュラチャネル	1	1	0
1 つの (1) レギュラまたはインジェクトチャネル	1	1	1

1. AWDyCH[4:0] ビットによって選択チャネルは、適切なレギュラまたはインジェクトシーケンスで変換されるようにプログラムする必要があります。

アナログウォッチドッグ（AWD1）ステータスビットは、ADC によって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。

これらの閾値は、アナログウォッチドッグ 1 の ADCx_HTR1 レジスタの HTR1[25:0] ビットおよび ADCx_LTR1 レジスタの LTR1[25:0] ビットでプログラムされます。

閾値は、最高 26 ビット（オーバーサンプリング付き 16 ビット分解能、OSR=1024）まで可能です。

16 ビット未満の分解能でデータを変換するときには（RES[2:0] ビットに従って）、内部比較は常に 16 ビット全体の変換データ（半ワード境界で左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

表 200 に、アナログウォッチドッグ 1、2、3 に可能なすべての分解能での比較方法を示します。

表 200. アナログウォッチドッグ 1、2、3 の比較

分解能（RES[2:0] ビット）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
000：16 ビット	DATA[15:0]	LTR1[25:0] および HTR1[25:0]	-
001：14 ビット	DATA[15:2],00	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[1:0] と HTR1[1:0] を 00 に設定する必要があります。
010：12 ビット	DATA[15:4],0000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[3:0] と HTR1[3:0] を 0000 に設定する必要があります。
011：10 ビット	DATA[15:6],000000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[5:0] と HTR1[5:0] を 000000 に設定する必要があります。
100：8 ビット	DATA[15:8],00000000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[7:0] と HTR1[7:0] を 00000000 に設定する必要があります。

1. 配置計算の前、かつオフセットを適用する前に、元の変換データに対してウォッチドッグ比較が行われます（比較対象のデータは符号なし）。

アナログウォッチドッグ 2 および 3 の説明

2 番目と 3 番目のアナログウォッチドッグはより柔軟性の高く、AWDCHy[19:0]（y=2、3）の対応するビットをプログラムすることにより、いくつかの選択されたチャネルをガードすることができます。

対応するウォッチドッグは、AWDCHy[19:0]（y=2、3）の任意のビットがセットされると有効になります。

閾値は、最高 26 ビット（オーバーサンプリング付き 16 ビット分解能、OSR=1024）まで可能であり、ADCx_HTR2、ADCx_LTR2、ADCx_LTR3、および ADCx_HTR3 のレジスタでプログラムされます。

16 ビット未満の分解能でデータを変換するときには（RES[2:0] ビットに従って）、内部比較は常に 16 ビット全体の変換データ（半ワード境界で左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

ADCx_AWDy_OUT 信号出力生成

各アナログウォッチドッグは、一部のオンチップタイマに ETR 入力（外部トリガ）が直接接続された内部ハードウェア信号 ADCx_AWDy_OUT （x=ADC の数、y=ウォッチドッグの数）と関連付けられています。ETR として ADCx_AWDy_OUT 信号を選択する方法を理解するには、オンチップタイマのセクションを参照してください。

ADCx_AWDy_OUT は、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADCx_AWDy_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADCx_AWDy_OUT は、次のガードされた変換がプログラムされた閾値内にある場合、終了後にリセットされます（次のガードされた変換がプログラムされた閾値の外にある場合は 1 のまま維持されます）。
- ADCx_AWDy_OUT は ADC を無効にする場合にもリセットされます（ADDIS=1 に設定する場合）。レギュラまたはインジェクト変換の停止（ADSTP=1 または JADSTP=1 に設定）は、ADCy_AWDx_OUT の生成にまったく影響しない点に注意してください。

注： ***AWDx フラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。AWDy フラグは、ADCx_AWDy_OUT の生成にまったく影響しません（ソフトウェアによって AWDx フラグがクリアされず、このフラグが 1 のまま維持される間、ADCy_AWDy_OUT をトグルできます）。***

図 165. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）

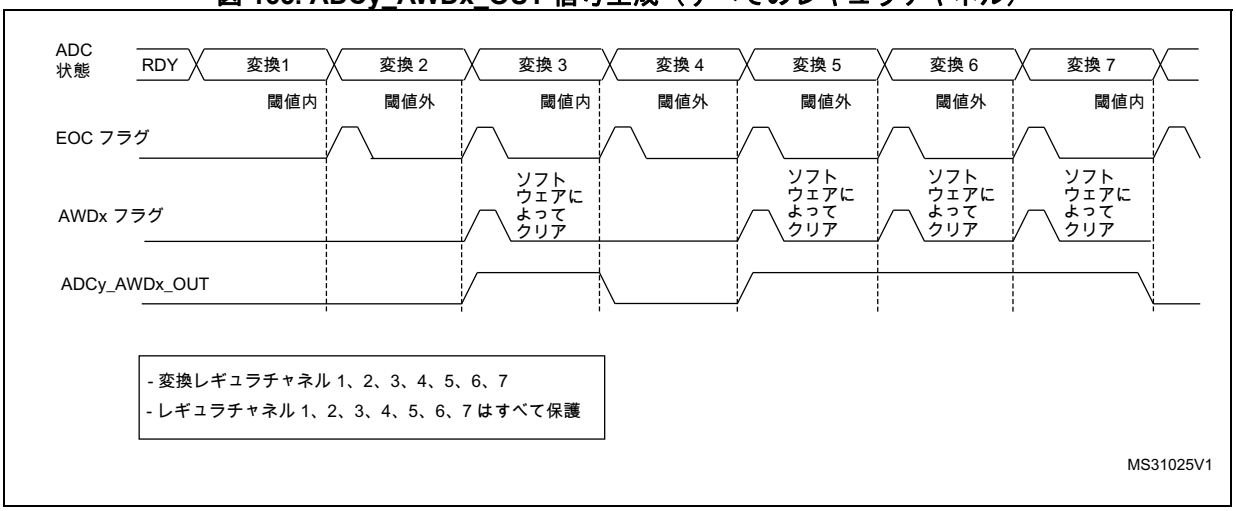


図 166. ADCy_AWDx_OUT 信号生成（ソフトウェアによってAWDx フラグがクリアされない場合）

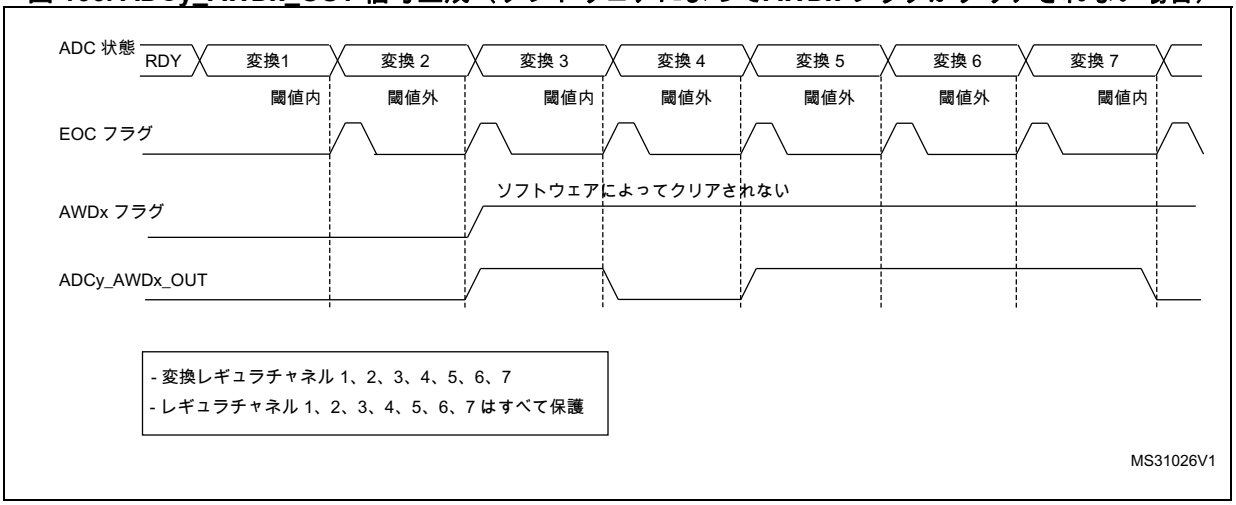


図 167. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）

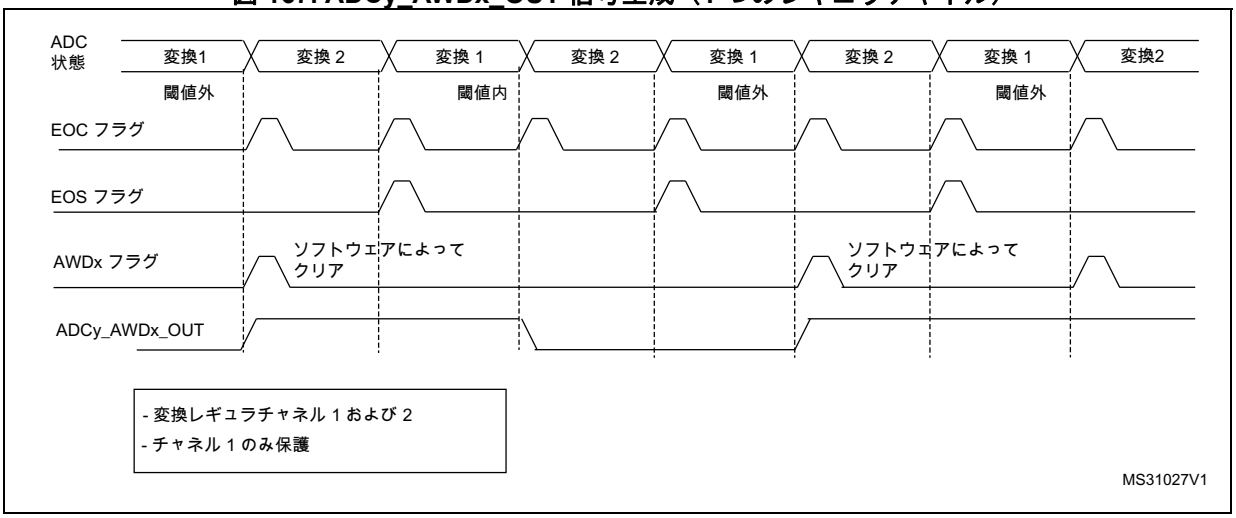
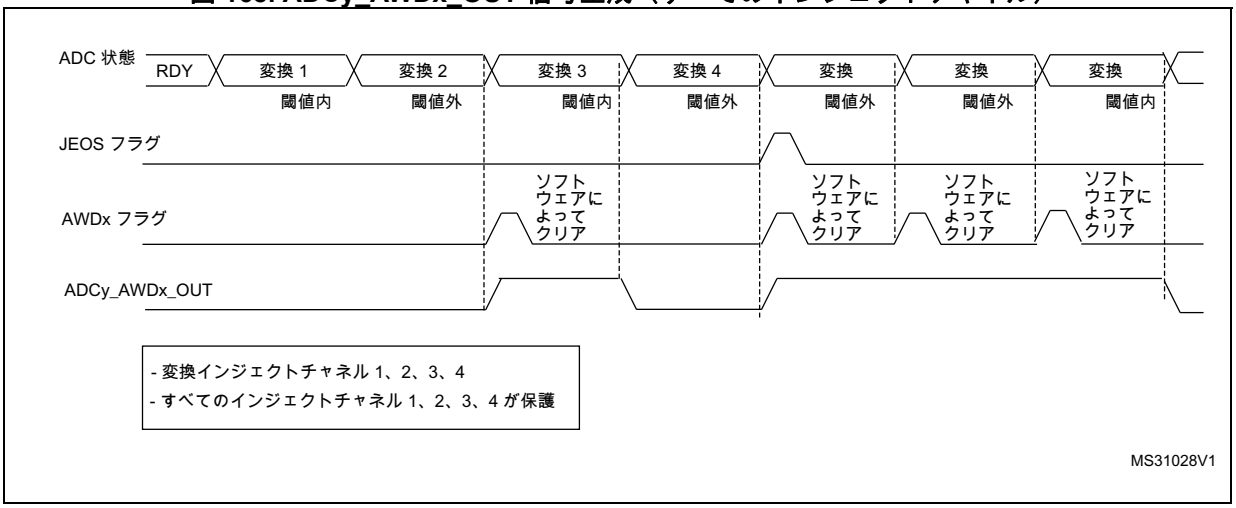


図 168. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル）



25.3.31 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPU の負荷を軽減します。複数の変換を処理して、最大 26 ビット（16 ビットの値と OSR = 1024）幅の単一データに平均化できます。

以下の形式で結果を提供します。N および M は調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

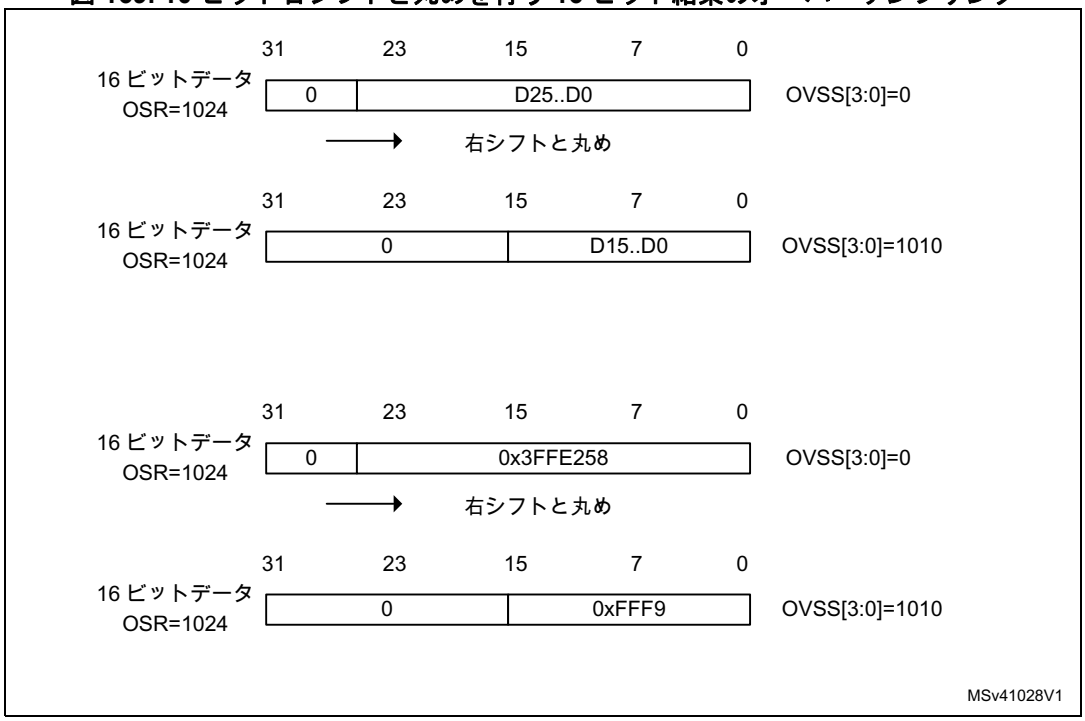
平均化、データレートの削減、SNR の向上、基本的フィルタリングをハードウェアによって実行できます。

オーバーサンプリング比 N は、ADCx_CFGR2 レジスタの OSR[9:0] ビットによって定義され、その範囲は 2～1024倍 です。分周係数 M は、最大 10 ビットの右ビットシフトから成っており、ADCx_CFGR2 レジスタの OVSS[3:0] ビットを使用して定義します。

合計ユニットは最大 26 ビット（1024 x 16 ビットの結果）の結果をもたらし、この値は左または右へシフトできます。右シフトが選ばれた場合には、シフトによって残った最下位側ビットを使用して最も近い値に丸められてから ADCx_DR データレジスタに転送されます。

表 169 に、元の 26 ビットの累積データから最終的な 16 ビットの結果への処理の数値例を示します。

図 169. 10 ビット右シフトと丸めを行う 16 ビット結果のオーバーサンプリング



オーバーサンプリングモードでの変換タイミングに変更はありません。サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータは N 回の変換ごとに提供され、同等遅延は N x T_{CONV} = N x (t_{SMPL} + t_{SAR}）に等しくなります。フラグは、次のようにセットされます。

- サンプリングフェーズの終了（EOSMP）は、各サンプリングフェーズ後にセットされます。
- 変換の終了（EOC）は、N 回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOS）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N x シーケンス長の変換合計後）。

オーバーサンプリング時のシングル ADC 動作モードのサポート

オーバーサンプリングモードでは、ほとんどの ADC 動作モードが維持されます。

- シングルまたは連続モードの変換
- ソフトウェアまたはトリガによる ADC 変換の開始
- 変換中の ADC の停止（中止）
- オーバーラン検出時の CPU または DMA 経由でのデータの読み出し
- 低電力モード（AUTDLY）
- プログラム可能な分解能：この場合、削減された変換値（ADCx_CFGR レジスタの RES[2:0] ビットに従って）の累積、切り詰め、丸め、およびシフトは、16 ビット変換と同様に行われます。

注： ***オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADCx_CFGR の ALIGN ビットは無視され、データは常に右詰めで提供されます。***

注： ***オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSE および／または JOVSE ビットがセットされている場合、ADCx_OFRy レジスタの OFFSETy_EN ビットの内容は無視されます（リセットとみなされます）。***

アナログウォッチドッグ

アナログウォッチドッグ機能は保持されますが（AWDSGL および AWDEN ビット）、次のような違いがあります。

- RES[2:0] ビットは無視され、比較は常に 12 ビット値 HT[11:0] および LT[11:0] を使用して行われます。
- 比較は、16 ビットのオーバーサンプリングされた結果の上位 12 ビット ADCx_DR[15:4] に対して行われます。

注： ***高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなります。たとえば、オーバーサンプリングされた結果が 4 ビットシフトされた場合、12 ビットの右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8 ビットに対してだけです。比較は ADCx_DR[11:4] と HT[0:7] / LT[[0:7] の間で行われ、HT[11:8] / LT[11:8] はリセットする必要があります。***

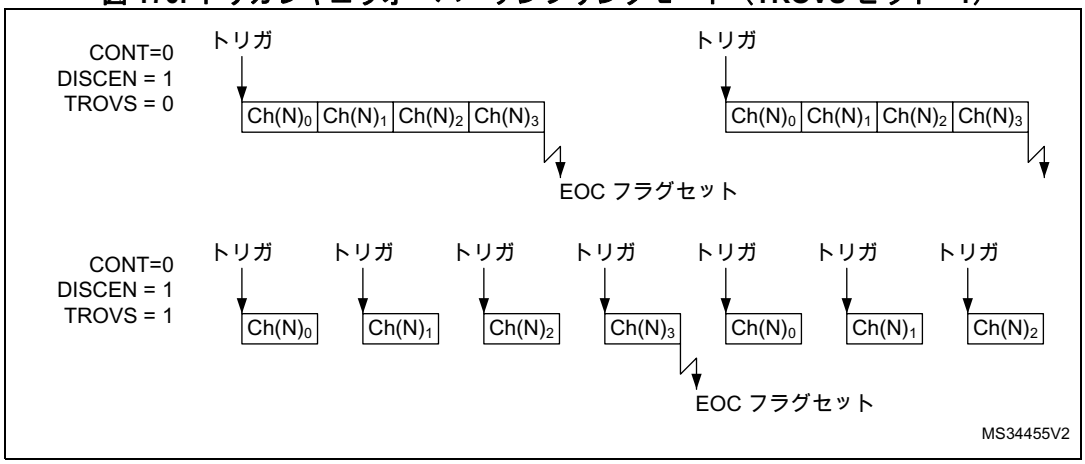
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり強力なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、ADCx_CFGR2 の TROVS ビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

図 170 に、不連続モード中にトリガに反応して変換を開始する方法を示します。

TROVS ビットがセットされている場合、DISCEN ビットの内容は無視され、1 とみなされます。

図 170. トリガレギュラオーバーサンプリングモード（TROVS ビット =1）



オーバーサンプリング時のインジェクトおよびレギュラシーケンサ管理

オーバーサンプリングモードでは、インジェクトおよびレギュラシーケンサに異なる動作を行わさせることができます。シーケンサを同時に使用する必要がある場合、それら両方を制限付きでオーバーサンプリングを有効にできます（これは、独自の累算ユニットに関連します）。

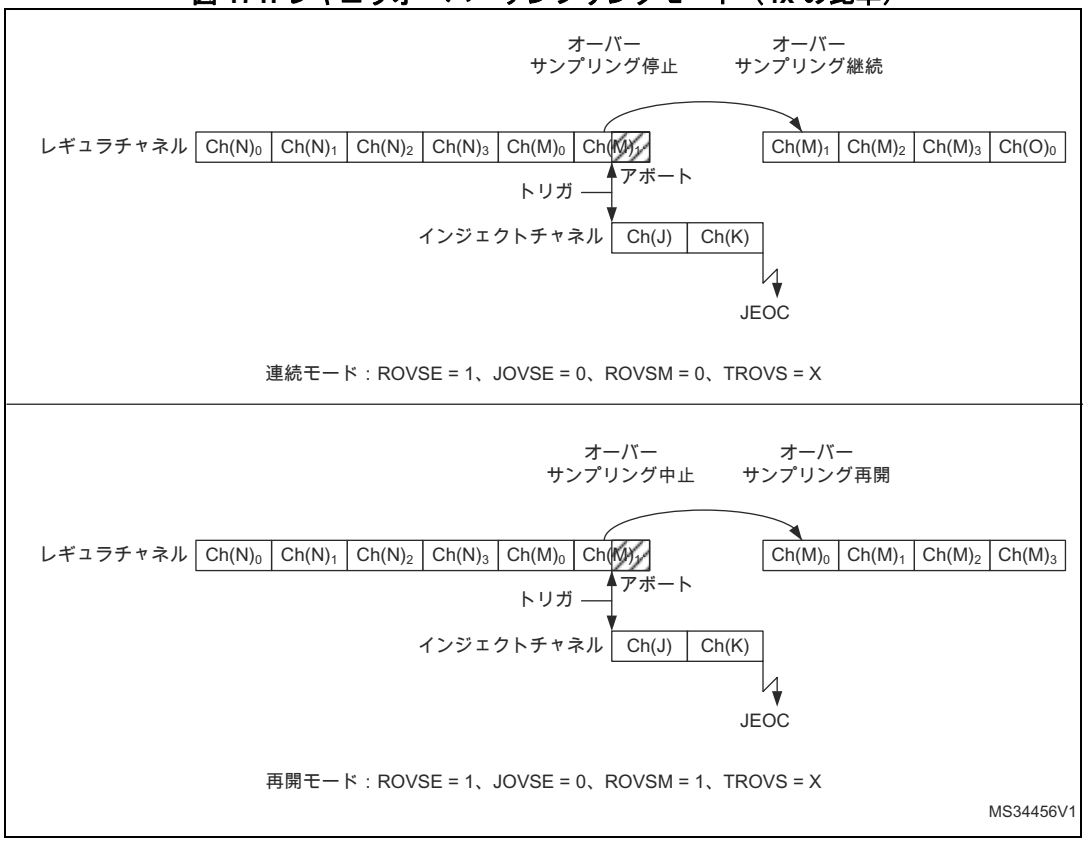
オーバーサンプリングレギュラチャネルのみ

レギュラオーバーサンプリングモードビット ROVSM は、インジェクト変換によって割り込みが発生した場合のレギュラサンプリングシーケンスの再開方法を定義します。

- 連続モードでは、最後の有効なデータ（インジェクトトリガによる変換中止リクエストの前）から累算を再開します。これにより、インジェクションの頻度に関係なく、オーバーサンプリングを完了できます（トリガ間で最低 1 回のレギュラ変換を完了できます）。
- 再開モードでは、累算が 0 から再開します（前の変換結果は無視されます）。このモードでは、オーバーサンプリングに使用されるすべてのデータが 1 つの時間スロット内で連続で変換されたことを保証できます。オーバーサンプリング時間を超えるインジェクショントリガ時間がかかることに注意が必要です。この条件が満たされない場合、オーバーサンプリングを完了できず、レギュラシーケンサはブロックされます。

図 171 では、4倍のオーバーサンプリング比の例を示しています。

図 171. レギュラオーバーサンプリングモード（4x の比率）



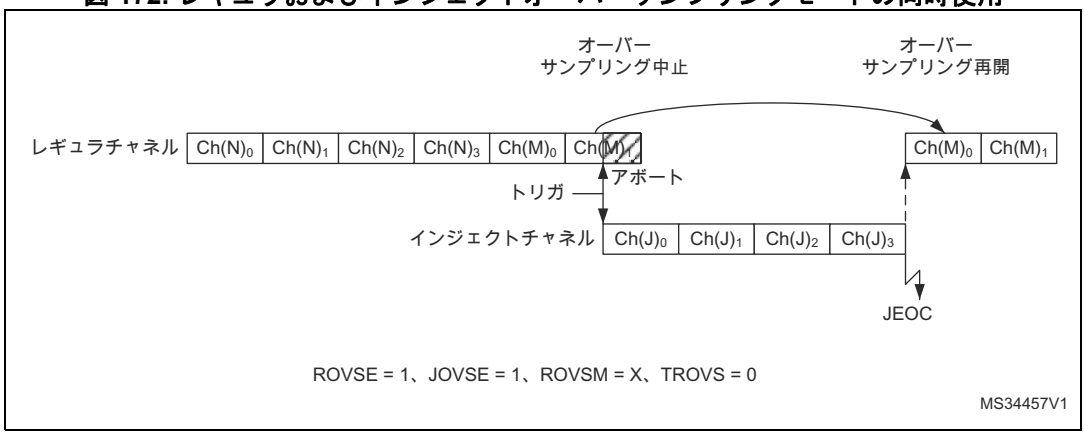
オーバーサンプリングインジェクトチャネルのみ

インジェクトオーバーサンプリングモードビット JOVSE は、インジェクトシーケンサでの変換専用のオーバーサンプリングを有効にします。

オーバーサンプリングレギュラおよびインジェクトチャネル

ROVSE ビットおよび JOVSE ビットの両方をセットできます。この場合、レギュラオーバーサンプリングモードは以下の*図 172* に示すように、強制的に再開モードになります（ROVSM ビット無視）。

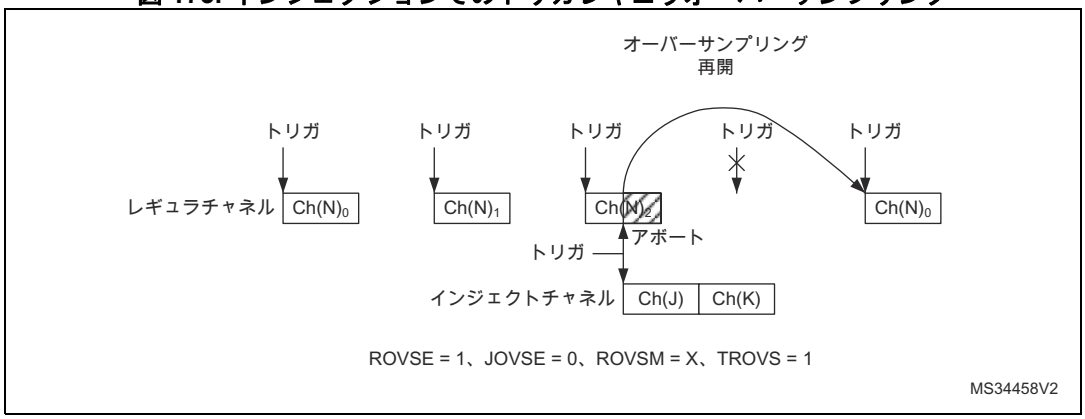
図 172. レギュラおよびインジェクトオーバーサンプリングモードの同時使用



インジェクト変換でのトリガレギュラオーバーサンプリングモード

インジェクト変換でのトリガレギュラオーバーサンプリングモードが可能です。この場合、インジェクトモードオーバーサンプリングモードは無効になり、ROVSM ビットは無視されます（再開モードが強制されます）。JOVSE ビットをリセットする必要があります。動作については、以下*図 173* に示します。

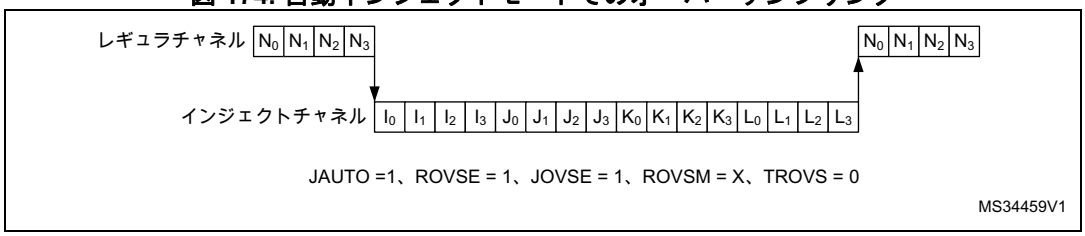
図 173. インジェクションでのトリガレギュラオーバーサンプリング



自動インジェクトモード

自動インジェクトシーケンスをオーバーサンプリングして、レジスタにすべての変換結果を格納し、DMA リソースを保存できます。このモードは、レギュラおよびインジェクトオーバーサンプリング両方がアクティブの状態でのみ使用可能です。JAUTO = 1、ROVSE = 1 および JOVSE = 1 の場合です。その他の組み合わせはサポートされません。ROVSM ビットは自動インジェクトモードでは無視されます。以下の*図 174* に、変換シーケンスの方法を示します。

図 174. 自動インジェクトモードでのオーバーサンプリング



TROVS ビットを使用してトリガモードを有効にすることもできます。この場合、ADC は次のように設定する必要があります。JAUTO = 1、DISCEN = 0、JDISCEN = 0、ROVSE = 1、JOVSE = 1 および TROVSE = 1。

オーバーサンプリング時のデュアル ADC モードのサポート

デュアル ADC 設定で動作時、インジェクト同時モードおよびレギュラ同時モードに対してオーバーサンプリングを有効にできます。この場合、2 つの ADC をまったく同じ設定（オーバーサンプリング含む）でプログラムする必要があります。

他のすべてのデュアル ADC モードは、レギュラまたはインジェクトオーバーサンプリングが有効になっている場合（ROVSE = 1 または JOVSE = 1）サポートされません。

組み合わせモードの概要

以下の*表 201* に、サポートされていないモードを含むすべての組み合わせの概要を示します。

表 201. オーバーサンプリング動作モードの概要

レギュラオーバーサンプリング ROVSE	インジェクトオーバーサンプリング JOVSE	オーバーサンプリングモード ROVSM 0 = 連続 1 = 再開	トリガレギュラモード TROVS	コメント
1	0	0	0	レギュラ連続モード
1	0	0	1	サポートされません
1	0	1	0	レギュラ再開モード
1	0	1	1	トリガレギュラ再開モード
1	1	0	X	サポートされません
1	1	1	0	インジェクトおよびレギュラ再開モード
1	1	1	1	サポートされません
0	1	X	X	インジェクトオーバーサンプリング

25.3.32 デュアル ADC モード

2 つ以上の ADC を持つデバイスでは、デュアル ADC モードを使用することができます（*図 175*を参照）。

- ADC1 および ADC2 は、デュアルモードで一緒に使用できます（ADC1 がマスタ）。

デュアル ADC モードでは、変換の開始は ADCx_CCR レジスタのDUAL[4:0] ビットによって選択されたモードに応じて、ADCx マスタにより ADC スレーブに交互または同時にトリガされます。

次の 4 つのモードを備えています。

- インジェクト同時モード
- レギュラ同時モード
- インタリーブモード
- オルタネートトリガモード

これらのモードを次のように組み合わせて使用することも可能です。

- インジェクト同時モード + レギュラ同時モード
- レギュラ同時モード + オルタネートトリガモード
- インジェクト同時モード + インタリーブモード

デュアル ADC モードで（ADCx_CCR レジスタの DUAL[4:0] ビットがゼロ以外の場合）、ADCx_CFGR レジスタの CONT、AUTDLY、DISCEN、DISCNUM[2:0]、JDISCEN、JQM、JAUTO ビットは、マスタおよびスレーブ ADC の間で共有されます。スレーブ ADC のビットは、常にマスタ ADC の対応するビットと等しくなります。

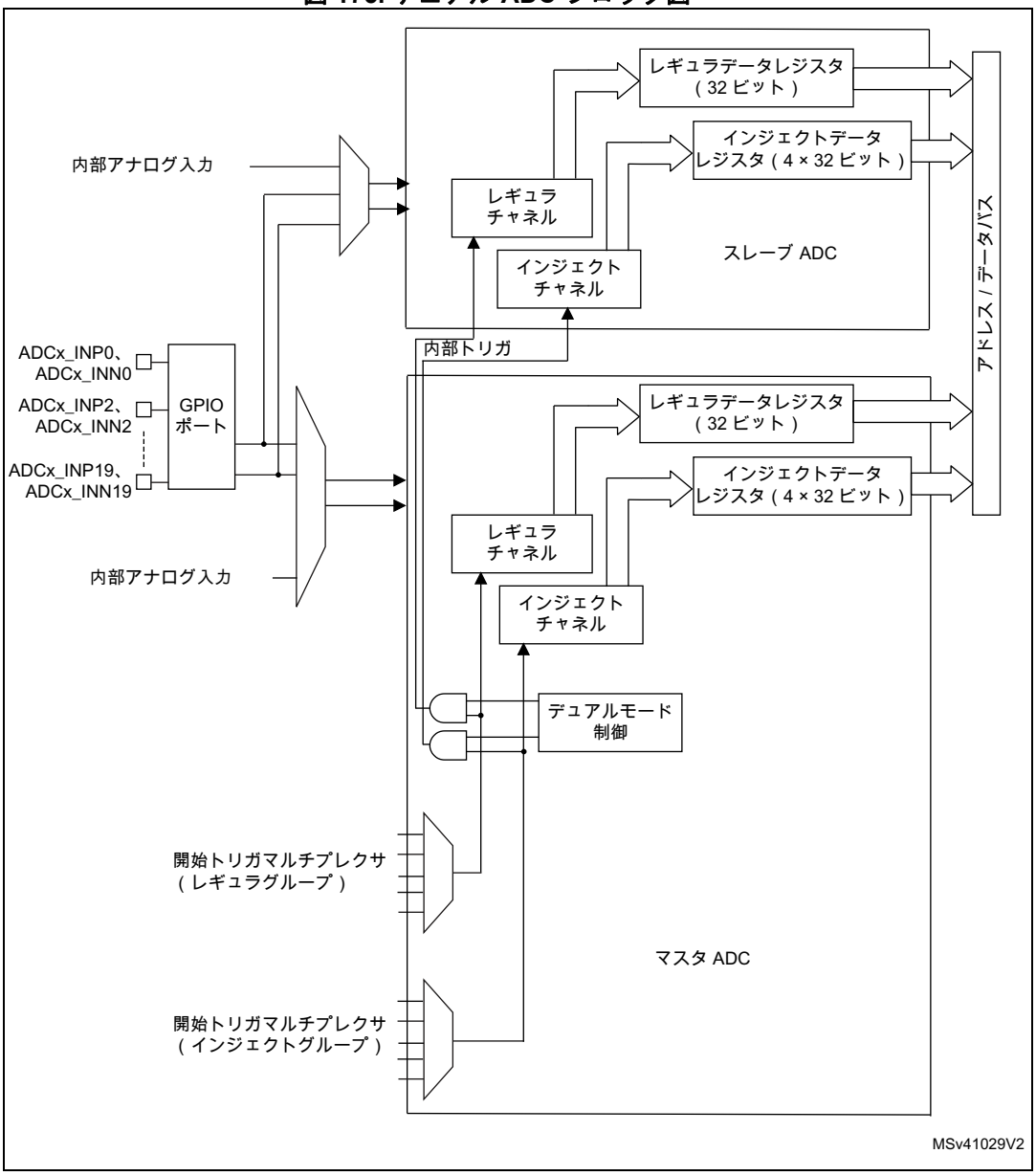
デュアルモードで変換を開始するには、ソフトウェアまたはハードウェアトリガやレギュラまたはインジェクトトリガを設定するために、マスタ ADC のEXTEN、EXTSEL、JEXTEN、JEXTSELビットのみをプログラムする必要があります（スレーブ ADC のEXTEN[1:0] および JEXTEN[1:0] ビットは無視されます）。

レギュラ同時モードまたはインタリーブモードでは、マスタ ADC のADSTART または ADSTPビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のADSTART または ADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

インジェクト同時モードまたはオルタネートトリガモードでは、マスタ ADC のJADSTART または JADSTP ビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のJADSTART または JADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

デュアル ADC モードでは、ADC 共通データレジスタ（ADCx_CDR）を読み出すことで、マスタおよびスレーブ ADC の変換データを並列で読み出すことができます。ステータスビットも、デュアルモードステータスレジスタ（ADCx_CSR）を読み出すことで、並列で読み出すことができます。

図 175. デュアル ADC ブロック図 (1)



1. スレーブ ADC にも外部トリガは存在しますが、この図には表示していません。
2. ADC 共通データレジスタ（ADCx_CDR）にはマスタとスレーブの両方の ADC レギュラ変換データが格納されます。

インジェクト同時モード

このモードは、ビット DUAL[4:0]=00101 をプログラムすることによって選択されます。

このモードは、インジェクトチャネルグループを変換します。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクス（ADCx_JSQR レジスタの JEXTSEL[4:0] ビットによって選択）から供給されます。

注： ***2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。***

同時モードでは、同じ長さのシーケンスを変換し、シーケンスの中では、マスタとスレーブの N 番目の変換は同じサンプリング時間で設定されなければなりません。

レギュラ変換であれば、1 つまたはすべての ADC で行うことができます。その場合、それらは互いに独立していてインジェクトイベントが発生すると中断されます。それら（レギュラ変換）はインジェクト変換グループ終了時に再開されます。

- マスタ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはマスタ ADCx_JDRy レジスタに格納され、JEOS 割り込みが生成されます（有効な場合）。
- スレーブ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはスレーブ ADCx_JDRy レジスタに格納され、JEOS 割り込みが生成されます（有効な場合）。
- マスタインジェクトシーケンスの長さがスレーブインジェクトシーケンスの長さと等しい場合（例：*図 176*）、ソフトウェアでは 2 つの JEOS 割り込みのうちの 1 つのみを有効にすることができ（例：マスタ JEOS）、（マスタ ADCx_JDRy とスレーブ ADCx_JDRy レジスタの）両方の変換データを読み出すことができます。

図 176. 4 チャネルのインジェクト同時モード：デュアル ADC モード



JDISCEN=1 の場合、インジェクトシーケンスの各同時変換にはインジェクトトリガイベントが発生する必要があります。

このモードは、AUTDLY モードと組み合わせることができます。

- 変換の同時インジェクトシーケンスの終了後、マスタとスレーブADCの両方の JEOS ビットがクリアされた場合（遅延フェーズ）にのみ、新しいインジェクトトリガイベントは受け入れられます。実行中のインジェクトシーケンスで発生したすべての新しいインジェクトトリガイベントと、これに関連する遅延フェーズは無視されます。
- マスタ ADC のレギュラ変換シーケンスが終了すると、マスタデータレジスタ（ADCx_DR）が読み出された場合にのみ、マスタ ADC の新しいレギュラトリガイベントが受け入れられます。実行中のレギュラシーケンスでマスタ ADC に対して発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。
  スレーブ ADC で発生するレギュラシーケンスでも同じ挙動を示します。

独立インジェクト変換に対応したレギュラ同時モード

このモードは、ビット DUAL[4:0]=00110 をプログラムすることによって選択されます。

このモードは、レギュラチャネルグループに対して行われます。外部トリガのソースは、マスタ ADC のレギュラグループマルチプレクス（ADCx_CFGR レジスタの EXTSEL[4:0] ビットによって選択）から供給されます。同時トリガは、スレーブ ADC に供給されます。

このモードでは、独立インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションリクエストにより、現在の同時変換がアボートされます。これは、インジェクト変換の完了後に再開されます。

注： ***2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。***

レギュラ同時モードでは、同じ長さのシーケンスを変換し、シーケンスの中では、マスタとスレーブの N 番目の変換は同じサンプリング時間で設定されなければなりません。

データを読み出すタイミングは、割り込みによってソフトウェアに通知されます。

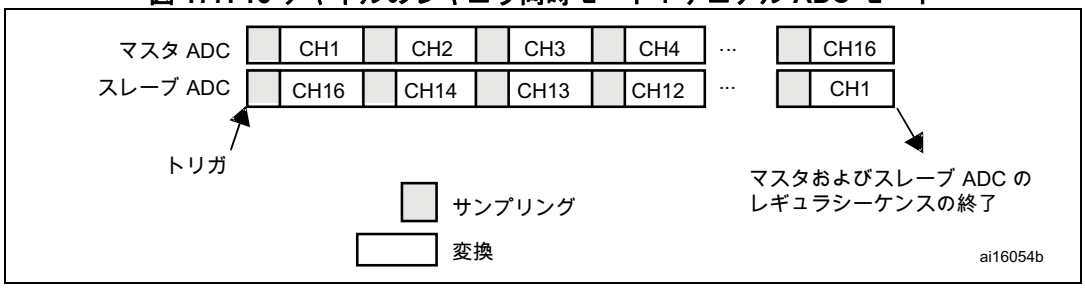
- マスタ ADC での各変換イベント（EOS）の終了時、マスタ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADCx_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOS）の終了時、スレーブ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADCx_DR を読み出すことができます。
- マスタレギュラシーケンスの長さがスレーブレギュラシーケンスの長さと等しい場合（例：*図 177*）、ソフトウェアでは 2 つの EOC 割り込みのうちの 1 つのみを有効にすることができ（例：マスタ EOC）、共通データレジスタ（ADCx_CDR）から両方の変換データを読み出すことができます。

DMA を使用してレギュラデータを読み出すことができます。次の 2 つの方法が可能です。

- 2 つの DMA チャネルを使用します（1つはマスタ用、もう 1つはスレーブ用）。この場合、ビット DAMDF[1:0] をクリア状態に保つ必要があります。
  - マスタから ADCx_DR を読み出すために、DMA のマスタ ADC チャネルを設定します。DMA リクエストは、マスタ ADC の各 EOC イベントで生成されます。
  - スレーブから ADCx_DR を読み出すために、DMA のスレーブ ADC チャネルを設定します。DMA リクエストは、スレーブ ADC の各 EOC イベントで生成されます。
- デュアル ADC モードデータフォーマットの DAMDF[1:0] ビットを設定し、他の用途のために 1 つのDMA チャネルを残します。
  - DAMDF[1:0]=0b10 または 0b11 を設定します（分解能による）。
  - DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADCx_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
  - マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ ADC の変換データを ADCx_CDR 32 ビットレジスタの上位ハーフワードで使用でき、またマスタ ADC の変換データを ADCx_CCR レジスタの下位ハーフワードで使用できます。
  - EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

注： ***DAMDF[1:0]=0b10 または 0b11の場合には、ユーザはマスタのシーケンスとスレーブのシーケンスに同じ変換数をプログラムする必要があります。そうしないと、残りの変換では DMA リクエストは生成されません。***

図 177. 16 チャネルのレギュラ同時モード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換でレギュラトリガイベントが発生する必要があります（「n」は DISCNUM で定義される）。

このモードは、AUTDLY モードと組み合わせることができます。

- シーケンスの同時変換の終了後、シーケンスの次の変換は、共通データレジスタ ADCx_CDR（またはマスタ ADC のレギュラデータレジスタ）が読み出された場合（遅延フェーズ）にのみ開始されます。
- 変換の同時レギュラシーケンスの終了後、新しいレギュラトリガイベントは共通データレジスタ（ADCx_CDR）が読み出された場合（遅延フェーズ）にのみ受け入れられます。実行中のレギュラシーケンスで発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。

AUTDLY モードと組み合わせたレギュラ同時モードでデータを処理するために、DMA を使用することができます（このとき、マルチ DMA モードが使われるものとします）。DAMDF ビットは 0b10 または 0b11 にセットする必要があります。

レギュラ同時モードを AUTDLY モードと組み合わせる場合、ユーザは次の内容を確認する必要があります。

- マスタのシーケンスの変換数が、スレーブの変換数と同じであること。
- シーケンスが同時変換されるたび、スレーブ ADC の変換の長さがマスタ ADC の変換の長さよりも短いこと。シーケンスの長さは、変換するチャネルの数および各チャネルのサンプリングタイムと分解能に依存する点に注意してください。

注： ***このレギュラ同時モードと AUTDLY モードの組み合わせは、レギュラチャネルのみがプログラムされている場合に制限されます。この組み合わせモードでインジェクトチャネルをプログラムすることは禁止されています。***

独立インジェクト変換に対応したインタリーブモード

このモードは、ビット DUAL[4:0]=00111 をプログラムすることによって選択されます。

このモードは、レギュラグループ（通常は 1 つのチャネル）でのみ開始できます。外部トリガソースは、マスタ ADC のレギュラチャネルマルチプレクスから供給されます。

外部トリガが発生した後、

- マスタ ADC はすぐに開始します。
- スレーブ ADC は、マスタ ADC のサンプリングフェーズが完了した後で、数 ADC クロックサイクルの遅延後に開始します。

インタリーブモードにおける 2 つの変換間の最小遅延は ADCx_CCR レジスタの DELAY ビットで設定します。遅延のカウントは、マスタ変換のサンプリングフェーズ終了後に開始します。これにより、ADC は相補 ADC が依然として入力をサンプリングしている場合、変換を開始できません（与えられた時間に 1 つの ADC のみ入力信号をサンプリングすることができます）。

- DELAY に設定可能な最小値は 1 です。これにより、マスタ ADC サンプリングフェーズのアナログスイッチを開いてスレーブ ADC サンプリングフェーズのアナログスイッチを閉じるまでに、最低 1 つのサイクル時間を確保できます。
- 最大 DELAY 値は選択した分解能に対するサイクル数と同じです。ただし、ほかの ADC が依然として入力をサンプリングしているときに ADC が変換を開始しないよう、ユーザはこの遅延を適切に計算する必要があります。

マスタとスレーブの両方の ADC で CONT ビットがセットされている場合、両方の ADC の選択されたレギュラチャネルが連続的に変換されます。

データを読み出すタイミングは、割り込みによってソフトウェアに通知されます。

- マスタ ADC での各変換イベント（EOS）の終了時、マスタ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADCx_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOS）の終了時、スレーブ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADCx_DR を読み出すことができます。

注： ***スレーブの EOC 割り込みのみを有効にして共通データレジスタ（ADCx_CDR）を読み出すことができます。ただしこの場合、ユーザは変換時間が適合しており、それによって新しいマスタ変換をリスタートする前に、シーケンス内では常にマスタ変換に続いてスレーブ変換が実行されることを確認する必要があります。***

DMA で転送されたレギュラデータを持つこともできます。この場合、各 ADC の個別の DMA リクエストを使用できず、次のように MDMA モードの使用が必須になります。

- DAMDF[1:0]=0b10 または 0b11 を設定します（分解能による）。
- DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADCx_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
- マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ ADC の変換データを ADCx_CDR 32 ビットレジスタの上位ハーフワードで使用でき、またマスタ ADC の変換データを ADCx_CCR レジスタの下位ハーフワードで使用できます。
- EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

図 178. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード

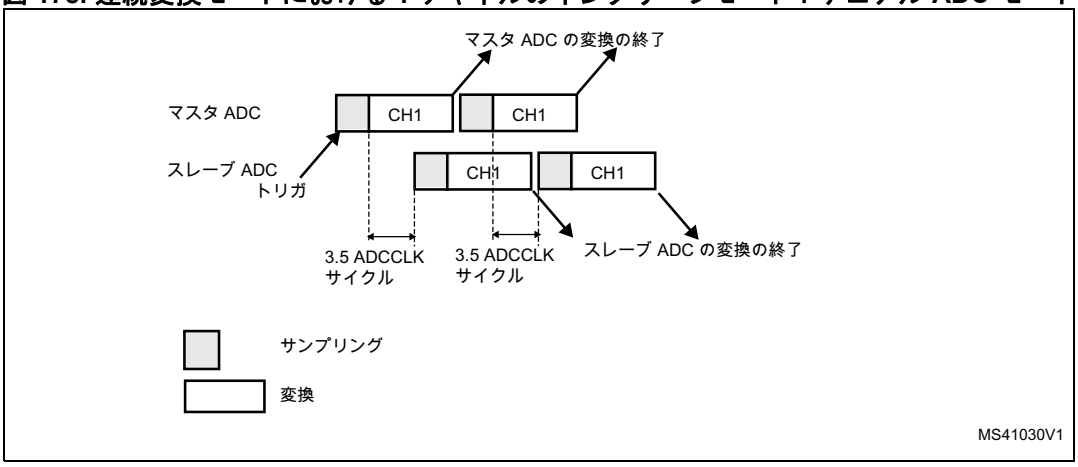
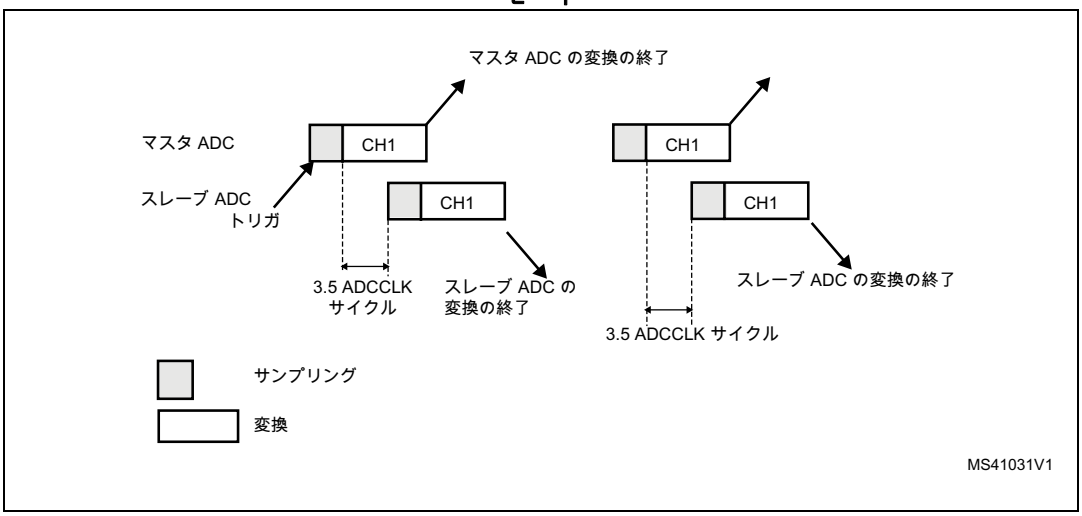


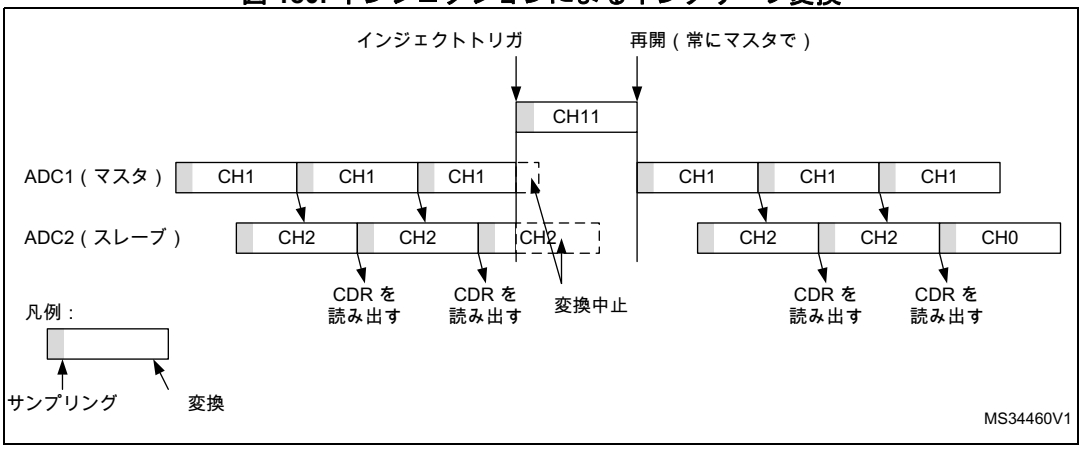
図 179. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換（「n」は DISCNUM で定義される）でレギュラトリガイベントが発生する必要があります。

このモードでは、インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションが完了すると、マスタとスレーブの両方のレギュラ変換がアボートされ、マスタからシーケンスが再開されます（以下の*図 180* を参照）。

図 180. インジェクションによるインタリーブ変換



オルタネートトリガモード

このモードは、ビット DUAL[4:0]=01001 をプログラムすることによって選択されます。

このモードは、インジェクトグループでのみ開始できます。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクスです。

このモードは、ハードウェアトリガの選択時にのみ使用できます。JEXTEN を 0x0 にセットすることはできません。

インジェクト不連続モードは無効です（両方の ADC で JDISCEN=0）。

1. 最初のトリガが発生すると、グループ内のすべてのインジェクトされたマスタ ADC チャネルが変換されます。
2. 2 番目のトリガが発生すると、グループ内のすべてのインジェクトされたスレーブ ADC チャネルが変換されます。
3. 以下同様です。

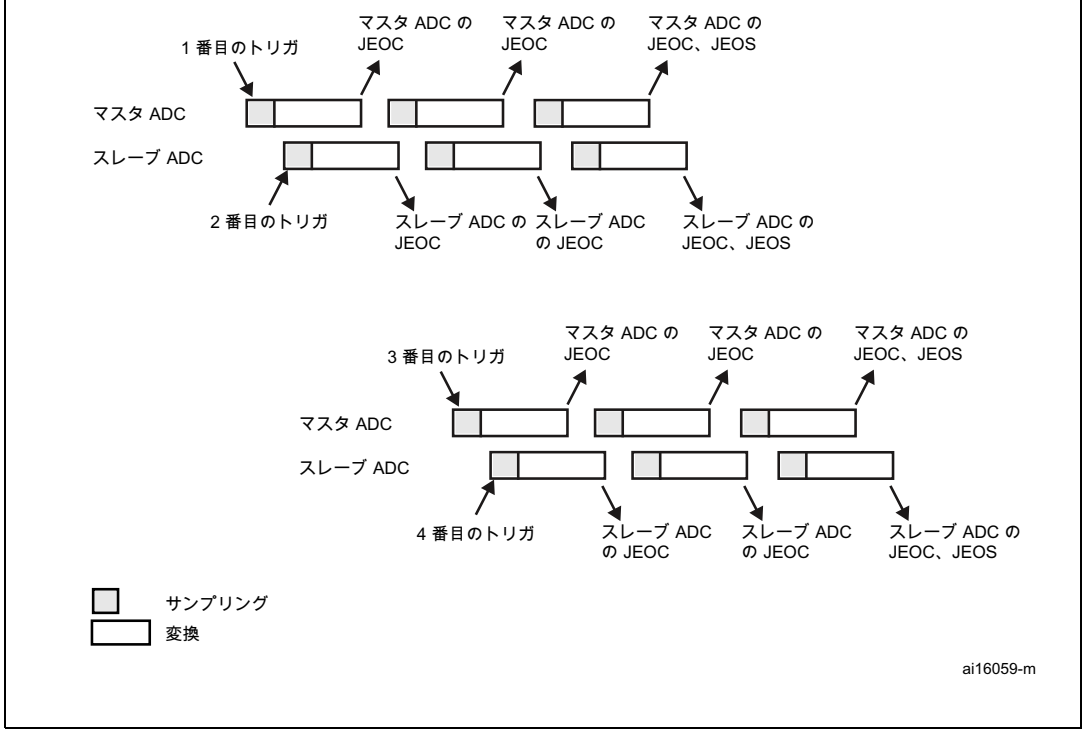
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

JEOC 割り込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内すべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、グループ内のマスタ ADC のインジェクトチャネルを変換することによって、オルタネートトリガプロセスがリスタートします。

図 181. オルタネートトリガ：各 ADC のインジェクトグループ



注： ***レギュラ変換であれば1 つまたはすべての ADC で有効にすることができます。この場合、レギュラ変換は互いに独立しています。ADC がインジェクト変換を行う必要があるとき、レギュラ変換は中断されます。インジェクト変換が終了すると(レギュラ変換が)再開されます。***

2 つのトリガイベント間の間隔は 1 ADC クロック期間と同等か長い必要があります。同じ ADC で変換を開始する 2 つのトリガイベント間の最小間隔はシングル ADC モードと同じです。

インジェクト不連続モードは有効です（両方の ADC で JDISCEN=1）。

マスタとスレーブの両方の ADC に対してインジェクト不連続モードが有効な場合、

- 最初のトリガが発生すると、マスタ ADC の最初のインジェクトチャネルが変換されます。
- 2 番目のトリガが発生すると、スレーブ ADC の最初のインジェクトチャネルが変換されます。
- 以下同様です。

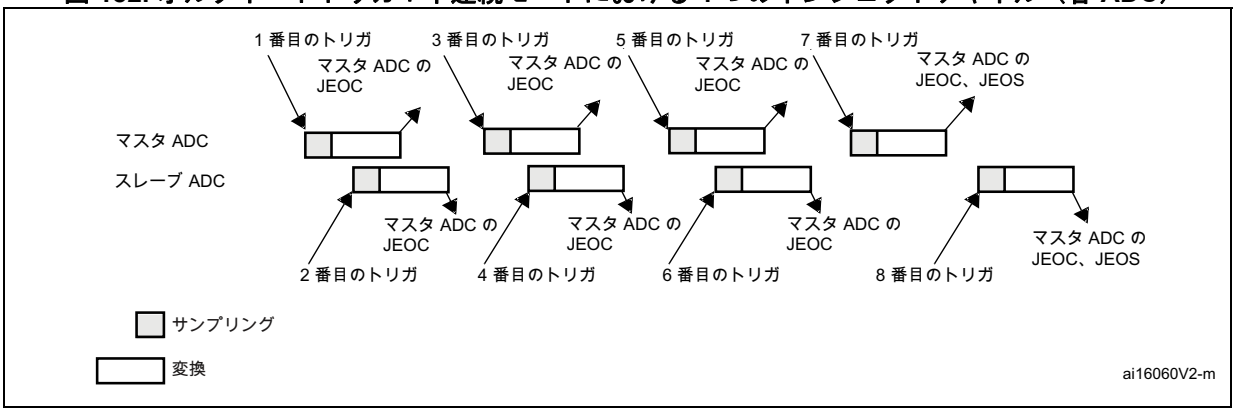
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

JEOC 割り込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内のすべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、オルタネートトリガプロセスがリスタートします。

図 182. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）



レギュラ／インジェクト同時モードの組み合わせ

このモードは、ビット DUAL[4:0]=00001 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループの同時変換を開始することができます。

注： ***シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が 2 つのシーケンスの変換時間が長い方より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC が再度開始することがあります。***

レギュラ同時 + オルタネートトリガモードの組み合わせ

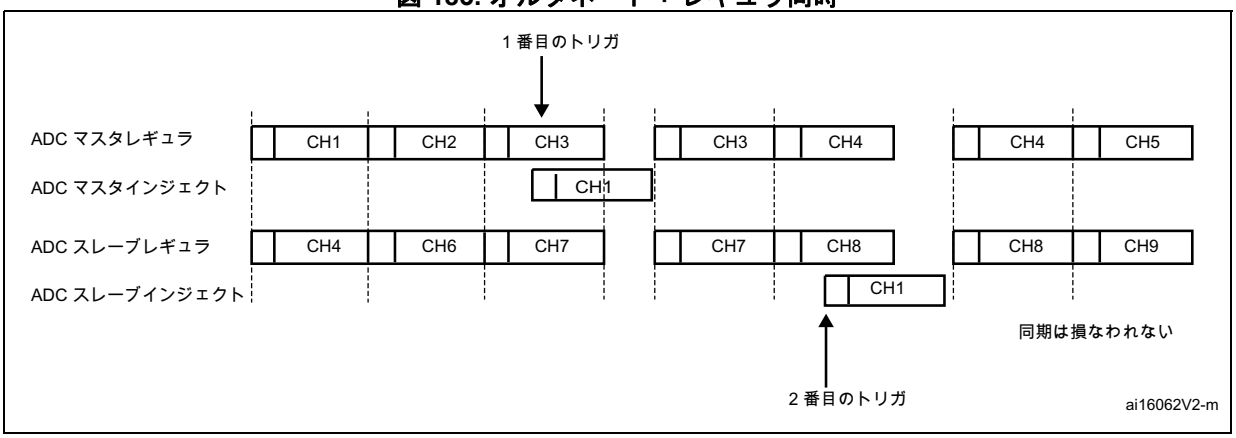
このモードは、ビット DUAL[4:0]=00010 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループのオルタネートトリガ変換を開始することができます。*図 183*は、同時レギュラ変換に割り込むオルタネートトリガの動作を示します。

インジェクトオルタネート変換は、インジェクトイベント後、ただちに開始されます。すでにレギュラ変換が実行中の場合、インジェクト変換後の同期を確保するために、すべての（マスタ／スレーブ）ADC のレギュラ変換は停止し、インジェクト変換の終了と同期して再開されます。

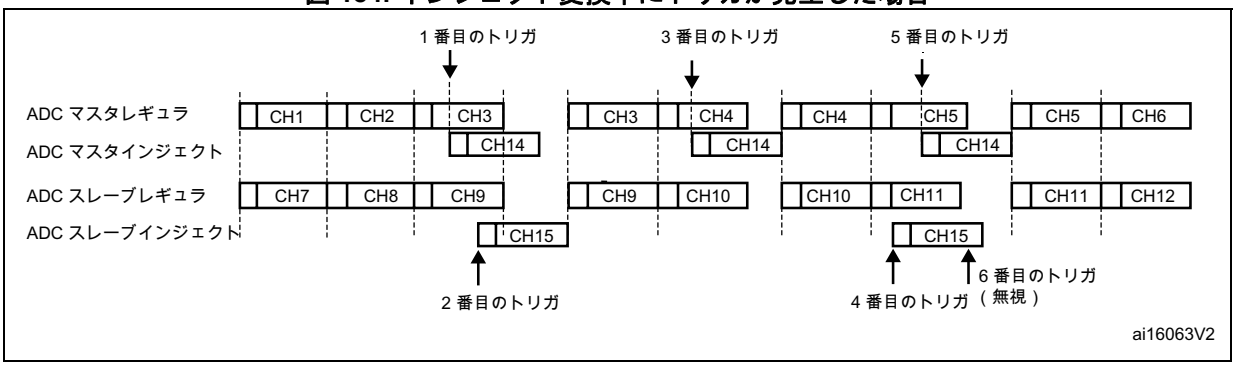
注： ***シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が 2 つのシーケンスの変換時間が長い方より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。***

図 183. オルタネート + レギュラ同時



レギュラ変換に割り込んだインジェクト変換の最中にトリガが発生した場合はオルタネートトリガが処理されます。*図 184*に、この場合の挙動を示します（関連付けられたオルタネート変換は完了していないため、6 番目のトリガは無視されます）。

図 184. インジェクト変換中にトリガが発生した場合



インジェクト同時モードとインタリーブの組み合わせ

このモードは、ビット DUAL[4:0]=00011 をプログラムすることによって選択されます。

インタリーブ変換に同時インジェクトイベントでの割り込みが可能です。

この場合、インタリーブ変換は即座に割り込まれ、同時インジェクト変換が開始します。インジェクトシーケンス終了時に、インタリーブ変換が再開されます。インタリーブレギュラ変換が再開すると、実行される最初のレギュラ変換は常にマスタのものです。*図 185*、*図 186* および*図 187* には、例を使用して動作を示しています。

注意： ***このモードでは、1 回の読み出しアクセスでレギュラデータを読み出すために共通データレジスタを使用する必要があります。逆に、マスタ - スレーブのデータコヒーレンシは保証されません。***

図 185. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0

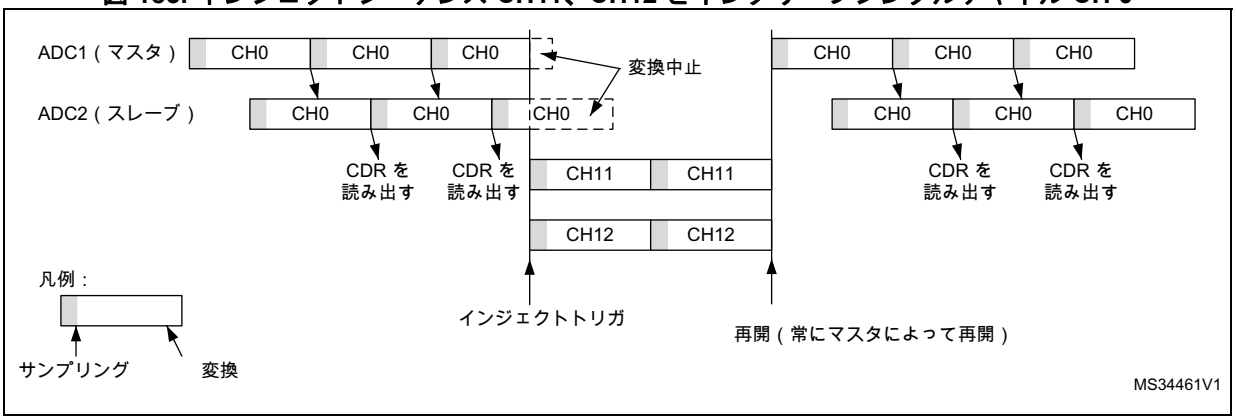


図 186. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割り込み優先

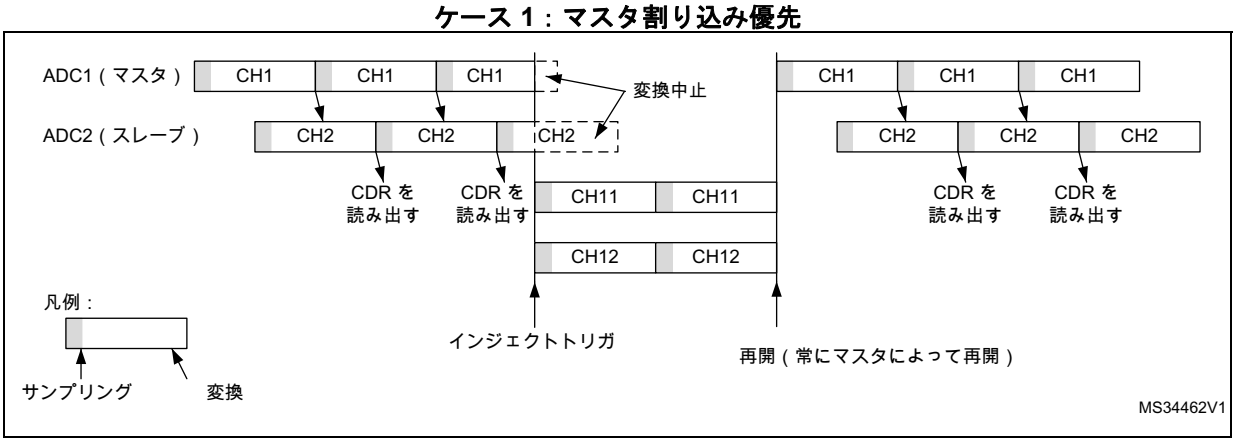
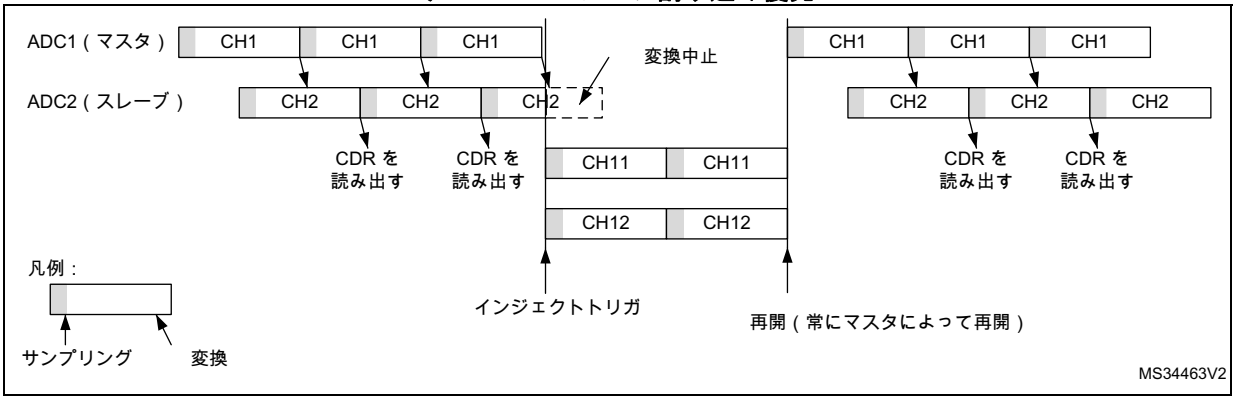


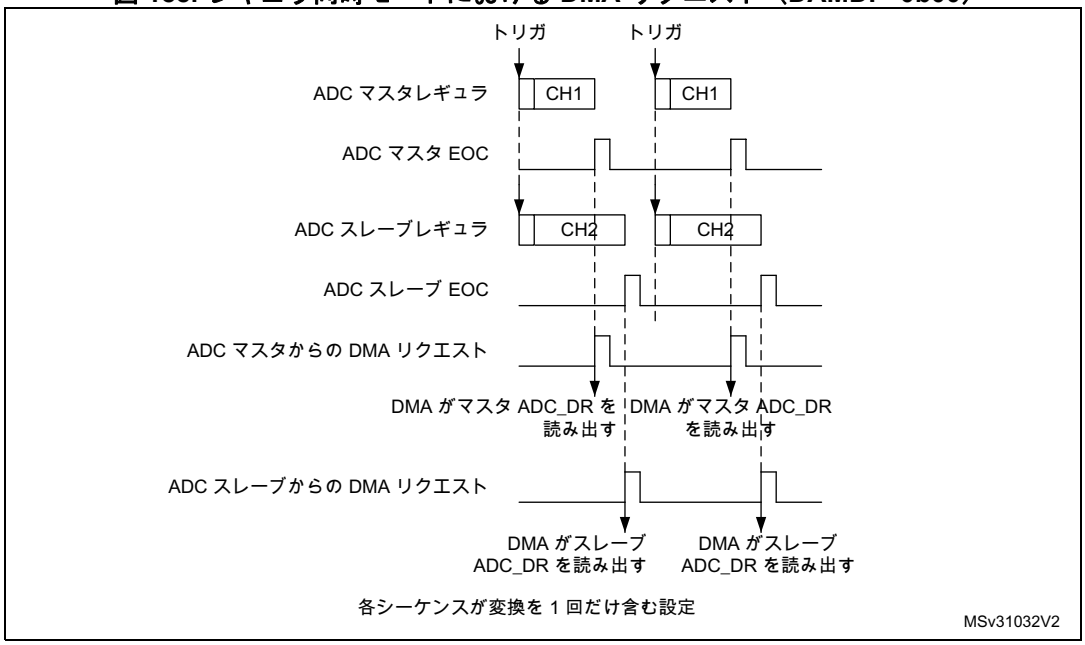
図 187. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割り込み優先



デュアル ADC モードにおける DMA リクエスト

すべてのデュアル ADC モードでは、シングルモードの場合と同様、データを転送するために 2 つの DMA チャネル（マスタ用 1 つとスレーブ用 1 つ）を使用することができます（*図 188：レギュラ同時モードにおける DMA リクエスト（DAMDF=0b00）*を参照）。

図 188. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b00）



レギュラ同時モードおよびインタリーブモードでは、1 つの DMA チャネルを確保し、シングル DMA チャネルを使用して両方のデータを転送することもできます。そのためには、ADCx_CCR レジスタで DAMDF ビットを設定する必要があります。

- **DAMDF=0b10、32 ビットフォーマット**：マスタとスレーブいずれかの EOC イベントが発生するたびに、代わりに DMA リクエストが 1 つ生成されます。そのとき、データ項目は代わりに ADCx_CDR2 32 ビットレジスタにあります。このモードは、分解能が 16 ビットを超えている場合に、インタリーブモードおよびレギュラ同時モードで使用されます。**例**：

  インタリーブデュアルモード: 新しい 32 ビットデータが使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADCx_CDR2[31:0] = MST_ADC_DR[31:0]

  2 番目の DMA リクエスト：ADCx_CDR2[31:0] = SLV_ADC_DR[31:0]

- **DAMDF=0b10、16 ビットフォーマット**：マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、2 つのデータアイテムが使用可能です。また、32 ビットレジスタ ADCx_CDR には、2 つの ADC で変換されたデータアイテムを表す 2 つのハーフワードが含まれます。スレーブ ADC データは上位ハーフワードを使用し、マスタ ADC データは下位ハーフワードを使用します。
  このモードは、分解能が 10 ビットから 16 ビットの範囲の場合に、インタリーブモードおよびレギュラ同時モードで使用されます。マスタコンバータまたはスレーブコンバータの 16 ビットを超えるすべての値は、LSB側 16 ビットに切り捨てられます。

例：

インタリーブデュアルモード：2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

最初の DMA リクエスト：ADCx_CDR[31:0] = SLV_ADCx_DR[15:0] | MST_ADCx_DR[15:0]

2 番目の DMA リクエスト：ADCx_CDR[31:0] = SLV_ADCx_DR[15:0] | MST_ADCx_DR[15:0]

図 189. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b10）

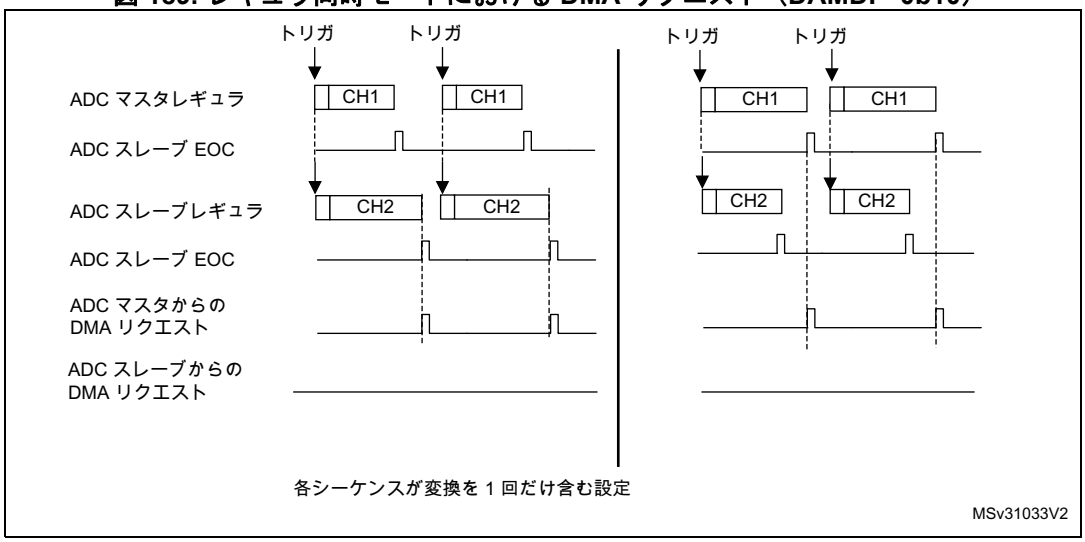
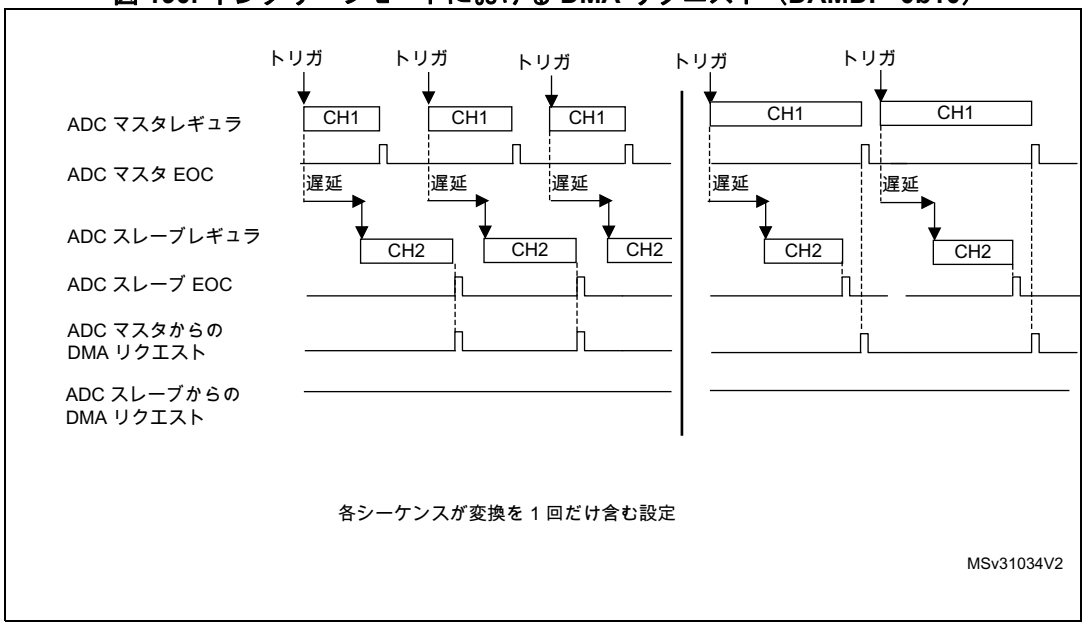


図 190. インタリーブモードにおける DMA リクエスト（DAMDF=0b10）



注： ***マルチ ADC モードを使用する場合は、マスタおよびスレーブ変換時間を適切に設定するよう注意する必要があります。これにより、新しい変換が使用可能になる前に DMA リクエストを生成して両方のデータ（マスタ + スレーブ）の読み出しを処理できます。***

- **DAMDF=0b11**：このモードは DAMDF=0b10 に類似しています。唯一の違いは、DMA リクエストごとに（データ項目は 2 つ使用可能）、2 つの ADC 変換データを表す 2 バイトがハーフワードとして転送されます。

  このモードは、結果が 8 ビットである場合に、インタリーブモードおよびレギュラ同時モードで使用されます。新しい 8 ビット値が 4 つある場合に、DMA リクエストが 1 つ発行されます。

  例：

  インタリーブデュアルモード：4 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます（t0、t1、... は連続するサンプリングインスタントに当たります）。

  最初の DMA リクエスト：

  ADCx_CDR[7:0] = MST_ADC_DR[7:0]$_{t0}$

  ADCx_CDR[15:8] = SLV_ADC_DR[7:0]$_{t0}$

  ADCx_CDR[23:16] = MST_ADC_DR[7:0]$_{t1}$

  ADCx_CDR[31:24] = SLV_ADC_DR[7:0]$_{t1}$

  2 番目の DMA リクエスト：

  ADCx_CDR[7:0] = MST_ADC_DR[7:0]$_{t2}$

  ADCx_CDR[15:8] = SLV_ADC_DR[7:0]$_{t2}$

  ADCx_CDR[23:16] = MST_ADC_DR[7:0]$_{t3}$

  ADCx_CDR[31:24] = SLV_ADC_DR[7:0]$_{t3}$

オーバーラン検出

デュアル ADC モード（DUAL[4:0] が b00000 以外の場合）で、ADC のうちの 1 つにオーバーランが検出された場合、RAM に転送されたすべてのデータが有効であることを保証するため DMA リクエストの発行を停止します（この動作は DAMDF の設定にかかわらず発生します）。ある ADC のデータレジスタに有効なデータが格納されていることによりその ADC に対応する EOC ビットがセットされたままになることがあります。

マルチ ADC モード選択時の DMA ワンショットモード／DMA サーキュラモード

DAMDF モードが選択されている場合（0b10 または 0b11）、*セクション：DMA を使用した変換の管理*で説明しているように、DMA ワンショットモード／サーキュラモードのどちらかを選択するために、マスタ ADC の ADCx_CCR レジスタのビット DMNGT[1:0]=0b10 も設定する必要があります。

デュアル ADC モードにおける変換の停止

デュアル ADC モードにおける ADC の変換を両方とも停止するには、マスタ ADC の制御ビットに ADSTP/JADSTP をセットする必要があります。デュアル ADC モードには、スレーブ ADC のその他の ADSTP 制御ビットの効果はありません。

両方の ADC が実質的に無効になると、マスタおよびスレーブ ADC の両方のADSTART/JADSTART ビット がハードウェアによってクリアされます。

デュアル ADC モードインタリーブモードにおける DFSDM モード

デュアル ADC インタリーブモードでは、ADC 変換結果は、デルタシグマモジュレータのデジタルフィルタ（DFSDM）に直接転送できます。

このモードは、マスタ ADC の ADCx_CFGR レジスタのビット DMNGT[1:0] = 0b10 をセットすることによって有効になります。

ADC は、マスタおよびスレーブコンバータから DFSDM のシングルチャネルにレギュレータデータの最下位 16 ビットを代わりに転送します。

データフォーマットは、16 ビット符号である必要があります。

ADCx_DR[31:16] = 0x0000

ADCx_DR[15] = 符号

ADCx_DR[14:0] = データ

どのコンバータでも、16 ビット符号付きフォーマットを超えるあらゆるデータは切り捨てられます。

デュアル ADC 同時モードにおける DFSDM モード

変換データは個別チャネルによって扱われるため、デュアル ADC 同時モードで DFSDM を使用する時には、DFSDMのデュアルモードは必要ありません。同じトリガソースでのシングルモードは、DFSDM インタフェースでの同時変換になります。

25.3.33 温度センサ

温度センサは、デバイスの周辺温度（T_A）を測定できます。

主な特徴

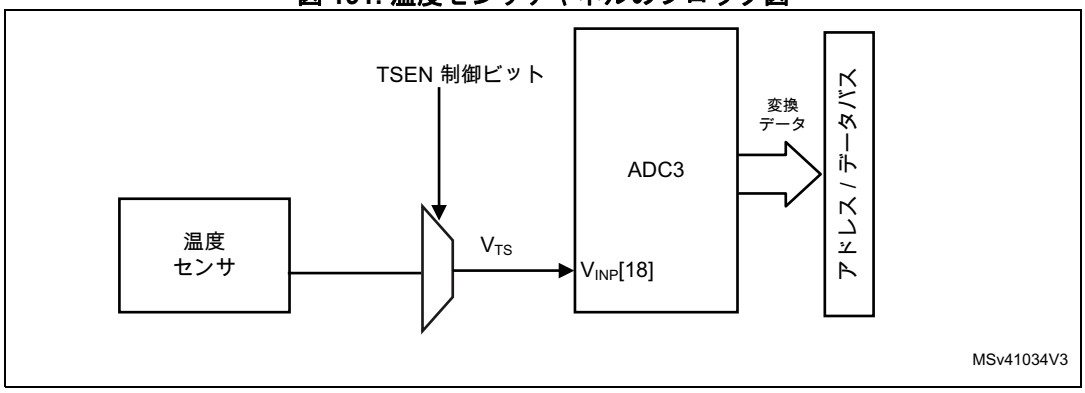
- サポートしている温度範囲：–40 ～ 125 ℃
- 精度：±2 °C

温度センサは、センサの出力電圧をデジタル値に変換する ADC3 V_{INP}[18] 入力チャネルに内部接続されています。温度センサのアナログピンのサンプリング時間は、製品データシートに記載された整定時間より大きい必要があります。

使用しないときには、センサをパワーダウンモードにすることができます。

図 191 に、温度センサのブロック図を示します。

図 191. 温度センサチャネルのブロック図



注： ***内部チャネル ADC3 V_{INP}[18]（温度センサ、V_{TS}）の変換を有効にするには、TSEN ビットをセットする必要があります。***

温度の読み出し

温度センサを使用するには：

1. ADC3 V_{INP}[18] 入力チャネルを選択します（適切なサンプリング時間を使用）。
2. 適切なサンプリング時間をプログラムします（デバイスデータシートの電気的特性のセクションを参照）。
3. ADC3_CCR レジスタの TSEN ビットをセットして、温度センサをパワーダウンモードからウェイクアップします。
4. ADC 変換を開始します。
5. ADC データレジスタの V_{TS} データを読み出します。
6. 次の式を使用して、実際の温度を計算します。

$$\text{温度}((°C)) = \frac{110\,°C - 30\,°C}{TS_CAL2 - TS_CAL1} \times (TS_DATA - TS_CAL1) + 30\,°C$$

ここで、

- TS_CAL2 は、110°Cで得られた温度センサの較正値です。
- TS_CAL1 は、30℃で得られた温度センサの較正値です。
- TS_DATA は、ADC によって変換された実際の温度センサの出力値です。

  TS_CAL1 および TS_CAL2 較正ポイントの詳細については、デバイスデータシートを参照してください。

注： ***センサがパワーダウンモードからウェイクアップして、正しいレベルで V_{TS} を出力できるようになるまでには時間がかかります（スタートアップ時間）。ADC にも起動後のスタートアップ時間があるので、遅延を最小にするには、ADEN ビットと TSEN ビットを同時にセットしてください。***

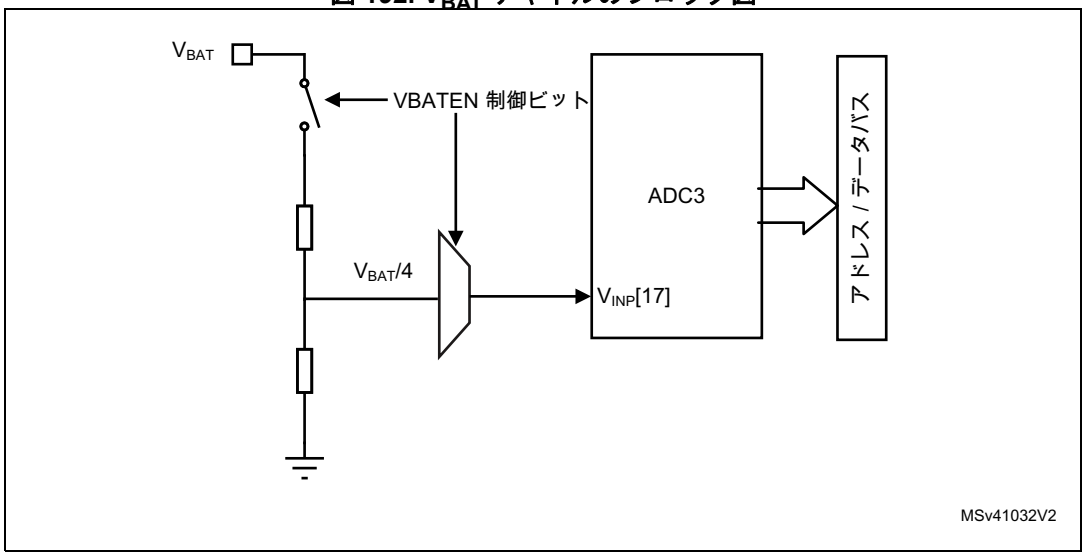
25.3.34 V_{BAT} 電源監視

ADC3_CCR レジスタの VBATEN ビットはバッテリ電圧に切り替えるために使用します。V_{BAT} 電圧は V_{DDA} より高くなることがあるので、ADC の正しい動作を確保するために、V_{BAT} ピンはブリッジ 4 分圧回路に内部接続されています。このブリッジは、VBATEN ビットがセットされると自動的に有効になり、V_{BAT}/4 を ADC3 V_{INP}[17] 入力チャネルに接続します。結果として、変換されたデジタル値は V_{BAT} 電圧の 1/4 です。不要なバッテリ消費を避けるには、ADC 変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

V_{BAT} /4 電圧の変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性を参照してください。

図 192 に、V_{BAT} センシング機能のブロック図を示します。

図 192. V_{BAT} チャネルのブロック図



注： ***内部チャネル ADC3 V_{INP}[17]（V_{BAT}/4）の変換を有効にするには、VBATEN ビットをセットする必要があります。***

25.3.35 内部電圧基準の監視

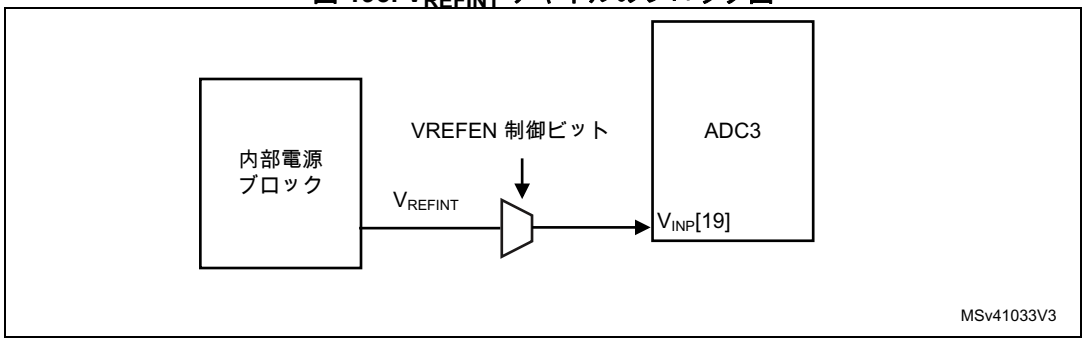
ADC V_{REF+} 電圧レベルを評価するための基準点を持つために、内部電圧基準を監視することができます。

内部電圧基準は、入力チャネル ADC3 V_{INP}[19] に内部的に接続されています。

このチャネルのサンプリング時間は、製品データシートに記載された安定時間より大きい必要があります。

図 192 に、V_{REFINT} センシング機能のブロック図を示します。

図 193. V_{REFINT} チャネルのブロック図



注： ***内部チャネル ADC3 V_{INP}[19]（V_{REFINT}）の変換を有効にするには、ADCx_CCR レジスタに VREFEN ビットをセットする必要があります。***

内部基準電圧を使用した実際の V_{DDA} 電圧の計算

マイクロコントローラに印加されるV_{DDA} 電源電圧は、ばらつきがあり、または正確にはわからないことがあります。内蔵の内部電圧基準（V_{REFINT}）と、製造プロセス時に V_{DDA} = 3.3 V で ADC によって得られた較正データを使用して、実際の V_{DDA} 電圧レベルを評価することができます。

デバイスに印加される実際の V_{DDA} 電圧は、次の式で求められます。

$$V_{DDA} = 3.3\ V \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。

供給に相対的な ADC 測定値から絶対電圧値への変換

ADC は、アナログ電源と変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。ほとんどのアプリケーションの使用事例では、この比を V_{DDA} に依存しない電圧に変換する必要があります。V_{DDA} が既知であり、ADC によって変換された値が右詰めされるアプリケーションでは、次の式を使用して、この絶対値を求めることができます。

$$V_{CHANNELx} = \frac{V_{DDA}}{FULL_SCALE} \times ADCx_DATA$$

V_{DDA} 値が既知ではないアプリケーションの場合、内部電圧基準を使用する必要があり、V_{DDA} を *セクション：内部基準電圧を使用した実際の V_{DDA} 電圧の計算* に記載されている式で置き換えることができ、次の式を使用できます。

$$V_{CHANNELx} = \frac{3.3\ V \times VREFINT_CAL \times ADCx_DATA}{VREFINT_DATA \times FULL_SCALE}$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- ADCx_DATA は、チャネル x で ADC によって測定された値です（右詰め）。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。
- FULL_SCALE は、ADC 出力の最大デジタル値です。たとえば、16 ビット分解能では 2^{16} - 1 = 65535 になり、8 ビット分解能では 2^8 - 1 = 255 になります。

注： ***ADC 測定が 16 ビット右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。***

25.4 ADC 割り込み

各 ADC に、割り込みを生成することができます。

- ADC パワーアップ後、ADC の準備ができたとき（ADRDY フラグ）
- レギュラグループの任意の変換終了時（EOC フラグ）
- レギュラグループの任意の変換シーケンス終了時（EOS フラグ）
- インジェクトグループの任意の変換終了時（JEOC フラグ）
- インジェクトグループの任意の変換シーケンス終了時（JEOS フラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、および AWD3 フラグ）
- サンプリングフェーズ終了の発生時（EOSMP フラグ）
- データオーバーランの発生時（OVR フラグ）
- インジェクトシーケンスのコンテキストキューのオーバーフロー時（JQOVF フラグ）

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 202. ADC ごとの ADC 割り込み

割り込みイベント	イベントフラグ	イネーブル制御ビット
ADC レディ	ADRDY	ADRDYIE
レギュラグループの変換終了	EOC	EOCIE
レギュラグループの変換シーケンスの終了	EOS	EOSIE
インジェクトグループの変換終了	JEOC	JEOCIE
インジェクトグループの変換シーケンスの終了	JEOS	JEOSIE
アナログウォッチドッグ 1 のステータスビットのセット	AWD1	AWD1IE
アナログウォッチドッグ 2 のステータスビットのセット	AWD2	AWD2IE
アナログウォッチドッグ 3 のステータスビットのセット	AWD3	AWD3IE
サンプリングフェーズの終了	EOSMP	EOSMPIE
オーバーラン	OVR	OVRIE
インジェクトコンテキストキューのオーバーフロー	JQOVF	JQOVFIE

25.5 ADC レジスタ（ADC ごと）

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

25.5.1 ADC 割り込みおよびステータスレジスタ（ADCx_ISR、x=1..3）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
					r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	rc_w1	r_w1	r_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF**：インジェクトコンテキストキューのオーバーフロー

このビットは、コンテキストのインジェクトキューのオーバーフローが発生したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。詳細については、*セクション 25.3.22：インジェクト変換のコンテキストのキュー*を参照してください。

0：インジェクトコンテキストキューのオーバーフローは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトコンテキストキューのオーバーフローが発生しました。

ビット 9 **AWD3**：アナログウォッチドッグ 3 フラグ

このビットは、変換された電圧が、ADCx_TR3 レジスタのフィールド LT3[7:0] および HT3[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 3 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 3 イベントが発生しました。

ビット 8 **AWD2**：アナログウォッチドッグ 2 フラグ

このビットは、変換された電圧が、ADCx_TR2 レジスタのフィールド LT2[7:0] および HT2[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 2 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 2 イベントが発生しました。

ビット 7 **AWD1**：アナログウォッチドッグ 1 フラグ

このビットは、変換された電圧が、ADCx_TR1 レジスタのフィールド LT1[11:0] および HT1[11:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 1 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 1 イベントが発生しました。

ビット 6 **JEOS**：インジェクトチャネルのシーケンス終了フラグ

このビットは、グループ内のすべてのインジェクトチャネル変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：インジェクト変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクト変換シーケンスが完了しました。

ビット 5 **JEOC**：インジェクトチャネルの変換終了フラグ

このビットは、チャネルの各インジェクト変換の終了時に、新しいデータが対応する ADCx_JDRy レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または対応する ADCx_JDRy レジスタを読み出すことによってクリアされます。

0：インジェクトチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトチャネル変換が完了しました。

ビット 4 **OVR**：ADC オーバーラン

このビットは、レギュラチャネルでオーバーランが発生したときにハードウェアによってセットされ、EOC フラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで 1 を書き込むことによってクリアされます。

0：オーバーランは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット 3 **EOS**：レギュラシーケンス終了フラグ

このビットは、レギュラチャネルシーケンスの変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：レギュラ変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラ変換シーケンスが完了しました。

ビット 2 **EOC**：変換終了フラグ

このビットは、チャネルの各レギュラ変換の終了時に、新しいデータが ADCx_DR レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または ADCx_DR レジスタを読み出すことによってクリアされます。

0：レギュラチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラチャネル変換が完了しました。

ビット 1 **EOSMP**：サンプリング終了フラグ

このビットは、任意のチャネル（レギュラチャネルのみ）の変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズは終了しました。

ビット 0 **ADRDY**：ADC レディ

このビットは、ADC が有効にされた後（ビット ADEN=1）、ADC が変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで 1 を書き込むことによってクリアされます。

0：ADC はまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：ADC は変換を開始する準備ができました。

25.5.2 ADC 割り込み有効レジスタ（ADCx_IER、x=1..3）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVFIE**：インジェクトコンテキストキューのオーバーフロー割り込みイネーブル

このビットは、インジェクトコンテキストキューのオーバーフローの割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトコンテキストキューのオーバーフロー割り込みは無効です。

1：インジェクトコンテキストキューのオーバーフロー割り込みは有効です。JQOVF ビットがセットされると、割り込みが生成されます。

注： ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **AWD3IE**：アナログウォッチドッグ 3 割り込みイネーブル

このビットは、アナログウォッチドッグ 2 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 3 割り込みは無効です。

1：アナログウォッチドッグ 3 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8 **AWD2IE**：アナログウォッチドッグ 2 割り込みイネーブル

このビットは、アナログウォッチドッグ 2 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 2 割り込みは無効です。

1：アナログウォッチドッグ 2 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 7 **AWD1IE**：アナログウォッチドッグ 1 割り込みイネーブル

このビットは、アナログウォッチドッグ 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 1 割り込みは無効です。

1：アナログウォッチドッグ 1 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 6 **JEOSIE**：インジェクト変換シーケンス終了割り込みイネーブル

このビットは、インジェクト変換シーケンス終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOS 割り込みは無効です。

1：JEOS 割り込みは有効です。JEOS ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 5 **JEOCIE**：インジェクト変換終了割り込みイネーブル

このビットは、インジェクト変換終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOC 割り込みは無効です。

1：JEOC 割り込みは有効です。JEOC ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、JADSTART=0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 4 **OVRIE**：オーバーラン割り込みイネーブル

このビットは、レギュラ変換のオーバーラン割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割り込みは無効です。

1：オーバーラン割り込みは有効です。OVR ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 3 **EOSIE**：レギュラ変換シーケンス終了割り込みイネーブル

このビットは、レギュラ変換シーケンス終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS 割り込みは無効です。

1：EOS 割り込みは有効です。EOS ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 2 **EOCIE**：レギュラ変換終了割り込みイネーブル

このビットは、レギュラ変換終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC 割り込みは無効です。

1：EOC 割り込みは有効です。EOC ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 1 **EOSMPIE**：レギュラ変換のサンプリング終了フラグ割り込みイネーブル

このビットは、レギュラ変換のサンプリングフェーズ終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP 割り込みは無効です。

1：EOSMP 割り込みは有効です。EOSMP ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 0 **ADRDYIE**：ADC レディ割り込みイネーブル

このビットは、ADC レディ割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY 割り込みは無効です。

1：ADRDY 割り込みは有効です。ADRDY ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

25.5.3 ADC 制御レジスタ（ADCx_CR、x=1..3）

アドレスオフセット：0x08

リセット値：0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	LINCALRDYW6	LINCALRDYW5	LINCALRDYW4	LINCALRDYW3	LINCALRDYW2	LINCALRDYW1	Res.	Res.	Res.	Res.	Res.	ADCALLIN
rs	rw	rw	rw	rw	rw	rw	rw	rw	rw						rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOST	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
							rw			rs	rs	rs	rs	rs	rs

ビット 31 ADCAL：ADC 較正

このビットは、ADC の較正を開始するためにソフトウェアによってセットされます。最初にADCALDIF ビットをプログラムして、この較正がシングルエンドまたは差動入力モードに適用されるかどうかを確認します。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：ADC を較正するには、1 を書き込みます。1 として読み出されたときには、較正が実行中であることを意味します。

注： ソフトウェアは、ADEN=0 のときだけ ADCAL をセットして、較正を起動することができます。

注： ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、ADCx_CALFACT を書き込むことによって較正係数を更新できます。

ビット 30 ADCALDIF：較正の差動モード

このビットは、較正のシングルエンド入力または差動入力モードを設定するために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、シングルエンド入力モードで較正が起動されます。

1：ADCAL を書き込むと、差動入力モードで較正が起動されます。

注： ソフトウェアは、ADC が無効で較正されていないときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。

ビット 29 DEEPPWD：ディープパワーダウンイネーブル

このビットは、ADC をディープパワーダウンモードにするために、ソフトウェアによってセット／クリアされます。

0：ADC のディープパワーダウンモードは無効です。

1：ADC のディープパワーダウンモードは有効です（デフォルトのリセット状態）。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。

ビット 28 ADVREGEN：ADC 電圧レギュレータイネーブル

これらのビットは、ADC 電圧レギュレータを有効にするために、ソフトウェアによってセットされます。

較正の起動や ADC の有効化などの操作を実行する前に、まずは ADC 電圧レギュレータを有効にし、ソフトウェアはレギュレータの起動時間まで待つ必要があります。

0：ADC 電圧レギュレータは無効です。

1：ADC 電圧レギュレータは有効です。

ADC 電圧レギュレータの有効化および無効化シーケンスの詳細については、*セクション 25.3.6：ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）*を参照してください。

ソフトウェアは、ADC が無効のときのみ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットフィールドをプログラムできます。

ビット 27 **LINCALRDYW6**：直線性較正レディワード 6

このコントロール／ステータスビットにより、6 番目の直線性較正係数の読み出し／書き込みが行えます。

直線性較正が完了すると、このビットがセットされます。ビットをクリアすると、ADC_CALFACT2 レジスタの LINCALFACT[29:0] に対して、直線性係数 6 の転送が開始されます。ADC_CALFACT2 レジスタの読出しが可能となると、そのビットはハードウェアによってリセットされます（ソフトウェアはクリアされるまでそのビットをポーリングする必要があります）。

*LINCALRDYW6 ビットがリセットされると、*ADC_CALFACT2 レジスタの LINCALFACT[29:0] *に対して、直線性係数 6 の新しい値を書き込むことができます。*ビットがセットされると、直線性係数 6 の更新が開始され、更新が終わると、そのビットがハードウェアによって有効にセットされます（ビットがセットされて書込みが有効であることが示されるまで、ソフトウェアはそのビットをポーリングする必要があります）。

注： ***ADC_CALFACT2[29:10] には 0 が格納されています。ADC_CALFACT2[9:0] は、直線性補正係数ビット [159:150] に対応しています。***

ソフトウェアは、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1 の各ビットが変更されていないときのみ、このビットをトグルできます。詳細は、25.3.8：較正（ADCAL、ADCALDIF、ADCALLIN、ADCx_CALFACT）を参照してください。

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、LINCALRDYWx を書き込むことによって直線性較正係数を更新できます。

ビット 26 **LINCALRDYW5**：直線性較正レディワード 5

LINCALRDYW6 の説明を参照してください。

注： ***ADC_CALFACT2[29:0] は、直線性補正係数ビット [149:120] に対応しています。***

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1 の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 25 **LINCALRDYW4**：直線性較正レディワード 4

LINCALRDYW6 の説明を参照してください。

注： ***ADC_CALFACT2[29:0] は、直線性補正係数ビット [119:90] に対応しています。***

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1 の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 24 **LINCALRDYW3**：直線性較正レディワード 3

LINCALRDYW6 の説明を参照してください。

注： ***ADC_CALFACT2[29:0] は、直線性補正係数ビット [89:60] に対応しています。***

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW2、および LINCALRDYW1 の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 23 **LINCALRDYW2**：直線性較正レディワード 2

LINCALRDYW6 の説明を参照してください。

注： ***ADC_CALFACT2[29:0] は、直線性補正係数ビット [59:30] に対応しています。***

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、および LINCALRDYW1 の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 22 **LINCALRDYW1**：直線性較正レディワード 1

LINCALRDYW6 の説明を参照してください。

注： ***ADC_CALFACT2[29:0] は、直線性補正係数ビット [29:0] に対応しています。***

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、および LINCALRDYW2 の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 21:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **ADCALLIN**：直線性較正

このビットは、直線性較正を有効にするために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、直線性較正なしで較正が起動されます。

1：ADCAL を書き込むと、直線性較正ありで較正が起動されます。

注： ***ソフトウェアは、ADC　が無効で較正されていないときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **BOOST**：ブーストモード制御

このビットは、ブーストモードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ブーストモードオフ。ADC クロックが 20 MHz 未満であるときに、クロック周波数を下げて電力を節約するために使用されます。

1：ブーストモードオン。ADC クロックが 20 MHz を超える場合に、使用する必要があります。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADCx_CCR レジスタ の DAMDF ビット が 0 以外）である場合、スレーブ ADC の BOOST ビットへの書き込みができなくなり、その内容はマスタ ADC のAUTDLY ビットと等しくなります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JADSTP**：ADC インジェクト変換停止コマンド

このビットは、実行中のインジェクト変換を停止および破棄するためにソフトウェアによってセットされます（JADSTP コマンド）。

変換が効果的に破棄され、ADC インジェクトシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいインジェクト変換の開始（JADSTART コマンド）を受け入れる準備ができます。

0：ADC インジェクト変換停止コマンドは実行中ではありません。

1：実行中の ADC インジェクト変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： ***ソフトウェアは、JADSTART=1 かつ ADDIS=0 である（ADC が有効であり、最終的にインジェクト変換が変換され、ADC を無効にする保留中のリクエストがない）ときだけ、JADSTP をセットできます。***

自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

ビット 4 **ADSTP**：ADC レギュラ変換停止コマンド

このビットは、実行中のレギュラ変換を停止および破棄するためにソフトウェアによってセットされます（ADSTP コマンド）。

変換が効果的に破棄され、ADC レギュラシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいレギュラ変換の開始（ADSTART コマンド）を受け入れる準備ができます。

0：ADC レギュラ変換停止コマンドは実行中ではありません。

1：実行中の ADC レギュラ変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： ***ソフトウェアは、ADSTART=1 かつ ADDIS=0 である（ADC が有効であり、最終的にレギュラ変換が変換され、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTP をセットできます。***

自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

デュアル ADC レギュラ同時モードとインタリーブモードでは、レギュラ変換を停止するためにマスタ ADC の ADSTP ビットを使用する必要があります。その他の ADSTP ビットはインアクティブです。

ビット 3 **JADSTART**：ADC インジェクト変換開始

このビットは、インジェクトチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット JEXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはインジェクトハードウェアトリガイベントが発生（ハードウェアトリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モードでは、ソフトウェアトリガが選択されたとき（JEXTSEL=0x0）：インジェクト変換シーケンス終了（JEOS）フラグのアサート時。
- すべての場合：JADSTP コマンドの実行後、JADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC インジェクト変換は実行中ではありません。

1：インジェクト変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にインジェクトチャネルが変換されることを意味します。

注： ***ソフトウェアは、ADEN=1 かつ ADDIS=0 である（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、JADSTART をセットできます。***

自動インジェクションモード（JAUTO=1）では、設定ビット ADSTART を使用してレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 2 **ADSTART**：ADC レギュラ変換開始

このビットは、レギュラチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット EXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはレギュラハードウェアトリガイベントが発生したときに（ハードウェアトリガ設定）開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モード（CONT=0、DISCEN=0）では、ソフトウェアトリガが選択されたとき（EXTEN=0x0）：レギュラ変換シーケンス終了（EOS）フラグのアサート時。
- 不連続変換モード（CONT=0、DISCEN=1）では、ソフトウェアトリガが選択されたとき（EXTEN=0x0）：変換終了（EOS）フラグ時。
- これ以外のすべての場合：ADSTP コマンドの実行後、ADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC レギュラ変換は実行中ではありません。

1：レギュラ変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にレギュラチャネルが変換されることを意味します。

注： ***ソフトウェアは、ADEN=1 かつ ADDIS=0（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTART をセットできます。***

自動インジェクションモード（JAUTO=1）では、設定ビット ADSTART を使用してレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 1 **ADDIS**：ADC 無効化コマンド

このビットは、ADC を無効にして（ADDIS コマンド）、パワーダウン状態（OFF 状態）にするためにソフトウェアによってセットされます。

ADC が効果的に無効化されると、ハードウェアによってクリアされます（ADEN もこの時点でハードウェアによってクリアされます）。

0：ADDIS コマンドは実行中ではありません。

1：ADC を無効にするには、1 を書き込みます。1 として読み出されたときには、ADDIS コマンドが実行中であることを意味します。

注： ***ソフトウェアは、ADEN=1 で ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、ADDIS をセットできます。***

ビット 0 **ADEN**：ADC イネーブル制御

このビットは、ADC を有効にするために、ソフトウェアによってセットされます。フラグ ADRDY がセットされると、ADC は動作する準備ができています。

ADDIS コマンドの実行後、ADC が無効になるとハードウェアによってクリアされます。

0：ADC は無効です（OFF 状態）。

1：ADC を有効にするには、1 を書き込みます。

注： ***ソフトウェアは、1 にセットする必要がある ADVREGEN ビットを除き、ADCx_CR レジスタのすべてのビットが 0 のとき（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）だけ、ADEN をセットできます（また、ソフトウェアは、電圧レギュレータの起動時間まで待っていなければなりません）。***

25.5.4 ADC 設定レジスタ（ADCx_CFGR、x=1..3）

アドレスオフセット：0x0C

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JQDIS	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM[2:0]			DISCEN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL[4:0]					RES[2:0]			DMNGT[1:0]	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **JQDIS**：インジェクトキューディセーブル

これらのビットは、インジェクトキューメカニズムを無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトキューは有効です。

1：インジェクトキューは無効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。***

JQDIS ビットをセットまたはリセットすることで、インジェクトキューはフラッシュされ、JSQR レジスタはクリアされます。

ビット 30:26 **AWD1CH[4:0]**：アナログウォッチドッグ 1 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：AWD1 によって監視されている ADC アナログ入力チャネル 0

00001：AWD1 によって監視されている ADC アナログ入力チャネル 1

.....

10010：AWD1 によって監視されている ADC アナログ入力チャネル 19

その他。予約済み。使用できません。

注： ***AWD1CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。***

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 25 **JAUTO**：自動インジェクトグループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動インジェクトグループ変換は無効です。

1：自動インジェクトグループ変換は有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の JAUTO ビットへの書き込みができなくなり、その内容はマスタ ADC の JAUTO ビットと等しくなります。

ビット 24 **JAWD1EN**：アナログウォッチドッグ 1 有効（インジェクトチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルのアナログウォッチドッグ 1 は無効です。

1：インジェクトチャネルのアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 23 **AWD1EN**：アナログウォッチドッグ 1 有効（レギュラチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルのアナログウォッチドッグ 1 は無効です。

1：レギュラチャネルのアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 22 **AWD1SGL**：ウォッチドッグ 1 を単一チャネルまたはすべてのチャネルで有効にします。

このビットは、AWD1CH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ 1 は有効です。

1：単一チャネルでアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 21 **JQM**：JSQR キューモード

このビットは、ソフトウェアによってセット／クリアされます。

空のキューの管理方法について定義します。

0：JSQR モード 0：キューは空にならず、JSQR に最後に書き込んだ設定を保持します。

1：JSQR モード 1：キューは空である場合もありますが、このとき、インジェクトシーケンスのソフトウェア／ハードウェアトリガは、最後の有効なインジェクトシーケンスの完了後、すぐに両方とも内部的に無効化されます。

詳細については、*セクション 25.3.22：インジェクト変換のコンテキストのキュー*を参照してください。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の JQM ビットへの書き込みができなくなり、その内容はマスタ ADC の JQM ビットと等しくなります。

ビット 20 **JDISCEN**：インジェクトチャネルの不連続モード

このビットは、グループのインジェクトチャネルに対する不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルの不連続モードは無効です。

1：インジェクトチャネルの不連続モードは有効です。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の JDISCEN ビットへの書き込みができなくなり、その内容はマスタ ADC の JDISCEN ビットと等しくなります。

ビット 19:17 **DISCNUM[2:0]**：不連続モードチャネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるレギュラチャネルの数を定義するために、ソフトウェアによって書き込まれます。

000：1 チャネル

001：2 チャネル

...

111：8 チャネル

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の DISCNUM[2:0] ビットへの書き込みができなくなり、その内容はマスタ ADC の DISCNUM[2:0] ビットと等しくなります。

ビット 16 **DISCEN**：レギュラチャネルの不連続モード

このビットは、レギュラチャネルの不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルの不連続モードは無効です。

1：レギュラチャネルの不連続モードは有効です。

注： ***不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の DISCEN ビットへの書き込みができなくなり、その内容はマスタ ADC の DISCEN ビットと等しくなります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **AUTDLY**：遅延変換モード

このビットは、自動遅延変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動遅延変換モードはオフです。

1：自動遅延変換モードはオンです。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の AUTDLY ビットへの書き込みができなくなり、その内容はマスタ ADC の AUTDLY ビットと等しくなります。

ビット 13 **CONT**：レギュラ変換のシングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的にレギュラ変換が行われます。

0：シングル変換モード

1：連続変換モード

注： ***不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。***

ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0 以外）である場合、スレーブ ADC の CONT ビットへの書き込みができなくなり、その内容はマスタ ADC の CONT ビットと等しくなります。

ビット 12 **OVRMOD**：オーバーランモード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADCx_DR レジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADCx_DR レジスタは最後の変換結果で上書きされます。

注： ***ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 11:10 **EXTEN[1:0]**：レギュラチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しレギュラグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 9:5 **EXTSEL[4:0]**：レギュラグループの外部トリガ選択

これらのビットは、レギュラグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0
00001：イベント 1
00010：イベント 2
00011：イベント 3
00100：イベント 4
00101：イベント 5
00110：イベント 6
00111：イベント 7
...
11111：イベント 31

注：　ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 4:2 **RES[2:0]**：データ分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

000：16 ビット
001：14 ビット
010：12 ビット
011：10 ビット
100：8 ビット
その他のコードはすべて予約済み

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 1:0 **DMNGT[1:0]**：データ管理設定

このビットは、ソフトウェアによってセット／クリアされ、ADC インタフェース出力データを管理する方法を選択します。

00：レギュラ変換データは DR でのみ保存されます。
01：DMA ワンショットモードが選択されています。
10：DFSDM モードが選択されます。
11：DMA サーキュラモードが選択されています。

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル ADC モードでは、このビットは無関係で、ADCx_CCR レジスタの制御ビット DAMDF に置き換えられます。

25.5.5 ADC 設定レジスタ 2（ADCx_CFGR2、x=1..3）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LSHIFT[3:0]				Res.	Res.	OSR[9:0]									
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RSHIFT4	RSHIFT3	RSHIFT2	RSHIFT1	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw

ビット 31:28 **LSHIFT[3:0]**：左シフト係数

このビットフィールドは、オーバーサンプリングありとなしの最終結果に適用する左シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：左シフトなし
0001：左シフト 1 ビット
0010：左シフト 2 ビット
0011：左シフト 3 ビット
0100：左シフト 4 ビット
0101：左シフト 5 ビット
0110：左シフト 6 ビット
0111：左シフト 7 ビット
1000：左シフト 8 ビット
1001：左シフト 9 ビット
1010：左シフト 10 ビット
1011：左シフト 11 ビット
1100：左シフト 12 ビット
1101：左シフト 13 ビット
1101：左シフト 14 ビット
1111：左シフト 15 ビット

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 27:26　予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **OSR[9:0]**：オーバーサンプリング比

このビットフィールドは、オーバーサンプリング比を定義するために、ソフトウェアによってセット／クリアされます。

0：1x（オーバーサンプリングなし）
1：2x
2：3x
...
1023：1024x

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 15　予約済みであり、リセット値に保持する必要があります。

ビット 14 **RSHIFT4**：オフセット 4 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 13 **RSHIFT3**：オフセット 3 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 12 **RSHIFT2**：オフセット 2 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 11 **RSHIFT1**：オフセット 1 修正後の右シフト

このビットフィールドは、オフセット 1 修正後にデータを 1 ビット右シフトするために、ソフトウェアによってセット／クリアされます。このビットは、8 ビットフォーマットと 16 ビットフォーマットでのみ使用できます（詳細は*セクション：データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）*参照）。

0：右シフト無効

1：データを 1 ビット右シフト

ビット 10 **ROVSM**：レギュラオーバーサンプリングモード

このビットは、レギュラオーバーサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：連続モード：インジェクト変換がトリガされると、オーバーサンプリングが一時的に停止され、インジェクションシーケンス後に続行されます（オーバーサンプリングバッファはインジェクションシーケンス中、維持されます）。

1：再開モード：インジェクト変換がトリガされると、現在のオーバーサンプリングが中止され、インジェクションシーケンス後に最初から再開されます（オーバーサンプリングバッファはインジェクションシーケンス開始によって 0 になります）。

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **TROVS**：トリガレギュラオーバーサンプリング

このビットは、トリガオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとに新しいトリガが必要です。

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8:5 **OVSS[3:0]**：オーバーサンプリング右シフト

このビットフィールドは、そのままのオーバーサンプリング結果に適用する右シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：右シフトなし
0001：右シフト 1 ビット
0010：右シフト 2 ビット
0011：右シフト 3 ビット
0100：右シフト 4 ビット
0101：右シフト 5 ビット
0110：右シフト 6 ビット
0111：右シフト 7 ビット
1000：右シフト 8 ビット
1001：右シフト 9 ビット
1010：右シフト 10 ビット
1011：右シフト 11 ビット
その他のコードは予約済み

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 4:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**JOVSE**：インジェクトオーバーサンプリング有効化

このビットは、インジェクトオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトオーバーサンプリングは無効です。

1：インジェクトオーバーサンプリングは有効です。

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 0　**ROVSE**：レギュラオーバーサンプリング有効化

このビットは、レギュラオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラオーバーサンプリングは無効です。

1：レギュラオーバーサンプリングは有効です。

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

25.5.6　ADC サンプル時間レジスタ 1 （ADCx_SMPR1、x=1..3）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP9[2:0]			SMP8[2:0]			SMP7[2:0]			SMP6[2:0]			SMP5[2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]	SMP4[2:0]			SMP3[2:0]			SMP2[2:0]			SMP1[2:0]			SMP0[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29:0　**SMPx[2:0]**：チャネル x サンプリング時間選択

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000：1.5 ADC クロックサイクル

001：2.5 ADC クロックサイクル

010：8.5 ADC クロックサイクル

011：16.5 ADC クロックサイクル

100：32.5 ADC クロックサイクル

101：64.5 ADC クロックサイクル

110：387.5.5 ADC クロックサイクル

111：810.5 ADC クロックサイクル

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

25.5.7 ADC サンプル時間レジスタ 2 （ADCx_SMPR2、x=1..3）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP19[2:0]			SMP18[2:0]			SMP17[2:0]			SMP16[2:0]			SMP15[2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	SMP14[2:0]			SMP13[2:0]			SMP12[2:0]			SMP11[2:0]			SMP10[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29:0　**SMPx[2:0]**：チャネル x サンプリング時間選択

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプリングサイクル中は、チャネル選択ビットを変更することはできません。

000：1.5 ADC クロックサイクル
001：2.5 ADC クロックサイクル
010：8.5 ADC クロックサイクル
011：16.5 ADC クロックサイクル
100：32.5 ADC クロックサイクル
101：64.5 ADC クロックサイクル
110：387.5 ADC クロックサイクル
111：810.5 ADC クロックサイクル

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

25.5.8 ADC チャネル事前選択レジスタ（ADCx_PCSEL、x=1..3）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCSEL15	PCSEL14	PCSEL13	PCSEL12	PCSEL11	PCSEL10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

??? 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 PCSELx：チャネル x（$V_{INP[i]}$）事前選択

これらのビットは、変換する IO インスタンスの入力チャネルを事前選択するために、ソフトウェアによって書き込まれます。

0：入力チャネル x（Vinp x）は変換対象として事前選択されません。このチャネルのADC変換結果は誤った結果となります。

1：入力チャネル x（Vinp x）は変換対象として事前選択されます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.9 ADC ウォッチドッグ閾値レジスタ 1 （ADCx_LTR1、x=1..3）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR1[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR1[25:0]**：アナログウォッチドッグ 1 低閾値

これらのビットは、アナログウォッチドッグ 1 の低閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.10 ADC ウォッチドッグ閾値レジスタ 1 （ADCx_LHTR1、x=1..3）

アドレスオフセット：0x24

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR1[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR1[25:0]**：アナログウォッチドッグ 1 高閾値

これらのビットは、アナログウォッチドッグ 1 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.11 ADC レギュラシーケンスレジスタ 1 （ADCx_SQR1、x=1..3）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ4[4:0]					Res.	SQ3[4:0]					Res.	SQ2[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ2[3:0]				Res.	SQ1[4:0]					Res.	Res.	L[3:0]			
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ4[4:0]**：レギュラシーケンスの 4 番目の変換

これらのビットには、レギュラ変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ3[4:0]**：レギュラシーケンスの 3 番目の変換

これらのビットには、レギュラ変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ2[4:0]**：レギュラシーケンスの 2 番目の変換

これらのビットには、レギュラ変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ1[4:0]**：レギュラシーケンスの 1 番目の変換

これらのビットには、レギュラ変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **L[3:0]**：レギュラチャネルシーケンス長

これらのビットは、レギュラチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

0000：1 変換

0001：2 変換

...

1111：16 変換

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

25.5.12 ADC レギュラシーケンスレジスタ 2 （ADCx_SQR2、x=1..3）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ9[4:0]					Res.	SQ8[4:0]					Res.	SQ7[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ7[3:0]				Res.	SQ6[4:0]					Res.	SQ5[4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29　予約済みであり、リセット値に保持する必要があります。

ビット 28:24　**SQ9[4:0]**：レギュラシーケンスの 9 番目の変換

これらのビットには、レギュラ変換シーケンス内の 9 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 23　予約済みであり、リセット値に保持する必要があります。

ビット 22:18　**SQ8[4:0]**：レギュラシーケンスの 8 番目の変換

これらのビットには、レギュラ変換シーケンス内の 8 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 17　予約済みであり、リセット値に保持する必要があります。

ビット 16:12　**SQ7[4:0]**：レギュラシーケンスの 7 番目の変換

これらのビットには、レギュラ変換シーケンス内の 7 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 11　予約済みであり、リセット値に保持する必要があります。

ビット 10:6　**SQ6[4:0]**：レギュラシーケンスの 6 番目の変換

これらのビットには、レギュラ変換シーケンス内の 6 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 5　予約済みであり、リセット値に保持する必要があります。

ビット 4:0　**SQ5[4:0]**：レギュラシーケンスの 5 番目の変換

これらのビットには、レギュラ変換シーケンス内の 5 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

25.5.13 ADC レギュラシーケンスレジスタ 3 （ADCx_SQR3、x=1..3）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ14[4:0]					Res.	SQ13[4:0]					Res.	SQ12[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ12[3:0]				Res.	SQ11[4:0]					Res.	SQ10[4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29　予約済みであり、リセット値に保持する必要があります。

ビット 28:24　**SQ14[4:0]**：レギュラシーケンスの 14 番目の変換

これらのビットには、レギュラ変換シーケンス内の 14 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 23　予約済みであり、リセット値に保持する必要があります。

ビット 22:18　**SQ13[4:0]**：レギュラシーケンスの 13 番目の変換

これらのビットには、レギュラ変換シーケンス内の 13 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 17　予約済みであり、リセット値に保持する必要があります。

ビット 16:12　**SQ12[4:0]**：レギュラシーケンスの 12 番目の変換

これらのビットには、レギュラ変換シーケンス内の 12 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 11　予約済みであり、リセット値に保持する必要があります。

ビット 10:6　**SQ11[4:0]**：レギュラシーケンスの 11 番目の変換

これらのビットには、レギュラ変換シーケンス内の 11 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 5　予約済みであり、リセット値に保持する必要があります。

ビット 4:0　**SQ10[4:0]**：レギュラシーケンスの 10 番目の変換

これらのビットには、レギュラ変換シーケンス内の 10 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

25.5.14 ADC レギュラシーケンスレジスタ 4 （ADCx_SQR4、x=1..3）

アドレスオフセット：0x3C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SQ16[4:0]					Res.	SQ15[4:0]				
					rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:6　**SQ16[4:0]**：レギュラシーケンスの 16 番目の変換

これらのビットには、レギュラ変換シーケンス内の 16 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 5　予約済みであり、リセット値に保持する必要があります。

ビット 4:0　**SQ15[4:0]**：レギュラシーケンスの 15 番目の変換

これらのビットには、レギュラ変換シーケンス内の 15 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

25.5.15 ADC レギュラデータレジスタ（ADCx_DR、x=1..3）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RDATA[31:0]**：レギュラデータ変換

これらのビットは読み出し専用です。これらは、最後の変換されたレギュラチャネルの変換結果を含んでいます。*セクション 25.3.27：データ管理* に示すように、データは左詰めまたは右詰めされています。

25.5.16 ADC インジェクトシーケンスレジスタ（ADCx_JSQR、x=1..3）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JSQ4[4:0]					Res.	JSQ3[4:0]					Res.	JSQ2[4:1]			
rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ2[0]	Res.	JSQ1[4:0]					JEXTEN[1:0]		JEXTSEL[4:0]					JL[1:0]	
rw		rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:27 **JSQ4[4:0]**：インジェクトシーケンスの 4 番目の変換

これらのビットには、インジェクト変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **JSQ3[4:0]**：インジェクトシーケンスの 3 番目の変換

これらのビットには、インジェクト変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:15 **JSQ2[4:0]**：インジェクトシーケンスの 2 番目の変換

これらのビットには、インジェクト変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:9 **JSQ1[4:0]**：インジェクトシーケンスの 1 番目の変換

これらのビットには、インジェクト変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 8:7 **JEXTEN[1:0]**：インジェクトチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しインジェクトグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：JQDIS=0（キュー有効）の場合、ハードウェアおよびソフトウェアトリガ検出は無効になります。

00：JQDIS=1（キュー無効）の場合、ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

JQM=1 で、コンテキストのキューが空である場合、インジェクトシーケンスのソフトウェアおよびハードウェアトリガは両方とも内部的に無効化されます（セクション 25.3.22：インジェクト変換のコンテキストのキューを参照）。

ビット 6:2 **JEXTSEL[4:0]**：インジェクトグループの外部トリガ選択

これらのビットは、インジェクトグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0

00001：イベント 1

00010：イベント 2

00011：イベント 3

00100：イベント 4

00101：イベント 5

00110：イベント 6

00111：イベント 7

...

11111：イベント 31：

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

ビット 1:0 **JL[1:0]**：インジェクトチャネルシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

00：1 変換

01：2 変換

10：3 変換

11：4 変換

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

25.5.17 ADC オフセットレジスタ（ADCx_OFRy、x=1..3）（y=1..4）

アドレスオフセット：0x60、0x64、0x68、0x6C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SSATE	OFFSETy_CH[4:0]					OFFSETy[25:16]									
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSETy[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SSATE**：符号付き飽和イネーブル

このビットは、符号付き飽和機能を有効または無効にするために、ソフトウェアによって書き込まれます。このビットは、8 ビットフォーマットと 16 ビットフォーマットでのみ有効にできます（詳細は*セクション：データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）*参照）。

0：オフセットが減算され、データの完全性を維持して結果のサイズを拡張（9 ビットまたは 17 ビット符号付きフォーマット）します。

1：オフセットが減算され、結果のサイズを維持するために結果を飽和させます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 30:26 **OFFSETy_CH[4:0]**：データオフセット y のチャネル選択

このビットは、OFFSETy[25:0] ビットにプログラムされたオフセットが適用されるチャネルを定義するために、ソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 25:0 **OFFSETy[25:0]**：OFFSETy_CH[4:0] ビットにプログラムされたチャネルのデータオフセット y

これらのビットは、チャネル（レギュラまたはインジェクトのどちらか）を変換するときに元の変換データから引かれるオフセット y を定義するために、ソフトウェアによって書き込まれます。データオフセット y を適用するチャネルは、ビット OFFSETy_CH[4:0] ビットにプログラムする必要があります。変換結果は、ADCx_DR レジスタ（レギュラ変換）または ADCx_JDRyi レジスタ（インジェクト変換）から読み出すことができます。

OFFSETy[25:0] ビットフィールドがリセットされると、オフセット補償は無効化されます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

複数のオフセット（OFFSETy）が同じチャネルを指している場合、減算時には最小のx を持つオフセットのみが考慮されます。

例：OFFSET1_CH[4:0]=4 かつ OFFSET2_CH[4:0]=4 である場合、これはチャネル 4 を変換するときに減算される OFFSET1[25:0] です。

25.5.18 ADC インジェクトデータレジスタ（ADCx_JDRy、x=1..3、y= 1..4）

アドレスオフセット：0x80～0x8C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **JDATA[31:0]**：インジェクトデータ

これらのビットは読み出し専用です。これらのビットには、インジェクトチャネル y の変換結果が格納されます。*セクション 25.3.27：データ管理* に示すように、データは左詰めまたは右詰めされています。

25.5.19 ADC アナログウォッチドッグ 2 設定レジスタ（ADCx_AWD2CR、x=1..3）

アドレスオフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **AWD2CH[19:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[19:0] = 000..0 である場合、アナログウォッチドッグ 2 は無効になります。

注： ***AWD2CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。***

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

25.5.20 ADC アナログウォッチドッグ 3 設定レジスタ（ADCx_AWD3CR、x=1..3）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **AWD2CH[19:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[19:0] = 000..0 である場合、アナログウォッチドッグ 2 は無効になります。

注： ***AWD3CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。***

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

25.5.21 ADC ウォッチドッグ低閾値レジスタ 2 （ADCx_LTR2、x=1..3）

アドレスオフセット：0xB0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR2[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR2[25:0]**：アナログウォッチドッグ 2 低閾値

これらのビットは、アナログウォッチドッグ 2 の低閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.22 ADC ウォッチドッグ高閾値レジスタ 2 （ADCx_HTR2、x=1..3）

アドレスオフセット：0xB4

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR2[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR2[25:0]**：アナログウォッチドッグ 2 高閾値

これらのビットは、アナログウォッチドッグ 2 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.23 ADC ウォッチドッグ低閾値レジスタ 3 （ADCx_LTR3、x=1..3）

アドレスオフセット：0xB8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR3[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR3[25:0]**：アナログウォッチドッグ 3 低閾値

これらのビットは、アナログウォッチドッグ 3 の低閾値を定義するために、ソフトウェアによって書き込まれます。

参照先：*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.24 ADC ウォッチドッグ高閾値レジスタ 3 （ADCx_HTR3、x=1..3）

アドレスオフセット：0xBC

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR3[25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR3[25:0]**：アナログウォッチドッグ 3 高閾値

これらのビットは、アナログウォッチドッグ 3 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 25.3.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

25.5.25 ADC 差動モード選択レジスタ （ADCx_DIFSEL、x=1..3）

アドレスオフセット：0xC0

リセット値：0x0000 0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **DIFSEL[19:0]**：チャネル（19 ～ 0）の差動モード

これらのビットは、ソフトウェアによってセット／クリアされます。これにより、チャネルの設定をシングルエンドモードまたは差動モードから選択できます。

DIFSEL[i] = 0：ADC アナログ入力チャネル i はシングルエンドモードで設定されます。

DIFSEL[i] = 1：ADC アナログ入力チャネル i は差動モードで設定されます。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

25.5.26 ADC 較正係数レジスタ（ADCx_CALFACT、x=1..3）

アドレスオフセット：0xC4

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:16 **CALFACT_D[10:0]**：差動モードでの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

差動入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しい差動較正の起動時に適用されます。

注： ***ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。***

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **CALFACT_S[10:0]**：シングルエンドモードの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しいシングルエンド較正の起動時に適用されます。

注： ***ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。***

25.5.27 ADC 較正係数レジスタ 2（ADCx_CALFACT2、x=1..3）

アドレスオフセット：0xC8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	LINCALFACT[29:16]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LINCALFACT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **LINCALFACT[29:0]**：直線性較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

ここには、160 ビット直線性較正係数のうち、30 ビットが保持されます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しいシングルエンド較正の起動時に適用されます。

注： ***ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。***

25.5.28 ADC オプションレジスタ（ADCx_OR、x=1..3）

アドレスオフセット：0xD0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:0 予約済みであり、リセット値に保持する必要があります。

25.6 ADC 共通レジスタ

これらのレジスタは、マスタおよびスレーブ ADC に共通の制御およびステータスレジスタを定義します。

25.6.1 ADC 共通ステータスレジスタ（ADCx_CSR、x=12 または 3）

アドレスオフセット：0x00（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

このレジスタにより、別の ADC のステータスビットの状態を確認することができます。ただし、これは読み出し専用で、別の ADC のステータスビットをクリアすることはできません。その代わり、対応する ADCx_ISR レジスタに 0 を書き込むことによって各ステータスビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV
					r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **JQOVF_SLV**：スレーブ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADCx+1_ISR レジスタの JQOVF ビットのコピーです。

ビット 25 **AWD3_SLV**：スレーブ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD3 ビットのコピーです。

ビット 24 **AWD2_SLV**：スレーブ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD2 ビットのコピーです。

ビット 23 **AWD1_SLV**：スレーブ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD1 ビットのコピーです。

ビット 22 **JEOS_SLV**：スレーブ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの JEOS ビットのコピーです。

ビット 21 **JEOC_SLV**：スレーブ ADC のインジェクト変換の終了フラグ
このビットは対応する ADCx+1_ISR レジスタの JEOC ビットのコピーです。

ビット 20 **OVR_SLV**：スレーブ ADC のオーバーランフラグ
このビットは対応する ADCx+1_ISR レジスタの OVR ビットのコピーです。

ビット 19 **EOS_SLV**：スレーブ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOS ビットのコピーです。

ビット 18 **EOC_SLV**：スレーブ ADC のレギュラ変換の終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOC ビットのコピーです。

ビット 17 **EOSMP_SLV**：スレーブ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOSMP2 ビットのコピーです。

ビット 16 **ADRDY_SLV**：スレーブ ADC レディ
このビットは対応する ADCx+1_ISR レジスタの ADRDY ビットのコピーです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF_MST**：マスタ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADCx_ISR レジスタの JQOVF ビットのコピーです。

ビット 9 **AWD3_MST**：マスタ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADCx_ISR レジスタの AWD3 ビットのコピーです。

ビット 8 **AWD2_MST**：マスタ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADCx_ISR レジスタの AWD2 ビットのコピーです。

ビット 7 **AWD1_MST**：マスタ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADCx_ISR レジスタの AWD1 ビットのコピーです。

ビット 6 **JEOS_MST**：マスタ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADCx_ISR レジスタの JEOS ビットのコピーです。

ビット 5 **JEOC_MST**：マスタ ADC のインジェクト変換の終了フラグ
このビットは対応する ADCx_ISR レジスタの JEOC ビットのコピーです。

ビット 4 **OVR_MST**：マスタ ADC のオーバーランフラグ
このビットは対応する ADCx_ISR レジスタの OVR ビットのコピーです。

ビット 3 **EOS_MST**：マスタ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADCx_ISR レジスタの EOS ビットのコピーです。

ビット 2 **EOC_MST**：マスタ ADC のレギュラ変換の終了フラグ
このビットは対応する ADCx_ISR レジスタの EOC ビットのコピーです。

ビット 1 **EOSMP_MST**：マスタ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADCx_ISR レジスタの EOSMP ビットのコピーです。

ビット 0 **ADRDY_MST**：マスタ ADC レディ
このビットは対応する ADCx_ISR レジスタの ADRDY ビットのコピーです。

25.6.2 ADC 共通制御レジスタ（ADCx_CCR、x=12 または 3）

アドレスオフセット：0x08（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	TSEN	VREFEN	PRESC[3:0]				CKMODE[1:0]	
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DAMDF[1:0]		Res.	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
rw	rw			rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **VBATEN**：VBAT イネーブル

このビットは、ADC3 のチャネル 17 を制御するために、ソフトウェアによってセット／クリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 23 **TSEN**：**温度センサ有効**

このビットは、ADC3 のチャネル 18 を制御するために、ソフトウェアによってセット／クリアされます。

0：温度センサチャネルは無効です。

1：温度センサチャネルは有効です。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 22 **VREFEN**：V_{REFINT} 有効

このビットは、V_{REFINT} チャネルを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} チャネルは無効です。

1：V_{REFINT} チャネルは有効です。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 21:18 **PRESC[3:0]**：ADC プリスケーラ

これらのビットは、ADC へのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべての ADC に共通です。

0000：ADC 入力クロックは分周されません。
0001：ADC 入力クロックが 2 分周されます。
0010：ADC 入力クロックが 4 分周されます。
0011：ADC 入力クロックが 6 分周されます。
0100：ADC 入力クロックが 8 分周されます。
0101：ADC 入力クロックが 10 分周されます。
0110：ADC 入力クロックが 12 分周されます。
0111：ADC 入力クロックが 16 分周されます。
1000：ADC 入力クロックが 32 分周されます。
1001：ADC 入力クロックが 64 分周されます。
1010：ADC 入力クロックが 128 分周されます。
1011：ADC 入力クロックが 256 分周されます。
その他：予約済み

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。CKMODE[1:0] = 0b00 である場合のみ、ADC プリスケーラ値が適用されます。***

ビット 17:16 **CKMODE[1:0]**：ADC クロックモード

これらのビットは、ADC クロックスキーム（マスタとスレーブの両方の ADC に共通）を定義するために、ソフトウェアによってセット／クリアされます。

00：CK_ADCx（x=123）（非同期クロックモード）。製品レベルで生成されます（*セクション 8：リセットおよびクロック制御（RCC）*を参照）。
01：adc_hclk/1（同期クロックモード）。この設定は、AHB クロックプリスケーラが 1 にセットされた場合で（RCC_CFGR レジスタの HPRE[3:0] = 0xxx）、システムクロックが 50% のデューティサイクルを持つ場合にのみ有効にする必要があります。
10：adc_hclk/2（同期クロックモード）。
11：adc_hclk/4（同期クロックモード）。

すべての同期クロックモードにおいて、タイマトリガから変換開始までの遅延にジッタはありません。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

ビット 15:14 **DAMDF[1:0]**：デュアル ADC モードデータフォーマット

このビットフィールドは、ソフトウェアによってセット／クリアされます。共通データレジスタ ADCx_CDR のデータフォーマットを指定します。

00：データパッキングなしデュアル ADC モード（ADCx_CDR レジスタと ADCx_CDR2 レジスタは未使用）
01：予約済み
10：32 ビット分解能以下 10 ビット分解能までのデータフォーマット設定モード
11：8 ビット分解能のデータフォーマット設定モード

注： ***ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **DELAY**：2 つのサンプリングフェーズ間の遅延

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットはデュアルインタリーブモードで使用します。ADC 分解能の値対 DELAY ビット値については、*表 203* を参照してください。

注：　ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DUAL[4:0]**：デュアル ADC モード選択

これらのビットは、動作モードを選択するために、ソフトウェアによって書き込まれます。
すべての ADC が単独で動作します。
00000：独立モード

00001 から 01001：デュアルモード、マスタおよびスレーブ ADCは協調して動作
00001：レギュラ同時 + インジェクト同時モードの組み合わせ
00010：レギュラ同時 + オルタネートトリガモードの組み合わせ
00011：インタリーブモード + インジェクト同時モードの組み合わせ
00100：予約済み
00101：インジェクト同時モードのみ
00110：レギュラ同時モードのみ
00111：インタリーブモードのみ
01001：オルタネートトリガモードのみ
他のすべての組み合わせは予約済みであり、プログラムすることはできません。

注：　ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。

表 203. DELAY ビット対 ADC 分解能

DELAY ビット	16 ビット分解能	14 ビット分解能	12 ビット分解能	10 ビット分解能	8 ビット分解能
0000	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$
0001	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$
0010	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0011	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0100	5.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0101	6.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0110	7.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0111	7.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
1000	9 * $T_{adc_ker_ck}$	9 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
その他：予約済み	-	-	-	-	-

25.6.3 デュアルモード用 ADC 共通レギュラデータレジスタ（ADCx_CDR、x=12 または 3）

アドレスオフセット：0x0C（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA_SLV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA_MST[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **RDATA_SLV[15:0]**：スレーブ ADC のレギュラデータ

デュアルモードでは、これらのビットにはスレーブ ADC のレギュラデータが格納されます。*セクション 25.3.32：デュアル ADC モード*を参照してください。

データの配置は、*セクション：データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）*に示すように適用されています。

ビット 15:0 **RDATA_MST[15:0]**：マスタ ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC のレギュラデータが格納されます。*セクション 25.3.32：デュアル ADC モード*を参照してください。

データの配置は、*セクション：データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）*に示すように適用されています。

MDMA=0b11 モードでは、ビット 15:8 には SLV_ADC_DR[7:0] が、ビット 7:0 には MST_ADC_DR[7:0] が含まれています。

25.6.4 32 ビットデュアルモード用 ADC 共通レギュラデータレジスタ（ADCx_CDR2、x=12 または 3）

アドレスオフセット：0x10（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA_ALT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA_ALT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RDATA_ALT[31:0]**：マスタ／スレーブオルタネート ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC とスレーブ ADC のレギュラ 32 ビットデータが代わりに格納されます。*セクション 25.3.32：デュアル ADC モード*を参照してください。

データの配置は、*セクション：　データレジスタ、データの配置およびデータオフセット（ADCx_DR、ADCx_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）*に示すように適用されています。

25.6.5 ADC レジスタマップ

次の表は ADC レジスタの一覧です。

表 204. ADC グローバルレジスタマップ

オフセット	レジスタ
0x000～0x0D0	マスタ ADC1 またはマスタ ADC3
0x0D4～0x0FC	予約済み
0x100～0x1D0	スレーブ ADC2
0x1D4～0x2FC	予約済み
0x300～0x310	マスタおよびスレーブ ADC に共通のレジスタ（ADC12 または ADC3）

表 205. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADCx_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x04	**ADCx_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x08	**ADCx_CR**	ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	LINCALRDYW6	LINCALRDYW5	LINCALRDYW4	LINCALRDYW3	LINCALRDYW2	LINCALRDYW1	Res.	Res.	Res.	Res.	Res.	ADCALLIN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOST	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
	リセット値	0	0	1	0	0	0	0	0	0	0						0							0			0	0	0	0	0	0
0x0C	**ADCx_CFGR**	JQDIS	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM [2:0]			DISCEN	Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL [4:0]					RES [2:0]			DMN GT [1:0]	
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**ADCx_CFGR2**	Res.	Res.	Res.	Res.	Res.	Res.	OSR[9:0]										Res.	RSHIFT4	RSHIFT3	RSHIFT2	RSHIFT1	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE
	リセット値							0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0				0	0
0x14	**ADCx_SMPR1**	Res.	Res.	SMP9[2:0]			SMP8[2:0]			SMP7[2:0]			SMP6[2:0]			SMP5[2:0]		SMP4[2:0]			SMP3[2:0]			SMP2[2:0]			SMP1[2:0]			SMP0[2:0]		
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**ADCx_SMPR2**	Res.	Res.	Res.	Res.	Res.	SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:0]		SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**ADCx_PCSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16	PCSEL15	PCSEL14	PCSEL13	PCSEL12	PCSEL11	PCSEL10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**ADCx_LTR1**	Res.	Res.	Res.	Res.	Res.	Res.	LTR1[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 205. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**ADCx_HTR1**	Res.	Res.	Res.	Res.	Res.	Res.	HTR1[25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x28	予約済み	Res.																															
0x2C	予約済み	Res.																															
0x30	**ADCx_SQR1**	Res.	Res.	Res.	SQ4[4:0]				Res.	SQ3[4:0]						Res.	SQ2[4:0]					Res.	SQ1[4:0]					Res.	Res.	L[3:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0
0x34	**ADCx_SQR2**	Res.	Res.	Res.	SQ9[4:0]				Res.	SQ8[4:0]						Res.	SQ7[4:0]					Res.	SQ6[4:0]					Res.	Res.	SQ5[4:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0
0x38	**ADCx_SQR3**	Res.	Res.	Res.	SQ14[4:0]				Res.	SQ13[4:0]						Res.	SQ12[4:0]					Res.	SQ11[4:0]					Res.	Res.	SQ10[4:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0
0x3C	**ADCx_SQR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SQ16[4:0]					Res.	SQ15[4:0]				
	リセット値																						0	0	0	0	0		0	0	0	0	0
0x40	**ADCx_DR**	RDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44-0x48	予約済み	Res.																															
0x4C	**ADCx_JSQR**	JSQ4[4:0]					Res.	JSQ3[4:0]					Res.	JSQ2[4:0]				Res.	JSQ1[4:0]					JEXTEN[1:0]		JEXTSEL[4:0]					JL[1:0]		
	リセット値	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50-0x5C	予約済み	Res.																															
0x60	**ADCx_OFR1**	SSATE	OFFSET1_CH[4:0]					OFFSET1[25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	**ADCx_OFR2**	SSATE	OFFSET2_CH[4:0]					OFFSET2[25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	**ADCx_OFR3**	SSATE	OFFSET3_CH[4:0]					OFFSET3[25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	**ADCx_OFR4**	SSATE	OFFSET4_CH[4:0]					OFFSET4[25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 205. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x70-0x7C	予約済み	Res.																															
0x80	**ADCx_JDR1**	JDATA1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**ADCx_JDR2**	JDATA2[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	**ADCx_JDR3**	JDATA3[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**ADCx_JDR4**	JDATA4[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C-0x9C	予約済み	Res.																															
0xA0	**ADCx_AWD2CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	**ADCx_AWD3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA8-0xAC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB0	**ADCx_LTR2**	Res.	Res.	Res.	Res.	Res.	Res.	LTR2[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**ADCx_HTR2**	Res.	Res.	Res.	Res.	Res.	Res.	HTR2[25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0xB8	**ADCx_LTR3**	Res.	Res.	Res.	Res.	Res.	Res.	LTR3[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xBC	**ADCx_HTR3**	Res.	Res.	Res.	Res.	Res.	Res.	HTR3[25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0xC0	**ADCx_DIFSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC4	**ADCx_CALFACT**	Res.	Res.	Res.	Res.	Res.	CALFACT_D[10:0]											Res.	Res.	Res.	Res.	Res.	CALFACT_S[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0xC8	**ADCx_ CALFACT2**	Res.	Res.	LINCALFACT[29:0]																													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xCC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xD0	ADCx_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 206. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	ADCx_CSR	Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV	Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
							スレーブ ADC2																マスタ ADC1										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x04	予約済み	Res.																															
0x08	ADCx_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	TSEN	VREFEN	PRESC[3:0]				CKMODE[1:0]		DAMDF[1:0]		DMACFG	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0				0	0	0	0	0
0x0C	ADCx_CDR	RDATA_SLV[15:0]																RDATA_MST[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	ADCx_CDR2	RDATA_ALT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

参考資料

26 D/A コンバータ（DAC）

26.1 概要

DAC モジュールは、12 ビットの電圧出力デジタルアナログコンバータです。DAC は、8 または 12 ビットモードで設定でき、DMA コントローラと組み合わせて使用することもできます。12 ビットモードでは、データを左詰め右詰めのどちらにも配置できます。DAC には 2 つの出力チャネルがあり、それぞれがコンバータを搭載しています。デュアル DAC チャネルモードでは、変換は独立して行うか、両方のチャネルが同期更新操作のためにグループ化されているときには同時に行うことができます。精度を高めるために、入力基準ピン V_{REF+}（その他のアナログペリフェラルと共用）を使用することができます。内部基準も、同じ入力でセットできます。*電圧基準バッファ（VREFBUF）*セクションを参照してください。

DAC 出力が出力パッドから切断され、オンチップペリフェラルに接続されている場合、DAC_OUTx ピンは汎用入力／出力（GPIO）として使用できます。駆動出力電流を上げるために、オプションで DAC 出力バッファを有効にできます。個別の較正が各 DAC 出力チャネルに適用可能です。DAC 出力チャネルは低電力モード（サンプルおよびホールドモード）をサポートしています。

26.2 DAC の主な機能

DAC の主な機能は以下のとおりです（*図 194：DAC チャネルブロック図*を参照）。

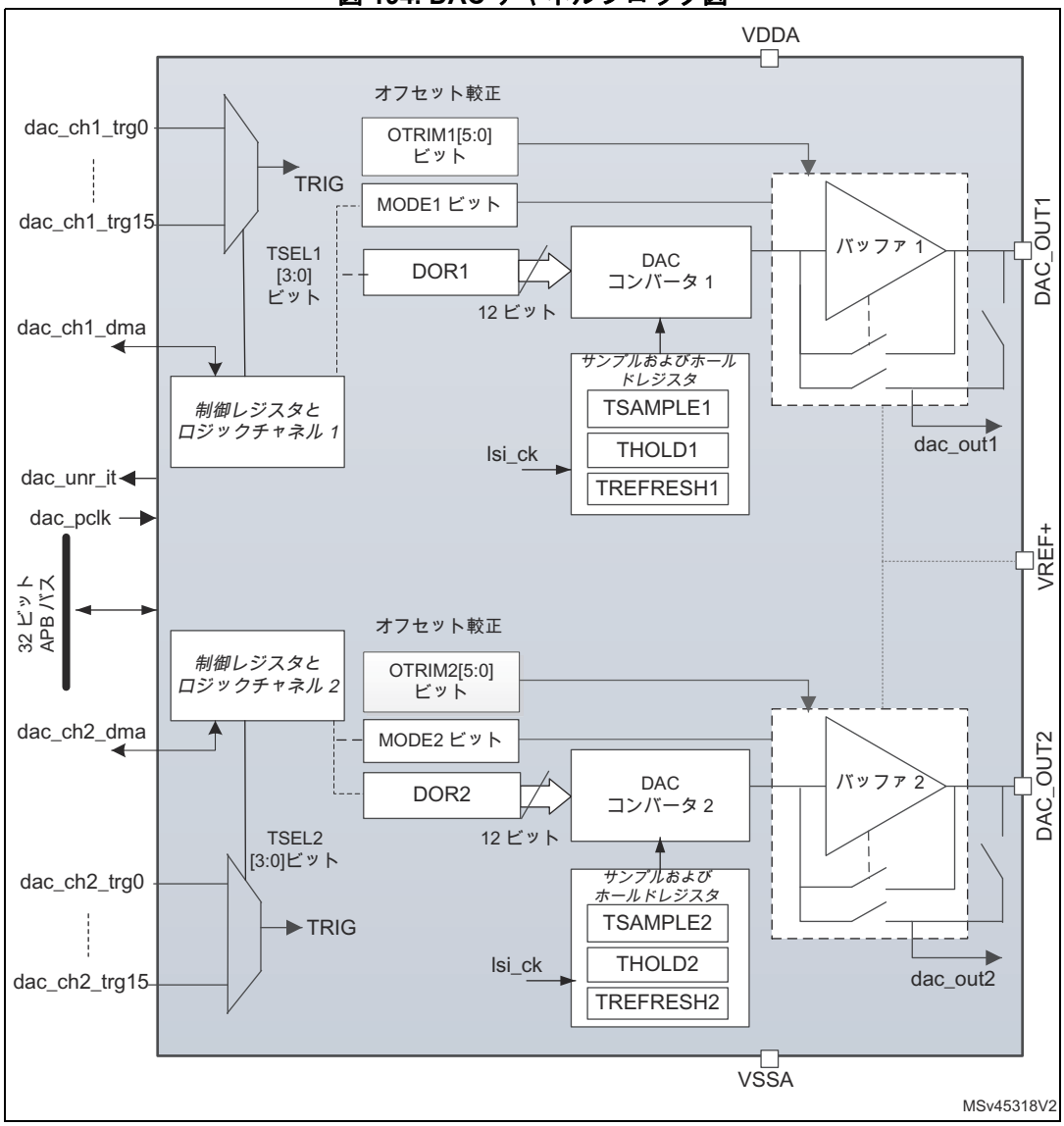
- 2 つの DAC コンバータ：それぞれ 1 つの出力チャネル
- 12 ビットモードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波および三角波生成
- デュアル DAC チャネルの独立または同時変換
- DMA アンダーランエラー検出を含む各チャネルの DMA 機能
- 変換外部トリガ
- DAC 出力チャネルバッファ／バッファ無モード
- バッファオフセット較正
- 各 DAC 出力は DAC_OUTx 出力ピンから切り離し可能
- オンチップペリフェラルへの DAC 出力接続
- STOP モードの低電力動作でのサンプルおよびホールドモード
- V_{REF+} ピンまたは内部 VREFBUF 基準からの入力電圧基準

図 194 は DAC チャネルのブロック図を、*表 207* はピンの概要を示します。

26.3 DAC の機能詳細

26.3.1 DAC ブロック図

図 194. DAC チャネルブロック図



1. 出力モードコントローラはバッファ有／バッファ無の通常モードとサンプルおよびホールドモードを切り替えます。

26.3.2 DAC ピンおよび内部信号

DAC には次の要素が含まれます。

- 最大 2 つの出力チャネル
- DAC_OUTx を出力ピンから切り離して標準の GPIO として使用可能
- DAC_OUTx でコンパレータおよび OPAMP などオンチップペリフェラルへの内部ピン接続が使用可能
- DAC 出力チャネルのバッファ有またはバッファ無
- LSI クロックソース (lsi_ck) を使用し、スタティック変換のために STOP モードで動作可能なサンプルホールドブロックとレジスタ

DAC には最大 2 つの個別出力チャネルが搭載されています。各出力チャネルは、COMP、OPAMP および ADC などオンチップペリフェラルに接続できます。この場合、DAC 出力チャネルは DAC_OUTx 出力ピンから切断でき、対応する GPIO を別の目的に使用できます。

DAC 出力はバッファの有無を設定できます。サンプルホールドブロックと、それに関連するレジスタは、LSI クロックソース (lsi_ck) を使用して STOP モードで動作可能です。

表 207. DAC ピン

名前	信号タイプ	説明
V_{REF+}	入力、アナログ基準電圧正	DAC のハイレベル／正基準電圧 $V_{DDAmin} \leq V_{REF+} \leq V_{DDA}$（データシートを参照）
V_{DDA}	入力、アナログ電源供給	アナログ電源供給
V_{SSA}	入力、アナログ供給グラウンド	アナログ電源供給のグラウンド
DAC_OUTx	アナログ出力信号	DAC チャネル x アナログ出力

表 208. DAC 内部信号

内部信号名	信号タイプ	説明
dac_ch1_dma	デジタル入出力	DAC チャネル 1 DMA リクエスト
dac_ch2_dma	デジタル入出力	DAC チャネル 2 DMA リクエスト
dac_ch1_trg[0:15]	デジタル入力	DAC チャネル 1
dac_ch2_trg[0:15]	デジタル入力	DAC チャネル 2
dac_unr_it	デジタル出力	DAC アンダーラン割り込み
dac_pclk	デジタル入力	DAC ペリフェラルクロック
dac_out1	アナログ出力	オンチップペリフェラル用 DAC チャネル 1 出力
dac_out2	アナログ出力	オンチップペリフェラル用 DAC チャネル 2 出力

26.3.3 DAC チャネルイネーブル

各 DAC チャネルは、DAC_CR レジスタの対応する ENx ビットをセットすることによって起動できます。DAC チャネルは、スタートアップ時間 t_{WAKEUP}後に有効になります。

注： ***ENx ビットは、アナログ DAC チャネル x のみを有効にします。DAC チャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。***

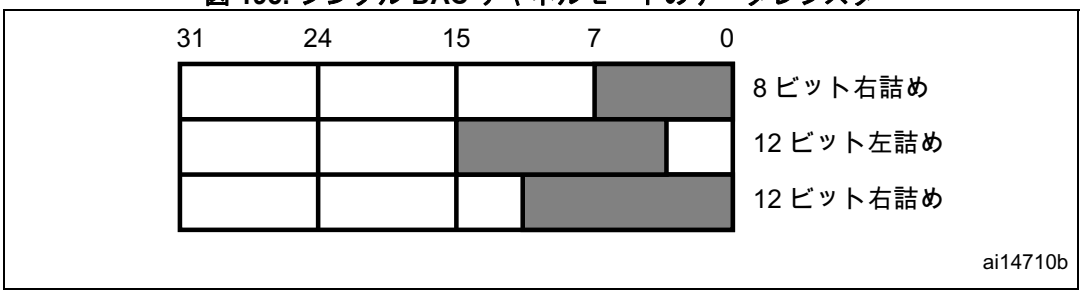
26.3.4 DAC データフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- シングル DAC チャネルx では、3 つの設定が可能です。
  - 8 ビット右詰め：データは、DAC_DHR8Rx [7:0] ビット（DHRx [11:4] ビットに格納）にソフトウェアによってロードされること。
  - 12 ビット左詰め：データは、DAC_DHR12Lx [15:4] ビット（DHRx[11:0] ビットに格納）にソフトウェアによってロードされること。
  - 12 ビット右詰め：データは、DAC_DHR12Rx [11:0] ビット（DHRx [11:0] ビットに格納）にソフトウェアによってロードされること。

ユーザによって書き込まれたデータは、ロードされた DAC_DHRyyyx レジスタに応じて、シフトされてから、DHRx（メモリマップされない内部レジスタであるデータ保持レジスタ x）に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、DORx レジスタにロードされます。

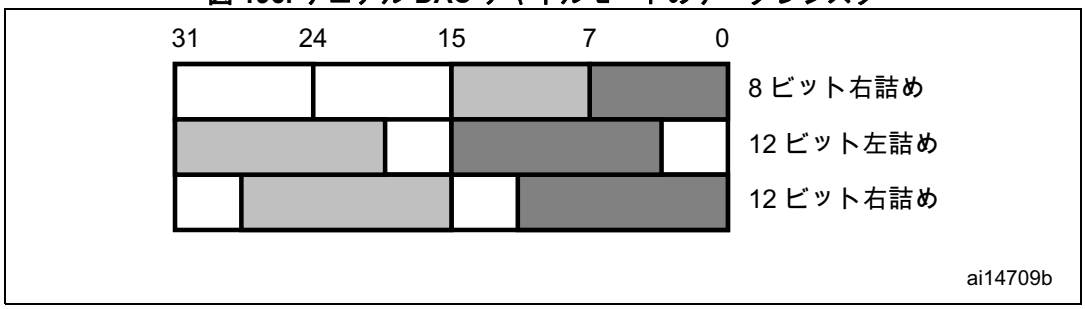
図 195. シングル DAC チャネルモードのデータレジスタ



- デュアル DAC チャネル（使用可能な場合）では、3 つの設定が可能です。
  - 8 ビット右詰め：DAC チャネル 1 のデータは、DAC_DHR8RD [7:0] ビット（DHR1[11:4] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR8RD [15:8] ビット（DHR2[11:4] ビットに格納）にロードされます。
  - 12 ビット左詰め：DAC チャネル 1 のデータは、DAC_DHR12LD [15:4] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12LD [31:20] ビット（DHR2[11:0]ビットに格納）にロードされます。
  - 12 ビット右詰め：DAC チャネル 1 のデータは、DAC_DHR12RD [11:0] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12RD [27:16] ビット（DHR2[11:0]ビットに格納）にロードされます。

ロードされた DAC_DHRyyyD レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1 および DHR2（Data Holding Register、メモリマップされない内部レジスタ）に格納されます。その後、DHR1 および DHR2 レジスタは、自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、それぞれ DOR1 および DOR2 レジスタにロードされます。

図 196. デュアル DAC チャネルモードのデータレジスタ



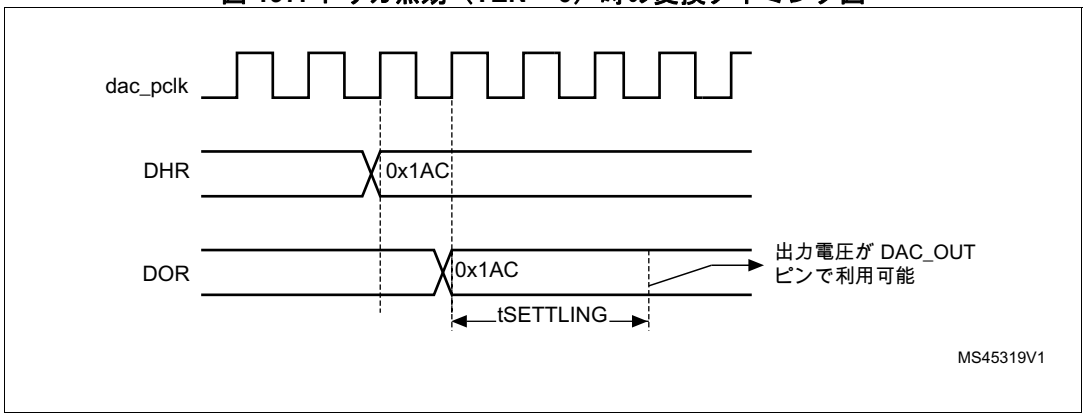
26.3.5 DAC 変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって（DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx、DAC_DHR8RD、DAC_DHR12RD、または DAC_DHR12LD への書き込み）、DAC チャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない（DAC_CR レジスタの TENx ビットがリセットされている）場合に、1 dac_pclk クロックサイクル後に DAC_DORx レジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている（DAC_CR レジスタの TENx ビットがセットされている）ときには、トリガが発生すると、転送は 3 dac_pclk クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される $t_{SETTLING}$ 時間後にアナログ出力電圧が使用可能になります。

図 197. トリガ無効（TEN = 0）時の変換タイミング図



26.3.6 DAC 出力電圧

デジタル入力は、0 から V_{REF+} までのリニア変換で出力電圧に変換されます。

各 DAC チャネルピンのアナログ出力電圧は、次の式によって求められます。

$$DACoutput = V_{REF} \times \frac{DOR}{4096}$$

26.3.7 DAC トリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割り込みラインなど）によって変換をトリガできます。*セクション 26.5.1：DAC 制御レジスタ（DAC_CR）*の TSEL1[3:0] ビットおよび TSEL2[3:0] ビットに示すように、16 個の考えられるイベントのうち、どのイベントが変換をトリガするかは TSELx[3:0] 制御ビットによって決まります。

DAC インタフェースが選択されたトリガソース（次の表を参照）で立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータが DAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから 3 dac_pclk サイクル後に更新されます。

ソフトウェアトリガが選択されている場合、変換は、SWTRIG ビットがセットされると開始されます。SWTRIG ビットは、DAC_DHRx レジスタの内容が DAC_DORx レジスタにロードされると、ハードウェアによってリセットされます。

注： ***ENx ビットがセットされているときには、TSELx[3:0] ビットを変更することはできません。***

ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DORx レジスタへの転送は、わずか 1 APB クロックサイクルで行われます。

表 209. DAC トリガ選択

転送元	タイプ	TSELx[3:0]
SWTRIG	ソフトウェア制御ビット	0000
TIM1_TRGO	オンチップタイマからの内部信号	0001
TIM2_TRGO	オンチップタイマからの内部信号	0010
TIM4_TRGO	オンチップタイマからの内部信号	0011
TIM5_TRGO	オンチップタイマからの内部信号	0100
TIM6_TRGO	オンチップタイマからの内部信号	0101
TIM7_TRGO	オンチップタイマからの内部信号	0110
TIM8_TRGO	オンチップタイマからの内部信号	0111
TIM15_TRGO	オンチップタイマからの内部信号	1000
HRTIM1_DACTRG1	オンチップタイマからの内部信号	1001
HRTIM1_DACTRG2	オンチップタイマからの内部信号	1010
LPTIM1_OUT	オンチップタイマからの内部信号	1011
LPTIM2_OUT	オンチップタイマからの内部信号	1100
EXTI9	外部ピン	1101
予約済み	-	1110
予約済み	-	1111

26.3.8 DMA リクエスト

各 DAC チャネルは、DMA 機能を備えています。DAC チャネルの DMA リクエストは、2 つの DMA チャネルを使用して処理されます。

DAC DMA リクエストは、DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガでなく）が発生したときに生成されます。その場合、DAC_DHRx レジスタの値が DAC_DORxレジスタに転送されます。

デュアルモードでは、両方の DMAENx ビットがセットされている場合、2 つの DMA リクエストが生成されます。1 つの DMA リクエストしか必要ない場合には、対応する DMAENx ビットだけをセットしてください。これにより、アプリケーションは 1 つの DMA リクエストと一意な DMA チャネルを使用して、両方の DAC チャネルをデュアルモードで管理することがきます。

DMA アンダーラン

DAC DMA リクエストはキューされないので、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に 2 番目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタの DMA チャネル x アンダーランフラグ DMAUDRx がセットされてエラー状態を報告します。DAC チャネル x は、古いデータを変換し続けます。

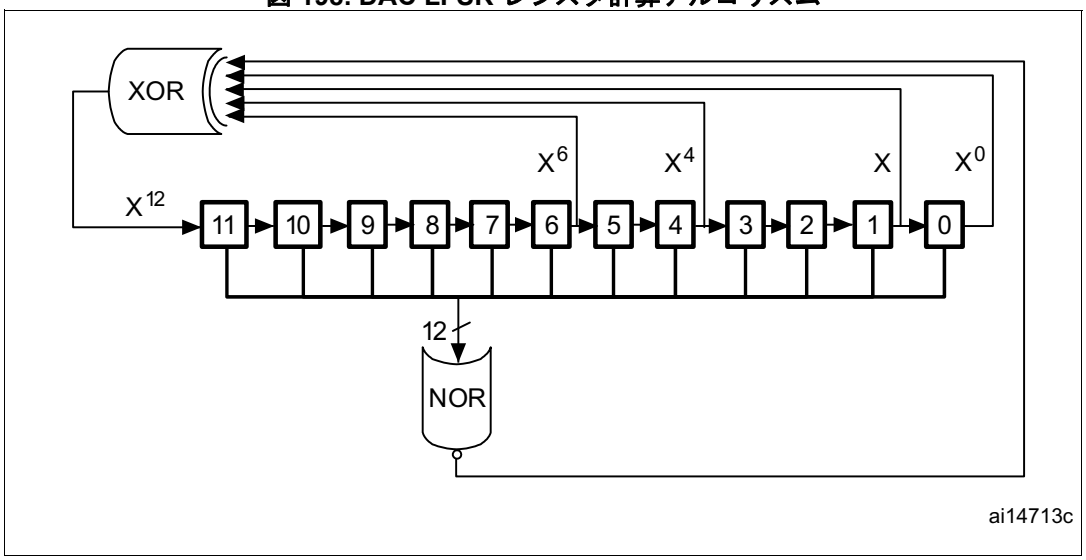
ソフトウェアでは、“1”を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし､DMA と DAC のチャネル x を再初期化して転送を正しくリスタートさせてください。また、DAC トリガ変換周波数を変更するか DMA の負荷を軽減して、新しい DMA アンダーランを回避してください。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャネルx では、DAC_CR レジスタの対応する DMAUDRIEx ビットが有効にされた場合、割り込みも生成されます。

26.3.9 ノイズ生成

リニアフィードバックシフトレジスタ（LFSR）を使用して、可変振幅の擬似ノイズを生成することができます。DAC ノイズ生成を選択するには、WAVEx[1:0] に“01”をセットします。LFSRにプリロードされる値は 0xAAA です。このレジスタは、各トリガイベントの 3 dac_pclk クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図 198. DAC LFSR レジスタ計算アルゴリズム

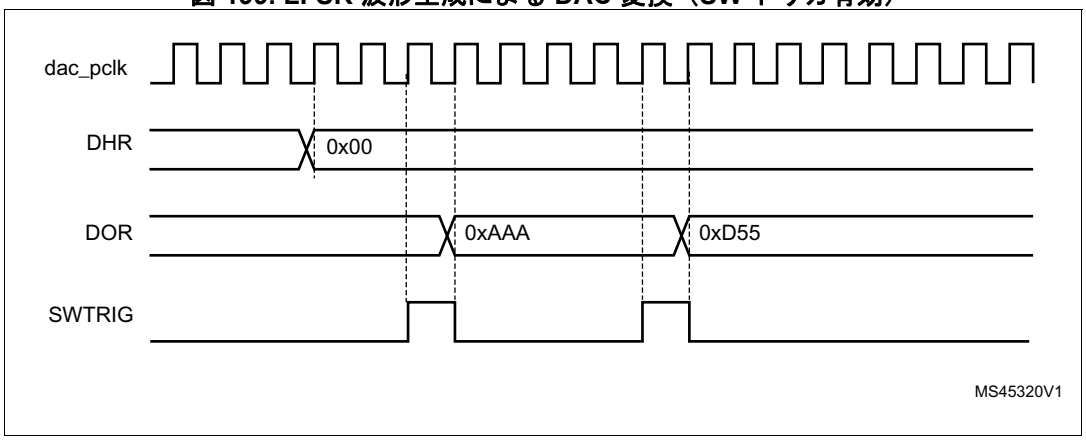


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算され、DAC_DORx レジスタに転送されます。

LFSR が 0x0000 の場合、“1”がインジェクトされます（アンチロックアップメカニズム）。

WAVEx[1:0] ビットをリセットすることによって、LFSR 波形生成をリセットできます。

図 199. LFSR 波形生成による DAC 変換（SW トリガ有効）



注： ***ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。***

26.3.10 三角波生成

DC または低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC 三角波生成を選択するには、WAVEx[1:0] を“10” にセットします。振幅は、DAC_CR レジスタのMAMPx[3:0] ビットを介して設定されます。内部三角波カウンタは、各トリガイベントの 3 dac_pclk クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしに DAC_DHRx レジスタに加えられ、合計は DAC_DORx レジスタに転送されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは 0 にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波生成をリセットすることができます。

図 200. DAC 三角波生成

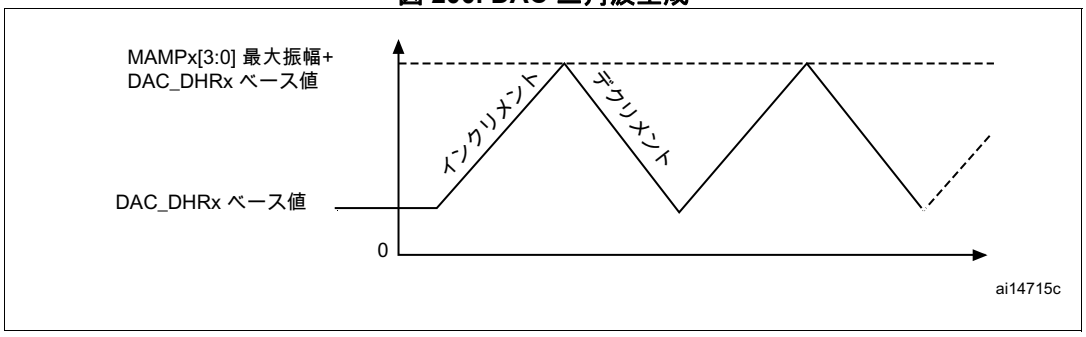
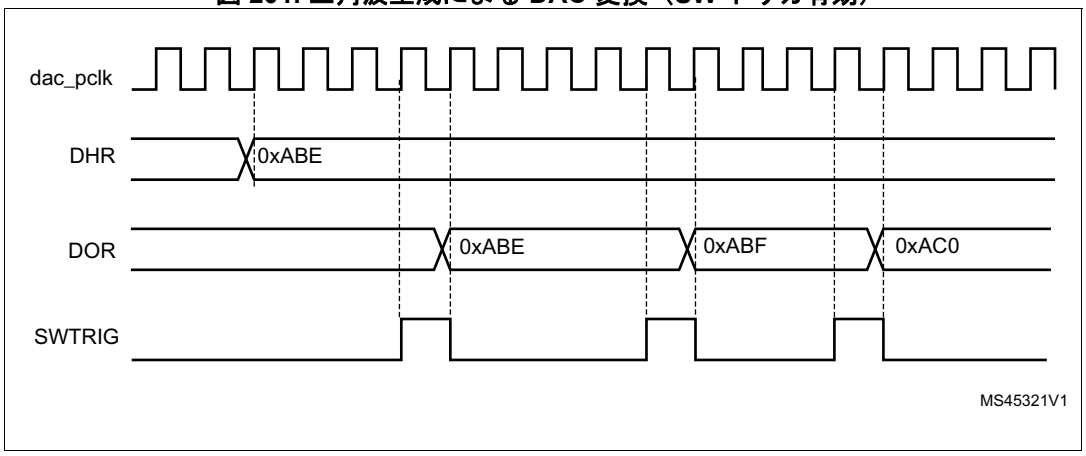


図 201. 三角波生成による DAC 変換（SW トリガ有効）



注： ***ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。***

DAC を有効にするには、その前に MAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

26.3.11 DAC チャネルモード

各 DAC チャネルは、通常モードまたはサンプルおよびホールドモードのいずれかに設定できます。高駆動機能を許可するために、出力バッファを有効にできます。出力バッファを有効にする前に、電圧オフセットを較正する必要があります。この較正は出荷時に実施され（リセット後にロードされ）、アプリケーション動作中にソフトウェアで調整できます。

通常モード

通常モードでは、バッファ状態の変更と DAC_OUTx ピン相互接続による 4 つの組み合わせがあります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 000：DAC を外部ピンに接続
- 001：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 010：DAC を外部ピンに接続
- 011：DAC をオンチップペリフェラルに接続

サンプルおよびホールドモード

サンプルおよびホールドモードでは、DAC コアがトリガ変換でデータを変換してから、コンデンサで変換された電圧を保持します。変換しない場合、DAC コアおよびバッファはサンプル間で完全にオフになり、DAC 出力はトライステートになるため、全体の消費電力を低減します。それぞれ新しい変換を行う前に、新しい安定時間（バッファ状態に応じて $T_{stab-BON}$ または $T_{stab-BOFF}$）が必要になります。

このモードでは、DAC コアならびにすべての対応するロジックとレジスタは、dac_pclk クロックに加えてロースピードクロック（lsi_ck：ロースピード内部オシレータ）によっても駆動されますので、STOP モードなどの超低電力モードで DAC チャネルを使用できます。

サンプル／ホールドモードの動作は、3 つのフェーズに分けられます。

1. サンプルフェーズ : サンプル／ホールド要素が対象の電圧までチャージされます。チャージ時間はコンデンサ値（ユーザによって選択された内部または外部）によって異なります。サンプリング時間は、DAC_SHSRx レジスタの TSAMx[9:0] ビットで設定されます。TSAMx[9:0] ビットの書き込み中、両方のクロックドメイン（APB およびロースピードクロック）を同期するために DAC_SR レジスタの BWSTx ビットが 1 にセットされ、DAC チャネル動作中にソフトウェアによってサンプルフェーズの値を変更できるようになります。
2. ホールドフェーズ : DAC 出力チャネルはトライステートになり、DAC コアおよびバッファはオフになって消費電流を低減します。ホールド時間は、DAC_SHHR レジスタの THOLDx[9:0] ビットで設定されます。
3. リフレッシュフェーズ : リフレッシュ時間は、DAC_SHRR レジスタの TREFx[7:0] ビットで設定されます。

上記 3 つのフェーズのタイミングは、lsi_ck クロックの単位で行われます。たとえば、350µs のサンプル時間、2ms のホールド時間、100µs のリフレッシュ時間を設定するために、lsi_ck ～32KHz を選択するとします。

12 サイクルがサンプルフェーズで必要になります。TSAMx[9:0] = 11 に設定します。62 サイクルがホールドフェーズで必要になります。THOLDx[9:0]= 62 に設定します。4 サイクルがリフレッシュ時間で必要になります。TREFx[7:0] = 4 に設定します。

この例では、消費電力が通常モードに対してほぼ 15分の1に低減されます。

適切なサンプルおよびリフレッシュタイミングを計算する公式は、以下の表に記載しており、ホールド時間はリーク電流によって異なります。

表 210. サンプルおよびリフレッシュタイミング

バッファ状態	$t_{sampling}$ (1)(3)	$t_{refresh}$ (2)(3)
有効化	$T_{stab-BON} + (10*R_{BON}*C_{load})$	$T_{stab-BON} + (R_{BON}*C_{load})*ln(2*Nlsb)$
無効化	$T_{stab-BOFF} + (10*R_{BOFF}*Cload)$	$T_{stab-BOFF} + (R_{BOFF}*Cload)*ln(2*Nlsb)$

注 : 上記の公式で、½ LSBエラー精度での目標のコード値までの整定には、12 ビットの分解能の場合、10 定数時間が必要です。8 ビット分解能の場合、整定時間は 7 定数時間です。

*ホールドフェーズ中に許容される電圧低下「Vd」は、コンデンサが出力リーク電流で放電した後の LSB の数で表されます。½ LSB エラー精度で目標のコード値まで整定し直すには、DAC の ln(2*Nlsb) 定数時間が必要です。*

パラメータ$T_{stab-BON}$、$T_{stab-BOFF}$、R_{BON}およびR_{BOFF}はデータシートで指定されています。

出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例

注 : 以下に使用される値は、例示のみを目的としています。製品データについては、製品データシートを参照してください。

C_{load} = 100 nF

V_{DDA} = 3.0 V

サンプリングフェーズ :

$t_{sampling}$ = 7 µs + (10 * 2000 * 100 * 10^{-9}) = 2.007 ms

（ここで、$T_{stab-BON}$ = 7 µs, RBON = 2 kΩ）

リフレッシュフェーズ：

$t_{refresh}$ = 7 µs + (2000 * 100 * 10^{-9}) * ln(2*10) = 606.1 µs

（ここで、N_{lsb} = 10（ホールドフェーズ中は 10 LSB 低下））

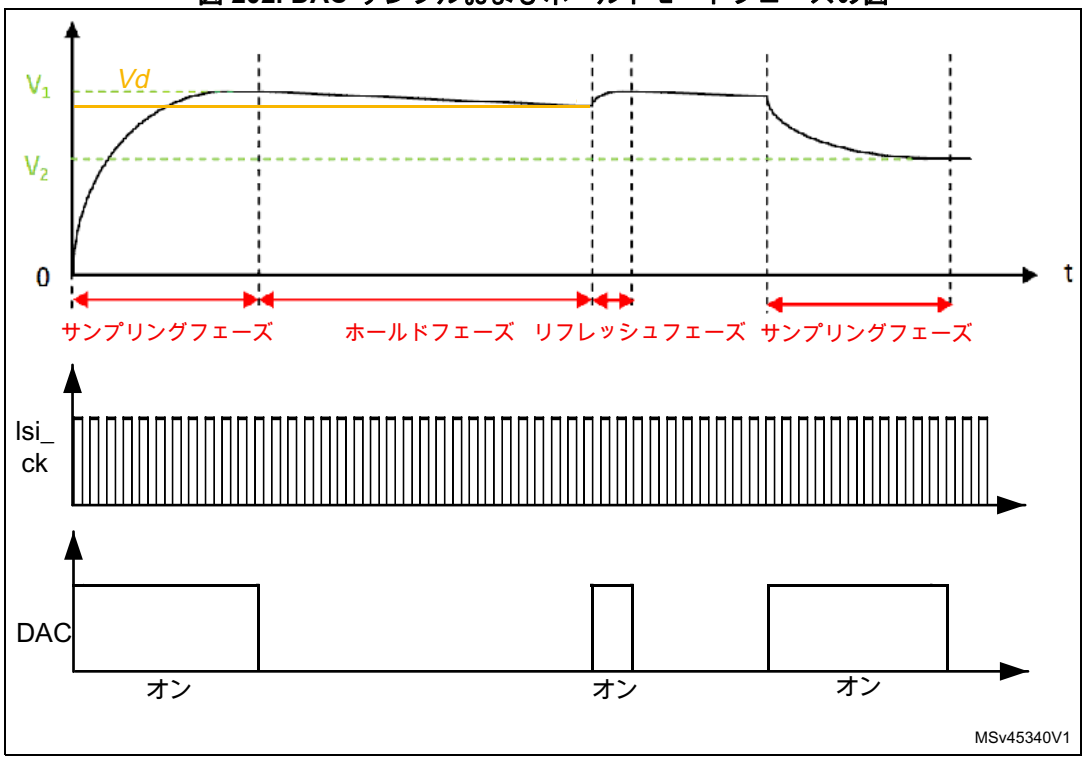
ホールドフェーズ：

$D_V = i_{leak} * t_{hold} / C_{load}$ = 0.0073 V（3 V で 12 ビットの 10 LSB）

i_{leak} = 150 nA（すべての温度範囲の IO リークで最悪のケース）

t_{hold} = 0.0073 * 100 * 10^{-9} / (150 * 10^{-9}) = 4.867 ms

図 202. DAC サンプルおよびホールドモードフェーズの図



通常モードと同様に、サンプルおよびホールドモードには個別の設定があります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 100：DAC を外部ピンに接続
- 101：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 110：DAC を外部ピンとオンチップペリフェラルに接続
- 111：DAC をオンチップペリフェラルに接続

DAC_MCR レジスタの MODEx[2:0] ビットが 111 と等しい場合、内部コンデンサ「$C_{loadint}$」は、DAC コアの電圧出力を保持し、オンチップペリフェラルに対して駆動します。

すべてのサンプルおよびホールドフェーズは割り込み可能で、DAC_DHRx のあらゆる変更によって新しいサンプルフェーズが即座にトリガされます。

表 211. チャネル出力モードの概要

MODEx[2:0]			モード	バッファ	出力接続
0	0	0	通常モード	有効	外部ピンに接続
0	0	1			外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
0	1	0		無効	外部ピンに接続
0	1	1			オンチップペリフェラルに接続（例：コンパレータ）
1	0	0	サンプルおよびホールドモード	有効	外部ピンに接続
1	0	1			外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
1	1	0		無効	外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
1	1	1			オンチップペリフェラルに接続（例：コンパレータ）

26.3.12 DAC 出力バッファ較正

N ビットのデジタルアナログコンバータ（DAC）の転送関数は次のとおりです。

$$V_{out} = ((D/2^{N-1}) \times G \times V_{ref}) + V_{OS}$$

ここで、VOUT はアナログ出力、D はデジタル入力、G はゲイン、V_{ref} は公称フルスケール電圧、V_{os} はオフセット電圧です。理想的な DAC チャネルでは、G = 1、V_{os} = 0 となります。

出力バッファの特性によって、オフセット電圧が部分的に異なり、アナログ出力で絶対的なオフセットエラーをもたらす場合があります。V_{os} を補正するには、トリミング技法による較正が必要です。

この較正は、DAC チャネル x がバッファ有効状態で動作している場合（MODEx[2:0] = 000b または 001b または 100b または 101b）のみ有効です。バッファがオフのときに他のモードで適用される場合は影響しません。較正中は次のようになります。

- バッファ出力がピンの内部／外部接続から切断され、トライステートモード（ハイインピーダンス）になります。
- バッファがミドルコード値 0x800 を検出して、内部ブリッジで VREF+/2 信号と比較するためにコンパレータとして機能し、比較結果（CAL_FLAGx ビット）に応じて出力信号を 0 または 1 にトグルします。

次の 2 つの較正技法があります。

- 出荷時トリミング（常に有効）

DAC バッファオフセットが出荷時にトリミングされます。DAC_CCR レジスタの OTRIMx[4:0] ビットのデフォルト値は出荷時トリミング値となり、DAC デジタルインタフェースがリセットされるとロードされます。

- ユーザトリミング

動作条件が出荷時トリミング条件と異なる場合、特に VDD/VDDA 電圧、温度、VREF+ の値が変化する場合に、ユーザトリミングはソフトウェアによって適用時にいつでも実行できます。

注： ***出荷時トリミング条件の詳細については、データシートを参照してください。***

また、VDD/VDDA が削除される場合（デバイスが STANDBY モードや VBAT モードに移行する場合など）、較正が必要になります。

ユーザトリミング較正を実行する手順は次のとおりです。

1. DAC チャネルがアクティブな場合、DAC_CR の ENx ビットに 0 を書き込み、チャネルを無効にします。
2. DACx_MCR レジスタに MODEx[2:0] = 000b または 001b または 100b または 101b を書き込んで、バッファが有効なモードを選択し
3. DACx_CR レジスタの CENx ビットを 1 にセットして、DAC チャネル x の較正を開始します。
4. トリミングアルゴリズムを適用します。
   a) OTRIMx[4:0] ビットにコードを書き込んで、00000b で開始します。
   b) tOFFTRIMmax の遅延を待ちます。
   c) DACx_SR の CAL_FLAGx ビットが 1 にセットされているか確認します。
   d) CAL_FLAGx が 1 にセットされている場合、トリミングコード OTRIMx[4:0] が検索され、出力値を補正する操作中に使用されます。それ以外は OTRIMx[4:0] をインクリメントし、（a）から（d）のサブステップを繰り返します。

ソフトウェアアルゴリズムによって、逐次比較や二分岐法を使用して、より早く OTRIMx[4:0] ビットの内容を計算してセットできます。

CAL_FLAGx ビットの遷移／トグルは、オフセットが正しく補正されたことを示し、対応するトリミングコードを DAC_CCR レジスタの OTRIMx[4:0] ビットに保持しておく必要があります。

注： *tOFFTRIMmax の遅延は、正しい値を取得するために DAC_SR レジスタでの OTRIMx[4:0] ビットの書き込みと CAL_FLAGx ビットの読み出しの間に行う必要があります。このパラメータは、データシートの電気的特性のセクションで指定されています。*

STANDBY モードおよび VBAT モードに頻繁に移行しつつ、デバイス動作中に VDD/VDDA、VREF+ および温度の条件が変化しない場合、ソフトウェアはフラッシュまたはバックアップレジスタの最初のユーザ較正に見られる OTRIMx[4:0] ビットを保存する場合があります。デバイスの電源が復帰したときに直接それらに読み書きを行うために、新しい較正時間を待機しなくて済むようになります。

26.3.13 デュアル DAC チャネル変換（使用できる場合）

同時に 2 つの DAC チャネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、および DHR12LD の 3 つのデュアルレジスタが搭載されています。両方の DAC チャネルを同時に駆動するには、一意なレジスタアクセスが必要です。

2 つの DAC チャネルとこれらのデュアルレジスタを使用することで、11 の変換モードが使用可能です。すべての変換モードは、必要な場合には、別の DHRx レジスタを使っても利用できます。

すべてのモードについて、以下の節で説明します。

波形生成なしの独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、DHR1 レジスタが DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DHR2 レジスタが DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。

1 つの LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
- 目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）に、デュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMP1[3:0] ビットと MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“1x”に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“1x”に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

同時ソフトウェア開始

DAC をこの変換モードに設定するには、次の手順が必要です。

- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）に、デュアル DAC チャネルデータをロードします。

この設定では、1 dac_pclk クロックサイクル後に、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます。

波形生成なしの同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）に、デュアル DAC チャネルデータをロードします。

トリガが発生すると、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます（3 dac_pclk クロックサイクル後）。

1 つの LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
- 目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）に、デュアル DAC チャネルデータをロードします。

トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。同時に、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“01”に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。
同時に、MAMP2[3:0] で設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“1x”に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、同じ三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR 1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。
同時に、同じ三角波振幅を持つ DAC チャネル2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[3:0] および TSEL2[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“1x”に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つ DAC チャネル2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

26.4 DAC 低電力モード

表 212. 低電力モードが DAC に与える影響

モード	説明
SLEEP	影響はありません。DAC は DMA とともに使用できます。
STOP	サンプルホールドモードが lsi_ck クロックを使用して選択されている場合、DAC はスタティック出力値でアクティブなままになります。
STANDBY	DAC ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

26.5 DAC レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1* を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

26.5.1 DAC 制御レジスタ（DAC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2[1:0]		TSEL23	TSEL22	TSEL21	TSEL20	TEN2	EN2
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1[1:0]		TSEL13	TSEL12	TSEL11	TSEL10	TEN1	EN1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CEN2**：DAC チャネル 2 較正有効化

このビットは、DAC チャネル 2 較正を有効化／無効化するためにソフトウェアによってセット／クリアされます。DAC_CR でビット EN2=0 の場合のみ（DAC チャネルが無効化された場合のみ較正モードに移行/終了できます）書き込み可能で、それ以外の書き込み操作は無視されます。

0：DAC チャネル 2 は通常動作モードです。

1：DAC チャネル 2 は較正モードです。

ビット 29 **DMAUDRIE2**：DAC チャネル2 DMA アンダーラン割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA アンダーラン割り込みは無効です。

1：DAC チャネル2 DMA アンダーラン割り込みは有効です。

ビット 28 **DMAEN2**：DAC チャネル2 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA モードは無効です。

1：DAC チャネル2 DMA モードは有効です。

ビット 27:24 **MAMP2[3:0]**：DAC チャネル2 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。
0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。
0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。
0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。
0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。
0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。
0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。
0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。
1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。
1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。
1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。
? 1011：LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 23:22 **WAVE2[1:0]**：DAC チャネル2 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／リセットされます。

00：波形生成は無効です。
01：ノイズ波生成は有効です。
1x：三角波生成は有効です。

注：　ビット TEN2＝1（DAC チャネル2 トリガ有効）の場合のみ使用されます。

ビット 21:18 **TSEL2[3:0]**：DAC チャネル2 トリガ選択

これらのビットは、DAC チャネル2 をトリガするために使用される外部イベントを選択します。

0000：ソフトウェアトリガ
0001：TIM1_TRGO
0010：TIM2_TRGO
0011：TIM4_TRGO
0100：TIM5_TRGO
0101：TIM6_TRGO
0110：TIM7_TRGO
0111：TIM8_TRGO
1000：TIM15_TRGO
1001：HRTIM1_DACTRG1
1010：HRTIM1_DACTRG2
1011：LPTIM1_OUT
1100：LPTIM2_OUT
1101：EXTI9
1110：予約済み
1111：予約済み

注：　ビット TEN2 = 1（DAC チャネル2 トリガ有効）の場合のみ使用されます。

ビット 17 **TEN2**：DAC チャネル2 トリガイネーブル

このビットは、DAC チャネル2 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 トリガは無効であり、DAC_DHRx レジスタに書き込まれたデータは、1 dac_pclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。
1：DAC チャネル 2 トリガは有効であり、DAC_DHRx レジスタから転送されたデータは、3 dac_pclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。

注：　ソフトウェアトリガが選択されているときには、DAC_DHRx から DAC_DOR2 レジスタへの転送には、わずか 1 dac_pclk クロックサイクルで行われます。

ビット 16 **EN2**：DAC チャネル2 イネーブル

このビットは、DAC チャネル2 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 は無効です。

1：DAC チャネル2 は有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CEN1**：DAC チャネル 1 較正有効化

このビットは、DAC チャネル 1 較正を有効化／無効化するためにソフトウェアによってセット／クリアされます。DAC_CR でビット EN1=0 の場合のみ（DAC チャネルが無効化された場合のみ較正モードに移行/終了できます）書き込み可能で、それ以外の書き込み操作は無視されます。

0：DAC チャネル 1 は通常動作モードです。

1：DAC チャネル 1 は較正モードです。

ビット 13 **DMAUDRIE1**：DAC チャネル 1 DMA アンダーラン割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA アンダーラン割り込みは無効です。

1：DAC チャネル 1 DMA アンダーラン割り込みは有効です。

ビット 12 **DMAEN1**：DAC チャネル 1 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA モードは無効です。

1：DAC チャネル 1 DMA モードは有効です。

ビット 11:8 **MAMP1[3:0]**：DAC チャネル 1 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。

0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。

0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。

0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。

0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。

0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。

0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。

0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。

1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。

1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。

1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。

? 1011：LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 7:6 **WAVE1[1:0]**：DAC チャネル 1 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：波形生成は無効です。

01：ノイズ波生成は有効です。

1x：三角波生成は有効です。

注： ビット TEN1 ＝ 1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 5:2 **TSEL1[3:0]**：DAC チャネル 1 トリガ選択

これらのビットは、DAC チャネル 1 をトリガするために使用される外部イベントを選択します。

0000：ソフトウェアトリガ
0001：TIM1_TRGO
0010：TIM2_TRGO
0011：TIM4_TRGO
0100：TIM5_TRGO
0101：TIM6_TRGO
0110：TIM7_TRGO
0111：TIM8_TRGO
1000：TIM15_TRGO
1001：HRTIM1_DACTRG1
1010：HRTIM1_DACTRG2
1011：LPTIM1_OUT
1100：LPTIM2_OUT
1101：EXTI9
1110：予約済み
1111：予約済み

注： ビット TEN1 ＝ 1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 1 **TEN1**：DAC チャネル 1 トリガイネーブル

このビットは、DAC チャネル 1 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 トリガは無効であり、DAC_DHRx レジスタに書き込まれたデータは、1 dac_pclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

1：DAC チャネル 1 トリガは有効であり、DAC_DHRx レジスタから転送されたデータは、3 dac_pclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DOR1 レジスタへの転送は、わずか 1 dac_pclk クロックサイクルで行われます。

ビット 0 **EN1**：DAC チャネル 1 イネーブル

このビットは、DAC チャネル 1 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 は無効です。
1：DAC チャネル 1 は有効です。

26.5.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR）

アドレスオフセット：0x04
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
														w	w

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWTRIG2**：DAC チャネル2 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR2 レジスタの値が DAC_DOR2 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_pclk クロックサイクル後）。

ビット 0 **SWTRIG1**：DAC チャネル 1 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR1 レジスタの値が DAC_DOR1 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_pclk クロックサイクル後）。

26.5.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1）

アドレスオフセット：0x08
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

26.5.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

26.5.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

26.5.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[7:0]								DACC1DHR[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

26.5.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）

アドレスオフセット：0x2C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DOR[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DOR[11:0]**：DAC チャネル 1 データ出力

これらのビットは読み出し専用であり、DAC チャネル 1 のデータ出力を含みます。

26.5.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）

アドレスオフセット：0x30

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DOR[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DOR[11:0]**：DAC チャネル2 データ出力

これらのビットは読み出し専用であり、DAC チャネル2 のデータ出力を含みます。

26.5.14 DAC ステータスレジスタ（DAC_SR）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWST2	CAL_FLAG2	DMAUDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BWST1	CAL_FLAG1	DMAUDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													

ビット 31 **BWST2**：DAC チャネル 2 ビジー書き込みサンプル時間フラグ

このビットは、ソフトウェアが DAC_SHSR2 レジスタに書き込むたびにサンプルおよびホールドモードを有効化した直後にシステムによってセットされます。DAC_SHSR2 の書き込み操作が完了するとハードウェアによってクリアされます。（同期に約 3 lsi_ck 周期かかります。）

0：進行中の DAC_SHSR2 の書き込み操作はありません。DAC_SHSR2 は書き込み可能です。

1：進行中の DAC_SHSR2 の書き込み操作があります。DAC_SHSR2 は書き込みできません。

ビット 30 **CAL_FLAG2**：DAC チャネル 2 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 29 **DMAUDR2**：DAC チャネル2 DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル2 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル2 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル2 変換を駆動しています）。

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **BWST1**：DAC チャネル 1 ビジー書き込みサンプル時間フラグ

このビットは、ソフトウェアが DAC_SHSR1 レジスタに書き込むたびにサンプルおよびホールドモードを有効化した直後にシステムによってセットされます。DAC_SHSR1 の書き込み操作が完了するとハードウェアによってクリアされます。（同期に約 3 lsi_ck 周期かかります。）

0：進行中の DAC_SHSR1 の書き込み操作はありません。DAC_SHSR1 は書き込み可能です。

1：進行中の DAC_SHSR1 の書き込み操作があります。DAC_SHSR1 は書き込みできません。

ビット 14 **CAL_FLAG1**：DAC チャネル 1 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 13 **DMAUDR1**：DAC チャネル 1 の DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル 1 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル1 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 1 変換を駆動しています）。

ビット 12:0 予約済みであり、リセット値に保持する必要があります。

26.5.15 DAC 較正制御レジスタ（DAC_CCR）

アドレスオフセット：0x38

リセット値：0x00XX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2[4:0]				
											rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1[4:0]				
											rw				

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 OTRIM2[4:0]：DAC チャネル 2 オフセットトリミング値

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 OTRIM1[4:0]：DAC チャネル 1 オフセットトリミング値

26.5.16 DAC モード制御レジスタ（DAC_MCR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2[2:0]		
													rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1[2:0]		
													rw		

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **MODE2[2:0]：**DAC チャネル 2 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN2=0 および CEN2 =0）のみ書き込み可能です。EN2=1 または CEN2 =1 の場合、書き込み操作は無視されます。

このビットは、DAC チャネル 2 モードを選択するために、ソフトウェアによってセット／クリアされます。

– 通常モードの DAC チャネル 2

000：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 2 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

– サンプルおよびホールドモードの DAC チャネル 2

100：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 2 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **MODE1[2:0]**：DAC チャネル 1 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN1=0 および CEN1 =0）のみ書き込み可能です。EN1=1 または CEN1 =1 の場合、書き込み操作は無視されます。

このビットは、DAC チャネル 1 モードを選択するために、ソフトウェアによってセット／クリアされます。

– 通常モードの DAC チャネル 1

000：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 1 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

– サンプルおよびホールドモードの DAC チャネル 1

100：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 1 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

26.5.17 DAC サンプルおよびホールドのサンプル時間レジスタ 1（DAC_SHSR1）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE1[9:0]**：DAC チャネル 1 サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 1 が無効である場合や通常動作中にも書き込み可能です。後者の場合、DAC_SR レジスタの BWSTx がローのときのみ書き込み可能で、BWSTx=1 の場合の書き込み操作は無視されます。

注： ***これは、サンプルフェーズを実行する lsi_ckクロックの数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x T lsi_ck です。***

26.5.18 DAC サンプルおよびホールドのサンプル時間レジスタ 2（DAC_SHSR2）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE2[9:0]**：DAC チャネル 2 サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 2 が無効である場合や通常動作中にも書き込み可能です。後者の場合、DAC_SR レジスタの BWSTx がローのときのみ書き込み可能で、BWSTx=1 の場合の書き込み操作は無視されます。

注： ***これは、サンプルフェーズを実行する lsi_ckクロックの数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x T lsi_ck です。***

26.5.19 DAC サンプルおよびホールドのホールド時間レジスタ（DAC_SHHR）

アドレスオフセット：0x48

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	THOLD2[9:0]									
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	THOLD1[9:0]									
						rw									

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **THOLD2[9:0]**：DAC チャネル 2 ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x T lsi_ck

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **THOLD1[9:0]**：DAC チャネル 1 ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x T lsi_ck

注： ***これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタでビット ENx=0 および CENx =0）のみ書き込み可能です。ENx=1 または CENx=1 の場合、書き込み操作は無視されます。***

26.5.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）

アドレスオフセット：0x4C

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2[7:0]							
								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1[7:0]							
								rw							

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **TREFRESH2[7:0]**：DAC チャネル 2 リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x T lsi_ck

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TREFRESH1[7:0]**：DAC チャネル 1 リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x T lsi_ck

注： ***これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタでビット ENx=0 および CENx =0）のみ書き込み可能です。ENx=1 または CENx=1 の場合、書き込み操作は無視されます。***

26.5.21 DAC オプションレジスタ（DAC_OR）

アドレスオフセット：0x50

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:0 予約済みであり、リセット値に保持する必要があります。

26.5.22 DAC レジスタマップ

表 213 に DAC レジスタの要約を示します。

表 213. DAC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DAC_CR	Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2[2:0]		TSEL23	TSEL22	TSEL21	TSEL20	TEN2	EN2	Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1[2:0]		TSEL13	TSEL12	TSEL11	TSEL10	TEN1	EN1
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	DAC_SWTRGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
	リセット値																															0	0
0x08	DAC_DHR12R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DAC_DHR12L1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x10	DAC_DHR8R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x14	DAC_DHR12R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x18	DAC_DHR12L2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	DAC_DHR8R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x20	DAC_DHR12RD	Res.	Res.	Res.	Res.	DACC2DHR[11:0]												Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x24	DAC_DHR12LD	DACC2DHR[11:0]												Res.	Res.	Res.	Res.	DACC1DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x28	DAC_DHR8RD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]								DACC1DHR[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	DAC_DOR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DOR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x30	DAC_DOR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DOR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x34	DAC_SR	BWST2	CAL_FLAG2	DMAUDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BWST1	CAL_FLAG1	DMAUDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0														0	0	0													

表 213. DAC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	DAC_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1[4:0]				
	リセット値												X	X	X	X	X												X	X	X	X	X
0x3C	DAC_MCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1[2:0]		
	リセット値														0	0	0														0	0	0
0x40	DAC_SHSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x44	DAC_SHSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x48	DAC_SHHR	Res.	Res.	Res.	Res.	Res.	Res.	THOLD2[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	THOLD1[9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	1							0	0	0	0	0	0	0	0	0	1
0x4C	DAC_SHRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1[7:0]							
	リセット値									0	0	0	0	0	0	0	1									0	0	0	0	0	0	0	1
0x50	DAC_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

27 電圧基準バッファ（VREFBUF）

27.1 概要

STM32H7x3 デバイスには電圧基準バッファが搭載されており、ADC や DAC の電圧基準や、VREF+ ピンを介した外部コンポーネントの電圧基準としても使用できます。

27.2 VREFBUF の機能説明

内部基準電圧バッファは次の 5 つの電圧をサポートしており、VREF_CSR レジスタの VRS ビットで設定します。

- VRS=000：約 2.5 V。これには、2.8 V 以上の V_{DDA} が必要です。
- VRS=001：約 2.048 V。これには、2.4 V 以上の V_{DDA} が必要です。
- VRS=010：約 1.8 V。これには、2.1 V 以上の V_{DDA} が必要です。
- VRS=011：約 1.5 V。これには、1.8 V 以上の V_{DDA} が必要です。

内部電圧基準は ENVR および HIZ ビットの設定に応じて 4 つの個別モードに設定できます。これらのモードを以下の表に示します。

表 214. VREF バッファモード

ENVR	HIZ	VREF バッファ設定
0	0	VREFBUF オフ： – V_{REF+} ピンが V_{SSA} にプルダウン
0	1	外部電圧基準モード（デフォルト値）： – VREFBUF オフ – V_{REF+} ピン入力モード
1	0	内部電圧基準モード： – VREFBUF オン – V_{REF+} ピンが VREFBUF バッファ出力に接続
1	1	ホールドモード： – VREFBUF オフ – V_{REF+} ピンフローティング。電圧が外部コンデンサで保持

VREFBUF_CSR レジスタの ENVR ビットをセットすることで VREFBUF を有効化した後、ユーザは電圧基準出力が期待値に達したことを示す VRR ビットがセットされるまで待つ必要があります。

27.3 VREFBUF レジスタ

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

アドレスオフセット：0x00

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRS[2:0]			VRR	Res.	HIZ	ENVR
									rw	rw	rw	r		rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **VRS[2:0]：**電圧基準スケール

これらのビットは、電圧基準バッファによって生成される値を選択します。

000：電圧基準が 2.5 V にセットされます。

001：電圧基準が 2.048 V にセットされます。

010：電圧基準が 1.8 V にセットされます。

011：電圧基準が 1.5 V にセットされます。

その他：予約済み

ビット 3 **VRR：**電圧基準バッファ レディ

0：電圧基準バッファ出力がレディ状態ではありません。

1：電圧基準バッファ出力がリクエストされたレベルに達しました。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HIZ：**ハイインピーダンスモード

このビットによって、V_{REF+} ピンを接続／切断するためのアナログスイッチを制御します。

0：V_{REF+} ピンが 電圧基準バッファ出力に内部接続されます。

1：V_{REF+} ピンをハイインピーダンスにします。

ENVR ビットの設定に応じたモード説明については、*表 214：VREF バッファモード*を参照してください。

ビット 0 **ENVR：**電圧基準バッファモード有効化

このビットは、電圧基準バッファモードの有効化に使用されます。

0：内部電圧基準モード無効化（外部電圧基準モード）

1：内部電圧基準モード（基準バッファ有効化またはホールドモード）有効化

27.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）

アドレスオフセット：0x04

リセット値：0x0000 00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TRIM[5:0]**：トリミングコード

これらのビットは、生産試験中にフラッシュメモリに格納されたトリミング値でリセットした後に自動的に初期化されます。これらのビットに書き込むことで、内部基準電圧バッファを調整できます。

27.3.3 VREFBUF レジスタマップ

次の表に、VREFBUF レジスタマップとリセット値を示します。

表 215. VREFBUF レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	VREFBUF_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRS[2:0]			VRR	Res.	HIZ	ENVR
	リセット値																										0	0	0	0		1	0
0x04	VREFBUF_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]						
	リセット値																											x	x	x	x	x	x

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

参考資料

28 コンパレータ（COMP）

28.1 概要

このデバイスには、2 つの超低電力コンパレータチャネル（COMP1 および COMP2）が内蔵されています。コンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低電力モードからのウェイクアップ
- アナログ信号調節
- タイマからの PWM 出力を組み合わせた場合のサイクルごとの電流制御ループ

28.2 COMP の主な機能

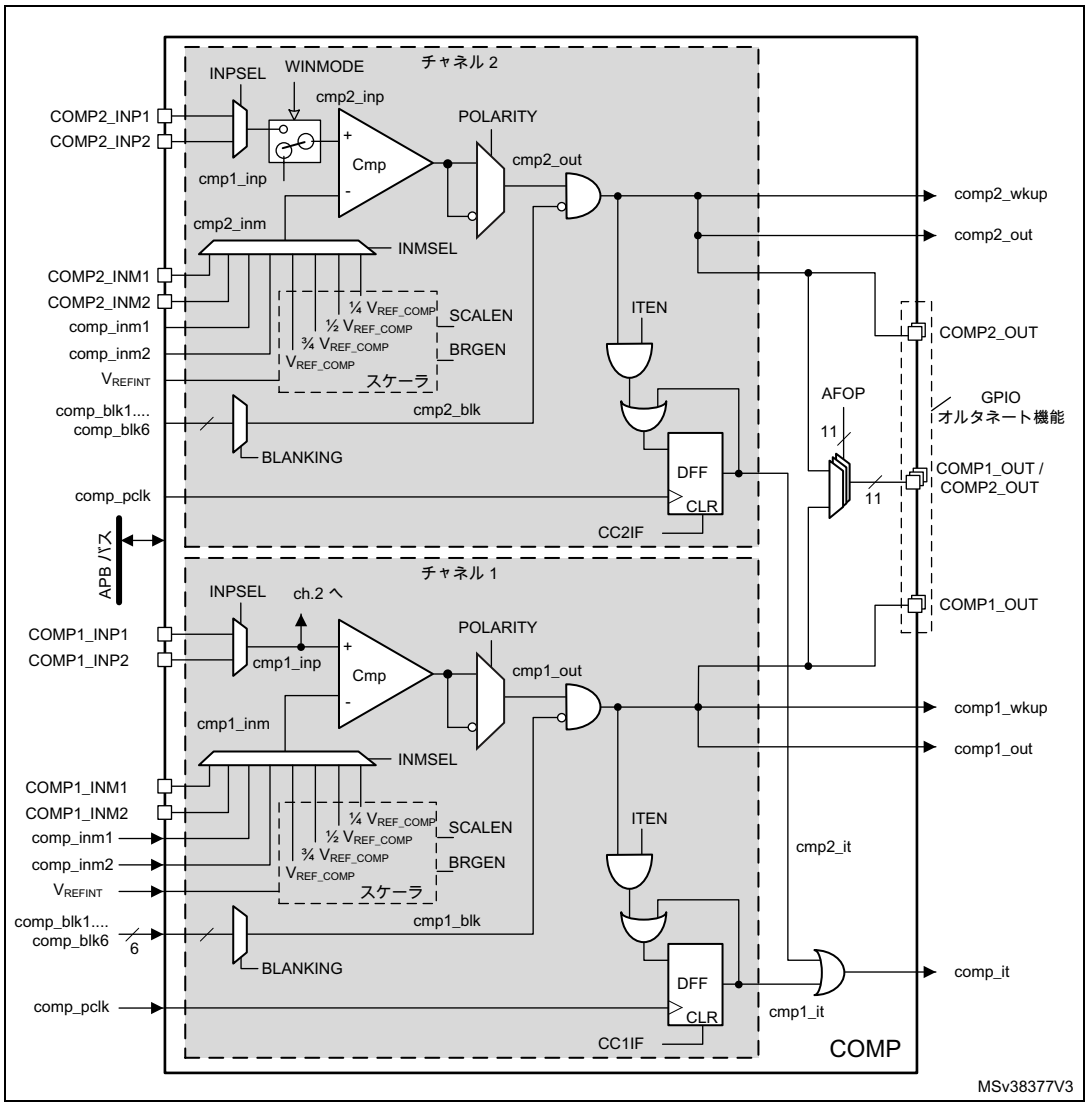
- 選択可能な反転アナログ入力：
  - I/O ピン（各チャネルで異なる）
  - DAC チャネル 1 およびチャネル 2 出力
  - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および 3 つの分圧比（1/4、1/2、3/4）
- 非反転アナログ入力として選択可能なチャネルあたり 2 本の I/O ピン
- プログラム可能なヒステリシス
- プログラム可能なスピード／消費
- I/O への出力マッピング
- 次のイベントをトリガするためのタイマ入力への出力先の変更：
  - キャプチャイベント
  - OCREF_CLR イベント（サイクルごとの電流制御）
  - 高速 PWM 停止のブレークイベント
- コンパレータ出力のブランキング
- ウィンドウコンパレータ
- SLEEP および STOP モードからのウェイクアップ（EXTI コントローラ経由）を備えた割り込み生成機能
- CPU への直接割り込み出力

28.3 COMP の機能説明

28.3.1 COMP ブロック図

コンパレータのブロック図を*図 203：コンパレータの機能ブロック図*に示します。

図 203. コンパレータの機能ブロック図



28.3.2 COMP ピンおよび内部信号

コンパレータ入力として使用される I/O は、GPIO レジスタのアナログモードで設定する必要があります。

コンパレータ出力は、オルタネート機能を通じて I/O に接続できます。製品データシートを参照してください。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- BKIN および BKIN2 入力を使用した PWM 信号の緊急停止
- タイマの ETR 入力を使用したサイクルごとの電流制御
- タイミング測定のための入力キャプチャ

コンパレータの出力は、内部と I/O ピンに同時に配信可能です。

表 216. COMP 入出力内部信号

信号名	信号タイプ	説明
comp_inm1	アナログ入力	両COMPチャネルに対する反転入力源：DAC チャネル 1
comp_inm2	アナログ入力	両COMPチャネルに対する反転入力源：DAC チャネル 2
comp_blk1	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM1 OC5
comp_blk2	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM2 OC3
comp_blk3	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM3 OC3
comp_blk4	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM3 OC4
comp_blk5	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM8 OC5
comp_blk6	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM15 OC1
comp_pclk	デジタル入力	両COMPチャネルに対する APB クロック
comp1_wkup	デジタル出力	COMP チャネル 1 ウェイクアップ出力
comp1_out	デジタル出力	COMP チャネル 1 出力
comp2_wkup	デジタル出力	COMP チャネル 2 ウェイクアップ出力
comp2_out	デジタル出力	COMP チャネル 2 出力
comp_it	デジタル出力	COMP 割り込み出力

表 217. COMP 入出力ピン

信号名	信号タイプ	説明
COMP1_INM1	アナログ入力	COMP チャネル 1 反転入力源 1（PB1）
COMP1_INM2	アナログ入力	COMP チャネル 1 反転入力源 2（PC4）
COMP1_INP1	アナログ入力	COMP チャネル 1 非反転入力源 1（PB0）
COMP1_INP2	アナログ入力	COMP チャネル 1 非反転入力源 2（PB2）
COMP2_INM1	アナログ入力	COMP チャネル 2 反転入力源 1（PE10）
COMP2_INM2	アナログ入力	COMP チャネル 2 反転入力源 2（PE7）
COMP2_INP1	アナログ入力	COMP チャネル 2 非反転入力源 1（PE9）
COMP2_INP2	アナログ入力	COMP チャネル 2 非反転入力源 2（PE11）
COMP1_OUT	デジタル出力	COMP チャネル 1 出力: *セクション 28.3.8: GPIO へのコンパレータ出力*参照
COMP2_OUT	デジタル出力	COMP チャネル 2 出力: *セクション 28.3.8: GPIO へのコンパレータ出力*参照

28.3.3 COMP のリセットおよびクロック

クロックコントローラによって提供される comp_pclk クロックは、APB クロックと同期しています。

注： ***重要：極性選択ロジックおよびポートへの出力先変更は、APB クロックから個別に機能します。これにより、コンパレータは STOP モードでも機能することができます。CPU の NVIC に接続されている割り込みラインは、動作に APB クロック（comp_pclk）を必要とします。APB クロックが存在しない場合には、割り込み信号 comp_it は生成されません。***

28.3.4 コンパレータのロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるようなアプリケーションでは、プログラムカウンタの破壊のような発生するおそれのある好ましくない変更から、コンパレータの設定を保護できます。

このような目的で、コンパレータの設定レジスタを書き込み保護することができます（読み出し専用）。

コンパレータチャネルの設定時には、その LOCK ビットを 1 に設定してください。こうすることで、コンパレータチャネルのレジスタセット全体に加えて共通 COMP_OR レジスタが、LOCK ビットを含めて読み出し専用となります。

書き込み保護は MCU リセットによってのみ解除可能です。

COMP_OR レジスタは、COMP_CFGR1 または COMP_CFGR2 の LOCK ビットでロックされます。

28.3.5 ウィンドウコンパレータ

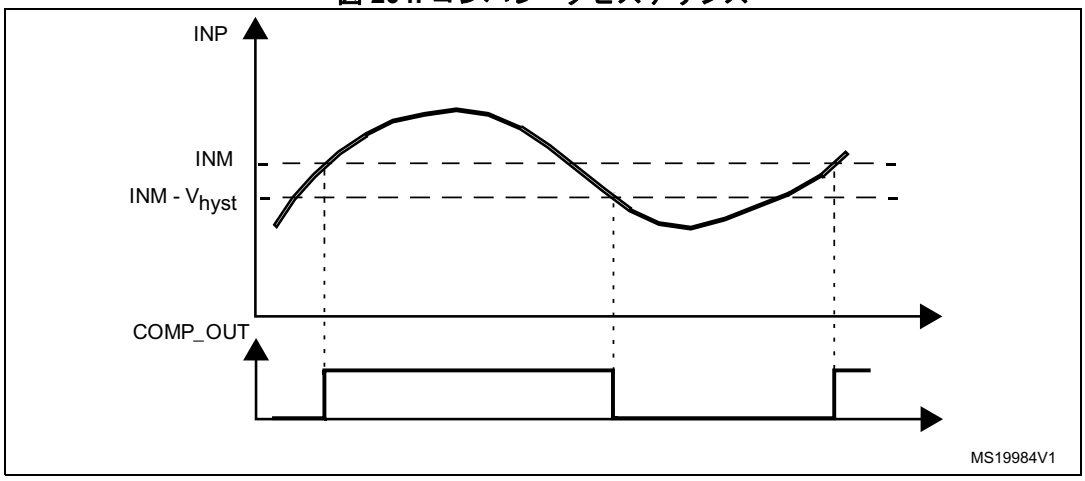
ウィンドウコンパレータの目的は、アナログ電圧を監視して、下限および上限閾値によって定義された指定の電圧範囲内にあるかどうか確認することです。

ウィンドウコンパレータには両方のCOMPチャネルが必要です。監視されるアナログ電圧は、コンパレータの非反転（正）入力に接続され、上限および下限閾値の電圧は、どちらかのコンパレータの反転（負）入力にそれぞれ接続されます。WINMODE ビットをイネーブルすることによって、COMP チャネル 2 の非反転入力を、COMP チャネル 1 の非反転入力に内部で接続できます。これにより、COMP チャネル 2 の入力ピンを他の目的のために取っておくことができます。*図 203：コンパレータの機能ブロック図*を参照してください。

28.3.6 ヒステリシス

コンパレータには、ノイズの多い信号での疑似出力遷移を避けるために、プログラム可能なヒステリシスが含まれています。不要な場合は、ヒステリシスを無効にすることができます（低電力モードを終了する場合など）。これにより、外部コンポーネントを使用してヒステリシス値を強制できます。

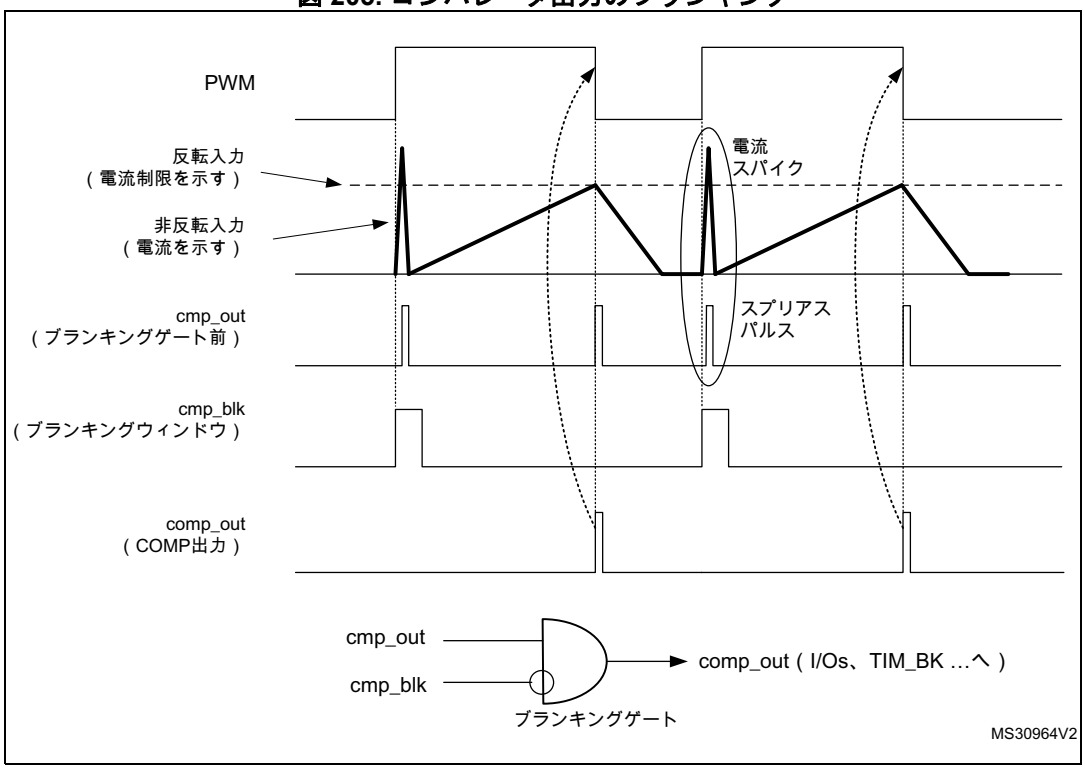
図 204. コンパレータヒステリシス



28.3.7 コンパレータの出力のブランキング機能

ブランキング機能の目的は、PWM 周期の開始時に短絡電流スパイクでトリップするために現在のレギュレーションを防ぐことです（通常は電源スイッチのアンチパラレルダイオードのリカバリ電流）。タイマ出力比較信号で定義されたブランキングウィンドウを使用します。選択可能ブランキング信号に関するレジスタの説明を参照してください。ブランキング信号は、*図 205*に図示されているような（COMP チャネル番号は示されていません）、comp_out から電流スパイクによるスプリアスパルスを除去するといった目的のために、内部コンパレータ出力をゲートします。

図 205. コンパレータ出力のブランキング



28.3.8 GPIO へのコンパレータ出力

コンパレータチャネルの COMP1_OUT 出力と COMP2_OUT 出力は、COMP_OR レジスタのビット [10:0] の AFOP フィールドならびに GPIO オルタネート機能を通じて、GPIO にマッピングされます。

表 218. GPIO に対する COMP1_OUT の割り当て

COMP1_OUT	オルタネート機能
PC5	AF13
PE12	AF13
PA6	AF10、AF12（タイマブレーク入力として使用可能）
PA8	AF12（タイマブレーク入力として使用可能）
PB12	AF13（タイマブレーク入力として使用可能）
PE6	AF11（タイマブレーク入力として使用可能）
PE15	AF13（タイマブレーク入力として使用可能）
PG2	AF11（タイマブレーク入力として使用可能）
PG3	AF11（タイマブレーク入力として使用可能）
PG4	AF11（タイマブレーク入力として使用可能）
PI1	AF11（タイマブレーク入力として使用可能）
PI4	AF11（タイマブレーク入力として使用可能）
PK2	AF10、AF11（タイマブレーク入力として使用可能）

表 219. GPIO に対する COMP2_OUT の割り当て

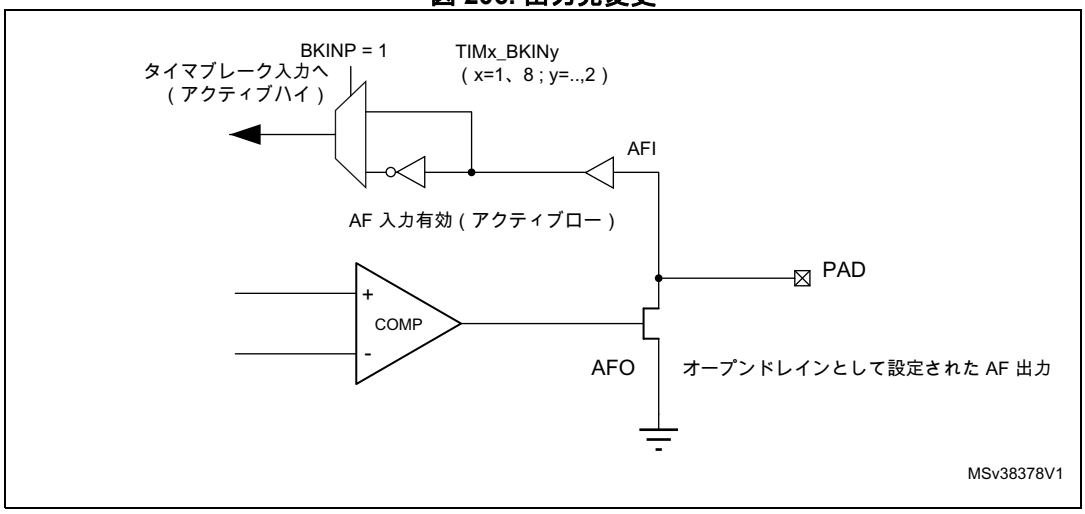
COMP2_OUT	オルタネート機能
PE8	AF13
PE13	AF13
PA6	AF10、AF12（タイマブレーク入力として使用可能）
PA8	AF12（タイマブレーク入力として使用可能）
PB12	AF13（タイマブレーク入力として使用可能）
PE6	AF11（タイマブレーク入力として使用可能）
PE15	AF13（タイマブレーク入力として使用可能）
PG2	AF11（タイマブレーク入力として使用可能）
PG3	AF11（タイマブレーク入力として使用可能）
PG4	AF11（タイマブレーク入力として使用可能）
PI1	AF11（タイマブレーク入力として使用可能）
PI4	AF11（タイマブレーク入力として使用可能）
PK2	AF10、AF11（タイマブレーク入力として使用可能）

いずれかのコンパレータチャネルのレジスタをロックすると、共通 COMP_OR レジスタがロックされるため、コンパレータチャネル両方の出力の GPIO に対する割り当ては、いずれかのチャネルのレジスタをロックする前に行う必要があります。

28.3.9 コンパレータ出力先変更

*図 206*に示すように、どちらの COMP チャネルの出力も、タイマブレーク入力（TIMx_BKIN または TIMx_BKIN2）へと出力先を変更できます。そのため、COMP チャネルの出力は、オルタネート機能でタイマブレーク入力としてプログラム可能な GPIO の 1 つに接続されています。*表 218*と*表 219*を参照してください。選択された GPIO は、オープンドレインモードで設定される必要があります。COMP 出力は、GPIO を通ってタイマブレーク入力に入ります。プルアップ抵抗を用いて、選択された GPIO は、コンパレータ出力との論理和が取られたタイマブレーク入力として使用できます。

図 206. 出力先変更



28.3.10 COMP 電力とスピードモード

COMP チャネルの電力消費と伝搬遅延を調節して、特定のアプリケーションに最適なトレードオフを実現させることができます。

COMP_CFGRx レジスタのビット PWRMODE[1:0] は次のようにプログラムできます：

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

28.4 COMP 低電力モード

表 220. 低電力モードでのコンパレータの動作

モード	説明
SLEEP	コンパレータへの影響はありません。 コンパレータ割り込みによって、デバイスは SLEEP モードから復帰します。
STOP	コンパレータへの影響はありません。 コンパレータ割り込みによって、デバイスは STOP モードから復帰します。

注： ***コンパレータは、内部基準電圧がオフされているときに、デバイスを SLEEP モードまたは STOP モードから抜けさせるために使用することはできません。***

28.5 COMP 割り込み

コンパレータを割り込み原因として使用するには、2 つの方法があります。

コンパレータの出力は拡張割り込み／イベントコントローラに内部的に接続されています。各コンパレータには専用の EXTI ラインがあり、割り込みまたはイベントを生成して、デバイスを低電力モードから抜けさせることができます。

コンパレータは、CPU の NVIC に対する割り込みラインも提供しています。この機能は、低遅延割り込みを処理するために CPU がアクティブであるときに使用されます。APB クロックが動作している必要があります。

28.5.1 EXTI ブロック経由の割り込み

詳細については、割り込みおよびイベントのセクションを参照してください。

EXTI ブロックを経由する COMPx 割り込みを有効にするシーケンスは次のとおりです：

1. 割り込みモードで comp_wkup 信号を受信する EXTI ラインを設定し、立ち上がり、立ち下がり、または両エッジの感度を選択し、EXTI ラインを有効にします。
2. 対応する EXTI ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
3. COMPx を有効にします。

表 221. 割り込み制御ビット

割り込みイベント	イベントフラグ	イネーブル制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
comp1_wkup	EXTI 経由	EXTI 経由	あり	あり	N/A
comp2_whup	EXTI 経由	EXTI 経由	あり	あり	N/A

28.5.2 CPU の NVIC 経由の割り込み

CPU の NVIC を経由する COMPx 割り込みを有効にするシーケンスは次のとおりです：

1. comp_it ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
2. COMP_CFGRx で ITEN を設定し、有効にします。
3. COMPx を有効にします。

表 222. 割り込み制御ビット

割り込みイベント	割り込みフラグ	イネーブル制御ビット	割り込みクリアビット	SLEEP モードの終了	STOP モードの終了
comp_it	C1IF in	COMP_CFGR1 の ITEN	CC1IF	あり（APB クロックにて）	なし
comp_it	C2IF in	COMP_CFGR2 の ITEN	CC2IF	あり（APB クロックにて）	なし

注： ***この割り込みを使用するには、APB クロックを有効にする必要があります。クロックが有効でない場合、割り込みは発生しません。***

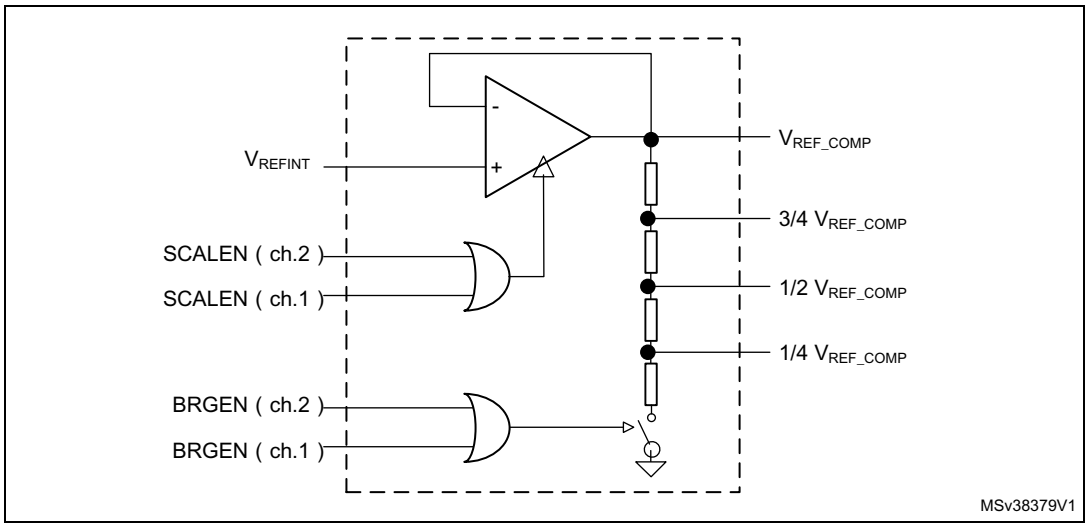
28.6 SCALER 機能

スケーラブロックは、コンパレータ入力に異なった電圧リファレンスレベルを与えるために使用できます。これは抵抗器ブリッジを駆動する増幅器をベースにしています。増幅器入力は内部電圧リファレンスに接続されています。

増幅器と抵抗器ブリッジは、別々に有効化できます。増幅器は、COMP_CFGRx レジスタの SCALEN ビットによって有効になります。抵抗器ブリッジは、COMP_CFGRx レジスタの BRGEN ビットによって有効になります。

抵抗器で分割された電圧を使用しないときには、消費電力削減のために抵抗器ブリッジを切り離すことができます。抵抗器ブリッジが切り離されると、1/4 V_{REF_COMP} と 1/2 V_{REF_COMP} と 3/4 V_{REF_COMP} の各レベルは V_{REF_COMP} と等しくなります。

図 207. スケーラブロック図



28.7 COMP レジスタ

28.7.1 コンパレータステータスレジスタ（COMP_SR）

COMP_SR は、コンパレータステータスレジスタです。

アドレスオフセット：0x00

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2IF	C1IF
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2VAL	C1VAL
														r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **C2IF**：COMP チャネル 2 割り込みフラグ
このビットは、COMP チャネル 2 出力がセットされているときに、ハードウェアによってセットされます。
このビットは、COMP_ICFR レジスタの CC2IF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

ビット 16 **C1IF**：COMP チャネル 1 割り込みフラグ
このビットは、COMP チャネル 1 出力がセットされているときに、ハードウェアによってセットされます。
このビットは、COMP_ICFR レジスタの CC1IF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **C2VAL**：COMP チャネル 2 出力ステータスビット
このビットは読み出し専用です。このビットは、POLARITY ビットと BLANKING ビットの影響を考慮して現在の COMP チャネル 2 出力を反映します。

ビット 0 **C1VAL**：COMP チャネル 1 出力ステータスビット
このビットは読み出し専用です。このビットは、POLARITY ビットと BLANKING ビットの影響を考慮して現在の COMP チャネル 1 出力を反映します。

28.7.2 コンパレータ割り込みクリアフラグレジスタ（COMP_ICFR）

COMP_ICFR は、コンパレータ割り込みクリアフラグレジスタです。

アドレスオフセット：0x00

システムリセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IF	CC1IF
														w1o	w1o
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC2IF**：クリア COMP チャネル 2 割り込みフラグ
このビットに 1 を書き込むと、COMP_SR レジスタの C2IF フラグがクリアされます。

ビット 16 **CC1IF**：クリア COMP チャネル 1 割り込みフラグ
このビットに 1 を書き込むと、COMP_SR レジスタの C1IF フラグがクリアされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

28.7.3 コンパレータオプションレジスタ（COMP_OR）

COMP_OR は、コンパレータオプションレジスタです。

アドレスオフセット：0x08

システムリセット値：0x0000 0000

OR_CFG=0 の場合

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OR15	OR14	OR13	OR12	OR11	AFOP										
rw	rw	rw	rw	rw	rw										

OR_CFG=1 の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OR31	OR30	OR29	OR28	OR27	OR26	OR25	OR24	OR23	OR22	OR21	OR20	OR19	OR18	OR17	OR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OR15	OR14	OR13	OR12	OR11	AFOP										
rw	rw	rw	rw	rw	rw										

ビット 31:11 **OR**：オプションレジスタ（OR_CFG=1 の場合）

ビット 15:11 **OR**：オプションレジスタ（OR_CFG=0 の場合）

ビット 10:0 **AFOP[10:0]**：出力ポートのオルタネート機能用ソース選択
このフィールドのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。
出力ポート（GPIO）の対応：
ビット 10　ビット 9　ビット 8　ビット 7　ビット 6　ビット 5　ビット 4　ビット 3
ビット 2　ビット 1　ビット 0
PK2　PI4　PI1　PG4　PG3　PG2　PE15　PE6　PB12　PA8　PA6
各ビットに対して：
0：COMP1_OUT が対応する GPIO のオルタネート機能として選択されます。
1：COMP2_OUT が対応する GPIO のオルタネート機能として選択されます。

28.7.4 コンパレータ設定レジスタ 1 （COMP_CFGR1）

COMP_CFGR1 は、COMP チャネル 1 の設定レジスタです。

アドレスオフセット：0x0C

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	Res.	Res.	INPSEL	Res.	INMSEL[2:0]		
rw				rw							rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	Res.	POLARITY	SCALEN	BRGEN	EN
		rw				rw			rw			rw	rw	rw	rw

ビット 31 **LOCK**：ロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、COMP チャネル 1 設定レジスタ COMP_CFGR1[31:0] の内容全体と COMP_OR レジスタをロックします。

0：COMP_CFGR1[31:0] レジスタは読み出し／書き込み用です。

1：COMP_CFGR1[31:0] レジスタと COMP_OR レジスタは読み出し専用です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **BLANKING[3:0]**：COMP チャネル 1 ブランキングソース選択ビット

このフィールドのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

このフィールドは、COMP チャネル 1 出力ブランキングの入力ソースを選択します。

0000：ブランキングなし

0001：comp_blk1

0010：comp_blk2

0011：comp_blk3

0100：comp_blk4

0101：comp_blk5

0110：comp_blk6

その他の値は予約済みです。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INPSEL**：COMP チャネル 1 非反転入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：COMP1_INP1（PB0）

1：COMP1_INP2（PB2）

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **INMSEL[2:0]**：COMP チャネル 1 反転入力選択フィールド

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 の負入力に接続される入力を選択します。

000 = 1/4 V_{REF_COMP}

001 = 1/2 V_{REF_COMP}

010 = 3/4 V_{REF_COMP}

011 = V_{REF_COMP}

100 = comp_inm1（DAC チャネル 1 出力）

101 = comp_inm2（DAC チャネル 2 出力）

110 = COMP1_INM1（PB1）

111 = COMP1_INM2（PC4）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **PWRMODE[1:0]**：COMP チャネル 1 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 の電力／スピードを制御します。

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **HYST[1:0]**：COMP チャネル 1 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ITEN**：COMP チャネル 1 割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネル 1 の 割り込み生成を有効にします。

0：COMP チャネル 1 の割り込み生成無効

1：COMP チャネル 1 の割り込み生成有効

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **POLARITY**：COMP チャネル 1 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは COMP チャネル 1 の極性を反転させます。

0：COMP チャネル 1 出力は非反転です。

1：COMP チャネル 1 出力は反転です。

ビット 2 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネルの V_{REFINT} スケーラを有効にします。

0：V_{REFINT} スケーラは無効です（COMP_CFGR2 レジスタの SCALEN ビットも Low である場合）。

1：V_{REFINT} スケーラは有効です。

ビット 1 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP_CFGR2 レジスタの BRGEN ビットも Low である場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、スケーラ出力は 4 本とも同一レベルの V_{REF_COMP}（V_{REFINT} とほぼ同じ）となります。

SCALEN および BRGEN がセットされている場合、スケーラ出力 4 本のレベルは、それぞれ V_{REF_COMP}、3/4 V_{REF_COMP}、1/2 V_{REF_COMP}、および 1/4 V_{REF_COMP} となります。

ビット 0 **EN**：COMP チャネル 1 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。COMP チャネル 1 を有効化します。

0：無効化

1：有効化

28.7.5 コンパレータ設定レジスタ 2 （COMP_CFGR2）

COMP_CFGR2 は、COMP チャネル 1 の設定レジスタです。

アドレスオフセット：0x10

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	Res.	Res.	INPSEL	Res.	INMSEL[2:0]		
rw				rw							rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	WINMODE	POLARITY	SCALEN	BRGEN	EN
		rw				rw			rw		rw	rw	rw	rw	rw

ビット 31 **LOCK**：ロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、COMP チャネル 2 設定レジスタ COMP_CFGR2[31:0] の内容全体と COMP_OR レジスタをロックします。

0：COMP_CFGR2[31:0] レジスタは読み出し／書き込み用です。

1：COMP_CFGR2[31:0] レジスタと COMP_OR レジスタは読み出し専用です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **BLANKING[3:0]**：COMP チャネル 2 ブランキングソース選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットによって COMP チャネル 2 出力のブランキングを制御するタイマ出力を選択します。

0000：ブランキングなし

0001：TIM1 OC5 がブランキングソースとして選択されます。

0010：TIM2 OC3 がブランキングソースとして選択されます。

0011：TIM3 OC3 がブランキングソースとして選択されます。

0100：TIM3 OC4 がブランキングソースとして選択されます。

0101：TIM8 OC5 がブランキングソースとして選択されます。

0110：TIM15 OC1 がブランキングソースとして選択されます。

その他の値は予約済みです。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INPSEL**：COMP チャネル 2 非反転入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：COMP2_INP1（PE9）

1：COMP2_INP2（PE11）

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **INMSEL[2:0]**：COMP チャネル 2 反転入力選択フィールド

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 の負入力に接続される入力を選択します。

000 = 1/4 V_{REF_COMP}

001 = 1/2 V_{REF_COMP}

010 = 3/4 V_{REF_COMP}

011 = V_{REF_COMP}

100 = comp_inm1（DAC チャネル 1 出力）

101 = comp_inm2（DAC チャネル 2 出力）

110 = COMP2_INM1（PE10）

111 = COMP2_INM2（PE7）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **PWRMODE[1:0]**：COMP チャネル 2 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 の電力／スピードを制御します。

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **HYST[1:0]**：COMP チャネル 2 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ITEN**：COMP チャネル 2 割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネル 2 の 割り込み生成を有効にします。

0：COMP チャネル 2 の割り込み生成無効

1：COMP チャネル 2 の割り込み生成有効

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WINMODE**：ウィンドウコンパレータモード選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、コンパレータのウィンドウモードを選択します。セットされると、COMP チャネル 2 の非反転入力は、COMP チャネル 1 の非反転入力に内部で接続されます。

ビット値に応じて、COMP チャネル 2 の非反転入力は次のものに接続されます。

0：COMP2_INP 入力セレクタ

1：COMP チャネル 1 の非反転入力 comp1_inp

ビット 3 **POLARITY**：COMP チャネル 2 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは COMP チャネル 2 の極性を反転させます。

0：COMP チャネル 2 出力は非反転です。

1：COMP チャネル 2 出力は反転です。

ビット 2 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネルの V_{REFINT} スケーラを有効にします。

0：V_{REFINT} スケーラは無効です（COMP_CFGR1 レジスタの SCALEN ビットも Low である場合）。

1：V_{REFINT} スケーラは有効です。

ビット 1 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP_CFGR1 レジスタの BRGEN ビットも Low である場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、スケーラ出力は 4 本とも同一レベルの V_{REF_COMP}（V_{REFINT} とほぼ同じ）となります。

SCALEN および BRGEN がセットされている場合、スケーラ出力 4 本のレベルは、それぞれ V_{REF_COMP}、3/4 V_{REF_COMP}、1/2 V_{REF_COMP}、および 1/4 V_{REF_COMP} となります。

ビット 0 **EN**：COMP チャネル 2 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。COMP チャネル 2 を有効化します。

0：無効化

1：有効化

28.7.6 COMP レジスタマップ

次の表に コンパレータ レジスタの一覧を示します。

表 223. COMP レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	COMP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2IF	C1IF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2VAL	C1VAL
	リセット値															0	0															0	0
0x04	COMP_ICFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IF	CC1IF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0																
0x08	COMP_OR（OR_CFG=0）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OR15	OR14	OR13	OR12	OR11	AFOP										
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	COMP_OR（OR_CFG=1）	OR31	OR30	OR29	OR28	OR27	OR26	OR25	OR24	OR23	OR22	OR21	OR20	OR19	OR18	OR17	OR16	OR15	OR14	OR13	OR12	OR11	AFOP										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	COMP_CFGR1	LOCK	Res.	Res.	Res.	BLANKING				Res.	Res.	Res.	INPSEL	Res.	INMSEL			Res.	Res.	PWRMODE		Res.	Res.	HYST		Res.	ITEN	Res.	Res.	POLARITY	SCALEN	BRGEN	EN
	リセット値	0				0	0	0	0				0		0	0	0			0	0			0	0		0			0	0	0	0
0x10	COMP_CFGR2	LOCK	Res.	Res.	Res.	BLANKING				Res.	Res.	Res.	INPSEL	Res.	INMSEL			Res.	Res.	PWRMODE		Res.	Res.	HYST		Res.	ITEN	Res.	WINMODE	POLARITY	SCALEN	BRGEN	EN
	リセット値	0				0	0	0	0				0		0	0	0			0	0			0	0		0		0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

29 オペアンプ（OPAMP）

29.1 概要

このデバイスは、2 個のオペアンプを内蔵しており、それぞれ入力 2 本と出力 1 本を持っています。3 つの I/O は外部ピンにつなげることができ、あらゆる種類の外部相互接続が可能です。オペアンプは、フォロアとして、または 2～16 の範囲の非反転ゲインもしくは -1～-15 の範囲の反転ゲインを持つアンプとして、内部的に設定できます。

正入力は、内部 DAC に接続できます。

出力の 1 本は、内部 ADC に接続できます。

29.2 OPAMP の主な機能

- レールツーレール入出力電圧範囲
- 低い入力バイアス電流（最低 1 nA）
- 低い入力オフセット電圧（較正後 1.5 mV、出荷時較正で 10 mV）
- 7 MHz ゲイン帯域幅
- スルーレートが改善されるハイスピードモード

注： ***OPAMP 特性の詳細については、製品データシートを参照してください。***

29.3 OPAMP の機能説明

OPAMP にはいくつかのモードがあります。

各 OPAMP は個別に有効化でき、無効になっているときの出力はハイインピーダンスです。

有効時は、較正モードにしてから OPAMP のすべての入出力を切断したり、機能モードにしたりできます。

ハイスピードモードと通常モードの 2 種類の機能モードがあります。機能モードでは、OPAMP の入出力は*セクション 29.3.3：信号接続*に示すように接続されます。

29.3.1 OPAMP のリセットおよびクロック

オペアンプのクロックはレジスタへのアクセスに必要です。アプリケーションがこれらのレジスタに読み書きアクセスを必要としない場合、ペリフェラルクロックイネーブルレジスタでクロックをオフに切り替えられます（*セクション 8.7.43：RCC APB1 クロックレジスタ（RCC_APB1LENR）*の OPAMPEN ビットを参照）。

OPAEN ビットは、OPAMP の動作を有効化または無効化します。OPAMP レジスタ設定は、出力に誤った影響を及ぼさないように OPAEN ビットを有効化する前に変更する必要があります。

オペアンプの出力が必要なくなったときに、オペアンプを無効にして電力を低減することができます。以前にセットしたすべての設定（較正を含む）は、OPAMP が無効化されても保持されます。

29.3.2 初期設定

オペアンプのデフォルト設定は、3 つの入出力が外部ピンに接続されている機能モードです。デフォルトのモードでは、オペアンプはオフセット較正に出荷時のトリミング値を使用します。出荷時トリミング条件についてはデータシートの電気的特性のセクションを参照してください。通常、温度は 30 °C で、電圧は 3 V です。トリミング値は調整可能で、変更するには *セクション 29.3.5：較正*を参照してください。デフォルト設定が通常モードを使用する場合、標準のパフォーマンスを提供します。オペアンプをハイスピードモードに切り替えてスルーレートを改善するように、OPAHSM ビットでセットできます。通常モードとハイスピードモードの両方の特性は、データシートの *セクション「電気的特性」* に定義されています。

OPAMPx_CSR レジスタの OPAEN ビットがセットされるとすぐに、オペアンプは機能します。2 個の入力ピンおよび出力ピンは、*セクション 29.3.3：信号接続*に定義されているように接続されており、デフォルト接続設定は変更できます。

注： ***入力ピンおよび出力ピンは、対応する GPIOx_MODER レジスタでアナログモード（デフォルト状態）で設定する必要があります。***

29.3.3 信号接続

オペアンプのピン接続は、OPAMPx_CSR レジスタで決まります。

2 個のオペアンプ（OPAMP1 と OPAMP2）の接続を以下の表に記載しています。

表 224. オペアンプの可能な接続

信号	ピン	内部	コメント
OPAMP1_VINM	PC5(INM0) PA7(INM1)	ADC1_IN9 ADC2_IN9 OPAMP1_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットで制御します。
OPAMP1_VINP	PB0	dac_out1 ADC1_IN8 ADC2_IN8 COMP1_INP	ビット VP_SEL で制御します。
OPAMP1_VOUT	PC4	ADC1_IN4 ADC2_IN4 COMP1_INM7	OPAMP が有効になっているときにピンが接続されます。ADC 入力は、ADC で制御されます。
OPAMP2_VINM	PE8(INM0) PG1(INM1)	OPAMP2_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットで制御します。
OPAMP2_VINP	PE9	dac_out2 COMP2_INP	ビット VP_SEL で制御します。
OPAMP2_VOUT	PE7	COMP2_INM7	-

29.3.4 OPAMP モード

すべてのオペアンプの入出力に、端子からアクセスできます。アンプは複数の設定環境で使用できます。

- スタンドアロンモード（外部ゲイン設定モード）
- フォロア設定モード
- PGA モード

注： ***出力インピーダンスを最小限に抑えるために、アンプ出力ピンは出力パッドに直接接続されています。アンプが有効化されると、アンプが PGA として設定され、内部チャネルにのみ接続されていても、汎用 I/O として使用することはできません。***

信号のインピーダンスを、（ソースの抵抗性電圧降下が原因で）重大な乱れをつくる入力リークが発生しないレベル以下に維持する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。

スタンドアロンモード（外部ゲイン設定モード）

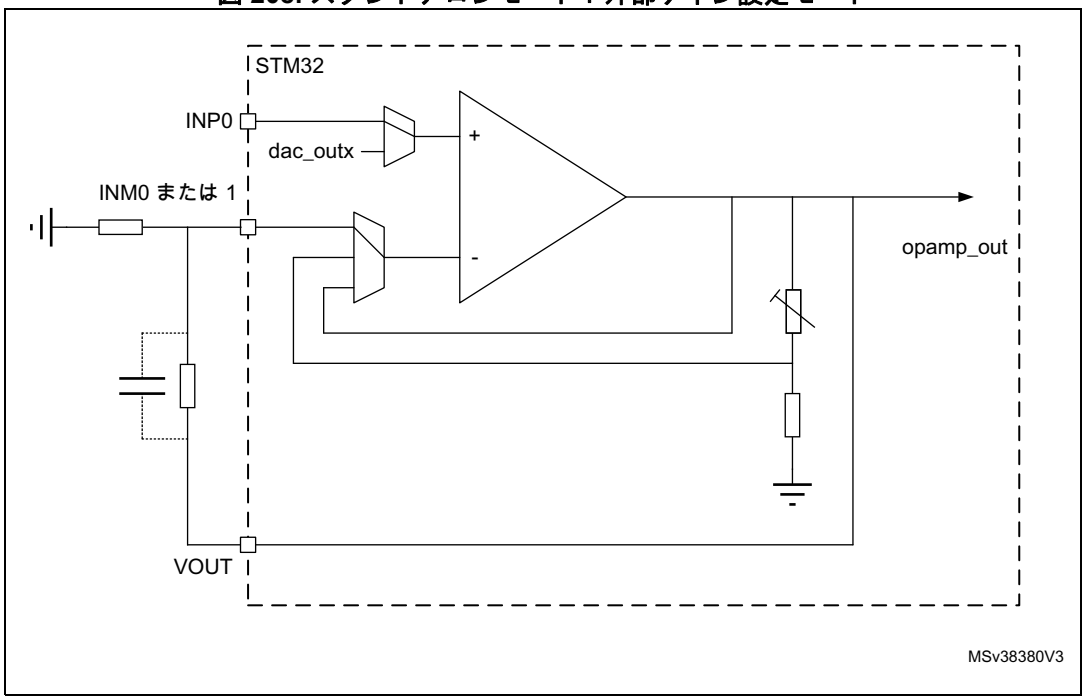
スタンドアロンモードで OPAMP を使用する手順をここで説明します。

OPAMPx_CSR のデフォルト値と GPIOx_MODER のデフォルト状態から始めて、OPAEN ビットがセットされるとすぐに、入力ピン2つと出力ピン1つがオペアンプに接続されます。

デフォルト設定が出荷時トリミング値を使用して、通常モード（最高のパフォーマンス）で動作します。OPAMP の動作は、次のように変更できます。

- 高スルーレートとするために OPAHSM を「オペアンプハイスピード」モードにセットできます。
- USERTRIM をセットして、入力オフセットのためのトリミング値を変更できます。

図 208. スタンドアロンモード：外部ゲイン設定モード



フォロア設定モード

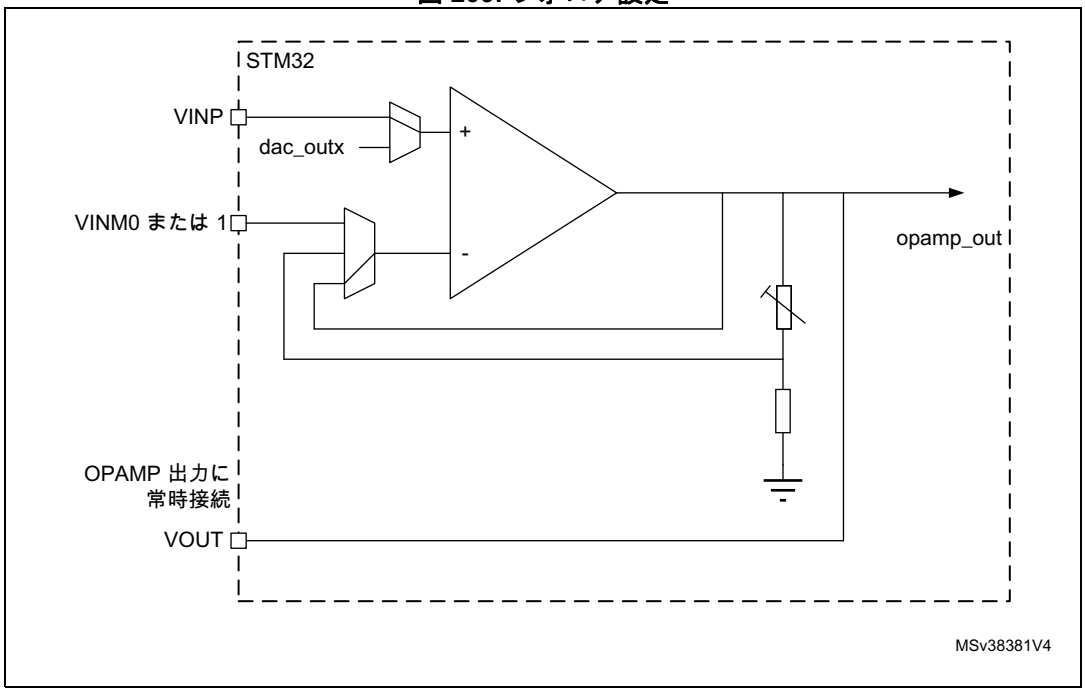
フォロアモードで OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「OPAMPx_VINM 入力に接続された opamp_out」である 11 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。
- OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が OPAMPx_VOUT ピンにバッファリングされます。

注： ***OPAMPx_VINM に対応するピンは、他の用途のために解放されます。***

OPAMP1 出力の信号も ADC 入力と見なされます。結果として、フォロアモードに設定された OPAMP を使用して、入力信号周波数がオペアンプのゲイン周波数の仕様と互換性があると想定して、入力信号を ADC 入力に供給する前にインピーダンス適合を実行できます。

図 209. フォロア設定



参考資料

プログラマブルゲインアンプモード

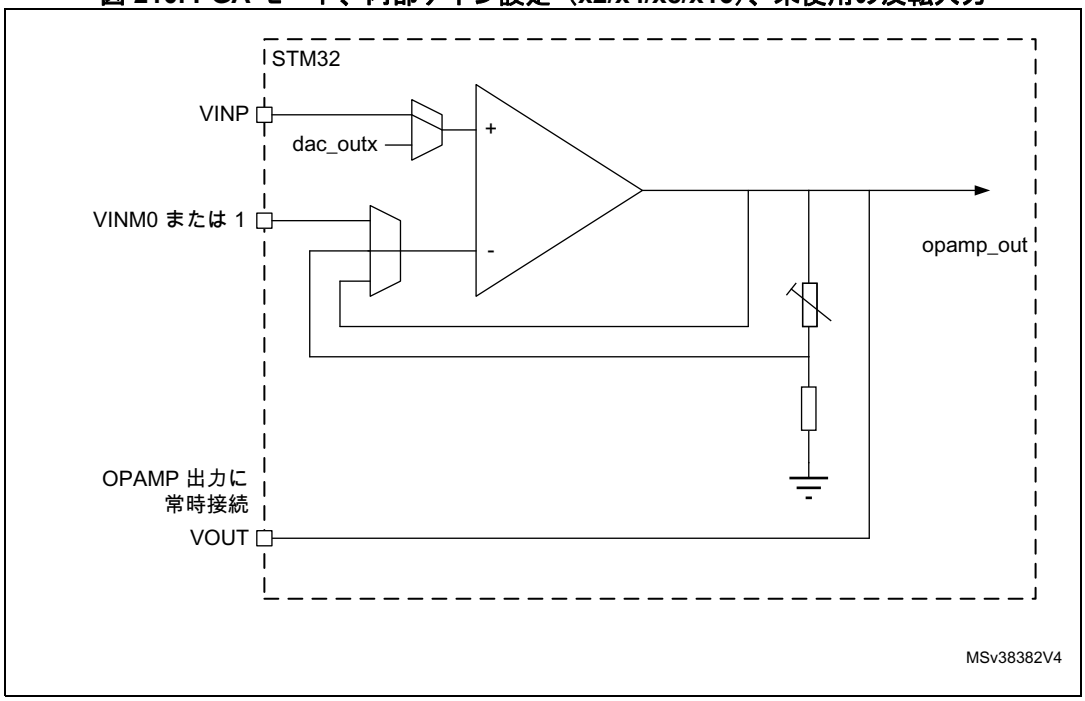
プログラム可能なゲインアンプとして OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「内部ゲイン 2、4、8 または 16」である 0000～0011 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。

  OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が選択されたゲインで増幅し、OPAMPx_VOUT ピンで確認できます。

注： ***飽和を避けるために、入力電圧は選択されたゲインで分圧された V_{DDA} 未満にしておく必要があります。***

図 210. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力



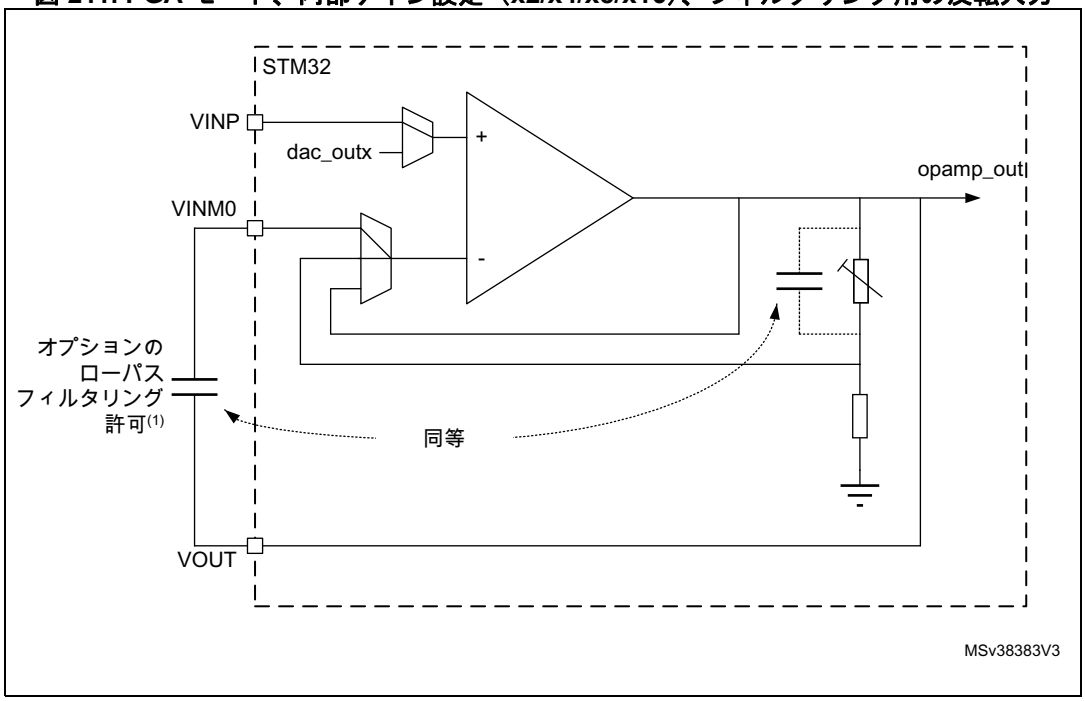
外部フィルタリングでのプログラム可能なゲインアンプモード

入力信号を増幅するために外部フィルタリングで OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「INM0 のフィルタリング付き内部ゲイン 2、4、8 または 16」である 0100～0111 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

  INM であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と INM の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 211. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力



1. ゲインはカットオフ周波数によって決まります。

プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたは反転モード

非反転モードまたは反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「INM0の反転ゲイン = -1、-3、-7、-15／非反転ゲイン = 2、4、8、16」である 1000～1011 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

図 212. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）または反転ゲイン設定（x-1/x-3/x-7/x-15）

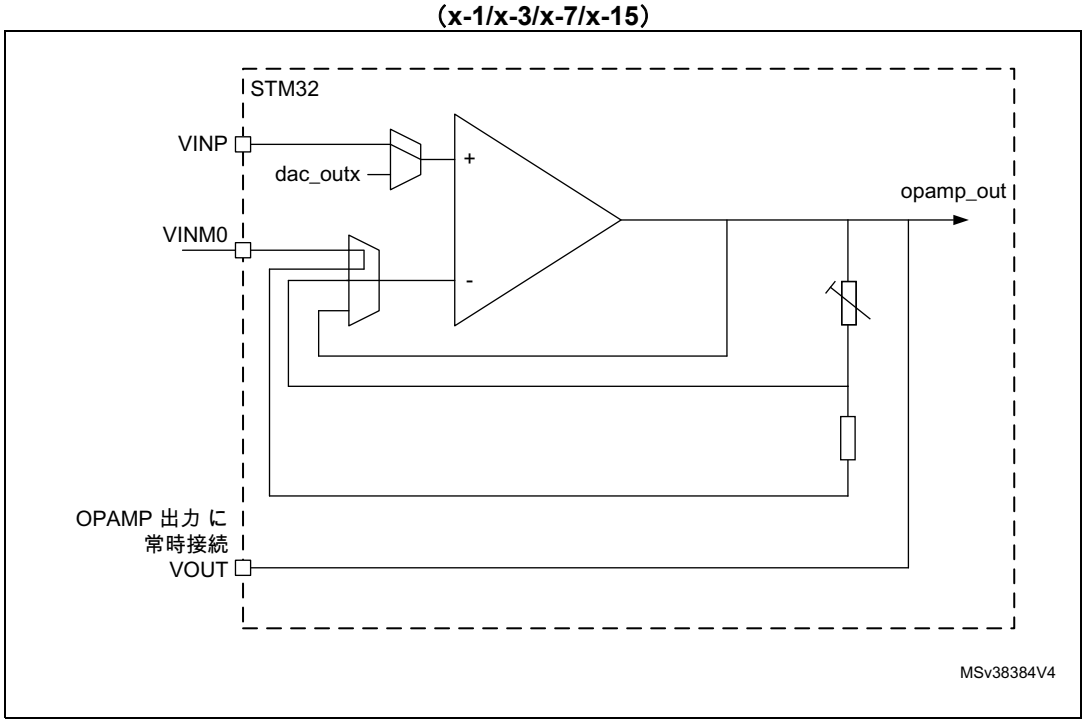
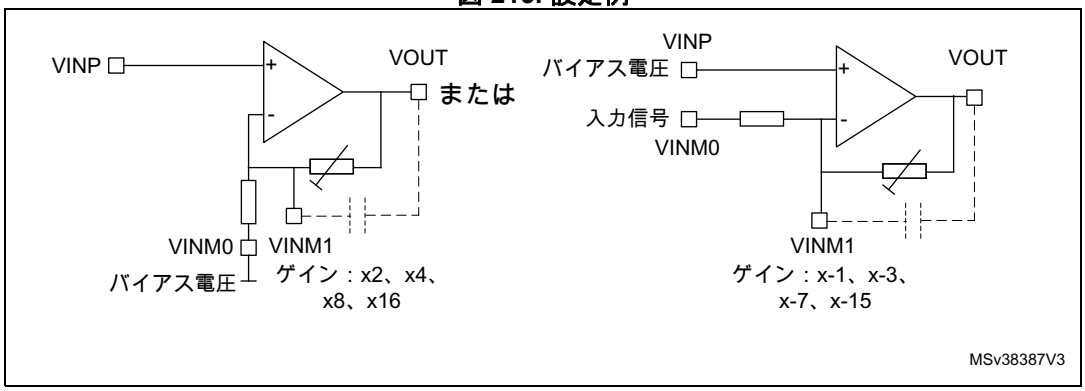


図 213. 設定例



プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたはフィルタリング付き反転モード

非反転モードまたはフィルタリング付き反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「フィルタリング用ノード INM0、INM1 の反転ゲイン = -1、-3、-7、-15／非反転ゲイン = 2、4、8、16」である 1100～1111 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

  VM1 であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と VM1 の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 214. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）またはフィルタリング付き反転ゲイン設定（x-1/x-3/x-7/x-15）

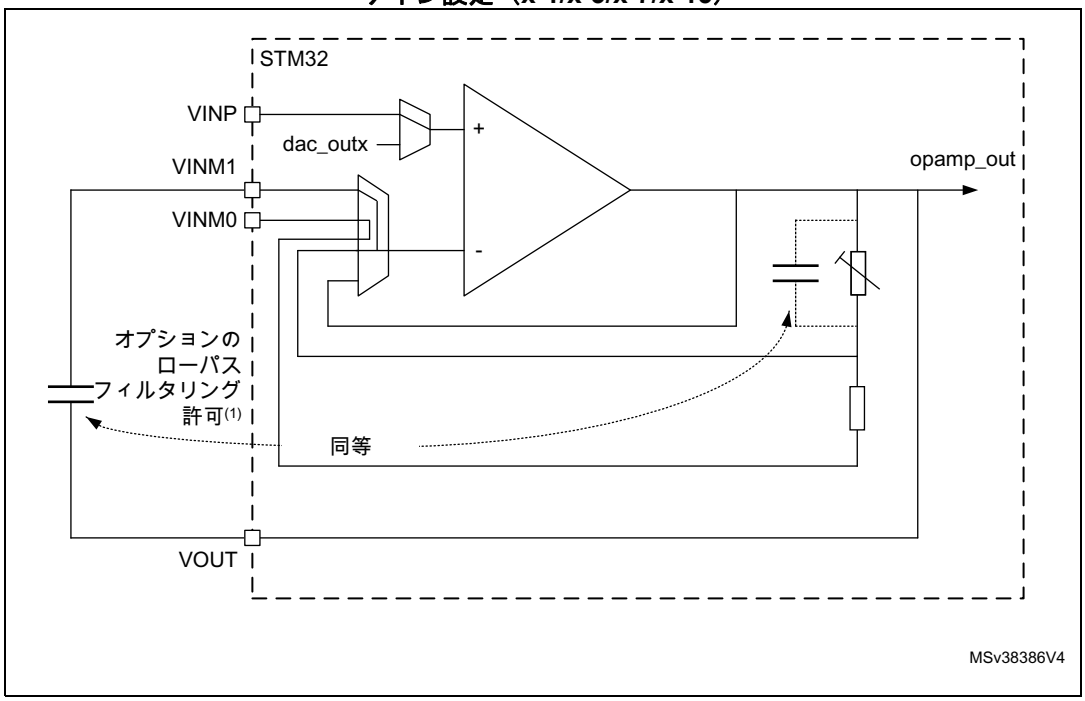
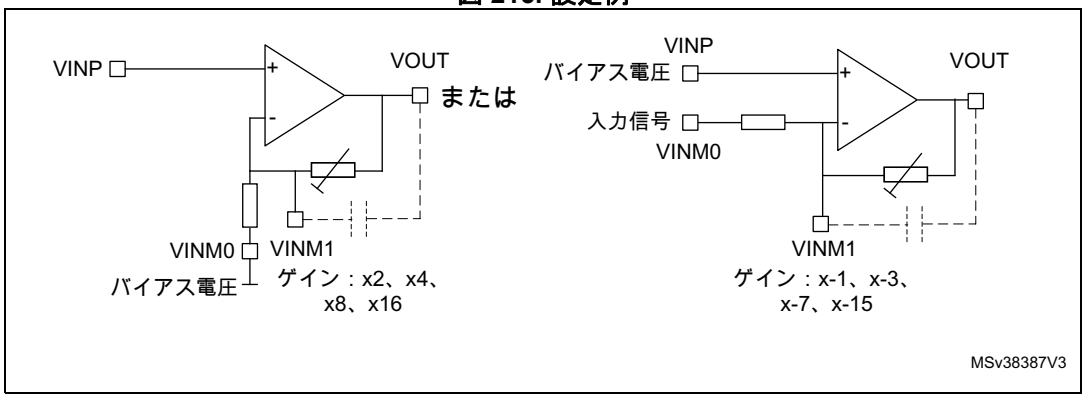


図 215. 設定例



29.3.5 較正

OPAMP インタフェースでは、トリミングされたオフセット値をオペアンプに送信し続けます。起動時、トリミング値はプリセットされた「出荷時」トリミング値に初期化されます。

各オペアンプを、ユーザがトリミングすることもできます。特定のレジスタでは、通常モードとハイスピードモードで異なるトリミング値を持つことができます。

較正の目的は、OPAMP 入力オフセット電圧をできる限りキャンセルすることです。較正回路を使えば、安定した電圧条件と温度条件の範囲で、入力オフセット電圧を +/-1.5 mV 未満まで低減できます。

各オペアンプおよび各モードに対して、N 差分ペアと P 差分ペアの 2 つのトリミング値をトリミングする必要があります。

各オペアンプのオフセットをトリミングするために、通常モード（OPAMPx_OTR）とハイスピードモード（OPAMPx_HSOTR）の 2 つのレジスタがあります。各レジスタは、P 差分ペアトリミング用の 5 個のビット、N 差分ペアトリミングの 5 個のビットで構成されています。これらは「ユーザ」による値です。

ユーザは、OPAMPx_CSR レジスタの USERTRIM ビットを使用して、「出荷時」の値から「ユーザ」がトリミングした値に切り替えることができます。このビットは起動時にリセットされるので、「出荷時」の値がデフォルトで OPAMP オプションレジスタに適用されます。

ユーザは較正モードまたは機能モードでトリミング値を変更する必要があります。

オフセットトリミングレジスタは通常、ビット CALON を 1 にセットすることで較正操作が初期化された後に設定されます。CALON = 1 の場合、オペアンプの入力は機能環境から切断されます。

- CALSEL を 01 にセットすると、P 差分ペアのオフセット較正が初期化されます（低電圧基準使用）。
- CALSEL を 11 にリセットすると、N 差分ペアのオフセット較正が初期化されます（高電圧基準使用）。

CALON = 1 の場合、ビット CALOUT は CALSEL および OPAHSM で選択されたトリミング値の影響を反映します。ソフトウェアによって OPAMP 制御レジスタの TRIMOFFSETN ビットを 0x00 からインクリメントし、OPAMP 制御レジスタの CALOUT ビットが 1 から 0 に変わる最初の値にする必要があります。CALOUT ビットがリセットされると、オフセットが正しく較正されるので対応するトリミング値を格納する必要があります。CALOUT フラグは、トリミング値が変更され安定した後、最大 1 ms 必要とします（データシートの電気的特性セクションにある $t_{OFFTRIM}$ 最大遅延仕様を参照）。

注： ***トリミング値が最適なトリミング値に近いほど、安定化にかかる時間が長くなります（いずれの場合でも最大安定時間が 1 ms 未満のままの状態）。***

表 225. 動作モードと較正

モード	制御ビット				出力	
	OPAEN	OPAHSM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
通常動作モード	1	0	0	X	アナログ	0
ハイスピードモード	1	1	0	X	アナログ	0
パワーダウン	0	X	X	X	Z	0
通常モードのオフセット較正 N	1	0	1	11	アナログ	X
通常モードのオフセット較正 P	1	0	1	01	アナログ	X

表 225. 動作モードと較正（続き）

モード	制御ビット				出力	
	OPAEN	OPAHSM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
ハイスピードモードのオフセット較正 N	1	1	1	11	アナログ	X
ハイスピードモードのオフセット較正 P	1	1	1	01	アナログ	X

較正手順

オペアンプのいずれか 1 つを完全に較正するための手順を示します。

1. OPAMPx_CSR の OPAEN ビットを 1 にセットしてオペアンプを有効にします。
2. OPAMPx_CSR レジスタのUSERTRIM ビットを 1 にセットします。
3. 較正モードを選択します（*表 225：動作モードと較正*を参照）。手順 3 と 4 は 4 回繰り返す必要があります。初回では、次を選択します。
   – 通常モードと N 差分ペア

   上記の較正モードは、OPAMPx_CSR レジスタの OPAHSM=0 および CALSEL=11 に対応します。
4. OPAMPx_CSR の CALOUT が 0 に変わるまで、OPAMPx_OTR の TRIMOFFSETN[4:0] を 00000b から増やします。

注： ***OPAMPx_OTR レジスタの書き込みと CALOUT 値の読み出しの間に、データシートの電気的特性セクションで指定された $t_{OFFTRIM}$ 最大遅延を待って、正しい CALOUT 値を取得します。***

このひずみ修正は、オフセットが正しく補正されることと、対応するトリミングコードを OPAMPx_OTR レジスタに保存する必要があることを示します。

次の場合、手順 3～4 を繰り返します。

- 通常モード、P 差分ペア、CALSEL=01
- ハイスピードモードと N 差分ペア
- ハイスピードモードと P 差分ペア

モードを使用しない場合は、対応する較正を実行する必要はありません。

すべてのオペアンプは、同時に較正できます。

注： ***較正フェーズ全体において、オペアンプ出力の外部接続によって、500 µA を超えるプルアップまたはプルダウンはしないでください。***

29.4 OPAMP 低電力モード

表 226. 低消費電力モードが OPAMP に与える影響

モード	説明
SLEEP	影響はありません。
D2 STOP	影響はありません。OPAMP レジスタの内容は保たれます。
STANDBY	OPAMP レジスタはパワーダウンされ、STANBY モード終了後に再初期化する必要があります。

29.5 OPAMP PGA ゲイン

OPAMP が PGA モードとして設定された場合には、非反転モードではx2、x4、x8、x16、反転モードではx-1、x-3、x-7、x-15 のゲインを選択できます。

OPAMP が非反転モードとして設定された場合には、ゲインエラーは製品データシートを参照してください。反転モードとして設定された場合には、ゲイン係数は、オンチップのフィードバック抵抗によってだけではなく、信号源の出力インピーダンスによっても定まります。PGA のフィードバック抵抗と比較して、信号源の出力インピーダンスが無視できない場合には、ゲインエラーが生じます。製品データシートの PGA 抵抗値を参照してください。

29.6 OPAMP レジスタ

29.6.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR）

アドレス：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CAL OUT	TST REF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USER TRIM	PGA_GAIN	
	r	rw											rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN		CALSEL		CALON	Res.	Res.	OPA HSM	Res.	VM_SEL		Res.	VP_SEL		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw		rw	rw		rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CALOUT**：オペアンプ較正出力

OPAMP 出力ステータスフラグ較正モードの間、OPAMP はコンパレータとして使用されます。

0：非反転 < 反転

1：非反転 >反転

ビット 29 **TSTREF**：OPAMP 較正基準電圧出力制御（テスト用に予約済み）

0：OPAMP の INTVREF は出力されません。

1：OPAMP の INTVREF は出力されます。

ビット 28:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **USERTRIM**：ユーザートリミング有効

このビットによって、「出荷時」AOP オフセットトリミング値から「ユーザ」AOP オフセットトリミング値への切り替え可能

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 17:14 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値

0000：非反転内部ゲイン 2、VREF基準

0001：非反転内部ゲイン 4、VREF基準

0010：非反転内部ゲイン 8、VREF基準

0011：非反転内部ゲイン 16、VREF基準

0100：非反転内部ゲイン 2、INM0フィルタリング付、VREF基準

0101：非反転内部ゲイン 4、INM0フィルタリング付、VREF基準

0110：非反転内部ゲイン 8、INM0フィルタリング付、VREF基準

0111：非反転内部ゲイン 16、INM0フィルタリング付、VREF基準

1000：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノード付

1001：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノード付

1010：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノード付

1011：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノード付

1100：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1101：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1110：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1111：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

ビット 13:12 **CALSEL**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA

01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）

10：OPAMP 入力に印加される 0.5*VDDA

11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード

1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAHSM**：オペアンプハイスピードモード

この設定を変更するには、オペアンプを無効化する必要があります。

0：通常モードのオペアンプ

1：ハイスピードモードのオペアンプ

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:5 **VM_SEL**：反転入力選択

これらのビットは、OPAMODE = 00、01、10 または 11 の場合のみ使用されます。

00 :OPAMP INM 入力に接続されている INM0

01：OPAMP NM 入力に接続されている INM1

10 :フィードバック抵抗は OPAMP INM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out は OPAMP INM 入力（フォロアモード）に接続されています。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **VP_SEL**：非反転入力選択

00：OPAMPx_VINP に接続されている GPIO

01：OPAMPx_VINP に接続されている dac_outx

10：予約済み

11：予約済み

ビット 1 **FORCE_VP：VP の 強制内部基準（テスト用に予約済み）**

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

29.6.2 通常モードでの OPAMP1 トリミングレジスタ（OPAMP1_OTR）

アドレス：0x04

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMOFFSETP					Res.	Res.	Res.	TRIMOFFSETN				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

29.6.3 ハイスピードモードでの OPAMP1 トリミングレジスタ（OPAMP1_HSOTR）

アドレス：0x08

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMHSOFFSETP					Res.	Res.	Res.	TRIMHSOFFSETN				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMHSOFFSETP[4:0]**：PMOS 差分ペアのハイスピードモードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMHSOFFSETN[4:0]**：NMOS 差分ペアのハイスピードモードトリミング

29.6.4 OPAMP オプションレジスタ（OPAMP_OR）

アドレス：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:0 予約済みであり、リセット値に保持する必要があります。

29.6.5 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR）

アドレス：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CAL OUT	TST REF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USER TRIM	PGA_GAIN	
	r	rw											rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN		CALSEL		CALON	OPA MODE		OPA HSM	Res.	VM_SEL		Res.	VP_SEL		FORCE _VP	OPAEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CALOUT**：オペアンプ較正出力

OPAMP 出力ステータスフラグ較正モードの間、OPAMP はコンパレータとして使用されます。

0：非反転 < 反転

1：非反転 >反転

ビット 29 **TSTREF**：OPAMP 較正基準電圧出力制御（テスト用に予約済み）

0：OPAMP の INTVREF は出力されません。

1：OPAMP の INTVREF は出力されます。

ビット 28:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **USERTRIM**：ユーザートリミング有効

このビットによって、「出荷時」AOP オフセットトリミング値から「ユーザ」AOP オフセットトリミング値への切り替え可能

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 17:14 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値

0000：非反転内部ゲイン 2、VREF基準

0001：非反転内部ゲイン 4、VREF基準

0010：非反転内部ゲイン 8、VREF基準

0011：非反転内部ゲイン 16、VREF基準

0100：非反転内部ゲイン 2、INM0フィルタリング付、VREF基準

0101：非反転内部ゲイン 4、INM0フィルタリング付、VREF基準

0110：非反転内部ゲイン 8、INMINM0フィルタリング付、VREF基準

0111：非反転内部ゲイン 16、INM0フィルタリング付、VREF基準

1000：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノード付

1001：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノード付

1010：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノード付

1011：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノード付

1100：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1101：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1110：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1111：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

ビット 13:12 **CALSEL**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正パスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA

01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）

10：OPAMP 入力に印加される 0.5*VDDA

11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード

1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAHSM**：オペアンプハイスピードモード
この設定を変更するには、オペアンプを無効化する必要があります。
0：通常モードのオペアンプ
1：ハイスピードモードのオペアンプ

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:5 **VM_SEL**：反転入力選択
これらのビットは、OPAMODE = 00、01、10 または 11 の場合のみ使用されます。
00 :OPAMP INM 入力に接続されている INM0
01：OPAMP INM 入力に接続されている INM1
10 :フィードバック抵抗は OPAMP INM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。
11：opamp_out は OPAMP INM 入力（フォロアモード）に接続されています。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **VP_SEL**：非反転入力選択
00：OPAMPx_VINP に接続されている GPIO
01：OPAMPx_VINP に接続されている DAC
10：予約済み
11：予約済み

ビット 1 **FORCE_VP：VP の 強制内部基準（テスト用に予約済み）**
0：通常動作モード。非反転入力は入力に接続されます。
1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

29.6.6 通常モードでの OPAMP2 トリミングレジスタ（OPAMP2_OTR）

アドレス：0x14

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TRIMOFFSETP					Res.	Res.	Res.	TRIMOFFSETN				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

29.6.7 ハイスピードモードでの OPAMP2 トリミングレジスタ（OPAMP2_HSOTR）

アドレス：0x18

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMHSOFFSETP[4:0]**：PMOS 差分ペアのハイスピードモードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMHSOFFSETN[4:0]**：NMOS 差分ペアのハイスピードモードトリミング

29.6.8 OPAMP レジスタマップ

表 227. OPAMP レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	OPAMP1_CSR	Res.	CALOUT	TSTREF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USERTRIM	PGA_GAIN				CALSEL		CALON	Res.	Res.	OPAHSM	Res.	VM_SEL		Res.	VP_SEL		FORCE_VP	OPAEN
	リセット値		0	0											0	0	0	0	0	0	0	0			0		0	0		0	0	0	0
0x04	OPAMP1_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x08	OPAMP1_ HSOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMHS OFFSETP[4:0]					Res.	Res.	Res.	TRIMHS OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x0C	OPAMP_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x10	OPAMP2_CSR	Res.	CALOUT	TSTREF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USERTRIM	PGA_GAIN				CALSEL		CALON	Res.	Res.	OPAHSM	Res.	VM_SEL		Res.	VP_SEL		FORCE_VP	OPAEN
	リセット値		0	0											0	0	0	0	0	0	0	0			0		0	0		0	0	0	0
0x14	OPAMP2_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x18	OPAMP2_ HSOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMHS OFFSETP[4:0]					Res.	Res.	Res.	TRIMHS OFFSETN[4:0]				
	リセット値																				(1)								(1)				

1. 出荷時のトリミング値。

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス* を参照してください。

30 デルタシグマモジュレータのデジタルフィルタ（DFSDM）

30.1 概要

デルタシグマモジュレータのデジタルフィルタ（DFSDM）は、外部デルタシグマモジュレータをマイクロコントローラにインタフェース接続するための専用ハイパフォーマンスモジュールです。最大 8 つまでの外部デジタルシリアルインタフェース（チャネル）と、柔軟性の高いデルタシグマストリームデジタル処理オプションを持つ最大 4 つのデジタルフィルタの機能があり、最終的に最大 24 ビットの ADC の分解能を提供します。DFSDM は、内部 ADC ペリフェラルまたはマイクロコントローラメモリからのオプションの並列データストリーム入力も備えています。

外部デルタシグマモジュレータにより、外部デルタシグマモジュレータのアナログ入力から変換されたアナログ値のデジタルデータストリームが利用できます。このデジタルデータストリームは、シリアルインタフェースを介して DFSDM 入力チャネルに送信されます。DFSDM は、いくつかの規格をサポートしており、次のさまざまなデルタシグマモジュレータ出力に接続します。SPI インタフェースおよびマンチェスタ符号化 1 線インタフェース（ともに調整可能なパラメータあり）。DFSDM モジュールは、最大 8 つの多重化入力デジタルシリアルチャネルの接続をサポートしており、最大 4 つの DFSDM モジュールで接続を共有します。また、DFSDM モジュールは最大 8 つの内部 16 ビットデータチャネルからの代替並列データ入力にもサポートしています（内部 ADC またはマイクロコントローラメモリから）。

DFSDM は入力データストリームを最終デジタルデータワードに変換して、デルタシグマモジュレータのアナログ入力のアナログ入力値を表します。変換は、入力シリアルデータストリームのデジタルフィルタリングおよびデシメーションといった設定可能なデジタル処理に基づいて行われます。

変換速度および分解能は、フィルタタイプ、フィルタ段数、フィルタの長さ、積分器の長さといったデジタル処理に関する設定可能なパラメータに合わせて調整できます。最大出力データの分解能は 24 ビットです。変換モードには、シングル変換モードと連続モードの 2 種類があります。データは、DMA を介してシステム RAM バッファに自動的に保存され、ソフトウェアの負荷を軽減できます。

柔軟性の高いタイマトリガシステムは、DFSDM の変換開始を制御するために使用できます。このタイミング制御により、同時変換のトリガや、変換の間にプログラム可能な遅延の挿入が可能です。

DFSDM にはアナログウォッチドッグ機能があります。入力チャネルデータストリームや最終出力データに、アナログウォッチドッグを割り当てることができます。アナログウォッチドッグには、独自の入力データストリームのデジタルフィルタリングがあり、監視するデータに必要な速度および分解能を満たすことができます。

制御アプリケーションにおける短絡を検出するために、短絡検出回路があります。このブロックは、各入力チャネルデータストリームにおいて一定時間内の安定したデータの発生（入力データストリームでの 0 または 1 について）を監視します。

極値検出回路ブロックは、最終出力データを監視して、出力データ値の最大値と最小値を保存します。保存された極値は、ソフトウェアによって復元することができます。

通常モードと STOP モードの 2 種類の電力モードがあります。

30.2 DFSDM の主な機能

- 最大 8 つの多重化入力デジタルシリアルチャネル：
  - さまざまなデルタシグマモジュレータを接続するために設定可能な SPI インタフェース
  - 設定可能なマンチェスタ符号化 1 線インタフェースのサポート
  - デルタシグマモジュレータのクロック出力
- 最大 8 つの内部デジタル並列チャネルからの代替入力：
  - 最大 16 ビットの分解能を持つ入力
  - 内部ソース：ADC データ または メモリ（CPU/DMA 書き込み）データストリーム
- 調整可能なデジタル信号処理：
  - $Sinc^x$ フィルタ：フィルタ段数／タイプ（1～5）、オーバーサンプリング比（最大 1～1024）
  - 積分器：オーバーサンプリング比（1～256）
- 最大 24 ビット出力データの分解能：
  - 最終データの右ビットシフタ（0～31 ビット）
- 符号付き出力データフォーマット
- 自動データオフセット補正（オフセットはユーザよってレジスタに保存）
- 連続またはシングル変換
- 変換開始の同期：
  - ソフトウェアトリガ
  - 内部タイマ
  - 外部イベント
  - 最初の DFSDM フィルタ（DFSDM_FLT0）と同期した変換開始
- アナログウォッチドッグの機能：
  - データの低閾値および高閾値レジスタ
  - 独自の設定可能な $Sinc^x$ デジタルフィルタ（段数 = 1～3、オーバーサンプリング比 = 1～32）
  - 出力データレジスタまたは 1 つ以上の入力デジタルシリアルチャネルからの入力
  - 標準変換から独立した連続監視
- 飽和したアナログ入力値（下限と上限）を検出する短絡検出回路：
  - 入力データストリームで 1～256 の連続した 0 と 1 を検出する最大 8 ビットカウンタ
  - 各チャネルの連続監視（8 つのシリアルチャネルのトランシーバ出力）
- アナログウォッチドッグイベントや短絡検出回路イベントでのブレーク生成
- 極値検出回路：
  - 出力データの最小値と最大値の保存
  - ソフトウェアによるリフレッシュ
- DMA を使用した変換データ読み出し可能
- 割り込み：変換終了、オーバーラン、アナログウォッチドッグ、短絡、チャネルクロック無
- 「レギュラ」または「インジェクト」変換：
  - 「レギュラ」変換は「インジェクト」変換のタイミングに影響しない限り、連続モードでもいつでもリクエスト可能

30.3 DFSDM の実装

このセクションでは、DFSDMx に実装されている設定について説明しています。

表 228. DFSDM1 の実装

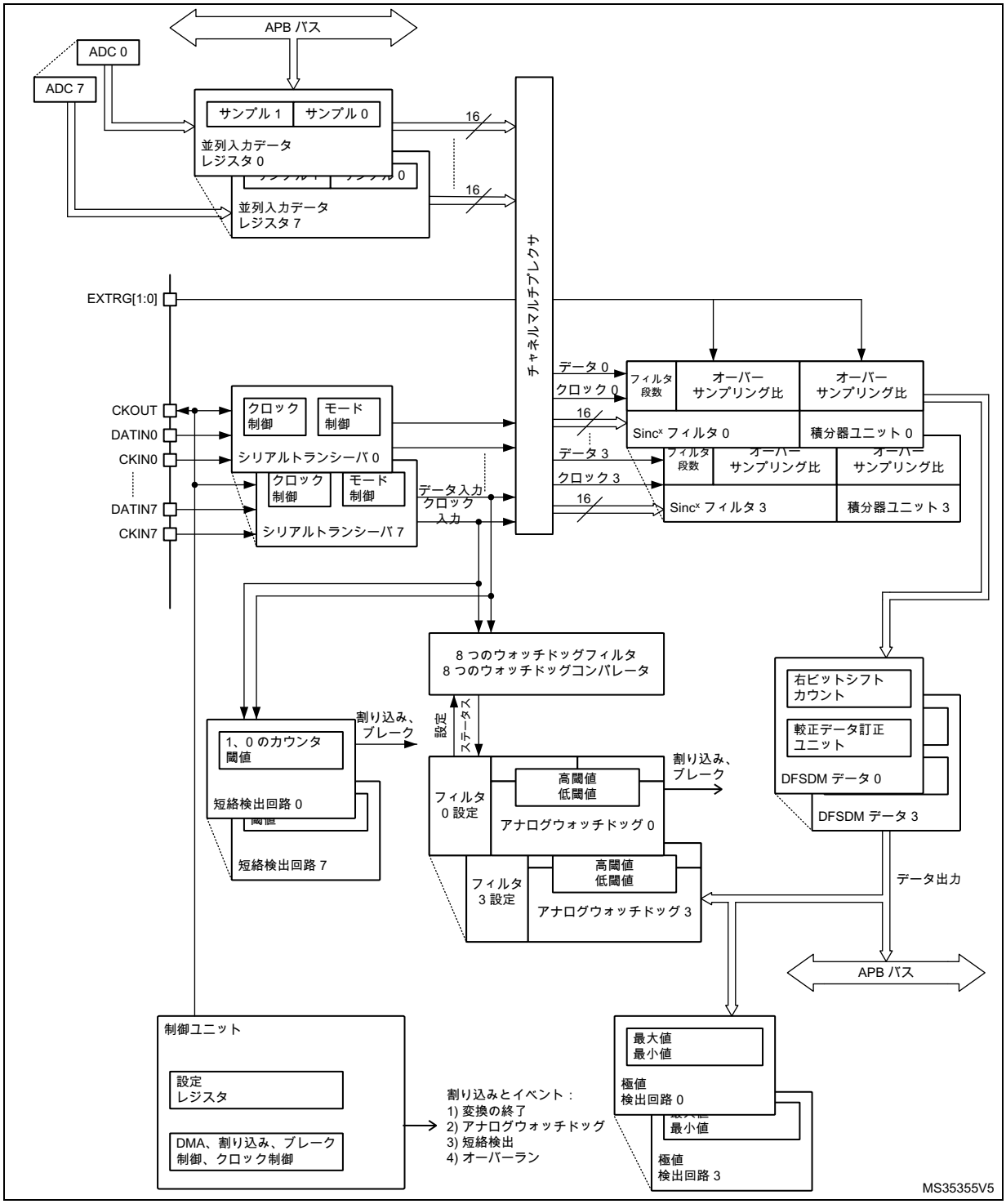
DFSDM の機能	DFSDM1
チャネル数	8
フィルタ数	4
内部 ADC からの入力	X
サポートしているトリガソース	32
パルススキッパ	-
ID レジスタサポート	-

参考資料

30.4 DFSDM の機能詳細

30.4.1 DFSDM ブロック図

図 216. 1 つの DFSDM ブロック図



1. この例では、4 つの DFSDM フィルタと 8 つの入力チャネル（最大設定）を示します。

30.4.2 DFSDM ピンおよび内部信号

表 229. DFSDM 外部ピン

名前	信号タイプ	説明
VDD	電源	デジタル電源供給。
VSS	電源	デジタルグランド電源供給。
CKIN[7:0]	クロック入力	外部デルタシグマモジュレータから供給されるクロック信号。FT 入力。
DATIN[7:0]	データ入力	外部デルタシグマモジュレータから供給されるデータ信号。FT 入力。
CKOUT	クロック出力	外部デルタシグマモジュレータにクロック信号を供給するためのクロック出力。
EXTRG[1:0]	外部トリガ信号	アナログ変換を開始するための 2 つの EXTI 信号からの入力トリガ（GPIO：EXTI11、EXTI15）。

表 230. DFSDM 内部信号

名前	信号タイプ	説明
dfsdm_jtrg[31:0]	内部／ 外部トリガ信号	アナログ変換を開始するための内部／外部トリガソースからの入力トリガ（内部ソースからは同期入力、外部ソースからは同期をともなった非同期入力）。詳細は、*表 231* を参照してください。
dfsdm_break[3:0]	ブレーク信号出力	アナログウォッチドッグや短絡検出回路からのブレーク信号イベント生成。
dfsdm_dma[3:0]	DMA リクエスト信号	各 DFSDM_FLTx（x=0～3）からの DMA リクエスト信号：インジェクト変換イベントの終了。
dfsdm_it[3:0]	割り込みリクエスト信号	各 DFSDM_FLTx の割り込み信号（x = 0～3）
dfsdm_dat_adc[15:0]	ADC 入力データ	並列入力としての最大 4 つの内部 ADC データバス。

表 231. DFSDM トリガ接続

トリガ名	トリガソース
dfsdm_jtrg0	TIM1_TRGO
dfsdm_jtrg1	TIM1_TRGO2
dfsdm_jtrg2	TIM8_TRGO
dfsdm_jtrg3	TIM8_TRGO2
dfsdm_jtrg4	TIM3_TRGO
dfsdm_jtrg5	TIM4_TRGO
dfsdm_jtrg6	TIM16_OC1
dfsdm_jtrg7	TIM6_TRGO
dfsdm_jtrg8	TIM7_TRGO
dfsdm_jtrg9	HRTIM1_ADCTRG1
dfsdm_jtrg10	HRTIM1_ADCTRG3

表 231. DFSDM トリガ接続（続き）

トリガ名	トリガソース
dfsdm_jtrg[23:11]	予約済み
dfsdm_jtrg24	EXTI11
dfsdm_jtrg25	EXTI15
dfsdm_jtrg26	LPTIMER1
dfsdm_jtrg27	LPTIMER2
dfsdm_jtrg28	LPTIMER3
dfsdm_jtrg[31:29]	予約済み

表 232. DFSDM ブレーク接続

ブレーク名	ブレーク転送先
dfsdm_break[0]	TIM15 ブレーク
dfsdm_break[1]	TIM16 ブレーク 2
dfsdm_break[2]	TIM1/TIM17/TIM8 ブレーク
dfsdm_break[3]	TIM1/TIM8/ブレーク 2

30.4.3 DFSDM のリセットおよびクロック

DFSDM のオン／オフ制御

DFSDM インタフェースは、DFSDM_CH0CFGR1 レジスタで DFSDMEN=1 をセットすることによって全体的に有効になります。DFSDM が全体的に有効になると、すべての入力チャネル（y=0～7）およびデジタルフィルタ DFSDM_FLTx（x=0～3）はイネーブルビット（DFSDM_CHyCFGR1 の チャネルイネーブルビット CHEN および DFSDM_FLTxCR1 の DFSDM_FLTx イネーブルビット DFEN）がセットされた場合に動作を開始します。

デジタルフィルタ x DFSDM_FLTx（x=0～3）は、DFSDM_FLTxCR1 レジスタで DFEN=1 をセットすることによって有効になります。DFSDM_FLTx が有効（DFEN=1）になると、$Sinc^x$ デジタルフィルタユニットおよび積分器ユニット両方が再初期化されます。

DFEN をクリアすると、変換が進行中であった場合、即座に停止し、DFSDM_FLTx が STOP モードに入ります。DFSDM_FLTxAWSR および DFSDM_FLTxISR （リセット済み）を除き、すべてのレジスタ設定は変更されません。

チャネル y（y=0～7）は、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって有効になります。チャネルが有効になると、外部デルタシグマモジュレータまたは並列内部データソース（メモリからの ADC または CPU/DMA ワイヤ）からシリアルデータを受信します。

DFSDM は、デバイスが STOP モードに入るためにシステムクロックを停止する前に全体的に無効（DFSDM_CH0CFGR1 の DFSDMEN=0）にしておく必要があります。

DFSDM クロック

内部 DFSDM クロック $f_{DFSDMCLK}$ は、チャネルトランシーバ、デジタル処理ブロック（デジタルフィルタ、積分器）および次に追加されるブロック（アナログウォッチドッグ、短絡検出回路、極値検出回路、制御ブロック）の駆動に使用され、RCC ブロックにより生成され、システムクロック SYSCLK またはペリフェラルクロック PCLK2 から生成されます。DFSDM クロックは STOP モード（すべての DFSDM_FLTx、x=0～3 で DFEN = 0 の場合）で自動的に停止します。

DFSDM シリアルチャネルトランシーバは、外部シリアルクロックを受信して、外部シリアルデータストリームをサンプリングできます。内部 DFSDM クロックは、標準 SPI 符号化を使用する場合、外部シリアルクロックの 4 倍以上、マンチェスタ符号化を使用する場合、6 倍以上の速さが必要になります。

DFSDM は外部出力クロック信号を供給して、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT ピンから供給されます。この出力クロック信号は、特定のデバイスのデータシートに指定された範囲内である必要があり、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）からプログラム可能な分周器によって範囲 2～256（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）で生成されます。オーディオクロックソースは、RCC 設定の SAI1SEL[1:0] フィールドで選択された SAI1 クロックです（を参照）。

30.4.4 シリアルチャネルトランシーバ

8 つの多重化シリアルデータチャネルがあり、各フィルタによる変換、アナログウォッチドッグもしくは短絡検出回路に入力できます。これらのシリアルトランシーバは、外部デルタシグマモジュレータからデータストリームを受信します。データストリームは SPI フォーマットまたはマンチェスタ符号化フォーマットで送信できます（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。
チャネルは、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって動作が有効になります。

チャネル入力の選択

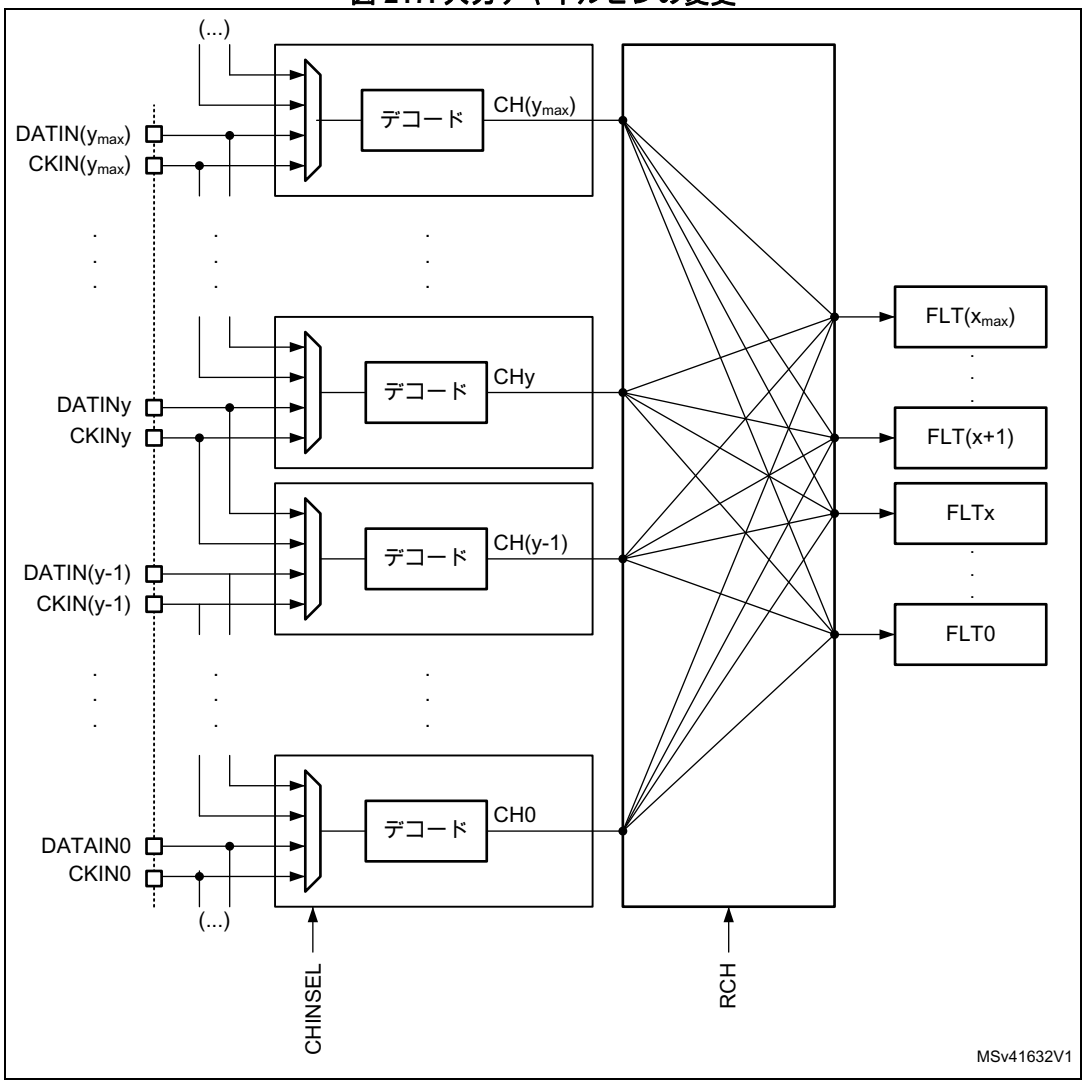
DATINy ピンおよび CKINy ピンからのシリアル入力（データおよびクロック信号）は、次のチャネルピンから変更できます。このシリアル入力チャネル変更は DFSDM_CHyCFGR1 レジスタの CHINSEL ビットでセットされます。

チャネル変更は PDM（パルス密度変調）ステレオマイクロフォンタイプからのオーディオデータの収集に使用できます。PDM ステレオマイクロフォンにはデータ信号とクロック信号が 1 つずつあります。データ信号は、左右のオーディオチャネルの情報を供給します（左チャネルは立ち上がりクロックエッジサンプル、右チャネルは立ち下がりクロックエッジサンプル）。

PDM マイクロフォン入力のシリアルチャネルの設定：

- PDM マイクロフォン信号（データ、クロック）は、DFSDM 入力シリアルチャネル y（DATINy、CKOUT）ピンに接続されます。
- チャネル y は次のように設定されます。CHINSEL = 0（特定のチャネルピンからの入力：DATINy、CKINy）。
- チャネル（y-1）（モジュロ 8）は次のように設定されます。CHINSEL = 1（次のチャネル（（y-1）+1）ピンからの入力：DATINy、CKINy）。
- チャネル y：SITP[1:0] = 0（データストローブのための立ち上がりエッジ）=> チャネル y の左オーディオチャネル。
- チャネル（y-1）：SITP[1:0] = 1（データストローブのための立ち下がりエッジ）=> チャネル y-1 の右オーディオチャネル。
- 2 つの DFSDM フィルタがチャネル y およびチャネル（y-1）に割り当てられます（PDM マイクロフォンからの左右チャネルをフィルタするため）。

図 217. 入力チャネルピンの変更



出力クロック生成

クロック信号は CKOUT ピンで供給され、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT 信号の周波数は、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）から前置分周器（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）によって生成されます。出力クロックを停止すると、CKOUT 信号はロー状態にセットされます（出力クロックは DFSDM_CHyCFGR1 レジスタの CKOUTDIV=0 または DFSDM_CH0CFGR1 レジスタの DFSDMEN=0 で停止できます）。出力クロック停止は次に対して実行されます。

- DFSDMEN クリア後の 4 つのシステムクロック（CKOUTSRC=0 の場合）
- DFSDMEN クリア後の 1 つのシステムクロックおよび 3 つのオーディオクロック（CKOUTSRC=1 の場合）

CKOUTSRC を変更する前に、ソフトウェアは CKOUT ピンのグリッチを回避するために CKOUT の停止を待つ必要があります。出力クロック信号の周波数は、0～20 MHz の範囲でセットする必要があります。

SPI データ入力フォーマットの操作

SPI フォーマットでは、データストリームはデータ信号およびクロック信号を介してシリアルフォーマットで送信されます。データ信号は常に DATINy ピン から供給されます。クロック信号は外部では CKINy ピンから、内部では CKOUT 信号ソースから生成される信号で供給できます。

外部クロックソースを選択する場合（SPICKSEL[1:0]=0）、データ信号（DATINy ピン）は立ち上がりクロックエッジまたは立ち下がりクロックエッジ（CKINy ピン）で SITP[1:0] ビットの設定（DFSDM_CHyCFGR1 レジスタ）に応じてサンプルされます。

内部クロックソース - DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] を参照：

- CKOUT 信号：
  - クロック入力（CKOUT から）に直接使用している外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合
  - サンプリングポイント：SITP[1:0] 設定に応じた立ち上がり／立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち上がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち上がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち下がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち上がりエッジ

注： ***内部クロックソースは、外部デルタシグマモジュレータが CKOUT 信号をクロック入力として（同期クロックおよびデータ操作のために）使用する場合のみ使用できます。***

内部クロックソースの使用で、CKINy ピンを接続しないで済みます（CKINy ピンは他の目的に使用できます）。

クロックソースの信号の周波数は、SPI 符号化では 0～20 MHz の範囲で $f_{DFSDMCLK}/4$ 未満にセットする必要があります。

マンチェスタ符号化データ入力フォーマットの操作

マンチェスタ符号化フォーマットでは、データストリームは DATINy ピンのみを介してシリアルフォーマットで送信されます。デコードされたデータおよびクロック信号は、マンチェスタデコードの後、シリアルストリームから復元されます。マンチェスタ符号化で可能な設定は 2 つあります（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。

- 信号立ち上がりエッジ = log 0; 信号立ち下がりエッジ = log 1
- 信号立ち上がりエッジ = log 1; 信号立ち下がりエッジ = log 0

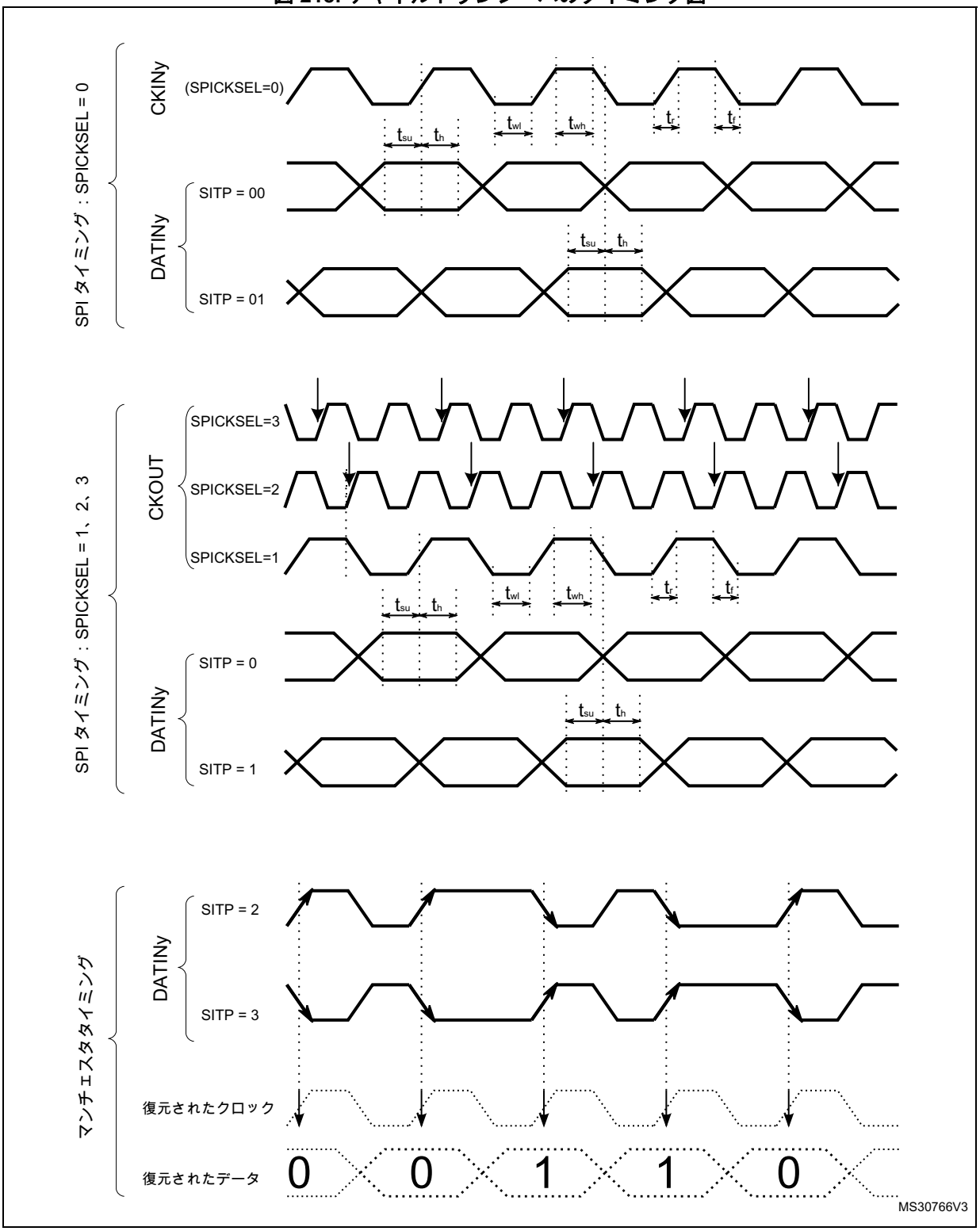
マンチェスタ符号化の復元されたクロック信号周波数は、
0～10 MHz および $f_{DFSDMCLK}/6$ 未満の範囲でなければなりません。

マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

参考資料

図 218. チャネルトランシーバのタイミング図



クロック無検出

チャネルシリアルクロック入力は、クロック有無のチェックが可能で、変換およびエラーレポートの正しい動作を保証します。クロック無検出は、DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって、各入力チャネル y で有効/無効にできます。有効にした場合、このクロック無検出は特定のチャネルで連続して実行されます。クロック無フラグがセット（CKABF[y] = 1）され、入力クロックエラーの際に割り込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。クロック無ステータスビット CKABF[y] は、対応チャネル y が無効である場合、ハードウェアでもセットされます（CHEN[y] = 0 の場合、CKABF[y] はセットで保持されます）。

クロック無イベントが発生した場合、データ変換（またはアナログウォッチドッグおよび短絡検出回路）は誤ったデータを提供します。ユーザはこのイベントを管理し、クロック無がレポートされた際に特定のデータを破棄する必要があります。

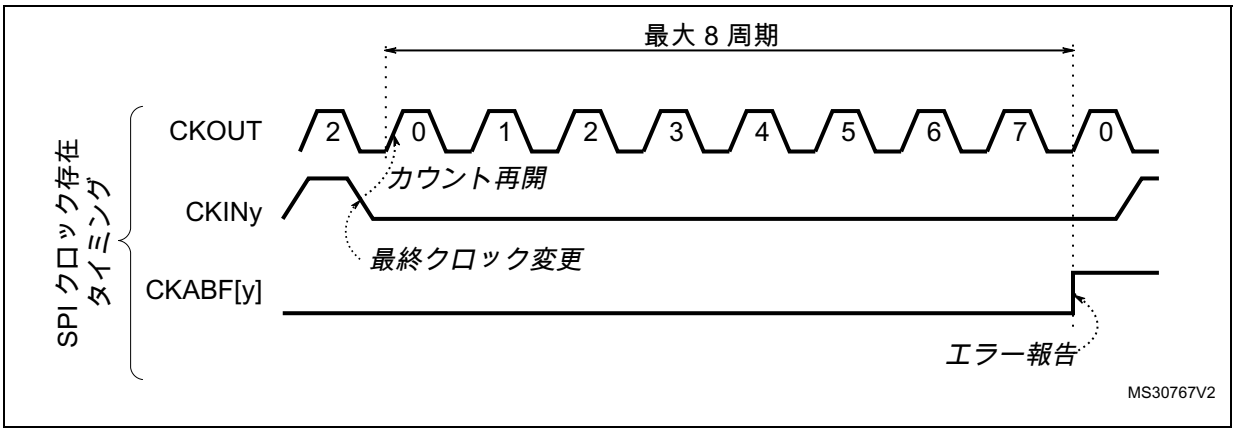
クロック無機能は、システムクロックが CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTSRC=0）に使用されている場合のみ使用可能です。

トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。クロック無検出機能を含むソフトウェアシーケンスは次のとおりにする必要があります。

- CHEN = 1 で特定のチャネルを有効化します。
- クロック無フラグが実際にクリア（CKABF = 0）されるまで、（CLRCKABF = 1 によって）クロック無フラグのクリアを試みます。この時に、トランシーバが同期され（信号クロックが有効になり）、データを受信できます。
- クロック無機能 CKABEN = 1 と関連する割り込み CKABIE = 1 を有効にし、SPI クロックが損失したか、マンチェスタデータエッジが不足しているかを検出します。

SPI データフォーマットを使用している場合、クロック無検出は出力クロック生成（CKOUT 信号）のある外部入力クロックの比較に基づきます。入力チャネルへの外部入力クロック信号は、CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV フィールドによって制御）の 8 信号周期に最低 1 回変更する必要があります。

図 219. SPI でのクロック無タイミング検出図



マンチェスタデータフォーマットを使用している場合、クロック無はクロック復元がマンチェスタ符号化信号から実行できないことを示します。正しいクロック復元には、まず 1 から 0 または 0 から 1 の遷移を持つデータを受信する必要があります（マンチェスタ同期については、*図 221* を参照）。

マンチェスタ符号化でのクロック無の検出（最初に成功した同期の後）は、出力クロック生成（CKOUT 信号）のある符号化されたシリアルデータ入力信号の変更比較に基づきます。CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV ビットによって制御）の 2 周期間に DATINy ピンの電圧レベル変更が必要になります。この条件は、正しくマンチェスタ符号化データとクロック信号を復元できるように最低データレートも定義します。

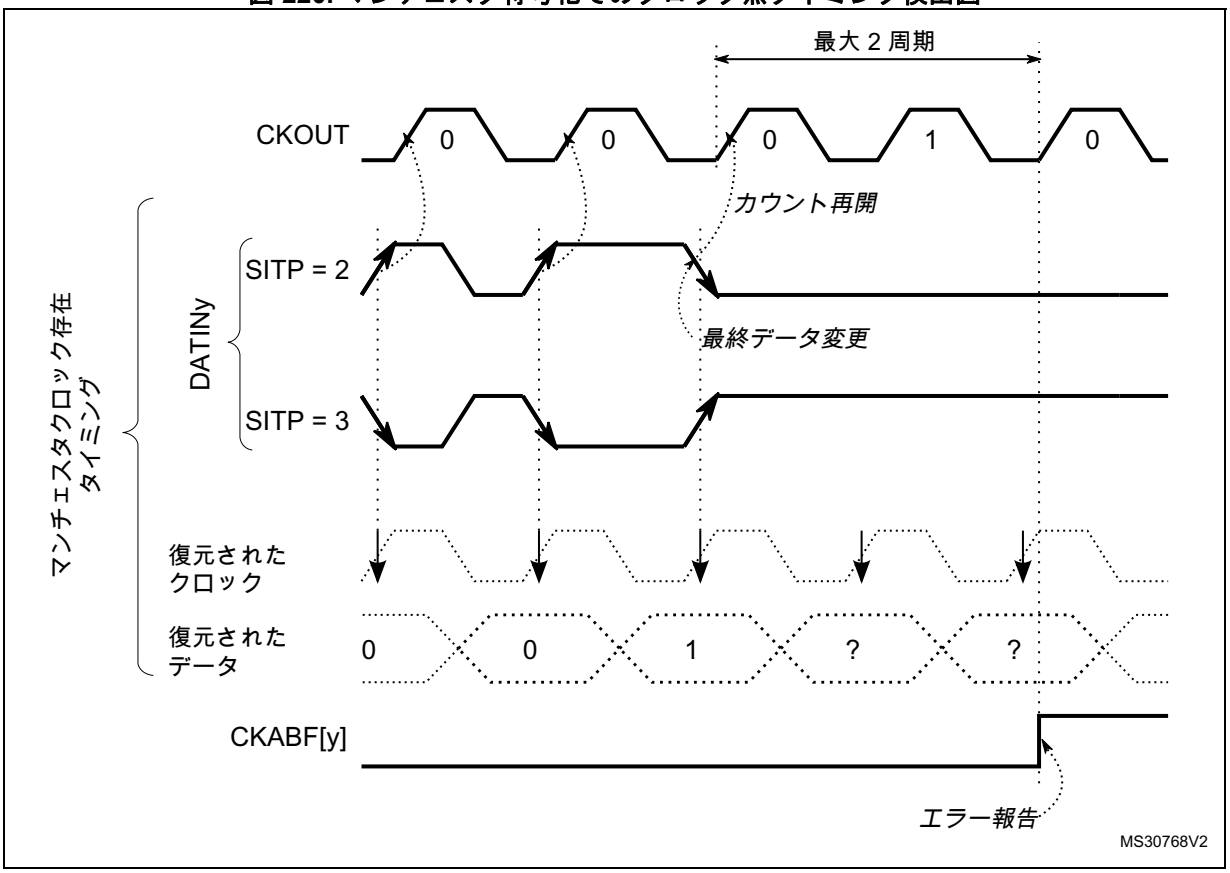
マンチェスタ符号化データの最大データレートは、CKOUT 信号未満とする必要があります。

そのため、マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器は、次の式に応じてセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

クロック無フラグがセット（CKABF[y] = 1）され、入力クロック復元エラーの際に割り込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。

図 220. マンチェスタ符号化でのクロック無タイミング検出図



マンチェスタ／SPI のコード同期

チャネルを有効にした直後、マンチェスタ符号化ストリームを同期する必要があります（DFSDM_CHyCFGR1 レジスタの CHEN=1）。0 から 1 または 1 から 0 のデータ遷移を（有効なデータエッジを検出可能にするために）受信した時に同期が終了します。同期の終了は、以下に詳述しているソフトウェアシーケンスに従って DFSDM_FLT0ICR の CLRCKABF[y] によってクリアされた後、特定のチャネルの CKABF[y]=0 をポーリングしてチェックできます。

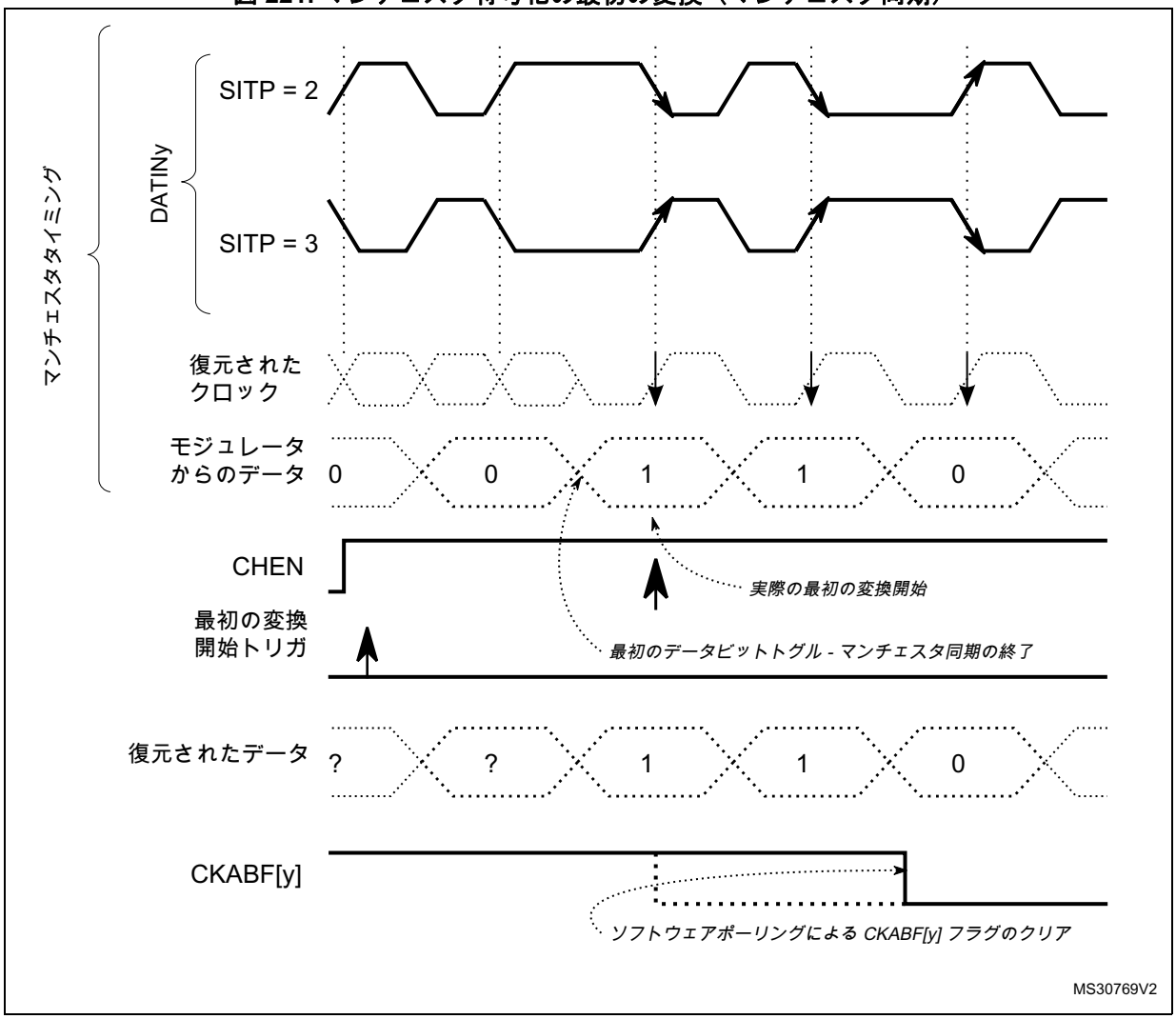
CLRCKABF[y] ビットをセットすることによって、CKABF[y] フラグはクリアされます。チャネル y がまだ同期されていない場合、ハードウェアは即座に CKABF[y] フラグをセットします。ソフトウェアは CKABF[y] フラグを読み出してセットされている場合、CLRCKABF[y] ビットをセットすることによってこのフラグのクリアを再実行します。このソフトウェアシーケンス（CKABF[y] フラグのポーリング）は、CKABF[y] フラグがセットされる（マンチェスタストリームが同期されていることを通知する）まで続行します。マンチェスタ符号化データを同期／受信できるようにするには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

SPI 符号化ストリームはクロック入力信号（有効な立ち上がり／立ち下がりエッジ）を最初に検出した後に同期されます。

注： ***トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。***

図 221. マンチェスタ符号化の最初の変換（マンチェスタ同期）



外部シリアルクロック周波数の測定

チャネルシリアルクロック入力周波数の測定で、アプリケーションに重要な外部デルタシグマモジュレータの実際のデータレートを確認できます。

外部シリアルクロック入力周波数は、1 回の変換時間内にタイマカウント DFSDM クロック（$f_{DFSDMCLK}$）で測定できます。カウントは、変換開始（レギュラまたはインジェクト）後の最初の入力データクロックで開始し、変換終了（変換フラグの終了がセット）前の直前の入力データクロックまでに終了します。変換が終了（JEOCF=1 または REOCF=1）したときに、各変換時間（最初から最後のシリアルサンプルまでの時間）が DFSDM_FLTxCNVTIMR レジスタのカウンタ CNVCNT[27:0] で更新されます。その後、ユーザはデジタルフィルタ設定（FORD、FOSR、IOSR、FAST）に応じてデータレートの計算ができます。フィルタがバイパスされた場合（FOSR=0、積分器のみ有効、DFSDM_FLTxCNVTIMR レジスタで CNVCNT[27:0]=0）のみ、外部シリアル周波数の測定は停止します。

並列データ入力（*セクション 30.4.6：並列データ入力*）の場合、測定された周波数は、1 回の変換中の平均入力データレートです。

注： ***変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、割り込み時間も CNVCNT[27:0] でカウントされます。そのため、正しい変換時間結果を得るには変換に割り込みが発生しないようにすることを推奨します。***

変換時間：

FAST = 0 であるインジェクト変換またはレギュラ変換（または FAST=1 の場合の最初の変換）：

$Sinc^x$ フィルタ（x = 1～5）の場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1 であるレギュラ変換（最初の変換を除く）：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、

- f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または入力データレート（並列データ入力の場合）です。
- F_{OSR} は、フィルタのオーバーサンプリング比です。F_{OSR} = FOSR[9:0]+1（DFSDM_FLTxFCR レジスタを参照）
- I_{OSR} は、積分器のオーバーサンプリング比です。I_{OSR} = IOSR[7:0]+1（DFSDM_FLTxFCR レジスタを参照）
- F_{ORD} は、フィルタ段数です。F_{ORD} = FORD[2:0]（DFSDM_FLTxFCR レジスタを参照）

チャネルのオフセット設定

各チャネルには独自のオフセット設定（レジスタ）があり、特定のチャネルの各変換結果（インジェクトまたはレギュラ）から最終的に減算されます。オフセット補正は、データの右ビットシフトの後に実行されます。オフセットは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] フィールドに 24 ビット符号値として保存されます。

データの右ビットシフト

24 ビット値に整列された結果を得るために、各チャネルは多くの右ビットシフトを定義し、特定のチャネルの各変換結果（インジェクトまたはレギュラ）に適用します。データのビットシフト数は、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットで保存されます。

右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 ビット符号フォーマットを持つために維持されます。

30.4.5 入力シリアルインタフェースの設定

次のパラメータは入力シリアルインタフェースに合わせて設定する必要があります。

- **出力クロックの前置分周器**。DFSDM クロック（2～256）から出力クロックを生成するためのプログラム可能な前置分周器があります。これは、DFSDM_CH0CFGR1 レジスタの CKOUTDIV[7:0] ビットによって定義されます。
- **シリアルインタフェースのタイプおよび入力クロックの位相。**SPI 符号化またはマンチェスタ符号化の選択および入力クロックのサンプリングエッジ。これは、DFSDM_CHyCFGR1 レジスタの SITP [1:0] ビットによって定義されます。
- **入力クロックソース。**CKINy ピンからの外部ソースまたは CKOUT ピンからの内部ソース。これは、DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] フィールドによって定義されます。
- **最終データの右ビットシフト。**24 ビット値に整列された結果を得るために最終データの右ビットシフトを定義します。これは、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] によって定義されます。
- **チャネルあたりのチャネルオフセット。**特定のシリアルチャネルのアナログオフセット（接続された外部デルタシグマモジュレータのオフセット）を定義します。これは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] ビットによって定義されます。
- **チャネルごとの短絡検出回路およびクロック無の有効化。**DFSDM_CHyCFGR1 レジスタの特定のシリアルチャネルで短絡検出回路（SCDEN ビット）およびクロック無監視（CKABEN ビット）を有効化／無効化します。
- **アナログウォッチドッグフィルタや短絡検出回路の閾値設定。**アナログウォッチドッグフィルタのパラメータおよびチャネル短絡検出回路パラメータを設定します。設定は、DFSDM_CHyAWSCDR レジスタで定義されます。

30.4.6 並列データ入力

各入力チャネルには、16 ビット並列データ入力（シリアルデータ入力を除く）用のレジスタがあります。各 16 ビット並列入力は、次の内部データソースからのみ転送できます。

- 内部 ADC 結果(2)
- 直接 CPU/DMA 書き込み。

特定のチャネルのシリアルデータ入力または並列データ入力の選択は、DFSDM_CHyCFGR1 レジスタの DATMPX[1:0] フィールドで行われます。DATMPX[1:0] は次の並列データソースも定義します。内部 ADC (2) または CPU/DMA による直接書き込み。

各チャネルには、32 ビットデータ入力レジスタ DFSDM_CHyDATINR があり、16 ビットデータで書き込むことができます。データは 16 ビット符号フォーマットです。これらのデータは 16 ビット並列データを受け取るデジタルフィルタへの入力として使用できます。

シリアルデータ入力が選択されている場合（DATMPX[1:0] = 0）、DFSDM_CHyDATINR レジスタは書き込み保護されます。

内部 ADC からの入力 (2)

ADC データ並列入力（DATMPX[1:0]=1）の場合、ADC[y+1] の結果はチャネル y 入力に割り当てられます（ADC1 は DFSDM_CHDATIN0R レジスタを、ADC2 は DFSDM_CHDATIN1R レジスタを、と続き、ADC8 は DFSDM_CHDATIN7R レジスタを埋めます）。ADC[y+1] からの変換イベントが終了すると、チャネル y データが更新されます（ADC[y+1] からの並列データはデジタルフィルタへの次のサンプル

2. ADC1 および ADC2 のみ。

として配置されます）。ADC[y+1] からのデータは、変換イベント終了時に DFSDM_CHyDATINR レジスタ（フィールド INDAT0[15:0]）に書き込まれます。

ADC データ入力の際に、データパッキングモードの設定（DFSDM_CHyCFGR1 レジスタの DATPACK[1:0]）にはまったく影響はありません。

注： ***ADC 仕様の拡張：内部 ADC がインタリーブモードに設定される場合（ADC1 とともに ADC2 など、ADC 仕様を参照）、ADC1 または ADC2 からの各結果は同じ 16 ビットバス（ADC のバス）に送られ、DFSDM チャネル 0（固定接続）となります。そのため、DFSDM チャネル 0 への2倍の入力データレートが発生します（ADC1 からの偶数サンプルおよび ADC2 からの奇数サンプル）。ADC2 に関連付けられたチャネル 1 は開放されます。***

メモリからの入力（直接 CPU/DMA 書き込み）

CPU または DMA（DATMPX[1:0]=2）による DFSDM_CHyDATINR レジスタへの直接データ書き込みは、メモリまたはペリフェラルからのデジタルデータストリームを処理するためのデータ入力に使用できます。

データは、CPU または DMA によって DFSDM_CHyDATINR レジスタに書き込むことができます。

1. **CPU データ書き込み**：

   入力データは、CPU によって DFSDM_CHyDATINR レジスタに直接書き込まれます。

2. **DMA データ書き込み**：

   DFSDM_CHyDATINR レジスタにメモリバッファからデータを転送するには、DMA をメモリ間転送モードに設定する必要があります。転送先メモリアドレスは DFSDM_CHyDATINR レジスタのアドレスです。データは DMA の転送速度でメモリから DFSDM 並列入力に転送されます。

   この DMA 転送は、DFSDM 変換結果の読み出しに使用される DMA とは異なります。入力データ書き込み用の最初の DMA（メモリ間転送として設定）と、データ結果読み出し用の 2 つ目の DMA（ペリフェラルからメモリへの転送として設定）は、同時に使用可能です。

DFSDM_CHyDATINR へのアクセスは、16 ビット幅または 32 ビット幅のいずれかで、1 回の書き込み操作で 1 つまたは 2 つのサンプルをそれぞれロードすることができます。32 ビット入力データレジスタ（DFSDM_CHyDATINR）は、DFSDM_CHyCFGR1 レジスタの DATPACK[1:0] フィールドで定義されるデータパッキング動作モードに応じて、1 つまたは 2 つの 16 ビットデータサンプルで満たすことができます。

1. **標準モード（DATPACK[1:0]=0）**：

   チャネル y の入力データとして使用される DFSDM_CHyDATINR レジスタの INDAT0[15:0] フィールドにサンプルが 1 つだけ保存されます。上位 16 ビット（INDAT1[15:0]）は無視され、書き込み保護されます。デジタルフィルタは、CPU/DMA によって満たされた後にデータレジスタを空にするために 1 回の入力サンプリング（INDAT0[15:0] から）を実行する必要があります。このモードは書き込み動作ごとに 1 つのサンプルをロードするために DFSDM_CHyDATINR レジスタへの 16 ビット CPU/DMA アクセスとともに使用されます。

2. **インタリーブモード（DATPACK[1:0]=1）**：

   DFSDM_CHyDATINR レジスタは、2 つのサンプルバッファとして使用されます。最初のサンプルは INDAT0[15:0] に、2 つ目のサンプルは INDAT1[15:0] に保存されます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 2 回の入力サンプリングを実行する必要があります。このモードは書き込み動作ごとに 2 つのサンプルをロードするために DFSDM_CHyDATINR レジスタへの 32 ビット CPU/DMA アクセスとともに使用されます。

3. **デュアルモード（DATPACK[1:0]=2）**：

   2 つのサンプルが、DFSDM_CHyDATINR レジスタに書き込まれます。INDAT0[15:0] のデータはチャネル y 用、INDAT1[15:0] のデータはチャネル y+1 用です。INDAT1[15:0] のデータは、次の（y+1）チャネルデータレジスタ DFSDM_CH[y+1]DATINR の INDAT0[15:0] に自動的にコピー

されます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 1 回、チャネル（y+1）から 1 回の 2 回のサンプリングを実行する必要があります。

デュアルモード設定（DATPACK[1:0]=2）は、偶数チャネル数（y = 0、2、4、6）でのみ使用できます。奇数チャネル（y = 1、35、7）をデュアルモードにセットすると、このチャネルの INDAT0[15:0] および INDAT1[15:0] 両方が書き込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

DFSDM_CHyDATINR レジスタデータモードおよびチャネルへのデータサンプルの割り当てについては、*図 222* を参照してください。

図 222. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て



1 つまたは 2 つのサンプルをロードするための DFSDM_CHyDATINR レジスタへの書き込みは、選択した入力チャネル（チャネル y）のデータ収集を有効（チャネル y の変換開始）にしてから実行される必要があります。そうしないと、書き込まれたデータは次の処理で失われます。

たとえば、シングル変換モードおよびインタリーブモードの場合、シングル変換開始前に DFSDM_CHyDATINR へのデータサンプルの書き込みを開始しないでください（変換開始前の DFSDM_CHyDATINR に存在するすべてのデータが破棄されます）。

30.4.7 チャネル選択

インジェクトチャネルグループおよびレギュラチャネルによる変換で選択できる 8 つの多重化チャネルがあります。

インジェクトチャネルグループは、8 チャネルのいずれかまたはすべてを選択したものを指します。DFSDM_FLTxJCHGR レジスタ の JCHG[7:0] は、JCHG[y]=1 がチャネル y を選択していることを示す場合、インジェクトグループのチャネルを選択します。

インジェクト変換はスキャンモード（JSCAN=1）またはシングルモード（JSCAN=0）で動作可能です。スキャンモードでは、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、JCHG[7:0] によって選択されたすべてのチャネルが変換されるまで、次に上位のチャネルを変換し続けます。シングルモード（JSCAN=0）では、選択されたチャネルの 1 つだけを変換し、チャネル選択が次のチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

インジェクト変換は、ソフトウェアまたはトリガから開始できます。レギュラ変換によって割り込まれることはありません。

レギュラチャネルは、8 チャネルのうち 1 つだけ選択したものを指します。DFSDM_FLTxCR1 レジスタの RCH[2:0] は、選択されたチャネルを示します。

レギュラ変換は、ソフトウェア（またはトリガ）から開始できます。連続したレギュラ変換のシーケンスは、インジェクト変換がリクエストされた際に一時的に割り込みが発生します。

無効化されたチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=0）の変換を実行すると、供給される入力データ（クロック信号）がないため変換が終了しなくなります。この場合、特定のチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=1）を有効化するか、DFSDM_FLTxCR1 レジスタを DFEN=0 にして変換を停止する必要があります。

30.4.8 デジタルフィルタ設定

DFSDM には、$Sinc^x$ タイプのデジタルフィルタが実装されています。この $Sinc^x$ フィルタは入力デジタルデータストリームフィルタリングを実行し、出力データレートの減少（デシメーション）および出力データ分解能の向上につなげます。$Sinc^x$ フィルタは設定可能で、必要な出力データレートおよび出力データ分解能を満たすことができます。設定可能なパラメータは次のとおりです。

- フィルタ段数／タイプ：（DFSDM_FLTxFCR レジスタの FORD[2:0] ビットを参照）：
  - FastSinc
  - $Sinc^1$
  - $Sinc^2$
  - $Sinc^3$
  - $Sinc^4$
  - $Sinc^5$
- オーバーサンプリング比／デシメーション率（DFSDM_FLTxFCR レジスタの FOSR[9:0] ビットを参照）
  - FOSR = 1-1024 - FastSinc フィルタおよび $Sinc^x$ フィルタ x = F_{ORD} = 1～3 の場合
  - FOSR = 1-215 - $Sinc^x$ フィルタ x = F_{ORD} = 4 の場合
  - FOSR = 1-73 - $Sinc^x$ フィルタ x = F_{ORD} = 5 の場合

フィルタには次の転送機能（H ドメインのインパルスレスポンス）があります。

- $Sinc^x$ フィルタタイプ： $$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$
- FastSinc フィルタタイプ： $$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

図 223. 例：Sinc^3 フィルタレスポンス：

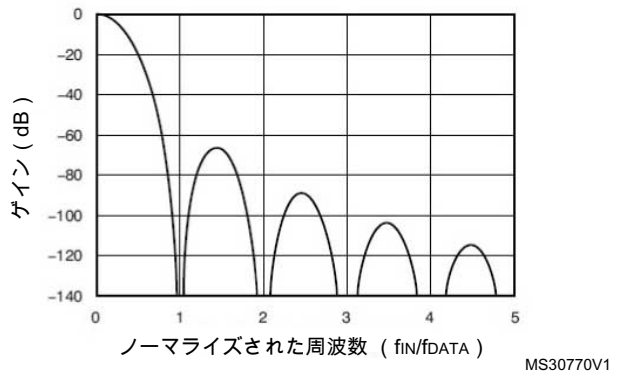


表 233. フィルタの最大出力分解能（フィルタ出力のピークデータ値）
一部の FOSR 値

FOSR	Sinc^1	Sinc^2	FastSinc	Sinc^3	Sinc^4	Sinc^5
x	+/- x	+/- x^2	+/- 2x^2	+/- x^3	+/- x^4	+/- x^5
4	+/- 4	+/- 16	+/- 32	+/- 64	+/- 256	+/- 1024
8	+/- 8	+/- 64	+/- 128	+/- 512	+/- 4096	-
32	+/- 32	+/- 1024	+/- 2048	+/- 32768	+/- 1048576	+/- 33554432
64	+/- 64	+/- 4096	+/- 8192	+/- 262144	+/- 16777216	+/- 1073741824
128	+/- 128	+/- 16384	+/- 32768	+/- 2097152	+/- 268435456	結果はフルスケール入力（> 32 ビット符号整数）でオーバーフローできます
256	+/- 256	+/- 65536	+/- 131072	+/- 16777216		
1024	+/- 1024	+/- 1048576	+/- 2097152	+/- 1073741824		

Sinc フィルタタイプのプロパティおよび使用の詳細については、デジタルフィルタに関する理論を学習されることを推奨します（さらに多くの資料をインターネットからダウンロードできます）。

30.4.9 積分器ユニット

積分器は、デジタルフィルタからのデータにおける追加のデシメーションおよび分解能向上を実行します。積分器は、単純にフィルタからの特定のデータサンプル数に対してデジタルフィルタからのデータの合計を実行します。

積分器のオーバーサンプリング比パラメータは、積分器からの 1 回のデータ出力に合計されるデータカウント数を定義します。IOSR は、1～256 の範囲でセットできます（DFSDM_FLTxFCR レジスタの IOSR[7:0] ビットの説明を参照）

表 234. 積分器の最大出力分解能（積分器出力のピークデータ値）
（一部の IOSR 値、FOSR = 256 および $Sinc^3$ フィルタタイプ（最大データ）の場合）

IOSR	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
x	+/- FOSR. x	+/- $FOSR^2$. x	+/- 2.$FOSR^2$. x	+/- $FOSR^3$. x	+/- $FOSR^4$. x	+/- $FOSR^5$. x
4	-	-	-	+/- 67 108 864	-	-
32	-	-	-	+/- 536 870 912	-	-
128	-	-	-	+/- 2 147 483 648	-	-
256	-	-	-	+/- 2^{32}	-	-

30.4.10 アナログウォッチドッグ

アナログウォッチドッグの目的は、アナログ信号が特定の最大および最低閾値に達した場合に、外部信号（ブレークまたは割り込み）をトリガすることです。この時、割り込み／イベント／ブレーク生成が呼び出されます。

各アナログウォッチドッグは、AWFSEL ビット設定（DFSDM_FLTxCR1 レジスタ）に応じて、シリアルデータレシーバ出力（各チャネルでアナログウォッチドッグフィルタ後）またはデータ出力レジスタ（現在のインジェクト変換またはレギュラ変換の結果）を監視します。アナログウォッチドッグ x での監視対象入力チャネルかどうかは、DFSDM_FLTxCR2 レジスタの AWDCH[7:0] で選択されます。

入力チャネルのアナログウォッチドッグ変換は、標準変換から独立します。この場合、アナログウォッチドッグはメインのインジェクト変換またはレギュラ変換から独立した各入力チャネルで独自のフィルタと信号処理を使用します。アナログウォッチドッグ変換は、メインのインジェクト変換またはレギュラ変換が一時停止（RCIP = 0、JCIP = 0）状態でもチャネルを監視するために、選択された入力チャネルで連続モードで実行されます。

特定のデータ値（DFSDM_FLTxAWHTR レジスタの AWHT[23:0] ビットおよび DFSDM_FLTxAWLTR レジスタの AWLT[23:0] ビットでセット）と比較される高閾値および低閾値があります。

データ値を持つ閾値レジスタを比較する 2 つのオプションがあります。

- オプション 1 : この場合、入力データは最終出力データレジスタ（AWFSEL=0）から取得されます。このオプションは次の特徴があります。
  - 入力データの高分解能（最大 24 ビット）
  - スローな応答時間 - 過電流検出のような高速応答アプリケーションには不適切です。
  - 比較のために最終データはビットシフトおよびオフセットデータ補正後に取得されます。
  - メインのレギュラ変換またはインジェクト変換が実行された後のみ最終データを利用できます。
  - 並列入力データソース（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に使用可能です。
- オプション 2 : この場合、入力データはあらゆるシリアルデータレシーバ出力（AWFSEL=1）から取得されます。このオプションは次の特徴があります。
  - 入力シリアルデータは、設定可能なオーバーサンプリング比（1 ～ 32）およびフィルタ段数（1～3）がある専用のアナログウォッチドッグ $Sinc^x$ チャネルフィルタによって処理されます（DFSDM_CHyAWSCDR レジスタの AWFOSR[4:0] および AWFORD[1:0] ビットの設定を参照）。
  - 低分解能（最大 16 ビット）
  - 高速な応答時間 - 過電流／過電圧検出のような高速応答が必要なアプリケーションに適切です。

– データは、メインのレギュラ変換またはインジェクト変換アクティビティとは独立した連続モードで利用できます。

入力チャネルを監視している場合（AWFSEL=1）、閾値と比較するデータは、AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルから取得されます。選択されたチャネルフィルタの各結果は、1 つの閾値ペア（AWHT[23:0] / AWLT[23:0]）と比較されます。この場合、アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため、上位の 16 ビット（AWHT[23:8] / AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWHT[7:0] / AWLT[7:0] ビットは、この場合（AWFSEL=1）の比較では取得されません。

各入力チャネルのアナログウォッチドッグフィルタ設定のパラメータは、DFSDM_CHyAWSCDR レジスタ（フィルタ段数 AWFORD[1:0] およびフィルタオーバーサンプリング比 AWFOSR[4:0]）でセットされます。

各入力チャネルには、アナログウォッチドッグデータ（アナログウォッチドッグフィルタから）をアナログウォッチドッグの閾値（AWHT/AWLT）を比較する独自のコンパレータがあります。いくつかのチャネル（DFSDM_FLTxCR2 レジスタの AWDCH[7:0] フィールド）が選択されている場合、複数の比較リクエストを同時に受信する場合があります。この場合、最低値のチャネルリクエストがまず管理され、選択されたチャネルを昇順に管理していきます。チャネルごとに、結果を個別のフラグ（DFSDM_FLTxAWSR レジスタの AWHTF[7:0]、AWLTF[7:0] フィールド）で記録できます。各チャネルリクエストは 8 DFSDM クロックサイクルで実行されます。そのため、各チャネルの帯域幅は 8 DFSDM クロックサイクルに制限されます（AWDCH[7:0] = 0xFF の場合）。最大入力チャネルのサンプリングクロック周波数は、4 で割った DFSDM クロック周波数となるため、設定 AWFOSR = 0（アナログウォッチドッグフィルタがバイパス）はこの入力クロック速度でアナログウォッチドッグ機能に使用できません。そのため、ユーザは入力サンプリングクロック速度および DFSDM 周波数の点で、監視されるチャネルの数とアナログウォッチドッグフィルタのパラメータを適切に設定する必要があります。

特定のチャネル y のアナログウォッチドッグフィルタデータは、DFSDM_CHyWDATR レジスタの WDATA[15:0] フィールドでファームウェアによる準備に使用できます。そのアナログウォッチドッグフィルタデータは（DFSDM_CHyCFGR1 レジスタが CHEN=1 の場合）、アナログウォッチドッグフィルタ設定およびチャネル入力クロック周波数によって提供されるデータレートで連続して変換されます。

アナログウォッチドッグフィルタ変換は、積分器のないレギュラ高速連続変換のように機能します。アナログウォッチドッグフィルタ出力からの 1 つの結果に必要なシリアルサンプル数（チャネル入力クロック周波数 f_{CKIN} において）：

最初の変換：

$Sinc^x$ フィルタ（x = 1～5）の場合：サンプル数 = [F_{OSR} * F_{ORD} + F_{ORD} + 1]

FastSinc フィルタの場合：サンプル数 = [F_{OSR} * 4 + 2 + 1]

次の変換：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：サンプル数 = [FOSR * IOSR]

ここで、

F_{OSR} ..... フィルタのオーバーサンプリング比 : F_{OSR} = AWFOSR[4:0]+1（DFSDM_CHyAWSCDR レジスタを参照）

F_{ORD} ..... フィルタ段数：F_{ORD} = AWFORD[1:0]（DFSDM_CHyAWSCDR レジスタを参照）

出力データレジスタを監視する場合（AWFSEL=0）、最終データの右ビットシフトおよびオフセット補正（DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] および DTRBS[4:0] フィールドを参照）後に実行されます。比較は、各インジェクト変換またはレギュラ変換の終了後に AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルに実行されます。

アナログウォッチドッグイベントのステータスは、特定のイベントがラッチされる DFSDM_FLTxAWSR レジスタで信号化されます。AWHTF[y]=1 フラグは、チャネル y を跨ぐ AWHT[23:0] 値を信号化します。AWLTF[y]=1 フラグは、チャネル y を跨ぐ AWLT[23:0] 値を信号化します。DFSDM_FLTxAWSR レジスタでラッチされたイベントは、DFSDM_FLTxAWCFR レジスタの対応するクリアリングビット CLRAWHTF[y] または CLRAWLTF[y] に“1”を書き込むことによってクリアされます。

アナログウォッチドッグのグローバルステータスは、DFSDM_FLTxISR レジスタの AWDF フラグビットによって信号化されます（割り込みソースの高速検出に使用されます）。AWDF=1 は、最低 1 つのウォッチドッグイベントが発生すると信号化します（最低 1 チャネルに対して AWHTF[y]=1 または AWLTF[y]=1）。AWDF ビットはすべての AWHTF[7:0] および AWLTF[7:0] がクリアされたときにクリアされます。

アナログウォッチドッグイベントはブレーク出力信号に割り当てることができます。高閾値または低閾値に跨るイベントに割り当てる 4 つのブレーク出力があります（dfsdm_break[3:0]）。特定のアナログウォッチドッグイベントへのブレーク信号割り当ては、DFSDM_FLTxAWHTR レジスタおよび DFSDM_FLTxAWLTR レジスタの BKAWH[3:0] および BKAWL[3:0] フィールドで実行されます。

30.4.11 短絡検出回路

短絡検出回路の目的は、アナログ信号が飽和した値（フルスケール範囲外）に達してこの値が一定時間残った場合に、非常に高速な応答時間で信号化することです。この挙動は、短絡を検出したり、回路のエラー（例：過電流または過電圧）を明らかにできます。割り込み／イベント／ブレーク生成が呼び出されます。

短絡検出回路への入力データは、チャネルトランシーバ出力から取得されます。

各入力チャネルにアップカウントするカウンタがあり、シリアルデータレシーバ出力で連続した 0 と 1 をカウントしています。受信したデータストリームの変更（データ信号の 1 から 0、0 から 1 への変更）がある場合に、カウンタが再開します。このカウンタは短絡閾値レジスタ値（DFSDM_CHyAWSCDR レジスタの SCDT[7:0] ビット）に達すると、短絡イベントが呼び出されます。各入力チャネルには、短絡検出回路があります。SCDEN ビット（DFSDM_CHyCFGR1 レジスタ）をセットすることで、すべてのチャネルを継続的な監視対象として選択でき、それぞれ独自の短絡検出回路設定があります（SCDT[7:0] ビットの閾値、ステータスビット SCDF[7:0]、ステータスクリアリングビット CLRSCDF[7:0]）。ステータスフラグ SCDF[y] は、対応チャネル y が無効である場合（CHEN[y] = 0）、ハードウェアでもクリアされます。

各チャネルで、短絡検出回路イベントはブレーク出力信号 dfsdm_break[3:0] に割り当てることができます。短絡検出回路イベントに割り当てる 4 つのブレーク出力があります。特定のチャネル短絡検出回路イベントへのブレーク信号割り当ては、DFSDM_CHyAWSCDR レジスタの BKSCD[3:0] フィールドで実行されます。

並列入力データチャネル選択（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に短絡検出回路は使用できません。

4 つのブレーク出力がすべて使用できます（アナログウォッチドッグ機能と共用）。

30.4.12 極値検出回路

極値検出回路の目的は、最終出力データワードの最小値と最大値（ピークトゥピーク値）を収集することです。

出力データワードが、極値検出回路の最大レジスタ（DFSDM_FLTxEXMAX レジスタの EXMAX[23:0] ビット）に保存された値を超える場合、このレジスタは現在の出力データワード値およびデータが EXMAXCH[2:0] ビット（DFSDM_FLTxEXMAX レジスタ）に保存されるチャネルとともに更新されます。

出力データワードが、極値検出回路の最小レジスタ（DFSDM_FLTxEXMIN レジスタの EXMIN[23:0] ビット）に保存された値を下回る場合、このレジスタは現在の出力データワード値およびデータが EXMINCH[2:0] ビット（DFSDM_FLTxEXMIN レジスタ）に保存されるチャネルとともに更新されます。

最小および最大レジスタ値は（特定の DFSDM_FLTxEXMAX または DFSDM_FLTxEXMIN の読み出しで）ソフトウェアによってリフレッシュできます。リフレッシュ後、極値検出回路の最小データレジスタ DFSDM_FLTxEXMIN には 0x7FFFFF（正の最大値）が、最大レジスタ DFSDM_FLTxEXMAX には 0x800000（負の最小値）が入ります。

右ビットシフトおよびオフセットデータ補正後、極値検出回路は比較を実行します。各極値検出回路に対して、極値の計算に考慮される入力チャネルが EXCH[7:0] ビット（DFSDM_FLTxCR2 レジスタ）で選択されます。

30.4.13 データユニットブロック

データユニットブロックは全体の処理パスの最後のブロックです。外部デルタシグマモジュレータ - シリアルトランシーバ - Sinc フィルタ - 積分器 - データユニットブロック。

出力データレートは、シリアルデータストリームレート、フィルタ、積分器の設定に依存します。最大出力データレートは次のとおりです。

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0, Sincx filter}$$

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0, FastSinc filter}$$

または

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot I_{OSR}} \quad \text{...FAST = 1}$$

並列データ入力の場合の最大出力データレート：

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0, Sincx filter}$$

または

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0, FastSinc filter}$$

または

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot I_{OSR}} \quad \text{...FAST=1 またはあらゆるフィルタバイパスの場合 } (F_{OSR} = 1)$$

where: f_{DATAIN_RATE}...input data rate from ADC or from CPU/DMA

最終データ幅は 24 ビットとなり、処理パスからのデータは最大 32 ビットまで可能であるため、最終データの右ビットシフトはこのモジュールで実行されます。この右ビットシフトは選択された各入力チャネルに対して 0～31 ビットの範囲で設定可能です（DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットを参照）。右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 ビット符号フォーマットを持つために維持されます。

次のステップでは、結果のオフセット補正が実行されます。オフセット補正値（レジスタ DFSDM_CHyCFGR2 に保存された OFFSET[23:0]）は、特定のチャネルの出力データから減算されます。OFFSET[23:0] フィールドのデータは、適切な較正ルーチンでソフトウェアによってセットされます。

デジタル処理のすべての動作は 32 ビット符号レジスタで実行されるため、結果がオーバーフローしないように次の条件を満たす必要があります。

FOSR ^{FORD} .IOSR <= 2^{31} ... Sinc^{x} フィルタ、x = 1～5 の場合
2 .FOSR ^{2} .IOSR <= 2^{31} ... FastSinc フィルタの場合

注： ***フィルタおよび積分器バイパスの場合（IOSR[7:0]=0、FOSR[9:0]=0）、すべての出力データを読み出せるように入力データレート（f_{DATAIN_RATE}）を制限する必要があります。***
$f_{DATAIN_RATE} \leq f_{APB}$
ここで、f_{APB} は DFSDM ペリフェラルが接続されているバス周波数です。

30.4.14 符号付きデータフォーマット

DFSDM 入力シリアルチャネルにつき、1 つの外部デルタシグマモジュレータに接続できます。外部デルタシグマモジュレータには、2 つの異なる入力（正と負）を持つことができ、差分またはシングルエンド信号測定に使用できます。

デルタシグマモジュレータ出力は常に符号付きフォーマットで想定されています（デルタシグマモジュレータからの 0 と 1 のデータストリームは -1 および +1 の値を表します）。

レジスタの符号付きデータフォーマット：データは、最終出力データ、アナログウォッチドッグ、極値検出回路、オフセット補正のレジスタで符号付きフォーマットになります。出力データワードの最上位ビットは、値の符号を表します（2 の補数フォーマット）。

30.4.15 変換の開始

インジェクト変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの JSWSTART に“1”を書き込みます。
- トリガ：JEXTSEL[4:0] は、同時に JEXTEN がトリガを有効化してアクティブエッジを選択する際にトリガ信号を選択します（DFSDM_FLTxCR1 レジスタを参照）。
- JSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、インジェクト変換は自動的に開始します。DFSDM_FLT0 の場合、インジェクト変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで JSWSTART=1）。DFSDM_FLTx（x>0）での各インジェクト変換は、常にローカル設定（JSCAN、JCHG など）に応じて実行されます。

スキャン変換が有効である場合（ビット JSCAN=1）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたすべてのチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が最下位のチャネル（選択されている場合チャネル 0）から順番に変換されます。

スキャン変換が無効である場合（ビット JSCAN=0）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が 1 つだけ変換されると、チャネル選択が次に選択されたチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のインジェクトチャネルにセットされます。

指定した時間で実行できるインジェクト変換は 1 回だけです。このため、インジェクト変換の他のリクエストがすでに発行されており、未完である場合、インジェクト変換を開始するあらゆるリクエストは無視されます。

レギュラ変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの RSWSTART に“1”を書き込みます。
- RSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、レギュラ変換は自動的に開始します。DFSDM_FLT0 の場合、レギュラ変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで RSWSTART=1）。DFSDM_FLTx（x>0）での各レギュラ変換は、常にローカル設定（RCONT、RCH など）に応じて実行されます。

指定した時間で保留または実行できるレギュラ変換は 1 回だけです。このため、レギュラ変換の他のリクエストがすでに発行されており、未完である場合、レギュラ変換を開始するあらゆるリクエストは無視されます。レギュラ変換は、インジェクト変換による割り込み、またはインジェクト変換進行中に開始が発生した場合、保留できます。この保留中のレギュラ変換は延期され、すべてのインジェクト変換終了後に実行されます。延期されたレギュラ変換は、DFSDM_FLTxRDATAR レジスタの RPEND ビットで信号化されます。

30.4.16 連続モードおよび高速連続モード

DFSDM_FLTxCR1 レジスタの RCONT をセットすると、レギュラ変換が連続モードで実行されます。RCONT=1 は、RSWSTART に“1”が書き込まれた後に RCH[2:0] によって選択されたチャネルが繰り返し変換されることを示します。

連続モードで実行されるレギュラ変換は、RCONT に“0”を書き込むことで停止できます。RCONT をクリアすると、進行中の変換は即座に停止します。

連続モードでは、DFSDM_FLTxCR1 レジスタの FAST ビットをセットすることによって、データレートを上げることができます。この場合、1 つのチャネルから連続して変換する場合、フィルタ内のデータは前にサンプルした連続データから有効になるため、フィルタに新しいデータを入れる必要はありません。速度の増加は、選択したフィルタ段数に依存します。RSWSTART=1 による連続変換の開始後、高速モード（FAST=1）での最初の変換は、フルタイム（FAST=0 の場合と同じくらい）かかりますが、後続の変換はより短い間隔で終了します。

連続モードでの変換時間：

FAST = 0（または FAST=1 の場合の最初の変換）の場合：

Sinc^x フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1（最初の変換を除く）の場合：

Sinc^x フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

連続モードはインジェクト変換では使用できません。インジェクト変換はタイマトリガによって開始でき、正確なタイミングで連続モードをエミュレートします。

レギュラ連続変換が進行中の場合（RCONT=1）およびレギュラ連続変換をリクエストする DFSDM_FLTxCR1 レジスタに書き込みアクセス（RCONT=1）が実行された場合、レギュラ連続変換は次の変換サイクルから再開されます（DFSDM_FLTxCR1 レジスタに変更がない場合でも、新しいレギュラ連続変換が新しいチャネル選択に適用されるのと同様）。

30.4.17 リクエスト優先度

インジェクト変換は、レギュラ変換より優先されます。すでに進行中のレギュラ変換は、インジェクト変換のリクエストによって即座に割り込まれます。このレギュラ変換は、インジェクト変換終了後に再開されます。

他のインジェクト変換が保留中またはすでに進行中の場合、インジェクト変換は開始できません。インジェクト変換を開始するあらゆるリクエスト（JSWSTART またはトリガいずれかによるもの）は、ビット JCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

同様に、他のレギュラ変換が保留中またはすでに進行中の場合、レギュラ変換は開始できません。レギュラ変換を開始するあらゆるリクエスト（RSWSTART によるもの）は、ビット RCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

しかし、レギュラ変換がすでに進行中の状態でインジェクト変換がリクエストされた場合、レギュラ変換は即座に停止し、インジェクト変換が開始されます。その後、レギュラ変換は再開し、この延期された開始はビット RPEND で信号化されます。

インジェクト変換は一時的に連続レギュラ変換のシーケンスに割り込める点で、レギュラ変換より優先されます。インジェクト変換のシーケンスが終了すると、RCONT がまだセットされている場合、連続レギュラ変換が再開します（また、RPEND ビットは最初のレギュラ変換結果において延期した開始を信号化します）。

優先度には、DFSDM に同じ書き込みによっていつアクションが開始されたか、他のアクションの最後に複数のアクションが保留されているかどうかが影響します。たとえば、インジェクト変換の処理中（JCIP=1）に、DFSDM_FLTxCR1 へのシングル書き込み操作があり、RSWSTART に“1”が書き込まれ、レギュラ変換をリクエストしたとします。インジェクト変換シーケンスが終了すると、優先度からレギュラ変換を次に実行して、延期された開始が RPEND ビットで信号化されることになります。

30.4.18 RUN モードでの省電力

消費電力を低減するために、DFSDM フィルタおよび積分器は、変換で使用していない場合に自動的にアイドル状態に入ります（RCIP=0、JCIP=0）。

30.5 DFSDM 割り込み

CPU 性能を向上させるため、CPU イベント発生に関連する割り込みが実装されています。

- インジェクト変換終了割り込み：
  - DFSDM_FLTxCR2 レジスタの JEOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JEOCF ビットで示されます。
  - DFSDM_FLTxJDATAR レジスタの読み出しによってクリアされます（インジェクトデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxJDATAR レジスタの JDATACH[2:0] ビットでレポートされます。
- レギュラ変換終了割り込み：
  - DFSDM_FLTxCR2 レジスタの REOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの REOCF ビットで示されます。
  - DFSDM_FLTxRDATAR レジスタの読み出しによってクリアされます（レギュラデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxRDATAR レジスタの RDATACH[2:0] ビットでレポートされます。

- インジェクト変換のデータオーバーラン割り込み：
  - インジェクト変換データが（CPU または DMA によって）DFSDM_FLTxJDATAR レジスタから読み出されず、新しいインジェクト変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの JOVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JOVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRJOVRF ビットに“1”を書き込むことによってクリアされます。
- レギュラ変換のデータオーバーラン割り込み：
  - レギュラ変換データが（CPU または DMA によって）DFSDM_FLTxRDATAR レジスタから読み出されず、新しいレギュラ変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの ROVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの ROVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRROVRF ビットに“1”を書き込むことによってクリアされます。
- アナログウォッチドッグ割り込み：
  - 変換されたデータ（DFSDM_FLTxCR1 レジスタの AWFSEL ビット設定に応じた出力データまたはアナログウォッチドッグフィルタからのデータ）が DFSDM_FLTxAWHTR / DFSDM_FLTxAWLTR レジスタの高／低閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの AWDIE ビット（選択されたチャネル AWDCH[7:0]）によって有効になります。
  - DFSDM_FLTxISR レジスタの AWDF ビットで示されます。
  - DFSDM_FLTxAWSR レジスタの AWHTF[7:0] フィールドおよび AWLTF[7:0] フィールドによってアナログウォッチドッグの高閾値または低閾値を個別に示します。
  - DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[7:0] ビットまたは CLRAWLTF[7:0] ビットに“1”を書き込むことによってクリアされます。
- 短絡検出回路割り込み：
  - 安定したデータ数が DFSDM_CHyAWSCDR レジスタの閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの SCDIE ビット（DFSDM_CHyCFGR1 レジスタの SCDEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの SCDF[7:0] ビットで示されます（短絡検出回路イベントが発生したチャネルもレポート）。
  - DFSDM_FLTxICR レジスタの対応する CLRSCDF[7:0] ビットに“1”を書き込むことによってクリアされます。
- チャネルクロック無割り込み：
  - CKINy ピンにクロックがない場合に発生します（*セクション 30.4.4：シリアルチャネルトランシーバ*の*クロック無検出*を参照）
  - DFSDM_FLTxCR2 レジスタの CKABIE ビット（DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの CKABF[y] ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRCKABF[y] ビットに“1”を書き込むことによってクリアされます。

表 235. DFSDM 割り込みリクエスト

割り込みイベント	イベントフラグ	イベント／割り込みのクリア方法	割り込み有効制御ビット
インジェクト変換終了	JEOCF	DFSDM_FLTxJDATAR を読み出す	JEOCIE
レギュラ変換終了	REOCF	DFSDM_FLTxRDATAR を読み出す	REOCIE
インジェクトデータオーバーラン	JOVRF	CLRJOVRF = 1 を書き込む	JOVRIE
レギュラデータオーバーラン	ROVRF	CLRROVRF = 1 を書き込む	ROVRIE
アナログウォッチドッグ	AWDF、AWHTF[7:0]、AWLTF[7:0]	CLRAWHTF[7:0] = 1 を書き込む CLRAWLTF[7:0] = 1 を書き込む	AWDIE、(AWDCH[7:0])
短絡検出回路	SCDF[7:0]	CLRSCDF[7:0] = 1 を書き込む	SCDIE、（SCDEN）
チャネルクロック無	CKABF[7:0]	CLRCKABF[7:0] = 1 を書き込む	CKABIE、（CKABEN）

30.6 DFSDM の DMA 転送

CPU 介入を低減するために、変換を DMA 転送を使用してメモリに転送することができます。インジェクト変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの JDMAEN=1 ビットをセットすることで有効になります。レギュラ変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの RDMAEN=1 ビットをセットすることで有効になります。

注： ***DMA 転送で、DMA が DFSDM_FLTxJDATAR レジスタまたは DFSDM_FLTxRDATAR レジスタを読み出すため、割り込みフラグはインジェクト変換またはレギュラ変換（DFSDM_FLTxISR レジスタの JEOCF または REOCF ビット）の最後に自動的にクリアされます。***

30.7 DFSDM チャネル y レジスタ（y=0～7）

30.7.1 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR1）（y=0～7）

このレジスタは、チャネル y（y = 0～7）によって使用されるパラメータを指定します。

アドレスオフセット：0x00 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]							
rw	rw							rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
rw	rw	rw	rw				rw	rw	rw	rw		rw	rw	rw	rw

ビット 31 **DFSDMEN**：DFSDM インタフェースのグローバル有効ビット

0：DFSDM インタフェース無効

1：DFSDM インタフェース有効

DFSDM インタフェースが有効である場合、有効化された y チャネルおよび x フィルタ設定（DFSDM_CHyCFGR1 の CHEN ビットおよび DFSDM_FLTxCR1 の DFEN ビット）に応じて動作が開始します。DFSDMEN=0 をセットすることでクリアされるデータ：

–すべてのレジスタ DFSDM_FLTxISR（x = 0～3）がリセット状態にセットされます。

–すべてのレジスタ DFSDM_FLTxAWSR（x = 0～3）がリセット状態にセットされます。

注： DFSDMEN は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 30 **CKOUTSRC**：出力シリアルクロックソースの選択

0：システムクロックからの出力クロックソース

1：オーディオクロックからの出力クロックソース

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

注： CKOUTSRC は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CKOUTDIV[7:0]**出力シリアルクロック分周回路

0：出力クロック生成は無効です（CKOUT 信号はロー状態にセットされます）。

1～255 : 2～256 の範囲で CKOUT 信号のシリアルクロック出力のシステムクロックの分周を定義します（分周回路 = CKOUTDIV+1）。

CKOUTDIV はクロック無検出の閾値も定義します。

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合、CKOUT 信号はロー状態にセットされます（設定は DFSDMEN=0 の後の 1 DFSDM クロックサイクルで実行されます）。

注： CKOUTDIV は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 15:14 **DATPACK[1:0]**DFSDM_CHyDATINR レジスタのデータパッキングモード

0 :標準：DFSDM_CHyDATINR レジスタの入力データは INDAT0[15:0] にのみ保存されます。
DFSDM_CHyDATINR レジスタを空にするには、1 つのサンプルをチャネル y から DFSDM フィルタで読み出す必要があります。

1：インタリーブ：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

–INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

–INDAT1[15:0] の 2 つ目のサンプル（チャネル y に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、2 つのサンプルをチャネル y からデジタルフィルタで読み出す必要があります（INDAT0[15:0] パートは最初のサンプルとして、INDAT1[15:0] パートは次のサンプルとして読み出されます）。

2：デュアル：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

–INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

–INDAT1[15:0] の 2 つ目のサンプル（チャネル y+1 に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、最初のサンプルをチャネル y からデジタルフィルタで、2 つ目のサンプルをチャネル y+1 から別のデジタルフィルタで読み出す必要があります。デュアルモードは、偶数チャネル数（y = 0、2、4、6）で使用でき、奇数チャネル数（y = 1、3、5、7）の場合、DFSDM_CHyDATINR は書き込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

3：予約済み

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 13:12 **DATMPX[1:0]**チャネル y の入力データマルチプレクス

0 :チャネル y へのデータは外部シリアル入力から 1 ビット値で取得されます。DFSDM_CHyDATINR レジスタは書き込み保護されています。

1 : チャネル y へのデータは内部アナログからデジタルコンバータ ADC $_{y+1}$ 出力レジスタの更新まで 16 ビット値で取得されます（ADC_{y+1} が使用可能な場合）。ADC からのデータが、DFSDM_CHyDATINR レジスタの INDAT0[15:0] 部分に書き込まれます。

2 :チャネル y へのデータは内部 DFSDM_CHyDATINR レジスタから直接 CPU/DMA 書き込みで取得されます。DATPACK[1:0] ビットフィールド設定に応じて 1 つまたは 2 つの 16 ビットデータサンプルを書き込むことができます。

3 :予約済み

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

注： ADC1 および ADC2 によってのみ DATMPX[1:0] = 1 がサポートされます。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CHINSEL**：チャネル入力の選択

0：チャネル入力が同じチャネル y のピンから取得されます。

1：チャネル入力が次のチャネル（チャネル（y+1）モジュロ 8）のピンから取得されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 7 **CHEN**：チャネル y 有効化

0：チャネル y 無効

1：チャネル y 有効

チャネル y が有効である場合、シリアルデータの受信は特定のチャネル設定に応じて開始されます。

ビット 6 **CKABEN**：チャネル y のクロック無検出回路有効化

0：チャネル y のクロック無検出回路無効

1：チャネル y のクロック無検出回路有効

ビット 5 **SCDEN**：チャネル y の短絡検出回路有効化

0：入力チャネル y が、短絡検出回路によって保護されません。

1：入力チャネル y が、短絡検出回路によって継続的に保護されます。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **SPICKSEL[1:0]**：チャネル y の SPI クロック選択

0 :外部 CKINy 入力からのクロック - SITP[1:0] に応じたサンプリングポイント

1 :内部 CKOUT 出力からのクロック - SITP[1:0] に応じたサンプリングポイント

2 :内部 CKOUT からのクロック - 毎秒の CKOUT 立ち下がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が
各クロック入力の立ち上がりエッジで有効である場合）

3 :内部 CKOUT 出力からのクロック - 毎秒の CKOUT 立ち上がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 1:0 **SITP[1:0]**：チャネル y のシリアルインタフェースのタイプ

00：データをストローブする立ち上がりエッジのある SPI

01：データをストローブする立ち下がりエッジのある SPI

10:DATINy ピンでのマンチェスタ符号化入力:立ち上がりエッジ = ロジック 0、立ち下がりエッジ = ロジック 1

11:DATINy ピンでのマンチェスタ符号化入力:立ち上がりエッジ = ロジック 1、立ち下がりエッジ = ロジック 0

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

30.7.2 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR2）（y=0～7）

このレジスタは、チャネル y（y = 0～7）によって使用されるパラメータを指定します。

アドレスオフセット：0x04 + 0x20 * y

リセット値：0x0000 0000

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

ビット 31:8 **OFFSET[23:0]**：チャネル y の 24 ビット 較正オフセット

チャネル y では、OFFSET がチャネルからの各変換結果に適用されます。

この値はソフトウェアでセットされます。

ビット 7:3 **DTRBS[4:0]**：チャネル y のデータ右ビットシフト

0～31：積分器からのデータ結果のシフト（最終結果を得るために右に何ビットシフトするか）を定義します。ビットシフトは、オフセット較正前に実行されます。データシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、（結果データの有効な 24 ビット符号フォーマットを持つために）維持されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

30.7.3 DFSDM チャネルアナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）（y=0～7）

チャネル y（y = 0～7）の短絡検出回路およびアナログウォッチドッグ設定

アドレスオフセット：0x08 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]				
								rw	rw		rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **AWFORD[1:0]**：チャネル y のアナログウォッチドッグ Sinc フィルタ段数

0：FastSinc フィルタタイプ

1：$Sinc^1$ フィルタタイプ

2：$Sinc^2$ フィルタタイプ

3：$Sinc^3$ フィルタタイプ

$Sinc^x$ フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **AWFOSR[4:0]**：チャネル y のアナログウォッチドッグフィルタのオーバーサンプリング比（デシメーション率）

0～31：Sinc タイプフィルタの長さを 1～32 の範囲で定義します（AWFOSR + 1）。この数字は、アナログデータレートのデシメーション率でもあります。

このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

注： AWFOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:12 **BKSCD[3:0]**：チャネル y の短絡検出回路のブレーク信号割り当て

BKSCD[i] = 0：チャネル y の短絡検出回路に割り当てられていないブレーク i 信号

BKSCD[i] = 1：チャネル y の短絡検出回路に割り当てられたブレーク i 信号

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SCDT[7:0]**：チャネル y の短絡検出回路の閾値

これらのビットは、短絡検出回路の閾値カウンタを定義するために、ソフトウェアによって書き込まれます。この値に達すると、特定のチャネルで短絡検出回路イベントが発生します。

30.7.4 DFSDM チャネルウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）（y=0～7）

このレジスタは、入力チャネル y（y = 0～7）に関連付けされたアナログウォッチドッグフィルタからのデータを含みます。

アドレスオフセット：0x0C + 0x20 * y

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WDATA[15:0]**：入力チャネル y ウォッチドッグデータ

入力チャネル y のアナログウォッチドッグフィルタによって変換されたデータです。このデータは、このチャネルに対して制限付き分解能（OSR=1～32/Sinc 段数 = 1～3）で連続して変換されます（トリガなし）。

30.7.5 DFSDM チャネルデータ入力レジスタ（DFSDM_CHyDATINR）（y=0～7）

このレジスタには、16 ビットの入力データが含まれており、DFSDM フィルタモジュールによって処理されます。

アドレスオフセット：0x10 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INDAT1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INDAT0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **INDAT0[15:0]**：チャネル y またはチャネル y+1 の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

INDAT0[15:0] は書き込み保護されています（入力サンプルには使用しません）。

DATPACK[1:0]=1（インタリーブモード）の場合

2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：INDAT1[15:0] のサンプルは、チャネル（y+1）の INDAT0[15:0] に自動的にコピーされます。

奇数 y チャネルの場合：INDAT1[15:0] は書き込み保護されています。

詳細は、*セクション 30.4.6：並列データ入力* を参照してください。

INDAT0[15:1] は 16 ビット符号フォーマットです。

ビット 15:0 **INDAT0[15:0]**：チャネル y の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

チャネル y データサンプルは、INDAT0[15:0] に保存されます。

DATPACK[1:0]=1（インタリーブモード）の場合

最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：チャネル y データサンプルは、INDAT0[15:0] に保存されます。

奇数 y チャネルの場合：INDAT0[15:0] は書き込み保護されています。

詳細は、*セクション 30.4.6：並列データ入力* を参照してください。

INDAT0[15:0] は 16 ビット符号フォーマットです。

30.8 DFSDM フィルタ x モジュールレジスタ（x=0～3）

30.8.1 DFSDM 制御レジスタ 1（DFSDM_FLTxCR1）

アドレスオフセット：0x100 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSWSTART	Res.
	rw	rw			rw	rw	rw			rw		rw	rw	r0w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSWSTART	DFEN
	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw		r0w	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **AWFSEL**：アナログウォッチドッグ高速モード選択

0：データ出力値のアナログウォッチドッグ（デジタルフィルタ後）。オフセット補正およびシフト後に比較が実行されます

1：チャネルトランシーバ値のアナログウォッチドッグ（ウォッチドッグフィルタ後）

ビット 29 **FAST**：レギュラ変換の高速変換モード選択

0：高速変換モード無効

1：高速変換モード有効

連続モードでレギュラ変換を行うとき、高速モードを有効にすると、各変換（最初を除く）を標準モードより速く実行できます。このビットは、連続していない変換では効果がありません。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^X$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

$F_{OSR} = F_{OSR}[9:0]+1 = 1$（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレートです。

ビット 28:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **RCH[2:0]**：レギュラチャネル選択

0：チャネル 0 がレギュラチャネルとして選択されます。

1：チャネル 1 がレギュラチャネルとして選択されます。

...

7：チャネル 7 がレギュラチャネルとして選択されます。

RCIP=1 のときにこれらのビットに書き込むと、次のレギュラ変換開始時に有効になります。これは、連続モードで特に役立ちます（RCONT=1 の場合）。保留中のレギュラ変換にも影響します（進行中のインジェクト変換のため）。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RDMAEN**：DMA チャネルのレギュラ変換のデータ読み出し有効化

0：DMA チャネルがレギュラデータの読み出し有効になりません。

1：DMA チャネルがレギュラデータの読み出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **RSYNC**：DFSDM_FLT0 と同期したレギュラ変換開始

0：DFSDM_FLT0 と同期してレギュラ変換を開始しません。

1：DFSDM_FLT0 でレギュラ変換が開始された時点で、この DFSDM_FLTx でレギュラ変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 18 **RCONT**：レギュラ変換の連続モード選択

0：レギュラチャネルが、各変換リクエストにつき 1 回だけ変換されます。

1：レギュラチャネルが、各変換リクエストの後に繰り返し変換されます。

連続レギュラ変換がすでに進行中の場合にこのビットに“0”を書き込むと、即座に連続モードを停止します。

ビット 17 **RSWSTART**：ソフトウェアのレギュラチャネルの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、レギュラ変換開始のリクエストをして、RCIP が“1”になります。すでに RCIP=1 の場合、RSWSTART に書き込んでも変化はありません。RSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 16:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **JEXTEN[1:0]**：インジェクト変換のトリガ有効化およびトリガエッジ選択

00：トリガ検出は無効です。

01：選択されたトリガの各立ち上がりエッジがインジェクト変換の開始をリクエストします。

10：選択されたトリガの各立ち下がりエッジがインジェクト変換の開始をリクエストします。

11：選択されたトリガの立ち上がりエッジおよび立ち下がりエッジ両方がインジェクト変換の開始をリクエストします。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 12:8 **JEXTSEL[4:0]**：インジェクト変換開始のトリガ信号選択

0x0～0x1F：次の表で選択されるトリガ入力（内部トリガまたは外部トリガ）

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： ***同期トリガは最大 1 $f_{DFSDMCLK}$ クロックサイクルの遅延（決定的ジッタ）があり、非同期トリガは 2～3 $f_{DFSDMCLK}$ クロックサイクルの遅延（最大 1 サイクルまでのジッタ）があります。***

DFSDM_FLT0		DFSDM_FLT1	DFSDM_FLT2	DFSDM_FLT3
0x00	dfsdm_jtrg0	dfsdm_jtrg0	dfsdm_jtrg0	dfsdm_jtrg0
0x01	dfsdm_jtrg1	dfsdm_jtrg1	dfsdm_jtrg1	dfsdm_jtrg1
...				
0x1E	dfsdm_jtrg30	dfsdm_jtrg30	dfsdm_jtrg30	dfsdm_jtrg30
0x1F	dfsdm_jtrg31	dfsdm_jtrg31	dfsdm_jtrg31	dfsdm_jtrg31

*表 231：DFSDM トリガ接続*を参照してください。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JDMAEN**：DMA チャネルのインジェクトチャネルグループのデータ読み出し有効化

0：DMA チャネルがインジェクトデータの読み出し有効になりません。

1：DMA チャネルがインジェクトデータの読み出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 4 **JSCAN**：インジェクト変換のスキャン変換モード

0：1 つのチャネル変換がチャネルグループから実行されてから、このグループで選択されたチャネルが選択されます。

1：インジェクトグループチャネルの一連の変換が実行され、選択された最下位のチャネルから開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

ビット 3 **JSYNC**：DFSDM_FLT0 JSWSTART トリガと同期したインジェクト変換開始

0：DFSDM_FLT0 と同期してインジェクト変換を開始しません。

1：JSWSTART トリガによって DFSDM_FLT0 でインジェクト変換が開始された時点で、この DFSDM_FLTx でインジェクト変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **JSWSTART**：インジェクトチャネルグループの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、インジェクト変換グループでのチャネル変換のリクエストをして、同時に JCIP が“1”になります。すでに JCIP=1 の場合、JSWSTART に書き込んでも変化はありません。JSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 0 **DFEN**：DFSDM_FLTx 有効化

0：DFSDM_FLTx は無効です。特定の DFSDM_FLTx のすべての変換が即座に停止し、すべての DFSDM_FLTx 機能が停止します。

1：DFSDM_FLTx は有効です。DFSDM_FLTx が有効である場合、DFSDM_FLTx は設定に応じて動作を開始します。

DFEN=0 をセットすることでクリアされるデータ：

–レジスタ DFSDM_FLTxISR は、リセット状態にセットされます。

–レジスタ DFSDM_FLTxAWSR は、リセット状態にセットされます。

30.8.2 DFSDM 制御レジスタ 2（DFSDM_FLTxCR2）

アドレスオフセット：0x104 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **AWDCH[7:0]**：アナログウォッチドッグチャネル選択

これらのビットは、アナログウォッチドッグによって継続的に保護される入力チャネルを選択します。

AWDCH[y] = 0：チャネル y のアナログウォッチドッグは無効です。

AWDCH[y] = 1：チャネル y のアナログウォッチドッグは有効です。

ビット 15:8 **EXCH[7:0]**：極値検出回路チャネル選択

これらのビットは、極値検出回路によって取得される入力チャネルを選択します。

EXCH[y] = 0：極値検出回路はチャネル y からデータを受け取りません。

EXCH[y] = 1：極値検出回路はチャネル y からデータを受け取ります。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CKABIE**：クロック無割り込み有効化

0：チャネル入力クロック無割り込みの検出を無効にします。

1：チャネル入力クロック無割り込みの検出を有効にします。

CKABF[7:0]（DFSDM_FLTxISR）の説明を参照してください。

注： CKABIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 5 **SCDIE**：短絡検出回路割り込み有効化
0：短絡検出回路割り込み無効
1：短絡検出回路割り込み有効
SCDF[7:0]（DFSDM_FLTxISR）の説明を参照してください。
注： SCDIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 4 **AWDIE**：アナログウォッチドッグ割り込みイネーブル
0：アナログウォッチドッグ割り込みは無効です。
1：アナログウォッチドッグ割り込みは有効です。
AWDF（DFSDM_FLTxISR）の説明を参照してください。

ビット 3 **ROVRIE**：レギュラデータオーバーラン割り込み有効化
0：レギュラデータオーバーラン割り込み無効
1：レギュラデータオーバーラン割り込み有効
ROVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 2 **JOVRIE**：インジェクトデータオーバーラン割り込み有効化
0：インジェクトデータオーバーラン割り込み無効
1：インジェクトデータオーバーラン割り込み有効
JOVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 1 **REOCIE**：レギュラ変換終了割り込み有効化
0：レギュラ変換終了割り込み無効
1：レギュラ変換終了割り込み有効
REOCF（DFSDM_FLTxISR）の説明を参照してください。

ビット 0 **JEOCIE**：インジェクト変換終了割り込み有効化
0：インジェクト変換終了割り込み無効
1：インジェクト変換終了割り込み有効
JEOCF（DFSDM_FLTxISR）の説明を参照してください。

30.8.3 DFSDM 割り込みおよびステータスレジスタ（DFSDM_FLTxISR）

アドレスオフセット：0x108 + 0x80 * x、x = 0～3

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCDF[7:0]								CKABF[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	r	r									r	r	r	r	r

ビット 31:24 **SCDF[7:0]**：短絡検出回路フラグ

SDCF[y]=0：チャネル y の短絡検出回路イベントが発生していません。

SDCF[y]=1：チャネル y で短絡検出回路カウンタが DFSDM_CHyAWSCDR レジスタにプログラムされた値に達しています。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの対応する CLRSCDF[y] ビットで、ソフトウェアによってクリアできます。SCDF[y] は、CHEN[y] = 0（特定のチャネルが無効）の場合、ハードウェアでも

クリアされます。

注： SCDF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 23:16 **CKABF[7:0]**：クロック無フラグ

CKABF[y]=0：チャネル y のクロック信号があります。

CKABF[y]=1：チャネル y のクロック信号がありません。

特定の y ビットは、クロック無がチャネル y で検出された際にハードウェアによってセットされます。CHEN=0（DFSDM_CHyCFGR1 レジスタを参照）の場合、ハードウェアによって CKABF[y]=1 状態で保持されます。トランシーバがまだ同期されていない場合、CKABF[y]=1 状態で保持されます。DFSDM_FLTxICR レジスタの対応する CLRCKABF[y] ビットで、ソフトウェアによってクリアできます。

注： CKABF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **RCIP**：進行中のレギュラ変換ステータス

0：発行されたレギュラチャネルの変換のリクエストはありません。

1：レギュラチャネルの変換が進行中またはレギュラ変換のリクエストが保留中です。

RCIP=1 の場合、レギュラ変換開始のリクエストは無視されます。

ビット 13 **JCIP**：進行中のインジェクト変換ステータス

0：（ソフトウェアあるいはトリガによって）発行されたインジェクトチャネルグループの変換のリクエストはありません。

1：JSWSTART に“1”が書き込まれているか、トリガを検出したため、インジェクトチャネルグループの変換が進行中またはインジェクト変換のリクエストが保留中です。

JCIP=1 の場合、インジェクト変換開始のリクエストは無視されます。

ビット 12:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **AWDF**：アナログウォッチドッグ

0：アナログウォッチドッグイベントは発生していません。

1：アナログウォッチドッグブロックが、DFSDM_FLTxAWLTR レジスタまたは DFSDM_FLTxAWHTR レジスタでプログラムされた値に跨る電圧を検出しました。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxAWSR レジスタ のすべてのソースフラグビット AWHTF[7:0] および AWLTF[7:0] をクリアする（DFSDM_FLTxAWCFR レジスタのクリアビットに“1”を書き込む）ことで、ソフトウェアによってクリアされます。

ビット 3 **ROVRF**：レギュラ変換オーバーランフラグ

0：レギュラ変換オーバーランは発生していません。

1：レギュラ変換オーバーランが発生し、REOCF が“1”のときにレギュラ変換が終了したことを示します。RDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRROVRF ビットで、ソフトウェアによってクリアできます。

ビット 2 **JOVRF**：インジェクト変換オーバーランフラグ

0：インジェクト変換オーバーランは発生していません。

1：インジェクト変換オーバーランが発生し、JEOCF が“1”のときにインジェクト変換が終了したことを示します。JDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRJOVRF ビットで、ソフトウェアによってクリアできます。

ビット 1 **REOCF**：レギュラ変換終了フラグ

0：レギュラ変換が完了していません。

1：レギュラ変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxRDATAR を読み出す際にクリアされます。

ビット 0 **JEOCF**：インジェクト変換終了フラグ

0：インジェクト変換が完了していません。

1：インジェクト変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxJDATAR を読み出す際にクリアされます。

注： ***各フラグビットで、DFSDM_FLTxCR2 の対応するビットをセットすることによって、割り込みを有効にできます。割り込みが呼び出された場合、割り込みサービスルーチンが終了する前にフラグがクリアされる必要があります。***

DFEN=0 の場合、DFSDM_FLTxISR のすべてのビットは自動的にリセットされます。

30.8.4 DFSDM 割り込みフラグクリアレジスタ（DFSDM_FLTxICR）

アドレスオフセット：0x10C + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRSCDF[7:0]								CLRCKABF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRR OVRF	CLR J OVRF	Res.	Res.
												rc_w1	rc_w1		

ビット 31:24 **CLRSCDF[7:0]**：短絡検出回路フラグのクリア

CLRSCDF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRSCDF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する SCDF[y] ビットをクリアします。

注： ***CLRSCDF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。***

ビット 23:16 **CLRCKABF[7:0]**：クロック無フラグのクリア

CLRCKABF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRCKABF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する CKABF[y] ビットをクリアします。トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビットでクリアできません。

注： ***CLRCKABF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。***

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CLRROVRF**：レギュラ変換オーバーランフラグのクリア
0：“0”を書き込んでも、ビットの値は変化しません。
1：“1”を書き込むと DFSDM_FLTxISR レジスタの ROVRF ビットをクリアします。

ビット 2 **CLRJOVRF**：インジェクト変換オーバーランフラグのクリア
0：“0”を書き込んでも、ビットの値は変化しません。
1：“1”を書き込むと DFSDM_FLTxISR レジスタの JOVRF ビットをクリアします。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

注： ***DFSDM_FLTxICR のビットは常に 0 として読み出されます。***

30.8.5 DFSDM インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR）

アドレスオフセット：0x110 + 0x80 * x、x = 0～3

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **JCHG[7:0]**：インジェクトチャネルグループ選択
JCHG[y]=0：チャネル y はインジェクトグループの一部ではありません。
JCHG[y]=1：チャネル y はインジェクトグループの一部です。
JSCAN=1 の場合、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、選択された最上位のチャネルでシーケンスが終了します。
JSCAN=0 の場合、選択されたチャネルから 1 つだけを変換し、チャネル選択が次のチャネルに移ります。
JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。
最低 1 チャネルは、必ずインジェクトグループに選択されている必要があります。すべての JCHG ビットを 0 にする書き込みは無視されます。

30.8.6 DFSDM フィルタ制御レジスタ（DFSDM_FLTxFCR）

アドレスオフセット：0x114 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]									
rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **FORD[2:0]**：Sinc フィルタ段数

0：FastSinc フィルタタイプ

1：$Sinc^1$ フィルタタイプ

2：$Sinc^2$ フィルタタイプ

3：$Sinc^3$ フィルタタイプ

4：$Sinc^4$ フィルタタイプ

5：$Sinc^5$ フィルタタイプ

6～7：予約済み

$Sinc^x$ フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}} \right)^x$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}} \right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **FOSR[9:0]**：Sinc フィルタのオーバーサンプリング比（デシメーション率）

0～1023：Sinc タイプフィルタの長さを 1～1024 の範囲で定義します（F_{OSR} = FOSR[9:0] +1）。この数字は、フィルタからの出力データレートのデシメーション率でもあります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： FOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **IOSR[7:0]**：積分器のオーバーサンプリング比（平均長）

0～255：1～256 の積分器の長さ（IOSR + 1）積分器からの 1 つの出力データサンプルに合計する Sinc フィルタからのサンプル数を定義します。積分器からの出力データレートは、この数字（追加のデータデシメーション率）によって減少します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： IOSR = 0 の場合、積分器に効果はありません（積分器バイパス）。

30.8.7 インジェクトグループの DFSDM データレジスタ（DFSDM_FLTxJDATAR）

アドレスオフセット：0x118 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JDATA[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JDATA[7:0]								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
r	r	r	r	r	r	r	r						r	r	r

ビット 31:8 **JDATA[23:0]**：インジェクトグループ変換データ

インジェクトグループチャネルの各変換が終了すると、結果データがこのフィールドに保存されます。データは、JEOCF=1 のときに有効です。このレジスタを読み出すと、対応する JEOCF がクリアされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **JDATACH[2:0]**：最近変換したインジェクトチャネル

インジェクトグループチャネルの各変換が終了すると、どのチャネルを変換したかを示すために JDATACH[2:0] が更新されます。そのため、JDATA[23:0] は JDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

注： ***DMA を使用して、このレジスタからデータを読み出せます。ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。***

このレジスタを読み出すと、DFSDM_FLTxISR の JEOCF がクリアされます。そのため、DMA が有効でこのレジスタからデータを読み出している場合、ファームウェアがこのレジスタを読み出さないようにしてください。

30.8.8 レギュラチャネルの DFSDM データレジスタ（DFSDM_FLTxRDATAR）

アドレスオフセット：0x11C + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA[23:8]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[7:0]								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
r	r	r	r	r	r	r	r				r		r	r	r

ビット 31:8 **RDATA[23:0]**：レギュラチャネル変換データ

各レギュラ変換が終了すると、データがこのレジスタに保存されます。データは、REOCF=1 のときに有効です。このレジスタを読み出すと、対応する REOCF がクリアされます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RPEND**：レギュラチャネル保留データ

RDATA[23:0] のレギュラデータが、変換中のインジェクトチャネルトリガによって遅延しました。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **RDATACH[2:0]**：最近変換したレギュラチャネル

各レギュラチャネルが終了すると、どのチャネルを変換したかを示すために RDATACH[2:0] が更新されます（DFSDM_FLTxCR1 レジスタのレギュラチャネル選択 RCH[2:0] がレギュラ変換中に更新できるため）。そのため、RDATA[23:0] は RDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

注： ***ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。***

このレジスタを読み出すと、DFSDM_FLTxISR の REOCF がクリアされます。

30.8.9 DFSDM アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）

アドレスオフセット：0x120 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:8 **AWHT[23:0]**：アナログウォッチドッグ高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。

注： ***チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため）上位の 16 ビット（AWHT[23:8]）が、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWHT[7:0] ビットは、この場合の比較では取得されません。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWH[3:0]**：アナログウォッチドッグ高閾値イベントへのブレーク信号割り当て

BKAWH[i] = 0：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられません。

BKAWH[i] = 1：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられます。

30.8.10 DFSDM アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）

アドレスオフセット：0x124 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:8 **AWLT[23:0]**：アナログウォッチドッグ低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。

注： ***チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため）上位の 16 ビット（AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWLT[7:0] ビットは、この場合の比較では取得されません。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWL[3:0]**：アナログウォッチドッグ低閾値イベントへのブレーク信号割り当て

BKAWL[i] = 0：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられません。

BKAWL[i] = 1：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられます。

30.8.11 DFSDM アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）

アドレスオフセット：0x128 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **AWHTF[7:0]**：アナログウォッチドッグ高閾値フラグ

AWHTF[y]=1 は、チャネル y での高閾値エラーを示します。ハードウェアによってセットされます。

DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[y] ビットで、ソフトウェアによってクリアできます。

ビット 7:0 **AWLTF[7:0]**：アナログウォッチドッグ低閾値フラグ

AWLTF[y]=1 は、チャネル y での低閾値エラーを示します。ハードウェアによってセットされます。

DFSDM_FLTxAWCFR レジスタの対応する CLRAWLTF[y] ビットで、ソフトウェアによってクリアできます。

注： ***DFEN=0 の場合、DFSDM_FLTxAWSR のすべてのビットは自動的にリセットされます。***

30.8.12 DFSDM アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）

アドレスオフセット：0x12C + 0x80 * x、x = 0〜3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLRAWHTF[7:0]								CLRAWLTF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **CLRAWHTF[7:0]**：アナログウォッチドッグ高閾値フラグのクリア

CLRAWHTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWHTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWHTF[y] ビットをクリアします。

ビット 7:0 **CLRAWLTF[7:0]**：アナログウォッチドッグ低閾値フラグのクリア

CLRAWLTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWLTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWLTF[y] ビットをクリアします。

30.8.13 DFSDM 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）

アドレスオフセット：0x130 + 0x80 * x、x = 0〜3

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMAX[23:8]															
r1	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMAX[7:0]								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
r0	r0	r0	r0	r0	r0	r0	r0						r	r	r

ビット 31:8 **EXMAX[23:0]**：極値検出回路最大値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最高値を示します。EXMAX[23:0] ビットは、このレジスタの読み出しによって値（0x800000）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMAXCH[2:0]**：極値検出回路最大データチャネル

これらのビットは、データが EXMAX[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読み出しによってクリアされます。

30.8.14 DFSDM 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）

アドレスオフセット：0x134 + 0x80 * x、x = 0～3

リセット値：0x7FFF FF00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMIN[23:8]															
r0	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMIN[7:0]								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
r1	r1	r1	r1	r1	r1	r1	r1						r	r	r

ビット 31:8 **EXMIN[23:0]**：極値検出回路最小値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最低値を示します。EXMIN[23:0] ビットは、このレジスタの読み出しによって値（0x7FFFFF）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMINCH[2:0]**：極値検出回路最小データチャネル

これらのビットは、データが EXMIN[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読み出しによってクリアされます。

30.8.15 DFSDM 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）

アドレスオフセット：0x138 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:4 **CNVCNT[27:0]：**28 ビットタイマカウント変換時間 t = CNVCNT[27:0] / $f_{DFSDMCLK}$

タイマには、DFSDM クロック（システムクロック $f_{DFSDMCLK}$）からの入力クロックがあります。変換時間測定は、各変換の開始時に開始し、変換終了時に停止します（最初から最後のシリアルサンプルまでの間隔）。フィルタバイパス（FOSR[9:0] = 0）の場合のみ、変換時間測定が停止し、CNVCNT[27:0] = 0.となります。カウント時間を次に示します。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^x$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

CNVCNT = 0（カウント停止、変換時間：$t = I_{OSR} / f_{CKIN}$）

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレート（内部 ADC または CPU/DMA 書き込みからの）です。

注：　変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、タイマはこの割り込み時間もカウントします。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

30.8.16 DFSDM レジスタマップ

次の表に DFSDM レジスタの一覧を示します。

表 236. DFSDM レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**DFSDM_CH0CFGR1**	DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]								DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値	0	0							0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0		0	0	0	0
0x04	**DFSDM_CH0CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0																								0							
0x08	**DFSDM_CH0AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0x0C	**DFSDM_CH0WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**DFSDM_CH0DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14 - 0x1C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	**DFSDM_CH1CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**DFSDM_CH1CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x28	**DFSDM_CH1AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x2C	**DFSDM_CH1WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	**DFSDM_CH1DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 - 0x3C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x40	**DFSDM_CH2CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x44	**DFSDM_CH2CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x48	**DFSDM_CH2AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x4C	**DFSDM_CH2WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**DFSDM_CH2DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54 - 0x5C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x60	**DFSDM_CH3CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x64	**DFSDM_CH3CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x68	**DFSDM_CH3AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x6C	**DFSDM_CH3WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**DFSDM_CH3DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	**DFSDM_CH4CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x84	**DFSDM_CH4CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x88	**DFSDM_CH4AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x8C	**DFSDM_CH4WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	**DFSDM_CH4DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94 - 0x9C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA0	**DFSDM_CH5CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xA4	**DFSDM_CH5CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xA8	**DFSDM_CH5AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xAC	**DFSDM_CH5WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB0	**DFSDM_CH5DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4 - 0xBC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xC0	**DFSDM_CH6CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xC4	**DFSDM_CH6CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xC8	**DFSDM_CH6AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xCC	**DFSDM_CH6WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD0	**DFSDM_CH6DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD4 - 0xDC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xE0	**DFSDM_CH7CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xE4	**DFSDM_CH7CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xE8	**DFSDM_CH7AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xEC	**DFSDM_CH7WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xF0	**DFSDM_CH7DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xF4 - 0xFC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	**DFSDM_FLT0CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x104	**DFSDM_FLT0CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x108	**DFSDM_FLT0ISR**	SCDF[7:0]								CKABF[7:0]								Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1		0	0									0	0	0	0	0
0x10C	**DFSDM_FLT0ICR**	CLRSCDF[7:0]								CLRCKABF[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0													0	0		
0x110	**DFSDM_FLT0JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x114	**DFSDM_FLT0FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x118	**DFSDM_FLT0JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH [2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x11C	**DFSDM_FLT0RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x120	**DFSDM_FLT0AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x124	**DFSDM_FLT0AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x128	**DFSDM_FLT0AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x12C	**DFSDM_FLT0AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x130	**DFSDM_FLT0EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x134	**DFSDM_FLT0EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x138	**DFSDM_FLT0CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x13C ～ 0x17C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x180	**DFSDM_FLT1CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0		0	0	0	0			0	0	0		0	0
0x184	**DFSDM_FLT1CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x188	**DFSDM_FLT1ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x18C	**DFSDM_FLT1ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x190	**DFSDM_FLT1JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x194	**DFSDM_FLT1FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x198	**DFSDM_FLT1JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x19C	**DFSDM_FLT1RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x1A0	**DFSDM_FLT1AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A4	**DFSDM_FLT1AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A8	**DFSDM_FLT1AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1AC	**DFSDM_FLT1AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1B0	**DFSDM_FLT1EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x1B4	**DFSDM_FLT1EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x1B8	**DFSDM_FLT1CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x1BC - 0x1FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x200	**DFSDM_FLT2CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x204	**DFSDM_FLT2CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	-	-	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		-	-	0	0	0	0	0
0x208	**DFSDM_FLT2ISR**	-								-								Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値	-								-									0	0									0	0	0	0	0
0x20C	**DFSDM_FLT2ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x210	**DFSDM_FLT2JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x214	**DFSDM_FLT2FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x218	**DFSDM_FLT2JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x21C	**DFSDM_FLT2RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x220	**DFSDM_FLT2AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x224	**DFSDM_FLT2AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x228	**DFSDM_FLT2AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x22C	**DFSDM_FLT2AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x230	**DFSDM_FLT2EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x234	**DFSDM_FLT2EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x238	**DFSDM_FLT2CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				

参考資料

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x23C ～ 0x27C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x280	**DFSDM_FLT3CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x284	**DFSDM_FLT3CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x288	**DFSDM_FLT3ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x28C	**DFSDM_FLT3ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x290	**DFSDM_FLT3JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x294	**DFSDM_FLT3FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x298	**DFSDM_FLT3JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x29C	**DFSDM_FLT3RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x2A0	**DFSDM_FLT3AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A4	**DFSDM_FLT3AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A8	**DFSDM_FLT3AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 236. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2AC	**DFSDM_FLT3AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2B0	**DFSDM_FLT3EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x2B4	**DFSDM_FLT3EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x2B8	**DFSDM_FLT3CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x2BC - 0x3FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス* を参照してください。

参考資料

31 デジタルカメラインタフェース（DCMI）

31.1 DCMI の概要

このデジタルカメラインタフェースは、外部 8、10、12、または14 ビット CMOS カメラモジュールからハイスピードデータフローを受信できる同期型パラレルインタフェースです。YCbCr4:2:2/RGB565 プログレッシブビデオおよび圧縮データ（JPEG）など、さまざまなデータフォーマットをサポートします。

このインタフェースはモノクロカメラ、X24 および X5 カメラで使用し、サイズ変更などすべての前処理はカメラモジュール内で行われることを前提としています。

31.2 DCMI の主な機能

- 8、10、12、または 14 ビットのパラレルインタフェース
- 埋め込み／外部ラインおよびフレーム同期
- 連続モードまたはスナップショットモード
- クロップ機能
- 以下のデータフォーマットに対応：
  - 8/10/12/14 ビットのプログレッシブビデオ：モノクロまたは Raw ベイヤー
  - YCbCr 4:2:2 プログレッシブビデオ
  - RGB 565 プログレッシブビデオ
  - 圧縮データ：JPEG

31.3 DCMI クロック

デジタルカメラインタフェースでは、DCMI_PIXCLK および HCLK の 2 つのクロックドメインを使用します。DCMI_PIXCLK と HCLK が安定した後、DCMI_PIXCLK で生成された信号は、HCLK の立ち上がりエッジでサンプリングされます。カメラからのデータが安定し、サンプリング可能状態であることを示すため、イネーブル信号が HCLK ドメインで生成されます。最大 DCMI_PIXCLK 周期は、HCLK 2.5 周期分よりも長い必要があります。

31.4 DCMI 機能概要

デジタルカメラインタフェースは、高速のデータフローを受信できる同期型パラレルインタフェースです。最大 14 本のデータライン（D13-D0）と 1 本のピクセルクロックライン（DCMI_PIXCLK）で構成されています。ピクセルクロックの極性はプログラム可能であるため、データはピクセルクロックの立ち上がりエッジまたは立ち下がりエッジのどちらでもキャプチャすることができます。

データは 32 ビットのデータレジスタ（DCMI_DR）に格納され、汎用 DMA チャネルを介して転送されます。画像バッファはカメラインタフェースではなく DMA が管理します。

カメラから受信したデータは、ライン/フレーム（Raw YUV/RGB/Bayer モード）で構成したり、一連の JPEG 画像にすることができます。JPEG 画像を受信できるようにするには、JPEG ビット（DCMI_CR レジスタのビット 3）をセットする必要があります。

データフローは、オプションの DCMI_HSYNC（水平同期）信号および DCMI_VSYNC（垂直同期）信号を用いてハードウェアで同期をとることもできれば、データフローに埋め込まれてた同期コードで同期をとることもできます。

31.4.1 DCMI のブロック図

図 224 に DCMI ブロック図を示します。

図 224. DCMI のブロック図

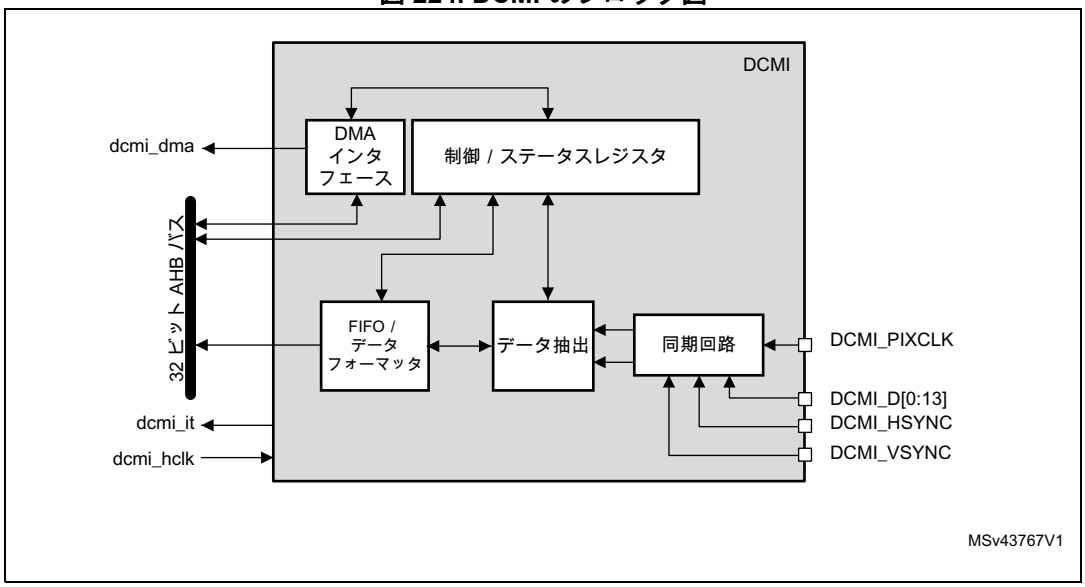
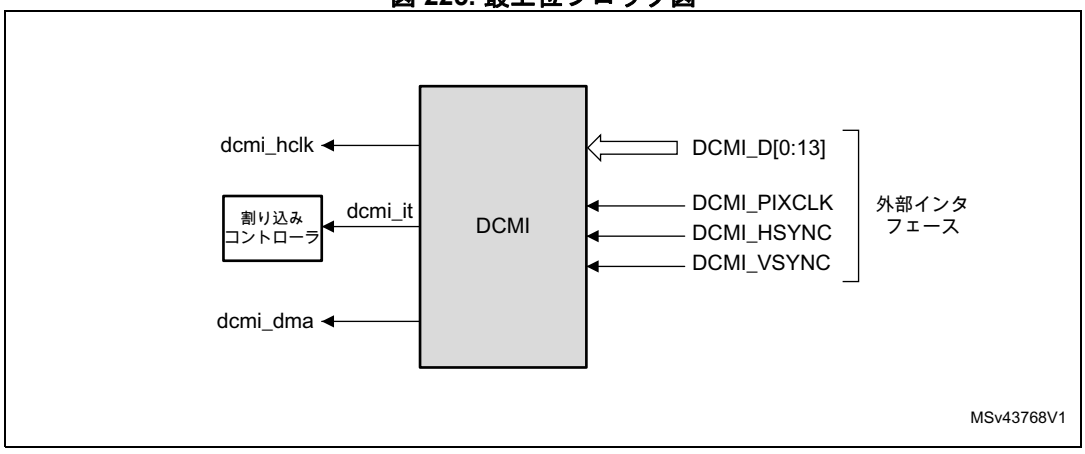


図 225. 最上位ブロック図



参考資料

31.4.2 DCMI 内部信号

表 237 に DCMI 内部信号を示します。

表 237. DCMI 内部信号

名前	信号タイプ	説明
dcmi_dma	デジタル出力	DCMI DMA リクエスト
dcmi_it	デジタル出力	DCMI 割り込みリクエスト
dcmi_hclk	デジタル入力	DCMI インタフェースクロック

31.4.3 DMA インタフェース

DCMI_CR レジスタの CAPTURE ビットがセットされているときには、DMA インタフェースがアクティブです。カメラインタフェースがレジスタにまとまった 32 ビットデータブロックを受信するたびに、DMA リクエストが生成されます。

31.4.4 DCMI 物理インタフェース

インタフェースは、11/13/15/17 個の入力で構成されています。スレーブモードにのみ対応しています。

カメラインタフェースは、DCMI_CR レジスタの EDM[1:0] ビットに従って、8、10、12、または 14 ビットのデータをキャプチャすることができます。使用されるのが 14 ビット未満である場合、使用されない入力ピンはグラウンドに接続する必要があります。

表 238 に DCMI ピンを示します。

表 238. DCMI 外部信号

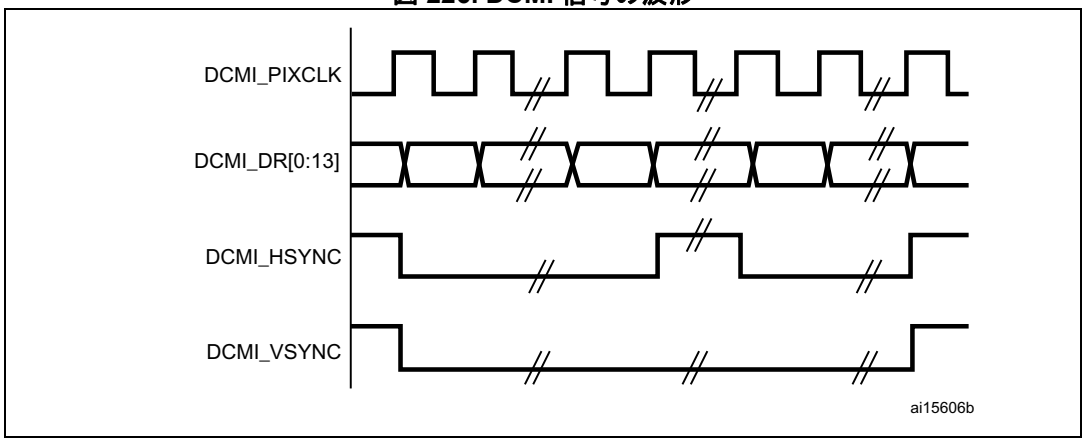
信号名		信号タイプ	信号説明
8 ビット 10 ビット 12 ビット 14 ビット	DCMI_D[0..7] DCMI_D[0..9] DCMI_D[0..11] DCMI_D[0..13]	デジタル入力	DCMI データ
DCMI_PIXCLK		デジタル入力	ピクセルクロック
DCMI_HSYNC		デジタル入力	水平同期/データ有効
DCMI_VSYNC		デジタル入力	垂直同期

データは DCMI_PIXCLK と同期していますが、ピクセルクロックの立ち上がり/立ち下がりエッジにおけるデータは極性によって変わります。

DCMI_HSYNC 信号は、ラインの開始/終了を示します。

DCMI_VSYNC 信号は、フレームの開始/終了を示します。

図 226. DCMI 信号の波形



1. DCMI_PIXCLK のキャプチャエッジは立ち下がりエッジであり、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

8 ビットデータ

DCMI_CR の EDM[1:0] が“00”にプログラムされていると、このインタフェースは入力（DCMI_D[0:7]）で LSB の 8 ビットをキャプチャし、それを 8 ビットデータとして記憶します。DCMI_D[13:8] 入力は無視されます。この場合、32 ビットワードをキャプチャするには、カメラインタフェースでピクセルクロック 4 サイクルが必要です。

最初にキャプチャされたデータバイトは 32 ビットワードの LSB 位置に置かれ、4 番目にキャプチャされたデータバイトは 32 ビットワードの MSB 位置に置かれます。*表 239* に、2 つの 32 ビットワードでキャプチャされたデータバイトの配置例を示します。

表 239. 32 ビットワード（8 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

10 ビットデータ

DCMI_CR の EDM[1:0] が“01”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..9]）で10 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 10 ビットとして記憶します。DCMI_DR レジスタの残りの MSB（ビット 11 から 15）はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 240 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 240. 32 ビットワード（10 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:26	25:16	15:10	9:0
0	0	D_{n+1}[9:0]	0	D_n[9:0]
4	0	D_{n+3}[9:0]	0	D_{n+2}[9:0]

12 ビットデータ

DCMI_CR の EDM[1:0] が“10”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..11]）で12 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 12 ビットとして記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 241 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 241. 32 ビットワード（12 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:28	27:16	15:12	11:0
0	0	D_{n+1}[11:0]	0	D_n[11:0]
4	0	D_{n+3}[11:0]	0	D_{n+2}[11:0]

14 ビットデータ

DCMI_CR の EDM[1:0] が“11”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..13]）で14 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 14 ビットとして記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 242 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 242. 32 ビットワード（14 ビット幅）でキャプチャされたデータバイトの配置

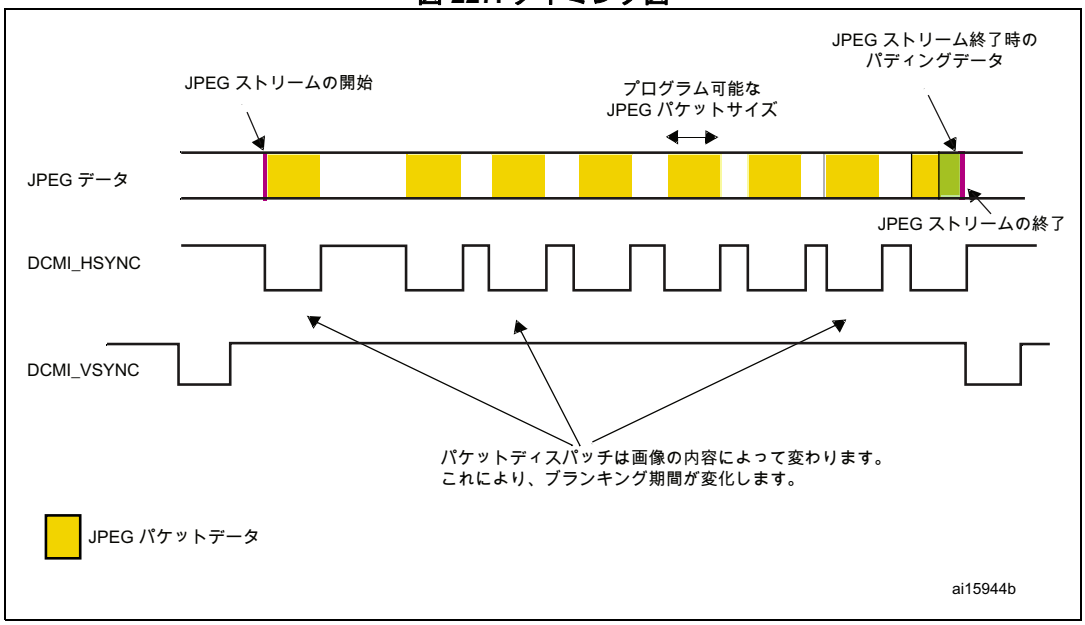
バイトアドレス	31:30	29:16	15:14	13:0
0	0	D_{n+1}[13:0]	0	D_n[13:0]
4	0	D_{n+3}[13:0]	0	D_{n+2}[13:0]

31.4.5 同期

このデジタルカメラインタフェースは、データ同期またはハードウェア（DCMI_HSYNC および DCMI_VSYNC）同期に対応します。埋め込み同期を使用する場合、0x00 および 0xFF の値が（有効画像データではなく）同期のためにのみ使用されているのを確認するのはデジタルカメラモジュールの仕様に依存します。埋め込み同期コードは、8 ビットパラレルデータインタフェース幅（すなわち DCMI_CR レジスタの EDM[1:0] ビットを“00”にクリア）に対してのみサポートされています。

圧縮データに対しては、DCMI はハードウェア同期モードにのみ対応します。この場合、DCMI_VSYNC は画像の開始／終了として使用し、DCMI_HSYNC はデータ有効信号として使用されます。*図 227* に対応するタイミング図を示します。

図 227. タイミング図



ハードウェア同期モード

ハードウェア同期モードでは、2 つの同期信号（DCMI_HSYNC/DCMI_VSYNC）を使用します。

カメラモジュール／モードによっては、データが水平／垂直同期期間中に転送されることがあります。DCMI_HSYNC/DCMI_VSYNC のアクティブ期間に受信されるデータはすべて無視されるため、これらの信号はブランキング信号のように振る舞います。

また、画像を正しく DMA/RAM バッファに転送するため、データ転送は DCMI_VSYNC 信号と同期します。ハードウェア同期モードを選択し、キャプチャを有効（DCMI_CR の CAPTURE ビットをセットする）にすると、データ転送は DCMI_VSYNC 信号の無効化（フレームの次の開始）と同期します。

その後、連続したバッファまたは同じ／サーキュラバッファへ DMA によって転送された連続フレームにより、連続的な転送ができるようになります。連続したフレームの DMA 管理を可能にするため、各フレームの終了時には VSIF（垂直同期割り込みフラグ）がアクティブになります。

データ同期モード

この同期モードでは、データフローに埋め込まれている 32 ビットコードを使用してデータフローを同期させます。このコードには、有効画像データでは使用されて*いない* 0x00/0xFF 値を使用します。コードには 4 種類あり、いずれも 0xFF0000XY フォーマットを用います。埋め込み同期コードは、8 ビットパラレルデータ幅キャプチャ（DCMI_CR レジスタの EDM[1:0] ビットを“00”にプログラム）においてのみサポートされています。他のデータ幅に対しては、このモードは予期しない動作となるため、使用しないでください。

注： ***カメラモジュールでは、このようなコードが8 つ使用可能です（インタリーブモード）。このため、インタリーブモードはDCMI ではサポートされません（ハーフフレームごとに破棄されてしまうため）。***

- モード 2 :

  4 種類の同期コードにより、次のイベントが通知されます。

  – フレーム開始（FS）
  – フレーム終了（FE）
  – ライン開始（LS）
  – ライン終了（LE）

  4 種類のコードの 0xFF0000XY フォーマットにおける XY 値は、プログラム可能です（*セクション 31.7.7 : DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）*を参照）。

  「フレーム終了」としてプログラムされた値“0xFF”は、使用されていないすべてのコードが有効なフレーム終了コードとして解釈されることを意味します。

  このモードでは、カメラインタフェースが有効になると、最初に発生したフレーム終了（FE）コードの後にフレーム開始（FS）コードが続くと、フレームのキャプチャを開始します。

- モード 1 :

  もう一方のデータ同期モードとして、カメラモード 1 が使用できます。このモードは ITU656 対応です。

  以下の同期コードにより次の各イベントが通知されます。

  – SAV（アクティブライン）- ライン開始
  – EAV（アクティブライン）- ライン終了
  – SAV（ブランキング）- インタフレームブランキング期間中のライン終了
  – EAV（ブランキング）- インタフレームブランキング期間中のライン終了

このモードは、次のコードをプログラムすることにより対応できます。

- FS ≤ 0xFF
- FE ≤ 0xFF
- LS ≤ SAV（アクティブ）
- LS ≤ EAV（アクティブ）

埋め込み型マスク解除コードもまた、フレーム／ラインの開始コードおよびラインの終了コード用に準備されています。これを使用すれば、プログラムされたコードに対して選択したマスク解除ビットのみを比較することができます。したがって、1 つのビットを選択して埋め込みコード内で比較し、フレーム／ラインの開始や終了を検出することができます。つまり、マスク解除ビットの位置が同じでも、フレーム／ラインの開始や終了のコードは異なる可能性があるということです。

例

FS = 0xA5

FS のマスク解除コード = 0x10

この場合、フレーム開始コードがフレーム開始コードのビット 4 に埋め込まれています。

31.4.6 キャプチャモード

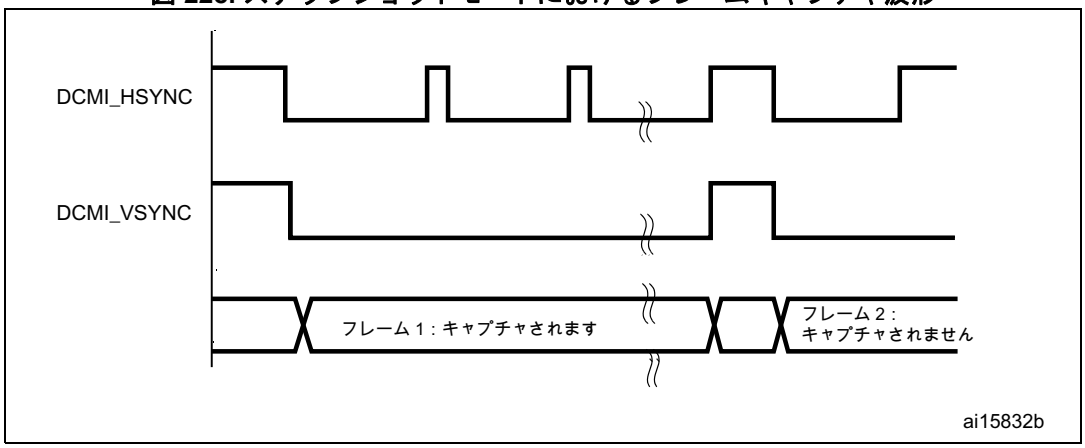
このインタフェースは、スナップショット（単一フレーム）、連続グラブの 2 種類のキャプチャをサポートしています。

スナップショットモード（単一フレーム）

このモードでは、1 つのフレームをキャプチャします（DCMI_CR レジスタの CM = 1）。DCMI_CR で CAPTURE ビットがセットされると、インタフェースはフレームの開始が検出されるのを待ってからデータをサンプリングします。最初の完全なフレームを受信した後、カメラインタフェースは自動的に無効になります（DCMI_CR の CAPTURE ビットがクリアされます)。割り込みが有効になっている場合、割り込み（IT_FRAME）が生成されます。

オーバーランの場合、フレームが失われ、CAPTURE ビットがクリアされます。

図 228. スナップショットモードにおけるフレームキャプチャ波形

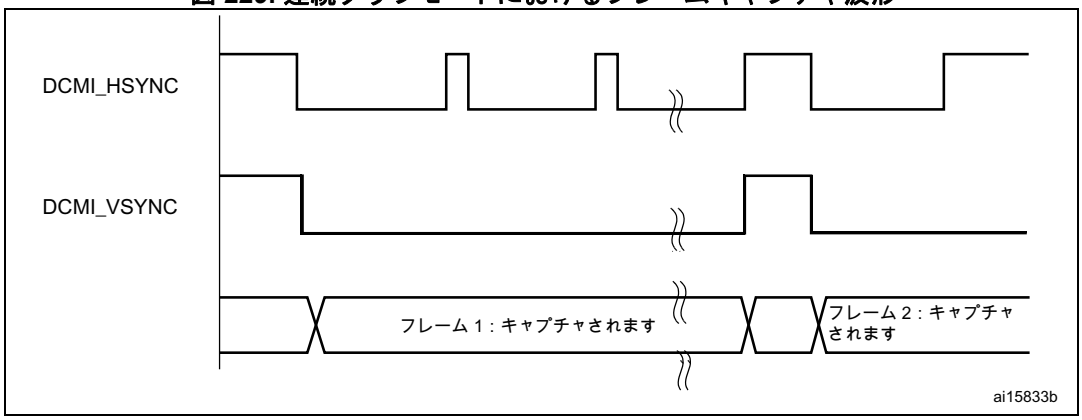


1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

連続グラブモード

このモード（DCMI_CR の CM ビット = 0）では、DCMI_CR の CAPTURE ビットがセットされると、モードによっては次の DCMI_VSYNC または埋め込みフレーム開始でグラブ処理が開始されます。この処理は、DCMI_CR の CAPTURE ビットがクリアされるまで続きます。いったん CAPTURE ビットがクリアされると、現在のフレームの終了までグラブ処理が続きます。

図 229. 連続グラブモードにおけるフレームキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

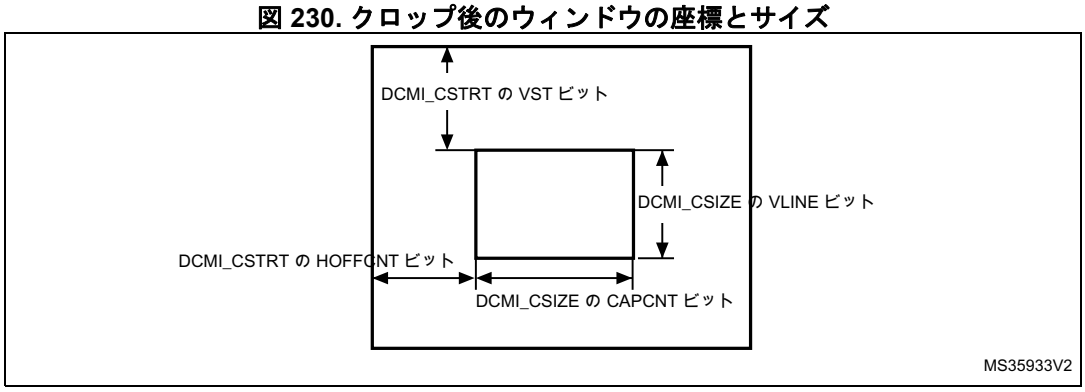
連続グラブモードでは、DCMI_CR で FCRC ビットを設定し、すべての画像をグラブ、1 枚おきにグラブ、または 4 枚おきにグラブしてフレームキャプチャ速度を下げることができます。

注： ***ハードウェア同期モード (DCMI_CR の ESS = 0) では、IT_VSYNC 割り込みが有効になっていれば DCMI_CR で CAPTURE = 0 になっていても IT_VSYNC 割り込みが発生しますので、スナップショットモードと併せて IT_VSYNC 割り込みを使用し、2 つのキャプチャ間でのフレーム数を数えることにより、さらにフレームキャプチャレートを下げることができます。これは埋め込みデータ同期モードでは行うことができません。***

31.4.7 クロップ機能

クロップ機能を用いて、カメラインタフェースが受信した画像から長方形のウィンドウを選択することができます。2 つの 32 ビットレジスタ（DCMI_CWSTRT および DCMI_CWSIZE）を使用して開始（左上の角）座標とサイズ（水平方向はピクセルクロック数、垂直方向はライン数）を指定します。ウィンドウのサイズは、ピクセルクロック数（水平方向）とライン数（垂直方向）で指定します。

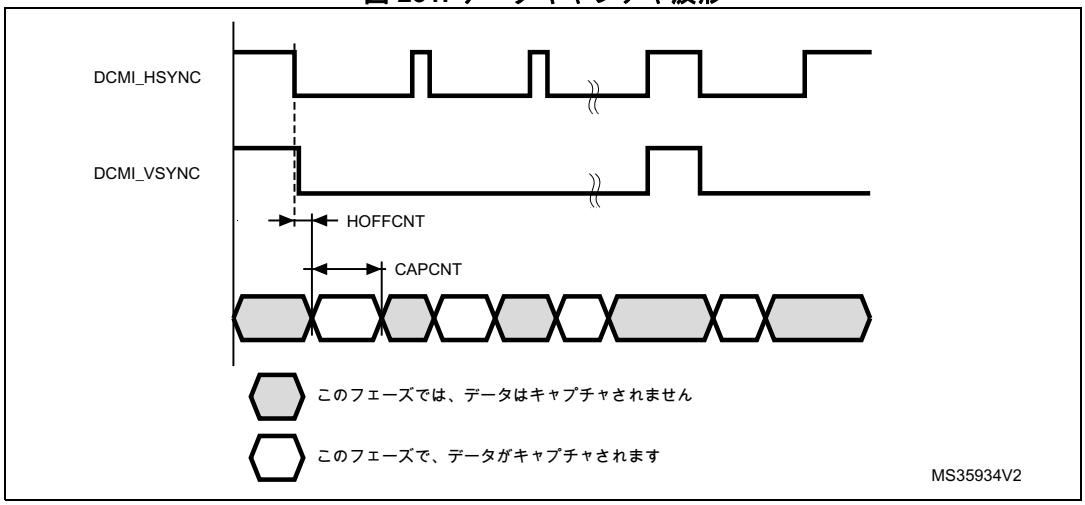
図 230. クロップ後のウィンドウの座標とサイズ



これらのレジスタを用いて、キャプチャウィンドウの開始点の座標をライン番号（フレーム内、0 から開始）とピクセルクロック数（ライン上、0 から開始）で指定し、ウィンドウのサイズをライン番号とピクセルクロック数で指定します。CAPCNT 値は、DMA を介してデータを正しく転送できるよう、必ず 4 の倍数となります（LSB の 2 ビットは強制的に 0）。

DCMI_CWSIZE レジスタでライン番号が指定される前に VSYNC 信号がアクティブになると、キャプチャが停止し、IT_FRAME 割り込みが有効になっていれば IT_FRAME 割り込みが生成されます。

図 231. データキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

31.4.8 JPEG フォーマット

JPEG 画像を受信できるようにするには、DCMI_CR レジスタの JPEG ビットをセットする必要があります。JPEG 画像はラインやフレームとして保存されないので、DCMI_HSYNC がデータイネーブル信号として使用されている間に、DCMI_VSYNC 信号を使用してキャプチャを開始します。1 ライン内のバイト数は 4 の倍数でないこともあります。キャプチャされたデータから完全な 32 ビットワードが構成されるたびに DMA リクエストが生成されるため、この場合のデータ操作には注意が必要です。フレームの終了が検出され、転送される 32 ビットワードが完全に受信されていない場合、残りのデータは“0”でパディングされ、DMA リクエストが生成されます。

クロップ機能および埋め込み同期コードは、JPEG 形式では使用できません。

31.4.9 FIFO

入力モード

AHB 上でのデータ転送レートを管理するため、4 ワードの FIFO が搭載されています。このDCMI には、カメラインタフェースが AHB から読み出しを行うたびにインクリメントされる読み出しポインタと、FIFO に書き込みを行うたびにインクリメントされる書き込みポインタを持った単純な FIFO コントローラを備えています。AHB インタフェースがデータ転送速度を維持できない場合にデータの上書きを防ぐオーバーラン保護はありません。

オーバーランや同期信号でのエラーの場合、FIFO はリセットされ、DCMI インタフェースは新規フレームの開始を待ちます。

参考資料

31.5 データフォーマットの説明

31.5.1 データフォーマット

サポートされるデータは次の 3 種類です。

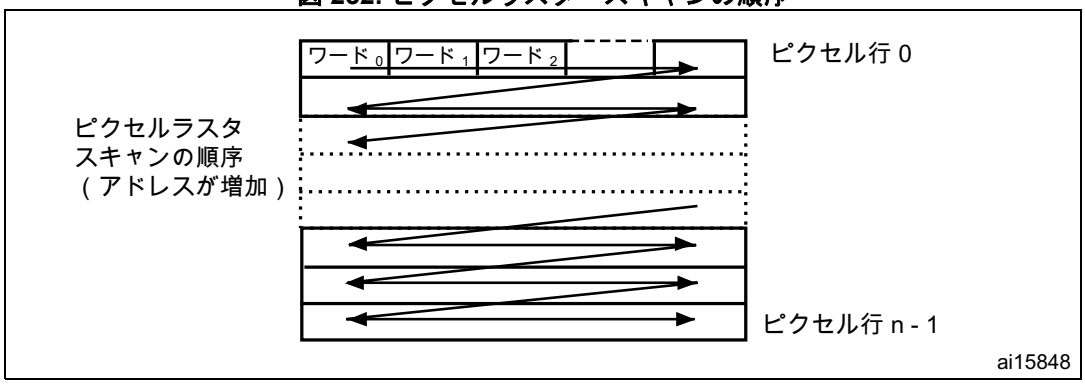
- 8 ビットのプログレッシブビデオ：モノクロまたは Raw ベイヤーフォーマット
- YCbCr 4:2:2 プログレッシブビデオ
- RGB565 プログレッシブビデオ。16 ビット（青が 5 ビット、赤が 5 ビット、緑が 6 ビット）で符号化されたピクセルの転送には、クロック 2 サイクルが必要です。

圧縮データ：JPEG

モノクロ、YCbCr、RGB データの場合、最大入力サイズは 2048 × 2048 ピクセルです。JPEG 圧縮モードでは制限はありません。

モノクロ、RGB、YCbCr の場合、フレームバッファはラスターモードで保存されます。32 ビットワードが使用されます。リトルエンディアン形式のみをサポートします。

図 232. ピクセルラスタースキャンの順序



31.5.2 モノクロフォーマット

特徴：

- ラスタフォーマット
- ピクセル当たり 8 ビット

表 243 に、データ保存方法を示します。

表 243. モノクロプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	n + 3	n + 2	n + 1	n
4	n + 7	n + 6	n + 5	n + 4

参考資料

31.5.3 RGB フォーマット

特徴：

- ラスタフォーマット
- RGB
- インタリーブ：バッファ 1 つ：R、G、B をインタリーブ：BRGBRGBRG、など。
- ディスプレイ出力に対して最適

RGB プレーナフォーマットは、標準 OS フレームバッファディスプレイフォーマットと互換性があります。
サポートされるのは 16 BPP（bits per pixel）：RGB565（32 ビットワード当たり 2 ピクセル）だけです。24 BPP（パレットフォーマット）とグレースケールフォーマットは、サポートされません。ピクセルはラスタースキャン順、つまりピクセル行に対して上から下、ピクセル行内では左から右の順で保存されます。ピクセルコンポーネントは R（赤）、G（緑）、B（青）です。すべてのコンポーネントの空間分解能は同じ（4：4：4 フォーマット）です。ピクセル単位でインタリーブされたコンポーネントと共に、フレームは単一パートで保存されます。

表 244 に、データ保存方法を示します。

表 244. RGB プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:27	26:21	20:16	15:11	10:5	4:0
0	赤 n + 1	緑 n + 1	青 n + 1	赤 n	緑 n	青 n
4	赤 n + 4	緑 n + 3	青 n + 3	赤 n + 2	緑 n + 2	青 n + 2

31.5.4 YCbCr フォーマット

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- インタリーブ：バッファ 1 つ：Y、Cb、Cr をインタリーブ：CbYCrYCbYCr、など。

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。各コンポーネントは 8 ビットで符号化されます。輝度と色度は、*表 245* に示すように、（インタリーブして）一緒に保存されます。

表 245. YCbCr プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 1	Cr n	Y n	Cb n
4	Y n + 3	Cr n + 2	Y n + 2	Cb n + 2

参考資料

31.5.5 YCbCr フォーマット - Y のみ

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- バッファには Y 情報のみが含まれる - モノクロ画像

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。このモードでは、色度情報がドロップされます。8 ビットでエンコードされた各ピクセルのルマコンポーネントのみが*表 246* に示すように保存されます。

結果として、元の YCbCr データと同じ分解能を持つモノクロ画像が得られます。

表 246. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 3	Y n + 2	Y n + 1	Y n
4	Y n + 7	Y n + 6	Y n + 5	Y n + 4

31.5.6 ハーフ解像度画像抽出

以前の受信モードを変更したもので、モノクロ、RGB、または Y 抽出モードに適用可能です。

このモードでは、ハーフ解像度の画像のみを保存できます。OELS および LSM 制御ビットを通して選択されます。

31.6 DCMI 割り込み

5 種類の割り込みが生成されます。すべての割り込みは、ソフトウェアでマスク可能です。グローバル割り込み（dcmi_it）は、個別割り込みすべての論理和（OR）です。*表 247* にすべての割り込みのリストを示します。

表 247. DCMI 割り込み

割り込み名	割り込みイベント
IT_LINE	ライン終了を示します
IT_FRAME	フレームキャプチャ終了を示します
IT_OVR	データ受信のオーバーランを示します
IT_VSYNC	同期フレームを示します
IT_ERR	埋め込み同期フレーム検出におけるエラーの検出を示します
dcmi_it	上記割り込みの論理和（OR）です

31.7 DCMI レジスタの説明

すべての DCMI レジスタは、32 ビットワードとしてアクセスする必要があり、そうしないとバスエラーが発生します。

31.7.1 DCMI 制御レジスタ（DCMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM		Res.	ENABLE	Res.	Res.	EDM		FCRC		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
											rw	rw	rw	rw	rw		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **OELS**：奇数／偶数ライン選択（ライン選択開始）

このビットは LSM フィールドと連携して動作します（LSM = 1）。

0：インタフェースはフレーム開始後に最初のラインをキャプチャします（2 番目のラインはドロップされる）。

1：インタフェースはフレーム開始から 2 番目のラインをキャプチャします（最初のラインはドロップされる）。

ビット 19 **LSM**：ライン選択モード

0：インタフェースはすべての受信したラインをキャプチャします。

1：インタフェースは 2 つのラインのうちの 1 つをキャプチャします。

ビット 18 **OEBS**：奇数／偶数バイト選択（バイト選択開始）

このビットは BSM フィールドと連携して動作します（BSM <> 00）。

0：インタフェースはフレーム／ライン開始から最初のデータ（バイトまたはダブルバイト）をキャプチャします（2 番目のデータはドロップされる）。

1：インタフェースはフレーム／ライン開始から 2 番目のデータ（バイトまたはダブルバイト）をキャプチャします（最初のデータはドロップされる）。

ビット 17:16 **BSM[1:0]**：バイト選択モード

00：インタフェースはすべての受信したデータをキャプチャします。

01：インタフェースは受信したデータから 1 つおきにバイトをキャプチャします。

10：インタフェースは 4 つのバイトのうちの 1 つをキャプチャします。

11：インタフェースは 4 つのバイトのうちの 2 つをキャプチャします。

注： ***このモードは EDM[1:0]=00 でのみ動作します。その他すべての EDM 値に対し、このビットフィールドをリセット値にプログラムする必要があります。***

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ENABLE**：DCMI イネーブル

0：DCMI 無効

1：DCMI 有効

注： ***このビットを有効にする前に、DCMI 設定レジスタを正しくプログラムする必要があります。***

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはピクセルクロックごとに 8 ビットデータをキャプチャします
01：インタフェースはピクセルクロックごとに 10 ビットデータをキャプチャします
10：インタフェースはピクセルクロックごとに 12 ビットデータをキャプチャします
11：インタフェースはピクセルクロックごとに 14 ビットデータをキャプチャします

ビット 9:8 **FCRC[1:0]**：フレームキャプチャレート制御

このビットによって、フレームキャプチャの頻度を規定します。これらのビットは、連続グラブモードでなければ意味を持ちません。スナップショットモードでは無視されます。

00：全フレームをキャプチャ
01：フレームを 1 つおきにキャプチャ（帯域幅 50 % 減少）
10：4 フレームごとに 1 フレームキャプチャ（帯域幅 75 % 減少）
11：予約済み

ビット 7 **VSPOL**：垂直同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_VSYNC ピン上でのレベルを示します。

0：DCMI_VSYNC はアクティブローです。
1：DCMI_VSYNC はアクティブハイです。

ビット 6 **HSPOL**：水平同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_HSYNC ピン上でのレベルを示します。

0：DCMI_HSYNC はアクティブローです。
1：DCMI_HSYNC はアクティブハイです。

ビット 5 **PCKPOL**：ピクセルクロック極性

このビットによってピクセルクロックのキャプチャエッジを設定します。

0：立ち下がりエッジでアクティブ
1：立ち上がりエッジでアクティブ

ビット 4 **ESS**：埋め込み同期選択

0：ハードウェア同期。データキャプチャ（フレーム／ ラインの開始／ 終了）は DCMI_HSYNC/DCMI_VSYNC 信号と同期します。
1：埋め込み同期。データキャプチャはデータフローに埋め込まれた同期コードと同期します。

注： 8 ビットのパラレルデータに対してのみ有効です。ESS ビットがセットされると、HSPOL/VSPOL は無視されます。

このビットは、JPEG モードでは無効になります。

ビット 3 **JPEG**：JPEG フォーマット

0：非圧縮ビデオフォーマット
1：このビットは JPEG データ転送に使用します。DCMI_HSYNC 信号はデータイネーブルとして使用します。このモードでは、クロップおよび埋め込み同期機能（ESS ビット）は使用できません。

ビット 2 **CROP**：クロップ機能

0：画像全体がキャプチャされます。この場合、1 枚の画像フレーム内の総バイト数は 4 の倍数となります。
1：クロップレジスタで指定されたウィンドウ内のデータのみがキャプチャされます。クロップウィンドウのサイズがピクチャサイズを超える場合は、そのピクチャサイズのみがキャプチャされます。

ビット 1 **CM**：キャプチャモード

0：連続グラブモード - 受信したデータは DMA を通じて転送先メモリに転送されます。バッファの場所やモード（リニアバッファまたはサーキュラバッファ）は、システム DMA を通じて制御されます。

1：スナップショットモード（単一フレーム）- アクティブになると、インタフェースはフレーム開始を待ってから DMA を通じて単一フレームを転送します。フレーム終了時、CAPTURE ビットは、自動的にリセットされます。

ビット 0 **CAPTURE**：キャプチャイネーブル

0：キャプチャは無効です。

1：キャプチャは有効です。

カメラインタフェースは最初のフレーム開始を待ち、その後 DMA リクエストが生成されて受信したデータが転送先メモリに転送されます。

スナップショットモードでは、受信した最初のフレームの終了時に CAPTURE ビットが自動的にクリアされます。

連続グラブモードにおいて、キャプチャ進行中にソフトウエアがこのビットをクリアすると、そのビットは実質的にフレーム終了時にクリアされます。

注： ***DMA コントローラとすべての DCMI 設定レジスタを正しくプログラムしてから、このビットを有効にする必要があります。***

31.7.2 DCMI ステータスレジスタ（DCMI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0x0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FNE**：FIFO ノットエンプティ

このビットは、FIFO のステータスを示します。

1：FIFO に有効なデータあり

0：FIFO エンプティ

ビット 1 **VSYNC**

このビットは、極性が正しくプログラムされた DCMI_VSYNC ピンの状態を示します。

埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブフレーム

1：フレーム間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

ビット 0 **HSYNC**

このビットは、極性が正しくプログラムされた DCMI_HSYNC ピンの状態を示します。

埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブライン

1：ライン間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

31.7.3 DCMI raw 割り込みステータスレジスタ（DCMI_RIS）

アドレスオフセット：0x08

リセット値：0x0000 0x0000

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

DCMI_RIS は raw 割り込みステータスを示し、読み出し専用でアクセスできます。読み出し時、このレジスタは DCMI_IER レジスタ値でマスクする前の関連する割り込みのステータスを返します。

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_RIS**：ライン raw 割り込みステータス

DCMI_HSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。ラインが有効でなくてもハイになります。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR の LINE_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 3 **VSYNC_RIS**：DCMI_VSYNC raw 割り込みステータス

DCMI_VSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR の VSYNC_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 2 **ERR_RIS**：同期エラー raw 割り込みステータス

0：同期エラーは検出されていません。

1：埋め込み同期文字が正しい順番で受信されません。

このビットは、埋め込み同期モードでのみ有効です。DCMI_ICR の ERR_ISC ビットに“1”を書き込むことによってクリアされます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_RIS**：オーバーラン raw 割り込みステータス

0：データバッファオーバーランは発生していません。

1：データバッファオーバーランが発生し、データ FIFO が不正です。

このビットは、DCMI_ICR の OVR_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 0 **FRAME_RIS**：キャプチャ完了 raw 割り込みステータス

0：新規キャプチャはありません。

1：フレームキャプチャ完了しています。

このビットは、フレームやウィンドウがキャプチャされるとセットされます。

クロップされたウィンドウの場合、このビットは、クロップの最後のラインの終了時にセットされます。キャプチャされたフレームが空でもセットされます（例：フレーム外でクロップされたウィンドウなど）。

このビットは、DCMI_ICR の FRAME_ISC ビットに“1”を書き込むことによってクリアされます。

31.7.4 DCMI 割り込み有効レジスタ（DCMI_IER）

アドレスオフセット：0x0C

リセット値：0x0000 0x0000

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

DCMI_IER レジスタは、割り込みを有効にするために使用します。DCMI_IER ビットの 1 つがセットされると、該当する割り込みが有効になります。このレジスタは、読み出しと書き込みの両方でアクセスできます。

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_IE**：ライン割り込みイネーブル
0：ライン受信時に割り込みは生成されません。
1：ラインすべてを受信したときに割り込みが生成されます。

ビット 3 **VSYNC_IE**：DCMI_VSYNC 割り込みイネーブル
0：割り込みは生成されません。
1:DCMI_VSYNC がインアクティブからアクティブ状態に遷移するたびに割り込みが生成されます。
DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_IE**：同期エラー割り込みイネーブル
0：割り込みは生成されません。
1：埋め込み同期コードが正しい順番で受信されない場合に割り込みが生成されます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_IE**：オーバーラン割り込みイネーブル
0：割り込みは生成されません。
1：DMA が新しいデータ（32 ビット）を受信する前に最後のデータを転送できなかった場合に割り込みが生成されます。

ビット 0 **FRAME_IE**：キャプチャ完了割り込みイネーブル
0：割り込みは生成されません。
1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に割り込みが生成されます。

31.7.5 DCMI マスク済み割り込みステータスレジスタ（DCMI_MIS）

DCMI_MIS レジスタは、読み出し専用レジスタです。読み出し時、該当する割り込みの現在のマスクステータス値（DCMI_IER の値によって変わります）を返します。DCMI_IER 内の該当するイネーブルビットがセットされ、DCMI_RIS 内の該当ビットがセットされると、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_MIS**：マスクされたライン割り込みステータス

このビットは、マスクされたライン割り込みのステータスを示します。

0：ライン受信時に割り込みは生成されません。

1：ラインが完全に受信され、DCMI_IER の LINE_IE ビットがセットされると割り込みが生成されます。

ビット 3 **VSYNC_MIS**：マスクされた VSYNC 割り込みステータス

このビットは、マスクされた VSYNC 割り込みのステータスを示します。

0：DCMI_VSYNC の遷移で割り込みは生成されません。

1：DCMI_VSYNC がインアクティブからアクティブ状態に遷移し、DCMI_IER の VSYNC_IE ビットがセットされるたびに割り込みが生成されます。

DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_MIS**：マスクされた同期エラー割り込みステータス

このビットは、マスクされた同期エラー割り込みのステータスを示します。

0：同期エラー時、割り込みは生成されません。

1：埋め込み同期コードが正しい順番で受信されず、DCMI_IER の ERR_IE ビットがセットされると割り込みが生成されます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_MIS**：マスクされたオーバーラン割り込みステータス

このビットは、マスクされたオーバ-フロー割り込みのステータスを示します。

0：オーバーラン時、割り込みは生成されません。

1：DMA が新しいデータ（32 ビット）を受信する前に最後のデータを転送できず、DCMI_IER の OVR_IE ビットがセットされると割り込みが生成されます。

ビット 0 **FRAME_MIS**：マスクされたキャプチャ完了割り込みステータス

このビットは、マスクされたキャプチャ完了割り込みのステータスを示します。

0：キャプチャ完了で割り込みは生成されません。

1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に DCMI_IER の FRAME_IE ビットがセットされると割り込みが生成されます。

31.7.6 DCMI 割り込みクリアレジスタ（DCMI_ICR）

アドレスオフセット：0x14

リセット値：0x0000 0x0000

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

DCMI_ICR レジスタは、書き込み専用です。このレジスタのビットに“1”を書き込むことによって DCMI_RIS および DCMI_MIS レジスタの対応するビットをクリアします。“0”を書き込んでも、ビットの値は変化しません。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_ISC**：ライン割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの LINE_RIS ビットがクリアされます。

ビット 3 **VSYNC_ISC**：垂直同期割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの VSYNC_RIS ビットがクリアされます。

ビット 2 **ERR_ISC**：同期エラー割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの ERR_RIS ビットがクリアされます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_ISC**：オーバーラン割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS の OVR_RIS ビットがクリアされます。

ビット 0 **FRAME_ISC**：キャプチャ完了割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS の FRAME_RIS ビットがクリアされます。

31.7.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）

アドレスオフセット：0x18

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEC								LEC								LSC								FSC							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **FEC**：フレーム終了デリミタコード

このバイトによって、フレーム終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FEC の形式で 4 バイトで構成されています。

FEC が 0xFF にプログラムされている場合、使用されていないすべてのコード（0xFF0000XY）はフレーム終了デリミタとして解釈されます。

ビット 23:16 **LEC**：ライン終了デリミタコード

このバイトによって、ライン終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LEC の形式で 4 バイトで構成されています。

ビット 15:8 **LSC**：ライン開始デリミタコード

このバイトによって、ライン開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LSC の形式で 4 バイトで構成されています。

ビット 7:0 **FSC**：フレーム開始デリミタコード

このバイトによって、フレーム開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FSC の形式で 4 バイトで構成されています。

FSC が 0xFF にプログラムされている場合、フレーム開始デリミタは検出されません。

FEC コードの後に現れる最初の LSC は、フレームの開始デリミタとして解釈されます。

31.7.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR）

アドレスオフセット：0x1C

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEU								LEU								LSU								FSU							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **FEU**：フレーム終了デリミタマスク解除

このバイトによって、フレーム終了デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットがマスクされます。

1 : 受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットが比較されます。

ビット 23:16 **LEU**：ライン終了デリミタマスク解除

このバイトによって、ライン終了デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットがマスクされます。

1 : 受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットが比較されます。

ビット 15:8 **LSU**：ライン開始デリミタマスク解除

このバイトによって、ライン開始デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットがマスクされます。

1 : 受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットが比較されます。

ビット 7:0 **FSU**：フレーム開始デリミタマスク解除

このバイトによって、フレーム開始デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットがマスクされます。

1 : 受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットが比較されます。

31.7.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT）

アドレスオフセット：0x20

リセット値：0x0000 0x0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:16 **VST[12:0]**：垂直方向開始ラインカウント

このライン番号で画像キャプチャを開始します。これより前のラインデータは無視されます。

0x0000 => ライン 1

0x0001 => ライン 2

0x0002 => ライン 3

....

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **HOFFCNT[13:0]**：水平方向オフセットカウント

キャプチャ開始前にカウントするピクセルクロック数を示します。

31.7.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE）

アドレスオフセット：0x24

リセット値：0x0000 0x0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **VLINE[13:0]**：垂直方向ラインカウント

開始点からのキャプチャライン数を示します。

0x0000 => 1 ライン

0x0001 => 2 ライン

0x0002 => 3 ライン

....

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **CAPCNT[13:0]**：キャプチャカウント

同一ライン上でキャプチャする開始点からのピクセルクロック数を示します。この値は、パラレルインタフェースのさまざまなデータ幅に対してワードアライメントが施されたデータに対応します。

0x0000 => 1 ピクセル

0x0001 => 2 ピクセル

0x0002 => 3 ピクセル

....

31.7.11 DCMI データレジスタ（DCMI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Byte3								バイト 2								Byte1								Byte0							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 データバイト 3

ビット 23:16 データバイト 2

ビット 15:8 データバイト 1

ビット 7:0 データバイト 0

このデジタルカメラインタフェースは、DMA 転送を要求する前に受信したデータをすべて 32 ビット形式にまとめます。DMA 転送に十分な時間を与え、DMA オーバーランを回避するため、深さ 4 ワードの FIFO が使用できます。

31.7.12 DCMI レジスタマップ

表 248 に DCMI レジスタの要約を示します。

表 248. DCMI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DCMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM		Res.	ENABLE	Res.	Res.	EDM		FCRC		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	リセット値												0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0	0	0
0x04	DCMI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FNE	VSYNC	HSYNC
	リセット値																														0	0	0
0x08	DCMI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
	リセット値																												0	0	0	0	0
0x0C	DCMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_IE	VSYNC_IE	ERR_IE	OVR_IE	FRAME_IE
	リセット値																												0	0	0	0	0
0x10	DCMI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
	リセット値																												0	0	0	0	0
0x14	DCMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
	リセット値																												0	0	0	0	0
0x18	DCMI_ESCR	FEC								LEC								LSC								FSC							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	DCMI_ESUR	FEU								LEU								LSU								FSU							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DCMI_CWSTRT	Res.	Res.	Res.	VST[12:0													Res.	Res.	HOFFCNT[13:0]													
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	DCMI_CWSIZE	Res.	Res.	VLINE13:0]														Res.	Res.	CAPCNT[13:0]													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	DCMI_DR	Byte3								バイト 2								Byte1								Byte0							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

32 LCD-TFT ディスプレイコントローラ（LTDC）

32.1 概要

この LCD-TFT（TFT 液晶）ディスプレイコントローラは、さまざまな LCD および TFT パネルに直接接続するため、デジタルパラレル RGB（赤、緑、青）や、水平／垂直同期、ピクセルクロック、およびデータイネーブルの出力信号を提供します。

32.2 LTDC の主な機能

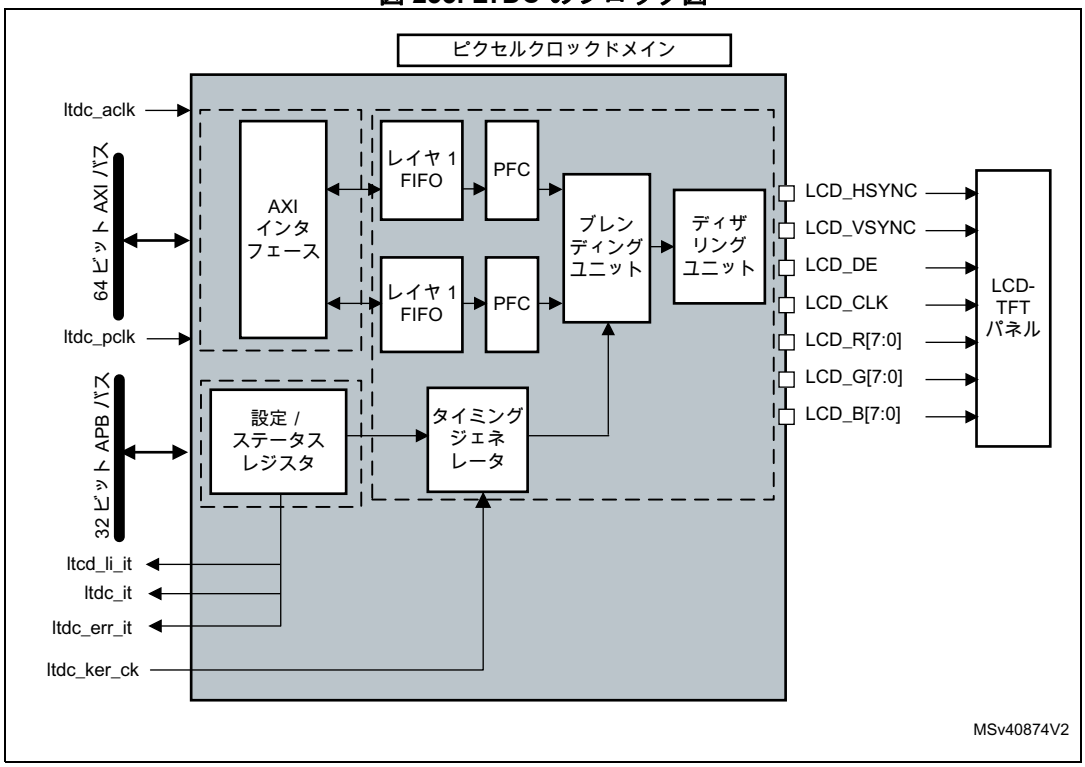
- 24 ビット RGB パラレル出力、8 ビット／ピクセル（RGB888）
- 専用 FIFO（64x64 ビット）を備えた 2 つの表示レイヤ
- 各レイヤにつき最大 256 色（256x24 ビット）のカラールックアップテーブル（CLUT）
- さまざまなディスプレイパネルに対応するプログラム可能なタイミング
- プログラム可能なバックグラウンドカラー
- HSync、VSync、データイネーブルに対するプログラム可能な極性
- 各レイヤにつき最大 8 種類の入力カラーフォーマットが選択可能
  - ARGB8888
  - RGB888
  - RGB565
  - ARGB1555
  - ARGB4444
  - L8（8 ビット輝度または CLUT）
  - AL44（4 ビットアルファ＋4 ビット輝度）
  - AL88（8 ビットアルファ 8 ビット輝度）
- 各チャネルにつき、低ビットに対する疑似ランダムディザリング出力
  - 赤、緑、青に対するディザ幅 2 ビット
- アルファ値（ピクセルごとまたは一定値）を用いた 2 つのレイヤ間の柔軟性の高いブレンディング
- カラーキーイング（透明色）
- プログラム可能なウィンドウ位置とサイズ
- TFT カラー液晶ディスプレイをサポート
- 16 ダブルワードのバースト付き AXI マスタインタフェース
- 最大 4 つのプログラム可能な割り込みイベント

32.3 LTDC の機能詳細

32.3.1 LTDC のブロック図

LTDC のブロック図を*図 233 : LTDC のブロック図*に示します。

図 233. LTDC のブロック図



レイヤ FIFO：レイヤごとに FIFO 64x64 ビットを 1 つずつ搭載しています。

PFC：レイヤに対して選択した入力ピクセルフォーマットからワードへのピクセルフォーマット変換を実行するピクセルフォーマットコンバータです。

AXI インタフェース：メモリから FIFO へのデータ転送用です。

ブレンディングユニット、ディザリングユニット、タイミングジェネレータ：*セクション 32.4.1* および *セクション 32.4.2* を参照してください。

32.3.2 LCD-TFT 内部信号

表 249 に LCD-TFT 内部信号のリストを示します。

表 249. LCD-TFT 内部信号

名前	信号タイプ	説明
ltdc_aclk	デジタル入力	AXI クロックドメインの LCD-TFT レジスタ用クロック
ltdc_pclk	デジタル入力	LCD-TFT レジスタインタフェースクロック
ltdc_ker_ck	デジタル入力	LCD_CLK（ピクセルクロック）生成に使用される LCD-TFT カーネルクロック

表 249. LCD-TFT 内部信号（続き）

名前	信号タイプ	説明
ltdc_li_it	デジタル出力	MDMA の LCD-TFT ライン割り込みトリガ
ltdc_it	デジタル出力	LCD-TFT グローバル割り込みリクエスト
ltdc_err_it	デジタル出力	LCD-TFT グローバルエラー割り込みリクエスト

32.3.3 LCD-TFT ピンと外部信号インタフェース

表 250 に、LTDC 信号インタフェースの概要を示します。

表 250. LCD-TFT ピンと信号インタフェース

LCD-TFT 信号	I/O	説明
LCD_CLK	O	クロック出力
LCD_HSYNC	O	水平同期
LCD_VSYNC	O	垂直同期
LCD_DE	O	非データイネーブル
LCD_R[7:0]	O	データ：8 ビットの赤のデータ
LCD_G[7:0]	O	データ：8 ビットの緑のデータ
LCD_B[7:0]	O	データ：8 ビットの青のデータ

LTDC-TFT コントローラのピンは、ユーザアプリケーションで設定する必要があります。使用されていないピンは、他の目的に使用できます。

24 ビットまでの LTDC 出力（RGB888）に対して、16 ビット（RGB565）または 18 ビット（RGB666）カラーのディスプレイに接続するため、8bpp 未満の色深度を使用する場合、ディスプレイ側の RGB データラインは LTDC の RGB データラインの MSB 側に接続する必要があります。例えば、16 ビット（RGB565）カラーのディスプレイを接続する場合、ディスプレイ側の R[4:0]、G[5:0]、B[4:0]のデータラインのピンは LTDC の LCD_R[7:3]、LCD_G[7:2]、LCD_B[7:3] に接続する必要があります。

32.3.4 LTDC のリセットおよびクロック

LTDC では 3 つのクロックドメインを使用します。

- AXI クロックドメイン（ltdc_aclk）

  このドメインには、メモリからレイヤ FIFO へのデータ転送のためのLCD-TFT AXI マスタインタフェースとフレームバッファ設定レジスタが含まれています。
- APB クロックドメイン（ltdc_pclk）：

  このドメインには、グローバル設定レジスタと割り込みレジスタが含まれています。
- ピクセルクロックドメイン（LCD_CLK）

  このドメインには、ピクセルデータ生成、レイヤ設定レジスタとともに LCD-TFT インタフェース信号生成器が含まれています。LCD_CLK 出力は、パネル要件に従って設定する必要があります。LCD_CLK は、特定の PLL 出力（リセットとクロック制御のセクションを参照）から生成されます。

表 251 に、各レジスタのクロックドメインの概要を示します。

表 251. 各レジスタのクロックドメイン

LTDC レジスタ	クロックドメイン
LTDC_LxCR	ltdc_aclk
LTDC_LxCFBAR	
LTDC_LxCFBLR	
LTDC_LxCFBLNR	
LTDC_SRCR	ltdc_pclk
LTDC_IER	
LTDC_ISR	
LTDC_ICR	
LTDC_SSCR	ピクセルクロック（LCD_CLK）
LTDC_BPCR	
LTDC_AWCR	
LTDC_TWCR	
LTDC_GCR	
LTDC_BCCR	
LTDC_LIPCR	
LTDC_CPSR	
LTDC_CDSR	
LTDC_LxWHPCR	
LTDC_LxWVPCR	
LTDC_LxCKCR	
LTDC_LxPFCR	
LTDC_LxCACR	
LTDC_LxDCCR	
LTDC_LxBFCR	
LTDC_LxCLUTWR	

LTDC レジスタにアクセスするとき、ある特定の時間、APB バスがストールするので、注意する必要があります（*表 252* 参照）。

表 252. LTDC レジスタアクセスと更新時間

	レジスタクロックドメイン		
	AXI ドメイン	APB ドメイン	ピクセルクロックドメイン
レジスタ読み出しアクセス時間	7 x ltdc_pclk + 5 x ltdc_aclk	7 x ltdc_pclk	7 x ltdc_pclk + 5 x ltdc_ker_clk
レジスタ書き込みアクセス時間	6 x ltdc_pclk + 5 x ltdc_aclk	6 x ltdc_pclk	6 x ltdc_pclk + 5 x ltdc_ker_clk

LCD コントローラは、RCC_APBRSTR レジスタの該当するビットをセットすることによってリセットできます。このレジスタで 3 つのクロックドメインをリセットします。

32.4 LTDC のプログラム可能なパラメータ

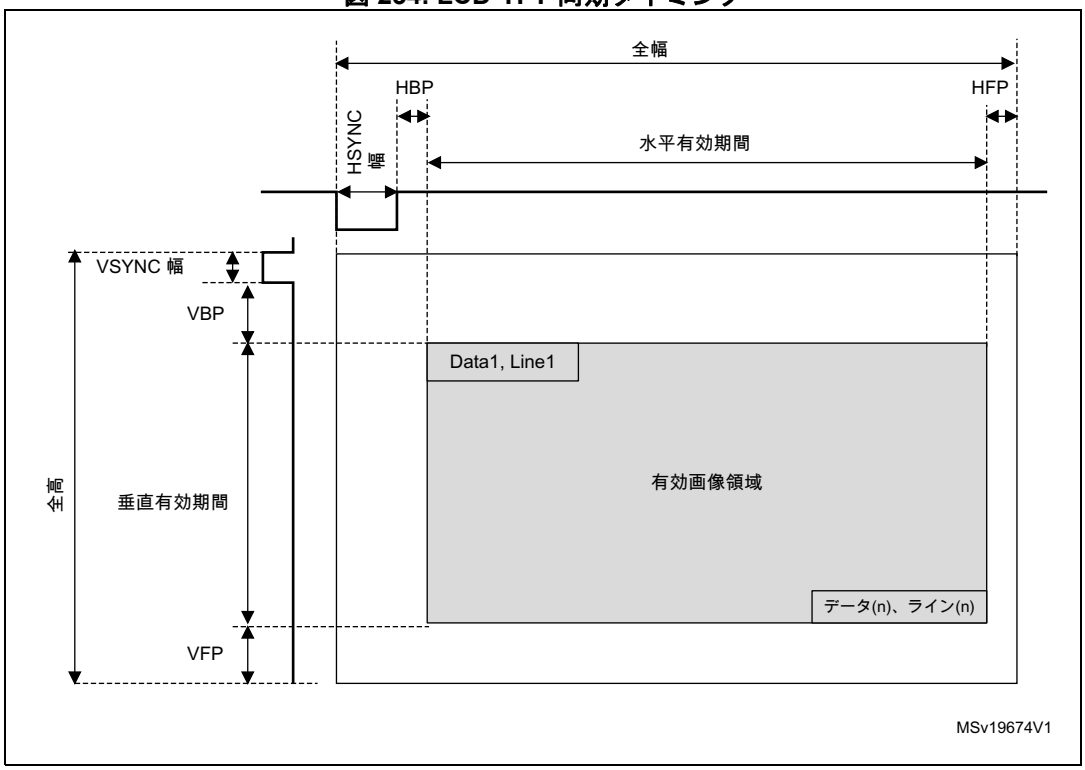
LCD-TFT コントローラのパラメータは、フレキシブルに設定可能です。**LTDC_GCR** レジスタを介して有効または無効にすることができます。

32.4.1 LTDC グローバル設定パラメータ

同期タイミング：

図 234 は、*図 233* のブロック図で示す同期タイミングジェネレータブロックで生成される設定可能なタイミングパラメータを示します。水平／垂直同期信号、ピクセルクロックおよびデータイネーブル信号が生成されます。

図 234. LCD-TFT 同期タイミング



注： ***HBP および HFP は、それぞれ水平バックポーチ期間およびフロントポーチ期間です。***

VBP および VFP は、それぞれ垂直バックポーチ期間およびフロントポーチ期間です。

LCD-TFT のプログラム可能な同期タイミングは次のとおりです。

- HSYNC および VSYNC 幅 : **LTDC_SSCR** レジスタにおける ***HSYNC 幅 - 1*** および ***VSYNC 幅 - 1*** の値をプログラムして設定される水平および垂直同期信号幅です。
- HBP および VBP : **LTDC_BPCR** レジスタにおける ***HSYNC 幅 + HBP - 1*** の合計値と ***VSYNC 幅 + VBP - 1*** の合計値をプログラムして設定される水平および垂直バックポーチ幅です。
- アクティブ幅とアクティブ高さ : **LTDC_AWCR** レジスタで ***HSYNC 幅 + HBP + 水平有効期間 - 1*** の合計値と ***VSYNC 幅 + VBP + 垂直有効期間 - 1*** の合計値をプログラムして設定します（1024x768 までをサポート）。
- 全幅 : **LTDC_TWCR** レジスタで ***HSYNC 幅 + HBP + 水平有効期間 + HFP - 1*** の合計値をプログラムして設定します。HFP は、水平フロントポーチ期間です。
- 全高 : **LTDC_TWCR** レジスタで ***VSYNC 高さ + VBP + 垂直有効期間 + VFP - 1*** の合計値をプログラムして設定します。VFP は、垂直フロントポーチ期間です。

注： ***LTDC が有効の場合、垂直同期エリアの最初の水平同期ピクセルとして、X/Y=0/0 位置でタイミング生成が開始し、その後バックポーチ、アクティブデータ表示エリア、フロントポーチと続きます。***

LTDC が無効の場合、タイミングジェネレータブロックは X = *全幅 - 1*、Y = *全高 - 1* にリセットされ、垂直同期フェーズの前の最後のピクセルを保持し、FIFO は一掃されます。したがって、連続的に出力されるのはブランキングデータのみとなります。

同期タイミング設定の例：

TFT-LCD タイミング（パネルデータシートより抽出）:

- 水平および垂直同期幅 : 0xA ピクセルおよび 0x2 ライン
- 水平および垂直バックポーチ : 0x14 ピクセルおよび 0x2 ライン
- アクティブ幅とアクティブ高さ : 0x140 ピクセル、0xF0 ライン（320x240）
- 水平フロントポーチ : 0xA ピクセル
- 垂直フロントポーチ : 0x4 ライン

LTDC タイミングレジスタでプログラムされる値は次のようになります。

- LTDC_SSCR レジスタ : 0x00090001 にプログラム（HSW[11:0] は 0x9、VSH[10:0] は 0x1）
- LTDC_BPCR レジスタ : 0x001D0003 にプログラム（AHBP[11:0] は 0x1D（0xA + 0x13）、AVBP[10:0] は 0x3（0x2 + 0x1））
- LTDC_AWCR レジスタ : 0x015D00F3 にプログラム（AAW[11:0] は 0x15D（0xA + 0x14 + 0x13F）、AAH[10:0] は 0xF3（0x2 +0x2 + 0xEF））
- LTDC_TWCR レジスタ : 0x00000167 にプログラム（TOTALW[11:0] は0x167（0xA + 0x14 + 0x140 + 0x9））
- LTDC_THCR レジスタ : 0x000000F7 にプログラム（TOTALH[10:0] は 0xF7（0x2 +0x2 + 0xF0 + 3）

プログラム可能な極性

水平／垂直同期信号、データイネーブル信号、ピクセルクロックの極性は、**LTDC_GCR** レジスタでアクティブハイまたはアクティブローにプログラムできます。

バックグラウンドカラー

一定のバックグラウンドカラー（RGB888）を **LTDC_BCCR** レジスタでプログラムできます。ボトムレイヤとのブレンディングに使用します。

ディザリング

LFSR を使用した疑似乱数ディザリングを使用して、各ピクセルカラーチャネル（R、G、または B）値に小さな乱数値（閾値）を加えますので、18 ビットディスプレイで 24 ビットデータを表示する際に、MSB が切り上げられる場合があります。このようにして、フレームごとに異なるデータの四捨五入にはディザリングが使用されす。

疑似乱数ディザリングは、LSB を閾値と比較して LSBが閾値以上であれば MSB にのみ 1 を加算することと同じです。ディザリングが適用されると、LSB は通常切り捨てられます。

加算される疑似乱数値の幅は、各カラーチャネルに対して 2 ビット、つまり赤に対して 2 ビット、緑に対して 2 ビット、青に対して 2 ビットです。

LTDC が有効になると、LFSR が最初のアクティブなピクセルで動作を開始し、ブランキング期間中やディザリングがオフになってもそのまま動作を続けます。LTDC が無効になると、LFSR はリセットされます。

ディザリングは、**LTDC_GCR** レジスタを使って動作中にオン／オフを切り替えることができます。

シャドウレジスタの再ロード

シャドウイングされる設定レジスタがいくつかあります。シャドウレジスタの値は、**LTDC_SRCR** レジスタでの設定に従い、アクティブなレジスタへの書き込み時、または垂直ブランキング期間の始めに、これらのレジスタに直ちに再ロードすることができます。即時再ロード設定が選択されている場合、すべての新規レジスタが書き込まれてから再ロードがアクティブになるようにする必要があります。

シャドウレジスタは、再ロードが完了するまでは再変更してはいけません。シャドウレジスタを読み出すと、実際にアクティブな値が読み出されます。新しく書き込まれた値は、再ロードが実行されるまでは読み出せません。

LTDC_IER レジスタで有効になっていれば、レジスタ再ロード割り込みを生成することができます。

LTDC_LxCLUTWR レジスタを除くすべてのレイヤ 1 とレイヤ 2 のレジスタはシャドウレジスタです。

割り込み生成イベント

割り込み設定については、*セクション 32.5 : LTDC 割り込み*を参照してください。

32.4.2 レイヤのプログラム可能なパラメータ

最大 2 つのレイヤが有効化、無効化でき、また別々に設定することができます。レイヤの表示順序は固定で、下から上に順番に表示されます。2 つのレイヤが有効になっている場合は、レイヤ 2 が上部に表示されるウィンドウとなります。

ウィンドウイング

すべてのレイヤは位置決めとサイズ変更ができますが、アクティブ表示エリア内であることが必要です。

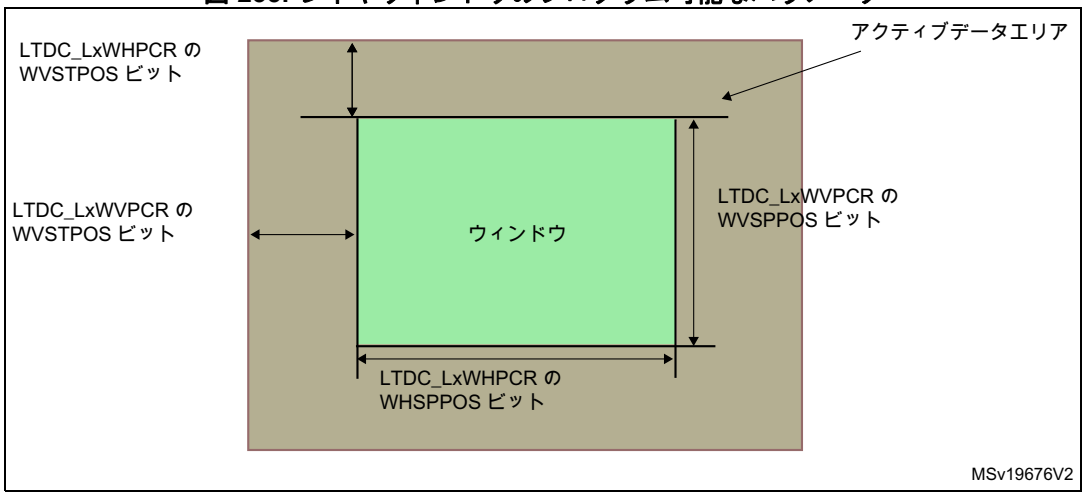
ウィンドウ位置およびサイズは、左上と右下の X/Y 位置、および内部タイミングジェネレータによって設定され、この内部タイミングジェネレータは、同期、バックポーチサイズ、アクティブデータエリアから成っています。**LTDC_LxWHPCR** および **LTDC_WVPCR** レジスタを参照してください。

プログラム可能なレイヤ位置とサイズにより、ラインの最初／最後の見えるピクセルと、ウィンドウの最初／最後の見えるラインが決まります。画像フレームの全体または一部のいずれでも表示することができます。*図 235* を参照してください。

- レイヤ内の最初と最後の見えるピクセルは、**LTDC_LxWHPCR** レジスタ内の WHSTPOS[11:0] および WHSPPOS[11:0] を設定することによりセットされます。

- レイヤ内の最初と最後の見えるラインは、**LTDC_LxWVPCR** レジスタ内の WVSTPOS[10:0] および WVSPPOS[10:0] を設定することによりセットされます。

図 235. レイヤウィンドウのプログラム可能なパラメータ



ピクセル入力フォーマット

レイヤのフレームバッファ内に保存されるデータには、プログラム可能なピクセルフォーマットが使用されます。

LTDC_LxPFCR レジスタで、各レイヤに対して最大 8 種類までの入力ピクセルフォーマットを設定できます。

ピクセルデータはフレームバッファから読み出され、次のように内部 8888 (ARGB) フォーマットに変換されます。

- 幅が 8 ビット未満のコンポーネントは、ビットレプリケーションによって 8 ビットまで拡張されます。選択されたビット範囲は、8 ビットより長くなるまで複数回連結されます。得られたベクタから、8 つの MSB ビットが選択されます。例 : RGB565 赤チャネルの 5 ビットは次のようになります（ビット位置）: 43210432（3 つの LSB に 5 ビットの 3 つの MSB が入る）

下図で、選択したフォーマットに応じたピクセルデータのマッピングを説明しています。

表 253. ピクセルデータマッピングとカラーフォーマット

ARGB8888			
@+3 A_x[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 A_{x+1}[7:0]	@+6 R_{x+1}[7:0]	@+5 G_{x+1}[7:0]	@+4 B_{x+1}[7:0]
RGB888			
@+3 B_{x+1}[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 G_{x+2}[7:0]	@+6 B_{x+2}[7:0]	@+5 R_{x+1}[7:0]	@+4 G_{x+1}[7:0]
RGB565			

表 253. ピクセルデータマッピングとカラーフォーマット（続き）

ARGB8888			
@+3 $R_{x+1}[4:0]$ $G_{x+1}[5:3]$	@+2 $G_{x+1}[2:0]$ $B_{x+1}[4:0]$	@+1 $R_x[4:0]$ $G_x[5:3]$	@ $G_x[2:0]$ $B_x[4:0]$
@+7 $R_{x+3}[4:0]$ $G_{x+3}[5:3]$	@+6 $G_{x+3}[2:0]$ $B_{x+3}[4:0]$	@+5 $R_{x+2}[4:0]$ $G_{x+2}[5:3]$	@+4 $G_{x+2}[2:0]$ $B_{x+2}[4:0]$
ARGB1555			
@+3 $A_{x+1}[0]R_{x+1}[4:0]$ $G_{x+1}[4:3]$	@+2 $G_{x+1}[2:0]$ $B_{x+1}[4:0]$	@+1 $A_x[0]$ $R_x[4:0]$ $G_x[4:3]$	@ $G_x[2:0]$ $B_x[4:0]$
@+7 $A_{x+3}[0]R_{x+3}[4:0]$ $G_{x+3}[4:3]$	@+6 $G_{x+3}[2:0]$ $B_{x+3}[4:0]$	@+5 $A_{x+2}[0]R_{x+2}[4:0]G_{x+2}[4:3]$	@+4 $G_{x+2}[2:0]$ $B_{x+2}[4:0]$
ARGB4444			
@+3 $A_{x+1}[3:0]R_{x+1}[3:0]$	@+2 $G_{x+1}[3:0]$ $B_{x+1}[3:0]$	@+1 $A_x[3:0]$ $R_x[3:0]$	@ $G_x[3:0]$ $B_x[3:0]$
@+7 $A_{x+3}[3:0]R_{x+3}[3:0]$	@+6 $G_{x+3}[3:0]$ $B_{x+3}[3:0]$	@+5 $A_{x+2}[3:0]R_{x+2}[3:0]$	@+4 $G_{x+2}[3:0]$ $B_{x+2}[3:0]$
L8			
@+3 $L_{x+3}[7:0]$	@+2 $L_{x+2}[7:0]$	@+1 $L_{x+1}[7:0]$	@ $L_x[7:0]$
@+7 $L_{x+7}[7:0]$	@+6 $L_{x+6}[7:0]$	@+5 $L_{x+5}[7:0]$	@+4 $L_{x+4}[7:0]$
AL44			
@+3 $A_{x+3}[3:0]$ $L_{x+3}[3:0]$	@+2 $A_{x+2}[3:0]$ $L_{x+2}[3:0]$	@+1 $A_{x+1}[3:0]$ $L_{x+1}[3:0]$	@ $A_x[3:0]$ $L_x[3:0]$
@+7 $A_{x+7}[3:0]$ $L_{x+7}[3:0]$	@+6 $A_{x+6}[3:0]$ $L_{x+6}[3:0]$	@+5 $A_{x+5}[3:0]$ $L_{x+5}[3:0]$	@+4 $A_{x+4}[3:0]$ $L_{x+4}[3:0]$
AL88			
@+3 $A_{x+1}[7:0]$	@+2 $L_{x+1}[7:0]$	@+1 $A_x[7:0]$	@ $L_x[7:0]$
@+7 $A_{x+3}[7:0]$	@+6 $L_{x+3}[7:0]$	@+5 $A_{x+2}[7:0]$	@+4 $L_{x+2}[7:0]$

カラールックアップテーブル（CLUT）

CLUT は、**LTDC_LxCR** レジスタですべてのレイヤに対してランタイム時に有効にできますが、L8、AL44、AL88 入力ピクセルフォーマット使用時のインデックスカラーの場合にのみ役立ちます。

まず、CLUT にそのピクセルの本来の R、G、B 値に代わる R、G、B 値をロードする必要があります（インデックスカラー）。各色（RGB 値）には、それ自体のアドレスがあり、CLUT 内の位置を表します。

R、G、B 値とそれぞれのアドレスは **LTDC_LxCLUTWR** レジスタでプログラムされます。

- L8 および AL88 入力ピクセルフォーマットの場合、CLUT は 256 色でロードする必要があります。各色のアドレスは、**LTDC_LxCLUTWR** レジスタ内の CLUTADD ビットで設定されます。
- AL44 入力ピクセルフォーマットの場合、CLUT は 16 色でロードする必要があります。各色のアドレスは、4 ビット L チャネルを次のように 8 ビットに複製することによって埋める必要があります。
  - L0（インデックスカラー 0）、アドレス 0x00
  - L1、アドレス 0x11
  - L2、アドレス 0x22
  - .....
  - L15、アドレス 0xFF

カラーフレームバッファアドレス

各レイヤには、**LTDC_LxCFBAR** レジスタで設定されるカラーフレームバッファの開始アドレスがあります。

レイヤが有効になると、カラーフレームバッファからデータがフェッチされます。

カラーフレームバッファ長

すべてのレイヤには、バイト数単位でのカラーフレームバッファの総ライン長設定とフレームバッファ内のライン数設定があり、それぞれ **LTDC_LxCFBLR** および **LTDC_LxCFBLNR** レジスタで設定可能です。

ライン長とライン数の設定は、フレームバッファの最後でレイヤ FIFO へのデータのプリフェッチを停止するために使用されます。

- 必要なバイト数よりも少なく設定されている場合、FIFO アンダーラン割り込みが事前に有効になっていると、該当する割り込みが生成されます。
- 実際に必要なバイト数よりも多く設定されている場合、FIFO から読み出された不要なデータは破棄されます。不要なデータは表示されません。

カラーフレームバッファピッチ

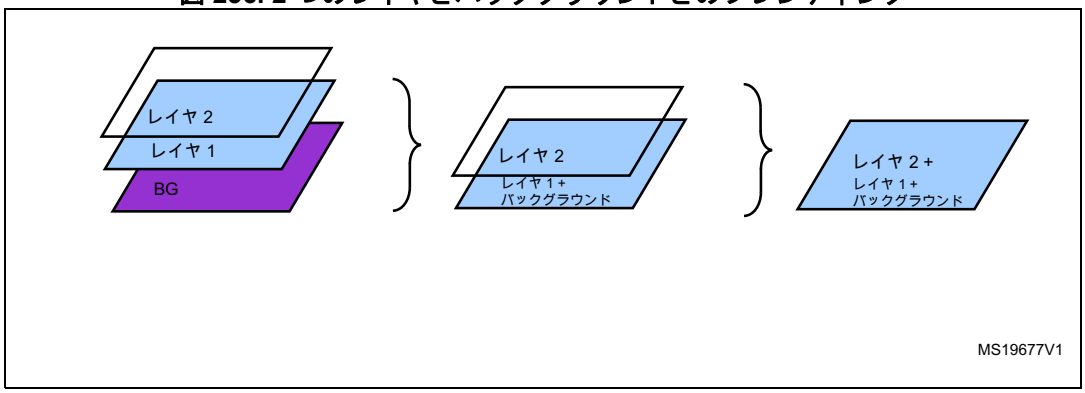
すべてのレイヤには、カラーフレームバッファに対して設定可能なピッチがあります。これはあるラインの開始から次のラインの始まりまでの距離をバイト数で表したものです。ピッチは、**LTDC_LxCFBLR** レジスタで設定します。

レイヤのブレンディング

ブレンディングは常にアクティブとなっており、2 つのレイヤは **LTDC_LxBFCR** レジスタで設定されるブレンディング係数に従ってブレンドすることができます。

混合順序は固定で、下から上に順番に混合されます。2 つのレイヤが有効になっている場合、まずレイヤ 1 がバックグラウンドカラーと混合され、その後レイヤ 2 がレイヤ 1 とバックグラウンドとの混合色と混合されます。*図 236*を参照してください。

図 236. 2 つのレイヤとバックグラウンドとのブレンディング



デフォルト色

すべてのレイヤには、フォーマット ARGB でデフォルト色を設定することができ、このデフォルト色は定義されているレイヤウィンドウ外またはレイヤが無効になっているときに使用されます。

デフォルト色は、**LTDC_LxDCCR** レジスタで設定します。

ブレンディングは、レイヤの 1 つが無効になっていても必ず 2 つのレイヤ間で実施されます。レイヤが無効になっているときにデフォルト色が表示されないようにするには、LTDC_LxBFCR レジスタの該当レイヤのブレンディング係数をリセット値に保持します。

カラーキーイング

カラーキー（RGB）を設定し、透明なピクセルを表すことができます。

カラーキーイングが有効になっていると、現在のピクセル（フォーマット変換後でCLUT個別ブレンディング前）がカラーキーと比較されます。プログラムされた RGB 値と一致した場合、そのピクセルのすべてのチャネル（ARGB）が 0 にセットされます。

カラーキー値を設定することにより、ランタイム時にその値を使用してピクセル RGB 値を置換することができます。

カラーキーイングは、**LTDC_LxCKCR** レジスタで有効にします。

カラーキーイングは、LTDC_LxCKCR レジスタで設定します。プログラムされた値は、ARGB888 へのピクセルフォーマット変換後の現在のピクセルと比較するピクセルフォーマットに依存しています。

例：ミッドイエロー色（赤 50% + 緑 50%）が透明カラーキーとして使用された場合：

- RGB565では、ミッドイエロー色は 0x8400 です。LTDC_LxCKCR を 0x848200 にセットします。
- ARGB8888では、ミッドイエロー色は 0x808000 ですので、LTDC_LxCKCR を 0x808000 に設定します。
- すべての CLUT ベースカラーモード（L8、AL88、AL44）で、パレットエントリの 1 つを0x808000 のミッドイエロー色とし、LTDC_LxCKCR を 0x808000 に設定します。

32.5 LTDC 割り込み

LTDC により、2 つの割り込みベクタの論理和をとった 4 つのマスク可能な割り込みが使用できます。

割り込みソースは、**LTDC_IER** レジスタを介して別々に有効または無効にできます。適切なマスクビットを 1 にセットすることで、対応する割り込みが有効になります。

次のイベントによって 2 つの割り込みが生成されます。

- ライン割り込み：プログラムされたライン到達時に生成されます。ライン割り込み位置は、LTDC_LIPCR レジスタでプログラムします。
- レジスタ再ロード割り込み:垂直ブランキング期間中にシャドウレジスタの再ロードが実施されると生成されます。
- FIFO アンダーラン割り込み：空のレイヤ FIFO からピクセルが要求されると生成されます。
- 転送エラー割り込み：データ転送中に AXI バスエラーが発生すると生成されます。

これらの割り込みイベントは、下図に説明するように NVIC コントローラに接続されています。

図 237. 割り込みイベント

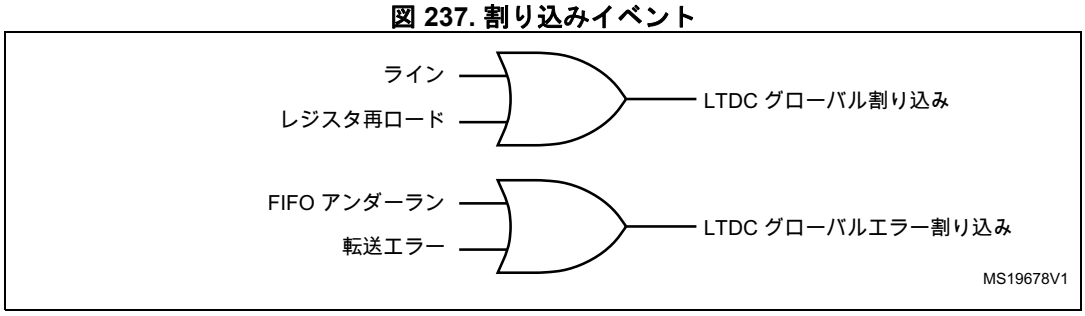


表 254. LTDC 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
ライン	LIF	LIE
レジスタ再ロード	RRIF	RRIEN
FIFO アンダーラン	FUDERRIF	FUDERRIE
転送エラー	TERRIF	TERRIE

32.6 LTDC プログラミング手順

- RCC レジスタで LTDC クロックを有効にします。
- パネルデータシートに従って必要なピクセルクロックを設定します。
- 同期タイミングの設定: *セクション 32.4.1 : LTDC グローバル設定パラメータ*に説明されているように、パネルデータシートに従って VSYNC、HSYNC、垂直／水平バックポーチ、アクティブデータエリア、フロントポーチなどのタイミング。
- **LTDC_GCR** レジスタで、同期信号とクロック極性を設定します。
- 必要であれば、**LTDC_BCCR** レジスタでバックグラウンドカラーを設定します。
- **LTDC_IER** および **LTDC_LIPCR** レジスタで必要な割り込みを設定します。
- 以下をプログラムすることにより、レイヤ 1/2 のパラメータを設定します。
  - **LTDC_LxWHPCR** および **LTDC_WVPCR** レジスタ内のレイヤウィンドウの水平／垂直位置。レイヤウィンドウは、アクティブデータエリア内にある必要があります。
  - **LTDC_LxPFCR** レジスタでピクセル入力フォーマットをプログラム
  - **LTDC_LxCFBAR** レジスタでカラーフレームバッファ開始アドレスをプログラム
  - **LTDC_LxCFBLR** レジスタでカラーフレームバッファのライン長およびピッチをプログラム
  - **LTDC_LxCFBLNR** レジスタでカラーフレームバッファのライン数をプログラム
  - 必要であれば、**LTDC_LxCLUTWR** レジスタで RGB 値とそのアドレスとともに CLUT をロードします。
  - 必要であれば、**LTDC_LxDCCR** および **LTDC_LxBFCR** レジスタでデフォルト色とブレンディング係数をそれぞれ設定します。
- レイヤ 1/2 を有効にし、必要であれば **LTDC_LxCR** レジスタで CLUT を有効にします。
- 必要であれば、**LTDC_GCR および LTDC_LxCKCR** レジスタでディザリングとカラーキーイングをそれぞれ有効にすることができます。また、動作中に有効にすることもできます。
- **LTDC_SRCR** レジスタを介してシャドウレジスタをアクティブレジスタに再ロードします。
- **LTDC_GCR** レジスタで LCD-TFT コントローラを有効にします。
- CLUT を除き、すべてのレイヤパラメータは動作中に変更できます。新しい設定は、直ちに再ロードするか、**LTDC_SRCR** レジスタを設定することにより、垂直ブランキング期間中に再ロードする必要があります。

注： ***すべてのレイヤのレジスタはシャドウイングされます。レジスタに書き込まれたら、再ロードが完了するまでは再変更してはなりません。したがって、同じレジスタへの新しい書き込みは、まだ再ロードされていなければその前の設定に優先します。***

32.7 LTDC レジスタ

32.7.1 LTDC 同期サイズ設定レジスタ（LTDC_SSCR）

このレジスタによって、水平同期ピクセル数から 1 を引いた数と、垂直同期ライン数から 1 を引いた数を指定します。設定例については*図 234* および*セクション 32.4 : LTDC のプログラム可能なパラメータ*を参照してください。

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HSW[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	VSH[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **HSW[11:0]**：水平同期幅（ピクセルクロック周期の単位）

このビットによって、水平同期ピクセル数から 1 を引いた数を指定します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **VSH[10:0]**：垂直同期高さ（水平スキャンラインの単位）

このビットによって、垂直同期の高さから 1 を引いた数を指定します。これは、水平同期ライン数を表します。

32.7.2 LTDC バックポーチ設定レジスタ（LTDC_BPCR）

このレジスタによって、水平同期およびバックポーチの累積ピクセル数から 1 を引いた数（**HSYNC 幅 + HBP- 1**）と垂直同期およびバックポーチの累積ライン数から 1 を引いた数（**VSYNC 高さ + VBP-1**）を指定します。設定例については*図 234* および*セクション 32.4 : LTDC のプログラム可能なパラメータ*を参照してください。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AHBP[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	AVBP[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **AHBP[11:0]**：累積水平バックポーチ（ピクセルクロック周期の単位）

このビットにより、水平同期および水平バックポーチピクセル数から 1 を引いた数から成る累積水平バックポーチ幅を指定します。

水平バックポーチは、水平同期がインアクティブになってから次のスキャンラインのアクティブ表示エリアの開始までの期間です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **AVBP[10:0]**：累積垂直バックポーチ（水平スキャンラインの単位）

このビットにより、垂直同期および垂直バックポーチライン数から 1 を引いた数から成る累積垂直バックポーチ幅を指定します。

垂直バックポーチは、フレーム開始時から次のフレームの最初のアクティブなスキャンライン開始までの水平スキャンライン数です。

32.7.3 LTDC アクティブ幅設定レジスタ（LTDC_AWCR）

このレジスタによって、水平同期、バックポーチ、アクティブピクセルの累積数から 1 を引いた数（**HSYNC 幅 + HBP + アクティブ幅 - 1**）と垂直同期、バックポーチライン、アクティブラインの累積数から 1 を引いた数（**VSYNC 高さ + BVBP + アクティブ高さ - 1**）を指定します。設定例については *図 234* および *セクション 32.4 : LTDC のプログラム可能なパラメータ*を参照してください。

アドレスオフセット：0x10

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **AAW[11:0]**：累積アクティブ幅（ピクセルクロック周期の単位）

このビットにより、水平同期、水平バックポーチ、アクティブピクセル数から 1 を引いた数から成る累積アクティブ幅を定義します。

アクティブ幅は、パネルスキャンラインのアクティブ表示エリアにおけるピクセル数です。

最大ピクセルクロックに従った対応最大アクティブ幅については、デバイスデータシートを参照してください。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **AAH[10:0]**：累積アクティブ高さ（水平スキャンラインの単位）

このビットにより、垂直同期、垂直バックポーチ、アクティブ高さのライン数から 1 を引いた数から成る累積高さを指定します。アクティブ高さは、パネル内のアクティブなラインの数です。

最大ピクセルクロックに従った対応最大アクティブ高さについては、デバイスデータシートを参照してください。

32.7.4 LTDC 全幅設定レジスタ（LTDC_TWCR）

このレジスタによって、水平同期、バックポーチ、アクティブ、フロントポーチピクセルの累積数から 1 を引いた数（**HSYNC 幅 + HBP + アクティブ幅 + HFP - 1**）と垂直同期、バックポーチライン、アクティブ、フロントラインの累積数から 1 を引いた数（**VSYNC 高さ + BVBP + アクティブ高さ + VFP- 1**）を指定します。設定例については*図 234* および*セクション 32.4 : LTDC のプログラム可能なパラメータ*を参照してください。

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TOTALW[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
16	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TOTALH[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TOTALW[11:0]** : 全幅（ピクセルクロック周期の単位）
このビットにより、水平同期、水平バックポーチ、アクティブ幅、水平フロントポーチピクセル数から 1 を引いた数から成る累積全幅を指定します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **TOTALH[10:0]** : 全高（水平スキャンラインの単位）
このビットにより、垂直同期、垂直バックポーチ、アクティブ高さ、垂直フロントポーチ高さのライン数から 1 を引いた数から成る累積高さを指定します。

32.7.5 LTDC グローバル制御レジスタ（LTDC_GCR）

このレジスタにより、LCD-TFT コントローラのグローバル設定を指定します。

アドレスオフセット：0x18

リセット値：0x0000 2220

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN
rw	rw	rw	rw												rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	r	r	r		r	r	r		r	r	r				rw

ビット 31 **HSPOL**：水平同期極性
このビットは、ソフトウェアによってセット／クリアされます。
0：水平同期極性アクティブロー
1：水平同期極性アクティハイ

ビット 30 **VSPOL**：垂直同期極性
このビットは、ソフトウェアによってセット／クリアされます。
0：垂直同期アクティブロー
1：垂直同期アクティブハイ

ビット 29 **DEPOL**：非データイネーブル極性
このビットは、ソフトウェアによってセット／クリアされます。
0：非データイネーブル極性アクティブロー
1：非データイネーブル極性アクティブハイ

ビット 28 **PCPOL**：ピクセルクロック極性
このビットは、ソフトウェアによってセット／クリアされます。
0：ピクセルクロック極性アクティブロー
1：ピクセルクロック極性アクティブハイ

ビット 27:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **DEN**：ディザリングイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ディザリングディセーブル
1：ディザリングイネーブル

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **DRW[2:0]**：ディザ赤の幅
このビットはディザ赤のビットを返します。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DGW[2:0]**：ディザ緑の幅
このビットはディザ緑のビットを返します。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **DBW[2:0]**：ディザ青の幅
このビットはディザ青のビットを返します。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **LTDCEN**：LCD-TFT コントローライネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0：LTDC ディセーブル
1：LTDC イネーブル

32.7.6 LTDC シャドウ再ロード設定レジスタ（LTDC_SRCR）

このレジスタにより、シャドウレジスタ値をアクティブレジスタに直ちに再ロードすることもできれば、垂直ブランキング期間中にすることもできます。LTDC_L1CLUTWR および LTDC_L2CLUTWR を除き、レイヤ 1 とレイヤ 2 のすべてのレジスタはシャドウレジスタです。

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **VBR**：垂直ブランキングの再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます。（いったんセットされるとレジスタの書き込みによってクリアすることはできません）

0：影響なし。

1：シャドウレジスタは、垂直ブランキング期間中（アクティブ表示エリア後の最初のライン開始時）に再ロードされます。

ビット 0 **IMR**：即時再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます。

0：影響なし。

1：シャドウレジスタが直ちに再ロードされます。

注： ***シャドウレジスタは、アクティブな値を読み出します。再ロードが完了するまでは、「古い」値が読み出されます。***

32.7.7 LTDC バックグラウンドカラー設定レジスタ（LTDC_BCCR）

このレジスタにより、バックグラウンドカラー（RGB888）を指定します。

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCGREEN[7:0]								BCBLUE[7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **BCRED[7:0]**：バックグラウンドカラー、赤の値
このビットにより、バックグラウンドの赤の値を設定します。

ビット 15:8 **BCGREEN[7:0]**：バックグラウンドカラー、緑の値
このビットにより、バックグラウンドの緑の値を設定します。

ビット 7:0 **BCBLUE[7:0]**：バックグラウンドカラー、青の値
このビットにより、バックグラウンドの青の値を設定します。

32.7.8 LTDC 割り込み有効レジスタ（LTDC_IER）

このレジスタは、対応するビットを 1 にセットすることで、どのステータスフラグが割り込み要求を生成するかを決定します。

アドレスオフセット：0x34

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RRIE**：レジスタ再ロード割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：レジスタ再ロード割り込みディセーブル
1：レジスタ再ロード割り込みイネーブル

ビット 2 **TERRIE**：転送エラー割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：転送エラー割り込みディセーブル
1：転送エラー割り込みイネーブル

ビット 1 **FUIE**：FIFO アンダーラン割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFO アンダーラン割り込みディセーブル
1：FIFO アンダーラン割り込みイネーブル

ビット 0 **LIE**：ライン割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ライン割り込みディセーブル
1：ライン割り込みイネーブル

32.7.9 LTDC 割り込みステータスレジスタ（LTDC_ISR）

このレジスタは、割り込みステータスフラグを返します。

アドレスオフセット：0x38

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RRIF**：レジスタ再ロード割り込みフラグ

0：レジスタ再ロード割り込みは生成されません。

1：垂直ブランキング再ロード発生時に（およびアクティブエリア到達後の最初のラインで）レジスタ再ロード割り込みが生成されます。

ビット 2 **TERRIF**：転送エラー割り込みフラグ

0：転送エラー割り込みは生成されません。

1：バスエラー発生時に転送エラー割り込みが生成されます。

ビット 1 **FUIF**：FIFO アンダーラン割り込みフラグ

0：FIFO アンダーラン割り込みは生成されません。

1：レイヤ FIFO のいずれかが空で、FIFO からピクセルデータが読み出された場合、FIFO アンダーラン割り込みが生成されます。

ビット 0 **LIF**：ライン割り込みフラグ

0：ライン割り込みは生成されません。

1：プログラムされたライン到達時にライン割り込みが生成されます。

32.7.10 LTDC 割り込みクリアレジスタ（LTDC_ICR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRRIF**：レジスタ再ロード割り込みフラグをクリアします。
0：影響なし。
1：LTDC_ISR レジスタの RRIF フラグをクリアします。

ビット 2 **CTERRIF**：転送エラー割り込みフラグをクリアします。
0：影響なし。
1：LTDC_ISR レジスタの TERRIF フラグをクリアします。

ビット 1 **CFUIF**：FIFO アンダーラン割り込みフラグをクリアします。
0：影響なし。
1：LTDC_ISR レジスタの FUDERRIF フラグをクリアします。

ビット 0 **CLIF**：ライン割り込みフラグをクリアします。
0：影響なし。
1：LTDC_ISR レジスタの LIF フラグをクリアします。

32.7.11 LTDC ライン割り込み位置設定レジスタ（LTDC_LIPCR）

このレジスタにより、ライン割り込みの位置を指定します。プログラムされるライン値は、タイミングパラメータに依存します。*図 234*を参照してください。

アドレスオフセット：0x40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	LIPOS[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **LIPOS[10:0]**：ライン割り込み位置
このビットにより、ライン割り込み位置を設定します。

32.7.12 LTDC 現在位置ステータスレジスタ（LTDC_CPSR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CXPOS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CYPOS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **CXPOS[15:0]**：現在の X 位置
このビットは、現在の X 位置を返します。

ビット 15:0 **CYPOS[15:0]**：現在の Y 位置
このビットは、現在の Y 位置を返します。

32.7.13 LTDC 現在表示ステータスレジスタ（LTDC_CDSR）

このレジスタは、HSYNC、VSYNC、水平／垂直 DE 信号が制御する現在の表示フェーズのステータスを返します。

例 : 現在の表示フェーズが垂直同期であれば、VSYNCS ビットがセットされます（アクティブハイ）。現在の表示フェーズが水平同期であれば、HSYNCS ビットがアクティブハイになります。

アドレスオフセット：0x48

リセット値 : 0x0000 000F

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HSYNCS**：水平同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 2 **VSYNCS**：垂直同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 1 **HDES**：水平データ有効表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 0 **VDES**：垂直データ有効表示ステータス
0：アクティブロー
1：アクティブハイ

注 : 返されるステータスは、LTDC_GCR レジスタで設定されている極性には依存しません。代わりに、現在のアクティブ表示フェーズを返します。

32.7.14 LTDC レイヤ x 制御レジスタ（LTDC_LxCR）（x=1..2）

アドレスオフセット：0x84 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CLUTEN**：カラールックアップテーブルイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：カラールックアップテーブル無効
1：カラールックアップテーブル有効
CLUT は、L8、AL44、AL88 ピクセルフォーマットの場合にのみ意味を持ちます。*1138 ページのカラールックアップテーブル（CLUT）*を参照してください。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **COLKEN**：カラーキーイングイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：カラーキーイング無効
1：カラーキーイング有効

ビット 0 **LEN**：レイヤイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：レイヤ無効
1：レイヤ有効

32.7.15 LTDC レイヤ x ウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）（x=1..2）

このレジスタにより、レイヤ 1 または 2 ウィンドウの水平位置（最初と最後のピクセル）を指定します。

ラインの最初の見えるピクセルは、**LTDC_BPCR** レジスタにおける ***AHBP[10:0] ビット + 1*** のプログラム値です。

ラインの最後の見えるピクセルは、**LTDC_AWCR** レジスタにおける ***AAW[10:0] ビット***のプログラム値です。

アドレスオフセット：0x88 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **WHSPPOS[11:0]**：ウィンドウ水平停止位置
このビットにより、レイヤウィンドウにおけるラインの最後の見えるピクセルを設定します。
WHSPPOS[11:0] は ***AHBP[10:0] ビット + 1*** 以上である必要があります（LTDC_BPCR レジスタにプログラム）。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WHSTPOS[11:0]**：ウィンドウ水平開始位置
このビットにより、レイヤウィンドウにおけるラインの最初の見えるピクセルを設定します。
WHSTPOS[11:0] は ***AAW[10:0] ビット*** 以下である必要があります（LTDC_AWCR レジスタにプログラム）。

例：

LTDC_BPCR レジスタは 0x000E0005（AHBP[11:0] は 0xE）に設定され、LTDC_AWCR レジスタは 0x028E01E5（AAW[11:0] は 0x28E）に設定されます。アクティブデータエリアで水平開始オフセットを 5 ピクセルにして、サイズ 630x460 のウィンドウにおける水平位置を設定するには、

1. レイヤウィンドウの最初のピクセル：WHSTPOS[11:0] を 0x14 （0xE+1+0x5）にプログラムします。
2. レイヤウィンドウの最後のピクセル：WHSPPOS[11:0] を 0x28A にプログラムします。

32.7.16 LTDC レイヤ x ウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）（x=1..2）

このレジスタにより、レイヤ 1 または 2 ウィンドウの垂直位置（最初と最後のライン）を指定します。

フレームの最初の見えるラインは、**LTDC_BPCR** レジスタにおける ***AVBP[10:0] ビット + 1*** プログラム値です。

フレームの最後の見えるラインは、**LTDC_AWCR** レジスタにおける ***AAH[10:0] ビット***のプログラム値です。

アドレスオフセット：0x8C + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:16 **WVSPPOS[10:0]**：ウィンドウ垂直停止位置
このビットにより、レイヤウィンドウの最後の見えるラインを設定します。
WVSPPOS[10:0] は ***AVBP[10:0] ビット + 1*** 以上である必要があります（LTDC_BPCR レジスタにプログラム）。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **WVSTPOS[10:0]**：ウィンドウ垂直開始位置
このビットにより、レイヤウィンドウの最初の見えるラインを設定します。
WVSTPOS[10:0] は ***AAH[10:0] ビット*** 以下である必要があります（LTDC_AWCR レジスタにプログラム）。

例：

LTDC_BPCR レジスタは 0x000E0005（AVBP[10:0] は 0x5）に設定され、LTDC_AWCR レジスタは 0x028E01E5（AAH[10:0] は 0x1E5）に設定されます。アクティブデータエリアで垂直開始オフセットを 8 ラインにして、サイズ 630x460 のウィンドウにおける垂直位置を設定するには、

1. レイヤウィンドウの最初のライン：WVSTPOS[10:0] を 0xE（0x5 + 1 + 0x8）にプログラムします。
2. レイヤウィンドウの最後のライン：WVSPPOS[10:0] を 0x1DA にプログラムします。

32.7.17 LTDC レイヤ x カラーキーイング設定レジスタ（LTDC_LxCKCR）（x=1..2）

このレジスタにより、カラーキーイングによって使用されるカラーキー値（RGB）を指定します。

アドレスオフセット：0x90 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKGREEN[7:0]								CKBLUE[7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CKRED[7:0]**：カラーキー赤の値

ビット 15:8 **CKGREEN[7:0]**：カラーキー緑の値

ビット 7:0 **CKBLUE[7:0]**：カラーキー青の値

32.7.18 LTDC レイヤ x ピクセルフォーマット設定レジスタ（LTDC_LxPFCR）（x=1..2）

このレジスタにより、レイヤのフレームバッファ内に保存されるデータに使用するピクセルフォーマットを指定します。ピクセルデータはフレームバッファから読み出され、内部フォーマット 8888(ARGB) に変換されます。

アドレスオフセット：0x94 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PF[2:0]**：ピクセルフォーマット

このビットにより、ピクセルフォーマットを設定します。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

101：L8（8 ビット輝度）

110：AL44（4 ビットアルファ、4 ビット輝度）

111：AL88（8 ビットアルファ、8 ビット輝度）

32.7.19 LTDC レイヤ x 定数アルファ設定レジスタ（LTDC_LxCACR）（x=1..2）

このレジスタにより、アルファブレンディングで使用する定数アルファ値（ハードウェアによって 255 で除される）を指定します。LTDC_LxBFCR レジスタを参照してください。

アドレスオフセット：0x98 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：（レイヤ x -1）0x0000 00FF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CONSTA[7:0]**：定数アルファ

このビットにより、ブレンディングに使用する定数アルファを設定します。定数アルファは、ハードウェアによって 255 で除されます。

例：プログラムされている定数アルファが 0xFF であれば、定数アルファ値は 255/255=1 となります。

32.7.20 LTDC レイヤ x デフォルト色設定レジスタ（LTDC_LxDCCR）（x=1..2）

このレジスタにより、フォーマット ARGB におけるレイヤのデフォルト色を指定します。このデフォルト色は、指定されたレイヤウィンドウ外や、レイヤが無効になっているときに使用されます。リセット値 0x00000000 は、透明な黒を表します。

アドレスオフセット：0x9C + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DCALPHA[7:0]								DCRED[7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCGREEN[7:0]								DCBLUE[7:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DCALPHA[7:0]**：デフォルト色アルファ

このビットにより、デフォルトのアルファ値を設定します。

ビット 23:16 **DCRED[7:0]**：デフォルト色赤

このビットにより、デフォルト赤の値を設定します。

ビット 15:8 **DCGREEN[7:0]**：デフォルト色緑

このビットにより、デフォルト緑の値を設定します。

ビット 7:0 **DCBLUE[7:0]**：デフォルト色青

このビットにより、デフォルト青の値を設定します。

32.7.21 LTDC レイヤ x ブレンディング係数設定レジスタ（LTDC_LxBFCR）（x=1..2）

このレジスタにより、ブレンディング係数 F1 および F2 を指定します。

基本的なブレンディングの計算式：BC = BF1 x C + BF2 x Cs

- BC = ブレンドされた色
- BF1 = ブレンディング係数 1
- C = 現在のレイヤの色
- BF2 = ブレンディング係数 2
- Cs = ブレンドされた下側レイヤの色

アドレスオフセット：0xA0 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0607

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2[2:0]		
					rw	rw	rw						rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **BF1[2:0]**：ブレンディング係数 1
このビットにより、ブレンディング係数 F1 を選択します。
000：予約済み
001：予約済み
010：予約済み
011：予約済み
100：定数アルファ
101：予約済み
110：ピクセルアルファ x 定数アルファ
111：予約済み

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **BF2[2:0]**：ブレンディング係数 2
このビットにより、ブレンディング係数 F2 を選択します。
000：予約済み
001：予約済み
010：予約済み
011：予約済み
100：予約済み
101：1 - 定数アルファ
110：予約済み
111：1 -（ピクセルアルファ x 定数アルファ）

注： ***定数アルファ値は、LxCACR レジスタのプログラム値をハードウェアにより 255 で除したものです。***

例：レイヤ 1 のみ有効、BF1 を定数アルファに設定

BF2 を“1 - 定数アルファ ”に設定

定数アルファ：LxCACR レジスタでプログラムされている定数アルファは 240 （0xF0）です。従って、定数アルファ値は 240/255 = 0.94 となります。

C：現在のレイヤ色は 128

Cs：バックグラウンドカラーは 48

レイヤ 1 はバックグラウンドカラーとブレンドされます。

BC = 定数アルファ x C +（1 - 定数アルファ）x Cs = 0.94 x 128 +（1- 0.94）x 48 = 123

32.7.22 LTDC レイヤ x カラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）（x=1..2）

このレジスタにより、カラーフレームバッファ開始アドレスを指定します。このアドレスは、フレームバッファでレイヤの左上ピクセルのピクセルデータが保存されているアドレスを示している必要があります。

アドレスオフセット：0xAC + 0x80 x（*レイヤ x* -1）、*レイヤ x* = 1 または 2

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CFBADD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFBADD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CFBADD[31:0]**：カラーフレームバッファ開始アドレス
このビットにより、カラーフレームバッファ開始アドレスを指定します。

32.7.23 LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）（x=1..2）

このレジスタにより、カラーフレームバッファのライン長とピッチを指定します。

アドレスオフセット：0xB0 + 0x80 x（*レイヤ x* -1）、*レイヤ x* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:29 予約済みであり、リセット値のままにしておかなければなりません。

ビット 28:16 **CFBP[12:0]**：カラーフレームバッファピッチ（バイト単位）
このビットにより、あるピクセルのラインの開始から次のラインの開始までの増分であるピッチをバイト単位で指定します。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **CFBLL[12:0]**：カラーフレームバッファのライン長
このビットにより、バイト単位のピクセルライン長 + 7 を指定します。
ライン長は次のようにして計算します。アクティブハイの幅 x ピクセル当たりのバイト数 + 7

例：

- フォーマット RGB565（ピクセル当たり 2 バイト）、幅 256 ピクセル（ライン当たりの総バイト数は 256x2=512 バイト）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x02000207 という値を書き込みます。
- フォーマット RGB888（ピクセル当たり 3 バイト）、幅 320 ピクセル（ライン当たりの総バイト数は 320x3=960）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x03C003C7 という値を書き込みます。

32.7.24 LTDC レイヤ x カラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）（x=1..2）

このレジスタにより、カラーフレームバッファ内のライン数を指定します。

アドレスオフセット：0xB4 + 0x80 x（*レイヤ x* -1）、*レイヤ x* = 1 または 2

リセット値：0x0000 0000

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

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **CFBLNBR[10:0]**：フレームバッファライン数
このビットにより、アクティブハイ幅に相当するフレームバッファ内ライン数を指定します。

注： ***このライン数とライン長の設定により、各レイヤに対してフレームごとのデータフェッチ数を指定します。必要なバイト数よりも少なく設定されている場合、FIFO アンダーラン割り込みが有効になっていれば、該当する割り込みが生成されます。***

一方、開始アドレスとピッチ設定により、メモリ内の各ラインの正しい開始位置を指定します。

32.7.25 LTDC レイヤ x CLUT 書き込みレジスタ（LTDC_LxCLUTWR）（x=1..2）

このレジスタにより、CLUT アドレスと RGB 値を指定します。

アドレスオフセット：0xC4 + 0x80 x（*レイヤx* -1）、*レイヤx* = 1 または 2

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLUTADD[7:0]								RED[7:0]							
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:24 **CLUTADD[7:0]**：CLUT アドレス
このビットにより、各 RGB 値の CLUT アドレス（CLUT 内のカラー位置）を指定します。

ビット 23:16 **RED[7:0]**：赤の値
このビットにより、赤の値を設定します。

ビット 15:8 **GREEN[7:0]**：緑の値
このビットにより、緑の値を設定します。

ビット 7:0 **BLUE[7:0]**：青の値
このビットにより、青の値を設定します。

注： ***CLUT 書き込みレジスタは、ブランキング期間中またはレイヤが無効になっている場合にのみ設定してください。CLUT は、LTDC_LxCR レジスタで有効または無効にできます。***

CLUT は、L8、AL44、AL88 ピクセルフォーマットの場合にのみ意味を持ちます。

32.7.26 LTDC レジスタマップ

次の表は LTDC レジスタの一覧です。LTDC レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 255. LTDC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0008	LTDC_SSCR	Res.	Res.	Res.	Res.	HSW[11:0]												Res.	Res.	Res.	Res.	Res.	VSH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x000C	LTDC_BPCR	Res.	Res.	Res.	Res.	AHBP[11:0]												Res.	Res.	Res.	Res.	Res.	AVBP[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0010	LTDC_AWCR	Res.	Res.	Res.	Res.	AAV[11:0]												Res.	Res.	Res.	Res.	Res.	AAH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0014	LTDC_TWCR	Res.	Res.	Res.	Res.	TOTALW[11:0]												Res.	Res.	Res.	Res.	Res.	TOTALH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0018	LTDC_GCR	HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN	Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	リセット値	0	0	0	0												0		0	1	0		0	1	0		0	1	0				0
0x0024	LTDC_SRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBR	IMR
	リセット値																															0	0
0x002C	LTDC_BCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]								BCGREEN[7:0]					BCBLUE[7:0]										
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	LTDC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIE	TERRIE	FUIE	LIE
	リセット値																													0	0	0	0
0x0038	LTDC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIF	TERRIF	FUIF	LIF
	リセット値																													0	0	0	0
0x003C	LTDC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRRIF	CTERRIF	CFUIF	CLIF
	リセット値																													0	0	0	0

参考資料

表 255. LTDC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0040	LTDC_LIPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x0044	LTDC_CPSR	CXPOS[15:0]																CYPOS[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	LTDC_CDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSYNCS	VSYNCS	HDES	VDES
	リセット値																													1	1	1	1
0x0084	LTDC_L1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x0088	LTDC_L1WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x008C	LTDC_L1WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0090	LTDC_L1CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094	LTDC_L1PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x0098	LTDC_L1CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x009C	LTDC_L1DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A0	LTDC_L1BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2[2:0]		
	リセット値																						1	1	0						1	1	1
0x00AC	LTDC_L1CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	LTDC_L1CFBLR	Res.	Res.	Res.	CFBP[17:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	LTDC_L1CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x00C4	LTDC_L1CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 255. LTDC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0104	LTDC_L2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x0108	LTDC_L2WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x010C	LTDC_L2WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0110	LTDC_L2CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114	LTDC_L2PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x0118	LTDC_L2CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x011C	LTDC_L2DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0120	LTDC_L2BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2[2:0]		
	リセット値																						1	1	0						1	1	1
0x012C	LTDC_L2CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0130	LTDC_L2CFBLR	Res.	Res.	Res.	CFBP[12:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134	LTDC_L2CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x0144	LTDC_L2CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

33 JPEG コーデック（JPEG）

33.1 概要

ハードウェア 8 ビット JPEG コーデックでは、非圧縮画像データストリームのエンコードまたはJPEG圧縮された画像データストリームのデコードを行います。JPEG ヘッダの完全な管理も行います。

33.2 JPEG コーデックの主な機能

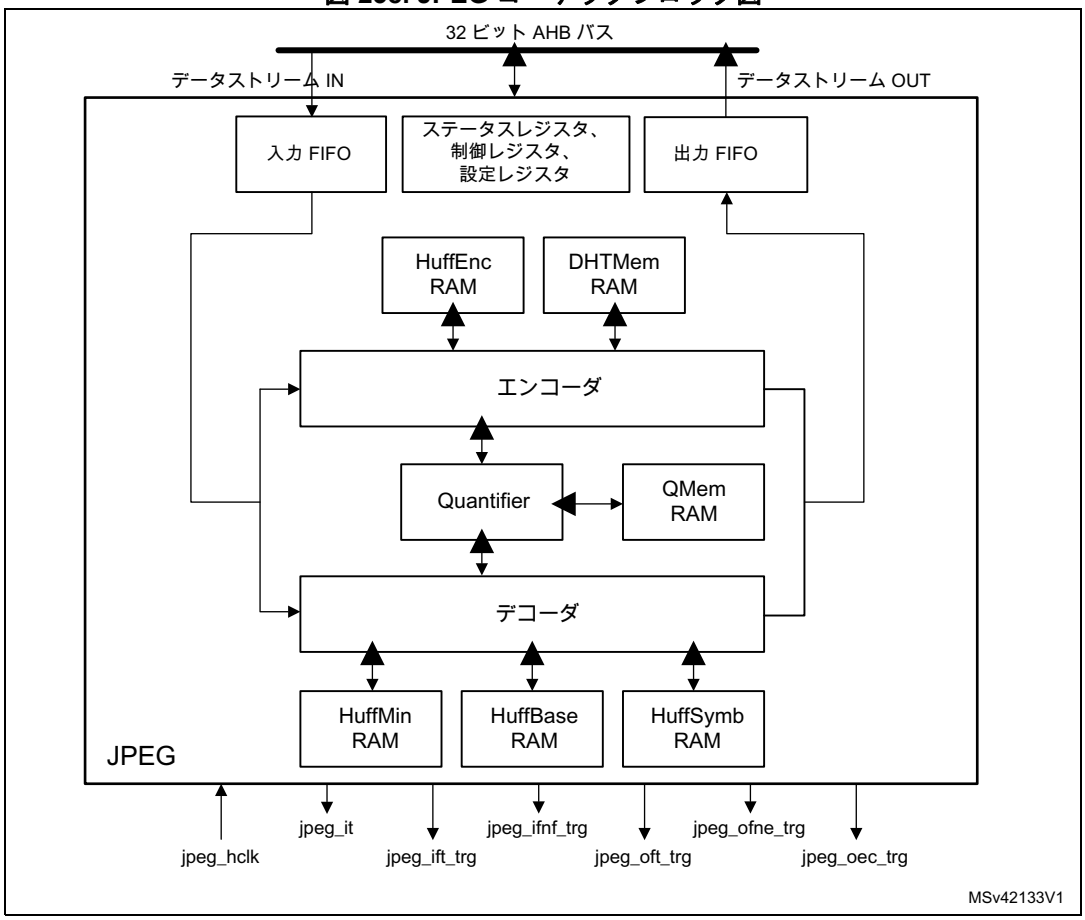
- ハイスピード完全同期動作
- エンコーダまたはデコーダとして設定可能
- ピクセルあたりシングルクロックのエンコード／デコード
- RGB、YCbCr、YCMK、BW（グレイスケール）画像色空間対応
- エンコード／デコード時の画像成分あたり 8 ビット深さ
- イネーブル／ディスエーブル付きの JPEG ヘッダ生成器／パーサ
- 4 つのプログラム可能な量子化テーブル
- シングルクロックのハフマンコーディング／デコーディング
- 完全プログラム可能なハフマンテーブル（2 つの AC と 2 つの DC）
- 完全プログラム可能な最小符号化単位（MCU）
- 同時入出力データストリームインタフェース

33.3 JPEG コーデックブロック の機能説明

33.3.1 概要

JPEG コーデックのブロック図を*図 238 : JPEG コーデックブロック図*に示します。

図 238. JPEG コーデックブロック図



33.3.2 JPEG 内部信号

表 256 に JPEG 内部信号を示します。

表 256. JPEG 内部信号

信号名	信号タイプ	説明
jpeg_hclk	デジタル入力	JPEG カーネル／レジスタインタフェースクロック
jpeg_it	デジタル出力	JPEG グローバル割り込み
jpeg_ift_trg	デジタル出力	MDMA に対する JPEG 入力 FIFO 閾値
jpeg_ifnf_trg	デジタル出力	MDMA に対する JPEG 入力 FIFO ノットフル
jpeg_oft_trg	デジタル出力	MDMA に対する JPEG 出力 FIFO 閾値

表 256. JPEG 内部信号（続き）

信号名	信号タイプ	説明
jpeg_ofne_trg	デジタル出力	MDMA に対する JPEG 出力 FIFO ノットエンプティ
jpeg_oec_trg	デジタル出力	MDMA に対する JPEG 変換終了

33.3.3 JPEG デコード手順

JPEG コーデックは、*ISO/IEC 10918-1* 規格で定義されている JPEG ストリームをデコード可能です。

オプションとして、JPEG ヘッダのパースと、それに応じて JPEG コーデックレジスタと量子化テーブルとハフマンテーブルの更新を行えます。

JPEG コーデックは、JPEG_CONFR1 レジスタの DE ビット（デコードイネーブル）をセットすることにより、デコードモードに設定します。

JPEG_CONFR0 レジスタの START ビットをセットすることにより JPEG デコードを開始します。

JPEG コーデックは、次のいずれかを生成することにより、その入力 FIFO にデータを要求します。

- MDMA トリガ
- 割り込み

入力 FIFO に対する割り込みまたは MDMA トリガの生成

入力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割り込みまたは MDMA トリガを用いて管理できます。

- 入力 FIFO ノットフルフラグ：32 ビット値を書込み可能
- 入力 FIFO 閾値フラグ：8 ワード（32 バイト）を書き込み可能

割り込みまたは MDMA トリガの生成は、JPEG_CONFR0 レジスタの START ビットには依存していません。入力 FIFO フラグは、JPEG コーデックカーネルの状態とは無関係に生成されます。

入力 FIFO がフルであると、書き込みは無視されます。

デコード処理の最後には、余分なバイトが入力 FIFO に残っていたり、割り込みリクエスト／MDMA トリガが未処理となっていることがあります。JPEG_CR レジスタの IFF ビット（入力 FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 入力 FIFO の割り込みを無効として、FIFO一掃時の不要な割り込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

デコード処理の最後に FIFO を一掃しないと、その結果として、残りデータが次回の JPEG デコードに使われてしまいます。

ヘッダの解析

ヘッダの解析は、JPEG_CONFR1 レジスタの HDR ビットをセットすることによってアクティブ化できます。

JPEG ヘッダパーサは、*ISO/IEC 10918-1 規格附属書B*に示されているJPEG ベースラインアルゴリズムに該当するすべてのマーカに対応しています。

対応マーカの解析時には、必要なパラメータをJPEG ヘッダパーサが抽出して、シャドウレジスタに格納します。解析の最後に、JPEG コーデックレジスタは更新されます。

DQT マーカセグメントが位置している場合には、それに関連する量子化データが量子化テーブルメモリに書き込まれます。

DHT マーカセグメントが位置している場合には、それに関連するハフマンテーブルデータが 3 種類の異なるテーブル形式（HuffMin、HuffBase、HuffSymb）へと変換されて、それぞれのメモリに格納されます。

解析動作が完了すると、JPEG_SR レジスタの HPDF （ヘッダ解析完了フラグ）ビットがセットされます。JPEG_CR レジスタの EHPIE （ヘッダ解析完了割り込みイネーブル）ビットがセットされていると、割り込みが生成されます。

JPEG デコード

JPEG ヘッダの解析が行われるか、JPEG コーデックのレジスタとメモリが正常にプログラムされるかすると、受信データストリームはデコードされ、結果として得られた MCU は出力 FIFO に送られます。

画像 2 個を連続デコードする場合には、JPEG_CONFR0 レジスタの START ビットは、2 番目の画像のヘッダ処理が干渉したあとに、（すでに 1 であったとしても）再びセットする必要があります。

出力 FIFO に対する割り込みまたは MDMA トリガの生成

出力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割り込みまたは MDMA トリガを用いて管理できます。

- 出力 FIFO ノットエンプティフラグ：32 ビット値を読み出し可能
- 出力 FIFO 閾値フラグ：8 ワード（32 バイト）を読み出し可能

出力 FIFO がエンプティのときには、読むと 0 が返ります。

JPEG_CONFR0 レジスタの START ビットをリセットすることにより、JPEG コーデック処理が異常終了した場合には、出力 FIFO を一掃できます。FIFO を一掃する必要がある場合には、JPEG_CR レジスタの FF ビット（FIFO 一掃）を設定することにより、ソフトウェアで行ってください。

FIFO を一掃する前には、次を行ってください。

- 出力 FIFO の割り込みを無効として、FIFO一掃時の不要な割り込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

JPEG 設定が何か変化する前に、処理の最後で出力 FIFO を一掃する必要があります。

33.3.4 JPEG エンコード手順

JPEG コーデックは、*ISO/IEC 10918-1* 規格で定義されている JPEG ストリームをデエンコード可能です。

オプションで、JPEG ヘッダの生成も行います。

JPEG コーデックは、JPEG_CONFR1 レジスタの DE ビット（デコードイネーブル）をリセットすることにより、エンコードモードに設定します。

JPEGのエンコードに使用された設定は、JPEG コーデックにロードされる必要があります。

- JPEG コーデック設定レジスタ
- 量子化テーブル
- ハフマンテーブル

JPEG_CONFR0 レジスタの START ビットをセットすることにより JPEG コーデックを開始します。

いったんJPEG コーデックが開始されると、次のいずれかを生成する入力 FIFO のデータを要求します。

- MDMA トリガ
- 割り込み

入力 FIFO に対する割り込みまたは MDMA トリガの生成

入力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割り込みまたは MDMA トリガを用いて管理できます。

- 入力 FIFO ノットフルフラグ：32 ビット値を書込み可能
- 入力 FIFO 閾値フラグ：8 ワード（32 バイト）を書き込み可能

割り込みまたは MDMA トリガの生成は、JPEG_CONFR0 レジスタの START ビットには依存していません。入力 FIFO フラグは、JPEG コーデックカーネルの状態とは無関係に生成されます。

入力 FIFO がフルであると、書き込みは無視されます。

エンコード処理の最後には、余分なバイトが入力 FIFO に残っていたり、割り込みリクエスト／MDMA トリガが未処理となっていることがあります。JPEG_CR レジスタの IFF ビット（入力 FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 入力 FIFO の割り込みを無効として、FIFO一掃時の不要な割り込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

エンコード処理の最後に FIFO を一掃しないと、その結果として、残りデータが次回の JPEG エンコードに使われてしまいます。

JPEG エンコード

JPEG ヘッダが生成されると、受信 MCU はエンコードされて、結果として得られたデータストリームは出力 FIFO に送られます。

出力 FIFO に対する割り込みまたは MDMA トリガの生成

出力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割り込みまたは MDMA トリガを用いて管理できます。

- 出力 FIFO ノットエンプティフラグ：32 ビット値を読み出し可能
- 出力 FIFO 閾値フラグ：8 ワード（32 バイト）を読み出し可能

出力 FIFO がエンプティのときには、読むと 0 が返ります。

JPEG_CONFR0 レジスタの START ビットをリセットすることにより、JPEG コーデック処理が異常終了した場合には、出力 FIFO を一掃できます。JPEG_CR レジスタの FF ビット（FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 出力 FIFO の割り込みを無効として、FIFO一掃時の不要な割り込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

JPEG 設定が何か変化する前に、処理の最後で出力 FIFO を一掃する必要があります。

JPEG_SR レジスタの EOCF ビット（変換終了フラグ）は、出力 FIFO がエンプティであるときにのみクリアできます。

JPEG_CONFR1 レジスタの HDR ビット（ヘッダ処理）か、JPEG_CR レジスタの JCEN ビット（JPEG コーデックイネーブル）かをクリアできるのは、JPEG_SR レジスタの EOCF ビットがクリアされているときに限られます。

33.4 JPEG コーデック割り込み

割り込みは次のイベントによって生成できます。

- 入力 FIFO 閾値到達
- 入力 FIFO ノットフル
- 出力 FIFO 閾値到達
- 出力 FIFO ノットエンプティ
- 変換終了
- ヘッダ解析終了

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 257. JPEG コーデック割り込み

割り込みイベント	イベントフラグ	イネーブル制御ビット
入力 FIFO 閾値到達	IFTF	IFTIE
入力 FIFO ノットフル	IFNFF	IFNFIE
出力 FIFO 閾値到達	OFTF	OFTIE
出力 FIFO ノットエンプティ	OFNEF	OFNEIE
変換の終了	EOCF	EOCIE
ヘッダ解析終了	HPDF	HPDIE

33.5 JPEG コーデックレジスタ

33.5.1 JPEG コーデック制御レジスタ（JPEG_CONFR0）

アドレスオフセット：0x0000

リセット値：0x0000 0000

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

ビット 31:1 予約済み

ビット 0 **START**：スタート

このビットは、エンコード処理またはデコード処理の開始または停止を行います。

0：STOP/ABORT

1：Start

このビットを読み出すと常に 0 が返されます。

33.5.2 JPEG コーデック設定レジスタ 1（JPEG_CONFR1）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
YSIZE[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	

ビット 31:16 **YSIZE[15:0]**：Y サイズ

このフィールドにより、ソース画像内のライン数を定義します。

ビット 15:9 予約済み

ビット 8 **HDR**：ヘッダ処理

このビットにより、ヘッダ処理（生成／解析）が有効となります。

0：無効化

1：有効化

ビット 7:6 **NS[1:0]**：スキャンする成分数

このフィールドにより、スキャンヘッダマーカセグメントの成分数から 1 を引いた値を定義します。

ビット 5:4 **COLORSPACE[1:0]**：色空間

このフィールドにより、出力ストリームに挿入する量子化テーブル数から 1 を引いた値を定義します。

00：グレイスケール（量子化テーブル 1 つ）

01：YUV（量子化テーブル 2 つ）

10：RGB（量子化テーブル 3 つ）

11：CMYK（量子化テーブル 4 つ）

ビット 3 **DE**：エンコーダまたはデコーダとしてのコーデック操作

このビットは、エンコード処理またはデコード処理を選択します。

0：エンコード

1：デコード

ビット 2 予約済み

ビット 1:0 **NF[1:0]**：色成分の数

このフィールドにより、色成分数から 1 を引いた値を定義します。

00：グレイスケール（色成分 1 つ）

01：-（色成分 2 つ）

10：YUV または RGB（色成分 3 つ）

11：CMYK（色成分 4 つ）

33.5.3 JPEG コーデック設定レジスタ 2（JPEG_CONFR2）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	NMCU[25:16]									
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NMCU[15:0]															
rw															

ビット 31:26 予約済み

ビット 25:0 **NMCU[25:0]**：MCUの数

エンコード時; このフィールドにより、エンコードするMCU単位数から 1 を引いた値を定義します。

デコード時；このフィールドは、デコードされる完全なMCU単位数から 1 を引いた値を示します（このフィールドはJPEGヘッダの解析後に更新されます）。デコード画像の X や Y のサイズが 8 または 16（サブサンプリング処理に依存します）の倍数ではない場合には、生成される MCU の総数を得るためには、結果として得られる不完全または空である MCU をこの値に加算する必要があります。

33.5.4 JPEG コーデック設定レジスタ 3（JPEG_CONFR3）

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XSIZE[15:0]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:16 **XSIZE[15:0]**：X サイズ

このフィールドにより、ラインあたりのピクセル数を定義します。

ビット 15:0 予約済み

33.5.5 JPEG コーデック設定レジスタ 4-7（JPEG_CONFR4-7）

アドレスオフセット：0x0010 + 0x4 * i、ここで i は 0 から 3 までの画像成分です。

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
rw				rw				rw				rw		rw	rw

ビット 31:16 予約済み

ビット 15:12 **HSF[3:0]**：水平サンプリング係数
成分 i の水平サンプリング係数

ビット 11:8 **VSF[3:0]**：垂直サンプリング係数
成分 i の垂直サンプリング係数

ビット 7:4 **NB[3:0]**：ブロック数
MCUの中である特定色に属するデータ単位数から 1 を引いた値

ビット 3:2 **QT[1:0]**：量子化テーブル
成分 i の量子化テーブルを選択
00：量子化テーブル 0
01：量子化テーブル 1
10：量子化テーブル 2
11：量子化テーブル 3

ビット 1 **HA**：ハフマン AC
AC 係数のエンコードのためのハフマンテーブルを選択
0：未選択
1：選択

ビット 0 **HD**：ハフマン DC
DC 係数のエンコードのためのハフマンテーブルを選択
0：未選択
1：選択

33.5.6 JPEG 制御レジスタ（JPEG_CR）

アドレスオフセット：0x0030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	オフ	IFF	Res.	Res.	Res.	Res.	Res.	Res.	HPDIE	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN
	r0	r0							rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済み

ビット 14 **OFF**：出力 FIFO 一掃

このビットは、出力 FIFO を一掃します。

0：影響なし。

1：出力 FIFO を一掃

このビットを読み出すと常に 0 が返されます。

ビット 13 **IFF**：入力 FIFO 一掃

このビットは、入力 FIFO を一掃します。

0：影響なし。

1：入力 FIFO を一掃

このビットを読み出すと常に 0 が返されます。

ビット 127 予約済み

ビット 6 **HPDIE**：ヘッダ解析終了割り込みイネーブル

このビットは、ヘッダ解析動作の完了時の割り込み生成を有効化します。

0：無効

1：有効

ビット 5 **EOCIE**：変換終了割り込み有効

このビットは、変換終了時の割り込み生成を有効にします。

0：無効

1：有効

ビット 4 **OFNEIE**：出力 FIFO ノットエンプティ割り込み有効

このビットは、出力 FIFO がエンプティではないときの割り込み生成を有効にします。

0：無効

1：有効

ビット 3 **OFTIE**：出力 FIFO 閾値割り込み有効

このビットは、出力 FIFO が閾値に達したときの割り込み生成を有効にします。

0：無効

1：有効

ビット 2 **IFNFIE**：入力 FIFO ノットフル割り込み有効
このビットは、入力 FIFO がエンプティではないときの割り込み生成を有効にします。
0：無効
1：有効

ビット 1 **IFTIE**：入力 FIFO 閾値割り込み有効
このビットは、入力 FIFO が閾値に達したときの割り込み生成を有効にします。
0：無効
1：有効

ビット 0 **JCEN**：JPEG コアイネーブル
このビットは、JPEG コーデックコアを有効にします。
0：無効（内部レジスタがリセット）
1：有効（内部レジスタがアクセス可能）

33.5.7 JPEG ステータスレジスタ（JPEG_SR）

アドレスオフセット：0x0034

リセット値：0x0000 0006

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.
								ro	ro	ro	ro	ro	ro	ro	

ビット 31:8 予約済み

ビット 7 **COF**：コーデック操作フラグ
このビットは、エンコード／デコード操作が進行中であることを示します。
0：非進行中
1：進行中

ビット 6 **HPDF**：ヘッダ解析終了フラグ
デコードモードにおいて、このビットはヘッダ構文解析と内部レジスタの更新が完了したことを示します。
0：未完了
1：完了

ビット 5 **EOCF**：変換終了フラグ
このビットは、エンコード／デコード処理と出力 FIFO に対するデータ転送が完了したことを示します。
0：未完了
1：完了

ビット 4 **OFNEF**：出力 FIFO ノットエンプティフラグ
このビットは、データが出力 FIFO にあることを示します。
0：エンプティ（データなし）
1：ノットエンプティ（データあり）

ビット 4 **OFTF**：出力 FIFO 閾値フラグ
このビットは、出力 FIFO のデータ量が閾値に達したか超過したことを示します。
0：閾値未満
1：閾値以上

ビット 2 **IFNFF**：入力 FIFO ノットフルフラグ
このビットは、入力 FIFO がフルではないことを示します（データ書き込み可）。
0：フル
1：ノットフル

ビット 1 **IFTF**：入力 FIFO 閾値フラグ
このビットは、入力 FIFO のデータ量が閾値未満であることを示します。
0：閾値以上
1：閾値未満

ビット 0 予約済み

33.5.8 JPEG クリアフラグレジスタ（JPEG_CFR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
									w1c	w1c	Res.	Res.	Res.	Res.	Res.

ビット 31:7 予約済み

ビット 6 **CHPDF**：クリアヘッダ解析終了フラグ
“1”を書き込むと JPEG_SR レジスタの HPDF ビットをクリアします。
0：影響なし。
1：クリア

ビット 5 **CEOCF**：クリア変換終了フラグ
“1”を書き込むと JPEG_SR レジスタの ECF ビットをクリアします。
0：影響なし。
1：クリア

ビット 4:0 予約済み

参考資料

33.5.9 JPEG データ入力レジスタ（JPEG_DIR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAIN[31:16]															
wo															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAIN[15:0]															
wo															

ビット 31:0 **DATAIN[31:0]**：データ入力 FIFO
入力 FIFO データレジスタ

33.5.10 JPEG データ出力レジスタ（JPEG_DOR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAOUT[31:16]															
ro															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAOUT[15:0]															
ro															

ビット 31:0 **DATAOUT[31:0]**：データ出力 FIFO
出力 FIFO データレジスタ

33.5.11 JPEG コーデックレジスタマップ

次の表に JPEG コーデックレジスタの一覧を示します。JPEG コーデックレジスタのベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 258. JPEG コーデックレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	JPEG_CONFR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START
	リセット値																																0
0x0004	JPEG_CONFR1	YSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0		0	0
0x0008	JPEG_CONFR2	Res.	Res.	Res.	Res.	Res.	Res.	NMCU[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	JPEG_CONFR3	XSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
0x0010	JPEG_CONFR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	JPEG_CONFR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	JPEG_CONFR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	JPEG_CONFR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020-0x002C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0030	JPEG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	オフ	IFF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN
	リセット値																		0	0								0	0	0	0	0	0
0x0034	JPEG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.
	リセット値																									0	0	0	0	0	1	1	
0x0038	JPEG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
	リセット値																										0	0					
0x0040	JPEG_DIR	DATAIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	JPEG_DOR	DATAOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050-0x014C	QMEM	QMem RAM																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

表 258. JPEG コーデックレジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0150-0x018C	HUFFMIN	HuffMin RAM																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0190-0x020C	HUFFBASE	Res	Res	Res	Res	Res	Res	Res	HuffBase RAM									Res	Res	Res	Res	Res	Res	Res	HuffBase RAM								
	リセット値								X	X	X	X	X	X	X	X	X								X	X	X	X	X	X	X	X	X
0x0210-0x035C	HUFFSYMB	HuffSymb RAM																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0360-0x04FC	DHTMEM	DHTMem RAM																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0500-0x07FC	HUFFENC	Res	Res	Res	Res	HuffEnc RAM												Res	Res	Res	Res	HuffEnc RAM											
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

参考資料

34 真の乱数発生器（RNG）

34.1 概要

RNG は、完全エントロピー出力を 32 ビットサンプルとしてアプリケーションに提供する真の乱数発生器です。ライブエントロピーソース（アナログ）と内部条件付けコンポーネントから構成されています。

RNG は、ライブエントロピーソースとして作用する NIST 準拠の決定論的乱数発生器（DRBG）の構築に使用できます。

RNG の真の乱数発生器は、ドイツの AIS-31 規格に従って検証されました。

34.2 RNG の主な機能

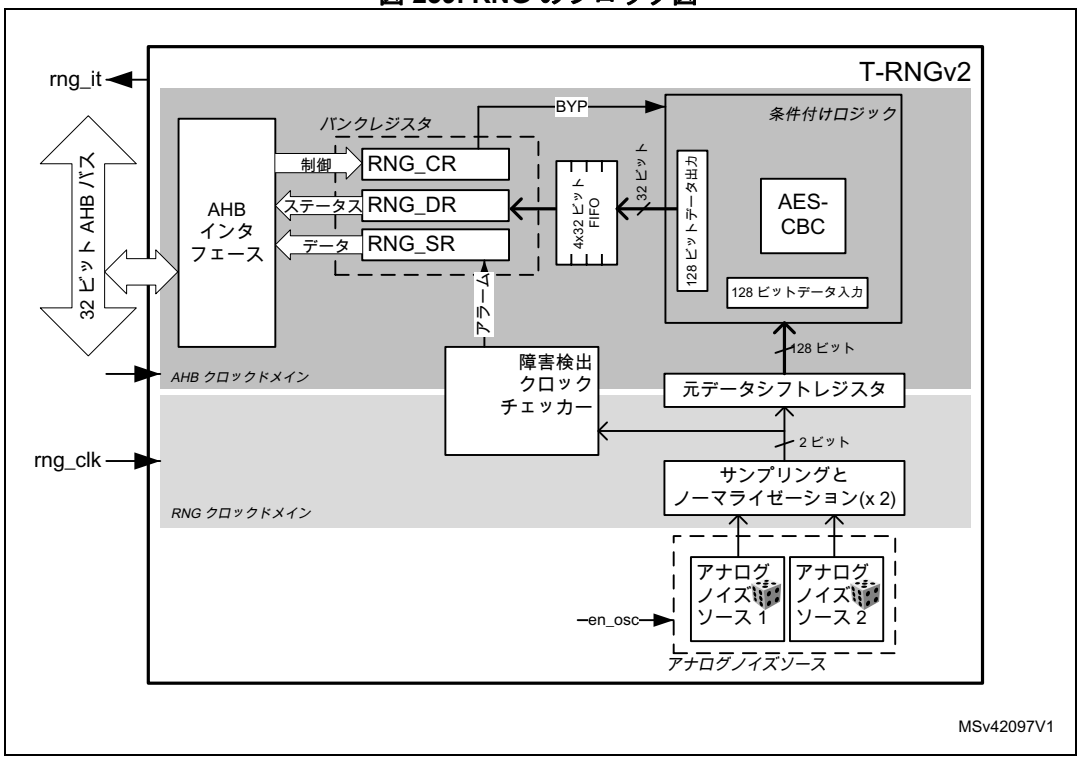
- RNG は、ブロック暗号 AES-CBC で条件付けされたアナログエントロピーソースによって作成された 32 ビットの真の乱数を供給します。
- ドイツコモンクライテリア（CC）スキームの一部である AIS-31 事前定義クラス PTG.2 評価法に従って検証されます。
- 4 個の 32 ビットランダムサンプルを 4x54 AHB クロックサイクルごとに生成します。
- 関連したエラー管理を使った連続した基本的な健全性試験が内蔵されており使用可能です。
  - 非常に低いサンプリングクロック検出および反復カウント試験が搭載されています。
- 無効化して消費電力を低減することができます。
- AMBA AHB スレーブペリフェラルがあり、32 ビットワードのシングルアクセスでのみアクセス可能です（それ以外の書き込みアクセスでは AHB バスエラーが発生します）。＜警告＞ 32 ビットに等しくない書き込みを行うと、レジスタの内容が破壊されることがあります。

34.3 RNG の機能説明

34.3.1 RNG のブロック図

図 239 に RNG ブロック図を示します。

図 239. RNG のブロック図



34.3.2 RNG 内部信号

表 259 では、RNGをより良く理解するために内部の信号のリストを記載しております。 STM32の入出力の信号ではありません。

表 259. RNG 内部入力／出力信号

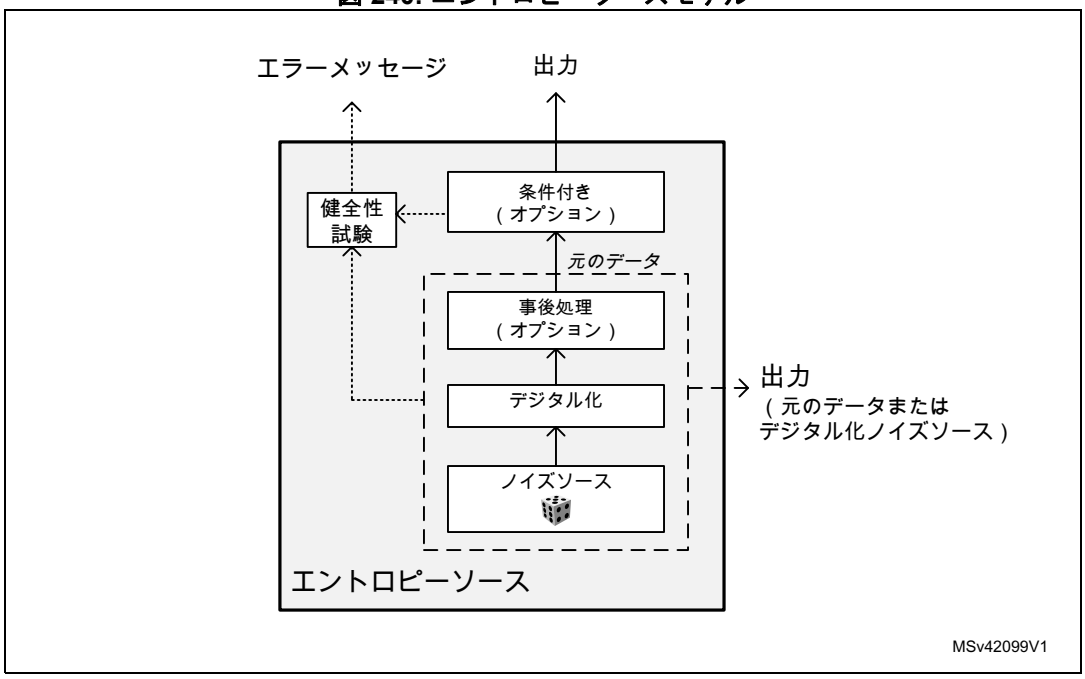
信号名	信号タイプ	説明
rng_it	デジタル出力	RNG グローバル割り込みリクエスト
rng_hclk2	デジタル入力	AHB2 クロック
rng_clk	デジタル入力	RNG 専用クロック、rng_hclk2 と非同期

34.3.3 乱数の生成

真の乱数発生器（RNG）は、決定論的な間隔で AHB インタフェースを介して真にランダムなデータを伝送します。RNG は、*図 240* に描かれているエントロピーソースモデルを実装し、アプリケーションに 3 つの主な機能を提供します。

- エントロピーソースボックスのビット列出力の収集
- 検証のためのノイズソースのサンプル取得
- 連続した健全性試験によるエラーメッセージの収集

図 240. エントロピーソースモデル



RNG の主なコンポーネントは次の通りです。

- 物理的ランダム性のソース（アナログノイズソース）
- これらのアナログノイズソースのデジタル化ステージ
- 暗号的に条件付けされたノイズソースを配信するステージ
- エントロピーソース出力（バッファ付き）とノイズソースサンプル（こちらもバッファ付き）の両方に対する出力バッファ
- エントロピーソース全体に対してテストを行う健全性監視ブロック

これらすべてのコンポーネントについて以下に詳述します。

ノイズソース

ノイズソースは、非決定論的なエントロピーを生成する構成要素のノイズソースで、これは出力の不確実性に関連したものです(デジタル化されたノイズソースやソースデータ)以下の要素で構成されます。

- それぞれXORされたフリーランのリングオシレータの出力 3 つに基づく 2 個のアナログノイズソース *セクション 34.4 : RNG の低消費電力時の取り扱い* に記載しているとおり、電力低減のためにそれらのアナログオシレータは無効化できます。
- 専用クロック入力(**rng_clk**)でクロックされたこれらの出力のサンプリングステージでは、2ビットの元データ出力を伝えます。

ノイズソースサンプリングステージで、ノイズソース出力信号と専用 **rng_clk** 入力の非同期によって発生する可能性がある準安定問題を解決します。ノイズソースサンプリングは、AHB インタフェースクロック周波数（**rng_hclk**）に対しても独立しています。

注： ***セクション 34.7：エントロピーソース検証では、推奨される RNG クロック周波数が示されています。***

後処理

真の乱数ノイズソースから取得されるサンプル値は、2 ビットのビット列から成ります。このノイズソース出力はバイアスされるため、RNG が許容レベルまでそのバイアスを軽減する後処理コンポーネントを実行します。

より具体的に言えば、ノイズソースの 2 ビットそれぞれに対して、RNG は、サンプルノイズソースと反転したサンプルノイズソースからビットを半分ずつ取得します。こうして、ソースが“0”より“1”を多く生成する（またはその逆）場合、フィルタされます。

条件付け

RNG の条件付けコンポーネントは、結果として得られる固定長ビット列出力（128ビット）のエントロピー率を増加させる決定論的機能です。

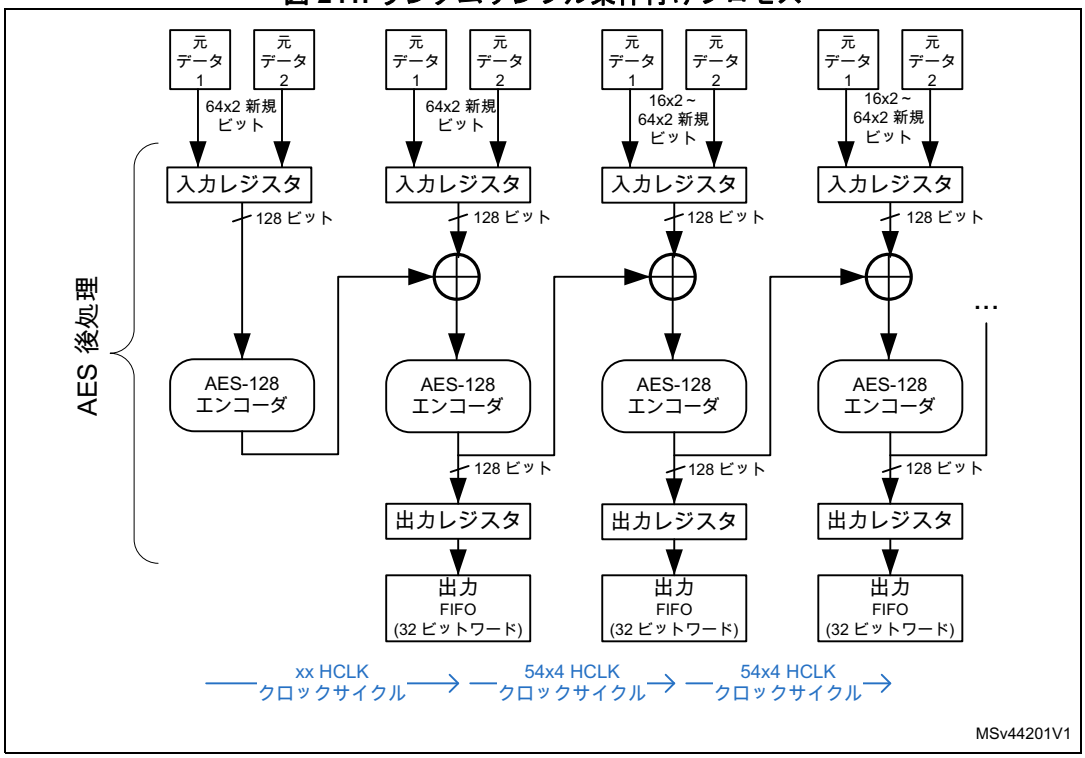
使用される条件付けアルゴリズムは、キーの長さが 128 ビットの AES CBC に基づくブロック暗号であり、*図 241* に示すように、54x4 AHB クロックサイクルごとに 128 ビットのランダムサンプルが生成されます。

注： ***RNG 初期化中のレイテンシは、セクション 34.6：RNG 処理時間に記載されています。***

デジタル化ノイズソース 2 個からやってくる元データは、AES 後処理ステージの入力において一緒にインターリーブされます（元ビット 1 + 元ビット 2 + 元ビット 1 + 元ビット 2 …）。

32 ビット以上の元データが受信済みであり、出力 FIFO を再フィルする必要がある場合に、AES 計算がトリガされることにも注意してください。したがって、64 RNG クロックサイクル後にアプリケーションによって RNG 128 ビット FIFO が空になる場合に、RNG 出力エントロピーは最大となります。

図 241. ランダムサンプル条件付けプロセス



条件付けコンポーネントは、より高速な AHB クロックによりクロック制御されます。

注： ***RNG クロックサイクルあたり 2 ビットのノイズソースが AES 入力バッファにロードできます。***

出力バッファ

データ出力バッファは、条件付けコンポーネント（AES-CBC）から出力された 32 ビットワードを 4 つまで格納できます。RNG_DR レジスタを通じて出力 FIFO から 4 ワード読み出されると、128 ビット条件付け出力レジスタの内容は出力 FIFO にプッシュされ、新たな AES 計算が自動的に始まります。213 AHB クロックサイクル（AES 計算実行時間）後に、新たな 4 ワードが条件付け出力レジスタに追加されます。

乱数が RNG_DR レジスタを介して使用できるときは必ず、DRDY フラグが“0”から“1”に遷移します。このフラグは、RNG_DR レジスタから 4 ワード読み出した後、出力バッファが空になるまで”1”になります。

注： ***割り込みが有効の場合、このデータレディフラグが“0”から“1”に遷移するときに割り込みが生成されます。その後、上記で説明したように割り込みは RNG によって自動的にクリアされます。***

健全性チェック

このコンポーネントはエントロピーソース全体（そのノイズソースとともに）を開始し、その後期待通りに実行し、高い確立と信頼性ですばやく不具合の取り込みを保証することを確実にします。

RNG は、以下のような健全性チェック機能を実行します。

1. *ランタイム時に*エントロピーソースに適用される動作試験
   - 繰り返しのカウントテスト、エラーフラグは以下の場合:
   a) ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して提供しました。
   b) 両方のノイズソースが 32 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 16 ビット以上連続して提供しました。
2. ベンダ固有の連続した試験
   - リアルタイムの「遅すぎる」サンプリングクロック検出回路。1 RNG クロックサイクル（分周後）が 32 で分周された AHB クロックサイクルより小さい場合にエラーフラグを立てます。

エラーコンディションが検出された時に、RNG_SR レジスタの CECS および SECS ステータスビットに表示されます、詳細は*セクション 34.3.7：エラー管理*

注： ***エラーが検出された場合、割り込みを生成することができます。***

34.3.4 RNG 初期化

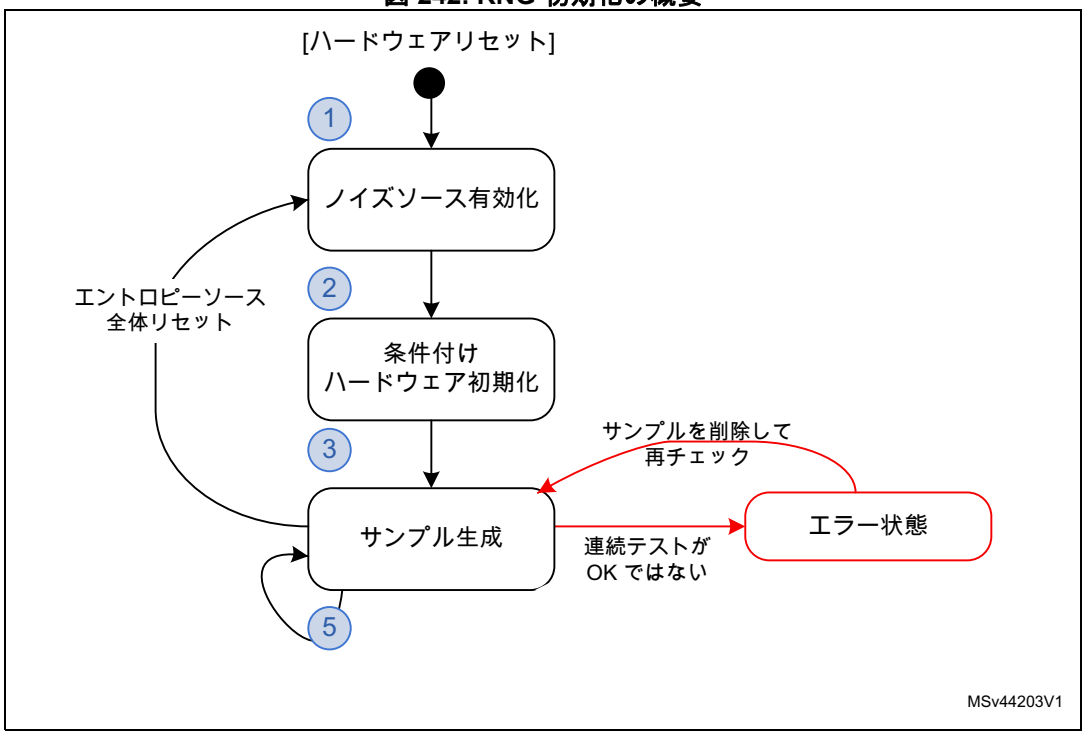
図 242 に RNG 簡易ステートマシンを図示します。

ハードウェアリセットが発生したときに、次の一連のイベントが発生します。

1. アナログノイズソースが有効になり、その直後にロジックがアナログ出力のサンプリングを開始し、128 ビット条件付けシフトレジスタに入力します。
2. 条件付けロジックが有効になり、AES-CBC 後処理の内容は 2 つの 128 ビットノイズソースを用いて初期化されます（128 ビットはキー用です）。
3. AES 内部入力データバッファに 128 ビット再入力され、AES 後処理ループが 1 回実行されます。その後、出力バッファには AES 後処理の結果が入力されます。
4. 出力バッファは RNG の使用方法に応じて自動的に再入力されます。

関連する初期化時間については、*セクション 34.6 : RNG 処理時間*を参照してください。

図 242. RNG 初期化の概要



34.3.5 RNG 動作

通常動作

割り込みを使用して RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの IE ビットをセットすることによって、割り込みを有効にします。同時に、RNGEN=1 にセットすることにより、RNG を有効にします。
2. 乱数の準備ができたとき、またはエラーが発生したとき、割り込みが生成されるようになります。そのため、割り込みごとに次をチェックします。
   - 発生したエラーはありません。RNG_SR レジスタで SEIS および CEIS ビットを“0”にセットする必要があります。
   - 乱数が準備できています。RNG_SR レジスタで DRDY ビットを“1”にセットする必要があります。
   - 上記 2 つの条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容が読み出せるようになります。AES 出力バッファに有効なデータがある場合には、アプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットは High のままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、*セクション 34.3.7* に記載されたリカバリシーケンスを使用する必要があります。

ポーリングモードで RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの RNGEN ビットを“1”にセットして、乱数発生を有効にします。
2. RNG_SR レジスタを読み出し、次を確認します。
   – 発生したエラーはありません（SEIS および CEIS ビットは“0”にセットされています）。
   – 乱数が準備できています（DRDY ビットは“1”にセットされています）。
3. 上記の条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容を読み出します。AES 出力バッファに有効なデータがある場合には、アプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットは High のままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、*セクション 34.3.7* に記載されたリカバリシーケンスを使用する必要があります。

注： ***データが準備できていない場合（DRDY=”0”）、RNG_DR は 0 を返します。***

低電力動作

消費電力がアプリケーションに対して懸念となる場合、*セクション 34.4：1186 ページのRNG の低消費電力時の取り扱い*に示すように低電圧ストラテジーを使用できます。

ソフトウェア後処理

AIS-31 の承認を満たすには、特別なソフトウェア後処理／条件付けは必要ありません。NIST が承認した 128 ビットのセキュリティ強度を持つ DRBG が必須である場合、RNG 真の乱数発生器付近で認められた乱数発生器ソフトウェアを構築する必要があります。

内蔵されている健全性チェック機能の説明は*セクション 34.3.3：乱数の生成* を参照してください。

34.3.6 RNG クロック供給

RNG は、AHB バスクロックと専用 RNG クロックの 2 種類の異なるクロックで動作します。

AHB クロックは、AHB バンクレジスタおよび条件付けコンポーネントにクロック供給するために使用されます。RNG クロックは、ノイズソースサンプリングに使用されます。推奨されるクロック設定は、*セクション 34.7：エントロピーソース検証*で詳しく説明されています。

注意： ***RNG_CR レジスタの CED ビットが“0”にセットされている場合、RNG クロック周波数は 32 で分周された AHB クロック周波数より高くする必要があります。そうしないと、クロックチェッカーがクロックエラーのフラグ（RNG_SR レジスタの CECS または CEIS）を立て、RNG は乱数の生成を停止します。***

詳細（AHB および RNG クロックドメイン）については、*セクション 34.3.1：RNG のブロック図*を参照してください。

34.3.7 エラー管理

このセクションに詳細を示しているとおり、乱数生成と並行して、健全性チェックブロックにより、正しいノイズソース動作と RNG ソースクロックの周波数が検証されます。関連するエラー状態も記載されています。

クロックエラー検出

クロックエラー検出が有効（CED = 0）となっており、RNG クロック周波数が低すぎる場合、RNG は乱数生成を停止し、**CEIS** および **CECS** ビットを両方“1”にセットして、クロックエラーが発生したことを示します。この場合、アプリケーションによって RNG クロックが正しく設定されていること（*セクション 34.3.6：RNG クロック供給*を参照）をチェックして、CEIS ビットの割り込みフラグを

クリアする必要があります。RNG クロックが正しく動作するとすぐに、CECS ビットは自動的にクリアされます。

CECS フラグが“0”にセットされたときのみ、RNG が正しく動作します。ただし、クロックエラーはその前に発生した乱数には影響しないため、RNG_DR レジスタの内容は使用できます。

ノイズソースエラー検出

ノイズソース（またはシード）エラーが発生すると、RNG は乱数生成を停止し、**SEIS** および **SECS** ビットを両方“1”にセットして、シードエラーが発生したことを示します。RNG_DR レジスタにおける値が利用可能な場合であっても、エントロピーが十分でない可能性があるので使用することはできません。初期化フェーズ中にエラーが検出された場合、RNG によって初期化シーケンス全体が自動的に再スタートされます。

RNG 初期化後にシードエラーからの完全なリカバリを行うには、次のシーケンスを使用します。

1. SEIS に“0”を書き込んでクリアします。
2. RNG_DR レジスタから 12 ワード読み出し、各ワードを破棄してパイプラインをきれいにします。
3. SEIS がクリアされたままであることを確認します。乱数発生は正常に戻りました。

34.4 RNG の低消費電力時の取り扱い

消費電力が懸念される場合、RNG_CR レジスタの RNGEN ビットを“0”にセットして DRDY ビットを“1”にセットするとすぐに、RNG を無効化できます。RNGEN=’0’であっても AES 後処理ロジックと出力バッファは動作を続けていますので、ソフトウェアは次の機能を使用できます。

- 出力バッファに有効なワードが存在する場合には、RNG_DR レジスタから乱数をもう 4 個読み出すことができます。
- AES 出力内部レジスタに有効なビットが存在する場合には、RNG_DR レジスタから乱数をもう 4 個読み出すことができます。そうではない場合には、ノイズソースから新たに最低でも 32 ビット取得して AES 計算が完全に終わるまで、RNG はアプリケーションによって再度有効化される必要があります。新たなビットのサンプルには 16 RNG クロックサイクル、AES 処理 1 回には 216 AHB クロックサイクルかかります。

RNG を無効化する場合、すべてのアナログシード発生回路を無効にします。その消費電力は、データシートの電気的特性セクションに示されています。また、RNG クロックによってクロックの供給を受けるロジックはすべてゲートします。この方法では、RNG 初期化時間のために、RNG_DR レジスタでランダムサンプルが利用可能となるまでのレイテンシが増加することに注意してください。

初期化中（すなわちDRDY ビットが初めて立ち上がるよりも十分前に）に RNG ブロックが無効化された場合、初期化シーケンスは、RNGEN ビットが“1”にセットされて停止したところから再開されます。

34.5 RNG 割り込み

RNG では、割り込みは次のイベントによって生成できます。

- データレディフラグ
- シードエラー、*セクション 34.3.7：エラー管理*を参照
- クロックエラー、*セクション 34.3.7：エラー管理*を参照

表 260 に示すように専用割り込みイネーブル制御ビットが使用できます。

表 260. RNG 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
データレディフラグ	DRDY	IE
シードエラーフラグ	SEIS	IE
クロックエラーフラグ	CEIS	IE

RNG_CR レジスタのマスクビットまたは全体的な割り込み制御ビット IE を変更することにより、上記の割り込みソースを個別に有効にしたり無効にしたりすることができます。個別の割り込みソースのステータスは RNG_SR レジスタから読み出すことができます。

注： ***割り込みは、RNG が有効化されている場合にのみ生成されます。***

34.6 RNG 処理時間

AES は 4x54 AHB クロックサイクルごとに 32 ビット乱数を 4 つ生成できますが、デバイスがリセットを抜けた後初回の乱数にはそれ以上の時間が必要です（*セクション 34.3.4 : RNG 初期化*参照）。

初めて RNG を有効化した後、次のいずれかの時間の後に初めてランダムデータが利用可能となります。

- 128 RNG クロックサイクル + 426 AHB サイクル（f_{AHB} < 160 MHz かつ f_{RNG} = 48 MHz である場合）
- 192 RNG クロックサイクル + 213 AHB サイクル（f_{AHB} ≥ 160 MHz かつ f_{RNG} = 48 MHz である場合）

34.7 エントロピーソース検証

34.7.1 概要

RNG から使用できるエントロピーの量にアクセスするために、STMicroelectronics は AIS-31 PTG.2 の試験セットに対して RNG の試験を行いました。結果は要求に応じて提供でき、顧客側で AIS リファレンス・ソフトウェアを使用して測定を再現できます。旧 NIST SP800-22 試験セットに対して RNG を試験することもできます。

34.7.2 検証条件

STMicroelectronics は、以下の条件下で RNG 真の乱数発生器を検証しました。

- RNG クロック rng_clk= 48 MHz（RNG_CR レジスタの CED ビット = ’0’）および rng_clk = 400 kHz（RNG_CR レジスタの CED ビット = ‘1’）
- AHB クロック rng_hclk= 216 MHz

34.7.3 データ収集

統計テストを行うためには、サンプルを元データレベルでのエントロピーソースからだけではなく、エントロピーソースの出力からも収集する必要があります。

RNG の手法は、条件付けステージをバイパスするしないに RNG_CR レジスタの BYP ビットを使用して、同一の 32 ビットバッファを使用して両方のサンプルを出力するというものです。言い換えれば、アプリケーションによって元データを取得する必要がある場合には、次のシーケンスを推奨します。

1. RNG_CR レジスタの RNGEN ビットに“0”を、BYP ビットに“1”を書き込みます。
2. RNG_CR レジスタの RNGEN ビットに“1”を、BYP ビットに“1”を書き込みます。通常の出力サンプルと同じ方法で元サンプルが取得されます。

34.8 RNG レジスタ

RNG は、制御レジスタ、データレジスタ、ステータスレジスタと連動します。

34.8.1 RNG 制御レジスタ（RNG_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **BYP**：バイパスモード有効

このビットは、後処理／条件付きロジックのバイパスを有効化または無効化します。この機能は、RNG 検証に使用されます。

0：バイパスモードは無効です。ノイズソースサンプルは後処理され、条件付けされます。

1：バイパスモードは有効です。ノイズソースサンプルは後処理されず、条件付けもされません。アプリケーションによって直接読み出しできます。

このビットへの書き込みは、RNGEN=0 の場合のみ考慮されます。

ビット 5 **CED**：クロックエラー検出

0：クロックエラー検出は有効です。

1：クロックエラー検出は無効です。

RNG が有効になっている場合、動作中にクロックエラー検出を有効化／無効化できません。CED を有効化／無効化するには、RNG を無効化する必要があります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **IE**：割り込みイネーブル

0：RNG 割り込みは無効です。

1：RNG 割り込みは有効です。RNG_SR レジスタで DRDY =“1”、SEIS=“1” または CEIS=“1”になると、割り込みは直ちにペンディングとなります。

ビット 2 **RNGEN**：真の乱数発生器イネーブル

0：真の乱数発生器は無効です。アナログノイズソースの電力がオフになり、RNG クロックによって供給されるロジックがゲートされます。

1：真の乱数発生器は有効です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

34.8.2 RNG ステータスレジスタ（RNG_SR）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
									rc_w0	rc_w0			r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **SEIS**：シードエラー割り込みステータス

このビットは、SECS として同じ時間でセットされます。“0”を書き込むことによってクリアされます。

0：障害シーケンスは検出されませんでした。

1：1 つ以上の障害シーケンスが検出されました。詳細については、**SECS** ビットの説明を参照してください。

RNG_CR レジスタで IE = “1” である場合、割り込みがペンディングとなります。

ビット 5 **CEIS**：クロックエラー割り込みステータス

このビットは、CECS として同じ時間でセットされます。“0”を書き込むことによってクリアされます。

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。

1：RNG クロックが遅すぎると検出されました（fRNGCLK < fHCLK/32）。

RNG_CR レジスタで IE = “1” である場合、割り込みがペンディングとなります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SECS**：シードエラーの現在のステータス

0：障害のあるシーケンスは現在のところ検出されていません。SEIS ビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。

1：次の障害シーケンスのうち 1 つ以上が検出されました。

– ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して提供しました。

– 両方のノイズソースが 32 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 16 ビット以上連続して提供しました。

ビット 1 **CECS**：クロックエラーの現在のステータス

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。CEIS ビットがセットされている場合、これは遅いクロックが検出されたが、回復していることを意味します。

1：遅すぎる RNG クロックです（fRNGCLK< fHCLK/32）。

注： CECS ビットは、RNG_CR レジスタの CED ビットが “0”にセットされたときにのみ有効です。

ビット 0 **DRDY**：データレディ

0：RNG_DR レジスタがまだ有効でなく、乱数データは利用できません。

1：RNG_DR レジスタに有効な乱数データが入っています。

出力バッファがいったん空になると（RNG_DR レジスタを読んだ後）、新しい有効な乱数値が生成されるまでこのビットは“0”に戻ります。

注： ペリフェラルが無効化されている場合（RNG_CR レジスタのRNGEN=’0’）に、DRDY ビットが立ち上がることがあります。

RNG_CR レジスタで IE=“1”の場合、DRDY=“1”のときに割り込みが生成されます。

34.8.3 RNG データレジスタ（RNG_DR）

アドレスオフセット：0x008

リセット値：0x0000 0000

RNG_DR レジスタは読み出し専用レジスタであり、読み出されると 32 ビットの乱数値を返します。このレジスタは、いったん読み出されると、出力 FIFO が空の場合、AHB クロック 216 周期の後、新しい乱数値を出力します。

このレジスタの内容は、RNGEN=“0”であっても、DRDY=“1”の場合に有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RNDATA[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RNDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RNDATA[31:0]**：乱数データ

DRDY=“1”のときに有効な 32 ビット乱数データです。DRDY=“0”の場合、RNDATA の値は 0 です。

34.8.4 RNG レジスタマップ

表 261 に、RNG レジスタマップとリセット値を示します。

表 261. RNG レジスタマップとリセット値

オフセット	レジスタ名 / リセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RNG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYP	CED	Res.	IE	RNGEN	Res.	Res.
	リセット値																										0	0		0	0		
0x004	RNG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
	リセット値																										0	0			0	0	0
0x008	RNG_DR	RNDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

35 暗号プロセッサ（CRYP）

35.1 概要

暗号プロセッサ（CRYP）は、DES、トリプル DES、または AESアルゴリズムを使用したデータの暗号化と復号化の両方に使用できます。次の規格に完全に準拠して実装されています。

- 連邦情報処理規格公報（FIPS：Federal Information Processing Standards Publication）（FIPS PUB 46-3, Oct 1999）および米国規格協会（ANSI X9.52）によって規定されているデータ暗号化標準（DES）およびトリプル DES（TDES）。
- 連邦情報処理規格公報（FIPS PUB 197, Nov 2001）によって規定されている高度暗号化標準（AES：Advanced Encryption Standard）。

複数のキーサイズおよび連鎖モードがサポートされます。

- キーあたり 8 ビットパリティの標準 56 ビットキーに対応した DES/TDES 連鎖モード ECB および CBC
- キーサイズが 128、192 または 256 ビットの AES 連鎖モード ECB、CBC、CTR、GCM、GMAC、CCM

CRYP は 32 ビットの AHB ペリフェラルです。入力および出力データの DMA 転送をサポートします（2 つの DMA チャネルが必要です）。ペリフェラルには、性能改善のために入力 FIFO と出力 FIFO（各 8 ワードの深さ）も含まれています。

CRYP ペリフェラルによって、STM32 暗号ライブラリにパッケージされた AES と DES の暗号化アルゴリズムがハードウェアで高速化されます。

35.2 CRYP の主な機能

- 次の規格に準拠して実装されています。
  - NIST *FIPS 公報 46-3、データ暗号化標準（DES）*
  - ANSI *X9.52、トリプルデータ暗号化アルゴリズム動作モード*
  - NIST *FIPS 公報 197、高度暗号化標準（AES）*
- AES対称ブロック暗号実装
  - 128 ビットデータブロックの処理
  - 128、192、256 ビットの暗号キー長に対応
  - 次の複数の連鎖モードで暗号化と復号化：電子コードブック（ECB）、暗号ブロック連鎖（CBC）、カウンタモード（CTR）、ガロアカウンタモード（GCM）、ガロアメッセージ認証コードモード（GMAC）およびCBC-MACカウンタ（CCM）
  - AES-ECB モードで 128 ビット（または 256 ビット）キーを使用して 1 個の 128 ビットブロックのデータを処理する場合は 14（または 18）クロックサイクル
  - キー導出ステージ（ECB または CBC の復号化のみ）を備えた内蔵キースケジューラ
- DES／トリプル DES 暗号化／復号化実装
  - 64 ビットデータブロックの処理
  - 64、128、192 ビットの暗号キー長（パリティ付き）に対応
  - ECB と CBC の連鎖モードに対応した暗号化と復号化
  - 単純 DES アルゴリズムの直接実装（シングルキー K1 を使用）
  - DES（またはトリプル DES）ECB モードで 1 個の 64 ビットブロックのデータを処理する場合は 16（または 64）クロックサイクル

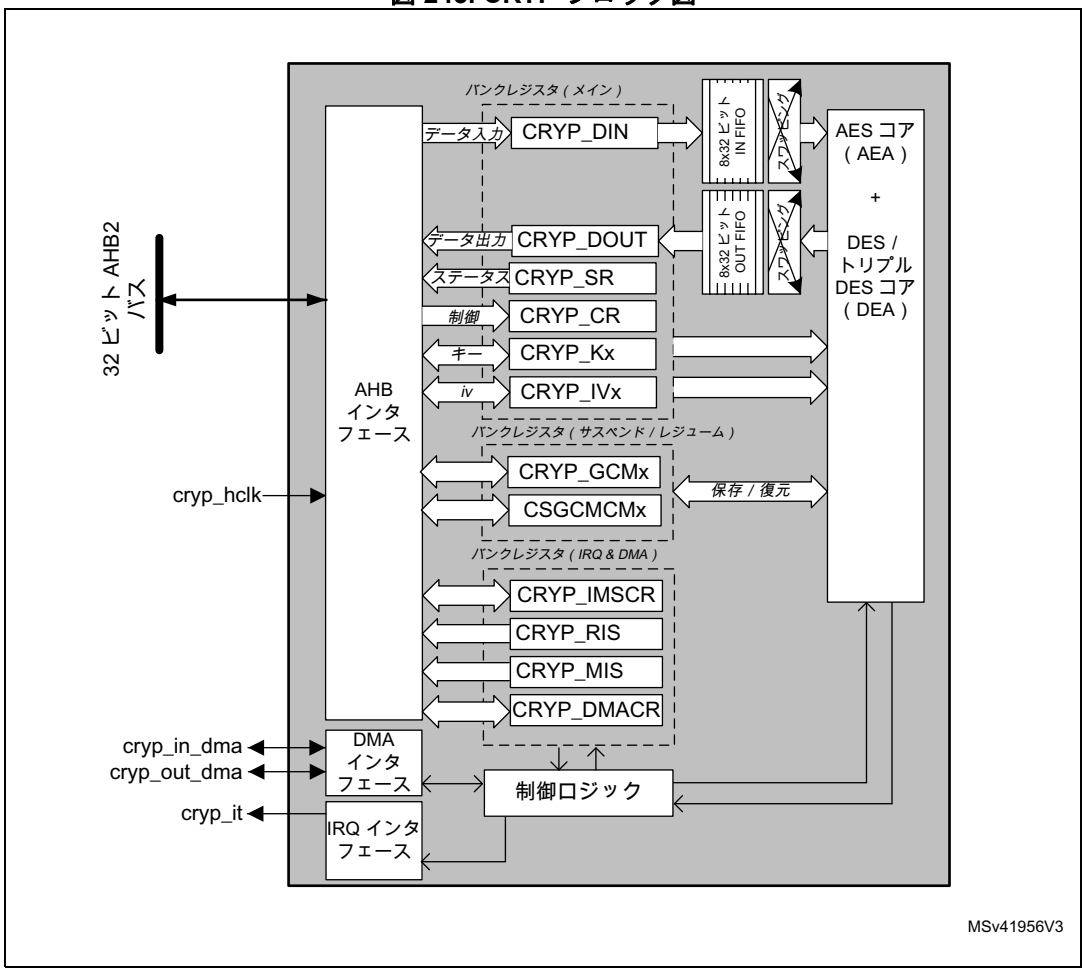
  - 暗号文スチーリングのソフトウェア実装
- DES／トリプル DES および AES に対して共通の特徴
  - 32 ビットワードのシングルアクセスでのみアクセス可能な AMBA AHB スレーブペリフェラル（それ以外は AHB バスエラーが発生し、書き込みアクセスは無視されます）
  - 暗号化キーを格納するための 256 ビットレジスタ（8x 32 ビットレジスタ）
  - 初期化ベクタを格納するための 128 ビットレジスタ（4x 32 ビット）
  - DES ブロック 4 個または AES ブロック 2 個の入力に対応する 8 個の 32 ビットワードの内部入力 FIFO が付いた 1x32 ビット入力バッファ
  - 処理済みの DES ブロック 4 個または AES ブロック 2 個に対応する 8 個の 32 ビットワードの内部出力 FIFO が付いた 1x32 ビット出力バッファ
  - 2 つのチャネル（1 チャンネルは入力データ用、 1 チャンネルは処理済みデータ用）を使用したダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御シングル転送とバースト転送に対応
  - 1、8、16、32 ビットデータをサポートするデータスワッピングロジック
  - 暗号プロセッサがより優先度の高い別メッセージを処理する必要がある場合には、ソフトウェアはメッセージをサスペンド可能（サスペンド／レジューム動作）

35.3 CRYP の機能詳細

35.3.1 CRYP ブロック図

図 243 に、暗号プロセッサのブロック図を示します。

図 243. CRYP ブロック図



35.3.2 CRYP 内部信号

表 262 には、知っておくと便利な暗号プロセッサレベルの内部信号のリストが記載されています。STM32の製品レベルの信号（パッド信号）ではありません。

表 262. CRYP 内部入力／出力信号

信号名	信号タイプ	説明
cryp_hclk	デジタル入力	AHB バスクロック
cryp_it	デジタル出力	暗号プロセッサグローバル割り込みリクエスト
cryp_in_dma	デジタル入出力	入力 FIFO DMA バーストリクエスト／応答
cryp_out_dma	デジタル入出力	出力 FIFO DMA バーストリクエスト／応答（DESに対するシングルリクエスト付き）

35.3.3 CRYP DES／トリプル DES 暗号コア

概要

DES/トリプル DES 暗号コアは、3 つのコンポーネントで構成されています。

- DES アルゴリズム（DEA コア）
- 複数のキー（DES アルゴリズムに 1 つ、トリプル DES アルゴリズムに 1 から 3 つ）
- 初期化ベクタ（CBC モードのみで使用）

DES/トリプル DES 暗号コアは、次の 2 種類のモードで動作します。

- **ALGODIR=0**：CRYP_Kx レジスタに格納されたキーを使用した平文暗号化
- **ALGODIR=1**：CRYP_Kx レジスタに格納されたキーを使用した暗号文復号化

動作モードは、CRYP_CR レジスタの ALGODIR ビットをプログラムして選択します。

標準的データ処理

DES モードでの暗号プロセッサの標準的な使用方法については、*セクション 35.3.10：CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）*を参照してください。

注： ***CBC モードの IV レジスタを除き、中間 DEA ステージの出力が暗号境界の外に明かされることは決してありません。***

DES キーイングと連鎖モード

トリプルDES では、3 種類のキーイングオプションが可能です。

- *3 つの独立キー。*

  最初のオプションでは、すべてのキーが独立、すなわち K1、K2、K3 が独立していると規定しています。FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、このオプションをキーイングオプション 1、トリプルDES を 3 キートリプルDES としています。

- *2 つの独立キー。*

  2 番目のオプションでは、K1 と K2 が独立しており、K3 が K1 に等しい、すなわち K1 と K2 が独立し、K3 = K1 であると規定しています。FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、この 2 つ目のオプションをキーイングオプション 2、トリプルDES を 2 キートリプルDES としています。

- *3 つの同一キー。*

  3 番目のオプションでは、K1、K2、K3 が等しい、すなわち

  K1 = K2 = K3 であると規定しています。

  FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、この 3 番目のオプションをキーイングオプション 3 としています。この「1 キー」トリプルDES は、シングル DES と同等です。

DES ハードウェアでは次の連鎖アルゴリズムがサポートされ、CRYP_CR レジスタの ALGOMODE ビットによって選択できます。

- 電子コードブック（ECB）
- CBC（暗号ブロック連鎖）

これらのモードについては、*セクション 35.3.10：CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）*で詳しく説明されています。

35.3.4 CRYP AES暗号コア

概要

AES 暗号コアは、次のコンポーネントで構成されています。

- AES アルゴリズム（AES コア）
- バイナリガロアフィールド（GF2mul）の乗算器
- キー情報
- 初期化ベクタ（IV）またはノンス情報
- 連鎖アルゴリズムロジック（XOR、フィードバック／カウンタ、マスク）

AES コアは 128 ビット、192 ビット、または 256 ビットのキー長を使用して 128 ビットデータブロック（4 ワード）で動作を行います。連鎖モード次第で、ペリフェラルは 0 個または 1 個の 128 ビット初期化ベクタ（IV）を必要とします。

暗号ペリフェラルには次の 2 種類の動作モードがあります。

- **ALGODIR=0**：CRYP_Kx レジスタに格納されたキーを使用した平文暗号化
- **ALGODIR=1**：CRYP_Kx レジスタに格納されたキーを使用した暗号文復号化ECB または CBC の連鎖モードが選択されると、暗号ペリフェラルによって初期化キー導出プロセスが自動的に実行されます。

動作モードは、CRYP_CR レジスタの ALGODIR ビットをプログラムして選択します。

標準的データ処理

AES モードでの暗号プロセッサの標準的な使用方法については、*セクション 35.3.11：CRYP AES基本連鎖モード（ECB、CBC）*を参照してください。

注： ***IV レジスタを除き、中間 AEA ステージの出力が暗号境界の外に明かされることは決してありません。***

AES連鎖モード

暗号プロセッサでは次の連鎖アルゴリズムがサポートされ、CRYP_CR レジスタの ALGOMODE ビットによって選択できます。

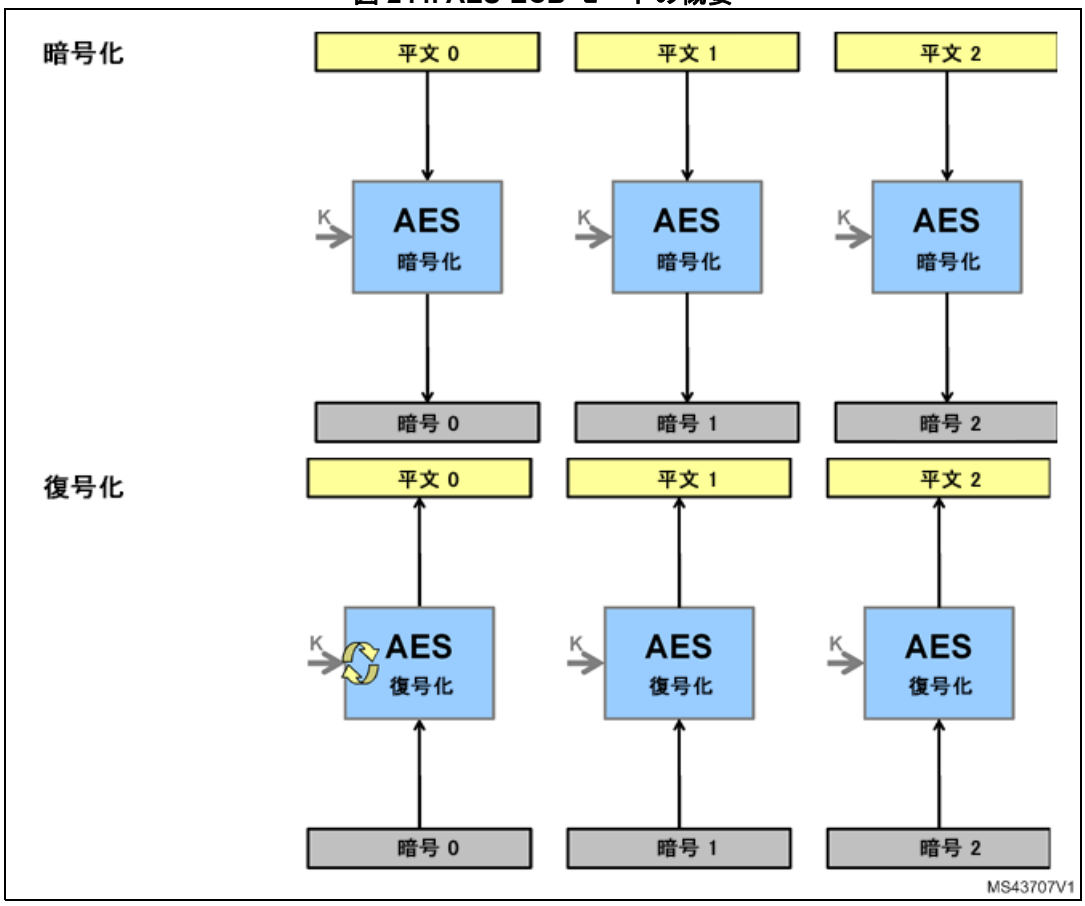
- 電子コードブック（ECB）
- CBC（暗号ブロック連鎖）
- CTR（カウンタモード）
- ガロア／カウンタモード（GCM）
- ガロアメッセージ認証コードモード（GMAC）
- CBC-MAC付きカウンタ（CCM）

これらの連鎖モードに関する簡単な紹介は、以下の項目に記載されています。

詳細については、*セクション 35.3.11：CRYP AES基本連鎖モード（ECB、CBC）*以降を参照してください。

AES ECB（電子コードブック）

図 244. AES-ECB モードの概要

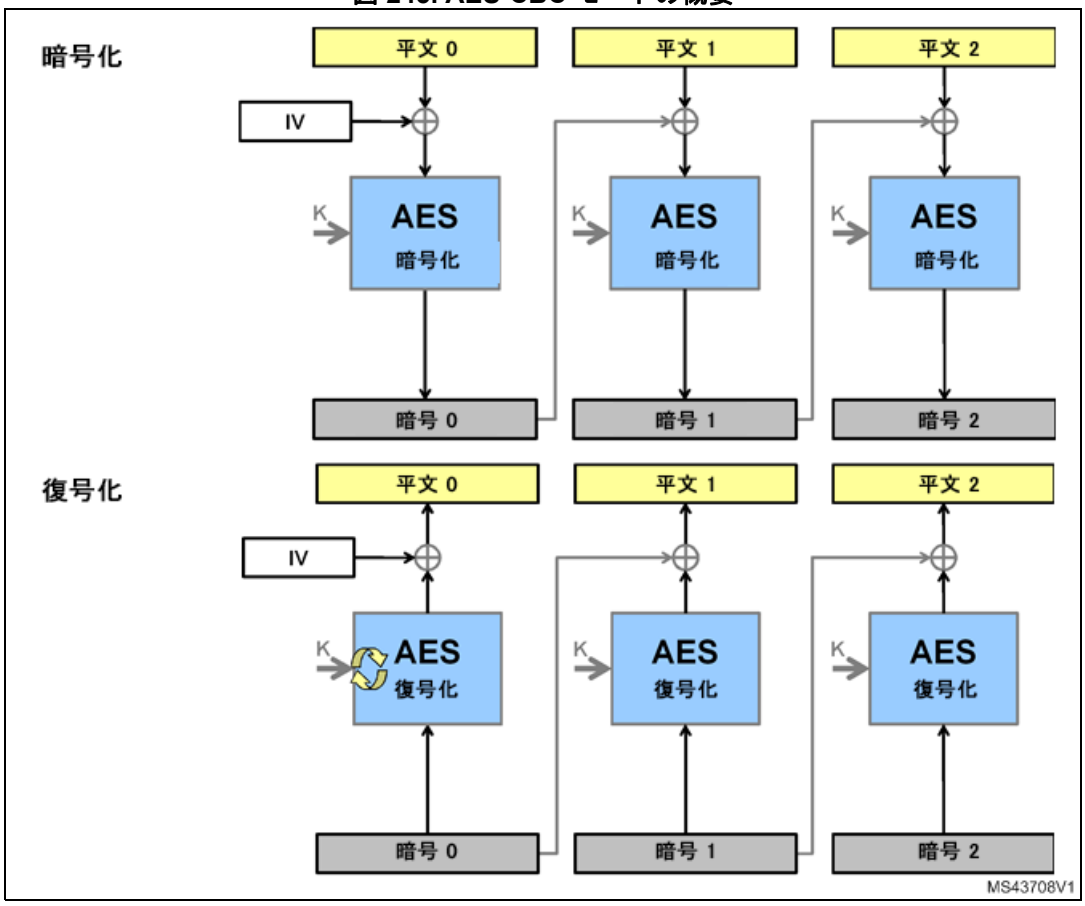


ECBは最も単純な動作モードです。連鎖操作も、特別な初期化ステージもありません。メッセージはブロックに分割され、各ブロックが個別に暗号化または復号化されます。

注： ***復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。***

AES 暗号ブロック連鎖（CBC）

図 245. AES-CBC モードの概要

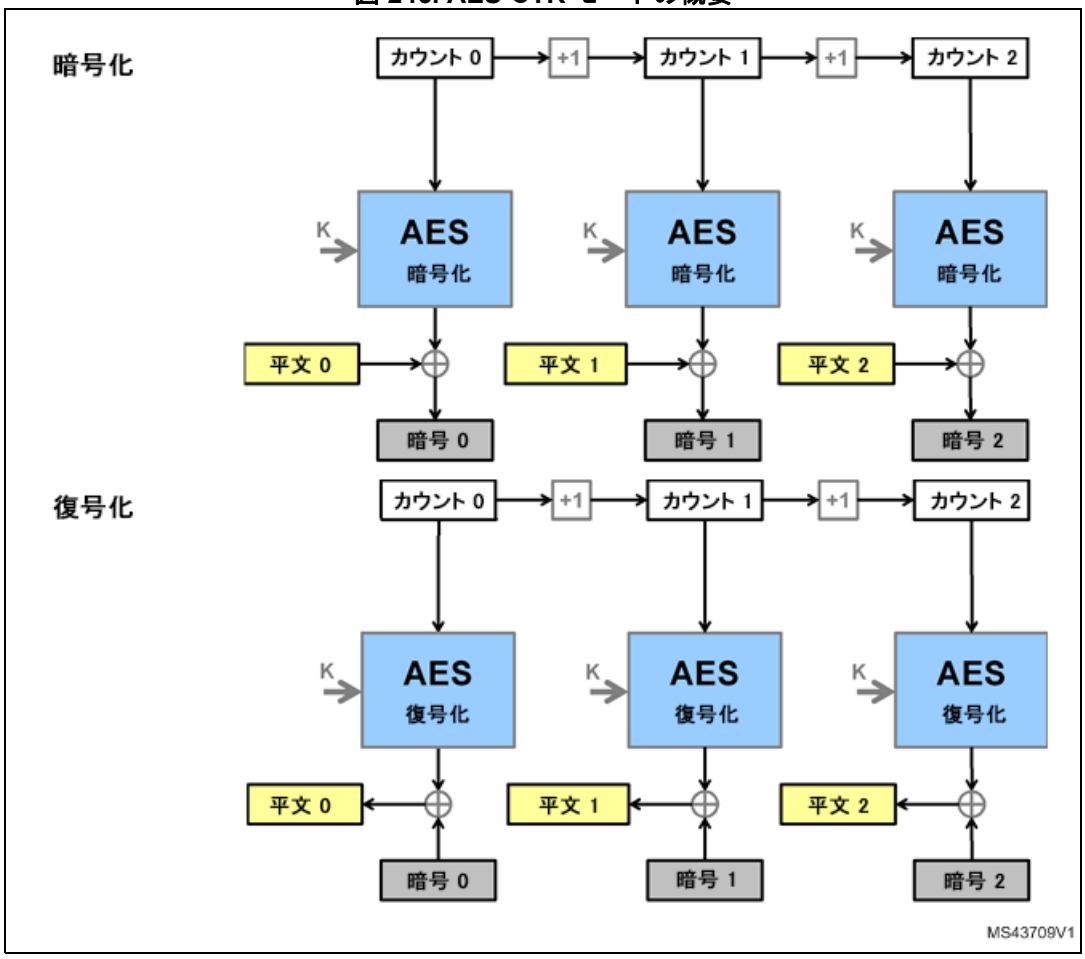


CBC動作モードでは、各ブロックの出力が次のブロックの入力に連鎖されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクタが使用されます。

注： ***復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。***

AES カウンタモード（CTR）

図 246. AES-CTR モードの概要

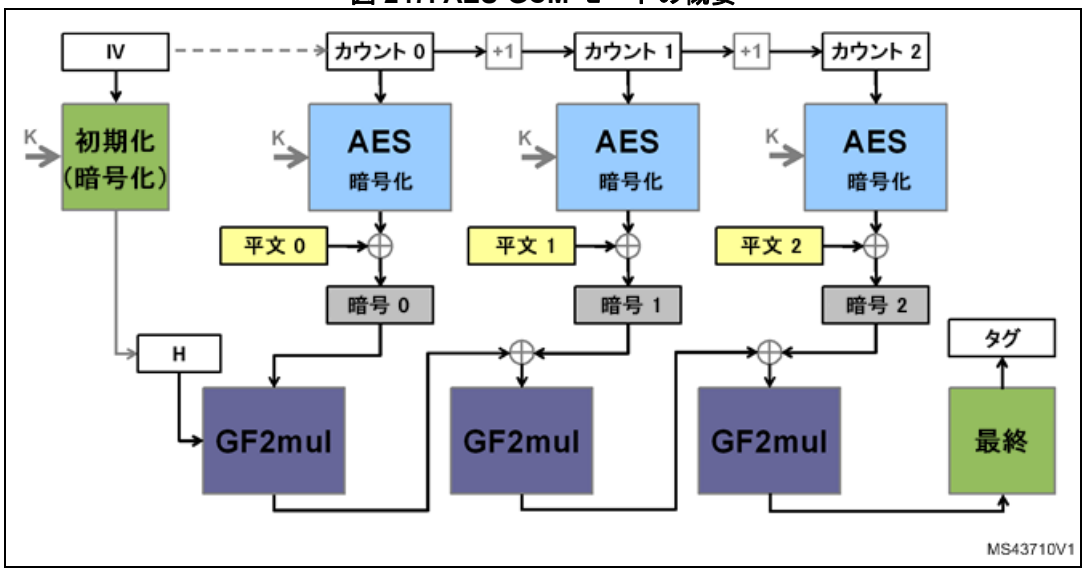


CTR モードでは、AES コアを使用してキーストリームを生成します。次に、これらのキーは、NIST *特別公報 800-38A『ブロック暗号の推奨動作モード』* で規定されている暗号文を得るために、平文との排他的論理和をとられます。

注： ***この連鎖スキームでは、AES コアはカウンタブロックの生成に必ず暗号化モードで使用されているため、ECB モードや CBC モードとは異なり、CTR 復号化にはキースケジューリングは必要ありません。***

AES ガロア／カウンタモード（GCM）

図 247. AES-GCM モードの概要

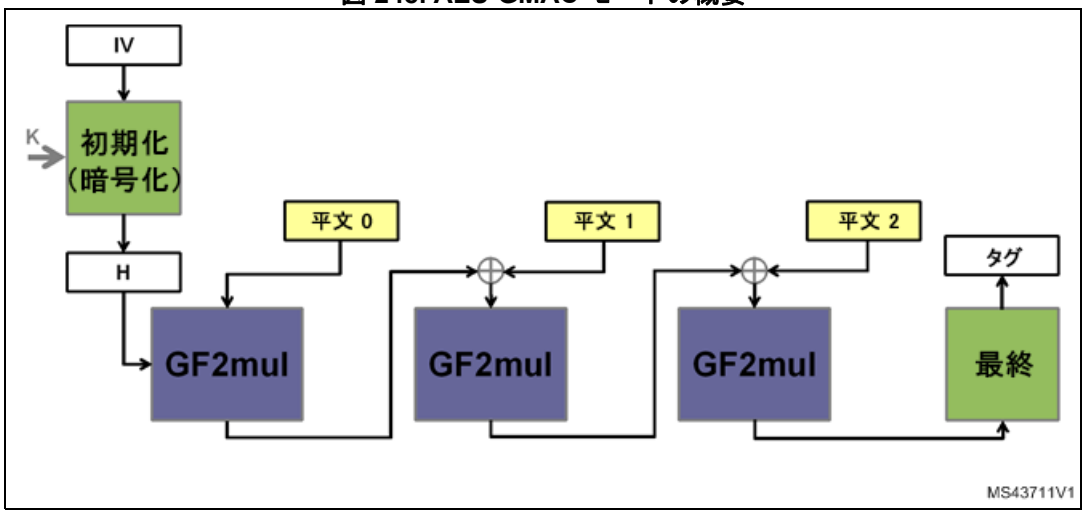


ガロア／カウンタモードでは、平文メッセ―ジが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文とその MAC（認証タグとも言います）が生成されます。これは、NIST *特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』* の中に定義されています。

GCM モードは、機密性については AES カウンタモードに基づいています。固定の有限体に乗算器を使用して、メッセージ認証コードを計算します。初期値ならびにメッセージの最後に特殊な 128 ビットブロックを必要とします。

AES ガロアメッセージ認証コード（GMAC）

図 248. AES-GMAC モードの概要

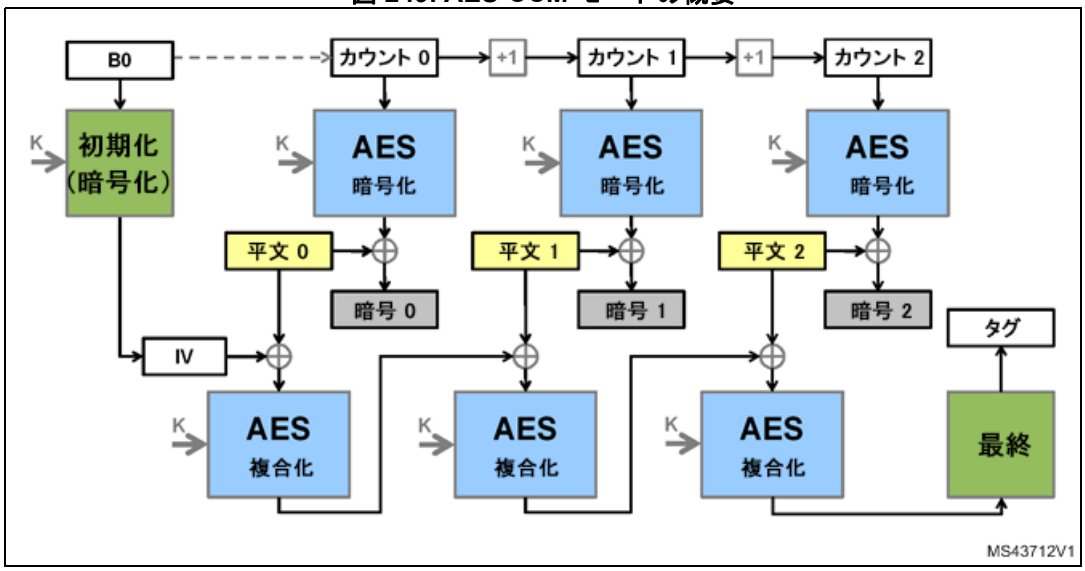


ガロアメッセージ認証コード（GMAC）を使用すると、メッセージの認証と、対応するメッセージ認証コード（MAC）の生成が可能となります。これは、NIST *特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』*の中に定義されています。

平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみ可でペイロードは不可）に適用されることを除くと、GMAC はガロア／カウンタモード（GCM）と似ています。

AES CBC-MAC 付き カウンタ （CCM）

図 249. AES-CCM モードの概要



暗号ブロック連鎖-メッセージ認証コード付きカウンタ（CCM）では、平文メッセ―ジが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文と対応する MAC（タグとも言います）が生成されます。これは、NIST *特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のためのCCMモード』*の中に説明されています。

CCM モードは、機密性のためにカウンタモードの AES に基づいており、CBC を使用してメッセージ認証コードを計算します。初期値が必要です。

GCM CCM 連鎖モードと同様に、AES-CCM モードは、平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみ可でペイロードは不可）に適用できます。CCM をこのように使用することは CMAC とは呼ばれず（GCM/GMACとは異なります）、その使用方法は NIST による推奨ではないことに注意してください。

35.3.5 暗号操作を行うための CRYP の手順

概要

暗号ペリフェラルがどのように動作するか理解するために、標準的な暗号操作を次に示します。暗号モードによるペリフェラルの使用方法の詳細は、*セクション 35.3.11：CRYP AES基本連鎖モード（ECB、CBC）*など個別の項目を参照してください。

図 250 と*図 251* のフローチャートに、STM32 暗号ライブラリに DES と AES のアルゴリズムが実装されている方法をそれぞれ示します。暗号プロセッサによって、次の暗号アルゴリズムの実行が高速化されます。

- 次のモードの AES-128、AES-192、AES-256：ECB、CBC、CTR、CCM、GCM
- 次のモードの DES とトリプルDES：ECB、CBC

注： ***暗号ライブラリの詳細については、ユーザマニュアル UM1924『STM32 暗号ライブラリ』を参照してください（www.st.com からダウンロード可能）。***

図 250. STM32 暗号ライブラリ DES/トリプル DES フローチャート

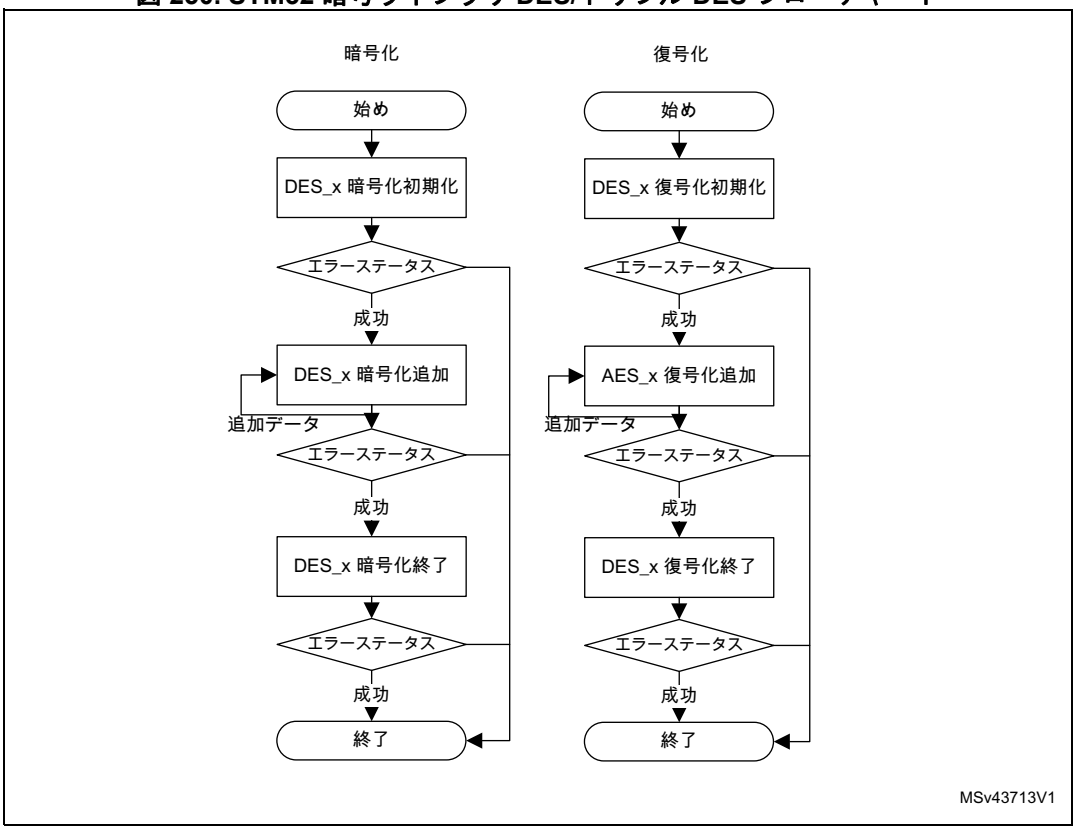
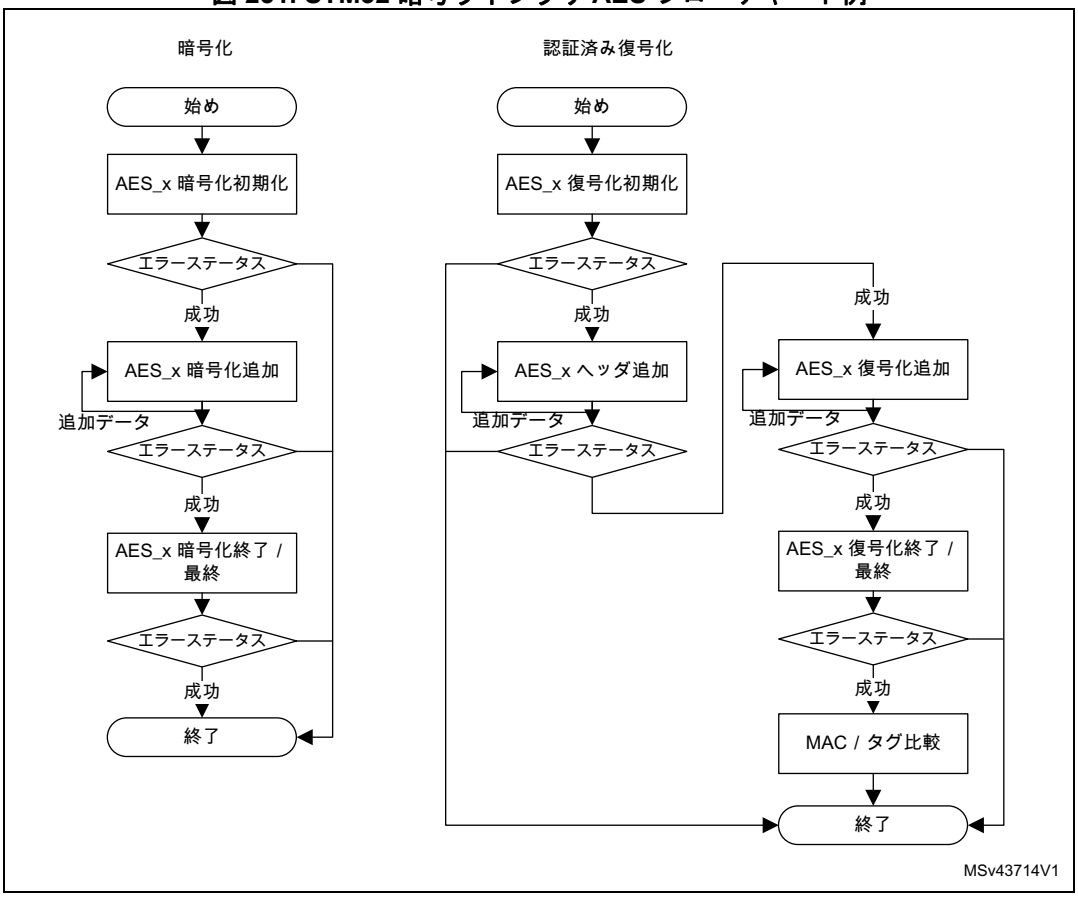


図 251. STM32 暗号ライブラリ AES フローチャート例



CRYP の初期化

1. 暗号プロセッサを初期化します。キーの準備に特別なシーケンスが必要な AES-ECB と AES-CBC の復号化を除いて、操作順序は重要ではありません。
   a) CRYP_CR レジスタの CRYPEN ビットを 0 にセットして、暗号プロセッサを無効にします。
   b) CRYP_CR レジスタの KEYSIZE ビットでキーサイズ（AES の場合にのみ 128、192、256 ビット）を設定します。
   c) 対称キーを CRYP_KxL/R レジスタに書き込みます（アルゴリズムによって書き込まれるレジスタは 2 個から 8 個と異なります）。
   d) CRYP_CR レジスタの DATATYPE ビットでデータ型（1、8、16、32 ビット）を設定します。
   e) AES-ECB モードまたは AES-CBC モードでの復号化の場合、書き込まれているキーを準備します。最初に、CRYP_CR レジスタで ALGOMODE ビットを 0b111 にセットしてキー準備モードを設定します。次に、CRYPEN ビットに 1 を書き込みます。これで BUSY ビットが自動的にセットされます。BUSY が 0 に戻るまで待ちます（CRYPEN も自動的にクリアされます）。これで復号化用にキーが準備できました。
   f) CRYP_CR レジスタの ALGOMODE ビットでアルゴリズムと連鎖を設定します。
   g) CRYP_CR レジスタの ALGODIR ビットで方向（暗号化／復号化）を設定します。
   h) 必要であれば（CBC または CTR の連鎖モードなど）、初期化ベクタを CRYP_IVxL/R レジスタに書き込みます。
2. CRYP_CR レジスタで FFLUSH ビットに 1 を書き込み、IN および OUT FIFO を一掃します。

すべての場合に対する事前警告

ECB モードか CBC モードが選択されて、データが 64 ビット（DESの場合）または 128 ビット（AESの場合）の倍数ではない場合には、2 番目と最後のブロック管理は、次のシーケンスよりも複雑になります。詳細については、*セクション 35.3.8 : CRYP スチーリングおよびデータパディング* を参照してください。

CPU をポーリングモードで使用したデータの追加

1. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして、暗号プロセッサを有効にします。
2. データを IN-FIFO に書き込みます（1 ブロックまたは FIFO がフルになるまで）。
3. 最後から 2 番目のデータのブロックが処理されるまで次のシーケンスを繰り返します。
   a) ノットエンプティフラグ OFNE が 1 にセットされるのを待ってから、OUT-FIFO を読み出します（1 ブロックまたは FIFO が空になるまで）。
   b) それが最後のブロックでなければ、ノットフルフラグ IFNF が 1 にセットされるのを待ってから、IN-FIFO を読み出します（1 ブロックまたは FIFO がフルになるまで）。
4. 暗号プロセッサによって BUSY ビットが自動的にセットされます。処理の最後には、BUSY ビットは 0 に戻り、どちらの FIFO も空になります（IN-FIFO エンプティフラグ IFEM=1 および OUT-FIFOノットエンプティフラグ OFNE=0）。
5. 次に処理するブロックが最終ブロックである場合には、CPU はデータをゼロで埋めて（該当する場合）完全なブロックとする必要があります。
6. 操作が完了したら、CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサを無効化できます。

CPU を割り込みモードで使用したデータの追加

1. CRYP_IMSCR レジスタの INIM ビットと OUTIM ビットをセットして、割り込みを有効にします。
2. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして、暗号プロセッサを有効にします。
3. 入力データを管理する割り込みサービスルーチンにおいて：
   a) 最終ブロックがロード中である場合には、CPU はデータをゼロで埋めて（該当する場合）完全なブロックとする必要があります。次に、そのブロックを IN FIFO にロードします。
   b) 最終ブロックではない場合には、データを IN FIFO にロードします。1 ブロック（DES の場合は 2 ワード、AES の場合は 4 ワード）だけロードすることも、FIFO がフルになるまでデータをロードすることもできます。
   c) どの場合でも、データの最後のワードが書き込まれたら、INIM 割り込みマスクをクリアして割り込みを無効にします。
4. 入力データを管理する割り込みサービスルーチンにおいて：
   a) 出力データを OUT FIFO から読み出します。1 ブロック（DES の場合は 2 ワード、AES の場合は 4 ワード）だけ読み出すことも、FIFO が空になるまでデータを読み出すこともできます。
   b) 最後のワードが読み出されると、INIM ビットと BUSY ビットは 0 にセットされ、どちらの FIFO も空になります（IFEM=1 および OFNE=0）。OUTIM ビットをクリアすることで割り込みを無効にでき、CRYPEN ビットをクリアすることでペリフェラルを無効にできます。
   c) 平文データ（復号時）の最終ブロックの読み出しなのであれば、必要に応じて、メッセージ／ペイロードの一部ではないデータを破棄します。

DMA を使用したデータの追加

1. 必要に応じてデータをゼロで埋めて完全なブロックとして、データの最終ブロックを用意します。
2. *セクション 35.3.19：CRYP DMA インタフェース* に示したように、入力データをメモリから転送し、出力データをペリフェラルからメモリに転送するように DMA コントローラを設定します。処理の完了を示すため、転送完了時に割り込みをセットするように DMA を設定します。
3. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして暗号プロセッサを有効にし、CRYP_DMACR レジスタの DIEN ビットおよび DOEN ビットを 1 にセットして、DMA IN リクエストと DMA OUT リクエストを有効にします。
4. すべての転送と処理は、DMA および暗号プロセッサによって管理されます。DMA 割り込みが、処理の完了を示します。どちらの FIFO も通常は空で BUSY フラグは 0 に設定されています。

注意： ***暗号プロセッサ入力 FIFO が一杯になる前に、DMA コントローラが暗号プロセッサ出力 FIFO を空にすることが重要です。このため、メモリから暗号ペリフェラルへの転送の優先度よりも、暗号ペリフェラルからメモリへの転送の優先度の方が高くなるように、DMA コントローラを設定してください。***

35.3.6 CRYP ビジーステート

下記の条件がすべて満たされると、暗号プロセッサはビジー状態でデータ処理中（CRYP_SR レジスタの BUSY が 1 にセット）となります。

- CRYP_CR レジスタでCRYPEN = 1。
- 入力 FIFO に十分なデータがある（DES／トリプル DES アルゴリズムモードでは 2 ワード以上、AES アルゴリズムモードでは 4 ワード以上）。
- 出力 FIFO に十分な空き空間がある（DESでは 2 ワード以上、AES では 4 ワード以上の位置）。

暗号プロセッサがビジーの場合（レジスタが未変更）、CRYP_Kx（L/R）R キーレジスタ、CRYP_IVx（L/R）R 初期化レジスタや CRYP_CR レジスタのビット [9:2] への書き込み操作は無視されます。そのため、暗号プロセッサがデータブロックを処理している間は、設定を変更することはできません。

BUSY ビットが 1 にセットされている間に、CRYPEN ビットはクリアできます。この場合、ハードウェアによって BUSY ビットがクリアされる前に、まずは進行中のDES/トリプル DES または AES の処理が完了します（結果ワードが出力 FIFO に書き込まれます）。

注： ***アプリケーションがメッセージをサスペンドして、より優先順位の高い別のメッセージを処理する必要がある場合には、セクション 35.3.9：CRYP サスペンド／レジューム操作を参照してください。***

ブロックが DES またはトリプルDES モードで処理される場合、出力 FIFO が一杯になり、入力 FIFO に 1 つ以上の新しいブロックが含まれている場合は、新しいブロックが入力 FIFO からポップされますが、この新しいブロックを出力 FIFO に保存できる十分なスペースができるまで、BUSY ビットはハイのままになります。

35.3.7 CRYP AES キーを復号化用に準備

AES **ECB または CBC 復号化**を実行するときには、AES キーを用意する必要があります。すなわち、復号化を行う前に、暗号化の完全なキースケジュールが必要です。言い換えれば、暗号化の最終ラウンドのキーを、復号化の初回ラウンドのキーとして使用する必要があります。

この準備は、AES ECB または CBC 復号化以外の AES モードでは不要です。

アプリケーションソフトウェアが復号用に準備された最初のキーを格納しているのであれば、所定の暗号キーで復号化されるすべてのデータに対して、キースケジューリング操作を 1 回だけ行うことができます。

注： ***キー準備操作のレイテンシは、キーサイズ（128、192、または256 ビット）次第で 14、16、または 18 クロックサイクルとなります。***

CRYP キーの準備プロセスは、次のように行われます。

1. 暗号化キーを K0～K3 キーレジスタに書き込みます。
2. ALGOMODE ビットを CRYP_CR の 0b111 にプログラムします。CRYPEN = 1 にセットされている場合、この値を書き込むと、ただちにキー準備用の AES ラウンドが開始となります。CRYP_SR レジスタの BUSY ビットは 1 にセットされます。
3. キー処理が完了すると、その結果生成されるキーが再び K0～K3 キーレジスタにコピーされ、BUSY ビットがクリアされます。

注： ***キーの準備の最後にCRYPEN ビットフィールドがハードウェアによりリセットされますので、アプリケーションソフトウェアは次の操作のために再セットする必要があります。***

35.3.8 CRYP スチーリングおよびデータパディング

ブロックサイズ（DES では 64 ビット、AES では 128 ビット）の倍数ではないメッセージを管理するために **ECB** モードまたは **CBC** モードで DES または AES のアルゴリズムを使用するときには、NIST *特別公報 800-38A『ブロック暗号の推奨動作モード：CBC モード用暗号文借用の 3 つの変種』*に記載されているもののような、暗号文借用技術を使用します。暗号プロセッサではこのような技術が実現されないため、**最後の 2 ブロック**はアプリケーションによって特別な方法で処理される必要があります。

注： ***暗号文借用技術は、この参考マニュアルには記載されていません。***

同様に、ECB と CBC 以外のモードで AES アルゴリズムが使用される場合には、不完全な入力データブロック（128 ビットより短いブロック）に対して、暗号化の前にアプリケーションがゼロでパディングする必要があります（データ列の後端に余分なビットを付加する必要があります）。復号後は、余分なビットを破棄する必要があります。暗号プロセッサでは、**最終ブロックに対する**自動データパディング操作が行われないため、アプリケーションが*セクション 35.3.5：暗号操作を行うための CRYP の手順*に示された推奨に従って、128 ビットの倍数ではないメッセ―ジの管理を行う必要があります。

注： ***パディングデータは、CRYP_CR レジスタの DATATYPE フィールドに従って、通常のデータと同様な方法で入れ替えられます（詳細は、セクション 35.3.16：CRYP データレジスタおよびデータスワッピングを参照してください）。***

このバージョンの暗号プロセッサでは、**128 ビットよりも小さい**ペイロードサイズの最終ブロックに対して **GCM 暗号化**または **CCM 復号化**を行いながら、認証タグを適切に計算するためには、特別なワークアラウンドが必要です。このワークアラウンドを以下に説明します。

- GCM 暗号化ペイロードフェーズの間で、128 ビットよりも小さい最後の平文ブロックを挿入する前に、アプリケーションは次のシーケンスに従う必要があります。
  a) CRYP_CR の CRYPEN ビットを 0 にセットして、ペリフェラルを無効にします。
  b) CRYP_IV1R レジスタの内容を一時変数にロードします。値を 1 だけ減らして、その結果を CRYP_IV1R レジスタに再挿入します。
  c) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b0110 を書き込んで、AES モードを CTR モードに変更します。
  a) CRYP_CR で CRYPEN ビットを“1”にセットして、ペリフェラルを再び有効にします。
  b) 最終ブロック（128 ビットよりも小さいもの）にゼロをパディングして 128 ビットの完全なブロックとし、そのデータを CRYP_DIN レジスタに書き込みます。
  c) 暗号化が完了したら、生成された 128 ビット暗号文を CRYP_DOUT レジスタから読み出して、中間データとして格納します。
  d) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b1000 を書き込んで、AES モードを GCM モードに再度変更します。
  e) CRYP_CR レジスタで GCM_CCMPH ビットフィールドに 0b11 を書き込んで、最終フェーズを選択します。
  f) 中間データの中で、最後のペイロードブロックのパディングされたビットに対応するビットを 0 にセットし、そこで得られた結果を CRYP_DIN レジスタに挿入します。
  g) 操作が完了したら、CRYP_DOUT からデータを読み出します。これらのデータは破棄する必要があります。
  h) *セクション 35.3.13：CRYP AES ガロア／カウンタモード（GCM）*に記載のようにして、通常の最終フェーズを適用します。

- CCM 復号化ペイロードフェーズの間で、128 ビットよりも小さい最後の暗号文ブロックを挿入する前に、アプリケーションは次のシーケンスに従う必要があります。
  a) ペリフェラルを無効とするために、CRYP_CR で CRYPEN ビットを“0”にセットします。
  b) CRYP_IV1R を一時変数（ここでの名前は `IV1temp` とします）にロードします。
  c) CRYP_CSGCMCCM0R、CRYP_CSGCMCCM1R、CRYP_CSGCMCCM2R、および CRYP_CSGCMCCM3R のレジスタの内容を、LSB から MSB の順番で 128 ビットの一時変数（ここでの名前は `temp1` とします）にロードします。
  d) 前に `IV1temp` に格納した内容を CRYP_IV1R にロードします。
  e) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b0110 を書き込んで、AES モードを CTR モードに変更します。
  a) CRYP_CR で CRYPEN ビットを“1”にセットして、ペリフェラルを再び有効にします。
  b) 最終ブロック（128 ビットよりも小さいもの）にゼロをパディングして 128 ビットの完全なブロックとし、そのデータを CRYP_DIN レジスタに書き込みます。
  c) 復号化が完了したら、生成された 128 ビットデータを DOUT レジスタから読み出して、中間データ（ここでの名前は `intdata_o` とします）として格納します。
  d) CRYP_CSGCMCCM0R、CRYP_CSGCMCCM1R、CRYP_CSGCMCCM2R、および CRYP_CSGCMCCM3R のレジスタの内容を、LSB から MSB の順番で 128 ビットの新しい一時変数（ここでの名前は `temp2` とします）に再び保存します。
  e) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b1001 を書き込んで、AES モードを CCM モードに再度変更します。

f) CRYP_CR レジスタで GCM_CCMPH ビットフィールドに 0b01 を書き込んで、ヘッダフェーズを選択します。

g) 中間データ（CTR で生成された intdata_o）の中で、最後のペイロードブロックのパディングされたビットに対応するビットを 0 にセットし、temp1 で排他的論理和をとってから temp2 で排他的論理和をとり、そこで得られた結果を CRYP_DIN レジスタに挿入します。言い換えれば、以下の式のようになります。

```
CRYP_DIN=(intdata_o AND mask) XOR temp1 XOR temp2.
```

h) 操作が完了するのを待ちます。

i) *セクション 35.3.15：CRYP AES CBC-MAC 付き カウンタ（CCM）*に記載のようにして、通常の最終フェーズを適用します。

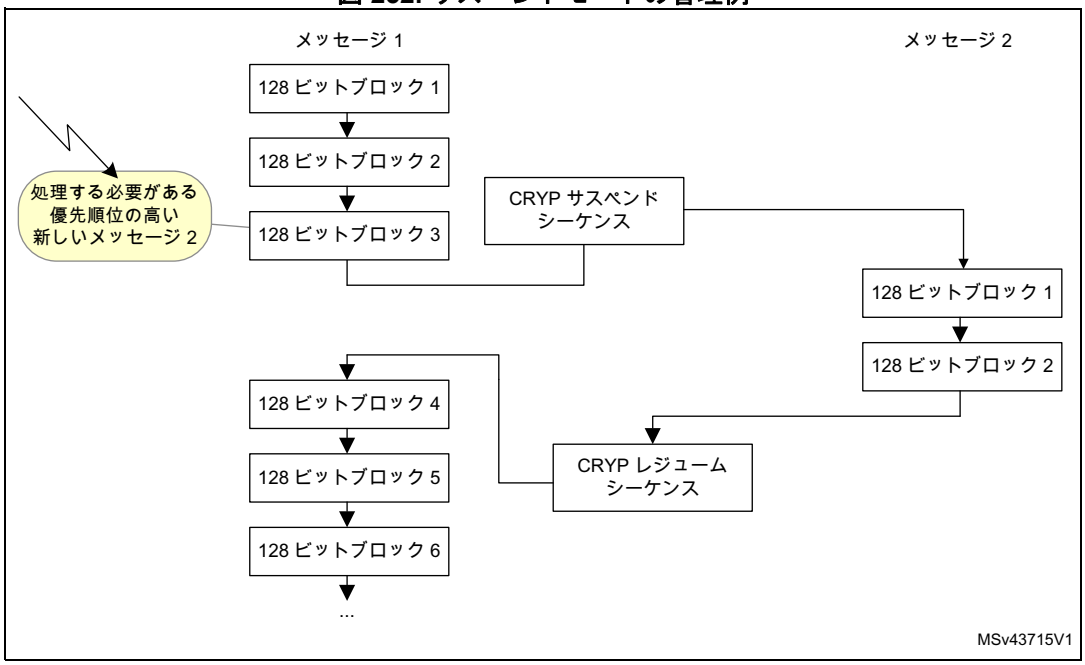
35.3.9 CRYP サスペンド／レジューム操作

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージは暗号化モードでも復号化モードでも再開できます。

サスペンド／レジューム操作によって連鎖操作が途切れることはなく、暗号プロセッサが再び有効になるとすぐにメッセージ処理を再開して、次のデータブロックを受信できます。

図 252 にサスペンド／レジューム操作の例を示します。ここでは、メッセージ 1 は、優先順位が高くてメッセージ 1 よりも短いメッセージ（メッセージ 2）を送信するためにサスペンドされています（AES アルゴリズム）。

図 252. サスペンドモードの管理例



サスペンド／レジューム操作の詳細は、AES モードの各セクションに説明があります。

35.3.10 CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）

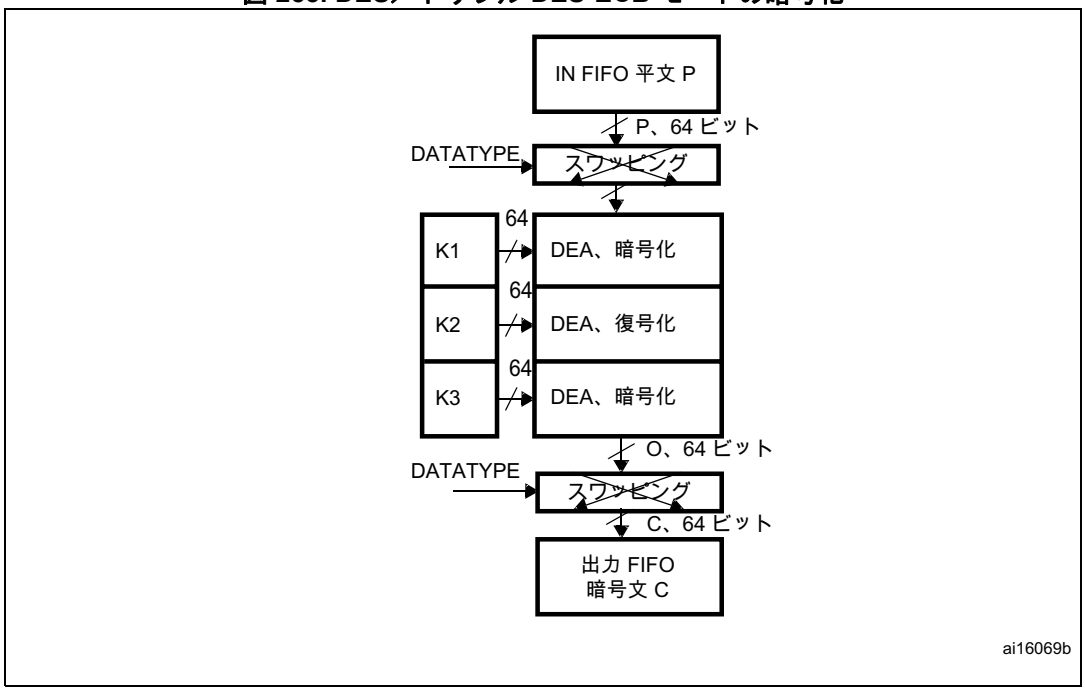
概要

FIPS PUB 46-3 – 1999（および ANSI X9.52-1998）では、DES 計算コアが提供するトリプル DES-ECB 暗号化、トリプル DES-ECB 復号化、トリプル DES-CBC 暗号化、トリプル DES-CBC 復号化の 4 つの動作モードに関する処理について詳細に説明しています。このセクションでは、各モードについての簡単な説明としています。

DES/トリプルDES-ECBの暗号化

図 253 に、DES およびトリプル DES 電子コードブック（DES／トリプル DES-ECB）モードにおける暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b000 を、ALGODIR に 0 を書き込むことによって選択されます。

図 253. DES／トリプル DES-ECB モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文

64 ビットの平文データブロック（P）は、ビット／バイト／ハーフワードスワッピングの後に、入力ブロック（I）として使用されます。入力ブロックは、K1 を使った暗号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K2 を使った復号ステートで DES が行われます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K3 を使った暗号ステートで DES が行われます。ビット／バイト／ハーフワードのスワッピング後は、その結果の 64 ビットの出力ブロック（O）が暗号文（C）として使用され、出力 FIFO に入ります。

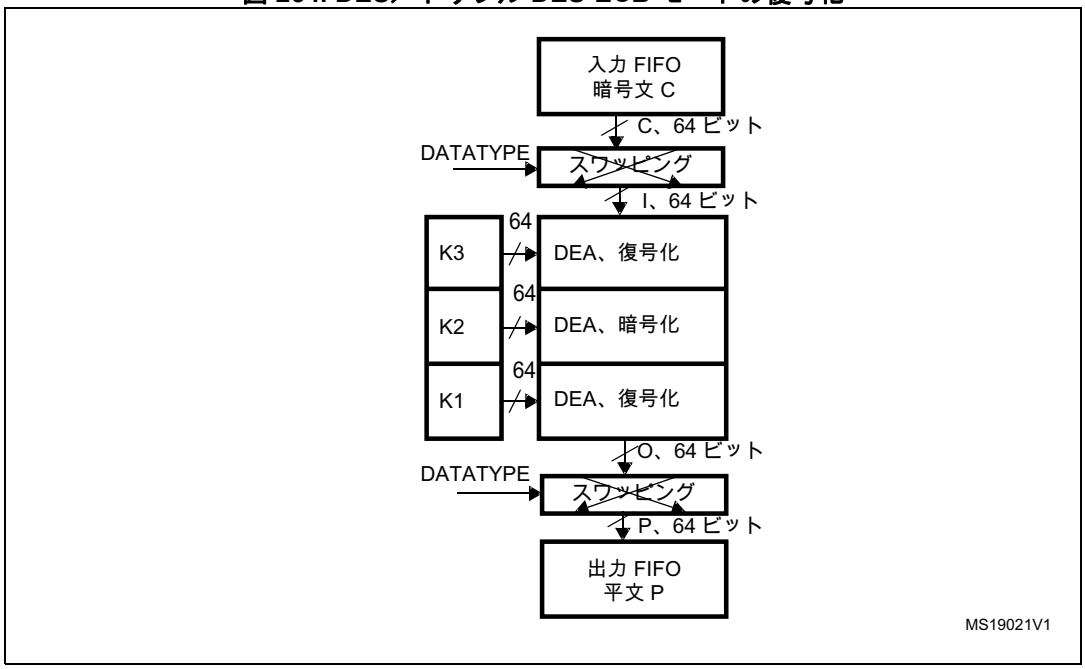
注： ***データスワッピングの詳細については、セクション 35.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。***

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 35.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-ECB モードの復号化

図 254 に、DES およびトリプル DES 電子コードブック（DES／トリプル DES-ECB）モードにおける復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b000 を、ALGODIR に 1 を書き込むことによって選択されます。

図 254. DES／トリプル DES-ECB モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文

ビット / バイト / ハーフワードのスワッピング後は、64 ビットの暗号文ブロック（C）を入力ブロック（I）として使用します。このキーイングシーケンスは、暗号処理で使用したものに対し、逆になっています。入力ブロックは、K3 を使った復号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K2 を使った暗号ステートで DES が行われます。新しい結果は、DEA の入力に直接されます。ここでは、K1 を使った復号ステートで DES が実施されます。ビット / バイト / ハーフワードのスワッピング後は、その結果の 64 ビットの出力ブロック（O）が平文（P）を生成します。

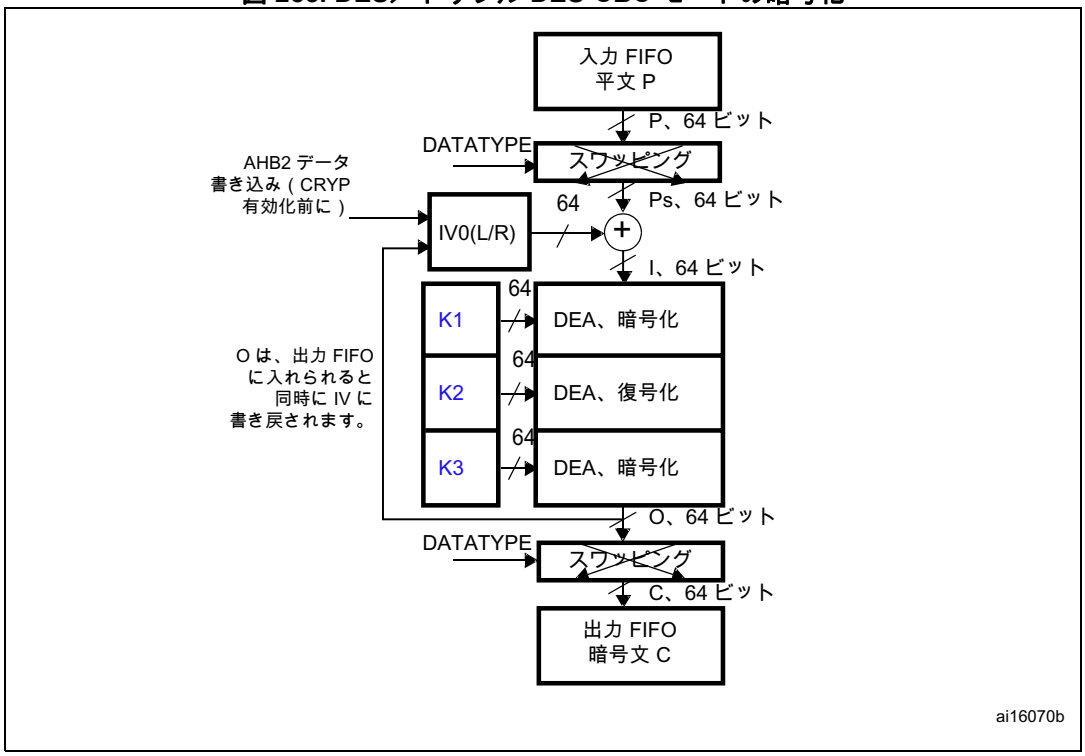
注： ***データスワッピングの詳細については、セクション 35.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。***

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 35.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-CBCの暗号化

図 255 に、DES およびトリプル DES 暗号ブロック連鎖（DES／トリプル DES-ECB）モードにおける暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b001 を、ALGODIR に 0 を書き込むことによって選択されます。

図 255. DES／トリプル DES-CBC モードの暗号化



K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ

このモードは、まず平文のメッセージを 64 ビットのデータブロックに分割することから開始します。TCBC 暗号化では、ビット／バイト／ハーフワードのスワッピング後に取得した最初の入力ブロック（I_1）は、最初の平文データブロック（P_1）と 64 ビットの初期化ベクタ IV（I_1 = IV ?P_1）の排他的論理和をとって形成します。入力ブロックは、K1 を使った暗号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。このとき K2 を使った復号ステートで DES が行われます。この処理の出力は、DEA の入力に直接フィードバックされます。このとき K3 を使った暗号ステートで DES が行われます。その結果の 64 ビットの出力ブロック（O_1）は暗号文（C_1）として直接使用されます。つまり、C_1 = O_1 となります。

この最初の暗号文ブロックは、その後 2 番目の平文データブロックと排他的論理和をとって 2 番目の入力ブロック（I_2）=（C_1？P_2）を生成します。I_2 と P_2 が今度は 2 番目のブロックを参照していることに注意してください。2 番目の入力ブロックは TDEA で処理され、2 番目の暗号文ブロックを生成します。

この暗号化処理は、メッセージ内の最後の平文ブロックが暗号化されるまで次の暗号ブロックと平文ブロックを連結し続けます。

メッセージが整数のデータブロックで構成されていない場合は、最後の不完全なデータブロックがアプリケーションで指定されている方法で暗号化されます。

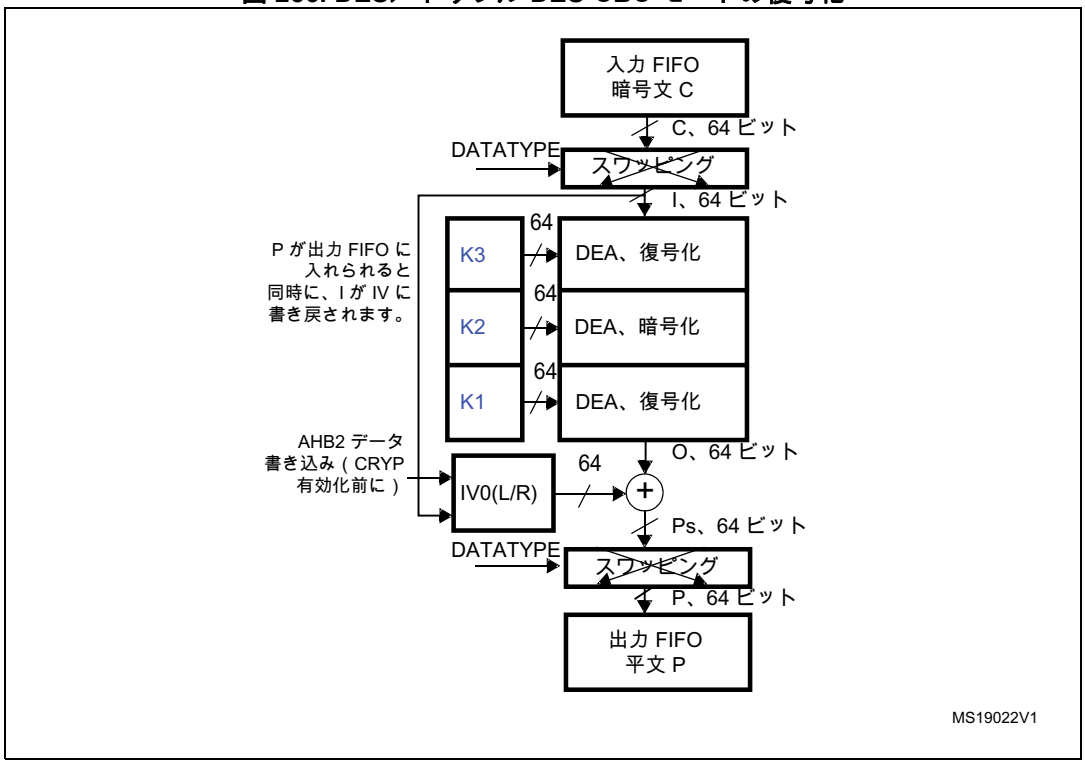
注： ***データスワッピングの詳細については、セクション 35.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。***

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 35.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-CBCの復号化

図 255 に、DES およびトリプル DES 暗号ブロック連鎖（DES／トリプル DES-ECB）モードにおける復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b001 を、ALGODIR に 1 を書き込むことによって選択されます。

図 256. DES／トリプル DES-CBC モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ

このモードでは、最初の暗号文ブロック（C_1）をそのまま入力ブロック（I_1）として使用します。キーイングシーケンスは、暗号処理で使用したものに対し、逆になっています。入力ブロックは、K3 を使った復号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接されます。このとき、K2 を使った暗号ステートで DES が処理されます。これによって生じる値は、DEA の入力に直接供給されます。このとき、K1 を使った復号ステートで DES が処理されます。その結果の出力ブロックと IV（暗号化で使用したのと同じである必要があります）の排他的論理和をとって最初の平文ブロック（$P_1 = O_1 \oplus IV$）を生成します。

その後 2 番目の暗号文ブロックが次の入力ブロックとして使用され、TDEA で処理されます。その結果の出力ブロックと最初の暗号文ブロックの排他的論理和をとって 2 番目の平文データブロック（$P_2 = O_2 \oplus C_1$）を生成します。P_2 と O_2 が 2 番目のデータブロックを参照していることに注意してください。

DES／トリプルDES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

不完全なデータブロックを表す暗号文は、アプリケーションで指定された方法で復号化する必要があります。

注： ***データスワッピングの詳細については、セクション 35.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。***

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 35.3.5：暗号操作を行うための CRYP の手順を参照してください。

ECB/CBC モードにおける DES／トリプルDES サスペンド／レジューム動作

現在のメッセージに割り込む前に、ユーザアプリケーションは次の手順に従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR で IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。あるいは、入力 FIFO には最大 4 つの未処理の DES ブロックを収容可能であるため、アプリケーションは、リアルタイム性のために、IN FIFO が空になるのを待たずに暗号化処理を中断することもできます。この場合には、代わりに次の処理を行います。
   a) OUT FIFO が空になる（OFNE=0）まで待ちます。
   b) 未処理の IN FIFO にロードされたデータを読み戻し、それらを FIFO が空になるまでメモリに保存します。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]）して、暗号プロセッサを無効にします。CBC モードが選択されている場合には、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： ***アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。***

5. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. CBC モードが選択されている場合には、保存されている設定を用いて CRYP_IVx レジスタを復元します。
5. 必要に応じて、コンテキストの保存中に保存されたデータを IN FIFO に書き込みます。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサに対する DMA リクエストを再び有効とします。

35.3.11 CRYP AES基本連鎖モード（ECB、CBC）

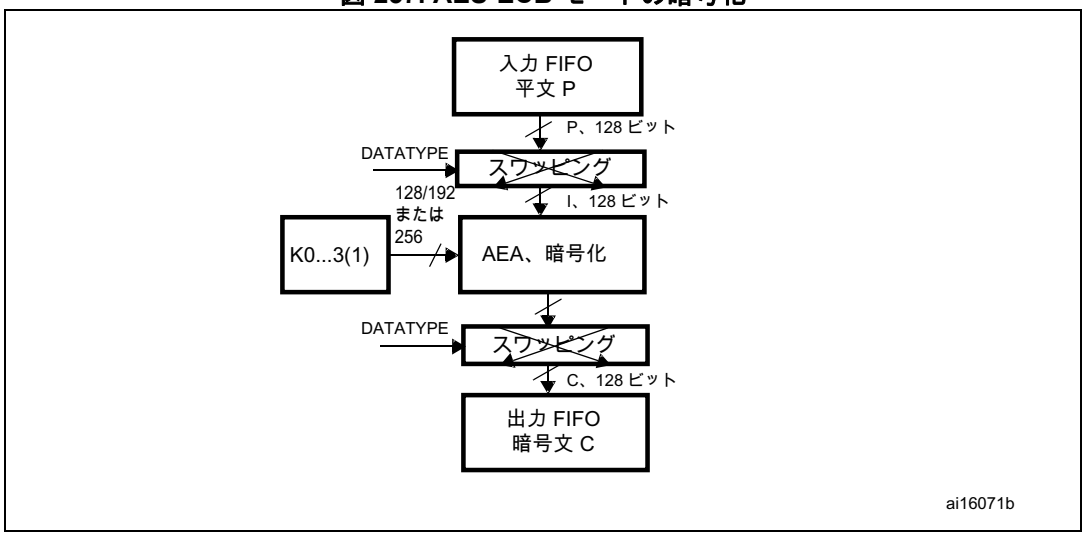
概要

FIPS PUB 197（2001年11月26日）では、AES 計算コアが提供するAES-ECB 暗号化、AES-ECB 復号化、AES-CBC 暗号化、AES-CBC 復号化の 4 つの基本動作モードに関する処理について詳細に説明しています。このセクションでは、各モードについての簡単な説明としています。

AES ECBの暗号化

図 257 に、AES 電子コードブック（AES-ECB）モードの暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b100 を、ALGODIR に 0 を書き込むことによって選択されます。

図 257. AES-ECB モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文
2. キーサイズ = 128 の場合：キー = [K3 K2]
   キーサイズ = 192 の場合：キー = [K3 K2 K1]
   キーサイズ = 256 の場合：キー = [K3 K2 K1 K0]

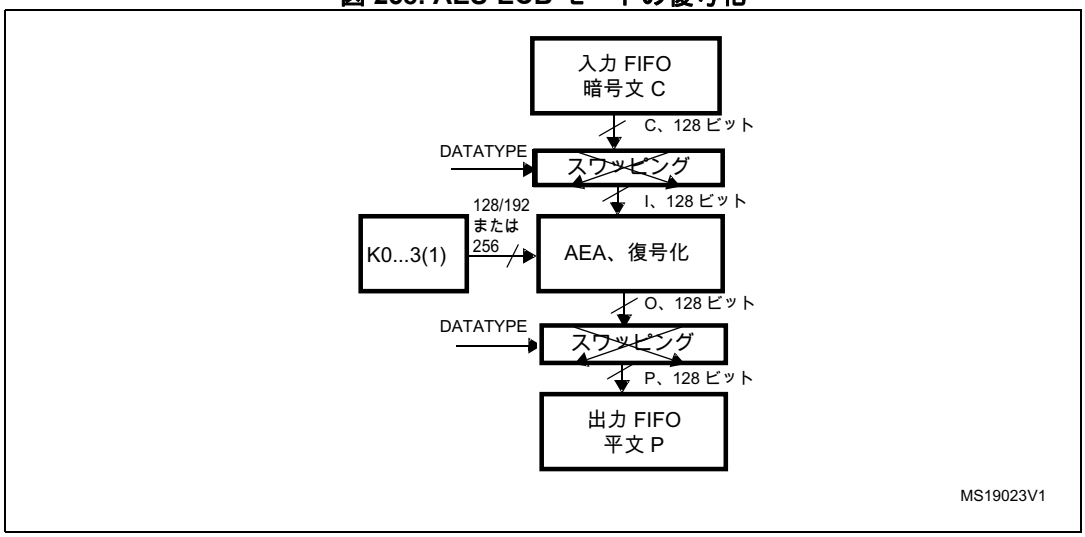
このモードでは、128 ビットの平文データブロック（P）は、ビット／バイト／ハーフワードのスワッピングの後に、入力ブロック（I）として使用されます。入力ブロックは、128、192、256 ビットキーを使った暗号ステートで AEA を通じて処理されます。ビット / バイト / ハーフワードのスワッピング後は、その結果の 128 ビットの出力ブロック（O）が暗号文（C）として使用されます。その後出力 FIFO に入れられます。

データスワッピングの詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング*を参照してください。

AES ECBの復号化

図 258 に、AES 電子コードブック（AES-ECB）モードの復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b100 を、ALGODIR に 1 を書き込むことによって選択されます。

図 258. AES-ECB モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文
2. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

ECB モードで AES 復号化を実施するには、最終ラウンドのキーを回収し、それを暗号文の復号化の最初のラウンドのキーとして使用して、秘密キーを準備する必要があります（暗号化のための完全なキースケジュールを実行する必要があります）。この準備フェーズは、AES コアで計算されます。キーを準備する方法の詳細については、*セクション 35.3.7：CRYP AES キーを復号化用に準備*を参照してください。

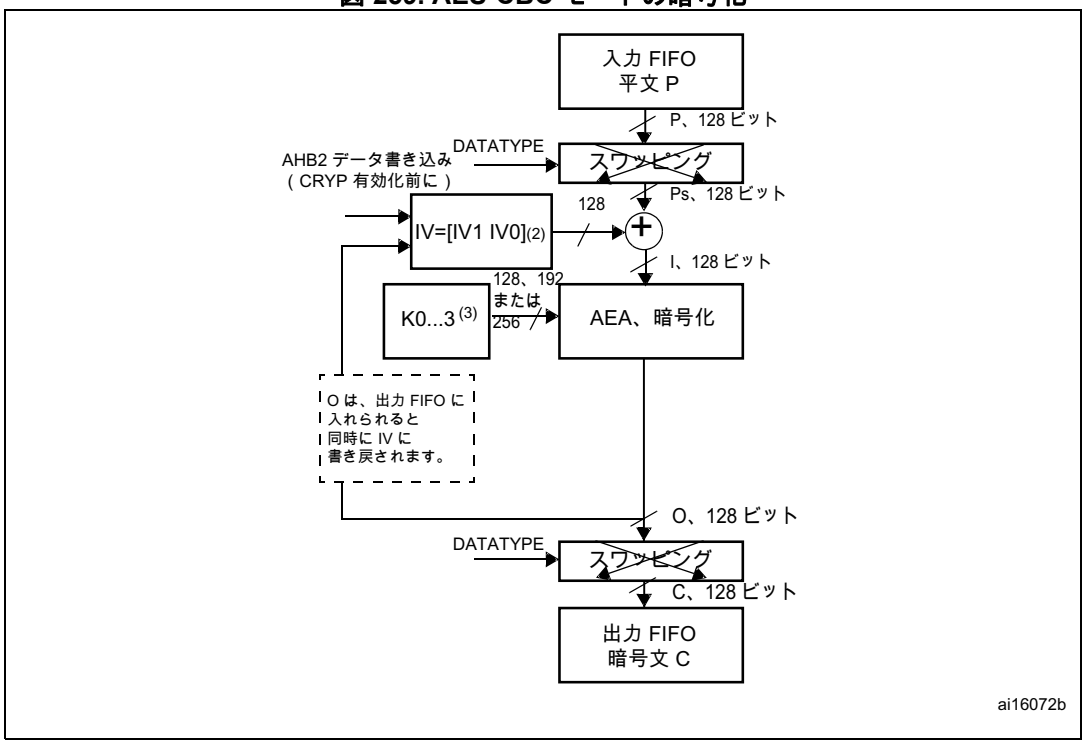
キーの準備が完了すると、復号化処理として、ビット / バイト / ハーフワードのスワッピング後、128 ビットの暗号文ブロック（C）が入力ブロック（I）として使用されます。このキーイングシーケンスは、暗号処理のものに対し、逆になっています。ビット／バイトまたはハーフワードのスワッピング後は、その結果の 128 ビットの出力ブロック（O）が平文（P）を生成します。AES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

データスワッピングの詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング*を参照してください。

AES CBCの暗号化

*図 259*に、AES 暗号ブロック連鎖（AES-CBC）モードの暗号化を示します。このモードは、CRYP_CR の ALGOMODE に 0b101 を、ALGODIR に 0 を書き込むことによって選択されます。

図 259. AES-CBC モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ
2. IVx=[IVxR IVxL]、R = 右、L = 左
3. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

このモードでは、ビット／バイト／ハーフワードのスワッピング後に取得した最初の入力ブロック（I_1）は、最初の平文データブロック（P_1）と 128 ビットの初期化ベクタ IV（$I_1 = IV \oplus P_1$）の排他的論理和をとって形成します。入力ブロックは、128、192、256 ビットキー（K0...K3）を使った暗号ステートで AEA を通じて処理されます。その結果の 128 ビットの出力ブロック（O_1）は暗号文（C_1）として直接使用されます。つまり、$C_1 = O_1$ となります。この最初の暗号文ブロックは、その後 2 番目の平文データブロックと排他的論理和をとって 2 番目の入力ブロック（I_2）=（$C_1 \oplus P_2$）を生成します。I_2 と P_2 が今度は 2 番目のブロックを参照していることに注意してください。2 番目の入力ブロックは AEA で処理され、2 番目の暗号文ブロックを生成します。この暗号化処理は、メッセージ内の最後の平文ブロックが暗号化されるまで次の暗号ブロックと平文ブロックを連結し続けます。

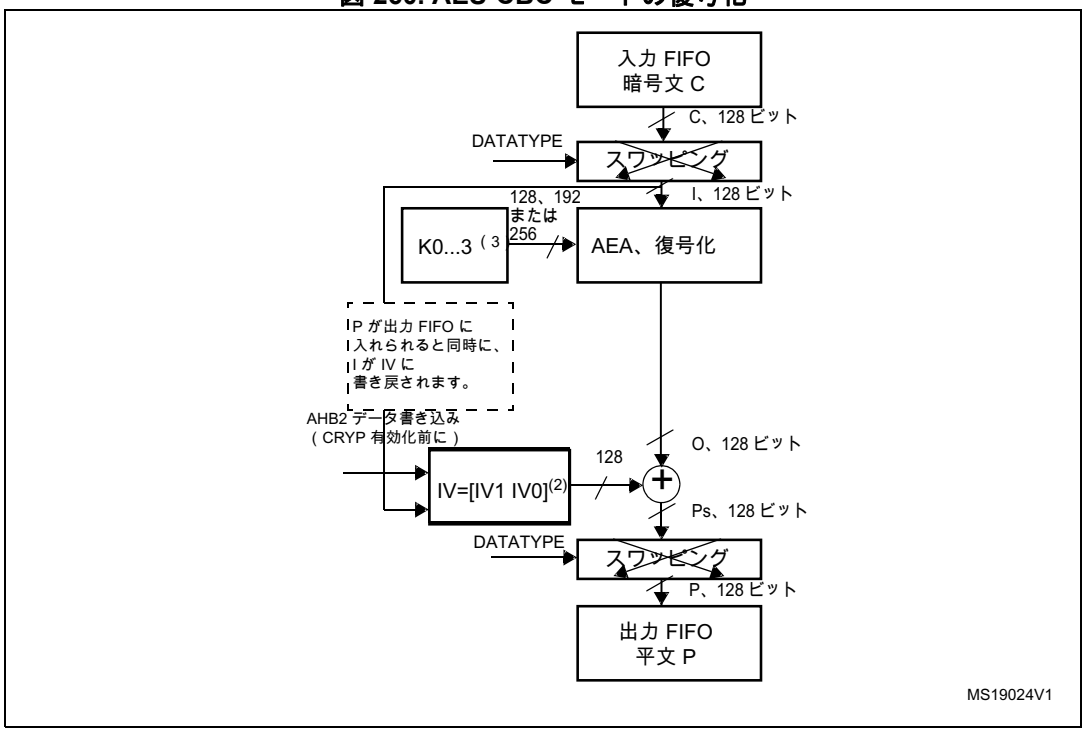
メッセージが整数のデータブロックで構成されていない場合は、*セクション 35.3.8 : CRYP スチーリングおよびデータパディング*に説明されているように、最後の不完全なデータブロックがアプリケーションで指定されている方法で暗号化されます。

データスワッピングの詳細については、*セクション 35.3.16 : CRYP データレジスタおよびデータスワッピング*を参照してください。

AES CBCの復号化

*図 260*に、AES 暗号ブロック連鎖（AES-CBC）モードの復号化を示します。このモードは、CRYP_CR の ALGOMODE に 0b101 を、ALGODIR に 1 を書き込むことによって選択されます。

図 260. AES-CBC モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ
2. IVx=[IVxR IVxL]、R = 右、L = 左
3. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

ECB モードの場合と同様に、CBC モードで AES 復号化を実施するには、秘密キーを準備する必要があります。キーを準備する方法の詳細については、*セクション 35.3.7：CRYP AES キーを復号化用に準備*を参照してください。

キーの準備が完了すると、復号化処理として、128 ビットの暗号文ブロック（C_1）が入力ブロック（I_1）として直接使用されます。入力ブロックは、128、192、256 ビットキーを使った復号ステートで AEA を通じて処理されます。その結果の出力ブロックと 128 ビットの初期化ベクタ IV（暗号化で使用したのと同じである必要があります）の排他的論理和をとって最初の平文ブロック（$P_1 = O_1 \oplus IV$）を生成します。

その後 2 番目の暗号文ブロックが次の入力ブロックとして使用され、AEA で処理されます。その結果の出力ブロックと最初の暗号文ブロックの排他的論理和をとって 2 番目の平文データブロック（$P_2 = O_2 \oplus C_1$）を生成します。P_2 と O_2 が 2 番目のデータブロックを参照していることに注意してください。AES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

不完全なデータブロックを表す暗号文は、*セクション 35.3.8：CRYP スチーリングおよびデータパディング*で説明されているように、アプリケーションで指定された方法で復号化する必要があります。

データスワッピングの詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング*を参照してください。

ECB/CBC モードにおける AES サスペンド／レジューム動作

現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR で IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]）して、CRYP を無効にします。ECB モードが選択されていない場合には、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： ***アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。***

5. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 に設定されている必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. *セクション 35.3.7：CRYP AES キーを復号化用に準備*に記載されているように、AES-ECB または AES-CBC 復号化の場合、キーをもう一度準備する必要があります。
5. ECB モードが選択されていない場合には、保存されている設定を用いて CRYP_IVx レジスタを復元します。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサからの DMA リクエストを再び有効とします。

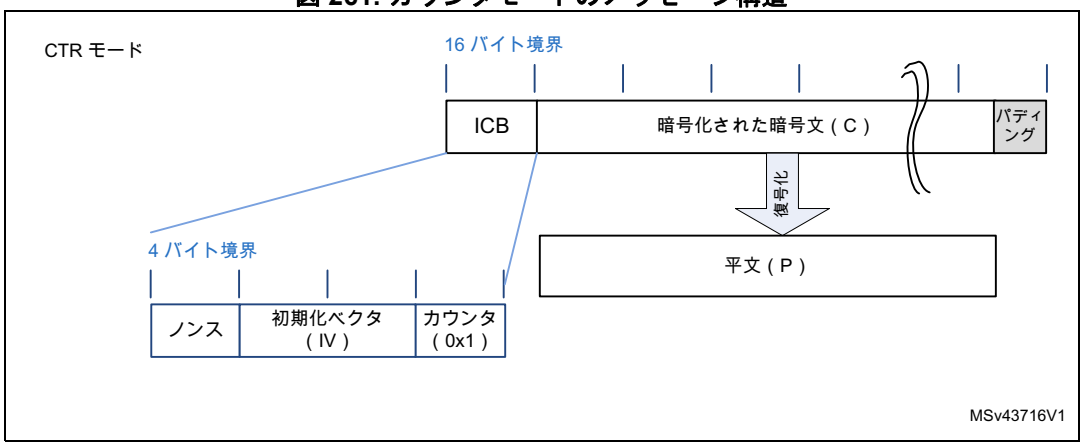
35.3.12 CRYP AES カウンタモード（AES-CTR）

概要

AES カウンタモード（CTR）では、AES ブロックをキーストリームジェネレータとして使用します。生成されたキーは、その後平文との排他的論理和をとって暗号文を得ます。

CTR 連鎖は、NIST *特別公報 800-38A『ブロック暗号の推奨動作モード』* の中に定義されています。CTR モードでの典型的なメッセージ構造を*図 261* に示します。

図 261. カウンタモードのメッセージ構造



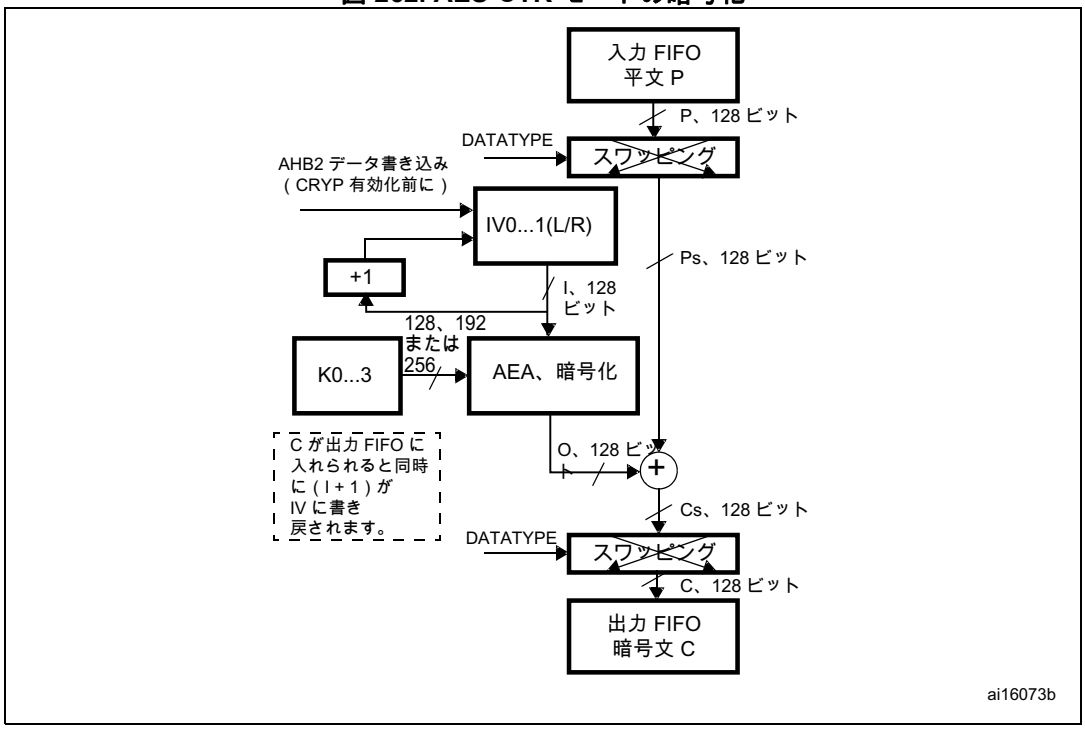
このメッセージの構造は次に示すとおりです。

- 16 バイトの初期カウンタブロック（ICB）は、次の 3 つの異なるフィールドで構成されています。
  - *ノンス*：32 ビットの使い捨て値（新しい通信のたびに新しいノンスを割り当てる必要があります）。
  - *初期化ベクタ*（IV）：ある特定のキーでのモードの実行それぞれに固有のものでなければならない 64 ビットの値。
  - *カウンタ*：ブロックが処理されるたびにインクリメントされるビッグエンディアンの 32 ビット整数。カウンタの初期値は1に設定する必要があります。
- 平文（P）には、既知の長さの暗号文 C として、認証と暗号化の両方が行われます。この長さは 16 バイトの倍数でなくとも構いませんが、その場合には平文のパディングが必要です。

AES CTR 処理

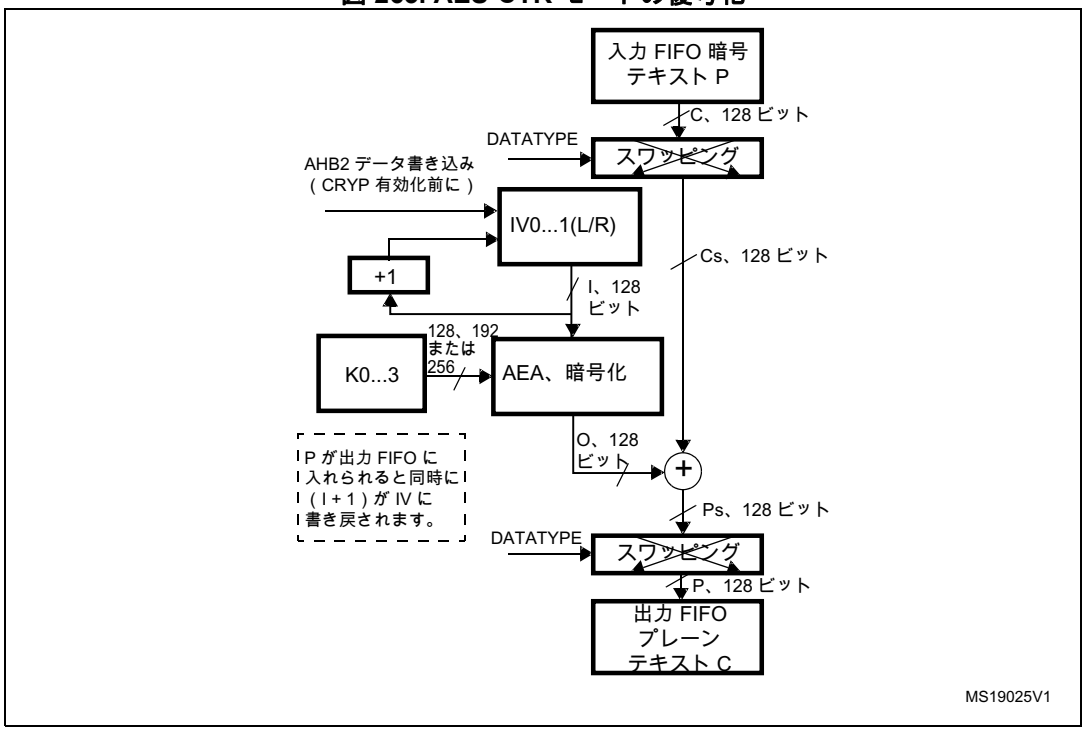
このペリフェラルの中で実行される AES-CTR の暗号化と復号化の処理を図 262と図 263にそれぞれ示します。このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

図 262. AES-CTR モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、o：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、Cs：スワッピング後（復号化時）またはスワッピング前（暗号化時）の暗号化テキスト、P：平文、IV：初期化ベクタ

図 263. AES-CTR モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、o：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、Cs：スワッピング後（復号化時）またはスワッピング前（暗号化時）の暗号化テキスト、P：平文、IV：初期化ベクタ

CTR モードでは、出力ブロックは、それに続く入力ブロックと排他的論理和がとられてから、アルゴリズムへ入力されます。ペリフェラルの初期化ベクタは、*表 263*のように初期化する必要があります。

表 263. カウンタモード初期化ベクタ

CRYP_IV1R[31:0]	CRYP_IV1L[31:0]	CRYP_IV0R[31:0]	CRYP_IV0L[31:0]
ノンス	IV[63:32]	IV[31:0]	32 ビットカウンタ = 0x1

最初のデータブロックを処理するときに一度だけ CRYP_IVx レジスタを使用する CBC モードとは違って、CTR モードでは、各データブロックを処理するたびに IV レジスタが使用され、ペリフェラルによって下位側 32 ビットがインクリメントされます（上位側 96 ビットは変更されません）。

コアが常に現在のカウンタブロックを暗号化して、入力として平文や暗号と排他的論理和がとられるキーストリームを生成するので、CTR 復号化は CTR 暗号化と変わりません。したがって、ALGOMODE が 0b110 にセットされると、ALGODIRは無視されます。

注： ***このモードでは、復号化のためにキーを準備する必要はありません。***

CTR 連鎖モードで暗号化または復号化を行うには、次のシーケンスを使用します。

1. CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。
2. 次のように CRYP_CR を設定します。
   a) ALGOMODE ビットを 0b110 にプログラムして CTR モードを選択します。
   b) DATATYPE ビットでデータ型（1、8、16、32 ビット）を設定します。
3. CRYP_KEYRx のキーレジスタ（128、192、256 ビット）を初期化し、*表 263*に示されているように初期化ベクタ（IV）を初期化します。
4. CRYP_CR レジスタで FFLUSH ビットに 1 を書き込み、IN および OUT FIFO を一掃します。
5. 最終ブロックの場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。
6. 暗号プロセッサにデータを追加して結果を読み出します。可能性のある 3 通りのケースは*セクション 35.3.5：暗号操作を行うための CRYP の手順*に記載されています。
7. 最後から 2 番目のブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。この最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、そのデータの一部ではないデータをドライバが破棄する必要があります。

CTR モードでのサスペンド／レジューム動作

CBC モードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。CBC シーケンスの詳細については、*セクション 35.3.11：CRYP AES基本連鎖モード（ECB、CBC）*を参照してください。

注： ***CBC モードと同様に、レジューム動作中に IV レジスタを再ロードする必要があります。***

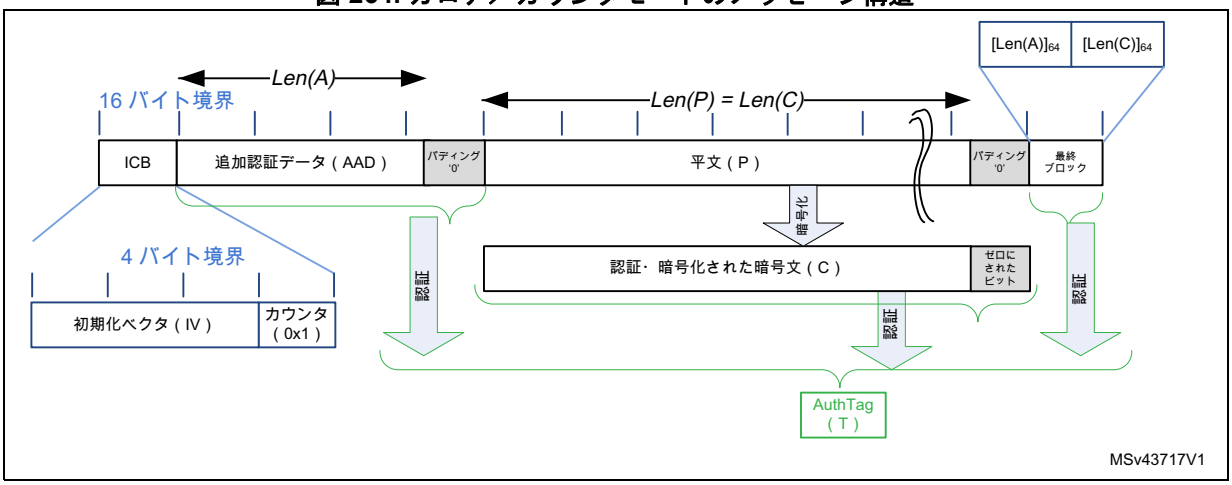
35.3.13 CRYP AES ガロア／カウンタモード（GCM）

概要

AES ガロア／カウンタモード（GCM）では、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、GCM アルゴリズムは AES カウンタモードに基づいています。固定の有限フィールドを超えて乗算器を使用しタグを生成します。

GCM 連鎖は、NIST *特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）およびGMAC』*の中に定義されています。GCM モードでの典型的なメッセージ構造を*図 264* に示します。

図 264. ガロア／カウンタモードのメッセージ構造



このメッセージの構造は次のとおりに定義されています。

- 16 バイトの初期カウンタブロック（ICB）は、次の 2 つの異なるフィールドで構成されています。
  - *初期化ベクタ*（IV）：ある特定のキーでのモードの実行それぞれに固有のものでなければならない 96 ビットの値。GCM 規格では 96 ビットよりも短い IV に対応していますが、この場合には規則が厳格に適用されることに注意してください。
  - *カウンタ*：ブロックが処理されるたびにインクリメントされるビッグエンディアンの 32 ビット整数。NIST 仕様によると、ペイロードの最初のブロックを処理するときのカウンタ値は 0x2 です。
- 認証済みヘッダ A（追加認証データとも言います）は、既知の長さである *Len(A)* を持ちますが、この長さは 16 バイトの倍数でなくても構いませんが、2^{64}-1 ビットを超えることはできません。メッセージのこの部分には認証だけ行われて、暗号化はされません。
- 平文メッセージ（P）には暗号文 C として認証と暗号化の両方が行われ、既知の長さである *Len(P)* を持ちますが、この長さは 16 バイトの倍数でなくても構いませんが、128 ビットの 2^{32}-2 ブロック分を超えることはできません。

注： ***GCM 規格では、暗号文 C は平文 P と同じビット長であることと規定されています。***

- メッセージ（AAD または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。
- *表 264*に示すように、最終ブロックは A の長さ（64 ビット）と暗号文 C の長さ（64 ビット）で構成されます。

表 264. GCM 最終ブロックの定義

エンディアン	ビット [0]	ビット [32]	ビット [64]	ビット [96]
入力データ	0x0	ヘッダ長 [31:0]	0x0	ペイロード長 [31:0]

AES GCM 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

GCM モードにおける平文の機密性のメカニズムはカウンタモードの変種であり、カウンタブロックの必要なシーケンスを生成する特別な 32 ビットインクリメント機能を備えています。

各データブロックの処理には CRYP_IV が使用されます。暗号プロセッサによって、カウンタブロックの下位側 32 ビットが自動的にインクリメントされます。アプリケーションによって書き込まれる最初のカウンタブロック（CB1）は、1 だけインクリメントされた初期カウンタブロックと等しくなります（*表 265*参照）。

表 265. GCM モード IV レジスタの初期化

レジスタ	CRYP_IV1R[31:0]	CRYP_IV1L[31:0]	CRYP_IV0R[31:0]	CRYP_IV0L[31:0]
入力データ	ICB[127:96]	ICB[95:64]	ICB[63:32]	ICB[31:0] 32 ビットカウンタ = 0x2

注： ***このモードでは、復号化のためにキーを準備する必要はありません。***

GCM モードにおける認証メカニズムは *GF2mul* と呼ばれるハッシュ機能に基づいており、このハッシュ機能では、バイナリガロア体の中で、ハッシュサブキー（H）と呼ばれる固定パラメータによる乗算が行われます。

GCM メッセージを処理するために、ドライバはこの先に記載されている 4 つのフェーズを行う必要があります。

- 初期フェーズ：ペリフェラルは GCM ハッシュキー（H）を準備し、IV 処理を行います。
- ヘッダフェーズ：ペリフェラルは追加認証データ（AAD）を処理して、ハッシュの計算のみを行います。
- ペイロードフェーズ：ペリフェラルは平文（P）を処理して、ハッシュの計算、キーストリームの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- 最終フェーズ：ペリフェラルは最終のデータブロックを使って認証タグ（T）を生成します。

1. **GCM 初期フェーズ**

この最初のステップでは、GCM ハッシュサブキー（H）が計算され、すべてのブロックの処理に使用されるように内部に保存されます。以下の手順に従うことを推奨します。

a) CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。

b) CRYP_CR で ALGOMODE ビットを 0b01000 にプログラムして GCM 連鎖モードを選択します。

c) CRYP_CR で GCM_CCMPH ビットを 0b00 に設定して初期フェーズが進行中であることを示します。

d) CRYP_KEYRx のキーレジスタ（128、192、256 ビット）を初期化し、*表 265*の定義のとおりに初期化ベクタ（IV）を初期化します。

e) CRYPEN ビットを 1 にセットして ハッシュキーの計算を開始します。

f) CRYPEN ビットが暗号プロセッサによって 0 にクリアされるのを待ってから、次のフェーズに移ります。

2. **GCM ヘッダフェーズ**

以下のシーケンスは、GCM 初期フェーズの後に実施する必要があります。ペイロードフェーズに進む前に完了している必要があります。シーケンスは、暗号化と復号化のシーケンスと同一です。

g) CRYP_CR で GCM_CCMPH ビットを 0b01 に設定してヘッダフェーズが進行中であることを示します。

h) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

i) 追加認証データの最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。

j) 追加認証データを暗号プロセッサに追加します。可能性のある 3 通りのケースは*セクション 35.3.5：暗号操作を行うための CRYP の手順*に記載されています。

k) 最後から 2 番目の追加認証データブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。すべての追加認証データが供給されたら、BUSY フラグがクリアされるまで待ってから、次のフェーズに移ります。

注： ***このフェーズは、追加認証データがなければ（Len(A)=0）スキップできます。***

ヘッダフェーズとペイロードフェーズでは、CRYPEN ビットは暗号プロセッサによって自動的にクリアされません。

3. **GCM ペイロードフェーズ（暗号化または復号化）**

ペイロードサイズが Null ではない場合には、このシーケンスは GCM ヘッダフェーズの後に実施する必要があります。このフェーズでは、暗号化／復号化したペイロードが CRYP_DOUT レジスタに保存されます。

l) CRYPEN ビットを 0 にセットします。

m) CRYP_CR レジスタで GCM_CCMPH ビットを 0b10 に設定してペイロードフェーズが進行中であることを示します。

n) CRYP_CR の ALGODIR ビットによってアルゴリズムの方向（暗号化は 0、復号化は 1）を選択します。

o) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

p) 平文または暗号文の最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。暗号化の詳細については、*セクション 35.3.8：CRYP スチーリングおよびデータパディング*を参照してください。

q) 暗号プロセッサにペイロードデータを追加して結果を読み出します。可能性のある 3 通りのケースは*セクション 35.3.5：暗号操作を行うための CRYP の手順*に記載されています。

r) 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を実行します。最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、平文または暗号文の一部ではないビットをドライバが破棄する必要があります。すべてのペイロードが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： ***このフェーズは、ペイロードデータがなければ（Len(C)=0）スキップできます（GMAC モード参照）。***

4. **GCM 最終フェーズ**

この最終フェーズで、暗号プロセッサによって GCM 認証タグが生成され、CRYP_DOUT レジスタに保存されます。

s) CRYP_CR で GCM_CCMPH[1:0] を 0b11 に設定して最終フェーズが進行中であることを示します。同じレジスタの ALGODIR ビットに 0 をセットします。

t) CRYP_DIN レジスタに入力を 4 回書き込みます。*表 264*に示すように、入力は、追加認証データのビット長（64 ビットで符号化）に続けて、ペイロードのビット長（64 ビットで符号化）で構成されている必要があります。

注： ***この最終フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。***

u) CRYP_SR レジスタで OFNE フラグ（FIFO 出力が空でない）が 1 にセットされるまで待ちます。

v) CRYP_DOUT レジスタを 4 回読み出します。この出力が認証タグに相当します。

w) 暗号プロセッサを無効にします（CRYP_CR の CRYPEN ビット =0）。

x) 認証された復号化が実行されている場合には、生成されたタグをメッセージとともに渡された予測タグと比較します。

GCM モードでのサスペンド／レジューム動作

ヘッダフェーズまたはペイロードフェーズで現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR レジスタで IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]、ビット [17:16]、ビット 19）して、暗号プロセッサを無効にします。さらに、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： ***アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。***

5. 次のコンテキストスワップレジスタを保存します：CRYP_CSGCMCCM0..7 および CRYP_CSGCM0..7
6. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. 次のコンテキストスワップレジスタを復元します：CRYP_CSGCMCCM0..7 および CRYP_CSGCM0..7
5. 保存されている設定を用いて CRYP_IVx レジスタを復元します。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサ DMA リクエストを再び有効とします。

注： ***ヘッダフェーズでは、DMA OUT FIFO 転送は使用されません。***

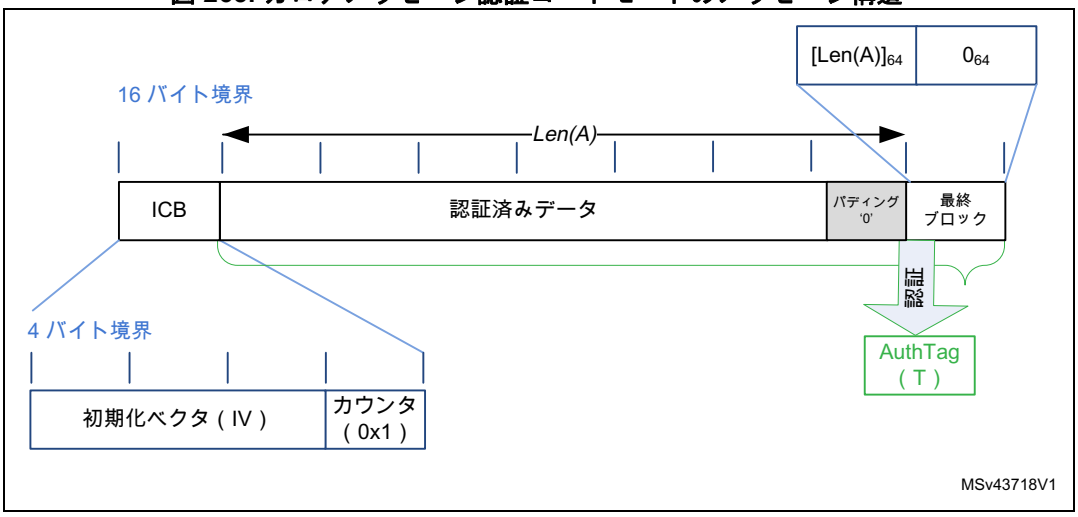
35.3.14 CRYP AES ガロアメッセージ認証コード（GMAC）

概要

ガロアメッセージ認証コード（GMAC）では、平文を認証して、対応するタグ情報（メッセージ認証コードとも言います）を生成することができます。NIST *特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』*に定義されている GCM アルゴリズムに基づいています。

GMAC モードでの典型的なメッセージ構造を*図 265* に示します。

図 265. ガロアメッセージ認証コードモードのメッセージ構造



AES GMAC 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

GMAC アルゴリズムはヘッダのみで構成されているメッセージに適用される GCM アルゴリズムに相当します。その結果、ペイロードフェーズ（3）が使用されないことを除くと、手順と設定は GCM モードとすべて同じです。

GMAC でのサスペンド／レジューム動作

GMAC は、ヘッダフェーズ（2）のみが割り込み可能であること以外 GCM アルゴリズムとまったく同じです。

35.3.15 CRYP AES CBC-MAC 付き カウンタ（CCM）

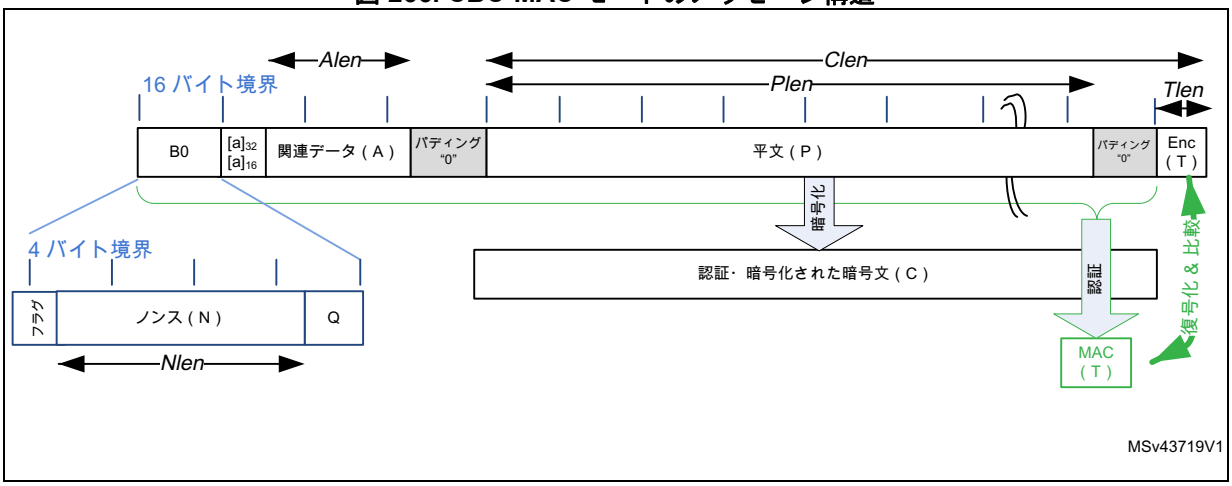
概要

AES 暗号ブロック連鎖-メッセージ認証コード（CCM）付きカウンタ（CCM）アルゴリズムでは、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、CCM アルゴリズムは AES カウンタモードに基づいています。暗号ブロック連鎖技術を使ってメッセージ認証コードを生成します。これは、一般的に CBC-MAC と呼ばれます。

注： ***NIST は、CCM 仕様の関係以外では CBC-MAC を認証モードとして承認していません。***

CCM 連鎖は、NIST *特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のための CCM モード』*の中に規定されています。CCM モードでの典型的なメッセージ構造を*図 266* に示します。

図 266. CBC-MAC モードのメッセージ構造



このメッセージの構造は次に示すとおりです。

- 次の 3 つの異なるフィールドで構成されている 16 バイトの初回認証ブロック（規格では B0 と呼ばれています）
  - Q : P のバイト長（Plen）のビットストリング表現
  - *ノンス*（N）*:* 使い捨て値（新しい通信のたびに新しいノンスを割り当てる必要があります）。ノンスのサイズである *Nlen* と *Plen* のサイズの和は 15 バイトに等しい必要があります。
  - *フラグ*: 規格に規定されている、制御情報のための 4 つのフラグを格納する最上位バイト。値 t（バイト単位での MAC の長さ）と *q*（Plen<2^{8q} などの平文の長さ）をエンコードするための 3 ビットのストリングが 2 つ格納されています。q に対応付けられるカウンタブロックの範囲は 2^{8q-4} に等しいことに注意してください。q の最大値が 8 であるとすると、暗号で使用されるカウンタブロックは 60 ビットとなります。

注： ***暗号ペリフェラルは、長さが Plen < 2^{36} +1 バイトのパディングされた平文と暗号文のみを管理可能です。***

- 関連データ（A）に関連付けられた 16 バイトブロック（B）

  メッセージのこの部分には認証だけ行われて、暗号化はされません。この部分は既知の長さである *ALen*, を持っており、この長さは 16 バイトの倍数でなくとも構いません（*図 266*参照）。規格には、最初のメッセージブロック（B1）の 上位側ビットにおいて、バイト数で表現された関連データ長（*a*）は次の定義のようにエンコードされなければならないとも示されています。

  – $0 < a < 2^{16}-2^{8}$ である場合には、$[a]_{16}$ のようにエンコードされますので 2 バイトとなります。

  – $2^{16}-2^{8} < a < 2^{32}$ である場合には、0xff || 0xfe || $[a]_{32}$ のようにエンコードされますので 6 バイトとなります。

  – $2^{32} < a < 2^{64}$ である場合には、0xff || 0xff || $[a]_{64}$ のようにエンコードされますので 10 バイトとなります。

- 既知の長さ *Plen* を持ち、暗号文 C として認証と暗号化の両方が行われている平文メッセージ（P）に関連付けられた 16 バイトブロック（B）。この長さは 16 バイトの倍数でなくとも構いません（*図 266*参照）が、128 ビットの 2^{32} ブロック分を超えることはできません。
- 全長が *Clen* である暗号文 C に追加された長さ *Tlen* の暗号化された MAC (T)。
- メッセージ（A または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。

注： ***CCM 連鎖モードは、関連データのみ（ペイロードなし）にも使用できます。***

例として、NIST 特別公報 800-38C のセクション C.1 には次のように示されています。

```
N?10111213 141516  （Nlen= 56 ビットまたは 0x7 バイト）
A?00010203 04050607（Alen= 64 ビットまたは 0x8 バイト）
P?20212223（Plen= 32 ビット、すなわち Q= 0x4 バイト）
T?6084341b（Tlen= 32 ビットまたは t= 4）
B0?4f101112 13141516 00000000 00000004
B1?00080001 02030405 06070000 00000000
B2?20212223 00000000 00000000 00000000
CTR0?0710111213 141516 00000000 00000000
CTR1?0710111213 141516 00000000 00000001
```

コントロールブロック CTRx の使用方法は、以下のセクションに説明されています。最初のブロック（B0）からの CTR0 の生成は、ソフトウェアで管理してください。

AES CCM 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b1001 を書き込むことによって選択されます。

生成-暗号化処理に入力するデータは、有効なナンス、有効なペイロードストリング、有効な関連データストリングであり、すべて適切にフォーマットされているものです。CBC 連鎖メカニズムはフォーマットされたデータに適用されて、既知の長さを持つ MAC を生成します。入力として十分な長さのカウンタブロックのシーケンスが必要であるカウンタモードの暗号化は、ペイロードストリングと MAC に個別に適用されます。暗号文 C と呼ばれる結果データは、平文 P に対する生成-暗号化処理の出力です。

各データブロックの処理には CRYP_IV が使用されます。暗号プロセッサによって、最初のブロック（B0）により定義されるビット長を持つ CTR カウンタが自動的にインクリメントされます。アプリケーションによって書き込まれる最初のカウンタである CTR1 は、最初の 5 ビットをゼロとし、P のバイト長が格納されている上位側ビットもゼロとした後に 1 だけインクリメントした B0 に等しくなります（*表 266*参照）。

表 266. CCM モード IV レジスタの初期化

レジスタ	CRYP_IV0L[31:0]	CRYP_IV0R[31:0]	CRYP_IV1L[31:0]	CRYP_IV1R[31:0]
エンディアン	IV[0:31]	IV[32:63]	IV[64:95]	IV[96:127]
入力データ	B0[31:0]、ただし上位側 5 ビットはゼロにクリア（フラグビット）	B0[63:32]	B0[95:64]	B0[127:96]、ただし 1 にセットされるビット 1 を除く長さビット Q はゼロにセット

注： ***このモードでは、復号化のためにキーを準備する必要はありません。***

CCM メッセージを処理するために、ドライバは以下に記載されている 4 つのフェーズを行う必要があります。

- 初期フェーズ：ペリフェラルは最初のブロックを処理してカウンタブロックを準備します。
- ヘッダフェーズ：ペリフェラルは関連データ（A）を処理して、ハッシュの計算のみを行います。
- ペイロードフェーズ：ペリフェラルは平文（P）を処理して、ハッシュの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- 最終フェーズ：ペリフェラルはメッセージ認証コード（MAC）を生成します。

1. **CCM 初期フェーズ**

   この最初のステップでは、CCM メッセージの最初のブロック（B0）が CRYP_DIN レジスタにプログラムされます。このフェーズでは、CRYP_DOUT レジスタには出力データは一切格納されていません。以下の手順に従うことを推奨します。

   a) CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。

   b) CRYP_CR レジスタで ALGOMODE ビットを 0b01001 にプログラムして、CCM 連鎖モードを選択します。

   c) CRYP_CR で GCM_CCMPH ビットを 0b00 に設定して初期フェーズにあることを示します。

   d) CRYP_KEYRx のキーレジスタ（128、192、256 ビット）を初期化し、*表 266*の定義のとおりに初期化ベクタ（IV）を CTR1 情報で初期化します。

   e) CRYP_CR で CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

   f) B0 パケットを CRYP_DIN レジスタに書き込み、CRYPEN ビットが暗号プロセッサによって 0 にクリアされるのを待ってから、次のフェーズに移ります。

注： ***この初期フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。***

2. CCM ヘッダフェーズ

以下のシーケンスは、CCM 初期フェーズの後に実施する必要があります。ペイロードフェーズに進む前に完了している必要があります。シーケンスは、暗号化と復号化のシーケンスと同一です。このフェーズでは、CRYP_DOUT レジスタには出力データは一切格納されていません。

g) CRYP_CR で GCM_CCMPH ビットを 0b01 に設定してヘッダフェーズが進行中であることを示します。

h) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

i) 関連データの最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。

j) 関連データを暗号プロセッサに追加します。可能性のある 3 通りのケースは*セクション 35.3.5：暗号操作を行うための CRYP の手順*に記載されています。

k) 最後から 2 番目の関連データブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。すべての追加認証データが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： ***このフェーズは、関連データがなければ（Alen=0）スキップできます。***

注： ***関連データ B1 の最初のブロックは、関連データ長でフォーマットする必要があります。このタスクはソフトウェアで管理します。***

3. CCM ペイロードフェーズ（暗号化または復号化）

ペイロードサイズが Null ではない場合には、このシーケンスは CCM ヘッダフェーズの後に実施する必要があります。このフェーズでは、暗号化／復号化したペイロードが CRYP_DOUT レジスタに保存されます。

l) CRYPEN ビットを 0 にセットします。

m) CRYP_CR で GCM_CCMPH ビットを 0b10 に設定してペイロードフェーズが進行中であることを示します。

n) CRYP_CR の ALGODIR ビットによってアルゴリズムの方向（暗号化は 0、復号化は 1）を選択します。

o) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

p) 平文の最終ブロックの場合には、必要に応じてデータをゼロで埋めて完全なブロックとします（暗号化のみ）。詳細については、*セクション 35.3.8：CRYP スチーリングおよびデータパディング*を参照してください。

q) 暗号プロセッサにペイロードデータを追加して結果を読み出します。可能性のある 3 通りのケースは*セクション 35.3.5：暗号操作を行うための CRYP の手順*に記載されています。

r) 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を実行します。暗号文の最終ブロックでは（復号化のみ）、最終ブロックのサイズが 16 バイト未満である場合に、その平文の一部ではないデータをドライバが破棄する必要があります。すべてのペイロードが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： ***このフェーズは、ペイロードデータがなければ（Plen=0 または Clen=Tlen）スキップできます。***

注： ***暗号文 C を復号化するときには、暗号化されたタグ情報である LSB_{Tlen}(C) の削除を忘れないでください。***

4. **CCM 最終フェーズ**

   この最終フェーズで、暗号プロセッサによって CCM 認証タグが生成され、CRYP_DOUT レジスタに保存されます。

   s) CRYP_CR で GCM_CCMPH[1:0] ビットを 0b11 に設定して最終フェーズが進行中であることを示し、同じレジスタの ALGODIR ビットを 0 にセットします。

   t) *表 266*に示した CTR0 情報のビット [0] を 0 にセットして、CRYP_DIN にロードします。

注： ***この最終フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。***

   u) CRYP_SR レジスタで OFNE フラグ（FIFO 出力が空でない）が 1 にセットされるまで待ちます。

   v) CRYP_DOUT レジスタを 4 回読み出します。出力は暗号化されたCCMタグに相当します。

   w) 暗号プロセッサを無効にします（CRYP_CR の CRYPEN ビットを 0 にセット）。

   x) 認証された復号化が実行されている場合には、生成された暗号タグを暗号文でパディングされた暗号タグと比較します（LSB_{Tlen}(C)= MSB_{Tlen}（CRYP_DOUT データ））。

CCM モードでのサスペンド／レジューム動作

ペイロードフェーズで現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR レジスタで IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]、ビット [17:16]、ビット 19）して、暗号プロセッサを無効にします。さらに、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： ***アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。***

5. 次のコンテキストスワップレジスタを保存します：CRYP_CSGCMCCM0..7
6. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. 次のコンテキストスワップレジスタを復元します：CRYP_CSGCMCCM0..7
5. 保存されている設定を用いて CRYP_IVx レジスタを復元します。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサ DMA リクエストを再び有効とします。

注： ***ヘッダフェーズでは、DMA OUT FIFO 転送は使用されません。***

35.3.16 CRYP データレジスタおよびデータスワッピング

概要

CRYP_DIN レジスタは、ペリフェラルの 32 ビット幅データ入力レジスタです。このレジスタを使用して、最大 4 つの 64 ビットブロック（トリプルDES）または 2 つの 128 ビットブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 ビットワードずつ入力 FIFO に入力します。

最初に FIFO に書き込まれるワードは、入力ブロックの LSB です。入力ブロックの MSB は、最後に書き込まれます。CRYP_DIN データのエンディアンは、DATATYPE=“00”（データスワッピングなし）のときに次のように示されます。

- DES/トリプル DES モード

  データブロックのビット 1（左端のビット）は FIFO に入力される最初のワードの MSB（ビット 31）に対応し、ビット 64（右端のビット）は FIFO に入力される 2 番目のワードの LSB（ビット 0）に対応します。

- AES モード

  データブロックのビット 0（左端のビット）は FIFO に書き込まれる最初のワードの MSB（ビット 31）に対応し、ビット 127（右端のビット）は FIFO に書き込まれる 4 番目のワードの LSB（ビット 0）に対応します。

同様に、CRYP_DOUT レジスタは、ペリフェラルの 32 ビット幅データ出力レジスタです。このレジスタは読み出し専用レジスタであり、最大 4 つの 64 ビットブロック（トリプルDES）または 2 つの 128 ビットブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 ビットワードずつ出力 FIFO から読み出すために使用されます。

入力データの場合と同様、出力 FIFO から最初に読み出されるワードが出力ブロックの LSB です。出力ブロックの MSB は、最後に読み出されます。CRYP_DOUT データのエンディアンは、DATATYPE="00"（データスワッピングなし）のときに次のように示されます。

- DES/トリプル DES モード

  データブロックのビット 1（左端のビット）は FIFO から読み出される最初のワードの MSB（ビット 31）に対応し、ビット 64（右端のビット）は FIFO から読み出される 2 番目のワードの LSB（ビット 0）に対応します。

- AES モード

  データブロックのビット 0（左端のビット）は FIFO から読み出される最初のワードの MSB（ビット 31）に対応し、ビット 127（右端のビット）は FIFO から読み出される 4 番目のワードの LSB（ビット 0）に対応します。

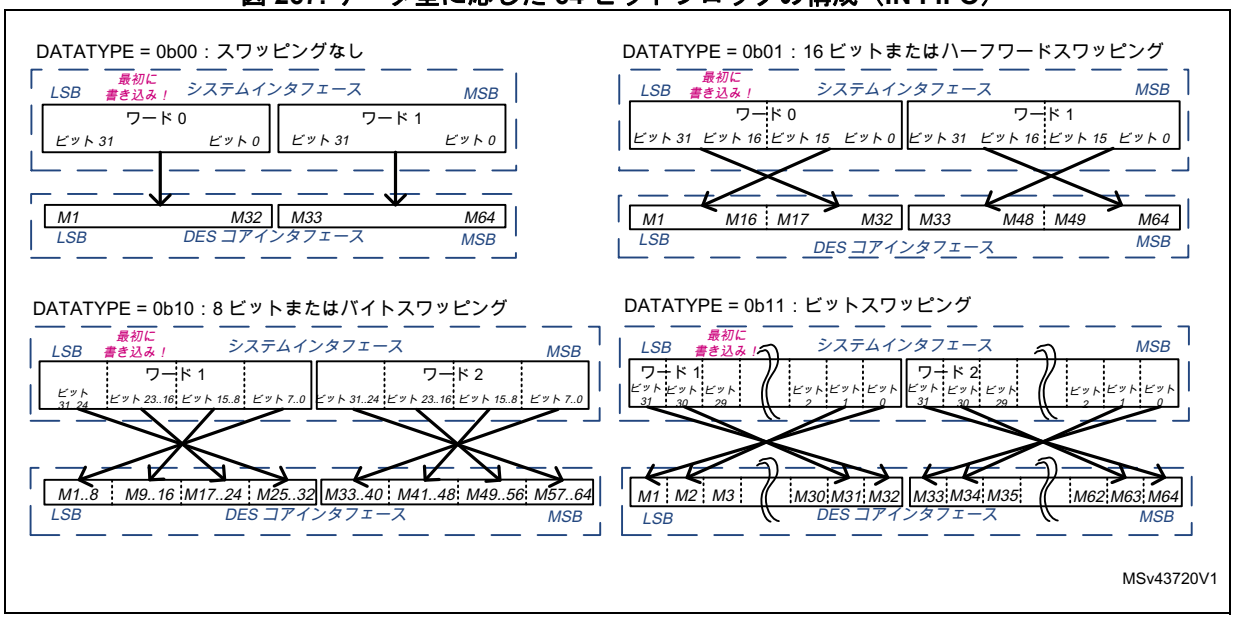
DES／トリプル DES データスワッピング機能

処理されるデータの種類（データが ASCII テキストストリームである場合のバイトスワッピングなど）次第では、入力 FIFO から読み込まれたデータをリトルエンディアン DES 処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの操作を実行する必要があります。同じスワッピング処理を、リトルエンディアン DES 処理コアによって生成されたデータに対して、それが出力 FIFO に書き込まれる前に行う必要があります。

図 267 に、DES 処理コアの 64 ビットデータブロック M1...64 が、ドライバによって IN FIFO にポップされる 2 つの連続した 32 ビットワードからどのように構成されるかを示します。この処理は、CRYP_CR レジスタの DATATYPE ビットフィールドに従って行われます。

注： ***IN FIFO と CRYP データブロック間および CRYP データブロックと OUT FIFO 間でも同じスワッピングが行われます。***

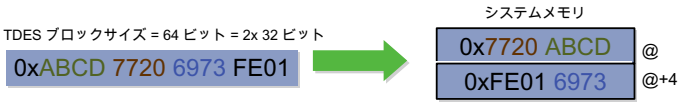
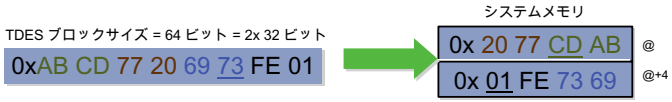
図 267. データ型に応じた 64 ビットブロックの構成（IN FIFO）



注： ***CRYP キーレジスタ（CRYP_Kx（L/R））と初期化レジスタ（CRYP_IVx（L/R））は、選択されたスワップモードの影響を受けません。これらはリトルエンディアン設定に固定です（それぞれセクション 35.3.17およびセクション 35.3.18を参照）。***

データスワッピングの典型例を*表 267* に示します。

表 267. DES／トリプル DES データスワッピング機能

CRYP_CR 内の DATATYPE	実施される スワッピング処理	データブロック表現（64 ビット） 0xABCD7720 6973FE01 システムメモリデータ （平文または暗号）
0b00	スワッピングなし	アドレス @：0xABCD7720（LSB、最初に書き込み） アドレス @+4：0x6973FE01 
0b01	ハーフワード（16 ビット）スワッピング	アドレス @：0x7720ABCD（スワップされたLSB、最初に書き込み） アドレス @+4：0xFE016973 
0b10	バイト（8 ビット）スワッピング	アドレス @：0x2077CDAB（スワップされたLSB、最初に書き込み） アドレス @+4：0x01FE7369 
0b11	ビットスワッピング	LSB データワード：0xABCD7720 0b1010 1011 1100 1101 0111 0111 0010 0000 MSB データワード：0x6973FE01 0b0110 1001 0111 0011 1111 1110 0000 0001 アドレス @：0x04EEB3D5（スワップされたLSB、最初に書き込み） アドレス @+4：0x807FCE96

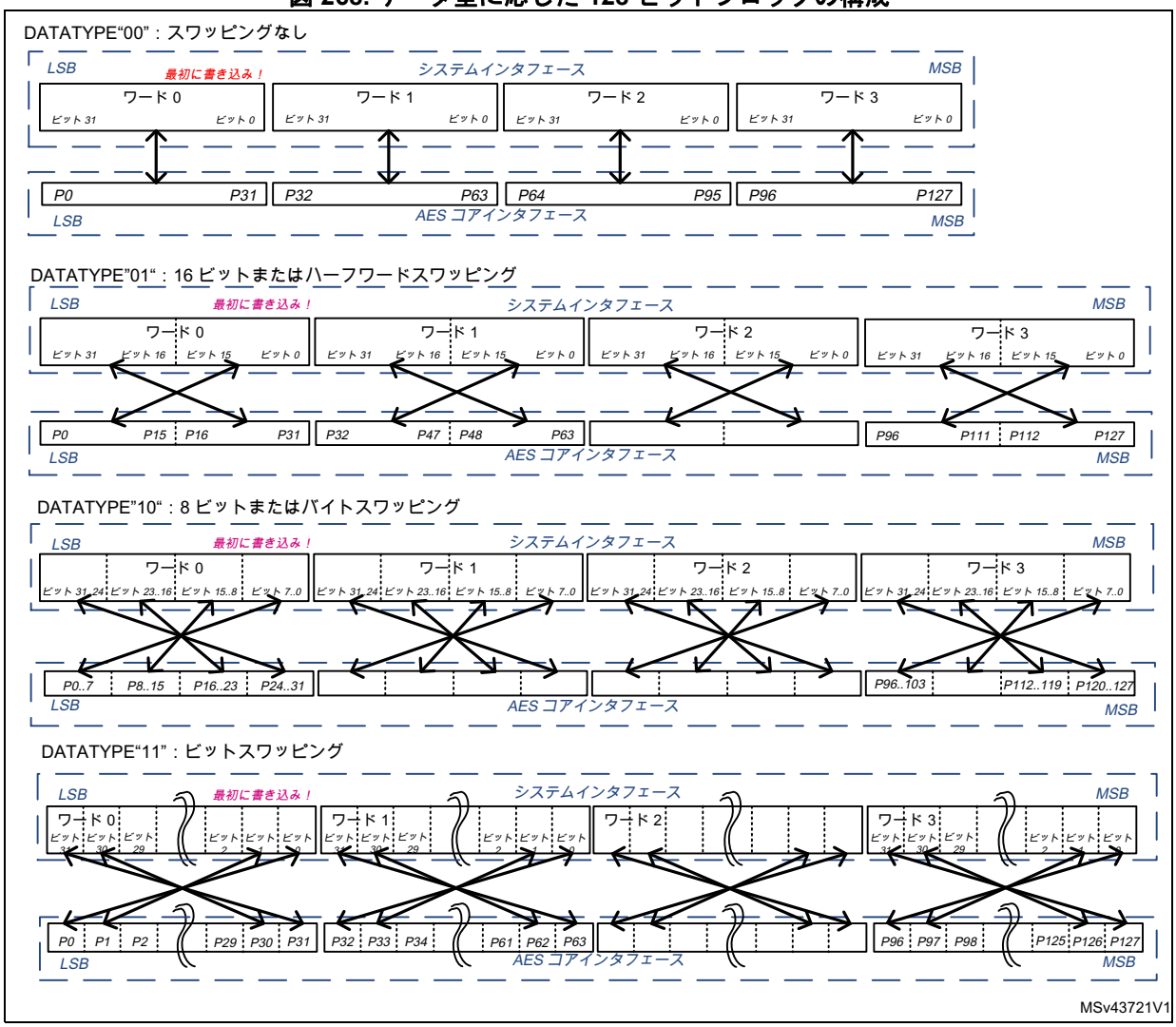
AES データスワッピング機能

処理されるデータの種類（データが ASCII テキストストリームである場合のバイトスワッピングなど）次第では、入力 FIFO から読み込まれたデータをリトルエンディアン AES 処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの操作を実行する必要があります。同じスワッピング処理を、リトルエンディアン AES 処理コアによって生成されたデータに対して、それが出力 FIFO に書き込まれる前に行う必要があります。

図 268 に、AES 処理コアの 128 ビットデータブロック P0..127 が、ドライバによって CRYP_DIN レジスタに書き込まれる 4 つの連続した 32 ビットワードからどのように構成されるかを示します。この処理は、CRYP コントロールレジスタ（CRYP_CR）の DATATYPE ビットフィールドに従って行われます。

注： ***CRYP_DIN と CRYP データブロック間および CRYP データブロックと CRYP_DOUT 間でも同じスワッピングが行われます。***

図 268. データ型に応じた 128 ビットブロックの構成



注： *スワッピング操作は、CRYP_DOUT および CRYP_DIN レジスタのみを対象とします。CRYP_KxL/KxR レジスタと CRYP_IVxL/IVxR レジスタは、選択されたスワップモードの影響を受けません。これらはリトルエンディアン設定に固定です（セクション 35.3.17 および セクション 35.3.18 を参照）。*

データスワッピングの典型例を*表 268* に示します。

表 268. AES データスワッピング機能

CRYP_CR 内の DATATYPE	実施されるスワッピング処理	データブロック表現（64 ビット） 0x4E6F7720 69732074
		システムメモリデータ（リトルエンディアン）
0b00	スワッピングなし	アドレス @：0x4E6F7720（LSB、最初に書き込み） アドレス @+4：0x69732074
0b01	ハーフワード（16 ビット）スワッピング	アドレス @：0x77204E6F（スワップされたLSB、最初に書き込み） アドレス @+4：0x20746973
0b10	バイト（8 ビット）スワッピング	アドレス @：0x20776F4E（スワップされたLSB、最初に書き込み） アドレス @+4：0x74207369
0b11	ビットスワッピング	LSB データワード：0x4E6F7720 0b0100 1110 0110 1111 0111 0111 0010 0000 MSB データワード：0x69732074 0b0110 1001 0111 0011 0010 0000 0111 0100 アドレス @：0x4EEF672（スワップされたLSB、最初に書き込み） アドレス @+4：0x2E04CE96

35.3.17 CRYP キーレジスタ

CRYP_Kx レジスタは暗号化または復号化キーを格納するために使用されます。

*表 269*に示すように、リトルエンディアン構造の8つのレジスタ として構成されています。

表 269. キーレジスタ CRYP_KxR/LR エンディアン（トリプル DES K1/2/3 および AES 128/192/256 ビットキー）

K0LR[31:0]	K0RR[31:0]	K1LR[31:0]	K1RR[31:0]	K2LR[31:0]	K2RR[31:0]	K3LR[31:0]	K3RR[31:0]
-	-	K1[1:32]	K1[33:64]	K2[1:32]	K2[33:64]	K2[1:32]	K2[33:64]
K0LR[31:0]	**K0RR[31:0]**	**K1LR[31:0]**	**K1RR[31:0]**	**K2LR[31:0]**	**K2RR[31:0]**	**K3LR[31:0]**	**K3RR[31:0]**
-	-	-	-	k[0:31]	k[32:63]	k[64:95]	k[96:127]
K0LR[31:0]	**K0RR[31:0]**	**K1LR[31:0]**	**K1RR[31:0]**	**K2LR[31:0]**	**K2RR[31:0]**	**K3LR[31:0]**	**K3RR[31:0]**
-	-	k[0:31]	k[32:63]	k[64:95]	k[96:127]	k[128:159]	k[160:191]
K0LR[31:0]	**K0RR[31:0]**	**K1LR[31:0]**	**K1RR[31:0]**	**K2LR[31:0]**	**K2RR[31:0]**	**K3LR[31:0]**	**K3RR[31:0]**
k[0:31]	k[32:63]	k[64:95]	k[96:127]	k[128:159]	k[160:191]	k[192:223]	k[224:255]

注： *DES／トリプル DES キーには、暗号プロセッサでは使用しない 8 ビットパリティ情報が含まれています。つまり、各 64 ビットのキー値 Kx[1:64] のビット 8、16、24、32、40、48、56、64 は使用されません。*

キーは、4 つの 64 ビットデータ項目とみなされます。したがって、キーはシステムメモリ内で平文や暗号データと同じデータ形式や表現とはなりません。

CRYP_SR レジスタの BUSY ビット = 1 にセットされているときには、CRYP_Kx（L/R）レジスタに対するあらゆる書き込み操作が無視されます（レジスタの内容は変更されません）。そのため、ソフトウェアはキーレジスタを変更する前に、BUSY が 0 であることを確認する必要があります。

キーレジスタは、CRYP_CR レジスタの DATATYPE の値によって制御されるデータスワッピング機能による影響を受けません。

CRYP_Kx（L/R）レジスタの詳細については、*セクション 35.6 : CRYP レジスタ*を参照してください。

35.3.18 CRYP 初期化ベクタレジスタ

CRYP_IVxL/IVxR レジスタは、選択された連鎖モードに応じて、初期化ベクタまたはノンスの格納に用いられます。これらのレジスタが使用されると、トリプル DES コアまたは AES コアの計算ラウンドの後にコアによって毎回更新されます。

*表 270*に示すように、リトルエンディアン構造のレジスタ 4 個として構成されています。

表 270. 初期化ベクトルレジスタ CRYP_IVxR エンディアン

CRYP_IV1R[31:0]	CRYP_IV1L[31:0]	CRYP_IV0R[31:0]	CRYP_IV0L[31:0]
IV[96:127]	IV[64:95]	IV[32:63]	IV[0:31]

初期化ベクタレジスタは、2 つの 64 ビットデータ項目とみなされます。したがって、キーはシステムメモリ内で平文や暗号データと同じデータ形式や表現とはなりません。

CRYP_SR レジスタの BUSY ビット = 1 にセットされているときには、CRYP_IV0...1（L/R）レジスタに対するあらゆる書き込み操作が無視されます（レジスタの内容は変更されません）。そのため、ソフトウェアは初期化ベクタを変更する前に、CRYP_SR レジスタで BUSY が 0 であることを確認する必要があります。

CRYP_IV0...1（L/R）レジスタを読み出すと、CCM/GCM を除いて、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

注 : ***DES／トリプルDES モードでは、CRYP_IV0x のみを使用します。***

初期化ベクタレジスタは、CRYP_CR レジスタの DATATYPE の値によって制御されるデータスワッピング機能による影響を受けません。

CRYP_IVxL/IVxR レジスタの詳細については、*セクション 35.6 : CRYP レジスタ*を参照してください。

35.3.19 CRYP DMA インタフェース

暗号プロセッサには、DMA（ダイレクトメモリアクセス）コントローラに接続するインタフェースが搭載されています。DMA 操作は CRYP DMA 制御レジスタ（CRYP_DMACR）を通じて制御します。

DMA を用いたデータ入力

DMA は、CRYP_DMACR レジスタの DIEN ビットをセットすることによって、暗号ペリフェラルへのデータ書き込みが可能となります。このビットがセットされると、暗号プロセッサは INPUT フェーズ中、CRYP_DIN レジスタにワードを書き込む必要があるたびに、DMA リクエストを開始します。

*表 271*に、DMA コントローラを通じてメモリから暗号プロセッサへデータを転送するための推奨設定を示します。

表 271. メモリからペリフェラルへの
DMA 転送のための暗号プロセッサの設定

DMA チャネルコントロールレジスタフィールド	プログラミング上の推奨事項
転送サイズ	メッセージ長は 128 ビットの倍数。この 128 ビット粒度は、DES の 2 ブロック、AES の 1 ブロックに対応しています。 選択されたアルゴリズムとモードに応じて、特別なパディングや暗号文借用が必要となることがあります。*詳細については、セクション 35.3.8：CRYP スチーリングおよびデータパディング*を参照してください。
転送元バーストサイズ（メモリ）	CRYP FIFO_size /2 /transfer_width = **4**
転送先バーストサイズ（ペリフェラル）	CRYP FIFO_size /2 /transfer_width = **4** （FIFO_size= 8x32 ビット、transfer_width= 32 ビット）
DMA FIFO サイズ	CRYP FIFO_size /2 = 16 バイト
転送元転送幅（メモリ）	32 ビットワード
転送先転送幅（ペリフェラル）	32 ビットワード
転送元アドレスインクリメント（メモリ）	あり。32 ビット転送後に毎回
転送先アドレスインクリメント（ペリフェラル）	固定アドレスの CRYP_DIN を使用します（インクリメントなし）。

DMA を用いたデータ出力

AES ペリフェラルからデータを読み込む DMA を有効とするには、CRYP_DMACR レジスタの DOEN ビットをセットします。このビットがセットされると、暗号プロセッサは OUTPUT フェーズ中、CRYP_DOUT レジスタからワードを読み込む必要があるたびに、DMA リクエストを開始します。

*表 272*に、DMA コントローラを通じて暗号プロセッサからメモリへデータを転送するための推奨設定を示します。

表 272. ペリフェラルからメモリへの
DMA 転送のための暗号プロセッサの設定

DMA チャネルコントロールレジスタフィールド	プログラミング上の推奨事項
転送サイズ	メッセージ長は 128 ビットの倍数。この 128 ビット粒度は、DES の 2 ブロック、AES の 1 ブロックに対応しています。 アルゴリズムによっては、余分なビットを破棄する必要があります。
転送元バーストサイズ（ペリフェラル）	*DES* を使用する場合： シングル転送（バーストサイズ =**1**） *AES* を使用する場合： CRYP FIFO_size /2 /transfer_width = **4** （FIFO_size= 8x32 ビット、transfer_width= 32 ビット）
転送先バーストサイズ（メモリ）	CRYP FIFO_size /2 /transfer_width = **4**
DMA FIFO サイズ	CRYP FIFO_size /2 = 16 バイト
転送元転送幅（ペリフェラル）	32 ビットワード

表 272. ペリフェラルからメモリへの
DMA 転送のための暗号プロセッサの設定

DMA チャネルコントロールレジスタフィールド	プログラミング上の推奨事項
メモリ転送幅（メモリ）	32 ビットワード
転送元アドレスインクリメント（ペリフェラル）	固定アドレスの CRYP_DOUT を使用します（インクリメントなし）
転送先アドレスインクリメント（メモリ）	あり。32 ビット転送後に毎回

DMA モード

AES 使用時には、暗号プロセッサは cryp_in_dma と cryp_out_dma の内部入出力信号のアサートされたものを通じて、2 本の DMA 転送リクエストを管理します。

- IN FIFO に対して：CRYP によって FIFO から 1 ブロック読み込まれるたび
- OUT FIFO に対して：暗号プロセッサによって FIFO に 1 ブロック書き込まれるたび

DES 使用時には、暗号プロセッサは cryp_in_dma と cryp_out_dma の内部入出力信号のアサートされたものを通じて、2 本の DMA 転送リクエストを管理します。

- IN FIFO に対して：暗号プロセッサによって FIFO から 2 ブロック読み込まれるたび
- OUT FIFO に対して：暗号プロセッサによって FIFO に 1 ワード書き込まれるたび（シングル転送）FIFO に 2 ブロック書き込まれるとバースト転送もトリガされることに注意してください。

暗号ペリフェラルが無効になるか、DMA イネーブルビットがクリアされると（CRYP_DMACR レジスタにおいて IN FIFO に対しては DIEN ビット、OUT FIFO に対しては DOEN ビット）、すべてのリクエスト信号がネゲートされます。

注意： ***CRYP 入力 FIFO が一杯になる前に、DMA コントローラが暗号ペリフェラル出力 FIFO を空にすることが重要です。このため、メモリからペリフェラルへの転送の優先度よりも、ペリフェラルからメモリへの転送の優先度の方が高くなるように、DMA コントローラを設定してください。***

DMA 動作の詳細については、*セクション 35.3.5：暗号操作を行うための CRYP の手順*を参照してください。

35.3.20 CRYP エラー管理

暗号プロセッサによって生成されるエラーフラグはありません。

35.4 CRYP 割り込み

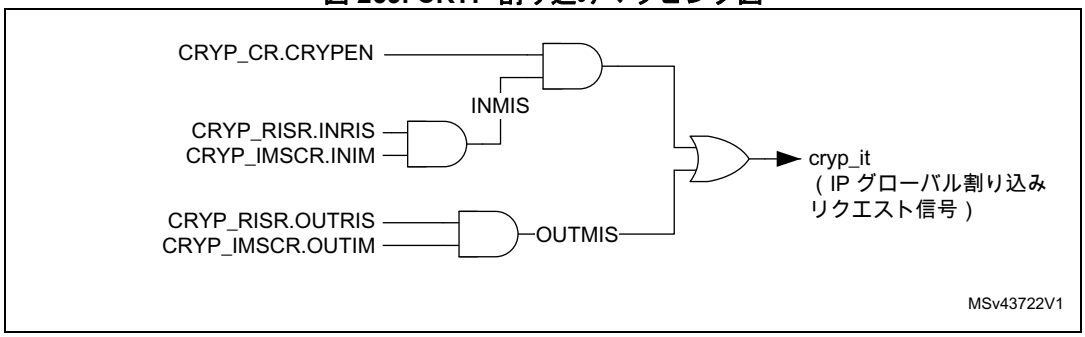
概要

暗号プロセッサによって生成され個別にマスク可能な割り込みソースは 2 つあり、次のイベントが通知されます。

- 入力 FIFO エンプティまたはノットフル
- 出力 FIFO フルまたはノットエンプティ

この 2 つのソースを組み合わせて 1 つの割り込み信号にします。これが NVIC（ネスト化されたベクタ割り込みコントローラ）を駆動する CRYP ペリフェラルからの唯一の割り込み信号です。*図 269*に*割り込みロジックの概要を示します。*

図 269. CRYP 割り込みマッピング図



CRYP_IMSCR レジスタのマスクビットを変更することにより、CRYP 割り込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割り込みが有効になります。

個別にマスク可能な割り込みソースのステータスは、マスク前の割り込みステータスに対しては CRYP_RISR レジスタから、マスクされた割り込みステータスに対しては CRYP_MISR レジスタから読み出すことができます。イベントフラグの個別のソースのステータスは CRYP_SR レジスタから読み出すことができます。

*表 273*に、使用可能な機能の概要を示します。

表 273. CRYP 割り込みリクエスト

割り込みイベント	イベントフラグ（割り込みステータス）	イネーブル制御ビット	イベントフラグ（ソース）
出力 FIFO フル	OUTRIS、OUTMIS	OUTIM と CRYPEN	OFFU
出力 FIFO ノットエンプティ			OFNE
入力 FIFO ノットフル	OUTRIS、OUTMIS	NIM と CRYPEN	IFNF
入力 FIFO エンプティ			IFEM

出力 FIFO サービス割り込み - OUTMIS

出力 FIFO に 1 つ以上のデータ項目（32 ビットワード）がある場合、出力 FIFO サービス割り込みがアサートされます。この割り込みは、有効な（32 ビット）ワードがなくなるまで出力 FIFO からデータを読み出すことによってクリアされます（割り込みが出力 FIFO ノットエンプティフラグ OFNE の状態に従うとこうなります）。

出力 FIFO サービス割り込み OUTMIS は、CRYP イネーブルビットでは有効になりません。結果として、出力 FIFO が空でない場合、CRYP を無効にしても OUTMIS 信号を強制的にローにすることはできません。

入力 FIFO サービス割り込み - INMIS

入力 FIFO が 4 ワード未満となった場合、入力 FIFO サービス割り込みがアサートされます。この割り込みは、入力 FIFO に 4 ワード以上を保持するようになるまで書き込み操作を行うことによってクリアされます。

入力 FIFO サービス割り込み INMIS は、CRYP イネーブルビットで有効になります。結果として、CRYP を無効にすると、入力 FIFO が空であっても INMIS 信号がローになります。

35.5 処理時間

各動作モードで 128 ビットブロックの処理に必要な時間の概要を次に示します。

表 274. 128 ビットブロックあたりの ECB、CBC および CTR の処理時間（クロックサイクル数）

アルゴリズム／キー長	ECB	CBC	CTR
128b	14	14	14
192b	16	16	16
256b	18	18	18

表 275. 128 ビットブロックあたりの GCM および CCM の処理時間（クロックサイクル数）

アルゴリズム／キー長	GCM					CCM				
	初期化	ヘッダ	ペイロード	タグ	合計	初期化	ヘッダ	ペイロード	タグ	合計
128b	24	10	14	14	**62**	12	14	25	14	**65**
192b	28	10	16	16	**70**	14	16	29	16	**75**
256b	32	10	18	18	**78**	16	18	33	18	**85**

35.6 CRYP レジスタ

暗号コアは、複数の制御およびステータスレジスタ、8 個のキーレジスタ、4 個の初期化ベクタレジスタと連動しています。

35.6.1 CRYP 制御レジスタ（CRYP_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				ALGOMODE[3]	Res.	GCM_CCMPH	
												rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRYPEN	FFLUSH	Res.	Res.	Res.	Res.	KEYSIZE		DATATYPE		ALGOMODE[2:0]			ALGODIR	Res.	Res.
rw	w					rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **ALGOMODE[3]**：ビット [5:3] の説明を参照してください。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **GCM_CCMPH**：GCM または CCM フェーズの選択

ALGOMODE フィールドで GCM、GMAC、または CCM アルゴリズムが選択されていない場合、このビットフィールドの影響はありません。

00：初期フェーズ
01：ヘッダフェーズ
10：ペイロードフェーズ
11：最終フェーズ

ビット 15 **CRYPEN**：CRYP プロセッサイネーブル

0：暗号プロセッサペリフェラルは無効です。
1：暗号プロセッサペリフェラルは有効です。
このビットは、キーの準備プロセスが終了するか（ALGOMODE=0b111)、GCM/GMAC または CCM 初期化フェーズの後は、自動的にハードウェアによってクリアされます。

ビット 14 **FFLUSH**：CRYP FIFO 一掃

0：FIFO を一掃しません。
1：FIFO 一掃は有効です。
CRYPEN = 0 の場合、このビットに 1 を書き込むと IN および OUT FIFO が一掃されます（つまり、FIFO の読み出しおよび書き込みポインタがリセットされます）。このビットに 0 を書き込んでも影響はありません。CRYPEN = 1 の場合、このビットに 0 や 1 を書き込んでも影響はありません。
このビットを読み出すと常に 0 が返されます。
FFLUSH ビットは、BUSY=0 のときにのみセットする必要があります。さもないと、FIFO は一掃されますが処理中のブロックは一掃操作の直後に出力 FIFO にプッシュされ、FIFO が空でなくなることがあります。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 KEYSIZE：キーサイズ選択（AES モードのみ）

このビットフィールドにより、AES 暗号コアに使用するキーのビット長を規定します。このビットフィールドは、DES またはトリプルDES モードでは意味を持ちません。

00：128 ビットのキー長

01：192 ビットのキー長

10：256 ビットのキー長

11：予約済み。この値は使用しないでください。

BUSY=1 の間に KEYSIZE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 7:6 DATATYPE：データ型選択

このビットフィールドによって、CRYP_DIN レジスタに書き込まれるデータまたは CRYP_DOUT レジスタから読み出されるデータのフォーマットを規定します。詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング*を参照してください。

00：32 ビットデータ。各ワードのスワッピング無し。IN FIFO にプッシュされる（または OUT FIFO からポップされる）最初のワードがデータブロックのビット 1 から 32 を形成し、2 番目のワードがビット 33 から 64 を形成します。

01：16 ビットデータ、またはハーフワード。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 2 つのハーフワードとみなされます。

10：8 ビットデータ、またはバイト。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 4 バイトとみなされます。

11：ビットデータ、またはビット列。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 32 ビット（位置 0 にあるビット列の最初のビット）とみなされます。

BUSY=1 の間に DATATYPE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 5:3 **ALGOMODE[2:0]**：アルゴリズムモード

次の定義にはビット 19 が含まれます。

0000：トリプル DES-ECB（トリプル DES 電子コードブック）

0001：トリプル DES-CBC（トリプル DES 暗号ブロック連鎖）

0010：DES-ECB（シンプル DES 電子コードブック）

0011：DES-CBC（シンプル DES 暗号ブロック連鎖）

0100：AES-ECB（AES 電子コードブック）

0101：AES-CBC（AES 暗号ブロック連鎖）

0110：AES-CTR（AES カウンタモード）：

0111：ECB 復号化または CBC 復号化用の AES キーの準備

1000：AES-GCM（ガロアカウンタモード）および AES-GMAC（ガロアメッセージ認証コードモード）

1001：AES-CCM（CBC-MAC 付き カウンタ）

BUSY=1 の間に ALGOMODE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 2 **ALGODIR**：アルゴリズムの方向

0：暗号化

1：復号化

BUSY=1 の間に ALGODIR ビットに書き込んでも影響はありません。このビットは、BUSY=0 のときにのみ設定できます。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

35.6.2 CRYP ステータスレジスタ（CRYP_SR）

アドレスオフセット：0x04

リセット値：0x0000 0003

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **BUSY**：ビジービット

0：CRYP コアは、いかなるデータも処理していません。その理由は、次のとおりです。

– CRYP コアが無効になっており（CRYP_CR レジスタで CRYPEN=0）、最後の処理が完了している、または

– CRYP コアが入力 FIFO に十分なデータが入るまで待っている、または出力 FIFO に十分な空きができるまで待っている（つまり、それぞれ DES の場合で 2 ワード以上、AES の場合で 4 ワード以上）。

1：CRYP コアが現在データブロックを処理中であるか、キー準備が進行中です（AES、ECBまたはCBC復号化のみ）。

ビット 3 **OFFU**：出力 FIFO フルフラグ

0：出力 FIFO はフルではありません。

1：出力 FIFO はフルです。

ビット 2 **OFNE**：出力 FIFO ノットエンプティフラグ

0：出力 FIFO は空です。

1：出力 FIFO は空ではありません。

ビット 1 **IFNF**：入力 FIFO ノットフルフラグ

0：入力 FIFO はフルです。

1：入力 FIFO はフルではありません。

ビット 0 **IFEM**：入力 FIFO エンプティフラグ

0：入力 FIFO は空ではありません。

1：入力 FIFO は空です。

35.6.3 CRYP データ入力レジスタ（CRYP_DIN）

アドレスオフセット：0x08

リセット値：0x0000 0000

CRYP_DIN レジスタは、データ入力レジスタです。このレジスタは 32 ビット幅です。このレジスタを使用して、最大 4 つの 64 ビットブロック（トリプルDES）または 2 つの 128 ビットブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 ビットワードずつ入力 FIFO に入力します。

さまざまなデータサイズに対応するため、CRYP_CR レジスタの DATATYPE ビットを設定することにより、データは、処理後にスワッピングすることができます。詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング* を参照してください。

CRYP_DIN レジスタに書き込まれると、データは入力 FIFO にプッシュされます。

- CRYPEN = 1 である場合、DES／トリプル DES モードにおいて、32 ビットワードが 2 ワード（AES モードでは 4 ワード）以上入力 FIFO にプッシュされており、かつ出力 FIFO に 2 ワード（AES モードでは 4 ワード）以上の空きがある場合、CRYP エンジンは暗号化または復号化プロセスを開始します。

CRYP_DIN レジスタが読み出されると、

- CRYPEN = 0 の場合、FIFO が作動し、最も古いもの（最初の読み出し）から最新のもの（最後の読み出し）まで、入力 FIFO にあるデータが返されます。各読み出し操作の前に、IFEM フラグをチェックして FIFO が空でないことを確認する必要があります。
- CRYPEN = 1 の場合、未定義の値が返されます。

注： ***CRYP_DIN レジスタが1回以上読み出された後は、新しいデータを処理する前に FFLUSH ビットをセットして FIFO を一掃する必要があります。***

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

ビット 31:0 **DATAIN**：データ入力

読み込み時には FIFO がポップされて（最後に書き込まれたデータが返されて）、CRYPEN=0 であればその値が返されます。CRYPEN=1 の時には、DATAIN レジスタは予期しない値を返します。

書き込み時には現在のレジスタの内容が FIFO にプッシュされます。

35.6.4 CRYP データ出力レジスタ（CRYP_DOUT）

アドレスオフセット：0x0C

リセット値：0x0000 0000

CRYP_DOUT レジスタは、データ出力レジスタです。読み出し専用で 32 ビット幅です。このレジスタを使用して、最大 4 つの 64 ビットブロック（トリプルDES）または 2 つの 128 ビットブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 ビットワードずつ出力 FIFO から取り出します。

さまざまなデータサイズに対応するため、CRYP_CR レジスタの DATATYPE ビットを設定することにより、データは、処理後にスワッピングすることができます。詳細については、*セクション 35.3.16：CRYP データレジスタおよびデータスワッピング* を参照してください。

CRYP_DOUT レジスタを読み出すと、出力 FIFO に最後に入ったデータ（読み出しポインタが指すデータ）が返されます。

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

ビット 31:0 **DATAOUT**：データ出力

読み出し時には出力 FIFO の内容（読み込みポインタが指している）を返し、それ以外の場合、未定義の値を返します。
書き込みには影響ありません。

35.6.5 CRYP DMA 制御レジスタ（CRYP_DMACR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOEN	DIEN
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DOEN**：DMA 出力イネーブル

このビットがセットされると、出力データフェーズ中にペリフェラルによって DMA リクエストが自動生成されます。
0：送信データ転送用 DMA は無効です。
1：送信データ転送用 DMA は有効です。

ビット 0 **DIEN**：DMA 入力イネーブル

このビットがセットされると、入力データフェーズ中にペリフェラルによって DMA リクエストが自動生成されます。
0：受信データ転送用 DMA は無効です。
1：受信データ転送用 DMA は有効です。

35.6.6 CRYP 割り込みマスクセット／クリアレジスタ（CRYP_IMSCR）

アドレスオフセット：0x14

リセット値：0x0000 0000

CRYP_IMSCR レジスタは、割り込みマスクセットまたはクリアのためのレジスタです。読み出し／書き込みレジスタです。読み出し操作時、このレジスタは、関連する割り込みに適用される現在のマスク値を示します。特定のビットに 1 を書き込むとマスクがセットされ、割り込みが読み出せるようになります。このビットに 0 を書き込むと、該当するマスクがクリアされます。ペリフェラルがリセットされると、すべてのビットが 0 にクリアされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTIM	INIM
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTIM**：出力 FIFO サービス割り込みマスク

0：出力 FIFO サービス割り込みはマスクされています。

1：出力 FIFO サービス割り込みはマスクされていません。

ビット 0 **INIM**：入力 FIFO サービス割り込みマスク

0：入力 FIFO サービス割り込みはマスクされています。

1：入力 FIFO サービス割り込みはマスクされていません。

35.6.7 CRYPマスク前割り込みステータスレジスタ（CRYP_RISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

CRYP_RISR レジスタは、マスク前割り込みステータスレジスタです。読み出し専用レジスタです。読み出し時、このレジスタは該当する割り込みの現在のマスク前のステータス（CRYP_IMSCR マスクが考慮されない割り込み情報）を示します。書き込み操作には影響がありません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTRIS	INRIS
														r	r

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTRIS**：出力 FIFO サービスマスク前割り込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない出力 FIFO 割り込み情報を示します。

0：マスク前割り込みは保留中ではありません。

1：マスク前割り込みは保留中です。

ビット 0 **INRIS**：入力 FIFO サービスマスク前割り込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない入力 FIFO 割り込み情報を示します。

0：マスク前割り込みは保留中ではありません。

1：マスク前割り込みは保留中です。

35.6.8 CRYP マスク済み割り込みステータスレジスタ（CRYP_MISR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

CRYP_MISR レジスタは、マスク済み割り込みステータスレジスタです。読み出し専用レジスタです。読み出し時、このレジスタは該当する割り込みの現在のマスク後のステータス（CRYP_IMSCR マスクが考慮された割り込み情報）を示します。書き込み操作には影響がありません。

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTMIS**：出力 FIFO サービスマスク済み割り込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない出力 FIFO 割り込み情報を示します。

0：割り込みは保留中ではありません。

1：割り込みが保留中です。

ビット 0 **INMIS**：入力 FIFO サービスマスク済み割り込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない入力 FIFO 割り込み情報を示します。

0：割り込みは保留中ではありません。

1：CRYPEN = 1 の場合、割り込みは保留中です。

35.6.9 CRYP キーレジスタ 0L（CRYP_K0LR）

アドレスオフセット：0x20

リセット値：0x0000 0000

CRYP キーレジスタには、暗号キーが入っています。

- DES／トリプルDES モードでは、キーは 64 ビットのバイナリ値（左から右に番号が振られています。つまり左端のビットがビット 1 です。）であり、K1、K2、K3（K0 は使用しません）と名づけられています。各キーは 56 の情報ビットと 8 つのパリティビットで構成されています。
- AES モードでは、キーは 1 つの 128、192、256 ビット長のシーケンス $K_0K_1K_2...K_{127/191/255}$ とみなされます。AES キーは次のようにレジスタに入力されます。
  - AES-128 の場合：$K_0..K_{127}$ は $b_{127}..b_0$ に対応（$b_{255}..b_{128}$ は使用されません）
  - AES-192 の場合：$K_0..K_{191}$ は $b_{191}..b_0$ に対応（$b_{255}..b_{192}$ は使用されません）
  - AES-256 の場合：$K_0..K_{255}$ は $b_{255}..b_0$ に対応

どの場合にも、キービット K_0 がCRYP 内部メモリの中で左端のビットであり、レジスタビット b_0 が対応する CRYP_KxLR キーレジスタの中で右端のビットとなります。

詳細については、*セクション 35.3.17：CRYP キーレジスタ*を参照してください。

注： ***暗号プロセッサがビジー（CRYP_SR レジスタで BUSY ビット = 1）のとき、これらのレジスタへの書き込みアクセスは無視されます。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K255	K254	K253	K252	K251	K250	K249	K248	K247	K246	K245	K244	K243	K242	K241	K240
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K239	K238	K237	K236	K235	K234	K233	K232	K231	K230	K229	K228	K227	K226	K225	K224
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 224 **Kx**：AES キービット x（x = 224～255）

注： このビットは DES モードでは使用しません。

35.6.10 CRYP キーレジスタ 0R（CRYP_K0RR）

アドレスオフセット：0x24

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K223	K222	K221	K220	K219	K218	K217	K216	K215	K214	K213	K212	K211	K210	K209	K208
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K207	K206	K205	K204	K203	K202	K201	K200	K199	K198	K197	K196	K195	K194	K193	K192
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 192 **Kx**：AES キービット x（x = 192～223）

注： このビットは DES モードでは使用しません。

35.6.11 CRYP キーレジスタ 1L（CRYP_K1LR）

アドレスオフセット：0x28

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K191	K190	K189	K188	K187	K186	K185	K184	K183	K182	K181	K180	K179	K178	K177	K176
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K175	K174	K173	K172	K171	K170	K169	K168	K167	K166	K165	K164	K163	K162	K161	K160
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 160 **Kx**：AES キービット x（x = 160～191）

DES モードでは、K192 は キー K1 ビット 1 に、K160 はキー K1 ビット 32 に対応します。

参考資料

35.6.12 CRYP キーレジスタ 1R（CRYP_K1RR）

アドレスオフセット：0x2C

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K159	K158	K157	K156	K155	K154	K153	K152	K151	K150	K149	K148	K147	K146	K145	K144
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K143	K142	K141	K140	K139	K138	K137	K136	K135	K134	K133	K132	K131	K130	K129	K128
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 128 **Kx**：AES キービット x（x = 128～159）

DES モードでは、K159 は キー K1 ビット 33 に、K128 はキー K1 ビット 64 に対応します。

35.6.13 CRYP キーレジスタ 2L（CRYP_K2LR）

アドレスオフセット：0x30

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K127	K126	K125	K124	K123	K122	K121	K120	K119	K118	K117	K116	K115	K114	K113	K112
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K111	K110	K109	K108	K107	K106	K105	K104	K103	K102	K101	K100	K99	K98	K97	K96
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 96 **Kx**：AES キービット x（x = 96～127）

DES モードでは、K127 は キー K2 ビット 1 に、K96 はキー K2 ビット 32 に対応します。

35.6.14 CRYP キーレジスタ 2R（CRYP_K2RR）

アドレスオフセット：0x34

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K95	K95	K93	K92	K91	K90	K89	K88	K87	K86	K85	K84	K83	K82	K81	K80
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K79	K78	K77	K76	K75	K74	K73	K72	K71	K70	K69	K68	K67	K66	K65	K64
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 64 **Kx**：AES キービット x（x = 64～95）

DES モードでは、K95 は キー K2 ビット 33 に、K64 はキー K2 ビット 64 に対応します。

35.6.15 CRYP キーレジスタ 3L（CRYP_K3LR）

アドレスオフセット：0x38

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K63	K62	K61	K60	K59	K58	K57	K56	K55	K54	K53	K52	K51	K50	K49	K48
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K47	K46	K45	K44	K43	K42	K41	K40	K39	K38	K37	K36	K35	K34	K33	K32
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x - 32 **Kx**：AES キービット x（x = 32～63）

DES モードでは、K63 は キー K3 ビット 1 に、K32 はキー K3 ビット 32 に対応します。

35.6.16 CRYP キーレジスタ 3R（CRYP_K3RR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

詳細については、*セクション 35.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K31	K30	K29	K28	K27	K26	K25	K24	K23	K22	K21	K20	K19	K18	K17	K16
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット x **Kx**：AES キービット x（x = 0～31）

DES モードでは、K31 は キー K3 ビット 33 に、K0 はキー K3 ビット 64 に対応します。

35.6.17 CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）

アドレスオフセット：0x40

リセット値：0x0000 0000

CRYP_IV0...1（L/R）R は、初期化ベクタの左ワードレジスタおよび右ワードレジスタ（DES／トリプル DES の場合は 64 ビット、AES の場合は 128 ビット）です。詳細については、*セクション 35.3.18：CRYP 初期化ベクタレジスタ*を参照してください。

IV0 は、初期化ベクタの左端のビットであり、IV63（DES、トリプル DES）または IV127（AES）は右端のビットです。IV1(L/R)R は、AES でのみ使用します。CRYP_IV0（L/R）は、DES／トリプル DES でのみ使用します。

注： ***暗号プロセッサがビジー（CRYP_SR レジスタで BUSY ビット = 1）のとき、これらのレジスタへの書き込みアクセスは無視されます。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV0	IV1	IV2	IV3	IV4	IV5	IV6	IV7	IV8	IV9	IV10	IV11	IV12	IV13	IV14	IV15
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV16	IV17	IV18	IV19	IV20	IV21	IV22	IV23	IV24	IV25	IV26	IV27	IV28	IV29	IV30	IV31
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 - x **IVx**：初期化ベクタビット x（x = 0～31）

35.6.18 CRYP 初期化ベクタレジスタ 0R（CRYP_IV0RR）

アドレスオフセット：0x44

リセット値：0x0000 0000

詳細については、*セクション 35.6.17：CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV32	IV33	IV34	IV35	IV36	IV37	IV38	IV39	IV40	IV41	IV42	IV43	IV44	IV45	IV46	IV47
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV48	IV49	IV50	IV51	IV52	IV53	IV54	IV55	IV56	IV57	IV58	IV59	IV60	IV61	IV62	IV63
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 63 - x **IVx**：初期化ベクタビット x（x = 32～63）

35.6.19 CRYP 初期化ベクタレジスタ 1L（CRYP_IV1LR）

アドレスオフセット：0x48

リセット値：0x0000 0000

詳細については、*セクション 35.6.17：CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV64	IV65	IV66	IV67	IV68	IV69	IV70	IV71	IV72	IV73	IV74	IV75	IV76	IV77	IV78	IV79
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV80	IV81	IV82	IV83	IV84	IV85	IV86	IV87	IV88	IV89	IV90	IV91	IV92	IV93	IV94	IV95
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 95 - x **IVx**：初期化ベクタビット x（x = 64～95）

注：　このビットは DES モードでは使用しません。

35.6.20 CRYP 初期化ベクタレジスタ 1R（CRYP_IV1RR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

詳細については、*セクション 35.6.17：CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV96	IV97	IV98	IV99	IV100	IV101	IV102	IV103	IV104	IV105	IV106	IV107	IV108	IV109	IV110	IV111
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV112	IV113	IV114	IV115	IV116	IV117	IV118	IV119	IV120	IV121	IV122	IV123	IV124	IV125	IV126	IV127
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 127 - x **IVx**：初期化ベクタビット x（x = 96～127）

注：　このビットは DES モードでは使用しません。

35.6.21 CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）

アドレスオフセット：0x050 + x* 0x4（x=0～7）

リセット値：0x0000 0000

GCM/GMAC または CCM アルゴリズムが選択された場合、CRYP プロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。暗号プロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクが暗号プロセッサを必要とするためにコンテキストスワップを実行する必要がある場合に役立ちます。

そのような場合、CRYP_CSGCMCCM0..7R および CRYP_CSGCM0..7R（GCM/GMAC モード時）または CRYP_CSGCMCCM0..7R（CCM モード時）レジスタを読み出し、取得した値をシステムメモリ空間に保存する必要があります。そうすると、暗号プロセッサは優先権のあるタスクによって使用できるようになります。暗号計算が完了すると保存されたコンテキストをメモリから読み出し、該当するコンテキストスワップレジスタに書き戻すことができます。

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

ビット 31:0 **CRYP_CSGCMCCMxR**：GCM、GMAC、CCMの各モード用の CRYP 内部ステートレジスタ

注：　このビットは、示されたもの以外の DES／トリプル DES または AES モードでは使用しません。

35.6.22 CRYP コンテキストスワップ GCM レジスタ（CRYP_CSGCMxR）

アドレスオフセット：0x070 + x* 0x4（x=0～7）

リセット値：0x0000 0000

詳細については、*セクション 35.6.21：CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）*を参照してください。

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

ビット 31:0 **CRYP_CSGCMxR**：GCM モードと GMAC モード用の CRYP 内部ステートレジスタ

注： ***このビットは、示されたもの以外の DES／トリプル DES または AES モードでは使用しません。***

参考資料

35.6.23 CRYP レジスタマップ

表 276. CRYP レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00 0x00	CRYP_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				ALGOMODE[3]	Res.	GCM_CCMPH		CRYPEN	FFLUSH	Res.	Res.	Res.	Res.	KEYSIZE		DATATYPE		ALGOMODE[2:0]			ALGODIR	Res.	Res.
	リセット値									0	0	0	0	0		0	0	0	0					0	0	0	0	0	0	0	0		
0x04	CRYP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	OFFU	OFNE	IFNF	IFEM
	リセット値																												0	0	0	1	1
0x08	CRYP_DIN	DATAIN																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	CRYP_DOUT	DATAOUT																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	CRYP_DMACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOEN	DIEN
	リセット値																															0	0
0x14	CRYP_IMSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTIM	INIM
	リセット値																															0	0
0x18	CRYP_RISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTRIS	INRIS
	リセット値																															0	1
0x1C	CRYP_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTMIS	INMIS
	リセット値																															0	0
0x20	CRYP_K0LR	CRYP_K0LR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	CRYP_K0RR	CRYP_K0RR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...																																	
0x38	CRYP_K3LR	CRYP_K3LR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	CRYP_K3RR	CRYP_K3RR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	CRYP_IV0LR	CRYP_IV0LR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 276. CRYP レジスタマップとリセット値（続き）

オフセット	レジスタ名 / リセット値	レジスタサイズ																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	CRYP_IV0RR	CRYP_IV0RR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	CRYP_IV1LR	CRYP_IV1LR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	CRYP_IV1RR	CRYP_IV1RR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	CRYP_CSGCMCCM0R	CRYP_CSGCMCCM0R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	CRYP_CSGCMCCM1R	CRYP_CSGCMCCM1R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x58	CRYP_CSGCMCCM2R	CRYP_CSGCMCCM2R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	CRYP_CSGCMCCM3R	CRYP_CSGCMCCM3R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	CRYP_CSGCMCCM4R	CRYP_CSGCMCCM4R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	CRYP_CSGCMCCM5R	CRYP_CSGCMCCM5R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	CRYP_CSGCMCCM6R	CRYP_CSGCMCCM6R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	CRYP_CSGCMCCM7R	CRYP_CSGCMCCM7R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	CRYP_CSGCM0R	CRYP_CSGCM0R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	CRYP_CSGCM1R	CRYP_CSGCM1R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	CRYP_CSGCM2R	CRYP_CSGCM2R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	CRYP_CSGCM3R	CRYP_CSGCM3R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x80	CRYP_CSGCM4R	CRYP_CSGCM4R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	CRYP_CSGCM5R	CRYP_CSGCM5R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 276. CRYP レジスタマップとリセット値（続き）

オフセット	レジスタ名 リセット値	レジスタサイズ																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	CRYP_CSGCM6R	CRYP_CSGCM6R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	CRYP_CSGCM7R	CRYP_CSGCM7R																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

36 ハッシュプロセッサ（HASH）

36.1 概要

ハッシュプロセッサは、各種のアプリケーションに適したセキュアハッシュアルゴリズム（SHA-1、SHA-224、SHA-256）、MD5（メッセージダイジェストアルゴリズム 5）ハッシュアルゴリズムおよび HMAC（鍵付きハッシュメッセージ認証コード）アルゴリズムに完全に準拠した処理系となっています。HMAC アルゴリズムによって、ハッシュ関数を用いたメッセージの認証を可能としています。これは、SHA-1、SHA-224、SHA-256 または MD5 のいずれかのハッシュ関数を 2 回呼び出すことを基本としています。

ハッシュプロセッサは、最長（$2^{64} – 1$）ビットのメッセージに対して、メッセージダイジェスト（SHA-1 アルゴリズムは 160 ビット、SHA-256 アルゴリズムは 256 ビット、SHA-224 アルゴリズムは 224 ビット、MD5 アルゴリズムは 128 ビット）を生成します。

36.2 HASH の主な機能

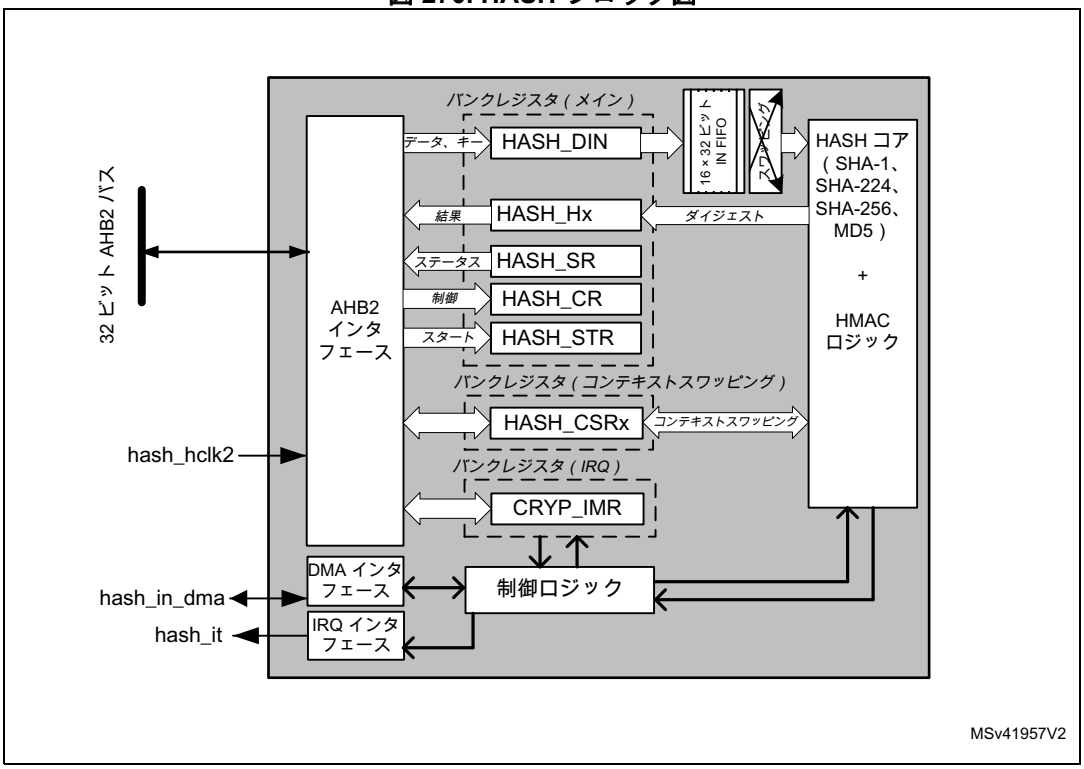
- 以下に準拠したデータ認証アプリケーションに好適
  - FIPS PUB 180-1（連邦情報処理規格公報 180-1）*セキュアハッシュ標準規格*（SHA-1）
  - FIPS PUB 180-2（連邦情報処理規格公報 180-2）*セキュアハッシュ標準規格*（SHA-224 および SHA-256）
  - インターネット技術タスクフォース（IETF）RFC 1321 *MD5 メッセージダイジェストアルゴリズム*
  - インターネット技術タスクフォース（IETF）RFC 2104 *HMAC：メッセージ認証のための鍵付きハッシング*
- 連続するメッセージブロックから、対応する 32 ビットワードのダイジェストが相互に追加され、メッセージ全体のダイジェストを形成
  - 入力ビット列の内部リトルエンディアン表現に準拠する自動 32 ビットワードスワッピング
  - 対応ワードスワッピング：ビット、バイト、ハーフワードおよび 32 ビットワード
- 512 ビット（16 × 32 ビット）のダイジェスト最低ブロックサイズに適合するよう入力ビット列を補完する自動パディング
- 1 個のブロックサイズに対応した 16 個の 32 ビットワードの内部入力 FIFO に関連するシングル 32 ビット入力レジスタ
- SHA-1、SHA-224、SHA-256 および MD5 の高速計算
  - SHA-1（それぞれ SHA-256）アルゴリズムを使用して 1 個の 512 ビットブロックのデータを処理する場合は 82（それぞれ 66）クロックサイクル
  - MD5 アルゴリズムを使用して 1 個の 512 ビットブロックのデータを処理する場合は 66 クロックサイクル
- 32 ビットワードのアクセスでのみ（それ以外は AHB エラーが発生）アクセス可能な AHB スレーブペリフェラル。
- 出力メッセージダイジェストの場合 8 × 32 ビットワード（H0 から H7）
- 1 つのチャネルを使用したダイレクトメモリアクセス（DMA）をサポートする自動データフロー制御。4 の固定バーストをサポート
- 32 ビットワード単位での割り込み可能なメッセージダイジェスト計算
  - 再ロード可能なダイジェストレジスタ
  - DMA の使用を含むハッシュ計算サスペンド／レジュームメカニズム

36.3 HASH の機能詳細

36.3.1 HASH ブロック図

図 270 に、ハッシュプロセッサのブロック図を示します。

図 270. HASH ブロック図



36.3.2 HASH 内部信号

表 277 では、製品レベル（パッド上）ではなく、HASH レベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表 277. HASH 内部入力／出力信号

信号名	信号タイプ	説明
hash_hclk2	デジタル入力	AHB2 バスクロック
hash_it	デジタル出力	ハッシュプロセッサグローバル割り込みリクエスト
hash_in_dma	デジタル入出力	DMA バーストリクエスト／確認

36.3.3 セキュアハッシュアルゴリズムについて

ハッシュプロセッサは、FIPS PUB 180-1 標準（SHA1）、FIPS PUB 180-2 標準（SHA-224、SHA-256）および IETF RFC1321 公報（MD5）によって定義されたセキュアハッシュアルゴリズムに完全に準拠した処理系となっています。

各アルゴリズムで、HASH はメッセージやデータファイルの圧縮表記を生成します。より具体的には、入力メッセージの長さが 2^{64} ビット未満の場合、SHA-1、SHA-224、SHA-256 および MD5 処理コアは、メッセージダイジェストと呼ばれる 160 ビット、224 ビット、256 ビット および 128 ビットの出力文字列をそれぞれ生成します。その後、メッセージの署名を作成したり検証したりするため、メッセージダイジェストはデジタル署名アルゴリズムを使用して処理することができるようになります。

メッセージ自体ではなくメッセージダイジェストに署名を行うことで、多くの場合処理効率が向上します。これは通常、メッセージダイジェストの方がメッセージよりもはるかにサイズが小さいからです。デジタル署名の検証者は、デジタル署名の作成者が使用したのと同じハッシュアルゴリズムを使用する必要があります。

SHA-1、SHA-224、SHA-256 および MD5 は「安全」であるとされています。それはあるメッセージダイジェストに対応するメッセージの発見、もしくは同じメッセージダイジェストを生成する 2 つの異なるメッセージの発見が計算上不可能だからです。メッセージの転送中に何らかの変更が加えられた場合、極めて高い確率で異なるメッセージダイジェストになり、その署名は検証を通過できません。

36.3.4 メッセージデータの供給

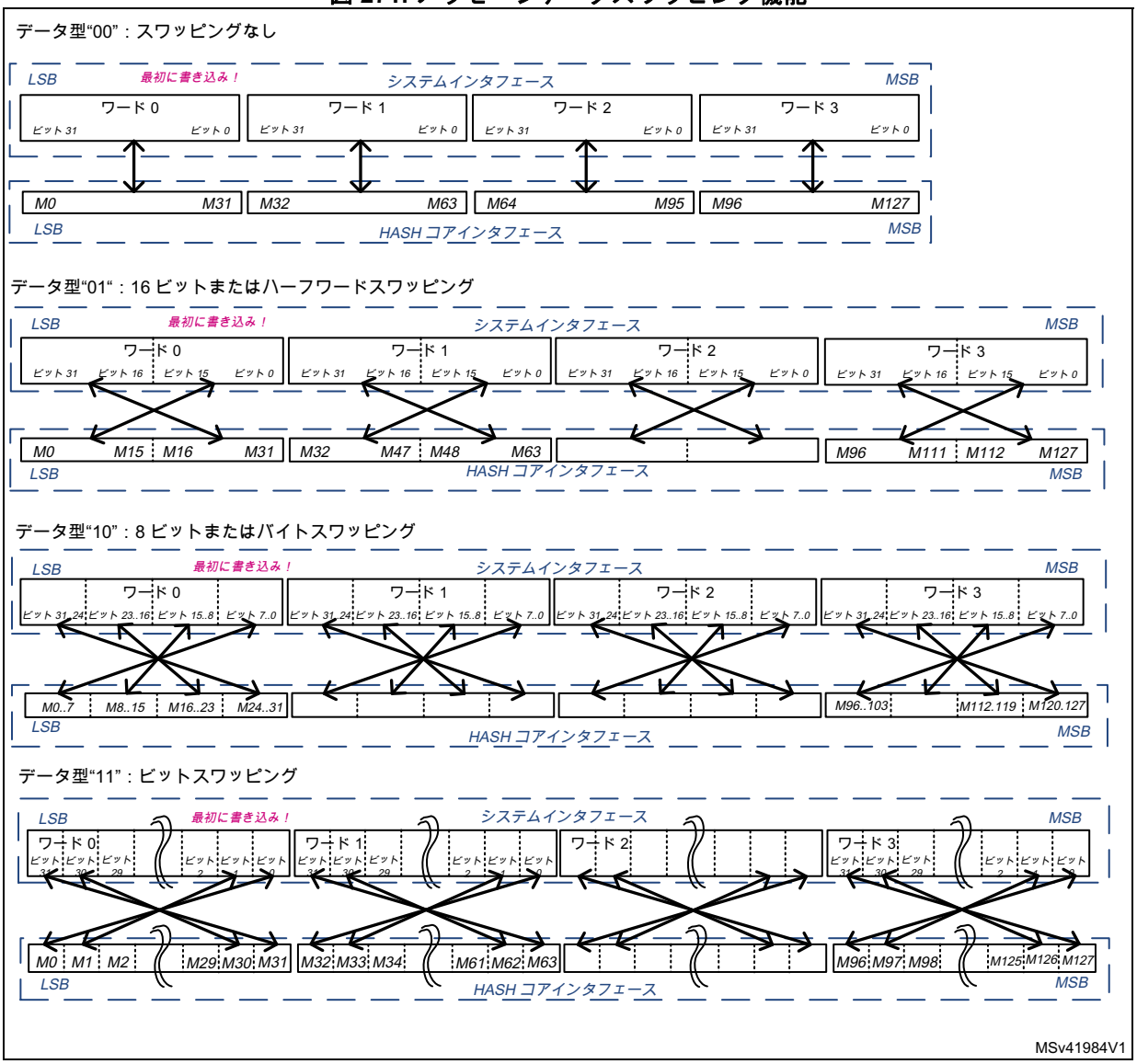
HASH で処理するメッセージ（またはデータファイル）はビット列とみなされます。FIPS PUB 180-1 および 180-2 標準により、このメッセージビット列はビッグエンディアン規約で表記された 16 進数ワードで左から右に表記されるため、最上位ビットは各ワード内で一番左のビット位置に格納されます。たとえば、「01100001 01100010 01100011」のビット列で表されるメッセージ文字列「abc」は、32 ビットワードで 0x00636261、8 ビットワードで 0x61626300 として表されます。

データは HASH_DIN レジスタに書き込むことにより 32 ビットワードごとに HASH に入力されます。HASH_DIN レジスタの現在の内容は、このレジスタに新しいデータが書き込まれるごとに、16 ワード入力 FIFO（IN FIFO）に転送されます。そのため、HASH_DIN と入力 FIFO で、17 の 32 ビットワード長の FIFO を形成します（IN バッファと呼びます）。

処理されるデータの種類（データが ASCII テキストストリームの場合のバイトスワッピングなど）に応じて、入力 FIFO からのデータをリトルエンディアンハッシュ処理コアに入力する前に、ビット、バイト、ハーフワードまたはスワッピングなしの動作を実行する必要があります。*図 271* に、HASH 制御レジスタ（HASH_CR）の DATATYPE ビットフィールドに従い、ハッシュ処理コア 32 ビットデータブロック M0～31 がドライバによって IN FIFO に出力される 32 ビットワードでどのように構成されるかを示します。

ビットスワッピングが無効（DATATYPE=“00”）になっている場合の HASH_DIN データのエンディアン形式は、次のように記載できます。メッセージの最下位ビットはハッシュプロセッサに入力される最初のワードの MSB の位置でなければならず、ビット列の 32 番目のビットはハッシュプロセッサに入力される 2 番目のワードの MSB の位置でなければなりません。それ以降は同様です。

図 271. メッセージデータスワッピング機能



36.3.5 メッセージダイジェストの計算

ハッシュプロセッサは、メッセージダイジェストを計算する際に、512 ビットのブロックを順次処理します。したがって、DMA または CPU により 16 × 32 ビットワード（= 512 ビット）がハッシュプロセッサに書き込まれるごとに HASH は自動的にメッセージダイジェストの計算を開始します。この操作は、「部分ダイジェスト計算」として知られています。

*セクション 36.3.4：メッセージデータの供給*に示すとおり、処理するメッセージは、HASH 32 ビットワードに入力され、入力 FIFO を満たすために HASH_DIN レジスタに書き込まれます。このデータでハッシュ計算を実行するために、以下の手順がアプリケーションによって使用されます。

1. HASH_CR レジスタを使用してハッシュプロセッサを初期化します。
   - ALGO フィールドを使用して正しいアルゴリズムを選択します。必要に応じて、HASH_CR の DATATYPE ビットフィールドを使用してメッセージ入力ワードに正しいスワッピング操作をプログラムします。
   - MODE=1 にセットして、HMAC モードを選択した場合は LKEY でキー長を選択します。
   - 有効ビット数が 32 ビットではない場合、NBLW を更新して最後のワードで有効ビット数を定義します。その場合、自動パディングが HASH によって適用される場合があります。
2. HASH_CR の INIT ビットを 1 にセットして、初期化を完了します。データを DMA で転送する場合、ビット DMAE も 1 にセットします。

注意： ***ステップ 2 でプログラムする際に、その前あるいはその時に正しい設定値（ALGO、DATATYPE、HMAC モード、キー長、NBLW）をセットアップすることが重要です。***

3. データが DMA で自動で転送されない場合、HASH_DIN レジスタに書き込んでデータ入力を開始します。ブロックの処理はブロックの最終値が IN_FIFO に入力された後で初めて開始できることに注意してください。部分または最終ダイジェスト計算の管理方法は、データのプロセッサへの供給方法によって異なります。

   a) データがソフトウェアで入力される場合：

   - 部分ダイジェスト計算は、ソフトウェアが HASH_DIN レジスタに追加のワード（実際には次のブロックの最初のワード）を書き込む際にトリガされます。プロセッサが再びレディ（HASH_SR の DINIS=1）になると、ソフトウェアは新しいデータを HASH_DIN に書き込めるようになります。このメカニズムによって、HASH によるウェイトステートの導入を回避します。
   - 最終ダイジェスト計算は、最終ブロックが入力され、ソフトウェアが DCAL ビットに 1 を書き込む際にトリガされます。メッセージ長が 512 ビットの整数倍ではない場合、DCAL ビットに書き込む前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、*セクション 36.3.6*を参照）。

   b) データが DMA によってシングル DMA 転送（MDMAT ビット = “0”）として入力される場合：

   - 部分ダイジェスト計算は、FIFO がフルになるたびに自動的にトリガされます。
   - 最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされます（DCAL ビットがハードウェアによって 1 にセットされます）。メッセージ長が 512 ビットの整数倍ではない場合、DMA を有効にする前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、*セクション 36.3.6*を参照）。

   c) データが多重 DMA 転送によって入力される（MDMAT = “1”）場合：

   - 部分ダイジェスト計算は、シングル DMA 転送に対してトリガされます。ただし、最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされません（DCAL ビットがハードウェアによって 1 にセットされません）。これにより、ハッシュプロセッサは、新しい DMA 転送をダイジェスト計算の一部として受信できます。最終ダイジェスト計算を開始するには、シングル DMA 転送に対して行うように最終ダイジェスト計算をトリガするために、最後の DMA 転送の前に、ソフトウェアで MDMAT ビットを 0 にセットする必要があります（上記参照）。

4. いったん計算が行われれば、ダイジェストは*表 278* に記載しているとおり、出力レジスタから読み出すことができます。

表 278. ハッシュプロセッサ出力

アルゴリズム	有効な出力レジスタ	最上位ビット	ダイジェストサイズ（ビット）
MD5	HASH_H0 から HASH_H3	HASH_H0[31]	128
SHA-1	HASH_H0 から HASH_H4	HASH_H0[31]	160
SHA-224	HASH_H0 から HASH_H6	HASH_H0[31]	224
SHA-256	HASH_H0 から HASH_H7	HASH_H0[31]	256

HMAC の詳細な説明については、*セクション 36.3.7 : HMAC 操作*を参照してください。

36.3.6 メッセージのパッディング

概要

メッセージの圧縮表記を計算する際に、データをハッシュプロセッサに（512 ビットブロックごとの自動部分ダイジェスト計算で）供給する処理が、オリジナルメッセージの最後のビットがHASH_DINレジスタに書き込まれるまでループします。

メッセージの長さ（ビット数）は任意の整数値をとることができるため、ハッシュプロセッサに書き込まれる最後のワードの有効ビット数は 1 から 32 まで可能です。最終メッセージダイジェスト計算の前にメッセージのパディングを正しく行うため、最後のワードの有効ビット数 NBLW をHASH_STR レジスタに書き込む必要があります。

パディング処理

DMA が有効または無効な状態での詳細なパディング手順については、*セクション 36.3.5 : メッセージダイジェストの計算*に記載しています。

パディングの例

連邦情報処理規格 PUB 180-1 および PUB 180-2 によって指定されているように、メッセージのパディングは、「1」の後に *k* 個の“0”が続き、その後にメッセージのビットの長さ *L* に等しい 64 ビット整数値が続きます。この 3 つのパディング操作により、長さ *L* + 1 + *k* + 64 のパディングされたメッセージが生成され、512 ビットの倍数となります。

ハッシュプロセッサでは、HASH_DIN レジスタに対して最後に書き込まれたワードの中の NBLW ビットフィールドで定義された位置に、「1」が追加され、残りの上位ビットは「0」にクリアされます。

FIPS PUB180-2 からの例

オリジナルメッセージが ASCII バイナリコード形式の「abc」で、長さ L = 24 であると仮定します。

```
byte 0    byte 1   byte 2   byte 3
01100001 01100010 01100011 UUUUUUUU
<-- HASH_DIN???????????-->
```

NBLW に値 24 をロードする必要があります。ビット列のビット位置 24（上記ビット列で左から右へカウントします）に「1」が付加されます。これは HASH_DIN レジスタのビット 31 に対応します（リトルエンディアン規約）：

```
01100001 01100010 01100011 1UUUUUUU
```

L = 24 であるため、上記のビット列のビット数は 25 ビットであり、423 個の「0」ビットが付加されて 448 ビットになります。

その結果、16 進数（ビッグエンディアン形式のバイトワード）で以下のようになります。

```
6162638 0 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000
```

2 ワード形式のメッセージ長の値 L（00000000 00000018）が付加されます。したがって、パディング処理により最終的に得られるメッセージは、16 進数（ビッグエンディアン形式のバイトワード）で次のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

ハッシュプロセッサが HASH_DIN 入力レジスタ内のバイトをスワップするようプログラムされている（HASH_CR で DATATYPE=10）場合、次の手順に従って上記メッセージを入力する必要があります。

1. HASH_DIN レジスタに 0xUU636261 を書き込みます（ここで'U' は無視を意味します）。
2. HASH_STR レジスタに 0x18 を書き込みます（オリジナルメッセージの長さは 24 ビットなので HASH_DIN レジスタに最後に書き込まれるワードの有効ビット数は 24 です）。
3. HASH_STR レジスタに 0x10 を書き込んでメッセージのパッディング処理（上記で説明）を開始し、ダイジェスト計算を実行します。
4. メッセージダイジェストが SHA-1 アルゴリズムに対して HASH_Hx レジスタ（x = 0～4）で使用できる状態で、ハッシュ計算が完了します。この FIPS の例の場合の期待値は次のとおりです。

```
HASH_H0 = 0xA9993E36
HASH_H1 = 0x4706816A
HASH_H2 = 0xBA3E2571
HASH_H3 = 0x7850C26C
HASH_H4 = 0x9CD0D89D
```

36.3.7 HMAC 操作

概要

インターネット技術タスクフォース *RFC2104*、*HMAC：メッセージ認証のための鍵付きハッシング* で指定されているように、HMAC アルゴリズムは、処理するメッセージをユーザが選択したキーに対して不可逆的に紐づけすることによるメッセージ認証に使用します。このアルゴリズムは 2 つのネストされたハッシュ操作で構成されています。

$$HMAC(message) = Hash((key \mid pad) \text{ XOR } [0x5C]_n \mid Hash((key \mid pad) \text{ XOR } [0x36]_n \mid message))$$

ここで、

- $[X]_n$は X を *n* 回繰り返すことを表し、*n*は SHA-1、SHA224、SHA-256、MD5 ハッシュアルゴリズムに対して 512 ビットで基礎となるハッシュ関数データブロックのサイズと等しくなります（例：n = 64）。
- pad は、上記で定義された長さ *n* までキーを拡張するために必要な 0 の連続です。キー長が *n* より大きくなる場合、アプリケーションはまず Hash() 関数を使用してキーをハッシュ処理して、その結果のバイト文字列を実際のキーとして HMAC に使用します。
- | は連結演算子を表します。

HMAC 処理

HMAC を計算するには、4 つの各手順が必要です。

1. MODE ビットが 1 の状態で ALGO ビットを望みのアルゴリズムに対応する値にセットして INIT ビットに 1 を書き込むことによりブロックを初期化します。使用するキーが 64 バイトを超える場合、LKEY ビットも 1 にセットする必要があります。この場合、HMAC の仕様で必要とされているように、ハッシュプロセッサは、実際のキーではなくキーのハッシュを使用します。
2. 内部ハッシュ関数に使用するキーをハッシュプロセッサに渡す必要があります。

   キーのローディング操作は、メッセージのビット列ローディングと同じメカニズムに従います。HASH_DIN にキーデータを書き込んで、HASH_STR レジスタに書き込むことによって転送を完了します。

注： ***エンディアン形式の詳細については、セクション 36.3.4：メッセージデータの供給を参照してください。***

3. 最後のキーのワードが入力され計算が開始されると、ハッシュプロセッサが内部キーマテリアルを算出します。この操作が完了すると、*セクション 36.3.4：メッセージデータの供給*に記載されているとおりメッセージのビット列を受け入れる準備ができます。
4. 最後のハッシュ処理終了後、ハッシュプロセッサは「レディ」を返して外部ハッシュ関数に使用するキーを受け取る用意ができたことを示します（通常、このキーは内部ハッシュ関数に使用するものと同じです）。キーの最後のワードが入力され計算が開始されると HASH_H0～HASH_H7 レジスタ内で HMAC の計算結果が確認できます。

注： ***HMAC 初期計算遅延は、セクション 36.5：HASH 処理時間に記載しているとおりキーとメッセージの長さによって変わります。***

HMAC の例

以下は、NIST によって指定されている HMAC SHA-1 アルゴリズム（HASH_CR で ALGO=“00”および MODE=“1”）の例です。

オリジナルメッセージが ASCII バイナリコード形式の「`Sample message for keylen=blocklen`」で、長さ L = 34 バイトであると仮定します。HASH がスワップなしモードでプログラムされている場合（HASH_CR で DATATYPE=00）、次のデータを HASH_DIN レジスタに連続でロードする必要があります。

1. NIST によって指定されている**内部ハッシュキー**入力（長さ = 64、パディングなし）。キー長 = 64 であるため、HASH_CR レジスタの LKEY ビットは 0 にセットされます。

```
00010203 04050607 08090A0B 0C0D0E0F 10111213 14151617
18191A1B 1C1D1E1F 20212223 24252627 28292A2B 2C2D2E2F
30313233 34353637 38393A3B 3C3D3E3F
```

2. **メッセージ**入力（長さ = 34、パディング必須）。HASH_STR を `0x20` にセットして、メッセージのパディングと内部ハッシュ計算を開始する必要があります（‘U’は無視とみなします）。

```
53616D70 6C65206D 65737361 67652066 6F72206B 65796C65
6E3D626C 6F636B6C 656EUUUU
```

3. **外部ハッシュキー**入力（長さ = 64、パディングなし）。ここで、内部ハッシュキーと同一のキーを入力します。
4. **最終外部ハッシュ計算**が、HASH によって実行されます。以下のように、HMAC-SHA1 が HASH_Hx レジスタ（x = 0～4）で使用できます。

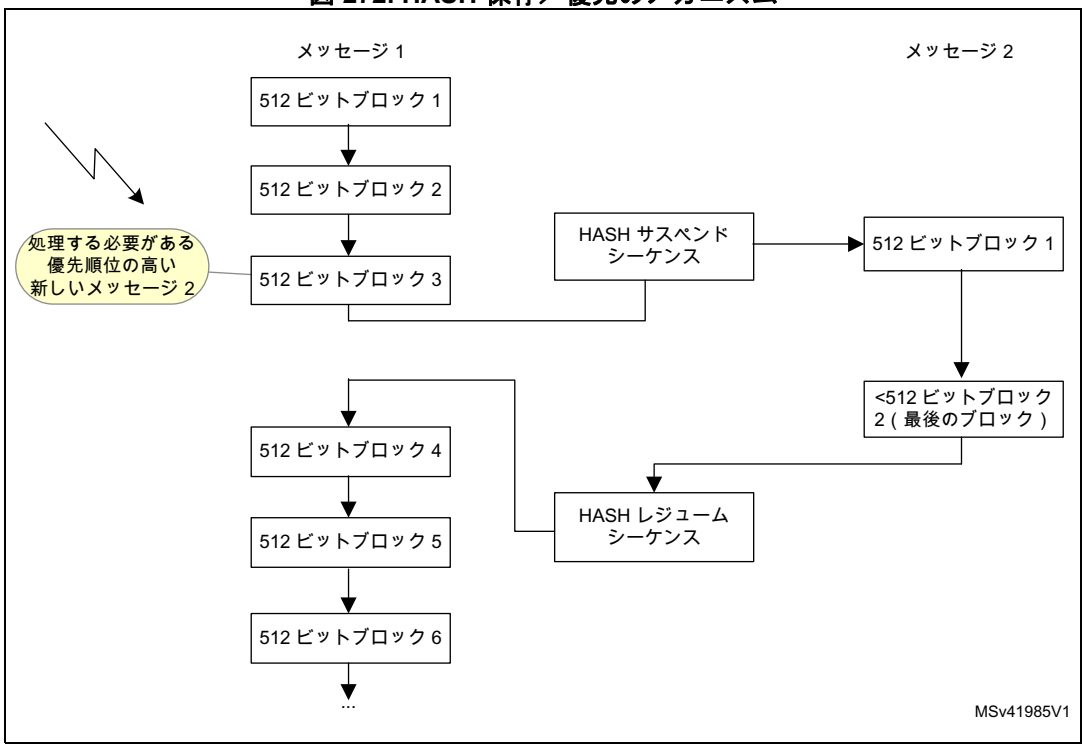
```
HASH_H0 = 0x5FD596EE
HASH_H1 = 0x78D5553C
HASH_H2 = 0x8FF4E72D
HASH_H3 = 0x266DFD19
HASH_H4 = 0x2366DA29
```

36.3.8 コンテキストスワッピング

概要

ハッシュ／HMAC 操作に割り込んでより優先度の高い別の処理を行うことができます。*図 272* に示すように、割り込まれた処理は優先度の高いタスクが処理された後で完了します。

図 272. HASH 保存／復元のメカニズム



これを行うには、割り込まれたタスクのコンテキストを HASH レジスタからメモリに保存し、その後 HASH レジスタからメモリに復元する必要があります。

データフローをソフトウェアまたは DMA によって制御する手順を以下に示します。

ソフトウェアによってロードされるデータ

DMA がハッシュプロセッサへのメッセージのロードに使用されていない場合、進行中のブロック処理がないときのみコンテキストを保存できます。つまり、ユーザアプリケーションは、DINIS = 1（最終ブロックの処理が完了し、入力 FIFO がエンプティの状態）または NBW ≠ 0（FIFO はフルではなく進行中の処理がない状態）になるまで待機する必要があります。詳細な手順は、以下のとおりです。

- **現在のコンテキストの保存**

  現在のメッセージダイジェスト計算に割り込む前に、アプリケーションは次のレジスタの内容をメモリに格納する必要があります。

  – HASH_IMR
  – HASH_STR
  – HASH_CR
  – HASH_CSR0 から HASH_CSR53

- **現在のコンテキストの復元**

  割り込まれたメッセージの処理を復元するには、アプリケーションは以下の手順に従う必要があります。

  a) メモリに保存した値を、HASH_IMR、HASH_STR および HASH_CR のレジスタに書き込みます。
  b) HASH_CR レジスタの INIT ビットをセットすることによって、ハッシュプロセッサを初期化します。
  c) メモリに保存した値を、HASH_CSR0 から HASH_CSR53 レジスタに書き込みます。
  d) 処理の割り込まれたところから再開します。

DMA によってロードされるデータ

DMA がハッシュプロセッサへのメッセージのロードに使用される場合、DMA 転送が進行中であるかどうか予測できません。したがって、ユーザアプリケーションが DMA 転送を停止して、現在のメッセージダイジェスト計算に割り込む前にハッシュプロセッサがレディ状態になるまで待機する必要があります。詳細な手順は、以下のとおりです。

- **現在のコンテキストの保存**

  DMA を使用して現在のメッセージダイジェスト計算に割り込む前に、アプリケーションは次の手順に従う必要があります。

  a) DMAE ビットをクリアして DMA インタフェースを無効にします。
  b) 現在の DMA 転送が完了するまで待ちます（HASH_SR レジスタ内で DMAS=0 になるのを待ちます）。ブロックが完全に HASH レジスタに転送されている場合と、そうでない場合があることに注意してください。
  c) DMA コントローラの対応するチャネルを無効にします。
  d) ハッシュプロセッサがレディになるまで待ちます（処理中のブロックがない状態）。つまり、DINIS=1 になるのを待ちます。

- **現在のコンテキストの復元**

  DMA を使用して割り込まれたメッセージの処理を復元するには、アプリケーションは以下の手順に従う必要があります。

  a) 再度割り込まれないうちに、引き続きメッセージを最後まで転送するために DMA コントローラを再設定します。
  b) DMAE ビットをセットすることにより処理が割り込まれたところから再開します。

注： ***コンテキストスワッピングが HMAC 操作を伴わない場合、HASH_CSR38 から HASH_CSR53 のレジスタを保存し復元する必要はありません。***

コンテキストスワッピングが 2 つのブロック間で発生する場合（最終ブロックが完全に処理されつつも、次のブロックはまだ IN FIFO にプッシュされていない状態。HASH_CR レジスタの NBW=000）、HASH_CSR22 から HASH_CSR37 のレジスタを保存し復元する必要はありません。

36.3.9 HASH DMA インタフェース

ハッシュプロセッサには、DMA コントローラに接続するインタフェースが搭載されています。この DMA は、HASH_CR レジスタの DMAE ビットをセットすることによって HASH にデータを書き込むために使用できます。このビットがセットされている場合、4 ワードのバーストをサポートする十分な空きワードが FIFO にある場合、HASH は DMA コントローラへのバーストリクエスト信号をアサートします。

4 つの 32 ビットワードを受信すると、HASH は自動的にこの処理を再開し、FIFO のサイズをチェックして、FIFO ステータスがバースト受信可能である場合に新しいリクエストをアサートします。詳細については、*セクション 36.3.5：メッセージダイジェストの計算*を参照してください。

DMA 転送を開始する前に、ソフトウェアは HASH_DIN レジスタにコピーされる最後のワードで有効ビット数をプログラムする必要があります。これは HASH_STR レジスタに次の値を書き込むことによって行います。

```
NBLW = Len(Message)% 32
```

ここで、「`x%32`」は、x を 32 で割ったあまりを示します。

HASH_SR レジスタの DMAS ビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE が 0 にクリアされ、DMA 転送中ではない場合にクリアされます。

注： ***DMAS ビットに関連する割り込みはありません。***

36.3.10 HASH エラー管理

HASH ハードウェアによって生成されるエラーフラグはありません。

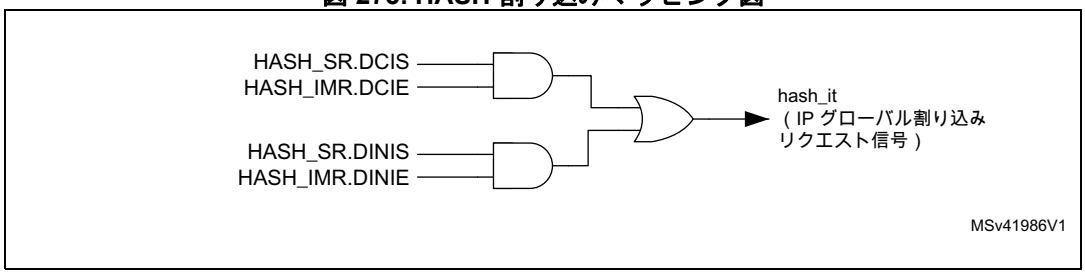
36.4 HASH 割り込み

マスク可能な 2 つの割り込みソースが個別に HASH プロセッサによって生成され、次のイベントが通知されます。

- ダイジェスト計算完了（DCIS）
- データ入力バッファレディ（DINIS）

両割り込みソースは、*図 273* に示すとおり同じグローバル割り込みリクエスト信号に接続されます。

図 273. HASH 割り込みマッピング図



HASH_IMR レジスタのマスクビットを変更することにより、上記の割り込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割り込みが有効になります。

個別の割り込みイベントのステータスは HASH_SR レジスタから読み出すことができます。*表 279* に、使用可能な機能の概要を示します。

表 279. HASH 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
ダイジェスト計算完了フラグ	DCIS	DCIE
新しいブロックフラグを取得するためのデータ入力バッファレディ	DINIS	DINIE

36.5 HASH 処理時間

表 280 に、各動作モードで 512 ビット中間ブロックの処理に必要な時間を示します。

表 280. 処理時間（クロックサイクル数）

動作モード	FIFO ロード(1)	計算フェーズ	合計
MD5	16	50	**66**
SHA-1	16	66	**82**
SHA-224	16	50	**66**
SHA-256	16	50	**66**

1. 16 ワードのブロックをプロセッサにロードするために必要な時間をこの値に追加する必要があります。

メッセージ（または HMAC のキー）の最終ブロックを処理するのに必要な時間はこれより長い場合があります。この時間は最終ブロックの長さとキー（HMAC モードの）の長さによって異なります。

中間ブロックの処理と比べて以下の倍数増える場合があります。

- ハッシュメッセージは **1～2.5 倍**
- HMAC 入力キーは～**2.5 倍**
- HMAC メッセージは **1～2.5 倍**
- ショートキーの場合の HMAC 出力キーは～**2.5 倍**
- ロングキーの場合の HMAC 出力キーは **3.5～5 倍**

36.6 HASH レジスタ

HASH コアは、いくつかの制御レジスタ、ステータスレジスタ、および 5 つのメッセージダイジェストレジスタと連携しています。これらのレジスタはすべて 32 ビットワード単位でのみアクセス可能です。それ以外は AHB2 エラーが発生します。

36.6.1 HASH 制御レジスタ（HASH_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	Res.	LKEY
													rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MDMAT	DINNE	NBW				ALGO[0]	MODE	DATATYPE		DMAE	INIT	Res.	Res.
		rw	r	r	r	r	r	rw	rw	rw	rw	rw	w		

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **ALGO[1]**：ビット 7 の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LKEY**：ロングキー選択

このビットによって、HMAC モードにおけるショートキー（≤ 64 バイト）またはロングキー（> 64 バイト）を選択します。

0：ショートキー（≤ 64 バイト）

1：ロングキー（> 64 バイト）

注： ***この選択は、INIT ビットがセットされ MODE = 1 の場合にのみ有効です。計算中にこのビットを変更しても影響はありません。***

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **MDMAT**：多重 DMA 転送

多重 DMA 転送が必要な大容量のファイルをハッシングするときに、このビットをセットします。

0：DCAL は DMA 転送の最後に自動的にセットされます。

1：DCAL は DMA 転送の最後に自動的にはセットされません。

ビット 12 **DINNE**：DIN ノットエンプティ

このビットは HASH_DIN レジスタに有効なデータがある場合（少なくとも 1 度は書き込まれたあと）にセットされます。このビットは INIT ビット（初期化）または DCAL ビット（前回のメッセージ処理の完了）のどちらかに 1 が書き込まれるとクリアされます。

0：データ入力バッファにデータがありません

1：入力バッファには少なくとも 1 ワードのデータがあります

ビット 11:8 **NBW**：既にプッシュされたワード数

このビットフィールドは、既に IN FIFO にプッシュされているメッセージ内のワード数を示します。NBW は、DINNE=1 のときに HASH_DIN レジスタに書き込みアクセスが行われると値が増加（+1）します。

NBW は、INIT ビットに 1 が書き込まれたとき、またはダイジェスト計算が開始されたとき（DCAL に 1 が書き込まれたか DMA 転送終了時）に 0 になります。

DMA を使用しない場合

0000 かつ DINNE=0 の場合：DIN バッファにプッシュされているワードはありません。（HASH_DIN レジスタと IN_FIFO の両方ともエンプティ）

0000 かつ DINNE=1 の場合：DIN バッファに 1 ワードがプッシュされています。（HASH_DIN レジスタには 1 ワードありで、IN_FIFO はエンプティ）

0001：DIN バッファに 2 ワードがプッシュされています。（HASH_DIN レジスタと IN_FIFO にそれぞれ 1 ワードあり）

...

1111：DIN バッファに 16 ワードがプッシュされています。

DMA を使用する場合

NBW には DMA によって IN FIFO にプッシュされている正確なワード数が格納されています。

ビット 18 および ビット 7 **ALGO[1:0]**：アルゴリズム選択

これらのビットによって SHA-1、SHA-224、SHA256 または MD5 のアルゴリズムを選択します。

00：SHA-1 アルゴリズムを選択

01：MD5 アルゴリズムを選択

10：SHA224 アルゴリズムを選択

11：SHA256 アルゴリズムを選択

注：　この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット 6 **MODE**：モード選択

このビットによって、選択したアルゴリズムに対して HASH または HMAC モードを選択します。

0：ハッシュモードを選択します

1：HMAC モードを選択します。使用するキーが 64 バイトを超える場合、LKEY をセットする必要があります。

注：　この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット 5:4 **DATATYPE**：データ型選択

これらのビットは、HASH_DIN レジスタに入力したデータのフォーマットを定義します。

00：32 ビットデータ。HASH_DIN に書き込まれたデータは順序を変えずに直接ハッシュ処理に使用されます。

01：16 ビットデータ、すなわちハーフワード。HASH_DIN に書き込まれたデータは 2 つのハーフワードとみなされ、ハッシュ処理に使用する前にスワップされます。

10：8 ビットデータ、すなわちバイト。HASH_DIN に書き込まれたデータは 4 バイトとみなされ、ハッシュ処理に使用する前にスワップされます。

11：ビットデータ、またはビット列。HASH_DIN に書き込まれたデータは 32 ビット（ビット位置 0 におけるデータ列の最初のビット）とみなされ、ハッシュ処理（ビット位置 31 におけるデータ列の最初のビット）に使用する前にスワップされます。

ビット 3 **DMAE**：DMA イネーブル

0：DMA 転送は無効です

1：DMA 転送は有効です。DMA リクエストは、ハッシュコアがデータを受け取る準備ができるとすぐに送られます。

このビットがセットされると、メッセージの最後のデータがハッシュプロセッサに書き込まれている間、ハードウェアによってクリアされます。

DMA 転送中にこのビットを 0 にセットすると、現在の転送はアボートされません。その代わり、転送が完了するか INIT に 1 が書き込まれるまで、HASH の DMA インタフェースは内部的に有効になっています。

INIT ビットを 1 にセットすることで、DMAE ビットはクリアされません。

ビット 2 **INIT**：メッセージダイジェスト計算の初期化

このビットに 1 を書き込むとハッシュプロセッサコアがリセットされるので、HASH は新しいメッセージのメッセージダイジェストを計算可能となります。

このビットに 0 を書き込んでも影響はありません。このビットを読み出すと常に 0 が返ります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

36.6.2 HASH データ入力レジスタ（HASH_DIN）

アドレスオフセット：0x04

リセット値：0x0000 0000

HASH_DIN はデータ入力レジスタです。このレジスタは 32 ビット幅です。このレジスタを使用して、512 ビットのブロック毎にメッセージを入力します。HASH_DIN レジスタにプログラムを行うと、AHB データバス上の値がハッシュコアに「プッシュ」され、このレジスタには AHB データバス上の新しい値が入ります。正しいメッセージフォーマットを取得するには、事前に HASH_CR レジスタの DATATYPE ビットを設定する必要があります。

HASH_DIN レジスタに 16 ワードのブロックが書き込まれると、以下のように中間ダイジェストの計算が開始されます。

- DMA を使用していない場合、HASH_DIN レジスタに新しいデータを書き込む（次のブロックの最初のワード）ことにより開始されます（中間ダイジェストの計算）。
- DMA を使用している場合は自動的に開始されます。

HASH_DIN レジスタに最後のブロックが書き込まれると、以下のように最終ダイジェストの計算（パディングを含む）が開始されます。

- HASH_STR レジスタの DCAL ビットに 1 を書き込むことにより開始されます（最終ダイジェストの計算）。
- DMA を使用し、かつ MDMAT ビットが 0 に設定されている場合は自動的に開始されます。

ダイジェストの計算（中間または最終）が進行中で、HASH_DIN レジスタへの新たな書き込みアクセスが実行される場合、ハッシュ計算が完了するまで AHB2 バスにウェイトステートが挿入されます。

HASH_DIN レジスタが読み出されると、このアドレスに書き込まれた最後のワードがアクセスされます（リセット後はゼロ）。

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

ビット 31:0 **DATAIN**：データ入力

このレジスタを読み出すと、レジスタの現在の内容を返します。

このレジスタに書き込むと、レジスタの現在の内容が IN FIFO にプッシュされ、レジスタには AHB データバス上の新しい値が入ります。

36.6.3 HASH スタートレジスタ（HASH_STR）

アドレスオフセット：0x08

リセット値：0x0000 0000

HASH_STR レジスタには 2 つの機能があります。

- ハッシュプロセッサに入力されたメッセージの最後のワードにおける有効ビット数を定義します（HASH_DIN レジスタに書き込まれた最後のデータの有効最下位ビット数）。
- DCAL ビットに 1 を書き込むことによりメッセージの最後のブロックの処理を開始します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW				
							w				rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DCAL**：ダイジェスト計算

このビットに 1 を書き込むと以前書き込まれた NBLW の値を使用してメッセージのパッディングが開始され、INIT ビットに最後に 1 が書き込まれてから IN FIFO に書き込まれたすべてのデータワードを使用して最終メッセージダイジェストの計算が開始されます。

このビットを読み出すと 0 が返されます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **NBLW**：最後のワードの有効ビット数

メッセージのビット列の最後のワードが HASH_DIN レジスタに書き込まれると、ハッシュプロセッサは内部データスワッピング後に以下に指定された有効ビットのみを取得します。

0x00：書き込まれた最後のデータのすべての 32 ビットが有効メッセージビットです。（M[31:0]）

0x01：書き込まれた最後のデータ（スワッピング後）の 1 ビットだけが有効です。（M[0]）

0x02：書き込まれた最後のデータ（スワッピング後）の 2 ビットだけが有効です。（M[1:0] ）

0x03：書き込まれた最後のデータ（スワッピング後）の 3 ビットだけが有効です。（M[2:0] ）

...

0x1F：書き込まれた最後のデータ（スワッピング後）の 31 ビットだけが有効です。（M[30:0] ）

上記メカニズムは、DCAL=0 のときのみ有効です。DCAL が 1 にセットされているときに NBLW ビットに書き込まれた場合、NBLW ビットフィールドは変更されません。つまり、同時に NBLW を設定しつつ DCAL をセットすることはできません。

NBLW ビットを読み出すと最後に NBLW に書き込まれた値が返されます。

36.6.4 HASH ダイジェストレジスタ（HASH_HR0..7）

これらのレジスタには、次のように指定されたメッセージダイジェストの計算結果が入っています。

1. SHA1 アルゴリズム仕様上、それぞれ H0、H1、H2、H3、および H4
   この場合、HASH_H5 から HASH_H7 のレジスタは使われず、ゼロが読み出されます。
2. MD5 アルゴリズム仕様上、それぞれ A、B、C、および D
   この場合、HASH_H4 から HASH_H7 のレジスタは使われず、ゼロが読み出されます。
3. SHA224 アルゴリズム仕様上、それぞれ H0 から H6
   この場合、HASH_H7 のレジスタは使われず、ゼロが読み出されます。
4. SHA256 アルゴリズム仕様上、それぞれ H0 から H7

すべてのケースで、ダイジェストの最上位ビットが HASH_H0[31] に格納されます。

ハッシュコアが中間ダイジェストまたは最終メッセージダイジェストを計算している最中にこれらのいずれかのレジスタへの読み出しアクセスが実行されると（DCAL ビットに 1 が書き込まれたとき）、この読み出し操作はハッシュ計算が完了するまで一時停止します。

注： ***H0、H1、H2、H3 および H4 のマッピングは 2 つのメモリ領域に複製されます。***

HASH_HR0

アドレスオフセット：0x0C および 0x310

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR1

アドレスオフセット：0x10 および 0x314

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR2

アドレスオフセット：0x14 および 0x318

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR3

アドレスオフセット：0x18 および 0x31C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H3															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H3															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR4

アドレスオフセット：0x1C および 0x320

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H4															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H4															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR5

アドレスオフセット：0x324

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H5															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H5															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR6

アドレスオフセット：0x328

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H6															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H6															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR7

アドレスオフセット：0x32C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H7															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H7															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

注： ***（INIT ビットに 1 を書き込むことによって）新しいビットストリームのダイジェスト計算を開始するとき、これらのレジスタは強制的にリセット値になります。***

36.6.5 HASH 割り込み有効レジスタ（HASH_IMR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DCIE**：ダイジェスト計算完了割り込みイネーブル
0：ダイジェスト計算完了割り込みは無効です
1：ダイジェスト計算完了割り込みは有効です

ビット 0 **DINIE**：データ入力割り込みイネーブル
0：データ入力割り込みは無効です
1：データ入力割り込みは有効です

36.6.6 HASH ステータスレジスタ（HASH_SR）

アドレスオフセット：0x24

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
												r	r	rc_w0	rc_w0

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **BUSY**：ビジービット

0：現在処理中のブロックはありません。
1：ハッシュコアが処理中のデータブロックがあります。

ビット 2 **DMAS**：DMA ステータス

このビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE=0、かつ DMA 転送中ではない場合にクリアされます。このビットに関連する割り込みはありません。
0：DMA インタフェースは無効（DMAE=0）、かつ転送中ではありません。
1：DMA インタフェースは有効（DMAE=1）、または転送中です。

ビット 1 **DCIS**：ダイジェスト計算完了割り込みステータス

このビットは、ダイジェストがレディーになったとき（メッセージ全体が処理済）、ハードウェアによってセットされます。このビットは、そこに 0 を書き込むか HASH_CR レジスタの INIT ビットに 1 を書き込むことによってクリアされます。
0：HASH_Hx レジスタに利用可能なダイジェストはありません。
1：ダイジェスト計算完了。HASH_Hx レジスタに利用可能なダイジェストがあります。HASH_IMR レジスタの DCIE ビットがセットされている場合、割り込みが生成されます。

ビット 0 **DINIS**：データ入力割り込みステータス

このビットは、入力バッファが新しいブロックを取得する準備ができたときにハードウェアによってセットされます（空き領域が16 か所）。このビットは、そこに 0 を書き込むか HASH_DIN レジスタに書き込みを行うことによってクリアされます。
0：入力バッファの空き領域は 16 か所未満です。
1：入力バッファに新しいブロックを取り込むことができます。HASH_IMR レジスタの DINIE ビットがセットされている場合、割り込みが生成されます。

36.6.7 HASH コンテキストスワップレジスタ（HASH_CSRx）

ハッシュプロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。ハッシュプロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクがハッシュプロセッサを使用する必要があるため、コンテキストスワップを実行する必要がある場合に役立ちます。

そのようなことが発生した場合、HASH_CSRx レジスタを読み出し、その値をシステムメモリ空間に保存する必要があります。このようにして、優先権のあるタスクがハッシュプロセッサを使用することができ、ハッシュ計算が完了した時点で、保存したコンテキストをメモリから読み出し、HASH_CSRx レジスタに書き戻すことができます。

HASH_CSR0

アドレスオフセット：0x0F8

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CS0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

HASH_CSRx（x= 1～53）

アドレスオフセット：0x0F8 + x * 0x4

リセット値：0x0000 0000

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

36.6.8 HASH レジスタマップ

表 9 に HASH レジスタマップの概要およびリセット値を示します。

表 281. HASH レジスタマップとリセット値

オフセット	レジスタ名とリセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	HASH_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	Res.	LKEY	Res.	Res.	MDMAT	DINNE	NBW			ALGO[0]	MODE	DATATYPE		DMAE	INIT	Res.	Res.
	リセット値														0		0			0	0	0	0	0	0	0	0	0	0	0		
0x04	HASH_DIN	DATAIN																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	HASH_STR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW				
	リセット値																								0				0	0	0	0	0
0x0C	HASH_HR0	H0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	HASH_HR1	H1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	HASH_HR2	H2																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	HASH_HR3	H3																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	HASH_HR4	H4																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	HASH_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCIE	DINIE
	リセット値																														0	0	
0x24	HASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
	リセット値																													0	0	0	1
0xF8	HASH_CSR0	CSR0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0xFC	HASH_CSR1	CSR1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...																																
0x1CC	HASH_CSR53	CSR53																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
予約済み																																
0x310	HASH_HR0	H0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x314	HASH_HR1	H1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x318	HASH_HR2	H2																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x31C	HASH_HR3	H3																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x320	HASH_HR4	H4																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x324	HASH_HR5	H5																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x328	HASH_HR6	H6																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x32C	HASH_HR7	H7																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

37 高分解能タイマ（HRTIM）

37.1 概要

高分解能タイマは高精度のタイミングで最大 10 個のデジタル信号を生成できます。主にスイッチモード電源や照明システムのような電力変換システムを駆動することを目的にしていますが、非常に細かいタイミングの分解能が期待される場合の汎用の用途にも使用できます。

モジュラー型のアーキテクチャにより、独立の波形または連動波形のどちらでも生成することができます。波形は（カウンタと比較ユニットを使用した）自己完結型のタイミングと、アナログまたはデジタルのフィードバックおよび同期信号などの広範囲な外部イベントによって定義されます。このため、多様な制御信号（PWM、位相シフト、コンスタント Ton、等）を生成したり、ほとんどの変換トポロジに対応することができます。

制御および監視の目的のために、タイマはタイミング測定の能力も持っており、内蔵の ADC および DAC コンバータにつながっています。最後に、タイマは軽負荷のマネージメントモードを備えており、安全に停止する目的のためにさまざまな障害を取り扱うことができます。

参考資料

37.2 主な特徴

- 複数のタイミングユニット
  - フル分解能はすべての出力で利用可能で、デューティサイクル、周波数、およびトリガされたワンパルスモードでのパルス幅を調整可能
  - 6 個の 16 ビットタイミングユニット（それぞれ独立のカウンタと 4 つの比較ユニットを装備）
  - どのタイミングユニットからも制御できる 10 個の出力、チャネルあたり最大 32 のセット／リセットソース
  - 1 または 2 スイッチを備える複数の独立コンバータか、少数の大型マルチスイッチ接続形態のどちらかに対応できるモジュラー型アーキテクチャ
- どのタイミングユニットでも利用可能な最大 10 個の外部イベント
  - プログラム可能な極性およびエッジ感度
  - 高速非同期モードでの 5 つのイベント
  - プログラム可能なデジタルフィルタを備えた 5 つのイベント
  - ブランキングおよびウィンドウイングモードによる誤ったイベントのフィルタリング
- 内蔵アナログペリフェラルへの複数リンク
  - ADC コンバータへの 4 つのトリガ
  - DAC コンバータへの 3 つのトリガ
  - アナログ信号調節用の 3 つのコンパレータ
- 多目的の保護スキーム
  - 5 つのフォールト入力をどのタイミングユニットにも結合し、関連付け可能
  - プログラム可能な極性、エッジ感度、およびプログラム可能なデジタルフィルタ
  - 共振コンバータ専用の遅延保護
- 複数の HRTIM インスタンスは外部同期入力／出力によって同期できます。
- 多様な出力ステージ
  - フル分解能デッドタイム挿入
  - プログラム可能な出力極性
  - チョッパモード
- 複数のコンバータで同期した軽負荷の動作を行わせるバーストモードコントローラ
- 7 つの割り込みベクトル、それぞれ最大 14 ソース
- 複数レジスタの更新のバーストモードで、最大 14 個のソースによる 6 つの DMA リクエスト

37.3 機能詳細

37.3.1 概要

HRTIM は次のサブ構成要素に分けることができます。

- マスタタイマ
- タイミングユニット（タイマ A～タイマ E）
- 出力ステージ
- バーストモードコントローラ
- すべてのタイマに共用される、外部イベントおよびフォールト信号の調整ロジック
- システムインタフェース

マスタタイマは 16 ビットのアップカウンタに基づいています。4 つの比較ユニット経由の 10 の出力のどれでもセット／リセットすることができ、また同期信号を 5 つのタイマユニットに供給します。主な目的は、それらタイマユニットを唯一のソースによって制御することです。マスタタイマが複数のユニット間の位相シフトを管理する典型的な適用例としては、インターリーブ型降圧コンバータがあります。

タイマユニットは独立に動作することも、マスタタイマも含めて他のタイマと連動して動作することもあります。各タイマには 2 つの出力用のコントロールがあります。出力のセット／リセットイベントは、タイミングユニットの比較レジスタによって、またはマスタタイマ、他のタイマ、あるいは外部イベントから到着するイベントによってトリガされます。

出力ステージにはいくつかの役目があります。

- 2 つの出力が相補 PWM モードに設定されているとき、デッドタイムを追加
- 変調信号の先頭にキャリア周波数を付加
- 出力を事前設定した安全なレベルに非同期にアサートすることによってフォールトイベントを管理

軽負荷の動作の場合、バーストモードコントローラが 1 つまたは複数のタイマを制御することができます。出力のアイドル状態だけでなく、バースト長および周期をプログラムすることが可能です。

外部イベントおよびフォールト信号の調整ロジックには次のものが含まれます。

- 入力選択 MUX（たとえば、デジタル入力か、所定の外部イベントチャネル向けのオンチップソースかの選択）
- 極性およびエッジ感度のプログラミング
- デジタルフィルタリング（10 チャネルのうち 5 チャネル）

システムインタフェースによって、HRTIM は以下の MCU の他の部分とのやり取りができます。

- CPU への割り込みリクエスト
- HRTIM の特定バーストモードなど、メモリへの／メモリからの自動アクセスのための DMA コントローラ
- ADC および DAC コンバータへのトリガ

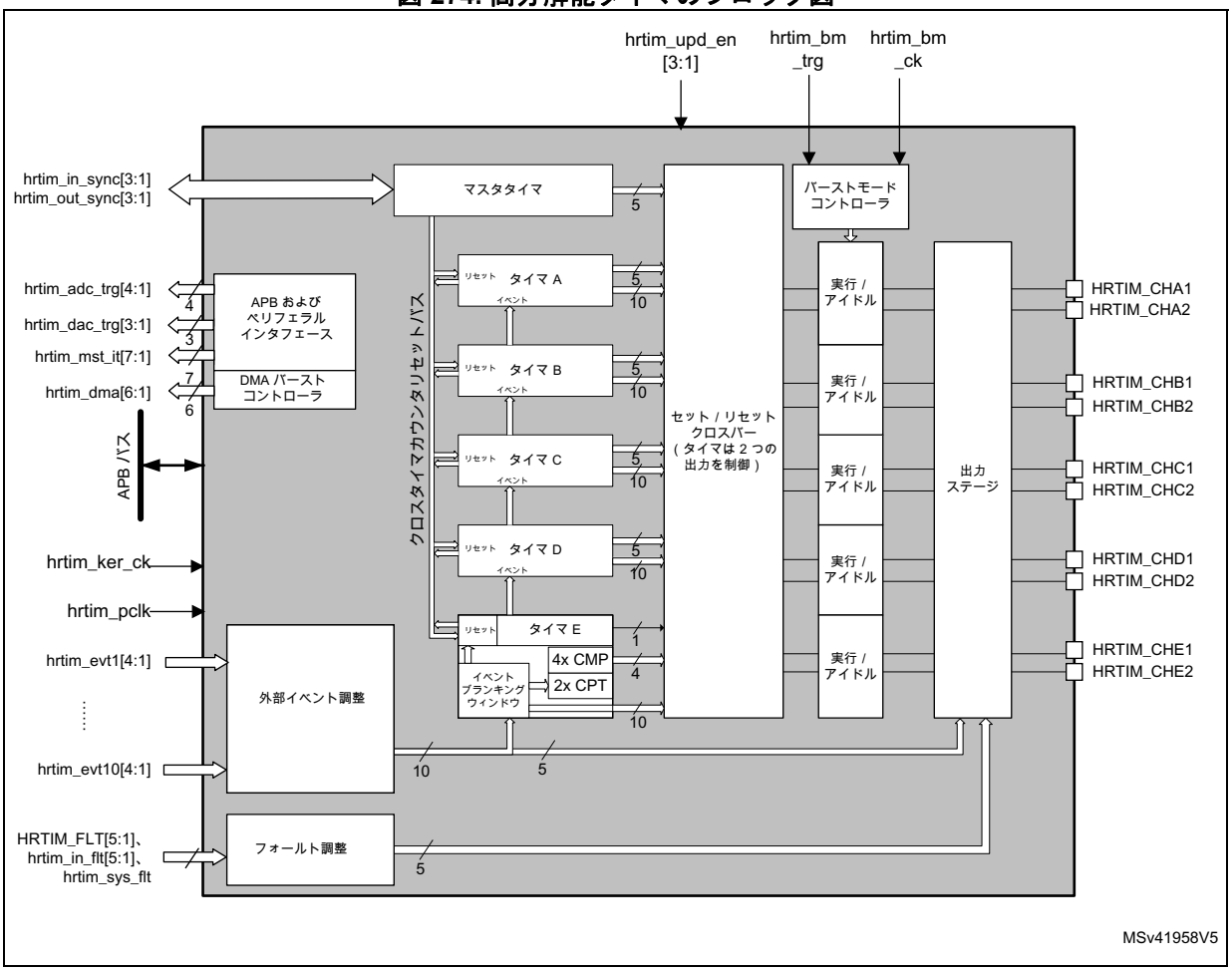
HRTIM レジスタは次の 7 つのグループに分かれます。

- マスタタイマのレジスタ
- タイマ A～タイマ E のレジスタ
- すべてのタイマユニットに共有される機能向けの共通レジスタ

注： ***表記法として、本文中およびレジスタ内で 5 つのタイミングユニットを参照する場合、「x」の文字を使用して一般化します。ここで、x は A～E のどの値も取ることができます。***

タイマのブロック図を*図 274*に示します。

図 274. 高分解能タイマのブロック図



37.3.2 HRTIM ピンおよび内部信号

次の表は、HRTIM の入力および出力（オンチップとオフチップの両方）の一覧です。

表 282. HRTIM の入出力の概要

信号名	信号タイプ	説明
HRTIM_CHA1、HRTIM_CHA2、HRTIM_CHB1、HRTIM_CHB2、HRTIM_CHC1、HRTIM_CHC2、HRTIM_CHD1、HRTIM_CHD2、HRTIM_CHE1、HRTIM_CHE2	出力	HRTIM タイマのメイン出力これらはデッドタイムを挿入してペア（HRTIM_CHx1 と HRTIM_CHx2）で連動させることも、独立に動作させることもできます。
HRTIM_FLT[5:1]、hrtim_in_flt[5:1]	デジタル入力	フォールト入力：アサートされた場合、直ちに HRTIM の出力を無効化します（5 つのオンチップ入力と 5 つのオフチップ HRTIM_FLTx 入力）。
hrtim_sys_flt	デジタル入力	MCU の内部フォールトを収集するシステム障害イベント（クロックセキュリティシステム、SRAM パリティエラー、Cortex®-M7 ロックアップ（ハードフォールト）、PVD 出力）
hrtim_in_sync[3:1]	デジタル入力	HRTIM 全体を下記の他の内部または外部タイマリソースと同期するための同期入力： hrtim_in_sync1：予約済み Hrtim_in_sync2：ソースは通常の TIMx タイマ（オンチップ相互接続経由） Hrtim_in_sync3：ソースは外部 HRTIM（HRTIM_SCIN 入力ピン経由）
hrtim_out_sync[2:1]	デジタル出力	この出力の目的は、いくつかの HRTIM インスタンスをカスケードにする、または同期することです（オンチップまたはオフチップ）。 hrtim_out_sync1：予約済み hrtim_out_sync2：出力先はオフチップの HRTIM またはペリフェラル（HRTIM_SCOUT 出力ピン経由）。
hrtim_evt1[4:1]	デジタル入力	外部イベント。10 個のイベントのそれぞれは、オンチップ（他の内蔵ペリフェラル：コンパレータ、ADC アナログウォッチドッグ、TIMx タイマ、トリガ出力）またはオフチップ（HRTIM_EEVx 入力ピン）の 4 つのソースから選択することが可能。
hrtim_evt2[4:1]		
hrtim_evt3[4:1]		
hrtim_evt4[4:1]		
hrtim_evt5[4:1]		
hrtim_evt6[4:1]		
hrtim_evt7[4:1]		
hrtim_evt8[4:1]		
hrtim_evt9[4:1]		
hrtim_evt10[4:1]		
hrtim_upd_en[3:1]	デジタル入力	HRTIM レジスタ更新イネーブル入力（オンチップ相互接続）はシャドウレジスタからアクティブレジスタへの転送をトリガします。
hrtim_bm_trg	デジタル入力	バーストモードトリガイベント（オンチップ相互接続）
hrtim_bm_ck[4:1]	デジタル入力	バーストモードクロック（オンチップ相互接続）

表 282. HRTIM の入出力の概要（続き）

信号名	信号タイプ	説明
hrtim_adc_trg[4:1]	デジタル出力	ADC 変換開始トリガ
hrtim_dac_trg[3:1]	デジタル出力	DAC 変換更新トリガ
hrtim_mst_it[7:1]	デジタル出力	割り込みリクエスト
hrtim_dma[6:1]	デジタル出力	DMA リクエスト
hrtim_pclk	デジタル入力	APB クロック
hrtim_ker_ck	デジタル入力	HRTIM カーネルクロック（以後、f_{HRTIM} と呼ぶ）

37.3.3 クロック

HRTIM は、フル分解能を提供するために、t_{HRTIM} システムクロックによって供給される必要があります。HRTIM にあるすべてのクロックはこのリファレンスクロックから導き出されます。

用語の定義

f_{HRTIM}：HRTIM メインクロック（hrtim_ker_ck）。以下に続くすべてのクロックはこのソースから導き出され、それに同期します。

f_{DTG}：デッドタイムジェネレータクロック。便宜上、本書では t_{DTG} 周期（$t_{DTG} = 1/f_{DTG}$）のみを使用します。

f_{CHPFRQ}：チョッパステージのクロックソース。

f_{1STPW}：チョッパモードでの最初のパルスの長さを定義するクロックソース。便宜上、本書では t_{1STPW} 周期（$t_{1STPW} = 1/f_{1STPW}$）のみを使用します。

f_{BRST}：バーストモードコントローラカウンタクロック。

$f_{SAMPLING}$：フォールトまたは外部イベント入力をサンプリングするのに必要なクロック。

f_{FLTS}：f_{HRTIM} から導き出されるクロックで、フォールトイベントをフィルタリングするために $f_{SAMPLING}$ のソースとして使用されます。

f_{EEVS}：f_{HRTIM} から導き出されるクロックで、外部イベントをフィルタリングするために $f_{SAMPLING}$ のソースとして使用されます。

f_{pclk}（hrtim_pclk）：レジスタの読み出し／書き込みアクセスに必要な APB バスクロック

タイマクロックおよびプリスケーラ

HRTIM の各タイマはそれぞれ独自の個別クロックプリスケーラを備えており、タイマの分解能を調整することができます（*表 283*を参照）。

表 283. タイマ分解能および最小 PWM 周波数（f_{HRTIM} = 400 MHz の場合）

CKPSC[2:0](1)	プリスケーリング比	$f_{COUNTER}$	分解能	最小 PWM 周波数
101	1	400 MHz	2.5 ns	6.1 kHz
110	2	400/2 MHz = 200 MHz	5 ns	3.05 kHz
111	4	400/4 MHz = 100MHz	10 ns	1.5 kHz

1. CKPSC[2:0] の値 000～100 は予約済み。

フル分解能は、エッジ位置決め、PWM 周期調整、および外部からトリガされるパルス幅に対して利用可能です。

初期化

起動時に、比較および周期レジスタを書き込む前に、最初にプリスケーラのビットフィールドを初期化することが必須です。タイマが有効化されると（HRTIM_MCR レジスタの MCEN または TxCEN のビット設定）、プリスケーラは変更できなくなります。

複数のタイマを有効化するときは、各プリスケーラは最初に起動したタイマのプリスケーラに同期します。

警告： **カウンタおよび出力動作が他のタイマの情報および信号に依存しない場合のみ、マスタタイマおよび TIMA～E タイマに異なるプリスケーリング比を与えることができます。次のイベントのうちの 1 つが 1 つのタイミングユニット（またはマスタタイマ）から別のタイミングユニットに伝播されるときは、これらのタイマでは同一のプリスケーリング比に設定することが必須です。そのイベントは、出力セット／リセットイベント、カウンタリセットイベント、更新イベント、外部イベントフィルタ、またはキャプチャトリガです。プリスケーラ分周比が異なると、予測できない結果になります。**

デッドタイムジェネレータクロック

デッドタイムプリスケーラは f_{HRTIM} / 8 / $2^{(DTPRSC[2:0])}$ で与えられ、HRTIM_DTxR レジスタの DTPRSC[2:0] ビットでプログラムされます。

t_{DTG} は、f_{HRTIM} = 400 MHz に対して 2.5 ns～20 ns の範囲の値を取ります。

チョッパステージクロック

チョッパステージクロックのソース f_{CHPFRQ} は、16～256 の範囲の分周比を使って f_{HRTIM} から導き出されます。したがって、f_{HRTIM} = 400 MHz に対して、1.56 MHz <= f_{CHPFRQ} <= 25 MHz となります。

t_{1STPW} はチョッパモードでの最初のパルスの長さですが、HRTIM_CHPxR レジスタの STRPW[3:0] ビットで次のようにプログラムされます。

t_{1STPW} = (STRPW[3:0]+1) x 16 x t_{HRTIM}

クロックソースとして f_{HRTIM} / 16 を使用しています（f_{HRTIM}= 400 MHz に対して 25 MHz）。

バーストモードプリスケーラ

バーストモードコントローラのカウンタクロック f_{BRST} はいくつかのソースから供給されます。その中の 1 つは f_{HRTIM} から導き出されます。

この場合、f_{BRST} の範囲は f_{HRTIM}～f_{HRTIM} / 32768（f_{HRTIM} = 400 MHz に対して 12.2 kHz）となります。

フォールト入力サンプリングクロック

フォールト入力ノイズ低減フィルタは $f_{SAMPLING}$ （f_{HRTIM} または f_{FLTS} のどちらかになります）で定義される時定数を持ちます。

f_{FLTS} は f_{HRTIM} から導き出され、その範囲は f_{HRTIM} = 400 MHz に対して 400 MHz～50 MHz となります。

外部イベント入力サンプリングクロック

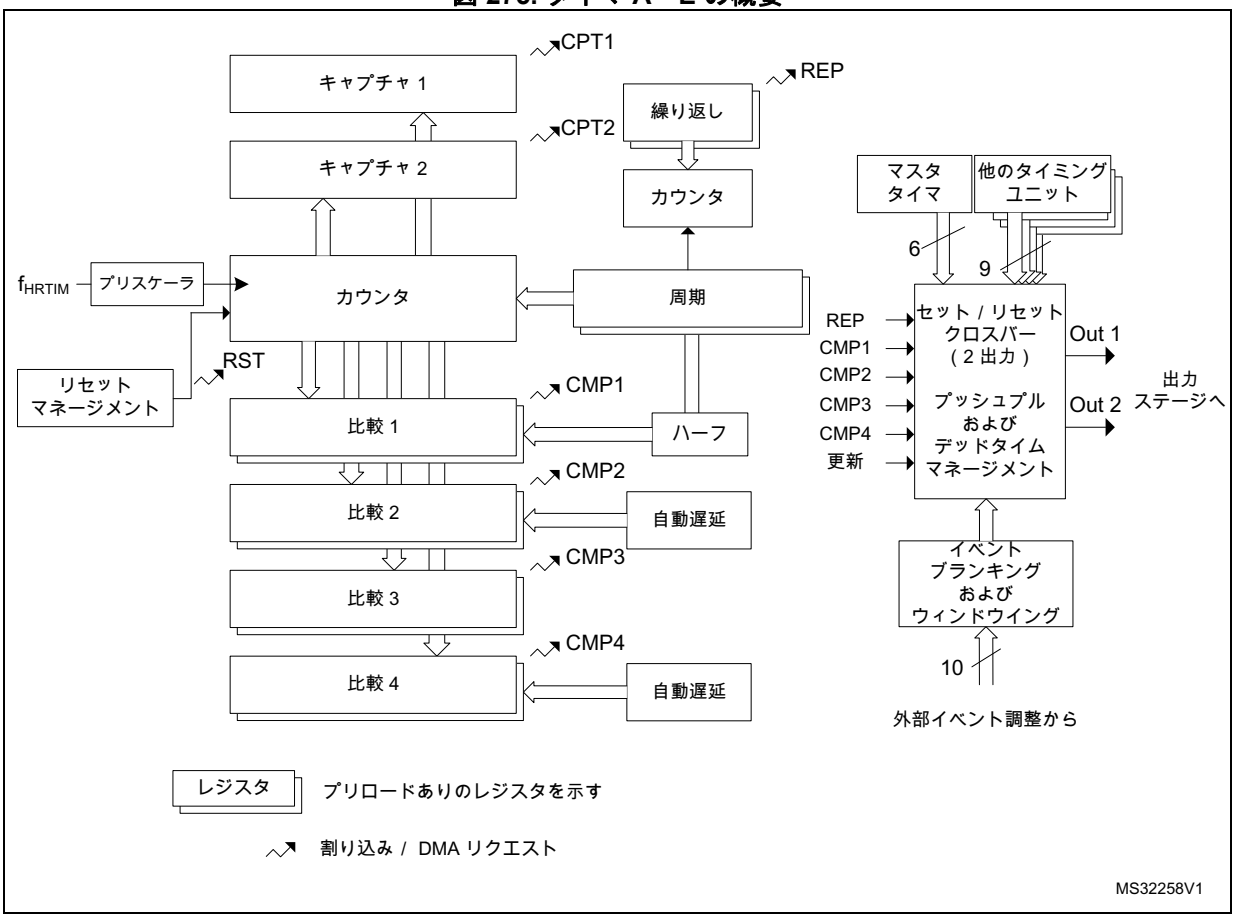
フォールト入力ノイズ低減フィルタは $f_{SAMPLING}$ （f_{HRTIM} または f_{EEVS} のどちらかになります）で定義される時定数を持ちます。

f_{EEVS} は f_{HRTIM} から導き出され、その範囲は f_{HRTIM} = 400 MHz に対して 400 MHz～50 MHz となります。

37.3.4 タイマ A～E のタイミングユニット

HRTIM は、*図 275*のように、カウント周期を定義するための自動再ロードメカニズムを持つ 16 ビットのアップカウンタと、4 つの比較ユニット、2 つのキャプチャユニットから成る 5 つの同じタイミングユニットを内蔵しています。各ユニットは、スタンドアローンのタイマとして動作できるように、2 つの出力のためのすべての制御機能を装備しています。

図 275. タイマ A～E の概要



周期および比較値は高分解能の実装に関連する下限値および上限値以内でなければならず、*表 284*に示すとおりです。

- 最小値は、f_{HRTIM} クロックの 3 周期以上である必要があります。
- 最大値は、f_{HRTIM} クロックの 0xFFFF - 1 周期以下である必要があります。

表 284. 周期および比較レジスタの最小値および最大値

CKPSC[2:0] の値(1)	最小値	最大値
≥ 5	0x0003	0xFFFD

1. CKPSC[2:0] の値 < 5 は予約済み。

注： ***比較値が周期のレジスタ値より大きいと、比較一致イベントは発生しません。***

カウンタ動作モード

タイマ A～E は連続モード（フリーランニングモード）またはシングルショット方式で動作することができ、HRTIM_TIMxCR 制御レジスタの CONT ビットを使用して、リセットイベントによりカウントが始まります。追加の RETRIG ビットによって、シングルショット動作を再トリガ可能にするか再トリガ不可能にするかを選択できます。動作の詳細を*表 285* および、*図 276* と *図 277* にまとめます。

表 285. タイマ動作モード

CONT	RETRIG	動作モード	開始／停止条件 クロック供給およびイベント生成
0	0	シングルショット 再トリガ不可能	TxEN ビットをセットすると、タイマが有効になりますが、カウンタは開始しません。 最初のリセットイベントによってカウントが始まり、カウンタが PER 値に達するまで後続のリセットは無視されます。 その後、PER イベントが生成され、カウンタは停止します。 リセットイベントがカウント動作を 0x0000 から再開始します。
0	1	シングルショット 再トリガ可能	TxEN ビットをセットすると、タイマが有効になりますが、カウンタは開始しません。 カウンタが停止していれば、リセットイベントはカウントを開始し、そうでなければカウンタをクリアします。カウンタが PER の値に達すると、PER イベントが生成され、カウンタは停止します。 リセットイベントがカウント動作を 0x0000 から再開始します。
1	X	連続モード	TxEN ビットをセットすると、タイマが有効になり、同時にカウンタが開始されます。 カウンタが PER の値に達すると、カウンタは 0x0000 に戻され、再開されます。 カウンタはいつでもリセットすることができます。

TxEN ビットはいつでもクリアでき、タイマを無効にしてカウントを停止させることができます。

図 276. 連続タイマ動作

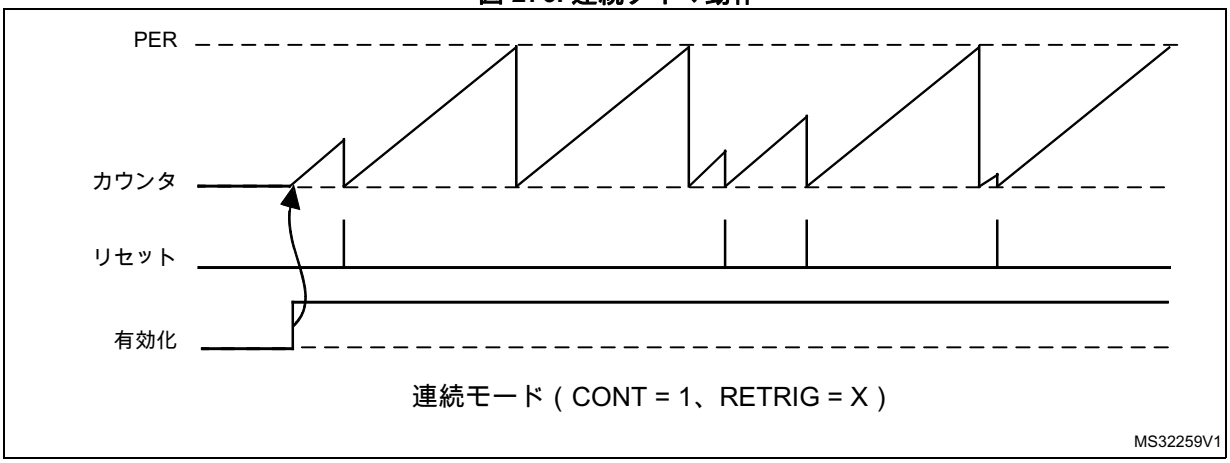
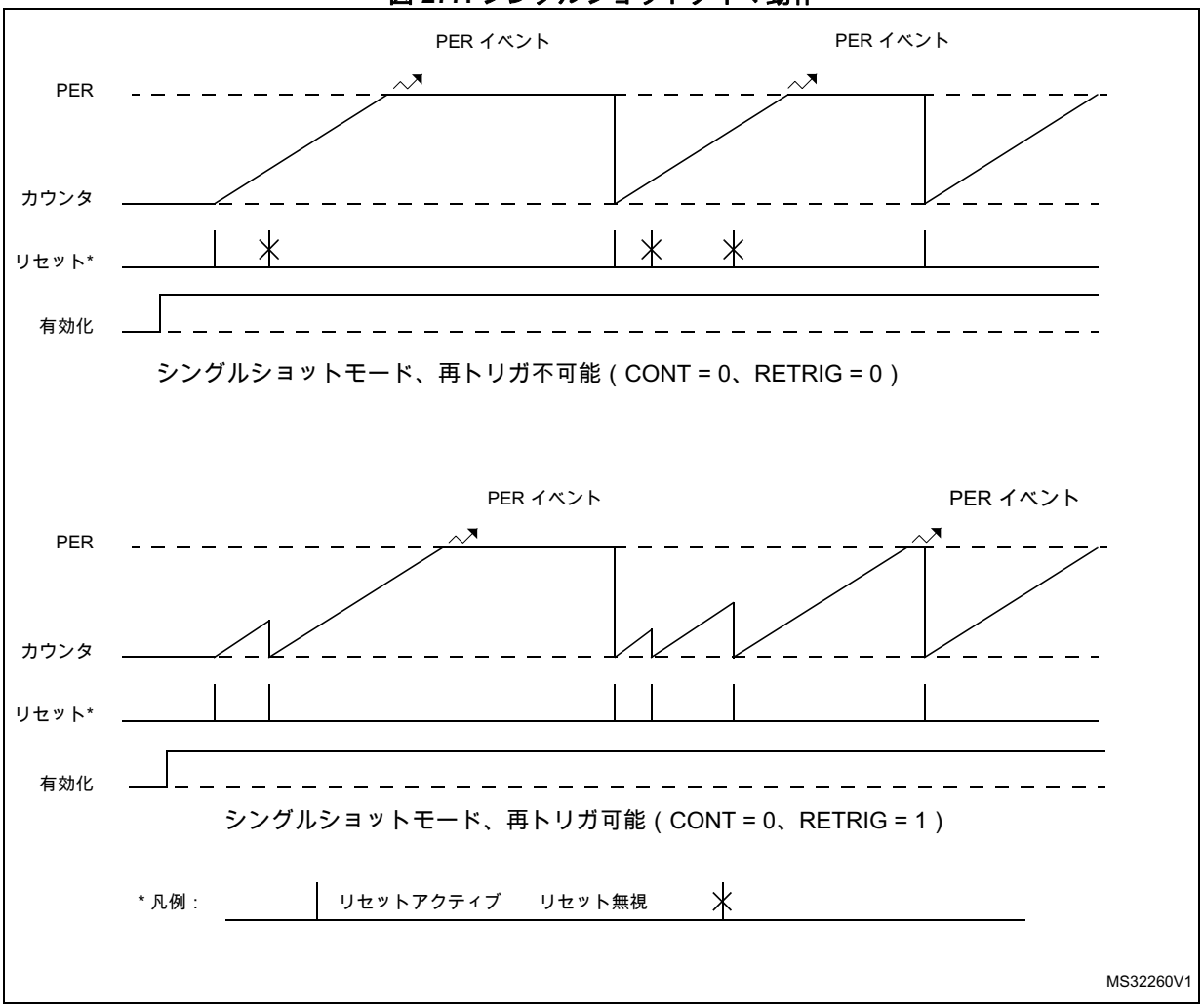


図 277. シングルショットタイマ動作



ロールオーバーイベント

連続モードにおいて、カウンタが HRTIM_PERxR レジスタで設定された周期値に達した後、0 に戻る時、カウンタのロールオーバーイベントが生成されます。

このイベントは、HRTIM において下記の複数の目的で使用されます。

- 出力をセット／リセットする
- レジスタの内容を更新するトリガとして（プリロードからアクティブへ転送）
- IRQ または DMA リクエストをトリガする
- バーストモードのクロックソースまたはバースト開始トリガとして働く
- ADC トリガとして
- 繰り返しカウンタをデクリメントする

タイマを開始する時、カウンタの初期値が周期の値より大きい場合、あるいはカウンタがすでにこの値より大きいときに新しい周期が設定される場合、カウンタはリセットされず、周期の最大値でオーバーフローし、繰り返しカウンタはデクリメントしません。

タイマのリセット

タイミングユニットのカウンタのリセットは最大 30 個のイベントによってトリガできます。それらは以下のソースから HRTIM_RSTxR レジスタで同時に選択できます。

- タイミングユニット：比較 2、比較 4 および 更新（3 つのイベント）
- マスタタイマ：リセットおよび比較 1～4（5 つのイベント）
- 外部イベント EXTEVNT1～10（10 個のイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～E）：比較 1、2 および 4（12 個のイベント）

いくつかのイベントを同時に選択して複数のリセットソースを扱うことができます。この場合、複数のリセットリクエストは論理和がとられます。同じ f_{HRTIM} クロックサイクル内に 2 つのカウンタリセットイベントが発生した場合、最後のカウンタリセットが考慮されます。

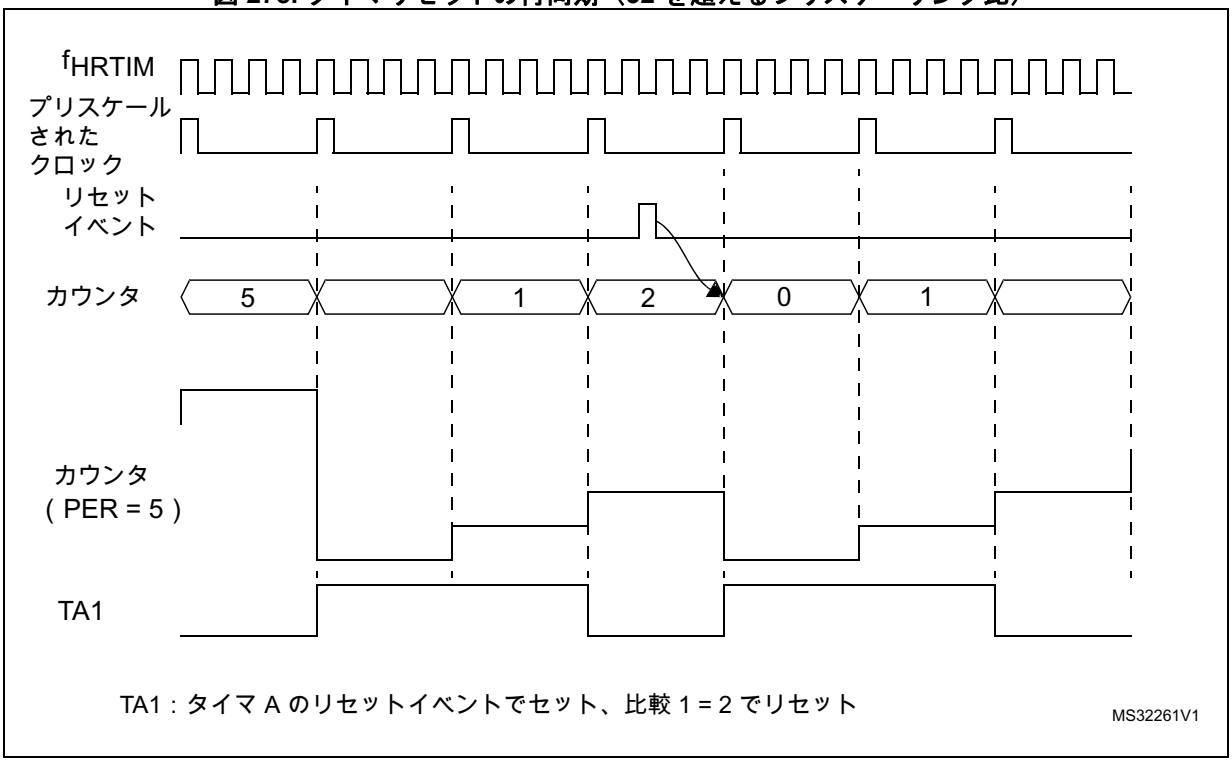
さらに、HRTIM_CR2 レジスタの TxRST ビットを使用して、カウンタのソフトウェアリセットをすることが可能です。これらの制御ビットは 1 つのレジスタ内にグループ化されているので、複数のカウンタを同時にリセットすることができます。

リセットリクエストは、関連カウンタが有効化（TxCEN ビットをセット）されてからのみ、考慮されます。

f_{HRTIM} クロックプリスケーリング比が 1 以上のとき、カウンタリセットイベントは、プリスケールされたクロックの次のアクティブエッジまで遅延されます。これによって、出力の遷移がリセットイベントに同期されているとき（コンスタント Ton 時間コンバータに典型的）、ジッタレス波形生成を維持することができます。

図 278 に、クロックのプリスケーリング比が 4 の（f_{HRTIM} を 4 で割った）場合に、リセットがどのように処理されるかを示します。

図 278. タイマリセットの再同期（32 を超えるプリスケーリング比）



繰り返しカウンタ

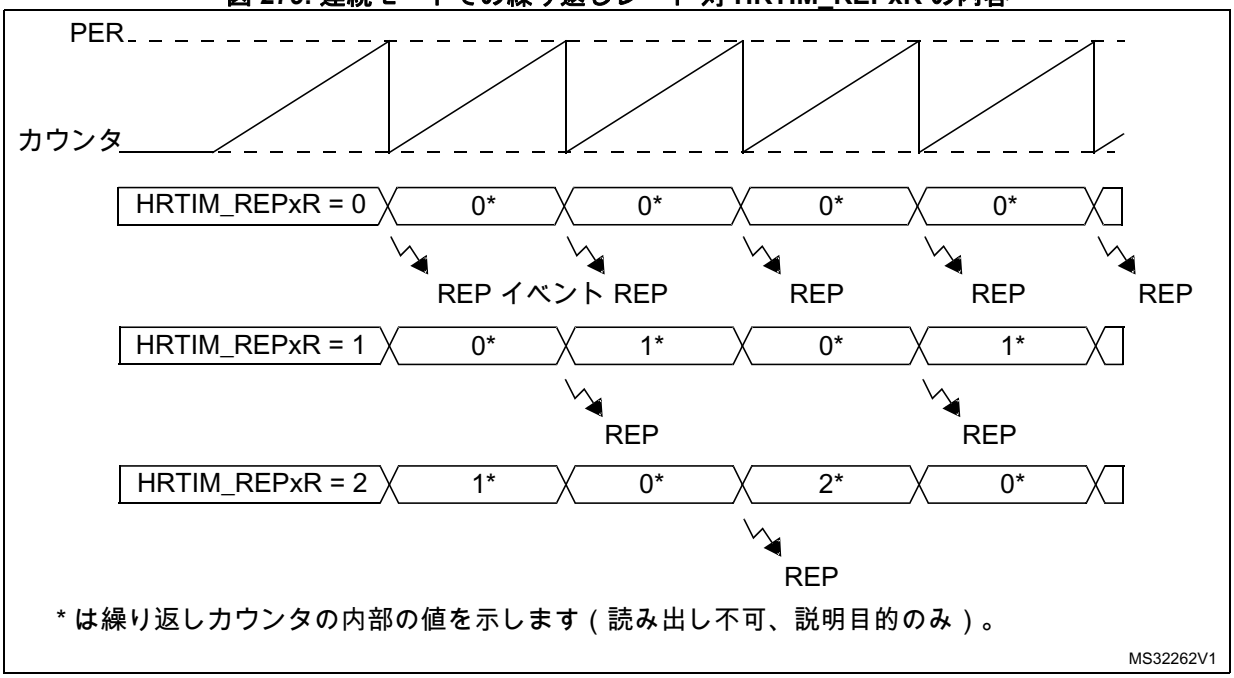
一般的なソフトウェアによる手法では、次の周期が始まる前に、処理時間の最大量が残るように、周期値に達したら割り込みを発生させます。繰り返しカウンタの主な目的は、周期割り込みレートを調整し、スイッチング周波数と割り込み周波数を分離することによって CPU の負荷を削減することです。

タイミングユニットは繰り返しカウンタを備えています。このカウンタは読み出しはできず、HRTIM_REPxR レジスタの自動再ロードの値でプログラムすることのみができます。

繰り返しカウンタは、タイマを有効にするとき（TXCEN ビットをセット）、HRTIM_REPxR レジスタの内容で初期化されます。いったんタイマを有効にすると、リセットイベントまたはカウンタのロールオーバーによってカウンタがクリアされる時に、繰り返しカウンタはデクリメントされます。カウンタがゼロに達すると、有効化されている場合（HRTIM_DIER レジスタの REPIE および REPDE ビット）、REP 割り込みまたは DMA リクエストが発行されます。

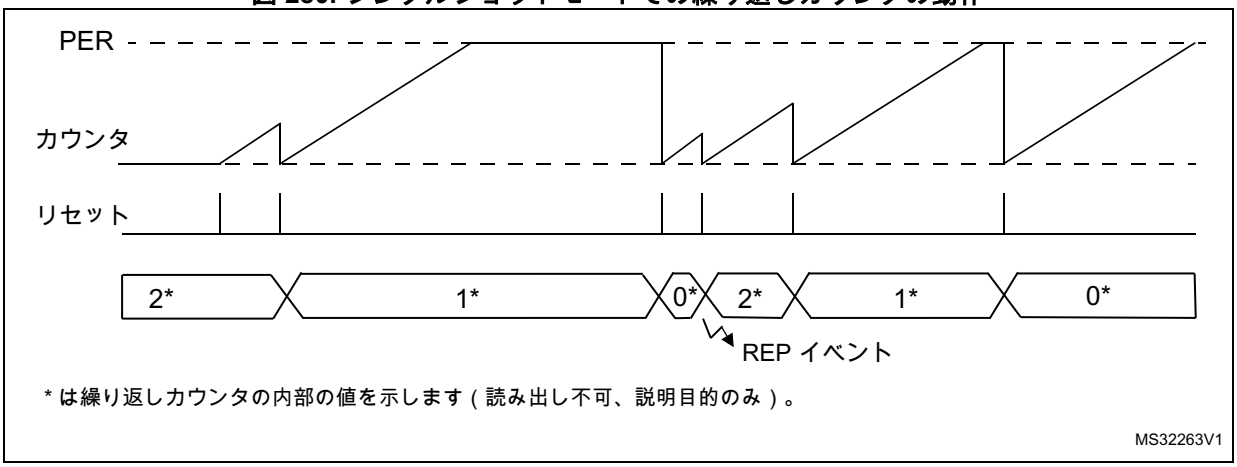
HRTIM_REPxR レジスタが 0 に設定されている場合、すべての周期ごとに割り込みが発生します。0 より大きい値では、（HRTIM_REPxR + 1）周期 の後に REP 割り込みが発生します。*図 279* に、連続モードでの種々の値に対する繰り返しカウンタの動作を示します。

図 279. 連続モードでの繰り返しレート 対 HRTIM_REPxR の内容



連続モードまたはシングルショットモード（以下の*図 280*）のどちらにおいても、カウンタが周期値に達する前にリセットされるとき（可変周波数動作）にも繰り返しカウンタが使用できます。リセットすると、繰り返しカウンタはデクリメントされます。ただし、カウンタ有効化（TxCEN ビットのセット）の直後の、最初の開始は除きます。

図 280. シングルショットモードでの繰り返しカウンタの動作



hrtim_in_sync[3:1] ソースからのリセットまたは開始イベントは、他のリセットと同じく、繰り返しカウンタのデクリメントを引き起こします。しかし、SYNCIN で開始されたシングルショットモード（HRTIM_TIMxCR レジスタの SYNCSTRTx ビットのセット）では、繰り返しカウンタは、周期の後の最初のリセットイベント時のみデクリメントされます。カウンタが hrtim_in_sync[3:1] の入力による新しいリクエストによって再開始されるまで、後続のリセットが来ても繰り返しカウンタは変更されません。

セット／リセットクロスバー

「セット」イベントは出力のアクティブ状態への遷移に対応し、「リセット」イベントは出力の非アクティブ状態への遷移に対応します。
波形の極性は、正または負論理の外部コンポーネントに適応させるために出力ステージで定義されます。アクティブレベルは、正の極性（POLx = 0）に対して論理レベル 1 に対応し、負の極性（POLx = 1）に対しては論理レベル 0 に対応します。

各タイミングユニットは 2 つの出力のためにセット／リセットクロスバーを操作します。これらの 2 つの出力は、以下のソースから選択できる最大 32 個のイベントによってセット、リセット、またはトグル反転ができます。

- タイミングユニット：周期、比較 1～4、レジスタ更新（6 つのイベント）
- マスタタイマ：周期、比較 1～4、HRTIM 同期（6 つのイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～E）：TIMEVNT1～9（*表 286* に記載された 9 つのイベント）
- 外部イベント EXTEVNT1～10（10 個のイベント）
- ソフトウェアの強制（1 つのイベント）

イベントスースは論理和がとられ、複数のイベントが同時に選択できます。

各出力は 2 つの 32 ビットレジスタによって制御されます。1 つはセット用（HRTIM_SETxyR）で、もう 1 つはリセット用（HRTIM_RSTxyR）です。ここで、x はタイミングユニット：A～E を表し、y は出力 1 または 2 を表します（たとえば、HRTIM_SETA1R、HRTIM_RSTC2R、など）。

同じイベントがセット用とリセット用の両方に選択された場合は、出力をトグル反転させます。出力状態をトグル反転することは t_{HRTIM} 周期あたり 1 回しかできません。同じサイクル内で 2 つの連続するトグル反転イベントが起きた場合、最初のイベントのみ考慮されます。

セットおよびリセットリクエストは、カウンタが有効にされて（TxCEN ビットをセット）からのみ考慮されます。ただし、ソフトウェアがタイマ起動時に出力の事前設定を許すリクエストを強制している場合を除きます。

表 286 に、出力をセット・リセットするために使用できる、他のタイミングユニットからのイベントをまとめます。数字はレジスタに挙がっているタイマイベント（TIMEVNTx など）に対応し、空の場所は利用できないイベントを示します。

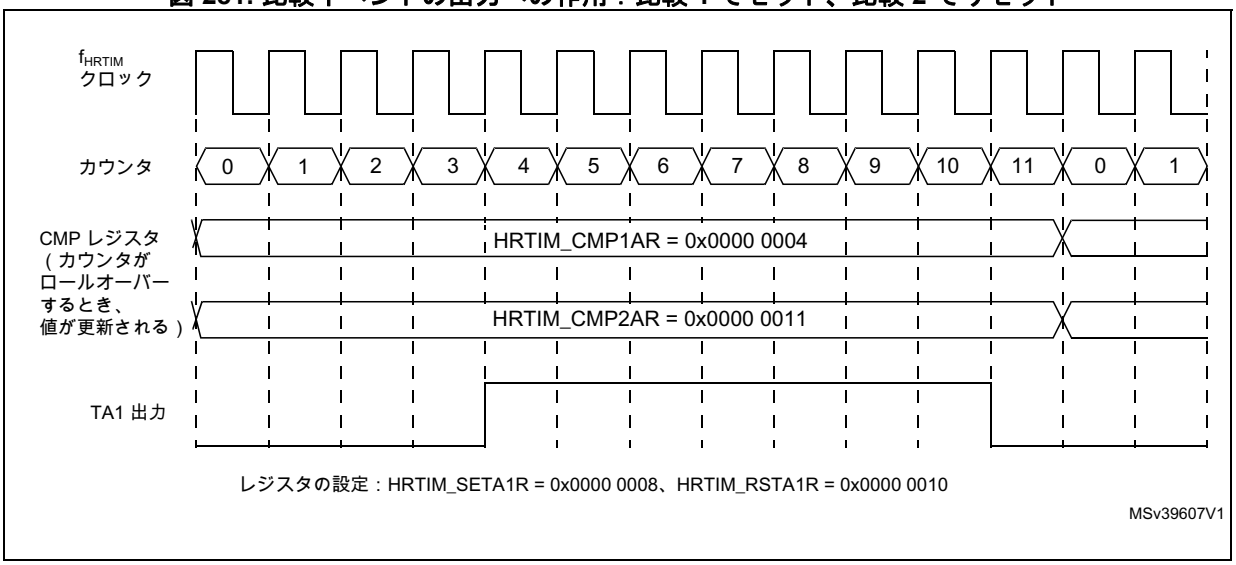
たとえば、タイマ A の出力は次のイベントによってセットまたはリセットできます。タイマ B の比較 1、2、および 4、タイマ C の比較 2 および 3、・・・なお、タイマ E の比較 3 は HRTIM_SETA1R で TIMEVNT8 としてリストされます。

表 286. タイマ A～E にわたるイベントマピング

転送元		タイマ A				タイマ B				タイマ C				タイマ D				タイマ E			
		CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4
転送先	タイマ A	-	-	-	-	1	2	-	3	-	4	5	-	6	7	-	-	-	-	8	9
	タイマ B	1	2	-	3	-	-	-	-	-	-	4	5	-	-	6	7	8	9	-	-
	タイマ C	-	1	2	-	-	3	4	-	-	-	-	-	-	5	-	6	-	7	8	9
	タイマ D	1	-	-	2	-	3	-	4	5	-	6	7	-	-	-	-	8	-	-	9
	タイマ E	-	-	1	2	-	-	3	4	5	6	-	-	7	8	-	9	-	-	-	-

図 281 は 2 つの比較イベントを使用して PWM を生成する方法を表します。

図 281. 比較イベントの出力への作用：比較 1 でセット、比較 2 でリセット



更新イベントによるセット／リセット

ハーフモード

このモードは、50% 固定デューティサイクルで可変周波数の矩形波信号（共振接続形態を使用したコンバータに典型的）の生成を目的にしています。新しい周期がプログラムされるとき、デューティサイクルを自動的に周期値の半分にすることができます。

このモードは、HRTIM_TIMxCR レジスタの HALF ビットに 1 を書き込むことによって有効になります。HRTIM_PERxR レジスタに値が書き込まれると、比較 1 の値が自動的に HRTIM_PERxR/2 の値に更新されます。

矩形波が生成される出力は、以下のように、CMP1 イベントで 1 回の遷移が起き、周期イベントで 1 回の遷移が起きるようにプログラムする必要があります。

– HRTIM_SETxyR = 0x0000 0008、HRTIM_RSTxyR = 0x0000 0004、または
– HRTIM_SETxyR = 0x0000 0004、HRTIM_RSTxyR = 0x0000 0008

ハーフモードでは、HRTIM_CMP1xR レジスタの内容が無効にされます。HRTIM_PERxR レジスタへのアクセスによってのみ、比較 1 の内部レジスタが更新されます。ユーザアクセス可能な HRTIM_CMP1xR レジスタは HRTIM_PERxR / 2 の値で更新はされません。

プリロードが有効になっている場合（PREEN = 1、MUDIS、TxUDIS）、比較 1 アクティブレジスタは更新イベントによりリフレッシュされます。プリロードが無効になっている場合（PREEN = 0）、比較 1 アクティブレジスタは HRTIM_PERxR が書き込まれるとすぐに更新されます。

HALF モードが有効になっているとき、周期は f_{HRTIM} クロックの 6 周期以上である必要があります。

キャプチャ

タイミングユニットには、内部および外部イベントをトリガにしてカウンタ値を取得する機能があります。用途は、次のとおりです。

- イベント到着のタイミングや発生間隔を測定する。
- 自動遅延モードにおいて比較 2 と比較 4 の値を更新する（*自動遅延モード*を参照）。

取得は f_{HRTIM} の分解能で行われます。

タイマには次の 2 つのキャプチャレジスタがあります。HRTIM_CPT1xR および HRTIM_CPT2xR。キャプチャトリガは HRTIM_CPT1xCR および HRTIM_CPT2xCR レジスタでプログラムされます。

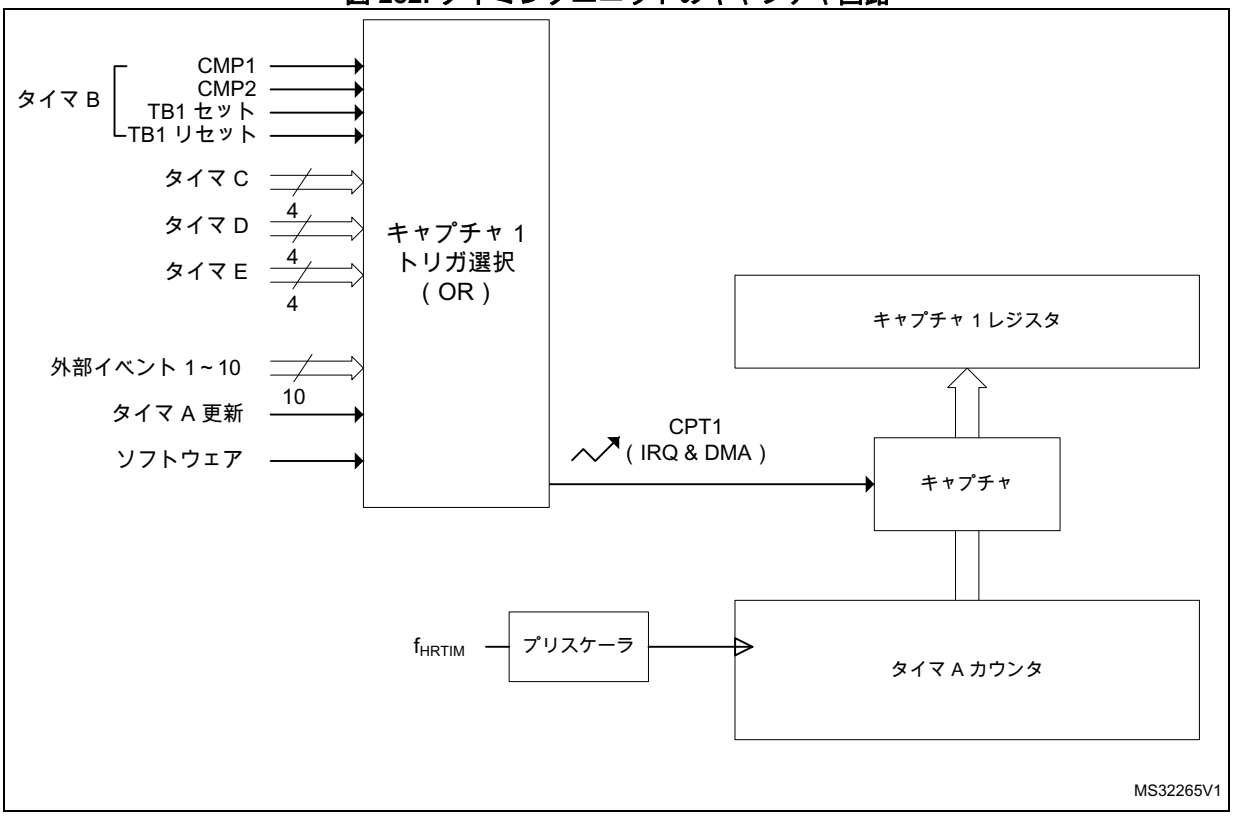
タイミングユニットのカウンタのキャプチャは最大 28 個のイベントによってトリガできます。それらは以下のソースから HRTIM_CPT1xCR および HRTIM_CPT2xCR レジスタで同時に選択できます。

- 外部イベント、EXTEVNT1～10（10 個のイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～E）：比較 1、2 および出力 1 セット／リセットイベント（16 個のイベント）
- タイミングユニット：更新（1 つのイベント）
- ソフトウェアキャプチャ（1 つのイベント）

いくつかのイベントを同時に選択して複数のキャプチャトリガを処理することができます。この場合、同時のトリガリクエストは論理和がとられます。HRTIM_TIMxDIER レジスタに CPTxIE および CPTxDE ビットがセットされているとき、キャプチャによって割り込みまたは DMA リクエストを生成することができます。

オーバーキャプチャは回路では防げません。前の値が読み出されていなくても、あるいはキャプチャフラグがクリアされていなくても、新しいキャプチャはトリガされます。

図 282. タイミングユニットのキャプチャ回路



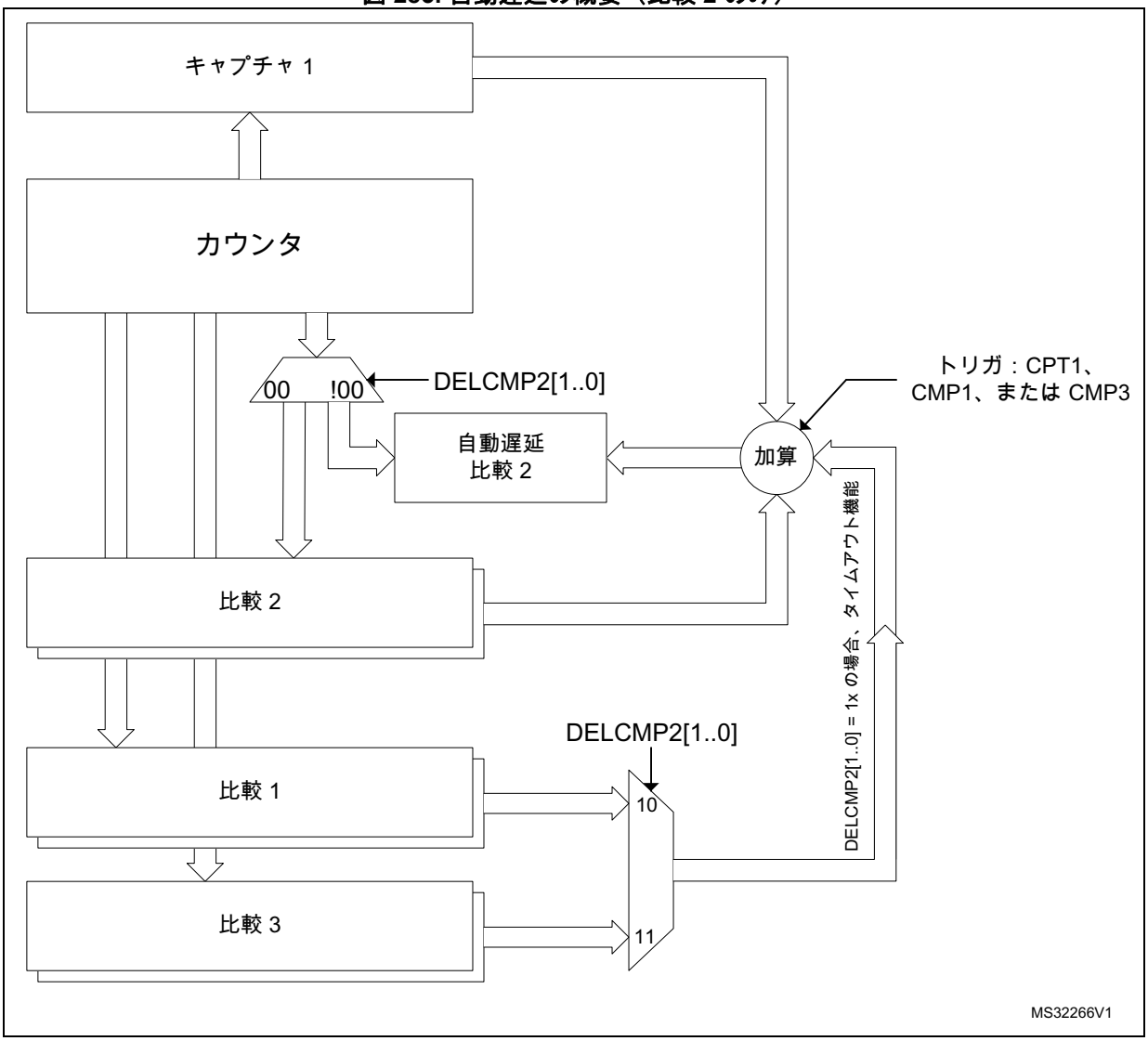
自動遅延モード

このモードではキャプチャイベントに相対的に比較イベントを生成させることができるので、たとえば、キャプチャの後にプログラムされたタイミングで出力を変化させることができます。この場合、比較一致はタイマのカウンタ値とは独立に起きます。ソフトウェアでの計算や割り込み処理を必要とせずに、外部イベントに同期したタイミングで波形を生成することが可能になります。

キャプチャがトリガされない限り、HRTIM_CMPxR レジスタの内容は無視されます（カウンタ値が比較値に一致したとき、比較イベントは生成されません）。キャプチャがトリガされると、HRTIM_CMPxR にプログラムされた比較値が HRTIM_CPTxyR のキャプチャされたカウンタ値と合計され、*図 283* に見られるように、内部自動遅延比較レジスタを更新します。自動遅延比較レジスタはタイミングユニットの内部にあり、読み出しはできません。HRTIM_CMPxR プリロードレジスタは、計算の後、変更されません。

この機能は、比較 2 レジスタと比較 4 レジスタに対してのみ使用できます。比較 2 はキャプチャ 1 に関連しており、比較 4 はキャプチャ 2 に関連しています。通常モードと同様に、HRTIM_CMP2xR および HRTIM_CMP4xR 比較は、f_{HRTIM} クロックの 3 周期を下回る値でプログラムすることはできません。

図 283. 自動遅延の概要（比較 2 のみ）



自動遅延比較は、キャプチャから周期イベントまでの間のみ有効です。カウンタが周期値に達したら、キャプチャが発生するまで、システムは比較を無効にして再び動作可能になります。

以下のように、HRTIM_TIMxCR レジスタの DELCMP2[1:0] および DELCMP4[1:0] ビットによって、自動遅延モードを設定できます。

- 00
  通常比較モード：HRTIM_CMP2xR および HRTIM_CMP4xR レジスタの内容はカウンタ値と直接比較されます。
- 01
  自動遅延モード：比較 2 および比較 4 の値は、キャプチャ 1/2 イベントの後、再計算されてカウンタとの比較に使用されます。

- 1X
  タイムアウト付き自動遅延モード：比較 2 および比較 4 の値は、キャプチャ 1/2 イベントの後、またはキャプチャ 1/2 イベントが失われた場合にタイムアウト機能が働くように、比較 1 が一致（DELCMPx[1:0]= 10）、または比較 3 が一致（DELCMPx[1:0]= 11）した後にも、再計算されてカウンタとの比較に使用されます。

キャプチャが起きると、比較は （HRTIM_CMP2/4xR + HRTIM_CPT1/2xR）の値で行われます。周期内でキャプチャがトリガされない場合、動作は以下のように DELCMPx[1:0] の値に依存します。

- DELCMPx[1:0] = 01：比較イベントは生成されません。
- DELCMPx[1:0] = 10 または 11 : 2 つの比較の合計（たとえば、HRTIM_CMP2xR + HRTIM_CMP1xR）で比較が行われます。キャプチャが CMPx + CMP1（または CMPx + CMP3）の後にトリガされた場合、そのキャプチャは考慮されません。

キャプチャは、次の PWM 周期の始めに再び有効化されます。

自動遅延の合計の結果が 0xFFFF を超えた場合（オーバーフロー）、その値は無視され、新しい周期が始まるまで比較イベントは生成されません。

注： ***DELCMPx[1:0] ビットフィールドをある値から別の値に再プログラムするときは、自動遅延メカニズムを適切に再初期化するために、そのビットフィールドをリセットする必要があります。たとえば以下のとおりです。***

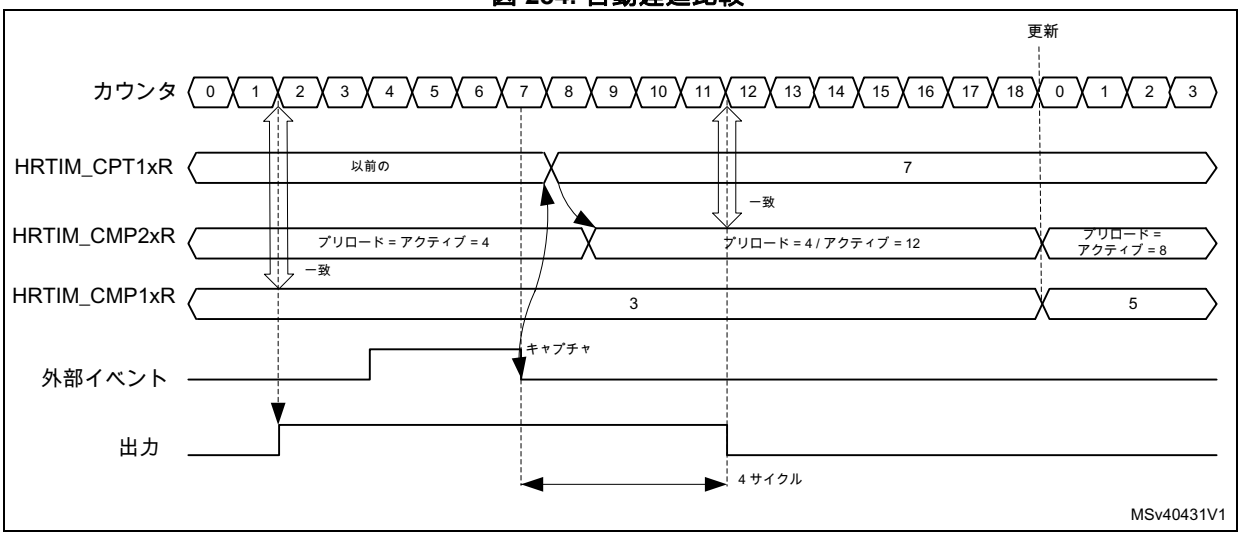
- DELCMPx[1:0] = 10
- DELCMPx[1:0] = 00
- DELCMPx[1:0] = 11

例として、次の信号がどのようにして生成されるかを*図 284* に示します。

- カウンタが比較 1 の値に等しい時に出力をセット
- 与えられた外部イベントの立ち下がりエッジの 4 サイクル後に出力をリセット

注： ***図を単純化するために、外部イベント信号は再同期遅延なしで表示されています。実際は、外部入力信号を処理するのに必要な内部再同期ステージのため、立ち下がりエッジとキャプチャイベントの間に f_{HRTIM} クロックの 1～2 周期の遅れがあります。***

図 284. 自動遅延比較



通常の比較チャネル（たとえば、比較 1）は出力のセットに使用されます。カウンタが比較レジスタの内容と一致すると、すぐに出力はアクティブ状態に移行します。

遅延比較は出力のリセットに使用されます。キャプチャイベントが発生した場合のみ、比較イベントが生成されます。カウンタが遅延比較値に一致するとき（カウンタ = 4）、イベントは生成されません。キャプチャイベントが外部イベントによってトリガされると、キャプチャレジスタの内容が遅延比較値に合計され、新しい比較値になります。例では、自動遅延値 4 が、7 に等しいキャプチャ値に合計され、自動遅延比較レジスタに 12 の値を与えます。これ以降、比較イベントは生成可能となり、カウンタが 12 に等しくなる時に生成され、出力をリセットします。

自動遅延モードにおけるオーバーキャプチャ管理

自動遅延モードが有効になっているとき（DELCMPx[1:0] = 01、10、11）、オーバーキャプチャは防止されます。

複数のキャプチャリクエストが同じカウント周期内で発生したとき、最初のキャプチャのみが考慮されて、自動遅延比較値が計算されます。新しいキャプチャは以下のときのみ、可能となります。

- 自動遅延比較がカウンタ値と一致した（比較イベント）後
- カウンタがロールオーバーした（周期）後
- タイマがリセットされた後

自動遅延比較値の変更

自動遅延比較値がプリロードされると（PREEN ビットのセット）、新しい比較値は、次に来る更新イベント時（たとえば、周期イベント時）に考慮されます。これは、比較レジスタがいつ書き込まれたか、およびキャプチャが発生したか否かに関わりません（*図 284* を参照ください。そこでは、カウンタがロールオーバーする時に、遅延が変更されています）。

プリロードが無効になっているときは（PREEN ビットのリセット）、たとえキャプチャイベントが発生した後に変更されても、新しい比較値は直ちに考慮されます。その例を以下に示します。

1. t1 で、DELCMP2 = 1
2. t2 で、CMP2_act = 0x40 => 比較は無効化
3. t3 で、キャプチャイベントが発生し、CPTR1 = 0x20 の値を取得=> 比較は有効化、比較値 = 0x60
4. t4 で、CMP2_act = 0x100（カウンタが CPTR1 + 0x40 の値に達する前に）=> 比較はまだ有効、新しい比較値 = 0x120
5. t5 で、カウンタが周期値に達する => 比較は無効化、cmp2_act = 0x100

同様に、DELCMPx = 10 または 11 で、プリロードが無効のとき、CMP1（CMP3）の値が変化した場合は、以下のとおり。

1. t1 で、DELCMP2 = 2
2. t2 で、CMP2_act = 0x40 => 比較は無効化
3. t3 で、キャプチャ 1 イベントが発生する前に、CMP3 イベントが発生 - CMP3_act = 0x50 => 比較は有効化、比較値 = 0x90
4. t4 で、CMP3_act = 0x100（カウンタが 0x90 の値に達する前に）=> 比較はまだ有効、比較 2 のイベントが = 0x140 で発生

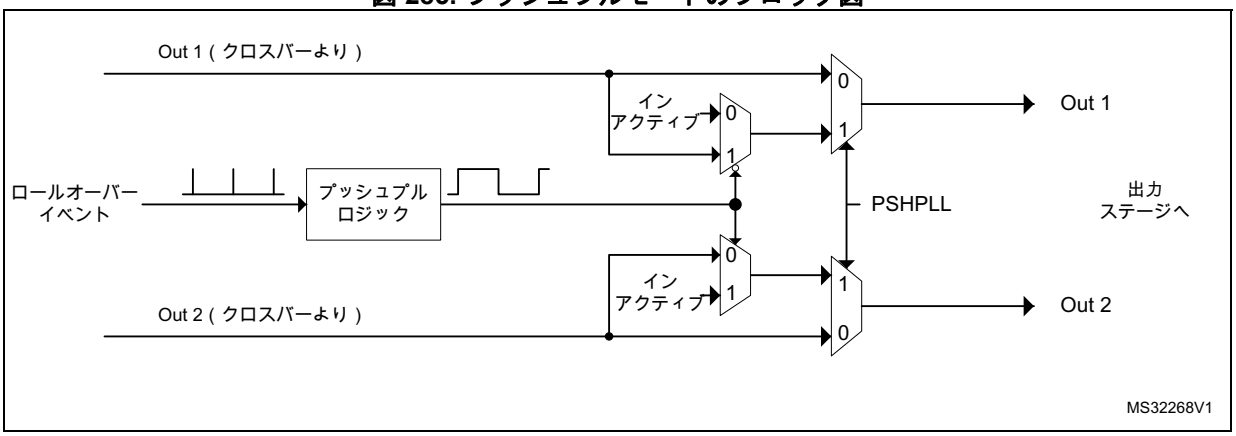
プッシュプルモード

このモードは主にプッシュプル接続形態を使用してコンバータを駆動することを目的にしています。共振コンバータが典型的ですが、遅延アイドル保護が必要なとき、これを有効にする必要があります（*セクション 37.3.9：遅延保護* を参照）。

プッシュプルモードは、HRTIM_TIMxCR レジスタの PSHPLL ビットをセットすることによって有効になります。

クロスバーで生成された信号を、周期ベースで、出力 1 と出力 2 に交互に適用し、適用してない他方の出力は非アクティブに保ちます。リダイレクトレート（プッシュプル周波数）は、*図 285* に示すように、タイマの周期イベントによって定義されます。プッシュプル周期はタイマのカウント周期の 2 倍です。

図 285. プッシュプルモードのブロック図



プッシュプルモードは、タイマが連続モードで動作しているときにのみ、利用できます。カウンタをいったん有効にすると（TxCEN ビットをセット）、リセットしてはなりません。プッシュプル動作を停止するためには、タイマを無効にする必要があり、それを再び有効にする前にはカウンタをリセットする必要があります。

信号波形は、両出力に対して HRTIM_SETxyR および HRTIM_RSTxyR を使用して定義します。HRTIM_SETx1R = HRTIM_SETx2R および HRTIM_RSTx1R = HRTIM_RSTx2R として、両出力を同じ波形とし、バランスのとれた動作にする必要があります。それでも、他の用途では、両出力に異なるプログラミングをすることができます。

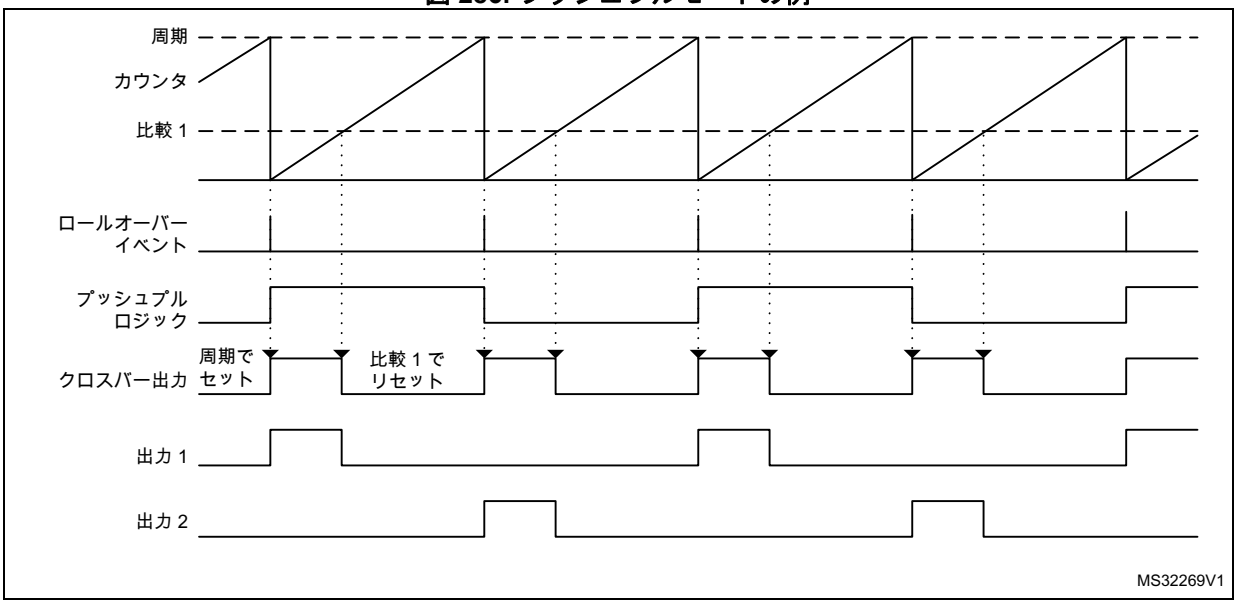
注： ***デッドタイムを有効にするときは、プッシュプル動作を使用することはできません（相互に排他的な機能です）。***

HRTIM_TIMxISR の CPPSAT ステータスビットは、信号が現在、どちらの出力にアクティブなのかを示します。プッシュプルモードが無効になると、CPPSTAT はリセットされます。

図 286 の例では、タイマの内部波形は次のように定義されています。

- 周期イベントで出力セット
- 比較 1 の一致イベントで出力リセット

図 286. プッシュプルモードの例



デッドタイム

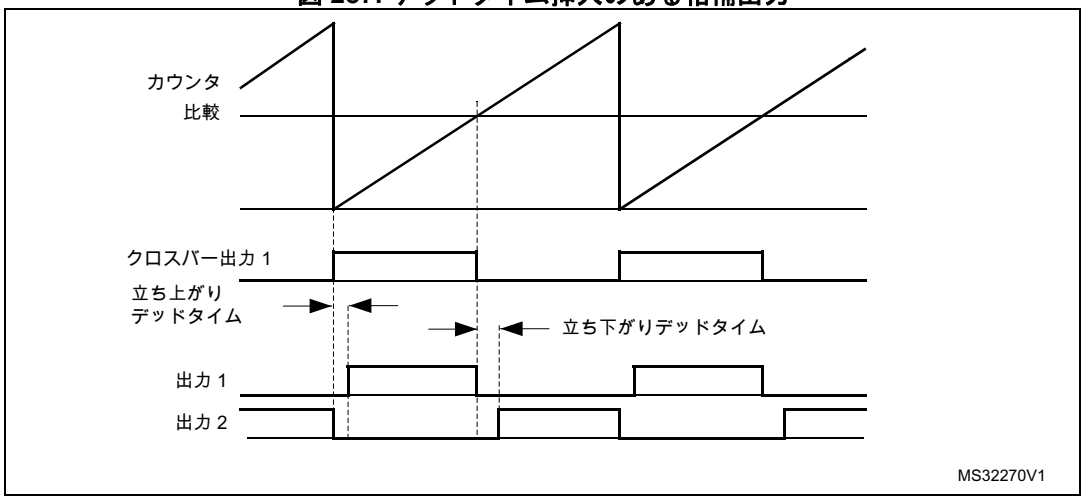
デッドタイム挿入ユニットによって、単一の基準波形から、アクティブ状態の遷移間にプログラム可能な遅れを設定した 1 対の相補信号を生成することができます。これは一般にハーフブリッジまたはフルブリッジを使用する接続形態に使用されます。これによってソフトウェアが簡単になります。2 つの出力を駆動するのに、1 つの波形のみをプログラムして制御するだけで済みます。

デッドタイム挿入を有効にするには、HRTIM_OUTxR レジスタの DTEN ビットをセットします。HRTIM_SETx1R および HRTIM_RSTx1R レジスタを使用して、出力 1 に対して定義された基準波形に基づいて相補信号が作成されます。DTEN ビットをセットしているとき、HRTIM_SETx2R および HRTIM_RSTx2R レジスタは意味がありません。

注： ***デッドタイムはプッシュプルモードと同時に使用することはできません。***

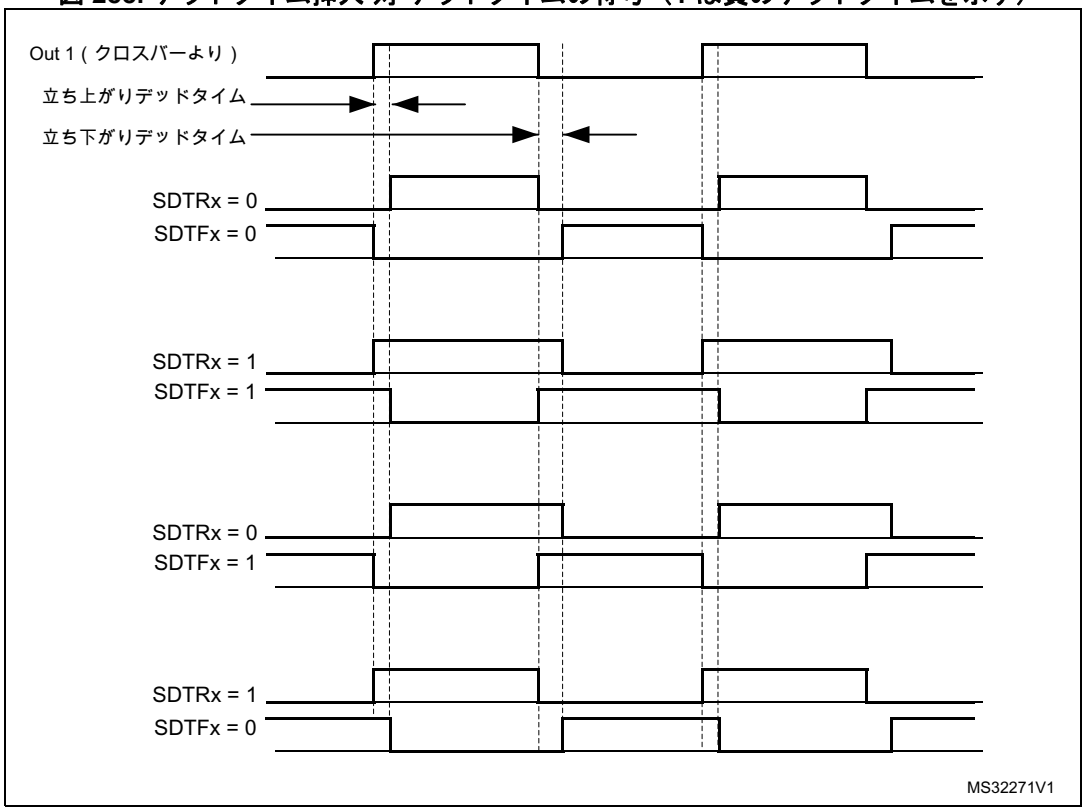
図 287 **のように、基準波形の立ち上がりエッジと立ち下がりエッジに関連して 2 つのデッドタイムを定義できます。**

図 287. デッドタイム挿入のある相補出力



制御のオーバーラップが必要なとき、負のデッドタイム値を定義できます。これは、デッドタイムの符号ビット（HRTIM_DTxR レジスタの SDTFx および SDTRx ビット）を使用することでできます。*図 288* に、各々の符号に応じた相補信号波形を示します。

図 288. デッドタイム挿入 対 デッドタイムの符号（1 は負のデッドタイムを示す）



デッドタイムの値は、以下のとおり、DTFx[8:0] および DTRx[8:0] ビットフィールドによって、および DTPRSC[2:0] ビットに従ってプリスケールされた特定のクロックに基づいて定義されます。

t_{DTx} = +/- DTx[8:0] x t_{DTG}

ここで、x は R または F、t_{DTG} = ($2^{(DTPRSC[2:0])}$) x t_{HRTIM}。

表 287 は、プリスケーラの値に応じた分解能と最大の絶対値を示します。

表 287. デッドタイム分解能と最大の絶対値

DTPRSC[2:0](1)	t_{DTG}	t_{DTx} max	f_{HRTIM}= 400 MHz	
			t_{DTG}（ns）	$\|t_{DTx}\|$ max（µs）
011	t_{HRTIM}	511 * t_{DTG}	2.5	1.28
100	2 * t_{HRTIM}		5	2.56
101	4 * t_{HRTIM}		10	5.11
110	8 * t_{HRTIM}		20	10.22
111	16 * t_{HRTIM}		40	20.44

1. DTPRSC[2:0] の値 000、001、010 は予約済み。

図 289～*図 292* に、すべてのデッドタイムの設定について、パルス幅がデッドタイムの値より小さい基準波形に対してデッドタイムジェネレータがどのように動作するかを示します。

図 289. 小パルス幅に対する相補出力（SDTRx = SDTFx = 0）

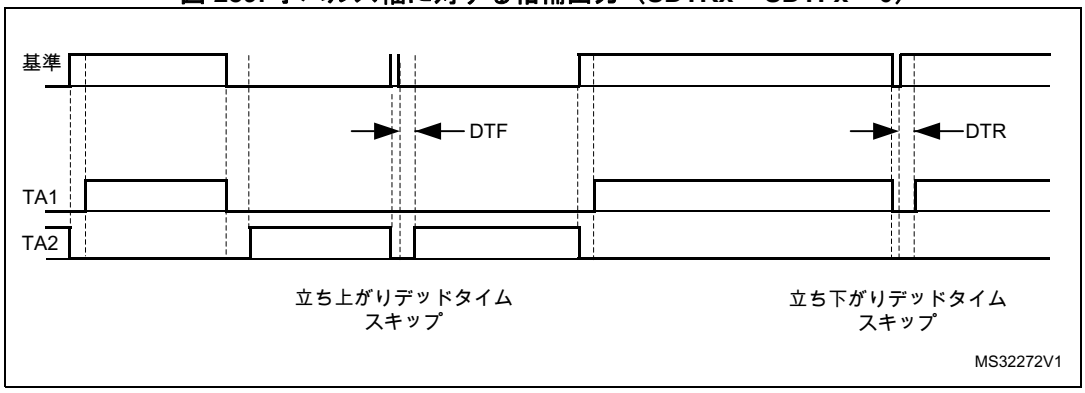


図 290. 小パルス幅に対する相補出力（SDTRx = SDTFx = 1）

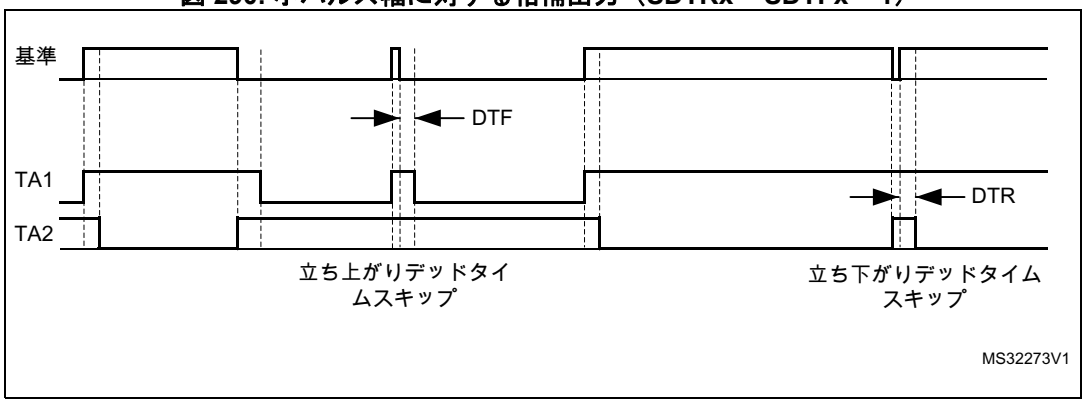


図 291. 小パルス幅に対する相補出力（SDTRx = 0、SDTFx = 1）

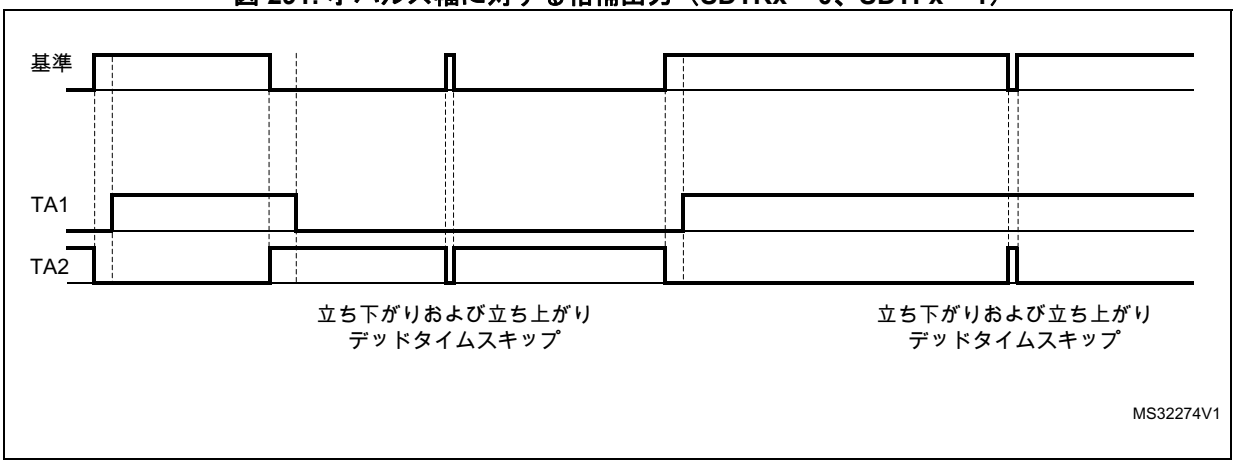


図 292. 小パルス幅に対する相補出力（SDTRx = 1、SDTFx = 0）



安全のため、DTFLKx、DTRLKx、DTFSLKx および DTRSLKx を使用して、デッドタイムの符号および／または値をロックすることによって、デッドタイムレジスタへの誤った書き込みを防ぐことができます。これらのビットがセットされると、次のシステムリセットまで関連ビットおよびビットフィールドが読み出し専用になります。

注意： ***以下の場合、DTEN ビットは変更してはなりません。***
- タイマが有効になっているとき（TxEN ビットをセット）
- タイマ出力が別のタイマによってセット／リセットされているとき（TxEN はリセットされている）
そうしないと、予測できない動作になることがあります。
したがって、タイマを無効化して（TxCEN ビットをリセット）、対応する出力を無効化する必要があります。

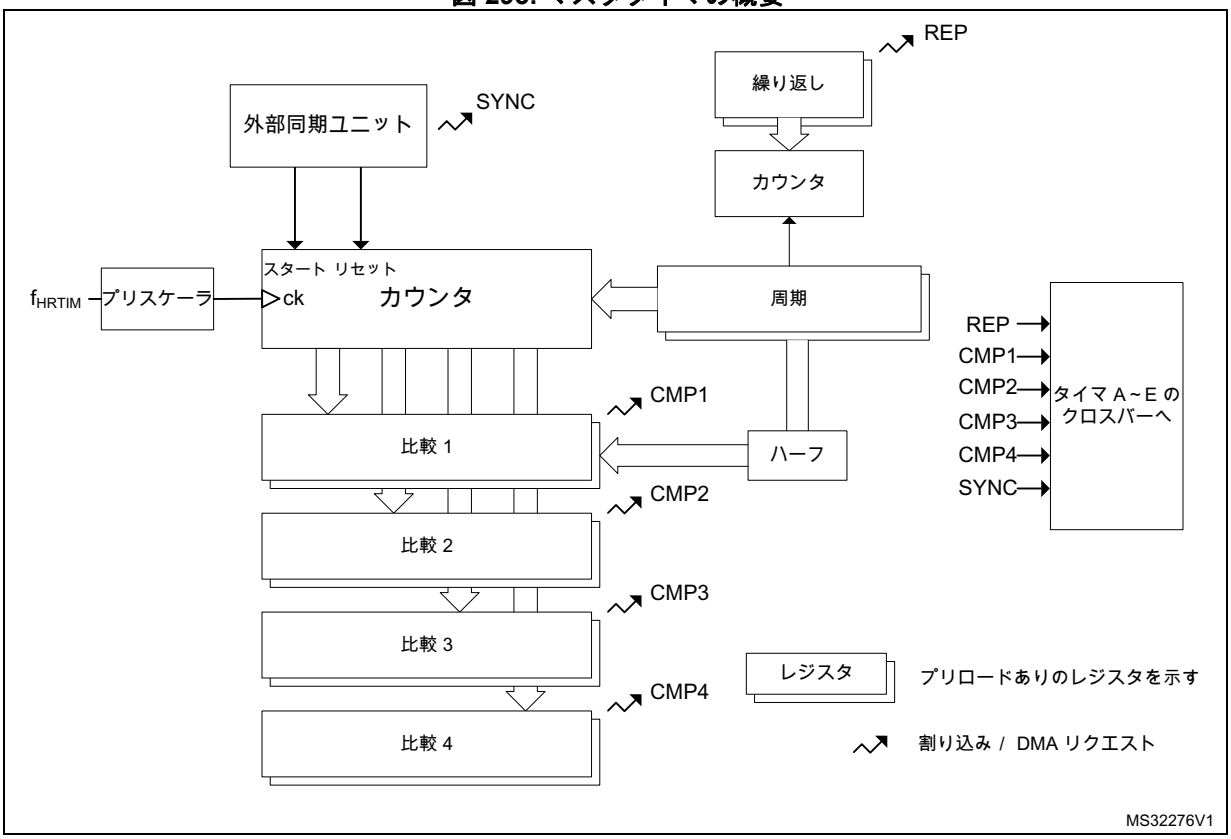
バーストモードが有効になっているときに、モード移行時にデッドタイムと共に DTEN をセットしなければならない特定の場合には（BME = 1、DIDL = 1、IDLEM = 1）、DTEN ビットをセットする前にソフトウェアコマンドで 2 つの出力を IDLES 状態に強制する（SST、RST ビット）必要があります。これは、バーストモードへの移行がデッドタイムが有効になる前に直ちに起きることから生じる副作用を避けるためです。

37.3.5 マスタタイマ

マスタタイマの主な目的は、同期目的のために、あるいは出力をセット／リセットするために、5 つのタイミングユニットに共通の信号を提供することです。マスタタイマは出力を直接制御することはしませんが、セット／リセットクロスバーによって間接的に行うことができます。

図 293 にマスタタイマの概要を示します。

図 293. マスタタイマの概要



マスタタイマはタイミングユニットとまったく同じアーキテクチャに基づいていますが、下記のような違いがあります。

- 関連する出力を持たず、出力に関連するコントロールも持ちません。
- 自身のクロスバーユニットも、プッシュプルまたはデッドタイムモードもありません。
- 外部の同期回路によってのみ、リセットできます。
- キャプチャユニットも、自動遅延モードもありません。
- 外部イベントのブランキングおよびウィンドウイング回路は含まれません。
- 次の限定された割り込み／DMA リクエストしかありません。比較 1～4、繰り返し、レジスタ更新、および外部同期イベント。

マスタタイマ制御レジスタには、マスタおよびタイマ A～E のタイミングユニット用のすべてのタイマイネーブルビットが含まれています。これによって、1 回の書き込みアクセスで、すべてのタイマを同期して開始することができます。

また、MCU の内部および外部（入出力）リソースの両方によって HRTIM タイマ全体の外部同期も行います（*セクション 37.3.17：他のタイマまたは HRTIM インスタンスとの HRTIM の同期*を参照）。

マスタタイマ制御レジスタは、タイミングユニットのレジスタと同じオフセットで配置されています。

37.3.6 セット／リセットイベントの優先順位と狭いパルスの管理

このセクションでは、複数のセットおよび／またはリセットリクエストが連続する t_{HRTIM} の 3 周期内に発生した場合に出力波形がどのように生成されるかを説明します。

各 t_{HRTIM} 周期内で以下の 2 ステップで調整が行われます。

1. 各アクティブイベントに対して、望まれる出力遷移を判断します（セット、リセット、または反転）。
2. アクティブイベントの中で、事前に定義された調整が行われます（最高から最低の優先順位に、CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1 ⟶ PER。*同時のセットリクエスト／同時のリセットリクエスト*を参照）。

2 つの異なるソースからセットとリセットのリクエストが同時に来たとき、リセット動作が最高の優先順位になります。

同時のセットリクエスト／同時のリセットリクエスト

セットイベントに複数のソースが選択された場合、同じ f_{HRTIM} クロック周期内で複数のリクエストが発生したとき、調整が行われます。

隣接するタイマからの複数のリクエスト（TIMEVNT1～9）の場合、最初に発生したリクエストが考慮されます。ソース（最高から最低の優先順位）によって調整が 2 ステップで行われます。すなわち、CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1。

マスタタイマから複数のリクエストが同じ f_{HRTIM} クロック周期内に発生した場合、事前に定義された調整が適用され、単一のリクエストが考慮されます（最高から最低の優先順位で）。

すなわち、MSTCMP4 → MSTCMP3 → MSTCMP2 → MSTCMP1 → MSTCMPER。

内部からタイマへの複数のリクエストが同じ f_{HRTIM} クロック周期内に発生した場合、事前に定義された調整が適用され、リクエストは、有効なタイミングであれば何であれ、次の優先順位で取り込まれます（最高から最低の順で）。

すなわち、CMP4 → CMP3 → CMP2 → CMP1 → PER。

注： ***実際上、自動遅延比較 2 と比較 4 を同時に使用するとき（すなわち、外部イベントに関係しているので、有効なセット／リセットを事前に決められないとき）のみ、これは最も重要です。この場合、最も高い優先順位の信号を CMP4 イベントに影響させる必要があります。***

最後に、最も高い優先順位は、タイミングに関連しない次のイベントです。EXTEVNT1～10、RESYNC（SYNCRSTx または SYNCSTRTx がセットされている場合の SYNC イベントから、またはソフトウェアリセットから来るもの）、更新、およびソフトウェアセット（SST）。

まとめると、同時イベントの場合、有効なセット（リセット）イベントは以下の間で調整されます。

- 任意の TIMEVNT1～9 イベント
- マスタからの単一のソース（上記で与えられた固定の調整による）
- タイマからの単一のソース
- 「タイミングに関連しないイベント」

同時のリセットリクエストにも同じ調整原則が適用されます。この場合、リセットリクエストは最も高い優先順位になります。

プリスケーラクロックサイクル内で起きるセットまたはリセットイベントは、たとえ調整が t_{HRTIM} サイクルごとに行われていても、プリスケーラクロックの次のアクティブエッジまで遅延されます（カウンタリセットに関して）。

同じプリスケーラクロックサイクル内で、リセットイベントの後にセットイベントが発生した場合、最後のイベントが考慮されます。

37.3.7 外部イベントのグローバル調整

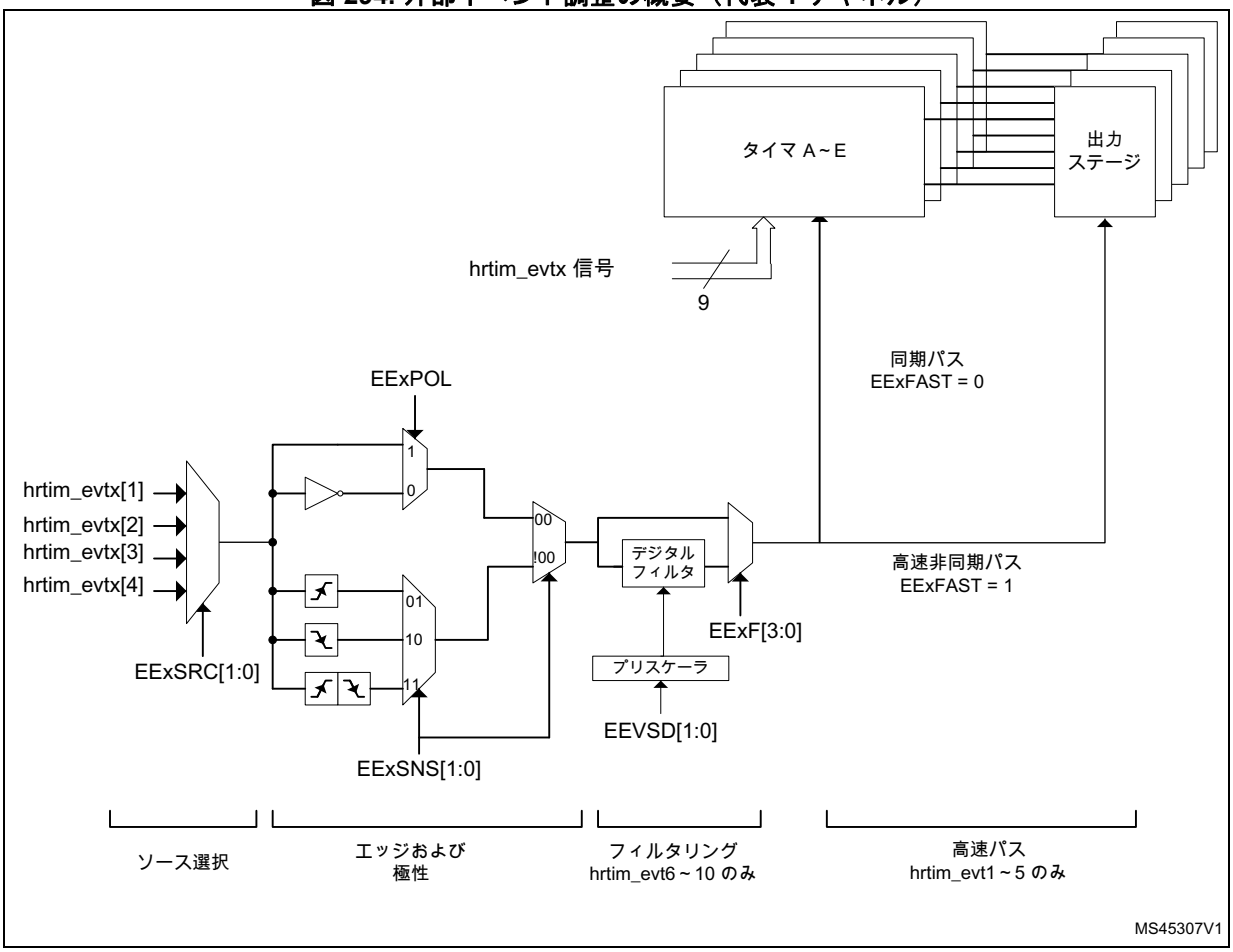
HRTIM タイマは、タイマ内で生成されないイベント（「外部イベント」と呼びます）を処理できます。これらの外部イベントは、以下のオンチップやオフチップの複数のソースから来ます。

- 内蔵コンパレータ、
- デジタル入力ピン（典型として、オフチップコンパレータおよびゼロ交差検出回路に接続されたピン）、
- 他のペリフェラル用のオンチップイベント（ADC のアナログウォッチドッグおよび汎用タイマトリガ出力）。

外部イベント調整回路によって、(4:1 マルチプレクスサ付きの）所定チャネルに対して信号ソースを選択すること、そして、それをクロスバーユニットで処理できる情報に変換すること（たとえば、外部イベントチャネルの立ち下がりエッジ検出をトリガとして出力をリセットさせるなど）ができます。

最大 10 個の外部イベントチャネルを調整でき、5 つのどのタイマにも同時に使用できます。この調整は、通常外部コンポーネント（ゼロ交差検出回路など）および環境条件によって要求されるので（一般にフィルタのセットアップはアプリケーションのノイズレベルや特性に関連します）、すべてのタイマに共通です。*図 294* に、単一のチャネルに対する調整ロジックの概要を示します。

図 294. 外部イベント調整の概要（代表 1 チャネル）



HRTIM_EECR1 および HRTIM EECR2 レジスタを使用して 10 個の外部イベントが以下のように初期化されます。

- EExSRC[1:0] ビットによって最大 4 つのソースを選択します。
- EExSNS[1:0] ビットによって感度を選択し、レベル対応にするか、またはエッジ対応（立ち上がりか、立ち下がりか、両方）にします。
- レベル対応の場合、EExPOL ビットによって極性を選択します。
- 外部イベント 1～5 に対して、EExFAST ビット（*外部イベントへの遅延時間*を参照）によって低遅延時間モードにします。

注： ***リセット、キャプチャ、バーストモード、ADC トリガ、および遅延保護のトリガとして使用される外部イベントは、EESNS ビットがリセット（レベル対応の選択）されていても、エッジ対応となります。POL = 0 の場合、外部イベントの立ち上がりエッジでトリガがアクティブになり、POL = 1 の場合、外部イベントの立ち下がりエッジでトリガがアクティブになります。***

出力状態の変化およびカウンタのリセットを防ぐためにカウンタが無効にされている限り（TxCEN ビットのリセット）、ADC トリガとして使用される場合を除き、外部イベントは破棄されます。

さらに、HRTIM_EECR3 レジスタの EExF[3:0] ビットを使用して、外部イベント 6～10 に対してデジタルノイズフィルタを有効にすることが可能です。

デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。カウンタが N の値に達する前に入力値が変化した場合、カウンタはリセットされ、変化は破棄されます（誤ったイベントとみなされます）。カウンタが N に達した場合、変化は有効とみなされ、正しい外部イベントとして転送されます。したがって、デジタルフィルタは、サンプリングクロックとフィルタ長（予想される有効なサンプルの数）に応じて、フィルタリングされる外部イベントに遅延時間を付加します。

サンプリングクロックは f_{HRTIM} クロックか、または f_{HRTIM} から導き出される特定のプリスケールされたクロック f_{EEVS}（HRTIM_EECR3 レジスタの EEVSD[1:0] ビットで定義されます）になります。

表 288 に、10 個の外部イベントチャネルの各々に関連して利用可能なソースおよび機能をまとめます。

表 288. 外部イベントのマッピングおよび関連機能

外部イベントチャネル	高速モード	デジタルフィルタ	バランスフォールトタイマ A、B、C	バランスフォールトタイマ D、E	Src1	Src 2	Src3	Src4
1	はい	-	-	-	PC10	COMP1	TIM1_TRGO	ADC1_AWD1
2	はい	-	-	-	PC12	COMP2	TIM2_TRGO	ADC1_AWD2
3	はい	-	-	-	PD5	-	TIM3_TRGO	ADC1_AWD3
4	はい	-	-	-	PG11	OPAMP1(1)	TIM7_TRGO	ADC2_AWD1
5	はい	-	-	-	PG12	-	LPTIM1 OUT	ADC2_AWD2
6	-	はい	はい	-	PB4	COMP1	TIM6_TRGO	ADC2_AWD3
7	-	はい	はい	-	PB5	COMP2	TIM7_TRGO	-
8	-	はい	-	はい	PB6	-	TIM6_TRGO	TTCAN_TMP
9	-	はい	-	はい	PB7	OPAMP1 (1)	TIM15_TRGO	TTCAN_RTP
10	-	はい	-	-	PG13	-	LPTIM2 OUT	TTCAN_SOC

1. OPAMP1_OUT は、高分解能タイマの内部イベントソースとして使用できます。この場合、ソフトウェアで OPAMP1_DIG を PA6 オルタネート機能（AF13）として設定し、OPAMP1_VOUT 信号を HRTIM 外部イベントにシュミットトリガ経由でリダイレクトする必要があります。

外部イベントへの遅延時間

外部イベント調整によって、性能の期待に応じて外部イベント処理時間（および関連する遅延時間）を以下のように調節することができます。

- 出力クロスバーに作用する前に外部イベントがクロックで再サンプリングされる、通常の動作モード。これによって、いくらかの遅延時間が付加されますが、すべてのクロスバー機能にアクセスできます。外部からトリガされる高分解能パルスを生成できます。
- 外部イベントと出力への作用の間の遅延時間が最小になる高速動作モード。このモードは、たとえば、超高速の過電流保護に便利です。

HRTIM_EECR1 レジスタの EExFAST ビットによって、チャネル 1～5 に対する動作を定義することができます。これによって、下記表に要約されるように、出力パルス上の遅延時間とジッタが影響を受けます。

表 289. 出力のセット／リセット遅延時間とジッタ 対 外部イベント動作モード

EExFAST	応答時間の遅延	応答時間のジッタ	出力パルス上のジッタ （外部イベントによるカウンタリセット）
0	f_{HRTIM} クロックの 5～6 サイクル	f_{HRTIM} クロックの 1 サイクル	ジッタなし、パルス幅は高分解能で維持
1	最小遅延時間 （コンパレータが 使用されるかデジタル入力が 使用されるかによる）	最小ジッタ	f_{HRTIM} クロックの 1 サイクルのジッタ、 パルス幅分解能は t_{HRTIM} より下がる

EExFAST はレベル対応のプログラミング（EExSNS[1:0] = 00）でのみ使用可能であり、エッジ対応はプログラムできません。

外部イベントにイベントフィルタを適用できます（EExFLTR[3:0] != 0000 によりブランキングとウィンドウイングの両方。*セクション 37.3.8* 参照）。この場合、EExLTCHx ビットはリセットする必要があります。延期モードはサポートされておらず、ウィンドウイングのタイムアウト機能も同じです。

注： ***関連する EExFAST ビットをセットすると、外部イベントの設定（ソースおよび極性）は変更してはなりません。***

高速外部イベントは出力のトグルには使用できません。HRTIM_SETxyR または HRTIM_RSTxyR のどちらかで（両方ではなく）有効にする必要があります。

独立の 2 つの高速外部イベントからセットとリセットのイベントが同時に発生した場合は、クロスバーではリセットが最大の優先順位を持ち、出力は非アクティブになります。

EExFAST ビットがセットされているとき、外部イベントの後、f_{HRTIM} の 11 クロック周期の間、出力は変更できません。

図 295 と *図 296* に、出力セット／リセットおよびカウンタリセットの外部イベントへの反応時間の実用例を示します。

参考資料

図 295. 外部イベントの立ち下がりエッジの遅延時間（カウンタリセットおよび出力セット）

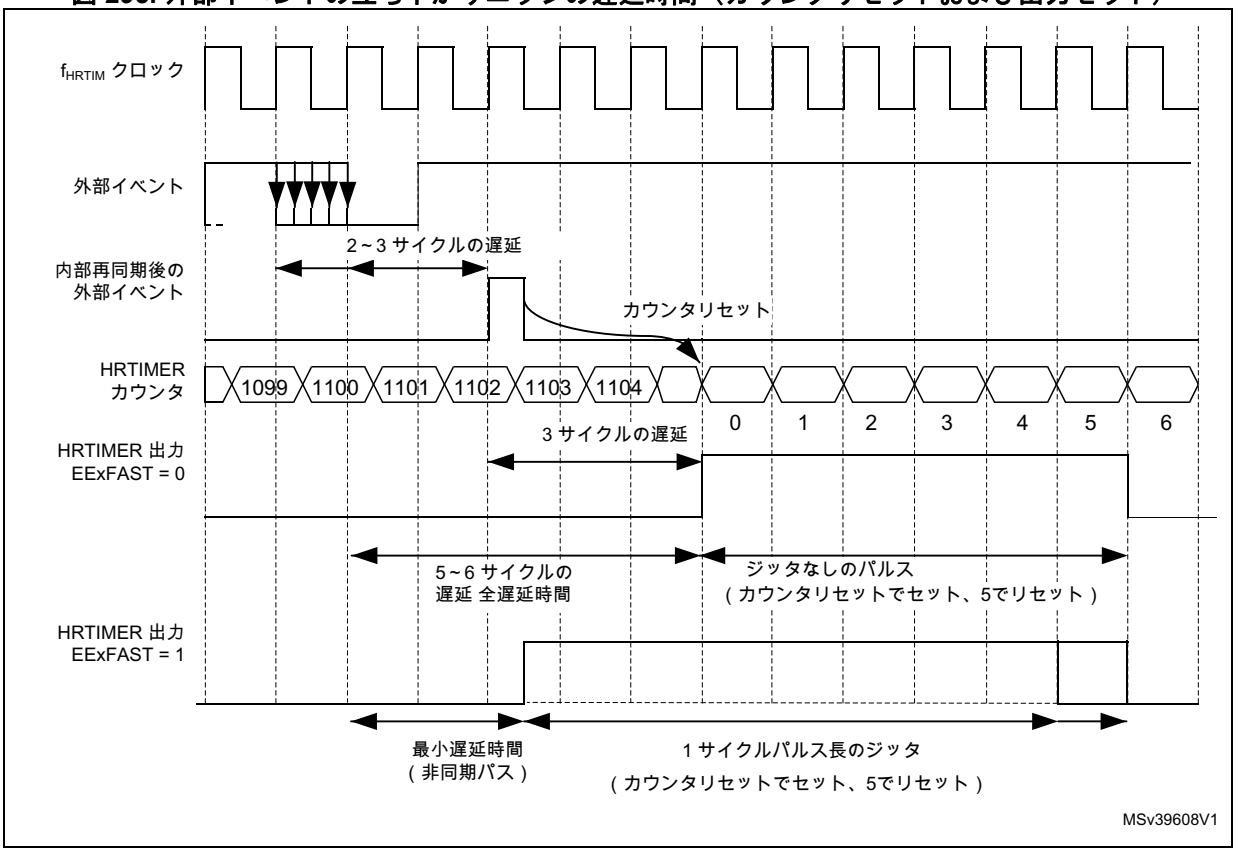
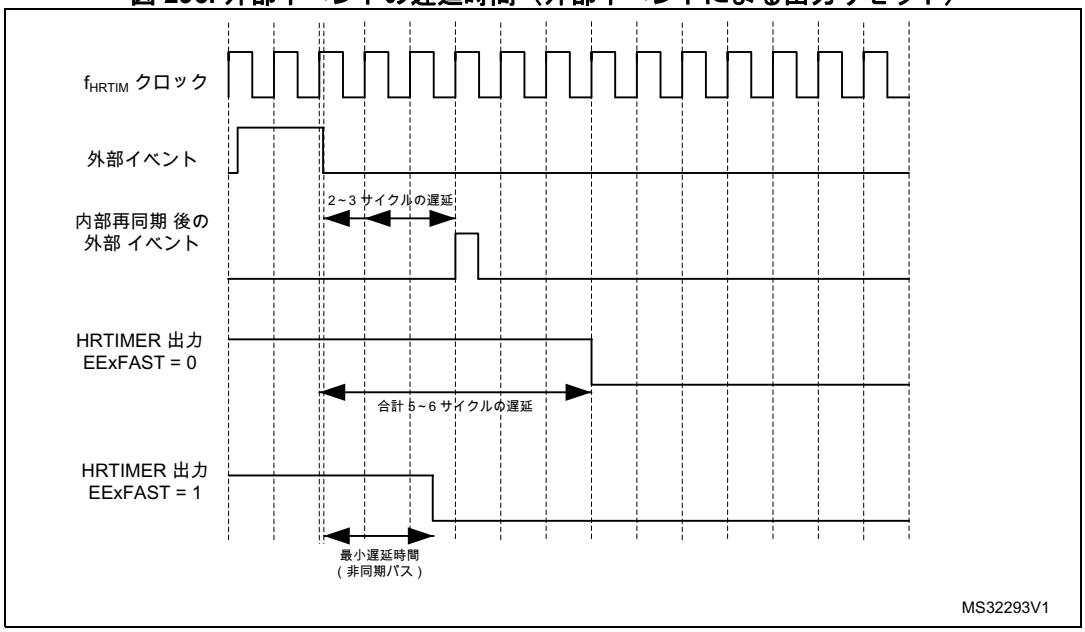


図 296. 外部イベントの遅延時間（外部イベントによる出力リセット）



37.3.8 タイミングユニットでの外部イベントのフィルタリング

調整が終わると、すべてのタイミングユニットで 10 個の外部イベントが利用可能になります。

これらは直接使用可能で、タイミングユニットのカウンタが有効になると（TxCEN ビットのセット）すぐにアクティブになります。

また、通常、カウント周期に関連して、動作に時間的制限を与えるようにフィルタリングすることもできます。次の 2 つの動作を行わせることができます。

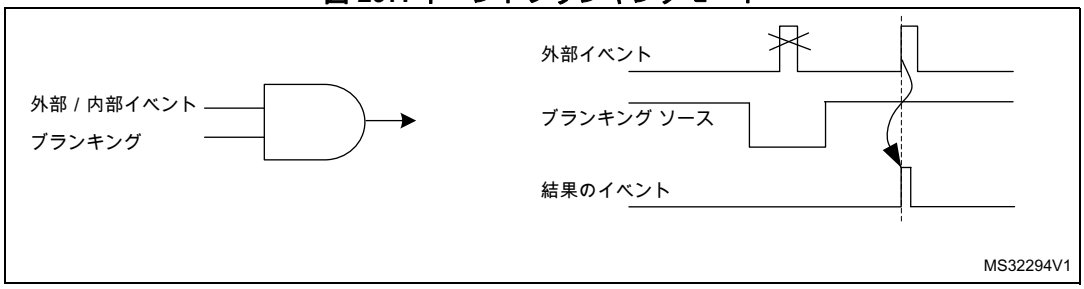
- ブランキング。定義された時間の間、外部イベントをマスクします。
- ウィンドウイング。定義された時間の間のみ、外部イベントを有効にします。

これらのモードは、HRTIM_EEFxR1 および HRTIM_EEFxR2 レジスタの HRTIM_EExFLTR[3:0] ビットを使って有効化されます。5 つのタイマ A～E のタイミングユニットの各々には、10 個の外部イベントに対して独自のプログラム可能なフィルタ設定が用意されています。

ブランキングモード

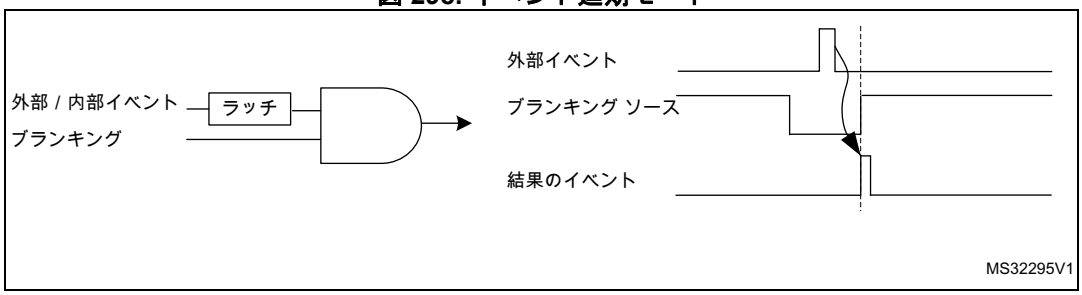
イベントブランキングモードでは（*図 297* 参照）、決められたブランキング期間中に発生した外部イベントは無視されます。これは、たとえば、PWM 周期の始めのスイッチングノイズによるトリップの電流限度を避けるのに便利です。このモードは、0001～1100 の範囲の EExFLTR[3:0] ビットフィールドの値に対してアクティブになります。

図 297. イベントブランキングモード



イベント延期モードでは、*図 298* に示すように、外部イベントはすぐには考慮されませんが、記憶され（ラッチされ）て、ブランキング期間が終わると直ちに生成されます。このモードを有効にするには、HRTIM_EEFxR1 および HRTIM_EEFxR2 レジスタの EExLTCH ビットをセットします。

図 298. イベント延期モード



ブランキング信号は以下のいくつかのソースから受信します。

- タイマ自身：ブランキングはカウンタリセットから比較一致まで続きます（比較 1 ～比較 4 に対して、EExFLTR[3:0] = 0001～0100）。
- 他のタイミングユニットから（EExFLTR[3:0] = 0101～1100）：ブランキングは、選択したタイミングユニットのカウンタリセットから比較の 1 つの一致まで続き、あるいは Tx2 出力の波形として完全にプログラム可能です。この場合、Tx2 信号が非アクティブである限り、イベントはマスクされます（出力を有効にする必要はなく、信号は出力ステージの前で取られます）。

EEXFLTR[3:0] の 0101～1100 の設定は、ビットの説明では TIMFLTR1～TIMFLTR8 と呼ばれ、タイミングユニットによって内容が異なります。*表 290* はタイマごとの 8 つの利用可能なオプションを示します。CMPx は、カウンタリセットから比較一致までのブランキングを指し、Tx2 は、HRTIM_SETx2 および HRTIM_RSTx2 レジスタで定義される、タイミングユニット TIMx 出力 2 の波形を指します。たとえば、タイマ B（TIMFLTR6）はタイマ C の出力 2 の波形となります。

表 290. タイマごとのフィルタリング信号のマッピング

	転送元	タイマ A				タイマ B				タイマ C				タイマ D				タイマ E			
		CMP1	CMP2	CMP4	TA2	CMP1	CMP2	CMP4	TB2	CMP1	CMP2	CMP4	TC2	CMP1	CMP2	CMP4	TD2	CMP1	CMP2	CMP4	TE2
転送先	タイマ A	-	-	-	-	1	-	2	3	4	-	5	6	7	-	-	-	-	8	-	-
	タイマ B	1	-	2	3	-	-	-	-	4	5	-	6	-	7	-	-	8	-	-	-
	タイマ C	-	1	-	-	2	-	3	4	-	-	-	-	5	-	6	7	-	-	8	-
	タイマ D	1	-	-	-	-	2	-	-	3	4	-	5	-	-	-	-	6	-	7	8
	タイマ E	-	1	-	-	2	-	-	-	3	-	4	5	6	-	7	8	-	-	-	-

図 299 および *図 300* に、通常および延期モードでの、すべてのエッジ対応とレベル対応に対する外部イベントのブランキングの例を示します。

参考資料

図 299. エッジ対応トリガによる外部トリガブランキング

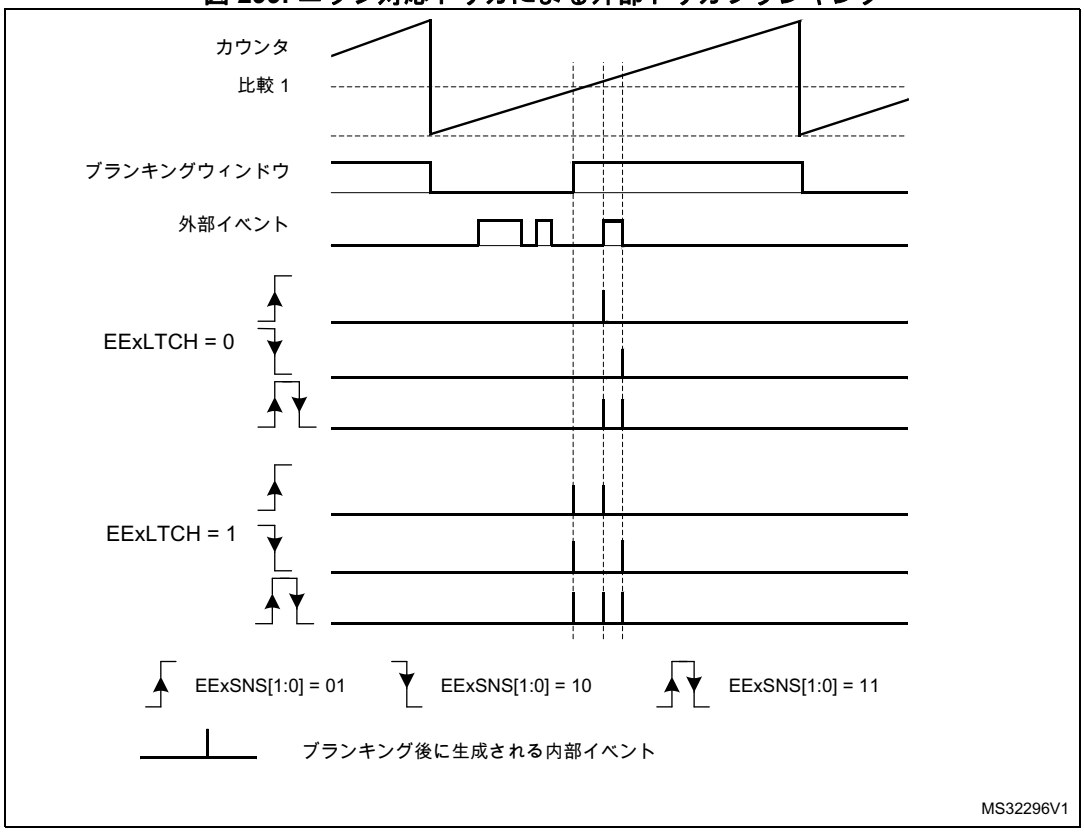
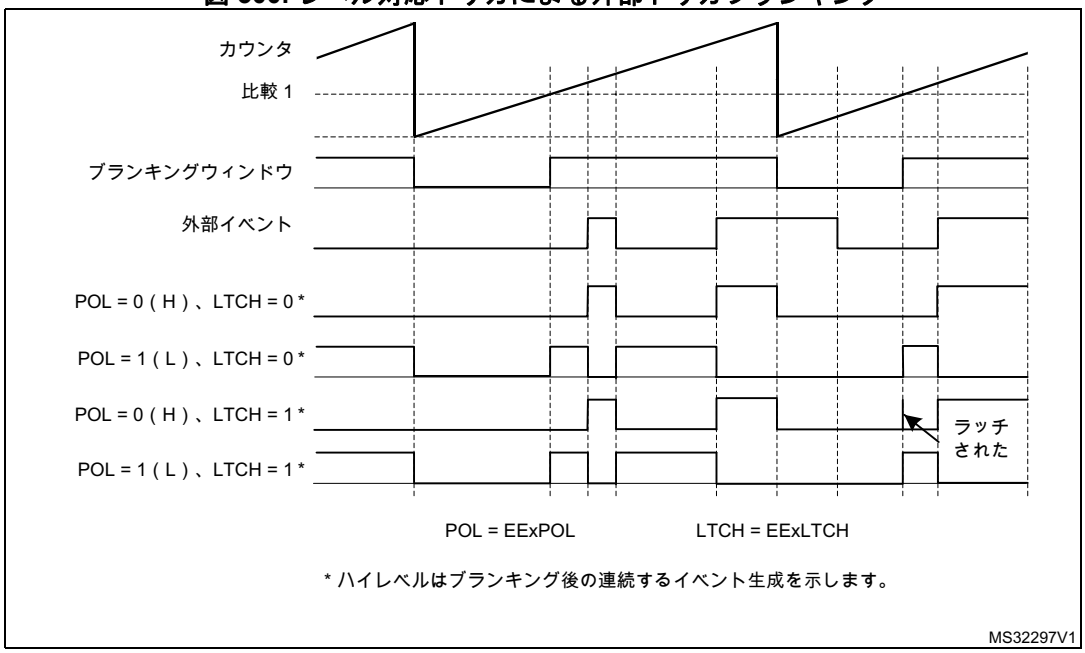


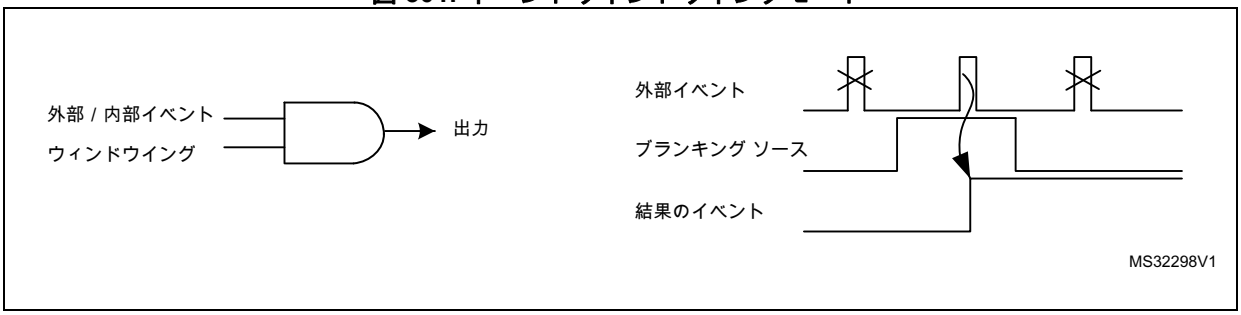
図 300. レベル対応トリガによる外部トリガブランキング



ウィンドウイングモード

イベントウィンドウイングモードでは、イベントは与えられた時間ウィンドウ内で発生した場合のみ、考慮され、そうでない場合は無視されます。このモードは、1101～1111 の範囲の EExFLTR[3:0] に対してアクティブになります。

図 301. イベントウインドウイングモード



EEFxR1 および EEFxR2 レジスタの EExLTCH ビットが 1 にセットされた場合、信号をラッチすることができます。この場合、イベントがウィンドウ内で発生した場合、受け付けられますが、そのイベントはウィンドウの終りまで遅延されます。

- EExLTCH ビットがリセットされていて、信号がウィンドウ内で発生した場合は、それは直接、パススルーされます。
- EExLTCH ビットがリセットされていて、信号が発生しない場合は、ウィンドウの終りにタイムアウトイベントが生成されます。

ウインドウイングモードの使用例は同期信号のフィルタリングです。期待される同期イベントが欠如したとき（たとえば、コンバータの起動中）、タイムアウトを発生させてデフォルトの同期イベントを強制実行できます。

各外部イベントのウィンドウイングのために 3 つのソースがあり、以下のようにコーディングされます。

- 1101 および 1110 : ウィンドウイングはカウンタリセットから比較一致まで続きます（それぞれ比較 2 と比較 3 に対応）。
- 1111 : ウィンドウイングは別のタイミングユニットに関係しており、そのカウンタリセットからその比較 2の 一致まで続きます。そのソースはビットの説明において TIMWIN として説明されており、*表 291* に示されています。例として、タイマ B での外部イベントを、タイマ A のカウンタリセットから始まりタイマ A の比較 2 までのウィンドウによってフィルタリングすることができます。

表 291. タイマごとのウィンドウイング信号のマッピング（EEFLTR[3:0] = 1111）

転送先	タイマ A	タイマ B	タイマ C	タイマ D	タイマ E
TIMWIN（ソース）	タイマ B CMP2	タイマ A CMP2	タイマ D CMP2	タイマ C CMP2	タイマ D CMP2

注 : ***外部イベントが高速モードでプログラムされている場合、タイムアウトイベントの生成はサポートされません。***

図 302 および *図 303* に、さまざまなエッジ対応およびレベル対応に対して、また EExLTCH ビットの設定に応じて、イベントがどのように生成されるかが示されています。タイムアウトイベントは、明確にするために、特に注記しています。

図 302. エッジ対応トリガによる外部トリガウィンドウイング

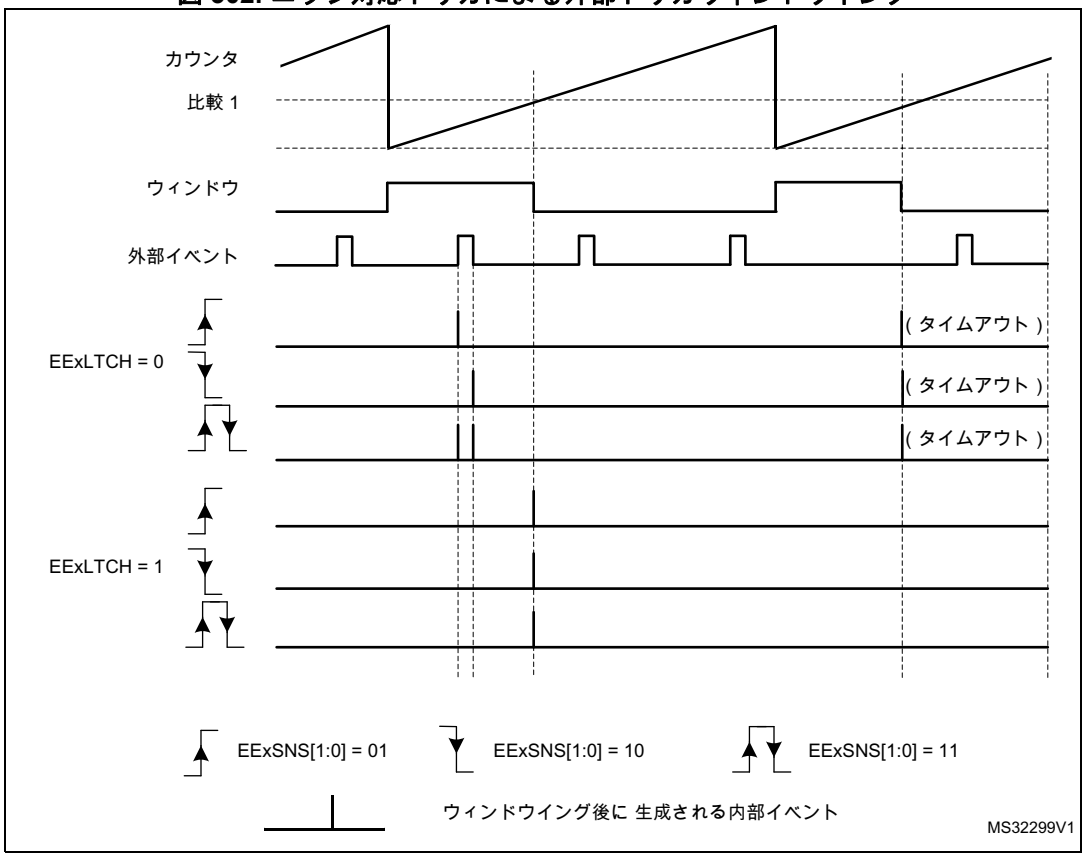
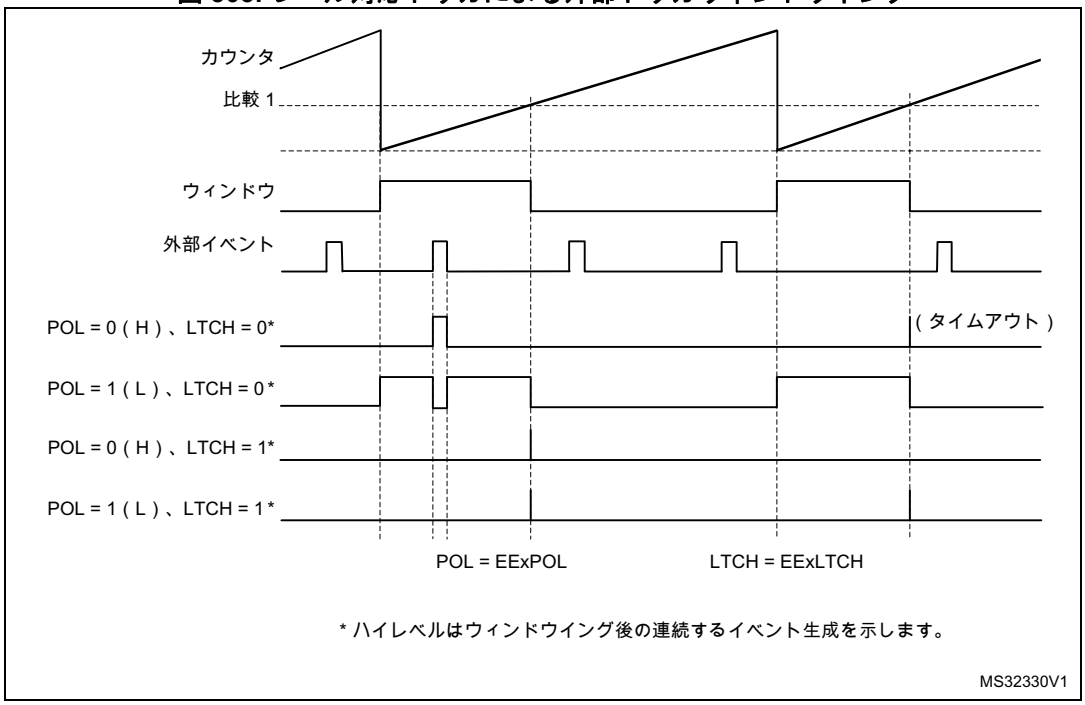


図 303. レベル対応トリガによる外部トリガウィンドウイング



37.3.9 遅延保護

HRTIM は、共振コンバータに典型的ですが、アクティブパルスが完了した後、またはプッシュプル周期が完了した後に PWM 出力を遅延方式で停止する必要があるときのために、特定の保護スキームを備えています。これらの機能は HRTIM_OUTxR レジスタの DLYPRTEN ビットで有効になり、特定の外部イベントチャネルを使用しています。

遅延アイドル

このモードでは、保護が有効化される前にアクティブパルスが完了します。選択された外部イベントは、（HRTIM_RSTx1R または HRTIM_RSTx2R で出力リセットイベントによって定義された）アクティブパルスの終りに出力をアイドルモードに移行させます。

保護にトリガがかかると、アイドルモードは永続的に維持されますが、カウンタは出力が再び有効化されるまで動作し続けます。Tx1OEN および Tx2OEN ビットは、遅延アイドルへの移行の影響を受けません。遅延アイドルから出て動作を再開するには、Tx1OEN および Tx2OEN ビットを 1 で上書きする必要があります。出力の状態は、出力イネーブルコマンドの後、アクティブ状態への最初の遷移時に変わります。

注： ***遅延アイドルモードは、アクティブパルスが完了する前には、モードに入った直後には出ることはできません。実行モードを再開する前に出力がアイドル状態にあることを確認することが必須です。これは、たとえば、次の周期まで待つことによって、あるいは TIMxISR レジスタの O1CPY および／または O2CPY ステータスビットを調査することによって、行うことができます。***

遅延アイドルモードは単一出力（DLYPRT[2:0] = x00 または x01）または両出力（DLYPRT[2:0] = x10）に適用できます。

遅延アイドルモードへの移行に反応して割り込みまたは DMA リクエストを生成することができます。出力上のアクティブパルスの終了とは独立に、外部イベントが着信するとすぐに、HRTIM_TIMxISR の DLYPRT フラグがセットされます。

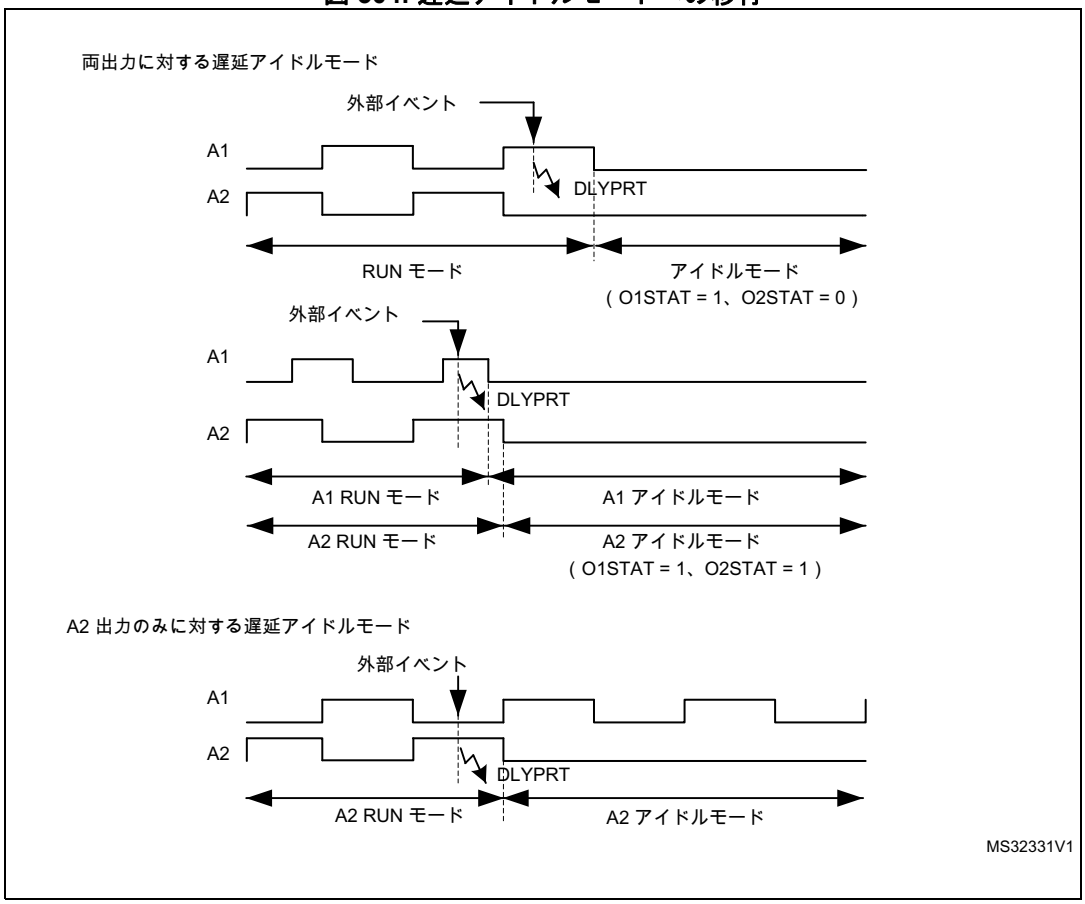
遅延アイドルモードにトリガがかかると、出力状態は HRTIM_TIMxISR の O1STAT および O2STAT を使用して決定できます。遅延アイドルが単一の出力に適用されている場合でも、両方のステータスビットは更新されます。プッシュプルモードが有効になっているとき、HRTIM_TIMxISR の IPPSTAT フラグは、どの周期内で遅延保護リクエストが発生したかを示します。

このモードは、タイマ動作モードが何（通常、プッシュプル、デッドタイム）であれ、使用できます。これは次の 2 つの外部イベントのみで使用できます。

- タイマ A、B、および C に対する hrtim_evt6 および hrtim_evt7
- タイマ D および E に対する hrtim_evt8 および hrtim_evt9

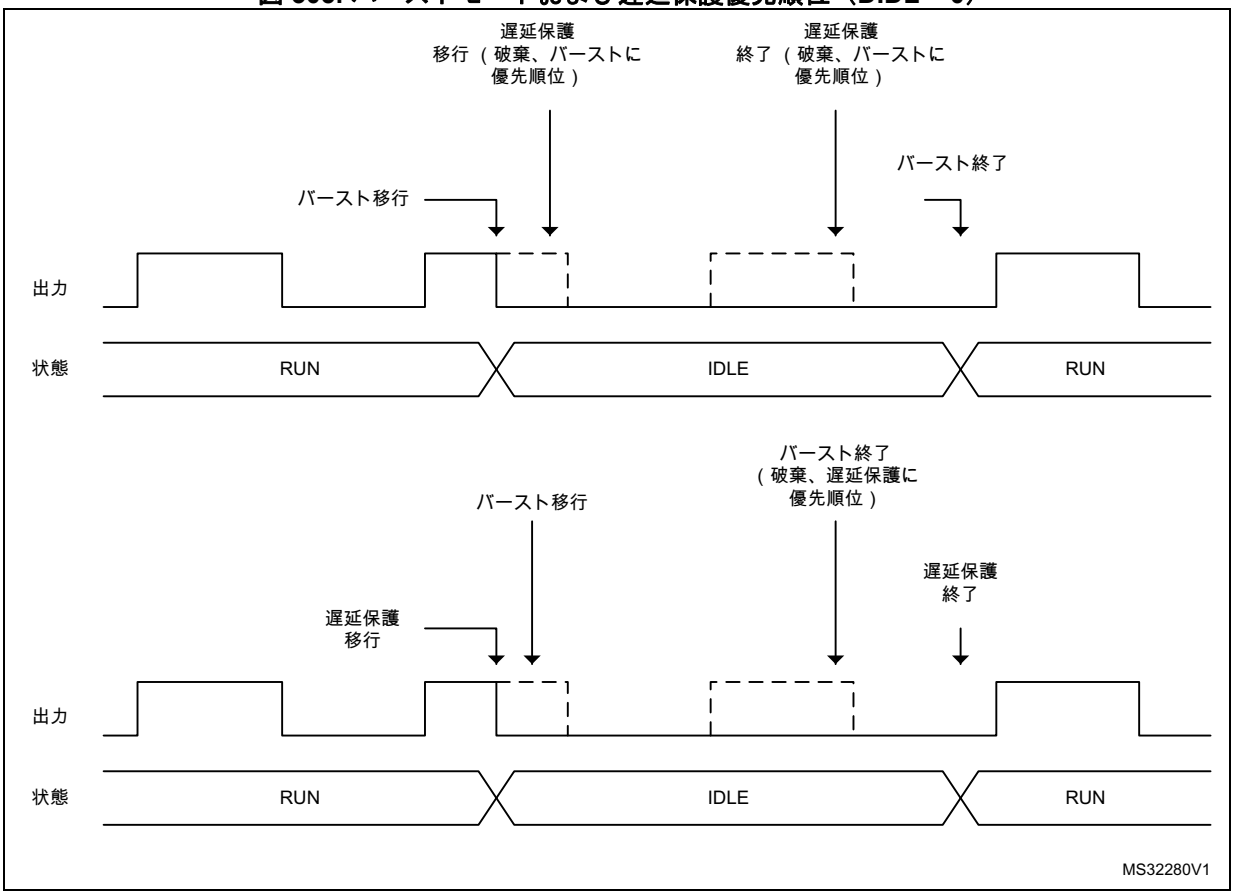
遅延保護モードは、カウンタが有効になっているときのみ（TxCEN ビットのセット）、トリガできます。TxEN ビットがリセットされても、TxyOEN ビットがセットされるまでアクティブのままです。

図 304. 遅延アイドルモードへの移行



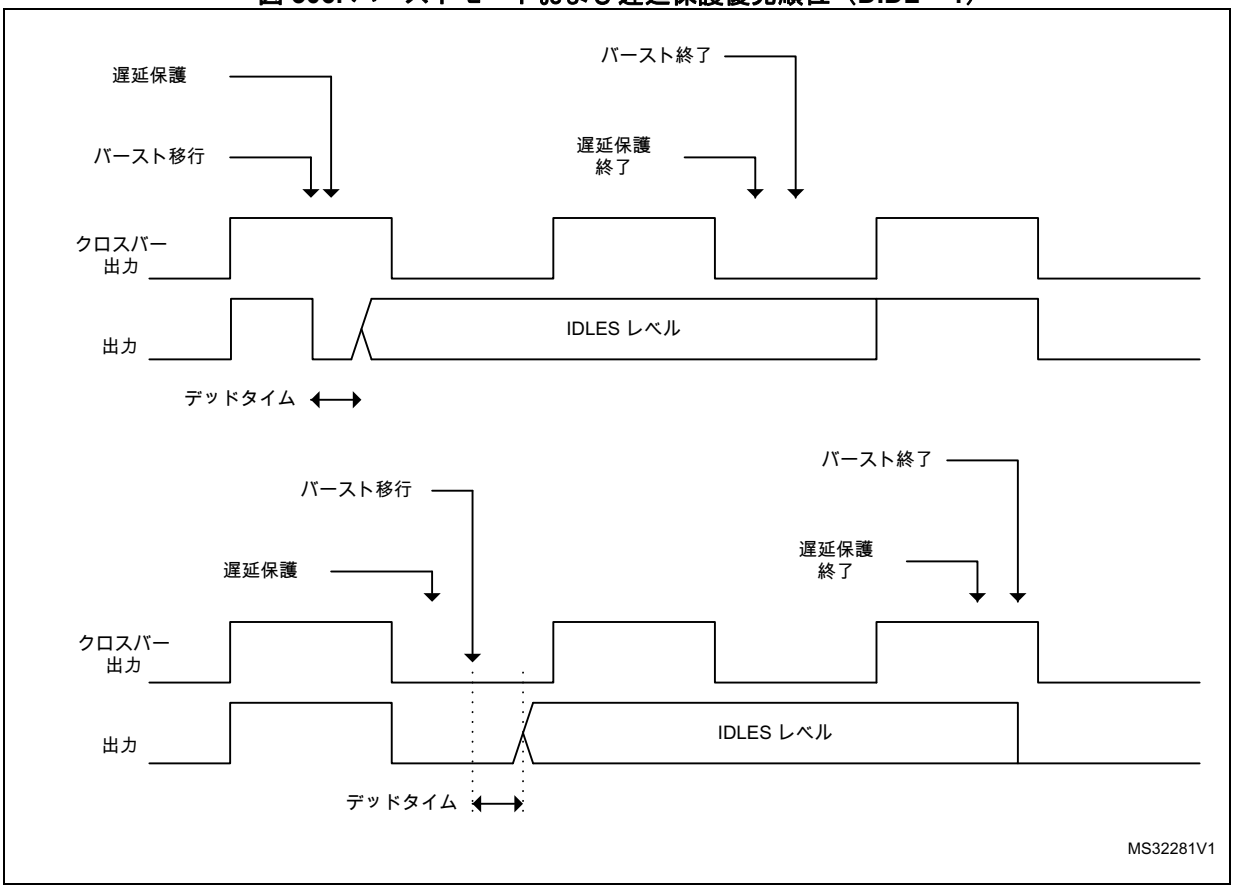
遅延アイドルモードはバーストモードよりも高い優先順位を持ちます。いったん遅延アイドル保護がトリガされるとバーストモードの終了リクエストは破棄されます。反対に、バーストモードがアクティブなときに、遅延保護を終了した場合、バーストモードは正常に再開され、バーストモードが終了するまで、出力はアイドル状態に保持されます。*図 305* に、これらのさまざまなシナリオの概要を示します。

図 305. バーストモードおよび遅延保護優先順位（DIDL = 0）



遅延バーストモードへの移行が有効になっているとき（DIDL ビットのセット）、以下の*図 306* に示すように、同じ優先順位が適用されます。

図 306. バーストモードおよび遅延保護優先順位（DIDL = 1）



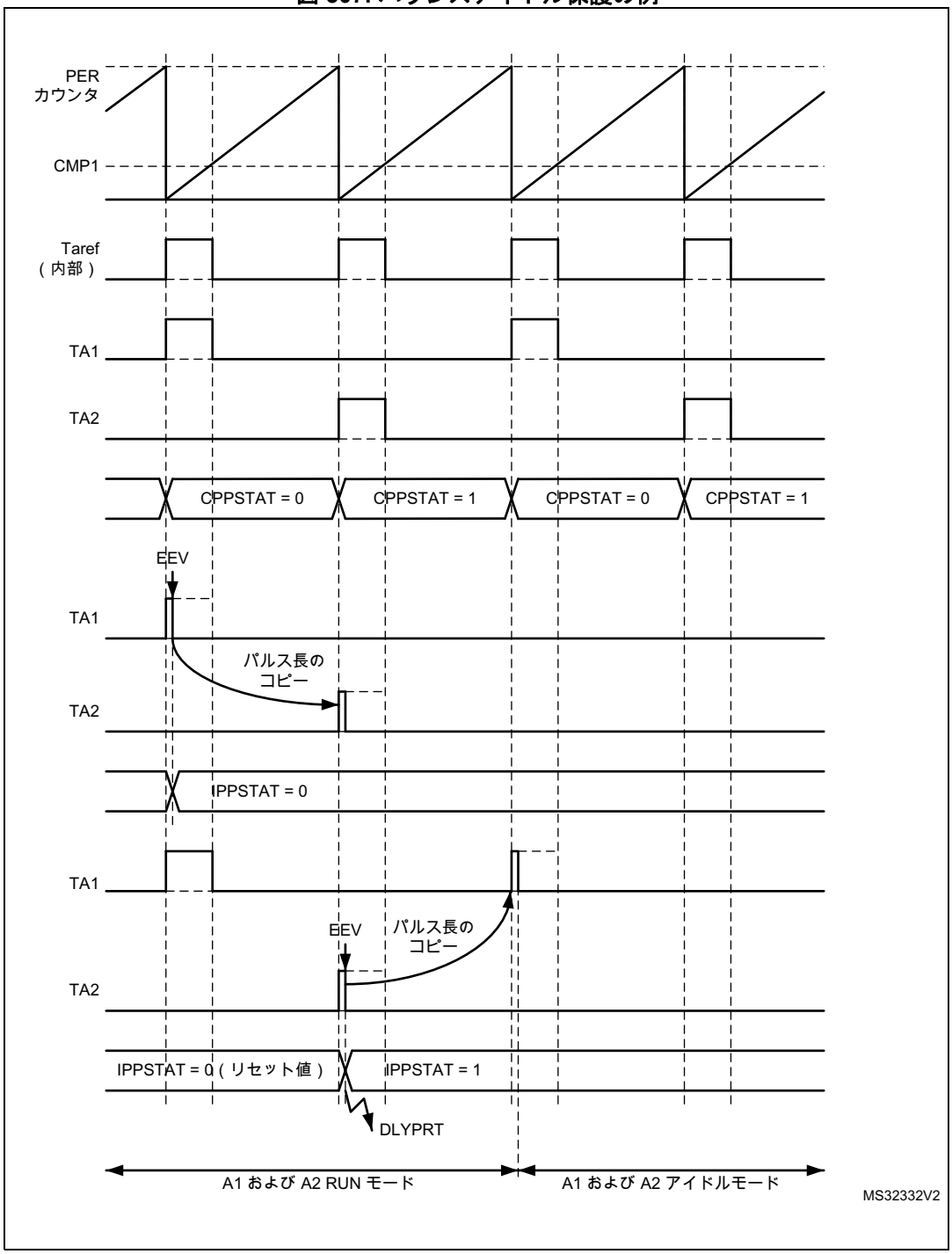
バランスアイドル

プッシュプルモードでのみ利用可能で、アクティブパルスの一方の幅が保護によって短くされたとき、2 つの出力のパルス幅をバランスさせることができます。プログラムされた幅より早く終了されたパルス幅がオルタネート出力にコピーされ、その後、2 つの出力はアイドル状態に入り、ソフトウェアによって通常動作が再開されるまで続きます。このモードは、HRTIM_OUTxR レジスタの DLYPRT[2:0] ビットフィールドに x11 を書き込むことによって有効になります。

このモードは次の 2 つの外部イベントのみで使用できます。

- タイマ A、B、および C に対する hrtim_evt6 および hrtim_evt7
- タイマ D および E に対する hrtim_evt8 および hrtim_evt9

図 307. バランスアイドル保護の例



バランスアイドルモードが有効になっているとき、選択された外部イベントは、比較 4 アクティブレジスタ（この値はユーザにはアクセス不可）へのカウンタ値の取得をトリガします。短縮されたパルスが繰り返されるように、プッシュプルはさらに 1 周期の間、維持されます。通常の出力セットイベントが維持されている間に、新しい出力リセットイベントが生成されます。

その後、アイドルモードに入り、出力は、HRTIM_OUTxR レジスタの IDLESx ビットで定義されたレベルになります。バランスアイドルモードへの移行は DLYPRT フラグによって示され、一方、IPPSTAT フラグはどの周期中に外部イベントが発生したかを示し、短縮されたパルスの順序を決定します（HRTIM_CHA1 から HRTIM_CHA2、または逆）。

タイマの動作は中断されません（カウンタは実行を続けます）。

バランスアイドルモードを有効にするには、以下の初期化が必要です。

- 連続モードでタイマを動作（CONT = 1）
- プッシュプルモードを有効化
- HRTIM_CMP4xR を 0 にセットし、内容をアクティブレジスタに転送する必要があります（たとえば、ソフトウェア更新を強制することによって）。
- DELCMP4[1:0] ビットフィールドを 00 にセットする必要があります（自動遅延モードを無効化）。
- DLYPRT[2:0] = x11（遅延保護を有効化）

注： ***バランスアイドル動作中には、HRTIM_CMP4xR レジスタへは書き込みを行わないでください。CMP4 イベントは予約済みであり、別の目的に使用することはできません。***

バランスアイドルモードでは、複数の外部イベントや出力リセットの原因になるソフトウェアによるリセットイベントは避けることを推奨します。バランスアイドルリクエストの前に同じ周期内でそのようなイベントが着信すると、出力パルスがアンバランスになります（最初のパルス長は外部イベントまたはソフトウェアリセットによって定義されますが、2 番目のパルスはバランスアイドルモードへの移行によって定義されます）。

バランスアイドルモードで処理できる最小のパルス幅は f_{HRTIM} クロックの 4 周期です。

カウンタがこの最小値に達する前にキャプチャが起きると、現在のパルスは、二次出力にコピーされる前に、最大 f_{HRTIM} クロックの 4 周期分まで拡張されます。どの場合でも、パルス幅は常にバランスがとられます。

Tx1OEN および Tx2OEN ビットは、バランスアイドルへの移行の影響を受けません。バランスアイドルから出て動作を再開するには、Tx1OEN および Tx2OEN ビットを同時に 1 で上書きする必要があります。出力の状態は、出力の有効化の後、アクティブへの最初の遷移時に変わります。

遅延アイドルへの移行と同じように動作を再開することができます。たとえば、出力 1 がアクティブのときに外部イベントが到着した場合（出力 2 パルスの後、遅延アイドルが有効）、出力 1 の再開始シーケンスを最初に開始できます。こうするには、HRTIM_TIMxISR レジスタの CPPSTAT ビットを調査する必要があります。上記の例を使用して（IPPSTAT フラグ = 0）、CPPSTAT ビットが 0 のとき、動作は再開されます。

特定の再開シーケンスを行うために、CPPSTAT を調査して、どの出力が最初にアクティブになるかを知ることができます。こうすることで、たとえば、アイドル移行シーケンスと同じシーケンスで再開することができます。つまり、出力 1 がアクティブの間に EEV が到着した場合、出力 1 がアクティブ（CPPSTAT = 0）のときに再開シーケンスが開始されます。

注： ***パルスバランスシーケンスが進行中のときは、バランスアイドルモードは無効にしてはいけません。CMP4 フラグがセットされて、シーケンスが完了したことを示すまで待って、DLYPRTEN ビットをリセットする必要があります。***

バランスアイドル保護モードは、カウンタが有効になっているときのみ（TxCEN ビットのセット）、トリガできます。TxCEN ビットがリセットされても、TxyOEN ビットがセットされるまでアクティブのままです。

以下の条件の下では、バランスアイドルはバーストモードと共に使用できます。

- TxBM ビットをリセットしなければならない（バースト中にカウンタクロックが維持される。*セクション 37.3.13* を参照）。
- 出力がバーストアイドル状態になっているときにバランスアイドル保護をトリガしてはならない。

バランスアイドルモードはバーストモードよりも高い優先順位を持ちます。いったんバランスアイドル保護がトリガされるとバーストモードの終了リクエストは破棄されます。反対に、バーストモードがアクティブなときに、遅延保護を終了した場合、バーストモードは正常に再開されます。

注： ***出力の状態がアイドルモードに固定されていても、遅延保護の後のアイドル期間中に下記のような複数のイベントが補助出力上に生成されます（セクション 37.3.16 参照）。***
- 出力のセット／リセット割り込みまたは DMA リクエスト
- 出力信号に基づく外部イベントのフィルタリング
- セット／リセットにトリガされたキャプチャイベント

37.3.10 レジスタのプリロードおよび更新管理

大部分の HRTIM レジスタはバッファされ、必要な場合、プリロードできます。典型的には、これによって、アクティブなイベントに同期しないレジスタ更新（セット／リセット）によって波形が変更されるのを防ぐことができます。

プリロードモードが有効になっているとき、アクセスされるレジスタはシャドウレジスタです。ソフトウェアによる更新リクエスト、またはイベントに同期した更新リクエストの後に、その内容がアクティブレジスタに転送されます。

デフォルトでは、HRTIM_MCR および HRTIM_TIMxCR レジスタの PREEN ビットはリセットされ、レジスタはプリロードされません。どの書き込みも直接、アクティブレジスタを更新します。タイマが動作中で、プリロードが有効になっていたときに PREEN ビットがリセットされた場合、プリロードレジスタの内容は直接、アクティブレジスタに転送されます。

各タイミングユニットおよびマスタタイマには独自の PREEN ビットがあります。PRREN がセットされると、プリロードレジスタが有効になり、更新イベント時にのみ、アクティブレジスタに転送されます。

プリロード機能が必要なとき、タイマを初期化するには以下の 2 つのオプションがあります。

- タイマの初期化の一番最後に PREEN ビットを有効にし、（MCEN および TxCEN ビットをセットすることで）タイマを有効にする前にプリロードレジスタをアクティブレジスタに転送します。
- 初期化中の任意の時点で PREEN ビットを有効にし、開始直前にソフトウェアによる更新を強制します。

表 292 に、プリロード可能なレジスタを、利用できる更新イベントの概要と共に一覧表示します。

表 292. HRTIM のプリロード可能な制御レジスタおよび関連する更新ソース

タイマ	プリロード可能なレジスタ	プリロードイネーブル	更新ソース
マスタタイマ	HRTIM_DIER HRTIM_MPER HRTIM_MREP HRTIM_MCMP1R HRTIM_MCMP2R HRTIM_MCMP3R HRTIM_MCMP4R	HRTIM_MCR の PREEN ビット	ソフトウェア 繰り返しイベント バースト DMA イベント バースト DMA イベント後の繰り返しイベント
タイマ x x = A～E	HRTIM_TIMxDIER HRTIM_TIMxPER HRTIM_TIMxREP HRTIM_TIMxCMP1R HRTIM_TIMxCMP1CR HRTIM_TIMxCMP2R HRTIM_TIMxCMP3R HRTIM_TIMxCMP4R HRTIM_DTxR HRTIM_SETx1R HRTIM_RSTx1R HRTIM_SETx2R HRTIM_RSTx2R HRTIM_RSTxR	HRTIM_TIMxCR の PREEN ビット	ソフトウェア TIMx 繰り返しイベント TIMx リセットイベント バースト DMA イベント 他のタイマ（TIMy、マスタ）からの更新イベント バースト DMA イベント後の更新イベント 更新イネーブル入力 1～3 更新イネーブル入力 1～3 後の更新イベント
HRTIM 共通	HRTIM_ADC1R HRTIM_ADC2R HRTIM_ADC3R HRTIM_ADC4R	選択したタイマで PREEN = 1 の場合、HRTIM_CR1 の ADxUSRC[2:0] ビットに応じた、TIMx タイマ またはマスタタイマの更新	

マスタタイマには次の 4 つの更新オプションがあります。

1. ソフトウェア : HRTIM_CR2 の MSWU ビットに 1 を書き込むことによって、レジスタの即時更新を強制します。この場合、保留になっていたハードウェアの更新リクエストはキャンセルされます。
2. マスタカウンタがロールオーバーし、マスタ繰り返しカウンタが 0 に等しくなるときに行われる更新。これは、HRTIM_MCR の MREPU ビットがセットされているとき、有効になります。
3. バースト DMA が完了した後、行われる更新（詳細は *セクション 37.3.21* を参照）。これは、HRTIM_MCR の BRSTDMA[1:0] = 01 のとき、有効になります。MREPU=1 および BRSTDMA=01 の両方をセットできます。
   注 : SWU ビットがセットされた場合（すなわち、強制更新モード）、更新はバーストシーケンスの終了直後に起こすことができます。SWU ビットがリセットされた場合、更新は、バーストシーケンスの終了後の次の更新イベント時に行われます。
4. バースト DMA 完了後にマスタカウンタがロールオーバーするときに行われる更新。これは、HRTIM_MCR の BRSTDMA[1:0] = 10 のとき、有効になります。

割り込みまたは DMA リクエストはマスタ更新イベントによって生成できます。

各タイマ（TIMA～E）は、以下のように更新させることもできます。

- ソフトウェアによる：HRTIM_CR2 の TxSWU ビットに 1 を書き込むことによって、レジスタの即時更新を強制します。この場合、保留になっていたハードウェアの更新リクエストはキャンセルされます。
- カウンタがロールオーバーし、繰り返しカウンタが 0 に等しくなるときに行われる更新。これは、HRTIM_TIMxCR の TxREPU ビットがセットされているとき、有効になります。
- カウンタがリセットされた、または連続モードでロールオーバーしたときに行われる更新。これは、HRTIM_TIMxCR の TxRSTU ビットがセットされているとき、有効になります。これは、たとえば、シングルショットモードで動作しているタイマに使用されます。
- バースト DMA が完了した後、行われる更新。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0001 のとき、有効になります。
- バースト DMA の完了後の更新イベント時に行われる更新（イベントは TxREPU、MSTU 、または TxU により有効化できます）。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0010 のとき、有効になります。
- 更新イネーブル入力 1～3 によるリクエストを受信したときに行われる更新。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0011、0100、0101 のとき、有効になります。
- 更新イネーブル入力 1～3 によるリクエスト後の更新イベント時に行われる更新（イベントは TxREPU、MSTU 、または TxU により有効化できます）。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0110、0111、1000 のとき、有効になります。
- 他のタイマまたはマスタ更新と同期して行われる更新（たとえば、TIMA は TIMB と同時に更新できます）。これは、複数のタイマを必要とするコンバータ用に使用され、HRTIM_TIMxCR レジスタの MSTU および TxU ビットのセットによって有効になります。

更新イネーブル入力 1～3 によって、更新イベントを汎用タイマから来るオンチップイベントと同期させることができます。これらの入力は立ち上がりエッジ対応です。

表 293 に、更新イネーブル入力とオンチップソースとの接続を示します。

表 293. 更新イネーブル入力とソース

更新イネーブル入力	更新ソース
更新イネーブル入力 1	TIM16_OC
更新イネーブル入力 2	TIM17_OC
更新イネーブル入力 3	TIM6_TRGO

これによって、低周波数の更新リクエストを高周波信号と同期させることができます（たとえば、100 Hz の速度で行う必要のある、100 kHz PWM のカウンタロールオーバー時の更新）。

注： ***CKPSC[2:0] > 5 のとき、更新イベントはプリスケーラクロックに同期します。***

割り込みまたは DMA リクエストは Timx 更新イベントによって生成できます。

選択された更新イベントが何であれ、HRTIM_CR1 レジスタの MUDIS および TxUDIS ビットによって、プリロードレジスタからアクティブレジスタへの転送を一時的に無効にすることができます。これによって、複数のタイマのいくつかのレジスタが変更できるようになります。これらのビットがリセットされると、通常の更新イベントが発生します。

MUDIS および TxUDIS ビットはすべて同じレジスタにグループ化されています。このため、複数のタイマの更新を（必ずしも同期する必要はないが）同時に無効にして再開することが可能になります。

以下の例は実際的な使用事例です。最初の電力コンバータはマスタ、TIMB、および TIMC によって制御されています。TIMB および TIMC はマスタタイマの繰り返しイベントによって同時に更新する必要があります。2 番目のコンバータは TIMA、TIMD、および TIME と並行に動作し、TIMD、TIME は TIMA の繰り返しイベントによって更新する必要があります。

最初のコンバータ

HRTIM_MCR で、MREPU ビットをセットします。マスタタイマのカウンタの繰り返し周期の終りで更新が起きます。HRTIM_TIMBCR および HRTIM_TIMCCR の MSTU ビットがセットされ、TIMB および TIMC タイマがマスタタイマによって同時に更新されます。

電力コンバータの設定値をソフトウェアで調節する必要があるとき、レジスタに書き込みアクセスして値（たとえば、比較値）を更新する前に HRTIM_CR レジスタの MUDIS、TBUDIS、および TCUDIS ビットをセットする必要があります。この時点以降、ハードウェア更新リクエストは無視され、プリロードレジスタは、アクティブレジスタに転送される恐れなく、アクセスできます。ソフトウェア処理が終わると、MUDIS、TBUDIS、および TCUDIS ビットはリセットする必要があります。プリロードからアクティブレジスタへの転送は、マスタ繰り返しイベントが発生するとすぐに行われます。

2 番目のコンバータ

HRTIM_TIMACR で、TAREPU ビットをセットします。タイマ A カウンタの繰り返し周期の終りで更新が起きます。HRTIM_TIMDCR および HRTIM_TIMECR の TAU ビットがセットされ、TIMD および TIME タイマがタイマ A によって同時に更新されます。

電力コンバータの設定値をソフトウェアで調節する必要があるとき、レジスタに書き込みアクセスして値（たとえば、比較値）を更新する前に HRTIM_CR レジスタの TAUDIS、TDUDIS、および TEUDIS ビットをセットする必要があります。この時点以降、ハードウェア更新リクエストは無視され、プリロードレジスタは、アクティブレジスタに転送される恐れなく、アクセスできます。ソフトウェア処理が終わると、TAUDIS、TDUDIS、および TEUDIS ビットはリセットできます。プリロードからアクティブレジスタへの転送は、タイマ A の繰り返しイベントが発生するとすぐに行われます。

37.3.11 複数タイマ内またはそれらをまたがったイベントの伝播

HRTIM は、複数のタイミングユニット（マスタタイマを含む）にまたがってイベントをカスケード伝達または共有させる多くの可能性を提供し、そのモジュラーアーキテクチャの利点をフルに活用します。これらは複数の同期出力を必要とするコンバータ用の重要な機能です。

このセクションでは、さまざまなオプションを概説し、イベントが HRTIM 内で伝播されるかどうか、またどのように伝播されるかを具体的に述べます。

マスタタイマの更新によってトリガされる TIMx の更新

表 294 に列挙されたソースはマスタタイマの更新を生成します。表は、ソースイベントが、どの TIMx タイミングユニットにも同時に更新のトリガをかけるのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの MSTU ビットがセットされていることです。

表 294. マスタタイマ更新イベントの伝播

転送元	条件	伝播	コメント
バースト DMA 終了	BRSTDMA[1:0] = 01	なし	TIMxCR で行われる必要あり（UPDGAT[3:0] = 0001）
バースト DMA 終了後のロールオーバーイベント	BRSTDMA[1:0] = 10	あり	-
カウンタのロールオーバーに起因する繰り返しイベント	MREPU = 1	あり	-
カウンタのリセットに起因する繰り返しイベント（HRTIM_SCIN またはソフトウェアから）		なし	-
ソフトウェア更新	MSWU = 1	なし	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、同時更新に使用可能

TIMy の更新によってトリガされる TIMx の更新

表 295 に列挙されたソースは TIMy の更新を生成します。表は、与えられたイベントが、他のまたは複数の TIMx タイマに同時に更新のトリガをかけるのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの TyU ビットをセット（ソース = TIMy および転送先 = TIMx）

表 295. TIMx 更新イベントの伝播

転送元	条件	伝播	コメント
バースト DMA 終了	UPDGAT[3:0] = 0001	なし	直接、HRTIM_TIMxCR で行われる必要あり（UPDGAT[3:0] = 0001）
更新イネーブル入力に起因する更新	UPDGAT[3:0] = 0011、0100、0101	なし	直接、HRTIM_TIMxCR で行われる必要あり（UPDGAT[3:0] = 0011、0100、0101）
マスタ更新	HRTIM_TIMyCR で、MSTU = 1	なし	HRTIM_TIMxCR の MSTU = 1 で行われる必要あり
別の TIMx 更新（TIMz>TIMy>TIMx）	HRTIM_TIMyCR で、TzU = 1 TIMxCR で、TyU=1	なし	HRTIM_TIMxCR の TzU = 1 で行われる必要あり HRTIM_TIMyCR で、TzU = 1
カウンタのロールオーバーに起因する繰り返しイベント	TyREPU = 1	あり	-
カウンタのリセットに起因する繰り返しイベント	TyREPU = 1	-	下記のカウンタリセットの場合を参照
カウンタのロールオーバー	TyRSTU = 1	あり	-
カウンタのソフトウェアリセット	HRTIM_CR2 で、TyRST = 1	なし	HRTIM_CR2 レジスタの更新で同時に行うことが可能
TIMz 比較に起因するカウンタリセット	HRTIM_RSTyR の TIMzCMPn	なし	HRTIM_RSTxR の TIMzCMPn を使用して行われる必要あり
外部イベントに起因するカウンタリセット	HRTIM_RSTyR の EXTEVNTn	あり	-

表 295. TIMx 更新イベントの伝播（続き）

転送元	条件	伝播	コメント
マスタ比較またはマスタ周期に起因するカウンタリセット	HRTIM_RSTyR の MSTCMPn または MSTPER	なし	-
TIMy 比較に起因するカウンタリセット	HRTIM_RSTyR の CMPn	あり	-
更新に起因するカウンタリセット	HRTIM_RSTyR の UPDT	なし	伝播はロックアップ状況を招く恐れあり（更新はリセットを引き起こし、それは更新を引き起こす）
HRTIM_SCIN に起因するカウンタリセット	HRTIM_TIMyCR の SYNCRSTy	なし	-
ソフトウェア更新	TySWU = 1	なし	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、同時更新に使用可能

TIMx 更新を引き起こす TIMx カウンタリセット

表 296 はカウンタリセットのソースを挙げ、それらが更新を生成するのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの TxRSTU ビット

表 296. 更新を生成できるリセットイベント

転送元	条件	伝播	コメント
カウンタのロールオーバー		あり	
更新イベント	HRTIM_RSTxR の UPDT	なし	伝播はロックアップ状況を招く恐れあり（更新はリセットを引き起こし、それは更新を引き起こす）
外部イベント	HRTIM_RSTxR の EXTEVNTn	あり	-
TIMy 比較	HRTIM_RSTxR の TIMyCMPn	あり	-
マスタ比較	HRTIM_RSTxR の MSTCMPn	あり	-
マスタ周期	HRTIM_RSTxR の MSTPER	あり	-
比較 2 および 4	HRTIM_RSTxR の CMPn	あり	-
ソフトウェア	HRTIM_CR2 で、TxRST = 1	あり	-
HRTIM_SCIN	HRTIM_TIMxCR の SYNCRSTx	あり	-

TIMx カウンタのリセットを引き起こす TIMx 更新

表 297 は更新イベントのソースを挙げ、それらがカウンタリセットを生成するのに使用できるかどうかを示しています。

動作条件：HRTIM_RSTxR で UPDT ビットのセット

表 297. タイマのリセットのための更新イベントの伝播

転送元	条件	伝播	コメント
バースト DMA 終了	UPDGAT[3:0] = 0001	あり	-
更新イネーブル入力に起因する更新	UPDGAT[3:0] = 0011、0100、0101	あり	-
バースト DMA 後のロールオーバーに起因するマスタ更新	HRTIM_TIMxCR で、MSTU = 1 HRTIM_MCR で、BRSTDMA[1:0] = 10	あり	-
ロールオーバー後の繰り返しイベントに起因するマスタ更新	HRTIM_TIMxCR で、MSTU = 1 HRTIM_MCR で、MREPU = 1	あり	-
カウンタリセット後の繰り返しイベントに起因するマスタ更新（ソフトウェアまたは HRTIM_SCIN による）		なし	-
ソフトウェアトリガによるマスタタイマの更新	HRTIM_TIMxCR で、MSTU = 1 HRTIM_CR2 で、MSWU = 1	なし	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、同時更新に使用可能
TIMy カウンタのロールオーバーに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyRSTU = 1	あり	-
TIMy 繰り返しイベントに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyREPU = 1	あり	-
外部イベントまたは TIMy 比較に起因する TIMy 更新（TIMy リセットを通して）	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyRSTU = 1 EXTEVNTn または CMP4/2 （HRTIM_RSTyCR で）	あり	-
上記以外のソースに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1	なし	-
ロールオーバー後の繰り返しイベント	HRTIM_TIMxCR で、TxREPU = 1	あり	-
カウンタのリセット後の繰り返しイベント		なし	-

表 297. タイマのリセットのための更新イベントの伝播（続き）

転送元	条件	伝播	コメント
タイマのリセット	HRTIM_TIMxCR で、TxRSTU = 1	なし	伝播はロックアップ状況を招く恐れあり（リセットは更新を引き起こし、それはリセットを引き起こす）
ソフトウェア	HRTIM_CR2 の TxSWU	なし	-

37.3.12 出力管理

各タイミングユニットは 1 対の出力を制御します。出力には、3 つの動作状態があります。

- RUN：これは主動作モードであり、出力はクロスバーでプログラムされたとおり、アクティブまたは非アクティブになります。
- IDLE：この状態は、出力がソフトウェアによって無効にされているとき、またはバーストモード動作中の（そこでは、通常動作モード中、出力は一時的に無効にされます。詳細は *セクション 37.3.13* を参照）、HRTIM リセット後のデフォルトの動作状態です。永続的にアクティブまたは非アクティブになります。
- FAULT：これは安全状態であり、FAULTx 入力時の停止リクエストの場合に、この状態になります。永続的にアクティブ、非アクティブ、またはハイインピーダンスになります。

表 298 に示すように、出力ステータスは HRTIM_OENR レジスタの TxyOEN ビットおよび HRTIM_ODSR レジスタの TxyODS ビットによって示されます。

表 298. 出力状態のプログラミング（x = A～E、y = 1 または 2）

TxyOEN（制御／ステータス）（ソフトウェアでセットされ、ハードウェアでクリアされる）	TxyODS（ステータス）	出力動作状態
1	x	RUN
0	0	IDLE
0	1	FAULT

TxyOEN ビットは制御とステータスの両方のビットです。出力を RUN モードにするにはソフトウェアでセットする必要があります。出力が IDLE または FAULT モードになったときにハードウェアによってクリアされます。TxyOEN ビットがクリアされたとき、TxyODS ビットは出力が IDLE 状態なのか、または FAULT 状態なのかを示します。HRTIM_ODISR レジスタの 3 番目のビットによって、出力をソフトウェアで無効にできます。

図 308. 出力管理の概要

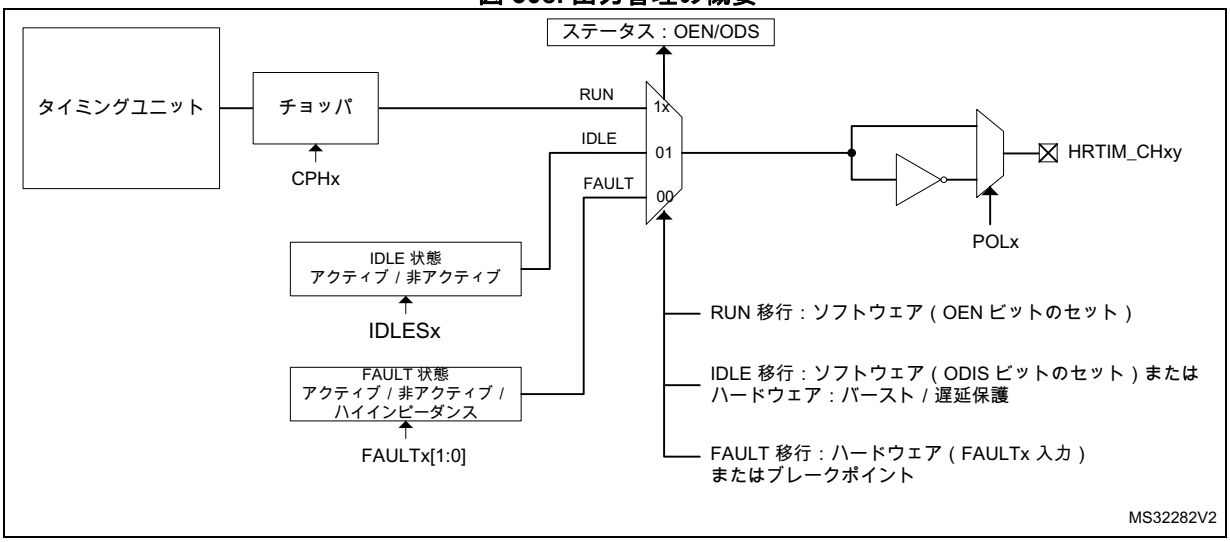
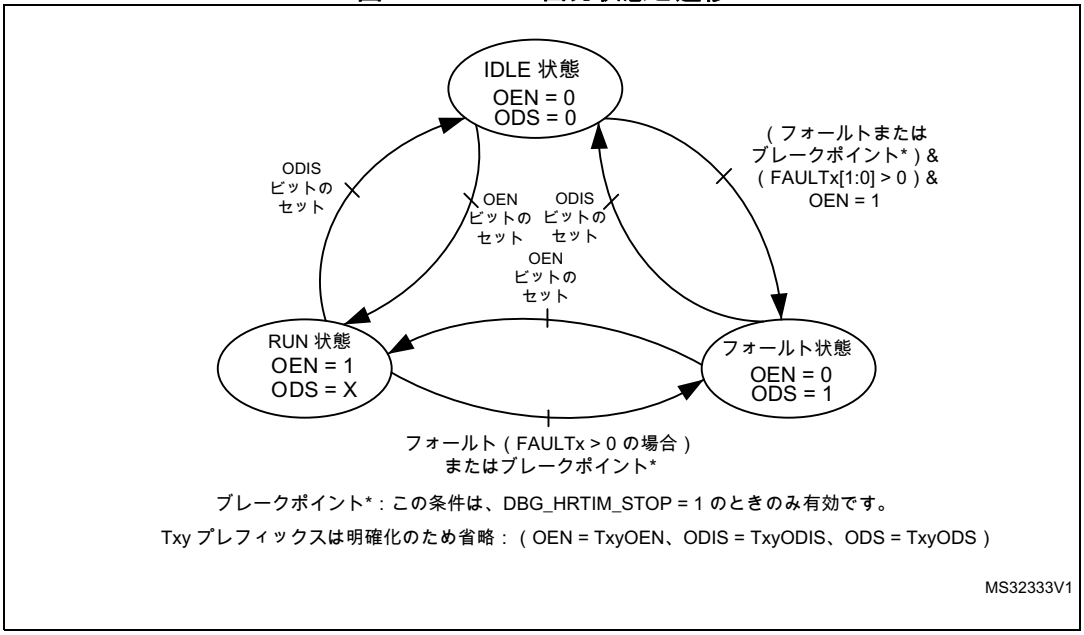


図 309 は、3 つの状態に対するビット値と、遷移がどのようにトリガされるかを示します。*セクション 37.3.15* に挙げたように、フォールトは外部または内部フォールトソースによってトリガされますが、バーストモードまたは遅延保護がアクティブになっているとき、アイドル状態に移行することができます。

図 309. HRTIM 出力状態と遷移



FAULT および IDLE レベルはアクティブまたは非アクティブとして定義されます。アクティブ（または非アクティブ）は、電源スイッチを閉（非アクティブ状態の場合は開）に導くタイマ出力のレベルを指します。

IDLE 状態は最も高い優先順位です。FAULT 状態が有効でも、ODIS ビットのセットにトリガされて FAULT → IDLE の遷移は可能です。

FAULT 状態は RUN 状態より優先順位が高く、TxyOEN ビットがフォールトイベントと同時にセットされた場合、FAULT 状態に移行します。IDLE → FAULT の遷移は、*図 309* のように、次の条件で起

きます。すなわち、フォールト保護が有効化され（FAULTx[1:0] ビット = 01、10、11）、かつ、フォールトがアクティブで（または DBG_HRTIM_STOP = 1 の場合、ブレークポイント中に）OEN ビットがセットされる必要があります。

出力の極性は HRTIM_OUTxR の POLx ビットを使用してプログラムされます。POLx = 0 のとき極性は正（出力はアクティブハイ）で、負の極性（POLx = 1）の場合は、アクティブローとなります。実用的には、極性は駆動する電源スイッチ（PMOS 対 NMOS）に応じて、またはゲートドライバの極性に応じて定義されます。

FAULT 状態の出力レベルは、以下のように各出力に対して、HRTIM_OUTxR の FAULTx[1:0] ビットを使用して設定します。

- 00：出力はフォールト状態には移行せず、RUN または IDLE 状態に留まります。
- 01：FAULT 状態のとき、出力はアクティブレベルになります。
- 10：FAULT 状態のとき、出力は非アクティブレベルになります。
- 11：FAULT 状態のとき、出力はトライステートになります。たとえば、安全状態はプルアップ抵抗またはプルダウン抵抗によって外部的に強制する必要があります。

注： ***FAULTx[1:0] ビットは、出力が FAULT 状態である限り、変更してはなりません。***

IDLE 状態の出力レベルは、HRTIM_OUTxR の IDLESx ビットを使用して以下のように設定します。

- 0：IDLE 状態のとき、出力は非アクティブレベルになります。
- 1：IDLE 状態のとき、出力はアクティブレベルになります。

TxyOEN ビットがセットされて RUN 状態に移行すると、出力は直ちにクロスバー出力に接続されます。タイマクロックが停止すると、レベルは非アクティブになるか（HRTIM リセット後）、または RUN レベルに対応します（タイマが停止して出力が無効になったとき）。

HRTIM 初期化中に、HRTIM_SETx1R および HRTIM_RSTx1R レジスタでのソフトウェアによる出力セット・リセットを使用して、出力レベルを RUN モードに移行する前に事前設定できます。

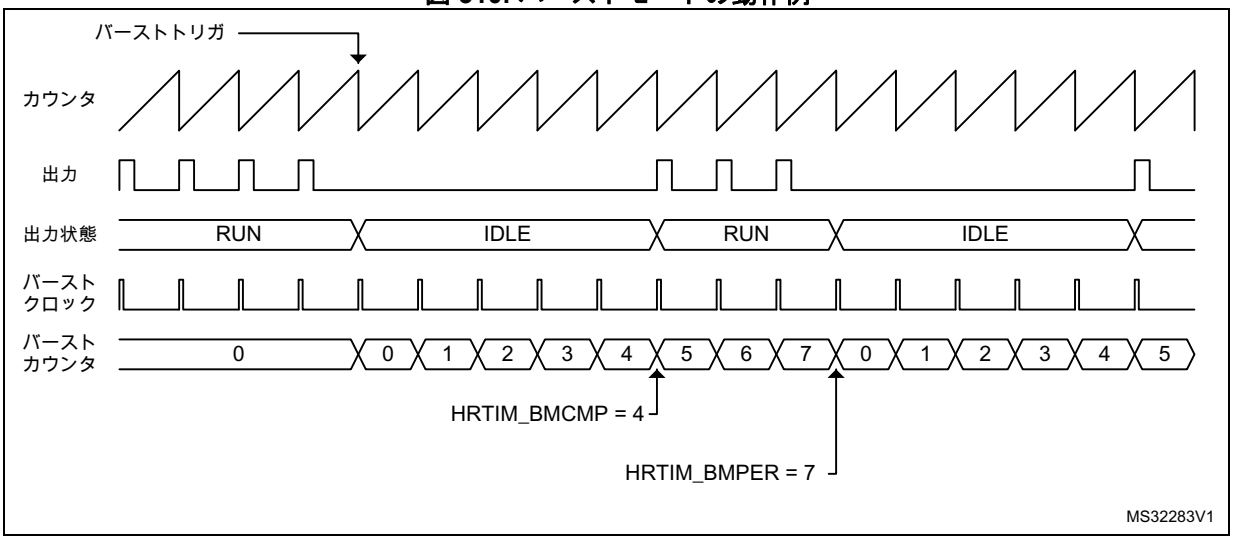
37.3.13 バーストモードコントローラ

バーストモードコントローラによって、出力をハードウェアで交互に IDLE 状態と RUN 状態にすることができます。こうすることによって、いくつかのスイッチング周期をプログラム可能な周期性とデューティサイクルでスキップすることができます。

バーストモード動作は、軽負荷で動作するときの電力コンバータで一般に使用されています。これによって、出力の遷移回数とそれに伴うスイッチング損失を低減することによって、コンバータの効率を大幅に向上できます。

バーストモードで動作するとき、カウントの数周期に等しいアイドル期間に続いて 1 つまたは数パルスが出力されます。一般的にアイドル期間では、*図 310* の例に示すように、出力パルスは生成されません。

図 310. バーストモードの動作例



バーストモードコントローラは以下のものから構成されます。

- HRTIM の内部または外部（典型として PWM 周期の終端）のさまざまなソースによってクロック供給されるカウンタ。
- アイドル周期の数を定義する比較レジスタ：HRTIM_BMCMP。
- バーストの繰り返しレート（アイドル期間と実行期間の合計に相当）を定義する周期レジスタ：HRTIM_BMPER。

バーストモードコントローラは 10 個の PWM 出力を制御することができます。*表 299* に示すように、バーストモード動作中の各出力の状態は、HRTIM_OUTxR レジスタの IDLESx および IDLEMx ビットを使用してプログラムされます。

表 299. バーストモードのためのタイマ出力のプログラミング

IDLEMx	IDLESx	バーストモード中の出力の状態
0	X	作用なし：出力はバーストモード動作の影響を受けません。
1	0	バーストモード中、出力は非アクティブ
1	1	バーストモード中、出力はアクティブ

注： ***IDLEMx ビットは、バーストモードがアクティブである間は、変更してはなりません。***

バーストモードコントローラは出力ステージにのみ作用します。アイドル期間中も、以下の多くのイベントが発生します。

- 出力のセット／リセット割り込みまたは DMA リクエスト
- Tx2 出力信号に基づく外部イベントのフィルタリング
- 出力セット／リセットにトリガされたキャプチャイベント

バーストモード中には、TxBM ビットがセットされていても、HRTIM_SCOUT 出力には開始イベントもリセットイベントも生成されません。

動作モード

所定のタイミングユニットにバーストモードを使用する前にカウンタを有効に（TxCEN ビットをセット）する必要があります。バーストモードは HRTIM_BMCR レジスタの BME ビットで有効にします。

HRTIM_BMCR レジスタの BMOM ビットを使用して、連続またはシングルショットモードで動作させることができます。連続モードは、BMOM =1 のときに有効になります。バースト動作は、HRTIM_BMCR の BMSTAT ビットをリセットして動作を終了させるまで維持されます。

シングルショットモード（BMOM = 0）では、バーストモードのトリガ後、アイドルシーケンスが 1 回実行され、その後すぐに、通常のタイマ動作が再開されます。

アイドル期間および実行期間の持続時間はバーストモードカウンタと 2 つのレジスタで定義されます。HRTIM_BMCMPR レジスタは、選択されたタイマがアイドル状態にある間（アイドル期間）のカウント数を定義します。HRTIM_BMPER は、全体のバーストモードの周期（アイドル期間と実行期間の合計）を定義します。最初のバーストモードのトリガが発生すると、アイドル期間の長さは HRTIM_BMCMPR+1 となり、全体のバースト周期は HRTIM_BMPER+1 となります。

注： ***バーストモードの周期は、DTRx[8:0] および DTFx[8:0] ビットフィールドで定義されたデッドタイム期間以下であってはなりません。***

バーストモード動作中、タイミングユニットおよびマスタタイマのカウンタは停止させてリセットすることができます。このために、HRTIM_BMCR には、次の 6 つの制御ビットがあります。MTBM（マスタ）および、タイマ A～E 用の TABM～TEBM です。

MTBM または TxBM ビットがリセットされているとき、カウンタクロックは維持されます。これによって、たとえば、多相システムにおいて他のタイマとの位相関係を保つことができます。

MTBM または TxBM ビットがセットされているとき、対応するカウンタは、アイドル期間中に停止され、リセット状態に維持されます。これによって、アイドルから出た時に、タイマをフル周期でリスタートさせることができます。SYNCSRC[1:0] = 00 または 10（マスタ開始時またはタイマ A 開始時に同期出力）である場合、バーストモードが終了した時、HRTIM_SCOUT 出力にパルスが送信されます。

注： ***TxBM ビットは、バランスアイドルモードがアクティブなときは（DLYPRT[1:0] = 0x11）、セットしてはいけません。***

バーストモードクロック

バーストモードコントローラのカウンタは、HRTIM_BMCR レジスタの BMCLK[3:0] ビットで選択される下記のいくつかのソースをクロックとして動作できます。

- BMCLK[3:0] = 0000～0101：マスタタイマおよび TIMA～E リセット／ロールオーバーイベント。これによって、バーストモードのアイドルおよび実行周期をタイミングユニットのカウント周期にそろえることができます（フリーランニングモードおよびカウンタリセットモードの両方で）。
- BMCLK[3:0] = 0110～1001：*表 300* のように、クロックは汎用タイマによって供給されます。この場合、バーストモードのアイドルおよび実行周期は、必ずしもタイミングユニットのカウント周期にそろうとは限りません（たとえば、出力上のパルスは中断されて、修正されたデューティサイクルの波形になることもあります）。
- BMCLK[3:0] = 1010：HRTIM_BMCR レジスタの BMPRSC[3:0] ビットで定義された分周比によってプリスケールされた f_{HRTIM} クロック。この場合、バーストモードのアイドルおよび実行周期は、必ずしもタイミングユニットのカウント周期にそろうとは限りません（たとえば、出力上のパルスは中断されて、修正されたデューティサイクルの波形になることもあります）。

表 300. 汎用タイマからのバーストモードのクロックソース

BMCLK[3:0]	クロックソース
0110	hrtim_bm_ck1：TIM16 OC
0111	hrtim_bm_ck2：TIM17 OC
1000	hrtim_bm_ck3：TIM7 TRGO
1001	hrtim_bm_ck4：予約済み

TIMxx OC 出力のパルス幅は、HRTIM バーストモードコントローラが検出できるように、少なくとも f_{HRTIM} の N クロックサイクルの長さでなければなりません。

バーストモードトリガ

バースト動作をトリガするために、以下の 32 個のソースが利用でき、HRTIM_BMTRGR レジスタを使用して選択します。

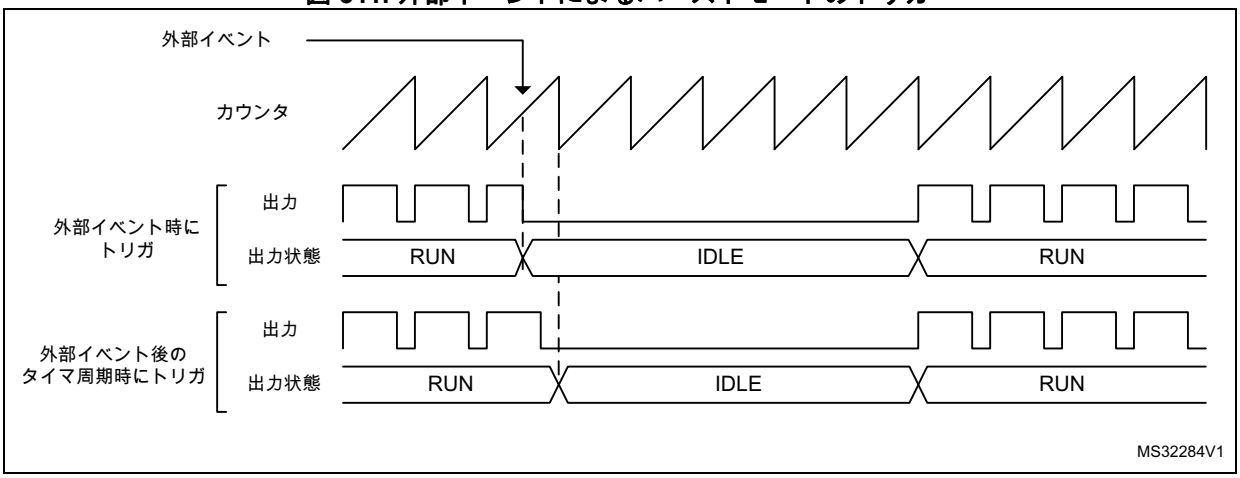
- ソフトウェアトリガ（ソフトウェアによってセットされ、ハードウェアによってリセットされます）
- 6 つのマスタタイマイベント：繰り返し、リセット／ロールオーバー、比較 1～4
- タイマ A～E からの 5 x 4 イベント：繰り返し、リセット／ロールオーバー、比較 1 および 2
- hrtim_evt7（TIMA イベントフィルタリングを含む）および hrtim_evt8（TIMD イベントフィルタリングを含む）
- hrtim_evt7（TIMA イベントフィルタリングを含む）後のタイマ A 周期
- hrtim_evt8（TIMD イベントフィルタリングを含む）後のタイマ D 周期
- 他の汎用タイマから来るオンチップイベント（hrtim_bm_trg output：TIM7_TRGO 出力）

これらのソースは組み合わせて複数同時トリガとすることができます。

バーストモードは再トリガ可能ではありません。連続モードでは、バーストモードが終了するまで新しいトリガは無視されますが、シングルショットモードでは、実行期間（HRTIM_BMPER+1 サイクル）を含む現在のバーストが完了するまで、トリガが無視されます。これは、ソフトウェアトリガについてもあてはまります（ソフトウェアビットは、破棄された場合、ハードウェアイベントによってリセットされます）。

図 311 に、バーストモードがどのように外部イベントに反応して開始されるか（直ちにか、あるいはイベント後のタイマ周期時か）を示します。

図 311. 外部イベントによるバーストモードのトリガ



TAEEV7 と TDEEV8 の組み合わせトリガ（外部イベント後のタイマ周期時にトリガ）に対しては、バーストモードのプログラミングおよび進行中のバースト動作にかかわらず、以下のように、外部イベントの検出は常にアクティブです。

- 外部イベントとタイマ周期イベントの間で、バーストモードが有効（BME=1）、またはトリガが有効（BMTRG レジスタの TAEEV7 または TDEEV8 ビットがセット）になると、バーストはトリガされます。
- シングルショットバーストモードは、バーストが終わる前に外部イベントが発生しても（対応する周期がバーストの後に起きる限り）、再トリガされます。

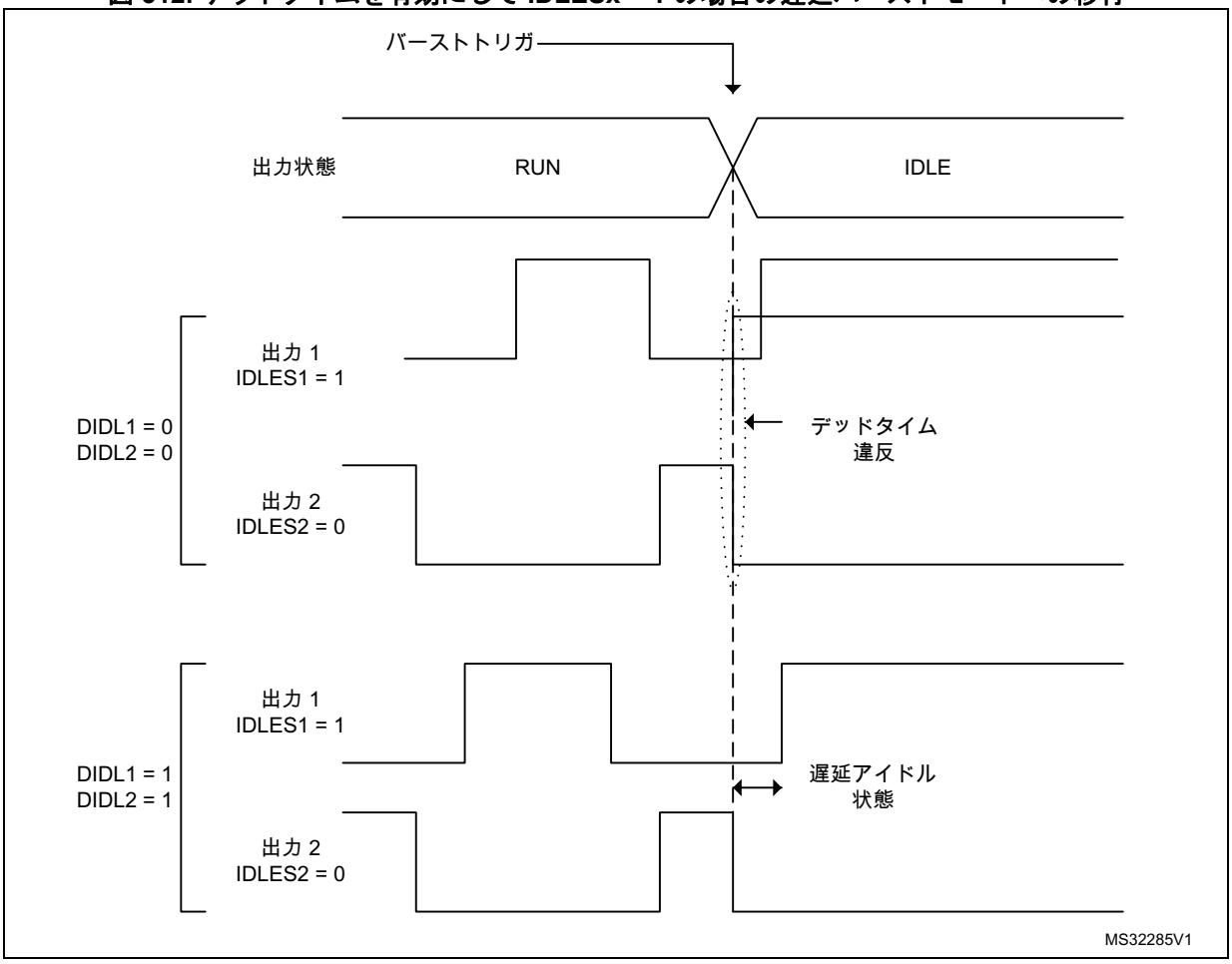
注： ***TAEEV7 および TDEEV8 トリガは周期イベントの後にのみ有効です。周期イベントの前にカウンタがリセットされると、保留中の hrtim_evt7/8 イベントは破棄されます。***

バーストモードの遅延移行

デフォルトにより、バーストモードのトリガ直後、出力は（IDLES1 および IDLES2 の設定により）アイドルレベルになります。

バーストモードへの移行を遅らせて、出力がそのアイドル状態になる前にプログラム可能な期間の間、出力を非アクティブ状態に強制することも可能です。これは、*図 312* に示すように、2 つの相補出力を駆動していて、そのうちの 1 つがアクティブアイドル状態であるとき、デッドタイム違反を避けるのに有用です。これはハーフブリッジでのシュートスルー電流のリスクを防ぎますが、バーストモードへの移行の応答が遅延します。

図 312. デッドタイムを有効にして IDLESx = 1 の場合の遅延バーストモードへの移行



遅延バースト移行モードは HRTIM_OUTxR レジスタの DIDLx ビット（出力あたり 1 つのイネーブルビット）によって有効になります。出力がアイドル状態になる前にデッドタイムの挿入が強制されます。TIMx 出力ごとに、それぞれ独自のデッドタイム値を次のように設定します。

- DIDL1 = 1 のとき、出力 1 で DTRx[8:0]
- DIDL2 = 1 のとき、出力 2 で DTFx[8:0]

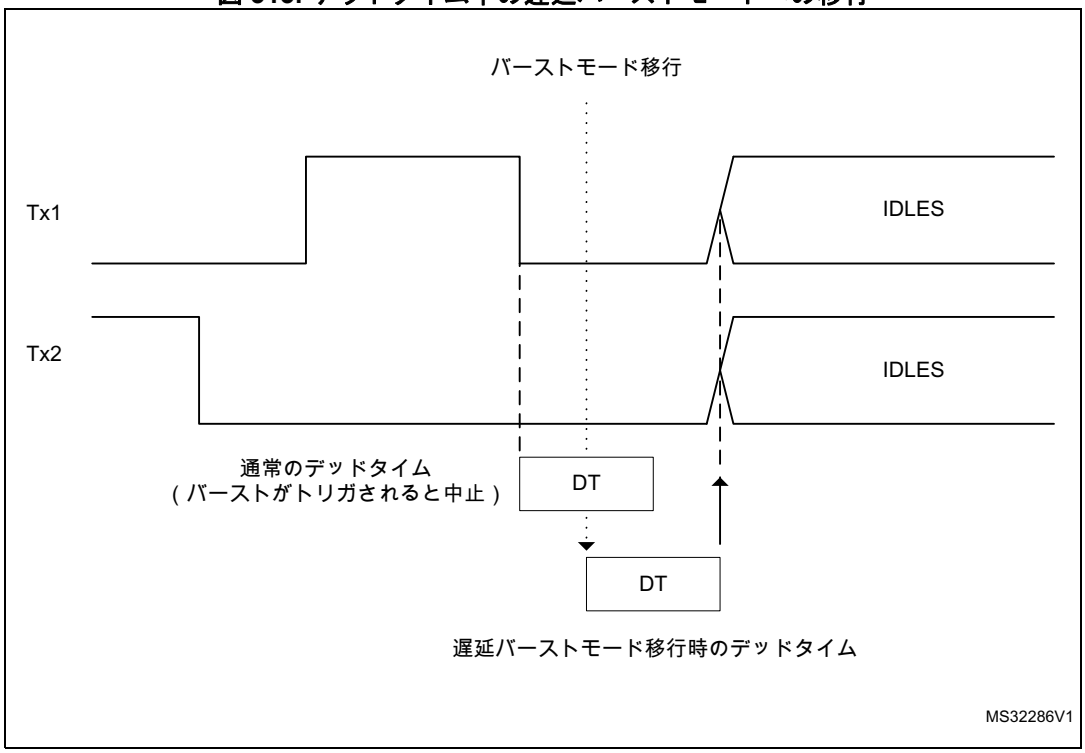
バーストモード中に出力の 1 つがアクティブアイドルレベルとなっている場合（IDLES = 1）のみ、および、正のデッドタイムが使用される（SDTR/SDTF が 0 にセットされている）ときのみ、DIDLx ビットをセットすることができます。

注： ***遅延バースト移行モードはデッドタイムジェネレータリソースを使用します。したがって、2 つの DIDLx ビットのどちらかがセットされ、対応するタイミングユニットがデッドタイム挿入を使用する（HRTIM_OUTxR の DTEN ビットのセット）ときは、タイマ x 出力 2 を外部イベント用のフィルタとして使用することはできません（Tx2 フィルタ信号は利用できません）。***

DTRx[8:0] および DTFx[8:0] によって定義された期間が f_{HRTIM} クロックの 3 サイクル周期より小さいとき、*セクション 37.3.6* に記載されている狭パルス管理に関する制限を適用する必要があります。

バーストモード移行が通常のデッドタイムの間に起きると、デッドタイムは中止され、*図 313* のように、非アクティブ期間に対応して新しいデッドタイムが再び開始されます。

図 313. デッドタイム中の遅延バーストモードへの移行



バーストモードの終了

バーストモードの終了は、ソフトウェアによって強制されるか（連続モード時）、またはアイドル期間が経過したときに起きます（シングルショットモード時）。どちらの場合も、カウンタは直ちに再開始されますが（MTBM または TxBM ビット = 1 によってリセット状態に保持されていた場合）、出力状態のアイドルからアクティブモードへの実効遷移はプログラムされたセット／リセットイベント後にのみ起きます。

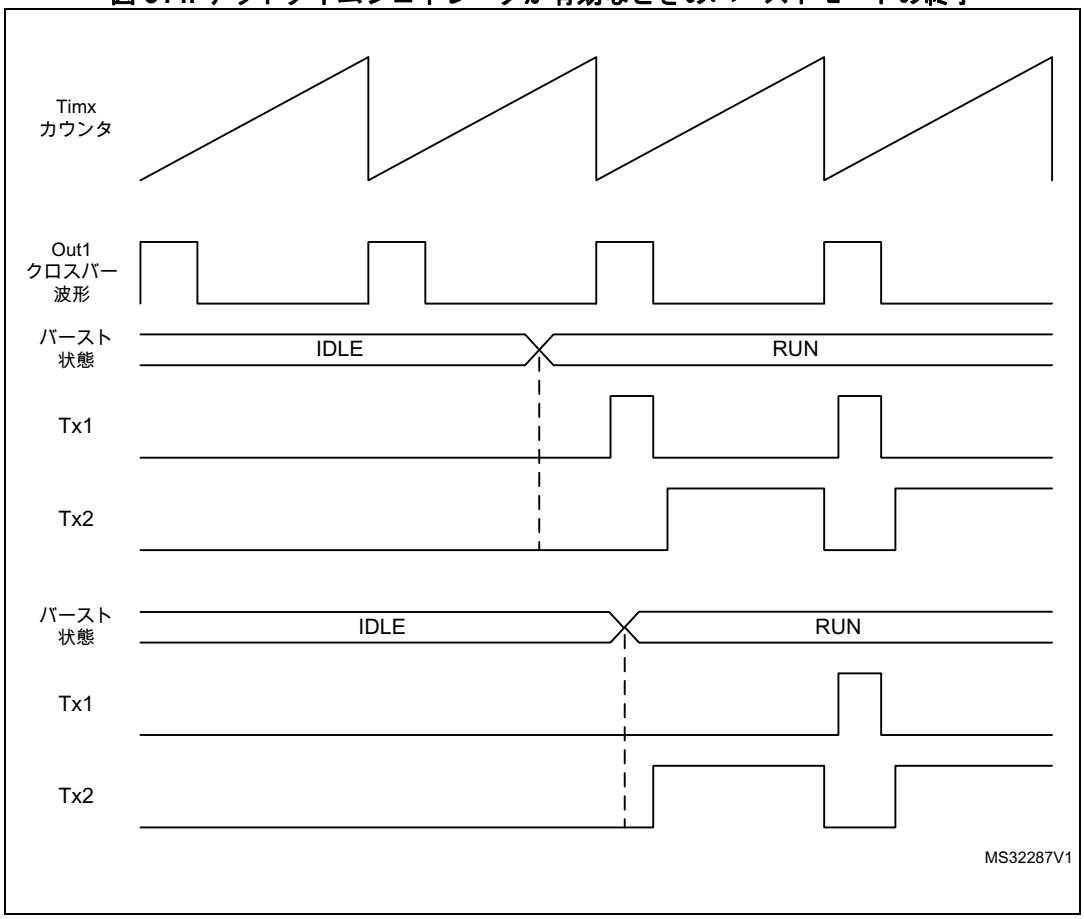
HRTIM_IER レジスタにおいて BMPERIE イネーブルビットがセットされていると、シングルショットおよび連続モードにおいてバースト周期の割り込みが生成されます。この割り込みはバーストモードの終了を連続バーストモードでのバースト周期に同期させるために使用できます。

図 314 に、デッドタイムが有効になっているとき、通常の動作がどのように再開されるかを示します。バーストモードの終了が即時であっても、これは、相補出力のどちらかの最初のセットイベントでのみ有効になります。

次の 2 つの異なるケースを示します。

1. 信号がクロスバー出力波形上で非アクティブの間にバーストモードが終了します。Tx1 出力のセットイベント時に、Tx1 および Tx2 上でアクティブ状態が再開され、Tx2 出力はバースト終了時には相補レベルになりません。
2. クロスバー出力波形がアクティブの間にバーストモードが終了します。動作は Tx2 出力のセットイベント時に再開され、Tx1 はバースト終了時にすぐにはアクティブレベルになりません。

図 314. デッドタイムジェネレータが有効なときのバーストモードの終了



プッシュプルモードが有効なときは、上述の動作は若干異なります。出力が非アクティブの場合、プッシュプルモードは周期の始めに出力を強制的にリセットするか、あるいは、出力が前の周期の間にハイであった場合、アクティブレベルに強制します。

したがって、アクティブアイドル状態の出力は、遷移が明示的にプログラムされていなくても、バーストモードが終了する時点でリセットできます。反対に、出力がアイドル状態に入った時にアクティブであった場合、遷移が明示的にプログラムされていなくても、出力はバーストモードが終了する時点でセットできます。

バーストモードレジスタのプリロードおよび更新

BMPREN ビット（バーストモードプリロードイネーブル）によって、バーストモードの比較および周期レジスタ（HRTIM_BMCMP および HRTIM_BMPER）をプリロードさせることができます。

BMPREN がセットされていると、以下の時、プリロードからアクティブレジスタへの転送が起きます。

- バーストモードが有効になる時（BME = 1）
- バーストモード周期の終了時

レジスタが変更されるときの 2 つのレジスタの一貫性を確保するために、HRTIM_BMPER 周期レジスタへの書き込み時は、HRTIM_BMCMP 比較レジスタが書き込まれるまで、更新が一時的に無効になります。

比較レジスタのみ変更する必要がある場合は、シングル書き込みをする必要があります。周期のみを変更する必要がある場合は、新しい値が考慮されるように比較も再書き込みが必要です。

BMPREN ビットがリセットされているとき、BMCMPR および BMPER へ書き込みアクセスすると、アクティブレジスタが直接、更新されます。この場合、以下の 2 つのケースについて、全体バースト周期中のどの時点で更新するかを考慮する必要があります。

a) 比較レジスタの更新

新しい比較値が現在のバーストモードカウンタ値より大きい場合、新しい比較は現在の周期に考慮されます。

新しい比較値が現在のバーストモードカウンタ値より小さい場合、連続モードでの新しい比較は次のバースト周期で考慮され、シングルショットモードでは無視されます（比較一致は起きず、アイドル周期の終りまでアイドル状態が続きます）。

b) 周期レジスタの更新

新しい周期値が現在のバーストモードカウンタ値より大きい場合、変更は現在の周期に考慮されます。

注： ***新しい周期値が現在のバーストモードカウンタ値より小さい場合、新しい周期は考慮されず、バーストモードカウンタは（0xFFFF で）オーバーフローし、変更は次の周期で有効になります。シングルショットモードでは、カウンタは 0xFFFF でロールオーバーし、バーストモードは新しいプログラム値までの別の周期で再開始します。***

合成レジスタを使用したバーストモードのエミュレーション

バーストモードコントローラは 1 つのコンバータについて 1 つまたは 1 セットのタイマのみを制御します。複数の独立したタイマにバーストモードが必要なとき、DMA と HRTIM_CMP1CxR 合成レジスタ（繰り返しレジスタと比較 1 レジスタの両方のエイリアスを持つ）を使用して単純なバーストモードコントローラをエミュレートすることができます。

これは、デューティサイクルのみ更新する必要がある単純な PWM のみを必要とするコンバータ（降圧コンバータが典型的）に適用できます。この場合、出力をリセット（そしてデューティサイクルを定義）するのに CMP1 レジスタを使用し、セットは周期イベントで行います。

この場合、デューティサイクル（CMP1 値で定義）および、このデューティサイクルが維持される周期の数（繰り返し値で定義）を定義するためには、CMP1CxR へ 32 ビット書き込みアクセスを 1 回するだけで十分です。バーストモードを実装するには、次のように構成した 2 つの 32 ビットデータを DMA によって（繰り返しイベント時に）連続モードで転送することしか必要ありません。

CMPC1xR = {REP_Run; CMP1 = デューティサイクル}、{REP_Idle; CMP1 = 0}

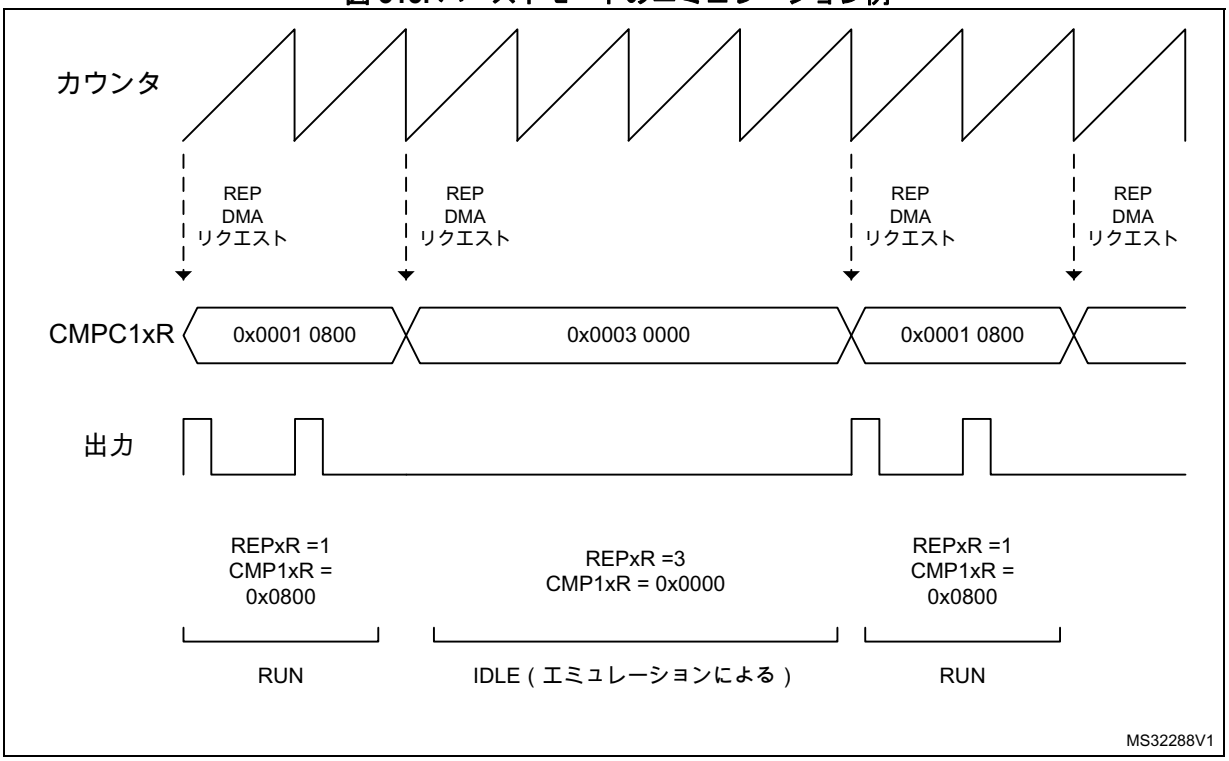
たとえば、次の値：

{0x0003 0000}：3 周期間、CMP1 = 0

{0x0001 0800}：1 周期間、CMP1 = 0x0800

は、*図 315* に示すように、PWM の 6 周期ごとに 2 周期アクティブであるバーストモードを提供します。

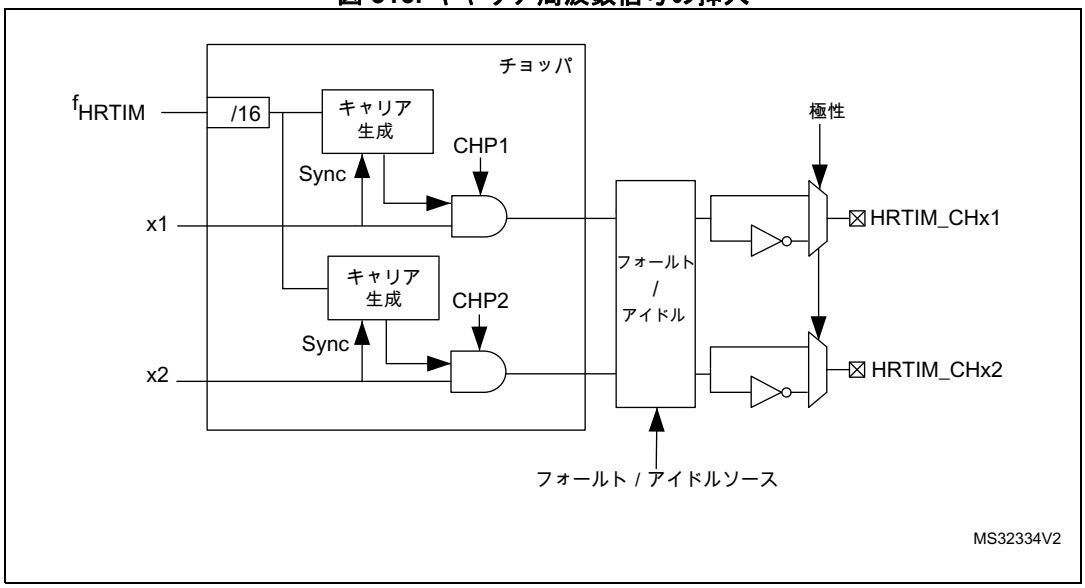
図 315. バーストモードのエミュレーション例



37.3.14 チョッパ

タイミングユニットの出力信号の先頭で高周波キャリアを付加して絶縁型変成器を駆動できます。これは、*図 316* に示すように、出力 1 と 2 それぞれでチョッパを有効にするために、HRTIM_OUTxR レジスタの CHP1 および CHP2 ビットを使用して、出力ステージで極性の挿入の前に行われます

図 316. キャリア周波数信号の挿入



チョッパのパラメータは HRIM_CHPxR レジスタを使用して調節できます。すなわち、*図 317* のように、パルスの始めに特定のパルス幅を定義し、その後に、プログラム可能な周波数とデューティサイクルを持ったキャリア周波数が続くように指定できます。

CARFRQ[3:0] ビットによって、式 $F_{CHPFRQ} = f_{HRTIM} /（16 x（CARFRQ[3:0]+1））$に従って、156 MHz～25 MHz（$f_{HRTIM}$ = 400 MHz に対して）の範囲の周波数が定義されます。

デューティサイクルは、CARDTY[2:0] によって 0/8 から 7/8 デューティサイクルまで 1/8 ステップずつ調節することができます。CARDTY[2:0] = 000（デューティサイクル = 0/8）のとき、出力波形はリファレンス波形の立ち上がりエッジの後に、搬送波のない開始パルスのみを含むものになります。

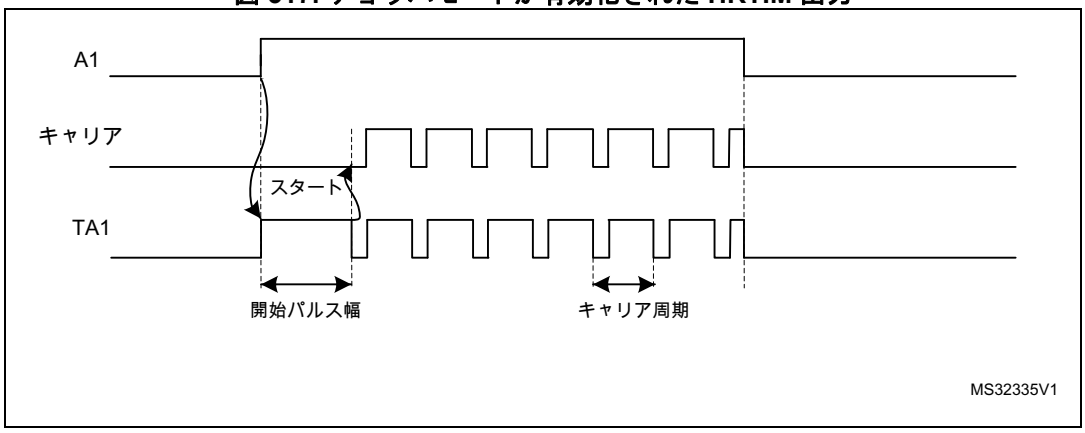
最初のパルスのパルス幅は STRPW[3:0] ビットフィールドを使用して次のように定義されます：t1STPW = （STRPW[3:0]+1）x 16 x t_{HRTIM} であり、その範囲は 40 ns～0.63 µs（f_{HRTIM}=400 MHz に対して）です。

キャリア周波数のパラメータは f_{HRTIM} 周波数に基づいて定義されており、CKPSC[2:0] の設定には依存しません。

チョッパモードでは、キャリア周波数と最初のパルス幅は、AND 機能を使用してリファレンス波形と結合されます。同期は最初のパルスの終りに行われ、繰り返し信号の形になります。

チョッピング信号は、現在のキャリア周期が完了するまで待たずに、出力波形のアクティブ状態の終りに停止します。よって、プログラムされたパルス幅より短いパルスが含まれることがあります。

図 317. チョッパモードが有効化された HRTIM 出力



注： ***CHP1 および CHP2 ビットは、HRTIM_OENR レジスタの TxyOEN ビットで出力を有効にする前にセットする必要があります。***

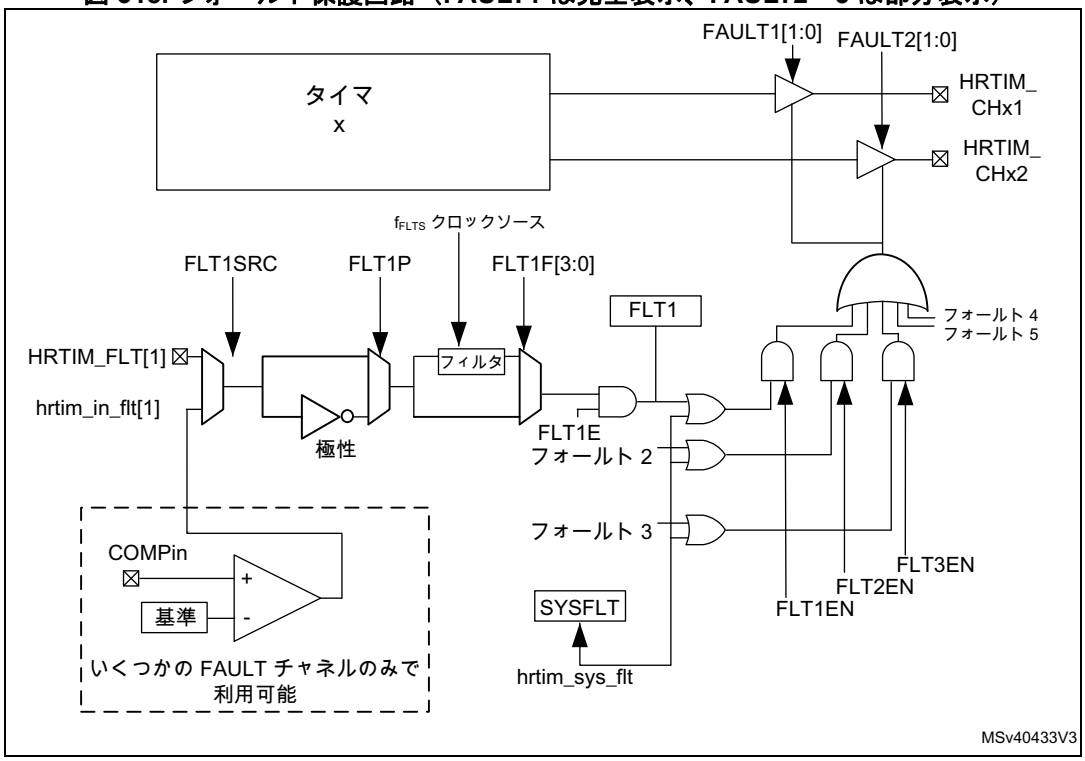
CARFRQ[2:0]、CARDTY[2:0]、および STRPW[3:0] ビットフィールドは、チョッパモードがアクティブである（2 つの CHPx ビットの少なくとも 1 つがセットされている）間は変更できません。

37.3.15 フォールト保護

HRTIMER は、多目的なフォールト保護回路を備えており、異常な動作の場合に出力を無効にします。フォールトがトリガされると、出力は事前に定義された安全状態になります。この状態は出力がソフトウェアによって再度有効にされるまで維持されます。永続的フォールトのリクエストの場合は、出力は、ソフトウェアが再度有効化しようとしても、フォールトソースが消えるまで、フォールト状態に留まります。

HRTIM には 5 つの FAULT 入力チャネルがあり、*図 318* に示すように、それらすべては使用可能で、5 つのタイミングユニットのそれぞれに組み合わせられます。

図 318. フォールト保護回路（FAULT1 は完全表示、FAULT2～5 は部分表示）



各フォールトチャネルは、タイミングユニットに接続される前に HRTIM_FLTINR1 および HRTIM_FLTINR2 レジスタを使用して完全に設定可能です。FLTxSRC ビットによってフォールト信号のソースを選択します。ソースはデジタル入力のことも、内部イベント（内蔵コンパレータの出力）のこともあります。

表 301 に、10 個のフォールトチャネルの各々に対して利用可能なソースをまとめます。

表 301. フォールト入力

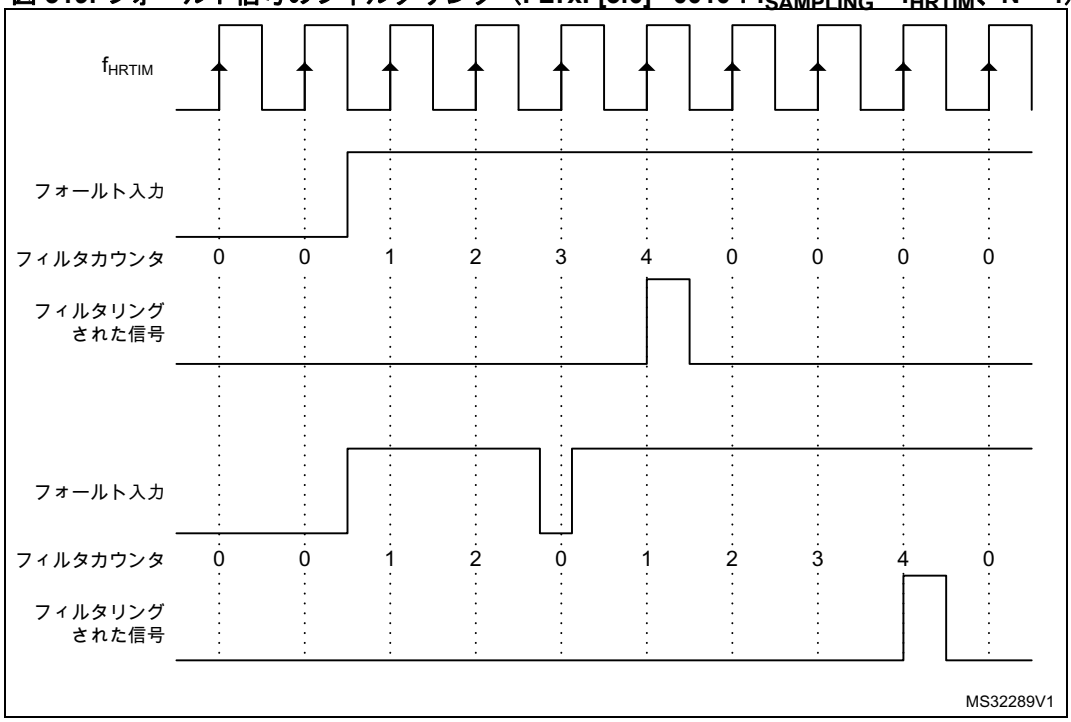
フォールトチャネル	外部入力（FLTxSRC = 0）	オンチップソース（FLTxSRC = 1）
FAULT 1	PA15	COMP1
FAULT 2	PC11	COMP2
FAULT 3	PD4	NC
FAULT 4	PB3	NC
FAULT 5	PG10	NC

HRTIM_FLTINRx レジスタの FLTxP 極性ビットを使用して、アクティブレベルを定義するために信号の極性を選択できます。FLTxP = 0 の場合は、信号はローレベルでアクティブであり、FLTxP = 1 の場合は、ハイレベルでアクティブです。

フォールト情報は極性の設定の後、フィルタリングできます。FLTxF[3:0] ビットフィールドを 0000 にセットした場合は、信号はフィルタリングされず、f_{HRTIM} クロックと独立に非同期で動作します。その他の FLTxF[3:0] ビットフィールド値すべてに対しては、信号はデジタルフィルタリングされます。デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。カウンタが N の値に達する前に入力値が変化した場合、カウンタはリセットされ、変化は破棄されます（誤ったイベントとみなされます）。カウンタが N に達した場合、変化は有効とみなされ、正しい外部イベントとして転送されます。したがって、デジタルフィルタは、サンプリン

グクロックとフィルタ長（予想される有効なサンプルの数）に応じて、フィルタリングされる外部イベントに遅延時間を付加します。*図 319* に、誤ったフォールト信号がどのようにフィルタリングされるかを示します。

図 319. フォールト信号のフィルタリング（FLTxF[3:0]= 0010：$f_{SAMPLING} = f_{HRTIM}$、N = 4）



フィルタ周期は f_{HRTIM} クロックの 2 サイクルから f_{FLTS} クロックの 8 サイクルを 32 で割った値までの範囲になります。f_{FLTS} は HRTIM_FLTINR2 レジスタの FLTSD[1:0] ビットを使用して定義されます。*表 302* に、サンプリングレートとフィルタ長をまとめます。サンプリングによる不確かさを考慮し、有効なフィルタリングを行うために、1 サンプリングクロック周期のジッタをフィルタ長から差し引く必要があります。

表 302. サンプリングレートおよびフィルタ長 対 FLTFxF[3:0] およびクロック設定

	f_{FLTS} 対 FLTSD[1:0]				f_{HRTIM}= 400 MHz に対するフィルタ長	
FLTFxF[3:0]	00	01	10	11	最小値	最大値
0001、0010、0011	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}、N =2 5 ns	f_{HRTIM}、N =8 20 ns
0100、0101	f_{HRTIM} /2	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /2、N =6 30 ns	f_{HRTIM} /16、N =8 320 ns
0110、0111	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /4、N =6 60 ns	f_{HRTIM} /32、N =8 640 ns
1000、1001	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /8、N =6 120 ns	f_{HRTIM} /64、N =8 1.28 µs
1010、1011、1100	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /16、N =5 200 ns	f_{HRTIM} /128、N =8 2.56 µs
1101、1110、1111	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /256	f_{HRTIM} /32、N =5 400 ns	f_{HRTIM} /256、N =8 5.12 µs

システムフォールト入力（hrtim_sys_flt）

この障害（フォールト）は MCU のクラス B 回路（詳細はシステム構成コントローラ（SYSCFG）セクションを参照）によって提供され、下記から着信するシステムフォールトに相当します。

- クロックセキュリティシステム
- SRAM パリティチェッカ
- Cortex®-M7-ロックアップ信号
- PVD 検出回路

この入力は FAULT 入力より優先され、FAULTy[1:0] = 01、10、11 であるすべての出力を無効にします。

各 FAULT チャネルに対して、機能安全の目的で HRTIM_FLTxR レジスタの 1 回のみ書き込み可能の FLTxLCK ビットによって、FLTxE、FLTxP、FLTxSRC、FLTxF[3:0] ビットをロックすることができます（これらを読み出し専用にします）。有効になっていた場合は、フォールト調整の設定は、次の HRTIM リセットまたはシステムリセットまで維持されます。

上で説明したように、フォールト信号が調整されると、それはタイミングユニット（複数）に配信されます。タイミングユニットのいずれに対しても、HRTIM_FLTxR レジスタの FLT1EN～FLT5EN ビットを使用して 5 つのフォールトチャネルが有効化され、同時に選択可能になります（システムフォールトは出力がフォールトメカニズムによって保護されている限り、自動的に有効になります）。これにより、たとえば以下のことが可能になります。

- 1 つのフォールトチャネルが同時に複数のタイミングユニットを無効にする。
- 複数のフォールトチャネルの論理和がとられて、1 つのタイミングユニットを無効にする。

機能安全の目的で、HRTIM_FLTxR レジスタの 1 回のみ書き込み可能の FLTLCK ビットによって、次のリセットまで FLTxEN ビットをロックすることができます（これらを読み出し専用にします）。有効になっていた場合は、タイミングユニットのフォールトに関連する設定は、次の HRTIM リセットまたはシステムリセットまで維持されます。

タイマごとに、フォールト中の出力状態は HRTIM_OUTxR レジスタの FAULT1[1:0] および FAULT2[1:0] ビットによって定義されます（*セクション 37.3.12* を参照）。

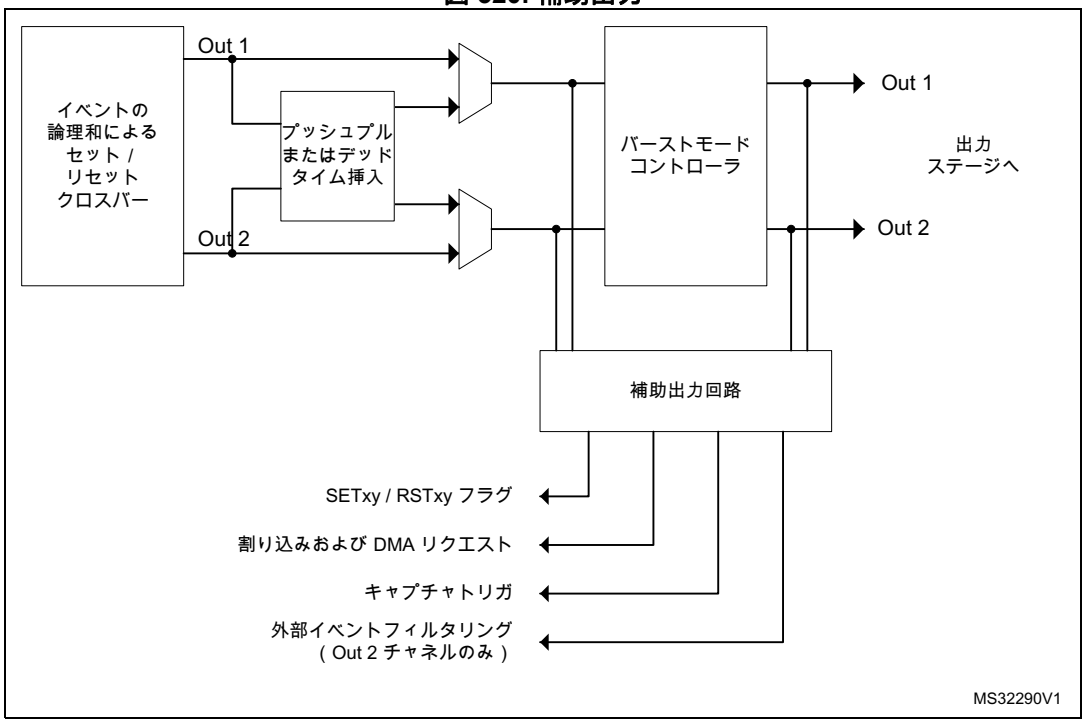
37.3.16 補助出力

タイマ A～E には、出力ステージへ行く通常の出力と並行して補助出力が備わっています。それらは以下の内部ステータス、イベント、および信号を提供します。

- SETxy および RSTxy ステータスフラグ、および対応する割り込みおよび DMA リクエスト
- 出力セット／リセット時のキャプチャトリガ
- Tx2 出力コピー後の外部イベントフィルタ（*セクション 37.3.8* の詳細を参照）。

補助出力は、HRTIM 動作モードに応じてバーストモードコントローラの前または後で取られます。概要を *図 320* に示します。

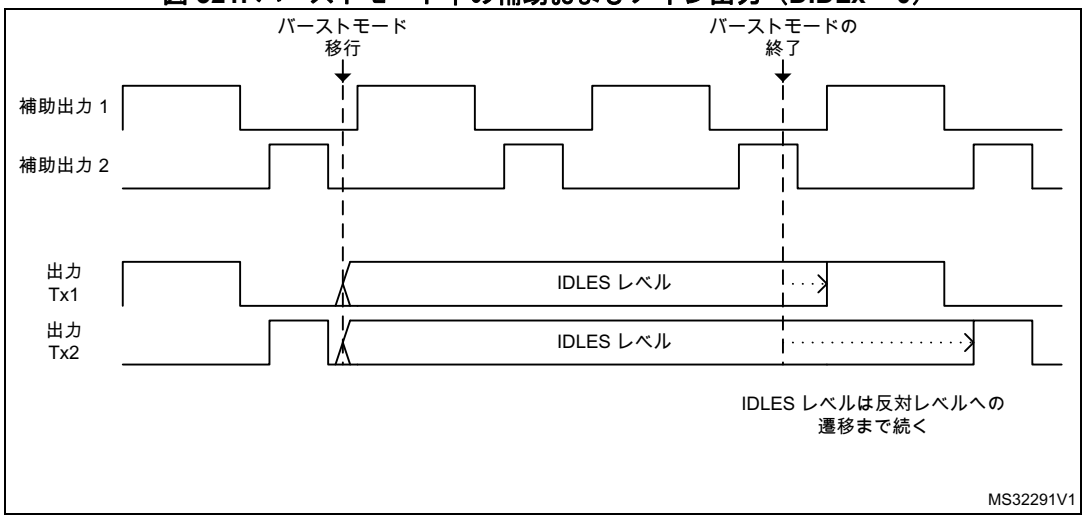
図 320. 補助出力



デフォルトでは、補助出力は出力 Tx1 と Tx2 のコピーです。ただし、例外は以下のとおりです。

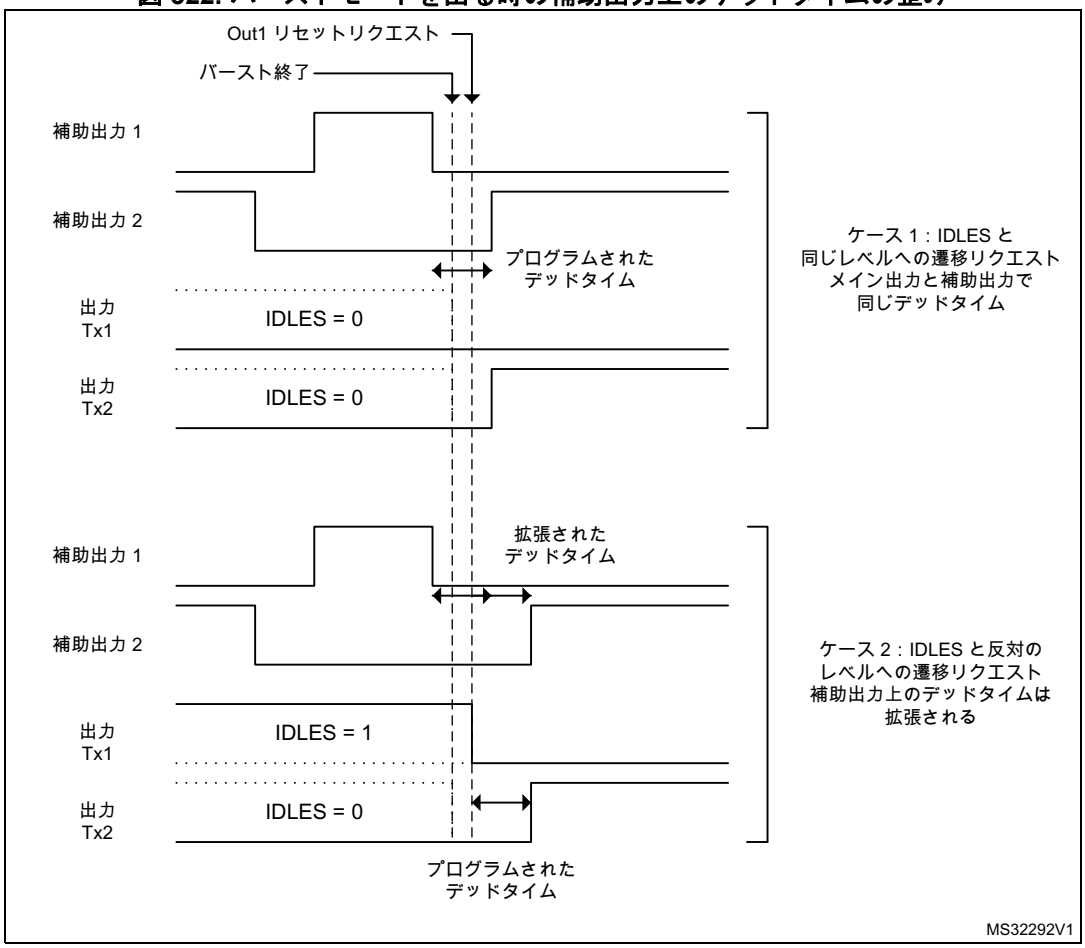
- デッドタイムが無効（DTEN = 0）になっているときの、遅延アイドルおよびバランスアイドル保護。保護がトリガされると、補助出力は維持され、クロスバーから来る信号に従います。反対に、デッドタイムが有効（DTEN = 1）になっている場合は、メイン出力と補助出力の両方とも非アクティブレベルに強制されます。
- バーストモード（TCEN=1、IDLEMx=1）。次の 2 つの場合があります。
  a) DTEN=0 または DIDLx=0 の場合、補助出力はバーストモードへの移行に影響されず、クロスバーから来るリファレンス信号に従い続けます（*図 321* 参照）。
  b) 遅延バーストモードへの移行（DIDLx=1）と共にデッドタイムが有効（DTEN = 1）になっている場合は、補助出力はメイン出力と同じ動作になります。デッドタイム期間の後、それらは IDLES レベルに強制され、すべてのバースト周期の間、このレベルを保持します。バーストモードが終了すると、メイン出力と同様に、IDLES レベルは反対レベルへの遷移が起きるまで維持されます。

図 321. バーストモード中の補助およびメイン出力（DIDLx = 0）



バーストモードから出る時、または遅延保護の後に出力を再有効化する時、これがデッドタイム中に起きた場合には、補助出力上の信号はわずかに歪むことがあります。この場合、メイン出力上のデッドタイムが尊重されるように、補助出力に適用されるデッドタイムは拡張されます。*図 322* に、いくつかの例を示します。

図 322. バーストモードを出る時の補助出力上のデッドタイムの歪み



37.3.17 他のタイマまたは HRTIM インスタンスとの HRTIM の同期

HRTIM は、複数の HRTIM インスタンスをマスタユニット（同期信号を生成）として、またはスレーブ（同期されるトリガを待機）として、同期させるオプションを提供しています。この機能は、HRTIM を外部またはオンチップの他のタイマと同期させるためにも使用することができます。同期回路はマスタタイマの内部で制御されます。

同期出力

このセクションでは、外部リソースを同期させてマスタユニットとして動作するためには、HRTIM をどのように設定する必要があるかを説明します。

同期出力に転送されるソースとして 4 つのイベントを選択できます。これは、HRTIM_MCR レジスタの SYNCSRC[1:0] ビットを使用して、以下のように行います。

- 00：マスタタイマの開始
  このイベントが生成されるのは、MCEN ビットがセットされる時か、またはタイマがシングルショットモードで周期値に達した後に再開始される時です。カウント中に起きるリセット時（CONT または RETRIG ビットがセットされているとき）にも生成されます。
- 01：マスタタイマの比較 1 イベント
- 10：タイマ A の開始
  このイベントが生成されるのは、TACEN ビットがセットされる時か、またはカウンタがリセットされて、このリセットに応答してカウントを再開始する時です。連続モードでのカウンタロールオーバー、およびシングルショットの再トリガ不可能モードで破棄されたリセットリクエストといった、カウンタリセットイベントは同期出力には伝播されません。リセットは、カウント中に起きたとき（CONT または RETRIG ビットがセットされているとき）のみ考慮されます。
- 11：タイマ A の比較 1 イベント

HRTIM_MCR レジスタの SYNCOUT[1:0] ビットは、同期イベントがどのように生成されるかを指定します。

SYNCOUT[1:0] = 1x によって、同期パルスが HRTIM_SCOUT 出力ピンに生成されます。SYNCOUT[0] ビットは、同期信号の極性を指定します。SYNCOUT[0] = 0 の場合、HRTIM_SCOUT ピンはローアイドルレベルとなり、同期のために f_{HRTIM} クロックの 16 サイクル長の正のパルスを発行します。SYNCOUT[0] = 1 の場合、アイドルレベルはハイとなり、負のパルスが生成されます。

注： ***同期パルスの後に、f_{HRTIM} クロックの 16 サイクル長のアイドルレベルが続き、その間は、新しい同期リクエストは破棄されます。したがって、最大の同期周波数は、$f_{HRTIM}/32$ となります。***

HRTIM_SCOUT ピンのアイドルレベルは、SYNCOUT[1:0] ビットが有効になる（つまり、ビットフィールドの値が 00 と異なる）とすぐに適用されます。

同期出力の初期化手順は、MCU 出力の設定およびカウンタ有効化の前に、以下の順に行う必要があります。

1. HRTIM_MCR の SYNCOUT[1:0] および SYNCSRC[1:0] ビットフィールドの設定
2. HRTIM_SCOUT ピンの設定（汎用 I/O のセクションを参照）
3. マスタまたはタイマ A のカウンタの有効化（MCEN または TACEN ビットのセット）

同期入力モードが有効になっていて、同期出力モード（SYNCSRC[1:0] = 00 または 10）と同時に（SYNCSTRTM/SYNCSTRTx ビットを使用して）カウンタを開始すると、カウンタが開始している時または実行中にリセットされた時にのみ出力パルスが生成されます。カウンタの開始を引き起こさずにそれをクリアするリセットリクエストは同期出力には影響しません。

同期入力

HRTIM_MCR レジスタの下記の SYNCIN[1:0] ビットにより、HRTIM は外部ソースによって同期させることができます。

- 00：同期入力は無効
- 01：予約済みの設定
- 10：オンチップの TIM1 汎用タイマ（TIM1 TRGO 出力）
- 11：HRTIM_SCIN 入力ピン上の正のパルス

このビットフィールドは、転送先のタイマ（マスタタイマまたはタイミングユニット）が有効化（MCEN および／または TxCEN ビットのセット）された後は、変更できません。

HRTIM_SCIN 入力は立ち上がりエッジ対応です。タイマの動作は、HRTIM_MCR および HRTIM_TIMxCR レジスタにある以下のビットによって定義されます（詳細は*表 303* を参照）。

- 同期開始：到着信号がタイマのカウンタを開始させます（SYNCSTRTM および／または SYNCSTRTx ビットのセット）。タイマを有効にしてカウンタを開始準備させるためには、TxCEN（MCEN）ビットをセットする必要があります。連続モードでは、同期信号を受け取るまではカウンタは開始しません。
- 同期リセット：到着信号がカウンタをリセットします（SYNCRSTM および／または SYNCRSTx ビットのセット）。このイベントは他のリセットイベントと同様に繰り返しカウンタをデクリメントします。

同期イベントは、関連カウンタが有効化（MCEN または TxCEN ビットをセット）されてからのみ、考慮されます。同期リクエストは SYNC 割り込みをトリガします。

注： ***現在のカウンタ値がアクティブな周期値を超えている場合は、同期開始イベントはカウンタをリセットします。***

同期イベントの効果は、*表 303* にまとめているように、タイマの動作モードに依存します。

表 303. 同期イベントの効果 対 動作モード

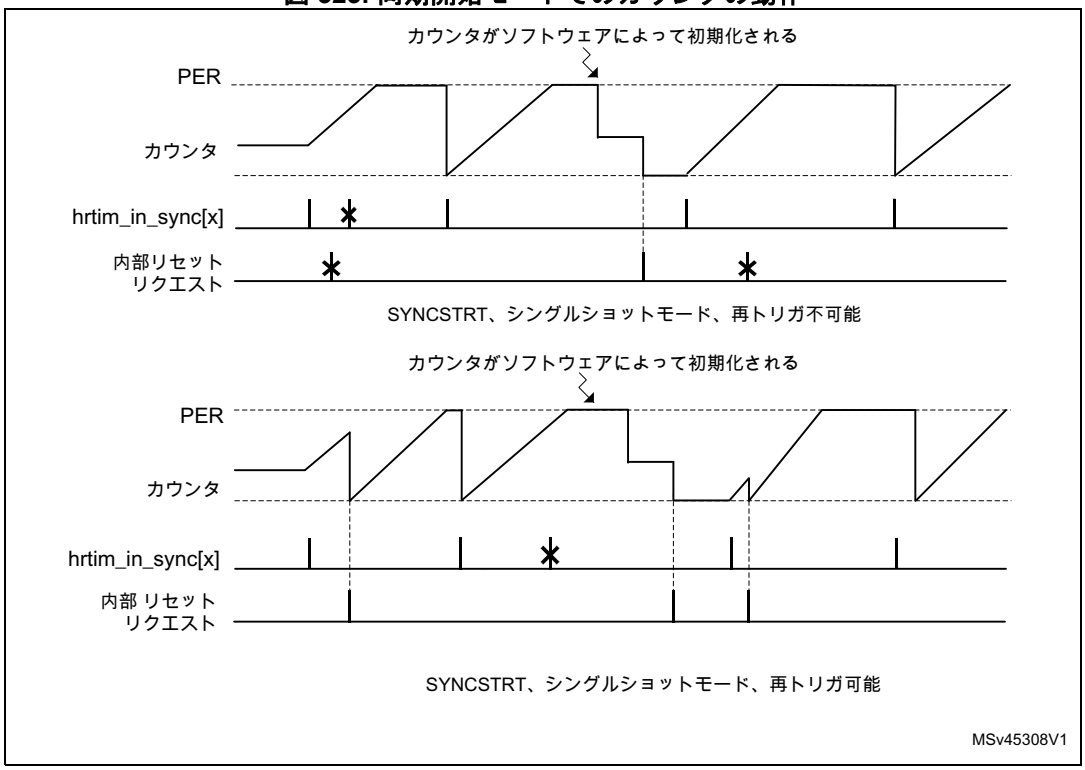
動作モード	SYNC RSTx	SYNC STRTx	SYNC リセットまたは開始イベントの後の動作
シングルショット再トリガ不可能	0	1	開始イベントは、カウンタが停止していて、かつ、以下の場合に、考慮されます。 – MCEN または TxCEN ビットがセットされた後 – 周期に達した後 カウンタが周期値で停止しているときに起きる開始はカウンタをリセットします。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません（カウンタは同期によってのみ、再開始できます）。カウント中に起きるリセットは無視されます（通常の再トリガ不可能モード中と同様）。
	1	X	リセットイベントはタイマカウントを開始します。カウンタが停止していて、かつ、以下の場合にのみ、考慮されます。 – MCEN または TxCEN ビットがセットされた後 – 周期に達した後 複数のリセットリクエストが選択されたとき（HRTIM_SCIN から、および内部イベントから）、最初に到着したリクエストのみが考慮されます。

表 303. 同期イベントの効果 対 動作モード（続き）

動作モード	SYNC RSTx	SYNC STRTx	SYNC リセットまたは開始イベントの後の動作
シングルショット再トリガ可能	0	1	カウンタ開始は、カウンタが開始していない場合、または周期が経過している場合にのみ、有効です。カウンタ開始後に起きる同期イベントは効果がありません。 カウンタが周期値で停止しているときに起きる開始はカウンタをリセットします。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません（カウンタは同期によってのみ、開始できます）。カウント中に起きるリセットは考慮されます（通常の再トリガ可能モード中と同様）。
	1	X	HRTIM_SCIN からのリセットは、内部イベントからの HRTIM カウンタのリセットと同様に、考慮され、タイマのカウントを開始または再開始します。 複数のリセットリクエストが選択されたとき、最初に到着したリクエストが考慮されます。
連続モード	0	1	タイマが有効化され（MCEN または TxCEN のビットセット）、カウンタを開始するための同期イベントを待ちます。カウンタ開始後に起きる同期イベントは効果がありません（カウンタは同期によってのみ、開始できます）。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません。
	1	X	HRTIM_SCIN からのリセットは、内部イベントからの HRTIM カウンタのリセットと同様に、考慮され、タイマのカウントを開始または再開始します。複数のリセットリクエストが選択されたとき、最初に到着したリクエストが考慮されます。

図 323 に、シングルショットモードで同期開始がどのように行われるかを示します。

図 323. 同期開始モードでのカウンタの動作



37.3.18 ADC トリガ

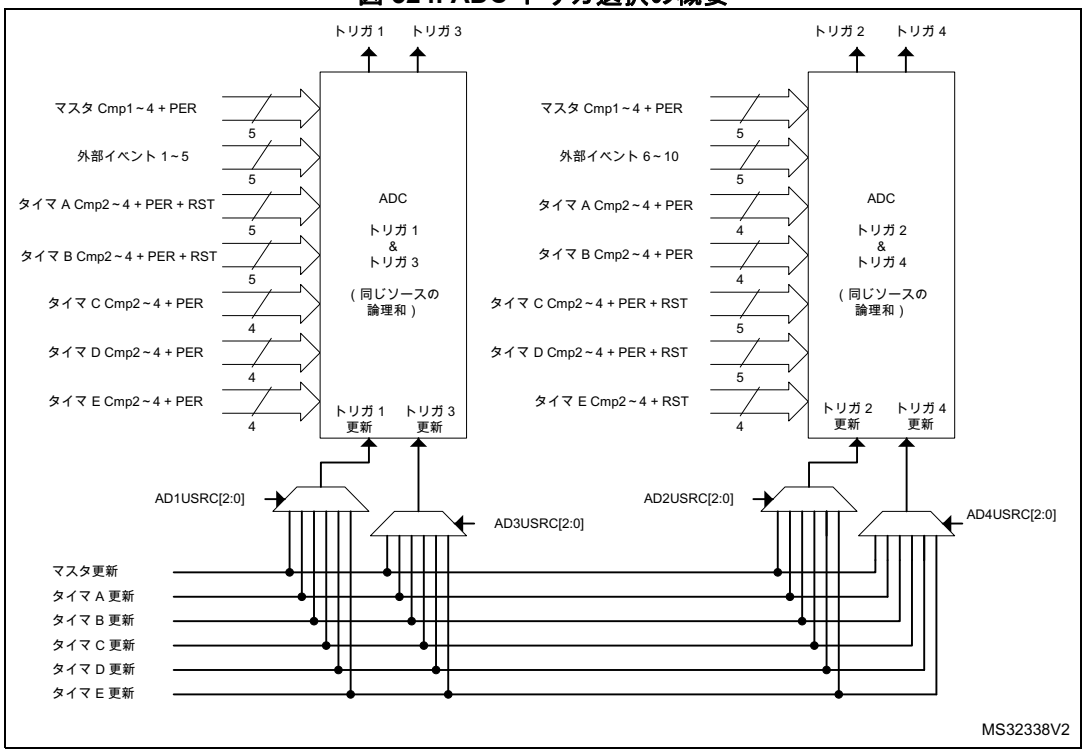
ADC は、マスタおよび 5 つのタイミングユニットによってトリガできます。

2 つの ADC のレギュラおよびインジェクトシーケンスの両方を開始するために、4 つの独立したトリガが利用できます。*図 324* に示すように、HRTIM_ADC1R～HRTIM_ADC4R レジスタにおいて、各トリガ出力に対して最大 32 のイベントを組み合わせる（論理和）ことができます。トリガ 1/3 およびトリガ 2/4 は同じソースセットを使用します。

外部イベントをトリガとして使用できます。それらは、HRTIM_EECRx レジスタで定義された調整の直後に取り込まれ、EEFxR1 および EEFxR2 レジスタの設定に依存しません。

複数のソースを同時に選択することによって、1 つのスイッチング周期内で複数のトリガをかけることができます。典型的な使用事例は、単一の ADC トリガ出力を使用してすべての位相を連続的にサンプルできる、非オーバーラップ多相コンバータです。

図 324. ADC トリガ選択の概要



HRTIM_ADC1R～HRTIM_ADC4R レジスタはプリロードされ、関連するタイマと同期して更新できます。更新ソースは、HRTIM_CR1 レジスタの ADxUSRC[2:0] ビットによって定義されます。

たとえば、ADC トリガ 1 がタイマ A CMP2 イベントを出力する場合（HRTIM_ADC1R = 0x0000 0400）、HRTIM_ADC1R は一般的にタイマ A（AD1USRC[2:0] = 001）と同時に更新されます。

ソースタイマでプリロードが無効になっている場合（PREEN ビットのリセット）、HRTIM_ADCxR レジスタもプリロードされません。書き込みアクセスをするとトリガソースはすぐに更新されます。

37.3.19 DAC トリガ

HRTIMER によって、内蔵 DAC をタイマの更新と同期して更新させることができます。

マスタタイマおよびタイマユニットからの更新イベントは、3 つの hrtim_dac_trgx 出力のいずれにも DAC 更新トリガを生成できます。

注： ***各タイマには DAC 関連の独自の制御レジスタがあります。***

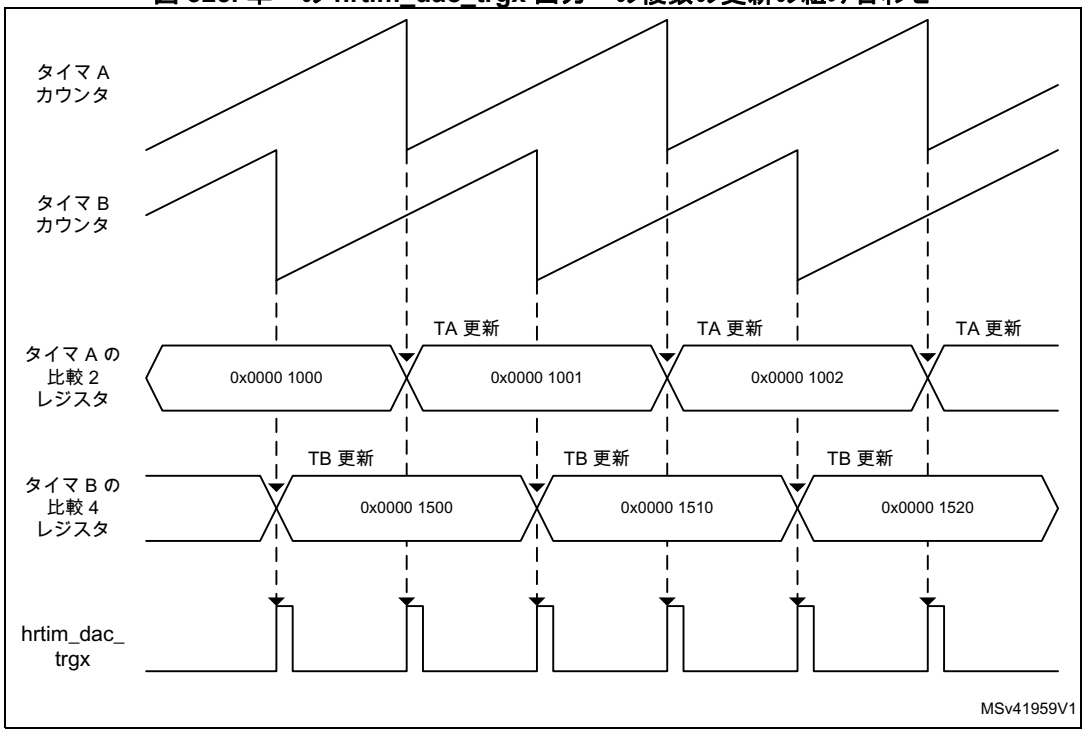
HRTIM_MCR および HRTIM_TIMxCR レジスタの DACSYNC[1:0] ビットは次のようにプログラムされます。

- 00：更新は生成されません。
- 01：更新が hrtim_dac_trg1 に生成されます。
- 10：更新が hrtim_dac_trg2 に生成されます。
- 11：更新が hrtim_dac_trg3 に生成されます。

f_{HRTIM} クロックの 1 周期の出力パルスが hrtim_dac_trgx 出力に生成されます。

DACSYNC[1:0] ビットが複数のタイマで有効にされると、hrtim_dac_trgx 出力はすべてのタイマの更新イベントの論理和から構成されます。たとえば、タイマ A とタイマ B で DACSYNC = 1 とすると、*図 325* に示すように、タイマ A での更新イベントはタイマ B での更新イベントと論理和がとられ、対応する hrtim_dac_trgx 出力に DAC 更新トリガを生成します。

図 325. 単一の hrtim_dac_trgx 出力への複数の更新の組み合わせ



Hrtim_dac_trgx ピンは以下のように DAC に接続されています。

- hrtim_dac_trg1：DAC1_CH1 トリガ入力 9（DAC1 ペリフェラルの DAC_CR で TSEL1[2:0] = 1001）
- hrtim_dac_trg2：DAC1_CH2 トリガ入力 10（DAC1 ペリフェラルの DAC_CR で TSEL1[2:0] = 1010）
- hrtim_dac_trg3：接続されていません

参考資料

37.3.20 HRTIM 割り込み

以下の 7 つの割り込みがマスタタイマによって生成できます。

- マスタタイマのレジスタ更新
- 受信した同期イベント
- マスタタイマの繰り返しイベント
- マスタ比較 1～4 イベント

以下の 14 個の割り込みが各タイミングユニットによって生成できます。

- トリガされた遅延保護
- カウンタのリセットまたはロールオーバーイベント
- 出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）
- 出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）
- 比較 1 および 2 のイベント
- タイミングユニットのレジスタ更新
- 繰り返しイベント
- 比較 1～4 イベント

HRTIM 全体に対して、8 つのグローバル割り込みが生成されます。

- システム障害およびフォールト 1～5（タイミングユニットの帰属にかかわらず）
- バーストモード周期完了

割り込みリクエストは以下のように 7 つのベクトルにグループ分けされます。

- hrtim_mst_it：マスタタイマ割り込み（マスタ更新、同期入力、繰り返し、MCMP1～4）、およびフォールトを除くグローバル割り込み（バーストモード周期）
- hrtim_tima_it：TIMA 割り込み
- hrtim_timb_it：TIMB 割り込み
- hrtim_timc_it：TIMC 割り込み
- hrtim_timd_it：TIMD 割り込み
- hrtim_time_it：TIME 割り込み
- hrtim_fault_it：高優先順位割り込み処理を可能にするすべてのフォールト割り込みの専用ベクタ

*表 304*に、割り込みリクエスト、そのマッピングと関連制御およびステータスビットをまとめます。

表 304. HRTIM 割り込みの概要

割り込みベクタ	割り込みイベント	イベントフラグ	イネーブル制御ビット	フラグクリアビット
hrtim_mst_it	バーストモード周期完了	BMPER	BMPERIE	BMPERC
	マスタタイマのレジスタ更新	MUPD	MUPDIE	MUPDC
	受信した同期イベント	SYNC	SYNCIE	SYNCC
	マスタタイマの繰り返しイベント	MREP	MREPIE	MREPC
	マスタ比較 1～4 イベント	MCMP1	MCMP1IE	MCP1C
		MCMP2	MCMP2IE	MCP2C
		MCMP3	MCMP3IE	MCP3C
		MCMP4	MCMP4IE	MCP4C

表 304. HRTIM 割り込みの概要（続き）

割り込みベクタ	割り込みイベント	イベントフラグ	イネーブル制御ビット	フラグクリアビット
hrtim_tima_it hrtim_timb_it hrtim_timc_it hrtim_timd_it hrtim_time_it	トリガされた遅延保護	DLYPRT	DLYPRTIE	DLYPRTC
	カウンタのリセットまたはロールオーバーイベント	RST	RSTIE	RSTC
	出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）	RSTx1	RSTx1IE	RSTx1C
		RSTx2	RSTx2IE	RSTx2C
	出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）	SETx1	SETx1IE	SETx1C
		SETx2	SETx2IE	SETx2C
	比較 1 および 2 のイベント	CPT1	CPT1IE	CPT1C
		CPT2	CPT2IE	CPT2C
	タイミングユニットのレジスタ更新	UPD	UPDIE	UPDC
	繰り返しイベント	REP	REPIE	REPC
	比較 1～4 イベント	CMP1	CMP1IE	CMP1C
		CMP2	CMP2IE	CMP2C
		CMP3	CMP3IE	CMP3C
		CMP4	CMP4IE	CMP4C
hrtim_fault_it	システムフォールト	SYSFLT	SYSFLTIE	SYSFLTC
	フォールト 1～5	FLT1	FLT1IE	FLT1C
		FLT2	FLT2IE	FLT2C
		FLT3	FLT3IE	FLT3C
		FLT4	FLT4IE	FLT4C
		FLT5	FLT5IE	FLT5C

37.3.21 DMA

割り込みを生成可能なほとんどのイベントは DMA リクエストも、しかも両方同時に、生成することができます。各タイマ（マスタ、TIMA～E）には独自の DMA イネーブルレジスタがあります。

個々の DMA リクエストは論理和がとられて以下のように 6 つのチャネルに入ります。

- マスタタイマ用に 1 チャネル
- タイミングユニットごとに 1 チャネル

注： ***DMA チャネルを無効に（TIMxDIER で DMA イネーブルビットをリセット）する前に、まず DMA コントローラを無効にする必要があります。***

表 305 は、イベントおよびそれと関連する DMA イネーブルビットの概要です。

表 305. HRTIM の DMA リクエストの概要

DMA チャネル	イベント	DMA 可能	DMA 有効
hrtim_dma1（マスタタイマ）	バーストモード周期完了	いいえ	N/A
	マスタタイマのレジスタ更新	はい	MUPDDE
	受信した同期イベント	はい	SYNCDE
	マスタタイマの繰り返しイベント	はい	MREPDE
	マスタ比較 1～4 イベント	はい	MCMP1DE
		はい	MCMP2DE
		はい	MCMP3DE
		はい	MCMP4DE
hrtim_dma2（タイマ A） hrtim_dma3（タイマ B） hrtim_dma4（タイマ C） hrtim_dma5（タイマ D） hrtim_dma6（タイマ E）	トリガされた遅延保護	はい	DLYPRTDE
	カウンタのリセットまたはロールオーバーイベント	はい	RSTDE
	出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）	はい	RSTx1DE
		はい	RSTx2DE
	出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）	はい	SETx1DE
		はい	SETx2DE
	比較 1 および 2 のイベント	はい	CPT1DE
		はい	CPT2DE
	タイミングユニットのレジスタ更新	はい	UPDDE
	繰り返しイベント	はい	REPDE
	比較 1～4 イベント	はい	CMP1DE
		はい	CMP2DE
		はい	CMP3DE
		はい	CMP4DE
N/A	システム障害	いいえ	N/A
	フォールト 1～5	いいえ	N/A
	バーストモード周期完了	いいえ	N/A

バースト DMA 転送

標準の DMA リクエストに加え、HRTIM は、1 つの DMA リクエストによって複数のレジスタを更新する DMA バーストコントローラを備えています。これによって次のことが可能です。

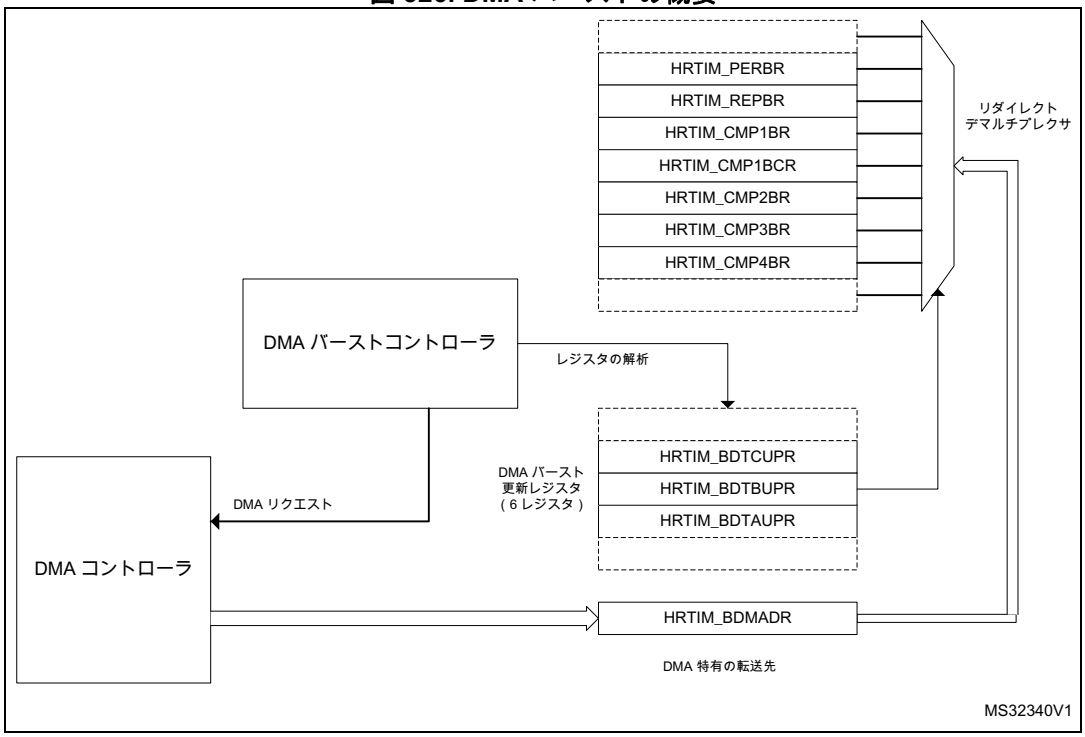
- 1 つの DMA チャネルのみで複数データレジスタを更新する。
- 複数のタイマ出力を使用するコンバータのために、1 つ以上のタイミングユニットを動的にプログラムする。

バースト DMA 機能は 1 つの DMA チャネルのみで使用可能ですが、6 つのチャネルのどれでもバースト DMA 転送用に選択できます。

原則は、どのレジスタが DMA によって書き込まれるかをプログラムすることです。マスタタイマおよび TIMA～E にはバースト DMA 更新レジスタがあり、そこで大部分の制御およびデータレジスタが次の選択ビットで関連付けられています：HRTIM_BDMUPR、HRTIM_BDTAUPR～HRTIM_BDTEUPR（こ

れは書き込みアクセス可能なレジスタのみに適用できます）。*図 326* に示すように、リダイレクトメカニズムによって、DMA 書き込みアクセスを HRTIM レジスタに自動的に転送することができます。

図 326. DMA バーストの概要



DMA トリガが発生すると、HRTIM は複数の 32 ビットの DMA リクエストを生成し、更新レジスタを解析します。制御ビットがセットされている場合、書き込みアクセスは関連するレジスタにリダイレクトされます。ビットがリセットされている場合、レジスタ更新はスキップされ、新しいリクエストをトリガするためにレジスタの解析は新しいビットのセットが検出されるまで再開されます。6 つの更新レジスタ（HRTIM_BDMUPR、5 x HRTIM_BDTxUPR）の解析が終わると、バーストは完了し、システムは別の DMA トリガの受け入れが可能になります（*図 327* のフローチャート参照）。

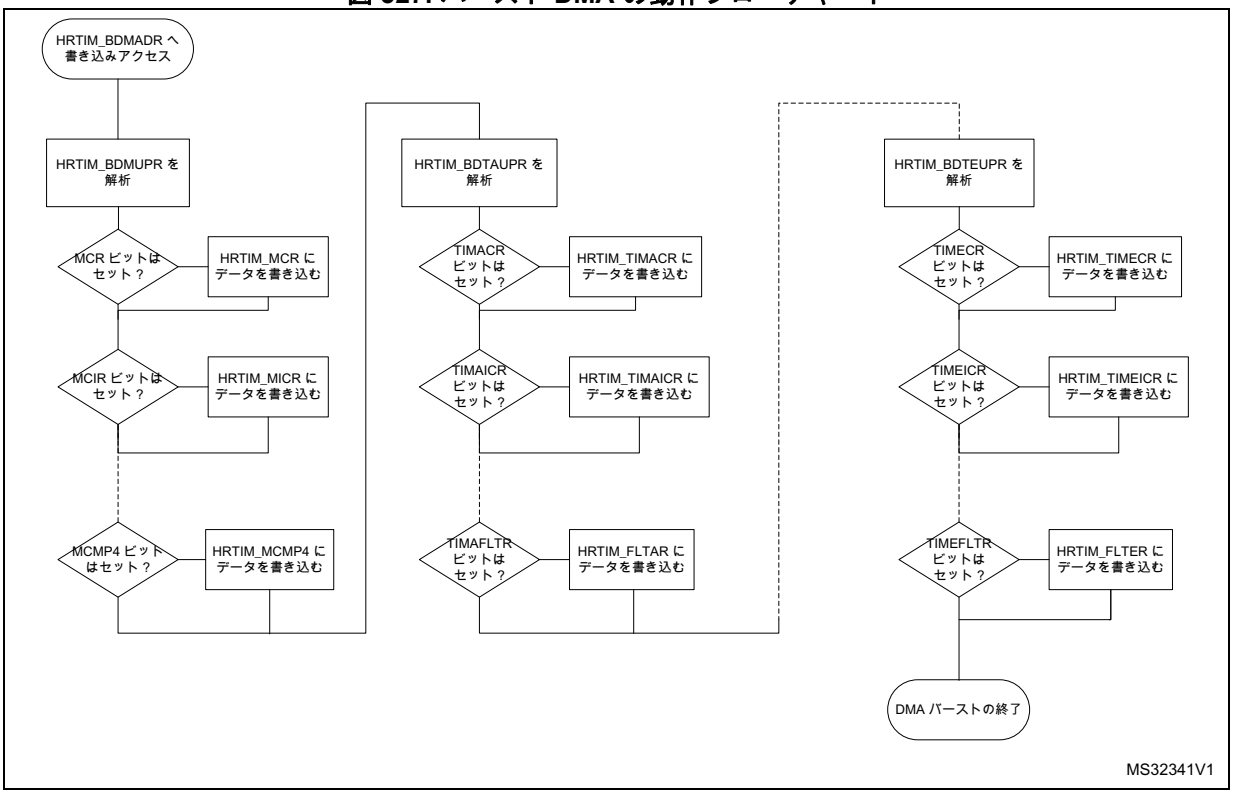
注： ***バーストが進行中に起きるトリガは、一番最後のデータ転送中に起きた場合を除き、破棄されます。***

バースト DMA モードは永続的に有効です（イネーブルビットはありません）。HRTIM_BDMADR レジスタへの初回書き込みアクセスで、バースト DMA 動作が開始します。

必要なことは、ペリフェラルアドレスのインクリメントモードを無効にしてメモリ上のペリフェラル設定を HRTIM_BDMADR レジスタへ転送するようにDMAコントローラを設定するだけです（HRTIM が最終転送先レジスタへデータを送る処理を内部で行います）。

バースト DMA モードがトランザクション中に中断された場合に、それを再初期化するには、6 つの更新レジスタのうち少なくとも 1 つに書き込む必要があります。

図 327. バースト DMA の動作フローチャート



DMA バーストが完了した後、レジスタの更新方法に応じて、いくつかのオプションが利用可能です。

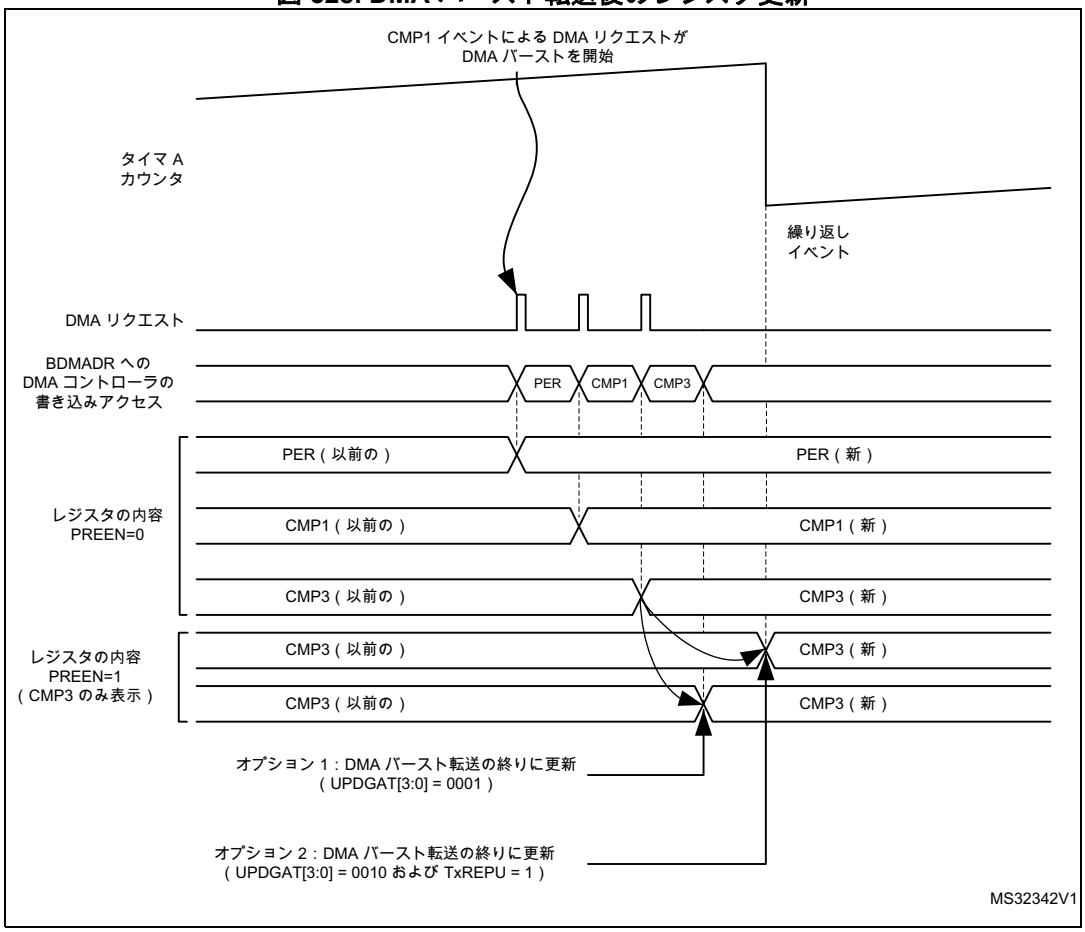
PREEN ビットがリセット（プリロードは無効）されている場合、DMA によって書き込まれた値は直ちにアクティブレジスタへ転送され、レジスタは、DMA のトランザクションのペースに従って、順次更新されます。

プリロードが有効になっている（PREEN ビットのセット）ときは、以下の 3 つの使用事例があります。

1. 更新は DMA バースト転送とは独立に行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0000 および HRTIM_MCR の BRSTDMA[1:0] = 00）。この場合、転送されたすべてのデータを同時に考慮させる必要がある場合、ユーザは、更新イベントが起きる前に DMA バーストが完了していることをチェックする必要があります。反対に、DMA 転送が進行中に更新イベントが起きた場合、レジスタの一部のみがロードされ、完全にレジスタを更新するには 2 つの連続する更新イベントが必要です。
2. 更新は DMA バースト転送が完了したときに行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0000 および HRTIM_MCR の BRSTDMA[1:0] = 01）。このモードでは、すべての新しいレジスタ値が同時に転送されることが保証されます。これはカウンタ値とは独立に行われ、必要なら、通常の更新イベントと組み合わせることができます（たとえば、TxRSTU がセットされているとき、カウンタリセット時に更新）。
3. 更新は DMA バースト転送が完了した後、更新イベント時に行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0010 および HRTIM_MCR の BRSTDMA[1:0] = 10）。このモードでは、転送されたすべてのデータの首尾一貫した更新、およびタイマカウンタを使用した、通常の更新イベントとの同期の両方が保証されます。この場合、転送が進行中に通常の更新リクエストが起きた場合、それは破棄され、次に来る更新リクエスト時に有効な更新が起きます。

図 328 の時間経過図に、次の 3 つのケースに対するアクティブレジスタの内容を示します：PREEN=0、UPDGAT[3:0] = 0001 、および UPDGAT[3:0] = 0001（PREEN = 1 のとき）。

図 328. DMA バースト転送後のレジスタ更新



37.3.22 HRTIM の初期化

このセクションでは、MCU の他の関連ペリフェラルも含めて、HRTIM の推奨の初期化手順について説明します。

HRTIM クロックソースは、リセットおよびクロック制御ユニット（RCC）において有効にする必要があります。

HRTIM 制御レジスタは電力コンバータの接続形態やタイミングユニットの使用事例に従って初期化することができます。すべての入力を設定する必要があります（ソース、極性、エッジ対応）。

最終的には次の順序で HRTIM 出力を設定する必要があります。

- 極性は HRTIM_OUTxR の POLx ビットを使用して定義する必要があります。
- FAULT および IDLE 状態は HRTIM_OUTxR の FAULTx[1:0] および IDLESx ビットを使用して設定する必要があります。

HRTIM 出力が MCU の I/O 類に接続可能となります。GPIO コントローラにおいて、選択された HRTIM の I/O 類を製品データシートのオルタネート機能配置表によって設定する必要があります。

この時点から、HRTIM は IDLE 状態にある出力を制御します。

出力は、HRTIM_OENR レジスタの TxyOEN ビットをセットすることによって、RUN モードに設定されます。2 つの出力は、RUN モードでの最初の有効なセット／リセットイベントまで、非アクティブ状態です。TxCEN ビットがリセットされている限り、いかなる出力セット／リセットイベントも

（SST、SRT を使用したソフトウェアリクエストを除き）無視されます。バーストモードリクエストも同じく無視されます（IDLEM ビットの値は無視されます）。同様に、バーストモードコントローラから来るカウンタリセットリクエストも無視されます（TxBM ビットがセットされている場合）。

注： ***デッドタイム挿入が有効（DTEN ビットのセット）になっているとき、RUN モードに入るとすぐに出力が相補状態になるように、SST および RST ビットを使用して、ソフトウェアによって出力状態を強制する必要があります。***

HRTIM 動作は、最終的に HRTIM_MCR の TxCEN または MCEN ビットをセットすることによって開始することができます。

HRTIM ペリフェラルが、リセットおよびクロックコントローラによってリセットされた場合、HRTIM 出力はローレベルで IDLE モードになります。ペリフェラルのリセットを実施する前に、最初に（GPIO コントローラを使用して）HRTIMER を出力から切り離すことをお勧めします。

37.3.23 デバッグ

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBG モジュールの下記 DBG_HRTIM_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

- DBG_HRTIM_STOP = 0：動作の変更はなく、HRTIM は動作を続けます。
- DBG_HRTIM_STOP = 1：マスタタイマを含め、すべての HRTIM タイマは停止します。RUN モードの出力は、FAULTx[1:0] = 01、10、11 の場合、FAULT 状態になるか、または FAULTx[1:0] = 00 の場合、現在の状態を保ちます。アイドル状態の出力はその状態を維持します。これは、MCU が停止モードを出た場合も、永続的に維持されます。これによって、ステップ実行中に安全な状態を維持することができます。出力は、TxyOEN ビットをセットすることによって（デバッガの使用を要求）、再び有効化できます。

DBG_HRTIM_STOP = 1 のとき、MCU 停止中のタイマ動作

セット／リセットクロスバー、デッドタイムおよびプッシュプルユニット、アイドル／バランス障害検出、および RUN モードで通常の出力を駆動しているすべてのロジックはデバッグの影響を受けません。出力は、（MCU の停止中またはその後）TxyOEN が再びセットされる時に出力の通常信号を回復できるように、内部的にトグルを維持します。出力が無効になっているとき、関連するトリガおよびフィルタも内部波形に従っています。

FAULT 入力およびイベント（どのソースも）は、MCU の停止中、有効になっています。

MCU の停止中に、フォールトがその時に発生した場合、フォールトステータスビットはセットでき、TxyOEN ビットはリセットできます（TxyOEN および TxyODS は DBG_HRTIM_STOP ビット状態の影響を受けません）。

キャプチャイベントのみならず、同期、カウンタリセット、開始、およびリセット開始イベントは、デバッグモードでは破棄されます。これは、MCU が停止している限り、関連するすべてのレジスタを安定に保つためです。

ブレークポイントに達するとカウンタはカウントを停止します。しかし、カウンタイネーブル信号はリセットされません。したがって、デバッグから出た時、開始イベントは発行されません。外部イベントのみならず、すべてのカウンタリセットおよびキャプチャトリガは無効になります（MCU が停止している限り、無視されます）。強制ソフトウェアセット／リセットの場合を除き、出力 SET および RST フラグは凍結されます。レベル対応のイベントは、デバッグ中はマスクされますが、デバッグが終了するとすぐに、再びアクティブになります。エッジ対応のイベントに対しては、MCU が停止中に信号がアクティブに維持される場合は、デバッグから出る時、新しいエッジは生成されません。

更新イベントは破棄されます。これによって、hrtim_upd_en[3:1] 入力への更新トリガが防止されます。DMA トリガは無効になります。バーストモード回路は停止します：トリガは無視され、バーストモードカウンタは停止します。

37.4 アプリケーションの使用事例

37.4.1 降圧コンバータ

降圧コンバータはステップダウンコンバータとして一般的に使用されます。HRTIM は 6 つの独立なスイッチング周波数を使って最大 10 個の降圧コンバータを制御できます。

コンバータは通常、固定の周波数で動作し、Vin/Vout 比は次のように電源スイッチに適用されるデューティサイクル D に依存します。

$$V_{out} = D \times V_{in}$$

接続形態は*図 329* に示すとおりで、電圧の読みのために ADC へ接続されています。

図 329. 降圧コンバータの接続形態

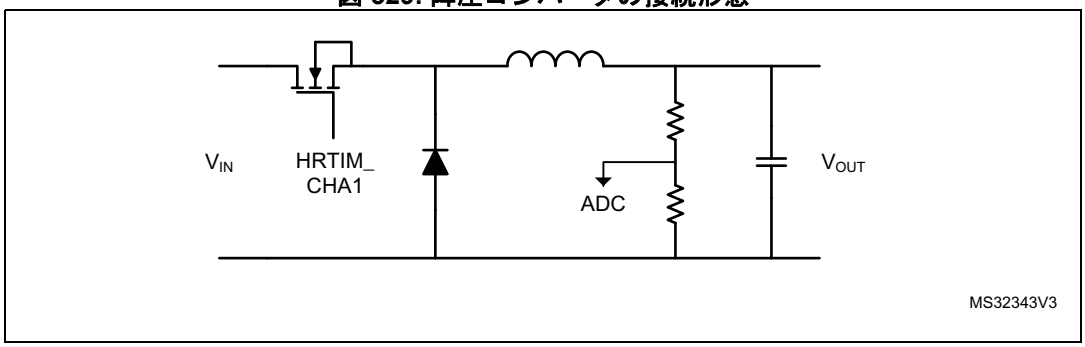
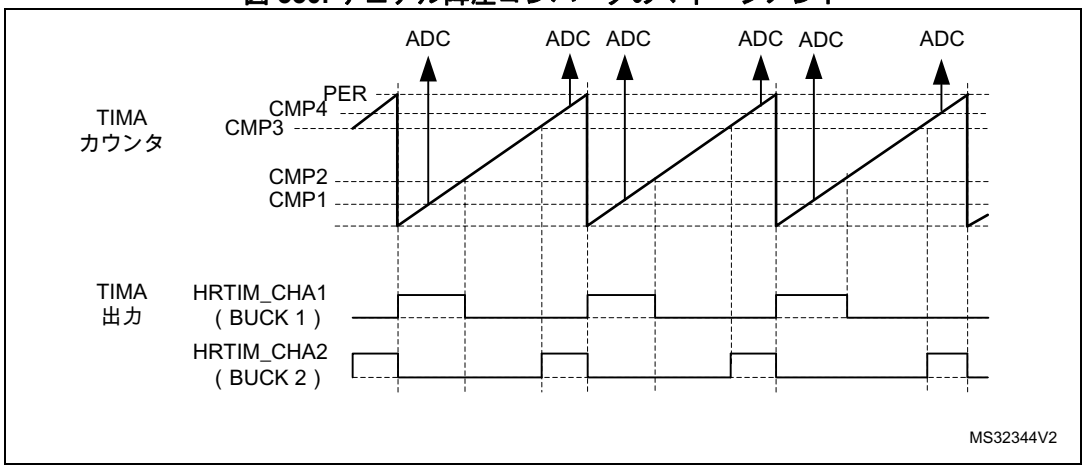


図 330 に、同じ周波数の PWM 信号を使用した 2 つのコンバータのマネージメントを示します。出力は、次のように定義されています。

- HRTIM_CHA1 は周期でセット、CMP1 でリセット
- HRTIM_CHA2 は CMP3 でセット、PER でリセット

CMP2 および CMP4 イベントを使用して、ADC は、正確に ON タイムの中央で周期あたり 2 回トリガされています。

図 330. デュアル降圧コンバータのマネージメント

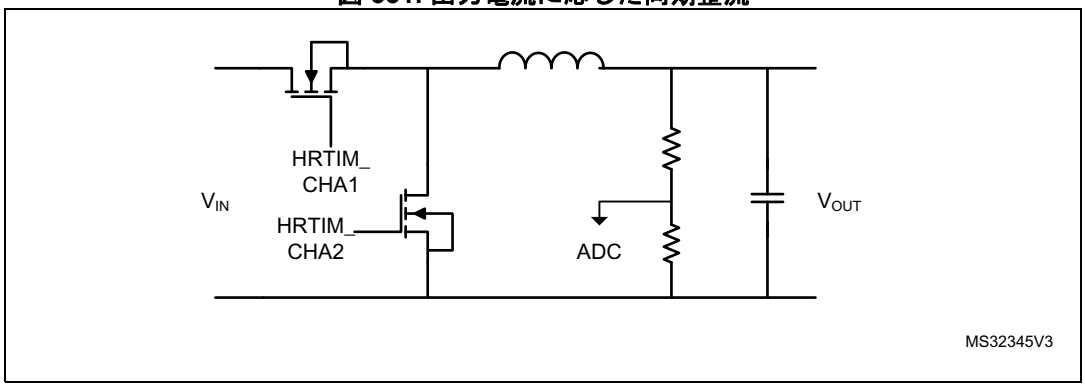


タイマ A～E は、ペア（両方とも同じスイッチング周波数）に結合された 10 個の降圧コンバータ、または、マスタタイマを 6 番目のタイムベースとして使用して 6 個の完全に独立したコンバータ（それぞれが異なるスイッチング周波数を持つ）を提供できます。

37.4.2 同期整流式の降圧コンバータ

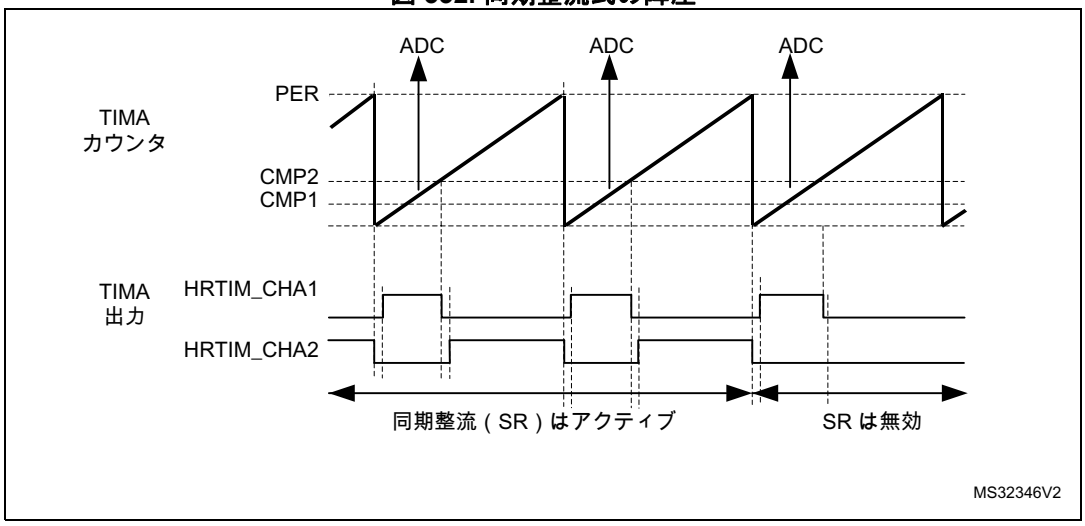
同期整流によって、フリーホイーリングダイオードに代わる FET を使って降圧コンバータの損失を最小にすることができます。同期整流は、*図 331* に示すように、出力の電流レベルに応じて臨機応変にオンまたはオフにできます。

図 331. 出力電流に応じた同期整流



シングルスイッチの降圧コンバータに対する主な違いは、HRTIM_CHA1 の基準波形に基づいて HRTIM_CHA2 上にほとんど相補的な波形を生成するためにデッドタイムを付加することです（*図 332* 参照）。

図 332. 同期整流式の降圧



37.4.3 多相コンバータ

多相技術は複数の電力変換接続形態（降圧、フライバック）に適用できます。これらの主要な利点は以下のとおりです。

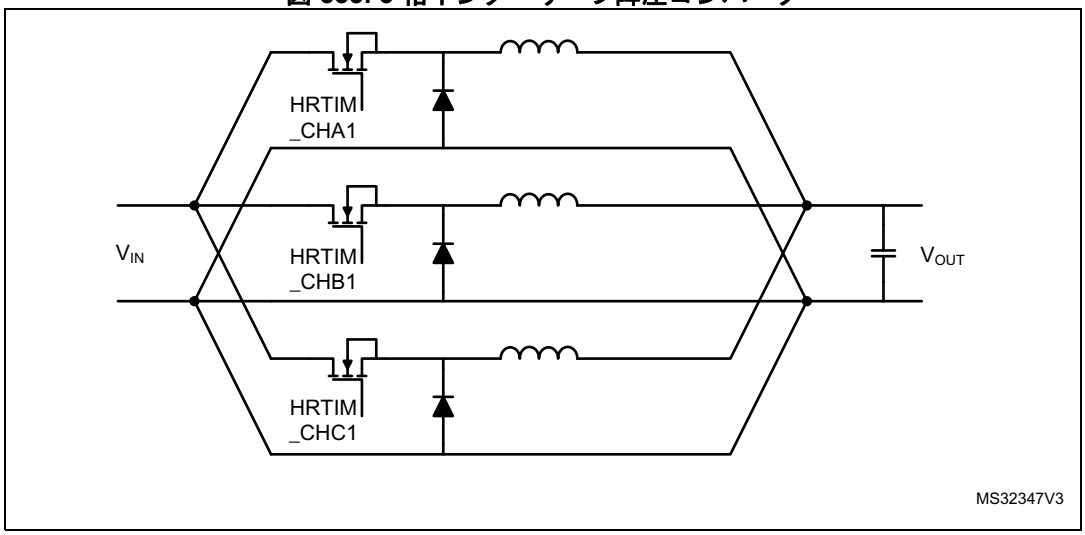
- 入力および出力コンデンサの電流リップルの削減
- EMI の低減
- 位相の数を動的に変えること（位相シェディング）による軽負荷で高効率

HRTIM は複数のコンバータを管理できます。制御できるコンバータの数は、次のように、接続形態と使用するリソース（ADC トリガを含む）に依存します。

- マスタタイマと 5 つのタイマを使用する 5 つの同期整流（SR）式の降圧コンバータ
- マスタタイマと 2 つのタイマを使用する 4 つの降圧コンバータ（SR なし）
- ...

図 334 は、3 相インターリーブ降圧コンバータの接続形態を示します。

図 333. 3 相インターリーブ降圧コンバータ



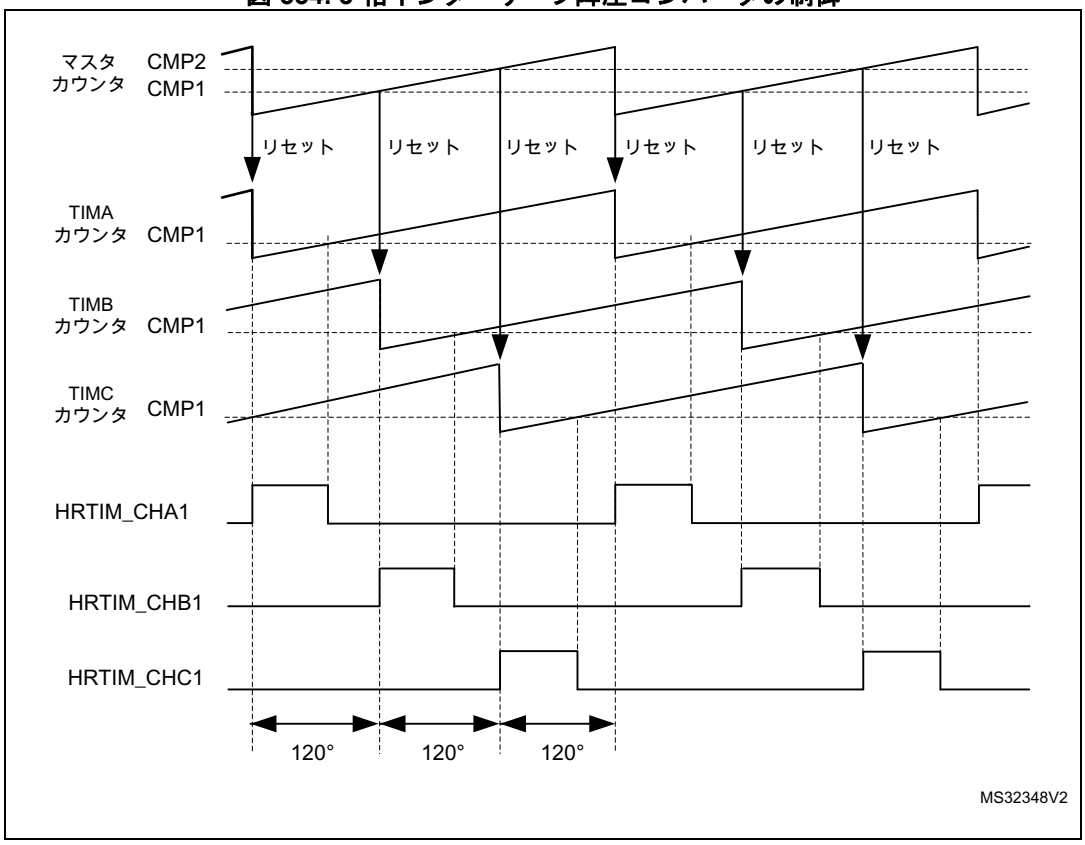
マスタタイマは位相管理を受け持ちます。タイマを周期的にリセットすることによってコンバータ間の位相関係を定義します。位相シフトは 360°を位相の数で割った値で、この例では 120°です。

デューティサイクルは各タイマにプログラムされます。出力は、次のように定義されています。

- HRTIM_CHA1 はマスタタイマの周期でセット、TACMP1 でリセット
- HRTIM_CHB1 はマスタタイマの MCMP1 でセット、TBCMP1 でリセット
- HRTIM_CHC1 はマスタタイマの MCMP2 でセット、TCCMP1 でリセット

ADC トリガは TxCMP2 比較イベントによって生成できます。コンバータの接続形態によりすべての ADC トリガソースは位相がずれているため、それらすべてを 1 つの ADC トリガに結合して ADC リソースを節約できます（たとえば、完全マルチフェーズコンバータに対して 1 つの ADC 通常チャネル）。

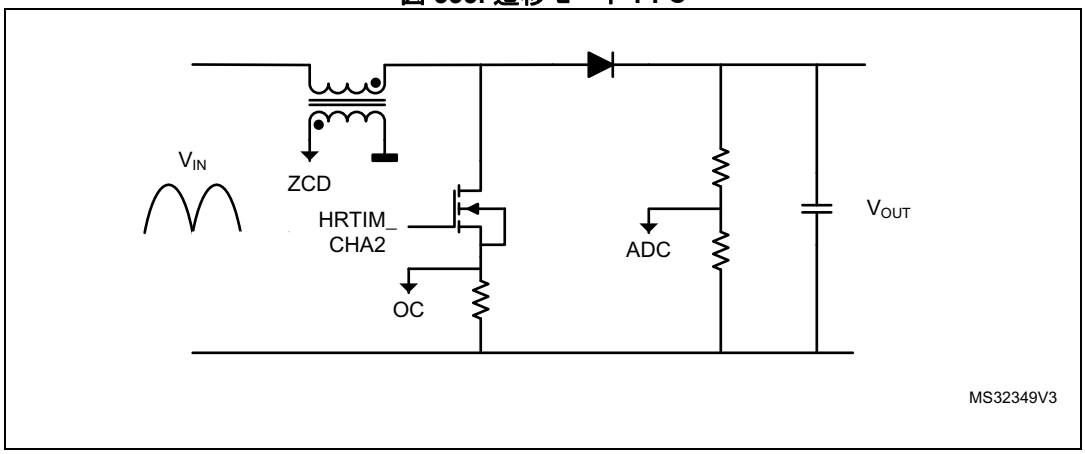
図 334. 3 相インターリーブ降圧コンバータの制御



37.4.4 遷移モード力率補正回路

基本的動作原理は、固定の Ton 時間中に電流をインダクタに蓄積することです。その後、この電流は Toff 時間中に減衰し、null になった時に周期が再開されます。これは、*図 335* に示すように、ゼロ交差検出回路（ZCD）を使用して検出されます。一定の Ton 時間により、インダクタ内のピーク電流値は整流された AC 入力電圧に比例し、それによって力率が補正されます。

図 335. 遷移モード PFC



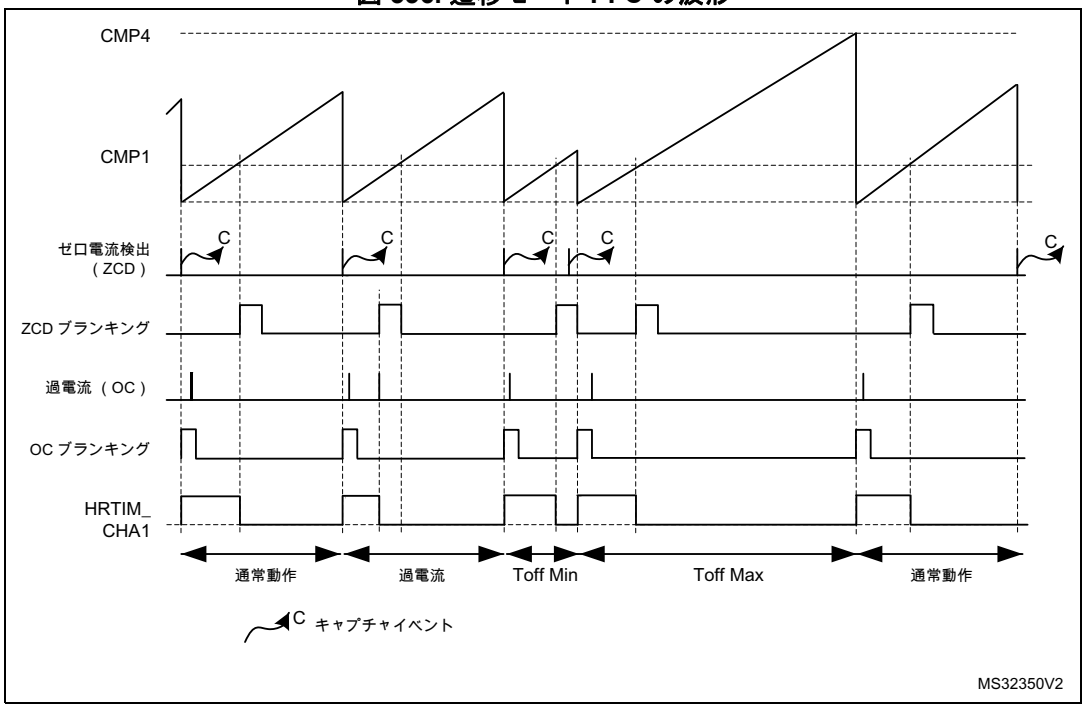
このコンバータは、一定の Ton 時間と、Toff 時間の変動（入力電圧に依存）による可変周波数によって動作します。また、ゼロ交差が検出されないとき動作し、または過電流（OC）の場合に Ton 時間を制限する機能も含まれている必要があります。OC フィードバックは通常、内蔵コンパレータで調整され、外部イベント入力に配送されます。

図 336 に、下記のパラメータ定義による、さまざまな動作モード中の波形を示します。

- Ton Min：誤った過電流（フリーホイーリングダイオードリカバリ電流）をマスクします。OC ブランキングとして表されます。
- Ton Max：実用上、コンバータの設定値。CMP1 によって定義されます。
- Toff Min：電流制限がゼロに近いとき（消磁は非常に速い）、周波数を制限します。CMP2 によって定義されます。
- Toff Max：ZCD が働かない場合に、システムが動かなくなることを防ぎます。自動遅延モードで CMP4 によって定義されます。

両方の Toff 値は、出力立ち下がりエッジに関連しなければならないため、自動遅延されます。

図 336. 遷移モード PFC の波形



37.5 HRTIM レジスタ

37.5.1 HRTIM マスタタイマ制御レジスタ（HRTIM_MCR）

アドレスオフセット：0x0000h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	Res.	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN
rw	rw	rw		rw	rw	rw				rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		Res.	Res.	HALF	RETRIG	CONT	CKPSC[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:30 **BRSTDMA[1:0]**：*バースト DMA 更新*

これらのビットは、バースト DMA トランザクションに関連して更新がどのように起きるかを指定します。

00：DMA バースト転送完了と独立に更新が行われます。

01：DMA バースト転送が完了した時に更新が行われます。

10：DMA バースト転送完了後にマスタタイマがロールオーバーするときに更新が行われます。このモードは、連続モードでのみ働きます。

11：予約済み

ビット 29 **MREPU**：マスタ*タイマ繰り返し更新*

このビットは、マスタタイマ繰り返し周期が（ロールオーバーまたはリセットイベントにより）完了した時、更新が起きるかどうかを決定します。MREPU は、BRSTDMA[1:0] = 00 または 01 の場合のみセットできます。

0：繰り返し時の更新は無効です。

1：繰り返し時の更新は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **PREEN**：*プリロードイネーブル*

このビットは、レジスタプリロードメカニズムを有効にし、メモリに配置されたレジスタへの書き込みアクセスが HRTIM のアクティブレジスタに行われるか、プリロードレジスタに行われるかを定義します。

0：プリロード無効：書き込みアクセスは直接、アクティブレジスタに行われます。

1：プリロード有効：書き込みアクセスはプリロードレジスタに行われます。

ビット 26:25 **DACSYNC[1:0]** *DAC 同期*

DAC 同期イベントを有効化でき、マスタタイマ更新が起きた時に生成されます。これらのビットは、どの出力に DAC 同期が送られるかを定義します（接続の詳細は*セクション 37.3.19：DAC トリガ*を参照）。

00：DAC トリガは生成されません。

01：トリガは hrtim_dac_trg1 に生成されます。

10：トリガは hrtim_dac_trg2 に生成されます。

11：トリガは hrtim_dac_trg3 に生成されます。

ビット 24:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TECEN**：*タイマ E カウンタイネーブル*

このビットはタイマ E カウンタを開始します。

0：タイマ E カウンタ無効化

1：タイマ E カウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 20 **TDCEN**：*タイマ D カウンタイネーブル*

このビットはタイマ D カウンタを開始します。

0：タイマ D カウンタ無効化

1：タイマ D カウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 19 **TCCEN**：*タイマ C カウンタイネーブル*

このビットはタイマ C カウンタを開始します。

0：タイマ C カウンタ無効化

1：タイマ C カウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 18 **TBCEN**：*タイマ B カウンタイネーブル*

このビットはタイマ B カウンタを開始します。

0：タイマ B カウンタ無効化

1：タイマ B カウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 17 **TACEN**：*タイマ A カウンタイネーブル*

このビットはタイマ A カウンタを開始します。

0：タイマ A カウンタ無効化

1：タイマ A カウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 16 **MCEN**：*マスタタイマカウンタイネーブル*

このビットはマスタタイマのカウンタを開始します。

0：マスタカウンタ無効化

1：マスタカウンタ有効化

注：　このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 15:14 **SYNCSRC[1:0]**：*同期ソース*

これらのビットは、同期出力 SYNCOUT[2:1] に送るソースおよびイベントを定義します。

00：マスタタイマの開始

01：マスタタイマの比較 1 イベント

10：タイマ A の開始／リセット

11：タイマ A の比較 1 イベント

ビット 13:12 **SYNCOUT[1:0]**：*同期出力*

これらのビットによって、同期出力イベントのルーチンと調整を定義します。

00：無効

01：予約済み。

10：HRTIM_SCOUT 出力へ正のパルス（16x f_{HRTIM} クロックサイクル）

11：HRTIM_SCOUT 出力へ負のパルス（16x f_{HRTIM} クロックサイクル）

注：　このビットフィールドは、カウンタが有効化（TxCEN　ビットのセット）された後には変更してはなりません。

ビット 11 **SYNCSTRTM**：*同期マスタ開始*

このビットは、同期入力イベントを受信した時、マスタタイマ開始を有効にします。

0：マスタタイマへの影響はありません。

1：同期入力イベントはマスタタイマを開始します。

ビット 10 **SYNCRSTM**：*同期マスタリセット*

このビットは、同期入力イベントを受信した時、マスタタイマリセットを有効にします。

0：マスタタイマへの影響はありません。

1：同期入力イベントはマスタタイマをリセットします。

ビット 9:8 **SYNCIN[1:0]** *同期入力*

これらのビットは、同期入力ソースを定義します。

00：無効HRTIM は同期されず、スタンドアローンモードで動作します。

01：予約済み。

10：内部イベント：HRTIM はオンチップタイマに同期します（*同期入力*を参照）。

11：外部イベント（入力ピン）HRTIM_SCIN 入力の正のパルスが HRTIM をトリガします。

注： このパラメータは、影響を受けたタイマが有効化された後は、変更できません。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **HALF**：*ハーフモード*

このビットは、ハーフデューティサイクルモードを有効にします：HRTIM_MCMP1xR アクティブレジスタは、HRTIM_MPER レジスタが書き込まれているとき、HRTIM_MPER/2 の値で自動的に更新されます。

0：ハーフモードは無効です。

1：ハーフモードは有効です。

ビット 4 **RETRIG**：*再トリガ可能モード*

このビットは、シングルショットモードでのマスタタイマのカウンタの動作を定義します。

0：タイマは再トリガ可能ではありません：カウンタリセットは、カウンタが停止（周期が経過）した場合のみ行われます。

1：タイマは再トリガ可能です：カウンタリセットは、カウンタの状態が何（動作中または停止中）であれ、行われます。

ビット 3 **CONT**：*連続モード*

0：タイマはシングルショットモードで動作し、MPER 値に達した時、停止します。

1：タイマは連続（フリーランニング）モードで動作し、MPER 値に達した時、ゼロに戻ります。

ビット 2:0 **CKPSC[2:0]**：*クロックプリスケーラ*

これらのビットは、マスタタイマのクロックプリスケーラ比を定義します。

カウンタクロックの等価周波数（$f_{COUNTER}$）は $f_{HRCK} / 2^{(CKPSC[2:0]-5)}$ に等しくなりす。

プリスケーラ比は、タイマが有効化された後は、変更できません。

000：予約済み

001：予約済み

010：予約済み

011：予約済み

100：予約済み

101：$f_{COUNTER} = f_{HRTIM}$

110：$f_{COUNTER} = f_{HRTIM} / 2$

111：$f_{COUNTER} = f_{HRTIM} / 4$

37.5.2 HRTIM マスタタイマ割り込みステータスレジスタ（HRTIM_MISR）

アドレスオフセット：0x0004h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPD**：マスタ更新割り込みフラグ

このビットは、マスタタイマレジスタが更新された時にハードウェアによってセットされます。

0：マスタ更新割り込みは起きていません。

1：マスタ更新割り込みが起きました。

ビット 5 **SYNC**：同期入力割り込みフラグ

このビットは、同期入力イベントが受信された時に、ハードウェアによってセットされます。

0：同期入力割り込みは起きていません。

1：同期入力割り込みが起きました。

ビット 4 **MREP**：マスタ繰り返し割り込みフラグ

このビットは、マスタタイマ繰り返し周期が経過した時に、ハードウェアによってセットされます。

0：マスタ繰り返し割り込みは起きていません。

1：マスタ繰り返し割り込みが起きました。

ビット 3 **MCMP4**：マスタ比較 4 割り込みフラグ

MCMP1 の説明を参照してください。

ビット 2 **MCMP3**：マスタ比較 3 割り込みフラグ

MCMP1 の説明を参照してください。

ビット 1 **MCMP2**：マスタ比較 2 割り込みフラグ

MCMP1 の説明を参照してください。

ビット 0 **MCMP1**：マスタ比較 1 割り込みフラグ

このビットは、マスタタイマカウンタが、マスタ比較 1 レジスタでプログラムされた値と一致した時に、ハードウェアによってセットされます。

0：マスタ比較 1 割り込みは起きていません。

1：マスタ比較 1 割り込みが起きました。

37.5.3 HRTIM マスタタイマ割り込みクリアレジスタ（HRTIM_MICR）

アドレスオフセット：0x0008h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
									w	w	w	w	w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPDC**：マスタ更新割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MUPDC フラグがクリアされます。

ビット 5 **SYNCC**：同期入力割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの SYNC フラグがクリアされます。

ビット 4 **MREPC**：繰り返し割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MREP フラグがクリアされます。

ビット 3 **MCMP4C**：マスタ比較 4 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP4 フラグがクリアされます。

ビット 2 **MCMP3C**：マスタ比較 3 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP3 フラグがクリアされます。

ビット 1 **MCMP2C**：マスタ比較 2 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP2 フラグがクリアされます。

ビット 0 **MCMP1C**：マスタ比較 1 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP1 フラグがクリアされます。

37.5.4 HRTIM マスタタイマ DMA／割り込みイネーブルレジスタ（HRTIM_MDIER）

アドレスオフセット：0x000Ch

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **MUPDDE**：マスタ更新 DMA リクエストイネーブル

このビットは、マスタ更新 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：マスタ更新 DMA リクエストは無効です。

1：マスタ更新 DMA リクエストは有効です。

ビット 21 **SYNCDE**：同期入力 DMA リクエストイネーブル

このビットは、同期入力 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：同期入力 DMA リクエストは無効です。

1：同期入力 DMA リクエストは有効です。

ビット 20 **MREPDE**：マスタ繰り返し DMA リクエストイネーブル

このビットは、マスタタイマ繰り返し DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：繰り返し DMA リクエストは無効です。

1：繰り返し DMA リクエストは有効です。

ビット 19 **MCMP4DE**：マスタ比較 4 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 18 **MCMP3DE**：マスタ比較 3 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 17 **MCMP2DE**：マスタ比較 2 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 16 **MCMP1DE**：マスタ比較 1 DMA リクエストイネーブル

このビットは、マスタタイマ比較 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：比較 1 DMA リクエストは無効です。

1：比較 1 DMA リクエストは有効です。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPDIE**：マスタ更新割り込みイネーブル

このビットは、マスタタイマレジスタ更新割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：マスタ更新割り込みは無効です。

1：マスタ更新割り込みは有効です。

ビット 5 **SYNCIE**：同期入力割り込みイネーブル

このビットは、同期入力割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：同期入力割り込みは無効です。

1：同期入力割り込みは有効です。

ビット 4 **MREPIE**：マスタ繰り返し割り込みイネーブル

このビットは、マスタタイマ繰り返し割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：マスタ繰り返し割り込みは無効です。

1：マスタ繰り返し割り込みは有効です。

ビット 3 **MCMP4IE**：マスタ比較 4 割り込みイネーブル

MCMP1IE の説明を参照してください。

ビット 2 **MCMP3IE**：マスタ比較 3 割り込みイネーブル

MCMP1IE の説明を参照してください。

ビット 1 **MCMP2IE**：マスタ比較 2 割り込みイネーブル

MCMP1IE の説明を参照してください。

ビット 0 **MCMP1IE**：マスタ比較 1 割り込みイネーブル

このビットは、マスタタイマ比較 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：比較 1 割り込みは無効です。

1：比較 1 割り込みは有効です。

37.5.5 HRTIM マスタタイマカウンタレジスタ（HRTIM_MCNTR）

アドレスオフセット：0x0010h

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCNT[15:0]**：*カウンタ値*

マスタタイマカウンタ値を保持します。このレジスタは、マスタタイマが停止している（HRTIM_MCR で MCEN = 0）ときのみ書き込むことができます。

注： カウンタ値が HRTIM_MPER レジスタ値を超えてセットされた場合は、タイマの動作は保証されません。

37.5.6 HRTIM マスタタイマ周期レジスタ（HRTIM_MPER）

アドレスオフセット：0x0014h

リセット値：0x0000 FFDF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MPER[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MPER[15:0]**：*マスタタイマ周期値*

このレジスタは、カウンタオーバーフロー値を定義します。

周期値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

最大値は 0x0000 FFDF です。

37.5.7 HRTIM マスタタイマ繰り返しレジスタ（HRTIM_MREP）

アドレスオフセット：0x0018h

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **MREP[7:0]**：*マスタタイマ繰り返し周期値*

このレジスタはマスタカウンタの繰り返し周期値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

37.5.8 HRTIM マスタタイマ比較 1 レジスタ（HRTIM_MCMP1R）

アドレスオフセット：0x001Ch

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP1[15:0]**：*マスタタイマの比較 1 の値*

このレジスタはマスタタイマ比較 1 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.9 HRTIM マスタタイマ比較 2 レジスタ（HRTIM_MCMP2R）

アドレスオフセット：0x0024h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP2[15:0]**：*マスタタイマの比較 2 の値*

このレジスタはマスタタイマ比較 2 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.10 HRTIM マスタタイマ比較 3 レジスタ（HRTIM_MCMP3R）

アドレスオフセット：0x0028h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP3[15:0]**：*マスタタイマの比較 3 の値*

このレジスタはマスタタイマ比較 3 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.11 HRTIM マスタタイマ比較 4 レジスタ（HRTIM_MCMP4R）

アドレスオフセット：0x002Ch

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP4[15:0]**：*マスタタイマの比較 4 の値*

このレジスタはマスタタイマ比較 4 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.12 HRTIM タイマ x 制御レジスタ（HRTIM_TIMxCR）

アドレスオフセット：0X0000h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDGAT[3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	Res.	TxRSTU	TxREPU	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DELCMP4[1:0]		DELCMP2[1:0]		SYNCSTRTx	SYNCRSTx	Res.	Res.	Res.	PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **UPDGAT[3:0]**：*更新ゲート操作*

これらのビットは、バースト DMA トランザクションおよび、更新イネーブル入力 1～3（*表 293：更新イネーブル入力とソース*参照）の外部更新リクエストに関連して更新がどのように起きるかを指定します。

以下に述べる更新イベントになり得るのは：MSTU、TEU、TDU、TCU、TBU、TAU、TxRSTU、TxREPU です。

0000：DMA バースト転送と独立に更新が起きます。

0001：DMA バースト転送が完了した時に更新が起きます。

0010：DMA バースト転送が完了した後の更新イベント時に更新が起きます。

0011：HRTIM 更新イネーブル入力 1（hrtim_upd_en1）の立ち上がりエッジで更新が起きます。

0100：HRTIM 更新イネーブル入力 2（hrtim_upd_en2）の立ち上がりエッジで更新が起きます。

0101：HRTIM 更新イネーブル入力 3（hrtim_upd_en3）の立ち上がりエッジで更新が起きます。

0110：HRTIM 更新イネーブル入力 1（hrtim_upd_en1）の立ち上がりエッジ後の更新イベント時に、更新が起きます。

0111：HRTIM 更新イネーブル入力 2（hrtim_upd_en2）の立ち上がりエッジ後の更新イベント時に、更新が起きます。

1000：HRTIM 更新イネーブル入力 3（hrtim_upd_en3）の立ち上がりエッジ後の更新イベント時に、更新が起きます。

その他のコード：予約済み

注：　このビットフィールドは、新しい値をプログラムする前にリセットする必要があります。

0001、0011、0100、0101 に等しい UPDGAT[3:0] の値に対しては、複数の同時更新ソースを受けることが可能です（たとえば、RSTU と DMA バースト）。

ビット 27 **PREEN**：*プリロードイネーブル*

このビットは、レジスタプリロードメカニズムを有効にし、プリロード可能なレジスタへの書き込みアクセスがアクティブレジスタに行われるか、プリロードレジスタに行われるかを決定します。

0：プリロード無効：書き込みアクセスは直接、アクティブレジスタに行われます。

1：プリロード有効：書き込みアクセスはプリロードレジスタに行われます。

ビット 26:25 **DACSYNC[1:0]** *DAC 同期*

タイマ更新が起きた時、DAC 同期イベントが生成されます。これらのビットは、どの出力に DAC 同期が送られるかを定義します（接続の詳細は*セクション 37.3.19：DAC トリガ*を参照）。

00：DAC トリガは生成されません。

01：トリガは hrtim_dac_trg1 に生成されます。

10：トリガは hrtim_dac_trg2 に生成されます。

11：トリガは hrtim_dac_trg3 に生成されます。

ビット 24 **MSTU**：*マスタタイマ更新*
レジスタ更新はマスタタイマの更新によってトリガされます。
0：マスタタイマによる更新は無効です。
1：マスタタイマによる更新は有効です。

ビット 23 HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMDCR において：
TEU：*タイマ E 更新*
レジスタ更新はタイマ E の更新によってトリガされます。
0：タイマ E による更新は無効です。
1：タイマ E による更新は有効です。

HRTIM_TIMECR において：
予約済みであり、リセット値に保持する必要があります。

ビット 22 HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMECR において：
TDU：*タイマ D 更新*
レジスタ更新はタイマ D の更新によってトリガされます。
0：タイマ D による更新は無効です。
1：タイマ D による更新は有効です。

HRTIM_TIMDCR において：
予約済みであり、リセット値に保持する必要があります。

ビット 21 HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMDCR、HRTIM_TIMECR において：
TCU：*タイマ C 更新*
レジスタ更新はタイマ C の更新によってトリガされます。
0：タイマ C による更新は無効です。
1：タイマ C による更新は有効です。

HRTIM_TIMCCR において：
予約済みであり、リセット値に保持する必要があります。

ビット 20 HRTIM_TIMACR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMECR において：
TBU：*タイマ B 更新*
レジスタ更新はタイマ B の更新によってトリガされます。
0：タイマ B による更新は無効です。
1：タイマ B による更新は有効です。

HRTIM_TIMBCR において：
予約済みであり、リセット値に保持する必要があります。

ビット 19 HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMECR において：
TAU：*タイマ A 更新*
レジスタ更新はタイマ A の更新によってトリガされます。
0：タイマ A による更新は無効です。
1：タイマ A による更新は有効です。

HRTIM_TIMACR において：
予約済みであり、リセット値に保持する必要があります。

ビット 18 **TxRSTU**：*タイマx リセット更新*

レジスタ更新は、タイマ x カウンタのリセットによって、または連続モードで周期値に達した後の 0 へのロールオーバーによってトリガされます。

0：タイマ x のリセット／ロールオーバーによる更新は無効です。

1：タイマ x のリセット／ロールオーバーによる更新は有効です。

ビット 17 **TxREPU**：*タイマx 繰り返し更新*

レジスタ更新はカウンタがロールオーバーし、HRTIM_REPx = 0 になった時、トリガされます。

0：繰り返し時の更新は無効です。

1：繰り返し時の更新は有効です。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **DELCMP4[1:0]**：*CMP4 自動遅延モード*

このビットフィールドは、比較レジスタが標準モード（カウンタが比較値に等しくなるとすぐに比較一致が発行される）で動作するか、自動遅延モード（*自動遅延モード*を参照）で動作するかを定義します。

00：CMP4 レジスタは常にアクティブです（標準比較モード）。

01：CMP4 の値はキャプチャ 2 イベント後に再計算されてアクティブになります。

10：CMP4 の値は、キャプチャ 2 イベントの後に再計算されてアクティブになるか、または比較 1 の一致後に再計算されてアクティブになります（キャプチャ 2 イベントが失われた場合のタイムアウト機能）。

11：CMP4 の値は、キャプチャイベントの後、再計算されてアクティブになるか、または比較 3 の一致後、再計算されてアクティブになります（キャプチャイベントが失われた場合のタイムアウト機能）。

注：　このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 13:12 **DELCMP2[1:0]**：*CMP2 自動遅延モード*

このビットフィールドは、比較レジスタが標準モード（カウンタが比較値に等しくなるとすぐに比較一致が発行される）で動作するか、自動遅延モード（*自動遅延モード*を参照）で動作するかを定義します。

00：CMP2 レジスタは常にアクティブです（標準比較モード）。

01：CMP2 の値はキャプチャ 1 イベント後に再計算されてアクティブになります。

10：CMP2 の値は、キャプチャ 1 イベントの後に再計算されてアクティブになるか、または比較 1 の一致後に再計算されてアクティブになります（キャプチャイベントが失われた場合のタイムアウト機能）。

11：CMP2 の値は、キャプチャ 1 イベントの後に再計算されてアクティブになるか、または比較 3 の一致後に再計算されてアクティブになります（キャプチャイベントが失われた場合のタイムアウト機能）。

注：　このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 11 **SYNCSTRTx**：*同期タイマx 開始*

このビットは、同期イベント後のタイマ x の動作を定義します。

0：タイマ x に影響しません。

1：同期入力イベントはタイマ x を開始します。

ビット 10 **SYNCRSTx**：*同期タイマx リセット*

このビットは、同期イベント後のタイマ x の動作を定義します。

0：タイマ x に影響しません。

1：同期入力イベントはタイマ x をリセットします。

ビット 9:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **PSHPLL**：*プッシュプルモードイネーブル*

このビットはプッシュプルモードを有効にします。

0：プッシュプルモードは無効です。

1：プッシュプルモードイは有効です。

注：　このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 5 **HALF**：*ハーフモードイネーブル*

このビットは、ハーフデューティサイクルモードを有効にします：HRTIM_CMP1xR アクティブレジスタは、HRTIM_PERxR レジスタが書き込まれているとき、HRTIM_PERxR/2 の値で自動的に更新されます。

0：ハーフモードは無効です。

1：ハーフモードは有効です。

ビット 4 **RETRIG**：*再トリガ可能モード*

このビットは、シングルショットモードでのカウンタの動作を定義します。

0：タイマは再トリガ可能ではありません：カウンタリセットは、カウンタが停止（シングルショットモードで周期が経過、または連続モードでカウンタが停止）した場合に行われます。

1：タイマは再トリガ可能です：カウンタリセットは、カウンタの状態がどうであれ、行われます。

ビット 3 **CONT**：*連続モード*

このビットはタイマの動作モードを定義します。

0：タイマはシングルショットモードで動作し、TIMxPER 値に達した時、停止します。

1：タイマは連続モードで動作し、TIMxPER 値に達した時、ゼロに戻ります。

ビット 2:0 **CKPSCx[2:0]**：*HRTIM タイマ x クロックプリスケーラ*

これらのビットは、マスタタイマのクロックプリスケーラ比を定義します。

カウンタクロックの等価周波数（$f_{COUNTER}$）は $f_{HRCK} / 2^{(CKPSC[2:0]-5)}$ に等しくなりす。

プリスケーラ比は、タイマが有効化された後は、変更できません。

000：予約済み

001：予約済み

010：予約済み

011：予約済み

100：予約済み

101：$f_{COUNTER} = f_{HRTIM}$

110：$f_{COUNTER} = f_{HRTIM} / 2$

111：$f_{COUNTER} = f_{HRTIM} / 4$

37.5.13 HRTIM タイマ x 割り込みステータスレジスタ（HRTIM_TIMxISR）

アドレスオフセット：0x0004h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2STAT	O1STAT	IPPSTAT	CPPSTAT
										r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	r	r	r	r	r	r	r	r	r		r	r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **O2CPY**：出力 2 コピー

このステータスビットは、出力ステージ（チョッパ、極性）前の、出力 2 状態の生のコピーです。これによって、遅延保護後に出力を再有効化する前に現在の出力状態を確認できます。

0：出力 2 は非アクティブです。

1：出力 2 はアクティブです。

ビット 20 **O1CPY**：出力 1 コピー

このステータスビットは、出力ステージ（チョッパ、極性）前の、出力 1 状態の生のコピーです。これによって、遅延保護後に出力を再有効化する前に現在の出力状態を確認できます。

0：出力 1 は非アクティブです。

1：出力 1 はアクティブです。

ビット 19 **O2STAT**：出力 2 ステータス

このステータスビットは、遅延アイドル保護がトリガされた時、出力 2 の状態を示します。このビットは、新たに遅延保護に移行した時に更新されます。このビットはバランスアイドルでは更新されません。

0：出力 2 は非アクティブでした。

1：出力 2 はアクティブでした。

ビット 18 **O1STAT**：出力 1 ステータス

このステータスビットは、遅延アイドル保護がトリガされた時、出力 1 の状態を示します。このビットは、新たに遅延保護に移行した時に更新されます。このビットはバランスアイドルでは更新されません。

0：出力 1 は非アクティブでした。

1：出力 1 はアクティブでした。

ビット 17 **IPPSTAT**：アイドルプッシュプルステータス

このステータスビットは、プッシュプルモード、バランスフォールトモード、または遅延アイドルモードで、保護がトリガされた時（出力状態がアクティブであれ、非アクティブであれ）、どちらの出力に信号が適用されていたかを示します。

0：出力 1 がアクティブで、出力 2 が非アクティブに強制されていた時に、保護が発生しました。

1：出力 2 がアクティブで、出力 1 が非アクティブに強制されていた時に、保護が発生しました。

ビット 16 **CPPSTAT**：現在のプッシュプルステータス

このステータスビットは、プッシュプルモードで信号が現在、どちらの出力に適用されているかを示します。これは、この設定でのみ意味を持ちます。

0：信号は出力 1 に適用されており、出力 2 は非アクティブに強制されています。

1：信号は出力 2 に適用されており、出力 1 は非アクティブに強制されています。

ビット 15 予約済み

ビット 14 **DLYPRT**：遅延保護フラグ

このビットは、保護アイドルモードまたはバランスアイドルモードへの移行を示します。

ビット 13 **RST**：リセットおよび／またはロールオーバー割り込みフラグ

このビットは、タイマ x カウンタがリセットされた、または連続モードでロールオーバーした時に、ハードウェアによってセットされます。

0：TIMx カウンタのリセット／ロールオーバー割り込みは起きていません。

1：TIMx カウンタのリセット／ロールオーバー割り込みが起きました。

ビット 12 **RSTx2**：出力 2 リセット割り込みフラグ

RSTx1 の説明を参照してください。

ビット 11 **SETx2**：出力 2 セット割り込みフラグ

SETx1 の説明を参照してください。

ビット 10 **RSTx1**：出力 1 リセット割り込みフラグ

このビットは、Tx1 出力がリセット（アクティブから非アクティブモードに移行）された時に、ハードウェアによってセットされます。

0：Tx1 出力リセット割り込みは起きていません。

1：Tx1 出力リセット割り込みが起きました。

ビット 9 **SETx1**：出力 1 セット割り込みフラグ

このビットは、Tx1 出力がセット（非アクティブからアクティブモードに移行）された時に、ハードウェアによってセットされます。

0：Tx1 出力セット割り込みは起きていません。

1：Tx1 出力セット割り込みが起きました。

ビット 8 **CPT2**：キャプチャ 2 割り込みフラグ

CPT1 の説明を参照してください。

ビット 7 **CPT1**：キャプチャ 1 割り込みフラグ

このビットは、タイマ x キャプチャ 1 イベントが発生した時に、ハードウェアによってセットされます。

0：タイマ x キャプチャ 1 リセット割り込みは起きていません。

1：タイマ x 出力 1 リセット割り込みが起きました。

ビット 6 **UPD**：更新割り込みフラグ

このビットは、タイマ x 更新イベントが発生した時に、ハードウェアによってセットされます。

0：タイマ x 更新割り込みは起きていません。

1：タイマ x 更新割り込みが起きました。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REP**：繰り返し割り込みフラグ

このビットは、タイマ x 繰り返し周期が経過した時に、ハードウェアによってセットされます。

0：タイマ x 繰り返し割り込みは起きていません。

1：タイマ x 繰り返し割り込みが起きました。

ビット 3 **CMP4**：比較 4 割り込みフラグ

CMP1 の説明を参照してください。

ビット 2 **CMP3**：比較 3 割り込みフラグ
CMP1 の説明を参照してください。

ビット 1 **CMP2**：比較 2 割り込みフラグ
CMP1 の説明を参照してください。

ビット 0 **CMP1**：比較 1 割り込みフラグ
このビットは、タイマ x カウンタが、比較 1 レジスタでプログラムされた値に一致した時に、ハードウェアによってセットされます。
0：比較 1 割り込みは起きていません。
1：比較 1 割り込みが起きました。

37.5.14 HRTIM タイマ x 割り込みクリアレジスタ（HRTIM_TIMxICR）

アドレスオフセット：0x0008h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DLYPRTC**：遅延保護フラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの DLYPRT フラグがクリアされます。

ビット 13 **RSTC**：リセット割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RST フラグがクリアされます。

ビット 12 **RSTx2C**：出力 2 リセットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RSTx2 フラグがクリアされます。

ビット 11 **SETx2C**：出力 2 セットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの SETx2 フラグがクリアされます。

ビット 10 **RSTx1C**：出力 1 リセットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RSTx1 フラグがクリアされます。

ビット 9 **SETx1C**：出力 1 セットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの SETx1 フラグがクリアされます。

ビット 8 **CPT2C**：キャプチャ 2 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CPT2 フラグがクリアされます。

ビット 7 **CPT1C**：キャプチャ 1 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CPT1 フラグがクリアされます。

ビット 6 **UPDC**：更新割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの UPD フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REPC**：繰り返し割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの REP フラグがクリアされます。

ビット 3 **CMP4C**：比較 4 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP4 フラグがクリアされます。

ビット 2 **CMP3C**：比較 3 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP3 フラグがクリアされます。

ビット 1 **CMP2C**：比較 2 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP2 フラグがクリアされます。

ビット 0 **CMP1C**：比較 1 割り込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP1 フラグがクリアされます。

37.5.15 HRTIM タイマ x DMA／割り込みイネーブルレジスタ（HRTIM_TIMxDIER）

アドレスオフセット：0x000Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SETx1DE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31 予約済み

ビット 30 **DLYPRTDE**：遅延保護 DMA リクエストイネーブル
このビットは、遅延保護の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：遅延保護 DMA リクエストは無効です。
1：遅延保護 DMA リクエストは有効です。

ビット 29 **RSTDE**：リセット／ロールオーバー DMA リクエストイネーブル
このビットは、タイマ x カウンタのリセットまたは連続モードでのロールオーバー時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：タイマ x カウンタのリセット／ロールオーバー DMA リクエストは無効です。
1：タイマ x カウンタのリセット／ロールオーバー DMA リクエストは有効です。

ビット 28 **RSTx2DE**：出力 2 リセット DMA リクエストイネーブル
RSTx1DE の説明を参照してください。

ビット 27 **SETx2DE**：出力 2 セット DMA リクエストイネーブル
SETx1DE の説明を参照してください。

ビット 26 **RSTx1DE**：出力 1 リセット DMA リクエストイネーブル
このビットは、Tx1 出力リセット DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：Tx1 出力リセット DMA リクエストは無効です。
1：Tx1 出力リセット DMA リクエストは有効です。

ビット 25 **SETx1DE**：出力 1 セット DMA リクエストイネーブル

このビットは、Tx1 出力セット DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx1 出力セット DMA リクエストは無効です。

1：Tx1 出力セット DMA リクエストは有効です。

ビット 24 **CPT2DE**：キャプチャ 2 DMA リクエストイネーブル

CPT1DE の説明を参照してください。

ビット 23 **CPT1DE**：キャプチャ 1 DMA リクエストイネーブル

このビットは、キャプチャ 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：キャプチャ 1 DMA リクエストは無効です。

1：キャプチャ 1 DMA リクエストは有効です。

ビット 22 **UPDDE**：更新 DMA リクエストイネーブル

このビットは、更新イベント時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **REPDE**：繰り返し DMA リクエストイネーブル

このビットは、繰り返しイベント時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：繰り返し DMA リクエストは無効です。

1：繰り返し DMA リクエストは有効です。

ビット 19 **CMP4DE**：比較 4 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 18 **CMP3DE**：比較 3 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 17 **CMP2DE**：比較 2 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 16 **CMP1DE**：比較 1 DMA リクエストイネーブル

このビットは、比較 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：比較 1 DMA リクエストは無効です。

1：比較 1 DMA リクエストは有効です。

ビット 15 予約済み

ビット 14 **DLYPRTIE**：遅延保護割り込みイネーブル

このビットは、遅延保護の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：遅延保護割り込みは無効です。

1：遅延保護割り込みは有効です。

ビット 13 **RSTIE**：リセット／ロールオーバー割り込みイネーブル

このビットは、タイマ x カウンタのリセットまたは連続モードでのロールオーバー時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：タイマ x カウンタのリセット／ロールオーバー割り込みは無効です。

1：タイマ x カウンタのリセット／ロールオーバー割り込みは有効です。

ビット 12 **RSTx2IE**：出力 2 リセット割り込みイネーブル

RSTx1IE の説明を参照してください。

ビット 11 **SETx2IE**：出力 2 セット割り込みイネーブル
SETx1IE の説明を参照してください。

ビット 10 **RSTx1IE**：出力 1 リセット割り込みイネーブル
このビットは、Tx1 出力リセット割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：Tx1 出力リセット割り込みは無効です。
1：Tx1 出力リセット割り込みは有効です。

ビット 9 **SETx1IE**：出力 1 セット割り込みイネーブル
このビットは、Tx1 出力セット割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：Tx1 出力セット割り込みは無効です。
1：Tx1 出力セット割り込みは有効です。

ビット 8 **CPT2IE**：キャプチャ割り込みイネーブル
CPT1IE の説明を参照してください。

ビット 7 **CPT1IE**：キャプチャ割り込みイネーブル
このビットは、キャプチャ 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：キャプチャ 1 割り込みは無効です。
1：キャプチャ 1 割り込みは有効です。

ビット 6 **UPDIE**：更新割り込みイネーブル
このビットは、更新イベント割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：更新割り込みは無効です。
1：更新割り込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REPIE**：繰り返し割り込みイネーブル
このビットは、繰り返しイベント割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：繰り返し割り込みは無効です。
1：繰り返し割り込みは有効です。

ビット 3 **CMP4IE**：比較 4 割り込みイネーブル
CMP1IE の説明を参照してください。

ビット 2 **CMP3IE**：比較 3 割り込みイネーブル
CMP1IE の説明を参照してください。

ビット 1 **CMP2IE**：比較 2 割り込みイネーブル
CMP1IE の説明を参照してください。

ビット 0 **CMP1IE**：比較 1 割り込みイネーブル
このビットは、比較 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：比較 1 割り込みは無効です。
1：比較 1 割り込みは有効です。

37.5.16 HRTIM タイマ x カウンタレジスタ（HRTIM_CNTxR）

アドレスオフセット：0x0010h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNTx[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNTx[15:0]**：*タイマ x カウンタ値*

このレジスタは、タイマ x カウンタ値を保持します。これは、タイマが停止している（HRTIM_TIMxCR で TxCEN = 0）ときのみ書き込むことができます。

注： カウンタ値が HRTIM_PERxR レジスタ値を超えている場合は、タイマの動作は保証されません。

37.5.17 HRTIM タイマ x 周期レジスタ（HRTIM_PERxR）

アドレスオフセット：0x14h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 FFDF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERx[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PERx[15:0]**：*タイマ x 周期値*

このレジスタは、タイマ x 周期値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

周期値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

最大値は 0x0000 FFDF です。

37.5.18 HRTIM タイマ x 繰り返しレジスタ（HRTIM_REPxR）

アドレスオフセット：0x18h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7:0 **REPx[7:0]**：*タイマ x 繰り返し周期値*

このレジスタは、繰り返し周期値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

37.5.19 HRTIM タイマ x 比較 1 レジスタ（HRTIM_CMP1xR）

アドレスオフセット：0x1Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP1x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP1x[15:0]**：*タイマ x 比較 1 値*

このレジスタは比較 1 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.20 HRTIM タイマ x 比較 1 合成レジスタ（HRTIM_CMP1CxR）

アドレスオフセット : 0x20h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP1x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **REPx[7:0]** : *タイマ x 繰り返し値（HRTIM_REPx レジスタからのエイリアス）*

このビットフィールドは、HRTIMx_REPxR レジスタの REPx[7:0] ビットフィールドからのエイリアスです。

ビット 15:0 **CMP1x[15:0]** : *タイマ x 比較 1 値*

このビットフィールドは、HRTIMx_CMP1xR レジスタの CMP1x[15:0] ビットフィールドからのエイリアスです。

37.5.21 HRTIM タイマ x 比較 2 レジスタ（HRTIM_CMP2xR）

アドレスオフセット : 0x24h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP2x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP2x[15:0]** : *タイマ x 比較 2 値*

このレジスタは比較 2 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

このレジスタは、HRTIM_TIMxCR の DELCMP2[1:0] ビットで有効にされている場合、自動遅延比較レジスタとして動作できます。

37.5.22 HRTIM タイマ x 比較 3 レジスタ（HRTIM_CMP3xR）

アドレスオフセット : 0x28h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP3x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP3x[15:0]** : *タイマ x 比較 3 値*

このレジスタは比較 3 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

37.5.23 HRTIM タイマ x 比較 4 レジスタ（HRTIM_CMP4xR）

アドレスオフセット : 0x2Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP4x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP4x[15:0]** : *タイマ x 比較 4 値*

このレジスタは比較 4 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は、f_{HRTIM} クロックの 3 周期以上である必要があります。

このレジスタは、HRTIM_TIMxCR の DELCMP4[1:0] ビットで有効にされている場合、自動遅延比較レジスタとして動作できます。

参考資料

37.5.24 HRTIM タイマ x キャプチャ 1 レジスタ（HRTIM_CPT1xR）

アドレスオフセット : 0x30h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT1x[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CPT1x[15:0]** : *タイマ x キャプチャ 1 値*

このレジスタは、キャプチャ 1 イベントが発生した時のカウンタ値を保持します。

37.5.25 HRTIM タイマ x キャプチャ 2 レジスタ（HRTIM_CPT2xR）

アドレスオフセット : 0x34h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT2x[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CPT2x[15:0]** : *タイマ x キャプチャ 2 値*

このレジスタは、キャプチャ 2 イベントが発生した時のカウンタ値を保持します。

37.5.26 HRTIM タイマ x デッドタイムレジスタ（HRTIM_DTxR）

アドレスオフセット : 0x38h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]								
rwo	rwo					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTRLKx	DTRSLKx	Res.	DTPRSC[1:0]			SDTRx	DTRx[8:0]								
rwo	rwo		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DTFLKx** : *デッドタイム立ち下がりロック*

この 1 回のみ書き込み可能のビットは、デッドタイム（正負符号および値）が変更されることを防ぎます（有効な場合）。

0 : デッドタイム立ち下がりの値と正負符号は書き込み可能です。

1 : デッドタイム立ち下がりの値と正負符号は読み出し専用です。

注 : このビットはプリロードされません。

ビット 30 **DTFSLKx** : *デッドタイム立ち下がり正負符号ロック*

この 1 回のみ書き込み可能のビットは、立ち下がりデッドタイムの正負符号が変更されることを防ぎます（有効な場合）。

0 : デッドタイム立ち下がりの正負符号は書き込み可能です。

1 : デッドタイム立ち下がりの正負符号は読み出し専用です。

注 : このビットはプリロードされません。

ビット 29:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **SDTFx** : *デッドタイム立ち下がり値の正負符号*

このレジスタは、デッドタイムが正か（信号はオーバーラップしない）、または負か（信号はオーバーラップする）を決定します。

0 : 立ち下がりエッジ時の正のデッドタイム

1 : 立ち下がりエッジ時の負のデッドタイム

ビット 24:16 **DTFx[8:0]** : *デッドタイム立ち下がり値*

このレジスタは、リファレンス PWM 信号の立ち下がりエッジの後のデッドタイムの値を保持します。

t_{DTF} = DTFx[8:0] x t_{DTG}

ビット 15 **DTRLKx** : *デッドタイム立ち上がりロック*

この 1 回のみ書き込み可能のビットは、デッドタイム（正負符号および値）が変更されることを防ぎます（有効な場合）。

0 : デッドタイム立ち上がりの値と正負符号は書き込み可能です。

1 : デッドタイム立ち上がりの値と正負符号は読み出し専用です。

注 : このビットはプリロードされません。

ビット 14 **DTRSLKx** : *デッドタイム立ち上がり正負符号ロック*

この 1 回のみ書き込み可能のビットは、デッドタイムの正負符号が変更されることを防ぎます（有効な場合）。

0 : デッドタイム立ち上がりの正負符号は書き込み可能です。

1 : デッドタイム立ち上がりの正負符号は読み出し専用です。

注 : このビットはプリロードされません。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:10 **DTPRSC[2:0]**：*デッドタイムプリスケーラ*

このレジスタは、デッドタイムのクロックプリスケーラの値を保持します。

t_{DTG} = （$2^{(DTPRSC[2:0]-3)}$）x t_{HRTIM}

000：予約済み

001：予約済み

010：予約済み

011：tDTG= t_{HRTIM}

100：tDTG= t_{HRTIM} x 2

101：tDTG= t_{HRTIM} x 4

110：tDTG= t_{HRTIM} x 8

111：tDTG= t_{HRTIM} x 16

このビットフィールドは、ロックビット（DTFLKs、DTFSLKx、DTRLKx、DTRSLKx）のどれかが有効になるとすぐに読み出し専用になります。

ビット 9 **SDTRx**：*デッドタイム立ち上がり値の正負符号*

このレジスタは、デッドタイムが正か、または負か（信号はオーバーラップする）を決定します。

0：立ち上がりエッジ時の正のデッドタイム

1：立ち上がりエッジ時の負のデッドタイム

ビット 8:0 **DTRx[8:0]**：*デッドタイム立ち上がり値*

このレジスタは、リファレンス PWM 信号の立ち上がりエッジの後のデッドタイムの値を保持します。

t_{DTR} = DTRx[8:0] x t_{DTG}

37.5.27 HRTIM タイマ x 出力 1 セットレジスタ（HRTIM_SETx1R）

アドレスオフセット : 0x3Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UPDATE**：*レジスタ更新（プリロードからアクティブへ転送）*
レジスタ更新イベントは出力をアクティブ状態に強制します。

ビット 30 **EXTEVNT10**：*外部イベント 10*
EXTEVNT1 の説明を参照してください。

ビット 29 **EXTEVNT9**：*外部イベント 9*
EXTEVNT1 の説明を参照してください。

ビット 28 **EXTEVNT8**：*外部イベント 8*
EXTEVNT1 の説明を参照してください。

ビット 27 **EXTEVNT7**：*外部イベント 7*
EXTEVNT1 の説明を参照してください。

ビット 26 **EXTEVNT6**：*外部イベント 6*
EXTEVNT1 の説明を参照してください。

ビット 25 **EXTEVNT5**：*外部イベント 5*
EXTEVNT1 の説明を参照してください。

ビット 24 **EXTEVNT4**：*外部イベント 4*
EXTEVNT1 の説明を参照してください。

ビット 23 **EXTEVNT3**：*外部イベント 3*
EXTEVNT1 の説明を参照してください。

ビット 22 **EXTEVNT2**：*外部イベント 2*
EXTEVNT1 の説明を参照してください。

ビット 21 **EXTEVNT1**：*外部イベント 1*
外部イベント 1 は出力をアクティブ状態に強制します。

ビット 20 **TIMEVNT9**：*タイマイベント 9*
TIMEVNT1 の説明を参照してください。

ビット 19 **TIMEVNT8**：*タイマイベント 8*
TIMEVNT1 の説明を参照してください。

ビット 18 **TIMEVNT7**：*タイマイベント 7*
TIMEVNT1 の説明を参照してください。

ビット 17 **TIMEVNT6**：*タイマイベント 6*
TIMEVNT1 の説明を参照してください。

ビット 16 **TIMEVNT5**：*タイマイベント 5*
TIMEVNT1 の説明を参照してください。

ビット 15 **TIMEVNT4**：*タイマイベント 4*
TIMEVNT1 の説明を参照してください。

ビット 14 **TIMEVNT3**：*タイマイベント 3*
TIMEVNT1 の説明を参照してください。

ビット 13 **TIMEVNT2**：*タイマイベント 2*
TIMEVNT1 の説明を参照してください。

ビット 12 **TIMEVNT1**：*タイマイベント 1*
タイマイベント 1 は出力をアクティブ状態に強制します（タイマイベントの割り当てについては*表 286* を参照）。

ビット 11 **MSTCMP4**：*マスタ比較 4*
マスタタイマ比較 4 イベントは出力をアクティブ状態に強制します。

ビット 10 **MSTCMP3**：*マスタ比較 3*
マスタタイマ比較 3 イベントは出力をアクティブ状態に強制します。

ビット 9 **MSTCMP2**：*マスタ比較 2*
マスタタイマ比較 2 イベントは出力をアクティブ状態に強制します。

ビット 8 **MSTCMP1**：*マスタ比較 1*
マスタタイマ比較 1 イベントは出力をアクティブ状態に強制します。

ビット 7 **MSTPER**：*マスタ周期*
連続モードでのマスタタイマのカウンタのロールオーバー、またはシングルショットモードでのマスタタイマのリセットは、出力をアクティブ状態に強制します。

ビット 6 **CMP4**：*タイマ x の比較 4*
タイマ A 比較 4 イベントは出力をアクティブ状態に強制します。

ビット 5 **CMP3**：*タイマ x の比較 3*
タイマ A 比較 3 イベントは出力をアクティブ状態に強制します。

ビット 4 **CMP2**：*タイマ x の比較 2*
タイマ A 比較 2 イベントは出力をアクティブ状態に強制します。

ビット 3 **CMP1**：*タイマ x の比較 1*
タイマ A 比較 1 イベントは出力をアクティブ状態に強制します。

ビット 2 **PER**：*タイマ x 周期*
タイマ A 周期イベントは出力をアクティブ状態に強制します。

ビット 1 **RESYNC**：*タイマ A 再同期*
ソフトウェアまたは SYNC 入力からのみのタイマ A リセットイベントは出力をアクティブ状態に強制します。
注： RESYNC=1 のとき、その他のタイマリセットは出力に影響しません。

ビット 0 **SST**：*ソフトウェアセットトリガ*
このビットは出力をアクティブ状態に強制します。このビットは、ソフトウェアによってのみセットでき、ハードウェアによってリセットされます。
注： このビットはプリロードされません。

37.5.28 HRTIM タイマ x 出力 1 リセットレジスタ（HRTIM_RSTx1R）

アドレスオフセット : 0x40h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx1 出力を非アクティブ状態に強制できるソースを定義しています。

37.5.29 HRTIM タイマ x 出力 2 セットレジスタ（HRTIM_SETx2R）

アドレスオフセット : 0x44h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx2 出力をアクティブ状態に強制できるソースを定義しています。

37.5.30 HRTIM タイマ x 出力 2 リセットレジスタ（HRTIM_RSTx2R）

アドレスオフセット : 0x48h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx2 出力を非アクティブ状態に強制できるソースを定義しています。

37.5.31 HRTIM タイマ x 外部イベントフィルタリングレジスタ 1（HRTIM_EEFxR1）

アドレスオフセット : 0x4Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE5FLTR[3:0]				EE5LTCH	Res.	EE4FLTR[3:0]				EE4LTCH	Res.	EE3FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE3FLTR[2:0]			EE3LTCH	Res.	EE2FLTR[3:0]				EE2LTCH	Res.	EE1FLTR[3:0]				EE1LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **EE5FLTR[3:0]**：*外部イベント 5 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 24 **EE5LTCH**：*外部イベント 5 ラッチ*
EE1LTCH の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **EE4FLTR[3:0]**：*外部イベント 4 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 18 **EE4LTCH**：*外部イベント 4 ラッチ*
EE1LTCH の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:13 **EE3FLTR[3:0]**：*外部イベント 3 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 12 **EE3LTCH**：*外部イベント 3 ラッチ*
EE1LTCH の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **EE2FLTR[3:0]**：*外部イベント 2 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 6 **EE2LTCH**：*外部イベント 2 ラッチ*
EE1LTCH の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:1 **EE1FLTR[3:0]**：*外部イベント 1 フィルタ*

0000：フィルタリングなし
0001：カウンタのリセット／ロールオーバーから比較 1 までブランキング
0010：カウンタのリセット／ロールオーバーから比較 2 までブランキング
0011：カウンタのリセット／ロールオーバーから比較 3 までブランキング
0100：カウンタのリセット／ロールオーバーから比較 4 までブランキング
0101：別のタイミングユニットからブランキング：TIMFLTR1 ソース（詳細は*表 290* を参照）
0110：別のタイミングユニットからブランキング：TIMFLTR2 ソース（詳細は*表 290* を参照）
0111：別のタイミングユニットからブランキング：TIMFLTR3 ソース（詳細は*表 290* を参照）
1000：別のタイミングユニットからブランキング：TIMFLTR4 ソース（詳細は*表 290* を参照）
1001：別のタイミングユニットからブランキング：TIMFLTR5 ソース（詳細は*表 290* を参照）
1010：別のタイミングユニットからブランキング：TIMFLTR6 ソース（詳細は*表 290* を参照）
1011：別のタイミングユニットからブランキング：TIMFLTR7 ソース（詳細は*表 290* を参照）
1100：別のタイミングユニットからブランキング：TIMFLTR8 ソース（詳細は*表 290* を参照）
1101：カウンタのリセット／ロールオーバーから比較 2 までウィンドウイング
1110：カウンタのリセット／ロールオーバーから比較 3 までウィンドウイング
1111：別のタイミングユニットからウィンドウイング：TIMWIN ソース（詳細は*表 291* を参照）

注： ***比較レジスタがフィルタリングに使用される時は、その値は必ず 0 より大きくなければなりません。***
このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 0 **EE1LTCH**：*外部イベント 1 ラッチ*

0：イベント 1 は、ブランク中に発生した場合は無視され、ウィンドウ中では通過します。
1：イベント 1 はラッチされ、ブランキングまたはウィンドウイング期間の終りまで遅延されます。

注： ***ウィンドウモード（EE1FLTR[3:0]=1101、1110、1111）では、EE1LTCH = 0 の場合、タイムアウトイベントが生成されます。ただし、外部イベントが高速モードでプログラムされている（EExFAST = 1）場合は除きます。***
このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

37.5.32 HRTIM タイマ x 外部イベントフィルタリングレジスタ 2（HRTIM_EEFxR2）

アドレスオフセット : 0x50h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10FLTR[3:0]				EE10LTCH	Res.	EE9FLTR[3:0]				EE9LTCH	Res.	EE8FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8FLTR[2:0]			EE8LTCH	Res.	EE7FLTR[3:0]				EE7LTCH	Res.	EE6FLTR[3:0]				EE6LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **EE10FLTR[3:0]** : *外部イベント 10 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 24 **EE10LTCH** : *外部イベント 10 ラッチ*
EE1LTCH の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **EE9FLTR[3:0]** : *外部イベント 9 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 18 **EE9LTCH** : *外部イベント 9 ラッチ*
EE1LTCH の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:13 **EE8FLTR[3:0]** : *外部イベント 8 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 12 **EE8LTCH** : *外部イベント 8 ラッチ*
EE1LTCH の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **EE7FLTR[3:0]** : *外部イベント 7 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 6 **EE7LTCH** : *外部イベント 7 ラッチ*
EE1LTCH の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:1 **EE6FLTR[3:0]** : *外部イベント 6 フィルタ*
EE1FLTR[3:0] の説明を参照してください。

ビット 0 **EE6LTCH** : *外部イベント 6 ラッチ*
EE1LTCH の説明を参照してください。

37.5.33 HRTIM タイマ x リセットレジスタ（HRTIM_RSTxR）

HRTIM タイマ A リセットレジスタ（HRTIM_RSTAR）

アドレスオフセット：0xD4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **TECPM4**：*タイマ E の比較 4*
タイマ E の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 29 **TECMP2**：*タイマ E の比較 2*
タイマ E の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 28 **TECMP1**：*タイマ E の比較 1*
タイマ E の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 27 **TDCMP4**：*タイマ D の比較 4*
タイマ D の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 26 **TDCMP2**：*タイマ D の比較 2*
タイマ D の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 25 **TDCMP1**：*タイマ D の比較 1*
タイマ D の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 24 **TCCMP4**：*タイマ C の比較 4*
タイマ C の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 23 **TCCMP2**：*タイマ C の比較 2*
タイマ C の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 22 **TCCMP1**：*タイマ C の比較 1*
タイマ C の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 21 **TBCMP4**：*タイマ B の比較 4*
タイマ B の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 20 **TBCMP2**：*タイマ B の比較 2*
タイマ B の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 19 **TBCMP1**：*タイマ B の比較 1*
タイマ B の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 18 **EXTEVNT10**：*外部イベント*
外部イベント 10 の時にタイマ A カウンタがリセットされます。

ビット 17 **EXTEVNT9**：*外部イベント 9*
外部イベント 9 の時にタイマ A カウンタがリセットされます。

ビット 16 **EXTEVNT8**：*外部イベント 8*
外部イベント 8 の時にタイマ A カウンタがリセットされます。

ビット 15 **EXTEVNT7**：*外部イベント 7*
外部イベント 7 の時にタイマ A カウンタがリセットされます。

ビット 14 **EXTEVNT6**：*外部イベント 6*
外部イベント 6 の時にタイマ A カウンタがリセットされます。

ビット 13 **EXTEVNT5**：*外部イベント 5*
外部イベント 5 の時にタイマ A カウンタがリセットされます。

ビット 12 **EXTEVNT4**：*外部イベント 4*
外部イベント 4 の時にタイマ A カウンタがリセットされます。

ビット 11 **EXTEVNT3**：*外部イベント 3*
外部イベント 3 の時にタイマ A カウンタがリセットされます。

ビット 10 **EXTEVNT2**：*外部イベント 2*
外部イベント 2 の時にタイマ A カウンタがリセットされます。

ビット 9 **EXTEVNT1**：*外部イベント 1*
外部イベント 1 の時にタイマ A カウンタがリセットされます。

ビット 8 **MSTCMP4**：*マスタ比較 4*
マスタタイマの比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 7 **MSTCMP3**：*マスタ比較 3*
マスタタイマの比較 3 イベント時にタイマ A カウンタがリセットされます。

ビット 6 **MSTCMP2**：*マスタ比較 2*
マスタタイマの比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 5 **MSTCMP1**：*マスタ比較 1*
マスタタイマの比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 4 **MSTPER** *マスタタイマ周期*
マスタタイマの周期イベント時にタイマ A カウンタがリセットされます。

ビット 3 **CMP4**：*タイマ A の比較 4 リセット*
タイマ A の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 2 **CMP2**：*タイマ A の比較 2 リセット*
タイマ A の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 1 **UPDT**：*タイマ A 更新リセット*
タイマ A カウンタは、更新イベント時にリセットされます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

HRTIM タイマ B リセットレジスタ（HRTIM_RSTBR）

アドレスオフセット：0x154h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 30:1 HRTIM_RSTAR ビットの説明を参照してください。
ビット 30:19 は異なります（リセット信号は TIMA、TIMC、TIMD、および TIME から来ます）。

HRTIM タイマ C リセットレジスタ（HRTIM_RSTCR）

アドレスオフセット：0x1D4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 30:1 HRTIM_RSTAR ビットの説明を参照してください。
ビット 30:19 は異なります（リセット信号は TIMA、TIMB、TIMD、および TIME から来ます）。

HRTIM タイマ D リセットレジスタ（HRTIM_RSTDR）

アドレスオフセット：0x254h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 30:1 HRTIM_RSTAR ビットの説明を参照してください。
ビット 30:19 は異なります（リセット信号は TIMA、TIMB、TIMC、および TIME から来ます）。

HRTIM タイマ x リセットレジスタ（HRTIM_RSTER）

アドレスオフセット：0x2D4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXTEV NT10	EXTEV NT9	EXTEV NT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEV NT7	EXTEV NT6	EXTEV NT5	EXTEV NT4	EXTEV NT3	EXTEV NT2	EXTEV NT1	MSTC MP4	MSTC MP3	MSTC MP2	MSTC MP1	MSTPE R	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 30:1 HRTIM_RSTAR ビットの説明を参照してください。
ビット 30:19 は異なります（リセット信号は TIMA、TIMB、TIMC、および TIMD から来ます）。

37.5.34 HRTIM タイマ x チョッパレジスタ（HRTIM_CHPxR）

アドレスオフセット：0x58h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	STRTPW[3:0]				CARDTY[2:0]			CARFRQ[3:0]			
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **STRPW[3:0]**：*タイマ x 開始パルス幅*
このレジスタは、出力信号の立ち上がりエッジ後の最初のパルス幅を定義します。
このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
t_{1STPW} = （STRPW[3:0]+1）x 16 x t_{HRTIM}

0000：40 ns（1/25 MHz）
...
1111：640 ns（16/25 MHz）

ビット 6:4 **CARDTY[2:0]**：*タイマ x チョッパデューティサイクル値*
このレジスタは、キャリア信号のデューティサイクルを定義します。このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
000：0/8（すなわち、最初のパルスのみ存在します）
...
111：7/8

ビット 3:0 **CARFRQ[3:0]**：*タイマ x キャリア周波数の値*
このレジスタは、キャリア周波数 F_{CHPFRQ} = f_{HRTIM} /（16 x （CARFRQ[3:0]+1））を定義します。
このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
0000：25 MHz（f_{HRTIM}/ 16）
...
1111：1.56 MHz（f_{HRTIM} / 256）

37.5.35 HRTIM タイマ x キャプチャ 1 制御レジスタ（HRTIM_CPT1xCR）

アドレスオフセット : 0x5Ch（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み（TIME のみ）				予約済み（TIMD のみ）				予約済み（TIMC のみ）				予約済み（TIMB のみ）			
TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み（TIMA のみ）				EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
TACMP2	TACMP1	TA1RST	TA1SET												
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

37.5.36 HRTIM タイマ x キャプチャ 2 制御レジスタ（HRTIM_CPT2xCR）

アドレスオフセット : 0x60h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み（TIME のみ）				予約済み（TIMD のみ）				予約済み（TIMC のみ）				予約済み（TIMB のみ）			
TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み（TIMA のみ）				EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
TACMP2	TACMP1	TA1RST	TA1SET												
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TECMP2** : タイマ E の比較 2

TACMP1 の説明を参照してください。

注 : このビットはタイマ E では予約済みです。

ビット 30 **TECMP1** : タイマ E の比較 1

TACMP1 の説明を参照してください。

注 : このビットはタイマ E では予約済みです。

ビット 29 **TE1RST** : タイマ E の出力 1 リセット

TA1RST の説明を参照してください。

注 : このビットはタイマ E では予約済みです。

ビット 28 **TE1SET** : タイマ E の出力 1 セット

TA1SET の説明を参照してください。

注 : このビットはタイマ E では予約済みです。

ビット 27 **TDCMP2** : タイマ D の比較 2

TACMP1 の説明を参照してください。

注 : このビットはタイマ D では予約済みです。

ビット 26 **TDCMP1** : タイマ D の比較 1

TACMP1 の説明を参照してください。

注 : このビットはタイマ D では予約済みです。

ビット 25 **TD1RST** : タイマ D の出力 1 リセット

TA1RST の説明を参照してください。

注 : このビットはタイマ D では予約済みです。

ビット 24 **TD1SET** : タイマ D の出力 1 セット

TA1SET の説明を参照してください。

注 : このビットはタイマ D では予約済みです。

ビット 23 **TCCMP2** : タイマ C の比較 2

TACMP1 の説明を参照してください。

注 : このビットはタイマ C では予約済みです。

ビット 22 **TCCMP1**：タイマ C の比較 1
TACMP1 の説明を参照してください。
注： このビットはタイマ C では予約済みです。

ビット 21 **TC1RST**：タイマ C の出力 1 リセット
TA1RST の説明を参照してください。
注： このビットはタイマ C では予約済みです。

ビット 20 **TC1SET**：タイマ C の出力 1 セット
TA1SET の説明を参照してください。
注： このビットはタイマ C では予約済みです。

ビット 19 **TBCMP2**：タイマ B の比較 2
TACMP1 の説明を参照してください。
注： このビットはタイマ B では予約済みです。

ビット 18 **TBCMP1**：タイマ B の比較 1
TACMP1 の説明を参照してください。
注： このビットはタイマ B では予約済みです。

ビット 17 **TB1RST**：タイマ B の出力 1 リセット
TA1RST の説明を参照してください。
注： このビットはタイマ B では予約済みです。

ビット 16 **TB1SET**：タイマ B の出力 1 セット
TA1SET の説明を参照してください。
注： このビットはタイマ B では予約済みです。

ビット 15 **TACMP2**：タイマ A の比較 2
タイマ A 比較 2 はキャプチャ 2 をトリガします。
注： このビットはタイマ A では予約済みです。

ビット 14 **TACMP1**：タイマ A の比較 1
タイマ A 比較 1 はキャプチャ 2 をトリガします。
注： このビットはタイマ A では予約済みです。

ビット 13 **TA1RST**：タイマ B の出力 1 リセット
キャプチャ 2 は HRTIM_CHA1 出力のアクティブから非アクティブへの遷移によってトリガされます。
注： このビットはタイマ A では予約済みです。

ビット 12 **TA1SET**：タイマ B の出力 1 セット
キャプチャ 2 は HRTIM_CHA1 出力の非アクティブからアクティブへの遷移によってトリガされます。
注： このビットはタイマ A では予約済みです。

ビット 11 **EXEV10CPT**：*外部イベント 10 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 10 **EXEV9CPT**：*外部イベント 9 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 9 **EXEV8CPT**：*外部イベント 8 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 8 **EXEV7CPT**：*外部イベント 7 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 7 **EXEV6CPT**：*外部イベント 6 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 6 **EXEV5CPT**：*外部イベント 5 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 5 **EXEV4CPT**：*外部イベント 4 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 4 **EXEV3CPT**：*外部イベント 3 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 3 **EXEV2CPT**：*外部イベント 2 キャプチャ*
EXEV1CPT の説明を参照してください。

ビット 2 **EXEV1CPT**：*外部イベント 1 キャプチャ*
外部イベント 1 はキャプチャ 2 をトリガします。

ビット 1 **UPDCPT**：*更新キャプチャ*
更新イベントはキャプチャ 2 をトリガします。

ビット 0 **SWCPT**：*ソフトウェアキャプチャ*
このビットはソフトウェアによってキャプチャ 2 を強制します。このビットは、セットのみで、ハードウェアによってリセットされます。

37.5.37 HRTIM タイマ x 出力レジスタ（HRTIM_OUTxR）

アドレスオフセット：0x64h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM2	POL2	Res.
								rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DLYPRT[2:0]			DLYPRTEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM1	POL1	Res.
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **DIDL2**：*バーストモードアイドル移行時の出力 2 デッドタイム*

このビットは、出力をアイドル状態に切り替える前にデッドタイム挿入を強制することによって、アイドルモードへの移行を遅らせることができます。この設定は、バーストモード動作中にアイドル状態に移行する時にのみ適用されます。

0：プログラムされたアイドル状態は直ちに出力 2 に適用されます。

1：アイドルモードに移行する前に、デッドタイム（非アクティブレベル）が出力 2 に挿入されます。デッドタイム値は DTFx[8:0] によってセットされます。

注： ***このパラメータは、タイマ x が有効化された後は、変更できません。***
バーストモード中に出力の 1 つがアクティブで（IDLES=1）、正のデッドタイム（SDTR/SDTF が 0 にセットされている）の場合のみ、DIDL=1 をセットすることができます。

ビット 22 **CHP2**：*出力 2 チョッパイネーブル*

このビットは、出力 2 にチョッパを有効にします。

0：出力信号は変更されません。

1：出力信号はキャリア信号でチョップ処理されます。

注： ***このパラメータは、タイマ x が有効化された後は、変更できません。***

ビット 21:20 **FAULT2[1:0]**：*出力 2 フォールト状態*

これらのビットは、フォールトイベント後の出力 2 の状態を選択します。

00：作用なし：出力はフォールト入力に影響されず、動作モードに留まります。

01：アクティブ

10：インアクティブ

11：ハイインピーダンス

注： ***FLTENx ビットがセットされている場合、または出力が FAULT 状態の場合は、このパラメータは、いったんタイマ x が有効になると（TxCEN ビットのセット）、変更できません。***

ビット 19 **IDLES2**：*出力 2 アイドル状態*

このビットは出力 2 のアイドル状態を選択します。

0：インアクティブ

1：アクティブ

注： ***このパラメータは、HRTIM が出力を制御する前にセットする必要があります。***

ビット 18 **IDLEM2**：*出力 2 アイドルモード*

このビットは出力 2 のアイドルモードを選択します。

0：作用なし：出力はバーストモード動作の影響を受けません。

1：バーストモードコントローラによってリクエストされると、出力はアイドル状態になります。

注： ***このビットはプリロードされ、実行中に変更できますが、バーストモードがアクティブである間は、変更してはなりません。***

ビット 17 **POL2**：*出力 2 極性*

このビットは出力 2 の極性を選択します。

0：正の極性（出力はアクティブハイ）

1：負の極性（出力はアクティブロー）

注：　このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 16:12 予約済みであり、リセット値に保持する必要があります。

ビット 12:10 **DLYPRT[2:0]**：*遅延保護*

これらのビットは、遅延保護スキームを適用するソースおよび出力を定義します。

HRTIM_OUTAR、HRTIM_OUTBR、HRTIM_OUTCR において：

000：外部ベント 6 による出力 1 遅延アイドル

001：外部ベント 6 による出力 2 遅延アイドル

010：外部ベント 6 による出力 1 および出力 2 遅延アイドル

011：外部ベント 6 によるバランスアイドル

100：外部ベント 7 による出力 1 遅延アイドル

101：外部ベント 7 による出力 2 遅延アイドル

110：外部ベント 7 による出力 1 および出力 2 遅延アイドル

111：外部ベント 7 によるバランスアイドル

HRTIM_OUTDR、HRTIM_OUTER において：

000：外部ベント 8 による出力 1 遅延アイドル

001：外部ベント 8 による出力 2 遅延アイドル

010：外部ベント 8 による出力 1 および出力 2 遅延アイドル

011：外部ベント 8 によるバランスアイドル

100：外部ベント 9 による出力 1 遅延アイドル

101：外部ベント 9 による出力 2 遅延アイドル

110：外部ベント 9 による出力 1 および出力 2 遅延アイドル

111：外部ベント 9 によるバランスアイドル

注：　このビットフィールドは、遅延保護が有効化（DLYPRTEN ビットのセット）された後には変更してはなりません。

ビット 9 **DLYPRTEN**：*遅延保護イネーブル*

このビットは、遅延保護スキームを有効にします。

0：影響なし。

1：DLYPRT[2:0] ビットによって遅延保護が有効化されます。

注：　このパラメータは、タイマ x が有効化（TxEN ビットのセット）された後は、変更できません。

ビット 8 **DTEN**：*デッドタイムイネーブル*

このビットは、出力 1 と出力 2 へのデッドタイム挿入を有効化します。

0：出力 1 と出力 2 の信号は独立です。

1：デッドタイムが出力 1 と出力 2 の間に挿入されます（基準信号は出力 1 の信号ジェネレータです）。

注：　このパラメータは、いったんタイマが動作すると（TxEN ビットのセット）、またはその出力が有効化され、別のタイマによってセット／リセットされる場合は、変更できません。

ビット 7 **DIDL1**：*バーストモードアイドル移行時の出力 1 デッドタイム*

このビットは、出力をアイドル状態に切り替える前にデッドタイム挿入を強制することによって、アイドルモードへの移行を遅らせることができます。この設定は、バーストモード動作中にアイドル状態に移行する時にのみ適用されます。

0：プログラムされたアイドル状態は直ちに出力 1 に適用されます。

1：アイドルモードに移行する前に、デッドタイム（非アクティブレベル）が出力 1 に挿入されます。デッドタイム値は DTRx[8:0] によってセットされます。

注：　このパラメータは、タイマ x が有効化された後は、変更できません。

バーストモード中に出力の 1 つがアクティブで（IDLES=1）、正のデッドタイム（SDTR/SDTF が 0 にセットされている）の場合のみ、DIDL=1 をセットすることができます。

ビット 6 **CHP1**：*出力 1 チョッパイネーブル*

このビットは、出力 1 にチョッパを有効にします。

0：出力信号は変更されません。

1：出力信号はキャリア信号でチョップ処理されます。

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 5:4 **FAULT1[1:0]**：*出力 1 フォールト状態*

これらのビットは、フォールトイベント後の出力 1 の状態を選択します。

00：作用なし：出力はフォールト入力に影響されず、動作モードに留まります。

01：アクティブ

10：インアクティブ

11：ハイインピーダンス

注： FLTENx ビットがセットされている場合、または出力が FAULT 状態の場合は、このパラメータは、いったんタイマ x が有効になると（TxCEN ビットのセット）、変更できません。

ビット 3 **IDLES1**：*出力 1 アイドル状態*

このビットは出力 1 のアイドル状態を選択します。

0：インアクティブ

1：アクティブ

注： このパラメータは、HRTIM が出力を制御する前にセットする必要があります。

ビット 2 **IDLEM1**：*出力 1 アイドルモード*

このビットは出力 1 のアイドルモードを選択します。

0：作用なし：出力はバーストモード動作の影響を受けません。

1：バーストモードコントローラによってリクエストされると、出力はアイドル状態になります。

注： このビットはプリロードされ、実行中に変更できますが、バーストモードがアクティブである間は、変更してはなりません。

ビット 1 **POL1**：*出力 1 極性*

このビットは出力 1 の極性を選択します。

0：正の極性（出力はアクティブハイ）

1：負の極性（出力はアクティブロー）

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 0 予約済み

37.5.38 HRTIM タイマ x フォールトレジスタ（HRTIM_FLTxR）

アドレスオフセット：0x68h（このオフセットアドレスはタイマ x ベースアドレスに対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLTLCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rwo															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5EN	FLT4EN	FLT3EN	FLT2EN	FLT1EN
											rw	rw	rw	rw	rw

ビット 31 **FLTLCK**：*フォールトソースロック*

0：FLT1EN～FLT5EN ビットは読み出し／書き込み用です。

1：FLT1EN～FLT5EN ビットは読み出し専用です。

FLTLCK ビットは 1 回のみ書き込み可能です。いったんセットされると、次のシステムリセットまで変更できません。

ビット 30:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **FLT5EN**：*フォールト 5 イネーブル*

0：フォールト 5 入力は無視されます。

1：フォールト 5 入力はアクティブで、HRTIM 出力を無効にできます。

ビット 3 **FLT4EN**：*フォールト 4 イネーブル*

0：フォールト 4 入力は無視されます。

1：フォールト 4 入力はアクティブで、HRTIM 出力を無効にできます。

ビット 2 **FLT3EN**：*フォールト 3 イネーブル*

0：フォールト 3 入力は無視されます。

1：フォールト 3 入力はアクティブで、HRTIM 出力を無効にできます。

ビット 1 **FLT2EN**：*フォールト 2 イネーブル*

0：フォールト 2 入力は無視されます。

1：フォールト 2 入力はアクティブで、HRTIM 出力を無効にできます。

ビット 0 **FLT1EN**：*フォールト 1 イネーブル*

0：フォールト 1 入力は無視されます。

1：フォールト 1 入力はアクティブで、HRTIM 出力を無効にできます。

37.5.39 HRTIM 制御レジスタ 1（HRTIM_CR1）

アドレスオフセット：0x380h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AD4USRC[2:0]			AD3USRC[2:0]			AD2USRC[2:0]			AD1USRC[2:0]		
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEUDIS	TDUDIS	TCUDIS	TBUDIS	TAUDIS	MUDIS
										rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:25 **AD4USRC[2:0]**：*ADC トリガ 4 更新ソース*
AD1USRC[2:0] の説明を参照してください。

ビット 24:22 **AD3USRC[2:0]**：*ADC トリガ 3 更新ソース*
AD1USRC[2:0] の説明を参照してください。

ビット 21:19 **AD2USRC[2:0]**：*ADC トリガ 2 更新ソース*
AD1USRC[2:0] の説明を参照してください。

ビット 18:16 **AD1USRC[2:0]**：*ADC トリガ 1 更新ソース*
これらのビットは、HRTIM_ADC1R レジスタの更新（プリロードからアクティブレジスタへ転送）をトリガするソースを定義します。ソースタイマのみを定義します。正確な条件はタイマ自身の中（HRTIM_MCR または HRTIM_TIMxCR）で定義されます。
000：マスタタイマ
001：タイマ A
010：タイマ B
011：タイマ C
100：タイマ D
101：タイマ E
110, 111：予約済み

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TEUDIS**：*タイマ E 更新ディスエーブル*
TAUDIS の説明を参照してください。

ビット 4 **TDUDIS**：*タイマ D 更新ディスエーブル*
TAUDIS の説明を参照してください。

ビット 3 **TCUDIS**：*タイマ C 更新ディスエーブル*
TAUDIS の説明を参照してください。

ビット 2 **TBUDIS**：*タイマ B 更新ディスエーブル*

TAUDIS の説明を参照してください。

ビット 1 **TAUDIS**：*タイマ A 更新ディスエーブル*

このビットは、タイマ A で一時的に更新イベントの生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新は有効です。更新は、選択されたソースの生成時に起きます。

1：更新は無効です。同時に考慮する必要のある複数のレジスタにソフトウェアが書き込めるように、更新は一時的に無効にされます。

ビット 0 **MUDIS**：マスタ *更新ディスエーブル*

このビットは、一時的に更新イベントの生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新は有効です。

1：更新は無効です。同時に考慮する必要のある複数のレジスタにソフトウェアが書き込めるように、更新は一時的に無効にされます。

37.5.40 HRTIM 制御レジスタ 2（HRTIM_CR2）

アドレスオフセット：0x384h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	Res.	TESWU	TDSWU	TCSWU	TBSWU	TASWU	MSWU
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **TERST**：*タイマ E カウンタのソフトウェアリセット*
TARST の説明を参照してください。

ビット 12 **TDRST**：*タイマ D カウンタのソフトウェアリセット*
TARST の説明を参照してください。

ビット 11 **TCRST**：*タイマ C カウンタのソフトウェアリセット*
TARST の説明を参照してください。

ビット 10 **TBRST**：*タイマ B カウンタのソフトウェアリセット*
TARST の説明を参照してください。

ビット 9 **TARST**：*タイマ A カウンタのソフトウェアリセット*
このビットをセットすると、タイマ A カウンタがリセットされます。
このビットは、ハードウェアによって自動的にリセットされます。

ビット 8 **MRST**：マスタ *カウンタのソフトウェアリセット*
このビットをセットすると、マスタタイマのカウンタがリセットされます。
このビットは、ハードウェアによって自動的にリセットされます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TESWU**：*タイマ E ソフトウェア更新*
TASWU の説明を参照してください。

ビット 4 **TDSWU**：*タイマ D ソフトウェア更新*
TASWU の説明を参照してください。

ビット 3 **TCSWU**：*タイマ C ソフトウェア更新*
TASWU の説明を参照してください。

ビット 2 **TBSWU**：*タイマ B ソフトウェア更新*
TASWU の説明を参照してください。

ビット 1 **TASWU**：*タイマ A ソフトウェア更新*
このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。これはプリロードからアクティブレジスタへの転送を直ちに強制し、保留中の更新リクエストはキャンセルされます。

ビット 0 **MSWU**：*マスタタイマソフトウェア更新*
このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。これはマスタタイマのプリロードからアクティブレジスタへの転送を直ちに強制し、保留中の更新リクエストはキャンセルされます。

37.5.41 HRTIM 割り込みステータスレジスタ（HRTIM_ISR）

アドレスオフセット：0x388h

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
											r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPER**：バーストモード周期割り込みフラグ

このビットは、シングルショットバーストモード動作が完了した時、または連続モードでのバーストモード周期の終りに、ハードウェアによってセットされます。HRTIM_ICRレジスタの対応するビットに1をソフトウェアで書き込むことによってクリアされます。

0：バーストモード周期割り込みは起きていません。

1：バーストモード周期割り込みが起きました。

ビット 16:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SYSFLT**：システムフォールト割り込みフラグ

FLT1 の説明を参照してください。

ビット 4 **FLT5**：フォールト 5 割り込みフラグ

FLT1 の説明を参照してください。

ビット 3 **FLT4**：フォールト 4 割り込みフラグ

FLT1 の説明を参照してください。

ビット 2 **FLT3**：フォールト 3 割り込みフラグ

FLT1 の説明を参照してください。

ビット 1 **FLT2**：フォールト 2 割り込みフラグ

FLT1 の説明を参照してください。

ビット 0 **FLT1**：フォールト 1 割り込みフラグ

このビットは、フォールト 1 イベントが発生したときに、ハードウェアによってセットされます。HRTIM_ICR レジスタの対応するビットに1をソフトウェアで書き込むことによってクリアされます。

0：フォールト 1 割り込みは起きていません。

1：フォールト 1 割り込みが起きました。

37.5.42 HRTIM 割り込みクリアスレジスタ（HRTIM_ICR）

アドレスオフセット：0x38Ch

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	Res.
														w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
											w	w	w	w	w

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPERC**：バーストモード周期フラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの BMPER フラグがクリアされます。

ビット 16:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SYSFLTC**：システムフォールト割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの SYSFLT フラグがクリアされます。

ビット 4 **FLT5C**：フォールト 5 割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT5 フラグがクリアされます。

ビット 3 **FLT4C**：フォールト 4 割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT4 フラグがクリアされます。

ビット 2 **FLT3C**：フォールト 3 割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT3 フラグがクリアされます。

ビット 1 **FLT2C**：フォールト 2 割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT2 フラグがクリアされます。

ビット 0 **FLT1C**：フォールト 1 割り込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT1 フラグがクリアされます。

37.5.43 HRTIM 割り込み有効レジスタ（HRTIM_IER）

アドレスオフセット：0x390h

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
											rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPERIE**：バーストモード周期割り込みイネーブル

このビットは、バーストモード周期割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：バーストモード周期割り込みは無効です。

1：バーストモード周期割り込みは有効です。

ビット 16:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SYSFLTIE**：システムフォールト割り込みイネーブル

FLT1IE の説明を参照してください。

ビット 4 **FLT5IE**：フォールト 5 割り込みイネーブル

FLT1IE の説明を参照してください。

ビット 3 **FLT4IE**：フォールト 4 割り込みイネーブル

FLT1IE の説明を参照してください。

ビット 2 **FLT3IE**：フォールト 3 割り込みイネーブル

FLT1IE の説明を参照してください。

ビット 1 **FLT2IE**：フォールト 2 割り込みイネーブル

FLT1IE の説明を参照してください。

ビット 0 **FLT1IE**：フォールト 1 割り込みイネーブル

このビットは、フォールト 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：フォールト 1 割り込みは無効です。

1：フォールト 1 割り込みは有効です。

37.5.44 HRTIM 出力有効レジスタ（HRTIM_OENR）

アドレスオフセット：0x394h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
						rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **TE2OEN**：タイマ E の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 8 **TE1OEN**：タイマ E の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 7 **TD2OEN**：タイマ D の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 6 **TD1OEN**：タイマ D の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 5 **TC2OEN**：タイマ C の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 4 **TC1OEN**：タイマ C の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 3 **TB2OEN**：タイマ B の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 2 **TB1OEN**：タイマ B の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 1 **TA2OEN**：タイマ A の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 0 **TA1OEN**：タイマ A の出力 1（HRTIM_CHA1）イネーブル
このビットをセットすると、タイマ A 出力 1 が有効になります。“0”を書き込んでも、ビットの値は変化しません。
ビットを読み出すと、出力の有効／無効ステータスが返されます。
このビットは、タイマ関連のフォールト入力がアクティブになるとすぐに、ハードウェアによって非同期にクリアされます。
0：出力 HRTIM_CHA1 は無効です。出力はフォールトまたはアイドル状態です。
1：出力 HRTIM_CHA1 は有効です。

注： 無効ステータスはアイドルとフォールト状態の両方に対応します。出力無効ステータスは HRTIM_ODSR レジスタの TA1ODS ビットによって得られます。

37.5.45 HRTIM 出力無効レジスタ（HRTIM_ODISR）

アドレスオフセット：0x398h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2ODIS	TE1ODIS	TD2ODIS	TD1ODIS	TC2ODIS	TC1ODIS	TB2ODIS	TB1ODIS	TA2ODIS	TA1ODIS
						w	w	w	w	w	w	w	w	w	w

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **TE2ODIS**：タイマ E の出力 2 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 8 **TE1ODIS**：タイマ E の出力 1 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 7 **TD2ODIS**：タイマ D の出力 2 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 6 **TD1ODIS**：タイマ D の出力 1 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 5 **TC2ODIS**：タイマ C の出力 2 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 4 **TC1ODIS**：タイマ C の出力 1 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 3 **TB2ODIS**：タイマ A の出力 2 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 2 **TB1ODIS**：タイマ A の出力 1 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 1 **TA2ODIS**：タイマ A の出力 2 ディスエーブル
TA1ODIS の説明を参照してください。

ビット 0 **TA1ODIS**：タイマ A の出力 1（HRTIM_CHA1）ディスエーブル
このビットをセットすると、タイマ A 出力 1 が無効になります。出力は、実行状態またはフォールト状態から、アイドル状態に移行します。
“0”を書き込んでも、ビットの値は変化しません。

37.5.46 HRTIM 出力無効ステータスレジスタ（HRTIM_ODSR）

アドレスオフセット：0x39Ch

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
						r	r	r	r	r	r	r	r	r	r

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **TE2ODS**：タイマ E の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 8 **TE1ODS**：タイマ E の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 7 **TD2ODS**：タイマ D の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 6 **TD1ODS**：タイマ D の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 5 **TC2ODS**：タイマ C の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 4 **TC1ODS**：タイマ C の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 3 **TB2ODS**：タイマ B の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 2 **TB1ODS**：タイマ B の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 1 **TA2ODS**：タイマ A の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 0 **TA1ODS**：タイマ A の出力 1 無効ステータス
ビットを読み出すと、出力の無効ステータスが返されます。出力がアクティブ（Tx1OEN または Tx2OEN = 1）のとき、これは意味がありません。
0：出力 HRTIM_CHA1 はアイドル状態で、無効です。
1：出力 HRTIM_CHA1 はフォールト状態で、無効です。

37.5.47 HRTIM バーストモード制御レジスタ（HRTIM_BMCR）

アドレスオフセット：0x3A0h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEBM	TDBM	TCBM	TBBM	TABM	MTBM
rc_w0										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **BMSTAT**：*バーストモードステータス*

このビットは、現在の動作状態を示します。

0：通常動作

1：バースト動作が進行中です。このビットに 0 を書き込むと、バーストモードが早期に終了します。

ビット 30:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TEBM**：*タイマ E バーストモード*

TABM の説明を参照してください。

ビット 20 **TDBM**：*タイマ D バーストモード*

TABM の説明を参照してください。

ビット 19 **TCBM**：*タイマ C バーストモード*

TABM の説明を参照してください。

ビット 18 **TBBM**：*タイマ B バーストモード*

TABM の説明を参照してください。

ビット 17 **TABM**：*タイマ A バーストモード*

このビットは、バーストモード動作中にタイマがどのように動作するかを指定します。このビットは、バーストモードが有効である間は、変更できません。

0：タイマ A のカウンタクロックは維持され、タイマは正常に動作します。

1：タイマ A のカウンタクロックは停止され、カウンタはリセットされます。

注： このビットは、バランスアイドルモードがアクティブなときは（DLYPRT[2:0] = 0x11）、セットしてはいけません。

ビット 16 **MTBM**：*マスタタイマバーストモード*

このビットは、バーストモード動作中にタイマがどのように動作するかを指定します。このビットは、バーストモードが有効である間は、変更できません。

0：マスタタイマのカウンタクロックは維持され、タイマは正常に動作します。

1：マスタタイマのカウンタクロックは停止され、カウンタはリセットされます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **BMPREN**：*バーストモードプリロードイネーブル*

このビットは、レジスタプリロードメカニズムを有効にし、プリロード可能なレジスタ（HRTIM_BMCMPR、HRTIM_BMPER）への書き込みアクセスがアクティブレジスタに行われるか、プリロードレジスタに行われるかを定義します。

0：プリロード無効：書き込みアクセスは直接、アクティブレジスタに行われます。

1：プリロード有効：書き込みアクセスはプリロードレジスタに行われます。

ビット 9:6 **BMPRSC[3:0]**：*バーストモードプリスケーラ*

バーストモードコントローラ用の、f_{HRTIM} クロックのプリスケーリング比を定義します。このビットは、バーストモードが有効である間は、変更できません。

0000：クロックは分周されません。
0001：分周比 2
0010：分周比 4
0011：分周比 8
0100：分周比 16
0101：分周比 32
0110：分周比 64
0111：分周比 128
1000：分周比 256
1001：分周比 512
1010：分周比 1024
1011：分周比 2048
1100：分周比 4096
1101：分周比 8192
1110：分周比 16384
1111：分周比 32768

ビット 5:2 **BMCLK[3:0]**：*バーストモードクロックソース*

このビットフィールドは、バーストモードカウンタ用のクロックソースを定義します。これは、バーストモードが有効である間は、変更できません（オンチップイベント 1～4 の接続の詳細は*表 300* を参照）。

0000：マスタタイマのカウンタリセット／ロールオーバー
0001：タイマ A カウンタリセット／ロールオーバー
0010：タイマ B カウンタリセット／ロールオーバー
0011：タイマ C カウンタリセット／ロールオーバー
0100：タイマ D カウンタリセット／ロールオーバー
0101：タイマ E カウンタリセット／ロールオーバー
0110：バーストモードカウンタクロックとして動作するオンチップイベント 1（hrtim_bm_ck1）
0111：バーストモードカウンタクロックとして動作するオンチップイベント 2（hrtim_bm_ck2）
1000：バーストモードカウンタクロックとして動作するオンチップイベント 3（hrtim_bm_ck3）
1001：バーストモードカウンタクロックとして動作するオンチップイベント 4（hrtim_bm_ck4）
1010：プリスケールされた f_{HRTIM} クロック（BMPRSC[3:0] の設定による）
その他のコードは予約済み

ビット 1 **BMOM**：*バーストモード動作モード*

このビットは、バーストモードに一度だけ移行するのか、連続的に動作するのかを決定します。

0：シングルショットモード
1：連続動作

ビット 0 **BME**：*バーストモードイネーブル*

このビットは、開始トリガを受信できる準備ができたバーストモードコントローラを開始します。
このビットに 0 を書き込むと、バーストモードが早期に終了します。

0：バーストモードは無効です。
1：バーストモードは有効です。

37.5.48 HRTIM バーストモードトリガレジスタ（HRTIM_BMTRGR）

アドレスオフセット：0x3A4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OCHPEV	EEV8	EEV7	TDEEV8	TAEEV7	TECMP2	TECMP1	TEREP	TERST	TDCMP2	TDCMP1	TDREP	TDRST	TCCMP2	TCCMP1	TCREP
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **OCHPEV**：*オンチップイベント*
オンチップイベントの立ち上がりエッジ（*セクション：バーストモードトリガ*を参照）がバーストモードへの移行をトリガします。

ビット 30 **EEV8**：*外部イベント 8（TIMD フィルタ適用）*
TIMD フィルタによって調整された外部イベント 8 がバーストモード動作を開始します。

ビット 29 **EEV7**：*外部イベント 7（TIMA フィルタ適用）*
TIMA フィルタによって調整された外部イベント 7 がバーストモード動作を開始します。

ビット 28 **TDEEV8**：*外部イベント 8 の後のタイマ D 周期*
（TIMD フィルタによって調整された）外部イベント 8 の後のタイマ D 周期がバーストモード動作を開始します。

ビット 27 **TAEEV7**：*外部イベント 7 の後のタイマ A 周期*
（TIMA フィルタによって調整された）外部イベント 7 の後のタイマ A 周期がバーストモード動作を開始します。

ビット 26 **TECMP2**：*タイマ E の比較 2 イベント*
TACMP1 の説明を参照してください。

ビット 25 **TECMP1**：*タイマ E の比較 1 イベント*
TACMP1 の説明を参照してください。

ビット 24 **TEREP**：*タイマ E 繰り返し*
TAREP の説明を参照してください。

ビット 23 **TERST**：*タイマ E カウンタリセットまたはロールオーバー*
TARST の説明を参照してください。

ビット 22 **TDCMP2**：*タイマ D の比較 2 イベント*
TACMP1 の説明を参照してください。

ビット 21 **TDCMP1**：*タイマ D の比較 1 イベント*
TACMP1 の説明を参照してください。

ビット 20 **TDREP**：*タイマ D 繰り返し*
TAREP の説明を参照してください。

ビット 19 **TDRST**：*タイマ D リセットまたはロールオーバー*
TARST の説明を参照してください。

ビット 18 **TCCMP2**：*タイマ C の比較 2 イベント*
TACMP1 の説明を参照してください。

ビット 17 **TCCMP1**：*タイマ C の比較 1 イベント*
TACMP1 の説明を参照してください。

ビット 16 **TCREP**：*タイマ C 繰り返し*
TAREP の説明を参照してください。

ビット 15 **TCRST**：*タイマ C リセットまたはロールオーバー*
TARST の説明を参照してください。

ビット 14 **TBCMP2**：*タイマ B の比較 2 イベント*
TACMP1 の説明を参照してください。

ビット 13 **TBCMP1**：*タイマ B の比較 1 イベント*
TACMP1 の説明を参照してください。

ビット 12 **TBREP**：*タイマ B 繰り返し*
TAREP の説明を参照してください。

ビット 11 **TBRST**：*タイマ B リセットまたはロールオーバー*
TARST の説明を参照してください。

ビット 10 **TACMP2**：*タイマ A の比較 2 イベント*
TACMP1 の説明を参照してください。

ビット 9 **TACMP1**：*タイマ A の比較 1 イベント*
タイマ A の比較 1 イベントがバーストモード動作を開始します。

ビット 8 **TAREP**：*タイマ A 繰り返し*
タイマ A の繰り返しイベントがバーストモード動作を開始します。

ビット 7 **TARST**：*タイマ A リセットまたはロールオーバー*
タイマ A のリセットまたはロールオーバーがバーストモード動作を開始します。

ビット 6 **MSTCMP4**：*マスタ比較 4*
MSTCMP1 の説明を参照してください。

ビット 5 **MSTCMP3**：*マスタ比較 3*
MSTCMP1 の説明を参照してください。

ビット 4 **MSTCMP2**：*マスタ比較 2*
MSTCMP1 の説明を参照してください。

ビット 3 **MSTCMP1**：*マスタ比較 1*
マスタタイマの比較 1 イベントがバーストモード動作を開始します。

ビット 2 **MSTREP**：*マスタ繰り返し*
マスタタイマの繰り返しイベントがバーストモード動作を開始します。

ビット 1 **MSTRST**：*マスタのリセットまたはロールオーバー*
マスタタイマのリセットおよびロールオーバーイベントがバーストモード動作を開始します。

ビット 0 **SW**：*ソフトウェア開始*
このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。
セットされると、直ちにバーストモード動作を開始します。
このビットは、バーストモードが有効でない場合は（BME ビットがリセット）、アクティブではありません。

37.5.49 HRTIM バーストモード比較レジスタ（HRTIM_BMCMPR）

アドレスオフセット：0x3A8h

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BMCMP[15:0]**：*バーストモード比較値*

選択されたタイマがアイドル状態にある周期の数を定義します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

注： ***バーストモードクロックのソースとしてプリスケーラなしの f_{HRTIM} クロックを使用している場合（BMCLK[3:0] = 1010 および BMPRESC[3:0] = 0000）は、BMCMP[15:0] は 0x0000 に設定することはできません。***

37.5.50 HRTIM バーストモード周期レジスタ（HRTIM_BMPER）

アドレスオフセット：0x3ACh

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BMPER[15:0]**：*バーストモード周期*

バーストモード繰り返し周期を定義します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

注： ***BMPER[15:0] は、バーストモードが有効であるときは、null ではいけません。***

37.5.51 HRTIM タイマ外部イベント制御レジスタ 1（HRTIM_EECR1）

アドレスオフセット：0x3B0h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1]
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE3SNS[0]	EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **EE5FAST**：*外部イベント 5 高速モード*
EE1FAST の説明を参照してください。

ビット 28:27 **EE5SNS[1:0]**：*外部イベント 5 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 26 **EE5POL**：*外部イベント 5 極性*
EE1POL の説明を参照してください。

ビット 25:24 **EE5SRC[1:0]**：*外部イベント 5 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 23 **EE4FAST**：*外部イベント 4 高速モード*
EE1FAST の説明を参照してください。

ビット 22:21 **EE4SNS[1:0]**：*外部イベント 4 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 20 **EE4POL**：*外部イベント 4 極性*
EE1POL の説明を参照してください。

ビット 19:18 **EE4SRC[1:0]**：*外部イベント 4 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 17 **EE3FAST**：*外部イベント 3 高速モード*
EE1FAST の説明を参照してください。

ビット 16:15 **EE3SNS[1:0]**：*外部イベント 3 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 14 **EE3POL**：*外部イベント 3 極性*
EE1POL の説明を参照してください。

ビット 13:12 **EE3SRC[1:0]**：*外部イベント 3 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 11 **EE2FAST**：*外部イベント 2 高速モード*
EE1FAST の説明を参照してください。

ビット 10:9 **EE2SNS[1:0]**：*外部イベント 2 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 8 **EE2POL**：*外部イベント 2 極性*
EE1POL の説明を参照してください。

ビット 7:6 **EE2SRC[1:0]**：*外部イベント 2 ソース*

EE1SRC[1:0] の説明を参照してください。

ビット 5 **EE1FAST**：*外部イベント 1 高速モード*

0：外部イベント 1 は、出力に作用する前に HRTIM のロジックによって再同期され、f_{HRTIM} クロックに関連する遅延時間が追加されます。

1：外部イベント 1 は非同期で出力に作用します（低遅延時間モード）

注：　このビットは、イベントが使用されるカウンタが有効化（TxCEN　ビットのセット）された後には変更してはなりません。

ビット 4:3 **EE1SNS[1:0]**：*外部イベント 1 感度*

00：EE1POL ビットで定義されたアクティブレベルが有効

01：EE1POL ビット値が何であれ、立ち上がりエッジ

10：EE1POL ビット値が何であれ、立ち下がりエッジ

11：EE1POL ビット値が何であれ、両エッジ

ビット 2 **EE1POL**：*外部イベント 1 極性*

このビットは、EE1SNS[1:0] = 00 の場合にのみ、意味があります。

0：外部イベントはアクティブハイです。

1：外部イベントはアクティブローです。

注：　このパラメータは、タイマ x が有効化された後は、変更できません。EE1FAST ビットをセットする前に設定する必要があります。

ビット 1:0 **EE1SRC[1:0]**：*外部イベント 1 ソース*

00：hrtim_evt11

01：hrtim_evt12

10：hrtim_evt13

11：hrtim_evt14

注：　このパラメータは、タイマ x が有効化された後は、変更できません。EE1FAST ビットをセットする前に設定する必要があります。

37.5.52 HRTIM タイマ外部イベント制御レジスタ 2（HRTIM_EECR2）

アドレスオフセット：0x3B4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS[1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS[1]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8SNS[0]	EE8POL	EE8SRC[1:0]		Res.	EE7SNS[1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS[1:0]		EE6POL	EE6SRC[1:0]	
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:27 **EE10SNS[1:0]**：*外部イベント 10 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 26 **EE10POL**：*外部イベント 10 極性*
EE1POL の説明を参照してください。

ビット 25:24 **EE10SRC[1:0]**：*外部イベント 10 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **EE9SNS[1:0]**：*外部イベント 9 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 20 **EE9POL**：*外部イベント 9 極性*
EE1POL の説明を参照してください。

ビット 19:18 **EE9SRC[1:0]**：*外部イベント 9 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:15 **EE8SNS[1:0]**：*外部イベント 8 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 14 **EE8POL**：*外部イベント 8 極性*
EE1POL の説明を参照してください。

ビット 13:12 **EE8SRC[1:0]**：*外部イベント 8 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:9 **EE7SNS[1:0]**：*外部イベント 7 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 8 **EE7POL**：*外部イベント 7 極性*
EE1POL の説明を参照してください。

ビット 7:6 **EE7SRC[1:0]**：*外部イベント 7 ソース*
EE1SRC[1:0] の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **EE6SNS[1:0]**：*外部イベント 6 感度*
EE1SNS[1:0] の説明を参照してください。

ビット 2 **EE6POL**：*外部イベント 6 極性*
EE1POL の説明を参照してください。

ビット 1:0 **EE6SRC[1:0]**：*外部イベント 6 ソース*
EE1SRC[1:0] の説明を参照してください。

37.5.53 HRTIM タイマ外部イベント制御レジスタ 3（HRTIM_EECR3）

アドレスオフセット：0x3B8h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EEVSD[1:0]		Res.	Res.	EE10F[3:0]				Res.	Res.	EE9F[3:0]				Res.	Res.
rw	rw			rw	rw	rw	rw			rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8F[3:0]				Res.	Res.	EE7F[3:0]				Res.	Res.	EE6F[3:0]			
rw	rw	rw	rw			rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:30 **EEVSD[1:0]**：外部イベントサンプリングクロック分周比
このビットフィールドは、タイマクロック周波数（f_{HRTIM}）と
デジタルフィルタに使用される外部イベント信号サンプリングクロック（f_{EEVS}）との間の分周比を示します。
00：$f_{EEVS}=f_{HRTIM}$
01：$f_{EEVS}=f_{HRTIM}$ / 2
10：$f_{EEVS}=f_{HRTIM}$ / 4
11：$f_{EEVS}=f_{HRTIM}$ / 8

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **EE10F[3:0]**：外部イベント 10 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:18 **EE9F[3:0]**：外部イベント 9 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EE8F[3:0]**：外部イベント 8 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:6 **EE7F[3:0]**：外部イベント 7 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 4:5 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **EE6F[3:0]**：外部イベント 6 フィルタ
このビットフィールドは、外部イベント 6 入力をサンプリングするのに使用される周波数と、hrtim_evt6 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。
0000：フィルタ無効
0001：$f_{SAMPLING}= f_{HRTIM}$、N = 2
0010：$f_{SAMPLING}= f_{HRTIM}$、N = 4
0011：$f_{SAMPLING}= f_{HRTIM}$、N = 8
0100：$f_{SAMPLING}= f_{EEVS}/2$、N=6
0101：$f_{SAMPLING}= f_{EEVS}/2$、N=8
0110：$f_{SAMPLING}= f_{EEVS}/4$、N=6
0111：$f_{SAMPLING}= f_{EEVS}/4$、N=8
1000：$f_{SAMPLING}= f_{EEVS}/8$、N=6
1001：$f_{SAMPLING}= f_{EEVS}/8$、N=8
1010：$f_{SAMPLING}= f_{EEVS}/16$、N=5
1011：$f_{SAMPLING}= f_{EEVS}/16$、N=6
1100：$f_{SAMPLING}= f_{EEVS}/16$、N=8
1101：$f_{SAMPLING}= f_{EEVS}/32$、N=5
1110：$f_{SAMPLING}= f_{EEVS}/32$、N=6
1111：$f_{SAMPLING}= f_{EEVS}/32$、N=8

37.5.54 HRTIM ADC トリガ 1 レジスタ（HRTIM_ADC1R）

アドレスオフセット：0x3BCh

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AD1TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 これらのビットによって ADC トリガ 1 出力（hrtim_adc_trg1）のためのトリガソースを選択します。詳細は、HRTIM_ADC3R ビットの説明を参照してください。

37.5.55 HRTIM ADC トリガ 2 レジスタ（HRTIM_ADC2R）

アドレスオフセット：0x3C0h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 これらのビットによって ADC トリガ 2 出力（hrtim_adc_trg2）のためのトリガソースを選択します。詳細は、HRTIM_ADC4R ビットの説明を参照してください。

37.5.56 HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R）

アドレスオフセット：0x3C4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC3TEPER	ADC3TEC4	ADC3TEC3	ADC3TEC2	ADC3TDPER	ADC3TDC4	ADC3TDC3	ADC3TDC2	ADC3TCPER	ADC3TCC4	ADC3TCC3	ADC3TCC2	ADC3TBRST	ADC3TBPER	ADC3TBC4	ADC3TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3TBC2	ADC3TARST	ADC3TAPER	ADC3TAC4	ADC3TAC3	ADC3TAC2	ADC3EEV5	ADC3EEV4	ADC3EEV3	ADC3EEV2	ADC3EEV1	ADC3MPER	ADC3MC4	ADC3MC3	ADC3MC2	ADC3MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ADC3TEPER**：*タイマ E 周期による ADC トリガ 3*
ADC3TAPER の説明を参照してください。

ビット 30 **ADC3TEC4**：*タイマ E の比較 4 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 29 **ADC3TEC3**：*タイマ E の比較 3 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 28 **ADC3TEC2**：*タイマ E の比較 2 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 27 **ADC3TDPER**：*タイマ D 周期による ADC トリガ 3*
ADC3TAPER の説明を参照してください。

ビット 26 **ADC3TDC4**：*タイマ D の比較 4 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 25 **ADC3TDC3**：*タイマ D の比較 3 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 24 **ADC3TDC2**：*タイマ D の比較 2 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 23 **ADC3TCPER**：*タイマ C 周期による ADC トリガ 3*
ADC3TAPER の説明を参照してください。

ビット 22 **ADC3TCC4**：*タイマ C の比較 4 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 21 **ADC3TCC3**：*タイマ C の比較 3 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 20 **ADC3TCC2**：*タイマ C の比較 2 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 19 **ADC3TBRST**：*タイマ B のリセットおよびカウンタロールオーバーによる ADC トリガ 3*
ADC3TBRST の説明を参照してください。

ビット 18 **ADC3TBPER**：*タイマ B 周期による ADC トリガ 3*
ADC3TAPER の説明を参照してください。

ビット 17 **ADC3TBC4**：*タイマ B の比較 4 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 16 **ADC3TBC3**：*タイマ B の比較 3 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 15 **ADC3TBC2**：*タイマ B の比較 2 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 14 **ADC3TARST**：*タイマ A のリセットおよびカウンタロールオーバーによる ADC トリガ 3*
このビットは、ADC トリガ 1 出力に、タイマ A のリセットおよびロールオーバーイベント時に ADC トリガを生成することを有効にします。

ビット 13 **ADC3TAPER**：*タイマ A 周期による ADC トリガ 3*
このビットは、ADC トリガ 3 出力（hrtim_adc_trg3）に、タイマ A 周期イベント時に ADC トリガを生成することを有効にします。

ビット 12 **ADC3TAC4**：*タイマ A 比較 4 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 11 **ADC3TAC3**：*タイマ A 比較 3 による ADC トリガ 3*
ADC3TAC2 の説明を参照してください。

ビット 10 **ADC3TAC2**：*タイマ A 比較 2 による ADC トリガ 3*
このビットは、ADC トリガ 3 出力（hrtim_adc_trg3）に、タイマ A 比較 2 イベント時に ADC トリガの生成を有効にします。

ビット 9 **ADC3EEV5**：*外部イベント 5 による ADC トリガ 3*
ADC3EEV1 の説明を参照してください。

ビット 8 **ADC3EEV4**：*外部イベント 4 による ADC トリガ 3*
ADC3EEV1 の説明を参照してください。

ビット 7 **ADC3EEV3**：*外部イベント 3 による ADC トリガ 3*
ADC3EEV1 の説明を参照してください。

ビット 6 **ADC3EEV2**：*外部イベント 2 による ADC トリガ 3*
ADC3EEV1 の説明を参照してください。

ビット 5 **ADC3EEV1**：*外部イベント 1 による ADC トリガ 3*
このビットは、ADC トリガ 3 出力（hrtim_adc_trg3）に、外部イベント 1 の時に ADC トリガの生成を有効にします。

ビット 4 **ADC3MPER**：*マスタ周期による ADC トリガ 3*
このビットは、ADC トリガ 3 出力（hrtim_adc_trg3）に、マスタタイマ周期イベント時に ADC トリガの生成を有効にします。

ビット 3 **ADC3MC4**：*マスタの比較 4 による ADC トリガ 3*
ADC3MC1 の説明を参照してください。

ビット 2 **ADC3MC3**：*マスタの比較 3 による ADC トリガ 3*
ADC3MC1 の説明を参照してください。

ビット 1 **ADC3MC2**：*マスタの比較 2 による ADC トリガ 3*
ADC3MC1 の説明を参照してください。

ビット 0 **ADC3MC1**：*マスタの比較 1 による ADC トリガ 3*
このビットは、ADC トリガ 3 出力（hrtim_adc_trg3）に、マスタの比較 1 イベント時に ADC トリガの生成を有効にします。

37.5.57 HRTIM ADC トリガ 4 レジスタ（HRTIM_ADC4R）

アドレスオフセット：0x3C8h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC4TERST	ADC4TEC4	ADC4TEC3	ADC4TEC2	ADC4TDRST	ADC4TDPER	ADC4TDC4	ADC4TDC3	ADC4TDC2	ADC4TCRST	ADC4TCPER	ADC4TCC4	ADC4TCC3	ADC4TCC2	ADC4TBPER	ADC4TBC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC4TBC3	ADC4TBC2	ADC4TAPER	ADC4TAC4	ADC4TAC3	ADC4TAC2	ADC4EEV10	ADC4EEV9	ADC4EEV8	ADC4EEV7	ADC4EEV6	ADC4MPER	ADC4MC4	ADC4MC3	ADC4MC2	ADC4MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ADC4TERST**：*タイマ E のリセットおよびカウンタロールオーバーによる ADC トリガ 4* (1)
ADC4TCRST の説明を参照してください。

ビット 30 **ADC4TEC4**：*タイマ E の比較 4 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 29 **ADC4TEC3**：*タイマ E の比較 3 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 28 **ADC4TEC2**：*タイマ E の比較 2 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 27 **ADC4TDRST**：*タイマ D のリセットおよびカウンタロールオーバーによる ADC トリガ 4* (1)
ADC4TCRST の説明を参照してください。

ビット 26 **ADC4TDPER**：*タイマ D 周期による ADC トリガ 4*
ADC4TAPER の説明を参照してください。

ビット 25 **ADC4TDC4**：*タイマ D の比較 4 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 24 **ADC4TDC3**：*タイマ D の比較 3 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 23 **ADC4TDC2**：*タイマ D の比較 2 による ADC トリガ 2*
ADC4TAC2 の説明を参照してください。

ビット 22 **ADC4TCRST**：*タイマ C のリセットおよびカウンタロールオーバーによる ADC トリガ 4* (1)
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、タイマ C のリセットおよびロールオーバーイベント時に ADC トリガの生成を有効にします。

ビット 21 **ADC4TCPER**：*タイマ C 周期による ADC トリガ 4*
ADC4TAPER の説明を参照してください。

ビット 20 **ADC4TCC4**：*タイマ C の比較 4 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 19 **ADC4TCC3**：*タイマ C の比較 3 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 18 **ADC4TCC2**：*タイマ C の比較 2 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 17 **ADC4TBPER**：*タイマ B 周期による ADC トリガ 4*
ADC4TAPER の説明を参照してください。

ビット 16 **ADC4TBC4**：*タイマ B の比較 4 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 15 **ADC4TBC3**：*タイマ B の比較 3 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 14 **ADC4TBC2**：*タイマ B の比較 2 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 13 **ADC4TAPER**：*タイマ A 周期による ADC トリガ 4*
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、タイマ A イベント時に ADC トリガの生成を有効にします。

ビット 12 **ADC4TAC4**：*タイマ A 比較 4 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 11 **ADC4TAC3**：*タイマ A 比較 3 による ADC トリガ 4*
ADC4TAC2 の説明を参照してください。

ビット 10 **ADC4TAC2**：*タイマ A 比較 2 による ADC トリガ 4*
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、タイマ A の比較 2 の時に ADC トリガの生成を有効にします。

ビット 9 **ADC4EEV10**：*外部イベント 10 による ADC トリガ 4* (1)
ADC4EEV6 の説明を参照してください。

ビット 8 **ADC4EEV9**：*外部イベント 9 による ADC トリガ 4* (1)
ADC4EEV6 の説明を参照してください。

ビット 7 **ADC4EEV8**：*外部イベント 8 による ADC トリガ 4* (1)
ADC4EEV6 の説明を参照してください。

ビット 6 **ADC4EEV7**：*外部イベント 7 による ADC トリガ 4* (1)
ADC4EEV6 の説明を参照してください。

ビット 5 **ADC4EEV6**：*外部イベント 6 による ADC トリガ 4* (1)
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、外部イベント 6 の時に ADC トリガの生成を有効にします。

ビット 4 **ADC4MPER**：*マスタ周期による ADC トリガ 4*
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、マスタ周期イベント時に ADC トリガの生成を有効にします。

ビット 3 **ADC4MC4**：*マスタの比較 4 による ADC トリガ 4*
ADC4MC1 の説明を参照してください。

ビット 2 **ADC4MC3**：*マスタの比較 3 による ADC トリガ 4*
ADC4MC1 の説明を参照してください。

ビット 1 **ADC4MC2**：*マスタの比較 2 による ADC トリガ 4*
ADC4MC1 の説明を参照してください。

ビット 0 **ADC4MC1**：*マスタの比較 1 による ADC トリガ 4*
このビットは、ADC トリガ 4 出力（hrtim_adc_trg4）に、マスタの比較 1 イベント時に ADC トリガを生成することを有効にします。

1. これらのトリガは、HRTIM_ADC1R/HRTIM_ADC3R と HRTIM_ADC2R/HRTIM_ADC4R とで異なります。

37.5.58 HRTIM フォールト入力レジスタ 1（HRTIM_FLTINR1）

アドレスオフセット：0x3D0h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLT4LCK	FLT4F[3:0]				FLT4SRC	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC	FLT3P	FLT3E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT2LCK	FLT2F[3:0]				FLT2SRC	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC	FLT1P	FLT1E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **FLT4LCK**：フォールト 4 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 30:27 **FLT4F[3:0]**：フォールト 4 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 26 **FLT4SRC**：フォールト 4 ソース
HRTIM_FLTINR2 レジスタの FLT5SRC の説明を参照してください。

ビット 25 **FLT4P**：フォールト 4 極性
HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 24 **FLT4E**：フォールト 4 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 23 **FLT3LCK**：フォールト 3 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 22:19 **FLT3F[3:0]**：フォールト 3 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 18 **FLT3SRC**：フォールト 3 ソース
HRTIM_FLTINR2 レジスタの FLT5SRC の説明を参照してください。

ビット 17 **FLT3P**：フォールト 3 極性
HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 16 **FLT3E**：フォールト 3 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 15 **FLT2LCK**：フォールト 2 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 14:11 **FLT2F[3:0]**：フォールト 2 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 10 **FLT2SRC**：フォールト 2 ソース
HRTIM_FLTINR2 レジスタの FLT5SRC の説明を参照してください。

ビット 9 **FLT2P**：フォールト 2 極性
HRTIM_FLTINR2 レジスタの FLT2P の説明を参照してください。

ビット 8 **FLT2E**：フォールト 2 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 7 **FLT1LCK**：フォールト 1 ロック

HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 6:3 **FLT1F[3:0]**：フォールト 1 フィルタ

HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 2 **FLT1SRC**：フォールト 1 ソース

HRTIM_FLTINR2 レジスタの FLT5SRC の説明を参照してください。

ビット 1 **FLT1P**：フォールト 1 極性

HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 0 **FLT1E**：フォールト 1 イネーブル

HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

37.5.59 HRTIM フォールト入力レジスタ 2（HRTIM_FLTINR2）

アドレスオフセット：0x3D4h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5LCK	FLT5F[3:0]				FLT5SRC	FLT5P	FLT5E
								rwo	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **FLTSD[1:0]**：フォールトサンプリングクロック分周比

このビットフィールドは、タイマクロック周波数（f_{HRTIM}）と
デジタルフィルタに使用されるフォールト信号サンプリングクロック（f_{FLTS}）との間の分周比を示します。

00：$f_{FLTS}=f_{HRTIM}$

01：$f_{FLTS}=f_{HRTIM}$ / 2

10：$f_{FLTS}=f_{HRTIM}$ / 4

11：$f_{FLTS}=f_{HRTIM}$ / 8

注： このビットフィールドは、どの FLTxE イネーブルビットの前にも書き込む必要があります。

ビット 23:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **FLT5LCK**：フォールト 5 ロック

誤った書き込みアクセスからフォールトプログラミングビットの書き込み属性を保護するように、FLT5LCK ビットでフォールトプログラミングビットの書き込み属性を変更します。

このビットは 1 回のみ書き込み可能です。いったんセットされると、次のシステムリセットまで変更できません。

0：FLT5E、FLT5P、FLT5SRC、FLT5F[3:0] ビットは読み出し／書き込み用です。

1：FLT5E、FLT5P、FLT5SRC、FLT5F[3:0] ビットはこれ以降、書き込みできません（読み出し専用モード）。

ビット 6:3 **FLT5F[3:0]**：フォールト 5 フィルタ

このビットフィールドは、FLT5 入力をサンプリングする周波数と、FLT5 に適用されるデジタルフィルタの長さを決定します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、FLT5 は非同期として動作します。

0001：$f_{SAMPLING}= f_{HRTIM}$、N = 2

0010：$f_{SAMPLING}= f_{HRTIM}$、N = 4

0011：$f_{SAMPLING}= f_{HRTIM}$、N = 8

0100：$f_{SAMPLING}= f_{FLTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{FLTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{FLTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{FLTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{FLTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{FLTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{FLTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{FLTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{FLTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{FLTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{FLTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{FLTS}/32$、N = 8

注：　このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。このビットフィールドは、FLT5LOCK がプログラムされているとき、変更できません。

ビット 2 **FLT5SRC**：フォールト 5 ソース

このビットは、FAULT5 入力ソースを選択します（接続の詳細は、*表 301* を参照）。

0：フォールト 1 入力は HRTIM_FLT5 入力ピンです。

1：フォールト 1 入力は hrtim_in_flt5 信号です。

注：　このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。

ビット 1 **FLT5P**：フォールト 5 極性

このビットは、FAULT5 入力の極性を選択します。

0：フォールト 5 入力はアクティブローです。

1：フォールト 5 入力はアクティハイです。

注：　このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。

ビット 0 **FLT5E**：フォールト 5 イネーブル

このビットは、グローバル FAULT5 入力回路を有効にします。

0：フォールト 5 入力は無効です。

1：フォールト 5 入力は有効です。

37.5.60 HRTIM バースト DMA マスタタイマ更新レジスタ（HRTIM_BDMUPR）

アドレスオフセット：0x3D8h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **MCMP4**：*MCMP4R レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 8 **MCMP3**：*MCMP3R レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 7 **MCMP2**：*MCMP2R レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 6 **MCMP1**：*MCMP1R レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 5 **MREP**：*MREP レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 4 **MPER**：*MPER レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 3 **MCNT**：*MCNTR レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 2 **MDIER**：*MDIER レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 1 **MICR**：*MICR レジスタ更新イネーブル*
MCR の説明を参照してください。

ビット 0 **MCR**：*MCR レジスタ更新イネーブル*
このビットは、マスタタイマ MCR レジスタが、バースト DMA によって更新されるレジスタ列の一部であるかどうかを決定します。
0：MCR レジスタはバースト DMA アクセスによって更新されません。
1：MCR レジスタはバースト DMA アクセスによって更新されます。

37.5.61 HRTIM バースト DMA タイマ x 更新レジスタ（HRTIM_BDTxUPR）

アドレスオフセット：0x3DCh～0x3ECh

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMxFLTR	TIMxOUTR	TIMxCHPR	TIMxRSTR	TIMxEEFR2
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMxEEFR1	TIMxRST2R	TIMxSET2R	TIMxRST1R	TIMxSET1R	TIMxDTxR	TIMxCMP4	TIMxCMP3	TIMxCMP2	TIMxCMP1	TIMxREP	TIMxPER	TIMxCNT	TIMxDIER	TIMxICR	TIMxCR
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TIMxFLTR**：HRTIM_FLTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 19 **TIMxOUTR**：HRTIM_OUTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 18 **TIMxCHPR**：HRTIM_CHPxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 17 **TIMxRSTR**：HRTIM_RSTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 16 **TIMxEEFR2**：HRTIM_EEFxR2 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 15 **TIMxEEFR1**：HRTIM_EEFxR1 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 14 **TIMxRST2R**：HRTIM_RST2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 13 **TIMxSET2R**：HRTIM_SET2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 12 **TIMxRST1R**：HRTIM_RST1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 11 **TIMxSET1R**：HRTIM_SET1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 10 **TIMxDTR**：HRTIM_DTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 9 **TIMxCMP4**：HRTIM_CMP4xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 8 **TIMxCMP3**：HRTIM_CMP3xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 7 **TIMxCMP2**：HRTIM_CMP2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 6 **TIMxCMP1**：HRTIM_CMP1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 5 **TIMxREP**：HRTIM_REPxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 4 **TIMxPER**：HRTIM_PERxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 3 **TIMxCNT**：HRTIM_CNTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 2 **TIMxDIER**：HRTIM_TIMxDIER レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 1 **TIMxICR**：HRTIM_TIMxICR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 0 **TIMxCR**：HRTIM_TIMxCR レジスタ更新イネーブル
このビットは、マスタタイマ MCR レジスタが、バースト DMA によって更新されるレジスタ列の一部であるかどうかを定義します。
0：HRTIM_TIMxCR レジスタはバースト DMA アクセスによって更新されません。
1：HRTIM_TIMxCR レジスタはバースト DMA アクセスによって更新されます。

37.5.62 HRTIM バースト DMA データレジスタ（HRTIM_BDMADR）

アドレスオフセット：0x3F0h

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BDMADR[31:16]															
wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMADR[15:0]															
wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo

ビット 31:0 **BDMADR[31:0]**：バースト DMA データレジスタ
このレジスタへ書き込みアクセスすると、以下のトリガがかかります。
– BDTxUPR および BDMUPR レジスタのビットで有効化されたレジスタにデータ値をコピー
– 次の書き込み位置にレジスタポインタをインクリメント

37.5.63 HRTIM レジスタマップ

下表に、HRTIM レジスタの配置の一覧を示します。*表 307* および*表 308* のアドレスオフセットは、*表 306* に示されたベースアドレスのオフセットで参照されます。

表 306. RTIM グローバルレジスタマップ

ベースアドレスオフセット	レジスタ
0x000 - 0x07F	マスタタイマ
0x080 - 0x0FF	タイマ A
0x100 - 0x17F	タイマ B
0x180 - 0x1FF	タイマ C
0x200 - 0x27F	タイマ D
0x280 - 0x2FF	タイマ E
0x300 - 0x37F	予約済み
0x380 - 0x3FF	共通レジスタ

表 307. HRTIM レジスタマップとリセット値：マスタタイマ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	HRTIM_MCR	BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	Res.	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN	SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		Res.	Res.	HALF	RETRIG	CONT	CKPSC[2:0]		
	リセット値	0	0	0		0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0004	HRTIM_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
	リセット値																										0	0	0	0	0	0	0
0x0008	HRTIM_MICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
	リセット値																										0	0	0	0	0	0	0
0x000C	HRTIM_MDIER(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
	リセット値										0	0	0	0	0	0	0										0	0	0	0	0	0	0
0x0010	HRTIM_MCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	HRTIM_MPER(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPER[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1

表 307. HRTIM レジスタマップとリセット値：マスタタイマ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0018	HRTIM_MREP (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MREP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x001C	HRTIM_MCMP1R (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	予約済み	Res.	Res.	Res	Res.	Res	Res.	Res.	Res	Res.	Res.	Res.	Res	Res.	Res	Res.	Res.	Res	Res	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res
	リセット値																																
0x0024	HRTIM_MCMP2R (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_MCMP3R (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	HRTIM_MCMP4R (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. このレジスタはプリロードできます（ *1327 ページの表 292* 参照）。

参考資料

表 308. HRTIM レジスタマップとリセット値：TIMx（x= A～E）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	HRTIM_TIMxCR	UPDGAT[3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	Res.	TxRSTU	TxREPU	Res.	DELCMP4[1:0]		DELCMP2[1:0]		SYNCSTRTx	SYNCRSTx	Res.	Res.	Res.	PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0		0	0		0	0	0	0	0	0				0	0	0	0	0	0	0
0x0004	HRTIM_TIMxISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2STAT	O1STAT	IPPSTAT	CPPSTAT	Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	リセット値											0	0	0	0	0	0		0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x0008	HRTIM_TIMxICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	リセット値																	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x000C	HRTIM_TIMxDIER(1)	Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SET1xDE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE	Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	リセット値		0	0	0	0	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x0010	HRTIM_CNTxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTx[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	HRTIM_PERxR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERx[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0x0018	HRTIM_REPxR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x001C	HRTIM_CMP1xR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP1x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	HRTIM_CMP1CxR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]								CMP1x[15:0]															
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	HRTIM_CMP2xR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP2x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_CMP3xR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP3x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	HRTIM_CMP4xR (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP4x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	HRTIM_CPT1xR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPT1x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	HRTIM_CPT2xR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPT2x[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 308. HRTIM レジスタマップとリセット値：TIMx（x= A～E）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0038	HRTIM_DTxR (1)	DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]									DTRLKx	DTRSLKx	Res.	DTPRSC[2:0]			SDTRx	DTRx[8:0]								
	リセット値	0	0					0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	HRTIM_SETx1R (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	HRTIM_RSTx1R (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	HRTIM_SETx2R (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	HRTIM_RSTx2R (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	HRTIM_EEFxR1	Res.	Res.	Res.	EE5FLTR[3:0]				EE5LTCH	Res.	EE4FLTR[3:0]				EE4LTCH	Res.	EE3FLTR[3:0]				EE3LTCH	Res.	EE2FLTR[3:0]				EE2LTCH	Res.	EE1FLTR[3:0]				EE1LTCH
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x0050	HRTIM_EEFxR2	Res.	Res.	Res.	EE10FLTR[3:0]				EE10LTCH	Res.	EE9FLTR[3:0]				EE9LTCH	Res.	EE8FLTR[3:0]				EE8LTCH	Res.	EE7FLTR[3:0]				EE7LTCH	Res.	EE6FLTR[3:0]				EE6LTCH
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x0054	HRTIM_RSTAR (1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_RSTBR (1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_RSTCR (1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

表 308. HRTIM レジスタマップとリセット値：TIMx（x= A～E）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0054	HRTIM_RSTDR (1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_RSTER (1)	Res.	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0058	HRTIM_CHPxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STRTPW [3:0]				CARDTY [2:0]			CARFRQ [3:0]			
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_CPT1ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	Res.	Res.	Res.	Res.	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_CPT1BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	Res.	Res.	Res.	Res.	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_CPT1CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	Res.	Res.	Res.	Res.	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_CPT1DCR	TECMP2	TECMP1	TE1RST	TE1SET	Res.	Res.	Res.	Res.	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_CPT1ECR	Res.	Res.	Res.	Res.	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_CPT2ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	Res.	Res.	Res.	Res.	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_CPT2BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	Res.	Res.	Res.	Res.	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 308. HRTIM レジスタマップとリセット値：TIMx（x= A～E）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0060	HRTIM_CPT2CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	Res.	Res.	Res.	Res.	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_CPT2DCR	TECMP2	TECMP1	TE1RST	TE1SET	Res.	Res.	Res.	Res.	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_CPT2ECR	Res.	Res.	Res.	Res.	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	HRTIM_OUTxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM2	POL2	Res.	Res.	Res.	Res.	DLYPRT[2:0]			DLYPRTEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM1	POL1	Res.
	リセット値									0	0	0	0	0	0	0									0	0	0	0	0	0	0	0	
0x0068	HRTIM_FLTxR	FLTLCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5EN	FLT4EN	FLT3EN	FLT2EN	FLT1EN
	リセット値	0																											0	0	0	0	0

1. このレジスタはプリロードできます（ *1327 ページの表 292* 参照）。

表 309. HRTIM レジスタマップとリセット値：共通機能

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	HRTIM_CR1	Res.	Res.	Res.	Res.	AD4USRC[2:0]			AD3USRC[2:0]			AD2USRC[2:0]			AD1USRC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEUDIS	TDUDIS	TCUDIS	TBUDIS	TAUDIS	MUDIS
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0											0	0	0	0	0	0
0x0004	HRTIM_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	Res.	Res.	TESWU	TDSWU	TCSWU	TBSWU	TASWU	MSWU
	リセット値																			0	0	0	0	0	0				0	0	0	0	0	0
0x008	HRTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
	リセット値															0												0	0	0	0	0	0	0
0x000C	HRTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
	リセット値															0												0	0	0	0	0	0	0

表 309. HRTIM レジスタマップとリセット値：共通機能（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0010	HRTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
	リセット値															0												0	0	0	0	0	0
0x0014	HRTIM_OENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0018	HRTIM_DISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2ODIS	TE1ODIS	TD2ODIS	TD1ODIS	TC2ODIS	TC1ODIS	TB2ODIS	TB1ODIS	TA2ODIS	TA1ODIS
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x001C	HRTIM_ODSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x0020	HRTIM_BMCR	BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEBM	TDBM	TCBM	TBBM	TABM	MTBM	Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
	リセット値	0										0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0024	HRTIM_BMTRG	OCHPEV	Res.	Res.	Res.	Res.	TECMP2	TECMP1	TEREP	TERST	TDCMP2	TDCMP1	TDREP	TDRST	TCCMP2	TCCMP1	TCREP	TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
	リセット値	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_BMCMPR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMCMP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	HRTIM_BMPER(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	HRTIM_EECR1	Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1:0]		EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	HRTIM_EECR2	Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS[1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS[1:0]		EE8POL	EE8SRC[1:0]		Res.	EE7SNS[1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS[1:0]		EE6POL	EE6SRC[1:0]	
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0
0x0038	HRTIM_EECR3	Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EE6SRC[1:0]	
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0

表 309. HRTIM レジスタマップとリセット値：共通機能（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x003C	HRTIM_ADC1R (1)	AD1TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3	AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	HRTIM_ADC2R (1)	AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4	AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	HRTIM_ADC3R (1)	ADC3TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3	AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	HRTIM_ADC4R (1)	AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4	AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0050	HRTIM_FLTINxR1	FLT4LCK	FLT4F[3:0]				FLT4SRC	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC	FLT3P	FLT3E	FLT2LCK	FLT2F[3:0]				FLT2SRC	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC	FLT1P	FLT1E
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	HRTIM_FLTINxR2	Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5LCK	FLT5F[3:0]				FLT5SRC	FLT5P	FLT5E
	リセット値							0	0																	0	0	0	0	0	0	0	0
0x0058	HRTIM_BDMUPDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_BDTAUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMAFLTR	TIMAOUTR	TIMACHPR	TIMARSTR	TIMAEEFR2	TIMAEEFR1	TIMARST2R	TIMASET2R	TIMARST1R	TIMASET1R	TIMADTxR	TIMACMP4	TIMACMP3	TIMACMP2	TIMACMP1	TIMAREP	TIMAPER	TIMACNT	TIMADIER	TIMAICR	TIMACR
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_BDTBUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMBFLTR	TIMBOUTR	TIMBCHPR	TIMBRSTR	TIMBEEFR2	TIMBEEFR1	TIMBRST2R	TIMBSET2R	TIMBRST1R	TIMBSET1R	TIMBDTxR	TIMBCMP4	TIMBCMP3	TIMBCMP2	TIMBCMP1	TIMBREP	TIMBPER	TIMBCNT	TIMBDIER	TIMBICR	TIMBCR
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	HRTIM_BDTCUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMCFLTR	TIMCOUTR	TIMCCHPR	TIMCRSTR	TIMCEEFR2	TIMCEEFR1	TIMCRST2R	TIMCSET2R	TIMCRST1R	TIMCSET1R	TIMCDTxR	TIMCCMP4	TIMCCMP3	TIMCCMP2	TIMCCMP1	TIMCREP	TIMCPER	TIMCCNT	TIMCDIER	TIMCICR	TIMCCR
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 309. HRTIM レジスタマップとリセット値：共通機能（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0068	HRTIM_BDTDUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMDFLTR	TIMDOUTR	TIMDCHPR	TIMDRSTR	TIMDEEFR2	TIMDEEFR1	TIMDRST2R	TIMDSET2R	TIMDRST1R	TIMDSET1R	TIMDDTxR	TIMDCMP4	TIMDCMP3	TIMDCMP2	TIMDCMP1	TIMDREP	TIMDPER	TIMDCNT	TIMDDIER	TIMDICR	TIMDCR
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x006C	HRTIM_BDTEUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEFLTR	TIMEOUTR	TIMECHPR	TIMERSTR	TIMEEEFR2	TIMEEEFR1	TIMERST2R	TIMESET2R	TIMERST1R	TIMESET1R	TIMEDTxR	TIMECMP4	TIMECMP3	TIMECMP2	TIMECMP1	TIMEREP	TIMEPER	TIMECNT	TIMEDIER	TIMEICR	TIMECR
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0070	HRTIM_BDMADR	BDMADR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. このレジスタはプリロードできます（*1327 ページの表 292* 参照）。

38 高機能制御タイマ（TIM1/TIM8）

38.1 TIM1/TIM8 の概要

高機能制御タイマ（TIM1/TIM8）は、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

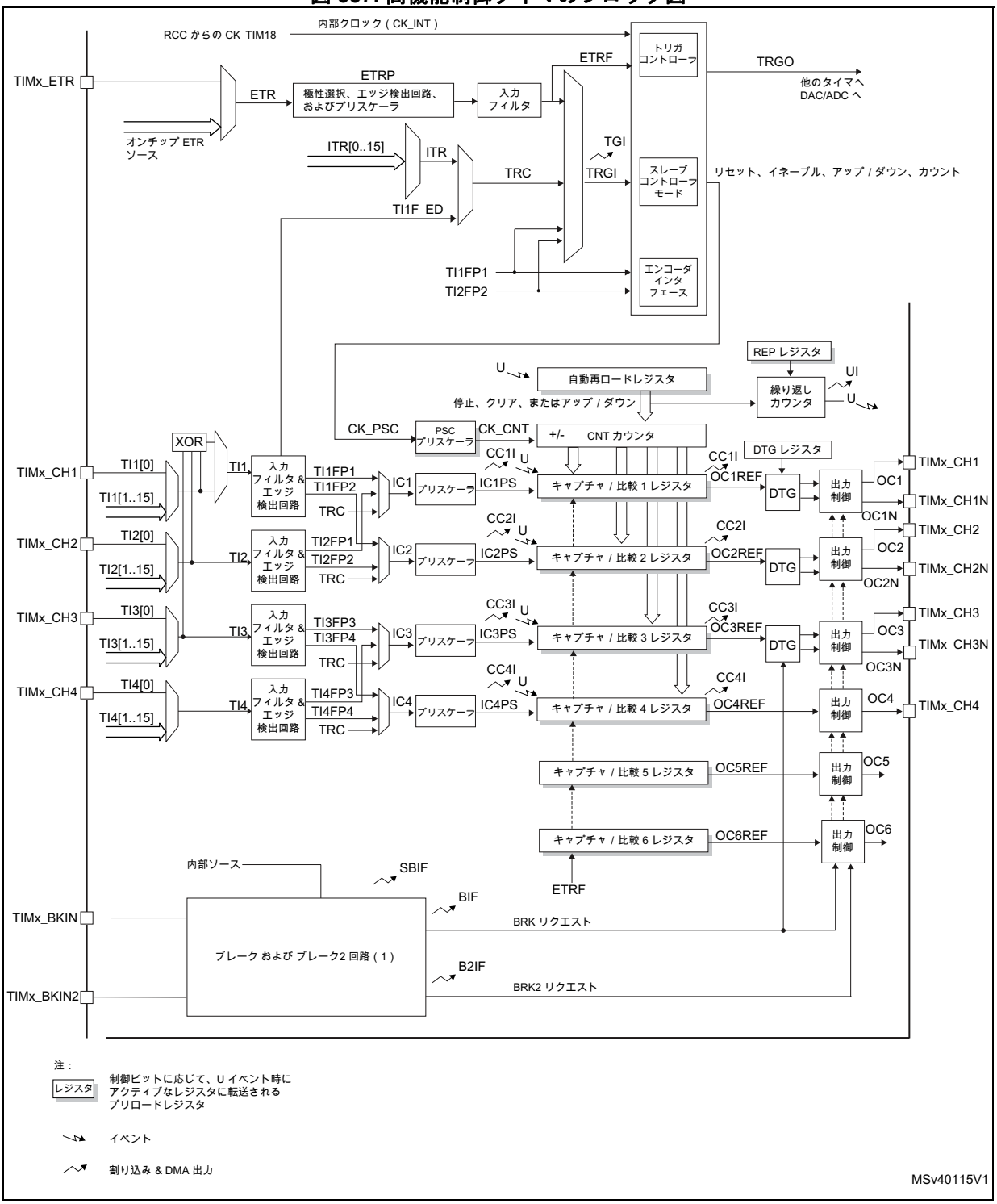
高機能制御タイマ（TIM1/TIM8）と汎用タイマ（TIMy）は、互いに独立しており、リソースを共有しません。これらのタイマは、*セクション 38.3.26：タイマの同期*に示すように、相互に同期させることができます。

38.2 TIM1/TIM8 の主な特長

TIM1/TIM8 タイマの主な機能は、次のとおりです。

- 16 ビットのアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- 16 ビットのプログラム可能なプリスケーラ。カウンタクロック周波数を 1 から 65536 の間で分周でき、分周比の動作中の変更も可能。
- 次の機能を持つ、最大 6 つの独立チャネル。
  - 入力キャプチャ（ただしチャネル 5 および 6）
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をユーザが選択可能な安全な設定にする 2 つのブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

図 337. 高機能制御タイマのブロック図



1. 詳細は、*図 380：ブレーク および ブレーク 2 回路の概要*を参照してください。

38.3 TIM1/TIM8機能詳細

38.3.1 タイムベースユニット

プログラム可能な高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ16 ビットカウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 338 と *図 339* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 338. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

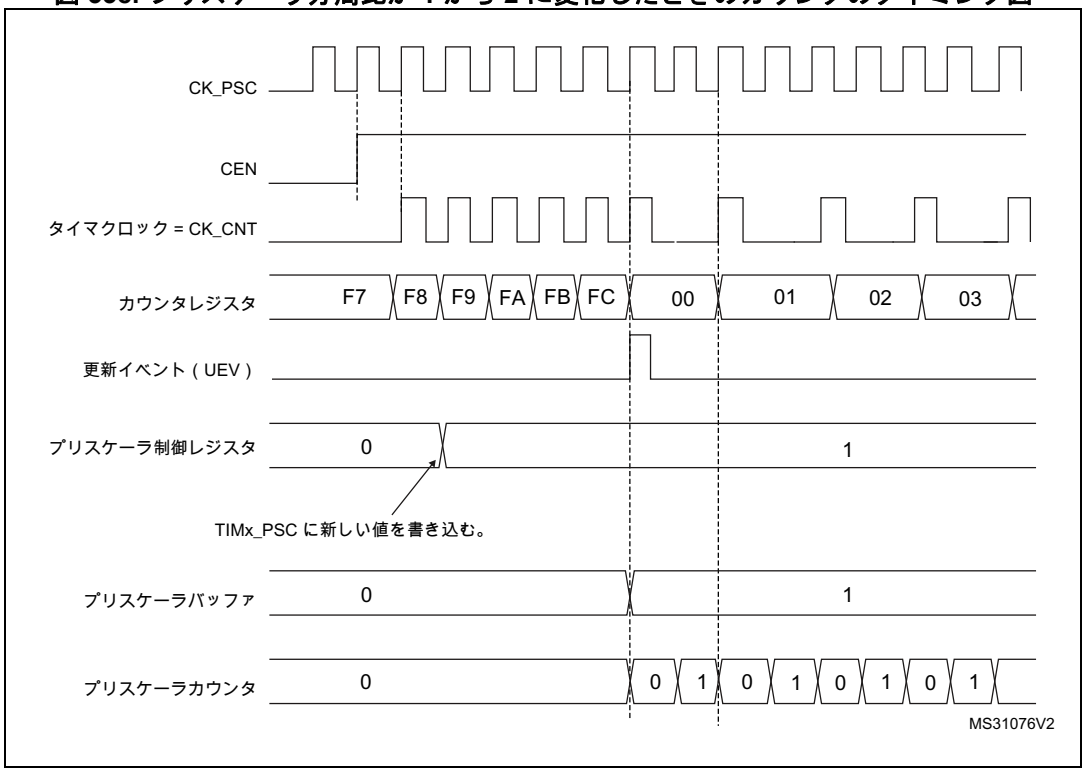
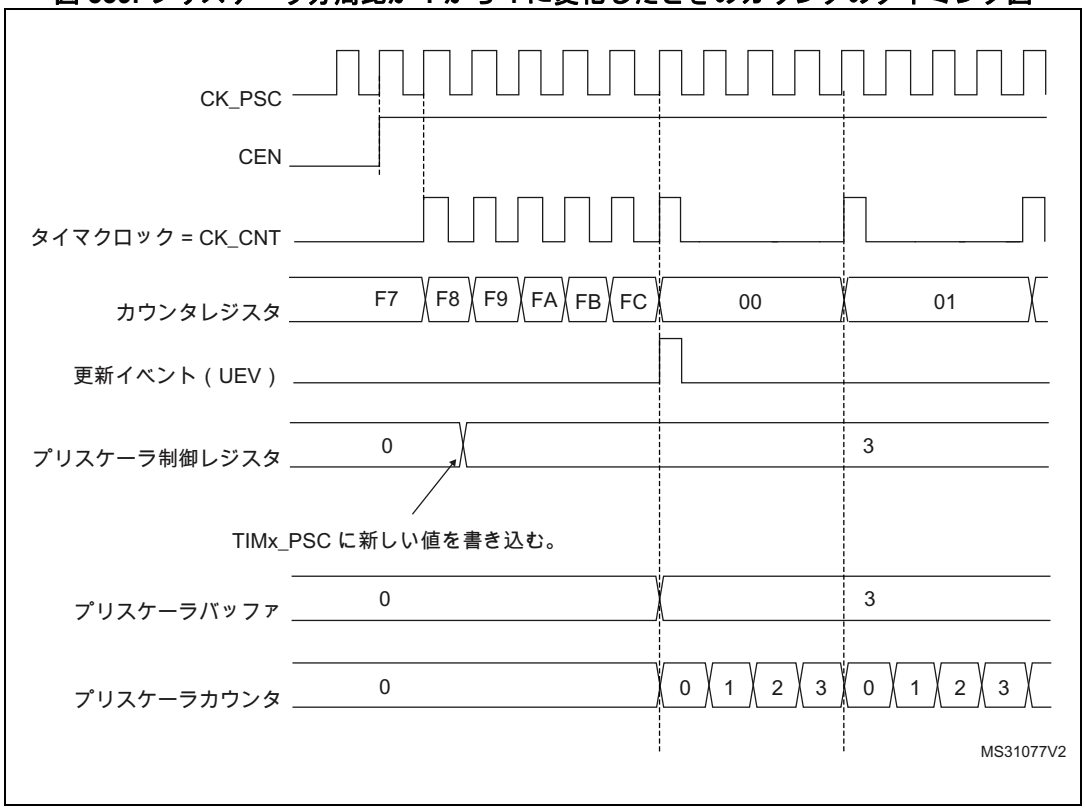


図 339. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



38.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

参考資料

図 340. 内部クロック分周比が 1 の場合のカウンタのタイミング図

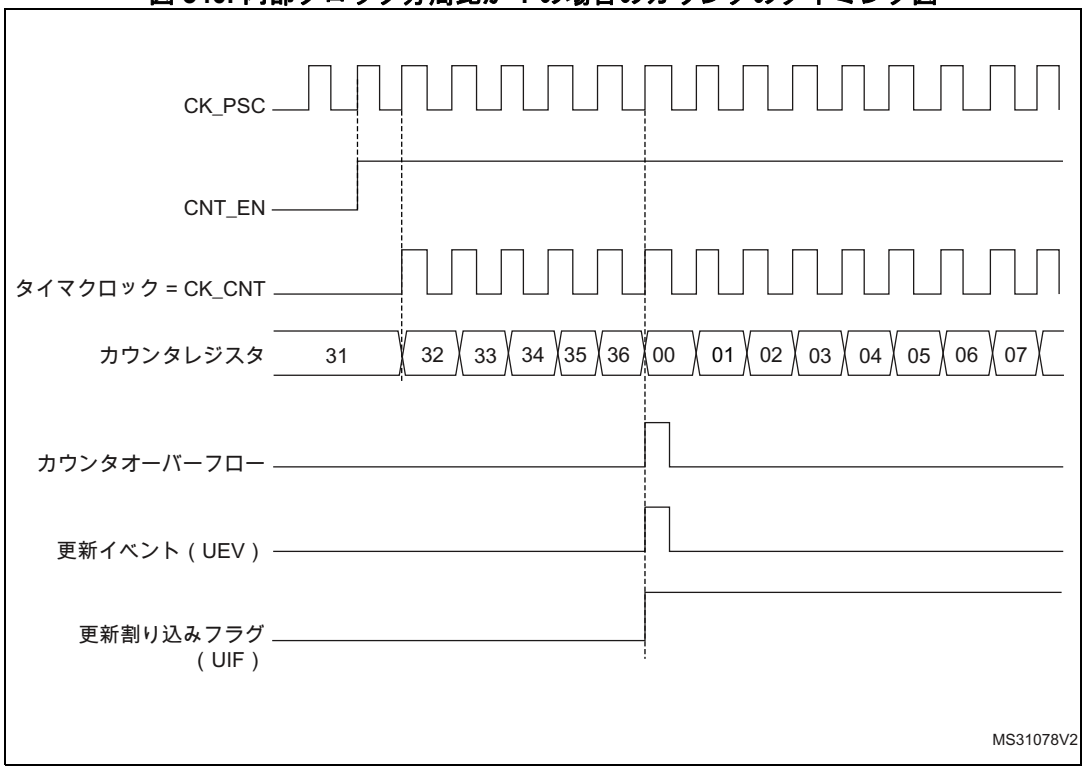


図 341. 内部クロック分周比が 2 の場合のカウンタのタイミング図

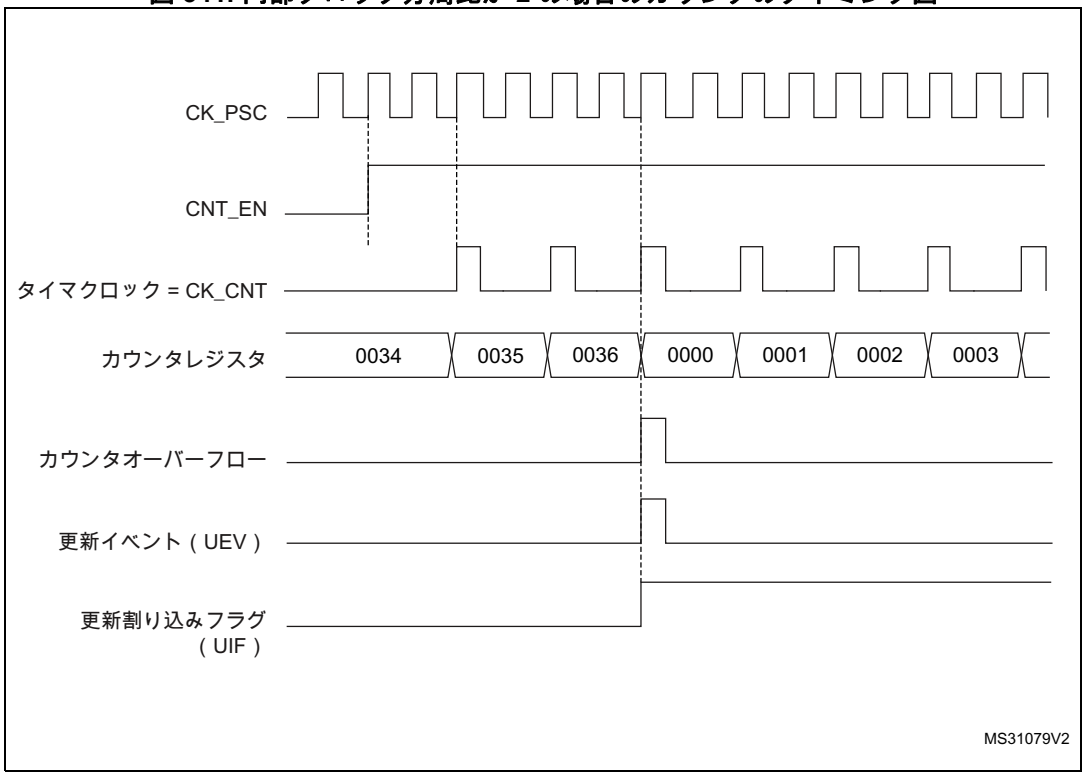


図 342. 内部クロック分周比が 4 の場合のカウンタのタイミング図

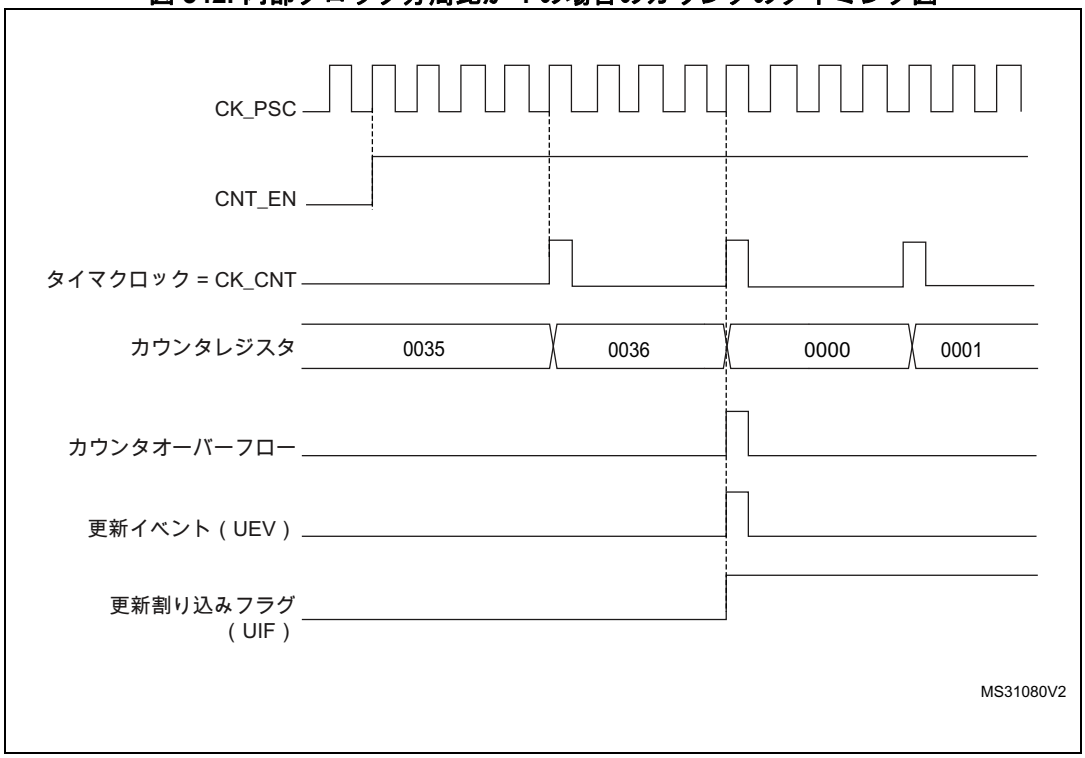


図 343. 内部クロック分周比が N の場合のカウンタのタイミング図

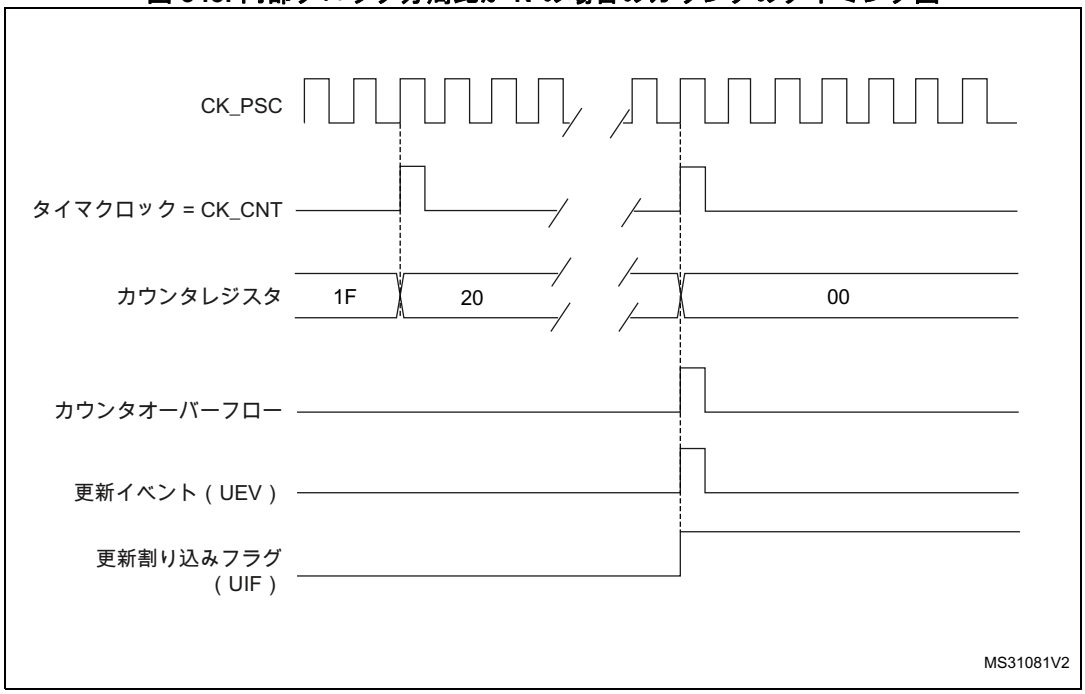


図 344. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

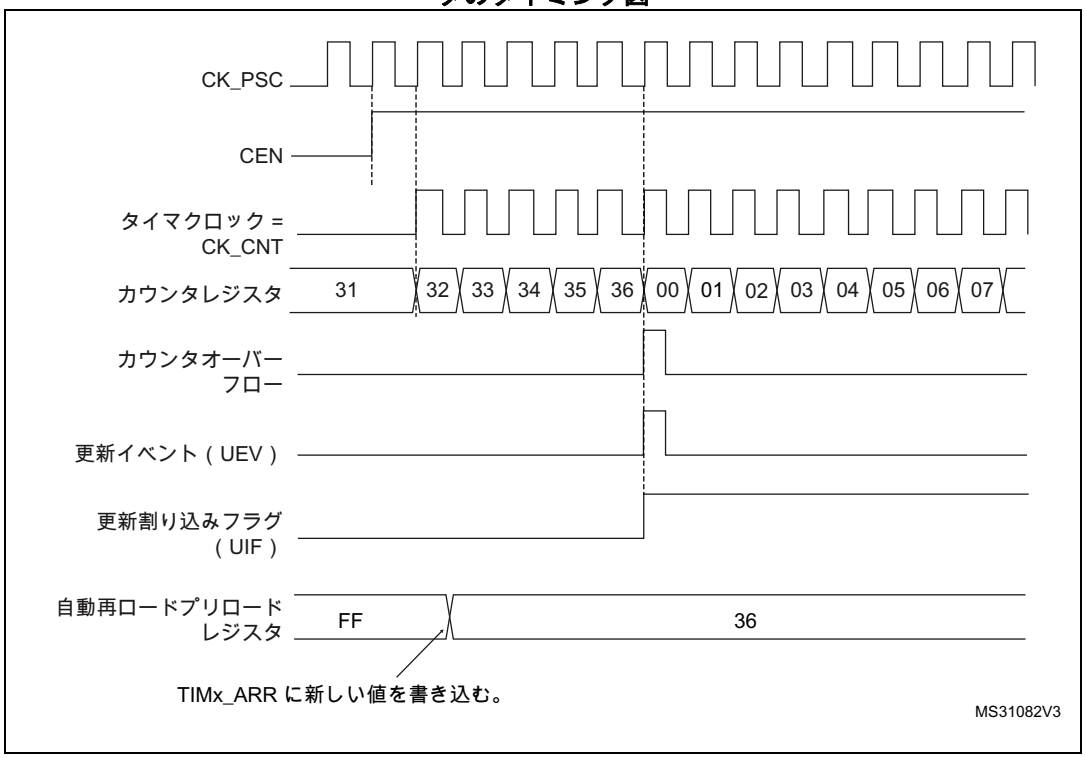
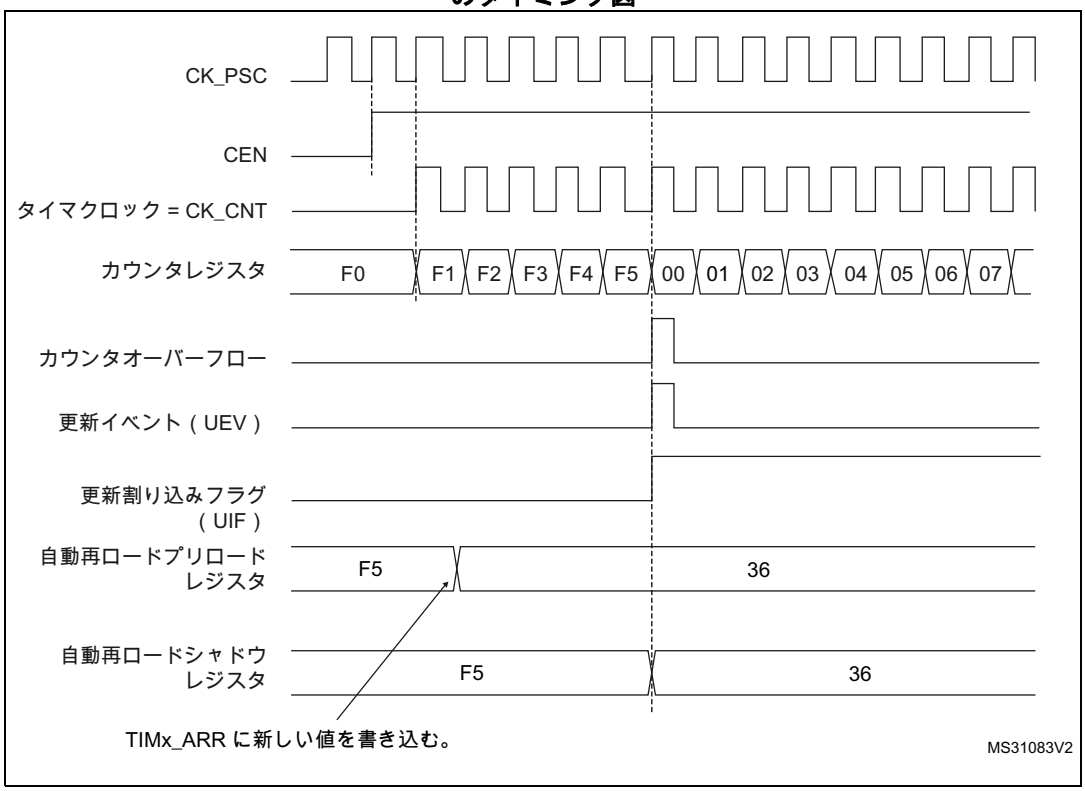


図 345. ARPE = 1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までダウンカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのアンダーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 346. 内部クロック分周比が 1 の場合のカウンタのタイミング図

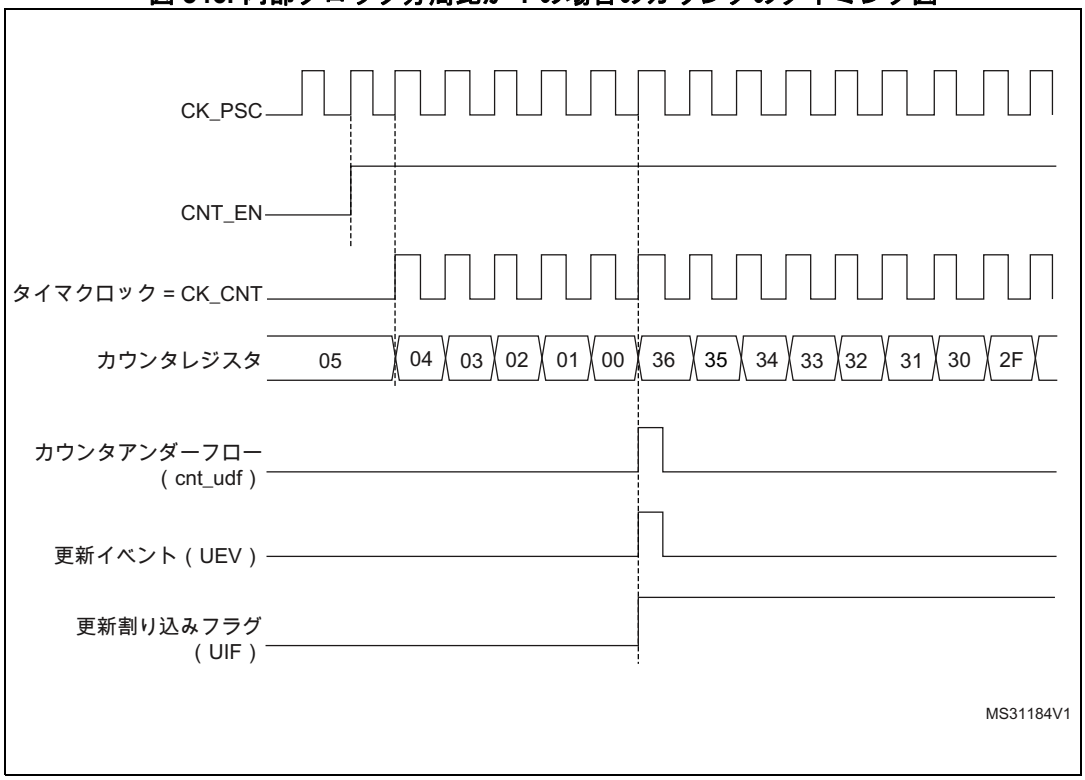
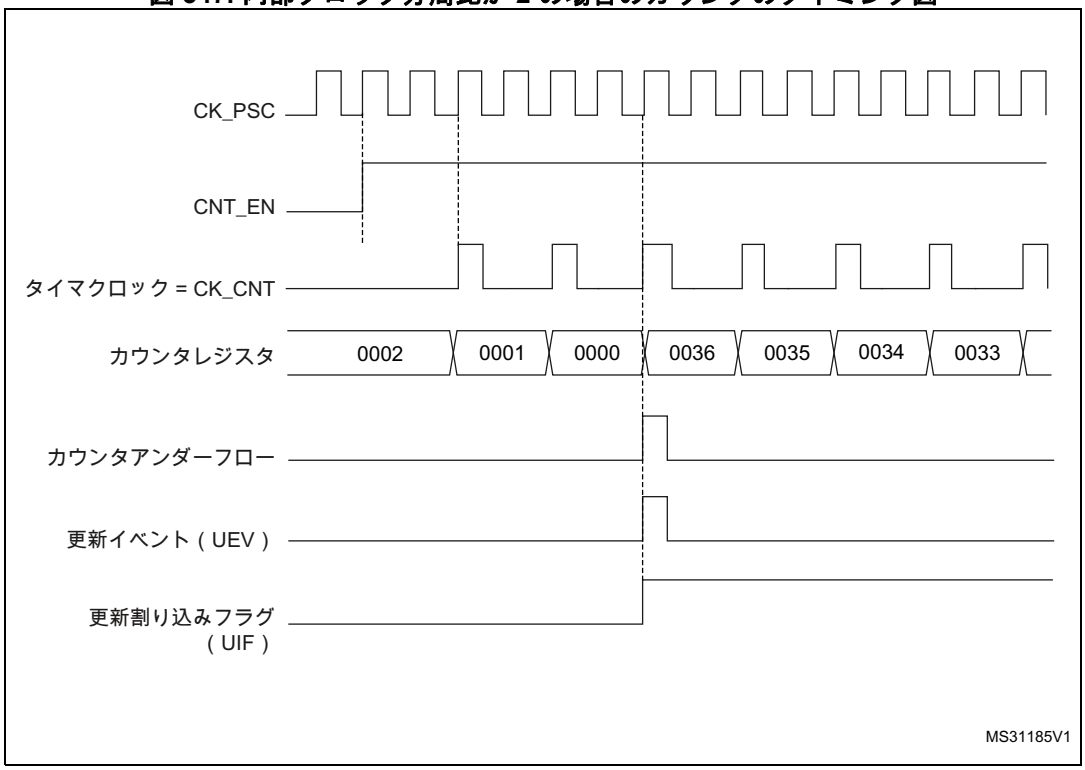


図 347. 内部クロック分周比が 2 の場合のカウンタのタイミング図



参考資料

図 348. 内部クロック分周比が 4 の場合のカウンタのタイミング図

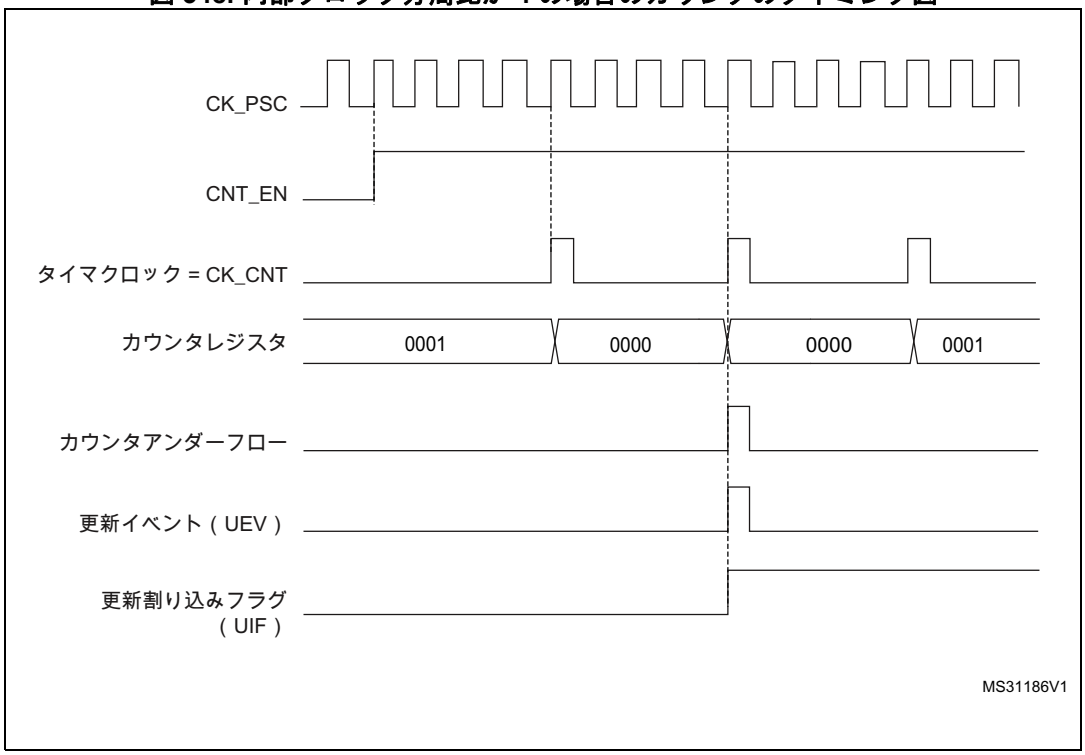


図 349. 内部クロック分周比が N の場合のカウンタのタイミング図

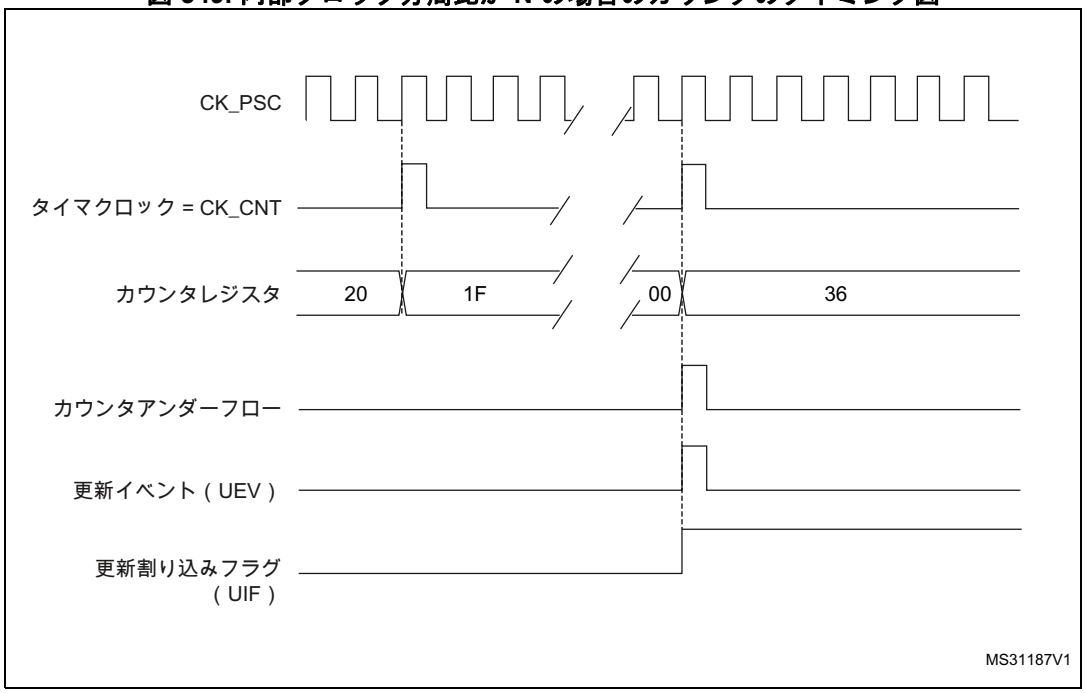
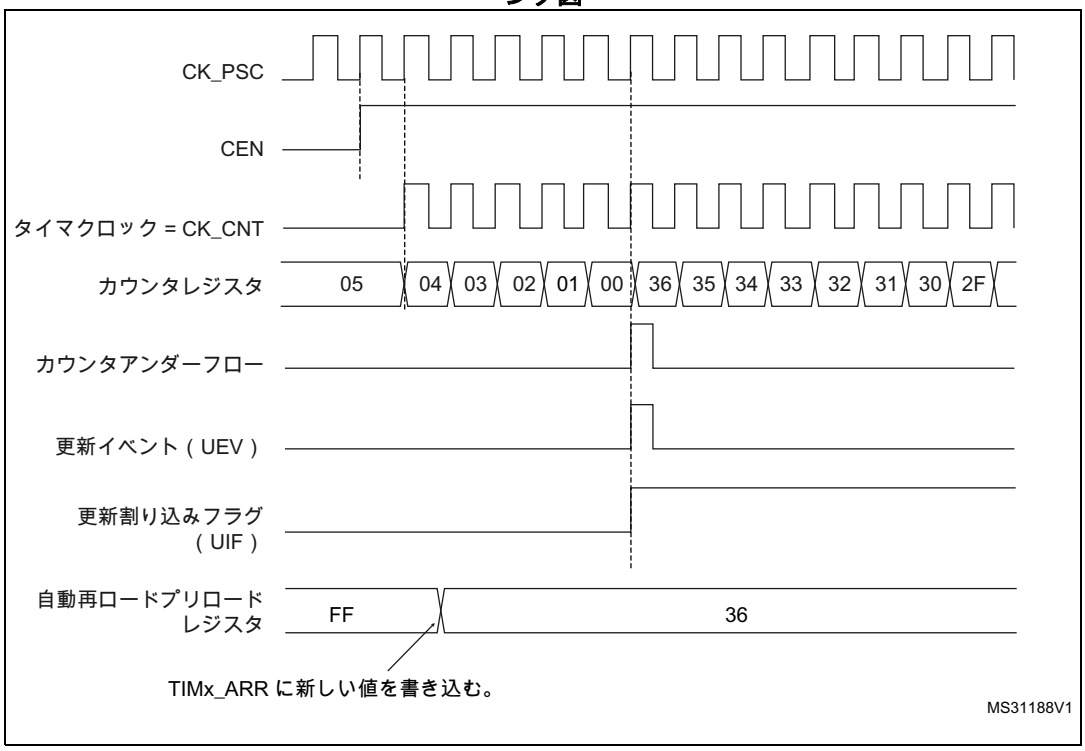


図 350. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割り込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、TIMx_CR1 レジスタの方向ビット（DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合、UG ビットをセットすると UEV 更新イベントが生成されますが、UIF フラグはセットされません（した

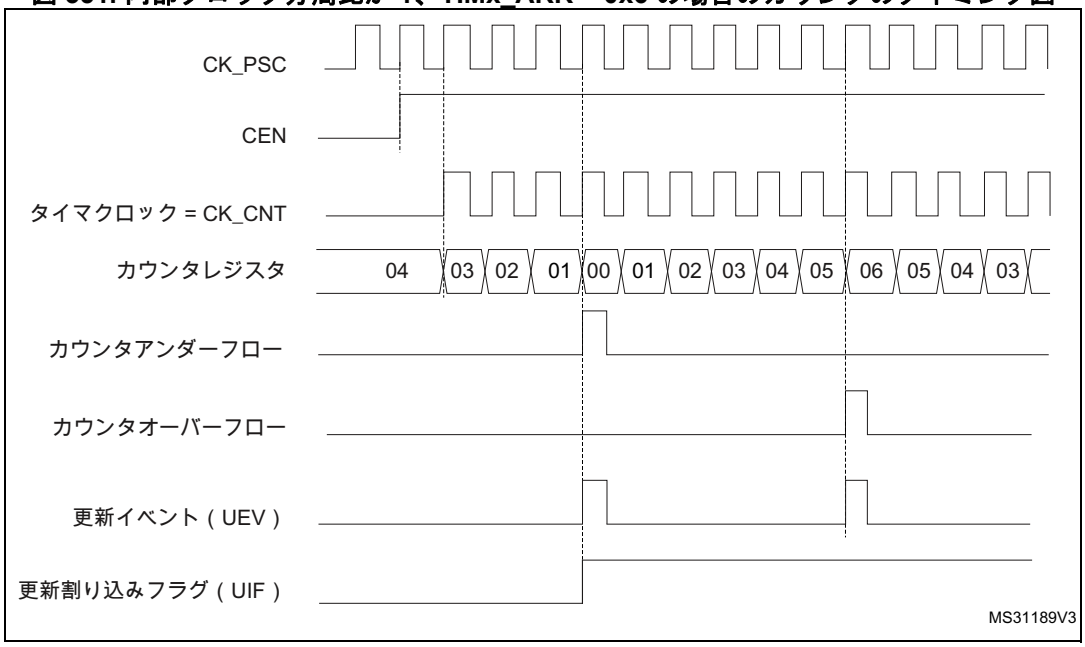
がって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 351. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、*セクション 38.4：TIM1/TIM8 レジスタ*を参照）。

図 352. 内部クロック分周比が 2 の場合のカウンタのタイミング図

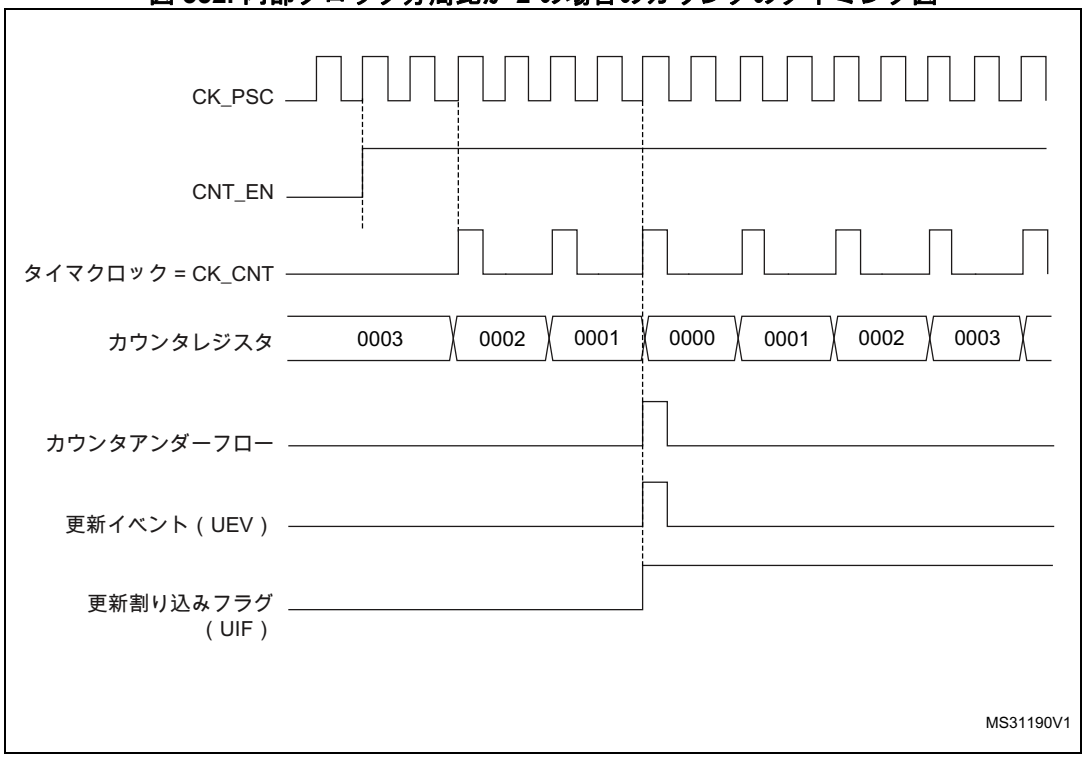


図 353. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図

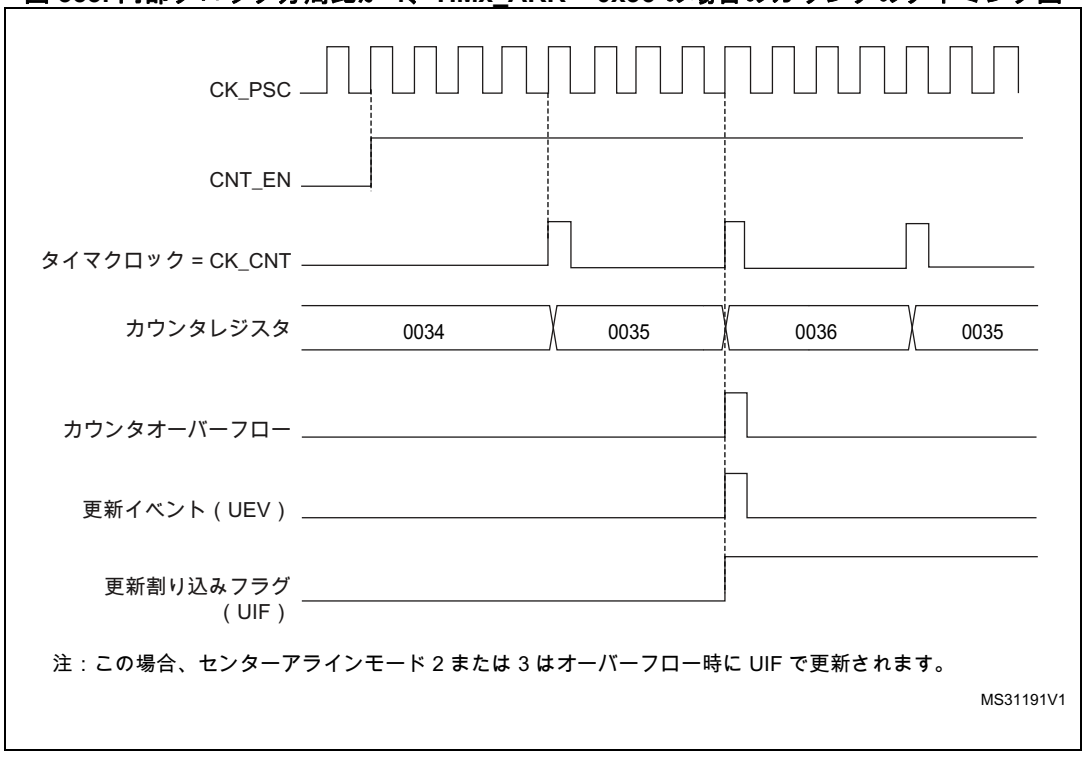


図 354. 内部クロック分周比が N の場合のカウンタのタイミング図

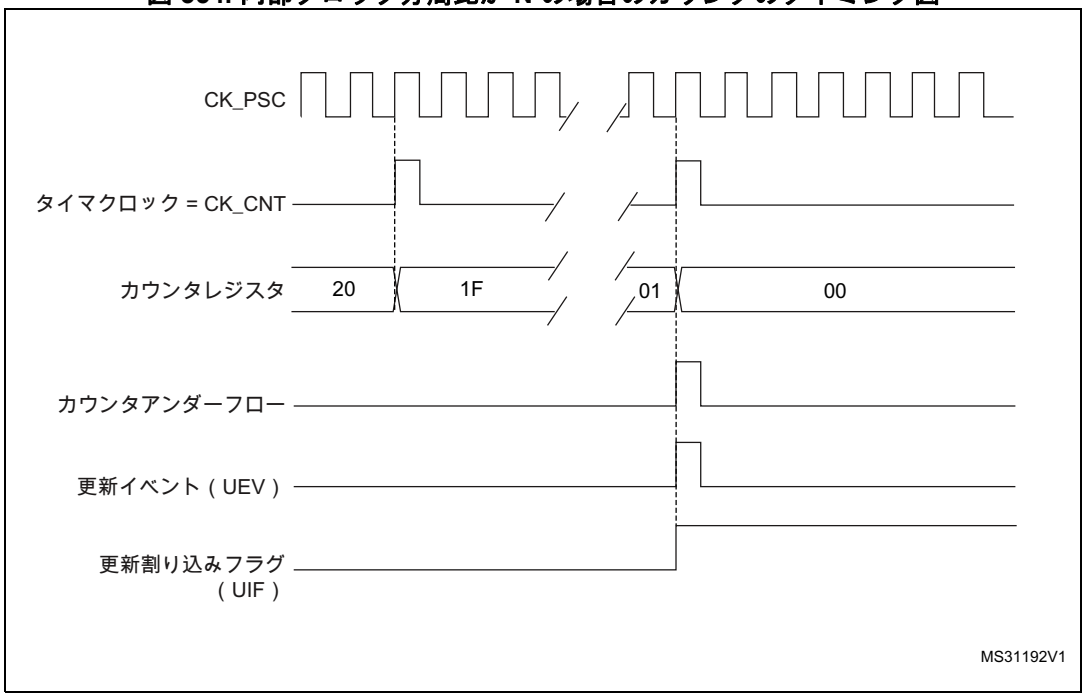


図 355. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図

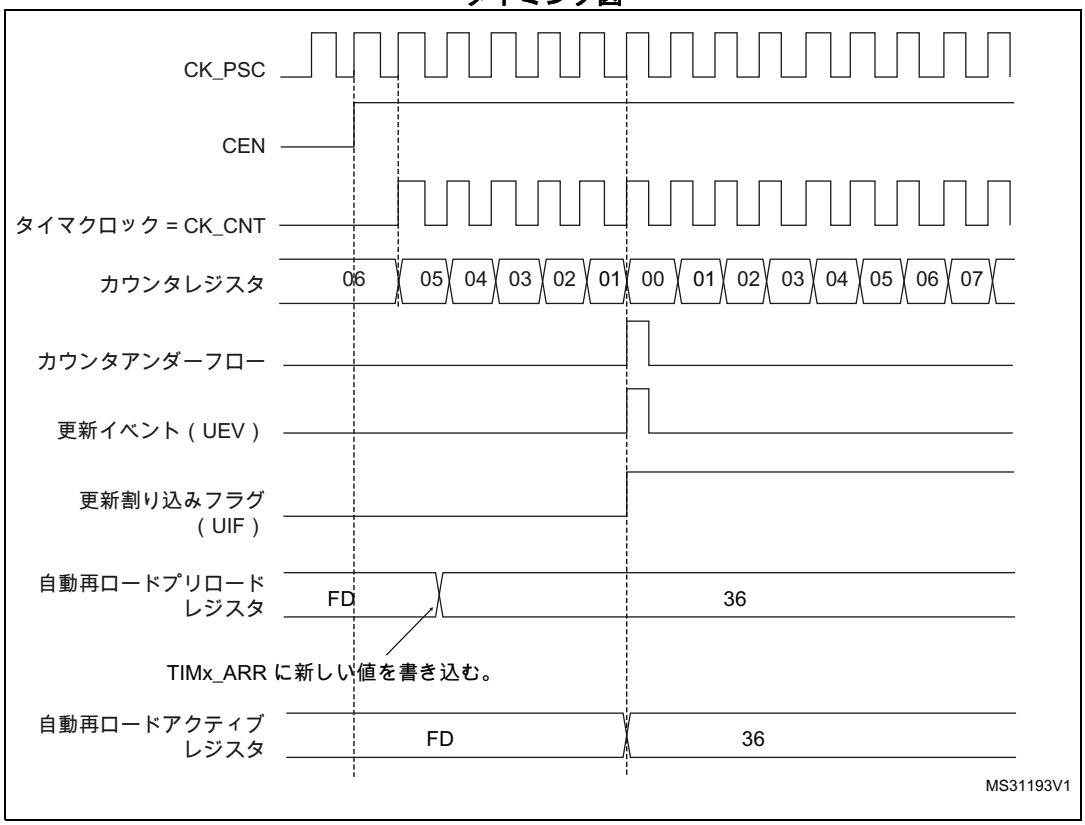
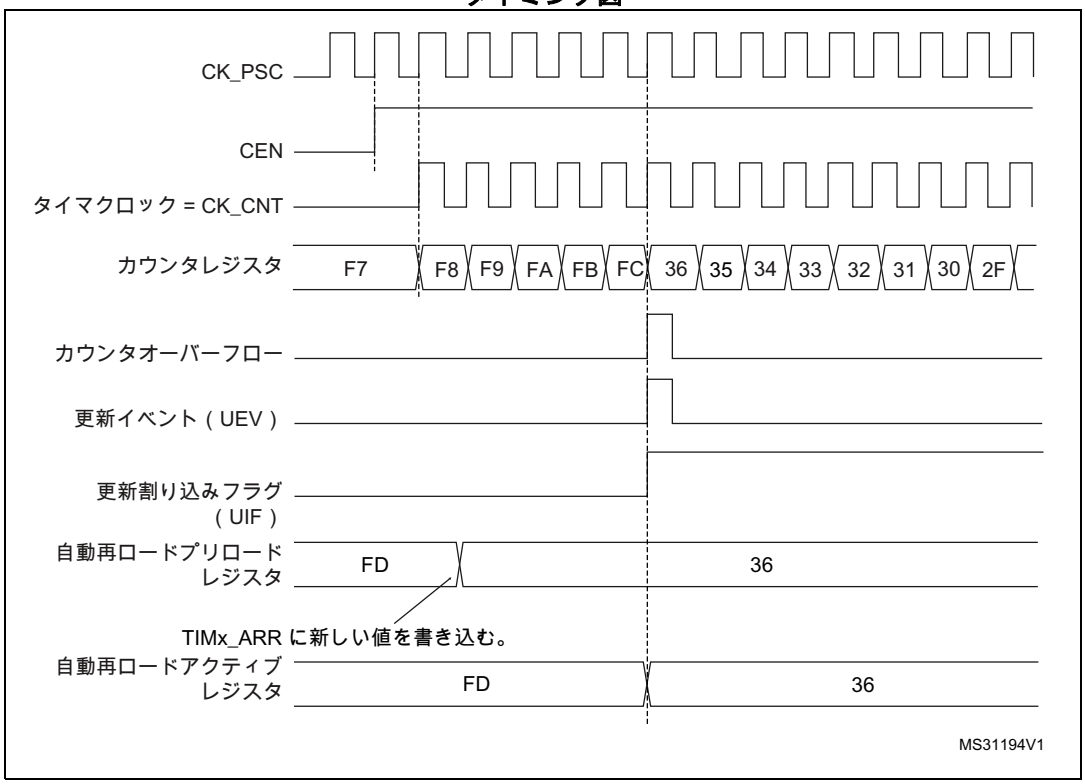


図 356. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



38.3.3 繰り返しカウンタ

*セクション 38.3.1：タイムベースユニット*に、カウンタオーバーフロー/アンダーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが 0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N+1 回目のカウンタオーバーフローまたはアンダーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、次の場合にデクリメントします。

- アップカウントモードで、カウンタオーバーフローごと
- ダウンカウントモードで、カウンタアンダーフローごと
- センターアラインモードで、カウンタオーバーフローとカウンタアンダーフローごと最大繰り返し回数は 32768 PWM サイクルに限られますが、PWM 周期ごとにデューティサイクルを 2 回更新することが可能になります。センターアラインモードで比較レジスタの値を PWM 周期あたり 1 回のみ更新するときには、パターンが対称なので、最大精度は $2xT_{ck}$ です。

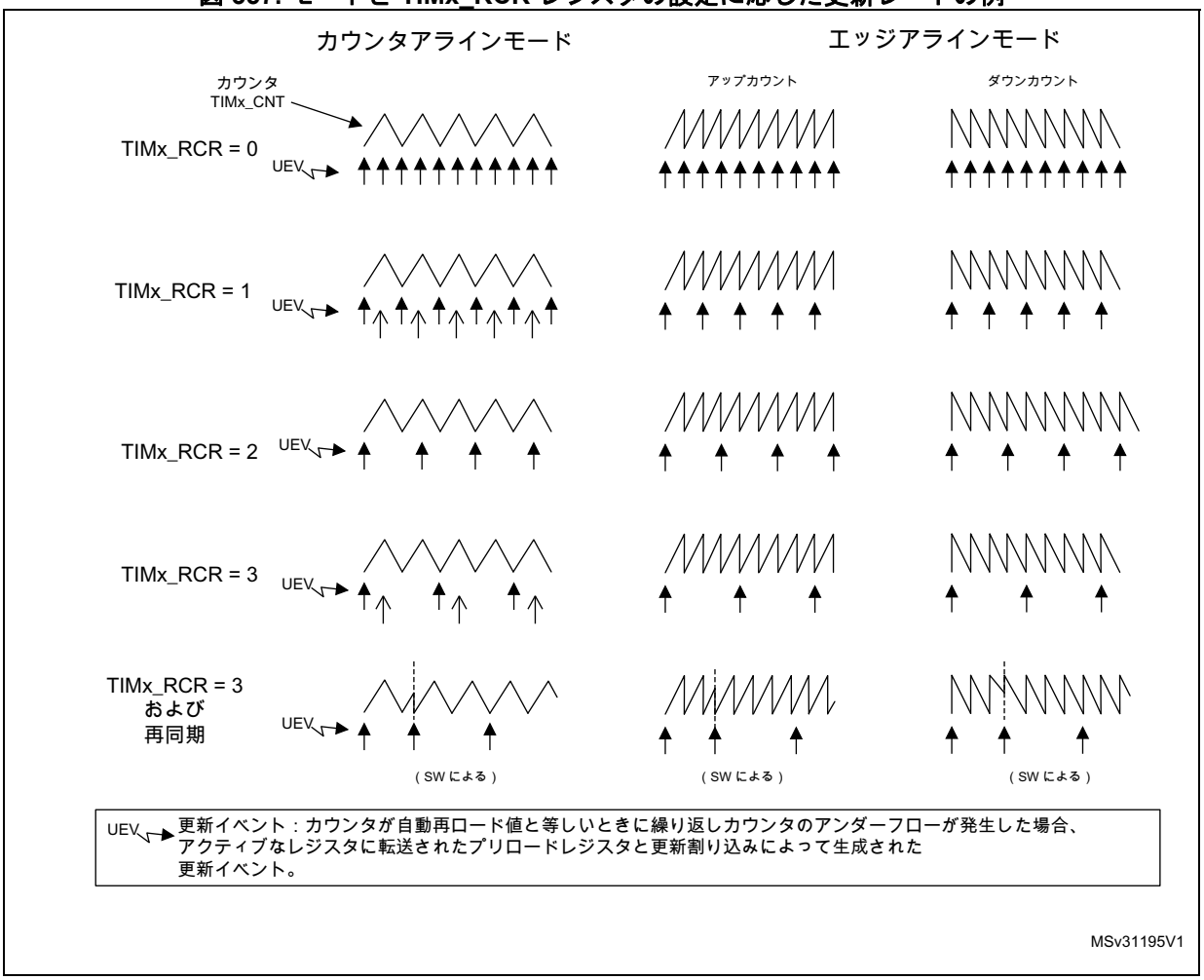
繰り返しダウンカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（*図 357* を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

センターアラインモードでは、RCR が奇数の場合、RCR レジスタが書き込まれたタイミングおよびカウンタが開始されたタイミングに応じてオーバーフローまたはアンダーフロー時に更新イベント

が発生します。カウントの開始前に RCR が書き込まれた場合は、オーバーフローで、UEV が発生します。カウントの開始後に RCR が書き込まれた場合は、アンダーフローで UEV が発生します。

たとえば、RCR = 3 の場合、RCR の書き込みタイミングに応じて 4 回目のオーバーフローイベントまたはアンダーフローイベントごとに UEV が発生します。

図 357. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



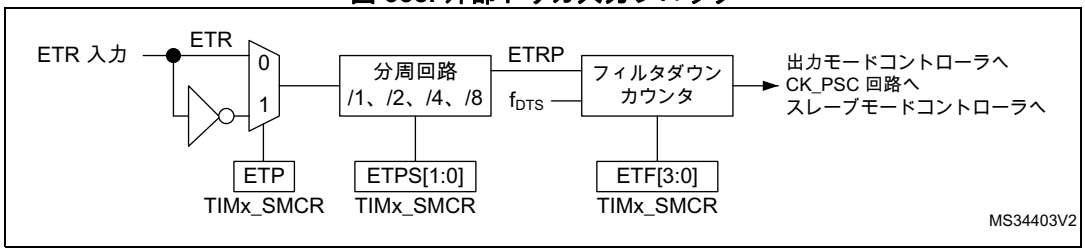
38.3.4 外部トリガ入力

タイマには外部トリガ入力 ETR 機能があります。以下の目的で使用できます。

- 外部クロック（外部クロックモード 2、*セクション 38.3.5* を参照）
- スレーブモードのトリガ（*セクション 38.3.26* を参照）
- サイクルごとの電流調整の PWM リセット入力（*セクション 38.3.7* を参照）

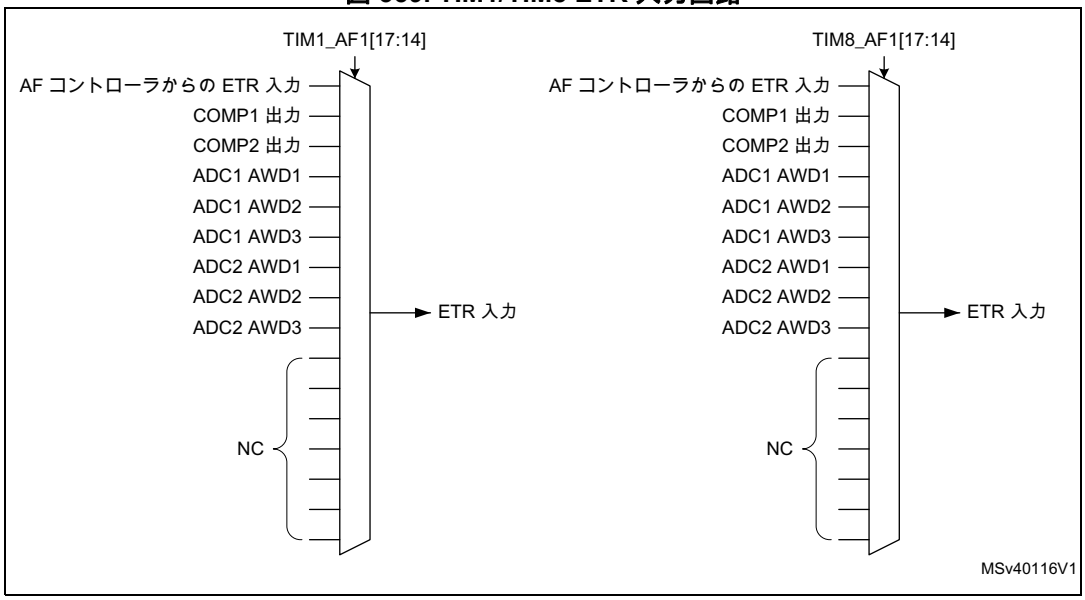
以下の*図 358* は、ETR の入力条件付けについて説明しています。入力の極性は、TIMxSMCR レジスタの ETP ビットで定義されています。トリガは ETPS[1:0] ビットフィールドでプログラムされた分周器でプリスケールし、ETF[3:0] ビットフィールドでデジタル的にフィルタリングすることができます。

図 358. 外部トリガ入力ブロック



ETR は、入力ピン（デフォルト設定）、コンパレータ出力およびアナログウォッチドッグといった複数のソースから入力されます。選択は ETRSEL[3:0] ビットフィールドによって行われます。

図 359. TIM1/TIM8 ETR 入力回路



38.3.5 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

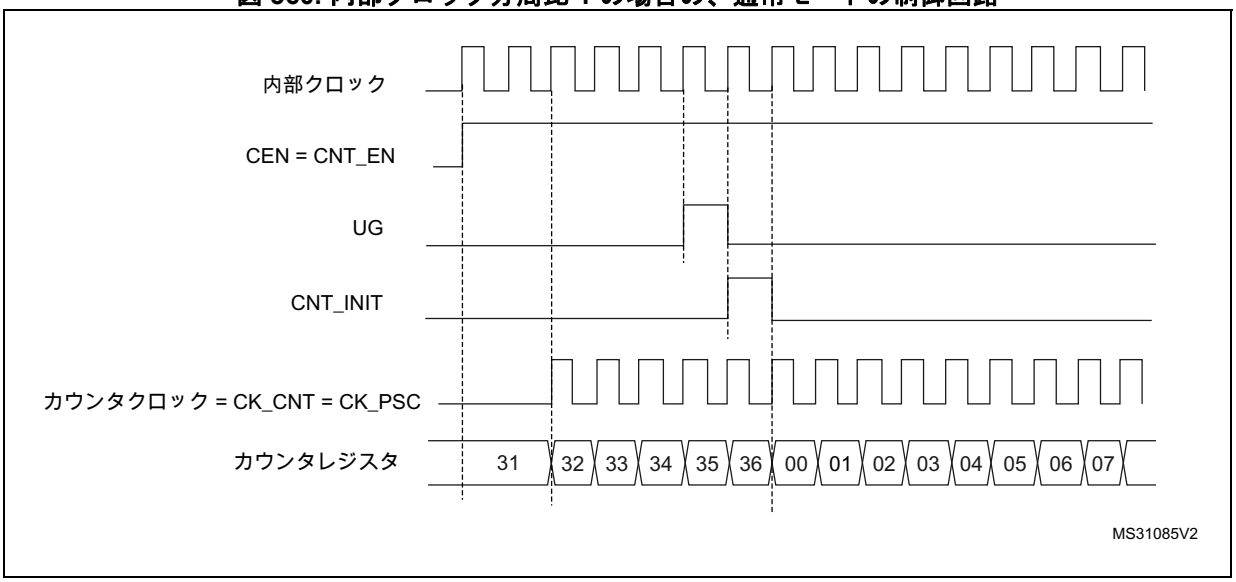
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 外部クロックモード 2：外部トリガ入力 ETR
- エンコーダモード

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN、DIR（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 360 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

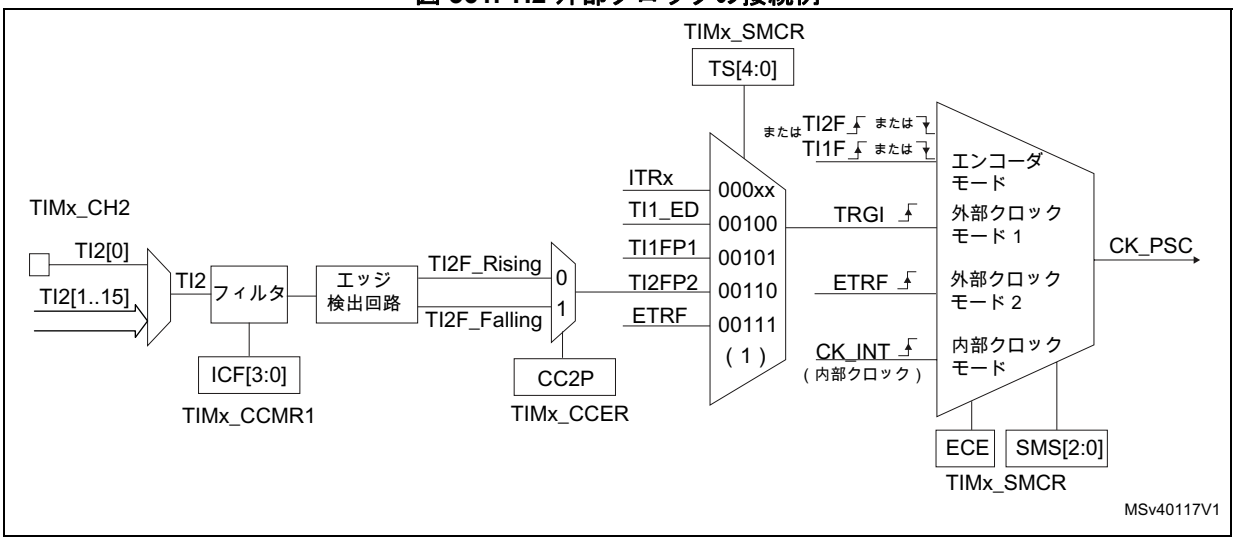
図 360. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 361. TI2 外部クロックの接続例



1. 01000 から 11111 の範囲のコードは予約済み

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

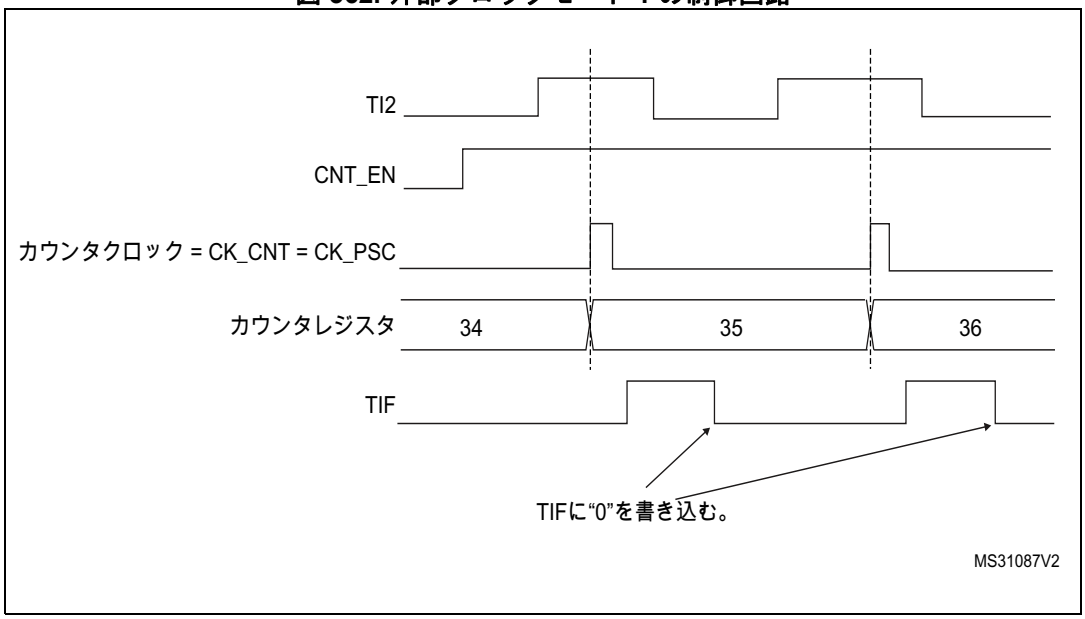
1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
3. CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 362. 外部クロックモード 1 の制御回路



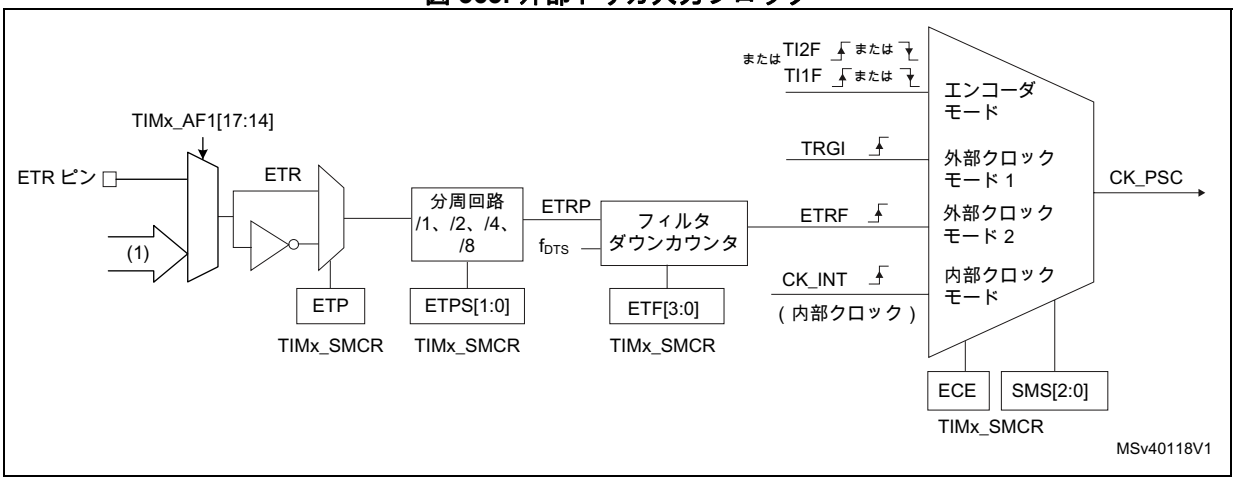
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 363 に、外部トリガ入力ブロックの概要を示します。

図 363. 外部トリガ入力ブロック



1. *図 359：TIM1/TIM8 ETR 入力回路*を参照してください。

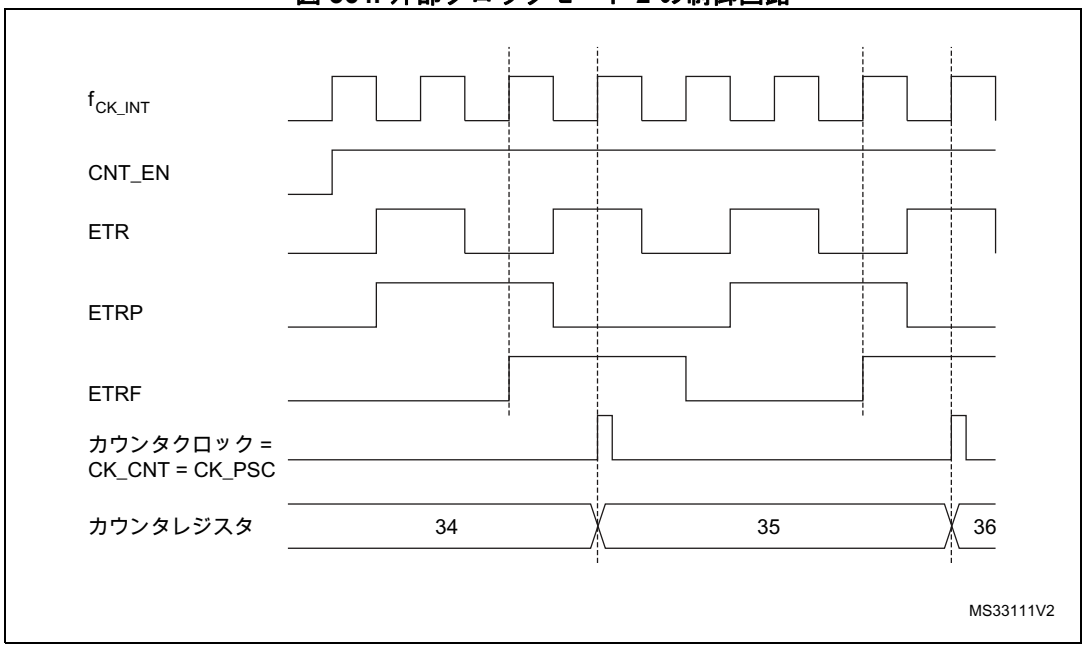
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 364. 外部クロックモード 2 の制御回路



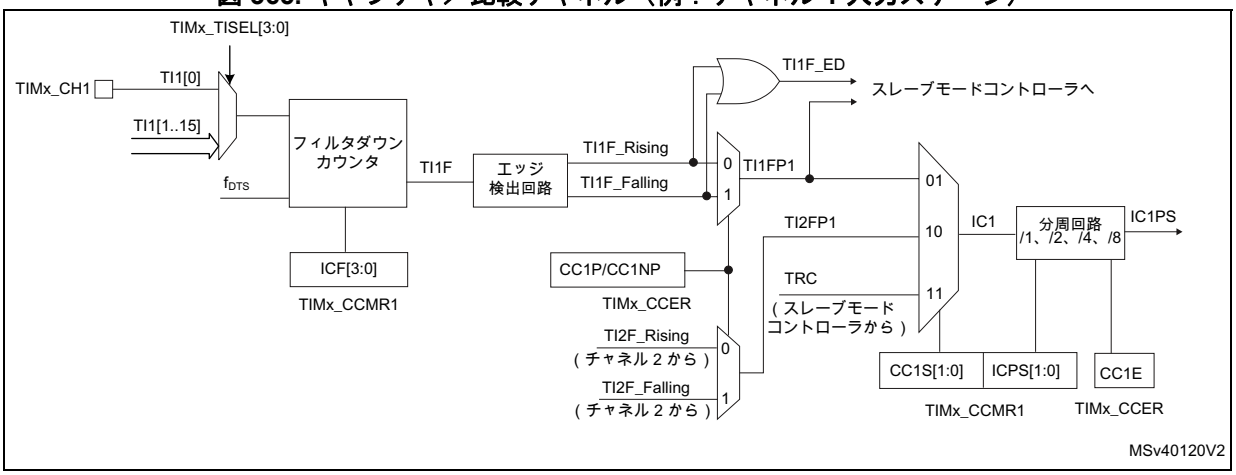
38.3.6 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（チャネル 5 および 6 を除くデジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 365 から *図 368* に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 365. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 366. キャプチャ／比較チャネル 1 メイン回路

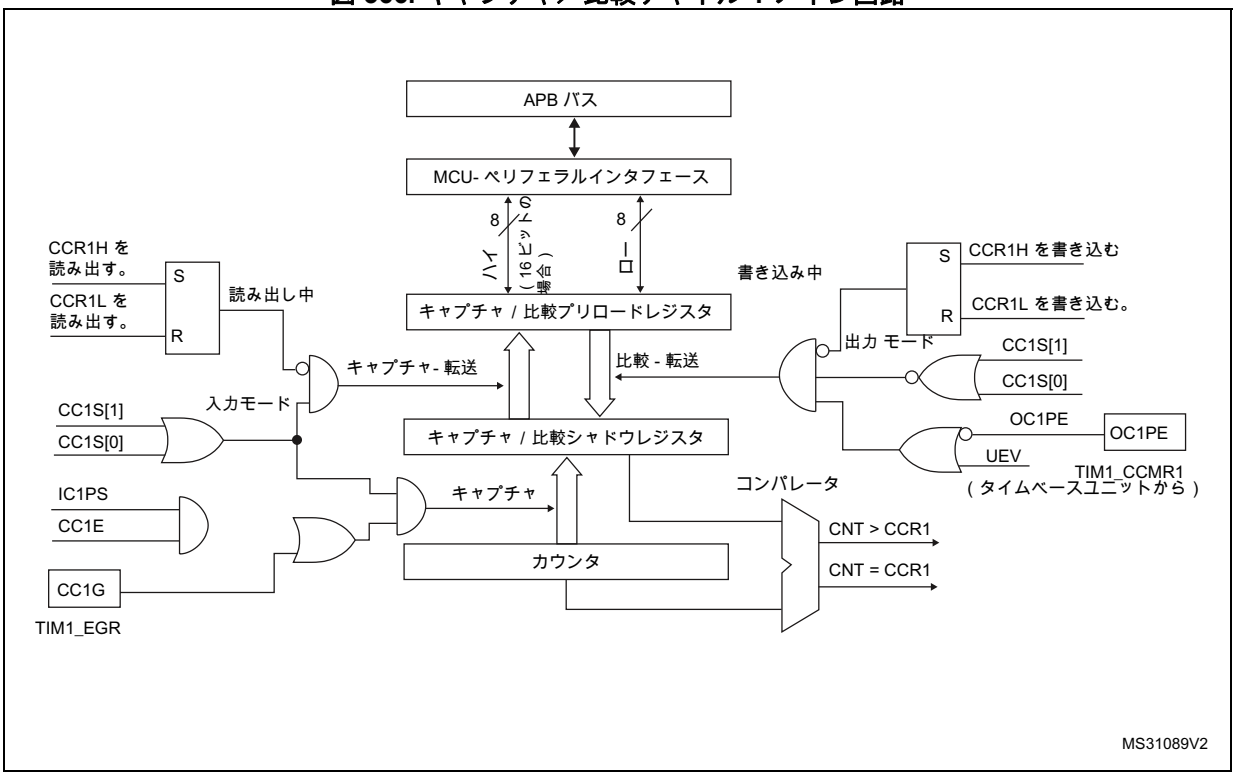
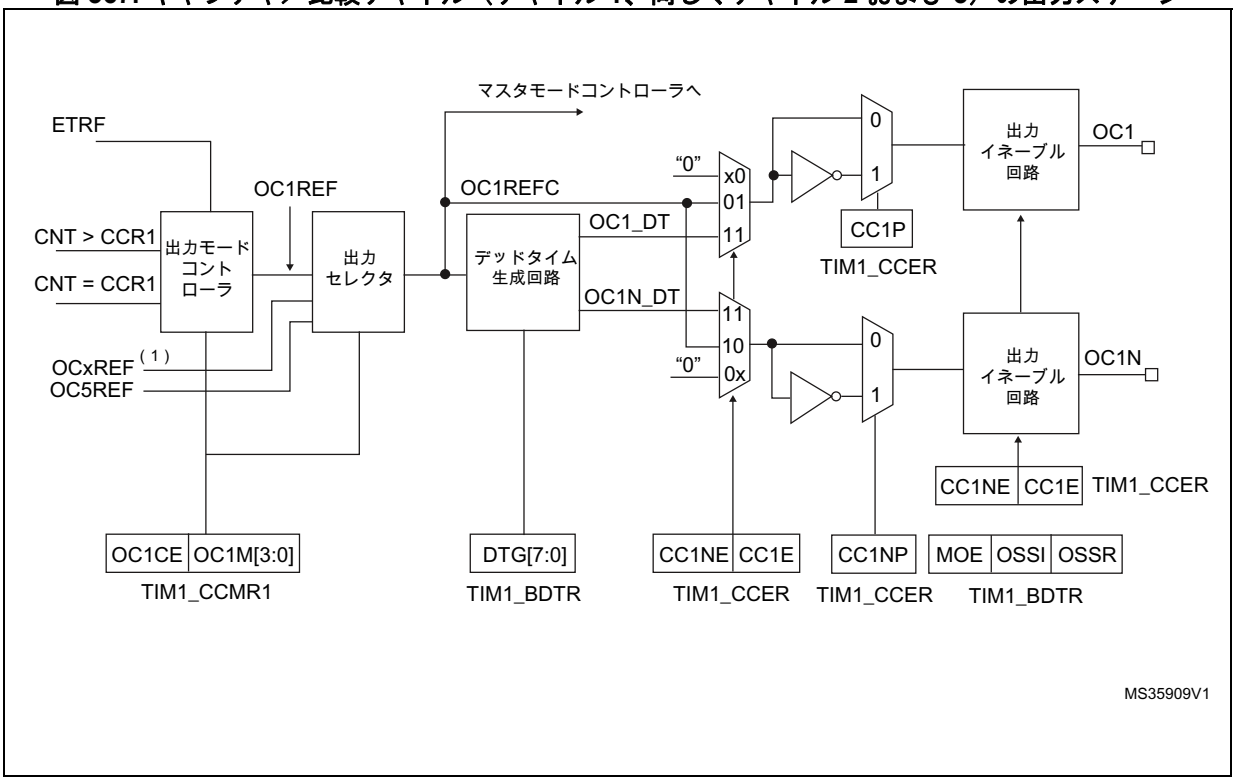


図 367. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ



1. OCxREF、ここで x は相補チャネルのランク

図 368. キャプチャ／比較チャネル（チャネル 4）の出力ステージ

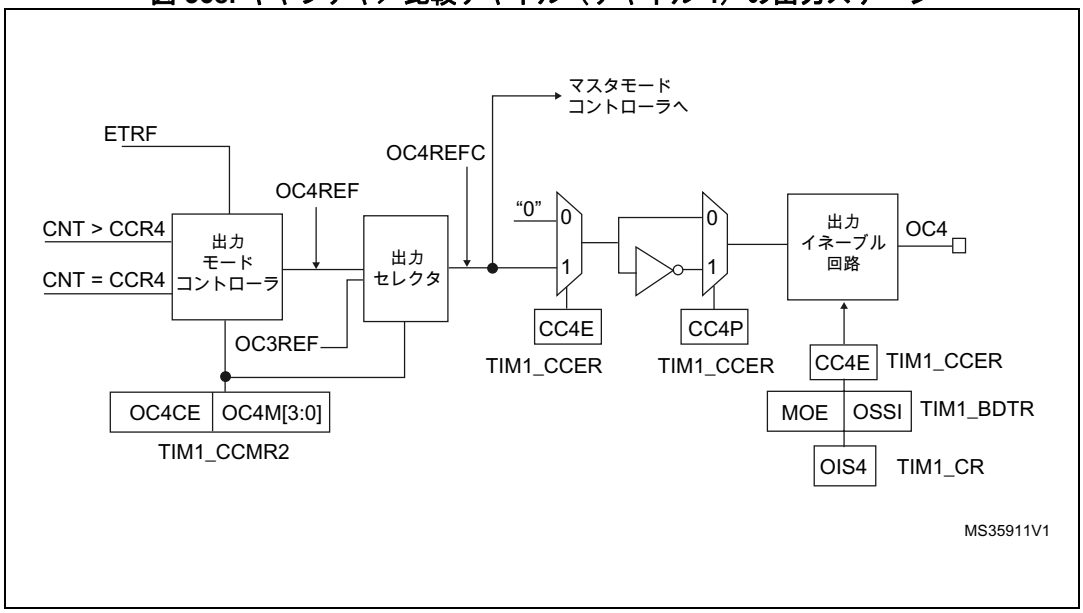
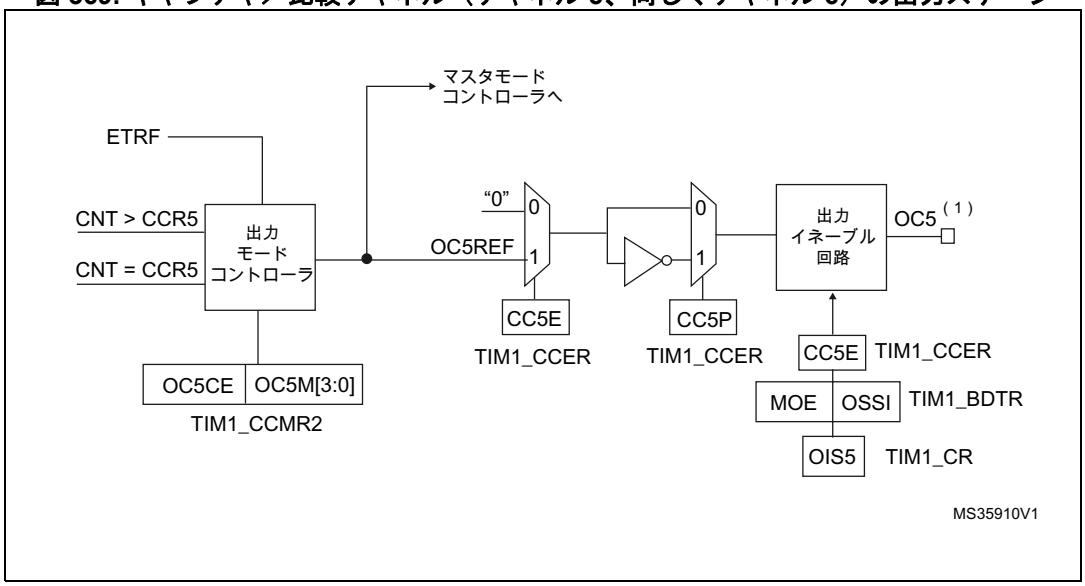


図 369. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ



1. 外部的には使用できません。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

38.3.7 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

- アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
- タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
- TI1 チャネルのアクティブ変化のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“0”を書き込みます（この場合、立ち上がりエッジの選択）。
- 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
- TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
- 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

38.3.8 PWM 入力モード

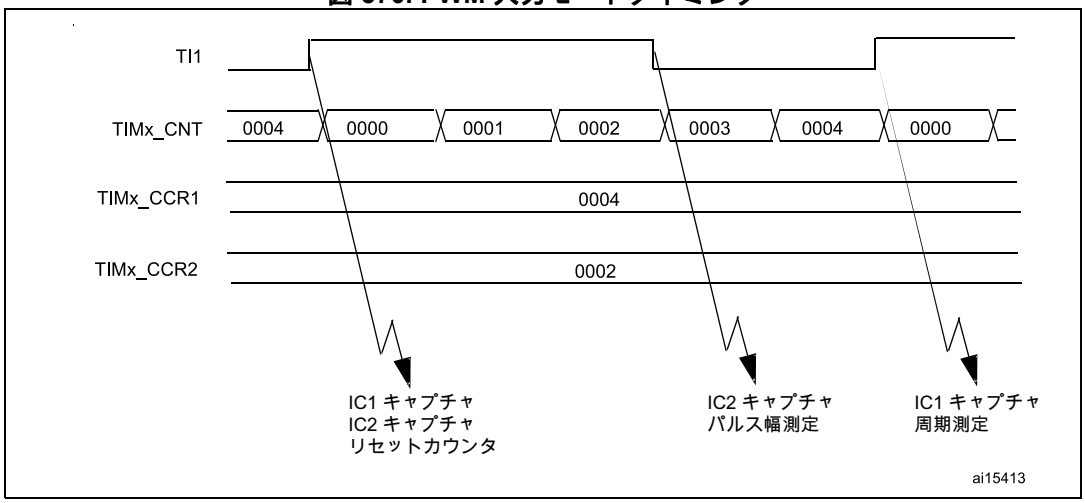
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

- TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
- CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
- TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
- CC2P ビットと CC2NP ビットに CC2P/CC2NP=“10”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
- TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
- TIMx_SMCR レジスタの SMS ビットに 0100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
- TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 370. PWM 入力モードタイミング



38.3.9 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 0101 を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“0100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

38.3.10 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。マイクロコントロール内でチャネル 5 および 6 のみが使用可能である場合に、チャネル 1 から 4 を出力できます（たとえば、合成波形生成または ADC トリガのため）。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

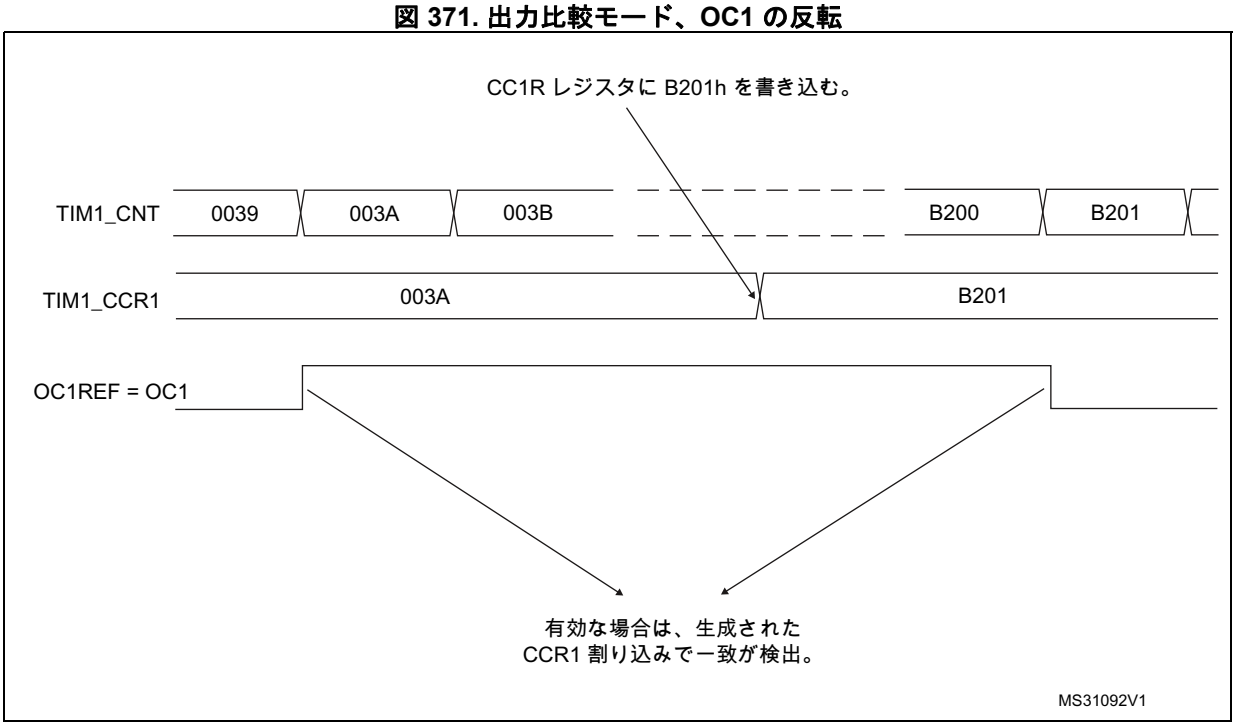
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは、単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 0011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 371* に示します。

図 371. 出力比較モード、OC1 の反転



38.3.11 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWMモード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

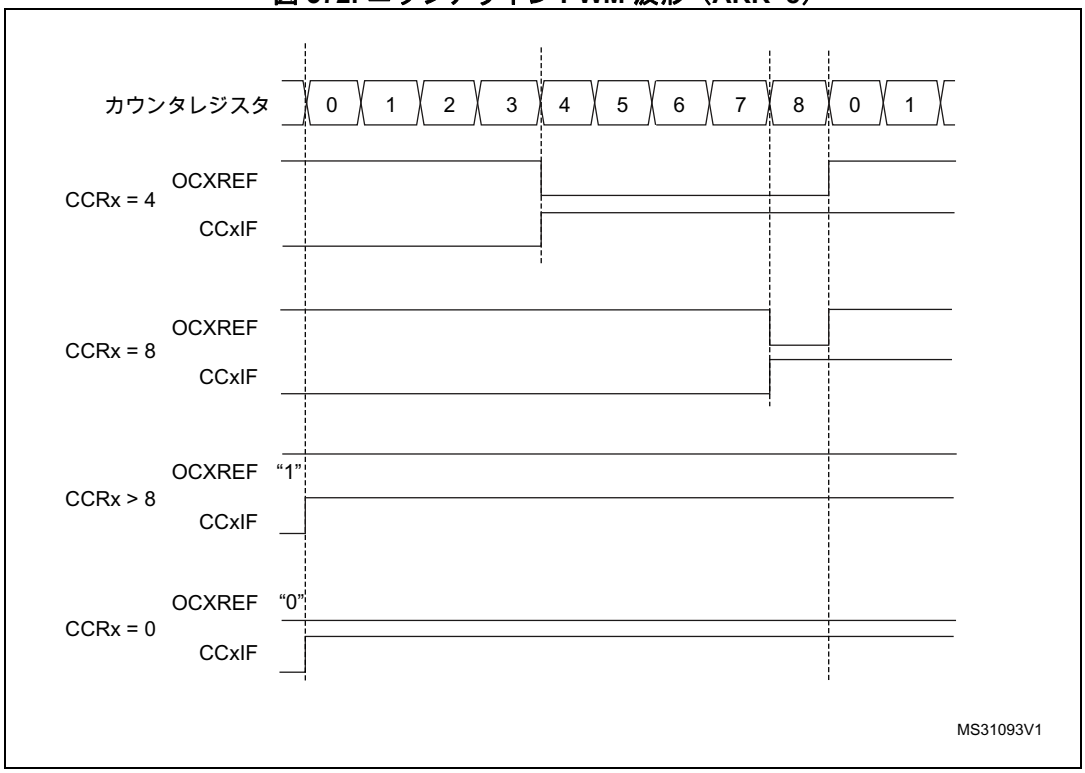
PWM エッジアラインモード

- アップカウント構成

  TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。*1460 ページのアップカウントモード*を参照してください。

  次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 372* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 372. エッジアライン PWM 波形（ARR=8）



- ダウンカウント構成

  TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。*1464 ページのダウンカウントモード*を参照してください。

  PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、OCxREF は“1”です。このモードでは、0 % の PWM 信号を生成することはできません。

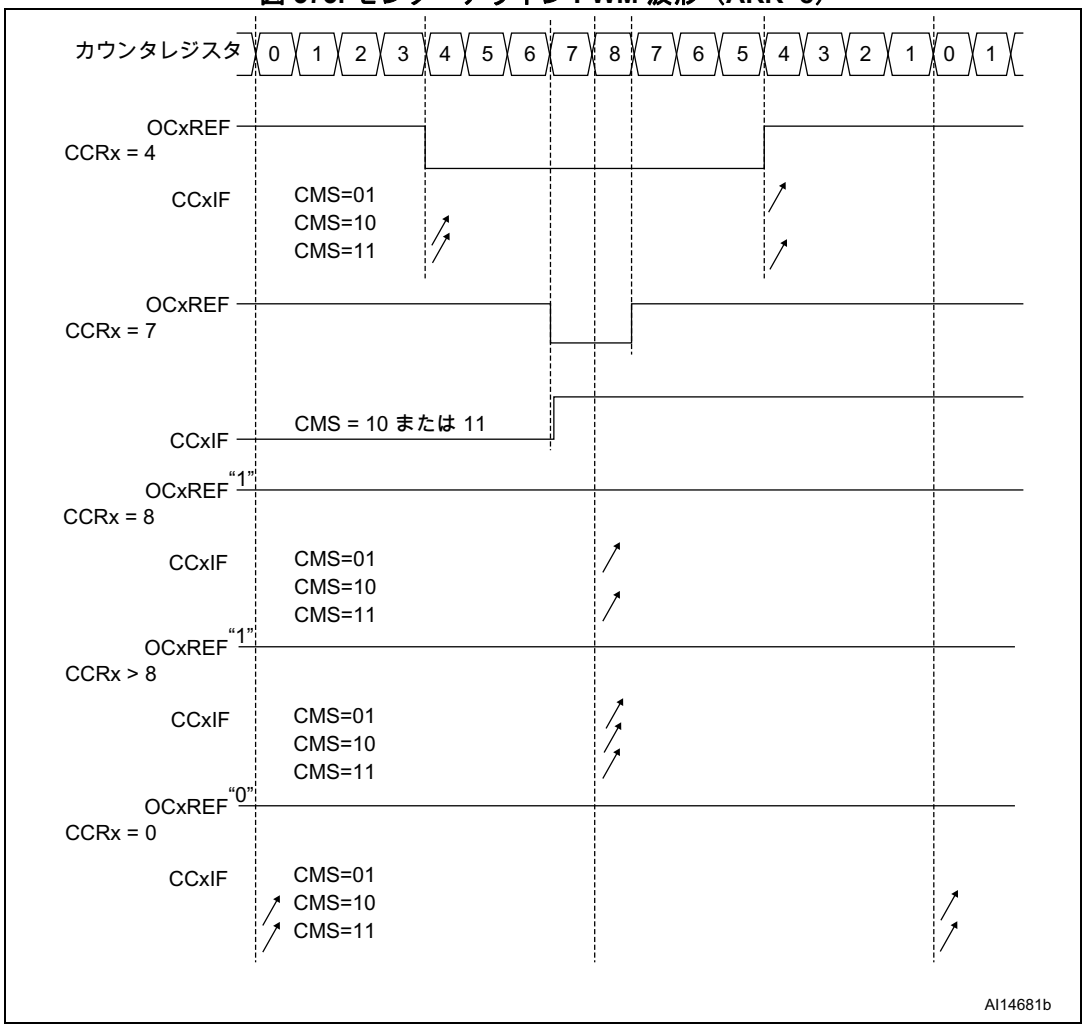
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての構成は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。*1467 ページのセンターアラインモード（アップ／ダウンカウント）*を参照してください。

図 373 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは､TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 373. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値をカウンタに書き込んだ場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書き込みを行わないことです。

38.3.12 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

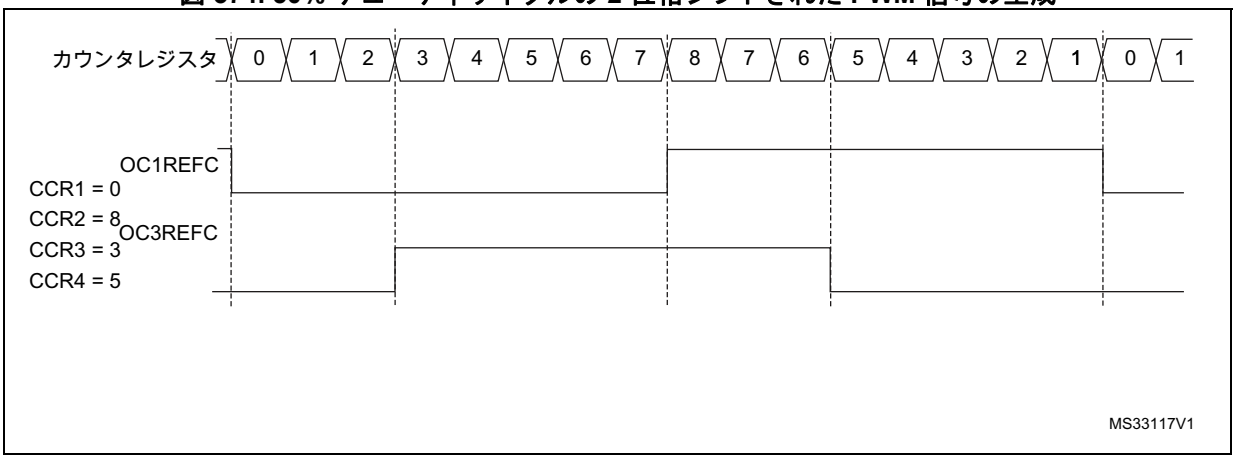
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

特定のチャネルが非対称の PWM チャネルとして使用されると、その相補チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 1 の結果として得られる OC2REFC 信号を出力できます。

図 374 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。これにより、デッドタイムジェネレータとともにフルブリッジ位相シフト DC-DC コンバータを制御できます。

図 374. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



38.3.13 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラム可能です。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

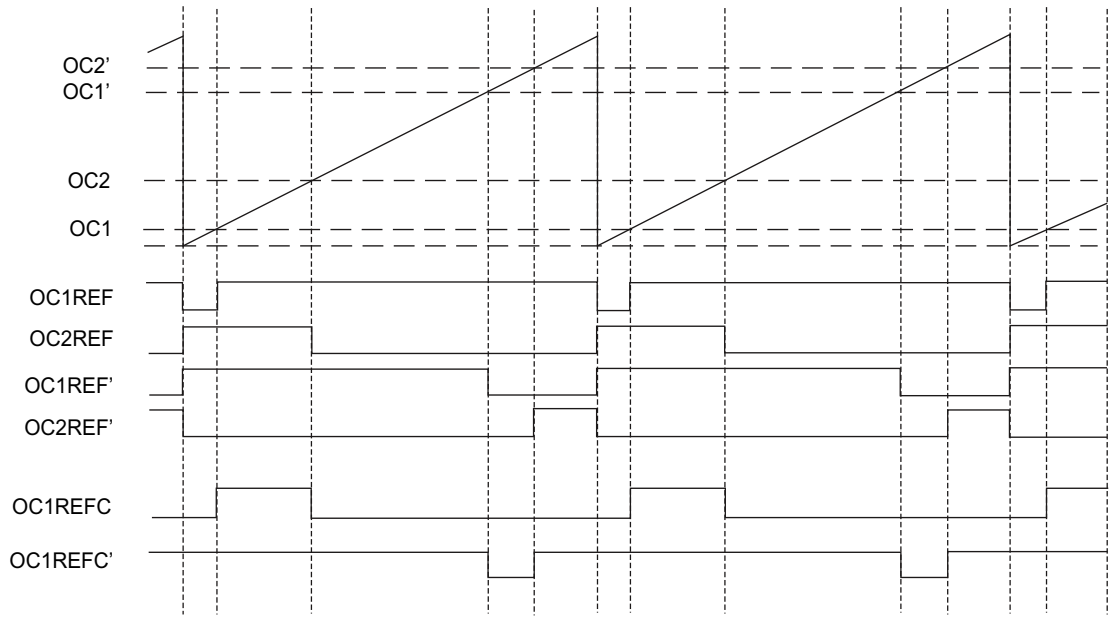
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 375 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 375. チャネル 1 および 3 における組み合わせ PWM モード



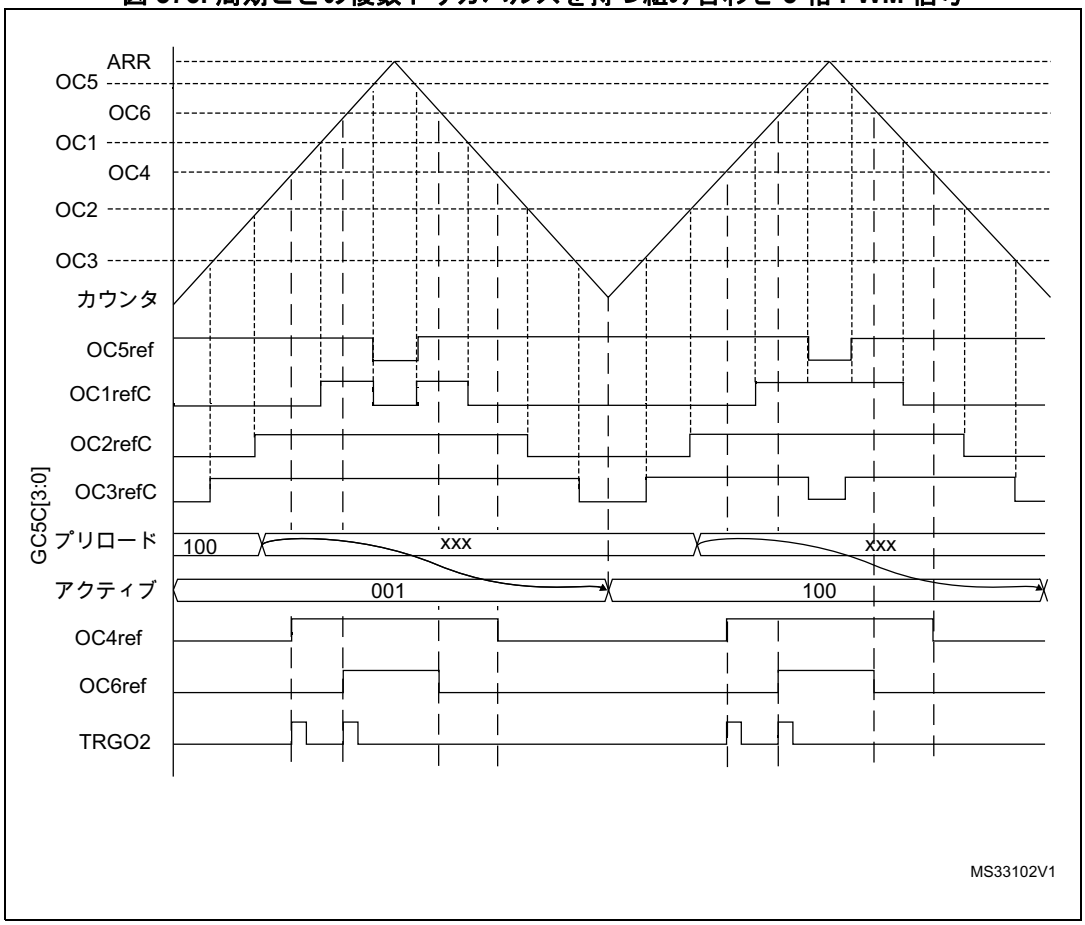
38.3.14 組み合わせ 3 相 PWM モード

組み合わせ 3 相 PWM モードでは、パルスの中間で論理積を取った単一のプログラム可能な信号とともに 1 つから 3 つのセンターアライン PWM 信号を生成できます。結果として得られる組み合わせ信号の定義には、OC5REF 信号が使用されます。TIMx_CCR5 の 3 ビット GC5C[3:1] では、OC5REF を組み合わせる基準信号を選択できます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の AND による論理結合から生成されます。

- GC5C1 がセットされると、OC1REFC は TIMx_CCR1 および TIMx_CCR5 によって制御されます。
- GC5C2 がセットされると、OC2REFC は TIMx_CCR2 および TIMx_CCR5 によって制御されます。
- GC5C3 がセットされると、OC3REFC は TIMx_CCR3 および TIMx_CCR5 によって制御されます。

組み合わせ 3 相 PWM モードは、少なくとも3 ビット GC5C[3:1] の 1 つをセットすることでチャネル 1 から 3 で個別に選択できます。

図 376. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号



TRGO2 波形は、特定の 3 相 PWM 信号での ADC の同期方法を示します。詳細については、*セクション 38.3.27 : ADC の同期*を参照してください。

38.3.15 相補出力とデッドタイム挿入

高機能制御タイマ（TIM1/TIM8）は、2 つの相補信号を出力して、出力時のスイッチオンおよびスイッチオフを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力の極性（主出力 OCx または補 OCxN）は、出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書き込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、*表 313：1536 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット* を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 377. デッドタイム挿入のある相補出力

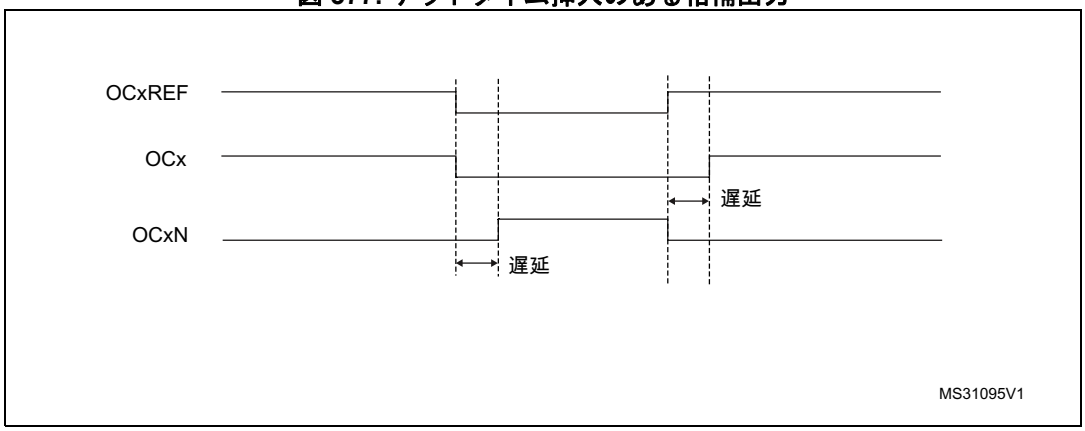


図 378. 負のパルスより長い遅延があるときのデッドタイムの波形

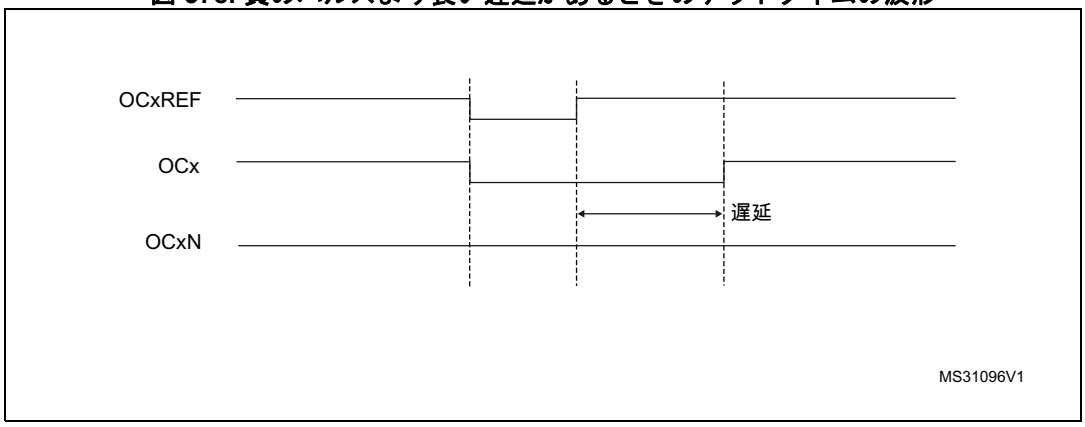
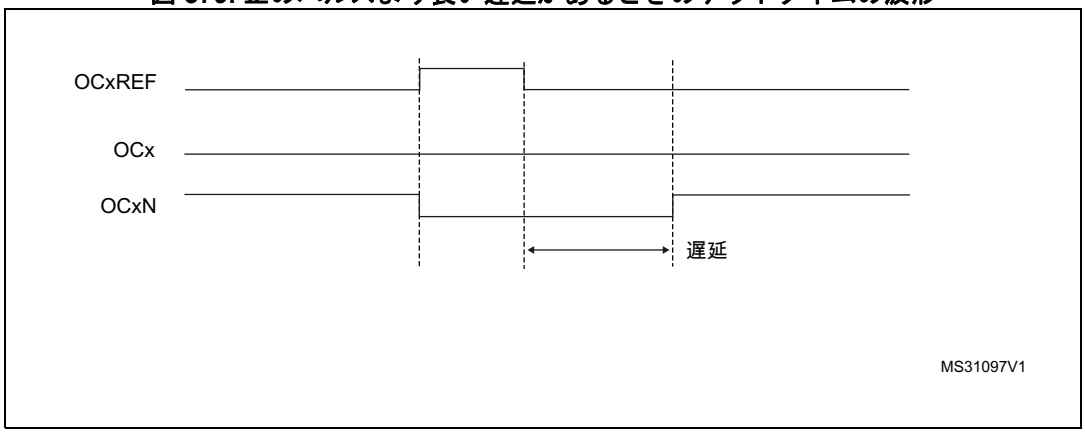


図 379. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、*セクション 38.4.18：TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）*を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： ***OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx とOCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。***

38.3.16 ブレーク機能の使用

ブレーク機能の目的は、TIM1 および TIM8 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。2 つのブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。出力の遮断をトリガするために、いくつかの内部 MCU イベントを選択することも可能です。

このブレークには、2 つのチャネルがあります。システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵コンパレータ）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できるブレーク チャネル。アプリケーション障害のみを含み、出力をインアクティブ状態に強制できるブレーク2 チャネル。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出力を両方ともアクティブレベルにセットすることはできません。詳細については、*表 313： 1536 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット* を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。TIMx_BDTR レジスタの BKE および BKE2 ビットをセットすることによって、ブレーク機能を有効にできます。ブレーク入力の極性は、同じレジスタの BKP および BKP2 ビットを設定することによって選択できます。BKEx と BKPx は、同時に変更できます。BKEx および BKPx ビットが書き込まれるとき、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書き込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 が書き込まれた場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、非同期信号を書き込んで、同期信号を読み出すからです。

ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力
  - システムブレーク：
    - Cortex®-M7 LOCKUP 出力
    - PVD 出力
    - SRAM パリティエラー信号
    - フラッシュ ECC エラー
    - CSS 検出回路によって生成されたクロック障害イベント

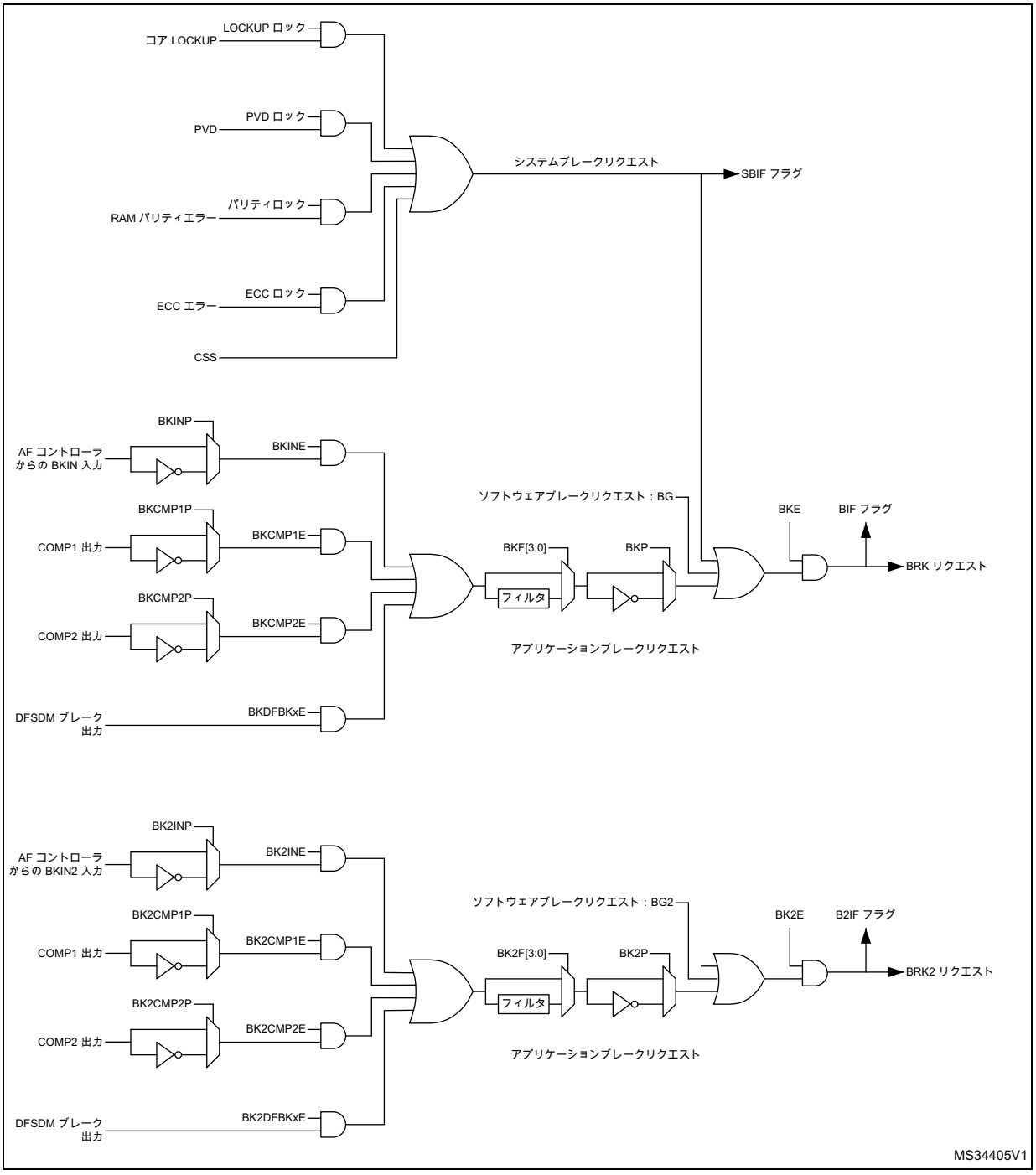
ブレーク 2（BRK2）のソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- コンパレータ出力からの内部ソース

ブレークイベントは、TIMx_EGR レジスタで BG および B2G ビットを使用して、ソフトウェアによって生成することもできます。

以下の*図 380* のとおり、すべてのソースはタイマ BRK または BRK2 入力がある前に論理和がとられます。

図 380. ブレーク および ブレーク 2 回路の概要



注： 非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。

ブレークが 1 つ発生すると（ブレーク入力の 1 つで選択されたレベル）、

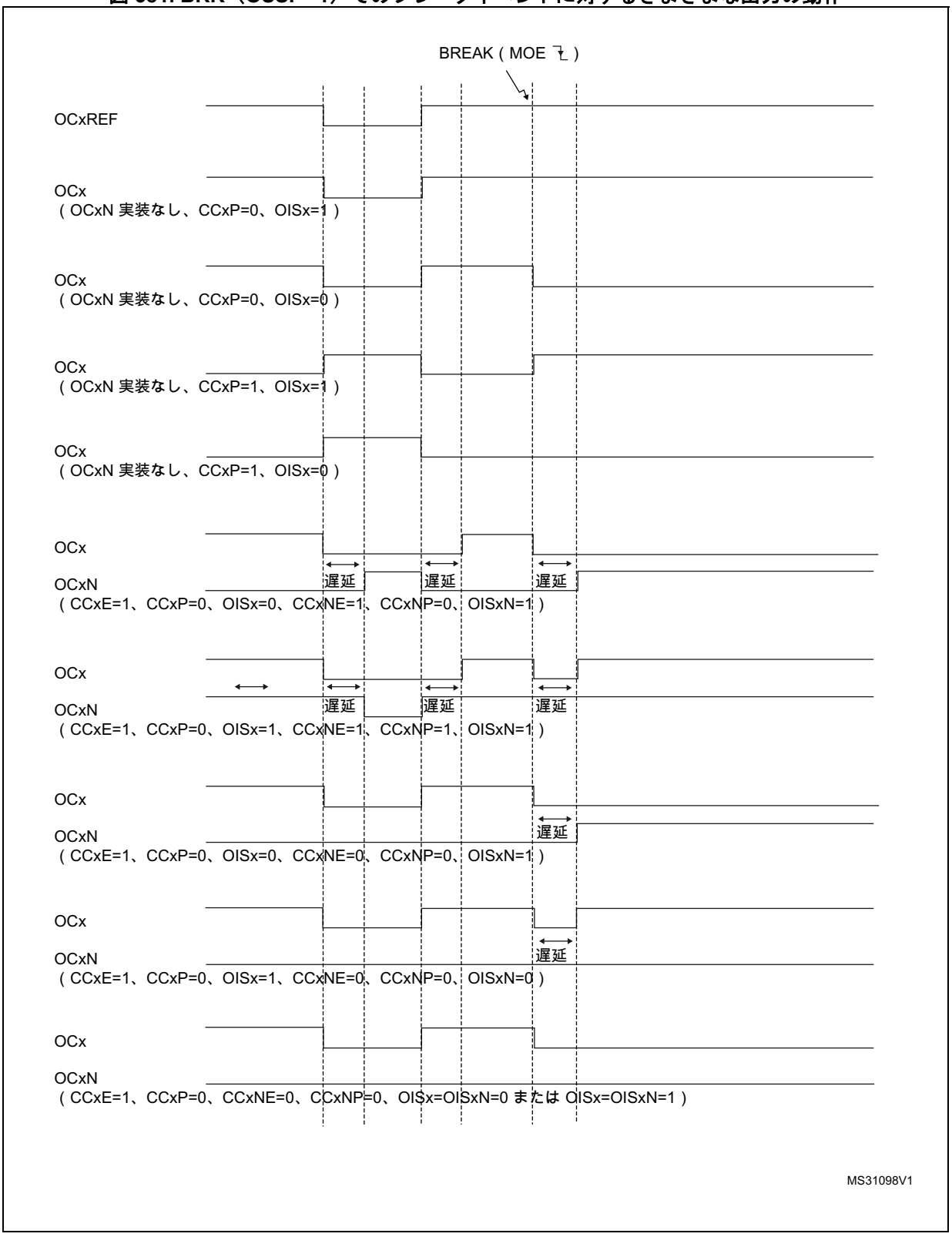
- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、GPIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合に有効です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。
  - OSSI=0 の場合、タイマは出力の制御（ハイインピーダンス状態を強制する GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。
- ブレーク状態フラグ（TIMx_SR レジスタの SBIF、BIF、および B2IF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合、割り込みが生成されます。TIMx_DIER レジスタのBDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント（UEV）で再び自動的にセットされます。たとえば、これを使用してレギュレーションを行うことができます。そうでない場合、MOE ビットはアプリケーションが再び“1”をセットするまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

注： ***ブレーク入力は、信号レベルに対してアクティブです。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF および B2IF をクリアできません。***

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書き込み保護機能を設けてあります。これにより、いくつかのパラメータ（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。*セクション 38.4.18：TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）*を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 381 に、ブレークに対する出力の動作例を示します。

図 381. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作



2 つのブレーク入力は、タイマ出力で異なる動作を示します。

- BRK 入力は、無効化（インアクティブ状態）するか、PWM 出力を強制的に事前定義した安全な状態に移行できます。
- BRK2 は、PWM 出力の無効化（インアクティブ状態）のみ可能です。

表 310 に示すように、BRK の優先順位は BRK2 入力よりも高いです。

注： ***BRK2 は OSSR = OSSI = 1 の場合にのみ使用してください。***

表 310. タイマ出力と BRK/BRK2 入力の動作

BRK	BRK2	タイマ出力状態	通常の使用例	
			OCxN 出力（ローサイドスイッチ）	OCx 出力（ハイサイドスイッチ）
アクティブ	X	– インアクティブから強制される出力状態（デッドタイム後） – OSSI = 0 の場合、出力は無効（GPIO ロジックが制御を引き継ぐ）	デッドタイム挿入後 ON	OFF
インアクティブ	アクティブ	インアクティブ	OFF	OFF

図 382 では、BRK および BRK2 入力で信号がアクティブな場合の、OCx および OCxN 出力の動作の例を示します。この場合、両方の出力がアクティブハイ極性になります（TIMx_CCER レジスタの CCxP = CCxNP = 0）。

図 382. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1）

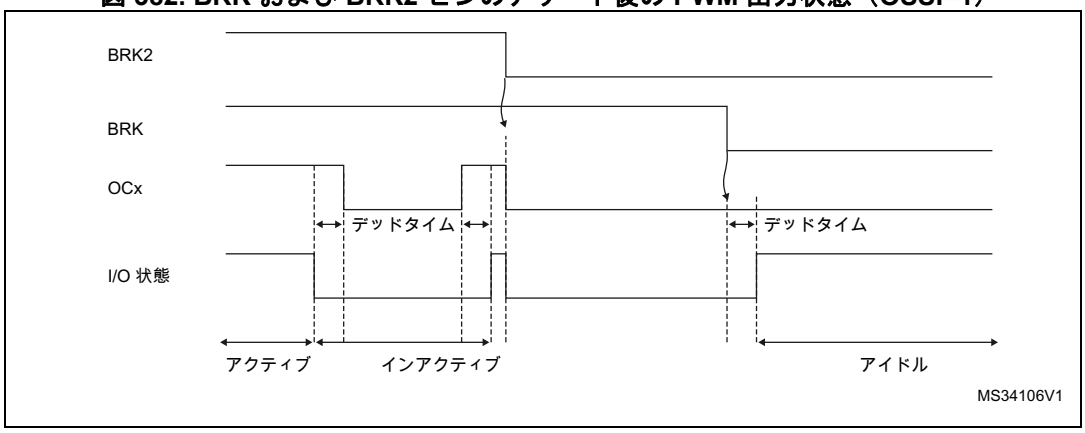
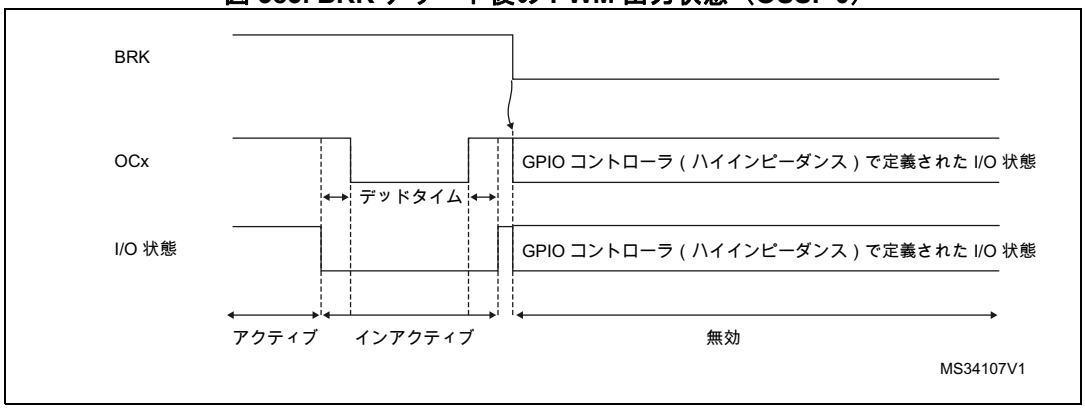


図 383. BRK アサート後の PWM 出力状態（OSSI=0）



38.3.17 双方向ブレーク入力

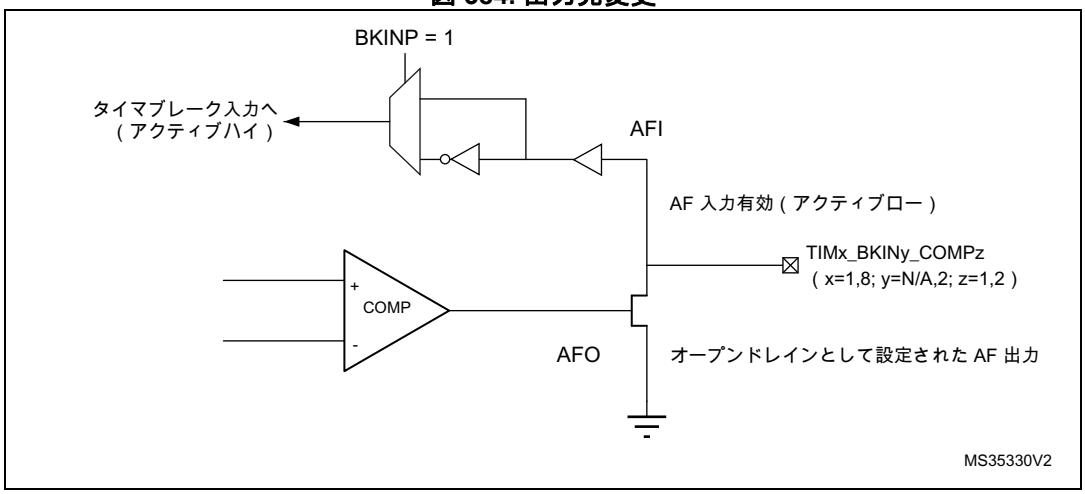
タイマ 1 および 8 には、コンパレータからのレギュラデジタルブレーク入力および内部ブレーク入力イベントの他に、*図 384* に示すとおり、2 つのソースを組み合わせる双方向ブレーク入出力が搭載されています。

TIMx_BKINy_COMPz ピンは、COMPz 出力（オープンドレインで設定）と Timerx の TIMx_BKINy 入力を組み合わせています。次のものを使用できます。

- シングルピンで外部 MCU またはゲートドライバシャットダウン入力に使用できるグローバルブレーク情報

複数の内部および外部ブレーク入力を統合する必要がある場合に、互いに論理和をとり、ブレークイベントをトリガする内部コンパレータおよび複数の外部オープンドレインコンパレータ出力

図 384. 出力先変更



38.3.18 外部イベントによる OCxREF 信号のクリア

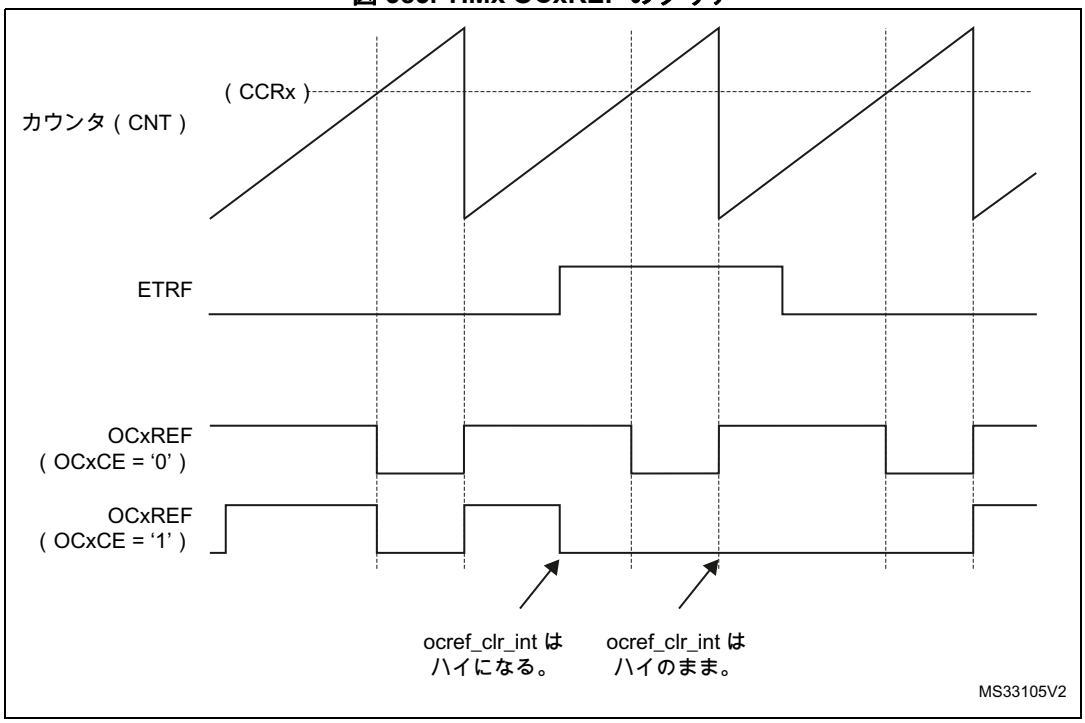
特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。ocref_clr_int は ETRF 信号（フィルタリング後の ETR）に接続されます。

ETRF が選択された場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラはオフ状態に維持します（TIMx_SMCR レジスタの ETPS[1:0] ビットを“00”にセット）。
2. 外部クロックモード 2 を無効にします（TIMx_SMCR レジスタの ECE ビットを“0”にセット）。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、ユーザのニーズに応じて設定できます。

図 385 に、イネーブルビット OCxCE の両方の値について、ETRF 入力がハイレベルになったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 385. TIMx OCxREF のクリア



注： ***100% デューティサイクルの PWM の場合 （CCRx > ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。***

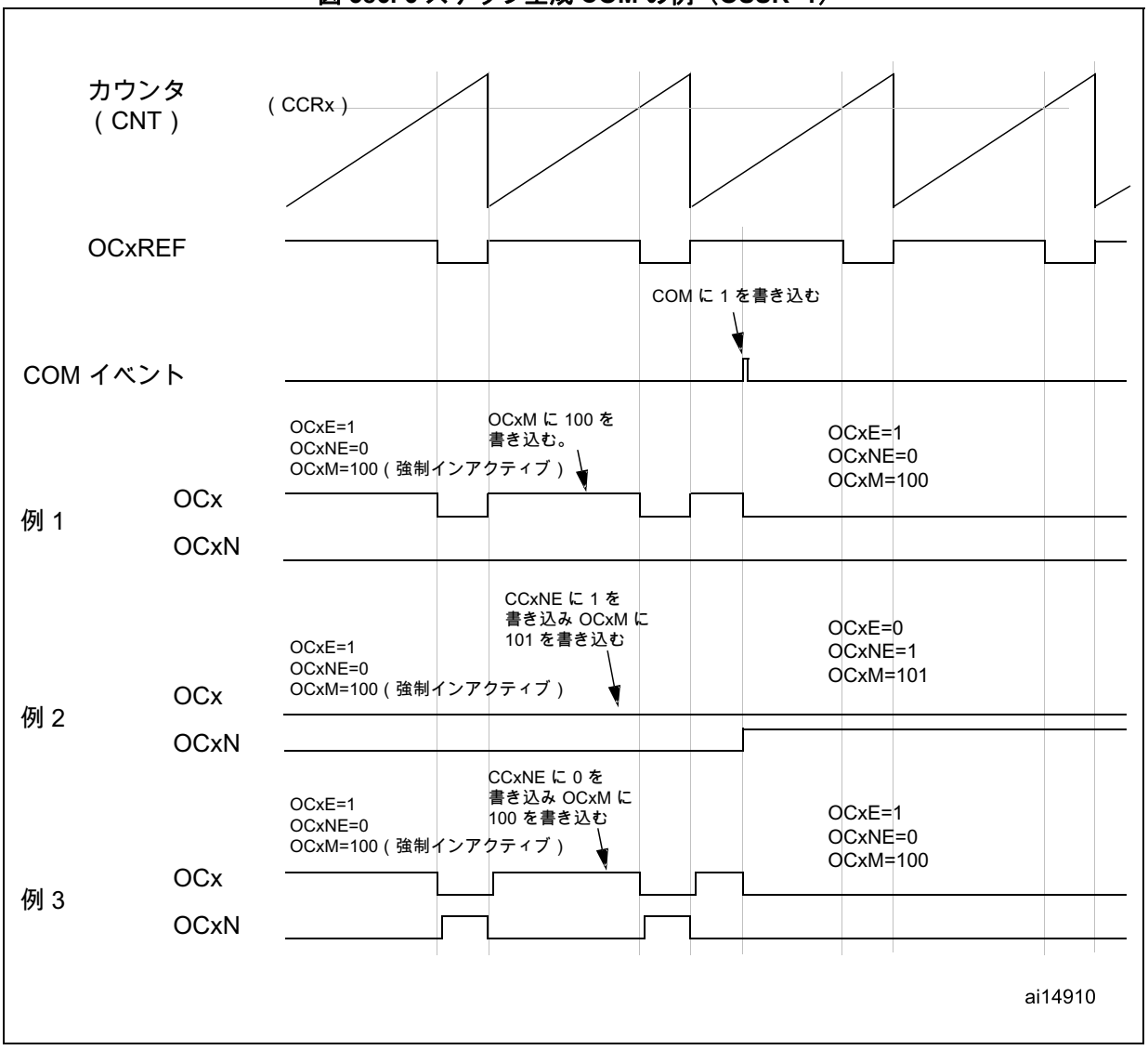
38.3.19 6 ステップ PWM 生成

チャネルで相補出力が使用されているときには、OCxM、CCxE、および CCxNE ビットでプリロードビットが使用できます。プリロードビットは、COM 遷移イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COM は、TIMx_EGR レジスタの COM ビットをセットすることによってソフトウェアによって、またはハードウェアによって（TRGI 立ち上がりエッジで）生成することができます。

フラグは、COM イベントが発生したときにセットされ（TIM x _SR レジスタの COMIF ビット）、これによって割り込み（TIMx_DIER レジスタの COMIE ビットがセットされている場合）または DMA リクエスト（TIMX_DIER レジスタの COMDE ビットがセットされている場合）を生成できます。

図 386 に、COM イベントが発生したときの OCx と OCxN 出力の動作を、3 種類のプログラミング構成の例で示します。

図 386. 6 ステップ生成 COM の例（OSSR=1）



38.3.20 ワンパルスモード

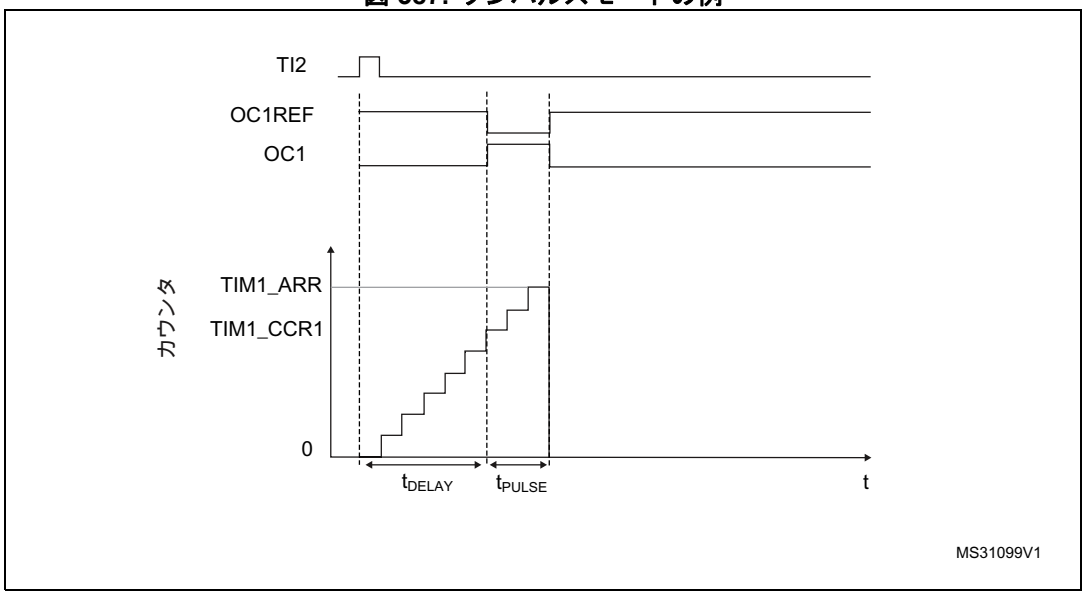
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- アップカウント時：CNT < CCRx ≤ ARR （特に、0 < CCRx）
- ダウンカウント時：CNT>CCRx

図 387. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
- TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
- TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに”1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

38.3.21 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、*セクション 38.3.20* で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

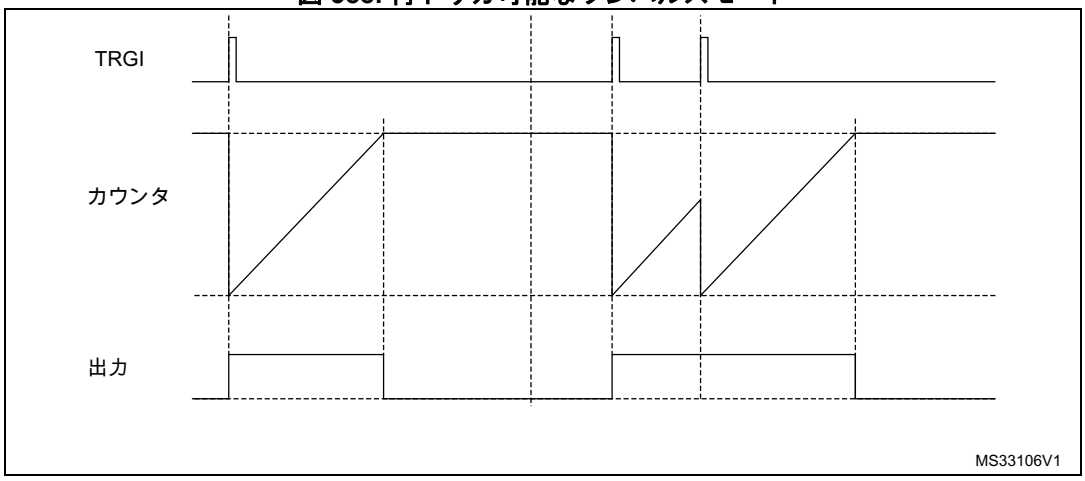
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： ***OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 388. 再トリガ可能なワンパルスモード



38.3.22 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、直交エンコーダとのインタフェースに使用されます。*表 311*を参照してください。カウンタのクロック供給は、TI1FP1 または TI2FP2（入力フィルタおよび極性選択の後は TI1 と TI2、フィルタされず反転されない場合は TI1FP1=TI1、フィルタされず反転されない場合は TI2FP2=TI2）での有効な遷移ごとに行われます。ただし、カウンタは有効である（TIMx_CR1レジスタの CEN ビットに“1”が書き込まれている）ことが前提です。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、繰り返しカウンタ、トリガ出力の機能は通常どおりに機能を続けます。エンコーダモードと外部クロックモード 2 は互換性がないので、同時に選択することはできません。

注： ***エンコーダモードを有効にするとき、プリスケーラをゼロに設定する必要があります。***

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 311. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

図 389 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=“01”（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S=“01”（TIMx_CCMR2 レジスタ、TI1FP2 は TI2 に配置）
- CC1P=“0”、CC1NP =“0”（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1）
- CC2P=“0”、CC2NP =“0”（TIMx_CCER レジスタ、TI1FP2 非反転、TI1FP2=TI2）
- SMS=“011”（TIMx_SMCR　レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=“1”（TIMx_CR1 レジスタ、カウンタ有効）

図 389. エンコーダインタフェースモードにおけるカウンタの動作例

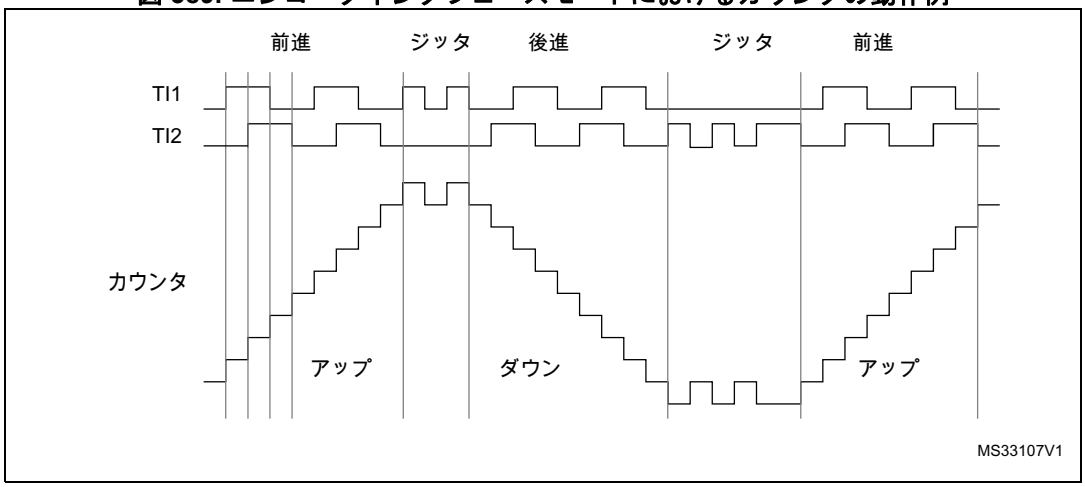
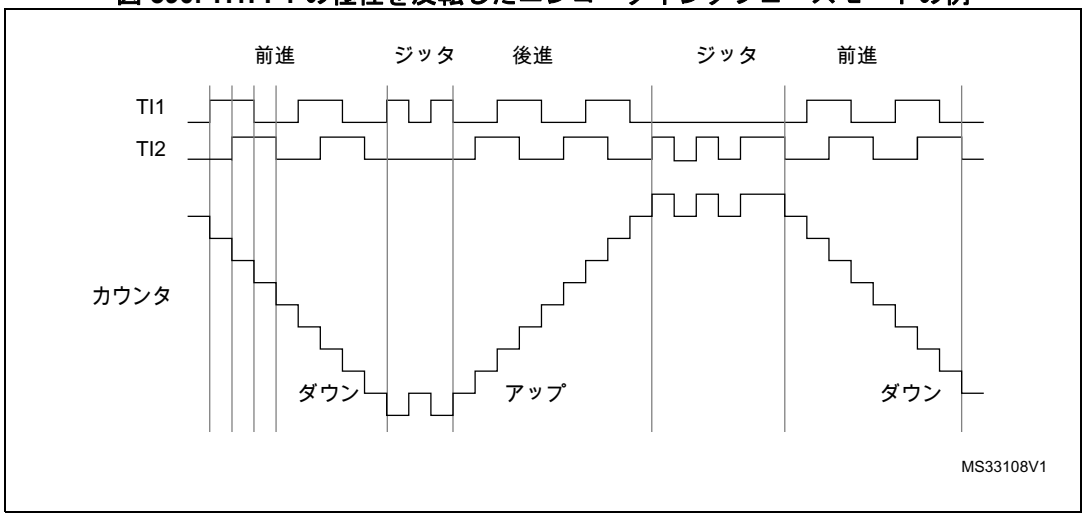


図 390 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=“1”）。

図 390. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、DMA リクエストを通じて値を読み出すことも可能です。

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読み出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書き込みモード時のみアクセス可能）。

38.3.23 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特殊なケースでは、バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

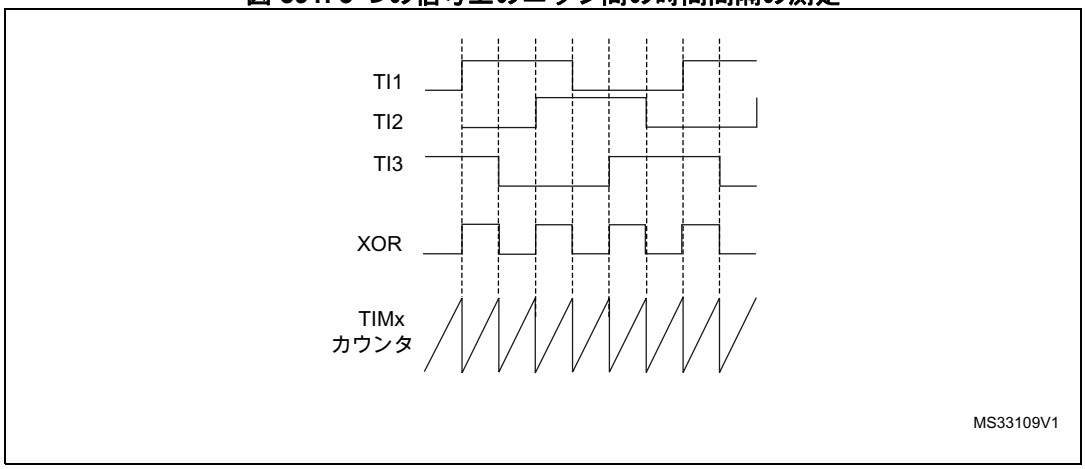
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

38.3.24 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピン TIMx_CH1、TIMx_CH2、および TIMx_CH3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の*図 391* に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 391. 3 つの信号上のエッジ間の時間間隔の測定



38.3.25 ホールセンサとのインタフェース

これは、高機能制御タイマ（TIM1 または TIM8）を使用して PWM 信号を生成し、モータと*図 392* で「インタフェースタイマ」と記されている別のタイマ TIMx（TIM2、TIM3、または TIM4）を駆動することによって実現します。「インタフェースタイマ」は、XOR を通じて TI1 入力チャネル（TIMx_CR2 レジスタの TI1S ビットをセットすることで選択できます）に接続された 3 つのタイマ入力ピン（CC1、CC2、CC3）をキャプチャします。

スレーブモードコントローラはリセットモードに設定され、スレーブ入力は TI1F_ED です。したがって、3 つの入力のいずれかが反転するごとに、カウンタは 0 からカウントをリスタートします。これが、ホール入力の変化によってトリガされるタイムベースとなります。

「インタフェースタイマ」上で、キャプチャ／比較チャネル 1 がキャプチャモードで設定され、キャプチャ信号は TRC です（*図 365：1478 ページのキャプチャ／比較チャネル（例：チャネル 1 入力ステージ）* を参照）。キャプチャされた値は、入力の 2 回の変化の間の経過時間に対応し、モータの速度情報を与えます。

「インタフェースタイマ」を出力モードで使用して、（COM イベントをトリガすることで）高機能制御タイマ（TIM1 または TIM8）のチャネルの設定を変更するパルスを生成することができます。TIM1 タイマは、モータを駆動する PWM 信号を生成するために使用されます。このためには、プログラミングした遅延の後に正パルスが生成されるように（出力比較モードまたは PWM モードで）インタフェースタイマチャネルをプログラミングする必要があります。このパルスは、TRGO 出力を通じて高機能制御タイマ（TIM1 または TIM8）に送られます。

例：TIMx タイマの 1 つに接続されているホール入力が変化するたびに、プログラミングした遅延の後に高機能制御タイマ TIM1 の PWM 設定を変更するとします。

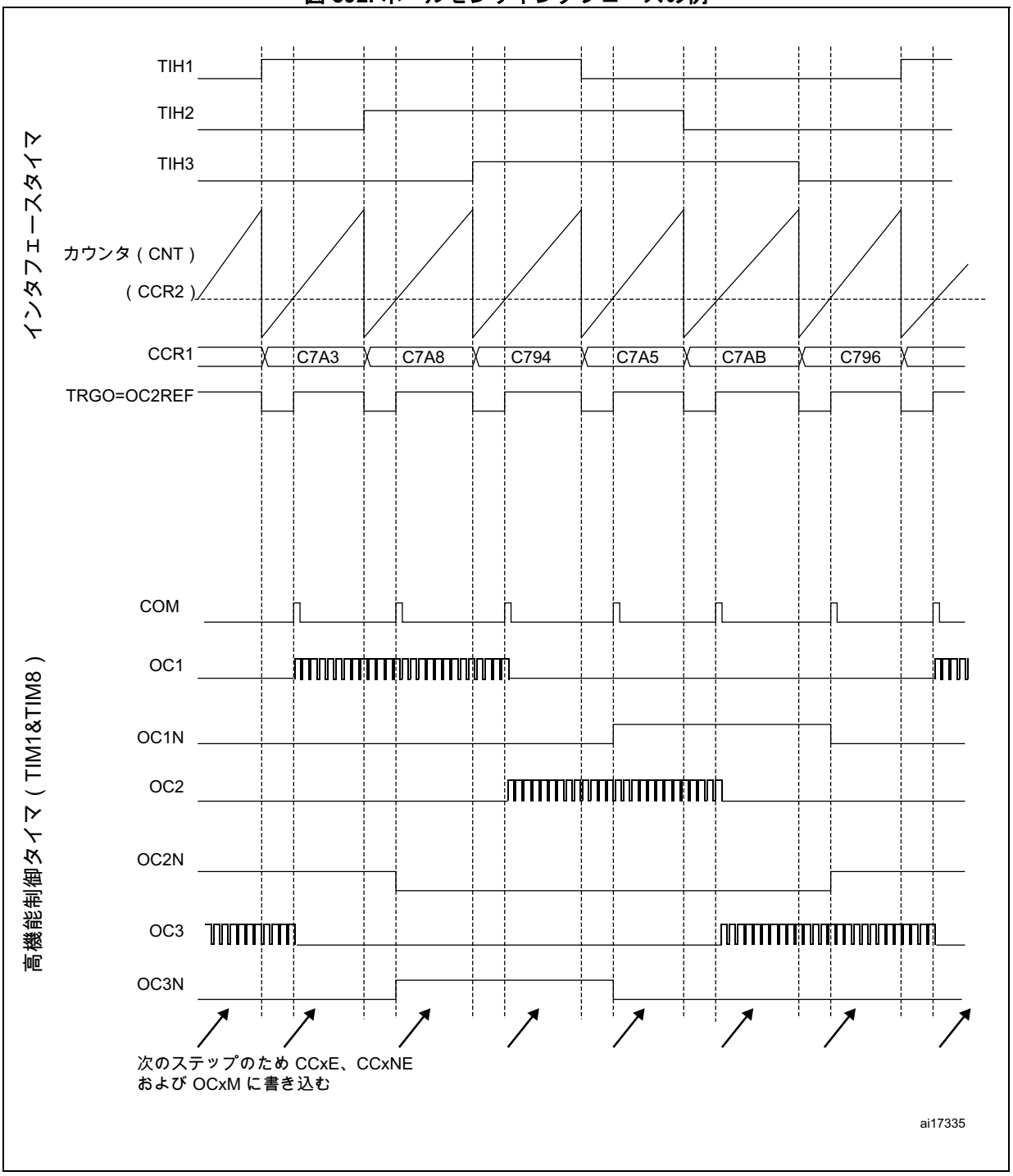
- 3 つのタイマ入力を TI1 入力チャネルに OR 接続します。このためには、TIMx_CR2 レジスタの TI1S ビットに“1”を書き込みます。
- タイムベースをプログラムします。このためには、TIMx_ARR に最大値を書き込み、TI1 の変化でカウンタがクリアされるようにします。最大カウンタ時間がセンサの 2 回の変化の間の時間より長くなるように、プリスケーラを設定します。
- チャネル 1 をキャプチャモード（TRC 選択）にプログラムします。すなわち、TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。必要な場合は、デジタルフィルタをプログラムすることもできます。
- チャネル 2 を PWM 2 モードにプログラミングし、希望の遅延を指定します。このためには、TIMx_CCMR1 レジスタの OC2M ビットに“111”を、CC2S ビットに“00”を書き込みます。
- TRGO 上のトリガ出力として OC2REF を選択します。このためには、TIMx_CR2 レジスタの MMS ビットに“101”を書き込みます。

高機能制御タイマ TIM1 で、トリガ入力として適切な ITR 入力を選択する必要があり、タイマが PWM 信号を生成するようにプログラミングし、キャプチャ /比較制御信号がプリロードされ（TIMx_CR2レジスタの CCPC=1）、COM イベントがトリガ入力によって制御されなければなりません（TIMx_CR2 レジスタの CCUS=1）。PWM 制御ビット（CCxE、OCxM）は、COM イベント後に次のステップのために書き込まれます（これは、OC2REF の立ち上がりエッジによって生成される割り込みサブルーチンで行うことができます）。

図 392 に、この例を示します。

参考資料

図 392. ホールセンサインタフェースの例



38.3.26 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。これらのタイマは、いくつかのモードで同期させることができます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

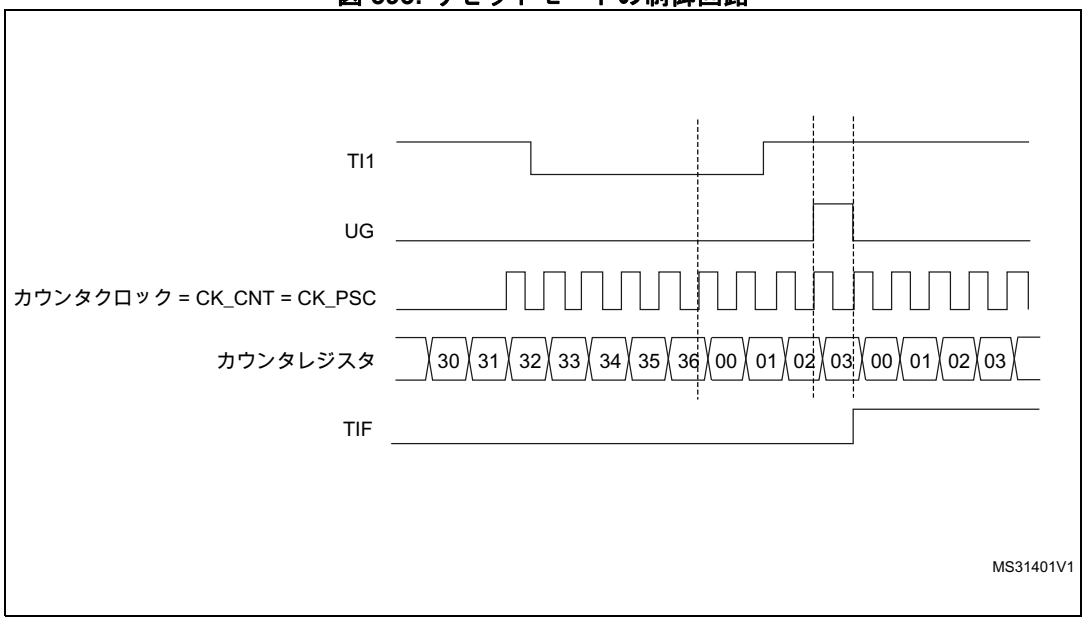
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

- TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタに CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビット）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 393. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

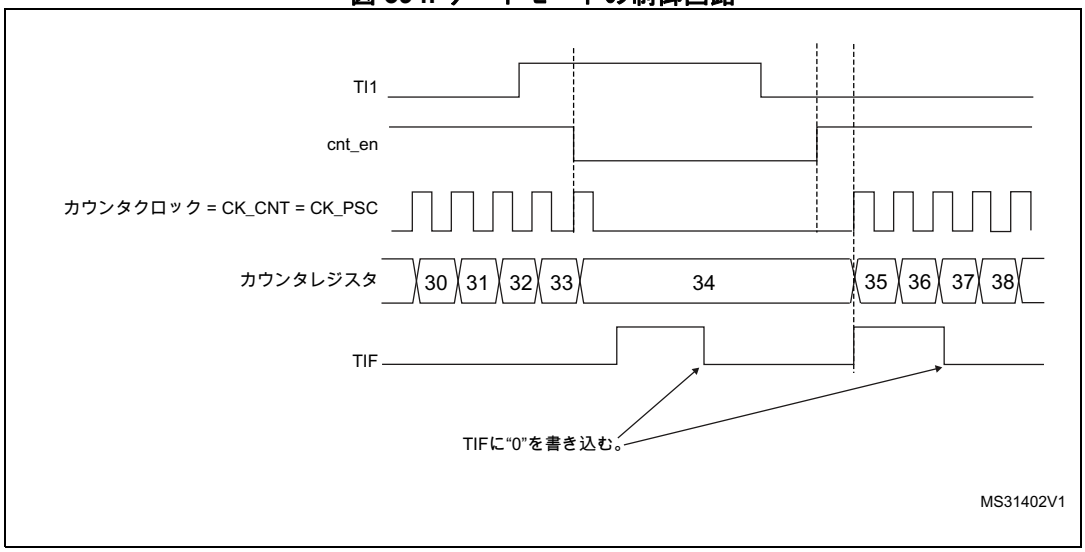
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

- TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタでCC1P=1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 394. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

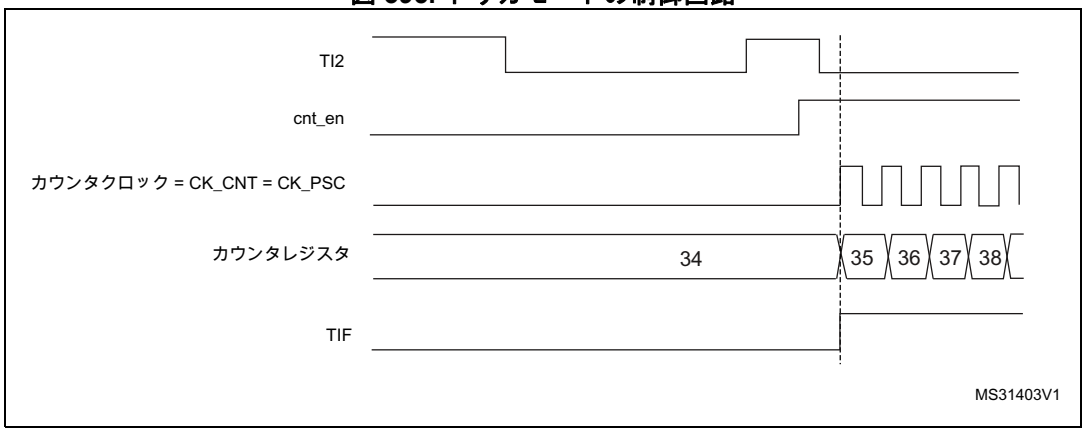
- TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。

- TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 395. トリガモードの制御回路



スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、またはトリガモード）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

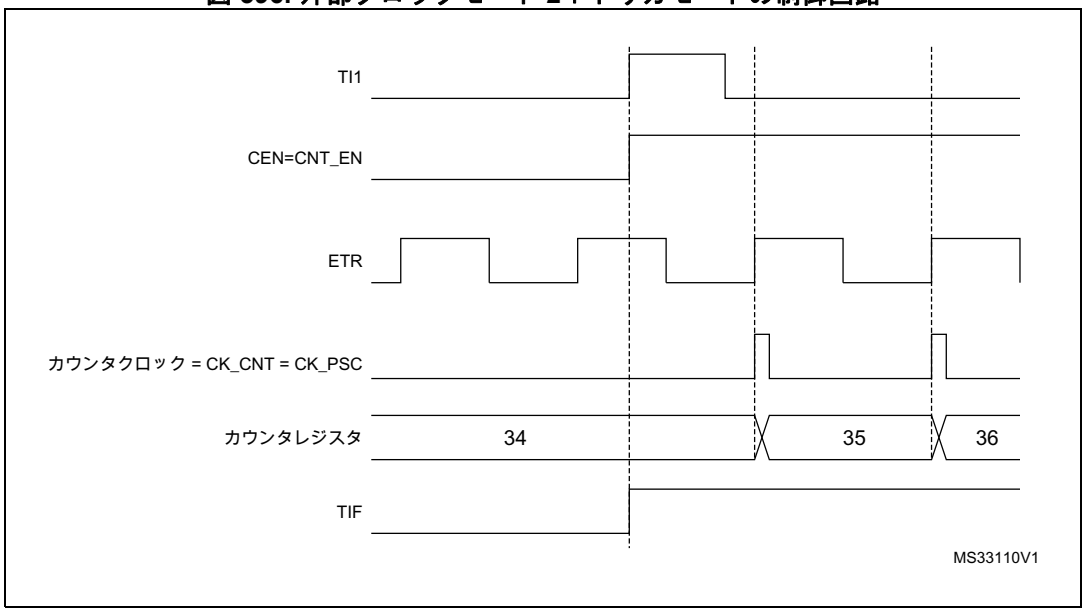
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 396. 外部クロックモード 2＋トリガモードの制御回路



注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

38.3.27 ADC の同期

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。以下のような内部エッジ検出から発行されたパルスを生成することもできます。

- OC4ref の立ち上がりおよび立ち下がりエッジ
- OC5ref の立ち上がりエッジまたは OC6ref の立ち下がりエッジ

トリガは、ADC にリダイレクトされる TRGO2 内部ラインで発行されます。使用できるイベントは全部で 16 個あり、TIMx_CR2 レジスタの MMS2[3:0] ビットを使用して選択できます。

3 相モータ駆動のアプリケーションの例については、*1490 ページの図 376* を参照してください。

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

注： ***ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

38.3.28 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

38.3.29 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 with FPU コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

安全のため、カウンタが停止すると（DBG_TIMx_STOP = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。通常強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

詳細については、*セクション 60.5.8：マイクロコントローラデバッグユニット（DBGMCU）*を参照してください。
安全のため、カウンタが停止すると（DBGMCU_APB2FZ1 の TIMx = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

38.4 TIM1/TIM8 レジスタ

レジスタの説明で使用されている略語のリストを参照してください。

38.4.1 TIM1/TIM8 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（ETR、TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。
1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読み出し専用です。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

38.4.2 TIM1/TIM8 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5
								rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **MMS2[3:0]**：マスタモード選択 2

これらのビットにより、選択される同期（TRGO2）について、ADC に情報を送信できるようになります。組み合わせは、次のとおりです。

0000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO2）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO2 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO2）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO2 には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（TRGO2）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO2）。

0100：**比較** - OC1REF 信号がトリガ出力（TRGO2）として使用されます。

0101：**比較** - OC2REF 信号がトリガ出力（TRGO2）として使用されます。

0110：**比較** - OC3REF 信号がトリガ出力（TRGO2）として使用されます。

0111：**比較** - OC4REF 信号がトリガ出力（TRGO2）として使用されます。

1000：**比較** - OC5REF 信号がトリガ出力（TRGO2）として使用されます。

1001：**比較** - OC6REF 信号がトリガ出力（TRGO2）として使用されます。

1010：**パルス比較** - OC4REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1011：**パルス比較** - OC6REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1100：**パルス比較** - OC4REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1101：**パルス比較** - OC4REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1110：**パルス比較** - OC5REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1111：**パルス比較** - OC5REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **OIS6**：出力アイドル状態 6（OC6 出力）

OIS1 ビットの説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OIS5**：出力アイドル状態 5（OC5 出力）
OIS1 ビットの説明を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OIS4**：出力アイドル状態 4（OC4 出力）
OIS1 ビットの説明を参照してください。

ビット 13 **OIS3N**：出力アイドル状態 3（OC3N 出力）
OIS1N ビットの説明を参照してください。

ビット 12 **OIS3**：出力アイドル状態 3（OC3 出力）
OIS1 ビットの説明を参照してください。

ビット 11 **OIS2N**：出力アイドル状態 2（OC2N 出力）
OIS1N ビットの説明を参照してください。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）
OIS1 ビットの説明を参照してください。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）
0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BDTR レジスタの LOCK ビット)。

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）
0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BDTR レジスタの LOCK ビット)。

ビット 7 **TI1S**：TI1 選択
0：TIMx_CH1 ピンが TI1 入力に接続されます。
1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[1:0]**：マスタモード選択
これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。
000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。
001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。
010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。
011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）。
100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。
101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。
110：比較 - OC3REF 信号がトリガ出力（TRGO）として使用されます。
111：比較 - OC4REF 信号がトリガ出力（TRGO）として使用されます。
注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込みの後、これらのビットは、遷移イベント （COM）が発生した時にのみ更新されます （CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

38.4.3 TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **TS[4:3]**：トリガ選択 - ビット 4:3

TS[2:0] 説明を参照 - ビット 6:4

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3

SMS 説明を参照 - ビット 2:0

ビット 15 **ETP**：外部トリガ極性

このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。

0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。

注： ***1:ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=00111）。***

2：外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 00111 でないことが必要）。

3：外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力は ETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 ETRP の周波数は、TIMxCLK 周波数の 1/4 までに制限されます。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。

00：プリスケーラオフ

01：ETRP 周波数は 2 分周されます。

10：ETRP 周波数は 4 分周されます。

11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N=6

0101：$f_{SAMPLING}= f_{DTS}/2$、N=8

0110：$f_{SAMPLING}= f_{DTS}/4$、N=6

0111：$f_{SAMPLING}= f_{DTS}/4$、N=8

1000：$f_{SAMPLING}= f_{DTS}/8$、N=6

1001：$f_{SAMPLING}= f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}/16$、N=5

1011：$f_{SAMPLING}= f_{DTS}/16$、N=6

1100：$f_{SAMPLING}= f_{DTS}/16$、N=8

1101：$f_{SAMPLING}= f_{DTS}/32$、N=5

1110：$f_{SAMPLING}= f_{DTS}/32$、N=6

1111：$f_{SAMPLING}= f_{DTS}/32$、N=8

ビット 7 **MSM**：マスタ/スレーブモード

0：影響なし。

1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[2:0]**：トリガ選択

このビットフィールドは TS[4:3] ビットと結合されます。

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

00111：外部トリガ入力（ETRF）

その他：予約済み

各タイマでの ITRx の詳細については、*表 312：1522 ページのTIMx 内部トリガ接続*を参照してください。

注： ***設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。***

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ/ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ/ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1000 以上のコード：予約済み。

注： ***トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。***

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

表 312. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 00000)	ITR1 (TS = 00001)	ITR2 (TS = 00010)	ITR3 (TS = 00011)
TIM1	TIM15	TIM2	TIM3	TIM4
TIM8	TIM1	TIM2	TIM4	TIM5

38.4.4 TIM1/TIM8 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル
0：COM DMA リクエストは無効です。
1：COM DMA リクエストは有効です。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル
0：ブレーク割り込みは無効です。
1：ブレーク割り込みは有効です。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 **COMIE**：COM 割り込みイネーブル
0：COM 割り込みは無効です。
1：COM 割り込みは有効です。

ビット 4 **CC4IE**：キャプチャ／比較 4 割り込みイネーブル
0：CC4 割り込みは無効です。
1：CC4 割り込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割り込みイネーブル
0：CC3 割り込みは無効です。
1：CC3 割り込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

38.4.5 TIM1/TIM8 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF
														rc_w0	rc_w0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC6IF**：比較 6 割り込みフラグ
CC1IF の説明を参照してください。（注：チャネル 6 は出力としてのみ設定できます。）

ビット 16 **CC5IF**：比較 5 割り込みフラグ
CC1IF の説明を参照してください。（注：チャネル 5 は出力としてのみ設定できます。）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SBIF**：システムブレーク割り込みフラグ
このフラグは、システムブレーク入力がアクティブになると、ハードウェアによってセットされます。システムブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。
PWM 動作をリスタートするには、このフラグをリセットする必要があります。
0：ブレークイベントは発生していません。
1：システムブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 **B2IF**：ブレーク 2 割り込みフラグ

このフラグは、ブレーク 2 入力がアクティブになると、ハードウェアによってセットされます。ブレーク 2 入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク 2 入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割り込みが保留中です。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割り込みがペンディング中です。

ビット 4 **CC4IF**：キャプチャ／比較 4 割り込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割り込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときにハードウェアによってセットされます（センターアラインモードでは、例外もあります。TIMx_CR1 レジスタの CMS ビットの説明を参照してください）。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰り返しカウンタ値に関するオーバーフローまたはアンダーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 38.4.3：TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR）*を参照）。

38.4.6 TIM1/TIM8 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **B2G**：ブレーク 2 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレーク 2 イベントが生成されます。MOE ビットがクリアされ、B2IF フラグがセットされます。有効な場合は、関連する割り込が発生します。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4 **CC4G**：キャプチャ /比較 4 イベント生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ /比較 3 イベント生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

38.4.7 TIM1/TIM8 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M[2:0]			OC2 PE	OC2 FE	CC2S[1:0]		OC1 CE	OC1M[2:0]			OC1 PE	OC1 FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3
ビット 14:12 の OC2M 説明を参照

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

Bits16 **OC1M[3]**：出力比較 1 モード - ビット 3
ビット 6:4 の OC1M 説明を参照

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル
0：OC1Ref は ETRF 入力の影響を受けません。
1：OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注： ***これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。***

注： ***PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。***

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。***

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。***

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

38.4.8 TIM1/TIM8 キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）

アドレスオフセット：0x1C

リセット値：0x0000 0000

上記の CCMR1 レジスタの説明を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4 CE	OC4M[2:0]			OC4 PE	OC4 FE	CC4S[1:0]		OC3 CE.	OC3M[2:0]			OC3 PE	OC3 FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]				IC3PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC4M[3]**：出力比較 4 モード - ビット 3

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OC3M[3]**：出力比較 3 モード - ビット 3

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 14:12 **OC4M**：出力比較 4 モード

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7 **OC3CE :** 出力比較 3 クリアイネーブル

ビット 6:4 **OC3M**：出力比較 3 モード

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7:4 **IC3F**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

38.4.9 TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E
										rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CC6P**：キャプチャ /比較 6 出力極性
CC1P の説明を参照してください。

ビット 20 **CC6E**：キャプチャ／比較 6 出力イネーブル
CC1E の説明を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC5P**：キャプチャ /比較 5 出力極性
CC1P の説明を参照してください。

ビット 16 **CC5E**：キャプチャ／比較 5 出力イネーブル
CC1E の説明を参照してください。

ビット 15 **CC4NP**：キャプチャ /比較 4 相補出力極性
CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：キャプチャ /比較 4 出力極性
CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ／比較 4 出力イネーブル
CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ /比較 3 相補出力極性
CC1NP の説明を参照してください。

ビット 10 **CC3NE**：キャプチャ /比較 3 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 9 **CC3P**：キャプチャ /比較 3 出力極性
CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ／比較 3 出力イネーブル
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ /比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 **CC2NE**：キャプチャ /比較 2 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 5 **CC2P**：キャプチャ /比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性
CC1 チャネルが出力として設定されている場合：
0：OC1N はアクティブハイです。
1：OC1N はアクティブローです。
CC1 チャネルが入力として設定されている場合：
このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされ、(TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力として設定）になった直後は書き込みできません。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、遷移イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル
0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。
1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NE アクティブビットは、遷移イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

01：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

10：予約済み。この設定は使用しないでください。

11：非反転/両エッジこの回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

注： ***このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません (TIMx_BDTR レジスタの LOCK ビット)。***

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、遷移イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。***

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1E アクティブビットは、遷移イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。***

表 313. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0、OCxN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP（BRK または BRK2 がトリガされた場合）	
			1	0		
			1	1	クロックがある場合（これは BRK がトリガされている場合にのみ有効）：アクティブな状態で OISx と OISxN が OCx と OCxN にそれぞれ対応しないとみなされる場合（ハーフブリッジ設定でスイッチを駆動した場合に短絡の原因となる）、デッドタイム後に OCx=OISx および OCxN=OISxN となります。 **注**：BRK2 は OSSI = OSSR = 1 の場合にのみ使用できます。	

1. チャネルの両方の出力が使用されないとき（GPIO が制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP はクリアされたままでなければなりません。

注： ***相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、GPIO レジスタに依存します。***

38.4.10 TIM1/TIM8 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMxCR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

38.4.11 TIM1/TIM8 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

38.4.12 TIM1/TIM8自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、*セクション 38.3.1： 1458 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

38.4.13 TIM1/TIM8 繰り返しカウンタレジスタ（TIMx_RCR）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **REP[15:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）は次のことを意味します。

エッジアラインモードでは、PWM 周期の数

センターアラインモードでは、PWM の 1/2 周期の数

38.4.14 TIM1/TIM8 キャプチャ／比較レジスタ 1 （TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：CR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読み出し専用レジスタで、プログラムできません。

38.4.15 TIM1/TIM8 キャプチャ／比較レジスタ 2 （TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読み出し専用レジスタで、プログラムできません。

38.4.16 TIM1/TIM8 キャプチャ／比較レジスタ 3 （TIMx_CCR3）

アドレスオフセット：0x3C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR3[15:0]**：キャプチャ /比較値

CC3 チャネルが出力として設定されている場合：CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。

CC3 チャネルが入力として設定されている場合：CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読み出し専用レジスタで、プログラムできません。

38.4.17 TIM1/TIM8 キャプチャ／比較レジスタ 4 （TIMx_CCR4）

アドレスオフセット：0x40

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR4[15:0]**：キャプチャ /比較値

CC4 チャネルが出力として設定されている場合：CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。

CC4 チャネルが入力として設定されている場合：CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読み出し専用レジスタで、プログラムできません。

38.4.18 TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]			
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***ビット BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] は、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初の書き込みアクセス時に、これらすべてを設定しなければならないことがあります。***

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **BK2P**：ブレーク 2 極性

0：ブレーク入力 BRK2 はアクティブローです。
1：ブレーク入力 BRK2 はアクティブハイです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 24 **BK2E**：ブレーク 2 イネーブル

このビットは、完全なブレーク 2 保護を有効にします（*図 380：ブレーク および ブレーク 2 回路の概要*のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク2 機能は無効です。
1：ブレーク2 機能は有効です。

注： ***BRKIN2 は OSSR = OSSI = 1 の場合にのみ使用してください。***

注： ***このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 23:20 **BK2F[3:0]**：ブレーク 2 フィルタ

このビットフィールドは、BRK2 入力をサンプリングする周波数と、BRK2 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK2 は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： ***このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： ***このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力の 1 つがアクティブになると、ハードウェアによって非同期にクリアされます（BRK または BRK2）。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：ブレーク 2 イベントへの対応。OC および OCN 出力は無効です。

ブレークイベントへの対応、または MOE が 0 に書き込まれた場合：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 38.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力 BRK および BRK2 のいずれもがアクティブでない場合）。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注：　このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

このビットは、完全なブレーク保護を有効にします（*図 380：ブレーク および ブレーク 2 回路の概要* のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク機能は無効です。

1：ブレーク機能は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注：　このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 38.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注：　このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE = 0 のとき、ブレークイベントまたはソフトウェアの書き込みにより、出力として設定されているチャネルで使用されます。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 38.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。
0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。
1：インアクティブのとき、OC/OCN 出力はまず強制的にインアクティブレベルにされ、次にデッドタイム後に強制的にアイドルレベルにされます。タイマは出力の制御を保持します。

注：　このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。
00：LOCK オフ - どのビットも書き込み保護されません。
01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。
10：LOCK レベル 2 ― LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。
11：LOCK レベル 3 ― LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注：　LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。
DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns（250 ns 単位）
32 µs から 63 µs（1 µs 単位）
64 µs から 126 µs（2 µs 単位）

注：　このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

38.4.19 TIM1/TIM8 DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）、つまり転送回数を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送
00010：3 回転送
...
10001：18 回転送

例：次の転送を考えます：DBL = 7 バイトかつ DBA = TIM2_CR1。

–DBL = 7 バイトおよび DBA = TIM2_CR1 が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1 アドレス）+ DBA +（DMA インデックス）、ここで DMA インデックス = DBL

この例では、（TIMx_CR1 アドレス）+ DBA に 7 バイトが追加され、データのコピー元／コピー先アドレスが与えられます。この場合、転送は、以下のアドレスから始めて、7 つのレジスタに対して行われます。（TIMx_CR1 アドレス）+ DBA

DMA データサイズの設定に応じて、いくつかのケースが想定されます。

–DMA データサイズをハーフワードで設定した場合、7 つのレジスタにそれぞれ 16 ビットのデータが転送されます。

–DMA データサイズをバイトで設定した場合も、データは 7 つのレジスタに転送されます（最初のレジスタには最初の MSB バイトが含まれ、2 番目のレジスタには最初の LSB バイトが含まれるなど、以降同様）。タイマへの転送で、DMA によって転送されるデータサイズを指定する必要もあります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し/書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1、
00001：TIMx_CR2、
00010：TIMx_SMCR
...

38.4.20 TIM1/TIM8 完全転送用の DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4。

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

38.4.21 TIM1/TIM8 キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）

アドレスオフセット：0x54

リセット値：0x0000 0000

上記の CCMR1 レジスタの説明を参照してください。チャネル 5 および 6 は出力としてのみ設定できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]
							rw								rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6 CE	OC6M[2:0]			OC6 PE	OC6FE	Res.	Res.	OC5 CE.	OC5M[2:0]			OC5PE	OC5FE	Res.	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC6M[3]**：出力比較 6 モード - ビット 3

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OC5M[3]**：出力比較 5 モード - ビット 3

ビット 15 **OC6CE**：出力比較 6 クリアイネーブル

ビット 14:12 **OC6M**：出力比較 6 モード

ビット 11 **OC6PE**：出力比較 6 プリロードイネーブル

ビット 10 **OC6FE**：出力比較 6 高速イネーブル

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **OC5CE**：出力比較 5 クリアイネーブル

ビット 6:4 **OC5M**：出力比較 5 モード

ビット 3 **OC5PE**：出力比較 5 プリロードイネーブル

ビット 2 **OC5FE**：出力比較 5 高速イネーブル

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

38.4.22 TIM1/TIM8 キャプチャ／比較レジスタ 5 （TIMx_CCR5）

アドレスオフセット：0x58

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR5[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **GC5C3**：チャネル 5 およびチャネル 3 のグループ化

チャネル 3 出力におけるひずみ：

0：OC5REF の OC3REFC への影響はありません。

1：OC3REFC は、OC3REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR2 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 30 **GC5C2**：チャネル 5 およびチャネル 2 のグループ化

チャネル 2 出力におけるひずみ：

0：OC5REF の OC2REFC への影響はありません。

1：OC2REFC は、OC2REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 29 **GC5C1**：チャネル 5 およびチャネル 1 のグループ化

チャネル 1 出力におけるひずみ：

0：OC5REF の OC1REFC5 への影響はありません。

1：OC1REFC は、OC1REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CCR5[15:0]**：キャプチャ／比較 5 値

CCR5 は、実際のキャプチャ／比較 5 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC5PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 5 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC5 出力に送信される値を含みます。

38.4.23 TIM1/TIM8 キャプチャ／比較レジスタ 6 （TIMx_CCR6）

アドレスオフセット：0x5C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR6[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR6[15:0]**：キャプチャ／比較 6 値

CCR6 は、実際のキャプチャ／比較 6 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC6PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 6 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC6 出力に送信される値を含みます。

38.4.24 TIM1 オルタネート機能オプションレジスタ 1 （TIM1_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BK CMP2P	BK CMP1P	BKINP	BKDF1 BK0E	Res.	Res.	Res.	Res.	Res.	BK CMP2E	BK CMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：ADC1 AWD1

0100：ADC1 AWD2

0101：ADC1 AWD3

0110：ADC3 AWD1

0111：ADC3 AWD2

1000：ADC3 AWD3

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 359：TIM1/TIM8 ETR 入力回路および図 380：ブレーク および ブレーク 2 回路の概要を参照してください。***

38.4.25 TIM1 オルタネート機能レジスタ 2 （TIM1_AF2）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK1E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BK2DF1BK1E**：BRK2 dfsdm1_break[1] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 380：ブレーク および ブレーク 2 回路の概要を参照してください。***

38.4.26 TIM8 オルタネート機能オプションレジスタ 1 （TIM8_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：ADC2 AWD1

0100：ADC2 AWD2

0101：ADC2 AWD3

0110：ADC3 AWD1

0111：ADC3 AWD2

1000：ADC3 AWD3

その他：予約済み

注： これらのビットは､LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[2] 入力は無効です。

1：dfsdm1_break[2] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 359：TIM1/TIM8 ETR 入力回路および図 380：ブレーク および ブレーク 2 回路の概要を参照してください。***

38.4.27 TIM8 オルタネート機能オプションレジスタ 2 （TIM8_AF2）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK3E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 8 **BK2DF1BK3E**：BRK2 dfsdm1_break[3] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[3] を有効化します。dfsdm1_break[3] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[3] 入力は無効です。

1：dfsdm1_break[3] 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 380：ブレーク および ブレーク 2 回路の概要を参照してください。***

38.4.28 TIM1 タイマ入力選択レジスタ（TIM1_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw						rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] の入力を選択します。
0000：TIM1_CH4 入力
その他：予約済み

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] の入力を選択します。
0000：TIM1_CH3 入力
その他：予約済み

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM1_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM1_CH1 入力
0001：COMP1 出力
その他：予約済み

38.4.29 TIM8 タイマ入力選択レジスタ（TIM8_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw						rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] の入力を選択します。
0000：TIM8_CH4 入力
その他：予約済み

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] の入力を選択します。
0000：TIM8_CH3 入力
その他：予約済み

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM8_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM8_CH1 入力
0001：COMP2 出力
その他：予約済み

38.4.30 TIM1 レジスタマップ

TIM1 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 314. TIM1 レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM1_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	TIM1_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	TIM1_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	TIM1_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	TIM1_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TIM1_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COM	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	TIM1_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	TIM1_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM1_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	Res.	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	TIM1_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 314. TIM1 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	TIM1_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM1_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM1_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	TIM1_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM1_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	TIM1_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	TIM1_CCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIM1_BDTR	Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM1_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM1_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	TIM1_CCMR3 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M[2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M[2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	TIM1_CCR5	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 314. TIM1 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	TIM1_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM1_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res. ETRSEL [3:0]						BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0			0	0	0	0						0	0	1
0x64	TIM1_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK1E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM1_TISEL	Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]				Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス* を参照してください。

38.4.31 TIM8 レジスタマップ

TIM8 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 315. TIM8 レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM8_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	TIM8_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	TIM8_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	TIM8_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	TIM8_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TIM8_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COM	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	TIM8_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	TIM8_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM8_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	Res.	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	TIM8_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 315. TIM8 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	**TIM8_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIM8_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	**TIM8_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	**TIMx_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIM8_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIM8_CCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIM8_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	**TIM8_BDTR**	Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**TIM8_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIM8_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	**TIM8_CCMR3** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M[2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M[2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	**TIM8_CCR5**	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 315. TIM8 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	TIM8_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM8_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res. ETRSEL [3:0]				Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0			0	0	0	0						0	0	1
	リセット値															0	0	0	0					0	0								1
0x64	TIM8_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK3E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM8_TISEL	Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]				Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2* を参照してください。

39 汎用タイマ（TIM2/TIM3/TIM4/TIM5）

39.1 TIM2/TIM3/TIM4/TIM5 の概要

この汎用タイマは、プログラマブルなプリスケーラによって駆動される 16 ビットまたは 32 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（*入力キャプチャ*）や出力波形の生成（*出力比較と PWM*）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、*セクション 39.3.19：タイマの同期*に示すように、相互に同期させることができます。

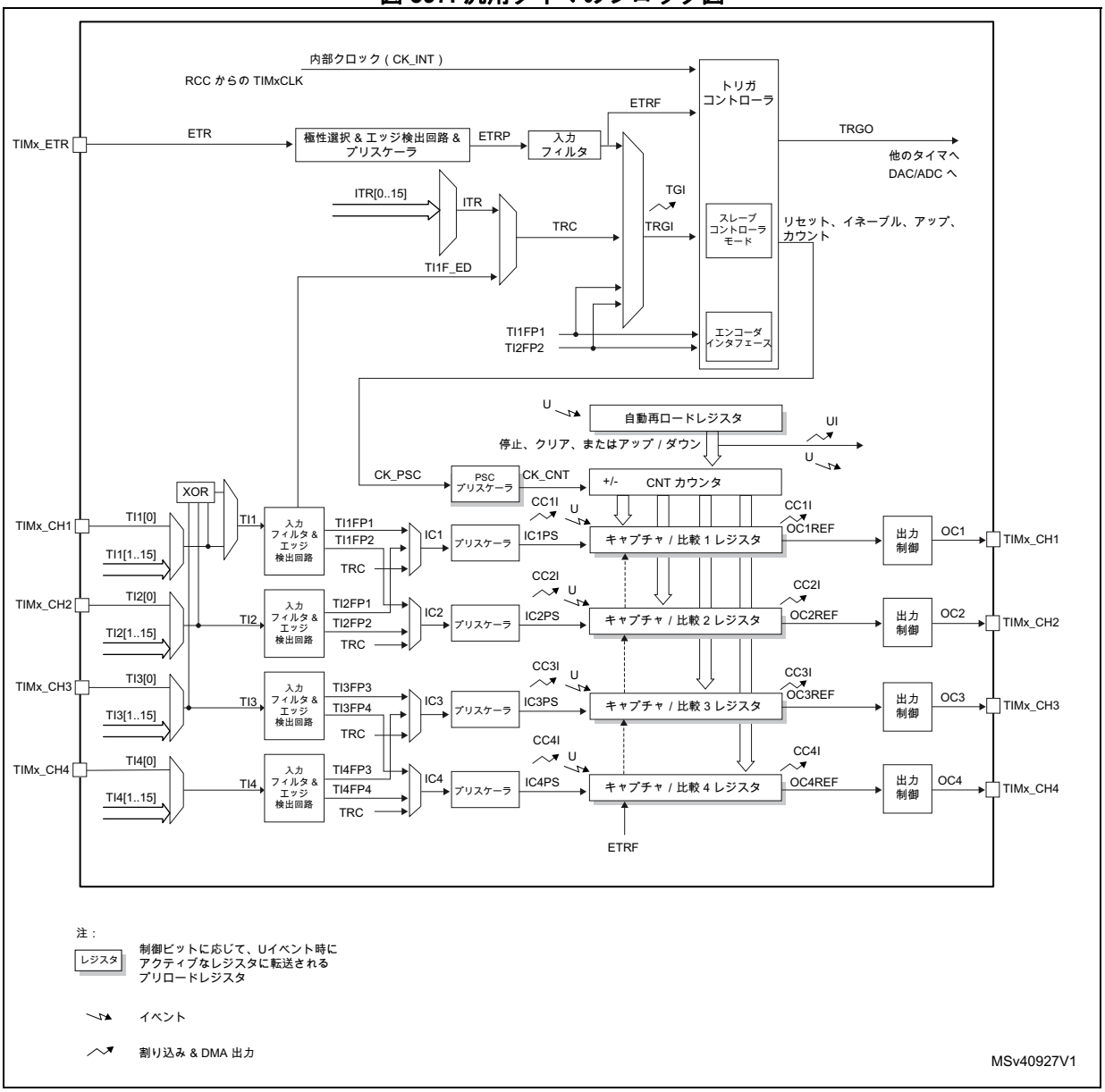
39.2 TIM2/TIM3/TIM4/TIM5 の主な特長

汎用 TIMx タイマの主な機能は、次のとおりです。

- 16 ビット（TIM3、TIM4）または 32 ビット（TIM2 および TIM5）のアップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1 から 65535 の間で分周する 16 ビットプログラム可能プリスケーラ。
- 次の機能を持つ、最大 4 つの独立チャネル。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

参考資料

図 397. 汎用タイマのブロック図



39.3 TIM2/TIM3/TIM4/TIM5機能詳細

39.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビット／32 ビットカウンタです。カウンタは、カウントアップ、カウントダウン、またはカウントアップとカウントダウンの両方を行います。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）:
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビット／32 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 398 と *図 399* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 398. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

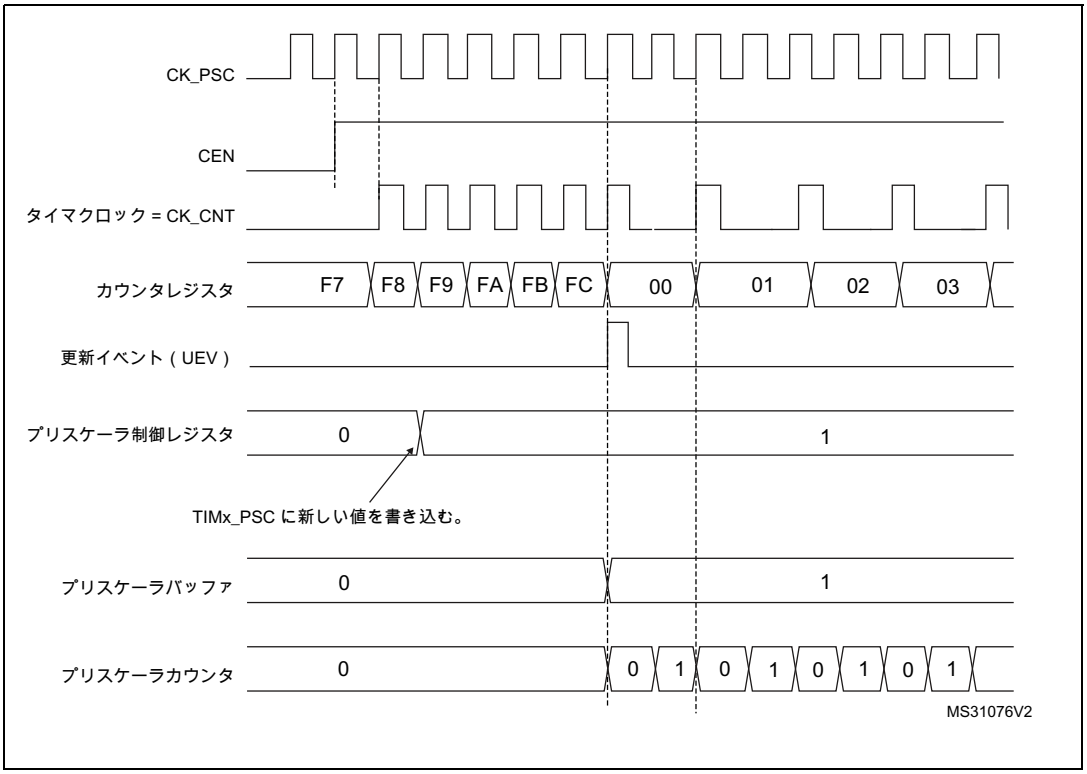
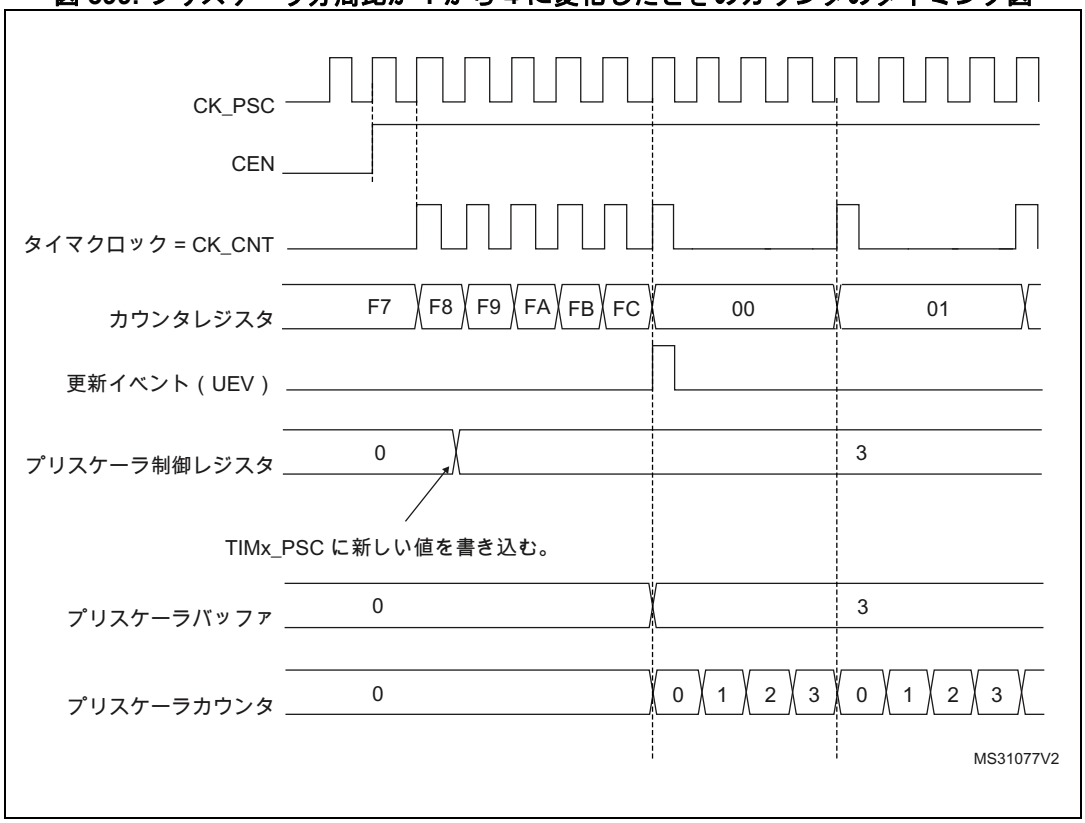


図 399. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



39.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることで生成できます。

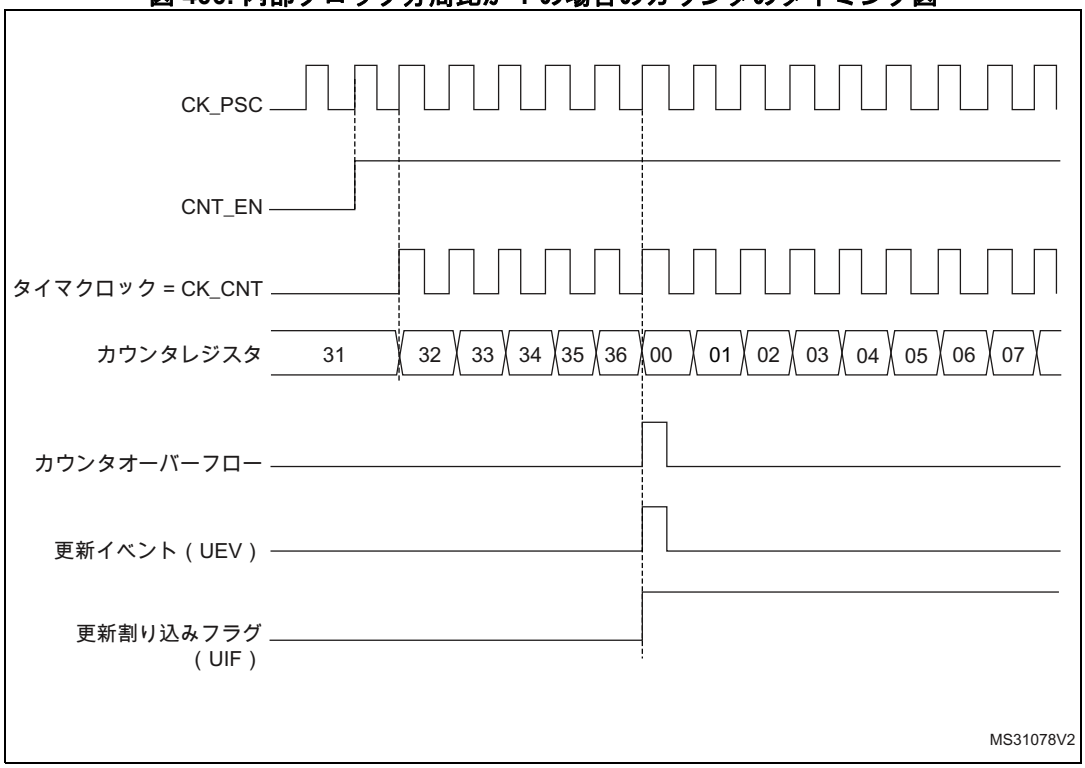
UEV イベントは、TIMx_CR1 レジスタの UDIS ビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 400. 内部クロック分周比が 1 の場合のカウンタのタイミング図



参考資料

図 401. 内部クロック分周比が 2 の場合のカウンタのタイミング図

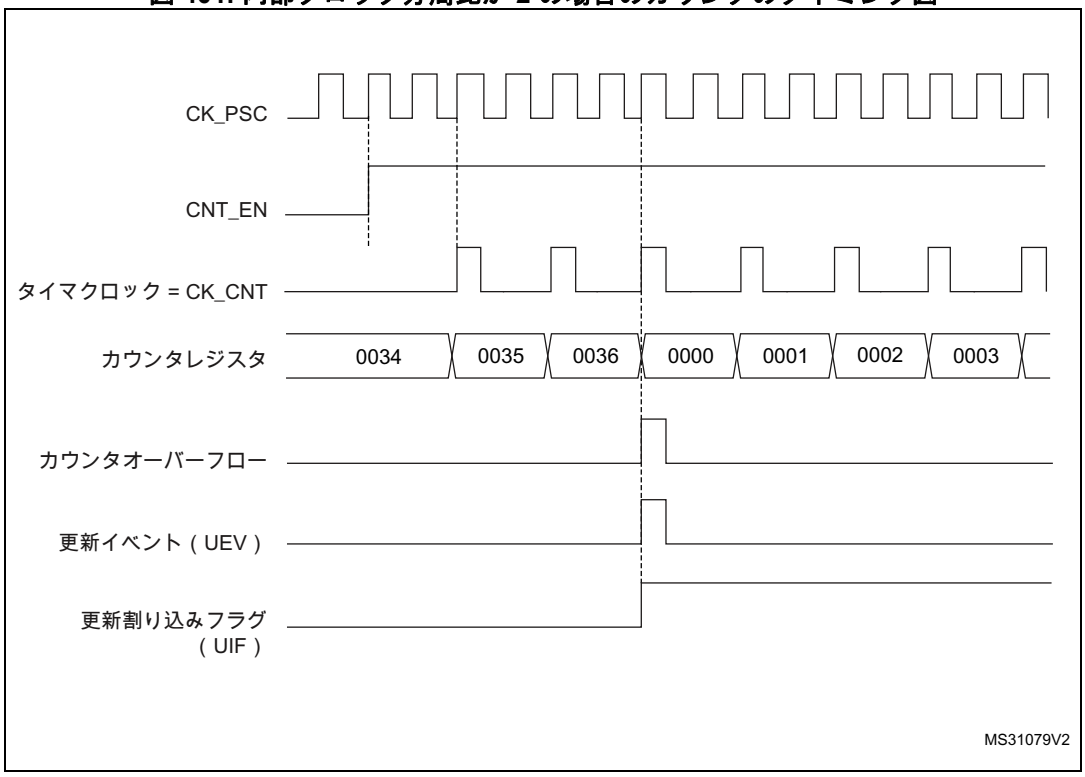


図 402. 内部クロック分周比が 4 の場合のカウンタのタイミング図

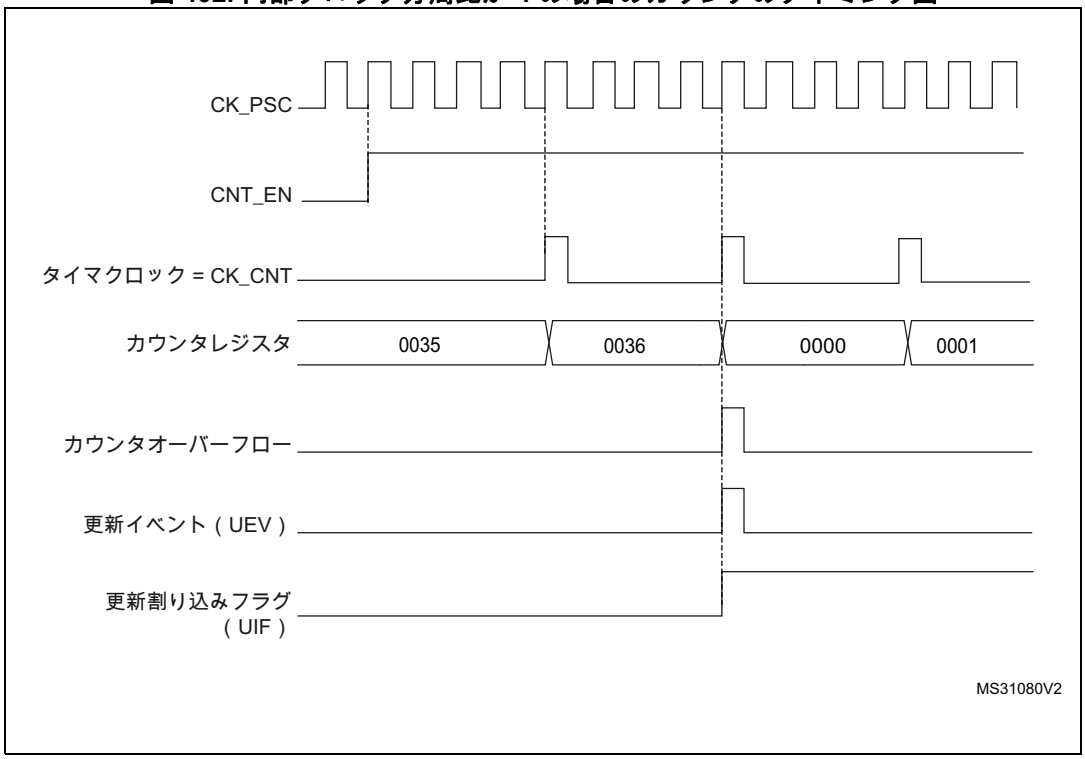


図 403. 内部クロック分周比が N の場合のカウンタのタイミング図

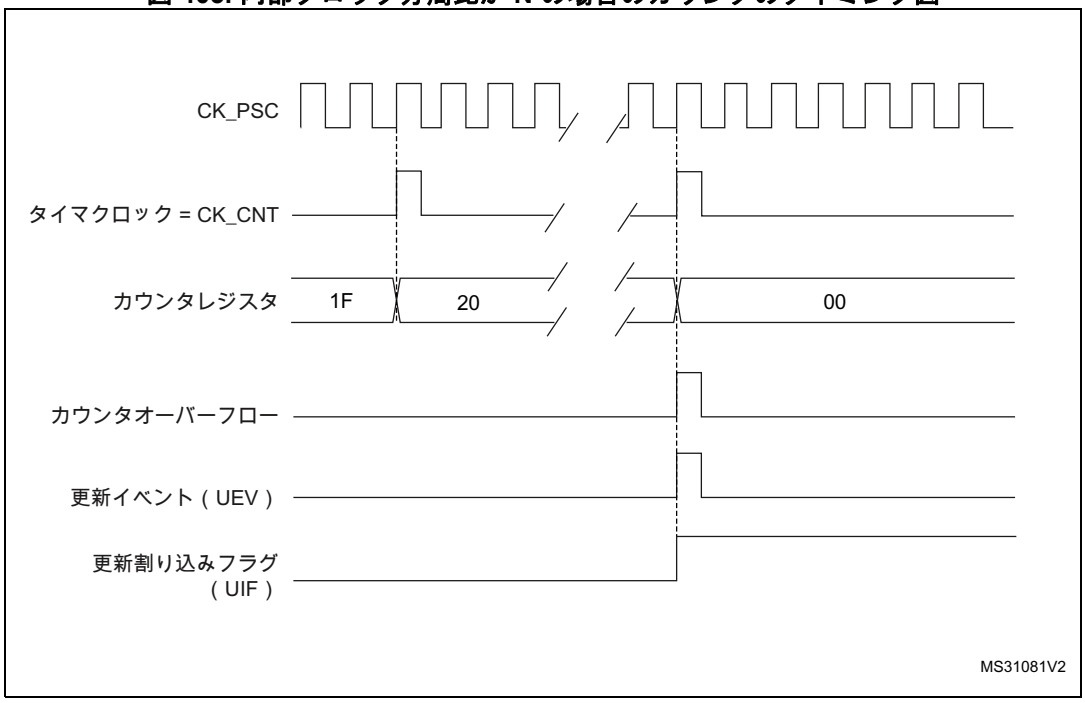


図 404. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

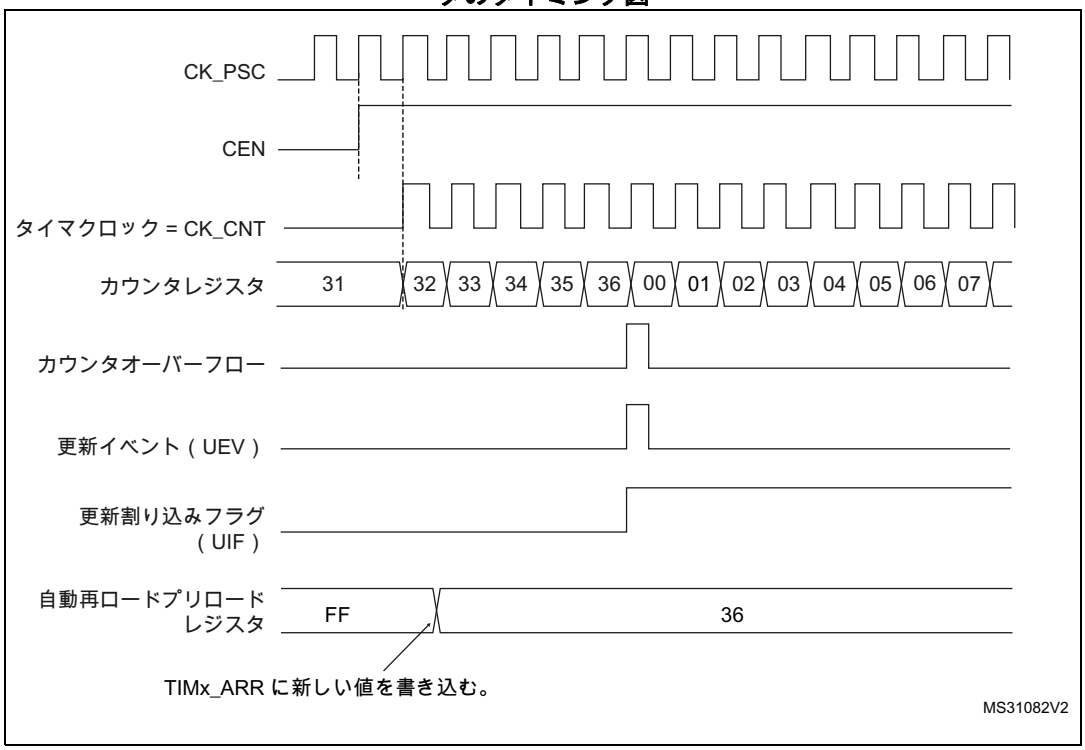
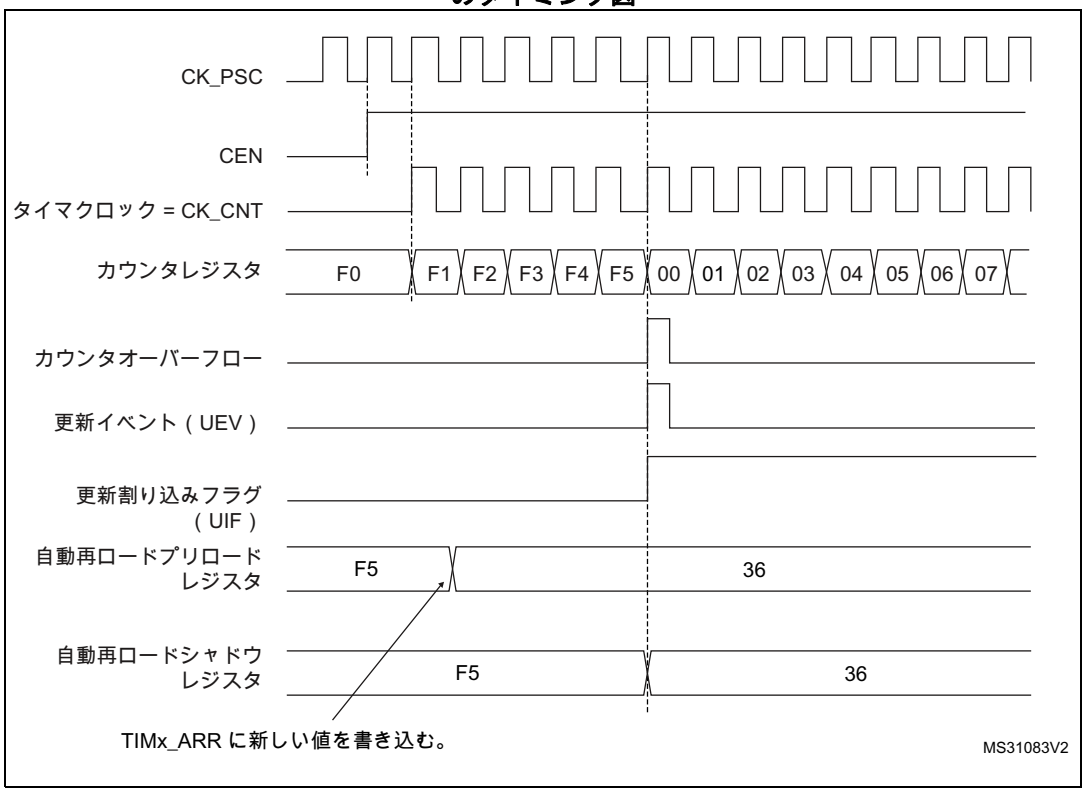


図 405. ARPE=1（TIMx_ARR はプリロードされる）の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 406. 内部クロック分周比が 1 の場合のカウンタのタイミング図

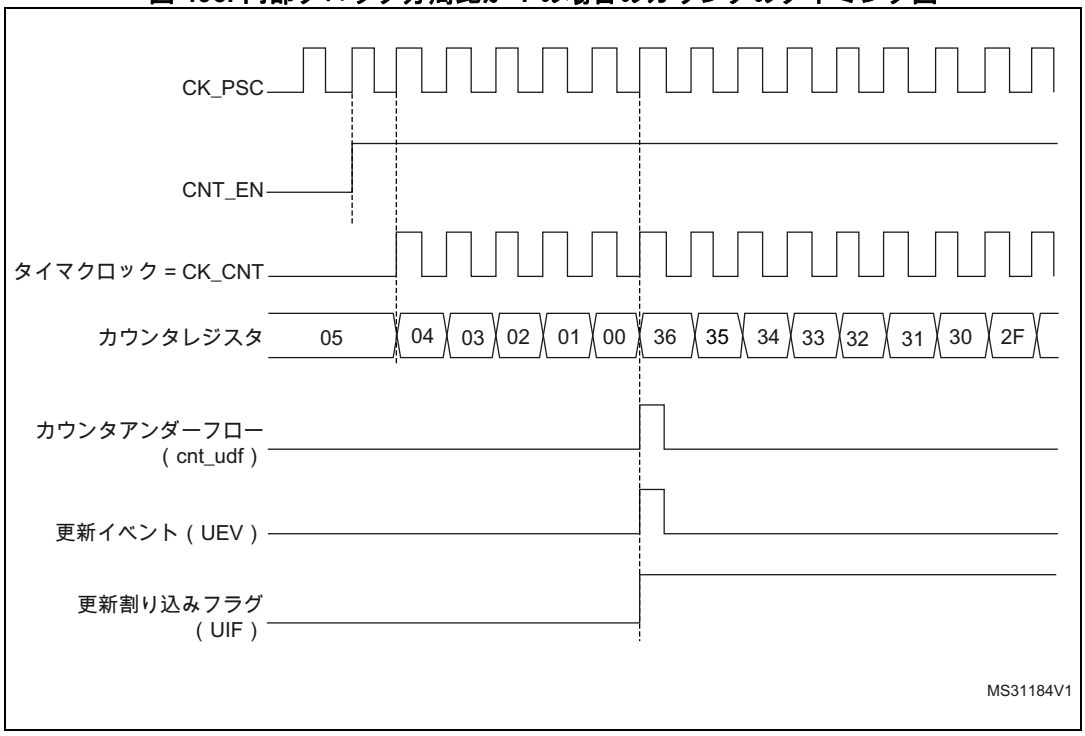


図 407. 内部クロック分周比が 2 の場合のカウンタのタイミング図

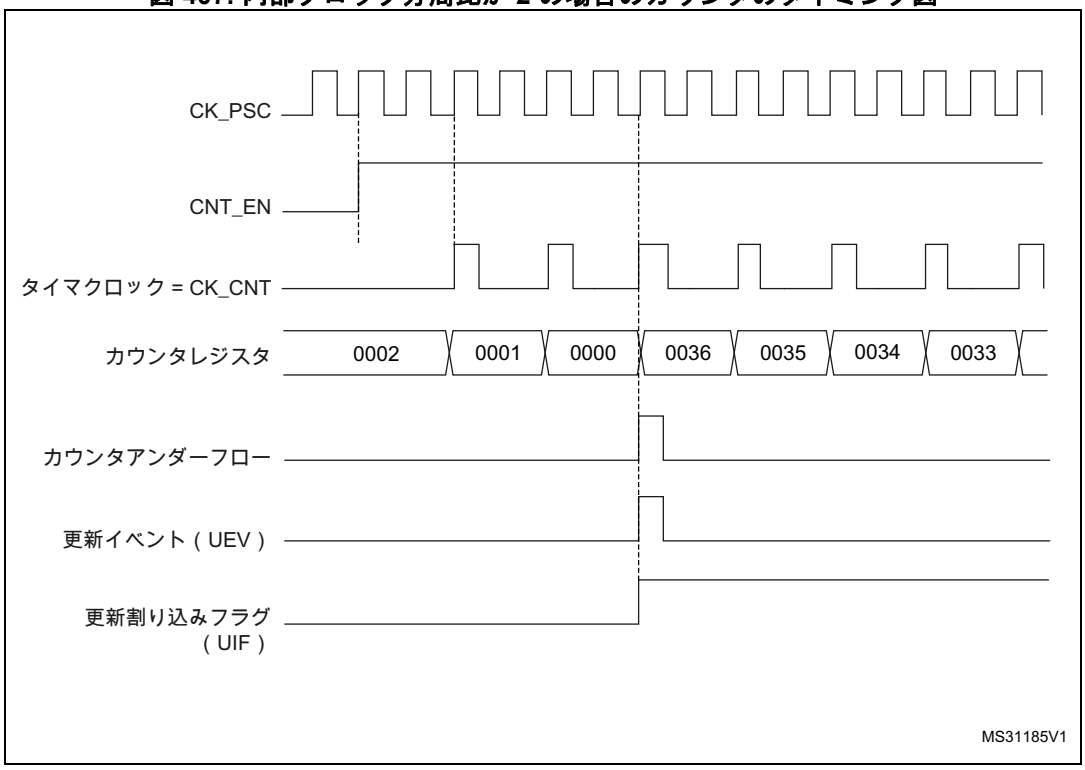


図 408. 内部クロック分周比が 4 の場合のカウンタのタイミング図

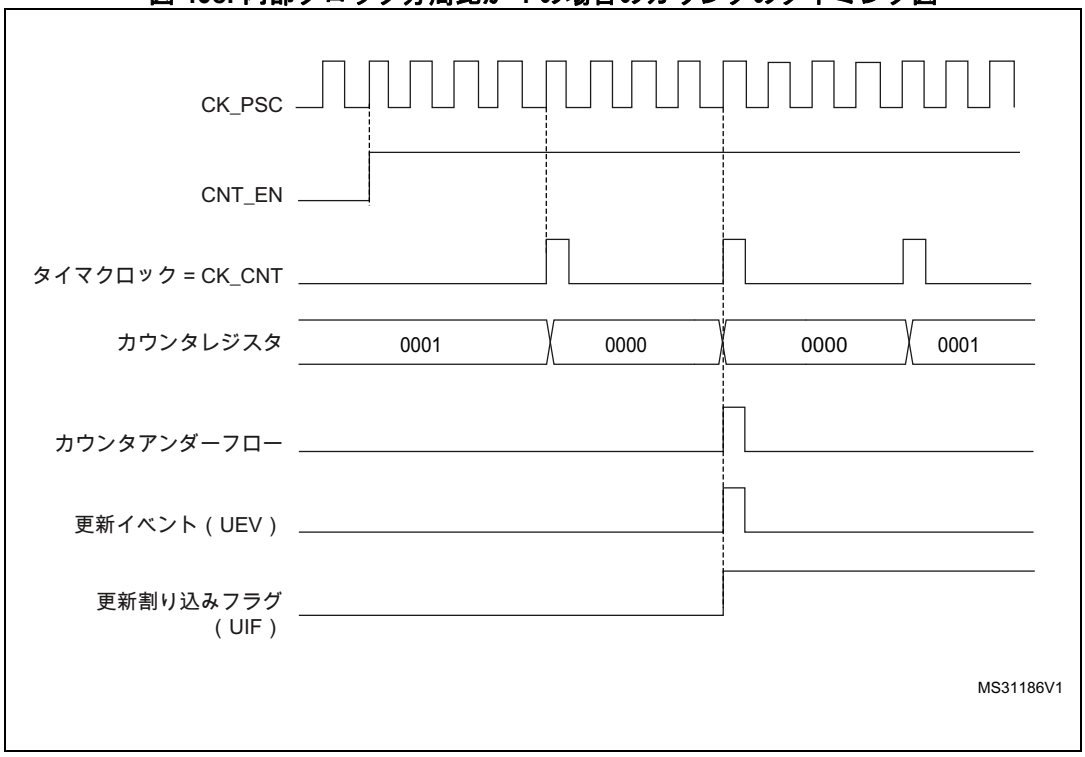


図 409. 内部クロック分周比が N の場合のカウンタのタイミング図

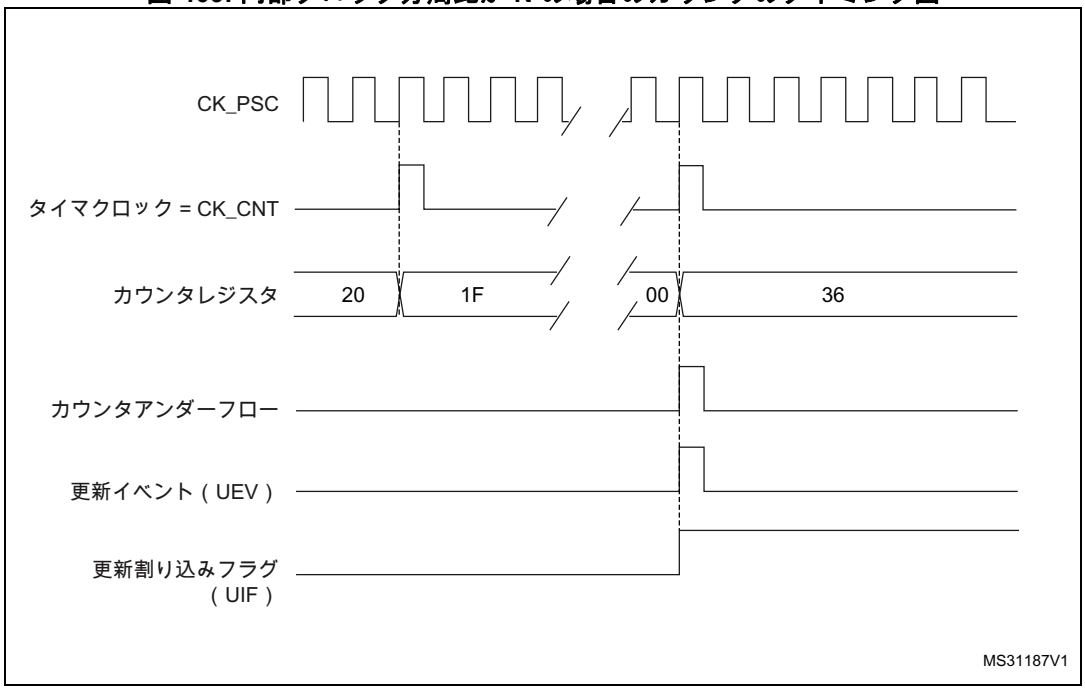
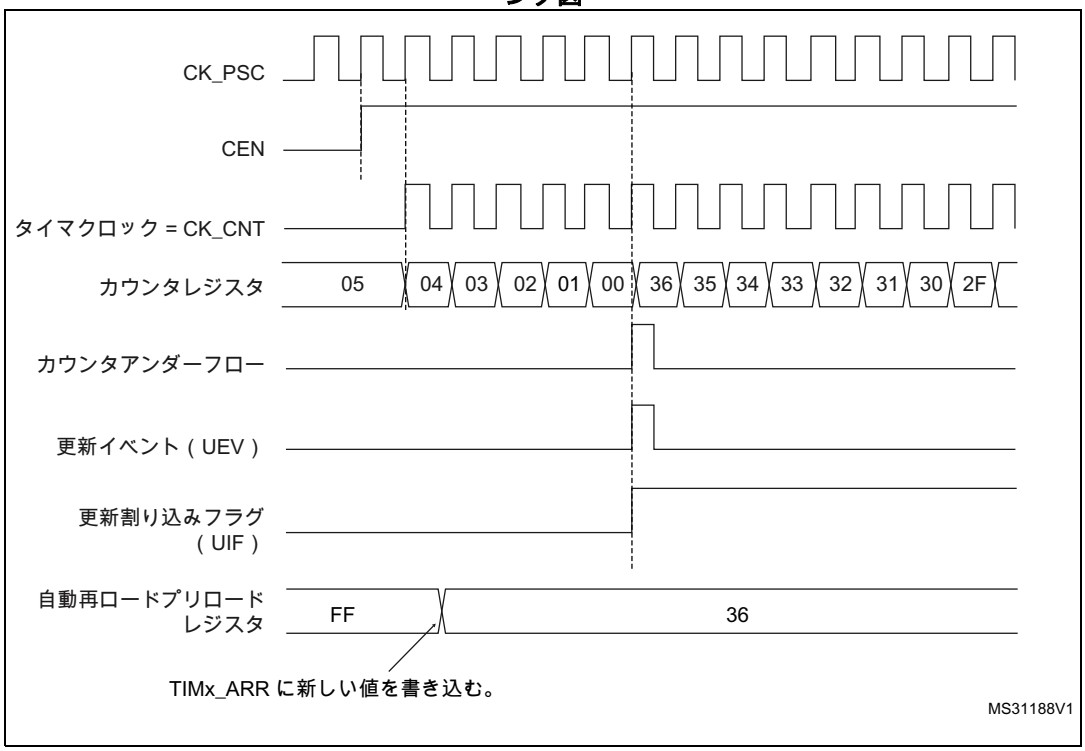


図 410. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割り込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、方向ビット（TIMx_CR1 レジスタの DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

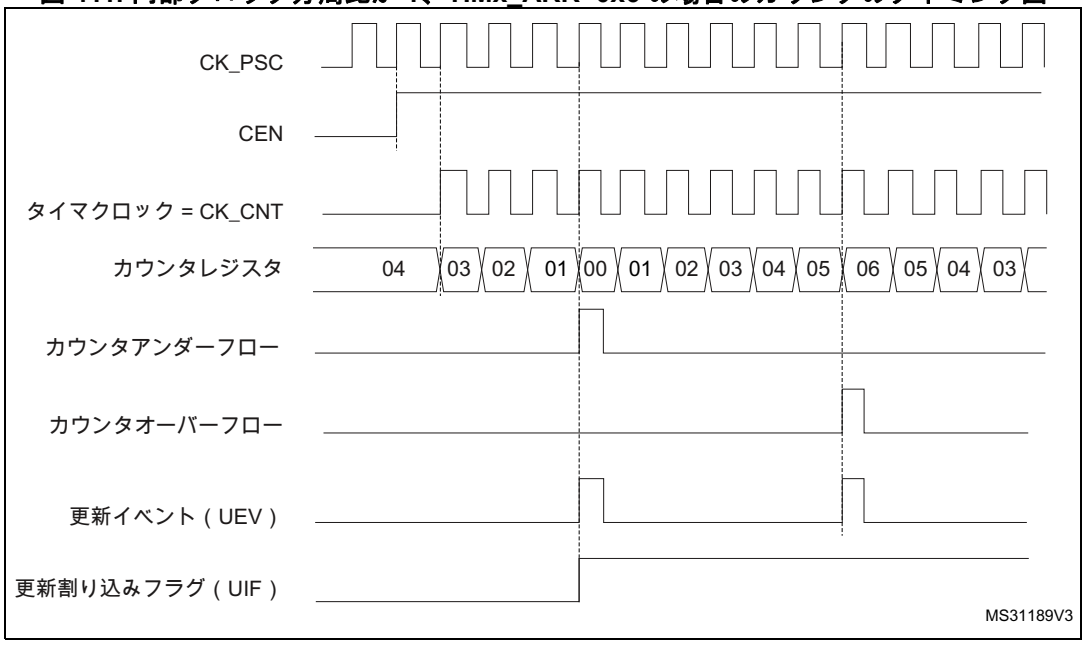
さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 411. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、*セクション 39.4.1： 1608 ページのTIMx 制御レジスタ 1（TIMx_CR1）*を参照）。

図 412. 内部クロック分周比が 2 の場合のカウンタのタイミング図

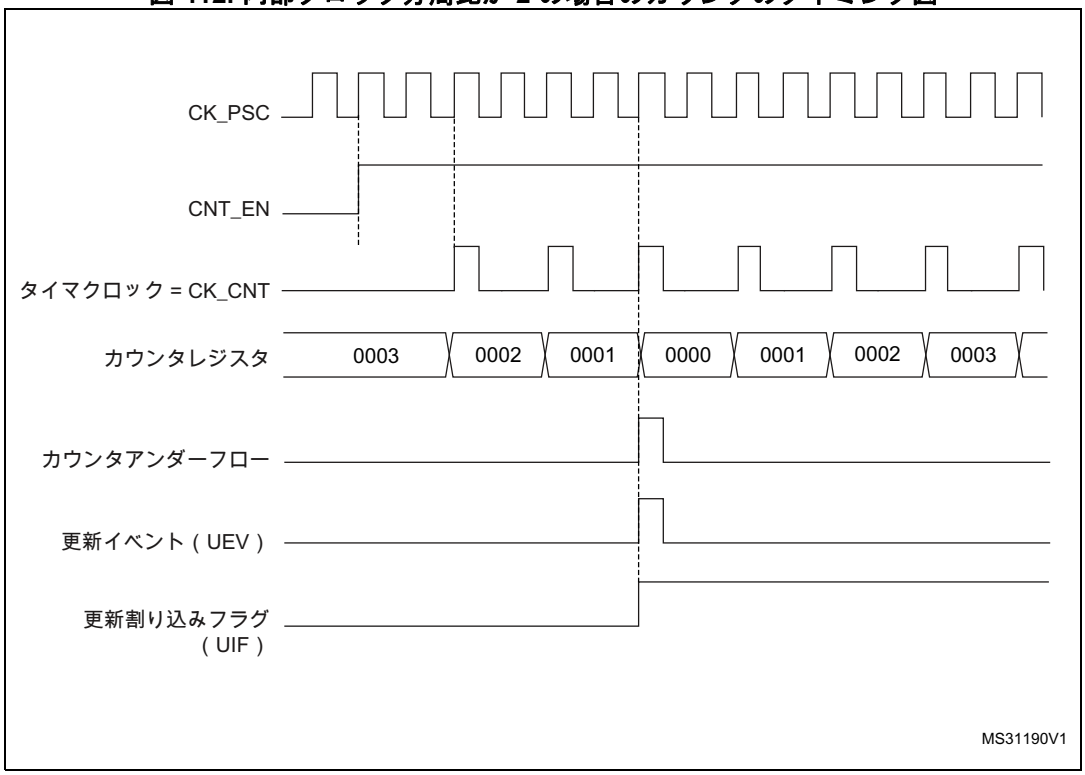
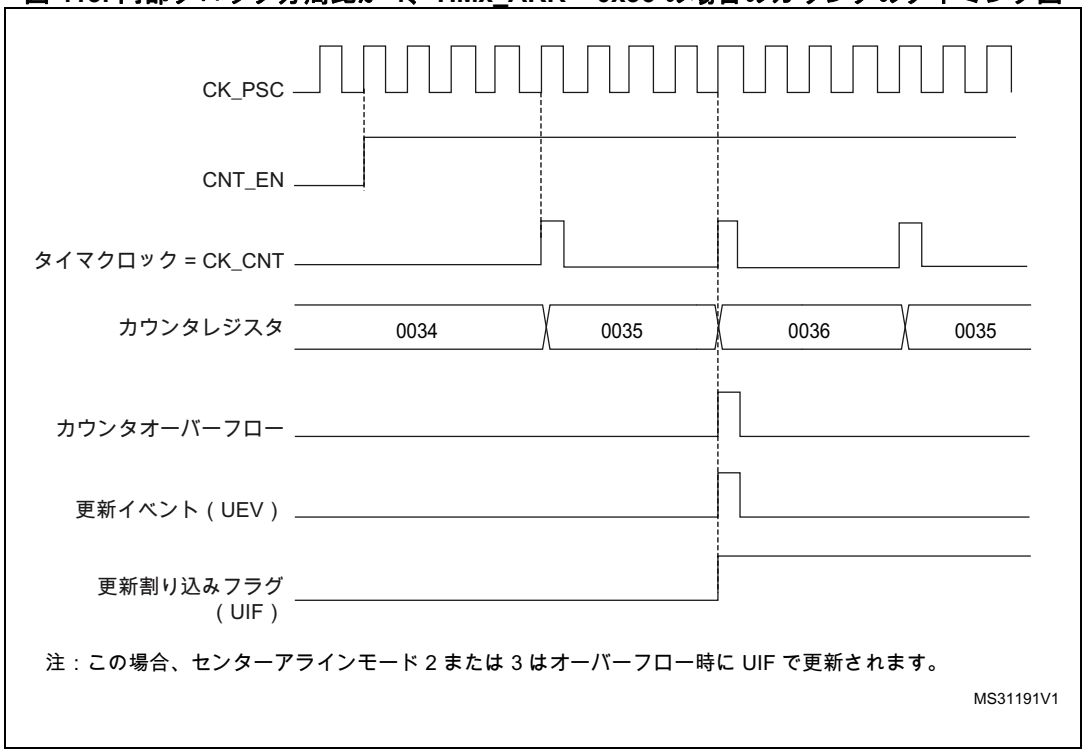


図 413. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 414. 内部クロック分周比が N の場合のカウンタのタイミング図

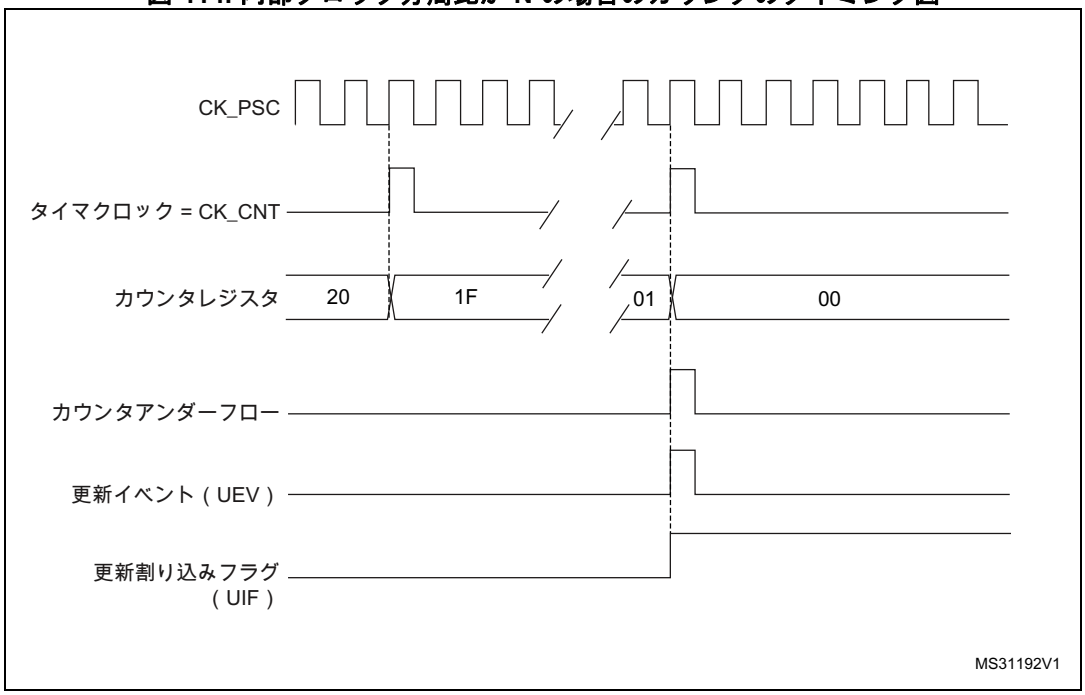


図 415. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図

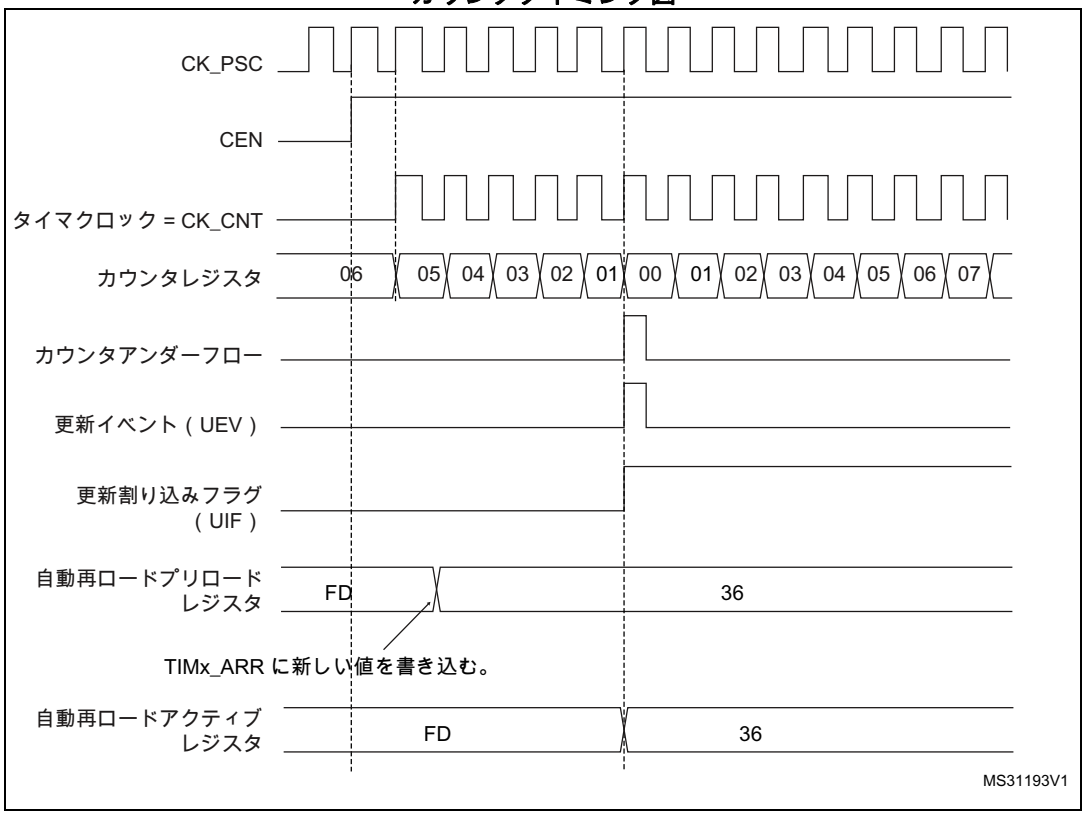
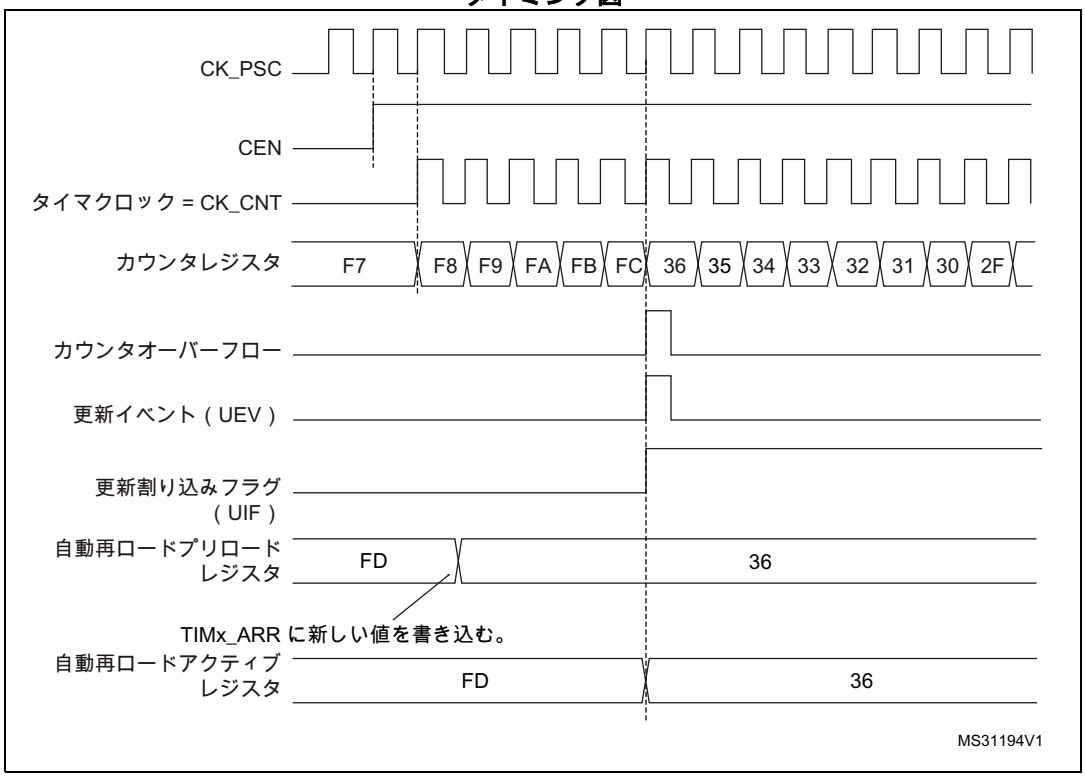


図 416. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



39.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

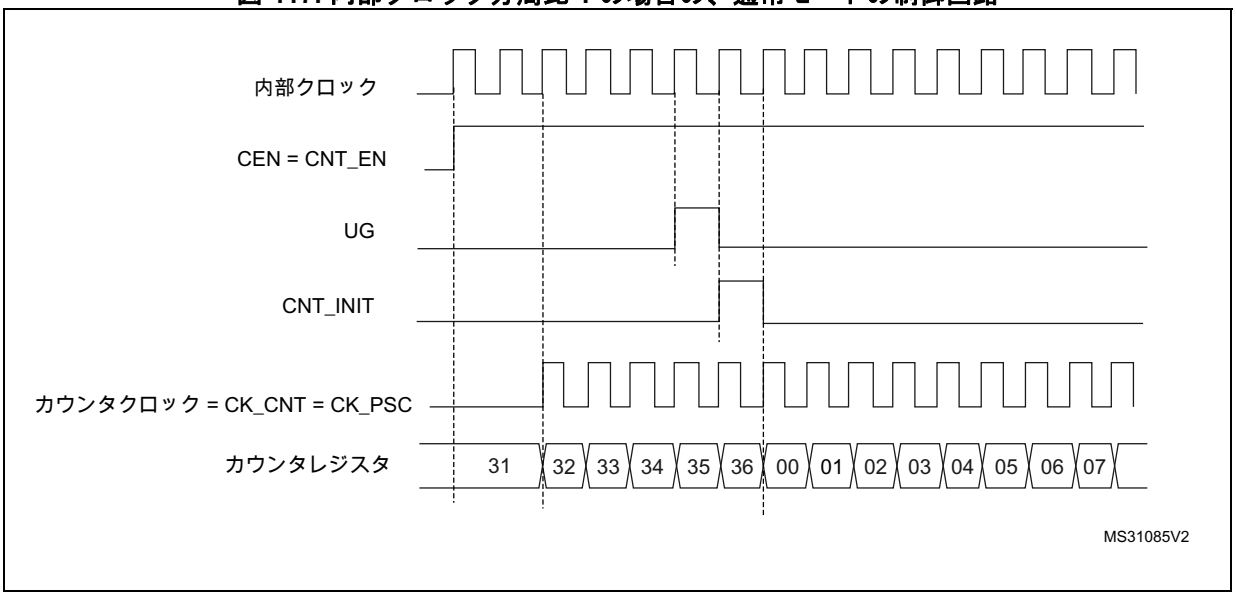
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン（TIx）
- 外部クロックモード 2：外部トリガ入力（ETR）
- 内部トリガ入力（ITRx）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、タイマ 13 がタイマ 2 のプリスケーラとして機能するように設定できます。詳細については、*：1602 ページのタイマを別のタイマのプリスケーラとして使用する* を参照してください。

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（TIMx_SMCR レジスタの SMS=000）、CEN、DIR ビット（TIMx_CR1 レジスタ）と UG ビット（TIMx_EGR レジスタ）が実際の制御ビットであり、ソフトウェアでのみ変更できます（自動的にクリアされたままの UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 417 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

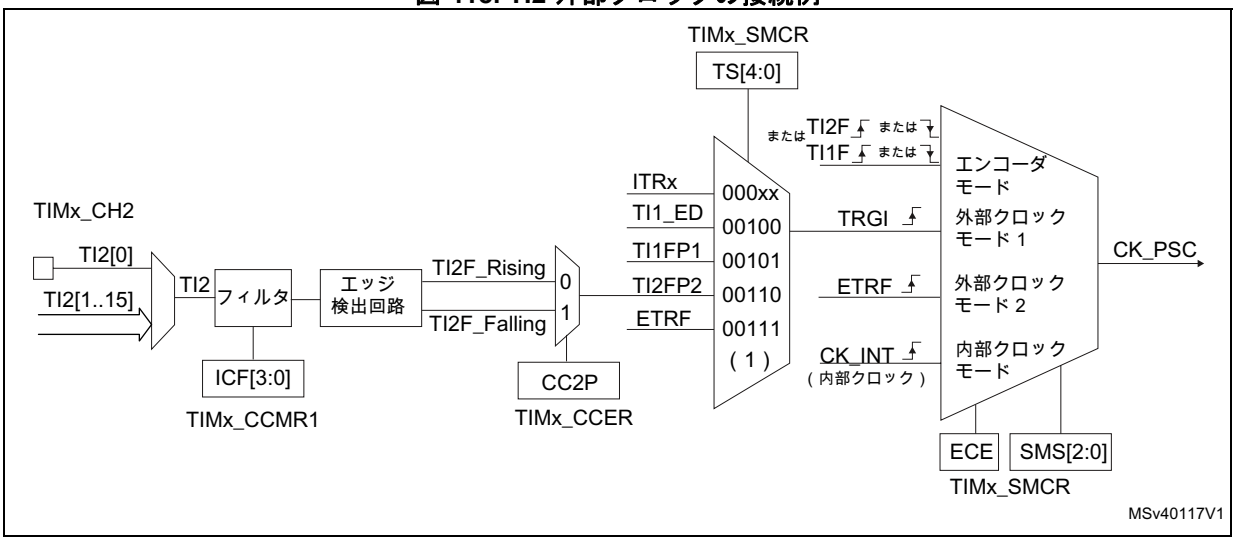
図 417. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 418. TI2 外部クロックの接続例



1. 01000 から 11111 の範囲のコード：ITRy

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。

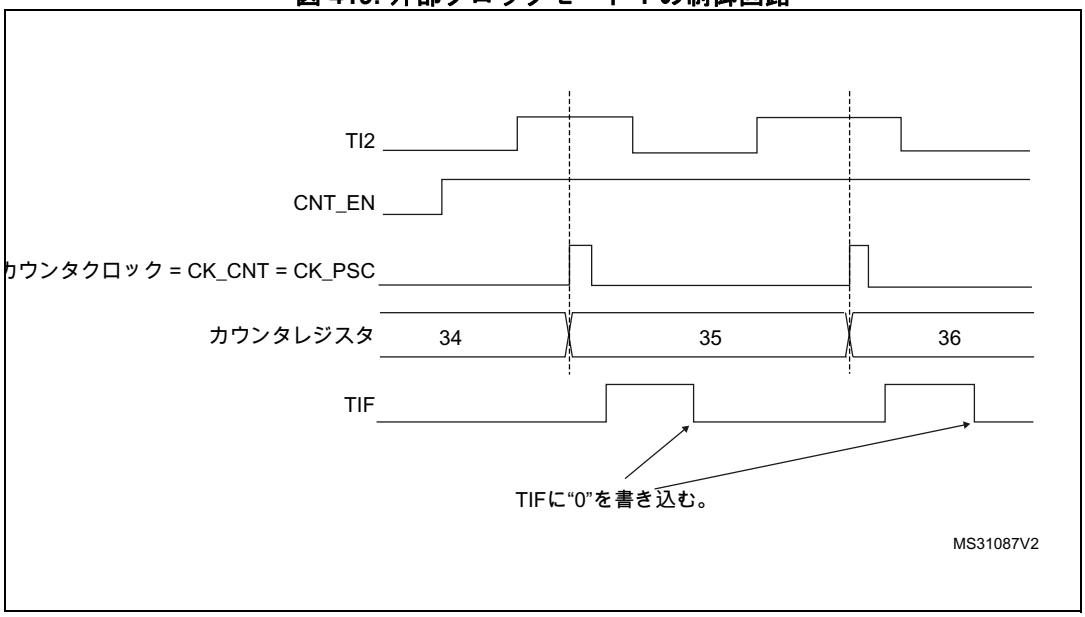
注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

4. CC2P=0、CC2NP=0、および CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 419. 外部クロックモード 1 の制御回路



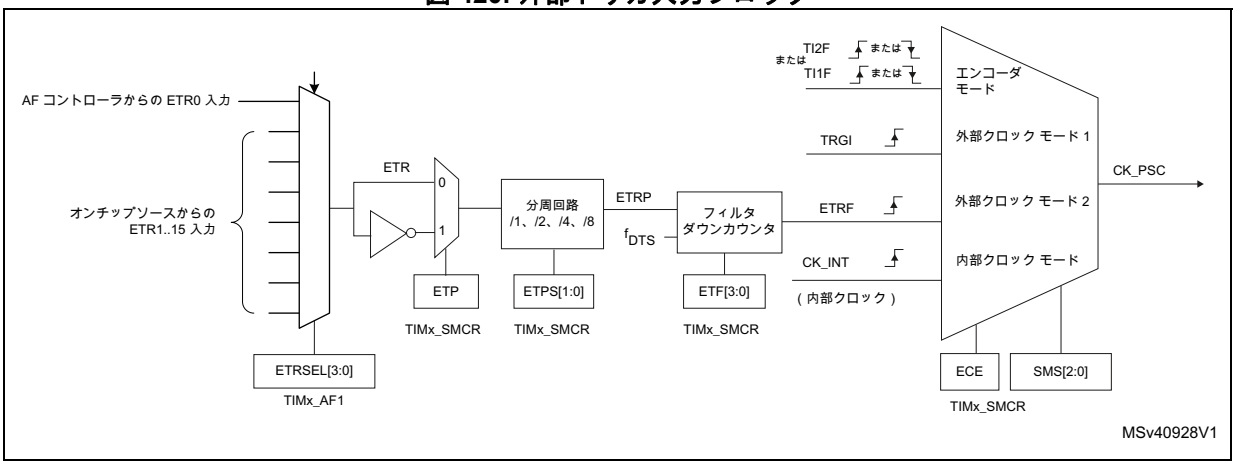
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 420 に、外部トリガ入力ブロックの概要を示します。

図 420. 外部トリガ入力ブロック



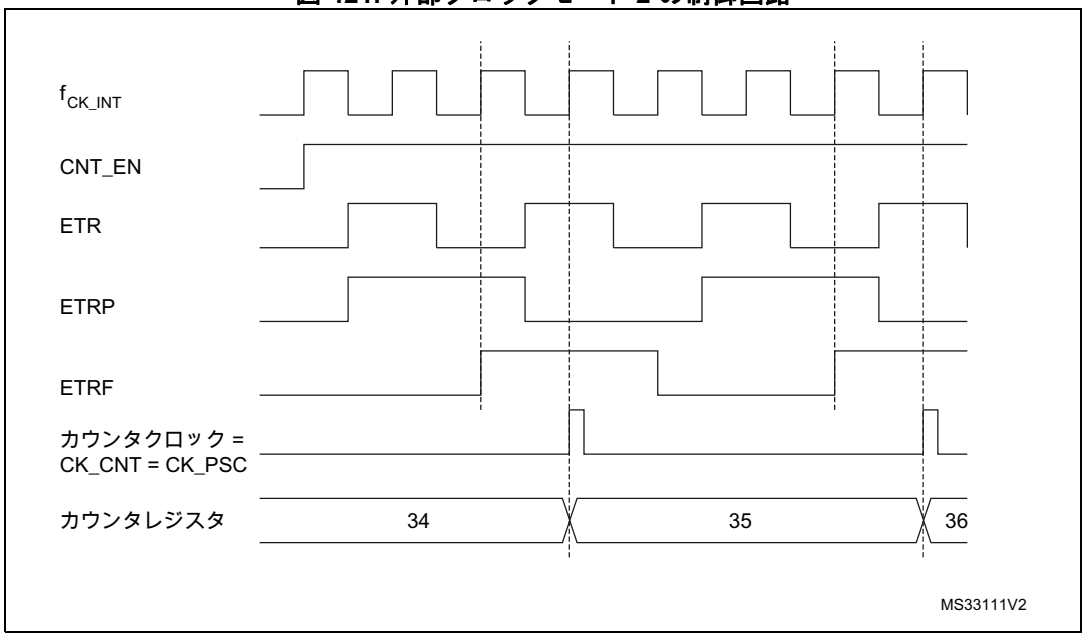
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. TIMx_AF1 レジスタの ETRSEL[3:0] ビットで、適切な ETR ソース（内部または外部）を選択します。
2. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
3. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
4. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
5. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 421. 外部クロックモード 2 の制御回路



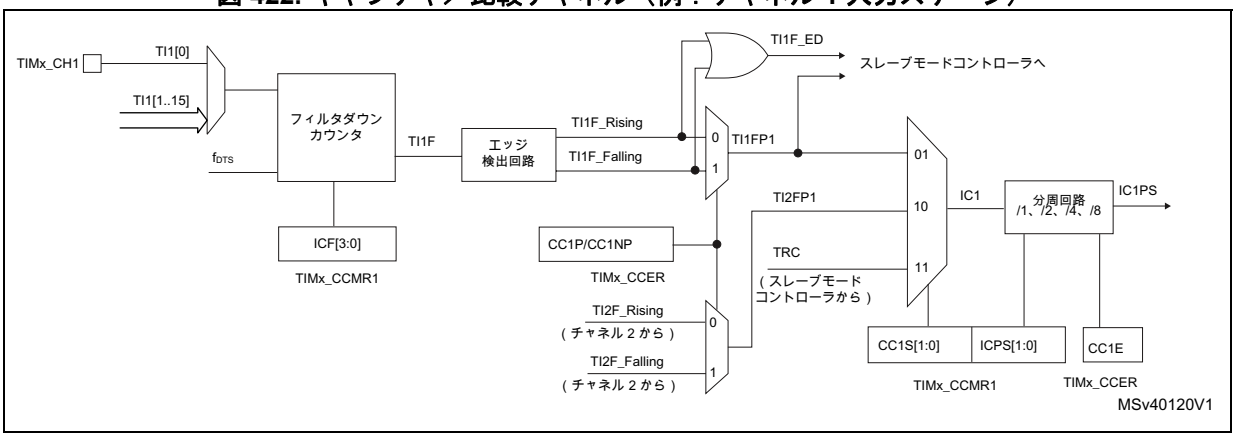
39.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 422. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 423. キャプチャ／比較チャネル 1 メイン回路

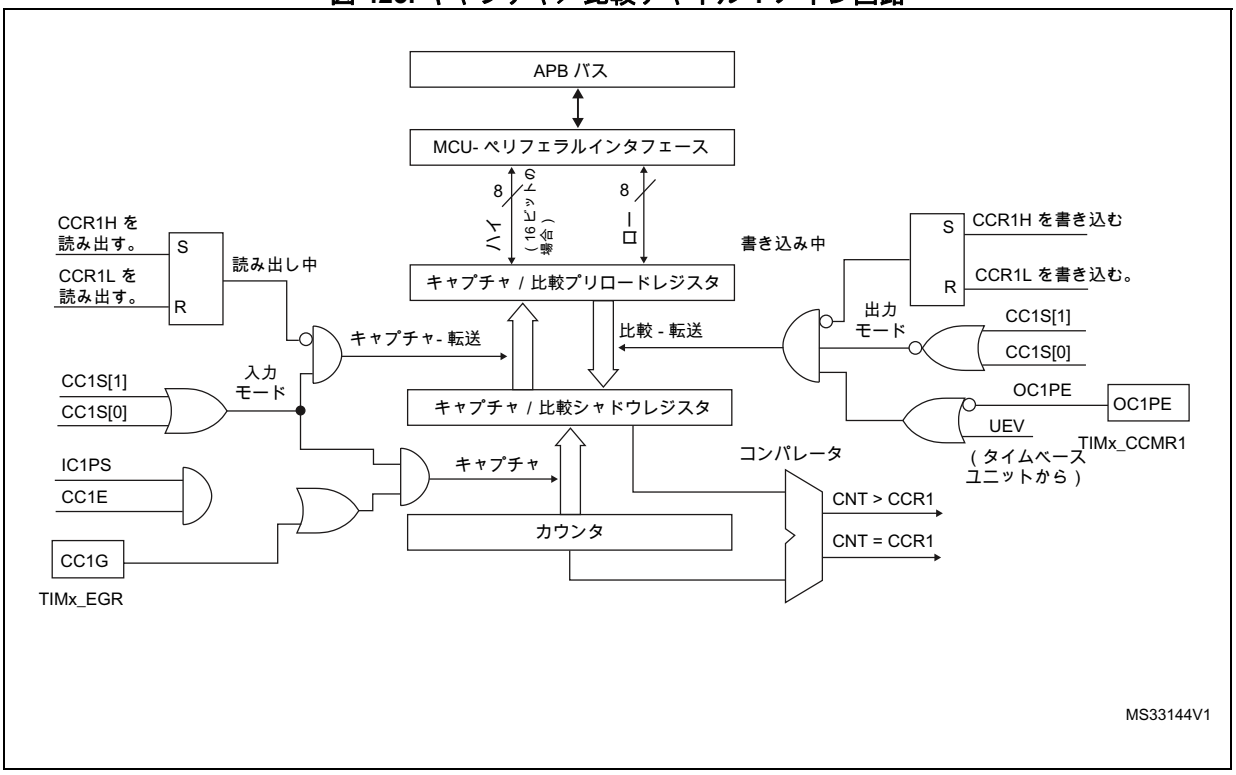
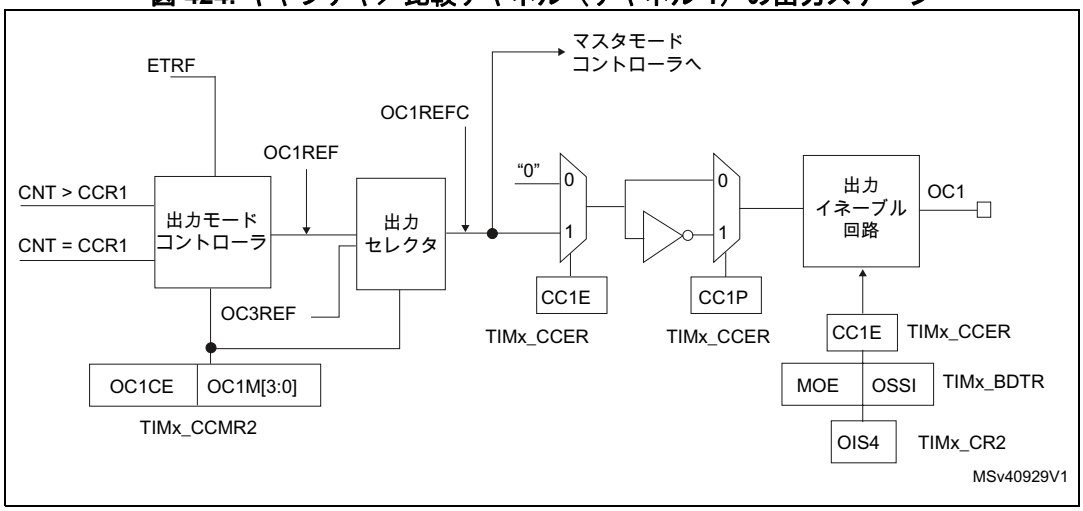


図 424. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

39.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
3. タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P、CC1NP、および CC1NP ビットに“000”を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに 00 を書き込みます）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

39.3.6 PWM 入力モード

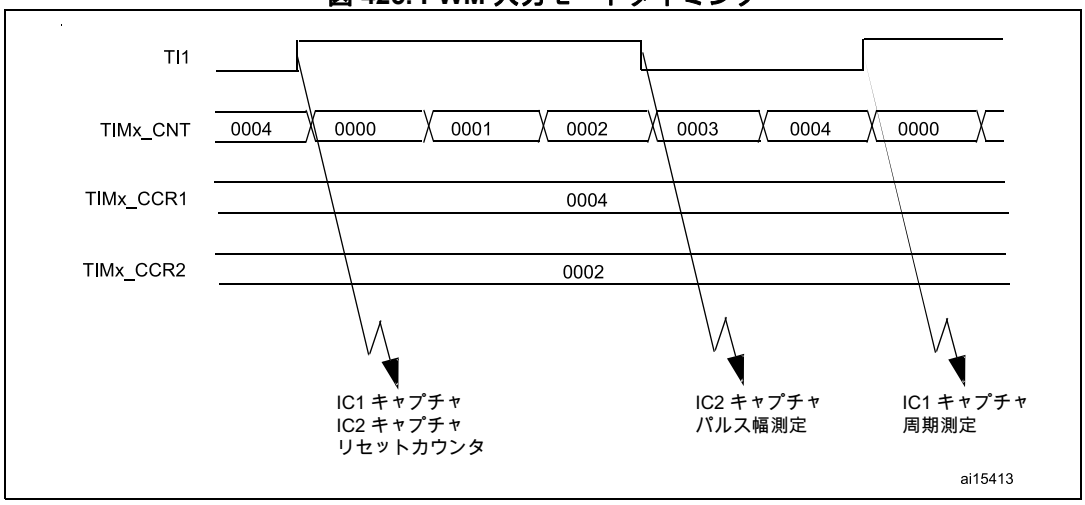
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
2. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブ極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用します）。
3. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR 2 のアクティブ入力を選択します。
4. CC2P ビットに“1”を、CC2NP ビットに“0”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
5. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
6. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
7. TIMx_CCER レジスタの CC1E と CC2E ビットに 1 を書き込むことによって、キャプチャを有効にします。

図 425. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

39.3.7 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS=00）では、各出力比較信号（OCxREF、そして OCx）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（OCxREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書き込みます。これにより、OCxREF は強制的にハイレベルになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、OCxREF 信号を強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、出力比較モードのセクションで説明します。

39.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

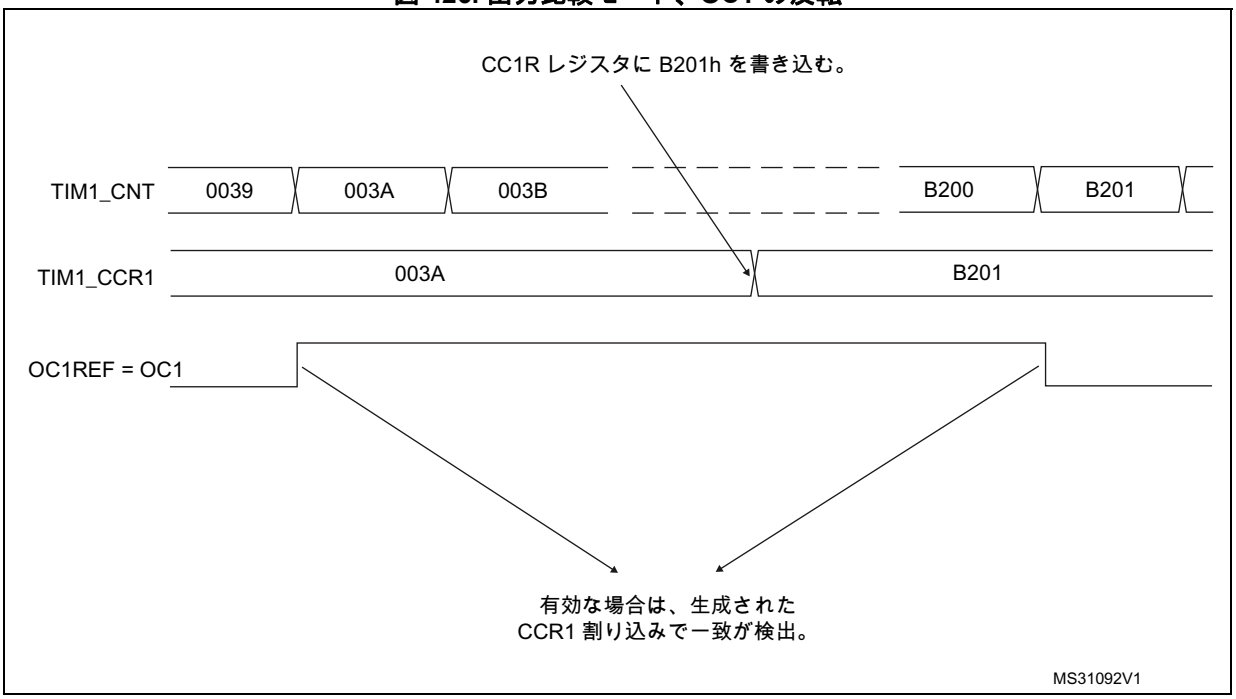
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエスト／DMA リクエストを生成する場合は、CCxIE ビット／CCxDE ビットをセットします。
4. 出力モードを選択します。たとえば、CNT が CCRx と一致したときに OCx 出力をトグルし、CCRx プリロードを使用せず、OCx が有効でアクティブハイのときには、OCxM=011、OCxPE=0、CCxP=0、CCxE=1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアで TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 426* に示します。

図 426. 出力比較モード、OC1 の反転



39.3.9 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、個々のチャネル（OCx 出力ごとに PWM 1 波形）で、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）や“111”（PWM モード 2）を書き込むことで、独自に選択できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

ただし、OCREF_CLR 機能（OCREF は、次の PWM 周期までは ETR 信号を通じて外部イベントによってクリアできる ）に従って、OCREF 信号は次の場合にのみアサートされます。

- 比較結果が変化したとき、または
- 出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）が停止構成（比較なし、OCxM=000）から PWM モードの 1 つ（OCxM=110 または 111）へ切り替えられたとき。

タイマの動作中は、ソフトウェアで強制的に PWM になります。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

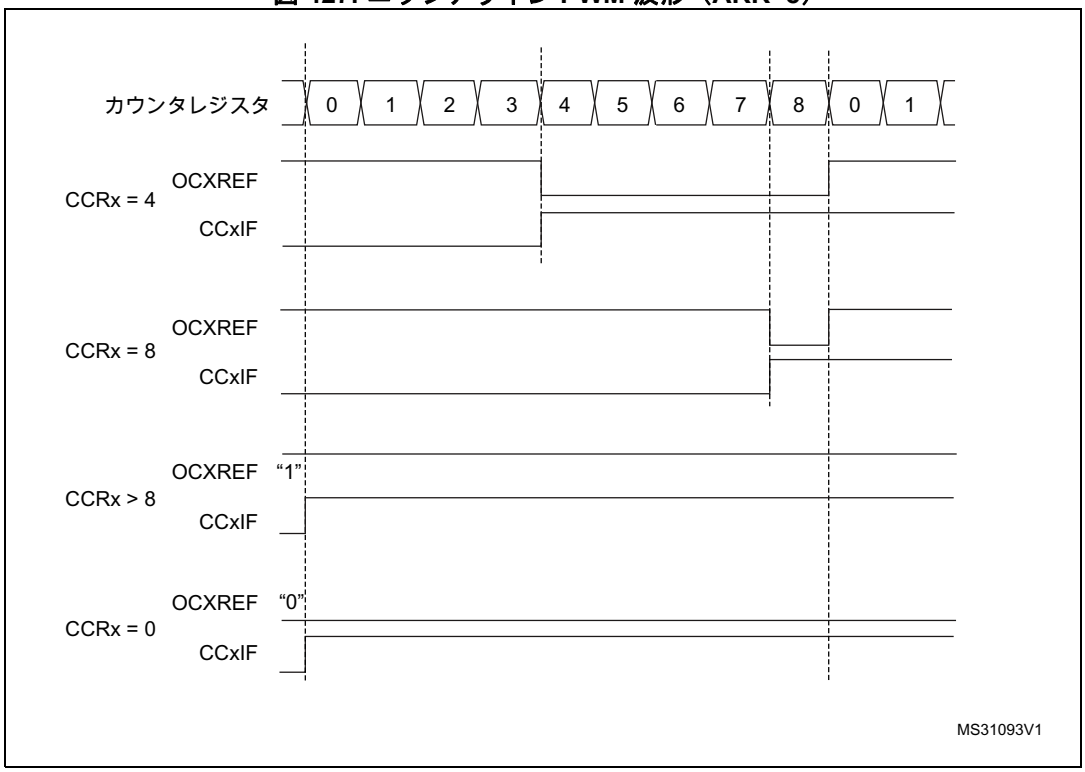
PWM エッジアラインモード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。*1567 ページのアップカウントモード*を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”です。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 427* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 427. エッジアライン PWM 波形（ARR=8）



ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。*1570 ページのダウンカウントモード*を参照してください。

PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、ocxref は 100% です。このモードでは、PWM 信号を生成することはできません。

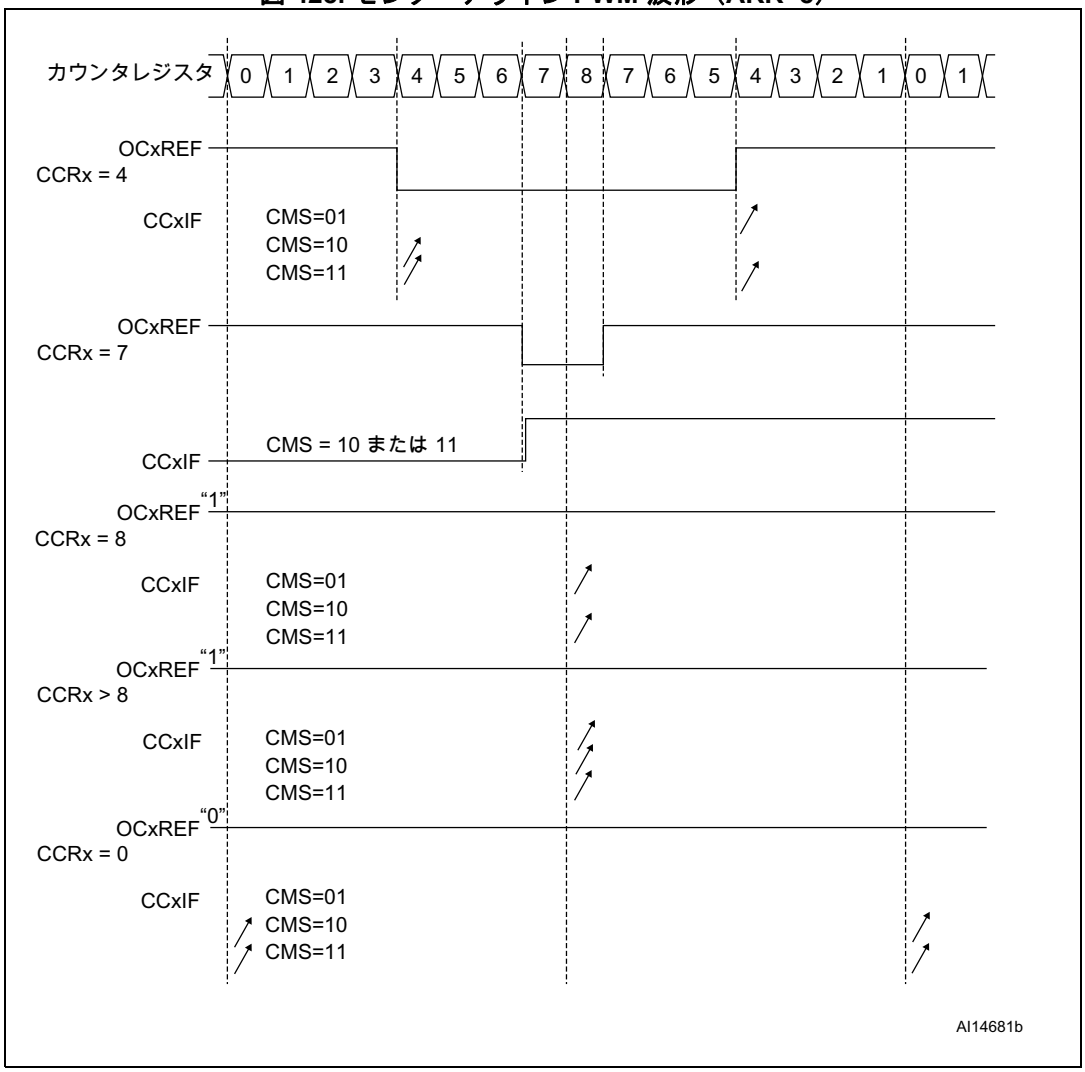
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての設定は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。*1573 ページのセンターアラインモード（アップ／ダウンカウント）*を参照してください。

図 428 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 428. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント：

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値をカウンタに書き込んだ場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書き込みを行わないことです。

39.3.10 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

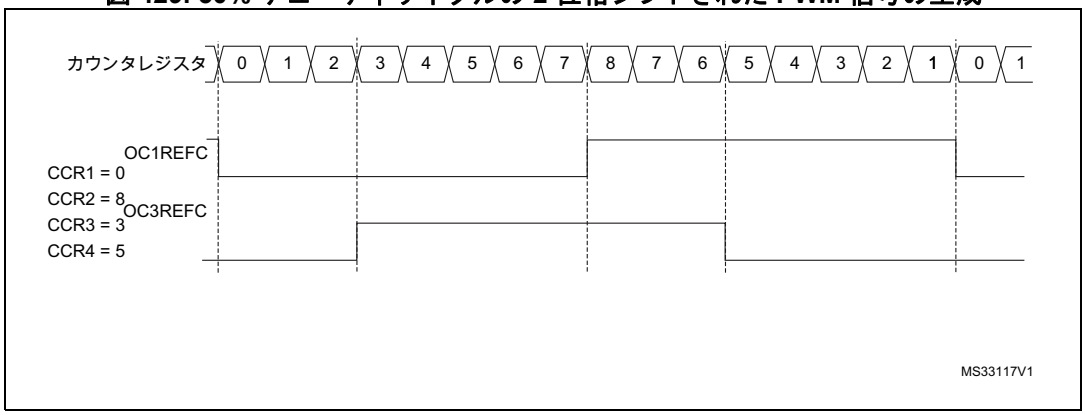
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

特定のチャネルが非対称の PWM チャネルとして使用されると、その 2 次チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 2 の結果として得られる OC2REFC 信号を出力できます。

図 429 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。

図 429. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



39.3.11 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

– OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。

– OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

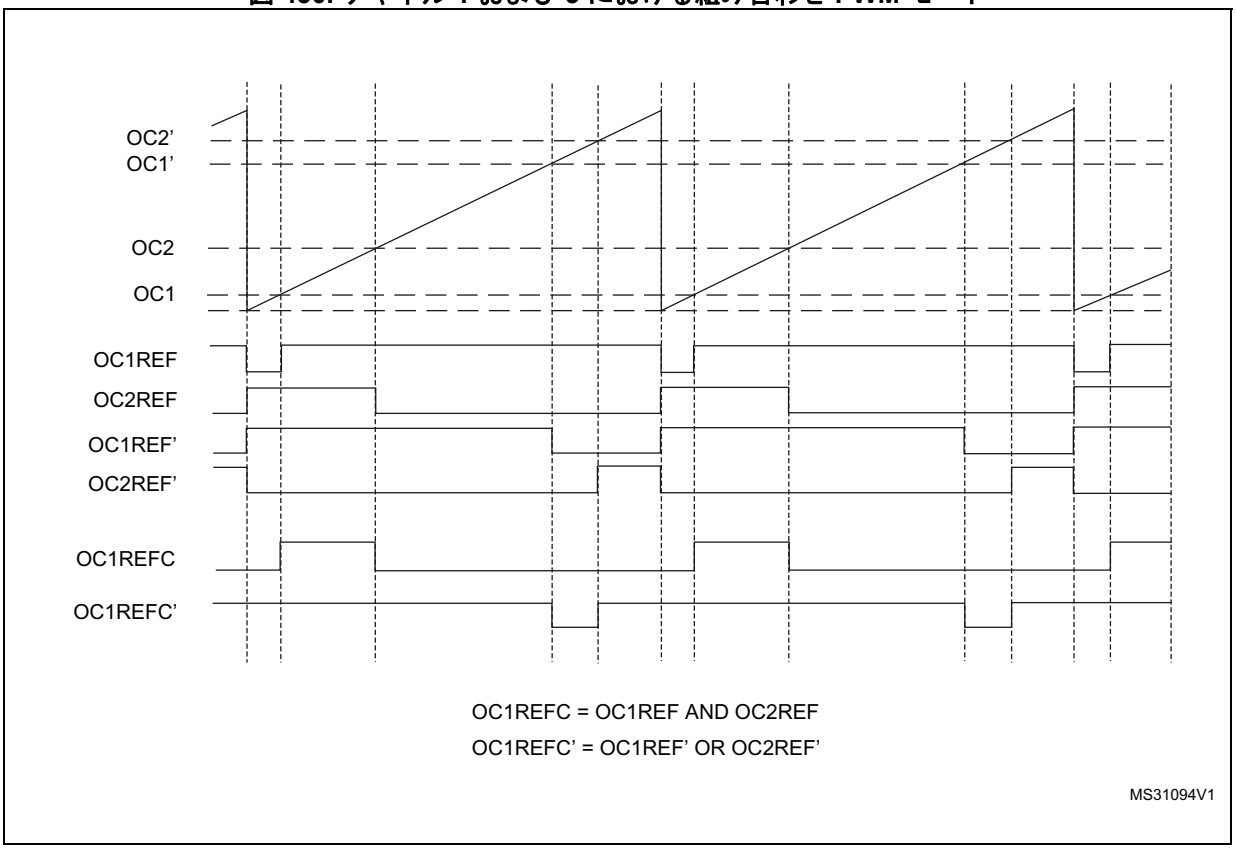
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、2 次チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 430 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 430. チャネル 1 および 3 における組み合わせ PWM モード



39.3.12 外部イベントによる OCxREF 信号のクリア

特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

ocref_clr_int は ETRF 信号（フィルタリング後の ETR）に接続されます。

特定のチャネルの OCxREF 信号は ETRF 入力をハイレベルとする（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを 1 にセットする）ことでリセットすることができます。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。

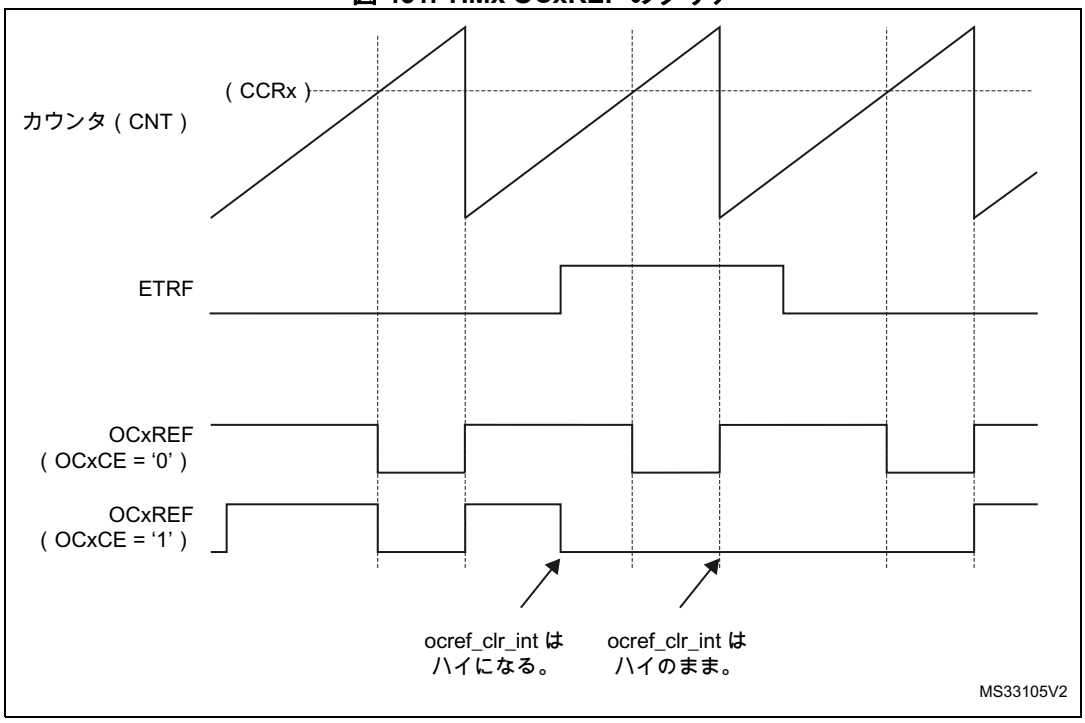
この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

たとえば、OCxREF 信号をコンパレータの出力に接続して、電流処理に使用することができます。この場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラをオフに保つ必要があります。すなわち、TIMx_SMCR レジスタのビット ETPS[1:0] が 00 にクリアされます。
2. 外部クロックモード 2 を無効にする必要があります。すなわち、TIM1_SMCR レジスタのビット ECE が 0 にクリアされます。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて設定できます。

図 431 に、OCxCE イネーブルビットの両方の値について、ETRF 入力がハイになったときの OCxREF信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 431. TIMx OCxREF のクリア



注： ***100% デューティサイクルの PWM の場合 （CCRx>ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。***

39.3.13 ワンパルスモード

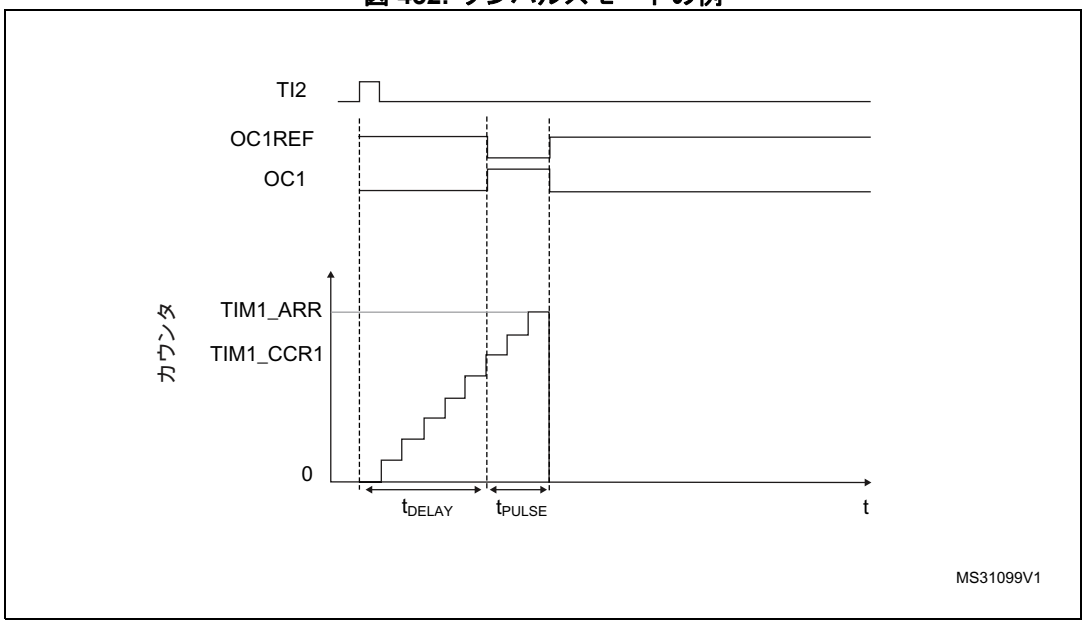
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 432. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタに IC2S=01 を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=「0」と CC2NP =「0」を書き込みます。
- TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
- TI2FP2 を使用してカウンタを開始するために、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。オプションで、TIMx_CCMR1 レジスタの OC1PE=1 と TIMx_CR1 レジスタの ARPE=1 を書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較を考慮せずにトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

39.3.14 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、*セクション 39.3.13* で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] = 「1000」（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が「1000」または「1001」にセットされています。

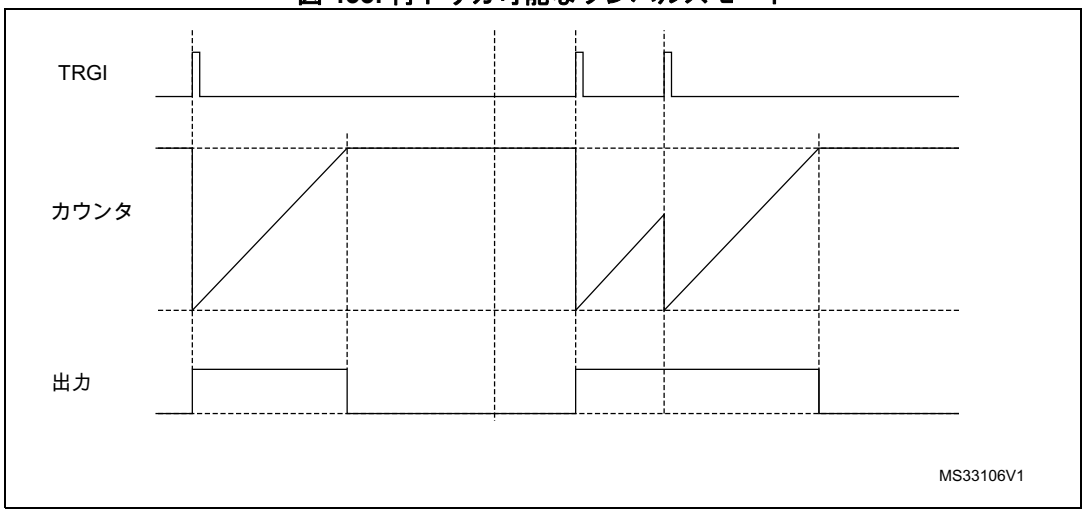
タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： ***再トリガ可能なワンパルスモードでは、CCxIF フラグは意味を持ちません。***

OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 433. 再トリガ可能なワンパルスモード



39.3.15 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。CC1NP と CC2NP はクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、インクリメンタルエンコーダとのインタフェースに使用されます。*表 316* を参照してください。カウンタのクロックは、TI1FP1 または TI2FP2（入力フィルタおよび極性選択後は TI1 および TI2。フィルタされず、反転されない場合は TI1FP1=TI1、フィルタされず、反転されない場合は TI2FP2=TI2）の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります（TIMx_CR1 レジスタの CEN ビットが“1”）。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、およびトリガ出力機能は、通常動作を続けます。

このモードでは、カウンタは直交 エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 316. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

外部インクリメンタルエンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

図 434 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S= 01（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S= 01（TIMx_CCMR2 レジスタ、TI2FP2 は TI2 に配置）
- CC1P = 0、CC1NP = 0、IC1F = 0000（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1 = TI1）
- CC2P = 0、CC2NP = 0、IC1F = 0000（TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2 = TI2）
- SMS=011（TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=1（TIMx_CR1 レジスタ、カウンタ有効）

図 434. エンコーダインタフェースモードにおけるカウンタの動作例

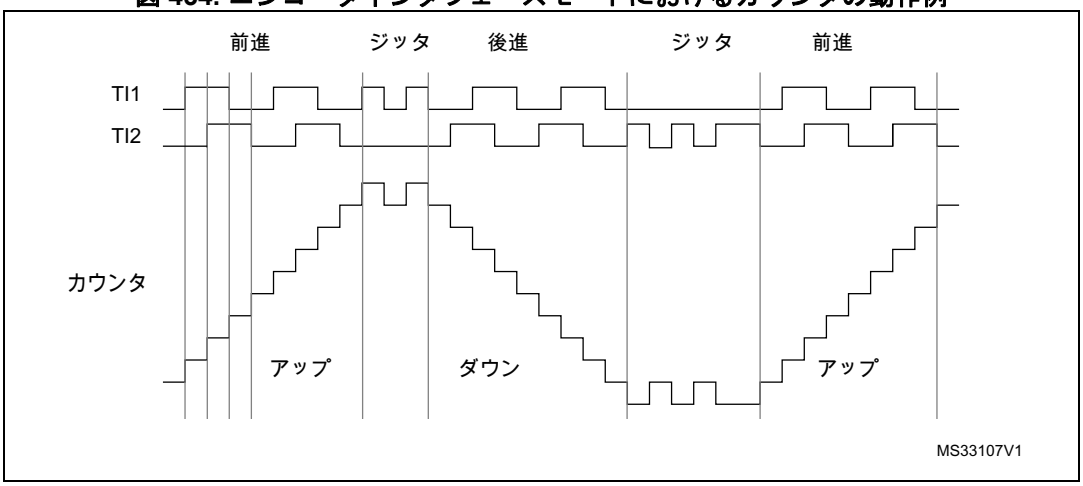
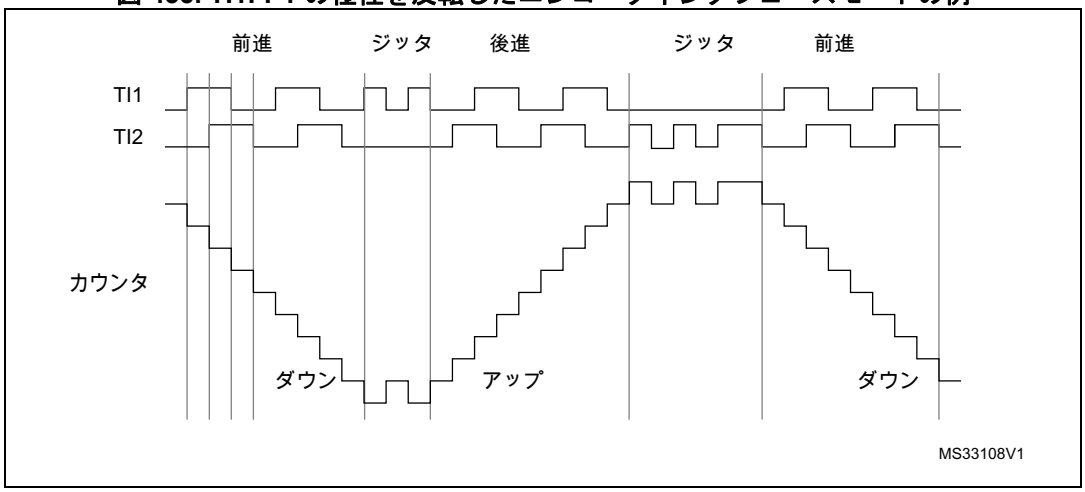


図 435 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図 435. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、リアルタイムクロックによって生成される DMA リクエストを通じて値を読み出すことも可能です。

39.3.16 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読み出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書き込みモード時のみアクセス可能）。

39.3.17 タイマ入力 XOR 機能

TIM1xx_CR2 レジスタの TI1S ビットを使用して、チャネル 1 の入力フィルタを、TIMx_CH1 から TIMx_CH3 までの 3 つの入力ピンを組み合わせた XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

この機能をホールセンサのインタフェースに使用した例を*セクション 38.3.25：1507 ページのホールセンサとのインタフェース*に示します。

39.3.18 タイマと外部トリガの同期

TIMx タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

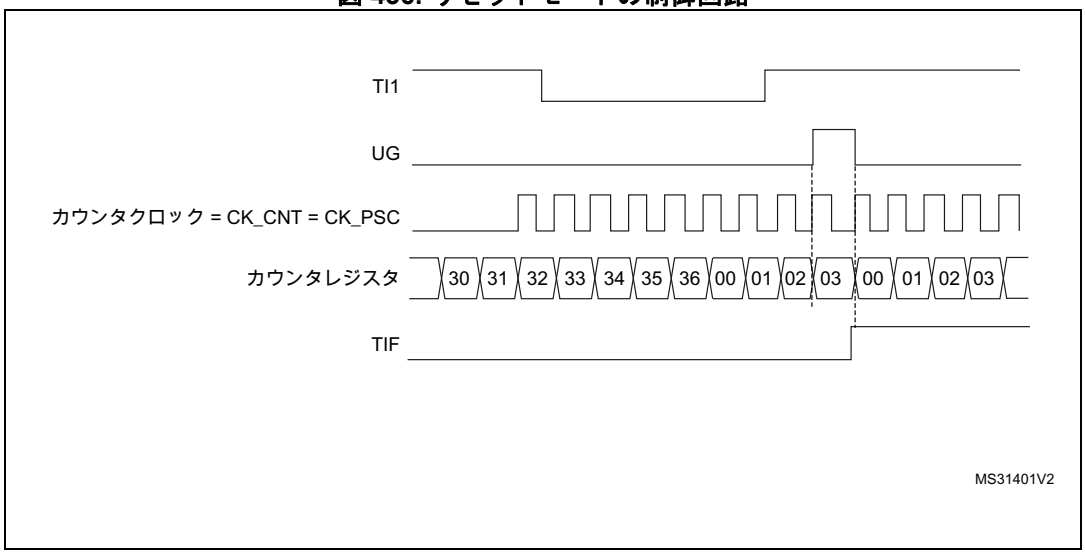
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 436. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

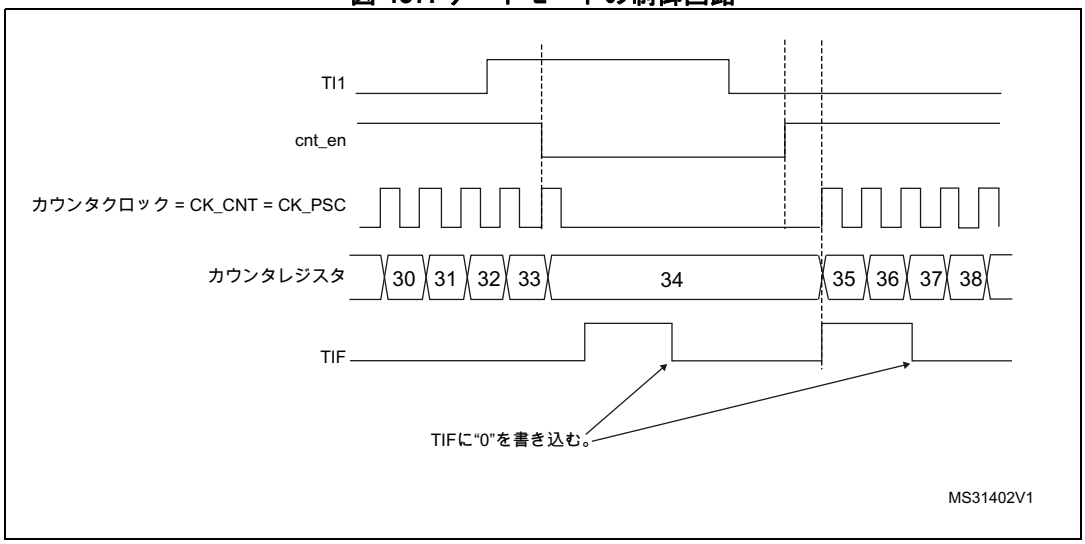
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 437. ゲートモードの制御回路



1. ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。

注： ***ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。***

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

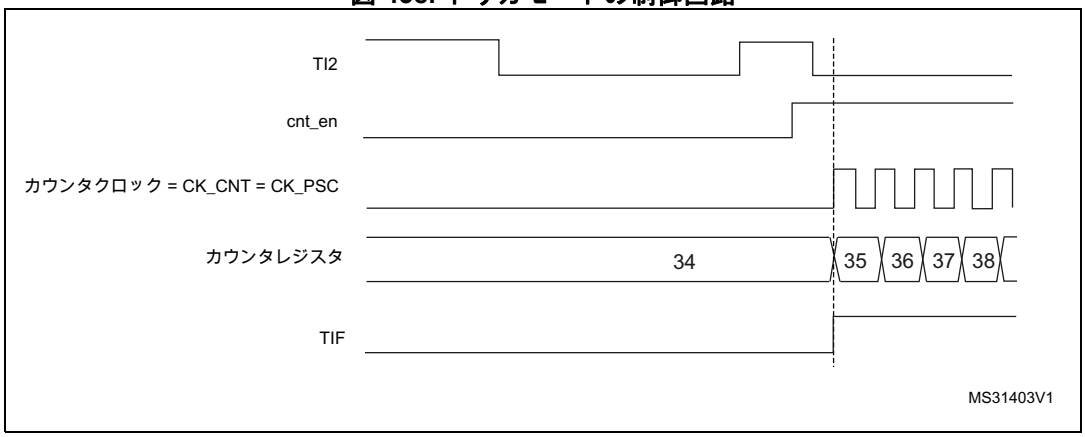
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 438. トリガモードの制御回路



スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

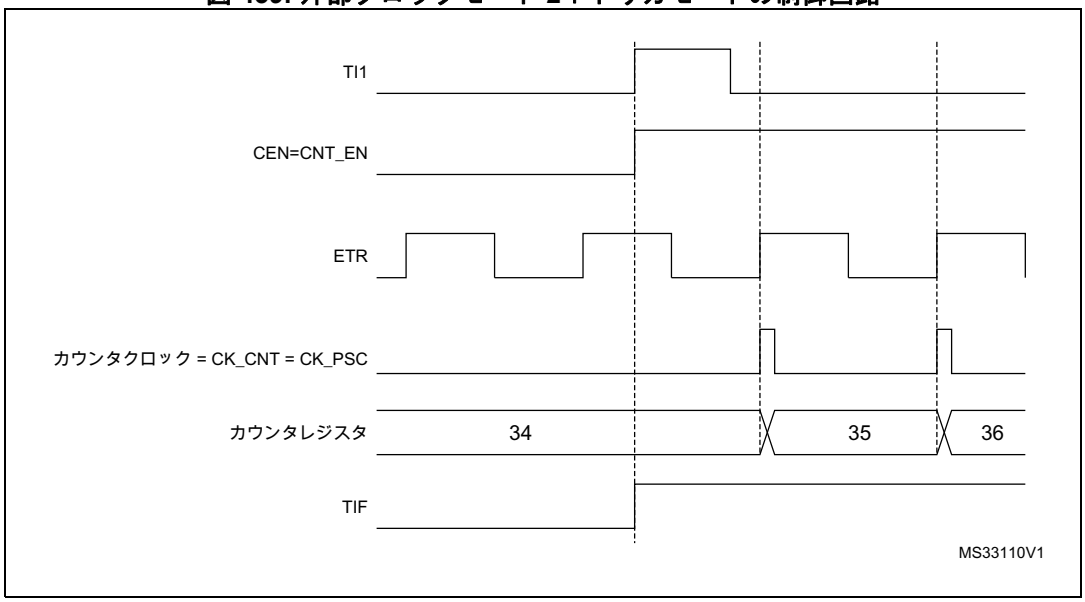
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 439. 外部クロックモード 2＋トリガモードの制御回路

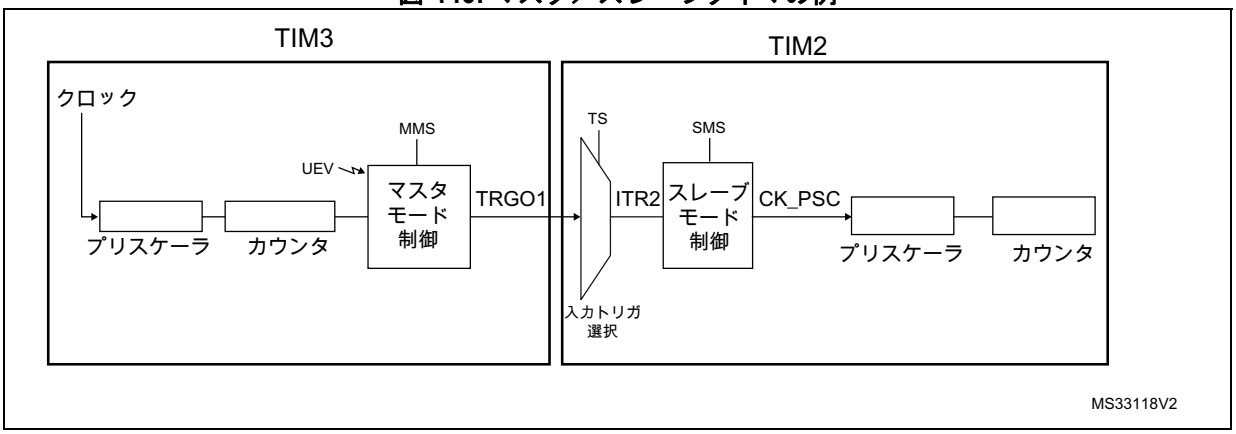


39.3.19 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

*図 440：マスタ／スレーブタイマの例*に、トリガ選択およびマスタモード選択ブロックの概要を示します。

図 440. マスタ／スレーブタイマの例



タイマを別のタイマのプリスケーラとして使用する

たとえば、TIM3 が TIM2 のプリスケーラとして動作するように設定できます。*図 440*を参照してください。このためには、次の操作を行います。

1. 更新イベント UEV ごとに周期的なトリガ信号を出力するように、TIM 3 をマスタモードに設定します。TIM3_CR2 レジスタの MMS=010 を書き込んだ場合、更新イベントが生成されるたびに、TRGO で立ち上がりエッジが出力されます。
2. TIM3 の TRGO 出力を TIM2 に接続するには、ITR2 を内部トリガとして使用して、TIM2 をスレーブモードで設定する必要があります。このためには、TIM2_SMCR レジスタの TS ビットを使用します（TS=00010 を書き込みます）。
3. 次に、スレーブモードコントローラを外部クロックモード 1 にします（TIM2_SMCR レジスタの SMS=111 を書き込みます）。これにより TIM2 は、TIM3 の周期的なトリガ信号の立ち上がりエッジ（TIM3 カウンタのオーバーフローに対応しています）をクロックとして動作します。
4. 最後に、それぞれの CEN ビット（TIMx_CR1 レジスタ）をセットすることによって、両方のタイマを有効にする必要があります。

注： ***TIM3 のトリガ出力として OCx が選択された場合（MMS=1xx）、その立ち上がりエッジが TIM2 カウンタのクロックとして使用されます。***

タイマを使用して別のタイマを有効にする

この例では、TIM2 の有効化を、タイマ 3 の出力比較 1 で制御します。接続については、*図 440* を参照してください。TIM2 は、TIM3 の OC1REF がハイのときのみ、分周された内部クロックをカウントします両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. CEN ビット（TIM2_CR1 レジスタ）に「1」を書き込んで、TIM2 を有効にします。
6. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

注： ***カウンタ 2 のクロックはカウンタ 1 と同期しないので、このモードは TIM2 カウンタのイネーブル信号にのみ影響します。***

図 441. TIM3 の OC1REF による TIM2 のゲート操作

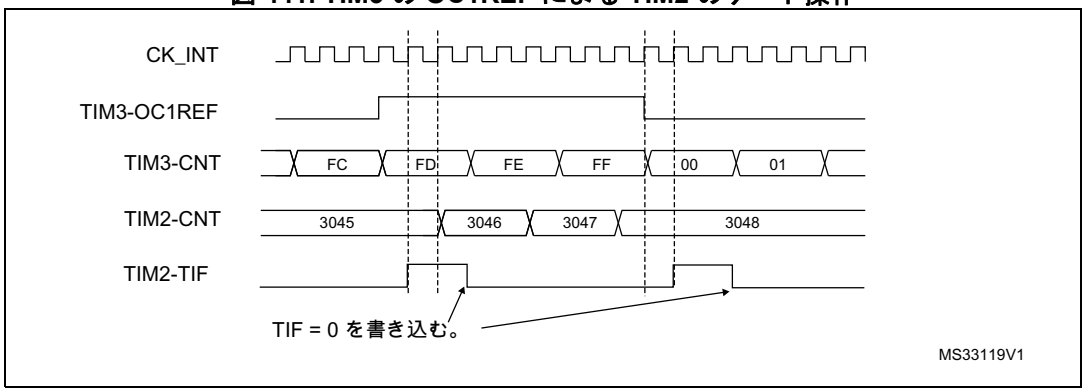
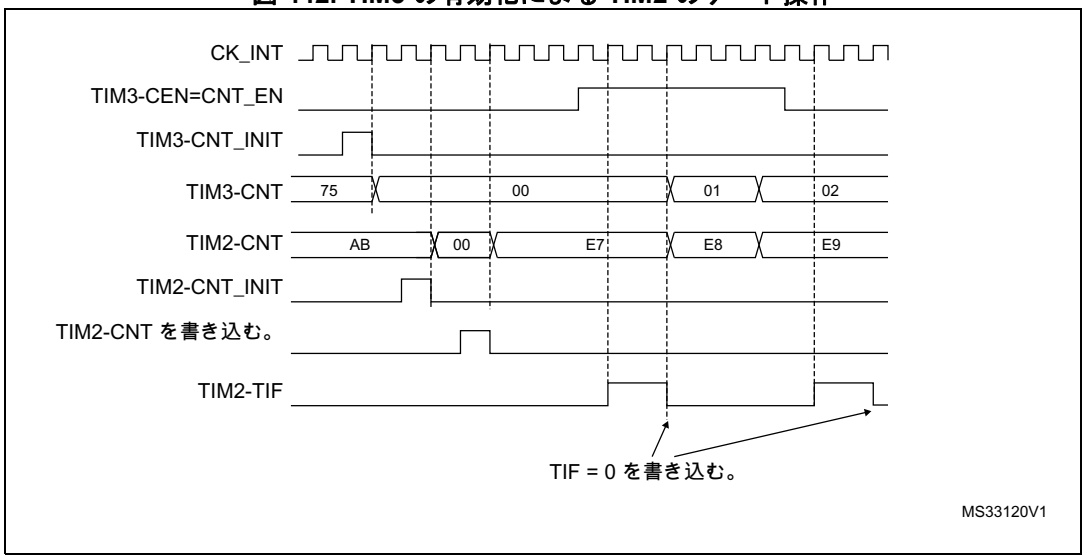


図 441 の例では、TIM2 のカウンタとプリスケーラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM 3 を開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。この場合、タイマカウンタに任意の値を書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では（*図 442* を参照）、TIM3 と TIM2 を同期させます。TIM 3 はマスタであり、0 からカウントを開始します。TIM2 はスレーブであり、0xE7 から開始します。プリスケーラの分周比は両方のタイマで同じです。TIM3_CR1 レジスタの CEN ビットに「0」を書き込むことによって TIM3 を無効にすると、TIM2 は停止します。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. UG ビット（TIM3_EGR レジスタ）に“1”を書き込んで、TIM3 をリセットします。
6. UG ビット（TIM2_EGR レジスタ）に“1”を書き込んで、TIM2 をリセットします。
7. TIM2 カウンタ（TIM2_CNTL）に“0xE7”を書き込んで、TIM2 を 0xE7 に初期化します。
8. CEN ビット（TIM2_CR1 レジスタ）に「1」を書き込んで、TIM2 を有効にします。
9. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。
10. CEN ビット（TIM3_CR1 レジスタ）に“0”を書き込んで、TIM3 を停止します。

図 442. TIM3 の有効化による TIM2 のゲート操作

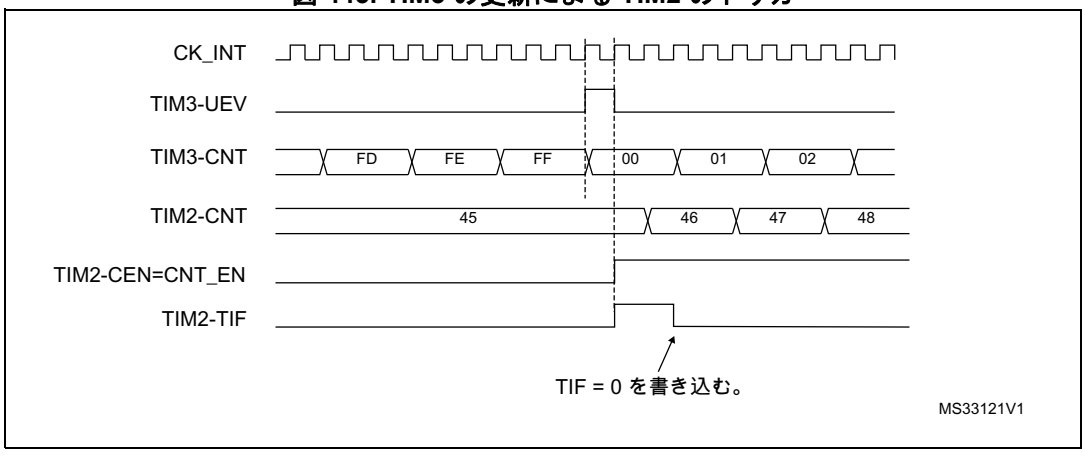


タイマを使用して別のタイマを開始する

この例では、タイマ 3 の更新イベントによってタイマ 2 の有効化を設定します。接続については、*図 440* を参照してください。タイマ 1 によって更新イベントが生成されると、タイマ 2 は、分周された内部クロックで現在値（ゼロである必要はありません）からカウントを開始します。タイマ 2 がトリガ信号を受信すると、その CEN ビットが自動的にセットされ、カウンタは TIM2_CR1 レジスタの CEN ビットに“0”が書き込まれるまでカウントします。両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

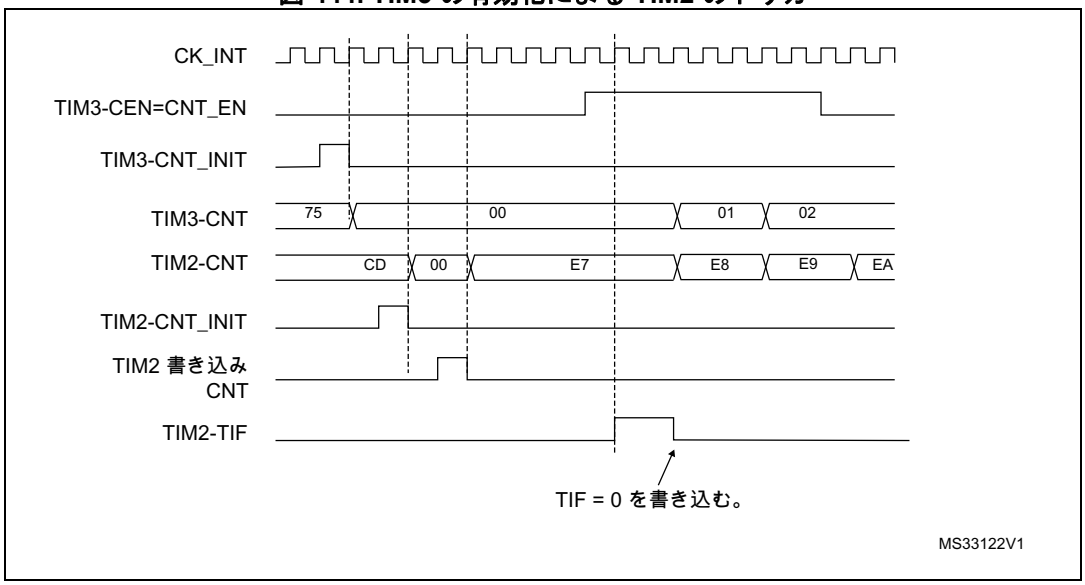
1. TIM3　をマスタモードに設定して、その更新イベント（UEV）をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=010）。
2. TIM3 の周期を設定します（TIM3_ARR レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。
5. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

図 443. TIM3 の更新による TIM2 のトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化することができます。*図 444* は *図 443* と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します（TIM2_SMCR レジスタの SMS=110）。

図 444. TIM3 の有効化による TIM2 のトリガ



外部トリガに対応して 2 つのタイマを同期して開始します。

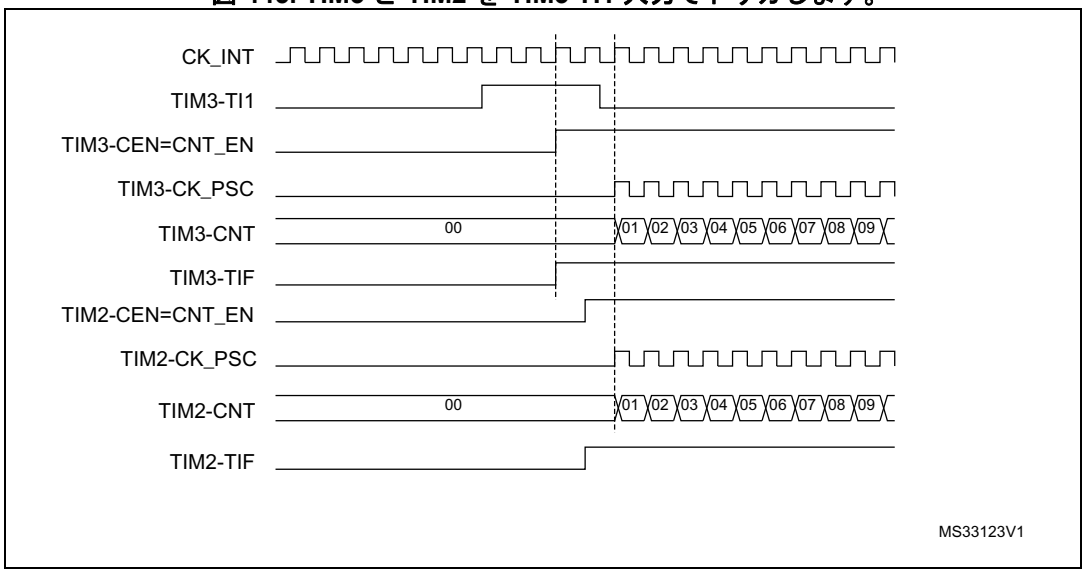
この例では、TI1 入力の立ち上がりエッジで TIM3 を有効にし、TIM3 の有効で、TIM2 を有効にします。接続については、*図 440* を参照してください。カウンタの同時性を確保するため、TIM3 はマスタ／スレーブモードに設定する必要があります（TI1 に対してはスレーブ、TIM2 に対してはマスタ）。

1. TIM 3 をマスタモードに設定して、その有効化をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=001）。
2. TIM3 をスレーブモードに設定して、TI1 から入力トリガを受け取るようにします（TIM3_SMCR レジスタの TS=00100）。
3. TIM3 をトリガモードに設定します（TIM3_SMCR レジスタの SMS=110）。
4. MSM=1（TIM3_SMCR レジスタ）を書き込むことによって、TIM3 をマスタ/スレーブモードに設定します。
5. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00000）。
6. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。

TI1（TIM3）で立ち上がりエッジが発生すると、両方のカウンタが同時に内部クロックによるカウントを開始し、両方の TIF フラグがセットされます。

注： ***この例では、両方のタイマが開始前に初期化されます（それぞれの UG ビットをセットすることによって）。両方のカウンタは 0 から開始しますが、カウンタレジスタ（TIMx_CNT）に書き込むことによって、容易にオフセットを挿入できます。マスタ／スレーブモードでは、TIM3 の CNT_EN と CK_PSC の間に遅延が挿入されます。***

図 445. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。



注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

39.3.20 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

39.3.21 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 with FPU コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの TIMx 設定ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、*セクション 60.5.8：マイクロコントローラデバッグユニット（DBGMCU）*を参照してください。

安全のため、カウンタが停止すると（DBGMCU_APB1FZ2 の TIMx = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

39.4 TIM2/TIM3/TIM4/TIM5 レジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

39.4.1 TIMx 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIF RE-MAP	Res.	CKD[1:0]		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（ETR、TIx）によって使用されるサンプリングクロックとの間の分周比を示します。
00: $t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2 \times t_{CK_INT}$
10：$t_{DTS} = 4 \times t_{CK_INT}$
11：予約済み

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： ***カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。***

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。
1：カウンタはダウンカウンタとして使用されます。

注： ***このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読み出し専用です。***

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ***外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

39.4.2 TIMx 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。参照先：*セクション 38.3.25：1507 ページのホールセンサとのインタフェース*

ビット 6:4 **MMS**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）

100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。

110：比較 - OC3REF 信号がトリガ出力（TRGO）として使用されます。

111：比較 - OC4REF 信号がトリガ出力（TRGO）として使用されます。

注：　スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

39.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **TS[4:3]**：トリガ選択 - ビット 4:3
TS[2:0] 説明を参照 - ビット 6:4

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3
SMS 説明を参照 - ビット 2:0

ビット 15 **ETP**：外部トリガ極性
このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。
0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。
1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブになります。

ビット 14 **ECE**：外部クロックイネーブル
このビットは、外部クロックモード 2 を有効にします。
0：外部クロックモード 2 は無効です。
1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。
1：ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=00111）。
2：外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 00111 でないことが必要）。
3：外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力はETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ
外部トリガ信号 ETRP の周波数は、最大でも CK_INT 周波数の 1/4 でなければなりません。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。
00：プリスケーラオフ
01：ETRP 周波数は 2 分周されます。
10：ETRP 周波数は 4 分周されます。
11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001： $f_{SAMPLING} = f_{CK_INT}$、N = 2
0010： $f_{SAMPLING} = f_{CK_INT}$、N = 4
0011： $f_{SAMPLING} = f_{CK_INT}$、N = 8
0100： $f_{SAMPLING} = f_{DTS}/2$、N = 6
0101： $f_{SAMPLING} = f_{DTS}/2$、N = 8
0110： $f_{SAMPLING} = f_{DTS}/4$、N = 6
0111： $f_{SAMPLING} = f_{DTS}/4$、N = 8
1000： $f_{SAMPLING} = f_{DTS}/8$、N = 6
1001： $f_{SAMPLING} = f_{DTS}/8$、N = 8
1010： $f_{SAMPLING} = f_{DTS}/16$、N = 5
1011： $f_{SAMPLING} = f_{DTS}/16$、N = 6
1100： $f_{SAMPLING} = f_{DTS}/16$、N = 8
1101： $f_{SAMPLING} = f_{DTS}/32$、N = 5
1110： $f_{SAMPLING} = f_{DTS}/32$、N = 6
1111： $f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ／スレーブモード

0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[4:0]**：トリガ選択（TS[4:3] にビット 21:20 を参照）

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：内部トリガ 2（ITR2）

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

00111：外部トリガ入力（ETRF）

01000：内部トリガ 4（ITR4）

01001：内部トリガ 5（ITR5）

01010：内部トリガ 6（ITR6）

01011：内部トリガ 7（ITR7）

01100：内部トリガ 8（ITR8）

その他：予約済み

各タイマでの ITRx の詳細については、*表 317：1614 ページのTIMx 内部トリガ接続*を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ/ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ/ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力の立ち上がりエッジ（TRGI）

カウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

注： トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

表 317. TIMx 内部トリガ接続

スレーブ TIM	TIM2	TIM3	TIM4	TIM5
ITR0	TIM1	TIM1	TIM1	TIM1
ITR1	TIM8	TIM2	TIM2	TIM8
ITR2	TIM3	TIM15	TIM3	TIM3
ITR3	TIM4	TIM4	TIM8	TIM4
ITR4	ETH PPS	ETH PPS	-	-
ITR5	USB1 OTG_HS_SOF	-	-	-
ITR6	USB2 OTG_FS_SOF	-	-	fdcan1_soc
ITR7	-	-	-	USB1 OTG_HS_SOF
ITR8	-	-	-	USB2 OTG_FS_SOF

39.4.4 TIMx DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IE**：キャプチャ／比較 4 割り込みイネーブル
0：CC4 割り込みは無効です。
1：CC4 割り込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割り込みイネーブル
0：CC3 割り込みは無効です。
1：CC3 割り込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

39.4.5 TIMx ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割り込みが保留中です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IF**：キャプチャ／比較 4 割り込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割り込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときにハードウェアによってセットされます。センターアラインモード（TIMx_CR1 レジスタの CMS ビットの説明を参照）および再トリガ可能なワンパルスモードでは、例外もあります。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（選択された極性に一致するエッジが IC1 で検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

オーバーフローまたはアンダーフロー時（TIM2 から TIM4）、および TIMx_CR1 レジスタの UDIS=0 である場合。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

39.4.6 TIMx イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4G**：キャプチャ／比較 4 生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

39.4.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済み、常に 0 として読み出されます。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み、常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード
ビット 6:4 の OC1M 説明を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル
0：OC1Ref は ETRF 入力の影響を受けません。
1：OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT >TIMx_CCR1 の場合は非アクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、CC1S=00 （チャネルは出力に設定）のときには、変更できません。***

2：PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。
1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、CC1S=00 （チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。
0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。
1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済み、常に 0 として読み出されます。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。***

ビット 7:4 **IC1F**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E=0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

39.4.8 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）

アドレスオフセット：0x1C

リセット値：0x0000

上記の CCMR1 レジスタの説明を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M [3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]				IC3PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済み、常に 0 として読み出されます。

ビット 24 **OC4M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み、常に 0 として読み出されます。

ビット 16 **OC3M[3]**：出力比較 1 モード - ビット 3

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 14:12 **OC4M**：出力比較 4 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル

ビット 6:4 **OC3M**：出力比較 3 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 31:16 予約済み、常に 0 として読み出されます。

ビット 15:12 **IC4F**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7:4 **IC3F**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

39.4.9 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット 15 **CC4NP**：*キャプチャ／比較 4 出力極性*

CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：*キャプチャ／比較 4 出力極性*

CC1P の説明を参照してください。

ビット 12 **CC4E**：*キャプチャ／比較 4 出力イネーブル。*

CC1E の説明を参照してください。

ビット 11 **CC3NP**：*キャプチャ／比較 3 出力極性*

CC1NP の説明を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC3P**：*キャプチャ／比較 3 出力極性*

CC1P の説明を参照してください。

ビット 8 **CC3E**：*キャプチャ／比較 3 出力イネーブル。*

CC1E の説明を参照してください。

ビット 7 **CC2NP**：*キャプチャ／比較 2 出力極性*

CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：*キャプチャ／比較 2 出力極性*

CC1P の説明を参照してください。

ビット 4 **CC2E**：*キャプチャ／比較 2 出力イネーブル。*

CC1E の説明を参照してください。

ビット 3 **CC1NP**：*キャプチャ／比較 1 出力極性*

CC1 チャネルが出力として設定されている場合：この場合、CC1NP はクリアされたままでなければなりません。

CC1 チャネルが入力として設定されている場合：このビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：*キャプチャ／比較 1 出力極性*

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ操作の TI1FP1 および TI2FP1 の極性を選択します。

00：非反転／立ち上がりエッジ

回路は TIxFP1 の立ち上がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ、エンコーダモード）。

01：反転／立ち下がりエッジ

回路は TIxFP1 の立ち下がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されます（ゲートモードでのトリガ、エンコーダモード）。

10：予約済み。この設定は使用しないでください。

11：非反転／両エッジ

回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ）。この設定をエンコーダモードに使用することはできません。

ビット 0 **CC1E**：*キャプチャ／比較 1 出力イネーブル。*

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。

1：オン - OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 318. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx=0、OCx_EN=0）
1	OCx=OCxREF + 極性、OCx_EN=1

注： ***標準 OCx チャネルに接続されている外部 IO ピンの状態は、OCx チャネルの状態と、GPIO および AFIO レジスタに依存します。***

39.4.10 TIMx カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT[31] または UIFCPY	CNT[30:16]（タイマに依存）														
rw または r	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 値は TIMx_CR1 の IUFREMAP に依存します。

UIFREMAP = 0 の場合：

CNT[31]：カウンタ値の最上位ビット（TIM2 および TIM5）

他のタイマで予約済み

UIFREMAP = 1 の場合：

UIFCPY：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー

ビット 30:16 **CNT[30:16]**：カウンタ値の最上位部分（TIM2 および TIM5）

ビット 15:0 **CNT[15:0]**：カウンタ値の最下位部分

39.4.11 TIMx プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

39.4.12 TIMx 自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARR[31:16]（タイマに依存）															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **ARR[31:16]**：自動再ロード値上位ビット（TIM2 および TIM5）

ビット 15:0 **ARR[15:0]**：自動再ロード値下位ビット

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、*セクション 39.3.1： 1565 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

39.4.13 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR1[31:16]**：キャプチャ／比較 1 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値下位ビット

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読み出し専用レジスタで、プログラムできません。

39.4.14 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR2[31:16]**：キャプチャ／比較 2 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値下位ビット

CC2 チャネルが出力として設定されている場合：
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。
CC2 チャネルが入力として設定されている場合：
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読み出し専用レジスタで、プログラムできません。

39.4.15 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCR3）

アドレスオフセット：0x3C

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR3[31:16]**：キャプチャ／比較 3 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR3[15:0]**：キャプチャ／比較値下位ビット

CC3 チャネルが出力として設定されている場合：
CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。
チャネル CC3 が入力として設定されている場合：
CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読み出し専用レジスタで、プログラムできません。

39.4.16 TIMx キャプチャ／比較モードレジスタ 4（TIMx_CCR4）

アドレスオフセット：0x40

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR4[31:16]**：キャプチャ／比較 4 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR4[15:0]**：キャプチャ／比較値下位ビット

1. CC4 チャネルが出力として設定されている場合（CC4S ビット）：
   CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。
   TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。
   アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。
2. CC4 チャネルが入力として設定されている場合（TIMx_CCMR4 レジスタの CC4S ビット）：
   CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読み出し専用レジスタで、プログラムできません。

39.4.17 TIMx DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、DMA 転送回数（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
...

例：次の転送を考えます：DBL = 7 回転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

39.4.18 完全転送の TIMx DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

39.4.19 TIM2 オルタネート機能オプションレジスタ 1 （TIM2_AF1）

アドレスオフセット：0x60

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：LSE

0100：SAI1 FS_A

0101：SAI1 FS_B

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

39.4.20 TIM3 オルタネート機能オプションレジスタ 1 （TIM3_AF1）

アドレスオフセット：0x60

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

39.4.21 TIM5 オルタネート機能オプションレジスタ 1 （TIM5_AF1）

アドレスオフセット：0x60

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：SAI2 FS_A が ETR 入力に接続されます。

0010：SAI2 FS_B が ETR 入力に接続されます。

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

39.4.22 TIM2 タイマ入力選択レジスタ（TIM2_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択

これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。

0000：TIM2_CH4 入力

0001：COMP1 出力

0010：COMP2 出力

0011：COMP1 出力または COMP2 出力

その他：予約済み

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択

これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。

0000：TIM2_CH3 入力

その他：予約済み

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM2_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM2_CH1 入力
その他：予約済み

39.4.23 TIM3 タイマ入力選択レジスタ（TIM3_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM3_CH4 入力
その他：予約済み

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM3_CH3 入力
その他：予約済み

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM3_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM3_CH1 入力
0001：COMP1 出力
0010：COMP2 出力
0011：COMP1 出力または COMP2 出力
その他：予約済み

39.4.24 TIM5 タイマ入力選択レジスタ（TIM5_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM5_CH4 入力
その他：予約済み

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM5_CH3 入力
その他：予約済み

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM5_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM5_CH1 入力
0001：fdcan1_tmp
0010：fdcan1_rtp
その他：予約済み

39.4.25 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマップされます。

表 319. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
	リセット値																									0	0	0	0	0			
0x08	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0		0		0	0	0	0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値																				0	0	0	0			0		0	0	0	0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																										0		0	0	0	0	0
0x18	**TIMx_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**TIMx_CCMR2** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	O24CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																	0		0	0	0		0	0	0		0	0	0		0	0

表 319. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**TIMx_CNT**	CNT[31] または UIFCPY	CNT[30:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）															CNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	ARR[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																ARR[15:0]															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	予約済み																																
0x34	**TIMx_CCR1**	CCR1[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR1[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIMx_CCR2**	CCR2[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR2[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIMx_CCR3**	CCR3[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR3[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIMx_CCR4**	CCR4[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR4[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	予約済み																																
0x48	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIMx_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	**TIM2_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														

表 319. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x60	**TIM3_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x60	**TIM5_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x68	**TIM2_TISEL**	Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]				Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x68	**TIM3_TISEL**	Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]				Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x68	**TIM5_TISEL**	Res.	Res.	Res.	Res.	TI4SEL[3:0]				Res.	Res.	Res.	Res.	TI3SEL[3:0]				Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

40 汎用タイマ（TIM12/TIM13/TIM14）

40.1 TIM12/TIM13/TIM14 の概要

TIM12/TIM13/TIM14 汎用タイマは、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

これらのカウンタは、入力信号パルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM）など様々な目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM12/TIM13/TIM14 タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、*セクション 40.3.17：タイマの同期（TIM12）*に示すように、相互に同期させることができます。

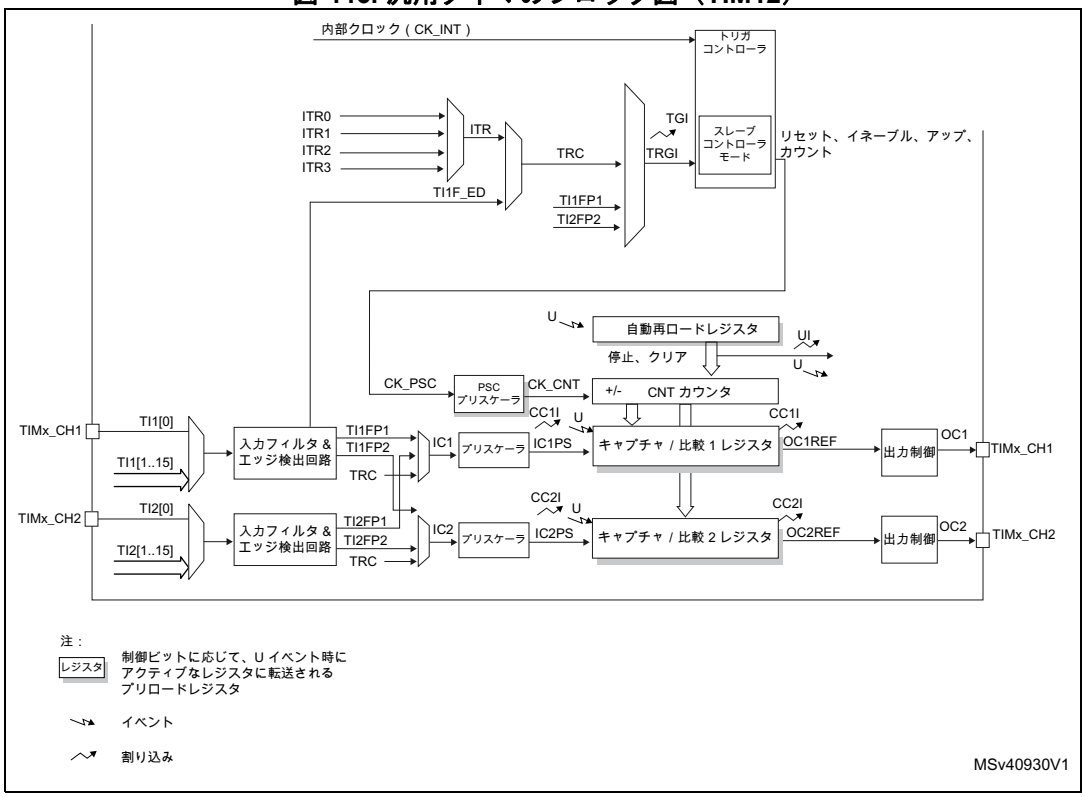
40.2 TIM12/TIM13/TIM14 の主な特長

40.2.1 TIM12 の主な特長

TIM12汎用タイマ　の機能は以下のとおりです。

- 16 ビット自動再ロードアップカウンタ
- カウンタクロック周波数を 1 から 65536 の間の値で分周するために使用される 16 ビットのプログラム可能なプリスケーラ（動作中に変更可能）。
- 次の機能を持つ、最大 2 つの独立チャネル。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割り込み生成。
  - 更新：カウンタオーバーフロー、カウンタ初期化（ソフトウェアまたは内部トリガによる）
  - トリガイベント（内部トリガによるカウンタの開始、停止、初期化、またはカウント）
  - 入力キャプチャ
  - 出力比較

図 446. 汎用タイマのブロック図（TIM12）

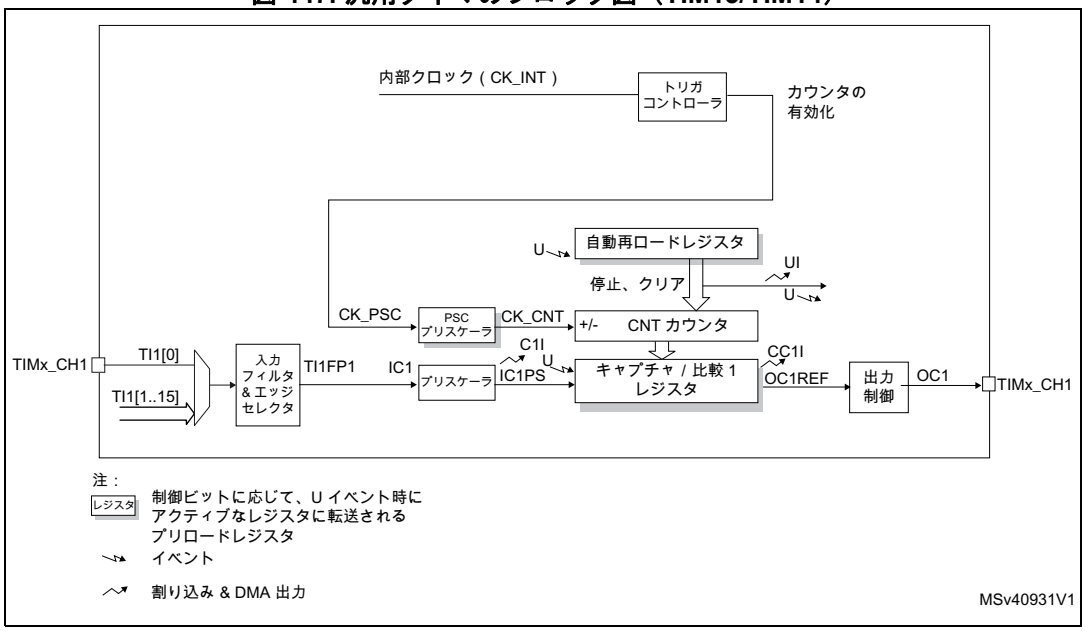


40.2.2 TIM13/TIM14 の主な特長

汎用タイマ TIM13/TIM14 の機能は以下のとおりです。

- 16 ビット自動再ロードアップカウンタ
- カウンタクロック周波数を 1 から 65536 の間の値で分周するために使用される 16 ビットのプログラム可能なプリスケーラ（動作中に変更可能）。
- 次の機能を持つ独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- 以下のイベント時の割り込み生成。
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアによる）
  - 入力キャプチャ
  - 出力比較

図 447. 汎用タイマのブロック図（TIM13/TIM14）



40.3 TIM12/TIM13/TIM14機能詳細

40.3.1 タイムベースユニット

タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。このカウンタはカウントアップします。

カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定ごとに詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 448 と *図 449* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 448. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

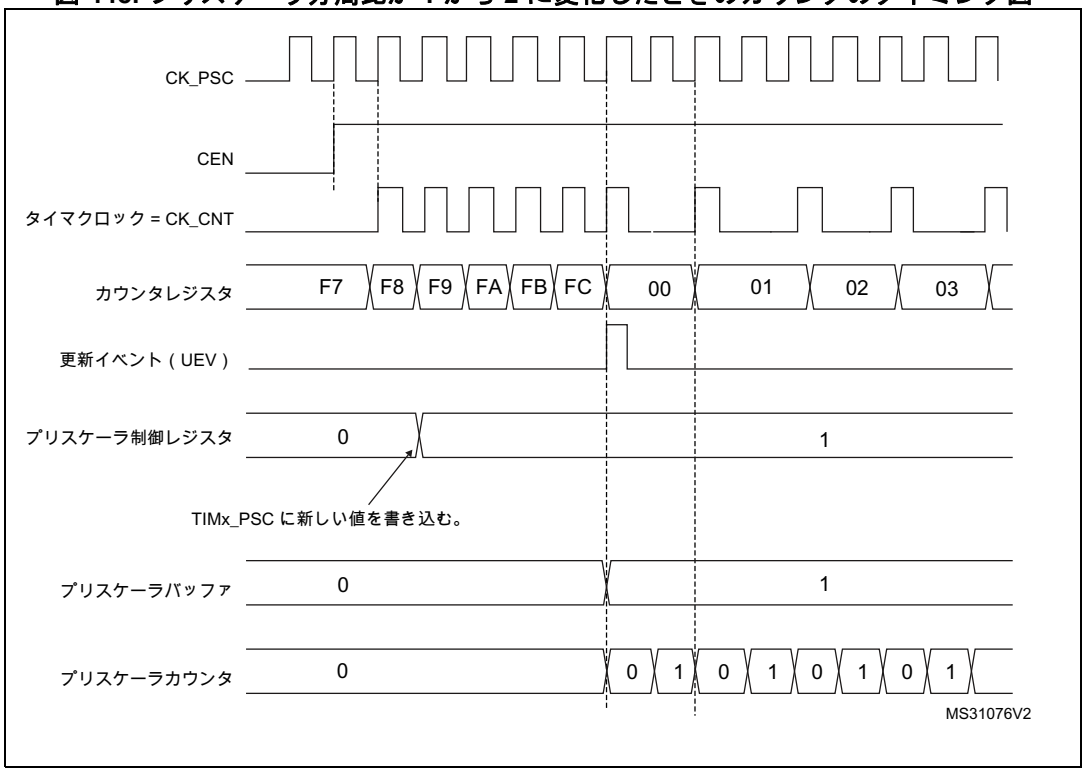
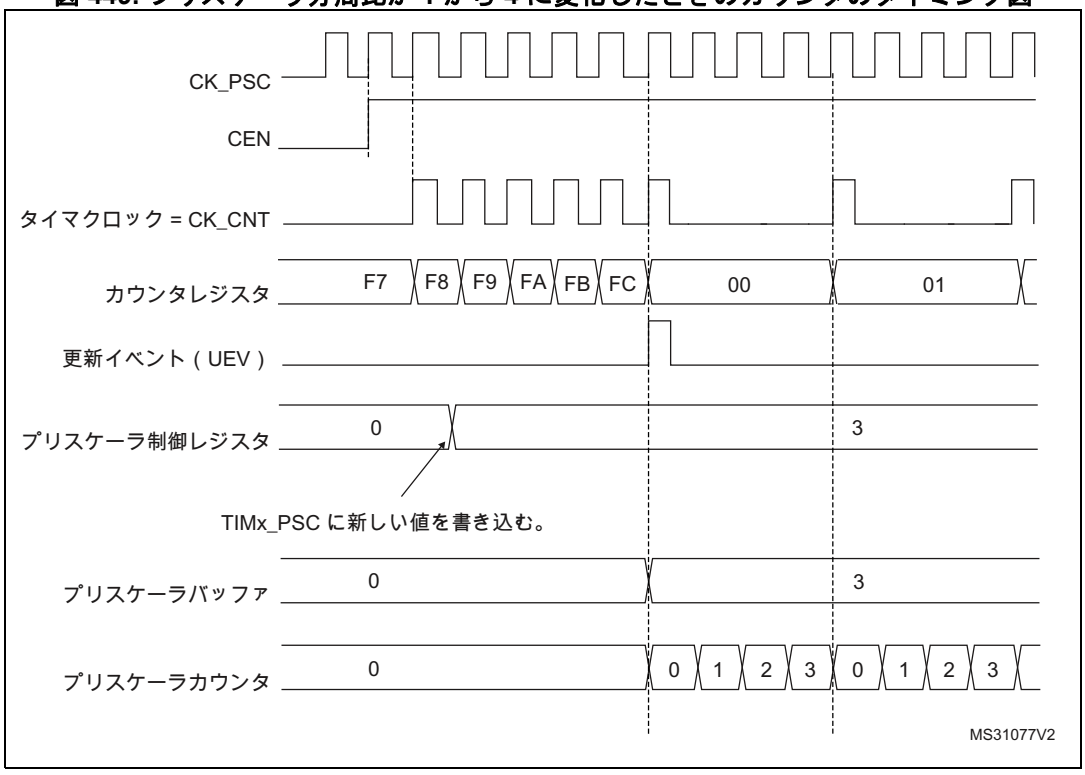


図 449. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



40.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

（ソフトウェアによって、または TIM12 のスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

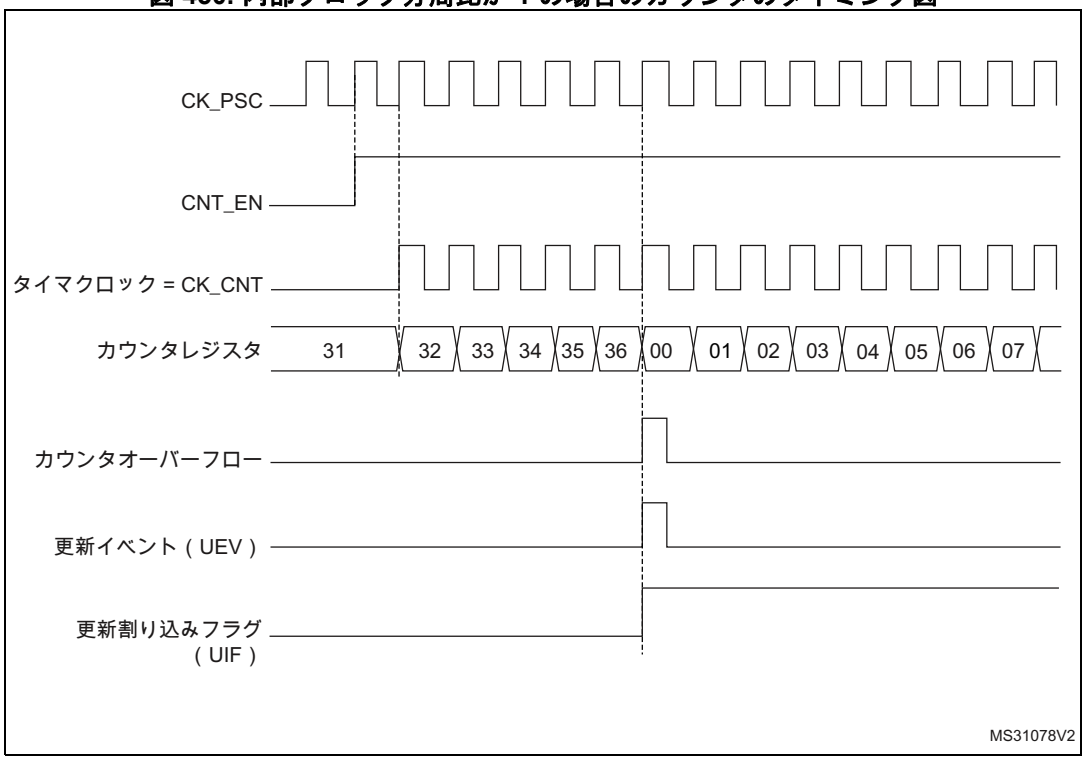
UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 450. 内部クロック分周比が 1 の場合のカウンタのタイミング図



参考資料

図 451. 内部クロック分周比が 2 の場合のカウンタのタイミング図

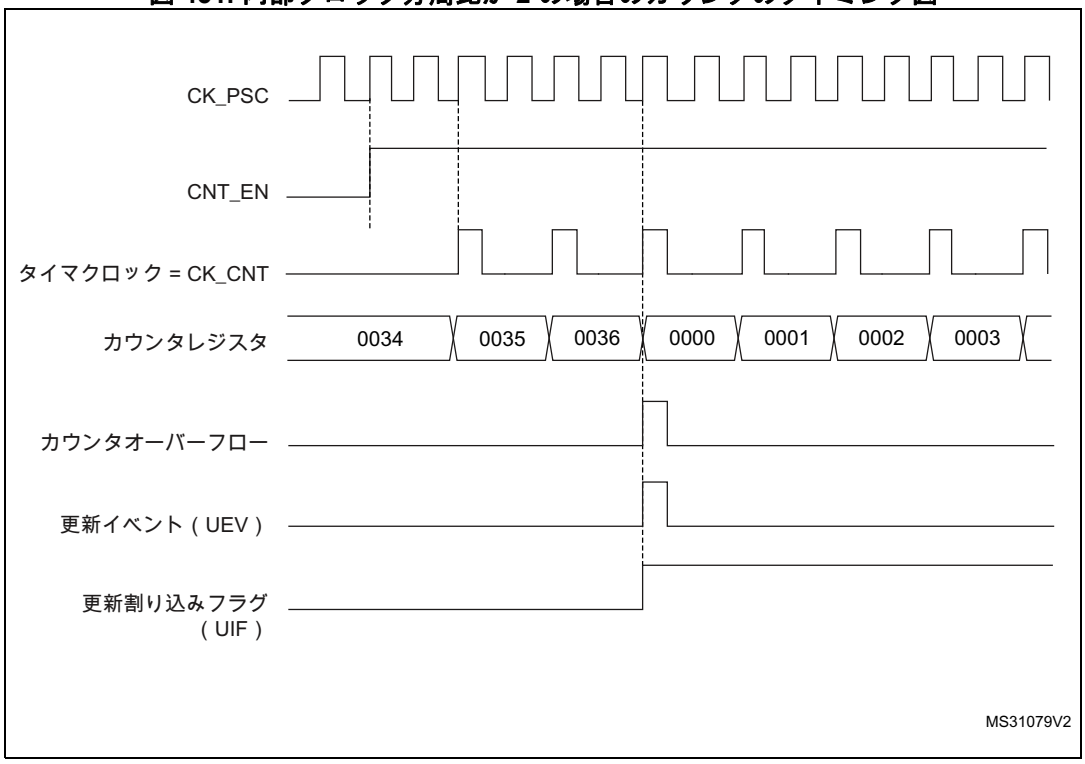


図 452. 内部クロック分周比が 4 の場合のカウンタのタイミング図

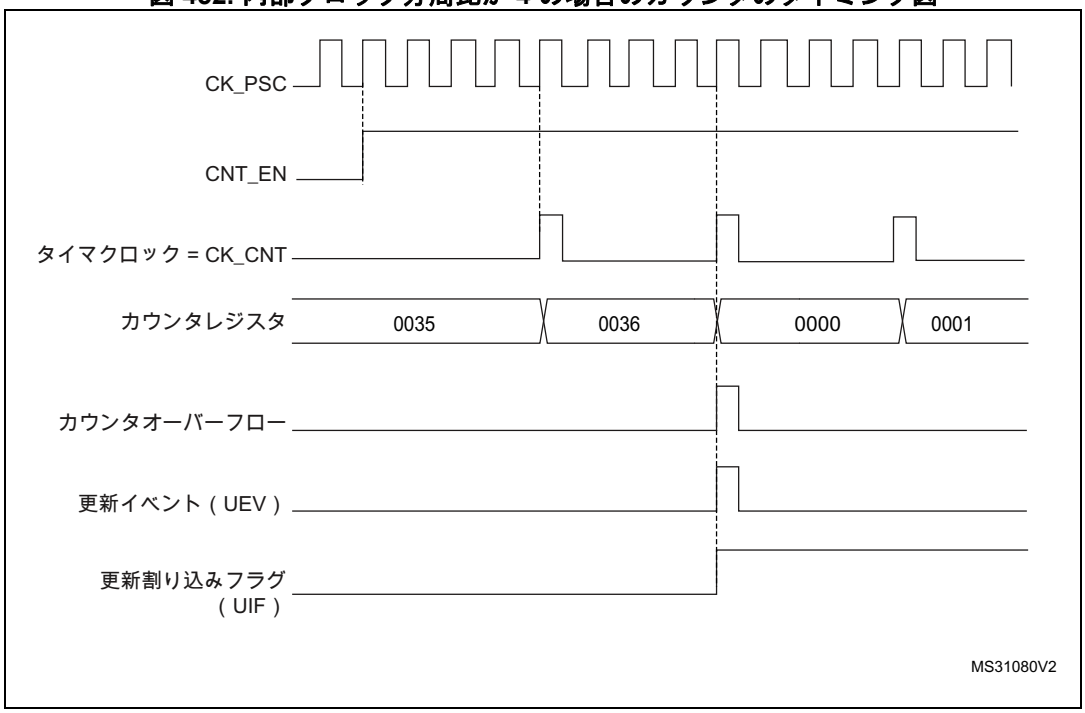


図 453. 内部クロック分周比が N の場合のカウンタのタイミング図

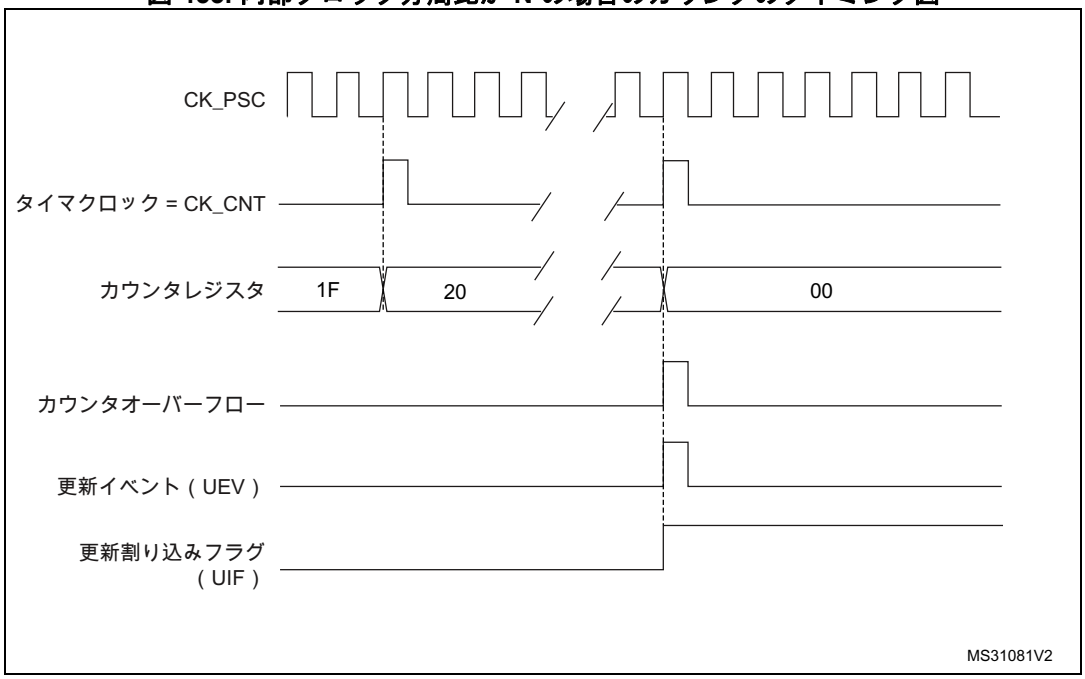


図 454. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

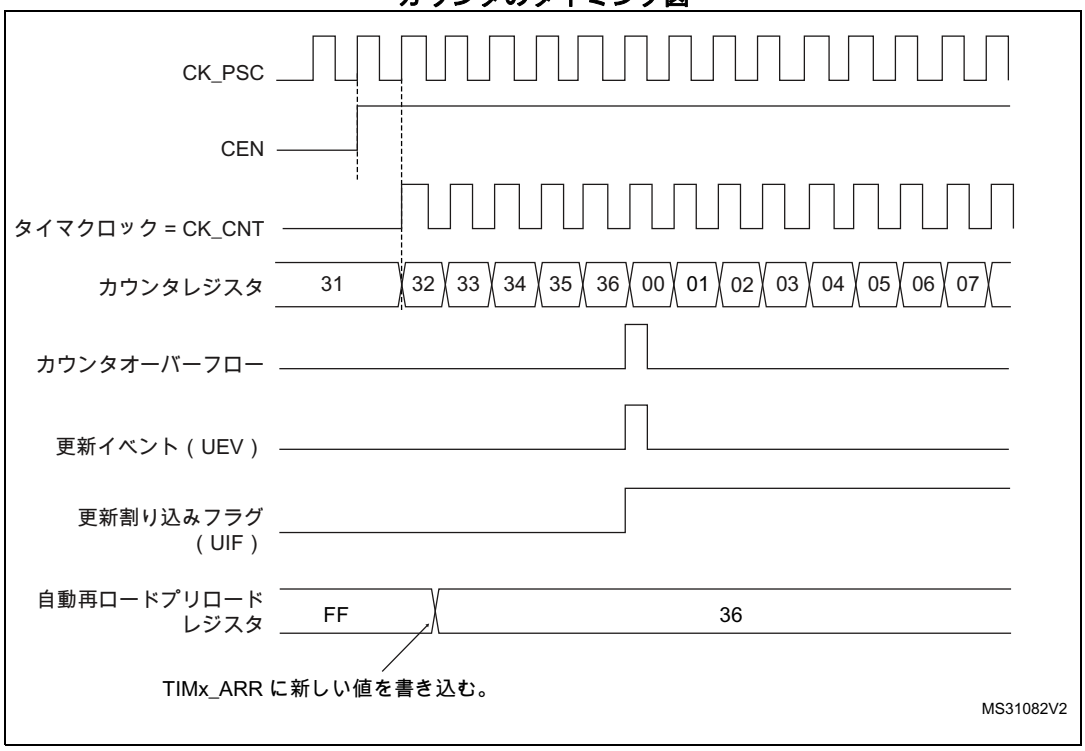
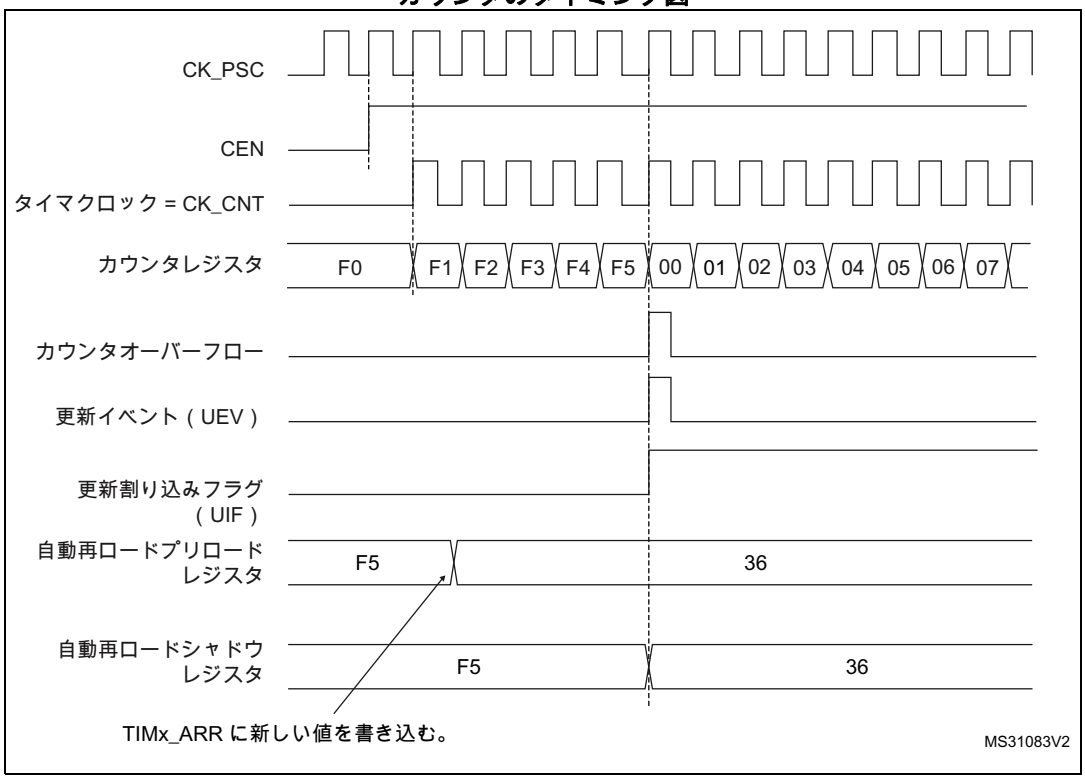


図 455. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



40.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

- 内部クロック（CK_INT）
- 外部クロックモード 1（TIM12 用）：外部入力ピン（TIx）
- 内部トリガ入力（ITRx）（TIM12 用）：他のタイマからトリガ出力を接続。たとえば、他のタイマを TIM12 のプリスケーラとして設定できます。詳細については、*セクション：タイマを別のタイマのプリスケーラとして使用する* を参照してください。

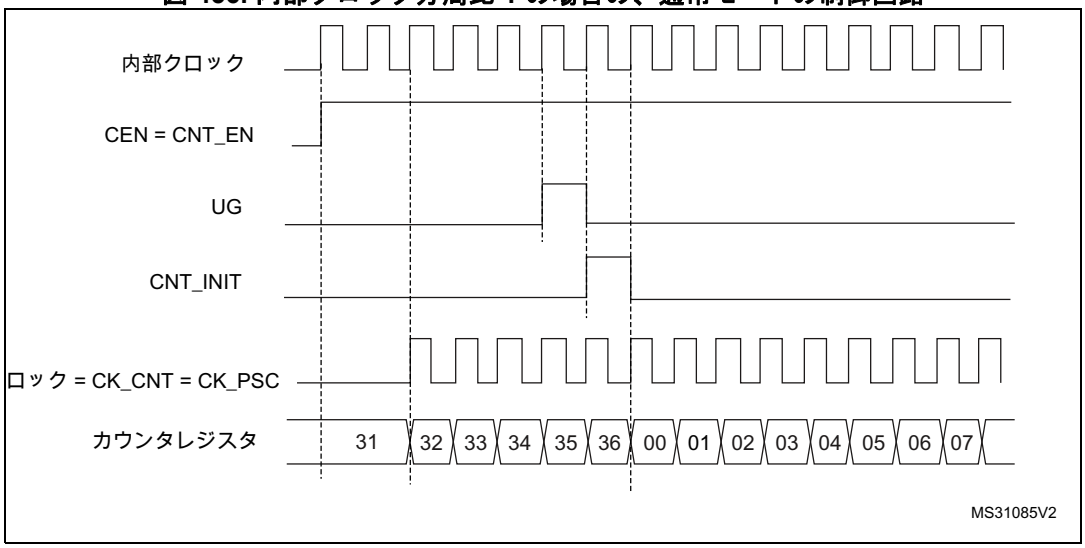
内部クロックソース（CK_INT）

内部クロックソースは、TIM13/TIM14 のデフォルトクロックソースです。

TIM12 では、スレーブモードコントローラが無効の場合（SMS = 000）、内部クロックソースが選択されます。その際、TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットが制御ビットとして使用され、ソフトウェアからのみ変更できます（クリアされたままの UG ビットは除く）。CEN ビットに 1 が設定されると直ちに、プリスケーラにはクロックとして内部クロック CK_INT が供給されます。

図 456 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

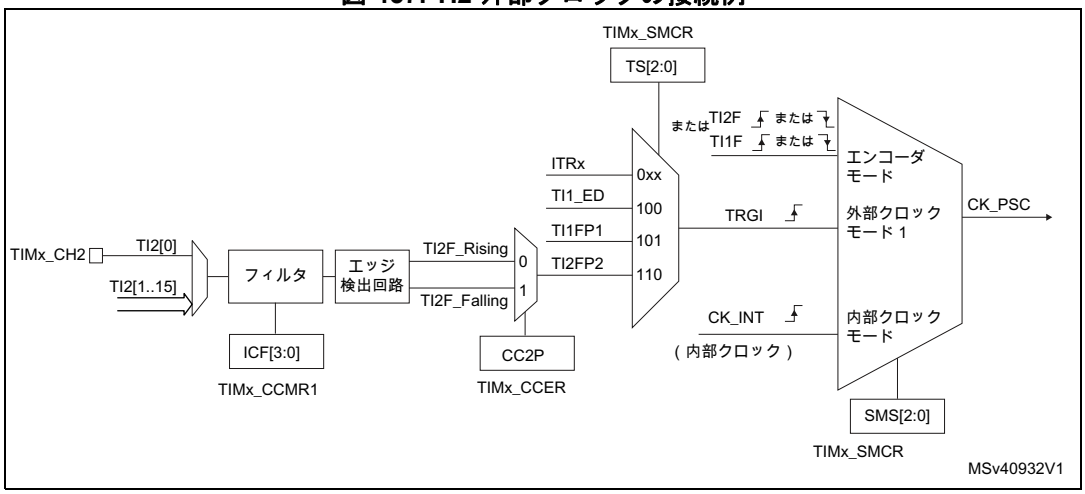
図 456. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1 （TIM12）

このモードは TIMx_SMCR レジスタの SMS = 111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 457. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

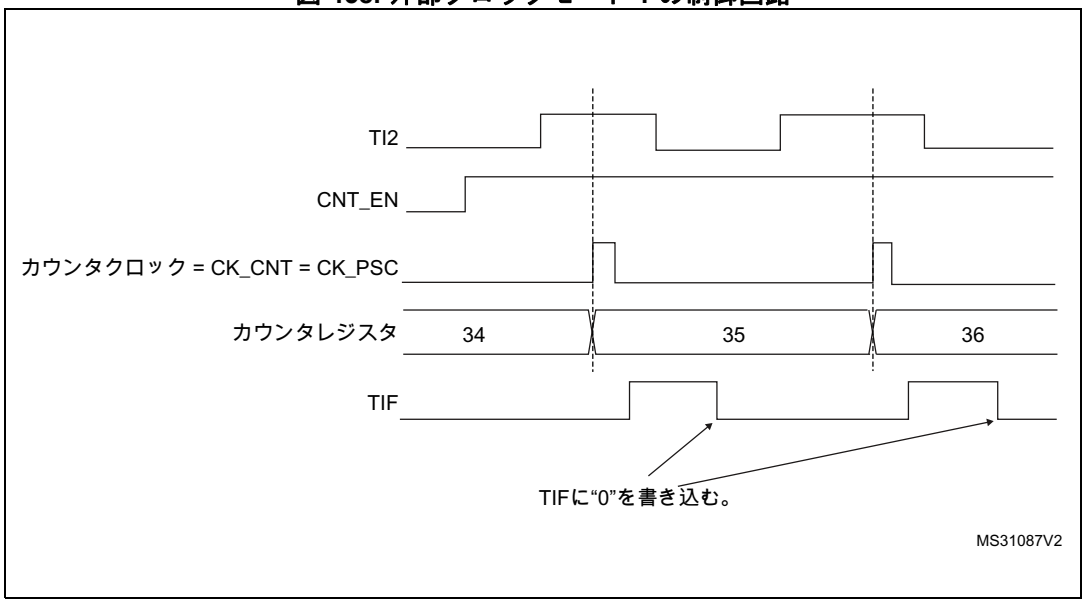
1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F = 0000 にしておきます）。
4. TIMx_CCER レジスタの CC2P ビットと CC2NP ビットに”0”を書き込むことによって、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタの SMS ビットに“111”を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタの TS ビットに“110”を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを有効にします。

注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 458. 外部クロックモード 1 の制御回路



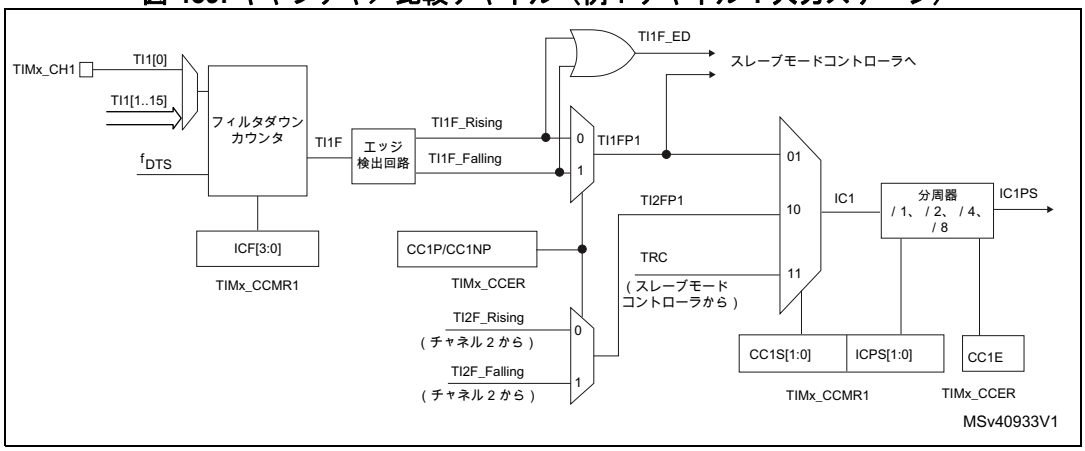
40.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 459 から *図 461* に、1 つのキャプチャ／比較 チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 459. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。OCxRef はアクティブハイです。信号の極性は最終出力に影響を与えます。

図 460. キャプチャ／比較チャネル 1 メイン回路

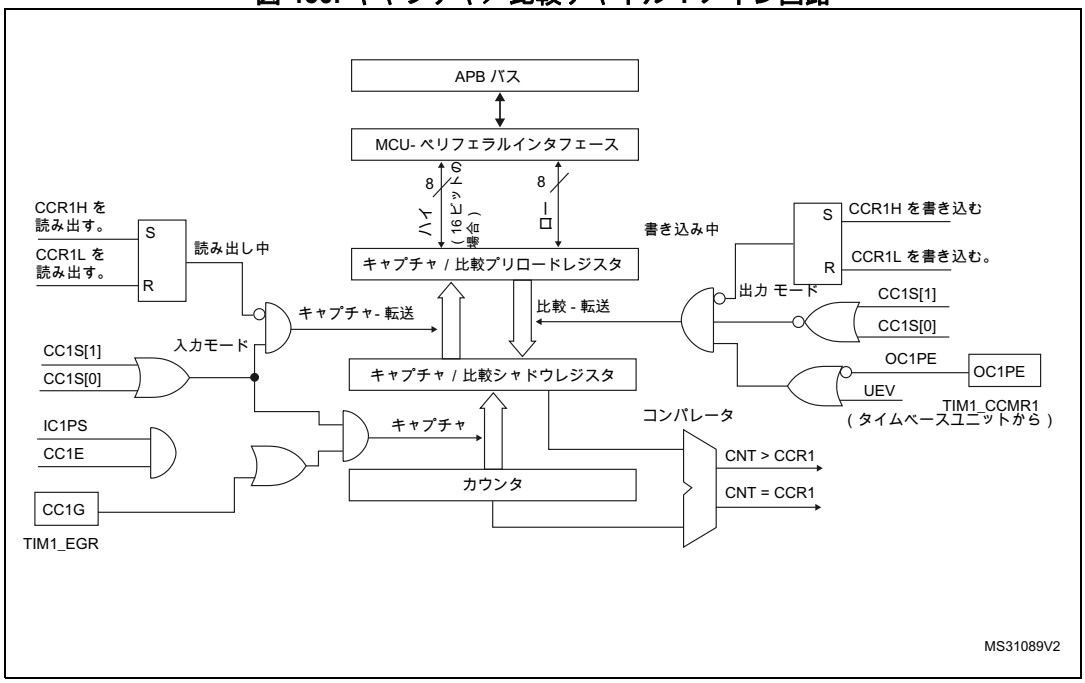
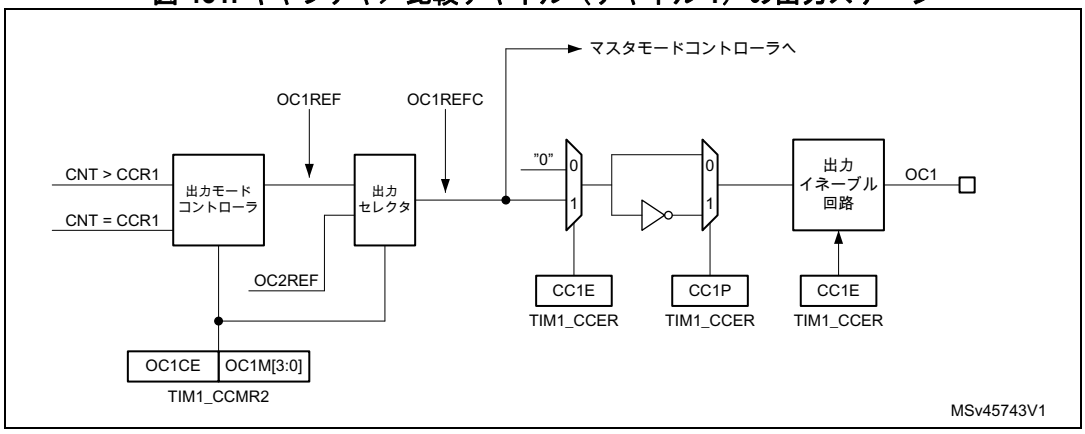


図 461. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

40.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力にリンクされていなければならないので、TIMx_CCMR1 レジスタの CC1S ビットに “01”を書き込みます。CC1S の値が“00”から変化するとすぐに、チャネルは入力モードに設定され、TIMx_CCR1 レジスタは読み出し専用になります。
3. タイマに接続する信号に対して必要とする入力フィルタ時間を設定します（入力が TIx 入力の内の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットを設定して行います）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。次に、TIMx_CCMR1 レジスタの IC1F ビットに“0011”を書き込みます。

4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“00”を設定します（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要に応じて、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***TIMx_EGR レジスタの対応する CCxG ビットをセットすることで、IC 割り込みリクエストをソフトウェアで発生させることができます。***

40.3.6 PWM 入力モード（TIM12 の場合のみ）

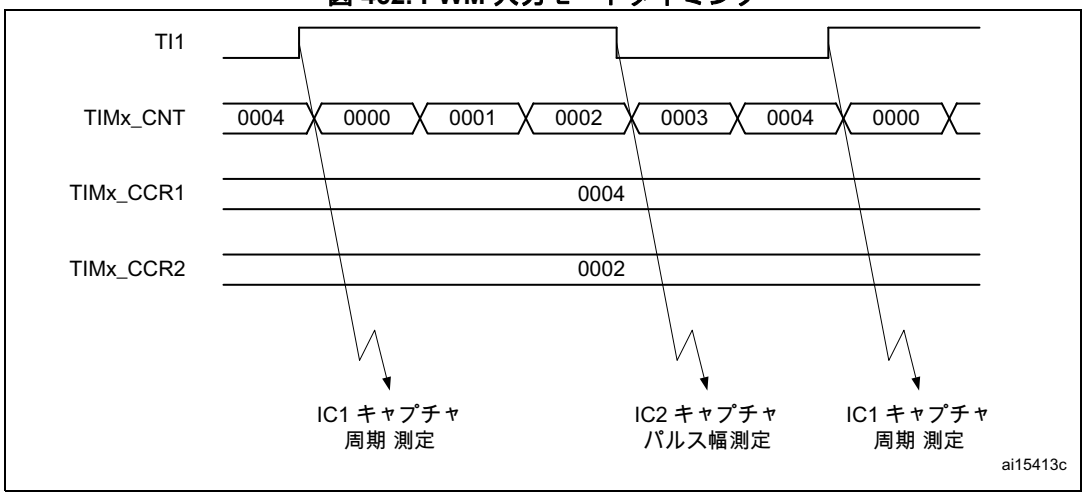
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“00”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに“11”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに“00101”を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに“100”を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 462. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

40.3.7 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット = 00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF、次に OCx）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに“0101” を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP = 0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“0100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込みリクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

40.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

1. 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット)によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011)。
2. 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット)。
3. 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

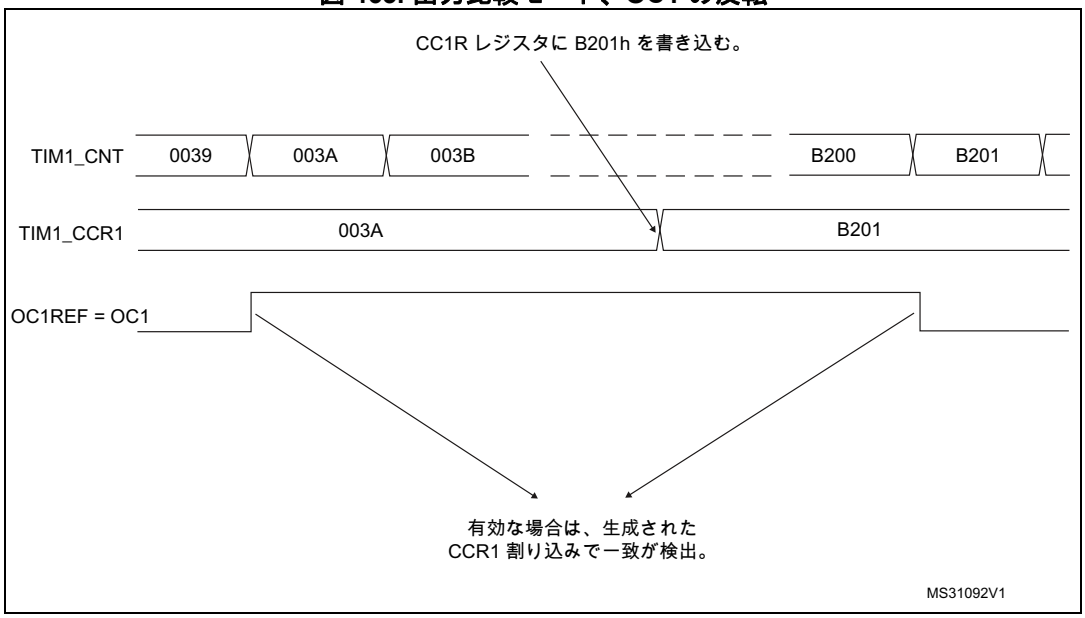
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード)。

手順：

1. カウンタクロックを選択します（内部、外部、プリスケーラ)。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに”0011”を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに”0”を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに”0”を書き込みます。
   - 出力を有効にするには、CCxE ビットに”1”を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0)。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 463* に示します。

図 463. 出力比較モード、OC1 の反転



40.3.9 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWMモード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

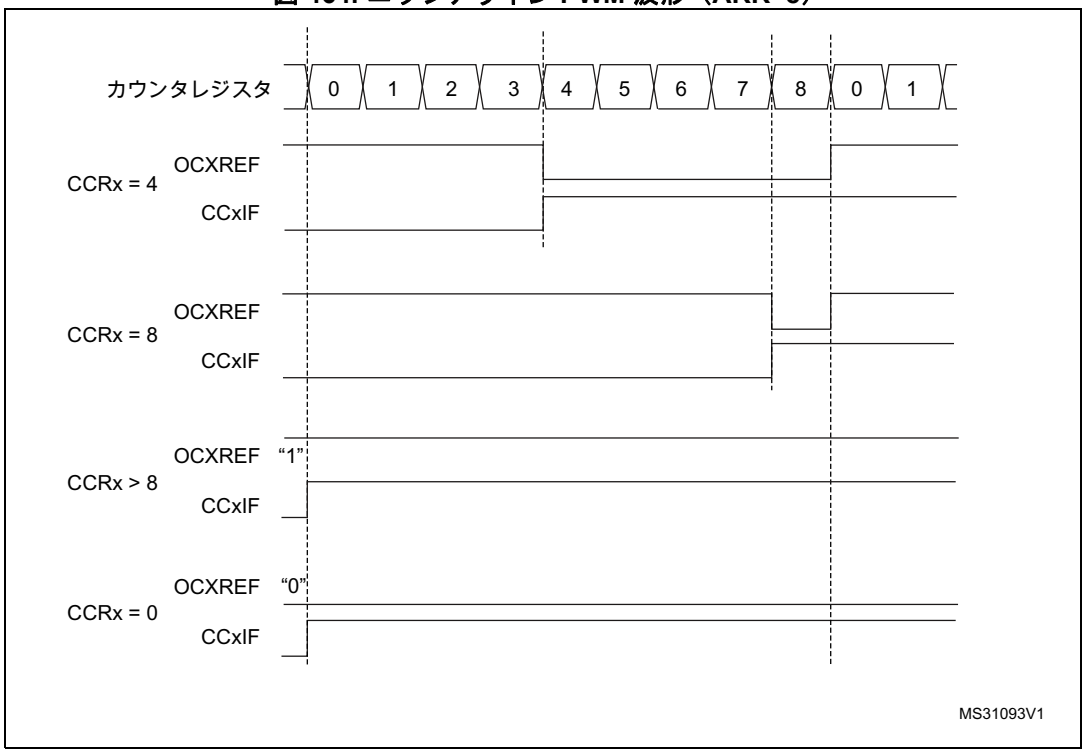
OCx の極性は、TIMx_CCER レジスタの CCxP ビットを使用して、ソフトウェアでプログラムできます。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効化されます。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます。

カウンタはカウントアップしているので、タイマはエッジアラインモードでのみ PWM を生成できます。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 464* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 464. エッジアライン PWM 波形（ARR=8）



40.3.10 組み合わせ PWM モード（TIM12 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

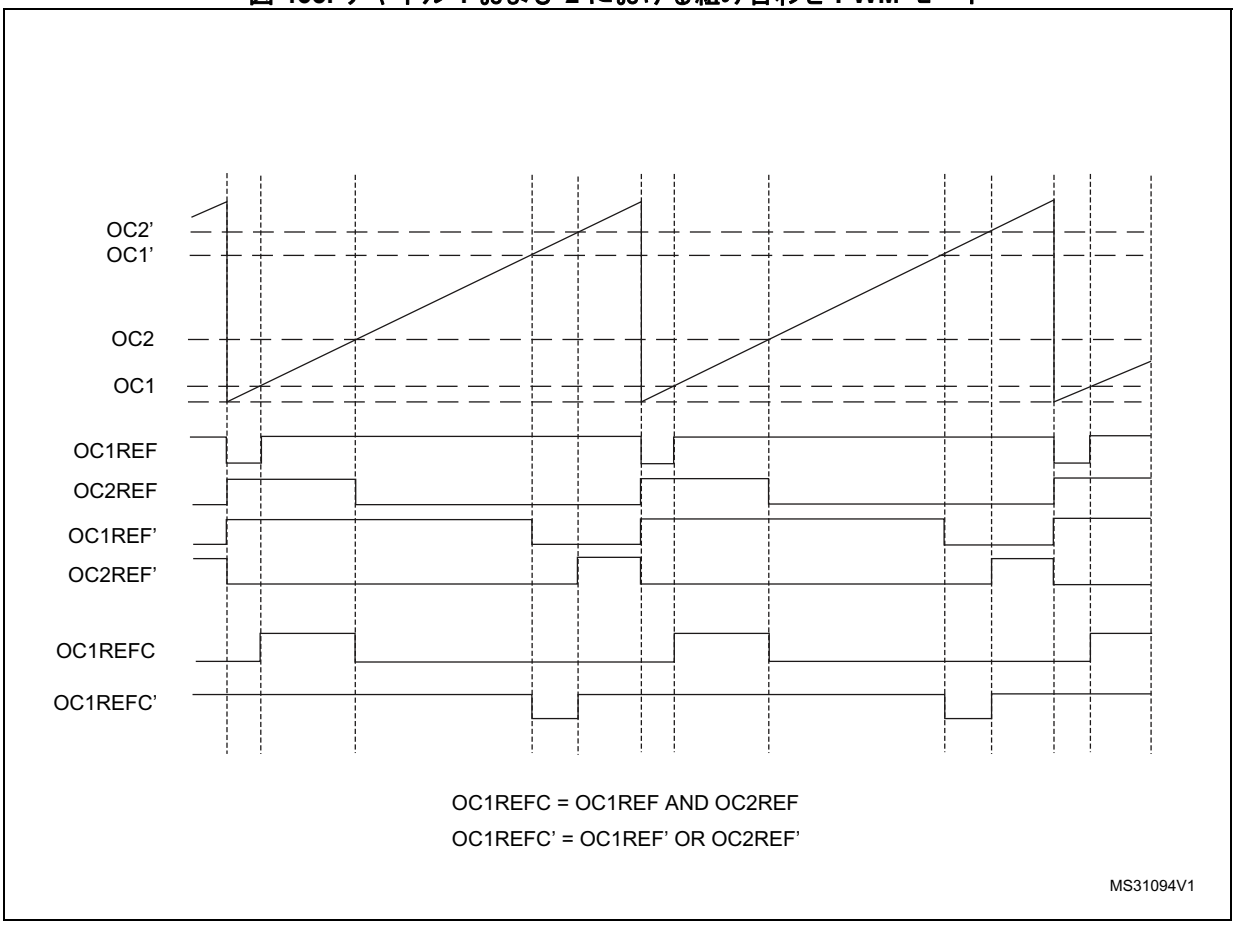
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 465 は、次の設定で取得可能な組み合わせ PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 465. チャネル 1 および 2 における組み合わせ PWM モード



40.3.11 ワンパルスモード

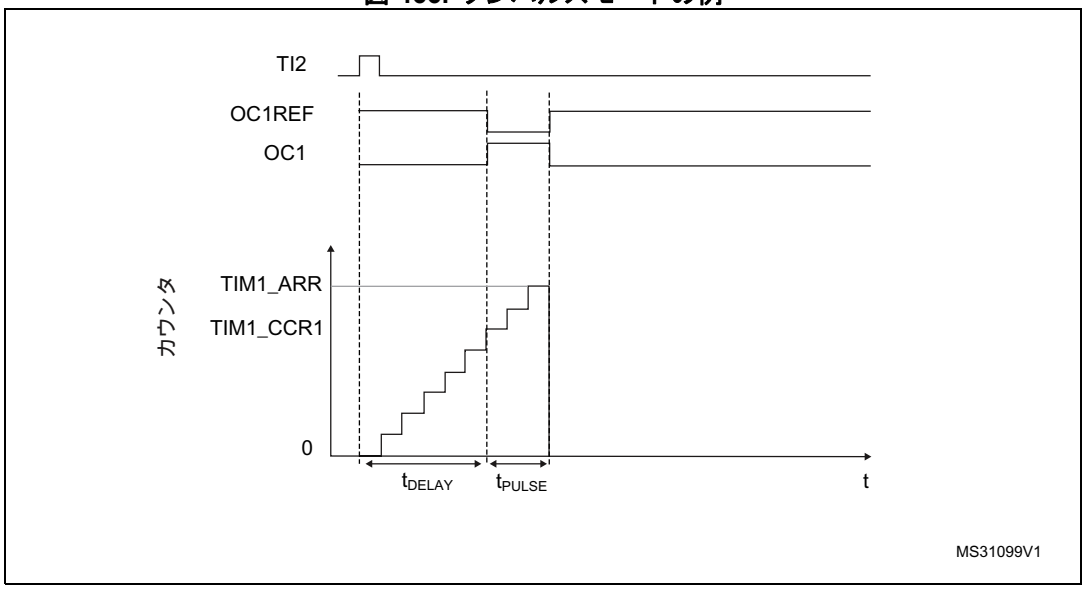
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

CNT < CCRx≤ ARR（特に、0 < CCRx）

図 466. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M ビットに“0111”を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに“1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

40.3.12 再トリガ可能なワンパルスモード（OPM）（TIM12 のみ）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、*セクション 40.3.11：ワンパルスモード*で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

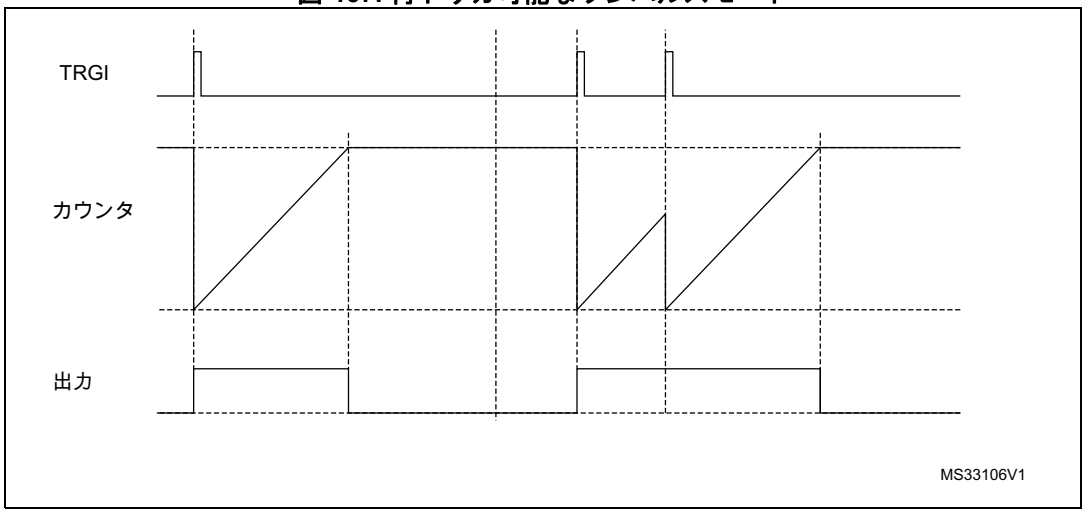
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： ***OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 467. 再トリガ可能なワンパルスモード



40.3.13 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読み出し）と割り込み（更新の割り込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

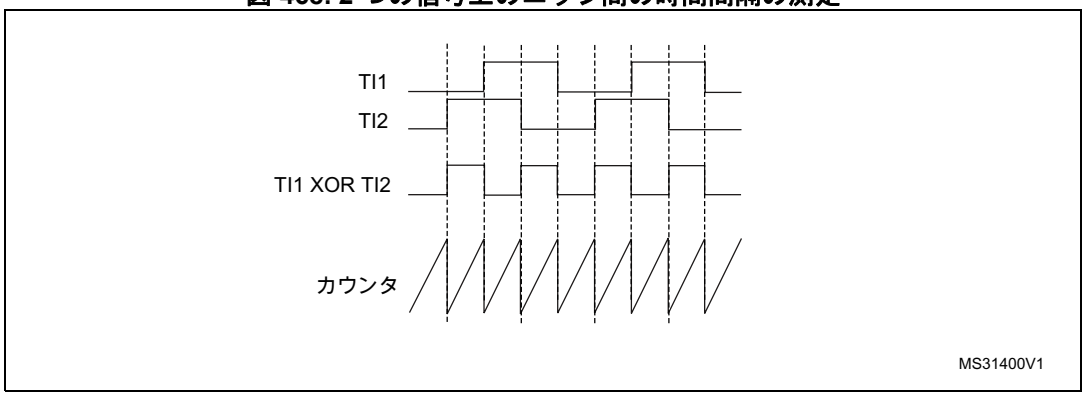
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

40.3.14 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン TIMx_CH1 および TIMx_CH2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の*図 468* に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 468. 2 つの信号上のエッジ間の時間間隔の測定



40.3.15 TIM12 外部トリガ同期

TIM12 タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

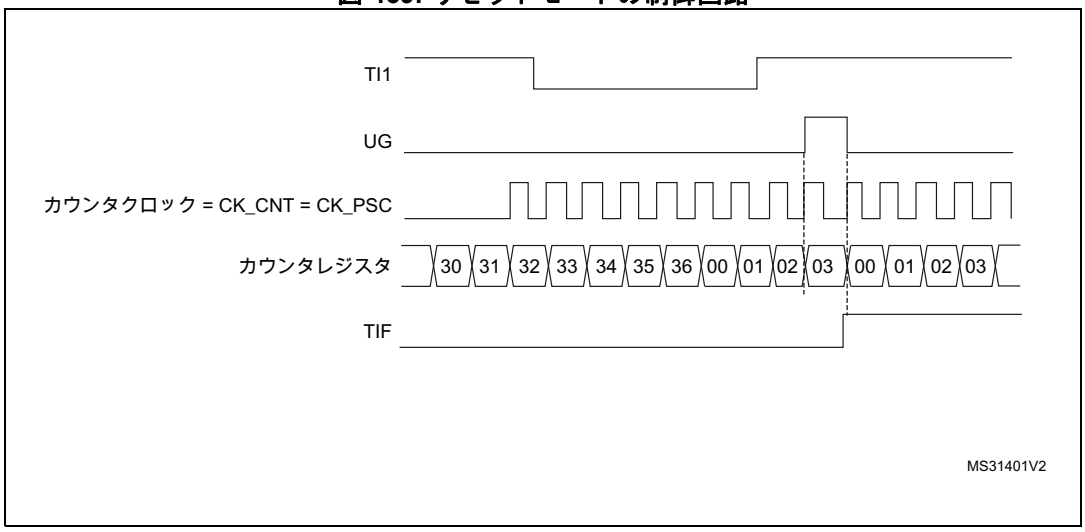
1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F = 0000 のままにしておく）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S = 01）。TIMx_CCER レジスタの CC1P と CC1NP に“00”を書き込んで、極性を有効にします（その後、立ち上がりエッジのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“100”を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=“00101” を書き込むことによって、入力ソースとして TI1 を選択します。

3. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、（TIMx_DIER レジスタの TIE ビットが）有効な場合は割り込みリクエストを送信できます。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 469. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

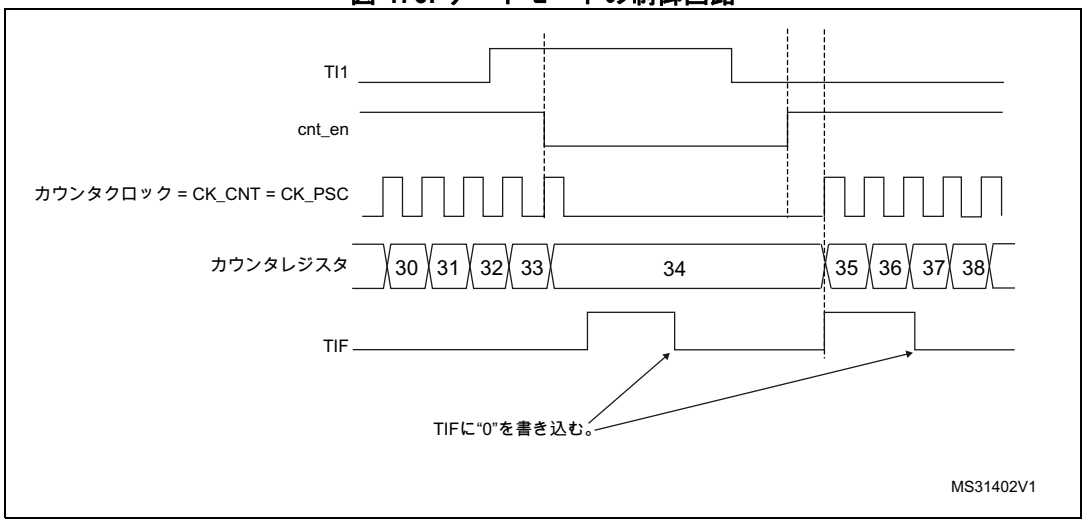
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F = 0000 のままにしておく）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S = 01）。TIMx_CCERレジスタの CC1P ビットに“1”、CC1NP ビットに”0”を書き込んで、極性を有効にします（その後、ローレベルのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“101”を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=“00101” を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタの CEN ビットに“1”を書き込んで、カウンタを有効にします（ゲートモードでは、CEN = 0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 470. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

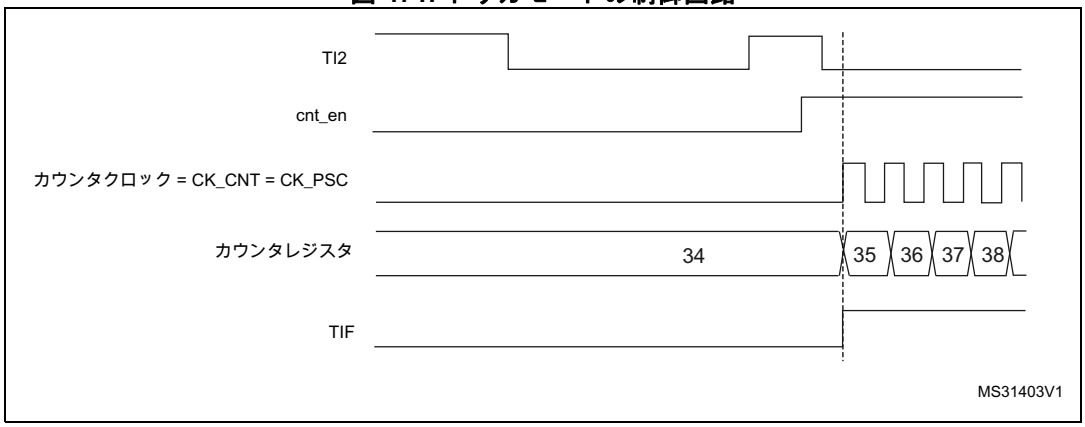
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F = 0000 のままにしておく）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S = 01）。TIMx_CCERレジスタの CC2P ビットに“1”、CC2NP ビットに“0”を書き込んで、極性を有効にします（その後、ローレベルのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“110”を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=“00110” を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 471. トリガモードの制御回路



40.3.16 スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

40.3.17 タイマの同期（TIM12）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。詳細については、*セクション 39.3.19：タイマの同期*を参照してください。

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

40.3.18 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 with FPU コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、*セクション 60.5.8：マイクロコントローラデバッグユニット（DBGMCU）*を参照してください。

40.4 TIM12 レジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位で書き込む必要があります。読み出しアクセスは、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）単位で行うことができます。

40.4.1 TIM12 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（TIx）によって使用されるサンプリングクロックとの間の分周比を示します。

00: $t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2 \times t_{CK_INT}$
10：$t_{DTS} = 4 \times t_{CK_INT}$
11：予約済み

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフローのみが更新割り込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、更新イベント（UEV）生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。UEV は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー

– UG ビットのセット

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。UEV は生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。カウンタとプリスケーラは、UG ビットがセットされた場合に再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

注：　外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。

ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

40.4.2 TIM12 スレーブモード制御レジスタ（TIMx_SMCR）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
								rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TS[4:3]**：トリガ選択 - ビット 4:3

TS[4:0] 説明を参照 - ビット 6:4

ビット 20:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3

SMS 説明を参照 - ビット 2:0

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ／スレーブモード

0：影響なし。

1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期する場合に役立ちます。

ビット 6:4 **TS[4:0]**：トリガ選択

この TS[4:0] ビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：内部トリガ 2（ITR2）

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

その他：予約済み

各タイマにとっての ITRx の意味の詳細については、*表 320：1664 ページのTIMx 内部トリガ接続* を参照してください。

注： ***遷移時の誤ったエッジ検出を避けるために、これらのビットは必ず使用されていないとき（SMS = 000 のときなど）に変更してください。***

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済み

0010：予約済み

0011：予約済み

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

その他：予約済み

注： ***トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。***

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

表 320. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = ‘00000’)	ITR1 (TS = ‘00001’)	ITR2 (TS = ‘00010’)	ITR3 (TS = ‘00011’)
TIM12	TIM4	TIM5	TIM13 OC1	TIM14 OC1

40.4.3 TIM12 割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE	Res.	Res.	Res.	CC2IE	CC1IE	UIE
									rw				rw	rw	rw

ビット 15:7　予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5:3　予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

40.4.4 TIM12 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	Res.	TIF	Res.	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0			rc_w0				rc_w0	rc_w0	rc_w0

ビット 15:11　予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7　予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割り込みが保留中です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：

このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（選択された極性に一致するエッジが IC1 で検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– オーバーフローが発生し、かつ TIMx_CR1 レジスタで UDIS = 0 の場合。

– TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

– TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

40.4.5 TIM12 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	Res.	Res.	CC2G	CC1G	UG
									w				w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込が発生します。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
現在のカウンタ値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラのカウンタもクリアされます（分周比は変化しません）。カウンタはクリアされます。

40.4.6 TIM12 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のすべてのビットは、入力モードと出力モードで機能が異なります。特定のビットについて、OCxx は、チャネルが出力モードに設定されているときの機能を記述し、ICxx は、チャネルが入力モードに設定されているときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済み。常に 0 として読み出されます。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3

ビット 14:12 の OC2M 説明を参照

ビット 23:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3
ビット 6:4 の OC1M 説明を参照

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード
ビットの説明は OC1M[3:0] を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E = 0）のときにのみ書き込み可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **OC1M[3:0]**：出力比較 1 モード（OC1M[3] はビット 16 を参照）
これらのビットは、OC1 の元となる出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 のアクティブレベルは CC1P に依存します。
0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。
0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ／比較レジスタ 1（TIMx_CCR1）に一致したとき、強制的にハイになります。
0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ／比較レジスタ 1（TIMx_CCR1）に一致したとき、強制的にローになります。
0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。
0100：強制インアクティブレベル - OC1REF は強制的にローになります。
0101：強制アクティブレベル - OC1REF は強制的にハイになります。
0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。
0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。
1000：再トリガ可能な OPM モード 1 - TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。
1001：再トリガ可能な OPM モード 2 - TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。
1010：予約済み。
1011：予約済み。
1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。
1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。
1110：予約済み。
1111：予約済み

注： PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 は、トリガがオンのときでも、カウンタと CCR1 の値に依存して、通常どおりに動作します。トリガ入力にエッジが発生してから CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E = 0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E = 0）のときにのみ書き込み可能です。

ビット 7:4 **IC1F**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。
0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E = 0）のときにのみ書き込み可能です。

40.4.7 TIM12 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
								rw		rw	rw	rw		rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ／比較 2 出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性
CC1 チャネルが出力として設定されている場合：CC1NP はクリア状態に維持する必要があります。
CC1 チャネルが入力として設定されている場合：CC1NP ビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：

CC1P ビットおよび CC1NP ビットは、キャプチャ動作での TI1FP1 の極性を選択します。

00：非反転/立ち上がりエッジ

回路は TIxFP1 の立ち上がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ、エンコーダモード）。

01：反転／立ち下がりエッジ

回路は TIxFP1 の立ち下がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されます（ゲートモードでのトリガ、エンコーダモード）。

10：予約済み。この設定は使用しないでください。

11：非反転／両エッジ

回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ）。この設定をエンコーダモードに使用することはできません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。

1：オン - OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：

このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 321. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx = 0、OCx_EN = 0）
1	OCx = OCxREF + 極性、OCx_EN = 1

注： ***標準 OCx チャネルに接続されている外部入出力ピンの状態は、OCx チャネルの状態と、GPIO レジスタに依存します。***

40.4.8 TIM12 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

40.4.9 TIM12 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブなプリスケーラレジスタにロードされる値を含みます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

40.4.10 TIM12自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。
ARP の更新と動作の詳細については、*セクション 40.3.1： 1640 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

40.4.11 TIM12 キャプチャ／比較レジスタ 1 （TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

40.4.12 TIM12 キャプチャ／比較モードレジスタ 2（TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：

CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

40.4.13 TIM12 タイマ入力選択レジスタ（TIM12_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw								rw			

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM12_CH2 入力
その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM12_CH1 入力
その他：予約済み

40.4.14 TIM12 レジスタマップ

TIM12 のレジスタは、次のように 16 ビットのアドレス可能レジスタとしてマップされます：

表 322. TIM12レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x08	TIMx_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3] Res.		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0									0	0	0	0		0	0	0
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE	Res.	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																										0				0	0	0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	Res.	TIF	Res.	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0			0				0	0	0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	Res.	Res.	CC2G	CC1G	UG
	リセット値																										0				0	0	0
0x18	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	TIMx_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																									0		0	0	0		0	0
0x24	TIMx_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	予約済み																ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 322. TIM12レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIMx_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C～0x64	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x68	TIM12_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値																					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

40.5 TIM13/TIM14 レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位で書き込む必要があります。読み出しアクセスは、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）単位で行うことができます。

40.5.1 TIM13/TIM14 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFRE MAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（TIx）によって使用されるサンプリングクロックとの間の分周比を示します。

00: $t_{DTS} = t_{CK_INT}$

01：$t_{DTS} = 2 \times t_{CK_INT}$

10：$t_{DTS} = 4 \times t_{CK_INT}$

11：予約済み

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、更新割り込み（UEV）ソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが UEV を発生します（有効な場合）。

– カウンタオーバーフロー

– UG ビットのセット

1：カウンタオーバーフローのみが UEV を発生します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、更新割り込み（UEV）イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。UEV は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー

– UG ビットのセット

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。UEV は生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。カウンタとプリスケーラは、UG ビットがセットされた場合に再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。

ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

40.5.2 TIM13/TIM14 割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IE	UIE
														rw	rw

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル

0：CC1 割り込みは無効です。

1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル

0：更新割り込みは無効です。

1：更新割り込みは有効です。

40.5.3 TIM13/TIM14 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IF	UIF
						rc_w0								rc_w0	rc_w0

ビット 15:10　予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバーキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:2　予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：

このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（選択された極性に一致するエッジが IC1 で検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– オーバーフローが発生し、かつ TIMx_CR1 レジスタで UDIS = 0 の場合。

– TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

40.5.4 TIM13/TIM14 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1G	UG
														w	w

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。カウンタはクリアされます。

40.5.5 TIM13/TIM14 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
															Res.
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F[3:0]				IC1PSC[1:0]			
								rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3

ビット 6:4 の OC1M 説明を参照

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **OC1M[3:0]**：出力比較 1 モード（OC1M[3] はビット 16 を参照）

これらのビットは、OC1 の元となる出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 のアクティブレベルは CC1P ビットに依存します。

0000：停止。出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他：予約済み

注： PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときに、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 15:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング1000：$f_{SAMPLING}=f_{DTS}/8$、N=6
0001：$f_{SAMPLING}=f_{CK_INT}$, N=21001：$f_{SAMPLING}=f_{DTS}/8$, N=8
0010：$f_{SAMPLING}=f_{CK_INT}$, N=41010：$f_{SAMPLING}=f_{DTS}/16$, N=5
0011：$f_{SAMPLING}=f_{CK_INT}$, N=81011：$f_{SAMPLING}=f_{DTS}/16$, N=6
0100：$f_{SAMPLING}=f_{DTS}/2$, N=61100：$f_{SAMPLING}=f_{DTS}/16$, N=8
0101：$f_{SAMPLING}=f_{DTS}/2$, N=81101：$f_{SAMPLING}=f_{DTS}/32$, N=5
0110：$f_{SAMPLING}=f_{DTS}/4$, N=61110：$f_{SAMPLING}=f_{DTS}/32$, N=6
0111：$f_{SAMPLING}=f_{DTS}/4$, N=81111：$f_{SAMPLING}=f_{DTS}/32$, N=8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：予約済み
11：予約済み

注：　CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

40.5.6 TIM13/TIM14 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	Res.	CC1P	CC1E
												rw		rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：CC1NP はクリアされたままにする必要があります。

CC1 チャネルが入力として設定されている場合：CC1NP ビットは、TI1FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：

CC1P ビットおよび CC1NP ビットは、キャプチャ動作での TI1FP1 の極性を選択します。

00：非反転／立ち上がりエッジ

回路は TI1FP1 の立上がりエッジに反応し（キャプチャモード）、TI1FP1 は反転されません。

01：反転／立ち下がりエッジ

回路は TI1FP1 の立下がりエッジに反応し（キャプチャモード）、TI1FP1 は反転されます。

10：予約済み。この設定は使用しないでください。

11：非反転／両エッジ

回路は TI1FP1 の立上がりエッジに反応し（キャプチャモード）、TI1FP1 は反転されません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル。

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。

1：オン - OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：

このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 323. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx=0、OCx_EN=0）
1	OCx = OCxREF + 極性、OCx_EN = 1

注： ***標準 OCx チャネルに接続されている外部入出力ピンの状態は、OCx チャネルの状態と、GPIO レジスタに依存します。***

40.5.7 TIM13/TIM14 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

40.5.8 TIM13/TIM14 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます。

（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

40.5.9 TIM13/TIM14自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

APR の更新と動作の詳細については、*セクション 40.3.1：1640 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

40.5.10 TIM13/TIM14 キャプチャ／比較レジスタ 1（TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

40.5.11 TIM13 タイマ入力選択レジスタ（TIM13_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TISEL[3:0]			
												rw			

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM13_CH1 入力
その他：予約済み

40.5.12 TIM14 タイマ入力選択レジスタ（TIM14_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL[3:0]			
												rw			

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM14_CH1 入力
その他：予約済み

40.5.13 TIM13/TIM14 レジスタマップ

TIMx レジスタは、次の表のように、16 ビットのアドレス可能レジスタとしてマップされます。

表 324. TIM13/TIM14レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x08	TIMx_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IE	UIE
	リセット値																															0	0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0								0	0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1G	UG
	リセット値																															0	0
0x18	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値																0										0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0x1C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	TIMx_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	Res.	CC1P	CC1E
	リセット値																													0		0	0
0x24	TIMx_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]														
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]														
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]														
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 324. TIM13/TIM14レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38～0x64	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x68	TIM13_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値																													0	0	0	0
0x68	TIM14_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

41 汎用タイマ（TIM15/TIM16/TIM17）

41.1 TIM15/TIM16/TIM17 の概要

TIM15/TIM16/TIM17 タイマは、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM15/TIM16/TIM17 タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、*セクション 41.4.20：タイマ同期（TIM15）*に示すように、相互に同期させることができます。

41.2 TIM15 の主な機能

TIM15 には以下の機能があります。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- 次の機能を持つ、最大 2 つの独立チャネル。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力（チャネル 1 のみ）
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力（割り込みリクエスト）

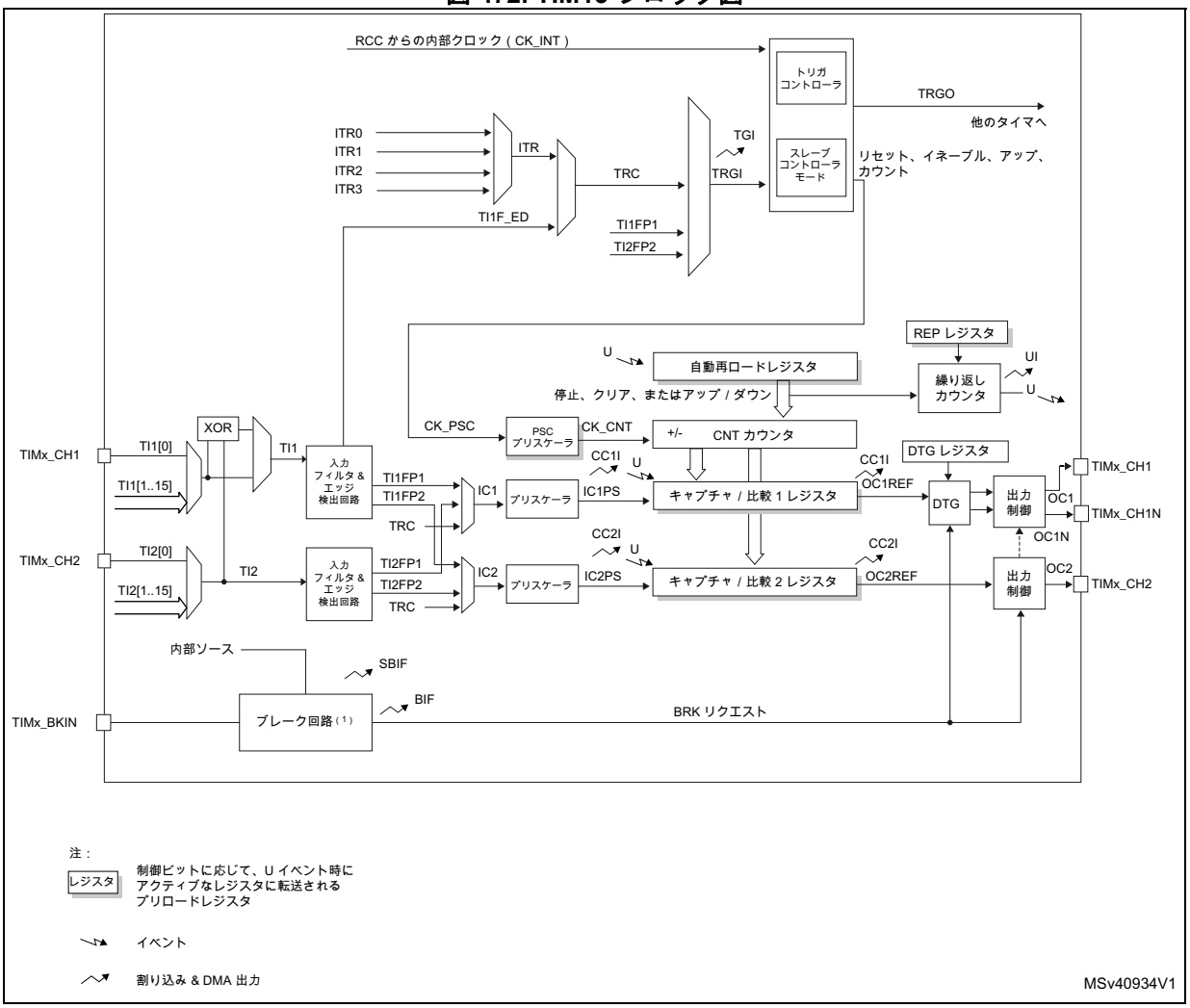
41.3 TIM16/TIM17 の主な特長

TIM16/TIM17 タイマには以下の機能があります。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- 次の機能を持つ、1 チャネルタイマ。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力

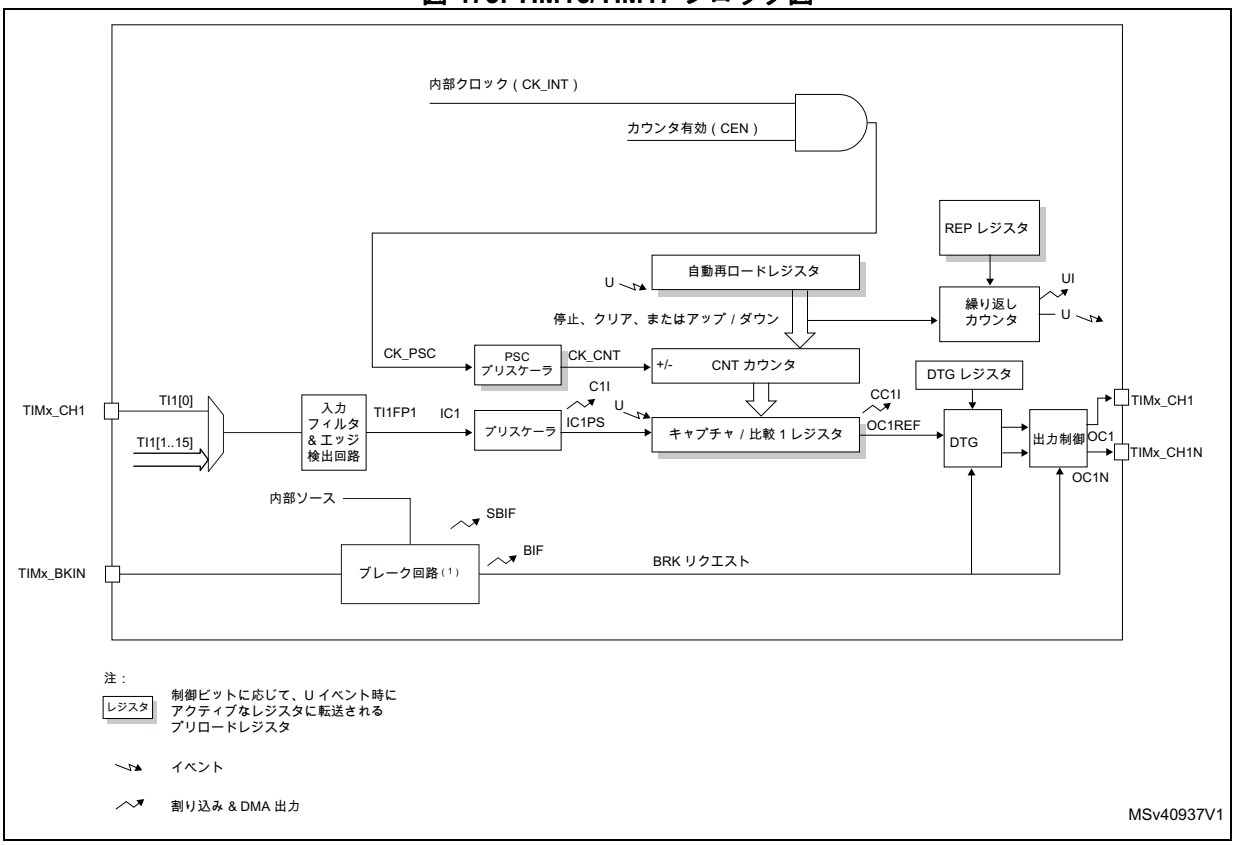
参考資料

図 472. TIM15 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、*セクション 8.5.3 : クロックセキュリティシステム（CSS）*を参照してください。
   - PVD 出力
   - SRAM パリティエラー信号
   - Cortex®-M7 with FPU LOCKUP （ハードフォルト）出力
   - COMP 出力

図 473. TIM16/TIM17 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、*セクション 8.5.3：クロックセキュリティシステム（CSS）*を参照してください。
   - PVD 出力
   - SRAM パリティエラー信号
   - Cortex®-M7 with FPU LOCKUP （ハードフォルト）出力
   - COMP 出力

41.4 TIM15/TIM16/TIM17機能詳細

41.4.1 タイムベースユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 474 と *図 475* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 474. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

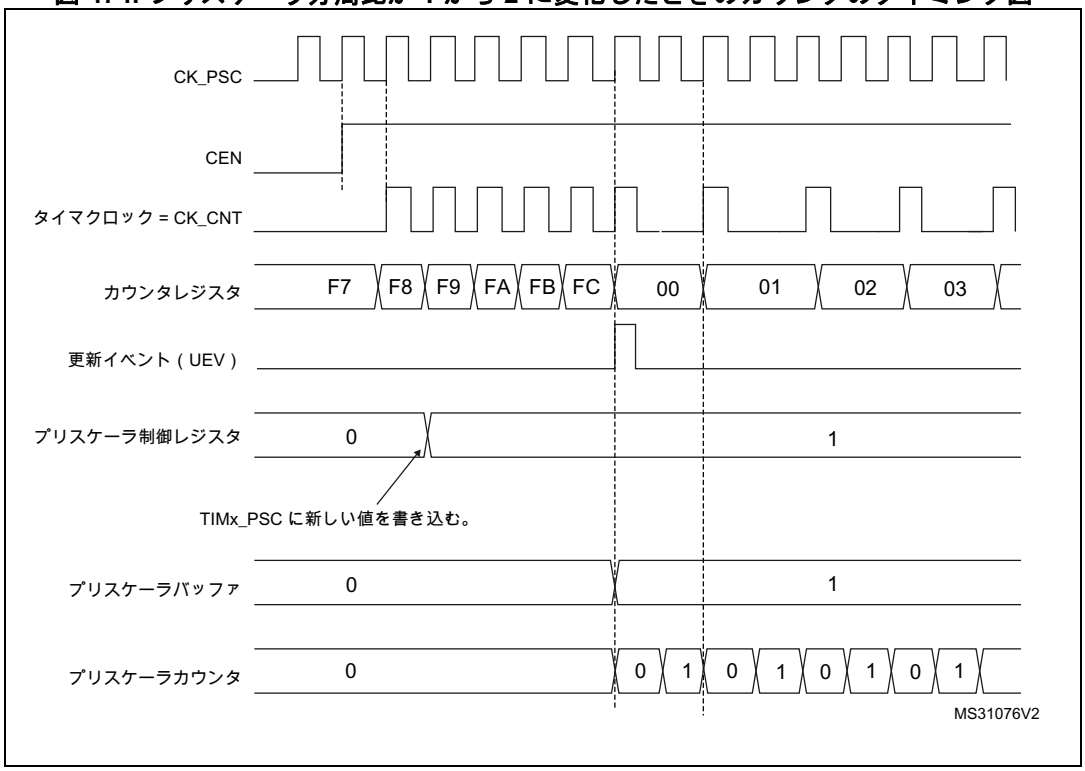
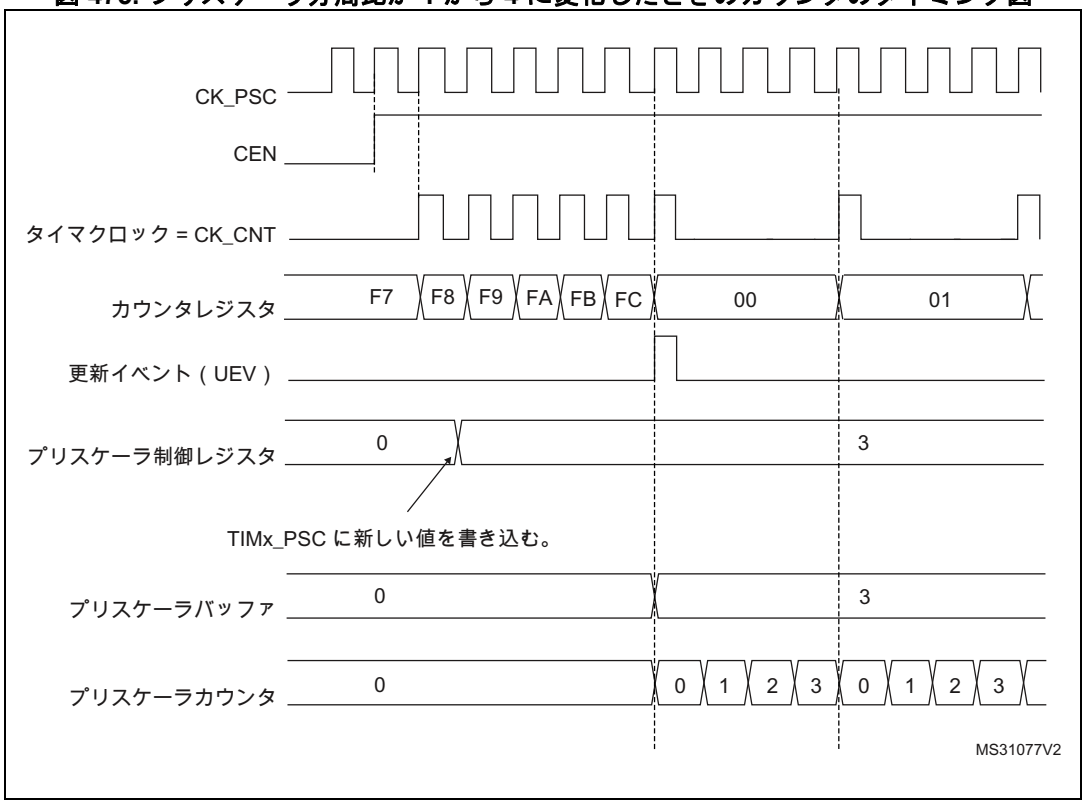


図 475. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



41.4.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 476. 内部クロック分周比が 1 の場合のカウンタのタイミング図

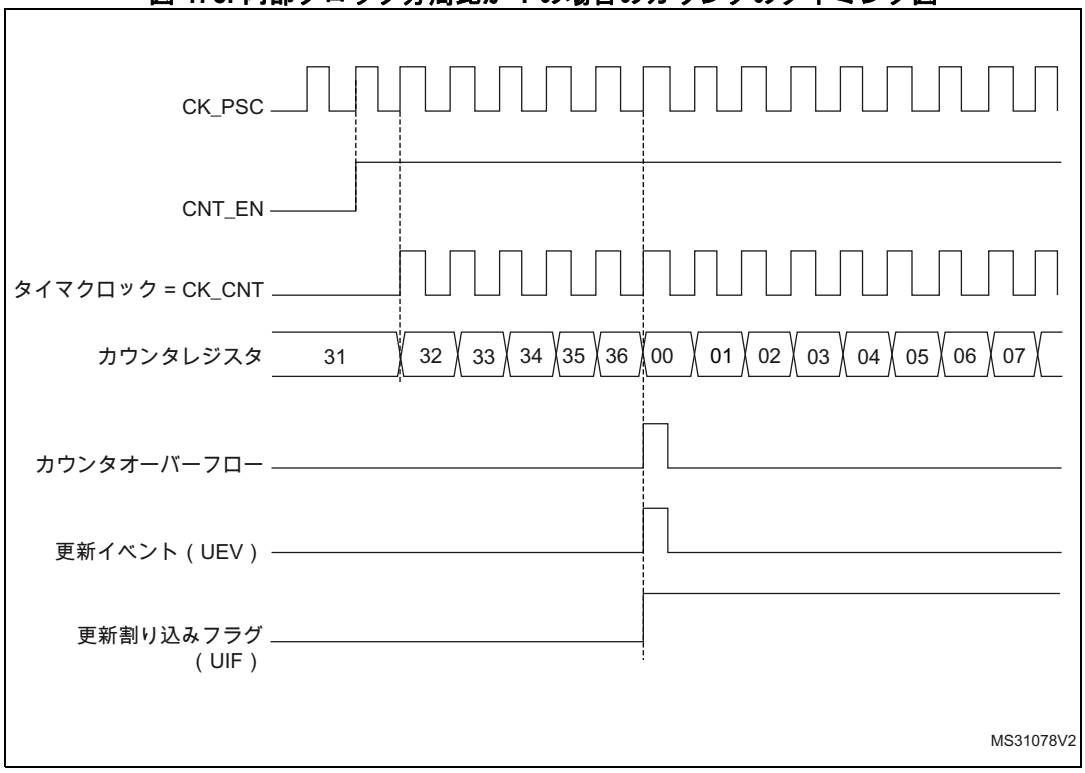


図 477. 内部クロック分周比が 2 の場合のカウンタのタイミング図

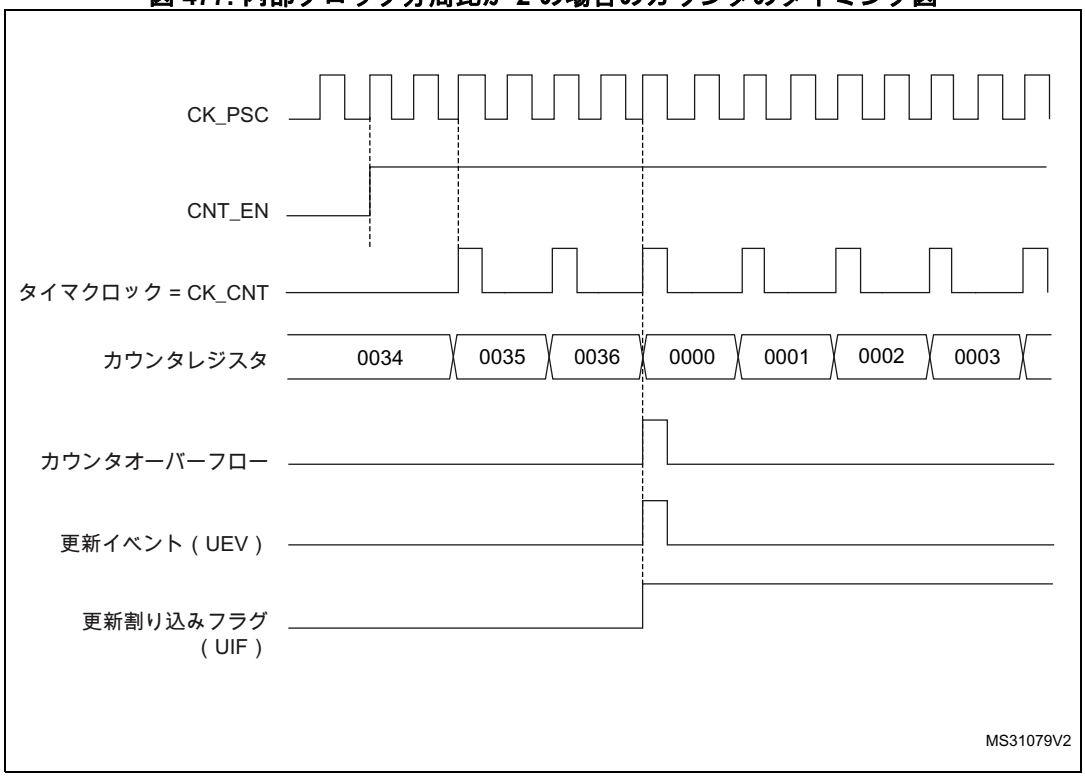


図 478. 内部クロック分周比が 4 の場合のカウンタのタイミング図

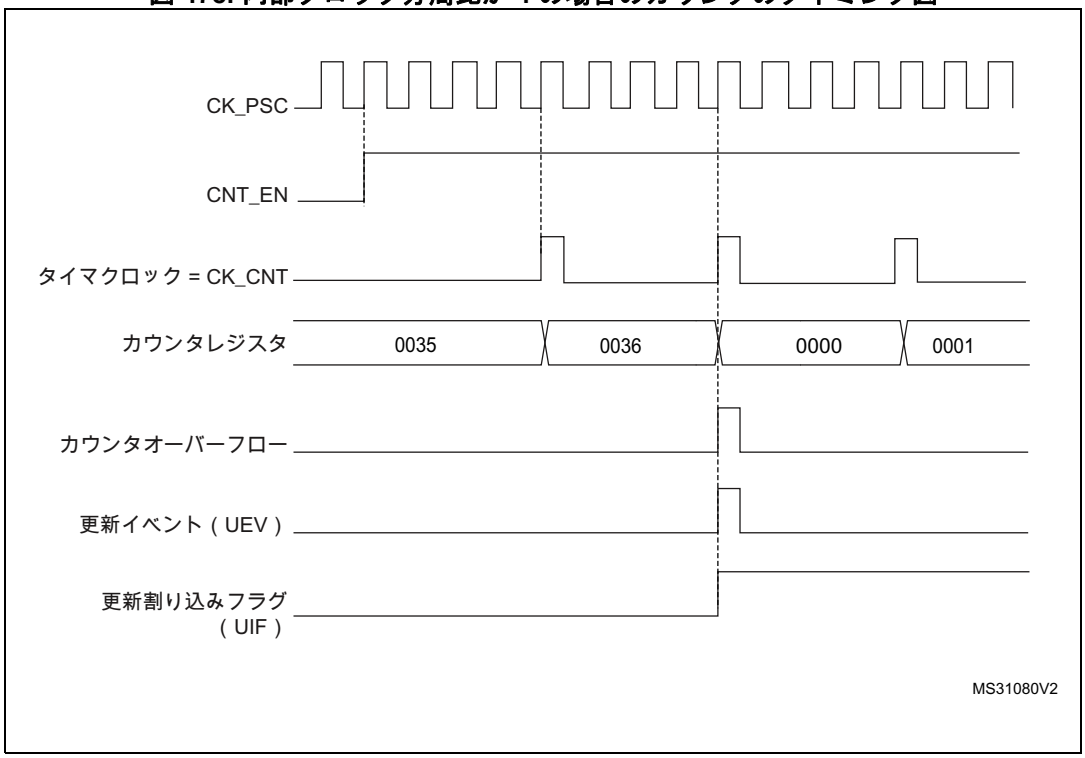


図 479. 内部クロック分周比が N の場合のカウンタのタイミング図

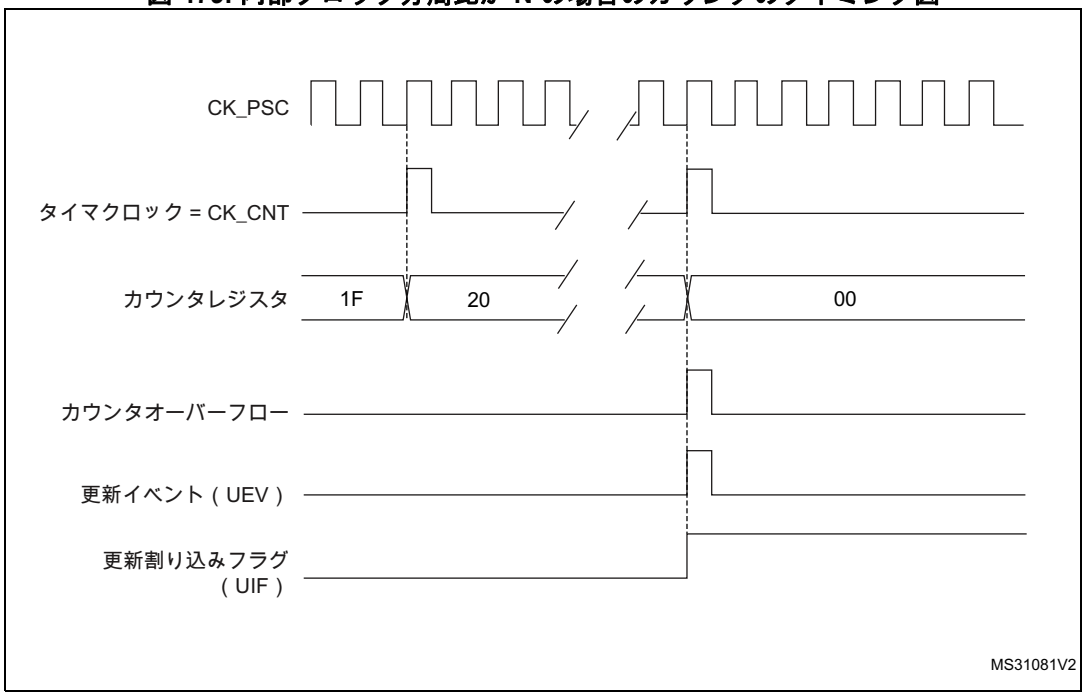


図 480. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

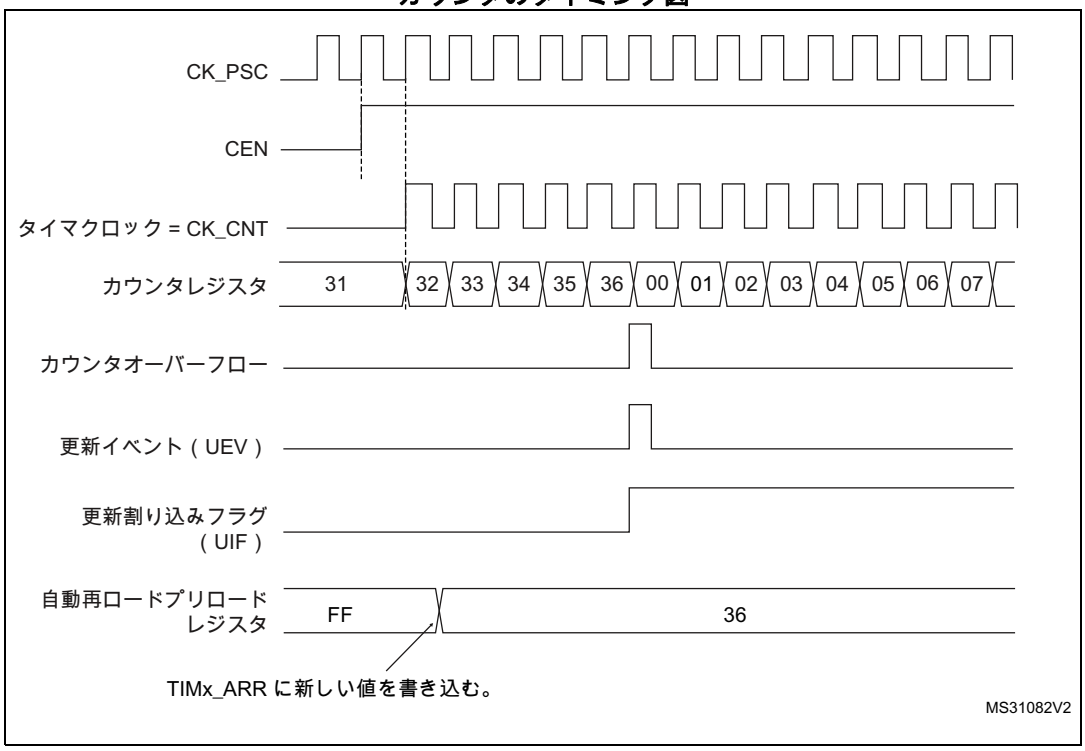
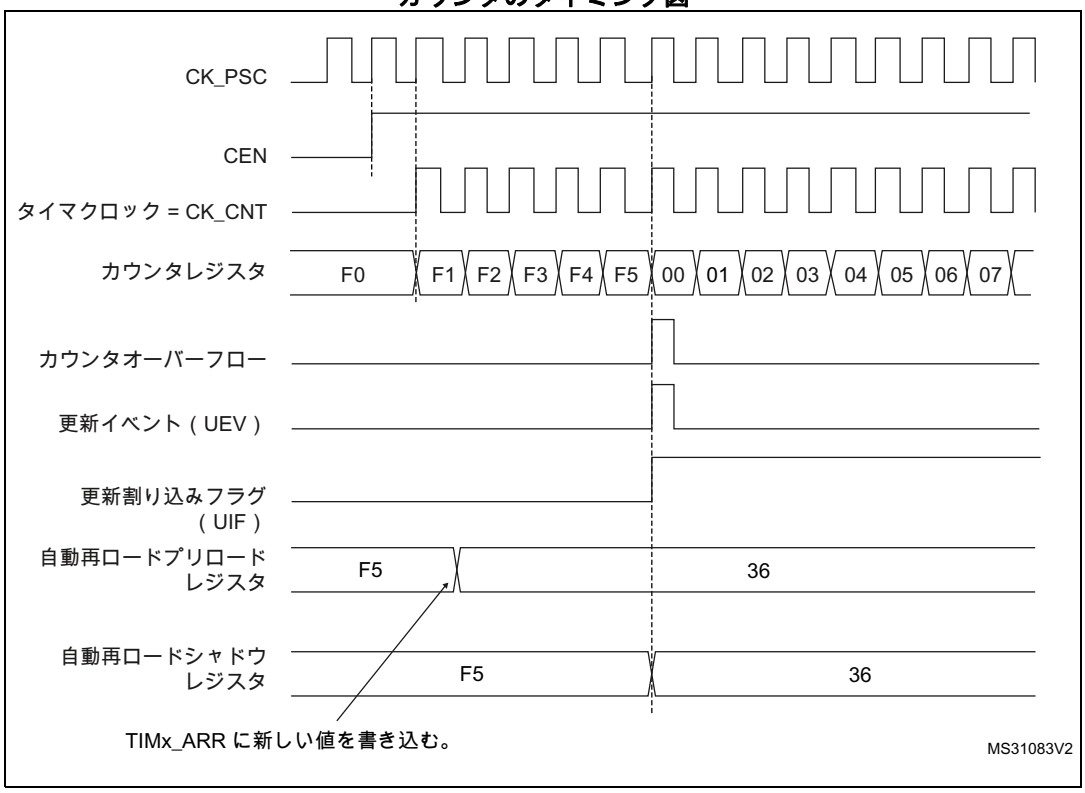


図 481. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



41.4.3 繰り返しカウンタ

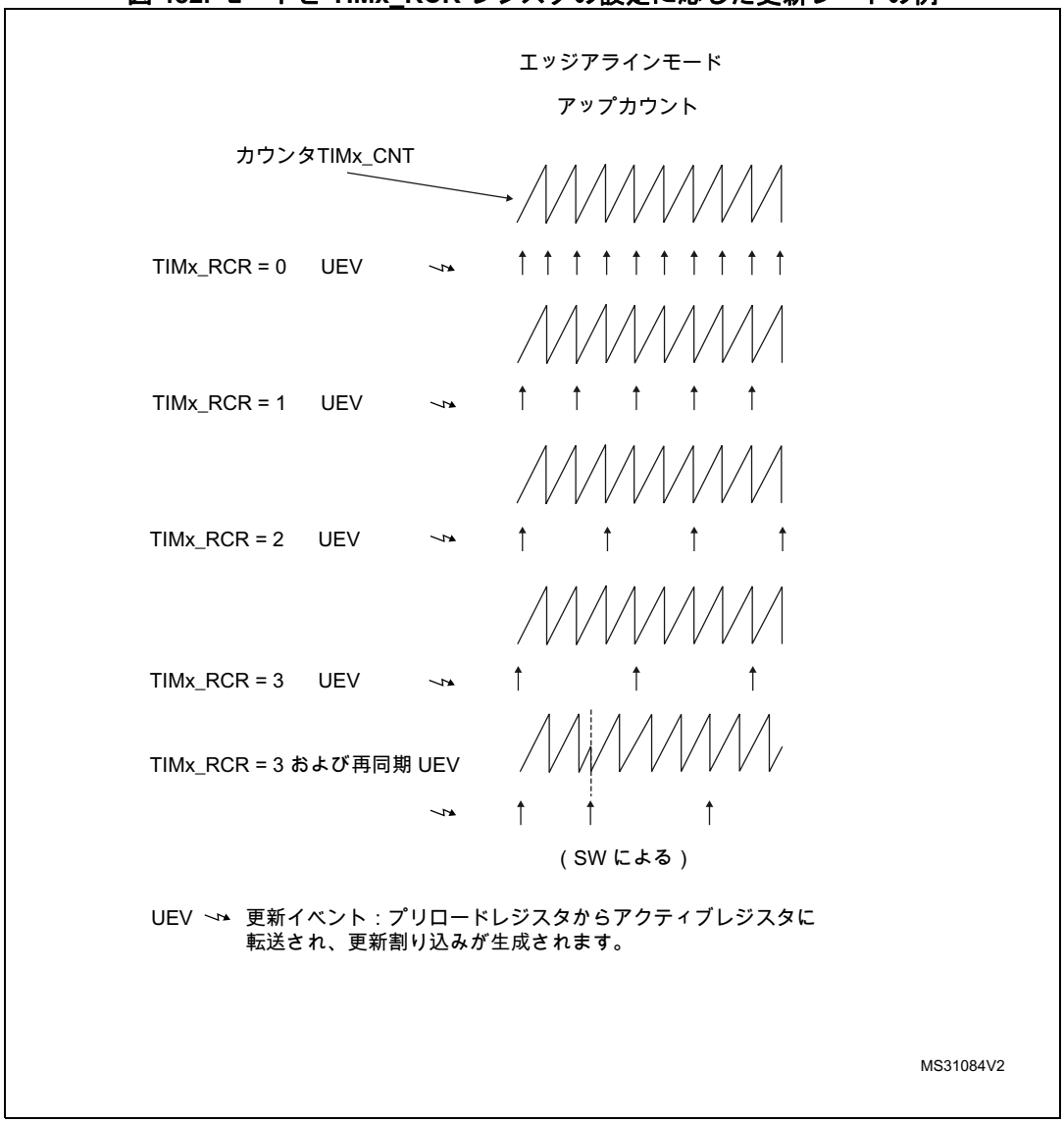
*セクション 41.4.1 : タイムベースユニット*に、カウンタオーバーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが 0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N 回目のカウンタオーバーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、カウンタオーバーフローごとにデクリメントされます。

繰り返しダウンカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（*図 482* を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

図 482. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



41.4.4 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 内部トリガ入力（ITRx）（TIM15 の場合のみ）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、TIM1 が TIM15 のプリスケーラとして機能するように設定できます。詳細については、*1602 ページのタイマを別のタイマのプリスケーラとして使用する* を参照してください。

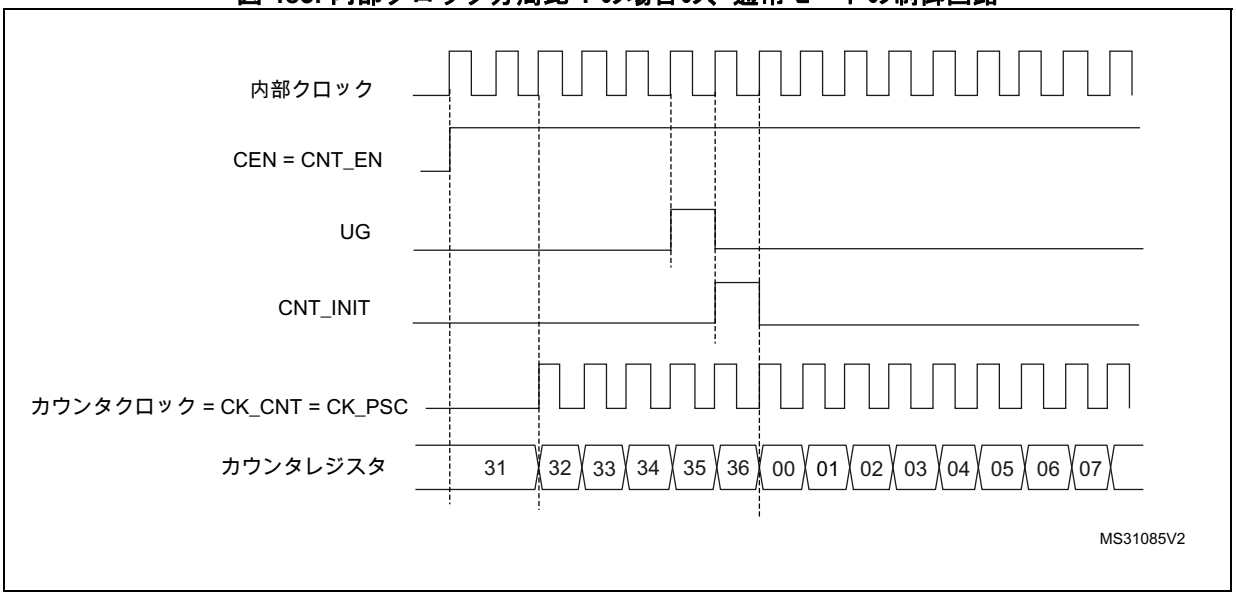
内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます

（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 483 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

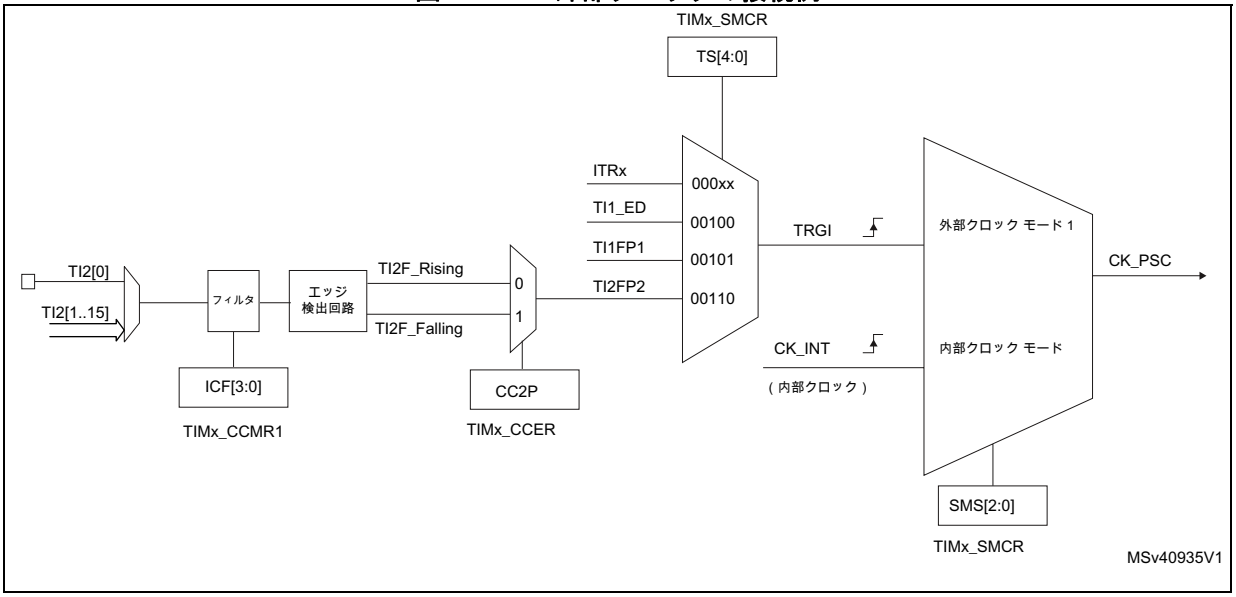
図 483. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 484. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

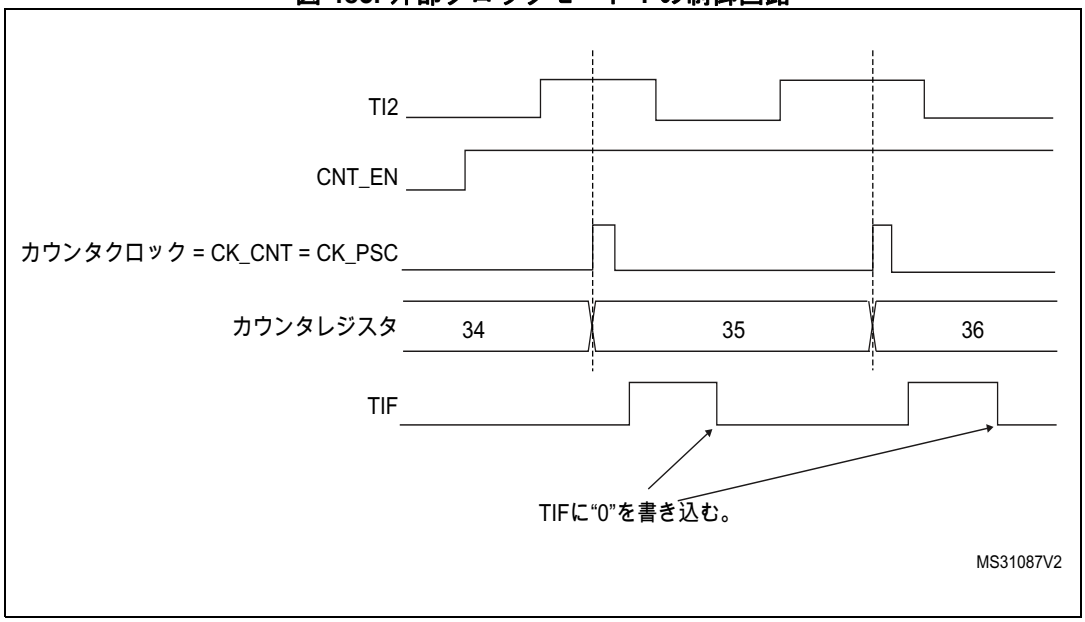
1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
4. CC2P=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 485. 外部クロックモード 1 の制御回路



41.4.5 キャプチャ／比較チャネル

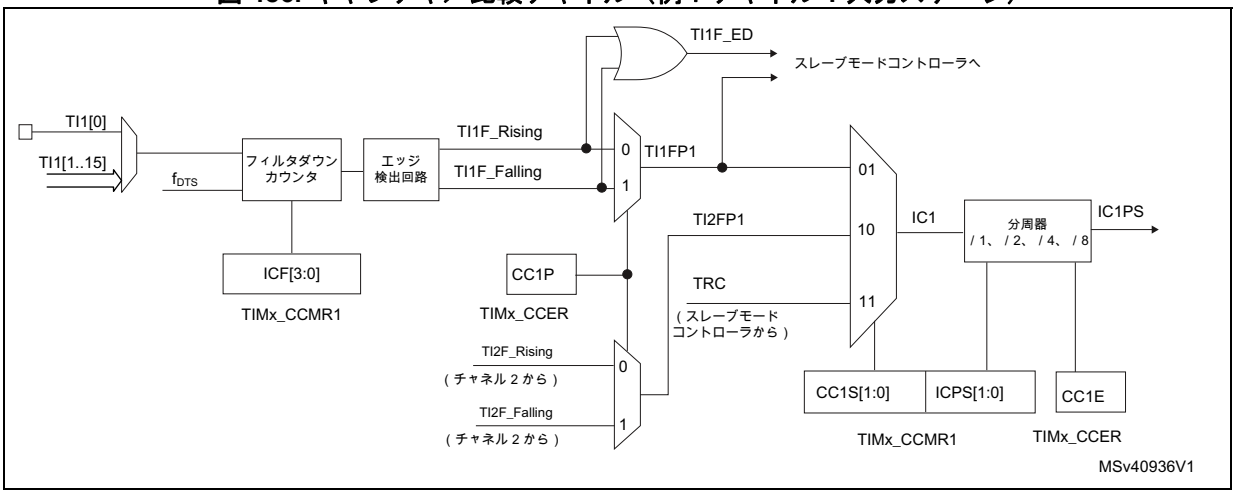
各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクサ、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 486 から *図 489* に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力とし

て、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 486. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 487. キャプチャ／比較チャネル 1 メイン回路

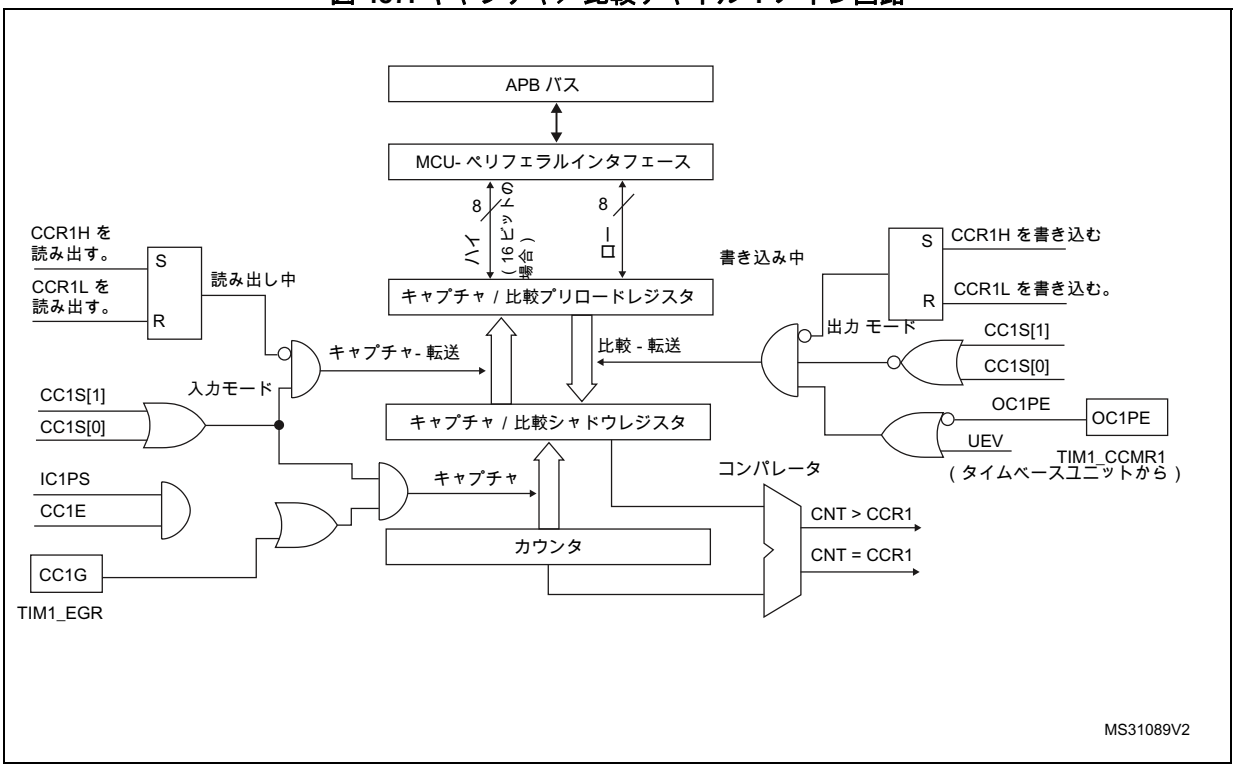


図 488. キャプチャ／比較チャネル（チャネル 1）の出力ステージ

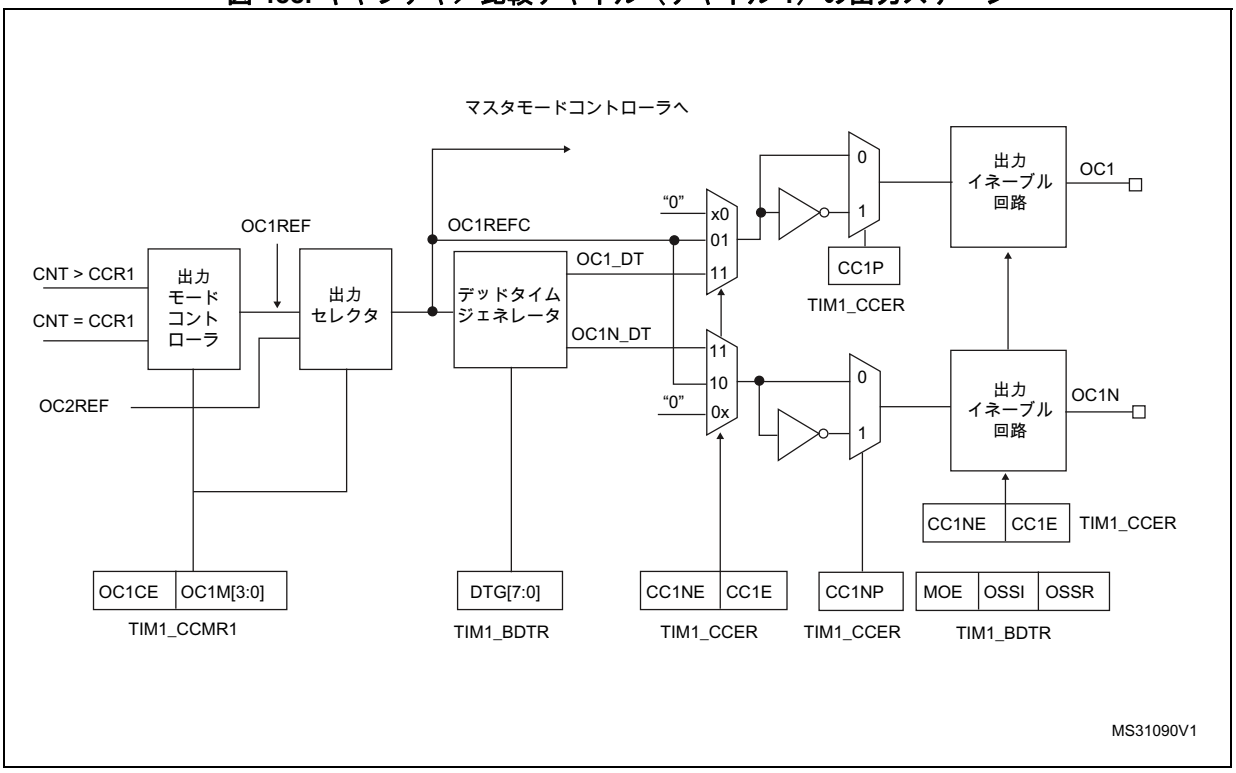
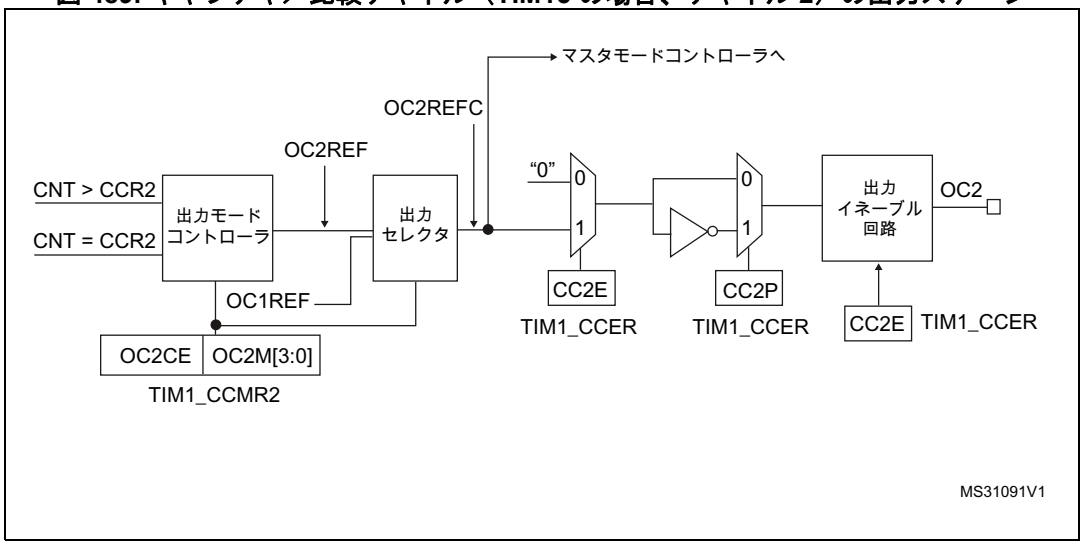


図 489. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

41.4.6 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
3. タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最低でも内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットに 0 を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

41.4.7 PWM 入力モード（TIM15 の場合のみ）

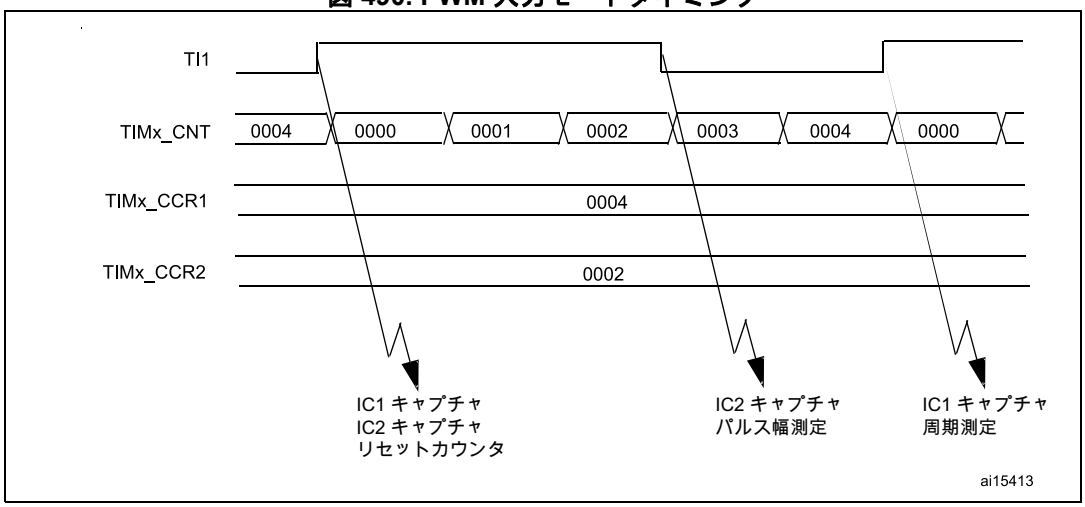
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに“1”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 490. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

41.4.8 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCxREF/OCx）を強制的にアクティブレベルとするには、対応する TIMx_OCMRx レジスタの OCxM ビットに“101” を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

41.4.9 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

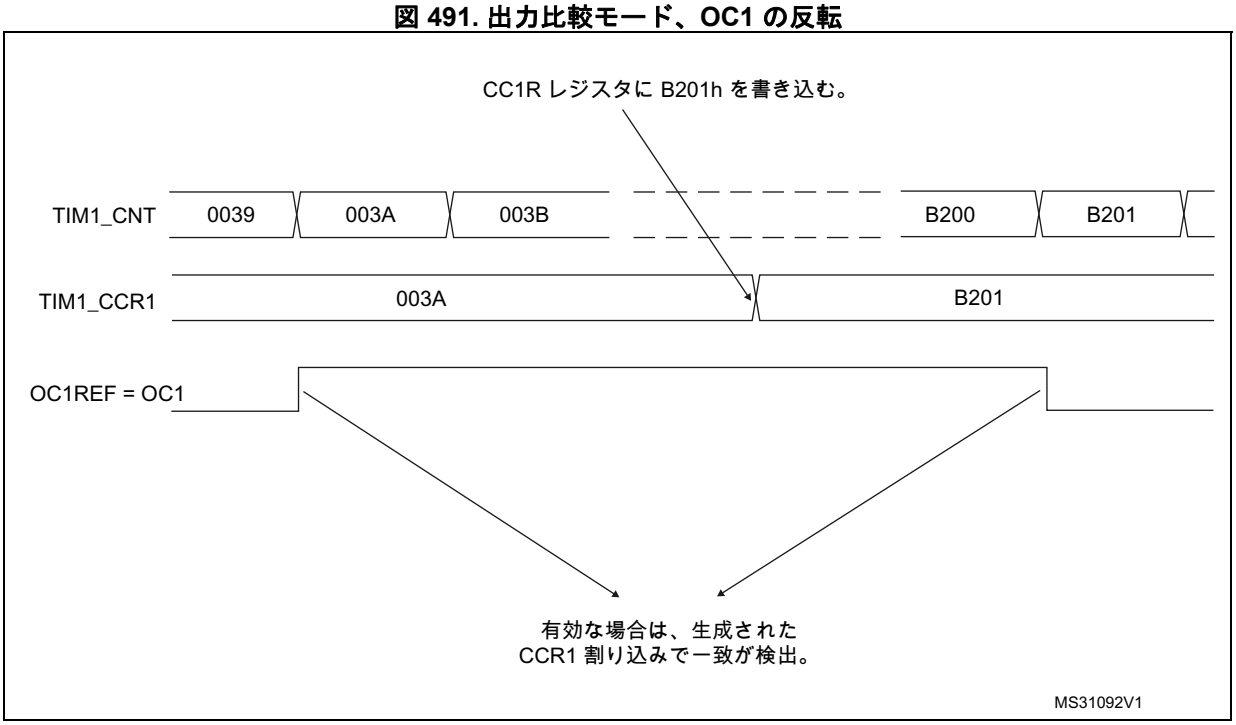
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 490* に示します。

図 491. 出力比較モード、OC1 の反転



41.4.10 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）または“111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

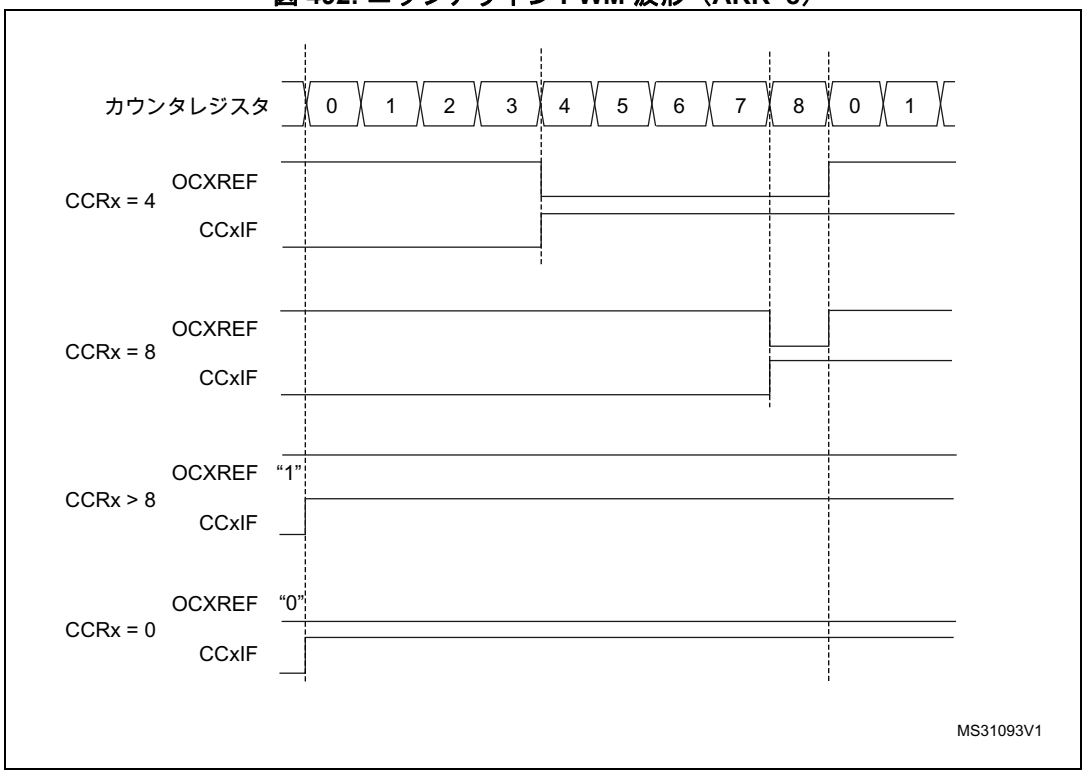
OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

TIM15/TIM16/TIM17 はアップカウント動作でのみ使用可能です。*1695 ページのアップカウントモード*を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 492* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 492. エッジアライン PWM 波形（ARR=8）



41.4.11 組み合わせ PWM モード（TIM15 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

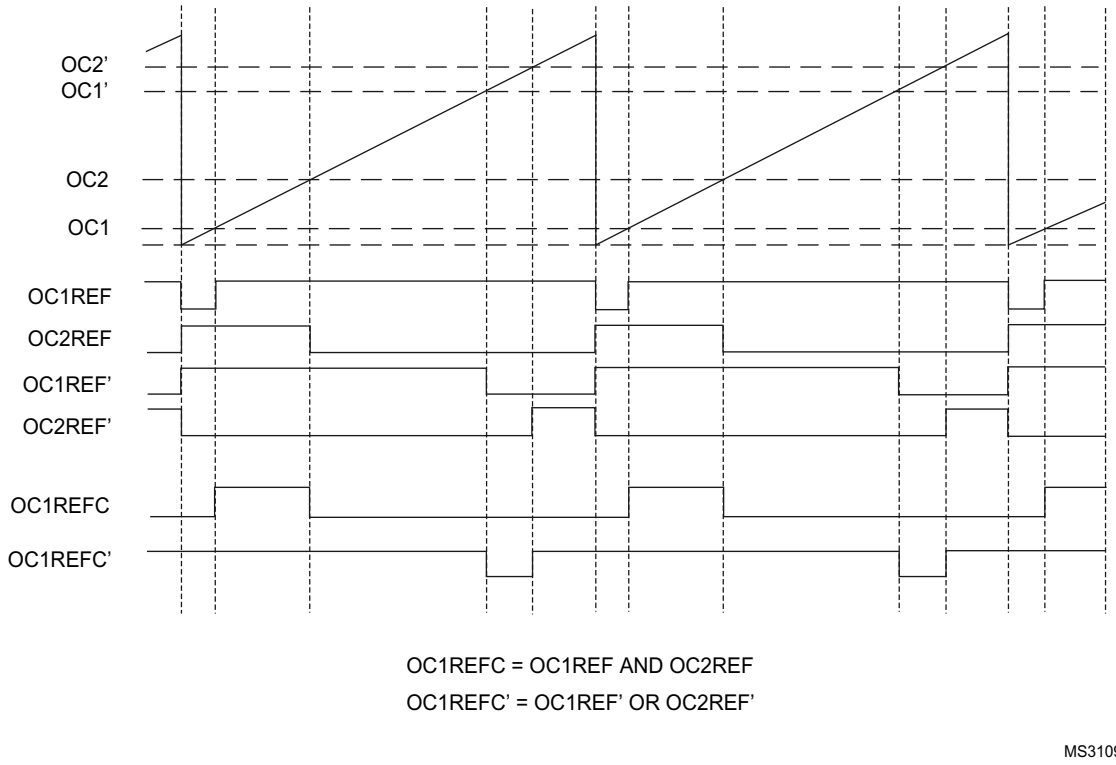
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 493 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 493. チャネル 1 および 2 における組み合わせ PWM モード



41.4.12 相補出力とデッドタイム挿入

TIM15/TIM16/TIM17 汎用タイマは、1 つの相補信号を出力して、出力時のスイッチオフおよびスイッチオンを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力の極性（主出力 OCx または補 OCxN）は、出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書き込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、*表 326：1739 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット （TIM15）*を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 494. デッドタイム挿入のある相補出力

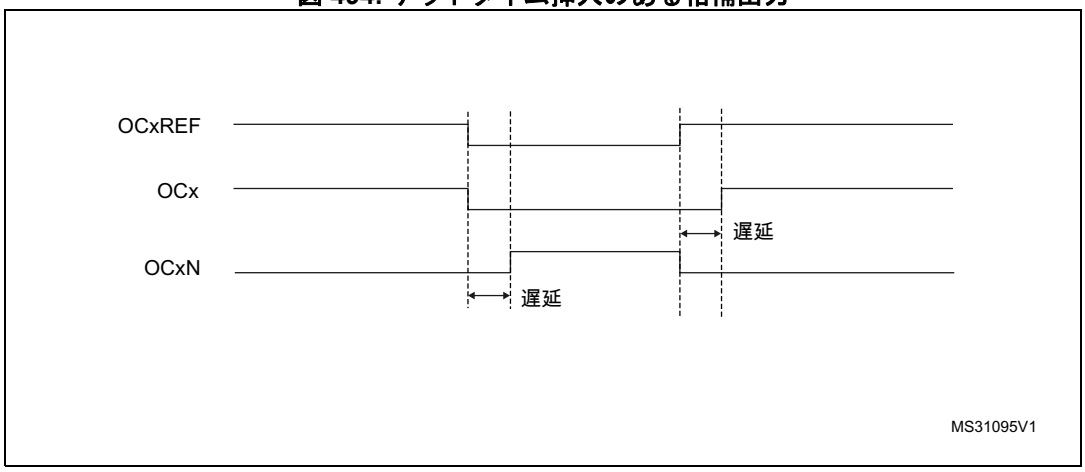


図 495. 負のパルスより長い遅延があるときのデッドタイムの波形

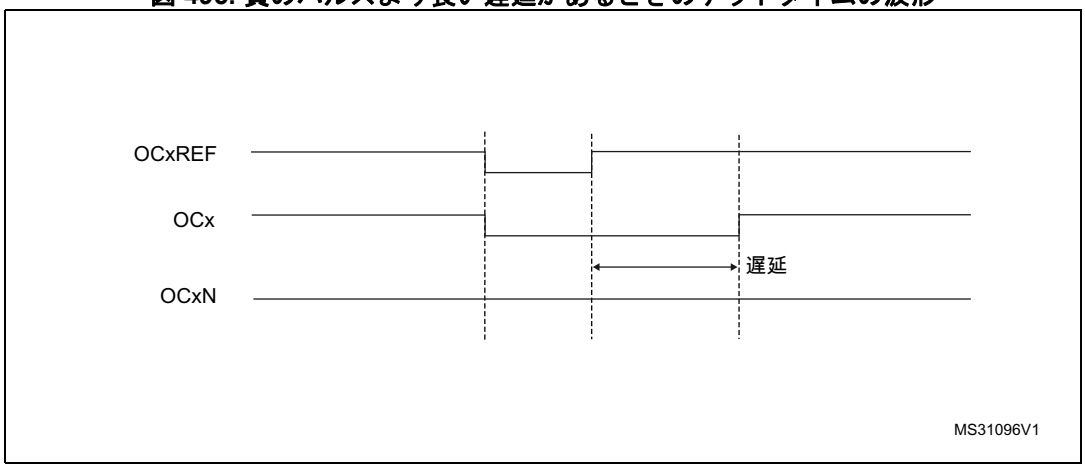
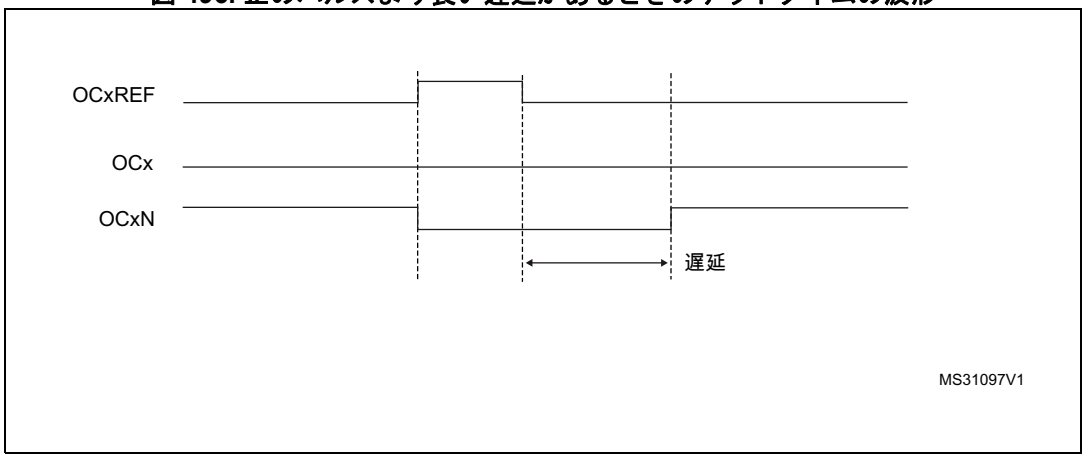


図 496. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、*セクション 41.5.15：1742 ページのTIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）*を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： ***OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx とOCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。***

41.4.13 ブレーク機能の使用

ブレーク機能の目的は、TIM15/TIM16/TIM17 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。ブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。

このブレーク チャネルは、システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵比較回路）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できます。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出力を両方ともアクティブレベルにセットすることはできません。詳細については、*表 326：1739*

*ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）*を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットをセットすることによって有効になります。ブレーク入力の極性は、同じレジスタの BKP ビットを設定することによって選択できます。BKE と BKP は、同時に変更できます。BKE および BKP ビットが書き込まれるとき、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書き込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 が書き込まれた場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、非同期信号を書き込んで、同期信号を読み出すからです。

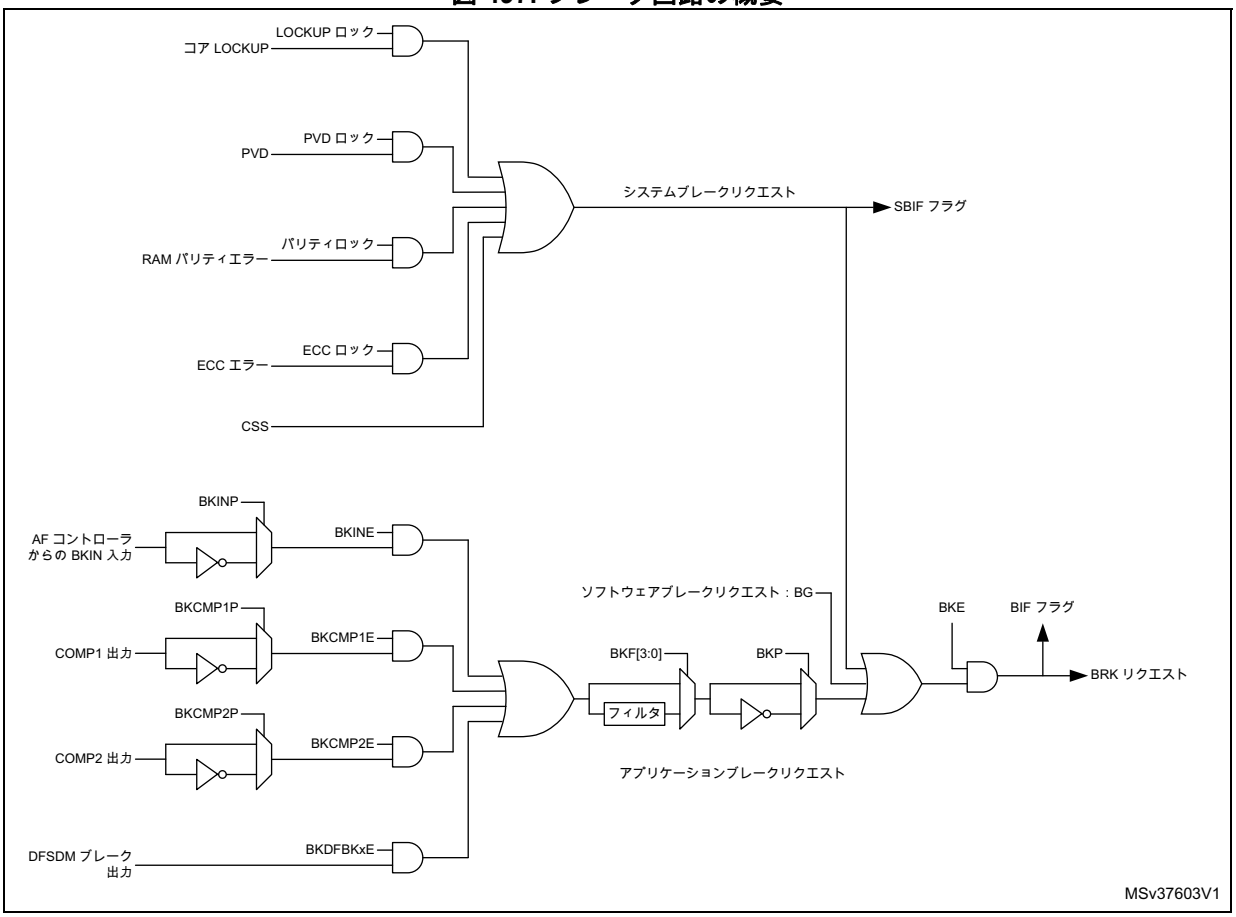
- 誤ったイベントを避けるためのプログラム可能なフィルタ（TIMx_BDTR レジスタの BKF[3:0] ビット）

ブレークは、TIMx_OR2 レジスタを使用して、個別に有効化でき、プログラム可能なエッジ感度を持つ複数のソースから生成できます。

ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力
  - システムブレーク：
    - Cortex®-M7 with FPU LOCKUP 出力
    - PVD 出力
    - SRAM パリティエラー信号
    - フラッシュ ECC エラー
    - CSS 検出回路によって生成されたクロック障害イベント

図 497. ブレーク回路の概要



注： ***非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。***

注意： ***非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。***

ブレークが発生すると（ブレーク入力で選択されたレベル）、

- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、AFIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合も同様です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずリセット状態のインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできませ

ん。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。

– OSSI=0 の場合、タイマはイネーブル出力（ハイインピーダンス状態を強制する AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。

- ブレーク状態フラグ（TIMx_SR レジスタの BIF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合は、割り込みを生成できます。TIMx_DIER レジスタの BDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント UEV で再び自動的にセットされます。これを使用して、たとえば、レギュレーションを行うことができます。そうでない場合、MOE ビットは、再び 1 が書き込まれるまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

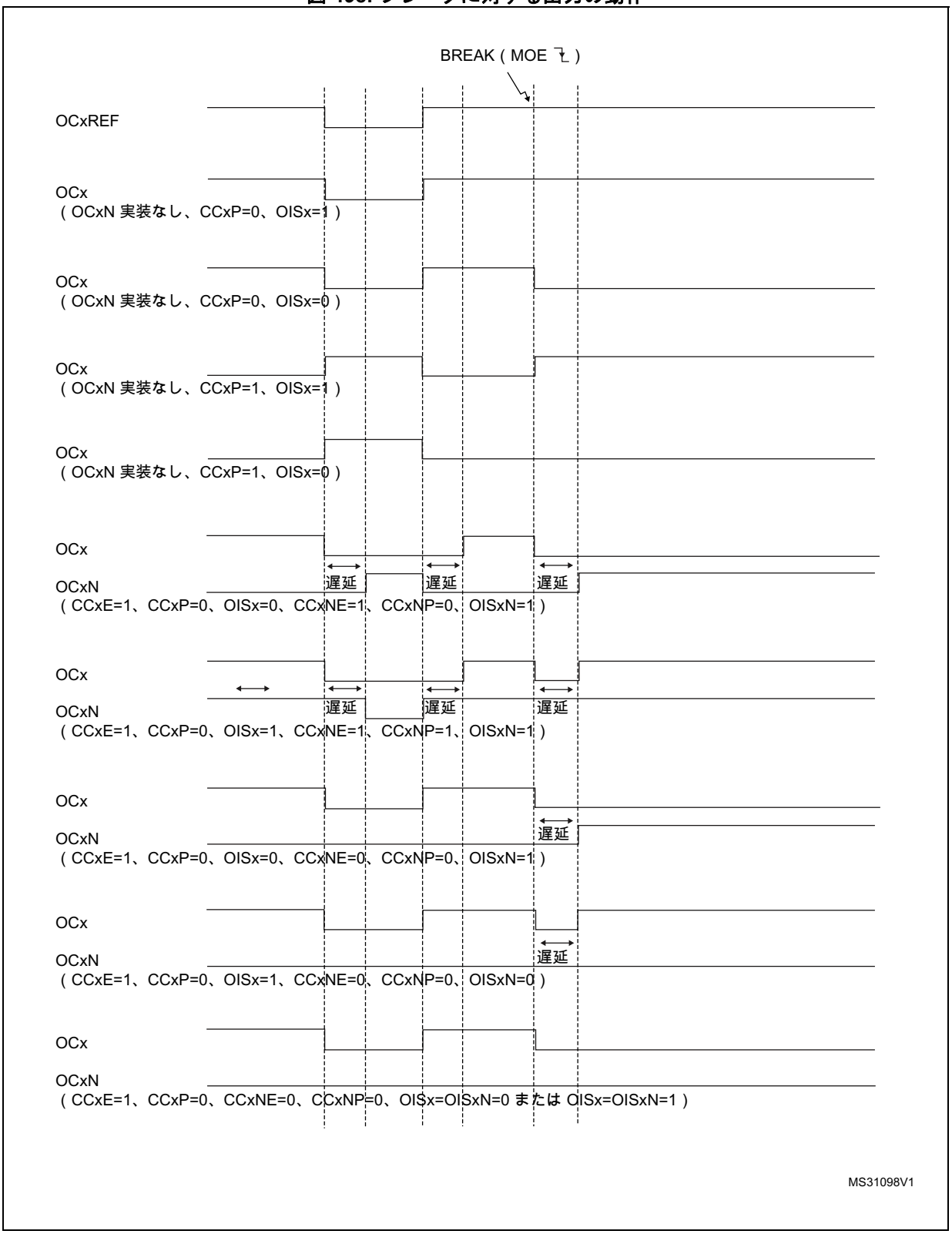
注： ***ブレーク入力は、信号レベルに対して動作します。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF をクリアできません。***

ブレークは、BRK 入力によって生成でき、BRK はプログラミング可能な極性を持ち、TIMx_BDTR レジスタの BKE がイネーブルビットです。

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書き込み保護機能を設けてあります。これにより、いくつかのパラメータ（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。*セクション 41.5.15：1742 ページのTIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）*を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 498 に、ブレークに対する出力の動作例を示します。

図 498. ブレークに対する出力の動作



41.4.14 ワンパルスモード

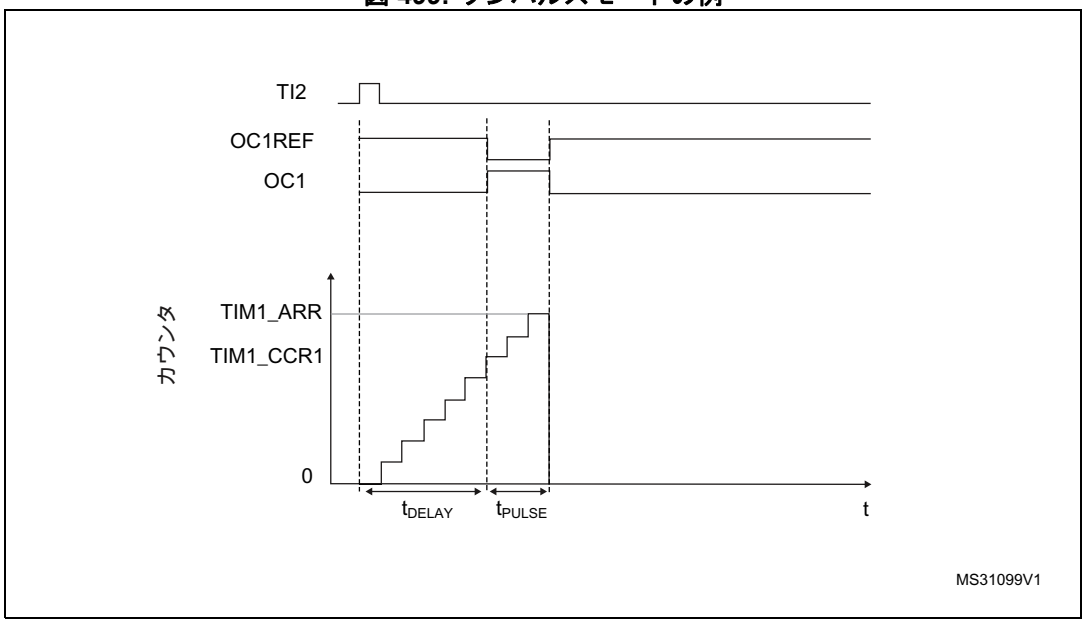
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 499. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに“1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。

特殊なケース：OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

41.4.15 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読み出し）と割り込み（更新の割り込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

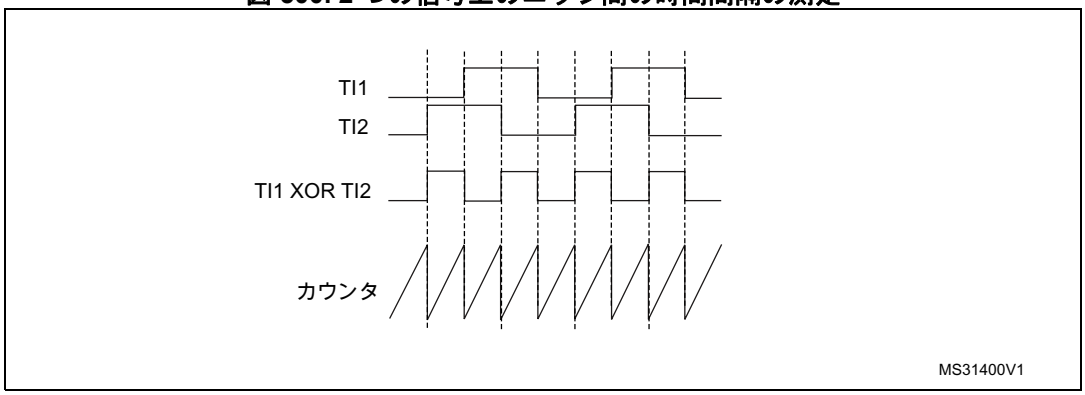
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

41.4.16 タイマ入力 XOR 機能（TIM15 のみ）

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン TIMx_CH1 および TIMx_CH2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の*図 500* に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 500. 2 つの信号上のエッジ間の時間間隔の測定



41.4.17 外部トリガ同期（TIM15 のみ）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。

TIM15 タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

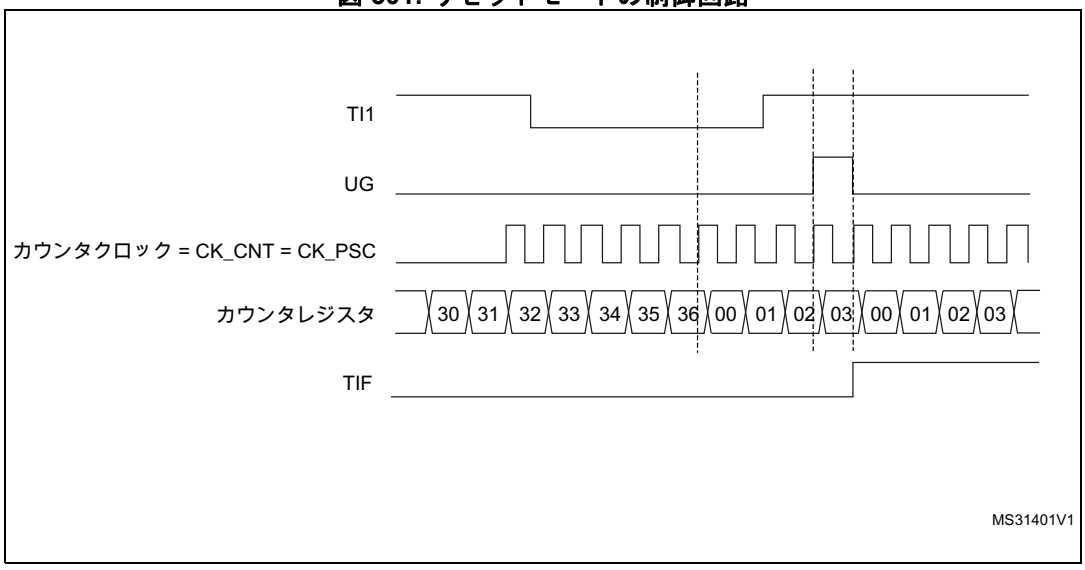
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビット）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 501. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

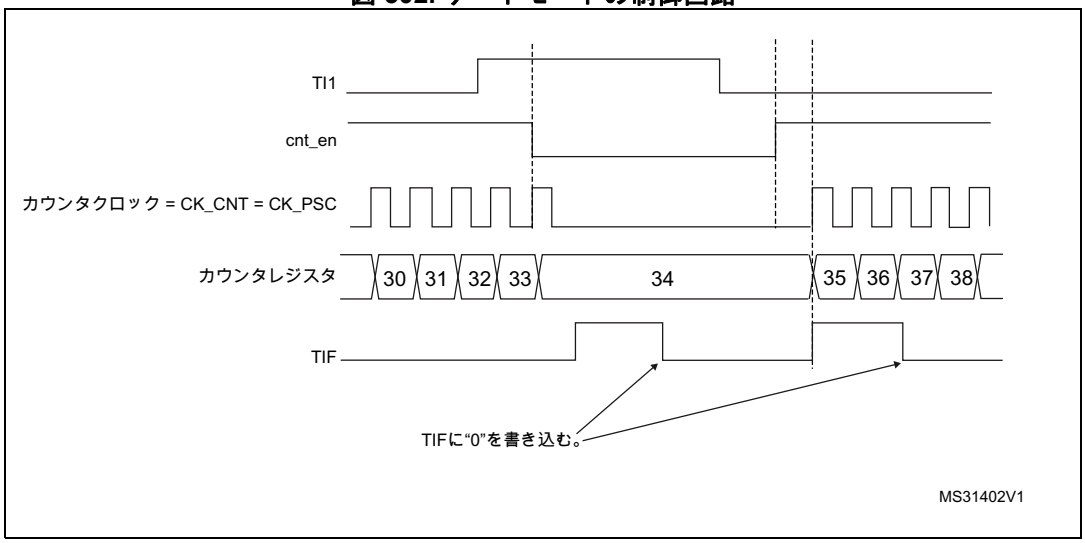
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 502. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

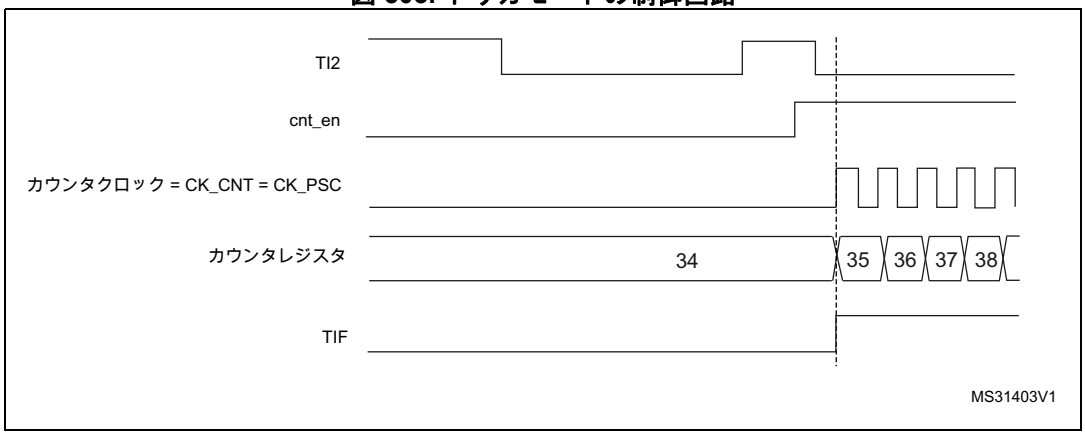
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P に「1」、CC2NP に「0」を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 503. トリガモードの制御回路



41.4.18 スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

41.4.19 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、いくつかのタイマレジスタをソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1、

00001：TIMx_CR2、

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

41.4.20 タイマ同期（TIM15）

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。詳細については、*セクション 41.4.20：タイマ同期（TIM15）*を参照してください。

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

41.4.21 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 with FPU コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、*セクション 60：デバッグインフラストラクチャ*を参照してください。

安全のため、カウンタが停止すると（DBGMCU_APB2FZ1 の TIMx = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

41.5 TIM15 レジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

41.5.1 TIM15 制御レジスタ 1 （TIM15_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2*t_{CK_INT}$
10：$t_{DTS} = 4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。
1：カウンタは有効です。

注： ***外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

41.5.2 TIM15 制御レジスタ 2 （TIM15_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
					rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）

0：MOE = 0 のとき、OC2=0
1：MOE = 1 のとき、OC2=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIMx_BKR レジスタの LOCK ビット）、変更できません。***

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKR レジスタの LOCK ビット）。***

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1 および CH2 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[1:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO）。

100：**比較** - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：**比較** - OC2REF 信号がトリガ出力（TRGO）として使用されます。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込みの後、これらのビットは、遷移イベント （COM）が発生した時にのみ更新されます （CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

41.5.3 TIM15 のスレーブモード制御レジスタ（TIM15_SMCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
								rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **TS[4:3]**：トリガ選択 - ビット 4:3
TS[4:0] 説明を参照 - ビット 6:4

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3
SMS 説明を参照 - ビット 2:0

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ/スレーブモード
0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[4:0]**：トリガ選択
このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。
00000：内部トリガ 0（ITR0）
00001：内部トリガ 1（ITR1）
00010：内部トリガ 2（ITR2）
00011：内部トリガ 3（ITR3）
00100：TI1 エッジ検出回路（TI1F_ED）
00101：フィルタタイマ入力 1（TI1FP1）
00110：フィルタタイマ入力 2（TI2FP2）
その他：予約済み
各タイマでの ITRx の詳細については、*表 325：1730 ページのTIMx 内部トリガ接続*を参照してください。

注： ***設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。***

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済み

0010：予約済み

0011：予約済み

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

その他：予約済み

注： ***トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。***

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

表 325. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 00000)	ITR1 (TS = 00001)	ITR2 (TS = 00010)	ITR3 (TS = 00011)
TIM15	TIM1	TIM3	TIM16 OC1	TIM17 OC1

41.5.4 TIM15 DMA／割り込み有効レジスタ（TIM15_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	rw	rw			rw	rw	rw	rw	rw	rw			rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル

0：トリガ DMA リクエストは無効です。

1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル
0：ブレーク割り込みは無効です。
1：ブレーク割り込みは有効です。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 **COMIE**：COM 割り込みイネーブル
0：COM 割り込みは無効です。
1：COM 割り込みは有効です。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

41.5.5 TIM15 ステータスレジスタ（TIM15_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0		rc_w0	rc_w0	rc_w0			rc_w0	rc_w0	rc_w0

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき、またはゲートモードが選択されている場合は、両方のエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割り込みが保留中です。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割り込みがペンディング中です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 41.5.3：TIM15 のスレーブモード制御レジスタ（TIM15_SMCR）*を参照）。

41.5.6 TIM15 のイベント生成レジスタ（TIM15_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
								w	w	rw			w	w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

41.5.7 TIM15 のキャプチャ／比較モードレジスタ 1（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:25 予約済み。常に 0 として読み出されます。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3
ビット 6:4 の OC1M 説明を参照

ビット 15 予約済み。常に 0 として読み出されます。

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 予約済み。常に 0 として読み出されます。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

1000：予約済み。

1001：予約済み。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：予約済み。

1111：予約済み。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

3：相補出力を持つチャネルでは、このビットはプリロードされます。もし、TIMx_CR2 レジスタの CCPC ビットがセットされているときには、OC1M のアクティブビットは、COM イベントが生成された場合のみプリロードされたビットから新しい値を取得します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済み。常に 0 として読み出されます。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。***

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

41.5.8 TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
								rw		rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ /比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ /比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を

参照してください。

注： 1. このビットは、LOCK レベル 2 または 3 がプログラムされ、（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力として設定）になった直後は書き込みできません。

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、遷移イベント(COM)が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 の極性を選択します。

00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

01：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 が反転されます（ゲートモードでのトリガ動作）。

10：予約済み。この設定は使用しないでください。

11：非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

注： 1.このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません (TIMx_BDTR レジスタの LOCK ビット)。

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、遷移イベント(COM)が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 326. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0 OCxN=0、OCxN_EN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP, OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： ***相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。***

41.5.9 TIM15 のカウンタ（TIM15_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

41.5.10 TIM15 のプリスケーラ（TIM15_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

41.5.11 TIM15 の自動再ロードレジスタ（TIM15_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値
ARR は、実際の自動再ロードレジスタにロードされる値です。
ARP の更新と動作の詳細については、*セクション 41.4.1： 1693 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

41.5.12 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

41.5.13 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

41.5.14 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。
CC2 チャネルが入力として設定されている場合：
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

41.5.15 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0](1)			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

1. BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初の書き込みアクセス時に、これらすべてを設定しなければならないことがあります。

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力信号をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： ***このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8：1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注： ***1：このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

2：このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。

1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。

このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8： 1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8： 1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。

00：LOCK オフ - どのビットも書き込み保護されません。

01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。

10：LOCK レベル 2 − LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 − LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns （250 ns 単位）

32 µs から 63 µs （1 µs 単位）

64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

41.5.16 TIM15 DMA 制御レジスタ（TIM15_DCR）

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送

00001：2 回転送、

00010：3 回転送、

...

10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1、

00001：TIMx_CR2、

00010：TIMx_SMCR

...

41.5.17 完全転送の TIM15 DMA アドレス（TIM15_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

41.5.18 TIM15 オルタネートレジスタ 1 （TIM15_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

41.5.19 TIM15 入力選択レジスタ（TIM15_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット [11:8] **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。

0000：TIM15_CH2 入力

0001：TIM2_CH2 入力

0010：TIM3_CH2 入力

0011：TIM4_CH2 入力

その他：予約済み

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット [3:0] **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM15_CH1 入力

0001：TIM2_CH1 入力

0010：TIM3_CH1 入力

0011：TIM4_CH1 入力

0100：LSE

0101：CSI

0110：MCO2

その他：予約済み

41.5.20 TIM15 レジスタマップ

TIM15 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 327. TIM15 レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM15_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIM15_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値																						0	0	0	0	0	0	0	0	0		0
0x08	TIM15_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0									0	0	0	0		0	0	0
0x0C	TIM15_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																		0	0			0	0	0	0	0	0			0	0	0
0x10	TIM15_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0		0	0	0			0	0	0
0x14	TIM15_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
	リセット値																									0	0	0			0	0	0
0x18	TIM15_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
	TIM15_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM15_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値																									0		0	0	0	0	0	0
0x24	TIM15_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 327. TIM15 レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	TIM15_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM15_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM15_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIM15_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM15_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIM15_BDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM15_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM15_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM15_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM15_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL[3:0]				Res.	Res.	Res.	Res.	TI1SEL[3:0]			
	リセット値																					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

41.6 TIM16/TIM17 レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

41.6.1 TIM16/TIM17 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。
0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ***外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

41.6.2 TIM16/TIM17 制御レジスタ 2 （TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0

1：MOE=1 のとき、デッドタイム後に OC1N=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。***

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： ***このビットは、相補出力を持つチャネルでのみ機能します。***

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込み後は、COM ビットがセットされた場合にのみ更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

41.6.3 TIM16/TIM17 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
		rw				rw	rw	rw		rw				rw	rw

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **COMDE**：COM DMA リクエストイネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル

0：CC1 DMA リクエストは無効です。

1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル

0：ブレーク割り込みは無効です。

1：ブレーク割り込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIE**：COM 割り込みイネーブル

0：COM 割り込みは無効です。

1：COM 割り込みは有効です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル

0：CC1 割り込みは無効です。

1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル

0：更新割り込みは無効です。

1：更新割り込みは有効です。

41.6.4 TIM16/TIM17 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割り込みがペンディング中です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：

このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。
0：更新は発生していません。
1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。
- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。
- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 41.5.3：TIM15 のスレーブモード制御レジスタ（TIM15_SMCR）*を参照）。

41.6.5 TIM16/TIM17 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
								w		w				w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

41.6.6 TIM16/TIM17 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
															Res.
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
								IC1F[3:0]				IC1PSC[1:0]			
								rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード（ビット 3）

ビット 15:7 予約済み

ビット 6:4 **OC1M[2:0]**：出力比較 1 モード（ビット 2 ～ 0）

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNTがキャプチャ／比較レジスタ 1(TIMx_CCR1)と一致したときに､強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他の値：予約済み

注： ***1：これらのビットは､LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは､LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}$= f_{CK_INT}、N = 2
0010：$f_{SAMPLING}$= f_{CK_INT}、N = 4
0011：$f_{SAMPLING}$= f_{CK_INT}、N = 8
0100：$f_{SAMPLING}$= f_{DTS}/2、N =
0101：$f_{SAMPLING}$= f_{DTS}/2、N = 8
0110：$f_{SAMPLING}$= f_{DTS}/4、N = 6
0111：$f_{SAMPLING}$= f_{DTS}/4、N = 8
1000：$f_{SAMPLING}$= f_{DTS}/8、N = 6
1001：$f_{SAMPLING}$= f_{DTS}/8、N = 8
1010：$f_{SAMPLING}$= f_{DTS}/16、N = 5
1011：$f_{SAMPLING}$ = f_{DTS}/16、N = 6
1100：$f_{SAMPLING}$= f_{DTS}/16、N = 8
1101：$f_{SAMPLING}$= f_{DTS}/32、N = 5
1110：$f_{SAMPLING}$= f_{DTS}/32、N = 6
1111：$f_{SAMPLING}$= f_{DTS}/32、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

41.6.7 TIM16/TIM17 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を

参照してください。

注： ***1.このビットは、LOCK レベル 2 または 3 がプログラムされ、(TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力として設定）になると書き込みできません。***

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、遷移イベント(COM)が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：

CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 の極性を選択します。

00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

01：反転／立下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 が反転されます（ゲートモードでのトリガ動作）。

10：予約済み。この設定は使用しないでください。

1：非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

注： ***1.このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません (TIMx_BDTR レジスタの LOCK ビット)。***

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、遷移イベント(COM)が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：

このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 328. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0 OCxN=0、OCxN_EN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP、 OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： ***相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。***

41.6.8 TIM16/TIM17 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

41.6.9 TIM16/TIM17 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

41.6.10 TIM16/TIM17自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、*セクション 41.4.1： 1693 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

41.6.11 TIM16/TIM17 繰り返しカウンタレジスタ（TIMx_RCR）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

41.6.12 TIM16/TIM17 キャプチャ／比較レジスタ 1 （TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

41.6.13 TIM16/TIM17 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初の書き込みアクセス時に、これらすべてを設定しなければならないことがあります。***

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。
1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8：1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注： ***1.このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

2.このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。

1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。

注： ***1.このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

2.このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8： 1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： ***このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 41.5.8： 1737 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。
0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。
00：LOCK オフ - どのビットも書き込み保護されません。
01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。
10：LOCK レベル 2 − LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。
11：LOCK レベル 3 − LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで凍結されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。
DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns （250 ns 単位）
32 µs から 63 µs （1 µs 単位）
64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

41.6.14 TIM16/TIM17 DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）、つまり転送回数を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1、
00001：TIMx_CR2、
00010：TIMx_SMCR
...

例：次の転送を考えます：DBL = 7 転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

41.6.15 完全転送の TIM16/TIM17 DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

41.6.16 TIM16 オルタネート機能レジスタ 1 （TIM16_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDFBK1E**：BRK dfsdm1_break[1] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

41.6.17　TIM16 入力選択レジスタ（TIM16_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット [3:0]　**TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM16_CH1 入力

0001：LSI

0010：LSE

0011：WKUP_IT

その他：予約済み

41.6.18 TIM17 オルタネート機能レジスタ 1 （TIM17_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[2] 入力は無効です。

1：dfsdm1_break[2] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

41.6.19　TIM17 入力選択レジスタ（TIM17_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット [3:0]　**TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM17_CH1 入力

0001：SPDIF FS

0010：HSE_1MHz

0011：MCO1

その他：予約済み

41.6.20 TIM16/TIM17 レジスタマップ

TIM16/TIM17 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 329. TIM16/TIM17レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1		Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
	リセット値																							0	0					0	0		0
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
	リセット値																			0				0	0	0		0				0	0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0		0		0				0	0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
	リセット値																									0		0				0	0
0x18	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]				OC1PE	OC1FE	CC1S[1:0]	
	リセット値																0										0	0	0			0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0x20	TIMx_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
	リセット値																													0	0	0	0
0x24	TIMx_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 329. TIM16/TIM17レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIMx_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIMx_BDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIMx_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIMx_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM16_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x60	TIM17_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM16_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TISEL[3:0]			
	リセット値																													0	0	0	0
0x68	TIM17_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TISEL[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

42 基本タイマ（TIM6/TIM7）

42.1 TIM6/TIM7 の概要

基本タイマ TIM6 および TIM7 は、プログラマブルなプリスケーラによって駆動される 16 ビット自動再ロードカウンタで構成されています。

このタイマは、タイムベース生成を目的とした汎用タイマとして使用できますが、特にデジタルアナログコンバータ（DAC）の駆動にも使用されます。実際、これらのタイマは内部で DAC に接続されており、トリガ出力を通じて駆動できます。

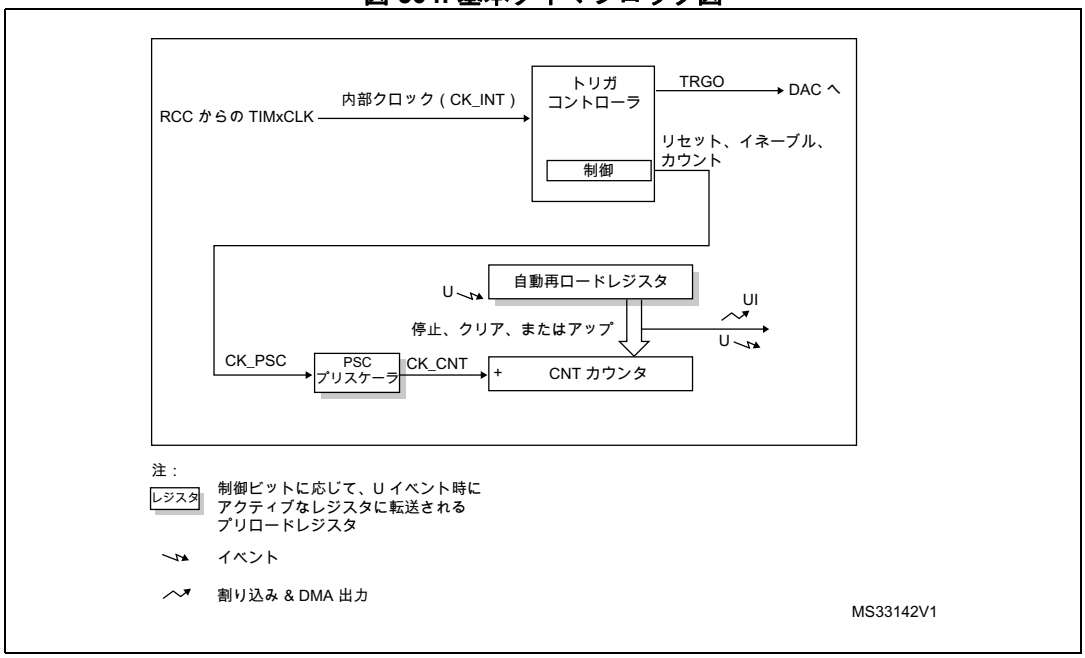
タイマは完全に独立していて、いかなるリソースも共有しません。

42.2 TIM6/TIM7 の主な機能

基本タイマ（TIM6/TIM7）の機能は、次のとおりです。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- DAC をトリガする同期回路
- 更新イベント時の割り込み／DMA 生成：カウンタオーバーフロー

図 504. 基本タイマブロック図



42.3 TIM6/TIM7 の機能詳細

42.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント UEV ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときのみ、プリスケーラ出力 CK_CNT から供給されます。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。TIMx_PSC 制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 505 と *図 506* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 505. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

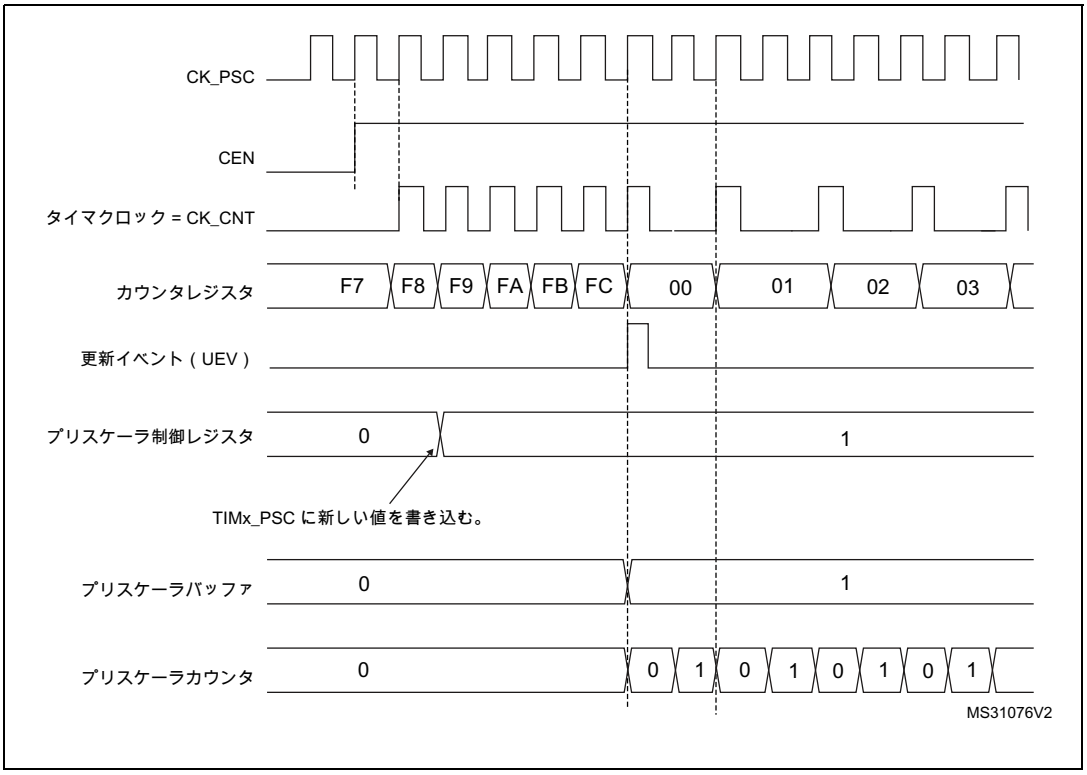
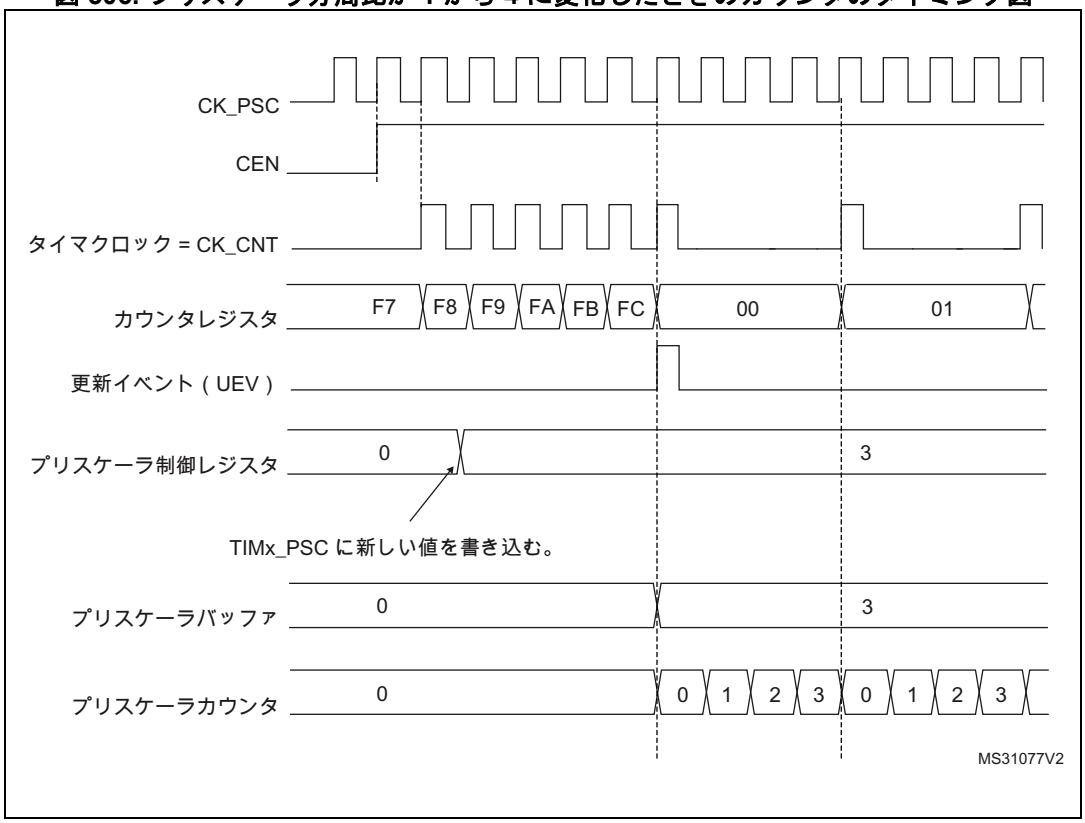


図 506. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



42.3.2 カウントモード

カウンタは、0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントした後、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGR レジスタの UG ビットをセットすることによって（ソフトウェアで、または、スレーブモードコントローラを使用して）生成できます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDIS ビットに 0 が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケーラカウンタは両方とも 0 からリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS（更新リクエスト選択）ビットがセットされている場合、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（割り込みや DMA リクエストは送信されません）。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットの設定に応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 507. 内部クロック分周比が 1 の場合のカウンタのタイミング図

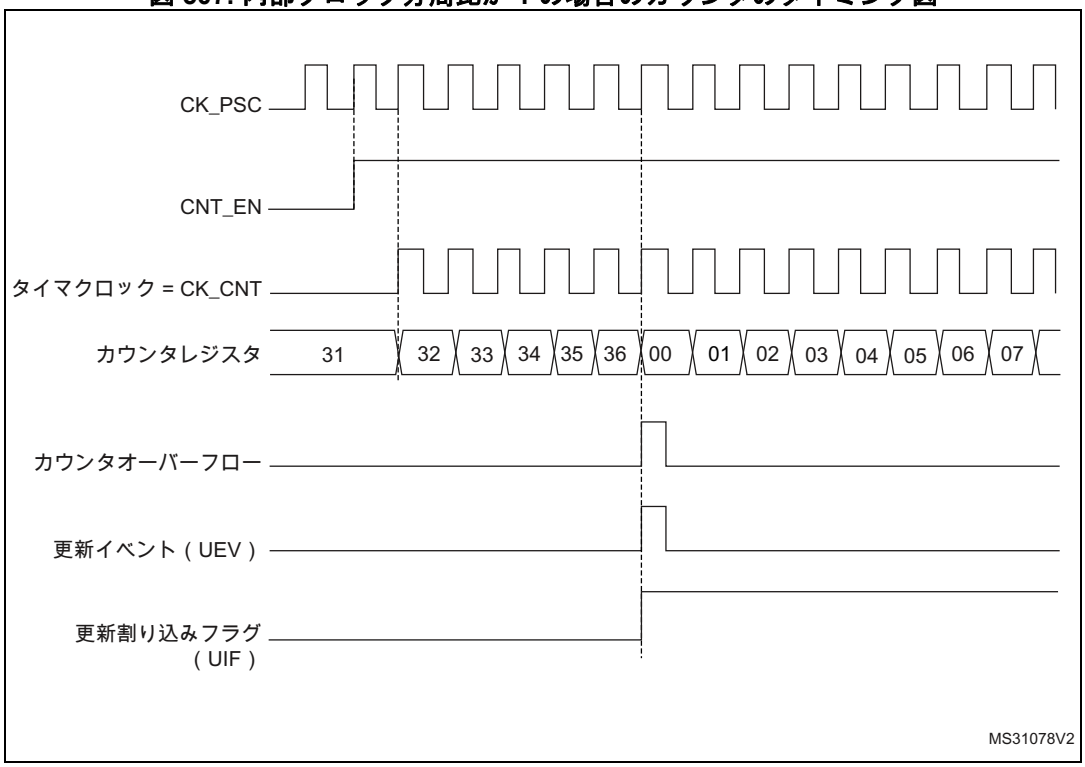


図 508. 内部クロック分周比が 2 の場合のカウンタのタイミング図

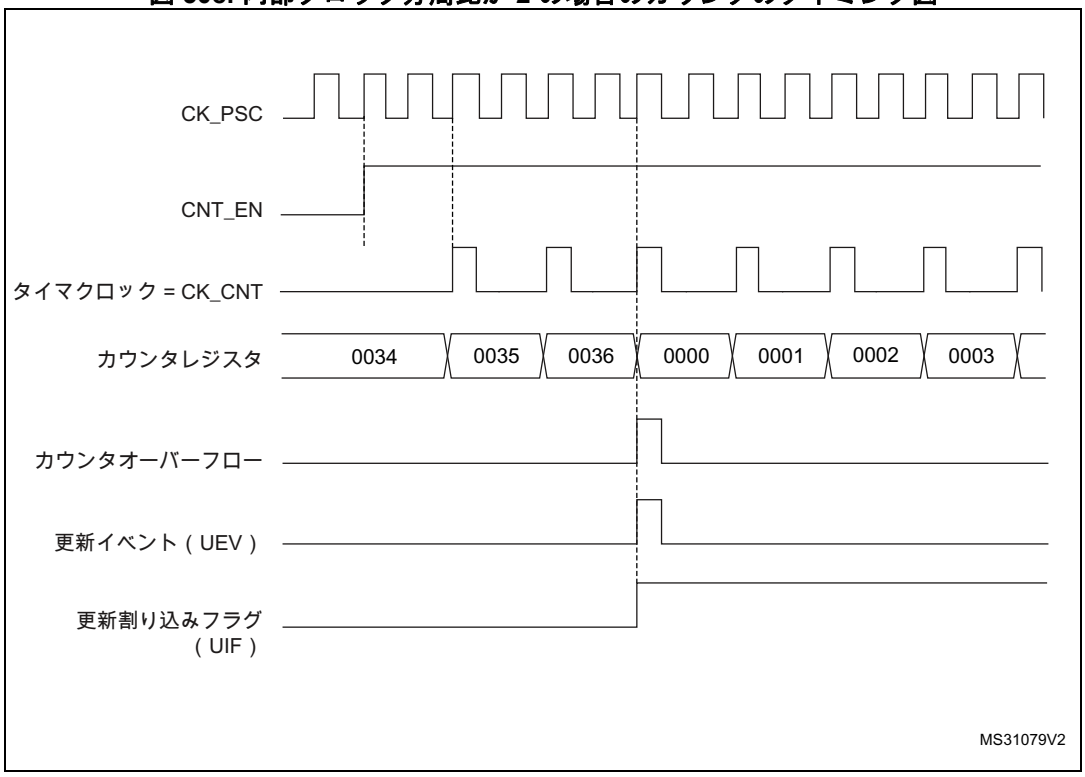


図 509. 内部クロック分周比が 4 の場合のカウンタのタイミング図

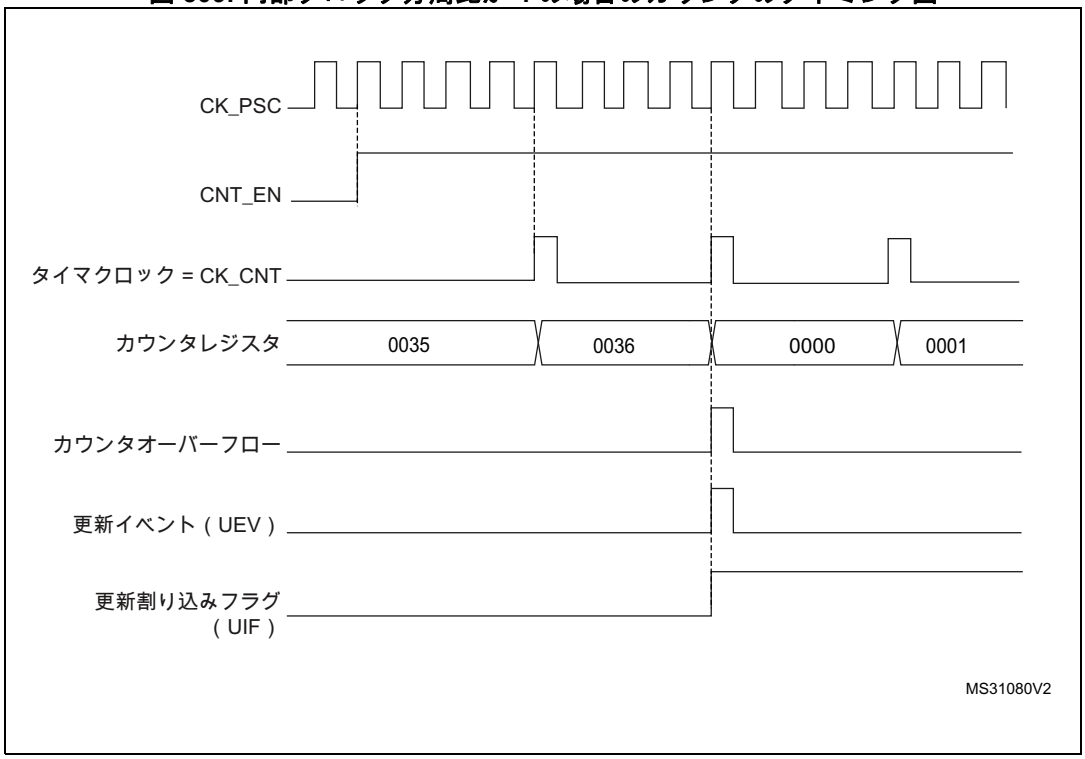


図 510. 内部クロック分周比が N の場合のカウンタのタイミング図

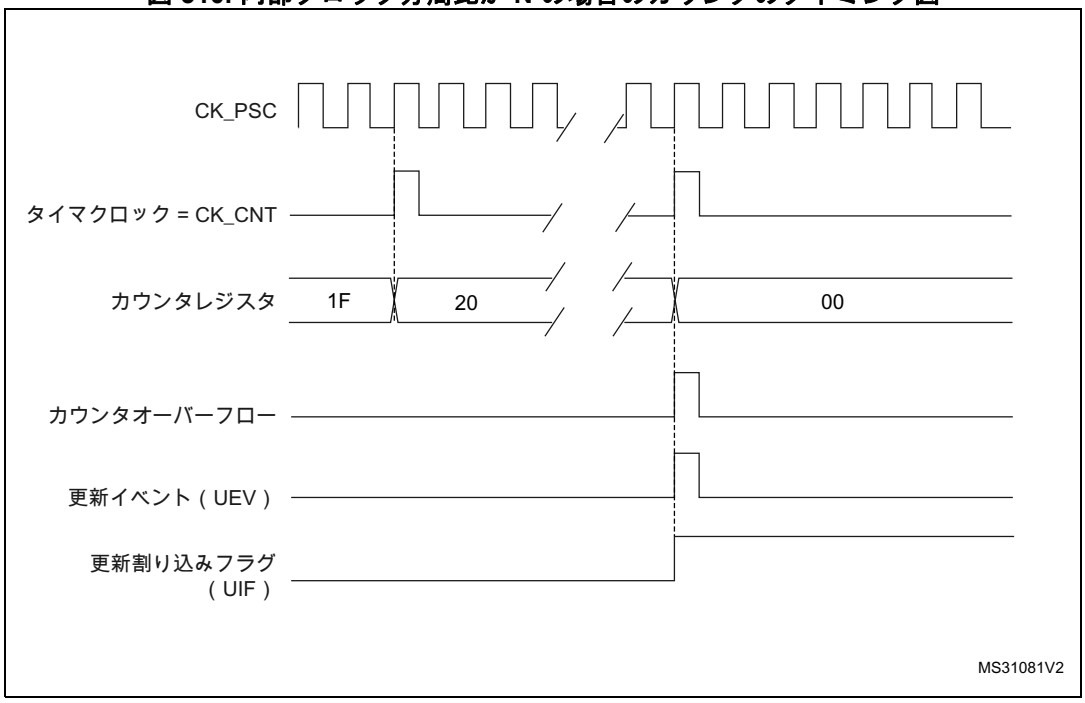


図 511. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

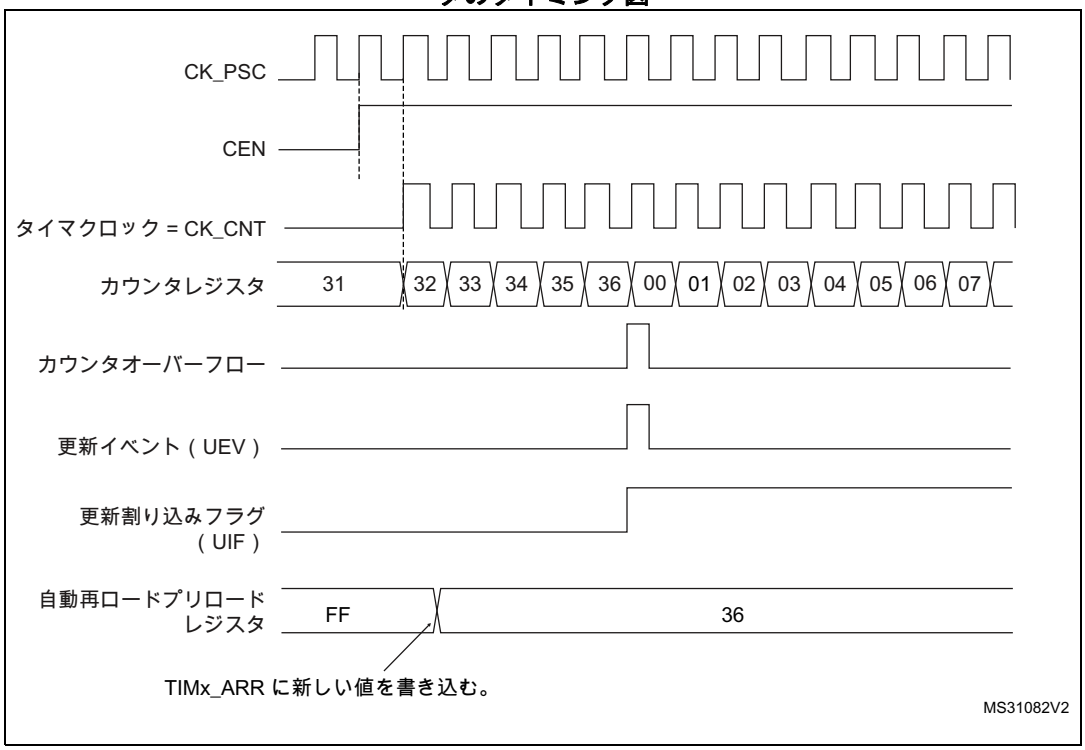
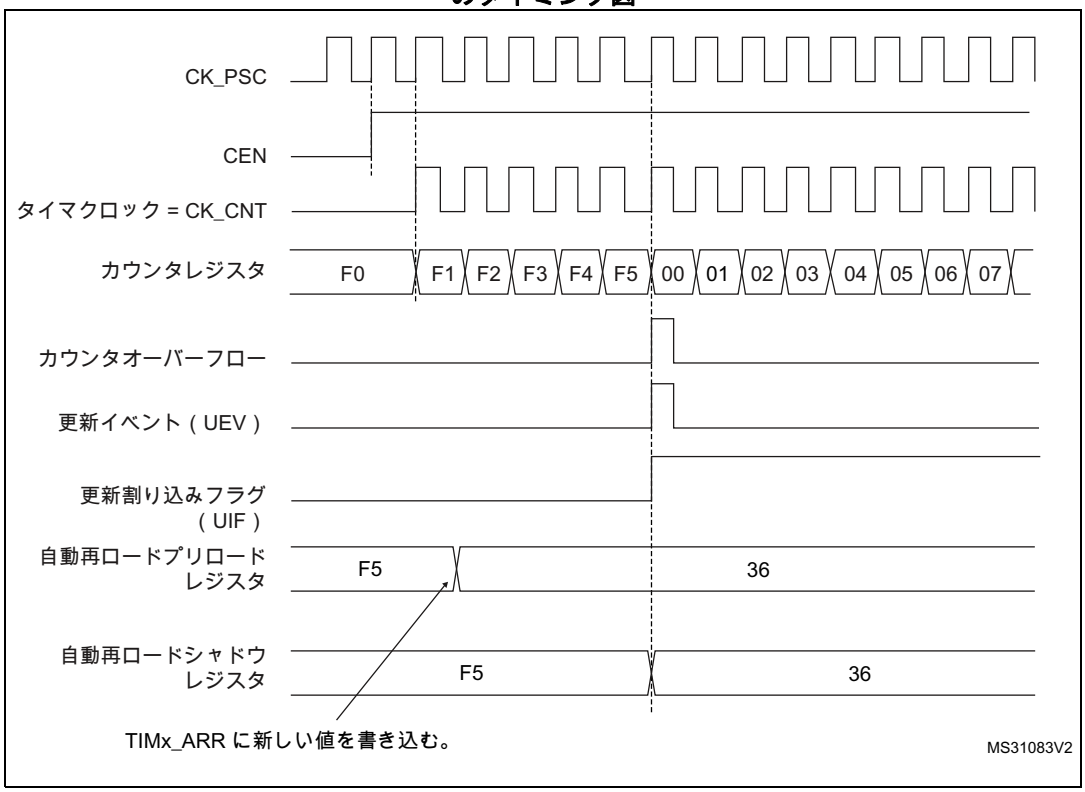


図 512. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



42.3.3 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読み出し）と割り込み（更新の割り込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

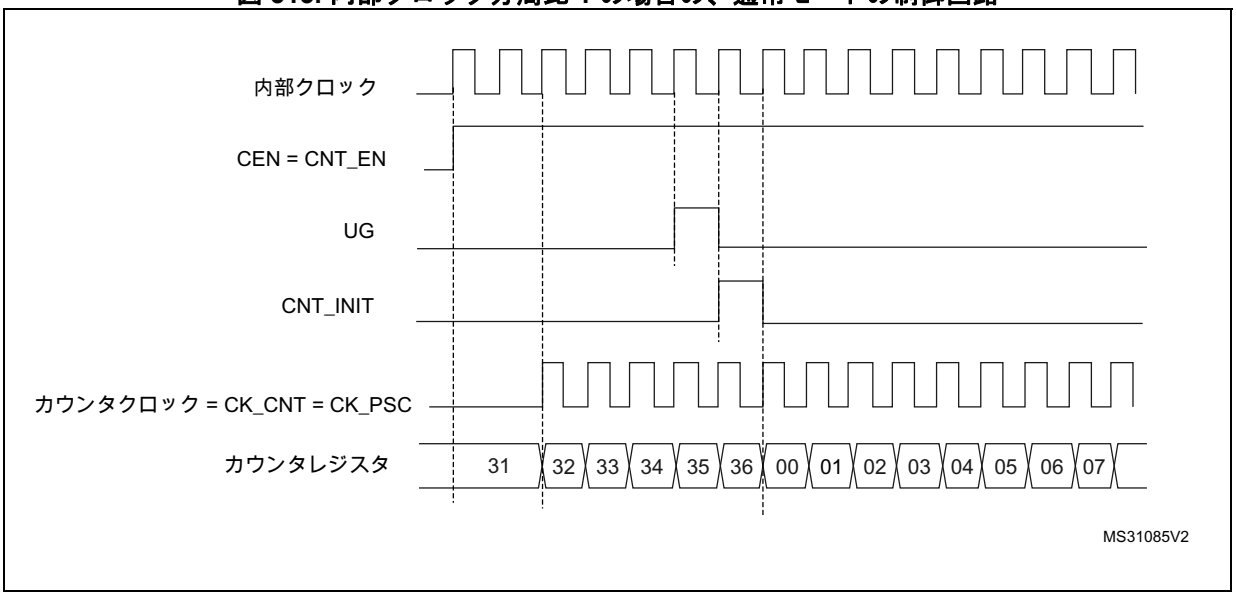
42.3.4 クロックソース

カウンタクロックは、内部クロック（CK_INT）ソースから供給されます。

TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます（ただし、自動的にクリア状態が保持される UG ビットを除く）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 513 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

図 513. 内部クロック分周比 1 の場合の、通常モードの制御回路



42.3.5 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 with FPU コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、*セクション 60.5.8：マイクロコントローラデバッグユニット（DBGMCU）*を参照してください。

42.4 TIM6/TIM7 レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

42.4.1 TIM6/TIM7 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIF RE-MAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw				rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置
0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル
0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード
0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース
このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。
0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。
– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成
1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル
このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。
– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成
バッファを持つレジスタにはプリロード値がロードされます。
1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC）はそれぞれの値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル
0：カウンタは無効です。
1：カウンタは有効です。

注： ゲートモードは、CEN ビットが事前にソフトウェアでセットされている場合にのみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

42.4.2 TIM6/TIM7 制御レジスタ 2 （TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS[2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（TRGO）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが発生したとき（スレーブモードコントローラがリセットモードに設定されているとき）TRGO 信号は実際のリセットから遅れて発生します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除いて、TRGO に遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

42.4.3 TIM6/TIM7 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
							rw								rw

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIE**：更新割り込みイネーブル

0：更新割り込みは無効です。

1：更新割り込みは有効です。

42.4.4 TIM6/TIM7 のステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
															rc_w0

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– TIMx_CR1 レジスタの UDIS = 0 であり、繰り返しカウンタ値でオーバーフローまたはアンダーフローが発生したとき。

– TIMx_CR1 レジスタの URS = 0 かつ UDIS = 0 の場合に、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで CNT が再初期化されたとき。

42.4.5 TIM6/TIM7 のイベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
															w

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

42.4.6 TIM6/TIM7 のカウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

42.4.7 TIM6/TIM7 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブなプリスケーラレジスタにロードされる値を含みます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

42.4.8 TIM6/TIM7 の自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：プリスケーラ値
ARR は、実際の自動再ロードレジスタにロードされる値です。
APR の更新と動作の詳細については、*セクション 42.3.1：1774 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

42.4.9 TIM6/TIM7 レジスタマップ

TIMx レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 330. TIM6/TIM7 レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0				0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS [2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x08	予約済み																																
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
	リセット値																								0								0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
	リセット値																																0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
	リセット値																																0
0x18～0x20	予約済み																																
0x24	TIMx_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

43 低電力タイマ（LPTIM）

43.1 概要

LPTIM は、消費電力削減の究極的な進展を利用した 16 ビットタイマです。クロックソースの多様性により、LPTIM は STANDBY モードを除くすべての電力モードで実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIM は、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIM はシステムを低電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIM の柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

43.2 LPTIM の主な機能

- 16 ビットアップカウンタ
- 8 つの分周比（1、2、4、8、16、32、64、128）を持つ 3 ビットプリスケーラ
- 選択可能なクロック
  - 内部クロックソース：LSE、LSI、HSI 、または APB クロック
  - LPTIM 入力経由の外部クロックソース（LP オシレータが実行していないときに動作、パルスカウンタアプリケーションによって使用）
- 16 ビット ARR 自動再ロードレジスタ
- 16 ビット比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能な I/O 極性
- エンコーダモード

43.3 LPTIM の実装

表 331 に、STM32H7x3 デバイスでの LPTIM の実装を示します。

表 331. STM32H7x3 LPTIM 機能

LPTIM モード／機能(1)	LPTIM1	LPTIM2	LPTIM3	LPTIM4	LPTIM5
エンコーダモード	X	X	-	-	-

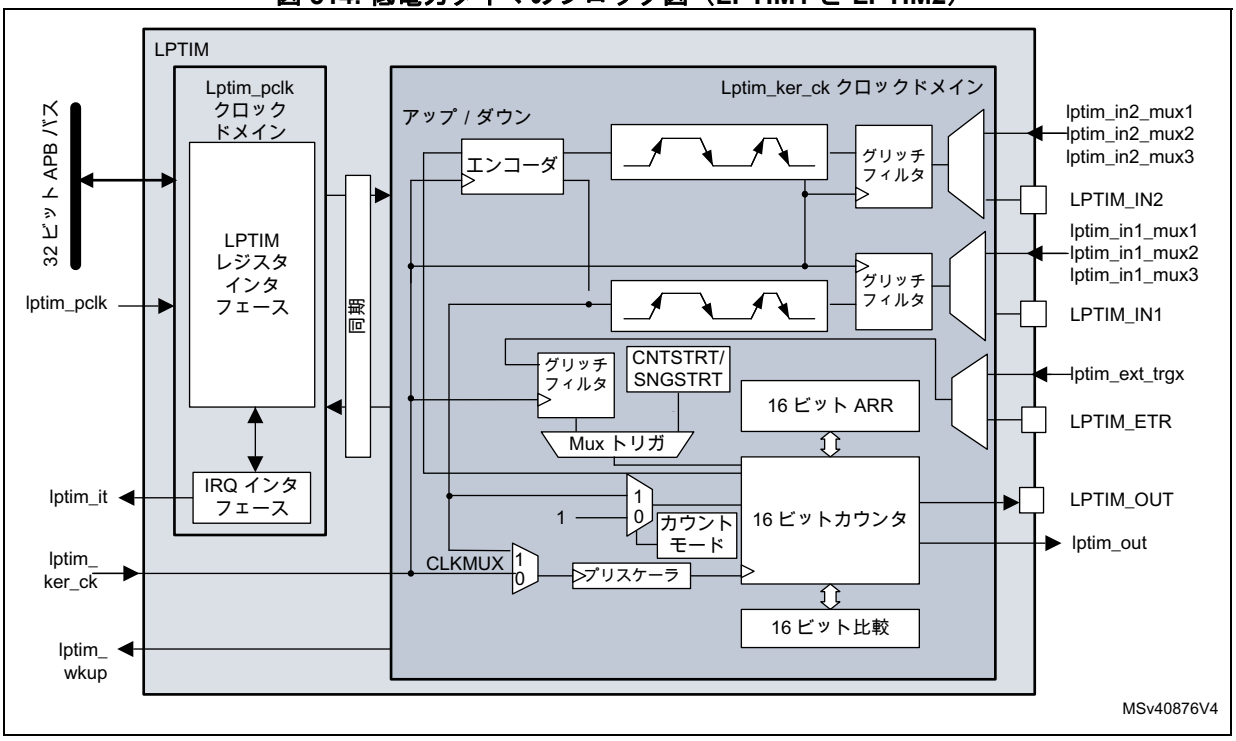
1. X：サポートされています。

参考資料

43.4 LPTIM の機能詳細

43.4.1 LPTIM ブロック図

図 514. 低電力タイマのブロック図（LPTIM1 と LPTIM2）



参考資料

図 515. 低電力タイマのブロック図（LPTIM3）

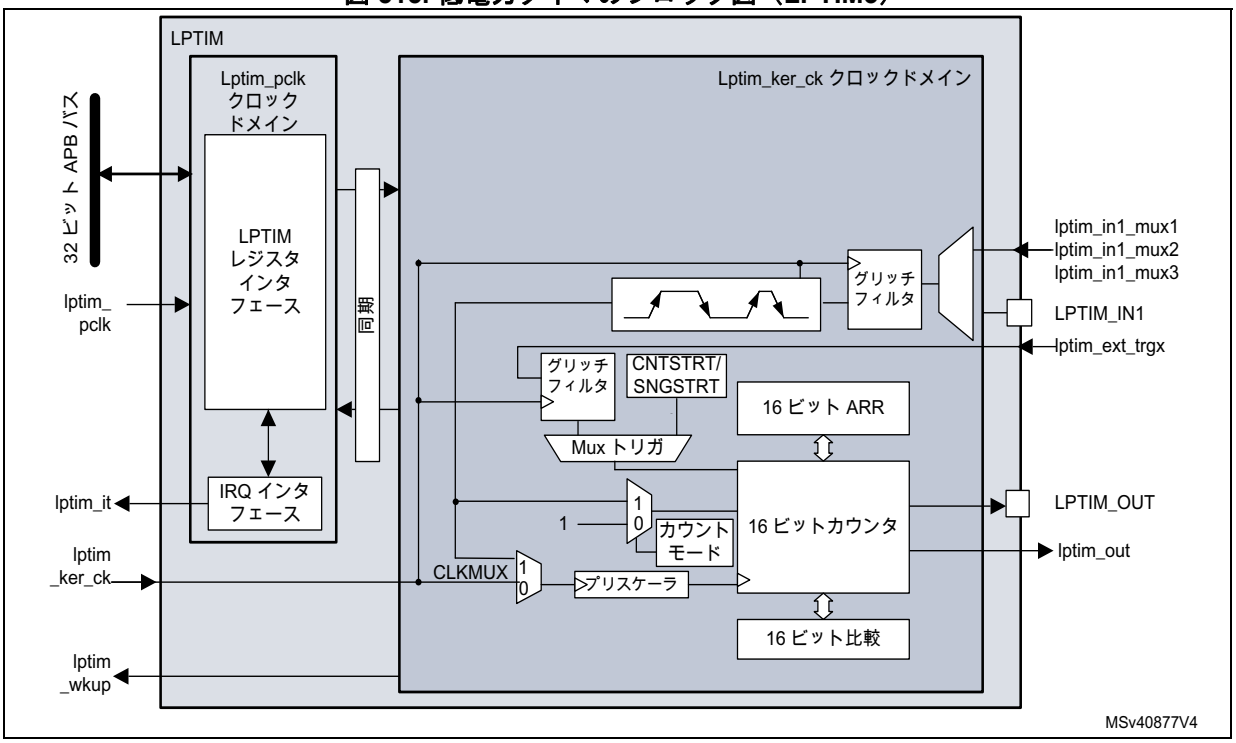
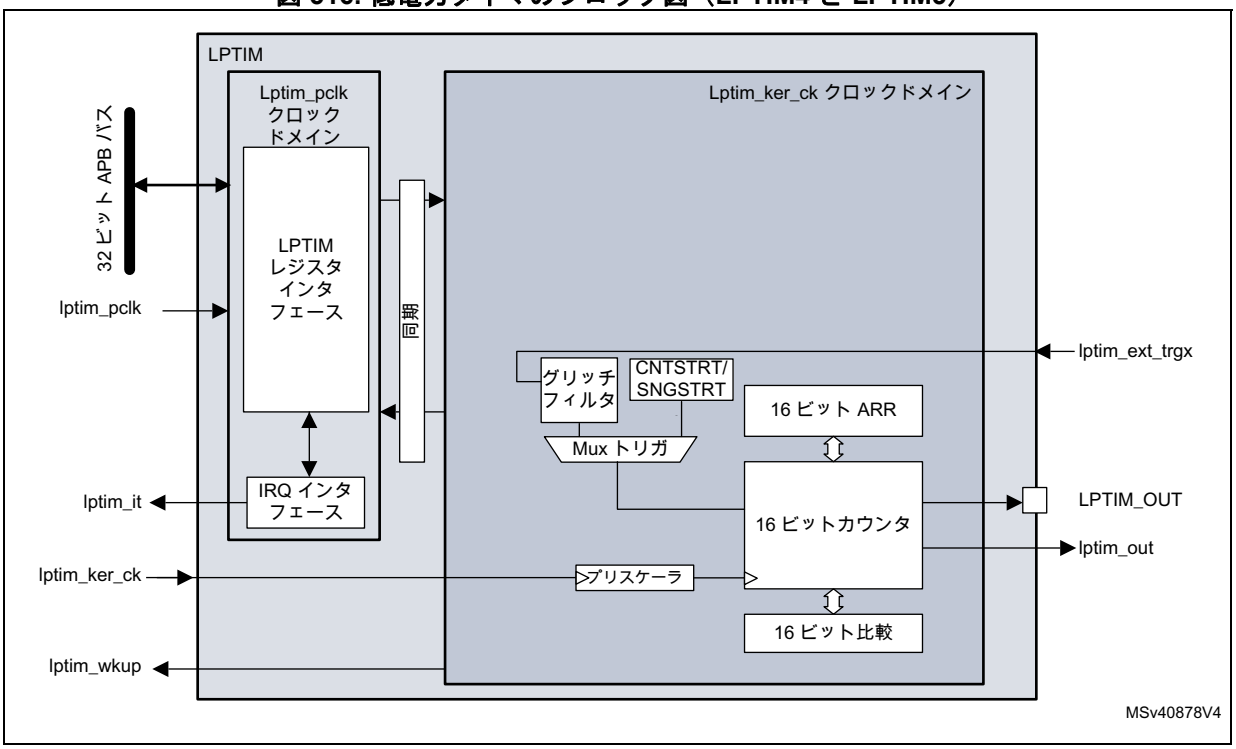


図 516. 低電力タイマのブロック図（LPTIM4 と LPTIM5）



参考資料

43.4.2 LPTIM ピンおよび内部信号

表 332 および *表 333* は、それぞれ LPTIM の内部信号と、ピンの一覧を示します。

表 332. LPTIM 内部信号

名前	信号タイプ	説明
lptim_pclk	デジタル入力	LPTIM APB クロックドメイン
lptim_ker_ck	デジタル入力	LPTIM カーネルクロック
lptim_in1_mux1	デジタル入力	マルチプレクサ入力 1 に接続される内部 LPTIM 入力 1
lptim_in1_mux2	デジタル入力	マルチプレクサ入力 2 に接続される内部 LPTIM 入力 1
lptim_in1_mux3	デジタル入力	マルチプレクサ入力 3 に接続される内部 LPTIM 入力 1
lptim_in2_mux1	デジタル入力	マルチプレクサ入力 1 に接続される内部 LPTIM 入力 2
lptim_in2_mux2	デジタル入力	マルチプレクサ入力 2 に接続される内部 LPTIM 入力 2
lptim_in2_mux3	デジタル入力	マルチプレクサ入力 3 に接続される内部 LPTIM 入力 2
lptim_ext_trgx	デジタル入力	LPTIM 外部トリガ入力 x
lptim_out	デジタル出力	LPTIM カウンタ出力
lptim_it	デジタル出力	LPTIM グローバル割り込み
lptim_wakeup	デジタル出力	LPTIM ウェイクアップイベント

43.4.3 LPTIM 入力およびトリガマッピング

表 334～*表 338* は LPTIM の外部トリガ接続を示し、*表 339*～*表 343* は LPTIM の入力 1 と入力 2 の接続を示します。

表 333. HRTIM の入出力ピン

名前	信号タイプ	説明
LPTIM_IN1	デジタル入力	GPIO ピンからマルチプレクサ入力 0 への LPTIM 入力 1
LPTIM_IN2	デジタル入力	GPIO ピンからマルチプレクサ入力 0 への LPTIM 入力 2
LPTIM_ETR	デジタル入力	LPTIM 外部トリガ GPIO ピン
LPTIM_OUT	デジタル出力	LPTIM 出力 GPIO ピン

表 334. LPTIM1 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM1_ETR のオルタネート機能としての GPIO ピン
lptim_ext_trig1	RTC_ALARM
lptim_ext_trig2	RTC_ALARMB
lptim_ext_trig3	RTC_TAMP1_OUT
lptim_ext_trig4	RTC_TAMP2_OUT
lptim_ext_trig5	RTC_TAMP3_OUT
lptim_ext_trig6	COMP1_OUT
lptim_ext_trig7	COMP2_OUT

表 335. LPTIM2 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_ETR のオルタネート機能としての GPIO ピン
lptim_ext_trig1	RTC_ALARM
lptim_ext_trig2	RTC_ALARMB
lptim_ext_trig3	RTC_TAMP1_OUT
lptim_ext_trig4	RTC_TAMP2_OUT
lptim_ext_trig5	RTC_TAMP3_OUT
lptim_ext_trig6	COMP1_OUT
lptim_ext_trig7	COMP2_OUT

表 336. LPTIM3 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_OUT
lptim_ext_trig1	未接続
lptim_ext_trig2	LPTIM4_OUT
lptim_ext_trig3	LPTIM5_OUT
lptim_ext_trig4	SAI1_FS_A
lptim_ext_trig5	SAI1_FS_B
lptim_ext_trig6	未接続
lptim_ext_trig7	未接続

表 337. LPTIM4 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_OUT
lptim_ext_trig1	LPTIM3_OUT
lptim_ext_trig2	未接続
lptim_ext_trig3	LPTIM5_OUT
lptim_ext_trig4	SAI2_FS_A
lptim_ext_trig5	SAI2_FS_B
lptim_ext_trig6	未接続
lptim_ext_trig7	未接続

表 338. LPTIM5 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_OUT
lptim_ext_trig1	LPTIM3_OUT
lptim_ext_trig2	LPTIM4_OUT

表 338. LPTIM5 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig3	SAI4_FS_A
lptim_ext_trig4	SAI4_FS_B
lptim_ext_trig5	未接続
lptim_ext_trig6	未接続
lptim_ext_trig7	未接続

表 339. LPTIM1 入力 1 接続

lptim_in1_mux	LPTIM1 入力 1 接続先
lptim_in1_mux0	LPTIM1_IN1 のオルタネート機能としての GPIO ピン
lptim_in1_mux1	COMP1_OUT
lptim_in1_mux2	未接続
lptim_in1_mux3	未接続

表 340. LPTIM1 入力 2 接続

lptim_in2_mux	LPTIM1 入力 2 接続先
lptim_int2_mux0	LPTIM1_IN2 のオルタネート機能としての GPIO ピン
lptim_in2_mux1	COMP2_OUT
lptim_in2_mux2	未接続
lptim_in2_mux3	未接続

表 341. LPTIM2 入力 1 接続

lptim_in1_mux	LPTIM2 入力 1 接続先
lptim_in1_mux0	LPTIM2_IN1 のオルタネート機能としての GPIO ピン
lptim_in1_mux1	COMP1_OUT
lptim_in1_mux2	COMP2_OUT
lptim_in1_mux3	COMP1_OUT または COMP2_OUT

表 342. LPTIM2 入力 2 接続

lptim_in2_mux	LPTIM2 入力 2 接続先
lptim_int2_mux0	LPTIM2_IN2 のオルタネート機能としての GPIO ピン
lptim_in2_mux1	COMP2_OUT
lptim_in2_mux2	未接続
lptim_in2_mux3	未接続

表 343. LPTIM3 入力 1 接続

lptim_in1_mux	LPTIM3 入力 1 接続先
lptim_in1_mux0	未接続
lptim_in1_mux1	SAI4_FS_A
lptim_in1_mux2	SAI4_FS_B
lptim_in1_mux3	未接続

43.4.4 LPTIM のリセットとクロック

LPTIM のクロックには、いくつかのクロックソースを使用できます。リセットおよびクロックコントローラ（RCC）によって APB、LSI、LSE、または HSI ソースから選択できる内部クロック信号を使用できます。また、LPTIM のクロックには、外部 Input1 に入力された外部クロック信号を使用することもできます。外部クロックソースでクロック供給されるときには、LPTIM は、次の 2 つの構成のいずれかで動作します。

- 最初の構成では、LPTIM のクロックは外部信号によって供給されますが、同時に、APB またはその他の埋め込みオシレータ（LSE、LSI、HSIなど）から内部クロック信号も LPTIM に供給されます。
- 2 番目の構成では、LPTIM のクロックは外部 Input1 を通じて外部クロックソースによってのみ供給されます。この構成は、低電力モードになった後、すべての埋め込みオシレータがオフになるときに、タイムアウト機能またはパルスカウンタ機能を実現するために使用されます。

CKSEL ビットおよび COUNTMODE ビットに書き込むことによって、LPTIM が外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOL ビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給されます（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の 4 倍以上である必要があります。

43.4.5 グリッチフィルタ

LPTIM 入力の外部入力（マイクロコントローラの GPIO にマップされたもの）または内部入力（内蔵コンパレータのように、チップレベルで他の内蔵ペリフェラルにマップされたもの）はデジタルフィルタによって保護され、グリッチとノイズの心配が LPTIM 内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースを LPTIM に供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

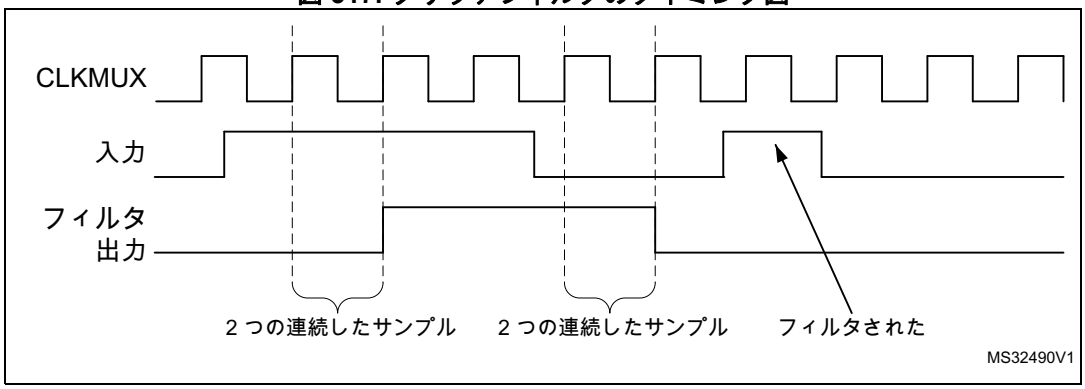
デジタルフィルタは、2 つのグループに分けられます。

- 最初のグループのデジタルフィルタは、LPTIM の外部入力を保護します。デジタルフィルタの感度は CKFLT ビットによって制御されます。
- 2 番目のグループのデジタルフィルタは、LPTIM の内部トリガ入力を保護します。デジタルフィルタの感度は TRGFLT ビットによって制御されます。

注： ***デジタルフィルタの感度は、グループごとに制御されます。同じグループ内の各デジタルフィルタの感度を個別に設定することはできません。***

フィルタの感度は、信号のレベル変化を有効な遷移とみなすために、LPTIM 入力の1 つで検出される連続した等しいサンプルの数に基づいて作用します。*図 517* に、2 つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図 517. グリッチフィルタのタイミング図



注： ***内部クロック信号が供給されない場合は、CKFLT および TRGFLT ビットを 0 にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIM の外部入力をグリッチから保護できます。***

43.4.6 プリスケーラ

LPTIM 16 ビットカウンタの前には、設定可能な 2 のべき乗プリスケーラがあります。プリスケーラの分周比は PRESC[2:0] 3 ビットフィールドによって制御されます。下の表に、可能な分周比を示します。

表 344. プリスケーラの分周比

プログラミング	分周比
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

43.4.7 トリガマルチプレクサ

LPTIM カウンタは、ソフトウェアによって、または 8 つのトリガ入力の 1 つのアクティブエッジの検出後に開始できます。

LPTIM のトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が 00 の場合、LPTIM カウンタは、CNTSTRT または SNGSTRT ビットがソフトウェアによってセットされるとすぐに開始します。
- TRIGEN[1:0] の残りの 3 つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIM カウンタは、アクティブエッジが検出されるとすぐに開始します。

TRIGEN[1:0] が 00 以外のときには、カウンタの開始に使用される 8 つのトリガ入力の 1 つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIM の場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに 2 カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： ***SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。***

43.4.8 動作モード

LPTIM には 2 つの動作モードがあります。

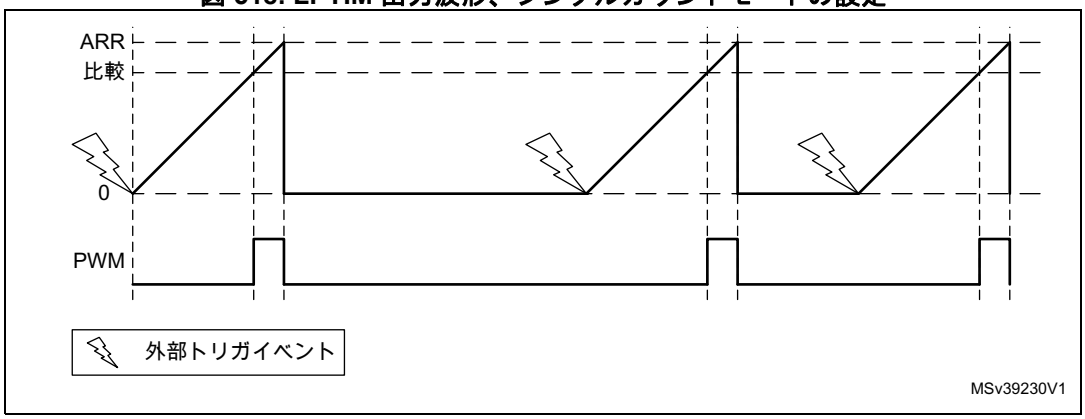
- 連続モード : タイマはフリーランニングし、トリガイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード : タイマはトリガイベントによって開始し、ARR 値に達すると停止します。

新しいトリガイベントが発生すると、タイマは再開始します。カウンタの開始後、カウンタが ARR に達する前に発生したトリガイベントは破棄されます。

ワンショットカウントを有効にするには、SNGSTRT ビットをセットする必要があります。

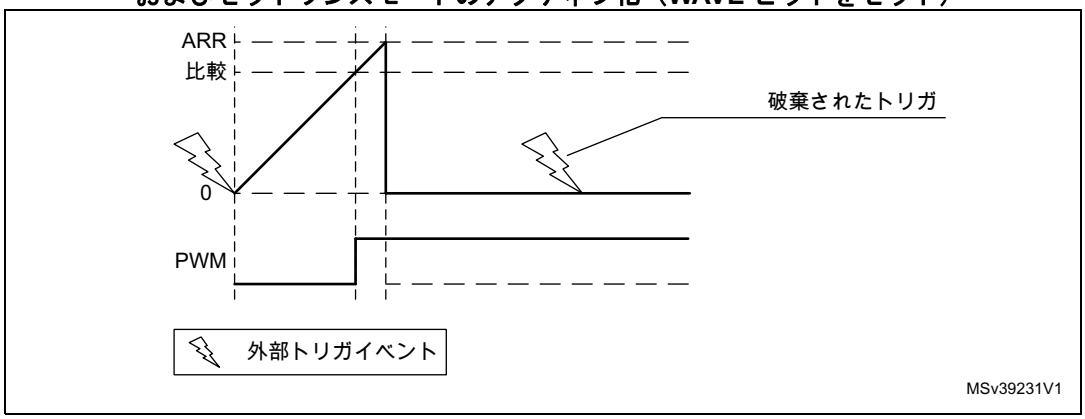
外部トリガが選択された場合、SNGSTRT ビットがセットされ、かつカウンタレジスタが停止された（ゼロ値を含む）後に各外部トリガイベントが着信すると、カウンタは新たなワンショットカウントサイクルを開始します（*図 518* を参照）。

図 518. LPTIM 出力波形、シングルカウントモードの設定



LPTIM_CFGR レジスタの WAVE ビットフィールドがセットされると、セットワンスモードがアクティブになることに注意してください。この場合、最初のトリガに続いて一度だけ開始され、その後発生したトリガイベントはすべて破棄されます（*図 518*を参照）。

図 519. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット）



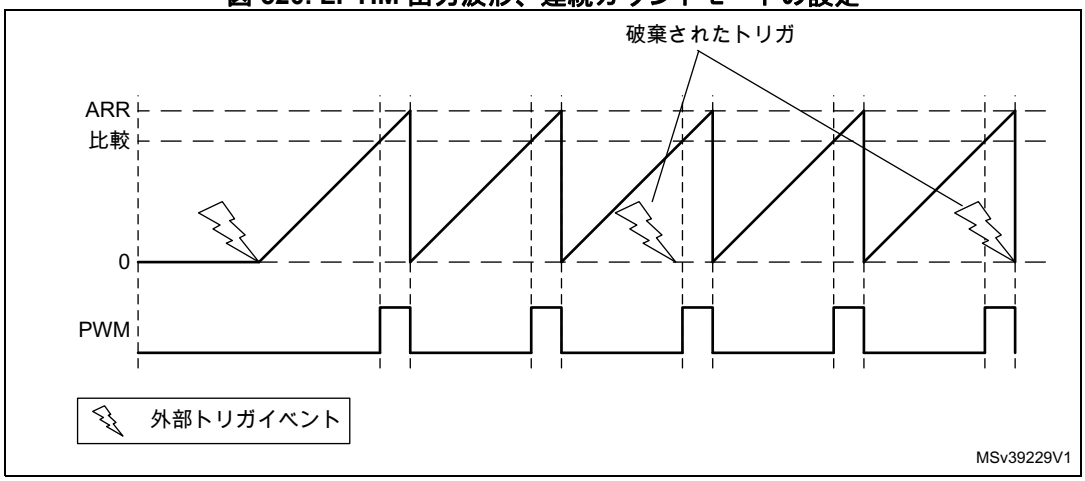
ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、SNGSTRT をセットすると、カウンタはワンショットカウントを開始します。

連続カウントを有効にするには、CNTSTRT ビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRT がセットされた後に外部トリガイベントが着信すると、カウンタは連続カウントを開始します。その後発生した外部トリガイベントはすべて、*図 520*に示すように破棄されます。

ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、CNTSTRT をセットすると、カウンタは連続カウントを開始します。

図 520. LPTIM 出力波形、連続カウントモードの設定



SNGSTRT および CNTSTRT ビットは、タイマが有効なときのみ（イネーブルビットが 1 にセットされている）、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRT をセットすると、LPTIM はワンショットモードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRT をセットすると、LPTIM は連続モードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに再開始します。

43.4.9 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIM カウンタをリセットできます。この機能は、TIMOUT ビットで制御されます。

最初のトリガイベントでタイマが開始し、その後のトリガイベントでカウントがリセットされ、タイマが再開始します。

低電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによって MCU がウェイクアップします。

43.4.10 波形生成

2 つの 16 ビットレジスタ LPTIM_ARR（自動再ロードレジスタ）と LPTIM_CMP（比較レジスタ）は、LPTIM 出力のいくつかの異なった波形を生成するために使用されます。

タイマは次の波形を生成できます。

- PWM モード : LPTIM 出力は、LPTIM_CMP レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにセットされます。LPTIM 出力は、LPTIM_ARR レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにリセットされます。
- ワンパルスモード : 出力波形は、最初のパルスについては PWM モードの波形と同様であり、その後、出力は永続的にリセットされます。
- セットワンスモード : 出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます（設定された出力極性に応じて）。

上記のモードでは、LPTIM_ARR レジスタの値は LPTIM_CMP レジスタの値より大きい必要があります。

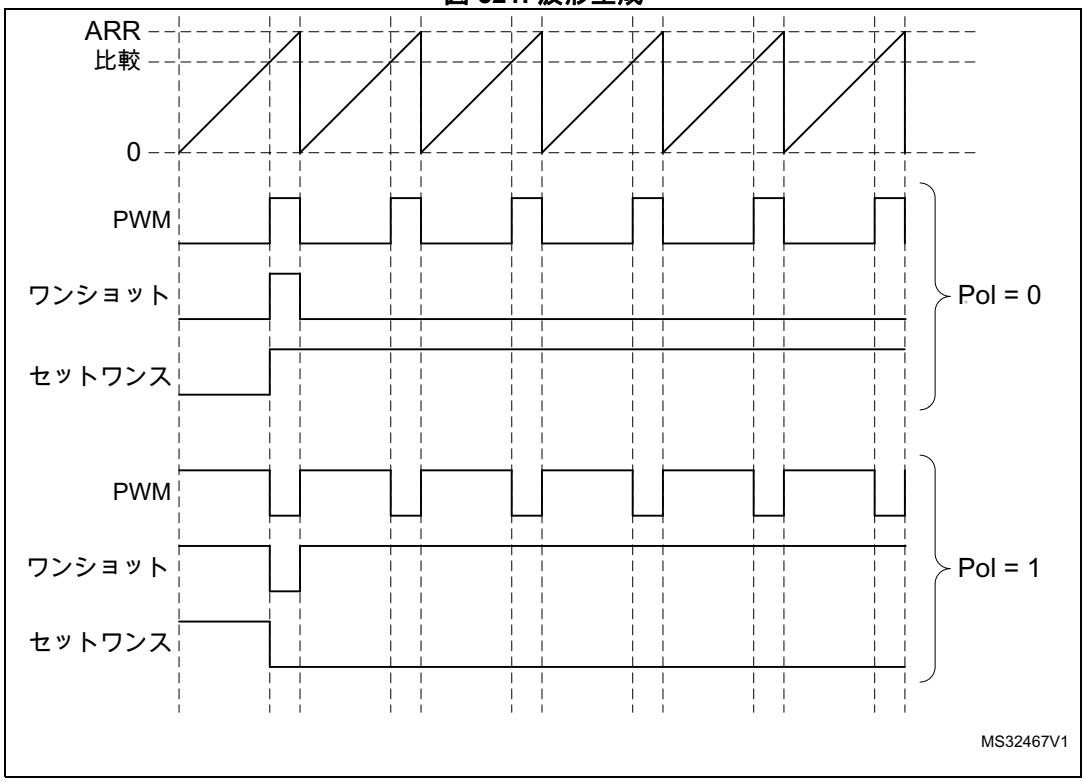
LPTIM 出力波形は、WAVE ビットによって次のように設定できます。

- WAVE ビットを 0 にリセットすると、LPTIM は CNTSTRT または SNGSTRT のセットされているビットに応じて、PWM 波形またはワンパルス波形のいずれかを生成します。
- WAVE ビットを 1 にセットすると、LPTIM はセットワンスモード波形を生成します。

WAVPOL ビットは、LPTIM 出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力はただちに変更されます。

LPTIM クロック周波数を 2 分周した周波数までの信号を生成できます。*図 521* に、LPTIM 出力で生成できる 3 つの波形を示します。また、WAVPOL ビットを使用して極性を変更したときの効果も示します。

図 521. 波形生成



43.4.11 レジスタの更新

LPTIM_ARR レジスタと LPTIM_CMP レジスタは、APB バス書き込み操作の直後に、またはタイマがすでに開始していた場合は現在の周期の終了時に更新されます。

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

- PRELOAD ビットが“0”にリセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは書き込みアクセスの直後に更新されます。
- PRELOAD ビットが“1”にセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは、タイマがすでに開始していた場合、現在の周期の終了時に更新されます。

LPTIM APB インタフェースと LPTIM カーネルロジックは異なるクロックを使用するので、APB 書き込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延中は、これらのレジスタへの追加の書き込みを避ける必要があります。

LPTIM_ISR レジスタの ARROK フラグと CMPOK フラグは、それぞれ、LPTIM_ARR レジスタと LPTIM_CMP レジスタへの書き込み操作が完了したことを示します。

LPTIM_ARR レジスタまたは LPTIM_CMP レジスタへの書き込みの後、同じレジスタへの新しい書き込み操作は、前の書き込み操作が完了してからでなければ実行できません。ARROK フラグまたは CMPOK フラグがセットされる前に連続した書き込みが行われると、予測不能な結果になります。

43.4.12 カウンタモード

LPTIM カウンタを使用して、LPTIM Input1 の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。CKSEL ビットおよび COUNTMODE ビットは、カウンタの更新にどのソースを使用するかを決定します。

LPTIM が Input1 の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジで更新できます。

CKSEL および COUNTMODE の値に応じて、以下に示すカウントモードを選択できます。

- CKSEL = 0：LPTIM のクロックは、内部クロックソースによって供給されます。
  - COUNTMODE = 0

    LPTIM が内部クロックソースによってクロック供給されるように設定され、LPTIM カウンタが LPTIM 外部 Input1 で検出されたアクティブエッジによって更新されるように設定されたときには、LPTIM に供給される内部クロックをプリスケールしてはなりません（PRESC[2:0] = 000）。
  - COUNTMODE = 1

    LPTIM 外部 Input1 は、LPTIM に供給される内部クロックでサンプリングされます。結果として、イベントをミスしないためには、外部 Input1 信号の変化の周波数が、LPTIM に供給される内部クロックの周波数を超えない必要があります。
- CKSEL = 1：LPTIM のクロックは、外部クロックソースによって供給されます。

  COUNTMODE の 値は無視されます。

  この構成では、LPTIM は内部クロックソースを必要としません（グリッチフィルタ有効である場合を除く）。LPTIM 外部 Input1 に入力された信号が LPTIM のシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。

  この構成の場合、LPTIM カウンタは、Input1 クロック信号の立ち上がりエッジまたは立ち下がりエッジで更新できますが、立ち上がりと立ち下がりの両方のエッジで更新することはできません。

  LPTIM 外部 Input1 に入力された信号は LPTIM カーネルロジックのクロック動作にも使用されるので、カウンタがインクリメントされる前（LPTIM が有効にされた後）、初期遅延があります。より正確には、LPTIM 外部 Input1 の（LPTIM が有効になった後の）最初の 5 つのアクティブエッジは失われます。

43.4.13 タイマ有効

LPTIM_CR レジスタのイネーブルビットは、LPTIM カーネルロジックを有効化／無効化するために使用されます。イネーブルビットをセットした後、LPTIM が実際に有効になるまで、2 カウンタクロックの遅延が必要です。

LPTIM_CFGR および LPTIM_IER レジスタの変更は、LPTIM が無効なときにのみ行う必要があります。

43.4.14 タイマカウンタのリセット

LPTIM_CNT レジスタの内容をゼロにリセットするために、以下の 2 つのリセットメカニズムが実装されています。

- 同期リセットメカニズム：同期リセットは、LPTIM_CR レジスタの COUNTRST ビットによって制御されます。COUNTRST ビットフィールドを“1”にセットした後、リセット信号が LPTIM カーネルクロックドメイン内に伝播されます。したがって、リセットが考慮される前に、LPTIM カーネルロジックの若干のクロックパルスが経過することに注意することが重要です。このため、リセットがトリガされてから有効になるまでの間に、LPTIM カウンタは若干の余分なパルスをカウントすることになります。COUNTRST ビットは APB クロックドメインに位置し、LPTIM カウンタは LPTIM カーネルクロックドメインに位置しているので、COUNTRST ビットに“1”を書き込んだ時に APB クロックによって発行されるリセット信号に同期するのに、カーネルクロックの 3 クロックサイクルの遅延が必要となります。
- 非同期リセットメカニズム：非同期リセットは、LPTIM_CR レジスタにある RSTARE ビットによって制御されます。このビットが“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読み出しアクセスによって、その内容がゼロにリセットされます。非同期リセットは、LPTIM コアクロックが提供されない時間枠内にトリガされるべきです。たとえば、LPTIM 入力 1 が外部クロックソースとして使用されているとき、非同期リセットは、LPTIM 入力 1 に反転が起きないという十分な備えがあるときにのみ適用すべきです。
  LPTIM_CNT レジスタの内容を信頼できるように読み出すためには、2 回の連続した読み出しアクセスを実行して比較する必要があることに注意してください。2 回の読み出しアクセスで得られた値が同じであるとき、各読み出しアクセスは信頼できると考えられます。しかし残念ながら、非同期リセットが有効になっているとき、LPTIM_CNT レジスタを 2 度読み出すことはできません。

警告： LPTIM 内部には、2 つのリセットメカニズムを同時に使用することを防ぐメカニズムはありません。したがって、開発者はこれらの 2 つのメカニズムを排他的に使用することを確実にするべきです。

43.4.15 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが 0 と LPTIM_ARR レジスタでプログラムされた自動再ロード値の間で（方向に応じて、0 から ARR まで、または ARR から 0 まで）連続的にカウントすることを意味します。したがって、開始前に LPTIM_ARR を設定する必要があります。Input1 と Input2 の 2 つの外部入力信号から、LPTIM カウンタのクロックのためのクロック信号が生成されます。この 2 つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIM が内部クロックソースからクロック供給されるときにのみ使用できます。Input1 と Input2 の両方の信号周波数は、LPTIM 内部クロック周波数を 4 分周したものを超えてはなりません。これは、LPTIM の正しい動作を保証するために必要です。

方向の変更は、LPTIM_ISR レジスタの Down と Up の 2 つのフラグによって通知されます。また、LPTIM_IER レジスタを通じて有効化された場合、両方の方向変更イベントで割り込みを生成できます。

エンコーダモードを有効にするには、ENC ビットを 1 にセットする必要があります。LPTIM を、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIM カウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウント方向は、Up および Down フラグによって通知され、エンコーダのロータの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ検出に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組み合わせを示します（Input1 と Input2 は同時に切り替わらないと想定しています）。

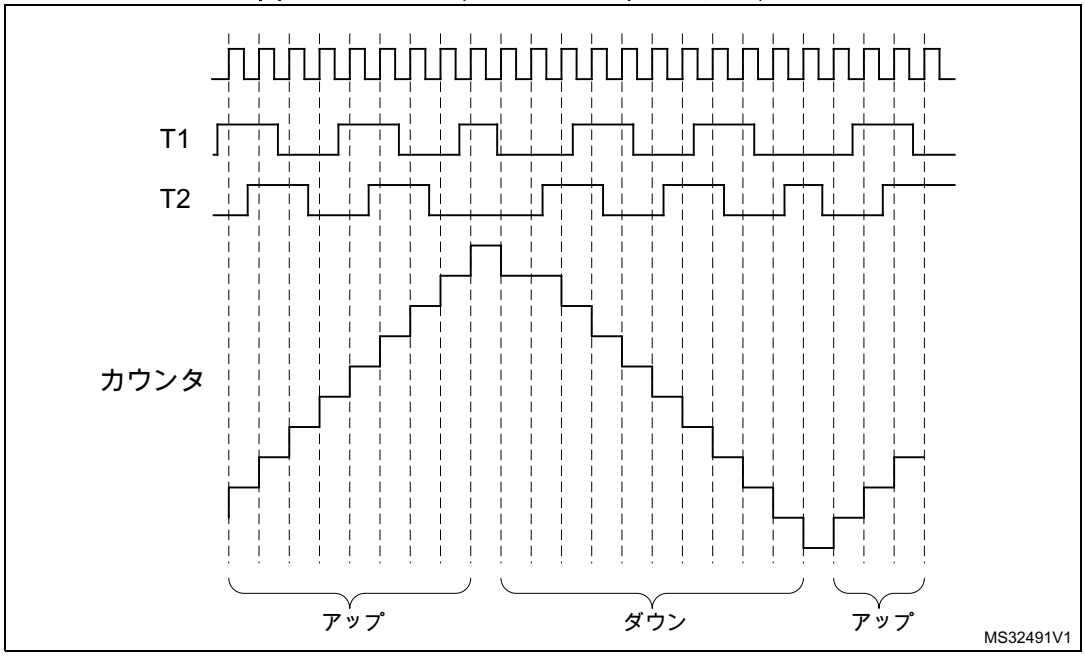
表 345. エンコーダのカウントシナリオ

アクティブエッジ	他方の信号のレベル（Input2 に対する Input1、Input1 に対する Input2）	Input1 信号		Input2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
立ち上がりエッジ	高	ダウン	カウントなし	アップ	カウントなし
	低	アップ	カウントなし	ダウン	カウントなし
立ち下がりエッジ	高	カウントなし	アップ	カウントなし	ダウン
	低	カウントなし	ダウン	カウントなし	アップ
両エッジ	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

次の図に、両方のエッジ検出が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： ***このモードでは、LPTIM のクロックは内部クロックソースによって供給される必要があるため、CKSEL ビットをリセット値（0）に維持する必要があります。また、プリスケーラの分周比はリセット値である 1 に等しくなければなりません（PRESC[2:0] ビットが 000 である必要があります）。***

図 522. エンコーダモードのカウントシーケンス



43.5 LPTIM 割り込み

LPTIM_IER レジスタで有効化されていた場合、次のイベントが発生すると、割り込み／ウェイクアップイベントが生成されます。

- 比較一致
- 自動再ロード一致（エンコーダモードの場合は方向にかかわらず）
- 外部トリガイベント
- 自動再ロードレジスタへの書き込み完了
- 比較レジスタへの書き込み完了
- 方向変更（エンコーダモード）、プログラム可能（アップ／ダウン／両方）

注： ***LPTIM_ISR レジスタ（ステータスレジスタ）の対応するフラグがセットされた後で LPTIM_IER レジスタ（割り込み有効レジスタ）のビットがセットされた場合、割り込みはアサートされません。***

表 346. 割り込みイベント

割り込みイベント	説明
比較一致	カウンタレジスタ（LPTIM_CNT）の内容が比較レジスタ（LPTIM_CMP）の内容と一致したときに、割り込みフラグが立ちます。
自動再ロード一致	カウンタレジスタ（LPTIM_CNT）の内容が自動再ロードレジスタ（LPTIM_ARR）の内容と一致したときに、割り込みフラグが立ちます。
外部トリガイベント	外部トリガイベントが検出されたときに、割り込みフラグが立ちます。
自動再ロードレジスタへの書き込み完了	LPTIM_ARR レジスタへの書き込み動作が完了したときに、割り込みフラグが立ちます。
比較レジスタへの書き込み完了	LPTIM_CMP レジスタへの書き込み動作が完了したときに、割り込みフラグが立ちます。
方向の変更	エンコーダモードで使用されます。方向の変更を通知するために次の 2 つの割り込みフラグが内蔵されています。 – UP フラグはアップカウント方向への変更を通知します。 – DOWN フラグはダウンカウント方向への変更を通知します。

43.6 LPTIM レジスタ

43.6.1 LPTIM 割り込みおよびステータスレジスタ（LPTIM_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWN ビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UP ビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 4 **ARROK**：自動再ロードレジスタ更新 OK

ARROK は、LPTIM_ARR レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 3 **CMPOK**：比較レジスタ更新 OK

CMPOK は、LPTIM_CMP レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIG は、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。

ビット 1 **ARRM**：自動再ロード一致

ARRM は、LPTIM_CNT レジスタの値が LPTIM_ARR レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 0 **CMPM**：比較一致

CMPM は、LPTIM_CNT レジスタの値が LPTIM_CMP レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

43.6.2 LPTIM 割り込みクリアレジスタ（LPTIM_ICR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
									w	w	w	w	w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの DOWN フラグがクリアされます。

ビット 5 **UPCF**：アップへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの UP フラグがクリアされます。

ビット 4 **ARROKCF**：自動再ロードレジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARROK フラグがクリアされます。

ビット 3 **CMPOKCF**：比較レジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMPOK フラグがクリアされます。

ビット 2 **EXTTRIGCF**：外部トリガ有効エッジフラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの EXTTRIG フラグがクリアされます。

ビット 1 **ARRMCF**：自動再ロード一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARRM フラグがクリアされます。

ビット 0 **CMPMCF**：比較一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMP フラグがクリアされます。

43.6.3 LPTIM 割り込み有効レジスタ（LPTIM_IER）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNIE**：ダウンへの方向変更割り込み有効化
- 0： DOWN 割り込み無効
- 1： DOWN 割り込み有効

ビット 5 **UPIE**：アップへの方向変更割り込み有効化
- 0： UP 割り込み無効
- 1： UP 割り込み有効

ビット 4 **ARROKIE**：自動再ロードレジスタ更新 OK 割り込み有効化
- 0： ARROK 割り込み無効
- 1： ARROK 割り込み有効

ビット 3 **CMPOKIE**：比較レジスタ更新 OK 割り込み有効化
- 0： CMPOK 割り込み無効
- 1： CMPOK 割り込み有効

ビット 2 **EXTTRIGIE**：外部トリガ有効エッジ割り込み有効化
- 0： EXTTRIG 割り込み無効
- 1： EXTTRIG 割り込み有効

ビット 1 **ARRMIE**：自動再ロード一致割り込み有効化
- 0： ARRM 割り込み無効
- 1： ARRM 割り込み有効

ビット 0 **CMPMIE**：比較一致割り込み有効化
- 0： CMPM 割り込み無効
- 1： CMPM 割り込み有効

注意： LPTIM_IER レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

43.6.4 LPTIM 設定レジスタ（LPTIM_CFGR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNT MODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN		Res.
							rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGSEL			Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ENC**：エンコーダモード有効化
ENC ビットは、エンコーダモードを制御します。
0： エンコーダモード無効
1： エンコーダモード有効

ビット 23 **COUNTMODE**：カウンタモード有効化
COUNTMODE ビットは、LPTIM がカウンタのクロックに使用するクロックソースを選択します。
0：カウンタは各内部クロックのパルスに従ってインクリメントされます。
1：カウンタは LPTIM 外部 Input1 の各有効なクロックパルスに従ってインクリメントされます。

ビット 22 **PRELOAD**：レジスタ更新モード
PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。
0： レジスタは、各 APB バス書き込みアクセス後に更新されます。
1： レジスタは、現在の LPTIM 周期の終了時に更新されます。

ビット 21 **WAVPOL**：波形極性
WAVPOL ビットは、出力の極性を制御します。
0： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果を反映します。
1： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果の逆を反映します。

ビット 20 **WAVE**：波形
WAVE ビットは、出力波形を制御します。
0： セットワンスモード、PWM／ワンパルス波形（OPMODE ビットに応じて）を非アクティブにします。
1： セットワンスモードをアクティブにします。

ビット 19 **TIMOUT**：タイムアウト有効化
TIMOUT ビットは、タイムアウト機能を制御します。
0：タイマがすでに開始しているときに着信したトリガイベントは無視されます。
1： タイマがすでに開始しているときにトリガイベントが着信すると、カウンタがリセットされ、再開始します。

ビット 18:17 **TRIGEN**：トリガ有効化および極性
TRIGEN ビットは､LPTIM カウンタが外部トリガによって開始されるかどうかを制御します｡外部トリガオプションが選択された場合、トリガのアクティブエッジについて 3 つの構成が可能です。
00：ソフトウェアトリガ（カウントの開始はソフトウェアによって行われます）。
01：立ち上がりエッジがアクティブエッジです。
10：立ち下がりエッジがアクティブエッジです。
11：両方のエッジがアクティブエッジです。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **TRIGSEL**：トリガセレクタ
TRIGSEL ビットは、次の 8 つのソースの中のLPTIM のトリガイベントの役目を果たすトリガソースを選択します。

000：lptim_ext_trig0
001：lptim_ext_trig1
010：lptim_ext_trig2
011：lptim_ext_trig3
100：lptim_ext_trig4
101：lptim_ext_trig5
110：lptim_ext_trig6
111：lptim_ext_trig7

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **PRESC**：クロックプリスケーラ

PRESC ビットは、プリスケーラ分周比を設定します。次の分周比から選択できます。

000 :/1
001 :/2
010 :/4
011 :/8
100 :/16
101 :/32
110 :/64
111 :/128

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **TRGFLT**：設定可能なトリガ用デジタルフィルタ

TRGFLT 値は、有効なレベル遷移とみなされる前に、内部トリガでレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：トリガのアクティブレベル変更は、有効なトリガとみなされます。
01: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。
10: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。
11: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **CKFLT**：設定可能な外部クロック用デジタルフィルタ

CKFLT 値は、有効なレベル遷移とみなされる前に、外部クロック信号でレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：外部クロック信号のレベル変更は、有効な遷移とみなされます。

01：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 2:1 **CKPOL**：クロック極性

LPTIM のクロックが、外部クロックソースによって供給される場合：

LPTIM のクロックが外部クロックソースによって供給されるとき、CKPOL ビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。

00：立ち上がりエッジが、カウントに使用されるアクティブエッジです。

01：立ち下がりエッジが、カウントに使用されるアクティブエッジです。

10：両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIM のクロックは内部クロックソースからも供給される必要があり、その周波数は外部クロック周波数の 4 倍以上である必要があります。

11：使用できません。

LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合：

00：エンコーダサブモード 1 がアクティブです。

01：エンコーダサブモード 2 がアクティブです。

10：エンコーダサブモード 3 がアクティブです。

エンコーダモードのサブモードの詳細については、*セクション 43.4.15：エンコーダモード*を参照してください。

ビット 0 **CKSEL**：クロックセレクタ

CKSEL ビットは、LPTIM が使用するクロックソースを選択します：

0： LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。

1： LPTIM のクロックは、LPTIM 外部 Input1 を通じて外部クロックソースによって供給されます。

注意： ***LPTIM_CFGR レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。***

43.6.5 LPTIM 制御レジスタ（LPTIM_CR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
											w	rs	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RSTARE**：読み出し後リセット有効化

このビットは、ソフトウェアによってセット／クリアされます。RSTARE が“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読み出しアクセスによって、LPTIM_CNT レジスタの内容が非同期的にリセットされます。

注意： ***このビットフィールドは書き込み専用です。つまり、このビットは書き込まれた値を検証するための読み戻しができません。例として、このビットを 1 にセットした場合、読み戻そうとすると、「読み出し後リセット」機能が有効になっていても 0 が返ってきます（このビットフィールドにすでに 1 が書き込まれているため）。「読み出し後リセット」をオフにしたり、オフになっていることを確認したりするには、すでに 0 が含まれている場合でも（0 をプログラムして）このビットをリセットする必要があります。***

ビット 3 **COUNTRST**：カウンタリセット

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。“1”にセットされると、このビットは LPTIM_CNT カウンタレジスタの同期リセットをトリガします。このリセットの同期的な性質のため、3 LP タイマコアクロックサイクルの同期的遅延の後にのみ、この状況が発生します（LP タイマコアクロック は APB クロックと異なる場合があります）。

注意： ***COUNTRST は、ハードウェアによって“0”にクリアされる前に、ソフトウェアによって“1”にセットしてはなりません。したがって、ソフトウェアで COUNTRST ビットを“1”にセットしようとする前に、それがすでに“0”にクリアされていることを確認するべきです。***

ビット 2 **CNTSTRT**：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM は連続モードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。

シングルパルスモードでのカウント中にこのビットがセットされた場合、LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときにタイマは停止せず、LPTIM カウンタは連続モードでのカウントを続行します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 1 **SNGSTRT**：シングルモードで LPTIM 開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM はシングルパルスモードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIMはシングルパルスモードで開始します。

LPTIM が連続カウントモードのときにこのビットがセットされた場合、LPTIM は LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときに停止します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 0 **ENABLE**：LPTIM 有効化

イネーブルビットは、ソフトウェアによってセット／クリアされます。

0 :LPTIM は無効です。

1 :LPTIM は有効です。

43.6.6 LPTIM 比較レジスタ（LPTIM_CMP）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP**：比較値

CMP は、LPTIM によって使用される比較値です。

LPTIM_CMP レジスタの内容の変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。

43.6.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）

アドレスオフセット：0x18

リセット値：0x0000 0001

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ARR**：自動再ロード値

ARR は、LPTIM の自動再ロード値です。

この値は、CMP[15:0] 値より大きくなければなりません。

LPTIM_ARR レジスタの内容の変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。

43.6.8 LPTIM カウンタレジスタ（LPTIM_CNT）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
r															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT**：カウンタ値

LPTIM が非同期クロックで実行しているとき、LPTIM_CNT レジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2 つの連続した読み出しアクセスを実行して、返された 2 つの値が同じかどうかを確認する必要があります。

信頼できる LPTIM_CNT レジスタの読み出しアクセスでは、2 回の連続した読み出しアクセスを実行して比較する必要があることに注意してください。2 回の連続読み出しアクセスで得られた値が同じであるとき、各読み出しアクセスは信頼できると考えられます。

43.6.9 LPTIM 設定レジスタ 2 （LPTIMx_CFGR2、x=1、2）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL		Res.	Res.	IN1SEL	
										rw	rw			rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **IN2SEL**：LPTIMx 入力 2 選択

IN2SEL ビットは LPTIMx 入力 2 マルチプレクサを制御し、LPTIMx 入力 2 を利用可能な入力のうちの 1 つに接続します。

00：lptim_in2_mux0

01：lptim_in2_mux1

10：lptim_in2_mux2

11：lptim_in2_mux3

接続の詳細は、LPTIM1 入力 2 の接続については*表 340：LPTIM1 入力 2 接続*を、LPTIM2 入力 2 の接続については*表 342：LPTIM2 入力 2 接続*を参照してください。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **IN1SEL**：LPTIMx 入力 1 選択

IN1SEL ビットは LPTIMx 入力 1 マルチプレクサを制御し、LPTIMx 入力 1 を利用可能な入力のうちの 1 つに接続します。

00：lptim_in1_mux0

01：lptim_in1_mux1

10：lptim_in1_mux2

11：lptim_in1_mux3

接続の詳細は、LPTIM1 入力 1 の接続については表 *表 339：LPTIM1 入力 1 接続* を参照し、
LPTIM2 入力 1 の接続については*表 341：LPTIM2 入力 1 接続*を参照してください。

注意： ***LPTIMx_CFGR2 レジスタの変更は、LPTIM が無効（イネーブルビットが"0"にリセットされている）のときにのみ行う必要があります。***

43.6.10 LPTIM3 設定レジスタ 2（LPTIM3_CFGR2）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN1SEL	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **IN1SEL**：LPTIM3 入力 1 選択

IN1SEL ビットは LPTIM3 入力 1 マルチプレクサを制御し、LPTIM3 入力 1 を利用可能な入力のうちの 1 つに接続します。

00：lptim_in1_mux0

01：lptim_in1_mux1

10：lptim_in1_mux2

11：lptim_in1_mux3

接続の詳細については、*表 343：LPTIM3 入力 1 接続* を参照してください。

注意： ***LPTIM3_CFGR2 レジスタの変更は、LPTIM が無効（イネーブルビットが"0"にリセットされている）のときにのみ行う必要があります。***

43.6.11 LPTIM レジスタマップ

次の表に LPTIM レジスタの一覧を示します。

表 347. LPTIM レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
	リセット値																										0	0	0	0	0	0	0
0x04	LPTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
	リセット値																										0	0	0	0	0	0	0
0x08	LPTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
	リセット値																										0	0	0	0	0	0	0
0x0C	LPTIM_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNTMODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN		TRIGSEL				Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL
	リセット値								0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0		0	0	0	0	0
0x10	LPTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	LPTIM_CMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	LPTIM_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	LPTIM_CNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	LPTIMx_CFGR2、x=1、2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL		Res.	Res.	IN1SEL	
	リセット値																											0	0			0	0
0x24	LPTIM3_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN1SEL	
	リセット値																															0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

44 システムウィンドウ型ウォッチドッグ（WWDG）

44.1 概要

システムウィンドウ型ウォッチドッグ（WWDG）は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。ウォッチドッグ回路は、T6 ビットがクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後に リセットを生成します。リセットは、ダウンカウンタがウィンドウレジスタ値に達する前に 7 ビットのダウンカウンタの値（制御レジスタ内）がリフレッシュされた場合にも生成されます。このことは、限られた時間枠（time-window）の間にカウンタがリフレッシュされなければならないことを意味します。

WWDG クロックは、APB クロックから分周され、また設定可能な時間枠（time-window）があるので、これをプログラムしてアプリケーション動作の異常な進み・遅れを検出できます。

WWDG は、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

44.2 WWDG の主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
  - ダウンカウンタの値が 0x40 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）（*図 524*を参照）。
- 早期ウェイクアップ割り込み（EWI）: ダウンカウンタが 0x40 になったときにトリガ（有効であり、ウォッチドッグがアクティブな場合）

44.3 WWDG の機能説明

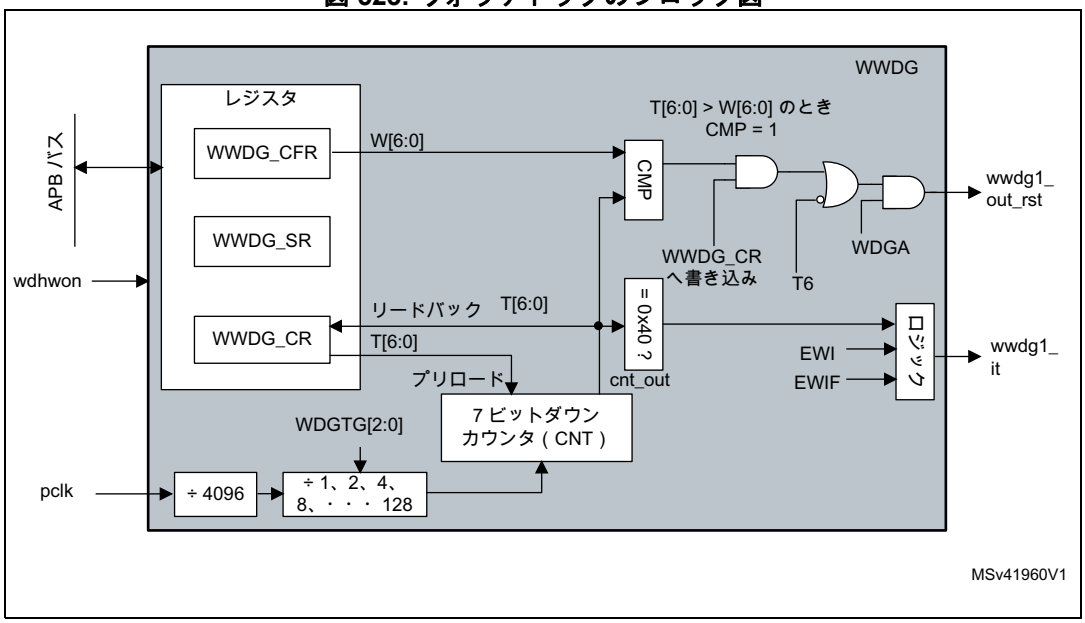
ウォッチドッグが有効（WWDG_CR レジスタの WDGA ビットがセットされている）な場合、7 ビットのダウンカウンタ（T[6:0] ビット）が 0x40 に達して 0x3F にデクリメントされた（T6 がクリアされた）時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

アプリケーションプログラムは、通常動作時には定期的に WWDG_CR レジスタへの書き込みを行って、リセットを防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値より小さいとき、かつ 0x3F より高いときに限られます。WWDG_CR レジスタに格納される値は、0xFF から 0xC0 の間でなければなりません。

WWDG のブロック図については、*図 523* および *セクション 44.3.2: WWDG 内部信号* を参照してください。

44.3.1 WWDG ブロック図

図 523. ウォッチドッグのブロック図



44.3.2 WWDG 内部信号

表 348 に WWDG の内部信号のリストを示します。

表 348. WWDG 内部入力／出力信号

信号名	信号タイプ	説明
pclk	デジタル入力	APB バスクロック
wwdg1_out_rst	デジタル出力	WWDG1 リセット信号出力
wwdg1_it	デジタル出力	WWDG1 割り込み出力

44.3.3 ウォッチドッグの有効化

ウォッチドッグはリセット後は常に無効です。これを有効にするには、WWDG_CR レジスタの WDGA ビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

44.3.4 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6 ビットをセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。このタイミングは、WWDG_CR レジスタへの書き込み時のプリスケーラの状態が不明なので最小値から最大値の間で変化します（*図 524* 参照）。設定レジスタ（WWDG_CFR）は、ウィンドウの上限値を含みます。リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x3F よりも大きい間にダウンカウンタを再ロードする必要があります。*図 524* に、ウィンドウ型ウォッチドッグのプロセスを示します。

注： ***T6 ビットを使用して、ソフトウェアリセットを生成することができます（WDGA ビットはセット、T6 ビットはクリアされる）。***

44.3.5 高度なウォッチドッグ割り込み機能

実際にリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割り込み（EWI）が使用できます。EWI 割り込みは、WWDG_CFR レジスタの EWI ビットをセットすることによって有効になります。ダウンカウンタ値が 0x40 に到達すると、EWI 割り込みが生成され、対応する割り込みサービスルーチン（ISR）を使用してデバイスをリセットする前に特定の処理（通信やデータロギングなど）をトリガすることができます。

アプリケーションによっては、EWI 割り込みを使用して、WWDG リセットを生成せずにソフトウェアのシステムチェックやシステム復旧／グレースフルデグラデーションを管理することができます。この場合、対応する割り込みサービスルーチン（ISR）で WWDG カウンタを再ロードし、WWDG リセットを回避してから必要な操作をトリガしてください。

EWI 割り込みは、WWDG_SR レジスタの EWIF ビットに“0”を書き込むことによってクリアされます。

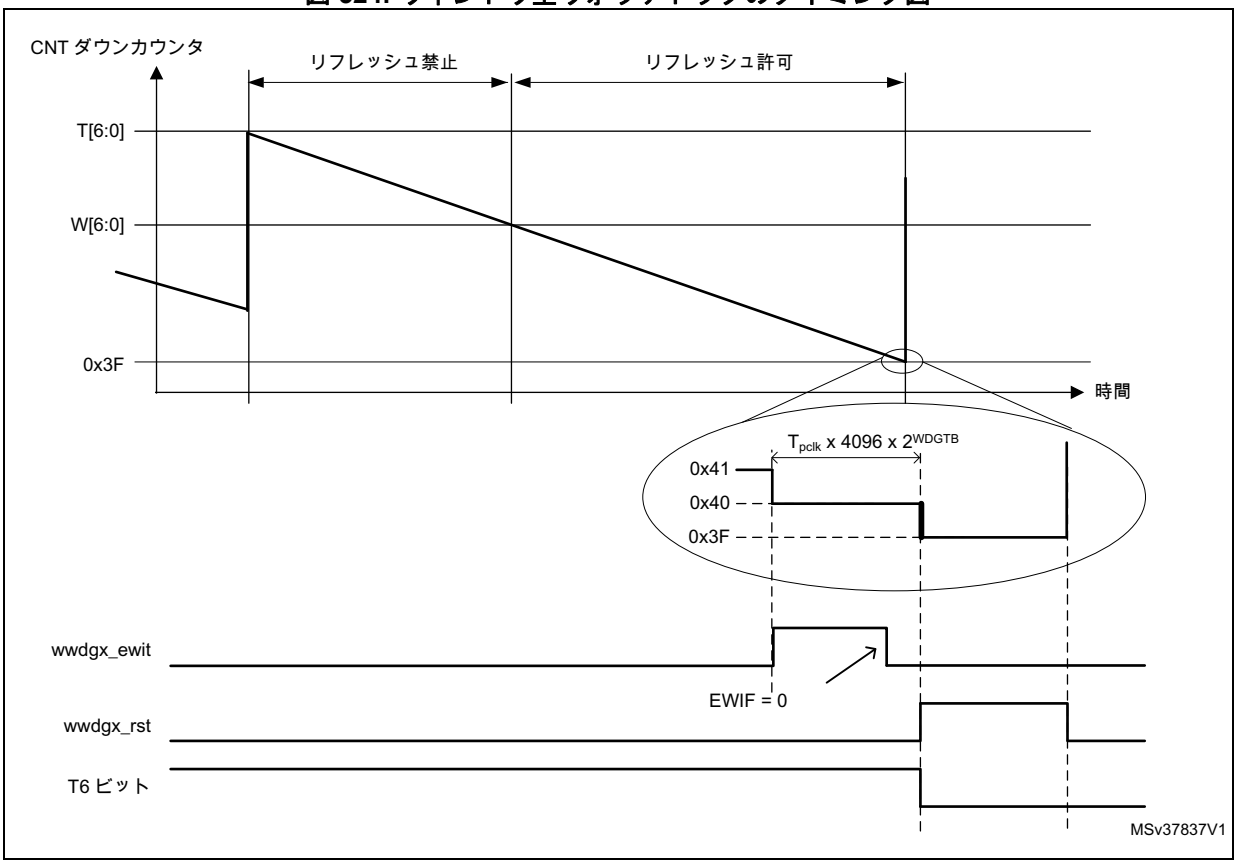
注： ***優先順位の高いタスクにおけるシステムロックなどによって EWI 割り込みが使用できない場合、最終的には WWDG リセットが生成されます。***

44.3.6 ウォッチドッグタイムアウトをプログラムする方法

図 524 の式を使用して、WWDG のタイムアウトを計算することができます。

警告： **WWDG_CR レジスタに書き込むときには、ただちにリセットされるのを防ぐために、常に T6 ビットに 1 を書き込んでください。**

図 524. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGTB[2:0]} \times (T[5:0] + 1) \quad (ms)$$

ここで、

t_{WWDG}：WWDG タイムアウト

t_{PCLK}：APB クロック周期の測定値（ms）

4096：内部分周器に対応する値

たとえば、APB 周波数が 48 MHz と等しい場合、WDGTB[2:0] は 3 にセットされ、T[5:0] は 63 にセットされます。

$$t_{WWDG} = 1 \S 48000 \times 4096 ¥ 2^{3} \times (63 + 1) = 43.69 \text{ ms}$$

t_{WWDG} の最小値と最大値については、データシートを参照してください。

44.3.7 デバッグモード

CPU がデバッグモードになると、DBGMCU_APB3LFZ1 に応じて、WWDG1 は通常どおりに動作を続けるか、または停止します。詳細については、*セクション 60：デバッグインフラストラクチャ*を参照してください。

44.4 .WWDG レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

44.4.1 制御レジスタ （WWDG_CR）

アドレスオフセット：0x00

リセット値：0x0000 007F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
								rs	rw						

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **WDGA**：有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。WDGA = 1 のとき、ウォッチドッグはリセットを生成できます。

0：ウォッチドッグは無効です。
1：ウォッチドッグは有効です。

ビット 6:0 **T[6:0]**：7 ビットカウンタ（MSB から LSB まで）

これらのビットにはウォッチドッグのカウンタの値が入っており、（4096 x $2^{WDGTB[2:0]}$）PCLK サイクルごとにデクリメントされます。0x40 に達して 0x3F にデクリメントされると（T6 がクリアされると）、リセットが生成されます。

44.4.2 設定レジスタ（WWDG_CFR）

アドレスオフセット：0x04

リセット値：0x0000 007F

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:11 **WDGTB[2:0]**：タイマーベース

プリスケーラのタイムベースは、次のように変更できます。

000：CK カウンタクロック（PCLK/4096）1 分周

001：CK カウンタクロック（PCLK/4096）2 分周

010：CK カウンタクロック（PCLK/4096）4 分周

011：CK カウンタクロック（PCLK/4096）8 分周

100：CK カウンタクロック（PCLK/4096）16 分周

101：CK カウンタクロック（PCLK/4096）32 分周

110：CK カウンタクロック（PCLK/4096）64 分周

111：CK カウンタクロック（PCLK/4096）128 分周

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EWI**：早期ウェイクアップ割り込み

このビットがセットされているときには、カウンタの値が 0x40 に達したときに割り込みが発生します。この割り込みは、リセット後にハードウェアによってのみクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **W[6:0]**：7 ビットウィンドウ値

これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

44.4.3 ステータスレジスタ（WWDG_SR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
															rc_w0

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EWIF**：早期ウェイクアップ割り込みフラグ

このビットは、カウンタの値が 0x40 に達したときにハードウェアによってセットされます。“0”を書き込んでソフトウェアでクリアする必要があります。“1”を書き込んでも、ビットの値は変化しません。このビットは、割り込みが有効でない場合にもセットされます。

44.4.4 WWDG レジスタマップ

次の表に、WWDG のレジスタマップとリセット値を示します。

表 349. WWDG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	WWDG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
	リセット値																									0	1	1	1	1	1	1	1
0x04	WWDG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGTB[1:0]			Res.	EWI	Res.	Res.	W[6:0]						
	リセット値																			0	0	0		0			1	1	1	1	1	1	1
0x08	WWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
	リセット値																																0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

45 独立型ウォッチドッグ（IWDG）

45.1 概要

デバイス は、内蔵ウォッチドッグペリフェラルを搭載しており、使用上、高い安全レベル、タイミングの正確さ、および柔軟性を兼ね備えています。独立型ウォッチドッグペリフェラルは、ソフトウェア障害による誤動作を検出および解決し、カウンタが所定のタイムアウト値に達すると、システムリセットをトリガします。

独立型ウォッチドッグ（IWDG）は、独自の低速クロック（LSI）によってクロック供給されるので、メインクロックに障害があってもアクティブなままです。

IWDG は、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、タイミング精度が低いという制約があります。ウィンドウ型ウォッチドッグの詳細については、*1814 ページのセクション 44*を参照してください。

45.2 IWDG の主な機能

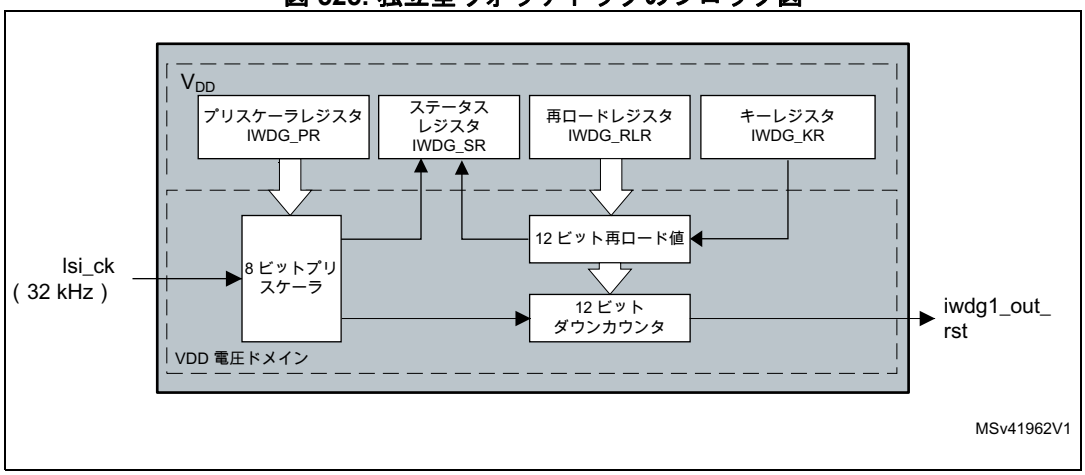
- フリーランニングダウンカウンタ
- 独立した RC オシレータからのクロック供給（STANDBY および STOP モードで動作可能）
- 条件付きリセット
  - ダウンカウンタの値が 0x000 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）。

45.3 IWDG の機能説明

45.3.1 IWDG ブロック図

図 525 に、独立型ウォッチドッグモジュールの機能ブロックを示します。

図 525. 独立型ウォッチドッグのブロック図



1. ウォッチドッグ機能は、STOP モードおよび STANDBY モードでも機能する V_{DD} 電圧ドメインに実装されています。

*キーレジスタ（IWDG_KR）*に値 0x0000 CCCC が書き込まれることによって独立型ウォッチドッグが開始すると、カウンタはリセット値 0xFFF からカウントダウンを開始します。カウント値の終わり（0x000）に達すると、リセット信号が生成されます（IWDG_reset）。

*キーレジスタ（IWDG_KR）*にキー値 0x0000 AAAA が書き込まれると、IWDG_RLR の値がカウンタに再ロードされ、ウォッチドッグのリセットが防止されます。

45.3.2 IWDG 内部信号

表 350 に IWDG の内部信号のリストを示します。

表 350. IWDG 内部入力／出力信号

信号名	信号タイプ	説明
lsi_ck	デジタル入力	LSI クロック
iwdg1_out_rst	デジタル出力	IWDG1 リセット信号出力

45.3.3 ウィンドウオプション

IWDG は、*ウィンドウレジスタ（IWDG_WINR）*に適切なウィンドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタが*ウィンドウレジスタ（IWDG_WINR）*に格納された値より大きい間に再ロード操作が行われると、リセットが生成されます。

*ウィンドウレジスタ（IWDG_WINR）*のデフォルト値は 0x0000 0FFF です。この値が更新されない場合は、ウィンドウオプションは無効にされます。

ウィンドウ値が変わるとすぐに再ロード操作が行われ、ダウンカウンタを*再ロードレジスタ（IWDG_RLR）*値にリセットし、次の再ロードを生成するためのサイクル数計算を容易にします。

ウィンドウオプションが有効な場合の IWDG の設定

1. *キーレジスタ（IWDG_KR）*に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. *キーレジスタ（IWDG_KR）*に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. *プリスケーラレジスタ（IWDG_PR）*を 0 から 7 までプログラムすることによって、IWDG のプリスケーラに書き込みを行います。
4. *再ロードレジスタ（IWDG_RLR）*に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. *ウィンドウレジスタ（IWDG_WINR）*に書き込みます。これにより、*再ロードレジスタ（IWDG_RLR）*のカウンタ値が自動的にリフレッシュされます。

注： ***ウィンドウ値を書き込むことで、ステータスレジスタ（IWDG_SR）が “0x0000 0000” にセットされた時点でカウンタ値を RLR でリフレッシュすることができます。***

ウィンドウオプションが無効な場合の IWDG の設定

ウィンドウオプションが使用されていない場合、IWDG は以下のように設定することができます。

1. *キーレジスタ（IWDG_KR）*に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. *キーレジスタ（IWDG_KR）*に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. *プリスケーラレジスタ（IWDG_PR）*を 0 から 7 までプログラムすることによって、プリスケーラに書き込みを行います。
4. *再ロードレジスタ（IWDG_RLR）*に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. カウンタ値を IWDG_RLR（IWDG_KR = 0x0000 AAAA）でリフレッシュします。

45.3.4 ハードウェアウォッチドッグ

デバイスのオプションビットを使って「ハードウェアウォッチドッグ」機能が有効化されると、ウォッチドッグは電源投入時に自動的に有効になり、カウンタがカウントの終わりに達する前にソフトウェアによって*キーレジスタ（IWDG_KR）*へ書き込まれない限り、またはダウンカウンタがウィンドウ外で再ロードされた場合は、リセットを生成します。

45.3.5 低電力モードでの取り扱い

IWDG_FZ_STOP および IWDG_FZ_STBY のオプション設定に応じて、IWDG では STOP モード中および STANDBY モード中にそれぞれカウントを継続または停止できます。STOP または STANDBY モード中に IWDG を実行したままにすると、このモードからデバイスをウェイクアップできます。詳細については、*Section : User and read protection option bytes* を参照してください。

45.3.6 STOP および STANDBY モードでの動作

一度起動すると、IWDG は停止できません。

45.3.7 レジスタのアクセス保護

プリスケーラレジスタ（IWDG_PR）、*再ロードレジスタ（IWDG_RLR）*、および*ウィンドウレジスタ（IWDG_WINR）*への書き込みアクセスは保護されます。これらを変更するには、まず、*キーレジスタ（IWDG_KR）*にコード 0x0000 5555 を書き込む必要があります。このレジスタに別の値を書き込むと、シーケンスがブレークされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作（0x0000 AAAA の書き込み）の場合に相当します。

ステータスレジスタは、プリスケーラの更新、あるいはダウンカウンタ再ロード値やウィンドウ値の更新が行われていることを示すために使用されます。

45.3.8 デバッグモード

マイクロコントローラがデバッグモードになると（コアは停止状態）、IWDG カウンタは、DBG モジュールの DBG_IWDG_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

45.4 IWDG レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

45.4.1 キーレジスタ（IWDG_KR）

アドレスオフセット：0x00

リセット値：0x0000 0000（STANDBY モードによりリセットされる）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **KEY[15:0]**：キー値（書き込み専用、読み出しでは 0x0000）

これらのビットには、ソフトウェアによって一定間隔でキー値 0xAAAA が書き込まれなければなりません。そうしないと、カウンタが 0 に達した時点でウォッチドッグがリセットを生成します。

キー値 0x5555 を書き込むことによって、IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへのアクセスが可能になります（*セクション 45.3.7：レジスタのアクセス保護*を参照）。

キー値 0xCCCC を書き込むと、ウォッチドッグが開始します（ハードウェアウォッチドッグオプションが選択されている場合を除く）。

45.4.2 プリスケーラレジスタ（IWDG_PR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PR[2:0]**：プリスケーラ分周回路

これらのビットは、書き込みアクセス保護されています（*セクション 45.3.7：レジスタのアクセス保護*を参照）。カウンタクロックを供給するプリスケーラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケーラ分周回路を変更できるようにするには、*ステータスレジスタ（IWDG_SR）*の PVU ビットがリセットされる必要があります。

000：4 分周
001：8 分周
010：16 分周
011：32 分周
100：64 分周
101：128 分周
110：256 分周
111：256 分周

注： ***このレジスタを読み出すと、V_{DD} 電圧ドメインからプリスケーラ値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、ステータスレジスタ（IWDG_SR）の PVU ビットがリセットされているときのみとなります。***

45.4.3 再ロードレジスタ（IWDG_RLR）

アドレスオフセット：0x08

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RL[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **RL[11:0]**：ウォッチドッグカウンタ再ロード値

これらのビットは、書き込みアクセス保護されています（*レジスタのアクセス保護*を参照）。*キーレジスタ（IWDG_KR）*に値 0xAAAA が書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とクロックプリスケーラによって決まります。タイムアウトに関する詳細はデータシートを参照してください。

再ロード値を変更できるようにするには、*ステータスレジスタ（IWDG_SR）*の RVU ビットがリセットされる必要があります。

注： ***このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、*ステータスレジスタ（IWDG_SR）*の RVU ビットがリセットされているときのみとなります。***

45.4.4 ステータスレジスタ（IWDG_SR）

アドレスオフセット：0x0C

リセット値：0x0000 0000（STANDBY モードによりリセットされません）

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **WVU**：ウォッチドッグカウンタウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

ウィンドウ値は、WVU ビットがリセットされているときのみ更新できます。

このビットは、一般のウィンドウが 1 の場合のみ生成されます。

ビット 1 **RVU**：ウォッチドッグカウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

再ロード値は、RVU ビットがリセットされているときのみ更新できます。

ビット 0 **PVU**：ウォッチドッグプリスケーラ値の更新

このビットは、プリスケーラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインでプリスケーラ更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

プリスケーラ値は、PVU ビットがリセットされているときのみ更新できます。

注： ***複数の再ロード値、プリスケーラ値、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前に RVU ビットがリセットされるまで待つか、プリスケーラ値を変更する前に PVU ビットがリセットされるまで待つか、またはウィンドウ値を変更する前に WVU ビットがリセットされるまで待つ必要があります。ただし、プリスケーラ値、再ロード値、またはウィンドウ値を更新した後は、RVU、PVU 、または WVU がリセットされるのを待たずに、コード実行を続けることができます（低電力モードに入った場合を除く）。***

45.4.5 ウィンドウレジスタ（IWDG_WINR）

アドレスオフセット：0x10

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WIN[11:0]**：ウォッチドッグカウンタウィンドウ値

これらのビットは書き込みアクセス保護されており（*セクション 45.3.7* を参照）、ダウンカウンタと比較されるウィンドウ値の上限を格納しています。

リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x0 よりも大きい間にダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、*ステータスレジスタ（IWDG_SR）* の WVU ビットがリセットされる必要があります。

注： ***このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、ステータスレジスタ（IWDG_SR）の WVU ビットがリセットされているときのみとなります。***

45.4.6 IWDG レジスタマップ

次の表に、IWDG レジスタマップとリセット値を示します。

表 351. IWDG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	IWDG_KR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	IWDG_PR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[2:0]		
	リセット値																														0	0	0
0x08	IWDG_RLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RL[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	IWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WVU	RVU	PVU
	リセット値																														0	0	0
0x10	IWDG_WINR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WIN[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

46 リアルタイムクロック（RTC）

46.1 概要

RTC は、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック（RTC）は、独立した BCD タイマ／カウンタです。RTC は、プログラム可能なアラーム割り込み機能を備えた時刻クロック／カレンダを搭載しています。

また、割り込み機能を備えたプログラム可能な周期的ウェイクアップフラグも搭載しています。

2 つの 32 ビットレジスタには、2 進化 10 進数形式（BCD）で表現した秒、分、時（12 時間または 24 時間形式）、曜日、日、月、年が含まれています。サブセカンドの値もバイナリ形式で利用できます。

28 日、29 日（うるう年）、30 日、31 日の補正は、自動的に行われます。サマータイム補正も行われます。

サブセカンド、秒、分、時、曜日、日付のプログラム可能なアラームを備えた 32 ビットレジスタが追加されています。

クリスタルオシレータ精度の偏差を補正するために、デジタル較正機能が利用可能です。

Backup ドメインリセット後、すべての RTC レジスタは、起こりうる不要な書き込みアクセスから保護されます。

供給電圧が動作範囲内にある間は、デバイスのステータス（実行モード、低電力モード、またはリセット中）に関係なく、RTC が停止することはありません。

46.2 RTC の主な機能

RTC ユニットの主な機能を以下に示します（*図 527：詳細な RTC ブロック図*を参照）。

- サブセカンド、秒、分、時（12 または 24 時間形式）、曜日、日、月、年に対応するカレンダ。
- ソフトウェアでプログラム可能なサマータイム補正。
- 割り込み機能を備えたプログラム可能なアラーム。アラームは、カレンダ項目のどの組み合わせでもトリガ可能。
- 自動ウェイクアップ割り込みをトリガする周期的なフラグを生成する自動ウェイクアップユニット。
- リファレンスクロック検出：より正確な秒のクロックソース（50 または 60 Hz）の使用で、カレンダの精度を向上。
- サブセカンドシフト機能を使用する外部クロックとの正確な同期。
- デジタル較正回路（周期的なカウンタ修正）：数秒の較正時間範囲で得られる 0.95 ppm の精度。
- イベントを保存するタイムスタンプ機能。
- 設定可能なフィルタおよび内部プルアップのあるタンパ検出イベント。
- マスク可能な割り込み／イベント：
  - アラーム A
  - アラーム B
  - ウェイクアップ割り込み
  - タイムスタンプ
  - タンパ検出
- 32 バックアップレジスタ。

46.3 RTC の機能説明

46.3.1 RTC ブロック図

図 526. RTC ブロックの概要

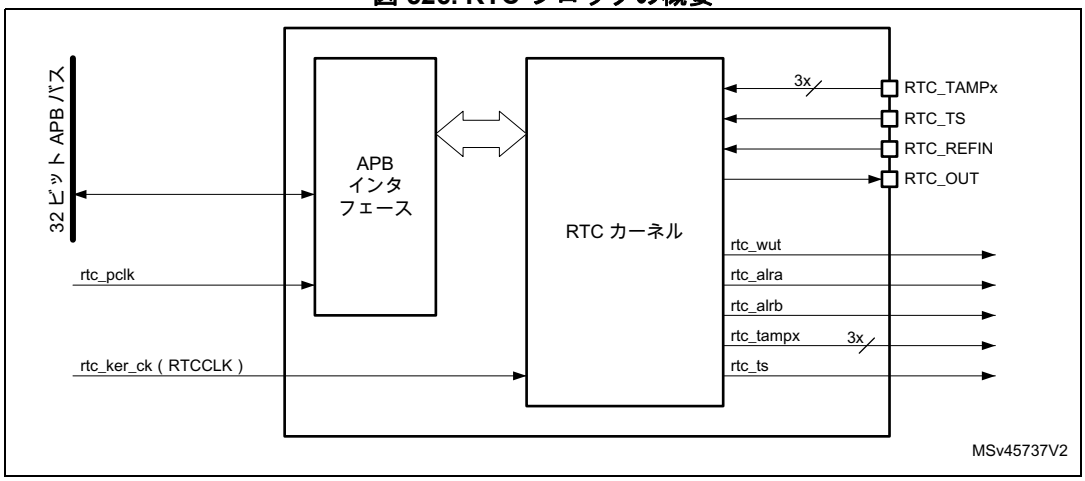


図 527. 詳細な RTC ブロック図

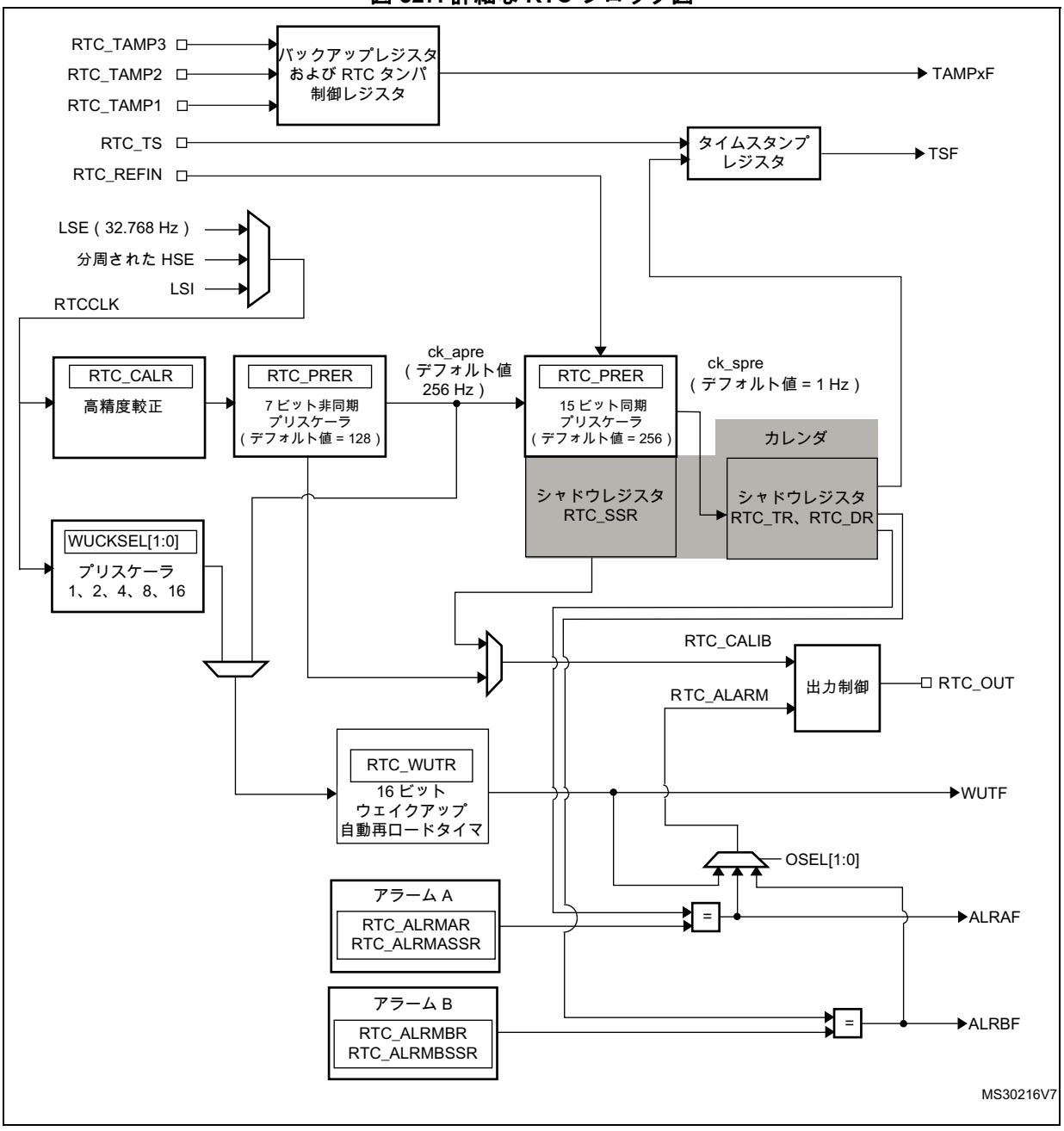
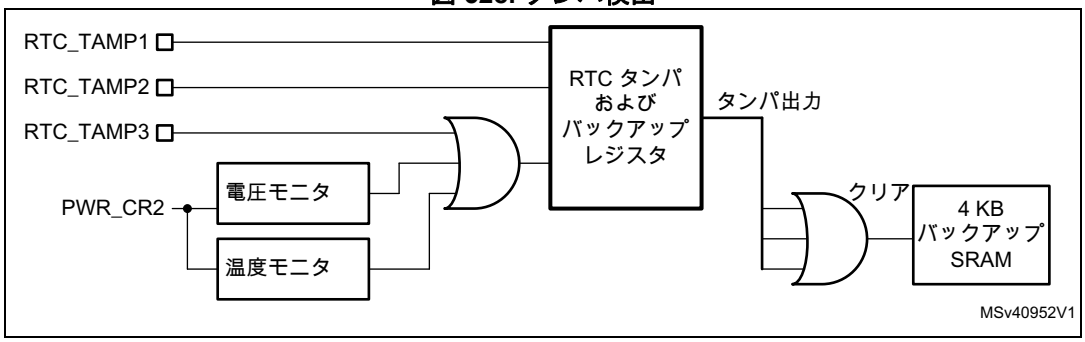


図 528. タンパ検出



RTC には次の要素が含まれます。

- 2 本のアラーム
- I/O からの 3 つのタンパイベント
  - タンパ検出でバックアップレジスタおよびバックアップ RAM は消去されます。
  - さらに、タンパ検出によるバックアップ SRAM の消去が終了するまで、ソフトウェアによるバックアップ SRAM へのアクセスが禁止されます。*セクション 46.3.15：タンパ検出* を参照してください。
  - I/O のイベント、または RTC 電源供給ドメインの電圧超過または不足、または高温または低温の検出のいずれかによって、タンパ 3 イベント検出が生成されます。これらの電圧および温度モニタ検出は*PWR 制御レジスタ 2（PWR_CR2）*で有効にします。
- I/O からの 1 つのタイムスタンプイベント
- タンパイベント検出によってタイムスタンプイベントを生成できます。
- V_{BAT} への切り替え発生時にタイムスタンプを生成できます。
- 32x 32 ビットバックアップレジスタ
  - バックアップレジスタ（RTC_BKPxR）は、VDD 電源が遮断された場合に VBAT によって電源が供給される、RTC ドメインに搭載されています。
- 出力機能：次の 2 つの出力のうち 1 つを選択する RTC_OUT です。
  - RTC_CALIB：512 Hz または 1Hz のクロック出力（LSE 周波数 32.768 kHz の場合）。この出力は、RTC_CR レジスタの COE ビットをセットして有効にします。
  - RTC_ALARM：この出力は、アラーム A、アラーム B、またはウェイクアップ出力を選択する RTC_CR レジスタの OSEL [1:0] ビットを設定することで有効にします。
- 入力機能：
  - RTC_TS：タイムスタンプイベント
  - RTC_TAMP1：タンパ 1 イベント検出
  - RTC_TAMP2：タンパ 2 イベント検出
  - RTC_TAMP3：タンパ 3 イベント検出
  - RTC_REFIN：50 または 60 Hz のリファレンスクロック入力

46.3.2 RTC ピンおよび内部信号

表 352. RTC ピンおよび内部信号

信号名	信号タイプ	説明
RTC_TS	入力	タイムスタンプ入力
RTC_TAMPx（x = 1、2、3）	入力	タンパ入力
RTC_REFIN	入力	リファレンスクロック入力
RTC_OUT	出力	RTC 出力
rtc_ker_ck（*RTCCLK*）	内部入力	RTC クロックソース（LSE クロック、LSI クロック、および HSE クロック）
rtc_pclk	内部入力	RTC APB インタフェースクロック
rtc_wut	内部出力	オンチップペリフェラル用の RTC ウェイクアップイベント出力
rtc_alra	内部出力	オンチップペリフェラル用の RTC アラーム A イベント出力
rtc_alrb	内部出力	オンチップペリフェラル用の RTC アラーム B イベント出力
rtc_tampx	内部出力	オンチップペリフェラル用の RTC タンパ [1～3] イベント出力
rtc_ts	内部出力	オンチップペリフェラル用の RTC タイムスタンプイベント出力

46.3.3 RTC によって制御される GPIO

RTC_OUT、RTC_TS、および RTC_TAMP1 は、同一ピン（PC13）上に配置されます。PC13 ピンの設定は、RTC_ALARM 出力のオープンドレインモードを除いて、PC13 GPIO の設定にかかわらず、RTC によって制御されます。PC13 に配置された RTC 機能はすべての低電力モードおよび VBAT モードで使用できます。

出力形式は *表 353*に示す優先順位に従います。

表 353. RTC ピン PC13 の設定(1)

PC13 ピンの設定と機能	OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	RTC_ALARM_TYPE ビット	TAMP1E ビット（RTC_TAMP1 入力イネーブル）	TSE ビット（RTC_TS 入力イネーブル）
RTC_ALARM 出力 OD	01、10、または 11	無視	0	0	無視	無視
			1			
RTC_ALARM 出力 PP	01、10、または 11	無視	0	1	無視	無視
			1			
RTC_CALIB 出力 PP	00	1	0	無視	無視	無視
RTC_TAMP1 入力フローティング	00	0	無視	無視	1	0
	00	1	1			
	01、10、または 11	0				

表 353. RTC ピン PC13 の設定(1)（続き）

PC13 ピンの設定と機能	OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	RTC_ALARM_TYPE ビット	TAMP1E ビット（RTC_TAMP1 入力イネーブル）	TSE ビット（RTC_TS 入力イネーブル）
RTC_TS および RTC_TAMP1 入力フローティング	00	0	無視	無視	1	1
	00	1	1			
	01、10、または 11	0				
RTC_TS 入力フローティング	00	0	無視	無視	0	1
	00	1	1			
	01、10、または 11	0				
ウェイクアップピンまたは標準 GPIO	00	0	無視	無視	0	0
	00	1	1			
	01、10、または 11	0				

1. OD：オープンドレイン、PP：プッシュプル

さらに、RTC_OUT_RMP ビットにより、RTC_OUT を PB2 ピンに再配置することも可能です。この場合、PB2 GPIO レジスタを適切なタイプのオルタネート機能として設定してください。再配置機能は *表 354*に示されています。

表 354. RTC_OUT の配置

OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	PC13 上の RTC_OUT	PB2上の RTC_OUT
00	0	0	-	-
00	1		RTC_CALIB	-
01、10、または 11	無視		RTC_ALARM	-
00	0	1	-	-
00	1		-	RTC_CALIB
01、10、または 11	0		-	RTC_ALARM
01、10、または 11	1		RTC_ALARM	RTC_CALIB

次の表は、すべてのモードでの RTC ピンと機能の一覧です。

表 355. RTC 機能（モード共通）

ピン	RTC 機能	すべての低電力モードで使用可能な機能（STANDBY モードを除く）	STANDBY モードで使用可能な機能	VBAT モードで使用可能な機能
PC13	RTC_TAMP1 RTC_TS RTC_OUT	はい	はい	はい
PI8	RTC_TAMP2	はい	はい	はい
PC1	RTC_TAMP3	はい	はい	はい
PB2	RTC_OUT	はい	いいえ	いいえ
PB15	RTC_REFIN	はい	いいえ	いいえ

46.3.4 クロックとプリスケーラ

RTC クロックソース（RTCCLK）は、LSE クロック、LSI オシレータクロック、HSE クロックのうちから、クロックコントローラを介して選択されます。RTC クロックソースの設定に関する詳細は、*セクション 8：リセットおよびクロック制御（RCC）*を参照してください。

プログラム可能なプリスケーラステージで、カレンダの更新に使用する1 Hz のクロックを生成します。消費電力を最少に抑えるため、プリスケーラは以下に示す 2 つのプログラム可能なプリスケーラに分割されます（*図 527：詳細な RTC ブロック図*を参照）。

- RTC_PRER レジスタの PREDIV_A ビットで設定される 7 ビットの非同期プリスケーラ
- RTC_PRER レジスタの PREDIV_S ビットで設定される 15 ビットの同期プリスケーラ

注： ***両方のプリスケーラを使用する場合は、非同期プリスケーラを高い値に設定して消費を最低限に抑えることをお勧めします。***

LSE 周波数 32.768 kHz で 1 Hz （ck_spre）の内部クロック周波数を得るため、非同期プリスケーラ分周比は 128、同期プリスケーラの分周比は 256 に設定されます。

最低分周比は 1、最大分周比は 2^{22}です。

これは、約 4 MHz の最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apre クロックは、サブセカンドダウンカウンタであるバイナリ RTC_SSR にクロックを供給するために使用されます。値がゼロになると、RTC_SSR は、PREDIV_S の内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spre クロックは、カレンダの更新に、または 16 ビットウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16 ビットウェイクアップ自動再ロードタイマを、プログラム可能な 4 ビット非同期プリスケーラで分周した RTCCLK で動作させることもできます（詳細は *セクション 46.3.7：周期的自動ウェイクアップ* を参照）。

46.3.5 リアルタイムクロックとカレンダ

RTC カレンダの時刻および日付レジスタには、PCLK（APB クロック）と同期するシャドウレジスタからアクセスします。同期するための待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用 RTC_SSR
- 時刻用 RTC_TR
- 日付用 RTC_DR

RTCCLK 2 サイクルごとに現在のカレンダ値がシャドウレジスタにコピーされ、RTC_ISR レジスタの RSF ビットがセットされます（*セクション 46.6.4：RTC 初期化とステータスレジスタ（RTC_ISR）* を参照）。STOP モードおよび STANDBY モードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 2 サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CR レジスタの BYPSHAD 制御ビットをセットすることにより、カレンダレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TR または RTC_DR レジスタを BYPSHAD = 0 の状態で読み出す際は、APB クロックの周波数 （f_{APB}）は、RTC クロック（f_{RTCCLK}）の周波数の 7 倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

46.3.6 プログラム可能なアラーム

RTC ユニットは、以下に示すプログラム可能なアラーム、アラーム A およびアラーム B を搭載しています。以下に示すのはアラーム A に関する説明ですが、アラーム B についても同様です。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAE ビットを通じて有効にします。ALRAF は、カレンダのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASSR および RTC_ALRMAR にプログラムされている値と一致する場合は 1 にセットされます。各カレンダ項目は、RTC_ALRMAR レジスタの MSKx ビットおよび RTC_ALRMASSR レジスタの MASKSSx ビットで個別に選択できます。アラームの割り込みは、RTC_CR レジスタの ALRAIE ビットを通じて有効にします。

注意： ***秒の項目が選択されている（RTC_ALRMAR で MSK1 ビットがリセットされている）場合、正しい動作を保証するため、RTC_PRER レジスタでセットされる同期プリスケーラの分周比は 3 以上でなければなりません。***

アラーム A および アラーム B（RTC_CR レジスタの OSEL[0:1] ビットで有効になっている場合）は、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

46.3.7 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16 ビットのプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は 17 ビットまで拡張できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットを通じて有効にします。

ウェイクアップタイマクロック入力には、次のものが使用できます。

- 2、4、8、または 16 分周した RTC クロック（RTCCLK）

  RTCCLK が LSE（32.768kHz）である場合、最小分解能 61 µs で、ウェイクアップ割り込み周期を 122 µs から 32 s の範囲で設定できます。
- ck_spre（通常は 1 Hz の内部クロック）

  ck_spre 周波数が 1 Hz の場合、1 秒の分解能でウェイクアップ時間を 1 秒 からおよそ 36 時間までの範囲で設定できます。このプログラム可能な広い時間範囲は、2 つの部分に分かれます。
  - WUCKSEL [2:1] = 10 の場合は 1 秒から 18 時間、
  - WUCKSEL[2:1] = 11 の場合は約 18 時間から 36 時間です。後者の場合、16 ビットカウンタの現在値に 216 が加算されます。初期化シーケンスが完了すると（*1840 ページのウェイクアップタイマのプログラミング*を参照）、タイマがカウントダウンを開始します。ウェイクアップ機能が有効な場合、低電力モードでもカウントダウンはアクティブのままとなります。さらに、カウンタがゼロに到達すると、RTC_ISR レジスタの WUTF フラグがセットされ、ウェイクアップカウンタが再ロード値（RTC_WUTR レジスタ値）で自動的に再ロードされます。

その後、WUTF フラグはソフトウェアでクリアする必要があります。

RTC_CR2 レジスタの WUTIE ビットをセットして周期的ウェイクアップ割り込みを有効にすると、デバイスは低電力モードを終了できます。

周期的なウェイクアップフラグは、RTC_CR レジスタの OSEL[0:1] ビットを通じて有効になっている場合に限り、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

低電力モード（SLEEP、STOP、STANDBY）と同様に、システムリセットもウェイクアップタイマには影響しません。

46.3.8 RTC の初期化と設定

RTC レジスタアクセス

RTC レジスタは、32 ビットのレジスタです。APB インターフェイスは、RTC レジスタアクセスに 2 ウェイトステートを挿入します。ただし、BYPSHAD = 0 のときのカレンダシャドウレジスタへの読み出しアクセスは除きます。

RTC レジスタ書き込み保護

システムリセット後、RTC レジスタは、PWR_CR1 レジスタの DBP ビットをクリアすることによって、不要な書き込みアクセスから保護されます（電源制御のセクションを参照）。RTC レジスタ書き込みアクセスを可能にするには、DBP ビットをセットする必要があります。

Backup ドメインリセット後、すべての RTC レジスタは書き込み保護されます。RTC レジスタへの書き込みは、書き込み保護レジスタ RTC_WPR にキーを書き込むことにより有効になります。

RTC_TAMPCR、RTC_BKPxR、RTC_OR 、および RTC_ISR[13:8] を除くすべての RTC レジスタの書き込み保護を解除するには、次のステップが必要です。

1. RTC_WPR レジスタに“0xCA”を書き込みます。
2. RTC_WPR レジスタに“0x53”を書き込みます。

誤ったキーを書き込むと、書き込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

カレンダの初期化と設定

時間形式やプリスケーラ設定を含むカレンダ時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ISR レジスタで INIT ビットを 1 にセットして、初期化モードに入ります。このモードでは、カレンダカウンタが停止し、その値を更新することができます。
2. RTC_ISR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. カレンダカウンタのための 1 Hz クロックを生成するには、RTC_PRER レジスタで両方のプリスケーラ分周比をプログラムします。
4. シャドウレジスタ（RTC_TR および RTC_DR）に時刻と日付の初期値をロードし、RTC_CR レジスタの FMT ビットを介して時間形式（12 時間または 24 時間）を設定します。
5. INIT ビットをクリアして初期化モードを終了します。その後、カレンダカウンタの実際の値が自動的にロードされ、4 RTCCLK クロックサイクル後にカウントが再開します。

初期化シーケンスが完了すると、カレンダがカウントを開始します。

注： ***システムリセット後、アプリケーションは RTC_ISR レジスタの INITS フラグを読み出し、カレンダが初期化されたか否かを確認できるようになります。このフラグが 0 であれば、カレンダの年の項目が Backup ドメインリセットデフォルト値（0x00）にセットされているため、初期化されていません。***

初期化後にカレンダを読み出すには、まずソフトウェアで RTC_ISR レジスタの RSF フラグがセットされていることを確認する必要があります。

サマータイム

サマータイム管理は、RTC_CR レジスタの SUB1H ビット、ADD1H ビット、BKP ビットを介して行われます。

SUB1H または ADD1H を使用すると、ソフトウェアは初期化手順を踏まずに 1 度の操作で、カレンダから 1 時間引いたり足したりすることができます。

さらに、ソフトウェアは BKP ビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下に示すのはアラーム A の手順ですが、アラーム B についても同様です。

1. RTC_CR の ALRAE をクリアしてアラーム A を無効にします。
2. アラーム A レジスタ（RTC_ALRMASSR/RTC_ALRMAR）をプログラムします。
3. RTC_CR レジスタで ALRAE をセットしてアラーム A を再び有効にします。

注： ***RTC_CR レジスタの各変更は、クロック同期のため RTCCLK クロック約 2 サイクル後に有効になります。***

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値（RTC_WUTR の WUT[15:0]）の設定または変更には、次の手順が必要です。

1. RTC_CR の WUTE をクリアしてウェイクアップタイマを無効にします。
2. RTC_ISR の WUTWF がセットされ、ウェイクアップ自動再ロードカウンタおよび WUCKSEL [2:0] ビットへのアクセスが許可されていることが確認されるまで WUTWF をポーリングします。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. ウェイクアップ自動再ロード値 WUT [15:0] およびウェイクアップクロック選択（RTC_CR の WUCKSEL [2:0] ビット）をプログラムします。RTC_CR で WUTE をセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウンを再開します。WUTWF ビットは、クロックの同期化により、WUTE クリア後、2 RTCCLK クロックサイクルまでクリアされます。

46.3.9 カレンダの読み出し

RTC_CR レジスタの BYPSHAD 制御ビットがクリアされている場合

RTC カレンダレジスタ（RTC_SSR、RTC_TR、および RTC_DR）を正しく読み出すには、APB クロック周波数（f_{PCLK}）が RTC クロック周波数（f_{RTCCLK}）の 7 倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APB クロック周波数が RTC クロック周波数の 7 倍未満である場合、ソフトウェアによってカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。どの場合も、APB クロック周波数は必ず RTC クロック周波数以上でなければなりません。

RTC_ISR レジスタの RSF ビットは、カレンダレジスタが RTC_SSR、RTC_TR、および RTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーは、2 RTCCLK サイクルごとに行われます。3 つの値における一貫性を保証するため、RTC_SSR または RTC_TR のどちらかを読み出すと、高次カレンダシャドウレジスタの値は RTC_DR が読み出されるまでロックされます。ソフトウェアが 2 RTCCLK サイクル未満の間隔でカレンダの読み出しアクセスを行う場合、最初のカレンダ読み出し後に RSF をソフトウェアでクリアする必要があり、その後ソフトウェアは、RSF ビットがセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

低電力モード（STOP または STANDBY）からのウェイクアップ後は、RSF をソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度 RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

RSF ビットは、ウェイクアップ後にクリアする必要がありますが、低電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化（*1839 ページのカレンダの初期化と設定*を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

同期（*セクション 46.3.11 : RTC の同期*を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

RTC_CR レジスタ（バイパスシャドウレジスタ）の BYPSHAD 制御ビットがセットされている場合

カレンダレジスタを読み出すと、カレンダカウンタの値が直接与えられるため、RSF ビットがセットされるのを待つ必要がありません。シャドウレジスタは低電力モード（STOP または STANDBY）では更新されないため、低電力モード終了後に特にこのような読み出しが有用です。

BYPSHAD ビットが 1 にセットされている場合、レジスタへの 2 回の読み出しアクセス間で RTCCLK エッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読み出し操作中に RTCCLK エッジが発生した場合、レジスタの 1 つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを 2 回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わり、ソフトウェアはカレンダレジスタの最下位の数値を 2 回比較するだけで構いません。

注： ***BYPSHAD = 1 の間、カレンダレジスタの読み出し命令が完了するには 1 APB サイクルだけ余計に必要となります。***

46.3.10 RTC のリセット

カレンダシャドウレジスタ（RTC_SSR、RTC_TR、RTC_DR）および RTC ステータスレジスタ（RTC_ISR）の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタは Backup ドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTC の現在のカレンダレジスタ、RTC 制御レジスタ（RTC_CR）、プリスケーラレジスタ（RTC_PRER）、RTC 較正レジスタ（RTC_CALR）、RTC シフトレジスタ（RTC_SHIFTR）、RTC タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR および RTC_TSDR）、RTC タンパ設定レジスタ（RTC_TAMPCR）、RTC バックアップレジスタ（RTC_BKPxR）、ウェイクアップタイマレジスタ（RTC_WUTR）、アラーム A とアラーム B レジスタ（RTC_ALRMASSR/RTC_ALRMAR と RTC_ALRMBSSR/RTC_ALRMBR）、およびオプションレジスタ（RTC_OR）。

さらに、LSE クロックによって駆動されている際にリセットソースが Backup ドメインリセットと異なる場合、システムリセットがかかっても RTC は動作を続けます（システムリセットの影響を受けない RTC クロックソースのリストの詳細については、リセットおよびクロックコントローラの RTC クロック セクションを参照してください）。Backup ドメインリセットが発生すると、RTC は停止し、すべての RTC レジスタがリセット値にセットされます。

46.3.11 RTC の同期

RTC は、高精度でリモートクロックと同期できます。サブセカンド項目（RTC_SSR または RTC_TSSSR）を読み出すと、リモートクロックによって維持されている時刻と RTC 間の正確なオフセットが計算できます。その後、RTC_SHIFTR を使用してほんの一瞬クロックを「シフト」することによって RTC を調整し、このオフセットを取り除くことができます。

RTC_SSR には、同期プリスケーラのカウンタの値が入っています。これにより、RTC によって維持されている正確な時刻を 計算でき、その分解能は 1/(PREDIV_S + 1) 秒です。その結果、同期プリスケーラ値（PREDIV_S[14:0]）を増加させることにより分解能を改善できます。許可されている最大分解能（32768 Hz クロックで 30.52 µs）は、PREDIV_S を 0x7FFF にセットすることにより得られます。

ただし、PREDIV_S を増加させるということは、同期プリスケーラの出力を 1 Hz に維持するため PREDIV_A を減らす必要があることを意味します。このように、非同期プリスケーラの出力周波数が増加すると、RTC の動的消費電力が増加する場合があります。

RTC は、RTC シフト制御レジスタ（RTC_SHIFTR）を使って微調整できます。RTC_SHIFTR に書き込むことにより、1/（PREDIV_S + 1）秒の分解能で、クロックを最大 1 秒だけシフト（遅れ／進み）させることができます。このシフト操作の本質は、同期プリスケーラのカウンタ SS[15:0] に SUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。同時に ADD1S ビットがセットされた場合、1 秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。

注意： ***シフト操作を始める前に、ユーザは SS[15] = 0 であることを確認し、オーバーフローが発生しないようにする必要があります。***

RTC_SHIFTR レジスタへの書き込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示す SHPF フラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： ***この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1 のときにファームウェアから RTC_SHIFTR への書き込みはできません。***

46.3.12 RTC リファレンスクロック検出

RTC カレンダの更新は、リファレンスクロックである RTC_REFIN に同期させることができます。通常は商用電源（50 または 60 Hz）です。RTC_REFIN リファレンスクロックには、32.768 kHz LSE クロックより高い精度が必要です。RTC_REFIN 検出を有効にした際（RTC_CR の REFCKON ビットが 1 にセット）、カレンダは引き続き LSE クロックによって駆動され、RTC_REFIN はカレンダ更新周波数（1 Hz）の誤差の補正に使用されます。

各 1 Hz クロックエッジは、一番近い RTC_REFIN クロックエッジ（所与の時間枠内に見つかった場合）と比較されます。ほとんどの場合、2 つのクロックエッジは正しく整列しています。LSE クロックが不正確なために 1 Hz のクロックがずれた場合、RTC は 1 Hz のクロックを少しシフトさせ、その後の 1 Hz のクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTC は、32.768 kHz クォーツから生成される 256 Hz クロック（ck_apre）を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新（1 秒ごと）程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠は ck_apre 7 周期に等しくなります。その後のカレンダ更新では、ck_apre 3 周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_apre クロックを出力する非同期プリスケーラは強制的に再ロードされます。プリスケーラは同時に再ロードされるので、リファレンスクロックおよび 1 Hz のクロックが整列するタイミングには影響しません。クロックが整列していない場合、後の 1 Hz クロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した（ck_apre 3 周期の枠内でリファレンスクロックエッジが発生しない）場合、カレンダは LSE クロックのみを基準にして更新が継続されます。その後 RTC は ck_spre エッジを中心として ck_apre 7 周期という広い検出時間枠でリファレンスクロックを待ちます。

基準クロック検出を有効にした場合、PREDIV_A および PREDIV_S を以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREVID_S = 0x00FF

注： ***RTC_REFIN クロック検出は、STANDBY モードでは利用できません。***

46.3.13 RTC の高精度デジタル較正

RTC 周波数の精度は、-487.1～+488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル的に較正できます。周波数の修正は、一連の微調整（個々の RTCCLK パルスの追加や削除）によって行われます。このような調整は、短い期間で観測された場合でも RTC が十分に較正されるように、かなり広範に分散して行われます。

この高精度デジタル較正は、入力周波数が 32768 Hz の場合、RTCCLK 約 2^{20} パルスのサイクルまたは 32 秒の間で行われます。このサイクルは、RTCCLK によって駆動される 20 ビットカウンタ、cal_cnt[19:0] によって維持されます。

高精度較正レジスタ（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK クロックサイクル数を指定します。

- CALM[0] ビットを 1 にセットすると、32 秒サイクルの中でちょうど 1 パルス

がマスクされます。

- CALM[1] ビットを 1 にセットすると、さらに 2 サイクルがマスクされます。
- CALM[2] ビットを 1 にセットすると、さらに 4 サイクルがマスクされます。
- CALM[8] ビットを 1 にセットするまで続けると、256 クロックがマスクされます。

注： ***CALM[8:0]（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK パルス数を指定します。CALM[0] ビットを 1 にセットすると、cal_cnt[19:0] = 0x80000 になった時点で、32 秒サイクル中でちょうど 1 パルスがマスクされます。CALM[1] = 1 では、さらに 2 サイクルがマスクされ（cal_cnt = 0x40000 および 0xC0000）、CALM[2] = 1 では、さらに 4 サイクルがマスクされ（cal_cnt = 0x20000/0x60000/0xA0000/ 0xE0000）、CALM[8] = 1 まで続けると、256 クロックがマスクされます（cal_cnt = 0xXX800）。***

CALM では、細かい分解能で RTC 周波数を最大 487.1 ppm 負の方向に調整することができ、CALP ビットでは周波数を 488.5 ppm 正の方向に調整することができます。CALP を 1 にセットすることにより、実質上は、RTCCLK 2^{11} サイクルごとに、RTCCLK パルスが 1 パルス追加で挿入されます。すなわち、32 秒サイクルごとに 512 クロックが追加されることとなります。

CALM を CALP と合わせて使用すると、32 秒サイクルの間に RTCCLK -511 から +512 サイクルまでのオフセットが追加でき、これは約 0.954 ppm の分解能で較正範囲 -487.1～+488.5 ppm に換算されます。

有効較正周波数（FCAL）を入力周波数（FRTCCLK）に対して求める計算式は次のとおりです。

$$F_{CAL} = F_{RTCCLK} \times [1 + (CALP \times 512 - CALM) / (2^{20} + CALM - CALP \times 512)]$$

PREDIV_A < 3 の場合の較正

非同期プリスケーラ値（RTC_PRER レジスタの PREDIV_A ビット）が 3 未満の場合、CALP ビットを 1 にセットすることはできません。CALP がすでに 1 にセットされていて、PREDIV_A ビットが 3 未満の値にセットされた場合、CALP の設定値は無視され、CALP が 0 に設定された場合と同じように較正されます。

PREDIV_A が 3 未満の状態で較正を実施するには、各秒のカウントが 8 RTCCLK クロックサイクル分早められるように同期プリスケーラ値（PREDIV_S）を小さくする必要があります。これは 32 秒毎に 256 クロックサイクル追加することに相当します。結果として、CALM ビットのみを使用して 32 秒周期の間に 255 クロックパルスから 256 クロックパルス（243.3 から 244.1 ppm の較正範囲に相当）を追加することができます。

公称 RTCCLK 周波数が 32768 Hz で、PREDIV_A が 1（分周比 2）の場合、PREDIV_S を 16383 ではなく 16379（4 少ない）にセットする必要があります。また、PREDIV_A が 0 の場合、PREDIV_S を 32767 ではなく 32759 （8 少ない）にセットする必要がありますので注意して下さい。

PREDIV_S をこのように減少させた場合、較正された入力クロックの有効周波数の式は次のようになります。

$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100（CALM 設定範囲の中間値）と等しくなるのが正しい設定です。

RTC 較正値の確認

RTC の精度は、RTCCLK の正確な周波数を測定し、正しい CALM 値および CALP 値を計算することにより保証されます。オプションの 1 Hz 出力が搭載されており、アプリケーションによって RTC 精度の測定と確認を行うことができます。

ある時間間隔で RTC の周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大 2 RTCCLK クロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は 32 秒です。

このモードを使用して正確に 32 秒で 1 Hz 出力の精度を測定すると、その精度は 0.477 ppm（較正分解能の制限により 32 秒で 0.5 RTCCLK サイクル）以内となることが保証されます。

- RTC_CALR レジスタの CALW16 ビットを 1 にセットして、較正サイクル周期を強制的に 16 秒にすることができます。

この場合、RTC 精度は最大誤差 0.954 ppm（16 秒で 0.5 RTCCLK サイクル）で 16 秒間で測定できます。ただし、較正分解能が下がるため、長期的な RTC 精度もまた 0.954 ppm に下がります。CALW16 が 1 にセットされると、CALM[0] ビットは 0 のままとなります。

- RTC_CALR レジスタの CALW8 ビットを 1 にセットして、較正サイクル周期を強制的に 8 秒にすることができます。

この場合、RTC 精度は最大誤差 1.907 ppm（8 秒で 0.5 RTCCLK サイクル）で 8 秒で測定できます。長期的な RTC 精度もまた 1.907 ppm に下がります。CALW8 が 1 にセットされると、CALM[1:0] ビットは 00 のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ISR/INITF = 0 の間でも、較正レジスタ（RTC_CALR）を動作中に更新することができます。

1. RTC_ISR/RECALPF（再較正保留フラグ）をポーリングします。
2. このフラグが 0 にセットされている場合は、必要に応じて新しい値を RTC_CALR に書き込みます。すると、RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書き込み動作後 ck_apre 3 サイクル以内に、新しい較正設定が有効になります。

46.3.14 タイムスタンプ機能

タイムスタンプは、RTC_CR レジスタの TSE または ITSE ビットを 1 にセットすることにより有効になります。

TSE がセットされている場合：

RTC_TS ピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

ITSE がセットされている場合：

内部タイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。内部タイムスタンプイベントは、VBAT 電源への切り替えによって生成されます。

また、タイムスタンプイベントが発生すると、内部または外部イベントによって、RTC_ISR レジスタのタイムスタンプフラグビット（TSF）がセットされます。イベントが内部である場合は、ITSF フラグが RTC_ISR レジスタにもセットされます。

RTC_CR レジスタの TSIE ビットをセットすることにより、タイムスタンプイベントが発生したときに割り込みが生成されます。

タイムスタンプフラグ（TSF）がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ（TSOVF）がセットされ、タイムスタンプレジスタ（RTC_TSTR および RTC_TSDR）は、その前のイベントの結果を維持します。

注： ***同期処理のため、TSF はタイムスタンプイベント発生から ck_apre 2 サイクル後にセットされます。一方、TSOVF のセットに遅延はありません。これは、2 つのタイムスタンプイベントの発生したタイミングが近い場合、TSF がまだ“0”であっても TSOVF が“1”と検出される可能性があることを意味します。よって、TSOVF のポーリングは TSF がセットされた後に実施することをお勧めします。***

注意： ***TSF ビットがクリアされたと思われた直後にタイムスタンプイベントが発生した場合、TSF および TSOVF ビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、TSF がすでに“1”と読み出されていない限りは、アプリケーションによる“0”クリアの処理を行ってはなりません。***

オプション機能として、タンパイベントによってタイムスタンプイベントを記録することもできます。TAMPTS 制御ビットの詳細については、*セクション 46.6.16：RTC タンパ設定レジスタ（RTC_TAMPCR）*を参照してください。

46.3.15 タンパ検出

RTC_TAMPx 入力イベントは、フィルタを使用してエッジ検出またはレベル検出のどちらにも設定できます。

タンパ検出は次の目的で設定することができます。

- RTC バックアップレジスタおよびバックアップ SRAM（デフォルト設定）を消去する。
- STOP モードおよび STANDBY モードからのウェイクアップが可能な割り込みを生成する。
- 低電力タイマのためのハードウェアトリガを生成する。

RTC バックアップレジスタ

バックアップレジスタ（RTC_BKPxR）は、システムリセットや STANDBY モードからのウェイクアップではリセットされません。

バックアップレジスタは、タンパ検出イベント発生時にリセットされます（*セクション 46.6.20 : RTC バックアップレジスタ（RTC_BKPxR）* および *1846 ページのタンパ検出の初期化*を参照。）が、TAMPxNOERASE ビットがセットされている場合や RTC_TAMPCR レジスタの TAMPxMF がセットされている場合は例外です。

タンパ検出の初期化

各入力は、RTC_TAMPCR レジスタにおいて該当する TAMPxE ビットを 1 にセットすることにより有効にできます。

各 RTC_TAMPx タンパ検出入力は、RTC_ISR レジスタの TAMPxF フラグに関連付けられています。

TAMPxMF がクリアされている場合

TAMPxF フラグは、ピン上でタンパイベントが発生した後にアサートされます。その際の遅延時間を以下に示します。

- TAMPFLT = 0x0 以外の場合、ck_apre 3 サイクル（フィルタを使ったレベル検出）
- TAMPTS = 1 の場合、ck_apre 3 サイクル（タンパイベント時のタイムスタンプ）
- TAMPFLT = 0x0（エッジ検出）および TAMPTS = 0 の場合、遅延なし

TAMPxF がセットされている場合に、この周期中に同一ピンで発生した新たなタンパイベントを検出することはできません。

TAMPxMF がセットされている場合

同一ピンで発生した新たなタンパイベントは、上述した遅延の間とさらに ck_rtc 2.5 サイクルの間は検出できません。

RTC_TAMPCR レジスタの TAMPIE ビットをセットすることにより、タンパ検出イベント発生時に割り込みが生成されます （TAMPxF がセットされている場合）。1 つ以上の S TAMPxMF がセットされている場合、TAMPIE をセットすることはできません。

TAMPIE がクリアされている場合は、各タンパピンイベント／割り込みは、RTC_TAMPCR レジスタの対応する TAMPxIE ビットをセットすることによって個別に有効にすることができます。対応する TAMPxMF がセットされている場合、TAMPxIE をセットすることはできません。

タンパイベント時のトリガ出力の生成

タンパイベント検出は、低電力タイマによるトリガ入力として使用できます。

RTC_TAMPCR レジスタ の TAMPxMF ビットがクリアされている場合、TAMPxF フラグをソフトウェアでクリアして、同一ピンで新たなタンパイベントを検出できるようにする必要があります。

TAMPxMF ビットがセットされている場合、TAMPxF フラグはマスクされ、RTC_ISR レジスタでクリアされたままとなります。この設定により、TAMPxF をクリアするためにシステムをウェイクアップする必要もなく、STOP モードのまま低電力タイマを自動的にトリガすることができます。この場合、バックアップレジスタはクリアされません。

タンパイベント時のタイムスタンプ

TAMPTS を“1”にセットすると、すべてのタンパイベントがタイムスタンプイベントを発生させるようになります。この場合、通常のタイムスタンプイベント発生時と同様に TSF ビットまたは TSOVF ビットが RTC_ISR でセットされます。TSF または TSOVF がセットされるのと同時に、影響を受けるタンパフラグレジスタ、TAMPxF がセットされます。

タンパ入力でのエッジ検出

TAMPFLT ビットが“00”の場合、該当する TAMPxTRG ビットに応じて立ち上がりエッジまたは立ち下がりエッジが観測されると、RTC_TAMPx ピンがタンパ検出イベントを生成します。エッジ検出を選択すると、RTC_TAMPx 入力の内部プルアップ抵抗が無効になります。

注意： ***エッジ検出を使用する場合は、タンパ検出の有効化直後のタンパピンのレベルをソフトウェアでチェックすることをお勧めします（GPIO レジスタを読み出す）。また、タンパイベントの検出を有効化する前にアクティブエッジが発生していないかどうかを確認するため、影響を受ける値をバックアップレジスタに書き込む前にソフトウェアでチェックすることをお勧めします。***
TAMPFLT="00" で TAMPxTRG = 0（立ち上がりエッジ検出）のとき、タンパ検出を有効にする前にタンパ入力がすでにハイレベルにある場合は、タンパイベントをハードウェアで検出できます。

タンパイベントが検出されクリアされた後に、バックアップレジスタ（RTC_BKPxR）を再プログラムする場合には、事前に、RTC_TAMPx を無効にしてから再度有効にする（TAMPxE を 1 にセット）必要があります。これによって、RTC_TAMPx 入力の値がタンパ検出を示している間に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、RTC_TAMPx の入力でのレベル検出に相当します。

注： ***タンパ検出は、V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不必要なリセットを避けるには、RTC_TAMPx が設定されているピンを外部で適切な信号レベルに接続しておく必要があります。***

RTC_TAMPx 入力でのフィルタを使ったレベル検出

フィルタを使ったレベル検出は、TAMPFLT を 0 以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLT に応じて）2、4 または 8 回のいずれかの連続したサンプルが TAMPxTRG ビットで指定するレベルで観測されたときに生成されます。

RTC_TAMPx 入力は、TAMPPUDIS が 1 にセットされて無効な状態になっていない限り、その状態がサンプリングされる前に I/O の内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間は TAMPPRCH ビットによって決定され、RTC_TAMPx 入力ピンにおけるより大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQ を使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： ***プルアップ抵抗の電気的特性については、データシートを参照してください。***

46.3.16 較正クロック出力

RTC_CR レジスタで COE ビットが 1 にセットされると、RTC_CALIB デバイス出力にリファレンスクロックが供給されます。

RTC_CR レジスタの COSEL ビットがリセットされ、かつ PREDIV_A = 0x7F である場合、RTC_CALIB 周波数は $f_{RTCCLK/64}$です。これは 32.768 kHz の RTCCLK 周波数に対する 512 Hz の較正出力に相当します。立ち下がりエッジには軽いジッタがあるため、RTC_CALIB のデューティサイクルは不規則になります。したがって、立ち上がりエッジの使用が推奨されます。

COSEL がセットされ、かつ “PREDIV_S+1”がゼロ以外の 256 の倍数である場合（すなわち、PREDIV_S[7:0] = 0xFF）、RTC_CALIB 周波数は fRTCCLK/(256 * (PREDIV_A+1))となります。これは、RTCCLK 周波数が 32.768 kHz で、プリスケーラデフォルト値（PREDIV_A = 0x7F、PREDIV_S = 0xFF）に対する 1 Hz の較正出力に相当します。シフト操作が進行中の場合、1 Hz 出力が影響を受け、シフト操作中にトグルされる可能性があります（SHPF=1）。

注： ***RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力に設定されます。***

COSEL ビットがクリアされると、RTC_CALIB 出力は非同期プリスケーラの 6 番目のステージの出力になります。

COSEL ビットがセットされると、RTC_CALIB 出力は同期プリスケーラの 8 番目のステージの出力になります。

46.3.17 アラーム出力

RTC_CR レジスタの OSEL[1:0] 制御ビットを使用してアラーム出力、RTC_ALARM を有効にし、出力となる機能を選択します。これらの機能は、RTC_ISR レジスタの該当するフラグの内容を反映します。

RTC_CR の POL 制御ビットは、POL が 1 にセットされているときに選択されたフラグビットの逆が出力されるよう、出力の極性を決定します。

アラーム出力

RTC_ALARM ピンは、制御ビット RTC_ALARM_TYPE（RTC_OR レジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。

注： ***RTC_ALARM 出力が有効になると、この設定は、RTC_CALIB（COE ビットは無視され、クリアされたまま）の設定よりも優先されます。***

RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力に設定されます。

46.4 RTC 低電力モード

表 356. 低消費電力モードが RTC に与える影響

モード	説明
STOP	ペリフェラルレジスタの内容は保たれます。
STANDBY	RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC アラーム、RTC タンパイベント、RTC タイムスタンプイベント、RTC ウェイクアップにより、デバイスは STANDBY モードから復帰します。

46.5 RTC 割り込み

すべての RTC 割り込みは、EXTI コントローラに接続されています。*セクション 20：拡張割り込み／イベントコントローラ（EXTI）を参照してください。*

RTC アラーム割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC アラームイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_ALARM IRQ チャネルを設定し、有効にします。
3. RTC が RTC アラームを生成するように設定します。

RTC タンパ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC タンパイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_TAMP_STAMP IRQ チャネルを設定し、有効にします。
3. RTC が RTC タンパイベントを検出するように設定します。

RTC タイムスタンプ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC タイムスタンプイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_TAMP_STAMP IRQ チャネルを設定し、有効にします。
3. RTC タイムスタンプイベントを検出するよう RTC を設定します。

ウェイクアップタイマ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードでウェイクアップタイマ割り込みイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_WKUP IRQ チャネルを設定し、有効にします。
3. RTC が RTC ウェイクアップタイマイベントを検出するように設定します。

表 357. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
アラーム A	ALRAF	ALRAIE	あり	あり(1)	あり (1)
アラーム B	ALRBF	ALRBIE	あり	あり (1)	あり (1)
RTC_TS 入力（タイムスタンプ）	TSF	TSIE	あり	あり (1)	あり (1)
RTC_TAMP1 入力検出	TAMP1F	TAMPIE	あり	あり (1)	あり (1)
RTC_TAMP2 入力検出	TAMP2F	TAMPIE	あり	あり (1)	あり (1)
RTC_TAMP3 入力検出	TAMP3F	TAMPIE	あり	あり (1)	あり (1)
ウェイクアップタイマ割り込み	WUTF	WUTIE	あり	あり (1)	あり (1)

1. STOP モードおよび STANDBY モードからのウェイクアップは、RTC クロックソースが LSE または LSI のときのみ可能です。

46.6 RTC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*96 ページのセクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

46.6.1 RTC 時刻レジスタ（RTC_TR）

RTC_TR は、カレンダ時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。*1839 ページのカレンダの初期化と設定* および *1840 ページのカレンダの読み出し* を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x00

Backup ドメインリセット値：0x0000 0000

システムリセット：BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31-23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

46.6.2 RTC 日付レジスタ（RTC_DR）

RTC_DR は、カレンダ日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。*1839 ページのカレンダの初期化と設定* および *1840 ページのカレンダの読み出し* を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x04

Backup ドメインリセット値：0x0000 2101

システムリセット：BYPSHAD = 0 の場合、0x0000 2101 です。BYPSHAD = 1 の場合、影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **YT[3:0]**：BCD 形式での年の十の位

ビット 19:16 **YU[3:0]**：BCD 形式での年の一の位

ビット 15:13 **WDU[2:0]**：曜日

000：禁止
001：月曜日
...
111：日曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

46.6.3 RTC 制御レジスタ （RTC_CR）

アドレスオフセット：0x08

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
							rw	rw	rw	rw	rw	rw	rw	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPS HAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ITSE**：内部イベントでのタイムスタンプイネーブル

0：内部イベントでのタイムスタンプは無効です。
1：内部イベントでのタイムスタンプは有効です。

ビット 23 **COE**：較正出力イネーブル

このビットは、RTC_CALIB 出力を有効にします。

0：較正出力は無効です。
1：較正出力は有効です。

ビット 22:21 **OSEL[1:0]**：出力選択

これらのビットは、RTC_ALARM 出力に送られるフラグの選択に使用します。

00：出力は無効です。
01：アラーム A 出力は有効です。
10：アラーム B 出力は有効です。
11：ウェイクアップ出力は有効です。

ビット 20 **POL**：出力極性

このビットは、RTC_ALARM 出力の極性の設定に使用します。

0：ALRAF/ALRBF/WUTF がアサートされると、このピンがハイになります（OSEL[1:0] に応じて）。
1：ALRAF/ALRBF/WUTF がアサートされると、このピンがローになります（OSEL[1:0] に応じて）。

ビット 19 **COSEL**：較正出力選択

COE = 1 のとき、このビットによって RTC_CALIB に出力される信号を選択します。

0：較正出力は 512 Hz です（デフォルトのプリスケーラ設定）
1：較正出力は 1 Hz です（デフォルトのプリスケーラ設定）

これらの周波数は、RTCCLK が 32.768 kHz で、プリスケーラがデフォルト値（PREDIV_A = 127 および PREDIV_S = 255）の場合に有効です。*セクション 46.3.16：較正クロック出力* を参照してください。

ビット 18 **BKP**：バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザが書き込むことができます。

ビット 17 **SUB1H**：*1 時間*差し引き（冬時間変更）

このビットをセットすると、現在時刻が 0 でない場合にカレンダ時刻から 1 時間を差し引きます。このビットは常に 0 として読み出されます。

現在時間が 0 のときにこのビットをセットしても、影響はありません。

0：影響なし。
1：現在時刻から 1 時間差し引きます。これは、初期化モード以外の冬時間変更に使用できます。

ビット 16 **ADD1H**：1 時間加算（サマータイム変更）

このビットをセットすると、カレンダ時刻に 1 時間加算します。このビットは常に 0 として読み出されます。

0：影響なし。

1：現在時刻に 1 時間加算します。これは、初期化モード以外のサマータイム変更に使用できます。

ビット 15 **TSIE**：タイムスタンプ割り込みイネーブル

0：タイムスタンプ割り込みは無効です。

1：タイムスタンプ割り込みは有効です。

ビット 14 **WUTIE**：ウェイクアップタイマ割り込みイネーブル

0：ウェイクアップタイマ割り込みは無効です。

1：ウェイクアップタイマ割り込みは有効です。

ビット 13 **ALRBIE**：*アラーム B 割り込みイネーブル*

0：アラーム B 割り込みは無効です。

1：アラーム B 割り込みは有効です。

ビット 12 **ALRAIE**：アラーム A 割り込みイネーブル

0：アラーム A 割り込みは無効です。

1：アラーム A 割り込みは有効です。

ビット 11 **TSE**：タイムスタンプイネーブル

0：タイムスタンプは無効です。

1：タイムスタンプは有効です。

ビット 10 **WUTE**：ウェイクアップタイマイネーブル

0：ウェイクアップタイマは無効です。

1：ウェイクアップタイマは有効です。

ビット 9 **ALRBE**：*アラーム B イネーブル*

0：アラーム B は無効です。

1：アラーム B は有効です。

ビット 8 **ALRAE**：アラーム A イネーブル

0：アラーム A は無効です。

1：アラーム A は有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FMT**：時間形式

0：24 時間／日形式

1：AM／PM 時間形式

ビット 5 **BYPSHAD**：シャドウレジスタをバイパスします。

0：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、シャドウレジスタから取得され、これらは 2 RTCCLK サイクルごとに 1 回更新されます。

1：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、カレンダカウンタから直接取得されます。

注： ***APB クロックの周波数が RTCCLK の 7 倍未満である場合、BYPSHAD は“1”にセットする必要があります。***

ビット 4 **REFCKON**：RTC_REFIN リファレンスクロック検出イネーブル（50 または 60 Hz）

0：RTC_REFIN 検出は無効です。

1：RTC_REFIN 検出は有効です。

注： PREDIV_S は 0x00FF である必要があります。

ビット 3 **TSEDGE**：タイムスタンプイベントアクティブエッジ

0：RTC_TS 入力の立ち上がりエッジによってタイムスタンプイベントを生成します。

1：RTC_TS 入力の立ち下がりエッジによってタイムスタンプイベントを生成します。

不要な TSF 設定を回避するため、TSEDGE が変化した場合には TSE をリセットする必要があります。

ビット 2:0 **WUCKSEL[2:0]**：ウェイクアップクロック選択

000：RTC/16 クロックが選択されます。

001：RTC/8 クロックが選択されます。

010：RTC/4 クロックが選択されます。

011：RTC/2 クロックが選択されます。

10x：ck_spre（通常は 1 Hz）クロックが選択されます。

11x：ck_spre（通常は 1 Hz）クロックが選択され、2^{16} が WUT カウンタ値に加算されます（下記注を参照）。

注： 初期化モード（RTC_ISR/INITF = 1）の場合のみ、このレジスタのビット 7、6、4 が書き込めます。

WUT = ウェイクアップユニットカウンタ値WUT = (0x0000 〜0xFFFF) + 0x10000（WUCKSEL[2:1] = 11 の場合追加されます。）

このレジスタのビット 2〜0 は、RTC_CR WUTE ビット = 0 かつ RTC_ISR WUTWF ビット = 1 の場合にのみ書き込めます。

カレンダの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1H および SUB1H の変更は、次の秒から有効になります。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、1838 ページのRTC レジスタ書き込み保護を参照してください。

注意： TSE は、TSEDGE が変更されたらリセットし、TSF の不要なセットを回避する必要があります。

46.6.4 RTC 初期化とステータスレジスタ（RTC_ISR）

このレジスタは、書き込み保護されています（RTC_ISR[13:8] ビットを除く）。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x0C

Backup ドメインリセット値：0x0000 0007

システムリセット：0 にクリアされる INIT、INITF、RSF ビット以外は影響されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	RECALPF
														rc_w0	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAMP3F	TAMP2F	TAMP1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rw	r	rc_w0	r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **ITSF**：内部タイムスタンプフラグ

このフラグは、内部イベントでタイムスタンプが発生したときに、ハードウェアによってセットされます。

このフラグは、0 を書き込むことでソフトウェアによってクリアされ、両方のビットに 0 を書き込むことで TSF ビットとともにクリアする必要があります。

ビット 16 **RECALPF**：再較正保留フラグ

ソフトウェアによって RTC_CALR レジスタに書き込みが行われると、RECALPF ステータスフラグが自動的に“1”にセットされ、RTC_CALR レジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは“0”に戻ります。*動作中の再較正*を参照してください。

ビット 15 **TAMP3F**：RTC_TAMP3 検出フラグ

このフラグは、RTC_TAMP3 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 14 **TAMP2F**：RTC_TAMP2 検出フラグ

このフラグは、RTC_TAMP2 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 13 **TAMP1F**：RTC_TAMP1 検出フラグ

このフラグは、RTC_TAMP1 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 12 **TSOVF**：タイムスタンプオーバーフローフラグ

このフラグは、TSF が既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。そうしないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 11 **TSF**：タイムスタンプフラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。ITSF フラグがセットされている場合は、TSF は両方のビットに 0 を書き込むことで、ITSF とともにクリアする必要があります。

ビット 10 **WUTF**：ウェイクアップタイマフラグ

このフラグは、ウェイクアップ自動再ロードカウンタが 0 に到達したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 9 **ALRBF**：アラーム B フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム B レジスタ（RTC_ALRMBR）と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 8 **ALRAF**：アラーム A フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム A レジスタ（RTC_ALRMAR）と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 7 **INIT**：初期化モード

0：フリーランニングモード

1：時刻と日付レジスタ（RTC_TR と RTC_DR）、およびプリスケーラレジスタ（RTC_PRER）のプログラムに使用する初期化モードです。INIT がリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット 6 **INITF**：初期化フラグ

このビットが 1 にセットされると、RTC は初期化状態となり、時刻、日付およびプリスケーラレジスタが更新できます。

0：カレンダレジスタを更新できません。

1：カレンダレジスタを更新できます。

ビット 5 **RSF**：レジスタ同期フラグ

このビットは、カレンダレジスタがシャドウレジスタ（RTC_SSRx、RTC_TRx および RTC_DRx）にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中（SHPF = 1）に初期化モードで、またはバイパスシャドウレジスタモード（BYPSHAD = 1）で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0：カレンダシャドウレジスタはまだ同期していません。

1：カレンダシャドウレジスタは同期しています。

ビット 4 **INITS**：初期化ステータスフラグ

このビットは、カレンダの年の項目が 0 ではないとき（Backup ドメインリセット状態）にハードウェアによってセットされます。

0：カレンダは初期化されていません。

1：カレンダは初期化されています。

ビット 3 **SHPF**：シフト操作保留

0：保留中のシフト操作はありません。

1：保留中のシフト操作があります。

このフラグは、RTC_SHIFTR への書き込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPF ビットに書き込んでも影響はありません。

ビット 2 **WUTWF**：ウェイクアップタイマ書き込みフラグ

このビットは、ハードウェアによって WUTE ビットが RTC_CR で 0 にセットされた後で 2 RTCCLK サイクルまでセットされ、WUTE ビットが 1 にセットされた後で 2 RTCCLK サイクルまでクリアされます。ウェイクアップタイマ値は、WUTE ビットがクリアされ、WUTWF がセットされたときに変更されます。

0：ウェイクアップタイマ設定は更新できません。

1：ウェイクアップタイマ設定は更新できます。

ビット 1 **ALRBWF**：アラーム B 書き込みフラグ

このビットは、RTC_CR で ALRBIE ビットが 0 にセットされた後、アラーム B 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム B は更新できません。

1：アラーム B は更新できます。

ビット 0 **ALRAWF**：アラーム A 書き込みフラグ

このビットは、RTC_CR で ALRAE ビットが 0 にセットされた後、アラーム A 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム A は更新できません。

1：アラーム A は更新できます。

注： ***ALRAF、ALRBF、WUTF 、および TSF ビットは、0 にプログラムされてから 2 APB クロック サイクル後にクリアされます。***

46.6.5 RTC プリスケーラレジスタ（RTC_PRER）

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2 回の書き込みアクセスに分けて行う必要があります。*1839 ページのカレンダの初期化と設定*を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x10

Backup ドメインリセット値：0x007F 00FF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PREDIV_S[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **PREDIV_A[6:0]**：非同期プリスケーラ分周比
非同期分周比です。
ck_apre 周波数 = RTCCLK 周波数／(PREDIV_A+1)

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PREDIV_S[14:0]**：同期プリスケーラ分周比
同期分周比です。
ck_spre 周波数 = ck_apre 周波数／(PREDIV_S+1)

46.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ISR の WUTWF が 1 にセットされているときにのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x14

Backup ドメインリセット値：0x0000 FFFF

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTE が 1 にセット）なとき、ck_wut の（WUT[15:0] + 1）サイクル毎に WUTF フラグがセットされます。ck_wut の周期は、RTC_CR レジスタの WUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1 のとき、ウェイクアップタイマは 17 ビットとなり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットである WUT[16] となります。

WUTF の最初のアサートは、WUTE がセットされてから ck_wut の （WUT+1）サイクル後に発生します。

WUCKSEL[2:0] = 011 (RTCCLK/2) のときに WUT[15:0] を 0x0000 にセットすることはできません。

46.6.7 RTC アラーム A レジスタ（RTC_ALRMAR）

このレジスタは、RTC_ISR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x1C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム A 日付マスク
0：日付／曜日が一致すると、アラーム A がセットされます。
1：アラーム A の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム A 時マスク
0：時が一致すると、アラーム A がセットされます。
1：アラーム A の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム A 分マスク
0：分が一致すると、アラーム A がセットされます。
1：アラーム A の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム A 秒マスク
0：秒が一致すると、アラーム A がセットされます。
1：アラーム A の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

46.6.8 RTC アラーム B レジスタ（RTC_ALRMBR）

このレジスタは、RTC_ISR の ALRBWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x20

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム B 日付マスク
0：日付／曜日が一致すると、アラーム B がセットされます。
1：アラーム B の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム B 時マスク
0：時が一致すると、アラーム B がセットされます。
1：アラーム B の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム B 分マスク
0：分が一致すると、アラーム B がセットされます。
1：アラーム B の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム 秒マスク
0：秒が一致すると、アラーム B がセットされます。
1：アラーム B の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

46.6.9 RTC 書き込み保護レジスタ（RTC_WPR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY							
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY**：書き込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に 0x00 が返されます。

RTC レジスタの書き込み保護解除方法については、*RTC レジスタ書き込み保護* を参照してください。

46.6.10 RTC サブセカンドレジスタ（RTC_SSR）

アドレスオフセット：0x28

Backup ドメインリセット値：0x0000 0000

システムリセット：BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **SS**：サブセカンド値

SS[15:0] は、同期プリスケーラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。

秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)

注： ***SS は、シフト操作後に限り、PREDIV_S より大きな値となる場合があります。この場合、正確な時刻／日付は、RTC_TR/RTC_DR で示される値よりも 1 秒少ない値となります。***

46.6.11 RTC シフト制御レジスタ（RTC_SHIFTR）

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護を参照してください。*

アドレスオフセット：0x2C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS[14:0]														
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **ADD1S**：1 秒加算

0：影響なし。

1：時計／カレンダに 1 秒加算します。

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ISR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS（下記説明を参照）と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SUBFS**：秒の小数部差し引き

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ISR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

SUBFS に書き込まれた値は、同期プリスケーラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き（遅延）されます。

遅れ（秒）= SUBFS / (PREDIV_S + 1)

ADD1S 機能が SUBFS と共に用いられた場合、秒の小数部を効果的にクロックに加算する（クロックを進める）ことができ、実際のクロックの進みは次の式のとおりとなります。

進み（秒）= (1 - (SUBFS / (PREDIV_S + 1)))

注： ***SUBFS に書き込むことにより RSF はクリアされます。その後、ソフトウェアが RSF = 1 まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実になります。***

46.6.12 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x30

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM :** AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

46.6.13 RTC タイムスタンプ日付レジスタ（RTC_TSDR）

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x34

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **WDU[1:0]**：曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

46.6.14 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_ISR/TSF がセットされている場合にのみ有効です。また、RTC_ISR/TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x38

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS**：サブセカンド値

SS[15:0] は、タイムスタンプイベントが発生したときの同期プリスケーラのカウンタの値です。

46.6.15 RTC 較正レジスタ（RTC_CALR）

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1838 ページのRTC レジスタ書き込み保護を参照してください。*

アドレスオフセット：0x3C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALP**：RTC 周波数を 488.5 ppm 増加

0：RTCCLK パルスは加えられません。

1：RTCCLK の 2^{11} パルスごとに 1 パルス効果的に挿入されます（周波数が 488.5 ppm 増加）。

この機能は、CALM と共に使用されることを想定しており、カレンダの周波数を高分解能で下げることができます。入力周波数が 32768 Hz の場合、32 秒枠の間に追加される RTCCLK パルスの数は次のように算出されます：(512 * CALP) - CALM

*セクション 46.3.13：RTC の高精度デジタル較正*を参照してください。

ビット 14 **CALW8**：8 秒較正サイクル周期の使用

CALW8 が“1”にセットされると、8 秒較正サイクル周期が選択されます。

注： ***CALW8 = 1 の場合、CALM[1:0] は“00”に固定されます。セクション 46.3.13：RTC の高精度デジタル較正を参照してください。***

ビット 13 **CALW16**：16 秒較正サイクル周期の使用

CALW16 が“1”にセットされると、16 秒較正サイクル周期が選択されます。CALW8 = 1 の場合、このビットを“1”にセットすることはできません。

注： ***CALW16 = 1 の場合、CALM[0] は“0”に固定されます。セクション 46.3.13：RTC の高精度デジタル較正を参照してください。***

ビット 12:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **CALM[8:0]**：較正マイナス

RTCCLK 2^{20} パルス（入力周波数が 32768 Hz の場合 32 秒）内の CALM をマスクすることによって、カレンダの周波数が下げられます。この方法により、カレンダの周波数を 0.9537 ppm の分解能で下げることができます。

カレンダの周波数を上げるには、この機能を CALP と共に使用する必要があります。*セクション 46.3.13： 1843 ページのRTC の高精度デジタル較正*を参照してください。

46.6.16 RTC タンパ設定レジスタ（RTC_TAMPCR）

アドレスオフセット：0x40

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP3NOERASE	TAMP3IE	TAMP2MF	TAMP2NOERASE	TAMP2IE	TAMP1MF	TAMP1NOERASE	TAMP1IE
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]			TAMPTS	TAMP3TRG	TAMP3E	TAMP2TRG	TAMP2E	TAMPIE	TAMP1TRG	TAMP1E
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **TAMP3MF**：タンパ 3 マスクフラグ

0：タンパ 3 イベントによってトリガイベントが生成され、TAMP3F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 3 イベントによってトリガイベントが生成されます。TAMP3F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタおよびバックアップ SRAM は消去されません。

注： TAMP3MF がセットされている場合、タンパ 3 割り込みを有効にすることはできません。

ビット 23 **TAMP3NOERASE**：タンパ 3 消去なし

0：タンパ 3 イベントでバックアップレジスタおよびバックアップ SRAM は消去されます。

1：タンパ 3 イベントでバックアップレジスタおよびバックアップ SRAM は消去されません。

ビット 22 **TAMP3IE**：タンパ 3 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 3 割り込みは無効です。

1：タンパ 3 割り込みは有効です。

ビット 21 **TAMP2MF**：タンパ 2 マスクフラグ

0：タンパ 2 イベントによってトリガイベントが生成され、TAMP2F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 2 イベントによってトリガイベントが生成されます。TAMP2F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタおよびバックアップ SRAM は消去されません。

注： TAMP2MF がセットされている場合、タンパ 2 割り込みを有効にすることはできません。

ビット 20 **TAMP2NOERASE**：タンパ 2 消去なし

0：タンパ 2 イベントでバックアップレジスタおよびバックアップ SRAM は消去されます。

1：タンパ 2 イベントでバックアップレジスタおよびバックアップ SRAM は消去されません。

ビット 19 **TAMP2IE**：タンパ 2 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 2 割り込みは無効です。

1：タンパ 2 割り込みは有効です。

ビット 18 **TAMP1MF**：タンパ 1 マスクフラグ

0：タンパ 1 イベントによってトリガイベントが生成され、TAMP1F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 1 イベントによってトリガイベントが生成されます。TAMP1F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタおよびバックアップ SRAM は消去されません。

注： TAMP1MF がセットされている場合、タンパ 1 割り込みを有効にすることはできません。

ビット 17 **TAMP1NOERASE**：タンパ 1 消去なし

0：タンパ 1 イベントでバックアップレジスタおよびバックアップ SRAM は消去されます。

1：タンパ 1 イベントでバックアップレジスタおよびバックアップ SRAM は消去されません。

ビット 16 **TAMP1IE**：タンパ 1 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 1 割り込みは無効です。

1：タンパ 1 割り込みは有効です。

ビット 15 **TAMPPUDIS**：RTC_TAMPx プルアップディセーブル

このビットにより、毎回のサンプリング前に各 RTC_TAMPx ピンをプリチャージするかどうか決定します。

0：サンプリング前に RTC_TAMPx ピンをプリチャージします（内部プルアップを有効化）。

1：RTC_TAMPx ピンのプリチャージを無効化します。

ビット 14:13 **TAMPPRCH[1:0]**：RTC_TAMPx プリチャージ時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCH は、各 RTC_TAMPx 入力に対して有効です。

0x0：1 RTCCLK サイクル

0x1：2 RTCCLK サイクル

0x2：4 RTCCLK サイクル

0x3：8 RTCCLK サイクル

ビット 12:11 **TAMPFLT[1:0]**：RTC_TAMPx フィルタカウント

これらのビットにより、タンパイベントをアクティブにするのに必要な指定のレベル（TAMP*TRG）での連続サンプリングの数を決定します。TAMPFLT は、各 RTC_TAMPx 入力に対して有効です。

0x0：RTC_TAMPx 入力がアクティブレベル（RTC_TAMPx 入力における内部プルアップ無し）に遷移するときのエッジで、タンパイベントがアクティブになります。

0x1：アクティブレベルでの連続した 2 回のサンプリングの後、タンパイベントがアクティブになります。

0x2：アクティブレベルでの連続した 4 回のサンプリングの後、タンパイベントがアクティブになります。

0x3：アクティブレベルでの連続した 8 回のサンプリングの後、タンパイベントがアクティブになります。

ビット 10:8 **TAMPFREQ[2:0]**：タンパサンプリング周波数

各 RTC_TAMPx 入力がサンプリングされる周波数を決定します。

0x0：RTCCLK / 32768（RTCCLK = 32768 Hz の場合 1 Hz）

0x1：RTCCLK / 16384（RTCCLK = 32768 Hz の場合 2 Hz）

0x2：RTCCLK / 8192（RTCCLK = 32768 Hz の場合 4 Hz）

0x3：RTCCLK / 4096（RTCCLK = 32768 Hz の場合 8 Hz）

0x4：RTCCLK / 2048（RTCCLK = 32768 Hz の場合 16 Hz）

0x5：RTCCLK / 1024（RTCCLK = 32768 Hz の場合 32 Hz）

0x6：RTCCLK / 512（RTCCLK = 32768 Hz の場合 64 Hz）

0x7：RTCCLK / 256（RTCCLK = 32768 Hz の場合 128 Hz）

ビット 7 **TAMPTS**：タンパ検出イベント時のタイムスタンプの有効化

0：タンパ検出イベントがあっても、タイムスタンプは保存されません。

1：タンパ検出イベント時、タイムスタンプが保存されます。

RTC_CR レジスタで TSE = 0 であっても TAMPTS は有効です。

ビット 6 **TAMP3TRG**：RTC_TAMP3 入力のアクティブレベル

TAMPFLT ≠ 00 の場合：

0：RTC_TAMP3 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP3 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP3 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP3 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 5 **TAMP3E**：RTC_TAMP3 検出イネーブル

0：RTC_TAMP3 入力検出は無効です。

1：RTC_TAMP3 入力検出は有効です。

ビット 4 **TAMP2TRG**：RTC_TAMP2 入力のアクティブレベル

TAMPFLT ! = 00 の場合

0：RTC_TAMP2 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP2 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP2 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP2 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 3 **TAMP2E**：RTC_TAMP2 入力検出イネーブル

0：RTC_TAMP2 検出は無効です。

1：RTC_TAMP2 検出は有効です。

ビット 2 **TAMPIE**：タンパ割り込みイネーブル

0：タンパ割り込みは無効です。

1：タンパ割り込みは有効です。

注： ***このビットは、TAMPxIE レベルにかかわらず、すべてのタンパピンイベントに対する割り込みを有効にします。このビットがクリアされた場合、TAMPxIE をセットすることによって、各タンパイベントの割り込みを個別に有効にすることができます。***

ビット 1 **TAMP1TRG**：RTC_TAMP1 入力のアクティブレベル

TAMPFLT ! = 00 の場合

0：RTC_TAMP1 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP1 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP1 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP1 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 0 **TAMP1E**：RTC_TAMP1 入力検出イネーブル

0：RTC_TAMP1 検出は無効です。

1：RTC_TAMP1 検出は有効です。

注意： ***TAMPFLT = 0 の場合、TAMPxE ビットは、TAMPxTRG が変更されたらリセットし、TAMPxF の不要なセットを回避する必要があります。***

46.6.17 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）

このレジスタは、RTC_CR レジスタの ALRAIE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、RTC レジスタ書き込み保護を参照してください。*1838 ページのRTC レジスタ書き込み保護*

アドレスオフセット：0x44

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 MASKSS[3:0]：このビットから始まる最上位ビットのマスク

0：アラーム A に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
1：アラーム A の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
2：アラーム A の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
3：アラーム A の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
...
12：アラーム A の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
13：アラーム A の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
14：アラーム A の比較では SS[14] を無視します。SS[13:0] が比較されます。
15：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 SS[14:0]：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム A をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

46.6.18 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CR レジスタの ALRBE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*セクション：RTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x48

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0x0：アラーム B に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
0x1：アラーム B の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
0x2：アラーム B の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
0x3：アラーム B の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
...
0xC：アラーム B の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
0xD：アラーム B の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
0xE：アラーム B の比較では SS[14] を無視します。SS[13:0] が比較されます。
0xF：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム B をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

46.6.19 RTC オプションレジスタ （RTC_OR）

アドレスオフセット：0x4C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTC_OUT_RMP	RTC_ALARM_TYPE
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **RTC_OUT_RMP**：RTC_OUT 再配置

このビットを設定することで、以下のように、PB2 の RTC 出力を再配置することが可能になります。

RTC_OUT_RMP = 0 の場合

OSEL が“00”以外のとき、PC13 に RTC_ALARM が出力されます。

OSEL が“00”かつ COE が“1”のとき、PC13 に RTC_CALIB が出力されます。

RTC_OUT_RMP = 1 の場合

OSEL が“00”以外かつ COE が“0”のとき、PB2に RTC_ALARM が出力されます。

OSEL が“00”かつ COE が“1”のとき、PB2に RTC_CALIB が出力されます。

OSEL が“00”以外かつ COE が“1”のとき、PB2 に RTC_CALIB が、PC13 に RTC_ALARM が出力されます。

ビット 0 **RTC_ALARM_TYPE**：PC13 の RTC_ALARM 出力形式

このビットは、ソフトウェアによってセット／クリアされます。

0：RTC_ALARM は、PC13 に配置される場合は、オープンドレイン出力です。

1：RTC_ALARM は、PC13 に配置される場合は、プッシュプル出力です。

46.6.20 RTC バックアップレジスタ（RTC_BKPxR）

アドレスオフセット：0x50 から 0xCC

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKP[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BKP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:0 BKP[31:0]

アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。
これらのレジスタは、V_{DD} がオフになった場合、V_{BAT} によって電源が供給されるため、システムリセットによりリセットされず、デバイスが低電力モードで動作する場合、レジスタの内容は有効なまま保持されます。
このレジスタは、TAMPxF=1 である限り、タンパ検出イベントでリセットされます。

46.6.21 RTC レジスタマップ

表 358. RTC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RTC_TR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT [1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x04	RTC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]				WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0]			
	リセット値									0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1			0	0	0	0	0	1
0x08	RTC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSE	COE	OSEL [1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
	リセット値								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x0C	RTC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	RECALPF	TAMP3F	TAMP2F	TAMP1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
	リセット値															0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x10	RTC_PRER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]							PREDIV_S[14:0]															
	リセット値										1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x14	RTC_WUTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUT[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	RTC_ALRMAR	MSK4	WDSEL	DT [1:0]		DU[3:0]				MSK3	PM	HT [1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	RTC_ALRMBR	MSK4	WDSEL	DT [1:0]		DU[3:0]				MSK3	PM	HT [1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK2	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	RTC_WPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY							
	リセット値																									0	0	0	0	0	0	0	0
0x28	RTC_SSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBFS[14:0]														
	リセット値	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 358. RTC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	**RTC_TSTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x34	**RTC_TSDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
	リセット値																	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x38	**RTC_TSSSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**RTC_ CALR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
	リセット値																	0	0	0					0	0	0	0	0	0	0	0	0
0x40	**RTC_TAMPCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP3NOERASE	TAMP3IE	TAMP2MF	TAMP2NOERASE	TAMP2IE	TAMP1MF	TAMP1NOERASE	TAMP1IE	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]		TAMPTS	TAMP3TRG	TAMP3E	TAMP2TRG	TAMP2E	TAMPIE	TAMP1TRG	TAMP1E	
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	**RTC_ ALRMASSR**	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**RTC_ALRMBSSR**	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	**RTC_ OR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTC_OUT_RMP	RTC_ALARM_TYPE
	リセット値																															0	0
0x50 ～0xCC	**RTC_BKP0R**	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	to **RTC_BKP31R**	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

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

47.1 概要

I²C（Inter-integrated circuit）バスインタフェースは、マイクロコントローラとシリアル I²C バス間の通信を処理します。マルチマスタ機能を備え、すべての I²C バス固有のシーケンシング、プロトコル、アービトレーション、およびタイミングを制御します。標準モード（Sm）、高速モード（Fm）、および高速モードプラス（Fm+）をサポートします。

また、SMBus（System Management Bus）および PMBus（Power Management Bus）と互換性があります。

DMA を使用して、CPU の負荷を軽減できます。

47.2 I2C の主な機能

- I²C バス仕様 rev03 との互換性：
  - スレーブおよびマスタモード
  - マルチマスタ機能
  - 標準モード（最大 100 kHz）
  - 高速モード（最大 400 kHz）
  - 高速モードプラス（最大 1 MHz）
  - 7 ビットおよび 10 ビットアドレッシングモード
  - 複数の 7 ビットスレーブアドレス（2 つのアドレス、1 つは設定可能なマスク付き）
  - すべての 7 ビットアドレス確認応答モード
  - 同報（General call）コール
  - プログラム可能なセットアップおよびホールド時間
  - 使いやすいイベント管理
  - クロックストレッチオプション
  - ソフトウェアリセット
- DMA 機能付きの 1 バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます（*セクション 47.3：I2C の実装* を参照）：

- SMBus 仕様 rev 2.0 との互換性：
  - ハードウェア PEC（Packet Error Checking）の生成と ACK 制御による確認
  - コマンドおよびデータ確認応答制御
  - アドレス解決プロトコル（ARP）サポート
  - ホストおよびデバイスのサポート
  - SMBus アラート
  - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.1 標準との互換性
- 独立したクロック：独立したクロックソースの選択により、I2C の通信速度は i2c_pclk の再プログラミングから独立
- アドレス一致時に STOP モードからウェイクアップ

47.3 I2C の実装

このマニュアルでは I2C ペリフェラルで実装されているすべての機能を説明します。STM32H7xxx デバイスでは、I2C1、I2C2、I2C3、および I2C4 は、次の表に示すように、すべての機能を実装しています。

表 359. STM32H7x3I2C の実装

I2C の機能(1)	I2C1	I2C2	I2C3	I2C4
7 ビットアドレスモード	X	X	X	X
10 ビットアドレスモード	X	X	X	X
標準モード（最大 100 kbit/s）	X	X	X	X
高速モード（最大 400 kbit/s）	X	X	X	X
20mA 出力駆動 I/O 搭載高速モードプラス（最大 1 Mbit/s）	X	X	X	X
独立クロック	X	X	X	X
SMBus	X	X	X	X
STOP モードからのウェイクアップ	X	X	X	X

1. X : サポートされています。

47.4 I2C の機能詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割り込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によって I2C バスに接続されます。標準（最大 100 kHz）、高速モード（最大 400 kHz）、または高速モードプラス（最大 1 MHz）の I^2C バスで接続できます。

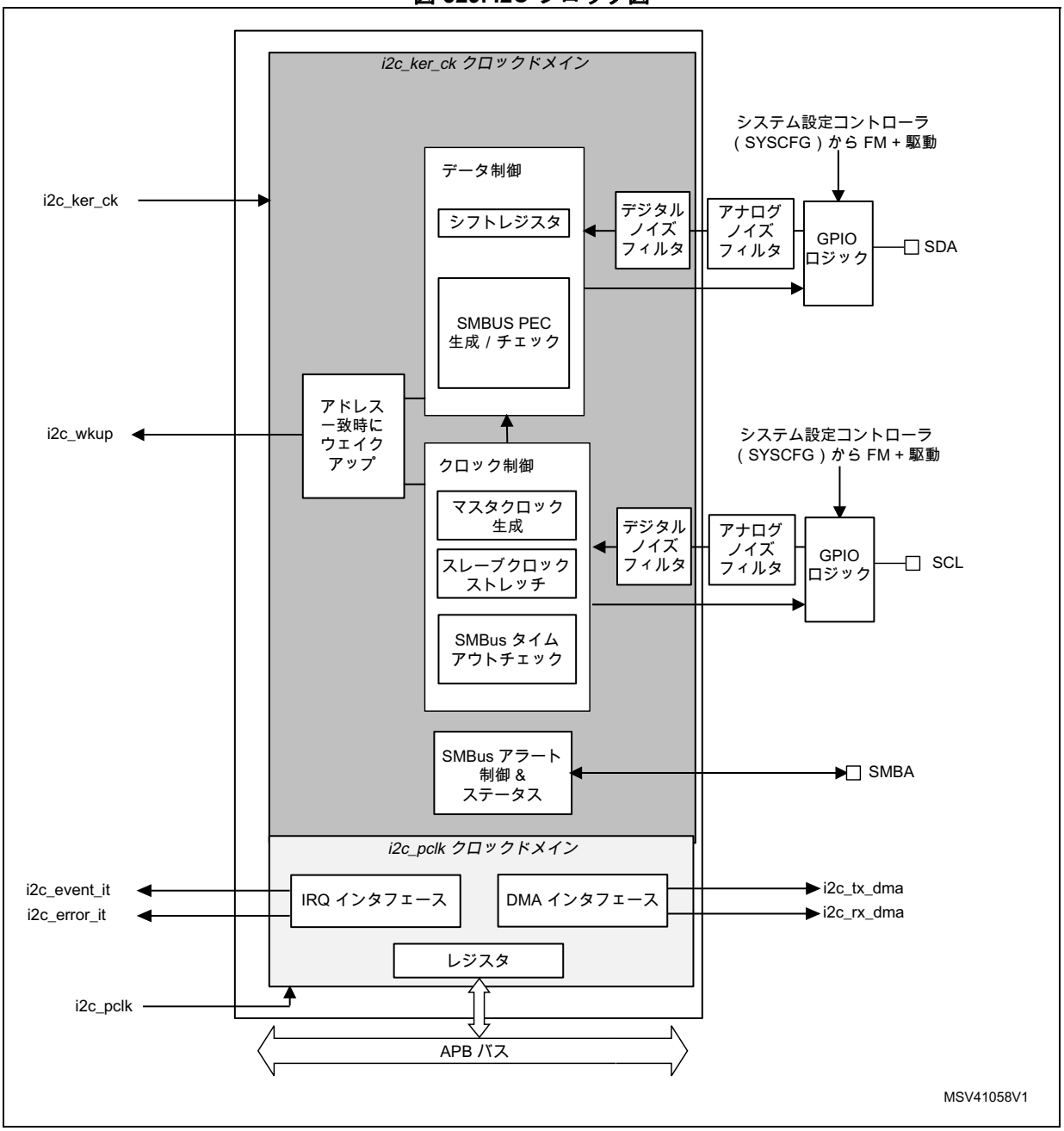
このインタフェースは、データピン（SDA）とクロックピン（SCL）によって SMBus に接続することもできます。

SMBus 機能がサポートされる場合、追加の SMBus アラートピン（SMBA）オプションも使用できます。

47.4.1 I2C ブロック図

I2C インタフェースのブロック図を *図 529* に示します。

図 529. I2C ブロック図



I2C は、独立したクロックソースによってクロック供給されるため、I2C は i2c_pclk 周波数から独立して動作できます。

詳細については、*図 43 : I2C のカーネルクロック分配*を参照してください。

I2C I/O は、高速モードプラス動作のための 20 mA 出力電流駆動をサポートします。これを有効にするには、*セクション 12：システム設定コントローラ（SYSCFG）の SCL および SDA の駆動機能制御ビットをセットします。*

47.4.2 I2C クロックの要件

I2C カーネルは i2c_ker_ck によってクロック供給されます。

i2c_ker_ck の周期 t_{I2CCLK} は、次の条件を満たす必要があります。

$t_{I2CCLK} < (t_{LOW} - t_{filters}) / 4$ および $t_{I2CCLK} < t_{HIGH}$

ここで：

t_{LOW}：SCL ロー時間、および t_{HIGH}：SCL ハイ時間

$t_{filters}$：有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

アナログフィルタの遅延は、最大 260 ns です。デジタルフィルタの遅延は、DNF x t_{I2CCLK} です。

i2c_pclk の周期 t_{PCLK} は、次の条件を満たす必要があります。

$t_{PCLK} < 4/3\ t_{SCL}$

t_{SCL}：SCL 周期

注意： ***I2C カーネルが i2c_pclk によってクロック供給されるとき、I2c_pclk は t_{I2CCLK} の条件を満たす必要があります。***

47.4.3 モード選択

このインタフェースは、次の 4 つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、START コンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失または STOP 生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

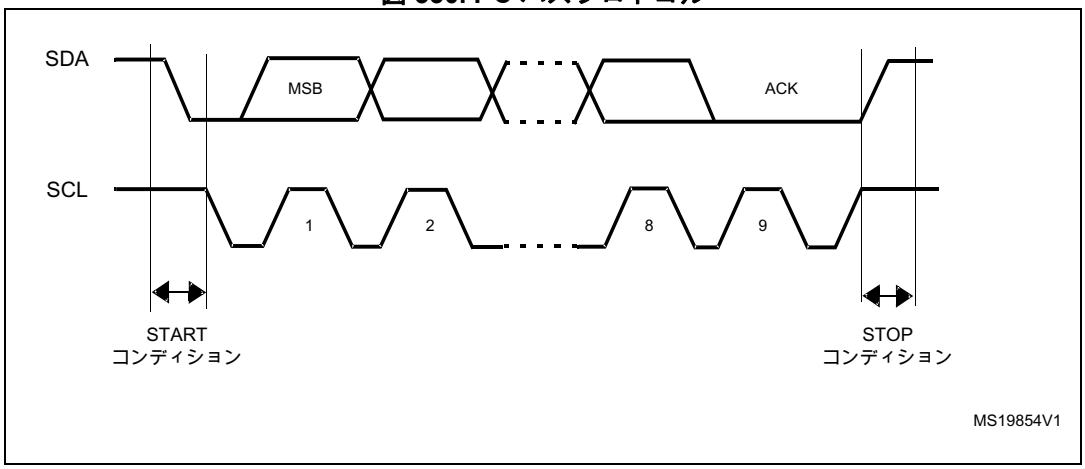
マスタモードでは、I2C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常に START コンディションで開始され、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

スレーブモードでは、このインタフェースは、自己アドレス（7 または 10 ビット）と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みの SMBus アドレスもソフトウェアによって有効にできます。

データとアドレスは、MSB ファーストの 8 ビットバイトとして転送されます。START コンディションの後に続く最初のバイト（7 ビットモードでは 1 バイト、10 ビットモードでは 2 バイト）にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8 クロックサイクルのバイト転送の後には 9 番目のクロックパルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります。次の図を参照してください。

図 530. I^2C バスプロトコル



確認応答（Acknowledge）は、ソフトウェアによって有効または無効にできます。I2C インタフェースのアドレスは、ソフトウェアによって選択できます。

47.4.4 I2C の初期化

ペリフェラルの有効化と無効化

I2C ペリフェラルクロックは、クロックコントローラで設定し、有効にする必要があります（*セクション 8：リセットおよびクロック制御（RCC）*を参照）。

そして、I2C_CR1 レジスタの PE ビットをセットすることによって、I2C を有効にできます。

I2C が無効なときには（PE=0）、I^2C はソフトウェアリセットを実行します。詳細については、*セクション 47.4.5：ソフトウェアリセット* を参照してください。

ノイズフィルタ

I2C_CR1 レジスタの PE ビットをセットすることによって I2C ペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDA および SCL 入力にアナログノイズフィルタがあります。このアナログフィルタは I^2C 仕様に準拠しており、高速モードおよび高速モードプラスで最大 50 ns のパルス幅を持つスパイクを抑制します。ANFOFF ビットをセットすることによって、このアナログフィルタを無効にし、I2C_CR1 レジスタの DNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCL または SDA ラインのレベルは、DNF x i2c_ker_ck 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な 1 ～ 15 i2c_ker_ck 周期の長さを持つスパイクを抑制できます。

表 360. アナログフィルタとデジタルフィルタの比較

	アナログフィルタ	デジタルフィルタ
抑制されるスパイクのパルス幅	≥ 50 ns	長さを 1 ～ 15 I2C ペリフェラルクロックにプログラム可能
利点	STOP モードで使用可能	– プログラム可能な長さ : 追加のフィルタリング機能対標準要件 – 安定した長さ
欠点	温度、電圧、プロセスのばらつき	デジタルフィルタが有効なときには、アドレス一致時の STOP モードからのウェイクアップは使用できない

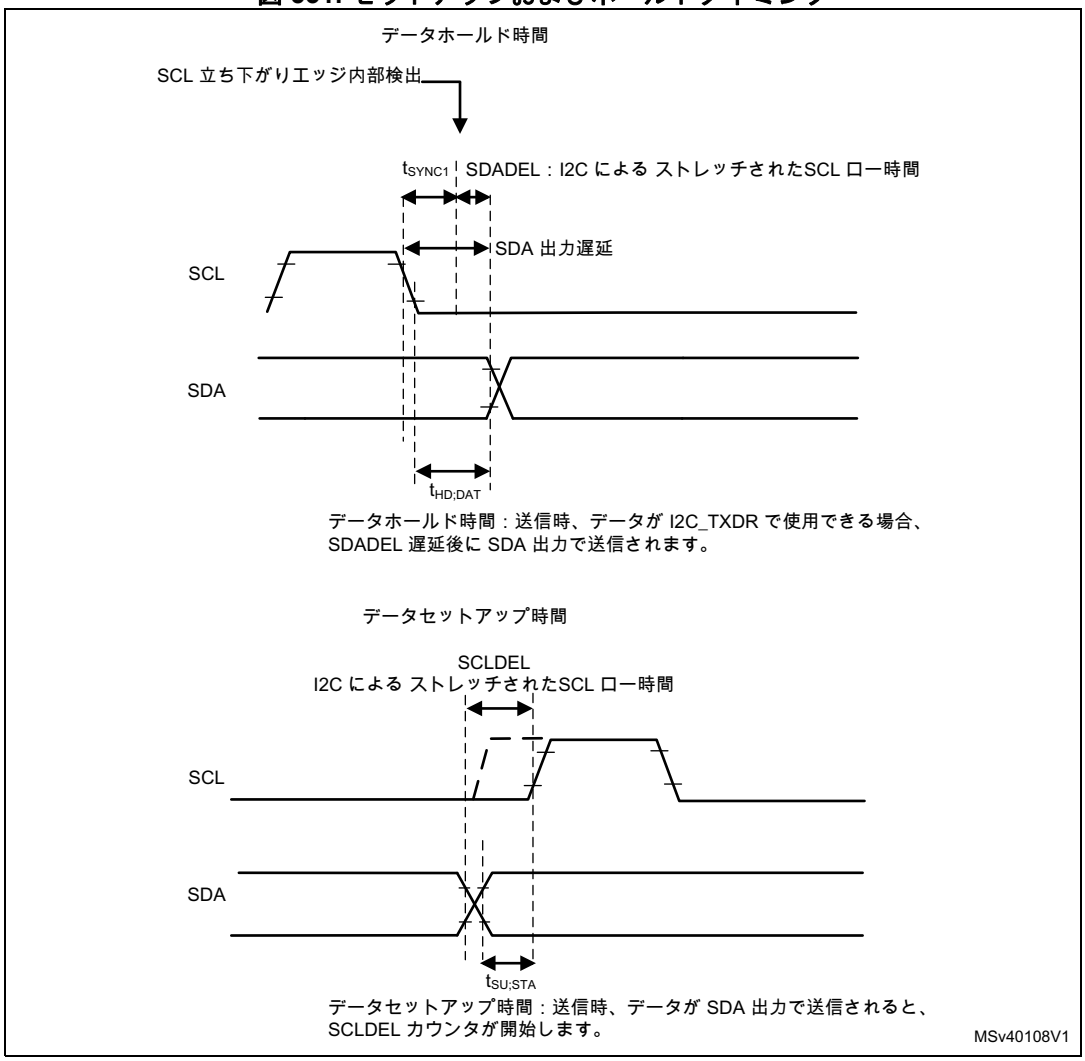
注意： ***I2C が有効なときには、フィルタ構成の変更はできません。***

I2C のタイミング

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間が使用されるのを保証するためには、タイミングを設定する必要があります。これを行うには、I2C_TIMINGR レジスタの PRESC[3:0]、SCLDEL[3:0]、および SDADEL[3:0] ビットをプログラムします。

STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

図 531. セットアップおよびホールドタイミング



- SCL 立ち下がりエッジが内部で検出されると、SDA 出力を送信する前に遅延が挿入されます。この遅延は、t_{SDADEL} = SDADEL x t_{PRESC} + t_{I2CCLK} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  T_{SDADEL} はホールド時間 $t_{HD;DAT}$ に影響を与えます。

SDA 出力遅延の合計は、次のとおりです：

t_{SYNC1} + {[SDADEL x (PRESC+1) + 1] x t_{I2CCLK} }

t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- アナログフィルタが有効なときの入力遅延：$t_{AF(min)} < t_{AF} < t_{AF(max)}$ ns
- デジタルフィルタが有効なときの入力遅延：t_{DNF} = DNF x t_{I2CCLK}
- SCL と i2c_ker_ck クロックの同期による遅延（2 ～ 3 i2c_ker_ck 周期）

SCL 立ち下がりエッジの未定義の領域をブリッジするためには、SDADEL を次のようにプログラムする必要があります：

$\{t_{f\ (max)} + t_{HD;DAT\ (min)} - t_{AF(min)} - [(DNF + 3) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\} \leq SDADEL$

$SDADEL \leq \{t_{HD;DAT\ (max)} - t_{AF(max)} - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： ***$t_{AF(min)}$ / $t_{AF(max)}$ は、アナログフィルタが有効なときのみ、等式に含まれます。t_{AF} の値については、デバイスのデータシートを参照してください。***

最大 $t_{HD;DAT}$ は、標準モード、高速モード、および高速モードプラスで 3.45 µs、0.9 µs、および 0.45 µs ですが、遷移時間による $t_{VD;DAT}$ の最大値より短い必要があります。この最大値を満たす必要があるのは、デバイスが SCL 信号の LOW 周期（t_{LOW}）をストレッチしない場合だけです。クロックが SCL をストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

SDA 立ち上がりエッジは、通常、最悪ケースであり、この場合、前の等式は次のようになります：

$SDADEL \leq \{t_{VD;DAT\ (max)} - t_{r\ (max)} - 260\ ns - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： ***NOSTRETCH=0 のときには、SCLDEL の値に従って、デバイスはセットアップ時間を保証するために SCL ロー をストレッチするので、この条件に違反することがあります。***

t_f、t_r、$t_{HD;DAT}$、および $t_{VD;DAT}$ の標準値については、*表 361：I2C-SMBUS 仕様のデータのセットアップおよびホールド時間* を参照してください。

- t_{SDADEL} 遅延の後や、データが I2C_TXDR レジスタにまだ書き込まれていないためスレーブがクロックをストレッチしなければならなかった場合の SDA 出力送信後、SCL ラインはセットアップ時間中、ローレベルで保持されます。このセットアップ時間は、t_{SCLDEL} = (SCLDEL+1) x t_{PRESC} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLDEL} は、セットアップ時間 $t_{SU;DAT}$ に影響を与えます。

SDA 遷移（立ち上がりエッジは通常、最悪のケース）の未定義の領域をブリッジするためには、SCLDEL を次のようにプログラムする必要があります：

$\{[t_{r\ (max)} + t_{SU;DAT\ (min)}] / [(PRESC+1)] \times t_{I2CCLK}]\} - 1 <= SCLDEL$

t_r および $t_{SU;DAT}$ の標準値については、*表 361：I2C-SMBUS 仕様のデータのセットアップおよびホールド時間* を参照してください。

使用される SDA および SCL 遷移時間の値は、アプリケーションの値です。標準から最大値を使用すると、SDADEL と SCLDEL の計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

注： ***各クロックパルスで、SCL 立ち下がりエッジの検出後、I2C マスタまたはスレーブは、最低 [(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、送信および受信の両モードで SCL ローをストレッチします。送信モードで、SDADEL カウンタ終了時にデータがまだ I2C_TXDR に書き込まれていない場合、I2C は次のデータが書き込まれるまで SCL ローをストレッチし続けます。その時、新しいデータ MSB が SDA 出力で送信され、SCLDEL カウンタが開始し、SCL ローのストレッチを継続して、データセットアップ時間を保証します。***

スレーブモードで NOSTRETCH=1 の場合、SCL はストレッチされません。そのため、十分なセットアップ時間を保証するためにも、SDADEL はこのようにプログラムされる必要があります。

表 361. I^2C-SMBUS 仕様のデータのセットアップおよびホールド時間

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBUS		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
$t_{HD;DAT}$	データホールド時間	0	-	0	-	0	-	0.3	-	µs
$t_{VD;DAT}$	データ有効時間	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	データセットアップ時間	250	-	100	-	50	-	250	-	ns
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	

また、マスタモードでは、I2C_TIMINGR レジスタの PRESC[3:0]、SCLH[7:0]、および SCLL[7:0] ビットをプログラムすることによって、SCL クロックのハイおよびローレベルを設定する必要があります。

- SCL 立ち下がりエッジが内部で検出されると、SCL 出力をリリースする前に遅延が挿入されます。この遅延は、t_{SCLL} = (SCLL+1) x t_{PRESC} であり、t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLL} は、SCL ロー時間 t_{LOW} に影響を与えます。
- SCL 立ち上がりエッジが内部で検出されると、SCL 出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、t_{SCLH} = (SCLH+1) x t_{PRESC} であり、t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。t_{SCLH} は、SCL ハイ時間 t_{HIGH} に影響を与えます。

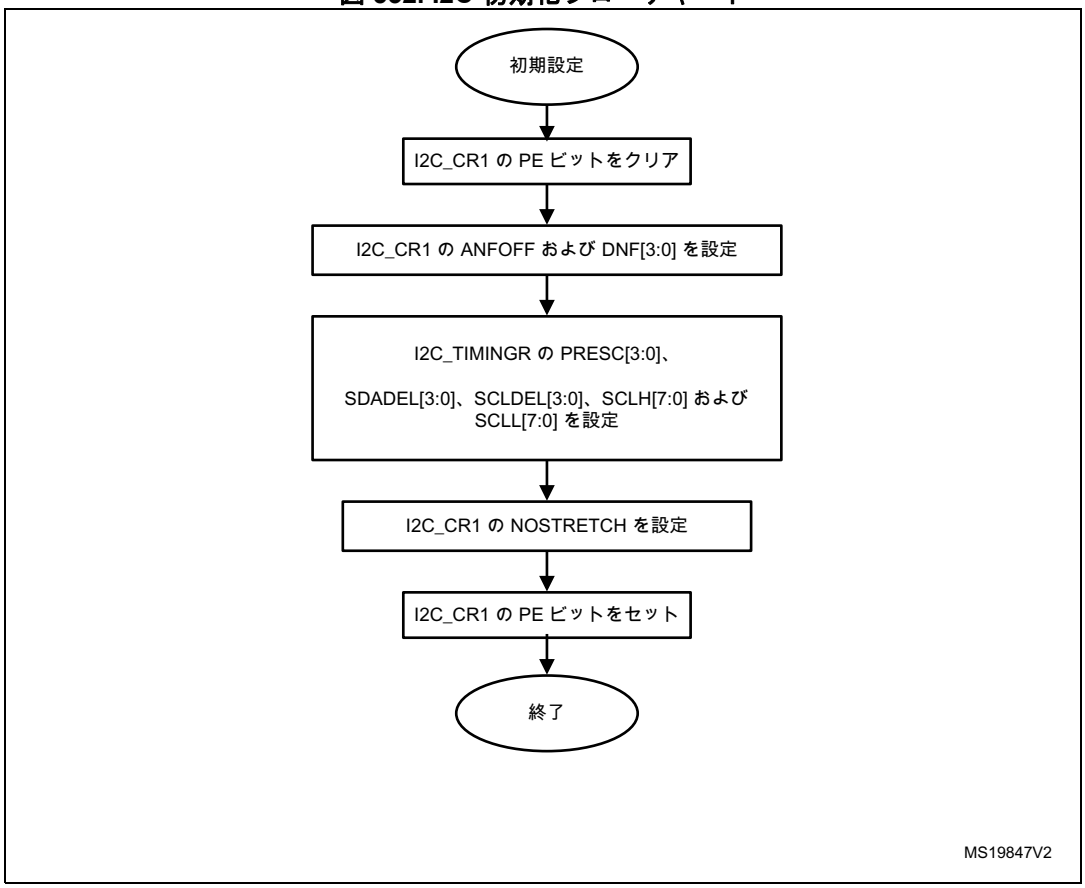
詳細については、*I2C マスタ初期化* を参照してください。

注意： ***I2C が有効なときには、タイミング構成の設定はできません。***

ペリフェラルを有効にする前に、I2C スレーブ NOSTRETCH モードも設定する必要があります。詳細については、*I2C スレーブ初期化* を参照してください。

注意： ***I2C が有効なときには、NOSTRETCH 構成の変更はできません。***

図 532. I2C 初期化フローチャート



47.4.5 ソフトウェアリセット

ソフトウェアリセットを行うには、I2C_CR1 レジスタの PE ビットをクリアします。その場合、I2C のライン SCL および SDA がリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2C_CR2 レジスタ：START、STOP、NACK
2. I2C_ISR レジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus 機能がサポートされるときには、以下も影響を受けます：

1. I2C_CR2 レジスタ：PECBYTE
2. I2C_ISR レジスタ：PECERR、TIMEOUT、ALERT

ソフトウェアリセットを実行するためには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれなければなりません。このためには、次のソフトウェアシーケンスを書き込みます：- PE=0 を書き込む - PE=0 を確認する - PE=1 を書き込む。

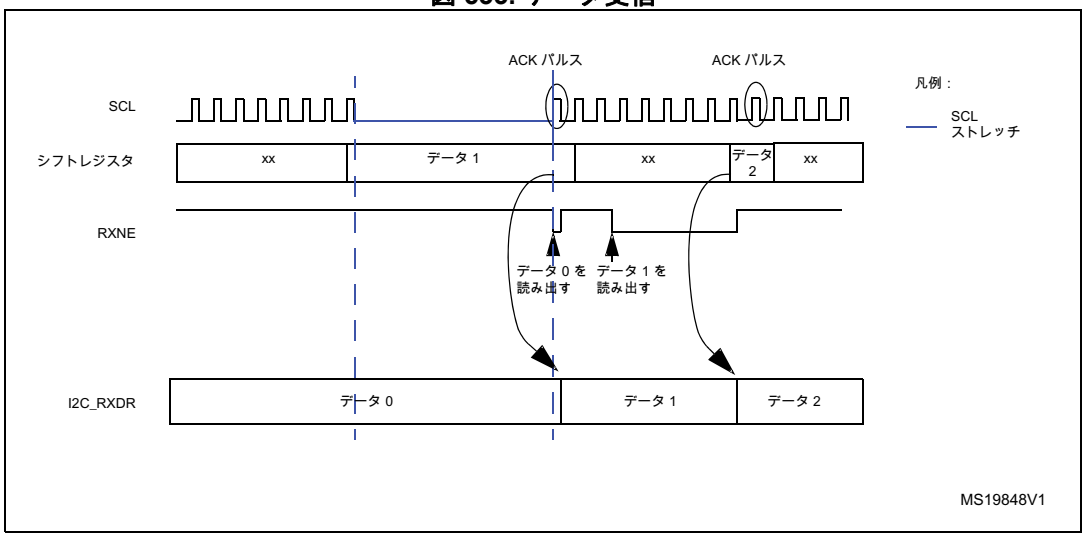
47.4.6 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA 入力はシフトレジスタに送られます。8 番目の SCL パルスの後（完全なデータバイトの受信後）、シフトレジスタは、I2C_RXDR レジスタが空の場合（RXNE=0）、このレジスタにコピーされます。RXNE=1 の場合、すなわち、前に受信されたデータバイトがまだ読み出されていなかった場合、SCL ラインは I2C_RXDR が読み出されるまでストレッチされます。ストレッチは、8 番目と 9 番目の SCL パルスの間（確認応答パルスの前）に挿入されます。

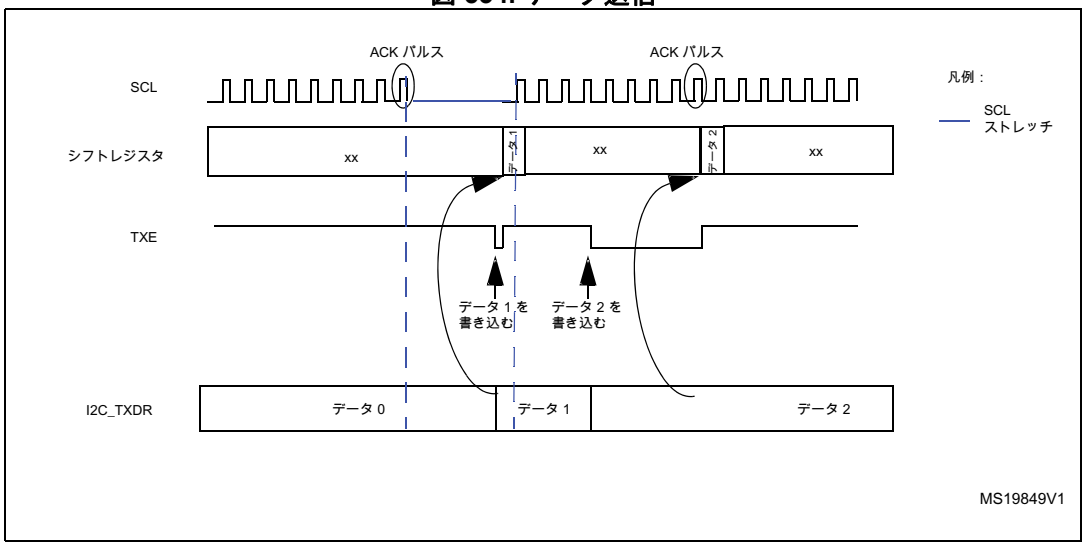
図 533. データ受信



送信

I2C_TXDR レジスタが空（TXE=0）でない場合、その内容が 9 番目の SCL パルス（確認応答パルス）の後、シフトレジスタにコピーされます。次に、シフトレジスタの内容が SDA ラインにシフトアウトされます。TXE=1 の場合、すなわち、I2C_TXDR にデータがまだ書き込まれていない場合、SCL ラインは I2C_TXDR に書き込まれるまでストレッチされます。ストレッチは、9 番目の SCL パルスの後で行われます。

図 534. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2C にはハードウェアにバイトカウンタが組み込まれています。

- マスタモードでの NACK、STOP、および ReSTART 生成
- スレーブレシーバモードでの ACK 制御
- SMBus 機能がサポートされているときの PEC 生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2C_CR2 レジスタの SBC（スレーブバイト制御）ビットをセットすることによって、ソフトウェアにより有効にできます。

転送されるバイト数は、I2C_CR2 レジスタの NBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数（NBYTES）が 255 より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTES でプログラムされたバイト数が転送されると、TCR フラグがセットされ、TCIE がセットされている場合は割り込みが生成されます。SCL は、TCR フラグがセットされている間、ストレッチされます。TCR は、NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTES カウンタに最後のバイト数が再ロードされたときには、RELOAD ビットがクリアされる必要があります。

マスタモードで RELOAD=0 のときには、カウンタは 2 つのモードで使用できます：

- **自動終了モード**（I2C_CR2 レジスタの AUTOEND = “1”）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタは STOP コンディションを自動的に送信します。
- **ソフトウェア終了モード**（I2C_CR2 レジスタの AUTOEND = 0）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェアアクションが求められます。TC フラグがセットされ、TCIE ビットがセットされている場合は割り込みが生成されます。SCL 信号は、TC フラグがセットされている間、ストレッチされます。TC フラグは、I2C_CR2 レジスタの START または STOP ビットがセットされたときに、ソフトウェアによってクリアされます。マスタが RESTART コンディションを送信したいときには、このモードを使用する必要があります。

注意： ***AUTOEND ビットは、RELOAD ビットがセットされているときには効果がありません。***

表 362. I2C 設定表

機能	SBC ビット	RELOAD ビット	AUTOEND ビット
マスタ Tx/Rx NBYTES + STOP	x	0	1
マスタ Tx/Rx + NBYTES + RESTART	x	0	0
スレーブ Tx/Rx すべての受信バイトに ACK	0	x	x
スレーブ Rx および ACK 制御	1	1	x

47.4.7 I2C スレーブモード

I2C スレーブ初期化

スレーブモードで動作するには、少なくとも 1 つのスレーブアドレスを有効にする必要があります。2 つのレジスタ I2C_OAR1 と I2C_OAR2 を使用して、スレーブ専用アドレス OA1 および OA2 をプログラムできます。

- OA1 は、I2C_OAR1 レジスタの OA1MODE ビットをセットすることによって、7 ビットモード（デフォルト）または 10 ビットアドレッシングモードに設定できます。

  OA1 を有効にするには、I2C_OAR1 レジスタの OA1EN ビットをセットします。
- 追加のスレーブアドレスが必要な場合は、2 番目のスレーブアドレス OA2 を設定できます。I2C_OAR2 レジスタの OA2MSK[2:0] ビットを設定することによって、最大 7 つの OA2 LSB をマスクできます。したがって、OA2MSK が 1 から 6 まで設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、または OA2[7] のみが受信アドレスと比較されます。OA2MSK が 0 に等しくなくなるとすぐに、OA2 のアドレスコンパレータは、確認応答されない I2C 予約済みアドレス（0000 XXX および 1111 XXX）を除外します。OA2MSK=7 の場合、受信されたすべてのアドレスが確認応答されます（予約済みアドレスを除く）。OA2 は常に 7 ビットアドレスです。

  これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2C_OAR1 または I2C_OAR2 レジスタが OA2MSK=0 でプログラムされた場合、確認応答できます。

  OA2 を有効にするには、I2C_OAR2 レジスタの OA2EN ビットをセットします。
- 同報アドレスは、I2C_CR1 レジスタの GCEN ビットをセットすることで有効になります。

I2C が有効アドレスの 1 つによって選択されると、ADDR 割り込みステータスフラグがセットされ、ADDRIE ビットがセットされている場合は割り込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用し、必要なときには、ソフトウェアアクションを実行するために、SCL 信号をローレベルでストレッチすることを意味します。マスタがクロックストレッチをサポートしない場合、I2C_CR1 レジスタの NOSTRETCH=1 で I2C を設定する必要があります。

ADDR 割り込みの受信後、いくつかのアドレスが有効な場合は、I2C_ISR レジスタの ADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIR フラグも確認する必要があります。

スレーブクロックストレッチ（NOSTRETCH = 0）

デフォルトモードでは、I2C スレーブは次の状況で SCL クロックをストレッチします :

- ADDR フラグがセットされると : 受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCF ビットをセットすることによりソフトウェアによって ADDR フラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了し、新しいデータが I2C_TXDR レジスタに書き込まれなかった場合、または ADDR フラグがクリアされたときに（TXE=1）、最初のデータバイトが書き込まれていなかった場合。このストレッチは、データが I2C_TXDR レジスタに書き込まれたときにリリースされます。
- 受信時、I2C_RXDR レジスタがまだ読み出されておらず、新しいデータ受信が完了したとき。このストレッチは、I2C_RXDR が読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード（SBC=1 および RELOAD=1）で TCR = 1 のとき、すなわち、最後データバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによって TCR がクリアされたときにリリースされます。
- SCL 立ち下がりエッジの検出後、I2C は、[(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、SCL ローをストレッチします。

クロックストレッチなしのスレーブ（NOSTRETCH = 1）

I2C_CR1 レジスタの NOSTRETCH = 1 のとき、I2C スレーブは SCL 信号をストレッチしません。

- ADDR フラグがセットされている間、SCL クロックはストレッチされません。
- 送信時、転送に対応する最初の SCL パルスが発生する前に、I2C_TXDR レジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2C_ISR レジスタで OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。OVR フラグは、最初のデータ送信が開始し、STOPF ビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後でのみ、前に転送の STOPF フラグをクリアすることによって、送信される最初のデータについても、OVR ステータスが提供されることを確実にできます。
- 受信時、次のデータバイトの 9 番目の SCL パルス（ACK パルス）が発生する前に、I2C_RXDR レジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

スレーブバイト制御モード

スレーブ受信モードでバイト ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。これは、SMBus 標準に準拠する必要があります。

スレーブ受信モードでバイト ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。各バイトの制御を得るには、ADDR 割り込みサブルーチンで NBYTES を 0x1 に初期化し、各受信バイト後に 0x1 に再ロードする必要があります。バイトが受信されると、TCR ビットがセットされ、8 番目と 9 番目の SCL パルスの間で、SCL 信号ローをストレッチします。I2C_RXDR レジスタからデータを読み出すことができ、その後、I2C_CR2 レジスタの ACK ビットを設定することによって、確認応答するかどうかを決定できます。SCL ストレッチは、NBYTES をゼロ以外の値にプログラムすることによってリリースされ、確認応答または非確認応答が送信され、次のバイトを受信できます。

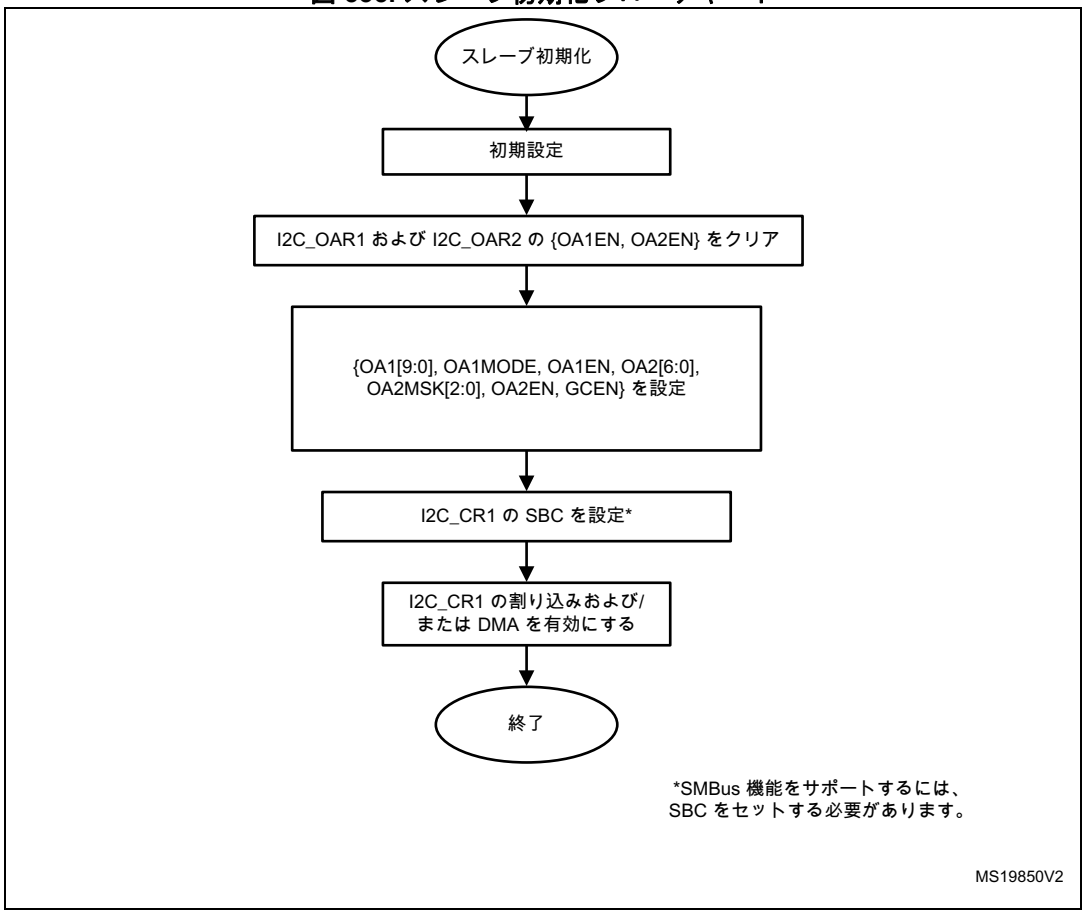
NBYTES に 0x1 より大きい値をロードでき、この場合、受信フローは NBYTES データ受信中、継続します。

注： ***SBC ビットは、I2C が無効なとき、またはスレーブがアドレス指定されていないとき、または ADDR=1 のときに設定する必要があります。***

RELOAD ビットの値は、ADDR=1 のとき、または TCR=1 のときに変更できます。

注意： ***スレーブバイト制御モードは、NOSTRETCH モードと互換性がありません。NOSTRETCH=1 のときに SBC をセットすることはできません。***

図 535. スレーブ初期化フローチャート



スレーブトランスミッタ

I2C_TXDR レジスタが空になると、送信割り込みステータス（TXIS）が生成されます。I2C_CR1 レジスタの TXIE ビットがセットされている場合は、割り込みが生成されます。

TXIS ビットは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACK が受信されると、I2C_ISR レジスタの NACKF ビットがセットされ、I2C_CR1 レジスタの NACKIE ビットがセットされていた場合は割り込みが生成されます。マスタが STOP または RESTART コンディションを実行できるように、スレーブは SCL および SDA ラインを自動的にリリースします。TXIS ビットは、NACK 受信時にはセットされません。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR レジスタの STOPF フラグがセットされ、割り込みが生成されます。ほとんどのアプリケーションでは、SBC は通常、0 にプログラムされます。この場合、スレーブアドレスが受信されたときに（ADDR=1）、TXE = 0 であった場合、I2C_TXDR レジスタの内容を最初のデータバイトとして送信するか、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをフラッシュするかを選択できます。

スレーブバイト制御モード（SBC=1）では、送信バイト数をアドレス一致割り込みサブルーチンの NBYTES でプログラムする必要があります（ADDR=1）。この場合、転送中の TXIS イベントの数は、NBYTES でプログラムされた値に対応します。

注意： ***NOSTRETCH=1 のとき、SCL クロックは ADDR フラグがセットされている間はストレッチされないので、最初のデータバイトをプログラムするために ADDR サブルーチンで I2C_TXDR レジスタの内容をフラッシュすることはできません。最初に送信されるデータバイトは、I2C_TXDR レジスタで前もってプログラムされている必要があります。***

- このデータは、前の送信メッセージの最後の TXIS イベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをフラッシュできます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPF ビットのクリアは、これらのアクションの後でのみ行う必要があります。

  最初のデータ送信が開始したときに STOPF がまだセットされていた場合、アンダーランエラーが生成されます（OVR フラグがセットされます）。

  TXIS イベントが必要な場合（送信割り込みまたは送信 DMA リクエスト）、TXIS イベントを生成するためには、TXE ビットに加えて TXIS ビットもセットする必要があります。

図 536. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=0）

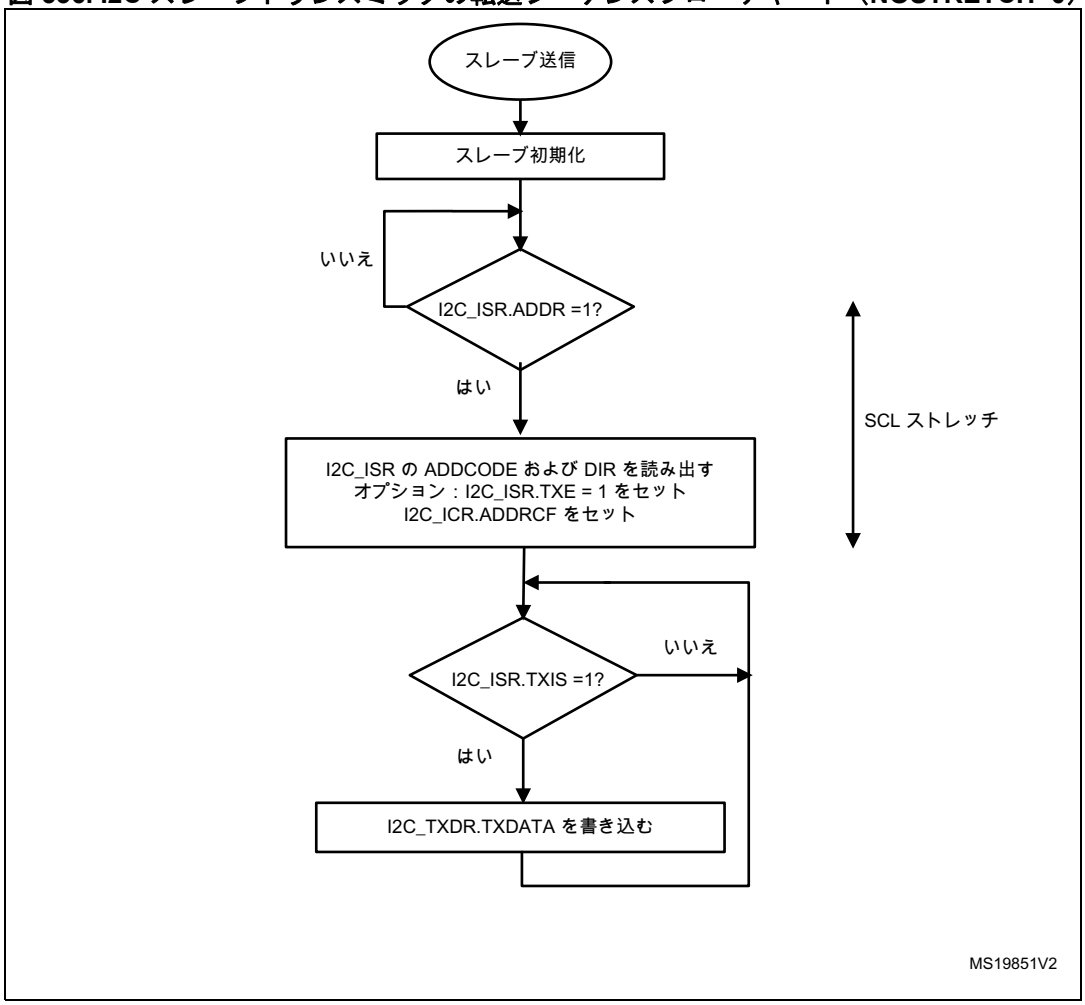
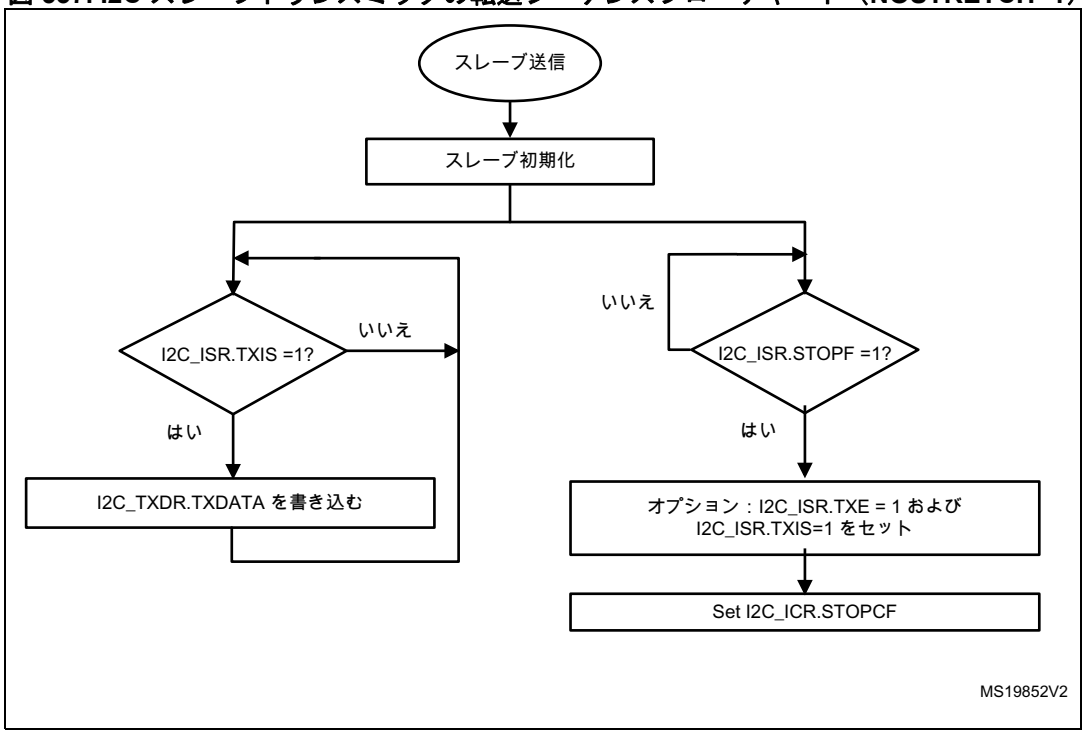
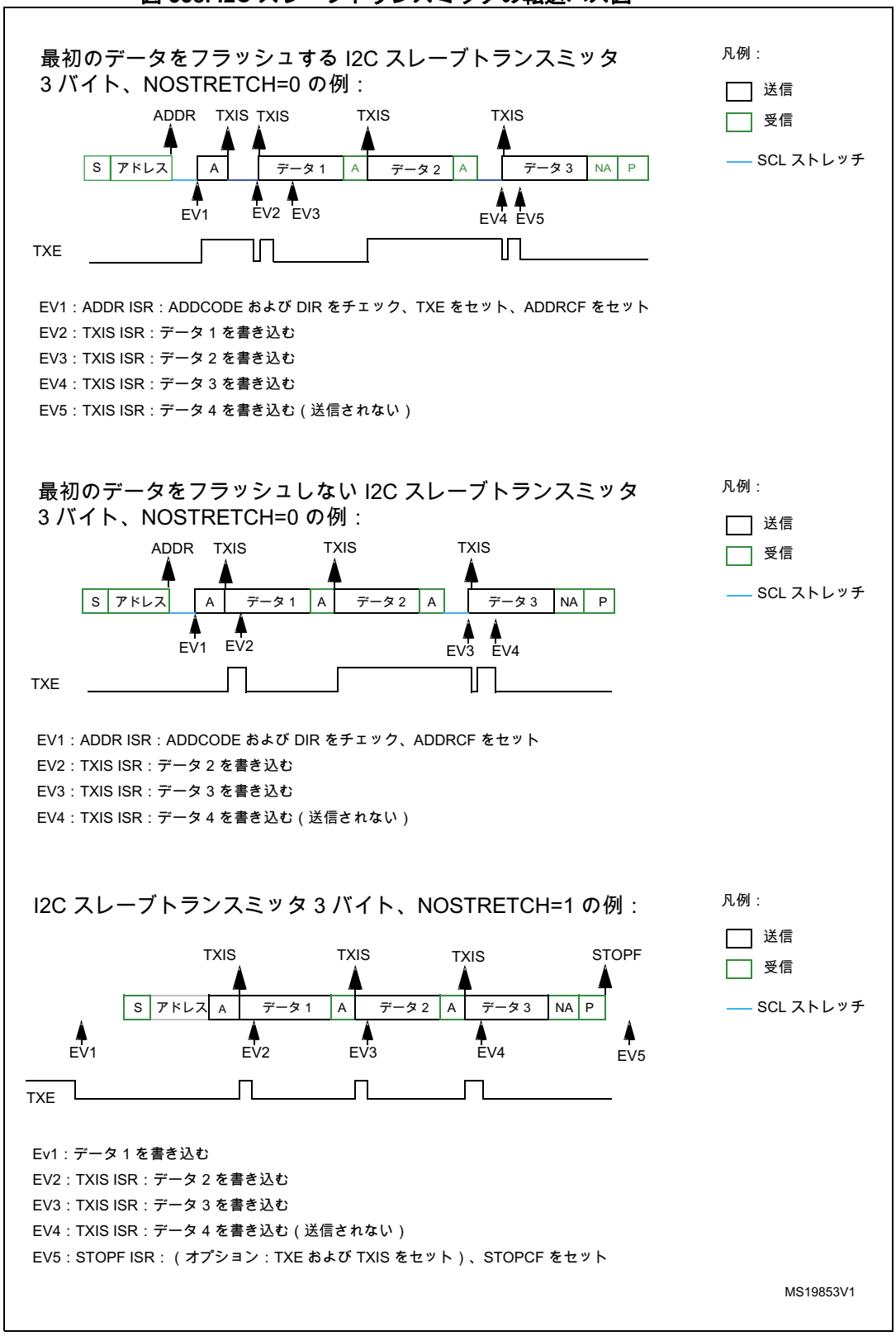


図 537. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=1）



参考資料

図 538. I2C スレーブトランスミッタの転送バス図



スレーブレシーバ

I2C_RXDR がフルのときには、I2C_ISR の RXNE がセットされ、I2C_CR1 の RXIE がセットされている場合は割り込みが生成されます。RXNE は、I2C_RXDR が読み出されたときにクリアされます。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR の STOPF がセットされ、割り込みが生成されます。

図 539. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0）

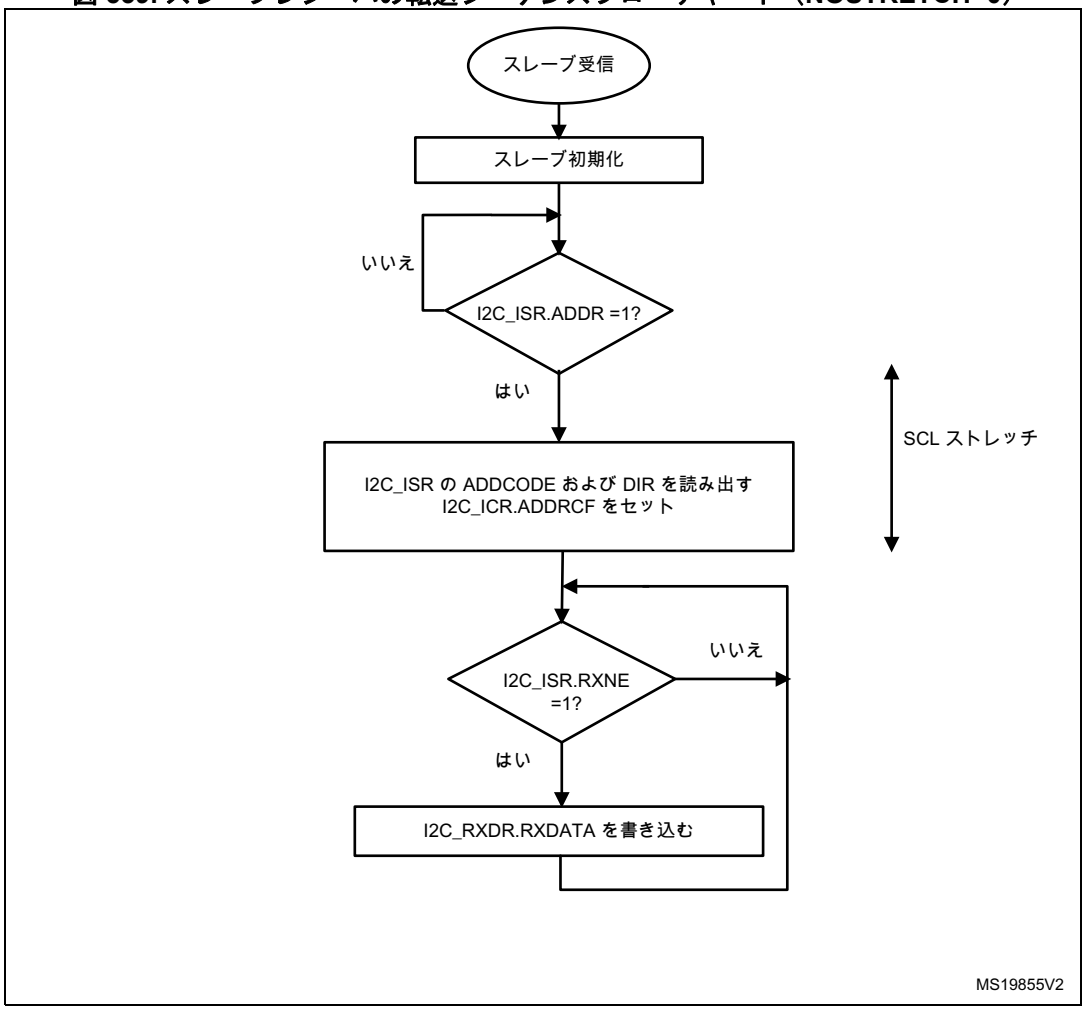


図 540. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1）

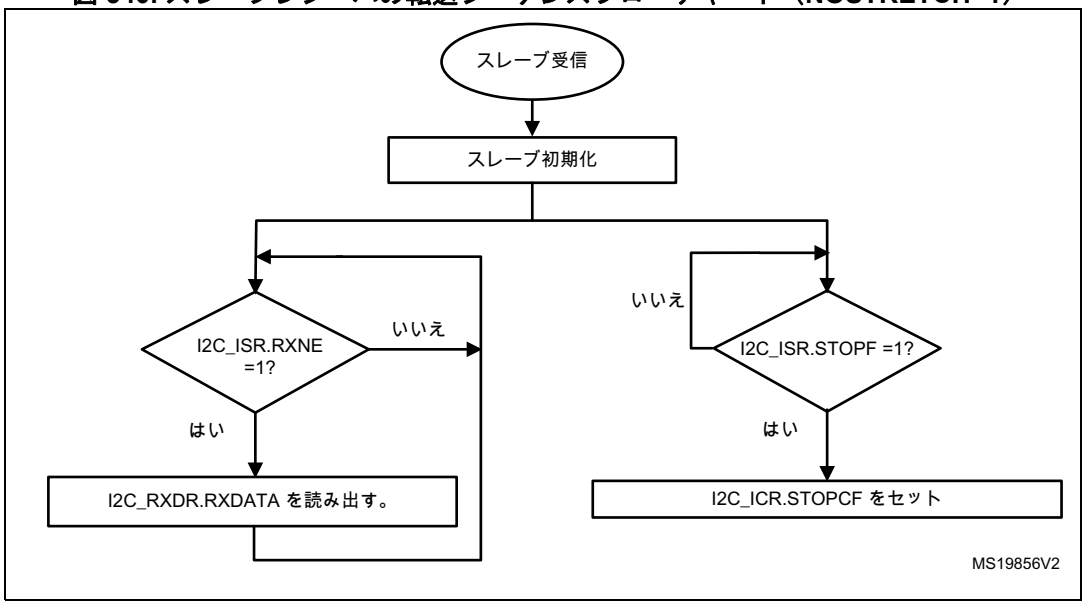
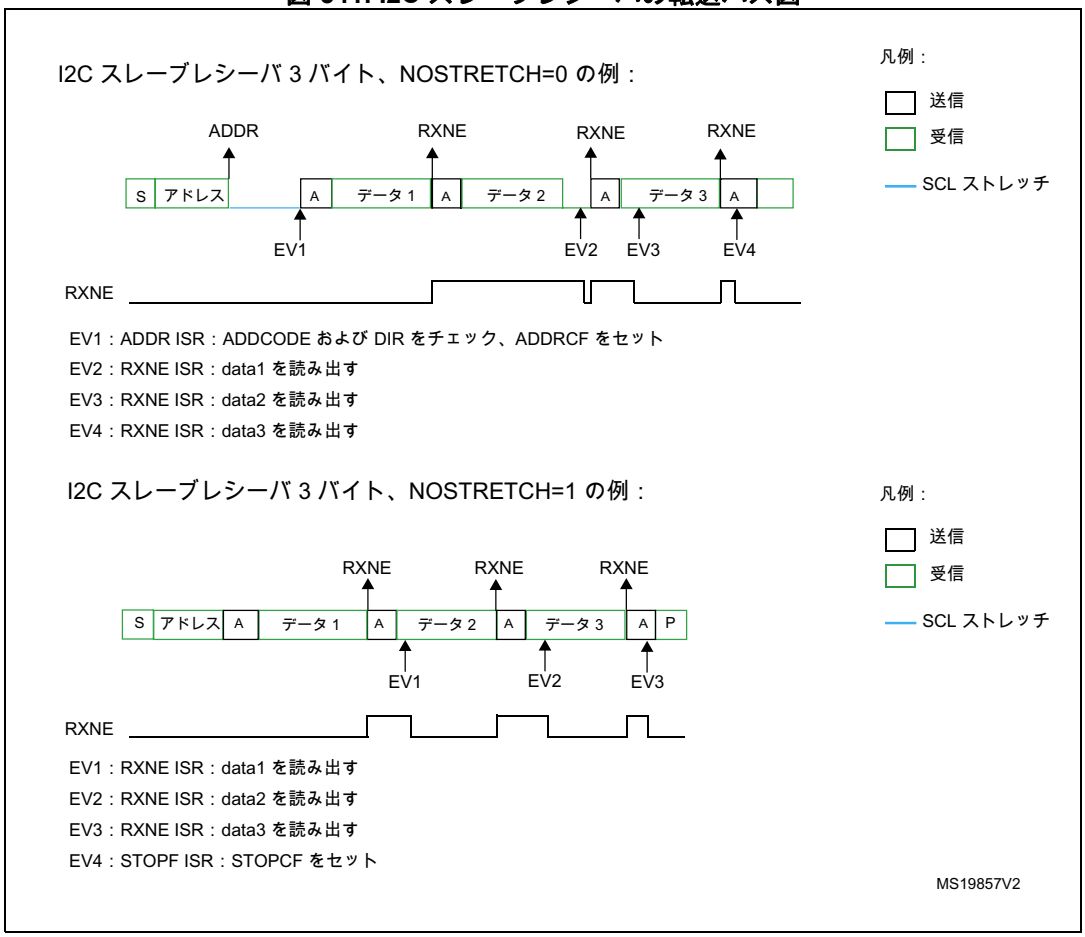


図 541. I2C スレーブレシーバの転送バス図



47.4.8 I2C マスタモード

I2C マスタ初期化

ペリフェラルを有効にする前に、I2C_TIMINGR レジスタの SCLH および SCLL ビットをセットすることによって、I2C マスタクロックを設定する必要があります。

STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルは SCLL カウンタを使用してカウントされ、SCL ローレベル内部検出から開始されます。
- クロックのハイレベルは SCLH カウンタを使用してカウントされ、SCL ハイレベル内部検出から開始されます。

I2C は、SCL 立ち下がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC1} の後に SCL ローレベルを検出します。SCLL カウンタが I2C_TIMINGR レジスタの SCLL[7:0] ビットでプログラムされた値に達すると、I2C は SCL をハイレベルにリリースします。

I2C は、SCL 立ち上がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC2} の後に SCL ハイレベルを検出します。SCLH カウンタが I2C_TIMINGR レジスタの SCLH[7:0] ビットでプログラムされた値に達すると、I2C は SCL をローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{[(SCLH+1) + (SCLL+1)] \times (PRESC+1) \times t_{I2CCLK}\}$

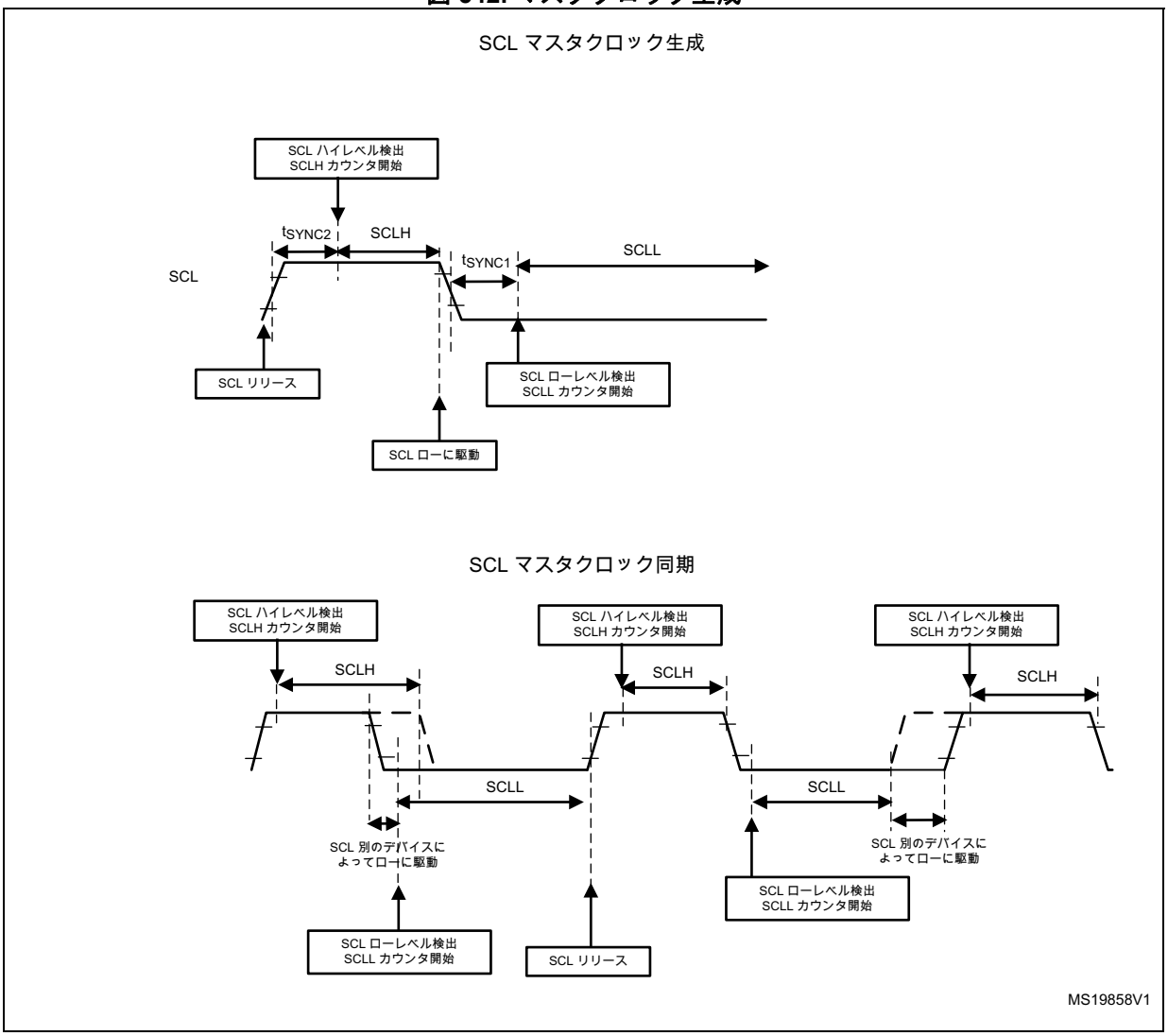
t_{SYNC1} の長さは、次のパラメータに依存します：

- SCL 立ち下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- i2c_ker_ck クロックとの SCL 同期による遅延（2 ～ 3 i2c_ker_ck 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL 立ち上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- i2c_ker_ck クロックとの SCL 同期による遅延（2 ～ 3 i2c_ker_ck 周期）

図 542. マスタクロック生成



注意： ***I^2C または SMBus 準拠のためには、マスタクロックは次のタイミングを満たす必要があります：***

表 363. I²C-SMBUS 仕様のクロックタイミング

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBUS		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
f_{SCL}	SCL クロック周波数	-	100	-	400	-	1000	-	100	kHz
$t_{HD:STA}$	（反復）START コンディションのホールド時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
$t_{SU:STA}$	反復 START コンディションのセットアップ時間	4.7	-	0.6	-	0.26	-	4.7	-	µs
$t_{SU:STO}$	STOP コンディションのセットアップ時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
t_{BUF}	STOP コンディションと START コンディションの間のバスフリー時間	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{LOW}	SCL クロックのロー周期	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{HIGH}	SCL クロックの周期	4.0	-	0.6	-	0.26	-	4.0	50	µs
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	ns
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	ns

注： ***SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングの生成にも使用されます。***

SCLH は、$t_{HD:STA}$ および $t_{SU:STO}$ タイミングの生成にも使用されます。

I2C_TIMINGR 設定と i2c_ker_ck 周波数の例については、*セクション 47.4.9：I2C_TIMINGR レジスタの設定例* を参照してください。

マスタ通信の初期化（アドレスフェーズ）

通信を初期化するためには、I2C_CR2 レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります。

- アドレッシングモード（7 ビットまたは 10 ビット）：ADD10
- 送信されるスレーブアドレス：SADD[9:0]
- 転送方向：RD_WRN
- 10 ビットアドレスが読み出される場合：HEAD10R ビット。HEAD10R を設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ（方向の変更の場合）かを示す必要があります。
- 転送されるバイト数：NBYTES[7:0]。バイト数が 255 バイト以上の場合は、NBYTES[7:0] に 0xFF を書き込む必要があります。

次に、I2C_CR2 レジスタの START ビットをセットする必要があります。START ビットがセットされているとき、上記のすべてのビットを変更することはできません。

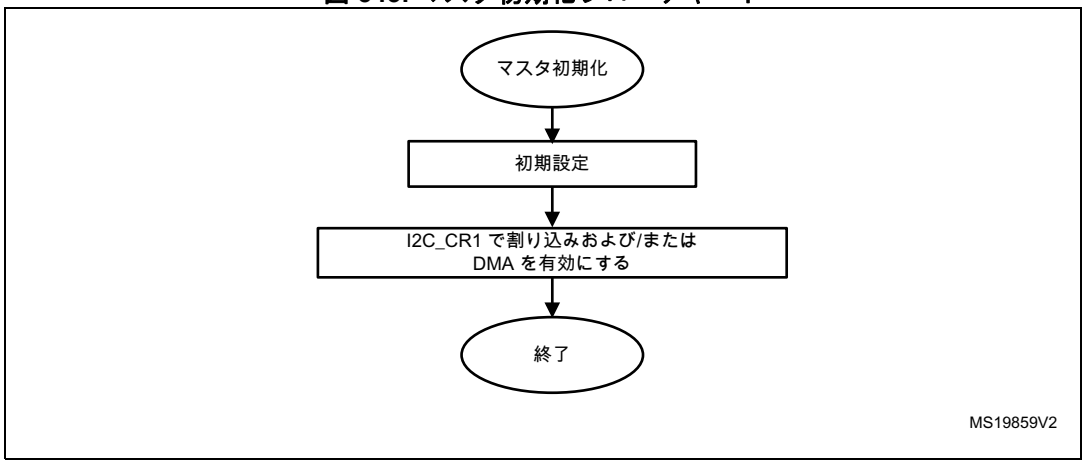
その場合、マスタは、バスがフリーである（BUSY = 0）ことを検出すると、t_{BUF} の遅延後に、自動的に START コンディションとスレーブアドレスを送信します。

アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

注： ***START　ビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。START ビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。***
10 ビットアドレッシングモードでは、スレーブアドレスの最初の 7 ビットがスレーブによって NACK されている場合、マスタは ACK が受信されるまで自動的にスレーブアドレスの送信を再開します。この場合、NACK をスレーブから受信するかどうか、ADDRCF をセットしてスレーブアドレスの送信を停止する必要があります。
START　ビットがセットされているときに、I2C　がスレーブとしてアドレス指定された場合（ADDR=1）、I2C はスレーブモードに切り替わり、ADDRCF ビットがセットされたときに START ビットがクリアされます。

注： ***反復スタートコンディションにも同じ手順が適用されます。この場合、BUSY=1 です。***

図 543. マスタ初期化フローチャート



10 ビットアドレススレーブをアドレス指定するマスタレシーバの初期化

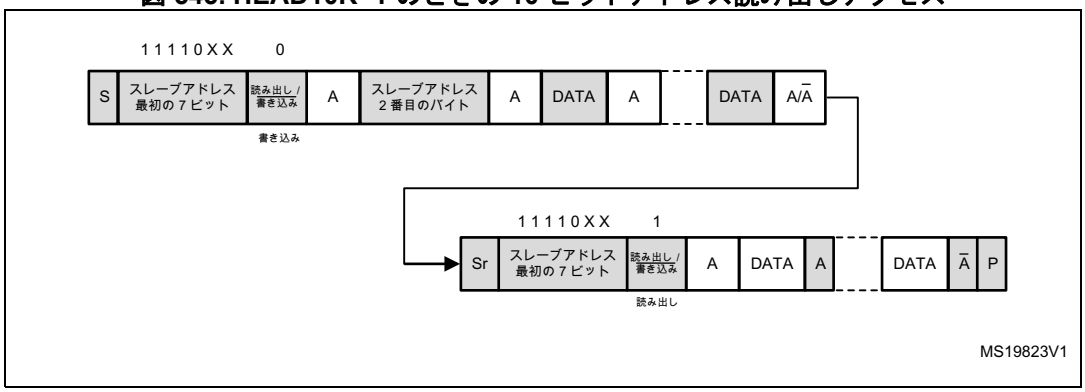
- スレーブアドレスが 10 ビット形式の場合、I2C_CR2 レジスタの HEAD10R ビットをクリアすることによって、完全な読み出しシーケンスを送信することができます。この場合、マスタは、START ビットがセットされた後、次のような完全なシーケンスを自動的に送信します：(Re) START + スレーブアドレス 10 ビットヘッダ書き込み + スレーブアドレスの 2 番目のバイト + REStart + スレーブアドレス 10 ビットヘッダ読み出し。

図 544. HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス



- マスタが 10 ビットアドレススレーブをアドレス指定して、このスレーブアドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その場合、反復開始が、HEAD10R=1 で設定された 10 ビットスレーブアドレスでセットされます。この場合、マスタは次のシーケンスを送信します：ReStart + スレーブアドレス 10 ビットヘッダ読み出し。

図 545. HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス



マスタトランスミッタ

書き込み転送の場合、ACK が受信されたときの 9 番目の SCL パルス後、各バイトの送信後に TXIS フラグがセットされます。

I2C_CR1 レジスタの TXIE ビットがセットされている場合、TXIS イベント時に割り込みが生成されます。このフラグは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中の TXIS イベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

TXIS フラグは、NACK 受信時にはセットされません。

- RELOAD=0 で NBYTES データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、TC フラグがセットされ、ソフトウェアアクションを実行するために SCL ラインがローでストレッチされます：

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。
- NACK が受信された場合：TXIS フラグはセットされず、NACK 受信後、自動的に STOP コンディションが送信されます。I2C_ISR レジスタの NACKF フラグがセットされ、NACKIE ビットがセットされていた場合は割り込みが生成されます。

参考資料

図 546. N≤255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート

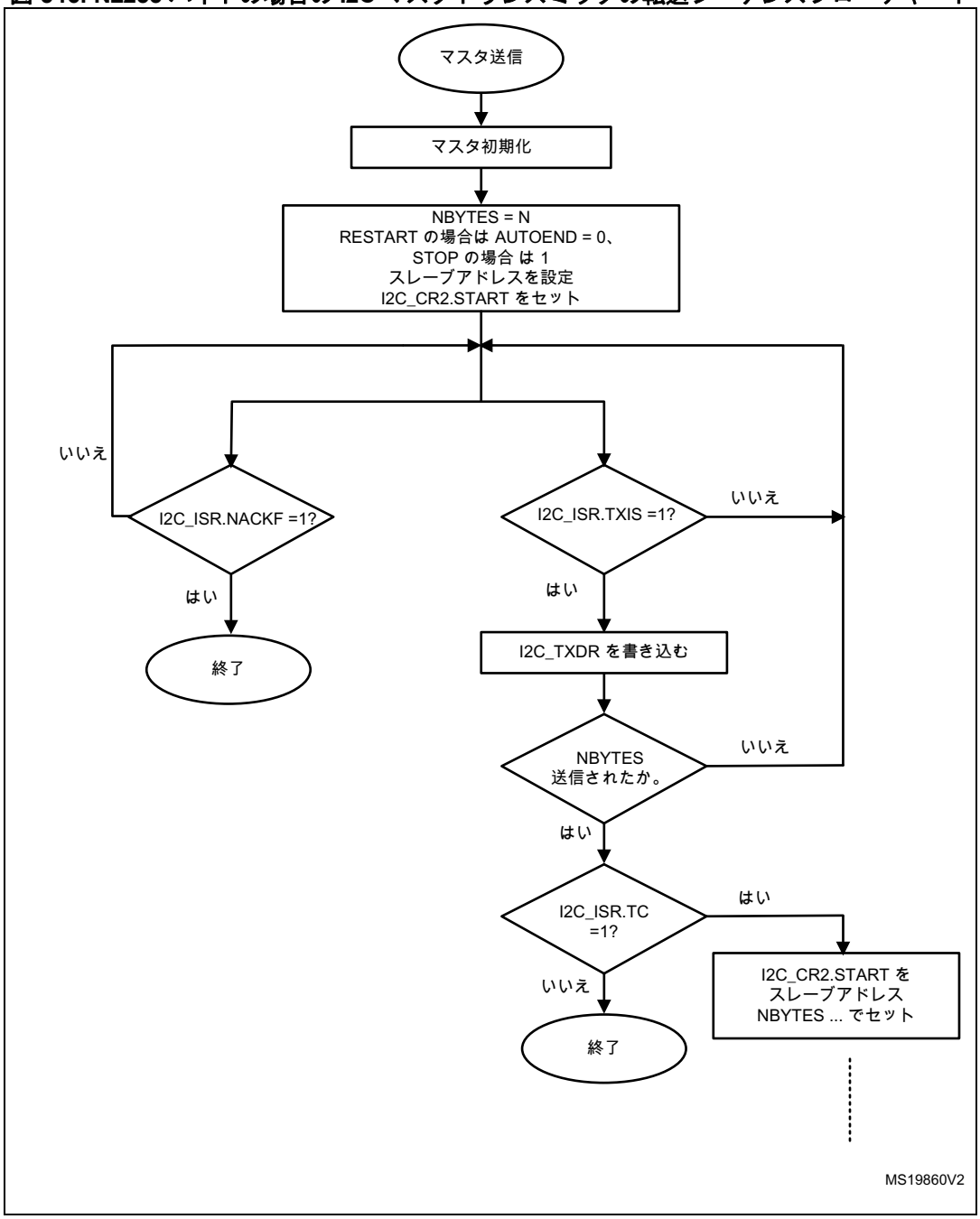
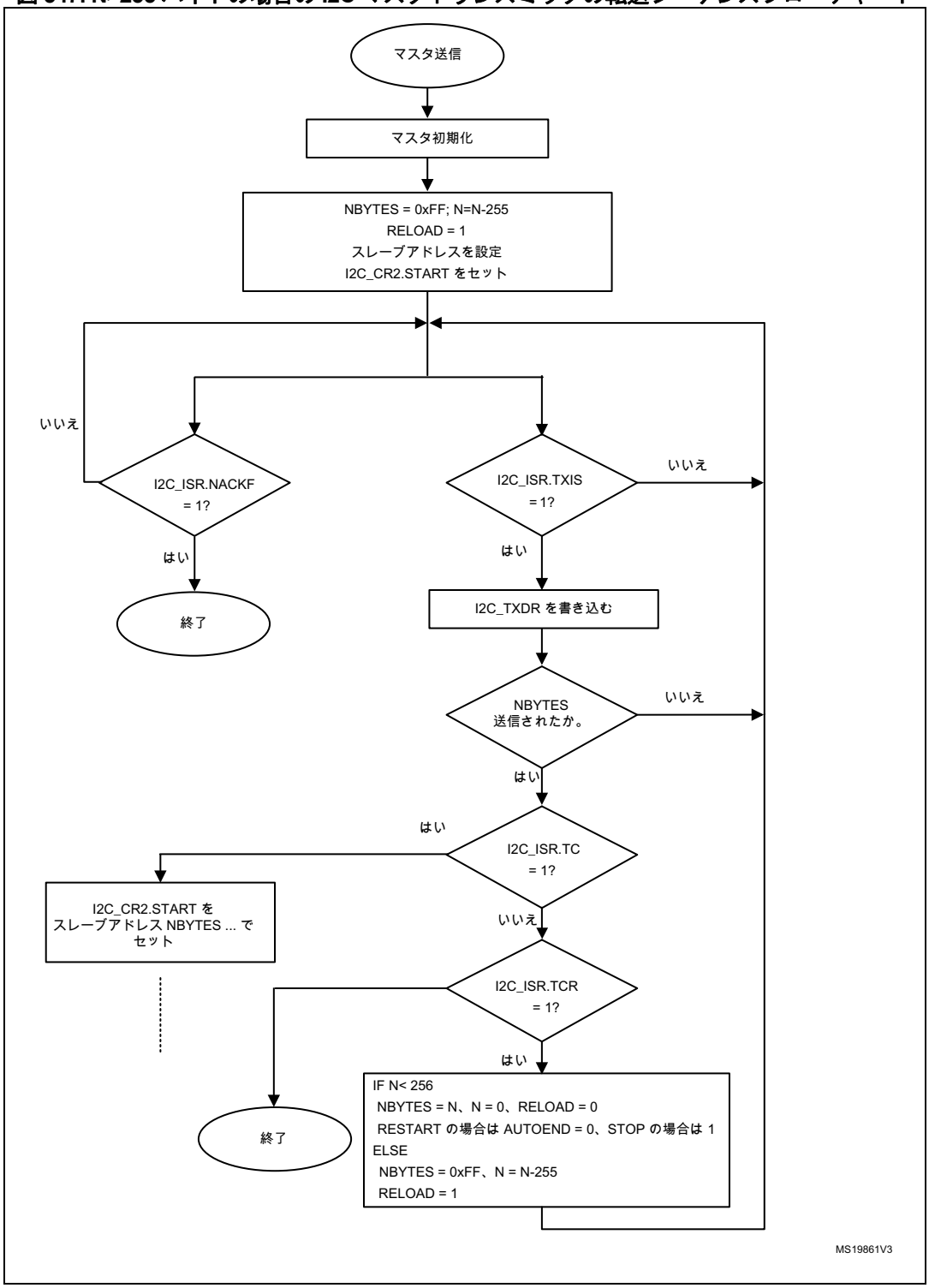
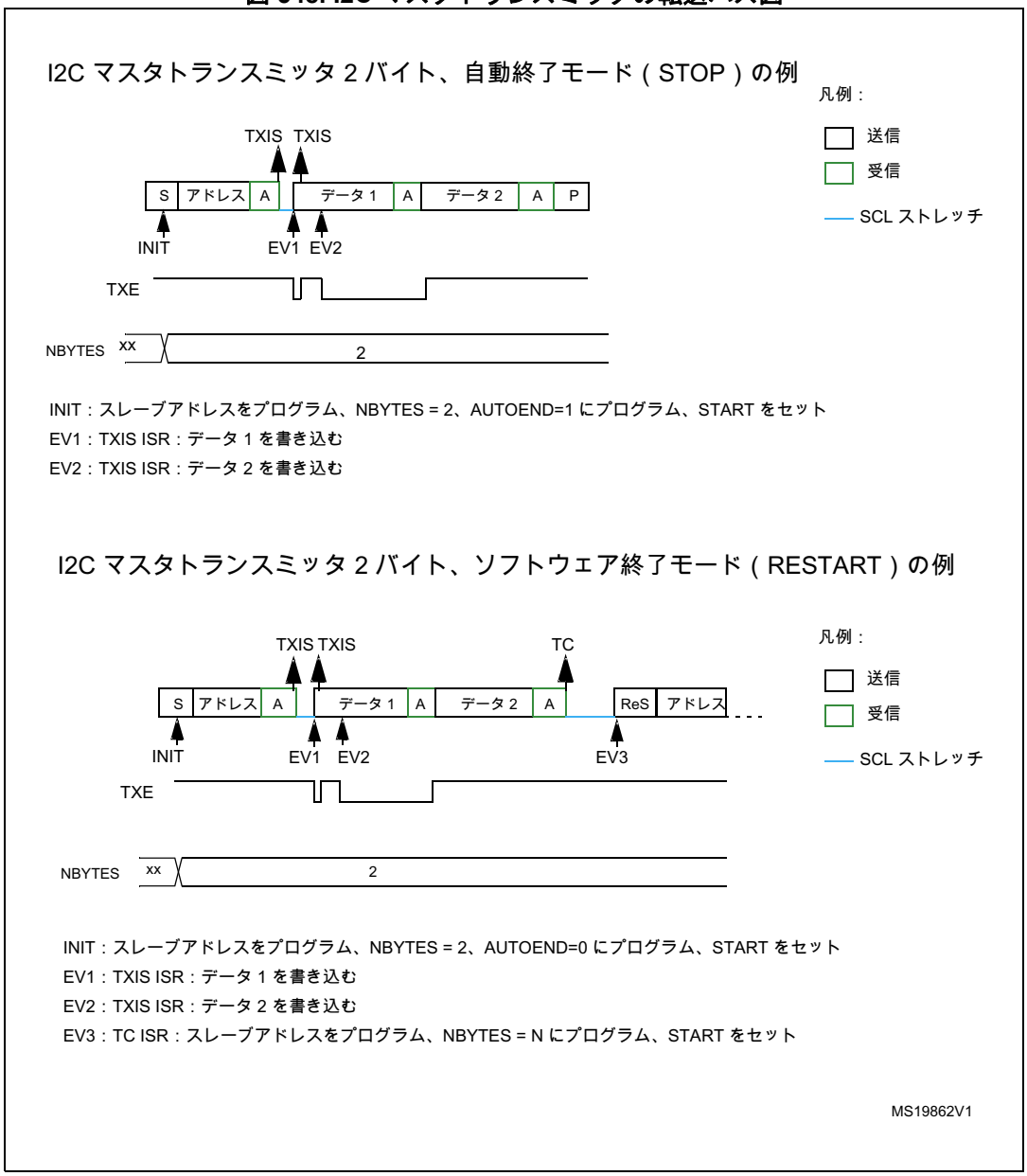


図 547. N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート



参考資料

図 548. I2C マスタトランスミッタの転送バス図



マスタレシーバ

読み出し転送の場合、各バイトの受信後や 8 番目の SCL パルス後に RXNE フラグがセットされます。I2C_CR1 レジスタの RXIE ビットがセットされている場合、RXNE イベント時に割り込みが生成されます。このフラグは、I2C_RXDR が読み出されたときにクリアされます。

受信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

- RELOAD=0 で NBYTES [7:0] データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、最後の受信バイト後に NACK および STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、最後の受信バイト後に NACK が自動的に送信され、TC フラグがセットされ、ソフトウェアアクションを実行できるように、SCL ラインがローでストレッチされます。

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションとスレーブアドレスがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。

図 549. N≤255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート

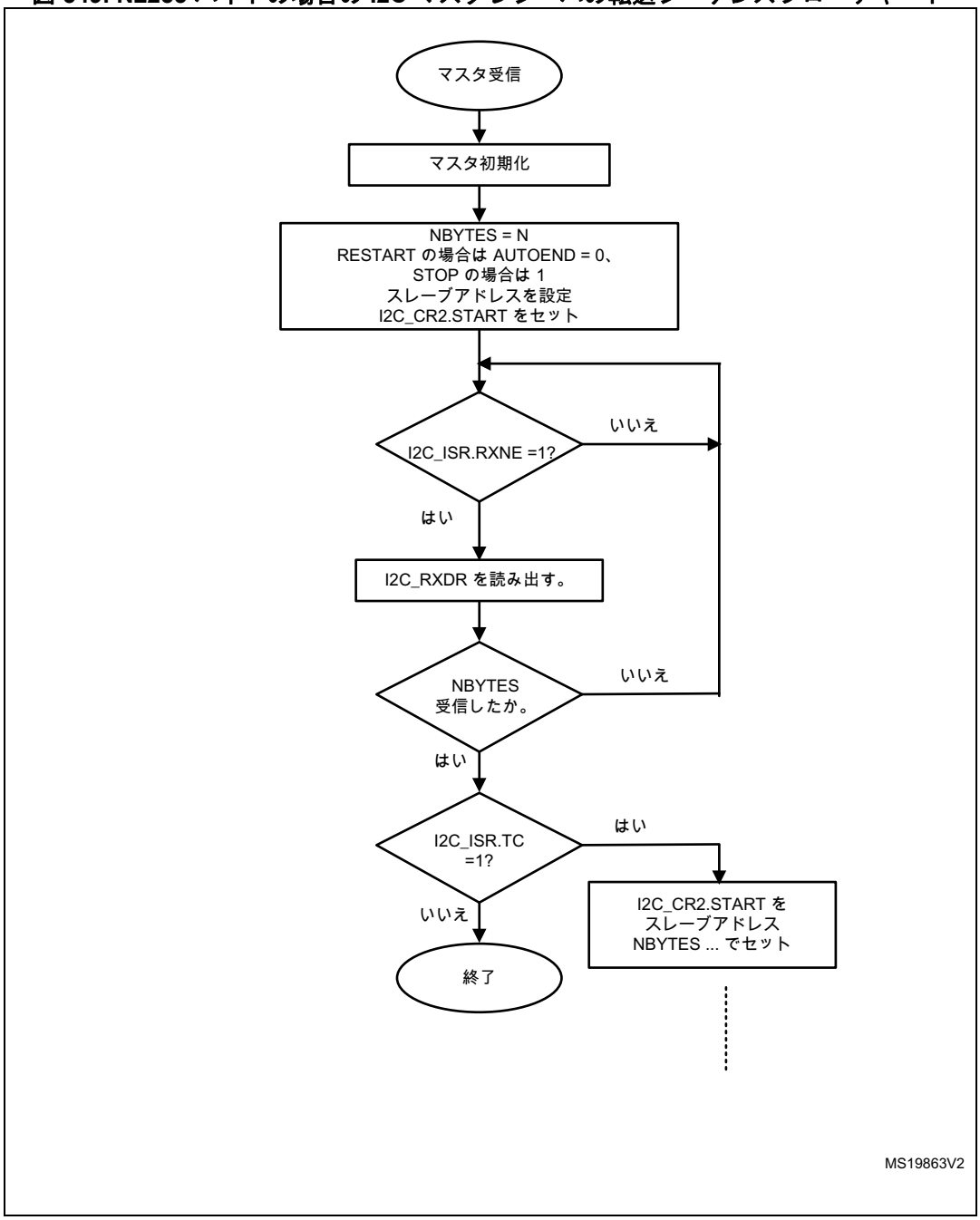
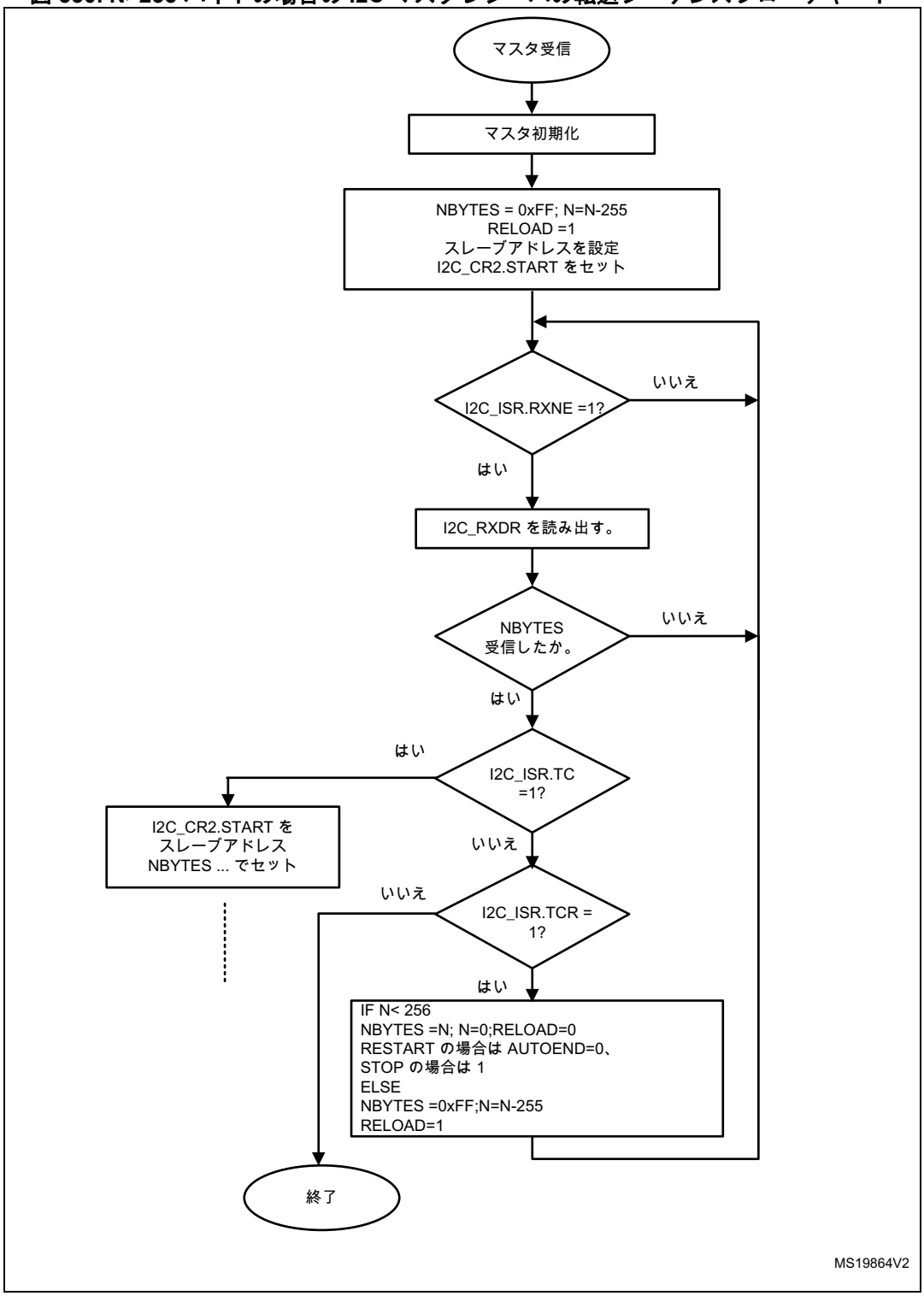
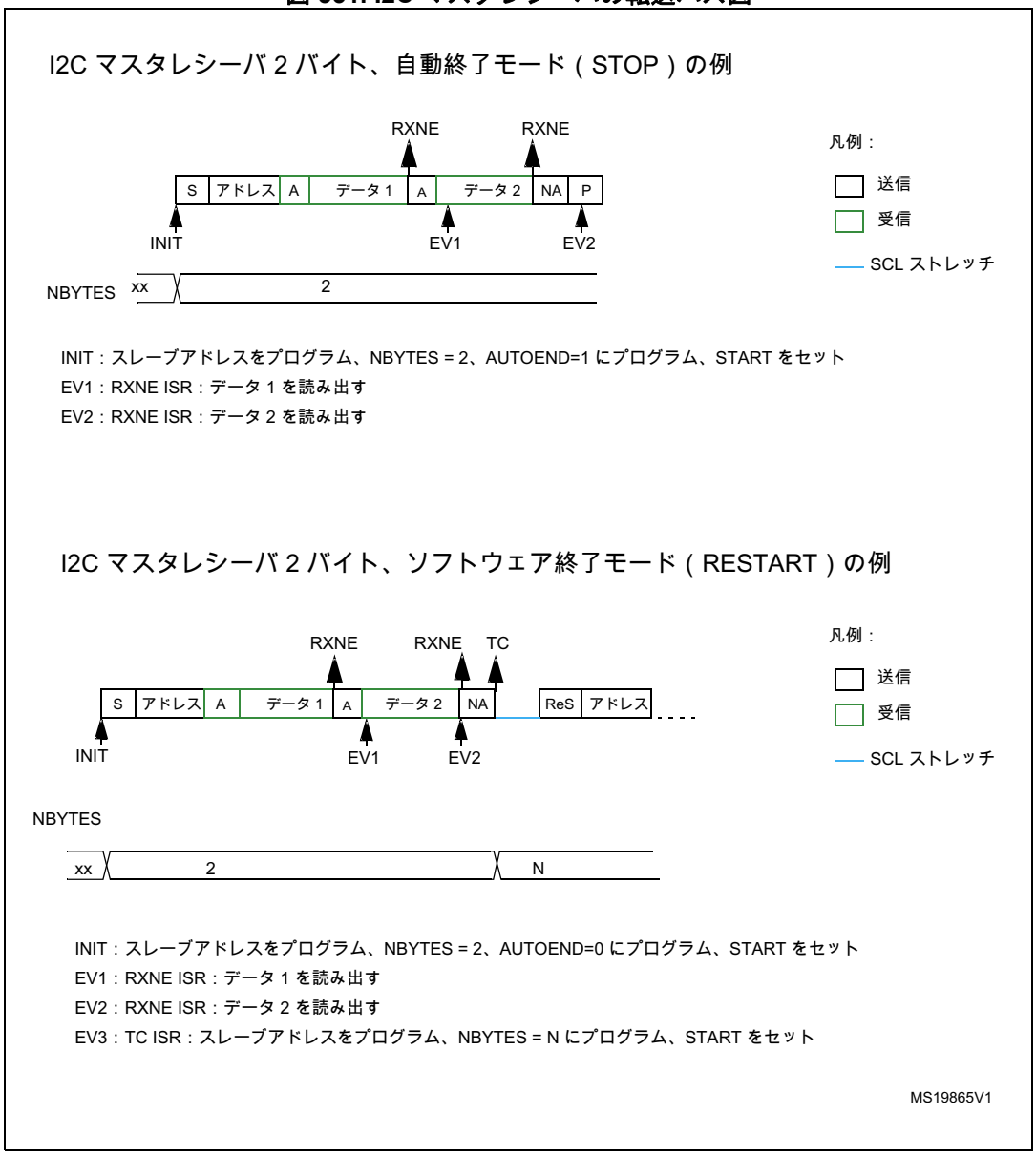


図 550. N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート



参考資料

図 551. I2C マスタレシーバの転送バス図



参考資料

47.4.9 I2C_TIMINGR レジスタの設定例

下の表に、I^2C 仕様に準拠したタイミングを得るための I2C_TIMINGR をプログラムする方法の例を示します。より正確な設定値を得るには、アプリケーションノート*I^2C タイミング設定ツール*（AN4235）および該当するソフトウェア STSW-STM32126 を参照してください。

表 364. f_{I2CCLK} = 8 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	500 kHz
PRESC	1	1	0	0
SCLL	0xC7	0x13	0x9	0x6
t_{SCLL}	200x250 ns = 50 µs	20x250 ns = 5.0 µs	10x125 ns = 1250 ns	7x125 ns = 875 ns
SCLH	0xC3	0xF	0x3	0x3
t_{SCLH}	196x250 ns = 49 µs	16x250 ns = 4.0µs	4x125ns = 500ns	4x125 ns = 500 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～2000 ns(4)
SDADEL	0x2	0x2	0x1	0x0
t_{SDADEL}	2x250 ns = 500 ns	2x250 ns = 500 ns	1x125 ns = 125 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5x250 ns = 1250 ns	5x250 ns = 1250 ns	4x125 ns = 500 ns	2x125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 655 ns

表 365. f_{I2CCLK} = 16 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	3	3	1	0
SCLL	0xC7	0x13	0x9	0x4
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	5 x 62.5 ns = 312.5 ns
SCLH	0xC3	0xF	0x3	0x2
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125ns = 500 ns	3 x 62.5 ns = 187.5 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～1000 ns(4)
SDADEL	0x2	0x2	0x2	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	2 x 125 ns = 250 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x2
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	3 x 62.5 ns = 187.5 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 500 ns

表 366. f_{I2CCLK} = 48 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	0xB	0xB	5	5
SCLL	0xC7	0x13	0x9	0x3
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	4 x 125 ns = 500 ns
SCLH	0xC3	0xF	0x3	0x1
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～875 ns(4)
SDADEL	0x2	0x2	0x3	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	3 x 125 ns = 375 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 250 ns

47.4.10 SMBus 固有の機能

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装* を参照してください。

概要

システム管理バス（SMBus）は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる 2 線インタフェースです。I^2C の動作原理に基づきます。SMBus により、システムおよびパワーマネジメント関連のタスク向けの制御バスが実現できます。

このペリフェラルは、SMBUS 仕様 rev 2.0（http://smbus.org）と互換性があります。

システム管理バス仕様では、3 種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムの CPU にメインインタフェースを提供する特殊なマスタです。ホストは、マスタ／スレーブとすることができ、SMBus ホスト通知プロトコルをサポートする必要があります。システム内では、ただ 1 つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

SMBUS は、I^2C 仕様 rev 2.1 に基づきます。

バスプロトコル

特定のデバイスについて、11 の可能なコマンドプロトコルがあります。デバイスは、11 のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、および Block Write-Block Read Process Call です。これらのプロトコルは、ユーザのソフトウェアによって実装してください。

これらのプロトコルの詳細については、SMBus 仕様バージョン 2.0（http://smbus.org）を参照してください。

アドレス解決プロトコル（ARP）

SMBus スレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子（UDID）を実装する必要があります。128 ビットの番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル（ARP）をサポートします。SMBus デバイスのデフォルトアドレス（0b11001100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。ARP コマンドは、ユーザのソフトウェアによって実装してください。

ARP サポートのために、スレーブモードでアービトレーションも行われます。

SMBus アドレス解決プロトコルの詳細については、SMBus 仕様バージョン 2.0（http://smbus.org）を参照してください。

受信コマンドおよびデータ確認応答制御

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、*1890 ページのスレーブバイト制御モード*を参照してください。

Host Notify プロトコル

このペリフェラルは、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって、Host Notify（ホスト通知）プロトコルをサポートします。この場合、ホストは SMBus ホストアドレス（0b0001 000）を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBus アラート

SMBus ALERT オプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストの SMBALERT# ピンを通じてホストに信号を送信します。ホストは、割り込みを処理し、アラート応答アドレス（0b0001 100）を通じて全 SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブデバイスとして設定されたとき（SMBHEN=0）、I2C_CR1 レジスタの ALERTEN ビットをセットすることによって、SMBA ピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき（SMBHEN=1）、SMBA ピンで立ち下がりエッジが検出され、ALERTEN=1 のとき、I2C_ISR レジスタの ALERT フラグがセットされます。I2C_CR1 レジスタの

ERRIE ビットがセットされている場合は、割り込みが生成されます。ALERTEN=0 のときには、外部 SMBA ピンがローの場合でも、ALERT ラインはハイとみなされます。

SMBus ALERT ピンが不要な場合には、ALERTEN=0 *の場合*、*SMBA ピンを標準 GPIO として使用できます。*

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus 仕様にパケットエラーチェックメカニズムが導入されました。パケットエラーチェックは、各メッセージ転送の終わりにパケットエラーコード（PEC）を付加することによって実装されます。PEC は、すべてのメッセージバイト（アドレスと読み出し／書き込みビットを含む）に対して $C(x) = x_8 + x^2 + x + 1$ CRC-8 多項式を使用して計算されます。

ペリフェラルはハードウェア PEC 計算機が組み込まれ、受信バイトがハードウェアによって計算された PEC に一致しないときには自動的に非確認応答を送信できます。

タイムアウト

このペリフェラルは、SMBus 仕様バージョン 2.0 で定義された 3 つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

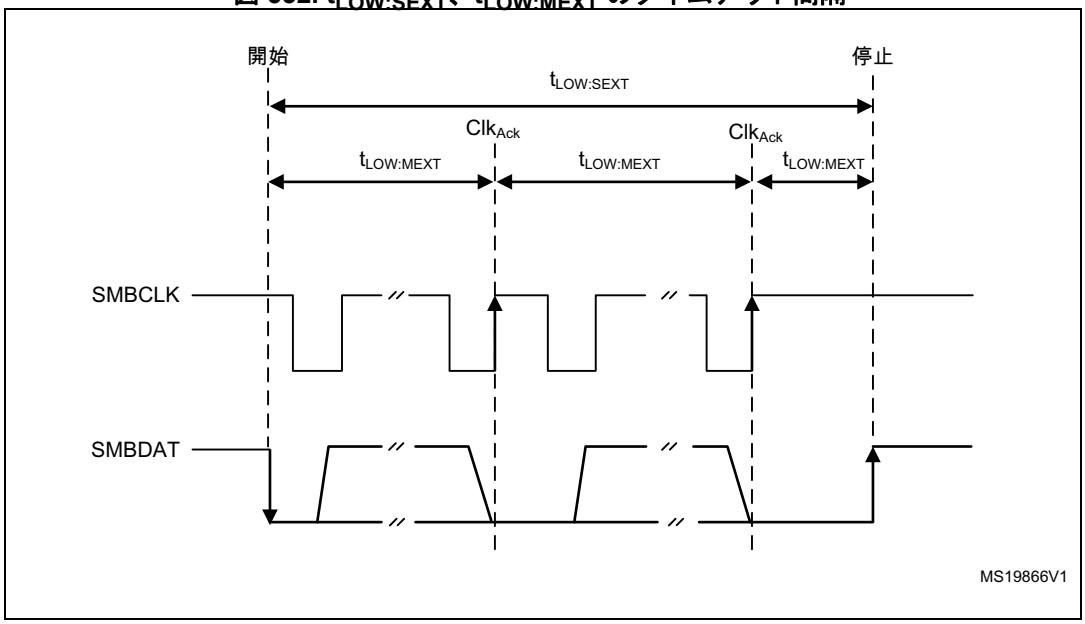
表 367. SMBus タイムアウト仕様

記号	パラメータ	リミット		単位
		最小値	最大値	
$t_{TIMEOUT}$	クロックロータイムアウト検出	25	35	ms
$t_{LOW:SEXT}$(1)	累積クロックロー延長時間（スレーブデバイス）	-	25	ms
$t_{LOW:MEXT}$(2)	累積クロックロー延長時間（マスタデバイス）	-	10	ms

1. $t_{LOW:SEXT}$ は、特定のスレーブデバイスが初めの START から STOP までの 1 つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が $t_{LOW:SEXT}$ より大きくなることがあります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。

2. $t_{LOW:MEXT}$ は、マスタがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、または ACK-to-STOP から定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が $t_{LOW:MEXT}$ より大きくなることがあります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図 552. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が $t_{HIGH,MAX}$ より大きい t_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます（*表 363：I2C-SMBUS 仕様のクロックタイミング* を参照してください）

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLK または SMBDAT ラインで状態遷移を検出しなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

47.4.11 SMBus 初期化

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装* を参照してください。

SMBus 通信を行うためには、I2C 初期化に加えて、他にも特定の初期化を行う必要があります。

受信コマンドおよびデータ確認応答制御（スレーブモード）

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、*1890 ページのスレーブバイト制御モード* を参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定の SMBus アドレスを有効にしてください。詳細については、*1913 ページのバスアイドル検出*を参照してください。

- SMBus デバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。
- SMBus ホストアドレス（0b0001 000）は、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって有効になります。
- アラート応答アドレス（0b0001100）は、I2C_CR1 レジスタの ALERTEN ビットをセットすることで有効になります。

パケットエラーチェック

PEC 計算を有効にするには、I2C_CR1 レジスタの PECEN ビットをセットします。その場合、PEC 転送はハードウェアバイトカウンタ（I2Cx_CR2 レジスタの NBYTES[7:0]）を使用して管理されます。PECEN ビットは、I2C を有効にする前に設定する必要があります。

PEC 転送はハードウェアバイトカウンタによって管理されるので、スレーブモードで SMBus とインタフェースするときには SBC ビットをセットする必要があります。PEC は、PECBYTE ビットがセットされ、RELOAD ビットがクリアされたとき、NBYTES-1 データの転送後に転送されます。RELOAD がセットされた場合、PECBYTE は効果がありません。

注意： ***I2C が有効なときには、PECEN 設定の変更はできません。***

表 368. SMBUS の PEC 設定

モード	SBC ビット	RELOAD ビット	AUTOEND ビット	PECBYTE ビット
マスタ Tx/Rx NBYTES + PEC+ STOP	x	0	1	1
マスタ Tx/Rx NBYTES + PEC + ReSTART	x	0	0	1
スレーブ Tx/Rx と PEC	1	0	x	1

タイムアウト検出

タイムアウト検出は、I2C_TIMEOUTR レジスタの TIMOUTEN および TEXTEN ビットをセットすることによって有効になります。SMBus 仕様バージョン 2.0 で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- $t_{TIMEOUT}$ チェック

  $t_{TIMEOUT}$ チェックを有効にするためには、12 ビットの TIMEOUTA [11:0] ビットを $t_{TIMEOUT}$ パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCL ローレベルタイムアウトを検出するためには、TIDLE ビットを 0 に設定する必要があります。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN をセットすることによって有効になります。

  SCL が (TIMEOUTA+1) x 2048 x t_{I2CCLK} より長い時間、ローに設定された場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  *表 369：さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）*を参照してください。

注意： ***TIMEOUTEN ビットがセットされているときには、TIMEOUTA[11:0] ビットおよび TIDLE ビットの設定変更はできません。***

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ チェック

  ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12 ビットの TIMEOUTB タイマは、スレーブの場合は $t_{LOW:SEXT}$ をチェックするために、マスタの場合は $t_{LOW:MEXT}$ をチェックするために、設定する必要があります。標準では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TEXTEN ビットをセットすることによって有効になります。

  SMBus ペリフェラルが、（TIMEOUTB+1）x 2048 x t_{I2CCLK} より長い時間および *1913 ページのバスアイドル検出* セクションで述べられているタイムアウト間隔で、累積 SCL ストレッチを実行した場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 370：さまざまな i2c_ker_ck 周波数での TIMEOUTB の設定例 を参照してください。

注意： ***TEXTEN ビットがセットされているときには、TIMEOUTB 設定変更はできません。***

バスアイドル検出

t_{IDLE} チェックを有効にするためには、12 ビットの TIMEOUTA [11:0] フィールドを t_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCL および SDA ハイレベルタイムアウトを検出するためには、TIDLE ビットを 1 に設定する必要があります。

その場合、タイマは、I2C_TIMEOUTR レジスタの TIMEOUTEN ビットをセットすることによって有効になります。

SCL および SDA の両方のラインが（TIMEOUTA+1）x 4 x t_{I2CCLK} より長い間ハイのままであった場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

表 371：さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs） を参照してください。

注意： ***TIMEOUTEN がセットされているときに、TIMEOUTA および TIDLE 設定を変更することはできません。***

47.4.12 SMBus：I2C_TIMEOUTR レジスタの設定例

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装* を参照してください。

- $t_{TIMEOUT}$ の最大時間を 25 ms に設定：

表 369. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	$t_{TIMEOUT}$
8 MHz	0x61	0	1	98 x 2048 x 125 ns = 25 ms
16 MHz	0xC3	0	1	196 x 2048 x 62.5 ns = 25 ms
48 MHz	0x249	0	1	586 x 2048 x 20.08 ns = 25 ms

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ の最大時間を 8 ms に設定：

表 370. さまざまな i2c_ker_ck **周波数での TIMEOUTB の設定例**

f_{I2CCLK}	TIMEOUTB[11:0] ビット	TEXTEN ビット	$t_{LOW:EXT}$
8 MHz	0x1F	1	32 x 2048 x 125 ns = 8 ms
16 MHz	0x3F	1	64 x 2048 x 62.5 ns = 8 ms
48 MHz	0xBB	1	188 x 2048 x 20.08 ns = 8 ms

- t_{IDLE} の最大時間を 50 µs に設定

表 371. さまざまな i2c_ker_ck **周波数での TIMEOUTA の設定例（最大値** t_{IDLE} **= 50 µs）**

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	t_{TIDLE}
8 MHz	0x63	1	1	100 x 4 x 125 ns = 50 µs
16 MHz	0xC7	1	1	200 x 4 x 62.5 ns = 50 µs
48 MHz	0x257	1	1	600 x 4 x 20.08 ns = 50 µs

47.4.13 SMBus スレーブモード

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3 : I2C の実装* を参照してください。

I2C スレーブ転送管理（*セクション 47.4.7 : I2C スレーブモード* を参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus スレーブトランスミッタ

IP が SMBus で使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC 送信を可能にするため、1 にプログラムする必要があります。PECBYTE ビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数には PEC 送信が含まれます。その場合、TXIS 割り込みの合計数は NBYTES-1 であり、NBYTES-1 データ転送後にマスタが追加のバイトをリクエストした場合、I2C_PECR レジスタの内容が自動的に送信されます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

参考資料

図 553. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート

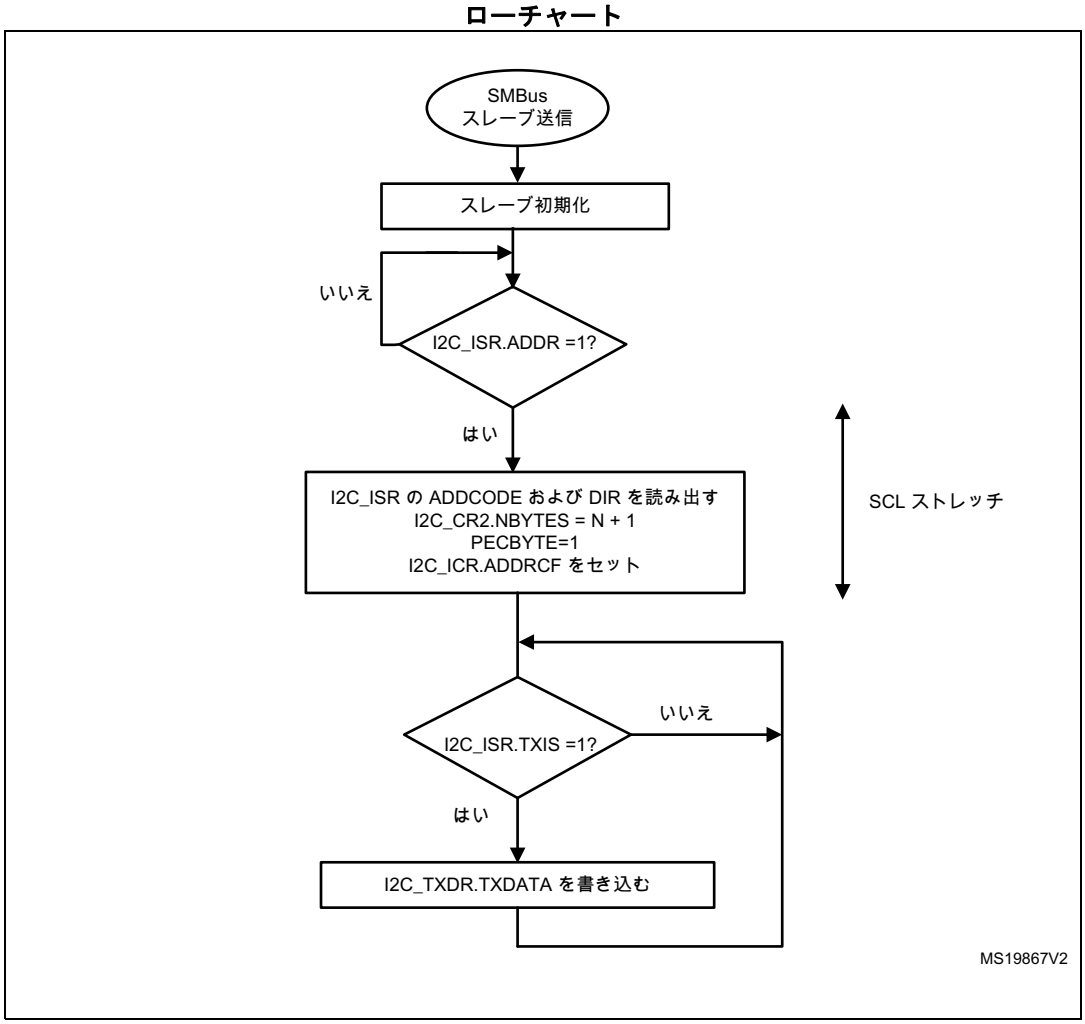
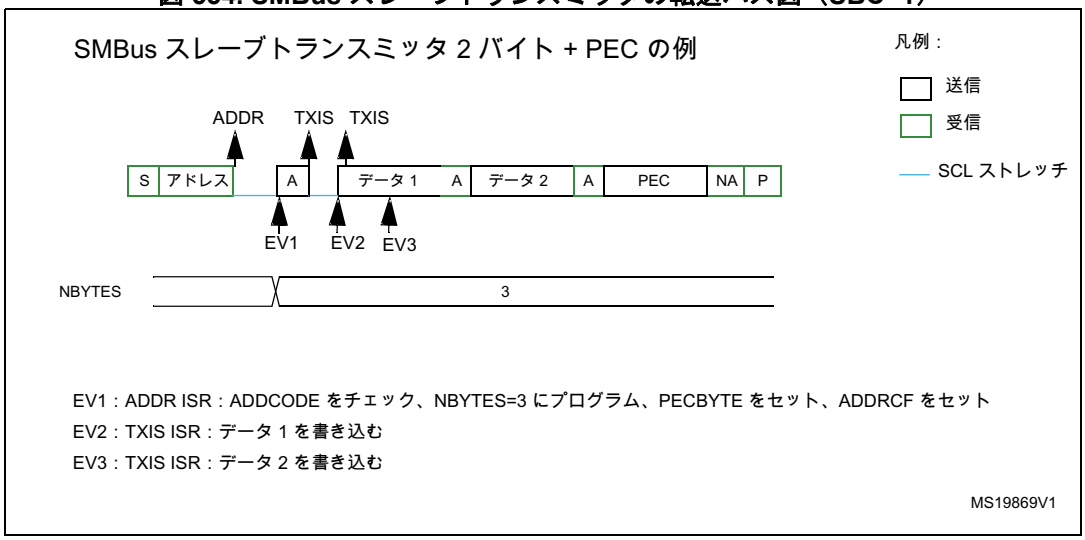


図 554. SMBus スレーブトランスミッタの転送バス図（SBC=1）



SMBus スレーブレシーバ

I2C が SMBus モードで使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC チェックを可能にするため、1 にプログラムする必要があります。各バイトの ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。詳細については、*1890 ページのスレーブバイト制御モード*を参照してください。

PEC バイトをチェックするためには、RELOAD ビットをクリアして、PECBYTE ビットをセットする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが内部 I2C_PECR レジスタの内容と比較されます。ACK ビットの値にかかわらず、比較が一致しなかった場合は NACK が自動的に生成され、比較が一致した場合は ACK が自動的に生成されます。PEC バイトが受信されると、他のデータと同様に I2C_RXDR レジスタにコピーされ、RXNE フラグがセットされます。

PEC 不一致の場合、PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

ACK ソフトウェア制御が不要な場合は、PECBYTE=1 をプログラムし、同じ書き込み操作で NBYTES を連続フローで受信するバイト数にプログラムします。NBYTES-1 が受信された後、次の受信バイトが PEC であるかどうかがチェックされます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

図 555. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート

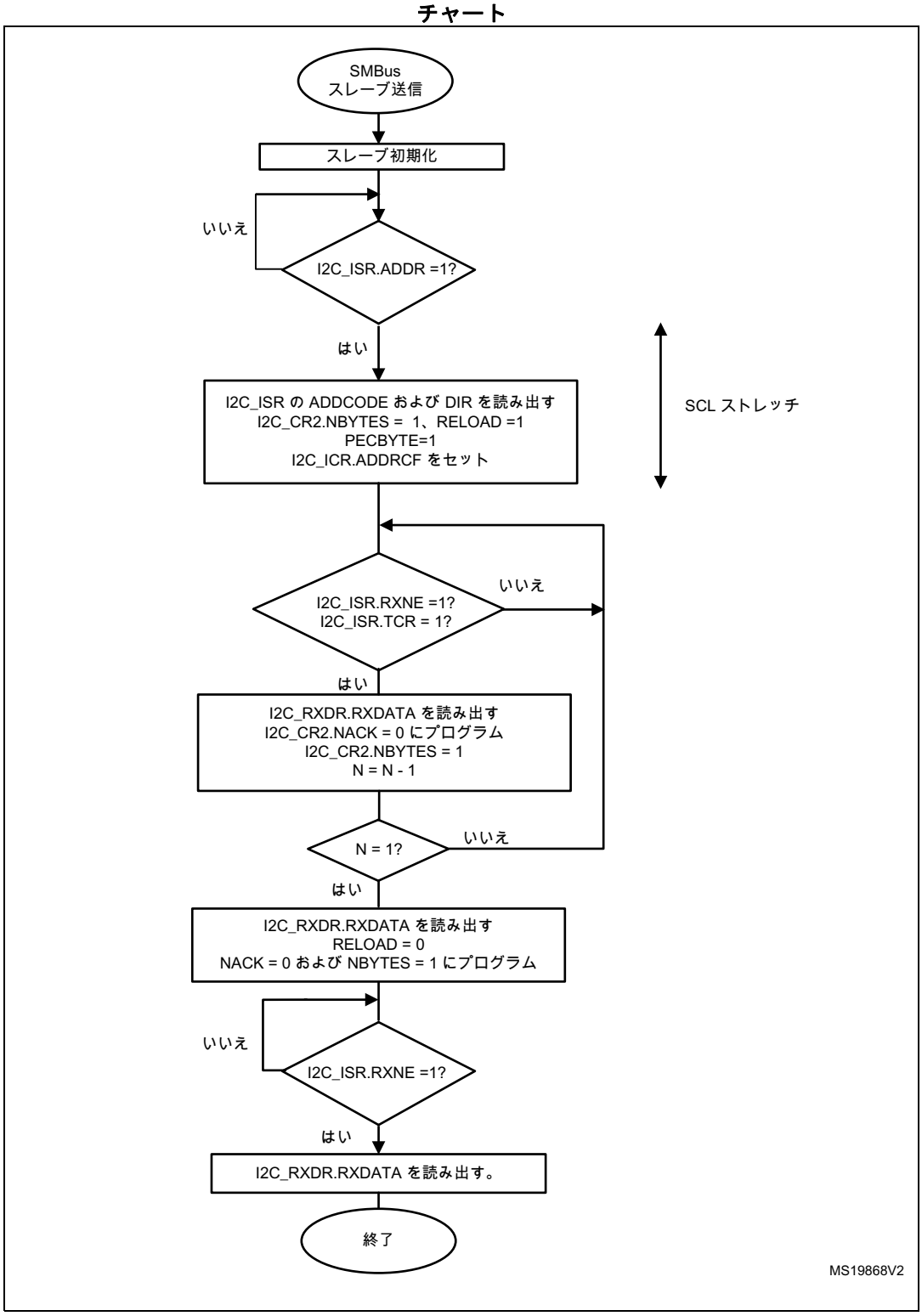
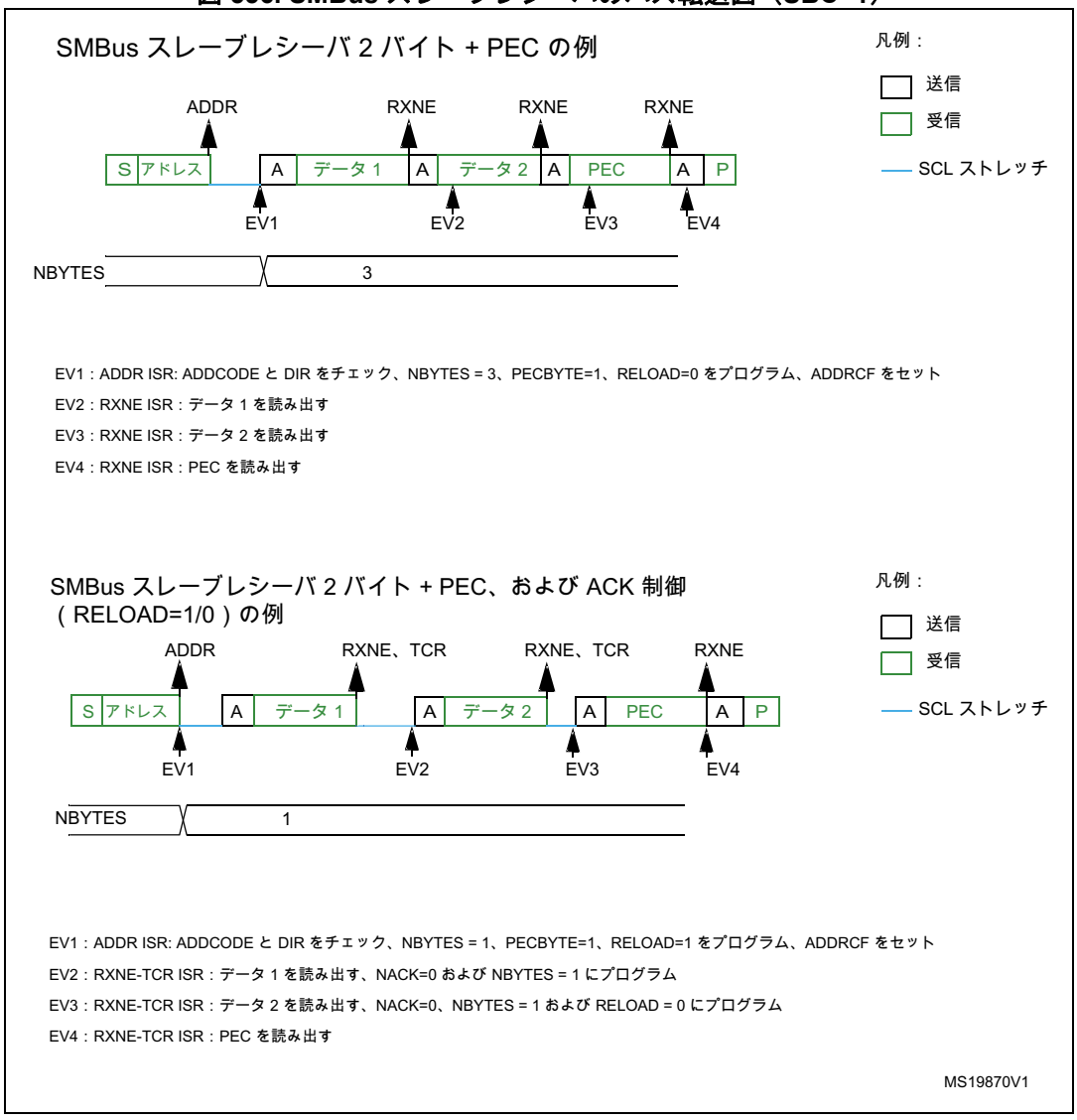


図 556. SMBus スレーブレシーバのバス転送図（SBC=1）



このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装 を参照してください。*

I2C マスタ転送管理（*セクション 47.4.8：I2C マスタモード* を参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus マスタトランスミッタ

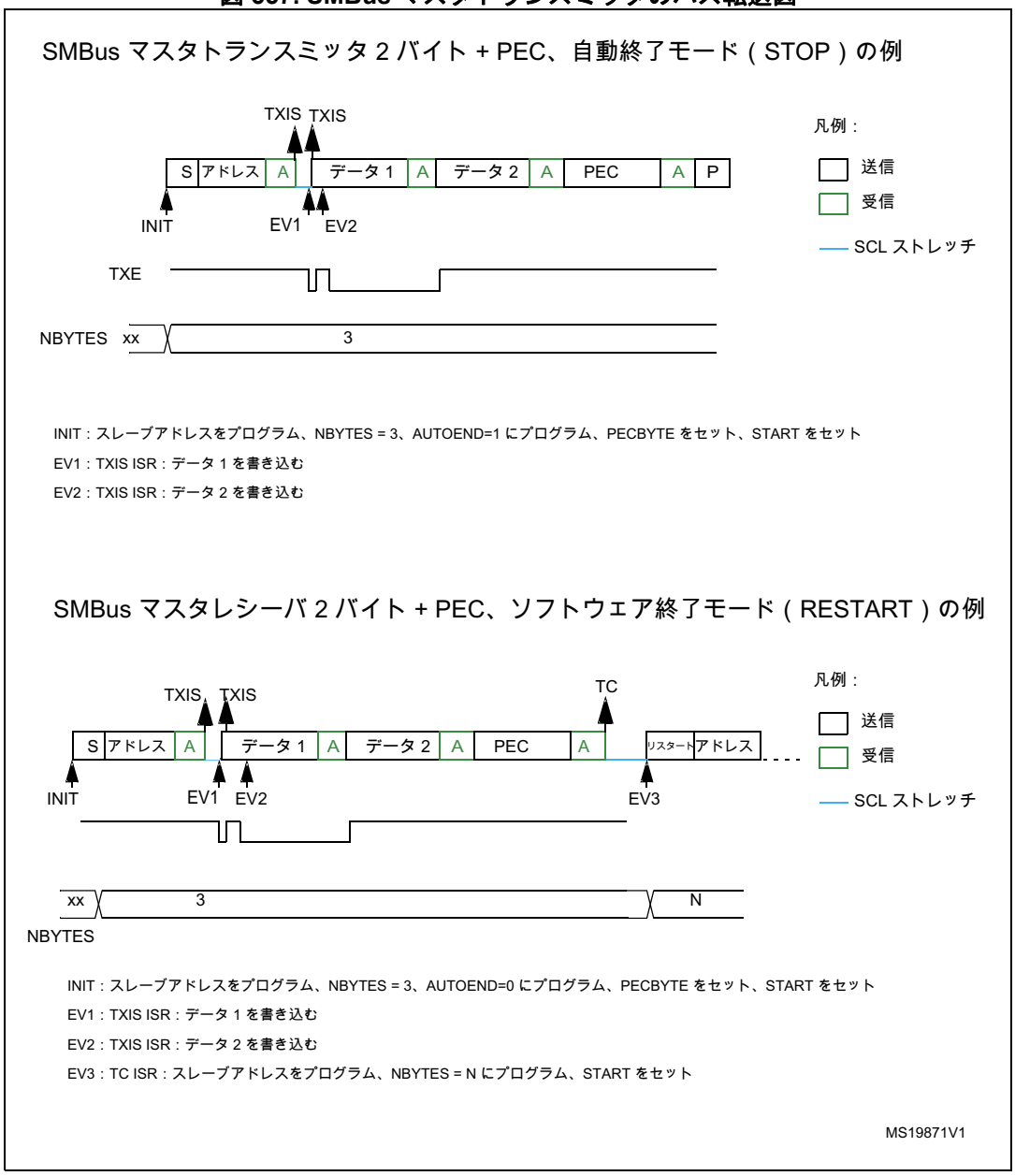
SMBus マスタが PEC を送信したいときには、START ビットをセットする前に、PECBYTE ビットをセットする必要があり、バイト数を NBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS 割り込みの合計数は NBYTES-1 になります。したがって、NBYTES=0x1 のときに PECBYTE ビットがセットされた場合、I2C_PECR レジスタの内容が自動的に送信されます。

SMBus マスタが PEC 後に STOP コンディションを送信したい場合は、自動終了モードを選択してください（AUTOEND=1）。この場合、PEC 送信に続いて、STOP コンディションが自動的に送信されます。

SMBus マスタが PEC 後に RESTART コンディションを送信したい場合は、ソフトウェアモードを選択してください（AUTOEND=0）。この場合、NBYTES-1 が送信されると、PEC 送信後に I2C_PECR レジスタの内容が送信され、TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションを TC 割り込みサブルーチンでプログラムする必要があります。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

図 557. SMBus マスタトランスミッタのバス転送図



SMBus マスタレシーバ

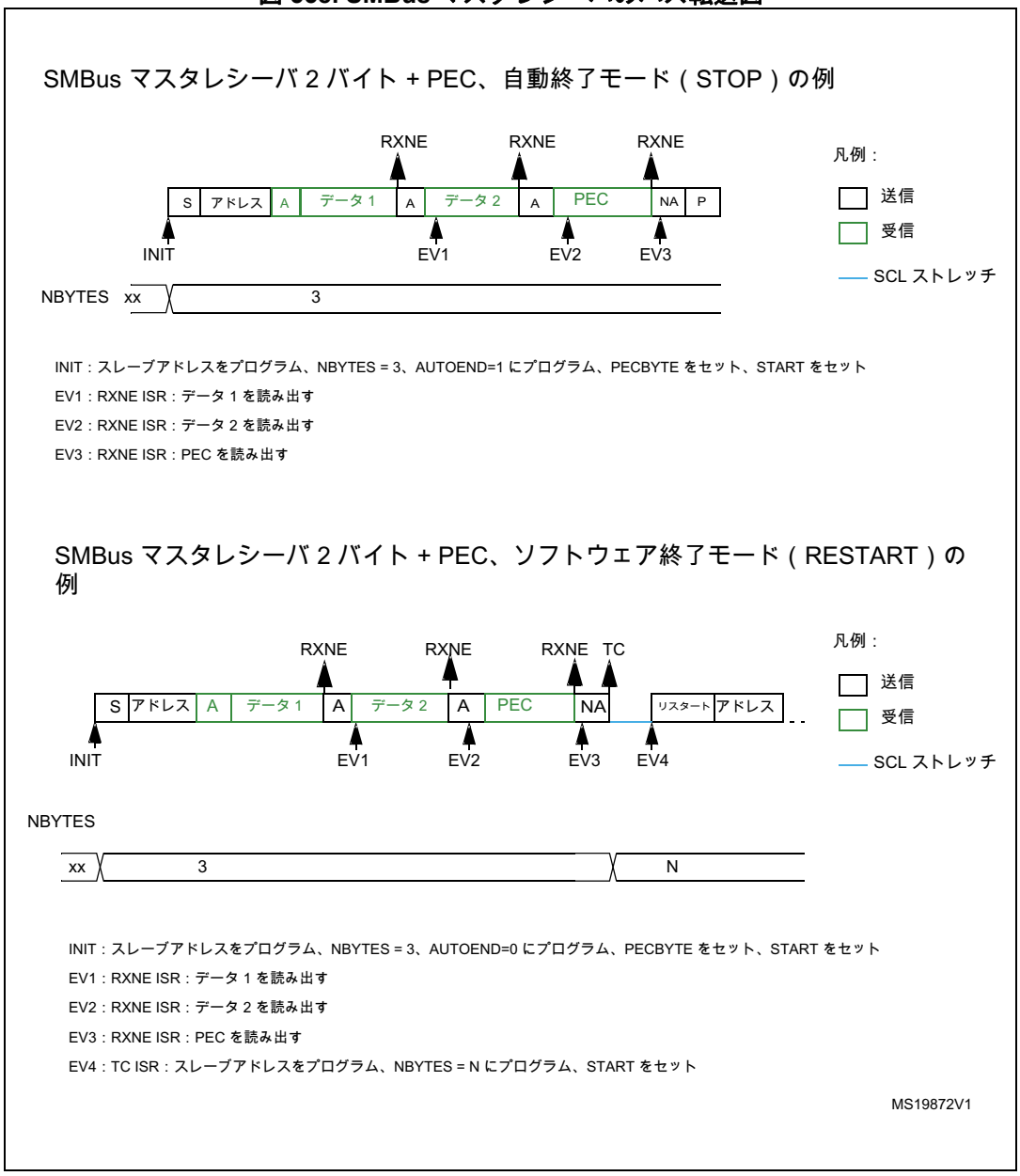
SMBus マスタが転送終了時に PEC を受信してから STOP を受信したいときには、自動終了モードを選択できます（AUTOEND=1）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイトに対して NACK 応答が与えられた後、STOP コンディションが送信されます。

SMBus マスタが転送終了時に PEC バイトを受信してから RESTART を受信したいときには、ソフトウェアモードを選択する必要があります（AUTOEND=0）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイト受信後に TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションは、TC 割り込みサブルーチンでプログラムできます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

参考資料

図 558. SMBus マスタレシーバのバス転送図



47.4.14 アドレス一致時に STOP モードからウェイクアップ

このセクションは、STOP モードからのウェイクアップ機能がサポートされるときにのみ適用されます。*セクション 47.3 : I2C の実装 を参照してください。*

I2C は、アドレス指定されたとき、MCU を STOP モードからウェイクアップできます（APB クロックはオフ）。すべてのアドレッシングモードがサポートされます。

STOP モードからのウェイクアップを有効にするには、I2C_CR1 レジスタの WUPEN ビットをセットします。STOP モードからのウェイクアップを可能にするには、HSI または CSI オシレータを i2c_ker_ck のクロックソースとして選択する必要があります。

STOP モード中、内部オシレータはオフです。START が検出されると、I2C インタフェースは内部オシレータをオンに切り替えて、内部オシレータがウェイクアップするまで SCL ローをストレッチします。

内部オシレータは、その後アドレス受信に使用されます。

アドレス一致の場合、I2C は、MCU のウェイクアップ時間の間、SCLローをストレッチします。ストレッチは、ADDR フラグがソフトウェアによってクリアされたときにリリースされ、転送は通常通りに続行されます。

アドレスが一致しなかった場合、内部オシレータは再びオフになり、MCU はウェイクアップしません。

注 : ***I2C クロックがシステムクロックの場合、または WUPEN = 0 の場合、START 受信後も内部オシレータはオンになりません。***

ADDR 割り込みによってのみ、MCU をウェイクアップできます。したがって、I2C がマスタとして、または ADDR フラグのセット後にアドレス指定されたスレーブとして転送を行っているときには、STOP モードに入らないでください。これを管理するには、ADDR 割り込みルーチンで SLEEPDEEP ビットをクリアして、STOPF フラグのセット後にのみ再びオンにセットします。

注意 : ***デジタルフィルタは、STOP モードからのウェイクアップ機能と互換性がありません。DNF ビットが 0 でない場合、WUPEN ビットをセットしても効果はありません。***

注意 : ***この機能は、I2C クロックソースが HSI または CSI オシレータのときのみ使用できます。***

注意 : ***STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロックストレッチを有効にする必要があります（NOSTRETCH=0）。***

注意 : ***STOP モードからのウェイクアップが無効な場合（WUPEN=0）、STOP モードに入る前に、I2C ペリフェラルを無効にする必要があります（PE=0）。***

47.4.15 エラー条件

以下は、通信エラーを引き起こす可能性のあるエラー条件です。

バスエラー（BERR）

バスエラーは、START または STOP コンディションが検出され、複数の 9 SCL クロックパルス後になかったときに検出されます。START または STOP コンディションは、SCL がハイとのときに SDA エッジが発生した場合に検出されます。

バスエラーフラグは、I2C がマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ（すなわち、スレーブモードのアドレスフェーズでないとき）、セットされます。

スレーブモードで START または RESTART の誤配置が検出された場合、I2C は、正しい START コンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2C_ISR レジスタの BERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アービトレーション喪失（ARLO）

アービトレーション喪失は、SDA ラインでハイレベルが送信されたが、SCL 立ち上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDA および SCL ラインはリリースされ、START 制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCL および SDA ラインがリリースされます。

アービトレーション喪失が検出されると、I2C_ISR レジスタの ARLO フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

オーバーラン／アンダーランエラー（OVR）

オーバーランまたはアンダーラインエラーは、スレーブモードで NOSTRETCH=1 のとき、および次のときに検出されます：

- 受信時、新しいバイトが受信され、RXDR レジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答として NACK が自動的に送信されます。
- 送信時：
  - STOPF=1 のときには、最初のデータバイトが送信されなければなりません。TXE=0、0xFF の場合、I2C_TXDR レジスタの内容が送信され、そうでない場合は送信されません。
  - 新しいバイトが送信されるべきときに、I2C_TXDR レジスタがまだ書き込まれていなかった場合、0xFF が送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

パケットエラーチェックエラー（PECERR）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装 を参照してください。*

PEC エラーは、受信した PEC バイトが I2C_PECR レジスタの内容と一致しなかったときに検出されます。正しくない PEC の受信後、NACK が自動的に送信されます。

PEC エラーが検出されると、I2C_ISR レジスタの PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

タイムアウトエラー（TIMEOUT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装 を参照してください。*

タイムアウトエラーは、次のような条件で発生します：

- TIDLE=0 であり、SCL が TIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これは SMBus タイムアウトの検出に使用されます。
- TIDLE=1 であり、SDA および SCL が TIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:MEXT}$ パラメータ）。
- スレーブ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:SEXT}$ パラメータ）。

マスタモードでタイムアウト違反が検出されると、STOP コンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDA および SCL ラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2C_ISR レジスタの TIMEOUT フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アラート（ALERT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 47.3：I2C の実装 を参照してください。*

ALERT フラグは、I2C インタフェースがホストとして設定され（SMBHEN=1）、アラートピン検出が有効であり（ALERTEN=1）、SMBA ピンで立ち下がりエッジが検出されたときにセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割り込みが生成されます。

47.4.16 DMA リクエスト

DMA を使用した送信

送信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの TXDMAEN ビットをセットします。TXIS ビットがセットされるたびに、データは、DMA ペリフェラル（*セクション 15：581 ページのダイレクトメモリアクセスコントローラ（DMA1、DMA2）*を参照）を使用して設定された SRAM 領域から I2C_TXDR レジスタにロードされます。

データのみが DMA で転送されます。

- マスタモード：初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます（送信されたスレーブアドレスを DMA で転送することはできません）。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。*1901 ページのマスタトランスミッタ*を参照してください。
- スレーブモードでは：
  - NOSTRETCH=0 では、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
  - NOSTRETCH=1 では、アドレス一致イベントの前に DMA を初期化する必要があります。
- SMBus をサポートする場合：PEC 転送は NBYTES カウンタによって管理されます。*1916 ページのSMBus スレーブトランスミッタ* および *1920 ページのSMBus マスタトランスミッタ* を参照してください。

注： ***DMA が送信に使用される場合、TXIE ビットが有効である必要はありません。***

DMA を使用した受信

受信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの RXDMAEN ビットをセットします。RXNE ビットがセットされているときには、データは、I2C_RXDR レジスタから DMA ペリフェラル（*セクション 15：581 ページのダイレクトメモリアクセスコントローラ（DMA1、DMA2）*を参照）を使用して設定された SRAM 領域にロードされます。データのみ（PEC を含む）が DMA で転送されます。

- マスタモード、初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。

- NOSTRETCH=0 のスレーブモードでは、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
- SMBus がサポートされる場合（*セクション 47.3：I2C の実装* を参照）：PEC 転送は NBYTES カウンタによって管理されます。*1918 ページのSMBus スレーブレシーバ*および*1922 ページのSMBus マスタレシーバ*を参照してください。

注： ***DMA が受信に使用される場合、RXIE ビットが有効である必要はありません。***

47.4.17 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールの DBG_I2Cx_ 設定ビットに応じて、SMBus タイムアウトは、通常の動作を続行するか、あるいは停止します。

47.5 I2C 低電力モード

表 372. 低電力モード

モード	説明
SLEEP	影響なし。 I2C 割り込みによって、デバイスは SLEEP モードから復帰します。
STOP	I2C レジスタの内容は保たれます。
STANDBY	I2C ペリフェラルはパワーダウンされ、STANDBY 終了後に再初期化する必要があります。

47.6 I2C 割り込み

I2C では、下表に記載されたイベントに従って 2 つの割り込み（i2c_event_it と i2c_error_it）およびウェイクアップイベント信号（i2c_wkup）を生成することができます。

表 373. I2C 割り込みリクエスト

割り込みイベント	イベントフラグ	イベントフラグ／割り込みのクリア方法	割り込み有効制御ビット	有効になる割り込み／ウェイクアップ		
				i2c_event_it	i2c_error_it	i2c_wkup
受信バッファノットエンプティ	RXNE	I2C_RXDR レジスタを読み出す	RXIE	はい	いいえ	いいえ
送信バッファ割り込みステータス	TXIS	I2C_TXDR レジスタに書き込む	TXIE			
STOP 検出割り込みフラグ	STOPF	STOPCF=1 を書き込む	STOPIE			
転送完了再ロード	TCR	I2C_CR2 の NBYTES[7:0] に 0 以外 を書き込む	TCIE			
転送完了	TC	START=1 または STOP=1 を書き込む				
アドレス一致	ADDR	ADDRCF=1 を書き込む	ADDRIE			はい(1)
NACK 受信	NACKF	NACKCF=1 を書き込む	NACKIE			いいえ
バスエラー	BERR	BERRCF=1 を書き込む	ERRIE	いいえ	はい	
アービトレーション喪失	ARLO	ARLOCF=1 を書き込む				
オーバーラン／アンダーラン	OVR	OVRCF=1 を書き込む				
PEC エラー	PECERR	PECERRCF=1 を書き込む				
タイムアウト／t_{LOW} エラー	TIMEOUT	TIMEOUTCF=1 を書き込む				
SMBus アラート	ALERT	ALERTCF=1 を書き込む				

1. WUPEN がセットされている場合。

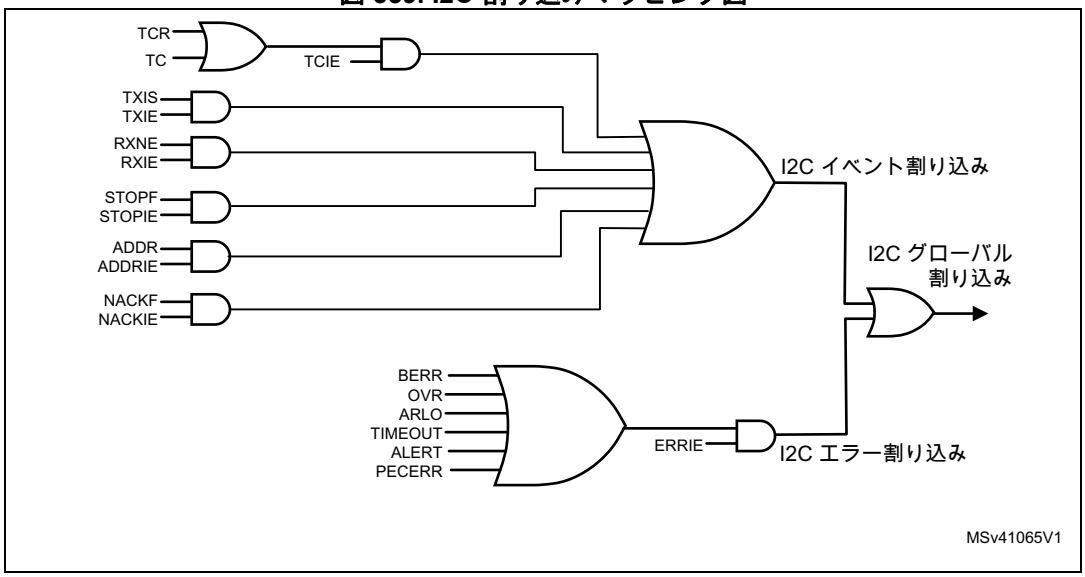
製品実装に応じて、これらの割り込みイベントのすべてで同じ割り込みベクトルを共有するか（I2C グローバル割り込み）、2 つの割り込みベクトル（I2C イベント割り込みと I2C エラー割り込み）にグループ化することができます。詳細については、*表 130：NVIC*を参照してください。

I2C 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で I2C IRQ チャネルを設定し、有効にします。
2. 割り込みを生成するように I2C を設定します。

I2C ウェイクアップイベントは EXTI コントローラに接続されます（*セクション 20：拡張割り込み／イベントコントローラ（EXTI）*を参照）。

図 559. I2C 割り込みマッピング図



47.7 I2C レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタは、ワード（32 ビット）単位でアクセスされます。

47.7.1 制御レジスタ 1（I2C_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERT EN	SMBD EN	SMBH EN	GCEN	WUPE N	NOSTR ETCH	SBC
								rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMA EN	TXDMA EN	Res.	ANF OFF	DNF				ERRIE	TCIE	STOP IE	NACK IE	ADDR IE	RXIE	TXIE	PE
rw	rw		rw	rw				rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **PECEN**：PEC イネーブル

0：PEC 計算は無効です。

1：PEC 計算は有効です。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。***

ビット 22 **ALERTEN**：SMBus アラート有効

デバイスモード（SMBHEN=0）：

0：SMBA ピンをハイにリリースし、アラート応答アドレスヘッダを無効にします。0001100x の後に NACK が続きます。

1：SMBA ピンをローに駆動し、アラート応答アドレスヘッダを有効にします：0001100x の後に ACK が続きます。

ホストモード（SMBHEN=1）：

0：SMBus アラートピン（SMBA）はサポートされません。

1：SMBus アラートピン（SMBA）はサポートされます。

注： ***ALERTEN=0 のときには、SMBA ピンを標準 GPIO として使用できます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 21 **SMBDEN**：SMBus デバイスデフォルトアドレス有効

0：デバイスデフォルトアドレス無効。アドレス 0b1100001x は NACK されます。

1：デバイスデフォルトアドレス有効。アドレス 0b1100001x は ACK されます。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。***

ビット 20 **SMBHEN**：SMBus ホストアドレス有効

0：ホストアドレス無効。アドレス 0b0001000x は NACK されます。

1：ホストアドレス有効。アドレス 0b0001000x は ACK されます。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。***

ビット 19 **GCEN**：同報イネーブル

0：同報は無効です。アドレス 0b00000000 は NACK されます。

1：同報は有効です。アドレス 0b00000000 は ACK されます。

ビット 18 **WUPEN**：STOP モードからのウェイクアップ有効

0：STOP モードからのウェイクアップ無効。

1：STOP モードからのウェイクアップ有効。

注： ***STOP モードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。***

注： ***WUPEN は、DNF = 0000 のときのみセットできます。***

ビット 17 **NOSTRETCH**：クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。マスタモードではクリアされたままでなければなりません。

0：クロックストレッチ有効

1：クロックストレッチ無効

注： ***このビットは、I2C が無効（PE = 0）のときのみプログラムできます。***

ビット 16 **SBC**：スレーブバイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

0：スレーブバイト制御無効

1：スレーブバイト制御有効

ビット 15 **RXDMAEN**：DMA 受信リクエスト有効
0：DMA モードは受信に無効
1：DMA モードは受信に有効

ビット 14 **TXDMAEN**：DMA 送信リクエスト有効
0：DMA モードは送信に無効
1：DMA モードは送信に有効

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **ANFOFF**：アナログノイズフィルタ OFF
0：アナログノイズフィルタ有効
1：アナログノイズフィルタ無効
注：　このビットは、I2C が無効（PE = 0）のときのみプログラムできます。

ビット 11:8 **DNF[3:0]**：デジタルノイズフィルタ
これらのビットは、SDA および SCL 入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大 DNF[3:0] * t_{I2CCLK} の長さのスパイクを除去します。
0000：デジタルフィルタ無効
0001：デジタルフィルタは有効であり、最大 1 t_{I2CCLK} の除去能力を持ちます。
...
1111：デジタルフィルタは有効であり、最大 15 t_{I2CCLK} の除去能力を持ちます。
注：　アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。
このフィルタは、I2C が無効（PE = 0）のときのみプログラムできます。

ビット 7 **ERRIE**：エラー割り込み有効
0：エラー検出割り込み無効
1：エラー検出割り込み有効
注：　次のようなエラーが発生すると、割り込みが生成されます：
アービトレーション喪失（ARLO）
バスエラー検出（BERR）
オーバーラン／アンダーラン（OVR）
タイムアウト検出（TIMEOUT）
PEC エラー検出（PECERR）
アラートピンイベント検出（ALERT）

ビット 6 **TCIE**：転送完了割り込み有効
0：転送完了割り込み無効
1：転送完了割り込み有効
注：　次のようなイベントが発生すると、割り込みが生成されます：
転送完了（TC）
転送完了再ロード（TCR）

ビット 5 **STOPIE**：STOP 検出割り込み有効
0：STOP 検出（STOPF）割り込み無効
1：STOP 検出（STOPF）割り込み有効

ビット 4 **NACKIE**：非確認応答受信割り込み有効
0：非確認応答（NACKF）受信割り込み無効
1：非確認応答（NACKF）受信割り込み有効

ビット 3 **ADDRIE**：アドレス一致割り込み有効（スレーブのみ）
0：アドレス一致（ADDR）割り込み無効
1：アドレス一致（ADDR）割り込み有効

ビット 2 **RXIE**：RX 割り込み有効

0：受信（RXNE）割り込み無効
1：受信（RXNE）割り込み有効

ビット 1 **TXIE**：TX 割り込み有効

0：送信（TXIS）割り込み無効
1：送信（TXIS）割り込み有効

ビット 0 **PE**：ペリフェラルは有効です。

0：ペリフェラルは無効です。
1：ペリフェラルは有効です。

注： PE=0 のとき、I2C SCL および SDA ラインはリリースされます。内部ステートマシンおよびステータスビットはリセット値に戻されます。クリアされたときには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれる必要があります。

47.7.2 制御レジスタ 2（I2C_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PEC BYTE	AUTO END	RE LOAD	NBYTES[7:0]							
					rs	rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACK	STOP	START	HEAD 10R	ADD10	RD_ WRN	SADD[9:0]									
rs	rs	rs	rw	rw	rw	rw									

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PECBYTE**：パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PEC が転送されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、また、PE=0 のとき、ハードウェアによってクリアされます。

0：PEC 転送なし。

1：PEC 送信／受信がリクエストされます。

注： ***このビットに 0 を書き込んでも、効果はありません。***

このビットは、RELOAD がセットされているときには効果がありません。

このビットは、SBC=0 のとき、スレーブモードでは効果がありません。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 25 **AUTOEND**：自動終了モード（マスタモード）

このビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェア終了モード：NBYTES データが転送されると TC フラグがセットされ、SCL ローをストレッチします。

1：自動終了モード：NBYTES データが転送されると、STOP コンディションが自動的に送信されます。

注： ***このビットは、スレーブモードまたは RELOAD ビットがセットされているときには効果がありません。***

ビット 24 **RELOAD**：NBYTES 再ロードモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送は、NBYTES データの転送後、完了します（STOP または RESTART が続きます）。

1：転送は、NBYTES データの転送後に完了しません（NBYTES が再ロードされます）。NBYTES データが転送されると TCR フラグがセットされ、SCL ローをストレッチします。

ビット 23:16 **NBYTES[7:0]**：バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC=0 のスレーブモードでは効果がありません。

注： ***START ビットがセットされているときに、これらのビットを変更することはできません。***

ビット 15 **NACK**：NACK 生成（スレーブモード）

このビットはソフトウェアによってセットされ、NACK が送信されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、または PE=0 のとき、ハードウェアによってクリアされます。

0：現在の受信バイト後に ACK が送信されます。

1：現在の受信バイト後に NACK が送信されます。

注： ***このビットに 0 を書き込んでも、効果はありません。***

このビットは、スレーブモードでのみ使用されます：マスタレシーバモードでは、NACK ビットの値にかかわらず、STOP または RESTART コンディション前の最後のバイト後に NACK が自動的に生成されます。

スレーブレシーバ NOSTRETCH モードでオーバーランが発生すると、NACK ビットの値にかかわらず、NACK が自動的に生成されます。

ハードウェア PEC チェックが有効なとき（PECBYTE=1）、PEC 確認応答値は NACK 値に依存しません。

ビット 14 **STOP**：STOP 生成（マスタモード）

このビットはソフトウェアによってセットされ、STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

マスタモード：

0：STOP 生成なし。

1：現在のバイト転送後の STOP 生成。

注： ***このビットに 0 を書き込んでも、効果はありません。***

ビット 13 **START**：START 生成

このビットはソフトウェアによってセットされ、START とアドレスシーケンスが送信された後、アービトレーション喪失によって、タイムアウトエラー検出によって、または PE=0 のときに、ハードウェアによってクリアされます。I2C_ICR レジスタの ADDRCF ビットに "1" を書き込むことによって、ソフトウェアによってクリアすることもできます。

0：START 生成なし。

1：RESTART/START 生成：

– I2C がすでにマスタモードであり、AUTOEND=0 の場合、このビットをセットすると、RELOAD=0 のとき、NBYTES 転送の終了後に REPEATED START コンディションが生成されます。

– そうでない場合、このビットをセットすると、バスがフリーになると、START コンディションが生成されます。

注：　このビットに 0 を書き込んでも、効果はありません。

バスが BUSY の場合、または I2C がスレーブモードの場合でも、START ビットをセットできます。

このビットは、RELOAD がセットされているときには効果がありません。10 ビットアドレッシングモードでは、アドレスの最初の部分で NACK が受信される場合、START ビットはハードウェアによってクリアされず、START ビットがソフトウェアによってクリアされない限り、マスタはアドレスシーケンスを再送します。

ビット 12 **HEAD10R**：読み出し方向のみの 10 ビットアドレスヘッダ（マスタレシーバモード）

0：マスタは完全な 10 ビットスレーブアドレス読み出しシーケンスを送信します：START + 2 バイトの書き込み方向の 10 ビットアドレス+ RESTART +読み出し方向の 10 ビットアドレスの最初の 7 ビット。

1：マスタは 10 ビットアドレスの最初の 7 ビットのみを送信し、その後に読み出し方向を送信します。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 11 **ADD10**：10 ビットアドレッシングモード（マスタモード）

0：マスタは 7 ビットアドレッシングモードで動作します。

1：マスタは 10 ビットアドレッシングモードで動作します。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 10 **RD_WRN**：転送方向（マスタモード）

0：マスタは書き込み転送をリクエストします。

1：マスタは読み出し転送をリクエストします。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 9:8 **SADD[9:8]**：スレーブアドレスビット 9:8（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

これらのビットは無視されます。

10 ビットアドレッシングモード（ADD10=1）：

これらのビットには、送信されるスレーブアドレスのビット 9:8 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 7:1 **SADD[7:1]**：スレーブアドレスビット 7:1（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

これらのビットには、送信される 7 ビットのスレーブアドレスを書き込みます。

10 ビットアドレッシングモード（ADD10=1）：

これらのビットには、送信されるスレーブアドレスのビット 7:1 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 0 **SADD0**：スレーブアドレスビット 0（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

このビットは無視されます。

10 ビットアドレッシングモード（ADD10=1）：

このビットには、送信されるスレーブアドレスのビット 0 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

参考資料

47.7.3 Own Address 1 レジスタ（I2C_OAR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA1EN	Res.	Res.	Res.	Res.	OA1 MODE	OA1[9:8]		OA1[7:1]							OA1[0]
rw					rw	rw		rw							rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA1EN**：Own Address 1 有効
0：Own Address 1 は無効です。受信されたスレーブアドレス OA1 は NACK されます。
1：Own Address 1 は有効です。受信されたスレーブアドレス OA1 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OA1MODE** Own Address 1 10 ビットモード
0：Own Address 1 は 7 ビットアドレスです。
1：Own Address 1 は 10 ビットアドレスです。
注： このビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 9:8 **OA1[9:8]**：インタフェースアドレス
7 ビットアドレッシングモード：無視されます。
10 ビットアドレッシングモード：アドレスのビット 9:8。
注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 7:1 **OA1[7:1]**：インタフェースアドレス
7 ビットアドレスモード：7 ビットアドレス
10 ビットアドレッシングモード：10 ビットアドレスのビット 7:1。
注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 0 **OA1[0]**：インタフェースアドレス
7 ビットアドレッシングモード：無視されます。
10 ビットアドレッシングモード：アドレスのビット 0。
注： このビットは、OA1EN=0 のときのみ書き込むことができます。

47.7.4 Own Address 2 レジスタ （I2C_OAR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2EN	Res.	Res.	Res.	Res.	OA2MSK[2:0]			OA2[7:1]							Res.
rw					rw			rw							

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA2EN**：Own Address 2 有効
0：Own Address 2 は無効です。受信されたスレーブアドレス OA2 は NACK されます。
1：Own Address 2 は有効です。受信されたスレーブアドレス OA2 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **OA2MSK[2:0]**：Own Address 2 マスク
000：マスクなし。
001：OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。
010：OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。
011：OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。
100：OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。
101：OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。
110：OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。
111：OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7 ビット受信アドレスが確認応答されます。

注： ***これらのビットは、OA2EN=0 のときのみ書き込むことができます。***
OA2MSK が 0 でなくなると、予約済み I2C アドレス（0b0000xxx および 0b1111xxx）は、比較が一致した場合でも確認応答されません。

ビット 7:1 **OA2[7:1]**：インタフェースアドレス
7 ビットアドレスモード：7 ビットアドレス

注： ***これらのビットは、OA2EN=0 のときのみ書き込むことができます。***

ビット 0 予約済みであり、リセット値に保持する必要があります。

47.7.5 タイミングレジスタ（I2C_TIMINGR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]			
rw								rw				rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLH[7:0]								SCLL[7:0]							
rw								rw							

ビット 31:28 **PRESC[3:0]**：タイミングプリスケーラ

このフィールドは、データのセットアップおよびホールドカウンタ（*1882 ページのI2C のタイミング* を参照）と SCL ハイおよびローレベルカウンタ（*1897 ページのI2C マスタ初期化* を参照）に使用されるクロック周期 t_{PRESC} を生成するために、i2c_ker_ck をプリスケールするのに使用されます。

t_{PRESC} = (PRESC+1) x t_{I2CCLK}

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **SCLDEL[3:0]**：データセットアップ時間

このフィールドは、SDA エッジと SCL 立ち上がりエッジの間に遅延 t_{SCLDEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SCLDEL} の間ローにストレッチされます。

t_{SCLDEL} = (SCLDEL+1) x t_{PRESC}

注： t_{SCLDEL} は、$t_{SU:DAT}$ タイミングを生成するために使用されます。

ビット 19:16 **SDADEL[3:0]**：データホールド時間

このフィールドは、SCL 立ち下がりエッジと SDA エッジの間に遅延 t_{SDADEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SDADEL} の間ローにストレッチされます。

t_{SDADEL}= SDADEL x t_{PRESC}

注： SDADEL は、$t_{HD:DAT}$ タイミングを生成するために使用されます。

ビット 15:8 **SCLH[7:0]**：SCL ハイ周期（マスタモード）

このフィールドは、マスタモードで SCL ハイ周期を生成するために使用されます。

t_{SCLH} = (SCLH+1) x t_{PRESC}

注： SCLH は、$t_{SU:STO}$ および $t_{HD:STA}$ タイミングを生成するためにも使用されます。

ビット 7:0 **SCLL[7:0]**：SCL ロー周期（マスタモード）

このフィールドは、マスタモードで SCL ロー周期を生成するために使用されます。

t_{SCLL} = (SCLL+1) x t_{PRESC}

注： SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングを生成するためにも使用されます。

注： このレジスタは、I2C が無効（PE=0）のときに設定する必要があります。

注： STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

47.7.6 タイムアウトレジスタ（I2C_TIMEOUTR）

アドレスオフセット：0x14

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.	Res.	Res.	TIMEOUTB [11:0]											
rw				rw											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA [11:0]											
rw			rw	rw											

ビット 31 **TEXTEN**：拡張クロックタイムアウト有効

0：拡張クロックタイムアウト検出は無効です。

1：拡張クロックタイムアウト検出は有効です。$t_{LOW:EXT}$ を超える累積 SCL ストレッチが I2C インタフェースによって行われると、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TIMEOUTB [11:0]**：バスタイムアウト B

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます：

マスタモードでは、マスタ累積クロックロー拡張時間（$t_{LOW:MEXT}$）が検出されます。

スレーブモードでは、スレーブ累積クロックロー拡張時間（$t_{LOW:SEXT}$）が検出されます。

$t_{LOW:EXT}$ = (TIMEOUTB+1) x 2048 x t_{I2CCLK}

注：　これらのビットは、TEXTEN=0 のときのみ書き込むことができます。

ビット 15 **TIMOUTEN**：クロックタイムアウト有効

0：SCL タイムアウト検出は無効です。

1：SCL タイムアウト検出は有効です。SCL が $t_{TIMEOUT}$（TIDLE=0）を超えてローであるか、t_{IDLE}（TIDLE=1）を超えてハイであった場合、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TIDLE**：アイドルクロックタイムアウト検出

0：TIMEOUTA は、SCL ロータイムアウトの検出に使用されます。

1：TIMEOUTA は、SCL と SDA の両方のハイタイムアウト（バスアイドル条件）の検出に使用されます。

注：　このビットは、TIMOUTEN=0 のときのみ書き込むことができます。

ビット 11:0 **TIMEOUTA[11:0]**：バスタイムアウト A

このフィールドは、以下を設定するために使用されます：

– TIDLE=0 のときの SCL ロータイムアウト条件 $t_{TIMEOUT}$

$t_{TIMEOUT}$ = (TIMEOUTA+1) x 2048 x t_{I2CCLK}

– TIDLE=1 のときのバスアイドル条件（SCL と SDA の両方のハイ）

t_{IDLE} = (TIMEOUTA+1) x 4 x t_{I2CCLK}

注：　これらのビットは、TIMOUTEN=0 のときのみ書き込むことができます。

注：　SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 47.3：I2C の実装 を参照してください。

47.7.7 割り込みおよびステータスレジスタ（I2C_ISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR
								r							r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.	ALERT	TIME OUT	PEC ERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
r		r	r	r	r	r	r	r	r	r	r	r	r	rs	rs

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 **ADDCODE[6:0]**：アドレス一致コード（スレーブモード）

これらのビットは、アドレス一致イベントが発生したときに（ADDR = 1）、受信したアドレスで更新されます。

10 ビットアドレスの場合、ADDCODE は 10 ビットのヘッダとその後のアドレスの 2 つの MSB を示します。

ビット 16 **DIR**：転送方向（スレーブモード）

このフラグは、アドレス一致イベントが発生したときに（ADDR=1）、更新されます。

0：書き込み転送、スレーブはレシーバモードになります。

1：読み出し転送、スレーブはトランスミッタモードになります。

ビット 15 **BUSY**：バスビジー

このフラグは、バスで通信が進行中であることを示します。START コンディションが検出されたときに、ハードウェアによってセットされます。STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERT**：SMBus アラート

このフラグは、SMBHEN=1（SMBus ホスト設定）、ALERTEN=1、および SMBALERT イベント（立ち下がりエッジ）が SMBA ピンで検出されたときに、ハードウェアによってセットされます。ALERTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 12 **TIMEOUT**：タイムアウトまたは t_{LOW} 検出フラグ

このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 11 **PECERR**：受信時の PEC エラー

このフラグは、受信した PEC が PEC レジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくない PEC の受信後、NACK が自動的に送信されます。PECCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 10 **OVR**：オーバーラン／アンダーラン（スレーブモード）

このフラグは、NOSTRETCH=1 のスレーブモードで、オーバーラン／アンダーランエラーが発生したときに、ハードウェアによってセットされます。OVRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 9 **ARLO**：アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 8 **BERR**：バスエラー

このフラグは、ペリフェラルが転送に関与しているので、START または STOP コンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 7 **TCR**：転送完了再ロード

このフラグは、RELOAD=1 および NBYTES データが転送されたときに、ハードウェアによってセットされます。NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

このフラグは、マスタモード、または SBC ビットがセットされているときのスレーブモードでのみ使用されます。

ビット 6 **TC**：転送完了（マスタモード）

このフラグは、RELOAD=0、AUTOEND=0、および NBYTES データが転送されたときに、ハードウェアによってセットされます。START ビットまたは STOP ビットがセットされたときに、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 5 **STOPF**：STOP 検出フラグ

このフラグは、バス上で STOP コンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます：

– マスタとして。ただし、STOP コンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 4 **NACKF**：非確認応答受信フラグ

このフラグは、バイト送信後に NACK を受信したときに、ハードウェアによってセットされます。NACKCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 3 **ADDR**：アドレス一致（スレーブモード）

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの 1 つに一致したときに、ハードウェアによってセットされます。ADDRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 2 **RXNE**：受信データレジスタノットエンプティ（レシーバ）

このビットは、受信データが I2C_RXDR レジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2C_RXDR が読み出されたときにクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 1 **TXIS**：送信割り込みステータス（トランスミッタ）

このビットは、I2C_TXDR レジスタが空であり、送信データを I2C_TXDR レジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、NOSTRETCH=1 のときのみ、ソフトウェアによって 1 を書き込んで、TXIS イベントを生成することができます（TXIE=1 の場合に割り込み、または TXDMAEN=1 の場合に DMA リクエスト）。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 0 **TXE**：送信データレジスタエンプティ（トランスミッタ）

このビットは、I2C_TXDR レジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、ソフトウェアによって 1 を書き込んで、送信データレジスタ I2C_TXDR をフラッシュできます。

注：　このビットは、PE=0 のとき、ハードウェアによってセットされます。

47.7.8　割り込みクリアレジスタ（I2C_ICR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ALERT CF	TIM OUTCF	PECCF	OVRCF	ARLO CF	BERR CF	Res.	Res.	STOP CF	NACK CF	ADDR CF	Res.	Res.	Res.
		w	w	w	w	w	w			w	w	w			

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERTCF**：アラートフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの ALERT フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 12 **TIMOUTCF**：タイムアウト検出フラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの TIMEOUT フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 11 **PECCF**：PEC エラーフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの PECERR フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 47.3：I2C の実装 を参照してください。

ビット 10 **OVRCF**：オーバーラン／アンダーランフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの OVR フラグがクリアされます。

ビット 9 **ARLOCF**：アービトレーション喪失フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ARLO フラグがクリアされます。

ビット 8 **BERRCF**：バスエラーフラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの BERRF フラグがクリアされます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **STOPCF**：STOP 検出フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの STOPF フラグがクリアされます。

ビット 4 **NACKCF**：非確認応答フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ACKF フラグがクリアされます。

ビット 3 **ADDRCF**：アドレス一致フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ADDR フラグがクリアされます。このビットに 1 を書き込むと、I2C_CR2 レジスタの START ビットもクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

47.7.9 PEC レジスタ（I2C_PECR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PEC[7:0]** パケットエラーチェックレジスタ
PECEN=1 のとき、このフィールドは内部 PEC を含みます。
PEC は、PE=0 のとき、ハードウェアによってクリアされます。

注： ***SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 47.3：I2C の実装 を参照してください。***

47.7.10 受信データレジスタ（I2C_RXDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXDATA[7:0]** 8 ビット受信データ
I^2C バスから受信したデータバイト。

47.7.11 送信データレジスタ（I2C_TXDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXDATA[7:0]** 8 ビット送信データ
I^2C バスに送信されるデータバイト。
注： これらのビットは、TXE=1 のときのみ書き込むことができます。

47.7.12 I2C レジスタマップ

次の表に、I2C のレジスタマップとリセット値を示します。

表 374. I2C レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	I2C_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC	RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
	リセット値									0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x4	I2C_CR2	Res.	Res.	Res.	Res.	Res.	PECBYTE	AUTOEND	RELOAD	NBYTES[7:0]								NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	I2C_OAR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA1EN	Res.	Res.	Res.	Res.	OA1MODE	OA1[9:0]									
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0
0xC	I2C_OAR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]	OA2[7:1]									Res.
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	
0x10	I2C_TIMINGR	PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]				SCLH[7:0]								SCLL[7:0]							
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	I2C_TIMEOUTR	TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]												TIMOUTEN	Res.		TIDLE	TIMEOUTA[11:0]											
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	I2C_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR	BUSY	Res.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
	リセット値									0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	I2C_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
	リセット値																			0	0	0	0	0	0			0	0	0			
0x20	I2C_PECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x24	I2C_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 374. I2C レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	I2C_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

48 USART（Universal synchronous asynchronous receiver transmitter）

このセクションでは、ユニバーサル同期非同期レシーバトランスミッタ（USART）について説明します。

48.1 USART の概要

USART を使用すると、業界標準の NRZ 非同期シリアルデータフォーマットを必要とする外部機器と柔軟性の高い全二重データ交換を行うことができます。フラクショナルボーレートジェネレータによって、非常に広範囲のボーレートが達成できます。

USART は同期式単方向通信と半二重単線通信の両方のほか、LIN（Local Interconnection Network）、スマートカードプロトコル、IrDA（infrared data association）SIR ENDEC 仕様、およびモデム動作（CTS/RTS）もサポートします。マルチプロセッサ通信もサポートされています。

マルチバッファ設定で DMA（直接メモリアクセス）を使用することによって、ハイスピードデータ通信が可能です。

参考資料

48.2 USART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- 16 倍または 8 倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の最適な調整を達成
- ボーレートジェネレータシステム
- データ送信および受信用の 2 つの内部 FIFO
  各 FIFO はソフトウェアで有効／無効にすることができ、ステータスフラグを装備
- 共通にプログラム可能な送信および受信ボーレート
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- 自動ボーレート検出
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 同期通信のための同期マスタ／スレーブモードとクロック出力／入力
- SPI スレーブ送信アンダーランエラーフラグ
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 通信制御／エラー検出フラグ
- パリティ制御：
  – パリティビットの送信
  – 受信したデータバイトのパリティ検査
- フラグ付き割り込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ

参考資料

48.3 USART の拡張機能

- LIN マスタの同期ブレーク送信機能と LIN スレーブのブレーク検出機能
  - USART が LIN 用にハードウェア設定されている場合、13 ビットのブレーク生成と 10/11 ビットのブレーク検出
- 通常モードで 3/16 ビット期間をサポートする IrDA SIR エンコーダデコーダ
- スマートカードモード
  - ISO/IEC 7816-3 標準で定義されているスマートカードの T="0"および T="1"非同期プロトコルをサポート
  - スマートカード動作用に 0.5 個および 1.5 個のストップビット
- Modbus 通信のサポート
  - タイムアウト機能
  - CR/LF キャラクタ認識

48.4 USART の実装

表 375 に、STM32H7x3 デバイスでの USART の実装を示します。

表 375. USART/LPUART の機能

USART のモード／機能(1)	USART1/2/3/6	UART4/5/7/8	LPUART
モデムのハードウェアフロー制御	X	X	X
DMA を使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDEC ブロック	X	X	-
LIN モード	X	X	-
デュアルクロックドメインと低電力モードからのウェイクアップ	X	X	X
レシーバタイムアウト割り込み	X	X	-
Modbus 通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USART データ長	7、8、および 9 ビット		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFO サイズ	16		

1. X：サポートされています。

参考資料

48.5 USART の機能詳細

48.5.1 USART のブロック図

図 560. USART のブロック図

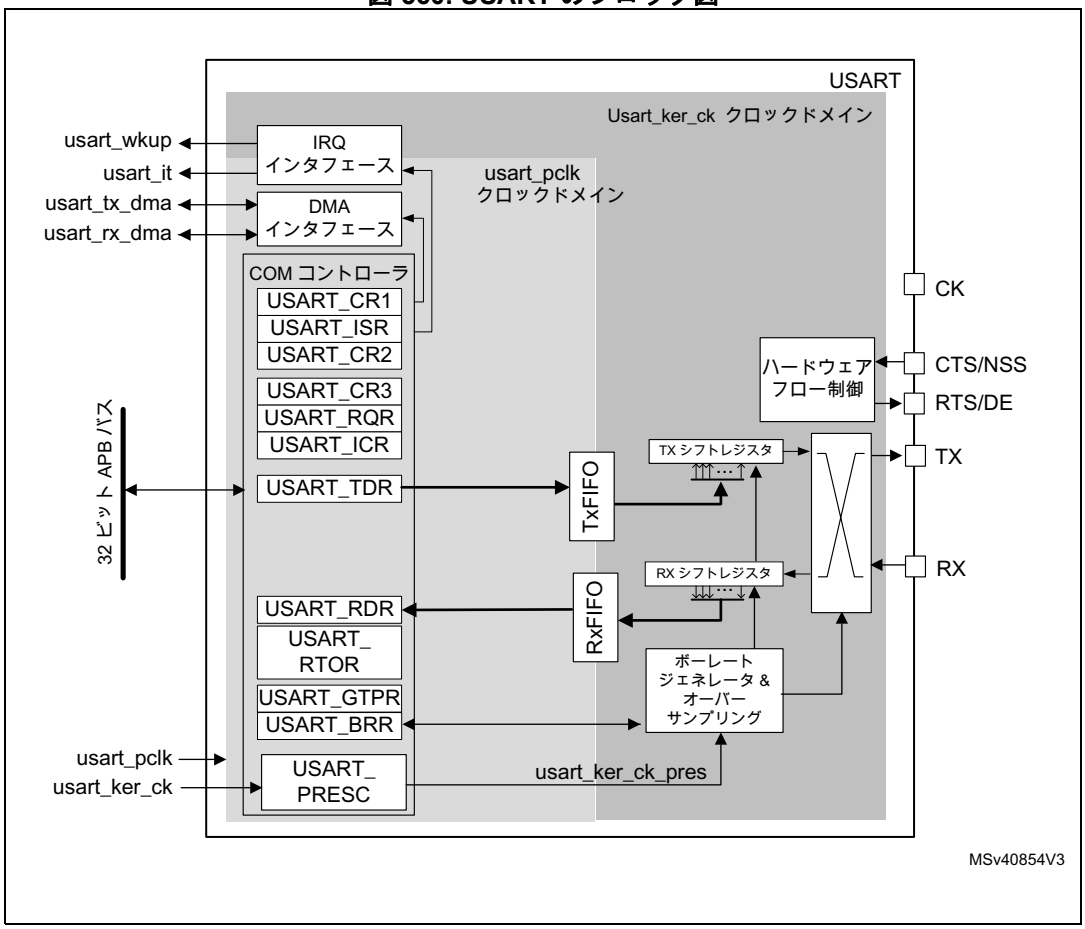


図 560 : USART のブロック図 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロックドメインが示されています。

- **usart_pclk** クロックドメイン

  Usart_pclkクロック信号はペリフェラルバスインタフェースにクロック供給します。USART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **Usart_ker_ck** カーネルクロックドメイン

  usart_ker_ck は USART クロックソースです。**usart_pclk** とは独立で、RCC によって供給されます。したがって、**usart_ker_ck** クロックが停止しているときでも、USART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**usart_ker_ck** クロックは **usart_pclk** クロックと同じになります。

usart_pclk と **usart_ker_ck** の間には何の制約もありません。**usart_ker_ck** は **usart_pclk** より速くすることも遅くすることもできます。唯一の限界は、十分に速く通信を管理できるソフトウェアの能力です。

USART が SPI スレーブモードで動作するときは、USART は、外部マスタ SPI デバイスによって提供される外部 SCLK 信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。**Usart_ker_ck** クロックは CK 入力のクロックより少なくとも 3 倍は速くなければなりません。

48.5.2 USART 信号

USART 双方向通信

USART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）

  RX はシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用されています。それによって、有効な入力データとノイズを区別しています。

- **TX**（送信データ出力）

  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、データを送信する必要がないとき、TX ピンはハイになります。単線およびスマートカードモードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）

  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。

- **RTS**（Request To Send）

  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェア制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（ドライバイネーブル）

  この信号は、外部トランシーバの送信モードを有効にします。

注： ***DE と RTS は同じピンを共有します。***

同期マスタ／スレーブモードおよびスマートカードモード

同期マスタ／スレーブモードおよびスマートカードモードでは、以下のピンが必要です。

- CK

  このピンは同期マスタモードおよびスマートカードモードで、クロック出力として機能します。

  同期スレーブモードでは、クロック入力として機能します。

  同期マスタモードでは、このピンは、SPI マスタモードに対応する同期送信用のトランスミッタデータクロックを出力します（スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します）。これと並行して、RX ピンでデータを同期受信できます。このメカニズムを使用して、シフトレジスタを持つペリフェラル（例：LCD ドライバ）を制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。

  スマートカードモードでは、CK 出力はスマートカードにクロックを供給します。

- NSS

  このピンは、同期スレーブモードでスレーブ選択入力として機能します。

注： ***NSS と CTS は同じピンを共有します。***

48.5.3 USART キャラクタの説明

ワード長は、USART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 ビットに設定できます（*図 561* を参照）。

- 7 ビットのキャラクタ長：M[1:0] = “10”
- 8 ビットのキャラクタ長：M[1:0] = “00”
- 9 ビットのキャラクタ長：M[1:0] = “01”

注： ***7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。***

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

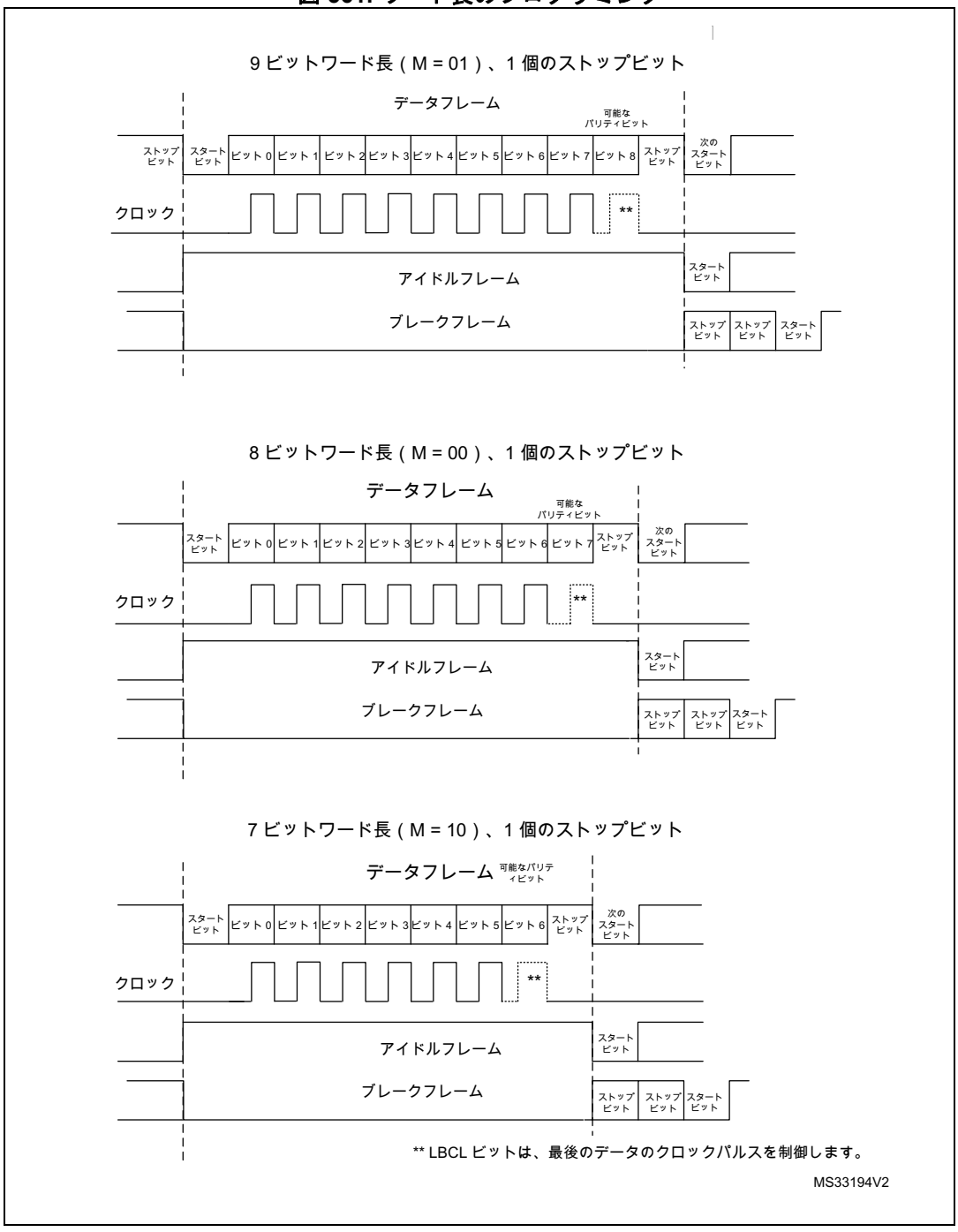
ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

参考資料

図 561. ワード長のプログラミング



48.5.4 USART の FIFO と閾値

USART は FIFO モードで動作できます。

USART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、USART_CR1 レジスタの FIFOEN（ビット 29）をセットすることによって有効になります。このモードは、UART、SPI、およびスマートカードモードでのみ、サポートされます。

最大のデータワード長が 9 ビットなので、TXFIFO は 9 ビット幅です。しかし、RXFIFO のデフォルト幅は 12 ビットです。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： ***受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。***

ステータスフラグは、USART_ISR レジスタで入手可能です。

Tx および Rx 割り込みがトリガされる TXFIFO および RXFIFO のレベルを設定することができます。これらの閾値は、USART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、Rx 割り込みが生成されます。

  この場合、USART_ISR レジスタの RXFT フラグがセットされます。これは、RXFTCFG 分のデータが受信されたことを意味します。つまり、1 データが USART_RDR にあり、（RXFTCFG - 1）データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された（FIFO サイズ -1 のデータが RXFIFO に入り、1 データが USART_RDR に入った）時に、RXFT フラグがセットされます。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO 内の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、Tx 割り込みが生成されます。

48.5.5 USART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは TX ピンで出力され、対応するクロックパルスは SCLK ピンで出力されます。

キャラクタ送信

USART 送信時、データは LSB ファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、USART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます。

FIFO モードが有効なとき、送信データレジスタ（USART_TDR）に書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 ビット周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 0.5、1、1.5、または 2 に設定できます。

注： ***送信データを USART_TDR に書き込む前に、TE ビットをセットする必要があります。***

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

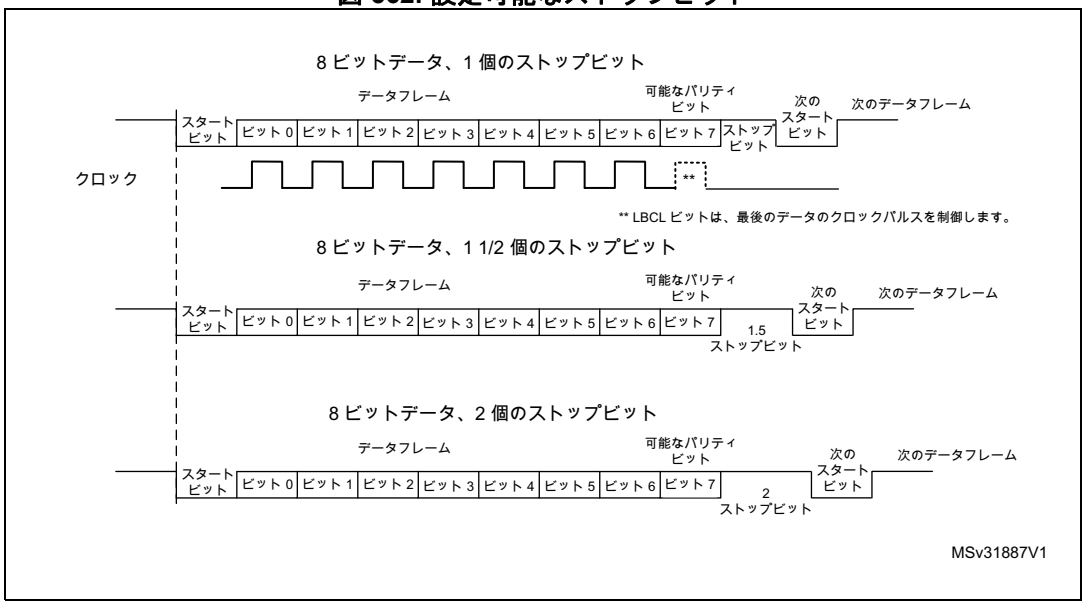
各キャラクタとともに送信されるストップビットの数は、USART_CR2 のビット 13、12 でプログラミングできます。

- ***1 個のストップビット***：ストップビット数のデフォルト値です。
- ***2 個のストップビット***：通常の USART モード、単線モード、およびモデムモードでサポートされます。
- ***1.5 個のストップビット***：スマートカードモードで使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10 個のロービット（M[1:0] = “00”のとき）または 11 個のロービット（M[1:0] = “01”のとき）または 9 個のロービット（M[1:0] = “10”のとき）の後に 2 個のストップビットが続きます（*図 562* を参照）。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 562. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. USART_BRR レジスタを使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAT）を選択します。*セクション 48.5.10 : USART マルチプロセッサ通信*の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを USART_TDR レジスタに書き込みます。シングルバッファの場合、送信される各データにこれを繰り返します。
   - FIFO モードが無効になっている場合は、USART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。
   - FIFO モードが有効になっている場合は、USART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、USART_TDR への書き込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。
8. USART_TDR レジスタに最後のデータを書き込んだら、TC=“1”になるまで待ちます。
   - FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   - FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

   このチェックは、USART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされます。同フラグは以下を示します。
  - データは USART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  - USART_TDR レジスタは空です。
  - 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、このフラグは割り込みを生成します。

  送信が行われているとき、USART_TDR レジスタへの書き込み命令によってデータが TDR バッファに格納されます。そして、データは現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、USART_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。
- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  - TXFIFO はフルではありません。
  - USART_TDR レジスタは空です。
  - 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。送信が行われているとき、USART_TDR レジスタへの書き込み命令によってデータが TXFIFO

に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

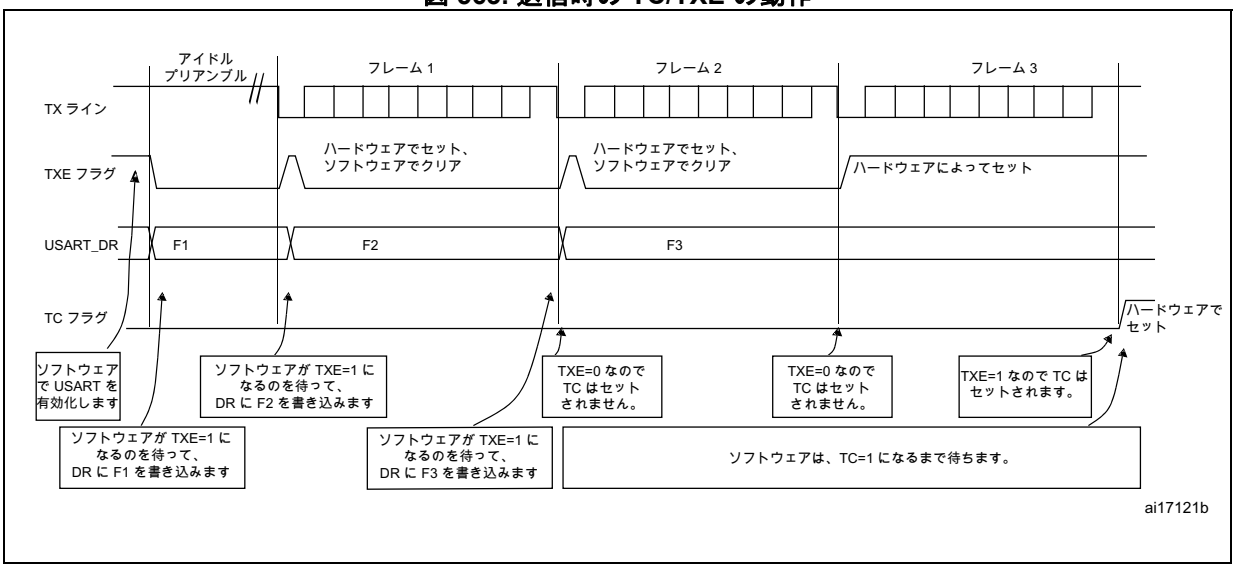
TXFIFO がフルではない場合、USART_TDR レジスタへの書き込み動作が終わった後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNFIE ビットがセットされている場合、このフラグは割り込みを生成します。

あるいは、TXFIFO 閾値に達した時、割り込みが生成され、データを FIFO に書き込むことができます。この場合、CPU は、プログラムされたトリガレベルによって定義されたデータのブロックを書き込むことができます。

フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC フラグはハイレベルになります。USART_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

USART_TDR レジスタに最後のデータを書き込んだ後は、USART を無効にしたり、マイクロコントローラを低電力モードにする前に TC がセットされるまで待つ必要があります（*図 563：送信時の TC/TXE の動作* を参照）。

図 563. 送信時の TC/TXE の動作



注： ***FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。***

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（*図 561* を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書き込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。USART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（ストップ）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、USART は、最初のデータフレームの前にアイドルフレームを送信します。

48.5.6 USART レシーバ

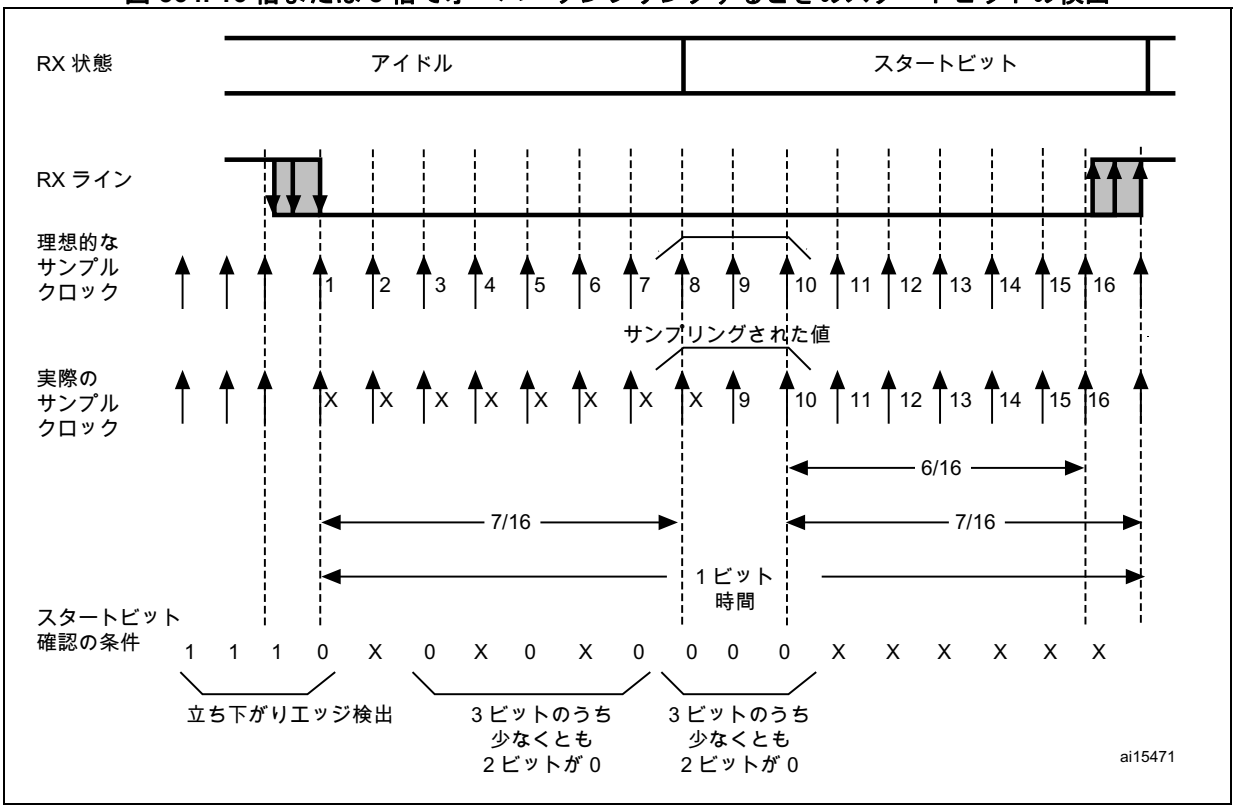
USART は、USART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

スタートビット検出シーケンスは、16 倍または 8 倍でオーバーサンプリングするときと同じです。

USART では、サンプルの特定シーケンスが認識されると、スタートビットが検出されます。このシーケンスは、1 1 1 0 X 0 X 0X 0X 0 X 0X 0 です。

図 564. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出



注： ***シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。***

サンプリングされた 3 つのビットが“0”の場合（第 3、第 5、および第 7 ビットでの最初のサンプリングで 3 つのビットが“0”であり、第 8、第 9、および第 10 ビットでの 2 回目のサンプリングでも 3 つのビットが“0”）、スタートビットが確認されます（RXNE フラグがセットされ、RXNEIE=“1”の場合は割り込みが生成されるか、または FIFO モードが有効な場合は、RXFNE フラグがセットされ、RXFNEIE=“1”の場合は割り込みが生成されます）。

次の場合、スタートビットは確認されますが、NE ノイズフラグがセットされます。

a) 両方のサンプリングについて、サンプリングされた 3 ビットのうちの 2 ビットが“0”の場合（第 3、第 5、および第 7 ビットのサンプリングと、第 8、第 9、および第 10 ビットのサンプリング）。

または

b) いずれか一方のサンプリングで（第 3、第 5、および第 7 ビットでのサンプリングまたは第 8、第 9、および第 10 ビットでのサンプリング）、3 つのビットのうち 2 つが“0”の場合。

上記のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

キャラクタの受信

USART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルトの設定）でシフトアウトされます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ USART_BRR を使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに“1”を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAR）を選択します。*セクション 48.5.10 : USART マルチプロセッサ通信*の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効の場合、RXNE ビットは、シフトレジスタの内容が RDR レジスタに転送されたことを示すためにセットされます。言い換えると、データは受信され、読み出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示すために RXFNE ビットがセットされます。USART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードが有効な場合は RXFNEIE）ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズ、パリティ、またはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効な場合は、RXNE フラグはバイトを受信するたびにセットされます。DMA が受信データレジスタを読み出すと、同フラグはクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる USART_RDR レジスタからの読み出しによって行われます。RXNE フラグは、USART_RQR レジスタの RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。オーバー

ランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE フラグをクリアする必要があります。

– FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE がセットされます。USART_RDR からの読み出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、USART_RQR の RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割り込みを生成します。あるいは、RXFIFO 閾値に達した時、割り込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割り込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていなくてキャラクタを受信すると、オーバーランエラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。前回のデータは USART_RDR レジスタを読み出すことによって、入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割り込みが生成されます。

- FIFO モードが有効な場合

シフトレジスタが転送される準備ができていて、受信 FIFO がフルのとき、オーバーランエラーが発生します。

RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから USART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

– ORE ビットがセットされます。
– RXFIFO の最初のエントリは失われません。それは、USART_RDR レジスタを読み出すことによって入手可能です。
– シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
– RXFNEIE ビットまたは EIE ビットがセットされている場合、割り込みが生成されます。

ORE ビットは、USART_ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： ***ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。***

FIFO モードが無効になっているとき、2 つの可能性があります。

- *RXNE="1"の場合、有効な最後のデータは、受信レジスタ（RDR）に格納され、読み出しが可能です。*
- *RXNE="0"の場合、最後の有効なデータはすでに読み出されたので、RDR レジスタには読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータが RDR レジスタで読み出されると同時に新しい（そして失われた）データが受信されると発生します。*

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（*セクション 8：リセットおよびクロック制御（RCC）*を参照）。クロックソースは、UE ビットのセットによって USART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

- USART を低電力モードで使用できること
- 通信速度

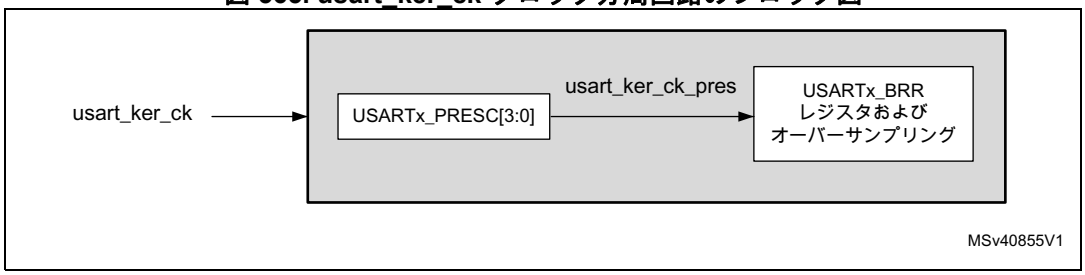
クロックソース周波数は、u

sart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、usart_ker_ck クロックソースは RCC で設定可能です（*セクション 8：リセットおよびクロック制御（RCC）*を参照）。そうしない場合、usart_ker_ck クロックは usart_pclk と同じになります。

usart_ker_ck クロックは、USART_PRESC レジスタで定義されたプログラム可能な係数によって分周できます。

図 565. usart_ker_ck クロック分周回路のブロック図



Usart_ker_ck ソースによっては、USART は MCU が低電力モードのときにデータを受信することができます。受信データと選択されたウェイクアップモードに応じて、USART より必要なときに MCU をウェイクアップし、ソフトウェアによる USART_RDR レジスタの読み出しによって、または DMA によって受信データを転送できます。

他のクロックソースの場合、USART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています（同期モードの場合を除く）。これにより、最大通信速度とノイズ／クロック精度の耐性との間の最適なトレードオフが図られます。

オーバーサンプリング方式は、USART_CR1 レジスタの OVER8 ビットをボーレートクロックの 16 倍または 8 倍にプログラムすることによって選択できます（*図 566* および *図 567* を参照）。

アプリケーションに応じて、

- 高速（最大 usart_ker_ck_pres/8）を達成するには 8 倍（OVER8=“1”）のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます（*セクション 48.5.8： 1965 ページのクロック偏差に対する USART レシーバの許容誤差* を参照）。
- クロック偏差に対するレシーバの許容誤差を増やすには、16 倍（OVER8=“0”）のオーバーサンプリングを選択します。この場合、最大速度は 最大 usart_ker_ck_pres/16 に制限されます（ここで、usart_ker_ck_pres は USART 入力クロックをプリスケーラで分周したものです）。

論理レベルの評価方法を選択するには、USART_CR3 レジスタの ONEBIT ビットをプログラミングします。次の 2 つのオプションが利用できます。

- 受信されたビットの中央にある 3 つのサンプルの多数決。この場合、多数決に使用された 3 つのサンプルが等しくないとき、NE ビットがセットされます。
- 受信されたビットの中央にある 1 つのサンプル。

  アプリケーションに応じて、
  - ノイズの多い環境で操作するときは 3 つのサンプルの多数決方式（ONEBIT=“0”）を選択します。そしてノイズが検出された場合（*図 376* を参照）は、サンプリング中にグリッチが発生していることになりますので、そのデータを除去します。
  - ラインがノイズフリーであるときは、1 つのサンプルによる方式（ONEBIT=“1”）を選択し、クロック偏差に対するレシーバの許容誤差を増やします（*セクション 48.5.8： 1965 ページのクロック偏差に対する USART レシーバの許容誤差* を参照）。この場合、NE ビットはセットされません。

フレーム内でノイズが検出された場合：

- RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）の立ち上がりエッジで、NE ビットがセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

NE ビットは、ICR レジスタの NFCF ビットをセットすることによってリセットされます。

注： ***ノイズエラーは SPI モードではサポートされていません。***

8 倍のオーバーサンプリングは、スマートカード、IrDA、および LIN の各モードでは使用できません。これらのモードでは、OVER8 ビットはハードウェアによって“0”に固定されています。

図 566. データサンプリング（16 倍のオーバーサンプリング）

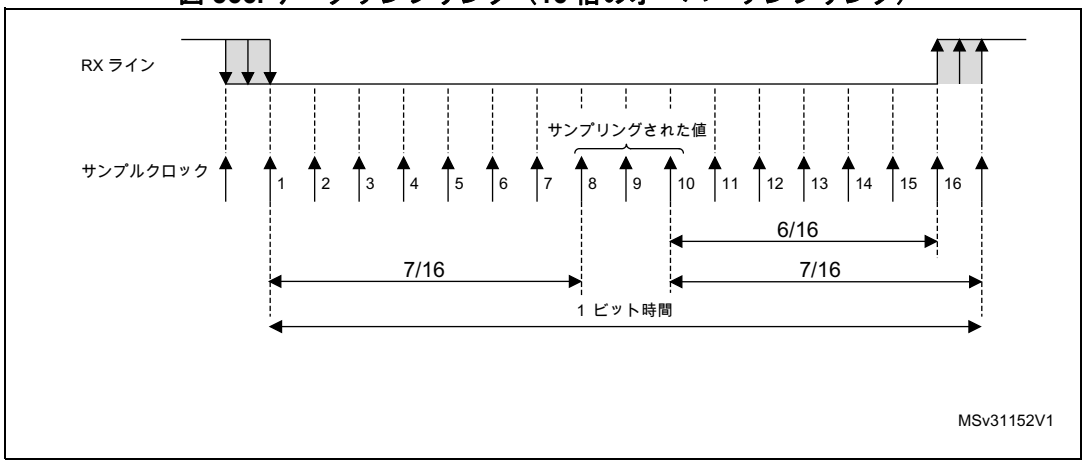


図 567. データサンプリング（8 倍のオーバーサンプリング）

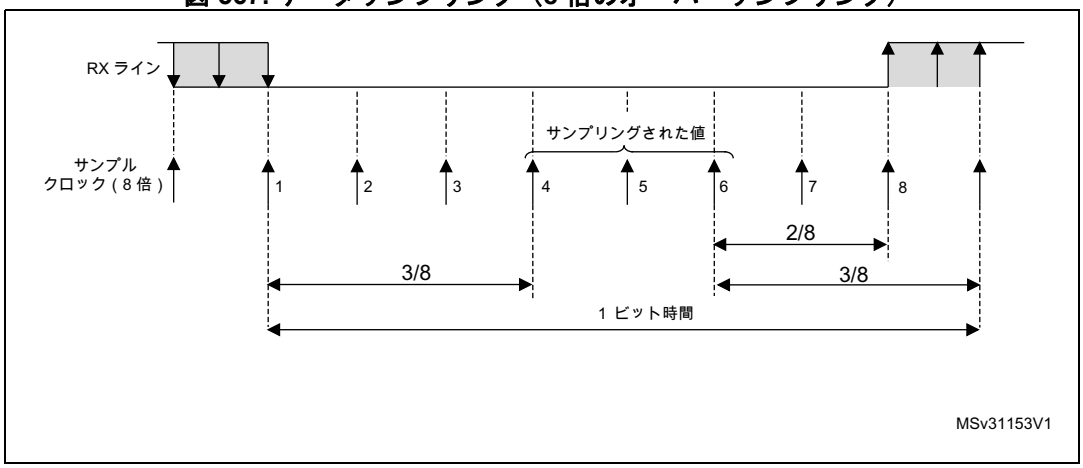


表 376. サンプリングされたデータからのノイズ検出

サンプリングされた値	NE ステータス	受信ビットの値
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタ（FIFO モードが有効な場合は RXFIFO）へ転送されます。
- 1 バイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

USART_ICR レジスタの FECF に“1”を書き込むことによって、FE ビットがリセットされます。

注： ***フレーミングエラーは SPI モードではサポートされていません。***

受信時の設定可能なストップビット

受信するストップビット数は、USART_CR の制御ビットを通じて設定でき、通常モードでは 1 または 2、スマートカードモードでは 0.5 または 1.5 にできます。

- **0.5 個のストップビット（スマートカードモードでの受信）**：0.5 個のストップビットでは、サンプリングは行われません。したがって、0.5 個のストップビットが選択されている場合、フレーミングエラーやブレークフレームは検出されません。
- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- 1.5 個のストップビット（スマートカードモード）

  スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバブロックを有効にし（USART_CR1 の RE=“1”）、ストップビットをチェックして、スマートカードがパリティエラーを検出したかどうかをテストする必要があります。

  パリティエラーが発生した場合、スマートカードはサンプリング時のデータ信号を強制的にローレベルにします（これは、フレーミングエラーとしてフラグされる NACK 信号です）。その後、1.5 個のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）を通して FE フラグがセットされます。ストップビット 1.5 個のサンプリングは、16 番目、17 番目、および 18 番目のサンプルで（ストップビットの開始から1 ボークロック周期後に）行われます。1.5 個のストップビットは、2 つの部分に分解できます。すなわち、何も起こらない 0.5

ボークロック周期と、途中でサンプリングが行われる通常の 1 ストップビット周期です（詳細は*セクション 48.5.16： 1977 ページのUSART レシーバタイムアウト*を参照）。

- 2 個のストップビット

  ストップビット 2 個のサンプリングは、最初のストップビットの 8 番目、9 番目、および 10 番目のサンプルで行われます。

  最初のストップビット中にフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。

  2 番目のストップビットでは、フレーミングエラーの検査は行われません。最初のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）がセットされます。

48.5.7 USART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、USART_BRR レジスタでプログラムされた値に設定されます。

Equation 1：標準 USART のボーレート（SPI モードを含む）（OVER8 = “0”または“1”）

16 倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

8 倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{2 \times \text{usart_ker_ckpres}}{\text{USARTDIV}}$$

Equation 2：スマートカード、LIN、および IrDA モードのボーレート（OVER8 = “0”）

ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

USARTDIV は、符号なしの固定小数点数であり、USART_BRR レジスタでコード化されます。

- OVER8 = “0”のとき、BRR = USARTDIV です。
- OVER8 = “1”のとき、
  - BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。
  - BRR[3] は、クリアされたままにする必要があります。
  - BRR[15:4] = USARTDIV[15:4]

注： ***ボーカウンタは、USART_BRR への書き込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。***

16 倍および 8 倍のオーバーサンプリングの場合、USARTDIV は 0d16 以上である必要があります。

USART_BRR レジスタの値から USARTDIV を得る方法

Example 1

usart_ker_ck_pres = 8 MHz で 9600 ボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 8 000 000/9600
  BRR = USARTDIV = 833d = 0341h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 8 000 000/9600
  USARTDIV = 1666,66 (1667d = 683h)
  BRR[3:0] = 3h >> 1 = 1h
  BRR = 0x681

Example 2

usart_ker_ck_pres = 48 MHz で 921.6 キロボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 48 000 000/921 600
  BRR = USARTDIV = 52d = 34h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 48 000 000/921 600
  USARTDIV = 104 (104d = 68h)
  BRR[3:0] = USARTDIV[3:0] >> 1 = 8h >> 1 = 4h
  BRR = 0x64

48.5.8 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバは、クロックシステムの合計偏差が USART レシーバの許容誤差未満の場合のみ、正しく動作します。

合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART receiver 許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

USART レシーバは、次の設定に応じて、*表 377* および *表 378* で指定された最大許容偏差まで、データを正しく受信できます。

- USART_CR1 レジスタの M ビットによって定義された 9、10 または 11 ビットのキャラクタ長
- USART_CR1 レジスタの OVER8 ビットによって定義された 8 倍または 16 倍のオーバーサンプリング
- USART_BRR レジスタのビット BRR[3:0] が 0000 に等しいかどうか。
- データのサンプリングに 1 ビットを使用するか 3 ビットを使用するか（USART_CR3 レジスタの ONEBIT ビットの値に依存）。

表 377. BRR [3:0] = 0000 のときの USART レシーバの許容誤差

M ビット	OVER8 ビット = “0”		OVER8 ビット = “1”	
	ONEBIT=“0”	ONEBIT=“1”	ONEBIT=“0”	ONEBIT=“1”
00	3.75%	4.375%	2.50%	3.75%
01	3.41%	3.97%	2.27%	3.41%
10	4.16	4.86	2.77	4.16

表 378. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差

M ビット	OVER8 ビット = “0”		OVER8 ビット = “1”	
	ONEBIT=“0”	ONEBIT=“1”	ONEBIT=“0”	ONEBIT=“1”
00	3.33%	3.88%	2%	3%
01	3.03%	3.53%	1.82%	2.73%
10	3.7	4.31	2.22	3.33

注： ***表 377** および **表 378** で指定されたデータは、M ビット = “00”のとき、受信フレームに正確に 10 ビット時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M = “01”のときには 11 ビット時間、または M = “10”のときには 9 ビット時間）。*

48.5.9 USART 自動ボーレート検出

USART は、1 キャラクタの受信に基づいて、USART_BRR レジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2 つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません。

- 16 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/16 までの範囲内です。
- 8 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/8 までの範囲内です。

自動ボーレート検出を有効にする前に、USART_CR2 レジスタの ABRMOD[1:0] フィールドによって自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、4 つのモードがあります。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

これらのモードは以下のとおりです。

- **モード 0** : “1”のビットで始まるキャラクタ。

  この場合、USART はスタートビットの時間を測定します（立ち下がりエッジから立ち上がりエッジまで）。
- **モード 1**：10xx ビットパターンで始まるキャラクタ。

  この場合、USART はスタートと最初のデータビットの時間を測定します。低速な信号スロープの場合の精度を高めるために、測定は立ち下がりエッジから立ち下がりエッジまでで行われます。
- **モード 2**: 0x7F キャラクタフレーム（LSB ファーストモードでは 0x7F キャラクタ、または MSB ファーストモードでは 0xFE）。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 6 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR6）。ビット 0 からビット 6 までが BRs でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。
- **モード 3** : 0x55 キャラクタフレーム。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 0 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR0）、最後にビット 6（BR6）の終了時に更新されます。ビット 0 は BRs でサンプリングされ、ビット 1 からビット 6 までは BR0 でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。並行して、RX ラインの中間遷移ごとに別のチェックが行われます。RX の遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット 0 で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USART_BRR レジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USART_CR2 レジスタの ABREN ビットをセットします。USART は、RX ラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USART_ISR レジスタの ABRF フラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR 値が破損して、ABRE エラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が 16 から 65536 までのクロック周期でなく（16 倍のオーバーサンプリング）、8 から 65536 までのクロック周期でない（8 倍のオーバーサンプリング））。

その後、ABRF フラグをリセットすることによって（“0”を書き込むことによって）、自動ボーレート検出を再開できます。

FIFO マネージメントが無効になっていて、自動ボーレートエラーが発生した場合、RXNE および FE ビットを通して ABRE フラグがセットされます。

FIFO マネージメントが有効になっていて、自動ボーレートエラーが発生した場合、RXFNE および FE ビットを通して ABRE フラグがセットされます。

FIFO モードが有効な場合、自動ボーレート検出は、RXFIFO の最初の場所のデータを使用して行うべきです。したがって、自動ボーレート検出を開始する前に、USART_ISR レジスタの RXFNE フラグをチェックして RXFIFO が空であることを確認してください。

注： ***自動ボーレート操作中に USART が無効化された場合（UE=“0”）、BRR 値が破損することがあります。***

48.5.10 USART マルチプロセッサ通信

USART のマルチプロセッサ通信が可能です（ネットワーク内で複数の USART を接続して）。たとえば、1 つの USART をマスタとして、その TX 出力を別の USART の RX 入力に接続することができます。一方、他の USART は、それぞれの TX 出力の論理和をとってマスタの RX 入力に接続して、スレーブとなります。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する USART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USART_CR1 レジスタの MME ビットをセットする必要があります。

注： ***FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 usart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。***

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- USART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、USART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

USART は、USART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

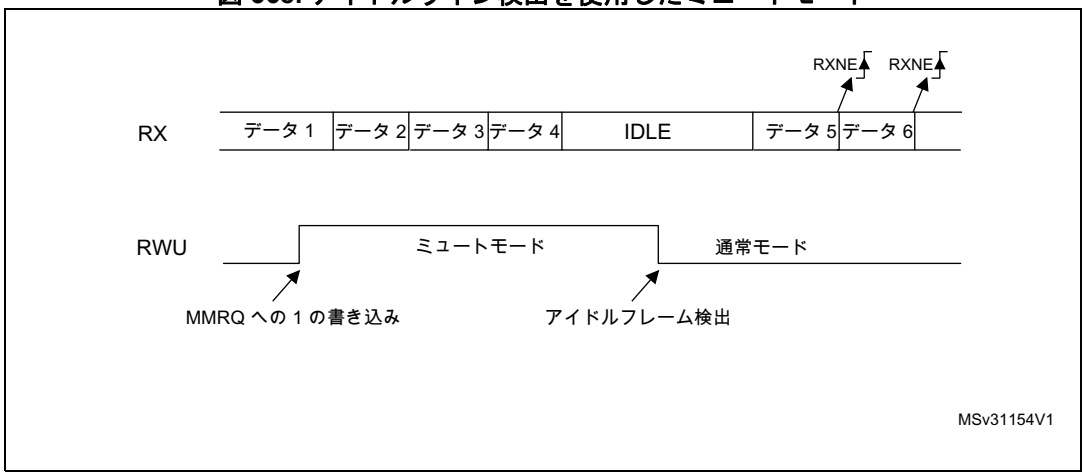
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=“0”）

MMRQ ビットに“1”が書き込まれ、RWU が自動的にセットされたときには、USART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、USART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を *図 568* に示します。

図 568. アイドルライン検出を使用したミュートモード



注： ***IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。***

ラインが IDLE のときに USART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビットアドレスマーク検出（WAKE="1"）

このモードでは、MSB が"1"の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、USART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。***

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。FIFO 管理が有効になっている場合、ミュートモードに入る前に、RXFIFO に少なくとも 1 つの空き場所があることをソフトウェアによって確認するべきです。

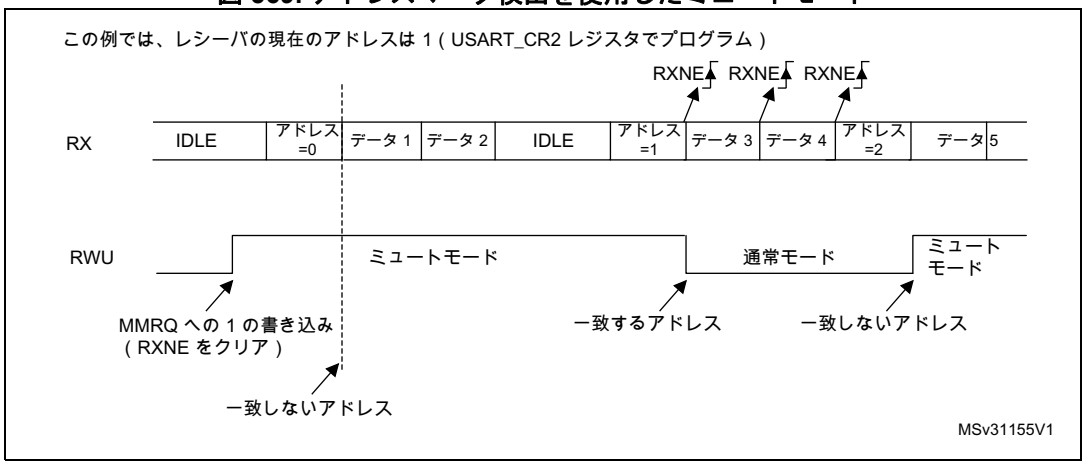
MMRQ ビットに 1 が書き込まれたときにも、USART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： ***FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ がセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。***

アドレスマーク検出を使用したミュートモードの動作例を *図 569* に示します。

図 569. アドレスマーク検出を使用したミュートモード



48.5.11 USART Modbus 通信

USART は、Modbus/RTU および Modbus/ASCII プロトコルの実装に対する基本的サポートを備えています。Modbus/RTU は、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USART は、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1 つのブロックの終了は 2 キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割り込みは、USART_CR2 レジスタの RTOEN ビットと、USART_CR1 レジスタの RTOIE を通じて有効にする必要があります。2 キャラクタ時間に対応する値（たとえば、22 x ビット時間）を RTO レジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割り込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USART は、キャラクター致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドで LF ASCII コードをプログラムし、キャラクター致割り込みを有効にすることによって（CMIE=“1”）、LF が受信されたときにソフトウェアに通知し、ソフトウェアは DMA バッファの CR/LF をチェックできます。

48.5.12 USART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、USART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な USART フレームフォーマットを *表 379* に示します。

表 379. USART フレームのフォーマット

M ビット	PCE ビット	USART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例 : SB : スタートビット、STB : ストップビット、PB : パリティビット。データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合（USART_CR1 の PS ビット =“0”）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合（USART_CR1 の PS ビット =“1”）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、USART_ISR レジスタの PE フラグがセットされ、USART_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、USART_ICR レジスタの PECF に 1 を書き込むことによってクリアされます。

送信中のパリティ生成

USART_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=“0”）は偶数個の「1」、奇数パリティが選択された場合（PS=“1”）は奇数個の「1」）。

48.5.13 USART LIN（Local Interconnection Network）モード

このセクションは、LIN モードがサポートされるときにのみ適用されます。*セクション 48.4 : 1948 ページのUSART の実装*を参照してください。

LIN モードを選択するには、USART_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの CLKEN
- USART_CR3 レジスタの STOP[1:0]、SCEN、HDSEL、および IREN

LIN 送信

LIN マスタ送信の場合、*セクション 48.5.4* に記述されている手順を適用する必要があります。通常の USART 送信と同じですが、次のような違いがあります。

- 8 ビットのワード長を設定するには M ビットをクリアします。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13 個の “0” ビットがブレークキャラクタとして送信されます。その後、値 “1” の 2 ビットが送信され、次の START 検出が可能になります。

LIN 受信

LIN モードが有効になると、ブレーク検出回路が有効になります。この検出は、通常の USART レシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生のたびにブレークが検出できます。

レシーバが有効になると（USART_CR1 レジスタの RE=“1”）、RX 入力の START 信号を探します。スタートビットの検出方法は、ブレークキャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます（8、9、および 10 番目のサンプル）。10 個（USART_CR2 レジスタの LBDL=“0”）または 11 個（USART_CR2 レジスタの LBDL=“1”）の連続したビットが “0” として検出され、その後にデリミタキャラクタが続く場合、USART_ISR レジスタの LBDF フラグがセットされます。LBDIE ビットが“1”の場合、割り込みが生成されます。ブレークを確認する前に、RX ラインがハイレベルに戻ったことを知らせるデリミタが検査されます。

この 10 または 11 が発生する前に“1”がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LIN モードが無効にされた場合（LINEN=“0”）、レシーバは、ブレーク検出を考慮することなく、通常の USART として機能し続けます。

LIN モードが有効にされた場合（LINEN=“1”）、フレーミングエラーが発生（つまり、ブレークフレームの場合と同様に、“0” の位置でストップビットを検出）すると、レシーバは停止し、ブレーク検出回路が “1” （ブレークワードが完全でなかった場合）またはデリミタキャラクタ（ブレークが検出された場合）を受信するまで停止状態を維持します。

ブレーク検出回路ステートマシンの動作とブレークフラグを *図 570： 1972 ページのLIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）* に示します。

ブレークフレームの例を *図 571： 1973 ページのLIN モードでのブレーク検出とフレーミングエラー検出* に示します。

図 570. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）

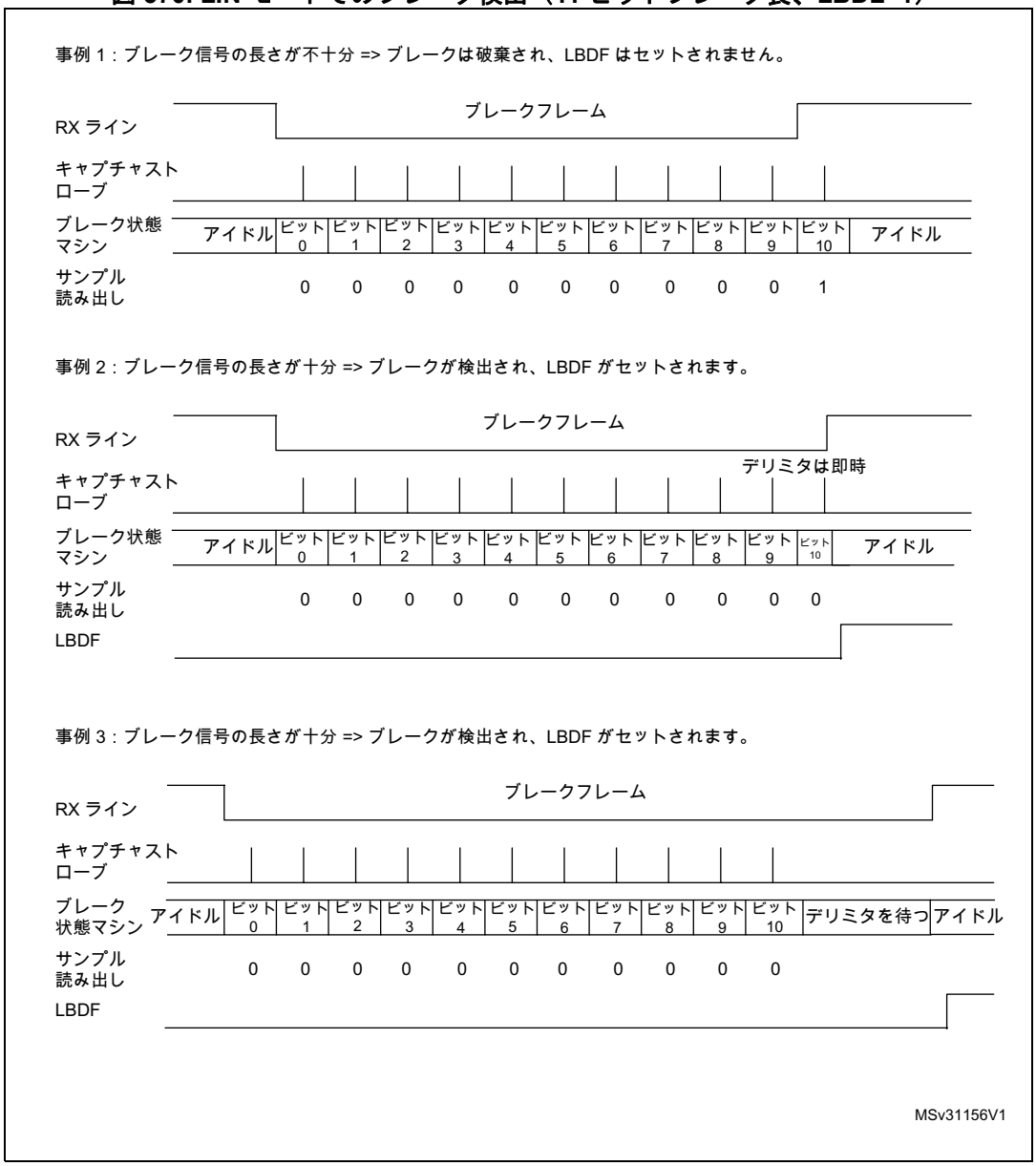
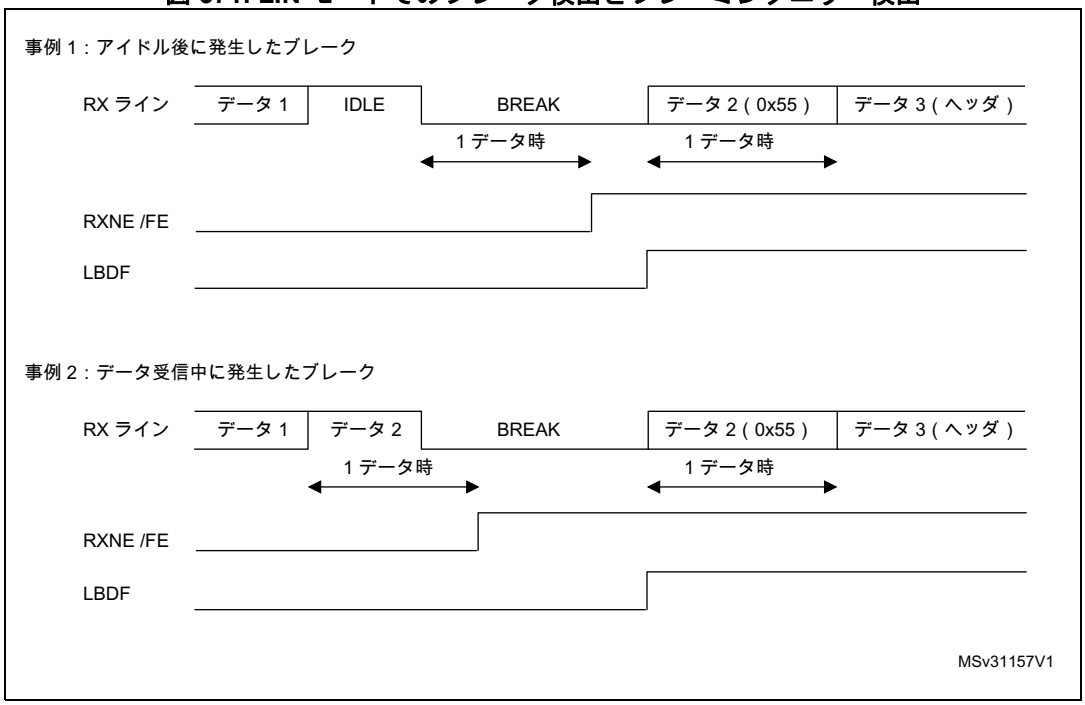


図 571. LIN モードでのブレーク検出とフレーミングエラー検出



48.5.14 USART 同期モード

マスタモード

同期マスタモードを選択するには、USART_CR2 レジスタの CLKEN ビットを“1”にプログラムします。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をマスタモードで制御できます。SCLK ピンは USART トランスミッタクロックの出力です。スタートビットとストップビットの処理中には、SCLK ピンにクロックパルスは送信されません。USART_CR2 レジスタの LBCL ビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USART_CR2 レジスタの CPOL ビットは、クロック極性を選択するために使用され、USART_CR2 レジスタの CPHA ビットは、外部クロックの位相を選択するために使用されます（*図 572*、*図 573*、および*図 574* を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、外部 SCLK クロックは起動されません。

同期マスタモードでは、USART トランスミッタは非同期モードの場合とまったく同じように動作します。しかし、CPOL と CPHA に基づいて SCLK が TX と同期するので、TX 上のデータが同期します。

同期マスタモードでは、USART レシーバは非同期モードの場合とは異なる動作をします。RE が“1”にセットされた場合、データはオーバーサンプリングなしで、SCLK（CPOL と CPHA に応じて立上りまたは立下りエッジ）でサンプリングされます。ボーレートに応じた所定（1/16 ビット時間）のセットアップ時間とホールド時間を守る必要があります。

注： *マスタモードでは、SCLK ピンは TX ピンと連携して動作します。したがって、クロックが供給されるのは、トランスミッタが有効であり（TE="1"）、データが送信中（USART_TDR データレジスタ への書き込み）の場合に限られます。つまり、データ送信を行わずに同期データを受信することはできません。*

図 572. USART の同期マスタ送信の例

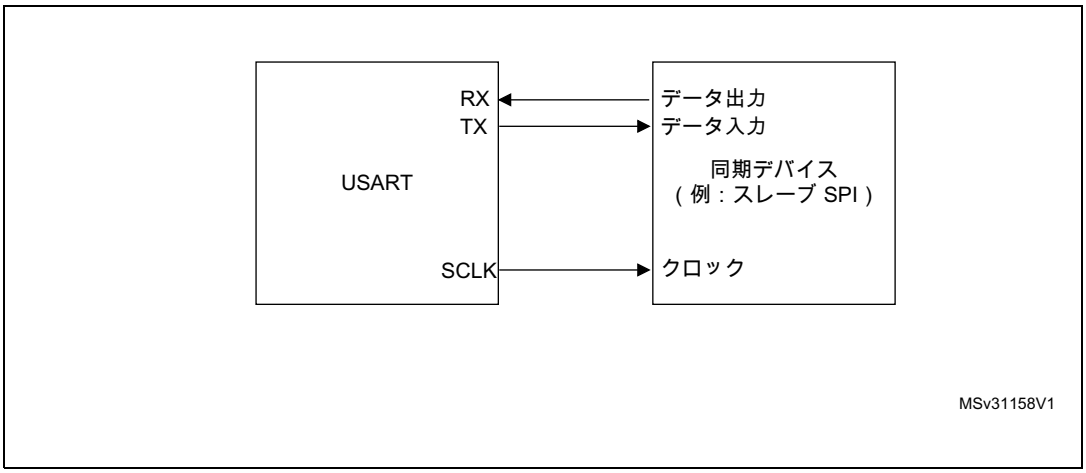


図 573. 同期マスタモードでの USART データクロックタイミング図（M ビット ="00"）

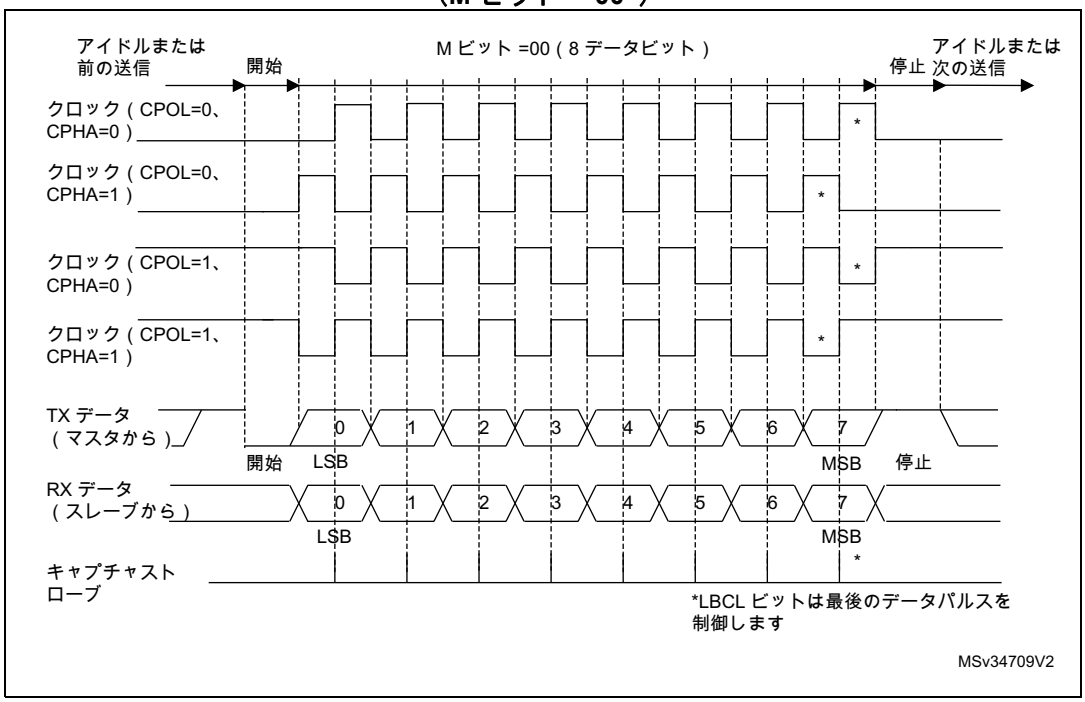
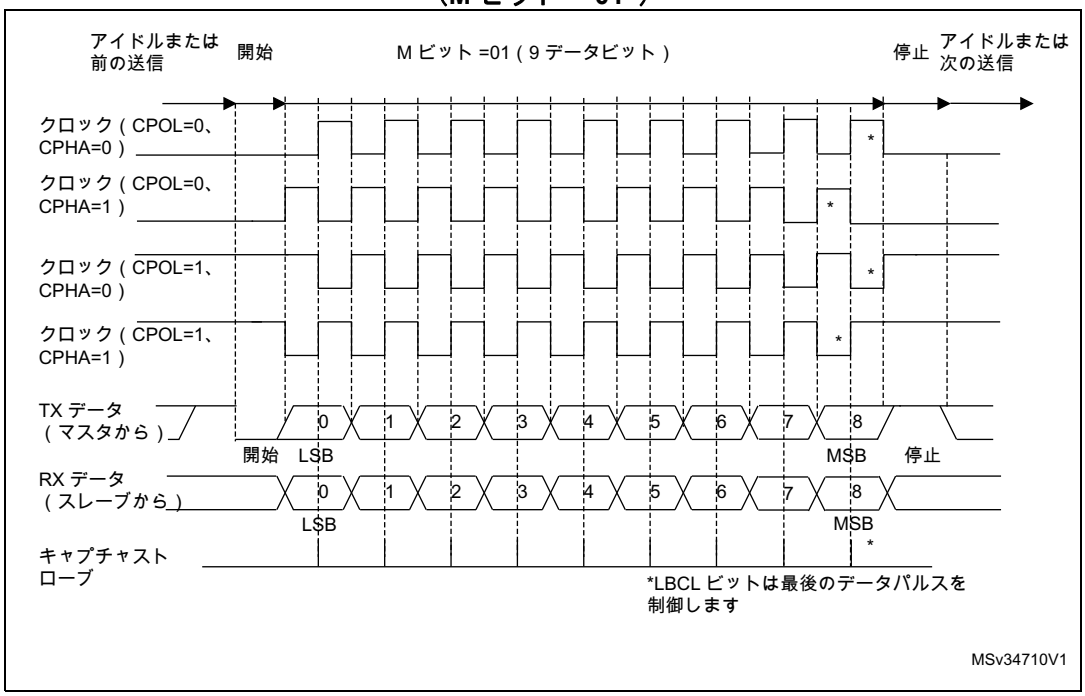


図 574. 同期マスタモードでの USART データクロックタイミング図（M ビット =“01”）



スレーブモード

同期スレーブモードを選択するには、USART_CR2 レジスタの SLVEN ビットを“1”にプログラムします。同期スレーブモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をスレーブモードで制御できます。SCLK ピンは、スレーブモードの USART の入力です。

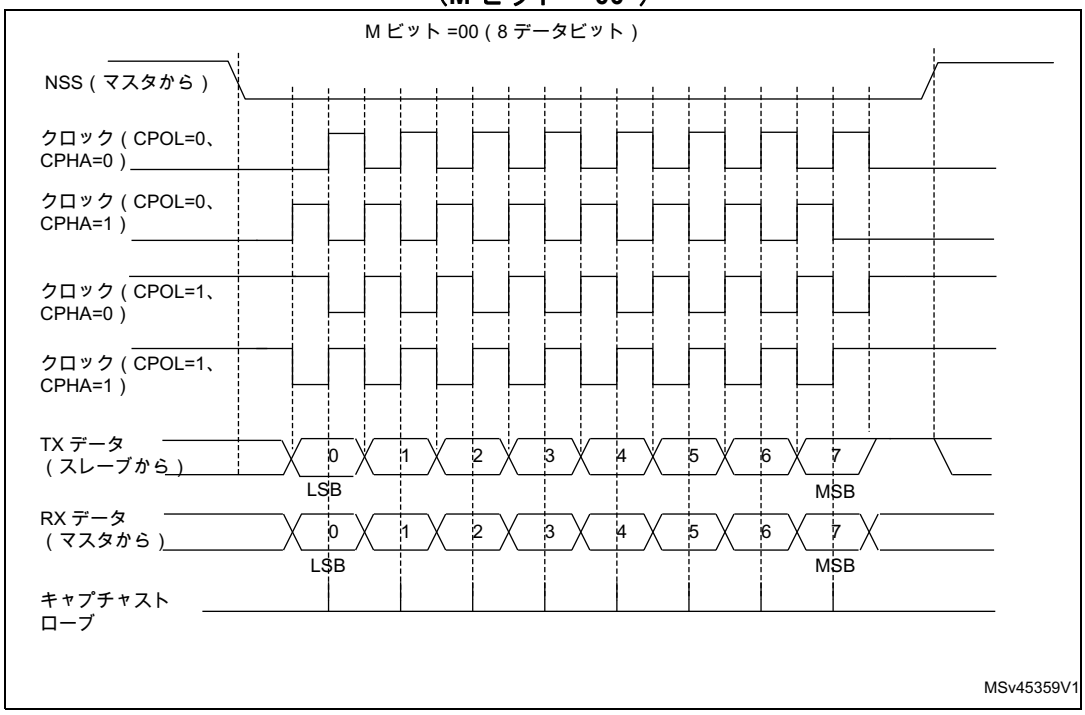
注： ***ペリフェラルが SPI のスレーブモードで使用されるとき、ペリフェラルのクロックソースの周波数（usart_ker_ck_pres）は CK 入力周波数の 3 倍より大きくなければなりません。***

USART_CR2 レジスタの CPOL ビットと CPHA ビットは、それぞれクロック極性と外部クロックの位相を選択するために使用されます（*図 575* を参照）。

アンダーランエラーフラグはスレーブ送信モードで使用できます。ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。

スレーブはハードウェアおよびソフトウェアの NSS 管理をサポートしています。

図 575. 同期スレーブモードでの USART データクロックタイミング図（M ビット =“00”）



スレーブ選択（NSS）ピンの管理

ハードウェアまたはソフトウェアのスレーブ選択管理は、USART_CR2 レジスタの DIS_NSS ビットを使用して設定することができます。

- ソフトウェア NSS 管理（DIS_NSS = “1”）

  SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

  外部NSS ピンは他のアプリケーションで使用できます。

- ハードウェア NSS 管理（DIS_NSS = “0”）

  SPI スレーブ選択は NSS 入力ピンに依存します。NSS がローレベルのときにスレーブが選択され、NSS がハイレベルのときには選択されません。

注： ***USART が無効にされたときには（UE=“0”）、クロックパルスが正常に機能するように、LBCL（SPI マスタモードでのみ使用）、CPOL、および CPHA ビットを選択する必要があります。***

SPI スレーブモードでは、マスタ通信を開始する前に（またはクロックが安定しているときのフレーム間で）、USART を有効にする必要があります。そうしないと、マスタがフレームの中央に位置しているときに USART スレーブが有効化されると、スレーブはマスタとの同期がとれなくなります。スレーブのデータレジスタは、通信クロックの最初のエッジまたは現在の通信の終了より前に、準備ができている必要があります。そうしないと、SPI スレーブはゼロを送信します。

SPI スレーブのアンダーランエラー

アンダーランエラーが起きると、USART_ISR レジスタの UDR フラグがセットされ、ソフトウェアによってアンダーランエラーフラグがクリアされるまで SPI スレーブは最後のデータを送信し続けます。

アンダーランフラグはフレームの開始時にセットされます。USART_CR3 レジスタの EIE ビットがセットされている場合、アンダーランエラー割り込みがトリガされます。

アンダーランエラーフラグは、USART_ICR レジスタのビット UDRCF をセットすることによってクリアされます。

アンダーランエラーの場合、TDR レジスタに書き込むことは可能です。アンダーランエラーをクリアすると、新しいデータが送信できるようになります。

アンダーランエラーが起きて、TDR に書き込まれる新しいデータがない場合は、フレームの終りに TC フラグがセットされます。

注： ***データが USART_TDR に書き込まれた時点が SCLK の最初の送信エッジに近すぎると、アンダーランエラーが起きることがあります。このアンダーランエラーを避けるために、USART_TDR への書き込みは、最初の SCLK エッジより usart_ker_ck の 3 サイクル前に行うべきです。***

48.5.15 USART 単線半二重通信

単線半二重モードを選択するには、USART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN および IREN ビット

USART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の USART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

48.5.16 USART レシーバタイムアウト

レシーバタイムアウト機能を有効にするには、USART_CR2 制御レジスタの RTOEN ビットをセットします。

タイムアウト経過時間は USART_RTOR レジスタの RTO ビットフィールドを使用してプログラムします。

レシーバタイムアウトカウンタは次の時点からカウントを開始します。

- STOP = “00” または STOP = “11” の場合、ストップビットの最後から
- STOP = “10” の場合、2 番目のストップビットの最後から。
- STOP = “01” の場合、ストップビットの最初から。

タイムアウト経過時間が過ぎると、USART_ISR レジスタの RTOF フラグがセットされます。USART_CR1 レジスタの RTOIE ビットがセットされている場合、タイムアウトが生成されます。

48.5.17 USART スマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。*セクション 48.4 : 1948 ページのUSART の実装*を参照してください。

スマートカードモードを選択するには、USART_CR3 レジスタの SCEN ビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL および IREN ビット

スマートカードにクロックを供給するために CLKEN ビットもセットすることができます。

スマートカードインタフェースは、ISO 7816-3 標準で定義された非同期スマートカードプロトコルをサポートするように設計されています。T="0"（キャラクタモード）と T="1"（ブロックモード）の両方がサポートされます。

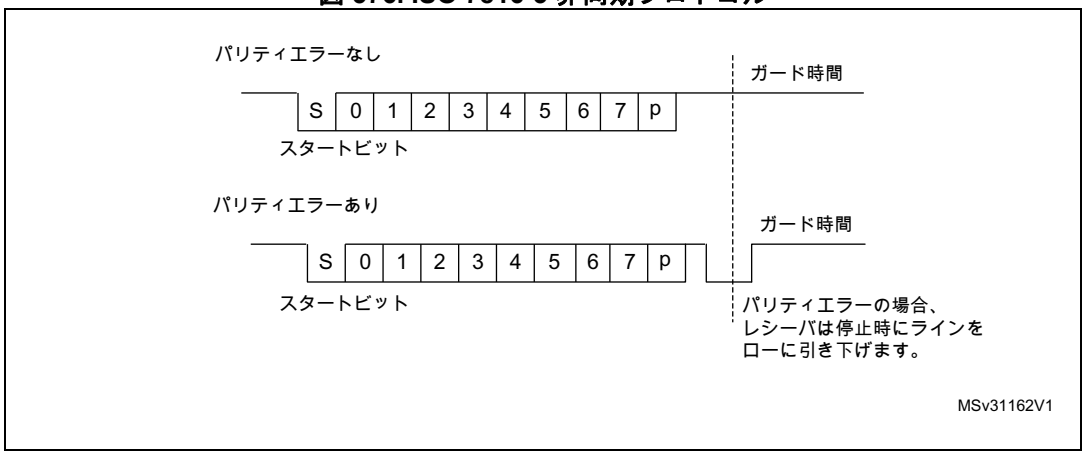
USART は次のように設定してください。

- 8 ビット＋パリティ : USART_CR1 レジスタの M="1"および PCE="1"
- データ送受信ではストップビット 1.5 個:USART_CR2 レジスタの STOP="11"。受信にはストップビット 0.5 個を選択することも可能です。

T="0"（キャラクタ）モードでは、パリティエラーはガードタイム中の各キャラクタの終わりに示されます。

図 576 に、パリティエラーの有無によるデータラインの状況の変化の例を示します。

図 576. ISO 7816-3 非同期プロトコル



スマートカードと接続されると、USART の TX 出力は、やはりスマートカードによって駆動される双方向ラインを駆動します。TX ピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

- 送信シフトレジスタからのデータの送信は、少なくとも 1/2 ボークロックの遅れが保証されます。通常動作では、フルの送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された 1/2 ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって（NACK）、この条件を USART に知らせます。この NACK 信号（1 ボークロックの間、送信ラインをローに引き下げ）は、1.5 個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USART は、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNT ビットフィールドでプログラムされます。プログ

ラムされた再試行回数後も USART が NACK を受信し続けた場合は、送信を停止して、エラーをフレーミングエラーとして通知します。TXE ビット（FIFO モードが有効な場合は TXFNF ビット）は、USART_RQR レジスタの TXFRQ ビットを使用してセットできます。

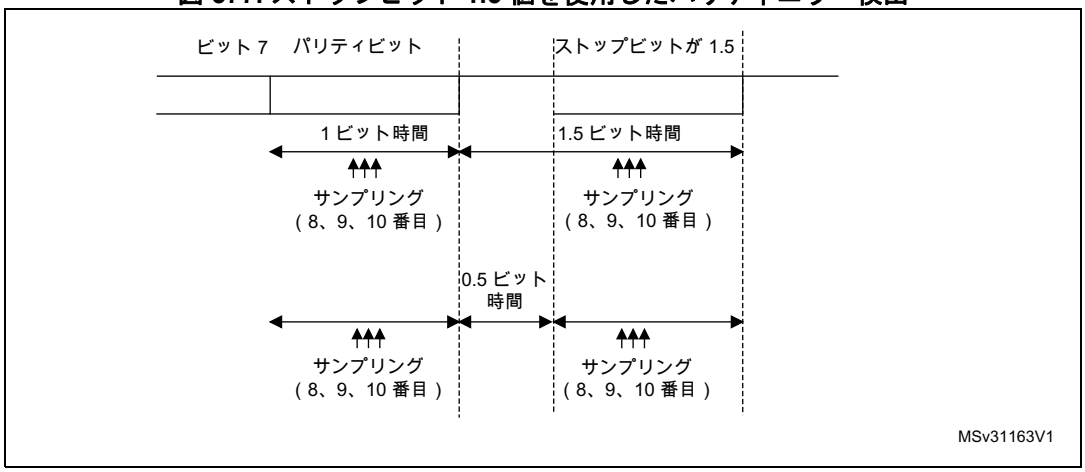
- 送信時のスマートカード自動再試行：USART による NACK の検出と反復キャラクタのスタートビットの間に 2.5 ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちに TC ビットがセットされます（ガードタイムはありません）。ソフトウェアで再び繰り返したい場合は、規格によって指定されている 2 ボー周期以上を確保する必要があります。
- 1.5　個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後 1 ボークロック周期間、送信ラインがローに引き下げられます。これは、USART に送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK 制御ビットがセットされている場合、パリティエラーはレシーバによって “NACK” され、そうでない場合、NACK は送信されません（T=1 モードで使用されます）。受信したキャラクタにエラーがあった場合、RXNE（FIFO モードが有効な場合は RXFNE）／受信 DMA リクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNT ビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USART は NACK の送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行：USART がカードを NACK したが、カードがキャラクタを繰り返さなかった場合、BUSY フラグはセットされたままです。
- 送信時、USART は 2 つの連続するキャラクタの間にガードタイム（ガードタイムレジスタでプログラム）を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的の CGT（7816-3 仕様で定義されている Character Guard Time）から 12（1 キャラクタの時間）を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。TCBGT フラグは、ガード時間完了を待たずにデータ転送の終わりを検出するために使用できます。このフラグは、フレーム送信終了後およびカードから NACK を受信しなかった場合にのみセットされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの NACK によってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、この NACK をスタートビットとして検出しません。ISO プロトコルによれば、受信される NACK の期間は 1 または 2 ボークロック周期です。
- レシーバ側では、パリティエラーが検出されて NACK が送信された場合、レシーバはこの NACK をスタートビットとして検出しません。

注： ***スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ 0x00 は、ブレークではなくデータとして処理されます。***

TE ビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISO プロトコルでは定義されていません。

図 577 に、USART による NACK 信号のサンプリング方法を示します。この例では、USART はデータを送信中であり、ストップビットが 1.5 個組み込まれています。データと NACK 信号の整合性を検査するために、USART のレシーバ部が有効にされます。

図 577. ストップビット 1.5 個を使用したパリティエラー検出



USART は、SCLK 出力を通じてスマートカードにクロックを供給できます。スマートカードモードでは、SCLK は通信に関係せず、5 ビットのプリスケーラを通じて単に内部のペリフェラル入力クロックから取得されます。この分周比は、USART_GTPR レジスタで設定されます。SCLK の周波数は usart_ker_ck_pres/2 から usart_ker_ck_pres/62 の間でプログラミングできます。ここで、usart_ker_ck_pres は、プログラムされたプリスケーラで分周されたペリフェラル入力クロックです。

ブロックモード（T=“1”）

T=“1”（ブロック）モードでは、パリティエラー送信は、UART_CR3 レジスタの NACK ビットをクリアすることによって無効化されます。

ブロックモードでスマートカードからの読み出しをリクエストするときには、ソフトウェアは RTOR レジスタを BWT（ブロックウェイトタイム）- 11 の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、タイムアウト割り込みが生成されます。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE/RXFNE 割り込みによって通知されます。

注： ***ブロックモードのスマートカードからの読み出しに USART を DMA モードで使用するときでも、RXNE/RXFNE 割り込みを有効にする必要があります。並行して、DMA は最初の受信バイトの後でのみ有効にする必要があります。***

2 つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後（RXNE/RXFNE 割り込み）、RTO レジスタを CWT（キャラクタウェイトタイム -11）の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT 未満の時間内に新しいキャラクタを送信しなかった場合、USART は RTOF フラグと割り込み（RTOIE ビットがセットされているとき）によって、これをソフトウェアに通知します。

注： ***スマートカードプロトコルの定義にあるように、BWT/CWT 値は最後のキャラクタの開始（スタートビット）から定義されるべきです。RTO レジスタは、最後のキャラクタ自体の長さを考慮して、それぞれ BWT -11 または CWT -11 にプログラムする必要があります。***

ブロック長カウンタは、USART が受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USART の送信時にリセットされます。ブロックの長さは、スマートカードによってブロックの 3 番目のバイト（プロローグフィールド）で伝えられます。この値を USART_RTOR レジスタの BLEN フィールドでプログラムする必要があります。DMA モードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値（0x0）にプログラムする必要があります。この値では、4 番目の受信キャラクタの後に割り込みが生成されます。ソフトウェアは LEN フィールド（3 番目のバイト）を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割り込み駆動受信モードでは、ブロックの長さはソフトウェアによって、または BLEN 値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLEN の最大値（0xFF）をプログラムすることができます。実際の値は、3 番目のキャラクタの受信後にプログラムされます。

ブロックが LRC 水平冗長検査（1 エピローグバイト）を使用している場合は、BLEN=LEN です。ブロックが CRC メカニズム（2 エピローグバイト）を使用している場合は、BLEN=LEN+1 をプログラムする必要があります。合計ブロック長（プロローグ、エピローグ、および情報フィールドを含む）は、BLEN+4 に等しくなります。ブロックの終わりは EOBF フラグと割り込み（EOBIE ビットがセットされているとき）によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりは RTO 割り込みによって通知されます（キャラクタウェイトタイムオーバーフロー）。

注： ***エラーチェックコード（LRC/CRC）は、ソフトウェアによって計算／確認されなければなりません。***

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの 2 つコンベンションを定義しています。

ダイレクトコンベンションは、LSB ファースト、論理ビットの値 1 がラインの H 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=“0”、DATAINV=“0”（デフォルト値）をプログラムする必要があります。

インバースコンベンションは、MSB ファースト、論理ビットの値 1 が単線の L 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=“1”、DATAINV=“1” をプログラムする必要があります。

注： ***論理データ値が反転されると（0=H、1=L）、パリティビットも同じように反転されます。***

カードのコンベンションを認識するために、カードは最初のキャラクタ TS を ATR（Answer To Reset）フレームの最初のキャラクタとして送信します。TS には、LHHL LLL LLH と LHHL HHH LLH の 2 つのパターンが使えます。

- (H) LHHL LLL LLH は、インバースコンベンションをセットアップします。状態 L が値 1 にエンコードされ、モーメント 2 は最上位ビットを含みます（MSB ファースト）。インバースコンベンションによってデコードされると、送受信されたバイトは 3F に等しくなります。
- (H) LHHL HHH LLH は、ダイレクトコンベンションをセットアップします。状態 H が値 1 にエンコードされ、モーメント 2 は最下位ビットを含みます（LSB ファースト）。ダイレクトコンベンションによってデコードされると、送受信されたバイトは 3B に等しくなります。

キャラクタパリティは、2 から 10 までの 9 個のモーメントに 1 にセットされた偶数個のビットがあったときに正しいとみなされます。

USART はカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USART がダイレクトコンベンション（デフォルト）で設定され、カードがインバースコンベンションで応答した場合、TS = LHHL LLL LLH になり、USART が受信したキャラクタは 03 になり、パリティは奇数です。

したがって、TS パターン認識には 2 つの方式を使用できます。

方式 1

USART は、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TS パターンの受信によってパリティエラー割り込みと、カードに対するエラー信号が生成されます。

- パリティエラー割り込みは、カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USART をインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割り込みに応答して、ソフトウェアは USART を再プログラムし、カードに対して新しいリセットコマンドを生成してから、TS を再び待つこともできます。

方式 2

USART は 9 ビット／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように 2 つの TS パターンのいずれかを受信します。

(H) LHHL LLL LLH = 0x103 -> インバースコンベンションを選択

(H) LHHL LLL LLH = 0x103 -> ダイレクトコンベンションを選択

ソフトウェアは受信されたキャラクタをこの 2 つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えて USART をプログラムします。

2 つのうちのどちらも認識されなかった場合、ネゴシエーションを再開始するためにカードリセットが生成されます。

48.5.18 USART IrDA SIR ENDEC ブロック

このセクションは、IrDA モードがサポートされるときにのみ適用されます。*セクション 48.4：1948 ページのUSART の実装*を参照してください。

IrDA モードを選択するには、USART_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN、STOP、および CLKEN ビット
- USART_CR3 レジスタの SCEN および HDSEL ビット

IrDA SIR 物理層は、ロジック 0 を赤外光パルスとして表現する RZI（Return to Zero, Inverted）変調方式の使用を指定します（*図 578* を参照）。

SIR 送信エンコーダは、USART からの NRZ（Non Return to Zero）送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線 LED に送信されます。SIR ENDEC の場合、USART は最大 115.2 Kbps のビットレートしかサポートしません。通常モードでは、送信されるパルス幅は、ビット周期の 3/16 と指定されます。

SIR 受信デコーダは、赤外線検出回路からの RZ（Return to Zero）ビットストリームを復調し、受信した NRZ シリアルビットストリームを USART に出力します。デコーダの入力は、アイドル状態のノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDA は半二重通信プロトコルです。トランスミッタがビジーである場合（USART が IrDA エンコーダにデータを送信しているとき）、IrDA デコーダは IrDA 受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USART がデコードされたデータを受信している）場合、IrDA は USART から IrDA への TX 上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。

- “0”はハイパルスとして送信され、“1”は“0”として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の 3/16 と規定されます（*図 579* を参照）。
- SIR デコーダは、IrDA 準拠の受信信号を USART 用のビットストリームに変換します。
- SIR 受信ロジックは、ハイ状態を論理値 1 とみなし、ローパルスを論理値 0 とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR 出力は、アイドル時にロー状態になります。
- IrDA 仕様では、1.41 us より大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ端のグリッチ検出回路は、PSC 2 周期（PSC は USART_GTPR でプログラムされたプリスケーラ値）より小さな幅のパルスをフィルタします。PSC 1 周期より小さな幅のパルスは常に拒否されますが、1 周期以上2 周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2 周期より大きな幅のパルスは、パルスとして受け入れられます。PSC=“0”のとき、IrDA エンコーダ/デコーダは機能しません。
- レシーバは、低電力トランスミッタと通信できます。
- IrDA モードでは、USART_CR2 レジスタのストップビットを“1 ストップビット”に設定する必要があります。

IrDA 低電力モード

- トランスミッタ

  低電力モードでは、パルス幅はビット周期の 3/16 に維持されません。代わりに、パルス幅は低電力ボーレート（最小で 1.42 MHz）の 3 倍となります。一般に、この値は 1.8432 MHz（1.42 MHz < PSC < 2.12 MHz）です。低電力モードのプログラム可能な分周器は、この値を得るためにシステムクロックを分周します。

- レシーバ

  低電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USART は 1/PSC よりも短いパルスを破棄する必要があります。有効なローレベルは、その期間が IrDA 低電力ボークロック（USART_GTPR の PSC 値）の 2 周期分を超える場合にのみ受け入れられます。

注： ***PSC 2 周期未満 1 周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。***

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA 物理層仕様では、送信と受信の間に最小 10 ms の遅延を指定しています（IrDA は半二重プロトコルです）。

図 578. IrDA SIR ENDEC ブロック図

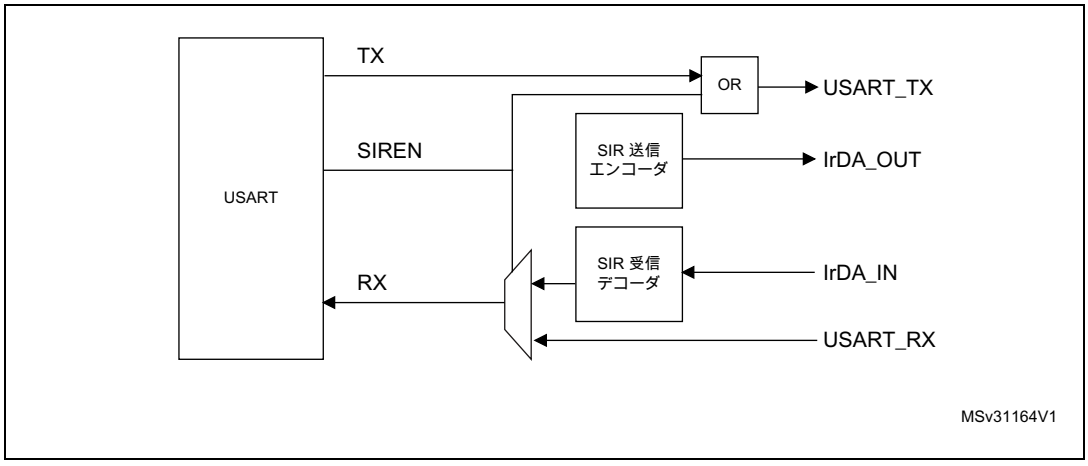
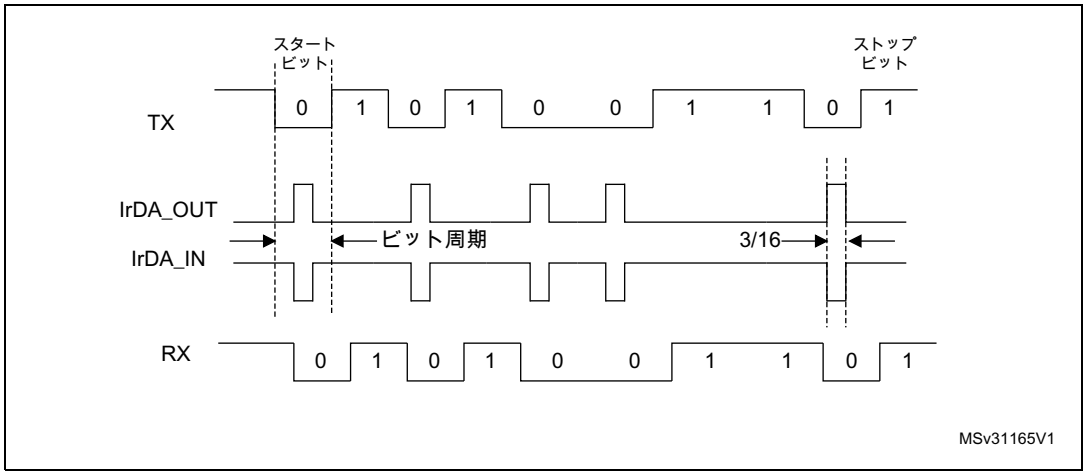


図 579. IrDA データ変調（3/16）- 通常モード



48.5.19 USART および DMA を使用した連続通信

USART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： ***DMA モードがサポートされるかどうかについては、セクション 48.4：1948 ページのUSART の実装を参照してください。DMA がサポートされない場合は、セクション 48.5.6 の説明に従って USART を使用してください。FIFO が無効の場合、連続通信を行うには、USART_ISR レジスタの TXE/ RXNE フラグをクリアします。***

DMA を使用した送信

DMA モードでの送信を有効にするには、USART_CR3 レジスタの DMAT ビットをセットします。TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされるたびに、データは、DMA ペリフェラル（*セクション 15：ダイレクトメモリアクセスコントローラ（DMA1、DMA2）*および*セクション 16：基本ダイレクトメモリアクセスコントローラ（BDMA）*を参照）を使用して設定された SRAM 領域から USART_TDR レジスタにロードされます。DMA チャネルを USART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

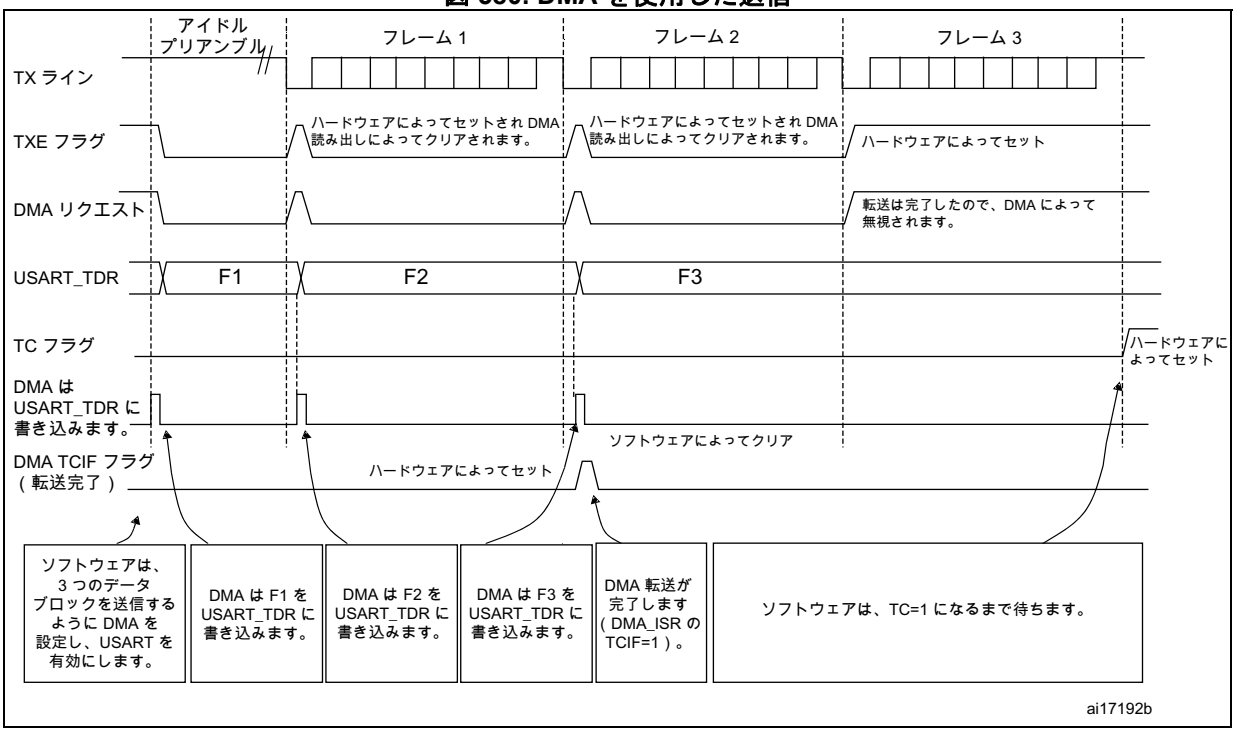
1. DMA 制御レジスタに USART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から USART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割り込み生成を設定します。
6. USART_ICR レジスタの TCCF ビットをセットすることによって、USART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して USART 通信の完了を確認することができます。これは、

USART を無効にする前に、またはペリフェラルクロックが無効のときシステムが低電力モードに入る前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=“1”になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 580. DMA を使用した送信



注： ***FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = “1”）によってトリガされます。***

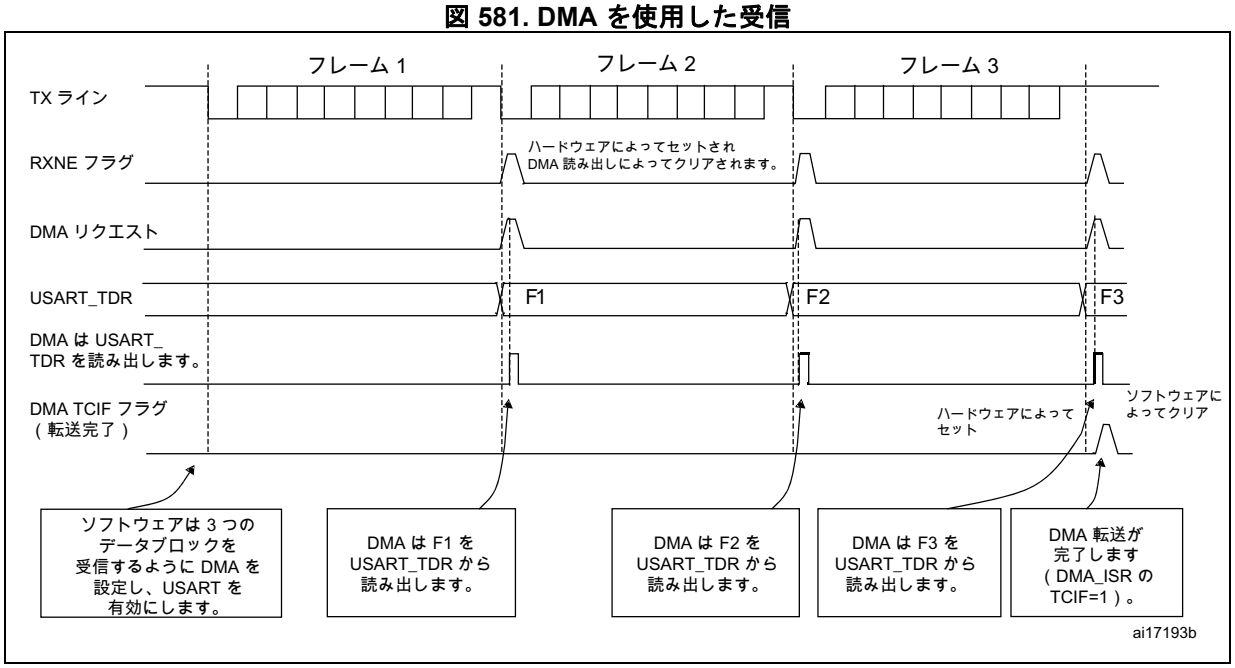
DMA を使用した受信

DMA モードでの受信を有効にするには、USART_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、USART_RDR レジスタから、DMA ペリフェラル（*セクション 15：ダイレクトメモリアクセスコントローラ（DMA1、DMA2）*および*セクション 16：基本ダイレクトメモリアクセスコントローラ（BDMA）*を参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを USART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに USART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、USART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

図 581. DMA を使用した受信



注： ***FIFO 管理が有効になっているとき、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = “1”）によってトリガされます。***

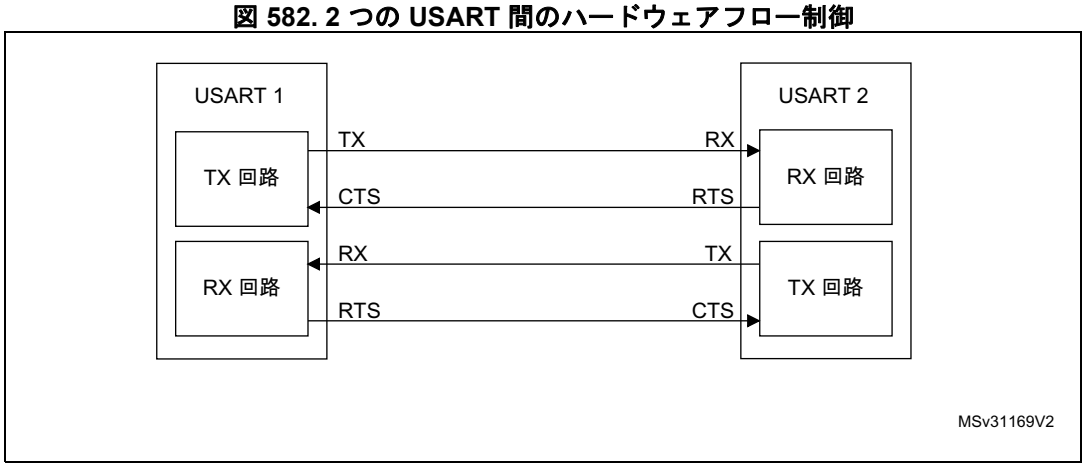
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り込み有効ビット（USART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

48.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。*図 582* に、このモードで 2 つのデバイスを接続する方法を示します。

図 582. 2 つの USART 間のハードウェアフロー制御

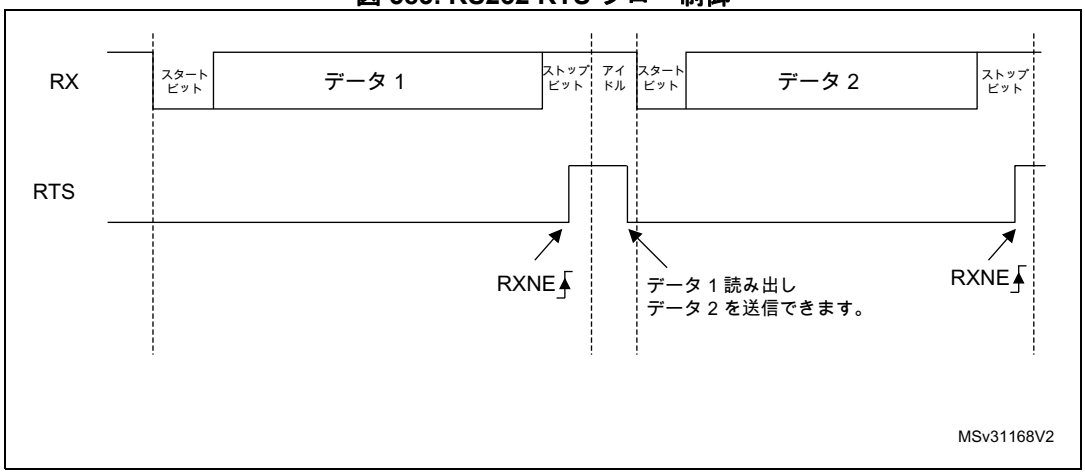


RS232 RTS と CTS のフロー制御は、USART_CR3 レジスタの RTSE ビットと CTSE ビットに“1”を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=“1”）、USART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます（ローレベル接続）。受信レジスタがフルになると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。*図 583* に、RTS フロー制御が有効な場合の通信例を示します。

図 583. RS232 RTS フロー制御



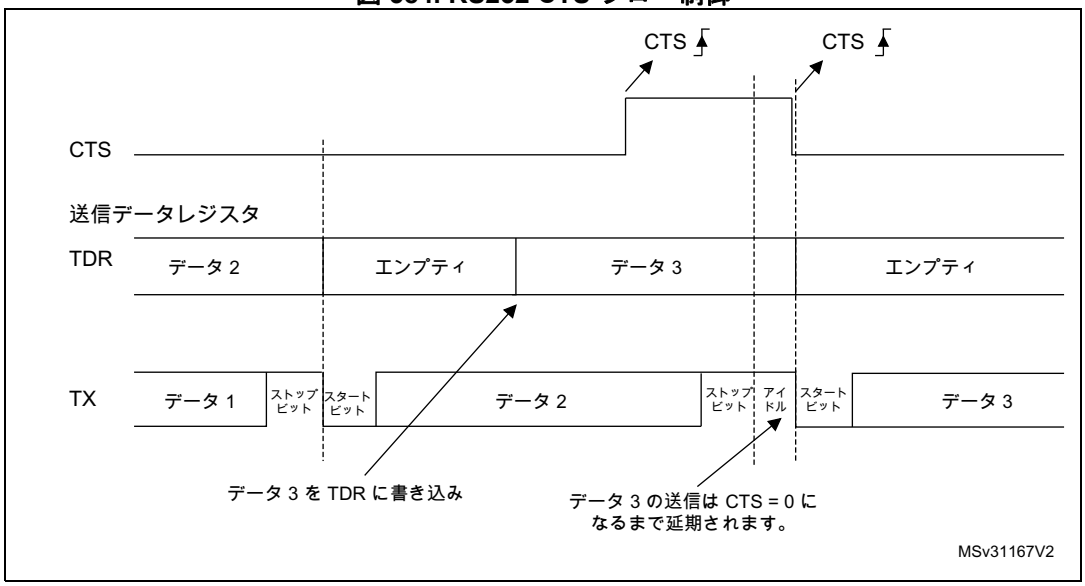
注： ***FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、nRTS がネゲートされます。***

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=“1”）、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE/TXFE=“0”の場合）。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=“1”の場合、nCTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。USART_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。*図 584* に、CTS フロー制御が有効な場合の通信例を示します。

図 584. RS232 CTS フロー制御



注： ***正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 USART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。***

RS485 ドライバ有効

ドライバ有効機能を有効にするには、USART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。USART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。USART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、USART_CR3 制御レジスタの DEP ビットを使用して設定できます。

USART では、DEAT および DEDT はサンプル時間単位（オーバーサンプリングレートに応じて 1/8 または 1/16 ビット時間）で表されます。

48.5.21 USART 低電力管理

USART には、高度な低電力モード機能があり、usart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

USART は、UESM ビットがセットされているとき、MCU を低電力モードからウェイクアップできます。

Usart_pclk がゲートされているとき、**usart_pclk** クロックの有効化を必要とする特定の動作が必要になった場合、USART はウェイクアップ割り込み（**usart_wkup**）を生成します。

- FIFO モードが無効の場合

  USART データレジスタを空にするために Usart_pclk クロックを有効にする必要があります。

  この場合、usart_wkup 割り込みのソースは“1”にセットされた RXNE です。低電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために Usart_pclk クロックを有効にする必要があります。

  – TXFIFO を満たすため
  – または RXFIFO を空にするため

  この場合、usart_wkup 割り込みのソースになる可能性のあるものは以下の通りです。

  – RXFIFO ノットエンプティ。この場合、低電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  – RXFIFO フル。この場合、低電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  – TXFIFO エンプティ。この場合、低電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、usart_wkup 割り込みソースになり得るのは次のイベントのうちの 1 つです。

  – TXFIFO 閾値に達した。この場合、低電力モードに入る前に TXFTIE ビットをセットする必要があります。
  – RXFIFO 閾値に達した。この場合、低電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信するのに必要な時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFO ノットエンプティ、および RXFIFO/TXFIFO 閾値割り込みを使用すれば、低電力モード中にできるだけ多くの USART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の **usart_wkup** 割り込みを選択することもできます。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は **usart_wkup** 割り込みが生成されます。この場合、**usart_wkup** 割り込みは必須ではなく、MCU を低電力モードからウェイクアップするためには WUF がセットされれば十分です。

注： ***低電力モードに移行する前に、USART 転送が進行中ではないことを確認してください。BUSY フラグをチェックすることでは、データ受信中に低電力モードに入らないことを保証できません。***

WUF フラグは、MCU が低電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低電力モードに入るときには、REACK ビットをチェックして、USART が有効であることを確認する必要があります。

受信に DMA が使用されるときには、低電力モードに入る前に無効化し、低電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低電力モードでのミュートモードの使用

低電力モードに入る前に USART がミュートモードになった場合は、

- アイドル検出は低電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： ***FIFO 管理が有効なとき、ミュートモードは何の制約もなく低電力モードからのウェイクアップとともに使用できます（すなわち、ミュートおよび低電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。***

低電力モードで USART カーネルクロック（usart_ker_ck）がオフのときの低電力モードからのウェイクアップ

低電力モード中、usart_ker_ck クロックがオフになっている場合、USART 受信ラインの立ち下がりエッジが検出されると、usart_ker_ck_req 信号によって USART インタフェースが usart_ker_ck クロックをオンにするようリクエストします。その後、usart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップイベントが確認されない場合、usart_ker_ck クロックが再度オフになり、MCU がウェイクアップせずに低電力モードに留まり、カーネルクロックリクエストが解除されます。

以下の例は、ウェイクアップイベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 585 に、ウェイクアップイベントが確認された時の USART の動作を示します。

図 585. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

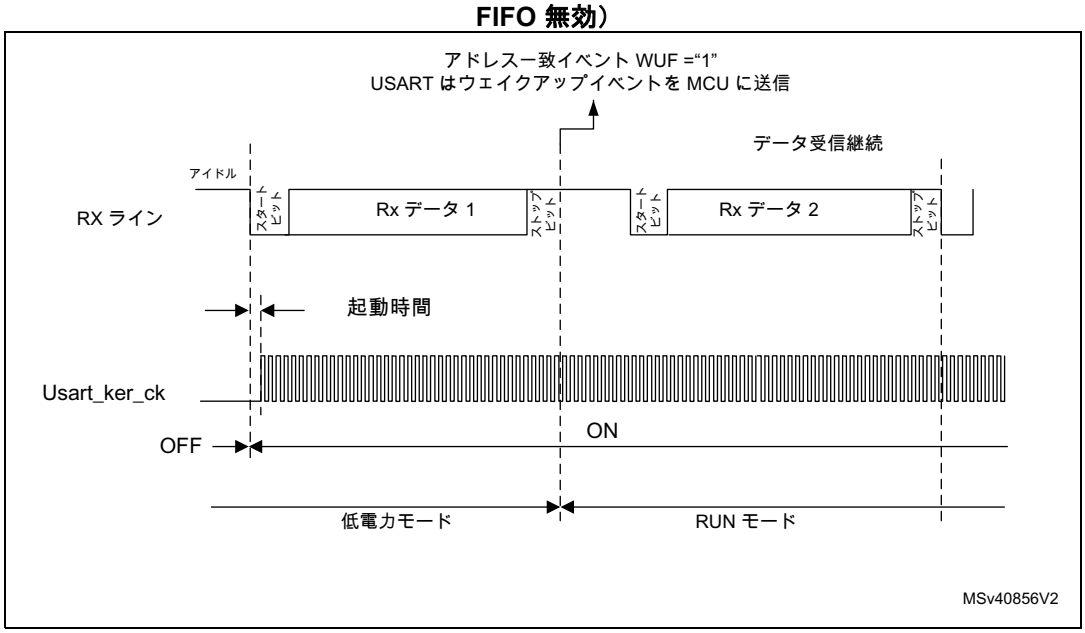
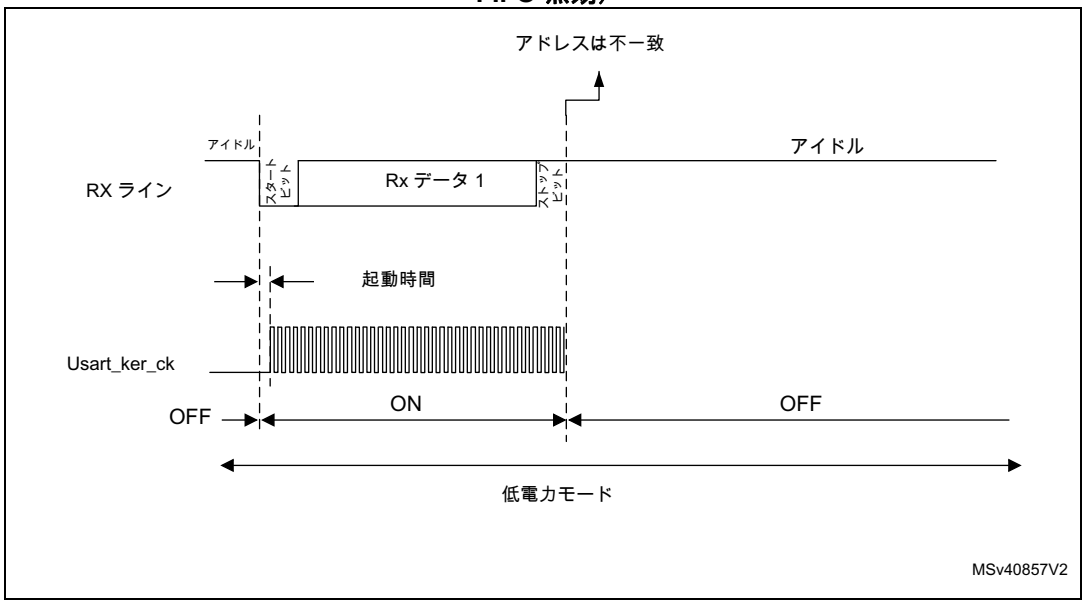


図 586 に、ウェイクアップイベントが確認されない時の USART の動作を示します。

図 586. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： ***上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、USART はスタートビットの終端にウェイクアップイベントを MCU に送ります。***

48.6 USART 割り込み

USART 通信中、割り込み（usart_it）はさまざまなイベントによって生成できます。USART ブロックもウェイクアップ割り込み（usart_wkup）を生成できます。

すべての USART 割り込みリクエストの詳細な説明については、*表 380* を参照してください。

表 380. USART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット	割り込みのクリア方法	有効になる割り込み	
				usart_it	usart_wkup
送信データレジスタエンプティ	TXE	TXEIE	TXE は、データが TDR に書き込まれたときにクリアされます。	はい	いいえ
送信 FIFO ノットフル	TXFNF	TXFNFIE	TXFNF は、TXFIFO がフルになると、クリアされます。	はい	いいえ
送信 FIFO エンプティ	TXFE	TXFEIE	TXFE は、TXFIFO に少なくとも 1 データが入ったとき、または TXFRQ ビットのセットによってクリアされます。	はい	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

参考資料

表 380. USART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット	割り込みのクリア方法	有効になる割り込み	
				usart_it	usart_wkup
CTS 割り込み	CTSIF	CTSIE	CTSIF は、CTSCF ビットをセットすることによって、ソフトウェアによってクリアされます。	はい	いいえ
送信完了	TC	TCIE	TC は、データが TDR に書き込まれたとき、または TCCF ビットのセットによって、クリアされます。	はい	いいえ
ガード時間前に送信完了	TCBGT	TCBGTIE	TCBGT は、データが TDR に書き込まれたとき、または TCBGTCF ビットのセットによってクリアされます。	はい	いいえ
受信データレジスタノットエンプティ（データの読み出し可能）	RXNE	RXNEIE	RXNE は、RDR の読み出しによって、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFNE は、RXFIFO が空になったとき、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO フル	RXFF(1)	RXFFIE	RXFF は、RXFIFO に少なくとも 1 つのデータが入ったとき、クリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
オーバーランエラー検出	ORE	RXNEIE/ RXFNEIE	ORE は、ORECF ビットをセットすることによってクリアされます。	はい	いいえ
アイドルライン検出	IDLE	IDLEIE	IDLE は、IDLECF ビットをセットすることによってクリアされます。	はい	いいえ
パリティエラー	PE	PEIE	PE は、PECF ビットをセットすることによってクリアされます。	はい	いいえ
LIN ブレーク	LBDF	LBDIE	LBDF は、LBDCF ビットをセットすることによってクリアされます。	はい	いいえ
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。	NE または ORE または FE	EIE	NE は、NFCF ビットをセットすることによってクリアされます。 ORE は、ORECF ビットをセットすることによってクリアされます。 FE フラグは、 FECF ビットをセットすることによってクリアされます。	はい	いいえ
キャラクター致	CMF	CMIE	CMF は、CMCF ビットをセットすることによってクリアされます。	はい	いいえ
レシーバタイムアウト	RTOF	RTOFIE	RTOF は、RTOCCF ビットをセットすることによってクリアされます。	はい	いいえ
ブロックの終了	EOBF	EOBIE	EOBF は、EOBCF ビットをセットすることによってクリアされます。	はい	いいえ

参考資料

表 380. USART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット	割り込みのクリア方法	有効になる割り込み	
				usart_it	usart_wkup
低電力モードからのウェイクアップ	WUF(2)	WUFIE	WUF は、WUCF ビットをセットすることによってクリアされます。	いいえ	はい
SPI スレーブのアンダーランエラー	UDR	EIE	UDR は、UDRCF ビットをセットすることによってクリアされます。	はい	いいえ
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

1. RXFF フラグは、USART が 次のように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。RXFIFO に n 個のデータ、USART_RDR に 1 個のデータ。STOP モードでは、USART_RDR はクロック供給されません。その結果として、このレジスタは書き込まれず、n 個のデータが受信されて RXFIFO に書き込まれた後に、RXFF 割り込みがアサートされます（RXFF フラグはセットされません）。
2. WUF 割り込みは、低電力モードでのみアクティブです。

48.7 USART レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

48.7.1 USART 制御レジスタ 1（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE/TXFNFIE	TCIE	RXNEIE/RXFNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの RXFF=“1”のときに、USART 割り込みが生成されます。

注： ***FIFO モードが無効のときは、このビットは予約済みであり、リセット値に保持する必要があります。***

ビット 30 **TXFEIE**：TXFIFO エンプティ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TXFE=“1”のときに、USART 割り込みが生成されます。

注： ***FIFO モードが無効のときは、このビットは予約済みであり、リセット値に保持する必要があります。***

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***FIFO モードは、SPI マスタ／スレーブモードおよびスマートカードモードでのみ、標準 UART 通信で使用できます。IrDA および LIN のモードでは有効にしてはいけません。***

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。***

ビット 27 **EOBIE**：ブロック終了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割り込みが生成されます。

注： ***USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装 を参照してください。***

ビット 26 **RTOIE**：レシーバタイムアウト割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの RTOF ビットがセットされると、USART 割り込みが生成されます。

注： ***USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。***

ビット 14 **CMIE**：キャラクター致割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割り込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは USART ミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=“1”の場合は 9 番目のビット、M=“0”の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの PE=“1”のときには、USART 割り込みが生成されます。

ビット 7 **TXEIE/TXFNFIE**：送信データレジスタエンプティ／TXFIFO ノットフル割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TXE/TXFNF=“1”のときには、USART 割り込みが生成されます。

ビット 6 **TCIE**：転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TC=“1”のときには、USART 割り込みが生成されます。

ビット 5 **RXNEIE/RXFNEIE**：受信データレジスタノットエンプティ／RXFIFO ノットエンプティ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの ORE=“1”または RXNE/RXFNE=“1”のときには、USART 割り込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの IDLE=“1”のときには、USART 割り込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： ***スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に“1”を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。***

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：USART 低電力モードで有効

このビットがクリアされると、USART は MCU を 低電力モードからウェイクアップできません。

このビットがセットされると、USART は MCU を 低電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は低電力モードから MCU をウェイクアップできません。

1：USART は低電力モードから MCU をウェイクアップできます。

注： ***低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。***

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ***ラインにエラーを生成せずに低電力モードに入るためには、TE ビットを事前にリセットする必要があり、ソフトウェアは USART_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。***

UE=“0”のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

スマートカードモードでは（SCEN = “1”）、SCLK は UE ビット値にかかわらず、CLKEN = “1”の場合に常に使用可能です。

48.7.2 USART 制御レジスタ 2（USART_CR2）

アドレスオフセット：0x04

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:4]				ADD[3:0]				RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
スワップ	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS.	Res.	Res.	SLVEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

ビット 31:28 **ADD[7:4]**：USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

ミュートモードまたは低電力モード中に、マルチプロセッサ通信で 7 ビットアドレスマーク検出によって MCU をウェークアップするために使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが非アクティブなときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のためにも使用することができます。この場合、受信されたキャラクタ全体（8 ビット）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき（RE=“0”）または USART が無効のとき（UE=“0”）のみ、書き込むことができます。

ビット 27:24 **ADD[3:0]**：USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、ミュートモードまたは低電力モード中に、マルチプロセッサ通信でアドレスマーク検出によってウェイクアップするために使用されます。

このビットフィールドは、受信が無効のとき（RE=“0”）または USART が無効のとき（UE=“0”）のみ、書き込むことができます。

ビット 23 **RTOEN**：レシーバタイムアウト有効

このビットは、ソフトウェアによってセット／クリアされます。

0：レシーバタイムアウト機能は無効です。

1：レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR（レシーバタイムアウトレジスタ）でプログラムされた時間にわたって RX ラインがアイドル（受信なし）であった場合、USART_ISR レジスタの RTOF フラグがセットされます。

注： ***USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 22:21 **ABRMOD[1:0]**：自動ボーレートモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：スタートビットの測定がボーレートの検出に使用されます。

01：立ち下がりエッジから立ち下がりエッジまでの測定（受信されたフレームはシングルビット = 1 で始まらなければならず、その場合、フレーム = Start10xxxxxx）

10：0x7F フレーム検出

11：0x55 フレーム検出

このビットフィールドは、ABREN=“0”または USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***DATAINV=“1”および／または MSBFIRST=“1”の場合、パターンはライン上で同じである必要があります（たとえば、MSBFIRST の場合は 0xAA）。***

USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 20 **ABREN**：自動ボーレート有効

このビットは、ソフトウェアによってセット／クリアされます。

0：自動ボーレート検出は無効です。

1：自動ボーレート検出は有効です。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TX ピン信号値は反転されます。（(V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RX ピン信号値は反転されます。（(V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 15 **スワップ**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 14 **LINEN**：LIN モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LIN モードは無効です。

1：LIN モードは有効です。

LIN モードでは、USART_CR1 レジスタの SBKRQ ビットを使用して LIN 同期ブレーク（下位 13 ビット）を送信し、LIN 同期ブレークを検出することができます。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に“0”に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：0.5 個のストップビット

10：2 個のストップビット

11：1.5 個のストップビット

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 11 **CLKEN**：クロック有効

このビットによって、SCLK ピンを有効にできます。

0：SCLK ピンは無効です。

1：SCLK ピンは有効です。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： 同期モードまたはスマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に“0”に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

スマートカードモードでは、スマートカードに SCLK クロックを正しく供給するには、次のステップを順守する必要があります。

- UE = “0”
- SCEN = “1”
- GTPR 設定
- CLKEN = “1”
- UE = “1”

ビット 10 **CPOL**：クロック極性

このビットによって、同期モードにおける SCLK ピンのクロック出力の極性を選択できます。CPHA ビットと連携して動作し、希望するクロック／データ関係になるようにします。

0：送信ウィンドウの外で、SCLK ピンはローレベルを維持します。

1：送信ウィンドウの外で、SCLK ピンはハイレベルを維持します。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 9 **CPHA**：クロック位相

このビットは、同期モードでの SCLK ピンのクロック出力の位相を選択するために使用されます。CPOL ビットと連携して動作し、希望するクロック／データ関係になるようにします（*図 566* および *図 567* を参照）。

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 8 **LBCL**：最終ビットのクロックパルス

このビットは、同期モードで送信される最終データビット（MSB）に関連するクロックパルスを、SCLK ピンに出力する必要があるかどうかを選択するために使用されます。

0：最終データビットのクロックパルスは、SCLK ピンに出力されません。

1：最終データビットのクロックパルスは、SCLK ピンに出力されます。

注意： 最終ビットは、USART_CR1 レジスタの M ビットによって選択された 7 または 8 または 9 ビットフォーマットに応じて送信された 7 番目または 8 番目または 9 番目のデータビットです。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LBDIE**：LIN ブレーク検出割り込み有効

ブレーク割り込みマスクです（ブレークデリミタを使用したブレーク検出）。

0：割り込みは禁止されています。

1：USART_ISR レジスタの LBDF="1"のときには、割り込みが生成されます。

注： ***LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 5 **LBDL**：LIN ブレーク検出長

このビットでは、10 ビットと 11 ビットのブレーク検出を選択します。

0：10 ビットブレーク検出

1：11 ビットブレーク検出

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。***

ビット 3 **DIS_NSS**

DIS_NSS ビットがセットされているとき、NSS ピンの入力は無視されます。

0：SPI スレーブ選択は NSS 入力ピンに依存します。

1：SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

注： ***SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 48.4：1948 ページのUSART の実装 を参照してください。***

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SLVEN**：同期スレーブモード有効

SLVEN ビットがセットされると、同期スレーブモードが有効になります。

0：スレーブモードは無効です。

1：スレーブモードは有効です。

注： ***SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 48.4：1948 ページのUSART の実装 を参照してください。***

注： ***トランスミッタが有効なときには、CPOL、CPHA、LBCL のビットに書き込まないでください。***

参考資料

48.7.3 USART 制御レジスタ 3（USART_CR3）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG			RXF TIE	RXFTCFG			TCBG TIE	TXFTIE	WUFIE	WUS [2:0]		SCARCNT2:0]			Res.
rw			rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	ONE BIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HD SEL	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **TXFTCFG：**TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
010：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済み

Bit28 **RXFTIE**：RXFIFO 閾値割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、USART 割り込みが生成されます。

ビット 27:25 **RXFTCFG：**受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
010：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済み

ビット 24 **TCBGTIE**：ガード時間前送信完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：USART_ISR レジスタの TCBGT=“1”のときには、USART 割り込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装 を参照してください。

ビット 23 **TXFTIE**：TXFIFO 閾値割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、USART 割り込みが生成されます。

ビット 22 **WUFIE**：低電力モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの WUF="1"のときには、USART 割り込みが生成されます。

注： ***WUFIE は、低電力モードに入る前にセットする必要があります。***

WUF 割り込みは、低電力モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。

ビット 21:20 **WUS [1:0]**：低電力モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低電力モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み。

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE/RXFNE 時に有効になります。

このビットフィールドは、USART が無効（UE="0"）のときのみ書き込むことができます。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。

ビット 19:17 **SCARCNT[2:0]**：スマートカード自動再試行カウント

このビットフィールドは、スマートカードモードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します（FE ビットをセット）。

受信モードでは、受信エラーが生成される（RXNE/RXFNE および PE ビットのセット）までの受信の試行エラー回数を指定します。

このビットフィールドは、USART が無効（UE="0"）のときのみプログラムする必要があります。

USART が有効になると（UE="1"）、このビットフィールドは再送信を停止するために 0x0 にのみ書き込み可能です。

0x0：再送信無効 - 送信モードでの自動再送信禁止

0x1 から 0x7：自動再送信試行回数（信号エラーの生成前）

注： ***スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます。（スマートカードモードで使用されます）

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR="0"）、RXNE（FIFO モードが有効な場合は RXFNE）をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 ：**OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが USART_RDR レジスタの前の内容に上書きされます。FIFO モードが有効なとき、RXFIFO はバイパスされ、データは直接、USART_RDR レジスタに書き込まれます。FIFO 管理が有効になっているときでも、RXNE フラグが使用されます。

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 **ONEBIT**：1 サンプルビット方式有効

このビットによって、サンプル方式を選択できます。1 サンプルビット方式が選択されると、ノイズ検出フラグ（NE）が無効になります。

0：3 サンプルビット方式

1：1 サンプルビット方式

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

ビット 10 **CTSIE**：CTS 割り込み有効

0：割り込みは禁止されています。

1：USART_ISR レジスタの CTSIF="1"のときには、割り込みが生成されます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは nCTS 入力がアサート（0 に固定）されている場合にのみ転送されます。データ転送中に nCTS 入力がネゲートされると、送信は停止前に完了します。nCTS がネゲートされている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、nRTS 出力がアサートされます（0 にプルされます）。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5 **SCEN**：スマートカードモード有効

このビットはスマートカードモードを有効にするために使用します。

0：スマートカードモードが無効です。

1：スマートカードモードが有効です。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装 を参照してください。***

ビット 4 **NACK**：スマートカード NACK 有効

0：パリティエラーの際の NACK 転送が無効です。

1：パリティエラー時の NACK 転送が有効です。

このビットフィールドは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装 を参照してください。***

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 2 **IRLP**：IrDA 低電力

このビットは、通常と低電力の IrDA モードの選択に使用されます。

0：通常モード

1：低電力モード

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 1 **IREN**：IrDA モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：IrDA は無効です。

1：IrDA は有効です。

このビットは、USART が無効（UE=“0”）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。

ビット 0 **EIE**：エラー割り込みイネーブル

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、ノイズフラグ、または SPI スレーブアンダーランエラー（USART_ISR レジスタの FE=“1”または ORE=“1”または NE=“1”または UDR =“1”）の場合に割り込み生成を有効にするために必要です。

0：割り込みは禁止されています。

1：USART_ISR レジスタの FE=“1”または ORE=“1”または NE=“1”または UDR=“1”（SPI スレーブモードで）のときに、割り込みが生成されます。

48.7.4 USART ボーレートレジスタ（USART_BRR）

このレジスタは、USART が無効（UE=“0”）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **BRR[15:4]**

BRR[15:4] = USARTDIV[15:4]

ビット 3:0 **BRR[3:0]**

OVER8 = “0”のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = “1”のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。

BRR[3] は、クリアされたままにする必要があります。

48.7.5 USART ガード時間およびプリスケーラレジスタ（USART_GTPR）

アドレスオフセット：0x10

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								PSC[7:0]							
rw								rw							

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **GT[7:0]**：ガード時間値

このビットフィールドは、ガード時間値をボークロック周期数でプログラムするために使用します。
これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。
このビットフィールドは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 7:0 **PSC[7:0]**：プリスケーラ値

IrDA 低電力および IrDA 通常モード：

PSC[7:0] = IrDA 通常および低電力ボーレート
USART クロックソースを分周して低電力周波数を得るためのプリスケーラのプログラミングに使用します。
クロックソースは、レジスタに与えられた値（上位 8 ビット）で分周されます。
00000000：予約済み - この値はプログラミングしないでください。
00000001：クロックソースは 1 で分周されます。
00000010：クロックソースは 2 で分周されます。
...

スマートカードモード：

PSC[4:0]：プリスケーラ値
USART クロックソースを分周してスマートカードのクロックを提供するプリスケーラのプログラミングに使用します。
レジスタで指定された値（上位 5 ビット）を 2 倍して、クロックソース周波数の分周比を求めます。
00000：予約済み - この値はプログラミングしないでください。
00001：クロックソースは 2 で分周されます。
00010：クロックソースは 4 で分周されます。
00011：クロックソースは 6 で分周されます。
...
このビットフィールドは、USART が無効（UE="0"）のときのみ書き込むことができます。

注： ***スマートカードモードが使用される場合、ビット [7:5] はクリアされたままにする必要があります。スマートカードモードや IrDA モードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に"0"に設定されます。セクション 48.4：1948 ページの USART の実装 を参照してください。***

48.7.6 USART レシーバタイムアウトレジスタ（USART_RTOR）

アドレスオフセット：0x14

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLEN[7:0]								RTO[23:16]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **BLEN[7:0]**：ブロック長

このビットフィールドは、受信時のスマートカード T=“1”のブロック長を指定します。この値は、情報文字の数 + エピローグフィールドの長さ（1-LEC/2-CRC）- 1 と等しくなります。

例：

BLEN = 0 -> 0 情報文字 + LEC

BLEN = 1 -> 0 情報文字 + CRC

BLEN = 255 -> 254 情報文字 + CRC（合計 256 文字）

スマートカードモードでは、TXE=“0”（FIFO モードが有効な場合は TXFE =“0”）のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。この場合、RE=“0”（レシーバ無効）のとき、および/または EOBCF ビットが 1 に書き込まれたときに、ブロック長カウンタがリセットされます。

注： この値は、ブロック受信の開始後にプログラムできます（プロローグフィールドの LEN 文字のデータを使用）。必ず受信したブロックにつき一度だけプログラムするようにしてください。

ビット 23:0 **RTO[23:0]**：レシーバタイムアウト値

このビットフィールドは、レシーバタイムアウト値をボークロック数で指定します。

標準モードでは、最後の受信キャラクタの後、RTO 値を超える間、新しいスタートビットが検出されなかった場合、RTOF フラグがセットされます。

スマートカードモードでは、この値は CWT および BWT を実装するために使用されます。詳細については、スマートカードの章を参照してください。この標準では、CWT/BWT 測定は最後の受信キャラクタのスタートビットから開始して行われます。

注： この値は、受信キャラクタごとにプログラムされる必要があります。

注： RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって 0x00000000 に強制的に設定されます。セクション 48.4： 1948 ページのUSART の実装 を参照してください。

48.7.7 USART リクエストレジスタ（USART_RQR）

アドレスオフセット：0x18

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
											w_r0	w_r0	w_r0	w_r0	w_r0

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

FIFO モードが無効になっている場合は、このビットに“1”を書き込むと、TXE フラグがセットされます。これにより、送信データを破棄できます。このビットは、エラー（NACK）によりデータが送信されなかった場合、および USART_ISR レジスタで FE フラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に“0”に設定されます。

FIFO モードが有効な場合、FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（送信 FIFO エンプティ、USART_ISR レジスタのビット 23）がセットされます。送信 FIFO の一掃は、UART モードおよびスマートカードモードの両方でサポートされています。

注： FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに“1”を書き込むと、受信 FIFO 全体を空にします（すなわち、RXFNE ビットをクリアします）。これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、USART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに“1”を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 **ABRRQ**：自動ボーレートリクエスト

このビットに“1”を書き込むと、USART_ISR の ABRF フラグがリセットされ、次の受信データフレームでの自動ボーレート測定をリクエストします。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

48.7.8 USART 割り込みおよびステータスレジスタ（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0（FIFO が無効の場合）

リセット値：0x0280 00C0（FIFO／スマートカードモード が有効の場合）

リセット値：0x0080 00C0（FIFO が有効で、スマートカードモード が無効の場合）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE/TX FNF	TC	RXNE/ RXFNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは、TXFIFO が USART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。USART_CR3 レジスタの TXFTIE ビット =“1”（ビット 23）の場合、割り込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、USART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したときに、ハードウェアによってセットされます。これは、受信 FIFO に（RXFTCFG - 1）個のデータがあり、USART_RDR レジスタに 1 個のデータがあることを意味します。USART_CR3 レジスタの RXFTIE ビット =“1”（ビット 28）の場合、割り込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

注： RXFTCFG 閾値が“101”に設定されているとき、16 個のデータが利用可能になった場合（すなわち、RXFIFO に 15 個のデータが入り、USART_RDR に 1 個のデータが入った場合）、RXFT フラグがセットされます。したがって、17 番目の受信データによってオーバーランエラーが起きることはありません。オーバーランエラーは 18 番目のデータを受信した後に発生します。

ビット 25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードが NACK を返信しなかった場合、このビットはハードウェアによってセットされます。また、USART_CR3 レジスタで TCBGTIE=“1”であれば、割り込みが生成されます。

このビットは、USART_ICR レジスタの TCBGTCF に“1”を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信が完了していないか、または送信が正常に完了していません（つまり、カードから NACK を受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードから NACK なし）。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。USART がスマートカードモードをサポートしている場合でこのモードが有効な場合、TCBGT のリセット値は“1”です。セクション 48.4：1948 ページのUSART の実装 を参照してください。

ビット 24 **RXFF**：RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、USART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

USART_CR1 レジスタの RXFFIE ビット =“1”である場合、割り込みが生成されます。

0：RXFIFO はフルではありません。

1：RXFIFO はフルです。

ビット 23 **TXFE**：TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、USART_RQR レジスタのビット TXFRQ（ビット 4）に“1”を書き込むことによってセットすることもできます。

USART_CR1 レジスタの TXFEIE ビット =“1”（ビット 30）の場合、割り込みが生成されます。

0：TXFIFO は空ではありません。

1：TXFIFO は空です。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、USART が受信できる状態であることを確認できます。

注： ***USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。***

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます

USART_CR1 レジスタで TE=“0”を書き込んだ後、TE=“1”を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=“0”の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE=“1”である場合、割り込みが生成されます。

注： ***UESM がクリアされると、WUF フラグもクリアされます。***

WUF 割り込みは、低電力モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに“1”を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： ***USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4：1948 ページの USART の実装を参照してください。***

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3 レジスタの SBKRQ ビットに“1”を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタは送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE=“1”である場合、割り込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたときにハードウェアによってセットされ（RXNE もセットされ、RXNEIE=“1”の場合は割り込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにセットされます（ABRE=“1”）（この場合、ABRE、RXNE、および FE もセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQR レジスタの ABRRQ に“1”を書き込むことによって、ソフトウェアによってクリアされます。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_CR3 レジスタの ABRRQ ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 13 **UDR**：SPI スレーブアンダーランエラーフラグ

スレーブ送信モードでは、ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICR レジスタの UDRCF ビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： ***USART が SPI スレーブモードをサポートしていない場合、このビットは予約済みであり、ハードウェアによって強制的に“0”に設定されます。セクション 48.4：1948 ページのUSART の実装を参照してください。***

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T="1"スマートカードモード）。検出は、受信バイト数が BLEN + 4 以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR2 レジスタの EOBIE="1"である場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に"1"を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 11 **RTOF**：レシーバタイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに"1"を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの RTOIE="1"の場合、割り込みが生成されます。

スマートカードモードでは、タイムアウトは CWT または BWT タイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE="0"の場合でもカウントしますが、RTOF は RE="1"のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 **CTSIF**：CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに"1"を書き込むことによって、ソフトウェアによってクリアされます。

また、USART_CR3 レジスタで CTSIE="1"であれば、割り込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 **LBDF**：LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に"1"を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE="1"である場合、割り込みが生成されます。

0：LIN ブレークは検出されませんでした。

1：LIN ブレークが検出されました。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 7 **TXE/TXFNF**：送信データレジスタエンプティ／TXFIFO ノットフル

FIFO モードが無効のとき、TXE は、USART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_TDR レジスタへの書き込みによってクリアされます。TXE フラグは、USART_RQR レジスタの TXFRQ に“1”を書き込んでセットし、データを破棄することもできます（スマートカード T=“0”モードでの送信失敗の場合のみ）。

FIFO モードが有効のとき、TXFNF は、TXFIFO がフルでないとき、つまりデータを USART_TDR に書き込めるとき、ハードウェアによってセットされます。USART_TDR レジスタへの書き込み動作ごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを USART_TDR に書き込むことができないことを示します。

USART_CR1 レジスタの TXEIE/TXFNFIE ビット =“1”である場合、割り込みが生成されます。

0：データレジスタはフル／送信 FIFO はフルです。

1：データレジスタ／送信 FIFO はフルではありません。

注： ***一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです（TXFNF および TXFE は同時にセットされます）。***

このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから送信されたことを示します。

データを含むフレームの送信が完了し、TXE/TXFE がセットされたとき、ハードウェアによってセットされます。

USART_CR1 レジスタの TCIE=“1”である場合、割り込みが生成されます。

TC ビットは、USART_ICR レジスタの TCCF に“1”を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信は完了していません。

1：送信は完了しています。

注： ***TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。***

ビット 5 **RXNE/RXFNE**：読み出しデータレジスタノットエンプティ／RXFIFO ノットエンプティ

RXNE ビットは、USART_RDR シフトレジスタの内容が USART_RDR レジスタに転送されると、ハードウェアによってセットされます。これは、USART_RDR レジスタからの読み出しによってクリアされます。RXNE フラグは、USART_RQR レジスタの RXFRQ に“1”を書き込むことによってクリアすることもできます。

RXFNE ビットは、RXFIFO が空でないとき、つまりデータが USART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。USART_RDR からの読み出し動作のたびに、RXFIFO の 1 つの場所が解放されます。

RXFIFO が空になると、RXFNE/RXFNE がクリアされます。RXNE/RXFNE フラグは、USART_RQR レジスタの RXFRQ に“1”を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXNEIE/RXFNEIE=“1”の場合、割り込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE=“1”である場合、割り込みが生成されます。USART_ICR レジスタの IDLECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： ***RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。***

ミュートモードが有効な場合（MME=“1”）、USART がミュートでない場合（RWU=“0”）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=“1”の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、シフトレジスタで現在受信中のデータを USART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。これは、USART_ICR レジスタの ORECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RXNEIE/ RXFNEIE=“1”または EIE=“1”の場合、割り込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： ***このビットがセットされると、USART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。***

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NECF ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： ***このビットは、割り込みを生成する RXNE/RXFNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割り込みが生成されます。***

ラインがノイズフリーであるとき、NE フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（セクション 48.5.8：1965 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。

FIFO モードでは、このエラーは USART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： ***FIFO モードでは、このエラーは USART_RDR 内のキャラクタに関連します。***

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=“1”である場合、割り込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： ***FIFO モードでは、このエラーは USART_RDR 内のキャラクタに関連します。***

48.7.9 USART 割り込みフラグクリアレジスタ（USART_ICR）

アドレスオフセット：0x20

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.
											w			w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
		w	w	w		w	w	w	w	w	w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低電力モードからのウェイクアップフラグクリア

このビットに“1”を書き込むと、USART_ISR レジスタの WUF フラグがクリアされます。

注： USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページの USART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに“1”を書き込むと、USART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **UDRCF**：SPI スレーブアンダーランフラグクリア

このビットに“1”を書き込むと、USART_ISR レジスタの UDRF フラグがクリアされます。

注： USART が SPI スレーブモードをサポートしていない場合、このビットは予約済みであり、ハードウェアによって強制的に“0”に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 12 **EOBCF**：ブロック終了クリアフラグ

このビットに“1”を書き込むと、USART_ISR レジスタの EOBF フラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装 を参照してください。

ビット 11 **RTOCF**：レシーバタイムアウトフラグクリア

このビットに“1”を書き込むと、USART_ISR レジスタの RTOF フラグがクリアされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに“1”を書き込むと、USART_ISR レジスタの CTSIF フラグがクリアされます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 8 **LBDCF**：LIN ブレーク検出クリアフラグ

このビットに“1”を書き込むと、USART_ISR レジスタの LBDF フラグがクリアされます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4： 1948 ページのUSART の実装を参照してください。

ビット 7 **TCBGTCF**：ガード時間前送信完了フラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの TCBGT フラグがクリアされます。

ビット 6 **TCCF**：送信完了フラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの TC フラグがクリアされます。

ビット 5 **TXFECF**：TXFIFO エンプティフラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの TXFE フラグがクリアされます。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア
このビットに“1”を書き込むと、USART_ISR レジスタの PE フラグがクリアされます。

48.7.10 USARTレシーバデータレジスタ（USART_RDR）

アドレスオフセット：0x24

リセット値：未定義

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値
受信データキャラクタを含みます。
RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（*図 560* を参照）。
パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

参考資料

48.7.11 USARTトランスミッタデータレジスタ（USART_TDR）

アドレスオフセット：0x28

リセット値：未定義

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

USART_TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（*図 560* を参照）。

パリティを有効にして（USART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF="1"のときのみ書き込む必要があります。

48.7.12 USART プリスケーラレジスタ（USART_PRESC）

このレジスタは、USART が無効（UE="0"）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

USART 入力クロックは、以下のようにプリスケーラ分周比によって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済み

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は“1011”になります。すなわち、入力クロックは 256 分周されます。

48.7.13 USART レジスタマップ

次の表に、USART のレジスタマップとリセット値を示します。

表 381. USART レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	USART_CR1	RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT4	DEAT3	DEAT2	DEAT1	DEAT0	DEDT4	DEDT3	DEDT2	DEDT1	DEDT0	OVER8	CMIE	MME	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	USART_CR2	ADD[7:4]				ADD[3:0]				RTOEN	ABRMOD1	ABRMOD0	ABREN	MSBFIRST	DATAINV	TXINV	RXINV	スワップ	LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0			0
0x08	USART_CR3	TXFT CFG			RXFTIE	RXFT CFG			TCBGTIE	TXFTIE	WUFIE	WUS [1:0]		SCAR CNT[2:0]			Res.	DEP	DEM	DDRE	OVRDIS	ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	USART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	USART_GTPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GT[7:0]								PSC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	USART_RTOR	BLEN[7:0]								RTO[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	USART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
	リセット値																												0	0	0	0	0
0x1C	USART_ISR	Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値					0	0	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0x20	USART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0				0	0	0		0	0	0	0	0	0	0	0	0	0
0x24	USART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	USART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x2C	USART_ PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

49 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）

このセクションでは、低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）について説明します。

49.1 LPUART の概要

LPUART は、限られた消費電力で双方向 UART 通信が可能な UART です。わずか 32.768 kHz LSE クロックで最大 9600 baud/s の UART 通信が可能です。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。

マイクロコントローラが低電力モードのときでも、LPUART は UART フレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUART には、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェアサポートが含まれています。

半二重単線通信とモデム操作（CTS/RTS）をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信に DMA（直接メモリアクセス）を使用できます。

49.2 LPUART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- プログラム可能なボーレート
- 32.768 kHz クロックソースを使用して、300 baud/s から 9600 baud/s まで。
- より高い周波数のクロックソースを使用することにより、より高いボーレートを達成可能
- データを送信および受信する 2 つの内部 FIFO

  各 FIFO はソフトウェアで有効／無効にすることができ、FIFO の状態用のステータスフラグを装備
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 転送検出フラグ：
  - 受信バッファフル
  - 送信バッファエンプティ
  - ビジーおよび送信終了フラグ
- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 4 つのエラー検出フラグ：
  - オーバーランエラー
  - ノイズ検出
  - フレームエラー
  - パリティエラー
- フラグ付き割り込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ

参考資料

49.3 LPUART 機能の詳細

49.3.1 LPUART ブロック図

図 587. LPUART ブロック図

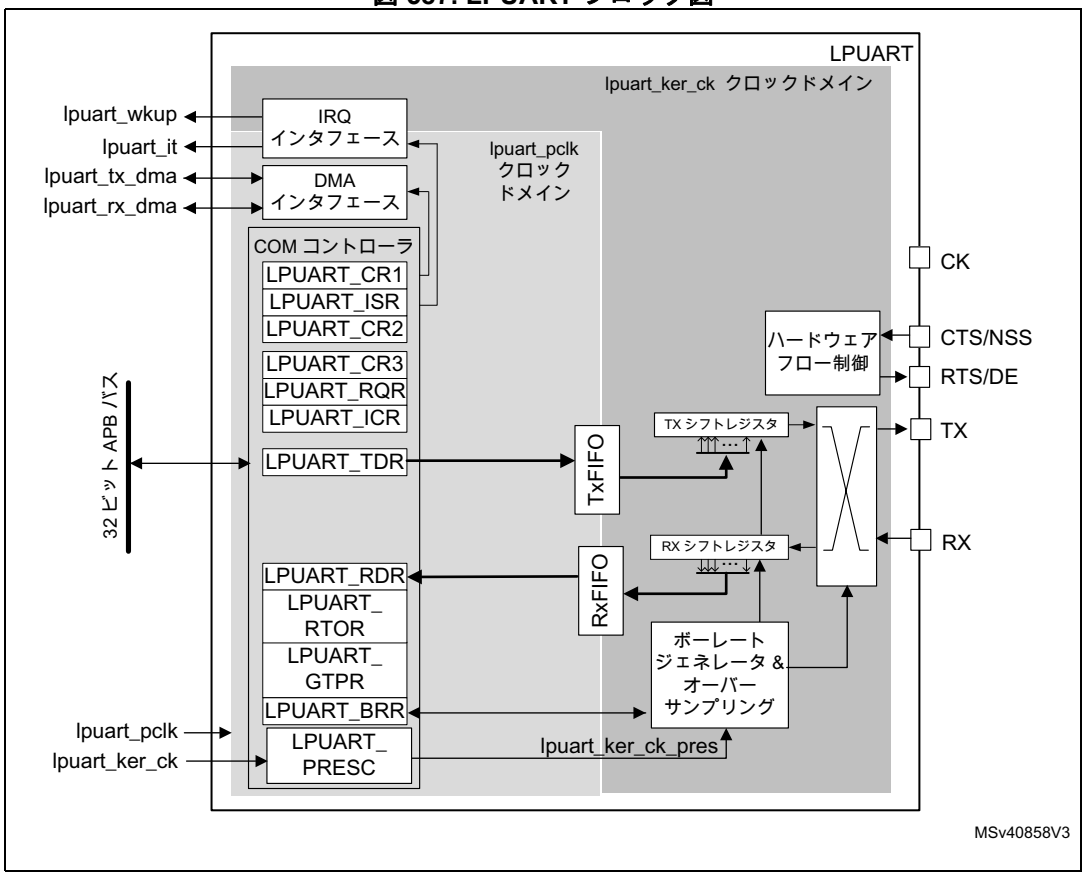


図 587 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロックドメインが示されています。

- **lpuart_pclk** クロックドメイン

  lpuart_pclk クロック信号はペリフェラルバスインタフェースにクロック供給します。LPUART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **lpuart_ker_ck** カーネルクロックドメイン

  lpuart_ker_ck は LPUART クロックソースです。**lpuart_pclk** とは独立で、RCC によって供給されます。したがって、**lpuart_ker_ck** クロックが停止しているときでも、LPUART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**lpuart_ker_ck** は lpuart_pclk クロックと同じになります。

lpuart_pclk と **lpuart_ker_ck** の間には何も制約はありません。**lpuart_ker_ck** は **lpuart_pclk** より速いことも遅いこともあり、十分に速く通信を管理できるソフトウェアの能力以外に制限はありません。

49.3.2 LPUART 信号

LPUART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）
  RX はシリアルデータ入力です。
- **TX** （送信データ出力）
  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線モードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）
  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。
- **RTS**（Request To Send）
  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェアフロー制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（ドライバイネーブル）
  この信号は、外部トランシーバの送信モードを有効にします。

注： ***DE と RTS は同じピンを共有します。***

49.3.3 LPUART キャラクタの説明

ワード長は、LPUART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 ビットに設定できます（*図 561* を参照）。

- 7 ビットのキャラクタ長：M[1:0] = “10”
- 8 ビットのキャラクタ長：M[1:0] = “00”
- 9 ビットのキャラクタ長：M[1:0] = “01”

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

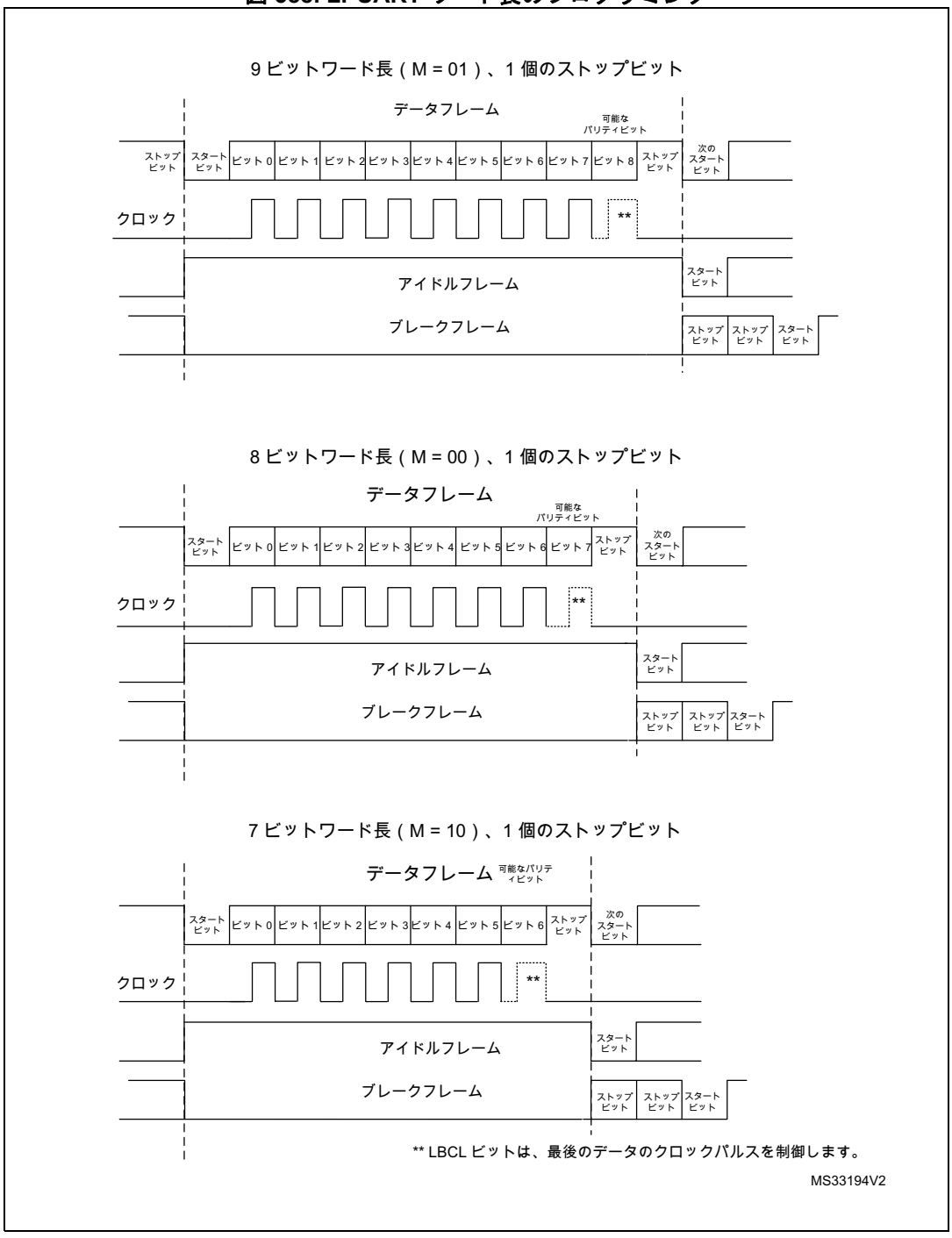
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 588. LPUART ワード長のプログラミング



49.3.4 LPUART の FIFO と閾値

LPUART は FIFO モードで動作できます。

LPUART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、LPUART_CR1 レジスタの FIFOEN ビット（ビット 29）をセットすることによって有効になります。

最大のデータワード長が 9 ビットなので、TXFIFO は 9 ビット幅です。しかし、RXFIFO のデフォルト幅は 12 ビットです。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： ***受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。***

ステータスフラグは、LPUART_ISR レジスタで入手可能です。

Tx および Rx 割り込みがトリガされる TXFIFO および RXFIFO のレベルを定義することができます。これらの閾値は、LPUART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、Rx 割り込みが生成されます。

  この場合、LPUART_ISR レジスタの RXFT フラグがセットされます。これは、RXFTCFG 分のデータが受信されたことを意味します。つまり、1 データが LPUART_RDR にあり、(RXFTCFG - 1）データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された時に、RXFT フラグがセットされます。つまり、FIFO サイズ - 1 のデータが RXFIFO に入り、1 データが LPUART_RDR に入っています。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO 内の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、Tx 割り込みが生成されます。

49.3.5 LPUART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは、TX ピンで出力されます。

キャラクタ送信

LPUART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、LPUART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（*図 587* を参照）。

FIFO モードが有効なとき、LPUART_TDR レジスタに書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 ビット周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 1 または 2 にできます。

注： ***送信データを LPUART_TDR に書き込む前に、TE ビットをセットする必要があります。***

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

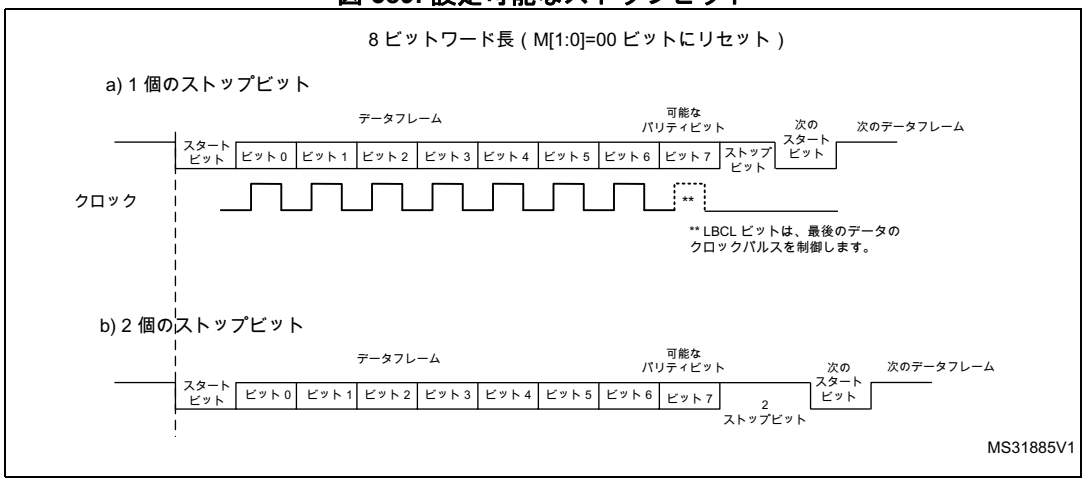
各キャラクタとともに送信されるストップビットの数は、LPUART_CR2（ビット 13、12）でプログラミングできます。

- ***1 個のストップビット***：ストップビット数のデフォルト値です。
- ***2 個のストップビット***：これは、通常の LPUART モード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10 個のロービット（M[1:0] = “00”のとき）または 11 個のロービット（M[1:0] = “01”のとき）または 9 個のロービット（M[1:0] = “10”のとき）の後に 2 個のストップビットが続きます。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 589. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. LPUART_BRR レジスタを使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合は、LPUART_CR3 レジスタの DMA 有効（DMAT）を選択します。*セクション 48.5.10 : USART マルチプロセッサ通信*の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを LPUART_TDR レジスタに書き込みますシングルバッファの場合、送信される各データにこの操作を繰り返します。

– FIFO モードが無効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。
– FIFO モードが有効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、LPUART_TDR への書き込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。

8. LPUART_TDR レジスタに最後のデータを書き込んだら、TC=“1”になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。
– FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
– FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

このチェックは、LPUART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされ、以下のことを示します。
  – データは LPUART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、TXE フラグは割り込みを生成します。

  送信が行われているとき、LPUART_TDR レジスタへの書き込み命令によってデータが TDR レジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、LPUART_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  – TXFIFO はフルではありません。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。送信が行われているとき、LPUART_TDR レジスタへの書き込み命令によってデータが TXFIFO に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

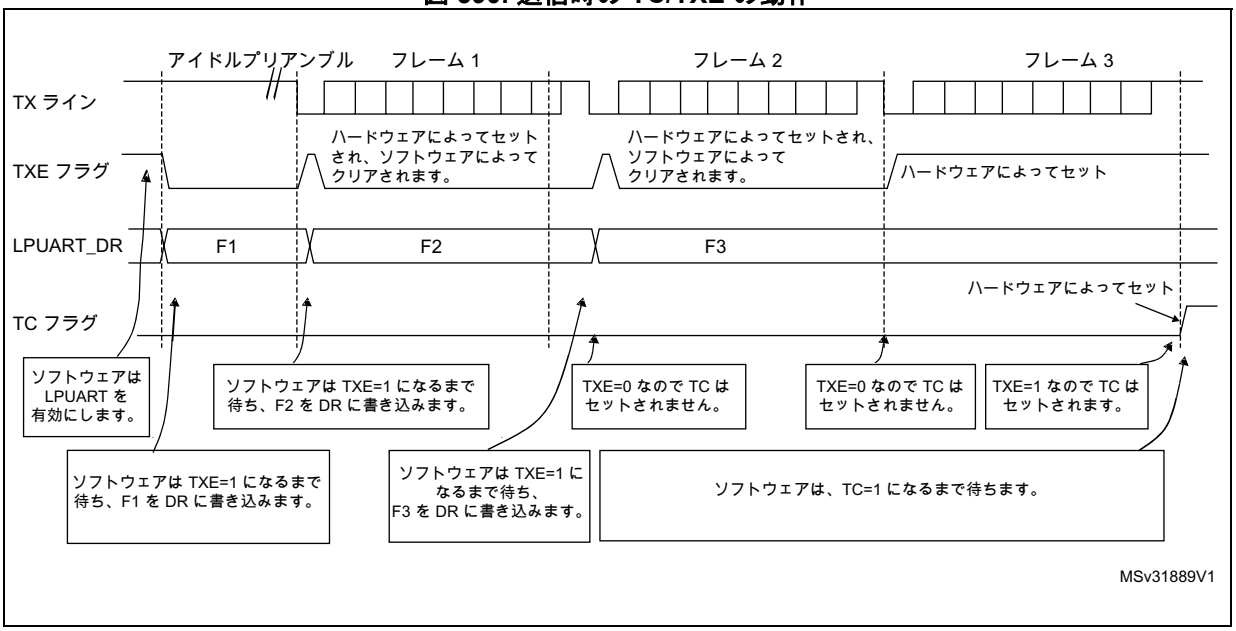
  TXFIFO がフルではない場合、LPUART_TDR レジスタへの書き込み後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNEIE ビットがセットされている場合、このフラグは割り込みを生成します。

  あるいは、TXFIFO 閾値に達した時、割り込みが生成され、データを TXFIFO に書き込むことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを書き込むことができます。

  フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC ビットはハイレベルになります。LPUART_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

  LPUART_TDR レジスタに最後のデータを書き込んだ後は、LPUART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=“1”になるまで待つ必要があります（*図 590：送信時の TC/TXE の動作* を参照）。

図 590. 送信時の TC/TXE の動作



注： ***FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。***

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（*図 588* を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書き込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。LPUART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（STOP）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、LPUART は、最初のデータフレームの前にアイドルフレームを送信します。

49.3.6 LPUART レシーバ

LPUART は、LPUART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

LPUART では、立ち下がりエッジが Rx ラインで発生したときスタートビットが検出され、その後、スタートビットの中央でサンプルが採取され、まだ“0”であることが確認されます。スタートサンプルが“1”であった場合、ノイズエラーフラグ（NE）がセットされ、スタートビットが破棄され、レシー

バは新しいスタートビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルト設定）でシフトインされます。このモードでは、LPUART_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ LPUART_BRR を使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3 レジスタの DMA 有効（DMAR）を選択します。*セクション 48.5.10 : USART マルチプロセッサ通信*の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効な場合は、RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読み出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示す RXFNE ビットがセットされます。LPUART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードの場合は RXFNEIE）ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効のとき、RXNE フラグはバイト受信ごとにセットされ、受信データレジスタの DMA 読み出しによってクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる LPUART_RDR レジスタからの読み出しによって行われます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に“1”を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。LPUART_RDR レジスタからの読み出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、LPUART_RQR レジスタの RXFRQ ビットに“1”を書き込むことによってもクリアす

ることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるため、次のキャラクタの受信が終了する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割り込みを生成します。あるいは、RXFIFO 閾値に達した時、割り込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割り込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。LPUART_RDR の読み出しが行われると、前のデータが使用できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割り込みが生成されます。

- FIFO モードが有効な場合

  受信 FIFO がフルのときにシフトレジスタが転送される準備ができると、オーバーランエラーが発生します。

  RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから LPUART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

  RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RXFIFO の最初のエントリは失われません。LPUART_RDR の読み出しを行うと、そのデータが入手できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIE ビットまたは EIE ビットがセットされている場合、割り込みが生成されます。

ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： ***ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。***

FIFO モードが無効になっているとき、2 つの可能性があります。

- *RXNE="1"の場合、有効な最後のデータは、受信レジスタ（RDR）に格納され、読み出しが可能です。*
- *RXNE="0"の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。*

クロックソースの選択

クロックソースの選択は、クロック制御システムを通じて行われます（*リセットおよびクロック制御（RCC）のセクション*を参照）。クロックソースは、UE ビットのセットによって LPUART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

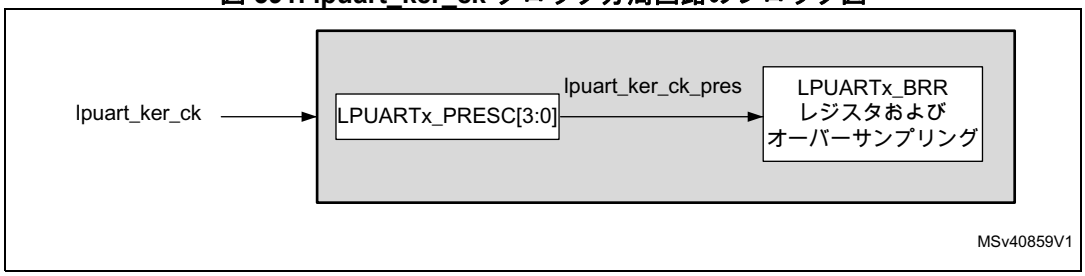
- LPUART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、lpuart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、lpuart_ker_ck クロックソースは RCC で設定できます（*リセットおよびクロック制御（RCC）のセクション*を参照）。そうしない場合、lpuart_ker_ck クロックは lpuart_pclk と同じになります。

lpuart_ker_ck クロックは、LPUART_PRESC レジスタのプログラム可能な係数によって分周できます。

図 591. lpuart_ker_ck クロック分周回路のブロック図



lpuart_ker_ck ソースによっては、LPUART は MCU が低電力モードのときにデータを受信することができます。受信データとウェイクアップモードの選択に応じて、LPUART は必要なときに MCU をウェイクアップして、ソフトウェアが LPUART_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、LPUART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは各着信ボーをボー周期のできるだけ中央でサンプリングします。各着信ボーについて 1 つのサンプルだけが採取されます。

注： ***データのノイズ検出は行われません。***

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから LPUART_RDR レジスタへ転送されます。
- 1 バイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNEビットと同時に立ち上がります。マルチバッファ通信の場合、LPUART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

FE ビットは、LPUART_ICR レジスタの FECF に “1”を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、LPUART_CR2 の制御ビットを通じて設定でき、通常モードでは 1 または 2 にできます。

- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- **2 個のストップビット**：2 個のストップビットのサンプリングは、2 番目のストップビットの中央で行われます。RXNE および FE フラグは、このサンプルの直後、すなわち 2 番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

49.3.7 LPUART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、LPUART_BRR レジスタでプログラムされた値に設定されます。

$$\text{Tx/Rx ボー} = \frac{256 \times \text{lpuartckpres}}{\text{LPUARTDIV}}$$

LPUARTDIV は、LPUART_BRR レジスタで定義されます。

注： ***ボーカウンタは、LPUART_BRR への書き込み操作後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。***

LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

f_{ck} は、3 x ボーレートから 4096 x ボーレートの範囲内でなければなりません。

LPUART クロックソースが LSE の場合に達成できる最大ボーレートは、9600 ボーです。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。たとえば、LPUART クロックソース周波数が 100 MHz の場合、達成できる最大ボーレートは約 33 M ボーです。

参考資料

表 382. lpuart_ker_ck_pres = 32,768 KHz でプログラムされたボーレートのエラー計算

ボーレート		lpuart_ker_ck_pres= 32,768 KHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差 =(計算値 - 目標値)B レート / 目標のB レート
1	0.3 KBps	0.3 KBps	0x6D3A	0
2	0.6 KBps	0.6 KBps	0x369D	0
3	1200 Bps	1200.087 Bps	0x1B4E	0.007
4	2400 Bps	2400.17 Bps	0xDA7	0.007
5	4800 Bps	4801.72 Bps	0x6D3	0.035
6	9600 KBps	9608.94 Bps	0x369	0.093

表 383. f_{CK} = 100 MHz でプログラムされたボーレートのエラー計算

ボーレート		f_{CK} = 100MHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差 =(計算値 - 目標値)B レート / 目標のB レート
1	38400 ボー	38400,04 ボー	A2C2A	0,0001
2	57600 ボー	57600,06 ボー	6C81C	0,0001
3	115200 ボー	115200,12 ボー	3640E	0,0001
4	230400 ボー	230400,23 ボー	1B207	0,0001
5	460800 ボー	460804,61 ボー	D903	0,001
6	921600 ボー	921625,81 ボー	6C81	0,0028
7	4000 K ボー	4000000,00 ボー	1900	0
8	10000 K ボー	10000000,00 ボー	A00	0
9	20000 K ボー	20000000,00 ボー	500	0
10	30000 K ボー	33032258,06 ボー	307	0,1

49.3.8 クロック偏差に対する LPUART レシーバの許容誤差

LPUART の非同期レシーバは、クロックシステムの合計偏差が LPUART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{LPUART レシーバの許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

LPUART レシーバは、*表 384* で指定された最大許容偏差まで、データを正しく受信できます。

- LPUART_CR2 レジスタの STOP[1:0] ビットによって定義されたストップビットの数
- LPUART_BRR レジスタの値

表 384. LPUART レシーバの許容誤差

M ビット	768 < BRR < 1024	1024 < BRR < 2048	2048 < BRR < 4096	4096 ≤ BRR
8 ビット（M=“00”）、ストップビット 1 個	1.82%	2.56%	3.90%	4.42%
9 ビット（M=“01”）、ストップビット 1 個	1.69%	2.33%	2.53%	4.14%
7 ビット（M=“10”）、ストップビット 1 個	2.08%	2.86%	4.35%	4.42%
8 ビット（M=“00”）、ストップビット 2 個	2.08%	2.86%	4.35%	4.42%
9 ビット（M=“01”）、ストップビット 2 個	1.82%	2.56%	3.90%	4.42%
7 ビット（M=“10”）、ストップビット 2 個	2.34%	3.23%	4.92%	4.42%

注： ***表 384 で指定されたデータは、M ビット = “00”のとき、受信フレームに正確に 10 ビット時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M ビット = “01”のときには 11 ビット時間、または M ビット = “10”のときには 9 ビット時間）。***

49.3.9 LPUART マルチプロセッサ通信

LPUART のマルチプロセッサ通信が可能です（ネットワーク内で複数の LPUART を接続して）。たとえば、1 つの LPUART をマスタとして、その TX 出力を別の LPUART の RX 入力に接続することができます。別の LPUART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する LPUART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUART_CR1 レジスタの MME ビットをセットする必要があります。

注： ***FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 lpuart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。***

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- LPUART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、LPUART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

LPUART は、LPUART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

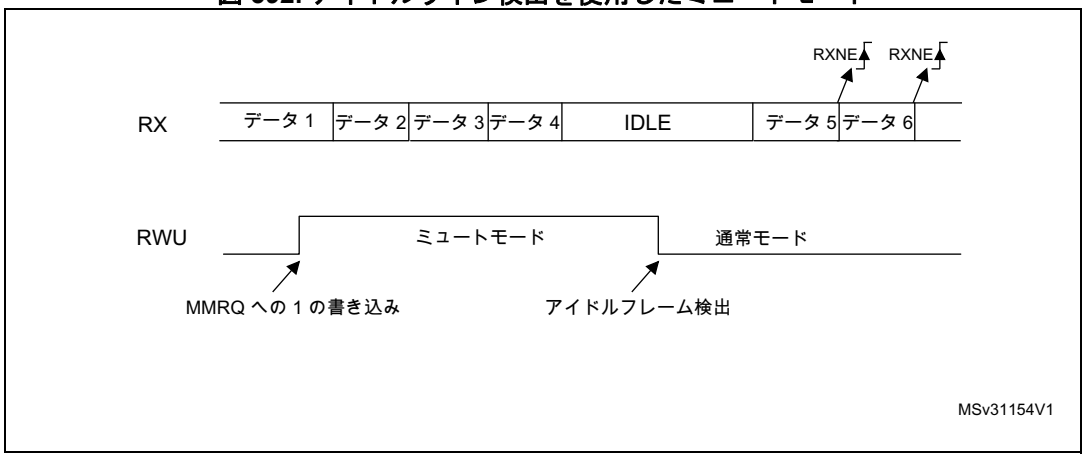
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=“0”）

MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、LPUART はミュートモードに入ります。

LPUART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、LPUART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を *図 592* に示します。

図 592. アイドルライン検出を使用したミュートモード



注： ***IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。***

ラインが IDLE のときに LPUART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビットアドレスマーク検出（WAKE="1"）

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、LPUART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。***

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。LPUART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

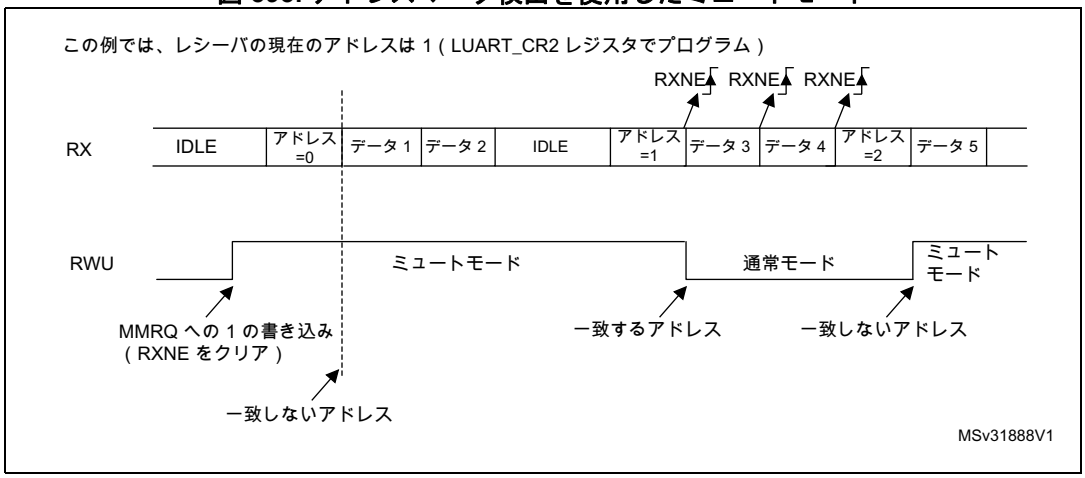
MMRQ ビットに"1"が書き込まれたときにも、LPUART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： ***FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ ビットがセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。***

アドレスマーク検出を使用したミュートモードの動作例を *図 593* に示します。

図 593. アドレスマーク検出を使用したミュートモード



49.3.10 LPUART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティ検査）を有効にするには、LPUART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な LPUART フレームフォーマットを *表 385* に示します。

表 385. LPUART フレームのフォーマット

M ビット	PCE ビット	LPUART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。
2. データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合（LPUART_CR1 の PS ビット =“0”）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合（LPUART_CR1 の PS ビット =“1”）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、LPUART_ISR レジスタの PE フラグがセットされ、LPUART_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、LPUART_ICR レジスタの PECF に“1”を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUART_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=“0”）は偶数個の「1」、奇数パリティが選択された場合（PS=“1”）は奇数個の「1」）。

49.3.11 LPUART 単線半二重通信

単線半二重モードを選択するには、LPUART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- LPUART_CR2 レジスタの LINEN および CLKEN ビット
- LPUART_CR3 レジスタの SCEN および IREN ビット

LPUART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の LPUART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： ***LPUART 通信では、1 個のストップビット設定の場合、ストップビットの中央で RXNE フラグがセットされます。***

49.3.12 DMA および LPUART を使用した連続通信

LPUART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： ***DMA モードがサポートされるかどうかについては、セクション 48.4：1948 ページのUSART の実装を参照してください。DMA がサポートされない場合は、セクション 48.5.6 の説明に従って LPUSRT を使用してください。連続通信を行うには。FIFO が無効のとき、LPUART_ISR レジスタの TXE/RXNE フラグをクリアします。***

DMA を使用した送信

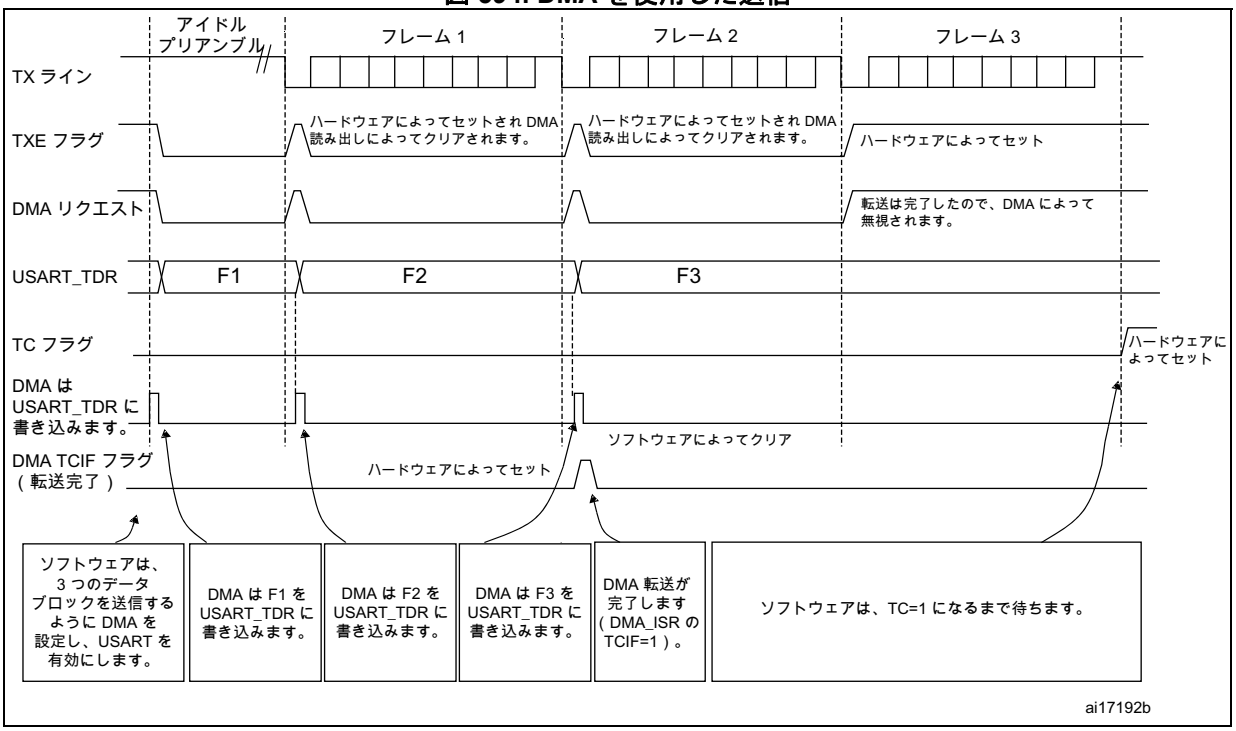
DMA モードでの送信を有効にするには、LPUART_CR3 レジスタの DMAT ビットをセットします。TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされるたびに、データは、DMA ペリフェラル（*セクション 15：581 ページのダイレクトメモリアクセスコントローラ（DMA1、DMA2）* および *セクション 16：614 ページの基本ダイレクトメモリアクセスコントローラ（BDMA）* を参照）を使用して設定された SRAM 領域から LPUART_TDR レジスタにロードされます。DMA チャネルを LPUART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

1. DMA 制御レジスタに LPUART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から LPUART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割り込み生成を設定します。
6. LPUART_ICR レジスタの TCCF ビットをセットすることによって、LPUART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して LPUART 通信の完了を確認することができます。これは、LPUART を無効にしたり 低電力モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC="1"になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 594. DMA を使用した送信



*注： **FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = "1"）によってトリガされます。***

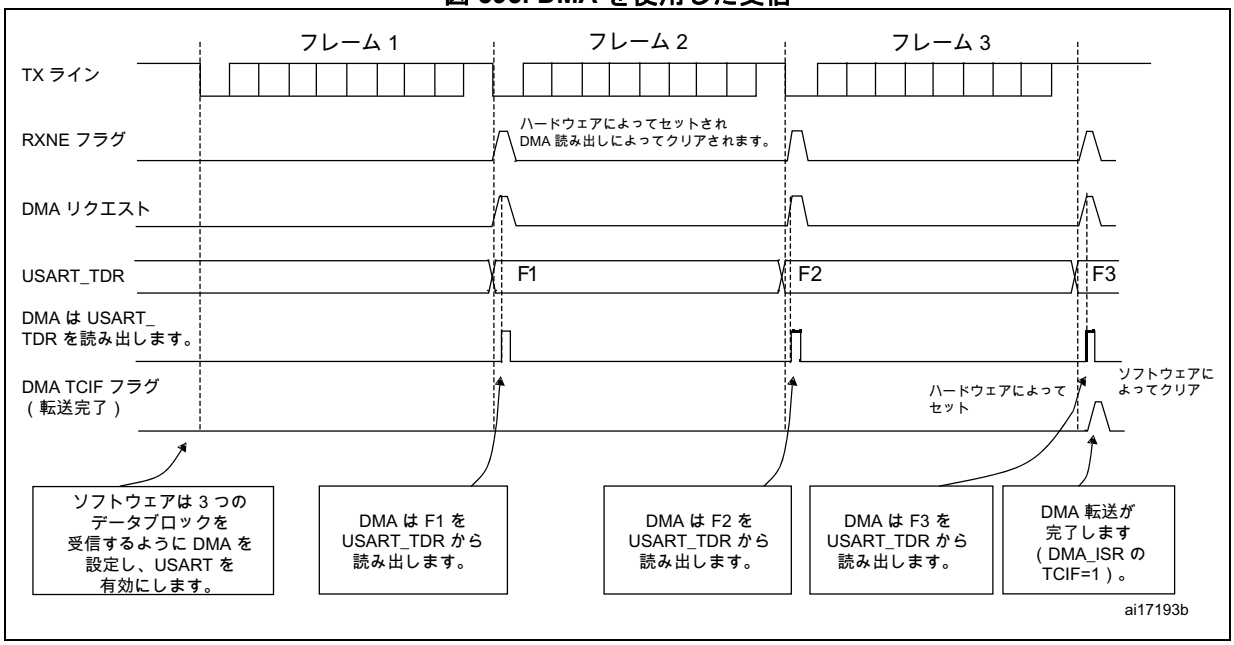
DMA を使用した受信

DMA モードでの受信を有効にするには、LPUART_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、LPUART_RDR レジスタから、DMA ペリフェラル（*セクション 15：581 ページのダイレクトメモリアクセスコントローラ（DMA1、DMA2）*および*セクション 16：614 ページの基本ダイレクトメモリアクセスコントローラ（BDMA）*を参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを LPUART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに LPUART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、LPUART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

図 595. DMA を使用した受信



注： ***FIFO 管理が有効になっているときは、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = “1”）によってトリガされます。***

マルチバッファ通信における割り込み生成とエラーフラグ

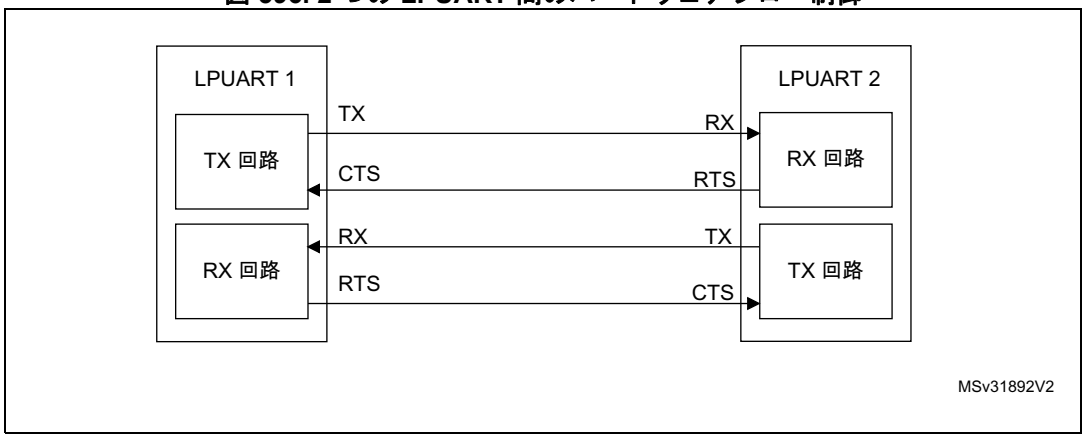
マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り

込み有効ビット（LPUART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

49.3.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。*図 582* に、このモードで 2 つのデバイスを接続する方法を示します。

図 596. 2 つの LPUART 間のハードウェアフロー制御

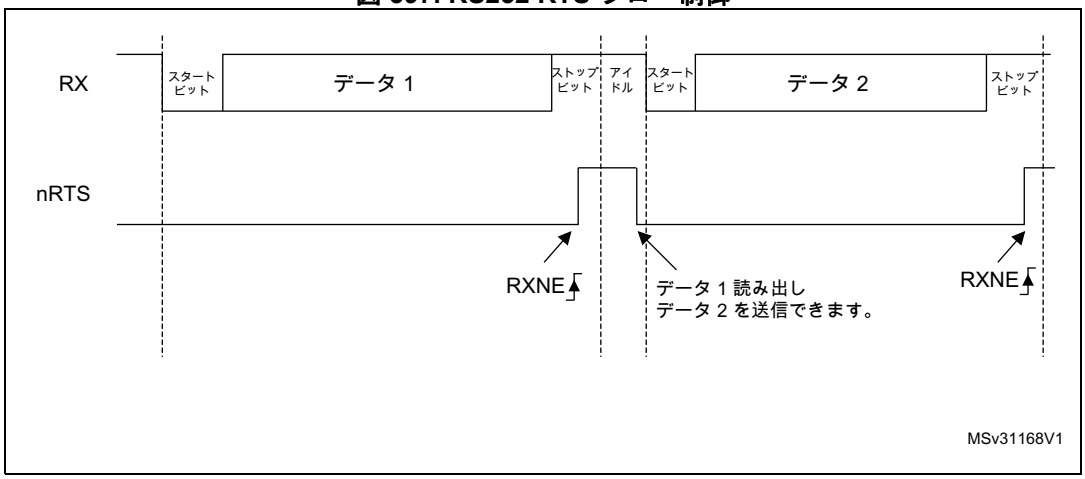


RS232 RTS と CTS のフロー制御は、LPUART_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=“1”）、LPUART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます（ローレベル接続）。受信レジスタがフルになると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。*図 597* に、RTS フロー制御が有効な場合の通信例を示します。

図 597. RS232 RTS フロー制御



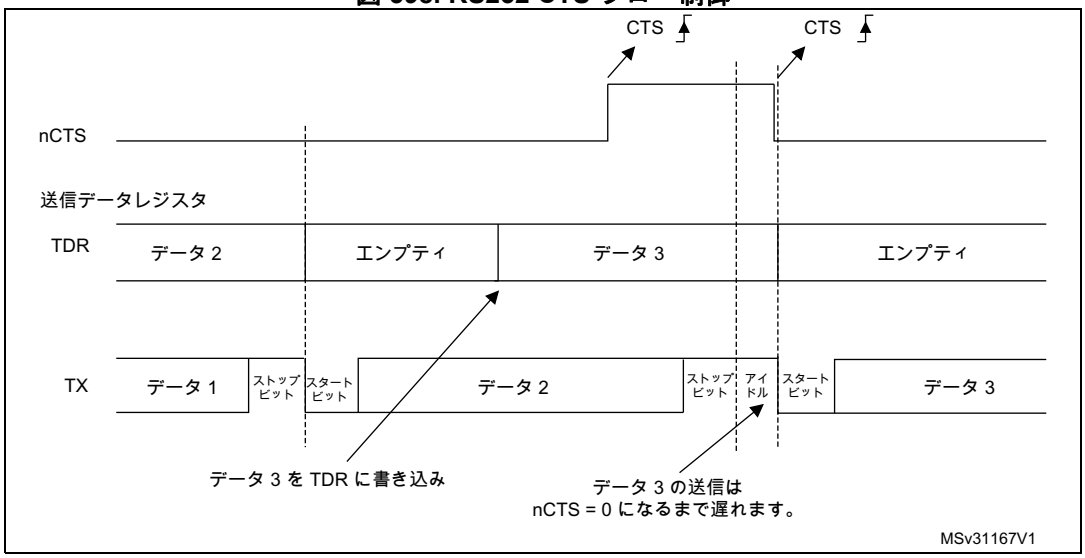
注： *FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、nRTS がネゲートされます。*

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=“1”）、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE/TXFE=“0”の場合）。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=“1”の場合、nCTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUART_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。*図 598* に、CTS フロー制御が有効な場合の通信例を示します。

図 598. RS232 CTS フロー制御



注： ***正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 LPUART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。***

RS485 ドライバ有効

ドライバ有効機能を有効にするには、LPUART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。LPUART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。LPUART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、LPUART_CR3 制御レジスタの DEP ビットを使用して設定できます。

LPUART の DEAT および DEDT は LPUART クロックソース（f_{CK}）サイクルで表されます。

- ドライバ有効アサーション時間は次のようになります。
  - (1 + (DEAT x P)) x f_{CK}、P ≠ 0 の場合
  - (1 + DEAT) x f_{CK}、P = 0 の場合
- ドライバ有効ネゲート時間は次のようになります。
  - (1 + (DEDT x P)) x f_{CK}、P ≠ 0 の場合
  - (1 + DEDT) x f_{CK}、P = 0 の場合

ここで、P = BRR[20:11]

49.3.14 LPUART 低電力管理

LPUART には、高度な低電力モード機能があり、lpuart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

LPUART は、UESM ビットがセットされているとき、MCU を低電力モードからウェイクアップできます。

Usart_pclk がゲートされているとき、**usart_pclk** クロックの有効化を必要とする特定の動作が必要になった場合、LPUART はウェイクアップ割り込み（**usart_wkup**）を生成します。

- FIFO モードが無効の場合

  LPUART データレジスタを空にするために luart_pclk クロックを有効にする必要があります。

  この場合、lpuart_wkup 割り込みのソースは“1”にセットされた RXNE です。低電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために luart_pclk クロックを有効にする必要があります。

  – TXFIFO を満たすため
  – または RXFIFO を空にするため

  この場合、lpuart_wkup 割り込みのソースになる可能性のあるものは以下の通りです。

  – RXFIFO ノットエンプティ。この場合、低電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  – RXFIFO フル。この場合、低電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  – TXFIFO は空です。この場合、低電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、lpuart_wkup 割り込みソースになり得るのは次のイベントのうちの 1 つです。

  – TXFIFO 閾値に達した。この場合、低電力モードに入る前に TXFTIE ビットをセットする必要があります。
  – RXFIFO 閾値に達した。この場合、低電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信する時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFOノットエンプティ、および RXFIFO/TXFIFO 閾値割り込みを使用すれば、低電力モード中にできるだけ多くの LPUART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の **lpuart_wkup** 割り込みを選択することもできます。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は **lpuart_wkup** 割り込みが生成されます。この場合、ウェイクアップのために **lpuart_wkup** 割り込みは必須ではありません。低電力モードから MCU をウェイクアップするには、WUF がセットされるだけで十分です。

注： ***低電力モードに移行する前に、LPUART 転送が進行中ではないことを確認してください。BUSY フラグをチェックすることでは、データ受信中に低電力モードに入らないことを保証できません。***

WUF フラグは、MCU が 低電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低電力モードに入るときには、REACK ビットをチェックして、LPUART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、低電力モードに入る前に無効化し、低電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低電力モードでのミュートモードの使用

低電力モードに入る前に LPUART がミュートモードになった場合は、

- アイドル検出は低電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： ***FIFO 管理が有効なとき、ミュートモードは何の制約もなく低電力モードからのウェイクアップとともに使用されます（すなわち、ミュートおよび低電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。***

低電力モードで LPUART カーネルクロック（lpuart_ker_ck）がオフのときの低電力モードからのウェイクアップ

低電力モード中、lpuart_ker_ck クロックがオフになっている場合、LPUART 受信ラインの立ち下がりエッジが検出されると、lpuart_ker_ck_req 信号によって LPUART インタフェースが lpuart_ker_ck クロックをオンにするようリクエストします。その後、lpuart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップイベントが確認されない場合、usart_ker_ck クロックが再度オフになり、MCU がウェイクアップせずに低電力モードに留まり、カーネルクロックリクエストが解除されます。

以下の例は、ウェイクアップイベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 599 に、ウェイクアップイベントが確認された時の 動作を示します。

図 599. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

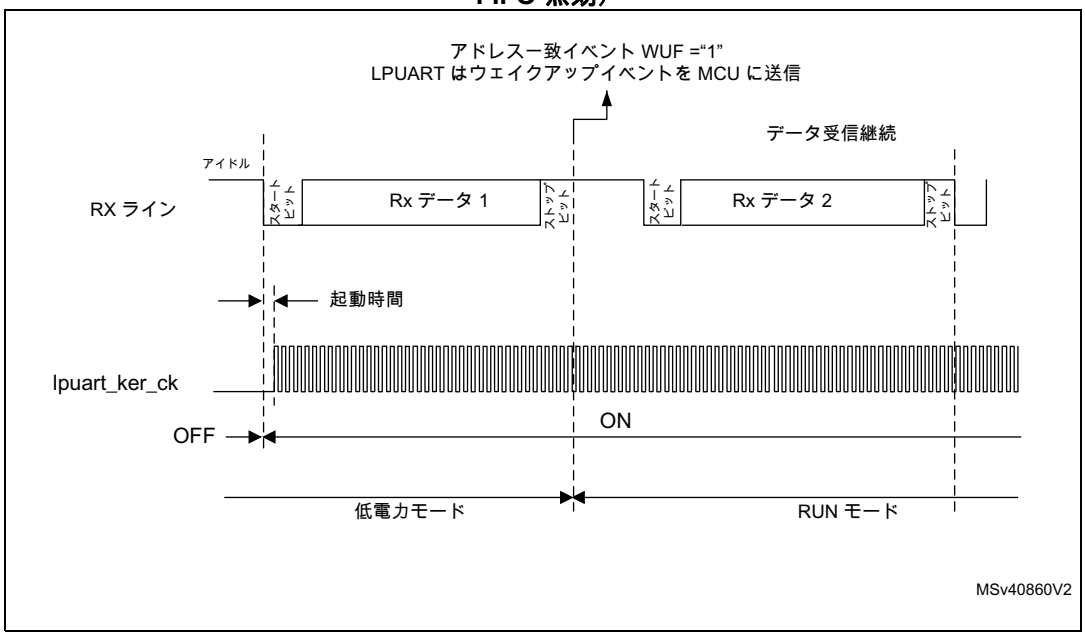
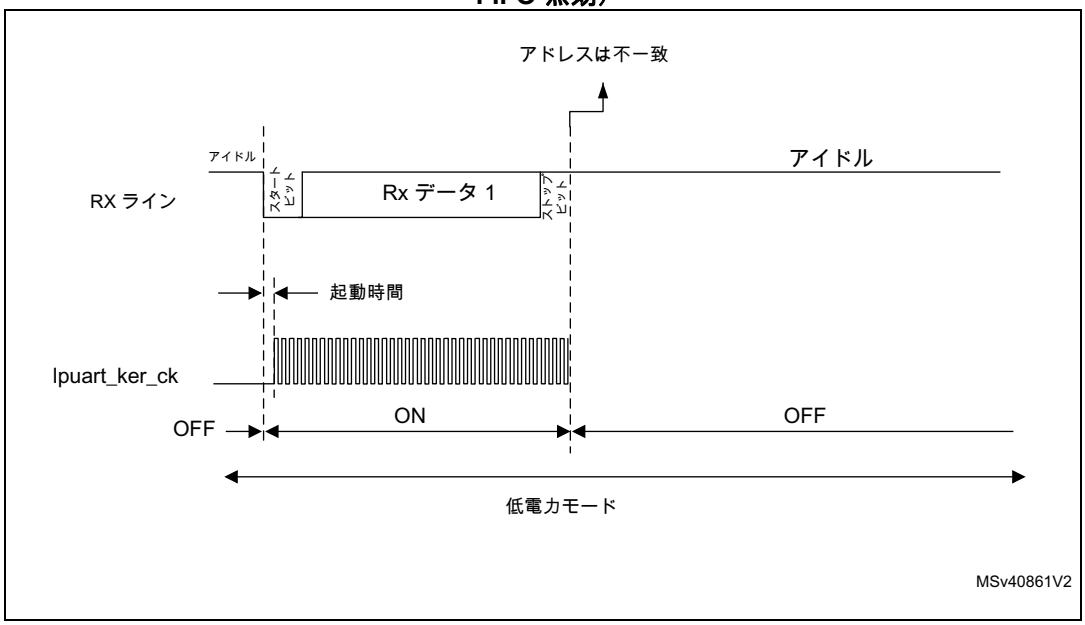


図 600 に、ウェイクアップイベントが確認されない時の動作を示します。

図 600. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： ***上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、LPUART はスタートビットの終端にウェイクアップイベントを MCU に送ります。***

49.4 LPUART 割り込み

すべての LPUART 割り込みリクエストの詳細な説明については、*表 380* を参照してください。

表 386. LPUART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット	割り込みのクリア方法	有効になる割り込み	
				lpuart_it	lpuart_wkup
送信データレジスタエンプティ	TXE	TXEIE	TXE は、データが TDR に書き込まれたときにクリアされます。	はい	いいえ
送信 FIFO ノットフル	TXFNF	TXFNFIE	TXFNF は、TXFIFO がフルになると、クリアされます。	はい	いいえ
送信 FIFO エンプティ	TXFE	TXFEIE	TXFE は、TXFIFO に少なくとも 1 つのデータが入ったとき、または TXFRQ ビットのセットによってクリアされます。	はい	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
CTS 割り込み	CTSIF	CTSIE	CTSIF は、CTSCF ビットをセットすることによって、ソフトウェアによってクリアされます。	はい	いいえ
送信完了	TC	TCIE	TC は、データが TDR に書き込まれたとき、または TCCF ビットのセットによって、クリアされます。	はい	いいえ
受信データレジスタノットエンプティ（データの読み出し可能）	RXNE	RXNEIE	RXNE は、RDR の読み出しによって、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFNE は、RXFIFO が空になったとき、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO フル	RXFF(1)	RXFFIE	RXFF は、RXFIFO に少なくとも 1 つのデータが入ったとき、クリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
オーバーランエラー検出	ORE	RXNEIE/ RXFNEIE	ORE は、ORECF ビットをセットすることによってクリアされます。	はい	いいえ
アイドルライン検出	IDLE	IDLEIE	IDLE は、IDLECF ビットをセットすることによってクリアされます。	はい	いいえ
パリティエラー	PE	PEIE	PE は、PECF ビットをセットすることによってクリアされます。	はい	いいえ

表 386. LPUART 割り込みリクエスト（続き）

割り込みイベント	イベントフラグ	イネーブル制御ビット	割り込みのクリア方法	有効になる割り込み	
				lpuart_it	lpuart_wkup
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。	NE または ORE または FE	EIE	NE は、NFCF ビットをセットすることによってクリアされます。 ORE は、ORECF ビットをセットすることによってクリアされます。 FE フラグは、FECF ビットをセットすることによってクリアされます。	はい	いいえ
キャラクター致	CMF	CMIE	CMF は、CMCF ビットをセットすることによってクリアされます。	はい	いいえ
低電力モードからのウェイクアップ	WUF(2)	WUFIE	WUF は、WUCF ビットをセットすることによってクリアされます。	いいえ	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

1. RXFF フラグは、LPUART が 次のように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。RXFIFO に n 個のデータ、LPUART_RDR に 1 個のデータ。STOP モードでは、LPUART_RDR はクロック供給されません。その結果として、このレジスタは書き込まれず、n 個のデータが受信されて RXFIFO に書き込まれた後に、RXFF 割り込みがアサートされます（RXFF フラグはセットされません）。
2. WUF 割り込みは、低電力モードでのみアクティブです。

49.5 LPUART レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

49.5.1 制御レジスタ 1（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIETXFNFIE	TCIE	RXNEIERXFNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで RXFF=“1”のときに LPUART 割り込みが生成されます。

注： FIFO モードが無効のときは、このビットは予約済みであり、リセット値に保持する必要があります。

ビット 30 **TXFEIE**：TXFIFO エンプティ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで TXFE=“1”のときに LPUART 割り込みが生成されます。

注： FIFO モードが無効のときは、このビットは予約済みであり、リセット値に保持する必要があります。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

注： 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。Lpuart_ker_ck クロックサイクルで表されます。詳細については、*セクション 48.5.20：RS232 ハードウェアフロー制御および RS485 ドライバ有効*を参照してください。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、*セクション 49.3.13：RS232 ハードウェアフロー制御および RS485 ドライバ有効*を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割り込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=“1”の場合は 9 番目のビット、M=“0” の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで PE=“1”のときには LPUART 割り込みが生成されます。

ビット 7 **TXEIE/TXFNFIE**：送信データレジスタエンプティ／TXFIFO ノットフル割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで TXE/TXFNF=“1”のときには LPUART 割り込みが生成されます。

ビット 6 **TCIE**：転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで TC=“1”のときには LPUART 割り込みが生成されます。

ビット 5 **RXNEIE/RXFNEIE**：受信データレジスタノットエンプティ／RXFIFO ノットエンプティ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの ORE=“1”または RXNE/RXFNE=“1”のときには、LPUART 割り込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで IDLE=“1”のときには LPUART 割り込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： ***送信中に TE ビットにローパルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。***

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を低電力モードからウェイクアップできません。

このビットがセットされると、LPUART は MCU を低電力モードからウェイクアップできますが、LPUART クロック選択が RCC にて HSI または LSE であることが条件です。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低電力モードから MCU をウェイクアップできません。

1：LPUART は低電力モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロックソースは HSI または LSE でなければなりません（RCC の章を参照）。

注： ***低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。***

ビット 0 **UE**：LPUART 有効

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低電力モードです。

1：LPUART は有効です。

注： ***ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをその前にリセットする必要があり、ソフトウェアは、UE ビットをリセットする前にLPUART_ISR の TC ビットがセットされるのを待つ必要があります。***

UE=“0”のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

49.5.2 制御レジスタ 2（LPUART_CR2）

アドレスオフセット：0x04

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:4]				ADD[3:0]				Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
スワップ	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
rw		rw	rw								rw				

ビット 31:28 **ADD[7:4]**：LPUART ノードのアドレス

このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

ミュートモードまたは Stop モード中に、マルチプロセッサ通信で 7 ビットアドレスマーク検出によって MCU をウェークアップするために使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが非アクティブなときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のためにも使用することができます。この場合、受信されたキャラクタ全体（8 ビット）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき（RE=“0”）または LPUART が無効のとき（UE=“0”）のみ、書き込むことができます。

ビット 27:24 **ADD[3:0]**：LPUART ノードのアドレス

このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

これは、ミュートモードまたは低電力モード中に、マルチプロセッサ通信でアドレスマーク検出によってウェイクアップするために使用されます。

このビットフィールドは、受信が無効のとき（RE=“0”）または LPUART が無効のとき（UE=“0”）のみ、書き込むことができます。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TX ピン信号値は反転されます。（(V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RX ピン信号値は反転されます。（(V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 15 **スワップ**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **STOP[1:0]**：STOP ビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：予約済み。

10：2 個のストップビット

11：予約済み

このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 11:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

注： 7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

49.5.3 制御レジスタ 3（LPUART_CR3）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFTCFG			RXFTIE	RXFTCFG			Res.	TXFTIE	WUFIE	WUS [2:0]		Res.	Res.	Res.	Res.
rw			rw	rw				rw	rw	rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

ビット 31:29 **TXFTCFG**：TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
110：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済み。

Bit28 **RXFTIE**：RXFIFO 閾値割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、LPUART 割り込みが生成されます。

ビット 27:25 **RXFTCFG**：受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
110：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済み。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TXFTIE**：TXFIFO 閾値割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、LPUART 割り込みが生成されます。

ビット 22 **WUFIE**：低電力モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割り込みは禁止されています。
1：LPUART_ISR レジスタで WUF=“1”のときには LPUART 割り込みが生成されます。

注： ***WUFIE は、低電力モードに入る前にセットする必要があります。***
WUF 割り込みは、低電力モードでのみアクティブです。
LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4 : USART の実装を参照してください。

ビット 21:20 **WUS [1:0]**：低電力モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低電力モードからのウェイクアップフラグ）。
00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。
01：予約済み
10：WUF はスタートビット検出時に有効になります。
11：WUF は RXNE 時に有効になります。
このビットフィールドは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

注： ***LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4 : USART の実装を参照してください。***

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR=“0”）、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが LPUART_RDR レジスタの前の内容に上書きされます。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTSIE**：CTS 割り込み有効

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの CTSIF=“1”のときには、割り込みが生成されます。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは nCTS 入力がアサート（0 に関係）されている場合にのみ転送されます。データ転送中に nCTS 入力がネゲートされると、送信は停止前に完了します。nCTS がアサートされている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、nRTS 出力がアサートされます（0 にプルされます）。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EIE**：エラー割り込みイネーブル

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（LPUART_ISR レジスタの FE=“1”または ORE=“1”または NE=“1”）の場合に割り込み生成を有効にするために必要です。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの FE=“1”、ORE=“1”、または NE=“1”のときに、割り込みが生成されます。

49.5.4 ボーレートレジスタ（LPUART_BRR）

このレジスタは、LPUART が無効（UE=“0”）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **BRR[19:0]**

注： ***LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。***

ただし、LPUART_BRR は ≥ 0x300 でなければならず、LPUART_BRR は 20 ビットであり、高い fck 値を使用して高いボーレートを生成するときには、注意が必要です。fck は [3 x ボーレート～4096 x ボーレート] の範囲内でなければなりません。

49.5.5 リクエストレジスタ（LPUART_RQR）

アドレスオフセット：0x18

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
											w	w	w	w	

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

このビットは FIFO モードが有効なときに使用されます。FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（TXFIFO エンプティ、LPUART_ISR レジスタのビット 23）がセットされます。

注： FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに“1”を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、LPUART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに“1”を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 予約済みであり、リセット値に保持する必要があります。

49.5.6 割り込みおよびステータスレジスタ（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x00C0（FIFO が無効の場合）

リセット値：0x08000C0（FIFO が有効の場合）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFE	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
				r	r		r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは、TXFIFO が LPUART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの TXFTIE ビット =“1”（ビット 31）の場合、割り込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、RXFIFO が LPUART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したとき、すなわち、受信 FIFO に RXFTCFG 分のデータが入ったときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの RXFTIE ビット =“1”（ビット 27）の場合、割り込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **RXFF**：RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、LPUART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

LPUART_CR1 レジスタの RXFFIE ビット =“1”である場合、割り込みが生成されます。

0：RXFIFO はフルではありません。

1：RXFIFO はフルです。

ビット 23 **TXFE**：TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、LPUART_RQR レジスタのビット TXFRQ（ビット 4）に“1”を書き込むことによってセットすることもできます。

LPUART_CR1 レジスタの TXFEIE ビット =“1”（ビット 30）の場合、割り込みが生成されます。

0：TXFIFO は空ではありません。

1：TXFIFO は空です。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： ***LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE=“0”を書き込んだ後、TE=“1”を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=“0”の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE=“1”である場合、割り込みが生成されます。

注： ***UESM がクリアされると、WUF フラグもクリアされます。***

WUF 割り込みは、低電力モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに“1”を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに“1”を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタは送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE=“1”である場合、割り込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 **CTSIF**：CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

また、LPUART_CR3 レジスタで CTSIE=“1”であれば、割り込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXE/TXFNF**：送信データレジスタエンプティ／TXFIFO ノットフル

FIFO モードが無効のとき、TXE は、LPUART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUART_TDR レジスタへの書き込みによってクリアされます。

FIFO モードが有効のとき、TXFNF は、TXFIFO がフルでないときに、ハードウェアによってセットされるため、データを LPUART_TDR に書き込むことができます。LPUART_TDR への書き込みごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを LPUART_TDR に書き込むことができないことを示します。

注：一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです。（TXFNF および TXFE は同時にセットされます）。

LPUART_CR1 レジスタの TXEIE/TXFNFIE ビット =“1”である場合、割り込みが生成されます。

0：データレジスタはフル／送信 FIFO はフルです。

1：データレジスタ／送信 FIFO はフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXE/TXFF がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE=“1”である場合、割り込みが生成されます。LPUART_ICR レジスタの TCCF に“1”を書き込むことによって、または LPUART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの TCIE=“1”である場合、割り込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE/RXFNE**：読み出しデータレジスタノットエンプティ／RXFIFO ノットエンプティ

RXNE ビットは、LPUART_RDR シフトレジスタの内容が LPUART_RDR レジスタに転送されると、ハードウェアによってセットされます。LPUART_RDR レジスタの読み出しによってクリアされます。

RXNE フラグは、LPUART_RQR レジスタの RXFRQ に“1”を書き込むことによってもクリアすることもできます。

RXFNE ビットは、RXFIFO が空でなく、したがってデータが LPUART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。LPUART_RDR からの読み出しのたびに、RXFIFO の 1 つの場所が解放されます。RXFIFO が空になると、クリアされます。

RXNE/RXFNE フラグは、LPUART_RQR レジスタの RXFRQ に“1”を書き込むことによってクリアすることもできます。

LPUART_CR1 レジスタの RXNEIE/RXFNEIE=“1”の場合、割り込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE=“1”である場合、割り込みが生成されます。LPUART_ICR レジスタの IDLECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合 (MME=“1”)、LPUART がミュートでない場合 (RWU=“0”)、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=“1”の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、シフトレジスタで現在受信中のデータを LPUART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの RXNEIE/ RXFNEIE=“1”または EIE=“1”の場合、割り込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：スタートビットノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NFCF ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割り込みを生成する RXNE/RXFNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割り込みが生成されます。

FIFO モードでは、このエラーは LPUART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに“1”を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： FIFO モードでは、このエラーは LPUART_RDR 内のキャラクタに関連します。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に“1”を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの PEIE=“1”である場合、割り込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： FIFO モードでは、このエラーは LPUART_RDR 内のキャラクタに関連します。

49.5.7 割り込みフラグクリアレジスタ（LPUART_ICR）

アドレスオフセット：0x20

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.
											w_r0			w_r0	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
						w_r0			w_r0		w_r0	w_r0	w_r0	w_r0	w_r0

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低電力モードからのウェイクアップフラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの WUF フラグがクリアされます。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 48.4 : USART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの CTSIF フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TCCF**：送信完了フラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア

このビットに“1”を書き込むと、LPUART_ISR レジスタの PE フラグがクリアされます。

49.5.8 受信データレジスタ（LPUART_RDR）

アドレスオフセット：0x24

リセット値：未定義

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（*図 587* を参照）。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

49.5.9 送信データレジスタ（LPUART_TDR）

アドレスオフセット：0x28

リセット値：未定義

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（*図 587* を参照）。

パリティを有効にして（LPUART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF="1"のときのみ書き込む必要があります。

49.5.10 プリスケーラレジスタ（LPUART_PRESC）

このレジスタは、LPUART が無効（UE="0"）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

LPUART 入力クロックは、以下のようにプリスケーラによって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済み。

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は «1011» になります。すなわち、入力クロックは 256 分周されます。

49.5.11 LPUART レジスタマップ

次の表に、LPUART のレジスタマップとリセット値を示します。

表 387. LPUART レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPUART_CR1	RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT4	DEAT3	DEAT2	DEAT1	DEAT0	DEDT4	DEDT3	DEDT2	DEDT1	DEDT0	Res.	CMIE	MME	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	LPUART_CR2	ADD[7:4]				ADD[3:0]				Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV	スワップ	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0		0	0								0				
0x08	LPUART_CR3	TXFT CFG			RXFTIE	RXFT CFG			Res.	TXFTIE	WUFIE	WUS[1:0]		Res.	Res.	Res.	Res.	DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
	リセット値	0	0	0	0	0	0	0		0	0	0	0					0	0	0	0		0	0	0	0	0			0			0
0x0C	LPUART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:0]																			
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10～0x14	予約済み																																
0x18	LPUART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
	リセット値																												0	0	0	0	
0x1C	LPUART_ISR	Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFF	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値					0	0		0	0	0	0	0	0	0	0	0						0	0		1	0	0	0	0	0	0	0
0x20	LPUART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0								0			0		0	0	0	0	0
0x24	LPUART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	LPUART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x2C	LPUART_PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

50 シリアルペリフェラルインタフェース（SPI）

50.1 概要

シリアルペリフェラルインタフェース（SPI）を使用して、特定の同期プロトコルを利用して外部デバイスと通信することができます。SPI インタフェースは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタまたはスレーブとして設定することが可能で、マルチスレーブまたはマルチマスタの設定で動作することができます。マスタの設定の場合、外部スレーブデバイスに通信クロック（SCK）を供給します。マスタからスレーブ選択信号を提供することができ、スレーブからも任意に受け入れることができます。デフォルトでモトローラのデータフォーマットが使用されますが、その他の特定のモードもサポートされています。

50.2 SPI の主な機能

- 3 本のラインでの全二重同期転送
- 2 本のラインでの半二重同期転送（双方向データライン有り）
- 2 本のラインでの単方向同期転送（単方向データライン有り）
- 4 ～ 32 ビットのデータサイズ選択
- マルチマスタまたはマルチスレーブモード機能
- PCLK　から独立したペリフェラルカーネル用の分離したクロックによるデュアルクロックドメイン
- 8個のマスタモードボーレートプリスケーラ（最大、カーネル周波数／2）
- 最大、カーネル周波数／2 までのスレーブモード周波数
- 構成と設定の保護
- マスタとスレーブの両方に対するハードウェア／ソフトウェアによる SS 管理
- データ間、および SS とデータフロー間の最小遅延を調節可能
- 設定可能な SS 信号の極性とタイミング、MISO と MOSI のスワップ機能
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- SS および CRC を制御するトランザクション内のデータ数をプログラム可能
- 専用の送受信フラグ（割り込み機能付き）
- ウェイクアップを伴う、STOP モード（ペリフェラルへのクロック供給なし）でのスレーブの送受信機能
- SPI のモトローラおよび TI フォーマットをサポート
- ハードウェアの CRC 機能によって、以下によりトランザクションの最後で確実な通信を確保
  - 送信モードで CRC 値の追加
  - 受信モードで CRC エラーの自動チェック
- マスタモードの障害、オーバーランまたはアンダーラン、および CRC エラー検出（割り込み機能付き）
- DMA 機能付きの 2つの 16x 8 または 8x 8 ビットの内蔵 RxFIFO および TxFIFO
- トランザクションでのプログラム可能なデータ数
- 設定可能な FIFO 閾値（データパッキング）
- スレーブのアンダーラン状態での設定可能な動作（カスケードサーキュラバッファ）

50.3 SPI の実装

表 388. STM32H7xx SPI の機能

SPI モード／機能	SPI2S1	SPI2S2	SPI2S3	SPI4	SPI5	SPI6
Rx および TxFIFO サイズ（N）[x 8 ビット]	16	16	16	8	8	8
設定可能最大データサイズ [ビット]	32	32	32	16	16	16
I2S の機能	有	有	有	無	無	無

50.4 SPI の機能説明

50.4.1 SPI ブロック図

SPI では、MCU と外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用の SPI 割り込みを使用することで、通信を管理することができます。SPI の主要要素およびそれらの相互作用を以下のブロック図（*図 601*）に示します。

図 601. SPI2S ブロック図

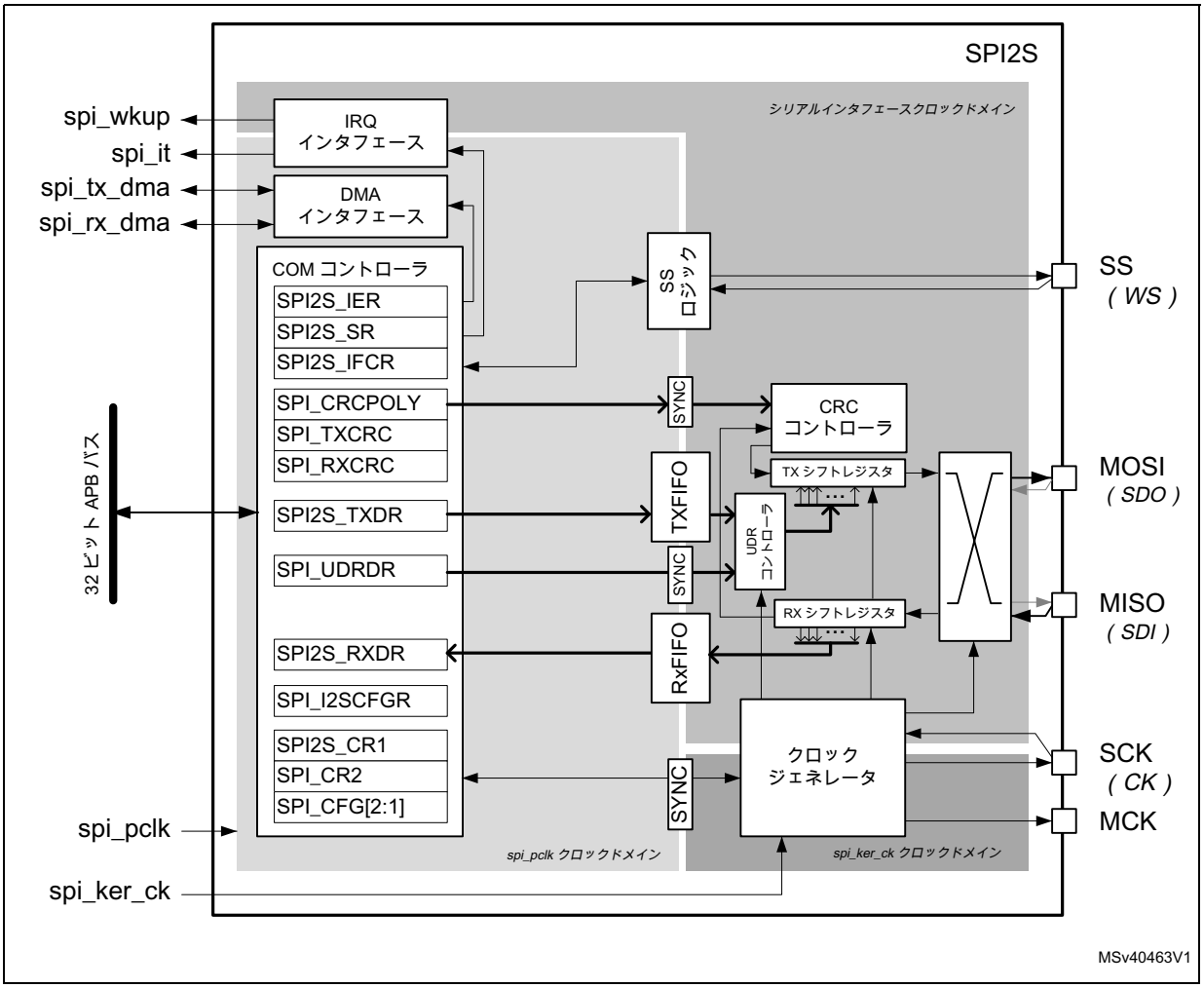


図 601 の簡易体系図には、以下の完全に独立した 3 つのクロックドメインが示されています。

- **spi_pclk** クロックドメイン
- **spi_ker_ck** カーネルクロックドメイン
- シリアルインタフェースクロックドメイン

これらのドメイン間のすべての制御およびステータス信号は厳密に同期されています。これらのクロック信号間の周波数の比に関しては特定の制約はありません。ユーザは、データのアンダーランやオーバーランイベントを避けるためだけに、データフロー速度と互換性のある周波数比を考慮することが必要です。

spi_pclk クロック信号はペリフェラルバスインタフェースにクロックを供給します。SPI レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

スレーブモードで動作している SPI は、外部マスタ SPI デバイスによって提供される外部 SCK 信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。このため、SPI スレーブは、**spi_pclk** および **spi_ker_ck** 信号が非アクティブになっている場合でもデータの送受信ができます。

SPI マスタでは、少なくともクロックジェネレータにクロックを供給するために、RCC から来るアクティブな **spi_ker_ck** カーネルクロック が必要なので、このことは当てはまりません。一方、シリア

ルインタフェースクロックドメイン内で動作する特定のスレーブのロジックでは、追加のトラフィックを正しく設定する必要があります（たとえば、アンダーランやオーバーランを評価する場合）。バスがアイドルになると、これを行うことはできません。特定のケースでは、スレーブはクロックジェネレータの動作も必要とします（*セクション 50.5.1 : TI モード*を参照）。

50.4.2 SPI 信号

4 本の I/O ピンが外部デバイスとの SPI 通信専用に使用されます。

- **MISO**：マスターイン／スレーブアウトデータ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使用されます。
- **MOSI**：マスターアウト／スレーブインデータ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使用されます。
- **SCK**：SPI マスタではシリアルクロックの出力に、SPI スレーブでは入力に使用されます。
- **SS**：スレーブ選択用のピンです。このピンは、SPI および SS の設定に応じて、以下のいずれかに使用できます。
  - 個々の通信用スレーブデバイスを選択する
  - データフレームを同期させる
  - 複数のマスタ間での競合を検出する

  詳細は、*セクション 50.4.6 : マルチマスタ通信*を参照してください。

SPI バスを使用することで、1 つのマスタデバイスと 1 つ以上のスレーブデバイスとの間で通信することができます。バスは 2 本以上の線から成り、1 本はクロック信号用、その他はデータの同期転送用です。SPI ノード間でのデータ交換とそれらのスレーブ選択信号管理に応じて、その他の信号を追加することができます。どの SPI モードでも、MOSI と MISO ピン間で機能を逆にすることができます（SPI_CFG2 レジスタの IOSWP ビットを参照）。

50.4.3 SPI 通信の一般的側面

SPI を使用することで、MCU は対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2 または 3 本の線（ソフトウェア SS 管理あり）、あるいは 3 または 4 本の線（ハードウェア SS 管理あり）が使われます。通信は常にマスタによって開始され制御されます。マスタは SCK ラインにクロック信号を供給し、SS ライン（ハードウェアによって管理される場合）によって通信相手のスレーブを選択したり、同期させたりします。マスタとスレーブ間のデータは MOSI および／または MISO ライン上を流れます。

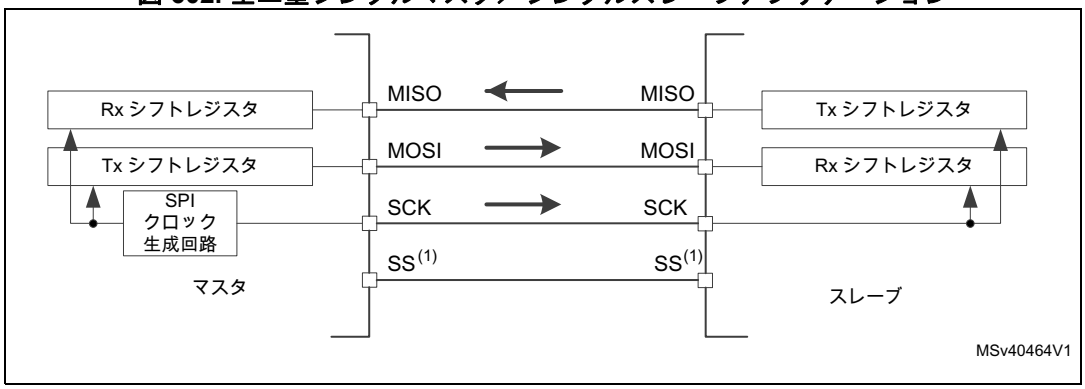
50.4.4 マスタとスレーブの 1 対 1 の通信

通信の流れは、全二重（3 線）、半二重（2 線）、単方向（2 線）の 3 つのモードのうちの 1 つを使用します。SS 信号は、シングルマスタ-スレーブ設定ではオプションであり、2 つの通信ノード間で接続されないことがしばしばあります。しかしながら、SS 信号は、この設定でデータフローを同期させるのに役立ち、特定の SPI モード（たとえば、TI モード）ではデフォルトで使用されます。

全二重通信

SPI は、デフォルトで全二重通信に設定されます（SPI_CFG2 レジスタのビット COMM[1:0]=00）。この設定では、マスタおよびスレーブのシフトレジスタは、MOSI ピンと MISO ピンの間に 2 本の単方向ラインを介してリンクされます。SPI 通信の間、データはマスタから供給される SCK クロックのエッジに同期してシフトされます。マスタは、送信すべきデータを MOSI ライン経由でスレーブに送信し、同時に MISO ライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図 602. 全二重シングルマスタ／シングルスレーブアプリケーション

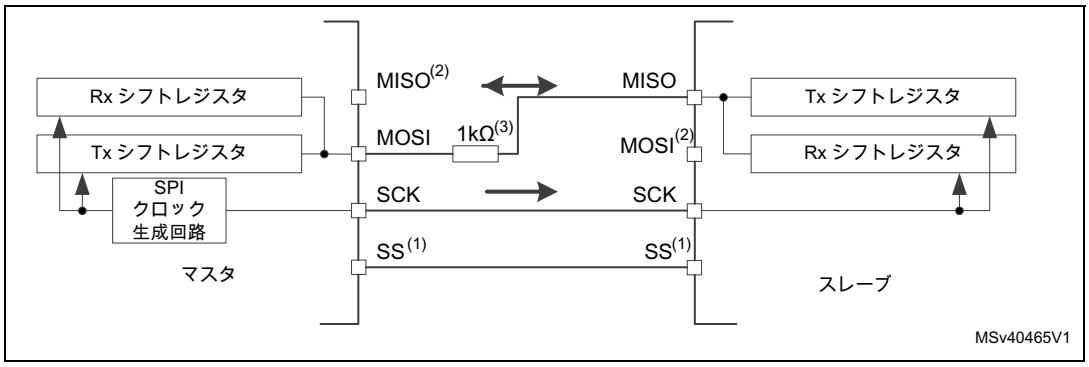


1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。

半二重通信

SPI_CFG2 レジスタで COMM[1:0]=11 にセットすることで、SPI は半二重モードで通信できます。この設定では、1 本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データは SCK クロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方が SPI_CR1 レジスタの HDDIR ビットを使用して相互に選択された転送方向となります。通信の方向を変えるときは SPI を無効にする必要があるので、注意してください。この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、他のアプリケーションで使用でき、GPIO として機能します。

図 603. 半二重シングルマスタ／シングルスレーブアプリケーション



1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。
2. この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、GPIO として使用できます。
3. 双方向モードで動作している 2 つのノード間で同期されず、通信方向が変更されて、新しいトランスミッタが共通のデータラインにアクセスし、前のトランスミッタが逆の値をライン上に保持している場合（値は SPI 設定および通信データによります）、重大な事態が発生する可能性があります。次のノードが対応して方向設定を変更するまで、ライン上に一時的に逆の出力レベルが提供され、両ノードが競合することがあります。このモードでは MISO ピンと MOSI ピンの間に直列抵抗を挿入して、出力を保護し、この状況で流れる電流を制限することをお奨めします。

単方向通信

SPI は、SPI_CFG2 レジスタの COMM[1:0] フィールドを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは 1 ラインのみです。残りの MISO ピンまたは MOSI ピンのペアは通信には使用されず、標準の GPIO として使用できます。

- **送信専用モード：COMM[1:0]=01**

  送信専用モードのマスタは、TxFIFO に準備できたデータがあり、マスタの転送が進行中である限り、クロックを生成します。

  送信専用モードのスレーブは、SCK ピンにクロックを受信し、SS ピン（またはソフトウェア管理の内部信号）がアクティブである限り、データを送信します（*50.4.6：マルチマスタ通信* を参照）。

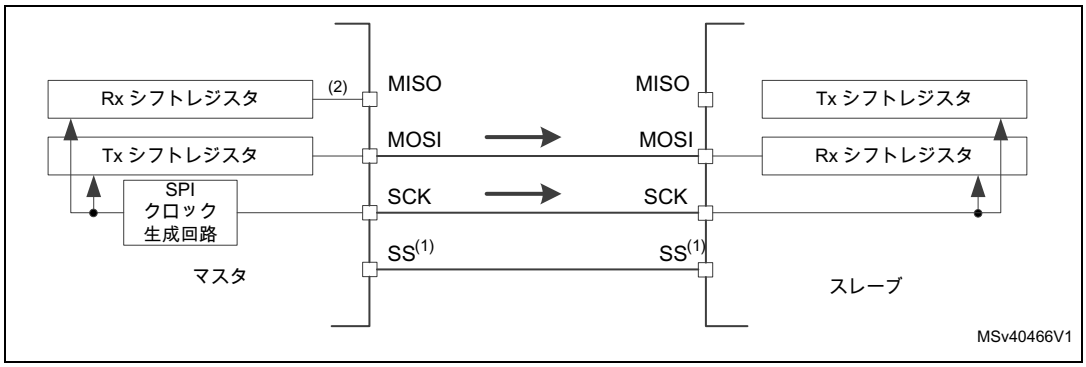
- **受信専用モード：COMM[1:0]=10**

  マスタモードでは、MOSI 出力が無効化され、GPIO として使用できます。SPI が有効で、SPI_CR1 レジスタの CSTART ビットがセットされている限り、クロック信号が生成され続けます。SPI_CR1 レジスタの CSUSP ビットをセットすることによってソフトウェアが明示的に停止を要求することによって、または SPI_CR1 の MASRX ビットがセットされているとき、RxFIFO がフルになったときに自動的にクロックが停止されます。

  スレーブ設定では、MISO 出力が無効化され、ピンを GPIO として使用することができます。スレーブ選択信号がアクティブな間は、スレーブは MOSI ピンからデータを受信し続けます（*50.4.6：マルチマスタ通信*を参照）。データバッファの設定に応じて、受信データイベントが出現します。

注： ***マスタおよびスレーブモードのどちらであっても、SPI_CFG2 レジスタの IOSWP ビットの値を変更することによって、送信専用のデータピンを受信専用のデータピンで置き換えること、またその逆もできます。（このビットは SPI が無効なときにのみ変更できます）。***
すべての単方向通信は、トランザクション方向の設定を固定した（双方向モードは 有効でHDDIR ビットは決して変更されない）半二重通信の変形型に置き換えることができます。

図 604. 単方向シングルマスタ／シングルスレーブアプリケーション（送信専用モードのマスタ／受信専用モードのスレーブ）

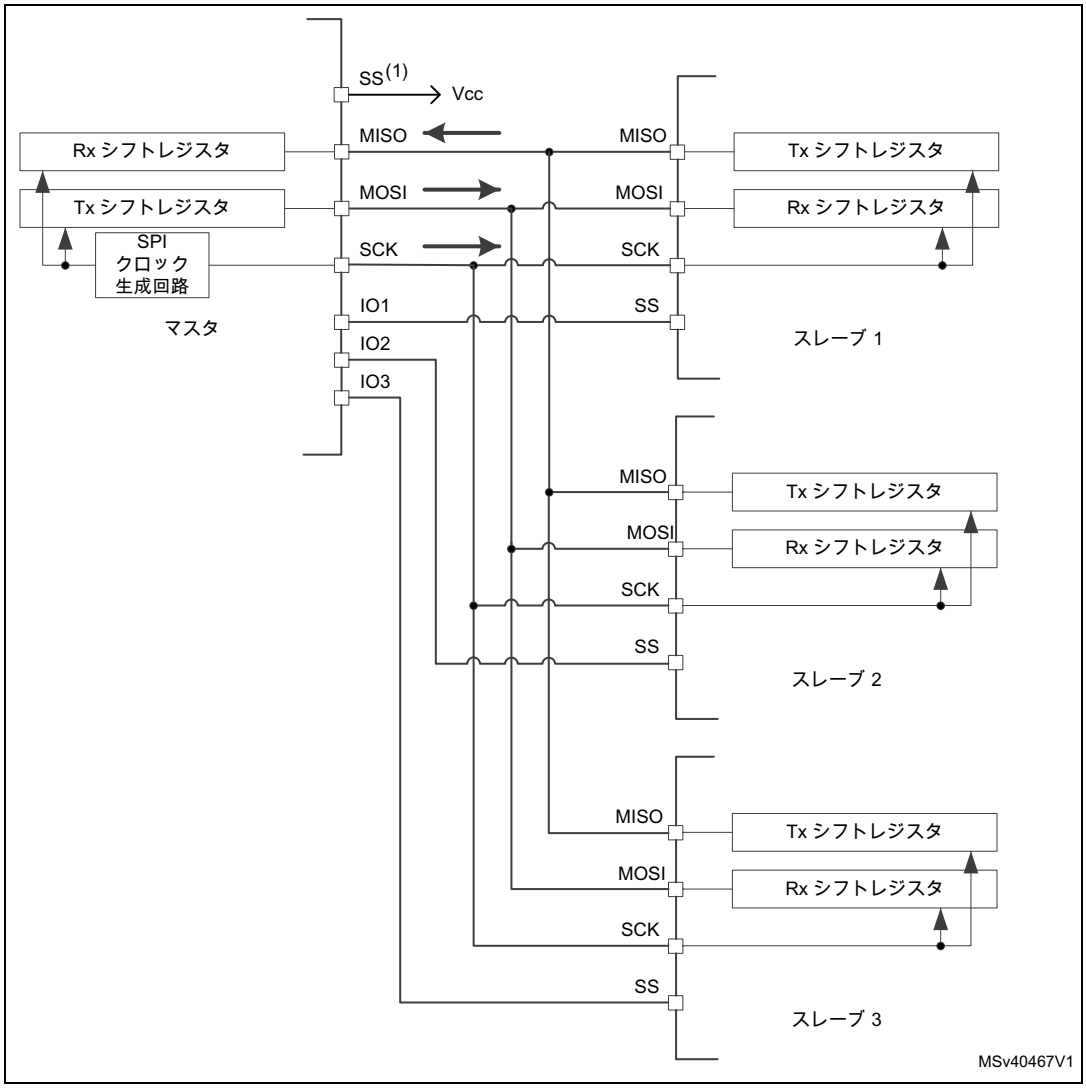


1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。
2. 入力情報はシフトレジスタでキャプチャされますが、標準の送信専用モード（たとえば、OVF フラグ）では無視する必要があります。
3. この設定では、両方のMISO ピンを GPIO として使用できます。

50.4.5 標準マルチスレーブ通信 kusa

2 つ以上の独立したスレーブがある設定の場合、マスタはスター接続形態を使用し、GPIO ピンを使用して、各スレーブのチップセレクトラインを別々に管理します（*図 605* を参照）。マスタは、スレーブの SS 入力に接続されている GPIO をローレベルにプルダウンすることによって、スレーブの 1 つを個別に選択する必要があります（一度に共通 MISO ライン上のデータを制御できるのは 1 つのスレーブのみです）。これを行うことにより、マスタと選択されたスレーブ間の通信が確立します。単純さ以外に、この接続形態の長所は、すべての通信セッションがシングルマスタ-スレーブペア内で別々に行われるので、特定の SPI 設定がスレーブごとに適用できることです。オプションとして、スレーブから情報を読み取る必要がないときは、マスタは同じ情報を複数のスレーブに送信することができます。

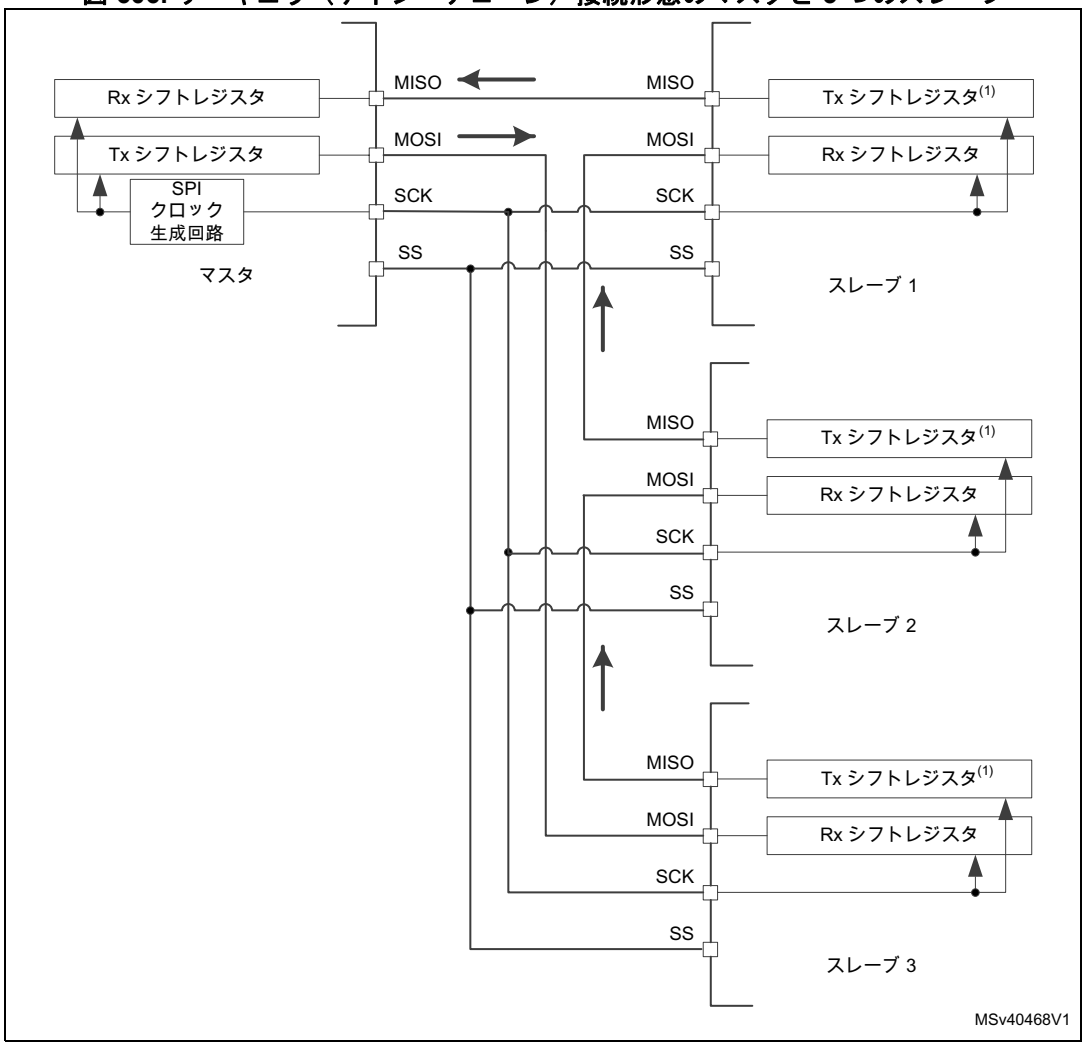
図 605. スター接続形態のマスタと 3 つの独立したスレーブ



1. この設定では、SS ピンはマスタ側では使用しません。内部的に管理（SSM=1、SSI=1）し、あらゆる MODF エラーを防ぐ必要があります。
2. スレーブの MISO ピンは相互接続されているので、すべてのスレーブにおいて、その MISO ピン の GPIO 設定をオルタネート機能オープンドレインとして設定する必要があります（*セクション 11.3.7：483 ページのI/O オルタネート機能の入力／出力* を参照）。

サーキュラ接続形態を適用した場合、マスタはすべてのスレーブと一度に SPI 通信を行うことができます（*図 606* を参照）。すべてのスレーブは、共通のスレーブ選択およびクロックの制御のもとで、シリアルチェーンで接続された単純なシフトレジスタのように動作します。すべての情報は同時にサークルを巡ってシフトされて、マスタに戻ります。セッションは長さが固定され、マスタによってトランザクションされるデータフレーム数はスレーブの数と同じです。そして、最初のデータフレームがチェーンにトランザクションされる時、マスタはチェーン内の最後のスレーブノードに向けられた情報を最初のスレーブノードの入力を経由して送信し、この時、マスタによって受信される最初の情報が最後のノードの出力から着信します。これと同様に、最初に出て行ったデータがセッション中に他のすべてのスレーブを通ってチェーンを巡った後にちょうどマスタの入力に到着する時に、セッションを終了させる最後のトランザクションデータが最初のスレーブノードに向けられます。この接続形態では、データフォーマットの設定とクロックの設定はチェーン内のすべてのノードで同じにする必要があります。受信および送信シフトレジスタは内部的には分離しているので、情報がレシーバとトランスミッタ間でハードウェアによってトランザクションされる時、TxFIFO スレーブで意図的なアンダーランのトリックを適用する必要があります。この場合、送信アンダーラン機能は、最後に受信したデータフレームを繰り返すモードに設定されます（UDRCFG[1:0]=01）。セッションはオプションによって、セッション開始前に各スレーブによって TxFIFO に書き込まれた単一のデータパターン（通常、スレーブステータス情報が適用されます）を使用して開始できます。この場合、アンダーランは実際に、この最初のデータフレームがトランザクションされた後に発生します（アンダーランは、スレーブでのデータのトランザクションの最後に検出されるように設定（UDRDET[1:0]=01）する必要があります）。内部のアンダーラン状態を直ちにクリアして、TxFIFO の内容で再度セッションをリスタートできるようにするには、ユーザはセッション間で SPI を無効化してまた有効化し、TxFIFO に新しい単一データパターンを入れる必要があります（これは、UDRC ビットによる標準の方法でアンダーランをクリアする場合に起きるクリア動作の伝播遅延を回避するためです）。

図 606. サーキュラ（デイジーチェーン）接続形態のマスタと 3 つのスレーブ



1. この設定では、スレーブの TxFIFO が空になった後、前に Rx シフトレジスタに受信したデータをスレーブが送信できるときに、アンダーラン機能がスレーブで使用されます。

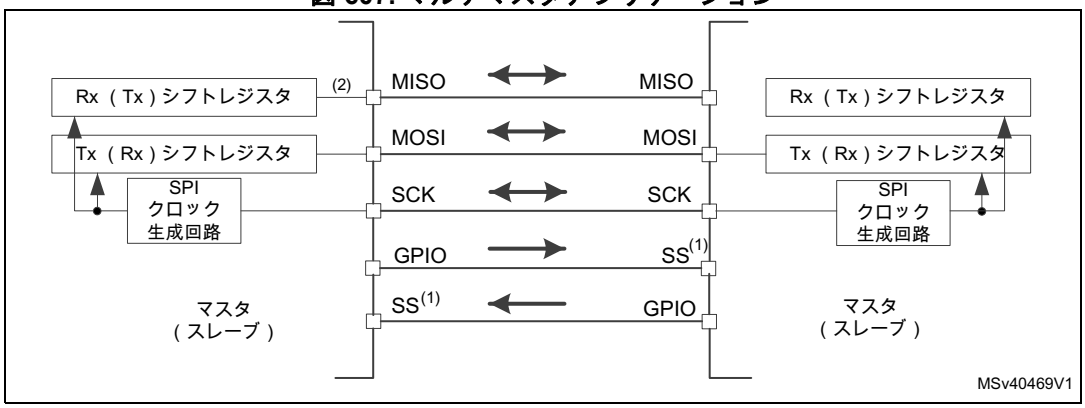
50.4.6 マルチマスタ通信

SPI バスが本来マルチマスタ機能向けに設計されていない限り、同時にバスを支配しようとする 2 つのノード間での競合の可能性を検出する内蔵機能を使用できます。この検出では、SS ピンをハードウェア入力モードに設定して使用します。共通のデータラインに一度に出力を提供できるノードは 1 つだけであるため、このモードで動作している 2 つを超える SPI ノードの接続は不可能です。

ノードが非アクティブであるとき、両ノードはデフォルトでスレーブモードのままとなります。1 つのノードがバスの支配権を得ようとすれば、自身をマスタモードに切り替え、専用の GPIO ピンを介して他方のノードのスレーブ選択入力をアクティブレベルにします。セッションが完了すると、アクティブスレーブ選択信号が解除され、バスを一時的に支配しているノードがパッシブスレーブモードに戻り、次のセッション開始を待機します。

同時に両方のノードのマスタリングリクエストが発生する可能性がある場合、バス競合イベントが発生します（モードフォールト MODF イベントを参照）。その場合、いくつかの簡単なアービトレーションプロセスを適用できます（たとえば、両方のノードに適用する事前定義された異なるタイムアウトによって次の試行を延期します）。

図 607. マルチマスタアプリケーション



1. SS ピンは両ノードでハードウェア入力モードで設定されます。アクティブレベルにすることで、パッシブノードがスレーブとして設定されるため、MISO ライン出力の制御が可能になります。

50.4.7 スレーブ選択（SS）ピンの管理

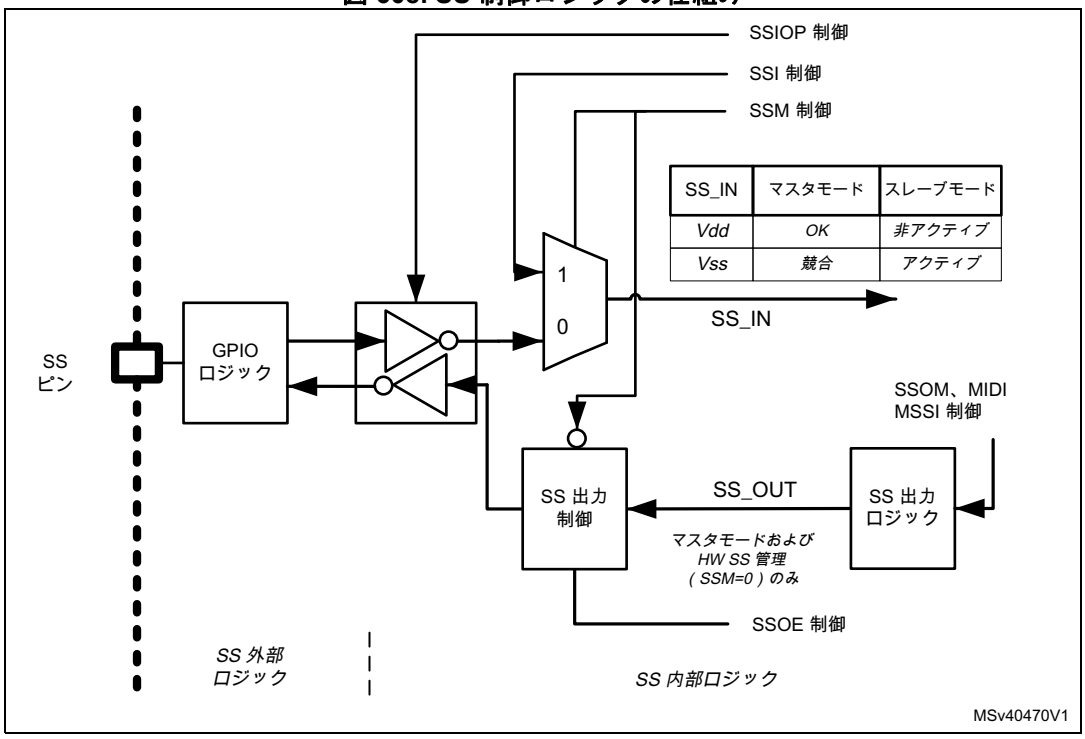
スレーブモードでは、SS は標準の“チップセレクト”入力として機能し、スレーブをマスタと通信させます。マスタモードでは、SS は出力としても入力としても使用できます。入力として使用する場合、SS はマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は 1 つのスレーブのスレーブ選択信号を駆動させることができます。SS 信号は内部的に管理することも（SS 入力のソフトウェアによる管理）、SS 入力および出力の両方が SS ピンに関連付けられているときは外部的に管理することもできます（ハードウェアによる SS 管理）。この入力／出力外部信号（SS ピンに存在）のどちらのレベルがアクティブとみなされるのかを SSIOP ビットのセットによって設定することができます。ローレベルが内部的にアクティブとみなされているとき、SSIOP=1 に設定すると、この論理を外部に対しては逆にすることができます。

ハードウェアまたはソフトウェアのスレーブ選択管理は、SPI_CFG2 レジスタの SSM ビットを使用して以下のように設定することができます。

- **ソフトウェア SS 管理（SSM = 1）**：この設定では、スレーブ選択情報は SPI_CR1 レジスタの SSI ビットの値によって内部で駆動されます。外部 SS ピンは他のアプリケーションで使用できます（GPIO またはその他のオルタネート機能として）。
- **ハードウェア SS 管理（SSM = 0）**：この場合、2 通りの設定が可能です。次のどちらの設定を使用するかは、SS 出力設定（SPI_CFG2 レジスタの SSOE ビット）によって決まります。
  - **SS 出力有効（SSOE = 1）**：この設定は、MCU がマスタとしてセットされている場合にのみ使用します。SS ピンはハードウェアによって管理されます。
    a) SSOM = 0 で、SP = 000 のとき、SS 信号は、マスタの転送が開始されると（CSTART=1）すぐにアクティブレベルになり、EOT フラグがセットされるか、送信がサスペンドされるまでアクティブに保たれます。
    b) SP = 001 のとき、TI モードで定義されたパルスが生成されます。
    c) SSOM=1、SP=000、および MIDI>1 のとき、SS はデータフレーム間で非アクティブにパルス出力され、MIDI 値から 1 を減じた値（1～14）によって定義される SPI クロック周期数の間、非アクティブに保たれます。
  - **SS 出力無効（SSM = 0、SSOE = 0）**：
    a) マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、SS ピンがアクティブレベルに引き込まれた場合、SPI はマスタモードのフォールト状態に入り、SPI は自動的にスレーブモードに再設定（MASTER=0）されます。
    b) スレーブモードでは、SS ピンは標準の“チップセレクト”入力として機能し、SS ラインがアクティブレベルの間はスレーブが選択されます。

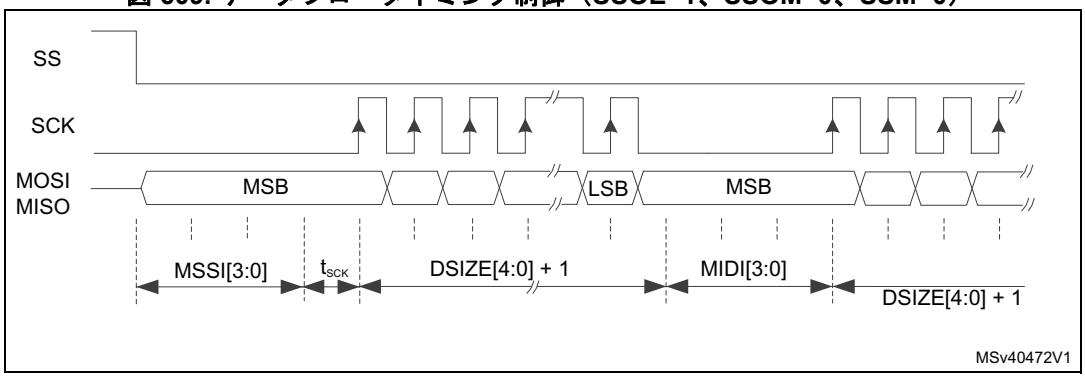
注： ***モードフォールト状態で自動的にスレーブモードに切り替える目的は、データおよびクロックラインでの競合の可能性を避けるためです。SPE は自動的にはリセットされません。リセットすると、RXFIFO および TxFIFO の両方が自動的に一掃され、現在のデータが失われることがあるからです。MODF イベントの後、ソフトウェアは FIFO の読み出し／一掃を正しく管理し、システムでのスレーブの役割を引き継ぐために SPI 設定を正しく再プログラムする必要があります。***

図 608. SS 制御ロジックの仕組み



ハードウェア出力 SS 制御が適用されるとき（SSM=0、SSOE=1）、MIDI[3:0] および MSSI[3:0] ビットフィールドの設定によって、データフレーム間の SS 信号のタイミングを制御し、（SS とクロック開始を分離するために）すべてのトランザクションの初めに余分の遅延時間を挿入することができます。これは、スレーブが正しいデータ処理のための十分な余裕を得るために流れの速度を落とす必要があるときに有用です（*図 609：データフロータイミング制御（SSOE=1、SSOM=0、SSM=0）* を参照）。

図 609. データフロータイミング制御（SSOE=1、SSOM=0、SSM=0）

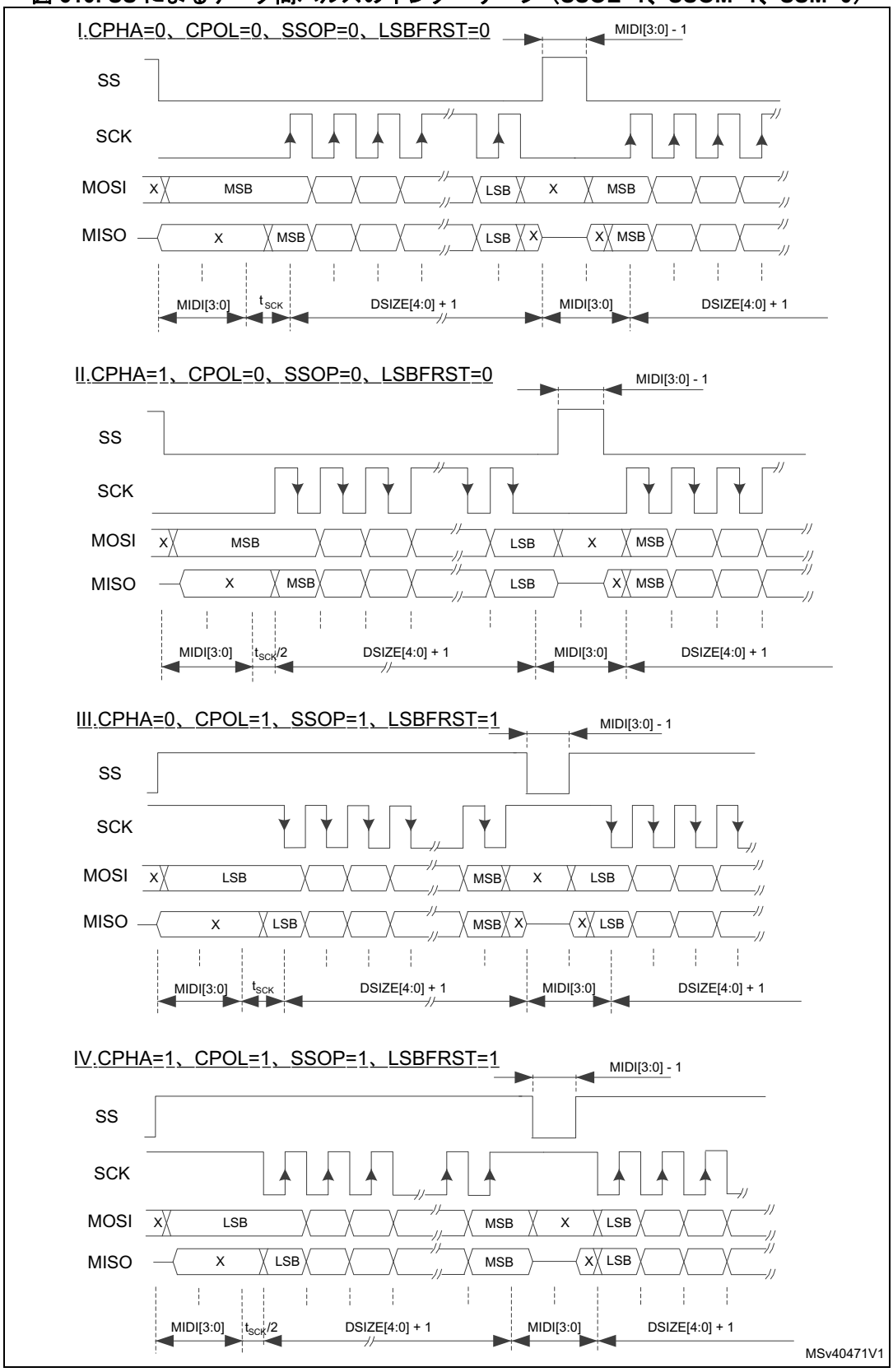


1. MSSI[3:0]=0011、MIDI[3:0]=0011（MIDI[3;0]=0 のときは、SCK フローは連続です）。
2. CPHA=0、CPOL=0、SSOP=0、LSBFRST=0。

さらに、ビット SSOM=1 の設定により、パルスを挿入する十分なスペースがあればデータフレーム間にパルスをインタリーブする特定のモードが呼び出されます（MIDI[3:0] は SPI の 1 周期より大きく設定する必要があります）。設定のいくつかの例を *図 610：SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0）* に示します。

参考資料

図 610. SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0）



1. MSSI[3:0]=0010、MIDI[3:0]=0010。
2. MIDI[3:0]>1 のとき、SS はデータ間にインターリーブします。

50.4.8 通信フォーマット

SPI 通信中は受信と送信の操作が同時に行われます。シリアルクロック（SCK）は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従い、正しく同期される必要があります。

クロックの位相および極性の制御

SPI_CFG2 レジスタの CPOL ビットと CPHA ビットを使用することによって、考えられる4 つのタイミングの関係をソフトウェアで選択できます。CPOL（クロック極性）ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOL がリセットされると、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされると、SCK ピンはハイレベルのアイドル状態になります。

CPHA ビットがセットされると、SCK ピンの 2 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがリセットされていれば立ち下がりエッジ、CPOL ビットがセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。CPHA ビットがリセットされている場合、SCK ピンの 1 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがセットされていれば立ち下がりエッジ、CPOL ビットがリセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。

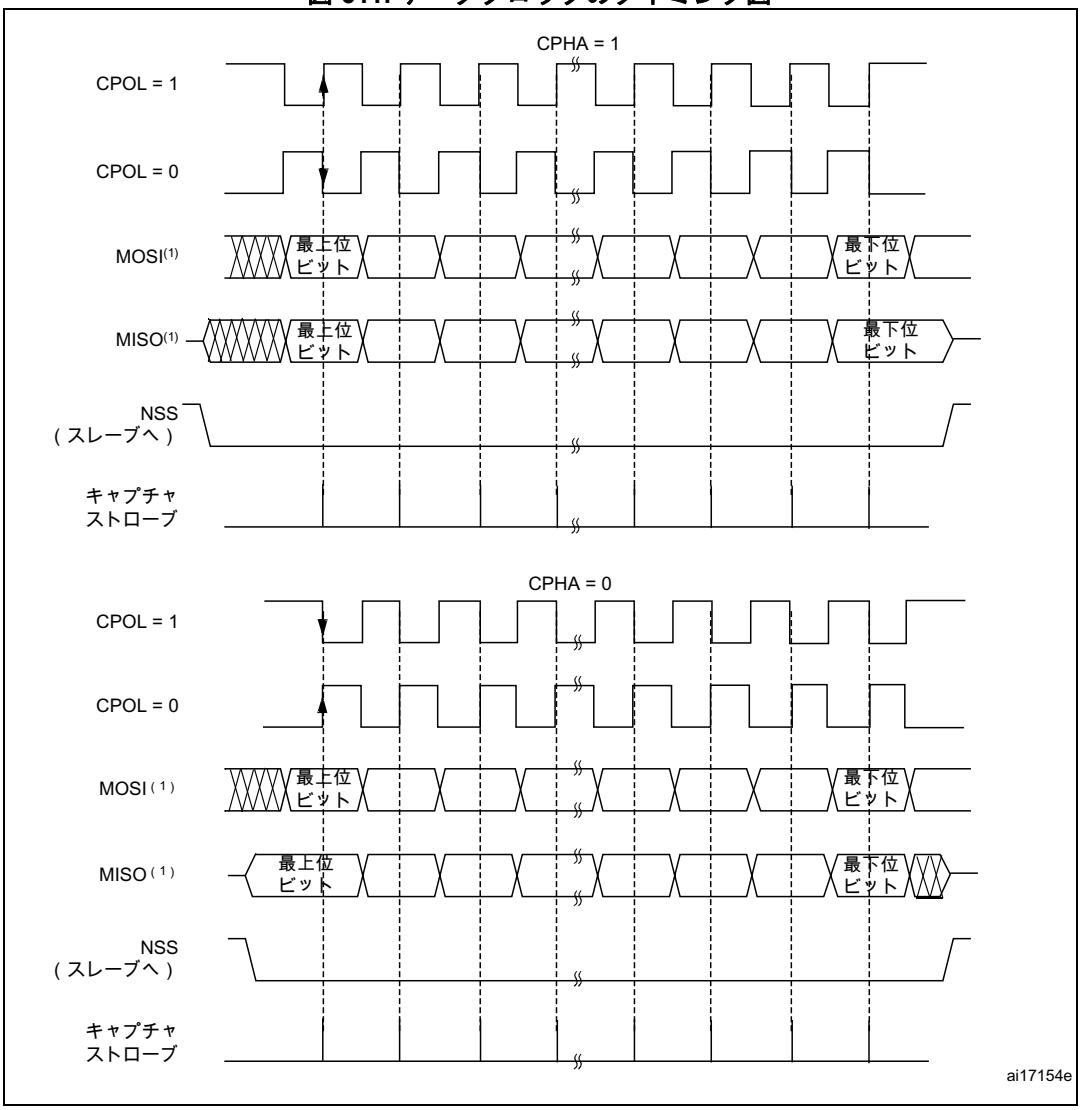
CPOL（クロック極性）ビットと CPHA（クロック位相）ビットの組み合わせによって、データキャプチャのクロックエッジを選択できます。

*図 611*は、CPHA ビットと CPOL ビットの 4 つの組み合わせによる SPI 全二重転送を示しています。

注： ***CPOL または CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。***

SCK のアイドル状態は、SPI_CFG2 レジスタで（CPOL = 1 なら SCK のプルアップ、CPOL = 0 なら SCK のプルダウンによって）選択された極性に一致する必要があります。

図 611. データクロックのタイミング図



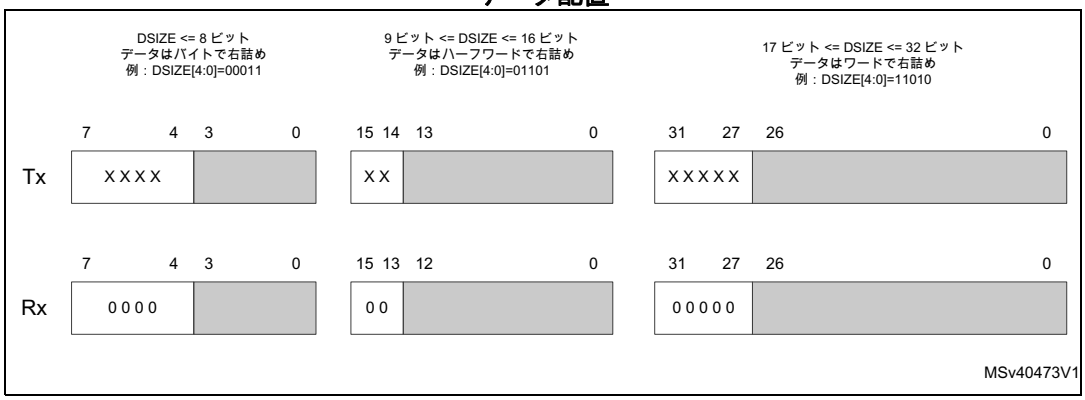
1. データビットの順序は LSBFRST ビットの設定値に依存します。

データフレームフォーマット

SPI_CFG2 レジスタの LSBFRST ビットの値に応じて、SPI シフトレジスタを設定することで、シフトを MSB ファーストまたは LSB ファーストに設定することができます。データフレームサイズは、DSIZE[4:0] ビットを使用して選択できます。4 ビットから 32 ビット長までの間にセットでき、この設定は送信と受信の両方に適用できます。SPI_TXDR/SPI_RXDR レジスタにアクセスすると、データフレームはバイト（データがバイトに適合する場合）、ハーフワード、またはワードのいずれかに常に右詰めされます（*図 612* を参照）。

アクセスが、1 つのデータフレームに必要な最小データサイズの倍数の場合、2 または 4 つのデータフレームがレジスタにパックされます。通信中、データフレーム内のビットのみがクロック供給され、送信されます。

図 612. データサイズが 8 ビット、16 ビット、または 32 ビットと等しくない場合のデータ配置



注： ***最小データ長は 4 ビットです。4 ビット未満のデータ長が選択された場合は、4 ビットのデータフレームサイズに固定されます。***

50.4.9 SPI の設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関する章を参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

7. 適切な GPIO レジスタに書き込みを行います。MOSIピン、MISOピン、SCKピンの GPIO 設定を行います。
8. SPI_CFG1 と SPI_CFG2 レジスタに書き込んで、以下を除いて、そこに含まれる予約済み以外のビットおよびビットフィールドすべてに適切な値を設定します。
   a) SSOM、SSOE、MBR[2:0]、MIDI[3:0]、および MSSI[3:0] はマスタモードでのみ必要です。MSSI ビットは SSOE がセットされたときに有効になります。MBR の設定は TI モードのスレーブに対して必要です。
   b) UDRDET[1:0] および UDRCFG[1:0] はスレーブモードでのみ必要です。
   c) CRCSIZE[4:0] は CRCEN がセットされた場合に必要です。
   d) CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、および SSM は TI モードでは必要ありません。
   e) SPI_CFG2 レジスタで AFCNTR ビットがセットされると、ペリフェラルが有効にもかかわらず、すべての SPI 出力は 関連する GPIO ピンに伝播され始めます。したがって、その後の SPI_CFG1 および SPI_CFG2 レジスタの設定変更は、これらのピンの信号のレベルに影響を与えます。
   f) SPI_I2SCGFR レジスタの I2SMOD ビットは、偶発的な I2S 設定の予期しない影響を防ぐために、クリアされたままにする必要があります。
9. 転送の長さを選択するために SPI_CR2 レジスタへ書き込みます。それが未知の場合は、TSIZE はゼロにプログラムしなければなりません。
10. 必要に応じて CRC 多項式と CRC 計算を設定するために、SPI_CRCPOLY と、SPI2S_CR1 レジスタの TCRCINI、RCRCINI、および CRC33_17 ビットに書き込みます。
11. DMA ストリームが使用されている場合は、DMA レジスタに SPI Tx および Rx 専用の DMA ストリームを設定します（DMA を使用した通信の章を参照）。
12. 設定の保護が必要な場合は（安全のため）、SPI_CFG1 レジスタの IOLOCK ビットをプログラムします。

50.4.10 SPI を有効にする手順

マスタがクロックを送信する前に SPI スレーブを設定して有効にすることを推奨しますが、トラフィックがバス上で進行中に設定および有効化手順を実行しても悪影響はありません。マスタがクロックを開始する前に、スレーブのトランスミッタのデータレジスタには送信するデータを格納するべきです。SPI スレーブが SS によって選択される前に、SCK 信号は選択した極性に対応するアイドル状態のレベルに確定していなければなりません。そうしないと、続くトランザクションが同期されないことがあります。

SPI スレーブがハードウェア SS 管理モードで有効化されたとき、SS がアクティブレベルであることが分かった場合でも、スレーブをマスタと同期させる SS 信号の開始（非アクティブからアクティブへのトランザクション）がスレーブに検出されるまで、すべてのトラフィックは無視されます。こういうわけで、外部 SS ピンが固定されているときは、ハードウェア管理モードは使用できません。SS のソフトウェア管理ではそのような保護はありません。この場合に限って、バスにトラフィックが無くて、SCK 信号が転送間でアイドル状態レベルにあるときに、SSI ビットを変更すべきです。

全二重（または送信専用モード）のマスタは、SPI が有効で CSTART ビットがセットされ、TXFIFO がエンプティでない場合、または次の TXFIFO への書き込み時に通信を開始します。

マスタ受信専用モードでは、SPI が有効化され、CSTART ビットがセットされた後、マスタは通信を開始し、クロックが動作を開始します。

DMA の扱いについては、*セクション 50.4.14：DMA（ダイレクトメモリアクセス）を使用する通信* を参照してください。

50.4.11 SPI のデータ送受信手順

RxFIFO および TxFIFO

すべての SPI データトランザクションは複数バイト（N x 8 ビット）で構成された内蔵 FIFO を通過します。FIFO のサイズ（N）は、製品およびペリフェラルのインスタンスに依存します。これにより、SPI は連続フローで動作できるようになり、またデータフレームサイズが短い場合や、割り込み／DMA 遅延時間が長すぎる場合にオーバーランを防ぐことができます。方向ごとにそれぞれ TxFIFO と RxFIFO と呼ばれる固有の FIFO があります。

FIFO の扱いはデータ交換モード（二重、単方向）、データフレームフォーマット（フレーム内のビット数）、FIFO データレジスタで実行されるアクセスサイズ（8 ビット、16 ビット、または 32 ビット）、およびパケットでのデータの構成方法に依存します。

SPI2S_RXDR レジスタへの読み出しアクセスからは、RxFIFO に保管された、まだ読み出されていない一番古い値が返されます。SPI2S_TXDR への書き込みアクセスでは、送信キューの最後に書き込まれたデータを TxFIFO に格納します。

SPI2S_RXDR レジスタへの読み出しアクセスは RXP イベントで管理する必要があります。このフラグは、受信がアクティブになっているとき、少なくとも 1 つの完全なデータパケット（SPI_CFG1 レジスタの FTHVL[3:0] ビットによってレシーバの閾値として定義されている）が受信 FIFO で利用可能になったときにハードウェアによってセットされます。RXP は、SPI2S_RXDR がソフトウェアまたは DMA によって読み出されたとき、RxFIFO に利用可能なデータが少なくなるとすぐにクリアされます。

RXP は、RXPIE ビットがセットされている場合は割り込みを、および／または RXDMAEN がセットされている場合は DMA リクエストをトリガします。

RXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタの読み出しを所要回数行って 1 つのデータパケットの内容をダウンロードします。完全なデータパケットがダウンロードされると、アプリケーションソフトウェアは RXP の値をチェックして受信 FIFO 内に他のパケットが保留されてないかを調べ、もしあれば、RXP が 0 になるまでそれらをパケットごとにダウ

ンロードします。N をバイトでの FIFO のサイズとして、RxFIFO が格納できる最大データフレーム数は、N 個（フレームサイズ =< 8 ビットの場合）、N/2 個（8 ビット < フレームサイズ =< 16 ビットの場合）、N/3 個（16 ビット < フレームサイズ=< 24 ビットの場合）、または N/4 個（フレームサイズ >24 ビットの場合）です。

受信の最後に、FTHVL レベルに達しておらず、いくらかのデータが RxFIFO 内でまだ利用可能になっていて、RXP がセットされないことがあります。この場合、FIFO 内に残っている受信データフレームの数が SPI_SR レジスタの RXWNE および RXPLVL フィールドによって示されます。これは、転送サイズとパケットサイズが整合していない場合に、転送で受信した最後のデータの数が設定パケットサイズを完全に満たすことができないときに発生します。それにもかかわらず、アプリケーションソフトウェアはそれまでの完全なデータパケットに使用された、RxFIFO からの標準の数の読み出しを問題なく行うことができます。矛盾のないデータ（完了したデータフレーム）のみが RxFIFO から取り出され、冗長な読み出し（または不完全なデータ）は 0 となります。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの受信を予知することや、RxFIFO から取り出すべき読み出し所要数を計算することの負担から解放されます。

同じように、送信するデータフレームの書き込みアクセスは TXP イベントで管理されます。このフラグは、送信がアクティブになっているとき、少なくとも 1 つの完全なデータパケット（SPI_CFG1 レジスタの FTHVL[3:0] ビットで定義されている）をアプリケーションソフトウェアが送信 FIFO にプッシュするのに十分なスペースができたときに、ハードウェアによってセットされます。TXP は、TxFIFO がソフトウェアおよび／または DMA によって充填され、次の完全なデータパケットのために現在利用できるスペースがなくなるとすぐにクリアされます。これによって、データが TxFIFO から放出され、新しいパケットがフレームごとに格納されるとき、TXP 信号が振動することになります。少なくとも 1 つのデータフレームを格納するだけの十分なスペースがないとき（TXP イベントは無視されます）、TXTF がセットされているとき、または SPI が無効になっているとき、TxFIFO への書き込みは無視されます。

TXP は、TXPIE ビットがセットされている場合は割り込みを、および／または TXDMAEN がセットされている場合は DMA リクエストをトリガします。TXPIE マスクは、TXTF フラグがセットされると、ハードウェアによってクリアされます。

TXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタへの書き込みを所定回数行って 1 つのデータパケット全体の内容をアップロードします。新しい完全なデータパケットがアップロードされると、アプリケーションソフトウェアは TXP の値をチェックして TxFIFO に他のパケットをプッシュできるかどうかを調べ、もしできるならば、パケットのロード後に TXP が 0 になるまで、それらをパケットごとにアップロードします。

転送サイズとパケットサイズが整合していない場合、転送の最後のデータの数が設定パケットサイズより短いことがあります。それにもかかわらず、アプリケーションソフトウェアはそれまでのパケットに使用された標準の数のデータレジスタ書き込みを問題なく行うことができます。矛盾のないデータのみが TxFIFO にプッシュされ、冗長な書き込みは破棄されます。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの送信を予知することや、最後のデータを TxFIFO にプッシュするための書き込み所要数を計算することの負担から解放されます。最後のデータの場合、現在の転送を完了するための残りのデータを格納するのに十分なスペースが TxFIFO にあれば、SPI によって TXP イベントがアサートされます。

TXP イベントと RXP イベントの両方を割り込みによってポーリングまたは処理できます。全二重モードでは TXP と RXP の共通のイベントとして DXP ビットを監視することができます。

DXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタへの書き込みを所要回数行って送信用の 1 つのデータパケット全体の内容をアップロードし、その後、SPI データレジスタから同じ数の読み出しを行い、1 つのデータパケットの内容をダウンロードします。1 つのデータパケットがアップロードされ、1 つがダウンロードされると、アプリケーションソフトウェアは DXP の値をチェックして他のパケットがプッシュおよびポップできないか順に調べ、もしできれば、DXP が 0 になるまでそれらをパケットごとにアップロード／ダウンロードします。

DXP は、DXPIE ビットがセットされている場合は割り込みを、および／または TXDMAEN と RXDMAEN がセットされている場合は DMA リクエストをトリガします。DXPIE マスクは、TXTF フラグがセットされると、ハードウェアによってクリアされます。

DXP は、全二重通信においてデータのアップロード／ダウンロードの性能を最適化するために有用であり、SPI の転送をサポートするのに必要な割り込み回数を削減し、その結果、特に SPI が STOP モードで動作しているとき、CPU の帯域幅およびシステム電力の要求を最小化します。

データ交換を管理するもう一つの方法は、DMA を使用することです（*DMA（ダイレクトメモリアクセス）を使用する通信*を参照）。

RxFIFO がフルのときに次のデータを受信した場合、オーバーランイベントが発生します（*セクション 50.5.2 : SPI エラーフラグ*の OVR フラグの説明を参照）。オーバーランイベントは割り込みによってポーリングまたは処理できます。

これはスレーブモードまたはマスタモード（全二重または MASRX = 0 で受信専用）で発生することがあります。マスタ受信専用モードでは、MASRX = 1 にすることで、RxFIFO がフルになったとき、生成されたクロックが自動的に停止し、したがってオーバーランは回避できます。

RxFIFO と TxFIFO の両方の内容は、SPI が無効（SPE=0）のときには一掃された状態に保たれます。

シーケンス処理

いくつかのデータフレームを単一シーケンスに渡してメッセージを完成させることができます。TSIZE および TSER フィールドに格納された値によって、メッセージ内のデータ数量を処理することができます。原則として、メッセージのトランザクションは CSTART ビットのセットによって SPI が有効になった時に開始し、要求された数量のデータがトランザクション処理された時に終了します。トランザクションの終わりに、CRC およびハードウェア SS 管理（適用される場合）が制御されます。CSTART がセットされる時、TSIZE がゼロに維持されている場合、終わりのないトランザクションが開始されます（データサイズの制御は適用されません）。トランザクションは、CSTART ビットをクリアする CSUSP によっていつでもサスペンドできます。

マスタモードでは、現在のセッション内のデータの数を拡張できます。TSIZE にプログラムされた数のデータがトランザクション処理された時、TSER にゼロでない値が入っていた場合、TSER の内容が TSIZE にコピーされて TSER 値は自動的にクリアされます。そうするとトランザクションは、TSIZE に再ロードされた値に相当するデータ数だけ拡張されます。この場合には、トランザクションは続行するので、EOT イベントは発生しません。再ロード操作の後、TSERF フラグがセットされ、TSERFIE がセットされていれば割り込みが発生します。次の再ロードが起きる前に TSER に次のゼロでない値を書き込むことができるので、このプロセスを繰り返している間、無制限の数のデータをトランザクション処理することができます。

データ拡張が適用されるときは、必ずデータパケットにそろえて開始されます。この理由で、拡張するデータ数は必ずパケットサイズにそろえることをお勧めします。そうしないと、拡張を適用する直前の最後のデータパケットを不完全なパケットとして取り扱わなければならなくなります（データパッキングの章を参照）。データの全体の数が整合していない場合は、最後の拡張サイクルの時に、そろってないデータの残りの数を TSER に実装し、そして最後の不完全なデータパケットを EOT イベントハンドラ内で標準的に処理する必要があります。

たとえば、8 ビットのデータサイズ、データパケットを4 つのデータに設定、FIFOは 32 ビットアクセスを使用する設定でデータ数拡張を適用して 23 バイトのデータを転送する場合、以下のシーケンスはすべて適正です。

- TSIZE=16、TSER=7
- TSIZE=12、TSER=8、最後の拡張で TSER=3

最後に、整合していない MSB バイトは、FIFO の最後（6 番目）のアクセスの中で無視されます。

次の場合のように、拡張するデータに対して整合していないシーケンスを適用すると、

- TSIZE=15、TSER=8 または
- TSIZE=8、TSER=7、最後の拡張で TSER=8

FIFO の 4 番目のアクセスの中で MSB バイトが無視され、他のアクセスは FIFO で常に 4 つのデータを取り扱います。

送信が有効な場合、シーケンスは開始し、マスタ側の TxFIFO に何らかのデータが存在する間、続行します。TxFIFO がエンプティになるまで、マスタによってクロック信号が永続的に供給され、その後停止して追加のデータを待ちます。

受信専用モード、半二重（COMM[1:0]=11、HDDIR=0）、または単方向（COMM[1:0]=10）では、SPI が有効化され、CSTART ビットによってトランザクションがリリースされると、マスタによってシーケンスが開始されます。マスタによってクロック信号が供給されますが、この信号はマスタが SPI または受信専用モードを無効／サスペンド状態にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。ソフトウェア制御によって SPI_CR1 レジスタの CSUSP ビットに 1 が書き込まれることによって、または MASRX=1 で、RxFIFO がフルになったときに自動的に、受信がサスペンドされます。また、SPI_CR2 レジスタの TSIZE および TSER フィールドにプログラムされたフレームの数が完了した時にも、受信は自動的に停止します。

マスタのレシーブ専用モードを無効にするには、最初に SPI をサスペンドする必要があります。SPI がサスペンドされると、現在のフレームが完了してから、設定を変更します。

注意： ***マスタにおいて、受信が進行中にサスペンドせずに SPE に 0 を書き込むと、現在のフレームが完了せずにクロックが停止し、RxFIFO が一掃されます。***

マスタはあらゆるトランザクションを連続モードで供給できる（SCK 信号は連続的）一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。必要に応じて、マスタは通信速度を下げ、より低速のクロックか、または MIDI[3:0] ビットの設定によって十分な遅延を含む分離したフレームやデータセッションを供給する必要があります。あるいは、スレーブがデータの準備のために十分な余裕を持てるように、トランザクションの開始を延期する MSSI[1:0] の設定によって初期遅延時間を提供する必要があります。スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されることに注意してください。スレーブが DMA を使用することが望ましく、特に、データフレームが短く、FIFO がバイトでアクセスされ、そして SPI バスが高速の場合に有効です。

スレーブのトランスミッタノードから SPI 通信のフローに何らかのソフトウェア制御を加えるためには、SPI_UDRDR（SPI アンダーランデータレジスタ）に書き込んだ特定の値を使用することができます。スレーブ側で、TxFIFO が空になると、この値が次のデータとして自動的に送出され、マスタレシーバ側のソフトウェアで解釈されます（単に読み捨てられるか、ソフトウェアによってマスタレシーバをサスペンドするために XOFF のようなコマンドとして解釈されます）。

マルチスレーブシステムにおいて通信用のスレーブを 1 つだけ選択するのと並行して、各シーケンスを SS パルスによって有効にする必要があります。単一のスレーブシステムでは SS によってスレーブを制御する必要はありませんが、スレーブを各データ転送シーケンスの開始と同期させるために、ここにもパルスを供給することをお勧めします。SS は、ソフトウェアとハードウェアの両方で管理できます（*セクション 50.4.6：マルチマスタ通信*を参照）。

50.4.12 SPI を無効にする手順

SPI を無効にする場合は、本段落に記載されている無効化手順に従ってください。

マスタモードでは、この手順はペリフェラルクロックが停止し、システムが低電力モードに入る前に行うことが重要です。そうしないと、この場合、進行中のトランザクションが破壊されることがあります。

スレーブモードでは、**spi_pclk** および **spi_ker_ck** クロックが停止したとき、通信の終了またはデータサービスリクエスト状態に至るまで、SPI 通信は中断することなく続行できます。**spi_pclk** は一般にシステムを STOP モードに設定することによって停止できます。詳細については、RCC のセクションを参照してください。

全二重または送信専用モードのマスタは、送信データの提供を停止したときにトランザクションを終了できます。この場合、クロックは最後のデータトランザクション後に停止します。最後のデータフレームが送信されるのを待つために、TXC フラグをポーリングする（または EOTIE=1 によって割り込みを有効にする）ことができます。

マスタが受信専用モードになっている場合、ペリフェラルを停止するためには、CSUSP を 1 にセットしてまず SPI 通信をサスペンドする必要があります。

SPI がサスペンドされたとき、受信されているが、まだ読み出されていないデータは RxFIFO に格納されたままになっています。

SPI が無効化されると、RxFIFO が一掃されます。未読のデータが失われないようにするためには、SPI を無効化するとき、残っているすべてのデータ（SPI_SR レジスタの RXP、RXWNE および RXPLVL フィールドに示されています）を読み出して RxFIFO が空であることを確認する必要があります。

標準的な無効化手順は、送信セッションが（完全に）終わったかどうかをチェックするために、EOT および／または TXC ステータスをポーリングすることに基づきます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- SS 信号がソフトウェアによって管理されており、マスタはスレーブに適切な SS パルスの終わりを提供する必要がある場合
- 最後のデータフレームまたは CRC フレームのトランザクションがまだペリフェラルバスで進行している間に、DMA または FIFO からのトランザクションのストリームが完了した場合

マスタモードでの正しい無効化手順を以下に示します（受信専用モードが使用されている場合を除く）。

1. TXC=1 および／または EOT=1（送信するデータがなくなり、最後のデータフレームが送信される）まで待ちます。CRC が使用されている場合、ブロックの最後のデータが処理された後に CRC が自動的に送信されます。この場合、CRC フレームが完了した時に TXC/EOT がセットされます。送信がサスペンドされるときは、ソフトウェアは CSTART ビットがクリアされるまで待つ必要があります。
2. すべての RxFIFO データを読み出します（RXWNE=0 および RXPLVL=00 になるまで）。
3. SPI を無効にします（SPE = 0）。

マスタ受信専用モードの場合の正しい無効化手順を以下に示します。

1. EOT まで待つか、または SPI をサスペンド（CSUSP=1）することによって受信フローを中断します。
2. 受信フローがサスペンドされたら、SUSP=1 （最後のデータフレームが処理される）まで待ちます。
3. すべての RxFIFO データを読み出します（RXWNE=0 および RXPLVL=00 になるまで）。
4. SPI を無効にします（SPE = 0）。

スレーブモードでは、SPI を無効にすると、処理中のデータが失われます。

50.4.13 データパッキング

ユーザの観点から、互いに重なり合う以下の 2 つのデータパッキングの方法があります。

- データを TxFIFO に書き込む、または RxFIFO から読み出すときのアクセスタイプ
  データサイズが、SPI2S_TXDR または SPI2S_RXDR レジスタで実行されるアクセス単位よりかなり小さい場合、複数のデータを 1 回のアクセスによって効率的にプッシュまたはフェッチすることができます。
- 1 回のソフトウェアサービス中に扱われるデータの数
  データをフレームごとに別々に取り扱う代わりに、データをパケットにグループ化し、もっぱらデータパケットの内容全体で FIFO サービスを累積するほうが便利です。ユーザは FIFO 閾値の設定によってパケットを定義できます。そうして、すべての FIFO 占有イベントはこの閾値レベルに関連し、要求されるサービスは割り込みやウェイクアップ機能による適切なフラグによって通知されます。

データフレームサイズが 1 バイト（8 ビット以下）におさまる場合、SPI2S_RXDR/SPI2S_TXDR レジスタで 16 ビットまたは 32 ビットの読み出しまたは書き込みアクセスが実行されたときに、自動的にデータパッキングが使用されます。この場合、多重データフレームパターンが並行して処理されます。最初に、SPI はアクセスしたワードの LSB に保管されたパターンを使用して動作します。次に、MSB に保管された残りデータを使用して動作します。*図 613* に、データパッキングモードのシーケンス処理の例を示します。そこでは DSIZE[3:0] が 4 ビットに設定され、トランスミッタの SPI2S_TXDR レジスタへの 16 ビットまたは 32 ビットの 1 回のアクセスの後、2 つまたは 4 つのデータフレームが TxFIFO に書き込まれます。

データフレームサイズが 9 ビットと 16 ビットの間である場合、32 ビットアクセスが実行されたとき、自動的にデータパッキングが使用されます。下位ハーフワードが最初に使用されます（LSBFRST の値にかかわらず）。

このシーケンスでは、RxFIFO 閾値が 1 フレームに設定されていた（データはパックされずにフレームベースで読み出される）場合、レシーバで 2 または 4 つの RXP イベントが生成される可能性があります。あるいは、SPI_CFG1 レジスタの FTHLV[3:0] フィールドが、複数フレームをパックモードで読み出す（16 ビットまたは 32 ビット読み出しアクセス）設定にプログラムされていた場合は、1 つの RXP イベントが生成されます。

データは *図 612：データサイズが 8 ビット、16 ビット、または 32 ビットと等しくない場合の データ配置* に従って配置されています。有効なビットだけがバスで実行されます。未使用のビットはトランスミッタでは考慮されず、レシーバではゼロで埋められます。

短いデータフレーム（8 ビット未満または 16 ビット未満）を、より大きいデータアクセスモード（16 ビットまたは 32 ビット）とともに使用する場合、FTHVL の値はフレームの数／データアクセスの倍数としてプログラムする必要があります（すなわち、最大 8 ビットのフレームに 32 ビットアクセスを使用する場合は 4 の倍数、あるいは、最大 8 ビットのフレームに 16 ビットアクセスを使用する場合、または最大 16 ビットのフレームに 32 ビットアクセスの場合は 2 の倍数）。

RxFIFO 閾値の設定は、次の読み出しアクセスサイズより必ず大きくなければなりません。そうしないと、誤った余分のデータが読み出されるからです。

設定したデータサイズより小さい FIFO データアクセスは禁止されます。必ず最低 1 つの完全なデータフレームをアクセスする必要があります。

4x8 ビットフレーム未満または 1 つの単一 16 ビットフレームが利用可能であるといった、不完全なデータパケットが FIFO で利用可能になっている場合は、特定の問題が現れます。

この問題を解決するには、2 つの方法があります。

A. TSIZE フィールドを使用しない方法

トランスミッタ側では、SPI2S_TXDR への 8 ビット／16 ビットアクセスによって任意の奇数シーケンスの最後のデータフレームを書き込むだけで十分です。

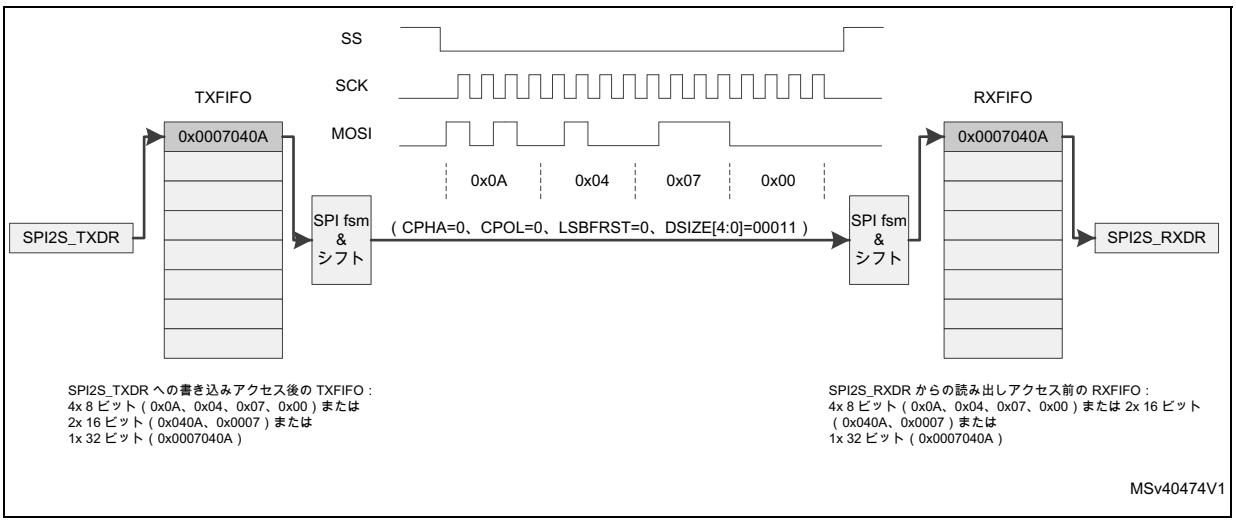
レシーバ側では、残りのデータは任意のアクセスによって読み出すことができます。余分なデータの読み出しはゼロで埋められます。受信データが RxFIFO に残っているかどうかを検出するためには、RXWNE および RXPLVL のポーリングが使用できます（ポーリングを検出するためにシステムレベルではタイムアウトが使用されることがあります）。

B. TSIZE フィールドを使用する方法

トランスミッタ側では、EOT イベントが発生するとトランザクションはマスタによって停止されます。

受信側では、EOT がセットされた時、RXP フラグはセットされません。受信するデータの数（TSIZE）がパケットサイズの倍数でなかった場合、残っているデータの数が SPI_SR レジスタの RXWNE および RXPLVL フィールドによって示されます。残りのデータは任意のアクセスによって読み出すことができます。余分な読み出しはゼロで埋められます。

図 613. 送受信用FIFO でのデータのパッキング



1. DSIZE[3:0] は 4 ビットに設定され、データは右詰め、有効なビットがバス上でのみ実行され、その順序は LSBFRST に依存し、LSB バイトの内容が最初にバスに出されます。

50.4.14 DMA（ダイレクトメモリアクセス）を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読み出し／書き込み処理を容易にするために、SPI は簡単なリクエスト／確認応答プロトコルを実現する DMA 機能を備えています。

SPI_CFG1 レジスタの TXDMAEN または RXDMAEN 有効ビットをセットすると、DMA アクセスがリクエストされます。TxバッファとRxバッファには、別々のリクエストを発行する必要があります。

- 送信では、TXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMA は SPI2S_TXDR レジスタに一連の書き込みを行います。
- 受信では、RXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMA は SPI2S_RXDR レジスタから一連の読み出しを行います。トランザクションの最後に EOT がセットされ、最後のデータパケットが不完全だった場合、RXWNE および RXPLVL[1:0] の設定に従って DMA リクエストが自動的にアクティブにされ、残りのデータが読み出されます。

SPI がデータの受信にのみ使用される場合、SPI Rx DMA チャネルのみを有効にすることができます。

SPI が受信専用モードにプログラムされている場合、UDR はセットされません。

SPI が送信モードにプログラムされている場合、送信データが利用できないことがあるので、スレーブ側で最終的に TXP および UDR がセットされることがあります。この場合、UDR 管理の選択によっていくらかのデータが TX ラインに送信されます。

SPI がデータの送信にのみ使用される場合、SPI Tx DMA チャネルのみを有効にすることができます。

SPI が送信専用モードにプログラムされている場合、RXP および OVR はセットされません。

SPI が全二重モードにプログラムされている場合、受信データが読み出されないので、最終的に RXP および OVR がセットされます。

送信モードで、DMA またはユーザがすべての送信データを書き込んだとき（SPI2C_SR レジスタの TXTF フラグがセットされます）、TXC フラグを監視することで SPI 通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPI を無効にする前、または マスタモードで **spi_pclk** を無効にする前にこの操作を行う必要があります。ソフトウェアは、まず EOT=1 および／または TXC=1 になるまで待つ必要があります。

DMA を使用して通信を開始する場合、DMA チャネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rx が使用される場合、SPI_CFG1 レジスタの RXDMAEN ビットの DMA Rx バッファを有効にします。
2. DMA が使用される場合、DMA レジスタにて Tx および Rx の DMA リクエストを有効にします。
3. DMA Tx が使用される場合、SPI_CFG1 レジスタの TXDMAEN ビットにてDMA Tx バッファを有効にします。
4. SPE ビットをセットして、SPI を有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. DMA が発行されていた場合、DMA レジスタにて Tx および Rx の DMA リクエストを無効にします。
2. SPI 無効化手順に従って SPI を無効にします。
3. DMA Tx および（または）DMA Rx が使用されている場合、SPI_CFG1 レジスタの TXDMAEN ビットおよびRXDMAEN ビットをクリアすることにより、DMA Tx バッファおよびRx バッファを無効にします。

DMA によるデータパッキング

DMA（SPI_CFG1 レジスタのTXDMAEN および RXDMAENをセット）で転送が管理される場合、パッキングモードは SPI TX および SPI RX DMA チャネルで設定される PSIZE 値によって、自動的に有効／無効になります。

DMA チャネルの PSIZE 値が 16 ビットまたは 32 ビットと等しく、SPI データサイズが 8 ビット以下の場合、パッキングモードは有効になります。同様に、DMA チャネルの PSIZE 値が 32 ビットと等しく、SPI データサイズが 16 ビット以下の場合、パッキングモードは有効になります。そうして DMA は SPI2S_TXDR レジスタへの書き込み動作を自動的に管理します。

フレームサイズが小さいとき、データパッキングモードが使用されていて、送信データ数が DMA データサイズ（16 ビットまたは 32 ビット）の倍数になっていない場合、DMA は TSIZE フィールドの設定に従って自動的に転送を完了します。

あるいは、シングル／アンパックモードで最後のデータフレームをソフトウェアで書き込むこともできます。

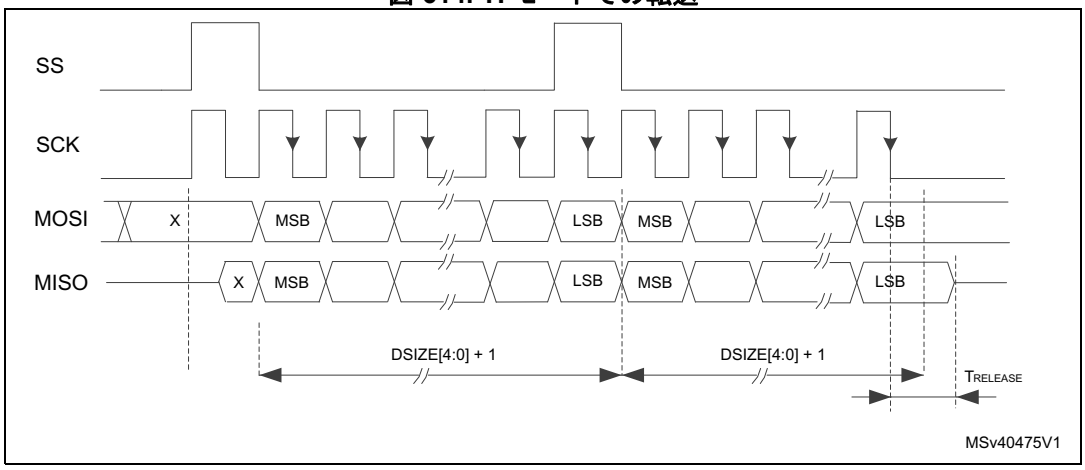
設定したデータサイズより小さい DMA データアクセスを設定することは禁止されます。必ず最小限 1 つの完全なデータフレームをアクセスする必要があります。

50.5 SPI の特定モード及び制御

50.5.1 TI モード

SPI_CFG2 レジスタの SP[2:0] ビットフィールドを特定の設定にすることによって、SPI を TI プロトコルに準拠させるように設定することができます。SPI が TI モードの設定になっているときは、SCK および SS 信号の極性、位相、フロー、およびビット順序は固定されているので、CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、および SSM の設定は必要ありません。SS 信号は、*図 614 : TI モードでの転送*に示すように、LSB データビット上のパルスによってプロトコルを同期させます。

図 614. TI モードでの転送



スレーブモードでは、現在のトランザクションが終了した時点で、MISO ピンでのスレーブ出力がハイインピーダンスになるタイミングを定義するためにクロックジェネレータが使用されます。マスタのボー設定が適用され、このタイミングを非常に柔軟に決定するために任意のボーレートが使用できます。MISO 信号がハイインピーダンスになる遅延時間（TRELEASE）も内部再同期に依存します。この値は次式で求められます：

$$\frac{Tbaud}{2} + 2 \times Tcom < Trelease < \frac{Tbaud}{2} + 4 \times Tcom$$

スレーブがデータトランザクション中に SS パルスの誤配置を検出すると、TIFRE フラグがセットされます。

50.5.2 SPI エラーフラグ

次のいずれかのエラーフラグがセットされて、対応する割り込み有効ビットをセットすることによって割り込みが有効になっている場合、SPI 割り込みが生成されます。

オーバーランフラグ（OVR）

マスタまたはスレーブでデータを受信し、RxFIFO に受信データを格納するための十分なスペースがない場合に、オーバーラン状態が発生します。これは、ソフトウェアまたは DMA が RxFIFO に格納された前の受信データを読み出すために必要な時間が十分になかった場合に発生する可能性があります。

オーバーラン状態が発生すると、OVR フラグがセットされ、RxFIFO にある前の値を新しく受信した値で上書きすることはありません。新しく受信した値は破棄され、それ以降に送信されるすべてのデータは失われます。OVRIE ビットがセットされている場合、OVR フラグは割り込みをトリガしま

す。OVR ビットのクリアは、SPI_IFCR の OVRC ビットに 1 を書き込むことによって行われます。次のオーバーランイベントを防ぐには、ソフトウェアの読み出しによって RxFIFO が空になってからビットのクリアをすべきです。マスタモードでは、自動通信サスペンド（MASRX ビット）によって RxFIFO オーバーランを防ぐことができます。

アンダーランフラグ（UDR）

スレーブ送信モードで､UDRDET ビットによって指定された時点にスレーブの TxFIFO に送信できるデータがなかった場合、アンダーラン状態がハードウェアによって内部的にキャプチャされます。その後、UDR フラグの設定はハードウェアによってステータスレジスタに伝えられます（下記の注を参照）。UDRIE ビットがセットされている場合、UDR は割り込みをトリガします。

アンダーランがキャプチャされると、次に提供される送信データは UDRCFG ビットによって決まります。スレーブは、最後に TxFIFO に格納されたデータ、または前にマスタから受信したデータ、またはユーザによって UDRDR レジスタに格納された一定のパターンのいずれかを提供できます。この 2 番目の設定はサーキュラ接続形態（*図 606* を参照）において使用できます。ソフトウェアによって UDR フラグがクリアされると、標準の送信が再有効化され、このクリア情報はハードウェアによって SPI ロジックに伝えられます。UDR フラグをクリアする前に、次のアンダーラン状態発生のキャプチャを防ぐために、いくらかのデータを TxFIFO に書き込む必要があります。

スレーブによってトランザクションされるデータは、特に TxFIFO が空で、アンダーラン状態がまだキャプチャされていないか、クリアされたばかりの時に、トランザクションが開始または続行されたときに、予測不可能です。一般的にこれは、UDRDET[1:0]=00 であるか、または SPI が有効になったばかりの時か、あるいは定義されたサイズでトランザクションが始まったばかりの時の場合です。さらに、この場合、データのトランザクションが開始される、あまりに直前にスレーブソフトウェアが空の TxFIFO に最初のデータを書き込んだとき、最初の数ビットは破壊される可能性があります（データを TxFIFO に伝播するのに APB クロックの数サイクルがかかります）。空の TxFIFO に、確実に間に合うようにデータを書き込むことができない場合、UDRDET[1:0]=00 の設定は避けるべきです。

アンダーラン制御機能を正しく取り扱うには、特に次の重大な侵害は避けるべきです。

- マスタがクロックを開始した時に、空の TxFIFO を満たすこと（特に UDRDET[1:0]=00 の場合）
- TxFIFO が空のときに、UDR フラグをクリアすること
- UDRCFG[1:0]=10 とともに UDRDET[1:0]=00 を設定すること
- SS 信号がフレーム間で反転しないうちに、各データフレームの後でアンダーランを検出しなければならないとき、UDRDET[1:0]=10 を設定すること
- SS がソフトウェアによって管理されているときに、UDRDET[1:0]=10 を設定すること

注： ***UDR フラグの変化をハードウェアで伝播するのにバス上で追加のトラフィックが必要になります。これには、イベント（ハードウェアによるアンダーランのキャプチャ、またはソフトウェアによる UDR フラグのクリア）が発生してから必ず SPI クロックの 3 周期がかかります。***

モードフォールト（MODF）

モードフォールトは、マスタデバイスが内部 SS 信号（SS ハードウェアモードでは SS ピン、SS ソフトウェアモードでは SSI ビット）をローレベルにプルダウンしたときに発生します。これは、自動的に SPI インタフェースに次のような影響を与えます。

- MODF ビットがセットされ、MODFIE ビットがセットされている場合は SPI 割り込みが生成されます。
- SPE　ビットがクリアされます。これによって、デバイスからのすべての出力がブロックされ、SPI インタフェースが無効になります。
- MASTER ビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODF は、SPI_IFCR レジスタの MODFC ビットに 1 を書き込むことによってクリアされます。

複数の MCU で構成されるシステムでスレーブ間の競合を避けるには、SPE ビットをセットすることによって SPI を再有効化する前に、SS ピンを非アクティブレベルにプルする必要があります。

安全のため、MODF ビットがセットされている間、ハードウェアは SPE ビットのセットを許可しません。スレーブデバイスでは、MODF ビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

マルチマスタシステムにおいてマスタがバスを支配するときの正しいソフトウェア手順は次の通りです。

- SSOE=0 のとき、マスタモードに切り替えます。
  （別のマスタがバスを占有していると、潜在的競合が起きる可能性があります。その場合、MODF が発生し、次のノードがマスタノードに切り替わることを防ぎます。）
- 別のマスタの SS 制御専用の GPIO ピンをアクティブレベルにします。
- データのトランザクションを実行します。
- 別のマスタの SS 制御専用の GPIO ピンを非アクティブレベルにします。
- スレーブモードに切り替えて戻します。

CRC エラー（CRCE）

このフラグを使用して、SPI_CFG1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。TSIZE によって定義された最後のデータが受信された後、シフトレジスタに受信された値が、レシーバの SPI_RXCRC の値と一致しなかった場合、SPI_SR レジスタの CRCE フラグがセットされます。CRCEIE ビットがセットされている場合、CRCE フラグは割り込みをトリガします。CRCE ビットのクリアは、SPI_IFCR の CRCEC ビットに 1 を書き込むことによって行われます。

TI モードフレームフォーマットエラー（TIFRE）

SPI がスレーブモードで動作し、かつ TI モードプロトコルに準拠した設定となっている場合、通信の実行中に SS パルスが発生すると、TI モードフレームフォーマットエラーが検出されます。このエラー発生すると、SPI_SR レジスタの TIFRE フラグがセットされます。エラー発生時には SPI は無効にされず、この SS パルスは無視されます。SPI は次の SS パルスを待ってから新規の転送を開始します。このエラーの検出により数バイトのデータが失われるため、データは破壊されている可能性があります。

TIFRE フラグは、SPI_IFCR の TIFREC ビットに 1 を書き込むことによってクリアされます。TIFREIE ビットがセットされている場合、SS エラー検出時に割り込みが生成されます。データの一貫性がもはや保証されなくなるため、ソフトウェアによってマスタとスレーブ間で通信を再起動するべきです。

50.5.3 CRC 計算

2 つの別々の 33 ビットの、または 2 つの別々の 17 ビットの CRC 計算機が、送信データおよび受信データの信頼性をチェックするために実装されています。SPI では任意の CRC 多項式の長さが提供されており、その長さはデータの最大サイズが 32 ビットの場合は 5 ビットから 33 ビットまで、データの最大サイズが 16 ビットに限られるペリフェラルインスタンスに対しては 5 ビットから 17 ビットまでです。多項式の長さは CRCPOLY レジスタに格納されている値の最上位ビットによって定義されます。その長さは DSIZE フィールドで定義されているデータフレーム長より大きくセットする必要があります。最大のデータサイズが適用されているときは、多項式ストリングのサイズを常にデータより大きく保ちながらその最上位ビットを定義するために、さらに CRC33_17 ビットをセットする必要があります。SPI_CFG1 の CRCSIZE フィールドでは、CRC 計算レジスタからいくつの最上位ビットが CRC フレームとしてトランザクションされ、比較されるのかを定義します。これはデータフレーム長とは独立に定義されますが、データフレームサイズと同じか、その倍数でなければなりません。

CRC の原理

CRC 計算は、SPI が有効（SPE = 1）になる前に、SPI_CFG1 レジスタの CRCEN ビットをセットすることによって有効になります。CRC 値は、CRCPOLY レジスタと CRC33_17 ビットによって定義された CRC 多項式を使用して計算されます。SPI が有効になっているとき、CRC 多項式は変更できますが、バス上にトラフィックがないときに限られます。

CRC 計算は、SPIx_CR1 レジスタの CPHA ビットと CPOL ビットによって定義されたサンプリングクロックエッジで、ビットごとに行われます。計算された CRC 値は、SPI_CR2 レジスタによって定義されたデータブロックの最後にのみ自動的にチェックされます。

受信データをもとに内部で計算された CRC 値とトランスミッタから受信した CRC 値の間に不一致が検出された場合、データ破壊エラーを示すために CRCERR フラグがセットされます。CRC を処理する正しい手順は、SPI の設定および選択された転送管理によって変わります。

CRC 転送管理

通信が開始され、最後のデータフレームが SPI_DR レジスタに送信または受信されるまで正常に続きます。

転送の長さは TSIZE および TSER によって定義される必要があります。要求されたデータ数がトランザクション処理されると、TXCRC が送信され、ラインで受信されたデータが RXCRC 値と比較されます。

CRC が有効な場合、TSIZE は 0xFFFF の値にセットすることはできません。たとえば、65535 個のデータを CRC とともに送信する正しい方法は、次のように設定することです。

- データパケットに 1 データが入るように設定された場合、TSIZE= 0xFFFE および TSER=1 とします。
- データパケットに 4 つのデータが入る場合、TSIZE= 0xFFFC および TSER=3 とします（その拡張が適用されたとき、TSIZE 値を確実にパケットサイズに整合させるため）。

送信では、CRC のトランザクション処理中、CRC 計算は停止し、TXCRC は CRCSIZE フィールドの値に等しい長さのフレームで送信します。

受信では、必要な数のデータがトランザクションされるとき、RXCRC は同じく停止します。その後、RXCRC レジスタの内容と比較される情報が、CRCSIZE 値と等しい長さのフレームで受信されます。

CRC フレームの受信が完了すると、受信した CRC 値と SPIx_RXCRC レジスタで計算された値を比較する自動チェックが行われます。ソフトウェアは、SPI_SR レジスタの CRCERR フラグをチェックして、データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCERRC に 1 を書き込んで CRCERR フラグをクリアします。

ユーザは冗長な CRC 情報を一掃する面倒をみる必要はなく、それは自動的に行われます。

SPIx_TXCRC および SPIx_RXCRC の値のリセット

CRC フェーズのあとで新しいデータがサンプリングされた時、SPI_TXCRC 値および SPI_RXCRC 値は自動的に初期化されます。これにより、データを中断することなく転送するために、DMA サーキュラモードが使用可能になります（いくつかのデータブロックは中間の CRC チェックフェーズによってカバーされます）。レシーバおよびトランスミッタ用の初期化パターンは、SPI2S_CR1 レジスタの TCRCINI および RCRCINI ビットの設定によってゼロまたはすべて 1 のどちらかに設定できます。

SPI が無効化されると、CRC の値はリセットされます。

50.6 低電力モード管理

SPI には、高度な低電力モード機能があり、**spi_pclk** クロックが無効になっているときでも FIFO とシリアルインタフェース間でデータを適切に転送することができます。

マスタモードでは、シリアルインタフェースのタイミングを提供するために、**spi_ker_ck** カーネルクロックが必要です。

スレーブモードでは、FIFO とシリアルインタフェース間でデータを転送中も **spi_ker_ck** クロックを取り除くことができます。このモードでは、クロックは外部の SPI デバイスから供給されます。

spi_pclk クロックがゲートされているとき（および SPI がスレーブの場合、**spi_ker_ck** クロックも）、**spi_pclk** クロックの有効化を必要とする次のような特定の動作が必要になった場合、SPI はウェイクアップイベント信号（**spi_wkup**）を提供します。

- TXFIFO を満たすこと
- RXFIFO を空にすること
- その他の通知：転送の終了、エラーなど

spi_ker_ck および **spi_pclk** クロックの生成は、レジスタの設定とプロセッサのモードに従って RCC ブロックによって制御されます。詳細については、RCC のセクションを参照してください。

アプリケーションは、SPI を低電力モードに切り替える（すなわち、**spi_pclk** を除去する）前に、保留中のすべての割り込みイベントに確認応答するべきです。

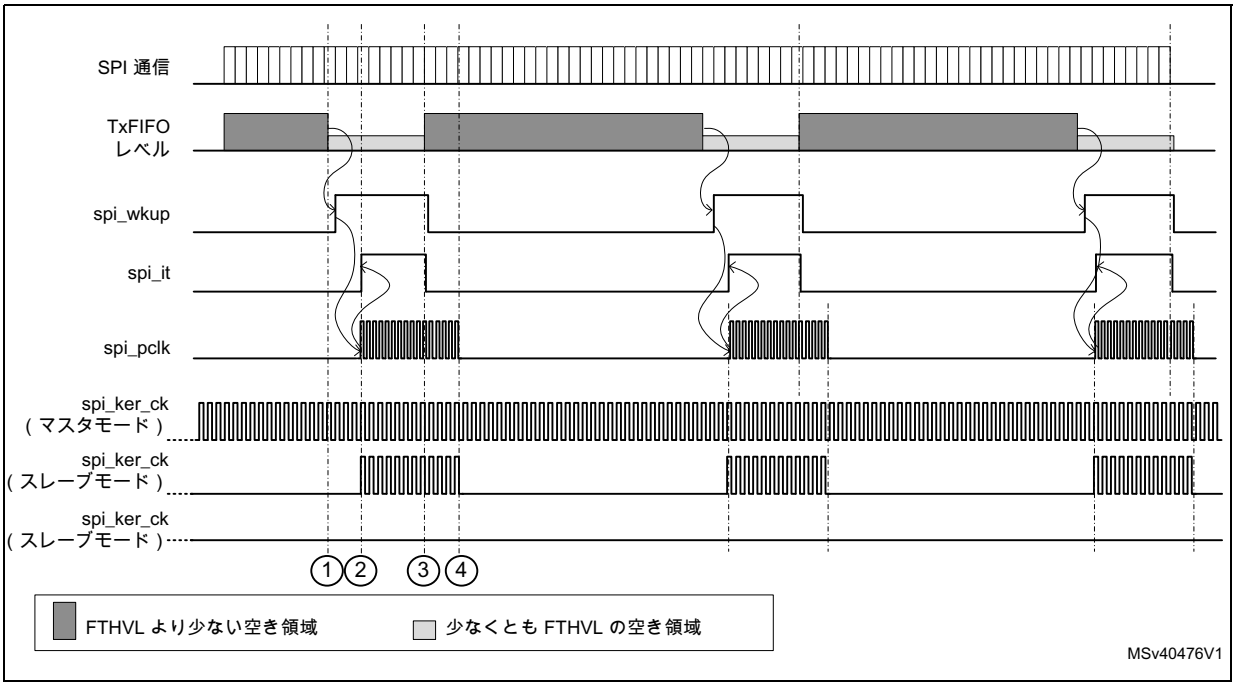
図 615 に、SPI2S が低電力モードで動作しているときのクロック処理の例を示します。この例は送信モードに対して示されています。

マスタモードでは、タイミング生成のために、**spi_ker_ck** クロックが必要です。

図 615 は、スレーブモードでの **spi_ker_ck** カーネルクロックの扱いについてサポートされている 下記 2 種類のシナリオを示しています。

- ほとんどのスレーブモードでは、**spi_ker_ck** カーネルクロックは無効にできます。
- いくつかの製品では、**spi_ker_ck** カーネルクロックのアクティベーションはシステムの状態に従うことがあります。

図 615. 低電力モードでのアプリケーションの例



図は、データをメモリから SPI2S の TxFIFO に転送する必要があるときに、**spi_pclk** を SPI2S に供給しなければならないことを明確に示しています。最も重要なステップの説明を以下に示します。

- **ステップ 1**
  TxFIFO のレベルがプログラムされた閾値より下がり、そのイベント（TXP）が **spi_wkup** 信号をアクティブにします。この信号は一般にシステムを低電力モードからウェイクアップするのに使用され、そしてバスクロック（**spi_pclk**）をアクティブにします。
- **ステップ 2**
  spi_pclk がアクティブになると、**spi_it** もアクティブになり、製品は DMA またはソフトウェアによって TxFIFO を充填する用意ができます。なお、いくつかの製品では、システムがウェイクアップすると自動的に **spi_ker_ck** カーネルクロックも有効化されることに注意してください。
- **ステップ 3**
  TxFIFO の空き容量が FTHVL より少なくなると、**spi_wkup** および **spi_it** 信号は非アクティブにされますが、TxFIFO の充填は続行することがあります。**spi_wkup** の立ち下がりエッジはシリアルインタフェースクロックドメインに整合し、**spi_it** の立ち下がりエッジは **spi_pclk** クロックドメインに整合することに注意してください。
- **ステップ 4**
  TxFIFO の充填が完了し、ソフトウェアは、次の **spi_wkup** が起きるまでシステムを低電力モードに切り替えて戻すことができます。

50.7 SPI ウェイクアップおよび割り込み

表 389 に、割り込みイベント（**spi_it**）を生成できる SPI イベントの概要を示します。そのうちのいくつかは、さらに低電力モードからのウェイクアップ機能（**spi_wkup**）を備えています。

それらの多くは、特定の割り込み有効制御ビットを使用することによって、独立に有効化および無効化することができます。

表 389. SPI ウェイクアップおよび割り込みリクエスト

割り込みイベント	イベントフラグ(1)	有効制御ビット	イベントのクリア方法	有効になる割り込み／ウェイクアップ	
				spi_it	spi_wkup
TxFIFO ロード準備済み（1 つのデータパケット - FIFO 閾値のスペースが利用可能）	TXP	TXPIE	TXP は、TxFIFO の空き容量が FTHVL より少なくなった時にハードウェアによってクリアされます。	はい	はい
データを RxFIFO に受信（1 つのデータパケットが利用可能 - FIFO 閾値）	RXP	RXPIE	RXP は、RxFIFO に入っているサンプルが FTHVL より少なくなった時にハードウェアによってクリアされます。		はい
TXP と RXP の両方がアクティブ	DXP	DXPIE	TXP または RXP がクリアされた時		はい
送信の転送フル	TXTF	TXTFIE	TXTFC に 1 を書き込む		いいえ
アンダーラン	UDR	UDRIE	UDRC に 1 を書き込む		はい
オーバーラン	OVR	OVRIE	OVRC に 1 を書き込む		はい
CRC エラー	CRCE	CRCEIE	CRCEC に 1 を書き込む		はい
TI フレームフォーマットエラー	TIFRE	TIFREIE	TIFREC に 1 を書き込む		いいえ
モードフォールト	MODF	MODFIE	MODFC に 1 を書き込む		いいえ
転送の終了（転送の全シーケンス完了 - TSIZE 値に基づいて）	EOT	EOTIE	EOTC に 1 を書き込む		はい
マスタモードサスペンド	SUSP		SUSPC に 1 を書き込む		はい
TXFIFO 送信完了（TxFIFO エンプティ）	TXC		TXC は、バス上で送信動作が開始された時にハードウェアによってクリアされます。		いいえ
TSER 値を TSIZE に転送（新しい値が TSER にロードされる）	TSERF	TSERFIE	TSERFC に 1 を書き込む		いいえ

1. イベントフラグについての詳細は、SPI2S レジスタの説明を参照してください。

50.8 I2S の主な機能

- 全二重通信
- 半二重通信（トランスミッタまたはレシーバのみ）
- マスタまたはスレーブ動作
- 8 ビットのプログラム可能なリニアプリスケーラ
- データ長は 16、24 または 32 ビット
- チャネル長は、マスタでは 16 または 32、スレーブでは任意の値
- プログラム可能なクロック極性
- 信頼性向上のためのエラーフラグ通知：アンダーラン、オーバーラン、およびフレームエラー
- 内蔵の RxFIFO および TxFIFO
- 以下の I^2S プロトコルをサポート。
  - フィリップス I^2S 規格
  - MSB 詰め規格（左詰め）
  - LSB 詰め規格（右詰め）
  - PCM 規格（ショートフレームおよびロングフレーム同期付き）
- データ順序はプログラム可能（LSb または MSb ファースト）
- 送受信用の DMA 機能
- 外部オーディオコンポーネントを駆動するためのマスタクロックを出力可能。周波数比は、256 × F_{WS} （F_{WS} はオーディオサンプリング周波数）に固定。

50.9 I2S の機能詳細

50.9.1 I2S の概要

図 601 に示したブロック図は I2S モードにも適用されます。

SPI/I2S ブロックは I2SMOD ビットが 1 にセットされたとき、I2S/PCM モードで動作します。I2S パラメータ専用の設定のために専用のレジスタ（SPI_I2SCFGR）が用意されており、そこにはクロックジェネレータやシリアルリンクインタフェースが含まれています。

SPI/I2S がマスタモードに設定されているとき、I2S/PCM 機能は通信クロックを発生させるためにクロックジェネレータを使用します。このクロックジェネレータは、マスタクロック出力（MCK）のソースでもあります。

RxFIFO、TxFIFO、DMA、および割り込み通知の部品といったリソースは SPI 機能と共用されています。低電力モード機能も I2S モードで利用可能です。*セクション 50.6：低電力モード管理* および *セクション 50.10：I2S ウェイクアップおよび割り込み* を参照してください。

50.9.2 SPI 機能とのピンの共用

I2S はSPI と、以下の 4 つの共通ピンを共用します。

- SDO：シリアルデータ出力（MOSI ピンにマップされます）は、マスタでオーディオサンプルを送信し、スレーブでオーディオサンプルを受信します。*セクション ： 2108 ページのシリアルデータラインのスワッピング* を参照してください。
- SDI：シリアルデータ入力（MISO ピンにマップされます）は、マスタでオーディオサンプルを受信し、スレーブでオーディオサンプルを送信します。*セクション ： 2108 ページのシリアルデータラインのスワッピング* を参照してください。
- WS：ワード選択（SS ピンにマップされます）はフレーム同期用です。これはマスタモードでは出力として、スレーブモードでは入力として設定されます。
- CK：シリアルクロック（SCK ピンにマップされます）はシリアルビットクロックです。これはマスタモードでは出力として、スレーブモードでは入力として設定されます。

外部オーディオデバイスにマスタクロック出力が必要な場合、追加のピンを使用できます。

- MCK（マスタクロック）：マスタクロック（個別にマップされます）は、I2S がマスタモードに設定されるときに使用されます。マスタクロック速度は、256 × F_{WS} （F_{WS} はオーディオサンプリング周波数）に固定されています。

参考資料

50.9.3 I2S/PCM モードで使用できるビットおよびフィールド

I2S/PCM モードが選択されているとき（I2SMOD = “1”）、いくつかのビットフィールドは無関係になり、I2S/PCM 機能の動作を保証するために特定の値に強制する必要があります。*表 390* は、I2S/PCM モードで使用できるビットとフィールドの一覧を示し、どれを特定の値に強制する必要があるかを示します。

表 390. PCM/I2S モードで使用できるビットフィールド

レジスタ名	PCM/I2S モードで使用できるビットフィールド	他のビットフィールドでの制約
SPI/I2S 制御レジスタ 1 （SPI2S_CR1）	IOLOCK、CSUSP、CSTART	他のフィールドはリセット値にセット
SPI 制御レジスタ 2（SPI_CR2）	-	リセット値にセット
SPI 設定レジスタ 1 （SPI_CFG1）	TXDMAEN、RXDMAEN、FTHVL	他のフィールドはリセット値にセット
SPI 設定レジスタ 2 （SPI_CFG2）	AFCNTR、LSBFRST、IOSWP	他のフィールドはリセット値にセット
SPI/I2S 割り込み有効レジスタ（SPI2S_IER）	TIFREIE、OVRIE、UDRIE、TXPIE、RXPIE	
SPI/I2S ステータスレジスタ（SPI2S_SR）	RXWNE、RXPLVL、SUSP、TIFRE、OVR、UDR、TXP、RXP	他のフラグは関係ない
SPI/I2S 割り込み／ステータスフラグクリアレジスタ（SPI2S_IFCR）	SUSPC、TIFREC、OVRC、UDRC	他のフィールドはリセット値にセット
SPI/I2S 送信データレジスタ（SPI2S_TXDR）	レジスタ全体	-
SPI/I2S 受信データレジスタ（SPI2S_RXDR）	レジスタ全体	-
SPI 多項式レジスタ（SPI_CRCPOLY）	-	リセット値にセット
SPI トランスミッタ CRC レジスタ（SPI_TXCRC）	-	
SPI レシーバ CRC レジスタ（SPI_RXCRC）	-	
SPI アンダーランデータレジスタ（SPI_UDRDR）	-	
SPI/I2S 設定レジスタ（SPI_I2SCGFR）	レジスタ全体	-

50.9.4 スレーブおよびマスタモード

SPI/I2S ブロックは、I2S および PCM プロトコルの両方に対してマスタとスレーブのモードをサポートしています。
マスタモードでは、CK、WSおよび MCKのすべての信号は出力にセットされます。

スレーブモードでは、CK および WS の両信号は入力にセットされます。MCK 信号はスレーブモードでは使用されません。

スレーブモードでの SPI/I2S ブロックの堅牢性を向上するために、ペリフェラルは各受信と送信を WS 信号に再同期させます。これは以下のことを意味します。

- フィリップス I2S 規格では、各データのシフトインまたはシフトアウトは、WS の各遷移の 1 ビットクロック後にトリガされます。
- I2S の MSB 詰め規格では、各データのシフトインまたはシフトアウトは、WS の遷移が検出されるとすぐにトリガされます。
- PCM 規格では、各データのシフトインまたはシフトアウトは、WS の立ち上がりの 1 ビットクロック後にトリガされます。

注： ***この再同期メカニズムは I2S の LSB 詰め規格では使用できません。***

注： ***SPI/I2S がスレーブモードに設定されているときは、カーネルクロックを供給する必要がないことにも注意してください。***

50.9.5 サポートされるオーディオプロトコル

I2S/PCM インタフェースは、SPIx_I2SCFGR レジスタの I2SSTD[1:0] と PCMSYNC ビットを使用して設定可能な 4 つのオーディオ標準をサポートします。

I2S プロトコルでは、オーディオデータは 2 つのチャネル（左チャネルと右チャネル）で時間多重化されています。WS 信号は、どちらのチャネルを左とし、どちらを右とするかを示すために使用されます。

I2S マスタモードでは、以下の 4 つのフレームフォーマットがサポートされています。

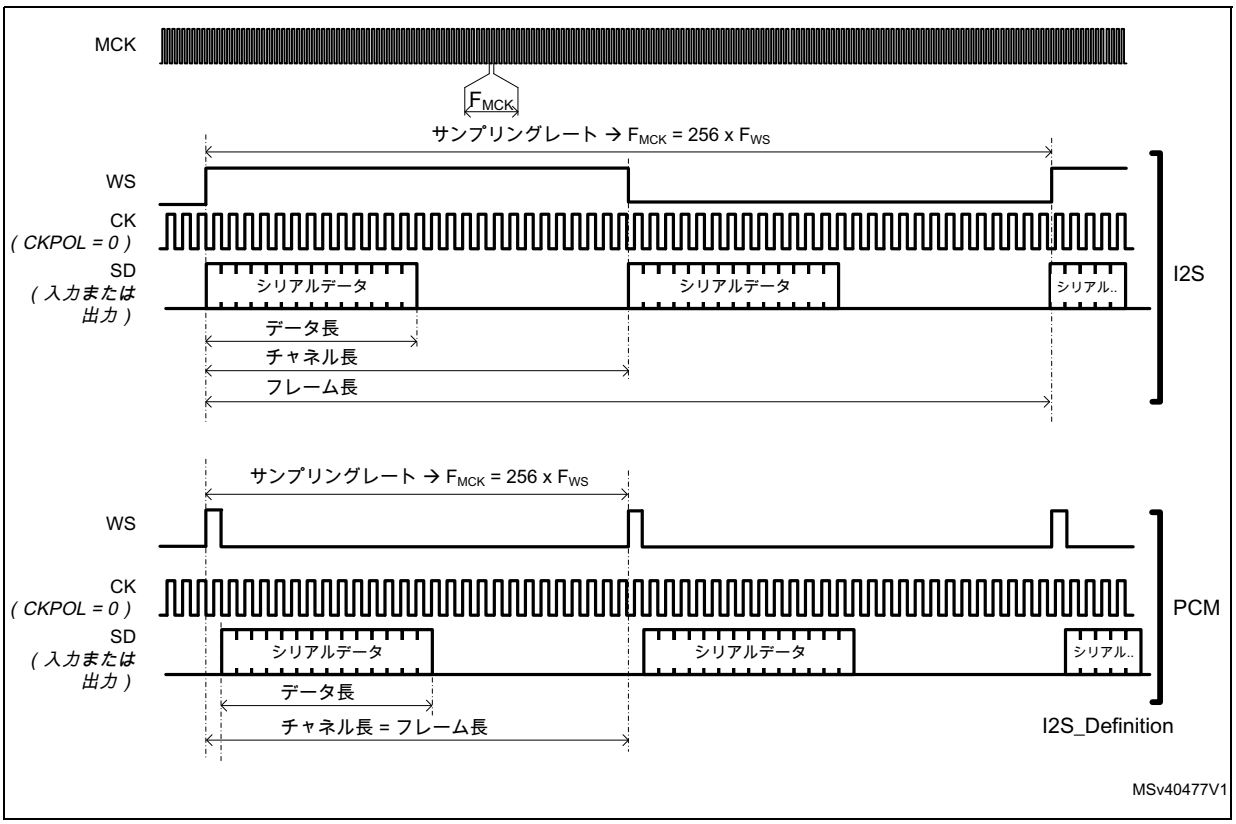
- 16 ビットチャネルにパックされた 16 ビットデータ
- 32 ビットチャネルにパックされた 16 ビットデータ
- 32 ビットチャネルにパックされた 24 ビットデータ
- 32 ビットチャネルにパックされた 32 ビットデータ

PCM マスタモードでは、以下の 3 つのフレームフォーマットがサポートされています。

- 16 ビットチャネルにパックされた 16 ビットデータ
- 32 ビットチャネルにパックされた 16 ビットデータ
- 32 ビットチャネルにパックされた 24 ビットデータ

以下の図に、本セクションで使用される、データ長、チャネル長、フレーム長の主な定義を示します。

図 616. 波形の例

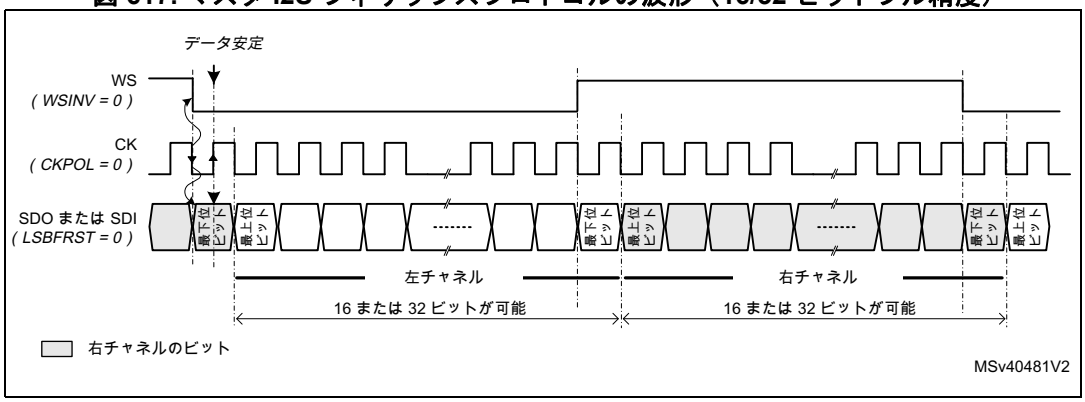


フィリップス I²S 規格

I2SSTD を 0b00 にセットすることによってフィリップス I2S 規格を選択できます。この規格はマスタおよびスレーブのモードでサポートされています。

この規格では、WS 信号が反転してから 1 CK クロックサイクル後に最初のビット（フィリップス I2S 規格では MSb）が使用可能になります。WS の立ち下がりエッジの遷移は、次の送信データが左チャネルであることを示し、立ち上がりエッジの遷移は、次の送信データが右チャネルであることを示します。

図 617. マスタ I2S フィリップスプロトコルの波形（16/32 ビットフル精度）

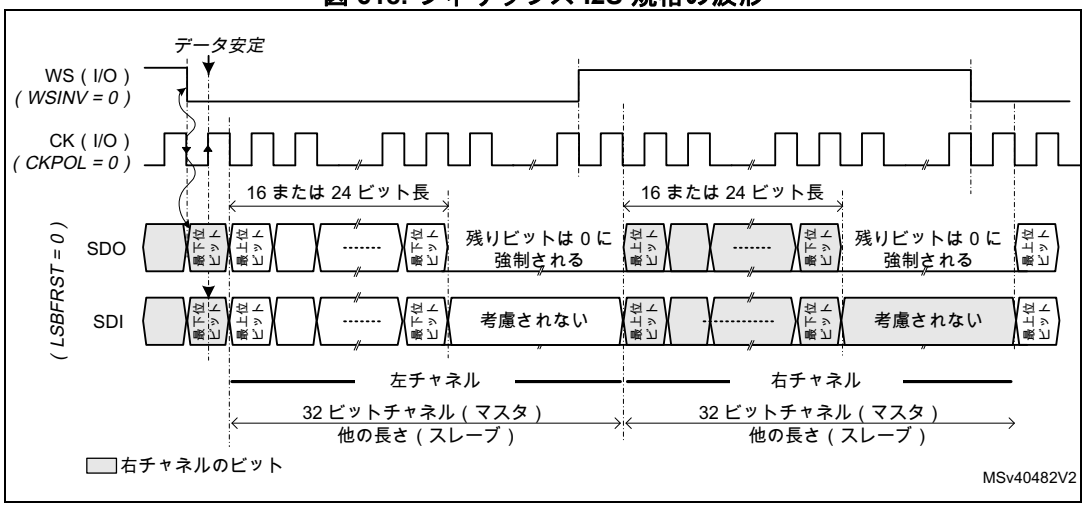


CKPOL は、I2S フィリップスプロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は *CK サンプリングエッジの選択* を参照してください。

図 617 は、チャネル長がデータ長に等しい場合に SPI/I2S によって生成される波形の例を示します。より正確には、CHLEN = 0 および DATLEN = 0b00 の場合、または CHLEN = 1 および DATLEN = 0b10 の場合です。

WS 信号の扱いに関する情報は *WS 反転の制御* を参照してください。

図 618. フィリップス I2S 規格の波形

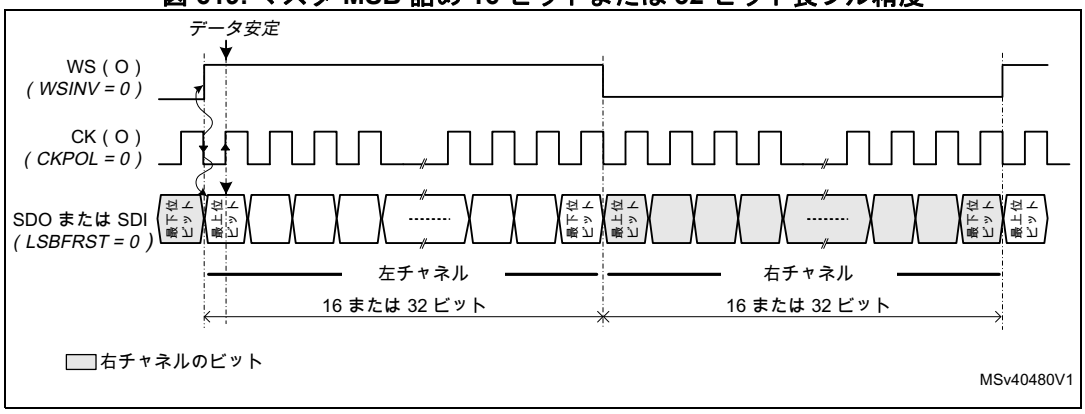


チャネル長がデータ長より大きい場合、SPI/I2S が送信モードに設定されているとき、残りのビットはゼロに強制されます。これはマスタモードとスレーブモードの両方に適用されます。

MSB 詰め規格

この規格では、WS 信号は最初のデータビットが提供される時に反転します。転送されるデータは、WS がハイの場合、左チャネルを表し、WS がローの場合、右チャネルを表します。

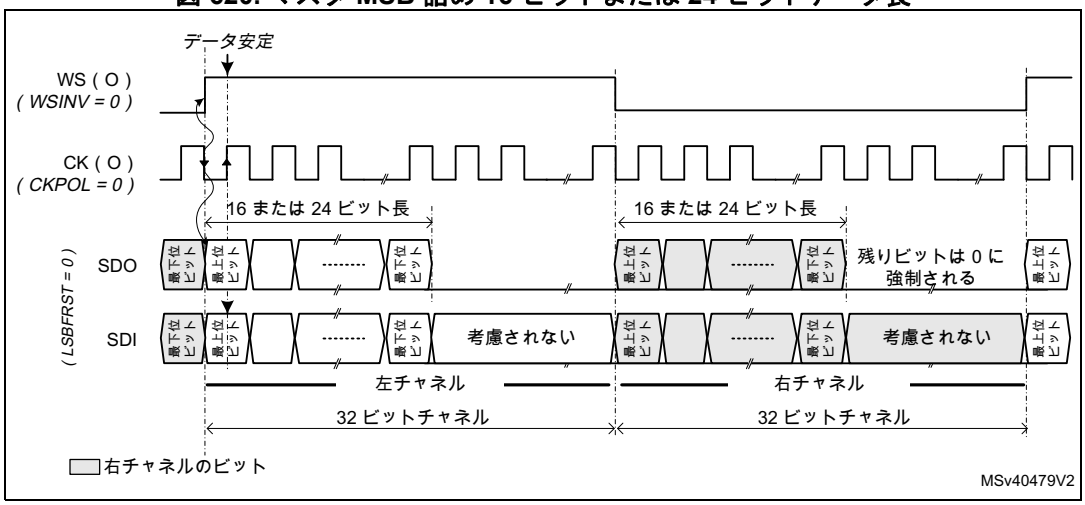
図 619. マスタ MSB 詰め 16 ビットまたは 32 ビット長フル精度



CKPOL は、I2S MSB 詰めプロトコルに合わせるために 0 に設定します。WS 信号の扱いに関する情報は *CK サンプリングエッジの選択* を参照してください。

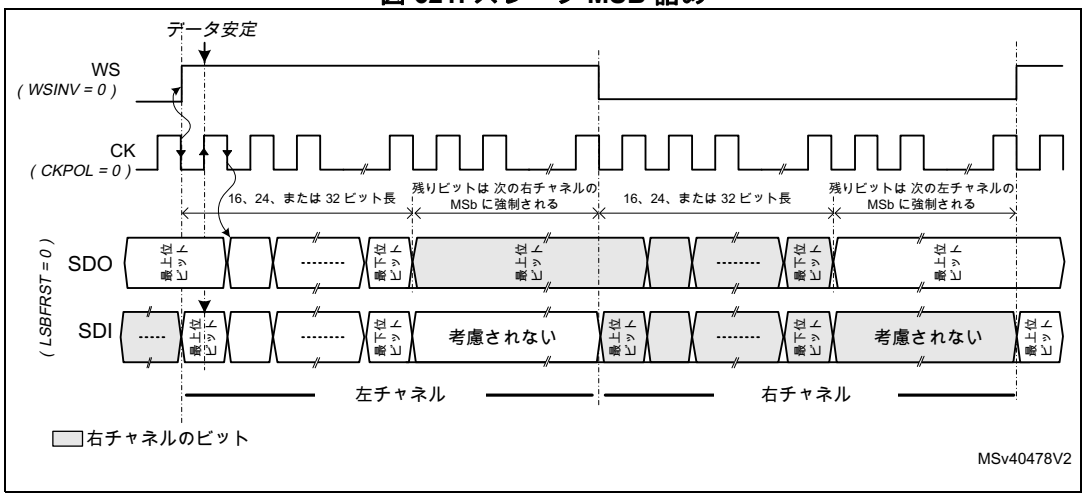
WS 信号の扱いに関する情報は *WS 反転の制御* を参照してください。

図 620. マスタ MSB 詰め 16 ビットまたは 24 ビットデータ長



チャネル長がデータ長より大きい場合、SPI/I2S がマスタ送信モードに設定されているとき、残りのビットはゼロに強制されます。スレーブ送信では、タイミングの問題を避けるために、残りのビットは生成される次のデータの最初のビット値に強制されます（*図 621* を参照）。

図 621. スレーブ MSB 詰め

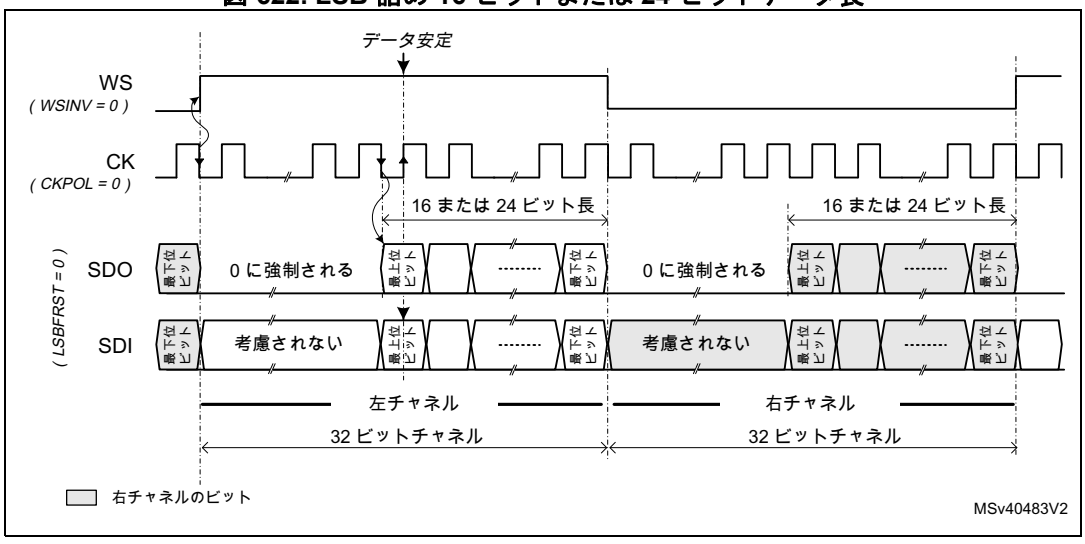


LSB 詰め規格

この規格は、マスタモードの MSB 詰め規格と似ています（16 ビットと 32 ビットのフル精度フレームフォーマットについては違いはありません）。LSB 詰め 16 または 32 ビットフル精度フォーマットでは、チャネルとデータが同じ長さなので、MSB 詰めモード（*図 619* を参照）と同様の波形になります。

注： ***LSB 詰めフォーマットでは、マスタおよびスレーブモードでは 16 および 32 ビットチャネル長しかサポートされていません。これは、チャネル長がトランスミッタおよびレシーバ側でわかっていないとデータを適切に転送することができないという事実のためです。***

図 622. LSB 詰め 16 ビットまたは 24 ビットデータ長



CKPOL は、I2S LSB 詰めプロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は *CK サンプリングエッジの選択* を参照してください。

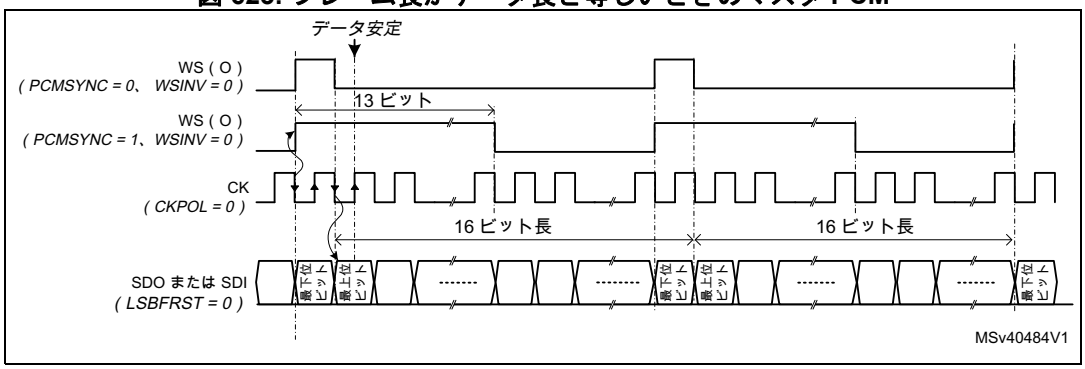
WS 信号の扱いに関する情報は *WS 反転の制御* を参照してください。

PCM 規格

PCM 規格では、チャネルサイド情報を使用する必要はありません。SPI_I2SCFGR レジスタの PCMSYNC ビットを使用して、2 つの PCM モード（ショートおよびロングフレーム）の使用および設定が可能です。

注： ***PCM のロングとショートフレームの違いはフレームの同期の幅だけです。両プロトコルとも、フレームのアクティブエッジは最初のビットの 1 CK クロックサイクル前に生成（スレーブモードでは予期）されます。***

図 623. フレーム長がデータ長と等しいときのマスタ PCM



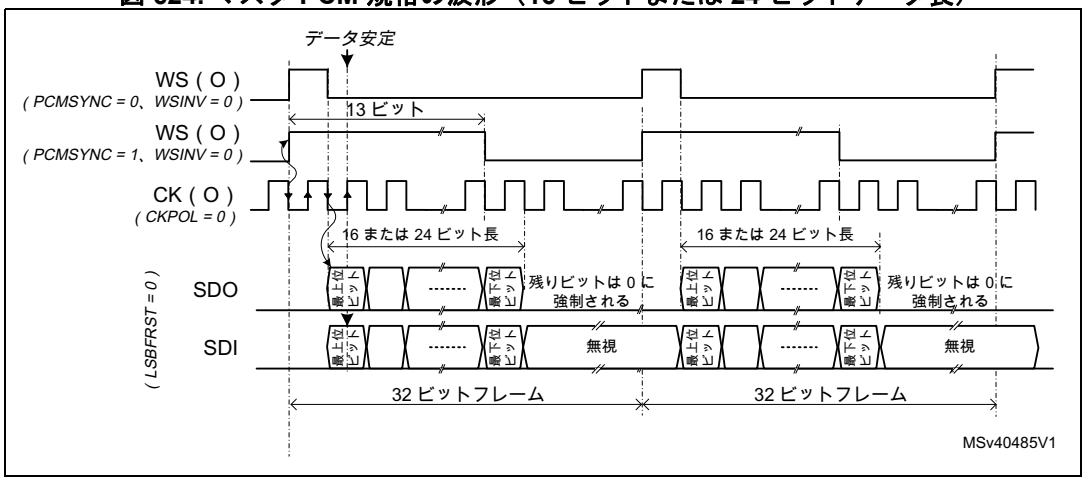
ロングフレーム同期では、WS 信号のアサーション時間はマスタモードで 13 ビットに固定されています。

チャネル長が 32 ビットにセットされているとき、16 または 24 ビットのデータサイズを使用することができます。

ショートフレーム同期では、WS 同期信号の長さは、わずか 1 サイクルです。

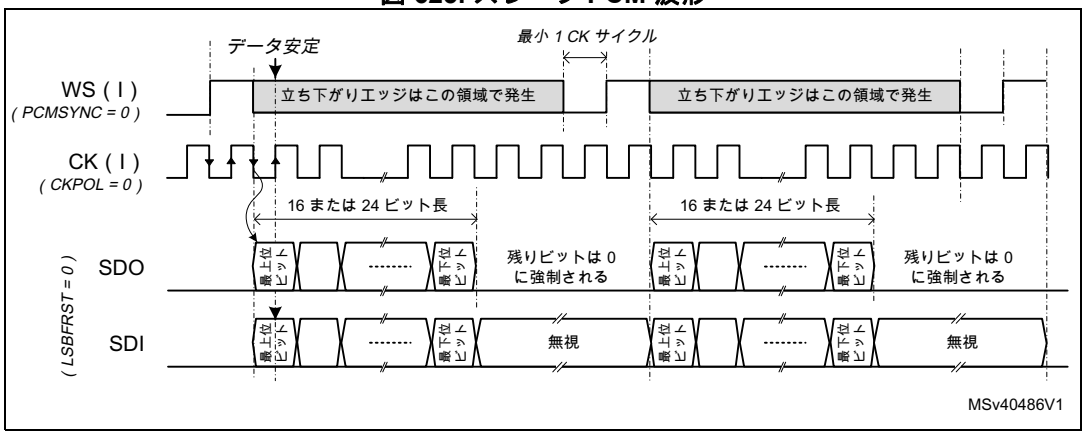
WS 信号の扱いに関する情報は *WS 反転の制御* を参照してください。

図 624. マスタ PCM 規格の波形（16 ビットまたは 24 ビットデータ長）



スレーブモードで PCM プロトコルが使用された場合、フレーム長は 16 または 32 ビットとは異なる可能性があります。*図 625* に示すように、スレーブモードでは、WS の立ち上がりエッジによってフレームの開始が検出されるので、さまざまな WS のパルス幅が許されます。唯一の制約は、WS は少なくとも 1 CK サイクル前に非アクティブ状態に戻らなければならないということです。

図 625. スレーブ PCM 波形



CKPOL は、PCM プロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は *CK サンプリングエッジの選択* を参照してください。

50.9.6 シリアルインタフェースの付加的な柔軟性

スレーブでの可変フレーム長

スレーブモードでは、チャネル長がデータ長より大きい限り、16 または 32 ビットと異なるチャネル長が許されます。これは、I2S LSB 詰めプロトコルを除いてすべてのプロトコルにあてはまります。

データ順序

すべてのデータフォーマットおよび通信規格に対して、*SPI 設定レジスタ 2 （SPI_CFG2）* にある LSBFRST ビットによってデータ順序（MSb または LSb ファースト）を選択することができます。

CK サンプリングエッジの選択

SPI/I2S 設定レジスタ（SPI_I2SCGFR） にある CKPOL ビットによって、すべてのプロトコルに対してスレーブおよびマスタモード用の CK のサンプリングエッジの極性を選択することができます。

- CKPOL = 0 のとき、シリアルデータ SDO および WS（マスタの時）は CK の立ち下がりエッジで変化し、シリアルデータ SDI および WS（スレーブの時）は 立ち上がりエッジで読み出されます。
- CKPOL = 1 のとき、シリアルデータ SDO および WS（マスタの時）は CK の立ち上がりエッジで変化し、シリアルデータ SDI および WS（スレーブの時）は 立ち下がりエッジで読み出されます。

WS 反転の制御

すべてのプロトコルに対して、WSINV を 1 にセットすることによって、マスタおよびスレーブモード用の WS 信号のデフォルトの極性を反転することができます。デフォルトでは、WS の極性は次のとおりです。

- フィリップス I2S 規格では、左チャネルに対して WS は LOW で、右チャネルに対して HIGH です。
- MSB/LSB 詰めモードでは、左チャネルに対して WS は HIGH で、右チャネルに対して LOW です。
- PCM モードでは、フレームの開始は WS の立ち上がりエッジで指示されます。

WSINV が 1 にセットされているとき、WS の極性は反転され、したがって次のようになります。

- フィリップス I2S 規格では、左チャネルに対して WS は HIGH で、右チャネルに対して LOW です。
- MSB/LSB 詰めモードでは、左チャネルに対して WS は LOW で、右チャネルに対して HIGH です。
- PCM モードでは、フレームの開始は WS の立ち下がりエッジで指示されます。

WSINV は *SPI/I2S 設定レジスタ（SPI_I2SCGFR）* に置かれています。

IO の制御

SPI 設定レジスタ 2 （SPI_CFG2） の AFCNTR ビットによって、SPI/I2S ブロックは SPI/I2S を有効にする前に WS および CK 信号を非アクティブ状態に確定することができます。

これは、次の手順を使用することによって CKPOL および WSINV をプログラムすることによって行うことができます。

AFCNTR は最初に 0 にセットされていると仮定します。

– I2SMOD = 1 にセットします（CK および WS の極性が CKPOL および WSINV 経由で制御されることをハードウェアに知らせるため）。
– CKPOL および WSINV ビットを希望の値にセットします。
– AFCNTR = 1 にセットします。
そうすると、SPI/I2S がまだ有効化されていなくても、CKPOL および WSINV の値に従って CK および WS IO の非アクティブレベルがセットされます。
– その後、I2S/PCM のアクティブ化手順を実行します。

表 391 に、AFCNTR ビットが 1 にセットされているとき、SPI/I2S ブロックが有効化される前（すなわち、非アクティブレベル）の、WS および CK 信号のレベルを示します。WS のレベルは選択したプロトコルにも依存することに注意してください。

表 391. AFCNTR = 1 のとき、SPI/I2S が有効化される前の WS および CK のレベル

WSINV	I2SSTD		SPI/I2S が有効化される前の WS のレベル
0	I2S 規格（00）	→	高
	その他	→	低
1	I2S 規格（00）	→	低
	その他	→	高

CKPOL		SPI/I2S が有効化される前の CK のレベル
0	→	低
1	→	高

注： ***SPI2S がスレーブモードのとき、AFCNTR ビットを 1 にセットするべきではありません。***

シリアルデータラインのスワッピング

SPI 設定レジスタ 2 （SPI_CFG2） にある IOSWP ビットによって、SPI/I2S は SDI ラインと SDO ラインの機能をスワップできます。*表 392* に、この機能の詳細を示します。

表 392. シリアルデータラインのスワッピング

設定	IOSWP	SDI の方向	SDO の方向
マスタ／スレーブ 受信	0	IN	-
	1	-	IN
マスタ／スレーブ 送信	0	-	OUT
	1	OUT	-

表 392. シリアルデータラインのスワッピング（続き）

設定	IOSWP	SDI の方向	SDO の方向
マスタ／スレーブ 全二重	0	IN	OUT
	1	OUT	IN

単純化のため、*I2S の機能詳細* のセクションに示された波形は IOSWP = 0 の場合で示されています。

50.9.7 起動シーケンス

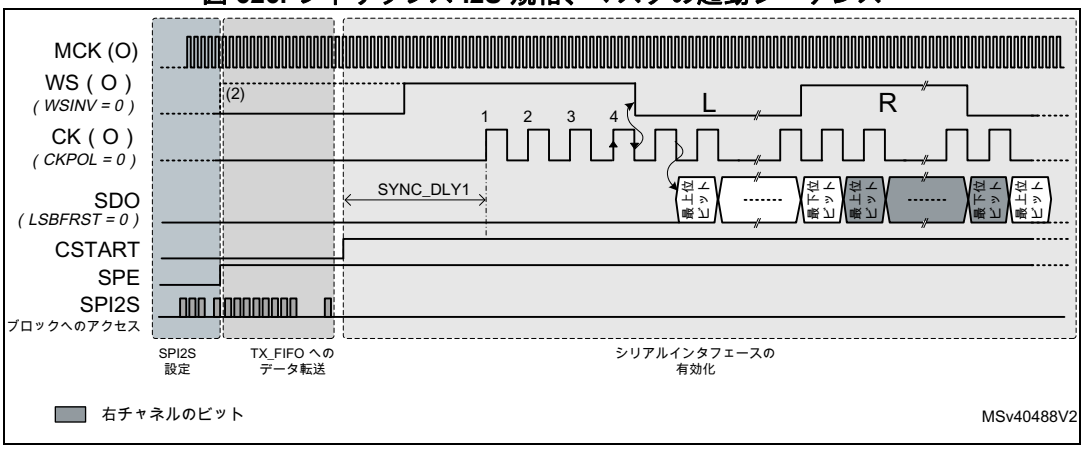
SPE ビットが 0 にセットされているとき、SPI2S_RXDR と SPI2S_TXDR レジスタに読み書きすることはできませんが、他のレジスタへのアクセスは許されます。

アプリケーションで SPI/I2S ブロックを使用したいときは、次の手順に従う必要があります。

1. SPE が 0 に設定されていることを確認し、そうでなければ SPE に 0 を書き込みます。
2. 希望の設定に従ってすべての設定と制御レジスタをプログラムします。詳細なプログラミング例については、*セクション 50.9.16* を参照してください。
3. SPI/I2S ブロックをアクティブ化するために SPE ビットを 1 にセットします。このビットがセットされると、シリアルインタフェースはまだ無効ですが、DMA と割り込みサービスは動作していて、たとえば、TxFIFO へのデータの転送は可能です。
4. シリアルインタフェースをアクティブ化するために CSTART ビットを 1 にセットします。

図 626 に示すように、フィリップス I2S 規格のマスタ送信では、CSTART ビットが 1 にセットされて TxFIFO が空でなくなるとすぐに、WS、MCK および CK 信号の生成が開始されます。外部のスレーブデバイスが 確実にWS の遷移を適切に検出するために、WS の立ち下がりエッジが発生する4回の立ち上がりエッジ前に、ビットクロック CK がアクティブになることに注意してください。他の規格も同様に動作します。

図 626. フィリップス I2S 規格、マスタの起動シーケンス



1. 本図では、SPE ビットを 1 にセットする前に、MCK が有効化されています。詳細については、*MCK の生成* を参照してください。
2. WS および CK 信号のレベルは、AFCNTR ビットが 1 にセットされるとすぐに、設定フェーズ中に SPI/I2S ブロックによって制御されることに注意してください。

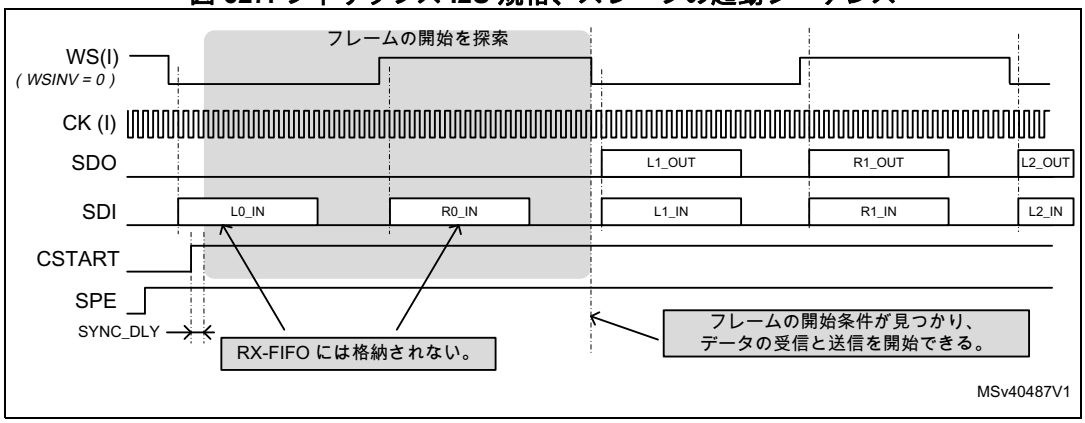
注： ***クロックドメインの再同期のために、CSTART ビットは CK クロックの約 3 周期（SYNC_DLY1）後にハードウェアによって考慮されます。***

スレーブモードでは、CSTART が 1 にセットされると、以下のフレームの開始条件が満たされるときにデータ転送が開始されます。

- フィリップス I2S 規格では、フレームの開始条件は WS 信号の立ち下がりエッジです。送信／受信は 1 ビットクロック後に開始されます。
  WSINV = 1 の場合は、フレームの開始条件は 立ち上がりエッジです。
- 他のプロトコルでは、フレームの開始条件は WS 信号の立ち上がりエッジです。送信／受信は、MSB 詰めプロトコルでは WS の立ち上がりエッジで開始されます。PCM プロトコルでは、送信／受信は 1 ビットクロック後に開始されます。
  WSINV = 1 の場合は、フレームの開始条件は 立ち下がりエッジです。

図 627 に、フィリップス I2S 規格、スレーブモードの起動シーケンスの例を示します。

図 627. フィリップス I2S 規格、スレーブの起動シーケンス



注： ***クロックドメインの再同期のために、CSTART ビットは CK クロックの 2 周期（SYNC_DLY）後にハードウェアによって考慮されます。***

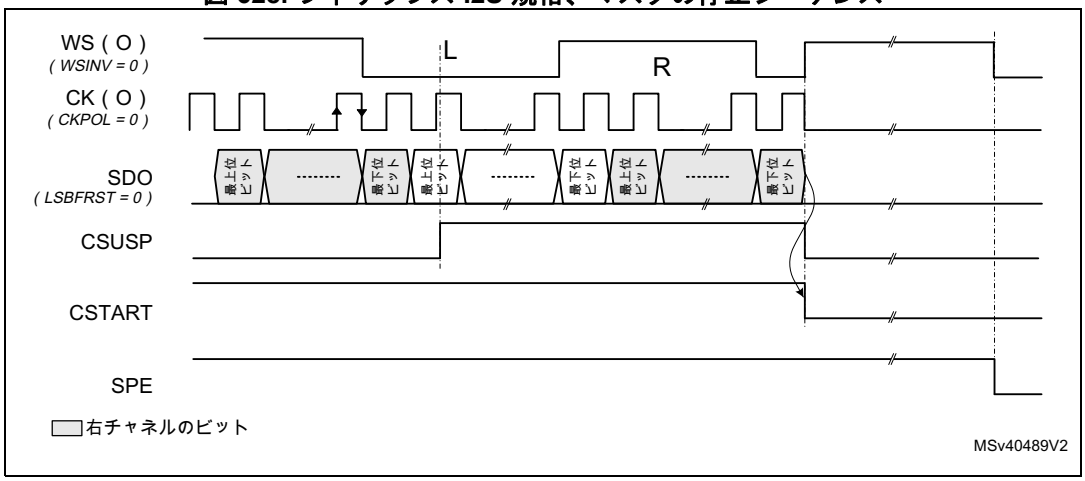
50.9.8 停止シーケンス

アプリケーションは SPE ビットを 0 に設定することによって I2S/PCM 転送を停止できます。その場合、現在のフレームの終了を待たずに通信は直ちに停止します。

マスタモードでは、現在のフレームの終了時点で I2S/PCM の転送を停止することもできます。その目的のために、CSUSP ビットを 1 にセットして、CSTART ビットが 0 に変わるまでポーリングする必要があります。CSTART ビットは、現在のステレオ（I2S モードが選択されていた場合）またはモノラルサンプルが完全にシフトイン／アウトされるとき、0 に変わります。その後、SPE ビットを 0 にセットすることができます。

図 628 に、マスタモードの場合の停止シーケンスの例を示します。左サンプルの送信中に CSUSP ビットが 1 にセットされ、右サンプルの最後のビットが転送されるまで転送は続行されます。その後、CSTART および CSUSP は 0 に戻り、CK と WS 信号が非アクティブ状態に戻り、そして SPE を 0 に設定できます。

図 628. フィリップス I2S 規格、マスタの停止シーケンス

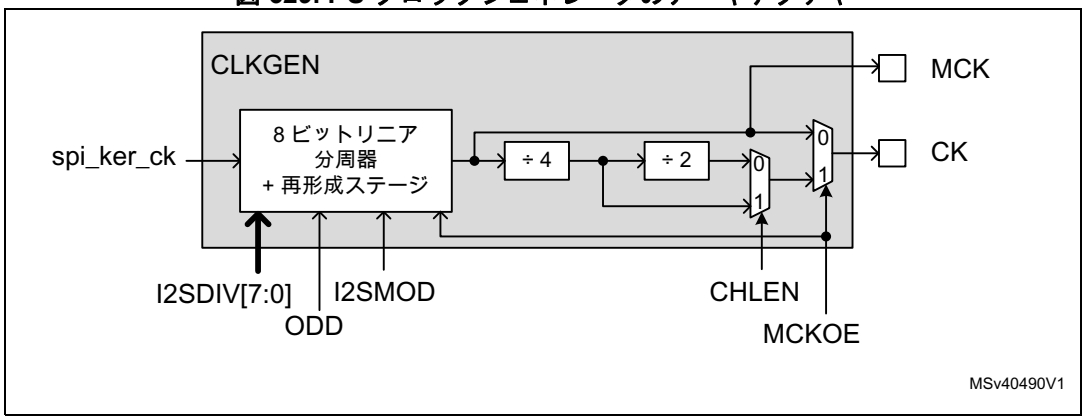


注： ***スレーブモードでは、停止シーケンスは SPE ビットによってのみ制御されます。***

50.9.9 クロックジェネレータ

I2S または PCM をマスタモードに設定する場合、フレーム同期（WS）、ビットクロック（CK）、およびマスタクロック（MCK）を希望の周波数で生成するためにクロックジェネレータをプログラムする必要があります。
I2S または PCM をスレーブモードで使用する場合は、クロックジェネレータを設定する必要はありません。

図 629. I^2S クロックジェネレータのアーキテクチャ



MCK、CK、および WS に生成される周波数は主に I2SDIV、ODD、CHLEN、および MCKOE に依存します。MCKOE ビットはマスタクロックの生成の要否を示します。マスタクロックはフレーム同期よりも 256 倍高い周波数を持ちます。このマスタクロックは外部オーディオコーデックにリファレンスクロックを提供するのにしばしば必要となります。

注： ***マスタモードでは、バスクロック速度（F_{pclk}）とビットクロック（F_{CK}）の比には特に制約はありません。バスクロック周波数は、データのスループットをサポートするために十分高くなければなりません。***

マスタクロックが生成される場合（MCKOE = 1）、フレーム同期の周波数は I2S モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{256 \times \{(2 \times I2SDIV) + ODD\}}$$

PCM モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{128 \times \{(2 \times I2SDIV) + ODD\}}$$

さらに、MCK（F_{MCK}）の周波数は次の式によって与えられます。

$$F_{MCK} = \frac{F_{i2s_clk}}{\{(2 \times I2SDIV) + ODD\}}$$

マスタクロックが無効の場合（MCKOE = 0）、フレーム同期の周波数は I2S モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{32 \times (CHLEN + 1) \times \{(2 \times I2SDIV) + ODD\}}$$

PCM モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{16 \times (CHLEN + 1) \times \{(2 \times I2SDIV) + ODD\}}$$

ここで、F_{WS} はフレーム同期の周波数、F_{i2s_clk} は SPI/I2S ブロックに供給されるカーネルクロックの周波数です。

注： ***CHLEN および ODD は 0 または 1 です。***
ODD が 0 のとき、I2SDIV は 0 から 255 までの任意の値をとることができますが、ODD = 1 のとき、I2SDIV = 1 の値は許されません。

I2SDIV = 0 のとき、{(2 x I2SDIV) + ODD} は 1 に強制されます。

注： ***{(2 x I2SDIV) + ODD} が奇数のとき、MCK または CK 信号のデューティサイクルは 50% になりません。奇数の比を使用する場合は、セットアップおよびホールド時間のマージンに影響を与える可能性があるので注意が必要です。たとえば、{(2 x I2SDIV) + ODD} = 5 の場合、デューティサイクルは 40% になります。***

表 393 に、I2S モードに対するクロックジェネレータのプログラミング例を示します。

MCK の生成

マスタクロック MCK は、SPE ビットに関係なく生成できます。MCK の生成は以下のビットによって制御されます。

- I2SMOD は 1 でなければなりません。
- I2SCFG はマスタモードを選択する必要があります。
- MCKOE は 1 にセットする必要があります。

表 393. 通常の I2S 周波数での CLKGEN のプログラミング例

i2s_clk（MHz）	チャネル長（ビット）	I2SDIV	ODD	MCK	サンプリングレート：F_{WS}（kHz）
12.288	16	12	0	無	16
12.288	32	6	0		16
12.288	16	6	0		32
12.288	32	3	0		32
49.152	16	16	0		48
49.152	32	8	0		48
49.152	16	8	0		96
49.152	32	4	0		96
49.152	16	4	0		192
49.152	32	2	0		192
4.096	16 または 32	0	-	有	16
24.576	16 または 32	3	0		
49.152	16 または 32	3	0		32
12.288	16 または 32	0	-		48
49.152	16 または 32	2	0		
61.44	16 または 32	2	1		
98.304	16 または 32	2	0		96
196.608	16 または 32	2	0		192

50.9.10 内部 FIFO

I2S インタフェースは RX および TX パス用の専用の FIFO を使用できます。送信サンプルは SPI2S_TXDR レジスタを経由して TxFIFO に書き込むことができます。RxFIFO からの読み出しは SPI2S_RXDR レジスタを経由して実行されます。

データの配置と順序

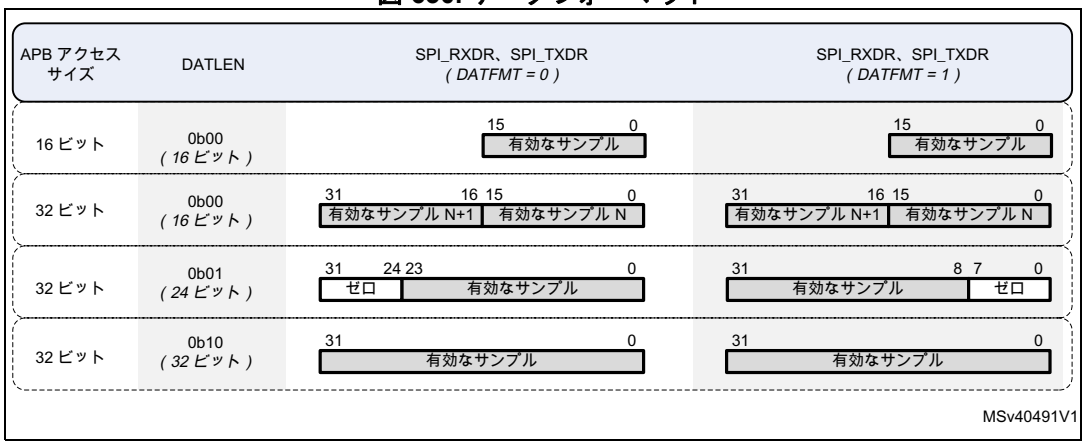
DATFMT ビットによって、SPI2S_RXDR および SPI2S_TXDR レジスタ内のデータ配置を選択することができます。

また、SPI2S_RXDR または SPI2S_TXDR に配置されるデータのフォーマットは、それらのレジスタが APB バス経由でアクセスされる方法にも依存することに注意してください。

図 630 は APB アクセスサイズ、DATFMT、および DATLEN の間で可能な設定を示します。

注： ***APB アクセスサイズが 32 ビットで、DATLEN = 0 の場合は注意が必要です。読み出し動作では、RxFIFO には少なくとも 2 つのデータが存在する必要があります。そうしないと、読み出しデータは不正になります。同様に書き込み動作では、TxFIFO には少なくとも 2 つの空き場所が存在する必要があります。そうしないと、データは失われる可能性があります。***

図 630. データフォーマット



1. I2S モードでは、サンプル N は左サンプルを表し、サンプル N+1 は右サンプルを表します。

プログラム可能な FIFO 閾値レベルに従って、割り込みまたは DMA リクエストを生成することができます。FIFO 閾値は RXFIFO と TxFIFO で共通で、FTHVL によって調節できます。

I2S モードでは、左と右のオーディオサンプルは FIFO に交互に入ります。つまり、送信操作では、TxFIFO を左サンプルで埋めることから開始し、続いて右サンプルというように交互に充填する必要があります。受信モードでは、RxFIFO から読み出される最初のデータは左チャネルを表すと考えられ、次のデータが右チャネルとなります。

FIFO の読み出し、書き込みのポインタは、SPE ビットが 0 にセットされると、リセットされることに注意してください。

さらに詳しい情報については、*セクション 50.9.11* および *セクション 50.9.15* を参照してください。

FIFO サイズの最適化

FIFO の基本要素はバイトです。これにより、FIFO の場所を最適化できます。たとえば、データサイズが 24 ビットに固定されているとき、各オーディオサンプルは 3 つの FIFO 基本要素を占めます。

たとえば、16 基本要素の FIFO の深さは以下のようになります。

- DATLEN = 0（16 ビット）の場合、8 サンプル
- DATLEN = 1（24 ビット）の場合、5 サンプル
- DATLEN = 2（32 ビット）の場合、4 サンプル

50.9.11 FIFO ステータスフラグ

アプリケーションが I2S インタフェースの状態を完全に監視できるように、2 つのステータスフラグが用意されています。両方のフラグは割り込みリクエストを生成できます。RXPIE ビットが有効化されている場合、受信割り込みが生成され、TXPIE ビットが有効化されている場合、送信割り込みが生成されます。これらのビットは SPI_IER レジスタに配置されています。

TxFIFO 閾値到達（TXP）

このフラグがセットされると、TxFIFO には少なくとも FTHVL の空き領域があることを示しています。したがって送信される新しい FTHVL 個のデータが SPI2S_TXDR に書き込めます。空き容量が FTHVL より少なくなると TXP フラグがリセットされます。I2S が無効にされている（SPE ビットがリセット）とき、TXP = 1 であることに注意してください。

RxFIFO 閾値到達（RXP）

このフラグがセットされると、RxFIFO に少なくとも FTHVL の有効なデータがあることを示しています。したがって SPI2S_RXDR を経由してそれらのデータを読み出すことができます。RXFIFO 内のデータが FTHVL より少なくなると、このフラグはリセットされます。

I2S モードでの割り込み機能についての詳しい情報は*セクション 50.10* を参照してください。

50.9.12 アンダーラン状況の取り扱い

送信モードで、TxFIFO がすでに空きになっているときに、新しいデータをシフトレジスタにロードする必要がある場合に、UDR フラグがセットされます。そのような状況では、少なくとも 1 つのデータが失われます。

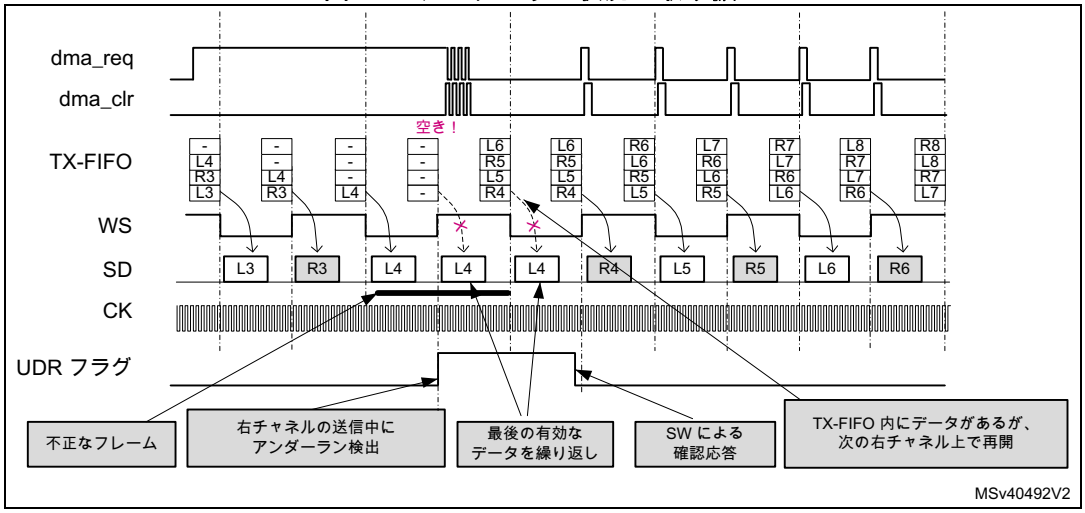
I2S モードでは、誤配置状況（左右チャネルの入れ替わり）を防止するためのハードウェアメカニズムがあります。次の図に示すように、アンダーランが起きると、再開始条件が満たされない限り、ペリフェラルは左と右のチャネルの有効な最後のデータをリプレイします。次の場合に送信が再開されます。

– TxFIFO に十分なデータがあるとき、および、
– UDR フラグがソフトウェアによってクリアされたとき。

その後、次に送信されるデータは以下のとおりです。

– 右チャネルのデータを送信する必要があった時にアンダーランが発生した場合は、右チャネル、または、
– 左チャネルのデータを送信する必要があった時にアンダーランが発生した場合は、左チャネル。

図 631. アンダーラン状況の取り扱い



SPI_IER レジスタの UDRIE ビットがセットされている場合、UDR フラグは割り込みをトリガできます。UDR ビットは、SPI_IFCR レジスタの UDRC ビットに 1 を書き込むことによってクリアされます。

ブロックを PCM モードに設定した場合、この再配置メカニズムはアクティブ化されません。

注： ***アンダーラン状況は、マスタまたはスレーブモードで起きる可能性があります。マスタモードでは、アンダーランが起きると、WS、CK、および MCK 信号はゲートされません。***

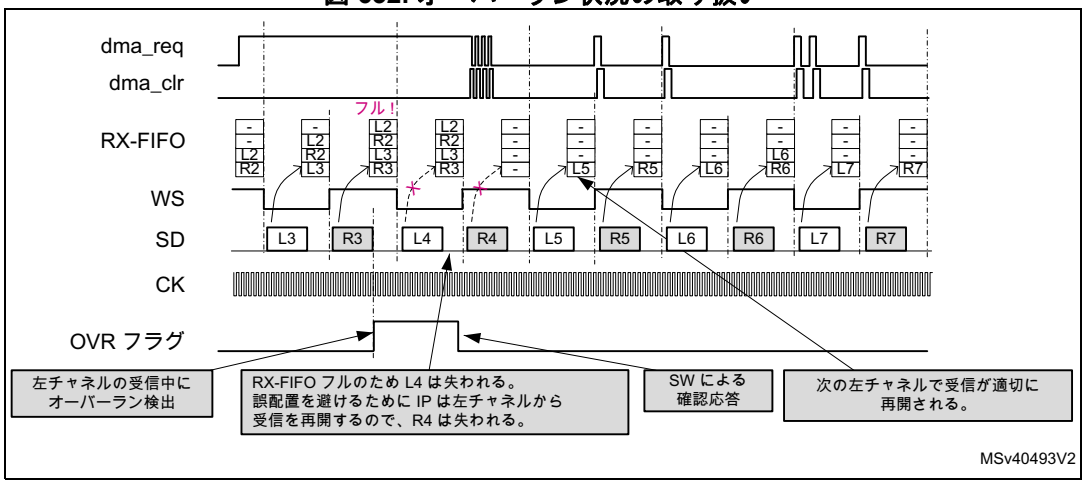
再同期のために、UDR フラグの変化は少なくとも CK クロックの 2 周期後にハードウェアによって考慮されます。

50.9.13 オーバーラン状況の取り扱い

RxFIFO がすでにフルになっているときに、受信データを RxFIFO に書き込む必要がある場合に、OVR フラグがセットされます。結果として、いくつかの受信データは失われます。

I2S モードでは、誤配置状況（左右チャネルの入れ替わり）を防止するためのハードウェアメカニズムがあります。次の図に示すように、オーバーランが起きると、再開始条件が満たされない限り、ペリフェラルはデータを RxFIFO に書き込むことを停止します。RxFIFO に十分な空きができて、OVR フラグがクリアされると、ブロックは、右チャネルのデータを受信した時にオーバーランが起きた場合は次の右チャネルを RxFIFO に書き込むことから開始し、左チャネルのデータを受信した時にオーバーランが起きた場合は次の左チャネルを書き込むことから開始します。

図 632. オーバーラン状況の取り扱い



SPI_IER レジスタの OVRIE ビットがセットされている場合、割り込みが生成されることがあります。OVR ビットは、SPI_IFCR レジスタの OVRC ビットに 1 を書き込むことによってクリアされます。

ブロックを PCM モードに設定した場合、この再配置メカニズムはアクティブ化されません。

注： ***オーバーラン状況は、マスタまたはスレーブモードで起きる可能性があります。マスタモードでは、オーバーランが起きると、WS、CK、および MCK 信号はゲートされません。***

50.9.14 フレームエラー検出

スレーブモードに設定されている場合、SPI/I2S ブロックは次の 2 種類のフレームエラーを検出します。

- 前のデータのシフトインまたはシフトアウトが完了していない時に受信したフレーム同期（早期フレームエラー）。このモードは、FIXCH = 0 によって選択されます。
- 想定外の位置で発生するフレーム同期。このモードは、FIXCH = 1 によって選択されます。

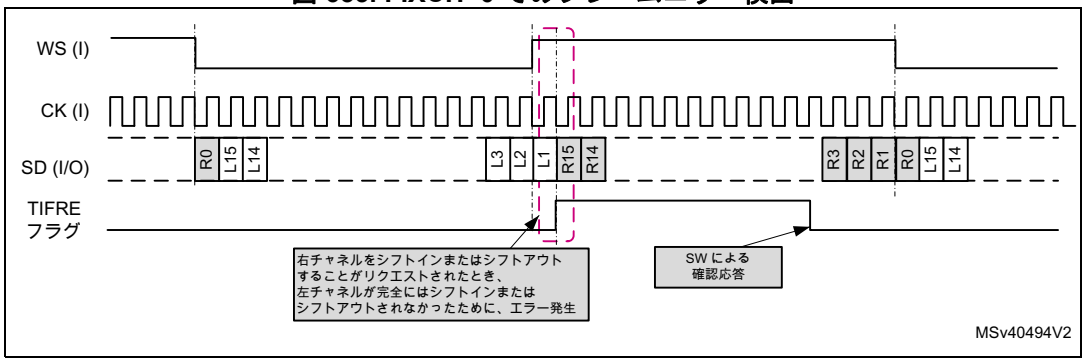
スレーブモードでは、外部のマスタデバイスによって提供されるフレーム長が 32 または 64 ビットと異なっている場合、FIXCH を 0 に設定する必要があります。SPI/I2S は各転送を WS で同期するので、ミスアライメントのリスクはありませんが、雑音の多い環境では、CK 信号にグリッチが起きた場合、サンプルは影響を受け、アプリケーションはこれに気付きません。

外部のマスタデバイスによって提供されるフレーム長が 32 または 64 ビットに等しい場合、FIXCH を 1 にセットすることができ、それに従って CHLEN を調節できます。SPI/I2S は各転送を WS で同期するので、同じくミスアライメントのリスクはありませんが、各チャネル境界間のビットクロックの量が CHLEN と異なっていた場合、フレームエラーフラグ（TIFRE）が 1 にセットされます。

図 633 に、フレームエラー検出の例を示します。SPI/I2S ブロックはスレーブモードで、左チャネル用のビットクロック周期の数がデータをシフトインまたはシフトアウトするのに十分ではありませ

ん。図は、WS 信号への整合を維持するために、実行中の転送が中断されて、次の転送が開始されたことを示しています。

図 633. FIXCH=0 でのフレームエラー検出

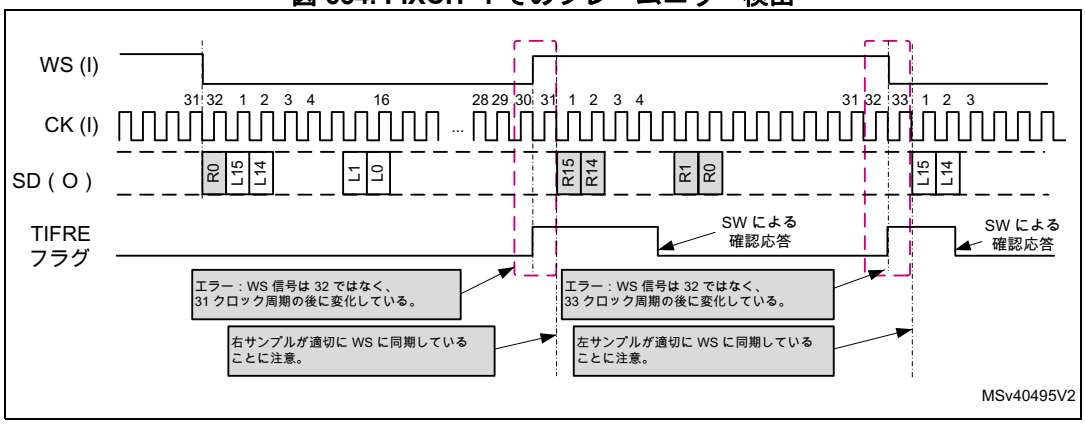


TIFREIE ビットをセットすれば、割り込みを生成させることができます。フレームエラーフラグ（TIFRE）は、SPI_IFCR レジスタの TIFREC ビットに 1 を書き込むことによってクリアされます。

FIXCH ビットを 1 にセットすることにより、フレームエラーフラグの適用範囲を拡張することができます。このビットを 1 にセットすると、SPI/I2S はスレーブモードで固定チャネル長を想定します。つまり、想定されるチャネル長は CHLEN に従って 16 または 32 ビットの可能性があるということです。*図 634* に示すように、このモードでは SPI/I2S ブロックは、WS 信号が予想時点で変化するかどうか（早すぎるか遅すぎるか）を検出することができます。

注： ***図 633 および図 634 はスレーブ送信モードに対するメカニズムを示していますが、これはスレーブ受信およびスレーブ全二重に対してもあてはまります。***

図 634. FIXCH=1 でのフレームエラー検出



フレームエラー検出は、一般に WS や CK 信号の良好な受信を乱す雑音の多い環境が原因になる可能性があります。

注： ***オーバーランおよび早期フレームが同じフレーム内で起きた場合は、SPI/I2S は適切に回復することができません。この場合は、SPI/I2S を無効にして再度有効化する必要があります。***

50.9.15 DMA インタフェース

I2S/PCM モードは SPI 機能と同じ DMA リクエストラインを共有します。送信パスと受信パスに対して別々の DMA チャネルがあります。各 DMA チャネルは、SPI_CFG1 レジスタの RXDMAEN と TXDMAEN ビットを介して有効にできます。

受信モードでは、DMA インタフェースは以下のように動作します。

1. ハードウェアが RxFIFO レベルを評価します。
2. RxFIFO に少なくとも FTHVL 個のサンプルが存在する場合、FTHVL DMA リクエストが生成されます。
   – FTHVL DMA リクエストが完了すると、ハードウェアは手順 1 に戻ります。
3. RxFIFO に FTHVL より少ないサンプルしかない場合、DMA リクエストは生成されず、ハードウェアは手順 1 に戻ります。

送信モードでは、DMA インタフェースは以下のように動作します。

1. ハードウェアが TxFIFO レベルを評価します。
2. TxFIFO に少なくとも FTHVL 個の空き場所が存在する場合、FTHVL DMA リクエストが生成されます。
   – FTHVL DMA リクエストが完了すると、ハードウェアは手順 1 に戻ります。
3. TxFIFO に FTHVL より少ない空き場所しかない場合、DMA リクエストは生成されず、ハードウェアは手順 1 に戻ります。

50.9.16 プログラミングの例

フィリップス I2S 規格のマスタ送信

この例では、48 kHz のサンプリングレート、マスタクロックを使用して、フィリップス I2S 規格のマスタ送信モードをサポートするインタフェースのプログラム方法を示します。ここでは、SPI/I2S は回路のクロックコントローラから 61.44 MHz のカーネルクロック（i2s_clk）を受信していることを想定します。

開始手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPI/I2S ブロックが適切にカーネル周波数（この例では 61.44 MHz）を受信していることを確認します。
3. SPE が 0 に設定されていることを確認します。
4. MCK クロックを提供し、フレーム同期レートを正確に 48 kHz に設定するために、クロックジェネレータをプログラムします。よって、I2SDIV = 2、ODD = 1、および MCKOE = 1。
5. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、CHLEN = 1（チャネルあたり 32 ビット）、DATLEN = 1（24 ビット）、I2SSTD = 0（フィリップス規格）、I2SCFG = 2（マスタ送信）、I2S/PCM モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを更新する必要があります。
6. 希望する値を FTHVL に設定することによって FIFO 閾値を調節します。たとえば、2 つのオーディオサンプルの閾値が必要な場合は FTHVL = 1 とします。
7. すべてのステータスフラグレジスタをクリアします。
8. UDRIE のような、割り込みを生成するべきフラグを有効にします。TIFRE はマスタモードでは意味を持たないことに注意してください。
9. データ転送が DMA を使用する場合は、
   – DMA ペリフェラルをプログラムします。
   – メモリバッファを有効なオーディオサンプルで初期化します。
   – DMA チャネルを有効にします。
10. データ転送を割り込み経由で行う場合は、TXPIE ビットを 1 にセットして割り込みを有効にする必要があります。
11. SPE を 1 にセットします。このビットが 1 にセットされるとすぐに、以下の動作が行われます。
    – 割り込み生成が有効になっている場合は、SPI/I2S は、割り込みハンドラが TxFIFO を充填できるように割り込みリクエストを生成します。
    – DMA 転送が有効になっていた場合は（TXDMAEN = 1）、SPI/I2S は、TxFIFO を充填するために DMA リクエストを生成します。
12. 最後に、シリアルインタフェースを有効化する前に、TxFIFO が空でないことを確認する必要があります。これは、インタフェースが有効化される時にアンダーラン状態を避けるためには重要です。その後、CSTART ビットを 1 にセットすることによって SPI/I2S ブロックを有効にできます。CSTART ビットは SPI_CR1 レジスタにあります。

停止手順（マスタモード）

1. 実行中の転送を停止するために CSUSP ビットを 1 にセットします。
2. CSTART ビットの値をチェックして、0 になるまで待ちます。
3. DMA ペリフェラル、バスクロック、等を停止します。
4. SPI/I2S ブロックを無効にするために SPE ビットを 0 にセットします。

マスタ I2S MSB 詰め、全二重

この例では、マスタ全二重モードで、48 kHz のサンプリングレート、マスタクロックを使用しないで、I2S MSB 詰めプロトコルをサポートするインタフェースのプログラム方法を示します。SPI/I2S は回路のクロックコントローラから 12.288 MHz のカーネルクロック（i2s_clk）を受信していると想定します。

手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPI/I2S ブロックが適切にカーネル周波数（この例では 12.288 MHz）を受信していることを確認します。
3. SPE が 0 にセットされていることを確認します。
4. MCK クロックを提供し、フレーム同期レートを正確に 48 kHz に設定するために、クロックジェネレータをプログラムします。よって、I2SDIV = 2、ODD = 0、および MCKOE = 0。
5. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、
   CHLEN = 1（チャネルあたり 32 ビット）、DATLEN = 1（24 ビット）、I2SSTD = 1（MSB 詰め）、I2SCFG = 5（マスタ全二重）、I2S/PCM モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを更新する必要があります。
6. 希望する値を FTHVL に設定することによって FIFO 閾値を調節します。たとえば、2 つのオーディオサンプルの閾値が必要な場合は FTHVL = 1 とします。
7. すべてのステータスフラグレジスタをクリアします。
8. UDRIE のような、割り込みを生成するべきフラグを有効にします。TIFRE はマスタモードでは意味を持たないことに注意してください。
9. データ転送が DMA を使用する場合は、
   - DMA ペリフェラルを次のようにプログラムします：2 チャネルで、1 つは受信用、1 つは送信用。
   - メモリバッファを送信パス用の有効なオーディオサンプルで初期化します。
   - DMA チャネルを有効にします。
   - SPI/I2S ブロックで、TXDMAEN および RXDMAEN ビットを 1 にセットすることによって DMA を有効にします。これらのビットが 1 にセットされるとすぐに、SPI/I2S は DMA リクエストを送信することによって TxFIFO を充填し始めます。
10. データ転送を割り込み経由で行う場合は、TXPIE および RXPIE ビットを 1 にセットして割り込みを有効にする必要があります。
11. SPE を 1 にセットします。このビットが 1 にセットされるとすぐに、以下の動作が行われます。
    - 割り込み生成が有効になっていた場合は、SPI/I2S は、割り込みハンドラが TxFIFO を充填できるように割り込みリクエストを生成します。
    - DMA 転送が有効な場合、SPI/I2S は、TxFIFO を充填するために DMA リクエストを生成します。
12. 最後に、シリアルインタフェースを有効化する前に、TxFIFO が空でないことを確認する必要があります。これは、インタフェースが有効化される時にアンダーラン状態を避けるためには重要です。その後、CSTART ビットを 1 にセットすることによって SPI/I2S ブロックを有効にできます。CSTART ビットは SPI_CR1 レジスタにあります。

停止シーケンスについての詳細は、*停止手順（マスタモード）*を参照してください。

50.9.17 フィリップス I2S 規格のスレーブ受信

この例では、48 kHz のサンプリングレートで、フィリップス I2S 規格プロトコルのスレーブレシーバモードをサポートするインタフェースのプログラム方法を示します。スレーブモードでは、SPI/I2S ブロックは受信サンプルのサンプリングレートを制御できないことに注意してください。この例では、外部マスタデバイスが 24 ビットのチャネル長の I2S フレーム構造を配信していると想定します。したがって、FIXCH が 1 にセットされているときのフレームエラー検出のために提供されている機能は使用できません。

手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPE が 0 にセットされていることを確認します。
3. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、FIXCH = 0（チャネル長が 16 および 32 ビットと異なっているため）、DATLEN = 0（16 ビット）、I2SSTD = 0（フィリップスプロトコル）、I2SCFG = 1（スレーブ受信）、I2S モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを適切にプログラムする必要があります。
4. 希望する値を FTHVL に設定することによって FIFO 閾値を調節します。たとえば、2 つのオーディオサンプルの閾値が必要な場合は FTHVL = 1 とします。
5. すべてのステータスフラグレジスタをクリアします。
6. UDRIE や TIFRE のような、割り込みを生成するべきフラグを有効にします。
7. データ転送が DMA を使用する場合は、
   – DMA ペリフェラルを、1 つの受信チャネルにプログラムします。
   – DMA チャネルを有効にします。
   – SPI/I2S ブロックで、RXDMAEN ビットを 1 にセットすることによって DMA を有効にします。
8. データ転送を割り込み経由で行う場合は、RXPIE ビットを 1 にセットして割り込みを有効にする必要があります。
9. SPE を 1 にセットします。
10. 最後に、シリアルインタフェースを有効にするために CSTART ビットを 1 にセットすることができます。外部マスタデバイスによる左データの送信が次に発生した時に、SPI/I2S は RxFIFO へのデータの格納を開始します。

停止手順（スレーブモード）

1. SPI/I2S ブロックを無効にするために SPE ビットを 0 にセットします。
2. DMA ペリフェラル、バスクロック、等を停止します。

50.10 I2S ウェイクアップおよび割り込み

PCM/I2S モードでは、*表 394* に記載されたイベントに従って割り込み（**spi_it**）またはウェイクアップイベント信号（**spi_wkup**）を生成することができます。

割り込みイベントは個別に有効または無効にできます。

表 394. I2S 割り込みリクエスト

割り込みイベント	イベントフラグ	有効制御ビット	イベントのクリア方法	有効になる割り込み／ウェイクアップ	
				spi_it	spi_wkup
TxFIFO 閾値到達	TXP	TXPIE	TXP フラグは、TxFIFO の空き領域が FTHVL より少なくなった時にクリアされます。	はい	はい
RxFIFO 閾値到達	RXP	RXPIE	RXP フラグは、RxFIFO のサンプルが FTHVL より少なくなった時にクリアされます。		
オーバーランエラー	OVR	OVRIE	OVR は、OVRC に 1 を書き込むことによってクリアされます。		
アンダーランエラー	UDR	UDRIE	UDR は、UDRC に 1 を書き込むことによってクリアされます。		
フレームエラーフラグ	TIFRE	TIFREIE	TIFRE は、TIFREC に 1 を書き込むことによってクリアされます。		いいえ

50.11 SPI/I2S レジスタ

50.11.1 SPI/I2S 制御レジスタ 1 （SPI2S_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRCINI	RCRCINI	CRC33_17	SSI	HDDIR	CSUSP	CSTART	MASRX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
rw	rw	rw	rw	rw	w	rs	rw								rw

ビット 31:17 予約済み

ビット 16 **IOLOCK**：関連 IO の AF 設定のロック

このビットは、ソフトウェアでセットされ、次回のデバイスリセット時にハードウェアによってクリアされます。

0：AF 設定はロックされません。

1：AF 設定はロックされます。

このビットをセットすると、SPI_CFG2 レジスタの内容はそれ以降、変更できません。

このビットは SPI が無効なときに設定してください。SPE=1 のときは、書き込み保護されています。

ビット 15 **TCRCI**：トランスミッタ用 CRC 計算の初期化パターン制御

0：すべてゼロのパターンを適用します。

1：すべて 1 のパターンを適用します。

ビット 14 **RCRCI**：レシーバ用 CRC 計算の初期化パターン制御

0：すべてゼロのパターンを適用します。

1：すべて 1 のパターンを適用します。

ビット 13 **CRC33_17**：32 ビットの CRC 多項式設定

0：フルサイズ（33 ビットまたは 17 ビット）の CRC 多項式は使用されません。

1：フルサイズ（33 ビットまたは 17 ビット）の CRC 多項式が使用されます。

ビット 12 **SSI**：内部 SS 信号入力レベル

このビットは、SSM ビットがセットされているときにのみ有効です。このビットの値は強制的に ペリフェラルの SS 入力に設定され、SS ピンの I/O 値は無視されます。

ビット 11 **HDDIR**：半二重モードでの Rx/Tx の方向

半二重の設定において、HDDIR ビットは Rx/Tx のデータ転送方向を確定します。このビットは、全二重または単方向の設定では無視されます。

0：SPI はレシーバです。

1：SPI はトランスミッタです。

ビット 10 **CSUSP**：マスタのサスペンドリクエスト

このビットは 0 として読み出されます。

マスタモードにおいて、このビットがソフトウェアによってセットされると、現在のフレームの終了時点で CSTART ビットがリセットされ、SPI 通信がサスペンドされます。フレームのトランザクションの終了を確認するために、ユーザは SUSP フラグをチェックする必要があります。

SPI を無効にする前、または低電力モードに移行する前に、（このビットを使用するか、または TXDR を空に保持して）マスタモードの通信をサスペンドする必要があります。

ビット 9 **CSTART**：マスタ転送の開始

このビットは、マスタモードで SPI の転送を開始するためにソフトウェアによってセットされます。これは、転送の終了（EOT）フラグがセットされた時、または CSUSP リクエストが受理された時に、ハードウェアによってクリアされます。

0：マスタ転送はアイドル状態です。

1：マスタ転送は動作中、または自動サスペンドによって一時的にサスペンドされています。

SPI モードでは、CSTART は SPE=1 および MASTER=1 のときのみセットできます。

SPI モードでは、マスタ送信が有効になっている場合、送信 FIFO でデータが利用可能になった場合のみ、通信が開始、または続行します。

I2S/PCM モードでは、CSTART は SPE = 1 のときセットできます。

ビット 8 **MASRX**：受信モードでのマスタ自動 SUSP

このビットは、ソフトウェアによってセット／クリアされ、オーバーラン状態を避けるために、マスタレシーバモードで SPI の連続転送と自動管理を制御します。

0：オーバーラン状態にかかわらず、SPI のフロー／クロック生成は続行します（データは失われます）。

1：オーバーラン状態になる前に、RxFIFO がフル状態になった時、SPI フローは一時的にサスペンドされます。SPI 通信がサスペンドされたとき、SUSP フラグがセットされます。

オーバーラン状態を防ぐために SPI 通信がサスペンドされたとき、内部の同期遅延のために次のフレームの数ビットがすでにクロックアウトされていることが起きる可能性があります。RxFIFO が読み出されると、通信が再開され、その後の制約もなく、その後に続くビットのトランザクションから続行されます。

同じ理由により、データのサイズが 8 ビット未満のとき、自動サスペンドは完全に信頼できるわけではありません。この場合、MIDI パラメータが非ゼロ値であるときに適用される、データフレーム間に挿入された遅延時間と組み合わせることによって安全なサスペンドが達成されます。データサイズとインターリーブされた SPI サイクルの合計により、少なくとも SPI クロックの 8 周期の長さの間隔が常に生成されます。

ビット 7:1 予約済み

ビット 0 **SPE**：シリアルペリフェラル有効

このビットは、ソフトウェアによってセット／クリアされます。

0：シリアルペリフェラルは無効です。

1：シリアルペリフェラルは有効です。

SPE=1 のとき、SPI データ転送は有効になり、設定レジスタおよび SPI_CR1 の IOLOCK ビットは書き込み保護になります。それらは SPE=0 のときのみ、変更できます。

SPE=0 のとき、SPI 動作は停止して無効になり、内部ステートマシンはリセットされ、すべての FIFO の内容は一掃され、CRC 計算は初期化され、受信データレジスタはゼロが読み出されます。

MODF エラーフラグがアクティブになっているとき、SPE はセットできません。

50.11.2 SPI 制御レジスタ 2（SPI_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSER[15:0]															
rs															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIZE[15:0]															
rw															

ビット 31:16 **TSER[15:0]**：前のTSIZE に格納されたデータ数のデータがトランザクションされた直後に、TSIZE に再ロードされるデータ転送拡張のデータ数

このレジスタは、その内容がクリアされたときのみ、ソフトウェアによってセットできます。TSIZE の再ロードが行われると、ハードウェアによってクリアされます。TSER 値は、CTSIZE カウンタがゼロになる前に、事前にプログラムする必要があります。そうしないと、再ロードされず、トラフィックは通常の EOT イベントとして終了します。

ビット 15:0 **TSIZE[15:0]**：現在の転送でのデータ数

これらのビットは、ソフトウェアによって変更されます。この値は、CSTART ビットがセットされている時は、変更してはいけません。

TSIZE にゼロ値が格納されているときに CSTART がセットされると、終わりのないトランザクションが開始されます。CRC が有効な場合、TSIZE は 0xFFFF の値にセットすることはできません。

50.11.3 SPI 設定レジスタ 1 （SPI_CFG1）

アドレスオフセット：0x08

リセット値：0x0007 0007

このレジスタの内容は、SPI が有効になっているとき、書き込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	CRCSIZE[4:0]				
	rw								rw		rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXDMAEN	RXDMAEN	Res.	UDRDET[1:0]		UDRCFG[1:0]		FTHLV[3:0]				DSIZE[4:0]				
							rw				rw				

ビット 31 予約済み

ビット 30:28 **MBR [2:0]：**マスタボーレート

000：SPI マスタクロック／2
001：SPI マスタクロック／4
010：SPI マスタクロック／8
011：SPI マスタクロック／16
100：SPI マスタクロック／32
101：SPI マスタクロック／64
110：SPI マスタクロック／128
111：SPI マスタクロック／256

ビット 27:23 予約済み

ビット 22 **CRCEN：**ハードウェア CRC 計算有効

0：CRC 計算は無効です。
1：CRC 計算は有効です。

ビット 21 予約済み

ビット 20:16 **CRCSIZE [4:0]：**トランザクションおよび比較される CRC フレームの長さ

CRC 結果がトランザクションまたは比較されるとき、多項式の計算から最上位の数ビットが考慮されます。多項式の長さは、この設定の影響を受けません。

00000：未使用
00001：未使用
00010：未使用
00011：4 ビット
00100：5 ビット
00101：6 ビット
00110：7 ビット
00111：8 ビット
.....
11101：30 ビット
11110：31 ビット
11111：32 ビット

値は、DSIZE の長さと同じか、倍数である必要があります。

注：　CRCSIZE ビットフィールドの最上位ビットは、データサイズが 16 ビットに限定されているペリフェラルのインスタンスで予約済みです。

ビット 15 **TXDMAEN：**Tx DMA ストリーム有効

0：Tx DMA は無効です。
1：Tx DMA は有効です。

ビット 14 **RXDMAEN：**Rx DMA ストリーム有効

0：Rx-DMA は無効です。
1：Rx-DMA は有効です。

ビット 13 予約済み

ビット 12:11 **UDRDET [1:0]：**スレーブトランスミッタでのアンダーラン状態の検出

00：アンダーランはデータフレームの開始時に検出されます（最初のビットの保護はなし）。
01：アンダーランは最後のデータフレームの終了時に検出されます。
10：アンダーランはアクティブな SS 信号の開始時に検出されます。
11：予約済み

スレーブレシーバでアンダーラン状態をいつ、どのように検出するかを、ここで定義できます。

ビット 10:9 **UDRCFG [1:0]**：アンダーラン状態でのスレーブトランスミッタの動作

00：スレーブは、ユーザによって SPI_UDRDR レジスタに定義された一定のパターンを送信します。

01：スレーブは、マスタから最後に受信したデータフレームを繰り返します。

10：スレーブは、最後に送信したデータフレームを繰り返します。

11：予約済み

TxFIFO が空のとき、送信シフトレジスタは SPI_UDRDR レジスタの値を保持します。このレジスタでは、ユーザに定義された値を送信するように固定したり、マスタから受信した最後のトランザクションフレームによって自動的に更新したり、スレーブ TxFIFO に最後に（TXDR に書き込むことによって）格納されたデータによって更新したりすることができます。

ビット 8:5 **FTHVL [3:0]**：FIFO 閾値レベル

1 つのデータパケットのデータフレーム数を定義します。パケットのサイズは FIFO スペースの 1/2 を超えてはいけません。

0000：1 データ
0001：2 データ
0010：3 データ
0011：4 データ
0100：5 データ
0101：6 データ
0110：7 データ
0111：8 データ
1000：9 データ
1001：10 データ
1010：11 データ
1011：12 データ
1100：13 データ
1101：14 データ
1110：15 データ
1111：16 データ

以下のように、設定されたパケットサイズがデータレジスタのアクセス並行度に整合する場合、SPI インタフェースはより効率的になります。

– SPI データレジスタが 16 ビットレジスタとしてアクセスされ、DSIZE<=8 ビットの場合、FTHVL=2、4、6 等を選択することを推奨します。
– SPI データレジスタが 32 ビットレジスタとしてアクセスされ、DSIZE>8 ビットの場合、FTHVL=2、4、6 等を選択し、DSIZE<=8 ビットの場合は FTHVL=4、8、12 等を選択するのがよいです。

ビット 4:0 **DSIZE [4:0]**：1 つの SPI データフレームのビット数

00000：未使用
00001：未使用
00010：未使用
00011：4 ビット
00100：5 ビット
00101：6 ビット
00110：7 ビット
00111：8 ビット
.....
11101：30 ビット
11110：31 ビット
11111：32 ビット

注： DSIZE ビットフィールドの最上位ビットは、データサイズが 16 ビットに限定されているペリフェラルのインスタンスで予約済みです。

50.11.4 SPI 設定レジスタ 2 （SPI_CFG2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このレジスタの内容は、SPI が有効になっているか、または SPI2S_CR1 で IOLOCK ビットがセットされているとき、書き込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFCNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSBFRST	MASTER	SP[2:0]			COMM[1:0]		Res.
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw			rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOSWP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIDI [3:0]				MSSI [3:0]			
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **AFCNTR**：GPIO オルタネート機能制御

このビットは、SPE=0 のときのみ考慮されます。

0：ペリフェラルは無効になっているとき、GPIO の制御をしません。

1：ペリフェラルは常に、関連するすべての GPIO を制御します。

特定の設定の理由（たとえば、CRC リセット、CPHA または HDDIR 変更）のために SPI マスタを一時的に無効にする必要があるときに、このビットをセットすると、オルタネート機能モードで設定された関連出力を、現在の SPI 設定に対応する状態に強制することによって、そのグリッチを防止できます。スレーブトランスミッタは、SPI が無効になると、その MISO 出力を強制してはけないので、このビットはスレーブモードでは使用してはいけません。

注： このビットは、PCM および I2S モードでも使用できます。

ビット 30 **SSOM**：マスタモードでの SS 出力管理

このビットは SSOE が有効なときにマスタモードで使用されます。これによって、連続する 2 つのデータ転送の間に SS 出力を設定することが可能になります。

0：データ転送が完了するまで、SS はアクティブレベルに保持され、EOT フラグによって非アクティブになります。

1：MIDI[3:0]>1 のとき、SPI データフレームは SS の非アクティブパルスによってインターリーブされます。

ビット 29 **SSOE**：SS 出力有効

このビットは、マスタモードでのみ考慮されます。

0：SS 出力は無効にされ、SPI はマルチマスタ設定で機能できます。

1：SS 出力は有効です。SPI はマルチマスタ環境では機能できません。SSOM、MIDI、MSSI、SSIOP ビットの設定に従って転送した後、SS ピンを非アクティブに強制します。

ビット 28 **SSIOP**：SS 入出力の極性

0：SS 信号に関してローレベルがアクティブです。

1：SS 信号に関してハイレベルがアクティブです。

ビット 27 予約済み

ビット 26 **SSM**：SS 信号入力のソフトウェア管理

0：SS 入力値は SS PAD によって決定されます。

1：SS 入力値は SSI ビットによって決定されます。

マスタモードで SS 出力モードが有効になっている（SSOE=1）とき、SS 信号入力はソフトウェアによって（SSM=1、SSI=1）管理する必要があります。

ビット 25 **CPOL**：クロック極性

0：アイドル時に SCK 信号は 0 となります。

1：アイドル時に SCK 信号は 1 となります。

ビット 24 **CPHA**：クロック位相
0：最初のクロック遷移が最初のデータキャプチャエッジです。
1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

ビット 23 **LSBFRST**：データフレームフォーマット
0：MSB が最初に送信されます。
1：LSB が最初に送信されます。
注： このビットは、PCM および I2S モードでも使用できます。

ビット 22 **MASTER**：SPI マスタ
0：SPI スレーブ
1：SPI マスタ

ビット 21:19 **SP[2:0]**：シリアルプロトコル
000：SPI モトローラ
001：SPI TI
上記以外：予約済み。使用できません。

ビット 18:17 **COMM**：SPI 通信モード
00：全二重
01：単方向トランスミッタ
10：単方向レシーバ
11：半二重

ビット 16 予約済み

ビット 15 **IOSWP**：MISO および MOSI ピンのスワップ機能
0：スワップなし。
1：MOSI および MISO はスワップされます。
このビットがセットされた場合、MISO および MOSI ピンのオルタネートの機能は逆になります。
元の MISO ピンが MOSI になり、元の MOSI ピンが MISO になります。
このビットは、PCM および I2S モードでも使用できることに注意してください。

ビット 14:8 予約済み

ビット 7:4 **MIDI [3:0]**：マスタのデータ間アイドル状態
マスタモードで 2 つの連続するデータフレーム間に挿入される最小時間遅延（SPI クロックサイクルの周期で表される）を指定します。
0000：遅延なし
0001：1 クロックサイクル周期の遅延
...
1111：15 クロックサイクル周期の遅延
注： このビットは、TI モードではサポートされていません。

ビット 3:0 **MSSI [3:0]**：マスタの SS アイドル状態
SSOE が有効なとき、マスタモードで、SS のアクティブエッジと最初のデータのトランザクション開始との間に追加で挿入される余分な遅延（SPI クロックサイクル周期の数で表される）を指定します。
0000：余分な遅延なし
0001：1 クロックサイクル周期の遅延を付加
...
1111：15 クロックサイクル周期の遅延を付加
注： このビットは、TI モードではサポートされていません。

50.11.5 SPI/I2S 割り込み有効レジスタ（SPI2S_IER）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TSERFIE	MODFIE	TIFREIE	CRCEIE	OVRIE	UDRIE	TXTFIE	EOTIE	DPXPIE	TXPIE	RXPIE
					rw	rw	rw	rw	rw	rw	rw	rw	rs	rs	rw

ビット 31:11 予約済み

ビット 10 **TSERFIE**：トランザクションの追加数の再ロード割り込み有効

0：TSERF 割り込みは無効です。

1：TSERF 割り込みは有効です。

ビット 9 **MODFIE**：モードフォールト割り込み有効

0：MODF 割り込みは無効です。

1：MODF 割り込みは有効です。

ビット 8 **TIFREIE**：TIFRE 割り込み有効

0：TIFRE 割り込みは無効です。

1：TIFRE 割り込みは有効です。

ビット 7 **CRCEIE**：CRC エラー割り込みイネーブル

0：CRCエラー 割り込みは無効です。

1：CRC エラー割り込みは有効です。

ビット 6 **OVRIE**：OVR 割り込み有効

0：OVR 割り込みは無効です。

1：OVR 割り込みは有効です。

ビット 5 **UDRIE**：UDR 割り込み有効

0：UDR 割り込みは無効です。

1：UDR 割り込みは有効です。

ビット 4 **TXTFIE**：TXTF 割り込み有効

0：TXTF 割り込みは無効です。

1：TXTF 割り込みは有効です。

ビット 3 **EOTIE**：EOT、SUSP、および TXC 割り込み有効

0：EOT/SUSP/TXC 割り込みは無効です。

1：EOT/SUSP/TXC 割り込みは有効です。

ビット 2 **DXPIE**：DXP 割り込み有効
DXPIE は、ソフトウェアによってセットされ、TXTF フラグのセットイベントによってクリアされます。
0：DXP 割り込みは無効です。
1：DXP 割り込みは有効です。

ビット 1 **TXPIE：**TXP 割り込み有効
TXPIE は、ソフトウェアによってセットされ、TXTF フラグのセットイベントによってクリアされます。
0：TXP 割り込みは無効です。
1：TXP 割り込みは有効です。

ビット 0 **RXPIE：**RXP 割り込み有効
0：RXP 割り込みは無効です。
1：RXP 割り込みは有効です。

50.11.6 SPI/I2S ステータスレジスタ（SPI2S_SR）

アドレスオフセット：0x14

リセット値：0x0000 1002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTSIZE[15:0]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXWNE	RXPLVL[1:0]		TXC	SUSP	TSERF	MODF	TIFRE	CRCE	OVR	UDR	TXTF	EOT	DPXP	TXP	RXP
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **CTSIZE[15:0]**：現在の TSIZE セッションで残っているデータフレームの数
この値は、トラフィックがバスで進行中は完全には信頼できません。

ビット 15 **RXWNE：**RxFIFO ワードノットエンプティ
RxFIFO には 1 つ以上の 32 ビットデータが入っています。
0：RxFIFO では 32 ビットのデータフレーム未満のデータが利用可能です。
1：RxFIFO では少なくとも 1 つの 32 ビットのデータが利用可能です。

ビット 14:13 **RXPLVL[1:0]：**RxFIFO パッキングレベル
RxFIFO の最後の 32 ビットワード内の利用可能なパックされたフレームの数。このフレーム数は RXWNE=0 のとき、RxFIFO から読み出されます。
フレームサイズが 8 ビット以下の場合は、
00：RxFIFO 内のフレーム数は 0 か、4 の倍数です。
01：RXWNE=0 のとき、1 つのフレームが利用可能です。
10：RXWNE=0 のとき、2 つのフレームが利用可能です。
11：RXWNE=0 のとき、3 つのフレームが利用可能です。
フレームサイズが 16 ビット以下で、8 ビットより大きい場合は、RXPLVL[1:0] は 0 か 1 の値を取り得ます。
00：RxFIFO 内のフレーム数は 0 か、2 の倍数です。
01：RxFIFO 内のフレーム数は 奇数です。RXWNE=0 のときは、1 つの 16 ビットのデータが利用可能です。
その他の値は禁止されています。
フレームサイズが 16 ビットより大きい場合は、これらのビットは 00 となります。
00：-
その他の値は禁止されています。

ビット 12 **TXC**：TxFIFO 送信完了

このフラグは、ハードウェアによって変更されます。

TSIZE=0 のとき、TxFIFO が空になってバス上での動作がなくなるたびに TXC がセットされます。

TSIZE <> 0 のとき、TxFIFO の占有率がどうであれ、転送の終わりに TXC がセットされます。

このフラグがセットされると、マスタ送信は完了します。このフラグは、マスタの受信の終わりを考慮するのには信頼できません。CRC モードが有効な場合は、TXC は CRC の送信の後でのみセットされます。EOTIE がセットされているとき、TXC は割り込みを生成します。

0：現在のデータのトランザクションはまだ進行中です。データは TxFIFO で利用可能になっているか、または最後のフレームの送信（CRC も含めて）が進行中です。

1：最後の TxFIFO または CRC フレームの送信が完了しました。

ビット 11 **SUSP**：サスペンド状態

マスタモードにおいて、CSUSP リクエストが出され、現在のフレームが完了した時、またはマスタ自動サスペンド受信モード（SPI2S_CR1 レジスタの MASRX ビットがセットされている）で RxFIFO がフル状態になった時、直ちに SUSP がハードウェアによってセットされます。

EOTIE がセットされているとき、SUSP は割り込みを生成します。

このビットは、SPI2S_IFCR の SUSPC ビットに 1 を書き込むことによってクリアされます。

0：SPI はサスペンドされていません（マスタモードまたは他のモードがアクティブです）。

1：マスタモードはサスペンドされています（現在のフレームは完了しています）。

ビット 10 **TSERF**：トランザクションされる SPI データの追加の数が再ロードされました。

このビットは、SPI2S_IFCR の TSERFC ビットに 1 を書き込むことによって、または TSER[15:0]（SPI_CR2）レジスタに書き込むことによって、クリアされます。

0：受け付けなし。

1：追加のデータ数を受け付けました。現在のトランザクションは続行します。

ビット 9 **MODF**：モードフォールト

0：モードフォールトはありません。

1：モードフォールトを検出しました。

このビットは、SPI2S_IFCR の MODFC ビットに 1 を書き込むことによってクリアされます。

ビット 8 **TIFRE**：TI フレームフォーマットエラー

0：TI フレームエラーはありません。

1：TI フレームエラーを検出しました。

このビットは、SPI2S_IFCR の TIFREC ビットに 1 を書き込むことによってクリアされます。

ビット 7 **CRCE**：CRC エラー

0：CRC エラーはありません。

1：CRC エラーが検出されました。

このビットは、SPI2S_IFCR の CRCEC ビットに 1 を書き込むことによってクリアされます。

ビット 6 **OVR**：オーバーラン

0：オーバーランは起きていません。

1：オーバーランが検出されました。

このビットは、SPI2S_IFCR の OVRC ビットに 1 を書き込むことによってクリアされます。

ビット 5 **UDR**：スレーブ送信モードでアンダーラン

0：アンダーランは起きていません。

1：アンダーランが検出されました。

このビットは、SPI2S_IFCR の UDRC ビットに 1 を書き込むことによってクリアされます。

注：　UDR フラグはスレーブモードのみに適用されます。

ビット 4 **TXTF**：送信の転送フル

0：TxFIFO のアップロードは処理中または開始されていません。

1：TxFIFO のアップロードは終了しています。

アプリケーションソフトウェアまたは DMA によって転送内のすべてのデータパケットが送信用に提出されると、つまり TSIZE 数のデータが TxFIFO にプッシュされた時、すぐにハードウェアによって TXTF がセットされます。

このビットは、SPI2S_IFCR の TXTFC ビットにソフトウェアで 1 を書き込むことによってクリアされます。

TXTFIE ビットがセットされている場合、TXTF フラグは割り込みをトリガします。

TXTF をセットすると TXIE および DPXIE マスクがクリアされるので、アプリケーションソフトウェアは TXP および DXP 割り込みをいつ無効にするかの計算の負担を免れます。

ビット 3 **EOT**：転送終了

全転送が完了すると、つまり SPI で TSIZE 数のデータが送信および／または受信された時、すぐにハードウェアによって EOT がセットされます。EOT は、SPI2S_IFCR の EOTC ビットにソフトウェアで 1 を書き込むことによってクリアされます。

EOTIE ビットがセットされている場合、EOT フラグは割り込みをトリガします。

TXTF フラグがセットされるまで DXP フラグが使用され、DXPIE がクリアされている場合、RxFIFO に入っている最後のパケットを 1 回でダウンロードするために EOT を使用することができます。

0：転送は処理中または開始されていません。

1：転送完了

マスタでは、EOT イベントによって、データのトランザクションが終了し、オプションで SS 出力が処理されます。マスタとスレーブの両方で、このイベントにより CRC トランザクションが処理されます。

ビット 2 **DXP**：二重パケット

0：TxFIFO はフルで、および／または RxFIFO は空です。

1：TxFIFO は書き込むスペースがあり、RxFIFO は少なくとも 1 つの読み出すパケットを含んでいます。

DXP フラグは、TXP と RXP の両フラグがセットされた時、セットされます。

ビット 1 **TXP**：送信パケットスペース利用可能

0：次のデータパケットを TxFIFO に置くのに十分な領域がありません。

1：TxFIFO には 1 データパケットを入れるのに十分な空き領域があります。

TXP フラグは、ハードウェアによって変更されます。同フラグは、SPI が有効な場合、TxFIFO で現在使用可能な全体スペースを監視します。1 つの完全なデータパケットを TxFIFO に格納した後、これをチェックする必要があります。

ビット 0 **RXP**：受信パケット使用可能

0：RxFIFO は空であるか、または不完全なデータパケットが受信されています。

1：RxFIFO には 少なくとも 1 つのデータパケットが入っています。

RXP フラグは、ハードウェアによって変更されます。同フラグは、SPI が有効な場合、RxFIFO で現在使用可能な全体データ数を監視します。1 つのデータパケットが RxFIFO から完全に読み出された後、これをチェックする必要があります。

50.11.7 SPI/I2S 割り込み／ステータスフラグクリアレジスタ（SPI2S_IFCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SUSPC	TSERFC	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
				w	w	w	w	w	w	w	w	w			

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SUSPC**：サスペンドフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの SUSP フラグがクリアされます。

ビット 10 **TSERFC**：TSERFC フラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TSERF フラグがクリアされます。

注：*TSERF は、TSER[15:0]（SPI_CR2）レジスタに書き込むことによっても、リセットされます。*

ビット 9 **MODFC**：モードフォールトフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの MODF フラグがクリアされます。

ビット 8 **TIFREC**：TI フレームフォーマットエラーフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TIFRE フラグがクリアされます。

ビット 7 **CRCEC**：CRC エラーフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの CRCE フラグがクリアされます。

ビット 6 **OVRC**：オーバーランフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの OVR フラグがクリアされます。

ビット 5 **UDRC**：アンダーランフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの UDR フラグがクリアされます。

ビット 4 **TXTFC**：送信転送フルフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TXTF フラグがクリアされます。

ビット 3 **EOTC**：転送の終了フラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの EOT フラグがクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

50.11.8 SPI/I2S 送信データレジスタ（SPI2S_TXDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXDR[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXDR[15:0]															
w															

ビット 31:0 **TXDR[31:0]**：送信データレジスタ

このデータレジスタは、TxFIFO とのインタフェースとして使用できます。これに書き込むと TxFIFO へアクセスします。

注： ***データは常に右詰めです。未使用のビットは、レジスタへの書き込み時に無視され、レジスタの読み出し時にゼロとして読み出されます。***

注： ***DR はバイト単位に（8 ビットアクセスで）アクセス可能です。この場合、1 回のアクセスで 1 データバイトのみが書き込まれます。***

ハーフワード単位（16 ビットアクセス）の場合、1 回のアクセスで 2 データバイトまたは 1 ハーフワードデータが書き込めます。

ワード単位（32 ビットアクセス）の場合、1 回のアクセスで 4 データバイト、または 2 ハーフワードデータ、またはワードデータが書き込めます。

50.11.9 SPI/I2S 受信データレジスタ（SPI2S_RXDR）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXDR[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDR[15:0]															
r															

ビット 31:0 **RXDR[31:0]:** 受信データレジスタ

このレジスタは、RxFIFO とのインタフェースとして使用できます。これが読み出されるとき、RxFIFO がアクセスされます。

注： ***データは常に右詰めです。未使用のビットは、レジスタの読み出し時にはゼロとして読み出されます。このレジスタへの書き込みは無視されます。***

注： ***DR はバイト単位（8 ビットアクセス）でアクセス可能です。この場合、1 回のアクセスで 1 データバイトのみが読み出されます。***

ハーフワード単位（16 ビットアクセス）の場合、1 回のアクセスで 2 データバイトまたは 1 ハーフワードデータが読み出されます。

ワード単位（32 ビットアクセス）の場合、1 回のアクセスで 4 データバイト、または 2 ハーフワードデータ、またはワードデータが読み出されます。

50.11.10 SPI 多項式レジスタ（SPI_CRCPOLY）

アドレスオフセット：0x40

リセット値：0x0000 0107

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRCPOLY[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRCPOLY[15:0]															
rw															

ビット 31:0 **CRCPOLY[31:0]**：CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を格納します。

デフォルトの 9 ビット多項式の設定 0x107 は DSIZE がデフォルトの 8 ビット設定に対応しています。これは、ストリングの最上位ビットが常に隠されている固定長の多項式ストリングを備えた他のいくつかの ST 製品で使用されている設定 0x07 と互換性があります。

多項式の長さはこのレジスタに格納されている値の最上位ビットによって与えられます。その長さは DSIZE より大きく設定する必要があります。DSIZE が最大の 32 ビットまたは 16 ビットのサイズに設定され、CRC が有効になっているとき、多項式の長さをデータサイズより大きく保つために、CRCPOLY レジスタとともにさらに CRC33_17 ビットをセットする必要があります。

ビット 31～16 は、データサイズが 16 ビットに限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 ビットアクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。

50.11.11 SPI トランスミッタ CRC レジスタ（SPI_TXCRC）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXCRC[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXCRC[15:0]															
rw															

ビット 31:0 **TXCRC[31:0]**：トランスミッタ用 CRC レジスタ

CRC 計算が有効なとき、TXCRC[31:0] ビットには、その後に送信されたバイトから算出された CRC 値が格納されます。CRC 計算は、SPI_CR1 の CRCEN ビットに 1 が書き込まれた時、またはデータブロックが完全にトランザクションされた時に初期化されます。CRC は、SPI_CRCPOLY レジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLY レジスタと、SPI_CFG1 レジスタの CRCSIZE ビットの設定に依存します。

注： ***通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。***

I^2S モードでは使用しません。

ビット 31～16 は、データサイズが 16 ビットに限定されているペリフェラル実体では予約済みです。これらのアドレスに 32 ビットアクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常に読み値ゼロで、それらへの書込みは無視されます。

50.11.12 SPI レシーバ CRC レジスタ（SPI_RXCRC）

アドレスオフセット：0x48

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXCRC[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCRC[15:0]															
rw															

ビット 31:0 **RXCRC[31:0]**：レシーバ用 CRC レジスタ

CRC 計算が有効なとき、RXCRC[31:0] ビットには、その後に受信したバイトから算出された CRC 値が格納されています。CRC 計算は、SPI_CR1 の CRCEN ビットに 1 が書き込まれた時、またはデータブロックが完全にトランザクションされた時に初期化されます。CRC は、SPI_CRCPOLY レジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLY レジスタと、SPI_CFG1 レジスタの CRCSIZE ビットの設定に依存します。

注： ***通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。***

I^2S モードでは使用しません。

ビット 31～16 は、データサイズが 16 ビットに限定されているペリフェラル実体では予約済みです。これらのアドレスに 32 ビットアクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常に読み値ゼロで、それらへの書込みは無視されます。

50.11.13 SPI アンダーランデータレジスタ（SPI_UDRDR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UDRDR[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UDRDR[15:0]															
rw															

ビット 31:0 **UDRDR[31:0]**：スレーブのアンダーラン状態でのデータ

このレジスタは、スレーブモードで、アンダーラン状態でのみ考慮されます。考慮されるビット数は、SPI_CFG1 レジスタの DSIZE ビットの設定に依存します。アンダーラン状態の処理は SPI_CFG1 レジスタの UDRDET および UDRCFG ビットの設定に依存します。

ビット 31～16 は、データサイズが 16 ビットに限定されているペリフェラル実体では予約済みです。これらのアドレスに 32 ビットアクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常に読み値ゼロで、それらへの書込みは無視されます。

50.11.14 SPI/I2S 設定レジスタ（SPI_I2SCGFR）

アドレスオフセット：0x50

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MCKOE (1)	ODD (1)	I2SDIV[7:0] (1)							
						rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DATFMT (1)	WSINV (1)	FIXCH (1)	CKPOL (1)	CHLEN (1)	DATLEN[1:0] (1)		PCMSYNC (1)	Res.	I2SSTD[1:0] (1)		I2SCFG[2:0] (1)			I2SMOD (1)
	rw	rw	rw	rw	rw	rw		rw		rw		rw			rw

1. これらビットは、I2S が無効（SPE=0）のときに設定する必要があります。
SPI モードで 0 にセットする必要がある I2SMOD を除いて、これらのフィールドは SPI モードでは使用されません。

ビット 31:26 予約済み。ハードウェアによって 0 に固定されています。

ビット 25 **MCKOE**：マスタクロック出力有効

0：マスタクロック出力は無効です。

1：マスタクロック出力は有効です。

ビット 24 **ODD**：プリスケーラの奇数分周比

0：真の分周値 = I2SDIV *2

1：真の分周値 = （I2SDIV * 2）+ 1

詳細については、*セクション 50.9.9：クロックジェネレータ*を参照してください。

ビット 23:16 **I2SDIV**：I^2S リニアプリスケーラ

ODD が 1 のとき、I2SDIV は 1 の値を除いて任意の値をとることができます。

詳細については、*セクション 50.9.9：クロックジェネレータ*を参照してください。

ビット 15 予約済み。ハードウェアによって 0 に固定されています。

ビット 14 **DATFMT**：データフォーマット

0：SPI2S_RXDR または SPI2S_TXDR 内のデータは右詰めです。

1：SPI2S_RXDR または SPI2S_TXDR 内のデータは左詰めです。

ビット 13 **WSINV**：ワード選択の反転

このビットは、WS 信号のデフォルト極性を反転するために使用します。

0：フィリップス I2S 規格において、WS が LOW のとき、左チャネルが転送され、WS が HIGH のとき、右チャネルが転送されます。
MSB または LSB 詰めモードでは、WS が HIGH のとき、左チャネルが転送され、WS が LOW のとき、右チャネルが転送されます。
PCM モードでは、フレームの開始は立ち上がりエッジで指示されます。

1：フィリップス I2S 規格において、WS が HIGH のとき、左チャネルが転送され、WS が LOW のとき、右チャネルが転送されます。
MSB または LSB 詰めモードでは、WS が LOW のとき、左チャネルが転送され、WS が HIGH のとき、右チャネルが転送されます。
PCM モードでは、フレームの開始は立ち下がりエッジで指示されます。

ビット 12 **FIXCH**：スレーブでの固定チャネル長

0：スレーブモードでのチャネル長は 16 または 32 ビットと異なります（CHLEN は考慮されません）。

1：スレーブモードでのチャネル長は 16 または 32 ビットと推測されます（CHLEN による）。

ビット 11 **CKPOL**：シリアルオーディオクロックの極性

0 : SPI/I2S によって生成される信号（すなわち SDO および WS）は CK の立ち下がりエッジで変化し、SPI/I2S によって受信される信号（すなわち SDI および WS）は CK の立ち上がりエッジで読み出されます。

1 : SPI/I2S によって生成される信号（すなわち SDO および WS）は CK の立ち上がりエッジで変化し、SPI/I2S によって受信される信号（すなわち SDI および WS）は CK の立ち下がりエッジで読み出されます。

ビット 10 **CHLEN**：チャネル長（オーディオチャネルごとのビット数）

0：16 ビット幅

1：32 ビット幅

ビット書き込み動作は、DATLEN=00 のときにのみ意味を持ちます。そうでない場合、書き込まれた値とは無関係に、チャネル長はハードウェアによって 32 ビットに固定されます。

ビット 9:8 **DATLEN**：転送されるデータ長

00：16 ビットデータ長

01：24 ビットデータ長

10：32 ビットデータ長

11：設定禁止

ビット 7 **PCMSYNC**：PCM フレーム同期

0：ショートフレーム同期

1：ロングフレーム同期

ビット 6 予約済みであり、ハードウェアによって 0 に固定されています。

ビット 5:4 **I2SSTD**：I^2S 規格選択

00：フィリップス I^2S 規格

01：MSB 詰め規格（左詰め）

10：LSB 詰め規格（右詰め）

11：PCM 規格

I^2S 規格に関する詳細は、*セクション 50.9.5：サポートされるオーディオプロトコル* を参照してください。

ビット 3:1 **I2SCFG**：I2S 設定モード

000：スレーブ - 送信

001：スレーブ - 受信

010：マスタ - 送信

011：マスタ - 受信

100：スレーブ - 全二重

101：マスタ - 全二重

その他、未使用

ビット 0 **I2SMOD**：I2S モード選択

0：SPI モードが選択されます。

1：I2S/PCM モードが選択されます。

50.12 SPI レジスタマップとリセット値

表 395. SPI レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**SPI2S_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOLOCK	TCRCINI	RCRCINI	CRC33_17	SSI	HDDIR	CSUSP	CSTART	MASRX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
	リセット値																0	0	0	0	0	0	0	0	0								0
0x04	**SPI_CR2**	TSER[15:0]																TSIZE[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**SPI_CFG1**	Res.	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	CRCSIZE[4:0]					TXDMAEN	RXDMAEN	Res.	UDRDET [1:0]		UDRCFG [1:0]		FTHLV[3:0]				DSIZE[4:0]				
	リセット値		0	0	0						0		1	1	1	1	1	0	0		0	0	0	0	0	0	0	0	0	0	1	1	1
0x0C	**SPI_CFG2**	AFCNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSBFRST	MASTER	SP[2:0]			COMM [1:0]		Res.	IOSWP	Res.	Res.	Res.	Res.	Res.	Res.	MIDI[3:0]				MIDI[3:0]				
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0							0	0	0	0	0	0	0	0	0
0x10	**SPI2S_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSERFIE	MODFIE	TIFREIE	CRCIE	OVRIE	UDRIE	TXTFIE	EOTIE	DPXPIE	TXPIE	RXPIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x14	**SPI2S_SR**	CTSIZE[15:0]																RXWNE	RXPLVL [1:0]		TXC	SUSP	TSERF	MODF	TIFRE	CRCERR	OVR	UDR	TXTF	EOT	DPXP	TXP	RXP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
0x18	**SPI2S_IFCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSPC	TSERFC	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
	リセット値																					0	0	0	0	0	0	0	0	0			
0x20	**SPI2S_TXDR**	TXDR[31:16]																TXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24 - 0x2C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x30	**SPI2S_RXDR**	RXDR[31:16]																RXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 - 0x3C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x40	**SPI_CRCPOLY**	CRCPOLY[31:16](1)																CRCPOLY[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
0x44	**SPI_TXCRC**	TXCRC[31:16] (1)																TXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**SPI_RXCRC**	RXCRC[31:16] (1)																RXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	**SPI_UDRDR**	UDRDR[31:16] (1)																UDRDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**SPI_I2SCGFR**	Res.	Res.	Res.	Res.	Res.	Res.	MCKOE	ODD	I2SDIV[7:0]								Res.	Res.	WSINV	FIXCH	CKPOL	CHLEN	DATLEN		PCMSYNC	Res.	I2SSTD[1:0]		I2SCFG[2:0]		I2SMOD	
	リセット値							0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0		0	0	0	0	0	0

参考資料

1. ビット 31～16 は、データサイズが 16 ビットに限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 ビットアクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常に読み値ゼロで、それらへの書込みは無視されます。レジスタ境界アドレスの表を参照してください。

参考資料

51 シリアルオーディオインタフェース（SAI）

51.1 概要

この SAI インタフェース（シリアルオーディオインタフェース）は、柔軟性があり幅広い構成が可能なため、多様なオーディオプロトコルに対応できます。多くのステレオまたはモノラルオーディオアプリケーションが対象になります。例えば、I2S 規格、LSB/MSB 詰め、PCM/DSP、TDM、AC'97 の各プロトコルに対応することができます。オーディオブロックがトランスミッタとして設定された場合、SPDIF 出力が提供されます。

このレベルの柔軟性と再設定機能を実現するため、SAI は独立した 2 つのオーディオサブブロックを内蔵しています。各ブロックには専用のクロックジェネレータと I/O ラインコントローラが備わっています。

この SAI は、マスタまたはスレーブの設定で動作することができます。このオーディオサブブロックはレシーバにもトランスミッタにもなることができ、他の一方に対して同期または非同期で動作することができます。

SAI を他の SAI と接続することで、互いに同期して動作することができます。

51.2 SAI の主な機能

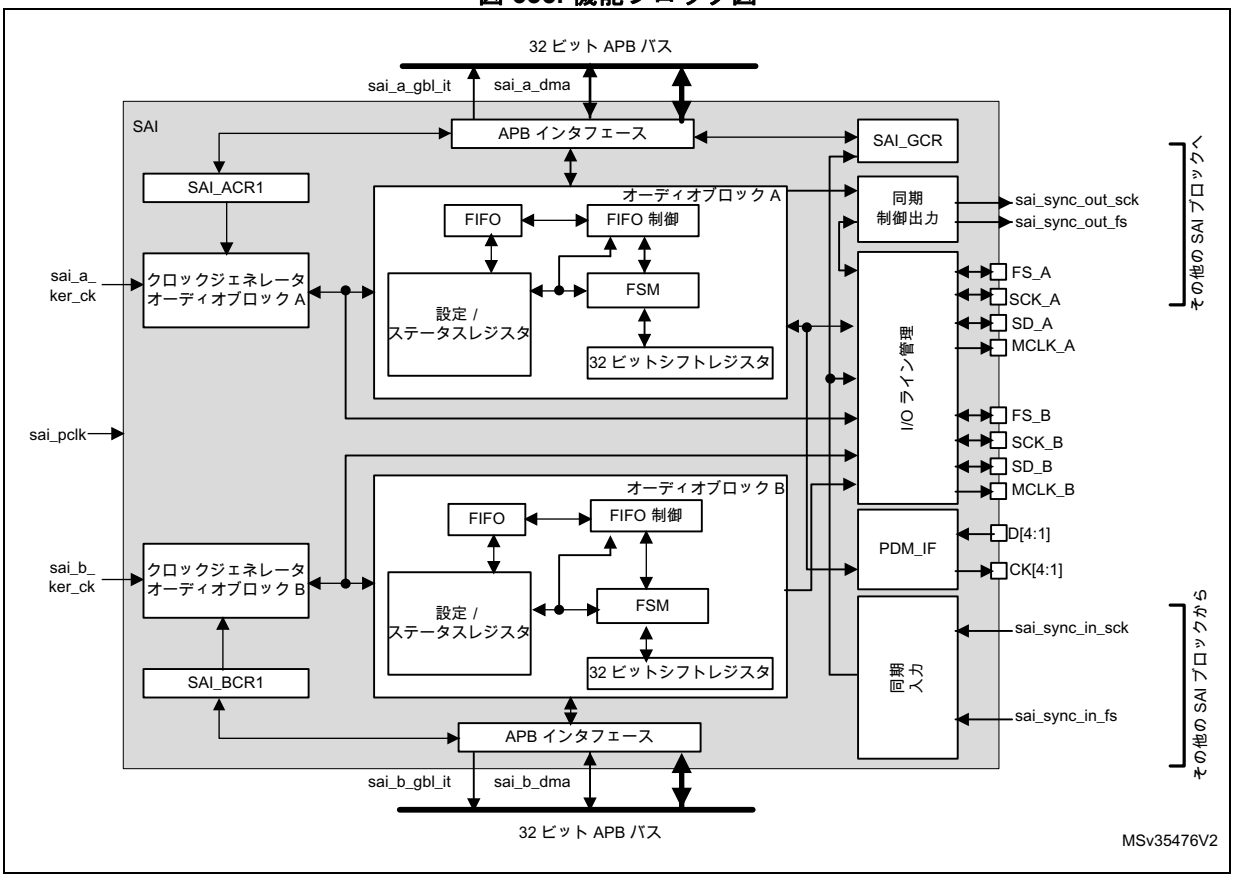
- 個別に FIFO を内蔵し、トランスミッタまたはレシーバとなることができる 2 つの独立したオーディオサブブロック。
- 各オーディオサブブロックに 8 ワード FIFO を内蔵。
- オーディオサブブロック間での同期モードまたは非同期モードが可能。
- 複数の SAI 間の同期が可能。
- 両オーディオサブブロックに対して独立してマスタ設定またはスレーブ設定が可能。
- 両オーディオサブブロックをマスタモードに設定した場合に、個別のオーディオサンプリング周波数に対応する各オーディオブロック用クロックジェネレータ。
- 設定可能なデータサイズ：8、10、16、20、24、32 ビット
- オーディオプロトコル：I2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97
- 最大 8 台のマイクロホンをサポートする PDM インタフェース。
- 必要に応じて、SPDIF 出力を利用可能。
- サイズ設定が可能な最大 16 個のスロットが利用可能。
- フレームごとのビット数が設定可能。
- フレーム同期アクティブレベルが設定可能（オフセット、ビット長、レベル）。
- スロット内の先頭アクティブビット位置が設定可能。
- データ転送における LSB ファーストまたは MSB ファーストが可能。
- ミュートモード。
- ステレオ／モノラルオーディオフレーム機能。
- 通信クロックストローブエッジが設定可能（SCK）。
- 対応する割り込み付きのエラーフラグ（それぞれイネーブル付き）
  - オーバーラン検出およびアンダーラン検出、
  - スレーブモードにおける予測フレーム同期信号検出、
  - スレーブモードにおけるレイトフレーム同期信号検出、
  - 受信時 AC'97 モードに対するコーデックノットレディ
- 割り込みソース（有効化時）：
  - エラー
  - FIFO リクエスト
- 2 チャネル DMA インタフェース

51.3 SAI の機能詳細

51.3.1 SAI ブロック図

図 635 に SAI ブロック図を、*表 396* および *表 397* に SAI 内部および外部信号のリストを示します。

図 635. 機能ブロック図



この SAI は大部分が 2 つのオーディオサブブロックから構成されており、それぞれに固有のクロックジェネレータを装備しています。各オーディオブロックは、それぞれの機能ステートマシンによって制御される 32 ビットシフトレジスタを内蔵しています。データは、専用 FIFO に保存されたり、そこから読み出されたりします。FIFO は CPU からアクセスすることもできれば、通信時に CPU を開放するため DMA からアクセスすることもできます。各オーディオブロックは独立しています。互いに同期することもできます。

I/O ラインコントローラは、SAI 内の指定オーディオブロックに対する 4 本の専用ピン（SD、SCK、FS、MCLK）を管理します。この 2 つのサブブロックを同期するように指定すると、いくつかのピンを共用していくつかのピンを開放し、汎用 I/O として使用することができます。アプリケーション、デコーダ要件、オーディオブロックがマスタとして設定されているか否かによって、MCLK ピンを出力にしたり、それ以外にしたりすることができます。

1 つの SAI が別の SAI と同期して動作するように設定されている場合、より多くの I/O を開放することができます（SD_x ピンを除く）。

この機能ステートマシンは、広範囲なオーディオプロトコルを処理するように設定することができます。所望のプロトコルを設定するため、いくつかのレジスタがあります（オーディオフレーム波形ジェネレータ）。

オーディオサブブロックは、マスタモードまたはスレーブモードでトランスミッタまたはレシーバとなり得ます。マスタモードでは、SCK_x ビットクロックとフレーム同期信号が SAI で生成されますが、スレーブモードでは、これらの信号を別の外部マスタまたは内部マスタから受け取ります。FS 信号の方向がマスタモードまたはスレーブモードの定義に直接結びつかない特別なケースがあります。AC'97 プロトコルでは、SAI （リンクコントローラ）が SCK クロックを入力するように設定されている場合でも （つまりスレーブモードでも）、FS 信号は SAI の出力になります。

注： 読みやすくするために、このセクションでは SAI_A または SAI_B を SAI_x と表記しています。「x」とは SAI A サブブロックまたは SAI B サブブロックのことです。

51.3.2 SAI ピンおよび内部信号

表 396. SAI 内部信号

内部信号名	信号タイプ	説明
sai_a_gbl_it/ sai_b_gbl_it	デジタル出力	オーディオブロック A および B グローバル割り込み
sai_a_dma、sai_b_dma	デジタル入出力	オーディオブロック A および B DMA 確認応答およびリクエスト
sai_sync_out_sck、sai_sync_out_fs	デジタル出力	他の SAI ブロックと交換される内部クロックおよびフレーム同期出力信号
sai_sync_in_sck、sai_sync_in_fs	デジタル入力	他の SAI ブロックと交換される内部クロックおよびフレーム同期入力信号
sai_a_ker_ck/ sai_b_ker_ck	デジタル入力	オーディオブロック A/B カーネルクロック
sai_pclk	デジタル入力	APB クロック

表 397. SAI ピン

名前	信号タイプ	コメント
SAI_SCK_A/B	デジタル入出力	オーディオブロック A/B ビットクロック
SAI_MCLK_A/B	デジタル出力	オーディオブロック A/B マスタクロック
SAI_SD_A/B	デジタル入出力	ブロック A/B のデータライン
SAI_FS_A/B	デジタル入出力	オーディオブロック A/B のフレーム同期ライン
SAI_CK[4:1]	デジタル出力	PDM ビットストリームクロック
SAI_D[4:1]	デジタル入力	PDM ビットストリームデータ

51.3.3 主要な SAI モード

SAI の各オーディオサブブロックは、選択したオーディオブロックの SAI_xCR1 レジスタの MODE ビットを使ってマスタまたはスレーブに設定することができます。

マスタモード

マスタモードでは、SAI は接続された外部デバイスへタイミング信号を送り出します。

- ビットクロックとフレーム同期はそれぞれ SCK_x ピンと FS_x ピンに出力されます。
- 必要であれば、SAI は MCLK_x ピンにマスタクロックを生成することもできます。

SCK_x、FS_x、および MCLK_x のいずれも出力として設定されます。

スレーブモード

SAI は、外部デバイスからのタイミング信号の受信を待ちます。

- SAI サブブロックが非同期モードに設定されている場合、SCK_x ピンおよび FS_x ピンは入力として設定されます。
- SAI サブブロックが 他の SAI インタフェースまたは 2 番目のオーディオサブブロックと同期して動作するよう設定されている場合、対応する SCK_x および FS_x ピンは汎用 I/O として使用できるよう空けておきます。

MCLK_x ピンはスレーブモードでは使用されず、別の機能に割り当てることができます。

スレーブデバイスを有効にした後でマスタを有効にすることをお勧めします。

SAI モードの設定と有効化

各オーディオサブブロックは、該当するオーディオブロックにある SAI_xCR1 レジスタの MODE ビットを使って個別にトランスミッタまたはレシーバに指定することができます。そのため、SAI_SD_x ピンは出力または入力としてそれぞれ設定されます。

同じ SAI にある 2 個のマスタオーディオブロックは、2 つの異なる MCLK および SCK クロック周波数で設定することができます。この場合、オーディオブロックは非同期モードに設定する必要があります。

SAI 内の各オーディオブロックは、SAI_xCR1 レジスタの SAIXEN ビットにより有効化されます。このビットがアクティブになると直ちに、トランスミッタまたはレシーバはスレーブモードでクロックライン、データライン、同期ラインの動作に反応するようになります。

マスタ TX モードでは、オーディオブロックを有効化すると直ちに、FIFO にデータがなくとも、外部スレーブに対するビットクロックを生成しますが、FS 信号の生成は、FIFO 内のデータの存在が条件となります。FIFO が先頭の送信データを受け取った後、このデータが外部スレーブに出力されます。FIFO に送信データがない場合、オーディオフレームにおいては 0 値が送信されて、アンダーランフラグが生成されます。

スレーブモードでは、オーディオブロックが有効化されて、フレームの開始が検出されたときに、オーディオフレームが開始されます。

スレーブ TX モードでは、オーディオブロックを有効化した後の最初のフレームでアンダーランイベントが生成されることはありません。理由は、次の動作シーケンスが必要となるためです。

1. ソフトウェアまたは DMA により SAI_xDR へ書き込みを行います。
2. FIFO 閾値 (FLH）フラグが 000b（FIFO エンプティ）でなくなるまで待ちます。
3. スレーブ送信ミッタモードでオーディオブロックを有効化します。

51.3.4 SAI 同期モード

オーディオサブブロックレベルと SAI レベルの 2 種類の同期レベルがあります。

内部同期

オーディオサブブロックは、同じ SAI 内の 2 番目のオーディオサブブロックと同期して動作できるよう設定できます。この場合、ビットクロック信号とフレーム同期信号が共用され、通信に使用する外部ピン数を少なくすることができます。同期モードに設定されたオーディオブロックでは、その SCK_x、FS_x、および MCLK_x ピンは GPIO として開放状態に戻されています。非同期モードに設定されたオーディオブロックのほうが、FS_x、SCK_x、MCLK_x の I/O ピンに適したブロックとなります（オーディオブロックをマスタと考えた場合）。

一般に、同期モードのオーディオブロックを使って、SAI を全二重モードに設定することができます。2 個のオーディオブロックの片方をマスタとして設定し、他方をスレーブとして設定することができます。あるいは、両方をスレーブとして設定し、1 つは非同期ブロック(SAI_xCR1 の対応する SYNCEN[1:0] ビットを 00 にセット）に、もう一方を同期ブロック（SAI_xCR1 の対応する SYNCEN[1:0] ビットを 01 にセット）にすることもできます。

*注： **内部再同期ステージのため、PCLK APB 周波数はビットレートクロック周波数の 2 倍以上である必要があります。***

外部同期

オーディオサブブロックは他の SAI と同期して動作するよう設定することもできます。この方法は以下の通りです。

1. SAI が、他の SAI の同期元として設定されている場合は、その相手の SAI に FS および SCK 信号を提供するオーディオサブブロックを定義する必要があります。これは SYNCOUT[1:0] ビットを設定することによって行います。
2. 同期信号を受信する側の SAI は、SYNCIN[1:0] ビットに適正な値をセットすることによって同期を提供してくれる SAI を選択する必要があります。2 個の SAI オーディオサブブロックのそれぞれに対して、ユーザはそのサブブロックが他方の SAI と同期して動作するかどうかを SYNCEN ビットで指定します。

*注： **SYNCIN[1:0] および SYNCOUT[1:0] ビットは SAI_GCR レジスタ内にあり、SYNCEN ビットは SAI_xCR1 レジスタ内にあります。***

SAI の 2 個のオーディオサブブロックがどちらも他の SAI と同期して動作する必要がある場合は、次の設定のいずれかを選択することができます。

- 他の SAI ブロックと同期させる各オーディオブロックを SYNCEN[1:0] ビットにより設定します。
- 他の SAI と同期させる 1 個のオーディオブロックを SYNCEN[1:0] ビットにより設定します。このとき、もう一方のオーディオブロックは 2 番目の SAI オーディオブロックと同期させるように、SYNCEN[1:0] ビットによって設定されます。

下表は、使用される SAI ブロックに応じた適切な同期信号の選び方を示します。たとえば、SAI2 は SAI2 SYNCIN を 0 に設定することで、SAI1 からの同期を選択できます。SAI1 に SAI2 からの同期を選択させたい場合は、SAI1 SYNCIN を 1 にセットする必要があります。「res」と記載されている位置は使用されていません。

表 398. 外部同期の選択

ブロックの例	SYNCIN= 3	SYNCIN= 2	SYNCIN= 1	SYNCIN= 0
SAI1	SAI4 同期	SAI3 同期	SAI2 同期	Res.
SAI2	SAI4 同期	SAI3 同期	Res.	SAI1 同期
SAI3	SAI4 同期	Res.	SAI2 同期	SAI1 同期
SAI4	Res.	SAI3 同期	SAI2 同期	SAI1 同期

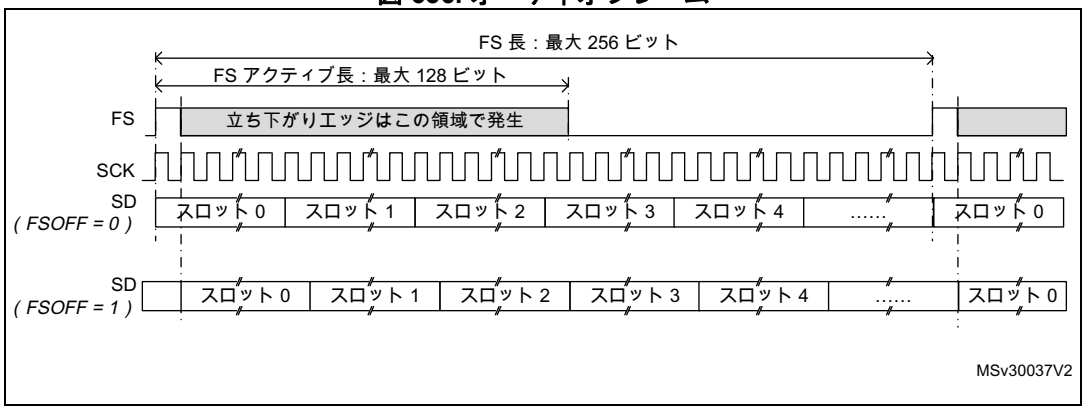
51.3.5 オーディオデータサイズ

オーディオフレームは、SAI_xCR1 レジスタの DS[2:0] ビットを設定することにより、様々なデータサイズを対象とすることができます。データサイズとしては、8、10、16、20、24、32 ビットが可能です。転送時、SAI_xCR1 レジスタの LSBFIRST ビットの設定によってデータの MSB を先頭にして送信することもできれば LSB を先頭にすることもできます。

51.3.6 フレーム同期

FS 信号は、オーディオフレームにおいてフレーム同期信号として機能します（フレーム開始）。この信号の波形は、このフレーム同期動作に関する独自の仕様を持つ様々なオーディオプロトコルに対応するため、全面的に設定可能となっています。この再設定は SAI_xFRCR レジスタを使って実行されます。*図 636* にこの柔軟性について図示します。

図 636. オーディオフレーム



AC'97 モード または SPDIF モード （SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01 ）では、フレーム同期波形を 強制的にAC'97 プロトコルに適合させます。SAI_xFRCR レジスタ値は無視されます。

各オーディオブロックは独立であるため、それぞれに固有の設定が必要となります。

フレーム長

- マスタモード

  オーディオフレーム長は、SAI_xFRCR レジスタの FRL[7:0] ビットを設定して、最大 256 ビットクロックに設定することができます。

  フレーム長がフレームに対して指定されたスロット数より大きい場合、SAI_xCR2 レジスタの TRIS ビットの状態に応じて、残りの送信ビットが 0 に拡張されるか、または SD ラインがハイインピーダンス（HI-Z）状態になります （*セクション ：FS 信号の機能*を参照）。受信モードでは、残りのビットは無視されます。

  NOMCK ビットがクリアされている場合、(FRL + 1) は8 から 256 のうちの 2 の累乗の数に調整する必要があります。これは、オーディオフレームがビットクロックサイクル当たり整数個の MCLK パルスを含むようにするためです。

  NOMCK ビットがセットされている場合、(FRL + 1) フィールドは 8 から 256 のうちの任意の値をとることができます。*セクション 51.3.8 ：SAI クロックジェネレータ*を参照してください。

- スレーブモード

  オーディオフレーム長は主に、外部マスタから送信されるオーディオフレーム当たりのビットクロック数をスレーブに対して指定するために使用されます。また、処理中のオーディオフレームでマスタからの想定されるフレーム同期信号または遅延したフレーム同期信号の発生を検出するためにも使用されます。この場合、エラーが生成されます。詳細については、*セクション 51.3.14 ：エラーフラグ*を参照してください。

  スレーブモードでは、SAI_xFRCR レジスタの FRL[7:0] ビット設定に制約はありません。

フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレーム内で転送する最小ビット数は 8 です。

フレーム同期極性

SAI_xFRCR レジスタの FSPOL ビットにより、フレームが開始される FS ピンのアクティブ極性が設定されます。フレーム開始は、エッジに反応します。

スレーブモードにおいて、オーディオブロックは有効なフレームを待って、送信または受信を開始します。フレーム開始はこの信号に同期します。これは、通信中のフレーム開始が検出されず、予測のフレーム開始と一致しない場合にのみ有効です （*セクション 51.3.14 : エラーフラグ*を参照）。

マスタモードでは、SAI_xCR1 レジスタの SAIXEN ビットがクリアされるまで、オーディオフレームが完了するごとにフレーム同期が毎回送信されます。前のオーディオフレームの終了時に FIFO にデータがない場合、アンダーラン状態は
*セクション 51.3.14 : エラーフラグ*の説明のように扱われますが、オーディオ通信フローが中断されることはありません。

フレーム同期アクティブレベル長

SAI_xFRCR レジスタの FSALL[6:0] ビットにより、フレーム同期信号のアクティブレベル長が設定されます。この長さは、1 から 128 のビットクロックに設定することができます。

例えば、I2S、LSB 詰めまたは MSB 詰めモードでは、アクティブ長をフレーム長の 1/2 に、PCM/DSP、TDM モードでは 1 ビット幅に、それぞれ設定することができます。

フレーム同期オフセット

アプリケーションで対象とするオーディオプロトコルに応じて、オーディオフレームの最終ビットまたは先頭ビットを送信したときフレーム同期信号をアサートすることができます（例えば、それぞれ I2S 規格プロトコルおよび MSB 詰めプロトコルの場合）。SAI_xFRCR レジスタの FSOFF ビットにより、2 つの設定のいずれかを選択することができます。

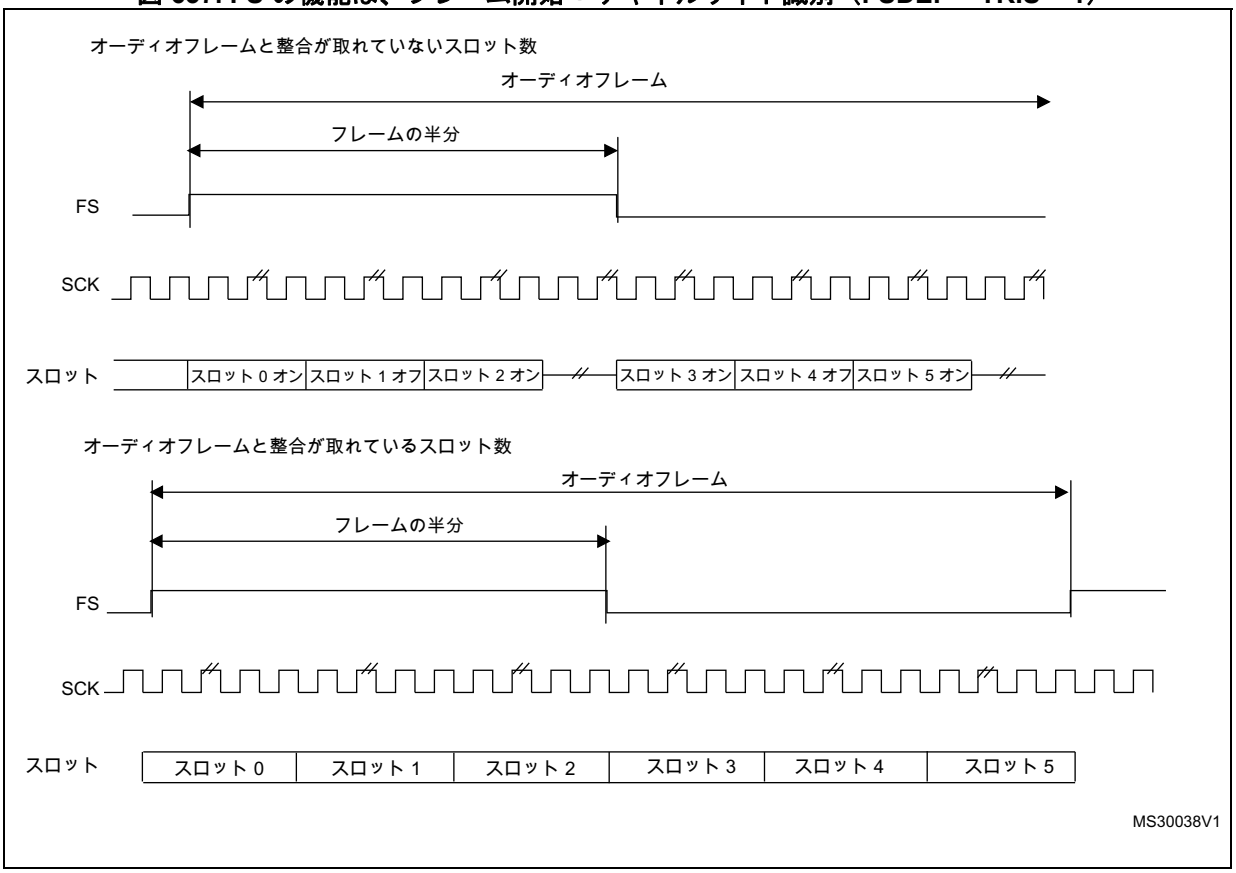
FS 信号の機能

FS 信号は、FS の機能によって意味が異なります。SAI_xFRCR レジスタの FSDEF ビットによりどちらの意味を持たせるかを選択します。

- 0 : 例えば PCM/DSP、TDM、AC’97 オーディオプロトコルのようなフレーム開始
- 1 : 例えば I2S、MSB 詰めまたは LSB 詰めプロトコルのような、オーディオフレーム内のチャネルサイド識別およびフレーム開始

FS 信号をフレーム開始およびフレーム内チャネルサイド識別と見なす場合、指定スロット数は半分を左チャネル用、もう半分を右チャネル用と見なす必要があります。ハーフオーディオフレームのビットクロック数がチャネルサイド専用スロット数より大きい場合、TRIS = 0 であれば、SAI_xCR2 レジスタの残りのビットクロックに対して 0 が送信されます。そうしないと、TRIS = 1 の場合、SD ラインがハイインピーダンスに開放されます。受信モード時、残りのビットクロックサイクルはチャネルサイドが変化するまで無視されます。

図 637. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1）

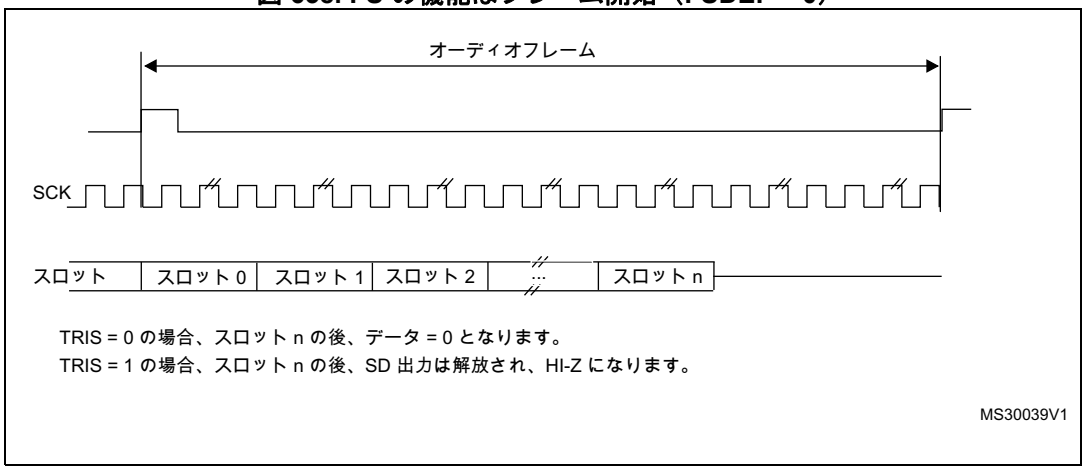


1. フレーム長は偶数である必要があります。

SAI_xFRCR の FSDEF ビットがクリアされたままの場合、FS 信号はフレーム開始に相当し、さらに SAI_xSLOTR の NBSLOT[3:0] ビットで指定されたスロット数に SAI_xSLOTR の SLOTSZ[1:0] ビットで設定されたスロット当たりのビット数を乗算した値が、フレームサイズ（SAI_xFRCR レジスタの FRL[7:0] ビット）より小さい場合、

- SAI_xCR2 レジスタで TRIS = 0 のとき、送信の場合はフレームの終わりまで、最終スロット後の残りのビットは強制的に 0 に設定されます。
- TRIS = 1 のとき、これら残りのビットの転送中、ラインはハイインピーダンス（HI-Z）になります。受信モードでは、これらのビットは破棄されます。

図 638. FS の機能はフレーム開始（FSDEF = 0）



送信モードでオーディオブロックが SD ラインの SPDIF 出力を取得するように設定されている場合は、FS 信号は使用されません。対応する FS の I/O は開放され、他の目的のために空けられます。

51.3.7 スロットの設定

スロットは、オーディオフレームの基本要素です。オーディオフレーム内のスロット数は NBSLOT[3:0] + 1 に等しくなります。

オーディオフレーム当たりの最大スロット数は 16 に固定されています。

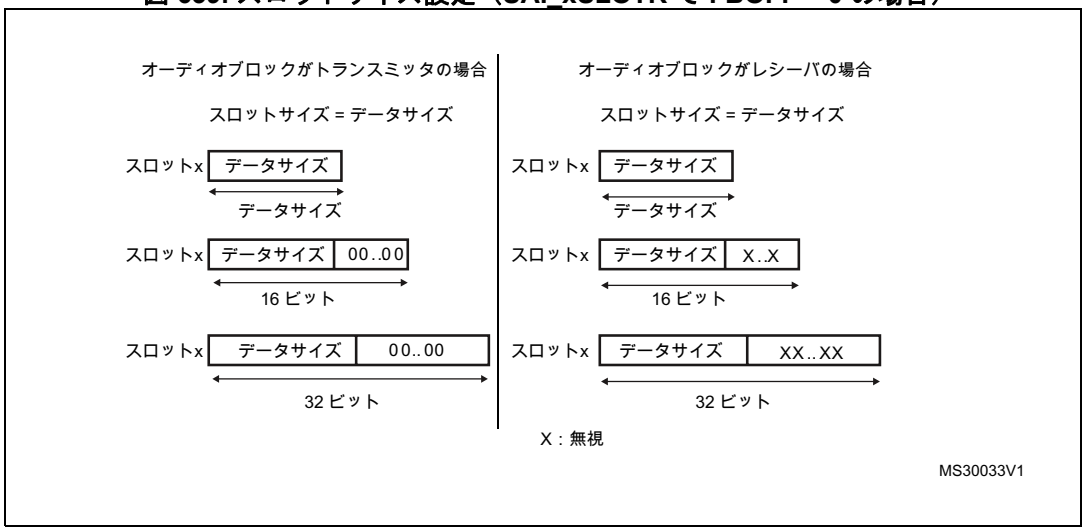
AC'97 プロトコル または SPDIF （PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01の場合）に対しては、スロット数は、そのプロトコル仕様を対象とするよう自動的に設定され、NBSLOT[3:0] の値は無視されます。

各スロットは、SAI_xSLOTR レジスタの SLOTEN[15:0] ビットにより有効/無効を指定することができます。

送信モードでは、無効なスロットが転送されると、SD データラインは TRIS ビットの設定に応じて強制的に 0 に設定されるか、または開放されてハイインピーダンス（HI-Z）状態になり開放されます（*セクション：非アクティブスロットでの出力データライン管理*を参照）。受信モードでは、このスロットの終わりからの受信した値は無視されます。結果として、FIFO アクセスがないため、この非アクティブなスロット状態にリンクした FIFOからの読み出しリクエストや FIFO への書き込みリクエストは起こりません。

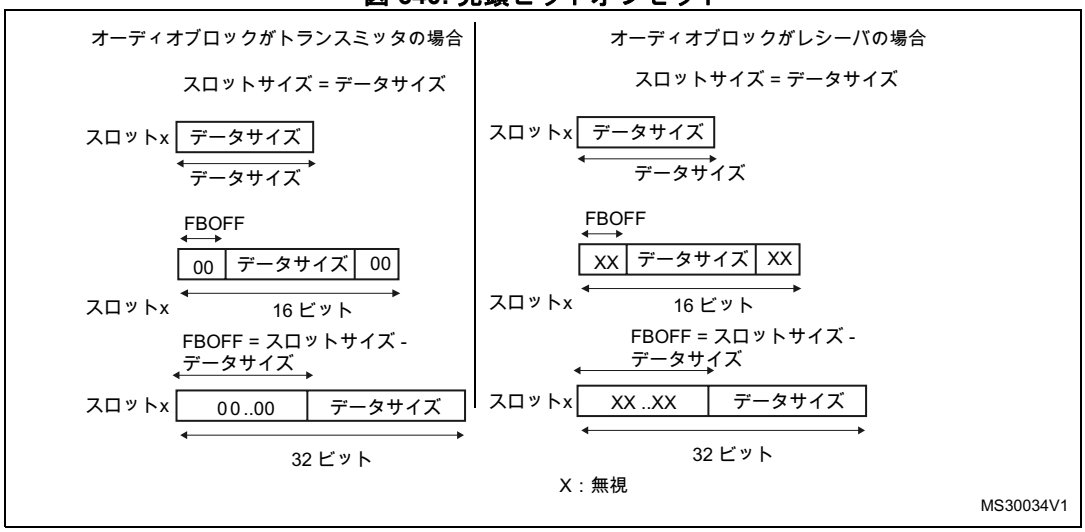
また、スロットのサイズも *図 639* に示すように設定可能です。スロットのサイズは、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビットを設定することで選択できます。サイズは、オーディオフレームの各スロットに同じように適用されます。

図 639. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合）



スロット内で転送する先頭データビットの位置を選択することができます。このオフセットは、SAI_xSLOTR レジスタの FBOFF[4:0] ビットによって設定されます。送信モードでは、スロットの開始からこのオフセット位置に到達するまでの間 0 値が挿入されます。受信時、オフセットフェーズ内のビットは無視されます。この機能は LSB 詰めプロトコルを対象としています （オフセットが、スロットサイズ - データサイズに等しい場合）。

図 640. 先頭ビットオフセット



SAI の誤動作を避けるためには、次の条件を満足する必要があります。

FBOFF ≤(SLOTSZ - DS)、

DS ≤SLOTSZ、

NBSLOT x SLOTSZ ≤FRL（フレーム長）、

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合でも、スロット数は偶数でなければなりません。

AC’97 および SPDIF プロトコル（PRTCFG[1:0] ビット= 10 または PRTCFG[1:0] ビット= 01）では、スロットサイズは、*セクション 51.3.11 : AC’97 リンクコントローラ*で示されるように自動的に設定されます。

51.3.8 SAI クロックジェネレータ

各オーディオサブブロックは、固有のクロックジェネレータを搭載しており、これら 2 つのブロックを完全に独立させています。これら 2 つのクロックジェネレータの間には機能的な違いはありません。

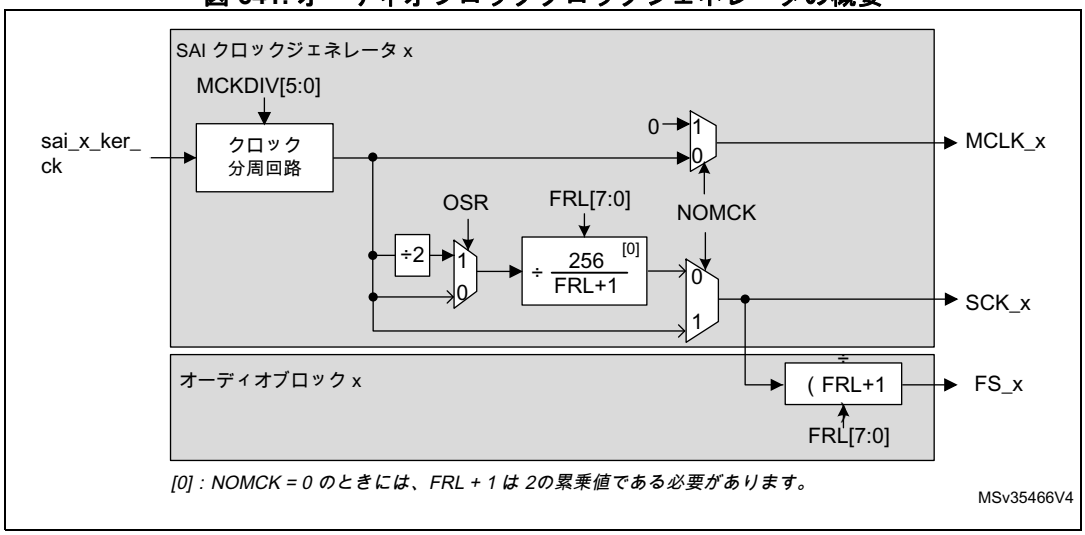
オーディオブロックをマスタに指定すると、クロックジェネレータはビットクロック（SCK_x）と外部デコーダ用のマスタクロック（MCLK_x）を供給します。フレーム同期（FS_x）は、クロックジェネレータから供給される信号からも得られます。SAI クロックジェネレータのクロックソース（sai_x_ker_ck）は、製品のクロックコントローラ（RCC）から供給されます。

オーディオブロックをスレーブに指定すると、クロックジェネレータは OFF になります。NOMCK、MCKDIV、OSR の各ビットの値は無視されます。さらに、MCLK_x I/O ピンがリリースされて、汎用 I/O として使用できます。

（SCK_x）のビットクロックストローブエッジは、SAI_xCR1 レジスタの CKSTR ビットで設定することができます。このビットは、マスタモードとスレーブモードで機能します。

図 641 に、オーディオブロッククロックジェネレータのアーキテクチャを示します。

図 641. オーディオブロッククロックジェネレータの概要



SAI_xCR1 レジスタの NOMCK ビットは、マスタクロックが生成されるかどうかを設定するために使用されます。

SAI がマスタモードで使用された場合には、マスタクロック（MCLK_x）の供給が必要であるかどうかによって、クロックジェネレータの設定は異なります。

NOMCK が 1 にセットされると、マスタクロックは生成されず、フレーム長とフレーム同期周波数の選択に自由度が広がります。また、このピンが GPIO ペリフェラル内で SAI ピンに設定されている場合、MCLK_x 信号は Low レベルにドライブされます。MCKDIV は、SCK_x クロックを必要な周波数に調整するためにそのまま使用できます。

NOMCK が 0 にセットされると、マスタクロックが生成されて、外部デコーダの基準クロックとして使用できます。この場合、フレーム同期とマスタクロックとの周波数比は 512 または 256 に固定され、フレーム長は 2 の累乗値である必要があります。詳細については、以下を参照してください。

MCLK を用いたクロックジェネレータのプログラミング（NOMCK = 0）

この場合、MCLK_x 周波数は次のようになります。

$$F_{MCLK_x} = 256 \times F_{FS_x} \text{ if OSR=0}$$

$$F_{MCLK_x} = 512 \times F_{FS_x} \text{ if OSR=1}$$

MCKDIV が 0 ではない場合、MCLK_x 周波数は下の式によって与えられます。

$$F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期周波数は次のようになります。

$$F_{FS_x} = \frac{F_{sia_x_ker_ck}}{MCKDIV \times (OSR + 1) \times 256}$$

ビットクロック（SCK_x）の周波数は下の式によって与えられます。

$$F_{SCK_x} = \frac{F_{MCLK_x} \times (FRL + 1)}{(OSR + 1) \times 256}$$

注： ***NOMCK = 0 の場合、(FRL + 1) は 2 の累乗値である必要があります。加えて、(FRL + 1) は 8～256 である必要があります（セクション：FS 信号の機能参照）。***

MCKDIV 分周比が奇数である場合には、MCLK のデューティサイクルは 50% にはなりません。MCKDIV が奇数、OSR が 0 に等しく、かつ (FRL+1) = 2^8 であると、ビットクロック信号 (SCK_x) についてもデューティサイクルが 50% とは異なることがあります。

MCKDIV を偶数か大きな値（10 を超える値）に設定することを推奨します。

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

MCLK を用いないクロックジェネレータのプログラミング（NOMCK = 1）

MCKDIV が 0 ではない場合、SCK_x 周波数は下の式によって与えられます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期周波数（FS_x）は下の式によって与えられます。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{(FRL + 1) \times MCKDIV}$$

注： ***NOMCK = 0 である場合、(FRL + 1) は 8 から 256 のうちの任意の値をとることができます。***

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

クロックジェネレータのプログラミング例

*表 399*に、48、96、192 kHz に対するプログラミング例を示します。

表 399. クロックジェネレータのプログラミング例

入力 sai_x_ker_ck クロック周波数	MCLK	F_{MCLK}/ F_{FS}	FRL (1)	OSR	NOMCK	MCKDIV[5:0]	オーディオサンプリング周波（F_{FS}）
98.304 MHz	Y	512	2^N-1	1	0	0 または 1	192 kHz
		512	2^N-1	1	0	2	96 kHz
		512	2^N-1	1	0	4	48 kHz
		256	2^N-1	0	0	2	192 kHz
		256	2^N-1	0	0	4	96 kHz
		256	2^N-1	0	0	8	48 kHz
	N	-	63	-	1	8	192 kHz
		-	63	-	1	16	96 kHz
		-	63	-	1	32	48 kHz

1. N は 3～8の整数値です。

51.3.9 内部 FIFO

SAI 内の各オーディオブロックには固有の FIFO を搭載しています。そのブロックがトランスミッタに指定されるかレシーバに指定されるかによって、FIFO にそれぞれ書き込んだり読み出したりすることができます。したがって、SAI_xSR レジスタの FREQ ビットに対応した FIFO リクエストは 1 つしかありません。

SAI_xIM レジスタの FREQIE ビットが有効化されている場合、割り込みが生成されます。これは次の条件によって変わります：

- FIFO 閾値設定（SAI_xCR2 の FLVL ビット）
- 通信方向（トランスミッタまたはレシーバ）*セクション：送信モードでの割り込み生成* および *セクション：受信モードでの割り込み生成* を参照してください。

送信モードでの割り込み生成

送信モードでの割り込みの生成は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 000b）に設定されている場合、SAI_xDR レジスタに有効なデータがない（SAI_xSR の FLVL[2:0] ビット < 001b）という条件で、割り込みが生成されます（SAI_XSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティ（空）でなくなった場合（SAI_xSR の FLVL[2:0] ビット≠000b）、すなわち FIFO に 1 個以上のデータが格納されている状態になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、データが FIFO の 1/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 010b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 010b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 010b）に設定されている場合、データが FIFO の 1/2 より少ない（SAI_xSR の FLVL[2:0] ビット < 011b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセット

されます）。FIFO の 1/2 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 011b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、データが FIFO の 3/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 100b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 3/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧100b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフル（FTH[2:0] = 100b）に設定されている場合、FIFO がフルでない（SAI_xSR の FLVL[2:0] ビット < 101b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフル（SAI_xSR の FLVL[2:0] ビット = 101b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

受信モードでの割り込み生成

受信モードでの割り込みの発生は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 000b）に設定されている場合、SAI_xDR レジスタに 1 個以上のデータがある（SAI_xSR の FLVL[2:0] ビット ≧ 001b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティになる（SAI_xSR の FLVL[2:0] ビット = 000b）、すなわち FIFO にデータがなくなると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、FIFO データロケーションの 1/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット ≧ 010b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/4 未満（SAI_xSR の FLVL[2:0] ビット < 010b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 010b）に設定されている場合、FIFO データロケーションの 1/2 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧011b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/2 未満（SAI_xSR の FLVL[2:0] ビット < 011b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、FIFO データロケーションの 3/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧100b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 3/4 未満（SAI_xSR の FLVL[2:0] ビット < 100b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO がフル（FTH[2:0] = 100b）に設定されている場合、FIFO がフル（SAI_xSR の FLVL[2:0] ビット = 101b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフルでなくなると（SAI_xSR の FLVL[2:0] ビット < 101b）、この割り込み（SAI_xSR レジスタのFREQ ビット）はハードウェアによってクリアされます。

SAI_xCR1 レジスタの DMAEN ビットがセットされている場合、SAI は、割り込み生成と同様に DMA を使用することができます。FREQ ビットアサーションメカニズムは、上記 FREQIE について説明した割り込み生成メカニズムと同じです。

各 FIFO は 8 ワードの FIFO です。FIFO に対する書き込み操作や読み出し操作は、そのアクセスサイズによらず、1 回につき FIFO の 1 ワード分を対象とします。FIFO の 1 ワードには 1 オーディオスロットが格納されます。FIFO ポインタは、SAI_xDR レジスタに対する各アクセスの後に 1 ワードずつインクリメントされます。

SAI_xDR に書き込む際、データは右詰めである必要があります。

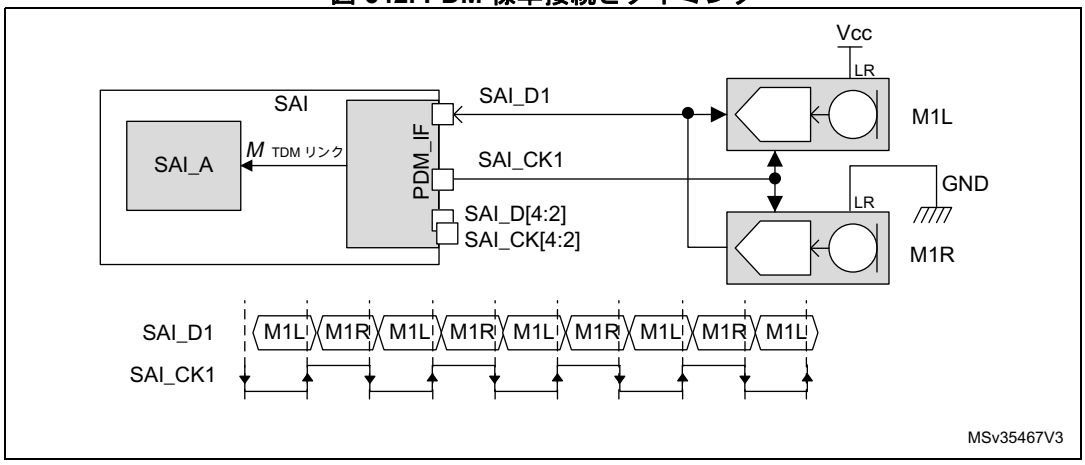
受信データは SAI_xDR で右詰めとなっています。

SAI_xCR2 レジスタの FFLUSH ビットをセットして SAI を無効化すると、FIFO ポインタを再初期化することができます。SAI が有効なときに FFLUSH をセットすると、FIFO 内にあるデータは自動的に失われます。

51.3.10 PDM インタフェース

PDM（パルス密度変調）インタフェースは、デジタルマイクロフォン対応のために提供されています。最大 4 組のデジタルマイクロフォンのペアに並列接続可能です。*図 642* に、PDM インタフェースを介したデジタルマイクロフォンのペアの標準的な接続を示します。どちらのマイクロフォンも同じビットストリームクロックとデータのラインを共有しています。設定ピン（LR）によって、片方のマイクロフォンが SAI_CKx の立ち上がりエッジで有効なデータを供給するのと同時に、もう一方が SAI_CKx の立ち下がりエッジで有効なデータを供給します。

図 642. PDM 標準接続とタイミング



PDM 機能は、TDM MASTER モードで設定された SAI_A サブクロックとともに使用されるためのものです。SAI_B サブクロックとは使用できません。PDM インタフェースは、SAI_A の TDM インタフェースから供給されるタイミング信号を使用して、それを作り替えてビットストリームクロック（SAI_CK[4:1]）を生成します。

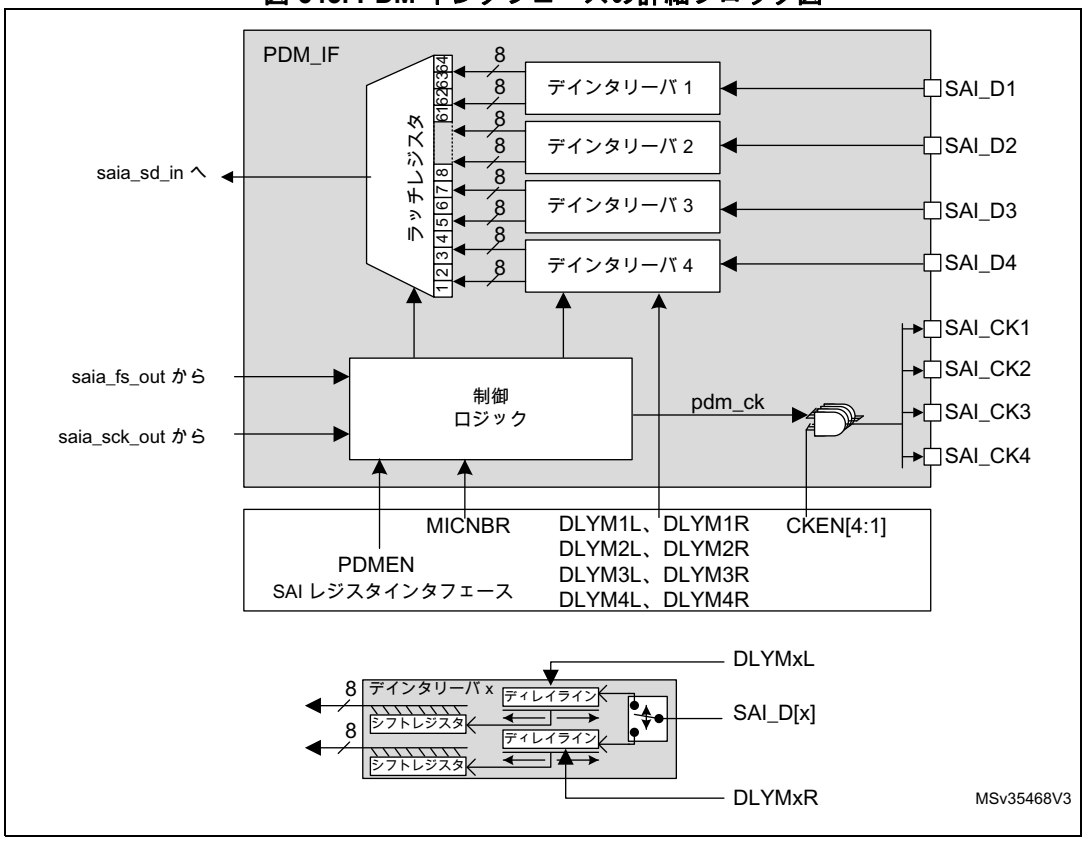
PDM へのデータ処理シーケンスは、次のとおりです。

1. PDM インタフェースは、SAI_A の TDM インタフェースから受信したビットクロックからビットストリームクロックを構築します。
2. マイクロフォンから受信したビットストリームデータ（SAI_D[4:1]）はインタリーブを解かれ、ビットストリームクロック精度で各マイクロフォンのディレイを微調整するために、7 ビットディレイラインを通ります。
3. シフトレジスタによって、それぞれのシリアルビットストリームがバイト列に変換されます。
4. 最後の操作では、TDM インタフェースのシリアルデータラインを通じて、結果として得られたバイト列が SAI_A にシフトアウトされます。

以下の*図 643*に、PDM インタフェースのブロック図をデインタリーバの詳細図とともに示します。

注： ***PDM インタフェースには、ビットストリームから PCM オーディオサンプルを構築するために必要なデシメーションフィルタは組み込まれていません。この操作を行うかどうかは、アプリケーションソフトウェア次第です。***

図 643. PDM インタフェースの詳細ブロック図



PDM インタフェースは、SAI_PDMCR レジスタの PDMEN ビットを通じて有効にできます。ただし、SAI_A ブロックを有効にする前に PDM インタフェースを有効にする必要があります。

メモリのフットプリントを削減するため、アプリケーションが必要とするマイクロフォンの本数を選択できます。この選択は、MICNBR[1:0] ビットで行うことができます。マイクロフォンの本数は、2 本、4 本、6 本、8 本から選択可能です。たとえば、アプリケーションがマイクロホンを 3 本使用するのであれば、4 本を選択する必要があります。

PDM インタフェースの有効化

PDM インタフェースを有効にするには、以下のシーケンスに従います。

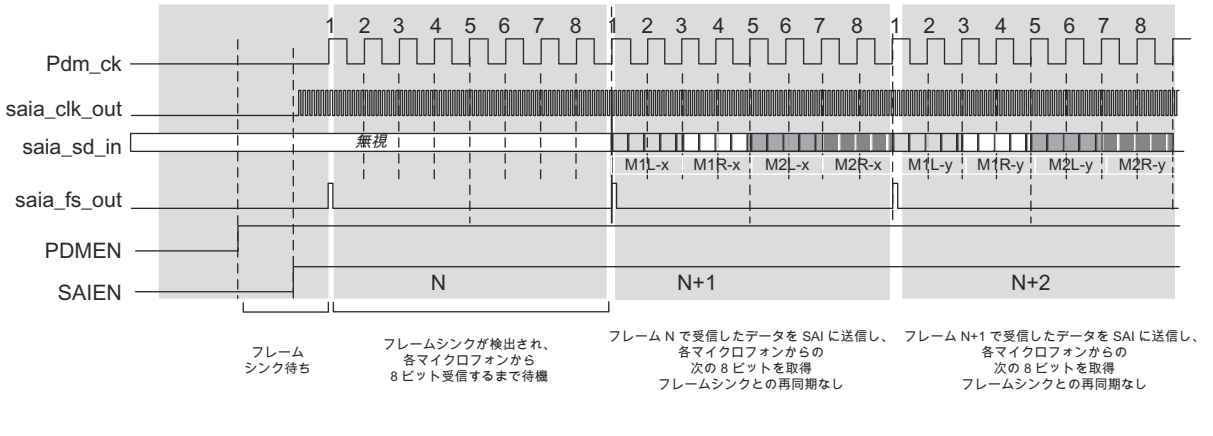
1. SAI_A を TDM MASTER モードに設定します（*表 400*参照）。
2. 次のように PDM インタフェースを設定します。
   a) MICNBR にデジタルマイクロフォンの本数を設定します。
   b) CKEN の対応するビットを 1 にセットして、アプリケーションで必要なビットストリームクロックを有効にします。
3. PDMEN ビットで PDM インタフェースを有効にします。
4. SAI_A を有効にします。

注： ***PDM インタフェースと SAI_A が有効になると、SAI_ADR で受信する最初の 2 つの TDMA フレームは無効ですので、破棄する必要があります。***

起動シーケンス

図 644 に起動シーケンスを示します。PDM インタフェースが有効になると、マイクロフォンサンプルの取得を始める前に、フレーム同期イベントを待ちます。SAI_CK クロック 8 周期後、各マイクロフォンから入力されるデータバイトが利用可能となり、TDM インタフェースを通じて SAI に転送されます。

図 644. 起動シーケンス



SAI_ADR データフォーマット

マイクロフォンから SAI_ADR レジスタに入力されるデータの配列は、次のパラメータに依存しています。

- マイクロフォンの本数
- 選択されたスロット幅
- LSBFIRST ビット

スロット幅によって、SAI_ADR に入力可能な各ワードの有効ビット数が設定されます。

32 ビットのスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 32 ビットの有効なビットが含まれます。これによって、メモリに格納されるワード数が減ります。ただし、対応するには、ソフトウェアが各マイクロフォンのデータのインタリーブを解くための何らかの操作を行う必要があります。

その一方で、8 ビットのスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 8 ビットの有効ビットが含まれます。これによって、メモリに格納されるワード数が増えます。ただし、各ワードにはマイクロフォン 1 本からの情報が含まれているため、これには余計な処理を避けるという利点があります。

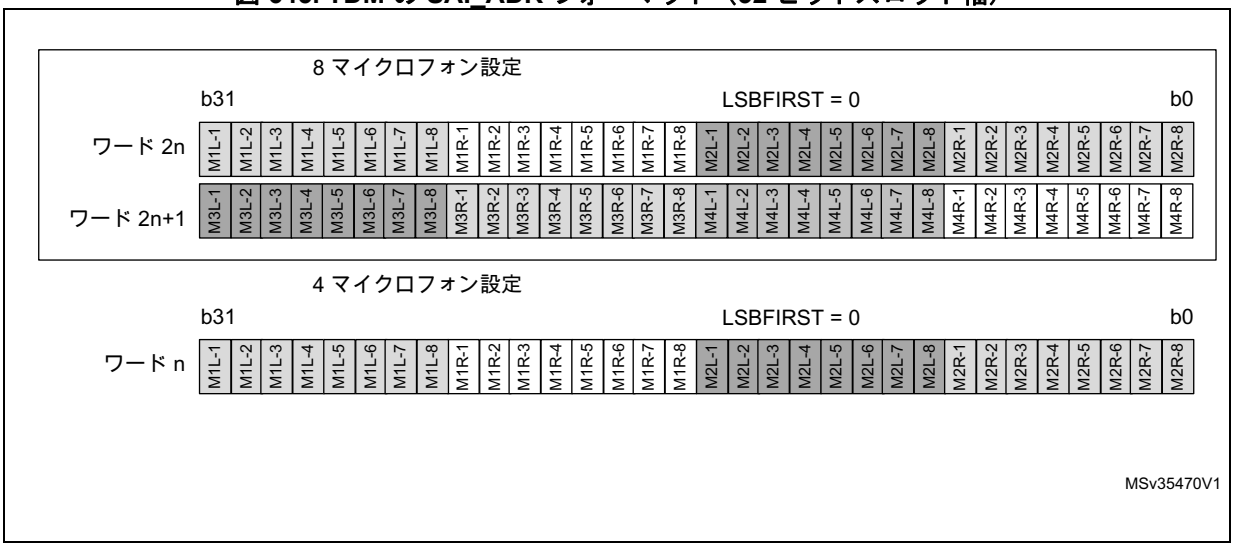
SAI_ADR データフォーマット例

- 32 ビットスロット幅（DS = 0b111 かつ SLOTSZ = 0）*図 645*を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 2 ワードに、各マイクロフォンからの 1 データバイトが含まれています。

  マイクロフォン 4 本の設定では、SAI_ADR レジスタから読み込まれた各ワードに、各マイクロフォンからの 1 データバイトが含まれています。

図 645. TDM の SAI_ADR フォーマット（32 ビットスロット幅）

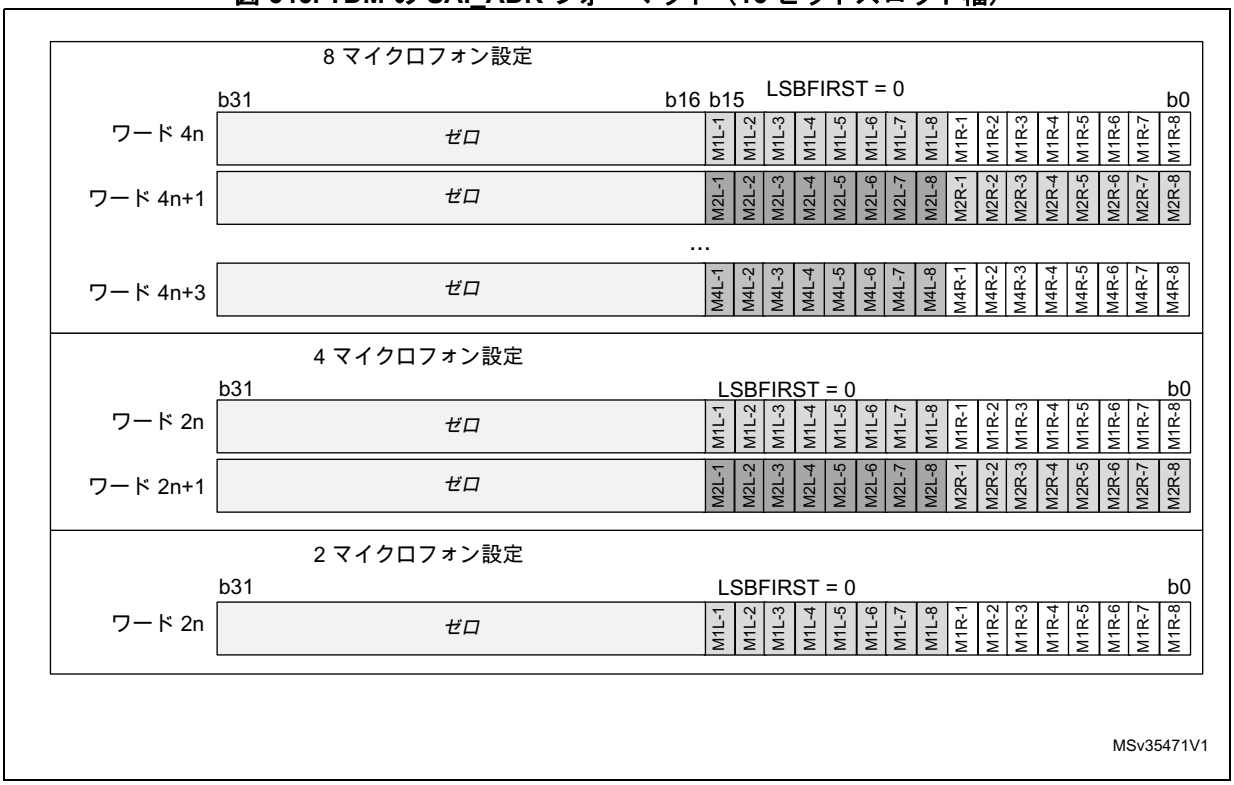


- **16 ビットスロット幅**（DS = 0b100 かつ SLOTSZ = 0）*図 646*を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 4 ワードに、各マイクロフォンからの 1 データバイトが含まれています。SAI_ADR の 16 ビットデータは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 16 ビットワードが最大で 2 ワード必要であり、マイクロフォン 2 本からであれば 1 ワード必要です。

図 646. TDM の SAI_ADR フォーマット（16 ビットスロット幅）



- **8 ビットスロット幅を使用**（DS = 0b010 かつ SLOTSZ = 0）*図 647*を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 8 ワードに、各マイクロフォンからのデータが 1 バイト含まれています。SAI_ADR の 8 ビットデータは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 8 ビットワードが最大で 4 ワード必要であり、マイクロフォン 2 本からであれば 2 ワード必要です。

図 647. TDM の SAI_ADR フォーマット（8 ビットスロット幅）



PDM インタフェースの TDM 設定

SAI_A TDM インタフェースは、マイクロフォンのサンプルを取得するために、内部で PDM インタフェースに接続されています。ユーザアプリケーションは、PDM インタフェースを*表 400*に示すように設定して、PDM インタフェースとの良好な接続を保証する必要があります。

表 400. TDM 設定

ビット フィールド	値	コメント
MODE	0b01	モードは MASTER レシーバである必要があります。
PRTCFG	0b00	TDM 用フリープロトコル
DS	X	フレーム長とスロット数（FRL と NBSLOT）に基づいて、必要なデータフォーマットに応じて調整します。*表 401*を参照してください。

表 400. TDM 設定（続き）

ビット フィールド	値	コメント
LSBFIRST	X	このパラメータは必要なデータフォーマットに応じて使用します。
CKSTR	0	信号の遷移は、SCK_A ビットクロックの立ち上りエッジで発生します。信号はビットクロックの立ち下がりでは安定しています。
MONO	0	ステレオモード
FRL	X	マイクロフォンの本数（MICNBR）に応じて調整します。*表 401*を参照してください。
FSALL	0	ハルス幅は 1 ビットクロックサイクルです。
FSDEF	0	FS 信号がフレームの開始です。
FSPOL	1	FS はアクティブハイです。
FSOFF	0	FS はスロット 0 の先頭ビットでアサートされます。
FBOFF	0	スロットにはオフセットはありません。
SLOTSZ	0	スロットサイズ = データサイズ
NBSLOT	X	スロットサイズとフレーム長（FRL と DS）に基づいて、必要なデータフォーマットに応じて調整します。*表 401*を参照してください。
SLOTEN	X	NBSLOT に応じて設定します。
NOMCK	1	マスタクロック MCLK の生成は不要です。
MCKDIV	X	sai_a_ker_ck 入力に供給される周波数に依存します。 このパラメータは、適切なビットストリームクロック周波数を生成するために調整する必要があります。*表 401*を参照してください。

ビットストリームクロックレートの調整

SAI TDM インタフェースを適切にプログラムするには、ユーザアプリケーションは*表 400*に記載された設定を考慮し、次の規則に従う必要があります。

1. 次の式を使用して、PDM ビットストリームクロックに必要な周波数に応じてビットクロック周波数（F_{SCK_A}）を調整します。

$$F_{SCK_A} = F_{PDM_CK} \times (MICNBR + 1) \times 2$$

MICNBR は、0、1、2、または 3 の値を取ります（0 = マイクロフォン 2本、*セクション 51.5.10 参照*）。

2. 次の式を使用して、フレーム長（FRL）を設定します。

$$FRL = (16 \times (MICNBR + 1)) - 1$$

3. スロットサイズ（DS）を（FRL+1）の倍数に設定します。

表 401. 許容される TDM フレーム設定(1)

マイクロフォンサンプルレート	マイクロフォン本数	必要な SAI_CKx 周波数 (2)	ビットクロック（SCK_A）	フレームシンク（FS_A）	FRL	DS	NBSLOT	コメント
48 kHz	最大 8	3.072 MHz	24.576 MHz	384 kHz	63	0b111	1	フレームあたり 32 ビット 2 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b100	3	フレームあたり 16 ビット 4 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b010	7	フレームあたり 8 ビット 8 スロット
	最大 6	3.072 MHz	18.432 MHz	384 kHz	47	0b110	1	フレームあたり 24 ビット 2 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b100	2	フレームあたり 16 ビット 3 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b010	5	フレームあたり 8 ビット 6 スロット
	最大 4	3.072 MHz	12.288 MHz	384 kHz	31	0b111	0	フレームあたり 32 ビット 1 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b100	1	フレームあたり 16 ビット 2 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b010	3	フレームあたり 8 ビット 4 スロット
	最大 2	3.072 MHz	6.144 MHz	384 kHz	15	0b100	0	フレームあたり 16 ビット 1 スロット
		3.072 MHz	6.144 MHz	384 kHz	15	0b010	1	フレームあたり 8 ビット 2 スロット
16 kHz	最大 8	1.024 MHz	8.192 MHz	128 kHz	63	0b111	1	フレームあたり 32 ビット 2 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b100	3	フレームあたり 16 ビット 4 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b010	7	フレームあたり 8 ビット 8 スロット
	最大 6	1.024 MHz	6.144 MHz	128 kHz	47	0b110	1	フレームあたり 24 ビット 2 スロット
		1.024 MHz	6.144 MHz	128 kHz	47	0b010	5	フレームあたり 8 ビット 6 スロット
	最大 4	1.024 MHz	4.096 MHz	128 kHz	31	0b111	0	フレームあたり 32 ビット 1 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b100	1	フレームあたり 16 ビット 2 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b010	3	フレームあたり 8 ビット 4 スロット
	最大 2	1.024 MHz	2.048 MHz	128 kHz	15	0b100	0	フレームあたり 16 ビット 1 スロット
		1.024 MHz	2.048 MHz	128 kHz	15	0b010	1	フレームあたり 8 ビット 2 スロット

1. TDM 設定の詳細については、*表 400：TDM 設定*を参照してください。SAI に供給される sai_a_ker_ck クロック周波数は、SCK_A 周波数の倍数である必要があり、MCKDIV はそれに応じてプログラムされる必要があります。
2. 上表は、デシメーション比が 64 である場合に許容される設定です。

ディレイラインの調整

PDM インタフェースが有効になると、アプリケーションは、SAI_PDMDLY レジスタを通じて各マイクロフォン入力のディレイセルを実行中に調整できます。

新たなディレイ値は、TDM フレームの 2 周期後に有効になります。

51.3.11 AC’97 リンクコントローラ

SAI は AC’97 リンクコントローラとしても機能することができます。このプロトコルの仕様は次のとおりです：

- スロット番号とスロットサイズは固定です。
- フレーム同期信号は完全に定義されており波形は固定です。

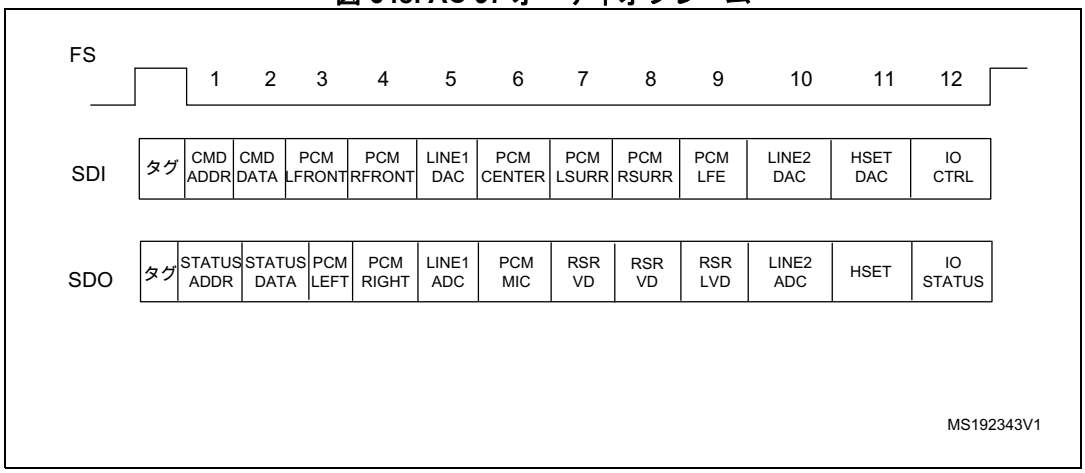
このプロトコルを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットを 10 に設定します。AC’97 モードを選択した場合、使用可能なデータサイズは 16 ビットまたは 20 ビットのみであり、その他の場合、SAI の動作は保証されません。

- したがって、NBSLOT[3:0] ビットと SLOTSZ[1:0] ビットは無視されます。
- スロット数は 13 スロットに固定されています。最初のスロットは 16 ビット幅で、他はすべて 20 ビット幅（データスロット）です。
- SAI_xSLOTR レジスタの FBOFF[4:0] ビットは無視されます。
- SAI_xFRCR レジスタは無視されます。
- MCLK は使用されません。

非同期指定されたブロックからの FS 信号は、自動的に出力として設定されます。これは、マスタ/スレーブの設定にかかわらず AC’97 コントローラリンクが FS 信号を駆動するためです。

図 648 に、AC’97 オーディオフレーム構造を示します。

図 648. AC’97 オーディオフレーム



注： ***AC’97 プロトコルでは、タグのビット 2 は予約済みです（常に 0）。したがって、SAI FIFO に書き込まれる値によらず、タグのビット 2 は 0 レベルに固定されます。***

タグ 表示についての詳細は、AC’97 プロトコル 規格を参照してください。

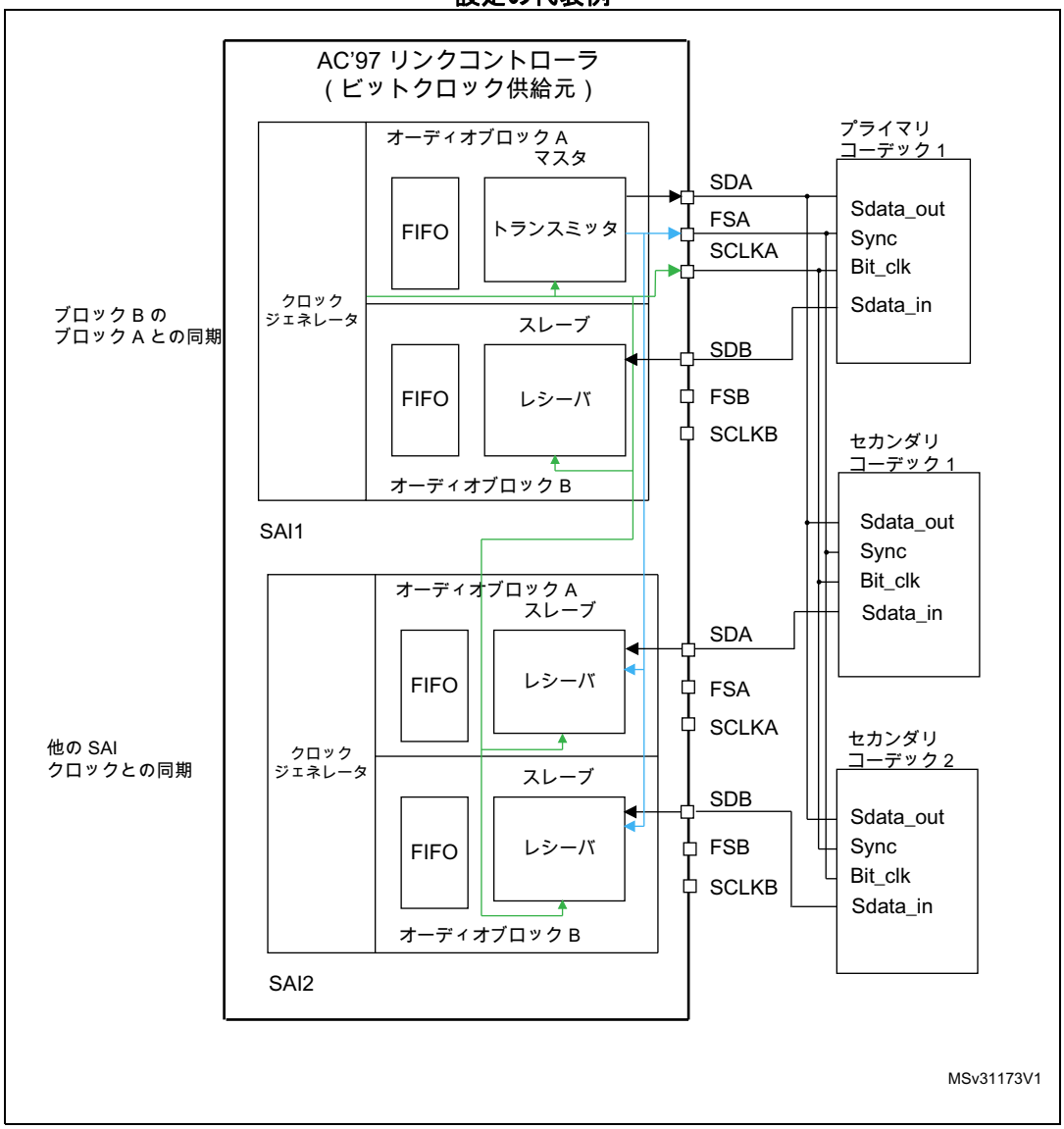
1 つの SAI を AC’97 の 2 点間通信に的を絞って使用することができます。

2 つの SAI を使用することで（デバイスに2 つの内蔵 SAI を搭載）、*図 649*に示す 3 個の外部 AC’97 デコーダを制御することができます。

SAI1 では、オーディオブロック A を非同期マスタトランスミッタに指定する必要がありますが、オーディオブロック B はスレーブレシーバに指定され、オーディオブロック A と内部的に同期します。

SAI2 では、オーディオブロック A と B がどちらもスレーブ受信モードで外部 SAI1 と同期するように設定されます。

図 649. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC'97 設定の代表例



受信モードでは、スロット 0 のコーデックレディビットがローにデコードされた場合、AC'97 リンクコントローラとして機能する SAI は、FIFO リクエストを必要としないため FIFO 内のデータストレージも不要です。SAI_xIM レジスタの CNRDYIE ビットが有効化された場合、SAI_xSR レジスタの CNRDY フラグがセットされ、割り込みが生成されます。このフラグは、AC'97 プロトコル専用です。

AC'97 モードでのクロックジェネレータのプログラミング

AC'97 モードでは、フレーム長は 256 ビットに固定され、その周波数は 48 kHz に設定されます。*セクション 51.3.8 : SAI クロックジェネレータ* に示す式には、適切なフレームレート（F_{FS_x}）を求めるために、FRL = 255 という条件が適用されます。

51.3.12 SPDIF 出力

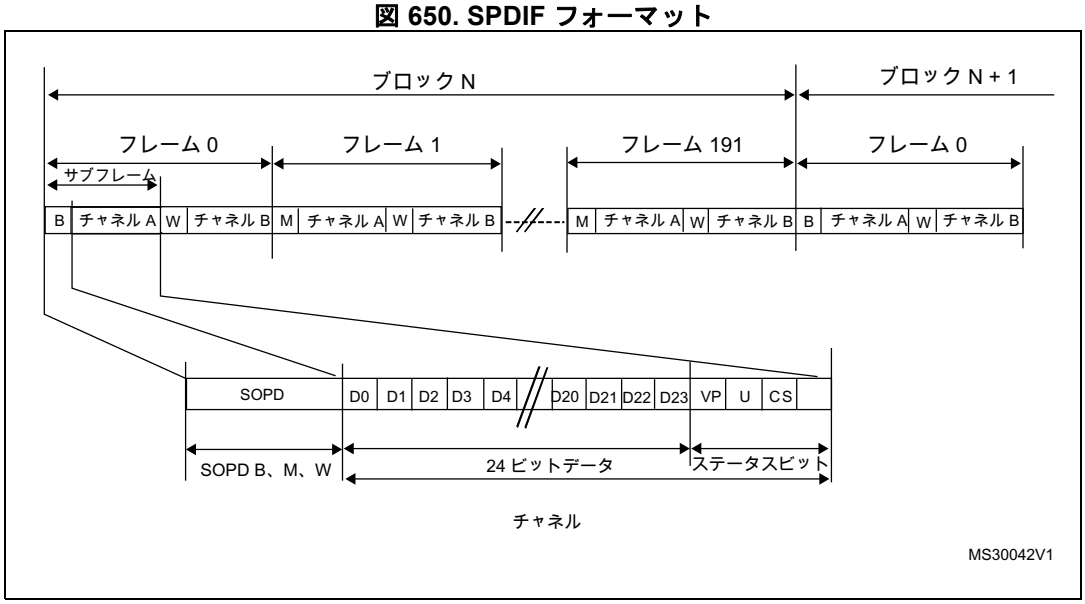
SPDIF インタフェースは送信モードでのみ使用できます。このインタフェースはオーディオ IEC60958 をサポートします。

SPDIF モードを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットに 01 をセットします。

SPDIF プロトコルを使用する場合

- SD データラインのみが有効です。
- I/O ピンの FS、SCK、MCLK は空けてあります。
- MODE[1]ビットはマスタモードで、SAI のクロックジェネレータを有効にしたりSD ラインでのデータレートを管理するためには、0を選択してください。
- データサイズは強制的に 24 ビットに設定されます。SAI_xCR1 レジスタの DS[2:0] ビットにセットされた値は無視されます。
- クロックジェネレータのシンボルレートを、ビットクロックがシンボルレートの 2 倍でなければならないことを考慮して、定義する必要があります。データはマンチェスタープロトコルでコード化されます。
- SAI_xFRCR レジスタおよび SAI_xSLOTR レジスタは無視されます。SAI は、SPDIF プロトコル要件（*図 650*を参照）に適合するように内部的に設定されます。

図 650. SPDIF フォーマット



SPDIF ブロックには 192 個のフレームが含まれています。各フレームは、2 個の 32 ビットサブフレーム（一般に左チャネル用と右チャネル用に 1 つずつ）で構成されています。各サブフレームは SOPD パターン（4 ビット）で構成されています。このパターンを使用して、サブフレームがブロックの先頭である（つまり、チャネル A を識別している）、ブロックのどこかでチャネル A を識別している、またはチャネル B を参照しているのいずれかを指定します（*表 402*を参照）。チャネル情報の次の 28 ビットは 24 ビットのデータと 4 ビットのステータス情報から成ります。

表 402. SOPD パターン

SOPD	プリアンブルのコード化		説明
	最終ビットが 0	最終ビットが 1	
B	11101000	00010111	ブロックの始めにあるチャネル A データ
W	11100100	00011011	ブロックのどこかにあるチャネル B データ
M	11100010	00011101	チャネル A データ

SAI_xDR に格納されているデータは 次のような内容でなければなりません。

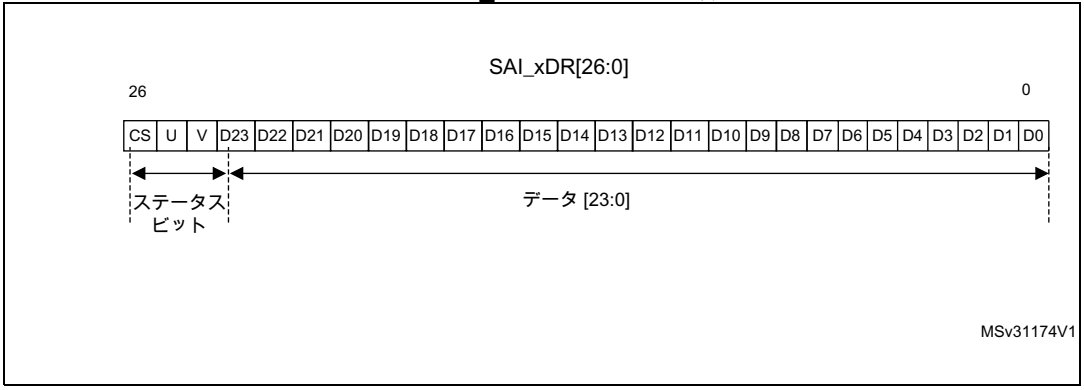
- SAI_xDR[26:24] は、チャネルのステータスビット、ユーザービット、有効性ビットを含みます。
- SAI_xDR[23:0] は、検討されたチャネルに関するデータ 24 ビットを含みます。

データサイズが 20 ビットの場合、データは SAI_xDR[23:4] に配置されます。

データサイズが 16 ビットの場合、データは SAI_xDR[23:8] に配置されます。

SAI_xDR[23] は常に MSB を表します。

図 651. SAI_xDR レジスタの順序



注： ***必ずLSB（最下位ビット）から転送されます。***

SAI はまず最初にブロック内の各サブフレームの適切なプリアンブル部を送信します。次に、SAI_xDR が SD ラインに送信されます（マンチェスターコード化）。SAI は、*表 403*の記載通りに計算されたパリティビットを転送することで、サブフレームを終了させます。

表 403. パリティビットの計算

SAI_xDR[26:0]	転送されるパリティビット P の値
奇数個の”0 ”	0
奇数個の”1 ”	1

SAI は送信モードでしか動作できないため、アンダーランは SPDIF モードの SAI_xSR レジスタで有効な唯一のエラーフラグです。そのため、アンダーラン割り込みまたはアンダーランステータスビッ

トによって検出されたアンダーランエラーから復帰するためには、次のシーケンスを実行する必要があります。

1. DMA が使用されている場合は、DMA ストリームを無効にします（DMA ペリフェラルを使用）。
2. SAI を無効にし、SAI_xCR1 レジスタの SAIXEN ビットをポーリングすることでペリフェラルが物理的に無効化されていることを確認します。
3. SAI_xCLRFR レジスタの COVRUNDR フラグをクリアします。
4. FIFO を一掃するために、SAI_xCR2 の FFLUSH ビットをセットします。

   ソフトウェアで、新しいブロックの始めに今後格納するデータ（プリアンブル部B用のデータ）のアドレスを指し示す必要があります。DMA が使用されている場合、DMA ソースベースアドレスのポインタは適宜更新してください。
5. DMA を使用して、新しいソースベースアドレスに従ってデータ転送を管理する場合は、DMA ストリーム（DMA ペリフェラル）を再び有効にします。
6. SAI_xCR1 レジスタの SAIXEN ビットをセットして、再び SAI を有効にします。

SPDIF ジェネレータモードでのクロックジェネレータのプログラミング

SPDIF ジェネレータモードの場合、SAI はシンボルレートに等しいビットクロックを供給します。下表では、オーディオサンプリングレートに対するシンボルレートの一般例を示します。

表 404. オーディオサンプリング周波とシンボルレート

オーディオサンプリング周波（F_S）	シンボルレート
44.1 kHz	2.8224 MHz
48 kHz	3.072 MHz
96 kHz	6.144 MHz
192 kHz	12.288 MHz

より一般的には、オーディオサンプリングレート（F_S）とビットクロックレート（F_{SCK_X}）の関係は、次の式で求められます。

51.3.13 特有の機能

SAI インタフェースには、選択したオーディオプロトコルに応じて役立つ特定の機能が組み込まれています。これらの機能へは、SAI_xCR2 レジスタの特定ビットを使ってアクセスすることができます。

ミュートモード

ミュートモードは、オーディオサブブロックがトランスミッタまたはレシーバであるとき使用することができます。

送信モード時のオーディオサブブロック

送信モード時、ミュートモードはいつでも選択可能です。ミュートモードは、オーディオフレーム全体に対して有効です。処理中のフレーム内で SAI_xCR2 レジスタの MUTE ビットがセットされると、ミュートモードが有効になります。

ミュートモードビットは、フレームの終わりでのみ認識されます。この時点でミュートモードビットがセットされた場合は、ミュートモードは新しいオーディオフレームの先頭から次に来るフレームの終了までフレーム全体に対して有効になります。その後、このビットを認識して、次のフレームもミュートフレームか否かを決定します。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 以下の場合、ミュートモード中に送信される値が 0 なのか、または各スロットの最後の値なのかを指定することができます。この選択は、SAI_xCR2 レジスタの MUTEVAL ビットを使って行います。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 より大きい場合、各スロットの各ビットで 0 値が送信されるため、SAI_xCR2 の MUTEVAL ビットは無意味となります。

ミュートモード時は、FIFO ポインタのインクリメントは続きます。つまり、FIFO にデータは供給され続けますが、ミュートモードの間は破棄されます。

受信モード時のオーディオサブブロック

受信モードでは、指定されかつ有効なオーディオフレームのすべてのスロットが、指定された連続した数のオーディオフレームに対して 0 を受信したとき、外部 トランスミッタから送信されたミュートモードを検出することが可能です（SAI_xCR2 レジスタの MUTECNT[5:0] ビット）。

ミュートフレーム数が検出されると、SAI_xSR レジスタの MUTEDET フラグがセットされて、SAI_xCR2 の MUTEDETIE ビットがセットされている場合は割り込みを生成することができます。

オーディオサブブロックが無効化されたとき、または有効なスロットがオーディオフレーム内で 1 つ以上のデータを受信したとき、ミュートフレームカウンタがクリアされます。カウンタが MUTECNT[5:0] ビットに指定された値に到達すると、割り込みが 1 回だけ生成されます。次に、カウンタがクリアされると割り込みイベントが再初期化されます。

注： ***ミュートモードは SPDIF オーディオブロックには使用できません。***

モノラル／ステレオモード

送信モードでは、スロット数が 2 と想定し（SAI_xSLOTR で NBSLOT[3:0] = 0001）、メモリ内でデータを前処理することなくモノラルモードを指定できます。このような場合、スロット 0 のデータがデータスロット 1 へ複製されるため、FIFO へのアクセスタイムは半減されます。

モノラルモードを有効にするには、

1. SAI_xCR1 レジスタの MONO ビットに 1 をセットします。
2. SAI_xSLOTR レジスタの NBSLOT ビットに 1、SLOTEN ビットに 3 をセットします。

受信モードでは、MONO ビットをセットすることができますが、送信モードの場合と同様にスロット数が 2 のときだけ意味があります。このビットをセットした場合、スロット 0 のデータだけが FIFO に格納されます。この場合、前のスロットと同じと見なされるため、スロット 1 に属するデータは無視されます。受信時のデータの流れが左右でデータが異なる真のステレオオーディオフローである場合、MONO ビットは意味を成しません。出力ステレオファイルから等価なモノラルファイルへの変換は、ソフトウェアで行われます。

コンパンディングモード

通信アプリケーションでは、データコンパンディングアルゴリズムにより送信データまたは受信データの処理が必要となることがあります。

*図 652.*に示すように、SAI_xCR2 レジスタの COMP[1:0] ビットに応じて（TDM モードを選択した場合のみ使用）、SD シリアル出力ラインに送信する前に処理（圧縮）するか否か、あるいは SD シリアル入力ラインから受信した後にデータを処理（伸長）するか否かをソフトウェアで選択することができます。サポートされている 2 つのコンパンディングモードは µ-Law と A-Law(l対数) で、これらは CCITT G.711 勧告の一部となっています。

米国と日本で採用されているコンパンディング規格は µ-Law であり、14 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 10）。

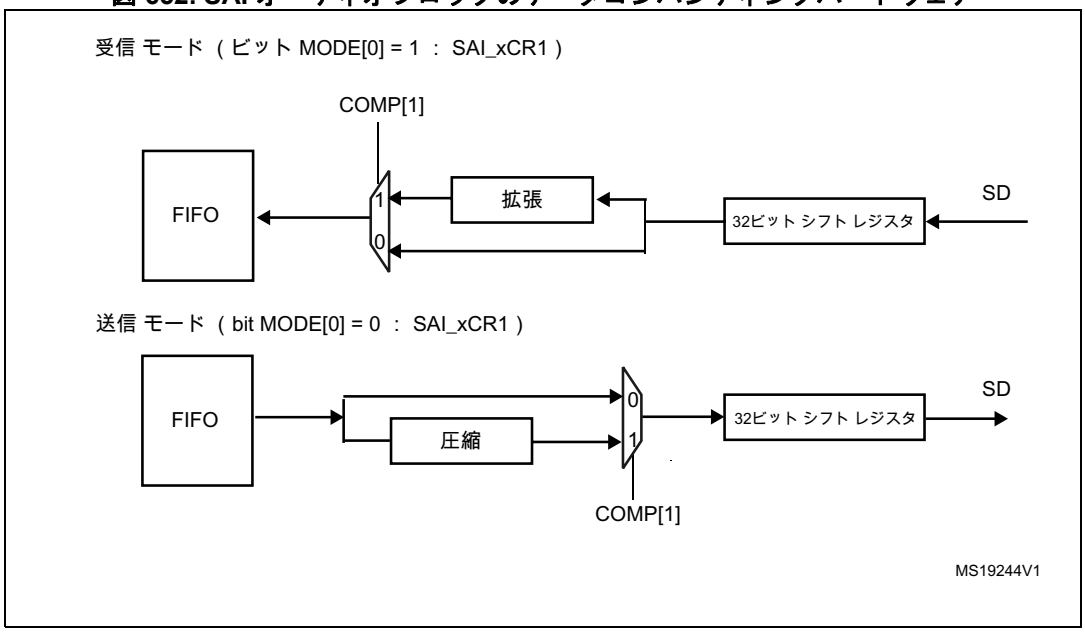
ヨーロッパのコンパンディング規格は A-Law であり、13 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 11）。

コンパンディング規格（µ-Law または A-Law）は、SAI_xCR2 レジスタの CPL ビットの設定に応じて、1 の補数表現または 2 の補数表現に基づいて計算することができます。

µ-Law および A-Law 規格では、データを MSB 詰めの 8 ビットコードにコード化します。コンパンドされたデータは常に 8 ビット幅です。このため、SAI オーディオブロックが有効化されていて（SAI_xCR1 レジスタで SAIXEN ビット = 1）、さらに、COMP[1:0] ビットによりこれら 2 つのコンパンディングモードのいずれかを選択している場合、SAI_xCR1 レジスタの DS[2:0] ビットは 010 に固定されます。

コンパンディングが不要な場合は、COMP[1:0] ビットをクリアしたままにします。

図 652. SAI オーディオブロックのデータコンパンディングハードウェア



1. AC'97 または SPDIF を選択した場合は適用されません。

伸張モードまたは圧縮モードは SAI_xCR2 により自動的に選択されます。

- SAI オーディオブロックがトランスミッタに設定され、かつ SAI_xCR2 レジスタの COMP[1] ビットがセットされている場合、圧縮モードが適用されます。
- SAI オーディオブロックがレシーバに指定された場合、伸長アルゴリズムが適用されます。

非アクティブスロットでの出力データライン管理

送信モードでは、非アクティブスロットがデータラインに送信されたときに出力となっている SDラインの動作を選択することができます（TRIS ビットを使用）。

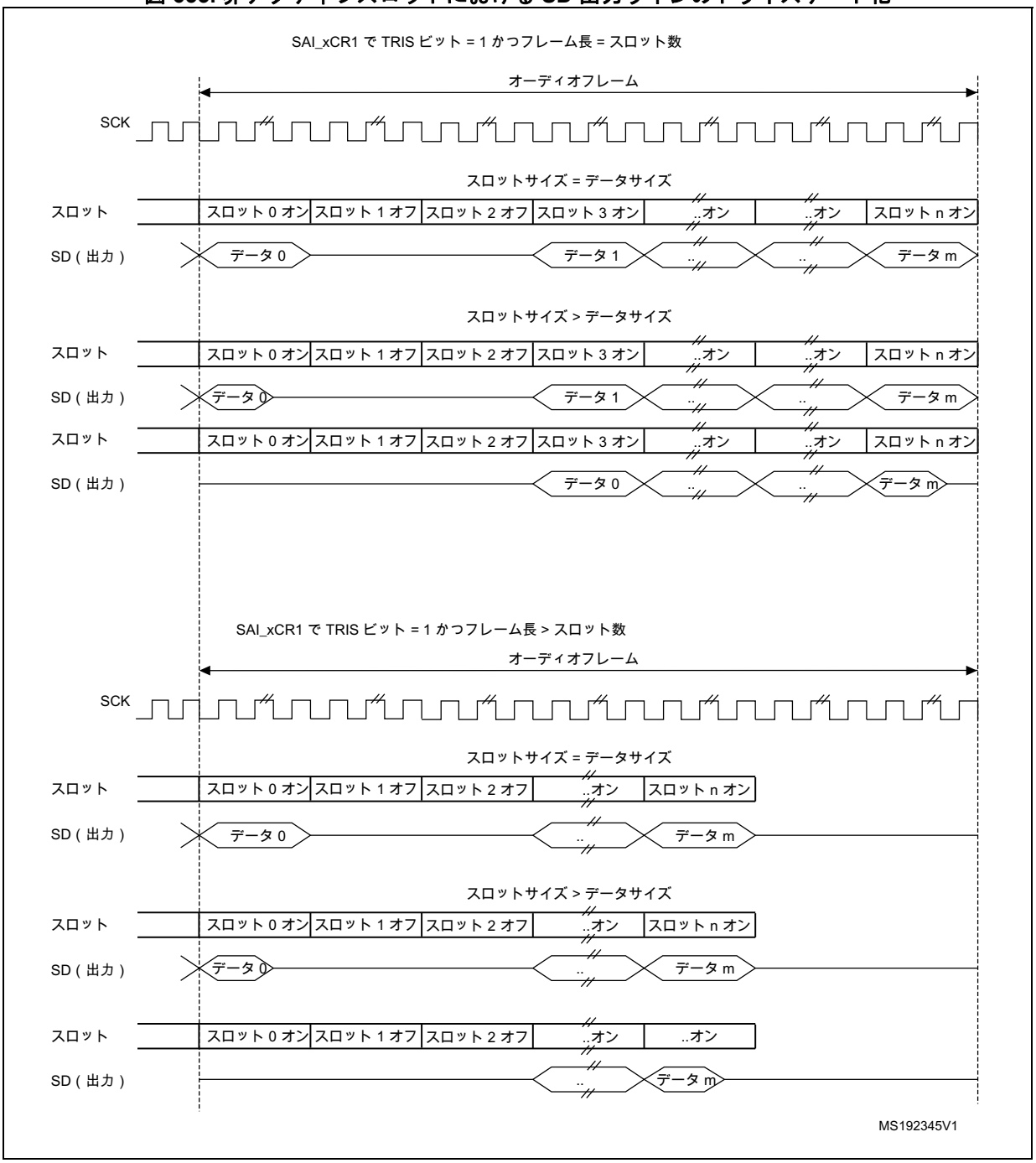
- 非アクティブスロットが送信されるとき、SAI が SD 出力ラインを 0 に固定することもできれば、
- 送信データの最終ビットの終わりでラインを HI-Z 状態にして、このノードに接続された他のトランスミッタへラインを開放することもできます。

2 つのトランスミッタが同じ SD 出力ピンを同時に駆動しないように注意することが重要です。同時駆動した場合、短絡の原因となります。送信間の間隔を確保するため、データが 32 ビットより短い場合は、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビット = 10 に設定することにより、データを 32 ビットに拡張することができます。すると、次のスロットが非アクティブに指定された場合、アクティブスロットの LSB の終わりで（データを 32 ビットに拡張するため 0 を詰めるフェーズで）SD 出力ピンがトライステートになります。

さらに、スロット数とスロットサイズの積がフレーム長より小さい場合、0 を詰めてオーディオフレームを完成させるときに SD 出力ラインがトライステートになります。

このような動作を*図 653* に示します。

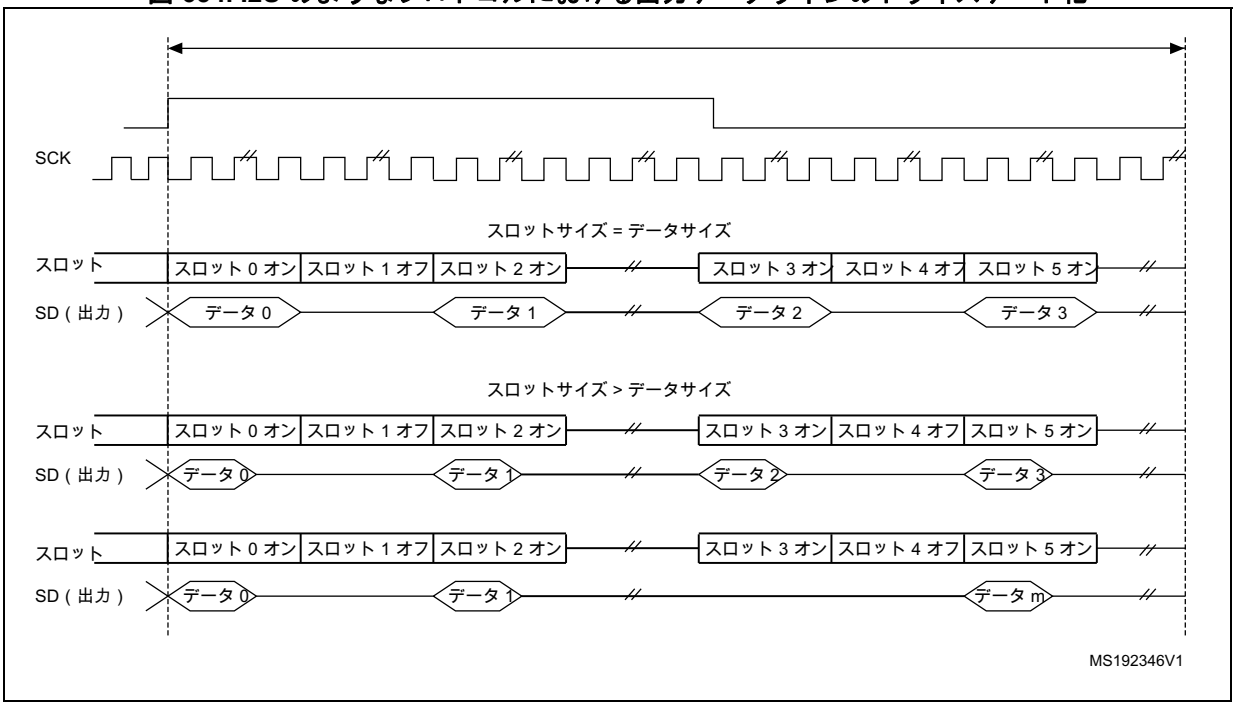
図 653. 非アクティブスロットにおける SD 出力ラインのトライステート化



選択したオーディオプロトコルが FS 信号をフレーム開始とチャネルサイド識別として使用する場合（SAI_xFRCR レジスタでビット FSDEF = 1）、*図 654* に従ってトライステートモードが管理されます

（この図では、SAI_xCR1 レジスタでビット TRIS = 1、FSDEF = 1、ハーフフレーム長 > スロット数 / 2、NBSLOT = 6）。

図 654. I2S のようなプロトコルにおける出力データラインのトライステート化



SAI_xCR2 レジスタの TRIS ビットがクリアされている場合、*図 653* と *図 654* に示す SD 出力ライン上のすべての高インピーダンス状態は 0 値での駆動として置き換えます。

51.3.14 エラーフラグ

SAI は次のエラーフラグを実装しています。

- FIFO オーバーラン／アンダーラン
- 予測フレーム同期検出
- レイトフレーム同期検出
- コーデックノットレディ（AC'97 専用）
- マスタモードにおける不正クロック設定

FIFO オーバーラン/アンダーラン（OVRUDR）

FIFO オーバーラン/アンダーランビットは、SAI_xSR レジスタで OVRUDR と呼ばれます。

オーディオブロックはレシーバまたはトランスミッタのいずれかであり、SAI 内の各オーディオブロックに固有の SAI_xSR レジスタを搭載しているため、オーバーランエラーとアンダーランエラーは、同じビットを使用しています。

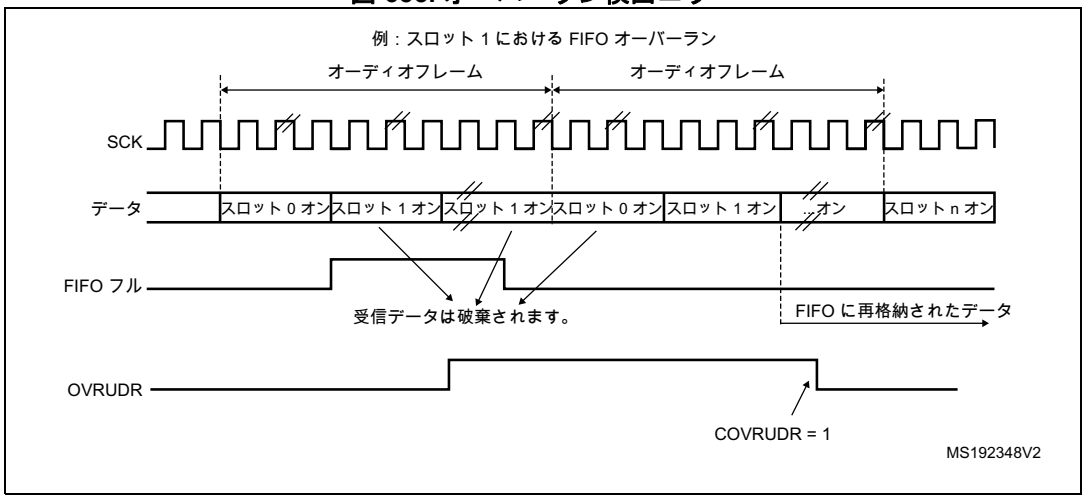
オーバーラン

オーディオブロックがレシーバに設定されていて、FIFO がフルで受信データを保存できないときにオーディオフレームでデータが受信された場合に、オーバーラン状態が発生します。この場合、受信データが失われ、SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割り込みが生成されます。オーバーランが発生したスロット番号が、内部に保存されます。FIFO が新しいデータを格納できるようになるまで、データは

FIFO に保存されません。FIFO に 1 データ分以上の空きがある場合、SAI オーディオブロックレシーバは、オーバーラン状態が検出されたとき内部に保存されたスロット番号からの新しいデータ（新しいオーディオフレームの）を保存します。これにより、転送先メモリでのデータスロットの配置の狂いを回避します（*図 655*を参照）。

SAI_xCLRFR レジスタの COVRUDR ビットがセットされると、OVRUDR フラグがクリアされます。

図 655. オーバーラン検出エラー



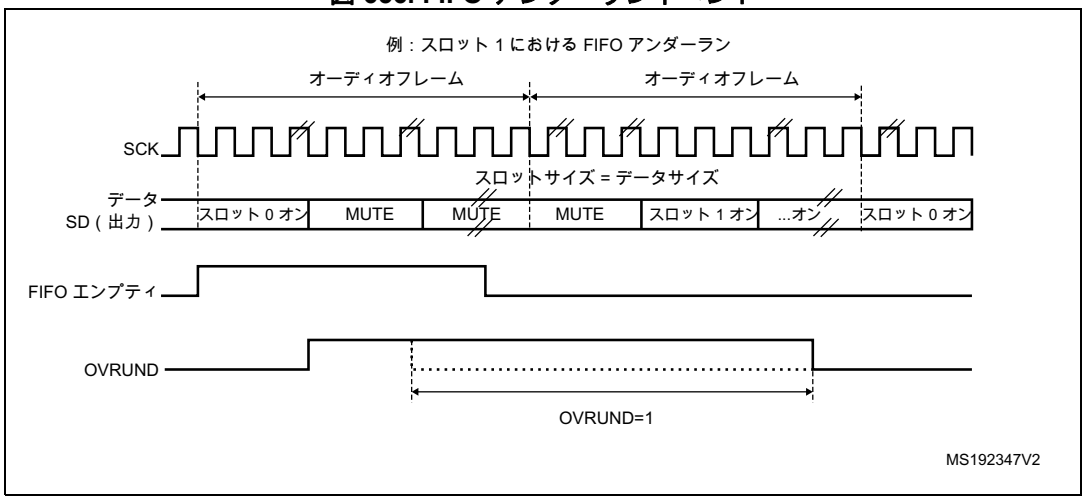
アンダーラン

SAI 内のオーディオブロックがトランスミッタで、かつデータの送信が必要なとき FIFO がエンプティ（空）の場合に、アンダーランが発生します。アンダーランが検出されると、イベントが発生したスロット番号が保存され、FIFO がアンダーランが検出されたスロットに対応するデータを送信する準備ができるまで MUTE 値（00）が送信されます（*図 656*を参照）。これにより、メモリポインタとオーディオフレーム内のスロットの同期はずれを回避します。

アンダーランイベントにより SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合、割り込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの COVRUDR ビットをセットします。

アンダーランイベントは、オーディオサブブロックがマスタまたはスレーブとして設定されている場合に発生します。

図 656. FIFO アンダーランイベント



予測フレーム同期検出（AFSDET）

AFSDET フラグはスレーブモードでのみ使用されます。マスタモードでは決してアサートされません。フレーム長、フレーム極性、フレームオフセットが設定されていて既知であるため、フレーム同期 （FS）が予測より早く検出されることを示します。

予測フレームの検出により、SAI_xSR レジスタの AFSDET フラグがセットされます。

この検出は、現在のオーディオフレームに影響を与えません。現在のオーディオフレームは予測 FS に影響されません。つまり、現在のオーディオフレームを乱すことなく、FS 信号に関する「寄生」イベントフラグが立つということです。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は、割り込みが生成されます。AFSDET フラグをクリアするには、SAI_xCLRFR レジスタの CAFSDET ビットをセットする必要があります。

予測フレーム検出エラーの後にマスタと再同期させるには、次の 4 ステップが必要です。

1. SAI_xCR1 レジスタの SAIXEN ビットをリセットして、SAI ブロックを無効化します。SAI が無効であることを確認するには、SAIXEN ビットを読み返し、それが 0 にセットされていることを確認します。
2. SAI_xCR2 レジスタの FFLUS ビットを使って FIFO を一掃します。
3. SAI ペリフェラルを再度有効化します（SAIXEN ビットを 1 に設定）。
4. SAI ブロックは FS の開始を待ってマスタとの同期を再開します。

注： ***SAIXEN フラグは AC'97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブに指定された場合でも FS 信号を生成するためです。FS 信号は使用されないため、このフラグは SPDIF モードでは 意味を持ちません。***

レイトフレーム同期検出

SAI オーディオブロックがスレーブとして動作する場合のみ、SAI_xSR レジスタの LFSDET フラグをセットすることができます。フレーム長、フレーム極性、フレームオフセット設定は、SAI_xFRCR レジスタから知ることができます。

外部マスタが FS 信号を予想タイミングで送信しない（信号の発生が遅すぎる）場合、LFSDET フラグがセットされ、SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割り込みが生成されます。

SAI_xCLRFR レジスタの CLFSDET ビットがセットされると、LFSDET フラグがクリアされます。

レイトフレーム同期検出フラグは、対応するエラーが検出されるとセットされます。SAI はマスタと再同期させる必要があります（シーケンスは、*セクション：予測フレーム同期検出（AFSDET）*を参照）。

ノイズの多い環境では、オーディオブロックステートマシンが SCK クロック上のグリッチを間違って検出し、SAI データが間違ったフレーム位置にシフトされることがあります。このようなイベントは SAI によって検出され、レイトフレーム同期検出エラーとして報告することができます。

外部マスタが連続モードでオーディオデータフレーム転送を管理していない場合は破壊は起こりませんが、ほとんどの場合、アプリケーションでこの様な使われ方はされません。この場合、LFSDET フラグがセットされます。

注： ***LFSDET フラグは AC'97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブと指定された場合でも FS 信号を生成するためです。FS 信号はプロトコルで使用されないため、このフラグは SPDIF モードでは 意味を持ちません。***

コーデックノットレディ（CNRDY AC'97）

SAI オーディオブロックが AC'97 モードで動作するように設定されている場合にのみ SAI_xSR レジスタの CNRDY フラグが関係します（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10）。SAI_xIM レジスタの CNRDYIE ビットがセットされている場合、CNRDY フラグがセットされたときに割り込みが生成されます。

AC'97 オーディオフレームのタグ 0（スロット 0）の受信中にコーデックが通信の準備ができていないとき、CNRDY フラグがアサートされます。この場合、コーデックが動作可能であることをタグ 0 が示すまではコーデックは動作可能状態にはないため、FIFO に自動的に保存されるデータはありません。コーデックが動作可能状態になると、SAI_xSLOTR レジスタで指定されたすべてのアクティブスロットがキャプチャされます。

CNRDY フラグをクリアするには、SAI_xCLRFR レジスタの CCNRDY ビットをセットする必要があります。

マスタモードでの不正なクロック設定（NOMCK = 0 の場合）

オーディオブロックがマスタとして動作し（MODE[1] = 0）、NOMCK ビットが 0 に等しい場合、次の条件が満たされて SAI が有効になるとすぐに、WCKCFG フラグがセットされます。

- (FRL + 1) は 2の累乗値ではない、かつ
- (FRL + 1) は 8～256の範囲外である。

MODE、NOMCK、SAIXEN ビットは SAI_xCR1レジスタに、FRL ビットは SAI_xFRCR レジスタに属します。

WCKCFGIE ビットがセットされている場合、SAI_xSR レジスタで WCKCFG フラグがセットされると割り込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの CWCKCFG ビットをセットします。

WCKCFG ビットがセットされると、オーディオブロックが自動的に無効化され、ハードウェアによって SAIXEN ビットがクリアされます。

51.3.15 SAI の無効化

SAI_xCR1 レジスタの SAIXEN ビットをクリアすることによって、SAI 内のオーディオブロックをいつでも無効化することができます。すでに開始しているすべてのフレームは、SAI が動作を停止する前に自動的に完了します。現在のオーディオフレーム転送の最後に SAI が完全にオフするまで、SAIXEN ビットはハイを保持します。

他の SAI と同期している SAI 内にオーディオブロックがある場合、マスタになっている SAI を先に無効化する必要があります。

51.3.16 SAI DMA インタフェース

CPU を開放し、バスの帯域幅を最適化するため、各 SAI オーディオブロックには SAI_xDR レジスタ（内部 FIFO へのアクセス用）に対して読み書きを行うための独立した DMA インタフェースを搭載しています。オーディオサブブロック毎に 1 つの DMA チャネルがあり、基本 的なDMA リクエスト／確認応答プロトコルに対応しています。

DMA 転送を行うようにオーディオブロックを設定するには、SAI_xCR1 レジスタの DMAEN ビットをセットします。DMA リクエストは、FIFO 閾値レベルに応じて FIFO コントローラによって直接管理されます（詳細は、*セクション 51.3.9：内部 FIFO*を参照してください）。DMA 転送方向は、SAI オーディオサブブロック設定と次の様に関連付けされています。

- オーディオブロックがトランスミッタとして動作する場合、オーディオブロックの FIFO コントローラが DMA リクエストを出力して SAI_xDR レジスタに書き込まれたデータを FIFO にロードします。
- オーディオブロックがレシーバとして動作する場合、DMA リクエストは SAI_xDR レジスタからの読み出し動作に関わります。

SAI インタフェースを DMA モードに設定するには、次のシーケンスに従ってください。

1. SAI と FIFO 閾値レベルを設定して、DMA リクエストを開始するタイミングを指定します。
2. SAI DMA チャネルを設定します。
3. DMA を有効にします。
4. SAI インタフェースを有効にします。

注： ***SAI ブロックを設定する前に、SAI DMA チャネルを無効化する必要があります。***

51.4 SAI 割り込み

SAI は *表 405*に示す 7 つの割り込みソースに対応しています。

表 405. SAI 割り込みソース

割り込みソース	割り込みグループ	オーディオブロックモード	割り込みイネーブル	割り込みクリア
FREQ	FREQ	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの FREQIE	下記に応じてクリアします。 – FIFO 閾値設定（SAI_xCR2 の FLVL ビット） – 通信方向（トランスミッタ／レシーバ） 詳細については、*セクション 51.3.9：内部 FIFO*を参照してください。
OVRUDR	ERROR	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの OVRUDRIE	SAI_xCLRFR レジスタで COVRUDR = 1
AFSDET	ERROR	スレーブ (AC'97 モード および SPDIF モードでは使用しません。)	SAI_xIM レジスタの AFSDETIE	SAI_xCLRFR レジスタで CAFSDET = 1
LFSDET	ERROR	スレーブ (AC'97 モード および SPDIF モードでは使用しません。)	SAI_xIM レジスタの LFSDETIE	SAI_xCLRFR レジスタで CLFSDET = 1
CNRDY	ERROR	スレーブ (AC'97 モードの場合のみ)	SAI_xIM レジスタの CNRDYIE	SAI_xCLRFR レジスタで CCNRDY = 1
MUTEDET	MUTE	マスタ／スレーブ 受信モードのみ	SAI_xIM レジスタの MUTEDETIE	SAI_xCLRFR レジスタで CMUTEDET = 1
WCKCFG	ERROR	マスタ（SAI_xCR1 レジスタで NOMCK = 0 の場合）	SAI_xIM レジスタの WCKCFGIE	SAI_xCLRFR レジスタで CWCKCFG = 1

割り込みを有効にするには、次のシーケンスに従ってください。

1. SAI 割り込みを無効化します。
2. SAI を設定します。
3. SAI 割り込みソースを設定します。
4. SAI を有効化します。

51.5 SAI レジスタ

51.5.1 グローバル設定レジスタ（SAI_GCR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **SYNCOUT[1:0]**：同期出力

これらのビットは、ソフトウェアによってセット／クリアされます。

00：同期信号は出力されません。オーディオブロックが SPDIF として設定されている場合、SYNCOUT[1:0] は「同期信号出力なし」に設定する必要があります。

01：ブロック A は他の SAI 用に追加された同期に使用します。

10：ブロック B は他の SAI 用に追加された同期に使用します。

11：予約済み。両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SYNCIN[1:0]**：同期入力

これらのビットは、ソフトウェアによってセット／クリアされます。

このフィールドの設定方法については、を参照してください。

両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

2 つのオーディオブロックのうち 1 つが外部 SAI で同期モードで動作するよう設定されている場合（SAI_ACR1 または SAI_BCR1 レジスタで SYNCEN[1:0] = 10）、これらのビットは意味を持ちます。

51.5.2 設定レジスタ 1（SAI_ACR1 / SAI_BCR1）

アドレスオフセット：ブロック A：0x004

アドレスオフセット：ブロック B：0x024

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	OSR	MCKDIV[5:0]						NOMCK	Res.	DMAEN	SAIXEN
					rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **OSR**：マスタクロックのオーバーサンプリング比

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット 25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比 1 のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタクロック周波数は*セクション 51.3.8：SAI クロックジェネレータ*で指定された式で計算されます。

オーディオブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 19 **NOMCK**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータは有効です。

1：マスタクロックジェネレータは無効です。MCKDIV によって制御されるクロック分周器は、ビットクロックの生成に使用できます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： ***オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。***

ビット 16 **SAIXEN**：オーディオブロック有効化（xはA または B）

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 にセットする前に、それが 0 にセットされていることを確認します。そうしないと、有効なコマンドが考慮されないことになります。

このビットを使って、SAIx オーディオブロックの状態を制御することができます。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAIx オーディオブロックは無効です。

1：SAIx オーディオブロックは有効です。

注：　SAIx ブロックをマスタモードに設定した場合、SAIXEN ビットをセットする前に SAIx の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIXEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注：　このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、*セクション：モノラル／ステレオモード*を参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]**：同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他の SAI 内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済み

注：　オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR**：クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST**：LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC'97 ではデータは常に MSB 先頭で転送されるため、このビットは AC'97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 ビットに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済み

001：予約済み

010：8 ビット

011：10 ビット

100：16 ビット

101：20 ビット

110：24 ビット

111：32 ビット

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用し、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（例えば、I2S、LSB/MSB 詰め、TDM、PCM/DSP）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済み

ビット 1:0 **MODE[1:0]**：SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： ***オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。マスタ送信モードでは、オーディオブロックはただちに FS およびクロックの生成を開始します。***

51.5.3 設定レジスタ 2（SAI_ACR2 / SAI_BCR2）

アドレスオフセット：ブロック A：0x008

アドレスオフセット：ブロック B：0x028

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTE VAL	MUTE	TRIS	F FLUSH	FTH		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law(対数) は CCITT G.711 勧告の一部であり、使用される補数のタイプは *CPL ビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、*セクション：コンパンディングモード*を参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注：　コンパンディングモードは、TDM が選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注：　コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割り込みが生成されます。

詳細については、*セクション：ミュートモード*を参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロック SAIXEN を有効にする前に書き込まれる必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、*セクション：ミュートモード*を参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、*セクション：ミュートモード*を参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、*セクション：非アクティブスロットでの出力データライン管理*を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読み出しと書き込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割り込みを無効化する必要があります。

ビット 2:0 **FTH**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済み

110：予約済み

111：予約済み

51.5.4 フレーム設定レジスタ（SAI_AFRCR / SAI_BFRCR）

アドレスオフセット：ブロック A：0x00C

アドレスオフセット：ブロック B：0x02C

リセット値：0x0000 0007

注：　このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19　予約済みであり、リセット値に保持する必要があります。

ビット 18　**FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 ビット前でアサートされます。

ビット 17　**FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16　**FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（例えば、このビットは、I2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。
これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、
ビットクロック数、（SCK）+ 1（FSALL[6:0] + 1）で指定します。
これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。
オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。
これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。
オーディオフレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが 8 ビットで、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。
マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NOMCK = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。
これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

51.5.5 スロットレジスタ（SAI_ASLOTR / SAI_BSLOTR）

アドレスオフセット：ブロック A：0x010

アドレスオフセット：ブロック B：0x030

リセット値：0x0000 0000

注： ***このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。
これらのビットは、ソフトウェアによってセット／クリアされます。
各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。
0：非アクティブスロット。
1：アクティブスロット。
オーディオブロックを無効化して、スロットを有効にする必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。
SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。
オーディオブロックを無効化して、スロット数を設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。

SD ラインの駆動方法については、*セクション ：非アクティブスロットでの出力データライン管理* を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。

01：16 ビット

10：32 ビット

11：予約済み

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

51.5.6 割り込みマスクレジスタ 2（SAI_AIM / SAI_BIM）

アドレスオフセット - ブロック A：0x014

アドレスオフセット - ブロック B：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの LFSDET ビットがセットされている場合は割り込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割り込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割り込み有効化（AC'97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

割り込みを有効化すると、オーディオブロックは、AC'97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割り込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC'97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割り込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割り込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割り込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NOMCK = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割り込みが生成されます。

注： このビットは TDM モードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割り込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割り込みが生成されます。

51.5.7 ステータスレジスタ（SAI_ASR / SAI_BSR）

アドレスオフセット - ブロック A：0x018

アドレスオフセット - ブロック B：0x038

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL**：FIFO レベル閾値。

このビットは読み出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

SAI ブロックをトランスミッタとして設定した場合

000：FIFO エンプティ

001：FIFO ≦ ¼ ただしエンプティではない

010：¼ ＜ FIFO ≦ ½

011：½ ＜ FIFO ≦ ¾

100：¾ ＜ FIFO ただしフルではない

101：FIFO フル

SAI ブロックをレシーバとして設定した場合：

000：FIFO エンプティ

001：FIFO ＜ ¼ ただしエンプティではない

010：¼ ≦ FIFO ＜ ½

011：½ ≦ FIFO ＜ ¾

100：¾ ≦ FIFO ただしフルではない

101：FIFO フル

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイトフレーム同期検出。

このビットは読み出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC’97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読み出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは AC’97または SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると、このフラグはクリアされます。

ビット 4 **CNRDY**：コーデックノットレディ。

このビットは読み出し専用です。

0：外部 AC’97 コーデックが動作可能状態です。

1：外部 AC’97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC’97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読み出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読み出しまたは書き込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書き込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読み出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割り込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読み出し専用です。

0：クロック設定は正常です。

1：クロック設定が *セクション 51.3.6：フレーム同期* で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NOMCK = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読み出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読み出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

51.5.8 クリアフラグレジスタ（SAI_ACLRFR / SAI_BCLRFR）

アドレスオフセット - ブロック A：0x01C

アドレスオフセット - ブロック B：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。
このビットは、AC'97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。
このビットは AC'97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデックノットレディフラグのクリア
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。
SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NOMCK = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

51.5.9 データレジスタ（SAI_ADR / SAI_BDR）

アドレスオフセット - ブロック A：0x020

アドレスオフセット - ブロック B：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATA[31:0]**：データ

FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書き込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読み出しが行われます。

51.5.10 PDM 制御レジスタ（SAI_PDMCR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CKEN4 (1)	CKEN3 (1)	CKEN2 (1)	CKEN1 (1)	Res.	Res.	MICNBR[1:0] (1)		Res.	Res.	Res.	PDMEN
				rw	rw	rw	rw			rw	rw				rw

1. これらのフィールドを PDMEN = 1 のときに設定するのは推奨しません。

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **CKEN4**：ビットストリームクロック 4 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK4 クロック無効
1：SAI_CK4 クロック有効

ビット 10 **CKEN3**：ビットストリームクロック 3 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK3 クロック無効
1：SAI_CK3 クロック有効

ビット 9 **CKEN2**：ビットストリームクロック 2 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK2 クロック無効
1：SAI_CK2 クロック有効

ビット 8 **CKEN1**：ビットストリームクロック 1 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK1 クロック無効
1：SAI_CK1 クロック有効

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **MICNBR**：マイクロフォンの本数
このビットは、ソフトウェアによってセット／クリアされます。
0b00：マイクロフォン 2 本の設定
0b01：マイクロフォン 4 本の設定
0b10：マイクロフォン 6 本の設定
0b11：マイクロフォン 8 本の設定

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **PDMEN**：PDM イネーブル
このビットは、ソフトウェアによってセット／クリアされます。このビットを使って、PDM インタフェースブロックの状態を制御することができます。
PDM インタフェースを有効にする前に SAI がすでに TDM マスタモードで動作していることを確認してください。
0：PDM インタフェースが無効です。
1：PDM インタフェースが有効です。

51.5.11 PDM ディレイレジスタ（SAI_PDMDLY）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYM4R[2:0]			Res.	DLYM4L[2:0]			Res.	DLYM3R[2:0]			Res.	DLYM3L[2:0]		
	rw				rw				rw				rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYM2R[2:0]			Res.	DLYM2L[2:0]			Res.	DLYM1R[2:0]			Res.	DLYM1L[2:0]		
	rw				rw				rw				rw		

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **DLYM4R**：**ペア No.4** の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **DLYM4L**：ペア No.4 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **DLYM3R**：ペア No.3 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **DLYM3L**：ペア No.3 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **DLYM2R**：ペア No.2 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DLYM2L**：ペア No.2 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
0b000：遅延なし
0b001：1 T_{SAI_CK} 周期のディレイ
0b010：2 T_{SAI_CK} 周期のディレイ
...
0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **DLYM1R**：ペア No.1 の 2 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

0b000：遅延なし

0b001：1 T_{SAI_CK} 周期のディレイ

0b010：2 T_{SAI_CK} 周期のディレイ

...

0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **DLYM1L**：ペア No.1 の 1 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

0b000：遅延なし

0b001：1 T_{SAI_CK} 周期のディレイ

0b010：2 T_{SAI_CK} 周期のディレイ

...

0b111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

51.5.12 SAI レジスタマップ

次の表に SAI レジスタの一覧を示します。

表 406. SAI レジスタマップとリセット値

オフセット	レジスタとリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	SAI_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCOUT[1:0]		Res.	Res.	SYNCIN[1:0]	
	リセット値																											0	0			0	0
0x0004 または 0x0024	SAI_xCR1	Res.	Res.	Res.	Res.	Res.	OSR	MCJDIV[3:0]						NOMCK	Res.	DMAEN	SAIXEN	Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]		Res.	PRTCFG[1:0]		MODE[1:0]	
	リセット値						0	0	0	0	0	0	0	0		0	0			0	0	0	0	0	0	0	1	0		0	0	0	0
0x0008 または 0x0028	SAI_xCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMP[1:0]		CPL	MUTECN[5:0]						MUTE VAL	MUTE	TRIS	FFLUS	FTH		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C または 0x002C	SAI_xFRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF	Res.	FSALL[6:0]							FRL[7:0]							
	リセット値														0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0010 または 0x0030	SAI_xSLOTR	SLOTEN[15:0]																Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0014 または 0x0034	SAI_xIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDETIE	CNRDYIE	FREQIE	WCKCFG	MUTEDET	OVRUDRIE
	リセット値																										0	0	0	0	0	0	0
0x0018 または 0x0038	SAI_xSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
	リセット値														0	0	0										0	0	0	0	1	0	0
0x001C または 0x003C	SAI_xCLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	CAFSDET	CNRDY	Res.	WCKCFG	MUTEDET	OVRUDR
	リセット値																										0	0	0		0	0	0
0x0020 または 0x0040	SAI_xDR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	SAI_PDMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
	リセット値																					0	0	0	0			0	0				0

表 406. SAI レジスタマップとリセット値（続き）

オフセット	レジスタとリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0048	SAI_PDMDLY	Res.	DLYM4R[2:0]			Res.	DLYM4L[2:0]			Res.	DLYM3R[2:0]			Res.	DLYM3L[2:0]			Res.	DLYM2R[2:0]			Res.	DLYM2L[2:0]			Res.	DLYM1R[2:0]			Res.	DLYM1L[2:0]		
	リセット値		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2*を参照してください。

52 SPDIF レシーバインタフェース（SPDIFRX）

52.1 SPDIFRX インタフェースの概要

SPDIFRX インタフェースは S/PDIF オーディオプロトコルを処理します。

52.2 SPDIFRX の主な機能

- 最大 4 個の入力が利用可能。
- 自動シンボルレート検出。
- 最大シンボルレート：12.288 MHz
- 8～192 kHz のステレオストリームをサポート。
- デジタルオーディオインタフェース規格 IEC-60958 および IEC-61937、民生用アプリケーションをサポート。
- S/PDIF フロー内部に SOPD の B、M、W を挿入。
- パリティビットの管理。
- オーディオサンプリング用の DMA を使用した通信。
- 制御およびユーザチャネル情報用の DMA を使用した通信。
- 割り込み機能を搭載。

52.3 SPDIFRX の機能詳細

SPDIFRX ペリフェラルは、IEC-60958 および IEC-61937 に適合した S/PDIF フローを受信するように設計されています。これらの規格は、高いサンプリングレートに適した単純なステレオストリーム、およびドルビーサウンドや DTS サウンドと言われるような圧縮されたマルチチャネルサラウンドサウンドをサポートしています。

レシーバは、シンボルレートを検出するために必要なすべての機能を提供し、受信データをでコードします。ユーザ情報およびチャネル情報の専用パスを使用して、インタフェース処理を容易にすることができます。*図 657* に簡略化されたブロック図を示します。

SPDIFRX_DC ブロックは、SPDIFRX_IN[4:1] 入力から受け取った S/PDIF ストリームをデコードするブロックです。このブロックは、受信信号の再サンプリングを行い、マンチェスタストリームをデコードし、フレーム、サブフレーム、ブロックの要素を認識します。また、このブロックは REG_IF 部分、デコードされたデータ、および関連のステータスフラグを出力します。

ペリフェラルは APB1 バスに経由で完全にに制御され、次の 2 つの DMA チャネルを処理することができます。

- オーディオサンプル転送専用の DMA チャネル
- IEC60958 チャネルステータスおよびユーザ情報の転送専用の DMA チャネル

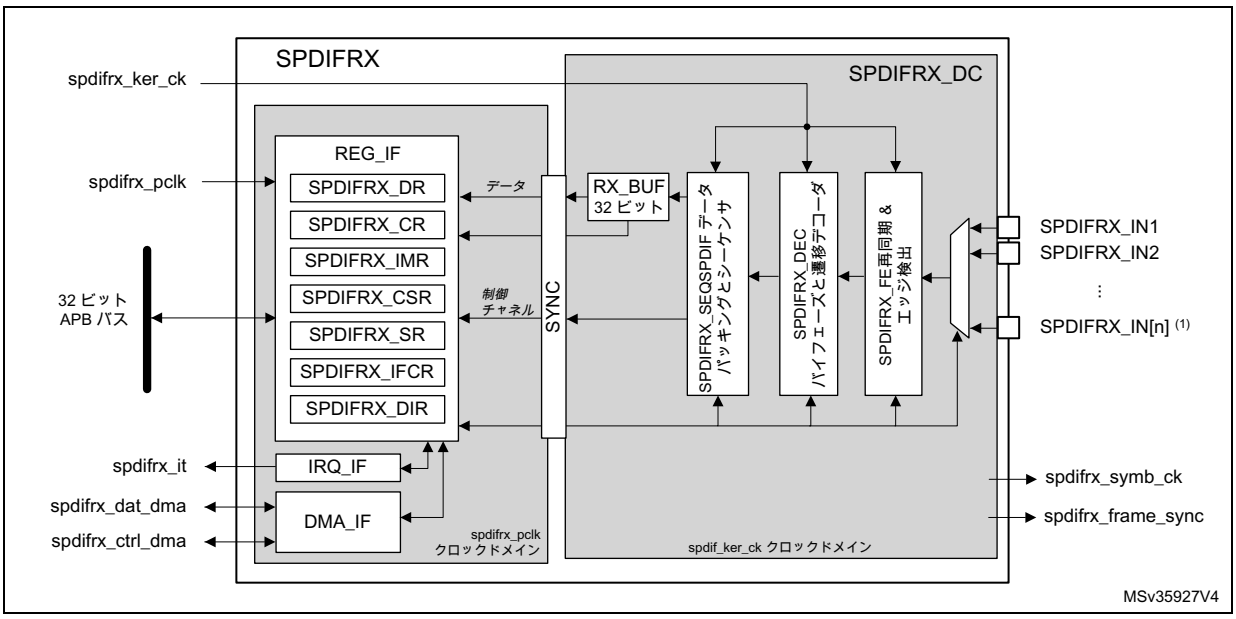
割り込みサービスは、DMA に対するオルタネート機能として、またはペリフェラルのシグナリングエラーやキーステータス用として使用することもできます。

また、SPDIFRXは、サブフレームのプリアンブルが検出されるたびにトグルする **spdifrx_frame_sync**と呼ばれる信号も提供します。したがって、デューティサイクルは 50 %、周波数はフレームレートに等しくなります。

この信号をタイマイベントに接続して、周波数ドリフトを計算することができます。

また、SPDIFRX はシンボルレートでトグルする **spdifrx_symb_ck** という名前の信号の提供も行います。

図 657. SPDIFRX ブロック図



1. "n"は 4 に固定されています。

52.3.1 SPDIFRX ピンおよび内部信号

*表 407*に SPDIFRX の内部入出力信号のリストを、*表 408*に SPDIFRX のピン（オルタネート機能）のリストをそれぞれ示します。

表 407. SPDIFRX 内部入力／出力信号

信号名	信号タイプ	説明
spdifrx_ker_ck	デジタル入力	SPDIFRX カーネルクロック
spdifrx_pclk	デジタル入力	SPDIFRX レジスタインタフェースクロック
spdifrx_it	デジタル出力	SPDIFRX グローバル割り込み
spdifrx_dat_dma	デジタル入出力	データ転送用の SPDIFRX DMA リクエスト（と確認応答）
spdifrx_ctrl_dma	デジタル入出力	チャネルステータスおよびユーザ情報の転送用の SPDIFRX DMA リクエスト（と確認応答）
spdifrx_frame_sync	デジタル出力	SPDIFRX フレームレート同期信号
spdifrx_symb_ck	デジタル出力	SPDIFRX チャネルシンボルクロック

表 408. SPDIFRX ピン

信号名	信号タイプ	説明
SPDIFRX_IN1	デジタル入力	S/PDIF 信号用入力 1
SPDIFRX_IN2	デジタル入力	S/PDIF 信号用入力 2
SPDIFRX_IN3	デジタル入力	S/PDIF 信号用入力 3
SPDIFRX_IN4	デジタル入力	S/PDIF 信号用入力 4

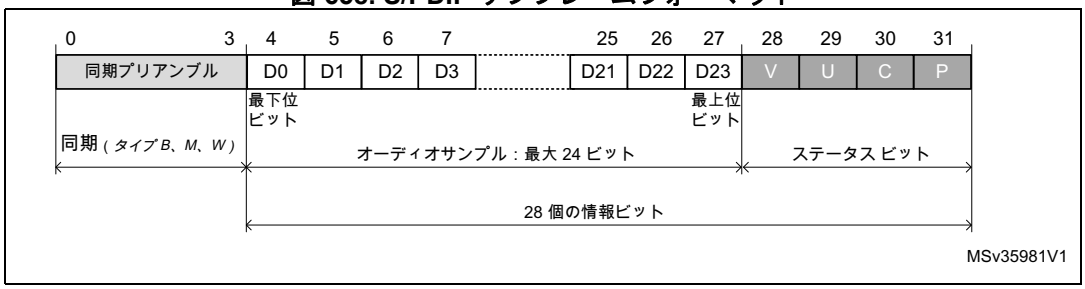
52.3.2 S/PDIF プロトコル（IEC-60958）

S/PDIF ブロック

S/PDIF フレームは 2 つのサブフレームで構成されています（*図 659*を参照）。各サブフレームは 32 ビット（またはタイムスロットと呼ばれる）で構成されます。

- ビット 0～3 には同期プリアンブルのうちの 1 つが入っています。
- ビット 4～27 にはオーディオサンプルワードがリニア 2 の補数表現で入っています。最上位ビット（MSB）はビット 27 に書き込まれています。20 ビットコーディング範囲が使用されている場合、ビット 8～27 にはオーディオサンプルワードが入り、ビット 8 は LSB が入ります。
- ビット 28（有効性ビット“V”）はデータが有効であるか否かを示します（たとえば、データをアナログ変換する場合）。
- ビット 29（ユーザデータビット“U”）にはユーザデータ情報（コンパクトディスクのトラック数など）が入っています。
- ビット 30（チャネルステータスビット“C”）には、チャネルステータス情報（サンプリングレート、複製保護など）。
- ビット 31（パリティビット“P”）には、パリティビットが入ります。たとえば、偶数パリティの場合、ビット 4～31のビット列に偶数個の“1”と偶数個の“0”が入るように送られてきます。

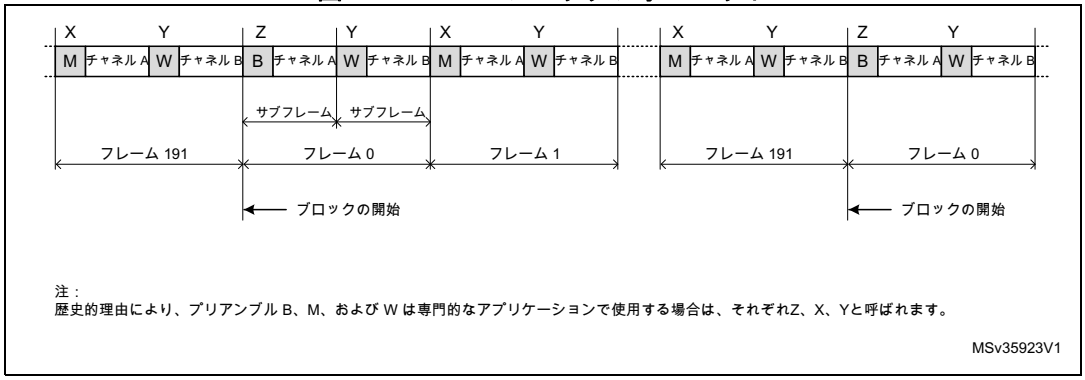
図 658. S/PDIF サブフレームフォーマット



線形コード化されたオーディオアプリケーションでは、最初のサブフレーム（左側、またはステレオ動作ではチャネル A／モノラル動作では一次チャネル）は通常プリアンブル M から始まります。ただし、プリアンブルは 192 フレームに 1 回プリアンブル B に変わることで、チャネルステータスおよびユーザ情報の構成に使用されるブロック構造の開始を識別します。2 番目のサブフレーム（右側、またはステレオ動作ではチャネル B／モノラル動作では二次チャネル）は常にプリアンブル W から始まります。

S/PDIF ブロックには 192 ペアの 32 ビットサブフレームが格納されています。

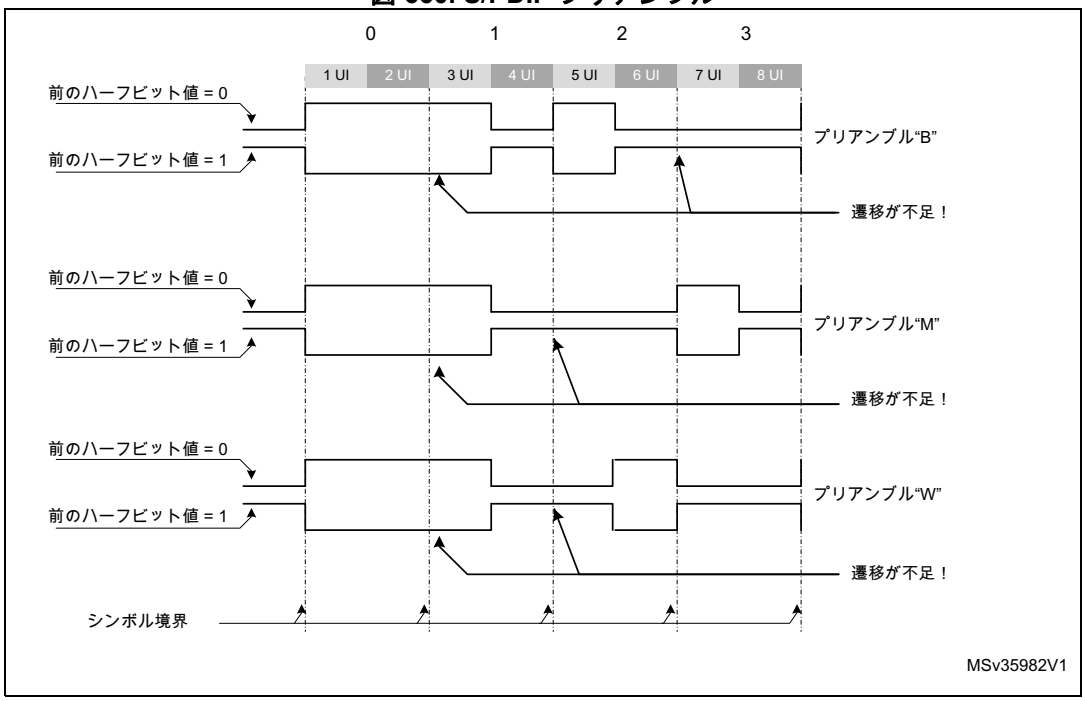
図 659. S/PDIF ブロックフォーマット



同期プリアンブル

プリアンブルパターンは、前のハーフビット値に応じて反転されたり、されなかったりします。ここで言う前のハーフビット値とは、最初のフレームの最初のプリアンブル B への転送を有効にする前のラインのレベルのことです。他のプリアンブルの場合の前のハーフビット値は、前のサブフレームのパリティビットの 2 番目のハーフビットのことです。プリアンブルパターン B、M、W を *図 660*に示します。

図 660. S/PDIF プリアンブル



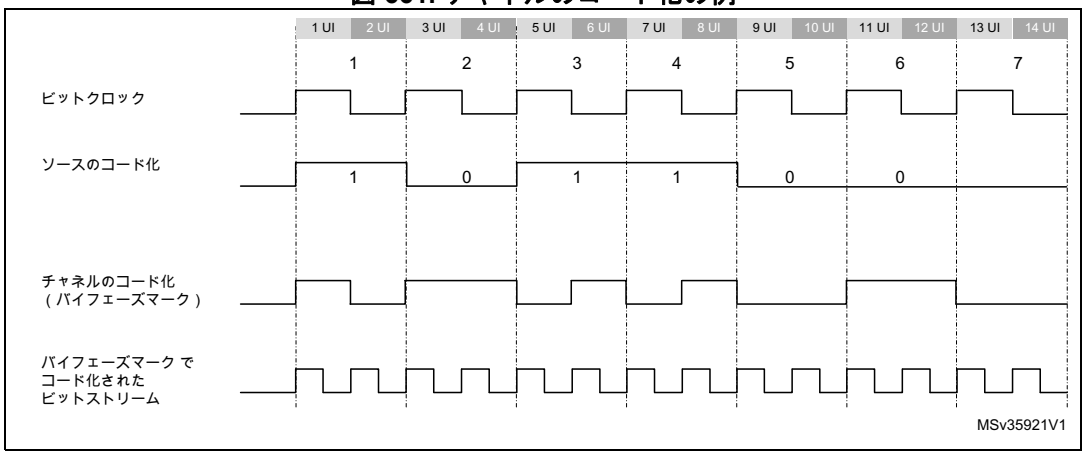
情報ビットのコード化

送信ライン上で DC 成分値を最小にし、データストリームからのクロックリカバリを容易にするために、ビット 4～31 を バイフェーズマークでエンコードします。

送信される各ビットは 2 つの連続したバイナリ状態から成るシンボルで表されます。シンボルの第 1 の状態とその前のシンボルの第 2 の状態は常に異なります。送信されるビットが論理“0”のとき、シンボルの第 2 の状態と第 1 の状態は同じですが、論理“1”の場合はそれらは異なります。このような状態は、IEC-60958 規格では“UI”（ユニットインターバル）呼ばれます。

24 個のデータビットが LSB ファーストで転送されます。

図 661. チャネルのコード化の例



52.3.3 SPDIFRX デコーダ（SPDIFRX_DC）

主要原理

S/PDIF ストリームをデコードするために SPDIFRX で使われる技法は、2 つの連続したエッジ間の時間間隔の測定に基づくものです。S/PDIF ストリームでは次の 3 種類の時間間隔が見られます。

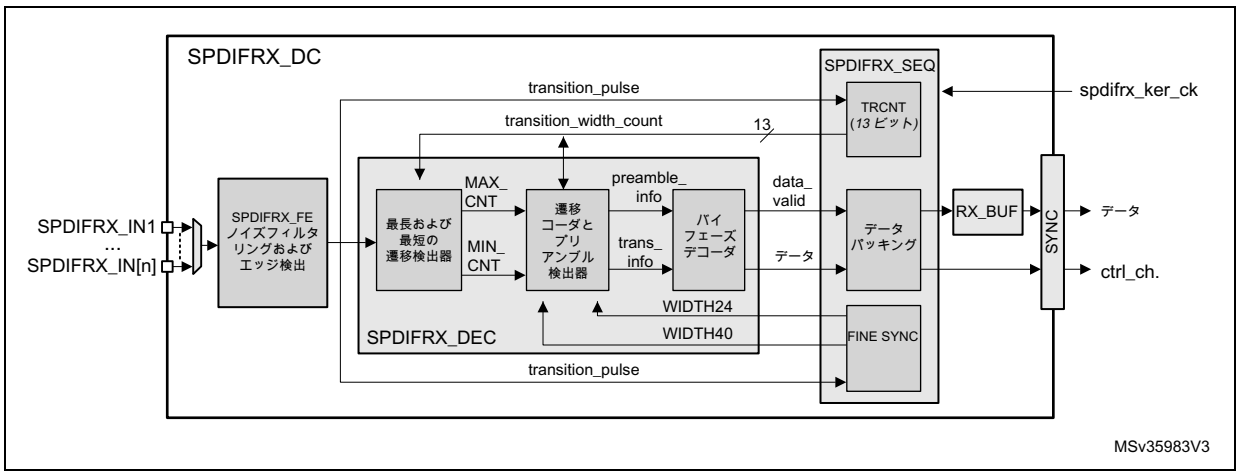
- 長い時間間隔（TL）。継続時間は 3 x UI。プリアンブル時のみ出現。
- ある程度長い時間間隔（TM）。継続時間は 2 x UI 。一部のプリアンブルおよび情報フィールドの両方に出現。
- 短い時間間隔（TS）。継続時間は 1 x UI。一部のプリアンブルおよび情報フィールドの両方に出現。

SPDIFRX_DC ブロックは受信した S/PDIF ストリームをデコードするブロックです。このブロックは次の機能を果たします。

- 受信信号の再サンプリングとフィルタリング
- 時間間隔の概算
- シンボルレートおよび同期の概算
- シリアルデータのデコード、整合性チェック
- ブロックおよびサブフレームプリアンブルの検出
- シンボルレートの連続追跡

図 662 に SPDIFRX デコーダの詳細図を示します。

図 662. SPDIFRX デコーダ

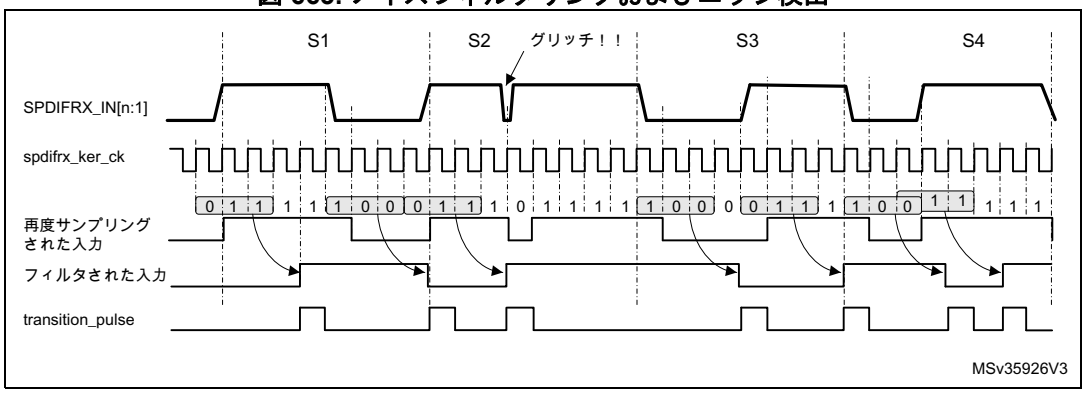


ノイズフィルタリングおよび立ち上がり／立ち下がりエッジ検出

選択された SPDIFRX_IN で受信した S/PDIF 信号は、spdifrx_ker_ck クロック（取得クロック）を使用して再サンプリングされます。スプールを取り消すには、簡単なフィルタリングを適用します。このフィルタリングは、エッジ遷移を検出するステージで行われます。エッジ遷移は次のように検出されます。

- 立ち上がりエッジは、シーケンス 0 とそれに続く 2個の“1”がサンプリングされたときに検出されます。
- 立ち下がりエッジは、シーケンス 1 とそれに続く 2個の“0”がサンプリングされたときに検出されます。
- 立ち上がりエッジの後に立ち下がりエッジシーケンスが予期されます。
- 立ち下がりエッジの後に立ち上がりエッジシーケンスが予期されます。

図 663. ノイズフィルタリングおよびエッジ検出



最長および最短の遷移検出器

最長および最短の遷移検出器のブロックは、2 つの遷移間の最大（MAX_CNT）および最小（MIN_CNT）の継続時間を検出します。TRCNT カウンタを使用して、間隔の継続時間を測定します。カウンタのクロックは spdifrx_ker_ck 信号によって供給されます。1 遷移パルスごとに、カウンタの値が保存され、カウンタはリセットされて再びカウントを開始します。

最大継続時間は通常プリアンブル期間中に確認されます。この最大継続時間は MAX_CNT として送り出されます。最小継続時間は MIN_CNT として送り出されます。

最長および最短遷移の検索は、遷移タイマがタイムアウトになった時点で停止されます。遷移タイマは、70 回の受信信号の遷移後にトリガを生成するウォッチドッグタイマのようなものです。70 回の遷移をカウントすることで、確実にサブフレームより少し長い遅延が発生しますのでご注意ください。

また、2 個のパルス間の間隔が長すぎて TRCNT がオーバーフローした場合、SPDIFRX が停止し、SPDIFRX_SR レジスタの TERR フラグが 1 にセットされます。

遷移コーダとプリアンブル検出器

遷移コーダとプリアンブル検出器のブロックは MAX_CNT および MIN_CNT を受信します。また、TRCNT カウンタから現在の遷移幅も受信します（*図 662*を参照）。このブロックは、現在の遷移幅を TH_{HI} および TH_{LO}と呼ばれる 2 つの異なる閾値と比較することでエンコードします。

- 現在の遷移幅が（TH_{LO} - 1）より小さい場合、受信データはデータビット“1”の半分の部分であり、TS としてコード化されます。
- 現在の遷移幅が（TH_{LO} - 1）より大きくかつ TH_{HI} より小さい場合、受信データはデータビット“0”であり、TM としてコード化されます。
- 現在の遷移幅が（TH_{LO} - 1）より大きい場合、受信データはプリアンブルの長パルスであり、TL としてコード化されます。
- そうでない場合、エラーコードが生成されます（FERR フラグがセットされます）。

閾値 TH_{HI} と TH_{LO} は 2 つの異なる方法で求められます。

ペリフェラルがその最初の同期（「粗い同期」と呼ばれる）を実行している場合、閾値は次式で求められます。

- TH_{LO} = MAX_CNT / 2
- TH_{HI} = MIN_CNT + MAX_CNT / 2

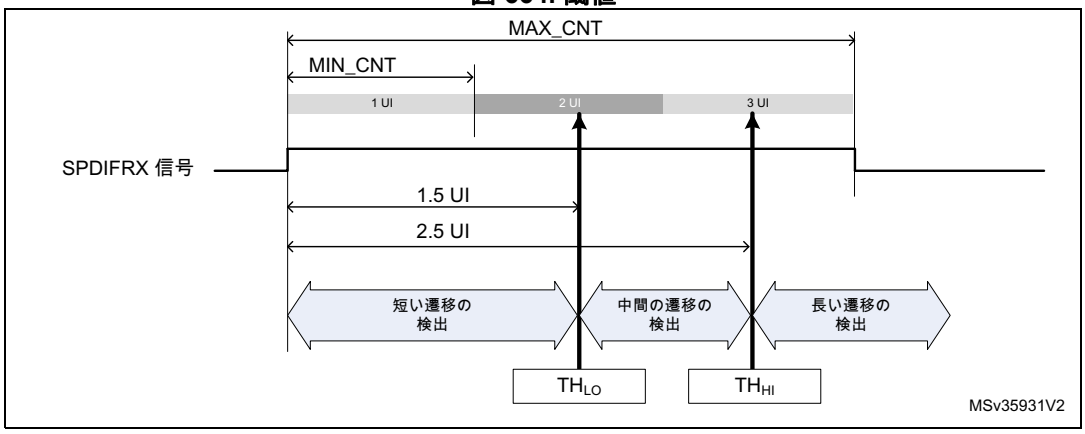
「粗い同期」モードが完了すると、SPDIFRX はより正確な基準を使用して閾値を算出します。SPDIFRX は、24 個のシンボル（WIDTH24）の長さを測定して TH_{LO} を設定し、40 個のシンボル（WIDTH40）の長さを測定して TH_{HI}を設定します。TH_{HI} と TH_{LO} は次のように計算します。

- TH_{LO} = (WIDTH24) / 32
- TH_{HI} = (WIDTH40) / 32

この2 番目の同期フェーズは「細かい同期」と呼ばれます。詳細については、*図 666*を参照してください。

下図に示すように、TH_{LO} は 1.5 UI に等しく、TH_{HI} は 2.5 UI に等しいのが理想的です。

図 664. 閾値



プリアンブル検出器は、特定のシーケンスの 4 個の連続した遷移をチェックして、それらがプリアンブルの一部を形成しているかどうかを判断します。TRANS0、TRANS1、TRANS2、TRANS3 が、前述のようにエンコードされた 4 個の連続した遷移を表しているとしましょう。*表 409* に、1つのプリアンブルを形成するこれら 4 個の遷移の値を示します。このパターンが検出されないと、これらの遷移はサブフレーム内のデータの一部を示し、バイフェーズデコーダがそれらをデコードするでしょう。

表 409. プリアンブルの遷移シーケンス

プリアンブルタイプ	バイフェーズデータパターン	TRANS3	TRANS2	TRANS1	TRANS0
プリアンブル B	11101000	TL	TS	TS	TL
プリアンブル M	11100010	TL	TL	TS	TS
プリアンブル W	11100100	TL	TM	TS	TM

バイフェーズデコーダ

バイフェーズデコーダは、**遷移コーダとプリアンブル検出器**のブロックが提供する遷移情報を使用して、入力バイフェーズマークが付いたデータストリームをデコードします。まず、プリアンブル検出情報を待ちます。プリアンブルが検出されたら、バイフェーズデコーダは次の遷移情報をデコードします。

- 受信した遷移情報が TM の場合、“0”としてデコードされます。
- 2 つの連続した TS は“1”としてデコードされます。
- その他の遷移シーケンスはエラー信号を生成します（FERR を 1 にセット）。

このように 28 個のデータビットをデコードし終えたら、バイフェーズデコーダは次のプリアンブルデータを探します。新たなプリアンブルが予想したものと違っていた場合、このブロックはエラー信号を生成します（FERR に 1 をセット）。エラーフラグの詳細については、*セクション 52.3.9：受信エラー*を参照してください。

データパッキング

このブロックは IEC-60958 フレームおよびブロックをデコードする役目があります。また、RX_BUF または SPDIFRX_CSR レジスタへの書き込みも処理します。

52.3.4 クロック偏差に対する SPDIFRX の許容誤差

クロック偏差に対する SPDIFRX の許容誤差は 1 ビットスロット内のサンプリングクロックサイクル数で決まります。spdifrx_ker_ck が高速であればあるほど、その受信はより安定したものになります。spdifrx_ker_ck 周波数とシンボルレートの比率は最低でも 11 とならなければなりません。

次の（少なくとも）2 種類の現象により、受信品質が落ちる可能性があります。

- 2 個の連続した遷移間の遷移長さの差を反映したサイクル間ジッタ。
- サイクル間ジッタの累積的影響を反映したロングタームジッタ。これは低周波数シンボルの変調とみなされます。

52.3.5 SPDIFRX の同期

同期フェーズは、SPDIFRXEN を 0b01 または 0b11 にセットした時点で開始されます。*図 665* に同期プロセスを示します。

SPDIFRX_CR レジスタの WFA ビットが 1 にセットされている場合、ペリフェラルは、同期プロセスを開始する前にまず選択された SPDIFRX_IN ラインでアクティビティを検出する必要があります。選択された SPDIFRX_IN ライン上の 4 個の遷移を検出することで、アクティビティを検出します。ペリフェラルはこの遷移状態が検出されるまでこのステートにとどまります。この機能は粗い同期のモードに入ってしまった時に、同期エラーの誤検出を防ぐために特に有効です。詳細については、*セクション 52.4：プログラミング手順* を参照してください。
ユーザは引き続き SPDIFRXEN を 0 にセットすることで、SPDIFRX を STATE_IDLE に設定することができます。WFA が 0 にセットされている場合、ペリフェラルはアクティビティをチェックせずに粗い同期モードを開始します。

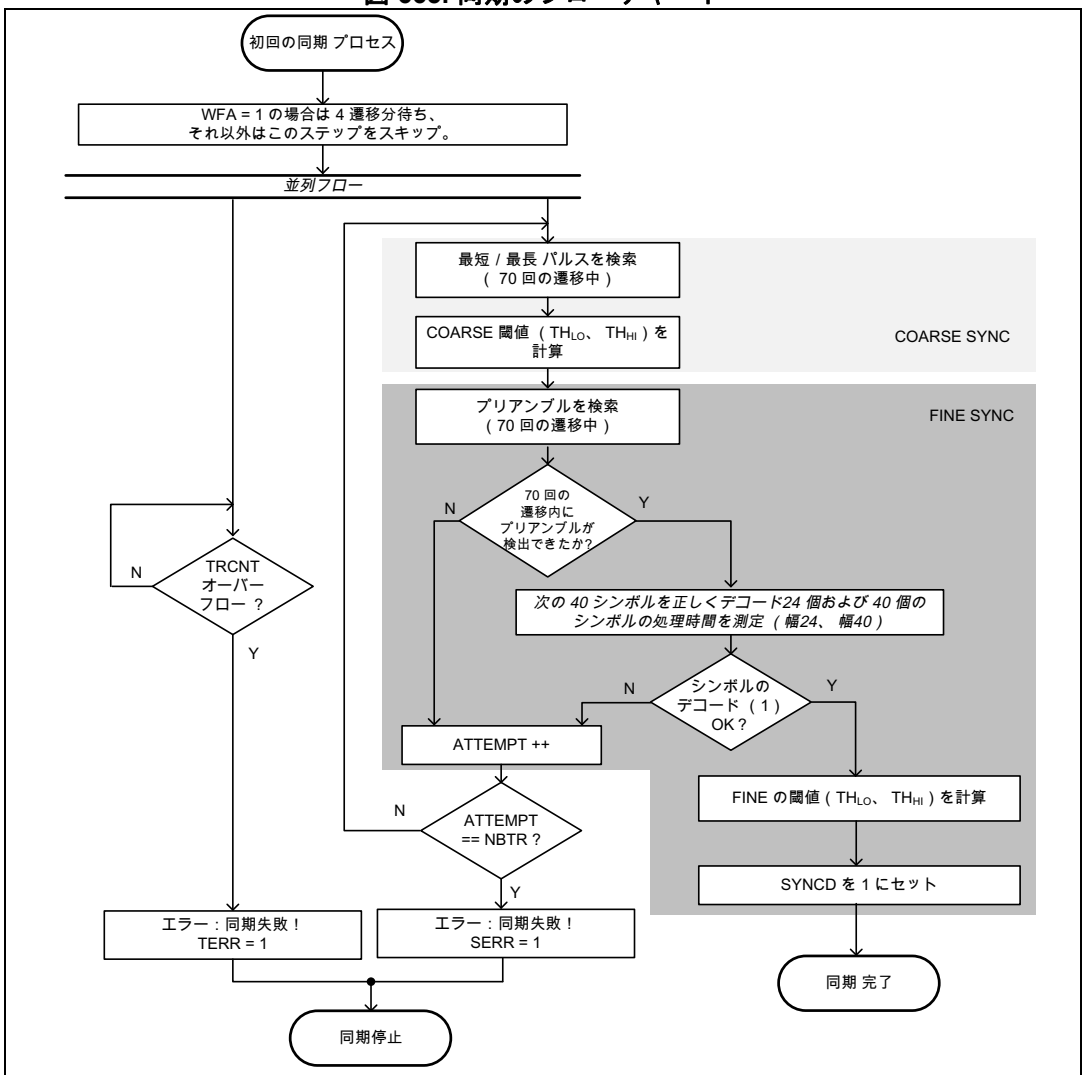
次のステップでは、細かい同期（FINE SYNC)モードを実行するために、閾値（COARSE SYNC）の最初の概算を行います。SPDIFRX ラインの障害により、初回のプロセスが実行されないという事態が発生する可能性があります。このような事態を回避するため、ユーザは SERR エラーフラグをセットする前に許容リトライ回数（NBTR）をプログラムすることができます。
SPDIFRX が 24 個および 40 個の連続したシンボルの継続時間を正確に測定できたら、細かい同期(FINE SYNC)モード は完了し、閾値が更新され、SYNCD フラグが 1 にセットされます。詳細については、*セクション ：遷移コーダとプリアンブル検出器*を参照してください。

次の 2 種類のエラーが検出されます。

- TRCNT がオーバーフローしました。このエラーは通常、有効な S/PDIF ストリームが入力ラインにないことを意味します。このオーバーフローは TERR フラグで示されます。
- 再試行回数がプログラムされた値に達しました。このエラーは、S/PDIF 信号に大きなジッタが存在することを意味します。このエラーは SERR フラグで示されます。

最初の FINE SYNC が完了すると、次のプリアンブル B が検出された時点でチャネルステータス（C）およびユーザデータ（U）の受信が始まります（*図 669*を参照）。次に、ユーザは SPDIFRX_CSR レジスタにより IEC-60958 C および U ビットを読み出すことができます。この情報を基に、ユーザは DRFMT および RXSTEO の適切な設定値を選択できます。たとえば、現在のオーディオストリームによるエンコードされたデータの転送を検出した場合、データ受信を開始する前に、RXSTEO に 0、DRFMT に 0b10 をセットすることができます。DRFMT および RXSTEO は SPDIFRXEN = 0b11 の場合は修正できませんので注意してください。SPDIFRXEN がすでに 0b11 である場合、これらのフィールドへの書き込みは無視されます。ただし、SPDIFRXEN を 0b11 にせよという同じ書き込み命令でこれらのフィールドを変更することはできます。
その後、SPDIFRX は SPDIFRXEN = 0b11 となり、プリアンブル B が検出されるのを待ってから、オーディオサンプルの保存を開始します。

図 665. 同期のフローチャート



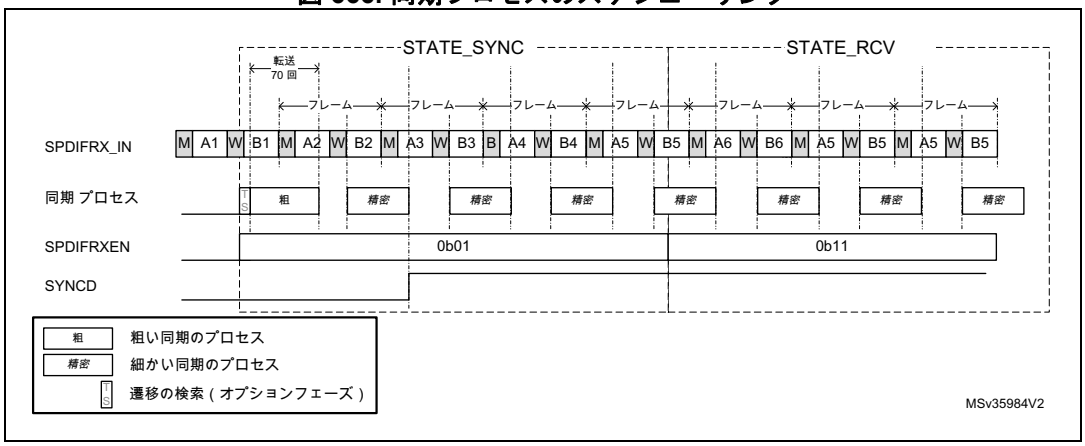
(1) - デコード処理は OK（ シンボルが適切にデコードされている場合 ）と判断され、 予期した 位置でプリアンブルが発生します。

MSv35932V1

TRCNT オーバーフローに関する詳細情報は、*フレーム構造と同期エラー* を参照してください。

FINE SYNC プロセスはフレームごとに再トリガされます。この目的は、*図 666* に示す閾値を更新して、S/PDIF 同期を連続的に追跡することです。

図 666. 同期プロセスのスケジューリング



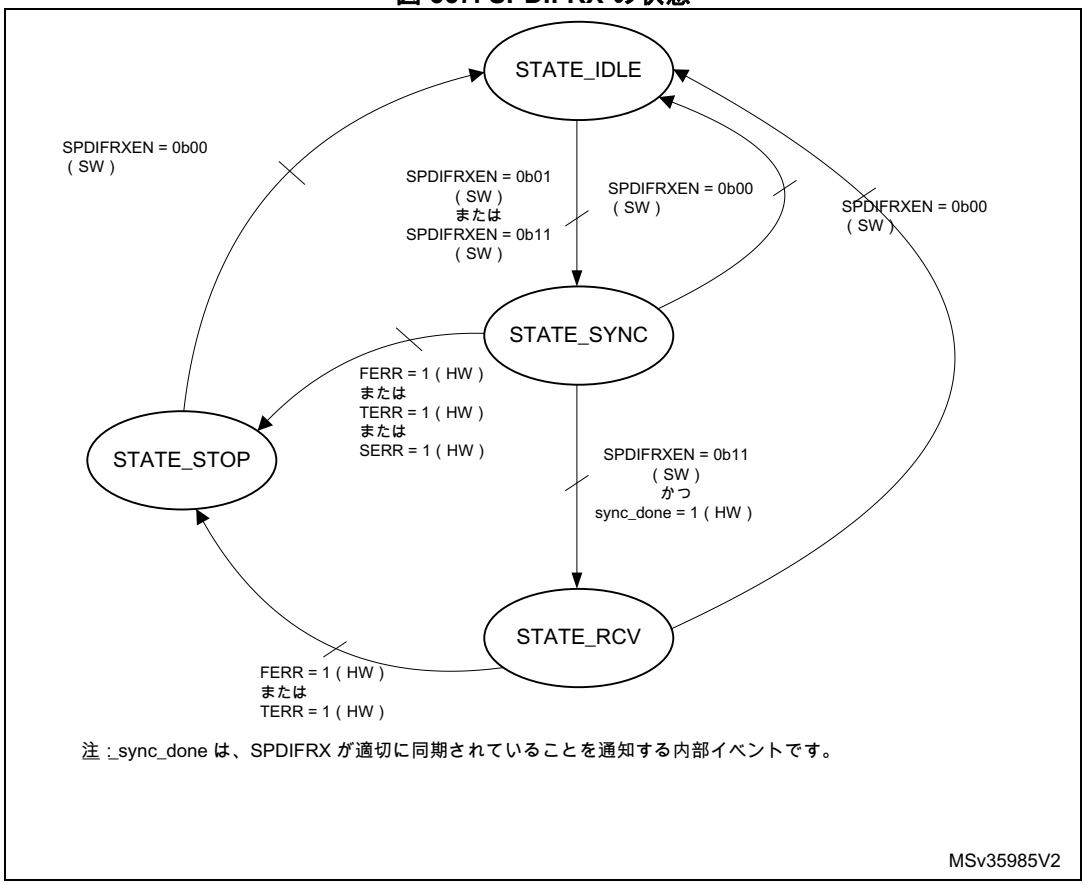
52.3.6 SPDIFRX 処理

ソフトウェアで SPDIFRXEN フィールドを使用して SPDIFRX の状態を制御することができます。SPDIFRX は次の状態のいずれかになり得ます。

- STATE_IDLE：
  ペリフェラルが無効化され、spdifrx_ker_ck ドメインがリセットされた状態です。spdifrx_pclk ドメインは機能しています。
- STATE_SYNC：
  ペリフェラルはストリームに同期され、閾値が定期的に更新され、DMA の割り込みによりユーザおよびチャネルステータスを読み出すことができる状態です。オーディオサンプルは受信バッファに提供されません。
- STATE_RCV：
  ペリフェラルはストリームに同期され、閾値が定期的に更新され、割り込みまたは DMA チャネルを通じてユーザ、チャネルステータス、オーディオサンプルを読み出すことができる状態です。SPDIFRXEN が 0b11 の場合、SPDIFRX はプリアンブル B を待ってから、オーディオサンプルの保存を開始します。
- STOP_STATE：
  ペリフェラルはもはや同期されず、ユーザ、チャネルステータス、オーディオサンプルの受信が停止した状態です。ソフトウェアによる SPDIFRX の再開が待たれます。

図 667 に、起こり得る SPDIFRX の状態、およびある状態から別の状態への遷移方法を示します。ソフトウェア制御されているビットの後ろに“(SW)”、IP 制御されているビットの後ろに“(HW)”を付けます。

図 667. SPDIFRX の状態



SPDIFRX が STATE_IDLE である場合

- ソフトウェアで SPDIFRXEN を 0b01 または 0b11 にセットすることで、STATE_SYNC へ遷移できます。

SPDIFRX が STATE_SYNC である場合

- 同期に失敗した場合、または受信データが適切にデコードされておらす、再同期なしで復帰する可能性はない場合（FERR、SERR、または TERR = 1）、SPDIFRX は STATE_STOP になり、ソフトウェアによる確認応答を待ちます。
- 同期フェーズが完了したとき、SPDIFRXEN = 0b01 であればペリフェラルはこの状態にとどまります。
- ソフトウェアにより SPDIFRXEN を 0 にセットできるときはいつでも、SPDIFRX はすぐに STATE_IDLE に戻ります。DMA 転送中は正常に完了します。
- SPDIFRXEN = 0b11 かつ SYNCD = 1 のとき、SPDIFRX は STATE_RCV になります。

SPDIFRX が STATE_RCV である場合

- 受信データが適切にデコードされておらず、再同期なしで復帰する可能性はない場合（FERR、SERR、または TERR = 1）、SPDIFRX は STATE_STOP になり、ソフトウェアによる確認応答を待ちます。
- ソフトウェアにより SPDIFRXEN を 0 にセットできるときはいつでも、SPDIFRX はすぐに STATE_IDLE に戻ります。DMA 転送中は正常に完了します。

SPDIFRX が STATE_STOP である場合

- SPDIFRX は受信および同期を停止し、エラーフラグをクリアするために、ソフトウェアによって SPDIFRXEN ビットが 0 にセットされるのを待ちます。

SPDIFRXEN が 0 にセットされている場合、IP は無効化されます。これは、ステートマシンがすべてリセットされ、RX_BUF が一掃されることを意味します。FERR、SERR、および TERR フラグもリセットされますのでご注意ください。

52.3.7 データ受信管理

SPDIFRX はオーディオサンプル受信のためにダブルバッファを提供します。32 ビットのバッファが、spdifrx_ker_ck クロックドメイン（RX_BUF）内およびSPDIFRX_DR レジスタに配置されます。RX_BUF に格納されている有効データは、SPDIFRX_DR がエンプティであればすぐに SPDIFRX_DR へ転送されます。

RX_BUF に格納されている有効データは、次の 2 つの条件が満たされると SPDIFRX_DR へ転送されます。

- パリティビット（P）と次のプリアンブルの間の遷移が検出されます（ワードが完全に受信されたことを意味します）。
- SPDIFRX_DR はエンプティです。

2 ワードバッファを備えることで、遅延の制約に対する柔軟性が向上します。

最大許容遅延を求める計算式：$T_{SAMPLE} - 2T_{PCLK} - 2T_{spdifrx_ker_ck}$

ここで T_{SAMPLE} は受信したステレオオーディオサンプルのオーディオサンプリングレートであり、T_{PCLK} は spdifrx_pclk クロックの周期、$T_{spdifrx_ker_ck}$ は spdifrx_ker_ck クロックの周期です。

SPDIFRX により、DMA（spdifrx_dat_dma と spdifrx_ctrl_dma）または割り込みによる オーディオサンプルのメモリ転送のいずれかを使用することができます。DMA を選択することをお勧めします。詳細については、*セクション 52.3.12：DMA インタフェース* を参照してください。

SPDIFRX では、受信データを処理する方法がいくつかあります。ユーザは、制御情報およびやオーディオサンプルに関して個別のフローを得るか、またはそれらすべてを 1 つにまとめることができます。

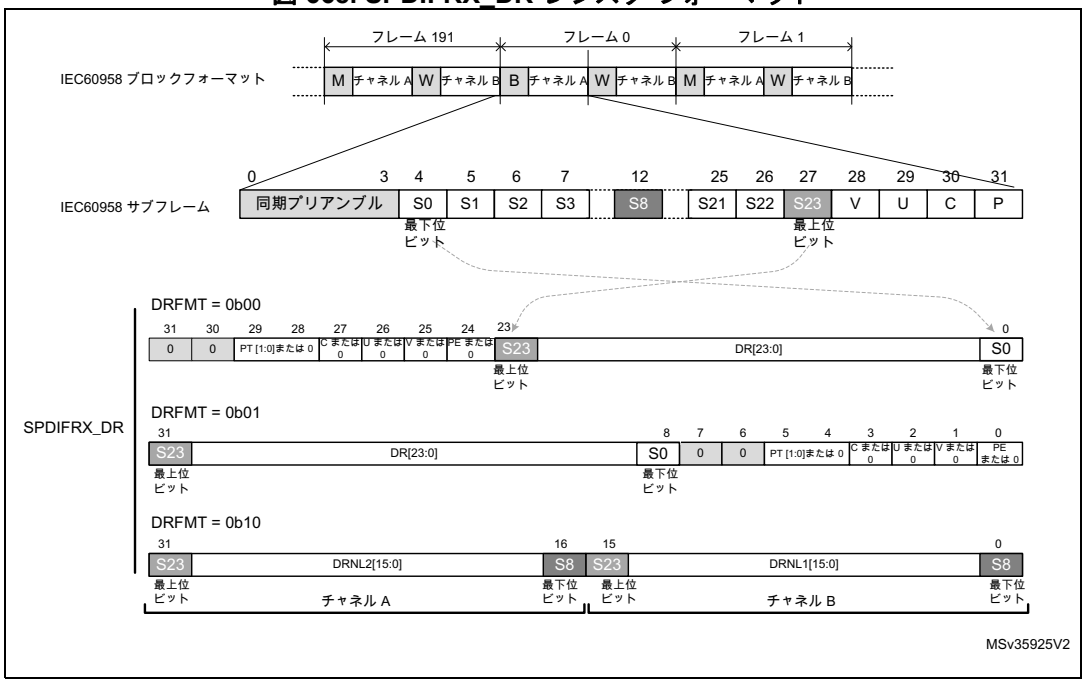
サブフレームごとに、データ受信レジスタの SPDIFRX_DR には 24 個の データビットと、オプションの V、U、C、PE ステータスビット、および PT が格納されています（*データの混合と制御フロー* を参照）。

PE ビットはパリティエラービットを表し、デコードされたサブフレームでパリティエラーが検出されると、このビットには 1 がセットされます。
PT フィールドにはプリアンブルタイプ（B、M、または W）が表示されます。

V、U、および C は S/PDIF インタフェースから受信した値を直接コピーしたものです。

DRFMT ビットにより、*図 668* に示す 3 種類のオーディオフォーマットから選択することができます。

図 668. SPDIFRX_DR レジスタ フォーマット



DRFMT を 0b00 または 0b01 にセットすることで、データを SPDIFRX_DR レジスタ内に右詰めまたは左詰めで格納することができます。ステータス情報は、ソフトウェアで希望する情報の処理方法に応じて、有効化するか、強制的に 0 に設定するかを選択できます。

DRFMT = 0b10 のときに与えられるフォーマットは、サブフレーム当たり 16 ビットしか使われないことからノンリニアモードを対象としています。このフォーマットを使うことにより、2 個の連続したサブフレームのデータを SPDIFRX_DR に格納し、メモリ半分にします。RXSTEO = 1 である場合、ミスアライメントのリスクは存在しない（ChA からのデータは常に SPDIFRX_DR[31:16] に格納される）ことに注意してください。RXSTEO = 0 のときに起こり得るミスアラインメントはオーバーランの状況です。その場合、SPDIFRX_DR[31:16] には常に一番古い値が含まれ、SPDIFRX_DR[15:0] にはより新しい値が含まれます（*図 670*を参照）。

このフォーマットでは、ステータス情報をデータと混在させることはできませんが、ユーザはその情報を SPDIFRX_CSR レジスタで取得し、専用の DMA チャネルまたは割り込みを利用した転送を使用して情報をメモリに転送することができます（*セクション 52.3.8：専用制御フロー*を参照）。

データの混合と制御フロー

ユーザはこのモードの使用を選択して、制御フローを十分な柔軟性を持って処理することができます。また、データレジスタ（SPDIFRX_DR）内のどのフィールドを確保すべきかも選択できます。

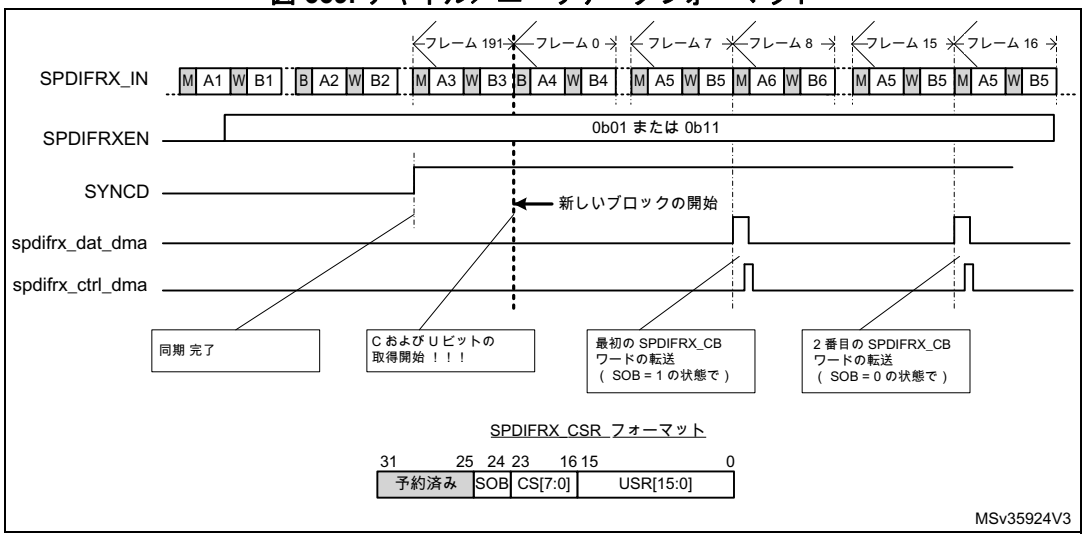
- PMSK ビット = 1 のとき、パリティエラー情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_DR にコピーされます。
- VMSK ビット = 1 のとき、有効性情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_DR にコピーされます。
- CUMSK ビット = 1 のとき、チャネルステータスおよびユーザデータ情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_DR にコピーされます。
- PTMSK ビット = 1 のとき、プリアンブルタイプはマスクされます（0 にセットされる）。そうしないと、SPDIFRX_DR にコピーされます。

52.3.8 専用制御フロー

SPDIFRX により、専用 DMA チャネルを使用してユーザデータとチャネルステータス情報の両方をキャッチすることができます。この機能を使用することで、SPDIFRX はチャネルステータス情報とユーザ情報を継続的に取得することができます。取得は IEC 60958 ブロックの開始時に開始されます。このパスを制御するため使用可能な 2 つのフィールドは、CBDMAEN と SPDIFRXEN です。SPDIFRXEN が 0b01 または 0x11 にセットされている場合、取得は同期フェーズの完了後に開始されます。8 個のチャネルステータスビットおよび 16 個のユーザデータビットが受信されると、それらはパックされて SPDIFRX_CSR レジスタに格納されます。CBDMAEN ビットが 1 にセットされると、DMA リクエストがトリガされます（*図 669*を参照）。

CS[0] が新しいブロックの先頭ビットに対応する場合、SOB ビットは 1 にセットされます。*セクション 52.5.8：チャネルステータスレジスタ（SPDIFRX_CSR）*を参照してください。CHSEL ビットを使用して、チャネル A または B のどちらからのチャネルステータス情報（C）をユーザが選択できます。

図 669. チャネル／ユーザデータフォーマット



注： ***ブロックの最初の開始が検出されると（プリアンブル B）、SPDIFRX は 8 フレームおきにプリアンブルタイプをチェックします。***

注： ***SPDIFRX_DR レジスタのオーバーランエラーはこのパスには影響を与えません。***

52.3.9 受信エラー

フレーム構造と同期エラー

SPDIFRX は、次のいずれかの条件が発生した場合にエラーを検出します。

- 以下の条件では、FERR ビットは 1 にセットされます。
  - 28 個の情報ビットのそれぞれについて、1 つのシンボル遷移シーケンスが正しくない場合。たとえば、ショートパルスがペアでグループ化されていない場合。
  - プリアンブルが予期せぬ場所で発生した場合、または予期したプリアンブルが受信されない場合。
- 同期に失敗すると SERR ビットがセットされます。これは、リトライ回数がプログラムされた値を超えたためです。
- 2 個の遷移間の幅を概算するために使われるカウンタがオーバーフローすると TERR ビットがセットされます（TRCNT）。

spdifrx_ker_ck クロックの 8192 周期中に一度も遷移が検出されない場合にオーバーフローが発生します。オーバーフローは最大でも 11.6 フレームの時間間隔を表します。

上記のフラグのうちのいずれかが 1 になると、選択された SPDIFRX_IN のトラフィックは無視され、ISPDIFRX_CR レジスタの IFEIE ビットがセットされていれば割り込みが生成されます。

上記のエラーのいずれかが発生した場合の通常の手順は次の通りです。

- SPDIFRXEN を 0 にセットして、エラーフラグをクリアします。
- SPDIFRXEN を 0b01 または 0b11 にセットして、IP を再開します。

詳細については、*図 667*を参照してください。

パリティエラー

各サブフレームでは、28 個の情報ビット内に偶数個の“0”および“1”が入ることが予想されます。そうでない場合、SPDIFRX_SR レジスタの パリティエラービット（PERR）がセットされ、SPDIFRX_CR レジスタのパリティ割り込みイネーブルビット PERRIE がセットされている場合は割り込みが生成されます。割り込みがペンディング状態のままでも、データの受信は一時中止することなく、SPDIFRX は SPDIFRX_DR へデータを送り出し続けます。

割り込みの確認応答をするには、PERRCF ビットによって PERR フラグをクリアします。

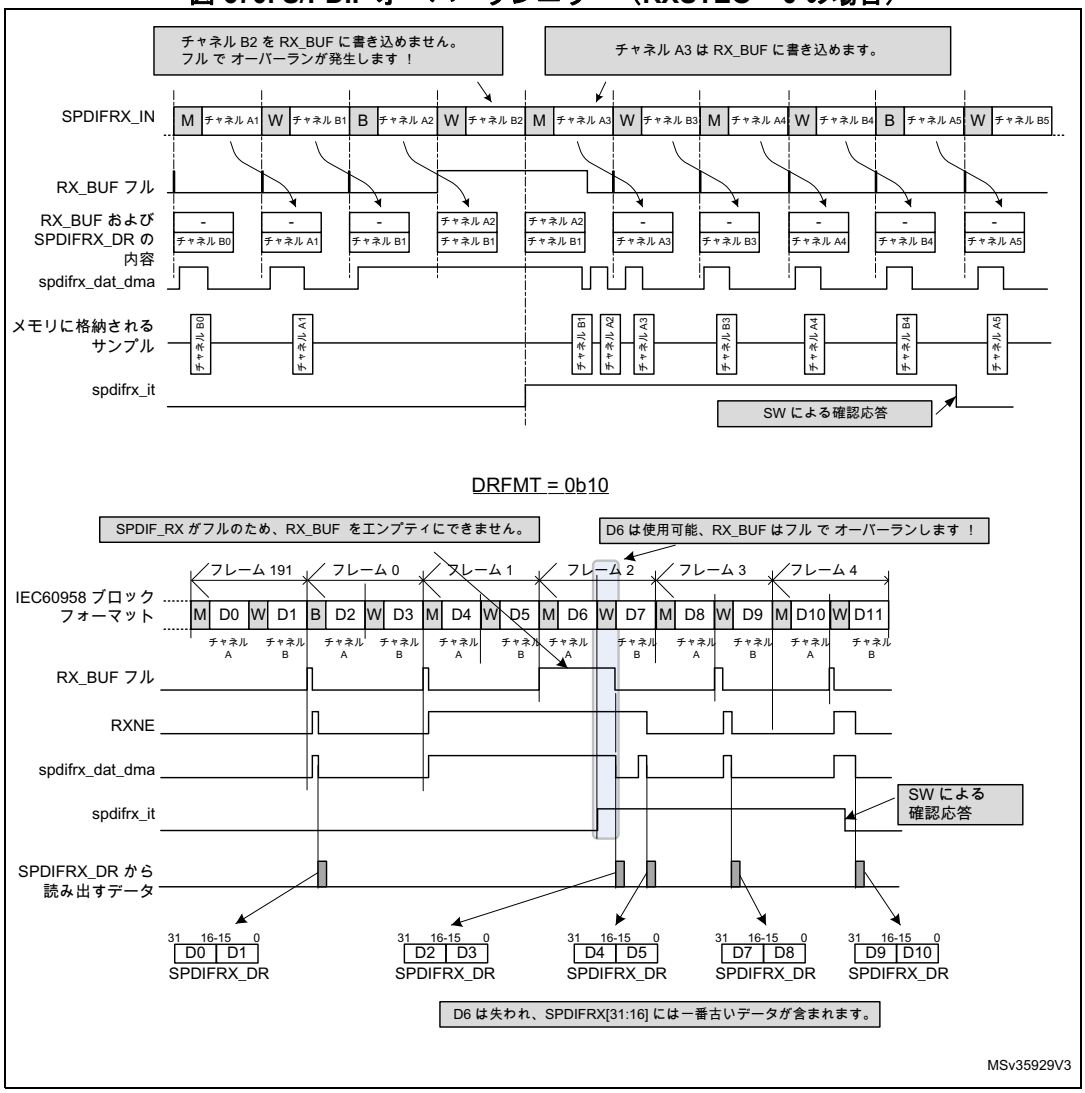
SPDIFRX_DR レジスタで読み出されたデータと PERR ビットの値の間の一貫性をソフトウェアで保証したい場合は、PMSK ビットを 0 にセットする必要があります。

オーバーランエラー

SPDIFRX_DR と RX_BUF の両方がフルの場合、SPDIFRX_DC が RX_BUF に新しいサンプルを書き込む必要がある場合、この新しいサンプルは捨てられされ、オーバーラン状態が開始されます。SPDIFRX_SR レジスタの OVR オーバーランエラーフラグがセットされ、SPDIFRX_CR レジスタの OVRIE ビットがセットされている場合は割り込みが生成されます。

RXSTEO ビットが 0 にセットされている場合、RX_BUF がエンプティになるとすぐに、IP は OVR フラグがペンディング中であっても次の受信データを格納します。この主な目的は、サンプルの損失をできるだけ少なくすることです。なお、動作は DRFMT 設定値に関係なく同様です。*図 670*を参照してください。

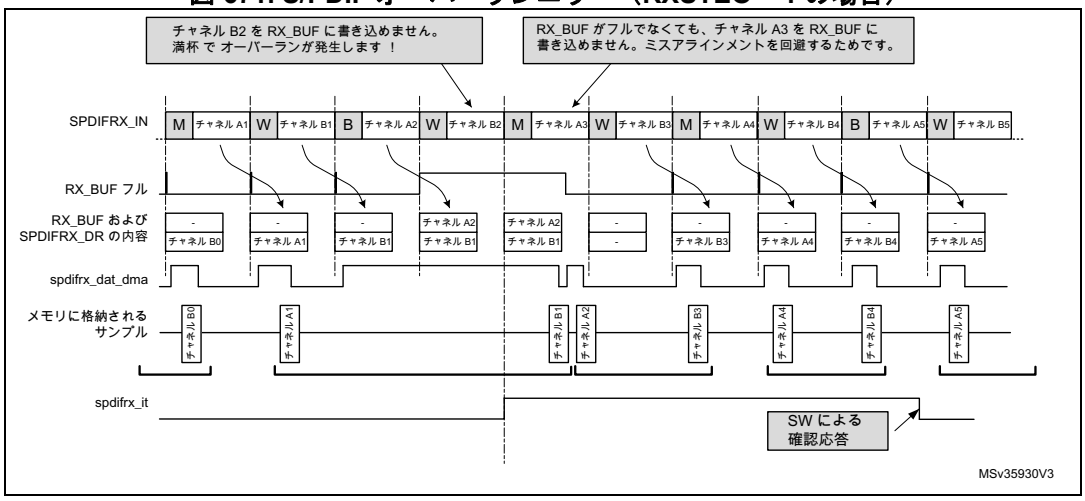
図 670. S/PDIF オーバーランエラー（RXSTEO = 0 の場合）



RXSTEO ビットが 1 にセットされている状態はステレオデータが転送されることを意味しますので、SPDIFRX は左チャネルと右チャネルの間のミスアラインメントを避ける必要があります。したがって、ペリフェラルは、たとえ RX_BUF 内に余裕があったとしても、ミスアラインメントを避けるために 2 番目のサンプルを捨てなければなりません。さらに、OVR フラグがまだペンディング状態であっても、通常は受信サンプルを RX_BUF に書き込むことができます。*図 671* を参照してください。

OVRCF ビットを 1 にセットすることによって、OVR フラグはソフトウェアによってクリアされます。

図 671. S/PDIF オーバーランエラー（RXSTEO = 1 の場合）



52.3.10 クロック供給の方法

SPDIFRX ブロックには次の 2 種類のクロックが必要です。

- APB クロック（spdifrx_pclk）: レジスタインタフェースに使われます。
- spdifrx_ker_ck : 主に SPDIFRX_DC 部分によって使われます。これらのクロックは非同期クロックになりますので、両方のクロックドメインを跨ぐすべての信号は再同期されます（<776/>の SYNC ブロック*図 657*）。

受信ストリーム S/PDIF を適切にデコードするために、SPDIFRX_DC は最大シンボルレートより最低でも 11 倍高速、またはオーディオサンプリングレートより 704 倍高速のクロックで受信データを再サンプリングします。たとえば、ユーザが最大 12.288 MHz のシンボルレートを受信したい場合、サンプリングレートは最低でも 135.2 MHz となります。SPDIFRX_DC が使用するクロックは、spdifrx_ker_ck です。

spdifrx_pclk の周波数はシンボルレート以上でなければなりません。

表 410. spdifrx_ker_ck 最小周波数とオーディオサンプリングレート

シンボルレート	spdifrx_ker_ck 最小周波数	コメント
3.072 MHz	33.8 MHz	48 kHz ストリーム用
6.144 MHz	67.6 MHz	96 kHz ストリーム用
12.288 MHz	135.2 MHz	192 kHz ストリーム用

52.3.11 シンボルクロック生成

SPDIFRX ブロックからは、**spdifrx_symb_ck** という名前の信号に対するシンボルクロックが供給され、SAI や SPI／I2S などの他のオーディオデバイスに対するリファレンスカーネルクロックとして用いることができます。これはSPDIFXをI2Sブリッジ機能に使用することが出来ます。

このシンボルクロックは、WIDTH24 と WIDTH40 の値とシンボル境界を使用して構築されます。

- サブフレーム同期プリアンブルの受信中に、シンボルクロックの立ち下がりエッジと立ち上がりエッジは、WIDTH24 と WIDTH40 の値から構築されます。SPDIFRX が STATE_STOP または STATE_IDLE のときには、WIDTH24 と WIDTH40 はシンボルクロックの生成にも使用されることに注意してください。詳細は、*表 411* を参照してください。
- サブフレームペイロードの受信中に、SPDIFRX は、シンボル境界を用いて立ち上がりエッジを生成し、WIDTH24 と WIDTH40 の値を用いて立ち下がりエッジを生成します。

シンボルクロックのデューティーサイクルは、サブフレームペイロードの受信中は 50% に近い値です。ただし、SPDIFRX が WIDTH24 と WIDTH40 で生成されるシンボルクロックからシンボルクロック境界によって生成されるクロックに遷移したり、その反対に遷移したりすると、デューティーサイクルは変化することがあります。

シンボルクロックには、主として次の原因によって大きなジッタが生じます。

- spdifrx_ker_ck クロックによる S/PDIF の再サンプリング
- シンボルクロック生成モードの遷移

そのため、シンボルクロックが A/D コンバータや D/A コンバータのリファレンスクロックとして使用される場合には、アプリケーションは品質の劣化を考慮する必要があります。

シンボルクロックの生成は、CKSEN ビットによって制御されます。CKSEN = ‘1’の場合には、SPDIFRX が最初の精密同期が正常に完了（SYNCD = 1）して、選択された SPDIFRX 入力から正しいデータを受信しているときに、クロックシンボルが生成されます。

SPDIFRX が STATE_STOP または STATE_IDLE になった場合、CKSBKPEN ビット = ‘0’であるとシンボルクロックはゲートされます。CKSBKPEN = ‘1’であれば、SPDIFRX が正しく同期されているならば（WIDTH24 と WIDTH40 の有効な値が存在）、バックアップシンボルクロックはそのまま生成されます。*表 411*に、シンボルクロックの生成を制御する条件の詳細が示されています。

表 411. spdifrx_symb_ck 生成条件

SPDIFRX の状態と条件	CKSEN	CKSBKPEN	spdifrx_symb_ck state
あらゆる状態	0	X	無効
– SPDIFRX が STATE_SYNC であり、精密同期が正常に完了（SYNCD = '1'）、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力から有効データを受信	1	0	有効
– SPDIFRX が STATE_IDLE、または – SPDIFRX が STATE_STOP、または – SPDIFRX が精密同期を未完了（進行中） – SPDIFRX が STATE_RCV であるが、選択された SPDIFRX 入力にデータ（遷移）が検出されない		0	無効

表 411. spdifrx_symb_ck 生成条件（続き）

SPDIFRX の状態と条件	CKSEN	CKSBKPEN	spdifrx_symb_ck state
– SPDIFRX が STATE_IDLE であるが、WIDTH40 と WIDTH24 の値が有効、または – SPDIFRX が STATE_SYNC であり、精密同期が正常に完了（SYNCD = '1'）、または – SPDIFRX が STATE_SYNC であり、進行中の精密同期は未完了であるが、WIDTH40 と WIDTH24 には前回同期の有効な値が含まれている、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力から有効データを受信、または – SPDIFRX が STATE_STOP であるが、WIDTH40 と WIDTH24 の値が有効	1	1	有効
– SPDIFRX が IDLE で WIDTH40 と WIDTH24 の値が無効、または – SPDIFRX が STOP で WIDTH40 と WIDTH24 の値が無効（SERR = '1'）、または – SPDIFRX が STATE_SYNC で WIDTH40 と WIDTH24 の値が無効であり、信号中の精密同期が未完了、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力に遷移が検出されない			無効

フラグ SERR が'1'にセットされている場合、同期が行われていないため、シンボルクロックとバックアップクロックのどちらも生成できないことに注意してください。

CKSEN と CKSBKPEN がどちらも'1'にセットされている場合、SPDIFRX が STATE_SYNC または STATE_RCV から STATE_STOP または STATE_IDLE に切り換わったときに、シンボルクロックの遷移がいくつか失われることに注意してください。

CKSEN ビットと CKSBKPEN ビットは*制御レジスタ（SPDIFRX_CR）*の中にあります。

52.3.12 DMA インタフェース

SPDIFRX インタフェースは、DMA を使用して通信できます。

注： ***DMA コントローラの有効性については製品仕様を参照してください。***

SPDIFRX は 2 つの独立した DMA チャネルを提供します。

- データ転送専用の DMA チャネル
- チャネルステータスおよびユーザデータの転送専用の DMA チャネル

データ受信用のDMA モードを有効にするには、SPDIFRX_CR レジスタの RXDMAEN ビットをセットします。この場合、SPDIFRX_DR がエンプティでなくなるとすぐに、SPDIFRX インタフェースは DMA に対して転送リクエストを送信します。DMA は、CPU の介入なしに SPDIFRX_DR レジスタによって受信したデータを読み出します。

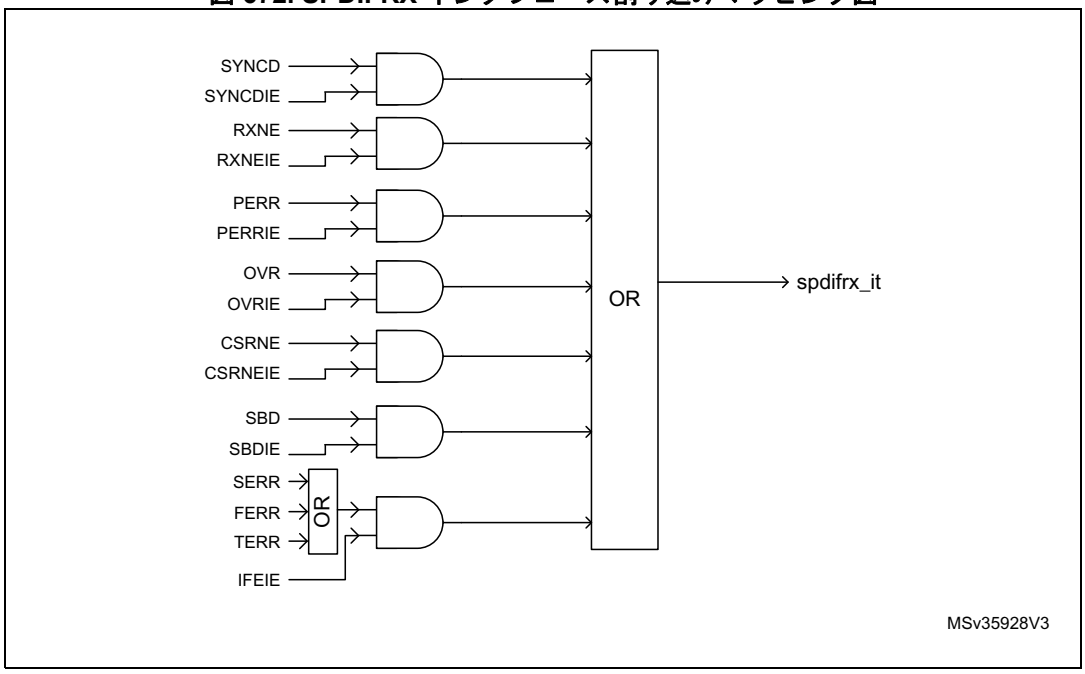
制御データの DMA の使用については、*セクション 52.3.8：専用制御フロー*を参照してください。

52.3.13 割り込みの生成

割り込みラインは次の間で共有されます。

- データフローの受信イベント（RXNE）
- 制御フローの受信イベント（CSRNE）
- データ破損検出（PERR）
- 転送フロー割り込み（OVR）
- フレーム構造と同期エラー（SERR、TERR、FERR）
- 新しいブロック割り込みの開始（SBD）
- 同期完了（SYNCD）

図 672. SPDIFRX インタフェース割り込みマッピング図



割り込みソースのクリア

- RXNE は、SPDIFRX_DR レジスタが読み出されたときにクリアされます。
- CSRNE は、SPDIFRX_CSR レジスタが読み出されたときにクリアされます。
- FERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- SERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- TERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- それ以外は SPDIFRX_IFCR レジスタによってクリアされます。

注： ***SBD イベントは、SPDIFRX が入力ストリームに同期される場合のみ起こり得ます。（SYNCD = 1）プリアンブル B を含むサブフレームがオーバーランにより失われた場合、SBD フラグの動作は保証されません。***

52.3.14 レジスタの保護

SPDIFRX ブロックは、制御レジスタの誤使用を回避するために、何らかのハードウェア保護を内蔵しています。下表に、SPDIFRX の状態に応じたビットフィールドの特性を示します。

表 412. ビットフィールドの特性と SPDIFRX の状態

レジスタ	フィールド	SPDIFRXEN		
		0b00（STATE_IDLE）	0b01（STATE_SYNC）	0b11（STATE_RCV）
SPDIFRX_CR	INSEL	rw	r	r
	WFA	rw	r	r
	NBTR	rw	r	r
	CHSEL	rw	r	r
	CBDMAEN	rw	rw	rw
	PTMSK	rw	rw	rw
	CUMSK	rw	rw	rw
	VMSK	rw	rw	rw
	PMSK	rw	rw	rw
	DRFMT	rw	rw	r
	RXSTEO	rw	rw	r
	RXDMAEN	rw	rw	rw
SPDIFRX_IMR	すべてのフィールド	rw	rw	rw

この表を見ると、IP が STATE_IDLE にある時に、INSEL などのフィールドを設定する必要があることが明らかです。他の IP 状態では、ハードウェアはこのフィールドへの書き込みを阻止します。

注： ***ハードウェアが CBDMAEN および RXDMAEN への「動作中の」書き込みを許可する場合でも、IP がすでにデータを受信しているときに DMA を有効化することはお勧めしません。***

注： ***各マスクビット（PMSK、VMSK、...）は IP がどんな状態でも動作中に変更することができますが、変更が SPDIFRX_DR に保持されているデータに影響を及ぼすことはありません。***

52.4 プログラミング手順

次の例は、SPDIFRX ブロックのアクティブ化の完全な手順を示します。データパスおよびチャネルステータスおよびユーザ情報はどちらも専用の DMA チャネルを使用します。アクティブ化のシーケンスは次のステップに分割されます。

- 選択された SPDIFRX_IN 入力で有効データを待ちます。
- S/PDIF ストリームに同期させます。
- チャネルステータスとユーザ情報を読み出して、完全なオーディオパスセットアップします。
- データ取得を開始します。

SPDIFRX_IN ラインに使用できる有効なデータがあるかどうかを確認する簡単な方法は、WFA ビットを 1 にセットしたまま、SPDIFRX を STATE_SYNC に切り替えることです。これ以降は、検出方法についての説明になります。この機能を次のように実装することも可能です。

- SPDIFRX が同期を検出できるかどうかをソフトウェアで時々チェックする必要があります（たとえば、100 ミリ秒に 1 回）。チェックとは、TERR ビットがセットされているかどうかのチェックです。セットされていれば、アクティビティは 1 つも検出されていないことが分かります。
- SPDIFRX_IN ラインの遷移を検出するために、SPDIFRX_IN 入力を外部割り込みイベントブロックに接続します。アクティビティが検出された場合、SPDIFRXEN を 0b01 または 0b11 にセットすることができます。

  このような 2 種類の実装では、WFA ビットを 0 にセットします。

52.4.1 初期化フェーズ

- 初期化機能について以下に説明します。
- オーディオサンプル、および IEC60958 チャネルステータスとユーザ情報（DMA チャネルの選択とアクティブ化、優先順位、データ転送回数、サーキュラモード／ノンサーキュラモード、DMA 割り込み）の両方について、DMA 転送を設定します。
- 転送先アドレスを設定します。
  - SPDIFRX_CSR レジスタのアドレスを、IEC60958 チャネルステータスおよびユーザ情報のソースアドレスとして設定します。
  - SPDIFRX_DR レジスタのアドレスを、オーディオサンプルのソースアドレスとして設定します。
  - spdifrx_ker_ck の生成を有効にします。最小クロック周波数とサポートされているオーディオサンプリングレートを定義するには、*表 410* を参照してください。
    受信したストリームのオーディオサンプリングレートを前もって知ることはできません。つまり、ユーザは spdifrx_ker_ck周波数として、アプリケーションで処理するはずの最大オーディオサンプリングレートよりも最低でも 704 倍高速の周波数を選択する必要があるということです。たとえば、アプリケーションで最高 96 kHz のストリームを処理できるのであれば、$F_{spdifrx_ker_ck}$ は最低でも 704 x 96 kHz = 67.6 MHz でなければなりません。
- エラーおよびイベントシグナリングの割り込みを有効にします（IFEIE、SYNCDIE、OVRIE、PERRIE を 1 に、その他を 0 にセットします）。なお、SYNCDIE は 0 にセットすることができます。
- SPDIFRX_CR レジスタを次のように設定します。
  - INSEL には希望の入力を選択する。
  - NBTR = 2、WFA = 1（16 回のリトライが可能。アクティビティを待ってから、同期フェーズに進むこと。）
  - PTMSK = CUMSK = 1（プリアンブル、C ビット、U ビットにはデータを混合させないこと。）
  - VMSK = PMSK = 0（パリティエラーおよび有効性ビットにはデータを有効にすること。）
  - CHSEL = 0（チャネルステータスはサブフレーム A から読み出される。）
  - DRFMT = 0b01（データ左詰め）
  - RXSTEO = 1（ステレオモードリニアが予期される。）
  - CBDMAEN = RXDMAEN = 1（DMA チャネルを有効にする。）
  - SPDIFRXEN = 0b01（SPDIFRX を STATE_SYNC に切り替える。）
- CPU は WFI モードに入ることができます。

その後、CPU は DMA または SPDIFRX からの割り込みを受信します。

52.4.2 SPDIFRX からの割り込みの処理

SPDIFRX からの割り込みが受信された場合、ソフトウェアによって、SPDIFRX_SR レジスタを読み出すことで、割り込みの転送元を確認する必要があります。

- SYNCD が 1 にセットされていれば、同期が問題なく完了したことを意味します。この場合、DMA はすでにプログラムされているケースでは、いかなるソフトウェアアクションも必要ありません。ソフトウェアに必要なのは、チャネルステータス情報を読み出すために、DMA 割り込みを待つことだけです。
  SPDIFRX_IFCR レジスタの SYNCDCF ビットを 1 にセットすることで、SYNCD フラグをクリアする必要があります。
- TERR、SERR、FERR のいずれかが 1 にセットされている場合は、ソフトウェアで SPDIFRXEN を 0 にセットし、初期化フェーズから再開する必要があります。
  - TERR は、初期化フェーズ中またはフェーズ後のタイムアウトの発生を示します。
  - SERR は、最大許容リトライ回数に達したことによる同期の失敗を示します。
  - FERR は、同期失敗後の情報の読み出しを示します（予期せぬプリアンブル、データのデコード不良など）。
- PERR が 1 にセットされている場合は、パリティエラーが検出され、それにより、受信したオーディオサンプル、チャネルステータス、ユーザデータのいずれかのビットが破損したことを意味します。ここで必要な操作はアプリケーションによって異なります。たとえば、現在のチャネルステータスブロックが信頼できないとして、ドロップすることもあります。同期は失われていないので、初期化フェーズから再開する必要はありません。
  SPDIFRX_IFCR レジスタの PERRCF ビットを 1 にセットすることで、PERR フラグをクリアする必要があります。

52.4.3 DMA からの割り込みの処理

チャネルステータス（SPDIFRX_CSR）に使用される DMA チャネルから割り込みを受信する場合

エラーが発生していない場合（PERR）、CPU はチャネル情報のデコードを開始できます。たとえば、チャネルステータスのビット 1 は現在のストリームがリニアか否かをユーザに通知します。この情報は、適切な処理チェインをセットアップする上でとても重要です。同様に、チャネルステータスのビット 24～27 は受信ストリームのサンプリング周波数を提供します。

この情報のおかげで、ユーザはデータ受信を開始する前に RXSTEO ビットおよび DRFMT フィールドを設定することができます。たとえば、現在のストリームがリニア PCM でない場合、RXSTEO は 0 にセットされ、DRFMT は 0b10 にセットされます。その場合、ユーザは SPDIFRXEN を 0b11 にセットすることで、データ受信を有効にできます。

SOB ビットが 1 にセットされている場合は新しいブロックの始めを意味します。この情報により、ソフトウェアによるチャネルステータスのビット 0 の識別が容易になります。24 個の値がメモリに転送されるたびに DMA が割り込みを生成する場合、その先頭ワードは常に新しいブロックの始めに相当することに注意してください。

オーディオサンプル（SPDIFRX_DR）に使用される DMA チャネルから割り込みを受信する場合

実行するプロセスはデータタイプ（リニアまたはノンリニア）、および選択されたデータフォーマットによって異なります。

たとえば、リニアモードでは、PE または V ビットがセットされている場合、ローカルに特殊な処理を実施して、出力でのスプールを回避することができます。ノンリニアモードでは、データフレームに独自のチェックサムがあるので、これらのビットは重要ではありません。

52.5 SPDIFRX インタフェースレジスタ

52.5.1 制御レジスタ（SPDIFRX_CR）

アドレスオフセット：0x00

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKSBKPEN	CKSEN	Res.	INSEL[2:0](1)		
										rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WFA (1)	NBTR[1:0] (1)		CHSEL (1)	CBDMAEN (1)	PTMSK (1)	CUMSK (1)	VMSK (1)	PMSK (1)	DRFMT[1:0] (1)		RXSTEO (1)	RXDMAEN (1)	SPDIFRXEN[1:0] (1)	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

1. フィールドの特性の詳細については、 *セクション 52.3.14：レジスタの保護* を参照してください。

ビット 31:22 予約済み、ハードウェアによって 0 に固定されています。

ビット 21 **CKSBKPEN**：バックアップシンボルクロックイネーブル
このビットは、ソフトウェアでセット / クリアされます。
1：CKSEN = ‘1’ である場合、SPDIFRX はバックアップシンボルクロックを生成します。
0：SPDIFRX はバックアップシンボルクロックを生成しません。

ビット 20 **CKSEN**：シンボルクロックイネーブル
このビットは、ソフトウェアでセット / クリアされます。
1：SPDIFRX はシンボルクロックを生成します。
0：SPDIFRX はシンボルクロックを生成しません。

ビット 19 予約済み：ハードウェアによって 0 に固定されています。

ビット 18:16 **INSEL[2:0]**：SPDIFRX 割り込みの選択
0b000：SPDIFRX_IN1 が選択されます。
0b001：SPDIFRX_IN2 が選択されます。
0b010：SPDIFRX_IN3 が選択されます。
0b011：SPDIFRX_IN4 が選択されます。
その他：予約済み

ビット 15 予約済み、ハードウェアによって 0 に固定されています。

ビット 14 **WFA**：アクティビティ待ち
このビットは、ソフトウェアでセット / クリアされます。
1：SPDIFRX は、SPDIFRX_IN ラインでのアクティビティ（4 個の遷移）を待ってから同期を実行します。
0：SPDIFRX は、SPDIFRX_IN ラインでのアクティビティを待たずに同期を実行します。

ビット 13:12 **NBTR[1:0]**：同期フェーズ中の最大許容リトライ回数
0b00：リトライは許可されません（1 回の試行のみ）。
0b01：3 回のリトライが許可されます。
0b10：15 回のリトライが許可されます。
0b11：63 回のリトライが許可されます。

ビット 11 **CHSEL**：チャネルの選択
このビットは、ソフトウェアでセット / クリアされます。
1：制御フローはチャネル B からチャネルステータスを取得します。
0：制御フローはチャネル A からチャネルステータスを取得します。

ビット 10 **CBDMAEN**：制御フロー用の制御バッファ DMA 有効化
このビットは、ソフトウェアでセット / クリアされます。
1：チャネルステータスおよびユーザデータ情報の受信に対して、DMA モードが有効です。
0：チャネルステータスおよびユーザデータ情報の受信に対して、DMA モードが無効です。
このビットがセットされると、CSRNE フラグがセットされるたびに DMA リクエストが行われます。

ビット 9 **PTMSK**：プリマブルタイプビットのマスク
このビットは、ソフトウェアでセット / クリアされます。
1：プリマブルタイプビットは SPDIFRX_DR にコピーされず、代わりにゼロが書き込まれます。
0：プリマブルタイプビットが SPDIFRX_DR にコピーされます。

ビット 8 **CUMSK**：チャネルステータスビットおよびユーザビットのマスク
このビットは、ソフトウェアでセット / クリアされます。
1：チャネルステータスビットおよびユーザビットは SPDIFRX_DR にコピーされず、代わりにゼロが書き込まれます。
0：チャネルステータスビットおよびユーザビットが SPDIFRX_DR にコピーされます。

ビット 7 **VMSK**：有効性ビットのマスク
このビットは、ソフトウェアでセット / クリアされます。
1：有効性ビットは SPDIFRX_DR にコピーされず、代わりにゼロが書き込まれます。
0：有効性ビットが SPDIFRX_DR にコピーされます。

ビット 6 **PMSK**：パリティエラービットのマスク
このビットは、ソフトウェアでセット / クリアされます。
1：パリティエラービットは SPDIFRX_DR にコピーされず、代わりにゼロが書き込まれます。
0：パリティエラービットが SPDIFRX_DR にコピーされます。

ビット 5:4 **DRFMT[1:0]**：RX データフォーマット
このビットは、ソフトウェアでセット / クリアされます。
0b11：予約済み
0b10：データサンプルは、2 個の 16 ビットサンプルを 32 ビットワードにセットすることでパックされます。
0b01：データサンプルは左詰めに配置されます（MSB）。
0b00：データサンプルは右詰めに配置されます（LSB）。

ビット 3 **RXSTEO**：ステレオモード

このビットは、ソフトウェアでセット / クリアされます。

1：ペリフェラルはステレオモードにあります。

0：ペリフェラルはモノラルモードにあります。

このビットは、オーバーラン状況が発生した場合にミスアラインメントを処理するためにも使われます。

ビット 2 **RXDMAEN**：データフロー用レシーバ DMA 有効化

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

このビットがセットされると、RXNE フラグがセットされるたびに DMA リクエストが行われます。

ビット 1:0 **SPDIFRXEN[1:0]**：ペリフェラルブロック有効化

このフィールドはソフトウェアで変更されます。

このフィールドは、次の 3 つの予想される状態のうち、ペリフェラルフェーズを変更するために使用されます。STATE_IDLE、STATE_SYNC、およびSTATE_RCV。

0b00：SPDIFRX（STATE_IDLE）を無効にします。

0b01：SPDIFRX 同期のみを有効にします。

0b10：予約済み

0b11：SPDIF レシーバを有効にします。

注： *1 STATE_RCV から STATE_SYNC への遷移はできません、まず STATE_IDLE へ進んでください。*

2 TATE_IDLE から STATE_RCV への遷移は可能です、ペリフェラルは STATE_IDLE から STATE_SYNC へ遷移し、同期が行われるとすぐに STATE_RCV へ進みます。

52.5.2 割り込みマスクレジスタ（SPDIFRX_IMR）

アドレスオフセット：0x04

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IFEIE	SYNCDIE	SBLKIE	OVRIE	PERRIE	CSRNEIE	RXNEIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済み、ハードウェアによって 0 に固定されています。

ビット 6 **IFEIE**：シリアルインタフェースエラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの SERR = 1、TERR = 1 または FERR = 1 のときには、SPDIFRX 割り込みが生成されます。

ビット 5 **SYNCDIE**：同期完了

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの SYNCD = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

ビット 4 **SBLKIE**：同期ブロック検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの SBD = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

ビット 3 **OVRIE**：オーバーランエラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの OVR = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

ビット 2 **PERRIE**：パリティエラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの PERR = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

ビット 1 **CSRNEIE：制御バッファレディ割り込み有効化**

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの CSRNE = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

ビット 0 **RXNEIE**：RXNE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：SPDIFRX_SR レジスタの RXNE = 1 のときには、SPDIFRX インタフェース割り込みが生成されます。

52.5.3 ステータスレジスタ（SPDIFRX_SR）

アドレスオフセット：0x08

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WIDTH5[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERR	SERR	FERR	SYNCD	SBD	OVR	PERR	CSRNE	RXNE
							r	r	r	r	r	r	r	r	r

ビット 31 予約済み、ハードウェアによって 0 に固定されています。

ビット 30:16 **WIDTH5[14:0]**：spdifrx_ker_ck でカウントされた 5 個のシンボルの継続時間

この値は、5 個の連続したシンボルの全長に含まれる spdifrx_ker_ck クロック周期の数を表します。この値を使用して、S/PDIF シンボルレートを概算できます。その精度は spdifrx_ker_ck の周波数によって制限されます。

たとえば、spdifrx_ker_ck が 84 MHz に固定されており、かつ WIDTH5 = 147d の場合、S/PDIF ストリームの概算サンプリングレートは、

$Fs = 5 \times F_{spdifrx_ker_ck} / (WIDTH5 \times 64) \sim 44.6$ kHz となり、したがって最も近い基準サンプリングレートは 44.1 kHz となります。

WIDTH5 は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

ビット 15:9 予約済み、ハードウェアによって 0 に固定されています。

ビット 8 **TERR**：タイムアウトエラー

このビットは、カウンタ TRCNT がその最大値に達したときにハードウェアによってセットされます。このビットは 2 個の遷移の間隔が長すぎることを示します。また、ほとんどの場合 SPDIFRX_IN 入力に有効な信号がないことを意味します。

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割り込みが生成されます。

0：シーケンスエラーは検出されていません。

1：シーケンスエラーが検出されました。

ビット 7 **SERR**：同期エラー

このビットは、NBTR のリトライ回数のために同期が失敗したとき、ハードウェアによってセットされます。

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割り込みが生成されます。

0：同期エラーは検出されていません。

1：同期エラーが検出されました。

ビット 6 **FERR**：フレーミングエラー

このビットは、データ受信中にエラーが発生した場合にハードウェアによってセットされます（予想した場所にプリアンブルがない、短い遷移はペアでグループ化されない、など）。

このビットは、同期が完了している場合にのみハードウェアによってセットされます。（SYNCD = 1）

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割り込みが生成されます。

0：マンチェスタプロトコル違反は検出されていません。

1：マンチェスタプロトコル違反が検出されました。

ビット 5 **SYNCD**：同期完了

このビットは、最初の同期フェーズが問題なく完了したときに、ハードウェアによってセットされます。
このフラグは、SPDIFRX_CLR_SR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
SPDIFRX_IMR レジスタの SYNCDIE = 1 である場合、割り込みが生成されます。
0：同期は完了していません。
1：同期が完了しました。

ビット 4 **SBD**：同期ブロック検出

このビットは、プリアンブル B が検出されたときに、ハードウェアによってセットされます。
このフラグは、SPDIFRX_CLR_SR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
SPDIFRX_IMR レジスタの SBLKIE = 1 である場合、割り込みが生成されます。
0：プリアンブル B は検出されていません。
1：プリアンブル B が検出されました。

ビット 3 **OVR**：オーバーランエラー

このビットは、RXNE=1 で、SPDIFRX_DR レジスタへ受信データの転送準備ができたとき、かつ SPDIFRX_DR と RX_BUF がどちらもフルの場合、ハードウェアによってセットされます。
このフラグは、SPDIFRX_CLR_SR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
SPDIFRX_IMR レジスタの OVRIE = 1 である場合、割り込みが生成されます。
0：オーバーランエラーはありません。
1：オーバーランエラーが検出されました。

注： このビットがセットされている場合、SPDIFRX_DR レジスタの内容は失われませんが、最後に受信したデータは失われます。

ビット 2 **PERR**：パリティエラー

このビットは、受信したサブフレームのデータおよびステータスビットに奇数個の“0”および“1”が含まれるときに、ハードウェアによってセットされます。
このフラグは、SPDIFRX_CLR_SR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
SPDIFRX_IMR レジスタの PIE = 1 である場合、割り込みが生成されます。
0：パリティエラーはありません。
1：パリティエラー

ビット 1 **CSRNE**：制御バッファレジスタはエンプティではありません。

このビットは、有効な制御情報が準備されているときに、ハードウェアによってセットされます。
このフラグは、SPDIFRX_CSR レジスタを読み出すときにクリアされます。
SPDIFRX_IMR レジスタの CBRDYIE = 1 である場合、割り込みが生成されます。
0：SPDIFRX_CSR レジスタには使用可能な制御ワードはありません。
1：SPDIFRX_CSR レジスタには使用可能な制御ワードがあります。

ビット 0 **RXNE**：読み出しデータレジスタノットエンプティ

このビットは、SPDIFRX_DR レジスタに使用可能な有効データがあるときに、ハードウェアによってセットされます。
このフラグは、SPDIFRX_DR レジスタの読み出しによってクリアされます。
SPDIFRX_IMR レジスタの RXNEIE = 1 である場合、割り込みが生成されます。
0：データは受信されていません。
1：受信データを読み出すことができます。

52.5.4 割り込みフラグクリアレジスタ（SPDIFRX_IFCR）

アドレスオフセット：0x0C

リセット値：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCDCF	SBDCF	OVRCF	PERRCF	Res.	Res.
										w	w	w	w		

ビット 31:6 予約済み、ハードウェアによって 0 に固定されています。

ビット 5 **SYNCDCF**：同期完了フラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの SYNCD フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **SBDCF**：同期ブロック検出フラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの SBD フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 **OVRCF**：オーバーランエラーフラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの OVR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 2 **PERRCF**：パリティエラーフラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの PERR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1:0 予約済み

52.5.5 データ入力レジスタ（SPDIFRX_DR）

アドレスオフセット：0x10

リセット値：0x00000000

このレジスタは DRFMT に応じて 3 通りのフォーマットを取ることができます。DRFMT = 0b00 の場合のフォーマットを以下に示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PT[1:0]		C	U	V	PE	DR[23:16]							
		r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:30 予約済み：ハードウェアによって 0 に固定されています。

ビット 29:28 **PT：**プリアンブルタイプ

これらのビットは受信したプリアンブルを示します。

00：未使用

01：プリアンブル B を受信しました。

10：プリアンブル M を受信しました。

11：プリアンブル W を受信しました。

PTMSK = 1 の場合、このフィールドはゼロに固定されますので、ご注意ください。

ビット 27 **C：**チャネルステータスビット

CUMSK = 0 の場合、受信したチャネルステータスビットを表示します。その他の場合は 0 に固定されます。

ビット 26 **U**：ユーザビット

CUMSK = 0 の場合、受信したユーザビットを表示します。その他の場合は 0 に固定されます。

ビット 25 **V**：有効性ビット

VMSK = 0 の場合、受信した有効性ビットを表示します。その他の場合は 0 に固定されます。

ビット 24 **PE：**パリティエラービット

PMSK = 0 の場合、PERR ビットのコピーを表示します。その他の場合は 0 に固定されます。

ビット 23:0 **DR**：データ値

24 個の受信データビット（D[23] に整列）を表示します。

52.5.6 データ入力レジスタ（SPDIFRX_DR）

アドレスオフセット：0x10

リセット値：0x00000000

このレジスタは DRFMT に応じて 3 通りのフォーマットを取ることができます。DRFMT = 0b01 の場合のフォーマットを以下に示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[7:0]								Res.	Res.	PT[1:0]		C	U	V	PE
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:8 **DR**：データ値

24 個の受信データビット（D[23] に整列）を表示します。

ビット 7:6 予約済み：ハードウェアによって 0 に固定されています。

ビット 5:4 **PT：**プリアンブルタイプ

これらのビットは受信したプリアンブルを示します。

00：未使用

01：プリアンブル B を受信しました。

10：プリアンブル M を受信しました。

11：プリアンブル W を受信しました。

PTMSK = 1 の場合、このフィールドはゼロに固定されますので、ご注意ください。

ビット 3 **C：**チャネルステータスビット

CUMSK = 0 の場合、受信したチャネルステータスビットを表示します。その他の場合は 0 に固定されます。

ビット 2 **U**：ユーザビット

CUMSK = 0 の場合、受信したユーザビットを表示します。その他の場合は 0 に固定されます。

ビット 1 **V**：有効性ビット

VMSK = 0 の場合、受信した有効性ビットを表示します。その他の場合は 0 に固定されます。

ビット 0 **PE：**パリティエラービット

PMSK = 0 の場合、PERR ビットのコピーを表示します。その他の場合は 0 に固定されます。

52.5.7 データ入力レジスタ（SPDIFRX_DR）

アドレスオフセット：0x10

リセット値：0x00000000

このレジスタは DRFMT に応じて 3 通りのフォーマットを取ることができます。

DRFMT = 0b10 の場合に提示されたデータフォーマットは 16 ビットしか使用されないため（S/PDIF サブフレームのビット 23～8）、ノンリニアモード専用です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DRNL2[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRNL1[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **DRNL2**：データ値

このフィールドはチャネル A を含みます。

ビット 15:0 **DRNL1**：データ値

このフィールドはチャネル B を含みます。

52.5.8 チャネルステータスレジスタ（SPDIFRX_CSR）

アドレスオフセット：0x14

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOB	CS[7:0]							
							r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:25 予約済み

ビット 24 **SOB**：ブロックの開始

このビットは、CS[0] ビットが新しいブロックの先頭ビットに対応するかどうかを示します。

0：CS[0] は新しいブロックの先頭ビットではありません。

1：CS[0] は新しいブロックの先頭ビットです。

ビット 23:16 **CS[7:0]**：チャネル A ステータス情報

CS[0] ビットは一番古い値です。

ビット 15:0 **USR[15:0]**：ユーザデータ情報

USR[0] ビットは一番古い値で、チャネル A から受信します。USR[1] はチャネル B から受信します。

したがって、USR[n] ビットの”n”は偶数であり、そうでない場合はチャネル B から受信したことになります。

52.5.9 デバッグ情報レジスタ（SPDIFRX_DIR）

アドレスオフセット：0x18

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	TLO[12:0]												
			r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	THI[12:0]												
			r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:29 予約済み、ハードウェアによって 0 に固定されています。

ビット 16:28 **TLO**：閾値ロー （TLO = 1.5 x UI / $T_{spdifrx_ker_ck}$）

このフィールドには現在の閾値ローの概算値が含まれています。この値を使用して、受信ストリームのサンプリングレートを概算することができます。TLO の精度は spdifrx_ker_ck の 1 周期に制限されます。サンプリングレートは次のように算出することができます。

サンプリングレート = [2 x TLO x $T_{spdifrx_ker_ck}$ +/- $T_{spdifrx_ker_ck}$] x 2/3

TLO は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

ビット 15:13 予約済み、ハードウェアによって 0 に固定されています。

ビット 12:0 **THI**：閾値ハイ（THI = 2.5 x UI / $T_{spdifrx_ker_ck}$）

このフィールドには現在の閾値ハイの概算値が含まれています。この値を使用して、受信ストリームのサンプリングレートを概算することができます。THI の精度は spdifrx_ker_ck の 1 周期に制限されます。サンプリングレートは次のように算出することができます。

サンプリングレート = [2 x THI x $T_{spdifrx_ker_ck}$ +/- $T_{spdifrx_ker_ck}$] x 2/5

THI は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

52.5.10 SPDIFRX バージョンレジスタ（SPDIFRX_VERR）

アドレスオフセット：0x03F4

リセット値：0x0000 0012

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MAJREV[3:0]				MINREV[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **MAJREV[3:0]**：メジャーリビジョン
このビットは SPDIFRX のメジャーリビジョンを返します。
メジャーリビジョンは 1 です。

ビット 3:0 **MINREV[3:0]**：マイナーリビジョン
このビットは SPDIFRX のマイナーリビジョンを返します。
マイナーリビジョンは 2 です。

52.5.11 SPDIFRX 識別レジスタ（SPDIFRX_IDR）

アドレスオフセット：0x03F8

リセット値：0x0013 0041

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID[15:0]															
r															

ビット 31:0 **ID[31:0]**：SPDIFRX 識別子
これらのビットは SPDIFRX の識別子を返します。

52.5.12 SPDIFRX サイズ識別レジスタ（SPDIFRX_SIDR）

アドレスオフセット：0x03FC

リセット値：0xA3C5 DD01

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SID[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SID[15:0]															
r															

ビット 31:0 **SID[31:0]**：サイズ識別

これらのビットは、SPDIFRX レジスタに割り当てられるメモリ領域のサイズを返します。

このメモリ領域のサイズは 1 kB です。

52.5.13 SPDIFRX インタフェースレジスタマップ

表 413 に、SPDIFRX レジスタマップとリセット値を示します。

表 413. SPDIFRX インタフェースレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**SPDIFRX_CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKSBKPEN	CKSEN	Res.	INSEL[2:0]			Res.	WFA	NBTR[1:0]		CHSEL	CBDMAEN	PTMSK	CUMSK	VMSK	PMSK	DRFMT[1:0]		RXSTEO	RXDMAEN	SPDIFRXEN[1:0]	
	リセット値											0	0		0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**SPDIFRX_IMR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IFEIE	SYNCDIE	SBLKIE	OVRIE	PERRIE	CSRNEIE	RXNEIE
	リセット値																										0	0	0	0	0	0	0
0x08	**SPDIFRX_SR**	Res.	WIDTH5[14:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERR	SERR	FERR	SYNCD	SBD	OVR	PERR	CSRNE	RXNE
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0	0	0	0
0x0C	**SPDIFRX_IFCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCDCF	SBDCF	OVRCF	PERRCF	Res.	Res.
	リセット値																											0	0	0	0		
0x10	**SPDIFRX_DR**	Res.	Res.	PT[1:0]		C	U	V	PE	DR[23:0]																							
		DR[23:0]																								Res.		PT[1:0]		C	U	V	PE
		DRNL2[15:0]																DRNL1[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**SPDIFRX_CSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOB	CS[7:0]								USR[15:0]															
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**SPDIFRX_DIR**	Res.	Res.	Res.	TLO[12:0]													Res.	Res.	Res.	THI[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x03F4	**SPDIFRX_VERR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MAJREV[3:0]				MINREV[3:0]			
	リセット値																									0	0	0	1	0	0	1	0
0x03F8	**SPDIFRX_IDR**	ID[31:16]																ID[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
0x03FC	**SPDIFRX_SIDR**	SID[31:16]																SID[15:0]															
	リセット値	1	0	1	0	0	0	1	1	1	1	0	0	0	1	0	1	1	1	0	1	1	1	0	1	0	0	0	0	0	0	0	1

53 シングルワイヤプロトコルマスタインタフェース（SWPMI）

53.1 概要

シングルワイヤプロトコルマスタインタフェース（SWPMI）は、ETSI TS 102 613 技術仕様で定義されたコンタクトレスフロントエンド（CLF）に対応するマスタインタフェースです。

シングルワイヤプロトコル（SWP）の原理は、全二重モードでのデジタル情報の送信に基づいています。

- S1 信号（マスタからスレーブ）は、電圧ドメインのデジタル変調（L または H）で送信されます（*図 673 : S1 信号コーディング*を参照）
- S2 信号（スレーブからマスタ）は、電流ドメインのデジタル変調（L または H）で送信されます（*図 674 : S2 信号コーディング*を参照）

図 673. S1 信号コーディング

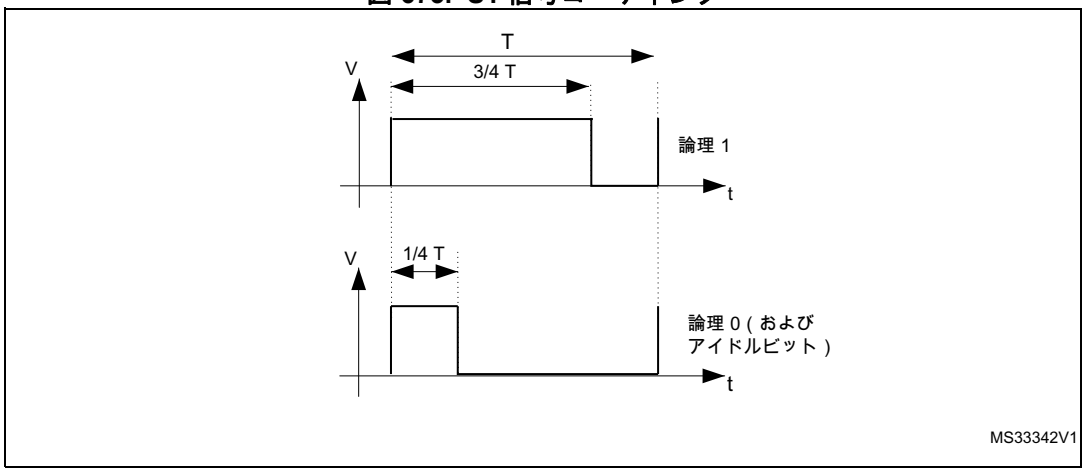
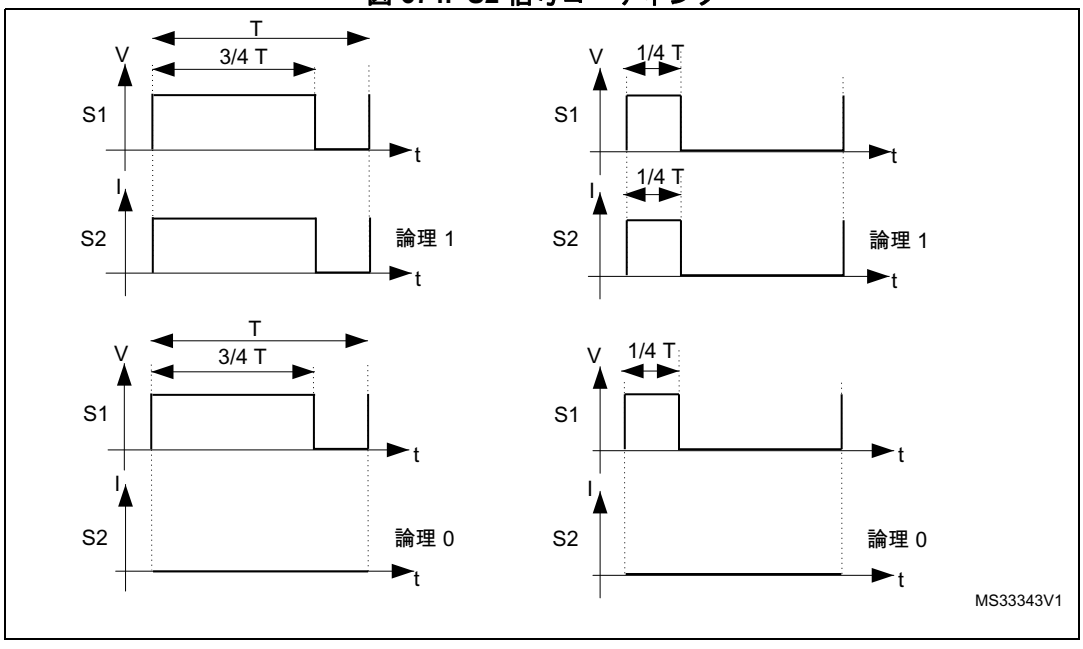


図 674. S2 信号コーディング



53.2 SWPMI の主な機能

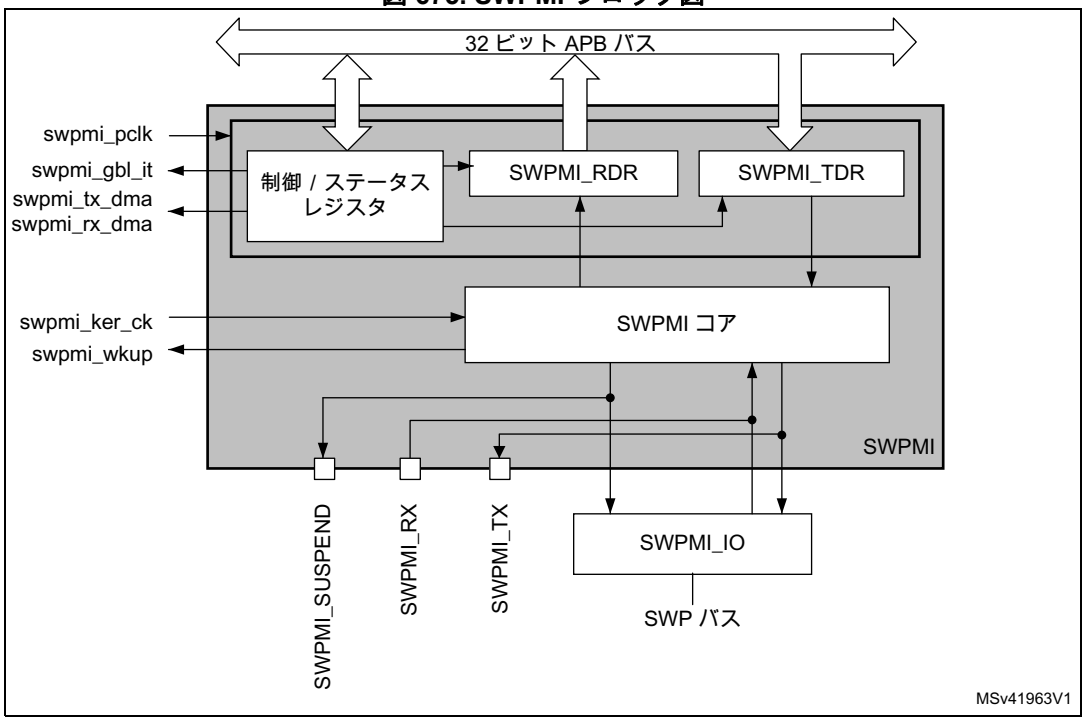
SWPMI モジュールの主な機能を以下に示します（*図 53.3.4：SWP バス状態*を参照）。

- 全二重通信モード
- 自動 SWP バス状態管理
- フレーム開始（SOF）の自動処理
- フレーム終了（EOF）の自動処理
- スタッフィングビットの自動処理
- 送信時の自動 CRC-16 計算および生成
- 受信時の自動 CRC-16 計算および確認
- 32 ビット送信データレジスタ
- 32 ビット受信データレジスタ
- 有効な DMA の実装とマルチフレームバッファリングのためのマルチソフトウェアバッファモード
- 最大 2 M ビット／秒の設定可能なビットレート
- 設定可能な割り込み
- CRC エラー、アンダーラン、オーバーランフラグ
- フレーム受信完了および送信完了フラグ
- スレーブレジューム検出フラグ
- 試験用ループバックモード
- ETSI TS 102 613 技術仕様に準拠した内蔵 SWPMI_IO トランシーバ
- GPIO での SWPMI_RX、SWPMI_TX、SWPMI_SUSPEND 信号出力専用モード（外部トランシーバ接続時）

53.3 SWPMI の機能詳細

53.3.1 SWPMI ブロック図

図 675. SWPMI ブロック図



swpmi_ker_ck（SWPMI コアクロックソース）を選択するには、*セクション 8.7.18 : RCC ドメイン 2 カーネルクロック設定レジスタ（RCC_D2CCIP1R）*の SWPSRC ビットを参照してください。

*注 : **スレーブによる RESUME での STOP モードからの復帰をサポートするために、swpmi_ker_ck の HSI を選択する必要があります。この機能が必要ではない場合、swpmi_pclk を選択でき、STOP モードに移行する前に SWPMI を無効にする必要があります。***

53.3.2 SWPMI ピンおよび内部信号

*表 414*にパッケージのピンまたはボールに接続された SWPMI スレーブ入出力信号のリストを、*表 415*に内部 SWPMI 信号のリストをそれぞれ示します。

表 414. パッケージのピンまたはボールに接続された SWPMI スレーブ入力／出力信号

信号名	信号タイプ	説明
SWPMI_SUSPEND	デジタル出力	SWPMI サスペンド信号
SWPMI_TX	デジタル出力	SWPMI 送信信号
SWPMI_RX	デジタル入力	SWPMI 受信信号

表 415. SWPMI 内部入力／出力信号

信号名	信号タイプ	説明
swpmi_pclk	デジタル入力	APB クロック
swpmi_ker_ck	デジタル入力	SWPMI カーネルクロック
swpmi_wkup	デジタル出力	SWPMI ウェイクアップ信号
swpmi_gbl_it	デジタル出力	SWPMI 割り込み信号
swpmi_tx_dma	デジタル出力	SWPMI DMA 送信リクエスト
swpmi_rx_dma	デジタル出力	SWPMI DMA 受信リクエスト

53.3.3 SWP の初期化と有効化

初期化と有効化により、SWPMI_IO 状態がローからハイにセットされます。

内部トランシーバの使用時には、手順は以下のようになります。

1. VDD 電圧（3 V または 1.8 V）に従って、SWPMI_OR レジスタの SWP_CLASS ビットを設定します。
2. SWPMI_CR レジスタの SWPTEN をセットして、SWPMI_IO トランシーバを有効にし、SWPMI_IO を Low レベルにセットします（SWP バスが DEACTIVATED になります）。
3. SWPMI_SR レジスタの RDYF フラグがセットされるのを待ちます（フラグをポーリングするか、SWPMI_IER レジスタで RDYIE ビットの割り込みを有効にします）。
4. SWPMI_CR レジスタの SWPACT ビットをセットして、SWP を ACTIVATE にします（DEACTIVATED から SUSPENDED に移行します）。

53.3.4 SWP バス状態

SWP バスには、次の状態があります。DEACTIVATED、SUSPENDED、ACTIVATED です。

いくつか遷移する方法があります。

- ACTIVATE：DEACTIVATED から SUSPENDED 状態への遷移
- SUSPEND：ACTIVATED から SUSPENDED 状態への遷移
- マスタによる RESUME：マスタによって開始される SUSPENDED から ACTIVATED 状態への遷移
- スレーブによる RESUME：スレーブによって開始される SUSPENDED から ACTIVATED 状態への遷移
- DEACTIVATE：SUSPENDED から DEACTIVATED 状態への遷移

有効化

リセット中とリセット直後、SWPMI_IO はアナログ モードに設定されます。SWP バスを有効化するには、*セクション 53.3.3：SWP の初期化と有効化*を参照してください。

サスペンド

送受信にかかわらずスレーブとの通信がある限り、SWP バスが ACTIVATED 状態になります。7 個のアイドルビット後に送受信動作がない場合、SWP バスは SUSPENDED 状態に戻ります。

マスタによる RESUME

SWPMI が有効になると、SWPMI フレーム送信をリクエストできます。SWPMI は、まず遷移シーケンスと 8 個のアイドルビットを送信（マスタによる RESUME）してから、フレーム送信を開始します。SWP は、マスタによる RESUME の後に SUSPENDED から ACTIVATED 状態に移行します（*図 676：SWP バス状態*を参照）。

スレーブによる RESUME

SWPMI が有効になると、SWPMI がスレーブから RESUME を受信した場合、SWP を SUSPENDED から ACTIVATED 状態に移行することもできます。スレーブによる RESUME によって、SWPMI_ISR レジスタの SRF フラグがセットされます。

無効化

無効化リクエスト

これ以上通信が必要ではない場合および SWP が SUSPENDED モードの場合、SWPMI ペリフェラルを無効にして SWP の DEACTIVATED モードへの切り替えをリクエストできます。DEACTIVATED モードをリクエストするには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをセットする必要があります。スレーブによる RESUME が SWPMI によって検出されない場合、DEACTF フラグが SWPMI_ISR レジスタでセットされ、SWPACT ビットが SWPMI_ICR レジスタでクリアされます。スレーブによる RESUME が SWPMI によって検出され、ソフトウェアによって DEACT ビットがセットされている場合、SRF フラグが SWPMI_ISR レジスタでセットされ、DEACTF はクリアされたまま、SWPACT はセットされたままになり、DEACT ビットはクリアされます。

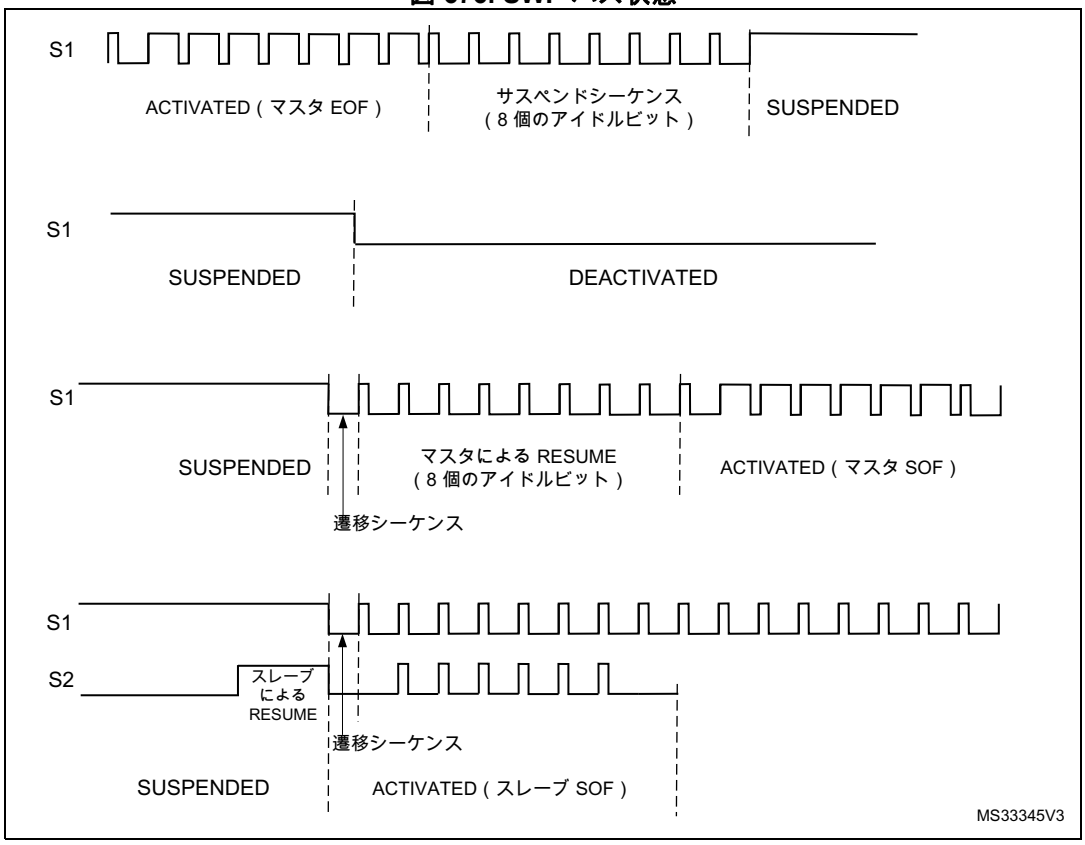
再度 SWP を有効化するには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをクリアしてから SWPACT ビットをセットする必要があります。

無効化モード

SWP を DEACTIVATED モードにすぐに切り替え、スレーブによる RESUME の受信をすべて無視するには、SWPMI_CR レジスタの SWPACT ビットをクリアする必要があります。

注： ***消費電流をさらに低減するには、GPIO コントローラで SWPMI_IO ポートを出力プッシュプルローに設定してから、SWPMI_CR レジスタの SWPTEN ビットをクリアします（セクション 11：汎用 I/O（GPIO）を参照）。***

図 676. SWP バス状態



53.3.5 SWPMI_IO（内部トランシーバ）バイパス

ETSI TS 102 613 技術仕様に準拠した SWPMI_IO（トランシーバ）は、マイクロコントローラに内蔵されています。それにもかかわらず、SWPMI_OR レジスタの SWP_TBYP ビットをセットすることでバイパスが可能です。この場合、SWPMI_IO は無効になり、SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は 3 つの GPIO でオルタネート機能として使用できます（製品データシートの「ピン出力およびピンの説明」を参照）。外部トランシーバを接続するには、この設定を選択します。

注： ***SWPMI_IO バイパスモードでは、SWPMI_CR レジスタの SWPTEN ビットをクリアされた状態に保つ必要があります。***

53.3.6 SWPMI ビットレート

次の式に応じて、SWPMI_BRR レジスタでこのビットレートをセットする必要があります。

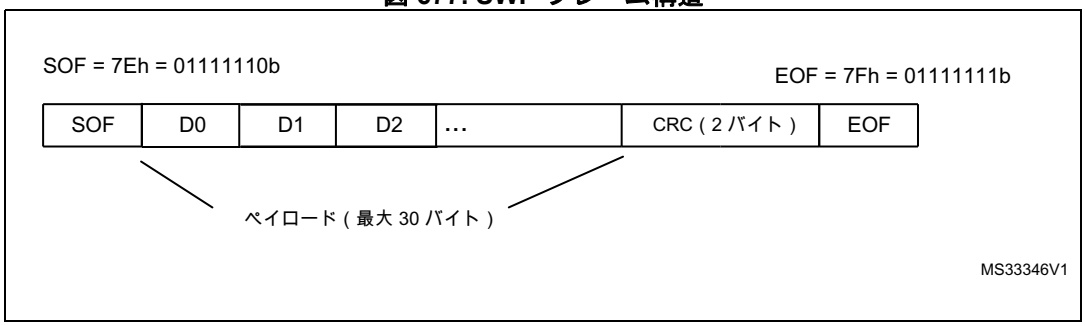
$F_{SWP} = F_{swpmi_ker_ck} / ((BR[7:0]+1)x4)$

注： ***最大ビットレートは 2 M ビット／秒です。***

53.3.7 SWPMI フレーム処理

SWP フレームはフレーム開始（SOF）、1～30 バイトのペイロード、16 ビット CRC、フレーム終了（EOF）で構成されます（*図 677 : SWP フレーム構造*を参照）。

図 677. SWP フレーム構造



SWPMI には、送信用に 32 ビットデータレジスタ（SWPMI_TDR）、受信用に 32 ビットデータレジスタ（SWPMI_RDR）が内蔵されています。

送信時、SOF 挿入、CRC 計算および挿入、EOF 挿入が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを提供する必要があります。SWPMI_TDR レジスタにデータが書き込まれた直後に、フレーム送信が開始します。専用のフラグは、空の送信データレジスタと完全なフレーム送信イベントを示します。

受信時、SOF 削除、CRC 計算および確認、EOF 削除が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを読み出す必要があります。専用のフラグは、フルの受信データレジスタと完全なフレーム受信イベントおよび場合によっては CRC エラーイベントを示します。

スタッフィングビット挿入（送信時）およびスタッフィングビット削除（受信時）は、SWPMI コアによって自動的に管理されます。これらの操作はユーザーに対して分かりやすくなっています。

53.3.8 送信手順

フレーム送信を開始する前に、ユーザは SWP を有効にする必要があります。*セクション 53.3.3 : SWP の初期化と有効化*を参照してください。

フレーム送信のために考えられるソフトウェア実装方法がいくつかあります。ソフトウェアバッファなしモード、シングルソフトウェアバッファモード、マルチソフトウェアバッファモードです。

ソフトウェアバッファの使用には、RAM メモリのソフトウェアバッファから SWPMI ペリフェラルの送信データレジスタにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割り込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム送信処理を行います。SWPMI に 32 ビット送信データレジスタ（SWPMI_TDR）があり、このレジスタに書き込むことで、最大 4 バイトの送信をトリガできます。

ソフトウェアバッファなしモードは、SWPMI_CR レジスタの TXDMA ビットをクリアすることで選択されます。

SWPMI_TDR レジスタへの初回書き込みで、フレーム送信が開始します。最初の 32 ビットワードの下位バイト（SWPMI_TDR レジスタに書き込まれたビット [7:0]）は、ペイロードのデータバイト数を示し、そのワードの他の 3 バイトには、ペイロードの最初の 3 バイト（ビット [15:8] にはペイロードの最初のバイト、ビット [23:16] には 2 番目のバイト、ビット [31:24] には 3 番目のバイト）を含

める必要があります。そして、SWPMI_TDR レジスタへの次の書き込みでは、各書き込みに最大 4 の次のペイロードデータバイトのみを含めます。

注： ***SWPMI_TDR レジスタに書き込まれた最初の 32 ビットワードの下位バイトによって、ペイロードのデータバイト数をコーディングします。この数は、1～30 になります。下位バイトの他の値はすべて無視され、送信は開始されません。***

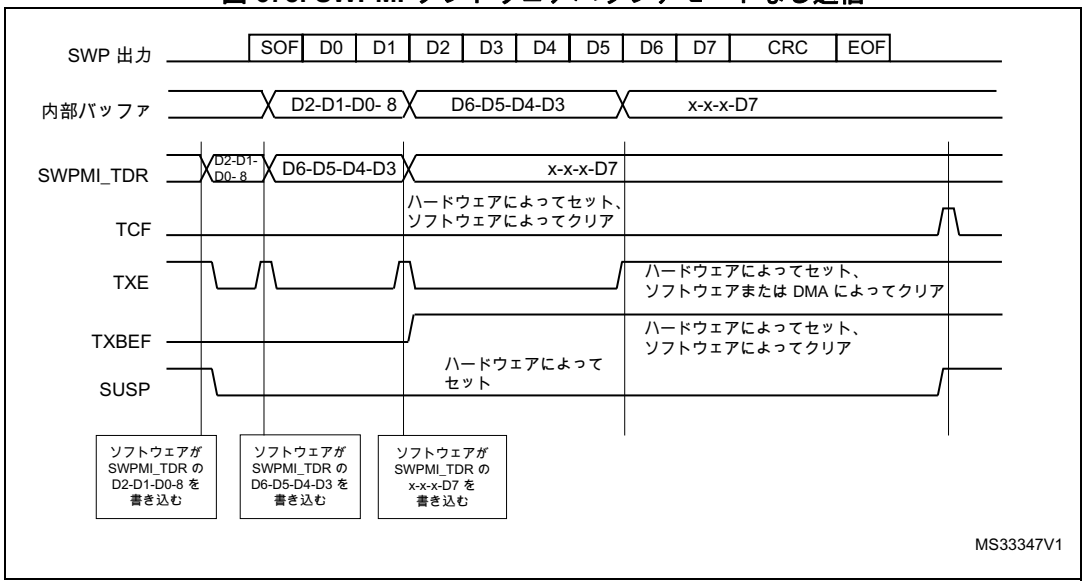
SWPMI_TDR レジスタに書き込むと、次のアクションが発生します。

- SWP バス状態が SUPENDED の場合、遷移シーケンスおよび 8 個のアイドルビット（マスタによる RESUME）を送信します（SWP バス状態がすでに ACTIVATED の場合は発生しません）。
- フレーム開始（SOF）を送信します。
- SWPMI_TRD レジスタの内容に応じてペイロードを送信します。ペイロードのバイト数が 3 より大きい場合、SWPMI_ISR レジスタの TXBEF フラグがセットされていない限り、SWPMI_ISR レジスタの TXE フラグがセットされるたびに、SWPMI_TDR をソフトウェアで再入力する必要があります。
- SWPMI コアによって自動計算された 16 ビット CRC を送信します。
- フレーム終了（EOF）を送信します。

TXE フラグは、ソフトウェアによる SWPMI_TDR レジスタへの書き込み時に自動的にクリアされます。

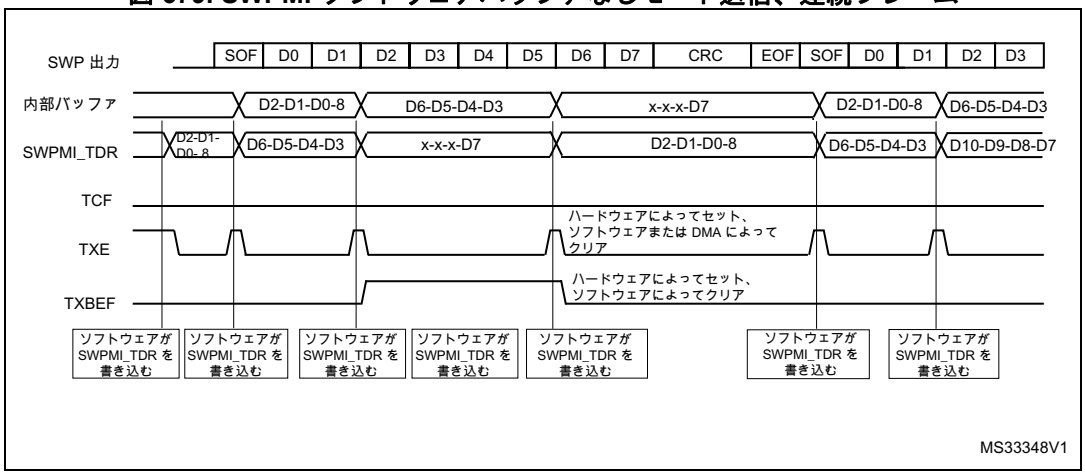
完全なフレームが送信され、他のフレーム送信がリクエストされていない場合（TXBEF フラグセットの後に再度 SWPMI_TDR に書き込まれなかった場合）、EOF 送信後 7 個のアイドルビットで TCF および SUSP フラグが SWPMI_ISR レジスタにセットされ、TCIE ビットが SWPMI_IER レジスタでセットされている場合、割り込みが生成されます（*図 678：SWPMI ソフトウェアバッファモードなし送信*を参照）。

図 678. SWPMI ソフトウェアバッファモードなし送信



EOF 送信の終了前に別のフレーム送信がリクエストされている場合、TCF フラグはセットされず、フレームはアイドルビットを 1 個だけ挟んで前のフレームと連続します（*図 679：SWPMI ソフトウェアバッファなしモード送信、連続フレーム*を参照）。

図 679. SWPMI ソフトウェアバッファなしモード送信、連続フレーム



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを送信できます。DMA が 32 ビット SWPMI_TDR レジスタに再入力すると、ソフトウェアが SWPMI_TXBEF フラグを使用してフレーム終了の送信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットをセットして、TXMODE ビットをクリアすることで選択されます。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、SWP フレーム長に応じて設定する必要があります。
- 転送元アドレスは、RAM の SWP フレームバッファです
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA ビットをセットします。
2. SWPMI_IER レジスタの TXBEIE ビットをセットします。
3. RAM メモリのバッファを（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA 転送およびフレーム送信を開始します。

SWPMI_ISR で TXE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。TXE フラグは、DMA による SWPMI_TDR レジスタへの書き込み時に自動的にクリアされます。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされており、別のフレームを送信する必要がある場合、次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. RAM メモリのバッファを、送信する次のフレームの内容で更新します。
3. DMA モジュールで転送すべきワード総数を設定します。
4. DMA モジュールでストリームまたはチャネルを有効化して、次のフレーム送信を開始します。
5. SWPMI_ICR レジスタの CTXBEF ビットをセットして、TXBEF フラグをクリアします。

マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA によってソフトウェアがバッファ更新の遅延を許容して確実に連続送信を行うことができます。ソフトウェアでいつでも DMA カウンタを確認して、RAM メモリで適宜 SWP フレームを更新できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

RAM メモリの各送信バッファは、SWP フレームペイロードのバイト数にかかわらず、8 個の 32 ビットワードの固定長でなければなりません。RAM メモリの送信バッファはソフトウェアによって満たされる必要があり、2 つの連続したバッファ間に 8 のオフセットを維持します。バッファの最初のデータバイトは、フレームペイロードのバイト数です。バッファの例については、*図 680 : SWPMI マルチソフトウェアバッファモード送信*を参照してください。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットおよび TXMODE ビットをセットすることで選択されます。

たとえば、4 個の送信バッファと連動するには、DMA を以下のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、32（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは、RAM のバッファ 1 です
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA をセットします。
2. SWPMI_IER レジスタの TXBEIE をセットします。
3. RAM メモリのバッファ 1、バッファ 2、バッファ 3、バッファ 4 を（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA を開始します。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CTXBEF ビットをセットして TXBEF フラグをクリアする必要があり、RAM メモリのバッファ 1 を更新できるようになります。

次の SWPMI 割り込みルーチンの発生では、バッファ 2 を更新するという形で進めます。

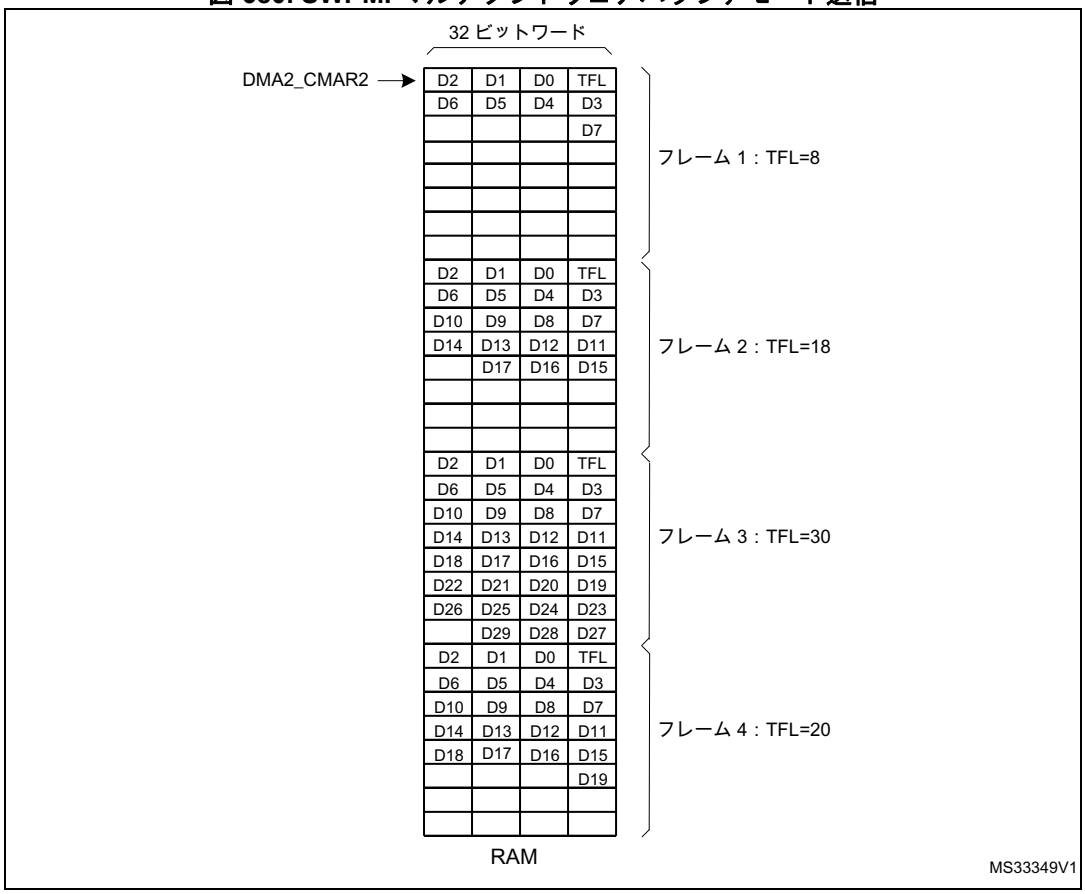
ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに RAM メモリから転送および送信されたフレームを取得できます。たとえば、ソフトウェアが 4 個の送信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で更新する準備ができていることを示します。これは、複数のフレームが送信されてからソフトウェアが SWPMI 割り込みを処理できる場合に役立ちます。この場合、ソフトウェアは複数のバッファを更新する必要があります。

送信するフレームがこれ以上ない場合、DMA モジュールでサーキュラモードを無効にする必要があります。送信は、バッファ 4 の送信終了時に停止します。

その前（バッファ 2 の終了時など）に送信を停止する必要がある場合、バッファ 3 およびバッファ 4 で最初のワードの下位バイトを 0 にセットする必要があります。

SWPMI_CR レジスタの TXDMA ビットは、ペイロードのデータバイト数が最初のワードの最下位バイトの 0 として読み出された直後に、ハードウェアによってクリアされます。

図 680. SWPMI マルチソフトウェアバッファモード送信



53.3.9 受信手順

フレーム受信を開始する前に、ユーザは SWP を有効にする必要があります（*セクション 53.3.3 : SWP の初期化と有効化*を参照）。

SWPACT ビットが SWPMI_CR レジスタでセットされると、スレーブによる RESUME 状態によって SWPMI_ISR レジスタで SRF フラグがセットされ、フレーム受信のために SWPMI が自動的に有効化されます。

SWP バスがすでに ACTIVATED 状態（フレーム送信中などのため）の場合、SWPMI コアはスレーブによる RESUME 状態を必要とせず、ただちに受信を行えます。

フレーム受信のために考えられるソフトウェア実装方法がいくつかあります。

- ソフトウェアバッファなしモード
- シングルソフトウェアバッファモード
- マルチソフトウェアバッファモード

ソフトウェアバッファの使用には、SWPMI ペリフェラルの受信データレジスタから RAM メモリのソフトウェアバッファにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割り込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム受信処理を行います。SWPMI に 32 ビット受信データレジスタ（SWPMI_RDR）があり、このレジスタを読み出す前に、最大 4 バイトを受信できます。

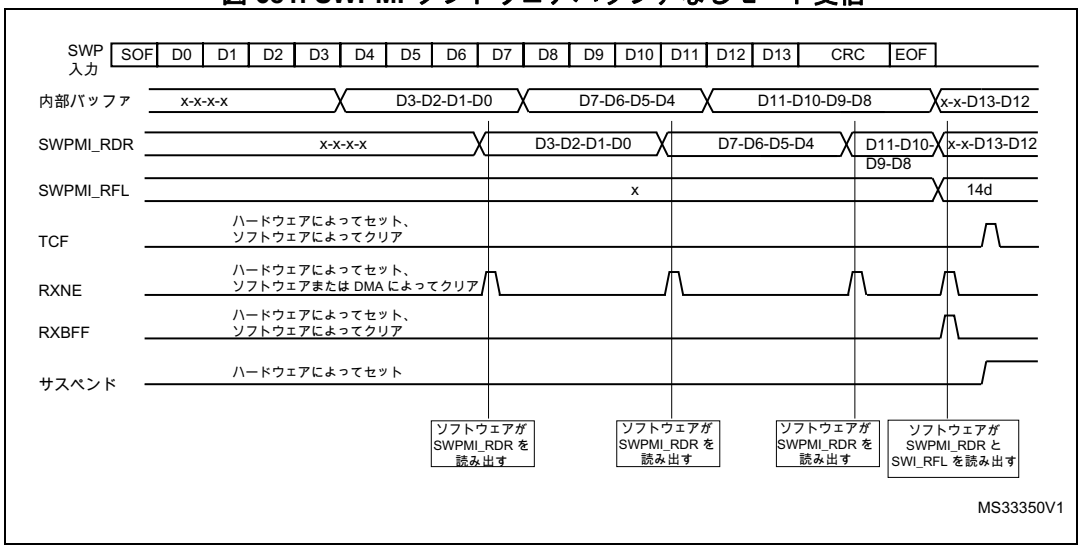
ソフトウェアバッファなしモードは、SWPMI_CR レジスタの RXDMA ビットをリセットすることで選択されます。

フレーム開始（SOF）を受信すると、次のバイト（ペイロード）が SWPMI_RDR レジスタに格納されます。SWPMI_RDR がフルになると、SWPMI_ISR の RXNE フラグがセットされ、SWPMI_IER レジスタの RIE ビットがセットされている場合は割り込みが生成されます。ユーザは SWPMI_RDR レジスタを読み出すことができ、RXNE フラグはソフトウェアによる SWPMI_RDR レジスタの読み出し時に自動的にクリアされます。

CRC およびフレーム終了（EOF）を含む完全なフレームを受信すると、SWPMI_ISR レジスタで RXNE および RXBFF フラグがセットされます。ユーザは SWPMI_RDR レジスタでペイロードの最後のバイトを読み出し、SWPMI_ICR で CRXBFF フラグをセットして、RXBFF フラグをクリアする必要があります。ペイロードのデータバイト数は、SWPMI_RFL レジスタで使用できます。再度、ソフトウェアが SWPMI_RDR レジスタを読み出しているときに RXNE フラグが自動的にリセットされます（*図 681 : SWPMI ソフトウェアバッファなしモード受信*を参照）。

RXNE がクリアされている状態で SWPMI_RDR レジスタを読み出すと、0 が返されます。

図 681. SWPMI ソフトウェアバッファなしモード受信



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを受信できます。DMA が 32 ビット SWPMI_RDR レジスタから RAM メモリに受信したデータを転送すると、ソフトウェアが SWPMI_RBFF フラグを使用してフレーム終了の受信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットをセットして、RXMODE ビットをクリアすることで選択されます。

DMA は次のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 に設定する必要があります。
- 転送元アドレスは SWPMI_RDR レジスタです。
- 転送先アドレスは、RAM の SWP フレームバッファです

そして、ユーザは次の操作が必要です。

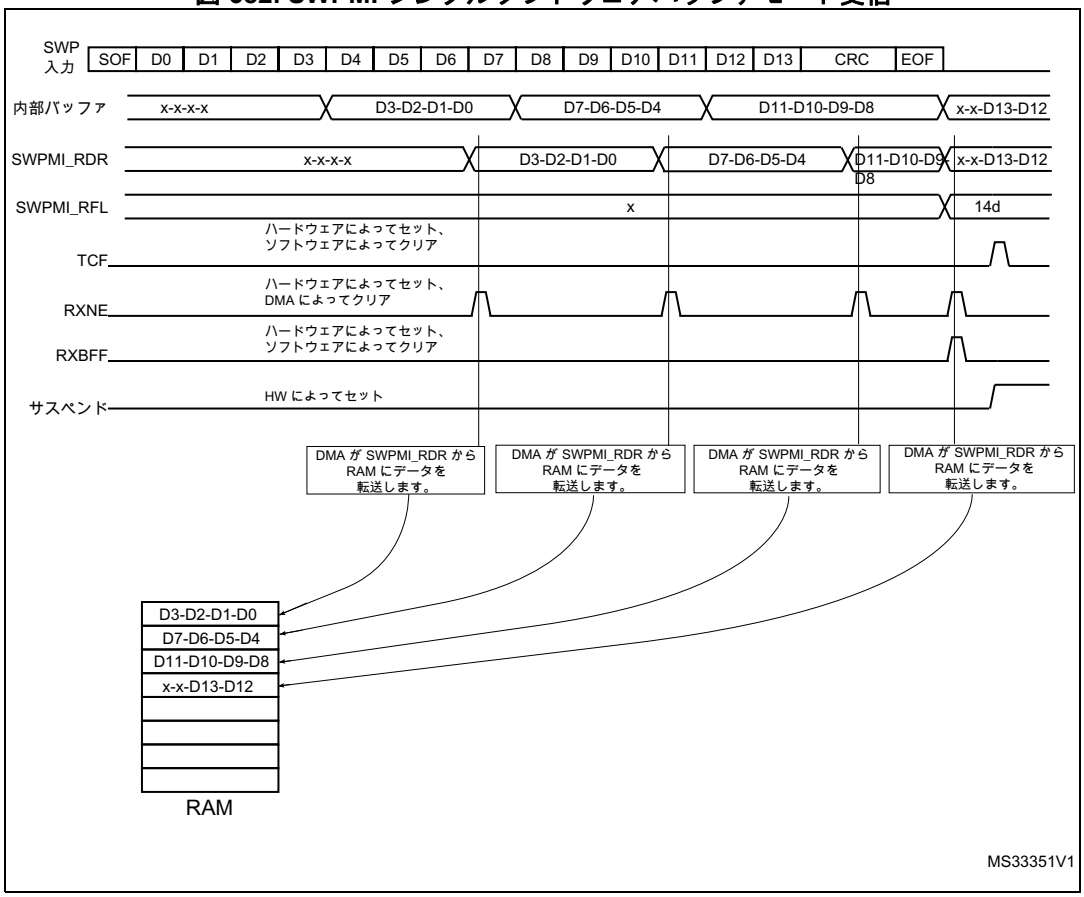
1. SWPMI_CR レジスタの RXDMA ビットをセットします。
2. SWPMI_IER レジスタの RXBFIE ビットをセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI_ISR で RXNE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。RXNE フラグは、DMA による SWPMI_RDR レジスタの読み出し時に自動的にクリアされます。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの RXBFF ビットを確認する必要があります。これがセットされている場合、ユーザは次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. SWPMI_RFL レジスタで受信したフレームペイロードのバイト数を読み出します。
3. RAM バッファでフレームペイロードを読み出します。
4. DMA モジュールでストリームまたはチャネルを有効にします。
5. SWPMI_ICR レジスタの CRXBFF ビットをセットして、RXBFF フラグをクリアします（*図 682: SWPMI シングルソフトウェアバッファモード受信*を参照）。

図 682. SWPMI シングルソフトウェアバッファモード受信



マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA を使用して確実に連続受信を行うことができます。フレームペイロードは、RAM メモリにフレームステータスフラグとともに格納されます。ソフトウェアでいつでも DMA カウンタとステータスフラグを確認して、RAM メモリで受信した SWP フレームを処理できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットおよび RXMODE ビットをセットすることで選択されます。

RAM で n 個の受信バッファと連動するには、DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 x n（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは SWPMI_TDR レジスタです。
- 転送先アドレスは、RAM のバッファ 1 アドレスです

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの RXDMA をセットします。
2. SWPMI_IER レジスタの RXBFIE をセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの RXBFF を確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CRXBFF ビットをセットして RXBFF フラグをクリアする必要があり、最初のバッファで受信した最初のフレームペイロード（DMA2_CMAR1 でセットされた RAM アドレスで）を読み出せるようになります。

ペイロードのデータバイト数は、最後の 8 番目のワードのビット [23:16] で使用できます。

次の SWPMI 割り込みルーチンの発生では、2 番目のバッファで受信した 2 番目のフレーム（DMA2_CMAR1 + 8 でセットされたアドレス）を読み出すいう形で進めます（*図 683 : SWPMI マルチソフトウェアバッファモード受信*を参照）。

アプリケーションソフトウェアが、次のフレーム受信の前に SMPMI 割り込みを確実に処理できない場合、各バッファステータスが 8 番目のバッファワードの最上位バイトで使用できます。

- CRC エラーフラグ（SWPMI_ISR レジスタの RXBERF フラグと同等）は、8 番目のワードのビット 24 で使用できます。CRC エラーの説明については、*セクション 53.3.10 : エラー管理*を参照してください。
- 受信オーバーランフラグ（SWPMI_ISR レジスタの RXOVRF フラグと同等）は、8 番目のワードのビット 25 で使用できます。オーバーランエラーの説明については、*セクション 53.3.10 : エラー管理*を参照してください。
- 受信バッファフルフラグ（SWPMI_ISR レジスタの RXBFF フラグと同等）は、8 番目のワードのビット 26 で使用できます。

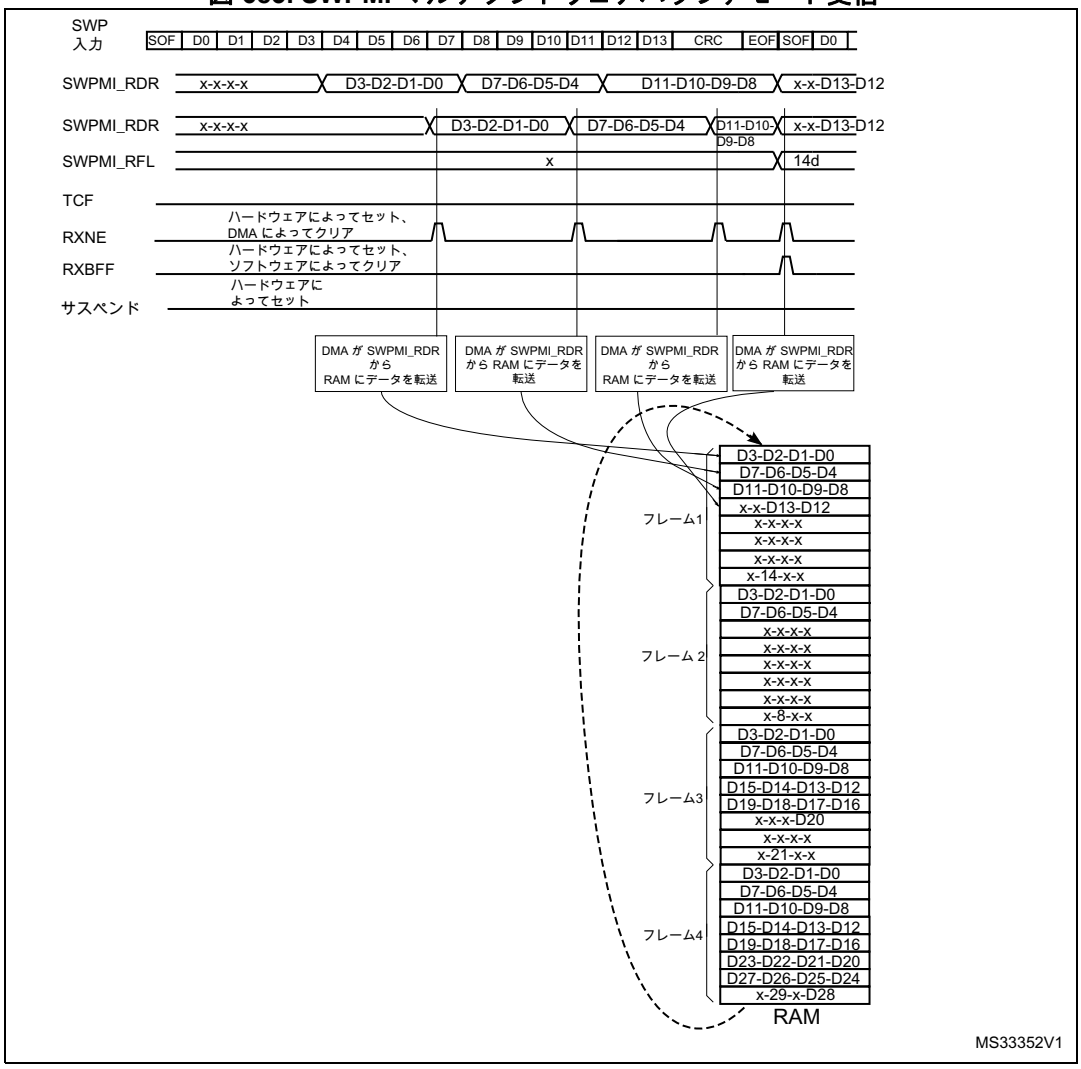
CRC エラーの場合、RXBFF および RXBERF フラグ両方がセットされ、ビット 24 およびビット 26 がセットされます。

オーバーランの場合、オーバーランフラグがセットされ、ビット 25 がセットされます。受信バッファフルフラグは、最後のワード受信中にオーバーランが発生した場合にのみセットされ、ビット 25 およびビット 26 が現在のフレーム受信と次のフレーム受信にセットされます。

ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに受信して DMA 経由で RAM メモリに転送されたフレームを取得できます。たとえば、ソフトウェアが 4 個の受信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で読み出す準備ができていることを示します。

マルチソフトウェアバッファ受信モードでは、ソフトウェアが 8 番目のワードのビット 24、25 および 26 を読み出している場合、各フレーム受信後に RXBERF、RXOVRF、RXBFF フラグをクリアする必要はありません。

図 683. SWPMI マルチソフトウェアバッファモード受信



53.3.10 エラー管理

ペイロード送信中のアンダーラン

フレームペイロードの送信中は、SWPMI_ISR レジスタの TXUNRF フラグによって送信アンダーランが示されます。SWPMI_IER レジスタの TXBUNREIE ビットがセットされている場合は、割り込みが生成されます。

送信アンダーランが発生した場合、SWPMI がペイロード送信を停止し、破損した CRC（送信された最初の CRC バイトの最初のビットが反転）が送信され、EOF になります。DMA が使用されている場合、SWPMI_CR レジスタの TXDMA ビットが自動的にクリアされます。

TXUNRF がセットされている状態での SWPMI_TDR レジスタへの追加の書き込みはすべて無視されます。SWPMI_ICR レジスタの CTXUNRF ビットをセットして、TXUNRF フラグをクリアする必要があります。

ペイロード受信中のオーバーラン

フレームペイロードの受信中は、SWPMI_ISR レジスタの RXOVRF フラグによって受信オーバーランが示されます。受信オーバーランが発生した場合、SWPMI が受信データで SWPMI_RDR を更新しません。受信データは失われます。

受信は EOF まで続行し、オーバーラン状況がなくなると、RXBFF フラグがセットされます。RXBFF フラグがセットされると、ユーザは RXOVRF を確認できます。SWPMI_ICR レジスタの CRXOVRF ビットをセットして、RXBOVRF フラグをクリアする必要があります。

ユーザがオーバーランをただちに検出したい場合、SWPMI_IER レジスタの RXBOVREIE ビットをセットしてオーバーランが発生した直後に割り込みを生成できます。

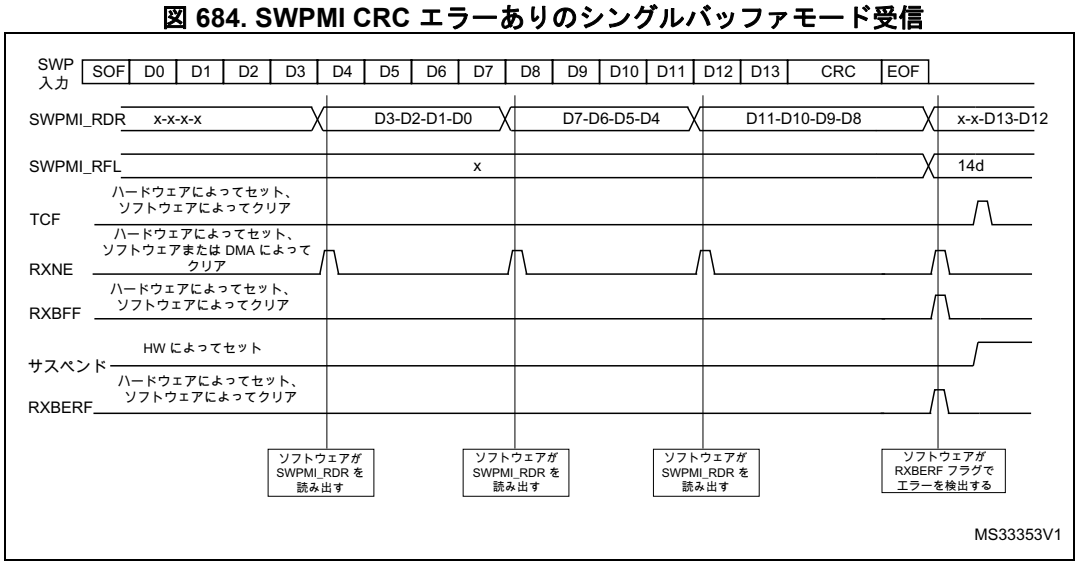
RXOVRF フラグは RXNE フラグと同じタイミングでセットされ、オーバーランイベント後に 2 つの SWPMI_RDR 読み出しが発生します。これは、1 つ以上の受信バイトが失われ、SWPMI_RDR でロードされたワードにオーバーランの直前に受信したバイトが含まれることを示します。

マルチソフトウェアバッファモードでは、RXOVRF フラグが受信フレームの最後のワードに対してセットされると、オーバーランビット（8 番目のワードのビット 25）が現在のフレームと次のフレーム両方にセットされます。

ペイロード受信中の CRC エラー

2 個の CRC バイトを受信して、CRC が誤っている場合、SWPMI_ISR レジスタの RXBERF フラグは EOF 受信後にセットされます。SWPMI_IER レジスタの RXBEIE ビットがセットされている場合、割り込みが生成されます（*図 684 : SWPMI CRC エラーありのシングルバッファモード受信*を参照）。SWPMI_ICR の CRXBERF ビットをセットして RXBERF フラグをクリアする必要があります。

図 684. SWPMI CRC エラーありのシングルバッファモード受信



ペイロード受信中のスタッフィングビットの不足または破損

スタッフィングビットがペイロードで不足または破損している場合、EOF 受信後に SWPMI_ISR で RXBERF および RXBFF フラグがセットされます。

破損した EOF 受信

SOF を受信すると、SWPMI は EOF の受信まで受信バイトを蓄積します（あらゆる SOF は無視）。EOF を受信すると、SWPMI は新しいフレーム受信を開始する準備ができ、SOF を待ちます。

EOF が破損している場合、RXBERF および RXBFF フラグが次の EOF 受信後に SWPMI_ISR レジスタでセットされます。

注： ***EOF 受信が破損している場合、ペイロード受信が続行され、受信バイト数が 30 を超えた場合、ペイロードのバイト数は値 31 を取得する場合があります。ペイロードのバイト数が、動作モードに応じて SWPMI_RFL レジスタまたは RAM メモリのバッファの 8 番目のワードのビット [23:16] に読み出されます。***

53.3.11 ループバックモード

ループバックモードは試験に使用できます。ループバックモードを有効にするには、SWPMI_CR レジスタの LPBK ビットをセットする必要があります。

ループバックモードが有効になると、SWPMI_TX および SWPMI_RX 信号が相互に接続されます。結果として、SWPMI によって送信されたすべてのフレームを再度受信します。

53.4 SWPMI 低電力モード

表 416. 低電力モードが SWPMI に与える影響

モード	説明
SLEEP	影響はありません。SWPMI 割り込みによって、デバイスは SLEEP モードから復帰します。
STOP	swpmi_ker_ck が HSI の場合、スレーブによって発行された SUSPENDED モードからの RESUME によってデバイスを STOP モードからウェイクアップさせられます（*セクション 53.3.1：SWPMI ブロック図*を参照）。
STANDBY	SWPMI は停止します。

53.5 SWPMI 割り込み

すべての SWPMI 割り込みは、NVIC に接続されています。

SWPMI 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で SWPMI 割り込みチャネルを設定し、有効にします。
2. SWPMI を、SWPMI 割り込みを生成するように設定します（SWPMI_IER レジスタを参照）。

表 417. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化制御ビット	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
受信バッファフル	RXBFF	RXBFIE	あり	なし	なし
送信バッファエンプティ	TXBEF	TXBEIE	あり	なし	なし
受信バッファエラー（CRC エラー）	RXBERF	RXBEIE	あり	なし	なし
受信バッファオーバーラン	RXOVRF	RXBOVEREIE	あり	なし	なし
送信バッファアンダーラン	TXUNRF	TXBUNREIE	あり	なし	なし
受信データレジスタノットエンプティ	RXNE	RIE	あり	なし	なし
送信データレジスタフル	TXE	TIE	あり	なし	なし
転送完了フラグ	TCF	TCIE	あり	なし	なし
スレーブレジュームフラグ	SRF	SRIE	あり	あり (1)	なし
トランシーバレディフラグ	RDYF	RDYIE	あり	なし	なし

1. HSI が swpmi_ker_ck に対して選択された場合

53.6 SWPMI レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの *セクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

53.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWP EN	DEACT	Res.	Res.	Res.	Res.	SWP ACT	LPBK	TXMOD E	RXMO DE	TXDMA	RXDMA
				rw	rw					rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWPTEN**：シングルワイヤプロトコル　マスタトランシーバ有効化

このビットは、トランシーバを有効にして、SWPMI で SWPMI_IO を制御するために使用します（*セクション 53.3.3：SWP の初期化と有効化*を参照）。

0：SPWMI_IO ピンが GPIO コントローラによって制御されます。

1：SPWMI_IO トランシーバが SWPMI によって制御されます。

ビット 10 **DEACT**：シングルワイヤプロトコル　マスタインタフェース無効化

このビットは SWP の DEACTIVATED 状態のリクエストに使用されます。このビットをセットすると、スレーブによる RESUME の受信によって SWP が ACTIVATED 状態のままになること以外は SWPACT をクリアしたときと同じ効果があります。

ビット 9:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SWPACT**：シングルワイヤプロトコル　マスタインタフェース有効化

このビットは SWP バスの有効化に使用されます（*セクション 53.3.3：SWP の初期化と有効化*を参照）。

0：SWPMI_IO がプルダウンされ、SWP バスが DEACTIVATED 状態に切り替わります。

1：SWPMI_IO が開放され、SWP バスが SUSPENDED 状態に切り替わります。

SWPACT ビットをセットできるようにするには、DEACT ビットを事前にクリアしておく必要があります。

ビット 4 **LPBK**：ループバックモード有効化

このビットは、ループバックモードの有効化に使用されます。

0：ループバックモードは無効です。

1：ループバックモードは有効です。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 3 **TXMODE**：送信バッファモード

このビットは、送信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（*表 418：送信／受信に対するバッファモードの選択*を参照）。

0：SWPMI が送信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が送信用マルチソフトウェアバッファモードで設定されます。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 2 **RXMODE**：受信バッファモード

このビットは、受信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（*表 418：送信／受信に対するバッファモードの選択*を参照）。

0：SWPMI が受信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が受信用マルチソフトウェアバッファモードで設定されます。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 1 **TXDMA**：送信 DMA 有効化

このビットは、送信時に DMA モードの有効化に使用されます。

0：DMA は送信に無効です。

1：DMA は送信に有効です。

注：　送信されたフレームのペイロードサイズが（フレームの最初のワードの TDR の最下位バイトで）0x00 として指定される場合、TXDMA は自動的にクリアされます。アンダーランイベント時（TXUNRF フラグが SWP_ISR レジスタでセットされる場合）にも自動的にクリアされます。

ビット 0 **RXDMA**：受信 DMA 有効化

このビットは、受信時に DMA モードの有効化に使用されます。

0：DMA は受信に無効です。

1：DMA は受信に有効です。

表 418. 送信／受信に対するバッファモードの選択

バッファモード	ソフトウェアバッファなし	シングルソフトウェアバッファ	マルチソフトウェアバッファ
RXMODE/TXMODE	x	0	1
RXDMA/TXDMA	0	1	1

53.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR）

アドレスオフセット：0x04

リセット値：0x0000 0001

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **BR[7:0]**：ビットレートプリスケーラ

このフィールドは、次の式に応じて　RCC（リセットおよびクロック制御）でプログラムされた $F_{swpmi_ker_ck}$ を考慮しつつ SWP バスのビットレートをプログラムしてセットする必要があります。

$F_{SWP} = F_{swpmi_ker_ck} / ((BR[7:0]+1)x4)$

注：　プログラムされたビットレートは、100 K ビット／秒～2 M ビット／秒の範囲に留める必要があります。

BR[7:0] は、SWPACT ビットが SWPMI_CR レジスタでセットされていると書き込めません。

53.6.3 SWPMI 割り込みおよびステータスレジスタ（SWPMI_ISR）

アドレスオフセット：0x0C

リセット値：0x0000 02C2

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RDYF	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RDYF**：トランシーバレディフラグ

このビットは、トランシーバがレディ状態になった直後に、ハードウェアによってセットされます。SWPMI_CR レジスタの SWPTEN ビットをセットして SWPMI_IO トランシーバを有効にした後、ソフトウェアは SWPACT ビットをセットして SWP バスを有効化する前に、このフラグがセットされるのを待機する必要があります。

0：トランシーバ非レディ状態

1：トランシーバレディ状態

ビット 10 **DEACTF**：DEACTIVATED フラグ

このビットは、DEACTIVATED モードに入るリクエストを確認するステータスフラグです。

0：SWP バスは ACTIVATED または SUSPENDED 状態です。

1：SWP バスは DEACTIVATED 状態です。

DEACT ビットがソフトウェアによってセットされているときに、スレーブによる RESUME 状態が SWPMI によって検出された場合、SRF フラグがセットされ、DEACTF はセットされず、SWP は ACTIVATED 状態で動作します。

ビット 9 **SUSP**：SUSPEND フラグ

このビットは SWP バスの状態をレポートするステータスフラグです。

0：SWP バスは ACTIVATED 状態です。

1：SWP バスは SUSPENDED または DEACTIVATED 状態です。

ビット 8 **SRF**：スレーブレジュームフラグ

このビットは、スレーブによる RESUME が検出されたことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CSRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：スレーブによる RESUME 状態が検出されません。

1：スレーブによる RESUME 状態が SWP バスが SUSPENDED 状態のときに検出されました。

ビット 7 **TCF**：転送完了フラグ

このフラグは、送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わった直後に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信または受信は完了していません。

1：送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わります。

ビット 6 **TXE**：送信データレジスタエンプティ

このフラグは送信データレジスタのステータスを示します。

0：送信データレジスタ SWPMI_TDR に書き込まれたデータはまだ送信されていません。

1：送信データレジスタ SWPMI_TDR に書き込まれたデータが送信され、SWPMI_TDR に再び書き込むことができます。

ビット 5 **RXNE**：受信データレジスタノットエンプティ

このフラグは受信データレジスタのステータスを示します。

0：データは SWPMI_RDR レジスタで受信されていません。

1：SWPMI_RDR レジスタで受信データを読み出す準備ができています。

ビット 4 **TXUNRF**：送信アンダーランエラーフラグ

このフラグは、ペイロード送信中のアンダーラン、つまり SWPMI_TDR がソフトウェアまたは DMA によって時間内に書き込まれなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXUNRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信時にアンダーランエラーはありませんでした。

1：送信時にアンダーランエラーが検出されました。

ビット 3 **RXOVRF**：受信オーバーランエラーフラグ

このフラグは、ペイロード受信中のオーバーラン、つまり SWPMI_RDR がソフトウェアまたは DMA によって時間内に読み出されなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXOVRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時にオーバーランはありませんでした。

1：受信時にオーバーランが検出されました。

ビット 2 **RXBERF**：受信 CRC エラーフラグ

このフラグは、受信フレームでの CRC エラーを示すために、ハードウェアによってセットされます。RXBFF フラグで同期的にセットされます。SWPMI_ICR レジスタの CRXBERF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時に CRC エラーはありませんでした。

1：受信時に CRC エラーが検出されました。

ビット 1 **TXBEF**：送信バッファエンプティフラグ

このフラグは、現在のフレーム送信の完了にこれ以上 SWPMI_TDR の更新が必要ないことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXBEF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：フレーム送信バッファはまだ空ではありません。

1：フレーム送信バッファが空になりました。

ビット 0 **RXBFF**：受信バッファフルフラグ

このフラグは、受信においてフレームの最後のワードが SWPMI_RDR で使用できる場合に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXBFF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信においてフレームの最後のワードがまだ SWPMI_RDR に到達していません。

1：受信においてフレームの最後のワードが SWPMI_RDR に到達しました。

53.6.4 SWPMI 割り込みフラグクリアレジスタ（SWPMI_ICR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CRDYF	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
				rc_w1			rc_w1	rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **CRDYF**：トランシーバレディフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RDYF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CSRF**：スレーブレジュームフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの SRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 7 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TCF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 6:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTXUNRF**：送信アンダーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXUNRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 3 **CRXOVRF**：受信オーバーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBOCREF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 2 **CRXBERF**：受信 CRC エラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBERF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 1 **CTXBEF**：送信バッファエンプティフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXBEF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 0 **CRXBFF**：受信バッファフルフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBFF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

53.6.5 SWPMI 割り込み有効レジスタ（SWMPMI_IER）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RDYIE	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBEIE	TXBEIE	RXBFIE
				rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RDYIE**：トランシーバレディ割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RDYF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRIE**：スレーブレジューム割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの SRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 7 **TCIE**：送信完了割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TCF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 6 **TIE**：送信割り込みイネーブル
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TXE フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 5 **RIE**：受信割り込みイネーブル
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXNE フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 4 **TXUNRIE**：送信アンダーランエラー割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TXBUNRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 3 **RXOVRIE**：受信オーバーランエラー割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXBOVRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 2 **RXBERIE**：受信 CRC エラー割り込み有効化

0：割り込みは禁止されています。

1：SWPMI_ISR レジスタの RXBERF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 1 **TXBEIE**：送信バッファエンプティ割り込み有効化

0：割り込みは禁止されています。

1：SWPMI_ISR レジスタの TXBEF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 0 **RXBFIE**：受信バッファフル割り込み有効化

0：割り込みは禁止されています。

1：SWPMI_ISR レジスタの RXBFF フラグがセットされているときには、SWPMI 割り込みが生成されます。

53.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFL[4:0]				
											r	r	r	r	r

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **RFL[4:0]**：受信フレーム長

RFL[4:0] は、受信フレームのペイロードのデータバイト数です。2 つの最下位ビット RFL[1:0] によって、最後の SWPMI_RDR レジスタの読み出しに関連するバイト数が示されます。

53.6.7 SWPMI 送信データレジスタ（SWPMI_TDR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TD[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TD[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **TD[31:0]**：送信データ

送信されるデータを含みます。

このレジスタに書き込むと、SOF 送信または次のペイロードデータ送信がトリガされ、TXE フラグがクリアされます。

53.6.8 SWPMI 受信データレジスタ（SWPMI_RDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RD[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RD[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RD[31:0]**：受信データ

受信データを含みます。

このレジスタを読み出すと、RXNE フラグがクリアされます。

53.6.9 SWPMI オプションレジスタ（SWPMI_OR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWP_CLASS**：SWP クラス選択

このビットは SWP クラスの選択に使用されます（*セクション 53.3.3：SWP の初期化と有効化*を参照）。

0：クラス C：SWPMI_IO が直接 VDD 電圧を使用してクラス C で動作します。
VDD が [1.62～1.98 V] の範囲内にあるときは、この設定を選択する必要があります。

1：クラス B：SWPMI_IO が内部電圧レギュレータを使用してクラス B で動作します。
VDD が [2.70～3.30 V] の範囲内にあるときは、この設定を選択する必要があります。

ビット 0 **SWP_TBYP**：SWP トランシーババイパス

このビットは、内部トランシーバ（SWPMI_IO）をバイパスして外部トランシーバに接続するために使用されます。

0：内部トランシーバは有効です。SWPMI の外部インタフェースは SWPMI_IO です（SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は GPIO で使用できません）。

1：内部トランシーバは無効です。SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号が GPIO でオルタネート機能として使用できます。外部トランシーバを接続するには、この設定を選択します。

53.6.10 SWPMI レジスタマップとリセット値の表

表 419. SWPMI レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SWPMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWPTEN	DEACT	Res.	Res.	Res.	Res.	SWPACT	LPBK	TXMODE	RXMODE	TXDMA	RXDMA
	リセット値																					0	0					0	0	0	0	0	0
0x04	SWPMI_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR [7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x08	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0C	SWPMI_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDYF	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
	リセット値																					0	0	1	0	1	1	0	0	0	0	1	0
0x10	SWPMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRDYF	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
	リセット値																					0			0	0			0	0	0	0	0
0x14	SWPMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDYIE	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBERIE	TXBEIE	RXBFIE
	リセット値																					0			0	0	0	0	0	0	0	0	0
0x18	SWPMI_RFL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFL[4:0]				
	リセット値																												0	0	0	0	0
0x1C	SWPMI_TDR	TD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SWPMI_RDR	RD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SWPMI_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWP_CLASS	SWP_TBYP
	リセット値																															0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

54 マネジメントデータ インプット/アウトプット（MDIOS）

54.1 MDIOS の概要

マスタチップが 1 つまたは複数のスレーブチップを管理（設定およびステータスデータの取得）を行う必要のあるシステムでは、MDIO バスが役に立ちます。バスプロトコルでは、次の 2 つの信号のみが使用されます。

- MDC：管理データクロック
- MDIO：オペコード（書き込みまたは読み出し）、スレーブ（ポート）アドレス、MDIOS レジスタアドレス、データが伝送されるデータライン

各トランザクションにおいて、マスタは、そのスレーブの中の 1 つの MDIOS レジスタの内容を読み取るか、そのスレーブの中の 1 つの MDIOS レジスタにデータを書き込むかのいずれかを行います。

MDIOS ペリフェラルは、MDIO バスに対するスレーブインタフェースとして機能します。MDIO マスタは、MDC/MDIO ラインを使って、MDIOS に収められた 32 個の 16 ビット MDIOS レジスタの読み書きができます。これらの MDIOS レジスタはファームウェアによって管理されますので、MDIO マスタは STM32 動作中にアプリケーションを設定して、そこからのステータス情報の取得が可能です。

MDIOS は Stop モードで動作可能であり、MDIO マスタがその MDIOS レジスタの 1 つに読み取りや書き込みを行う場合には、必要に応じて STM32 をウェイクアップします。

54.2 MDIOS の主な機能

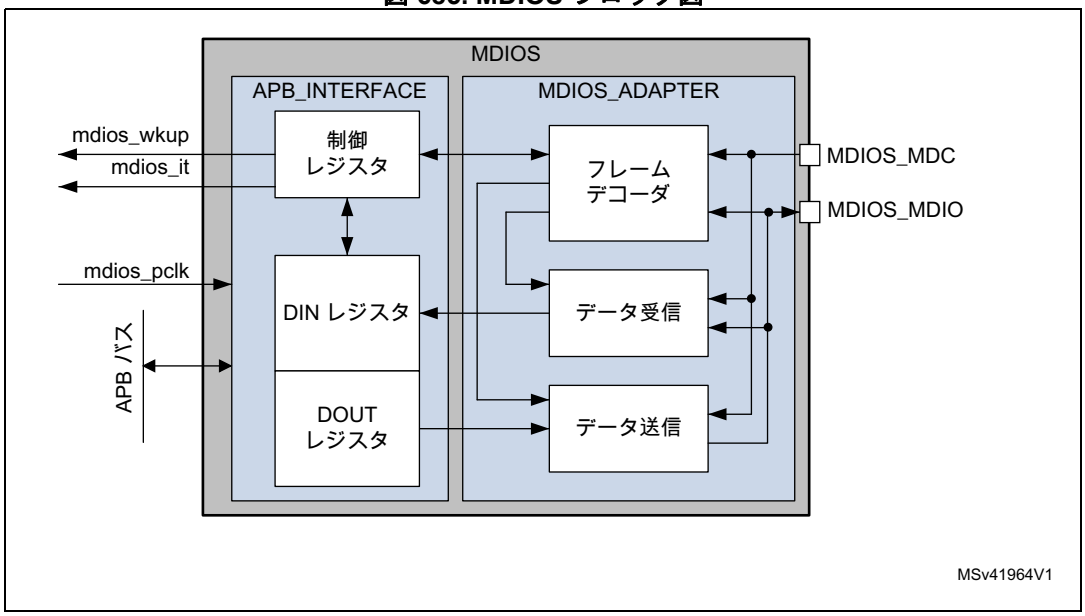
MDIOS には以下の機能があります。

- 入力と出力が独立したデータレジスタを使ってそれぞれ管理される 32 個の MDIOS レジスタアドレス
  - ファームウェアから読み書き可能、MDIOS から読み取り専用である 16 ビット出力データレジス タ 32 個
  - ファームウェアから読み取り専用、MDIOS から書き込み専用である 16 ビット入力データレジス タ 32 個
- 設定可能なスレーブ（ポート）アドレス
- 独立してマスク可能な割り込み／イベント：
  - MDIOS レジスタ書き込み
  - MDIOS レジスタ読み出し
  - MDIOS プロトコルエラー
- STOP モードでの動作とそこからのウェイクアップが可能

54.3 MDIOS の機能説明

54.3.1 MDIOS ブロック図

図 685. MDIOS ブロック図



54.3.2 MDIOS ピンおよび内部信号

*表 420*にパッケージのピンまたはボールに接続された MDIOS 入出力信号のリストを、*表 421*に内部 PWR 信号のリストをそれぞれ示します。

表 420. パッケージのピンまたはボールに接続された MDIOS スレーブ入力／出力信号

信号名	信号タイプ	説明
MDIOS_MDC	デジタル入力	MDIO マスタクロック
MDIOS_MDIO	デジタル入力／出力	MDIO 信号（オペコード、アドレス、入出力データ）

表 421. MDIOS 内部入力／出力信号

信号名	信号タイプ	説明
mdios_wkup	デジタル出力	MDIOS ウェイクアップ信号
mdios_it	デジタル出力	MDIOS 割り込み信号
mdios_pclk	デジタル入力	APB クロック

54.3.3 MDIOS プロトコル

MDIOS プロトコルでは、次の 2 つの信号が使用されます。

1. MDIOS_MDC：常にマスタによってドライブされるクロック
2. MDIOS_MDIO：オペコード、アドレス、双方向データが伝送される信号

それぞれのトランザクションは「フレーム」を使って実行されます。各フレームには次の 32 ビットが含まれます。14 ビットの制御ビット、2 ビットのターンアラウンドビット、16 ビットのデータビット（それぞれシリアル渡し）

- 制御ビット 14 ビット（マスタによりドライブ）
  - スタートビット 2 ビット：常に“01”
  - オペコードビット 2 ビット：読み出し=“10”、書き込み=“01”
  - ポートアドレスビット 5 ビット：どのスレーブデバイスがアドレス指定されているかを示します。
  - MDIOS レジスタアドレスビット 5 ビット：各スレーブ内で最大 32 個の MDIOS レジスタがアドレス指定できます。
- ターンアラウンド状態ビット 2 ビット
  - 書き込み操作では、マスタが“10”をドライブします。
  - 読み出し操作では、最初のビットはハイインピーダンスとなり、2 番目のビットはスレーブによって‘0’にドライブされます。
- データビット 16 ビット
  - 書き込み操作では、マスタによってドライブされたデータがスレーブのMDIOSレジスタに書き込まれます。
  - 読み出し操作では、スレーブによってドライブされたデータがスレーブのMDIOSレジスタから読まれます。

通常、各フレームの先頭にはプリアンブルがあり、MDIOS は MDC クロック 32 個の期間分‘1’のままとなります。送信するフレームがない場合には、マスタは MDIO を‘1’に保ったままとして、「アイドル」状態を示すことができます。

MDIO 信号がマスタによってドライブされる場合には、その信号は MDC の立ち上がりエッジを使って MDIOS にサンプリングされます。MDIOS が MDIO をドライブする場合には、出力は MDC の立ち上がりエッジで変化します。

図 686. MDIO プロトコルのライトフレーム波形

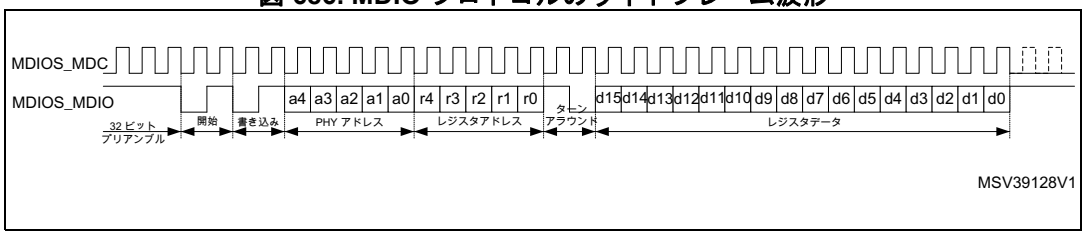
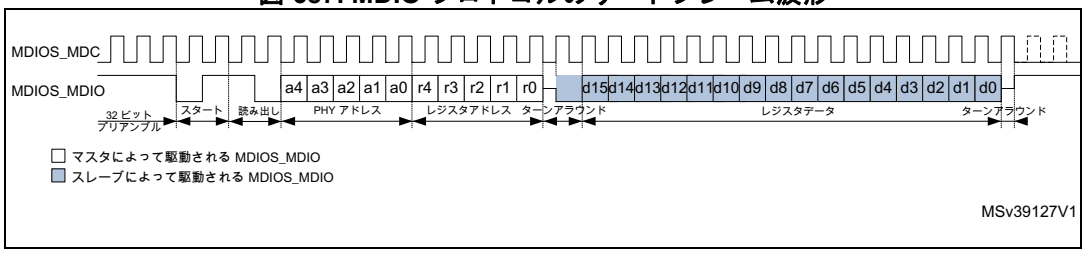


図 687. MDIO プロトコルのリードフレーム波形



54.3.4 MDIOS の有効化／無効化

MDIOSは、MDIOS_CR レジスタの EN ビットをセットすることによって有効になります。EN=1 である場合、MDIOSレジスタの一部にアドレス指定された、MDIO バスのモニタとフレームのサービスをします。

MDIOS が有効になると（EN を‘1’にセット）、MDIOS_CR レジスタへの同じ書き込み操作では、PORT_ADDRESS[4:0] フィールドを適切にセットして､スレーブポートアドレスを示す必要があります。あるフレームのポートアドレスが PORT_ADDRESS[4:0] と同じではない場合（おそらく別のスレーブ宛てのものであるため)、そのフレームは MDIOS から無視されます。

EN=0 である場合、MDIOS は MDC/MDIO ライン上で伝送されているフレームを無視し、IP は消費電力削減モードとなります。EN をクリアすると、DIN レジスタもすべてクリアされます。MDIOS が読み取りデータをドライブしている最中に EN がクリアされると、MDIOS は直ちにバスをリリースし、残りデータのドライブは行いません。MDIOS がフレームを受信中に EN がクリアされると、そのフレームは異常終了してデータは失われます。

MDIOS が有効になり、その次に無効となった後にまた有効となったときには、ステータスフラグはクリアされません。正しい動作のためには、ファームウェアはステータスフラグをクリアしてから MDIOS を再び有効化する必要があります。

54.3.5 MDIOS データ

MDIO マスタから見ると、MDIOS の中には読み書き可能な 16 ビット MDIOS レジスタが 32 個存在します。実際には、それぞれの MDIOS レジスタ‘n’に対して、DINn[15:0] と DOUTn[15:0] という 2 セットのレジスタが存在します。

入力データ

MDIOS の MDIOS レジスタ‘n’に書き込みを行うフレームを MDIO マスタが送信する場合、入力データで更新されるのは DINn[15:0] となります。DIN レジスタ（DIN0～DIN31）はファームウェアによって読み出し可能ですが、書き込みは MDIO バス経由で MDIO マスタによってのみ可能です。

DINn の内容は、最後のデータがサンプリングされる MDC 立ち上がりエッジの直後に変化します。

ファームウェアが、DINn の内容をそれが更新されている瞬間に偶然読み出すと、読み出された値は正しくない可能性があります（元の値と新しい値がビット単位で混在するため）。このため、**ファームウェアは、同じ DINn レジスタからの連続する 2 回の読み取りが必ず同一の値となり、データが読み取り時に安定していたことを確認する必要があります。**最悪の場合には、ファームウェアは DINn を 4 回読み出す必要があります。すなわち、最初に古い値を取得し、2 回目に不整合な値（レジスタが変化した瞬間に読み出した場合）を取得し、3 回目に新しい値を取得し、4 回目に新しい値を確認する場合です。

ファームウェアが WRF 割り込みを使用しており、新しい MDIOS ライトフレームが完成する前に DINn レジスタを読み取ることを保証できるのであれば、ファームウェアはシングルリードを行えます。

31 よりも大きなレジスタアドレスで MDIO マスタが書き込み操作を行うと、MDIOS はそのフレームを無視します（データは保存されず、フラグもセットされません）。

出力データ

レジスタ‘n’の読み取りを要求するフレームを MDIOS が受信すると、DOUTn[15:0] レジスタの値が返されます。したがって、MDIOS レジスタ‘n’に前に書き込んだ同じ値を読み取ることを MDIO マスタから期待されている場合には、新しいデータが DINn に書き込まれるたびに、ファームウェアはそのデータを DINn から DOUTn にコピーする必要があります。正しい操作のため、ファームウェアは、プリアンブル（マスタが各フレームの前にプリアンブルを送信している場合）に 15 サイクルを加えた時間の中で、DOUTn レジスタにデータをコピーする必要があります。

MDIO バスを経由して MDIOS レジスタが読み出される場合には、リードフレームの 15 番目のサイクルの間に、MDIOS は（該当する DOUTn レジスタから）MDIOS クロックドメインに 16 ビットの値を渡します。MDIO マスタが MDIOS レジスタ‘n’を読み出している間にファームウェアが DOUTn レジスタに書き込みを行おうとした場合、フレームの 15 番目のサイクルの間（1 つの MDC サイクルウィンドウ内）にそれが生じると、ファームウェアの書き込み操作は無視されます。したがって、**DOUTn レジスタに書き込みを行った後、ファームウェアは同一の DOUTn レジスタをリードバックして、その値が実際に書き込まれたことを確認する必要があります。**書き込まれた値がDOUTn レジスタに含まれていない場合、ファームウェアは、書き込みを試して再度読み直すだけで構いません。

mdios_pclk 周波数に比べて MDIOS 周波数が非常に遅い場合には、DOUT レジスタに連続して書き込みと読み直しを行って CPU に負荷をかけることは望ましくありません。DOUTn 値が MDIOS クロックドメインに渡されると直ちに、リードフラグ（RDFn）がセットされることに注意してください。したがって、DOUTn への書き込みが無視された場合（リードバックした値が直前に書き込んだ値ではない場合）には、ファームウェアは読み出しの割り込みを使用して DOUTn への書き込みがいつ可能であるか知ることができます。

MDC クロックが非常に遅いときに使用できる手順を次に示します。

1. DOUTn に書き込みます。
2. リードフラグがすべてゼロであることを確認します（MDIOS_RDFR = 0x0000）。必要あれば、MDIOS_CRDFR を使ってフラグをクリアします。
3. 同じ DOUTn レジスタをリードバックして、その値とステップ 1 で書き込まれた値とを比較します。
4. それらの値が同じであれば、この手順は終わりです。そうではない場合には、ステップ 5 に進みます。
5. MDIOS_CR1 の RDIE ビットをセットして、読み出しの割り込みを有効にします。
6. 割り込みルーチンにおいて、RDFn がセットされていることを確認します。(ビット n の前にそれ以外のリードフラグはセットされません）
7. リードバックと比較の実行を必要とせずに安全に DOUTn に書き込むために、31 サイクルにプリアンブルを加えた（マスタが各フレームの前にプリアンブルを送信している場合）時間ウィンドウが存在します。この最大ディレイが保証できない場合には、ステップ 1 に戻ります。

31 よりも大きなレジスタアドレスで MDIO マスタが読み出し操作を行うと、MDIOS はオールゼロの値のデータを返します。

54.3.6 MDIOS APB 周波数

ファームウェアが MDIOS_DINRn レジスタを読み出す際や MDIOS_DOUTRn に書き込む際には、APB バスの周波数は MDC 周波数の 1.5 倍以上である必要があります。たとえば MDC が 20MHz であるときには、APB は 30MHz 以上である必要があります。

54.3.7 ライト／リードフラグおよび割り込み

MDIO バスを経由して MDIOS レジスタ‘n’に書き込みが行われると、MDIOS_WRFR レジスタの WRFn ビットがセットされます。最後のデータビットがライトフレーム上でサンプリングされて DINn が更新される瞬間に、WRFn が‘1’になります。（MDIOS_CR レジスタで）WRIEN=1 である場合、割り込みが生成されます。（MDIOS_CWRFR レジスタで）CWRFn に‘1’を書き込むことによって、WRFn はソフトウェアによってクリアされます。

MDIO バスを経由して MDIOS レジスタ‘n’の読み取りが行われると､MDIOS_RDFR レジスタの RDFn ビットがセットされます。リードフレームの 15 サイクル目の DOUTn が MDC クロックドメインにコピーされた瞬間に、RDFn が‘1’になります。（MDIOS_CR レジスタで）RDIEN=1 である場合、割り込みが生成されます。（MDIOS_CRDFR レジスタで）CRDFn に‘1’を書き込むことによって、RDFn はソフトウェアによってクリアされます。

54.3.8 MDIOS エラー管理

エラーには次の 3 つのタイプがあり、対応するエラーフラグとともに示します。

- プリアンブルエラー：PERF（MDIOS_SR レジスタのビット 0）
- スタートエラー：SERF（MDIOS_SR レジスタのビット 1）
- ターンアラウンドエラー：TERF（MDIOS_SR レジスタのビット 2）

各エラーフラグは、対応するエラー状態が発生したときに、ハードウェアによってセットされます。各フラグは、クリアフラグレジスタ（MDIOS_CLRFR）の対応するビットに‘1’を書き込むことによってクリアできます。

EIE=1（MDIOS_CR）であるときに 3 つのエラーフラグのいずれかがセットされると、割り込みが発生します。

エラーフラグをセットする以外には、エラーが検出されたフレームに対して MDIOS は何の動作も行いません。つまり、DINn レジスタは更新されませんし、データフェーズ中に MDIO ラインが駆動されることもありません。

ある特定のフレームに対して、エラーが累積することはありません。たとえば、プリアンブルエラーが検出されても、現在のフレームの残り部分に対して、スタートエラーまたはターンアラウンドエラーに対するチェックは行われません。

DPC=0 である場合、検出されたエラーの後には、完全なプリアンブルが検出されるまで、新しいフレームとエラーはすべて無視されます。

DPC=1（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）である場合、エラーフラグの 1 つがセットされている限り、すべてのフレームと新しいエラーは無視されます。エラービットがクリアされると、MDIOS は直ちに開始シーケンスを探し始めます。したがって、アプリケーションは、現在進行中のフレームが存在しないことが確実であるときに限って、エラーフラグをクリアする必要があります。そうしないと、MDIOS が送信中のビットを誤って解釈し、マスタとの同期が取れなくなってしまうおそれがあります。

プリアンブルエラー

開始シーケンスが始まり（MDIO は‘0’でサンプリング）、その直後にプリアンブル（連続する 32 クロックの間、MDIO は‘1’でサンプリング）が続かない場合、プリアンブルエラーが発生します。

MDIOS が初めて有効（MDIOS_CR で EN=1）になった後、完全なプリアンブルが受信されるまでは、プリアンブルエラーは通知されません。これは、すでにプリアンブルまたはフレームが進行中であるときにペリフェラルフレーム検出が有効化された場合のエラー状態を回避するためです。この場合、MDIOS は最初のフレームを無視しますが（完全なプリアンブルを最初に検出していないため）、PERF のセットは行いません。

DPC ビット（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）がセットされると、MDIO マスタは先行するプリアンブルなしにフレームの送信が可能となり、プリアンブルエラーの通知は行われません。DPC=1 である場合、アプリケーションは、MDIOS が有効化された（EN がセットされた）瞬間にマスタがフレームの送信途中ではないことを確認する必要があります。そうしないと、スレーブはマスタの同期が取れなくなってしまうおそれがあります。

スタートエラー

不正な開始シーケンスが発生したり、不正なコマンドが与えられたりすると、スタートエラーが発生します。開始シーケンスは必ず“01”でなければならず、コマンドは“01”（書き込み）または“10”（読み出し）のいずれかでなければなりません。

プリアンブルエラーと同様に、完全なプリアンブルが受信されるまで、スタートエラーは通知されません。

ターンアラウンドエラー

ライトフレームのターンアラウンドビットの中でエラーが検出された場合、ターンアラウンドエラーが発生します。ライトフレームの 15 番目のビットは‘1’、16 番目のビットは‘0’である必要があります。

完全なプリアンブルが受信され、スタートエラーがなく、現在のフレームのポートアドレスが一致し、レジスタアドレスが対応範囲の 0 から 31 の中である場合にのみ、ターンアラウンドエラーが通知されます。

54.3.9 STOP モードでの MDIOS

MDIOS は STOP モードで動作可能であり、MDIOS 割り込みに基づいて、すべての読み取りに応答し、すべての書き込みを行い、STM32 を STOP モードからウェイクアップさせます。

54.3.10 MDIOS 割り込み

3 種類の割込み（書き込み、読み出し、エラー）に対して、存在する割り込みベクタは 1 つだけです。これらの割り込みソースのどれでも、STM32 を STOP モードからウェイクアップできます。割り込みラインをクリアするため、すべての割り込みフラグをクリアする必要があります。

表 422. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化 制御ビット
書き込み割り込み	WRF[31:0]	WRIE
読み出し割り込み	RDF[31:0]	RDIE
エラー割り込み	PERF（プリアンブル） SERF（スタート） TERF（ターンアラウンド）	EIE

54.4 MDIOS レジスタ

54.4.1 MDIOS 設定レジスタ（MDIOS_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PORT_ADDRESS[4:0]					DPC	Res.	Res.	Res.	EIE	RDIE	WRIE	EN
			rw	rw	rw	rw	rw	rw				rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **PORT_ADDRESS[4:0]**：スレーブアドレス

書き込めるのは、ペリフェラルが無効なとき（EN=0）だけです。

MDIO マスタから与えられたアドレスが PORT_ADRESS[4:0] と一致する場合に、MDIOS がフレームのサービスを行います。そうでない場合、フレームは無視されます。

ビット 7 **DPC**：ディスエーブルプリアンブルチェック

0：MDIO マスタは各フレームの前にプリアンブルを与える必要があります。

1：MDIO マスタは先行するプリアンブルを付けずに各フレームを送信可能であり、MDIOS はプリアンブルエラーを通知しません。

このビットがセットされている場合、アプリケーションは、MDIOS が有効化されたときに現在進行中のフレームが存在しないことが確実でなければなりません。そうしないと、MDIOS はマスタの同期が取れなくなってしまうことがあります。

EN=0 ではない限り、このビットは変更できません（ただし、EN のセットと同時の変更は可能です）。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **EIE**：エラー割り込みは有効です。

0：割り込みは無効です

1：割り込みは有効です

このビットがセットされている場合、いずれかのエラーフラグ（MDIOS_SR レジスタの PERF、SERF、TERF）がセットされると割り込みが生成されます。

ビット 2 **RDIE**：レジスタ読み出し割り込みイネーブル

0：割り込みは無効です

1：割り込みは有効です

このビットがセットされている場合、いずれかのリードフラグ（MDIOS_RDFR レジスタの RDF[31:0]）がセットされると割り込みが生成されます。

ビット 1 **WRIE:** レジスタ書き込み割り込みイネーブル

0：割り込みは無効です

1：割り込みは有効です

このビットがセットされている場合、いずれかのリードフラグ（MDIOS_WRFR レジスタの WRF[31:0]）がセットされると割り込みが生成されます。

ビット 0 **EN**：ペリフェラルイネーブル

0：MDIOS は無効です。

1：MDIOS は有効であり、MDIO バス（MDC/MDIO）の監視を行っています。

54.4.2 MDIOS ライトフラグレジスタ（MDIOS_WRFR）

アドレスオフセット：0x04

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **WRF[31:0]**：MDIOS レジスタ 0～31 のライトフラグ

各ビットは、対応する MDIOS レジスタに MDIO マスタが書き込みを行うと、ハードウェアによってセットされます。（MDIOS_CRで）WRIE がセットされると、割り込みが生成されます。

各ビットは、MDIOS_CWRFR レジスタの対応する CWRF ビットに‘1’を書き込むことによって、ソフトウェアによってクリアされます。

WRFn に対して：

0：MDIOS レジスタ‘n’は MDIO マスタによって書き込まれていません。

1：MDIOS レジスタ‘n’は MDIO マスタによって書き込まれていて、そのデータは MDIOS_DINRn レジスタの DINn[15:0] にあります。

54.4.3 MDIOS クリアライトフラグレジスタ（MDIOS_CWRFR）

アドレスオフセット：0x08

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CWRF[31:16]															
w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CWRF[15:0]															
w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0

ビット 31:0 **CWRF[31:0]**：ライトフラグのクリア

CWRFn に‘1‘を書き込むと、MDIOS_WRF レジスタの WRFn ビットがクリアされます。

54.4.4 MDIOS リードフラグレジスタ（MDIOS_RDFR）

アドレスオフセット：0x0C

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDF[**31**:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RDF[31:0]**：MDIOS レジスタ 0～31 のリードフラグ

各ビットは、対応する MDIOS レジスタから MDIO マスタが読み出しを行うと、ハードウェアによってセットされます。（MDIOS_CR）の RDIE がセットされると、割り込みが生成されます。

各ビットは、MDIOS_CRDFR レジスタの対応する CRDF ビットに‘1’を書き込むことによって、ソフトウェアによってクリアされます。

RDFn に対して：

0：MDIOS レジスタ‘n’は MDIO マスタによって読み出されていません。

1：MDIOS レジスタ‘n’は MDIO マスタによって読み出されています。

54.4.5 MDIOS クリアリードフラグレジスタ（MDIOS_CRDFR）

アドレスオフセット：0x10

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRDF[31:16]															
w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRDF[15:0]															
w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0	w_r0

ビット 31:0 **CRDF[31:0]**：リードフラグのクリア

CRDFn に‘1‘を書き込むと、MDIOS_RDF レジスタの RDFn ビットがクリアされます。

54.4.6 MDIOS ステータスレジスタ（MDIOS_SR）

アドレスオフセット：0x14

パワーオンリセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERF	SERF	PERF
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 TERF：ターンアラウンドエラーフラグ

0：発生したターンアラウンドエラーはありません。

1：ターンアラウンドエラーが発生しました。

CTERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

ビット 1 SERF：スタートエラーフラグ

0：発生したスタートエラーはありません。

1：スタートエラーが発生しました。

CSERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

ビット 0 PERF：プリアンブルエラーフラグ

0：発生したプリアンブルエラーはありません。

1：プリアンブルエラーが発生しました。

CPERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

DPC（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）がセットされると、このビットはセットされません。

注： ***MDIOS_SR への書き込みは無効です。***

54.4.7 MDIOS クリアフラグレジスタ（MDIOS_CLRFR）

アドレスオフセット：0x18

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTERF	CSERF	CPERF
													w_r0	w_r0	w_r0

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CTERF**：ターンアラウンドエラーフラグのクリア

このビットに 1 を書き込むと、TERF フラグ（MDIOS_SR）がクリアされます。

DPC='1'（MDIOS_CR[7]）である場合、すでに進行中のフレームが存在しない場合にのみ TERF フラグをクリアする必要があります。

ビット 1 **CSERF**：スタートエラーフラグのクリア

このビットに 1 を書き込むと、SERF フラグ（MDIOS_SR）がクリアされます。

DPC='1'（MDIOS_CR[7]）である場合、すでに進行中のフレームが存在しない場合にのみ SERF フラグをクリアする必要があります。

ビット 0 **CPERF**：プリアンブルエラーフラグのクリア

このビットに 1 を書き込むと、PERF フラグ（MDIOS_SR）がクリアされます。

注： ***MDIOS_CLRFR を読み出すとオールゼロが返されます。***

54.4.8 MDIOS 入力データレジスタ（MDIOS_DINR0-MDIOS_DINR31）

アドレスオフセット：0x100～0x17C

リセット値：0x0000_0000

31	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DINn[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **DINn[15:0]**：ライトフレーム中に MDIO マスタから受信した入力データ

このフィールドには、MDIOS レジスタ‘n’にアドレス指定されたライトフレームで受信した 16 ビットデータがハードウェアによって書き込まれます。

54.4.9 MDIOS 出力データレジスタ（MDIOS_DOUTR0-MDIOS_DOUTR31）

アドレスオフセット：0x180～0x1FC

リセット値：0x0000_0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOUTn[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **DOUTn[15:0]**：リードフレーム中に MDIO マスタに送信された出力データ

このフィールドはソフトウェアによって書き込まれます。MDIOS レジスタ‘n’をアドレス指定したリードフレーム中に、これらの 16 ビットが MDIO バスにシリアル出力されます。

参考資料

54.4.10 MDIOS レジスタマップ

表 423. MDIOS レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	MDIOS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PORT_ADDRESS[4:0]					Res.	Res.	Res.	Res.	EEI	RDIE	WRIE	EN
	リセット値																				0	0	0	0	0					0	0	0	0
0x04	MDIOS_WRFR	WRF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	MDIOS_CWRFR	CWRF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	MDIOS_RDFR	RDF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	MDIOS_CRDFR	CRDF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	MDIOS_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERF	SERF	PERF
	リセット値																														0	0	0
0x18	MDIOS_CLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTERF	CSERF	CPERF
	リセット値																														0	0	0
0x1C～0xFC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	MDIOS_DINR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN0[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x104	MDIOS_DINR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...																																	
0x17C	MDIOS_DINR31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN31[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x180	MDIOS_DOUTR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT0[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x184	MDIOS_DOUTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...																																	
0x1FC	MDIOS_DOUTR31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT31[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

55.1 SDMMC の主な機能

SD/SDIO、MultiMediaCard（MMC）ホストインタフェース（SDMMC）は、AHB バスと SD メモリカード、SDIO カード、および MMC デバイス間のインタフェースを提供します。

MMCA 技術委員会から発行された MultiMediaCard システム仕様書は、MultiMediaCard Association のウェブサイト www.mmca.org から入手できます。（訳注：現在はwww.jedec.orgから入手できます）

SD メモリカードと SD I/O カードシステム仕様書は、SD card Association のウェブサイト www.sdcard.org から入手できます。

SDMMC には以下の機能があります。

- *MultiMediaCard System Specification Version 4.51* に完全準拠。
  3 種類の異なるデータバスモードのカードに対応：1 ビット（デフォルト）、4 ビットおよび 8 ビット
- 以前のバージョンの MultiMediaCard との完全互換（下位互換性）
- *SD memory card specifications version 4.1* に完全準拠。
  （SDR104 SDMMC_CK 速度は最大許容 I/O 速度に制限され、SPI モードおよび UHS-II モードはサポートされません）
- *SDIO card specification version 4.0* に完全準拠。
  2 種類の異なるデータバスモードのカードに対応：1 ビット（デフォルト）および 4 ビット
  （SDR104 SDMMC_CK 速度は最大許容 I/O 速度に制限され、SPI モードおよび UHS-II モードはサポートされません）
- 8 ビットモードで最大 208 MB/秒のデータ転送
  （最大許容 I/O 速度に基づきます）
- 外部双方向ドライバを制御するデータおよびコマンド出力有効信号

MultiMediaCard/SD バスは、カードをホストに接続します。

現行バージョンの SDMMC は、1 度に 1 枚の SD/SDIO/MMC カードと、MMC バージョン 4.51 以前のスタックだけをサポートします。

55.2 SDMMC バストポロジ

バス上の通信は、コマンド／レスポンスとデータ転送に基づいて行われます。

SD/SDIO/MMC バス上の基本的なトランザクションは、コマンド／レスポンストランザクションです。これらの種類のバストランザクションでは、コマンドやレスポンスの決められた通信フォーマットでカードの情報を直接転送します。また、データトークンを持つ操作もあります。

データ転送は次の方法に従って行います。

- ブロックモード：N の範囲 0～14 でブロックサイズ 2^N バイトのデータブロック
- SDIO マルチバイトモード：ブロックサイズ範囲 1～512 バイトの単一データブロック
- MMC ストリームモード：連続的なデータストリーム

MMC カードに対するデータ転送は、データブロックまたはストリーム単位で行われます。

参考資料

図 688. SDMMC「no response」と「no data」操作

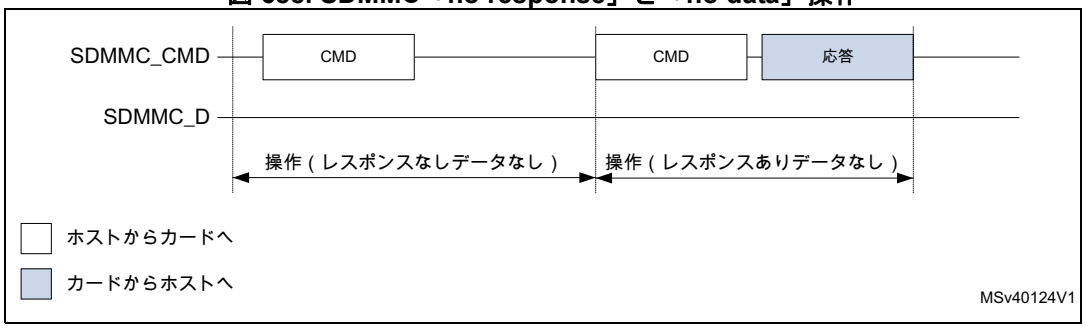
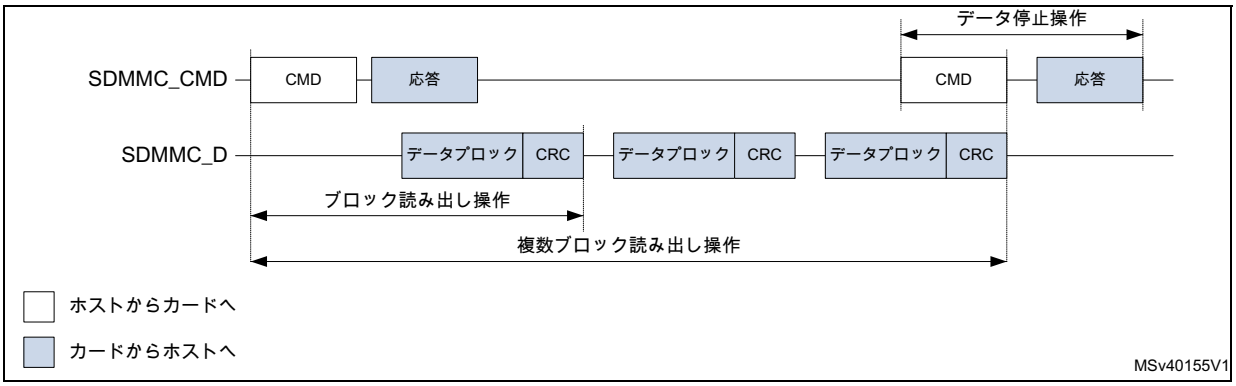
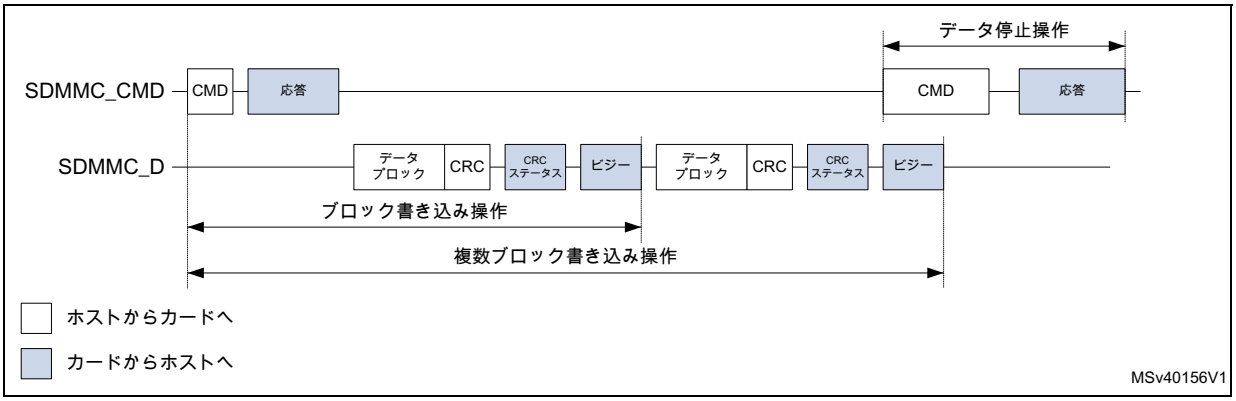


図 689. SDMMC（複数）ブロック読み出し操作



注：　事前定義されたブロック数での MMC 複数ブロック読み出し終了時に、送信停止コマンドは必要ありません。

図 690. SDMMC（複数）ブロック書き込み操作



注：　事前定義されたブロック数での MMC 複数ブロック書き込み終了時に、送信停止コマンドは必要ありません。

注：　SDMMC は、ビジー信号がアサートされている限り（SDMMC_D0 をロー）、データを一切送信しません。

図 691. SDMMC（シーケンシャル）ストリーム読み出し操作

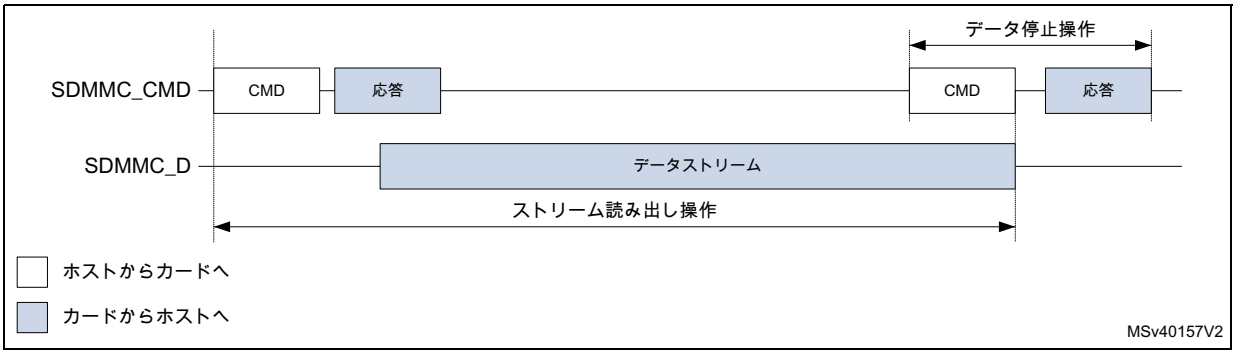
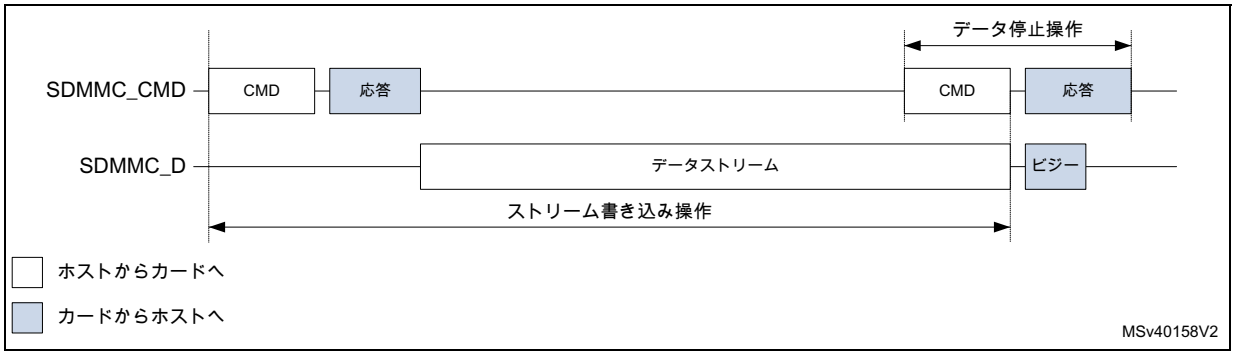


図 692. SDMMC（シーケンシャル）ストリーム書き込み操作



ストリームデータ転送は、シングルデータレートモード（DS、ハイスピード および SDR）の SDMMC_D0 で 1 ビット幅のビットバス設定でのみ動作します。

55.3 SDMMC 動作モード

表 424. SDMMC 動作モード SD および SDIO

SDIO バス速度モード(1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz]	信号電圧 [V]
DS（デフォルトスピード）	12.5	25	3.3
HS（ハイスピード）	25	50	3.3
SDR12	12.5	25	1.8
SDR25	25	50	1.8
DDR50	50	50	1.8
SDR50	50	100	1.8
SDR104	104	208(4)	1.8

1. SDR シングルデータレートシグナリング
2. DDR ダブルデータレートシグナリング（データが両方の SDMMC_CK クロックエッジでサンプリングされます）
3. 4 ビットバス幅での SDIO バス速度
4. 最大許容 I/O 速度に基づく最大周波数

SDR104 モードでは、サンプリングポイントチューニングを使用した可変遅延サポートが必要です。SDR50 モードでは、可変遅延の使用はオプションです。

表 425. SDMMC 動作モード eMMC

eMMC バス速度モード (1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz]	信号電圧 [V]
レガシー互換	26	26	3/1.8/1.2V
ハイスピード SDR	52	52	3/1.8/1.2V
ハイスピード DDR	104	52	3/1.8/1.2V
ハイスピード HS200	200	200(4)	1.8/1.2V

1. SDR シングルデータレートシグナリング
2. DDR ダブルデータレートシグナリング（データが両方の SDMMC_CK クロックエッジでサンプリングされます）
3. 8 ビットバス幅での eMMC バス速度
4. 最大許容 I/O 速度に基づく最大周波数

55.4 SDMMC の機能詳細

SDMMC は、3 つの部分で構成されています。

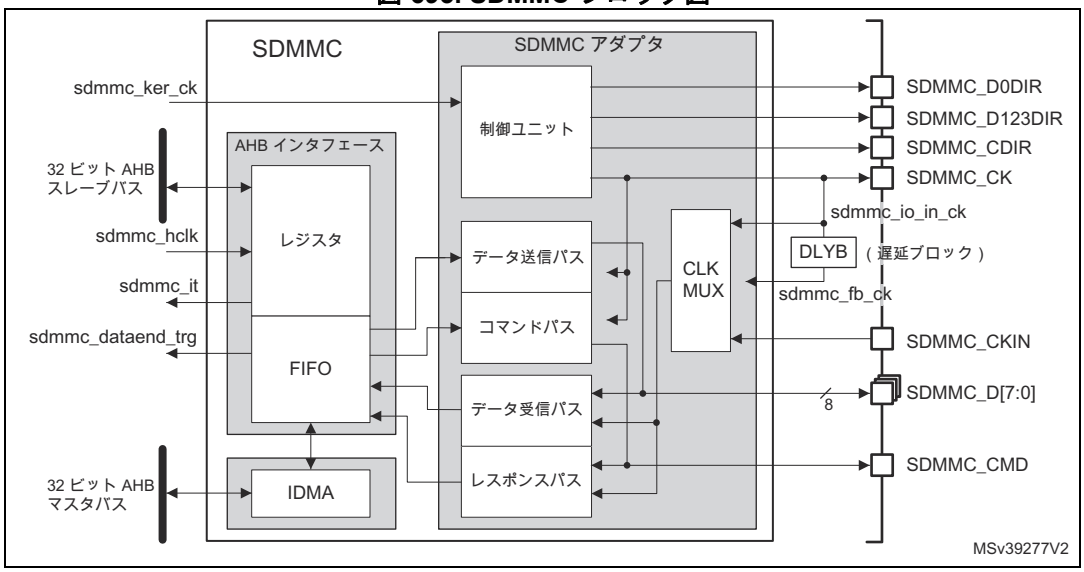
- AHB スレーブインタフェースは、SDMMC アダプタレジスタにアクセスして、割り込み信号および IDMA 制御信号を生成します。
- SDMMC アダプタブロックは、クロック生成ユニット、コマンドおよびデータ転送など、MMC/SD/SD I/O カードに固有のすべての機能を提供します。
- 内部 DMA（IDMA）ブロックは AHB マスタインタフェースを搭載しています。

デバイスの DLYB 遅延ブロックは、SDMMC で受信したデータでサンプリングクロックを調整するために使用できます。SDMMC では、SDR104 モードをサポートする必要があります。

55.4.1 SDMMC ブロック図

図 693 に SDMMC ブロック図を示します。

図 693. SDMMC ブロック図



55.4.2 SDMMC ピンおよび内部信号

表 426 では SDMMC 内部入力／出力信号を、*表 427* では SDMMC ピン（オルタネート機能）を一覧化しています。

表 426. SDMMC 内部入力／出力信号

信号名	信号タイプ	説明
sdmmc_ker_ck	デジタル入力	SDMMC カーネルクロック
sdmmc_hclk	デジタル入力	AHB クロック
sdmmc_it	デジタル出力	SDMMC グローバル割り込み
sdmmc_dataend_trg	デジタル出力	MDMA の SDMMC データエンドトリガ
sdmmc_io_in_ck	デジタル入力	SD/SDIO/MMC カードフィードバッククロック。この信号は、SDMMC_CK ピンに内部接続されます（DS および ハイスピードモードの場合）。
sdmmc_fb_ck	デジタル入力	DLYB 遅延ブロック後の SD/SDIO/MMC カード調整フィードバッククロック（SDR50、DDR50、SDR104 の場合）

表 427. SDMMC ピン群

信号名	信号タイプ	説明
SDMMC_CK	デジタル出力	SD/SDIO/MMC カードへのクロック
SDMMC_CKIN	デジタル入力	SD/SDIO/MMC カードの外部ドライバからのクロックフィードバック（SDR12、SDR25、SDR50、DDR50 の場合）
SDMMC_CMD	デジタル入出力	SD/SDIO/MMC カードの双方向コマンド／レスポンス信号
SDMMC_CDIR	デジタル出力	SD/SDIO/MMC カードの SDMMC_CMD 信号の I/O 方向指示
SDMMC_D[7:0]	デジタル入出力	SD/SDIO/MMC カードの双方向データライン
SDMMC_D0DIR	デジタル出力	SD/SDIO/MMC カードの SDMMC_D0 データラインの I/O 方向指示
SDMMC_D123DIR	デジタル出力	SD/SDIO/MMC カードのデータライン SDMMC_D[3:1] の I/O 方向指示

55.4.3 概要

SDMMC_D[7:0] ラインには異なる動作モードがあります。

- デフォルトでは、SDMMC_D0 ラインはデータ転送に使用されます。初期化後、ホストはデータバス幅を変更することができます。
- MMC では、1 ビット（SDMMC_D0）、4 ビット（SDMMC_D[3:0]）、または 8 ビット（SDMMC_D[7:0]）のデータバス幅を使用できます。
- SD カードまたは SDIO カードでは、1 ビット（SDMMC_D0）または 4 ビット（SDMMC_D[3:0]）を使用できます。すべてのデータラインはプッシュプルモードで動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、データラインでのデータフローの方向を I/O 方向信号で示します。**SDMMC_D0DIR** 信号は SDMMC_D0 データライン、**SDMMC_D123DIR** は SDMMC_D[3:1] データラインの I/O 方向を示します。

SDMMC_CMD はプッシュプルモードでのみ動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、SDMMC_CMD ラインでのデータフローの方向を I/O 方向信号 **SDMMC_CDIR** で示します。

カードへの **SDMMC_CK** クロックは **sdmmc_ker_ck** から生成されます。

- sdmmc_ker_ck クロックが 50% デューティサイクルの場合、バイパスモードでも使用できます（CLKDIV = 0）。
- sdmmc_ker_ck クロックのデューティサイクルが 50% ではない場合、2 以上で分周するために CLKDIV を使用する必要があります（CLKDIV > 0）。
- SDMMC_CMD / SDMMC_D[7:0] 出力と SDMMC_CK の間の位相関係は、NEGEDGE ビットで選択できます。この位相関係は、CLKDIV、NEGEDGE および DDR の設定に依存します。*図 694* を参照してください。

図 694. SDMMC コマンドおよびデータの位相関係

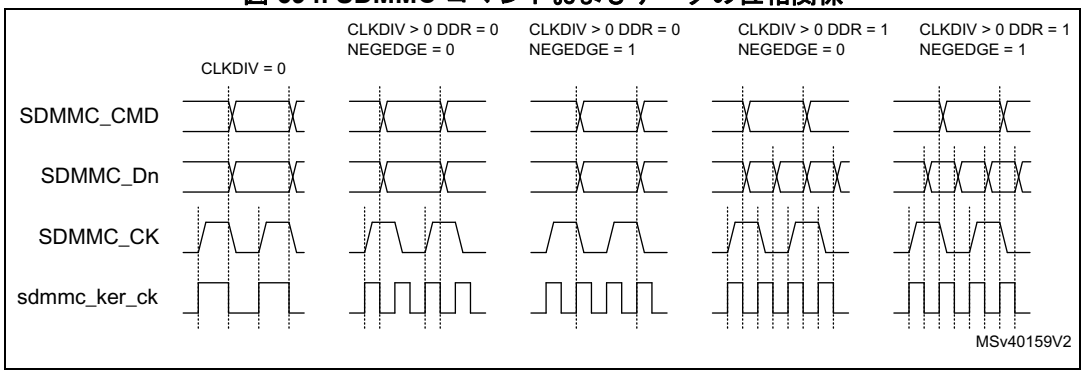


表 428. SDMMC コマンドおよびデータの位相選択

CLKDIV	DDR	NEGEDGE	SDMMC_CK	コマンド出力	データ出力
0	x	x	= sdmmc_ker_ck	sdmmc_ker_ck 立ち下がりエッジで生成	
>0	0	0	sdmmc_ker_ck 立ち上がりエッジで生成	SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで生成	
		1		SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで生成	
	1	0		SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで生成	SDMMC_CK エッジに続く sdmmc_ker_ck 立ち下がりエッジで生成
		1		SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで生成	

デフォルトでは、**sdmmc_io_in_ck** フィードバッククロック入力が、SDMMC 受信パスで受信データをサンプリングするために選択されます。これは SDMMC_CK ピンから生成されます。

サンプリングクロックの位相をチューニングして受信データタイミングを調整するために、デバイスで使用できる DLYB 遅延ブロックを **sdmmc_io_in_ck** 信号（DLYB 入力 dlyb_in_ck）と SDMMC の **sdmmc_fb_ck** クロック入力（DLYB 出力 dlyb_out_ck）の間に接続できます。受信パスで **sdmmc_fb_ck** クロック入力を選択すると、受信データの位相調整サンプリングクロックを使用できるようになります。これは、SDMMC で SDR104 動作モードをサポートするために必要であり、SDR50 モードおよび DDR50 モードではオプションです。

外部ドライバ（電圧スイッチトランシーバ）を使用する場合、SDMMC_CKIN フィードバッククロック入力を選択して受信データをサンプリングできます。

SD/SDIO/MMC カードの場合、クロック周波数は 0～208 MHz の範囲（最大 I/O 速度により制限）で可変です。

選択したバスモード（SDR または DDR）に応じて、各クロックサイクルで SDMMC_D[7:0] ラインに 1 ビットまたは 2 ビットが転送されます。SDMMC_CMD ラインは各クロックサイクルで 1 ビットだけ転送します。

55.4.4 SDMMC アダプタ

SDMMC アダプタ（*図 693 : SDMMC ブロック図*を参照）は、MultiMediaCard スタックやセキュアデジタルメモリカードとのインタフェースを提供するマルチメディア／セキュアデジタルメモリカードバスマスタです。これは次のサブユニットで構成されています。

- 制御ユニット
- データ送信パス
- コマンドパス
- データ受信パス
- レスポンスパス
- 受信データパスクロックマルチプレクサ
- アダプタレジスタブロック
- データ FIFO
- 内部 DMA（IDMA）

注： ***アダプタレジスタと FIFO は、AHB クロックドメイン（sdmmc_hclk）を使用します。制御ユニット、コマンドパス、およびデータ送信パスは、SDMMC アダプタクロックドメイン（sdmmc_ker_ck）を使用します。レスポンスパスとデータ受信パスは、sdmmc_io_in_ck、SDMMC_CKIN、または DLYB によって生成された sdmmc_fb_ck からの SDMMC アダプタフィードバッククロックドメインを使用します。***

デバイスの DLYB 遅延ブロックは、SDMMC アダプタと連携して、SDMMC 受信モードでサンプリングクロックの位相を調整してデータを受信するために使用できます。これは、SDMMC で SDR104 動作モードをサポートするために必要であり、SDR50 モードおよび DDR50 モードではオプションです。

アダプタレジスタブロック

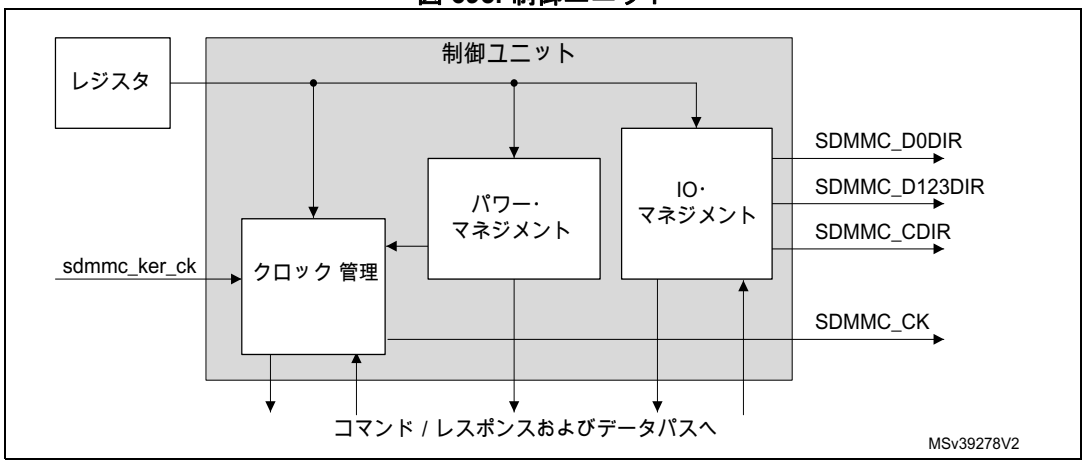
アダプタレジスタブロックには、すべてのシステム制御レジスタ、SDMMC コマンドおよびレスポンスレジスタ、データ FIFO が含まれています。

このブロックは、SDMMC アダプタのスタティックフラグをクリアする SDMMC クリアレジスタの対応するビット位置からの信号も生成します。

制御ユニット

図 695 に示された制御ユニットには、電源管理機能、分周回路での SDMMC_CK クロック管理、I/O 方向管理が含まれています。

図 695. 制御ユニット



電源管理サブユニットは、パワーオフおよびパワーアップフェーズの間、カードバス出力信号を無効にします。

3 つの電源フェーズがあります。

- パワーオフ
- パワーアップ
- パワーオン

クロック管理サブユニットは、sdmmc_ker_ck を使用して SDMMC_CK を生成し、分周制御を提供します。また、フロー制御などで SDMMC_CK を停止することもできます。

クロック出力は、次の場合にインアクティブです。

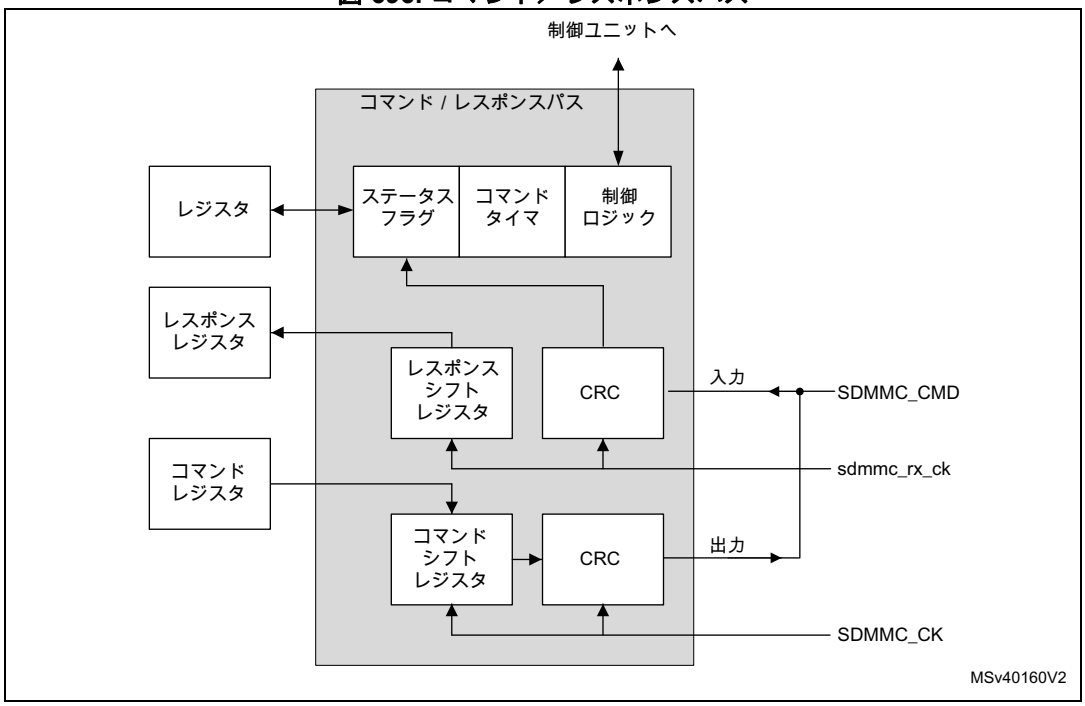
- リセット後
- パワーオフまたはパワーアップフェーズ中
- 節電モード（レジスタビット PWRSAV）が有効であり、カードバスが 8 クロック周期の間アイドル状態の場合。クロックは、コマンド／レスポンス CPSM およびデータパス DPSM サブユニット両方がアイドルフェーズに移行してから 8 サイクル後に停止します。クロックは、コマンド／レスポンス CPSM またはデータパス DPSM がアクティブになった（有効になった）ときに再開します。

I/O 管理サブユニットは、SDMMC_Dn および SDMMC_CMD の I/O 方向信号を管理し、外部電圧トランシーバを制御します。

コマンド／レスポンスパス

コマンド／レスポンスパスサブユニットは、SDMMC_CMD ラインでコマンドとレスポンスを転送します。コマンドパスは、SDMMC_CK でクロック供給され、カードにコマンドを送信します。レスポンスパスは、sdmmc_rx_ck でクロック供給され、カードからレスポンスを受信します。

図 696. コマンド／レスポンスパス

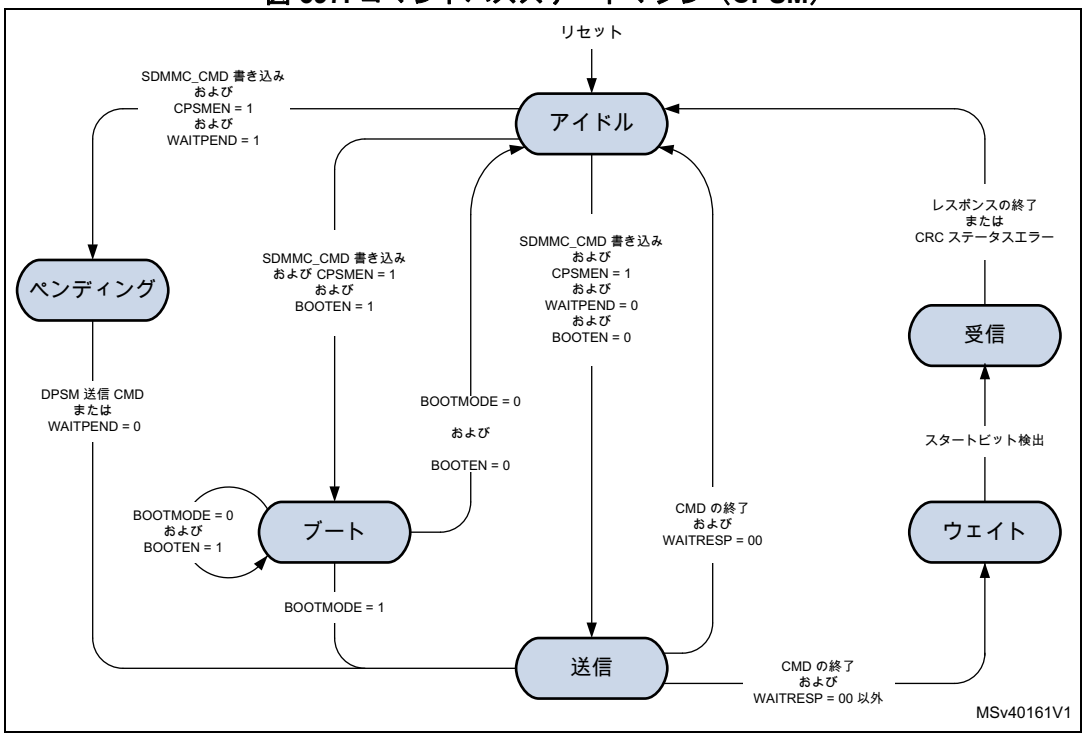


コマンド／レスポンスパスステートマシン（CPSM）

- コマンドレジスタが書き込まれて、有効ビットがセットされると、コマンド転送が開始します。コマンドが送信されると、CRC が付加され、コマンドパスステートマシン（CPSM）がステータスフラグをセットします。
  - レスポンスが必要ではない場合は、アイドル状態に入ります。
  - レスポンスが必要な場合は、レスポンスを待ちます。
- レスポンス受信時：
  - CRC 付きのレスポンスの場合、受信した CRC コードと内部生成されたコードを比較し、結果に応じて適切なステータスフラグをセットします。
  - CRC なしのレスポンスの場合、CRC はチェックされず、適切なステータスフラグがセットされません。

アイドル状態ではない場合など CPSM がアクティブであるときは必ず CPSMACT ビットがセットされます。

図 697. コマンドパスステートマシン（CPSM）



- **アイドル**：コマンドパスがインアクティブです。コマンド制御レジスタが書き込まれ、有効ビット（CPSMEN）がセットされると、CPSM が SDMMC_CK クロック（節電 PWRSAV ビットのために停止している場合）を有効化して次の状態に移行します。
  - WAITPEND = 0 かつ BOOTEN = 0 の場合は送信状態
  - WAITPEND = 1 の場合はペンディング状態
  - BOOTEN = 1 の場合はブート状態
- **送信**：コマンドが送信され、CRC が付加されます。
  - CMDTRANS ビットがセットされている場合、または BOOTEN ビットがセットされている場合、かつ BOOTMODE が代替ブートで、DTDIR = 受信の場合、コマンド終了時に CPSM DataEnable 信号が DPSM に発行されます。
  - CMDTRANS ビットがセットされている場合、かつ CMDSUSPEND ビットが 0 の場合、コマンド終了時に割り込み期間が終了します。
  - CMDSTOP ビットがセットされている場合、コマンド終了時に CPSM Abort 信号が DPSM に発行されます。
  - レスポンスが期待されていない（WAITRESP = 00）場合、CPSM はアイドル状態に移行し、CMDSENT フラグを生成します。BOOTMODE = 1 かつ BOOTEN = 0 の場合、CMDSENT フラグがコマンド終了ビット後 56 サイクル遅延するか、コマンド終了ビット直後に CMDSENT フラグが生成されます。
    RESPCMDR レジスタおよび RESPxR レジスタは変更されません。
  - コマンドレスポンスが期待されている（WAITRESP が 00 ではない）場合、CPSM はウェイトステートに移行し、レスポンスタイムアウトを開始します。

- **ウェイト**：コマンドパスがレスポンスを待ちます。
  - WAITINT ビットが 0 の場合、コマンドタイマが動作を開始し、CPSM がスタートビットを待ちます。
  a) スタートビットがタイムアウトまでに検出された場合、CPSM は受信状態に移行します。
  b) CPSM がレスポンススタートビットを検出する前にタイムアウトに達した場合には、タイムアウトフラグ（CTIMEOUT）がセットされ、CPSM がアイドル状態に移行します。RESPCMDR レジスタおよび RESPxR レジスタは変更されません。
  - WAITINT ビットが 1 の場合、タイマは無効になり、CPSM は1枚のカードからの割り込みリクエスト（レスポンススタートビット）を待ちます。
  a) スタートビットが検出された場合、CPSM は受信状態に移行します。
  b) WAITINT を 0 で書き込む（割り込みモードをアボートする）と、ホストは自身でレスポンスを送信し、スタートビット検出時に CPSM が受信状態に移行します。
- **受信**：コマンドレスポンスが受信されます。コマンド制御レジスタのレスポンスモードビット WAITRESP に応じて、CRC の有無によりレスポンスを短くしたり長くしたりできます。受信した CRC コードがある場合、内部で生成された CRC コードに対して確認されます。
  - CMDSUSPEND ビットがセットされている場合、かつ SDIO レスポンス ビット BS = 0（レスポンスビット [39]）の場合、レスポンスの後に割り込み期間が開始します。
  CMDSUSPEND ビットがクリアされている場合、または CMDSUSPEND ビットが 1 の場合、かつ SDIO レスポンス ビット BS = 1（レスポンスビット [39]）の場合、割り込み期間は開始しません。
  - CMDTRANS ビットと CMDSUSPEND ビットがセットされている場合、かつ SDIO レスポンス ビット DF = 1（レスポンスビット [32]）の場合、レスポンスの後に割り込み期間が終了します。
  - CRC ステータスがパスするか、CRC が存在しない場合、CMDREND フラグがセットされ、CPSM がアイドル状態に移行します。
  RESPCMDR レジスタおよび RESPxR レジスタは受信したレスポンスで更新されます。
  - BOOTMODE = 1 かつ BOOTEN = 0 の場合、CMDREND フラグがレスポンス終了ビット後 56 サイクル遅延するか、レスポンス終了ビット直後に CMDREND フラグが生成されます。
  - CMDTRANS ビットがセットされている場合、かつ DTDIR = 送信の場合、コマンドレスポンス終了時に CPSM DataEnable 信号が DPSM に発行されます。
  - CRC ステータスが失敗となる場合、CCRCFAIL フラグがセットされ、CPSM がアイドル状態に移行します。
  RESPCMDR レジスタおよび RESPxR レジスタは受信したレスポンスで更新されます。
- **ペンディング**：コマンドレジスタのペンディング WAITPEND ビットに応じて、CPSM はペンディング状態に入ります。
  - DATALENGTH =< 5 バイトの場合、CPSM は送信状態に入り、DataEnable 信号を生成して CMD12 送信停止コマンドで調整されたデータ転送を開始します。
  - DATALENGTH > 5 バイトの場合、データ転送を開始するために CPSM DataEnable 信号が DPSM に発行されます。CPSM は DPSMからのsendCMD 信号を待ってから、送信状態に入ります。これにより、CMD12 送信停止コマンドなどをデータで調整して送信できます。
  - WAITPEND を 0 で書き込むと、CPSM は送信状態に移行します。
- **ブート**：コマンドレジスタの BOOTEN ビットがセットされていると、CPSM は次の場合にブート状態に入ります。
  - BOOTMODE = 0 で SDMMC_CMD ラインがローに駆動される場合、かつ CMDTRANS ビットがセットされ、DTDIR = 受信の場合、CPSM DataEnable 信号が DPSM に発行されます。これで通常のブート動作が可能になります。この状態は、レジスタビット BOOTEN をクリアすることでブート手順の終了時に配置され、SDMMC_CMD ラインをハイに駆動させ、ア

イドル状態に移行する前に CPSM Abort 信号が DPSM に発行されます。CMDSENT フラグは SDMMC_CMD ラインがハイになった 56 サイクル後に生成されます。

– BOOTMODE = 1 の場合、送信状態へ移行します。これで CMD0（ブート）の送信が可能になります。BOOTEN をクリアしても影響はありません。

注： ***CPSM は、N_{CC} および N_{RC} タイミング制約を満たすために、少なくとも 8 SDMMC_CK 周期の間、アイドル状態を保ちます。N_{CC} は、2 つのホストコマンド間の最小遅延であり、N_{RC} は、ホストコマンドとカードレスポンスの間の最小遅延です。***

注： ***レスポンスタイムアウトは、64 SDMMC_CK クロック周期の固定値です。***

コマンドは、動作を開始するトークンです。コマンドは、ホストから 1 枚のカード（アドレス付きコマンド）または接続されたすべてのカードに送信されます（ブロードキャストコマンドは MMC V3.31 以前で使用可能）。コマンドは、SDMMC_CMD ライン上で連続的に転送されます。すべてのコマンドは、固定長 48 ビットです。SD メモリカード、SDIO カード、MMC カードのコマンドトークンの一般的なフォーマットを*表 429.* に示します。

コマンドトークンデータは 2 つのレジスタから取得され、1 つは 32 ビットの引数を含み、もう一方は 6 ビットのコマンドインデックス（カードに送信される 6 ビット）が含まれます。

表 429. コマンドトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	1	送信ビット
[45:40]	6	x	コマンドインデックス
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

コマンドデータの隣にはコマンドタイプ（WAITRESP）ビットがあり、コマンドパスステートマシン（CPSM）を制御します。これらのビットによって、コマンドがレスポンスを必要とするかどうかと、レスポンスが短い（48 ビット）か長い（136 ビット）か、CRC が存在するかどうかも決まります。

レスポンスは、以前に受信したコマンドへの応答として、アドレス指定されたカードまたはすべてのカードから同時にホストに送信されるトークンです。すべてのレスポンスはコマンドライン SDMMC_CMD 経由で送信されます。レスポンスの送信は、常に、レスポンスコードワードに対応するビット文字列の左のビットから始まります。コード長は、レスポンスのタイプに依存します。レスポンストークン R1、R2、R3、R4、R5、R6 には、コンテンツに応じてさまざまなコード体系があります。SD メモリカード、SDIO カード、MMC カードのレスポンストークンの一般的なフォーマットを*表 430*、*表 431*、*表 432* に示します。

レスポンスは、常にスタートビット（常に 0）から始まり、送信方向を示すビットが続きます（card=0）。下の表で x で示されている値は、可変エントリを示します。一部を除くほとんどのレスポンスは、CRC によって保護されます。すべてのコマンドコードワードは、終了ビット（常に 1）で終わります。

レスポンストークンデータは 5 つのレジスタに格納され、4 つは 32 ビットカードステータス、OCR レジスタ、引数または内部 CRC を含んだ 127 ビット CID や CSD レジスタを含み、1 つは 6 ビットコマンドインデックスを含みます。

表 430. CRC ありショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み 111111）
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

表 431. CRC なしショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み 111111）
[39:8]	32	x	引数
[7:1]	7	1111111	（予約済み 1111111）
0	1	1	終了ビット

表 432. CRC ありロングレスポンスのトークンフォーマット

ビット位置	幅	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	111111	予約済み
[127:1]	127:8	x	CID または CSD スライス
	7:1	x	CRC7（CID、CSD レジスタに含まれる）
0	1	1	終了ビット

コマンド／レスポンスパスは、コマンドを送信したりレスポンスを受信したりできるように半二重モードで動作します。CPSM が送信状態でない場合、SDMMC_CMD 出力はハイインピーダンス状態になります。SDMMC_CMD で送信されたデータは、NEGEDGE レジスタビットに応じて SDMMC_CK と同期します。*図 694* を参照してください。

コマンドおよび CRC ありのショートレスポンスでは、CRC 生成回路は、CRC コードの前のすべての 40 ビットについて CRC チェックサムを計算します。これには、スタートビット、送信ビット、コマンドインデックス、およびコマンド引数（またはカードステータス）が含まれます。

ロングレスポンスでは、R2 CID または CSD の 120 ビットを超える場合のみ CRC チェックサムを計算します。スタートビット、送信ビット、および 6 つの予約ビットは、CRC 計算では使用されませんので注意してください。

CRC チェックサムは 7 ビットの値です。

CRC[6:0] = 剰余 [(M(x) * x^7) / G(x)]

G(x) = $x^7 + x^3 + 1$

M(x) =（最初のビット）* x^n +（2 番目のビット）* x^{n-1}+...+（CRC 前の最後のビット）* x^0

n = 39 または 119 の場合。

CPSMEN がセットされている場合、CPSM は数多くの固有コマンドを送信して、さまざまな動作モードに対応できます。*表 433* を参照してください。

表 433. 固有コマンドの概要

VSWITCH	BOOTEN	BOOTMODE	CMDTRANS	WAITPEND	CMDSTOP	WAITINT	説明
1	x	x	x	x	x	x	電圧スイッチシーケンス開始
0	1	x	x	x	x	x	通常ブート開始
0	1	1	x	x	x	x	代替ブート開始
0	0	1	x	x	x	x	代替ブート停止
0	0	0	1	x	x	x	関連付けしたデータ転送でコマンド送信
0	0	0	0	1	1	x	MMC ストリームデータ転送、データ転送終了までコマンド（STOP_TRANSMISSION）ペンディング
0	0	0	0	1	0	x	MMC ストリームデータ転送、データ転送終了まで（STOP_TRANSMISSION）ペンディング以外のコマンド
0	0	0	0	0	1	x	コマンド送信（STOP_TRANSMISSION）、進行中のデータ送信を停止
0	0	0	0	0	0	1	MMC ウェイト割り込み（Wait-IRQ）モードへ移行
0	0	0	0	0	0	0	その他の固有コマンドなし

コマンド／レスポンスパスは、*表 434* に示すようにステータスフラグおよび関連付けられたクリアビットを実装します。

表 434. コマンドパスステータスフラグ

フラグ	説明
CMDSENT	レスポンスなしのコマンド終了時にセット。（CPSM が送信からアイドルに移行）
CMDREND	CRC が OK のときにコマンドレスポンス終了時にセット。（CPSM が受信からアイドルに移行）
CCRCFAIL	CRC が FAIL のときにコマンドレスポンス終了時にセット。（CPSM が受信からアイドルに移行）
CTIMEOUT	タイムアウト前に受信したレスポンススタートビットがないときにコマンド後にセット。（CPSM がウェイトからアイドルに移行）
CKSTOP	CRC が OK かつ SDMMC_CK が停止したときに電圧スイッチ（VSWITCHEN = 1）コマンドレスポンス後にセット。（CPSM に影響なし）
VSWEND	5ms + 1ms の電圧スイッチ（VSWITCH = 1）タイムアウト後にセット。（CPSM に影響なし）
CPSMACT	コマンド転送中です。（CPSM がアイドル状態ではない）

コマンドパスエラー処理は、*表 438* に示します。

表 435. コマンドパスエラー処理

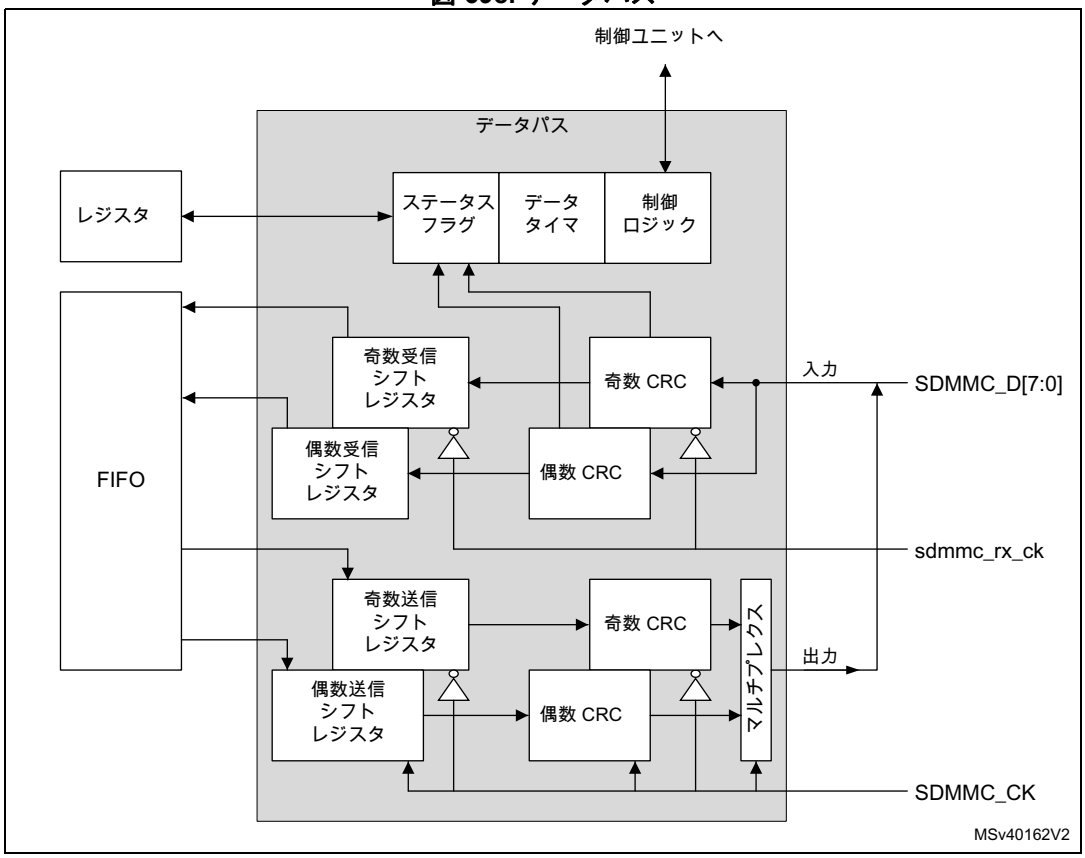
エラー	CPSM の状態	原因	カード動作	ホスト動作	CPSM 動作
タイムアウト	ウェイト	時間内にスタートビットなし	不明	リセットまたはサイクルパワーカード(1)	アイドルに移行
CRC ステータス	受信	否定ステータス	コマンド無視	コマンド再送信 (1)	アイドルに移行
		送信エラー	コマンド許可	コマンド再送信 (1)	

1. CMDTRANS がセットされている場合、stop_transmission コマンドも送信され、DPSM がアイドルに移行します。

データパス

データパスサブユニットは、カードとの間で SDMMC_D[7:0] ラインでデータ転送を行います。データ送信パスは、SDMMC_CK でクロック供給され、カードにデータを送信します。データ受信パスは、sdmmc_rx_ck でクロック供給され、カードからデータを受信します。*図 698* にデータパスブロック図を示します。

図 698. データパス



カードデータバス幅は、クロック制御レジスタビット WIDBUS でプログラムできます。サポートされるデータバス幅モードは次のとおりです。

- ワイドバスモードが有効でない場合は、SDMMC_D0 上で 1 ビットだけが転送されます。
- 4 ビット幅バスモードが有効な場合、データは SDMMC_D[3:0] 上で、4 ビットで転送されます。
- 8 ビット幅バスモードが有効な場合、データは SDMMC_D[7:0] 上で、8 ビットで転送されます。

データバス幅の次に、データサンプリングモードはクロック制御レジスタビット DDR でプログラムできます。サポートされるデータサンプリングモードは次のとおりです。

- シングルデータレートシグナリング（SDR）では、データはクロックの立ち上がりエッジでクロック供給されます。
- ダブルデータレートシグナリング（DDR）では、データはクロックの両方のエッジでクロック供給されます。DDR モードはワイドバスモード（4 ビット幅および 8 ビット幅）でのみサポートされます。

注： ***データサンプリングモードは、SDMMC_D[7:0] ラインのみに適用されます。(SDMMC_CMD ラインには適用できません)。***

DDR モードでは、次のルールに応じて SDMMC_CK の両方のエッジでデータがサンプリングされます。*図 699* および*図 700* も参照してください。

- クロックの立ち上がりエッジで奇数バイトがサンプリングされます。
- クロックの立ち下がりエッジで偶数バイトがサンプリングされます。
- データのペイロードサイズは常に 2 バイトの倍数になります。
- 2 つの CRC16 がデータラインごとに計算されます。
  - クロックの立ち下がりエッジでクロック供給された奇数ビット CRC16
  - クロックの立ち上がりエッジでクロック供給された偶数ビット CRC16
- スタートビット、終了ビットおよびアイドル状態がフルサイクルになります。
- CRC ステータス／ブート確認応答およびビジーシグナリングがフルサイクルになり、クロックの立ち上がりエッジでのみサンプリングされます。

DDR モードでは、SDMMC_CK クロック分周は 2 以上とします。

図 699. DDR モードのデータパケットクロック供給

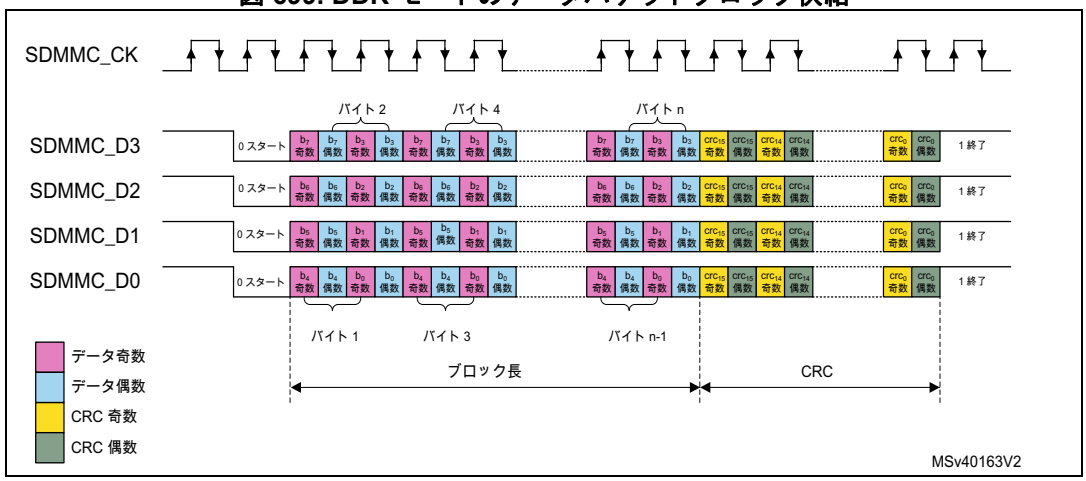
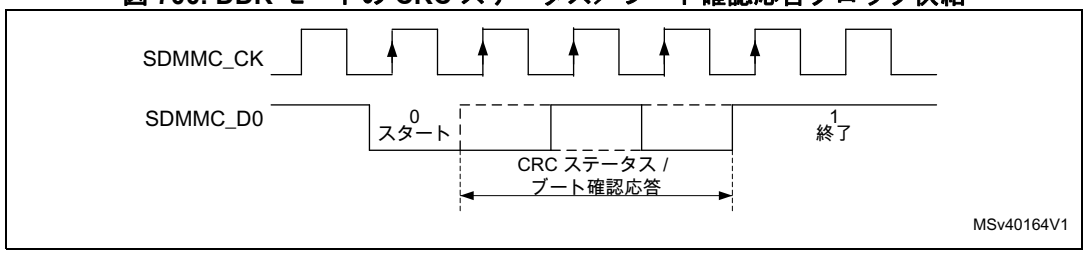


図 700. DDR モードの CRC ステータス／ブート確認応答クロック供給



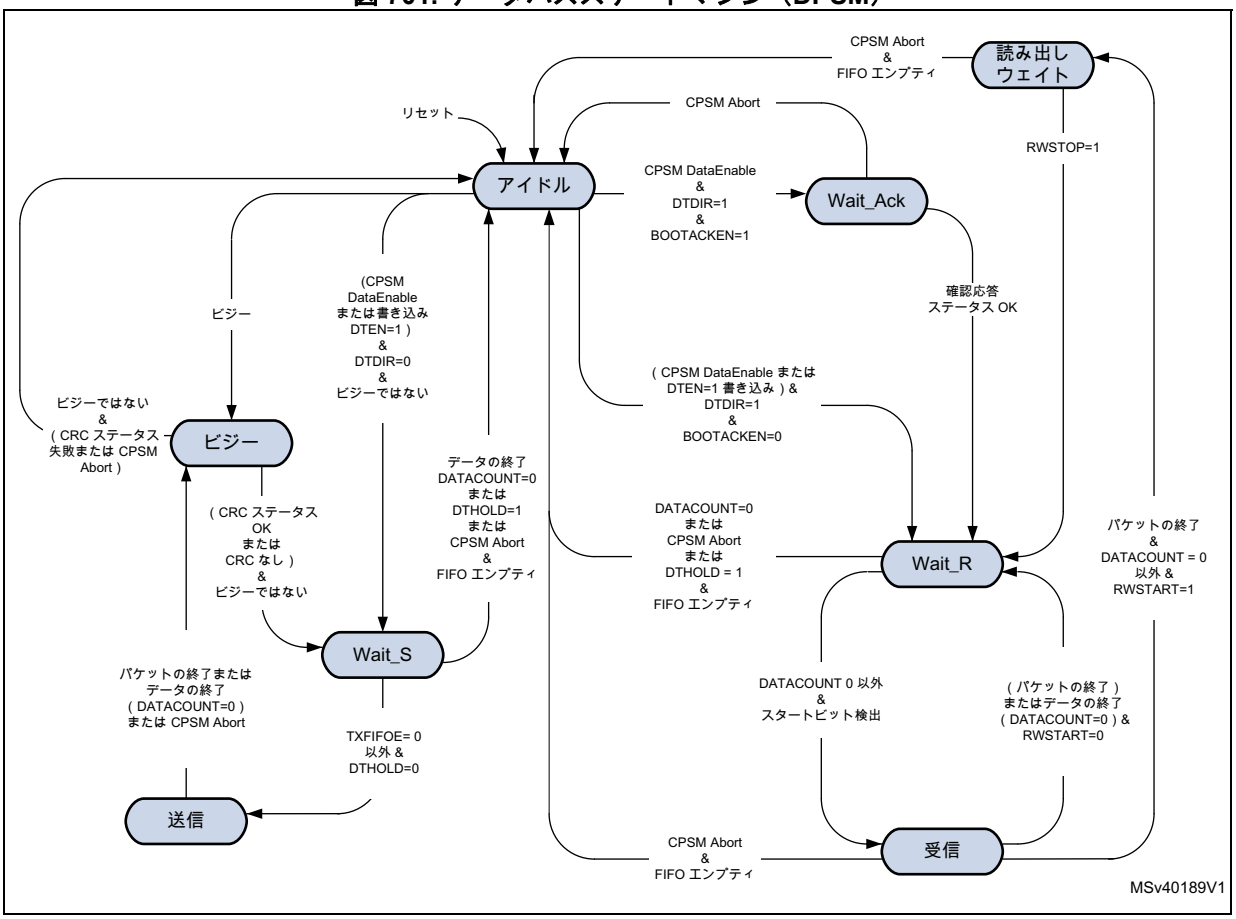
データパスステートマシン（DPSM）

転送の方向（送信または受信）によって、データパスステートマシン（DPSM）は、有効になったときに Wait_S または Wait_R 状態へ移行します。

- 送信 : DPSM は Wait_S 状態へ移行します。送信 FIFO にデータがある場合、DPSM は送信状態へ移行し、データパスサブユニットはカードへのデータ転送を開始します。
- 受信 : DPSM は Wait_R 状態へ移行して、スタートビットを待ちます。スタートビットを受信すると、DPSM は受信状態へ移行し、データパスサブユニットはカードからのデータ受信を開始します。
  確認応答ありのブート動作の場合、DPSM は Wait_Ack 状態に移行し、ブート確認応答を待ってから Wait_R 状態に移行します。

DPSM は、SDMMC_CK で動作します。*図 701* に示すように、DPSM には、次の状態があります。アイドル状態ではない場合など DPSM がアクティブであるときは必ず DPSMACT ビットがセットされます。

図 701. データパスステートマシン（DPSM）



- **アイドル**状態 : データパスはインアクティブであり、SDMMC_D[7:0] 出力は PWRCTRL の設定に従います。CMDTRANS ビットをセットしてコマンドを送信するか、DTEN ビットをセットするか、SDMMC_D0 でビジーを検出すること（R1b レスポンスありのコマンドの後）で DPSM が有効化されます。

  ビジーではない場合、DPSM は SDMMC_CK クロック（節電 PWRSAV ビットのために停止している場合）を有効化し、新しい（DATALENGTH）の値でデータカウンタをロードします。

– データ方向ビット（DTDIR）が送信を示している場合、Wait_S に移行します。
– データ方向ビット（DTDIR）が受信を示している場合、次のように移行します。
- BOOTACKEN レジスタビットがクリアされている場合は Wait_R になります。
- BOOTACKEN レジスタビットがセットされている場合は、Wait_Ack となり、確認応答タイムアウトが開始します。

ビジーの場合、DPSM は SDMMC_CK クロックをアクティブなままとし、ビジー状態に移行します。

注： ***DTEN は SD、SDIO および eMMC カードでデータ転送の開始に使用しません。***

- **Wait_Ack** 状態：データパスがブート確認応答トークンを待ちます。
  – DPSM がタイムアウト前にエラーなし確認応答を受信した場合、Wait_R 状態に移行します。
  – 確認応答とは異なるパターンを受信した場合、確認応答ステータスエラーが生成され、確認応答失敗ステータスフラグ（ACKFAIL）がセットされます。DPSM は、Wait_Ack のままとなります。
  – スタートビットを検出する前にタイムアウト（ACKTIME）に達した場合、タイムアウトステータスフラグ（ACKTIMEOUT）をセットします。DPSM は、Wait_Ack のままとなります。
  – CPSM Abort 信号がセットされている場合、アイドル状態へ移行し、DABORT フラグをセットします。
- **Wait_R** 状態：データカウンタがゼロでなく、データが保持されない場合、データパスは SDMMC_D[n:0] 上のスタートビットを待ちます。データカウンタがゼロで、データを保持している場合、FIFO が空になるのを待ちます。
  – ブロックモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DBLOCKSIZE でデータブロックカウンタをロードします。
  – SDIO マルチバイトモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DATALENGTH でデータブロックカウンタをロードします。
  – ストリームモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DATALENGTH でデータカウンタをロードします。
  – データカウンタ（DATACOUNT）がゼロ（データの終了）に等しい場合、DPSM は受信 FIFO が空になると、アイドル状態へ移行し、DATAEND フラグがセットされます。
  – スタートビットを検出する前にタイムアウト（DATATIME）に達した場合、タイムアウトステータスフラグ（DTIMEOUT）をセットし、DPSM は WAIT_R 状態のままとなります。
  – CPSM Abort 信号がセットされている場合：
  DATACOUNT > 0 の場合、FIFO が空になったときと IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
  DATACOUNT がゼロの場合、通常動作を続け、転送が正常に完了するため、DABORT フラグはセットされません。
  – DTHOLD ビットがセットされている場合：
  - DATACOUNT > 0 の場合、受信 FIFO が空になったときと IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行し、DHOLD フラグが発行されます。保持状態ではタイムアウトは無効になります。保持状態で CPSM Abort 信号を受信すると、転送はアボートされます。
  - DATACOUNT = 0 の場合、正常に転送が完了し、DHOLD フラグは発行されません。
  – DPSM が DTEN で開始した場合、エラー（DTIMEOUT）後、FIFO が空になったときと、IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行します。

- **ReadWait** 状態：データパスがバスの読み出しを待ちます。
  - ReadWait 停止ビット（RWSTOP）がセットされている場合、DPSM は Wait_R 状態に移行し、受信タイムアウトを開始します。
  - CPSM Abort 信号がセットされ、FIFO が空になり、IDMAEN = 0 の場合、FIFORST でリセットされるのを待つ場合、アイドル状態に移行し、DABORT フラグがセットされます。
- **受信**状態：データパスがカードからシリアルデータを受信します。データは数バイトにまとめられ、データ FIFO に書き込まれます。データ制御レジスタで選択された転送モード（DTMODE）に応じて、データ転送はブロックモードまたはストリームモードのいずれかとなります。
  - ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を受信すると、DPSM は CRC コードを受信するまで待ちます。
  - SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を受信すると、DPSM は CRC コードを受信するまで待ちます。

  a) 受信した CRC コードが内部生成された CRC コードと一致する場合、DPSM は次の状態へ移行します。
  - RWSTART= 0 の場合、Wait_R 状態となり、受信タイムアウトを開始します。
  - RWSTART= 1 かつ DATACOUNT > 0 の場合、ReadWait 状態となり、DBCKEND フラグを生成します。

  b) 受信した CRC コードが内部生成された CRC コードと一致しない場合、追加のデータ受信は阻止されます。
  - すべてのデータを受信していない場合（DATACOUNT > 0）、CRC 失敗ステータスフラグ（DCRCFAIL）をセットして、DPSM は受信状態のままとなります。
  - すべてのデータを受信した場合（DATACOUNT = 0）、FIFO が空になるのを待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）をセットして、DPSM はアイドル状態に移行します。

  - ストリームモードでは、DPSM はデータカウンタ DATACOUNT > 0 の間、データを受信します。カウンタがゼロになると、シフトレジスタ内の残りのデータがデータ FIFO に書き込まれ、DPSM は Wait_R 状態へ移行します。
  - FIFO オーバーランエラーが発生した場合、DPSM は FIFO オーバーランエラーフラグ（RXOVERR）をセットして、追加のデータ受信は阻止されます。DPSM は受信状態のままとなります。
  - CPSM_Abort 信号を受信した場合：
    - DATACOUNT = 0 でデータの最後の 2 ビットまでに CPSM_Abort 信号を受信した場合、転送がアボートされます。シフトレジスタに残っているデータがデータ FIFO に書き込まれ、FIFO が空になって、IDMAEN = 0 の場合、FIFORST でリセットされるのを待ってから、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
    - DATACOUNT = 0 で転送の最後の 2 ビット処理中または以降に CPSM_Abort 信号を受信した場合、転送は正常に完了します。DPSM は受信状態のままとなり、DABORT フラグは生成されません。
  - DPSM が DTEN で開始した場合、エラー（DATACOUNT > 0 時の DCRCFAIL または RXOVERR）後、FIFO が空になったときと、IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行します。
- **Wait_S** 状態：データパスが FIFO からのデータが使用できるようになるのを待ちます。
  - データカウンタ DATACOUNT > 0 の場合、データ FIFO エンプティフラグ（TXFIFOE）がネゲートされるまで待ってから、DTHOLD はセットされず、送信状態へ移行します。
  - データカウンタ（DATACOUNT）が 0 の場合、DPSM はアイドル状態に移行し、DATAEND フラグを生成します。

– DTHOLD がセットされ、DATACOUNT > 0 の場合
- IDMA が有効である場合、FIFO は一掃され、DPSM はアイドル状態に移行し、DHOLD フラグを発行します。(DBCKEND フラグもセットされ、DHOLD フラグと同時にクリアすることができます)。
- IDMA が無効である場合、DBCKEND フラグが生成され、FIFO が FIFORST でリセットされるのを待って、DPSM はアイドル状態に移行し、DHOLD フラグを発行します。

– DTHOLD がセットされ、DATACOUNT = 0 の場合、転送は正常に完了します。

– CPSM Abort 信号を受信した場合
- CPSM_Abort 信号を DATACOUNT = 0 でデータの最後の 2 ビットまでに受信した場合、転送がアボートされ、FIFO が空になり、IDMAEN = 0 の場合、FIFORST でリセットされるのを待ってから、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
- DATACOUNT = 0 で転送の最後の 2 ビット処理中または以降に CPSM_Abort 信号を受信した場合、通常動作を続け、転送が正常に完了するため、DABORT フラグはセットされません。

注： ***DPSM は、N_{WR} タイミングの要件を満たすため、少なくとも 2 クロック周期の間、Wait_S 状態を保ちます。ここで、N_{WR} は、カードレスポンスの受信とホストからのデータ転送の開始との間のクロックサイクル数です。***

- **送信**状態：DPSM はカードへのデータの送信を開始します。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモード、SDIO マルチバイトモードまたはストリームモードのいずれかとなります。
  – ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を送信すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を送信すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – ストリームモードでは、DPSM はデータカウンタ DATACOUNT > 0 の間、カードにデータを送信します。データカウンタが 0 に達した場合、ビジー状態に移行し、送信タイムアウトを開始します。
  DATACOUNT に応じて最後のストリームバイトを送信する前に、DPSM は sendCMD 信号にトリガを発行します。この信号は、CPSM が何らかのペンディングコマンドを送信するために使用します。(例：CMD12 送信停止コマンド)
  – FIFO アンダーランエラーが発生した場合、DPSM は FIFO アンダーランエラーフラグ（TXUNDERR）をセットします。DPSM は送信状態のままとなります。
  – CPSM Abort 信号を受信した場合
  - DATACOUNT = 0 で転送の最後の 2 ビットまでに CPSM_Abort 信号を受信した場合、転送がアボートされます。DPSM は終了ビットの前に最後のデータビットを送信します。FIFO は無効化／一掃されると、DPSM はビジー状態に移行し、ビジーではなくなるのを待ってから DABORT フラグをセットします。
  - DATACOUNT = 0 で転送の最後の 2 ビット処理中または以降に CPSM_Abort 信号を受信した場合、転送は正常に完了し、DABORT フラグはセットされません。
- **ビジー**状態：DPSM は期待されている場合に CRC ステータストークンを待って、ビジー以外の信号を待ちます。
  – CRC ステータストークンが期待され、「エラーなし送信」を示す場合や期待された CRC がない場合：
  – SDMMC_D0 がローでない（カードがビジーでない）場合は Wait_S 状態へ移行します。
  - カードがビジーで、SDMMC_D0 がローの場合、ビジー状態のままとなります。

– CRC ステータストークンが期待され、「エラーあり送信」を示す場合
- すべてのデータが送信されていない場合（DATACOUNT > 0）DPSM は、ビジー以外になるまで待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）がセットされます。FIFO は無効化／一掃され、DPSM はビジー状態のままとなります。
- すべてのデータが送信された場合（DATACOUNT = 0）DPSM は、ビジー以外になるまで待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）がセットされ、DPSM はアイドル状態に移行します。

– DPSM がビジー状態の間に CRC ステータス（Ncrc）タイムアウトが発生した場合には、DPSM はデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。

– DPSM がビジー状態の間にビジータイムアウトが発生した場合には、DPSM はデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。

– ビジー状態で CPSM Abort 信号を受信した場合：
- DATACOUNT > 0 で CRC レスポンスの最後の 2 ビットまでに CPSM_Abort 信号を受信した場合、転送がアボートされます。DPSM は、ビジー以外になり、FIFO が無効化／一掃されるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。
- DATACOUNT = 0 のとき、または期待されている CRC がなく、DATACOUNT = 0 で、DTIMEOUT エラーがないときに、CRC レスポンスの最後の 2 ビット処理中または以降に CPSM_Abort 信号を受信した場合、転送が正常に完了するまで DPSM はビジー状態のままで、DABORT フラグを生成しません。
- DTIMEOUT エラーが発生したときに CPSM_Abort 信号を受信した場合、DPSM は、ビジー以外になり、FIFO が無効化／一掃されるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。

– - 送信状態でのアボートによってビジー状態に移行したとき、DPSM は、ビジー以外になるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。

– DPSM が DTEN で開始した場合、エラー（DATACOUNT > 0 時の DCRCFAIL または DTIMEOUT）後、FIFO がリセットされたときに、DPSM はアイドル状態に移行します。

– SDMMC_D0 でビジーになっているため、DPSM が開始した場合、ビジー以外になるまで待ってから、ビジー終了ステータスフラグ（BUSYD0END）がセットされ、DPSM はアイドル状態に移行します。

データタイマ（DATATIME）は、DPSM が Wait_R またはビジー状態のときに、データブロック終了ビットまたはデータ読み出しコマンド終了ビット、又は R1b レスポンス後の 2 サイクル有効になり、以下の場合にデータタイムアウトエラー（DTIMEOUT）を生成します。

- データ送信時にタイムアウトが発生します。
  – CRCステータスが期待されており、8 SDMMC_CK サイクル内で受信するスタートビットがない場合、DTIMEOUT フラグがセットされます。
  – ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUT フラグがセットされます。
- データ受信時にタイムアウトが発生します。
  – DATACOUNT > 0 で受信するデータがまだあり、プログラムされたタイムアウト期間の前に受信するスタートビットがない場合、DTIMEOUT フラグがセットされます。
- R1b レスポンス後に、タイムアウトが発生します。
  – ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUT フラグがセットされます。

DATATIME = 0 の場合

- 受信時、スタートビットはデータブロック終了ビットまたはデータ読み出し終了ビットの後 2 サイクル存在します。
- 送信時、CRC トークン終了ビットまたはストリームデータ終了ビットの後 2 サイクルで、ビジーがタイムアウトします。
- R1b レスポンス後、レスポンス終了ビットの後 2 サイクルで、ビジーがタイムアウトします。

データは、カードからホスト（送信）へ、またはホストからカード（受信）へ転送できます。データは、SDMMC_Dn データラインを介して転送され、FIFO に格納されます。

表 436. データトークンフォーマット

説明	スタートビット	データ(1)	CRC16	終了ビット	DTMODE
ブロックデータ	0	（DBLOCKSIZE、DATALENGTH）	あり	1	00
SDIO マルチバイト	0	（DATALENGTH）	あり	1	01
MMC ストリーム	0	（DATALENGTH）	なし	1	10

1. 転送するデータの総数は DATALENGTH で指定します。ブロックデータの場合、各ブロックのデータの数は DBLOCKSIZE で指定します。

データ トークンフォーマットは、適宜レジスタビット DTMODE で選択されます。

データパスは、*表 437* に示すようにステータスフラグおよび関連付けられたクリアビットを実装します。

表 437. データパスステータスフラグとクリアビット

フラグ		説明
DATAEND	TX	CRC が OK で、ビジーが終了したときにすべてのデータ転送終了時にセット。（DATACOUNT = 0）。（DPSM が WAIT_S からアイドルに移行）
	RX	CRC が OK で、すべてのデータを読み出したときにすべてのデータ転送終了時にセット。（DATACOUNT = 0 かつ FIFO が空）。（DPSM が WAIT_R からアイドルに移行）
	BOOT	
DCRCFAIL	TX	FAIL で、ビジーが終了したときに CRC 終了時にセット。（送信するデータがまだあり、アボートを待っている場合、DPSM はビジーのまま）（すべてのデータが送信された場合、または DPSM が DTEN で開始したとき、DPSM はビジーからアイドル状態に移行）
	RX	FAIL で、FIFO が空のときに CRC 終了時にセット。（受信するデータがまだあり、アボートを待っている場合、DPSM は受信のまま）（すべてのデータが受信された場合、または DPSM が DTEN で開始したとき、DPSM は受信からアイドル状態に移行）
	BOOT	
ACKFAIL	BOOT	FAIL のときに BOOT ACK の終了時にセット。（DPSM は Wait_Ack のままでアボートを待機）
DTIMEOUT	CMD R1b	タイムアウト前に受信したビジー終了がないコマンドレスポンスの後にセット。（DPSM はビジーのままでアボートを待機）
	TX	Ncrc 内で受信した CRC トークンスタートビットがなかった場合、またはタイムアウト前に受信したビジー終了がなかった場合にセット。（DPSM はビジーのままでアボートを待機）（DPSM が DTEN で開始したときはアイドル状態に移行） 注：ビジータイムアウト前に CRC コマンド失敗時に DCRCFAIL フラグもセットされる場合があります。
	RX	タイムアウト前に受信したスタートビットがないときにセット。（DPSM は WAIT_R のままでアボートを待機）（DPSM が DTEN で開始したときはアイドル状態に移行）
	BOOT	
ACKTIMEOUT	BOOT	タイムアウト前に受信したスタートビットがないときにセット。（DPSM は Wait_Ack のままでアボートを待機）

表 437. データパスステータスフラグとクリアビット（続き）

フラグ		説明
DBCKEND	TX	DTHOLD = 1 の場合：CRC が OK で、ビジーが終了したとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が WAIT_S からアイドルに移行）
	RX	RWSTART = 1 の場合：CRC が OK のとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が受信から ReadWait に移行）
	BOOT	
DHOLD	TX	DTHOLD = 1 の場合：CRC が OK で、ビジーが終了したとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が WAIT_S からアイドルに移行）
	RX	DTHOLD = 1 の場合：CRC が OK で、すべてのデータが読み出された（FIFO が空になった）とき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が WAIT_R からアイドルに移行）
DABORT	CMD R1b	CPSM によってアボートイベントが送信され、ビジーが終了したとき。（DPSM がビジーからアイドルに移行）
	TX	転送の最後の 2 ビットまでに CPSM によってアボートイベントが送信されたとき。（DPSM が任意の状態からアイドルに移行）
	RX	
	BOOT	
BUSYD0END	CMD R1b	タイムアウト前にビジー終了になったときのコマンドレスポンスの後にセット。（DPSM がビジーからアイドルに移行）
DPSMACT		データ転送中。（DPSM がアイドル状態ではない）

データパスエラー処理は、*表 438* に示します。

表 438. データパスエラー処理

エラー	DPSM の状態	原因	カード動作	ホスト動作	DPSM 動作
タイムアウト	Wait_Ack	時間内に確認応答なし	不明	カードサイクルパワー	Wait_Ack のまま（RCC.SDMMCxRST レジスタビットで SDMMC をリセット）
	Wait_R	時間内にスタートビットなし	不明	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
			不明	ブート手順の停止	
	ビジー	長すぎるビジー（データ転送による）	不明	データ受信の停止 送信停止コマンドの送信	
		長すぎるビジー（R1b による）	不明	リセットコマンドの送信	
CRC	受信	送信エラー	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
CRC ステータス	ビジー	否定ステータス	追加データの無視	データ送信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
		送信エラー	追加データを待つ		
確認応答ステータス	Wait_Ack	送信エラー	ブートデータの送信	ブート手順の停止	CPSM_Abort でアイドルに移行

表 438. データパスエラー処理

エラー	DPSM の状態	原因	カード動作	ホスト動作	DPSM 動作
オーバーラン	受信	FIFO フル	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
アンダーラン	送信	FIFO エンプティ	追加データの受信	データ送信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行

データ FIFO

データ FIFO（ファーストインファーストアウト）サブユニットには、送受信データバッファが含まれます。シングル FIFO は、DTDIR ビットで選択されたとおり送信または受信に使用されます。FIFO には、32 ビット幅、16 ワードの深さのデータバッファと、制御ロジックがあります。データ FIFO は AHB クロックドメイン（sdmmc_hclk）で動作するため、SDMMC クロックドメイン（SDMMC_CK/sdmmc_rx_ck）のサブユニットからの信号はすべて再同期されます。

FIFO は次の状態のいずれかになり得ます。

- 送信 FIFO は、カードにデータを送信すると、送信ロジックとデータバッファを参照します。（DTDIR = 0）
- 受信 FIFO は、カードからデータを受信すると、受信ロジックとデータバッファを参照します。（DTDIR = 1）

正常に完了した FIFO からの SDMMC データ転送の終了は、データパスサブユニットによって駆動される DATAEND フラグで示されます。FIFO からの誤った（アボートされた）SDMMC データ転送は、データパスサブユニットによって駆動されるエラーフラグ（DCRCFAIL、DTIMEOUT、DABORT）のいずれかか、FIFO 制御によって駆動される FIFO エラーフラグ（TXUNDERR、RXOVERR）のいずれかで示されます。

データ FIFO には、次の方法でアクセスすることができます。*表 439* を参照してください。

表 439. データ FIFO アクセス

データ FIFO アクセス	IDMAEN
AHB スレーブインタフェース経由で FW から	0
AHB マスタインタフェース経由で IDMA から	1

送信 FIFO：

データは、DPSM が有効化されたとき（DPSMACT = 1）に、送信 FIFO に書き込むことができます。

IDMAEN = 1 の場合、FIFO は IDMA によって完全に処理されます。

IDMAEN = 0 の場合、FIFO は AHB スレーブインタフェース経由で FW によって制御されます。送信 FIFO は、連続アドレスを介してアクセス可能です。送信 FIFO には、読み出しポインタが示すデータワードを保持するデータ出力レジスタが含まれています。データパスサブユニットがシフトレジスタへロードすると、読み出しポインタがインクリメントされ、新しいデータが出力されます。送信 FIFO は、次の方法で処理されます。

1. データ長は DATALENGTH に、ブロック長は DBLOCKSIZE に書き込みます。
   – ブロックデータ転送（DTMODE = 0）の場合、DATALENGTH は DBLOCKSIZE の整数倍になります。
2. 送信モードで SDMMC をセットします（DTDIR = 0）。
   – 送信モードで FIFO を設定します。
3. 次のいずれかでデータ転送を有効化します。
   – CMDTRANS ビットをセットして CPSM からコマンドを送信
   – DTEN ビットをセット
4. （DPSMACT = 1）の場合、データを FIFO に書き込みます。
   – DPSM は FIFO がフル（TXFIFOF = 1）または DATALENGTH で示された数字になるまで Wait_S 状態のままとなります。
   – SDMMC は FIFO がエンプティではない限り、データ送信を開始します。
5. FIFO がハーフエンプティ（TXFIFOHE フラグ）の場合、FIFO がフル（TXFIFOF = 1）になるまで、または最後のデータが書き込まれるまでにデータを FIFO に書き込みます。
6. 最後のデータが書き込まれると、データの終了（DATAEND フラグ）を待ちます。
   – SDMMC がすべてのデータを完全に送信して、DPSM が無効になります（DPSMACT = 0）。

データ転送エラー時、または IDMAEN = 0 の場合の転送保留時、FW は FIFO への書き込みを停止し、FIFO を一掃して FIFORST レジスタビットでリセットします。

送信 FIFO ステータスフラグを*表 440* に示します。

表 440. 送信 FIFO ステータスフラグ

フラグ	説明
TXFIFOF	すべての送信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
TXFIFOE	送信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
TXFIFOHE	半分以上の送信 FIFO ワードが空のとき、ハイにセットされます。
TXUNDERR	アンダーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書き込みによってクリアされます。

受信 FIFO：

データは、DPSM が有効になっているとき（DPSMACT = 1）に、受信 FIFO から読み出すことができます。

IDMAEN = 1 の場合、FIFO は IDMA によって完全に処理されます。

IDMAEN = 0 の場合、FIFO は AHB スレーブインタフェース経由で FW によって制御されます。データパスサブユニットがワードデータを受信すると、書き込みデータバス上にデータが出力されます。書き込み操作が完了すると、書き込みポインタがインクリメントされます。読み出し側では、読み出しポインタの現在値が示す FIFO ワードの内容が、読み出しバス上に出力されます。受信 FIFO は、連続アドレスを介してアクセス可能です。受信 FIFO は、次の方法で処理されます。

1. データ長は DATALENGTH に、ブロック長は DBLOCKSIZE に書き込みます。
   – ブロックデータ転送（DTMODE = 0）の場合、DATALENGTH は DBLOCKSIZE の整数倍になります。
2. 受信モードで SDMMC をセットします（DTDIR = 1）。
   – 受信モードで FIFO を設定します。
3. 次のいずれかで DPSM 転送を有効化します。
   – CMDTRANS ビットをセットして CPSM からコマンドを送信
   – DTEN ビットをセット
4. DPSMACT = 1の場合、FIFO はデータを受信する準備ができています。
   – DPSM は受信したデータを FIFO に書き込みます。
5. FIFO がハーフフル（RXFIFOHF フラグ）の場合、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出します。
6. 最後のデータがデータの終了（DATAEND フラグ）を受信した場合、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出します。
   – SDMMC がすべてのデータを完全に受信して、DPSM が無効になります（DPSMACT = 0）。

IDMAEN = 0 の場合のデータ転送保留時、FW は FIFO がエンプティになるまで残りのデータを読み出し、FIFO を FIFORST レジスタビットでリセットします。これにより、DPSM がアイドル状態（DPSMACT = 0）に移行します。

IDMAEN = 0 の場合のデータ転送エラー時、FW は FIFO の読み出しを停止し、FIFO を一掃して FIFORST レジスタビットでリセットします。これにより、DPSM がアイドル状態（DPSMACT = 0）に移行します。

受信 FIFO ステータスフラグを*表 441* に示します。

表 441. 受信 FIFO ステータスフラグ

フラグ	説明
RXFIFOF	すべての受信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
RXFIFOE	受信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
RXFIFOHF	半分以上の受信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
RXOVERR	オーバーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書き込みによってクリアされます。

CLKMUX ユニット

CLKMUX は、受信したデータおよびコマンドレスポンスと使用するクロックのソース sdmmc_rx_ck を選択します。受信データのクロックソースは、次のうちのクロック制御レジスタビット SELCLKRX で選択できます。

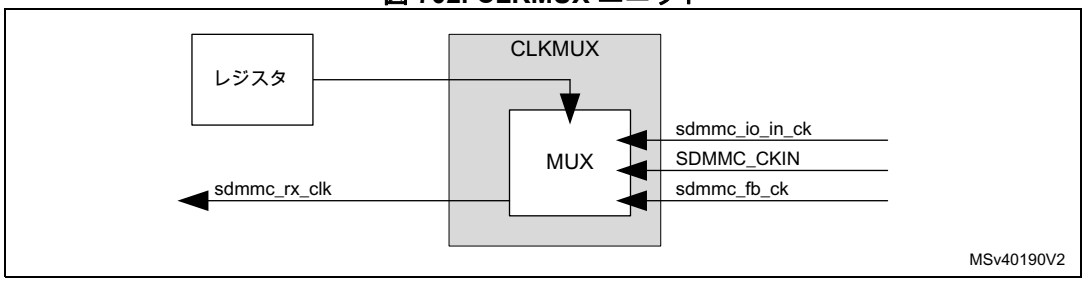
- sdmmc_io_in_ck バスマスタメインフィードバッククロック
- SDMMC_CKIN 外部バスフィードバッククロック
- sdmmc_fb_ck バス調整フィードバッククロック

sdmmc_io_in_ck は、外部ドライバがない場合に DS および ハイスピードで選択されます。

SDMMC_CKIN は、外部ドライバがある場合に SDR12、SDR25、SDR50 および DDR50 で選択されます。

デバイスの DLYB ブロックが SDR104 で使用され、オプションで SDR50 および DDR50 モードで使用されている場合、sdmmc_fb_ck クロック入力を選択しなければなりません。

図 702. CLKMUX ユニット



CPSM および DPSM がアイドル状態の場合、sdmmc_rx_ck ソースは変更します。

55.4.5 SDMMC AHB スレーブインタフェース

AHB スレーブインタフェースは、割り込みリクエストを生成し、SDMMC アダプタレジスタとデータ FIFO にアクセスします。このインタフェースは、データパス、レジスタデコーダ、および割り込みロジックで構成されています。

SDMMC FIFO

FIFO のアクセスは、ワードアクセスで制限されます。

- 送信 FIFO モードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32 ビット）で FIFO に書き込まれます。DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 1 ワード転送で書き込まれます。
- 受信 FIFO モードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32 ビット）で FIFO から読み出されます。DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 0 値バイトでパディングされた 1 ワード転送で書き込まれます。

ハーフワードまたはバイトアクセスで FIFO にアクセスすると、AHB バスフォールトが生成されます。

SDMMC 割り込み

割り込みロジックは、マスクされていないステータスフラグの少なくとも 1 つがアクティブのときにアサートされる割り込みリクエスト信号を生成します。割り込み生成条件が選択できるように、マスクレジスタが提供されています。ステータスフラグは、対応するマスクフラグがセットされている場合に割り込みリクエストを生成します。一部のステータスフラグは、クリアレジスタで間接的なクリアが必要です。

55.4.6 SDMMC AHB マスタインタフェース

AHB マスタインタフェースは、SDMMC IDMA を使用したメモリと FIFO の間でのデータ転送に使用されます。

SDMMC IDMA

DMA（Direct Memory Access）は、SDMMC FIFO とメモリの間でハイスピード転送を行うために使用します。AHB マスタはシステムバスのバンド幅を最適化します。SDMMC 内部 DMA（IDMA）には、送信または受信に使用するチャネルが 1 つ搭載されています。

IDMA は、IDMAEN ビットで有効になり、8 ビートのバースト転送をサポートします。

- 送信バースト転送モードの場合：
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、FIFO がバースト転送の数に対してエンプティであればいつでもメモリからバーストでフェッチされます。DATALENGTH がバーストサイズの整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。
    DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 1 ワード転送でフェッチされます。
- 受信バースト転送モードの場合：
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、FIFO にバースト転送の数がある場合に限りメモリにバーストで格納されます。
    DATALENGTH がバースト転送の整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。
    DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）はハーフワード転送またはバイト転送で格納されます。

さらに、IDMA には、IDMABMODE で選択されるチャネル設定が 2 つあります。

- シングルバッファチャネル
- ダブルバッファチャネル

シングルバッファ設定では、メモリ側のデータにベースアドレス IDMABASE0 から始まる直線的な方法でアクセスします。IDMA がすべてのデータの転送を終了し、DPSM が転送を完了したときに、DATAEND フラグがセットされます。

ダブルバッファ設定では、メモリ側のデータにベースアドレス IDMABASE0 にあるものとベースアドレス IDMABASE1 にあるもう 1 つのものの 2 つのバッファから連続的にアクセスします。これにより、IDMA が一方のメモリバッファにアクセスしている間に、ファームウェアがもう一方のメモリバッファを処理できます。メモリバッファのサイズは、IDMABSIZE によって定義されます。バッファサイズは、バーストサイズの整数倍となります。チャネルが有効である場合、随時バッファのベースアドレスを更新することができ、次のルールが適用されます。

- IDMABACT ビットが“0”の場合、IDMA ハードウェアは、IDMABASE0 を使用してメモリにアクセスします。ファームウェアがこのレジスタに書き込みを試みた場合、書き込みは破棄され、IDMABASE0 データは変わりません。ファームウェアは IDMABASE1 に書き込むことができます。
- IDMABACT ビットが“1”の場合、IDMA ハードウェアは、IDMABASE1 を使用してメモリにアクセスします。ファームウェアがこのレジスタに書き込みを試みた場合、書き込みは破棄され、IDMABASE1 データは変わりません。ファームウェアは IDMABASE0 に書き込むことができます。

IDMA が 1 つのバッファのデータを転送し終えたときに、バッファ転送完了フラグ（IDMABTC）がセットされ、IDMABACT ビットが切り替わり、IDMA はもう一方のバッファからのデータ転送を続行します。IDMA がすべてのデータの転送を終了し、DPSM が転送を完了したときに、DATAEND フラグがセットされます。

IDMABASEn アドレスはワード整列されます。

エラー管理

IDMA 転送エラーは、予約済みアドレス空間に対する読み出しや書き込み時に発生する可能性があります。IDMA 転送エラー時、その後の IDMA 転送は無効になり、IDMATE フラグがセットされます。IDMATE フラグの動作は、IDMA 転送エラーが SDMMC 転送中にいつ発生したかに依存します。

- IDMA 転送エラーは、あらゆる SDMMC 転送エラー（TXUNDERR、RXOVERR、DCRCFAIL、または DTIMEOUT）の前に検出されます。
  - IDMATE フラグは、SDMMC 転送エラーフラグと同時にセットされます。
  - TXUNDERR、RXOVERR、DCRCFAIL、または DTIMEOUT 割り込みが生成されます。
- IDMA 転送エラーは、STOP_TRANSNMISSION コマンド中に検出されます。
  - IDMATE フラグは、DABORT フラグと同時にセットされます。
  - DABORT 割り込みが生成されます。
- IDMA 転送エラーは、SDMMC 転送（HOLD または DATAEND）の終了後に検出されます。
  - IDMATE フラグは、SDMMC 転送の終了時にセットされます。
  - SDMMC 転送終了割り込みが生成され、HOLD または DATAEND フラグがセットされます。

IDMATE は、他の SDMMC 転送割り込み（TXUNDERR、RXOVERR、DCRCFAIL、DTIMEOUT、DABORT、HOLD、または DATAEND）で生成されます。

55.4.7 MDMA リクエストの生成

内部トリガライン （SDMMC から）は、MDMA コントローラに直接リクエストをパスして、CPU を使用せずに異なる内部 RAM アドレスと連続的な通信を可能にします。

カードとのデータ転送が正常に完了すると、ステータスレジスタの DATAEND フラグがセットされます。イベントは、sdmmc_dataend_trg 出力を通じて MDMA リクエスト入力に信号が送られます。これにより、DATAEND および CMDREND フラグのクリアがトリガでき、最終的に新しい転送が CPU を介さずに MDMA を通じた SDMMC 制御および設定レジスタに直接アクセスして開始します。

動作（SDMMC リクエストに応じて MDMA でプログラムするための動作）は、次の表に示します。

表 442. MDMA との SDMMC 接続

トリガ信号	イベント信号	イベント発生条件	MDMA 転送設定	MDMA 動作
sdmmc_ dataend_trg	正常なデータ転送の終了	DATAEND = 1	シングル	DATAENDC をセット

55.4.8 AHB と SDMMC_CK のクロック関係

AHB は、少なくとも SDMMC バス周波数の 3 倍以上にします。SDR50 4 ビットモード（50 MB/秒）の場合、sdmmc_hclk の最低周波数は 37.5 MHz（150 MB/秒）とします。

表 443. AHB と SDMMC_CK のクロック周波数関係

SDMMC バスモード	SDMMC バス幅	最大 SDMMC_CK [MHz]	最小 AHB クロック [MHz]
MMC DS	8	26	19.5
MMC ハイスピード	8	52	39
MMC DDR52	8	52	78
MMC HS200	8	200	150

表 443. AHB と SDMMC_CK のクロック周波数関係

SDMMC バスモード	SDMMC バス幅	最大 SDMMC_CK [MHz]	最小 AHB クロック [MHz]
SD DS / SDR12	4	25	9.4
SD ハイスピード / SDR25	4	50	18.8
SD DDR50	4	50	37.5
SD SDR50	4	100	37.5
SD SDR104	4	208	78

55.5 カード機能詳細

55.5.1 SD I/O モード

以下の機能は、SDMMC 固有の操作です。

- SDIO 割り込み
- SDIO サスペンド／レジューム操作（書き込みおよび読み出しサスペンド）
- クロックの停止による SDIO 読み出しウェイト操作
- SDMMC_D2 信号による SDIO 読み出しウェイト操作

表 444. SDIO 特殊動作制御

動作モード	SDIOEN	RWMOD	RWSTOP	RWSTART	DTDIR
割り込み検出	1	X	X	X	X
サスペンド／レジューム操作	X	X	X	X	X
ReadWait SDMMC_CK クロック停止（START）	X	1	0	1	1
ReadWait SDMMC_CK クロック停止（STOP）	X	1	1	1	1
ReadWait SDMMC_D2 信号（START）	X	0	0	1	1
ReadWait SDMMC_D2 信号（STOP）	X	0	1	1	1

SD I/O 割り込み

SD インタフェースのピン 8（4 ビットモードで SDMMC_D1 と共有）には、SD I/O カードからホストへの割り込みを可能にする割り込み機能が用意されています。割り込みの使用は、各カードまたはカード内の各機能でオプションとなっています。SD I/O 割り込みはレベル対応であるため、割り込みラインはホストによって認識され、割り込みが処理されるか、割り込み期間の終了によってネゲートされるまで、アクティブ（ロー）に保たれる必要があります。ホストが割り込み処理を終えると、SD I/O カードの内部レジスタの該当するビットへの I/O 書き込みによって割り込みステータスビットがクリアされます。すべての SD I/O カードの割り込み出力はアクティブローであり、アプリケーションはすべてのデータライン（SDMMC_D[3:0]）上に、外部プルアップ抵抗を提供する必要があります。

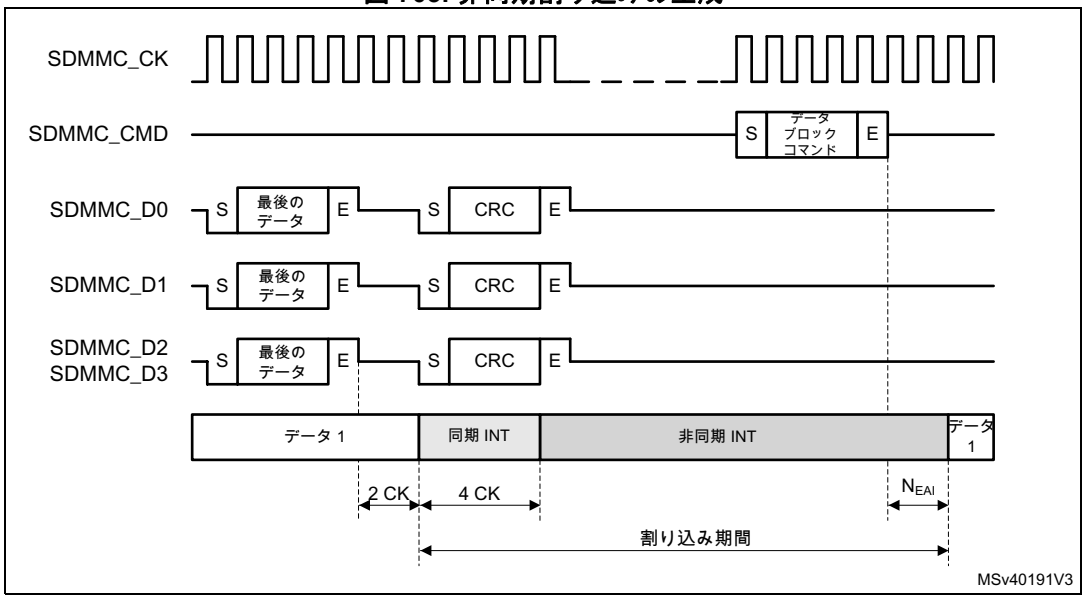
SD 1 ビットモードでは、ピン 8 が割り込み機能（IRQ）専用となり、割り込みに関するタイミングの制約がありません。

SD 4 ビットモードでは、ホストは、割り込み期間に限って、割り込み検出器にピン 8（SDMMC_D1/IRQ）のレベルをサンプリングします。その他の状況では、ホスト割り込みはこの値を無視します。割り込み期間は、割り込みがカードで有効化され、SDIOEN ビットがセットされた際に開始します。*表 444* のレジスタ設定を参照してください。

4 ビットモードでは、カードはカード CCCR レジスタ SAI および EAI ビットで示されたとおりに同期割り込みまたは非同期割り込みを生成できます。

- 同期割り込みは、SDMMC_CK をアクティブにする必要があります。
- 非同期割り込みは､SDMMC_CK が最後のデータブロック後のカード割り込み期間の開始後 4 サイクル停止したときに生成できます。

図 703. 非同期割り込みの生成



割り込み期間のタイミングは、バス速度モードに依存します。

レジスタビット BUSSPEED で選択する DS、ハイスピード、SDR12、SDR25 のモードでは、割り込み期間が SD クロックと同期します。

- 割り込み期間は、データブロックを転送するコマンドの終了ビット以降の次のクロック（CMDTRANS ビットで送信されたコマンドがセット）、または DTEN ビットがセットされたときに終了します。
- 割り込み期間は、データブロックの完了後に 2 SDMMC_CK で再開します。
- データブロックの差で、割り込み期間は 2 SDMMC_CK サイクルに制限されます。

注： ***DTEN は SD および eMMC カードでデータ転送の開始に使用しません。***

図 704. 同期割り込み期間のデータ読み出し

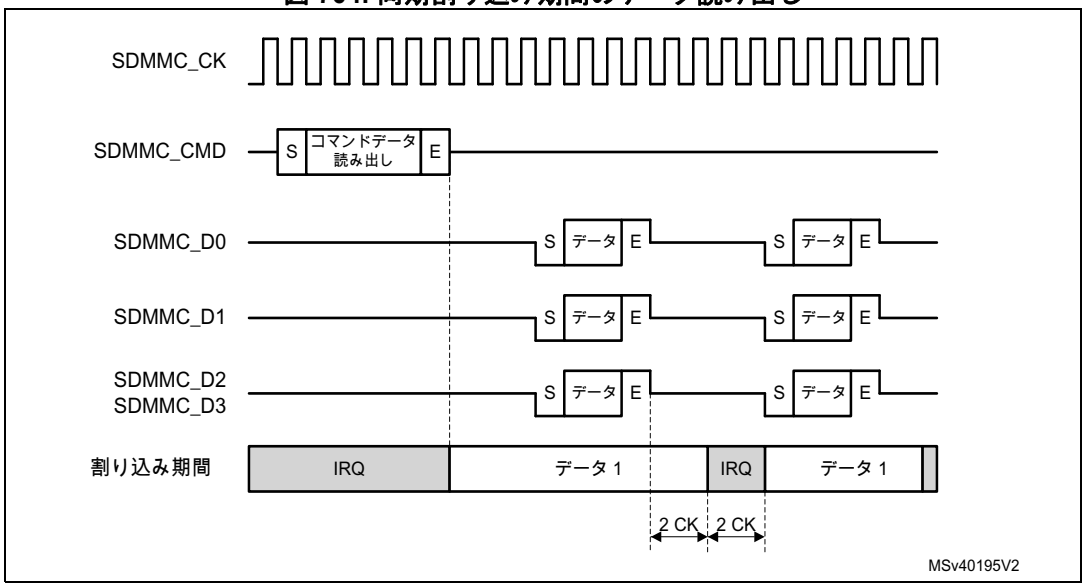
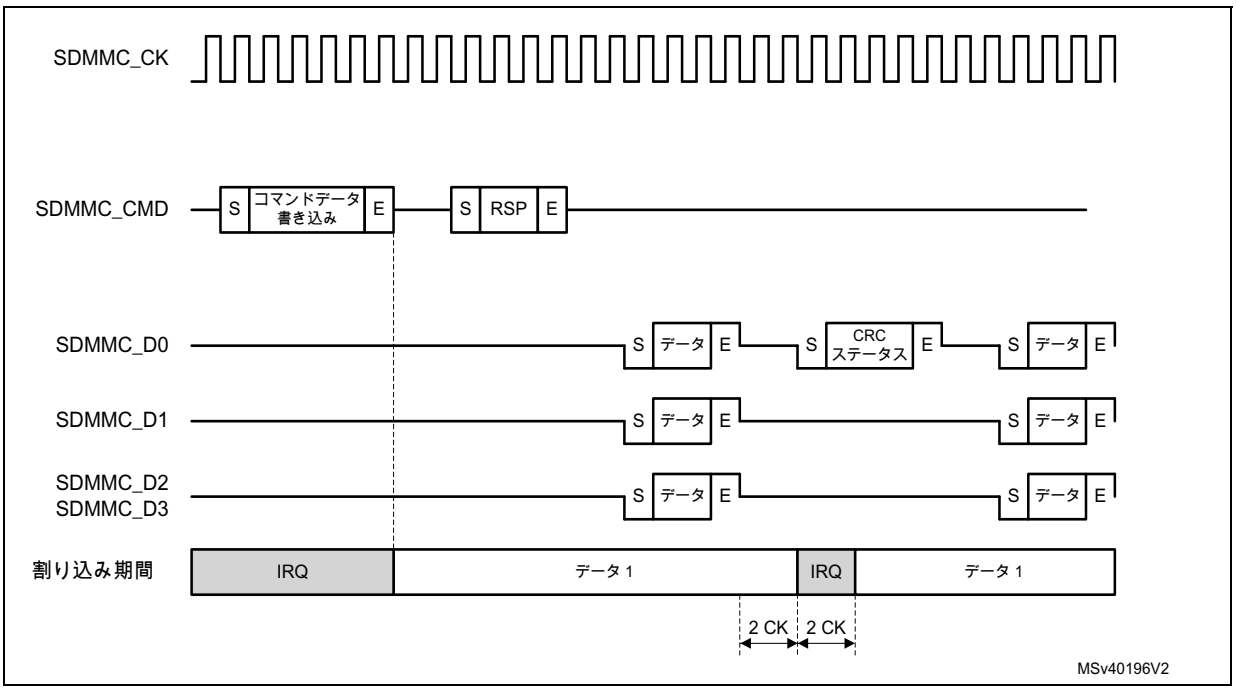


図 705. 同期割り込み期間のデータ書き込み



レジスタビット BUSSPEED で選択する SDR50、SDR104、および DDR50 では、カードからホストへの伝搬遅延によって、割り込み期間は非同期になります。

- カードの割り込み期間は、データブロックを転送するコマンドの終了ビット以降の 0～2 SDMMC_CK サイクル後（CMDTRANS ビットで送信されたコマンドがセット）、または DTEN ビットがセットされたときに終了します。ホストでは、割り込み期間がデータブロックを転送するコマンドの終了ビット以降に終了します。コマンド終了ビット後 1～2 サイクルで発行されたカードの割り込みは、この割り込み期間中ホストによって検出されません。

- カードの割り込み期間は、最後のデータブロックの完了後に 2～4 SDMMC_CK で再開します。ホストは、最後のデータブロック後に常に 2 サイクルで割り込み期間を再開します。
- データブロックの間で割り込み期間はありません。

注： ***DTEN は SD および eMMC カードでデータ転送の開始に使用しません。***

図 706. 非同期割り込み期間のデータ読み出し

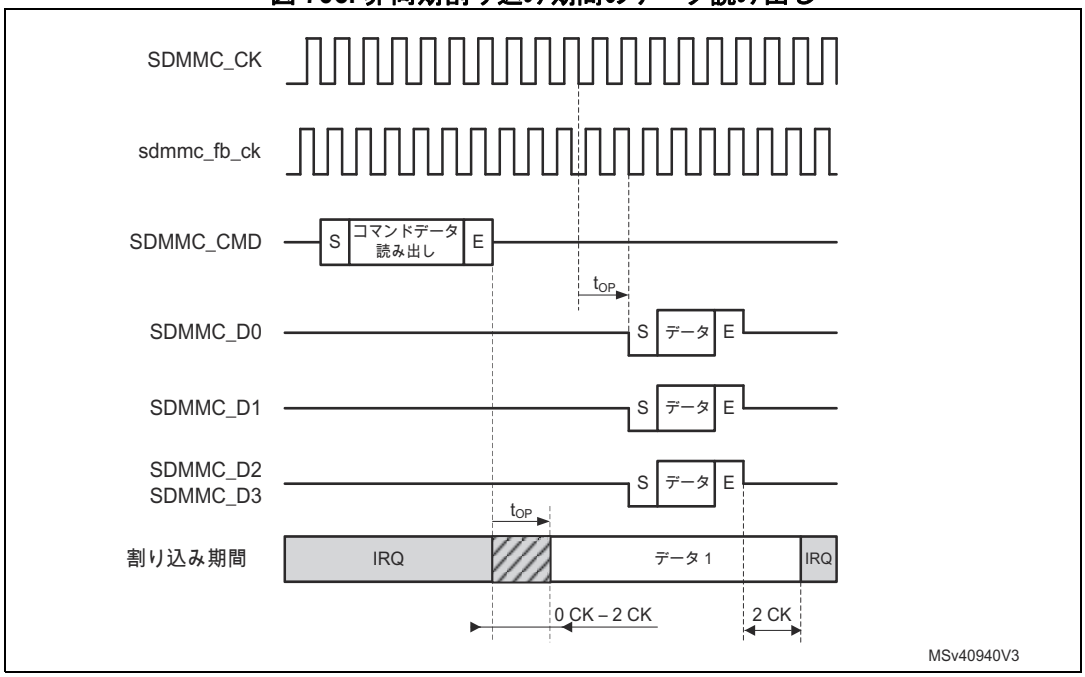
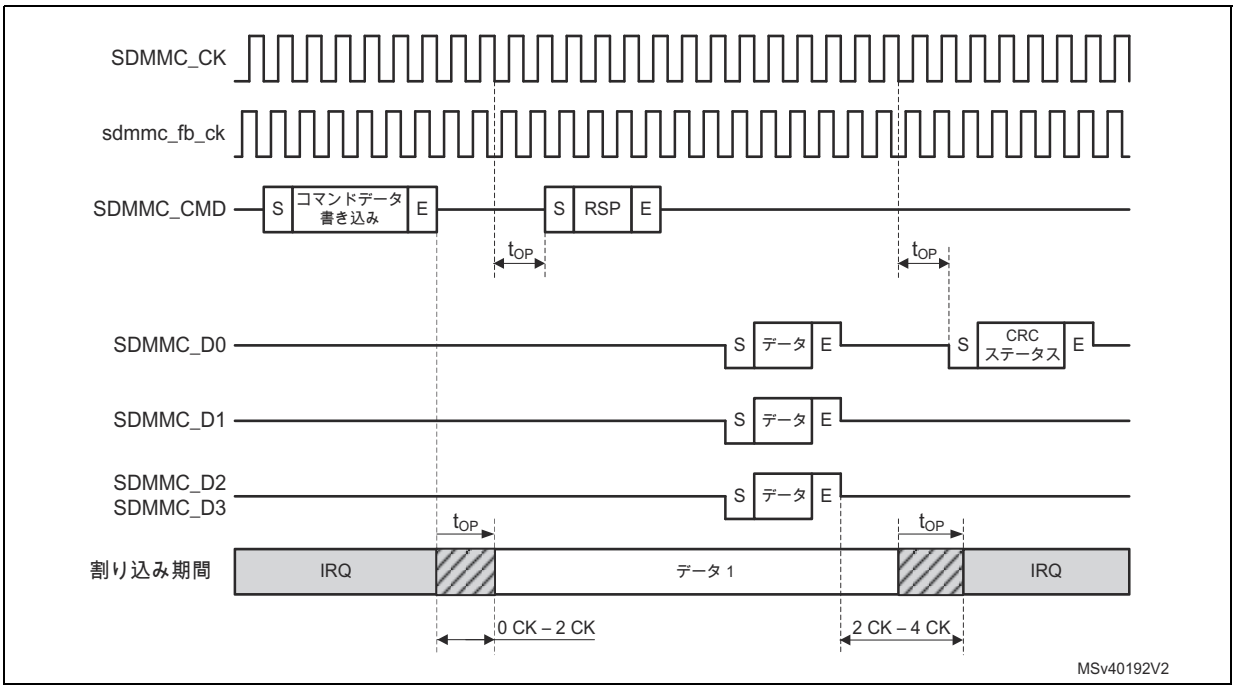


図 707. 非同期割り込み期間のデータ書き込み



オープンエンド複数ブロックデータを転送し、DTMODE「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」を使用する場合、SDMMC は最後のデータブロックから CMD12 STOP_TRANSMISSION コマンドの終了まで割り込み期間をマスクします。

割り込み期間は、メモリと I/O の両方の操作に適用されます。

4 ビットモードでは、*表 445* に従って他の信号と区別できます。

表 445. 4 ビットモードのスタート、割り込み、CRC ステータスの信号検出

SDMMC データライン	スタート	割り込み	CRC ステータス
SDMMC_D0	0	1 または CRC ステータス	0
SDMMC_D1	0	0	X
SDMMC_D2	0	1 または ReadWait	X
SDMMC_D3	0	1	X

SD I/O のサスペンドとレジューム

この機能は、SDIO バージョン 4.00 以降ではサポートされていません。

マルチファンクション SD I/O または I/O とメモリの両方の機能を持つカードでは、MMC/SD バスへのアクセスを共有する複数のデバイス（I/O とメモリ）があります。複数デバイス間でホストへのアクセスを共有するため、SD I/O およびコンボカードはオプションでサスペンド／レジュームの概念を実装します。カードがサスペンド／レジュームをサポートするときには、ホストは 1 つの機能またはメモリへのデータ転送を一時的に中止（サスペンド）して、別の機能またはメモリへの、より優先順位の高い転送用にバスを開放できます。この優先順位の高い転送が完了すると、本来の転送がアボートした所から再開（レジューム）されます。

バス上でサスペンド／レジューム操作を行うには、ホストは以下の手順を実行します。

1. SDMMC_D[3:0] ラインを現在使用している機能を特定します。
2. 優先順位が低いトランザクションや時間のかかるトランザクションをサスペンドするように要求します。
3. トランザクションのサスペンドの完了を待ちます。
4. 優先順位の高いトランザクションを開始します。
5. 優先順位の高いトランザクションの完了を待ちます。
6. サスペンドされたトランザクションをリストアします。

サスペンドコマンドを受信したカードは、現在のバスステータスで応答します。バスがカードによってサスペンドされた場合のみ、バスステータスがサスペンドが完了したことを示します。

次のような個別のサスペンドケースの条件があります。

- データ転送開始前に許可したサスペンドリクエスト。
- （データが同時に転送されたため）許可されなかったサスペンドリクエスト。ホストは許可されるまでリクエストをチェックし続けます。（データ転送がサスペンドしました）。
- 書き込みビジー中のサスペンドリクエスト。
- 複数書き込み中のサスペンドリクエスト。
- ReadWait 中のサスペンドリクエスト。

ホストでバスが開放されたかどうか確認するために、サスペンドが完了したというサスペンドリクエストのステータスを確認します。

サスペンドリクエストレスポンスのバスステータスがサスペンド完了を示すと、カードがバスを開放します。このとき、サスペンドされた操作の状態をその後で他の操作を開始できるように保存します。

サスペンドコマンドが CMDSUSPEND ビットをセットすると送信されます。これにより、バスがサスペンドされた（レスポンスビット BS = 0）ときに、サスペンドコマンドレスポンス後に割り込み期間を開始できます。

ハードウェアは、サスペンド操作を再開させるときに転送する残りのデータの数を保存しません。転送されたデータを判定して、正しい残りのデータバイト数で再開することはファームウェア次第となります。

カードからデータを受信している間、SDMMC は読み出しデータブロック終了（Wait_R の DPSM）以後の読み出し操作をサスペンドできます。カードからサスペンド確認応答レスポンスを受信した後、ファームウェアによって次の手順が実行されます。

1. 通常受信プロセスは、DTHOLD ビットをセットすることで停止します。
   a) FIFO の残りのデータバイト数が、受信 FIFO がエンプティになる（RXFIFOE フラグがセットされる）まで読み出され、IDMAEN = 0 の場合、FIFO は FIFORST でリセットされます。
2. すべてのデータが FIFO から読み出されたことと、サスペンドが完了したことの確認は、DHOLD フラグによって示されます。
   a) 操作再開時に読み出されるべき残りのデータバイト数（データブロックの倍数）は、DATACOUNT によって示される残りのバイト数から判定されます。

注： ***サスペンド手順中に DTIMEOUT フラグが発生すると、これは無視されます。***

カードからデータの受信を再開するには、ファームウェアによって次の手順が実行されます。

1. 残りのデータバイト数（データブロックの倍数）は DATALENGTH でプログラムされます。
2. DPSM が DTDIR でデータを受信するために設定されます。
3. レジュームコマンドが CMDTRANS ビットのセットと CMDSUSPEND ビットのセットで CPSM から送信され、データ転送が再開されたとき（レスポンスビット DF = 1）に割り込み期間が終了し、DPSM が有効になり、カードがデータ送信を再開します。

カードにデータを送信している間、SDMMC は書き込みデータブロック CRC ステータス終了（ビジーの DPSM）以後の書き込み操作をサスペンドできます。カードにサスペンドコマンドを送信する前に、ファームウェアによって次の手順が実行されます。

1. DHOLD フラグ（IDMAEN = 0 の場合は DBCKEND フラグも）を有効にします。
2. DTHOLD をセットすることで、DPSM は新しいデータブロックを送信開始できなくなります。
3. IDMAEN = 0 の場合 : DBCKEND フラグを受信すると、データ転送が停止します。ファームウェアが FIFO への入力を停止でき、その後 FIFO は FIFORST でリセットされます。FIFO に残っているバイトは、操作再開時に再書き込みする必要があります。
4. DHOLD フラグを受信すると、データ転送が停止します。再開時に書き込まれるべき残りのデータバイト数は、DATACOUNT によって示される残りのバイト数から判定されます。
5. カードをサスペンドするために、サスペンドコマンドが CMDSUSPEND ビットのセットで CPSM によって送信されます。これにより、バスがサスペンドされた（レスポンスビット BS = 0）ときに、サスペンドコマンドレスポンス後に割り込み期間を開始できます。

カードへのデータの送信を再開するには、ファームウェアによって次の手順が実行されます。

1. 残りのデータバイト数は DATALENGTH でプログラムされます。
2. DPSM は DTDIR セットで送信のために設定され、CMDTRANS ビットのセットと CMDSUSPEND ビットのセットで CPSM にレジュームコマンドを送信させることで有効になります。これにより、割り込み期間が終了し、データ転送を開始します。DPSM は、SDMMC_D0 がビジーの信号を発していない場合 Wait_S 状態に移行し、ビジーの信号を発している場合はビジー状態に移行します。

3. IDMAEN = 1 の場合:残りのバイトを転送するために DMA を再プログラムする必要があります。
4. IDMAEN = 0 の場合 : ファームウェアが残りのデータを FIFO に入力し始めます。

SD I/O ReadWait

ブロックのギャップの間にデータ転送を一時停止する 2 つの方法があります。

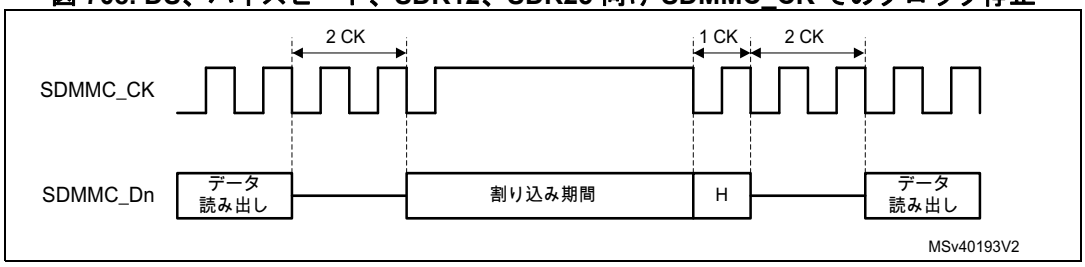
1. SDMMC_CK の停止
2. SDMMC_D2 での ReadWait 信号の使用

SDMMC は、*表 444* に従ったレジスタ設定で ReadWait を実行できます。

SDMMC 動作モード（DS、ハイスピード、SDR12、SDR25）または（SDR50、SDR104、DDR）に応じて、各方法には異なる特性があります。

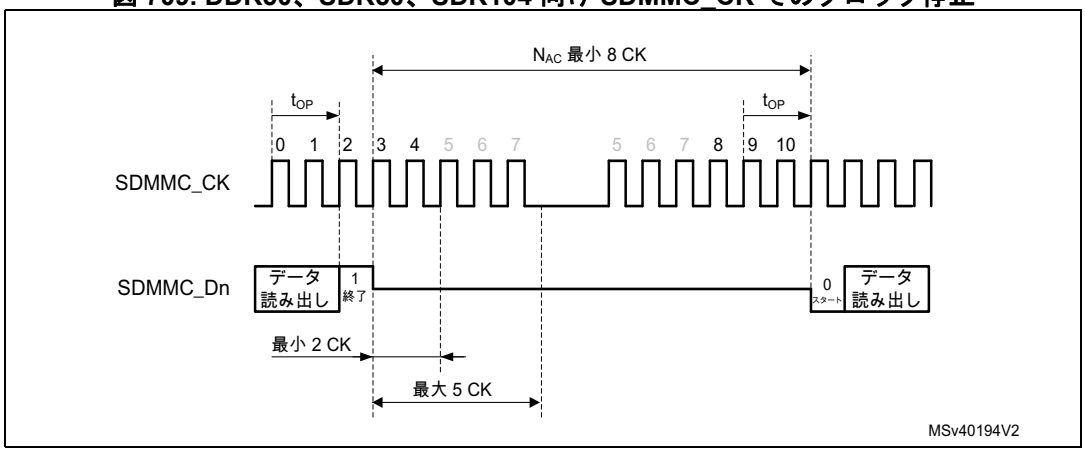
DS、ハイスピード、SDR12、SDR25 の場合 SDMMC_CK を停止することで読み出し操作を一時停止するタイミングは、終了ビット後の 2 SDMMC_CK サイクルで、SDMMC_CK が停止されます。準備が整うと、ホストはクロックを再開することで再開します。*図 708* を参照してください。

図 708. DS、ハイスピード、SDR12、SDR25 向け SDMMC_CK でのクロック停止



SDR50、SDR104、、DDR50 の場合 SDMMC_CK を停止することで読み出し操作を一時停止するタイミングは、終了ビット後の最小 2 SDMMC_CK サイクルおよび最大 5 SDMMC_CK サイクルで、SDMMC_CK が停止されます。準備が整うと、ホストはクロックを再開することで再開します。*図 709* を参照してください。（DDR50 モードでは、クロックラインがローの場合、SDMMC_CK は立ち下がりエッジの後でのみ停止されます。）

図 709. DDR50、SDR50、SDR104 向け SDMMC_CK でのクロック停止



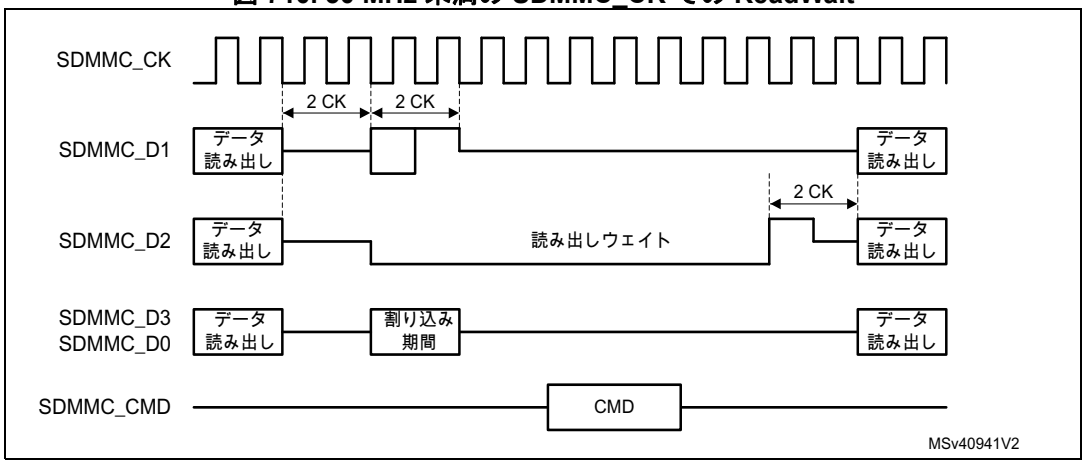
ReadWait SDMMC_CK クロック停止時、RWSTART がセットされている場合、DSPM は現在受信したデータブロック CRC の終了ビット後にクロックを停止します。RWSTOP ビットに 1 が書き込まれた後、クロックは再開し、その後、DPSM はカードからのスタートビットを待ちます。

SDMMC_CK が停止されると、コマンドをカードに発行できません。ReadWait インターバル中も、SDMMC は SDMMC_D1 上で SDIO 割り込みを検出することができます。

オプションの SDMMC_D2 での ReadWait 信号（RW）操作は、SD の 1 ビットおよび 4 ビットモードについてのみ定義されます。ReadWait 操作によって、ホストは複数のレジスタを読み出してカードに信号を送り（IO_RW_EXTENDED、CMD53）、データ転送を一時的にストールし、その間にホストは SD I/O デバイス内の機能にコマンドを送信できます。カードが ReadWait プロトコルをサポートしているかどうかを判断するには、ホストが内部カードレジスタの機能ビットをテストする必要があります。

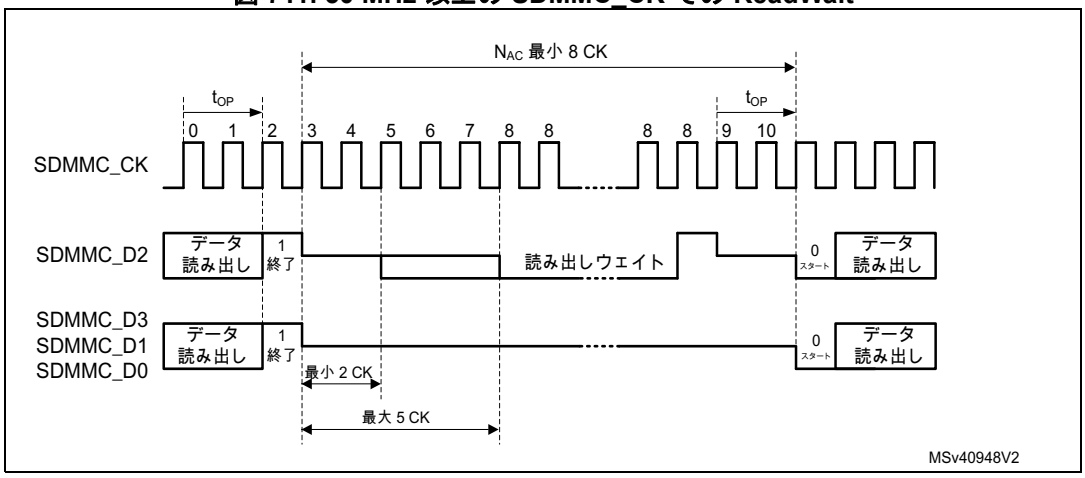
50 MHz 未満の SDMMC_CK での ReadWait のタイミング（DS、ハイスピード、SDR12、SDR25）は、SDMMC_D1 上のカードで生成された割り込み期間に基づきます。割り込み期間中に SDMMC_D2 をローでアサートすることで、ホストはカードに ReadWait に移行するようリクエストします。ReadWait を終了するには、ホストがハイインピーダンスにする前の 1 SDMMC_CK サイクル中に SDMMC_D2 をハイにします。*図 710* を参照してください。

図 710. 50 MHz 未満の SDMMC_CK での ReadWait



50 MHz 以上の SDMMC_CK の SDR50、SDR104 および DDR50 の場合、カードは SDMMC_D2 での ReadWait リクエストを非同期イベントとして処理します。最小 2 SDMMC_CK サイクルおよび最大5 SDMMC_CK サイクルの後に SDMMC_D2 をローでアサートすることで、ホストはカードに ReadWait に移行するようリクエストします。ReadWait を終了するには、ホストがハイインピーダンスにする前の 1 SDMMC_CK サイクル中に SDMMC_D2 をハイにします。ホストは SDMMC_CK クロックで SDMMC_D2 を立ち上げます（*図 711* を参照してください）。

図 711. 50 MHz 以上の SDMMC_CK での ReadWait



ReadWait SDMMC_D2 信号では、RWSTART がセットされている場合、DSPM は現在受信したデータブロック CRC の終了ビット後に SDMMC_D2 を駆動します。SDMMC_D2 での ReadWait 信号は、RWSTOP ビットに 1 が書き込まれると削除されます。DPSM は、さらに 2 SDMMC_CK クロックサイクルの間、ReadWait 状態にとどまり、1 クロックサイクルで SDMMC_D2 を 1 に（SDIO 仕様に従う）駆動して、DPSM はカードのスタートビットを待ちます。

SDMMC_D2 で ReadWait 信号がある間、コマンドをカードに発行できます。ReadWait インターバル中は、SDMMC は SDMMC_D1 上で SDIO 割り込みを検出することができます。

55.5.2 CMD12 送信タイミング

CMD12 は、データ転送の停止／アボートに使用され、送信停止コマンドの終了ビット後 2 クロックサイクルでカードのデータ送信は終了します。

表 446. 使用例

データ操作	送信停止コマンド CMD12 の説明
MMC ストリーム書き込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
MMC オープンエンド複数ブロックの書き込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数での MMC ブロック書き込み	この種類の複数ブロック書き込み終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを受信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
MMC ストリーム読み出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
MMC オープンエンド複数ブロックの読み出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数での MMC ブロック読み出し	この種類の複数ブロック読み出し終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを送信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） トランザクションは、送信停止コマンドを送信することでアボートできます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。

すべてのデータ書き込みと読み出しは、送信停止コマンド CMD12 でいつでもアボートできます。データ転送進行中は、次のデータアボート手順が適用されます。

1. レジスタで CMD12 送信停止コマンドをロードして、CMDSTOP ビットをセットします。
   a) これにより、コマンドが DPSM に送信されると、CPSM が Abort 信号を生成します。
2. 直ちにコマンドを送信するために CPSM を設定します（WAITPEND をクリアします）。
   a) データを送信すると、カードは送信停止コマンド終了ビット後 2 サイクルのデータ転送を停止します。
   データが送信されないと、カードは新しいデータの送信を開始しません。
   b) データを送信すると、ホストは送信停止コマンド終了ビット後に終了ビットの前の最後のデータビットを 1 つ送信します。
   データが送信されないと、ホストは新しいデータの送信を開始しません。

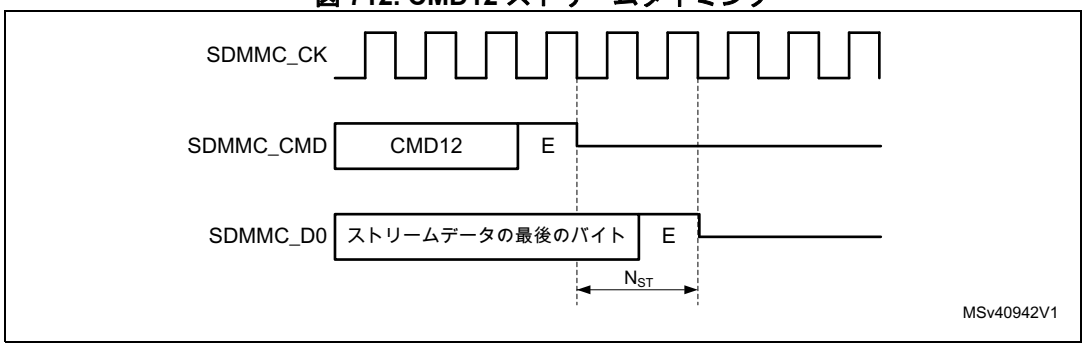
3. IDMAEN = 0 の場合、FIFO は FIFORST でリセットされる必要があります。
   a) カードにデータを書き込む場合。CMDREND フラグで、FW は FIFO へのデータ書き込みを停止します。その後、FIFO は FIFORST でリセットされ、これにより FIFO が一掃されます。
   b) カードからデータを読み出す場合。CMDREND フラグで、FW は FIFO から残りのデータを読み出します。その後、FIFO は FIFORST でリセットされます。
4. IDMAEN = 1 の場合、ハードウェアは FIFO を管理します。
   a) カードにデータを書き込む場合。Abort 信号で、ハードウェアは IDMA を停止し、FIFO が一掃されます。
   b) カードからデータを読み出す場合。Abort 信号で、ハードウェアは IDMA に、FIFO から RAM に残りのデータを転送するように命令します。
5. FIFO がエンプティになるか、リセットされた場合、DABORT フラグが生成されます。

ストリーム動作と CMD12

転送する最後のバイトの後にストリーム転送を停止するために、CMD12 終了ビットタイミングが最後のバイトのデータストリーム終了と合わせて送信されます。次の書き込みストリームデータ手順が適用されます。

1. DPSM で、DTMODE = MCC ストリームデータ転送の場合、ストリームデータを初期化します。
2. CMDTRANS = 1 で CPSM から WRITE_DATA_STREAM コマンドを送信します。
3. CMDSTOP ビットをセットして、コマンドレジスタで CMD12 をプリロードします。
4. CPSM を設定して、最後のデータ（DATALENGTH による）のペンディング待機（WAITPEND = 1）終了後にのみコマンドを送信します。
5. CPSM を有効にして、STOP_TRANSMISSION コマンドを送信すると、ストリームデータ終了ビットとコマンド終了ビットが調整されます。
   a) DATALENGTH > 5 バイトの場合、データ転送終了ビットで調整するために CPSM でコマンド CMD12 が待機します。
   b) DATALENGTH < 5 バイトの場合、コマンド CMD12 が開始され、CMD12 終了ビットでデータ転送終了を調整するために DPSM が Wait_S 状態のままとなります。
6. 書き込みストリームデータは、WAITPEND ビットをクリアすることでいつでもアボートできます。これにより、プリロードされた CMD12 を直ちに送信して、書き込みデータストリームを停止させます。

図 712. CMD12 ストリームタイミング



最後のバイトの後に読み出しストリーム転送を停止するために、CMD12 終了ビットタイミングが最後のデータストリームバイトの後に発生します。次の読み出しストリームデータ手順が適用されます。

1. すべてのデータが DPSM によって受信されるのを待ちます（DATAEND フラグ）。
   a) カードがより多くのデータを送信しても、DPSM は DATALENGTH で示された以上のデータを受信しません。
2. CPSM で CMD12 を送信します。
   a) CMD12 がカード送信データを停止します。

注： ***カードがデータを送信し続けても、DATACOUNT = 0 の場合、SDMMC はカードから追加のデータを受信しません。***

ブロック動作と CMD12

データの終了時にブロック転送を停止するために、CMD12 終了ビットが最後のブロック終了ビットの後に送信されます。

カードにデータを書き込むと、書き込みデータブロック CRC トークン終了ビットの後に、CMD12 終了ビットが送信されます。これには、データブロック送信タイミングと紐づいた CMD12 送信が必要になります。オープンエンド複数ブロック書き込みを停止するには、次の手順が適用されます。

1. データ転送開始前に、DTMODE を「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータが DPSM によって送信され、CRC トークンが受信されるのを待ちます（DATAEND フラグ）。
   a) DPSM は DATALENGTH で示された以上のデータを送信しません。
3. CPSM で CMD12 を送信します。
   a) CMD12 がカードをアイドルモードにセットします。

カードからデータを読み出すと、カード読み出しデータブロックの最後のデータビットと同時で最も早く CMD12 終了ビットが送信されます。これには、データブロック受信タイミングと紐づいた CMD12 送信が必要になります。次の停止オープンエンド複数ブロックの読み出しデータブロックの手順が適用されます。

1. データ転送開始前に、DTMODE を「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータが DPSM によって受信されるのを待ちます（DATAEND フラグ）。
   a) カードがより多くのデータを送信しても、DPSM は DATALENGTH で示された以上のデータを受信しません。
3. CPSM で CMDSTOP ビットをセットして CMD12 を送信します。
   a) CMD12 がカードの追加データ送信を停止し、カードをアイドルモードにセットします。カードによって進行中のあらゆるブロック転送がアボートされます。

注： ***カードがデータを送信し続けても、DATACOUNT = 0 の場合、SDMMC はカードから追加のデータを受信しません。***

55.5.3 SLEEP（CMD5）

MMC カードは、CMD5 によって SLEEP 状態と STANDBY 状態を切り替える場合があります。SLEEP 状態では、カードの消費電力が最小限になり、Vcc 電源がオフに切り替わります。

CMD5（SLEEP）は、STANDBY 状態から SLEEP 状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0 をプルダウンします。カードが SDMMC_DO ラインのプルダウンを停止すると、SLEEP 状態に到達します。

カードを SLEEP 状態にセットするには、次の手順が適用されます。

1. BUSYD0END の割り込みを有効にします。
2. CMD5（SLEEP）を送信します。
3. BUSYD0END 割り込みで、カードが SLEEP 状態になります。
4. Vcc 電源をオフに切り替えられます。

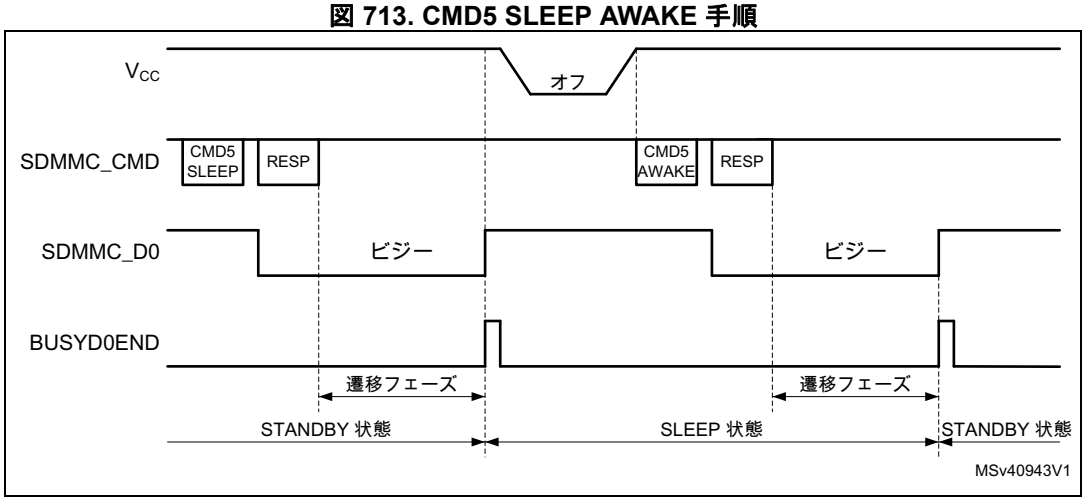
CMD5（AWAKE）は、SLEEP 状態から STANDBY 状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0 をプルダウンします。カードが SDMMC_DO ラインのプルダウンを停止すると、STANDBY 状態に到達します。

カードを STANDBY 状態にセットするには、次の手順が適用されます。

1. Vcc 電源をオンに切り替え、最低動作レベルに到達するまで待ちます。
2. BUSYD0END の割り込みを有効にします。
3. CMD5（AWAKE）を送信します。
4. BUSYD0END 割り込みで、カードが STANDBY 状態になります。

SLEEP 状態に到達した後のみ、Vcc 電源をオフに切り替えられます。Vcc 電源は、CMD5（AWAKE）が送信される前に再配置します。

図 713. CMD5 SLEEP AWAKE 手順



55.5.4 割り込みモード（Wait-IRQ）

ホストとカードは、同時に割り込みモード（Wait-IRQ）の移行と終了を行います。割り込みモードでは、データ転送はありません。許可されるメッセージは、カードまたはホストからの割り込みサービスリクエストレスポンスだけです。割り込みモードで正しく動作するために、静電容量負荷およびプルアップ抵抗に基づいて、オープンドレインモードで実現可能な SDMMC_CMD データレートに合わせて SDMMC_CK 周波数がセットされます。CLKDIV は >1 にセットし、SETCLKRX はソースで sdmmc_io_in_ck または SDMMC_CLK を選択します。

ホストは、カードが STANDBY 状態であることを CMD40（GO_IRQ_STATE）を発行する前に確認する必要があります。割り込みレスポンスを待っている間、SDMMC_CK クロック信号をアクティブに保つ必要があります。

割り込みモード（IRQ 状態）のカード：

- 内部カード割り込みイベントを待っています。イベントが発生すると、カードは割り込みサービスリクエストレスポンスの送信を開始します。レスポンスはオープンドレインモードで送信されます。
- 内部カード割り込みイベントを待っている間、カードは SDMMC_CMD ラインのスタートビットも監視します。スタートビットを検出すると、カードは割り込みモードをアボートして、STANDBY 状態に切り替わります。

割り込みモードのホスト（割り込みを待つ CPSM ウェイトステート）：

- カード割り込みサービスリクエストレスポンス（スタートビット）を待っています。
- カード割り込みサービスリクエストレスポンスを待っている間、ホストは（WAITINT レジスタビットをクリアすることで）割り込みモードをアボートする場合があり、オープンドレインモードで RCA = 0x0000 で割り込みサービスリクエストレスポンス R5 を送信します。

割り込みサービスリクエストレスポンスを送信すると、送信者ビット単位が SDMMC_CMD ビットストリームを監視します。割り込みサービスリクエストレスポンスビットが SDMMC_CMD ラインのビットに対応しない送信者は、送信を停止します。複数の送信者の場合、1 人だけが完全な割り込みサービスリクエストレスポンスを正常に送信します。たとえば、ホストが同時に送信した場合、ホストは送信ビット後に送信を失います。

割り込みモードを処理するには、次の手順が適用されます。

1. オープンドレインモードで実現可能な SDMMC_CMD データレートに合わせて SDMMC_CK 周波数をセットし、CLKDIV は >1 にセットし、SETCLKRX は sdmmc_io_in_ck を選択します。
2. コマンドレジスタで CMD40（GO_IRQ_STATE）をロードします。
3. WAITINT レジスタビットをセットして、割り込みの待機を有効にします。
4. 直ちにコマンドを送信するために CPSM を設定します。
   a) これにより、CMD40 が送信され、CPSM がウェイトステートで中止し、割り込みサービスリクエストレスポンスを待ちます。
5. 割り込みの待機状態（CPSM ウェイトステート）を終了する方法：
   a) 割り込みサービスリクエストレスポンスのスタートビットを検出すると、CPSM はレスポンスを受け取る受信状態に移行します。レスポンスの完全な受信は、CMDREND またはコマンド CRC エラーフラグによって示されます。
   b) 割り込みモードをアボートするために、ホストが WAITINT レジスタビットをクリアすると、ホストは単体で割り込みサービスリクエストレスポンスを送信します。これで CPSM が受信状態に移行します。レスポンスの完全な受信は、CMDREND またはコマンド CRC エラーフラグによって示されます。

注： ***同時送信した割り込みサービスリクエストレスポンスのスタートビットが衝突した場合、ホストは送信ビット後にバスアクセスを失います。***

55.5.5 ブート操作

ブート操作モードでは、ホストは 2 つのブート操作機能のうちいずれかでカードからブートデータを読み出せます。

1. 通常ブート（CMD ラインがローのまま）
2. 代替ブート（CMD0 を引数 0xFFFFFFFA で送信）

ブートデータは、カードレジスタ設定に基づいて、次の設定オプションに応じて読み出すことができます。

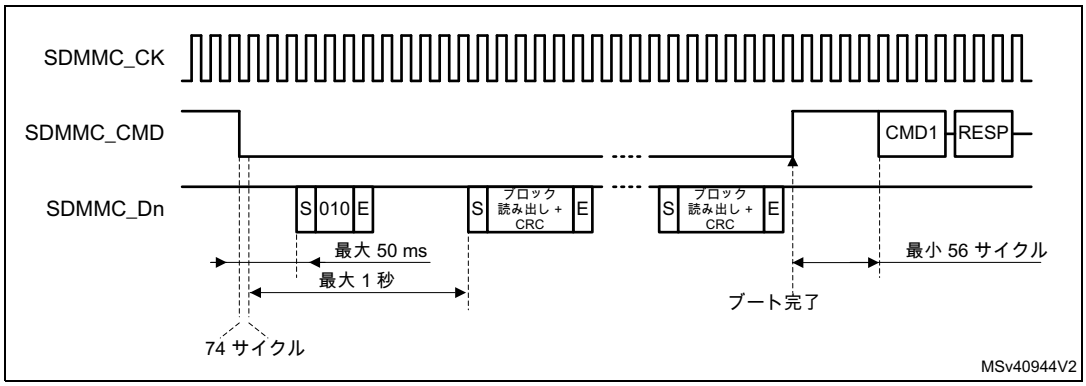
- ブートデータが読み出されるパーティション（EXT_CSD Byte[179]）
- ブートデータサイズ（EXT_CSD Byte[226]）
- ブート中のバス設定（EXT_CSD Byte[177]）
- カードからのブート確認応答の受信（EXT_CSD Byte[179]）

ブート確認応答が有効である場合、CMD ラインがローに移行、または CMD0 を引数 0xFFFFFFFA でブートモードがリクエストされた後の 50 ms 以内に、カードが SDMMC_D0 でパターン 010 を送信します。ブート確認応答タイムアウト（ACKTIMEOUT）と確認応答ステータス（ACKFAIL）が提供されます。

通常ブート動作

カードのパワーアップまたはリセットから最初のコマンドが発行されるまで最低 74 クロックサイクルの間、SDMMC_CMD ラインがローで保たれる場合、カードはブートモードが開始されていると認識します。CMD ラインがローになった後 1 秒以内に、カードは SDMMC_Dn ラインで最初のブートコードデータの送信を開始します。ホストは、すべてのブートデータが読み出されるまで SDMMC_CMD ラインをローのままにしておく必要があります。ホストは、SDMMC_CMD ラインをハイにすることでブートモードを終了できます。

図 714. 通常ブートモード動作



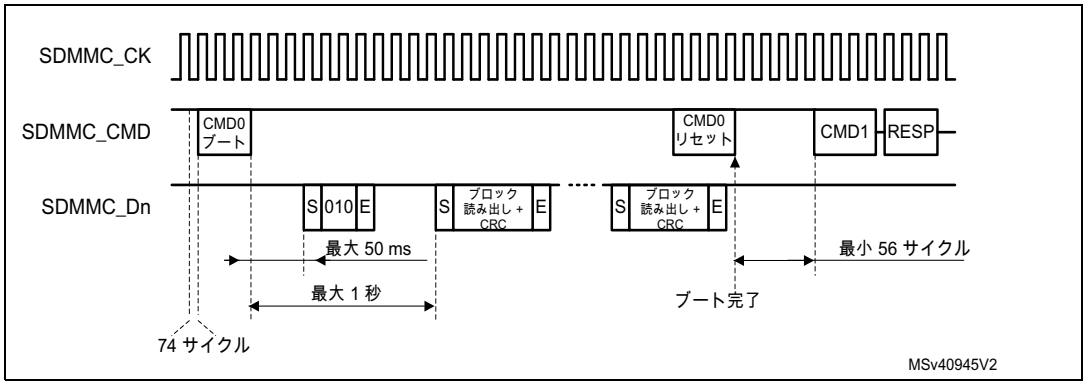
通常ブート手順を実行するには、次の手順が必要です。

1. カードをリセットします。
2. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKFAIL および ACKTIMEOUT 割り込みを有効にします。
3. 受信モード（DTDIR）で DPSM と DATALENGTH で受信するデータバイト数をセットしてデータ受信を有効にします。
4. ブートコマンドの終了確認に DTIMEOUT、DATAEND、および CMDSENT 割り込みを有効にします。
5. BOOTMODE で通常ブート動作モードを選択し、BOOTEN でブートを有効にします。ブート手順は、CPSMEN で CPSM を有効にすることで開始されます。これにより次のことが発生します。
   - SDMMC_CMD がローに駆動されます。（BOOTMODE = 通常ブート）。
   - ACK タイムアウトが開始します。
   - DPSM が有効になります。
6. ブート確認応答の誤受信は、ACKFAIL フラグまたは ACKTIMEOUT フラグが有効時に検出できます。
   - 正しくないブート確認応答が受信された場合、ACKFAIL フラグが発生します。
   - ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUT フラグが発生します。
7. すべてのブートデータが受信されると、DATAEND フラグが発生します。
   - データ CRC が失敗した場合、DCRCFAIL フラグも生成されます。
   - データタイムアウトが発生した場合、DTIMEOUT フラグも生成されます。
8. 最後のデータが受信されると、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出し、データの終了 DATAEND フラグが生成されます。
   - SDMMC がすべてのデータを完全に受信して、DPSM が無効になります。
9. ブート手順は、FW が BOOTEN をクリアすることで終了し、これにより SDMMC_CMD ラインがハイになります。CMDSENT フラグが 56 サイクル後に生成され、新しいコマンドを送信できることを示します。
   a) すべてのデータが受信される前にブート手順が FW によってアボートされた場合、CPSM Abort 信号はデータ受信を停止し、有効時に DABORT フラグを生成する DPSM を無効にします。
10. CMDSENT フラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。

代替ブート動作

カードのパワーアップまたはリセットの後、ホストが CMD0 が発行されるまでの 74 クロックサイクル後に引数 0xFFFFFFFA で CMD0 を送信する場合、カードはブートモードが開始されていると認識します。引数 0xFFFFFFFA で CMD0 が送信された後 1 秒以内に、カードは SDMMC_Dn ラインで最初のブートコードデータの送信を開始します。マスタが CMD0（リセット）を送信してブート動作を終了します。

図 715. 代替ブートモード動作



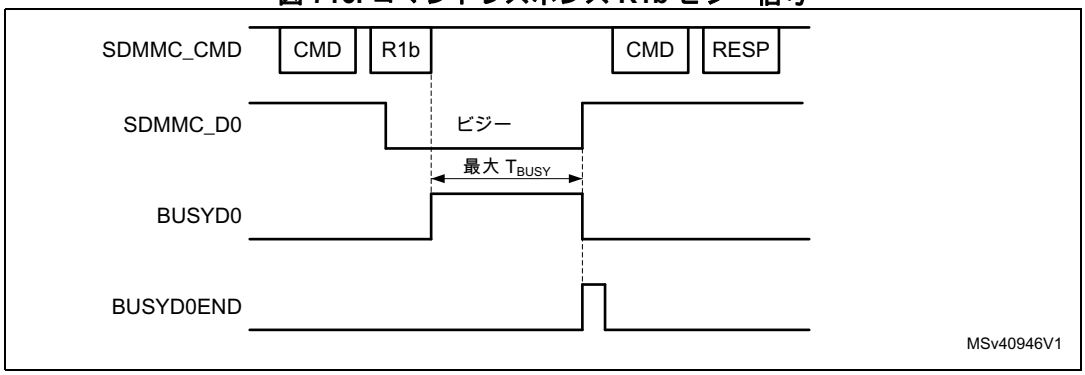
代替ブート手順を実行するには、次の手順が必要です。

1. SDMMC をパワーオフ状態に移行させ、カードをリセットします。
2. SDMMC をパワーオン状態に移行させます。これにより、あらゆるコマンドでクロック供給すべき 74 SCDMMC_CK サイクルを確保します。
3. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKTIMEOUT フラグを有効にします。
4. 受信モード（DTDIR）で DPSM と DATALENGTH で受信するデータ数をセットしてデータ受信を有効にします。DTIMEOUT と DATAEND フラグを有効にします。
5. BOOTMODE で代替ブート動作モードを選択し、コマンドレジスタで 0xFFFFFFFA の引数で CMD0 をロードします。ブートの終了コマンド確認に CMDSENT フラグを有効にして、BOOTEN でブートを有効にします。ブート手順は、CPSMEN で CPSM を有効にすることで開始されます。これにより次のことが発生します。
   – ロードされたコマンドと引数が送信されます。（BOOTMODE = 代替ブート）。
   – ACK タイムアウトが開始します。
   – DPSM が有効になります。
6. コマンドが送信されると、CMDSENT フラグが生成され、その時に BOOTEN ビットがクリアされます。
7. ブート確認応答の受信は、ACKFAIL フラグが有効時に検出できます。
   – ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUT フラグが発生します。
8. すべてのブートデータが受信されると、DATAEND フラグが発生します。
   – データ CRC が失敗した場合、DCRCFAIL フラグも生成されます。
   – データタイムアウトが発生した場合、DTIMEOUT フラグも生成されます。
9. 最後のデータが受信されると、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出し、データの終了 DATAEND フラグが生成されます。
   – SDMMC がすべてのデータを完全に受信して、DPSM が無効になります。
10. BOOTMODE = 代替モードで CMD0（リセット）を送信してブート手順を終了する前に BOOTEN ビットをクリアします。これにより、CMDSENT フラグがコマンドの後 56 サイクルで発生します。
    – すべてのデータが受信される前にブート手順が FW によってアボートされた場合、CPSM Abort 信号はデータ転送を停止し、有効時に DABORT フラグを生成する DPSM を無効にします。
11. CMDSENT フラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。RESET コマンドが正常に送信されると、ブート動作を終了するために BOOTMODE 制御ビットをクリアしなければなりません。

55.5.6 レスポンス R1b の処理

R1b レスポンスのあるコマンドを送信する場合、ビジー信号は BUSYD0 レジスタビットに反映され、BUSYD0END フラグでビジーの開放を反映します。SDMMC_D0 ラインは、R1b レスポンスの終了時にサンプリングされ、BUSYD0 レジスタビットで信号送信されます。BUSYD0 レジスタビットは、BUSYD0END フラグが生成されると同時に、SDMMC_D0 ラインがビジーを開放したときにビジーでない状態にリセットされます。

図 716. コマンドレスポンス R1b ビジー信号



期待される最大ビジー時間は、コマンド送信前に DATATIME レジスタでセットされます。有効時、R1b レスポンスビジーがプログラムされた時間より長くアクティブになっている場合、DTIMEOUT フラグがセットされます。

R1b レスポンスでコマンドを送信したときに SDMMC_D0 ビジー信号を検出するには、次の手順が適用されます。

- CMDREND フラグを有効にします。
- CPSM でコマンドを送信します。
- CMDREND フラグで、BUSYD0 レジスタビットをチェックします。
  - BUSYD0 がビジーではない場合、ビジー開放の信号をファームウェアに送信します。
  - BUSYD0 がビジーである場合、BUSYD0END フラグを待ちます。
- BUSYD0END フラグで、ビジー開放の信号をファームウェアに送信します。
- DTIMEOUT フラグで、ビジーがプログラムされた時間以上にアクティブになります。

55.5.7 リセットとカードサイクルパワー

リセット

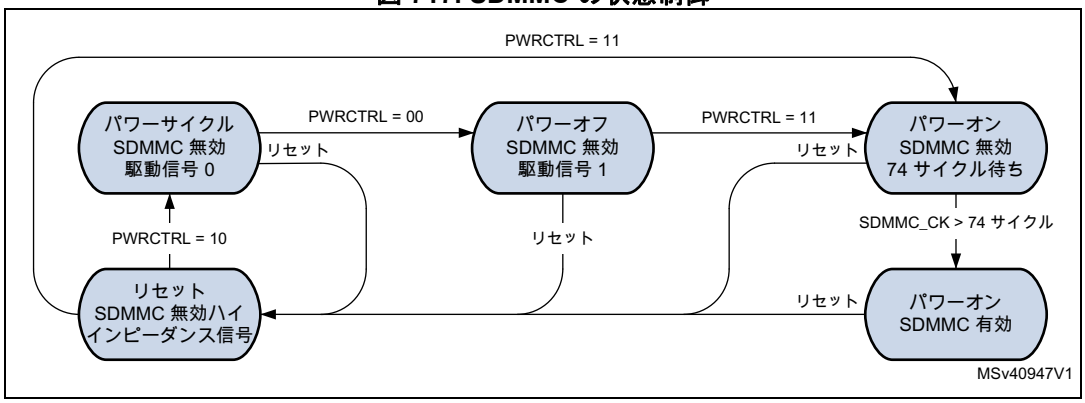
リセット後、SDMMC はリセット状態になります。この状態では、SDMMC が無効になり、コマンドやデータが転送できません。SDMMC_D[7:0] および SDMMC_CMD はハイインピーダンスになり、SDMMC_CK はローに駆動されます。

パワーオン状態に移行する前に、SDMMC を設定します。

パワーオン状態では、SDMMC_CK クロックが実行しています。最初に 74 SDMMC_CK サイクルがクロック供給され、その後に SDMMC が有効になり、コマンドとデータを転送できるようになります。

SDMMC 状態は、*図 717* に従った PWRCTL レジスタビットでファームウェアによって制御されます。

図 717. SDMMC の状態制御

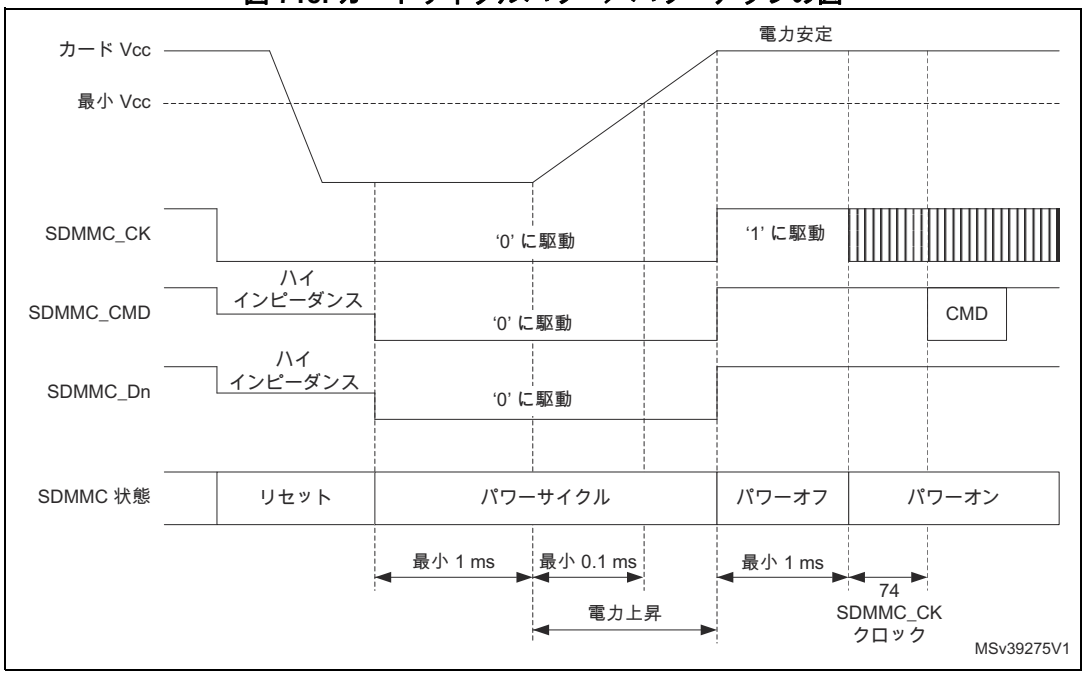


カードサイクルパワー

カードサイクルパワーを実行するには、次の手順が適用されます。

1. RCC.SDMMCxRST レジスタビットで SDMMC をリセットします。これにより、SDMMC がリセット状態にリセットされ、CPSM と DPSM はアイドル状態へ移行します。
2. カードへの Vcc 電源を無効にします。
3. パワーサイクル状態で SDMMC をセットします。これにより、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK がローに駆動され、カードが信号ラインから電源供給されることを防ぎます。
4. 最低 1 ms 後に、カードへの Vcc 電源を有効にします。
5. 電力上昇時間の後、最低 1 ms の間 SDMMC をパワーオフ状態にセットします。SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK がセットされ“1”を駆動します。
6. 1 ms の遅延後、SDMMC をパワーオン状態にセットし、SDMMC_CK クロックを有効にします。
7. 74 SDMMC_CK サイクル後、最初のコマンドをカードに送信できるようになります。

図 718. カードサイクルパワー／パワーアップの図



55.6 ハードウェアフロー制御

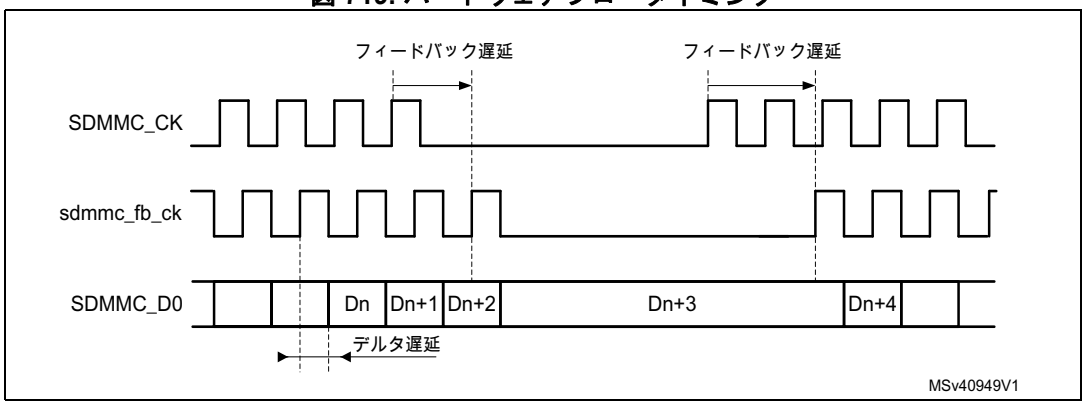
ハードウェアフロー制御機能は、FIFO アンダーラン（TX モード）およびオーバーラン（RX モード）エラーを回避するために使用します。

この機能は、データ転送中に SDMMC_CK を停止し、SDMMC ステートマシンをフリーズします。FIFO を使用してデータを送受信できない場合、データ転送がストールされます。送信 FIFO がハーフフルになるか、DATALENGHT に応じてすべてのデータが格納されるか、受信 FIFO がハーフエンプティになるまで、データ転送がストールされたままとなります。SDMMC_CK によってクロック供給されるステートマシンだけが停止し、AHB インタフェースは動作を続けます。したがって、フロー制御がアクティブであっても、FIFO がいっぱいになったり、空になることがあります。

ハードウェアフロー制御を有効にするには、HWFC_EN レジスタビットを 1 にセットする必要があります。リセット後、ハードウェアフロー制御は無効になります。

ハードウェアフロー制御は、SDMMC_Dn データが SDMMC_CK とサイクル調整されている場合のみ使用されます。t_{OP} および Dt_{OP} の遅延が 1 サイクルを超える SDR104 モードの場合など DLYB 遅延ブロックからの sdmmc_fb_ck が使用されている場合は、常にハードウェアフロー制御を使用できません。

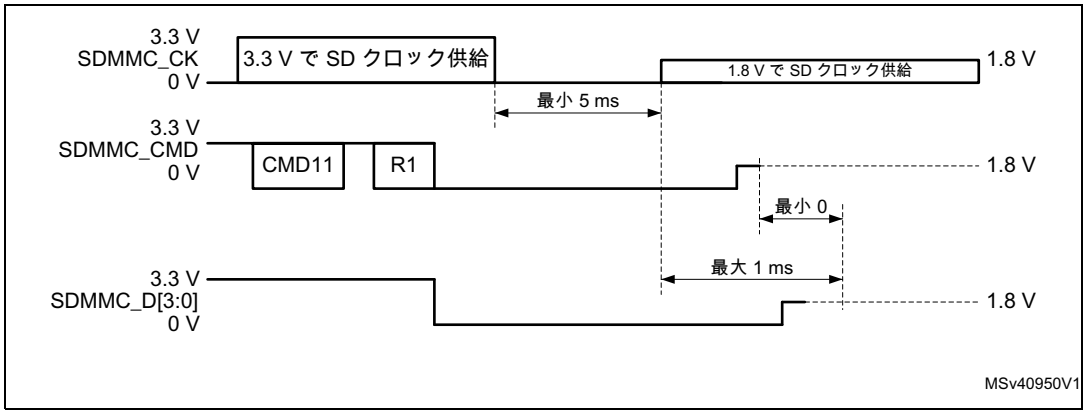
図 719. ハードウェアフロータイミング



55.7 超ハイスピードフェーズ I（UHS-I）の電圧スイッチ

UHS-I モード（SDR12、SDR25、SDR50、SDR104、および DDR50）は、1.8 V 信号のサポートが必須です。パワーアップ後、カードは 3.3 V モードで開始します。CMD11 が電圧スイッチシーケンスを呼び出し、1.8 V モードにします。電圧シーケンスが正常に完了すると、カードはデフォルトの SDR12 で UHS-I モードに移行し、カードの入力および出力タイミングが変わります。

図 720. CMD11 信号電圧スイッチシーケンス開始



信号電圧スイッチシーケンスを実行するには、次の手順が必要です。

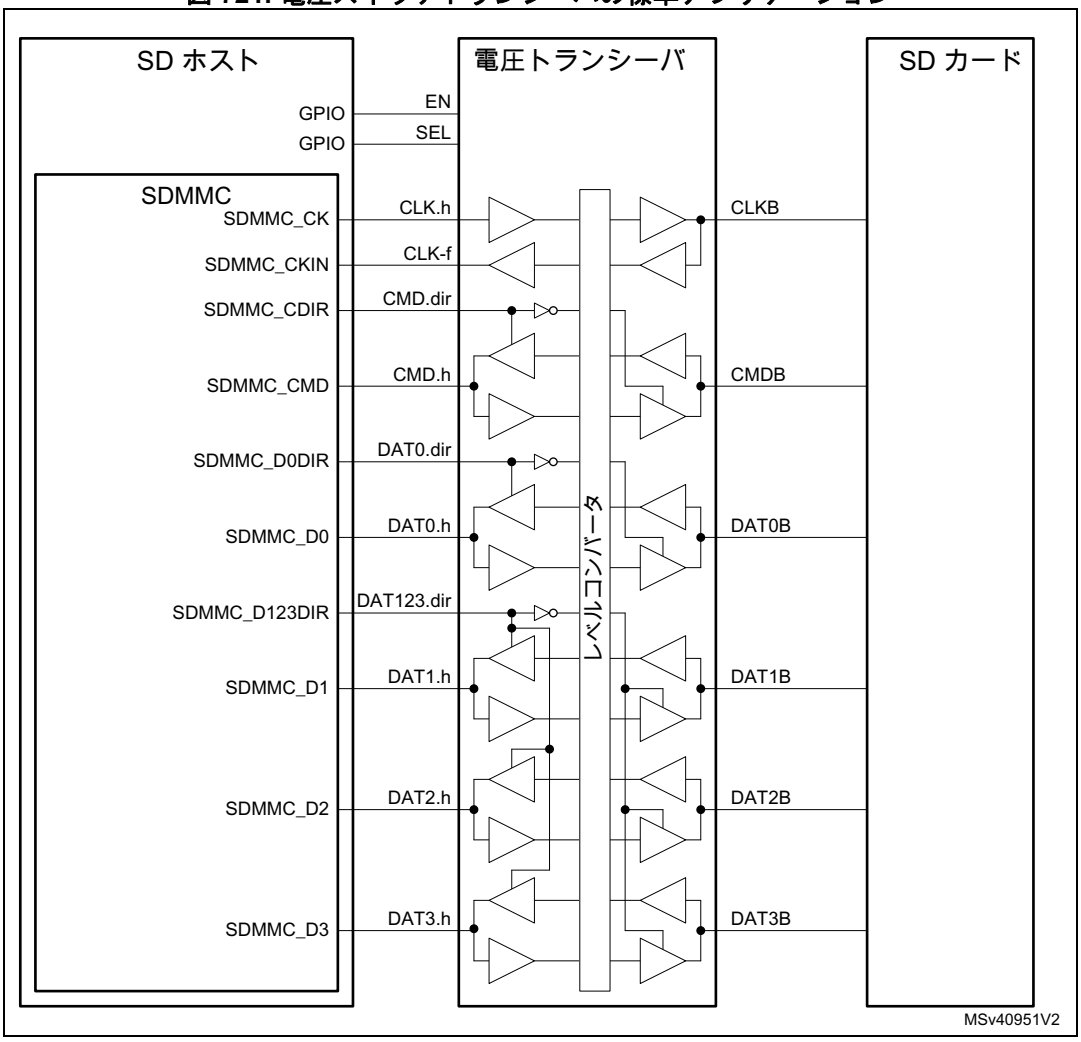
1. 電圧スイッチ手順を開始する前に、SDMMC_CK 出力周波数は、100～400 kHz の範囲でセットします。
2. ホストは、CMD11 を送信する前に VSWITCHEN ビットをセットして電圧スイッチを開始します。
3. カードが R1 レスポンスを返します。
   - レスポンス CRC がパスして、電圧スイッチ手順を続行する場合、ホストは電圧スイッチシーケンスが完了するまで CMD および SDMMC_D[3:0] 信号を駆動しません。レスポンス後の数サイクルで、SDMMC_CK は停止し、CKSTOP フラグがセットされます。
   - レスポンス CRC が失敗した場合（CCRCFAIL フラグ）、またはタイムアウトまでにレスポンスを受信しなかった場合（CTIMEOUT フラグ）、電圧スイッチ手順は停止します。
4. カードが、R1 レスポンス後の次のクロックで CMD および SDMMC_D[3:0] をローに駆動します。
5. ホストは、R1 レスポンスを受信した後、BUSYD0 レジスタビットで SDMMC_D0 ラインを監視します。SDMMC_D0 ラインは、レスポンス後 2 SDMMC_CK クロックサイクルでサンプリングされます。ファームウェアは、CKSTOP フラグの後に BUSYD0 レジスタビットを読み出す場合があります。
   - BUSYD0 がローで検出されると、ホスト FW は電圧レギュレータを 1.8 V に切り替え、SDMMC にレジスタビット VSWITCH をセットして電圧スイッチシーケンスのタイミングが重要なセクションを開始するよう命令します。ハードウェアは、最低 5 ms の間 SDMMC_CK をローに保って停止させ続けます。
   - BUSYD0 がハイで検出されると、ホストは電圧スイッチシーケンスをアボートして、カードにサイクルパワーします。
6. SDMMC_CK がローで検出された後、カードは信号電圧を 1.8 V に切り替え始めます。
7. 最低 5 ms 後に、ホスト SDMMC ハードウェアは SDMMC_CK を再開します。
8. SDMMC_CK の遷移を検出してから 1 ms 以内に、カードは最低 1 SDMMC_CK サイクルの間 CMD と DAT[3:0] をハイに駆動してから、CMD と DAT[3:0] の駆動を停止させます。
9. ホスト SDMMC ハードウェアでは、SDMMC_CK が再開してから 1 ms で SDMMC_D0 が BUSYD0 にサンプリングし、VSWEND フラグを生成します。
10. VSWEND フラグで、ホストは BUSYD0 レジスタビットを使って SDMMC_D0 ラインをチェックし、電圧スイッチシーケンスの完了を確認します。
    - BUSYD0 がハイで検出される場合、電圧スイッチが正常に完了しています。
    - BUSYD0 がローで検出されると、電圧スイッチに失敗し、ホストはカードパワーにサイクルパワーします。

SDMMC_CK を停止するための最低 5 ms の時間は、クロック分周回路 CLKDIV でセットされているとおり、内部のゲートされていない SDMMC_CK クロックから生じ、最大周波数は 25 MHz（SD モード）です。5ms を超える時間は、2^12 サイクル（10.24ms @ 400 kHz）でカウントされます。より低い SDMMC_CK 周波数がクロック分周回路 CLKDIV で選択されている場合、SDMMC_CK クロックが停止する時間は長くなります。

カードが SDMMC_Dn と SDMMC_CMD をハイに駆動する最大 1 ms の時間は、クロック分周回路 CLKDIV でセットされているとおり、内部のゲートされていない SDMMC_CK から生じ、最大周波数は 25 MHz（SD モード）です。SDMMC は、1ms を超える時間以降、2^9 サイクル（1.28ms @ 25MHz）でカウントしてラインをチェックします。より低い SDMMC_CK 周波数がクロック分周回路 CLKDIV で選択されている場合、ラインをチェックする時間は長くなります。

信号電圧レベルは、ST6G3244ME などの外部電圧変換トランシーバを通じてサポートされます。

図 721. 電圧スイッチトランシーバの標準アプリケーション



外部ドライバ（電圧スイッチトランシーバ）とインタフェース接続するために、標準信号とは別に、SDMMC は次の信号を使用します。

SDMMC_CKIN フィードバック入力クロック

SDMMC_CDIR CMD 信号の I/O 方向制御

SDMMC_D0DIR SDMMC_D0 信号の I/O 方向制御

SDMMC_D123DIR SDMMC_D1、SDMMC_D2 および SDMMC_D3 信号の I/O 方向制御

電圧トランシーバ信号 **EN** および **SEL** は、汎用 IO を通じて処理されます。

SDMMC_CDIR、SDMMC_D0DIR および SDMMC_D123DIR 信号の極性は、
SDMMC_POWER.DIRPOL 制御ビットで選択できます。

55.8 SDMMC レジスタ

デバイスは、AHB スレーブインタフェース経由でアクセス可能な 32 ビットの制御レジスタによってシステムと通信します。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。バイト（8 ビット）およびハーフワード（16 ビット）のアクセスは、AHB バスエラーを生成します。

55.8.1 SDMMC 電源制御レジスタ（SDMMC_POWER）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR POL	V SWITCH EN	V SWITCH	PWRCTRL[1:0]	
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **DIRPOL**：データおよびコマンドの方向信号の極性選択

このビットは、SDMMC がパワーオフ状態（PWRCTRL = 00）のときのみ書き込むことができます。

0：方向信号がローのときに電圧トランシーバの IO が出力として駆動されます。

1：方向信号がハイのときに電圧トランシーバの IO が出力として駆動されます。

ビット 3 **VSWITCHEN**：電圧スイッチ手順有効

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットは、電圧スイッチコマンドレスポンスの後 SDMMC_CK を停止するために使用されます。

0：コマンドレスポンスを正常に受信した後、SDMMC_CK クロックに変化はありません。

1：コマンドレスポンスを正常に受信した後、SDMMC_CK クロックが停止します。

ビット 2 **VSWITCH**：電圧スイッチシーケンス開始

このビットは、電圧スイッチシーケンスのタイミングの重要なセクションの開始に使用されます。

0：電圧スイッチシーケンスは開始されずアクティブではありません。

1：電圧スイッチシーケンスが開始されているかアクティブです。

ビット 1:0 **PWRCTRL[1:0]**：SDMMC の状態制御ビット

これらのビットは、SDMMC がパワーオン状態ではないとき（PWRCTRL ≠ 11）のみ書き込むことができます。

これらのビットは、SDMMC 信号の機能状態を定義するために使用されます。

00：リセット後、リセット：SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0] および SDMMC_CMD はハイインピーダンスになり、SDMMC_CK はローに駆動されます。
00 が書き込まれた場合、パワーオフ：SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK はハイに駆動されます。

01：予約済み。（01 が書き込まれた場合、PWRCTRL 値は変化しません）

10：パワーサイクル、SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK はローに駆動されます。

11：パワーオン：カードはクロック供給されます。最初の 74 SDMMC_CK サイクルは SDMMC が無効化されたままになります。74 サイクル後、SDMMC は有効になり、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK は SDMMC の動作に従って制御されます。

追加の書き込みは無視され、PWRCTRL 値は 11 のままとなります。

55.8.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

SDMMC_CLKCR レジスタは、SDMMC_CK の出力クロック、sdmmc_rx_ck 受信クロック、バス幅を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUS SPEED	DDR	HWFC_EN	NEG EDGE
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WID BUS[1:0]		Res.	PWR SAV	Res.	Res.	CLKDIV[9:0]									
rw	rw		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **SELCLKRX**：受信クロックの選択

これらのビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

00：sdmmc_io_in_ck が受信クロックとして選択されます。

01：SDMMC_CKIN フィードバッククロックが受信クロックとして選択されます。

10：sdmmc_fb_ck 調整フィードバッククロックが受信クロックとして選択されます。

11：予約済み（sdmmc_io_in_ck を選択）

ビット 19 **BUSSPEED**：DS、ハイスピード、SDR12、SDR25 および SDR50、DDR50、SDR104 からバス速度モードの選択

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

0：DS、ハイスピード、SDR12、SDR25 のバス速度モードが選択されます。

1：SDR50、DDR50、SDR104 のバス速度モードが選択されます。

ビット 18 **DDR**：データレート信号の選択

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

DDR レートは、4 ビットまたは 8 ビット幅のバスモードでのみ選択します。（WIDBUS > 00）。WIDBUS = 00（1 ビット幅バス）の場合、DDR = 1 は効果がありません。

DDR レートは、クロック分周が 1 より大きい場合のみ選択します。（CLKDIV > 0）。

0：SDR シングルデータレートシグナリング

1：DDR ダブルデータレートシグナリング

ビット 17 **HWFC_EN**：ハードウェアフロー制御イネーブル

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

0：ハードウェアフロー制御は無効です。

1：ハードウェアフロー制御は有効です。

ハードウェアフロー制御が有効なときの TXFIFOE および RXFIFOF フラグ変更の意味については、*セクション 55.8.11* の SDMMC ステータスレジスタの定義を参照してください。

ビット 16 **NEGEDGE**：データとコマンドの SDMMC_CK 位相反転選択ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

クロック分周 = 1（CLKDIV = 0）の場合、このビットは意味を持ちません。コマンドとデータは SDMMC_CK 立ち下がりエッジで変更されます。

クロック分周 >1（CLKDIV > 0）および DDR = 0 の場合：

0：- コマンドとデータは SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。

- SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

1：- コマンドとデータは SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで変更されます。

クロック分周 >1（CLKDIV > 0）および DDR = 1 の場合：

0：- コマンドは SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。

- データは SDMMC_CK エッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。
- SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

1：- コマンドは SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで変更されます。

- データは SDMMC_CK エッジに続く SDMMC_CK 立ち下がりエッジで変更されます。
- SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

ビット 15:14 **WIDBUS[1:0]**：ワイドバスモード有効ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

00：デフォルトの 1 ビット幅のバスモード：SDMMC_D0 使用（DDR をサポートしません）

01：4 ビット幅のバスモード：SDMMC_D[3:0] 使用

10：8 ビット幅のバスモード：SDMMC_D[7:0] 使用

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **PWRSAV**：節電構成ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

節電のため、バスがアイドル状態のときには、PWRSAV をセットして SDMMC_CK クロック出力を無効にできます。

0：SDMMC_CK クロックは常に有効です。

1：SDMMC_CK はバスがアクティブなときに限り有効です。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **CLKDIV[9:0]**：クロック分周ファクタ

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

このフィールドは、入力クロック（sdmmc_ker_ck）と出力クロック（SDMMC_CK）の間の分周ファクタを定義します。SDMMC_CK 周波数 = sdmmc_ker_ck / [2 * CLKDIV]。

000：SDMMC_CK 周波数 = sdmmc_ker_ck / 1（DDR をサポートしません）

001：SDMMC_CK 周波数 = sdmmc_ker_ck / 2

002：SDMMC_CK 周波数 = sdmmc_ker_ck / 4

0xx：同様に続く

080：SDMMC_CK 周波数 = sdmmc_ker_ck / 256

xxx：同様に続く

3FF：SDMMC_CK 周波数 = sdmmc_ker_ck / 2046

注： *1 SD/SDIO カードまたは MMC が識別モードのときには、SDMMC_CK 周波数は 400 kHz 未満である必要があります。*

2 クロック周波数は、相対カードアドレスがすべてのカードに割り当てられているときには、最大カードバス周波数に変更できます。

3 このレジスタへの各書き込みアクセス間には、7 sdmmc_hclk クロック周期以上が必要です。SDMMC_CK は、SD I/O カードの ReadWait インターバルの間、停止することができます。この場合、SDMMC_CLKCR レジスタは SDMMC_CK を制御しません。

55.8.3 SDMMC 引数レジスタ（SDMMC_ARGR）

アドレスオフセット：0x008

リセット値：0x0000 0000

SDMMC_ARGR レジスタは、コマンドメッセージの一部としてカードに送信される 32 ビットのコマンド引数を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMDARG[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMDARG[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CMDARG[31:0]**：コマンド引数

これらのビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。コマンドメッセージの一部としてカードに送られるコマンド引数です。コマンドが引数を含む場合、コマンドをコマンドレジスタに書き込む前に、このレジスタにロードする必要があります。

55.8.4 SDMMC コマンドレジスタ（SDMMC_CMDR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

SDMMC_CMDR レジスタは、コマンドインデックスおよびコマンドタイプビットを含みます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。コマンドタイプビットは、コマンドパスステートマシン（CPSM）を制御します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT EN	BOOT MODE	DT HOLD	CPSM EN	WAITP END	ウェイト INT	WAITRESP[1:0]		CMD STOP	CMD TRANS	CMDINDEX[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **CMDSUSPEND**：CPSM がコマンドをサスペンドまたはレジュームコマンドとして処理し、割り込み期間の開始／終了の信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

CMDSUSPEND = 1 および CMDTRANS = 0 のサスペンドコマンドで、レスポンスビット BS=0 の場合、割り込み期間を開始します。

CMDSUSPEND = 1 および CMDTRANS = 1 のレジュームコマンドとデータで、レスポンスビット DF=1 の場合、割り込み期間を終了します。

ビット 15 **BOOTEN**：ブートモード手順の有効

0：ブートモード手順は無効です。

1：ブートモード手順は有効です。

ビット 14 **BOOTMODE**：使用するブートモード手順の選択

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

0：通常ブートモード手順を選択します。

1：代替ブートモード手順を選択します。

ビット 13 **DTHOLD**：DPSM での新しいデータブロックの送受信の保留

このビットがセットされると、DPSM が Wait_S 状態から送信状態へ、Wait_R 状態から受信状態へ移行しません。

ビット 12 **CPSMEN**：コマンドパスステートマシン（CPSM）有効ビット

このビットは、CPSM がアイドル状態に移行すると、ファームウェアによって 1 が書き込まれ、ハードウェアによってクリアされます。

このビットがセットされると、CPSM が有効になります。

DTEN = 1 の場合、コマンドが転送されず、ブート手順も開始されません。CPSMEN は 0 にクリアされます。

SDMMC_CK を停止した ReadWait 状態の間、コマンドが送信されず、CPSMEN は 0 のままとなります。

ビット 11 **WAITPEND**：CPSM の DPSM からのデータ転送終了待ち（CmdPend 内部信号）。

このビットがセットされると、CPSM は、データ転送の終了を待ってから、コマンドの送信を開始します。

DTMODE = MMC ストリームデータ転送、WIDBUS = 1 ビット幅バスモード、DPSMACT = 1 かつ DTDIR = ホストからカードの場合のみ、WAITPEND は考慮されます。

ビット 10 **WAITINT**：CPSM の割り込みリクエスト待ち

このビットがセットされると、CPSM は、コマンドタイムアウトを無効にして、割り込みリクエスト（レスポンス）を待ちます。

CPSM ウェイトステートでこのビットがクリアされると、割り込みモードのアボートが発生します。

ビット 9:8 **WAITRESP[1:0]**：レスポンスビット待ち

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

これらは、CPSM がレスポンスを待つかどうか、および、待つ場合にはレスポンスの種類を設定するために使用されます。

00：レスポンスなし。CMDSENT フラグを待ちます。

01：ショートレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

10：ショートレスポンス。CMDREND フラグ（CRC なし）を待ちます。

11：ロングレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

ビット 7 **CMDSTOP**：CPSM がコマンドを送信停止コマンドとして処理し、DPSM に Abort 信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSM はコマンド送信時に Abort 信号を DPSM に発行します。

ビット 6 **CMDTRANS**：CPSM がコマンドをデータ転送コマンドとして処理し、割り込み期間を停止し、DPSM に DataEnable 信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSM はコマンド送信時に割り込み期間の終了と DataEnable 信号を DPSM に発行します。

ビット 5:0 **CMDINDEX[5:0]**：コマンドインデックス

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。

注： ***1 このレジスタへの各書き込みアクセス間には、7 sdmmc_hclk クロック周期以上が必要です。***

2 MultiMediaCard はショートレスポンス（48 ビット）またはロングレスポンス（136 ビット）の 2 種類のレスポンスを送信できます。SD カードと SD I/O カードはショートレスポンスのみ送信でき、引数はレスポンスのタイプによって異なることがあります。ソフトウェアは、送信コマンドに応じてレスポンスのタイプを区別します。

55.8.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMDR）

アドレスオフセット：0x010

リセット値：0x0000 0000

SDMMC_RESPCMDR レジスタは、最後に受信したコマンドレスポンスのコマンドインデックスフィールドを含みます。コマンドレスポンス送信にコマンドインデックスフィールドが含まれていない場合（ロングまたは OCR レスポンス）、RESPCMD フィールドは未知ですが、111111b（レスポンスの予約フィールドの値）が含まれている必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **RESPCMD[5:0]**：レスポンスコマンドインデックス
読み出し専用ビットフィールドです。最後に受信したコマンドレスポンスのコマンドインデックスを含みます。

55.8.6 SDMMC レスポンス 1～4 レジスタ（SDMMC_RESPxR）（x = 1～4）

アドレスオフセット：(0x010 + (4 × x))

リセット値：0x0000 0000

SDMMC_RESP1/2/3/4R レジスタは、受信したレスポンスの一部であるカードのステータスを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CARDSTATUSx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CARDSTATUSx[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **CARDSTATUSx[31:0]**：*表 447* を参照してください。

カードステータスサイズは、レスポンスのタイプによって 32 または 128 ビットです。

表 447. レスポンスタイプと SDMMC_RESPxR レジスタ

レジスタ(1)	ショートレスポンス	ロングレスポンス
SDMMC_RESP1R	カードステータス [31:0]	カードステータス [127:96]
SDMMC_RESP2R	すべて 0	カードステータス [95:64]
SDMMC_RESP3R	すべて 0	カードステータス [63:32]
SDMMC_RESP4R	すべて 0	カードステータス [31:0](2)

1. カードステータスの最上位ビットから先に受信します。
2. SDMMC_RESP4R レジスタ LSB は常に 0 です。

55.8.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）

アドレスオフセット：0x024

リセット値：0x0000 0000

SDMMC_DTIMER レジスタは、データタイムアウト期間（カードバスクロック周期単位）を含みます。

カウンタは SDMMC_DTIMER レジスタから値をロードして、データパスステートマシン（DPSM）が Wait_R またはビジー状態に入ったときに、デクリメントを開始します。DPSM がこのどちらかの状態のときにタイマが 0 に達した場合、タイムアウトステータスフラグがセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATATIME[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATATIME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATATIME[31:0]**：データおよび R1b ビジータイムアウト期間

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

カードバスクロック周期で表されたデータおよび R1b ビジータイムアウト期間です。

注： ***データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。***

55.8.8 SDMMC データ長レジスタ（SDMMC_DLENR）

アドレスオフセット：0x028

リセット値：0x0000 0000

SDMMC_DLENR レジスタは、転送されるデータバイト数を含みます。値は、データ転送が開始されたときに、データカウンタにロードされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATALENGTH[24:0]**：データ長の値

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

転送されるデータバイト数です。

DDR = 1 の場合、DATALENGTH は 2 の倍数に切り詰められます。（最後の奇数バイトは転送されません）

DATALENGTH = 0 の場合、CPSMEN でリクエストされたとき、データは転送されず、CMDTRANS = 1 ではコマンドは転送されません。DTEN および CPSMEN は 0 にクリアされます。

注： ***ブロックデータ転送では、データ長レジスタの値はブロックサイズの倍数である必要があります（SDMMC_DCTRL を参照してください）。データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。***

SDMMC マルチバイト転送の場合、データ長レジスタ内の値は 1 から 512 までででなければなりません。

55.8.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）

アドレスオフセット：0x02C

リセット値：0x0000 0000

SDMMC_DCTRL レジスタは、データパスステートマシン（DPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FIFO RST	BOOT ACKE N	SDIO EN	RW MOD	RW STOP	RW START	DBLOCKSIZE[3:0]				DTMODE		DTDIR	DTEN
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **FIFORST**：FIFO リセットで残りのデータを一掃します。

このビットは、IDMAEN= 0 および DPSM がアクティブ（DPSMACT = 1）のときのみファームウェアで書き込むことができます。このビットは、転送エラーまたは転送保留が発生したときのみ有効になります。

0：FIFO に影響がありません。

1：残りのデータを一掃して FIFO ポインタをリセットします。このビットは、DPSM がインアクティブ（DPSMACT = 0）になったとき、自動的にハードウェアによって 0 にクリアされます。

ビット 12 **BOOTACKEN**：ブート確認応答の受信を有効にします。

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ブート確認応答は無効になり、受信の予定はありません。

1：ブート確認応答は有効になり、受信の予定があります。

ビット 11 **SDIOEN**：SD I/O 割り込み有効機能

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされると、DPSM は SD I/O カード固有の割り込み操作を可能にします。

ビット 10 **RWMOD**：読み出しウェイトモード

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：SDMMC_D2 を使用する読み出しウェイト制御です。

1：SDMMC_CK を停止する読み出しウェイト制御です。

ビット 9 **RWSTOP**：読み出しウェイト停止

このビットは、ファームウェアによって書き込まれ、DPSM が READ_WAIT 状態から WAIT_R または IDLE 状態に移行すると、ハードウェアによって自動クリアされます。

0：読み出しウェイトはありません。

1：DPSM が READ_WAIT 状態の場合、読み出しウェイト停止が有効になります。

ビット 8 **RWSTART**：読み出しウェイト開始

このビットがセットされると、読み出しウェイト操作が開始されます。

ビット 7:4 **DBLOCKSIZE[3:0]**：データブロックサイズ

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ブロックデータ転送モードが選択されているときのデータブロック長を定義します。

0000：（0 10 進）ロック長 = 2^0 = 1 バイト

0001：（1 10 進）ロック長 = 2^1 = 2 バイト

0010：（2 10 進）ロック長 = 2^2 = 4 バイト

0011：（3 10 進）ロック長 = 2^3 = 8 バイト

0100：（4 10 進）ロック長 = 2^4 = 16 バイト

0101：（5 10 進）ロック長 = 2^5 = 32 バイト

0110：（6 10 進）ロック長 = 2^6 = 64 バイト

0111：（7 10 進）ロック長 = 2^7 = 128 バイト

1000：（8 10 進）ロック長 = 2^8 = 256 バイト

1001：（9 10 進）ロック長 = 2^9 = 512 バイト

1010：（10 10 進）ロック長 = 2^{10} = 1024 バイト

1011：（11 10 進）ロック長 = 2^{11} = 2048 バイト

1100：（12 10 進）ロック長 = 2^{12} = 4096 バイト

1101：（13 10 進）ロック長 = 2^{13} = 8192 バイト

1110：（14 10 進）ロック長 = 2^{14} = 16384 バイト

1111：（15 10 進）予約済み

DATALENGTH が DBLOCKSIZE の整数倍ではない場合、転送されたデータは DBLOCKSIZE の倍数に切り捨てられます。（残りのデータは転送されません）

DDR = 1 の場合、DBLOCKSIZE = 0000 は使用されません。（データは転送されません）

ビット 3:2 **DTMODE**：データ転送モード選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

00：ブロック数で終了するブロックデータ転送

01：SDIO マルチバイトデータ転送

10：MMC ストリームデータ転送（WIDBUS は 1 ビット幅のバスモードを選択します）

11：STOP_TRANSMISSION コマンドで終了するブロックデータ転送（DTEN で開始するデータ転送では使用しません）

ビット 1 **DTDIR**：データ転送方向選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ホストからカードへ

1：カードからホストへ

ビット 0 **DTEN**：データ転送有効ビット

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。このビットは、データ転送が完了したとき、ハードウェアによってクリアされます。

このビットは、SD カードや eMMC カードと使用しない場合など、関連するデータ転送コマンドが使用されないときのみ、データ転送に使用されます。

0：CPSM データ転送コマンドなしでデータ転送を開始しません。

1：CPSM データ転送コマンドなしでデータ転送を開始します。

55.8.10 SDMMC データカウンタレジスタ（SDMMC_DCNTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

SDMMC_DCNTR レジスタは、DPSM がアイドル状態から Wait_R または Wait_S 状態へ移行すると、データ長レジスタから値をロードします（SDMMC_DLENR を参照してください）。データが転送されると、カウンタは 0 に達するまで値をデクリメントします。その後、DPSM はアイドル状態へ移行し、エラーがない場合、データステータス終了フラグ（DATAEND）がセットされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATACOUNT[24:0]**：データカウント値

読み出すと、残りの転送バイト数が返されます。書き込み操作は何の影響も与えません。

注： ***このレジスタは、データ転送を完了または保留したときだけ読み出してください。エラーイベントの後に読み出す場合、読み出しデータカウント値は実際に転送されたデータバイト数と異なる場合があります。***

55.8.11 SDMMC ステータスレジスタ（SDMMC_STAR）

アドレスオフセット：0x034

リセット値：0x0000 0000

SDMMC_STAR レジスタは、読み出し専用レジスタです。2 種類のフラグを含みます。

- スタティックフラグ（ビット [29,21,11:0]）：これらのビットは、SDMMC 割り込みクリアレジスタ（SDMMC_ICR を参照）に書き込んでクリアされるまで、アサートされたままです。
- ダイナミックフラグ（ビット [20:12]）：これらのビットは、基礎回路の状態に応じて状態が変わります（たとえば、FIFO フルおよびエンプティフラグは、データが FIFO に書き込まれるとアサートおよびネゲートされます）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMA BTC	IDMA TE	CK STOP	VSW END	ACK TIME OUT	ACK FAIL	SDIOIT	BUSY D0END	BUSY D0	RX FIFOE	TX FIFOE	RX FIFOF	TX FIFOF
			r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RX FIFO HF	TX FIFO HE	CPSM ACT	DPSM ACT	D ABOR T	DBCK END	DHOLD	DATA END	CMD SENT	CMDR END	RX OVERR	TX UNDER R	D TIME OUT	C TIME OUT	DCRC FAIL	CCRC FAIL
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTC**：IDMA バッファ転送完了。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 27 **IDMATE**：IDMA 転送エラー。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 26 **CKSTOP**：SDMMC_CK が電圧スイッチ手順で停止されます。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 25 **VSWEND**：電圧スイッチの重要なタイミングセクションの完了。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 24 **ACKTIMEOUT**：ブート確認応答タイムアウト。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 23 **ACKFAIL**：ブート確認応答が受信されました（ブート確認応答チェック失敗）。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 22 **SDIOIT**：SDIO 割り込みが受信されました。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 21 **BUSYD0END**：CMD レスポンス検出後の SDMMC_D0 ビジーの終了。

これは、CMD レスポンス後のビジーの終了のみ示します。このビットは、データ転送によってビジー信号を送信しません。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

0：カード SDMMC_D0 信号は、ビジーからビジーでない状態への変更を伝えません。

1：カード SDMMC_D0 信号は、ビジーからビジーでない状態に変更されました。

ビット 20 **BUSYD0**：CMD レスポンスの終了時と、CMD レスポンス後の 2 SDMMC_CK サイクルでサンプリングされた SDMMC_D0 ラインの反転された値（ビジー）。

このビットは、SDMMCD0 ラインがビジーからビジーでない状態に変わる際に、ビジーでない状態にリセットされます。このビットは、データ転送によってビジー信号を送信しません。これはハードウェアステータスフラグのみで、割り込みは生成されません。

0：カードが SDMMC_D0 でビジーでない状態を伝えます。

1：カードが SDMMC_D0 でビジー状態であることを伝えます。

ビット 19 **RXFIFOE**：受信 FIFO エンプティ

これはハードウェアステータスフラグのみで、割り込みは生成されません。このビットは、1 つの FIFO の位置がフルになったときにクリアされます。

ビット 18 **TXFIFOE**：送信 FIFO エンプティ

このビットは、1 つの FIFO の位置がフルになったときにクリアされます。

ビット 17 **RXFIFOF**：受信 FIFO フル

このビットは、1 つの FIFO の位置がエンプティになったときにクリアされます。

ビット 16 **TXFIFOF**：送信 FIFO フル

これはハードウェアステータスフラグのみで、割り込みは生成されません。このビットは、1 つの FIFO の位置がエンプティになったときにクリアされます。

ビット 15 **RXFIFOHF**：受信 FIFO ハーフフル

FIFO には少なくともワード数の半分入っています。このビットは、FIFO がハーフ +1 空になったときにクリアされます。

ビット 14 **TXFIFOHE**：送信 FIFO ハーフエンプティ

FIFO に少なくともワード数の半分を書き込み可能です。このビットは、FIFO がハーフ +1 フルになったときにクリアされます。

ビット 13 **CPSMACT**：コマンドパスステートマシンがアイドル状態などではなくアクティブです。

これはハードウェアステータスフラグのみで、割り込みは生成されません。

ビット 12 **DPSMACT**：データパスステートマシンがアイドル状態などではなくアクティブです。

これはハードウェアステータスフラグのみで、割り込みは生成されません。

ビット 11 **DABORT**：データ転送が CMD12 によってアボートされます。

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 10 **DBCKEND**：データブロック送受信

（CRC 確認成功）かつ DPSM が READWAIT 状態へ移行します。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 9 **DHOLD**：データ転送保留

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 8 **DATAEND**：データ転送の正常終了

（データカウンタ、DATACOUNT は 0 で、発生したエラーはありません）。割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 7 **CMDSENT**：コマンド送信（レスポンス不要）

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 6 **CMDREND**：コマンドレスポンス受信（CRC 確認成功、または CRC なし）。

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 5 **RXOVERR**：受信 FIFO オーバーランエラーまたは IDMA 書き込み転送エラー

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 4 **TXUNDERR**：送信 FIFO アンダーランエラーまたは IDMA 読み出し転送エラー

割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 3 **DTIMEOUT**：データタイムアウト
割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 2 **CTIMEOUT**：コマンドレスポンスタイムアウト
割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。
コマンドタイムアウト期間は、固定値の 64 SDMMC_CK クロック周期です。

ビット 1 **DCRCFAIL**：データブロック送受信（CRC 確認失敗）
割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

ビット 0 **CCRCFAIL**：コマンドレスポンス受信（CRC 確認失敗）
割り込みフラグは、SDMMC_ICR で対応する割り込みクリアビットに書き込むことでクリアされます。

*注： **FIFO 割り込みフラグは、IDMA モード使用時 SDMMC_MASKR でマスクされます。***

55.8.12 SDMMC 割り込みクリアレジスタ（SDMMC_ICR）

アドレスオフセット：0x038

リセット値：0x0000 0000

SDMMC_ICR レジスタは、書き込み専用レジスタです。ビットに 1 を書き込むと、SDMMC_STAR ステータスレジスタの対応するビットがクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMA BTCC	IDMA TEC	CK STOP C	VSW ENDC	ACK TIME OUTC	ACK FAILC	SDIO ITC	BUSY D0 ENDC	Res.	Res.	Res.	Res.	Res.
			rw	rw	rw	rw	rw	rw	rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	D ABOR TC	DBCK ENDC	DHOLD C	DATA ENDC	CMD SENTC	CMDR ENDC	RX OVERR C	TX UNDER RC	D TIME OUTC	C TIME OUTC	DCRC FAILC	CCRC FAILC
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:39 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTCC**：IDMA バッファ転送完了クリアビット
IDMABTC フラグをクリアするために、ソフトウェアによってセットされます。
0：IDMABTC はクリアされません。
1：IDMABTC はクリアされます。

ビット 27 **IDMATEC**：IDMA 転送エラークリアビット
IDMATE フラグをクリアするために、ソフトウェアによってセットされます。
0：IDMATE はクリアされません。
1：IDMATE はクリアされます。

ビット 26 **CKSTOPC**：CKSTOP フラグクリアビット
CKSTOP フラグをクリアするために、ソフトウェアによってセットされます。
0：CKSTOP はクリアされません。
1：CKSTOP はクリアされます。

ビット 25 **VSWENDC**：VSWEND フラグクリアビット
VSWEND フラグをクリアするために、ソフトウェアによってセットされます。
0：VSWEND はクリアされません。
1：VSWEND はクリアされます。

ビット 24 **ACKTIMEOUTC**：ACKTIMEOUT フラグクリアビット
ACKTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：ACKTIMEOUT はクリアされません。
1：ACKTIMEOUT はクリアされます。

ビット 23 **ACKFAILC**：ACKFAIL フラグクリアビット
ACKFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：ACKFAIL はクリアされません。
1：ACKFAIL はクリアされます。

ビット 22 **SDIOITC**：SDIOIT フラグクリアビット
SDIOIT フラグをクリアするために、ソフトウェアによってセットされます。
0：SDIOIT はクリアされません。
1：SDIOIT はクリアされます。

ビット 21 **BUSYD0ENDC**：BUSYD0END フラグクリアビット
BUSYD0END フラグをクリアするために、ソフトウェアによってセットされます。
0：BUSYD0END はクリアされません。
1：BUSYD0END がクリアされます。

ビット 20:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DABORTC**：DABORT フラグクリアビット
DABORT フラグをクリアするために、ソフトウェアによってセットされます。
0：DABORT はクリアされません。
1：DABORT はクリアされます。

ビット 10 **DBCKENDC**：DBCKEND フラグクリアビット
DBCKEND フラグをクリアするために、ソフトウェアによってセットされます。
0：DBCKEND はクリアされません。
1：DBCKEND はクリアされます。

ビット 9 **DHOLDC**：DHOLD フラグクリアビット
DHOLD フラグをクリアするために、ソフトウェアによってセットされます。
0：DHOLD はクリアされません。
1：DHOLD はクリアされます。

ビット 8 **DATAENDC**：DATAEND フラグクリアビット
DATAEND フラグをクリアするために、ソフトウェアによってセットされます。
0：DATAEND はクリアされません。
1：DATAEND はクリアされます。

ビット 7 **CMDSENTC**：CMDSENT フラグクリアビット
CMDSENT フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDSENT はクリアされません。
1：CMDSENT はクリアされます。

ビット 6 **CMDRENDC**：CMDREND フラグクリアビット
CMDREND フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDREND はクリアされません。
1：CMDREND はクリアされます。

ビット 5 **RXOVERRC**：RXOVERR フラグクリアビット
RXOVERR フラグをクリアするために、ソフトウェアによってセットされます。
0：RXOVERR はクリアされません。
1：RXOVERR はクリアされます。

ビット 4 **TXUNDERRC**：TXUNDERR フラグクリアビット
TXUNDERR フラグをクリアするために、ソフトウェアによってセットされます。
0：TXUNDERR はクリアされません。
1：TXUNDERR はクリアされます。

ビット 3 **DTIMEOUTC**：DTIMEOUT フラグクリアビット
DTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：DTIMEOUT はクリアされません。
1：DTIMEOUT はクリアされます。

ビット 2 **CTIMEOUTC**：CTIMEOUT フラグクリアビット
CTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：CTIMEOUT はクリアされません。
1：CTIMEOUT はクリアされます。

ビット 1 **DCRCFAILC**：DCRCFAIL フラグクリアビット
DCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：DCRCFAIL はクリアされません。
1：DCRCFAIL はクリアされます。

ビット 0 **CCRCFAILC**：CCRCFAIL フラグクリアビット
CCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：CCRCFAIL はクリアされません。
1：CCRCFAIL はクリアされます。

55.8.13 SDMMC マスクレジスタ（SDMMC_MASKR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

割り込みマスクレジスタは、対応するビットを 1 にセットすることによって、割り込みリクエストを生成するステータスフラグを決めます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMABTCIE	Res.	CKSTOPIE	VSWENDIE	ACKTIMEOUTIE	ACKFAILIE	SDIOITIE	BUSYD0ENDIE	Res.	Res.	TXFIFOEIE	RXFIFOFIE	Res.
			rw		rw	rw	rw	rw	rw	rw			rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFIFOHFIE	TXFIFOHEIE	Res.	Res.	DABORTIE	DBCKENDIE	DHOLDIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTCIE**：IDMA バッファ転送完了割り込み有効

IDMA がメモリバッファに属するすべてのデータを転送したときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：IDMA バッファ転送完了割り込みは無効です。

1：IDMA バッファ転送完了割り込みは有効です。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **CKSTOPIE**：電圧スイッチクロック停止割り込み有効

電圧スイッチクロック停止による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチクロック停止割り込みは無効です。

1：電圧スイッチクロック停止割り込みは有効です。

ビット 25 **VSWENDIE**：電圧スイッチの重要なタイミングセクション完了割り込み有効

電圧スイッチの重要なタイミングセクションが完了したときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチの重要なタイミングセクション完了割り込みは無効です。

1：電圧スイッチの重要なタイミングセクション完了割り込みは有効です。

ビット 24 **ACKTIMEOUTIE**：確認応答タイムアウト割り込み有効

確認応答タイムアウトによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答タイムアウト割り込みは無効です。

1：確認応答タイムアウト割り込みは有効です。

ビット 23 **ACKFAILIE**：確認応答失敗割り込み有効

確認応答失敗による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答失敗割り込みは無効です。

1：確認応答失敗割り込みは有効です。

ビット 22 **SDIOITIE**：SDIO モード割り込み受信割り込み有効

SDIO モード割り込みを受信したときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：SDIO モード割り込み受信割り込みは無効です。

1：SDIO モード割り込み受信割り込みは有効です。

ビット 21 **BUSYD0ENDIE**：BUSYD0END 割り込み有効

SDMMC_D0 信号が CMD レスポンスの後にビジーからビジーではない状態に変更したときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：BUSYD0END 割り込みは無効です。

1：BUSYD0END 割り込みは有効です。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TXFIFOEIE**：Tx FIFO エンプティ割り込み有効

Tx FIFO エンプティによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO エンプティ割り込みは無効です。

1：Tx FIFO エンプティ割り込みは有効です。

ビット 17 **RXFIFOFIE**：Rx FIFO フル割り込み有効

Rx FIFO がフルになったときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO フル割り込みは無効です。

1：Rx FIFO フル割り込みは有効です。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **RXFIFOHFIE**：Rx FIFO ハーフフル割り込み有効

Rx FIFO がハーフフル状態になったときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO ハーフフル割り込みは無効です。

1：Rx FIFO ハーフフル割り込みは有効です。

ビット 14 **TXFIFOHEIE**：Tx FIFO ハーフエンプティ割り込み有効

Tx FIFO がハーフエンプティ状態になったときに生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO ハーフエンプティ割り込みは無効です。

1：Tx FIFO ハーフエンプティ割り込みは有効です。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DABORTIE**：データ転送アボート割り込み有効

アボートされているデータ転送による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ転送アボート割り込みは無効です。

1：データ転送アボート割り込みは有効です。

ビット 10 **DBCKENDIE**：データブロックエンド割り込み有効

データブロックエンドによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データブロックエンド割り込みは無効です。

1：データブロックエンド割り込みは有効です。

ビット 9 **DHOLDIE**：データホールド割り込み有効

DPSM の Wait_S 状態で新しいデータの送信が保留されている場合に生成される割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データホールド割り込みは無効です。

1：データホールド割り込みは有効です。

ビット 8 **DATAENDIE**：データエンド割り込み有効

データエンドによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データエンド割り込みは無効です。

1：データエンド割り込みは有効です。

ビット 7 **CMDSENTIE**：コマンド送信割り込み有効

コマンド送信による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド送信割り込みは無効です。

1：コマンド送信割り込みは有効です。

ビット 6 **CMDRENDIE**：コマンドレスポンス受信割り込み有効

コマンドレスポンスの受信による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドレスポンス受信割り込みは無効です。

1：コマンドレスポンス受信割り込みは有効です。

ビット 5 **RXOVERRIE**：Rx FIFO オーバーランエラー割り込み有効

Rx FIFO オーバーランエラーによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO オーバーランエラー割り込みは無効です。

1：Rx FIFO オーバーランエラー割り込みは有効です。

ビット 4 **TXUNDERRIE**：Tx FIFO アンダーランエラー割り込み有効

Tx FIFO アンダーランエラーによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO アンダーランエラー割り込みは無効です。

1：Tx FIFO アンダーランエラー割り込みは有効です。

ビット 3 **DTIMEOUTIE**：データタイムアウト割り込み有効

データタイムアウトによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データタイムアウト割り込みは無効です。

1：データタイムアウト割り込みは有効です。

ビット 2 **CTIMEOUTIE**：コマンドタイムアウト割り込み有効

コマンドタイムアウトによる割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドタイムアウト割り込みは無効です。

1：コマンドタイムアウト割り込みは有効です。

ビット 1 **DCRCFAILIE**：データ CRC 失敗割り込み有効

データ CRC 失敗による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ CRC 失敗割り込みは無効です。

1：データ CRC 失敗割り込みは有効です。

ビット 0 **CCRCFAILIE**：コマンド CRC 失敗割り込み有効

コマンド CRC 失敗による割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド CRC 失敗割り込みは無効です。

1：コマンド CRC 失敗割り込みは有効です。

55.8.14 SDMMC 確認応答タイマレジスタ（SDMMC_ACKTIMER）

アドレスオフセット：0x040

リセット値：0x0000 0000

SDMMC_ACKTIMER レジスタは、確認応答タイムアウト期間（SDMMC_CK バスクロック周期単位）を含みます。

カウンタは SDMMC_ACKTIMER レジスタから値をロードして、データパスステートマシン（DPSM）が Wait_Ack 状態に入ったときに、デクリメントを開始します。DPSM がこの状態のときにタイマが 0 に達した場合、確認応答タイムアウトステータスフラグがセットされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **ACKTIME[24:0]**：ブート確認応答タイムアウト期間

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。カードバスクロック周期で表されたブート確認応答タイムアウト期間です。

注： ***データ転送は、データ制御レジスタに書き込まれる前に、確認応答タイマレジスタに書き込まれなければなりません。***

55.8.15 SDMMC データ FIFO レジスタ（SDMMC_FIFOR）

アドレスオフセット：0x080 から 0x0BC

リセット値：0x0000 0000

受信および送信 FIFO は、ワード（32 ビット）幅のレジスタとしてのみ読み書きできます。FIFO は、連続したアドレスに 16 個のエントリを含みます。このため、CPUはソフトウェアによるFIFOへの読み書きを行うことができます。

ハーフワードまたはバイトアクセスで SDMMC_FIFOR にアクセスすると、AHB バスフォールトが生成されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIFODATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIFODATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **FIFODATA[31:0]**：受信および送信 FIFO データ

このレジスタは、DPSM がアクティブ（DPSMACT = 1）のときのみファームウェアで読み書きできます。FIFO データは、32 ビットワードの 16 個のエントリを占有します。

55.8.16 SDMMC DMA 制御レジスタ（SDMMC_IDMACTRLR）

アドレスオフセット：0x050

リセット値：0x0000 0000

受信および送信 FIFO は、32 ビット幅のレジスタとして読み書きできます。FIFO は、32 の連続したアドレスに 32 個のエントリを含みます。このため、CPUはソフトウェアによるFIFOへの読み書きを行うことができます。

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **IDMABACT**：ダブルバッファモードのアクティブなバッファ表示

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。このビットは、IDMA が有効であるときに、ハードウェアによってトグルされます。

0：IDMA が有効である場合、buffer0 を使用し、IDMABASE0 へのファームウェア書き込みアクセスは禁止されます。

1：IDMA が有効である場合、buffer1 を使用し、IDMABASE1 へのファームウェア書き込みアクセスは禁止されます。

ビット 1 **IDMABMODE**：バッファモード選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：シングルバッファモード

1：ダブルバッファモード

ビット 0 **IDMAEN**：IDMA 有効

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：IDMA は無効です。

1：IDMA は有効です。

55.8.17 SDMMC IDMA バッファサイズレジスタ（SDMMC_IDMABSIZER）

アドレスオフセット：0x054

リセット値：0x0000 0000

ダブルバッファ設定では、SDMMC_IDMABSIZER レジスタにバッファサイズが格納されます。

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:5 **IDMABNDT[7:0]**：バッファごとの転送数

8 ビット値は、8 倍にすると 32 ビットワードでのバッファのサイズになり、32 倍にするとバイト単位でのバッファのサイズになります。

例：IDMABNDT = 0x01：バッファサイズ = 8 ワード = 32 バイト。

これらのビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

55.8.18 SDMMC IDMA バッファ 0 ベースアドレスレジスタ（SDMMC_IDMABASE0R）

アドレスオフセット：0x058

リセット値：0x0000 0000

SDMMC_IDMABASE0R レジスタには、シングルバッファ設定でメモリバッファベースアドレス、ダブルバッファ設定でバッファ 0 ベースアドレスが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABASE0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABASE0[15:0]															
rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r

ビット 31:0 **IDMABASE0[31:0]** : バッファ 0 メモリベースアドレスビット [31:2] は、ワードで調整されます（ビット [1:0] は常に 0 で読み出し専用です）。

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときファームウェアによって書き込むことができ、DPSM がアクティブ（DPSMACT = 1）かつメモリバッファ 0 がインアクティブ（IDMABACT = ‘1’）のときはファームウェアによって動的に書き込むことができます。

55.8.19 SDMMC IDMA バッファ 1 ベースアドレスレジスタ（SDMMC_IDMABASE1R）

アドレスオフセット：0x05C

リセット値：0x0000 0000

SDMMC_IDMABASE1R レジスタには、ダブルバッファ設定の 2 番目のバッファメモリベースアドレスが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABASE1[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABASE1[15:0]															
rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r

ビット 31:0 **IDMABASE1[31:0]** : バッファ 1 メモリベースアドレスは、ワードで調整されます（ビット [1:0] は常に 0 で読み出し専用です）。

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアによって書き込むことができ、DPSM がアクティブ（DPSMACT = 1）かつメモリバッファ 1 がインアクティブ（IDMABACT = ‘0’）のときはファームウェアによって動的に書き込むことができます。

55.8.20 SDMMC レジスタマップ

次の表に SDMMC レジスタの一覧を示します。

表 448. SDMMC レジスタマップ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SDMMC_ POWER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIRPOL	VSWITCHEN	VSWITCH	PWRCTRL[1:0]	
	リセット値																												0	0	0	0	0
0x04	SDMMC_ CLKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUSSPEED	DDR	HWFC_EN	NEGEDGE	WIDBUS[1:0]		Res.	PWRSAV	Res.	Res.	CLKDIV[9:0]									
	リセット値											0	0	0	0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0
0x08	SDMMC_ ARGR	CMDARG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SDMMC_ CMDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDSUSPEND	BOOTEN	BOOTMODE	DTHOLD	CPSMEN	WAITPEND	WAITINT	WAITRESP[1:0]		CMDSTOP	CMDTRANS	CMDINDEX[5:0]					
	リセット値																																
0x10	SDMMC_ RESPCMDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
	リセット値																											0	0	0	0	0	0
0x14	SDMMC_ RESP1R	CARDSTATUS1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SDMMC_ RESP2R	CARDSTATUS2[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	SDMMC_ RESP3R	CARDSTATUS3[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SDMMC_ RESP4R	CARDSTATUS4[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SDMMC_ DTIMER	DATATIME[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	SDMMC_ DLENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATALENGTH[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	SDMMC_ DCTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFORST	BOOTACKEN	SDIOEN	RWMOD	RWSTOP	RWSTART	DBLOCK SIZE[3:0]				DTMODE	DTDIR	DTEN
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 448. SDMMC レジスタマップ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	SDMMC_DCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	SDMMC_STAR	Res.	Res.	Res.	IDMABTC	IDMATE	CKSTOP	VSWEND	ACKTIMEOUT	ACKFAIL	SDIOIT	BUSYD0END	BUSYD0	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF	RXFIFOHF	TXFIFOHE	CPSMACT	DPSMACT	DABORT	DBCKEND	DHOLD	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	SDMMC_ICR	Res.	Res.	Res.	IDMABTCC	IDMATEC	CKSTOPC	VSWEDNDC	ACKTIMEOUTC	ACKFAILC	SDIOITC	BUSYD0ENDC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DABORTC	DBCKENDC	DHOLDC	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
	リセット値				0	0	0	0	0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0
0x3C	SDMMC_MASKR	Res.	Res.	Res.	IDMABTCIE	Res.	CKSTOPIE	VSWENDIE	ACKTIMEOUTIE	ACKFAILIE	SDIOITIE	BUSYD0ENDIE	Res.	Res.	TXFIFOEIE	RXFIFOFIE	Res.	RXFIFOHFIE	TXFIFOHEIE	Res.	Res.	DABORTIE	DBCKENDIE	DHOLDIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
	リセット値				0		0	0	0	0	0	0			0	0		0	0			0	0	0	0	0	0	0	0	0	0	0	0
0x40	SDMMC_ACKTIMER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACKTIME[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44 - 0x4C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x50	SDMMC_IDMACTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABACT	IDMABMODE	IDMAEN
	リセット値																														0	0	0
0x54	SDMMC_IDMABSIZER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABNDT[7:0]								Res.	Res.	Res.	Res.	Res.
	リセット値																				0	0	0	0	0	0	0	0					
0x58	SDMMC_IDMABASE0R	IDMABASE0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	SDMMC_IDMABASE1R	IDMABASE1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60 - 0x7C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	SDMMC_FIFOR	FIF0Data[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84 - x3FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

56 FD コントローラエリアネットワーク（FDCAN）

56.1 概要

コントローラエリアネットワーク（CAN）のサブシステム（*図 722* を参照）は、2 つの CAN モジュール、共有メッセージ RAM メモリ、およびクロック較正ユニットで構成されます。これら 4 つの部分の各ベースアドレスについてはメモリマップを参照してください。

モジュール（FDCAN1 および FDCAN2）は、両方とも ISO 11898-1: 2015（CAN プロトコル仕様バージョン 2.0 パート A、B）および CAN FD プロトコル仕様バージョン 1.0 に準拠しています。

さらに、1 つ目の CAN モジュールである FDCAN1 は、イベント同期タイムトリガ通信、グローバルシステムタイム、クロックドリフト補正を含む ISO 11898-4 に指定されたタイムトリガ CAN（TTCAN）をサポートします。FDCAN1 には、タイムトリガ機能に特化した追加のレジスタがあります。CAN FD オプションは、イベントトリガおよびタイムトリガの CAN 通信でともに使用できます。

10 KB メッセージ RAM メモリには、フィルタ、受信 FIFO、受信バッファ、送信イベント FIFO、送信バッファ（および TTCAN のトリガ）が実装されます。このメッセージ RAM は、FDCAN1 モジュールと FDCAN2 モジュールで共有されます。

共通のクロック較正ユニットはオプションです。これは、FDCAN1 によって受信した CAN メッセージを評価することで、HSI 内部 RC オシレータおよび PLL から FDCAN1 および FDCAN2 両方に較正されたクロックを生成するために使用できます。

CAN サブシステム I/O 信号およびピンの詳細は、それぞれ*表 449* および*表 450* に示します。

表 449. CAN サブシステム I/O 信号

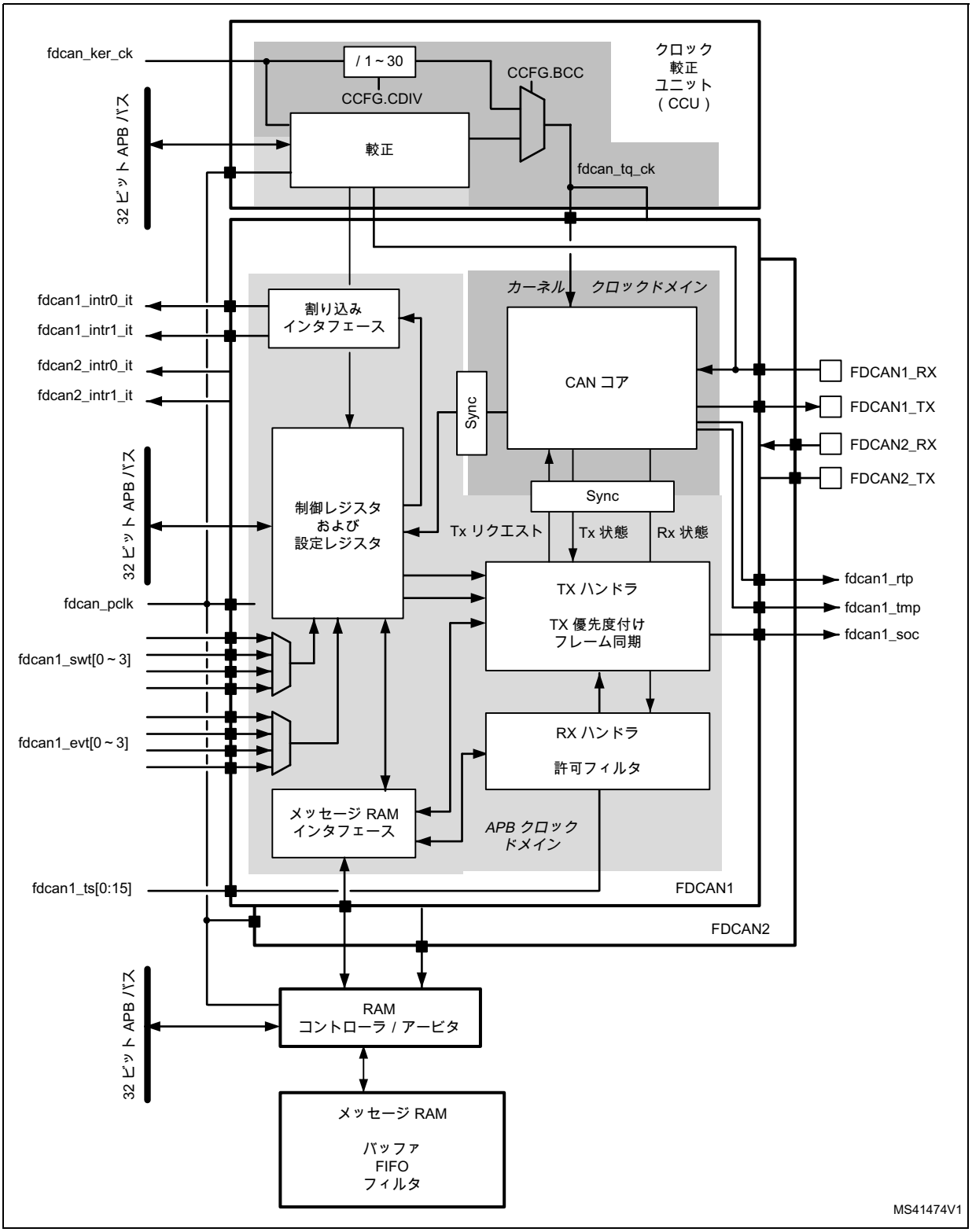
名前	タイプ	説明
fdcan_ker_ck	デジタル入力	CAN サブシステムカーネルクロック入力
fdcan_pclk		CAN サブシステム APB インタフェースクロック入力
fdan1_intr0_it	デジタル出力	FDCAN1 割り込み 0
fdan1_intr1_it		FDCAN1 割り込み 1
fdan2_intr0_it		FDCAN2 割り込み 0
fdan2_intr1_it		FDCAN2 割り込み 1
fdcan1_swt[0:3]	デジタル入力	ストップウォッチトリガ入力
fdcan1_evt[0:3]		イベントトリガ入力
fdcan1_ts[0:15]		外部タイムスタンプベクトル
fdcan1_soc	デジタル出力	サイクルパルスの開始
fdcan1_rtp		レジスタタイムマークパルス
fdcan1_tmp		トリガタイムマークパルス

参考資料

表 450. CAN サブシステム I/O ピン

名前	タイプ	説明
FDCAN1_RX	デジタル入力	FDCAN1 受信ピン
FDCAN1_TX	デジタル出力	FDCAN1 送信ピン
FDCAN2_RX	デジタル入力	FDCAN2 受信ピン
FDCAN2_TX	デジタル出力	FDCAN2 送信ピン

図 722. CAN サブシステム

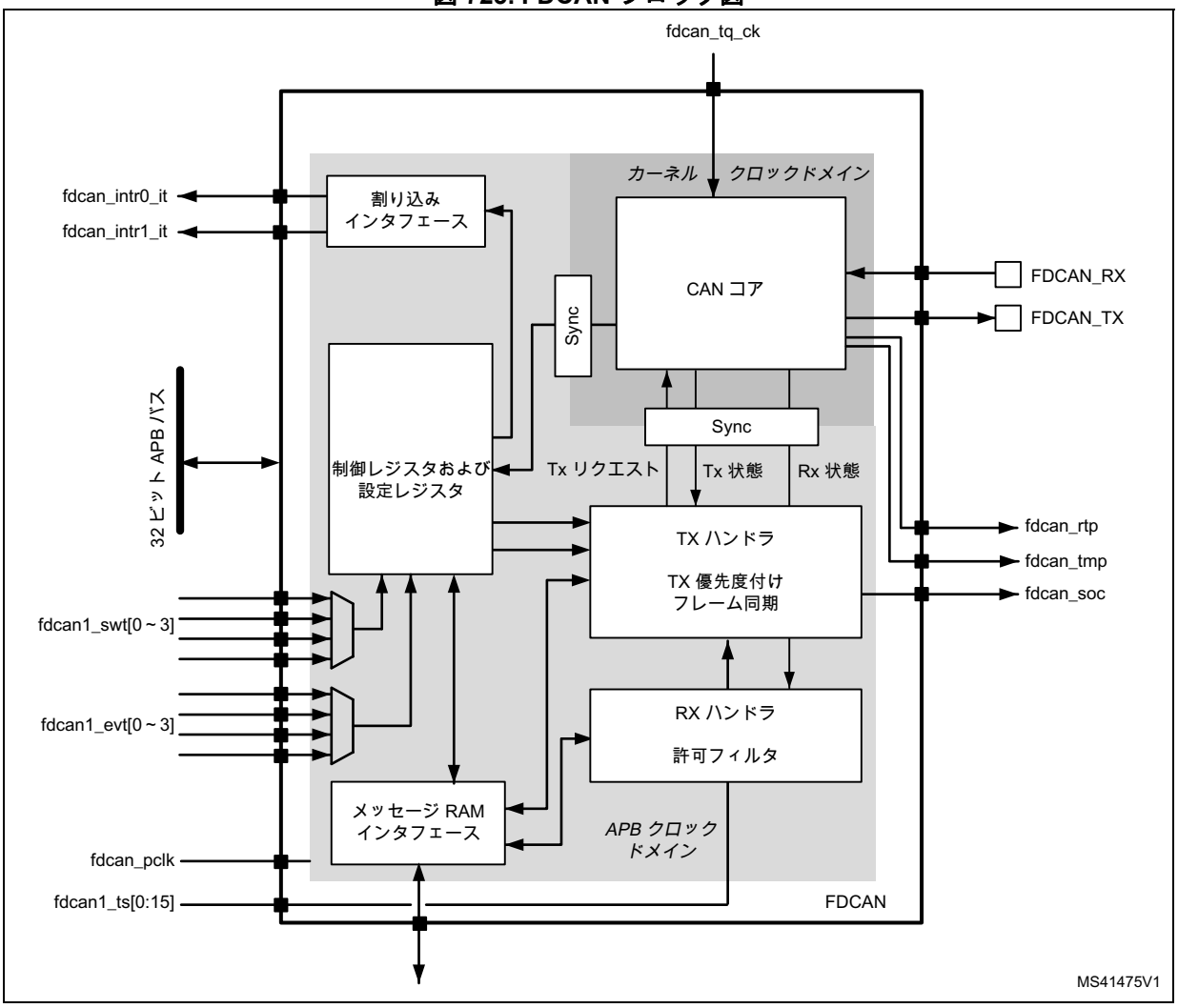


56.2 FDCAN の主な機能

- CAN プロトコルバージョン 2.0 パート A、Bおよび ISO 11898-1: 2015、-4
- 最大 64 データバイトにサポートした CAN FD
- ハードウェアで TTCAN プロトコルレベル 1 およびレベル 2 をサポート（FDCAN1 のみ）
- イベント同期タイムトリガ通信をサポート（FDCAN1 のみ）
- CAN エラーロギング
- AUTOSAR および J1939 サポート
- 改良許可フィルタ
- 設定可能な 2 つの受信 FIFO
- 高優先メッセージの受信で個別の信号
- 最大 64 の専用受信バッファ
- 最大 32 の専用送信バッファ
- 設定可能な送信 FIFO／キュー
- 設定可能な送信イベント FIFO
- FDCAN1 モジュールおよび FDCAN2 モジュールが同じメッセージ RAM を共有
- プログラム可能なループバックテストモード
- マスク可能なモジュール割り込み
- 2 つのクロックドメイン：APB バスインタフェースおよび CAN コアカーネルクロック
- パワーダウンサポート

56.3 FDCAN の機能詳細

図 723. FDCAN ブロック図



2 つの割り込みライン

FDCAN ペリフェラルには、fdcan_intr0_it および fdcan_intr1_it の 2 つの割り込みラインがあります。FDCAN_ILE レジスタで EINT0 および EINT1 ビットをプログラムすることで、割り込みラインを個別に有効化または無効化できます。

CAN コア

CAN コアには、プロトコルコントローラおよび受信／送信シフトレジスタがあります。コアはすべての ISO 11898-1を扱います: 2015 プロトコル機能を処理し、11 ビットおよび 29 ビットの識別子両方をサポートします。

Sync

同期ブロックは、APB クロックドメインと CAN カーネルクロックドメインとの信号を同期します。

Tx ハンドラ

メッセージ RAM から CAN コアへのメッセージ転送を制御します。送信に対して、最大 32 個の Tx バッファを設定できます。Tx バッファは、専用 Tx バッファ、Tx FIFO、Tx キューの一部として、またはそれらを組み合わせて使用できます。Tx イベント FIFO には、対応するメッセージ ID とともに Tx タイムスタンプを格納できます。送信キャンセルもサポートしています。

FDCAN1 では、Tx ハンドラは ISO11898-4 に従ってタイムトリガ通信を制御するフレーム同期エンティティ（FSE）も実装します。そして、CAN バスのリファレンスメッセージと同期させ、サイクルタイムとグローバルタイムを制御し、事前定義されたメッセージスケジュール、システムマトリックスに従って送信を処理します。メッセージ RAM のメッセージとリンクしたシステムマトリックスのタイムマークも処理します。ストップウォッチトリガ、イベントトリガ、タイムマーク割り込みは、同期インタフェースです。

Rx ハンドラ

CAN コアから外部メッセージ RAM への受信したメッセージの転送を制御します。Rx ハンドラは、それぞれ設定可能なサイズの 2 つの受信 FIFO と、許可フィルタを通過したすべてのメッセージのストレージ用に最大 64 個の専用 Rx バッファをサポートします。専用の Rx バッファは受信 FIFO とは対照的に、特定の識別子を持つメッセージのみを格納するために使用します。Rx タイムスタンプが、各メッセージとともに格納されます。11 ビットの ID には最大 128 個のフィルタ、29 ビットの ID には最大 64 個のフィルタを定義できます。

APB インタフェース

FDCAN はAPB バスに接続されています。

メッセージ RAM インタフェース

FDCAN にはRAM コントローラ／アービタ経由で外部の 10 KB メッセージ RAMが接続されています。

56.3.1 動作モード

ソフトウェアの初期化

ソフトウェアの初期化は、FDCAN_CCCR レジスタで INIT ビットをセットするか、ソフトウェアやハードウェアリセット、Bus_Off への移行で開始されます。FDCAN_CCCR レジスタで INIT ビットをセットしている間、CAN バスとのメッセージ転送は停止し、CAN バス出力 FDCAN_TX のステータスはリセッシブ（ハイ）になります。エラー管理ロジック（EML）のカウンタは変更されません。FDCAN_CCCR の INIT ビットのセットで、設定レジスタは変更されません。FDCAN_CCCR の INIT ビットのクリアで、ソフトウェア初期化は終了します。その後、ビットストリームプロセッサ（BSP）は 11 個の連続したリセッシブビット（Bus_Idle）の発生を待つことで、CAN バスでのデータ転送と同期してから、バスアクティビティに移行してメッセージ転送を開始できます。

FDCAN 設定レジスタへのアクセスは、FDCAN_CCCR レジスタの INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみ有効です。

FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがセットされている間のみセット／クリアできます。FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがクリアされているとき、自動的にクリアされます。

FDCAN_CCCR レジスタの CCE ビットがセットされている場合、次のレジスタはリセットされます。

- FDCAN_HPMS - 高優先メッセージステータス
- FDCAN_ RXF0S - Rx FIFO 0 ステータス
- FDCAN_RXF1S - Rx FIFO 1 ステータス
- FDCAN_TXFQS - Tx FIFO／キューステータス
- FDCAN_TXBRP - Tx バッファリクエストペンディング
- FDCAN_TXBTO - Tx バッファ送信発生
- FDCAN_TXBCF - Tx バッファキャンセル終了
- FDCAN_TXEFS - Tx イベント FIFO ステータス
- FDCAN_TTOST - TT 動作ステータス（FDCAN1 のみ）
- FDCAN_TTLGT - TT ローカルおよびグローバルタイム、グローバルタイムのみ、TTLGT.GT がリセット（FDCAN1 のみ）
- FDCAN_TTCTC - TT サイクルタイムおよびカウント（FDCAN1 のみ）
- FDCAN_TTCSM - TT サイクル同期マーク（FDCAN1 のみ）

FDCAN_TOCV レジスタのタイムアウトカウンター値 TOC ビットは、FDCAN_CCCR の CCE ビットがセットされているとき、FDCAN_TOCC レジスタの TOP ビットによって設定された値にプリセットされます。

さらに、Tx ハンドラおよび Rx ハンドラのステートマシンは、FDCAN_CCCR の CCE ビットがセットされている間はアイドル状態で保持されます。

FDCAN_CCCR レジスタの CCE ビットがクリアされている場合のみ、次のレジスタは書き込み可能です。

- TXBAR - Tx バッファ追加リクエスト
- TXBCR - Tx バッファキャンセルリクエスト

FDCAN_CCCR の TEST ビットおよび FDCAN_CCCR の MON ビットは、CCCR の INIT ビットと CCCR レジスタの CCE ビットの両方がセットされているときのみ、ソフトウェアによってセットできます。両ビットは常時リセットできます。FDCAN_CCCR の DAR ビットは、FDCAN_CCCR の INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみセット／クリアできます。

通常動作

ハードウェアリセット後の FDCAN1 のデフォルト動作モードは、タイムトリガなしのイベント駆動 CAN 通信です（TTOCF[OM] = ‘00’）。TT 動作モードに変更される前に、FDCAN_CCCR レジスタの INIT ビットと CCE ビット両方をセットする必要があります。

FDCAN が初期化され、FDCAN_CCCR レジスタの INIT ビットがクリアされると、FDCAN は CAN バスと同期し、通信できるようになります。

許可フィルタを通過した後、メッセージ ID と DLC を含む受信したメッセージは、専用の Rx バッファまたは Rx FIFO 0 や Rx FIFO 1 に格納されます。

送信されるメッセージの場合、専用 Tx バッファおよび／または Tx FIFO や Tx キューを初期化または更新できます。リモートフレーム受信時の自動送信はサポートされません。

CAN FD 動作

FDCAN プロトコルには 2 種類あり、1 つ目は CAN フレームのデータフィールドが 8 バイトより長くなるロングフレームモード（LFM）です。2 つ目は、CAN フレームの制御フィールド、データフィールド、および CRC フィールドがフレームの開始と終了時より高いビットレートで送信される高速フ

レームモード（FFM）です。高速フレームモードは、ロングフレームモードと組み合わせて使用できます。

11 ビットの識別子の CAN フレームで以前に予約済みのビットと、29 ビットの識別子の CAN フレームで一番最初に予約済みのビットは、FDF ビットとしてデコードされるようになります。FDF リセッシブは、CAN FD フレームを示し、FDF ドミナントはクラシック CAN フレームを示します。CAN FD フレームでは、FDF に続く 2 つのビットの res および BRS によって、この CAN FD フレーム内のビットレートを切り替えるかどうかが決まります。CAN FD ビットレートの切り替えは、res ドミナントと BRS リセッシブによって示されます。res リセッシブのコーディングは、プロトコルを今後拡張するために予約済みとなります。M_TTCAN が FDF リセッシブおよび res リセッシブでフレームを受信する場合、ビット PSR.PXE をセットすることでプロトコル例外イベントの信号を送信します。プロトコル例外処理が有効である場合（CCCR.PXHD = ‘0’）、動作状態が次のサンプルポイントで Receiver（PSR.ACT = “10”）からIntegrating（PSR.ACT = “00”）に変わります。プロトコル例外処理が無効である場合（CCCR.PXHD = ‘1’）、FDCAN はリセッシブの res ビットをフォームエラーとして扱い、エラーフレームで応答します。

CAN FD 動作は、CCCR.FDOE をプログラムすることで有効になります。CCCR.FDOE = ‘1’ の場合、CAN FD フレームの送受信は有効です。クラシック CAN フレームの送受信はいつでも可能です。CAN FD フレームまたはクラシック CAN フレームが送信されるかどうかは、それぞれの Tx バッファ要素のビット FDF 経由で設定できます。CCCR.FDOE = ‘0’ では、受信フレームはクラシック CAN フレームとして解釈され、CAN FD フレーム受信時にエラーフレームを送信します。CAN FD 動作が無効である場合、Tx バッファ要素のビット FDF がセットされている場合でも CAN FD フレームは送信されません。CCCR.FDOE および CCCR.BRSE は、CCCR.INIT および CCCR.CCE が両方セットされている間のみ変更できます。

CCCR.FDOE = ‘0’ では、ビット FDF および BRS の設定は無視され、フレームはクラシック CAN フォーマットで送信されます。CCCR.FDOE = ‘1’ かつ CCCR.BRSE = ‘0’ では、Tx バッファ要素のビット FDF だけが評価されます。CCCR.FDOE = ‘1’ かつ CCCR.BRSE = ‘1’ では、ビットレート切り替えでの CAN FD フレームの送信は有効です。ビット FDF および BRS がセットされたすべての Tx バッファ要素は、ビットレート切り替えして CAN FD フォーマットで送信されます。

CAN 動作中のモード変更は、次の条件下でのみ推奨されます。

- CAN FD データフェーズでの失敗率は、CAN FD アービトレーションフェーズより大幅に高くなります。この場合、送信用の CAN FD ビットレート切り替えオプションを無効にします。
- システム起動中、すべてのノードが CAN FD フォーマットで通信できることを検証するまでクラシック CAN メッセージを送信します。これが当てはまる場合、すべてのノードは CAN FD 動作に切り替わります。
- CAN パーシャルネットワーキングのウェイクアップメッセージは、クラシック CAN フォーマットで送信する必要があります。
- すべてのノードが CAN FD に対応していない場合のプログラミングの終了。非 CAN FD ノードは、プログラミングが完了するまでサイレントモードで保持されます。その後、すべてのノードがクラシック CAN 通信に切り替わります。

FDCAN フォーマットでは、DLC のコーディングは標準の CAN フォーマットとは異なります。DLC コード 0～8 は標準 CAN と同じコーディングで、コード 9～15（標準 CAN ではすべて 8 バイトのデータフィールドのコーディング）は*表 451* に従ってコーディングされます。

表 451. FDCAN での DLC コーディング

DLC	9	10	11	12	13	14	15
データバイト数	12	16	20	24	32	48	64

CAN FD 高速フレームでは、BRS（ビットレート切り替え）ビットがリセッシブである場合、このビットの後のフレーム内でビットタイミングが切り替わります。FDCAN アービトレーションフェーズで BRS ビットまでは、ビットタイミングおよびプリスケーラレジスタ BTP によって定義されたとおりに、標準の CAN ビットタイミングが使用されます。次の FDCAN データフェーズでは、高速ビットタイミングおよびプリスケーラレジスタ FBTP によって定義されたとおりに、高速 CAN ビットタイミングが使用されます。CRC デリミタまたはエラー検出時にどちらが先に発生しても、ビットタイミングが高速タイミングから再度切り替わります。

CAN FD データフェーズで設定可能な最大ビットレートは、FDCAN カーネルクロック周波数に依存します。たとえば、20 MHz の FDCAN カーネルクロック周波数で 4 タイムクォンタム（tq）の設定可能な最小ビット時間とした場合、データフェーズのビットレートは 5 Mbit/s となります。

CAN FD ロングフレームおよび CAN FD 高速フレームの両方のデータフレームフォーマットで、ビット ESI（エラーステータスインジケータ）の値は送信開始時のトランスミッタのエラー状態によって決まります。トランスミッタがエラーパッシブである場合、ESI はリセッシブで送信され、その他の場合はドミナントで送信されます。CAN FD リモートフレームでは、ESI ビットはトランスミッタのエラー状態にかかわらず、常にドミナントで送信されます。CAN FD リモートフレームのデータ長コードは 0 で送信されます。

FDCAN Tx バッファが DLC > 8 での FDCAN 送信に設定されている場合、最初の 8 バイトは Tx バッファで設定されたとおりに送信され、データフィールドの残りの部分は 0xCC でパディングされます。FDCAN が DLC > 8 で FDCAN フレームを受信する場合、フレームの最初の 8 バイトは適合する Rx バッファまたは Rx FIFO に格納されます。残りのバイトは破棄されます。

トランシーバ遅延補正

FDCAN 送信のデータフェーズの間、1 つのノードだけが送信して、その他すべてはレシーバになります。バスラインの長さに影響はありません。ピン FDCAN_TX 経由で送信する場合、プロトコルコントローラはピン FDCAN_RX 経由でローカル CAN トランシーバから送信されたデータを受信します。受信されるデータは、CAN トランシーバループ遅延により遅れます。この遅延が TSEG1（サンプルポイント前の時間セグメント）より大きくなる場合、ビットエラーが検出されます。トランシーバ遅延補正がない場合、FDCAN フレームのデータフェーズのビットレートはトランシーバループ遅延によって制限されます。

FDCAN は遅延補正メカニズムを実装して、CAN トランシーバループ遅延を補正することで、特定の CAN トランシーバの遅延にかかわらず、FDCAN データフェーズ中により高いビットレートでの送信を可能にします。

ノード送信のデータフェーズ中にビットエラーをチェックするために、遅延した送信データはセカンダリサンプルポイント SSP で受信したデータと比較されます。ビットエラーが検出される場合、トランスミッタは次のレギュラサンプルポイントでこのビットエラーに反応します。アービトレーションフェーズ中、遅延補正は常に無効です。

トランスミッタ遅延補正は、データビット時間がトランスミッタ遅延より短い場合に設定が可能になります。詳細は、新しい ISO11898-1 に記載されています。ビット DBTP.TDC をセットすることで有効になります。

受信したビットは、SSP で送信したビットと比較されます。SSP の位置は、トランシーバ経由で FDCAN の送信出力ピン FDCAN_TX から受信入力ピン FDCAN_RX までに測定された遅延と、TDCR.TDCO によって設定されたトランスミッタ遅延補正オフセットの合計として定義されます。トランスミッタ遅延補正オフセットは、受信したビット内の SSP の位置を調整するために使用します（例：データフェーズのビット時間の半分）。セカンダリサンプルポイントの位置は、次の mtq（fdcan_tq_ck クロックの 1 周期である最小タイムクォンタム）の整数に切り捨てられます。

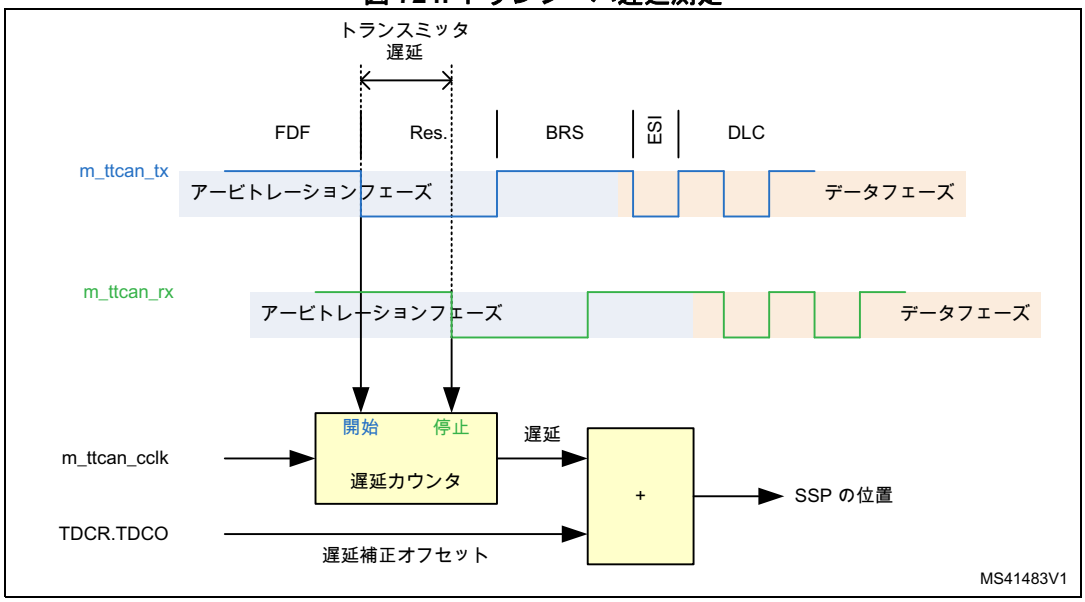
PSR.TDCV は、実際のトランスミッタ遅延補正値を示します。PSR.TDCV は、CCCR.INIT がセットされているとクリアされ、DBTP.TDC がセットされている間 FD フレームの各送信で更新されます。

FDCAN に実装されたトランスミッタ遅延補正では、次の境界条件を考慮する必要があります。

- m_ttcan_tx から m_ttcan_rx までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット TDCR.TDCO の合計は、データフェーズで 6 ビット時間未満でなければなりません。
- m_ttcan_tx から m_ttcan_rx までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット TDCR.TDCO の合計は、127 mtq 以下でなければなりません。この合計が 127 mtq を超える場合、最大値（127 mtq）はトランスミッタ遅延補正に使用されます。
- データフェーズは、CRC デリミタのサンプルポイントで終了し、SSP で受信したビットのチェックを停止します。

トランスミッタ遅延補正が、DBTP.TDC = '1' をプログラムすることで有効になった場合、ビット FDF からビット res の立ち下がりエッジで送信した各 CAN FD フレーム内で測定が開始されます。このエッジがトランスミッタの受信入力ピン FDCAN_TX で見られると測定は停止します。この測定の解析時間は、1 mtq です。

図 724. トランシーバ遅延測定



受信した FDF ビット内のドミナントグリッチによって、受信した res ビットの立ち下がりエッジ前に遅延補正測定を終了する（早い SSP の位置となる）ことを避けるために、TDCR.TDCF をプログラムすることでトランスミッタ遅延補正フィルタウィンドウの使用が有効になります。これで SSP の位置の最小値を定義します。より早い SSP の位置につながる m_ttcan_rx のドミナントエッジは、トランスミッタ遅延測定では無視されます。SSP の位置が少なくとも TDCR.TDCF で、FDCAN_RX がローになると、測定は停止します。

制限動作モード

制限動作モードでは、ノードはデータとリモートフレームを受信でき、有効なフレームに確認応答を付与できますが、データフレーム、リモートフレーム、アクティブエラーフレーム、またはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待って CAN 通信と再同期します。エラーカウンタ（ECR.REC、ECR.TEC）は、エラーロギング（ECR.CEL）がアクティブになっている間は停止します。ソフトウェアは、ビット CCCR.ASM をセットすることで FDCAN を制限動作モードにセットできます。このビットは、CCCR.CCE および CCCR.INIT の両方が '1' にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってクリアできます。

Tx ハンドラがメッセージ RAM から時間内にデータを読み出せなかったとき、自動的に制限動作モードに移行します。制限動作モードを終了するには、ソフトウェアで CCCR.ASM をリセットする必要があります。

制限動作モードは、異なる CAN ビットレートに適応できるアプリケーションで使用できます。この場合、アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。

CCCR.ASM は、クロック較正ユニットでも制御されます。クロック較正プロセスが有効になっているとき、制限動作モードに移行し、CCR.ASM ビットがセットされます。較正が完了すると、CCR.ASM ビットはクリアされます。

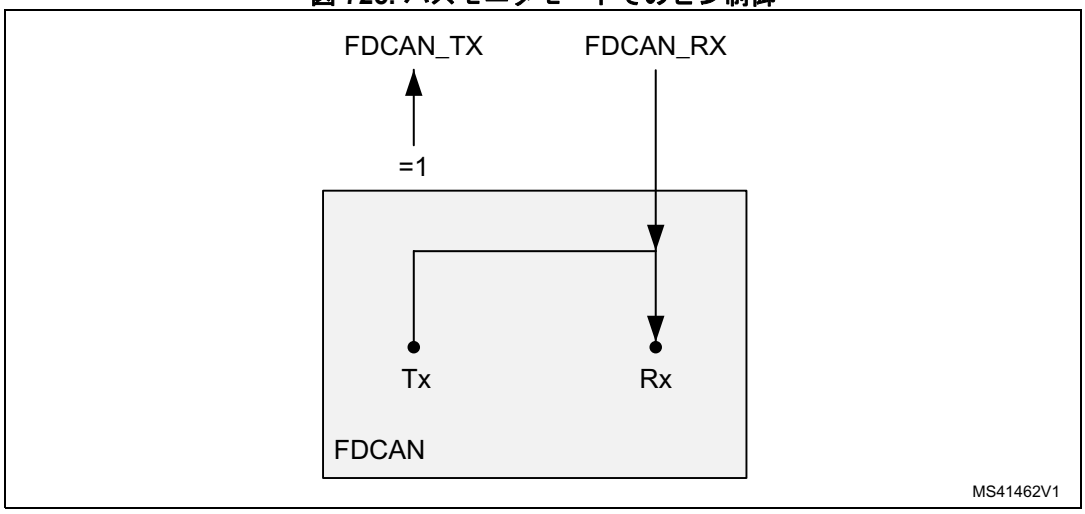
注： ***制限動作モードは、ループバックモード（内部または外部）と組み合わせないでください。***

バスモニタモード

FDCAN は、CCCR.MON ビットをセットするか、エラーレベル S3（TTOST[EL] = ‘11’）に移行した場合にバスモニタモードにセットされます。バスモニタモード（詳細は ISO11898-1、10.12 Bus monitoringを参照してください）では、FDCAN は有効なデータフレームとリモートフレームを受信できますが、送信は開始できません。このモードでは、CAN バスでリセッシブビットのみを送信し、FDCAN がドミナントビット（ACK ビット、オーバロードフラグ、アクティブエラーフラグ）を送信する必要がある場合、CAN バスがリセッシブ状態にとどまることがあっても FDCAN がこのドミナントビットを監視できるように、このビットは内部でルート変更されます。バスモニタモードでは、レジスタ TXBRP はリセット状態のまま保持されます。

バスモニタモードを使用すると、ドミナントビットの送信による影響を受けずに、CAN バス上のトラフィックを解析できます。*図 725* では、バスモニタモードでの FDCAN への FDCAN_TX および FDCAN_RX 信号の接続を示します。

図 725. バスモニタモードでのピン制御



自動再送信（DAR）モード無効

CAN 仕様（ISO11898-1、6.3.3 Recovery Management）に従い、FDCAN はアービトレーションを喪失したフレームや送信中にエラーで中断されたフレームを自動的に再送信する方法を提供します。デフォルトでは自動再送信は有効化されています。

タイムトリガ通信（ISO 11898-1: 2015, 第 9.2 章に準拠）をサポートするために、自動再送信は CCCR[DAR] 経由で無効化できます。

自動再送信（DAR）モード無効時のフレーム送信

DAR モードでは、すべての送信が CAN バスで開始した後で自動的にキャンセルされます。Tx バッファ Tx リクエストペンディングビット TXBRP.TRPx は、転送が成功した後、転送が始まっていない時点でのキャンセル時、ロストアービトレーションによるアボート時、もしくはフレーム転送中にエラーが発生した時にリセットされます。

- 送信成功：
  - 対応する Tx バッファ送信発生 TXBTO[TOx] セット
  - 対応する Tx バッファキャンセル終了 TXBCF[CFx] セットされない
- キャンセル時に送信成功：
  - 対応する Tx バッファ送信発生 TXBTO[TOx] セット
  - 対応する Tx バッファキャンセル終了 TXBCF[CFx] ビットセット
- アービトレーション喪失またはフレーム送信中断：
  - 対応する Tx バッファ送信発生 TXBTO[TOx] セットされない
  - 対応する Tx バッファキャンセル終了 TXBCF[CFx] セット

フレーム送信が成功した場合、かつ Tx イベントのストレージが有効になっている場合、Tx イベント FIFO 要素はイベントタイプ ET = ‘10’（キャンセル時送信）で書き込まれます。

パワーダウン（SLEEPモード）

FDCAN は、CC 制御レジスタ CCCR[CSR] 経由のクロック停止リクエスト入力によって制御されるパワーダウンモードにセットできます。クロック停止リクエストがアクティブである限り、ビット CCCR[CSR] は 1 で読み出されます。

すべてのペンディングされた送信リクエストが完了すると、FDCAN はバスアイドル状態が検出されるまで待ちます。その後、FDCAN は CCCR[INIT] を 1 にセットして、追加の CAN 転送を防ぎます。FDCAN は、CCCR[CSA] を 1 にセットすることでパワーダウンの準備が整ったことを確認します。この状態では、クロックがオフに切り替わる前に、追加のレジスタアクセスが可能です。CCCR[INIT] への書き込みアクセスは効果がありません。これで、モジュールクロック入力はオフに切り替えられます。

パワーダウンモードを終了するには、アプリケーションが CC 制御レジスタフラグ CCCR.CSR をリセットする前にモジュールクロックをオンにしなければなりません。FDCAN は、CCCR[CSA] をリセットすることでこれを確認します。その後、アプリケーションは、ビット CCCR[INIT] をリセットすることで CAN 通信を再開できます。

テストモード

FDCAN テストレジスタへの書き込みアクセス（*2421 ページのセクション 56.4.4* を参照）を有効にするには、ビット CCCR.TEST を 1 にセットして、テストモードおよび機能の設定を有効にしなければなりません。

TEST.TX をプログラムすることで、CAN 送信ピン m_can_tx で 4 つの出力機能が使用できます。デフォルト機能（シリアルデータ出力）に加えて、CAN サンプルポイント信号を駆動して FDCAN ビットタイミングを監視し、一定のドミナントまたはリセッシブ値を駆動できます。ピン m_can_rx での実際の値は、TEST.RX から読み出せます。両方の機能を使用して、CAN バス物理層をチェックできます。

CAN カーネルクロックと APB クロックドメインとの同期メカニズムによって、新しい設定が FDCAN_TX 出力ピンで確認できるようになるまで、TEST.TX への書き込み間で APB クロック周期数周期分の遅延が発生する場合があります。これは、TEST.RX 経由で FDCAN_RX 入力ピンを読み出すときにも適用されます。

注： ***テストモードは、プロダクションテストまたは自己診断機能にのみ使用することを推奨します。FDCAN_TX ピンのソフトウェア制御は、すべての CAN プロトコル機能に干渉します。アプリケーションにテストモードを使用することは推奨されません。***

外部ループバックモード

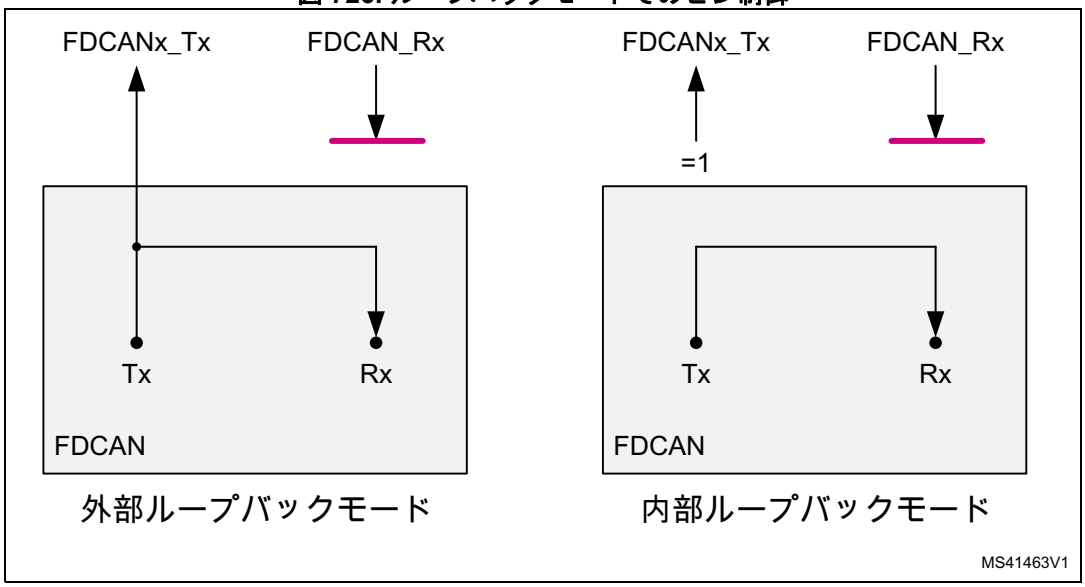
FDCAN を外部ループバックモードに設定するには、TEST.LBCK を 1 にセットします。ループバックモードでは、FDCAN はそれ自体が送信したメッセージを受信メッセージとして扱い、それらを Rx FIFO に格納します（許可フィルタを通過した場合）。*図 726* では、外部ループバックモードでの FDCAN への送受信信号 FDCAN_TX および FDCAN_RX の接続を示します。

このモードは、ハードウェアの自己診断機能のために用意されています。外部通信から独立させるために、ループバックモードでは、FDCAN は ACK エラーを無視します（データ／リモートフレームの ACK スロットではリセッシブビットがサンプリングされます）。このモードでは、FDCAN は送信出力から受信入力へ内部フィードバックを行います。FDCAN は、FDCAN_RX 入力ピンの実際の値を無視します。送信メッセージは FDCAN_TX 送信ピンで監視できます。

内部ループバックモード

TEST.LBCK と CCCR.MON を 1 にプログラムすることによって、内部ループバックモードに移行します。このモードは「ホットセルフテスト」に使用できます。つまり、FDCAN_TX および FDCAN_RX ピンに接続されている実行中の CAN システムに影響を与えずに、FDCAN をテストできます。このモードでは、FDCAN_RX ピンは FDCAN から切断され、FDCAN_TX ピンはリセッシブに保持されます。*図 726* では、内部ループバックモードの場合の FDCAN への FDCAN_TX および FDCAN_RX ピンの接続を示します。

図 726. ループバックモードでのピン制御



アプリケーションウォッチドッグ（FDCAN1 のみ）

アプリケーションウォッチドッグは、レジスタ TTOST を読み出すことで機能します。アプリケーションウォッチドッグが時間内に機能しない場合、ビット TTOST.AWE がセットされ、すべての TTCAN 通信が停止して、FDCAN1 はバスモニタモードにセットされます。

TT アプリケーションウォッチドッグは、アプリケーションウォッチドッグ制限 TTOCF[AWL] を 0x00 にプログラムすることで無効化できます。TTCAN アプリケーションプログラムで TT アプリケーションウォッチドッグを無効にすることは推奨しません。

タイムスタンプの生成

タイムスタンプの生成について、FDCAN は 16 ビットラップアラウンドカウンタを提供します。プリスケーラ TSCC.TCP を設定して、CAN ビット時間の倍数（1～16）でカウンタにクロック供給できます。カウンタは、TSCV[TCV] で読み出せます。レジスタ TSCV への書き込みアクセスによって、カウンタは 0 にリセットされます。タイムスタンプカウンタが0に戻ると、割り込みフラグ IR[TSW] がセットされます。

フレーム受信／送信の開始時、カウンタ値がキャプチャされ、Rx バッファ／Rx FIFO（RXTS[15:0]）または Tx イベント FIFO（TXTS[15:0]）要素のタイムスタンプセクションに格納されます。

ビット TSCC.TSS をプログラムすることで、16 ビットタイムスタンプを使用できます。

タイムアウトカウンタ

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO のタイムアウト条件の信号を送信するために、FDCAN は 16 ビットタイムアウトカウンタを提供します。ダウンカウンタとして動作し、TSCC[TCP] で制御される同じプリスケーラをタイムスタンプカウンタとして使用します。タイムアウトカウンタは、レジスタ TOCC 経由で設定されます。実際のカウンタ値は、TOCV[TOC] から読み出せます。タイムアウトカウンタは CCCR[INIT] = ‘0’ の間だけ開始できます。FDCAN が Bus_Off 状態に移行したときなど CCCR[INIT] = ‘1’ の場合は停止します。

動作モードは TOCC[TOS] で選択されます。連続モードで動作時、CCCR[INIT] がリセットされるとカウンタが開始します。TOCV への書き込みによって、カウンタは TOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。

タイムアウトカウンタが FIFO のうち 1 つで制御されている場合、空の FIFO は TOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。TOCV への書き込みは無効です。

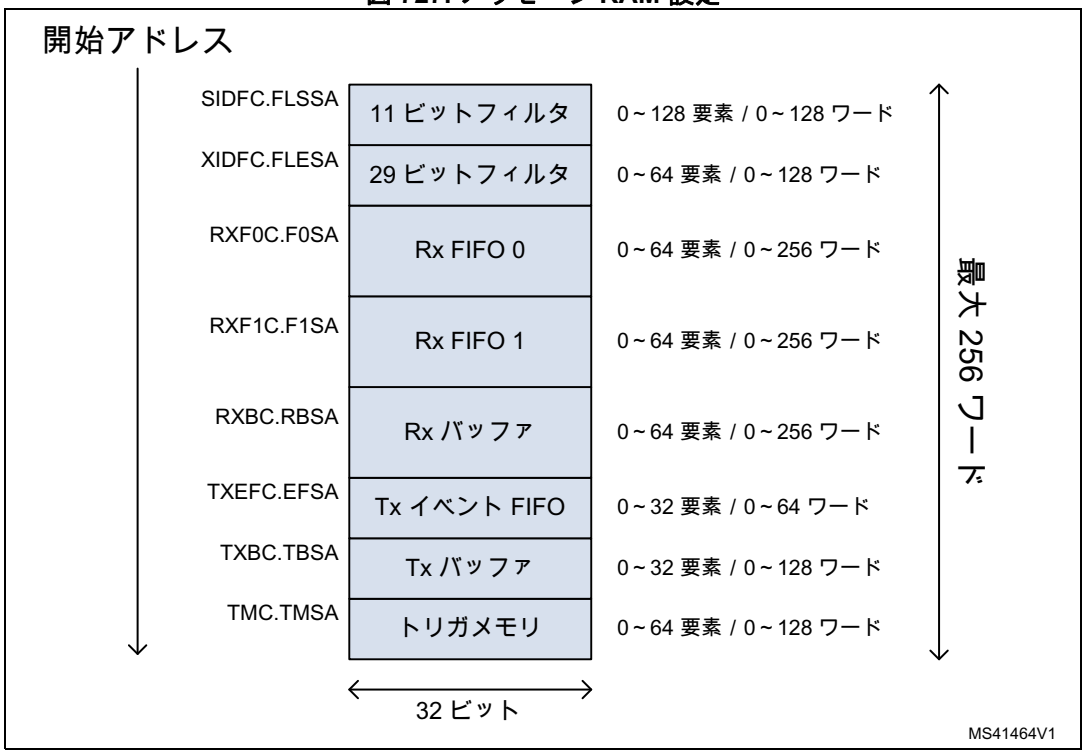
カウンタが 0 に達すると、割り込みフラグ IR[TOO] がセットされます。連続モードでは、カウンタは TOCC[TOP] ですぐに再開されます。

注： ***タイムアウトカウンタのクロック信号は、CAN コアサンプルポイント信号から生成されます。そのため、タイムアウトカウンタがデクリメントされる時間内のポイントは、CAN コアの同期／再同期メカニズムによって異なる場合があります。FDCAN のボーレートスイッチ機能が使用されている場合、タイムアウトカウンタはアービトレーションおよびデータフィールドで別にクロック供給されます。***

56.3.2 メッセージ RAM

メッセージ RAM は 32 ビット幅です。FDCAN モジュールを設定して、メッセージ RAM に最大 2560 ワードを割り当てることができます。*図 727* に示された各セクションを設定する必要はなく、セクションのシーケンスに関して制限もありません。

図 727. メッセージ RAM 設定



FDCAN がメッセージ RAM のアドレスを指定する場合、シングルバイトではなく 32 ビットワードを指定します。設定された開始アドレスは 32 ビットワードアドレスで、たとえば、ビット 15～2 だけが評価され、2 つの最下位ビットは無視されます。

注： ***FDCAN はメッセージ RAM の誤設定をチェックしません。特に、異なるセクションの開始アドレスと、各セクションの要素の数は慎重に設定し、改ざんやデータの損失を回避しなければなりません。***

Rx の処理

Rx ハンドラは、許可フィルタ、Rx バッファまたは 2 つの Rx FIFO のうち 1 つへの受信メッセージの転送、および Rx FIFO の Put と Get のインデックスを制御します。

許可フィルタ

FDCAN は、2 セットの許可フィルタを設定でき、1 つは標準識別子用で、もう 1 つは拡張識別子用です。これらのフィルタは、Rx バッファ、Rx FIFO 0、または Rx FIFO 1 に割り当てられます。許可フィルタでは、フィルタの各リストが要素 #0 から最初に一致する要素まで実行されます。許可フィルタは最初に一致する要素で停止します。次のフィルタ要素はこのメッセージでは評価されません。

主な機能：

- 各フィルタ要素は次のように設定できます。
  - 範囲フィルタ（～から～まで）
  - 1 つまたは 2 つの専用 ID のフィルタ
  - クラシックビットマスクフィルタ
- 各フィルタ要素は、許可フィルタまたは拒否フィルタで設定できます。
- 各フィルタ要素は個別に有効化／無効化できます。
- フィルタは連続でチェックされ、最初の一致フィルタ要素で実行を停止します。

関連する設定レジスタ：

- グローバルフィルタ設定（GFC）
- 標準 ID フィルタ設定（SIDFC）
- 拡張 ID フィルタ設定（XIDFC）
- 拡張 ID ANDマスク（XIDAM）

フィルタ要素の設定（SFEC/EFEC）に応じて、一致すると次の操作のいずれかがトリガされます。

- FIFO 0 または FIFO 1 に受信フレームを格納
- Rx バッファに受信フレームを格納
- Rx バッファに受信フレームを格納し、フィルタイベントピンでパルスを生成
- 受信フレームを拒否
- 高優先メッセージ割り込みフラグ IR[HPM] のセット
- 高優先メッセージ割り込みフラグ IR[HPM] をセットし、FIFO 0 または FIFO 1 に受信フレームを格納
- 高優先メッセージ割り込みフラグ IR.HPM をセットし、FIFO 0 または FIFO 1 に受信フレームを格納

許可フィルタは、完了識別子を受信した後に開始されます。許可フィルタが完了した後、一致する Rx バッファまたは Rx FIFO が見つかった場合、メッセージハンドラが 32 ビットの一部で受信したメッセージデータを、一致する Rx バッファまたは Rx FIFO に書き込み始めます。CAN プロトコルコントローラがエラー条件（CRC エラーなど）を検出した場合、このメッセージは次の影響を与えつつ破棄されます。

- **Rx バッファ**
  一致する Rx バッファの新規データフラグはセットされませんが、Rx バッファ（一部）が受信したデータで上書きされます。エラータイプについては、PSR.LEC および PSR.DLEC を参照してください。
- **Rx FIFO**
  一致する Rx FIFO の Put インデックスは更新されませんが、関連する FIFO 要素（一部）が受信したデータで上書きされます。エラータイプについては、PSR.LEC および PSR.DLEC を参照してください。一致する Rx FIFO が上書きモードで動作している場合、*Rx FIFO 上書きモード*に記載された境界条件を考慮する必要があります。

注： ***許可したメッセージが 2 つの Rx FIFO のうち 1 つまたは Rx バッファに書き込まれると、変更されていない受信 ID は、使用されるフィルタにかかわらず格納されます。許可フィルタプロセスの結果は、設定されたフィルタ要素のシーケンスに強く依存します。***

範囲フィルタ

フィルタが SF1ID/SF2ID および EF1ID/EF2ID で定義された範囲のメッセージ ID を持つすべての受信フレームに一致します。

範囲フィルタを拡張フレームと使用する場合、2 つのことが可能です。

- EFT = ‘00’：範囲フィルタが適用される前に、受信フレームのメッセージ ID は拡張 ID AND マスク（XIDAM）と AND でつながれます。
- EFT = ‘11’：拡張 ID AND マスク（XIDAM）は、範囲フィルタに使用されません。

専用 ID のフィルタ

フィルタ要素を設定して、1 つまたは 2 つの固有 ID のフィルタでフィルタできます。1 つの固有メッセージ ID でフィルタするには、フィルタを SF1ID=SF2ID および EF1ID=EF2ID で設定する必要があります。

クラシックビットマスクフィルタ

クラシックビットマスクフィルタは、受信したメッセージ ID のシングルビットをマスクして、メッセージ ID のグループをフィルタすることを目的としています。クラシックビットマスクフィルタで、SF1ID/EF1ID はメッセージ ID フィルタとして使用され、SF2ID/EF2ID はフィルタマスクとして使用されます。

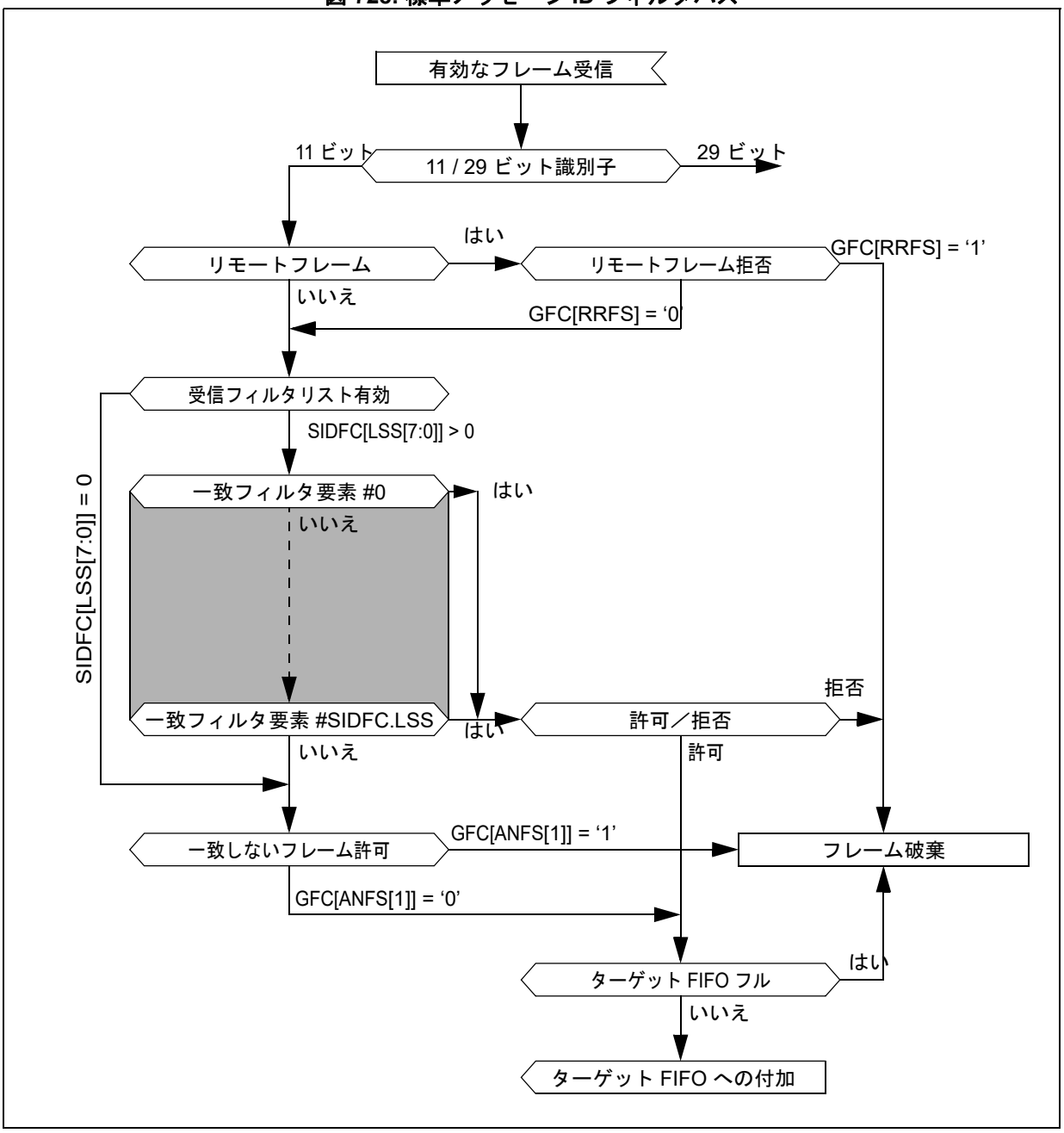
フィルタマスクでの 0 ビットは、設定された ID フィルタの対応するビット位置をマスクします。たとえば、そのビット位置で受信したメッセージ ID は許可フィルタの影響を受けません。対応するマスクビットが 1 の場合、受信したメッセージ ID のそれらのビットだけが許可フィルタの影響を受けます。

すべてのマスクビットが 1 の場合、受信したメッセージ ID とメッセージ ID フィルタが同一である場合のみ一致が発生します。すべてのマスクビットが 0 の場合、すべてのメッセージ ID が一致します。

標準メッセージ ID フィルタ

図 728 では、標準メッセージ ID（11 ビット識別子）フィルタのフローを示します。標準メッセージ ID フィルタ要素は、*2414 ページのセクション 56.3.19* を参照してください。

図 728. 標準メッセージ ID フィルタパス

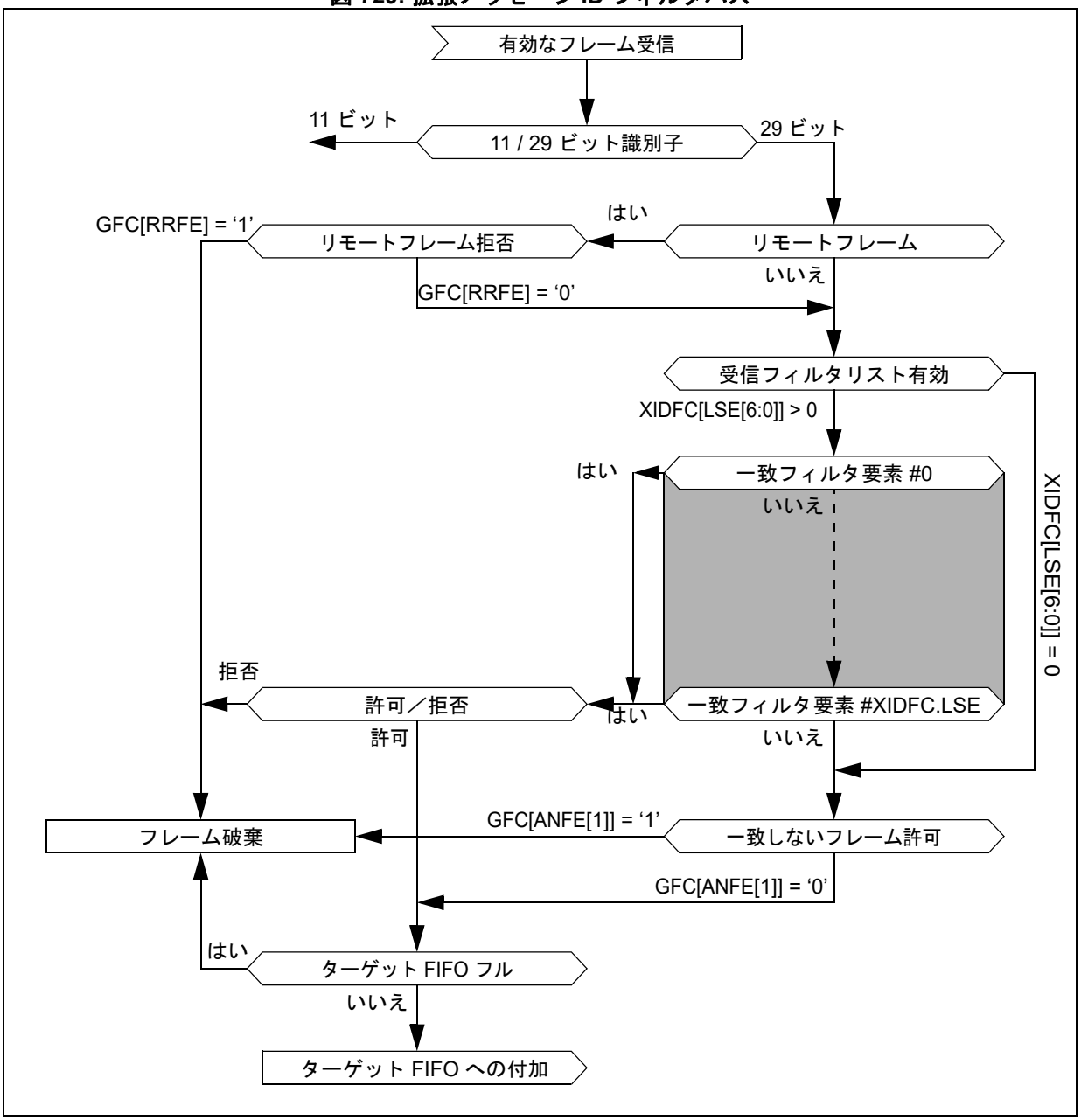


グローバルフィルタ設定（GFC）および標準 ID フィルタ設定（SIDFC）メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）が設定されたフィルタ要素のリストと比較されます。

拡張メッセージ ID フィルタ

図 729 では、拡張メッセージ ID（29 ビット識別子）フィルタのフローを示します。拡張メッセージ ID フィルタ要素は、*2415 ページのセクション 56.3.20* を参照してください。

図 729. 拡張メッセージ ID フィルタパス



グローバルフィルタ設定 GFC および拡張 ID フィルタ設定 XIDFC メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）が設定されたフィルタ要素のリストと比較されます。

フィルタリストが実行される前に、拡張 ID AND マスク（XIDAM）は受信識別子と AND でつながれます。

Rx FIFO

Rx FIFO 0 および Rx FIFO 1 を設定すると、それぞれ最大 64 個の要素を保持できます。2 つの Rx FIFO は、レジスタ RXF0C および RXF1C で設定します。

許可フィルタを通過した受信メッセージは、一致フィルタ要素で設定されたとおりに Rx FIFO に転送されます。Rx FIFO 0 および Rx FIFO 1 で使用できるフィルタメカニズムの詳細については、*許可フィルタ*を参照してください。Rx バッファおよび FIFO 要素は、*セクション 56.3.16 : FDCAN の Rx バッファおよび FIFO 要素* を参照してください。

Rx FIFO フル条件が IR[RFnF] によって信号送信されると、最低 1 つのメッセージが読み出され、Rx FIFO Get インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、割り込みフラグ IR[RFnL] がセットされます。

Rx FIFO オーバーフローを回避するために、Rx FIFO ウォーターマークを使用できます。Rx FIFO フィルレベルが RXFnC[FnWM] で設定された Rx FIFO ウォーターマークに達すると、割り込みフラグ IR[RFnW] がセットされます。

Rx FIFO から読み出すときに、Rx FIFO Get インデックス RXFnS[FnGI] + FIFO 要素サイズを対応する Rx FIFO 開始アドレス RXFnC[FnSA] に追加する必要があります。

Rx FIFO ブロックモード

Rx FIFO ブロックモードは、RXFnC.FnOM = ‘0’ で設定されます。これは、Rx FIFO のデフォルトの動作モードです。

Rx FIFO フル条件（RXFnS.FnPI = RXFnS.FnGI）に達すると、最低 1 つのメッセージが読み出され、Rx FIFO Get インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。Rx FIFO フル条件は、RXFnS.FnF = ‘1’ で信号送信されます。さらに、割り込みフラグ IR.RFnF がセットされます。

対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、メッセージ喪失条件が RXFnS.RFnL = ‘1’ で信号送信されます。さらに、割り込みフラグ IR.RFnL がセットされます。

Rx FIFO 上書きモード

Rx FIFO 上書きモードは、RXFnC.FnOM = ‘1’ で設定されます。

Rx FIFO フル条件（RXFnS.FnPI = RXFnS.FnGI）が RXFnS.FnF = ‘1’ によって信号送信されると、FIFO に許可された次のメッセージは一番古い FIFO メッセージを上書きします。Put および Get インデックスは両方 1 インクリメントされます。

Rx FIFO が上書きモードで動作し、Rx FIFO フル条件の信号を送信したとき、Rx FIFO 要素の読み出しは最低 Get インデックス + 1 で開始します。その理由は、CPU がメッセージ RAM（Get インデックス）から読み出しつつ、受信メッセージがメッセージ RAM（Put インデックス）に書き込まれる状況になりえるためです。この場合、一貫していないデータがそれぞれの Rx FIFO 要素から読み出される場合があります。Rx FIFO から読み出し時に Get インデックスにオフセットを加えることで、この問題を回避しています。このオフセットは、CPU が Rx FIFO にアクセスする速度に依存します。*図 731 : 混合設定専用 Tx バッファ／Tx キューの例* では、Rx FIFO 読み出し時の Get インデックスに関する 2 のオフセットを示しています。この場合、要素 1 と 2 に格納された 2 つのメッセージが失われます。

Rx FIFO から読み出した後、最後の要素読み出しの数を Rx FIFO 確認応答インデックス RXFnA.FnA に書き込む必要があります。これにより、Get インデックスがその要素数にインクリメントされます。Put インデックスがこの Rx FIFO 要素にインクリメントされなかった場合、Rx FIFO フル条件はリセットされます（RXFnS.FnF = ‘0’）。

専用 Rx バッファ

FDCAN は、最大 64 の専用 Rx バッファをサポートします。専用 Rx バッファセクションの開始アドレスは RXBC.RBSA 経由で設定されます。

Rx バッファごとに、SFEC/EFEC='111'および SFID2/EFID2[10:9]='00' の標準または拡張メッセージ ID フィルタ要素を設定する必要があります（*セクション 56.3.19 : FDCAN 標準メッセージ ID フィルタ要素*および*セクション 56.3.20 : FDCAN 拡張メッセージ ID フィルタ要素*を参照してください）。

受信メッセージがフィルタ要素によって許可された後、メッセージはフィルタ要素によって参照されるメッセージ RAM の Rx バッファに格納されます。フォーマットは、Rx FIFO 要素と同じです。さらに、割り込みレジスタでフラグ IR.DRX（専用 Rx バッファに格納されたメッセージ）がセットされます。

表 452. Rx バッファのフィルタ設定の例

フィルタ要素	SFID1[10:0] EFID1[28:0]	SFID2[10:9] EFID2[10:9]	SFID2[5:0] EFID2[5:0]
0	ID メッセージ 1	00	00 0000
1	ID メッセージ 2	00	00 0001
2	ID メッセージ 3	00	00 0010

一致している受信メッセージの最後のワードがメッセージ RAM に書き込まれた後、レジスタ NDAT1、2 でそれぞれの新規データフラグがセットされます。新規データフラグがセットされている限り、受信した一致しているフレームで更新されないようにそれぞれの Rx バッファがロックされます。それぞれのビット位置に 1 を書き込むことによって新規データフラグをホストでリセットする必要があります。

Rx バッファの新規データフラグがセットされている間、この固有 Rx バッファを参照しているメッセージ ID フィルタ要素は一致せず、許可フィルタを続行します。次のメッセージ ID フィルタ要素は、フィルタ設定によって受信メッセージの別の Rx バッファや Rx FIFO への格納、またはメッセージの拒否が発生する場合があります。

Rx バッファの処理

- 割り込みフラグ IR.DRX のリセット
- 新規データレジスタの読み出し
- メッセージ RAM からのメッセージの読み出し
- 処理したメッセージの新規データフラグのリセット

デバッグメッセージのフィルタ

デバッグメッセージのフィルタは、3 つのデバッグメッセージのそれぞれに 1 つの標準／拡張メッセージ ID フィルタ要素を設定することで実行されます。フィルタ要素でデバッグメッセージのフィルタを有効にするには、SFEC/EFEC を '111' にプログラムする必要があります。この場合では、フィールド SFID1/SFID2 および EFID1/EFID2 には異なる意味があります。SFID2/EFID2[10:9] はデバッグメッセージ処理ステートマシンを制御し、SFID2/EFID2[5:0] は受信デバッグメッセージの格納場所を制御します。

デバッグメッセージが格納されると、それぞれ新規データフラグや IR.DRX はセットされません。デバッグメッセージの受信は、RXF1S.DMS で監視できます。

表 453. デバッグメッセージのフィルタ設定の例

フィルタ要素	SFID1[10:0] EFID1[28:0]	SFID2[10:9] EFID2[10:9]	SFID2[5:0] EFID2[5:0]
0	ID デバッグメッセージ A	01	11 1101
1	ID デバッグメッセージ B	10	11 1110
2	ID デバッグメッセージ C	11	11 1111

Tx の処理

Tx ハンドラは、専用 Tx バッファ、Tx FIFO、Tx キューの送信リクエストを処理します。CAN コア、Put および Get インデックス、Tx イベント FIFO への送信メッセージの転送を制御します。最大 32 の Tx バッファをメッセージ送信にセットアップできます（*専用 Tx バッファ*を参照）。要素サイズ（RXESC）の設定に応じて、2～16 の 32 ビットワード（Rn = 3～17）が CAN メッセージデータフィールドの格納に使用されます。

表 454. フレーム送信の可能な設定

CCCR		Tx バッファ要素		フレーム送信
BRSE	FDOE	FDF	BRS	
無視	0	無視	無視	クラシック CAN
0	1	0	無視	クラシック CAN
0	1	1	無視	ビットレート切り替えなしの FD
1	1	0	無視	クラシック CAN
1	1	1	0	ビットレート切り替えなしの FD
1	1	1	1	ビットレート切り替えありの FD

注： ***AUTOSAR は最低 3 つの Tx キューバッファと送信キャンセルのサポートが必須です。***

Tx ハンドラは、Tx バッファリクエストペンディングレジスタ TXBRP が更新されたときや、送信が開始されたときに、Tx スキャンを開始して、最優先のペンディング中の Tx リクエストをチェックします。

送信一時停止

送信一時停止機能は、CAN メッセージ識別子が（完全に）固有の値に指定され、容易に変更できない CAN システムで使用することを目的とします。これらのメッセージ識別子は、他の定義されたメッセージより高い CAN アービトレーション優先度である場合がありますが、個別のアプリケーションでは関連するアービトレーション優先度が逆になります。これにより、他のメッセージが低い CAN アービトレーション優先度であるため、ある ECU が一気に CAN メッセージを送信して別の ECU CAN メッセージを遅延させる場合があります。

たとえば、CAN ECU-1 が機能を有効にして、アプリケーションソフトウェアによって 4 件のメッセージを送信するようリクエストされた場合、最初のメッセージ送信が成功した後、次のリクエストされたメッセージの開始が許可される前に 2 CAN ビット時間のバスアイドルを待ちます。ペンディング中のメッセージを保持する他の ECU がある場合、それらのメッセージはアイドル時間で開始され、ECU-1 の次のメッセージとアービトレーションする必要はありません。メッセージを受信した後、受信メッセージが CAN バスを開放すると、ECU-1 は次の送信開始が許可されます。

この機能は、CCCR レジスタの TXP ビットで制御されます。このビットがセットされている場合、FDCAN がメッセージの送信に成功するたびに、次の送信開始前の 2 CAN ビット時間、一時停止します。これにより、ネットワーク内の他の CAN ノードのメッセージが低い優先度識別子を持つものであってもメッセージを送信できます。デフォルトでは無効です（CCCR.TXP = ‘0’）。

この機能は、1 つのノードからのバースト送信を緩和し、アプリケーションプログラムが誤って大量の送信をリクエストする「バブリングイディオット」シナリオから保護します。

専用 Tx バッファ

専用 Tx バッファは、CPU に完全制御されたメッセージ送信を対象とします。各専用 Tx バッファは固有メッセージ ID で設定されます。複数の Tx バッファが同じメッセージ ID で設定される場合、一番小さいバッファ番号の Tx バッファが先に送信されます。

データセクションが更新された場合、送信は TXBAR[ARn] 経由の追加リクエストによってリクエストされます。リクエストされたメッセージは、任意の Tx FIFO または Tx キューからのメッセージとは内部で、CAN バス上のメッセージとは外部でアービトレーションし、それらのメッセージ ID に従って送信されます。

専用 Tx バッファはメッセージ RAM で 4 つの 32 ビットワードを割り当てます。そのため、メッセージ RAM での専用 Tx バッファの開始アドレスは、送信バッファインデックス（0～31）を Tx バッファ開始アドレス TXBC[TBSA] に 4 回追加することで計算されます。

表 455. Tx バッファ／FIFO - キュー要素サイズ

TXESC、TBDS[2;0]	データフィールド（バイト）	要素サイズ（RAM ワード）
000	8	4
001	12	5
010	16	6
011	20	7
100	24	8
101	32	10
110	48	14
111	64	18

Tx FIFO

Tx FIFO 動作は、TXBC[TFQM] を ‘0’ にプログラムすることによって設定されます。Tx FIFO に格納されたメッセージは、Get インデックス TXFQS[TFGI] によって参照されるメッセージから送信されます。各送信後、Get インデックスは Tx FIFO がエンプティになるまで周期的にインクリメントされます。Tx FIFO は、メッセージが Tx FIFO に書き込まれた順番で、異なる Tx バッファからの同じメッセージ ID を持つメッセージを送信できます。FDCAN は、Tx FIFO フリーレベル TXFQS[TFFL] を Get および Put インデックス間の差として計算します。これは、使用可能な（空き）Tx FIFO 要素の数を示します。

新しい送信メッセージは、Put インデックス TXFQS[TFQPI] によって参照される Tx バッファから Tx FIFO に書き込まれる必要があります。追加リクエストは、次の空き Tx FIFO 要素に Put インデックスをインクリメントします。Put インデックスが Get インデックスに達すると、Tx FIFO フル（TXFQS[TFQF]= ‘1’）の信号が送信されます。この場合、次のメッセージが送信され、Get インデックスがインクリメントされるまで、Tx FIFO に追加のメッセージは書き込まれません。

1 つのメッセージが Tx FIFO に追加される場合、Tx FIFO Put インデックスによって参照される Tx バッファに関連した TXBAR ビットに ‘1’ を書き込むことで送信がリクエストされます。

複数（n）件のメッセージが Tx FIFO に追加される場合、Put インデックスから始まる n 個の連続した Tx バッファに書き込まれます。送信は TXBAR 経由でリクエストされます。Put インデックスは周期的に n ずつインクリメントされます。リクエストされる Tx バッファの数は、Tx FIFO フリーレベルで示された空きの Tx バッファの数を超えてはなりません。

Get インデックスによって参照される Tx バッファに対する送信リクエストがキャンセルされた場合、Get インデックスは送信リクエストをペンディングして次の Tx バッファにインクリメントされ、Tx FIFO フリーレベルが再計算されます。送信キャンセルが他のあらゆる Tx バッファに適用される場合、Get インデックスと FIFO フリーレベルに変化はありません。

Tx FIFO 要素はメッセージ RAM で 4 つの 32 ビットワードを割り当てます。そのため、次に使用可能な（空き）Tx FIFO バッファの開始アドレスは、Put インデックス TXFQS[TFQPI]（0～31）を Tx バッファ開始アドレス TXBC[TBSA] に 4 回追加することで計算されます。

Tx キュー

Tx キュー動作は、TXBC[TFQM] を ‘1’ にプログラムすることによって設定されます。Tx キューに格納されたメッセージは、一番小さいメッセージ ID（最優先）のメッセージから送信されます。複数のキューバッファが同じメッセージ ID で設定される場合、一番小さいバッファ数のキューバッファが先に送信されます。

新しいメッセージは、Put インデックス TXFQS[TFQPI] によって参照される Tx バッファに書き込まれる必要があります。追加リクエストは、次の空き Tx バッファに Put インデックスをインクリメントします。Tx キューがフル（TXFQS[TFQF]= ‘1’）の場合、Put インデックスは有効ではなく、リクエストされるメッセージのうち最低 1 件が送信されるか、ペンディング中の送信リクエストがキャンセルされるまで、追加のメッセージは Tx キューに書き込まれません。

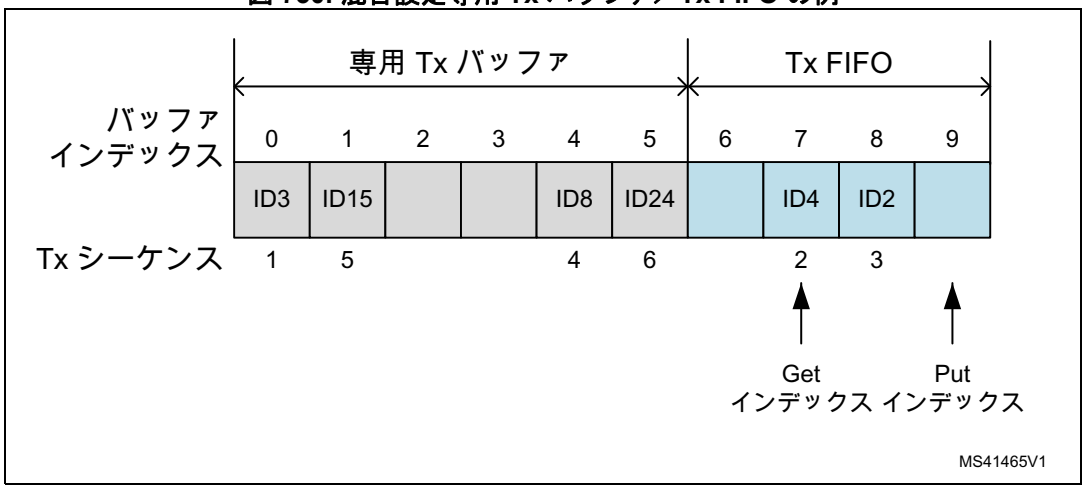
アプリケーションは Put インデックスの代わりにレジスタ TXBRP を使用し、送信リクエストをペンディングせずに Tx バッファにメッセージを配置する場合があります。

Tx キューバッファはメッセージ RAM で 4 つの 32 ビットワードを割り当てます。そのため、次に使用可能な（空き）Tx キューバッファの開始アドレスは、Tx キュー Put インデックス TXFQS[TFQPI]（0～31）を Tx バッファ開始アドレス TXBC[TBSA] に 4 回追加することで計算されます。

混合専用 Tx バッファ／Tx FIFO

このケースでは、メッセージ RAM の Tx バッファセクションが専用 Tx バッファと Tx FIFO のセットに分けられます。専用 Tx バッファの数は、TXBC[NDTB] によって設定されます。Tx FIFO に割り当てられる Tx バッファの数は、TXBC[TFQS] によって設定されます。TXBC[TFQS] が 0 にプログラムされる場合、専用 Tx バッファのみが使用されます。

図 730. 混合設定専用 Tx バッファ／Tx FIFO の例



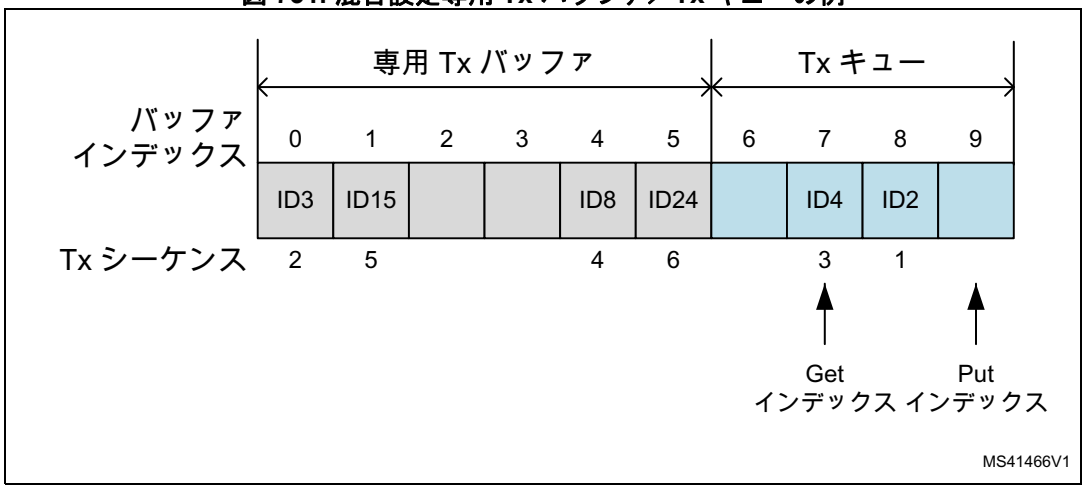
Tx 優先度付け：

- 専用 Tx バッファと一番古いペンディング中の Tx FIFO バッファ（TXFS[TFGI] によって参照）をスキャンします。
- 一番小さいメッセージ ID のバッファが最優先となり、次に送信されます。

混合専用 Tx バッファ／Tx キュー

このケースでは、メッセージ RAM の Tx バッファセクションが専用 Tx バッファと Tx キューのセットに分けられます。専用 Tx バッファの数は、TXBC[NDTB] によって設定されます。Tx キューバッファの数は、TXBC[TFQS] によって設定されます。TXBC[TFQS] が 0 にプログラムされる場合、専用 Tx バッファのみが使用されます。

図 731. 混合設定専用 Tx バッファ／Tx キューの例



Tx 優先設定：

- 有効化された送信リクエストのある Tx バッファすべてをスキャンします。
- 一番小さいメッセージ ID の Tx バッファが最優先となり、次に送信されます。

送信キャンセル

FDCAN は送信キャンセルをサポートします。専用 Tx バッファまたは Tx キューバッファからリクエストされた送信をキャンセルするには、ホストはレジスタ TXBCR の対応するビット位置（= Tx バッファの数）に ‘1’ を書き込む必要があります。送信キャンセルは Tx FIFO 動作を対象としません。

成功したキャンセルは、レジスタ TXBCF の対応するビットを ‘1’ にセットすることで信号送信されます。

Tx バッファからの送信がすでに進行中で、送信キャンセルがリクエストされた場合、対応する TXBRP ビットは送信が進行している限りセットされたままとなります。送信に成功した場合、対応する TXBTO および TXBCF ビットがセットされます。送信に成功しなかった場合、繰り返されず、対応する TXBCF ビットのみセットされます。

注： ***ペンディング中の送信が開始される直前にキャンセルされた場合、別のメッセージがこのノードにペンディングされている場合でも送信が開始されない短いタイムウィンドウが続きます。これにより、別のノードはそのノードの 2 番目の優先度のメッセージより低い優先度のメッセージを送信できます。***

Tx イベントの処理

Tx イベントの処理をサポートするために、FDCAN は Tx イベント FIFO を実装しました。FDCAN が CAN バスでメッセージを送信すると、メッセージ ID およびタイムスタンプが Tx イベント FIFO 要素に格納されます。Tx イベントを Tx イベント FIFO 要素にリンクするために、送信された Tx バッファのメッセージマーカーが Tx イベント FIFO 要素にコピーされます。

Tx イベント FIFO は、最大 32 個の要素に設定できます。Tx イベント FIFO 要素は、*Tx FIFO* を参照してください。要素サイズ（TXESC）の設定に応じて、2～16 の 32 ビットワード（Tn = 3～17）が CAN メッセージデータフィールドの格納に使用されます。

Tx イベント FIFO の目的は、Tx バッファが送信すべきメッセージのみを保持しつつ、送信ステータスは Tx イベント FIFO に別に格納されるなど、送信ステータス情報の処理を送信メッセージ処理から切り離すことです。これには、特に動的に管理される送信キューを操作する際に、Tx バッファが送信成功直後に新しいメッセージに使用できるという利点があります。Tx バッファを上書きする前に、その Tx バッファからの送信ステータス情報を保存する必要はありません。

Tx イベント FIFO フル条件が IR[TEFF] によって信号送信されると、最低 1 つの要素が読み出され、Tx イベント FIFO Get インデックスがインクリメントされるまで、Tx イベント FIFO に追加のメッセージは書き込まれません。Tx イベント FIFO がフルになっているときに Tx イベントが発生した場合、このイベントは破棄され、割り込みフラグ IR[TEFL] がセットされます。

Tx イベント FIFO オーバーフローを回避するために、Tx イベント FIFO ウォーターマークを使用できます。Tx イベント FIFO フィルレベルが TXEFC[EFWM] で設定された Tx イベント FIFO ウォーターマークに達すると、割り込みフラグ IR[TEFW] がセットされます。

Tx イベント FIFO から読み出すときに、2倍のTx イベント FIFO Get インデックス TXEFS[EFGI] を Tx イベント FIFO 開始アドレス TXEFC[EFSA] に追加する必要があります。

56.3.3 FIFO 確認応答の処理

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO の Get インデックスは、対応する FIFO 確認応答インデックスへの書き込みによって制御されます。*セクション 56.4.28 : CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）*、*セクション 56.4.32 : FDCAN Rx FIFO 1 確認応答レジスタ (FDCAN_RXF1A)*、*セクション 56.4.44 : FDCAN Tx イベント FIFO 設定レジスタ (FDCAN_TXEFC)* を参照してください。FIFO 確認応答インデックスへの書き込みは FIFO Get インデックスを FIFO 確認応答インデックス +1 にセットし、FIFO フィルレベルを更新します。これには、2 つの使用例があります。

1. 要素が 1 つだけ FIFO（Get インデックスによって指定されているもの）から読み出された場合、この Get インデックス値は FIFO 確認応答インデックスに書き込まれます。
2. 一連の要素が FIFO から読み出された場合、読み出しシーケンスの最後に 1 回だけ FIFO 確認応答インデックス（値：最後の要素読み出しのインデックス）を書き込んで、FIFO Get インデックスを更新するだけで十分です。

CPU が FDCAN メッセージ RAM に自由にアクセスできるため、任意の（Get インデックスが考慮されない）順番で FIFO 要素を読み出す際に特別な注意が必要です。これは、2 つの Rx FIFO のうち 1 つから高優先メッセージを読み出すときに役立つ場合があります。この場合、Get インデックスが誤った位置にセットされ、FIFO フィルレベルも変わるため、FIFO 確認応答インデックスへの書き込みは推奨しません。この場合、古いほうの FIFO 要素の一部は失われます。

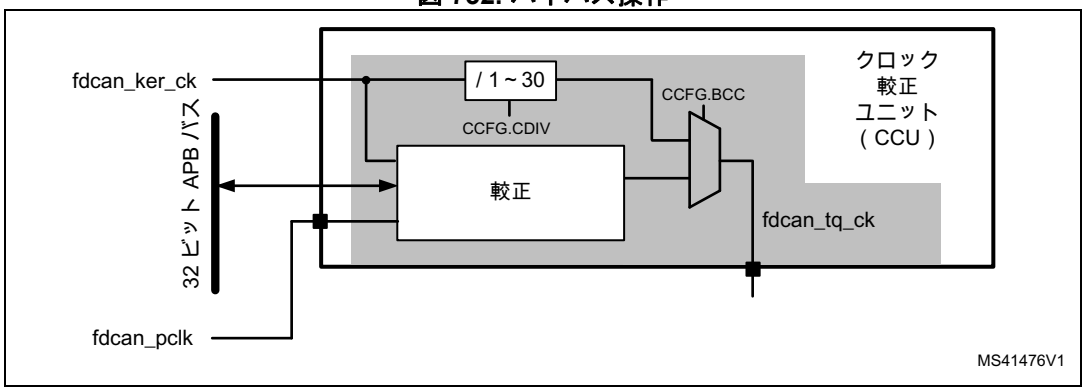
注： ***アプリケーションは、有効な値が FIFO 確認応答インデックスに書き込まれていることを確認する必要があります。FDCAN は誤った値をチェックしません。***

56.3.4 CAN のクロック較正

デバイスリセット後、クロック較正ユニット（CCU）は FDCAN1 および FDCAN2 に有効なクロック信号を供給しません。CCU を CCFG レジスタ経由で初期化する必要があります。CCFG レジスタに書き込めるのは、FDCAN1 で CCCR.CCE および CCCR.INIT の両方のビットがセットされているときだけです。つまり、FDCAN1 および／または FDCAN2 モジュールが動作できるようにするには、CCU および FDCAN1 の初期化を完了する必要があります。

CCFG.BCC = ‘1’ の場合、クロック較正はバイパスされます（*図 732* を参照）。

図 732. バイパス操作



動作条件

CAN ユニットのクロック較正は、次の条件で動作するように設計されています。

- CAN カーネルクロック周波数 fdcan_ker_ck 80～500 MHz
- FDCAN ビットレート 125 kbit/s～1 Mbit/s

FDCAN ユニットのクロック較正は、0.5～25 MHz の範囲で較正したタイムクォンタムクロック fdcan_tq_ck を生成します。

注： ***FDCAN では、常に CAN タイムクォンタムクロックが APB クロック以下（fdcan_tq_ck < fdcan_pclk）である必要があります。CAN ユニットのクロック較正がバイパスされる場合（CCFG.BCC = ‘1’）は、このことを考慮する必要があります。***

較正の精度

状態 Precision_Calibrated での較正の精度は、以下の要素に依存します。

- CAN カーネルクロック入力 fdcan_ker_ck での動的クロック許容誤差。
- 測定エラー。較正測定に使用される各ビットシーケンスにつき、1 fdcan_pclk 周期の最大エラーがあります。ビット時間の測定に使用されるビット数は、CCFG.CFL に従い 32 ビットか 64 ビットです。
- 較正メカニズムで許容されるエラー。

2 つの較正メッセージ間の距離を選択して、FDCAN1 モジュールのクロック許容誤差要件に適合させる必要があります。

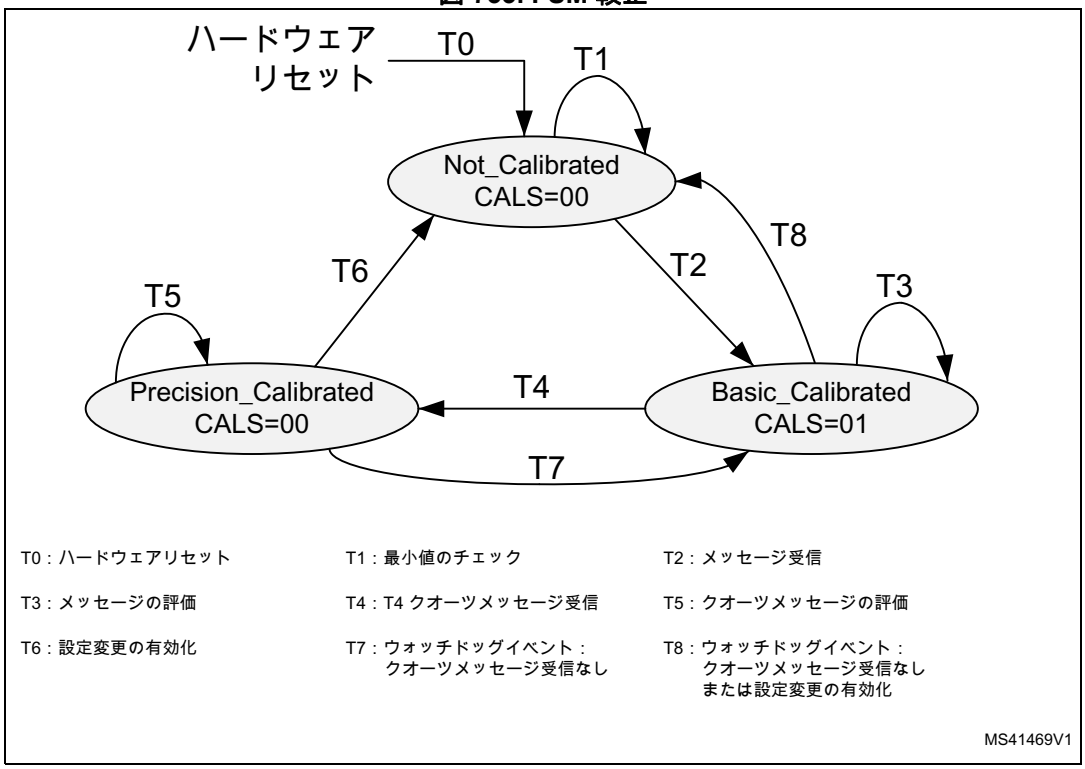
注： ***動的クロック許容誤差とは、温度や動作電圧などの変化による 2 つの較正メッセージ間のクロック周波数変動です。***

機能詳細

CAN メッセージ経由でのタイムクォンタムクロック fdcan_tq_ck の較正は、クロック fdcan_ker_ck から CAN プロトコルタイムクォンタム tq を生成するクロック分周回路を適応することで実行されます。

1. 最初の手順：基本較正(Basic Calibration)
   リセッシブからドミナントまでの 2 つのエッジ間の最小距離が測定され、この時間が PLL クロック周期でカウントされた 2 CAN ビット時間と想定されます。新しい測定でエッジ間の距離がより短くなったことが確認されるごとに、クロック分周回路が更新されます。CAN プロトコルコントローラが有効な CAN メッセージを検出すると、基本較正が達成されます。
2. 2 つ目の手順：精密較正(Precision Calibration)
   較正ステートマシンが、fdcan_ker_ck 周期の数をカウントすることで、CAN フレーム内のより長いビットシーケンスの長さを測定します。このビットシーケンスの長さは、CCFG.CLF 経由で 32 ビットまたは 64 ビットに設定できます。32/64 ビットの較正フィールド長では、最低 2/6 バイトのデータフィールドを持つ較正メッセージが必要です。精密較正は、より長いビットシーケンスの測定値から計算された新しいクロック分周値に基づきます。

図 733. FSM 較正



較正状態の変化により、割り込みフラグ CUIR.CSC もセットされます。CUIE.CSCE によって有効にされた場合、割り込みライン cu_int が有効化（ハイにセット）されます。割り込みライン cu_int は割り込みフラグ CUIR.CSC がリセットされるまでアクティブなままとなります。

精密較正が有効化されるまで、FDCAN1 および FDCAN2 は制限モード（フレーム送信なし、エラーまたはオーバーロードフラグ送信なし、エラーカウントなし）で動作します。レジスタ CSTAT から較正ステータスを評価することで、PLL の較正がソフトウェアによって実行される場合、CAN ユニットの較正が 状態 Precision_Calibrated になるまで、FDCAN1 および FDCAN2 を制限動作モード（CCCR.ASM = ‘1’）にセットする必要があります（*アプリケーション*を参照）。

精密較正は、安定したクオーツで制御されたノードによって送信される有効な CAN フレームでのみ実行できます。較正フレームは、FDCAN1 許可フィルタによって検出されます。フィルタ要素と Rx バッファを FDCAN1 で設定して較正メッセージを特定して格納する必要があります。較正メッセージ受信後、Rx バッファの新規データフラグをリセットして、次の較正メッセージの信号を有効にする必要があります。

ネットワークにクオーツクロックのある CAN トランスミッタが 1 つだけある場合、このノードはデータフィールドまたは識別子で最低 1 つの ‘1010’ シーケンスのある起動後最初のメッセージを送信する必要があります。これにより、クオーツのないノードが状態 Basic_Calibrated に移行でき、クオーツノードメッセージを確認できることが保証されます。

精密較正は、較正ウォッチドッグによって監視されている事前定義された最大の時間間隔で繰り返さなければなりません。

注： ***CAN ユニットのクロック較正が状態 Precision_Calibrated から Basic_Calibrated に遷移すると、較正 OK 信号はネゲートされ、FDCAN1 は進行中の送信を完了し、制限動作（フレーム送信なし、エラーまたはオーバーロードフラグなし、エラーカウントなし）に移行します。***

設定

CAN ユニットのクロック較正は、FDCAN1 が CCCR.CCE および CCCR.INIT ビットをセットしたときなどに、レジスタ CCFG で設定されます。

基本較正の場合、ピン FDCAN1_RX での 2 つの連続した立ち下がりエッジ間のオシレータ周期の最小数が測定されます。クロック周期の数は、入力 fdcan_ker_ck で適用されるクロック周波数に依存します。測定されたクロック周期の数が FDCAN1_RX のグリッチなどにより CCFG.OCPM で設定された最小値を下回る場合、その値は破棄され、測定が続行します。

2 つの CAN ビット時間よりわずかに少なく CCFG.OCPM を設定することを推奨します。

CCFG.OCPM <（(2 x CAN ビット時間）/ fdcan_ker_ck 周期）/ 32

精密較正に使用されるビットフィールドの長さは、CCFG.CFL 経由で 32 ビットまたは 64 ビットに設定できます。精密較正に使用されるビット数は、較正精度および 2 つの較正メッセージ間の最大距離に影響します。

CCFG.TQBT によって設定されたビット時間ごとのタイムクォンタムの数は、オシレータクロック周期の測定した数 CSTAT.OCPC とともに使用され、ビット時間ごとのオシレータクロックの数を定義します。

CCFG.BCC = ‘1’ と設定されることでクロック較正がバイパスされる場合、内部クロック分周回路を CCFG.CDIV 経由で設定して条件 fdcan_tq_ck < fdcan_pclk を満たす必要があります。

注： ***CAN のクロック較正がアクティブである場合（CCFG.BCC = ‘0’）、FDCAN1 および FDCAN2 のボーレートプリスケーラをインアクティブに設定する必要があります。***

ステータス信号

CAN ユニットのクロック較正のステータスは、レジスタ CSTAT を読み出すことで監視できます。状態 Precision_Calibrated の場合、オシレータクロック周期カウンタ CSTAT.OCPC は、較正フィールドでのオシレータクロック周期の数の信号を、CSTAT.TQC は較正フィールドでのタイムクォンタムの数の信号を送信します。

較正状態は、CSTAT.CALS によって監視されます。cu_csc パルスの継続時間は、one cu_hclk の周期です。較正状態の変化により、割り込みフラグ CUIR.CSC もセットされます。CUIE.CSCE によって有効にされた場合、割り込みライン cu_int が有効化（ハイにセット）されます。割り込みライン cu_int は割り込みフラグ CUIR.CSC がリセットされるまでアクティブなままとなります。

較正ウォッチドッグイベントにより、割り込みフラグ CUIR.CWE もセットされます。CUIE.CWEE によって有効にされた場合、割り込みライン cu_int が有効化（ハイにセット）されます。割り込みライン cu_int は割り込みフラグ CUIR.CWE がリセットされるまでアクティブなままとなります。

アプリケーション

クロック較正バイパス

CCU 内部クロック分周回路は、1 で分周設定されます（CCFG.CDIV = “0000”）。CCU 出力信号 cu_cok は ’1’ に固定されます。この動作モードでは、入力クロック fdcan_ker_ck が直接クロック出力 fdcan_tq_ck に送られます。この場合、fdcan_tq_ck は CCU に接続された FDCAN1 および FDCAN2 の設定およびステータスとは無関係です。20/40/80 MHz の fdcan_ker_ck で、CAN FD 動作は可能です。

注： ***これは、CCFG.BCC のリセット値が汎用パラメータ経由の構成で ‘1’ に設定されている場合のリセット後のデフォルト動作モードです。***

ソフトウェア較正

CAN ユニットのクロック較正は、オンチップオシレータのトリミングで fdcan_ker_ck のソフトウェア較正もサポートしています。トリミング値の計算では、ホストは CSTAT から CCU 状態を読み出す必要があります。この動作モードでは、CCU 出力信号 cu_cok が '1' に固定され、fdcan_ker_ck からのクロックは出力 fdcan_tq_ck に送られます（CCFG.BCC = '1'）。

入力クロック fdcan_ker_ck は、80～500 MHz の範囲でセットする必要があります。CCU のクロック分周回路を CCFG.CDIV で設定して、fdcan_tq_ck を有効な範囲に設定する必要があります。すべての他の設定パラメータは、CCFG 経由で設定しなければなりません。FDCAN1 および FDCAN2 が正しく動作するには、APB クロック fdcan_pclk をタイムクォンタムクロック（fdcan_tq_ck）以上にする必要があります。CAN FD 動作は不可です。

起動には、CCCR.INIT がリセットされる前に FDCAN1 および FDCAN2 を両方とも制限動作（CCCR.ASM = '1'）に設定しなければなりません。入力クロック fdcan_ker_ck は、CAN ユニットのクロック較正が状態 Precision_Calibrated に達するまで調整する必要があります。これで、ソフトウェアは CCCR.ASM をリセットできるようになり、CANFD1 および CANFD2 が通常動作を開始します。

動作中、ソフトウェアは CAN ユニットのクロック較正が状態 Precision_Calibrated のままであるかどうかを定期的にチェックする必要があります。CAN ユニットのクロック較正が fdcan_ker_ck のドリフトによって状態 Precision_Calibrated のままである場合、FDCAN1 および FDCAN2 は CCCR.INIT、CCCR.CCE、CCCR.ASM を '1' にプログラムして制限動作モードにセットする必要があります。fdcan_ker_ck が正常に調整された（CAN ユニットのクロック較正が状態 Precision_Calibrated になった）後、FDCAN1 および FDCAN2 は通常動作を再開できます。

注： ***トリミングは、設定されたビットレートの CAN クロック許容誤差要件を満たす十分な精度にする必要があります。***

クロック較正アクティブ

CCFG.BCC を '0' にセットすることで、この動作モードに入ります。この動作モードでは、CCU 出力信号 cu_cok および fdcan_ker_ck が CCU で制御されます。CCU が状態 Precision_Calibrated ではない場合、cu_cok は '0' です。

CCU 出力信号 fdcan_tq_ck および cu_cok の生成は、FDCAN1 の状態に依存します。入力クロック fdcan_ker_ck は、80～500 MHz の範囲でセットする必要があります。CCU および FDCAN1 の設定は必須です。CAN FD 動作は不可です。

FDCAN1 が Bus_Off に変わった場合や INIT ビットがホストコマンドによってセットされた場合（CCCR.INIT = '1'）、CCU は状態 Not_Calibrated に移行し、出力 cu_cok は '0' にリセットされます。CANFD1 および CANFD2 は制限動作モードに移行します。

注： ***これは、CCFG.BCC のリセット値が汎用パラメータ経由の構成で '0' に設定されている場合のリセット後のデフォルト動作モードです。***

56.3.5 TTCAN の動作（FDCAN1 のみ）

リファレンスメッセージ

リファレンスメッセージは、特定の CAN 識別子によってキャラクタライズされたデータフレームです。これは、タイムマスタ（リファレンスメッセージの送信側）を除くすべてのノードで受信され許可されます。

レベル 1 ではデータ長が最低 1、レベル 0、2 では最低 4 でなければならず、その他の場合はメッセージがリファレンスメッセージとして許可されません。リファレンスメッセージは、他のデータによって合計で 8 CAN データバイトまで拡張できます。3 つの LSB を除く識別子のすべてのビットが、リファレンスメッセージとしてメッセージを文字化します。最後の 3 ビットによって、8 個の潜在的な

タイムマスタの優先度を指定します。予約済みのビットは、論理 0 として送信され、受信側で無視されます。リファレンスメッセージは、レジスタ TTRMC 経由で設定されます。

タイムマスタがリファレンスメッセージを送信します。リファレンスメッセージがエラーによって中断された場合、すぐに再送信されます。再送信の場合、送信される Master_Ref_Mark が更新されます。リファレンスメッセージは周期的に送信されますが、周期的送信（Next_is_Gap ビット）を停止して、現在のタイムマスタまたはその他の潜在的なタイムマスタの 1 つが次の基本サイクルの開始においてイベント同期された送信を開始できます。

リファレンスメッセージを送信するノードは、現在のタイムマスタです。タイムマスタはその他のメッセージを送信できます。現在のタイムマスタが失敗した場合、その機能は最高優先度の潜在的なタイムマスタによって再現されます。タイムマスタでも潜在的なタイムマスタでもないノードは、タイム受信ノードとなります。

レベル 1

レベル 1 動作は、TTOCF[OM] = ‘01’ および TTOCF[GEN] 経由で設定されます。外部クロック同期は、レベル 1 で利用できません。リファレンスメッセージに関連した情報は、以下に示すとおり最初のデータビットに格納されます。Cycle_Count はオプションです。

表 456. レベル 1 リファレンスメッセージの最初のバイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済み	Cycle_Count[5;0]					

レベル 2

レベル 2 動作は、TTOCF[OM] = ‘10’ および TTOCF[GEN] 経由で設定されます。リファレンスメッセージに関連した情報は、以下に示すとおり最初の 4 つのデータバイトに格納されます。Cycle_Count および NTU_Res の下位 4 ビットはオプションです。TTCAN は、受信したリファレンスメッセージから NTU_Res[3:0] を評価せず、これらのビットは常に ‘0’ で送信されます。

表 457. レベル 2 リファレンスメッセージの最初の 4 バイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済み	Cycle_Count[5;0]					
2 つ目のバイト	NTU_Res[6:4]			NTU_Res[3:0]				Disc_Bit
3 つ目のバイト	Master_Ref_Mark[7:0]							
4 つ目のバイト	Master_Ref_Mark[15:8]							

レベル0

レベル 0 動作は、TTOCF[OM] = ‘11’ 経由で設定されます。外部イベント同期タイムトリガ動作は、レベル 0 で利用できません。リファレンスメッセージに関連した情報は、以下の表に示すとおり最初の 4 つのデータバイトに格納されます。レベル 0 では、Next_is_Gap は常に 0 です。Cycle_Count および NTU_Res の下位 4 ビットはオプションです。TTCAN は、受信したリファレンスメッセージから NTU_Res[3:0] を評価せず、これらのビットは常に ‘0’ で送信されます。

表 458. レベル 0 リファレンスメッセージの最初の 4 バイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済み	Cycle_Count[5;0]					
2 つ目のバイト	NTU_Res[6:4]			NTU_Res[3:0]				Disc_Bit
3 つ目のバイト	Master_Ref_Mark[7:0]							
4 つ目のバイト	Master_Ref_Mark[15:8]							

56.3.6 TTCAN の設定

TTCAN のタイミング

ネットワークタイムユニット（NTU）は、すべての時間を測定する際の単位です。NTU はネットワーク全体の定数で、ネットワークシステム設計者によって優先度として定義されます。TTCAN レベル 1 では、NTU は公称 CAN ビット時間です。TTCAN レベル 0 およびレベル 2 では、NTU は物理的な時間の断片です。

NTU はローカルタイムに対する時間ベースです。ローカルタイムの整数部（16 ビット値）は NTU ごとに 1 回インクリメントされます。サイクルタイムとグローバルタイムは両方ローカルタイムを基にしています。ローカルタイム、サイクルタイム、グローバルタイムの小数部（3 ビット値）は読み出せません。

TTCAN レベル 0 およびレベル 2 では、NTU の長さは時間単位比率（TUR）によって定義されます。TUR は通常、TUR = TURNA[NAV] / TURCF[DC] によって示される非整数です。NTU の長さは、NTU = CAN クロック周期 x TUR によって示されます。

TUR Numerator設定 NC は 18 ビットの数で、TURCF[NCL[15:0]] は 0x0000～0xFFFF の範囲でプログラムできます。TURCF[NCH[17:16]] は 0b01 にハードワイヤード(ハードで固定)になっています 0xnnnn が TURCF[NCL[15:0]] に書き込まれると、TURNA[NAV] が値 0x10000 + 0x0nnnn = 0x1nnnn で開始します。TUR Denominator設定 TURCF[DC] は 14 ビットの数です。TURCF[DC] は 0x0001～0x3FFF の範囲でプログラムできます（0x0000 は不正な値です）。

レベル 1 では、NC は 4 x TURCF[DC] 以上でなければなりません。レベル 0 およびレベル 2 では、NTU の内部少数部に 3 ビットの分解能を取得するために NC は 8 x TURCF[DC] 以上でなければなりません。

ハードウェアリセットで、TURCF[DC] は 0x1000 に、TURCF[NCL] は 0x10000 にプリセットされ、16 の CAN クロック周期から成る NTU となります。ローカルタイムおよびアプリケーションウォッチドッグは、CCCR[INIT] がリセットされるか、TURCF[ELT] がセットされる前に開始しません。TURCF[ELT] は NTU が設定される前にセットできません。TURCF[ELT] を ‘1’ にセットすると、レジスタ TURCF への書き込みアクセスもロックされます。

起動時、TURCF[ELT] がセットされていると、TURNA[NAV] は NC（= TURCF[NCL] + 0x10000）から更新されます。TTCAN レベル 1 では、ドリフト補正はありません。TURNA.NAV は動作中変化せず、常に NC になります。

TTCAN レベル 0 およびレベル 2 では、TURNA[NAV] が変化する 2 つの可能性があります。タイムスレーブまたはバックアップタイムマスタとして動作している場合、かつ TTOCF[ECC] がセットされている場合、TTCAN が同期状態 In_Schedule または In_Gap になっている限り、TURNA[NAV] は監視されたグローバルタイム速度から計算された値に自動的に更新されます。同期が失われると、NC

に戻ります。実際のタイムマスタとして動作している場合、かつ TTOCF[EECS] がセットされている場合、ホストが TURCF[NCL] を更新する場合があります。ホストが TTOCN[ECS] をセットしていると、TURNA[NAV] は次のリファレンスメッセージで NC の新しい値から更新されます。ステータスフラグ TTOST[WECS] は TTOCN[ECS] がセットされるとセットされ、TURNA[NAV] が更新されるとクリアされます。TTOST[WECS] がセットされている間、TURCF[NCL] は書き込みがロックされます。

TTCAN レベル 0 およびレベル 2 では、クロック較正プロセスは NC+/- 2(TTOCF[LDSDL]+5) の同期偏差制限 SDL の範囲で TURNA[NAV] を適応します。TURCF[NCL] は、TURNA[NAV] の計算で最高精度を達成するために適用できる最大の数値にプログラムされることが推奨されます。

同期偏差 SD は、NC と TURNA[NAV]（SD = |NC - TURNA[NAV]|）の差です。これは、同期偏差制限 SDL によって制限され、2 重対数 TTOCF[LDSDL] (SDL = 2(TTOCF[LDSDL]+5)) によって設定され、CAN ビットタイミング設定によって示されるクロック許容誤差を超えません。SD はそれぞれの新しい基本サイクルで計算されます。計算された TURNA[NAV] が NC から SDL 以上はずれる場合、またはリファレンスメッセージの Disc_Bit がセットされている場合、ドリフト補正はサスペンドされ、TTIR[GTE] はセットされ、TTOSC[QCS] がリセットされます。または、Disc_Bit = ‘1’ の場合、TTIR[GTD] がセットされます。

表 459. TUR 設定の例

TUR	8	10	24	50	510	125000	32.5	100/12	529/17
NC	0x1FFF8	0x1FFFE	0x1FFF8	0x1FFEA	0x1FFFE	0x1E848	0x1FFE0	0x19000	0x10880
TURCF.DC	0x3FFF	0x3333	0x1555	0x0A3D	0x0101	0x0001	0x0FC0	0x3000	0x0880

TTOCN[ECS] は、次のリファレンスメッセージによる有効化のために NC をスケジュールします。TTOCN[SGT] は、次のリファレンスメッセージによる有効化のために TTGTP[TP] をスケジュールします。TTOCN[ECS] および TTOCN[SGT] をセットするには、FDCAN が実際のタイムマスタであるときに TTOCF[EECS] をセット（外部クロック同期有効化）する必要があります。

TTCAN モジュールには、アプリケーションプログラムの機能を検証するためのアプリケーションウォッチドッグが備わっています。ホストはこのウォッチドッグを定期的に機能させる必要があり、そうしないとすべての CAN バス動作が停止します。アプリケーションウォッチドッグ制限 TTOCF[AWL] は、ウォッチドッグを機能させる必要がある 2 つの時間に挟まれた NTU の数値を指定します。NTU の最大数は 256 です。アプリケーションウォッチドッグは、レジスタ TTOST を読み出すことで機能します。TTOST[AWE] はウォッチドッグが時間内に機能したかどうかを示します。アプリケーションがアプリケーションウォッチドッグを機能させられなかった場合、割り込みフラグ TTIR[AW] がセットされます。ソフトウェア開発の場合、アプリケーションウォッチドッグは TTOCF[AWL] を 0x00 にプログラムすることによって無効化できます。*セクション 56.4.49 : FDCAN TT 動作設定レジスタ（FDCAN_TTOCF）*を参照してください。

インタフェース信号のタイミング

出力 FDCAN のトリガタイムマーク割り込みパルス m_ttcan_tmp と FDCAN のレジスタタイムマーク割り込みパルス pulsem_ttcan_rtp でパルスを発生させるタイミングイベントは、CAN クロックドメインで生成されます。同じイベントが APB クロックドメインに表示される前に考慮すべきクロックドメインに跨る遅延があります（TTIR[TTMI] がセットされる場合または TTIR[RTMI] がセットされる場合）。たとえば、信号を他の FDCAN ノード（fdcan_swt/fdcan_evt）のタイミング入力に接続して、2 つの TTCAN ネットワークを自動同期できます。

出力 cyclem_ttcan_soc の FDCAN 開始は、リファレンスメッセージが完了（送信または受信）すると必ずアクティブになります。この出力は APB クロックドメインで制御されます。

56.3.7 メッセージのスケジューリング

TTOCF[TM] は、TTCAN が潜在的なタイムマスタとタイムスレーブのどちらで動作するかを制御します。潜在的なタイムマスタの場合、リファレンスメッセージ識別子 TTRMC[RID] の 3 つの LSB が、最高で 0、最低で 7 のマスタ優先度を定義します。ネットワークに同じマスタ優先度を使用した 2 つのノードを配置できません。TTRMC[RID] はリファレンスメッセージの認識に使用されます。TTRMC[RMPS] はタイムスレーブでは関係ありません。

初期リファレンストリガオフセット TTOCF[IRTO] は、リファレンスメッセージが期待されるがバスがアイドルのままのときに、リファレンスメッセージの送信を開始する前にバックアップタイムマスタが待機する時間を（NTU 単位で）定義する 7 ビット値です。TTOCF[IRTO] の推奨値は、ネットワークの潜在的なタイムマスタ間で予測されるクロックドリフトに応じた係数で掛けられたマスタ優先度です。バックアップタイムマスタの連続した順番は、現在のタイムマスタが失敗した場合にそのうち 1 つがリファレンスメッセージを開始するときに、最大クロックドリフトでもマスタ優先度に対応します。

TTOCF[OM] は、ノードが TTCAN レベル 0、レベル 1、レベル 2 のいずれで動作するかを決定します。1 つのネットワークでは、すべての潜在的なタイムマスタが同じレベルで動作する必要があります。タイムスレーブはレベル 2 のネットワークでレベル 1 で動作できますが、その逆はできません。TTOCF[OM] 経由での TTCAN 動作モードの設定はセットアップの最後の手順です。TTOCF[OM] = ‘00’（イベント駆動 CAN 通信）では、FDCAN は ISO 11898-1: 2015 に従ってタイムトリガなしで動作します。TTOCF[OM] = ‘01’（レベル 1）では、FDCAN は ISO 11898-4 に従って動作しますが、基本サイクルを外部イベントに同期することはできず、リファレンスメッセージの Next_is_Gap ビットは無視されます。TTOCF[OM] = ‘10’（レベル 2）では、TTCAN は基本サイクルのイベント同期による開始を含め、ISO 11898-4 に従って動作します。TTOCF[OM] = ‘11’（レベル 0）では、FDCAN はイベント駆動 CAN として動作しますが、レベル 2 のように較正されたグローバルタイムベースを維持します。

TTOCF[EECS] は外部クロック同期を有効にし、現在のタイムマスタのアプリケーションプログラムがタイムトリガ動作中に TUR 設定を更新して、クロック速度およびグローバルクロックフェーズ（レベル 0 とレベル 2 のみ）を外部リファレンスに適応できるようにします。

TT マトリックス制限レジスタの TTMLM[ENTT] は、システムマトリックスで予測される Tx_Trigger の数を指定します。これは、Tx_Ref_Trigger を除く排他的なシングルアービトレーションおよび統合アービトレーションウィンドウの Tx_Trigger の合計です。これは通常、Tx_Trigger メモリ要素の数、システムマトリックスの基本サイクルの数ではなく、トリガ繰り返し係数を考慮する必要があります。TTMLM[ENTT] が正確に設定されていないと、Tx カウントアンダーフロー（TTIR[TXU] = ‘1’ および TTOST[EL] = ‘01’、深刻度 1）または Tx カウントオーバーフロー（TTIR[TXO] = ‘1’ および TTOST[EL] = ‘10’、深刻度 2）になります。

注： ***ノードで見られる最初のリファレンスメッセージで Cycle_Count が 0 にされていない場合、このノードは Tx カウントアンダーフロー条件となる Tx カウントのある最初のマトリックスサイクルで終了するでしょう。ノードが同期中の状態である限り、Tx_Trigger は送信に移行しません。***

TTMLM[CCM] はシステムマトリックスで最後の基本サイクルの数を指定します。基本サイクルのカウントは 0 で開始します。8 個の基本サイクルから成るシステムマトリックスでは、TTMLM[CCM] は 7 になります。TTMLM[CCM] がタイムスレーブで無視されると、受信側は実際の基本サイクルで有効なサイクルカウントを、受信したサイクルカウントとみなします。

TTMLM[TXEW] は NTU で Tx 有効化ウィンドウの長さを指定します。Tx 有効化ウィンドウは、送信が開始されるタイムウィンドウの冒頭の時間を指します。たとえば、前のタイムウィンドウのメッセージからわずかにオーバーラップがあるなど、Tx 有効化ウィンドウ内でメッセージの送信を開始できない場合、そのタイムウィンドウでは一切送信できません。TTMLM[TXEW] は、ネットワーク同期品質およびタイムウィンドウの長さとメッセージの長さの関係で選択される必要があります。

トリガメモリ

トリガメモリは、TTCAN が接続されている外部メッセージ RAM の一部です（*セクション 56.4.26 : FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）*を参照）。これには、最大 64 のトリガ要素が格納されます。トリガメモリ要素は、タイムマーク TM、サイクルコード CC、トリガタイプ TYPE、フィルタタイプ FTYPE、メッセージ数 MNR、メッセージステータスカウント MSC、タイムマークイベント内部 TMIN、タイムマークイベント外部 TMEX から成ります（*セクション 56.3.21 : FDCAN のトリガメモリ要素*を参照）。

タイムマークはトリガがアクティブになるサイクルタイムを定義します。トリガメモリのトリガは、タイムマークごとにソートされる必要があります。最小タイムマークのあるトリガ要素は、最初のトリガメモリワードに書き込まれます。メッセージ数とサイクルコードは、Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、Watch_Trigger_Gap、End_of_List タイプのトリガでは無視されます。

サイクルタイムが実際のトリガのタイムマークに達すると、FSE が次のトリガに切り替え、トリガメモリから次のトリガの読み出しを開始します。送信トリガの場合、FSE がそのトリガに切り替えると、Tx ハンドラがメッセージ RAM からメッセージの読み出しを開始します。RAM アクセス速度は、送信トリガとその前のトリガの間の最小タイムステップを定義し、Tx ハンドラは送信トリガタイムマークに達する前に送信を準備しなければなりません。RAM アクセス速度は、2 つの一致しているトリガ間の（サイクルコードにおける）不一致トリガの数を制限し、次の一致するトリガをタイムマークに達する前に読み出す必要があります。リファレンスメッセージが n NTU 長である場合、n 未満のタイムマークを持つトリガはアクティブになることはなく、設定エラーとして扱われます。

サイクルタイムの開始点は、リファレンスメッセージのフレーム開始ビットのサンプルポイントです。サイクルタイムが Tx_Ref_Trigger タイムマークに達すると、次のリファレンスメッセージがリクエストされます。FDCAN は次のサンプルポイントで送信リクエストに反応します。新しい Sync_Mark はフレーム開始ビットでキャプチャされますが、サイクルタイムはリファレンスメッセージが正常に送信（または受信）されるまでインクリメントされ、Sync_Mark が新しい Ref_Mark として取得されます。時間内のその位置で、サイクルタイムが再開します。結果として、サイクルタイムは（初期化を除いて）n 未満の値で見られることはなく、n は NTU で測定されたリファレンスメッセージの長さとなります。

基本サイクルの長さは、Tx_Ref_Trigger タイムマーク+1 NTU+1 CAN ビット時間です。

このトリガリストは、FDCAN ネットワークのすべてのノードで異なります。各ノードは、自身の送信メッセージの Tx_Trigger、このノードによって処理される受信メッセージの Rx_Trigger、リファレンスメッセージに関するトリガのみ把握しています。

トリガタイプ

Tx_Ref_Trigger（TYPE = ‘0000’）および Tx_Ref_Trigger_Gap（TYPE = ‘0001’）で、タイムマスタによってリファレンスメッセージが送信されます。設定エラー（TTOST[EL] = ‘11’、深刻度 3）は、タイムスレーブがトリガメモリで Tx_Ref_Trigger（_Gap）に遭遇したときに検出されます。Tx_Ref_Trigger_Gap は外部イベント同期
タイムトリガ動作モードでのみ使用されます。このモードでは、FDCAN 同期状態が In_Gap（TTOST[SYS] = ‘10’）であるときに Tx_Ref_Trigger は無視されます。

Tx_Trigger_Single（TYPE = ‘0010’）、Tx_Trigger_Continuous（TYPE = ‘0011’）、Tx_Trigger_Arbitration（TYPE = ‘0100’）、Tx_Trigger_Merged（TYPE = ‘0101’）で送信が開始されます。これらは、タイムウィンドウの開始を定義します。

メッセージバッファの送信リクエストペンディングビットがセットされているとき、Tx_Trigger_Single は排他的タイムウィンドウで 1 つの送信を開始します。正常に送信された後、送信リクエストペンディングビットはリセットされます。

メッセージバッファの送信リクエストペンディングビットがセットされているとき、Tx_Trigger_Continuous は排他的タイムウィンドウで送信を開始します。正常に送信された後、送信リクエストペンディングビットはセットされたままとなり、メッセージバッファが次の一致するタイムウィンドウで再度送信されます。

Tx_Trigger_Arbitration はアービトレーションタイムウィンドウを、Tx_Trigger_Merged は統合アービトレーションタイムウィンドウを開始します。統合アービトレーションタイムウィンドウの最後の Tx_Trigger は、タイプ Tx_Trigger_Arbitration でなければなりません。設定エラー（TTOST[EL] = ‘11’、深刻度 3）は、タイプ Tx_Trigger_Merged のトリガの後にタイプ Tx_Trigger_Merged または Tx_Trigger_Arbitration のうち 1 つを除く他の Tx_Trigger が続くときに検出されます。複数の Tx_Trigger を同じ Tx メッセージバッファに対して定義できます。それらのサイクルコードに応じて、一部の基本サイクルでは無視される場合があります。予測される Tx_Trigger の数（TTMLM[ENTT]）が計算されるときに、サイクルコードを考慮する必要があります。

Watch_Trigger（TYPE = ‘0110’）および Watch_Trigger_Gap（TYPE = ‘0111’）は、不足しているリファレンスメッセージをチェックします。これらはタイムマスタおよびタイムスレーブ両方で使用されます。Tx_Ref_Trigger_Gap は外部イベント同期タイムトリガ動作モードでのみ使用されます。このモードでは、FDCAN 同期状態が In_Gap （TTOST[SYS] = ‘10’）であるときに Watch_Trigger は無視されます。

Rx_Trigger（TYPE = ‘1000’）は、排他的タイムウィンドウでの周期的メッセージの受信をチェックするために使用します。Rx_Trigger は、状態 In_Schedule または In_Gap に達するまでアクティブではありません。Rx_Trigger のタイムマークはタイムウィンドウの境界にかかわらず、そのメッセージ送信終了後に配置されます。それらのサイクルコードに応じて、一部の基本サイクルでは Rx_Trigger が無視される場合があります。Rx_Trigger のタイムマークで、サイクルの開始または前の Rx_Trigger の後からこのタイムマークの前までで最後に受信したメッセージが MNR によって参照される許可フィルタ要素と一致したかどうかをチェックされます。許可されるメッセージは、Rx_Trigger にかかわらず、許可フィルタに従って 2 つの受信 FIFO のうち 1 つに格納されます。Rx_Trigger によって参照される許可フィルタ要素は、フィルタリストの最初に配置し、Rx_Trigger タイムマークに達する前にフィルタが終了することを保証します。

Time_Base_Trigger（TYPE = ‘1001’）は、TMIN および TMEX の設定に応じて内部／外部イベントの生成に使用します。

End_of_List（TYPE = ‘1010～1111’）は不正トリガタイプで、Watch_Trigger および Watch_Trigger_Gap の前にトリガメモリで End_of_List トリガに遭遇した場合に設定エラー（TTOST[EL] = ‘11’、深刻度 3）が検出されます。

ノードトリガリストの制限

2 つのトリガを同じサイクルタイムでアクティブにできませんが、異なる基本サイクル（異なるサイクルコード）でアクティブになっているトリガは同じタイムマークを共有できます。

Rx_Trigger および Time_Base_Trigger は、Tx_Trigger_Single/Continuous/Arbitration の Tx 有効化ウィンドウ内に配置できませんが、Tx_Trigger_Merged の後に配置できます。

Watch_Trigger（または TTOST[SYS] = ‘10’ の場合は Watch_Trigger_Gap）の後に配置されるトリガはアクティブになることはありません。リファレンスメッセージが時間通りに送信されたとき、ウォッチトリガ自体はアクティブになりません。

使用されていないすべてのメモリワード（Watch_Trigger の後、または TTOST[SYS] = ‘10’ の場合は Watch_Trigger_Gap の後）は、トリガタイプ End_of_List にセットする必要があります。

潜在的なタイムマスタの通常のトリガリストは、Tx_Trigger および Rx_Trigger の数で始まり、Tx_Ref_Trigger および Watch_Trigger に続きます。外部イベント同期タイムトリガ通信のあるネットワークでは、これに Tx_Ref_Trigger_Gap および Watch_Trigger_Gap が続きます。タイムスレーブのトリガリストは同じですが、Tx_Ref_Trigger および Tx_Ref_Trigger_Gap がありません。

リファレンスメッセージが受信または送信される各基本サイクルの開始時に、最初のトリガメモリ要素からトリガリストが処理されます。FSE は現在のサイクルカウントに一致するサイクルコードのある最初のトリガを検索します。FSE は、サイクルタイムがトリガタイムマークに達するまで待って、トリガを有効化します。その後、FSE は現在のサイクルカウントに一致するサイクルコードのあるリスト内の次のトリガを検索します。

Tx_Ref_Trigger および Tx_Ref_Trigger_Gap 前後の時間には特別な配慮が必要です。マスタ権限で競合するタイムマスタでは、Tx_Ref_Trigger の有効なタイムマークをリファレンスメッセージを開始する最初のノードにするためにデクリメントできます。バックアップタイムマスタでは、Tx_Ref_Trigger や Tx_Ref_Trigger_Gap の有効なタイムマークは、設定されたタイムマークとリファレンストリガオフセット TTOCF[IRTO] の合計です。エラーレベル 2 に達した場合（TTOST[EL] = ‘10’）、有効なタイムマークはそのタイムマークと 0x127 の合計になります。他のトリガ要素はこの範囲に配置しないようにします。そうしないと、タイムマークが正常に動作せず、設定エラーとしてフラグされる場合があります。Tx_Ref_Trigger 以降のトリガ要素は、リファレンスメッセージが時間内に到着しない限りアクティブになることはありません。

次のパラメータは相互依存しています。

- APB クロック周波数
- トリガ RAM アクセスの速度と待機時間
- 許可フィルタリストの長さ
- トリガ要素の数
- トリガ要素のサイクルコードフィルタの複雑性
- トリガ要素のタイムマーク間のオフセット

トリガ処理の例

この例では、ノードシステムマトリックスからトリガリストを生成する方法を示します。ノード A が最初のタイムマスタで、*表 460* に示されたシステムマトリックスのセクションについて把握しているものと仮定します。

表 460. システムマトリックス、ノード A

サイクルカウント	タイムマーク						
	1	2	3	4	5	6	7
0	Tx7	-	-	-	-	TxRef	エラー
1	Rx3	-	Tx2, Tx4		-	TxRef	エラー
2	-	-	-	-	-	TxRef	エラー
3	Tx7	-	Rx5	-	-	TxRef	エラー
4	Tx7	-	-	Rx6	-	TxRef	エラー

サイクルカウントは 0 から始まり、0、1、3、7、15、31、63 まで動作します（システムマトリックスで対応する基本サイクルの数は 1、2、4、8、16、32、64 です）。最大サイクルカウントは、TTMLM.CCM で設定されます。サイクルコード CC は、繰り返し係数（= 最上位 ‘1’ の値）とシステムマトリックスの最初の基本サイクルの数（= 最上位 ‘1’ の後のビットフィールド）で構成されます。

例 : 0b0010011 のサイクルコード（繰り返し係数 = 16、最初の基本サイクル = 3）および TTMLM.CCM = ‘0x3F’ の最大サイクルカウントでは、サイクルカウント 3、19、35、51 で一致が発生します。

トリガ要素は、タイムマーク TM、サイクルコード CC、トリガタイプ TYPE、メッセージ数 MNR から成ります。送信では、MNR は Tx バッファ数（0～31）を参照します。受信では、許可フィルタ中

に一致したフィルタ要素の数（0～127）を参照します。フィルタタイプ FTYPE の設定に応じて、11 ビットまたは 29 ビットのメッセージ ID フィルタリストが参照されます。

さらに、タイムマークイベント内部 TMIN とタイムマークイベント外部 TMEX の生成にトリガ要素を設定できます。メッセージステータスカウント MSC では、トリガ要素のタイムマークがアクティブになった時点で排他的タイムウィンドウで周期的メッセージのスケジュールエラーのカウンタ値（0～7）が保持されます。

表 461. トリガリスト、ノード A

トリガ	タイムマーク TM[15:0]	サイクルコード CC[6:0]	トリガタイプ TYPE [3:0]	メッセージ数 MNR [6:0]
0	マーク 1	0b0000100	Tx_Trigger_Single	7
1	マーク 1	0b1000000	Rx_Trigger	3
2	マーク 1	0b1000011	Tx_Trigger_Single	7
3	マーク 3	0b1000001	Tx_Trigger_Merged	2
4	マーク 3	0b1000011	Rx_Trigger	5
5	マーク 4	0b1000001	Tx_Trigger_Arbitration	4
6	マーク 4	0b1000100	Rx_Trigger	6
7	マーク 6	N/A	Tx_Ref_Trigger	0（Ref）
8	マーク 7	N/A	Watch_Trigger	N/A
9	N/A	N/A	End_of_List	N/A

Tx_Trigger_Single、Tx_Trigger_Continuous、Tx_Trigger_Merged、Tx_Trigger_Arbitration、Rx_Trigger、および Time_Base_Trigger は指定されたサイクルコードにのみ有効です。他のすべてのトリガタイプについては、サイクルコードは無視されます。

FSE は、サイクルタイムを超えるタイムマークおよび実際のサイクルカウントに一致するサイクルコード CC を持つトリガに達するか、タイプ Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、または Watch_Trigger_Gap のトリガに遭遇するまで 0 からトリガリストをスキャンしつつ基本サイクルを開始します。

サイクルタイムがタイムマーク TM に達すると、トリガタイプ TYPE とメッセージ数 MNR によって定義された動作が開始します。End_of_List に達すると、設定にエラーがあります。

マーク 6 で、リファレンスメッセージ（常に TxRef）が送信されます。リファレンスメッセージの送信後、FSE はトリガリストの最初に戻ります。マーク 7 でウォッチトリガに達すると、ノードはリファレンスメッセージを送信できなくなり、エラー処理が開始します。

設定エラーの検出

設定エラーは、次の場合に TTOST[EL] = ‘11’（深刻度 3）経由で信号を送信します。

- FSE は現在のサイクルカウントに一致するサイクルコードがあり、サイクルタイム未満のタイムマークがあるリスト内のトリガに達する場合
- 前のアクティブなトリガが Tx_Trigger_Merged であり、FSE は現在のサイクルカウントに一致するサイクルコードがあり、Tx_Trigger_Merged、Tx_Trigger_Arbitration、Time_Base_Trigger、Rx_Trigger のいずれでもないリスト内のトリガに達する場合
- TTOCF[TM]=‘0’のノードの FSE（タイムスレーブ）が Tx_Ref_Trigger または Tx_Ref_Trigger_Gap に遭遇する場合
- いずれかのタイムマークが、一致するサイクルコードのある Tx_Trigger の Tx 有効化ウィンドウ（TTMLM[TXEW] によって定義）内に配置された場合
- タイムマークが Tx_Ref_Trigger のタイムマーク付近に配置され、リファレンストリガオフセット TTOST[RTO] によりサイクルタイムで測定された連続する順番に逆転が発生する場合

TTCAN のスケジュール初期化

TTCAN のメッセージスケジュールへの同期は、CCCR[INIT] がリセットされると開始します。TTCAN は厳密なタイムトリガ（TTOCF[GEN] = ‘0’）または外部イベント同期タイムトリガ（TTOCF[GEN] = ‘1’）を動作できます。すべてのノードは、TTOST[SYS] = ‘00’（同期外）のトリガリストの最初にサイクルタイム 0 から開始し、リファレンスメッセージを除いて送信は有効になりません。外部イベント同期タイムトリガ動作モードのノードは、Tx_Ref_Trigger および Watch_Trigger を無視し、最初のリファレンスメッセージが Gap がアクティブかどうか判断するまで Tx_Ref_Trigger_Gap と Watch_Trigger_Gap を代わりに使用します。

タイムスレーブ

設定後、タイムスレーブは Watch_Trigger に達する前にメッセージを受信しなかった場合に、Watch_Trigger と Watch_Trigger_Gap を無視します。Init_Watch_Trigger に達した場合、割り込みフラグ TTIR[IWT] がセットされ、FSE は停止し、サイクルタイムは無効になりますが、ノードは （確認応答またはエラーフラグを送信するために）CAN バス通信に参加したままにできます。最初のリファレンスメッセージの受信によって FSE とサイクルタイムが再開します。

注： ***Init_Watch_Trigger はトリガリストの一部ではありません。0xFFFF = 最大サイクルタイムまでカウントする内部カウンタとして実装されています。***

タイムスレーブが Watch_Trigger に達する前にリファレンスメッセージ以外のメッセージを受信した場合、致命的エラー（TTOST[EL] = ‘11’、深刻度 3）とみなされ、割り込みフラグ TTIR[WT] がセットされ、CAN バス出力がオフになり、バスモニタモードに移行します（CCCR[MON] が ‘1’ にセットされます）。バスモニタモードでは、メッセージをまだ受信できますが、ドミナントビットを送信できないため、確認応答を付与できなくなります。

注： ***致命的エラー状態から抜けるには、ホストが CCCR[INIT] = ‘1’ をセットする必要があります。CCCR[INIT] のリセット後、ノードは TTCAN 通信を再開します。***

同期中にエラーに遭遇しなかった場合、最初のリファレンスメッセージは TTOST[SYS] = ‘01’（Synchronizing）をセットし、2 つ目が FDCAN 同期状態（Next_is_Gap ビットに応じて）を TTOST[SYS] = ‘11’（In_Schedule）または TTOST[SYS] = ‘10’（In_Gap）にセットしてすべての Tx_Trigger と Rx_Trigger を有効にします。

潜在的なタイムマスタ

設定後、潜在的なタイムマスタが Tx_Ref_Trigger（または外部イベント同期タイムトリガ動作の場合は Tx_Ref_Trigger_Gap）に達すると、リファレンスメッセージの送信を開始します。メッセージを受信しなかったり、Watch_Trigger に達する前にリファレンスメッセージを正常に送信できなかった

場合、Watch_Trigger および Watch_Trigger_Gap は無視されます（想定される理由：他のすべてのノードがリセット中または設定中で確認応答を付与しない）。Init_Watch_Trigger に達した場合、試行された送信はアボートされ、割り込みフラグ TTIR[IWT] がセットされ、FSE は停止し、サイクルタイムは無効になりますが、ノードは （確認応答またはエラーフラグを送信するために）CAN バス通信に参加したままにできます。TTIR[IWT] をリセットすると、Init_Watch_Trigger 条件が次に満たされるか、別の CAN メッセージを受信するまでリファレンスメッセージの送信を再度有効にします。FSE は、リファレンスメッセージの受信によって再開されません。

潜在的なタイムマスタがリファレンスメッセージ以外のメッセージを受信した後に Watch_Trigger に達した場合、致命的エラー（TTOST[EL] = ‘11’、深刻度 3）とみなされ、割り込みフラグ TTIR[WT] がセットされ、CAN バス出力がオフになり、バスモニタモードに移行します（CCCR[MON] が ‘1’ にセットされます）。バスモニタモードでは、メッセージをまだ受信できますが、ドミナントビットを送信できないため、確認応答を付与できなくなります。

初期化中にエラーが検出されなかった場合、最初のリファレンスメッセージは TTOST[SYS] = ‘01’（Synchronizing）をセットし、2 つ目が FDCAN 同期状態（Next_is_Gap ビットに応じて）を TTOST[SYS] = ‘11’（In_Schedule）または TTOST[SYS] = ‘10’（In_Gap）にセットしてすべての Tx_Trigger と Rx_Trigger を有効にします。

潜在的なタイムマスタが最後のリファレンスメッセージの送信側であった場合、現在のタイムマスタとなります（TTOST[MS] = ‘11’）。それ以外は、バックアップタイムマスタとなります（TTOST[MS] = ‘10’）。

すべての潜在的なタイムマスタが設定を終了すると、ネットワークの最高タイムマスタ優先度を持つノードが現在のタイムマスタとなります。

56.3.8 TTCAN ギャップ制御

ギャップ制御に関連するすべての機能は、FDCAN が外部イベント同期タイムトリガモード（TTOCF[GEN] = ‘1’）で動作している場合のみ適用されます。この動作モードでは、FDCAN メッセージスケジュールはシステムマトリックスの基本サイクルの間にギャップを挿入することで割り込みできます。CAN ネットワークに接続されたすべてのノードは、外部イベント同期タイムトリガ動作に設定する必要があります。

ギャップ中、すべての送信は停止し、CAN バスはアイドル状態のままになります。次のリファレンスメッセージによって新しい基本サイクルが開始するとギャップは終了します。ギャップは、ビット Next_is_Gap = ‘1’ のリファレンスメッセージによって開始された基本サイクルの最後に開始します。例えば、ギャップは現在のタイムマスタによって開始されます。

現在のタイムマスタにはギャップを開始する 2 つのオプションがあります。アプリケーションプログラムが TTOCN[NIG] = ‘1’ と書き込むと、ソフトウェア制御下でギャップを開始できるようになります。Next_is_Gap は、次のリファレンスメッセージとともに ‘1’ で送信されます。アプリケーションプログラムが TTOCN[NIG] = ‘1’ と書き込むことでイベントトリガ入力ピン fdcan_evt を有効にすると、ハードウェア制御下でもギャップを開始できるようになります。リファレンスメッセージが開始され、TTOCN[GCS] がセットされると、イベントトリガピン fdcan_evt のハイレベルによって、Next_is_Gap = ‘1’ がセットされます。

リファレンスメッセージが完了すると、TTOST[WFE] ビットによってタイムマスタとタイムスレーブにギャップが通知されます。現在の基本サイクルは最後のタイムウィンドウまで続行します。最後のタイムウィンドウの後の時間はギャップタイムです。

実際のタイムマスタおよび潜在的なタイムマスタでは、最後の基本サイクルが終了し、ギャップタイムが開始すると、TTOST[GSI] がセットされます。タイムスレーブのノードでは、ビット TTOST[GSI] は ‘0’ のままとなります。

潜在的なタイムマスタが同期状態 In_Gap（TTOST[SYS] = ‘10’）の場合、意図的にギャップを終了する 4 つのオプションがあります。

ソフトウェア制御では、TTOCN[FGP] = ‘1’ と書き込みます。

ハードウェア制御（TTOCN[GCS] = ‘1’）では、イベントトリガ入力ピン fdcan_evt のハイからローのエッジで TTOCN[FGP] をセットし、スケジュールを再開します。

3 つ目のオプションは、タイムトリガ再開です。TTOCN[TMG] = ‘1’ のとき、次のレジスタタイムマークの割り込み（TTIR[RTMI] = ‘1’）で TTOCN[FGP] をセットし、リファレンスメッセージを開始します。

最後は、潜在的なタイムマスタが同期するイベントが時間内に発生しないことを想定して、Tx_Ref_Trigger_Gap に達するとギャップを終了します。

これらのどのオプションでも、基本サイクルにリファレンスメッセージで割り込ませることはできません。

ギャップタイムが開始した後に TTOCN[FGP] をセットすることで、すぐにリファレンスメッセージの送信が開始され、メッセージスケジュールが同期されます。ギャップタイムが開始する前（基本サイクルが進行中）に TTOCN[FGP] をセットする場合、次のリファレンスメッセージが Tx_Ref_Trigger で基本サイクルの最後に開始されます。メッセージスケジュールにギャップタイムが生じません。

厳密なタイムトリガ動作では、リファレンスメッセージでのビット Next_is_Gap = ‘1’ は無視されます。イベントトリガ入力ピン fdcan_evt およびビット TTOCN[NIG]、TTOCN[FGP]、TTOCN[TMG] も同様です。

56.3.9 ストップウォッチ

ストップウォッチ機能は、外部イベントによってトリガされる FDCAN 内部時間値（ローカルタイム、サイクルタイム、またはグローバルタイム）のキャプチャを可能にします。

ストップウォッチ機能を有効にするには、まずアプリケーションプログラムで TTOCN[SWS] 経由でストップウォッチソースとしてローカルタイム、サイクルタイム、グローバルタイムを定義する必要があります。TTOCN[SWS] が ‘00’ ではなく、TT 割り込みレジスタフラグ TTIR[SWE] が ‘0’ である場合、TTTOCN[SWS] によって選択された時間の実際の値は、ストップウォッチトリガピン fdcan_swt の次の立ち上がり／立ち下がりエッジ（TTOCN[SWP] 経由で設定）の TTCPT[SWV] にコピーされます。これで、割り込みフラグ TTIR[SWE] がセットされます。アプリケーションプログラムが TTCPT[SWV] を読み出した後、TTIR[SWE] を ‘0’ にリセットすることで次のストップウォッチイベントを有効にできます。

56.3.10 ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期

タイムトリガ CAN には 2 つのレベルがあります。レベル 1 とレベル 2 です。レベル 1 はサイクルタイムを使用するタイムトリガ動作のみを提供します。レベル 2 はさらに同期品質を向上し、グローバルタイムと外部クロック同期を提供します。両方のレベルでは、すべてのタイミング機能がローカルタイムベースに基づきます。

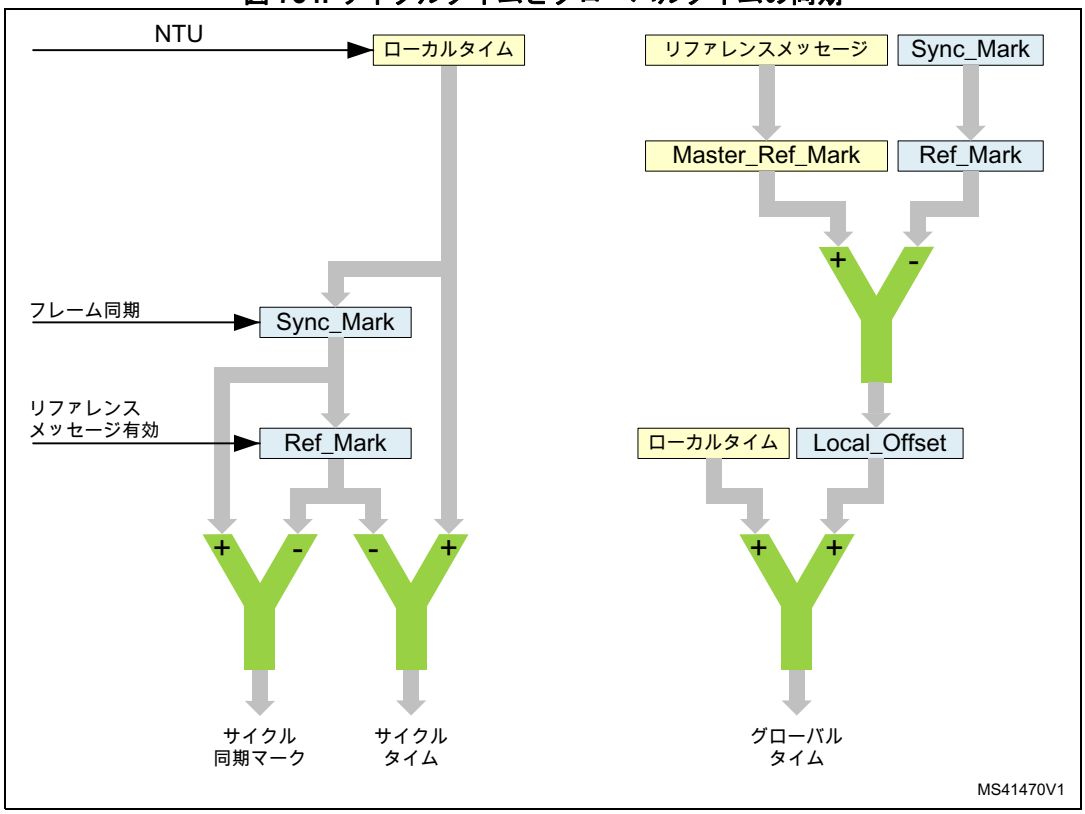
ローカルタイムは、16 ビットの巡回カウンタ値で、NTU ごとに 1 回インクリメントされます。内部では、NTU はローカルタイムの小数部分（3ビット）とみなすことができる 3 ビットカウンタによって示されます。一般的に、3 ビット NTU カウンタは NTU ごとに 8 回インクリメントされます。NTU の長さが（レベル 1 で設定できる、またはレベル 2 のクロック較正の結果としての）8 CAN クロック周期より短い場合、NTU 端数の長さが適応され、NTU カウンタは NTU ごとに 4 回だけインクリメントされます。

図 734 では、タイムマスタを含むすべての FDCAN ノードで同じ方法で実行されるサイクルタイムとグローバルタイムの同期について示しています。メッセージの受信または送信によって、メッセージで取得されるローカルタイムのキャプチャが起動し、フレーム同期イベントとなります。このフレーム同期イベントは、各フレーム開始（SoF）ビットのサンプルポイントで発生し、ローカルタイムを

Sync_Mark として格納します。Sync_Mark および Ref_Mark は 3 ビット小数部分を含めてキャプチャされます。

有効なリファレンスメッセージが送信または受信されるたびに、内部 Ref_Mark が Sync_Mark から更新されます。Ref_Mark と Sync_Mark の差は、レジスタ TTCSM に格納されるサイクル同期マーク（サイクル同期マーク = Sync_Mark - Ref_Mark）です。Ref_Mark とローカルタイムの実際の値の差における最上位の 16 ビットがサイクルタイム（サイクルタイム = ローカルタイム - Ref_Mark）となります。

図 734. サイクルタイムとグローバルタイムの同期



TTCTC[CT] から読み出すことができるサイクルタイムは、ノードのローカルタイムと Ref_Mark の差で、両方とも APB クロックドメインに同期され、16 ビットに切り詰められます。

グローバルタイムは TTCAN レベル 0 およびレベル 2 にのみ存在し、レベル 1 では無効です。グローバルタイムのノードの表示は、NTU でのグローバルタイムのローカルイメージです。設定すると、潜在的なタイムマスタは自身のローカルタイムをグローバルタイムとして使用します。タイムマスタはリファレンスメッセージ（バイト 3 および 4）で自身の Ref_Mark を Master_Ref_Mark として送信することでローカルタイムをグローバルタイムとして確立します。TTLGT[GT] から読み出すことができるグローバルタイムは、ノードのローカルタイムとローカルオフセットの合計で、両方とも APB クロックドメインに同期され、16 ビットに切り詰められます。小数部分は、クロック同期にのみ使用されます。

リファレンスメッセージを受信するノードは、ローカル Ref_Mark と受信した Master_Ref_Mark を比較することで、グローバルタイムに対してローカルオフセットを計算します（*図 735* を参照）。グローバルタイムのノードの表示は、グローバルタイム + ローカルオフセットです。他のタイムマスタのリファレンスメッセージを受信したことがない潜在的なタイムマスタでは、Local_Offset が 0 になります。ノードが最初に他のリファレンスメッセージを受信した後に現在のタイムマスタとなる場合、Local_Offset は最後の値で固定されます。タイム受信ノードでは、他のノードがタイムマスタになる

場合や、リファレンスメッセージで Disc_Bit の信号によってグローバルタイムが不連続であった場合、Local_Offset はクロックドリフトによる微調整に従います。グローバルタイムが不連続である場合を除いて、レジスタ TTLGT によってアプリケーションプログラムに提供されるグローバルタイムは、ローパスフィルタによって平滑化され、連続した単調な値を持ちます。

図 735. TTCAN レベル 0 およびレベル 2 のドリフト補正

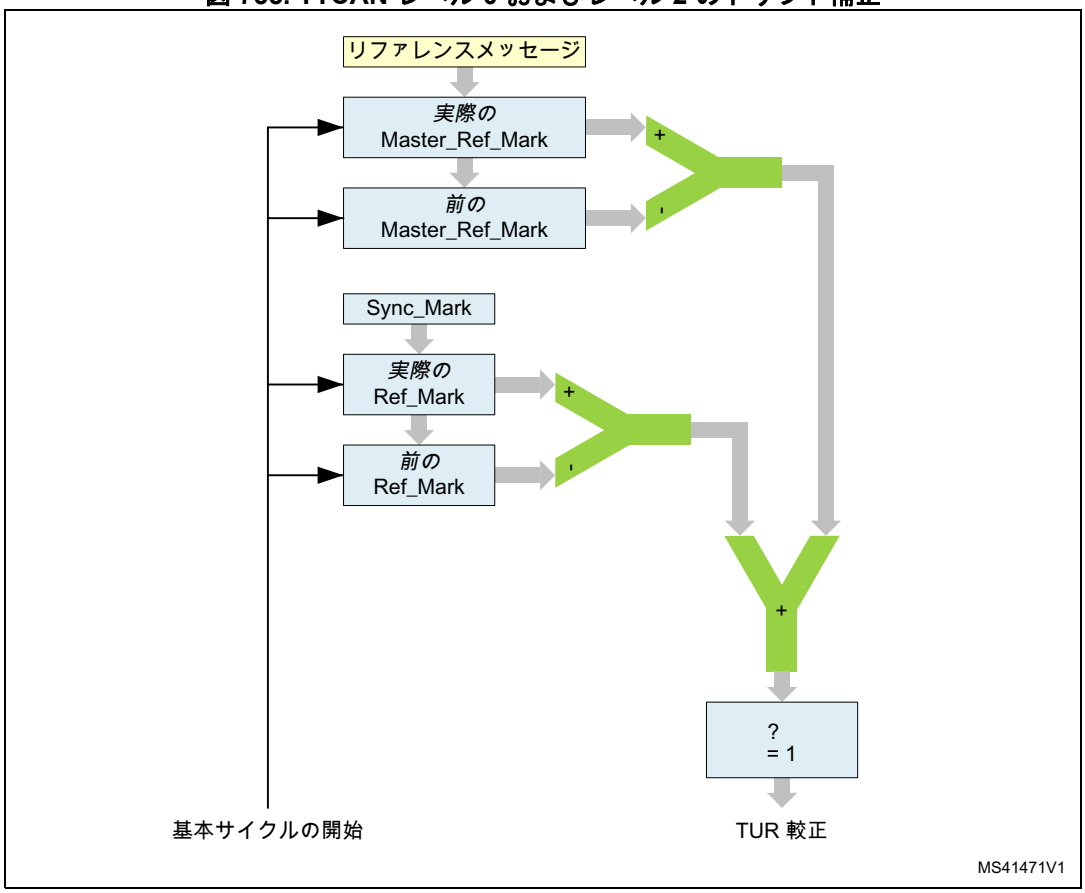


図 735 では、TTCAN レベル 0 および 2 において、各タイム受信ノードがローカルタイムとグローバルタイムの基本サイクルの長さを比較して、ローカルクロックとタイムマスタクロックのドリフトを補正する方法を示しています。2 つの値に差があり、リファレンスメッセージで Disc_Bit がセットされていない場合、TURNA[NAV] の新しい値が計算されます。同期偏差 SD = | NC - TURNA[NAV] | ≤ SDL（同期偏差制限）の場合、TURNA[NAV] の新しい値は有効になります。そうしないと、自動ドリフト補正はサスペンドされます。

TTCAN レベル 0 およびレベル 2 では、TTOST[QCS] は自動ドリフト補正がアクティブであるかサスペンドされているかを示します。TTCAN レベル 1 では、TTOST[QCS] は常に ‘1’ です。

現在のタイムマスタはローカルクロック速度とグローバルタイムフェーズを外部クロックソースに同期できます。これは、ビット TTOCF[EECS] によって有効にできます。

ストップウォッチ機能（*セクション 56.3.9 : ストップウォッチ*を参照）を使用して、ローカルクロックと外部クロックのクロック速度の差を測定できます。ローカルクロック速度は、最初に新しく計算されたNumerator設定ローを TURCF[NCL] に書き込むことで調整されます（動作中 TURCF[DC] は更新できません）。新しい値は、TTOCN[ECS] を ‘1’ と書き込むことで有効になります。

グローバルタイムフェーズは、最初にフェーズオフセットを TT グローバルタイムプリセットレジスタ TTGTP に書き込むことで調整されます。新しい値は、TTOCN[SGT] を ‘1’ と書き込むことで有効

になります。グローバルタイムフェーズの調整後に送信される最初のリファレンスメッセージでは、Disc_Bit が ‘1’ にセットされます。

TTOST[QGTP] は、ノードのグローバルタイムがタイムマスタのグローバルタイムと同調しているかどうかを示します。TTOST[QGTP] は、TTCAN レベル 1 の場合と、TTCAN レベル 0、2（TTOST[QCS] = ‘0’）で同期偏差制限を超えている場合では必ず ‘0’ となります。グローバルタイムがローパスフィルタで、連続した単調な値をアプリケーションに供給している間は、一時的に ‘0’ になります。最後のリファレンスメッセージが Disc_Bit = ‘1’ を含んでいる場合や TTOST[QCS]=‘0’ の場合、ローパスフィルタはありません。

56.3.11 TTCAN のエラーレベル

ISO 11898-4 では 4 つのレベルのエラー深刻度が指定されています。

- S0 - エラーなし。
- S1 - 警告 - アプリケーションの通知のみ、アプリケーション別に対応します。
- S2 エラー - アプリケーションの通知。排他的またはアービトレーションタイムウィンドウでのすべての送信が無効になります（例：データやリモートフレームを開始できません）。潜在的なタイムマスタは、リファレンストリガオフセットを最大値の 127 にしてリファレンスメッセージをまだ送信できます。
- S3 - 深刻なエラー。
- アプリケーションの通知。すべての CAN バス動作が停止します。例：ドミナントビットの送信が許可されず、CCCR[MON] がセットされます。S3 エラー状態は、アプリケーションが設定を更新するまでアクティブなままとなります（CCCR[CCE] をセットします）。

複数のエラーが同時に検出された場合、深刻度が最高のものが優先されます。エラーが検出されると、アプリケーションは TTIR[ELC] によって通知されます。エラーレベルは、TTOST[EL] によって監視されます。

TTCAN は次のエラー状態を ISO 11898-4 での必須事項として信号送信します。

- Config_Error（S3）。
- 統合アービトレーションタイムウィンドウが適切に閉じられなかった場合や、Tx_Ref_Trigger を超えるタイムマークのある Tx_Trigger がある場合、エラーレベル TTOST[EL] を ‘11’ にセットします。
- Watch_Trigger_Reached（S3）。
- リファレンスメッセージが不足しているため、ウォッチトリガに達した場合、エラーレベル TTOST[EL] を ‘11’ にセットします。
- Application_Watchdog（S3）。
- アプリケーションがアプリケーションウォッチドッグを機能させるのに失敗した場合、エラーレベル TTOST[EL] を ‘11’ にセットします。アプリケーションウォッチドッグは TTOCF[AWL] 経由で設定されます。これは、レジスタ TTOST を読み出すことで機能します。ウォッチドッグが時間内に機能しないとき、ビット TTOST[AWE] と割り込みフラグ TTIR[AW] がセットされ、すべての FDCAN 通信が停止し、FDCAN はバスモニタモードにセットされます（CCCR[MON] が ‘1’ にセットされます）。
- CAN_Bus_Off（S3）。
- CAN_Bus_Off 状態に移行すると、エラーレベル TTOST[EL] が ‘11’ にセットされます。CAN_Bus_Off 状態は、PSR[BO] = ‘1’ および CCCR[INIT] = ‘1’ によって示されます。
- Scheduling_Error_2（S2）。

- 1 つの Tx_Trigger の MSC が 7 に達した場合、エラーレベル TTOST[EL] を ‘10’ にセットします。さらに、割り込みフラグ TTIR[SE2] がセットされます。先のマトリックスサイクルで 7 の MSC を持つ Tx_Trigger がない場合、マトリックスサイクルの最初にエラーレベル TTOST[EL] が ‘00’ にリセットされます。
- Tx_Overflow（S2）。
- Tx カウントが Tx_Trigger の予測される数 TTMLM[ENTT] 以上になり、Tx_Trigger イベントが発生した場合、エラーレベル TTOST[EL] を ‘10’ にセットします。さらに、割り込みフラグ TTIR[TXO] がセットされます。Tx カウントが新しいマトリックスサイクル開始時に TTMLM[ENTT] 未満となる場合、エラーレベル TTOST[EL] が ‘00’ にリセットされます。
- Scheduling_Error_1（S1）。
- 1 つのマトリックスサイクル内で、（排他的タイムウィンドウの）すべてのトリガメモリ要素での最大 MSC と最小 MSC の差が 2 を超える場合、または排他的 Rx_Trigger の MSC のうち 1 つが 7 に達した場合、エラーレベル TTOST[EL] を ‘01’ にセットします。さらに、割り込みフラグ TTIR[SE1] がセットされます。1 つのマトリックスサイクル内で、これらの状態がいずれも有効ではない場合、エラーレベル TTOST[EL] が ‘00’ にリセットされます。
- Tx_Underflow（S1）。
- Tx カウントが新しいマトリックスサイクル開始時に Tx_Trigger の予測される数 TTMLM[ENTT] 未満となる場合、エラーレベル TTOST[EL] を ‘01’ にセットします。さらに、割り込みフラグ TTIR[TXU] がセットされます。Tx カウントが新しいマトリックスサイクル開始時に少なくとも TTMLM[ENTT] となる場合、エラーレベル TTOST[EL] が ‘00’ にリセットされます。

56.3.12 TTCAN のメッセージ処理

リファレンスメッセージ

潜在的なタイムマスタの場合、リファレンスメッセージの識別子は TTRMC[RID] 経由で設定されます。リファレンスメッセージの送信に専用 Tx バッファは必要ありません。リファレンスメッセージが送信されるとき、TTCAN レベル 1 の最初のデータバイト（TTCAN レベル 0 では最初の 4 つのデータバイト、TTCAN レベル 2 では最初の 4 つのデータバイト）が FSE によって提供されます。

リファレンスメッセージペイロード選択 TTRMC[RMPS] がセットされている場合、リファレンスメッセージペイロードの残り（レベル 1：バイト 2～8、レベル 0、2：バイト 5～6）は Tx バッファ 0 から取得されます。この場合、メッセージバッファ 0 からのデータ長 DLC コードが使用されます。

表 462. リファレンスメッセージとともに送信されるデータバイト数

TTRMC.RMPS	TXBRP.TRP0	レベル 0	レベル 1	レベル 2
0	0	4	1	4
0	1	4	1	4
1	0	4	1	4
1	1	4+MBO	1+MBO	4+MBO

レベル 1 でリファレンスメッセージとともに追加のペイロードを送信するには、レベル 0 の場合は DLC > 1、レベル 2 の場合は DLC > 4 に設定する必要があります。さらに、メッセージバッファ 0 の送信リクエストペンディングビット TXBRP[TRP0] をセットする必要があります（*表 462* を参照）。リファレンスメッセージが開始されたときにビット TXBRP[TRP0] がセットされていない場合、リファレンスメッセージは FSE によって供給されるデータバイトのみで送信されます。

リファレンスメッセージの許可フィルタには、リファレンス識別子 TTRMC[RID] が使用されます。

メッセージの受信

メッセージの受信は、イベント駆動 CAN 通信と同じ方法で 2 つの Rx FIFO 経由で実行されます（*Rx ハンドラ*を参照）。

メッセージステータスカウント MSC は、対応するトリガメモリ要素の一部で、設定中 0 に初期化する必要があります。これは、TTCAN が同期状態 In_Gap または In_Schedule であるときに更新されます。更新はメッセージ Rx_Trigger で発生します。この時点で、この基本サイクルで受信された最新のメッセージがどの許可フィルタ要素に一致したかチェックされます。一致するフィルタ数は、許可フィルタの結果として格納されます。これがこのトリガメモリ要素で定義されたフィルタ数と同じ場合、MSC は 1 ずつデクリメントされます。許可フィルタの結果がこのフィルタ要素で定義されたフィルタ数と同じではない場合や、許可フィルタの結果がクリアされた場合、MSC は 1 ずつインクリメントされます。各 Rx_Trigger 時および各サイクル開始時に、直近の許可フィルタの結果はクリアされます。

Rx_Trigger のタイムマークは、対象のメッセージの受信および許可フィルタが完了したことを確認できる値にセットすることを推奨します。これには、RAM アクセス時間とフィルタリストの順番を考慮する必要があります。Rx_Trigger に使用されるフィルタはフィルタリストの最初に配置することを推奨します。リファレンスメッセージに Rx_Trigger を使用することは推奨されません。

メッセージの送信

タイムトリガメッセージ送信の場合、TTCAN は 32 の専用 Tx バッファを供給します（*送信一時停止*を参照）。FDCAN がタイムトリガ動作に設定された場合（TTOCF[OM] = ‘01’ または ‘10’）、Tx FIFO または Tx キューは使用できません。

トリガメモリの各 Tx_Trigger は、特定のメッセージを含む特定の Tx バッファを指定します。その Tx バッファに基本サイクルまたはマトリックスサイクルで 1 回以上送信されるメッセージが含まれている場合、指定された Tx バッファに 1 つ以上の Tx_Trigger を配置できるでしょう。

アプリケーションプログラムは、データを定期的かつサイクルタイムに同期される時間どおりに更新する必要があります。ホスト CPU は、部分的に更新されたメッセージが送信されないように管理します。これを確認するために、ホストは次の方法で処理を進めなければなりません。

Tx_Trigger_Single / Tx_Trigger_Merged / Tx_Trigger_Arbitration

- TXBTO を読み出すことで前の送信が完了したかどうかをチェックします。
- Tx バッファの設定および／またはペイロードを更新します。
- 追加リクエストを発行して Tx バッファリクエストペンディングビットをセットします。

Tx_Trigger_Continous

- キャンセルリクエストを発行して Tx バッファリクエストペンディングビットをリセットします。
- TXBCF を読み出すことでキャンセルが終了したかどうかをチェックします。
- Tx バッファの設定および／またはペイロードを更新します。
- 追加リクエストを発行して Tx バッファリクエストペンディングビットをセットします。

対応する Tx_Trigger と格納されたメッセージの MSC で、送信成功に関する情報を取得できます。

CAN バスが対応する送信有効化ウィンドウ内でアイドルではなかったり、メッセージが開始され、正常に完了できなかったりしたために、送信を開始できなかった場合、MSC は 1 ずつインクリメントされます。メッセージを正常に送信できなかった場合や、送信が無効になっていた（TTCAN がエラーレベル S2 であった、またはホストがこの特定のメッセージを無効にした）ために送信有効化ウィンドウ内でメッセージを開始できなかった場合に、MSC は 1 ずつデクリメントされます。

Tx バッファは動的に管理できます。たとえば、異なる識別子を持つ複数のメッセージが同じ Tx バッファ要素を共有できます。この場合、ホストは送信リクエストが TXBRP をチェックすることで Tx バッファ要素を再設定するためにペンディングされていないことを確認する必要があります。

ペンディング中の送信リクエストがある Tx バッファを更新する場合、ホストは最初にキャンセルリクエストを発行し、TXBCF を読み出してキャンセルを完了してから更新を開始しなければなりません。

Tx ハンドラは、送信ウィンドウの開始を定義する Tx_Trigger 要素の直前にアクティブになる中間出力バッファに、メッセージ RAM からメッセージを転送します。転送時間中とその後は、送信メッセージを更新できず、TXBRP ビットを変更できません。この転送時間を制御するために、追加トリガ要素を Tx_Trigger の前に配置できます。この例として、他の動作を必要としない Time_Base_Trigger を使用できます。Tx_Trigger と先のトリガのタイムマークの差は、他のモジュールから高い RAM アクセス負荷がある場合でも Tx ハンドラがメッセージ RAM から 4 ワードを読み出せることを保証できる大きさでなければなりません。

排他的タイムウィンドウでの送信

サイクルタイムが Tx_Trigger_Single または Tx_Trigger_Continuous のタイムマークに達すると、タイムトリガされて送信が開始されます。他のノードからのメッセージとのバスでのアービトレーションはありません。送信を試みた結果に従って、MSC が更新されます。Tx_Trigger_Single によって開始された送信が成功した後、それぞれの Tx バッファリクエストペンディングビットはリセットされます。Tx_Trigger_Continuous によって開始された送信が成功した後、それぞれの Tx バッファリクエストペンディングビットはセットされたままです。外乱によって送信に成功しなかった場合、次に Tx_Trigger（のうち 1 つ）がアクティブになったときに繰り返されます。

アービトレーションタイムウィンドウでの送信

サイクルタイムが Tx_Trigger_Arbitration のタイムマークに達すると、タイムトリガされて送信が開始されます。複数のノードが同時に送信を開始できます。この場合、メッセージは他のノードからのメッセージとアービトレーションする必要があります。MSC は更新されません。送信に成功しなかった場合（ロストアービトレーションまたは外乱）、次に Tx_Trigger（のうち 1 つ）がアクティブになったときに繰り返されます。

統合アービトレーションタイムウィンドウでの送信

統合アービトレーションタイムウィンドウの目的は、複数のノードが CAN アービトレーションによって指定された順番に連続送信される制限された数のフレームを送信できるようにすることです。ノードによるバースト送信を目的としていません。ノードはこのタイムウィンドウ内に排他的アクセスがないため、すべてのリクエストされた送信が成功するとは限りません。

ロストアービトレーションのあるメッセージやエラーによって中断されたメッセージは、同じ統合アービトレーションタイムウィンドウ内で再送信できます。対応する送信リクエストペンディングフラグが Tx キャンセルの成功によってリセットされた場合、再送信は開始されません。

単一送信ウィンドウでは、Tx ハンドラはトリガ要素のメッセージ数によって示されたメッセージを送信します。統合アービトレーションタイムウィンドウでは、トリガリストから最大 3 つのメッセージ数を処理できます。トリガリストによって定義されたシーケンスで、送信が試行されます。1 つ目が送信される前に 4 つ目のメッセージのタイムマークに達した場合（つまり、ホストによってキャンセルされた場合）、4 つのリクエストは無視されます。

統合アービトレーションタイムウィンドウ内の送信はタイムトリガではありません。タイムマークの前、またはバスがアイドルではない場合のタイムマークの後にメッセージの送信を開始できます。

統合アービトレーションタイムウィンドウ内の特定のノードによって送信されたメッセージは Tx_Trigger の順に開始されるため、バストラフィックの強制がある場合、低い CAN 優先度のメッセージはより高い優先度の次のメッセージの送信成功を防ぐことができます。これには、トリガリストの設定を考慮する必要があります。Time_Base_Trigger は、連続する Tx_Trigger の間に配置され、対応する Tx バッファのデータの更新が必要になるまでの時間を定義できます。

56.3.13 TTCAN の割り込みとエラー処理

TT 割り込みレジスタ TTIR は 4 つのセグメントから成ります。各割り込みは、TT 割り込み有効レジスタ TTIE の対応するビットによって個別に有効にできます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に '1' を書き込むことによってクリアされます。

最初のセグメントはフラグ CER、AW、WT、IWT から成ります。各フラグは CAN 通信が停止する致命的エラーの状態を示します。IWT を除いて、これらのエラー状態では通信が復旧できるようになる前に FDCAN モジュールを再設定する必要があります。

2 つ目のセグメントはフラグ ELC、SE1、SE2、TXO、TXU、GTE から成ります。各フラグは CAN 通信が妨害されるエラーの状態を示します。これらが CAN バスでの外乱など送信失敗によって発生した場合、FDCAN プロトコル失敗処理によって処理され、アプリケーションプログラムによる介入を必要としません。

3 つ目のセグメントはフラグ GTD、GTW、SWE、TTMI、RTMI から成ります。最初の 2 つのフラグは、アプリケーションプログラムによる対応が必要となるグローバルタイムイベント（レベル 0、2 のみ）によって制御されます。ピン fdcan_swt の立ち上がりエッジによってトリガされるストップウォッチイベントで、内部時間の値がキャプチャされます。トリガタイムマーク割り込みは、特定の Time_Base_Trigger に達したアプリケーションを通知します。レジスタタイムマーク割り込みは、TTOCN[TMC] によって参照される時間（サイクル、ローカル、グローバル）がタイムマーク TTTMK[TM] と等しくなったことを伝えます。これはギャップの終了にも使用できます。

4 つ目のセグメントはフラグ SOG、CSM、SMC、SBC から成ります。これらのフラグによって、アプリケーションプログラムを通信スケジュールに同期する方法を使用できます。

56.3.14 レベル 0

TTCAN レベル 0 は ISO11898-4 の一部ではありません。この動作モードによって、TTCAN レベル 2 で較正されたグローバルタイムベースを維持するハードウェアを ISO11898-1 に従ってイベント駆動 CAN にも使用できるようになります。

レベル 0 動作は、TTOCF[OM] = '11' 経由で設定されます。このモードでは、FDCAN がイベント駆動 CAN 通信で動作し、固定スケジュールはなく、TTOCF[GEN] の設定は無視されます。外部イベント同期動作は、レベル 0 で利用できません。同期されるタイムベースは、リファレンスメッセージの送信によって管理されます。

レベル 0 では、トリガメモリはアクティブではないため、設定する必要はありません。サイクルタイムが TTOCF[IRTO] ´ 0x200 に達すると、タイムマーク割り込みフラグ（TTIR[TTMI]）がセットされます。これにより、ホストはメッセージバッファ 0 の送信リクエストをセットするようにリマインドされます。サイクルタイムが 0xFF00 に達すると、Watch_Trigger 割り込みフラグ（TTIR[WT]）がセットされます。これらの値は安定したクロック較正に十分なマージンを確保するために選択されています。追加の TT エラーチェックはありません。

レジスタタイムマーク割り込み（TTIR[RTMI]）も使用できます。

リファレンスメッセージは、レベル 2 動作に対して設定されます。受信したリファレンスメッセージは、レジスタ TTRMC で設定された識別子によって認識されます。リファレンスメッセージの送信では、メッセージバッファ 0 のみ使用できます。ノードは、ホストがメッセージバッファ 0 の送信リクエストをセットするといつでもリファレンスメッセージを送信します。リファレンストリガオフセットはありません。

レベル 0 動作は、次で設定されます。

- TTRMC
- EVTP、AWL、GEN を除く TTOCF
- ENTT、TXEW を除く TTMLM
- TURCF

レベル 0 動作は、次で制御されます。

- NIG、TMG、FGP、GCS、TTMIE を除く TTOCN
- TTGTP
- TTTMK
- ビット CER、AW、IWT SE2、SE1、TXO、TXU、SOG を除く TTIR（機能なし）
- TTIR 次のビットは機能を変更しました。
  - トリガメモリによって定義されない TTMI - サイクルタイム TTOCF[IRTO] 0x200 で有効化
  - トリガメモリによって定義されない WT - サイクルタイム 0xFF00 で有効化

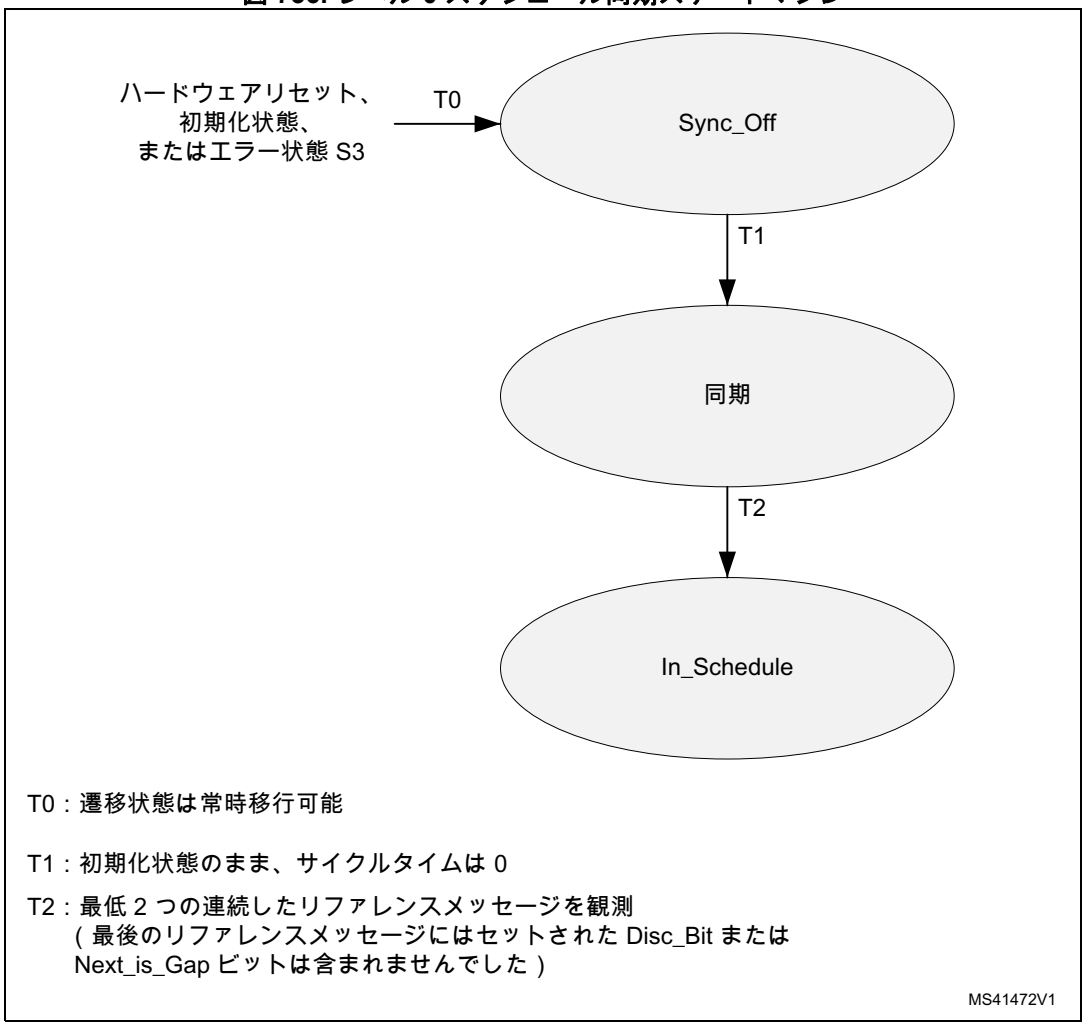
レベル 0 動作は、次で信号送信されます。

- ビット AWE、WFE、GSI、GFI、RTO を除く TTOST（機能なし）

同期

図 736 では、TTCAN レベル 0 動作での状態と状態遷移を示しています。レベル 0 には In_Gap 状態はありません。

図 736. レベル 0 スケジュール同期ステートマシン



エラーレベルの処理

レベル 0 動作中は、次のエラー状態のみ発生します。

- Watch_Trigger_Reached（S3）、サイクルタイム 0xFF00 に到達
- CAN_Bus_Off（S3）

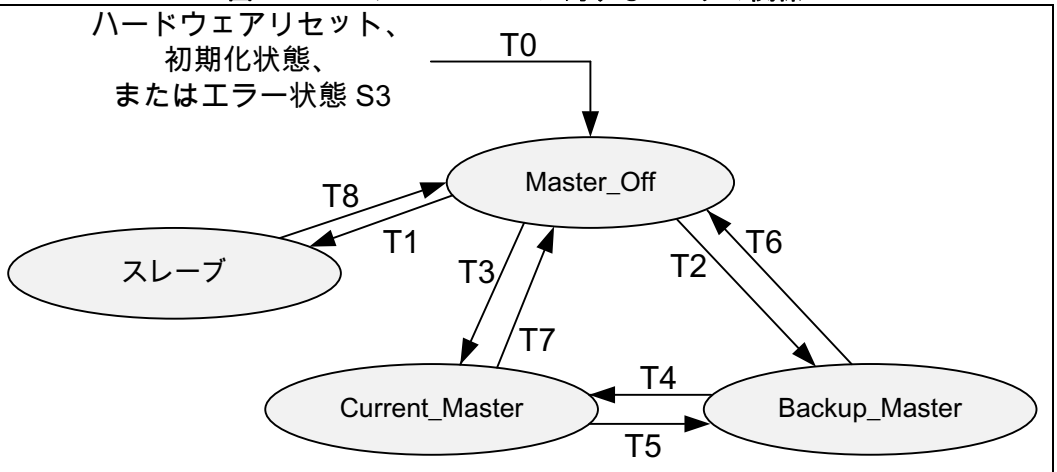
S1 および S2 エラーは発生しないため、エラーレベルを S0（エラーなし）と S3（深刻なエラー）でのみ切り替えられます。TTCAN レベル 0 では、S3 エラーは異なる処理がなされます。エラーレベル S3 に達すると、TTOST[SYS] と TTOST[MS] の両方がリセットされ、割り込みフラグ TTIR[GTE] と TTIR[GTD] がセットされます。

エラーレベル S3（TTOST[EL] = ‘11’）に移行するとき、（TTCAN レベル 1 とレベル 2 とは対照的に）バスモニタモードには移行しません。次のリファレンスメッセージの送信（タイムマスタ）または受信（タイムスレーブ）の後、自動的に S3 エラーレベルから移行します。

マスタスレーブ関係

図 737 では、TTCAN レベル 0 でのマスタスレーブ関係を示しています。S3 エラーが発生すると、FDCAN は状態 Master_Off に戻ります。

図 737. レベル 0 スレーブに対するマスタの関係



T0：遷移状態は常時移行可能

T1：潜在的なタイムマスタではない場合にリファレンスメッセージを観測

T2：自身のマスタ優先度ではないマスタ優先度で観測されるリファレンスメッセージ、エラー状態 ≠ S3

T3：自身のマスタ優先度と同じマスタ優先度で観測されるリファレンスメッセージ、エラー状態 ≠ S3

T4：自身のマスタ優先度で観測されるリファレンスメッセージ

T5：自身のマスタ優先度より高いマスタ優先度で観測されるリファレンスメッセージ

T6：エラー状態 S3

T7：エラー状態 S3

T8：エラー状態 S3

MS41473V1

56.3.15 外部タイムスケジュールへの同期

この機能を使用して、FDCAN スケジュールのフェーズを外部スケジュール（別の TTCAN ネットワークや FlexRay ネットワーク）に同期できます。FDCAN が現在のタイムマスタであるときのみ適用できます（TTOST[MS] = ‘11’）。

外部同期は、イベントトリガ入力ピン fdcan_evt によって制御されます。ビット TTOCN[ESCN] がセットされている場合、イベントトリガピン fdcan_evt の立ち上がりエッジで FDCAN は TTGTP[CTP] によって設定された対象フェーズ値と実際のサイクルタイムを比較します。

TTOCN[ESCN] をセットする前に、ホストは FDCAN ギャップ制御などを使用して 2 つのタイムスケジュールのフェーズを適応しなければなりません（*セクション 56.3.8: TTCAN ギャップ制御*を参照）。ホストが TTOCN[ESCN] をセットすると、TTOST[SPL] がセットされます。

サイクルタイムとイベントトリガピン fdcan_evt の立ち上がりエッジでの対象フェーズ値 TTGTP[CTP] の差が 9 NTU を超える場合、フェーズロックビット TTOST[SPL] はリセットされ、割り込みフラグ TTIR[CSM] がセットされます。他のノードがタイムマスタになった場合、TTOST[SPL] もリセットされます（そして、TTIR[CSM] がセットされます）。

TTOST[SPL] と TTOCN[ESCN] の両方がセットされた場合、およびサイクルタイムとイベントトリガピン fdcan_evt の立ち上がりエッジでの対象フェーズ値 TTGTP[CTP] の差が 9 NTU 以下となる場合、測定された差はリファレンストリガオフセット値として使用され、次に送信されるリファレンスメッセージでフェーズを調整します。

注： ***イベントトリガピン fdcan_evt での立ち上がりエッジ検出は、各基本サイクルの開始とともに有効になります。最初の立ち上がりエッジによって、実際のサイクルタイムと TTGTP[CTP] との比較がトリガされます。次の基本サイクルの開始までの追加のエッジはすべて無視されます。***

56.3.16 FDCAN の Rx バッファおよび FIFO 要素

最大 64 の Rx バッファと 2 つの Rx FIFO をメッセージ RAM に設定できます。各 Rx FIFO セクションを設定して、最大 64 個の受信メッセージを格納できます。Rx バッファ／FIFO 要素の構造は、*表 463* に示され、*表 464* に説明が示されています。

表 463. Rx バッファおよび FIFO 要素

ビット	31 24	23 16	15 8	7 0
R0	ESI / XTD / RTR / ID[28:0]			
R1	ANMF / FIDX[6:0]	Res. / FDF / BRS / DLC[3:0]	RXTS[15:0]	
R2	DB3[7:0]	DB2[7:0]	DB1[7:0]	DB0[7:0]
R3	DB7[7:0]	DB6[7:0]	DB5[7:0]	DB4[7:0]
⋮	⋮	⋮	⋮	
Rn	DBm[7:0]	DBm-1[7:0]	DBm-2[7:0]	DBm-3[7:0]

要素サイズは、最大 64 バイトデータフィールドがある CAN FD メッセージの格納についてレジスタ RXESC 経由で設定できます。

表 464. Rx バッファおよび FIFO 要素の説明

フィールド	説明
R0 ビット 31 ESI	エラー状態インジケータ 0：送信中のノードはエラーアクティブです。 1：送信中のノードはエラーパッシブです。
R0 ビット 30 XTD	拡張識別子 受信フレームに標準識別子または拡張識別子があるかどうかをホストに信号送信します。 0：11 ビット標準識別子です。 1：29 ビット拡張識別子です。
R0 ビット 29 RTR	リモート送信リクエスト 受信フレームがデータフレームかリモートフレームかどうかをホストに信号送信します。 0：受信フレームはデータフレームです。 1：受信フレームはリモートフレームです。
R0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に保存されます。
R1 ビット 31 ANMF	許可される一致しないフレーム 一致しないフレームの許可は、GFC[ANFS] および GFC[ANFE] 経由で有効にできます。 0：受信フレームがフィルタインデックスに一致します。 1：受信フレームがどの Rx フィルタ要素にも一致しません。
R1 ビット 30:24 FIDX[6:0]	フィルタインデックス 0-127 = 一致している Rx 許可フィルタ要素のインデックス（ANMF = ‘1’ の場合は無効）。 範囲は、0～SIDFC[LSS] - 1 または XIDFC[LSE] - 1 です。

表 464. Rx バッファおよび FIFO 要素の説明（続き）

フィールド	説明
R1 ビット 21 FDF	FD フォーマット 0：標準フレームフォーマットです。 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC）です。
R1 ビット 20 BRS	ビットレート切り替え 0：ビットレート切り替えなしで受信したフレームです。 1：ビットレート切り替えありで受信したフレームです。
R1 ビット 19:16 DLC[3:0]	データ長コード 0～8：CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 9～15：CAN + CAN FD：受信フレームに 8 のデータバイトがあります。 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
R1 ビット 15:0 RXTS[15:0]	Rx タイムスタンプ フレーム受信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ TSCC[TCP] の設定に依存します。
R2 ビット 31:24 DB3[7:0]	データバイト 3
R2 ビット 23:16 DB2[7:0]	データバイト 2
R2 ビット 15:8 DB1[7:0]	データバイト 1
R2 ビット 7:0 DB0[7:0]	データバイト 0
R3 ビット 31:24 DB7[7:0]	データバイト 7
R3 ビット 23:16 DB6[7:0]	データバイト 6
R3 ビット 15:8 DB5[7:0]	データバイト 5
R3 ビット 7:0 DB4[7:0]	データバイト 4
⋮	⋮
Rn ビット 31:24 DBm[7:0]	データバイト m
Rn ビット 23:16 DBm-1[7:0]	データバイト m-1

表 464. Rx バッファおよび FIFO 要素の説明（続き）

フィールド	説明
Rn ビット 15:8 DBm-2[7:0]	データバイト m-2
Rn ビット 7:0 DBm-3[7:0]	データバイト m-3

56.3.17 FDCAN の Tx バッファ要素

Tx バッファセクションを設定して、専用 Tx バッファと Tx FIFO / Tx キューを保持できます。Tx バッファセクションが専用 Tx バッファと" (2バイト文字の変更) Tx FIFO / Tx キューによって共有される場合、専用 Tx バッファは Tx FIFO または Tx キューに割り当てられるバッファのに続く Tx バッファセクションの最初から開始します。Tx ハンドラは Tx バッファ設定 TXBC.TFQS および TXBC.NDTB を評価することで、専用 Tx バッファと Tx FIFO / Tx キューを区別します。要素サイズは、最大 64 バイトデータフィールドがある CAN FD メッセージの格納についてレジスタ TXESC 経由で設定できます。

表 465. Tx バッファおよび FIFO 要素

ビット	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T0	ESI	XTD	RTR	ID[28:0]																												
T1	MM[7:0]								EFC	Res.	FDF	BPS	DLC[3:0]				Res.															
T2	DB3[7:0]								DB2[7:0]								DB1[7:0]								DB0[7:0]							
T3	DB7[7:0]								DB6[7:0]								DB5[7:0]								DB4[7:0]							
⋮	⋮								⋮								⋮															
Tn	DBm[7:0]								DBm-1[7:0]								DBm-2[7:0]								DBm-3[7:0]							

表 466. Tx バッファ要素の説明

フィールド	説明
T0 ビット 31 ESI(1)	エラー状態インジケータ 0：CAN FD フォーマットの ESI ビットはエラーパッシブフラグにのみ依存します。 1：CAN FD フォーマットの ESI ビットはリセッシブで送信されました。
T0 ビット 30 XTD	拡張識別子 0：11 ビット標準識別子です。 1：29 ビット拡張識別子です。
T0 ビット 29 RTR(2)	リモート送信リクエスト 0：データフレームを送信します。 1：リモートフレームを送信します。
T0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。

表 466. Tx バッファ要素の説明（続き）

フィールド	説明
T1 ビット 31:24 MM[7:0]	メッセージマーカー Tx バッファ設定中に CPU によって書き込まれます。Tx メッセージステータスの識別のために Tx イベント FIFO 要素にコピーされます。
T1 ビット 23 EFC	イベント FIFO 制御 0：Tx イベントを保存しません。 1：Tx イベントを保存します。
T1 ビット 21 FDF	FD フォーマット 0：クラシック CAN フォーマットでフレームが送信されます。 1：CAN FD フォーマットでフレームが送信されます。
T1 ビット 20 BRS(3)	ビットレート切り替え 0：ビットレート切り替えなしで送信された CAN FD フレームです。 1：ビットレート切り替えありで送信された CAN FD フレームです。
T1 ビット 19:16 DLC[3:0]	データ長コード 0～8：CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 9～15：CAN + CAN FD：受信フレームに 8 のデータバイトがあります。 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
T2 ビット 31:24 DB3[7:0]	データバイト 3
T2 ビット 23:16 DB2[7:0]	データバイト 2
T2 ビット 15:8 DB1[7:0]	データバイト 1
T2 ビット 7:0 DB0[7:0]	データバイト 0
T3 ビット 31:24 DB7[7:0]	データバイト 7
T3 ビット 23:16 DB6[7:0]	データバイト 6
T3 ビット 15:8 DB5[7:0]	データバイト 5
T3 ビット 7:0 DB4[7:0]	データバイト 4
⋮	⋮
Tn ビット 31:24 DBm[7:0]	データバイト m
Tn ビット 23:16 DBm-1[7:0]	データバイト m-1
Tn ビット 15:8 DBm-2[7:0]	データバイト m-2
Tn ビット 7:0 DBm-3[7:0]	データバイト m-3

1. 送信バッファの ESI ビットは、エラーパッシブフラグで論理和がとられ、送信される FD フレームでの ESI ビットの値を決定します。CAN FD プロトコルの仕様によって必須とされるとおり、エラーアクティブノードではオプションで ESI ビットリセッシブを送信できますが、エラーパッシブノードでは常に ESI ビットリセッシブを送信します。
2. RTR = 1 のとき、CCCR.FDOE が CAN FD フォーマットで送信を有効にしている場合でも、FDCAN は ISO11898-1 に従ってリモートフレームを送信します。
3. ビット ESI、FDF、BRS は CAN FD 動作が有効（CCCR.FDOE = 1）のときのみ評価されます。ビット BRS はさらに CCCR.BRSE = '1' のときのみ評価されます。

56.3.18 FDCAN Tx イベント FIFO 要素

各要素には、送信されたメッセージに関する情報が保存されています。Tx イベント FIFO を読み出すことで、ホスト CPU はメッセージが送信された順にこの情報を取得します。Tx イベント FIFO に関するステータス情報は、レジスタ TXEFS から取得することができます。

表 467. Tx イベント FIFO 要素

ビット	31			24	23			16	15	8	7	0
E0	ESI	XTD	RTR	ID[28:0]								
E1	MM[7:0]				ET[1:0]	EDL	BRS	DLC[3:0]	TXTS[15:0]			

表 468. Tx イベント FIFO 要素の説明

フィールド	説明
E0 ビット 31 ESI	エラー状態インジケータ 0：送信中のノードはエラーアクティブです。 1：送信中のノードはエラーパッシブです。
E0 ビット 30 XTD	拡張識別子 0：11 ビット標準識別子です。 1：29 ビット拡張識別子です。
E0 ビット 29 RTR	リモート送信リクエスト 0：データフレームを送信します。 1：リモートフレームを送信します。
E0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
E1 ビット 31:24 MM[7:0]	メッセージマーカー Tx メッセージステータスの識別のために Tx バッファから Tx イベント FIFO 要素にコピーされます。
E1 ビット 23:22 EFC	イベントタイプ 00：予約済み 01：Tx イベント 10：キャンセル時でも送信（DAR モードでの送信では常にセット） 11：予約済み
E1 ビット 21 EDL	拡張データ長 0：標準フレームフォーマットです。 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC））

表 468. Tx イベント FIFO 要素の説明（続き）

フィールド	説明
E1 ビット 20 BRS	ビットレート切り替え 0：ビットレート切り替えなしで送信されたフレームです。 1：ビットレート切り替えありで送信されたフレームです。
T1 ビット 19:16 DLC[3:0]	データ長コード 0～8：0～8 のデータバイトのあるフレームが送信されます。 9～15：8データバイトのあるフレームが送信されます。
E1 ビット 15:0 TXTS[15:0]	Tx タイムスタンプ フレーム送信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ TSCC[TCP] の設定に依存します。

56.3.19 FDCAN 標準メッセージ ID フィルタ要素

最大 128 のフィルタ要素を 11 ビット標準 ID に設定できます。標準メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの標準開始アドレス SIDFC.FLSSA とフィルタ要素のインデックス（0～127）です。

表 469. 標準メッセージ ID フィルタ要素

ビット	31 24		23 16	15 8	7 0
S0	SFT[1:0]	SFEC[2:0]	SFID1[10:0]	Res.	SFID2[10:0]

表 470. 標準メッセージ ID フィルタ要素フィールドの説明

フィールド	説明
ビット 31:30 SFT[1:0](1)	標準フィルタタイプ 00：SFID1 から SFID2 までの範囲のフィルタ 01：SFID1 または SFID2 のデュアル ID フィルタ 10：クラシックフィルタ：SFID1 = フィルタ、SFID2 = マスク 11：フィルタ要素無効化
ビット 29:27 SFEC[2:0]	標準フィルタ要素の設定 すべての有効化されたフィルタ要素は、標準フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。SFEC = “100”、“101”、または “110” の場合、一致すると割り込みフラグ IR.HPM がセットされ、有効であれば、割り込みが生成されます。この場合、レジスタ HPMS は優先度一致のステータスで更新されます。 000：フィルタ要素の無効化 001：フィルタが一致した場合に Rx FIFO 0 に格納 010：フィルタが一致した場合に Rx FIFO 1 に格納 011：フィルタが一致した場合に ID を拒否 100：フィルタが一致した場合に優先度をセット 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 111：Rx バッファへの格納またはデバッグメッセージとして格納、SFT[1:0] の設定を無視

表 470. 標準メッセージ ID フィルタ要素フィールドの説明（続き）

フィールド		説明
ビット 26:16 SFID1[10:0]		標準フィルタ ID 1 標準 ID フィルタ要素の最初の ID Rx バッファまたはデバッグメッセージのフィルタ時、このフィールドで格納する標準メッセージの ID を定義します。受信した識別子は完全一致とし、マスキングメカニズムは使用されません。
ビット 15:0	SFID2[15:10]	標準フィルタ ID 2 このビットフィールドは、SFEC の設定に応じて異なる意味を持ちます。 – SFEC = ‘001’～‘110’ では、標準 ID フィルタ要素の 2 つ目の ID – SFEC = ‘111’ では、Rx バッファまたはデバッグメッセージのフィルタ
	SFID2[10:9]	受信メッセージは Rx バッファに格納されるか、デバッグメッセージシーケンスのメッセージ A、B、C として処理されます。 00：Rx バッファにメッセージを格納 01：デバッグメッセージ A 10：デバッグメッセージ B 11：デバッグメッセージ C
	SFID2[8:6]	拡張インタフェースでフィルタイベントピンを制御するために使用されます。それぞれのビット位置で ‘1’ になると、フィルタが一致する場合に関連のフィルタイベントピンで 1 m_ttcan_hclk の周期でパルスを生成できます。 SFID2[8] は較正ユニットによって使用されます。
	SFID2[5:0]	一致するメッセージの格納に Rx バッファの開始アドレス RXBC.RBSA にオフセットを定義します。

1. SFT = “11” で、フィルタ要素は無効化され、許可フィルタは続行します（SFEC = “000” と同じ動作）。

注： ***予約済みの値が設定されている場合、フィルタ要素は無効とみなされます。***

56.3.20 FDCAN 拡張メッセージ ID フィルタ要素

最大 64 のフィルタ要素を 29 ビット拡張 ID に設定できます。拡張メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの拡張開始アドレス XIDFC[FLESA] に要素のインデックス（0～63）の2倍を加えたものです。

表 471. 拡張メッセージ ID フィルタ要素

ビット	31		24	23 16	15 8	7 0
F0	EFEC[2:0]		EFID1[28:0]			
F1	EFTI[1:0]	Res.	EFID2[28:0]			

表 472. 拡張メッセージ ID フィルタ要素フィールドの説明

フィールド		説明
F0 ビット 31:29 EFEC[2:0]		拡張フィルタ要素の設定 すべての有効化されたフィルタ要素は、拡張フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。EFEC = “100”、“101”、または “110” の場合、一致すると割り込みフラグ IR[HPM] がセットされ、有効であれば、割り込みが生成されます。この場合、レジスタ HPMS は優先度一致のステータスで更新されます。 000：フィルタ要素の無効化 001：フィルタが一致した場合に Rx FIFO 0 に格納 010：フィルタが一致した場合に Rx FIFO 1 に格納 011：フィルタが一致した場合に ID を拒否 100：フィルタが一致した場合に優先度をセット 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 111：Rx バッファへの格納、EFT[1:0] の設定を無視
F0 ビット 28:0 EFID1[28:0]		拡張フィルタ ID 1 拡張 ID フィルタ要素の最初の ID Rx バッファまたはデバッグメッセージのフィルタ時、このフィールドで格納する拡張メッセージの ID を定義します。受信した識別子は完全一致とし、XIDAM マスキングメカニズムのみです。
F1 ビット 31:30 EFT[1:0]		拡張フィルタタイプ 00：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID） 01：EF1ID または EF2ID のデュアル ID フィルタ 10：クラシックフィルタ：EF1ID = フィルタ、EF2ID = マスク 11：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID）、XIDAM 適用なし
F1 ビット 28:0	EFID2[10:0]	拡張フィルタ ID 2 このビットフィールドは、EFEC の設定に応じて異なる意味を持ちます。 – SFEC = ‘001’～‘110’ では、拡張 ID フィルタ要素の 2 つ目の ID – SFEC = ‘111’ では、Rx バッファまたはデバッグメッセージのフィルタ
	EFID2[10:9]	受信メッセージは Rx バッファに格納されるか、デバッグメッセージシーケンスのメッセージ A、B、C として処理されます。 00：Rx バッファにメッセージを格納 01：デバッグメッセージ A 10：デバッグメッセージ B 11：デバッグメッセージ C
	EFID2[8:6]	拡張インタフェースでフィルタイベントピンを制御するために使用されます。それぞれのビット位置で ‘1’ になると、フィルタが一致する場合に関連のフィルタイベントピンで 1 m_ttcan_hclk の周期でパルスを生成できます。 EFID2[8] インタフェースは較正ユニットによって使用されます。
	EFID2[5:0]	一致するメッセージの格納に Rx バッファの開始アドレス RXBC.RBSA にオフセットを定義します。

56.3.21 FDCAN のトリガメモリ要素

最大 64 のトリガメモリ要素を設定できます。トリガメモリ要素にアクセスするとき、そのアドレスはトリガメモリの開始アドレス TTTMC[TMSA] にトリガメモリ要素のインデックス（0〜63）を加えたものです。

表 473. トリガメモリ要素

ビット	31 24	23 16	15 8	7 0
T0	TM[15:0]		Res. CC[6:0]	Res. TMIN TMEX TYPE[3:0]
T1	Res.	FTYPE MNR[6:0]	Res.	MSC[2:0]

表 474. トリガメモリ要素の説明

フィールド	説明
T0 ビット 31:16 TM[15:0]	タイムマーク トリガがアクティブになるサイクルタイムです。
T0 ビット 14:8 CC[6:0]	サイクルコード トリガを有効であるサイクルカウントです。トリガタイプ Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、Watch_Trigger_Gap、End_of_List では無視されます。 0b000000x：すべてのサイクルで有効 0b000001c：サイクルカウント mod2 = c で第 2 サイクルごとに有効 0b00001cc：サイクルカウント mod4 = cc で第 4 サイクルごとに有効 0b0001ccc：サイクルカウント mod8 = ccc で第 8 サイクルごとに有効 0b001cccc：サイクルカウント mod16 = cccc で第 16 サイクルごとに有効 0b01ccccc：サイクルカウント mod32 = ccccc で第 32 サイクルごとに有効 0b1cccccc：サイクルカウント mod64 = cccccc で第 64 サイクルごとに有効
T0 ビット 5 TMIN	タイムマークイベント内部 0：影響なし。 1：トリガメモリ要素がアクティブである場合、TTIR.TTMI がセットされます。
T0 ビット 4 TMEX	タイムマークイベント外部 0：影響なし。 1：トリガメモリ要素のタイムマークがアクティブで、TTOCN.TTMIE = ‘1’ の場合、1 周期の長さで出力 m_ttcan_tmp のパルスが生成されます。
T0 ビット 3:0 TYPE[3:0]	トリガタイプ 0000 Tx_Ref_Trigger - ギャップではない場合に有効 0001 Tx_Ref_Trigger_Gap - ギャップの場合に有効 0010 Tx_Trigger_Single - 排他的タイムウィンドウでの単一送信を開始 0011 Tx_Trigger_Continuous - 排他的タイムウィンドウでの連続送信を開始 0100 Tx_Trigger_Arbitration - アービトレーションタイムウィンドウでの送信を開始 0101 Tx_Trigger_Merged - 統合アービトレーションウィンドウを開始 0110 Watch_Trigger - ギャップではない場合に有効 0111 Watch_Trigger_Gap - ギャップの場合に有効 1000 Rx_Trigger - 受信のチェック 1001 Time_Base_Trigger - TMIN、TMEX のみ制御 1010〜1111=End_of_List - 不正なタイプ、設定エラーが発生

表 474. トリガメモリ要素の説明（続き）

フィールド	説明
T1 ビット 23FTYPE	フィルタタイプ 0：11 ビット標準メッセージ ID です。 1：29 ビット拡張メッセージ ID です。
T1 ビット 22:16 MNR[6:0](1)	メッセージ数 – 送信：トリガは設定された Tx バッファ数に対して有効です。有効な値は 0～31 です。 – 受信：トリガは、標準／拡張メッセージ ID フィルタ要素の数に対して有効です。有効な値はそれぞれ 0～63 および 0～127 です。
T1 ビット 2:0 MSC[2:0]	メッセージステータスカウント 排他的タイムウィンドウで周期的メッセージのスケジュールエラーをカウントします。アービトレーションメッセージおよびイベント駆動 CAN 通信では機能がありません（ISO11898-1）。 0-7= 実際のステータス

1. FDCAN が INIT 状態の場合、トリガメモリ要素を書き込む必要があります。INIT 状態以外でのトリガメモリ要素への書き込みアクセスは認められません。タイプ Tx_Ref_Trigger のトリガメモリ要素の一部として定義されている場合、TMIN と TMEX は例外となります。この場合、これらは実際のリファレンストリガオフセット（TTOST[RTO]）によって変更されたタイムマークでアクティブになります。

56.4 FDCAN レジスタ

56.4.1 FDCAN コア開放レジスタ（FDCAN_CREL）

アドレスオフセット：0x0000

リセット値：0xrrd dddd

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r												d			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
d															

ビット 31:28 **REL**：コア開放
1 桁、BCD

ビット 27:24 **STEP**：コア開放のステップ
1 桁、BCD

ビット 23:20 **SUBSTEP**：コア開放のサブステップ
1 桁、BCD

ビット 19:16 **YEAR**：タイムスタンプの年
1 桁、BCD

ビット 15:8 **MON**：タイムスタンプの月
2 桁、BCD

ビット 7:0 **DAY**：タイムスタンプの日
2 桁、BCD

56.4.2 FDCAN コア開放レジスタ（FDCAN_ENDN）

アドレスオフセット：0x0004

リセット値：0x8765_4321

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ETV[31:0]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETV[15:0]															
r															

ビット 31:0 **ETV**：エンディアンテスト値
エンディアンテスト値は 0x8765 4321 です。

56.4.3 FDCAN データビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP）

アドレスオフセット：0x000C

リセット値：0x0000 0A33

このレジスタは、CCCR.CCE および CCCR.INIT がセットされている場合のみ書き込みできます。CAN ビット時間は、4～25 タイムクォンタムの範囲でプログラム可能です。CAN タイムクォンタムは、1～1024 FDCAN 周期の範囲でプログラム可能です。tq = (DBRP + 1) FDCAN クロック周期。

DTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。DTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さは（プログラムされた値）[DTSEG1 + DTSEG2 + 3] tq または（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]				
r	r	r	r	r	r	r	r	r	r	r	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DTSEG1[4:0]					DTSEG2[3:0]				DSJW[3:0]			
r	r	r	rw												

ビット 31:24 予約済み

ビット 23 **TDC**：トランシーバ遅延補正

0：トランシーバ遅延補正は無効です。

1：トランシーバ遅延補正は有効です。

ビット 22:21 予約済み

ビット 20:16 **DBRP**：データビットレートプリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。ボーレートプリスケーラに有効な値は 0～1023 です。ハードウェアはこの値をプログラムされた値 +1 として解釈します。

ビット 15:13 予約済み

ビット 12:8 **DTSEG1**：サンプルポイント前のデータ時間セグメント

有効な値は 1～15 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

ビット 7:4 **DTSEG1**：サンプルポイント後のデータ時間セグメント

有効な値は 1～7 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

ビット 3:0 **DSJW**：同期ジャンプ幅

有効な値は 0～15 です。この値のハードウェアによる実際の解釈は、ここでプログラムされた値より 1 つ大きな値が使用されます。

注： ***8 MHz の FDCAN クロックで、0x00000A33 のリセット値が FDCAN を 500 kbit/s の高速ビットレートに設定します。***

56.4.4 FDCAN テストレジスタ（FDCAN_TEST）

ビット CCCR[TEST] を ‘1’ にセットして、テストレジスタへの書き込みアクセスを有効にしなければなりません。CCCR[TEST] がリセットされると、すべてのテストレジスタ機能がリセット値にセットされます。

Tx ピン FDCANx_TX のループバックモードおよびソフトウェア制御はハードウェアテストモードです。TX を ‘00’ 以外にプログラムすると、CAN バスでのメッセージ転送を妨害します。

アドレス：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX[1:0]		LBCK	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	rw	rw	rw	rw	r	r	r	r

ビット 31:8 予約済み

ビット 7 **RX**：受信ピン
送信ピン FDCANx_RX の実際の値を監視します。
0：CAN バスはドミナントです（FDCANx_RX = ‘0’）。
1：CAN バスはリセッシブです（FDCANx_RX = ‘1’）。

ビット 6:5 **TX**：送信ピンの制御
00：リセット値、FDCANx_TX TX は CAN コアによって制御され、CAN ビット時間の最後に更新されます。
01：サンプルポイントはピン FDCANx_TX で監視できます。
10：ピン FDCANx_TX でドミナント（‘0’）レベルです。
11：ピン FDCANx_TX でリセッシブ（‘1’）です。

ビット 4 **LBCK**：ループバックモード
0：リセット値、ループバックモードは無効です。
1：ループバックモードは有効です（*テストモード*を参照）。

ビット 3:0 予約済み

56.4.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD）

RAM ウォッチドッグはメッセージ RAM の READY 出力を監視します。メッセージ RAM アクセスは、RWD[WDC] ビットで設定された値でメッセージ RAM ウォッチドッグカウンタを開始します。

メッセージ RAM が READY 出力を有効にすることで完了成功の信号を送信すると、カウンタは RWD[WDC] ビットで再ロードされます。カウンタが 0 までのカウントダウンを完了するまでにメッセージ RAM からレスポンスがない場合、カウンタは停止し、割り込みフラグ IR[WDI] ビットがセットされます。RAM ウォッチドッグカウンタは、fdcan_pclk クロックによってクロック供給されます。

アドレス：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDV[7:0]								WDC[7:0]							
r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み

ビット 15:8 **WDV**：ウォッチドッグの値

実際のメッセージ RAM ウォッチドッグカウンタの値です。

ビット 7:0 **WDC**：ウォッチドッグの設定

メッセージ RAM ウォッチドッグカウンタの開始値です。‘00’ のリセット値で、カウンタは無効になります。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

56.4.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）

アドレス：0x0018

リセット値：0x0000 0001

シングルビットのセットおよびリセットの詳細については、*ソフトウェアの初期化*を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NISO	TXP	EFBI	PHXD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INIT
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み

ビット 15 **NISO**：非 ISO 動作

このビットがセットされた場合、FDCAN は Bosch CAN FD Specification V1.0 によって指定された CAN FD フレームフォーマットを使用します。

0：ISO11898-1 に準拠した CAN FD フレームフォーマットです。

1：Bosch CAN FD Specification V1.0 に準拠した CAN FD フレームフォーマットです。

ビット 14 **TXP**

このビットがセットされた場合、FDCAN はフレームの送信成功後、次の送信を開始する前に 2 CAN ビット時間、一時停止します。

0：無効

1：有効

ビット 13 **EFBI**：バス統合中のエッジフィルタ

0：エッジフィルタは無効です。

1：2 つの連続したドミナント tq は、ハード同期のためにエッジを検出する必要があります。

ビット 12 **PXHD**：プロトコル例外処理の無効化

0：プロトコル例外処理は有効です。

1：プロトコル例外処理は無効です。

ビット 11:10 予約済み

ビット 9 **BSE**：FDCAN ビットレート切り替え

0：送信のビットレート切り替えは無効です。

1：送信のビットレート切り替えは有効です。

ビット 8 **FDOE**：FD 動作の有効化

0：FD 動作は無効です。

1：FD 動作は有効です。

ビット 7 **TEST**：テストモードの有効化

0：通常動作、レジスタ TEST はリセット値を保持します。

1：テストモード、レジスタ TEST への書き込みアクセスは有効です。

ビット 6 **DAR**：自動再送信の無効化

0：正常に送信されなかったメッセージの自動再送信は有効です。

1：自動再送信は無効です。

ビット 5 **MON**：バスモニタモード

ビット MON は、CCE および INIT の両方が ‘1’ にセットされているときのみソフトウェアによってセットできます。このビットはいつでもホストによってリセットできます。

0：バスモニタモードは無効です。

1：バスモニタモードは有効です。

ビット 4 **CSR**：クロック停止リクエスト

0：クロック停止はリクエストされていません。

1：クロック停止がリクエストされました。クロック停止がリクエストされたとき、すべてのペンディング中の

転送リクエストが完了して CAN バスがアイドルに達した後、まず INIT はセットされ、次に CSA がセットされます。

ビット 3 **CSA**：クロック停止の確認応答

0：クロック停止の確認応答はありません。

1：FDCAN は、APB クロックおよびカーネルクロックを停止することで、パワーダウンにセットできます。

ビット 2 **ASM**：ASM 制限動作モード

制限動作モードは、異なる CAN ビットレートに適応するアプリケーションを対象とします。アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。オプションの制限動作モードでは、ノードはデータとリモートフレームを送信および受信でき、有効なフレームに確認応答を付与しますが、アクティブエラーフレームまたはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待って CAN 通信と再同期します。エラーカウンタはインクリメントされません。ビット ASM は、CCE および INIT の両方が ‘1’ にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってリセットできます。

FDCAN が CAN ユニットのクロック較正に接続されている場合、較正が完了しない限り、ASM ビットはハードウェアによってセットされます。

0：通常 CAN 動作です。

1：制限動作モードはアクティブです。

ビット 1 **CCE**：設定変更の有効化

0：CPU による保護された設定レジスタへの書き込みアクセスはありません。

1：CPU による保護された設定レジスタへの書き込みアクセスがあります（CCCR.INIT = ‘1’ のとき）。

ビット 0 **INIT**：初期化

0：通常動作

1：初期化が開始されました。

注： ***2 つのクロックドメインの同期メカニズムによって、INIT に書き込まれた値を読み戻せるようになるまでに遅延が発生する場合があります。そのため、プログラマは INIT を新しい値にセットする前に、INIT を読み出して INIT に書き込まれた前の値が許可されたことを確認する必要があります。***

56.4.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）

アドレス：0x001C

リセット値：0x0000 0A33

このレジスタは、CCCR[CCE] および CCCR[INIT] がセットされている場合のみ書き込みできます。CAN ビット時間は、4～81 tq の範囲でプログラム可能です。CAN タイムクォンタムは、[1～1024] FDCAN カーネルクロック周期の範囲でプログラム可能です。

tq = (BRP + 1) FDCAN クロック周期 m_ttcan_cclk

NTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。NTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さは（プログラムされた値）[NTSEG1 + NTSEG2 + 3] tq または（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSJW[6:0]							NBRP[8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NTSEG1[7:0]								Res.	TSEG2[6:0]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 **NSJW**：公称（再）同期ジャンプ幅

有効な値は 1～127 です。この値のハードウェアによる実際の解釈は、使用される値が 1 でインクリメントされたプログラムされた値です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 24:16 **NBRP**：ビットレートプリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。有効な値は 0～511 です。この値のハードウェアによる実際の解釈は、ここでプログラムされた値より 1 つ大きな値が使用されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 15:8 **NTSEG1**：サンプルポイント前の公称時間セグメント

有効な値は 1～255 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 7 予約済み

ビット 6:0 **TSEG2**：サンプルポイント後の公称時間セグメント

有効な値は 1～127 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

注： ***8 MHz の CAN カーネルクロックで、0x00000A33 のリセット値が FDCAN を 500 kbit/s のビットレートに設定します。***

56.4.8 FDCAN タイムスタンプカウンタ設定レジスタ（FDCAN_TSCC）

アドレス：0x0020

リセット値：0x0000 0000

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

ビット 31:20 予約済み

ビット 19:16 **TCP**：タイムスタンプカウンタプリスケーラ

タイムスタンプとタイムアウトカウンタの時間単位を CAN ビット時間の倍数 [1～16] で設定します。

この値のハードウェアによる実際の解釈は、ここでプログラムされた値より 1 つ大きな値が使用されます。

CAN FD モードでは、アービトレーションフェーズとデータフェーズで異なる CAN ビット時間のため、内部タイムスタンプカウンタ TCP は一定の時間ベースを提供しません。CAN FD では、タイムスタンプ生成用の外部カウンタが必要です（TSS = ‘10’）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 15:2 予約済み

ビット 1:0 **TSS**：サンプルポイント前の公称時間セグメント

有効な値は 1～255 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 7 予約済み

ビット 6:0 **TSS**：タイムスタンプの選択

00：タイムスタンプカウンタ値は常に 0x0000 です。

01：タイムスタンプカウンタ値は TCP に従ってインクリメントされます。

10：TIM3 の値から外部タイムスタンプカウンタが使用されます（tim3_cnt[0:15]）。

11：‘00’ と同様です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

56.4.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV）

アドレス：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSC[15:0]															
w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c

ビット 31:16 予約済み

ビット 15:0 **TSC**：タイムスタンプカウンタ

内部／外部タイムスタンプカウンタ値は、フレーム受信の開始時（Rx および Tx の両方）にキャプチャされます。TSCC[TSS] = ‘01’ の場合、タイムスタンプカウンタは、TSCC[TCP] の設定に応じて CAN ビット時間 [1～16] の倍数でインクリメントされます。ラップアラウンドによって、割り込みフラグ IR[TSW] がセットされます。書き込みアクセスによって、カウンタは 0 にリセットされます。TSCC.TSS = ‘10’ の場合、TSC は外部タイムスタンプカウンタ値を反映します。書き込みアクセスによる影響はありません。

注： ***「ラップアラウンド」とは、TSCV への書き込みアクセス以外で、0 以外から 0 へのタイムスタンプカウンタ値の変化を指します。***

56.4.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）

アドレス：0x0028

リセット値：0xFFFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TOP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS[1:0]		ETOC
													rw	rw	rw

ビット 31:16 **TOP**：タイムアウト期間

タイムアウトカウンタ（ダウンカウンタ）の開始値です。タイムアウト期間を設定します。

ビット 15:3 予約済み

ビット 2:1 **TOS**：タイムアウト選択

連続モードで動作している場合、TOCV への書き込みによって、カウンタは TOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。タイムアウトカウンタが FIFO のうち 1 つで制御されている場合、空の FIFO は TOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。

00：連続動作

01：Tx イベント FIFO によって制御されるタイムアウト

10：Rx FIFO 0 によって制御されるタイムアウト

11：Rx FIFO 1 によって制御されるタイムアウト

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

ビット 0 **ETOC**：タイムアウトカウンタの有効化

0：タイムアウトカウンタは無効です。

1：タイムアウトカウンタは有効です。

これは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ書き込みアクセス可能です。

詳細については、*タイムアウトカウンタ* を参照してください。

56.4.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）

アドレス：0x002C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOC[15:0]															
w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c

ビット 31:16 予約済み

ビット 15:0 **TOC**：タイムアウトカウンタ

タイムアウトカウンタは、TSCC.TCP の設定に応じて CAN ビット時間 [1～16] の倍数でデクリメントされます。0 までデクリメントされると、割り込みフラグ IR.TOO がセットされ、タイムアウトカウンタが停止します。開始およびリセット／再開条件は、TOCC.TOS で設定します。

56.4.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR）

アドレス：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RP	TREC[6:0]							TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 予約済み

ビット 23:16 **CEL**：CAN エラーロギング

CAN プロトコルエラーによって送信エラーカウンタまたは受信エラーカウンタがインクリメントされるたびにカウンタがインクリメントされます。CEL の読み出しアクセスすることによってリセットされます。カウンタは 0xFF で停止し、TEC または REC の次のインクリメントによって割り込みフラグ IR[ELO] がセットされます。

アクセスタイプは RX で、読み出し時にリセットされます。

ビット 15 **RP**：受信エラーパッシブ

0：受信エラーカウンタが 128 のエラーパッシブレベル未満です。

1：受信エラーカウンタが 128 のエラーパッシブレベルに達しています。

ビット 14:8 **TREC：**受信エラーカウンタ

受信エラーカウンタの実際の状態、値の範囲は 0～127 です。

ビット 7:0 **TEC**：送信エラーカウンタ

送信エラーカウンタの実際の状態、値の範囲は 0～255 です。

CCCR.ASM がセットされている場合、CAN プロトコルエラーが検出されているが、CEL がインクリメントされているときに、CAN プロトコルコントローラは TEC と REC をインクリメントしません。

56.4.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）

アドレス：0x0044

リセット値：0x0000 0707

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済み

ビット 22:16 **TDCV**：トランスミッタ遅延補正値

セカンダリサンプルポイントの位置で、m_can_tx から m_can_rx までで測定された遅延と TDCR.TDCO の合計で定義されます。SSP の位置は、データフェーズでは送信されたビットとセカンダリサンプルポイントの間の最小タイムクォンタム（mtq）の数です。有効な値は 0～127 mtq です。

ビット 15 予約済み

ビット 14 **PXE**：プロトコル例外イベント

0：最後の読み出しアクセス以降に発生したプロトコル例外イベントはありません。

1：プロトコル例外イベントが発生しました。

ビット 13 **REDL**：受信 FDCAN メッセージ

このビットは、許可フィルタに関係なくセットされます。

0：このビットが CPU によってリセットされるまでに、受信した FDCAN メッセージはありません。

1：EDL フラグがセットされた FDCAN フォーマットのメッセージを受信しました。

アクセスタイプは RX で、読み出し時にリセットされます。

ビット 12 **RBRS：最後に受信した FDCAN メッセージの BRS フラグ**

このビットは、許可フィルタに関係なく REDL とともにセットされます。

0：最後に受信した FDCAN メッセージには BRS フラグがセットされていません。

1：最後に受信した FDCAN メッセージには BRS フラグがセットされています。

アクセスタイプは RX で、読み出し時にリセットされます。

ビット 11 **RESI**：最後に受信した FDCAN メッセージの ESI フラグ

このビットは、許可フィルタに関係なく REDL とともにセットされます。

0：最後に受信した FDCAN メッセージには ESI フラグがセットされていません。

1：最後に受信した FDCAN メッセージには ESI フラグがセットされています。

アクセスタイプは RX で、読み出し時にリセットされます。

ビット 10:8 **DLEC**：データの最後のエラーコード

BRS フラグをセットした FDCAN フォーマットフレームのデータフェーズで発生した最後のエラーのタイプです。コーディングは LEC と同じです。このフィールドは、BRS フラグをセットした FDCAN フォーマットフレームがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。

アクセスタイプは RS で、読み出し時にセットされます。

ビット 7 **BO**：Bus_Off ステータス

0：FDCAN は、Bus_Off ではありません。

1：FDCAN は、Bus_Off 状態です。

ビット 6 **EW**：警告ステータス

0：両方のエラーカウンタが 96 の Error_Warning 制限未満です。

1：少なくとも 1 つのエラーカウンタが 96 の Error_Warning 制限に達しました。

ビット 5 **EP**：エラーパッシブ

0：FDCAN は、Error_Active 状態です。通常、バス通信に参加して、エラーを検出した場合にアクティブエラーフラグを送信します。

1：FDCAN は、Error_Passive 状態です。

ビット 4:3 **ACT**：動作

モジュールの CAN 通信状態を監視します。

00：同期中：ノードが CAN 通信で同期しています。

01：アイドル：ノードはレシーバでもトランスミッタでもありません。

10：レシーバ：ノードはレシーバとして動作しています。

11：トランスミッタ：ノードはトランスミッタとして動作しています。

ビット 2:0 **LEC**：最後のエラーコード

LEC は、CAN バスで発生した最後のエラーのタイプを示します。このフィールドは、メッセージがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。

000：エラーなし：正常な受信または送信によって LEC がリセットされて以降に発生したエラーはありません。

001：スタッフエラー：シーケンス上で許容されていない6ビット以上の同一レベルの連続ビットが受信メッセージに含まれている

010：フォームエラー：受信フレームの固定フォーマット部分に誤ったフォーマットがあります。

011：確認応答エラー：FDCAN によって送信されたメッセージが他のノードによって確認応答されませんでした。

100：ビット 1 エラー：メッセージ（アービトレーションフィールドを除く）送信中、デバイスがリセッシブレベル（論理値のビット ‘1’）の送信を求めたが、監視されたバスの値がドミナントでした。

101：ビット 0 エラー：メッセージ（または確認応答ビット、アクティブエラーフラグ、オーバーロードフラグ）送信中、デバイスがドミナントレベル（データまたは識別子ビットの論理値 ‘0’）の送信を求めたが、監視されたバスの値がリセッシブでした。Bus_Off リカバリ中、11 のリセッシブビットのシーケンスが監視されるたびに、このステータスがセットされます。これにより、CPU が（バスがドミナントでスタックしていない、または継続的に妨害されていないことを示す）Bus_Off リカバリシーケンスの進行を監視できます。

110：CRC エラー：受信メッセージの CRC チェックサムが正しくありません。着信メッセージの CRC が、受信したデータから計算された CRC と一致しません。

111：変更なし：プロトコルステータスレジスタの読み出しアクセスによって、LEC が ‘7’ に再初期化されます。LEC が値 ‘7’ を示している場合、プロトコルステータスレジスタの最後の CPU 読み出しアクセス以降に CAN バスイベントは検出されませんでした。

アクセスタイプは RS で、読み出し時にセットされます。

注： ***FDCAN フォーマットのフレームが、BRS フラグがセットされたデータフェーズに達したとき、次の CAN イベント（エラーまたは有効なフレーム）は LEC の代わりに FLEC で示されます。FDCAN CRC シーケンスの固定スタッフビットでのエラーは、スタッフエラーではなく、フォームエラーとして示されます。***

注： ***Bus_Off リカバリシーケンス（CAN Specification Rev. 2.0 または ISO11898-1参照）は、CCCR[INIT] のセットやリセットで短縮できません。デバイスが Bus_Off になると、それ自体で CCCR.INIT をセットし、すべてのバス動作を停止します。CCCR[INIT] が CPU によってクリアされると、デバイスは通常動作を再開する前に、129 のバスアイドルの発生（129 × 11 の連続リセッシブビット）を待ちます。Bus_Off リカバリシーケンスの最後に、エラー管理カウンタがリセットされます。CCCR[INIT] リセット後の待機時間中、11 のリセッシブビットのシーケンスは監視され、ビット 0 エラーコードは PSR[LEC] に書き込まれ、CPU は CAN バスがドミナントで詰まっていないかどうかや、継続的に***

妨害されていないかをすぐにチェックし、Bus_Off リカバリシーケンスを監視できます。ECR[REC] はこれらのシーケンスをカウントするために使用されます。

56.4.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）

アドレス：0x0048

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:15 予約済み

ビット 14:8 **TDCO**：トランスミッタ遅延補正オフセット

FDCAN_TX から FDCAN_RX までに測定された遅延とセカンダリサンプルポイントの距離を定義するオフセット値です。有効な値は 0～127 mtq です。

ビット 7 予約済み

ビット 6:0 **TDCF**：トランスミッタ遅延補正フィルタのウィンドウ長

SSP の位置の最小値を定義し、より早い SSP の位置につながる FDCAN_RX のドミナントエッジは、トランスミッタ遅延測定では無視されます。

56.4.15 FDCAN 割り込みレジスタ（FDCAN_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に ‘1’ を書き込むことによってクリアされます。

“0”を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。IE の設定は、割り込みが生成されるかどうかを制御します。ILS の設定は、割り込みが信号送信される割り込みラインを制御します。

アドレス：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	Res.	Res.	DRX	TOO	MRAF	TSW
		rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFL	TEFF	TEFW	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1W	RF1N	RF0L	RF0F	RF0W	RF0N
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済み

ビット 29 **ARA**：予約済みアドレスへのアクセス

0：予約済みアドレスへのアクセスは発生していません。

1：予約済みアドレスへのアクセスが発生しました。

ビット 28 **PED**：データフェーズでのプロトコルエラー（データビット時間が使用されます）

0：データフェーズでのプロトコルエラーはありませんでした。

1：データフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.DLEC）。

ビット 27 **PEA**：アービトレーションフェーズでのプロトコルエラー（公称ビット時間が使用されます）

0：アービトレーションフェーズでのプロトコルエラーはありませんでした。

1：アービトレーションフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.LEC）。

ビット 26 **WDI**：ウォッチドッグ割り込み

0：メッセージ RAM ウォッチドッグイベントは発生していません。

1：不足した READY によるメッセージ RAM ウォッチドッグイベントがありました。

ビット 25 **BO**：Bus_Off ステータス

0：Bus_Off ステータスは変更されません。

1：Bus_Off ステータスは変更されます。

ビット 24 **EW**：警告ステータス

0：Error_Warning ステータスは変更されません。

1：Error_Warning ステータスは変更されます。

ビット 23 **EP**：エラーパッシブ

0：Error_Passive ステータスは変更されません。

1：Error_Passive ステータスは変更されます。

ビット 22 **ELO**：エラーロギングオーバーフロー

0：CAN エラーロギングカウンタはオーバーフローしていません。

1：CAN エラーロギングカウンタのオーバーフローが発生しました。

ビット 21:20 予約済み

ビット 19 **DRX**：専用 Rx バッファに格納されたメッセージ

受信メッセージが専用 Rx バッファに格納されたときに必ずこのフラグがセットされます。

0：Rx バッファは更新されません。

1：少なくとも 1 つの受信メッセージが Rx バッファに格納されました。

ビット 18 **TOO**：タイムアウト発生

0：タイムアウトなし

1：タイムアウト到達

ビット 17 **MRAF**：メッセージ RAM アクセス失敗

このフラグは、Rx ハンドラが次の場合にセットされます。

- 次のメッセージのアービトレーションフィールドを受信するまでに、許可フィルタまたは許可メッセージの格納を完了しなかった場合です。この場合、許可フィルタまたはメッセージ格納はアボートされ、Rx ハンドラが次のメッセージの処理を開始します。
- メッセージをメッセージ RAM に書き込めなかった場合です。この場合、メッセージ格納はアボートされます。

両方の場合に、FIFO Put インデックスは更新されないか、専用 Rx バッファの新規データフラグはセットされません。部分的に格納されたメッセージは、次のメッセージがこの位置に格納される際に上書きされます。

Tx ハンドラがメッセージ RAM から時間内にメッセージを読み出せなかったときもこのフラグがセットされます。この場合、メッセージ送信はアボートされます。Tx ハンドラアクセス失敗時、M_TTCAN は制限動作モードに切り替わります（*制限動作モード*を参照）。制限動作モードを終了するには、ホスト CPU で CCCR.ASM をリセットする必要があります。

0：メッセージ RAM アクセス失敗は発生していません。

1：メッセージ RAM アクセス失敗が発生しました。

ビット 16 **TSW**：タイムスタンプラップアラウンド

0：タイムスタンプカウンタのラップアラウンドはありません。

1：タイムスタンプカウンタのラップアラウンドが発生しました。

ビット 15 **TEFL**：Tx イベント FIFO 要素の喪失

0：Tx イベント FIFO 要素の喪失はありません。

1：Tx イベント FIFO 要素を送信しました。サイズ 0 の Tx イベント FIFO への書き込み試行後にもセットされます。

ビット 14 **TEFF**：Tx イベント FIFO フル

0：Tx イベント FIFO はフルではありません。

1：Tx イベント FIFO はフルです。

ビット 13 **TEFW**：Tx イベント FIFO ウォーターマーク到達

0：Tx イベント FIFO フィルレベルはウォーターマーク未満です。

1：Tx イベント FIFO フィルレベルはウォーターマークに達しています。

ビット 12 **TEFN**：Tx イベント FIFO 新規エントリ

0：Tx イベント FIFO は変更されません。

1：Tx ハンドラが Tx イベント FIFO 要素に書き込みました。

ビット 11 **TFE**：Tx FIFO エンプティ

0：Tx FIFO はエンプティではありません。

1：Tx FIFO はエンプティです。

ビット 10 **TCF**：送信キャンセル終了

0：終了した送信キャンセルはありません。

1：送信キャンセルが終了しました。

ビット 9 **TC**：送信完了

0：完了した送信はありません。

1：送信が完了しました。

ビット 8 **HPM**：高優先メッセージ

0：受信した高優先メッセージはありません。

1：高優先メッセージを受信しました。

ビット 7 **RF1L**：Rx FIFO 1 メッセージ喪失

0：喪失した Rx FIFO 1 メッセージはありません。

1：Rx FIFO 1 メッセージを喪失しました。サイズ 0 の Rx FIFO 1 への書き込み試行後にもセットされます。

ビット 6 **RF1F**：Rx FIFO 1 フル

0：Rx FIFO 1 はフルではありません。

1：Rx FIFO 1 はフルです。

ビット 5 **RF1W**：Rx FIFO 1 ウォーターマーク到達

0：Rx FIFO 1 フィルレベルはウォーターマーク未満です。

1：Rx FIFO 1 フィルレベルはウォーターマークに達しています。

ビット 4 **RF1N**：Rx FIFO 1 新規メッセージ

0：Rx FIFO 1 に書き込まれた新規メッセージはありません。

1：新規メッセージが Rx FIFO 1 に書き込まれました。

ビット 3 **RF0L**：Rx FIFO 0 メッセージ喪失

0：喪失した Rx FIFO 0 メッセージはありません。

1：Rx FIFO 0 メッセージを喪失しました。サイズ 0 の Rx FIFO 0 への書き込み試行後にもセットされます。

ビット 2 **RF0F**：Rx FIFO 0 フル

0：Rx FIFO 0 はフルではありません。

1：Rx FIFO 0 はフルです。

ビット 1 **RF0W**：Rx FIFO 0 ウォーターマーク到達

0：Rx FIFO 0 フィルレベルはウォーターマーク未満です。

1：Rx FIFO 0 フィルレベルはウォーターマークに達しています。

ビット 0 **RF0N**：Rx FIFO 0 新規メッセージ

0：Rx FIFO 0 に書き込まれた新規メッセージはありません。

1：新規メッセージが Rx FIFO 0 に書き込まれました。

56.4.16 FDCAN 割り込み有効レジスタ（FDCAN_IE）

割り込み有効レジスタの設定によって、割り込みレジスタでのどのステータス変更が割り込みラインで信号送信されるかが決まります。

アドレス：0x0054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	BEUE	BECE	DRXE	TOOE	MRAFE	TSWE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFLE	TEFFE	TEFWE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1WE	RF1NE	RF0LE	RF0FE	RF0WE	RF0NE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済み

ビット 29 **ARAE**：予約済みアドレスへのアクセスの有効化

ビット 28 **PEDE**：データフェーズでのプロトコルエラーの有効化

ビット 27 **PEAE**：アービトレーションフェーズでのプロトコルエラーの有効化

ビット 26 **WDIE**：ウォッチドッグの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 25 **BOE**：Bus_Off ステータス

0：割り込みは無効です。

1：割り込みは有効です。

ビット 24 **EWE**：警告ステータスの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 23 **EPE**：エラーパッシブの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 22 **ELOE**：エラーロギングオーバーフローの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 21 **BEUE**：ビットエラー未訂正の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 20 **BECE**：ビットエラー訂正の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 19 **DRXE**：専用 Rx バッファに格納されたメッセージの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 18 **TOOE**：タイムアウト発生の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 17 **MRAFE**：メッセージ RAM アクセス失敗の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 16 **TSWE**：タイムスタンプラップアラウンドの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 15 **TEFLE**：Tx イベント FIFO 要素の喪失の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 14 **TEFFE**：Tx イベント FIFO フルの割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 13 **TEFWE**：Tx イベント FIFO ウォーターマーク到達の割り込み有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 12 **TEFNE**：Tx イベント FIFO 新規エントリの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 11 **TFEE**：Tx FIFO エンプティの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 10 **TCFE**：送信キャンセル終了の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 9 **TCE**：送信完了の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 8 **HPME**：高優先メッセージの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 7 **RF1LE**：Rx FIFO 1 メッセージ喪失の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 6 **RF1FE**：Rx FIFO 1 フルの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 5 **RF1WE**：Rx FIFO 1 ウォーターマーク到達の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 4 **RF1NE**：Rx FIFO 1 新規メッセージの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 3 **RF0LE**：Rx FIFO 0 メッセージ喪失の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 2 **RF0FE**：Rx FIFO 0 フルの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 1 **RF0WE**：Rx FIFO 0 ウォーターマーク到達の割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

ビット 0 **RF0NE**：Rx FIFO 0 新規メッセージの割り込み有効化
0：割り込みは無効です。
1：割り込みは有効です。

56.4.17 FDCAN 割り込みライン選択レジスタ（FDCAN_ILS）

割り込みライン選択レジスタは、割り込みレジスタの特定の割り込みフラグによって生成される割り込みを、2 つのモジュール割り込みラインのうち 1 つに割り当てます。割り込み生成には、それぞれの割り込みラインを ILE[EINT0] および ILE[EINT1] で有効にする必要があります。

アドレス：0x0058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARAL	PEDL	PEAL	WDIL	BOL	EWL	EPL	ELOL	BEUL	BECL	DRXL	TOOL	MRAFL	TSWL
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFLL	TEFFL	TEFWL	TEFNL	TFEL	TCFL	TCL	HPML	RF1LL	RF1FL	RF1WL	RF1NL	RF0LL	RF0FL	RF0WL	RF0NL
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済み

ビット 29 **ARAL**：予約済みアドレスへのアクセスのライン

ビット 28 **PEDL**：データフェーズでのプロトコルエラーのライン

ビット 27 **PEAL**：アービトレーションフェーズでのプロトコルエラーのライン

ビット 26 **WDIL**：ウォッチドッグの割り込みライン

ビット 25 **BOL**：Bus_Off ステータス

ビット 24 **EWL**：警告ステータスの割り込みライン

ビット 23 **EPL**：エラーパッシブの割り込みライン

ビット 22 **ELOL**：エラーロギングオーバーフローの割り込みライン

ビット 21 **BEUL**：ビットエラー未訂正の割り込みライン

ビット 20 **BECL**：ビットエラー訂正の割り込みライン

ビット 19 **DRXL**：専用 Rx バッファに格納されたメッセージの割り込みライン

ビット 18 **TOOL**：タイムアウト発生の割り込みライン

ビット 17 **MRAFL**：メッセージ RAM アクセス失敗の割り込みライン

ビット 16 **TSWL**：タイムスタンプラップアラウンドの割り込みライン

ビット 15 **TEFLL**：Tx イベント FIFO 要素の喪失の割り込みライン

ビット 14 **TEFFL**：Tx イベント FIFO フルの割り込みライン

ビット 13 **TEFWL**：Tx イベント FIFO ウォーターマーク到達の割り込みライン

ビット 12 **TEFNL**：Tx イベント FIFO 新規エントリの割り込みライン

ビット 11 **TFEL**：Tx FIFO エンプティの割り込みライン

ビット 10 **TCFL**：送信キャンセル終了の割り込みライン

ビット 9 **TCL**：送信完了の割り込みライン

ビット 8 **HPML**：高優先メッセージの割り込みライン

ビット 7 **RF1LL**：Rx FIFO 1 メッセージ喪失の割り込みライン

ビット 6 **RF1FL**：Rx FIFO 1 フルの割り込みライン

ビット 5 **RF1WL**：Rx FIFO 1 ウォーターマーク到達の割り込みライン

ビット 4 **RF1NL**：Rx FIFO 1 新規メッセージの割り込みライン

ビット 3 **RF0LL**：Rx FIFO 0 メッセージ喪失の割り込みライン

ビット 2 **RF0FL**：Rx FIFO 0 フルの割り込みライン

ビット 1 **RF0WL**：Rx FIFO 0 ウォーターマーク到達の割り込みライン

ビット 0 **RF0NL**：Rx FIFO 0 新規メッセージの割り込みライン

56.4.18 FDCAN 割り込みライン有効レジスタ（FDCAN_ILE）

CPU への 2 つの割り込みラインは、それぞれビット EINT0 および EINT1 をプログラムすることで有効化／無効化できます。

アドレス：0x005C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
														rw	rw

ビット 31:2 予約済み

ビット 1 **EINT1**：割り込みライン 1 有効化

0：割り込みライン fdcan_intr0_it は無効です。

1：割り込みライン fdcan_intr0_it は有効です。

ビット 0 **EINT0**：割り込みライン 0 有効化

0：割り込みライン fdcan_intr1_it は無効です。

1：割り込みライン fdcan_intr1_it は有効です。

56.4.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_GFC）

メッセージ ID フィルタのグローバル設定です。グローバルフィルタ設定は、*図 728：標準メッセージ ID フィルタパス*および*図 729：拡張メッセージ ID フィルタパス*に示されたとおり、標準メッセージおよび拡張メッセージのフィルタパスを制御します。

アドレス：0x0080

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済み

ビット 5:4 **ANFS**：一致しないフレーム（標準）の許可

フィルタリストの要素と一致しない 11 ビット ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 3:2 **ANFE**：一致しないフレーム（拡張）の許可

フィルタリストの要素と一致しない 29 ビット ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1 **RRFS**：リモートフレーム（標準）の拒否

0：11 ビット標準 ID を持つリモートフレームをフィルタします。

1：11 ビット標準 ID を持つすべてのリモートフレームを拒否します。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 0 **RRFE**：リモートフレーム（拡張）の拒否

0：29 ビット標準 ID を持つリモートフレームをフィルタします。

1：29 ビット標準 ID を持つすべてのリモートフレームを拒否します。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

56.4.20 FDCAN 標準 ID フィルタ設定レジスタ（FDCAN_SIDFC）

11 ビット標準メッセージ ID フィルタの設定です。標準 ID フィルタ設定は、*図 728：標準メッセージ ID フィルタパス*に示されたとおり、標準メッセージのフィルタパスを制御します。

アドレス：0x0084

リセット値：0x0000 0000

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

ビット 31:24 予約済み

ビット 23:16 **LSS**：リストサイズ標準

0：標準メッセージ ID フィルタはありません。

1～128：標準メッセージ ID フィルタ要素の数です。

>128：128 を超える値は 128 として解釈されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **FLSSA**：フィルタリスト標準開始アドレス

標準メッセージ ID フィルタリストの開始アドレスです（32 ビットワードアドレス、*表 469：標準メッセージ ID フィルタ要素*を参照）。これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.21 FDCAN 拡張 ID フィルタ設定レジスタ（FDCAN_XIDFC）

29 ビット拡張メッセージ ID フィルタの設定です。拡張 ID フィルタ設定は、*図 729：拡張メッセージ ID フィルタパス*に示されたとおり、拡張メッセージのフィルタパスを制御します。

アドレス：0x0088

リセット値：0x0000 0000

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

ビット 31:24 予約済み

ビット 23:16 **LSE**：リストサイズ拡張

0：拡張メッセージ ID フィルタはありません。

1～128：標準メッセージ ID フィルタ要素の数です。

>128：128 を超える値は 128 として解釈されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **FLESA**：フィルタリスト拡張開始開始アドレス

拡張メッセージ ID フィルタリストの開始アドレスです（32 ビットワードアドレス、*表 471：拡張メッセージ ID フィルタ要素*を参照）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.22 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）

アドレス：0x0090

リセット値：0x1FFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EIDM[28:16]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIDM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済み

ビット 28:0 **EIDM**：拡張 ID マスク

拡張フレームの許可フィルタでは、拡張 ID AND マスクは受信フレームのメッセージ ID と AND でつながれます。SAE J1939 での 29 ビット ID のマスキングを対象とします。すべてのビットのリセット値を 1 にすると、マスクはアクティブではなくなります。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が '1' にセットされているときのみ、このビットによる書き込みアクセス可能です。

56.4.23 FDCAN 高優先メッセージステータスレジスタ（FDCAN_HPMS）

このレジスタは、メッセージ ID フィルタ要素が優先度イベント一致を生成するように設定するたびに更新されます。これを使用して、着信高優先メッセージのステータスを監視し、これらのメッセージへの高速アクセスが可能になります。

アドレス：0x0094

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLST	FIDX[6:0]							MSI[1:0]		BIDX[5:0]					
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済み

ビット 15 **FLST**：フィルタリスト

一致フィルタ要素のフィルタリストを示します。

0：標準フィルタリスト

1：拡張フィルタリスト

ビット 14:8 **FIDX**：フィルタインデックス

一致フィルタ要素のインデックスです。範囲は、0～SIDFC[LSS] - 1 または XIDFC[LSE] - 1 です。

ビット 7:6 **MSI**：メッセージストレージインジケータ

00：選択された FIFO がありません。

01：FIFO オーバーランです。

10：メッセージが FIFO 0 に格納されます。

11：メッセージが FIFO 1 に格納されます。

ビット 5:0 **BIDX**：バッファインデックス

メッセージが格納された Rx FIFO 要素のインデックスです。MSI[1] = ‘1’ の場合のみ有効です。

56.4.24 FDCAN 新規データ 1 レジスタ（FDCAN_NDAT1）

アドレス：0x0098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ND31	ND30	ND29	ND28	ND27	ND26	ND25	ND24	ND23	ND22	ND21	ND20	ND19	ND18	ND17	ND16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ND15	ND14	ND13	ND12	ND11	ND10	ND9	ND8	ND7	ND6	ND5	ND4	ND3	ND2	ND1	ND0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **NDn**：新規データ [31:0]

このレジスタは、Rx バッファ 0～31 の新規データフラグを保持します。それぞれの Rx バッファが受信フレームから更新されると、フラグがセットされます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に ‘1’ を書き込むことによってクリアされます。’0’ を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

0：Rx バッファは更新されません。

1：Rx バッファは新規メッセージから更新されます。

56.4.25 FDCAN 新規データ 2 レジスタ（FDCAN_NDAT2）

アドレス：0x009C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **NDn**：新規データ [63:32]

このレジスタは、Rx バッファ 32～63 の新規データフラグを保持します。それぞれの Rx バッファが受信フレームから更新されると、フラグがセットされます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に ‘1’ を書き込むことによってクリアされます。’0’ を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

0：Rx バッファは更新されません。

1：Rx バッファは新規メッセージから更新されます。

56.4.26 FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）

アドレス：0x00A0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
F0OM	F0WM[7:0]							Res.	F0S[7:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F0SA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 **F0OM**：FIFO 0 動作モード

FIFO 0 が、ブロックモードまたは上書きモードで動作可能です。

0：FIFO 0 ブロックモード

1：FIFO 0 上書きモード

ビット 30:24 **F0WM**：FIFO 0 ウォーターマーク

0：ウォーターマーク割り込みは無効です。

1～64：Rx FIFO 0 ウォーターマーク割り込み（IR[RF0W]）のレベルです。

>64：ウォーターマーク割り込みは無効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 23 予約済み

ビット 22:16 **F0S**：Rx FIFO 0 サイズ

0：Rx FIFO 0 はありません。

1～64：Rx FIFO 0 要素の数です。

>64：64 を超える値は 64 として解釈されます。

Rx FIFO 0 要素は 0～F0S-1 でインデックスされます。

ビット 15:2 **F0SA**：Rx FIFO 0 開始アドレス

メッセージ RAM での Rx FIFO 0 の開始アドレスです（32 ビットワードアドレス、*図 727：メッセージ RAM 設定*を参照）。

ビット 1:0 予約済み

56.4.27 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）

アドレス：0x00A4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	F0PI[5:0]					
						rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	F0GI[5:0]						Res.	F0FL[6:0]						
		rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済み

ビット 25 **RF0L**：Rx FIFO 0 メッセージ喪失

このビットは割り込みレジスタ IR[RF0L] のコピーです。IR[RF0L] がリセットされると、このビットもリセットされます。

0：喪失した Rx FIFO 0 メッセージはありません。

1：Rx FIFO 0 メッセージを喪失しました。サイズ 0 の Rx FIFO 0 への書き込み試行後にもセットされます。

ビット 24 **F0F**：Rx FIFO 0 フル

0：Rx FIFO 0 はフルではありません。

1：Rx FIFO 0 はフルです。

ビット 23:22 予約済み

ビット 21:16 **F0PI**：Rx FIFO 0 Put インデックス

Rx FIFO 0 書き込みインデックスポインタで、範囲は 0～63 です。

ビット 15:14 予約済み

ビット 13:8 **F0GI**：Rx FIFO 0 Get インデックス

Rx FIFO 0 読み出しインデックスポインタで、範囲は 0～63 です。

ビット 7 予約済み

ビット 6:0 **F0FL**：Rx FIFO フィルレベル

Rx FIFO 0 に格納される要素の数で、範囲は 0～64 です。

56.4.28 CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）

アドレス：0x00A8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済み

ビット 25 **FA01**：Rx FIFO 0 確認応答インデックス

ホストが Rx FIFO 0 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 0 から読み出された最後の要素のバッファインデックスを F0AI に書き込む必要があります。これにより、Rx FIFO 0 Get インデックス RXF0S[F0GI] が F0AI + 1 にセットされ、FIFO 0 フィルレベル RXF0S[F0FL] が更新されます。

56.4.29 FDCAN Rx バッファ設定レジスタ（FDCAN_RXBC）

アドレス：0x00AC

リセット値：0x0000 0000

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

ビット 31:16 予約済み

ビット 15:2 **RBSA**：Rx バッファ開始アドレス

メッセージ RAM での Rx バッファセクションの開始アドレス（32 ビットワードアドレス）を設定します。リファレンスデバッグメッセージ A、B、C にも使用されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.30 FDCAN Rx FIFO 1 設定レジスタ（FDCAN_RXF1C）

アドレス：0x00B0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
F1OM	F1WM[7:0]							Res.	F1S[7:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F1SA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 **F1OM**：FIFO 0 動作モード

FIFO 1 が、ブロックモードまたは上書きモードで動作可能です。

0：FIFO 1 ブロックモード

1：FIFO 1 上書きモード

ビット 30:24 **F1WM**：Rx FIFO 1 ウォーターマーク

0：ウォーターマーク割り込みは無効です。

1～64：Rx FIFO 1 ウォーターマーク割り込み（IR[RF1W]）のレベルです。

>64：ウォーターマーク割り込みは無効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 23 予約済み

ビット 22:16 **F1S**：Rx FIFO 1 サイズ

0：Rx FIFO 1 はありません。

1～64：Rx FIFO 1 要素の数です。

>64：64 を超える値は 64 として解釈されます。

Rx FIFO 1 要素は 0～F1S - 1 でインデックスされます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **F1SA**：Rx FIFO 1 開始アドレス

メッセージ RAM での Rx FIFO 1 の開始アドレスです（32 ビットワードアドレス、*図 727：メッセージ RAM 設定*を参照）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.31 FDCAN Rx FIFO 1 ステータスレジスタ（FDCAN_RXF1S）

アドレス：0x00B4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMS[1:0]		Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	F1PI[6:0]					
r	r					r	r			r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	F1GI[6:0]						Res.	F1FL[7:0]						
		r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:30 **DMS**：デバッグメッセージステータス

00：アイドル状態、デバッグメッセージの受信を待って、DMAリクエストはクリアされます。

01：デバッグメッセージ A を受信しました。

10：デバッグメッセージ A、B を受信しました。

11：デバッグメッセージ A、B、C を受信し、DMAリクエストがセットされました。

ビット 29:26 予約済み

ビット 25 **RF1L**：Rx FIFO 1 メッセージ喪失

このビットは割り込みレジスタ IR[RF1L] のコピーです。IR[RF1L] がリセットされると、このビットもリセットされます。

0：喪失した Rx FIFO 1 メッセージはありません。

1：Rx FIFO 1 メッセージを喪失しました。サイズ 0 の Rx FIFO 1 への書き込み試行後にもセットされます。

ビット 30:24 **F1F**：Rx FIFO 1 フル

0：Rx FIFO 1 はフルではありません。

1：Rx FIFO 1 はフルです。

ビット 23:22 予約済み

ビット 21:16 **F1PI**：Rx FIFO 1 Put インデックス

Rx FIFO 1 書き込みインデックスポインタで、範囲は 0～63 です。

ビット 15:14 予約済み

ビット 13:8 **F1GI**：Rx FIFO 1 Get インデックス
Rx FIFO 1 読み出しインデックスポインタで、範囲は 0～63 です。

ビット 7 予約済み

ビット 6:0 **F1FL**：Rx FIFO1 フィルレベル
Rx FIFO 1 に格納される要素の数で、範囲は 0～64 です。

56.4.32 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）

アドレス：0x00B8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済み

ビット 5:0 **F1AI**：Rx FIFO 1 確認応答インデックス
ホストが Rx FIFO 1 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 1 から読み出された最後の要素のバッファインデックスを F1AI に書き込む必要があります。これにより、Rx FIFO 1 Get インデックス RXF1S[F1GI] が F1AI + 1 にセットされ、FIFO 1 フィルレベル RXF1S[F1FL] が更新されます。

56.4.33 FDCAN Rx バッファ要素サイズ設定レジスタ（FDCAN_RXESC）

Rx バッファ／Rx FIFO 要素に属するデータバイトの数を設定します。8 バイトを超えるデータフィールドサイズは、CAN FD 動作のみを対象とします。

アドレス：0x00BC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	RBDS[2:0]			Res.	F1DS[2:0]			Res.	F0DS[2:0]		
					r	r	r		r	r	r		r	r	r

ビット 31:11 予約済み

ビット 10:8 **RBDS**：Rx バッファデータフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

ビット 7 予約済み

ビット 6:4 **F1DS**：Rx バッファ 0 データフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

ビット 3 予約済み

ビット 2:0 **F0DS**：Rx バッファ 1 データフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

56.4.34 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）

アドレス：0x00C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TFQM	TFQS[5:0]						Res.	Res.	NTDB[5:0]					
	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TBSA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 予約済み

ビット 30 **TFQM**：Tx FIFO／キューモードを示します。

0：Tx FIFO 動作

1：Tx キュー動作

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 29:24 **TFQS**：送信 FIFO／キューサイズを示します。

0：Tx FIFO／キューはありません。

1～32：Tx FIFO／キューに使用される Tx バッファの数を示します。

>32：32 を超える値は 32 として解釈されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 23:22 予約済み

ビット 21:16 **NDTB**：専用送信バッファの数です。

0：専用 Tx バッファはありません。

1～32：専用 Tx バッファの数です。

>32：32 を超える値は 32 として解釈されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **TBSA**：Tx バッファ開始アドレス

メッセージ RAM での Tx FIFO セクションの開始アドレスです（32 ビットワードアドレス、*図 727* を参照）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

注： ***TFQS および NDTB の合計は 33 以上にできません。誤った設定はチェックされません。メッセージ RAM の Tx バッファセクションは、専用 Tx バッファで開始します。***

56.4.35 FDCAN Tx FIFO／キューステータスレジスタ（FDCAN_TXFQS）

Tx FIFO／キューステータスは、レジスタ TXBRP に示されたペンディング中の Tx リクエストに関連します。そのため、追加／キャンセルリクエストの効果は、実行中の Tx スキャン（TXBRP がまだ更新されていない）によって遅延する場合があります。

アドレス：0x00C4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	TFQPI[4:0]				
										r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TFGI[4:0]					Res.	Res.	TFFL[5:0]					
			r	r	r	r	r			r	r	r	r	r	r

ビット 31:22 予約済み

ビット 21 **TFQF**：Tx FIFO／キューフル

0：Tx FIFO／キューはフルではありません。

1：Tx FIFO／キューはフルです。

ビット 20:16 **TFQPI**：Tx FIFO／キュー Put インデックス

Tx FIFO／キュー書き込みインデックスポインタで、範囲は 0～31 です。

ビット 15:13 予約済み

ビット 12:8 **TFGI**：

Tx FIFO Get インデックス

Tx FIFO 読み出しインデックスポインタで、範囲は 0～31 です。Tx キュー動作が設定された場合（TXBC.TFQM = '1'）、0 で読み出されます。

ビット 7:6 予約済み

ビット 5:0 **TFFL**：Tx FIFO フリーレベル

連続するフリー Tx FIFO 要素の数は TFGI から開始し、範囲は 0～32 です。Tx キュー動作が設定された場合（TXBC[TFQM] = '1'）、0 で読み出されます。

注： ***専用 Tx バッファを Tx FIFO や Tx キューと組み合わせた混合設定の場合、Put インデックスおよび Get インデックスは最初の専用 Tx バッファから始まる Tx バッファの数を示します。例：12 の専用 Tx バッファと 20 のバッファの Tx FIFO の設定の場合、15 ポイントの Put インデックスは、Tx FIFO の 4 番目のバッファを指します。***

56.4.36 FDCAN Tx バッファ要素サイズ設定レジスタ（FDCAN_TXESC）

Tx バッファ 要素に属するデータバイトの数を設定します。8 バイトを超えるデータフィールドサイズは、CAN FD 動作のみを対象とします。

アドレス：0x00C8

リセット値：0x0000 0000

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

ビット 31:3 予約済み

ビット 21 **TBDS**：Tx バッファデータフィールドサイズ：

000：8 バイトのデータフィールド

001：12 バイトのデータフィールド

010：16 バイトのデータフィールド

011：20 バイトのデータフィールド

100：24 バイトのデータフィールド

101：32 バイトのデータフィールド

110：48 バイトのデータフィールド

111：64 バイトのデータフィールド

56.4.37 FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）

アドレス：0x00C8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TRP[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **TRP**：送信リクエストペンディングです。

各 Tx バッファには独自の送信リクエストペンディングビットがあります。このビットは TXBAR 経由でセットされます。リクエストされた送信が完了した後か、レジスタ TXBCR 経由でキャンセルされた後に、ビットがリセットされます。

TXBC 経由で設定された Tx バッファにのみ TXBRP ビットがセットされます。TXBRP ビットがセットされた後、Tx スキャン（*デバッグメッセージのフィルタ*を参照）が開始され、最高優先度を持つペンディング中の Tx リクエスト（一番小さいメッセージ ID の Tx バッファ）をチェックします。

キャンセルリクエストで、レジスタ TXBRP の対応する送信リクエストペンディングビットがリセットされます。キャンセルがリクエストされたときに送信をすでに開始していた場合、送信が成功したかどうかに関係なく、送信の最後に実行されます。対応する TXBRP ビットがリセットされた直後に、キャンセルリクエストビットがリセットされます。

キャンセルがリクエストされた後、終了したキャンセルは次のタイミングで TXBCF 経由で信号送信されます。

– 対応する TXBTO ビットとともに送信が成功した後
– キャンセル時点で送信がまだ開始されていなかったとき
– ロストアービトレーションによって送信がアボートされたとき
– フレーム送信中にエラーが発生したとき

DAR モードでは、すべての送信が成功しなかった場合、自動的にキャンセルされます。すべての成功しなかった送信に対応する TXBCF ビットがセットされます。

0：送信リクエストペンディングはありません。
1：送信リクエストペンディングがあります。

注： ***Tx スキャンが進行中のときにセットされた TXBRP ビットは、この特定の Tx スキャン中は考慮されません。このような Tx バッファに対してキャンセルがリクエストされた場合、この追加リクエストはすぐにキャンセルされ、対応する TXBRP ビットはリセットされます。***

56.4.38 FDCAN Tx バッファ追加リクエストレジスタ（FDCAN_TXBAR）

アドレス：0x00D0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **AR**：追加リクエスト

各 Tx バッファには独自の追加リクエストビットがあります。‘1’ を書き込むと、対応する追加リクエストビットがセットされ、‘0’ を書き込んでも影響はありません。これにより、ホストは複数の Tx バッファに対する送信リクエストを TXBAR への 1 回の書き込みでセットできます。TXBC 経由で設定された Tx バッファにのみ TXBAR ビットがセットされます。実行中の Tx スキャンがないときはビットがすぐにリセットされ、それ以外の場合は Tx スキャンプロセスが完了するまでビットがセットされたままとなります。

0：追加された送信リクエストはありません。

1：送信リクエストが追加されました。

注： ***追加リクエストがペンディング中の送信リクエストを持つ Tx バッファに適用される（対応する TXBRP がすでにセットされている）場合、リクエストは無視されます。***

56.4.39 FDCAN Tx バッファキャンセルリクエストレジスタ（FDCAN_TXBCR）

アドレス：0x00D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CR**：キャンセルリクエスト

各 Tx バッファには独自のキャンセルリクエストビットがあります。‘1’ を書き込むと、対応するキャンセルリクエストビットがセットされ、‘0’ を書き込んでも影響はありません。

これにより、ホストは複数の Tx バッファに対するキャンセルリクエストを TXBCR への 1 回の書き込みでセットできます。TXBC 経由で設定された Tx バッファにのみ TXBCR ビットがセットされます。対応する TXBRP ビットがリセットされるまで、ビットはセットされたままとなります。

0：ペンディング中のキャンセルはありません。

1：キャンセルがペンディングされています。

56.4.40 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）

アドレス：0x00D8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TO[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TO[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **TO**：送信発生

各 Tx バッファには独自の送信発生ビットがあります。送信成功後に対応する TXBRP ビットがクリアされたとき、ビットがセットされます。レジスタ TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：発生した送信はありません。

1：送信が発生しました。

56.4.41 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）

アドレス：0x00DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **CF**：キャンセル終了

各 Tx バッファには独自のキャンセル終了ビットがあります。キャンセルが TXBCR ビットによってリクエストされた後に対応する TXBRP ビットがクリアされたとき、ビットがセットされます。対応する TXBRP ビットがキャンセル時点でセットされなかった場合、CF がすぐにセットされます。レジスタ TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：送信バッファキャンセルはありません。

1：送信バッファキャンセルが終了しました。

56.4.42 FDCAN Tx バッファ送信割り込み有効レジスタ（FDCAN_TXBTIE）

アドレス：0x00E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TIE**：送信の割り込み有効化

各 Tx バッファには独自の送信割り込み有効化ビットがあります。

0：送信割り込みは無効です。

1：送信割り込みは有効です。

56.4.43 FDCAN Tx バッファキャンセル終了割り込み有効レジスタ（FDCAN_ TXBCIE）

アドレス：0x00E4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CFIE[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFIE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CF**：キャンセル終了の割り込み有効化

各 Tx バッファには独自のキャンセル終了割り込み有効化ビットがあります。

0：キャンセル終了割り込みは無効です。

1：キャンセル終了割り込みは有効です。

56.4.44 FDCAN Tx イベント FIFO 設定レジスタ（FDCAN_TXEFC）

アドレス：0x00F0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EFWM[5:0]						Res.	Res.	EFS[5:0]					
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EFSA[15:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:30 予約済み

ビット 29:24 **EFWM**：イベント FIFO ウォーターマーク

0：ウォーターマーク割り込みは無効です。

1～32：Tx イベント FIFO ウォーターマーク割り込み（IR[TEFW]）のレベルです。

>32：ウォーターマーク割り込みは無効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 23:22 予約済み

ビット 21:16 **EFS**：イベント FIFO サイズ

0：Tx イベント FIFO は無効です。

1～32：Tx イベント FIFO 要素の数です。

>32：32 を超える値は 32 として解釈されます。

Tx イベント FIFO 要素は 0～EFS-1 でインデックスされます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **EFSA**：イベント FIFO 開始アドレス

メッセージ RAM での Tx イベント FIFO の開始アドレスです（32 ビットワードアドレス、*図 727：メッセージ RAM 設定*を参照）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.45 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）

アドレス：0x00F4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	EFPI[4:0]				
						r	r				r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EFGI[4:0]					Res.	Res.	EFFL[5:0]					
			r	r	r	r	r			r	r	r	r	r	r

ビット 31:26 予約済み

ビット 25 **TEFL**：Tx イベント FIFO 要素の喪失

このビットは割り込みレジスタ IR[TEFL] のコピーです。IR[TEFL] がリセットされると、このビットもリセットされます。

0：Tx イベント FIFO 要素の喪失はありません。

1：Tx イベント FIFO 要素を喪失しました。サイズ 0 の Tx イベント FIFO への書き込み試行後にもセットされます。

ビット 24 **EFF**：イベント FIFO フル

0：Tx イベント FIFO はフルではありません。

1：Tx イベント FIFO はフルです。

ビット 23:21 予約済み

ビット 20:16 **EFPI**：イベント FIFO Put インデックス

Tx イベント FIFO 書き込みインデックスポインタで、範囲は 0～31 です。

ビット 15:13 予約済み

ビット 12:8 **EFGI**：イベント FIFO Get インデックス

Tx イベント FIFO 読み出しインデックスポインタで、範囲は 0～31 です。

ビット 7:6 予約済み

ビット 5:0 **EFFL**：イベント FIFO フィルレベル

Tx イベント FIFO に格納される要素の数で、範囲は 0～31 です。

56.4.46 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）

アドレス：0x00F8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[4:0]				
											rw	rw	rw	rw	rw

ビット 31:5 予約済み

ビット 25 **EFAI**：イベント FIFO 確認応答インデックス

ホストが Tx イベント FIFO から要素または一連の要素を読み出した後、Tx イベント FIFO から読み出された最後の要素のインデックスを EFAI に書き込む必要があります。これにより、Tx イベント FIFO Get インデックス TXEFS[EFGI] が EFAI + 1 にセットされ、FIFO 0 フィルレベル TXEFS[EFFL] が更新されます。

56.4.47 FDCAN TT トリガメモリ設定レジスタ（FDCAN_TTTMC）

アドレス：0x100

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TME[6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMSA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:23 予約済み

ビット 22:16 **TME**：トリガメモリ要素

0：トリガメモリはありません。

1～64：トリガメモリ要素の数です。

>64：64 を超える値は 64 として解釈されます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:2 **TMSA**：トリガメモリ開始アドレス

メッセージ RAM でのトリガメモリの開始アドレスです（32 ビットワードアドレス、*図 727：メッセージ RAM 設定*を参照）。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 1:0 予約済み

56.4.48 FDCAN TT リファレンスメッセージ設定レジスタ（FDCAN_TTRMC）

アドレス：0x0104

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RMPS	XTD	Res.	RID[29:16]												
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RID[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RMPS**：リファレンスメッセージペイロード選択

タイムスレーブの場合は無視されます。

0：リファレンスメッセージに追加のペイロードはありません。

1：次の要素は Tx バッファ 0 から取得されます。

メッセージマーカー MM

イベント FIFO 制御 EFC

データ長コード DLC

データバイト DB（レベル 1：バイト 2～8、レベル 0、2：バイト 5～8）

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 30 **XTD**：拡張識別子

0：11 ビット標準識別子です。

1：29 ビット拡張識別子です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 29 予約済み

ビット 28:0 **RID**：リファレンス識別子

リファレンスメッセージとともに送信される識別子で、リファレンスメッセージフィルタに使用されます。ビット XTD に応じた標準リファレンス識別子または拡張リファレンス識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

56.4.49 FDCAN TT 動作設定レジスタ（FDCAN_TTOCF）

アドレス：0x0108

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	EVTP	ECC	EGTF	AWL[7:0]							
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EECS	IRTO[6:0]							LDSDL[2:0]			TM	GEN	Res.	OM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット 31:27 予約済み

ビット 26 **EVTP**：イベントトリガ極性

0：立ち上がりエッジトリガ

1：立ち下がりエッジトリガ

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 25 **ECC**：クロック較正有効化

0：FDCAN レベル 0、2 での自動クロック較正は無効です。

1：FDCAN レベル 0、2 での自動クロック較正は有効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 24 **EGTF**：グローバルタイムフィルタ有効化

0：FDCAN レベル 0、2 でのグローバルタイムフィルタは無効です。

1：FDCAN レベル 0、2 でのグローバルタイムフィルタは有効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 23:16 **AWL**：アプリケーションウォッチドッグ制限

アプリケーションウォッチドッグは、AWL を 0x00 にプログラムすることで無効化できます。

0x00–FF：アプリケーションがアプリケーションウォッチドッグを機能させる必要がある時点までの最大時間です。アプリケーションウォッチドッグは、256 NTU 経つとインクリメントされます。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15 **EECS**：外部クロック同期有効化

有効の場合、FDCAN 動作中に TUR 設定（TURCF[NCL] のみ）を更新できます。

0：FDCAN レベル 0、2 での外部クロック同期は無効です。

1：FDCAN レベル 0、2 での外部クロック同期は有効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 14:8 **IRTO**：初期リファレンストリガオフセット

0x00–7F：正のオフセットで、範囲は 0～127 です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 7:5 **LDSDL**：同期偏差制限の LD

同期偏差制限 SDL は、2 重対数 LDSDL によって SDL= 2*(LDSDL + 5) で設定されます。SDL は 32～4096 の範囲です。CAN ビットタイミング設定によって示されたクロック許容誤差を超えてはなりません。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 4 **TM**：タイムマスタ

0：タイムマスタ機能は無効です。

1：潜在的なタイムマスタです。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 3 **GEN**：ギャップ有効化

0：厳密なタイムトリガ動作

1：外部イベント同期タイムトリガ動作

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 2 予約済み

ビット 1:0 **OM**：動作モード

00：イベント駆動 CAN 通信（デフォルト）

01：TTCAN レベル 1

10：TTCAN レベル 2

11：TTCAN レベル 0

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

56.4.50 FDCAN TT マトリックス制限レジスタ（FDCAN_TTMLM）

アドレス：0x010C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	ENTT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TXEW[3:0]				CSS[1:0]		CCM[5:0]					
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済み

ビット 27:16 **ENTT**：期待される Tx トリガの数

0x000–FFF：1 つのマトリックスサイクルで期待される Tx トリガの数です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:12 予約済み

ビット 11:8 **TXEW**：Tx 有効化ウィンドウ

0x0–F Tx：有効化ウィンドウの長さ、1～16 NTU サイクルです。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 7:6 **CSS**：サイクル開始同期

同期パルス出力を有効化します。

00：同期パルスなし

01：基本サイクルの開始時に同期パルス

10：マトリックスサイクルの開始時に同期パルス

11：予約済み

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 5:0 **CCM**：サイクルカウント最大

0x00：マトリックスサイクルごとに 1 基本サイクル

0x01：マトリックスサイクルごとに 2 基本サイクル

0x03：マトリックスサイクルごとに 4 基本サイクル

0x07：マトリックスサイクルごとに 8 基本サイクル

0x0F：マトリックスサイクルごとに 16 基本サイクル

0x1F：マトリックスサイクルごとに 32 基本サイクル

0x3F：マトリックスサイクルごとに 64 基本サイクル

その他：予約済み

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

注： ***ISO 11898-4、セクション 5.2.1 では、リストされたサイクルカウント値のみ設定することが必須とされています。その他の値は可能ですが、一貫性のないマトリックスサイクルになる可能性があります。***

56.4.51 FDCAN TUR 設定レジスタ（FDCAN_TURCF）

NTU の長さは次のように求められます。NTU = CAN クロック周期 x NC/DC。

NC は 18 ビット値です。上位ビットNCH[17:16] は 0b01 にハードで組み込まれています。そのため、NC の範囲は、0x10000 から 0x1FFFF までとなります。NCL によって設定された値は、TURNA[NAV[15:0]] の初期値です。DC はハードウェアリセットによって 0x1000 にセットされ、0x0000 に書き込むことはできません。

- レベル 1：NC 4 × DC および NTU = CAN ビット時間
- レベル 0 とレベル 2：NC 8 × DC

TUR の実際の値は、TTCAN レベル 0 およびレベル 2 のクロックドリフト補正機能で変更でき、NTU のノードのローカル表示を NTU のタイムマスタ表示に調整できます。DC は自動ドリフト補正によって変更されず、TURNA[NAV] は TTOCF[LDSDL] で示された同期偏差制限の範囲で NC 前後に調整できます。ドリフト補正プロセスに最適な演算精度を可能にするために、TURCF[NCL] は、NC および DC は最適な値にプログラムすることを推奨します。

アドレス：0x0110

リセット値：0x1000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ELT	Res.	DC[14:0]													
rw	r	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NCL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ELT**：ローカルタイム有効化

0：ローカルタイムが停止されます（デフォルト）。

1：ローカルタイムは有効です。

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

注： ***ローカルタイムは ELT をセットすることで開始されます。ELT がリセットされるか、次のハードウェアリセットまでアクティブのままとなります。TURCF[ELT] = ‘1’ の場合、TURCF[DC] はロックされます。ELT に ‘0’ が書き込まれた場合、読み出し可能な値は新しい値が CAN クロックドメインに同期されるまで ‘1’ のままとなります。この時間、レジスタの他のビットへの書き込みはロックされたままとなります。***

ビット 30 予約済み

ビット 29:16 **DC**：Denominator設定

0x0000：不正な値

0x0001 から 3FFF：Denominator設定

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

ビット 15:0 **NCL**：Numerator設定ロー

TUR のNumerator設定ローへの書き込みアクセスは、TURCF[ELT] = ‘0’ の設定中、または TTOCF[EECS]（外部クロック同期有効）がセットされている場合のみ可能です。NCL の新しい値が TT 設定モード以外で書き込まれた場合、新しい値は TTOST.WECS が ‘0’ にクリアされたときに有効になります。NCL はロックされ、TTOST[WECS] は ‘1’ になります。

0x0000–FFFF：Numerator設定ロー

これらは書き込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が ‘1’ にセットされているときのみ、このビットによる書き込みアクセス可能です。

注： ***TTCAN レベル 1 で NC < 7 × DC の場合、トリガメモリのその後のタイムマークは少なくとも 2 NTU 異なっている必要があります。***

56.4.52 FDCAN TT 動作制御レジスタ（FDCAN_TTOCN）

アドレス：0x0114

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCKC	Res.	ESCN	NIG	TMG	FGP	GCS	TTIE	TMC[1:0]		RTIE	SWS[1:0]		SWP	ECS	SGT
r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み

ビット 15 **LCKC**：TT 動作制御レジスタロック

レジスタ TTOCN への書き込みアクセスによってセットされます。更新された設定が CAN クロックドメインに

同期されたときにリセットされます。

0：TTOCN への書き込みアクセスは有効です。

1：TTOCN への書き込みアクセスはロックされています。

ビット 14 予約済み

ビット 13 **ESCN**：外部同期制御

有効の場合、FDCAN は、サイクルタイムフェーズをイベントトリガピンの立ち上がりエッジによって信号送信される外部イベントに同期します（セクション 0.1.17、外部タイムスケジュールへの同期）。

0：外部同期は無効です。

1：外部同期は有効です。

ビット 12 **NIG**：ネクストイズギャップ

このビットは、FDCAN が実際のタイムマスタである場合と、外部イベント同期タイムトリガ動作（TTOCF[GEN] = ‘1’）に設定されている場合にのみセットできます。

0：動作なし、リファレンスメッセージの受信でリセットされます。

1：Next_is_Gap = ‘1’ のある次のリファレンスメッセージを送信します。

ビット 11 **TMG**：タイムマークギャップ

0：各リファレンスメッセージでリセットされます。

1：レジスタタイムマーク割り込み TTIR[RTMI] が有効になると、次のリファレンスメッセージが開始されます。

ビット 10 **FGP**：終了ギャップ

CPU によってセットされ、各リファレンスメッセージでリセットされます。

0：リクエストされたリファレンスメッセージはありません。

1：アプリケーションがリファレンスメッセージの開始をリクエストしました。

ビット 9 **GCS**：ギャップ制御選択

0：ギャップ制御はイベントトリガから独立しています。

1：ギャップ制御は入力イベントトリガピンによって行われます。

ビット 8 **TTIE**：トリガタイムマーク割り込みパルス有効化

外部タイムマークイベントはトリガメモリ要素 TMEX によって設定されます。トリガメモリ要素がアクティブになったときに、トリガタイムマーク割り込みパルスが生成され、FDCAN は同期状態 In_Schedule または In_Gap になります。

0：トリガタイムマーク割り込み出力 m_ttcan_tmp は無効です。

1：トリガタイムマーク割り込み出力 m_ttcan_tmp は有効です。

ビット 7:6 **TMC**：レジスタタイムマーク比較

00：生成されるレジスタタイムマーク割り込みはありません。

01：タイムマーク = サイクルタイムの場合にレジスタタイムマーク割り込みになります。

10：タイムマーク = ローカルタイムの場合にレジスタタイムマーク割り込みになります。

11：タイムマーク = グローバルタイムの場合にレジスタタイムマーク割り込みになります。

注： ***タイムマークリファレンス（サイクルタイム、ローカルタイム、グローバルタイム）を変更するとき、まず TMC = ‘00’ を書き込んでから TTTMK を設定し、最後に TMC を目的のタイムリファレンスにセットすることを推奨します。***

ビット 5 **RTIE**：レジスタタイムマーク割り込みパルス有効化

レジスタタイムマーク割り込みはレジスタ TTTMK によって設定されます。1 m_ttcan_clk 周期の長さのあるレジスタタイムマーク割り込みパルスは、TTOCN[TMC] によって参照される時間（サイクル、ローカル、グローバル）が同期状態に関係なく TTTMK[TM] と等しくなったときに生成されます。

0：レジスタタイムマーク割り込み出力は無効です。

1：レジスタタイムマーク割り込み出力は有効です。

ビット 4:3 **SWS**：ストップウォッチソース

00：ストップウォッチは無効です。

01：サイクルタイムの実際の値が TTCPT[SWV] にコピーされます。

10：ローカルタイムの実際の値が TTCPT[SWV] にコピーされます。

11：グローバルタイムの実際の値が TTCPT[SWV] にコピーされます。

ビット 2 **SWP**：ストップウォッチ極性
0：立ち上がりエッジトリガ
1：立ち下がりエッジトリガ

ビット 1 **ECS**：外部クロック同期
ノードが実際のタイムマスタである場合に、ECS に ‘1’ を書き込むと、TTOST[WECS] がセットされます。1 APB クロック周期後に ECS はリセットされます。次の基本サイクルの開始時に外部クロック同期が有効になります。

ビット 0 **SGT**：グローバルタイムのセット
ノードが実際のタイムマスタである場合に、SGT に ‘1’ を書き込むと、TTOST[WGDT] がセットされます。1 APB クロック周期後に SGT はリセットされます。ノードが TTGTP に書き込まれたプリセット値によって変更された Master_Ref_Mark のある次のリファレンスメッセージを送信するときに、グローバルタイムプリセットが有効になります。

56.4.53 FDCAN TT グローバルタイムプリセットレジスタ（CAN_TTGTP）

TTOST.WGDT がセットされている場合、次のリファレンスメッセージがプリセット値によって変更された Master_Ref_Mark とともに送信され、Disc_Bit = ‘1’ ですべてのノードのグローバルタイムを同時にプリセットします。

Disc_Bit = ‘1’のリファレンスメッセージが有効になるたびに、あるいはノードが現在のタイムマスタではない場合に、TP は 0x0000 にリセットされます。TTOCN[SGT] をセットした後 TTOST[WGTD] = ‘1’ である間、Disc_Bit = ‘1’のリファレンスメッセージが有効になるか、ノードが現在のタイムマスタではなくなるまで、TP はロックされます。

アドレス：0x0118

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **CTP**：サイクルタイムの対象フェーズ
TTOCN[ESCN] または TTOST[SPL] がセットされている間、CTP は書き込み保護されます（*セクション 56.3.15：外部タイムスケジュールへの同期*を参照）。
0x0000–FFFF：イベントトリガの立ち上がりエッジが期待されるとき、サイクルタイムの対象値を定義します。

ビット 15:0 **NCL**：タイムプリセット
TTOST[WGTD] がセットされている間、TP は書き込み保護されます。
0x0000–7FFF：次のマスタリファレンスマーク = マスタリファレンスマーク + TP
0x8000：予約済み
0x8001–FFFF：次のマスタリファレンスマーク = マスタリファレンスマーク - (0x10000 - TP)

56.4.54 FDCAN TT タイムマークレジスタ（FDCAN_TTTMK）

TTOCN[TMC] で示されたタイムベース（サイクルタイム、ローカルタイム、またはグローバルタイム）が TM と同じ値を持っているときに、タイムマーク割り込み（TTIR[TMI] = ‘1’）が生成されます。

アドレス：0x011C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LCKM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TICC[6:0]						
r	r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LCKM**：TT タイムマークレジスタロック

レジスタ TTOCN への書き込みアクセスによって常にセットされます。TTOCN[TMC] ‘00’ のとき、レジスタ TTTMK への書き込みアクセスによってセットされます。レジスタが CAN クロックドメインに同期されたときにリセットされます。

0：TTTMK への書き込みアクセスは有効です。

1：TTTMK への書き込みアクセスはロックされています。

ビット 30:23 予約済み

ビット 22:16 **TICC**：タイムマークサイクルコード

タイムマークを有効であるサイクルカウントです。

0b000000x：すべてのサイクルで有効

0b000001c：サイクルカウント mod2 = c で第 2 サイクルごとに有効

0b00001cc：サイクルカウント mod4 = cc で第 4 サイクルごとに有効

0b0001ccc：サイクルカウント mod8 = ccc で第 8 サイクルごとに有効

0b001cccc：サイクルカウント mod16 = cccc で第 16 サイクルごとに有効

0b01ccccc：サイクルカウント mod32 = ccccc で第 32 サイクルごとに有効

0b1cccccc：サイクルカウント mod64 = cccccc で第 64 サイクルごとに有効

ビット 15:0 **TM**：タイムマーク

0x0000–FFFF：タイムマーク

注： ***レジスタ TTTMK へのバイトアクセスを使用している場合、まずタイムマーク比較機能を無効（TTOCN[TMC] = ‘00’）にして、一貫性のないレジスタ値での比較を回避することを推奨します。***

56.4.55 FDCAN TT 割り込みレジスタ（FDCAN_TTIR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に '1' を書き込むことによってクリアされます。"0"を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

アドレス：0x0120

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWT	ELC	SE2	SE1	TXO	TXU	GTE	GTD	GTW	SWE	TTMI	RTMI	SOG	CSM	SMC	SBC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済み

ビット 18 **CER**：設定エラー
トリガの障害
0：トリガリストにエラーがありません。
1：トリガリストにエラーがありました。

ビット 17 **AW**：アプリケーションウォッチドッグ
0：アプリケーションウォッチドッグが時間内に機能しました。
1：アプリケーションウォッチドッグが時間内に機能しません。

ビット 16 **WT**：ウォッチトリガ
0：不足したリファレンスメッセージはありません。
1：リファレンスメッセージが不足しています（レベル 0：サイクルタイム 0xFF00）。

ビット 15 **IWTG**：初期化ウォッチトリガ
初期化が IWT をリセットすることで再開します。
0：システム起動中に不足したリファレンスメッセージはありません。
1：リファレンスメッセージが不足しているためシステム起動しません。

ビット 14 **ELC**：エラーレベル変更
初期化中にエラーレベルが変更されたときセットされません。
0：エラーレベルに変更はありません。
1：エラーレベルが変更されました。

ビット 13 **SE2**：スケジュールエラー 2
0：スケジュールエラー 2 はありません。
1：スケジュールエラー 2 が発生しました。

ビット 12 **SE1**：スケジュールエラー 1
0：スケジュールエラー 1 はありません。
1：スケジュールエラー 1 が発生しました。

ビット 11 **TXO**：Tx カウントオーバーフロー
0：期待された Tx トリガの数です。
1：1 サイクルで期待された以上の Tx トリガの数です。

ビット 10 **TXU**：Tx カウントアンダーフロー

0：期待された Tx トリガの数です。

1：1 サイクルで期待された未満の Tx トリガの数です。

ビット 9 **GTE**：グローバルタイムエラー

同期偏差 SD が TTOCF[LDSDL] によって指定された制限を超えています（TTCAN レベル 0、2 のみ）。

0：同期偏差は制限内です。

1：同期偏差が制限を超えました。

ビット 8 **GTD**：グローバルタイム不連続

0：グローバルタイムの不連続はありません。

1：グローバルタイムの不連続があります。

ビット 7 **GTW**：グローバルタイムラップ

0：グローバルタイムラップは発生していません。

1：0xFFFF～0x0000 のグローバルタイムラップが発生しました。

ビット 6 **SWE**：ストップウォッチイベント

0：ストップウォッチトリガで立ち上がり／立ち下がりエッジは検出されません。

1：ストップウォッチトリガで立ち上がり／立ち下がりエッジが検出されました。

ビット 5 **TTMI**：トリガタイムマークイベント内部

内部タイムマークイベントはトリガメモリ要素 TMIN によって設定されます（*セクション 56.3.21：FDCAN のトリガメモリ要素*を参照）。トリガメモリ要素がアクティブになったときにセットされ、FDCAN は同期状態 In_Gap または In_Schedule になります。

0：タイムマークに達していません。

1：タイムマークに達しました（レベル 0：サイクルタイム TTOCF[RTO] x 0x200）。

ビット 4 **RTMI**：レジスタタイムマーク割り込み

TTOCN[TMC] によって参照される時間（サイクル、ローカル、グローバル）が同期状態に関係なく TTTMK[TM] と等しくなったときにセットされます。

0：タイムマークに達していません。

1：タイムマークに達しました。

ビット 3 **SOG**：ギャップの開始

0：Next_is_Gap がセットされたリファレンスメッセージがありません。

1：Next_is_Gap がセットされたリファレンスメッセージが有効になります。

ビット 2 **CSM**：同期モードの変更

0：スレーブに対するマスタの関係やスケジュール同期に変更はありません。

1：スレーブに対するマスタの関係やスケジュール同期が変更されました。

また、TTOST[SPL] がリセットされるとセットされます。

ビット 1 **SMC**：マトリックスサイクルの開始

0：ビットがリセットされてから開始したマトリックスサイクルはありません。

1：マトリックスサイクルが開始しました。

ビット 0 **SBC**：基本サイクルの開始

0：ビットがリセットされてから開始した基本サイクルはありません。

1：基本サイクルが開始しました。

56.4.56 FDCAN TT 割り込み有効レジスタ（FDCAN_TTIE）

TT 割り込み有効レジスタの設定によって、TT 割り込みレジスタでのどのステータス変更が割り込みとなるかが決まります。

アドレス：0x0124

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERE	AWE	WTE
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWTE	ELCE	SE2E	SE1E	TXOE	TXUE	GTEE	GTDE	GTWE	SWEE	TTMIE	RTMIE	SOGE	CSME	SMCE	SBCE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済み

ビット 18 **CERE**：設定エラー割り込みイネーブル
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 17 **AWE**：アプリケーションウォッチドッグの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 16 **WTE**：ウォッチトリガの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 15 **IWTGE**：初期化ウォッチトリガの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 14 **ELCE**：エラーレベル変更の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 13 **SE2E**：スケジュールエラー 2 の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 12 **SE1E**：スケジュールエラー 1 の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 11 **TXOE**：Tx カウントオーバーフローの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 10 **TXUE**：Tx カウントアンダーフローの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 9 **GTEE**：グローバルタイムエラーの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 8 **GTDE**：グローバルタイム不連続の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 7 **GTWE**：グローバルタイムラップの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 6 **SWEE**：ストップウォッチイベントの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 5 **TTMIE**：トリガタイムマークイベント内部の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 4 **RTMIE**：レジスタタイムマークの割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 3 **SOGE**：ギャップの開始の割り込み有効
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 2 **CSME**：同期モードの変更の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 1 **SMCE**：マトリックスサイクルの開始の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

ビット 0 **SBCE**：基本サイクルの開始の割り込み有効化
0：TT 割り込みは無効です。
1：TT 割り込みは有効です。

56.4.57 FDCAN TT 割り込みライン選択レジスタ（FDCAN_TTILS）

TT 割り込みライン選択レジスタは、TT 割り込みレジスタの特定の割り込みフラグによって生成される割り込みを、2 つのモジュール割り込みラインのうち 1 つに割り当てます。割り込み生成には、それぞれの割り込みラインを ILE[EINT0] および ILE[EINT1] で有効にする必要があります。

アドレス：0x0128

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWTL	ELCL	SE2L	SE1L	TXOL	TXUL	GTEL	GTDL	GTWL	SWEL	TTMIL	RTMIL	SOGL	CSML	SMCL	SBCL
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済み

ビット 18 **CERL**：設定エラーの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 17 **AWL**：アプリケーションウォッチドッグの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 16 **WTL**：ウォッチトリガの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 15 **IWTGL**：初期化ウォッチトリガの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 14 **ELCL**：エラーレベル変更の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 13 **SE2L**：スケジュールエラー 2 の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 12 **SE1L**：スケジュールエラー 1 の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 11 **TXOL**：Tx カウントオーバーフローの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 10 **TXUL**：Tx カウントアンダーフローの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 9 **GTEL**：グローバルタイムエラーの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 8 **GTDL**：グローバルタイム不連続の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 7 **GTWL**：グローバルタイムラップの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 6 **SWEL**：ストップウォッチイベントの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 5 **TTMIL**：トリガタイムマークイベント内部の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 4 **RTMIL**：レジスタタイムマークの割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 3 **SOGL**：ギャップの開始の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 2 **CSML**：同期モードの変更の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 1 **SMCL**：マトリックスサイクルの開始の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

ビット 0 **SBCL**：基本サイクルの開始の割り込みライン
0：TT 割り込みは割り込みライン 0 に割り当てられます。
1：TT 割り込みは割り込みライン 1 に割り当てられます。

56.4.58 FDCAN TT 動作ステータスレジスタ（FDCAN_TTOST）

アドレス：0x012C

リセット値：0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPL	WECS	AWE	WFE	GSI	TMP[2:0]			GFI	WGTD	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	r	r						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTO[7:0]								QCS	QGTP	SYS[1:0]		MS[1:0]		EL[1:0]	
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **SPL**：スケジュールフェーズロック
このビットは、外部同期が有効化されている場合（TTOCN[ESCN] = ‘1’）にのみ有効です。この場合、TTGTP[CTP] によって設定されたサイクルタイムと、イベントトリガピンの立ち上がりエッジのサイクルタイムの差が 9 NTU 以下であることを伝えます（*セクション 56.3.15：外部タイムスケジュールへの同期*を参照）。
0：範囲外のフェーズ
1：範囲内のフェーズ

ビット 30 **WECS**：外部クロック同期の待機
0：ペンディング中の外部クロック同期はありません。
1：ノードが外部クロック同期が有効になるのを待っています。次の基本サイクルの開始時にビットがリセットされます。

ビット 29 **AWE**：アプリケーションウォッチドッグイベント
アプリケーションウォッチドッグは、TTOST を読み出すことで機能します。ウォッチドッグが時間内に機能しない場合、ビット AWE がセットされ、すべての FDCAN 通信が停止して、FDCAN はバスモニタモードにセットされます。
0：アプリケーションウォッチドッグが時間内に機能しました。
1：アプリケーションウォッチドッグが時間内に機能しませんでした。

ビット 28 **WFE**：イベントの待機

0：通知されたギャップはありません。Next_is_Gap = ‘0’ のあるリファレンスメッセージによってリセットされます。

1：Next_is_Gap = ‘1’ のある次のリファレンスメッセージを受信しました。

ビット 27 **GSI**：ギャップ開始インジケータ

0：スケジュールにギャップはありません。すべてのタイムスレーブに対して各リファレンスメッセージによってリセットされます。

1：基本サイクル後のギャップタイムが開始しました。

ビット 26:24 **TMP**：タイムマスタ優先度

0x0-7：実際のタイムマスタの優先度

ビット 23 **GFI**：ギャップ終了インジケータ

CPU が TTOCN[FGP] を書き込んだとき、または TMG = ‘1’ の場合はタイムマーク割り込み、TTOCN[GCS] = ‘1’ の場合は入力ピン（イベントトリガ）経由でセットされます。Ref_Trigger_Gap や、他のノードのリファレンスメッセージ送信によるギャップの終了ではセットされません。

0：各リファレンスメッセージの最後にリセットします。

1：FDCAN によってギャップが終了しました。

ビット 22 **WGTD**：グローバルタイム不連続の待機

0：グローバルタイムプリセットのペンディングはありません。

1：ノードがグローバルタイムプリセットが有効になるのを待っています。ノードが Disc_Bit = ‘1’ のリファレンスメッセージを送信したとき、またはリファレンスメッセージを受信したときに、ビットがリセットされます。

ビット 21:16 予約済み

ビット 15:8 リファレンストリガオフセット

リファレンストリガオフセットの値は、-127（0x81）～127（0x7F）の範囲の符号整数です。-127 の下限に達したときに通知はありません。FDCAN がタイムマスタになる場合（MS[1:0] = ‘11’）、ホストと CAN クロックドメインの同期によって RTO のリセットが遅延します。タイムスレーブの場合、TTOCF[IRTO] によって設定された値が読み出されます。

0x00-FF：実際のリファレンストリガオフセットの値

ビット 7 **QCS**：クロック速度の品質

TTCAN レベル 0 およびレベル 2 にのみ関連し、その他は ‘1’ に固定されます。

0：ローカルのクロック速度はタイムマスタのクロック速度と同期しません。

1：同期偏差 ≤ SDL になります。

ビット 6 **GTP**：グローバルタイムフェーズの品質

TTCAN レベル 0 およびレベル 2 にのみ関連し、その他は ‘0’ に固定されます。

0：グローバルタイムは有効ではありません。

1：グローバルタイムはタイムマスタと同調しています。

ビット 5:4 **SYS**：同期状態

00：同期していません。

01：FDCAN 通信に同期しています。

10：ギャップによってスケジュールがサスペンドされました（In_Gap）。

11：スケジュールに同期しました（In_Schedule）。

ビット 3:2 **MS**：マスタ状態
00：Master_Off、関連するマスタプロパティはありません。
01：タイムスレーブとして動作しています。
10：バックアップタイムマスタとして動作しています。
11：現在のタイムマスタとして動作しています。

ビット 1:0 **EL**：エラーレベル
00：深刻度 0 - エラーなし
01：深刻度 1 - 警告
10：深刻度 2 - エラー
11：深刻度 3 - 深刻なエラー

56.4.59 FDCAN TUR Numerator実測レジスタ（FDCAN_TURNA）

FDCAN レベル 1 では、ドリフト補正はありません（NAV = NC）。TTCAN レベル 0 およびレベル 2 では、ノードのローカルクロックとタイムマスタのローカルクロックのドリフトが計算されます。同期偏差（NC と 計算された NAV の差）が 2*(TTOCF[LDSDL] + 5) を超えない場合、ドリフトが補正されます。TTOCF[LDSDL] 7 で、(NC - 0x1000) ≤ NAV ≤ (NC + 0x1000) の NAV の最大範囲となります。

アドレス：0x0130

リセット値：0x0001 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAV[17:16]	
														r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NAV[17:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:18 予約済み

ビット 17:0 **NAV**：Numeratorの実測値
0x0EFFF：不正な値
0x0F000–20FFF：実際のNumeratorの値
0x21000：不正な値

56.4.60 FDCAN TT ローカルおよびグローバルタイムレジスタ（FDCAN_TTLGT）

アドレス：0x0134

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **GT**：グローバルタイム

ノードのローカルタイムとローカルオフセットの合計の小数ではない部分です（*セクション 56.3.10：ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期*を参照）。

0x0000–FFFF：FDCAN ネットワークのグローバルタイム値

ビット 15:0 **LT**：ローカルタイム

ローカル NTU 経つごとにインクリメントされるローカルタイムの小数ではない部分です（*セクション 56.3.10：ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期*を参照）。

0x0000–FFFF：FDCAN ノードのローカルタイム値

56.4.61 FDCAN TT サイクルタイムおよびカウントレジスタ（FDCAN_TTCTC）

アドレス：0x0138

リセット値：0x003F 0000

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

ビット 31:22 予約済み

ビット 21:16 **CC**：サイクルカウント

0x00–3F：システムマトリックスでの実際の基本サイクルの数

ビット 15:0 **CT**：サイクルタイム

ノードのローカルタイムと Ref_Mark の差の小数ではない部分です（*セクション 56.3.10：ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期*を参照）。

0x0000–FFFF：FDCAN 基本サイクルのサイクルタイム値

56.4.62 FDCAN TT キャプチャタイムレジスタ（FDCAN_TTCPT）

アドレス：0x013C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	CCV[5:0]					
										r	r	r	r	r	r

ビット 31:16 **SWV**：ストップウォッチ値

ストップウォッチトリガピンの立ち上がり／立ち下がりエッジ（TTOCN[SWP] 経由で設定）で、TTOCN[SWS] が ‘00’ ではなく、TTIR[SWE] が ‘0’ であるとき、TTOCN[SWS] によって選択された実際のタイム値（サイクル、ローカル、グローバル）は SWV にコピーされ、TTIR[SWE] は ‘1’ にセットされます。次のストップウォッチ値のキャプチャは TTIR[SWE] をリセットすることで有効になります。

0x0000–FFFF：キャプチャされるストップウォッチ値

ビット 15:6 予約済み

ビット 5:0 **CT**：サイクルカウント値

SWV とともにキャプチャされるサイクルカウント値です。

0x00–3F：キャプチャされるサイクルカウント値

56.4.63 FDCAN TT サイクル同期マークレジスタ（FDCAN_TTCSM）

アドレス：0x0140

リセット値：0x0000 0000

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

ビット 31:16 予約済み

ビット 15:0 **CSM**：サイクル同期マーク

サイクル同期マークはサイクルタイムで測定されます。リファレンスメッセージが有効になったときに更新され、次のリファレンスメッセージが有効になるまで値を保持します。

0x0000–FFFF：キャプチャされるサイクルタイム

56.4.64 FDCAN TT トリガ選択レジスタ（FDCAN_TTTS）

FDCAN_TTTS レジスタでの設定によって、イベントトリガおよびストップウォッチトリガとして使用される入力が選択されます。

アドレス：0x0300

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	Res:	EVTSEL[1:0]		Res:	Res:	SWTDEL[1:0]	
										rw	rw			rw	rw

ビット 31:6 予約済み

ビット 5:4 **EVTSEL**：イベントトリガ入力選択
これらのビットは、イベントトリガとして使用される入力の選択に使用します。
00：fdcan1_swt0
01：fdcan1_swt1
10：fdcan1_swt2
11：fdcan1_swt3

ビット 3:2 予約済み

ビット 1:0 **SWTDEL**：ストップウォッチトリガ入力選択
これらのビットは、ストップウォッチトリガとして使用される入力の選択に使用します。
00：fdcan1_evt0
01：fdcan1_evt1
10：fdcan1_evt2
11：fdcan1_evt3

56.4.65 FDCAN レジスタマップとリセット値の表

表 475. FDCAN レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	FDCAN_CREL	REL [3:0]				STEP [3:0]				SUBSTEP [3:0]				YEAR [3:0]				MON [7:0]								DAY [7:0]							
	リセット値																																
0x0004	FDCAN_ENDN	ETV[31:0]																															
	リセット値	1	0	0	0	0	1	1	1	0	1	1	0	0	1	1	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1
0x0008	予約済み	Res.																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	FDCAN_DBTP	Res.								TDC	Res.			DBRP[4:0]				Res.			DTSEG1[4:0]					DTSEG2[3:0]				DSJW[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0	1	1
0x0010	FDCAN_TEST	Res.																								RX	TX[1:0]		LBCK	CAT	CAM	TAT	TAM
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	0
0x0014	FDCAN_RWD	Res.																WDV[7:0]								WDC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	FDCAN _CCCR	Res.																NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x001C	FDCAN_NBTP	NSJW[6:0]								NBRP[8:0]								NTSEG1[7:0]								Res.	NTSEG2[6:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0	1	1
0x0020	FDCAN_TSCC	Res.												NBRP[3:0]				Res.								Res.						TSS[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	FDCAN_TSCV	Res.																TSC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	FDCAN_TOCC	TOP[15:0]																Res.								Res.					TOS[1:0]		ETOC
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	FDCAN_TOCV	Res.																TOC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0030 ~ 0x003F	予約済み																																
	リセット値																																
0x0040	FDCAN_ECR	Res.								TDCV[7:0]								RP	REC[6:0]							TEC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 475. FDCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0044	FDCAN_PSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]							Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0048	FDCAN_TDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCO[6:0]							Res.	TDCF[6:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
0x0050	FDCAN_IR	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	BEU	BEC	DRX	TOO	MRAF	TSW	TEFL	TEFF	TEFW	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1W	RF1N	RF0L	RF0F	RF0W	RF0N
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	FDCAN_IE	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	BEUE	BECE	DRXE	TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFWE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1WE	RF1NE	RF0LE	RF0FE	RF0WE	RF0NE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0058	FDCAN_ILS	Res.	Res.	ARAL	PEDL	PEAL	WDIL	BOL	EWL	EPL	ELOL	BEUL	BECL	DRXL	TOOL	MRAFL	TSWL	TEFLL	TEFFL	TEFWL	TEFNL	TFEL	TCFL	TCL	HPML	RF1LL	RF1FL	RF1WL	RF1NL	RF0LL	RF0FL	RF0WL	RF0NL
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	FDCAN_ILE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060 ~ 0x007F	予約済み																																
	リセット値																																
0x0080	FDCAN_GFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	FDCAN_SIDFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSS[7:0]								FLSSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0088	FDCAN_ILS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSE[6:0]							FLESA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0090	FDCAN_XIDAM	Res.	Res.	Res.	EIDM[28:0]																												
	リセット値	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0094	FDCAN_HPMS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLST	FIDX[6:0]							MSI[1:0]		BIDX[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0098	FDCAN_NDAT1	ND31	ND30	ND29	ND28	ND27	ND26	ND25	ND24	ND23	ND22	ND21	ND20	ND19	ND18	ND17	ND16	ND15	ND14	ND13	ND12	ND11	ND10	ND9	ND8	ND7	ND6	ND5	ND4	ND3	ND2	ND1	ND0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 475. FDCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x009C	FDCAN_NDAT2	ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48	ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A0	FDCAN_RXF0C	F0OM	F0WM[6:0]							Res.	F0S[6:0]							F0SA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A4	FDCAN_RXF0S	Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	F0PI[5:0]						Res.	Res.	F0GI[5:0]						Res.	F0FL[6:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A8	FDCAN_RXF0A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0FL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00AC	FDCAN_RXBC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	FDCAN_RXF1C	F1OM	F1WM[6:0]							Res.	F1S[6:0]							F1SA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	FDCAN_RXF1S	DMS[1:0]		Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.F1PI[5:0]						Res.	Res.	Res.F1GI[5:0]						Res.	Res.F1FL[6:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B8	FDCAN_RXF1A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1FL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00BC	FDCAN_RXESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBDS[2:0]			Res.	F1DS[2:0]			Res.	F0DS[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C0	FDCAN_TXBC	Res.	TFQM	TFQS[5:0]						Res.	Res.	NDTB[5:0]						TBSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C4	FDCAN_TXFQS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	TFQPI[4:0]					Res.	Res.	Res.	TFGI[4:0]					Res.	Res.	TFFL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8	FDCAN_TXESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0DS[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00CC	FDCAN_TXBRP	TRP31	TRP30	TRP29	TRP28	TRP27	TRP26	TRP25	TRP24	TRP23	TRP22	TRP21	TRP20	TRP19	TRP18	TRP17	TRP16	TRP15	TRP14	TRP13	TRP12	TRP11	TRP10	TRP9	TRP8	TRP7	TRP6	TRP5	TRP4	TRP3	TRP2	TRP1	TRP0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 475. FDCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00D0	FDCAN_TXBAR	AR31	AR30	AR29	AR28	AR27	AR26	AR25	AR24	AR23	AR22	AR21	AR20	AR19	AR18	AR17	AR16	AR15	AR14	AR13	AR12	AR11	AR10	AR9	AR8	AR7	AR6	AR5	AR4	AR3	AR2	AR1	AR0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00D4	FDCAN_TXBCR	CR31	CR30	CR29	CR28	CR27	CR26	CR25	CR24	CR23	CR22	CR21	CR20	CR19	CR18	CR17	CR16	CR15	CR14	CR13	CR12	CR11	CR10	CR9	CR8	CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00D8	FDCAN_TXBTO	TO31	TO30	TO29	TO28	TO27	TO26	TO25	TO24	TO23	TO22	TO21	TO20	TO19	TO18	TO17	TO16	TO15	TO14	TO13	TO12	TO11	TO10	TO9	TO8	TO7	TO6	TO5	TO4	TO3	TO2	TO1	TO0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00DC	FDCAN_TXBCF	CF31	CF30	CF29	CF28	CF27	CF26	CF25	CF24	CF23	CF22	CF21	CF20	CF19	CF18	CF17	CF16	CF15	CF14	CF13	CF12	CF11	CF10	CF9	CF8	CF7	CF6	CF5	CF4	CF3	CF2	CF1	CF0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00E0	FDCAN_TXBTIE	TIE31	TIE30	TIE29	TIE28	TIE27	TIE26	TIE25	TIE24	TIE23	TIE22	TIE21	TIE20	TIE19	TIE18	TIE17	TIE16	TIE15	TIE14	TIE13	TIE12	TIE11	TIE10	TIE9	TIE8	TIE7	TIE6	TIE5	TIE4	TIE3	TIE2	TIE1	TIE0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00E4	FDCAN_TXBCIE	CFIE31	CFIE30	CFIE29	CFIE28	CFIE27	CFIE26	CFIE25	CFIE24	CFIE23	CFIE22	CFIE21	CFIE20	CFIE19	CFIE18	CFIE17	CFIE16	CFIE15	CFIE14	CFIE13	CFIE12	CFIE11	CFIE10	CFIE9	CFIE8	CFIE7	CFIE6	CFIE5	CFIE4	CFIE3	CFIE2	CFIE1	CFIE0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F0	FDCAN_TXEFC	Res.	Res.	EFWM[5:0]						Res.	Res.	EFS[5:0]						EFSA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F4	FDCAN_TXEFS	Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	EFPI[4:0]					Res.	Res.	Res.	EFG[4:0]					Res.	Res.	EFFL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F8	FDCAN_TXEFA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0100	FDCAN_TTTMC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TME[6:0]							TMSA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0104	FDCAN_TTRMC	RMPS	XTD	Res.	RID[28:0]																												
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0108	FDCAN_TTOCF	Res.	Res.	Res.	Res.	Res.	Res.	EVTP	ECC	EGTF	AWL[7:0]							EECS	IRTO[6:0]							LDSDL[2:0]			TM	GEN	Res.	OM[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010C	FDCAN_TTMLM	Res.	Res.	Res.	Res.	ENTT[11:0]												Res.	Res.	Res.	Res.	TXEW[3:0]				CSS[1:0]		CCM[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 475. FDCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0110	FDCAN_TURCF	ELT	Res.	DC[13:0]														NCL[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114	FDCAN_TXBCIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKC	Res.	ESCN	NIG	TMG	FGP	GCS	TTIE	TMC[1:0]		RTIE	SWS[1:0]		SWP	ECS	SBC
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0118	FDCAN_TTGTP	CTP[15:0]																TP[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x011C	FDCAN_TTTMK	LCKM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TICC[6:0]							TM[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0120	FDCAN_TTIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CER	AW	WT	IWT	ELC	SE2	SE1	TXO	TXU	GTE	GTD	GTW	SWE	TTMI	RTMI	SOG	CSM	SMC	SBC
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0124	FDCAN_TTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERE	AWE	WTE	IWTE	ELCE	SE2E	SE1E	TXOE	TXUE	GTEE	GTDE	GTWE	SWEE	TTMIE	RTMIE	SOGE	CSME	SMCE	SBCE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0128	FDCAN_TTILS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERL	AWL	WTL	IWTL	ELCL	SE2L	SE1L	TXOL	TXUL	GTEL	GTDL	GTWL	SWEL	TTMIL	RTMIL	SOGL	CSML	SMCL	SBCL
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x012C	FDCAN_TTOST	SPL	WECS	AWE	WFE	GSI	TMP[2:0]			GFI	WGTD	Res.	Res.	Res.	Res.	Res.	Res.	RTO[7:0]								QCS	QGTP	SYS[1:0]		MS[1:0]		EL[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0130	FDCAN_TURNA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAV[17:0]																	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134	FDCAN_TTLGT	GT[15:0]																LT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0138	FDCAN_TTCTC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC[5:0]						CT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x013C	FDCAN_TTCTP	SWV[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCV[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0140	FDCAN_TTCSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSM[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0144 ～ 0x01FC	予約済み																																
	リセット値																																

表 475. FDCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0300	FDCAN_TTTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EVTSEL[1:0]		Res.	Res.	SWTSEL[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

56.5 CCU レジスタ

56.5.1 クロック較正ユニットコア開放レジスタ（CCU_CREL）

アドレスオフセット：0x0000

リセット値：0xrrrd dddd

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	d	d	d	d
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d

ビット 31:28 **REL**：コア開放
1 桁、BCD

ビット 27:24 **STEP**：コア開放のステップ
1 桁、BCD

ビット 23:20 **SUBSTEP**：コア開放のサブステップ
1 桁、BCD

ビット 19:16 **YEAR**：タイムスタンプの年
1 桁、BCD

ビット 15:8 **MON**：タイムスタンプの月
2 桁、BCD

ビット 7:0 **DAY**：タイムスタンプの日
2 桁、BCD

56.5.2 較正設定レジスタ（CCU_CCFG）

アドレスオフセット：0x0004

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDIV[3:0]			
rw												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OCPM[7:0]								CFL	BCC	Res.	TQBT[5:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31 **SWR**：ソフトウェアリセット

このビットに ‘1’ を書き込むと、較正 FSM が状態 Not_Calibrated にリセットされます（CSTAT.CALS = ‘00’）。較正ウォッチドッグ値 CWD.WDV もリセットされます。レジスタ CCFG、CSTAT および較正ウォッチドッグ設定 CWD.WDC は変更されません。リセットが完了するまで、ビットはセットされたままとなります。

「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

ビット 30:20 予約済み

ビット 19:16 **CDIV**：クロック分周回路

M_CAN 要件を満たしていることを保証するためにクロック較正がバイパス（BCC = ‘1’）される場合、クロック分周回路を設定する必要があります。

0000：1 分周

0001：2 分周

0010：4 分周

0011：6 分周

0100：8 分周

0101：10 分周

0110：12 分周

0111：14 分周

1000：16 分周

1001：18 分周

1010：20 分周

1011：22 分周

1100：24 分周

1101：26 分周

1110：28 分周

1111：30 分周

「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

ビット 15:8 **OCPM**：オシレータクロック周期最小

2 つの CAN ビット時間で最小周期を設定します。OCPM は基本較正で使用して、バスラインにグリッチがある場合の誤測定を回避します。設定される周期は OCPM × 32 です。この設定は周波数（80～500 MHz）および FDCAN1 と FDCAN2 で設定されたビットレート（125 kbit/s～1 Mbit/s）に依存します。2 つの CAN ビット時間よりわずかに下に値を設定することを推奨します。リセット値は、80 MHz fdcan_ker_ck および 1 Mbit/s CAN ビットレートで 1.6 ビット時間です。

「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

ビット 7 **CFL**：較正フィールド長

0：較正フィールド長は 32 ビットです。

1：較正フィールド長は 64 ビットです。

「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

ビット 6 **BCC**：バイパスクロック較正

このビットがセットされている場合、クロック入力 fdcan_ker_ck は CDIV 経由で設定可能なクロック分周回路を通じてタイムクォンタムクロックに送られます。cu_cok は常に ‘1’ です。この場合、接続された M_CAN のボーレートプリスケーラを設定して、M_CAN 内部タイムクォンタムクロックを生成する必要があります。

0：クロック較正ユニットがタイムクォンタムクロックを生成します。

1：クロック較正ユニットがバイパスされます（デフォルト設定）。

注： fdcan_ker_ck が 80 MHz 以上である限り、BCC = ‘1’ の場合でも CAN ユニットのクロック較正は機能します。較正状態を CSTAT レジスタから読み出すことができます。

ビット 5 予約済み

ビット 4:0 **TQBT**：ビット時間ごとのタイムクォンタム

ビット時間ごとのタイムクォンタムの数を設定します。FDCAN1 および FDCAN2 で設定される値と同様です。結果のタイムクォンタムクロック fdcan_tq_ck の範囲は、0.5 MHz（ビット時間ごとに 4 tq で 125 kbit/s のビットレート）～25 MHz（ビット時間ごとに 25 tq で 1 Mbit/s のビットレート）です。有効な値は 4～25 です。4 を下回る設定値は 4 として、25 を上回る値は 25 として解釈されます。

「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

56.5.3 較正ステータスレジスタ（CCU_CSTAT）

アドレスオフセット：0x0008

リセット値：0x0203 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALS[1:0]			TQC[11:0]											OCPC[17:16]	
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OCPC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:30 **CALS**：較正状態

00：Not_Calibrated

01：Basic_Calibrated

10：Precision_Calibrated

11：予約済み

ビット 29 予約済み

ビット 28:18 **TQC**：タイムクォンタムカウンタ

較正フィールド（32 または 64 ビット）でキャプチャされたタイムクォンタムの数です。クロック較正ユニットは状態 Precision_Calibrated の場合のみ有効です。

ビット 17:0 **OCPC**：オシレータクロック周期カウンタ

較正フィールド（32 または 64 ビット）でキャプチャされたオシレータクロック周期の数です。クロック較正ユニットは状態 Precision_Calibrated の場合のみ有効です。

56.5.4 較正ウォッチドッグレジスタ（CCU_CWD）

アドレスオフセット：0x000C

リセット値：0x0000 0000

較正 FSM が状態 Not_Calibrated（CSTAT.CALS = ‘00’）のとき、較正ウォッチドッグは最初の立ち下がりエッジの後に開始されます。この状態で、較正ウォッチドッグは受信したメッセージを監視します。較正ウォッチドッグが 0 までカウントダウンするまでにメッセージを受信しなかった場合、較正 FSM は状態 Not_Calibrated（CSTAT.CALS = ‘00’）のままとなり、カウンタが RWD.WDC で再ロードされ、基本較正は次の立ち下がりエッジの後に再開されます。

状態 Basic_Calibrated（CSTAT.CALS = ‘01’）の場合、較正ウォッチドッグは各受信メッセージで再開されます。較正ウォッチドッグが 0 までカウントダウンするまでにメッセージを受信しなかった場合、較正 FSM は状態 Not_Calibrated（CSTAT.CALS = ‘00’）に戻り、カウンタが RWD.WDC で再ロードされ、基本較正は次の立ち下がりエッジの後に再開されます。

クオーツメッセージを受信したとき、状態 Precision_Calibrated（CSTAT.CALS = ‘10’）に移行し、較正ウォッチドッグが再開されます。この状態で、較正ウォッチドッグはクオーツメッセージ受信入力を監視します。較正ウォッチドッグが 0 までカウントダウンするまでに、取り付けられた TTCAN によってクオーツ制御のノードからメッセージを受信しなかった場合、較正 FSM は状態 Basic_Calibrated（CSTAT.CALS = ‘01’）に再び遷移します。取り付けられた TTCAN の CAN プロトコルエンジンが開始されたとき（例：INIT ビットがリセットされたとき）、信号はアクティブになります。

較正ウォッチドッグイベントにより、割り込みフラグ CUIR.CWE もセットされます。CUIE.CWEE によって有効にされた場合、割り込みラインが有効化（ハイにセット）されます。割り込みラインは割り込みフラグ CUIR.CWE がリセットされるまでアクティブなままとなります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WDV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **WDV**：ウォッチドッグの値
実際の較正ウォッチドッグカウンタの値です。

ビット 15:0 **WDC**：WDC
ウォッチドッグの設定
較正ウォッチドッグカウンタの開始値です。‘00’ のリセット値で、カウンタは無効になります。
「P=書き込み保護」でマークされたレジスタ／ビットへのホスト CPU による書き込みアクセスは、M_CAN 制御ビット CCCR.CCE = ‘1’ かつ CCCR.INIT = ‘1’ の場合のみ可能です。

56.5.5 クロック較正ユニット割り込みレジスタ（CCU_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に ‘1’ を書き込むことによってクリアされます。’0’ を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。CUIE の設定は、割り込みが生成されるかどうかを制御します。

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSC	CWE
														rw	rw

ビット 31:2 予約済み

ビット 1 **CSC**：較正状態変更

0：較正状態は変更されていません。

1：較正状態が変更されました。

ビット 0 **CWE**：較正ウォッチドッグイベント

0：較正ウォッチドッグイベントはありません。

1：較正ウォッチドッグイベントが発生しました。

56.5.6 クロック較正ユニット割り込み有効レジスタ（CCU_IE）

アドレスオフセット：0x0014

リセット値：0x0000 0000

CU 割り込み有効レジスタの設定によって、CU 割り込みレジスタでのどのステータス変更が割り込みラインで信号送信されるかが決まります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSCE	CWEE
														rw	rw

ビット 31:2 予約済み

ビット 1 **CSCE**：較正状態変更有効化

0：割り込みは無効です。

1：割り込みは有効です。

ビット 0 **CWEE**：較正ウォッチドッグイベント有効化

0：割り込みは無効です。

1：割り込みは有効です。

56.5.7 CCU レジスタマップとリセット値の表

表 476. CCU レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	CCU_CREL	REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]				MON[7:0]								DAY[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0004	CCU_CCFG	SWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDIV[3:0]				CSM[7:0]								CFL	BCC	Res.	TQBT[4:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0008	CCU_CSTAT	CALS[1:0]		Res.	TQC[10:0]											OCPC[17:0]																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	CCU_CWD	WDV[15:0]																WDC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	CCU_IR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSC	CWE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	CCU_IR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSCE	CWEE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

57 USB On-The-Go ハイスピード（OTG_HS）

57.1 概要

Portions Copyright (C) 2004, 2005 Synopsys, Inc.掲載許可取得済み。

このセクションでは OTG_HS コントローラのアーキテクチャとプログラミングモデルについて説明します。

このセクションでは、次の略記が使用されます。

FS	フルスピード
LS	ロースピード
HS	ハイスピード
MAC	メディアアクセスコントローラ
OTG	On-The-Go
PFC	パケット FIFO コントローラ
PHY	物理層
USB	ユニバーサルシリアルバス
UTMI	USB 2.0 トランシーバマクロセルインタフェース（UTMI）
UTMI	USB トランシーバマクロセルインタフェース
ULPI	UTMI + ローピンインタフェース
LPM	リンク電源管理
BCD	バッテリ充電検出回路
HNP	ホストネゴシエーションプロトコル
SRP	セッションリクエストプロトコル

このセクションでは、参照文献として以下の文書が使用されています。

- USB On-The-Go Supplement, Revision 2.0
- Universal Serial Bus Revision 2.0 Specification
- USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007
- USB 2.0 ECN 用の正誤表：Link Power Management (LPM) - 7/2007
- バッテリ充電（BC）仕様 Revision 1.2

USB OTG は、デバイスとホストの両方の機能をサポートするデュアルロールデバイス（DRD）コントローラであり、*On-The-Go Supplement to the USB 2.0 Specification*に完全に準拠しています。USB OTG は、*USB 2.0 Specification に完全に準拠したホスト専用またはデバイス専用のコントローラとして設定することも可能です。*OTG_HS は以下の*表 477：サポートされる OTG_HS 速度*に定義される速度をサポートします。USB OTG は HNP と SRP の両方をサポートします。必要な外部デバイスは、OTG モードで使用される V_{BUS} 用のチャージポンプのみです。

表 477. サポートされる OTG_HS 速度

	HS（480 Mb/s）	FS（12 Mb/s）	LS（1.5 Mb/s）
ホストモード	X	X	X
デバイスモード	X	X	-

57.2 USB_OTG の主な機能

主な機能は、標準機能、ホストモード機能、およびデバイスモード機能の 3 つのカテゴリに分けて考えることができます。

57.2.1 標準機能

OTG_HS インタフェースの機能概要を次に示します。

- Universal Serial Bus Specification Rev 2.0 に準拠した USB-IF です。
- OTG HS は 次の 3 つの PHY インタフェースをサポートします。
  - フルスピード PHY を内蔵
  - 外部のフルスピード I^2C PHY を対象とする I^2C インタフェース
  - 外部のハイスピード PHY を対象とする ULPI インタフェース
- On-The-Go Supplement Rev 1.3 Specification に詳細が規定されているオプションの On-The-Go（OTG）プロトコルの完全サポート（PHY）を含みます。
  - A/B デバイス識別（ID ライン）を統合的にサポートしています。
  - ホストネゴシェーションプロトコル（HNP）およびセッションリクエストプロトコル（SRP）を統合的にサポートしています。
  - OTG アプリケーションでバッテリを節約するために、ホストで V_{BUS} をオフすることができます。
  - 内部コンパレータによって OTG に V_{BUS} のレベルを監視させることができます。
  - ホスト-ペリフェラルのダイナミックな役割切り替えをサポートします。
- 次の動作を行うようにソフトウェアでの設定が可能です。
  - SRP 対応の USB HS ペリフェラル（B デバイス）
  - SRP 対応の USB HS/LS ホスト（A デバイス）
  - USB On-The-Go フルスピードデュアルロールデバイス
- 以下の要素により HS SOF および LS Keep-alive をサポートします。
  - SOF パルス PAD 接続性
  - タイマ（TIMx）に対する SOF パルス内部接続
  - 設定可能なフレーム周期
  - 設定可能なフレームの終わりの割り込み
- OTG HS は、シェアホールディングサポートを持つ DMA を内蔵し、DMA モードで AHB バーストタイプをソフトウェアから選択できます。
- USB サスペンド中のシステム停止、デジタルコア、PHY、および DFIFO 電源管理回路内部のクロックドメインの電源オフなど、節電機能を内蔵しています。
- 以下のような高度な FIFO 制御を備えた専用の 4KB RAM を備えています。
  - RAM 空間を複数の FIFO に分割して、RAM を柔軟かつ効率的に使用できるように設定できます。
  - 各 FIFO は複数のパケットを保持できます。
  - 動的なメモリ割り当てが可能です。
  - FIFO を 2 のべき乗でないサイズに設定して、連続的なメモリ位置の使用が可能です。
- システムの介入なしで、最大 1 フレーム（1ms）の間、最大 USB バンド幅を保証します。
- バッテリ充電仕様 Revision 1.2 で説明されているとおり、充電ポート検出をサポートしています。

57.2.2 ホストモード機能

ホストモードでの OTG_HS インタフェースの主な機能と条件は次の通りです:

- V_{BUS} 電圧発生用の外部チャージポンプを持っています。
- 最大 16 個のホストチャネル（パイプ）を持っています。各チャネルはどんなタイプの USB 転送も割り当てられるように動的に設定可能です。
- 次を保持するハードウェアスケスケジューラを内蔵しています。
  - 最大 16 個の割り込みに加えて、周期的ハードウェアキュー内でのアイソクロナス転送リクエスト
  - 最大 16 つの制御に加えて、非周期的ハードウェアキュー内でのバルク転送リクエスト
- USB データ RAM の効率的な使用のために、共有 Rx FIFO、周期的 Tx FIFO、および非周期的 Tx FIFO を管理します。

57.2.3 ペリフェラルモード機能

ペリフェラルモードでの OTG_HS インタフェース機能は次の機能を持っています。

- 1 つの双方向コントロールエンドポイント 0 を持っています。
- 8 バルク、割り込み、またはアイソクロナス転送をサポートするために、設定可能な 5 つの IN エンドポイント（EP）を持っています。
- 8 バルク、割り込み、またはアイソクロナス転送をサポートするために、設定可能な 5 つの OUT エンドポイントを持っています。
- USB データ RAM の効率的な使用のために、共有 RX FIFO と Tx-OUT FIFO を管理します。
- アプリケーションの負荷を軽減するために、最大 9 つの専用 Tx-IN FIFO（有効な IN EP に対して 1 つずつ）を管理します。
- ソフト切断機能をサポートしています。

57.3 USB_OTG の実装

表 478. STM32H7の USB_OTG 実装(1)

USB の機能	OTG_HS1(2)	OTG_HS2(3)
デバイスの双方向エンドポイント（EP0 を含む）	9	
ホストモードチャネル	16	
専用 SRAM のサイズ	4 KB	
USB 2.0 リンク電源管理(LPM)のサポート	X	
サポートされている OTG レビジョン	2.0	
接続検出プロトコル（ADP）のサポート	-	
バッテリ充電検出（BCD）のサポート	X	
マルチプレクサ経由でプライマリ IO に使用できる ULPI	X	-

1. “X” はサポートされています。“-” はサポートされていません。
2. ハイスピード動作と互換性があります。
3. ハイスピード動作と互換性はありません。

57.4 USB OTG の機能詳細

57.4.1 USB OTG ブロック図

STM32H7 では、OTG_HS の 2 つのインスタンスが存在します（OTG_HS1 および OTG_HS2）。

両方ともハイスピード動作向けにプログラムできる可能性がありますが、OTG_HS1 のみ外部ハイスピードトランシーバを使用したハイスピード動作が可能になるアクセス可能な ULPI インタフェースを備えています。

参考資料

図 738. OTG ハイスピードブロック図（OTG_HS1）

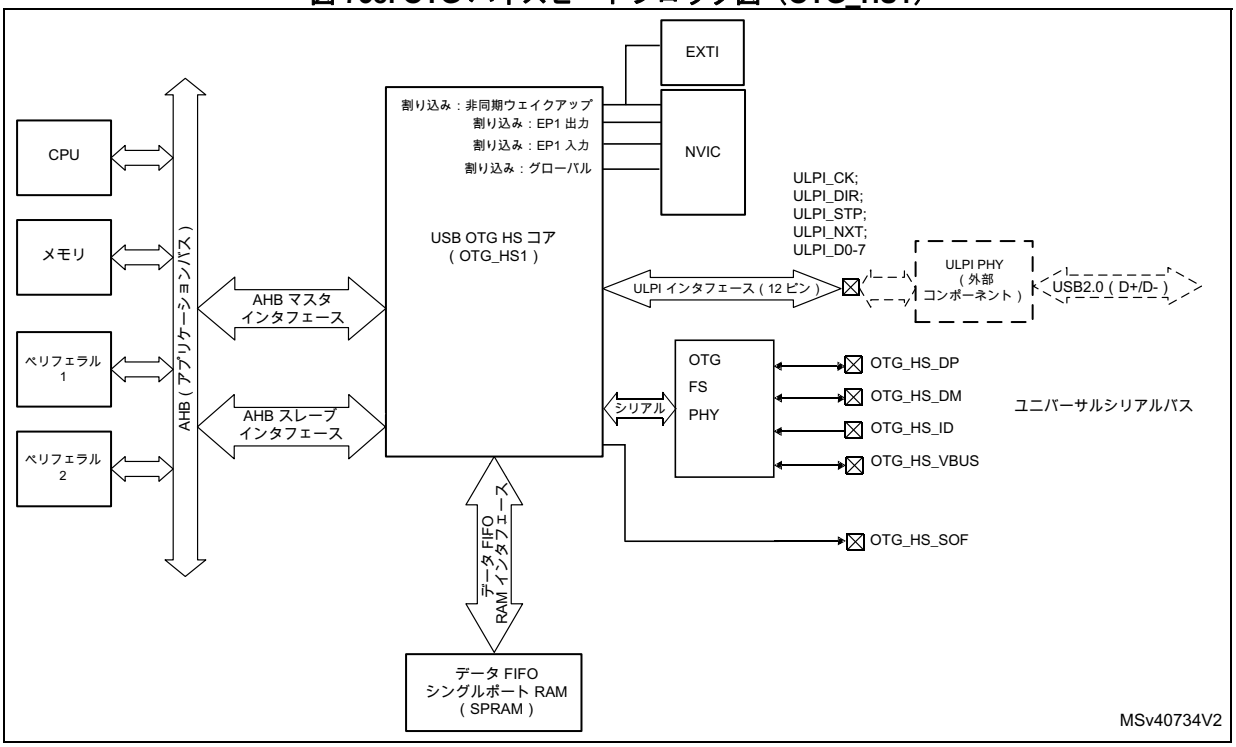
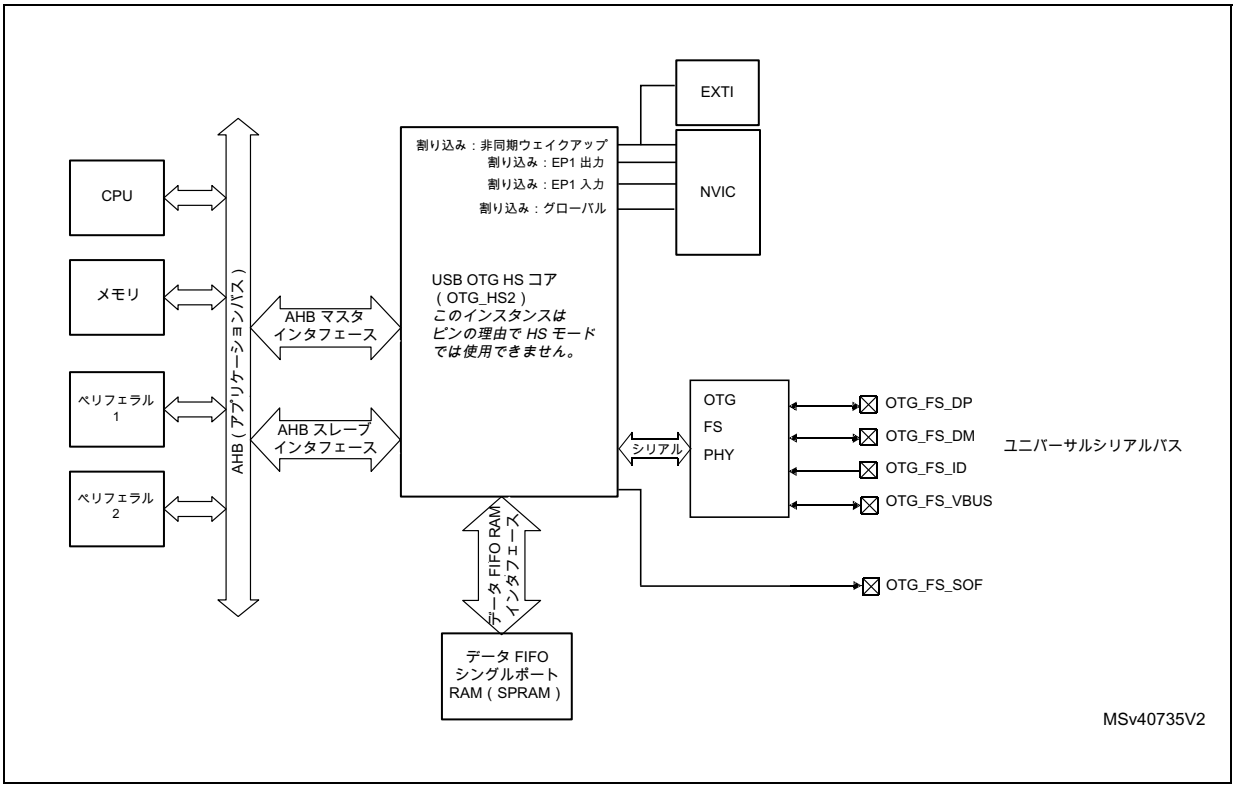


図 739. OTG ハイスピードブロック図（OTG_HS2）



57.4.2 USB OTG ピンおよび内部信号

表 479. OTG_FS/OTG_HS 入力／出力信号

信号名	信号タイプ	説明
usb_hclk	デジタル入力	USB OTG インタフェースクロック
usb_sof_evt	デジタル出力	オンチップペリフェラルの USB OTG フレーム開始イベント
usb_wkup	デジタル出力	USB OTG ウェイクアップイベント出力
usb_gbl_it	デジタル出力	USB OTG グローバル割り込み
usb_ep1_in_it	デジタル出力	USB OTG エンドポイント 1 入力割り込み
usb_ep1_out_it	デジタル出力	USB OTG エンドポイント 1 出力割り込み

表 480. OTG_FS/OTG_HS 入力／出力ピン

信号名	信号タイプ	説明
OTG_[HS/FS]_DP	デジタル入出力	USB OTG データ正ライン
OTG_[HS/FS]_DM	デジタル入出力	USB OTG データ負ライン
OTG_[HS/FS]_ID	デジタル入力	USB OTG ID
OTG_[HS/FS]_VBUS	デジタル双方向	USB OTG バスパワー
OTG_[HS/FS]_SOF	デジタル入力	GPIO の USB OTG フレーム開始イベント出力
OTG_HS_ULPI_CK	デジタル入力	USB OTG ULPI クロック
OTG_HS_ULPI_DIR	デジタル入力	USB OTG ULPI データバス方向制御
OTG_HS_ULPI_STP	デジタル出力	USB OTG ULPI データストリーム停止
OTG_HS_ULPI_NXT	デジタル入力	USB OTG ULPI 次のデータストリームリクエスト
OTG_HS_ULPI_D[0～7]	デジタル入出力	USB OTG ULPI 8 ビット双方向データバス

57.4.3 OTG コア

USB OTG は、外部水晶発振器を使用して、リセットおよびクロックコントローラ（RCC）から 48 MHz のクロックを受け取ります。USB クロックは、フルスピード（12 M ビット／秒）で 48 MHz ドメインを駆動するために使用され、OTG コアを設定する前に有効にされなければなりません。

CPU は、AHB ペリフェラルバスを使用して OTG コアレジスタに対する読み書きを実行します。USB イベントは 1 本の USB OTG 割り込みラインによって通知されます（*セクション 57.12：OTG_HS 割り込み*を参照）。

CPU は、専用の OTG 位置（プッシュレジスタ）に 32 ビットのワードを書き込むことによって、USB を介してデータを提供します。データは USB データ RAM 内に設定された Tx データ FIFO に自動的に格納されます。各 IN エンドポイント（ペリフェラルモードの場合）または OUT チャネル（ホストモードの場合）に対して 1 つの Tx FIFO プッシュレジスタがあります。

CPU は、専用の OTG アドレス（ポップレジスタ）から 32 ビットのワードを読み出すことによって、USB からデータを受け取ります。データは、4 KB の USB データ RAM 内に設定された共有 Rx FIFO から自動的に取得されます。各 OUT エンドポイントまたは IN チャネルに対して 1 つの Rx FIFO ポップレジスタがあります。

USB プロトコル層は、シリアルインタフェースエンジン（SIE）によって駆動され、オンチップ物理層（PHY）または I^2C インタフェースを使用した外部 OTG_HS PHY や外部 OTG_FS PHY 内のトランシーバモジュールによって USB 上で直列化（シリアライズ）されます。

57.4.4 内蔵フルスピード OTG PHY

フルスピード OTG PHY には、以下のコンポーネントが含まれます。

- ホストとデバイスの両方で使用される FS/LS トランシーバモジュール。シングルエンド USB ライン上で送受信を直接駆動します。
- A/B デバイス識別のための ID ラインをサンプリングするために使用される組み込みの ID プルアップ抵抗。
- デバイスの現在の役割に依存して OTG_HS コアによって制御される組み込みの DP/DM プルアップおよびプルダウン抵抗。ペリフェラルとして、DP プルアップ抵抗を有効にし、フルスピードペリフェラル接続を知らせます。信号を送るタイミングは、V_{BUS} が有効なレベルであると判断されてすぐです（B セッションが有効な場合）。ホストモードでは DP と DM の両方でプルダウン抵抗が有効にされます。プルアップ抵抗とプルダウン抵抗は、ペリフェラルの役割がホストネゴシエーションプロトコル（HNP）を通じて変更されたとき、動的に切り替えられます。
- プルアップ／プルダウン抵抗 ECN 回路。DP プルアップは、USB Rev2.0 に適用される抵抗の技術変更通知（Enginering Change Notice）に従って、OTG_HS から個別に制御される 2 つの抵抗で構成されます。DP プルアップの動的なトリミングは、ノイズの低減と Tx/Rx 信号品質の向上を可能にします。
- V_{BUS} レベル検出用コンパレータ。V_{BUS} の有効性、A-B セッションの有効性、およびセッション終了電圧閾値を検出するために使用されるヒステリシスを有します。これらは、セッションリクエストプロトコル（SRP）の駆動、有効な起動およびセッション終了条件の検出、および USB 操作中の V_{BUS} 電源の定常的な監視のために使用されます。

USB OTG HS ペリフェラルの正常動作を保証するためには AHB周波数を 30 MHz より高くする必要があります。

57.4.5 ハイスピード OTG PHY

USB OTG HS コアには、内部 HS PHY に接続される内部 ULPI インタフェースがあります。

USB OTG HS コアには、外部 HS PHY を接続する ULPI インタフェースがあります。

ULPI インタフェースは OTG HS1 でのみ使用できます（*図 738：OTG ハイスピードブロック図（OTG_HS1）*を参照）。

57.4.6 I^2C インタフェースを使用する外部フルスピード OTG PHY

USB OTG HS コアは、外部 FS PHY を接続する I^2C インタフェースを内蔵しています。

参考資料

57.5 OTG デュアルロールデバイス（DRD）

57.5.1 ID ラインの検出

ホストかペリフェラル（デフォルト）かの役割は、ID 入力ピンに応じて割り当てられます。ID ラインのステータスは、USB ケーブルでの挿入時に、MicroA または MicroB プラグがマイクロ AB レセプタクルに接続されたかによって決まります。

- USB ケーブルの B 側がフローティング ID ワイヤに接続された場合、組み込みのプルアップ抵抗が ID ラインのハイレベルを検出して、デフォルトのペリフェラルの役割が確立されます。この設定では、OTG_HS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。
- USB ケーブルの A 側がアース付き ID ラインに接続された場合、OTG_HS は、ホストのソフトウェア初期化のために ID ラインステータス変化割り込み（OTG_GINTSTS の CIDSCHG ビット）を発行して、自動的にホストの役割に切り替わります。この設定では、OTG_HS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。

57.5.2 HNP デュアルロールデバイス

グローバル USB 設定レジスタの HNP 対応ビット（OTG_ GUSBCFG の HNPCAP ビット）は、OTG_HS コアがホストネゴシエーションプロトコル（HNP）に従って、役割を A ホストから A ペリフェラルに（またその逆）、または B ペリフェラルから B ホスト（またその逆）に動的に変更できるようにします。現在のデバイスのステータスは、グローバル OTG 制御およびステータスレジスタのコネクタ ID ステータスビット（OTG_GOTGCTL の CIDSTS ビット）とグローバル割り込みおよびステータスレジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）の値の組み合わせによって読み出すことができます。

HNP プログラミングモデルの詳細は、*セクション 57.15 : OTG_HS プログラミングモデル*に記載されています。

57.5.3 SRP デュアルロールデバイス

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_HS コアが A デバイスの節電のために V_{BUS} の生成を停止できるようにします。A デバイスは OTG_HS の役割がホストかペリフェラルかに関係なく、常に、V_{BUS} を駆動する役割を持っていることに注意してください。

SRP A/B デバイスのプログラミングモデルの詳細は、*セクション 57.15: OTG_HS プログラミングモデル*に記載されています。

57.6 USB ペリフェラル

このセクションでは、USB ペリフェラルモードでの OTG_HS の機能について説明します。OTG_HS は、以下の環境で USB ペリフェラルとして機能します。

- OTG B ペリフェラル
  - USB ケーブルの B 側が挿入された場合の OTG B デバイスのデフォルトの状態
- OTG A ペリフェラル
  - HNP が OTG_HS の役割をペリフェラルに切り替えた後の OTG A デバイスの状態
- Bデバイス
  - ID ラインが存在し、機能していて、USB ケーブルの B 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。
- ペリフェラルのみ
  - *セクション 57.14.4：OTG USB 設定レジスタ（OTG_GUSBCFG）*の強制デバイスモードビット（FDMOD）が 1 にセットされると、OTG_HS コアは USB ペリフェラル専用として機能するよう強制されます。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。

注： ***B デバイスまたはペリフェラル専用設定の場合に電源がバスから供給されるデバイスを実装するには、V_{DD} チップ供給（V_{BUS}からの）を生成する外部レギュレータを追加する必要があります。***

57.6.1 SRP 対応ペリフェラル

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_HS コアがセッションリクエストプロトコル（SRP）をサポートできるようにします。この方法により、USB セッションがサスペンド状態にある間、V_{BUS} の供給を停止して、リモート A デバイスの消費電力を節約できます。

SRP ペリフェラルモードのプログラミングモデルの詳細は、*B デバイスセッションリクエストプロトコル* のセクションに記載されています。

57.6.2 ペリフェラルの状態

電源が供給された状態

V_{BUS} 入力は、USB ペリフェラルが電源供給状態に入ることが許される B セッション有効電圧を検出します（USB 2.0 のセクション 9.1 を参照）。その場合、OTG_HS は、DP プルアップ抵抗を自動的に接続して、フルスピードデバイス接続をホストに知らせ、セッションリクエスト割り込みを生成して（OTG_GINTSTS の SRQINT ビット）、電源供給状態を通知します。

V_{BUS} 入力は、また、USB 動作中にホストによって適切な V_{BUS} レベルが供給されることを保証します。V_{BUS} 電源が B セッションの有効値以下に低下したことが検出された場合（たとえば、電源の変動があった場合や、ホストポートの電源がオフとなった場合など）、OTG_HS は自動的に切断され、セッション終了検出割り込み（OTG_GOTGINT の SEDET ビット）が生成されて、OTG_HS が電源供給状態でなくなったことを通知します。

電源供給状態では、OTG_HS は、ホストからリセットシグナリングを受け取ることを期待しています。他の USB 操作を実行することはできません。リセットシグナリングを受け取ると、リセット検出割り込み（OTG_GINTSTS の USBRST ビット）が生成されます。リセットシグナリングが完了すると、エミュレーション終了割り込み（OTG_GINTSTS の ENUMDNE ビット）が生成され、OTG_HS はデフォルト状態に入ります。

ソフト切断

電源供給状態は、ソフト切断機能によってソフトウェアで終了させることができます。DP プルアップ抵抗は、デバイス制御レジスタのソフト切断ビット（OTG_DCTL の SDIS ビット）をセットすることによって取り除かれ、USB ケーブルが実際にはホストポートから取り除かれていなくても、ホスト側にデバイス切断検出割り込みを生成します。

デフォルト状態

デフォルト状態では、OTG_HS はホストから SET_ADDRESS コマンドを受け取ることを期待しています。他の USB 操作を実行することはできません。有効な SET_ADDRESS コマンドが USB 上でデコードされると、アプリケーションは対応する番号をデバイス設定レジスタのデバイスアドレスフィールドに書き込みます（OTG_DCFG の DAD ビット）。その場合、OTG_HS はアドレス状態に入り、設定された USB アドレスでホストトランザクションに応答する準備が整います。

サスペンド状態

OTG_HS ペリフェラルは、USB の動作状態を定常的に監視します。USB のアイドル状態が 3 ms 続くと、アーリーサスペンド割り込み（OTG_GINTSTS の ESUSP ビット）が発行され、適切な場合は 3 ms 後にサスペンド割り込み（OTG_GINTSTS の USBSUSP ビット）によって確認されます。その場合、デバイスステータスレジスタのデバイスサスペンドビット（OTG_DSTS の SUSPSTS ビット）が自動的にセットされ、OTG_HS はサスペンド状態に入ります。

サスペンド状態は、デバイス自身によって終了することもできます。この場合、アプリケーションはデバイス制御レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）をセットし、1～15 ms 後にクリアします。

ホストからレジュームシグナリングが検出されると、レジューム割り込み（OTG_GINTSTS の WKUPINT ビット）が生成され、デバイスサスペンドビットは自動的にクリアされます。

57.6.3 ペリフェラルエンドポイント

OTG_HS コアは、次の USB エンドポイントをインスタンス化します。

- コントロールエンドポイント 0 :
  - 双方向であり、制御メッセージのみを取り扱います。
  - 入力および出力トランザクションを処理する個別のレジスタセットを備えています。
  - 固有の制御レジスタ（OTG_DIEPCTL0/OTG_DOEPCTL0）、転送設定レジスタ（OTG_DIEPTSIZ0/OTG_DOEPTSIZ0）、およびステータス割り込みレジスタ（OTG_DIEPINT0/OTG_DOEPINT0）を備えています。制御および転送サイズレジスタ内で使用可能なビットのセットは、他のエンドポイントとは少し違います。
- 8 IN エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割り込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DIEPCTL*x*）、転送設定レジスタ（OTG_DIEPTSIZ*x*）、およびステータス割り込みレジスタ（OTG_DIEPINT*x*）を備えています。
  - デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）を使用して、すべての IN エンドポイント（EP0 も含む）上で 1 種類のエンドポイント割り込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR ビット）をサポートします。この割り込みは、周期的フレーム割り込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。

- 8 OUT エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割り込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DOEPCTL*x*）、転送設定レジスタ（OTG_DOEPTSIZ*x*）、およびステータス割り込みレジスタ（OTG_DOEPINT*x*）を備えています。
  - デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK）を使用して、すべての OUT エンドポイント（EP0 も含む）上で 1 種類のエンドポイント割り込みソースを有効／無効にすることができます。
  - 現在のフレーム内で転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス OUT 転送割り込み（OTG_GINTSTS の INCOMPISOOUT ビット）をサポートします。この割り込みは、周期的フレーム割り込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。

エンドポイント制御

- 以下のエンドポイント制御は、デバイスエンドポイント-*x* IN/OUT 制御レジスタ（OTG_DIEPCTL*x*/OTG_DOEPCTL*x*）を通じて、アプリケーションから使用できます。
  - エンドポイント有効／無効
  - 現在の設定でのエンドポイントのアクティブ化
  - USB 転送タイプ（アイソクロナス、バルク、割り込み）のプログラミング
  - サポートされるパケットサイズのプログラミング
  - IN エンドポイントに関連する Tx FIFO 番号のプログラミング
  - 期待される、または送信されるデータ 0／データ 1 PID（バルク／割り込みのみ）のプログラミング
  - トランザクションが送受信される偶数／奇数フレームのプログラミング（アイソクロナス転送のみ）
  - オプションで、FIFO のステータスに関係なく、常にホストに否定応答する NAK ビットのプログラミング
  - オプションで、エンドポイントに対するホストトークンを常に停止する STALL ビットのプログラミング
  - オプションで、OUT エンドポイントが受信データの CRC フィールドをチェックしないようにする SNOOP モードのプログラミング

エンドポイント転送

アプリケーションは、デバイスエンドポイント-*x* 転送サイズレジスタ（OTG_DIEPTSIZ*x*/OTG_DOEPTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。エンドポイント制御レジスタのエンドポイントイネーブルビットをセットする前にプログラミングを行う必要があります。エンドポイントが有効になると、これらのフィールドは、OTG_HS コアが現在の転送ステータスで更新すると、読み出し専用になります。

以下の転送パラメータがプログラム可能です。

- 転送サイズ（バイト単位）
- 全体の転送サイズを構成するパケットの数

エンドポイントのステータス／割り込み

デバイスエンドポイント-*x* 割り込みレジスタ（OTG_DIEPINT*x*/OTG_DOPEPINT*x*）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割り込みレジスタの OUT エンドポイント割り込みビットまたは IN エンドポイント割り込みビット（OTG_GINTSTS の OEPINT ビット、または OTG_GINTSTS の IEPINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、デバイスオールエンドポイント割り込み（OTG_DAINT）レジスタを読み出して、デバイスエンドポイント-*x* 割り込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

ペリフェラルコアは、以下のステータスチェックおよび割り込み生成を行います。

- データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割り込み
- セットアップステージが終了しているかどうか（制御 OUT のみ）
- 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
- NAK 応答がホストに送信されたかどうか（アイソクロナス IN のみ）
- Tx FIFO がエンプティのとき IN トークンが受け取られたかどうか（バルク IN/割り込み IN のみ）
- エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったかどうか
- バブルエラー条件が検出されたかどうか
- アプリケーションによるエンドポイントの無効化が可能かどうか
- アプリケーションによるエンドポイント NAK が可能かどうか（アイソクロナス IN のみ）
- 3 つを超える連続したセットアップパケットが受信されたかどうか（コントロール OUT のみ）
- タイムアウト条件が検出されたかどうか（コントロール IN のみ）
- アイソクロナス OUT パケットが割り込みを生成せずにドロップされたかどうか

57.7 USB ホスト

このセクションでは、USB ホストモードでの OTG_HS の機能について説明します。OTG_HS は次の環境で USB ホストとして機能します。

- OTG A ホスト
  - USB ケーブルの A 側が挿入されたときの OTG A デバイスのデフォルト状態
- OTG B ホスト
  - HNP がホストの役割に切り替わった後の OTG B デバイス
- A デバイス
  - ID ラインが存在し、機能していて、USB ケーブルの A 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。組み込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。
- ホストのみ
  - *57.14.4* グローバル USB 設定レジスタの強制ホストモードビット（OTG_GUSBCFG の FHMOD ビット）は、OTG_HS コアを強制的に USB ホスト専用として機能させます。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。組み込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。

注： ***オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力からでも駆動できます。これは、OTG A ホスト、A デバイス、およびホスト専用設定の場合に必要です。***

57.7.1 SRP 対応ホスト

SRP のサポートはグローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）を通じて使用可能です。SRP 機能が有効なときには、ホストは、USB セッションがサスペンド状態にある間、V_{BUS} の電力供給を停止することによって、消費電力を節約できます。

SRP ホストモードのプログラミングモデルの詳細は、*A デバイスセッションリクエストプロトコル* のセクションに記載されています。

57.7.2 USB ホストの状態

ホストポートパワー

オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力からでも駆動できます。アプリケーションが選んだ GPIO を使用して V_{BUS} に電源を供給することにした場合は、ホストポート制御およびステータスレジスタのポートパワービット（OTG_HPRT の PPWR ビット）もセットする必要があります。

有効な V_{BUS}

HNP または SRP が有効化された場合、VBUS センシングピンは V_{BUS}に接続する必要があります。V_{BUS} 入力は、USB 動作中にチャージポンプによって適切な V_{BUS} レベルが供給されることを保証します。予期せぬ V_{BUS} 電圧の低下（V_{BUS} の有効閾値 4.4 V 以下へ）が発生すると、セッション終了検出ビット（OTG_GOTGINT の SEDET ビット）によって OTG 割り込みがトリガされます。その場合、アプリケーションは V_{BUS} 電源供給を停止し、ポートパワービットをクリアする必要があります。

HNP と SRP が共に無効化された場合、VBUS センシングピンを V_{BUS} に接続する必要はなく、GPIO として使用できます。

チャージポンプの過電流フラグは、電気的ダメージを防止するためにも使用できます。チャージポンプからの過電流フラグ出力を任意の GPIO 入力に接続して、有効レベルになったときにポート割り込みを生成するように設定します。過電流 ISR は迅速に V_{BUS} の発生を無効にし、ポートパワービットをクリアする必要があります。

ペリフェラル接続のホスト検出

SRP または HNP を有効化し、かつ USB ペリフェラルまたは B デバイスを何時でも接続できる場合でも、V_{BUS} が有効レベル（5 V）で検出できなくなるまで、OTG_HS はバス接続を検出しません。V_{BUS} が有効なレベルにあり、リモート B デバイスが接続されると、OTG_HS コアは、ホストポート制御およびステータスレジスタのデバイス接続ビット（OTG_HPRT の PCDET ビット）によってトリガされるホストポート割り込みを発行します。

HNP と SRP が共に切り離されたとき、USB ペリフェラルまたは B デバイスが接続されると直ちに検出されます。OTG_HS コアは、ホストポート制御およびステータス内のデバイス接続ビット（OTG_HPRT の PCDET ビット）から発生するホストポート割り込みを発行します。

ペリフェラル切断のホスト検出

ペリフェラル切断イベントは、切断検出割り込み（OTG_GINTSTS の DISCINT ビット）をトリガします。

ホストのエニュメレーション

ペリフェラル接続を検出した後、ホストは、USB リセットおよび設定コマンドを新しいペリフェラルに送信することによって、エニュメレーションプロセスを開始しなければなりません。

USB リセットの駆動を開始する前に、アプリケーションは、デバウンス終了ビット（OTG_GOTGINT の DBCDNE ビット）によってトリガされる OTG 割り込みを待ちます。これは、DP（FS）または DM（LS）上のプルアップ抵抗の接続によって生じた電気的デバウンスの後、バスが再び安定状態に戻ったことを示します。

アプリケーションは、ホストポート制御およびステータスレジスタのポートリセットビット（OTG_HPRT の PRST ビット）を最短 10 ms、最長 20 ms の間セット状態に保つことによって、USB を介した USB リセットシグナリング（シングルエンド、ゼロ）を駆動します。アプリケーションは、タイミングカウントをチェックした後、ポートリセットビットがクリアされているかどうかを確認します。

USB リセットシーケンスが完了すると、ポートイネーブル／ディセーブル変更ビット（OTG_HPRT の PENCHNG ビット）によってホストポート割り込みがトリガされます。これにより、エニュメレーションされたペリフェラルの速度をホストポート制御およびステータスレジスタのポート速度フィールド（OTG_HPRT の PSPD ビット）から読み出せることと、ホストが SOF（FS）またはキープアライブ（LS）の駆動を開始しようとしていることがアプリケーションに通知されます。これで、ホストは、ペリフェラル設定コマンドを送信することによってペリフェラルエニュメレーションを完了する準備ができたことになります。

ホストのサスペンド

アプリケーションは、ホストポート制御およびステータスレジスタのポートサスペンドビット（OTG_HPRT の PSUSP ビット）をセットすることによって、USB の活動をサスペンド（一時停止）します。OTG_HS コアは SOF の送信を停止し、サスペンド状態に入ります。

サスペンド状態は、リモートデバイスが主導して（リモートウェイクアップ）、終了することもできます。この場合、ホストポート制御およびステータスレジスタのポートレジュームビット（OTG_HPRT の PRES ビット）セルフセットが検出されると、リモートウェイクアップ割り込み（OTG_GINTSTS の WKUPINT ビット）が生成され、USB によってレジュームシグナリングが自動的に駆動されます。アプリケーションは、レジュームウィンドウの時間を決めてから、ポートレジュームビットをクリアしてサスペンド状態を終了し、SOF をリスタートする必要があります。

サスペンド状態をホスト主導で終了した場合、アプリケーションは、ポートレジュームビットをセットして、ホストポートでのシグナリングのレジュームを開始し、レジュームウィンドウの時間を決め、最後に、ポートレジュームビットをクリアする必要があります。

57.7.3 ホストチャネル

OTG_HS コアは、16 つのホストチャネルをインスタンス化します。各ホストチャネルは USB ホスト転送をサポートします（USB パイプ）。ホストは、同時に 16 つを超える転送リクエストをサポートすることはできません。アプリケーションからの 16 つを超える転送リクエストがペンディング中の場合、ホストコントローラドライバ（HCD）は、チャネルが以前のデューティから使用可能になったとき、すなわち、転送完了およびチャネルホールト割り込みの受信後、チャネルの再割当てを行う必要があります。

各ホストチャネルは、入力／出力および任意のタイプの周期的/非周期的トランザクションをサポートするように設定できます。各ホストチャネルは、適切な制御レジスタ（OTG_HCCHAR*x*）、転送設定レジスタ（OTG_HCTSIZ*x*）、およびステータス／割り込みレジスタ（OTG_HCINT*x*）と関連するマスクレジスタ（OTG_HCINTMSK*x*）を使用します。

ホストチャネル制御

- アプリケーションは、ホストチャネル *x* 特性レジスタ（OTG_HCCHAR*x*）を通じて、以下のホストチャネル制御を使用できます。
  - チャネルの有効化／無効化
  - ターゲット USB ペリフェラルの HS/FS/LS 速度のプログラミング
  - ターゲット USB ペリフェラルのアドレスのプログラミング
  - ターゲット USB ペリフェラルのエンドポイント数のプログラミング
  - IN/OUT 転送方向のプログラミング
  - USB 転送タイプ（コントロール、バルク、割り込み、アイソクロナス）のプログラミング
  - 最大パケットサイズ（MPS）のプログラミング
  - 奇数／偶数フレームで実行される周期的転送のプログラミング

ホストチャネル転送

アプリケーションは、ホストチャネル転送サイズレジスタ（OTG_HCTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。プログラミングは、ホストチャネル特性レジスタのチャネルイネーブルビットをセットする前に行う必要があります。エンドポイントが有効になると OTG_HS コアが現在の転送ステータスに従って更新するので、パケットカウントフィールドは読み出し専用になります。

- 以下の転送パラメータがプログラム可能です。
  - 転送サイズ（バイト単位）
  - 全体の転送サイズを構成するパケットの数
  - 初期データ PID

ホストチャネルのステータス／割り込み

ホストチャネル *x* 割り込みレジスタ（OTG_HCINT*x*）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、ホストオールチャネル割り込み（OTG_HAINT）レジスタを読み出して、ホストチャネル x 割り込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。各チャネルの各割り込みソースのマスクビットも OTG_HCINTMSKx レジスタで使用可能です。

- ホストコアは、以下のステータスチェックおよび割り込み生成を行います。
  - データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割り込み
  - 転送完了、USB トランザクションエラー、またはアプリケーションからの無効化コマンドのためにチャネルが停止したかどうか
  - 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
  - ACK 応答を受け取った時
  - NAK 応答を受け取った時
  - STALL 応答を受け取った時
  - CRC エラー、タイムアウト、ビットスタッフエラー、偽 EOP による USB トランザクションエラー
  - バブルエラー
  - フレームオーバーラン
  - データトグルエラー

57.7.4 ホストスケジューラ

ホストコアは、アプリケーションによってポストされたUSBトランザクションリクエストを自律的に並べ替えて、管理する組み込みハードウェアスケジューラを備えています。各フレームの開始時に、ホストは、周期的（アイソクロナスおよび割り込み）トランザクションを最初に実行し、次に、非周期的（コントロールおよびバルク）トランザクションを行うことによって USB の仕様に従って、アイソクロナスおよび割り込み転送タイプを優先します。

ホストは、リクエストキュー（1 つは周期的転送のキュー、もう 1 つは非周期的転送のキュー）を通じて USB トランザクションを処理します。各リクエストキューは、最大 8 個のエントリを保持できます。各エントリはアプリケーションからのペンディング中のトランザクションリクエストを表し、IN または OUT チャネル番号と USB 上でトランザクションを実行するためのその他の情報を保持しています。リクエストがキューに書き込まれた順序によって USB インタフェース上のトランザクションの順序が決まります。

各フレームの先頭で、ホストは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。現在のフレームに対してスケジュールされたアイソクロナスまたは割り込みトランザクションが現在のフレームの終了時点でまだペンディング中である場合、ホストは、不完全周期転送割り込み（OTG_GINTSTS の IPXFR ビット）を発行します。OTG_HS コアは周期的と非周期的リクエストキューの管理を行います。周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）と非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）は読み出し専用レジスタであり、アプリケーションから各リクエストキューのステータスを読み出すときに使うことができます。これらのレジスタには次が含まれます。

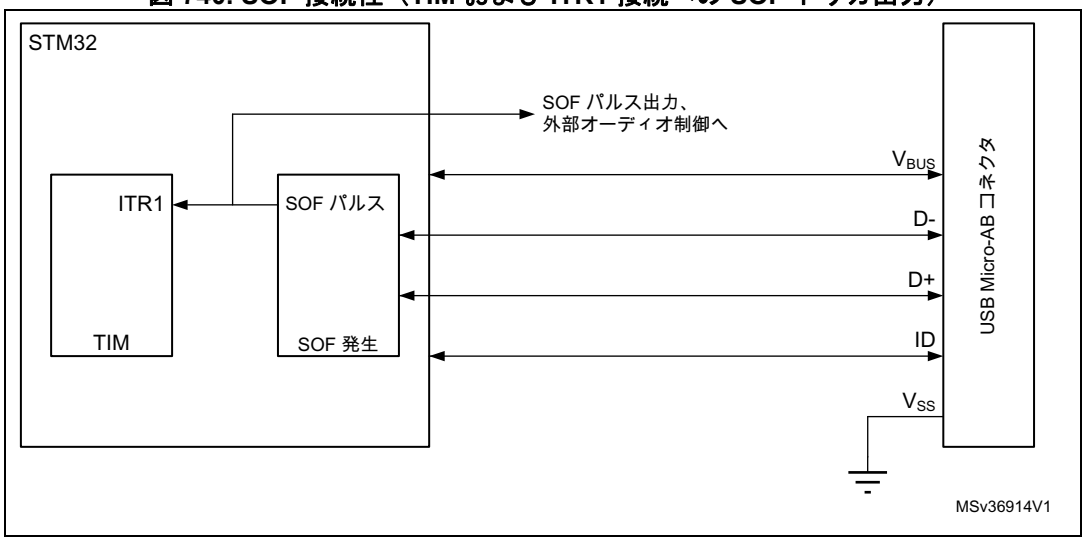
- 周期（非周期）リクエストキューで現在使用可能なフリーエントリの数（最大 8 エントリ）。
- 周期（非周期）Tx FIFO（OUT トランザクション）で現在使用可能な空きスペース
- IN/OUT トークン、ホストチャネル番号、およびその他のステータス情報

リクエストキューはそれぞれ最大 8 つのエントリを保持できるので、アプリケーションは、最大 8 つのペンディング中の周期トランザクションに加えて、8 つのペンディング中の非周期トランザクションについて、SB に物理的に到達する前に、ホストトランザクションを事前にスケジュールできます。

ホストスケスケジューラ（キュー）へトランザクションリクエストをポストするときは、アプリケーションは OTG_HNPTXSTS レジスタの PTXQSAV ビット、または OTG_HNPTXSTS レジスタの NPTQXSAV ビットを読み出して、周期的（非周期的）リクエストキュー内に少なくと 1 つのエントリが存在することを確認する必要があります。

57.8 SOF トリガ

図 740. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力）



OTG_HS コアは、ホスト、ペリフェラル、SOF パルス出力接続機能内で SOF フレーミングを監視、追跡、設定する手段を提供します。

これらのユーティリティは、オーディオペリフェラルが PC によって提供されるアイソクロナスストリームに同期する必要がある場合や、ホストがオーディオペリフェラルの要件に従ってフレーミング速度をトリム（微調整）する必要がある場合など、適応型オーディオクロック生成技術にとって特に役立ちます。

57.8.1 ホストの SOF

ホストモードでは、2 つの連続した SOF（HS/FS）またはキープアライブ（LS）トークンの生成の間に発生する PHY クロックの数は、ホストフレームインターバルレジスタ（HFIR）でプログラム可能であり、それにより、SOF フレーミング周期中にアプリケーションによる制御が可能です。割り込みは、どのフレームの開始時点でも生成されます（OTG_GINTSTS の SOF ビット）。現在のフレーム番号と次のSOF までの残り時間が、ホストフレーム番号レジスタ（HFNUM）で追跡されます。

SOF パルス信号は、任意の SOF 開始トークンで生成され 12 システムクロックサイクルの幅です。SOF パルスは、タイマの入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

57.8.2 ペリフェラルの SOF

デバイスモードでは、フレーム割り込みの開始は、SOF トークンが USB 上で受信されるたびに生成されます（OTG_GINTSTS の SOF ビット）。対応するフレーム番号は、デバイスステータスレジスタ（OTG_DSTS の FNSOF ビット）から読み出すことができます。12 システムクロックサイクルの幅を持つ SOF パルス信号も生成されます。SOF パルス信号は、TIM 入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

周期的フレーム割り込みの終わり（OTG_GINTSTS の EOPF）は、デバイス設定レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）に応じて、タイムフレームインターバルの 80 %、85 %、90 % または 95 % が経過した時点をアプリケーションに通知するために使用されます。この機能を使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

57.9 電力オプション

OTG PHY の消費電力は、サポートされる OTG リビジョンに応じて、一般コア設定レジスタの 2 つまたは 3 つのビットによって制御されます。

- PHY パワーダウン（OTG_GCCFG の PWRDWN）

  PHY のフルスピードトランシーバモジュールのオン／オフを切り替えます。事前にセットして、すべての USB 動作を可能にする必要があります。

- V_{BUS} 検出イネーブル（OTG_GCCFG の VBDEN）

  OTG の動作に関連する V_{BUS} レベル検出用コンパレータのオン／オフを切り替えます。

USB セッションがまだ有効でないか、デバイスが切断されているときには、USB サスペンド状態で節電テクニックを使用できます。

- PHY クロックを停止します（OTG_PCGCCTL の STPPCLK ビット）。

  クロックゲーティング制御レジスタのストップ PHY クロックビットをセットすると、OTG 全速コア内部の 48 MHz クロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。USB クロック切り替え機能による動的な電力消費は、48 MHz クロック入力がアプリケーションによって動作状態を維持された場合でも節減されます。

  ほとんどのトランシーバも無効になり、非同期レジュームやリモートウェイクアップイベントの検出を担当する部分だけが動作状態を維持します。

- ゲート HCLK（OTG_PCGCCTL の GATEHCLK ビット）

  クロックゲーティング制御レジスタのゲート HCLK ビットをセットすると OTG_HS コア内部のシステムクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。レジスタ読み出しおよび書き込みインタフェースのみが動作状態を維持します。USB クロック切り替え機能による動的な電力消費は、システムクロックが他の目的のためにアプリケーションによって動作状態を維持されている場合でも節減されます。

- USB システムの停止

  OTG_HS が USB サスペンド状態にある場合、アプリケーションはシステム内のすべてのクロックソースを完全にシャットダウンすることによって、電力消費全体を大幅に削減する方法を取ることができます。USBシステムの停止は、まず、ストップ PHY クロックビットをセットし、次に、パワー制御システムモジュール（PWR）でシステムディープスリープモードを設定することによって有効になります。

  OTG_HS コアは、USB 上でリモートウェイクアップ（ホストとして）またはリモートレジューム（デバイスとして）信号の非同期検出によって、システムと USB クロックの両方を自動的に再起動します。

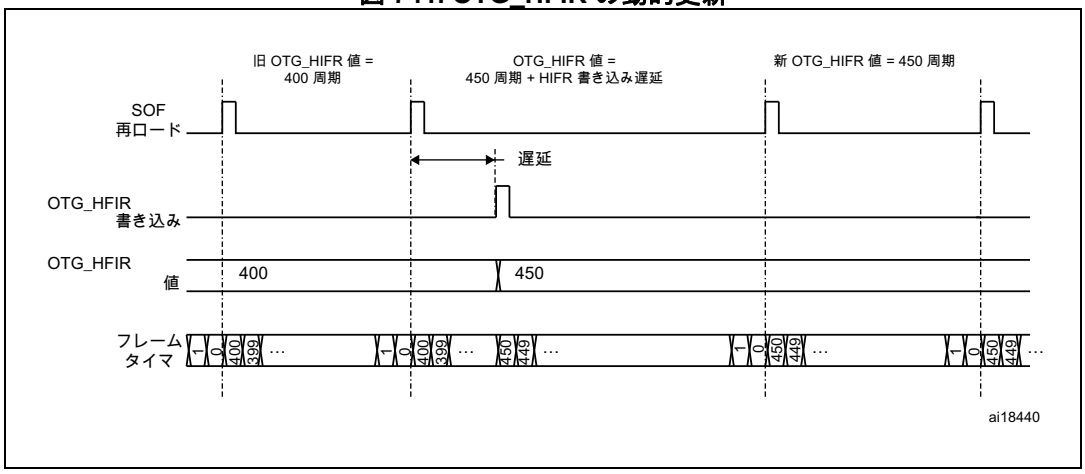
動的消費電力を節減するために、USB データ FIFO は、OTG_HS コアによってアクセスされたときだけクロック駆動されます。

57.10 OTG_HFIR レジスタの動的更新

USB コアはホストモードでマイクロ SOF フレーミング周期を動的にトリミングする機能を内蔵しています。この機能を使うと、外部デバイスをマイクロ SOF フレームと同期させることができます。

現在のマイクロ SOF フレーム内で OTG_HFIR レジスタを変更すると、*図 741* で説明するように次のフレーム SOF 周期補正が適用されます。

図 741. OTG_HFIR の動的更新

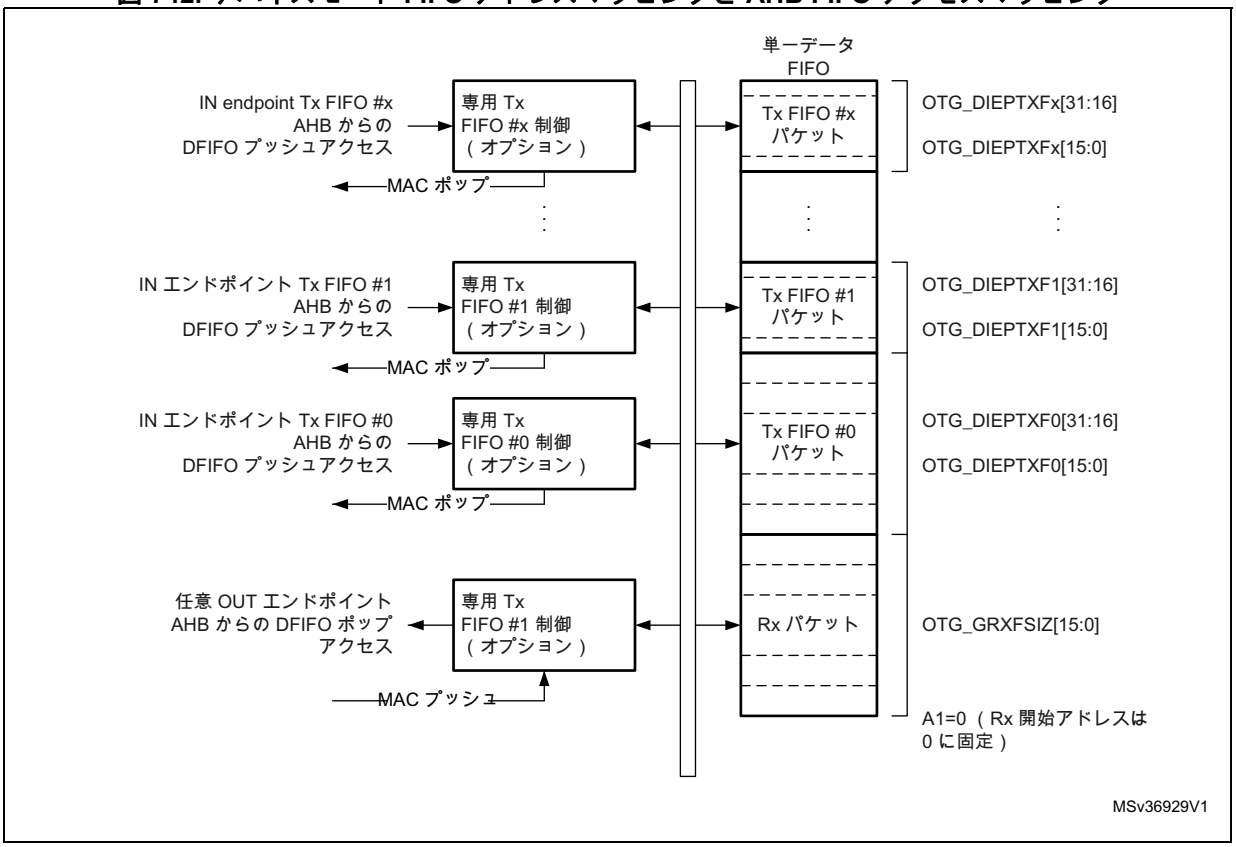


57.11 USB データ FIFO

USB システムは、高度な FIFO 制御メカニズムを持つ 4 KB の専用 RAM を備えています。OTG_HS コアのパケット FIFO コントローラモジュールは、RAM 空間を、アプリケーションが USB 転送の前に一時的にデータを格納する（プッシュする）Tx FIFO と、USB から受信したデータをアプリケーションが取得する（ポップする）前に一時的に格納される Rx FIFO に編成します。割り当てられる FIFO の数と、これらが RAM の内部でどのように構成されるかは、デバイスの役割に依存します。ペリフェラルモードでは、追加の Tx FIFO がアクティブな IN エンドポイントのそれぞれに対して割り当てられます。FIFO のサイズは、ソフトウェアで設定できます。

57.11.1 ペリフェラル FIFO アーキテクチャ

図 742. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ペリフェラル Rx FIFO

OTG ペリフェラルは、すべての OUT エンドポイントに向けたデータを受信する単一の受信 FIFO を使用します。受信パケットは、Rx FIFO の空きスペースが使用可能になるまで、連続的にスタックされます。受信パケットのステータス（OUT エンドポイントの宛先番号、バイト数、データ PID、および受信データの有効性を含みます）も、コアによってデータペイロードの最上部に格納されます。使用可能なスペースがなくなると、ホストトランザクションは NACK され、アドレス指定されたエンドポイントに割り込みが受信されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ペリフェラルによる受信 RAM バッファへの書き込みをより効率的にします。

- すべての OUT エンドポイントは同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、OUT トークンのいかなるホストシーケンスについても、限度まで受信 FIFO に書き込むことができます。

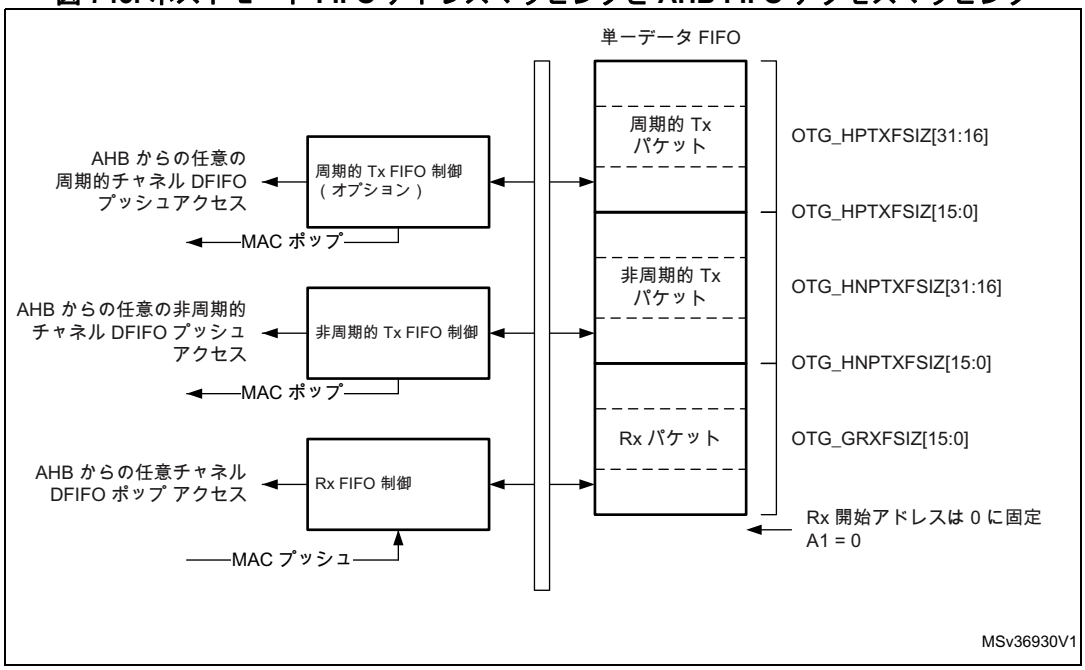
アプリケーションは Rx FIFO ノットエンプティ割り込み（OTG_GINTSTS の RXFLVL ビット）の受信を、少なくとも1 パケットがダウンロード可能である限り継続します。アプリケーションは、受信ステータス読み出しおよびポップレジスタ（OTG_GRXSTSP）からパケット情報を読み出し、最後に、エンドポイントに関するポップアドレスから読み取ることによって受信 FIFO からデータを取得します。

ペリフェラル Tx FIFO

コアは、各 IN エンドポイント用の専用 FIFO を備えています。アプリケーションは、IN エンドポイント 0 についてはエンドポイント 0 送信 FIFO サイズレジスタ（OTG_DIEPTXF0）に書き込み、IN エンドポイント x についてはデバイス IN エンドポイント送信 FIFOx レジスタ（OTG_DIEPTXFx）に書き込むことによって、FIFO のサイズを設定します。

57.11.2 ホスト FIFO アーキテクチャ

図 743. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ホスト Rx FIFO

ホストは、すべての周期的および非周期的トランザクションについて 1 つの受信 FIFO を使用します。この FIFO は、USB から受信したデータ（受信パケットのペイロード）をシステムメモリに転送されるまで保持する受信バッファとして使用されます。任意のリモート IN エンドポイントから受信されたパケットは、空きスペースがある限り連続的にスタックされます。各受信パケット（ホストチャネルデスティネーション、バイト数、データ PID、および受信データの有効性を含みます）のステータスも、FIFO に格納されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ホストによる受信データバッファへの書き込みをより効率的にします。

- IN に設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動された IN トークンのいかなるシーケンスについても、限度まで受信 FIFO に書き込むことができます。

アプリケーションは、少なくとも 1 つのパケットがダウンロード可能である限り、Rx FIFO ノットエンプティ割り込みを受け取ります。アプリケーションは、受信ステータス読み出しおよびポップレジスタからパケット情報を読み出し、最後に、受信 FIFO からデータを吸い上げます。

ホスト Tx FIFO

ホストは、すべての非周期的（コントロールおよびバルク）OUT トランザクションついて 1 つの送信 FIFO を、すべての周期的（アイソクロナスおよび割り込み）OUT トランザクションについて 1 つの送信 FIFO を使用します。FIFO は、USB 経由で送信されるデータ（送信パケットのペイロード）を保持する送信バッファとして使用されます。周期的（非周期的）Tx FIFO のサイズは、ホストの周期的（非周期的）送信 FIFO サイズレジスタ（OTG_HPTXFSIZ/OTG_HNPTXFSIZ）で設定されます。

Tx FIFO が 2 つ実装されているのは､USB フレームよりも周期的トラフィックタイプに対して高い優先度が与えられていることによります。各フレームの先頭で、組み込みのホストスケジューラは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。

2 つの送信 FIFO アーキテクチャにより、USB ホストは周期的送信データバッファと非周期的送信データバッファを個別に最適化できます。

- OUT 方向の周期的（非周期的）トランザクションをサポートするように設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動された OUT トークンのいかなるシーケンスについても、限度まで周期的（非周期的）送信 FIFO に書き込むことができます。

OTG_HS コアは、AHB 設定レジスタの周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）の値に応じて、周期的 Tx FIFO が半分または完全にエンプティである限り、周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の PTXFE ビット）を発行します。アプリケーションは、周期的 Tx FIFO と周期的リクエストキューの両方に空きスペースが存在する限り、前もって送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）を読み出すことで知ることができます。

OTG_HS コアは､AHB 設定レジスタの非周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）の値に応じて、非周期的 Tx FIFO が半分または完全にエンプティである限り、非周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の NPTXFE ビット）を発行します。アプリケーションは、非周期的 Tx FIFO と非周期的リクエストキューの両方に空きスペースが存在する限り、送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）を読み出すことで知ることができます。

57.11.3 FIFO RAM の割り当て

デバイスモード

受信 FIFO RAM 割り当て：アプリケーションは SETUP パケット用にRAMを割り当てる必要があります。

- コントロールエンドポイントで SETUP パケットを受信するため受信 FIFO に 10 ロケーションを確保する必要があります。コアは、SETUP パケット用に予約されている位置に他のデータを書き込むことはありません。
- 1 つの位置は、グローバル OUT NAK のために割り当てられます。
- 各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスエンドポイントが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ /4）+ 1 の 2 倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1 の 2 倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。
- 各エンドポイントの最後のパケットとともに、送信完了ステータス情報が FIFO にプッシュされます。各 OUT エンドポイントに対して 1 つのロケーションを予約することが推奨されます。

デバイス RxFIFO =

(4 * コントロールエンドポイント数 + 6) + （(最大 USB 使用パケット / 4）+ ステータス情報につき 1）+ （2 * OUT エンドポイント数）+ グローバル NAK につき 1

例：MPS は周期的 USB パケットでは 1,024 バイト、非周期的 USB パケットでは 512 バイトです。3 つの OUT エンドポイント、3 つの IN エンドポイント、1 つのコントロールエンドポイント、3 つのホストチャネルがあります。

デバイス RxFIFO = （4 * 1 + 6）+ （(1,024 / 4）+1）+ （2 * 4）+ 1 = 276

送信 FIFO RAM の割り当て： 各 IN エンドポイント送信 FIFO に必要な最小 RAM スペースは、その特定の IN エンドポイントに対する最大パケットサイズを収容できるスペースとなります。

注： ***送信 IN エンドポイント FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。***

ホストモード

受信 FIFO RAM 割り当て：

各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、(最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスチャネルが有効な場合、連続パケットを受信するには、少なくとも、(最大パケットサイズ / 4）+ 1 の 2 倍のスペースが割り当てられなければなりません。一般に、(最大パケットサイズ / 4）+ 1 の 2 倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。

ホストチャネルの最後のパケットとともに、転送完了ステータス情報が FIFO にプッシュされます。したがって、このために 1 つの位置が割り当てられなければなりません。

ホスト RxFIFO = （最大 USB 使用パケット / 4）+ ステータス情報につき 1 + 1 転送完了

例：ホスト RxFIFO = （(1,024 / 4）+ 1）+ 1 = 258

送信 FIFO RAM の割り当て：

ホストの非周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての非周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。

一般に、最大パケットサイズの 2 倍のスペースが推奨されます。現在のパケットが USB に転送されているときに、CPU が次のパケットを取得できるからです。

非周期的 TxFIFO = 最大非周期 USB 使用パケット / 4

例：非周期的 TxFIFO = （512 / 4）= 128

ホストの周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。少なくとも 1 つのアイソクロナス OUT エンドポイントがある場合には、スペースはそのチャネルの最大パケットサイズの少なくとも 2 倍でなければなりません。

ホスト周期的 TxFIFO = 最大周期 USB 使用パケット / 4

例：ホスト周期的 TxFIFO = （1,024 / 4）= 256

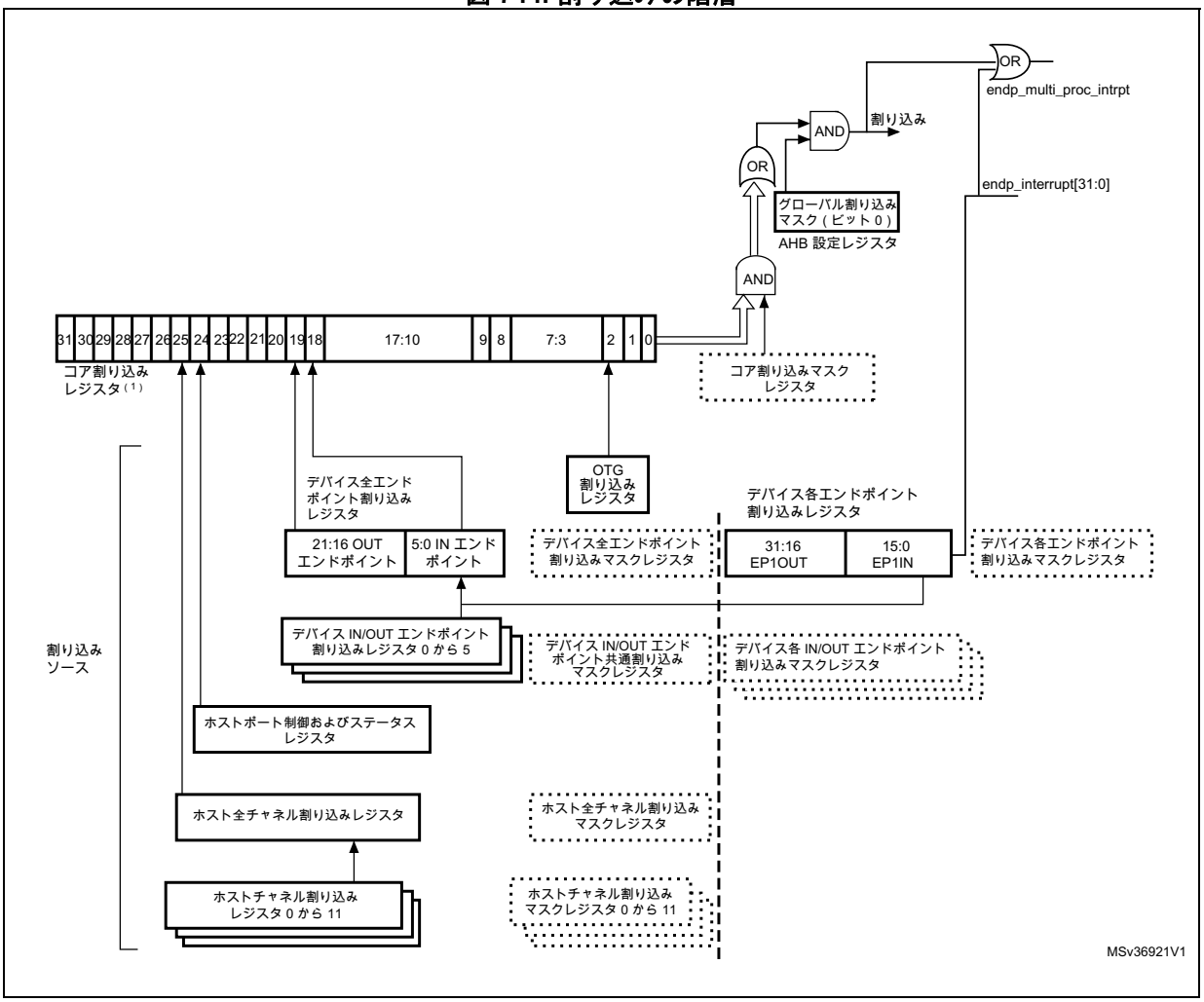
注： ***非周期的送信 FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。***

57.12 OTG_HS 割り込み

OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割り込みが生成され、コア割り込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

図 744 に、割り込みの階層を示します。

図 744. 割り込みの階層



1. コア割り込みレジスタビットは、*2529 ページのOTG コア割り込みレジスタ（OTG_GINTSTS）*に示されています。

57.13 OTG_HS 制御／ステータスレジスタ

アプリケーションは AHB スレーブインタフェースを通じて制御／ステータスレジスタ（CSR）を読み書きすることによって OTG_HS コントローラを制御します。これらのレジスタは 32 ビット幅であり、アドレスは 32 ビットブロックに配置されています。OTG_HS レジスタは、ワード（32 ビット）単位でアクセスする必要があります。

CSR は、次のように分類されます。

- コアグローバルレジスタ
- ホストモードレジスタ
- ホストグローバルレジスタ
- ホストポート CSR
- ホストチャネル固有レジスタ
- デバイスモードレジスタ
- デバイスグローバルレジスタ
- デバイスエンドポイント固有レジスタ
- パワーおよびクロックゲーティングレジスタ
- データ FIFO（DFIFO）アクセスレジスタ

コアグローバル、パワーおよびクロックゲーティング、データ FIFO（DFIFO）アクセス、およびホストポート CSR レジスタのみが、ホストモードおよびデバイスモードの両方でのアクセスが可能です。OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割り込みが生成され、コア割り込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

57.13.1 CSR メモリマップ

ホストモードおよびデバイスモードレジスタは、それぞれ異なるアドレスを占めます。すべてのレジスタは AHB クロックドメインで実装されます。

グローバル CSR マップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能です。

表 481. コアグローバル制御およびステータスレジスタ（CSR）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GOTGCTL	0x000	*2518 ページのOTG 制御およびステータスレジスタ（OTG_GOTGCTL）*
OTG_GOTGINT	0x004	*2521 ページのOTG 割り込みレジスタ（OTG_GOTGINT）*
OTG_GAHBCFG	0x008	*2522 ページのOTG AHB 設定レジスタ（OTG_GAHBCFG）*
OTG_GUSBCFG	0x00C	*2523 ページのOTG USB 設定レジスタ（OTG_GUSBCFG）*
OTG_GRSTCTL	0x010	*2527 ページのOTG リセットレジスタ（OTG_GRSTCTL）*
OTG_GINTSTS	0x014	*2529 ページのOTG コア割り込みレジスタ（OTG_GINTSTS）*
OTG_GINTMSK	0x018	*2534 ページのOTG 割り込みマスクレジスタ（OTG_GINTMSK）*

表 481. コアグローバル制御およびステータスレジスタ（CSR）（続き）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GRXSTSR	0x01C	2537 ページのOTG_FS 受信ステータスデバッグ読み出し／OTGステータス読み出しおよびポップレジスタ（OTG_GRXSTSR/OTG_GRXSTSP）
OTG_GRXSTSP	0x020	
OTG_GRXFSIZ	0x024	2538 ページのOTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）
OTG_HNPTXFSIZ/OTG_DIEPTXF0(1)	0x028	OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）
OTG_HNPTXSTS	0x02C	2540 ページのOTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）
OTG_GI2CCTL	0x030	2541 ページのOTG I^2C アクセスレジスタ（OTG_GI2CCTL）
OTG_GCCFG	0x038	2542 ページのOTG 一般コア設定レジスタ（OTG_GCCFG）
OTG_CID	0x03C	2543 ページのOTG コア ID レジスタ（OTG_CID）
OTG_GLPMCFG	0x54	2544 ページのOTG コア LPM 設定レジスタ（OTG_GLPMCFG）
OTG_HPTXFSIZ	0x100	2548 ページのOTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）
OTG_DIEPTXFx	0x104 0x124 ... 0x1B4	2548 ページのOTG デバイス IN エンドポイント送信 FIFO サイズレジスタ（OTG_DIEPTXFx）（x=1～8、ここで x はFIFO_number）

1. 一般的ルールとして、OTG_HNPTXFSIZ はホストモードで、OTG_DIEPTXF0 はデバイスモードで使用されます。

ホストモード CSR マップ

これらのレジスタは、コアがホストモードに変わるたびにプログラムされなければなりません。

表 482. ホストモード制御およびステータスレジスタ （CSR）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCFG	0x400	2549 ページのOTG ホスト設定レジスタ（OTG_HCFG）
OTG_HFIR	0x404	2549 ページのOTG ホストフレームインターバルレジスタ（OTG_HFIR）
OTG_HFNUM	0x408	2550 ページのOTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）
OTG_HPTXSTS	0x410	2551 ページのOTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）
OTG_HAINT	0x414	2552 ページのOTG ホスト全チャネル割り込みレジスタ（OTG_HAINT）
OTG_HAINTMSK	0x418	2552 ページのOTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK）
OTG_HPRT	0x440	2553 ページのOTG ホストポート制御およびステータスレジスタ（OTG_HPRT）

表 482. ホストモード制御およびステータスレジスタ （CSR）（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCCHARx	0x500 0x520 ... 0x6E0	*2555 ページのOTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0～15、ここで x = Channel_number）*
OTG_HCSPLTx	0x504 0x524 ... 0x6E4	*2557 ページのOTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）（x = 0..15、ここで x = Channel_number）*
OTG_HCDMAx	0x514 0x534 ... 0x6F4	*2561 ページのOTG ホストチャネル x DMA アドレスレジスタ（OTG_HCDMAx）（x = 0..15、ここで x = Channel_number）*
OTG_HCINTx	0x508 0x528 ... 0x6E8	*2557 ページのOTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0～15、ここで x = Channel_number）*
OTG_HCINTMSKx	0x50C 0x52C ... 0x6EC	*2558 ページのOTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）（x = 0～15、ここで x = Channel_number）*
OTG_HCTSIZx	0x510 0x530 ... 0x6F0	*2560 ページのOTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0～15、ここで x = Channel_number）*

デバイスモード CSR マップ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

表 483. デバイスモード制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_DCFG	0x800	*2561 ページのOTG デバイス設定レジスタ（OTG_DCFG）*
OTG_DCTL	0x804	*2563 ページのOTG デバイス制御レジスタ（OTG_DCTL）*
OTG_DSTS	0x808	*2566 ページのOTG デバイスステータスレジスタ（OTG_DSTS）*
OTG_DIEPMSK	0x810	*2567 ページのOTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）*
OTG_DOEPMSK	0x814	*2568 ページのOTG デバイス OUT エンドポイント共通割り込みマスクレジスタ （OTG_DOEPMSK）*
OTG_DAINT	0x818	*2569 ページのOTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT）*

表 483. デバイスモード制御およびステータスレジスタ（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_DAINTMSK	0x81C	2570 ページのOTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK）
OTG_DVBUSDIS	0x828	2570 ページのOTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）
OTG_DVBUSPULSE	0x82C	2571 ページのOTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）
OTG_DTHRCTL	0x0830	2571 ページのOTG デバイス閾値制御レジスタ（OTG_DTHRCTL）
OTG_DIEPEMPMSK	0x834	2573 ページのOTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ （OTG_DIEPEMPMSK）
OTG_DEACHINT	0x838	2572 ページのOTG デバイス各エンドポイント割り込みレジスタ（OTG_DEACHINT）
OTG_DEACHINTMSK	0x83C	2573 ページのOTG デバイス各エンドポイント割り込みレジスタマスク（OTG_DEACHINTMSK）
OTG_DIEPCTLx	0x900 0x920 ... 0x9E0	2574 ページのOTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 0～8、ここで x = Endpoint_number）
OTG_DIEPINTx	0x908 0x928 ... 0x9E8	2580 ページのOTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）（x = 0～8、ここで x = Endpoint_number）
OTG_DIEPTSIZ0	0x910	2583 ページのOTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）
OTG_DIEPDMAx	0x914	2583 ページのOTG デバイスチャネル x DMA アドレスレジスタ（OTG_DIEPDMAx）（x = 0～15、ここで x = Channel_number）
OTG_DTXFSTSx	0x918 0x938 ..... 0x9F8	2586 ページのOTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ （OTG_DTXFSTSx）（x = 0～8、ここで x = Endpoint_number）
OTG_DIEPTSIZx	0x930 0x950 ... 0x9F0	2585 ページのOTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1～8、ここで x = Endpoint_number）
OTG_DOEPCTL0	0xB00	2576 ページのOTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）
OTG_DOEPDMAx	0xB14	2584 ページのOTG デバイスチャネル x DMA アドレスレジスタ（OTG_DOEPDMAx）（x = 0～15、ここで x = Channel_number）
OTG_DOEPCTLx	0xB20 0xB40 ... 0xBE0	2578 ページのOTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1～8、ここで x = Endpoint_number）

表 483. デバイスモード制御およびステータスレジスタ（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_DOEPINTx	0xB08 0XB28 ... 0xBE8	*2582 ページのOTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）（x = 0～8、ここで x = Endpoint_number）*
OTG_DOEPTSIZ0	0xB10	*2584 ページのOTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）*
OTG_DOEPTSIZx	0xB30 0xB50 ... 0xBF0	*2587 ページのOTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1～8、ここで x =Endpoint_number）*

データ FIFO（DFIFO）アクセスレジスタマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能であり、特定のエンドポイントまたはチャネルの FIFO スペースを指定された方向で読み書きするために使用されます。ホストチャネルが IN タイプの場合 FIFO はそのチャネル上では読み出しのみが可能です。同様に、ホストチャネルが OUT タイプの場合 FIFO はそのチャネル上では書き込みのみが可能です。

表 484. データ FIFO（DFIFO）アクセスレジスタマップ

FIFO アクセスレジスタセクション	アドレス範囲	アクセス
デバイス IN エンドポイント 0／ホスト OUT チャネル 0 : DFIFO 書き込みアクセス デバイス OUT エンドポイント 0／ホスト IN チャネル 0 : DFIFO読み出しアクセス	0x1000～0x1FFC	w r
デバイス IN エンドポイント 1／ホスト OUT チャネル 1 : DFIFO 書き込みアクセス デバイス OUT エンドポイント 1／ホスト IN チャネル 1 : DFIFO読み出しアクセス	0x2000～0x2FFC	w r
...	...	...
デバイス IN エンドポイント x(1)／ホスト OUT チャネル x (1) : DFIFO 書き込みアクセス デバイス OUT エンドポイント x (1) ／ホスト IN チャネル x (1) : DFIFO読み出しアクセス	0xX000～0xXFFC	w r

1. デバイスモードでは x = 8、ホストモードでは x = 15 です。

パワーおよびクロックゲーティング CSR マップ

パワーおよびクロックゲーティングの場合、シングルレジスタです。このレジスタは、ホストモードとデバイスモードの両方で使用できます。

表 485. パワーおよびクロックゲーティング制御およびステータスレジスタ

レジスタ名	項目（略称）	オフセットアドレス：0xE00～0xFFF
パワーおよびクロックゲーティング制御レジスタ	PCGCCTL	0xE00～0xE04
予約済み	-	0xE05～0xFFF

57.14 OTG_HS レジスタ

これらのレジスタは、ホストモードとデバイスモードの両方で使用でき、これらのモード間で切り替わっても再プログラムの必要はありません。

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

57.14.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL）

アドレスオフセット：0x000

リセット値：0x0X01 0000

OTG_GOTGCTL レジスタは、コアの OTG 機能の動作を制御し、そのステータスを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTG VER	BSVLD	ASVLD	DBCT	CID STS
											rw	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EHEN	DHNP EN	HSHNP EN	HNP RQ	HNG SCS	BVALO VAL	BVALO EN	AVALO VAL	AVALO EN	VBVAL OVAL	VBVAL OEN	SRQ	SRQ SCS
			rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw	rw	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **OTGVER**：OTG バージョン

OTG バージョンを選択します。

0：OTG バージョン 1.3 です。

1：OTG バージョン 2.0 です。このバージョンでは、コアは SRP のデータラインパルシングのみをサポートしています。

ビット 19 **BSVLD**：B セッション有効

デバイスモードでのトランシーバのステータスを示します。

0：Bセッションは無効。

1：Bセッションは有効。

OTG モードでは、このビットを使用して、デバイスが接続されているか、切断されているかを判断できます。

注： デバイスモードでのみアクセス可能です。

ビット 18 **ASVLD**：A セッション有効

ホストモードでのトランシーバのステータスを示します。

0：Aセッションは無効。

1：Aセッションは有効。

注： ホストモードでのみアクセス可能です。

ビット 17 **DBCT**：ロング／ショートデバウンス時間

検出された接続のデバウンス時間を示します。

0：ロングデバウンス時間（100 ms + 2.5 µs）。物理的接続に使用されます。

1：ショートデバウンス時間（2.5 µs）。ソフト接続に使用されます。

注： ホストモードでのみアクセス可能です。

ビット 16 **CIDSTS**：コネクタ ID ステータス

接続イベント時のコネクタ ID のステータスを示します。

0：OTG_HS コントローラは A デバイスモードです。

1：OTG_HS コントローラは B デバイスモードです。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **EHEN**：組み込みホスト有効化

OTG A デバイスのステートマシンと組み込みホストのステートマシンから選択するために使用します。

0：OTG A デバイスのステートマシンを選択します。

1：組み込みホストのステートマシンを選択します。

ビット 11 **DHNPEN**：デバイス HNP 有効化

アプリケーションは、接続された USB ホストから SetFeature.SetHNPEnable コマンドを正常に受信したときに、このビットをセットします。

0：HNP は、アプリケーションでは無効です。

1：HNP は、アプリケーションで有効です。

注： デバイスモードでのみアクセス可能です。

ビット 10 **HSHNPEN**：ホストセット HNP 有効化

アプリケーションは、接続されたデバイスで HNP を正常に有効できたとき（SetFeature.SetHNPEnable コマンドを使用して）、このビットをセットします。

0：ホストセット HNP は無効です。

1：ホストセット HNP は有効です。

注： ホストモードでのみアクセス可能です。

ビット 9 **HNPRQ**：HNP リクエストあり

アプリケーションは、接続された USB ホストへの HNP リクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

0：HNP リクエストなし

1：HNP リクエストあり

注： デバイスモードでのみアクセス可能です。

ビット 8 **HNGSCS**：ホストネゴシエーション成功

コアは、ホストネゴシエーションに成功したとき、このビットをセットします。コアは、このレジスタの HNP リクエストビット（HNPRQ）がセットされたとき、このビットをクリアします。

0：ホストネゴシエーション失敗

1：ホストネゴシエーション成功

注： デバイスモードでのみアクセス可能です。

ビット 7 **BVALOVAL**：B ペリフェラルセッション有効上書き値

このビットは、BVALOEN ビットがセットされている場合に、Bvalid 信号に上書き値をセットするために使用されます。

0：BVALOEN = 0 のとき、Bvalid 値は“1”です。

1：BVALOEN = 1 のとき、Bvalid 値は“1”です。

注： デバイスモードでのみアクセス可能です。

ビット 6 **BVALOEN**：B ペリフェラルセッション有効値の上書き有効化

このビットは、BVALOVAL ビットを使用して Bvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Bvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Bvalid 信号は BVALOVAL ビット値で上書きされます。

注： デバイスモードでのみアクセス可能です。

ビット 5 **AVALOVAL**：A ペリフェラルセッション有効上書き値

このビットは、AVALOEN ビットがセットされている場合に、Avalid 信号に上書き値をセットするために使用されます。

0：AVALOEN = 0 のとき、Avalid 値は“1”です。

1：AVALOEN = 1 のとき、Avalid 値は“1”です。

注： ホストモードでのみアクセス可能です。

ビット 4 **AVALOEN**：A ペリフェラルセッション有効値の上書き有効化

このビットは、AVALOVAL ビットを使用して Avalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Avalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Avalid 信号は AVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 3 **VBVALOVAL**：V_{BUS} 有効上書き値

このビットは、VBVALOEN ビットがセットされている場合に、vbusvalid 信号に上書き値をセットするために使用されます。

0：VBVALOEN = 0 のとき、vbusvalid 値は“1”です。

1：VBVALOEN = 1 のとき、vbusvalid 値は“1”です。

注： ホストモードでのみアクセス可能です。

ビット 2 **VBVALOEN**：V_{BUS} 有効値上書き有効化

このビットは、VBVALOVAL ビットを使用して vbusvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの vbusvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った vbusvalid 信号は VBVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 1 **SRQ**：セッションリクエスト

アプリケーションは USB 上でセッションリクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

USB 1.1 フルスピードシリアルトランシーバインタフェースを使用してセッションリクエストを開始する場合、アプリケーションは、このレジスタの B セッション有効ビット（OTG_GOTGCTL の BSVLD ビット）がクリアされた後 V_{BUS} が 0.2 V へ放電するまで待つ必要があります。この放電時間は PHY ごとに異なり PHY ベンダーから得ることができます。

0：セッションリクエストなし

1：セッションリクエスト

注： デバイスモードでのみアクセス可能です。

ビット 0 **SRQSCS**：セッションリクエスト成功

コアは、セッションリクエストの開始に成功したとき、このビットをセットします。

0：セッションリクエスト失敗

1：セッションリクエスト成功

注： デバイスモードでのみアクセス可能です。

57.14.2 OTG 割り込みレジスタ（OTG_GOTGINT）

アドレスオフセット：0x04

リセット値：0x0000 0000

アプリケーションは OTG 割り込みがあると、このレジスタを読み出し、このレジスタのビットをクリアすることによって OTG 割り込みをクリアします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDCHNG	DBCDNE	ADTOCHG	HNGDET	Res.
											rc_w1	rc_w1	rc_w1	rc_w1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
						rc_w1	rc_w1						rc_w1		

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 IDCHNG：
このビットがセットされている場合、ID 入力ピンの値に変更があることを示します。

ビット 19 **DBCDNE**：デバウンス終了
コアは、デバイスが接続された後、デバウンスが完了したときに、このビットをセットします。アプリケーションは、この割り込みを確認した後に USB リセットの駆動を開始できます。このビットは、OTG_GUSBCFG レジスタの HNP 対応ビットまたは SRP 対応ビット（OTG_GUSBCFG の HNPCAP ビットと SRPCAP ビット）がセットされたときのみ有効です。
注： ホストモードでのみアクセス可能です。

ビット 18 **ADTOCHG**：A デバイスタイムアウト変更
コアは B デバイスの接続を待っている間に A デバイスがタイムアウトしたことを示すために、このビットをセットします。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 17 **HNGDET**：ホストネゴシエーションの検出
コアは USB 上でホストネゴシエーションリクエストを検出したときに、このビットをセットします。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **HNSSCHG**：ホストネゴシエーション成功ステータス変化
コアは USB ホストネゴシエーションリクエストの成功または失敗時に、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのホストネゴシエーション成功ビット（OTG_GOTGCTL の HNGSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRSSCHG**：セッションリクエスト成功ステータス変化

コアは、セッションリクエストの成功または失敗のときに、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのセッションリクエスト成功ビット（OTG_GOTGCTL の SRQSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **SEDET**：セッション終了検出

コアは、このビットをセットして、V_{BUS} の電圧レベルが B ペリフェラルセッションにとって有効でなくなったことを示します（V_{BUS} < 0.8 V のとき）。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

57.14.3 OTG AHB 設定レジスタ（OTG_GAHBCFG）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタを使用して、電源投入後、またはモード変化後のコアを設定できます。このレジスタは、主に AHB システム関連の設定パラメータを含んでいます。初期プログラミングの後は、このレジスタを変更しないでください。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PTXFELVL**：周期的 Tx FIFO エンプティレベル

OTG_GINTSTS レジスタの周期的 Tx FIFO エンプティ割り込みビット（OTG_GINTSTS の PTXFE ビット）がトリガされるタイミングを示します。

0：OTG_GINTSTS の PTXFE 割り込みビットは、周期的 TxFIFO が半エンプティであることを示します。

1：OTG_GINTSTS の PTXFE 割り込みビットは、周期的 Tx FIFO が完全エンプティであることを示します。

注： ホストモードでのみアクセス可能です。

ビット 7 **TXFELVL**：Tx FIFO エンプティレベル

デバイスモードでは、このビットは、IN エンドポイント送信 FIFO エンプティ割り込み（OTG_DIEPINTx の TXFE）がトリガされるタイミングを示します。

0：OTG_DIEPINTx の TXFE 割り込みは、IN エンドポイント TxFIFO が半エンプティであることを示します。

1：OTG_DIEPINTx の TXFE 割り込みは、IN エンドポイント TxFIFO が完全エンプティであることを示します。

ホストモードでは、このビットは、非周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の NPTXFE ビット）がトリガされるタイミングを示します。

0：OTG_GINTSTS の NPTXFE 割り込みは、非周期的 Tx FIFO が半エンプティであることを示します。

1：OTG_GINTSTS の NPTXFE 割り込みは、非周期的 Tx FIFO が完全エンプティであることを示します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DMAEN**：DMA は有効です。

0：コアはスレーブモードで動作します。
1：コアは DMA モードで動作します。

ビット 4:1 **HBSTLEN**：バースト長／タイプ

0000 シングル：バストランザクションがシングル 32 ビットアクセスを使用します（非推奨）。
0001 INCR：バストランザクションが指定されていない長さのアクセスを使用します（非推奨、INCR AHB バスコマンドを使用）。
0011 INCR4：バストランザクションが 4x 32 ビットアクセスをターゲットします。
0101 INCR8：バストランザクションが 8x 32 ビットアクセスをターゲットします。

0111 INCR16：バストランザクションが 16x 32 ビットアクセスに基づきます。
その他：予約済み

ビット 0 **GINTMSK**：グローバル割り込みマスク

アプリケーションは、このビットを使用して、アプリケーション自身に対する割り込みラインのアサーションをマスクまたはマスク解除します。このビットの設定に関係なく、割り込みステータスレジスタはコアによって更新されます。
0：アプリケーションへの割り込みアサーションをマスクします。
1：アプリケーションへの割り込みアサーションをマスク解除します。

注： デバイスモードとホストモードの両方でアクセス可能です。

57.14.4 OTG USB 設定レジスタ（OTG_GUSBCFG）

アドレスオフセット：0x00C

リセット値：0x0000 1400

このレジスタを使用して、電源投入後、またはホストモードあるいはデバイスモードへの変化後に、コアを設定することができます。このレジスタは USB および USB-PHY 関連の設定パラメータを含んでいます。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FD MOD	FH MOD	Res.	Res.	Res.	ULPIIPD	PTCI	PCCI	TSDPS	ULPIEVBUSI	ULPIEVBUSI	ULPICSM.	ULPIAR.	ULPIFSL.	Res.
	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PHYLPC.	Res.	TRDT				HNPCAP	SRPCAP	Res.	PHYSEL	Res.	ULPI_SEL	Res.	TOCAL		
rw		rw				rw	rw		rw		rw		rw		

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **FDMOD**：強制デバイスモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にデバイスモードになります。

0：通常モード

1：強制デバイスモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **FHMOD**：強制ホストモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にホストモードになります。

0：通常モード

1：強制ホストモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 28:26 予約済みであり、USB OTG HS および FS のリセット値に保持する必要があります。

ビット 25 **ULPIIPD**：ULPI インタフェース保護ディセーブル

このビットは PHY に組み込まれた回路を制御して、リンクが stp とデータをトライステートにすると ULPI インタフェースを保護します。この機能のために使用するプルアップ抵抗またはプルダウン抵抗は無効化できます。詳細については ULPI 仕様を参照してください。

0：インタフェース保護回路を有効化

1：インタフェース保護回路を無効化

ビット 24 **PTCI**：インジケータパススルー

このビットは、内部 V_{BUS} 有効コンパレータを使用して、RX CMD の V_{BUS} 状態で使用する前に、相補出力を評価するか否かを制御します。詳細については ULPI 仕様を参照してください。

0：相補出力信号を内部 V_{BUS} 有効コンパレータで評価します。

1：相補出力信号を内部 V_{BUS} 有効コンパレータで評価しません。

ビット 23 **PCCI**：インジケータ相補

このビットは PHY を制御して外部 /Vbus インジケータ入力信号を反転させて、相補出力を発生します。詳細については ULPI 仕様を参照してください。

0：PHY は外部／Vbus インジケータ信号を反転させません。

1：PHY は外部／Vbus インジケータ信号を反転させます。

ビット 22 **TSDPS**：TermSel DLine パルス選択

このビットは SRP （セッションリクエストプロトコル）時にデータラインパルスを駆動する utmi_termselect を選択します。

0：utmi_txvalid を使用するデータラインパルシング（デフォルト）

1：utmi_termsel を使用するデータラインパルシング

ビット 21 **ULPIEVBUSI**：ULPI 外部 V_{BUS} インジケータ

このビットは ULPI の PHY が外部 V_{BUS} 過電流のインジケータを使用するよう指定します。

0：PHY が内部 V_{BUS} 有効コンパレータを使用

1：PHY が外部 V_{BUS} 有効コンパレータを使用

ビット 20 **ULPIEVBUSD**：ULPI 外部 V_{BUS} ドライブ

このビットは、ULPI の PHYで V_{BUS}で 5 V を駆動するため内部電源または外部電源のいずれを使用するかを選択します。

0：PHY は 内部チャージポンプを使って V_{BUS} を駆動 （デフォルト）

1：PHY は外部電源を使って V_{BUS} を駆動

ビット 19 **ULPICSM**：ULPIクロックSuspendM

このビットは ULPI PHY のインタフェース制御レジスタのClockSuspendM ビットを設定します。このビットは、シリアルモードと carkit モードでのみ使用されます。

0：PHY はサスペンド中に内部クロックをパワーダウンさせます。

1：PHY は内部クロックをパワーダウンさせません。

ビット 18 **ULPIAR**：ULPI 自動レジューム

このビット PHY のインタフェース制御レジスタの自動レジュームビットを設定します。

0：PHY は自動レジューム機能を使用しません。

1：PHY は自動レジューム機能を使用します。

ビット 17 **ULPIFSLS**：ULPI FS/LS選択

アプリケーションではこのビットを使用し PHY シリアルインタフェースを選択します。このビットは PHY シリアルトランシーバが選択されたときにのみ有効です。

0：ULPI インタフェース

1：ULPI FS/LS シリアルインタフェース

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PHYLPCS**：PHY 低消費電力クロック選択

このビットは 480 MHz また 48 MHz （低消費電力）モードを選択します。FS モードと LS モードでは PHY は 48 MHz クロックで動作して消費電力を削減できます。

0：480 MHz 内部 PLL クロック

1：48 MHz 外部クロック

480 MHz モードでは UTMI インタフェースは選択した 8 ビットまた 16 ビットのデータ幅に応じ 60 または 30 MHz で動作します。48 MHz モードでは UTMI インタフェースはFS モードと LS モードで 48 MHz 動作します。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:10 **TRDT**：USB ターンアラウンド時間

ターンアラウンド時間を PHY クロックで設定します。これらは、*表 487：TRDT 値（HS）*によってアプリケーション AHB 周波数により設定する必要があります。TRDT 値が高ければ、IN トークンへの USB 応答時間を伸ばして、データ FIFO へのより長い AHB 読み出しアクセス遅延を補正できます。

注： デバイスモードでのみアクセス可能です。

ビット 9 **HNPCAP**：HNP 機能

アプリケーションは、このビットを使用して OTG_HS コントローラの HNP 機能を制御します。

0：HNP 機能は無効です。

1：HNP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 8 **SRPCAP**：SRP 機能

アプリケーションは、このビットを使用して OTG_HS コントローラの SRP 機能を制御します。コアが非 SRP 対応 B デバイスとして動作する場合、
デバイスは、接続された A デバイス（ホスト）に V_{BUS} を起動し、セッションを開始させるように要求することはできません。

0：SRP 機能は無効です。

1：SRP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **PHYSEL**：フルスピードシリアルトランシーバの選択

0：USB 2.0 外部 ULPI ハイスピード PHY または内部 UTMI ハイスピード PHY（ULPI_SEL も参照）

1：USB 1.1 フルスピードシリアルトランシーバ

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ULPI_SEL：使用するハイスピードインタフェースの選択**

0：UTMI インタフェースが選択されます。

1：ULPI インタフェースが選択されます。

Bit3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TOCAL**：FS タイムアウトの較正

アプリケーションがこのフィールドでプログラムする PHY クロックの数は、PHY によって導入される追加遅延を考慮して、コアのフルスピードパケット間タイムアウト時間に追加されます。ライン状態条件の生成時に PHY によって導入される遅延は PHY ごとに異なる場合があるので、較正が必要なことがあります。

フルスピード動作での USB の標準タイムアウト値は 16～18 ビット時間です。アプリケーションは、エニュメレーションのスピードに基づいて、このフィールドをプログラムする必要があります。PHY クロックごとに追加されるビット時間の数は 0.25 ビット時間です。

表 486. TRDT 値

AHB 周波数範囲（MHz）		TRDT 最小値
最小値	最大値	
14.2	15	0xF
15	16	0xE
16	17.2	0xD
17.2	18.5	0xC
18.5	20	0xB
20	21.8	0xA
21.8	24	0x9
24	27.5	0x8
27.5	32	0x7
32	-	0x6

表 487. TRDT 値（HS）

AHB 周波数範囲（MHz）		TRDT 最小値
最小値	最大値	
30	-	0x9

57.14.5 OTG リセットレジスタ（OTG_GRSTCTL）

アドレスオフセット：0x10

リセット値：0x8000 0000

アプリケーションは、このレジスタを使用して、コア内のさまざまなハードウェア機能をリセットします。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXFNUM					TXFFLSH	RXFFLSH	Res.	Res.	PSRST	CSRST
					rw					rs	rs			rs	rs

ビット 31 **AHBIDL**：AHB マスタアイドル

AHB マスタステートマシンがアイドル状態にあることを示します。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 30 **DMAREQ**：DMA リクエスト信号有効化

このビットは DMA リクエストが進行中であることを表示します。デバッグには使用しません。

ビット 29:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **TXFNUM**：Tx FIFO 番号

これは Tx FIFO フラッシュビットを使用して一掃しなければならない FIFO 番号です。コアが Tx FIFO フラッシュビットをクリアするまでは、このフィールドを変更しないでください。

00000：

– ホストモードでの非周期的 Tx FIFO の一掃

– デバイスモードでの Tx FIFO 0 の一掃

00001：

– ホストモードでの周期的 Tx FIFO の一掃

– デバイスモードでの Tx FIFO 1 の一掃

00010：デバイスモードでの Tx FIFO 2 の一掃

...

01111：デバイスモードでの Tx FIFO 15 の一掃

10000：デバイスモードまたはホストモードのすべての送信 FIFO をフラッシュします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 5 **TXFFLSH**：Tx FIFO の一掃

このビットは、単一またはすべての送信 FIFO を選択的にフラッシュしますが、コアがトランザクション中の場合はフラッシュできません。

アプリケーションは、コアが Tx FIFO への書き込み中か TxFIFO からの読み出し中のどちらでもないことを確認した後でなければ、このビットに書き込むことはできません。以下のレジスタを使用して確認してください。

読み出し — NAK 有効割り込みを使用すると、コアが FIFO から読み出しを行っていないことを確認できます。

書き込み — OTG_GRSTCTL の AHBIDL ビットで、コアが FIFO に書き込みを行っていないことを保証します。

FIFO が再設定される場合は通常一掃を実施することをお勧めします。デバイスエンドポイントが無効の間も、FIFO 一掃の実施をお勧めします。アプリケーションは、コアがこのビットをクリアするまで待ってから、他の操作を実行しなければなりません。phy_clk または hclk のより低速なクロックを使用してこのビットをクリアするには、8 クロックかかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 4 **RXFFLSH**：Rx FIFO の一掃

アプリケーションは、このビットを使用して Rx FIFO 全体を一掃できますが、まず、コアがトランザクション中でないことを確認する必要があります。

アプリケーションは、コアが Rx FIFO からの読み出し中、または Rx FIFO への書き込み中のどちらでもないことを確認した後でなければ、このビットに書き込んではなりません。

アプリケーションは、他の操作を実行する前に、このビットがクリアされるまで待つ必要があります。このビットがクリアされるまで、8 クロック（PHY または AHB クロックの中で最も低速なクロック）かかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **PSRST**：部分的なソフトリセット

内部ステートマシンをリセットしますが、エニュメレーション情報は保持されます。特定の PHY エラーからの復帰に使用できます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 0 **CSRST**：コアソフトリセット

以下のように HCLK および PHY クロックドメインをリセットします。

割り込みと、以下のビットを除くすべての CSR レジスタビットをクリアします。

– OTG_PCGCCTL レジスタ の GATEHCLK ビット
– OTG_PCGCCTL レジスタ の STPPCLK ビット
– OTG_HCFG レジスタ の FSLSPCS ビット
– OTG_DCFG レジスタ の DSPD ビット
– OTG_DCTL レジスタ の SDIS ビット
– OTG_GCCFG レジスタ

すべてのモジュールのステートマシン（AHB スレーブユニットを除く）がアイドル状態にリセットされ、すべての送信 FIFO および受信 FIFO がフラッシュされます。

AHB 転送の最後のデータフェーズの完了後 AHB マスタ上のすべてのトランザクションは速やかに終了されます。USB 上のすべてのトランザクションをただちに終了します。

アプリケーションは、いつでもこのビットに書き込むことによって、コアをリセットできます。これはセルフクリアビットであり、コア内のすべての必要なロジックがリセットされた後、コアがこのビットをクリアします。コアの現在の状態によって、クリアには数クロックかかります。このビットがクリアされると、ソフトウェアは PHY ドメインにアクセスできるようになるまで少なくとも 3 PHY クロック待つ必要があります（同期遅延）。また、ソフトウェアは、操作を開始する前に、このレジスタのビット 31 が 1 にセットされていること（AHB マスタがアイドル状態）を確認する必要があります。

一般に、ソフトウェアリセットは、ソフトウェア開発時に使用され、上記の USB 設定レジスタの PHY 選択ビットを動的に変更するときにも使用されます。PHY を変更するときには PHY の対応クロックが選択され PHY ドメインで使用されます。新しいクロックが選択されたら、適切な動作のためには PHY ドメインをリセットする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

57.14.6 OTG コア割り込みレジスタ（OTG_GINTSTS）

アドレスオフセット：0x014

リセット値：0x1400 0020

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割り込みをかけます。

このレジスタには、ホストモードでのみ有効なビットと、デバイスモードでのみ有効なビットがあります。このレジスタは、現在のモードも示します。rc_w1 タイプの割り込みステータスビットをクリアするには、アプリケーションは、そのビットに 1 を書き込む必要があります。

FIFO ステータス割り込みは読み出し専用です。これらの割り込みの処理中に、ソフトウェアが FIFO への書き込みまたは FIFO からの読み出しを行うと、FIFO 割り込み条件は自動的にクリアされます。

初期化の前に割り込みが発生するのを避けるためには、アプリケーションは、割り込みビットのマスクを解除する前に OTG_GINTSTS レジスタをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	Res.	PTXFE	HCINT	HPRT INT	Res.	DATAF SUSP	IPXFR/ IN COMP ISO OUT	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1		r	r	r		rc_w1	rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUM DNE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXF E	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット 31 **WKUPINT**：レジューム／リモートウェイクアップ検出割り込み

サスペンド（L2）または LPM（L1）状態中のウェイクアップ割り込み

– サスペンド（L2）中

デバイスモードでは、この割り込みは、USB 上でレジュームが検出されたときにアサートされます。ホストモードでは、この割り込みは USB 上でリモートウェイクアップが検出されたときにアサートされます。

– LPM（L1）中

この割り込みは USB 上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 30 **SRQINT**：セッションリクエスト／新規セッション検出割り込み

ホストモードでは、この割り込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割り込みは、V_{BUS} が B ペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割り込み

デバイスの切断が検出されたときにアサートされます。

注：　ホストモードでのみアクセス可能です。

ビット 28 **CIDSCHG**：コネクタ ID ステータス変化

コアは コネクタ ID ステータスに変化があったとき、このビットをセットします。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PTXFE**：周期的 TxFIFO エンプティ

周期的送信 FIFO が半エンプティか完全エンプティであり、周期的リクエストキューに少なくとも 1 つのエントリを書き込むためのスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）によって判断されます。

注：　ホストモードでのみアクセス可能です。

ビット 25 **HCINT**：ホストチャネル割り込み

コアは、コアのチャネルの 1 つで割り込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINT レジスタを読み出して、割り込みが発生しているチャネルの正確な数を確認した後、対応する OTG_HCINTx レジスタを読み出して、割り込みの正確な原因を確認する必要があります。アプリケーションは OTG_HCINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注：　ホストモードでのみアクセス可能です。

ビット 24 **HPRTINT**：ホストポート割り込み

コアは、ホストモードの OTG_HS コントローラポートの 1 つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRT レジスタを読み出して、この割り込みの原因となった正確なイベントを特定する必要があります。アプリケーションは OTG_HPRT レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **DATAFSUSP**：データフェッチサスペンド

この割り込みは DMA モードでのみ有効です。この割り込みは TxFIFO スペースまたはリクエストキュースペースがないためコアが IN エンドポイントに対するデータフェッチを停止したことを表示します。アプリケーションでは、エンドポイント不一致アルゴリズムに対してこの割り込みを使います。例えば、エンドポイント不一致を検出後、アプリケーションは以下を実行します。

- グローバルな非周期的 IN NAK ハンドシェイクを設定
- IN エンドポイントを無効化
- FIFO を一掃
- IN トークンシーケンス学習キューからトークンシーケンスを決定
- エンドポイントを再度有効化

グローバル非周期的 IN NAK がクリアされ、コアは IN エンドポイントのデータをフェッチしておらず、かつ IN トークンを受信した場合、すなわちコアが「FIFO エンプティ時 IN トークン受信」割り込みを生成した場合、グローバル非周期的 IN NAK ハンドシェイクをクリアします。次にOTGは NAK 応答をホストへ送信します。このシナリオを回避するため、アプリケーションは OTG_GINTSTS 内で FetSusp 割り込みをチェックすることができます。これにより、グローバル NAK ハンドシェイクをクリアする前に FIFO がフルであることが保証されます。あるいは、グローバル NAK ハンドシェイクをクリアするとき、アプリケーションが「FIFOエンプティ中にIN トークンを受信」割り込みはマスクされますすることができます。

ビット 21 **IPXFR**：不完全な周期的転送

ホストモードでは、コアは、現在のフレームでスケジュールされていて、まだペンディング中の不完全な周期的トランザクションが存在するとき、この割り込みビットをセットします。

INCOMPISOOUT：不完全アイソクロナス OUT 転送

デバイスモードでは、コアは、現在のフレームで転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあることを示すために、この割り込みをセットします。この割り込みは、このレジスタの周期的フレーム割り込み終了（EOPF）ビットとともにアサートされます。

ビット 20 **IISOIXFR**：不完全アイソクロナス IN 転送

コアは、現在のフレームで転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあることを示すために、この割り込みをセットします。この割り込みは、このレジスタの周期的フレーム割り込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割り込み

コアは、コアの OUT エンドポイントの 1 つで割り込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割り込みが発生した OUT エンドポイントの正確な数を確認した後、対応する OTG_DOEPINTx レジスタを読み出して、割り込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DOEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割り込み

コアは、コアの IN エンドポイントの 1 つで割り込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割り込みが発生した IN エンドポイントの正確な数を確認した後、対応する OTG_DIEPINTx レジスタを読み出して、割り込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DIEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPF**：周期的フレーム終了割り込み

現在のフレームで、OTG_DCFG レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRP**：アイソクロナス OUT パケット欠落割り込み

コアは、アイソクロナス OUT エンドポイントの最大サイズのパケットに対応できるだけの十分なスペースが Rx FIFO になかったために Rx FIFO にアイソクロナス OUT パケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNE**：エニュメレーション終了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセット

コアは USB 上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSP**：USB サスペンド

コアは USB 上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが 1 つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット 10 **ESUSP**：アーリーサスペンド

コアは USB 上で 3 ms 間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFF**：グローバル OUT NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル OUT NAK ビット（OTG_DCTL の SGONAK ビット）がコアで有効になったことを示します。このビットは、OTG_DCTL レジスタのクリアグローバル OUT NAK ビット（OTG_DCTL の CGONAK ビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFF**：グローバルIN非周期的NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル 非周期的 IN NAK ビット（OTG_DCTL の SGINAK ビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバル IN NAK ビットをサンプリングしました。このビットは、OTG_DCTL レジスタのクリアグローバル非周期的 IN NAK ビット（OTG_DCTLの CG I NAK ビット）をクリアすることによってクリアできます。

この割り込みは、必ずしも NAK ハンドシェイクが USB に送出されることを意味しません。STALL ビットが NAK ビットに優先します。

注： デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFE**：非周期的 Tx FIFO エンプティ

この割り込みは、非周期的 Tx FIFO が半エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも 1 つのエントリを書き込むスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの非周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット 4 **RXFLVL**：Rx FIFO ノットエンプティ

Rx FIFO から読み出されるべきペンディング中のパケットが少なくとも 1 つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 3 **SOF**：フレーム開始

ホストモードでは、コアは SOF（FS）またはキープアライブ（LS）が USB で送信されることを示すために、このビットをセットします。アプリケーションは、このビットに 1 を書き込むことによって、割り込みをクリアする必要があります。

デバイスモードでは、コアは、USB 上で SOF トークンが受信されたことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割り込みは、コアが FS で動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると“1”を返すことがあります。パワーオンリセット後すぐに“1”が読み出された場合、そのレジスタのビットが、SOF が送信された（ホストモードの場合）または SOF が受信された（デバイスモードの場合）ことを示すわけではありません。この割り込みの読み出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割り込み

コアは、OTG プロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションは OTG 割り込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割り込みの原因となった正確なイベントを確認する必要があります。アプリケーションは OTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 1 **MMIS**：モード不一致割り込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY 応答によって AHB 上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

57.14.7 OTG 割り込みマスクレジスタ（OTG_GINTMSK）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタは、コア割り込みレジスタとともに働き、アプリケーションに割り込みをかけます。割り込みビットがマスクされると、そのビットに関連する割り込みは生成されません。ただし、その割り込みに対応するコア割り込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	

ビット 31 **WUIM**：レジューム／リモートウェイクアップ検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 30 **SRQIM**：セッションリクエスト／新規セッション検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 28 **CIDSCHGM**：コネクタ ID ステータス変化マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 27 **LPMINTM**：LPM 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 26 **PTXFEM**：周期的 Tx FIFO エンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 25 **HCIM**：ホストチャネル割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 24 **PRTIM**：ホストポート割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 23 **RSTDETM**：リセット検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 22 **FSUSPM**：データフェッチサスペンドマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
ペリフェラルモードでのみアクセス可能です。

ビット 21 **IPXFRM**：不完全周期的転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。
IISOOXFRM：不完全アイソクロナス OUT 転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 20 **IISOIXFRM**：不完全アイソクロナス IN 転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPFM**：周期的フレーム終了割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRPM**：アイソクロナス OUT パケット欠落割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNEM**：エニュメレーション終了マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセットマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSPM**：USB サスペンドマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 10 **ESUSPM**：アーリーサスペンドマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFFM**：グローバル OUT NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFFM**：グローバル非周期的IN NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFEM**：非周期的 Tx FIFO エンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 4 **RXFLVLM**：受信 FIFOノットエンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 **SOFM**：フレーム開始マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 1 **MMISM**：モード不一致割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

57.14.8 OTG_FS 受信ステータスデバッグ読み出し／OTGステータス読み出しおよびポップレジスタ（OTG_GRXSTSR/OTG_GRXSTSP）

読み出しアドレスオフセット：0x01C

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

受信ステータスデバッグ読み出しレジスタを読み出すと、受信 FIFO の最上部の内容が返されます。受信ステータス読み出しおよびポップレジスタを読み出すと Rx FIFO の最上部のデータエントリも取り出され（ポップされ）ます。

受信ステータスの内容は、ホストモードとデバイスモードで異なった解釈をする必要があります。コアは、受信 FIFO がエンプティのときには、受信ステータスのポップ／読み出しを無視して、0x0000 0000 の値を返します。アプリケーションは、コア割り込みレジスタの受信 FIFO ノットエンプティビット（OTG_GINTSTS の RXFLVL ビット）がアサートされたときにのみ、受信ステータス FIFO をポップしなければなりません。

ホストモード：

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT											CHNUM			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:17 **PKTSTS**：パケットステータス

受信パケットのステータスを示します。
0010：IN データパケットが受信されました。
0011：IN 転送が完了しました（割り込みをトリガします）。
0101：データトグルエラー（割り込みをトリガします）。
0111：チャネル停止（割り込みをトリガします）。
その他：予約済み

ビット 16:15 **DPID**：データ PID

受信パケットのデータ PID を示します。
00：データ 0
10：データ 1
01：データ 2
11：MDATA

ビット 14:4 **BCNT**：バイトカウント

受信した IN データパケットのバイト数を示します。

ビット 3:0 **CHNUM**：チャネル番号

現在の受信パケットが属するチャネル番号を示します。

デバイスモード：

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT											EPNUM			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:21 **FRMNUM**：フレーム番号

これは USB 上でパケットが受信されるフレーム番号の下位 4 ビットです。このフィールドは、アイソクロナス OUT エンドポイントがサポートされる場合にのみサポートされます。

ビット 20:17 **PKTSTS**：パケットステータス

受信パケットのステータスを示します。
0001：グローバル OUT NAK（割り込みをトリガします）。
0010：OUT データパケットが受信されました。
0011：OUT 転送が完了しました（割り込みをトリガします）。
0100：SETUP トランザクションが完了しました（割り込みをトリガします）。
0110：SETUP データパケットが受信されました。
その他：予約済み

ビット 16:15 **DPID**：データ PID

受信した OUT データパケットのデータ PID を示します。
00：データ 0
10：データ 1
01：データ 2
11：MDATA

ビット 14:4 **BCNT**：バイトカウント

受信したデータパケットのバイト数を示します。

ビット 3:0 **EPNUM**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

57.14.9 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）

アドレスオフセット：0x024

リセット値：0x0000 0400

アプリケーションは RxFIFO に割り当てられなければならない RAM サイズをプログラムできます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFD															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RXFD**：Rx FIFO 深さ
この値は 32 ビットワード単位です。
最大値は 1024 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

57.14.10 OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ(OTG_DIEPTXF0)

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NPTXFD/TX0FD															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSA/TX0FSA															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ホストモード

ビット 31:16 **NPTXFD**：非周期的 Tx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **NPTXFSA**：非周期的送信 RAM 開始アドレス
このフィールドは、非周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

デバイスモード

ビット 31:16 **TX0FD**：エンドポイント 0 TxFIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **TX0FSA**：エンドポイント 0 送信 RAM 開始アドレス
このフィールドは、エンドポイント 0 送信 FIFO RAM のメモリ開始アドレスを設定します。

57.14.11 OTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）

アドレスオフセット：0x02C

注： **リセット値：0x0008 0400** ***デバイスモードでは、このレジスタは無効です。***

この読み出し専用レジスタは、非周期的 Tx FIFO および非周期的送信リクエストキューの空きスペース情報を含みます。

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **NPTXQTOP**：非周期的送信リクエストキューの先頭

MAC によって処理中の非周期的 Tx リクエストキューのエントリ。
ビット 30:27チャネル／エンドポイント番号
ビット 26:25
00：IN/OUT トークン
01：ゼロ長送信パケット（デバイス IN／ホスト OUT）
11：チャネル停止コマンド
ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **NPTQXSAV**：非周期的送信リクエストキューの使用可能スペース

非周期的送信リクエストキューの使用可能な空きスペースの量を示します。このキューは IN および OUT リクエストの両方を保持します。
0：非周期的送信リクエストキューはフルです。
1：1 位置が使用可能です。
2：2 位置が使用可能です。
n：n 位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット 15:0 **NPTXFSAV**：使用可能な非周期的 Tx FIFO スペース

非周期的 Tx FIFO 内で使用できる空きスペースの量を示します。
値は 32 ビットワード単位です。
0：非周期的 Tx FIFO はフルです。
1：1 ワードが使用可能です。
2：2 ワードが使用可能です。
n：n ワードが使用可能です （0 ≤ n ≤ 512）
その他：予約済み

57.14.12 OTG I²C アクセスレジスタ（OTG_GI2CCTL）

アドレスオフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BSYDNE	RW.	Res.	I2CDATSE	I2CDEVADR		Res.	ACK	I2CEN	ADDR						
rw	rw		rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGADDR								RWDATA							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **BSYDNE**：I²C ビジー／完了

アプリケーションはこのビットに 1 を設定して I²C インタフェースのリクエストを開始します。転送が完了すると、コアはこのビットに 0 を設定します。このビットがセットされ I²C インタフェースがビジーであることを表示しているかぎり、アプリケーションはインタフェースの次のリクエストを開始できません。

ビット 30 **RW**：読み出し／書き込みインジケータ

このビットは、インタフェースでレジスタ読み出しまたは書き込み転送を実行する必要があることを表示します。

0：書き込み

1：読み出し

注： バースト読み出し／書き込みは、レジスタに対してサポートされていません。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **I2CDATSE0**：I²C DatSe0 USB モード

このビットを使ってフルスピードインタフェース USB モードを選択します。

0：VP_VM USB モード

1：DAT_SE0 USB モード

ビット 27:26 **I2CDEVADR**：I²C デバイスアドレス

このビットは、コアが OTG シグナリングに使用する USB 1.1 フルスピードシリアルトランシーバ上の I²C スレーブのアドレスを選択します。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ACK**：I²C ACK

このビットは ACK 応答が I²C スレーブから受信されたか否かを示します。これは、アプリケーションが I²C アクセスを開始した後にコアが BSYDNE をクリアしたときに有効になります。

0：NAK

1：ACK

ビット 23 **I2CEN**：I²C 有効化

このビットは I²C マスタを有効化し、I²C インタフェース上でトランザクションを開始させます。

ビット 22:16 **ADDR**：I^2C アドレス

この 7 ビットの I^2C デバイスアドレスは、アプリケーションが 外部 I^2C スレーブにアクセスするときに使用するもので、USB 1.1 OTG フルスピードシリアルトランシーバの I^2C スレーブを含みます。

ビット 15:8 **REGADDR**：I^2C レジスタアドレス

これらのビットを使うと、読み出しまたは書き込み対象のレジスタアドレスをプログラムすることができます。

ビット 7:0 **RWDATA**：I^2C 読み出し／書き込みデータ

レジスタ読み出し動作後、これらのビットはアプリケーションの読み出しデータを保持します。
書き込み動作では、アプリケーションはこのレジスタを使用してレジスタへ書き込むデータをプログラムすることができます。

注：　設定レジスタは USB OTG HS にのみ適用されます。

57.14.13 OTG 一般コア設定レジスタ（OTG_GCCFG）

アドレスオフセット：0x038

リセット値：0x0000 XXX0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCD EN	BCDEN	PWR DWN
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2 DET	SDET	PDET	DCDET
												r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **VBDEN**：USB V_{BUS} 検出有効化

V_{BUS} レベル検出用コンパレータを有効にして、V_{BUS} の有効レベルを検出します（USB ホストおよびデバイス動作用の V_{BUS} パッド上で）。HNP または SRP サポートが有効であれば、V_{BUS} コンパレータは VBDEN 値とは関係なく、自動的に有効化されます。

0：V_{BUS} 検出無効
1：V_{BUS} 検出有効

ビット 20 **SDEN**：2 次検出（SD）モードイネーブル

このビットは、BCD を SD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 19 **PDEN**：1 次検出（PD）モードイネーブル

このビットは、BCD を PD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 18 **DCDEN**：データ接触検出（DCD）モードイネーブル

このビットは、BCD を DCD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 17 **BCDEN**：バッテリ充電検出（BCD）イネーブル

このビットは、USB デバイス内の BCD サポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHY は BCD によって完全に制御され、通常の通信では使用できなくなります。BCD の検出完了後は、通常の USB 動作を可能にするために、このビットを“0”にクリアして BCD を OFF モードにします。

ビット 16 **PWRDWN**：パワーダウン制御

送信／受信のためにトランシーバを起動するために使用されます。リセットされている場合、トランシーバはパワーダウン状態に保たれます。セットする場合、BCD 機能をオフにする必要があります（BCDEN=0）。

0：USB FS トランシーバ無効

1：USB FS トランシーバ有効

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PS2DET**：DM プルアップ検出ステータス

このビットは、PD 中にのみアクティブになって DM 電圧レベルと VLGC 閾値の比較の結果を示します。通常の状況では、DM レベルは閾値を下回る必要があります。閾値を上回る場合は、DM が外部的にハイレベルにプルアップされることを意味します。これは、PS2 ポート（DP および DM ラインの両方をプルアップする）への接続、または BCD 仕様に準拠しない一部の独自の充電器への接続によって発生します。

0：通常のポートが検出されました（SDP、CDP、または DCP に接続）。

1：PS2 ポートまたは独自の充電器が検出されました。

ビット 2 **SDET**：2 次検出（SD）ステータス

このビットは、SD の結果を示します。

0：CDP が検出されました。

1：DCP が検出されました。

ビット 1 **PDET**：1 次検出（PD）ステータス

このビットは、PD の結果を示します。

0：BCD サポートは検出されませんでした（SDP または独自デバイスに接続）。

1：BCD のサポートが検出されました（CDP、または DCP に接続）。

ビット 0 **DCDET**：データ接触検出（DCD）ステータス

このビットは、DCD の結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

57.14.14 OTG コア ID レジスタ（OTG_CID）

アドレスオフセット：0x03C

リセット値：0x0000 3100 これは、製品 ID を含む読み出し専用レジスタです。

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

ビット 31:0 **PRODUCT_ID**：製品 ID フィールド

アプリケーションによってプログラム可能な ID フィールド。

57.14.15 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EN BESL	LPMRCNTSTS			SND LPM	LPMRCNT			LPMCHIDX				L1RSM OK
			rw	r	r	r	rs	rw	rw	rw	rw	rw	rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLP STS	LPMRSP		L1DS EN	BESLTHRS				L1SS EN	REM WAKE	BESL				LPM ACK	LPM EN
r	r	r	rw	rw	rw	rw	rw	rw	rw/r	rw/r	rw/r	rw/r	rw/r	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **ENBESL**：ベストエフォート型サービス遅延の有効化

このビットは LPM 正誤表で定義されているように BESL 機能を有効にします。

0 : コアは以下の文書に記載されているように機能します。
USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007

1 : コアは LPM 正誤表に記載されているように機能します。
USB 2.0 ECN 用の正誤表 : Link Power Management (LPM) - 7/2007

注： この文書では更新された動作（LPM 正誤表に記載）のみが考慮されているため、アプリケーション切り替えで ENBESL ビットを 1 にセットする必要があります。

ビット 27:25 **LPMRCNTSTS**：LPM 再試行カウントステータス

現在の LPM シーケンスでは、送信すべき LPM ホストの再試行回数がまだ残っています。

注： ホストモードでのみアクセス可能です。

ビット 24 **SNDLPM**：送信 LPM トランザクション

アプリケーションソフトウェアがこのビットをセットすると、2 つのトークン EXT および LPM を含む LPM トランザクションが送信されます。デバイスから有効な応答（STALL、NYET、または ACK）を受信した時点、またはコアがプログラムされた回数の LPM 再試行を送信し終えた時点で、このビットはハードウェアによってクリアされます。

注： このビットは、ホストがローカルポートに接続されている場合のみセットする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23:21 **LPMRCNT**：LPM 再試行カウント

デバイスが ERROR 応答を返すと、このビットは追加の LPM 再試行回数となります。追加の再試行は、有効なデバイスからの応答（STALL、NYET、または ACK）が受信されるまで、ホストによって実行されます。

注： ホストモードでのみアクセス可能です。

ビット 20:17 **LPMCHIDX**：LPM チャネルインデックス

ローカルデバイスへの LPM トランザクションの送信中に LPM トランザクションを適用するべきチャネルの番号を示します。LPM チャネルインデックスに基づき、コアは自動的に対応するチャネルでプログラムされたデバイスアドレスとエンドポイント番号を LPM トランザクションに挿入します。

注： ホストモードでのみアクセス可能です。

ビット 16 **L1RSMOK**：スリープ状態レジューム OK

デバイスまたはホストがスリープ状態からレジュームを開始できることを示します。このビットは LPM スリープ状態（L1）のときに有効です。このビットは 50 µs（$T_{L1Residency}$）の遅延後に SLEEP モードでセットされます。

このビットは、SLPSTS = 0 のときにリセットされます。

1：アプリケーションまたはホストはスリープ状態からレジュームを開始できます。

0：アプリケーションまたはホストはスリープ状態からレジュームを開始できません。

ビット 15 **SLPSTS**：ポートスリープステータス

デバイスモード：

このビットは、USB バスにスリープ状態が存在する限りセットされます。コアは、ACK 応答が LPM トランザクションに送信され、$T_{L1TokenRetry}$ タイマが時間切れになった時点で、スリープ状態に入ります。PHY クロックを停止させるには、アプリケーションで OTG_PCGCCTL の STPPCLK ビットをセットすることにより、PHY サスペンド入力信号をアサートする必要があります。

アプリケーションはスリープへの遷移を確認する際に、LPMRSP の ACK ではなく、SLPSTS を信頼する必要があります。

コアは、以下の場合にスリープ状態から抜け出します。

– USB ライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL の RWUSIG ビットへ書き込むときやデバイスをリセットまたはソフト切断するとき。

ホストモード：

デバイスから ACK 応答を受け取ったローカルポートへのコアによる LPM トランザクション の成功の副作用として、ホストはスリープ状態（L1）へ遷移します。このビットの読み出し値は、ポートの現在のスリープ状態を反映しています。

コアは、次の動作後にこのビットをクリアします。

– コアがリモート L1 ウェイクアップ信号を検出します、

– アプリケーションで、OTG_HPRT レジスタの PRST ビットまたは PRES ビットをセットします、または

– アプリケーションで、コア割り込みレジスタの L1 レジューム／リモートウェイクアップ検出割り込みビットまたは切断検出割り込みビット(OTG_GINTSTS の WKUPINT または DISCINT ビット）をセットします。

0：コアは L1 状態ではありません。

1：コアは L1 状態です。

ビット 14:13 **LPMRST**：LPM 応答

デバイスモード：

受信した LPM トランザクションへのコアの応答は、これら 2 つのビットに反映されます。

ホストモード：

ローカルデバイスから受け取った LPM トランザクションへのハンドシェイク応答

11：ACK

10：NYET

01：STALL

00：ERROR（ハンドシェイク応答なし）

ビット 12 **L1DSEN**：L1 ディープスリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット11:8 **BESLTHRS**：BESL 閾値

デバイスモード：

BESL 値がこの BESL_Thres[3:0] フィールドの規定値以上である場合、コアは PHY を L1 状態でディープ低電力モードに移行させます。

ホストモード：

コアは PHY をL1 状態でディープ低電力モードに移行させます。BESLTHRS[3:0] は、デバイスから開始されたレジュームを検出した場合に、レジュームシグナリングに USB バス上のホスト（$T_{L1HubDrvResume2}$）が反映されるまでの時間を指定します。

ホストモードでは、BESLTHRS には 1100b より大きい値を設定してはなりません。この値が $T_{L1HubDrvResume2}$の最大値を超えてしまうからです。

Thres[3:0]ホストモードレジュームシグナリング時間（µs）

0000 : 75
0001 : 100
0010 : 150
0011 : 250
0100 : 350
0101 : 450
0110 : 950
その他の値は予約済みです。

ビット 7 **L1SSEN**：L1 シャロースリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット 6 **REMWAKE**：bRemoteWake 値

ホストモード：

LPM トランザクションの wIndex フィールドの送信されるリモートウェイクアップの値

デバイスモード（読み出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン bRemoteWake bmAttribute で更新されます。

ビット 5:2 **BESL：**ベストエフォート型サービス遅延

ホストモード：

LPM トランザクションの送信される BESL の値この値は、ホストから開始されたレジュームの処理時間（$T_{L1HubDrvResume1}$ ）の間にレジュームを開始するためにも使用されます。

デバイスモード（読み出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン BESL bmAttribute で更新されます。

BESL[3:0]T_{BESL} (µs)

0000 : 125
0001 : 150
0010 : 200
0011 : 300
0100 : 400
0101 : 500
0110 : 1000
0111 : 2000
1000 : 3000
1001 : 4000
1010 : 5000
1011 : 6000
1100 : 7000
1101 : 8000
1110 : 9000
1111 : 10000

ビット 1 **LPMACK：**LPM トークン確認応答有効化

デバイスのアプリケーションソフトウェアによって事前にプログラムされた LPM トークンへのハンドシェイク応答

1 : ACK

ACK は事前にプログラムされているものの、コア（デバイス）は成功した LPM トランザクションに対してのみ ACK 応答を返します。LPM トランザクションは次の場合に成功します。

– EXT トークンまたは LPM トークンのどちらにも PID/CRC5 エラーがない（それ以外の場合は ERROR）。
– LPM トランザクションで受信した有効な bLinkState = 0001B（L1）（それ以外の場合は STALL）。
– 送信シーケンスにペンディング中のデータがない（それ以外の場合は NYET）。

0 : NYET

事前にプログラムされたソフトウェアビットは、次の場合に、LPM トークンへの応答に上書きされます。

– 受信した bLinkState が L1 でない場合（STALL 応答）、または
– いずれかの LPM トークンパケットで、破損によるエラーが検出された場合（エラー応答）

注： デバイスモードでのみアクセス可能です。

ビット 0 **LPMEN：**LPM サポートイネーブル

アプリケーションは、このビットを使用して、OTG_HS コア LPM 機能を制御します。

コアが非 LPM 対応ホストとして動作する場合、ホストは接続されたデバイスまたはハブに LPM モードを有効にするように要求することはできません。

コアが非 LPM 対応デバイスとして動作する場合、デバイスはいかなる LPM トランザクションにも応答できません。

0：LPM 機能は無効です。
1：LPM 機能は有効です。

57.14.16 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）

アドレスオフセット：0x100

リセット値：0x0200 0400

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

ビット 31:16 **PTXFD**：ホスト周期的 Tx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。

ビット 15:0 **PTXSA**：ホスト周期的 Tx FIFO の開始アドレス
このフィールドは、周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

57.14.17 OTG デバイス IN エンドポイント送信 FIFO サイズレジスタ（OTG_DIEPTXFx）（x=1～8、ここで x はFIFO_number）

アドレスオフセット：0x104 + （FIFO_number – 1）* 0x04

リセット値：

FIFO_number = 8：0x0200 0200 + （8 * 0x200）

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

ビット 31:16 **INEPTXFD**：IN エンドポイント Tx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。

ビット 15:0 **INEPTXSA**：IN エンドポイント FIFOx 送信 RAM 開始アドレス
このフィールドは IN エンドポイント送信 FIFOx のメモリ開始アドレスを含みます。アドレスは 32 ビットメモリロケーションに整列されている必要があります。

57.14.18 ホストモードレジスタ

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

ホストモードレジスタは、ホストモードでのコアの動作に影響を与えます。.ホストモードレジスタは、結果が定義されていないため、デバイスモードではアクセスしないでください。ホストモードレジスタは、次のように分類することができます。

57.14.19 OTG ホスト設定レジスタ（OTG_HCFG）

アドレスオフセット：0x400

リセット値：0x0000 0000

このレジスタは、電源投入後のコアを設定します。ホストを初期化した後は、このレジスタを変更しないでください。

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FSLSS**：FS および LS のみのサポート

アプリケーションは、このビットを使用して、コアのエニュメレーション速度を制御します。このビットを使用して、アプリケーションは、接続されたデバイスが HS トラフィックをサポートする場合でも、FS ホストとしてコアにエニュメレーションさせることができます。初期プログラミングの後は、このフィールドを変更しないでください。

ビット 1:0 **FSLSPCS**：FS/LS PHY クロック選択

コアが FS ホストモードのとき、

01：PHY クロックは 48 MHz で動作しています。

その他：予約済み

コアが LS ホストモードのとき、

00：予約済み

01：PHY クロック周波数として 48 MHz を選択

10：PHY クロック周波数として 6 MHz を選択

11：予約済み

注： FSLSPCS は、接続されたデバイスの速度に応じて、接続イベント時にセットする必要があります（このビットを変更した後、ソフトウェアリセットを実行する必要があります）。

57.14.20 OTG ホストフレームインターバルレジスタ（OTG_HFIR）

アドレスオフセット：0x404

リセット値：0x0000 EA60

このレジスタは OTG_HS コントローラがエニュメレーションした現在のスピードでのフレームインターバル情報を格納します。

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RLDCTRL**：再ロード制御

このビットを使用して、ランタイム時に HFIR レジスタの動的再ロードが可能です。
0：HFIR は動的に再ロードすることができません。
1：HFIR はランタイム時に動的に再ロードすることができます。
このビットは、初期設定時にプログラムする必要があり、ランタイム時にそのを変更してはなりません。

ビット 15:0 **FRIVL**：フレームインターバル

アプリケーションがこのフィールドにプログラムする値は、2 つの連続したマイクロ SOF（HS）またはキープアライブトークン（LS）の間のインターバルを指定します。このフィールドは、必要なフレームインターバルを構成する PHY クロック数を含みます。アプリケーションは、ホストポート制御およびステータスレジスタのポートイネーブルビット（OTG_HPRT の PENA ビット）がセットされた後のみ、このレジスタに値を書き込むことができます。値がプログラムされていない場合、コアは、ホスト設定レジスタの FS/LS PHY クロック選択フィールド（OTG_HCFG の FSLSPCS）で指定された PHY クロックに基づいて値を計算します。初期設定の後は、RLDCTRL ビットがセットされている場合を除いて、このフィールドの値を変更しないでください。そのような場合、FRIVL には 各 SOF イベントが再ロードされます。

57.14.21 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）

アドレスオフセット：0x408

リセット値：0x0000 3FFF

このレジスタは、現在のフレーム番号を示します。また、現在のフレームの残り時間（PHY クロック数）も示します。

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

ビット 31:16 **FTREM**：残りフレーム時間

現在のフレームの残り時間を PHY クロック数で示します。このフィールドは PHY クロックごとにデクリメントされます。値がゼロになると、このフィールドにはフレームインターバルレジスタの値が再ロードされ、新しい SOF が　USB に送信されます。

ビット 15:0 **FRNUM**：フレーム番号

このフィールドは USB 上で新しい SOF が送信されるたびにインクリメントされ、0x3FFF に達すると 0 にクリアされます。

57.14.22 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）

アドレスオフセット：0x410

リセット値：0x0008 0100

この読み出し専用レジスタは、周期的 Tx FIFO および周期的送信リクエストキューの空きスペース情報を含みます。

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

ビット 31:24 **PTXQTOP**：周期的送信リクエストキューの先頭

これは MAC によって処理中の周期的 Tx リクエストキューのエントリを示します。
このレジスタは、デバッグに使用されます。
ビット 31：奇数／偶数フレーム
0：偶数フレームで送信
1：奇数フレームで送信
ビット 30:27チャネル／エンドポイント番号
ビット 26:25タイプ
00：IN/OUT
01：長さゼロのパケット
11：チャネルコマンドディセーブル
ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **PTXQSAV**：周期的送信リクエストキューの使用可能スペース

周期的送信リクエストキューで書き込みに使用できる空きロケーションの数を示します。このキューは IN および OUT リクエストの両方を保持します。
00：周期的送信リクエストキューはフルです。
01：1 位置が使用可能です。
10：2 位置が使用可能です。
bxn：n 位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット 15:0 **PTXFSAVL**：周期的送信データ FIFO の使用可能スペース

周期的 Tx FIFO で書き込みに使用できる空きロケーションの数を示します。
値は 32 ビットワード単位です。
0000：周期的 Tx FIFO はフルです。
0001：1 ワードが使用可能です。
0010：2 ワードが使用可能です。
bxn：n ワードが使用可能です （0 ≤ n ≤ PTXFD）
その他：予約済み

57.14.23 OTG ホスト全チャネル割り込みレジスタ（OTG_HAINT）

アドレスオフセット：0x414

リセット値：0x0000 000

チャネル上で重大なイベントが発生すると、ホスト全チャネル割り込みレジスタは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）を使用して、アプリケーションに割り込みをかけます。これを *図 744*に示します。チャネルごとに 1 つの割り込みビットがあります（最大 16 ビット）。このレジスタのビットは、アプリケーションが対応するホストチャネル x 割り込みレジスタのビットをセット／クリアしたときにセット／クリアされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINT															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINT**：チャネル割り込み

チャネルあたり 1 ビット: チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

57.14.24 OTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK）

アドレスオフセット：0x418

リセット値：0x0000 0000

ホスト全チャネル割り込みマスクレジスタは、ホスト全チャネル割り込みレジスタとともに働き、チャネル上でイベントが発生したときに、アプリケーションに割り込みをかけます。チャネルごとに 1 つの割り込みマスクビットがあります（最大 16 ビット）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINTM															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINTM**：チャネル割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

チャネルあたり 1 ビット: チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

57.14.25 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）

アドレスオフセット：0x440

リセット値：0x0000 0000

このレジスタは、ホストモードでのみ使用可能です。現時点では OTG ホストは 1 つのポートのみをサポートしています。

1 つのレジスタは、ポートごとの USB リセット、イネーブル、サスペンド、レジューム、接続ステータス、テストモードなどの USB ポート関連情報を保持します。*図 744* に示されています。このレジスタの rc_w1 ビットは、コア割り込みレジスタのホストポート割り込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割り込みをトリガできます。ポート割り込みの場合、アプリケーションは、このレジスタを読み出して、割り込みの原因となったビットをクリアする必要があります。rc_w1 ビットの場合、アプリケーションは、このビットに 1 を書き込んで、割り込みをクリアする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTCTL			PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POC CHNG	POCA	PEN CHNG	PENA	PCDET	PCSTS
rw	rw	rw	rw	r	r		rw	rs	rw	rc_w1	r	rc_w1	rc_w1	rc_w1	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:17 **PSPD**：ポートスピード

このポートに接続されたデバイスのスピードを示します。
01：フルスピード
10：ロースピード
11：予約済み
00：ハイスピード

ビット 16:13 **PTCTL**：ポートテスト制御

アプリケーションがこのフィールドにゼロ以外の値を書き込んで、ポートをテストモードにすると、対応するパターンがポートにシグナリングされます。
0000：テストモードは無効です
0001：Test_J モード
0010：Test_K モード
0011：Test_SE0_NAK モード
0100：Test_Packet モード
0101：Test_Force_Enable
その他：予約済み

ビット 12 **PPWR**：ポートパワー

アプリケーションは、このフィールドを使用して、このポートへの電力供給を制御し、コアは、過電流条件が発生すると、このビットをクリアします。
0：パワーオフ
1：パワーオン

ビット 11:10 **PLSTS**：ポートラインステータス

USB データラインの現在のロジックレベルを示します。

ビット 10：OTG_DP のロジックレベル

ビット 11：OTG_DM のロジックレベル

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PRST**：ポートリセット

アプリケーションがこのビットをセットすると、このポートでリセットシーケンスが開始されます。アプリケーションは、リセットシーケンスの完了後、リセット時間を決めて、このビットをクリアする必要があります。

0：ポートは非リセット中

1：ポートはリセット中

アプリケーションは、ポートのリセットが開始されるまで、少なくとも 10 ms の間、このビットをセット状態に保つ必要があります。USB 規格では最大限度が規定されていませんが、アプリケーションは、このビットをクリアする前に、必要最小時間に加えて、さらに 10 ms の間、このビットをセット状態に保つことができます。

ハイスピード：50 ms

フルスピード／ロースピード：10 ms

ビット 7 **PSUSP**：ポートサスペンド

アプリケーションは、このビットをセットして、このポートをサスペンドモードにします。コアは、このビットがセットされると SOF の送信のみを停止します。PHY クロックを停止するには、アプリケーションは PHY のサスペンド入力ピンをアサートするポートクロック停止ビットをセットする必要があります。

このビットの読み出し値は、ポートの現在のサスペンドステータスを反映しています。このビットは、リモートウェイクアップ信号の検出後、またはアプリケーションがこのレジスタのポートリセットビットまたはポートレジュームビット、またはコア割り込みレジスタのレジューム／リモートウェイクアップ検出割り込みビットまたは切断検出割り込みビット（それぞれ OTG_GINTSTS の WKUINT または DISCINT ビット）をセットしたときに、コアによってクリアされます。

0：ポートは非サスペンドモード

1：ポートはサスペンドモード

ビット 6 **PRES**：ポートレジューム

アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。コアは、アプリケーションがこのビットをクリアするまで、レジューム信号の駆動を続行します。

コアが、コア割り込みレジスタのポートレジューム／リモートウェイクアップ検出割り込みビット（OTG_GINTSTS の WKUINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、切断条件を検出すると、このビットをクリアします。このビットの読み出し値は、コアが現在レジューム信号を駆動中であるかどうかを示します。

0：レジューム信号を駆動していません。

1：レジューム信号を駆動しています。

LPM が有効で、コアが L1 状態にあるとき、このビットは次のように動作します。

1.アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。

2.コアは OTG_GLPMCFG レジスタの BESLTHRS[3:0] フィールドに指定された所定時間まで、レジューム信号の駆動を続行します。

3.コアが、コア割り込みレジスタのポート L1 レジューム／リモート L1 ウェイクアップ検出割り込みビット（OTG_GINTSTS の WKUPINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、レジュームの最後にビットをクリアします。このビットは、コアおよびアプリケーション両方でセット／クリアできます。このビットは、ホストにデバイスが 1 つも接続されていない場合でもコアによってクリアされます。

ビット 5 **POCCHNG**：ポート過電流の変化

コアは、このレジスタのポート過電流アクティブビット（ビット 4）に変化があったとき、このビットをセットします。

ビット 4 **POCA**：ポート過電流アクティブ

ポートの過電流条件を示します。

0：非過電流状態

1：過電流状態

ビット 3 **PENCHNG**：ポートイネーブル／ディセーブルの変化

コアは、このレジスタのポートイネーブルビット 2 のステータスに変化があったとき、このビットをセットします。

ビット 2 **PENA**：ポートイネーブル

ポートは、リセットシーケンス後、コアによってのみ有効にされ、過電流条件、切断条件、またはアプリケーションがこのビットをクリアすることによって無効にされます。アプリケーションは、レジスタへの書き込みによってこのビットをクリアすることはできません。クリアして、ポートを無効にすることだけができます。このビットは、アプリケーションへの割り込みをトリガしません。

0：ポートを無効化

1：ポートを有効化

ビット 1 **PCDET**：ポート接続の検出

コアは、デバイス接続が検出されると、このビットをセットして、コア割り込みレジスタのホストポート割り込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割り込みをトリガします。アプリケーションは、このビットに 1 を書き込むことによって、割り込みをクリアする必要があります。

ビット 0 **PCSTS**：ポート接続ステータス

0：ポートにはデバイスが接続されていません。

1：ポートにデバイスが接続されています。

57.14.26 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0〜15、ここで x = Channel_number）

アドレスオフセット：0x500 + (Channel_number * 0x20)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHENA	CHDIS	ODD FRM	DAD							MCNT		EPTYP		LSDEV	Res.
rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPDIR	EPNUM				MPSIZ										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **CHENA**：チャネルイネーブル

このフィールドは、アプリケーションによってセットされ OTG ホストによってクリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

ビット 30 **CHDIS**：チャネル無効化

アプリケーションは、チャネルへの転送が完了する前でも、このビットをセットして、そのチャネル上での送信／受信を停止することができます。アプリケーションは、チャネルを無効として扱う前に、チャネル無効割り込みを待つ必要があります。

ビット 29 **ODDFRM**：奇数フレーム

このフィールドは OTG ホストが奇数フレームで転送を実行しなければならないことを示すために、アプリケーションによってセット（リセット）されます。このフィールドは、周期的（アイソクロナスおよび割り込み）トランザクションに対してのみ適用されます。

0：偶数フレーム

1：奇数フレーム

ビット 28:22 **DAD**：デバイスアドレス

このフィールドは、データソースまたはデータシンクとして機能する特定のデバイスを選択します。

ビット 21:20 **MCNT**：マルチカウント

このフィールドは、この周期的エンドポイントについてフレームあたりで実行されなければならないトランザクションの数をホストに示します。非周期的転送の場合、このフィールドは使用されません。

00：予約済み。このフィールドは定義されていない結果をもたらします。

01：1 トランザクション

10：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 2 つです。

11：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 3 つです。

注： このフィールドは、01 以上にセットする必要があります。

ビット 19:18 **EPTYP**：エンドポイントタイプ

選択された転送タイプを示します。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **LSDEV**：ロースピードデバイス

このフィールドは、このチャネルがロースピードデバイスと通信中であることを示すために、アプリケーションによってセットされます。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EPDIR**：エンドポイントの方向

トランザクションが IN か OUT かを示します。

0：OUT

1：IN

ビット 14:11 **EPNUM**：エンドポイント番号

データソースまたはデータシンクとして機能しているデバイスのエンドポイント番号を示します。

ビット 10:0 **MPSIZ**：最大パケットサイズ

関連するエンドポイントの最大パケットサイズを示します。

57.14.27 OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）（x = 0..15、ここで x = Channel_number）

アドレスオフセット：0x504 + (Channel_number * 0x20)

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XACTPOS		HUBADDR							PRTADDR						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SPLITEN**：スプリットイネーブル
アプリケーションはこのビットをセットして、このチャネルがスプリットトランザクション処理用に有効化されていることを表示します。

ビット 30:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **COMPLSPLT**：完全なスプリットを実行
アプリケーションはこのビットをセットして OTG ホストによる完全なスプリットトランザクション実行を要求します。

ビット 15:14 **XACTPOS**：トランザクションポジション
このフィールドを使って、各OUTトランザクションと一緒に全て、先頭、中央、または最終のいずれのペイロードを送信するか否かを決めます。
11：全このトランザクションの全データペイロード （188 バイト以下）
10：先頭このトランザクションの先頭データペイロード（188 バイト超）
00：中央このトランザクションの中央のペイロード （188 バイト超）
01：最終このトランザクションの最終ペイロード （188 バイト超）

ビット 13:7 **HUBADDR**：ハブアドレス
このフィールドは、トランザクショントランスレータのハブのデバイスアドレスを保持します。

ビット 6:0 **PRTADDR**：ポートアドレス
このフィールドは、受信トランザクショントランスレータのポート番号です。

注： ***設定レジスタは USB OTG HS にのみ適用されます。***

57.14.28 OTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0～15、ここで x = Channel_number）

アドレスオフセット：0x508 + (Channel_number * 0x20)

リセット値：0x0000 0000

このレジスタは USB および AHB 関連イベントに関するチャネルのステータスを示します。*図 744* に示されています。アプリケーションは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、このレジスタを読み出す必要があります。アプリケーションがこれらのレジスタを読み出すためには、まず、ホスト全チャネル割り込みレジスタ（OTG_HAINT）を読み出して、ホストチャネル x 割り込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	FRM	FRM OR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBE RR	CHH	XFRC
					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DTERR**：データトグルエラー

ビット 9 **FRMOR**：フレームオーバーラン

ビット 8 **BBERR**：バブルエラー

ビット 7 **TXERR**：トランザクションエラー

USB 上で以下のエラーの 1 つが発生したことを示します。
CRC チェック失敗
タイムアウト
ビットスタッフエラー
偽の EOP

ビット 6 **NYET**：応答受信準備未完了割り込み

ビット 5 **ACK**：ACK レスポンス受信／送信割り込み

ビット 4 **NAK**：NAK レスポンス受信割り込み

ビット 3 **STALL**：STALL レスポンス受信割り込み

ビット 2 **AHBERR**：AHB エラー

このエラーは、AHB 読み出し／書き込み動作中に AHB エラーが発生したときに内部 DMA モードでのみ生成されます。
読み出し／書き込み操作アプリケーションは対応する DMA チャネルアドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット 1 **CHH**：チャネル停止

USB トランザクションエラーまたはアプリケーションによるディセーブルリクエストへの応答のいずれかの理由により、転送が異常終了したことを示します。

ビット 0 **XFRC**：転送完了

転送がエラーなく正常に完了しました。

57.14.29 OTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）（x = 0〜15、ここで x = Channel_number）

アドレスオフセット：0x50C + (Channel_number * 0x20)

リセット値：0x0000 0000

このレジスタは、前のセクションで説明した各チャネルステータスのマスクを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	AHBERRM	CHHM	XFRCM
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DTERRM**：データトグルエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 9 **FRMORM**：フレームオーバーランマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 8 **BBERRM**：バブルエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 7 **TXERRM**：トランザクションエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 6 **NYET**：応答受信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 5 **ACKM**：ACK レスポンス受信／送信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 4 **NAKM**：NAK レスポンス受信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 3 **STALLM**：STALL 応答受信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 **AHBERR**：AHB エラー
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 1 **CHHM**：チャネル停止マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 0 **XFRCM**：転送完了マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

57.14.30 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0～15、ここで x = Channel_number）

アドレスオフセット：0x510 + (Channel_number * 0x20)

リセット値：0x0000 0000

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **DPID**：データ PID

アプリケーションは、このフィールドを最初のトランザクションで使用する PID のタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：データ 0

01：データ 2

10：データ 1

11：SETUP（制御）／MDATA（非制御）

ビット 28:19 **PKTCNT**：パケットカウント

このフィールドは、アプリケーションによって、送信（OUT）または受信（IN）されるパケットの予想数でプログラムされます。

ホストは OUT/IN パケットの送信または受信が成功するたびに、このカウントをデクリメントします。このカウントがゼロに達すると、アプリケーションに割り込んで、正常完了を示します。

ビット 18:0 **XFRSIZ**：転送サイズ

OUT の場合、このフィールドは、ホストが転送時に送信するデータバイト数です。

IN の場合、このフィールドは、アプリケーションが転送のために確保したバッファサイズです。アプリケーションは、IN トランザクション（周期的および非周期的）の場合、このフィールドを最大パケットサイズの整数倍でプログラムすることが期待されます。

57.14.31 OTG ホストチャネル x DMA アドレスレジスタ（OTG_HCDMAx）（x = 0..15、ここで x = Channel_number）

アドレスオフセット：0x514 + (Channel_number * 0x20)

リセット値：0x0000 0000

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

ビット 31:0 **DMAADDR**：DMA アドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる、または格納される開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

注： ***設定レジスタは USB OTG HS にのみ適用されます。***

57.14.32 デバイスモードレジスタ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

57.14.33 OTG デバイス設定レジスタ（OTG_DCFG）

アドレスオフセット：0x800

リセット値：0x0220 0000

このレジスタは、電源投入後、または特定の制御コマンドあるいはエニュメレーションの後、コアをデバイスモードに設定します。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PERSCHIVL		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRATIM	XCVRDLY	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **PERSCHIVL**：周期的スケジュールインターバル

このフィールドは、周期的 IN エンドポイントデータのフェッチ用に内部 DMA エンジンが割り当てる必要のある時間を指定します。

周期的エンドポイントの数に基づいて、この値を（マイクロ）フレームの 25、50、75 % のいずれかに指定する必要があります。

- アクティブな周期的エンドポイントが存在するとき、内部 DMA エンジンは、フェッチ中の周期的 IN エンドポイントデータに所定の時間を割り当てます。
- アクティブな周期的エンドポイントが存在しない場合、内部 DMA エンジンは非周期的エンドポイントを処理し、このフィールドを無視します。
- （マイクロ）フレーム内で所定時間経過後 DMA は非周期的エンドポイントのフェッチへ切り替わります。

00：（マイクロ）フレームの25%

01：（マイクロ）フレームの50%

10：（マイクロ）フレームの75%

11：予約済み

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ERRATIM**：不規則エラー割り込みマスク

1：不規則エラー時のアーリーサスペンド割り込みをマスクします。

0：不規則エラー時にアーリーサスペンド割り込みが生成されます。

ビット 14 **XCVRDLY**：トランシーバ遅延

デバイスチャープ中の ULPI タイミングでの遅延を有効化／無効化します。

0：遅延無効（デフォルトのタイミングを使用）

1：デフォルトのタイミングに対して遅延を有効化（一部の ULPI PHY に必要）

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **PFIVL**：周期的フレームインターバル

周期的フレーム終了割り込みを使用してアプリケーションに通知しなければならないフレーム内の時間を示します。これを使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

00：フレームインターバルの 80%

01：フレームインターバルの 85%

10：フレームインターバルの 90%

11：フレームインターバルの 95%

ビット 10:4 **DAD**：デバイスアドレス

アプリケーションは、各 SetAddress 制御コマンドの後、このフィールドをプログラムする必要があります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **NZLSOHSK**：非ゼロ長ステータスOUTハンドシェイク

アプリケーションは、このフィールドを使用して、制御転送のステータスステージの OUT トランザクション中に、コアが非ゼロ長のデータパケットを受信したときに送信するハンドシェイクを選択できます。

1 :非ゼロ長ステータスの OUT トランザクション時には STALL ハンドシェイクを送信し、受信した OUT パケットをアプリケーションに送信しません。

0 :受信した OUT パケット（ゼロ長または非ゼロ長）をアプリケーションに送信し、デバイスエンドポイント制御レジスタの NAK および STALL ビットに基づいてハンドシェイクを送信します。

ビット 1:0 **DSPD**：デバイススピード

アプリケーションがコアにエニュメレーションを要求するスピード、またはアプリケーションがサポートできる最高スピードを示します。ただし、実際のバススピードは、チャープシーケンスの完了後に決まり、コアが接続される USB ホストのスピードに基づきます。

00：ハイスピード

01：予約済み

10：予約済み

11：フルスピード（USB 1.1 トランシーバのクロックは 48 MHz です）

57.14.34 OTG デバイス制御レジスタ（OTG_DCTL）

アドレスオフセット：0x804

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
				rw	w	w	w	w	rw	rw	rw	r	r	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DSBESLRJCT**：ディープスリープ BESL 拒否

コアは、プログラムされた BESL 閾値より大きい BESL 値を持つ LPM リクエストを拒否します。NYET 応答は、BESL 閾値より大きい BESL 値を持つ LPM トークンに送信されます。デフォルトでは、ディープスリープ BESL 拒否機能は無効です。

ビット 17:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **POPRGDNE**：パワーオンプログラミング終了

アプリケーションは、このビットを使用して、パワーダウンモードからのウェイクアップ後に、レジスタのプログラミングが終了したことを示します。

ビット 10 **CGONAK**：グローバル OUT NAK のクリア

このフィールドに書き込むと、グローバル OUT NAK がクリアされます。

ビット 9 **SGONAK**：グローバル OUT NAK のセット

このフィールドに書き込むと、グローバル OUT NAK がセットされます。
アプリケーションは、このビットを使用して、すべての OUT エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割り込みレジスタのグローバル OUT NAK 有効ビット（OTG_GINTSTS の GONAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 8 **CGINAK**：グローバル IN NAK のクリア

このフィールドに書き込むと、グローバル IN NAK がクリアされます。

ビット 7 **SGINAK**：グローバル IN NAK のセット

このフィールドに書き込むと、グローバル非周期的 IN NAK がセットされます。アプリケーションは、このビットを使用して、すべての非周期的 IN エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割り込みレジスタのグローバル IN NAK 有効ビット（OTG_GINTSTS の GINAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 6:4 **TCTL**：テスト制御

000：テストモードは無効です
001：Test_J モード
010：Test_K モード
011：Test_SE0_NAK モード
100：Test_Packet モード
101：Test_Force_Enable
その他：予約済み

ビット 3 **GONSTS**：グローバル OUT NAK のステータス

0 :ハンドシェイクは、FIFO ステータスと NAK および STALL ビットの設定に基づいて送信されます。
1 :使用可能なスペースに関係なく RxFIFO にはデータが書き込まれません。SETUP トランザクションを除き、すべてのパケットに NAK ハンドシェイクを送信します。すべてのアイソクロナス OUT パケットがドロップされます。

ビット 2 **GINSTS**：グローバル IN NAK ステータス

0 :ハンドシェイクは、送信 FIFO 内の使用可能なデータに基づいて送信されます。

1 :NAK ハンドシェイクは、送信 FIFO 内の使用可能なデータに関係なく、非周期的 IN エンドポイントに送出されます。

ビット 1 **SDIS**：ソフト切断

アプリケーションは、このビットを使用して USB OTG コアにソフト切断を実行することを伝えます。このビットがセットされている限り、ホストはデバイスが接続されているとは解釈せず、デバイスは USB 上で信号を受信しません。コアは、アプリケーションがこのビットをクリアするまで、切断状態にとどまります。

0 :通常動作。ソフト切断後にこのビットがクリアされると、コアは USB ホストに対してデバイス接続イベントを生成します。デバイスが再接続されると、USB ホストはデバイスのエニュメレーションをリスタートします。

1 :コアは、USB ホストに対して、デバイス切断イベントを生成します。

ビット 0 **RWUSIG**：リモートウェイクアップ信号

アプリケーションがこのビットをセットすると、コアはリモートシグナリングを実行して USB ホストをウェイクアップします。コアをサスペンド状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。USB 2.0 仕様で規定されているように、アプリケーションは、このビットをセットしてから 1 ms から 15 ms 後にクリアする必要があります。

LPM が有効で、コアが L1（スリープ）状態にあるとき、アプリケーションがこのビットをセットすると、コアは L1 リモートシグナリングを開始して、USB ホストをウエイクアップします。コアをスリープ状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。LPM 仕様で規定されているように、このビットはアプリケーションによってセットされてから 50 µs（$T_{L1DevDrvResume}$）後にハードウェアによって自動的クリアされます。アプリケーションは、前の LPM トランザクションからの bRemoteWake がゼロのときこのビットをセットする必要があります（GLPMCFG レジスタの REMWAKE ビットを参照）。

表 488 は USB ホストがデバイスの切断を検出するためにソフト切断（SDIS）ビットがセットされなければならない最短時間（デバイスの状態による）を示します。クロックジッタに対応するには、アプリケーションは指定された最短時間まで遅延を追加することが推奨されます。

表 488. ソフト切断のための最短時間

動作速度	デバイスの状態	最短時間
フルスピード	サスペンド状態	1 ms + 2.5 µs
フルスピード	アイドル	2.5 µs
フルスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	2.5 µs
ハイスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	125 µs

57.14.35 OTG デバイスステータスレジスタ（OTG_DSTS）

アドレスオフセット：0x808

リセット値：0x0000 0010

このレジスタは USB 関連イベントに関するコアのステータスを示します。デバイス全割り込みレジスタ（OTG_DAINT）からの割り込み時に読み出されなければなりません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FNSOF								Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSP STS
r	r	r	r	r	r	r	r					r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **DEVLNSTS**：デバイスラインステータス

USB データラインの現在のロジックレベルを示します。
ビット [23]：D+ のロジックレベル
ビット [22]：D- のロジックレベル

ビット 21:8 **FNSOF**：受信 SOF のフレーム番号

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **EERR**：不規則エラー

コアは、このビットをセットして、不規則エラーを報告します。
不規則エラーが発生すると、OTG_HS コントローラはサスペンド状態に移行し、OTG_GINTSTS レジスタのアーリーサスペンドビット（OTG_GINTSTS の ESUP ビット）でアプリケーションへの割り込みが生成されます。不規則エラーによるアーリーサスペンドがアサートされた場合、アプリケーションは、ソフト切断回復を実行するしかありません。

ビット 2:1 **ENUMSPD**：エニュメレーションされた速度

チャープシーケンスによるスピード検出後の OTG_HS コントローラのスピードを示します。
01：予約済み
10：予約済み
11：フルスピード（PHY クロックは 48 MHz で動作しています）
その他：予約済み

ビット 0 **SUSPSTS**：サスペンドステータス

デバイスモードでは、このビットは、USB 上でサスペンド状態が検出されている限り、セットされます。USB データライン上で 3 ms の間アクティビティが 1 つもなかった場合、コアはサスペンド状態に入ります。コアは、以下の場合にサスペンド状態から抜け出します。

– USB データライン上で何らかのアクティビティがあったとき。
– アプリケーションが OTG_DCTL レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）に書き込みを行ったとき。

57.14.36 OTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）

アドレスオフセット：0x810

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DIEPINTx レジスタとともに機能して、IN エンドポイントごとに 1 つの割り込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DIEPINTx レジスタの特定のステータスに対する IN エンドポイント割り込みはマスクできます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
								/							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	BMA	TXFURM	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
		rw				rw	rw		rw	rw	rw	rw		rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAKM**：NAK 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BIM**：BNA 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 8 **TXFURM**：FIFO アンダーランマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **INEPNEM**：IN エンドポイント NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 5 **INEPNMM**：EP 不一致付きで受信された IN トークンマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 4 **ITTXFEMSK**：TxFIFO エンプティ時に受信された IN トークンマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDM**：エンドポイントディセーブル割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 0 **XFRCM**：転送完了割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

57.14.37 OTG デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK）

アドレスオフセット：0x814

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DOEPINTx レジスタとともに機能して、OUT エンドポイントごとに 1 つの割り込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DOEPINTx レジスタの特定のステータスに対する OUT エンドポイント割り込みはマスクできます。ステータスビットはデフォルトでマスクされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NYET MSK	Res.	Res.	Res.	Res.	BOIM	TXFU RM	Res.	B2B STUP	Res.	OTEPD M	STUPM	Res.	EPDM	XFRC M
	rw					rw	rw		rw		rw	rw		rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **NYET**：NYET 割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 9 **BOIM**：BNA 割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 8 **TXFURM**：FIFO アンダーランマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUP**：連続 SETUP パケット受信マスク制御 OUT エンドポイントにのみ適用されます。これは です。

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 4 **OTEPDM**：.エンドポイント無効時 OUT トークン受信マスク制御 OUT エンドポイントにのみ適用されます。

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 3 **STUPM**：STUPM：SETUP フェーズ終了マスクコントロールエンドポイントにのみ適用されます。
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDM**：エンドポイントディセーブル割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 0 **XFRCM**：転送完了割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

57.14.38 OTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT）

アドレスオフセット：0x818

リセット値：0x0000 0000

エンドポイント上で重要なイベントが発生すると、OTG_DAINT レジスタは、OTG_GINTSTS レジスタのデバイス OUT エンドポイント割り込みビットまたはデバイス IN エンドポイント割り込みビット（それぞれ、OTG_GINTSTS の OEPINT または IEPINT ビット）を使用して、アプリケーションに割り込みをかけます。エンドポイントごとに 1 つの割り込みビットがあります（OUT エンドポイントおよび IN エンドポイントに対してそれぞれ最大 16 ビット）。双方向エンドポイントの場合、対応する IN および OUT 割り込みビットが使用されます。このレジスタのビットは、アプリケーションが対応するデバイスエンドポイント x 割り込みレジスタのビット（OTG_DIEPINTx/OTG_DOEPINTx）をセット／クリアしたときに、セット／クリアされます。

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

ビット 31:16 **OEPINT**：OUT エンドポイント割り込みビット
OUT エンドポイントチャネルあたり 1 ビット：
OUT エンドポイント 0 に対してビット 16、OUT エンドポイント 3 に対してビット 19

ビット 15:0 **IEPINT**：IN エンドポイント割り込みビット
IN エンドポイントあたり 1 ビット：
IN エンドポイント 0 に対してビット 0、IN エンドポイント 3 に対してビット 3

57.14.39 OTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK）

アドレスオフセット：0x81C

リセット値：0x0000 0000

OTG_DAINTMSK レジスタは、デバイスエンドポイント割り込みレジスタとともに機能して、デバイスエンドポイント上にイベントが発生すると、アプリケーションに割り込みます。ただし、その割り込みに対応する OTG_DAINT レジスタのビットはセットされたままです。

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

ビット 31:16 **OEPM**：OUT EP 割り込みマスクビット
OUT エンドポイント（EP）あたり 1 ビット：
OUT EP 0 に対してビット 16、OUT EP 3 に対してビット 19
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 15:0 **IEPM**：IN EP 割り込みマスクビット
IN エンドポイントあたり 1 ビット：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割り込みはマスクされます。
1：割り込みはマスクされません。

57.14.40 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）

アドレスオフセット：0x0828

リセット値：0x0000 17D7

このレジスタは SRP 中の V_{BUS} 放電時間（V_{BUS} パルシング後）を指定します。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **VBUSDT**：デバイス V_{BUS} 放電時間
SRP 中の V_{BUS} 放電時間（V_{BUS} パルシング後）を指定します。この値は以下のようになります。
V_{BUS} 放電時間（PHY クロック / 1024）
この値は V_{BUS} の負荷によって調整の必要があるかもしれません。

57.14.41 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）

アドレスオフセット：0x082C

リセット値：0x0000 05B8

このレジスタは SRP 期間中の V_{BUS} パルシング時間を指定します。

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DVBUSP**：デバイス V_{BUS} パルシング時間

SRP 中の V_{BUS} パルシング時間を指定します。この値は以下のようになります。

V_{BUS} パルシング時間（PHY クロック / 1024）

57.14.42 OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）

アドレスオフセット：0x0830

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	ARPEN	Res.	RXTHRLEN									RXTH REN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXTHRLEN									ISOT HREN	NONIS OTH REN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **ARPEN**：アービタパーキングイネーブル

このビットは、IN エンドポイントに対する内部DMA アービタパーキングを制御します。閾値が有効化され、かつこのビットに 1 が設定されると、アービタは USB でトークンが受信された IN エンドポイントにパークします。これは、アンダーラン状態を回避するために実行されます。デフォルトではパーキングは有効化されています。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25：17 **RXTHRLEN**：受信閾値長

このフィールドは、受信閾値サイズをワード単位で指定します。また、このフィールドはコアが AHB で送信を開始できる前に USB で受信するデータ量も指定します。閾値長は、少なくとも 8 ワードある必要があります。RXTHRLEN の推奨値は、設定された AHB バースト長（OTG_GAHBCFG の HBSTLEN ビット）と同じです。

ビット 16 **RXTHREN**：受信閾値イネーブル

このビットをセットすると、コアは受信方向で閾値を有効化します。

ビット 15：11　予約済みであり、リセット値に保持する必要があります。

ビット 10:2 **TXTHRLEN**：送信閾値長

このフィールドは、送信閾値サイズをワード単位で指定します。このフィールドは、コアが USB で送信を開始できる前に、対応するエンドポイント送信 FIFO に入力するデータバイト数を指定します。閾値長は、少なくとも 8 ワードある必要があります。このフィールドは、アイソクロナスと非アイソクロナスの IN エンドポイント閾値を制御します。TXTHRLEN の推奨値は、設定された AHB バースト長（OTG_GAHBCFG の HBSTLEN ビット）と同じです。

ビット 1 **ISOTHREN**：ISO IN エンドポイント閾値イネーブル

このビットをセットすると、コアはアイソクロナス IN エンドポイントで閾値を有効化します。

ビット 0 **NONISOTHREN**：非アイソクロナス IN エンドポイント閾値イネーブル

このビットをセットすると、コアは非アイソクロナス IN エンドポイントで閾値を有効化します。

注：　設定レジスタは USB OTG HS にのみ適用されます。

57.14.43 OTG デバイス各エンドポイント割り込みレジスタ（OTG_DEACHINT）

アドレスオフセット：0x0838

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1 INT	Res.
														r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1 INT	Res.
														r	

ビット 31:18　予約済みであり、リセット値に保持する必要があります。

ビット 17 **OEP1INT**：OUT エンドポイント 1 割り込みビット

ビット 16:2　予約済みであり、リセット値に保持する必要があります。

ビット 1 **IEP1INT**：IN エンドポイント 1 割り込みビット

ビット 0　予約済みであり、リセット値に保持する必要があります。

注：　設定レジスタは USB OTG HS にのみ適用されます。

57.14.44 OTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ（OTG_DIEPEMPMSK）

アドレスオフセット：0x834

リセット値：0x0000 0000

このレジスタは、IN エンドポイント FIFO エンプティ割り込みの生成（TXFE_OTG_DIEPINTx）を制御するために使用されます。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INEPTXFEM**：IN EP Tx FIFO エンプティ割り込みマスクビット
これらのビットは OTG_DIEPINTx に対するマスクビットとして機能します。
IN EP あたり TXFE 割り込み 1 ビット：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割り込みはマスクされます。
1：割り込みはマスクされません。

57.14.45 OTG デバイス各エンドポイント割り込みレジスタマスク（OTG_DEACHINTMSK）

アドレスオフセット：0x083C

リセット値：0x0000 0000

エンドポイント 1 の IN に対して 1 個の割り込みビットが、エンドポイント 1 の OUT に対して 1 個の割り込みビットが、それぞれ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INTM	Res.
														rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INTM	Res.
														rw	

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **OEP1INTM**：OUT エンドポイント 1 割り込みマスクビット

ビット 16:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **IEP1INTM**：IN エンドポイント 1 割り込みマスクビット

ビット 0 予約済みであり、リセット値に保持する必要があります。

注：　設定レジスタは USB OTG HS にのみ適用されます。

57.14.46 OTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 0～8、ここで x = Endpoint_number）

アドレスオフセット：0x900 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w	rw	rw	rw	rw	rw/rs		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。
コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了
– エンドポイント無効
– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割り込みを待つ必要があります。コアは、エンドポイント無効割り込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントのエンドポイントイネーブルがすでにセットされている場合のみ、このビットをセットしなければなりません。

ビット 29 **SODDFRM**：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。
このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

割り込み／バルク IN エンドポイントにのみ適用されます。
このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス IN エンドポイントにのみ適用されます。
このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、送信完了割り込み時、またはエンドポイントで SETUP パケットが受信された後、OUT エンドポイントに対してこのビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 **TXFNUM**：Tx FIFO 番号

これらのビットは、このエンドポイントに関連する FIFO 番号を指定します。各有効 IN エンドポイントは、別々の FIFO 番号にプログラムされなければなりません。

このフィールドは、IN エンドポイントについてのみ有効です。

ビット 21 **STALL**：STALL のハンドシェイク

非制御の非アイソクロナス IN エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプは rs です）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **EPTYP**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、IN エンドポイントでのデータの送信を停止します。

アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、ゼロ長のデータパケットを送出します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割り込み／バルク IN エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：データ 0

1：データ 1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

57.14.47 OTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）

アドレスオフセット：0xB00

リセット値：0x0000 8000

このセクションでは、OTG_DOEPCTL0 レジスタについて説明します。非ゼロコントロールエンドポイントは、エンドポイント 1～3 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAK STS	Res.
w	r			w	w					rs	rw	r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
r														r	r

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイント 0　上でデータ送信を開始します。コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了
– エンドポイント無効
– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、コントロール OUT エンドポイント 0 を無効にできません。

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、送信完了割り込み時、または SETUP パケットを受信した後に、このビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。NAK ビット、すなわち、グローバル OUT NAK がこのビットとともにセットされた場合は、STALL ビットが優先します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP**：エンドポイントタイプ

制御のために 2'b00 にハードコード化されています。

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、Rx FIFO に受信パケットを収容できるスペースがある場合でも、コアはデータの受信を停止します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **USBAEP**：USB アクティブエンドポイント

このビットは常に 1 にセットされ、コントロールエンドポイント 0 がすべての設定およびインタフェースにおいて常にアクティブであることを示します。

ビット 14:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MPSIZ**：最大パケットサイズ

制御 OUT エンドポイント 0 の最大パケットサイズは、制御 IN エンドポイント 0 でプログラムされるサイズと同じです。

00：64 バイト
01：32 バイト
10：16 バイト
11：8 バイト

57.14.48 OTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1〜8、ここで x = Endpoint_number）

OUT エンドポイントのアドレスオフセット：0xB00 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID/ SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w					rw/rs	rw	rw	rw	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル

IN および OUT エンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了
– エンドポイント無効
– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割り込みを待つ必要があります。コアは、エンドポイント無効割り込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントのエンドポイントイネーブルがすでにセットされている場合のみ、このビットをセットしなければなりません。

ビット 29 **SD1PID**：DATA1 PID の設定

割り込み／バルク IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA1 に設定されます。

SODDFRM：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

割り込み／バルク OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、転送完了割り込み時、または SETUP パケットを受信した後に、OUT エンドポイントに対してこのビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

非コントロールの非アイソクロナス OUT エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプは rs です）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

コアは、Rx FIFO に受信パケットを収容するためのスペースがある場合でも、OUT エンドポイントでのデータの受信を停止します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN および OUT エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割り込み／バルク OUT エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：データ 0

1：データ 1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

57.14.49 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）（x = 0〜8、ここで x = Endpoint_number）

アドレスオフセット：0x908 + (Endpoint_number * 0x20)

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。*図 744* に示されています。アプリケーションは、コア割り込みレジスタの IN エンドポイント割り込みビット（OTG_GINTSTS の IEPINT ビット）がセットされたとき、このレジスタを読み出さなければなりません。アプリケーションがこのレジスタを読み出すためには、まず、デバイス全エンドポイント割り込み（OTG_DAINT）レジスタを読み出して、デバイスのエンドポイント x 割り込みレジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAK	BERR	PKTDRPSTS	Res.	BNA	TXFIFOUDRN	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
		rc_w1	rc_w1	rc_w1		rc_w1	rc_w1	r	r		rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAK**：USB OTG HS の NAK 入力

コアは、デバイスが NAK を送受信したとき、この割り込みを生成します。

アイソクロナス IN エンドポイントの場合、Tx FIFO にデータがないためにゼロ長パケットが送信されたとき、この割り込みが生成されます。

ビット 12 **BERR**：バブルエラー割り込み

ビット 11 **PKTDRPSTS**：パケットドロップステータス

このビットは、ISOC OUT パケットがドロップされたことをアプリケーションに示します。このビットには

対応するマスクビットがなく、割り込みは生成されません。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNA**：バッファ使用不能割り込み

ホストビジーまたは DMA 終了のように、ディスクリプタがレディでなくコアで処理できないとき、コアはこの割り込みを生成します。

ビット 8 **TXFIFOUDRN**：送信 FIFO アンダーラン（TxfifoUndrn）

コアはこのエンドポイントの送信 FIFO アンダーラン状態を検出するとこの割り込みを生成します。依存性：この割り込みは、閾値が有効化されている場合にのみ有効です。

ビット 7 **TXFE**：送信 FIFO エンプティ

この割り込みは、このエンドポイントの Tx FIFO が半エンプティまたは完全エンプティのいずれかであるとき、アサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

ビット 6 **INEPNE**：IN エンドポイント NAK 有効

このビットは、アプリケーションが OTG_DIEPCTLx レジスタの CNAK ビットに書き込むことによって IN エンドポイント NAK をクリアするとクリアできます。

この割り込みは、コアが（アプリケーションまたはコアによって）設定された NAK ビットをサンプリングしたことを示します。この割り込みは、アプリケーションによってセットされた IN エンドポイント NAK ビットがコアで有効になったことを示します。

この割り込みは、必ずしも NAK ハンドシェイクが USB に送出されることを保証しません。STALL ビットが NAK ビットに優先します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ITTXFE**：TxFIFO がエンプティ時に受信された IN トークン

非アイソクロナス IN エンドポイントにのみ適用されます。

関連する Tx FIFO（周期的/非周期的）が空のときに IN トークンが受信されたことを示します。この割り込みは、IN トークンが受信されたエンドポイント上でアサートされます。

ビット 3 **TOC**：タイムアウト条件

コントロール IN エンドポイントにのみ適用されます。

このエンドポイント上の最後の IN トークンに対して、コアが USB 上でタイムアウト条件を検出したことを示します。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDISD**：エンドポイントディセーブル割り込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割り込み

このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

57.14.50 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）（x = 0〜8、ここで x = Endpoint_number）

アドレスオフセット：0xB08 + (Endpoint_number * 0x20)

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。*図 744* に示されています。アプリケーションは、OTG_GINTSTS レジスタの OUT エンドポイント割り込みビット（OTG_GINTSTS の OEPINT ビット）がセットされたとき、このレジスタを読み出さなければなりません。アプリケーションがこのレジスタを読み出すためには、まず、OTG_DAINT レジスタを読み出して、OTG_DOEPINTx レジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2B STUP	Res.	OTEP DIS	STUP	Res.	EP DISD	XFRC
									rc_w1		rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUP**：連続 SETUP パケットの受信
制御 OUT エンドポイントにのみ適用されます。
このビットは、コアがこの特定のエンドポイントで 3 つを超える連続 SETUP パケットを受信したことを示します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **OTEPDIS**：エンドポイント無効時 OUT トークン受信
コントロール OUT エンドポイントにのみ適用されます。
エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったことを示します。この割り込みは、OUT トークンが受信されたエンドポイントでアサートされます。

ビット 3 **STUP**：SETUP フェーズ終了
コントロール OUT エンドポイントにのみ適用されます。
コントロールエンドポイントの SETUP フェーズが完了し、現在の制御転送のために、これ以上、連続 SETUP パケットは受信されなかったことを示します。この割り込みにより、アプリケーションは、受信したSETUP データパケットをデコードできます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDISD**：エンドポイントディセーブル割り込み
このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割り込み
このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

57.14.51 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）

アドレスオフセット：0x910

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。デバイスコントロールエンドポイント 0 制御レジスタのエンドポイントイネーブルビット（OTG_DIEPCTL0 の EPENA）を使用して、エンドポイント 0 が有効にされると、コアは、このレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

非ゼロエンドポイントは、エンドポイント 1～3 のレジスタを使用します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
									rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:19 **PKTCNT**：パケットカウント

エンドポイント 0 のデータ転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

57.14.52 OTG デバイスチャネル x DMA アドレスレジスタ（OTG_DIEPDMAx）（x = 0～15、ここで x = Channel_number）

アドレスオフセット：0x914 + (Channel_number * 0x20)

リセット値：0x0000 0000

注： ***設定レジスタは USB OTG HS にのみ適用されます。***

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAADDR															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DMAADDR**：DMA アドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

57.14.53 OTG デバイスチャネル x DMA アドレスレジスタ（OTG_DOEPDMAx）（x = 0〜15、ここで x = Channel_number）

アドレスオフセット：0xB14 + (Channel_number * 0x20)

リセット値：0x0000 0000

注： ***設定レジスタは USB OTG HS にのみ適用されます。***

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

ビット 31:0 **DMAADDR**：DMA アドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

57.14.54 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）

アドレスオフセット：0xB10

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTL0 レジスタのエンドポイントイネーブルビット（OTG_DOEPCTL0 のEPENA ビット）を使用してエンドポイント 0 が有効にされると、コアは、このレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

非ゼロエンドポイントは、エンドポイント 1〜8 のレジスタを使用します。

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **STUPCNT**：SETUP パケットカウント

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **PKTCNT**：パケットカウント

このフィールドは、パケットが Rx FIFO に書き込まれた後、ゼロまでデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

57.14.55 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1～8、ここで x = Endpoint_number）

アドレスオフセット：0x910 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DIEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DIEPCTLx の EPENA ビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **MCNT**：マルチカウント

周期的 IN エンドポイントの場合、このフィールドは、USB でフレームごとに送信されなければならないパケット数を示します。コアは、このフィールドを使用して、アイソクロナス IN エンドポイントのデータ PID を計算します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:0 **XFRSIZ**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

57.14.56 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）（x = 0〜8、ここで x = Endpoint_number）

IN エンドポイントのアドレスオフセット：0x918 +（Endpoint_number × 0x20）この読み出し専用レジスタは、デバイス IN エンドポイント Tx FIFO の空きスペース情報を含みます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INEPTFSAV															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

31:16 予約済みであり、リセット値に保持する必要があります。

15:0 **INEPTFSAV**：IN エンドポイント Tx FIFO の使用可能スペース

エンドポイントの Tx FIFO で使用できる空きスペースの量を示します。

値は 32 ビットワード単位です。

0x0：エンドポイント Tx FIFO はフルです。

0x1：1 ワードが使用可能です。

0x2：2 ワードが使用可能です。

0xn：n ワードが使用可能です。

その他：予約済み

57.14.57 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1～8、ここで x =Endpoint_number）

アドレスオフセット：0xB10 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DOEPCTLx の EPENA ビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ		
	r/rw	r/rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **RXDPID**：受信データ PID

アイソクロナス OUT エンドポイントにのみ適用されます。

これは、このエンドポイントの最後のパケットで受信されたデータの PID です。

00：データ 0

01：データ 2

10：データ 1

11：MDATA

STUPCNT：SETUP パケットカウント

コントロール OUT エンドポイントにのみ適用されます。

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が Rx FIFO に書き込まれるたびにデクリメントされます。

ビット 18:0 **XFRSIZ**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

57.14.58 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

アドレスオフセット：0xE00

リセット値：0x0x200B 8000

このレジスタは、ホストモードとデバイスモードで使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHY SLEEP	ENL1 GTG	PHY SUSP	Res.	Res.	GATE HCLK	STPP CLK
								r	r	rw	r			rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **SUSP**：ディープスリープ

このビットは、L1 状態時に PHY がディープスリープであることを示します。

ビット 6 **PHYSLEEP**：スリープ状態の PHY

このビットは、PHY がスリープ状態であることを示します。

ビット 5 **ENL1GTG**：スリープクロックゲーティング有効化

このビットがセットされている場合、コアが utmi_l1_suspend_n をアサートできなければ、スリープ状態でコアの内部クロックゲーティングが有効化されます。このビットがセットされていない場合、PHY クロックはスリープ状態ではゲートされません。

ビット 4 **PHYSUSP**：PHYをサスペンド

PHY がサスペンドされたことを示します。このビットは、アプリケーションが STPPCLK ビットをセットした後、PHY がサスペンドされると更新されます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **GATEHCLK**：HCLK をゲート

USB がサスペンドされるか、セッションが有効でないとき、アプリケーションは、このビットをセットして、AHB スレーブとマスタおよびウェイクアッププロジック以外のモジュールへの HCLK をゲーティングします。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

ビット 0 **STPPCLK**：PHY クロックを停止

USB がサスペンドされるか、セッションが有効でないか、デバイスが切断されると、アプリケーションは、このビットをセットして、PHY クロックを停止します。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

57.14.59 OTG_HS レジスタマップ

次の表に、USB OTG のレジスタマップとリセット値を示します。

表 489. OTG_HS レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	OTG_GOTGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS	Res.	Res.	Res.	EHEN	DHNPEN	HSHNPEN	HNPRQ	HNGSCS	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	SRQ	SRQSCS
	リセット値												0	0	0	0	1				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	OTG_GOTGINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDCHNG	DBCDNE	ADTOCHG	HNGDET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
	リセット値												0	0	0	0								0	0						0		
0x008	OTG_GAHBCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	DMAEN	HBSTLEN				GINTMSK
	リセット値																								0	0		0	0	0	0	0	0
0x00C	OTG_GUSBCFG	Res.	FDMOD	FHMOD	Res.	Res.	Res.	ULPIIPD	PTCI	PCCI	TSDPS	ULPIEVBUSI	ULPIEVBUSD	ULPICSM	ULPIAR.	ULPIFSL.	Res.	PHYLPC.	Res.	TRDT				HNPCAP	SRPCAP	Res.	PHYSEL	Res.	Res.	Res.	TOCAL		
	リセット値		0	0				0	0	0	0	0	0	0	0	0		0		0	1	0	1	0	0		1				0	0	0
0x010	OTG_GRSTCTL	AHBIDL	DMAREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFNUM					TXFFLSH	RXFFLSH	Res.	Res.	PSRST	CSRST
	リセット値	1	0																				0	0	0	0	0	0	0			0	0
0x014	OTG_GINTSTS	WKUINT	SRQINT	DISCINT	CIDSCHG	Res.	PTXFE	HCINT	HPRTINT	Res.	DATAFSUSP	IPXFR/INCOMPISOOUT	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	1		1	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0
0x018	OTG_GINTMSK	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	OTG_GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSR（デバイスモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	OTG_GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSPR（デバイスモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	OTG_GRXFSIZ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFD															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x028	OTG_HNPTXFSIZ/ OTG_DIEPTXF0	NPTXFD/TX0FD																NPTXFSA/TX0FSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x02C	OTG_HNPTXSTS	Res.	NPTXQTOP							NPTQXSAV								NPTXFSAV															
	リセット値		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x030	OTG_GI2CCTL	BSYDNE	RW	Res.	I2CDATSE	I2CDEVADR		Res.	ACK	I2CEN	ADDR							REGADDR							RWDATA								
	リセット値	0	0		0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	OTG_GCCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCDEN	BCDEN	PWRDWN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET
	リセット値											0	0	0	0	0	0													0	0	0	0
0x03C	OTG_CID	PRODUCT_ID																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0x054	OTG_GLPMCFG	Res.	Res.	Res.	ENBESL	LPMR CNTSTS			SNDLPM	LPM RCNT			LPMCHIDX				L1RSMOK	SLPSTS	LPM RSP		L1DSEN	BESLTHRS			L1SSEN	REMWAKE	BESL			LPMACK	LPMEN		
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	OTG_HPTXFSIZ	PTXFSIZ																PTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x104	OTG_DIEPTXF1	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x108	OTG_DIEPTXF2	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
. . . .	. . . .	.																															
0x244	OTG_DIEPTXF7	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0x400	OTG_HCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLSPCS	
	リセット値																														0	0	0
0x404	OTG_HFIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLDCTRL	FRIVL															
	リセット値																0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0	0
0x408	OTG_HFNUM	FTREM																FRNUM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x410	OTG_HPTXSTS	PTXQTOP								PTXQSAV								PTXFSAVL															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x414	OTG_HAINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINT															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x418	OTG_HAINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINTM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x440	OTG_HPRT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD		PTCTL				PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS
	リセット値														0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x504	OTG_HCSPLT0	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x500	OTG_HCCHAR0	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x508	OTG_HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x510	OTG_HCTSIZ0	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50C	OTG_HCINTMSK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x514	OTG_HCDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x520	OTG_HCCHAR1	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x528	OTG_HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x52C	OTG_HCINTMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x530	OTG_HCTSIZ1	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. . . .																															
. . . .	. . . .	. . . .																															
. . . .	. . . .	. . . .																															
. . . .	. . . .	. . . .																															
0x6E0	OTG_HCCHAR15	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
. . . .	. . . .	.																															
0x6E4	OTG_ HCSPLT15	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x6EC	OTG_ HCINTMSK15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
. . . .	. . . .	.																															
0x6F0	OTG_ HCTSIZ15	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x6F4	OTG_ HCDMA15	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
. . . .	. . . .	.																															
0x7A8	OTG_ HCINT15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x800	OTG_ DCFG	Res.	Res.	Res.	Res.	Res.	Res.	PERSCHIVL		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM	XCVRDLY	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値																	0	0		0	0	0	0	0	0	0	0	0		0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x804	OTG_DCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.	Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
	リセット値														0							0	0	0	0	0	0	0	0	0	1	0
0x808	OTG_DSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS	FNSOF															Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSPSTS
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x810	OTG_DIEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	Res.	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
	リセット値																				0						0	0	0	0		0	0
0x814	OTG_DOEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NYETMSK	Res.	Res.	Res.	Res.	BOIM	TXFURM	Res.	B2BSTUP	Res.	OTEPDM	STUPM	Res.	EPDM	XFRCM
	リセット値																		0					0			0		0	0		0	0
0x818	OTG_DAINT	OEPINT																IEPINT															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x81C	OTG_DAINTMSK	OEPM																IEPM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x828	OTG_DVBUSDIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBUSDT															
	リセット値																	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1
0x82C	OTG_DVBUSPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DVBUSP															
	リセット値																	0	0	0	0	0	1	0	1	1	0	1	1	1	0	0	0
0x830	OTG_DTHRCTL	Res.	Res.	Res.	Res.	ARPEN	Res.	RXTHRLEN									RXTHREN	Res.	Res.	Res.	Res.	TXTHRLEN										ISOTHREN	NONISOTHREN
	リセット値					0											0					0	0	0	0	0	0	0	0	0	0	0	0
0x834	OTG_DIEPEMPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTXFEM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x838	OTG_DEACHINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INT	Res.
	リセット値															0																0	

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x83C	OTG_DEACHINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INTM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INTM	Res.
	リセット値															0																0	
0x900	OTG_DIEPCTL0	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x908	OTG_DIEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
0x910	OTG_DIEPTSIZ0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKT CNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値												0	0													0	0	0	0	0	0	0
0x914	OTG_DIEPDMA	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x918	OTG_DTXFSTS0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x920	OTG_DIEPCTL1	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x928	OTG_DIEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
0x930	OTG_DIEPTSIZ1	Res.	MCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x938	OTG_DTXFSTS1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x940	OTG_ DIEPCTL2	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . . .	. . . . .	.																															
. . . . .	. . . . .	.																															
. . . . .	. . . . .	.																															
. . . . .	. . . . .	.																															
0x9E0	OTG_ DIEPCTL7	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . . .	. . . . .	.																															
0x9E8	OTG_ DIEPINT7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
. . . . .	. . . . .	.																															
0x9F0	OTG_ DIEPTSIZ7	Res.	MCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . . .	. . . . .	.																															
0x9F8	OTG_ DTXFSTS7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB00	OTG_ DOEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0					0	0	0	0	0		1														0	0
0xB08	OTG_ DOEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
0xB10	OTG_ DOEPTSIZ0	Res.	STUPCNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値		0	0										0													0	0	0	0	0	0	0
0xB14	OTG_ DOEPDMA	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB20	OTG_ DOEPCTL1	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0xB28	OTG_ DOEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
0xB30	OTG_ DOEPTSIZ1	Res.	RXDPID/ STUPCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB50	OTG_ DOEPTSIZ2	Res.	RXDPID/ STUPCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. . . .																															
. . . .	. . . .	. . . .																															

表 489. OTG_HS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xBE0	OTG_ DOEPCTL7	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
. . . .	. . . .	.																															
. . . .	. . . .	.																															
0xBE8	OTG_ DOEPINT7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
. . . .	. . . .	.																															
0xBF0	OTG_ DOEPTSIZ7	Res.	RXDPID/ STUPCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE00	OTG_ PCGCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHYSLEEP	ENL1GTG	PHYSUSP	Res.	Res.	GATEHCLK	STPPCLK
	リセット値																									0	0	0	0			0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

57.15 OTG_HS プログラミングモデル

57.15.1 コアの初期化

アプリケーションはコア初期化シーケンスを実行する必要があります。パワーアップ中にケーブルが接続された場合、OTG_GINTSTS レジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）はモードを反映します。OTG_HS コントローラは、A プラグが接続されたときにはホストモードに、B プラグが接続されたときにはデバイスモードに入ります。

このセクションでは、電源投入後の OTG_HS コントローラの初期化について説明します。アプリケーションは、ホストモードかデバイスモードかに関係なく、初期化シーケンスに従う必要があります。コアのすべてのグローバルレジスタは、コアの設定に従って以下の手順で初期化されます。

1. OTG_GAHBCFG レジスタの以下のフィールドをプログラムします。
   - グローバル割り込みマスクビット GINTMSK = 1
   - Rx FIFO ノットエンプティ（OTG_GINTSTS の RXFLVL ビット）
   - 周期的 Tx FIFO エンプティレベル
2. OTG_GUSBCFG レジスタの以下のフィールドをプログラムします。
   - HNP 対応ビット
   - SRP 対応ビット
   - OTG_HS タイムアウト較正フィールド
   - USB ターンアラウンド時間フィールド
3. ソフトウェアは、OTG_GINTMSK レジスタの以下のビットのマスクを解除する必要があります。

   OTG 割り込みマスク

   モード不一致割り込みマスク
4. ソフトウェアは、OTG_GINTSTS レジスタの CMOD ビットを読み出すことによって、OTG_HS コントローラがホストモードで動作中か、デバイスモードで動作中かを判別することができます。

57.15.2 ホストの初期化

コアをホストとして初期化するには、アプリケーションは、以下の手順を実行しなければなりません。

1. OTG_GINTMSK レジスタの HPRTINT ビットをプログラムして、マスクを解除します。
2. OTG_HCFG レジスタをプログラムして、フルスピードホストを選択します。
3. OTG_HPRT レジスタの PPWR ビットを 1 にプログラムします。これによって USB 上で V_{BUS} が駆動されます。
4. OTG_HPRT0 の PCDET 割り込みを待ちます。これは、デバイスがポートに接続されていることを示します。
5. OTG_HPRT レジスタの PRST ビットを 1 にプログラムします。これによってリセットプロセスが開始されます。
6. リセットプロセスが完了するまで、少なくとも 10 ms 待ちます。
7. OTG_HPRT レジスタの PRST ビットを 0 にプログラムします。
8. OTG_HPRT レジスタの PENCHNG 割り込みを待ちます。
9. OTG_HPRT レジスタの PSPD ビットを読み出して、エニュメレーションされたスピードを取得します。
10. HFIR レジスタを、選択された PHY クロック 1 に対応する値でプログラムします。
11. OTG_HCFG レジスタの FSLSPCS フィールドを、手順 9 で検出したデバイスの速度に従ってプログラムします。FSLSPCS が変更されている場合は、ポートのリセットを行う必要があります。
12. OTG_GRXFSIZ レジスタをプログラムして、受信 FIFO のサイズを選択します。
13. OTG_HNPTXFSIZ レジスタをプログラムして、非周期的トランザクションの非周期的送信 FIFO のサイズと開始アドレスを選択します。
14. OTG_HPTXFSIZ レジスタをプログラムして、周期的トランザクションの周期的送信 FIFO のサイズと開始アドレスを選択します。

デバイスと通信するには、システムソフトウェアは、少なくとも 1 つのチャネルを初期化し、有効にする必要があります。

57.15.3 デバイスの初期化

アプリケーションは、パワーアップ時、またはホストモードからデバイスモードへの変更後、以下の手順を実行して、コアをデバイスとして初期化する必要があります。

1. OTG_DCFG レジスタの以下のフィールドをプログラムします。
   - デバイススピード
   - 非ゼロ長ステータスOUTハンドシェイク
2. OTG_GINTMSK レジスタをプログラムして、次の割り込みのマスクを解除します。
   - USB リセット
   - エニュメレーション終了
   - アーリーサスペンド
   - USB サスペンド
   - SOF
3. OTG_GINTSTS の USBRST 割り込みを待ちます。これは、この割り込みの受信時、約 10ms 続くリセットが USB で検出されたことを示します。

OTG_GINTSTS の ENUMDNE 割り込みを待ちます。この割り込みは、USB 上でリセットが終了したことを示します。この割り込みを受信したとき、アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーション速度を判別し、*2632 ページのエニュメレーション完了時のエンドポイント初期化*にリストされている手順を実行する必要があります。

この時点で、デバイスは、SOF パケットを受け入れて、コントロールエンドポイント 0 でコントロール転送を実行する準備ができたことになります。

57.15.4 DMA モード

OTG ホストは AHB マスタインタフェースを使って、送信パケットデータ （AHB から USB へ）をフェッチし、データ更新 （USB から AHB へ）を受信します。AHB マスタはプログラムされた DMA アドレス（ホストモードでの OTG_HCDMAx レジスタとペリフェラルモードでの OTG_DIEPDMAx/OTG_DOEPDMAx レジスタ）を使って、データバッファにアクセスします。

57.15.5 ホストプログラミングモデル

チャネルの初期化

アプリケーションが接続されたデバイスと通信するためには、1 つまたは複数のチャネルを初期化する必要があります。チャネルを初期化して、有効にするには、アプリケーションは、以下の手順を実行する必要があります。

1. OTG_GINTMSK レジスタをプログラムして、以下の割り込みのマスクを解除します。
2. チャネル割り込み
   - OUT トランザクションの非周期的送信 FIFO エンプティ割り込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
   - OUT トランザクションの非周期的送信 FIFO 半エンプティ割り込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
3. OTG_HAINTMSK レジスタをプログラムして、選択されたチャネルの割り込みのマスクを解除します。
4. OTG_HCINTMSK レジスタをプログラムして、ホストチャネル割り込みレジスタで指定された、所定のトランザクション関連割り込みのマスクを解除します。
5. 選択されたチャネルの OTG_HCTSIZx レジスタを、合計転送サイズ（バイト数）およびショートパケットを含む予想されるパケット数でプログラムします。アプリケーションは、PIDフィールドを初期データ PID（最初の OUT トランザクションで使用される PID、または最初のIN トランザクションから期待される PID）でプログラムする必要があります。
6. 選択されたチャネルの OTG_HCCHARx レジスタを、タイプ、スピード、方向など、デバイスのエンドポイントの特性でプログラムします。（チャネルは、アプリケーションがパケットを送信または受信する準備ができている場合にのみ、チャネルイネーブルビットを 1 にセットすることで有効にできます）。
7. OTG_HCSPLTx レジスタ内の選択したチャネルにハブアドレスとポートアドレスをプログラムします （スプリットトランザクションの場合）。
8. OTG_HCDMAx レジスタ内の選択したチャネルにバッファ開始アドレスをプログラムします（DMA トランザクションのみ）。

チャネルの停止

アプリケーションは、OTG_HCCHARx レジスタの CHDIS および CHENA ビットを 1 にセットすることによって、任意のチャネルを無効にできます。これにより、OTG_HS ホストは、要求されたリクエスト（ある場合）を一掃して、チャネル停止割り込みを生成できます。アプリケーションは、チャネルを他のトランザクションに再割当てする前に、OTG_HCINTx 内の CHH 割り込みを待つ必要があります。OTG_HS ホストは、USB 上ですでに転送が開始しているトランザクションに対しては割り込みを行いません。

DMA モード動作でチャネルを無効化するときは、アプリケーションはリクエストキュー内のスペースをチェックする必要はありません。OTG_HS ホストは、調停時の無効化されたチャネルの回転に関するディスエーブルリクエストを書き込むスペースがあるかをチェックします。

その間、OTG_HCCHARx の CHDIS ビットに 1 が設定されたとき、要求された全リクエストはリクエストキューからドロップされます。

チャネルを無効にする前に、アプリケーションは、非周期的リクエストキュー（非周期的チャネルを無効にするとき）、または周期的リクエストキュー（周期的チャネルを無効にするとき）に使用可能な空きスペースが少なくとも 1 つ以上存在することを確認する必要があります。アプリケーションは、リクエストキューがフル状態のときには（チャネルを無効化する前）、OTG_HCCHARx レジスタの CHDIS ビットを 1 にセットし CHENA ビットを 0 にクリアするだけで、ポストされたリクエストを一掃できます。

アプリケーションは、以下のいずれかの条件のとき、チャネルを無効にすることが期待されます。

1. IN または OUT チャネルで、OTG_HCINTx レジスタの STALL、TXERR、BBERR、または DTERR 割り込みが受信されたとき。アプリケーションは、停止割り込みを受信する前に、同じチャネルについての他の割り込み（DTERR、Nak、Data、TXERR）を受信できなければなりません。
2. 非周期的 IN 転送または広帯域割り込み IN 転送中に OTG_HCINTx の XFRC 割り込みが受信されたとき。
3. OTG_GINTSTS レジスタの DISCINT（デバイス切断）割り込みを受信したとき。（アプリケーションは、すべての有効なチャネルを無効にすることが期待されます）。
4. アプリケーションが正常完了の前に転送を中断したとき。

ピンプロトコル

OTG_HS ホストが ハイスピードで動作する場合、ハイスピードバルクエンドポイントまたは制御（データおよびステータスステージ）OUT エンドポイントと交信中に、アプリケーションはピンプロトコルを開始する必要があります。アプリケーションは NAK/NYET/TXERR 割り込みを受信したとき、ピンプロトコルを開始する必要があります。HS_OTG ホストが上記応答の 1 つを受信した場合、特定のエンドポイントに対するトランザクションを継続しないで、（リクエストキューから）ポストまたはフェッチした全 OUT リクエストをドロップさせ、（送信 FIFO から）対応するデータを一掃させます。これはスレーブモードでのみ有効です。スレーブモードでは、アプリケーションは、チャネルを有効化する前に OTG_HCTSIZx 内の DOPING ビットをセットするか、またはチャネルが有効化済みの場合 OTG_HCTSIZx レジスタの DOPING ビットにセットを書き込むことにより、ピントークンを送信することができます。これにより、OTG_HS ホストがピンリクエストエントリをリクエストキューに書き込めるようになります。アプリケーションは、ピントークンに対する応答（NAK、ACK、または TXERR 割り込み）を待った後に、トランザクションを継続するか、または別のピントークンを送信する必要があります。アプリケーションは、リクエストされたピンの OUT エンドポイントからの ACK を受信した後にのみデータトランザクションを継続することができます。DMA モード動作では、アプリケーションは、バルク／制御 OUT の場合 NAK/NYET 応答に対して OTG_HCTSIZx 内の DOPING ビットをセットする必要はありません。OTG_HS ホストは OTG_HCTSIZx 内の DOPING ビットを自動的にセットし、バルク／制御 OUT に対してピントークンを発行します。OTG_HS ホストは、ACK を受信するまでピントークンの送信を続け、自動的にデータトランザクションに切り替わります。

動作モデル

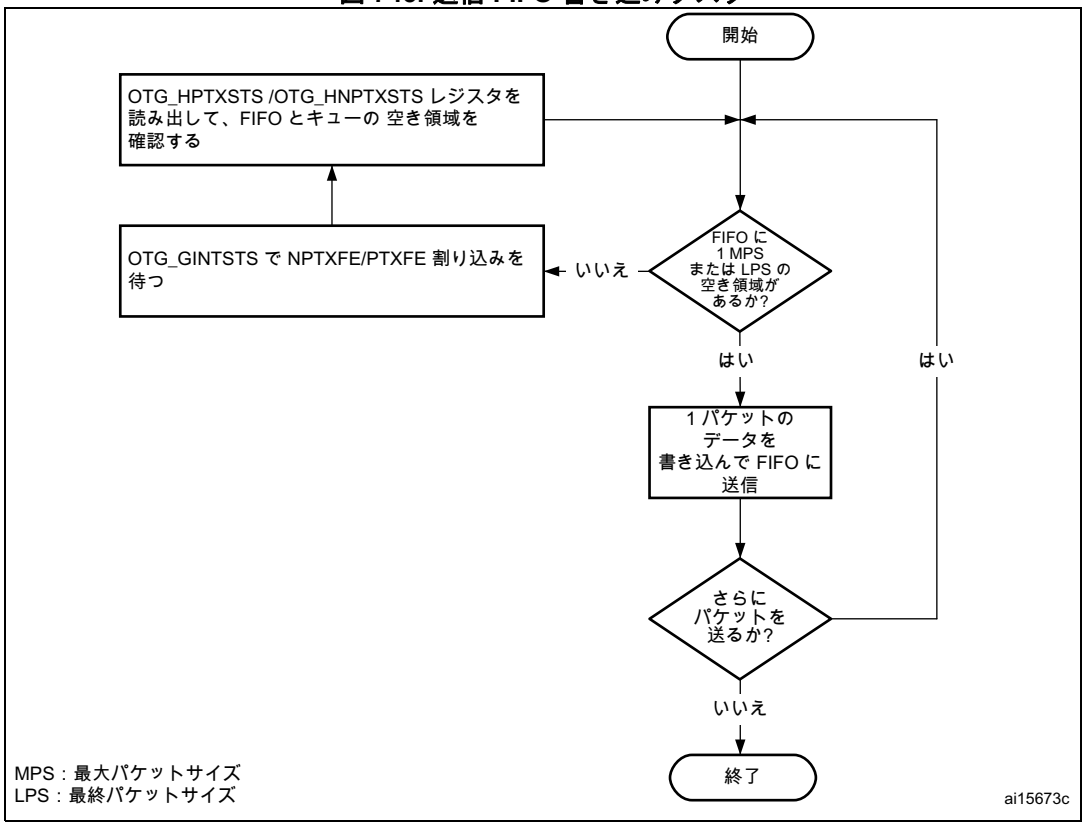
アプリケーションが接続されたデバイスと通信するためには、チャネルを初期化する必要があります。このセクションでは、さまざまなタイプの USB トランザクションについて、実行しなければならない動作のシーケンスについて説明します。

- **送信 FIFO への書き込み**

  OTG_HS ホストは、パケットの最後のダブルワードの書き込みとともに、周期的/非周期的リクエストキューにエントリ（OUT リクエスト）を自動的に書き込みます。アプリケーションは、

送信FIFO への書き込みを開始する前に、周期的/非周期的リクエストキューに少なくとも 1 つ以上の空きスペースがあることを確認しなければなりません。アプリケーションは送信 FIFO に必ずダブルワード単位で書き込みを行う必要があります。パケットサイズがDWORD 単位で揃えられていない場合、アプリケーションはパディングを使用する必要があります。OTG_HS ホストは、プログラムされた最大パケットサイズと転送サイズに基づいて、実際のパケットサイズを決定します。

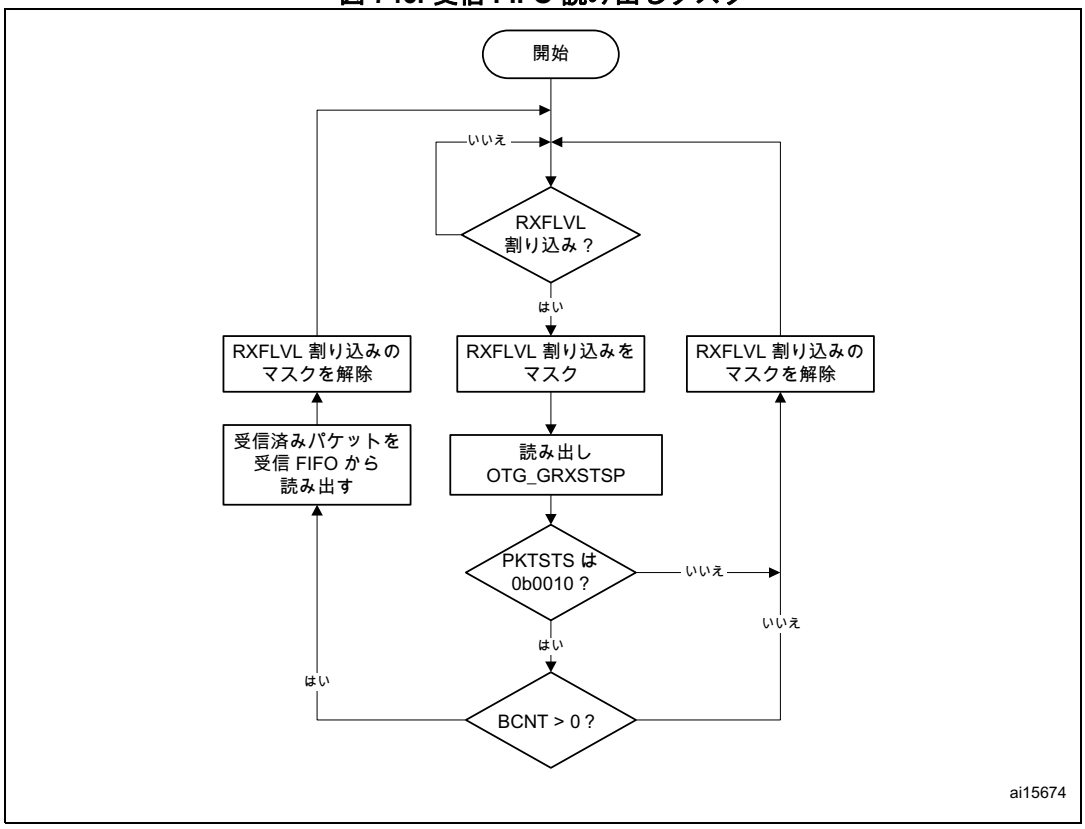
図 745. 送信 FIFO 書き込みタスク



- **受信 FIFO の読み出し**

  アプリケーションは、IN データパケット（bx0010）以外のすべてのパケットステータスを無視しなければなりません。

図 746. 受信 FIFO 読み出しタスク



- **バルクおよびコントロール OUT／SETUP トランザクション**

  典型的なバルクまたはコントロール OUT/SETUP パイプライントランザクションレベルの動作を *図 747*に示します。チャネル 1（ch_1）に注目してください。2 つのバルク OUT パケットが転送されます。コントロール SETUP トランザクションは同じように動作しますが、1 パケットのみです。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズの 2 つのパケットを送信しようとしています（転送サイズ = 1,024 バイト）。

  – 非周期的送信 FIFO は 2 つのパケットを保持することができます（HS の場合 1 KB）。

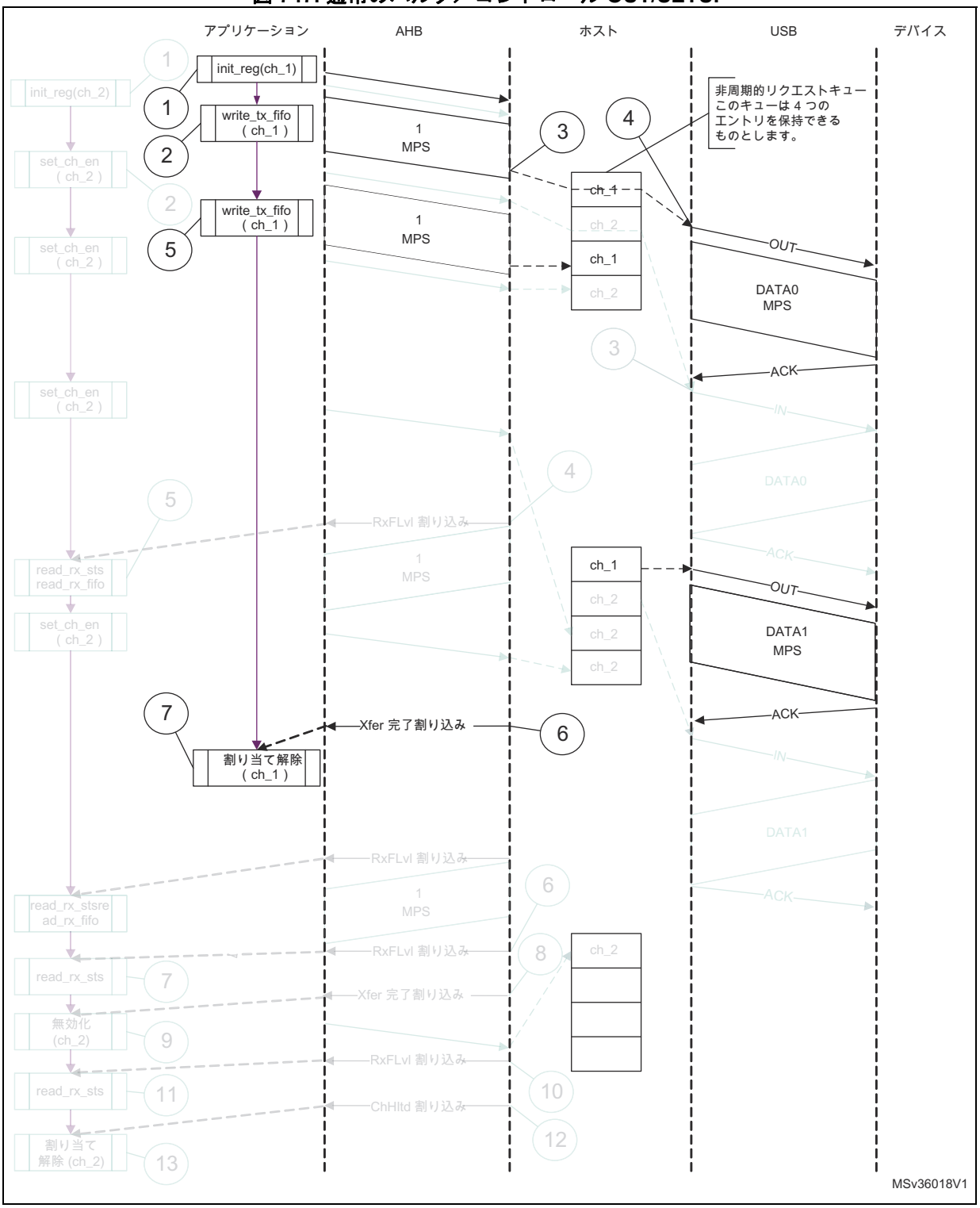
  – 非周期的リクエストキューの深さ = 4。

- **通常のバルクおよびコントロール OUT/SETUP 動作**

  動作シーケンス（チャネル 1）は、次のとおりです。

1. チャネル 1 を初期化します。
2. チャネル 1 の最初のパケットを書き込みます。
3. 最後のワードの書き込みと同時に、コアは、非周期的リクエストキューにエントリを書き込みます。
4. 非周期的キューがエンプティでなくなるとすぐに、コアは、現在のフレームで OUT トークンの送信を試みます。
5. チャネル 1 の 2 番目（最後）のパケットを書き込みます。
6. コアは、最後のトランザクションが正常に完了すると、XFRC 割り込みを生成します。
7. XFRC 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。
8. 非 ACK 応答の取り扱い

図 747. 通常のバルク／コントロール OUT/SETUP



1. 灰色の要素は、この図のコンテキストでは無関係です。

バルクおよびコントロール OUT/SETUP トランザクションのためのチャネル固有の割り込みサービスルーチンを、以下のコードサンプルで示します。

- **バルク／コントロール OUT/SETUP とバルクまたはコントロール IN トランザクションの割り込みサービスルーチン**

  a) バルク／コントロール OUT/SETUP

```
Unmask (NAK/TXERR/STALL/XFRC)
if (XFRC)
  {
  Reset Error Count
  Mask ACK
  De-allocate Channel
  }
else if (STALL)
  {
  Transfer Done = 1
  Unmask CHH
  Disable Channel
  }
else if (NAK or TXERR )
  {
  Rewind Buffer Pointers
  Unmask CHH
  Disable Channel
  if (TXERR)
    {
    Increment Error Count
    Unmask ACK
    }
  else
    {
    Reset Error Count
    }
    }
else if (CHH)
  {
  Mask CHH
  if (Transfer Done or (Error_count == 3))
    {
    De-allocate Channel
    }
  else
    {
```

```
      Re-initialize Channel
      }
   }
   else if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

アプリケーションは、送信 FIFO およびリクエストキューの中に使用可能なスペースがある場合、データパケットを送信 FIFO に書き込むことが期待されます。アプリケーションは、OTG_GINTSTS レジスタの NPTXFE 割り込みを使用して、送信 FIFO のスペースを見つけることができます。

b) バルク／コントロール IN

```
Unmask (TXERR/XFRC/BBERR/STALL/DTERR)
if (XFRC)
   {
   Reset Error Count
   Unmask CHH
   Disable Channel
   Reset Error Count
   Mask ACK
   }
else if (TXERR or BBERR or STALL)
   {
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
```

```
      }
   }
else if (ACK)
   {
   Reset Error Count
   Mask ACK
   }
else if (DTERR)
   {
   Reset Error Count
   }
```

アプリケーションは、リクエストキューのスペースが使用可能なときには、XFRC 割り込みが受信されるまで、リクエストを書き込むことが期待されます。

- **バルクおよびコントロール IN トランザクション**

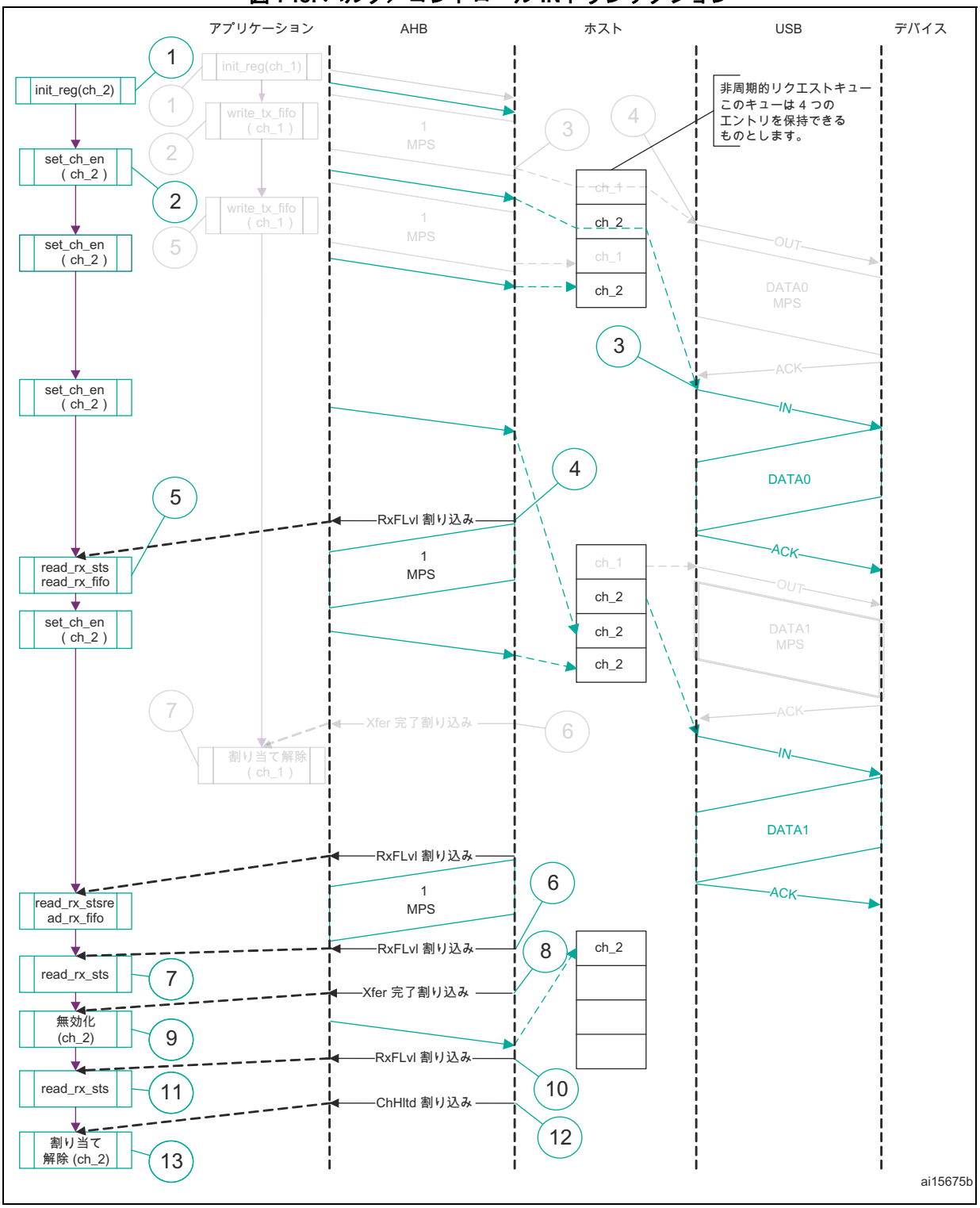
  典型的なバルクまたはコントロール IN パイプライントランザクションレベルの動作を *図 748* に示します。チャネル 2（ch_2）に注目してください。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズ（転送サイズ = 1,024 バイト）の 2 つのパケットを受信しようとしています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットと、パケットあたり 2 つのステータスワード（HS の場合 520 バイト）を収容することができます。

  – 非周期的リクエストキューの深さ = 4。

図 748. バルク／コントロール INトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。
2. OTG_HCCHAR2 の CHENA ビットをセットして、IN リクエストを非周期的リクエストキューに書き込みます。
3. コアは、現在の OUT トランザクションの完了後、IN トークンの送信を試みます。
4. コアは、受信したパケットが受信 FIFO に書き込まれると、RXFLVL 割り込みを生成します。
5. RXFLVL 割り込みに応答して、RXFLVL 割り込みをマスクし、受信パケットステータスを読み出して受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。続いて、RXFLVL 割り込みのマスクを解除します。
6. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。
7. アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（OTG_GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると、XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して、チャネルを無効にし、それ以降のリクエストについて、OTG_HCCHAR2 レジスタへの書き込みを停止します。コアは、OTG_HCCHAR2 に書き込まれると、非周期的リクエストキューにチャネルディセーブルリクエストを書き込みます。
10. コアは、停止ステータスが受信 FIFO に書き込まれると、RXFLVL 割り込みを生成します。
11. 受信パケットステータスを読み出して、それを無視します。
12. コアは、停止ステータスが受信 FIFO からポップされるとすぐに、CHH 割り込みを生成します。
13. CHH 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。
14. 非 ACK 応答の取り扱い

- **コントロールトランザクション**

  コントロール転送のセットアップ、データ、およびステータスステージは 3 つの個別の転送として実行されなければなりません。セットアップ、データ、およびステータスステージの OUT トランザクションは、すでに説明したバルク OUT トランザクションと同様に実行されます。データまたはステータスステージの IN トランザクションは、すでに説明したバルク IN トランザクションと同様に実行されます。3 つのステージすべてについて、アプリケーションは OTG_HCCHAR1 レジスタの EPTYP フィールドを Control に設定することが期待されます。セットアップステージでは、アプリケーションは OTG_HCTSIZ1 レジスタの PID フィールドを SETUP に設定することが期待されます。

- **割り込み OUT トランザクション**

  典型的な割り込み OUT 動作を *図 749*に示します。ここでは、以下のように仮定されています。

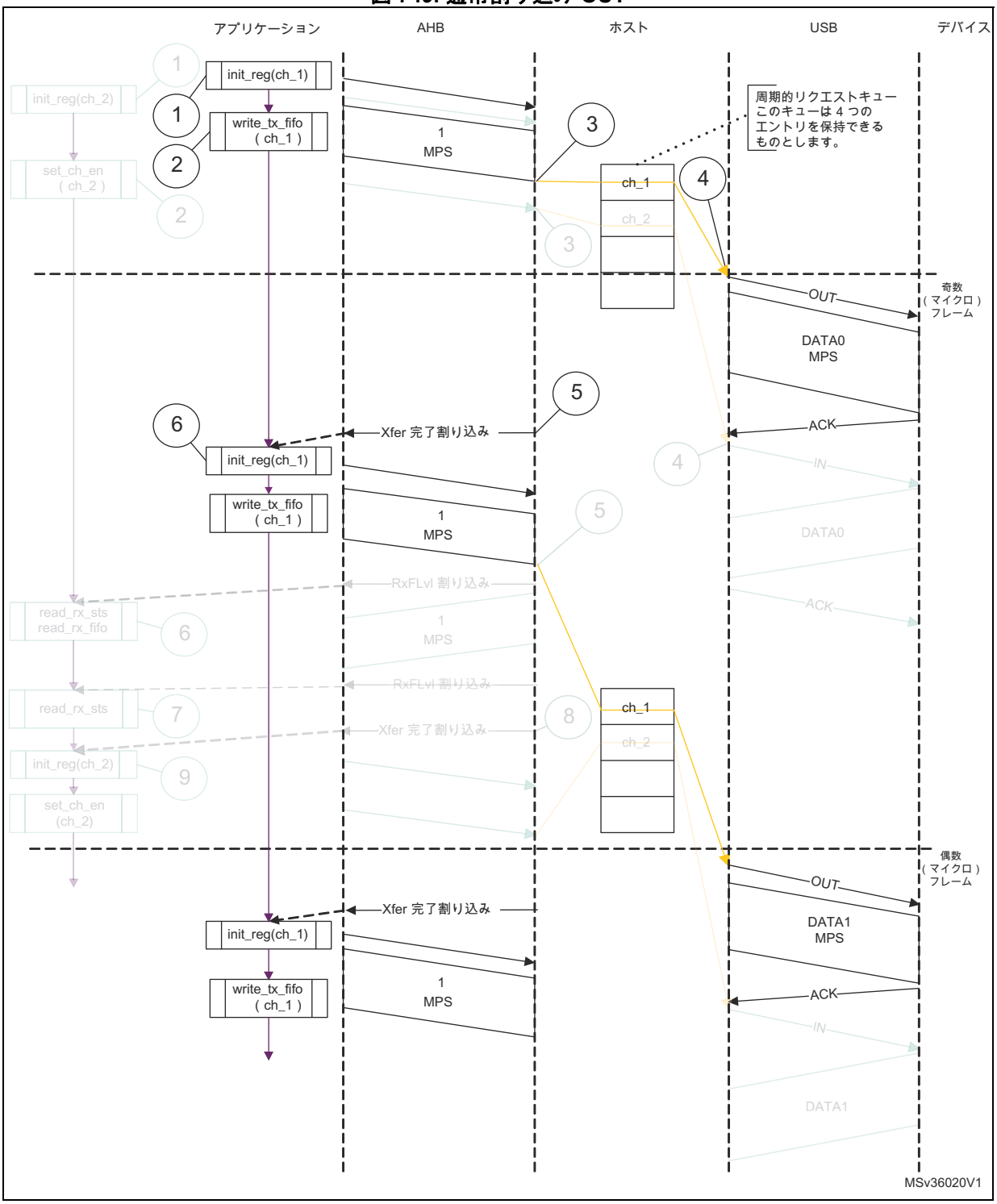
  – アプリケーションは、奇数フレーム（転送サイズ = 1 024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。
  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書き込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次の（奇数）フレームで OUT トークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC 割り込みを生成します。
6. XFRC 割り込みに応答して、次の転送のためにチャネルを再初期化します。

参考資料

図 749. 通常割り込み OUT



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **割り込み OUT/IN トランザクションのための割り込みサービスルーチン**

  a) 割り込み OUT

```
Unmask (NAK/TXERR/STALL/XFRC/FRMOR)
if (XFRC)
  {
  Reset Error Count
  Mask ACK
  De-allocate Channel
  }
else
  if (STALL or FRMOR)
    {
    Mask ACK
    Unmask CHH
    Disable Channel
    if (STALL)
      {
      Transfer Done = 1
      }
    }
  else
    if (NAK or TXERR)
      {
      Rewind Buffer Pointers
      Reset Error Count
      Mask ACK
      Unmask CHH
      Disable Channel
      }
    else
      if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
        {
        De-allocate Channel
        }
      else
        {
        Re-initialize Channel (in next b_interval - 1 Frame)
        }
      }
```

```
      else
        if (ACK)
        {
        Reset Error Count
        Mask ACK
        }
```

アプリケーションは OTG_GINTSTS レジスタの NPTXFE 割り込みを使用して、送信 FIFO のスペースを見つけます。

割り込み IN

```
Unmask (NAK/TXERR/XFRC/BBERR/STALL/FRMOR/DTERR)
if (XFRC)
  {
  Reset Error Count
  Mask ACK
  if (OTG_HCTSIZx.PKTCNT == 0)
    {
    De-allocate Channel
    }
  else
    {
    Transfer Done = 1
    Unmask CHH
    Disable Channel
    }
  }
else
  if (STALL or FRMOR or NAK or DTERR or BBERR)
    {
    Mask ACK
    Unmask CHH
    Disable Channel
    if (STALL or BBERR)
      {
      Reset Error Count
      Transfer Done = 1
      }
    else
      if (!FRMOR)
      {
      Reset Error Count
      }
    }
```

```
else
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      Unmask CHH
      Disable Channel
      }
else
   if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
         {
         De-allocate Channel
         }
      else
         Re-initialize Channel (in next b_interval - 1 /Frame)
      }
   }
else
   if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

- **割り込み IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

- **通常の割り込み IN 動作**

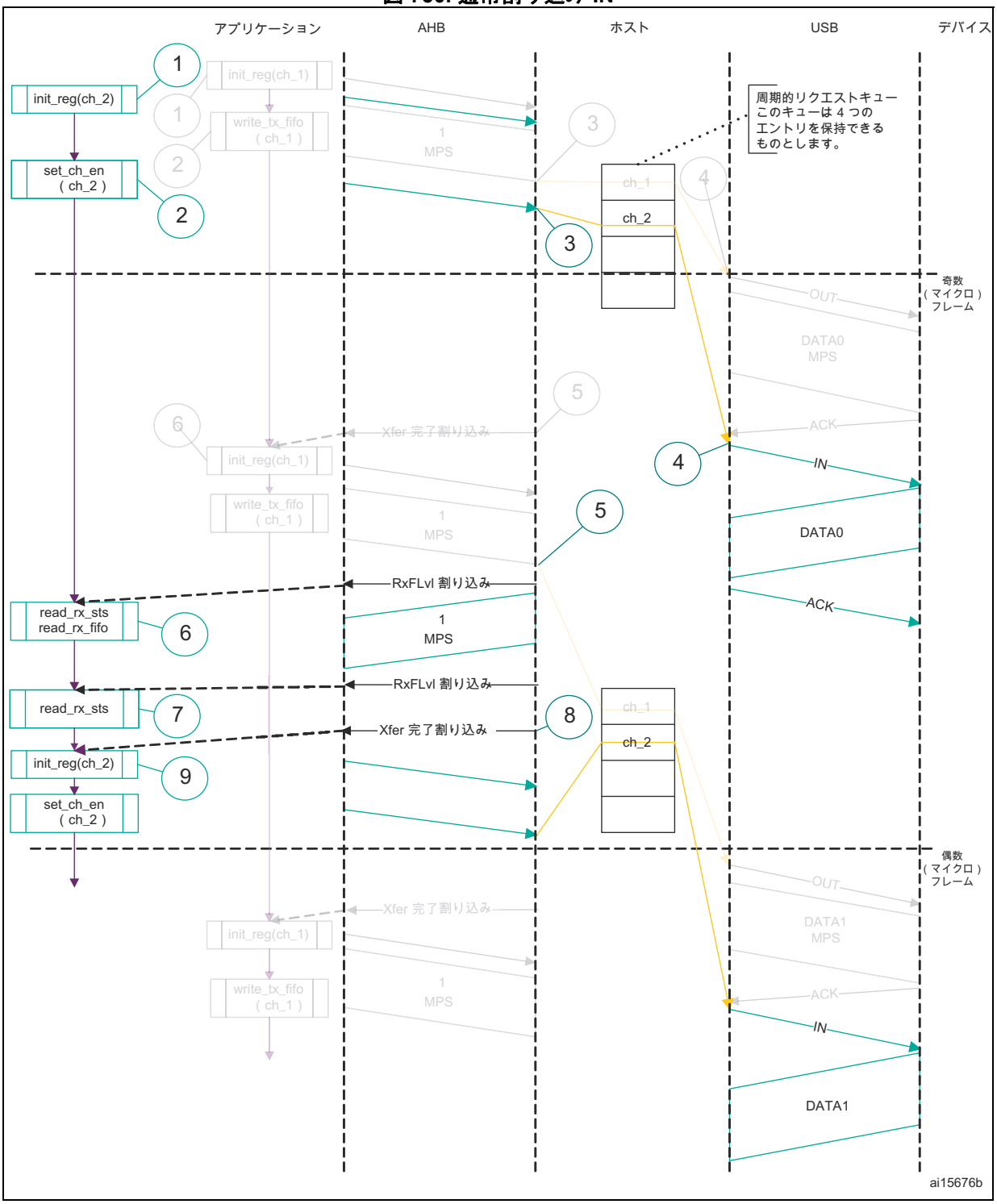
  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENA ビットがセットされた OTG_HCCHAR2 レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_HS ホストは、次の（奇数）フレームで IN トークンの送信を試みます。

5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは RXFLVL 割り込みを生成します。
6. RXFLVL 割り込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFLVL 割り込みをマスクし、すべてのパケットを読み出した後にマスクを解除しなければなりません。
7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。OTG_HCTSIZ2 の PKTCNT ビットが 0 でない場合は、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT ビットが 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

参考資料

図 750. 通常割り込み IN



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT トランザクション**

  典型的なアイソクロナス OUT 動作を *図 750*に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレームから始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。（転送サイズ = 1,024 バイト）。

  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。

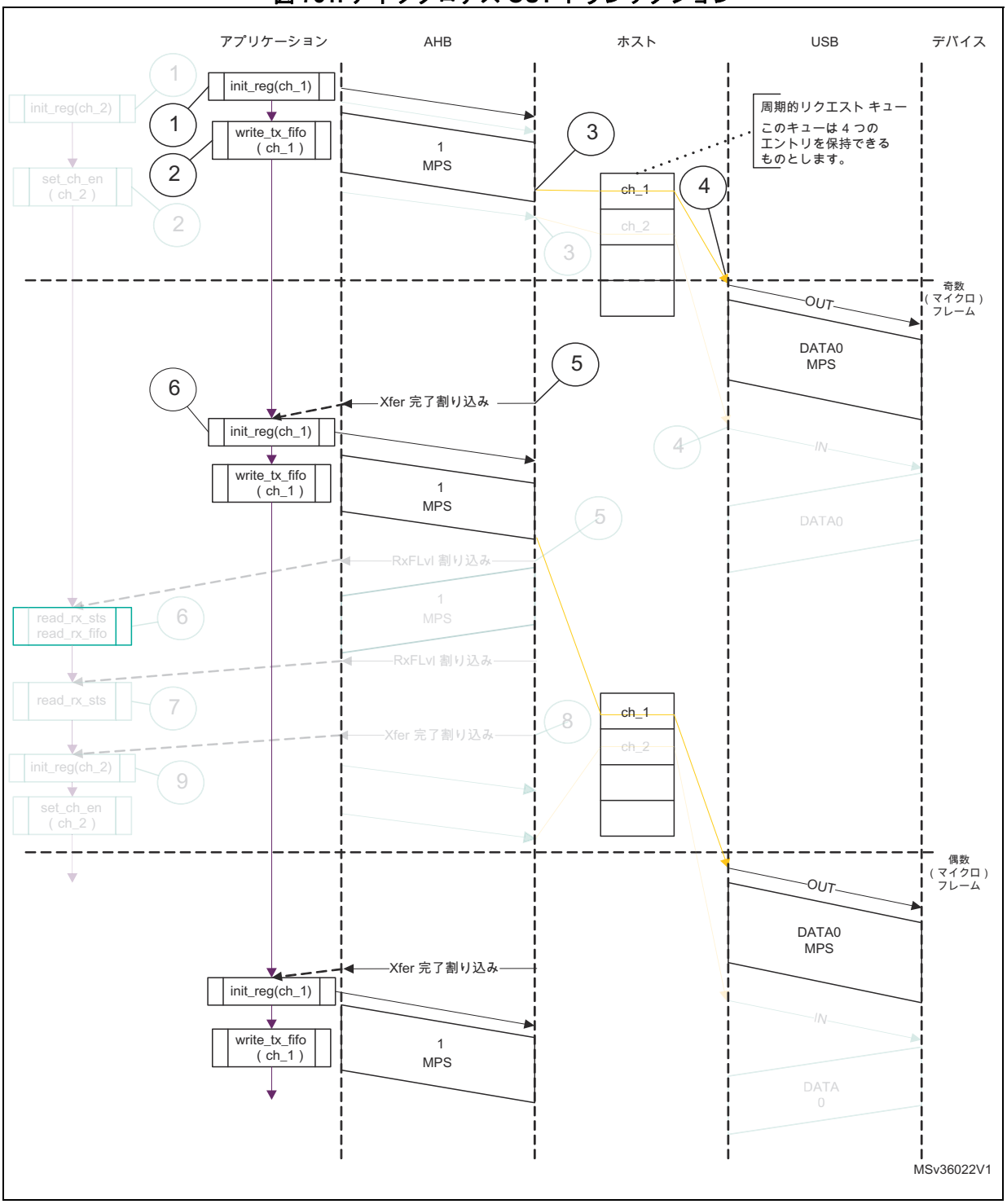
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書き込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次のフレーム（奇数）で OUT トークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC 割り込みを生成します。
6. XFRC 割り込みに応答して、次の転送のためにチャネルを再初期化します。
7. 非 ACK 応答の取り扱い

参考資料

図 751. アイソクロナス OUT トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT/IN トランザクションのための割り込みサービスルーチン**

  コードサンプル：アイソクロナス OUT

```
Unmask (FRMOR/XFRC)
if (XFRC)
  {
  De-allocate Channel
  }
else
  if (FRMOR)
    {
    Unmask CHH
    Disable Channel
    }
  else
  if (CHH)
    {
    Mask CHH
    De-allocate Channel
    }
```

コードサンプル：アイソクロナス IN

```
Unmask (TXERR/XFRC/FRMOR/BBERR)
if (XFRC or FRMOR)
  {
  if (XFRC and (OTG_HCTSIZx.PKTCNT == 0))
    {
    Reset Error Count
    De-allocate Channel
    }
  else
    {
    Unmask CHH
    Disable Channel
    }
  }
  else
    if (TXERR or BBERR)
      {
      Increment Error Count
      Unmask CHH
      Disable Channel
      }
    else
```

```
if (CHH)
  {
  Mask CHH
  if (Transfer Done or (Error_count == 3))
    {
    De-allocate Channel
    }
  else
    {
    Re-initialize Channel
    }
  }
```

- **アイソクロナス IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、次の奇数フレーム（転送サイズ = 1,024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

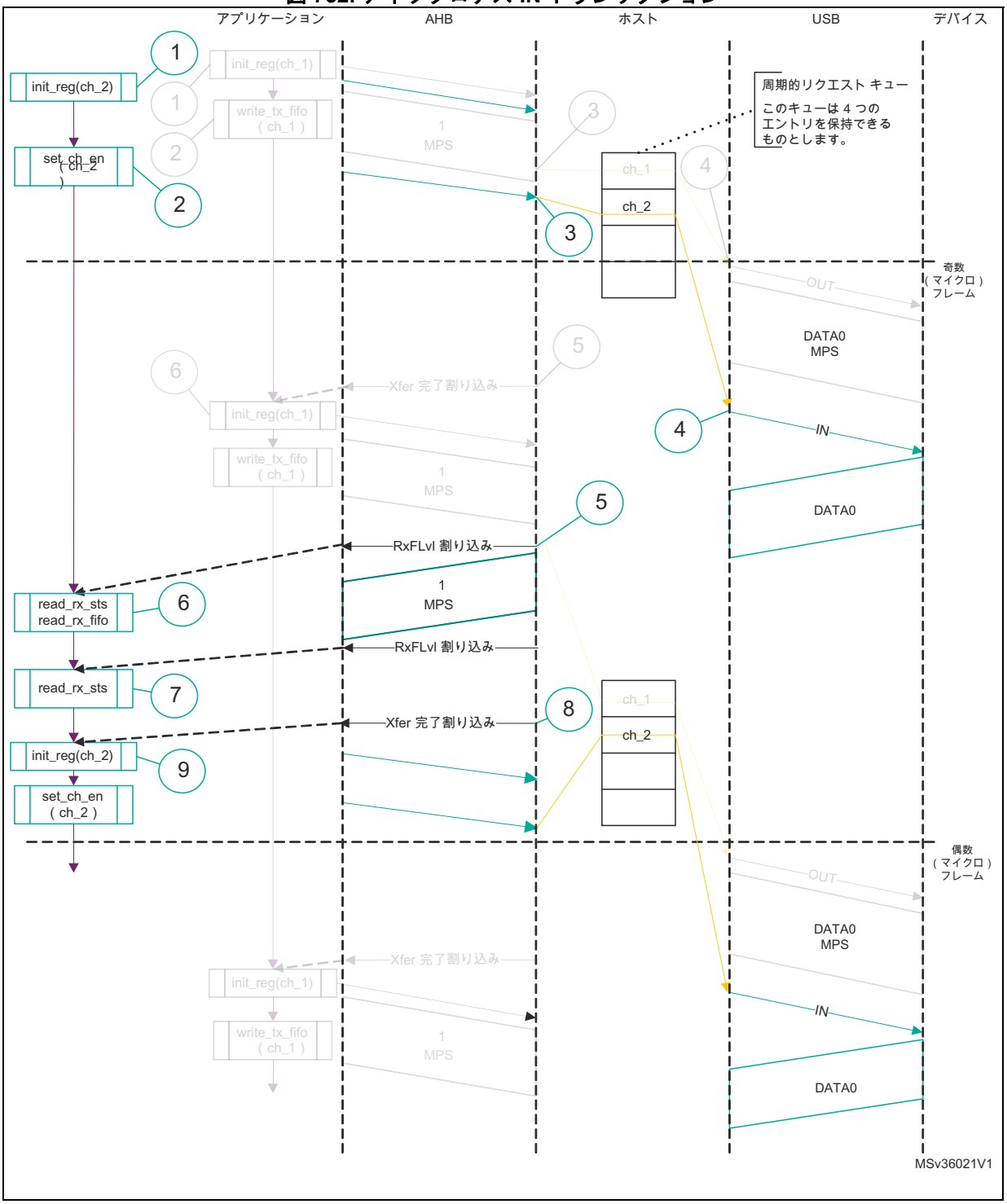
  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENA ビットがセットされた OTG_HCCHAR2 レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_HS ホストは、次の奇数フレームで IN トークンの送信を試みます。
5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは RXFLVL 割り込みを生成します。
6. RXFLVL 割り込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFVLV 割り込みをマスクし、すべてのパケットを読み出した後でマスクを解除しなければなりません。
7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。今度は、アプリケーションは受信パケットステータスを読み出して、それが IN データパケットでなかった場合（OTG_GRXSTSR ≠ 0b0010 の PKTSTS ビット）、無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。
   OTG_HCTSIZ2 の PKTCNT ≠ 0 の場合、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT = 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

図 752. アイソクロナス IN トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **キューの深さの選択**

  周期的および非周期的リクエストキューの深さは、アクセスされる周期的/非周期的エンドポイントの数に合うように注意して選択してください。

  非周期的リクエストキューの深さは、非周期的転送の性能に影響を与えます。キューが深いほど（FIFO のサイズが十分であれば）、コアは非周期的転送をより多くパイプライン化できます。キューのサイズが小さいと、コアはキューのスペースが空いたときしか新しいリクエストを入れることができません。

  コアの周期的リクエストキューの深さは、周期的転送をスケジュールどおりに実行するために不可欠です。周期的キューの深さは、マイクロフレーム内でスケジュールされた周期的転送の数に基づいて選択してください。周期的リクエストキューの深さがマイクロフレーム内でスケジュールされた周期的転送の数より小さい場合、フレームオーバーラン条件が発生します。

- **バブル条件の取り扱い**

  OTG_HS コントローラは、パケットバブルとポートバブルの 2 つのバブルを処理します。パケットバブルは、デバイスがチャネルの最大パケットサイズよりも多くのデータを送信した場合に発生します。ポートバブルは、コアが EOF2（フレーム 2 の終わりの SOF にきわめて近い）でデバイスからデータを受信し続けると発生します。

  OTG_HS コントローラがパケットバブルを検出すると、Rx バッファへのデータの書き込みを停止して、パケットの終了（EOP）を待ちます。EOP を検出すると コントローラは Rx バッファにすでに書き込まれたデータを一掃して、アプリケーションに対するバブル割り込みを生成します。

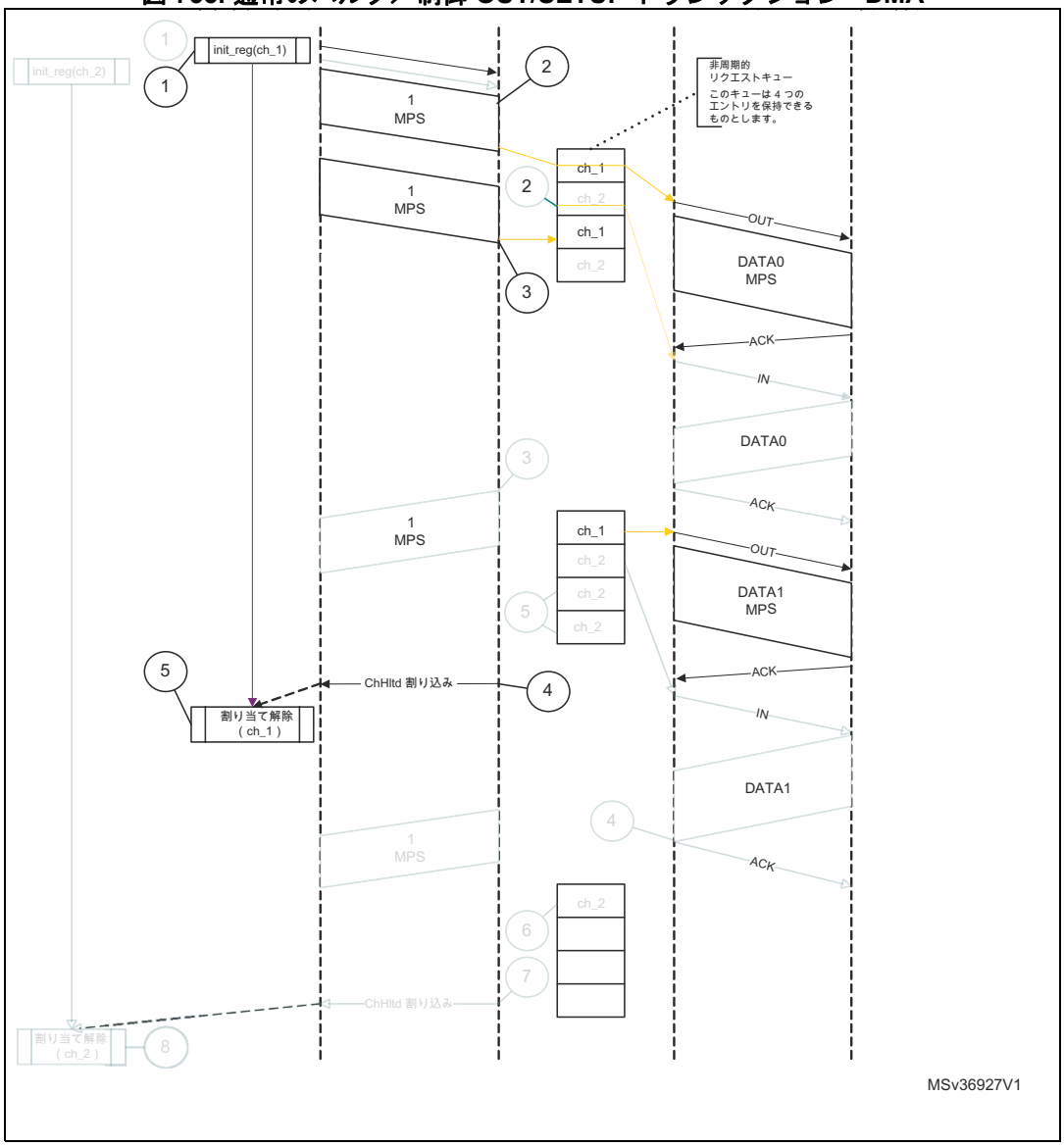
  OTG_HS コントローラがポートバブルを検出すると、Rx FIFO をフラッシュして、ポートを無効にします。コアは、ポートディセーブル割り込み（OTG_GINTSTS の HPRTINT、OTG_HPRT の PENCHNG）を生成します。この割り込みを受信すると、アプリケーションは OTG_HPRT の POCA ビットをチェックして、これが過電流条件（ポートディセーブル割り込みのもう 1 つの原因）によるものではないことを確認してから、ソフトリセットを行う必要があります。コアは、ポートバブル条件を検出した後は、それ以上トークンを送信しません。

- **バルクおよび制御 OUT／SETUP トランザクション - DMA モード**

  動作シーケンスは、次のとおりです。

1. *セクション：チャネルの初期化*で説明しているように、チャネル 1 を初期化し、有効にします。
2. HS_OTG　ホストはチャネルが有効化されると直ちに最初のパケットのフェッチを開始します。内部DMA モードの場合、OTG_HS ホストはパケットのフェッチにプログラムされ DMA アドレスを使います。
3. 2 つ目 （最後）のパケットの最後のダブルワードをフェッチした後、OTG_HS ホストは以降の調停に対し内部的にチャネル 1 をマスクします。
4. OTG_HS ホストは、最後のパケットが送信されると直ちに CHH 割り込みを生成します。
5. CHH 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。

図 753. 通常のバルク／制御 OUT/SETUP トランザクション - DMA



- **内部 DMA による NAK と NYET の処理**

1. OTG_HS ホストはバルク OUT トランザクションを送信します。
2. デバイスは NAK または NYET で応答します。
3. アプリケーションで NAK または NYET のマスクを解除した場合、コアはアプリケーションへ対応する割り込みを生成します。コアがバッファポインタの戻し処理を行っている場合およびアプリケーション介入なしでチャネルを再初期化している場合、アプリケーションはこれらの割り込みを生成する必要はありません。
4. コアは自動的にピントークンを発行します。
5. デバイスが ACK を返すと、コアは転送を続けます。オプションとして、アプリケーションはこれらの割り込みを使用することができます。この場合 NAK または NYET割り込みはアプリケーションからマスクされます。

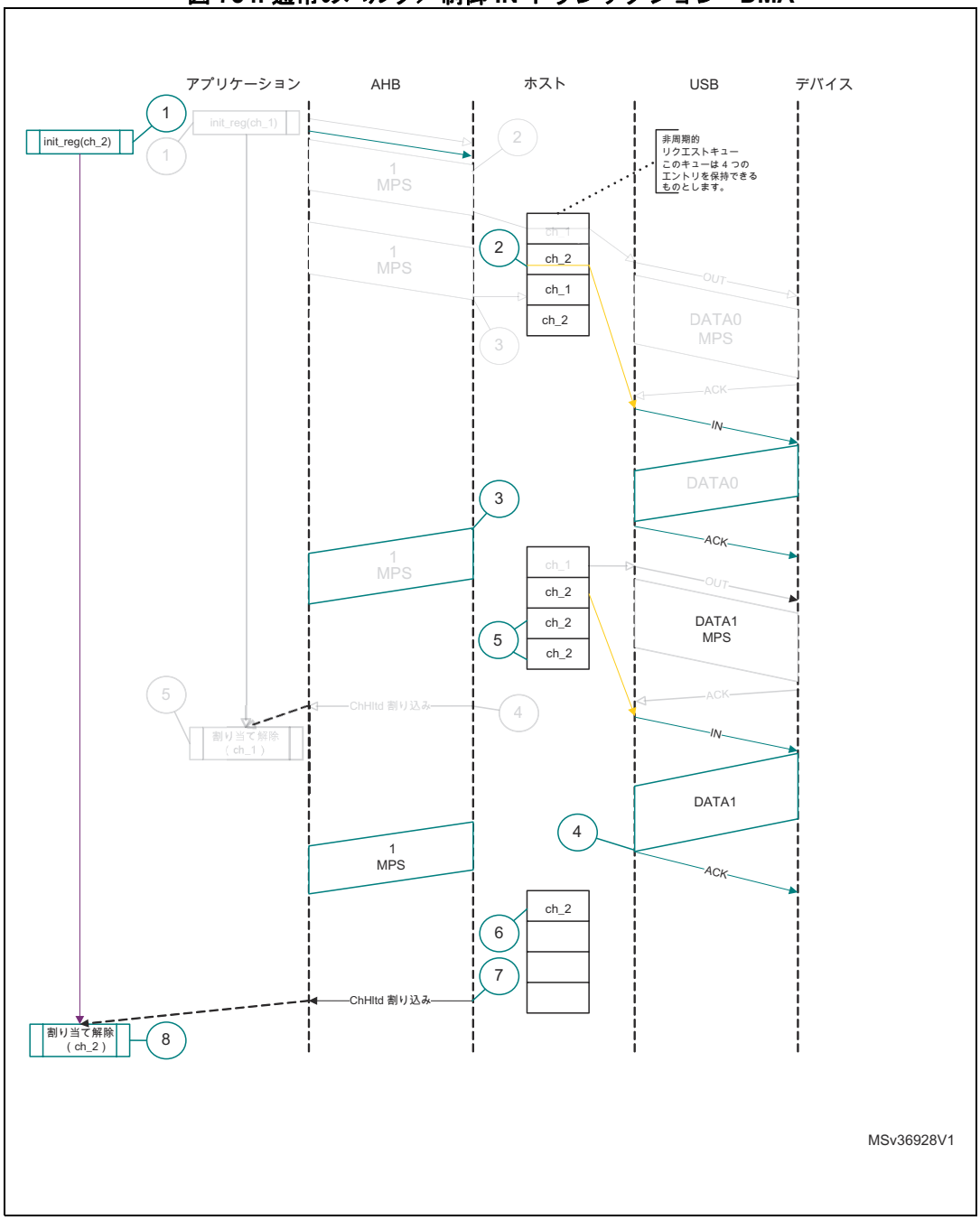
ホスト機能が NAK または NYET を受信した場合、コアは個別の割り込みを生成しません。

- **バルクおよび制御 IN トランザクション - DMA モード**

  動作シーケンスは、次のとおりです。

1. *セクション：チャネルの初期化*で説明しているように、使用するチャネル（チャネル x）を初期化および有効化します。
2. チャネルがアービタから許可を受信すると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます （調停はラウンドロビン方式で行われます）。
3. OTG_HS ホストはエラーなしで最終バイトを受信すると、直ちにシステムメモリへの受信データの書き込みを開始します。
4. OTG_HS ホストは最終パケットを受信すると、内部フラグをセットしてリクエストキューから余分な IN リクエストを削除します。
5. OTG_HS ホストは余分なリクエストを一掃します。
6. チャネル x を無効化する最終リクエストがリクエストキューに書き込まれます。この時点で、チャネル 2 は以降の調停に対して内部でマスクされます。
7. OTG_HS ホストは、ディセーブルリクエストがキューのトップに来ると直ちに CHH 割り込みを生成します。
8. CHH 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。

図 754. 通常のバルク／制御 IN トランザクション - DMA

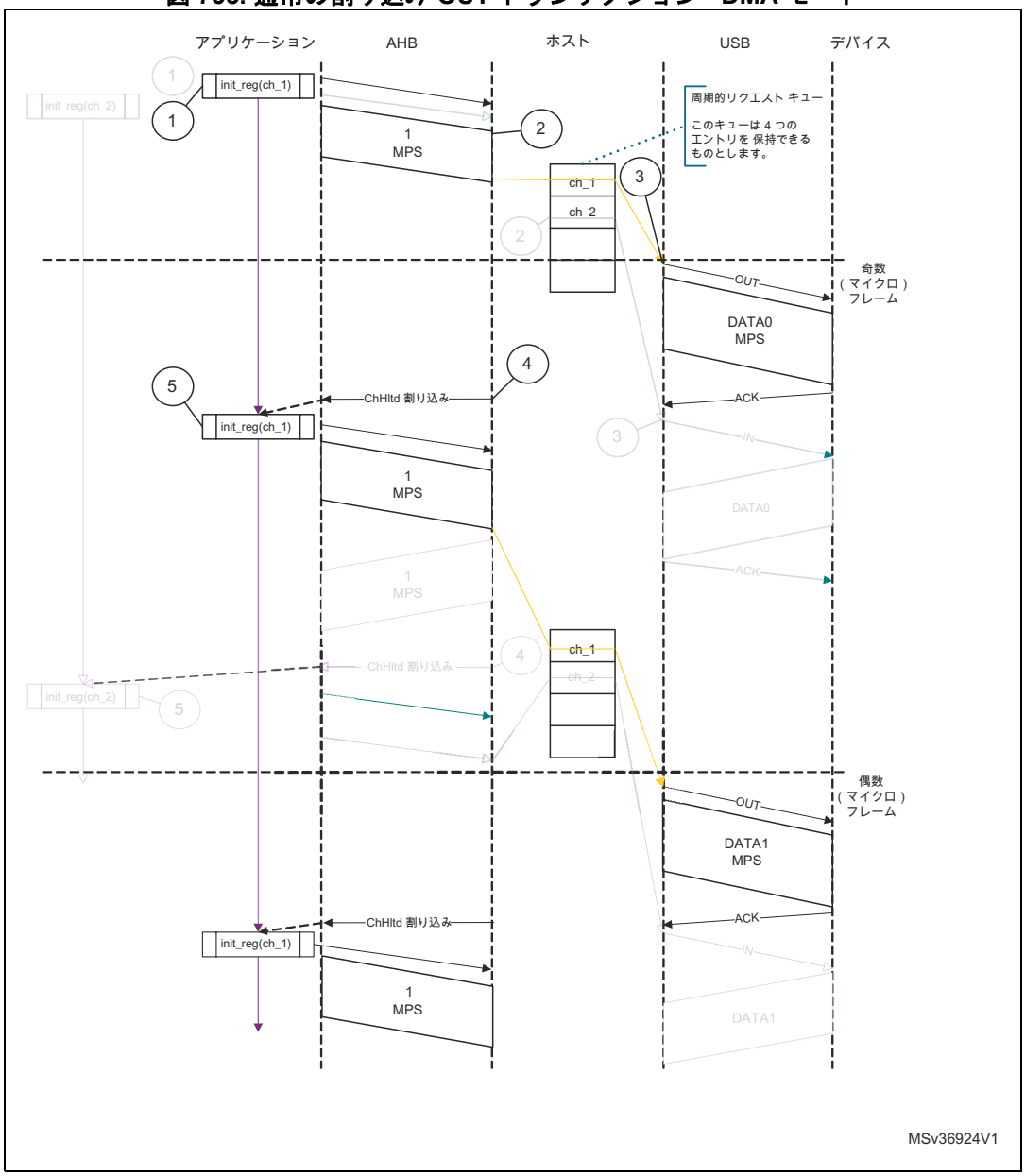


- **割り込み OUT トランザクション - DMA モード**

1. *セクション ： チャネルの初期化*で説明しているように、チャネル x を初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し OUT リクエストを最後のダブルワードフェッチと一緒に書き込みます。広帯域転送ではHS_OTG ホストは、次のパケットのフェッチを（MC フィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始で OUT トークンの送信を試みます。

4. パケットの正常送信の後、OTG_HS ホストは CHH 割り込みを生成します。
5. CHH 割り込みに応答して、次の転送のためにチャネルを再初期化します。

図 755. 通常の割り込み OUT トランザクション - DMA モード



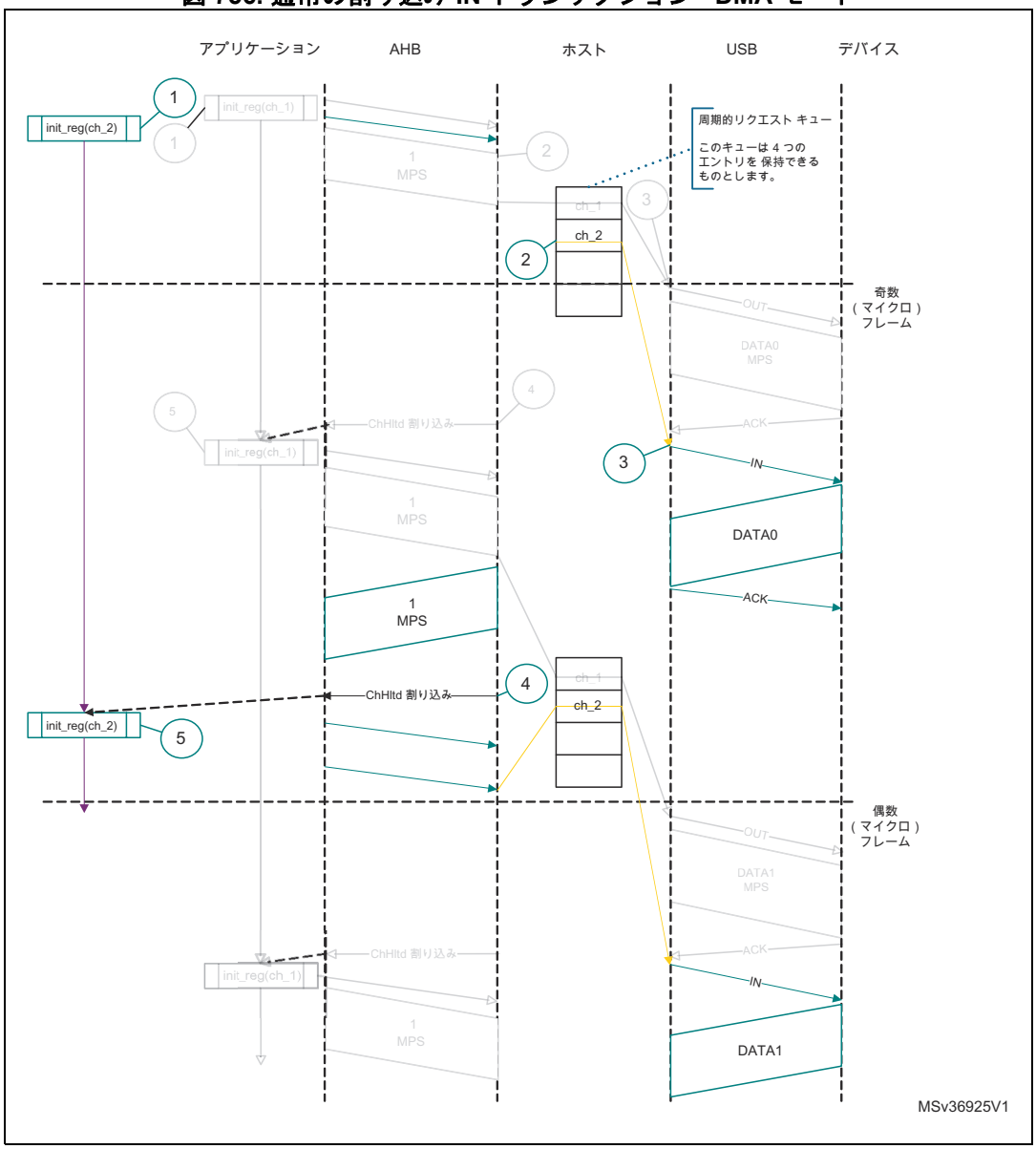
- **割り込み IN トランザクション - DMA モード**

  動作シーケンス （チャネル x）は次の通りです。

1. *セクション：チャネルの初期化*で説明しているように、チャネル x を初期化し、有効にします。
2. チャネル x がアービタから許可を受け取ると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます （ラウンドロビン方式、フェアネスを使用）。広帯域転送では OTG_HS ホストは MC 回までの連続書き込みを行います。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始で IN トークンの送信を試みます。

4. パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは CHH 割り込みを生成します。
5. CHH 割り込みに応答して、次の転送のためにチャネルを再初期化します。

図 756. 通常の割り込み IN トランザクション - DMA モード

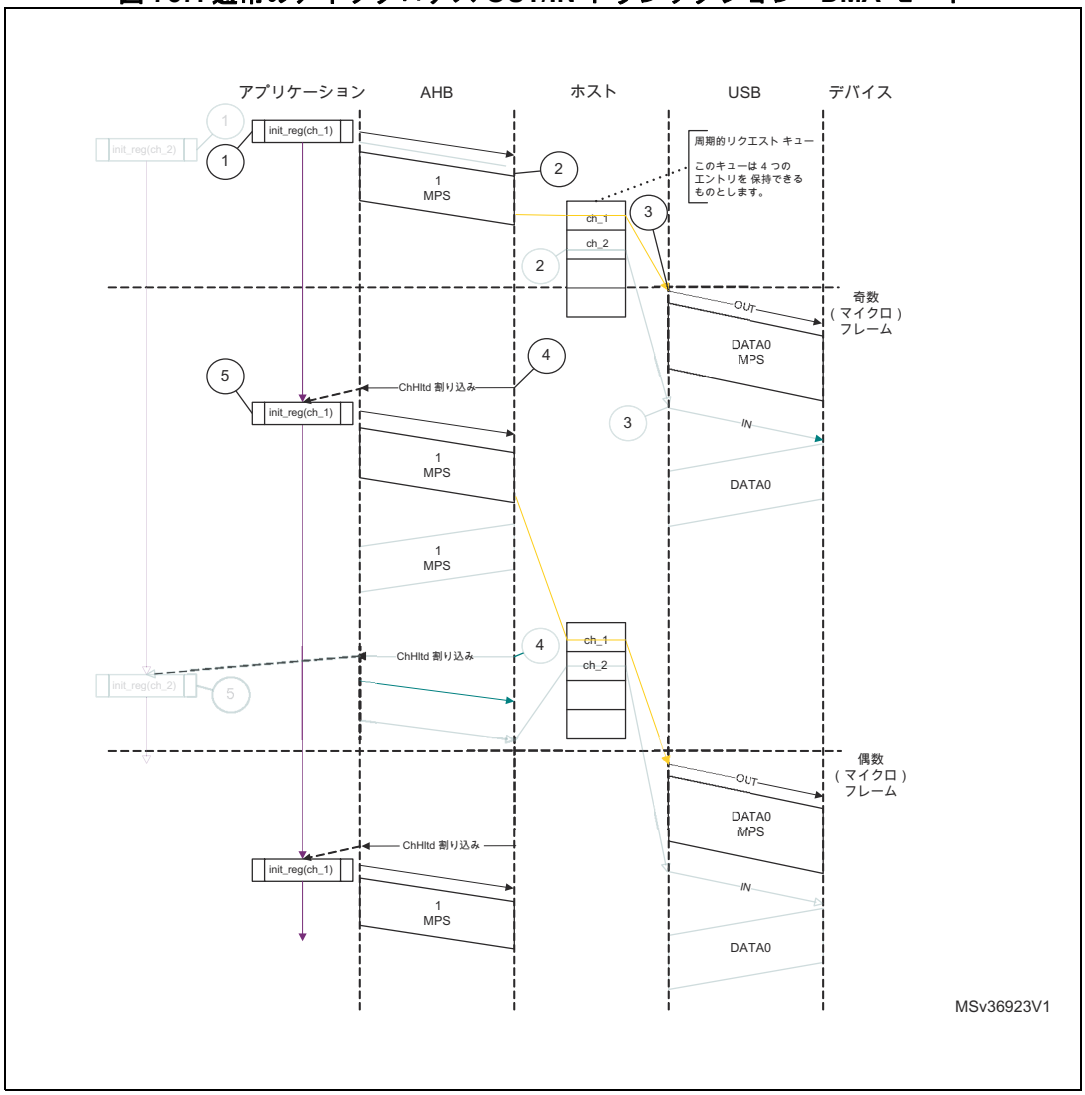


- **アイソクロナス OUT トランザクション - DMA モード**

1. *セクション：チャネルの初期化*で説明しているように、チャネル x を初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し、OUT リクエストを最後のダブルワードフェッチと一緒に書き込みます。広帯域転送では OTG_HS ホストは、次のパケットのフェッチを（MC フィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始で OUT トークンの送信を試みます。
4. パケットの正常送信の後、HS_OTG ホストは CHH 割り込みを生成します。

5. CHH 割り込みに応答して、次の転送のためにチャネルを再初期化します。

図 757. 通常のアイソクロナス OUT/IN トランザクション - DMA モード

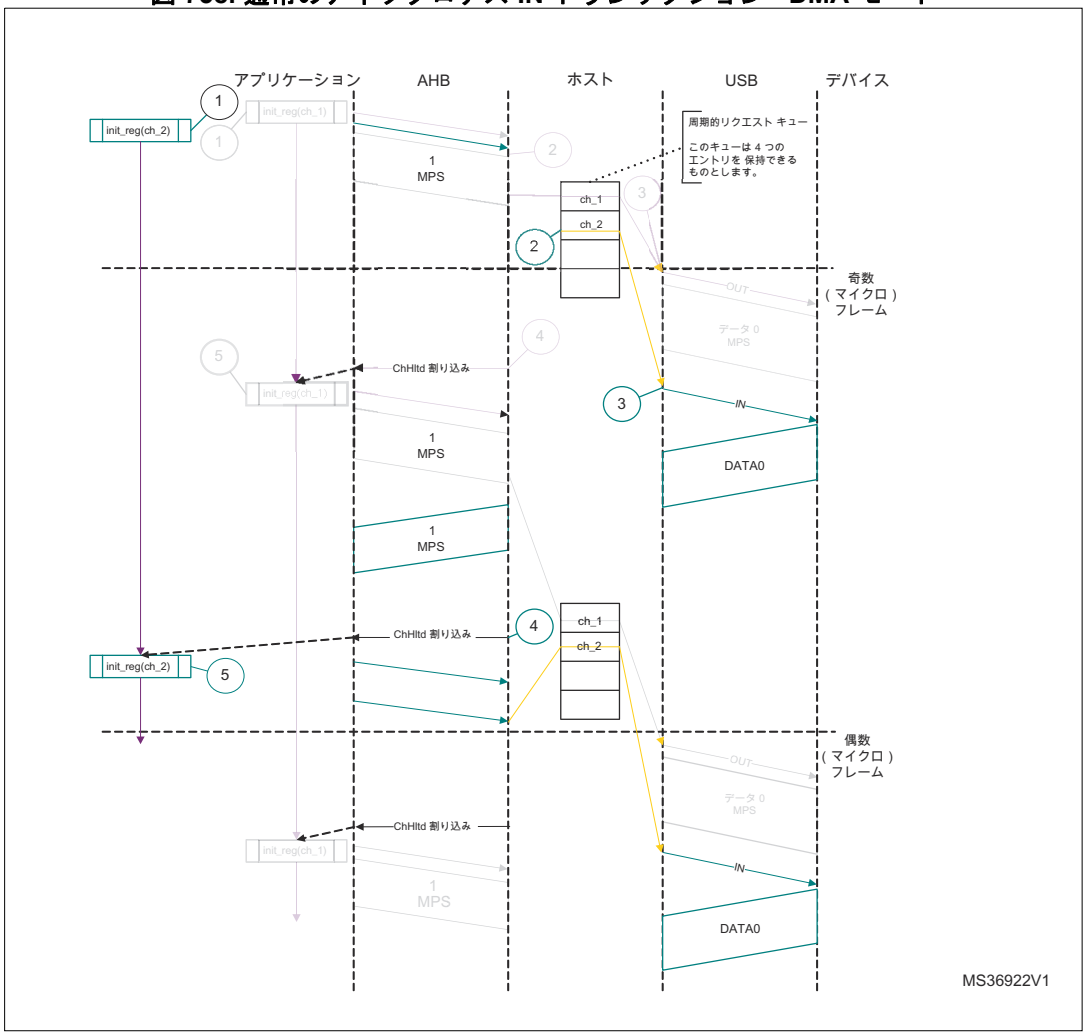


- **アイソクロナス IN トランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション：チャネルの初期化*で説明しているように、チャネル x を初期化し、有効にします。
2. チャネル x がアービタから許可を受け取ると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます （ラウンドロビン方式、フェアネスを使用）。広帯域転送では OTG_HS ホストは MC 回までの連続書き込みを行います。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始で IN トークンの送信を試みます。
4. パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは CHH 割り込みを生成します。
5. CHH 割り込みに応答して、次の転送のためにチャネルを再初期化します。

図 758. 通常のアイソクロナス IN トランザクション - DMA モード



- **バルクおよび制御 OUT／SETUP スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション ： チャネルの初期化*の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し OUT リクエストを最後のダブルワードフェッチと一緒に書き込みます。
3. スタートスプリットの正常送信の後 OTG_HS ホストは CHH 割り込みを生成します。
4. CHH 割り込みに応答して OTG_HCSPLT1 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
5. スプリット全体の正常送信の後 OTG_HS ホストは CHH 割り込みを生成します。
6. CHH 割り込みに応答して、チャネルの割り当てを解除します。

- **バルク／制御 IN スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション ： チャネルの初期化*で説明しているように、チャネル x を初期化し、有効にします。

2. アービタから許可を得た後、OTG_HS ホストはスタートスプリットリクエストを非周期的リクエストへ書き込みます。OTG_HS ホストは、リクエストを書き込んだ後に、調停のために内部でチャネル x をマスクします。
3. IN トークンが送信されると直ちに OTG_HS ホストは CHH 割り込みを生成します。
4. CHH 割り込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリットトークン全体を送信するためにチャネルを再度有効化します。これにより、調停用にチャネル x のマスクが解除されます。
5. アービタから許可を得た後、OTG_HS ホストはスプリット全体リクエストを非周期的リクエストへ書き込みます。
6. OTG_HS ホストはパケットを正常に受信した後、パケットのシステムメモリへの書き込みを開始します。
7. 受信パケットをシステムメモリへ書き込むと直ちに、OTG_HS ホストは CHH 割り込みを生成します。
8. CHH 割り込みに応答して、チャネルの割り当てを解除します。

- **割り込み OUT スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション：チャネルの初期化*の説明に従い、スタートスプリット用にチャネル 1 を初期化および有効化します。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HS ホストはパケットの読み出しを開始します。
3. OTG_HS ホストは、スタートスプリットトランザクションの送信を試みます。
4. スタートスプリットの正常送信の後 OTG_HS ホストは CHH 割り込みを生成します。
5. CHH 割り込みに応答して OTG_HCSPLT1 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. スプリット全体トランザクションの正常送信の後、OTG_HSホストは CHH 割り込みを生成します。
7. CHH 割り込みに応答して、チャネルの割り当てを解除します。

- **割り込み IN スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション：チャネルの初期化*の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. チャネル x がアービタから許可を受け取ると直ちに OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリット IN トークンの送信を試みます。
4. OTG_HS ホストはスタートスプリット IN トークンを正常に送信した後に CHH 割り込みを生成します。
5. CHH 割り込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書き込みを開始します。
7. OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH 割り込みを生成します。
8. CHH 割り込みに応答して、次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

- **アイソクロナス OUT スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション：チャネルの初期化*の説明に従い、スタートスプリット（開始）用にチャネル x を初期化および有効化します。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。MPS フィールドをプログラムします。
2. OTG_HS ホストはパケットの読み出しを開始します。
3. スタートスプリット（開始）の正常送信の後、OTG_HS ホストは CHH 割り込みを生成します。
4. CHH 割り込みに応答して、スタートスプリット（終わり）を送信するためレジスタを再初期化します。
5. スタートスプリット（終わり）の正常送信の後、OTG_HS ホストは CHH 割り込みを生成します。
6. CHH 割り込みに応答して、チャネルの割り当てを解除します。

- **アイソクロナス IN スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. *セクション：チャネルの初期化*の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. チャネル x がアービタから許可を受け取ると直ちに OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリット IN トークンの送信を試みます。
4. OTG_HS ホストはスタートスプリット IN トークンを正常に送信した後に CHH 割り込みを生成します。
5. CHH 割り込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書き込みを開始します。

   OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH 割り込みを生成します。CHH 割り込みに応答して、次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

注： ***このセクションの内容は、USB OTG HS にのみ適用されます。***

57.15.6 デバイスプログラミングモデル

USB リセット時のエンドポイントの初期化

1. すべての OUT エンドポイントの NAK ビットをセットします。
   – OTG_DOEPCTLx の SNAK = 1（すべての OUT エンドポイントについて）
2. 以下の割り込みビットのマスクを解除します。
   – OTG_DAINTMSK の INEP0 = 1（制御 0 IN エンドポイント）
   – OTG_DAINTMSK の OUTEP0 = 1（制御 0 OUT エンドポイント）
   – OTG_DOEPMSK の STUPM = 1
   – OTG_DOEPMSK の XFRCM = 1
   – OTG_DIEPMSK の XFRCM = 1
   – OTG_DIEPMSK の TOM = 1
3. 各 FIFO の DATA FIFO RAM をセットアップします。
   – OTG_GRXFSIZ レジスタをプログラムして、コントロール OUT データとセットアップデータを受信できるようにします。.閾値の設定が無効の場合、これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分 + 2 ワード（コントロール OUT データパケットのステータス用）+ 10 ワード（セットアップパケット用）でなければなりません。
   – OTG_DIEPTXF0 レジスタを、コントロール IN データを送信できるようにプログラムします（選択された FIFO 番号に応じて）。これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分以上でなければなりません。
4. SETUP パケットを受信するために、コントロール OUT エンドポイント 0 のエンドポイント固有レジスタの以下のフィールドをプログラムします。
   – OTG_DOEPTSIZ0 の STUPCNT = 3（最大 3 つの連続 SETUP パケットを受信するため）
5. DMA モードの USB_OTG_HS では、OTG_DOEPDMA0 レジスタは受信した全 SETUP パケットを保存する有効なメモリアドレスを持つ必要があります。

この時点で SETUP パケットを受信するために必要なすべての初期化が終了したことになります。

エニュメレーション完了時のエンドポイント初期化

1. エニュメレーション終了割り込み（OTG_GINTSTS の ENUMDNE ビット）時には OTG_DSTS レジスタを読み出して、エニュメレーションスピードを決めます。
2. OTG_DIEPCTL0 の MPSIZ フィールドをプログラムして、最大パケットサイズを設定します。このステップでは、コントロールエンドポイント 0 を設定します。コントロールエンドポイントの最大パケットサイズは、エニュメレーションスピードに依存します。
3. DMA モードの USB OTG HS では、制御 OUT エンドポイント 0 を有効化する OTG_DOEPCTL0 レジスタをプログラムして、SETUPパケットを受信するようにします。

この時点で、デバイスは SOF パケットを受信する準備ができ、コントロールエンドポイント 0 で制御転送を行うように設定されたことになります。

SetAddress コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetAddress コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. OTG_DCFG レジスタに SetAddressコマンドで受信したデバイスアドレスをプログラムします。
2. ステータス IN パケットを送信するように、コアをプログラムします。

SetConfiguration/SetInterface コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetConfiguration または SetInterface コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. SetConfiguration コマンドを受信したとき、アプリケーションは、新しい設定で有効なエンドポイントの特性で、エンドポイントのレジスタをプログラムする必要があります。
2. SetInterface コマンドを受信したとき、アプリケーションは、このコマンドの影響を受けるエンドポイントのレジスタをプログラムする必要があります。
3. 前の設定または代替設定ではアクティブであったエンドポイントが、新しい設定または代替設定では無効なことがあります。これらの無効なエンドポイントは、機能を停止させる必要があります。
4. アクティブな各エンドポイントの割り込みのマスクを解除し、OTG_DAINTMSK レジスタですべての非アクティブなエンドポイントに対する割り込みをマスクします。
5. 各 FIFO のデータ FIFO RAM をセットアップします。
6. 必要なすべてのエンドポイントを設定した後、アプリケーションは、ステータス IN パケットを送信するようにコアをプログラムする必要があります。

この時点で、デバイスのコアは、任意のタイプのデータパケットを送信／受信できるように設定されたことになります。

エンドポイントのアクティブ化

このセクションでは、デバイスエンドポイントをアクティブ化するか、既存のデバイスエンドポイントを新しいタイプに設定するために必要な手順について説明します。

1. 必要なエンドポイントの特性を OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の以下のフィールドにプログラムします。
   - 最大パケットサイズ
   - USB アクティブエンドポイント = 1
   - エンドポイント開始データトグル（割り込みおよびバルクエンドポイントの場合）
   - エンドポイントタイプ
   - Tx FIFO 番号
2. エンドポイントがアクティブ化されると、コアは、そのエンドポイントに宛てたトークンのデコードを開始し、そのエンドポイントで受信された有効な各トークンについて有効なハンドシェイクを送出します。

エンドポイントの機能停止

このセクションでは、既存のエンドポイントの機能を停止させるために必要な手順について説明します。

1. 非アクティブにすべきエンドポイントでは、OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の USB のアクティブエンドポイントビットをクリアします。
2. エンドポイントが非アクティブな状態になると、コアは、そのエンドポイントにアドレス指定されたトークンを無視するので USB 上でタイムアウトが発生します。

注： ***アプリケーションは、以下の条件を満たすように、デバイスコアがトラフィックを処理できるようにセットアップする必要があります。***
OTG_GINTMSK レジスタの NPTXFEM および RXFLVLM をクリアする必要があります。

動作モデル

SETUP および OUT データ転送

このセクションでは、データ OUT 転送および SETUP トランザクション時の内部データフローとアプリケーションレベルの動作について説明します。

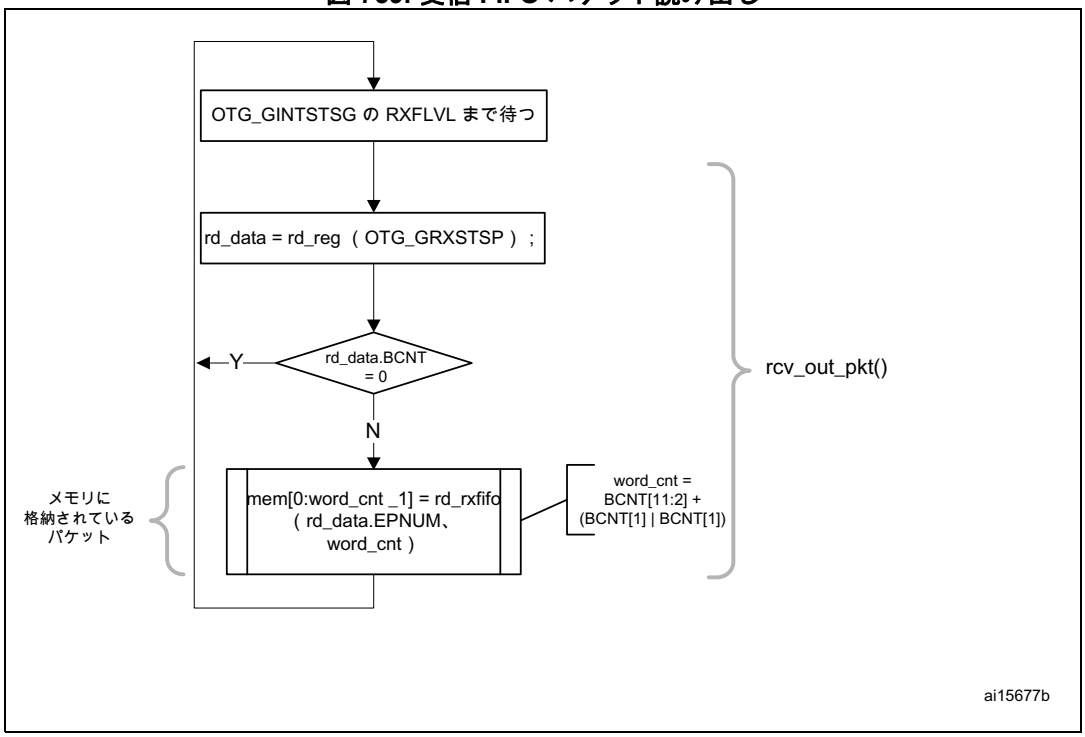
- **パケットの読み出し**

このセクションでは、受信 FIFO からパケット（OUT データおよび SETUP パケット）を読み出す方法について説明します。

1. RXFLVL 割り込み（OTG_GINTSTS レジスタ）を受信すると、アプリケーションは、受信ステータスポップレジスタ（OTG_GRXSTSP）を読み出す必要があります。
2. アプリケーションは、RXFLVL = 0（OTG_GINTMSK レジスタ）を書き込むことによって、受信 FIFO からそのパケットが読み出されるまで、RXFLVL 割り込み（OTG_GINTSTS レジスタ）をマスクすることができます。
3. 受信パケットのバイトカウントが 0 でない場合、データのバイトカウントは受信データ FIFO からポップされ、メモリに格納されます。受信パケットのバイトカウントが 0 の場合、受信データ FIFO からデータはポップされません。
4. FIFO のパケット受信ステータスの読み出し結果は、以下のいずれかを示します。
   a) グローバル OUT NAK のパターン
   PKTSTS = グローバル OUT NAK、BCNT = 0x000、EPNUM = （0x0）、DPID = （0b00）。
   これらのデータは、グローバル OUT NAK ビットが有効になっていることを示します。
   b) SETUP パケットのパターン
   PKTSTS = SETUP、BCNT = 0x008、EPNUM = 制御 EP 番号、DPID = DATA0。これらのデータは、指定されたエンドポイントの SETUP パケットを受信 FIFO から読み出せることを示します。
   c) セットアップステージ終了パターン
   PKTSTS = セットアップステージ終了、BCNT = 0x0、EPNUM = 制御 EP 番号、DPID = （0b00）。
   これらのデータは、指定されたエンドポイントのセットアップステージが完了し、データステージが開始されたことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定されたコントロール OUT エンドポイントでセットアップ割り込みをアサートします。
   d) データ OUT パケットのパターン
   PKTSTS = データ OUT、BCNT = 受信データ OUT パケットのサイズ（0 ≤ BCNT ≤ 1 024）、EPNUM = パケットを受信した EPNUM、DPID = 実際のデータ PID。
   e) データ転送完了パターン
   PKTSTS = データ OUT 転送終了、BCNT = 0x0、EPNUM = データ転送が完了した OUT EP 番号、DPID = （0b00）。
   これらのデータは、指定された OUT エンドポイントの OUT データ転送が完了したことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定された OUT エンドポイントで転送完了割り込みをアサートします。
5. データペイロードが受信 FIFO からポップされた後、RXFLVL 割り込み（OTG_GINTSTS）がマスク解除されなければなりません。
6. 手順 1～5 は、アプリケーションが OTG_GINTSTS レジスタの RXFLVL による割り込みラインのアサーションを検出するたびに繰り返されます。空の受信 FIFO を読み出すと、定義されていないコア動作を引き起こすことがあります。

図 759 上記の手順のフローチャートです。

図 759. 受信 FIFO パケット読み出し



SETUP トランザクション

このセクションでは、コアが SETUP パケットを処理する方法と、SETUP トランザクションを処理するアプリケーションのシーケンスについて説明します。

- **アプリケーションの要件**

1. SETUP パケットを受信するには、コントロール OUT エンドポイントの STUPCNT フィールド（OTG_DOEPTSIZx レジスタ）が、ゼロでない値にプログラムされなければなりません。アプリケーションが STUPCNT フィールドをゼロでない値にプログラムすると、NAK のステータスと OTG_DOEPCTLx レジスタの EPENA ビットの設定に関係なく、コアは、SETUP パケットを受信して、受信 FIFO に書き込みます。STUPCNT フィールドは、コントロールエンドポイントが SETUP パケットを受信するたびにデクリメントされます。SETUP パケットを受信する前に、STUPCNT フィールドが適切な値にプログラムされていなかった場合、コアは SETUP パケットを受信し、STUPCNT フィールドをデクリメントしますが、アプリケーションはコントロール転送のセットアップステージで受信した SETUP パケットの正しい数を判定できないことがあります。
   – OTG_DOEPTSIZx レジスタの STUPCNT = 3
2. コントロールエンドポイントで 3 つまでの SETUP パケットを受信するためには、アプリケーションは、常に受信データ FIFO に余分のスペースを割り当てておく必要があります。
   – 確保すべきスペースは 10 ワード分です。最初の SETUP パケット用に 3 ワードが必要であり、セットアップステージ終了ワードのために 1 ワード、すべてのコントロールエンドポイントにわたって 2 つの余分な SETUP パケットを格納するために 6 ワードが必要です。
   – 8 バイトの SETUP データと 4 バイトの SETUP ステータス（セットアップパケットパターン）を格納するには、SETUP パケットあたり 3 ワードが必要です。コアは、このスペースを受信データの中に確保します。
   – FIFO は SETUP データ書き込み専用であり、このスペースをデータパケットのために使うことはありません。

3. アプリケーションは、受信 FIFO から SETUP パケットの 2 ワードを読み出す必要があります。
4. アプリケーションは受信 FIFO からセットアップステージ終了ワードを読み出して、破棄する必要があります。

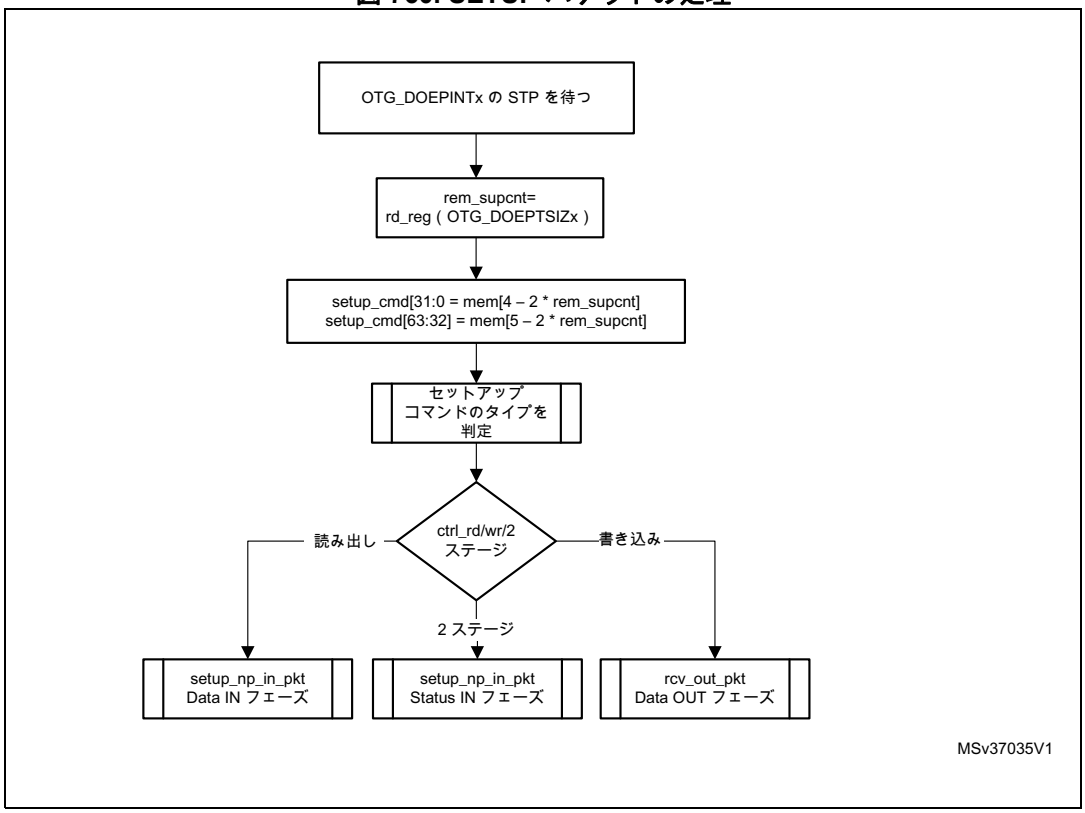
- **内部データフロー**

1. SETUP パケットが受信されると、コアは受信 FIFO 内の空きスペースをチェックせずに、また、エンドポイントの NAK および STALL ビットの設定に関係なく、受信したデータを受信 FIFO に書き込みます。
   – コアは、SETUP パケットが受信されたコントロール IN/OUT エンドポイントの IN NAK および OUT NAK ビットを内部でセットします。
2. USB 上で受信された各 SETUP パケットについて、3 ワードのデータが受信 FIFO に書き込まれ、STUPCNT フィールドが 1 ずつデクリメントされます。
   – 最初のワードは、コアが内部的に使用する制御情報を含みます。
   – 2 番目のワードは、SETUP コマンドの最初の 4 バイトを含みます。
   – 3 番目のワードは、SETUP コマンドの最後の 4 バイトを含みます。
3. セットアップステージがデータ IN/OUT ステージに変わると、コアは、エントリ（セットアップステージ終了ワード）を受信 FIFO に書き込んで、セットアップステージの完了を示します。
4. AHB 側では、SETUP パケットはアプリケーションによってエンプティにされます。
5. アプリケーションが受信 FIFO からセットアップステージ終了ワードをポップすると、コアは、STUP 割り込み（OTG_DOEPINTx）でアプリケーションに割り込んで、受信した SETUP パケットを処理できることを示します。
6. コアは、コントロール OUT エンドポイントのエンドポイントイネーブルビットをクリアします。

- **アプリケーションのプログラミングシーケンス**：

1. OTG_DOEPTSIZx レジスタをプログラムします。
   – STUPCNT = 3
2. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
3. STUP 割り込み（OTG_DOEPINTx）のアサートは、SETUP データ転送が正常に完了したことを示します。
   – この割り込み時、アプリケーションは、OTG_DOEPTSIZx レジスタを読み出して、受信した SETUP パケットの数を確認し、最後に受信した SETUP パケットを処理する必要があります。

図 760. SETUP パケットの処理



- **3 つを超える連続 SETUP パケットの処理**

USB 2.0 仕様に従い、通常 SETUP パケットエラー時にはホストは同じエンドポイントに対して 3 つを超える連続 SETUP パケットを送信しません。ただし USB 2.0 仕様では、ホストが同じエンドポイントに送信できる連続 SETUP パケットの数を制限してはいません。この条件が発生すると、OTG_HS コントローラは、割り込み（OTG_DOEPINTx の B2BSTUP）を生成します。

- **グローバル OUT NAK の設定**

内部データフロー：

1. アプリケーションがグローバル OUT NAK（OTG_DCTL の SGONAK ビット）をセットすると、コアは SETUP パケットを除き、受信 FIFO へのデータの書き込みを停止します。受信FIFO の使用可能なスペースの有無に関係なく、非アイソクロナス OUT トークンは NAK ハンドシェイクレスポンスを受信し、コアはアイソクロナス OUT データパケットを無視します。
2. コアは、グローバル OUT NAK パターンを受信 FIFO に書き込みます。アプリケーションがこのデータパターンを書き込むには、受信 FIFO に十分なスペースを確保する必要があります。
3. アプリケーションがグローバル OUT NAK パターンワードを受信 FIFO からポップすると、コアは、GONAKEFF 割り込み（OTG_GINTSTS）をセットします。
4. この割り込みを検出したアプリケーションは、コアがグローバル OUT NAK モードにあるとみなすことができます。アプリケーションは OTG_DCTL の SGONAK ビットをクリアすることによって、この割り込みをクリアできます。

アプリケーションのプログラミングシーケンス：

1. 受信 FIFO でのデータの受信も停止するには、アプリケーションは、以下のフィールドをプログラムすることによって、グローバル OUT NAK ビットをセットする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割り込みのアサートを待ちます。アサートされた場合、この割り込みは、コアが SETUP パケット以外のいかなるタイプのデータの受信も停止したことを示します。
3. アプリケーションは OTG_DCTL の SGONAK ビットをセットした後、コアが GONAKEFF 割り込み（OTG_GINTSTS）をアサートする前に、有効な OUT パケットを受信することができます。
4. アプリケーションは、OTG_GINTMSK レジスタの GONAKEFFM ビットに書き込むことによって、この割り込みを一時的にマスクできます。
   – OTG_GINTMSK レジスタの GONAKEFFM = 0
5. アプリケーションがグローバル OUT NAK モードを終了する準備ができたときには OTG_DCTL レジスタの SGONAK ビットをクリアする必要があります。これによって GONAKEFF 割り込み（OTG_GINTSTS）もクリアされます。
   – OTG_DCTL レジスタの CGONAK = 1
6. アプリケーションがこの割り込みをマスクするのが早すぎた場合は、以下のようにしてマスク解除を行う必要があります。
   – OTG_GINTMSK レジスタの GONAKEFFM = 1

- **OUT エンドポイントの無効化**

アプリケーションが有効にした OUT エンドポイントを無効にするには、このシーケンスを使用する必要があります。

アプリケーションのプログラミングシーケンス：

1. OUT エンドポイントを無効にする前に、アプリケーションは、コアのグローバル OUT NAK モードを有効にする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割り込みを待ちます。
3. 以下のフィールドをプログラムすることによって、必要な OUT エンドポイントを無効にします。
   – OTG_DOEPCTLx レジスタの EPDIS = 1
   – OTG_DOEPCTLx レジスタの SNAK = 1
4. EPDISD 割り込み（OTG_DOEPINTx）を待ちます。これは OUT エンドポイントが完全に無効にされたことを示します。EPDISD 割り込みがアサートされると、コアは、以下のビットもクリアします。
   – OTG_DOEPCTLx レジスタの EPDIS = 0
   – OTG_DOEPCTLx レジスタの EPENA = 0
5. アプリケーションは、無効にされていない他のエンドポイントからのデータ受信を開始するために、グローバル OUT NAK ビットをクリアする必要があります。
   – OTG_DCTL レジスタの SGONAK = 0

- **一般の非アイソクロナス OUT データ転送**

このセクションでは、通常の非アイソクロナス OUT データ転送（コントロール、バルク、または割り込み）について説明します。

アプリケーションの要件：

1. OUT 転送をセットアップする前に、アプリケーションは OUT 転送の一部として受信されるすべてのデータを収容できるバッファをメモリ内で割り当てる必要があります。
2. OUT 転送の場合、エンドポイントの転送サイズレジスタの転送サイズフィールドは、エンドポイントの最大パケットサイズの倍数でなければならず、ワードの境界に揃えられていなければなりません。
   - 転送サイズ [EPNUM] = *n* * (MPSIZ[EPNUM] + 4 – (MPSIZ[EPNUM] mod 4))
   - パケットカウント [EPNUM] = *n*
   - *n* > 0
3. OUT エンドポイント割り込み時には、アプリケーションは、エンドポイントの転送サイズレジスタを読み出して、メモリ内のペイロードのサイズを計算しなければなりません。受信したペイロードのサイズが、プログラムされた転送サイズより小さいこともあります。
   - メモリ内のペイロードのサイズ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   - このペイロードが受信された USB パケットの数 = アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

内部データフロー：

1. アプリケーションがデータを受信するためには、エンドポイント固有レジスタの転送サイズおよびパケットカウントのフィールドを設定し、NAK ビットをクリアし、エンドポイントを有効にする必要があります。
2. NAK ビットがクリアされると、コアは、データの受信を開始し、受信 FIFO にスペースがある限り、データを受信 FIFO に書き込みます。USB で受信された各データパケットについて、データパケットとそのステータスが受信 FIFO に書き込まれます。受信 FIFO にパケット（最大パケットサイズまたはショートパケット）が書き込まれるたびに、そのエンドポイントのパケットカウントフィールドが 1 ずつデクリメントされます。
   - 受信された OUT データパケットのデータ CRC が不良な場合、受信 FIFO から自動的に一掃されます。
   - USB 上のパケットに対して ACK を送信した後、コアは ACK を検出できないホストが再送信する非アイソクロナス OUT データパケットを破棄します。アプリケーションは、同じエンドポイント上では、同じデータ PID を持つ複数の連続データ OUT パケットを検出しません。この場合、パケットカウントはデクリメントされません。
   - 受信 FIFO にスペースがない場合、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれません。さらに、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
   - 上記の 3 つのケースのすべてにおいて、データは受信 FIFO に書き込まれないので、パケットカウントはデクリメントされません。
3. パケットカウントが 0 になるか、エンドポイント上でショートパケットが受信されると、そのエンドポイントの NAK ビットがセットされます。NAK ビットがセットされると、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれず、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
4. データが受信 FIFO に書き込まれた後、アプリケーションは受信 FIFO からデータを読み出して、エンドポイントあたり一度に 1 パケットずつ外部メモリに書き込みます。
5. AHB 上で外部メモリへのパケットの書き込みが終わるたびに、書き込まれたパケットのサイズだけエンドポイントの転送サイズがデクリメントされます。

6. 以下の条件の 1 つで OUT エンドポイントの OUT データ転送完了パターンが受信 FIFO に書き込まれます。
   – 転送サイズが 0、およびパケットカウントが 0。
   – 受信 FIFO に書き込まれた最後の OUT データパケットがショートパケット。
   (0 ≤ パケットサイズ < 最大パケットサイズ)
7. アプリケーションがこのエントリ（OUT データ転送完了）をポップすると、エンドポイントの転送完了割り込みが生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、EPENA ビットおよび CNAK ビットをセットします。
   – OTG_DOEPCTLx レジスタの EPENA = 1
   – OTG_DOEPCTLx レジスタの CNAK = 1
3. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割り込み（OTG_DOEPINTx）のアサートは、非アイソクロナス OUT データ転送が正常に完了したことを示します。
5. OTG_DOEPTSIZx レジスタを読み出して、受信したデータペイロードのサイズを確認します。

- **一般のアイソクロナス OUT データ転送**

このセクションでは、通常のアイソクロナス OUT データ転送について説明します。

アプリケーションの要件：

1. 非アイソクロナス OUT データ転送のアプリケーションの要件はすべて、アイソクロナス OUT データ転送にも適用されます。
2. アイソクロナス OUT データ転送の場合、転送サイズおよびパケットカウントフィールドは、常に、単一フレームで受信できる最大パケットサイズのパケット数を設定しなければなりません。アイソクロナス OUT データ転送は、複数のフレームにまたがってはいけません。
3. アプリケーションは、周期的フレームの終わり（OTG_GINTSTS の EOPF 割り込み）の前に、受信 FIFO からすべてのアイソクロナス OUT データパケット（データおよびステータス）を読み出す必要があります。
4. 次のフレームでデータを受信するには EOPF（OTG_GINTSTS）の後と SOF（OTG_GINTSTS）の前に、アイソクロナス OUT エンドポイントを有効にしなければなりません。

内部データフロー：

1. アイソクロナス OUT エンドポイントの内部データフローは、非アイソクロナス OUT エンドポイントの内部データフローと基本的に同じですが、少し異なったところがあります。
2. エンドポイントイネーブルビットををセットし、NAK ビットをクリアすることによって、アイソクロナス OUT エンドポイントが有効にされたときには、偶数／奇数フレームビットも適切にセットされなければなりません。コアは、以下の条件が満たされた場合に限り、アイソクロナス OUT エンドポイント上で特定のフレームのデータを受信します。
   – EONUM（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
3. アプリケーションが受信 FIFO からアイソクロナス OUT データパケット（データとステータス）を完全に読み出すと、コアは OTG_DOEPTSIZx レジスタの RXDPID フィールドを、受信 FIFO から読み出された最後のアイソクロナス OUT データパケットのデータ PID で更新します。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、エンドポイントイネーブル、ClearNAK、および偶数／奇数フレームの各ビットをセットします。
   – EPENA = 1
   – CNAK = 1
   – EONUM = （0：偶数／1：奇数）
3. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からのデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割り込み（OTG_DOEPINTx）のアサートは、アイソクロナス OUT データ転送が正常に完了したことを示します。この割り込みは、必ずしもメモリ内のデータが良好であることを意味しません。
5. この割り込みは、アイソクロナス OUT 転送で必ず検出されるとは限りません。その代わり、アプリケーションは OTG_GINTSTS で INCOMPISOOUT 割り込みを検出することができます。
6. OTG_DOEPTSIZx レジスタを読み出して、受信した転送データのサイズを確認し、フレームで受信したデータの有効性を確認します。アプリケーションは、以下の条件の 1 つが満たされた場合のみ、メモリに受信されたデータを有効として扱う必要があります。
   – RXDPID = DATA0（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 1
   – RXDPID = DATA1（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 2
   – RXDPID = D2（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 3

     このペイロードが受信された USB パケットの数 =
     アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

   アプリケーションは、無効なデータパケットを破棄できます。

- **不完全アイソクロナス OUT データ転送**

このセクションでは、アイソクロナス OUT データパケットがコアの内部でドロップされたときのアプリケーションのプログラミングシーケンスについて説明します。

内部データフロー：

1. アイソクロナス OUT エンドポイントの場合、XFRC 割り込み（OTG_DOEPINTx）は必ずアサートされるわけではありません。コアがアイソクロナス OUT データパケットをドロップした場合、アプリケーションは以下の状況で XFRC 割り込み（OTG_DOEPINTx）の検出に失敗することがあります。
   – 受信 FIFO が完全な ISO OUT データパケットを収容できない場合、コアは、受信した ISO OUT データをドロップします。
   – アイソクロナス OUT データパケットが CRC エラー付きで受信されたとき。
   – コアが受信したアイソクロナス OUT トークンが破損しているとき。
   – アプリケーションが受信 FIFO からデータを読み出すのに長時間かかっているとき。
2. コアが、すべてのアイソクロナス OUT エンドポイントへの転送を完了する前に周期的フレームの終わりを検出すると、不完全アイソクロナス OUT データ割り込み（OTG_GINTSTS レジスタの INCOMPISOOUT）をアサートして、アイソクロナス OUT エンドポイントのうち少なくとも 1 つで XFRC 割り込み（OTG_DOEPINTx）がアサートされていないことを示します。この時点

で、不完全転送のエンドポイントは有効なままですが USB 上のこのエンドポイントでは、アクティブな転送は進行していません。

アプリケーションのプログラミングシーケンス：

1. INCOMPISOOUT 割り込み（OTG_GINTSTS）のアサートは、現在のフレームで、少なくとも1つのアイソクロナス OUT エンドポイントが転送を完了していないことを示します。
2. アイソクロナス OUT データがエンドポイントから完全に出されていないためにこの割り込みが発生した場合、アプリケーションは処理を進める前に、受信 FIFO からすべてのアイソクロナス OUT データ（データとステータス）を出力する必要があります。
   – すべてのデータが受信 FIFO から出されると、アプリケーションは XFRC 割り込み（OTG_DOEPINTx）を検出することができます。この場合、アプリケーションは、次のフレームでアイソクロナス OUT データを受信するには、エンドポイントを再び有効にする必要があります。
3. INCOMPISOOUT 割り込み（OTG_GINTSTS）を受信すると、アプリケーションは、すべてのアイソクロナス OUT エンドポイントの制御レジスタ（OTG_DOEPCTLx）を読み出して、現在のマイクロフレームにおいて転送を完了しなかったエンドポイントを確認する必要があります。次の両方の条件が満たされた場合、エンドポイント転送は完了しません。
   – EONUM ビット（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
   – OTG_DOEPCTLx レジスタの EPENA = 1
4. 現在のフレーム番号が変更されないように、上記の手順は SOF 割り込み（OTG_GINTSTS）が検出される前に実行されなければなりません。
5. 不完全転送のアイソクロナス OUT エンドポイントの場合、アプリケーションは、メモリ内のデータを破棄し OTG_DOEPCTLx レジスタの EPDIS ビットをセットすることによってエンドポイントを無効にする必要があります。
6. EPDISD 割り込み（OTG_DOEPINTx）を待ち、次のフレームで新しいデータを受信するために、エンドポイントを有効にします。
   – コアがエンドポイントを無効にするには若干の時間がかかるので、アプリケーションは不良なアイソクロナスデータを受信した後、次のフレームのデータを受信できないことがあります。

- **非アイソクロナス OUT エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

1. コアをグローバル OUT NAK モードにします。
2. 必要なエンドポイントを無効にします。
   – エンドポイントを無効にするときには、OTG_DOEPCTL の SNAK ビットをセットする代わりに STALL = 1 にセットしてください（OTG_DOEPCTL）。
   STALL ビットは常に NAK ビットより優先されます。
3. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときは、STALL ビット（OTG_DOEPCTLx）をクリアしなければなりません。
4. アプリケーションが SetFeature.Endpoint Halt または ClearFeature.Endpoint Halt コマンドのためにエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは、アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

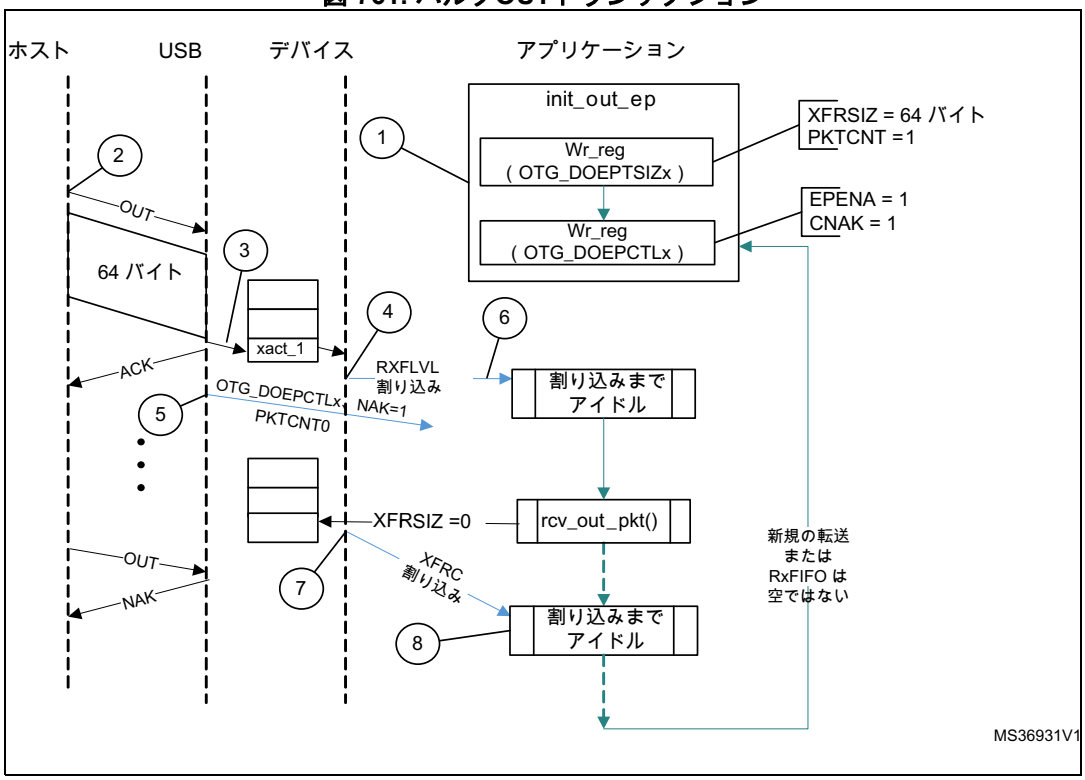
例

このセクションでは、いくつかの基本的な転送タイプとシナリオについて説明します。

- バルクOUTトランザクション

図 761 に USB から AHB への単一バルク OUT データパケットの受信と、このプロセスに伴うイベントを示します。

図 761. バルクOUTトランザクション



SetConfiguration/SetInterface コマンドの後、アプリケーションは CNAK = 1 および EPENA = 1（OTG_DOEPCTLx レジスタ）をセットし、OTG_DOEPTSIZx レジスタの適切な XFRSIZ および PKTCNT ビットをセットすることによって、すべての OUT エンドポイントを初期化します。

1. ホストは、データ（OUT トークン）のエンドポイントへの送信を試みます。
2. コアは、USB 上で OUT トークンを受信すると、スペースが使用可能なので、Rx FIFO にパケットを格納します。
3. Rx FIFO に完全なパケットを書き込んだ後、コアは RxFLVL 割り込み（OTG_GINTSTS）をアサートします。
4. USB パケットの PKTCNT 番号を受信すると、コアは、それ以上パケットが受信されないように、このエンドポイントの NAK ビットを内部でセットします。
5. アプリケーションは割り込みを処理して Rx FIFO からデータを読み出します。
6. アプリケーションがすべてのデータ（XFRSIZ に相当）を読み出すと、コアは、XFRC 割り込み（OTG_DOEPINTx）を生成します。
7. アプリケーションは割り込みを処理して、XFRC 割り込みビット（OTG_DOEPINTx）の設定を使用して、意図した転送が完了したかどうかを確認します。

IN データ転送

- **パケットの書き込み**

このセクションでは、専用の送信 FIFO が有効なときに、アプリケーションがデータパケットをエンドポイント FIFO に書き込む方法について説明します。

1. アプリケーションは、ポーリングまたは割り込みモードのいずれかを選択できます。
   - ポーリングモードでは、アプリケーションは、OTG_DTXFSTSx レジスタを読み出すことによってエンドポイント送信データ FIFO のステータスを監視して、データ FIFO に十分なスペースがあるかどうかを確認します。
   - 割り込みモードでは、アプリケーションは、TXFE 割り込み（OTG_DIEPINTx）を待ってから、OTG_DTXFSTSx レジスタを読み出して、データ FIFO 内に十分なスペースがあるかどうかを確認します。
   - 単一の非ゼロ長データパケットを書き込むには、パケット全体を書き込むためのスペースがデータ FIFO になければなりません。
   - ゼロ長パケットを書き込むには、アプリケーションは FIFO のスペースを考慮してはなりません。
2. 上記の方法の 1 つを使用して、アプリケーションが送信パケットを書き込むのに十分なスペースがあることを確認するときには、アプリケーションは、データをデータ FIFO に書き込む前に、まず、エンドポイント制御レジスタに書き込む必要があります。通常、アプリケーションは、エンドポイントイネーブルビットをセットする場合を除き、レジスタの内容を変更しないように、OTG_DIEPCTLx レジスタにリードモディファイライトを行う必要があります。

アプリケーションは、使用可能なスペースがあれば、同じエンドポイントに対する複数のパケットを送信 FIFO に書き込むことができます。周期的 IN エンドポイントの場合、アプリケーションは 1 つのマイクロフレームのパケットのみを書き込む必要があります。アプリケーションは、前のトランザクションの転送完了割り込みを受信した後でのみ、次の周期的トランザクションのパケットを書き込むことができます。

- **IN エンドポイント NAK の設定**

内部データフロー：

1. アプリケーションが特定のエンドポイントの IN NAK をセットすると、コアは、エンドポイントの送信 FIFO にデータがあるかどうかに関係なく、そのエンドポイントでのデータ送信を停止します。
2. 非アイソクロナス IN トークンは NAK ハンドシェイク応答を受信します。
   - アイソクロナス IN トークンはゼロデータ長のパケット応答を受信します。
3. コアは、OTG_DIEPCTLx の SNAK ビットに応答して、OTG_DIEPINTx の INEPNE 割り込み（IN エンドポイント NAK 有効）をアサートします。
4. この割り込みがアプリケーションによって検出されると、アプリケーションは、エンドポイントが IN NAK モードにあるとみなすことができます。この割り込みは、アプリケーションが OTG_DIEPCTLx の CNAK ビットをセットすることによってクリアできます。

アプリケーションのプログラミングシーケンス：

1. 特定の IN エンドポイントでのデータ送信を停止するには、アプリケーションは、IN NAK ビットをセットする必要があります。このビットをセットするには、以下のフィールドをプログラムする必要があります。
   - OTG_DIEPCTLx レジスタの SNAK = 1
2. OTG_DIEPINTx の INEPNE 割り込みのアサートを待ちます。この割り込みは、コアがエンドポイントでのデータ送信を停止したことを示します。

3. コアは、アプリケーションが NAK ビットをセットした後のNAK 有効割り込みがアサートされる前に、エンドポイントで有効な IN データを送信できます。
4. アプリケーションは OTG_DIEPMSK の INEPNEM ビットに書き込むことによって、この割り込みを一時的にマスクできます。
   – OTG_DIEPMSK レジスタの INEPNEM = 0
5. エンドポイント NAK モードを終了するには、アプリケーションは OTG_DIEPCTLx の NAKステータスビット（NAKSTS）をクリアする必要があります。これによって INEPNE 割り込み（OTG_DIEPINTx）もクリアされます。
   – OTG_DIEPCTLx レジスタの CNAK = 1
6. アプリケーションがこの割り込みをマスクするのが早すぎた場合、以下のようにしてマスク解除を行う必要があります。
   – OTG_DIEPMSK レジスタの INEPNEM = 1

- **IN エンドポイントディセーブル**

以前に有効にされた特定の IN エンドポイントを無効にするには、以下のシーケンスを使用します。

アプリケーションのプログラミングシーケンス：

1. アプリケーションは、AHB でのデータ書き込みを停止して、IN エンドポイントを無効にする必要があります。
2. アプリケーションは、エンドポイントを NAK モードに設定する必要があります。
   – OTG_DIEPCTLx レジスタの SNAK = 1
3. OTG_DIEPINTx の INEPNE 割り込みを待ちます。
4. 無効にしなければならないエンドポイントについて OTG_DIEPCTLx レジスタの以下のビットをセットします。
   – OTG_DIEPCTLx レジスタの EPDIS = 1
   – OTG_DIEPCTLx レジスタの SNAK = 1
5. OTG_DIEPINTx レジスタの EPDISD 割り込みのアサートは、コアが指定されたエンドポイントを完全に無効にしたことを示します。割り込みのアサートとともに、コアは、以下のビットもクリアします。
   – OTG_DIEPCTLx レジスタの EPENA = 0
   – OTG_DIEPCTLx レジスタの EPDIS = 0
6. アプリケーションは、周期的 IN EP の OTG_DIEPTSIZx レジスタを読み出して、エンドポイント上のどれだけのデータが USB で送信されたかを計算する必要があります。
7. アプリケーションは、OTG_GRSTCTL レジスタの以下のフィールドを設定することによって、エンドポイント送信 FIFO 内のデータを一掃する必要があります。
   – TXFNUM（OTG_GRSTCTL）= エンドポイント送信 FIFO 番号
   – TXFFLSH（OTG_GRSTCTL ）= 1

アプリケーションは、TXFFLSH ビットがコアによってクリアされるまで（すなわち、一掃動作の終了まで）、OTG_GRSTCTL レジスタにポーリングする必要があります。このエンドポイントで新しいデータを送信するために、アプリケーションは後で、エンドポイントを再び有効にできます。

- **一般の非周期的 IN データ転送**

アプリケーションの要件：

1. IN 転送をセットアップする前に、アプリケーションは、IN 転送の一部として送信されるすべてのデータが単一バッファの一部であることを確認する必要があります。
2. IN 転送の場合、エンドポイント転送サイズレジスタの転送サイズフィールドは、最大パケットサイズの複数のパケットと単一のショートパケットから成るペイロードを表します。このショートパケットは、転送の最後に送信されます。
   – 転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには：
     転送サイズ [EPNUM] = *x* * MPSIZ[EPNUM] + sp
     （sp > 0）の場合、パケットカウント [EPNUM] = x + 1。
     そうでない場合、パケットカウント [EPNUM] = *x*
   – 単一のゼロ長データパケットを送信するには：
     転送サイズ [EPNUM] = 0
     パケットカウント [EPNUM] = 1
   – 転送の終わりに少数の最大パケットサイズのパケットとゼロ長データパケットを送信するには、アプリケーションは転送を 2 つの部分に分ける必要があります。最初の部分では最大パケットサイズのデータパケットを送信し、2 番目の部分ではゼロ長データパケットのみを送信します。
     最初の転送：転送サイズ [EPNUM] = *x* * MPSIZ[epnum]；パケットカウント = *n*；
     2 番目の転送：転送サイズ [EPNUM] = 0；パケットカウント = 1；
3. エンドポイントがデータ転送のために有効にされると、コアは、転送サイズレジスタを更新します。IN 転送の終了時に、アプリケーションは、転送サイズレジスタを読み出して、送信 FIFO にポストされたデータのうち、どれだけが USB で送信されたかを確認する必要があります。
4. 送信 FIFO にフェッチされたデータ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – USB で送信されたデータ =（アプリケーションがプログラムした初期パケットカウント – コアが更新した最終パケットカウント）* MPSIZ[EPNUM]
   – USB 上でこれから送信されるデータ =（アプリケーションがプログラムした初期転送サイズ – USB で送信されたデータ）

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. アプリケーションは、要求されたデータをエンドポイントの送信 FIFO に書き込む必要もあります。
3. アプリケーションによってパケットが送信 FIFO に書き込まれるたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0になるまで、データはアプリケーションによってメモリからフェッチされます。FIFO にデータを書き込んだ後、「FIFO 内のパケット数」はインクリメントされます（これは 3 ビットのカウントであり、各 IN エンドポイントの送信 FIFO について、コアによって内部で維持されます。IN エンドポイント FIFO 内にコアによって一度に維持されるパケットの最大数は 8 です）。ゼロ長のパケットの場合、各 FIFO について、FIFO 内にデータがないことを示す個別のフラグがセットされます。
4. データが送信 FIFO に書き込まれると、コアは、IN トークンの受信時に読み出します。非アイソクロナス IN データパケットが ACK ハンドシェイクとともに送信されるたびに、エンドポイントのパケットカウントは、0 になるまで 1 ずつデクリメントされます。パケットカウントは、タイムアウト時にはデクリメントされません。

5. ゼロ長パケットの場合（内部ゼロ長フラグで示されます）、コアは IN トークンのゼロ長パケットを送出し、パケットカウントフィールドをデクリメントします。
6. 受信された IN トークンのデータが FIFO になく、そのエンドポイントのパケットカウントフィールドがゼロの場合、コアは、そのエンドポイントについて、「TxFIFO がエンプティのとき IN トークンが受信された」という割り込み（ITTXFE 割り込み）を生成します。ただし、エンドポイントの NAK ビットがセットされていない場合に限ります。コアは、USB 上の非アイソクロナスエンドポイントに対して NAK ハンドシェイクで応答します。
7. コアは、内部で FIFO ポインタを巻き戻し、タイムアウト割り込みは生成されません。
8. 転送サイズが 0 であり、パケットカウントが 0 のとき、エンドポイントの転送完了割り込み（XFRC）が生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPTSIZx レジスタで転送サイズと対応するパケットカウントをプログラムします。
2. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムし、CNAK ビットおよび EPENA（エンドポイントイネーブル）ビットをセットします。
3. 非ゼロ長データパケットを送信するときには、アプリケーションは、OTG_DTXFSTSx レジスタ（x はそのエンドポイントに関連する FIFO 番号）をポーリングして、データ FIFO に十分なスペースがあるかどうかを確認する必要があります。アプリケーションは、データを書き込む前に、オプションで TXFE（OTG_DIEPINTx）を使用することができます。

- **一般の周期的 IN データ転送**

このセクションでは、典型的な周期的 IN データ転送について説明します。

アプリケーションの要件：

1. *2646 ページの一般の非周期的 IN データ転送* のアプリケーションの要件 1、2、3、および 4 は、要件 2 が少し変更されていることを除いて、周期的 IN データ転送にも適用されます。
   - アプリケーションは複数の最大パケットサイズのデータパケットの送信、または複数の最大パケットサイズのパケットに最後にショートパケットを加えた送信のみを行うことができます。転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには、以下の条件が満たされなければなりません。

     転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
     （*x* は整数 ≥ 0、および 0 ≤ sp < MPSIZ[EPNUM]）

     （sp > 0）の場合、パケットカウント [EPNUM] = *x* + 1
     そうでない場合、パケットカウント [EPNUM] = *x*；

     MCNT[EPNUM] = パケットカウント [EPNUM]
   - アプリケーションは転送の最後にゼロ長データパケットを送信することはできません。自分で単一のゼロ長パケットを送信することは可能です。単一のゼロ長データパケットを送信するには：
   - 転送サイズ [EPNUM] = 0

     パケットカウント [EPNUM] = 1

     MCNT[EPNUM] = パケットカウント [EPNUM]
2. アプリケーションは一度に 1 フレームのデータ転送のみをスケジュールすることができます。
   - (MCNT – 1) × MPSIZ ≤ XFERSIZ ≤ MCNT × MPSIZ
   - PKTCNT = MCNT（OTG_DIEPTSIZx レジスタ）
   - XFERSIZ < MCNT * MPSIZ の場合、転送の最後のデータパケットはショートパケットです。

– 以下のことに注意してください。MCNT は OTG_DIEPTSIZx レジスタに、MPSIZ は OTG_DIEPCTLx レジスタに、PKTCNT は OTG_DIEPTSIZx レジスタに、また XFERSIZ は OTG_DIEPTSIZx レジスタにあります。

3. フレームで送信される完全なデータは、IN トークンが受信される前に、アプリケーションによって送信 FIFO に書き込まれなければなりません。IN トークンが受信されたときに、フレームあたりで送信されるデータの 1 ワードが送信 FIFO 内で欠落していても、コアは FIFO がエンプティである場合のように動作します。送信 FIFO がエンプティのとき：
   – アイソクロナス IN エンドポイントに対して USB 上でゼロデータ長のパケットが送信されます。
   – IN エンドポイントに割り込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. また、アプリケーションは、要求されたデータをエンドポイントの関連する送信 FIFO に書き込む必要があります。
3. アプリケーションがパケットを送信 FIFO に書き込むたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0 になるまで、アプリケーションメモリからデータがフェッチされます。
4. 周期的エンドポイントの IN トークンが受信されると、コアは、FIFO のデータ（ある場合）を送信します。フレームの完全なデータペイロード（専用 FIFO モードでは完全なパケット）が FIFOに存在しない場合、コアは、エンドポイントに対する Tx FIFO エンプティ割り込みがおきたときに IN トークンの受信を発生します。
   – アイソクロナス IN エンドポイントに対して、USB 上でゼロ長のデータパケットが送信されます。
   – IN エンドポイントに割り込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。
5. エンドポイントのパケットカウントは、以下の条件のもとで 1 ずつデクリメントされます。
   – アイソクロナスエンドポイントに対して、ゼロ長または非ゼロ長データパケットが送信されたとき。
   – 割り込みエンドポイントに対して、ACK ハンドシェイクが送信されたとき。
   – 転送サイズとパケットカウントの両方が 0 のとき、エンドポイントに対して転送完了割り込みが生成され、エンドポイントイネーブルビットがクリアされます。
6. 「周期的フレームインターバル」（OTG_DCFG の PFIVL によって制御されます）で、コアが現在のフレームについてスケジュールされたアイソクロナス IN エンドポイント FIFO のいずれかがノットエンプティであることを検出すると、コアは OTG_GINTSTS の IISOIXFR 割り込みを生成します。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムして、CNAK および EPENA ビットをセットします。
2. 次のフレームで送信するデータを送信 FIFO に書き込みます。
3. OTG_DIEPINTx レジスタの ITTXFE 割り込みのアサートは、アプリケーションが送信するすべてのデータを送信 FIFO にまだ書き込んでいないことを示します。
4. この割り込みが検出されたときに、割り込みエンドポイントがすでに有効だった場合、割り込みは無視されます。まだ有効になっていなかった場合は、次の IN トークン試行の際にデータを送信できるように、エンドポイントを有効にします。

5. OTG_DIEPINTx レジスタの ITTXFE 割り込みがない状態での XFRC 割り込み（OTG_DIEPINTx）のアサートは、アイソクロナス IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読み出し結果は、転送サイズ = 0、かつパケットカウント = 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
6. ITTXFE 割り込み（OTG_DIEPINTx）の有無にかかわらず、XFRC 割り込み（OTG_DIEPINTx）のアサートは、割り込み IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読み出し結果は、転送サイズ = 0、かつパケットカウント= 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
7. 前述のどの割り込みもない状態での OTG_GINTSTS の不完全アイソクロナス IN 転送（IISOIXFR）割り込みのアサートは、コアが現在のフレームで少なくとも 1 つの周期的 IN トークンも受信しなかったことを示します。

- **不完全アイソクロナス IN データ転送**

このセクションでは、アプリケーションが不完全なアイソクロナス IN データ転送で行う必要があることについて説明します。

内部データフロー：

1. アイソクロナス IN 転送は､以下の条件のいずれかがあてはまる場合に不完全として扱われます。
   a) コアが、少なくとも 1 つのアイソクロナス IN エンドポイントで破損したアイソクロナス IN トークンを受信した場合。この場合、アプリケーションは、不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR ビット）を検出します。
   b) アプリケーションが完全なデータペイロードを送信 FIFO に書き込むのに時間がかかり、完全なデータペイロードが FIFO に書き込まれる前に IN トークンが受信された場合。この場合、アプリケーションは、OTG_DIEPINTx の Tx FIFO エンプティ割り込み時に IN トークンの受信を検出します。最終的には、周期的フレームの終わりでの不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR）になるので、アプリケーションはこの割り込みを無視することができます。

   コアは、受信した IN トークンに応答して、USB 上にゼロ長データパケットを送信します。
2. アプリケーションは、できるだけ速やかに、送信 FIFO へのデータペイロードの書き込みを停止する必要があります。
3. アプリケーションは、エンドポイントの NAK ビットとディセーブルビットをセットする必要があります。
4. コアは、エンドポイントを無効にし、ディセーブルビットをクリアし、エンドポイントのエンドポイントディセーブル割り込みをアサートします。

アプリケーションのプログラミングシーケンス：

1. 最終的には不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS）になるので、アプリケーションは、アイソクロナス IN エンドポイントで OTG_DIEPINTx の Tx FIFO エンプティ割り込み時に受信された IN トークンを無視することができます。
2. 不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS）のアサートは、少なくとも 1 つのアイソクロナス IN エンドポイントで不完全アイソクロナス IN 転送があったことを示します。
3. アプリケーションは、すべてのアイソクロナス IN エンドポイントのエンドポイント制御レジスタを読み出して、不完全 IN データ転送があるエンドポイントを検出する必要があります。
4. アプリケーションは、AHB 上でこれらのエンドポイントに関連する周期的送信 FIFO へのデータの書き込みを停止する必要があります。
5. OTG_DIEPCTLx レジスタの以下のフィールドをプログラムして、エンドポイントを無効にします。
   – OTG_DIEPCTLx レジスタの SNAK = 1
   – OTG_DIEPCTLx レジスタの EPDIS = 1

6. OTG_DIEPINTx のエンドポイントディセーブル割り込みのアサートは、コアがエンドポイントを無効にしたことを示します。
   – この時点で、アプリケーションは、次のマイクロフレームで新しい転送を行うために、関連する送信 FIFO 内のデータを一掃するか、エンドポイントを有効にすることによって、FIFO 内の既存のデータに上書きする必要があります。データを一掃するには、アプリケーションは OTG_GRSTCTL レジスタを使用する必要があります。

- **非アイソクロナス IN エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

アプリケーションのプログラミングシーケンス：

1. 停止する IN エンドポイントを無効にします。STALL ビットもセットします。
2. エンドポイントがすでに有効になっているときには、OTG_DIEPCTLx の EPDIS = 1 にします。
   – OTG_DIEPCTLx レジスタの STALL = 1 にします。
   – STALL ビットは常に NAK ビットより優先されます。
3. エンドポイントディセーブル割り込み（OTG_DIEPINTx）のアサートは、コアが指定されたエンドポイントを無効にしたことをアプリケーションに示します。
4. アプリケーションは、エンドポイントのタイプに応じて、非周期的または周期的 FIFO をフラッシュする必要があります。非周期的エンドポイントの場合、アプリケーションは、データを送信するために、停止する必要がない他の非周期的エンドポイントを再び有効にする必要があります。
5. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときには、OTG_DIEPCTLx の STALL ビットがクリアされなければなりません。
6. アプリケーションが SetFeature.Endpoint Halt コマンドまたは ClearFeature.Endpoint Halt コマンドのためにエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは、アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

特殊なケース：コントロール OUT エンドポイントの停止

コントロール転送のデータステージで、ホストが SETUP パケットで指定されたより多くの IN/OUT トークンを送信した場合、コアは、IN/OUT トークンを停止する必要があります。この場合、アプリケーションは、コアが SETUP パケットで指定されたデータ量を転送した後、コントロール転送のデータステージで、OTG_DIEPINTx の ITTXFE 割り込みと OTG_DOEPINTx の OTEPDIS 割り込みを有効にしなければなりません。その後、アプリケーションがこの割り込みを受信したとき、アプリケーションは、対応するエンドポイント制御レジスタの STALL ビットをセットし、この割り込みをクリアする必要があります。

57.15.7 最悪ケースの応答時間

OTG_HS コントローラがデバイスとして機能するとき、アイソクロナス OUT に続く任意のトークンで、最悪ケースの応答時間があります。この最悪ケース応答時間は、AHB クロック周波数に依存します。

コアのレジスタは AHB ドメインにあり、コアはこれらのレジスタを更新するまでは、別のトークンを受け入れません。アイソクロナストランザクションの場合、ハンドシェイクはなく、次のトークンがすぐに入ってくる可能性があるため、アイソクロナス OUT に続くトークンで最悪ケースが生じます。この最悪ケース値は、AHB クロックが PHY クロックと同じときには 7 PHY クロックです。AHB クロックが高速なほど、この値は小さくなります。

この最悪ケース条件が発生した場合、コアは、バルク／割り込みトークンに NAK で応答し、アイソクロナスおよび SETUP トークンをドロップします。ホストは、これを SETUP のタイムアウト条件

と解釈して、SETUP パケットを再試行します。アイソクロナス転送の場合、不完全アイソクロナス IN 転送割り込み（IISOIXFR）と不完全アイソクロナス OUT 転送割り込み（IISOOXFR）は、アイソクロナス IN/OUT パケットがドロップされたことをアプリケーションに知らせます。

OTG_GUSBCFG の TRDT の値の選択

TRDT（OTG_GUSBCFG）の値は、MAC が IN トークンを受信した後、FIFO ステータスと PFC ブロックから最初のデータを取得するまでの時間（PHY クロック数）です。この時間には、PHY クロックと AHB クロック間の同期遅延も含まれます。この場合の最悪ケース遅延は、AHB クロックが PHY クロックと同じときに発生します。この場合、遅延は 5 クロックです。

MAC が IN トークンを受信すると、この情報（トークンの受信）は PFC（PFC は AHB クロックで動作します）によって AHB に同期されます。次に、PFC は、SPRAM からデータを読み出し、それらをデュアルクロックソースバッファに書き込みます。MAC は、データをソースバッファ（深さ 4）から読み出します。

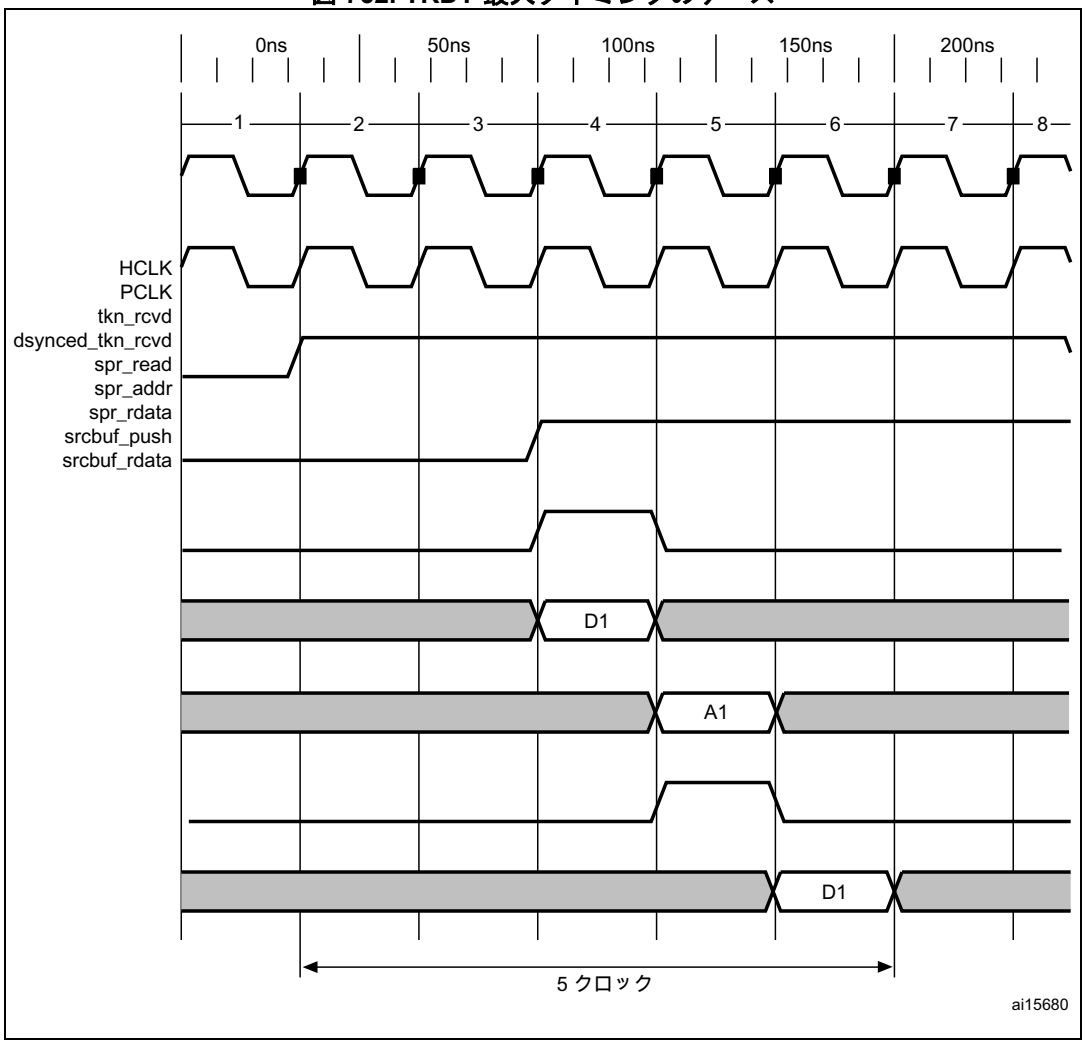
AHB が PHY より高い周波数で動作している場合、アプリケーションは TRDT（OTG_GUSBCFG）の値として、より小さな値を使用することができます。

図 762 に以下の信号を示します。

- tkn_rcvd：MAC から PFC への、トークンが受信されたという情報
- dynced_tkn_rcvd：PCLK から HCLK ドメインへの二重同期 tkn_rcvd
- spr_read：SPRAM への読み出し
- spr_addr：SPRAM へのアドレス
- spr_rdata：SPRAM からのデータの読み出し
- srcbuf_push：ソースバッファへのプッシュ
- srcbuf_rdata：ソースバッファからのデータの読み出しMAC によるデータの検出

TRDT の値を計算するには、または*表 487：TRDT 値（HS）*を参照してください。

図 762. TRDT 最大タイミングのケース



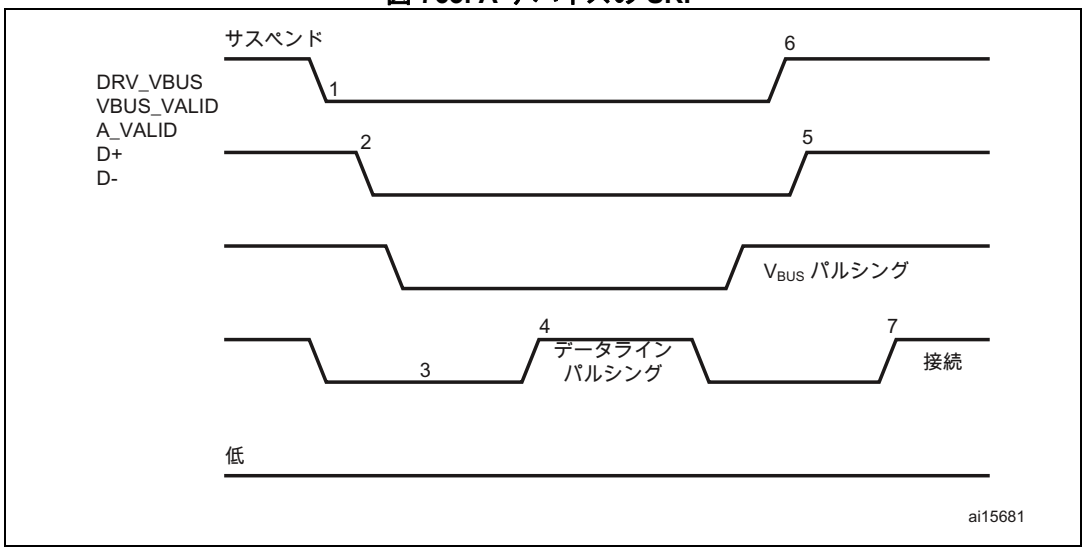
57.15.8 OTG プログラミングモデル

OTG_HS コントローラは HNP と SRP をサポートする OTG デバイスです。コアが A プラグに接続されると、A デバイスと呼ばれます。コアが B プラグに接続されると、B デバイスと呼ばれます。ホストモードでは、OTG_HS コントローラは、電力を節約するために V_{BUS} をオフにします。SPR は、B デバイスが A デバイスに V_{BUS} 電源をオンにするように信号を送る手段です。デバイスは、データラインパルシングと V_{BUS} パルシングの両方を実行する必要がありますが、ホストは SRP のデータラインパルシングか V_{BUS} パルシングのどちらかを検出することができます。HNP は、B デバイスがネゴシエイトし、役割をホストに切り替える手段です。HNP 後のネゴシエイトモードでは、B デバイスはバスをサスペンドし、その役割をデバイスに戻します。

A デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって OTG_HS コントローラは SRP を A デバイスとして検出できるようになります。

図 763. A デバイスの SRP



1. DRV_VBUS = PHY への V_{BUS} 駆動信号
   VBUS_VALID = PHY からの V_{BUS} 有効信号
   A_VALID = PHY への A ペリフェラル V_{BUS} レベル信号
   D+ = データ正ライン
   D- = データ負ライン

以下では、*図 763*に示された信号処理法を記載しています。

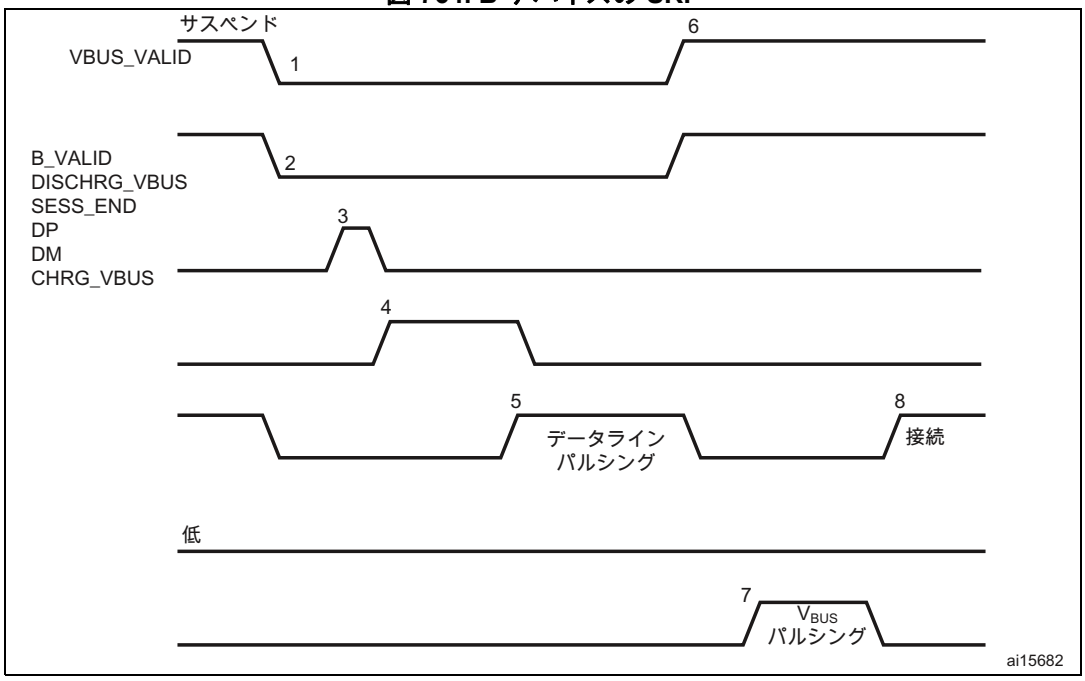
1. 節電のためには、アプリケーションは、バスがアイドル状態のときに、ホストポート制御およびステータスレジスタのポートサスペンドビットおよびポートパワービットに書き込むことによって、ポート電源をサスペンドおよびオフにします。
2. PHY は、VBUS_VALID 信号をネゲートすることによって、ポート電源のオフを示します。
3. デバイスは、V_{BUS} 電源がオフになると、SRP を開始するために少なくとも 2 ms 以上 SE0 を検出しなければなりません。
4. SRP を開始するには、デバイスはデータラインプルアップ抵抗を 5～10 ms の間オンにします。OTG_HS コントローラはデータラインのパルシングを検出します。
5. デバイスは、V_{BUS} を A デバイスセッション有効レベル以上（最低 2.0 V）に駆動します（V_{BUS} パルシングに対して）。
   OTG_HS コントローラは SRP を検出すると、アプリケーションに割り込みをかけます。グローバル割り込みステータスレジスタのセッションリクエスト検出ビットがセットされます（OTG_GINTSTS の SRQINT）。
6. アプリケーションは、セッションリクエスト検出割り込みを処理して、ホストポート制御およびステータスレジスタのポートパワービットに書き込むことによって、ポート電源をオンにする必要があります。PHY は、VBUS_VALID 信号をアサートすることによって、ポート電源がオンになったことを示します。
7. USB に電源が投入されると、デバイスが接続され、SRP プロセスが完了します。

B デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって OTG_HS コントローラは SRP を B デバイスとして扱えるようになります。SRP は OTG_HS コントローラがホストからの新しいセッションをリクエストする手段です。

参考資料

図 764. B デバイスの SRP



1. VBUS_VALID = PHY からの V_{BUS} 有効信号
   B_VALID = PHY への B ペリフェラル有効セッション信号
   DISCHRG_VBUS = PHY への放電信号
   SESS_END = PHY へのセッション終了信号
   CHRG_VBUS = PHY への V_{BUS} 充電信号
   DP = データ正ライン
   DM = データ負ライン

以下では、*図 764*に示された信号計算法を記載しています。

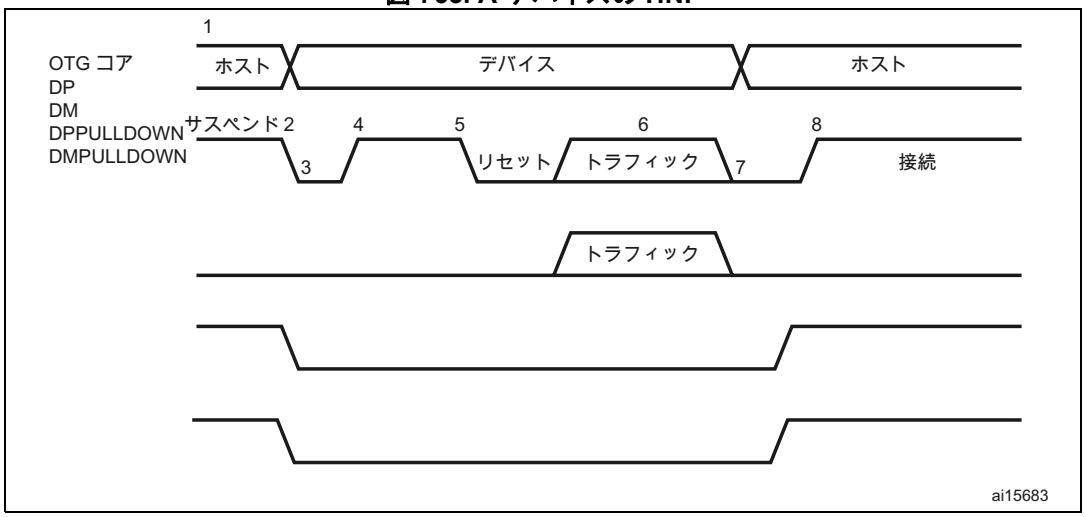
1. 節電のために、ホストはバスがアイドル状態のときはポート電源をサスペンドおよびオフにします。
   OTG_HS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。
   OTG_HS コントローラは、PHY に V_{BUS} を放電するように通知します。
2. PHY はデバイスにセッションの終了を知らせます。これが SRP の初期条件です。OTG_HS コントローラは SRP を初期化する前に 2 ms の SE0 を必要とします。
   USB 1.1 フルスピードシリアルトランシーバの場合、アプリケーションは、BSVLD（OTG_GOTGCTL）がネゲートされた後、V_{BUS} が 0.2 V まで放電するまで待つ必要があります。この放電時間の値は、トランシーバのベンダから入手でき、ベンダごとに異なります。
3. OTG_HS コアは PHY に V_{BUS} の放電を加速させることを通知します。
4. アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエストビットに書き込むことによって、SRP を開始します。OTG_HS コントローラは、データラインパルシングに続いて V_{BUS} パルシングを実行します。
5. ホストは、データラインパルシングまたは V_{BUS} パルシングから SRP を検出して、V_{BUS}をオンにします。PHY は、デバイスに V_{BUS} 電源オンを知らせます。
6. OTG_HS コントローラは、V_{BUS} パルシングを実行します。
   ホストは、V_{BUS}をオンにして、SRP が成功したことを示すことによって、新しいセッションを開始します。OTG_HS コントローラは OTG 割り込みステータスレジスタのセッションリクエスト成功ステータス変化ビットをセットすることによって、アプリケーションに割り込みをかけます。アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエスト成功ビットを読み出します。

7. USB に電源が投入されると OTG_HS コントローラが接続され SRP プロセスが完了します。

A デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を A デバイスから B デバイスに切り替えます。アプリケーションは OTG_HS コントローラが HNP を A デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 765. A デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、*図 765*に示された信号計算法を記載しています。

1. OTG_HS コントローラは Set Featune b_hnp_enable 記述子を B デバイスに送信して、HNP サポートを有効にします。B デバイスの ACK 応答は、B デバイスが HNP をサポートしていることを示します。アプリケーションは OTG 制御およびステータスレジスタのセット HNP イネーブルビットをセットして、B デバイスが HNP をサポートしていることを OTG_HS コントローラに示す必要があります。
2. アプリケーションがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによって、サスペンドします。
3. B デバイスが USB サスペンドを検出すると、切断して、HNP の初期条件を示します。B デバイスは、ホストの役割に切り替える必要があるときにのみ、HNP を開始します。そうでないときには、バスはサスペンド状態を継続します。

   OTG_HS コントローラは OTG 割り込みステータスレジスタのホストネゴシエーション検出割り込みをセットして HNP の開始を示します。

   OTG_HS コントローラは PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割を示します。PHY は、OTG_DP プルアップ抵抗を有効にして、B デバイスの接続を示します。

   アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、デバイスモードの動作を確認する必要があります。
4. B デバイスは接続を検出し USB リセットを発行し、データトラフィックのために OTG_HS コントローラをエニュメレーションします。
5. B デバイスはホストの役割を継続し、トラフィックを開始し、終了時にはバスをサスペンドします。

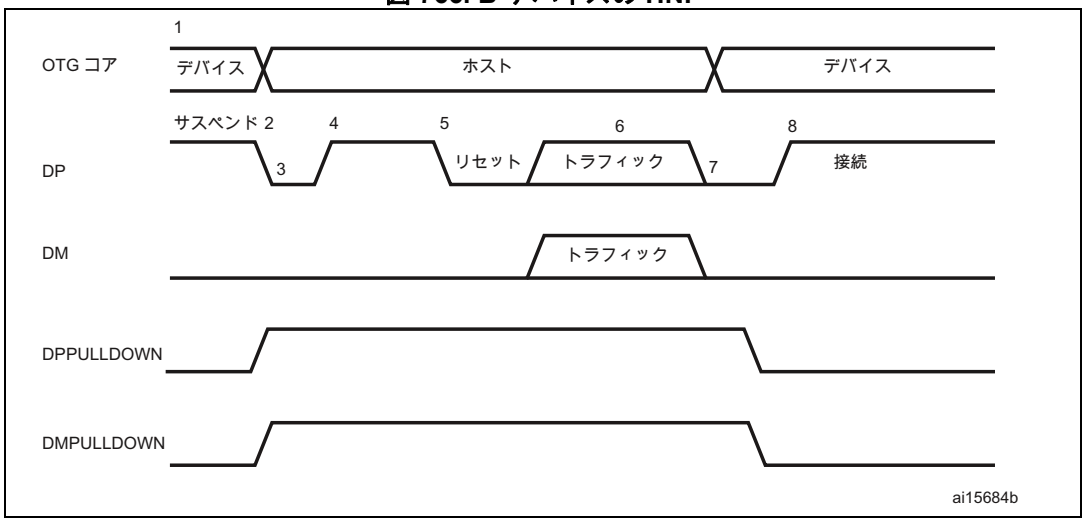
   OTG_HS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。

6. ネゴシエートモードでは OTG_HS コントローラは、サスペンドを検出して切断し、ホストの役割に戻ります。OTG_HS コントローラは、ホストの役割の前提を示すために PHY の DP プルダウンと DM プルダウンをアサートします。
7. OTG_HS コントローラは OTG 割り込みステータスレジスタのコネクタ ID ステータス変更割り込みをセットします。アプリケーションは、OTG 制御およびステータスレジスタのコネクタ ID ステータスを読み出して、OTG_HS コントローラの動作が A デバイスであることを確認する必要があります。これは、HNP の完了をアプリケーションに示します。アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
8. B デバイスが接続され、HNP プロセスが完了します。

B デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を B デバイスから A デバイスに切り替えます。アプリケーションは OTG_HS コントローラが HNP を B デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 766. B デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、*図 766*に示された信号計算法を記載しています。

1. A デバイスは、SetFeature b_hnp_enable 記述子を送信して、HNP サポートを有効にします。OTG_HS コントローラの ACK 応答は、それが HNP をサポートしていることを示します。アプリケーションは、OTG 制御およびステータスレジスタのデバイス HNP イネーブルビットをセットして、HNP をサポートしていることを示す必要があります。

   アプリケーションは、OTG 制御およびステータスレジスタの HNP リクエストビットをセットして、HNP を開始することを OTG_HS コントローラに示す必要があります。
2. A デバイスがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってサスペンドします。

   OTG_HS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。

   OTG_HS コントローラは切断され、A デバイスはバス上で SE0 を検出して HNP を示します。OTG_HS コントローラは、ホストの役割の前提を示すために PHY の DP プルダウンと DP プルダウンをアサートします。

A デバイスは、SE0 を検出してから 3 ms 以内に OTG_DP プルアップ抵抗を有効にすることによって応答します。OTG_HS コントローラはこれを接続として検出します。

OTG_HS コントローラは OTG 割り込みステータスレジスタのホストネゴシエーション成功ステータス変更割り込みをセットして、HNP ステータスを示します。アプリケーションは、OTG 制御およびステータスレジスタのホストネゴシエーション成功ビットを読み出して、ホストネゴシエーションの成功を確認する必要があります。アプリケーションは、コア割り込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。

3. アプリケーションはリセットビット（OTG_HPRT の PRST）をセットし、OTG_HS コントローラは USB リセットを発行し、データトラフィックのために A デバイスをエニュメレーションします。
4. OTG_HS コントローラは、トラフィックを開始するというホストの役割を継続し、これが終了すると、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってバスをサスペンドします。
5. ネゴシエートモードでは、A デバイスは、サスペンドを検出すると、接続を切断し、ホストの役割に戻ります。OTG_HS コントローラは PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割の前提を示します。
6. アプリケーションは、コア割り込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
7. OTG_HS コントローラが接続され HNP プロセスが完了します。

参考資料

58 Ethernet（ETH）：メディアアクセス制御（MAC）、DMA コントローラ付き

58.1 Ethernet の概要

Portions Copyright (c) 2004, 2005 Synopsys, Inc. All rights reserved.掲載許可取得済み。

Ethernet ペリフェラルでは、デバイスは、IEEE 802.3-2002 標準に準拠した Ethernet 経由でデータを送受信できます。

Ethernetは、さまざまなアプリケーションやユーザのニーズを満たす、設定可能で柔軟性の高いペリフェラルを提供します。イーサネットは、外部物理層（PHY）に対して 2 つの業界標準インタフェース、すなわち IEEE 802.3 で定義されているデフォルトのメディア独立インタフェース（MII）と、減少メディア独立インタフェース（RMII）をサポートしています。スイッチ、ネットワークインタフェースカードなど、多くのアプリケーションで使用できます。

IEEE 802.3 で定義されたデフォルトのインタフェースに加えて、Ethernet ペリフェラルは PHY に対する複数の業界標準インタフェースをサポートします。以下の標準に準拠しています。

- IEEE 802.3-2008（Ethernet MAC）、メディア独立インタフェース（MII）
- IEEE 1588-2008（高精度ネットワーク化クロック同期（PTP））
- IEEE 802.1AS-2011 および 802.1-Qav-2009（オーディオビデオ（AV）トラフィック）
- IEEE 802.3az-2010（省電力型 Ethernet（EEE））
- AMBA 2.0（AHB マスタおよび AHB スレーブポート）
- RMII 仕様バージョン 1.2（RMII コンソーシアム）

58.2 Ethernet の主な機能

Ethernet ペリフェラルは、ダイレクトメモリインタフェースの専用 DMA、複数のフォーマットをサポートするメディアアクセスコントローラ（MAC）および PHY インタフェースブロックを内蔵しています。

58.2.1 MAC コア機能

インタフェース

- アプリケーションへの個別の送信、受信、および制御インタフェース
- アプリケーション側での 32-ビットデータ転送インタフェース
- 以下の PHY インタフェースでの 10、100 のデータ転送速度
  - IEEE 802.3 準拠の MII（デフォルト）インタフェースで、外部Fast Ethernet PHY と通信
  - RMII インタフェースで、外部Fast Ethernet PHY と通信
- PHY デバイスの設定と管理のための MDIO（Clause 22 および Clause 45）マスタインタフェース
- RMON カウンタ（RFC2819/RFC2665）による強制ネットワーク統計をサポート

参考資料

主な動作

- 全二重と半二重の両方の動作をサポート
  - 半二重動作の CSMA/CD プロトコルをサポート
  - 全二重動作の IEEE 802.3x フロー制御をサポート
  - 全二重動作では、受信したポーズ制御フレームをユーザアプリケーションに転送するオプションを備えています。
  - 半二重動作でのバックプレッシャ
  - 全二重動作では、フロー制御入力のネゲート時に、ゼロクォンタムポーズフレームを自動送信します。
- 全二重フロー制御動作（IEEE 802.3x ポーズパケットおよび優先フロー制御）
- 送信モードでのプリアンブルとフレーム開始データ（SFD）の挿入、および受信パスでの削除
- フレームごとに制御可能な CRC とパッドの自動生成。
- 標準または最大 16 KB までのジャンボ Ethernet パケットをサポートするプログラム可能なパケット長
- プログラム可能なパケット間隔
- 受信パケットのレイヤ 3／レイヤ 4 チェックサムオフロード
- ストアアンドフォワードモードで送信されたフレームへの IPv4 ヘッダチェックサムと TCP、UDP、または ICMP チェックサムの計算と挿入
- 2 セットの FIFO：閾値を設定できる 2048 バイト の送信 FIFO と、閾値を設定できる 2048バイト の受信 FIFO
- MAC への送信に対するストアアンドフォワードメカニズムまたは閾値モード（カットスルー）
- プログラム可能な Rx キュー閾値（またはカットスルー）モード
- デバッグ用 MII での Tx から Rx への内部ループバック

VLAN 管理

- ソースアドレスフィールド挿入または書き換え、および VLAN 挿入、書き換え、およびパケット単位やスタティックグローバル制御のある送信されたパケットでの削除
- 最大 2 つの VLAN タグの挿入、書き換え、または削除
- IEEE 802.1Q VLAN タグ検知および受信パケットで VLAN タグを削除できる機能
- 最大 2 つの VLAN タグのストリッピングとステータスでのタグの提示

パケットフィルタ

- 柔軟なアドレスフィルタモード：
  - 各バイトのマスク付きの追加された 3 つの 48 ビット完全デスティネーションアドレス（DA）フィルタ
  - 各バイトのマスク付きの追加された 3 つの 48 ビットソースアドレス（SA）比較チェック
  - マルチキャストおよびユニキャスト（DA）アドレス用 64 ビットハッシュフィルタ
  - すべてのマルチキャストアドレスパケットを通過させるオプション
  - ネットワーク監視にフィルタリングを使用せずにすべてのパケットを通過させるプロミスキャスモード
  - ステータスレポート付きで、すべての着信パケットを（フィルタに従って）通過
- 追加パケットフィルタ：
  - VLAN タグベース：外部または内部 VLAN タグに基づいたフィルタによる完全一致およびハッシュベースフィルタ
  - レイヤ 3 およびレイヤ 4 ベース：IPv4 または IPv6 での TCP または UDP

参考資料

IEEE 1588-2008/PTPv2

- IEEE 1588-2002 および IEEE 1588-2008 仕様に示された Ethernet パケットのタイムスタンプ（PTP パケットの Tx または Rx ステータスで示される 64 ビットタイムスタンプ）。Tx 方向でワンステップおよびツーステップのタイムスタンプ両方をサポート
- Pulse-Per-Second（PPS）出力信号（ptp_pps_o）を制御する柔軟性

低電力モード

- 標準 IEEE 802.3az-2010（MII PHY での省電力型 Ethernet）
- リモートウェイクアップパケットおよび AMD マジックパケットを検出するモジュール

58.2.2 DMA の機能

DMA ブロックは、ペリフェラルとシステムメモリでデータを交換します。DMA 転送はソフトウェアディスクリプタ構造によって駆動されます。アプリケーションはレジスタのセット（*セクション 58.11.2 : Ethernet DMA レジスタ*を参照）を使用して DMA の動作を制御できます。DMA ブロックは、次の機能をサポートしています。

- 32 ビットデータ転送
- 送信パスと受信パスでの個別の DMA
- パケットデリミタ付きパケット指向 DMA 転送の最適化
- データバッファサポートのためのバイト整列アドレッシング。
- デュアルバッファ（リング）ディスクリプタのサポート
- 最小の CPU 介入で大きなデータブロックの転送を可能にするディスクリプタアーキテクチャ（各ディスクリプタは最大 32 KB のデータを転送できます）
- 正常動作と転送エラーについての包括的なステータスレポート
- 最適なホストバス利用のために、Tx DMA および Rx DMA エンジンのバースト長を個別にプログラム可能
- さまざまな動作条件に対応するプログラム可能な割り込みオプション。
- パケットごとの送信または受信完了割り込み制御
- 受信エンジンと送信エンジン間のラウンドロビン方式または固定優先順位のアービトレーション
- STARTモードおよび STOPモード
- ホスト制御（AHB）アクセス用およびホストデータインタフェース用の個別のポート
- Tx DMA チャネルTCP セグメンテーションオフロード（TSO）機能有効

58.2.3 バスインタフェースの機能

AHB マスタインタフェース

AHB マスタインタフェースの機能を次に示します。

- AHB を通じたアプリケーションとのインタフェース
- リトルエンディアンモード
- AHB マスタポートでの 32 ビットデータ
- スプリット、再試行、エラーの AHB レスポンス
- AHB 1K 境界バーストスプリット
- ソフトウェア選択による AHB バーストのタイプ（固定バースト、可変バースト、または両方の混合）

AHB マスタインタフェースは次のものを生成しません。

- ラップバースト
- 転送のロックまたは保護

AHB スレーブインタフェース

AHB スレーブインタフェースは、次の機能をサポートしています。

- AHB を通じたアプリケーションとのインタフェース
- リトルエンディアンモード
- CSR アクセス用 AHB スレーブインタフェース（32 ビット）
- すべての AHB バーストタイプ

AHB スレーブインタフェースは次のレスポンスを生成しません。

- スプリット
- 再試行
- エラー

58.3 Ethernet ピンおよび内部信号

表 490 では、パッケージのピンまたはボールに接続される RCC 入力および出力信号を一覧化しており、*表 491* では内部 RCC 信号を示しています。

表 490. Ethernet ペリフェラルピン

信号名	AF11
PA0	ETH_MII_CRS
PA1	ETH_MII_RX_CLK/ETH_RMII_REF_CLK
PA2	ETH_MDIO
PA3	ETH_MII_COL
PA7	ETH_MII_RX_DV/ETH_RMII_CRS_DV
PA9	ETH_TX_ER
PB0	ETH_MII_RXD2
PB1	ETH_MII_RXD3
PB2	ETH_TX_ER
PB5	ETH_PPS_OUT
PB8	ETH_MII_TXD3
PB10	ETH_MII_RX_ER
PB11	ETH_MII_TX_EN/ETH_RMII_TX_EN
PB12	ETH_MII_TXD0/ETH_RMII_TXD0
PB13	ETH_MII_TXD1/ETH_RMII_TXD1
PC1	ETH_MDC
PC2	ETH_MII_TXD2
PC3	ETH_MII_TX_CLK
PC4	ETH_MII_RXD0/ETH_RMII_RXD0
PC5	ETH_MII_RXD1/ETH_RMII_RXD1

表 490. Ethernet ペリフェラルピン（続き）

信号名	AF11
PE2	ETH_MII_TXD3
PG8	ETH_PPS_OUT
PG11	ETH_MII_TX_EN/ETH_RMII_TX_EN
PG12	ETH_MII_TXD1/ETH_RMII_TXD1
PG13	ETH_MII_TXD0/ETH_RMII_TXD0
PG14	ETH_MII_TXD1/ETH_RMII_TXD1
PH2	ETH_MII_CRS
PH3	ETH_MII_COL
PH6	ETH_MII_RXD2
PH7	ETH_MII_RXD3
PI10	ETH_MII_RX_ER
PI12	ETH_TX_ER

表 491. Ethernet 内部入力／出力信号

信号名	信号タイプ	説明
hclk_i	デジタル入力	AHB クロック
eth_sbd_intr_it	デジタル出力	メイン Ethernet 割り込み
lpi_intr_o	デジタル出力	トランスミッタまたはレシーバが LPI 状態の移行時または終了時に生成されるサイドバンド信号
pmt_intr_o	デジタル出力	有効なリモートウェイクアップパケットが受信されたときに生成されるサイドバンド信号
eth_mii_tx_clk	デジタル入力	MII Tx カーネルクロック
eth_mii_rx_clk	デジタル入力	MII Rx カーネルクロック
eth_rmii_ref_clk	デジタル入力	RMII リファレンスカーネルクロック

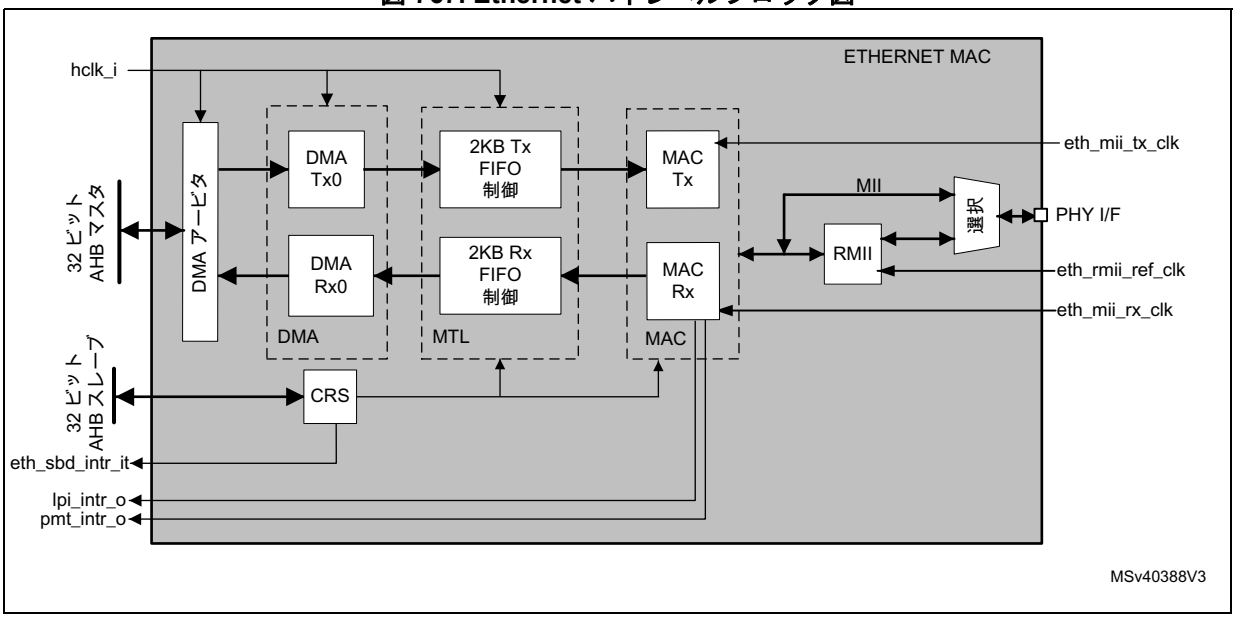
58.4 Ethernet アーキテクチャ

Ethernet ペリフェラルは 4 つの主な機能モジュールで構成されます。

- **制御およびステータスレジスタモジュール（CSR）**は、AHB 32 ビットスレーブインタフェースを通じてレジスタアクセスを制御します。
- **ダイレクトメモリアクセスインタフェース（DMA）**
  これは、受信用に 1 つ、送信用に 1 つの物理チャネルを持つ論理 DMA モジュールです。AMBA AHB 32 ビットマスタインタフェースを通じて MAC とシステムメモリでのデータ転送を制御します。
- **メディアアクセス制御モジュール（MAC）**は、Ethernet プロトコルを実装する役割を持っています。
- **MAC トランザクションレイヤ（MTL）**は、アプリケーションと MAC のデータフローを制御する役割を持っています。

RMII PHY メディア独立インタフェースをサポートするには、プロトコル適応モジュールが追加されます。

図 767. Ethernet ハイレベルブロック図



1. 内部信号の定義については、*表 491* を参照してください。

58.4.1 DMA コントローラ

DMA には、独立した送信（Tx）エンジンと受信（Rx）エンジンがあります。Tx エンジンは、データをシステムメモリから MAC トランザクションレイヤ（MTL）に転送し、Rx エンジンは、データをデバイスポート（PHY）からシステムメモリに転送します。

コントローラは、ディスクリプタを使用して、データを転送元から転送先に最小限のアプリケーション CPU 介入によって効率的に移動します。DMA は、Ethernet のパケットなど、パケット指向のデータ転送を行うように設計されています。コントローラは、パケット送受信完了やその他の正常またはエラー条件発生の状況でアプリケーション CPU に割り込むようにプログラムすることができます。

DMA データ構造

DMA とアプリケーションは、次の 2 つのデータ構造を使用して通信します。

- 制御およびステータスレジスタ（CSR）
- ディスクリプタリストおよびデータバッファ

DMA は、MAC によって受信したデータパケットをシステムメモリ内の Rx バッファに、Tx データパケットをシステムメモリ内の Tx バッファから転送します。システムメモリ内に存在するディスクリプタは、これらのバッファに対するポインタを含みます。

各リストのベースアドレスは、それぞれ Tx および Rx レジスタに書き込まれます（*チャネル Tx ディスクリプタリストアドレスレジスタ（ETH_DMACTxDLAR）*および*チャネル Rx ディスクリプタリストアドレスレジスタ（ETH_DMACRxDLAR）*）。

ディスクリプタリストは次にリンクされ、次のディスクリプタは常に現在のものに対する固定オフセットで考慮されます。リスト内のディスクリプタの数は、それぞれ Tx/Rx（*チャネル Tx ディスクリプタリング長レジスタ（ETH_DMACTxRLR）*および*チャネル Rx ディスクリプタリング長レジスタ（ETH_DMACRxRLR）*）でプログラムされます。

DMA がリストの最終ディスクリプタを処理すると、自動的にリストアドレスレジスタのディスクリプタにジャンプし、ディスクリプタリングを作成します。ディスクリプタリストは、アプリケーションの物理メモリアドレス空間にあります。各ディスクリプタは、最大 2 つのバッファを指し示すことができます。これにより、メモリ内で隣接したバッファでなくても、物理的にアドレス指定された 2 つのバッファを使用できます。

データバッファは、アプリケーション物理メモリ空間に置かれ、パケット全体または一部で構成されますが、1 パケットの長さを超えることはできません。バッファは、データのみを含みます。バッファのステータスは、ディスクリプタ内に保存されます。データ連結とは、複数のデータバッファにまたがるパケットを指します。ただし、1 つのディスクリプタが複数のパケットにまたがることはできません。DMA は、EOP が検出されると、次のパケットのデータバッファにスキップします。

ディスクリプタについては、*セクション 58.10：ディスクリプタ*で指定されています。

DMA アービトレーション

DMA モジュールには、AHB マスタインタフェースの Tx チャネルアクセスと Rx チャネルアクセスの間のアービトレーションを行うアービタが組み込まれています。次の 2 種類のアービトレーションをサポートしており、*DMA モードレジスタ（ETH_DMAMR）*で選択できます。

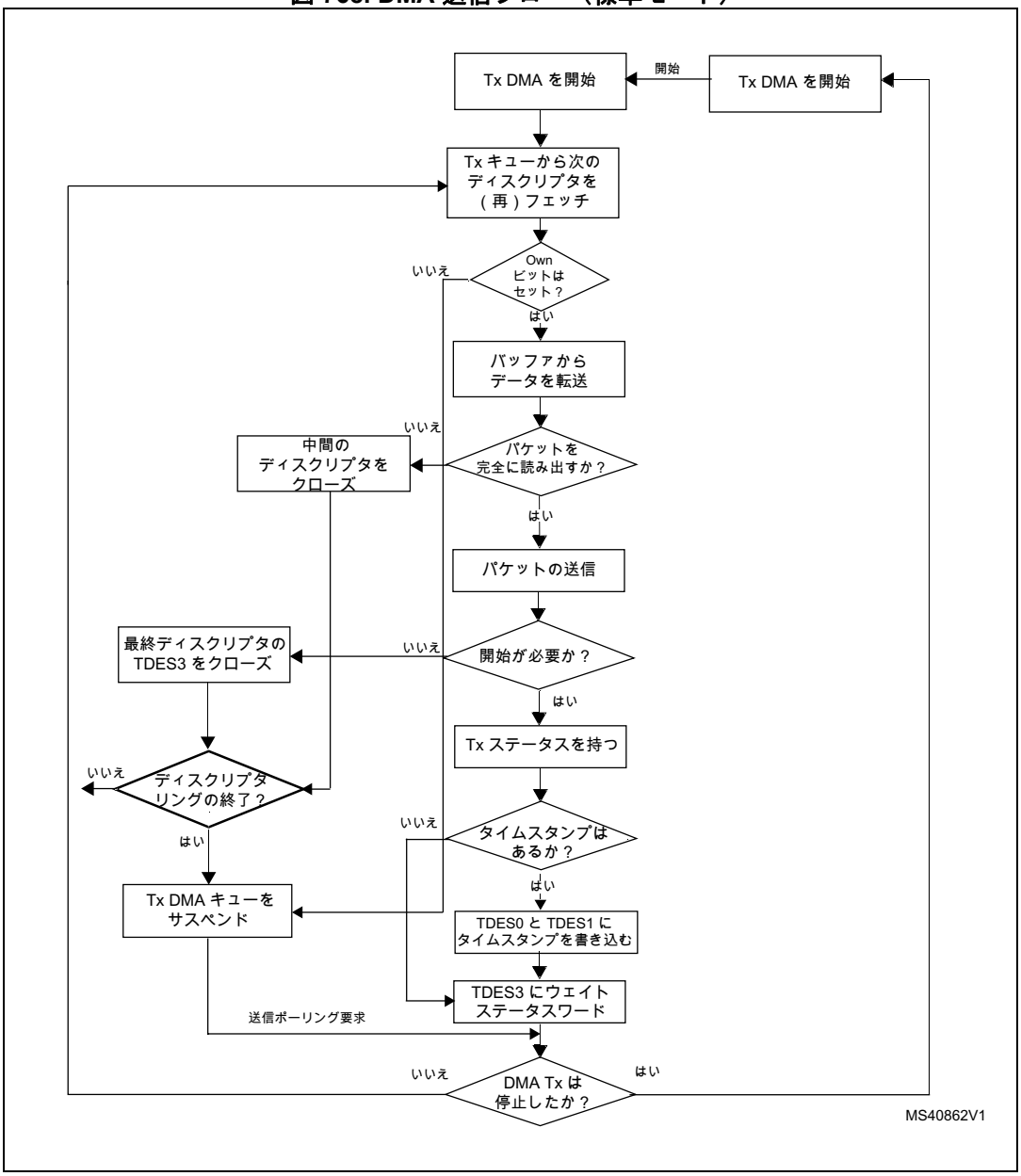
- ラウンドロビン方式アービトレーション：ETH_DMAMR のビット [14:12] でセットされた比率でアービタが Rx と Tx のデータバスを割り当てます。
- 固定優先順位アービトレーション：デフォルトで、Rx DMA は、データアクセスの際、常に Tx DMA より優先されます。ETH_DMAMR レジスタのビット 11 をセットすると、Tx DMA が優先されます。

デフォルトモードでの DMA 送信

デフォルトモードの Tx DMA エンジンは、次のように動作します。

1. アプリケーションは、送信ディスクリプタ（TDES0 ～ TDES3）をセットアップし、対応するデータバッファを Ethernet パケットデータでセットアップした後、Own ビット（TDES0[31]）をセットします。
2. アプリケーションは、送信チャネルのディスクリプタのテイルポインタオフセット値をシフトします。
3. DMA は、アプリケーションメモリからディスクリプタをフェッチします。
4. DMA が次の条件のうち 1 つを検出した場合、そのチャネルからの送信はサスペンドされ、対応する DMA チャネルステータスレジスタのビット 2 および 16 がセットされ、Tx エンジンは手順 11 に進みます。
   – ディスクリプタがアプリケーションによって所有されているというフラグが立てられた場合（TDES3[31] = ‘0’）
   – ディスクリプタのテイルポインタがリングディスクリプタリストモードの現在のディスクリプタと同じ場合
   – エラー条件が発生した場合
5. 取得したディスクリプタに DMA の所有であるというフラグが立てられていた場合（TDES3[31] = ‘1’）、DMA は、取得したディスクリプタから送信データバッファアドレスをデコードします。
6. DMA は、システムメモリから送信データをフェッチして、送信のために MTL にデータを転送します。
7. Ethernet パケットが複数のディスクリプタのデータバッファ経由で格納された場合、DMA は、中間のディスクリプタをクローズして、次のディスクリプタをフェッチします。Ethernet パケットデータの終わりが MTL に転送されるまで、手順 3～7 が繰り返されます。
8. パケット送信が完了すると、IEEE 1588 タイムスタンプ機能がそのパケットに対して有効な場合（Tx ステータスに示される）、MTL から取得したタイムスタンプの値が EOP バッファを含む Tx ディスクリプタ（TDES0 および TDES1）に書き込まれます。ステータス情報はこの Tx ディスクリプタ（TDES3）に書き込まれます。Own ビットがこの手順中にクリアされるため、アプリケーションがこのディスクリプタを所有するようになります。タイムスタンプ機能がこのパケットに対して無効である場合、DMA は TDES0 と TDES1 の内容を変更しません。
9. 最終ディスクリプタで完了時割り込み（TDES2[31]）がセットされているパケットの送信を完了した後、*チャネルステータスレジスタ(ETH_DMACSR)*のビット 0 がセットされます。DMA エンジンは手順 3 に戻ります。
10. サスペンド状態では、DMA がディスクリプタの再取得を試みます（その結果、手順 3 に戻ります）。送信ポーリング要求の受け取りと、アンダーフロー割り込みステータスビットがクリアされると、任意の値を*チャネル Tx ディスクリプタテイルポインタレジスタ（ETH_DMACTxDTPR）*に書き込むことでポーリング要求コマンドがトリガされます。アプリケーションが対応する DMA チャネルの送信制御レジスタのビット 0 をクリアすることで DMA を停止させた場合、DMA は停止状態に入ります。

図 768. DMA 送信フロー（標準モード）



OSP（2 番目のパケットに対する動作）モードでの DMA 送信

実行状態では、*チャネル 送信制御レジスタ（ETH_DMACTxCR）*でビット 4 がセットされている場合、送信プロセスで最初のパケットのステータスディスクリプタをクローズせずに 2 つのパケットを同時に取得できます。送信プロセスが最初のパケットの転送を完了すると、ただちに 2 番目のパケットの送信ディスクリプタリストにポーリングします。2 番目のパケットが有効な場合、送信プロセスは最初のパケットのステータス情報を書き込む前に、このパケットを転送します。

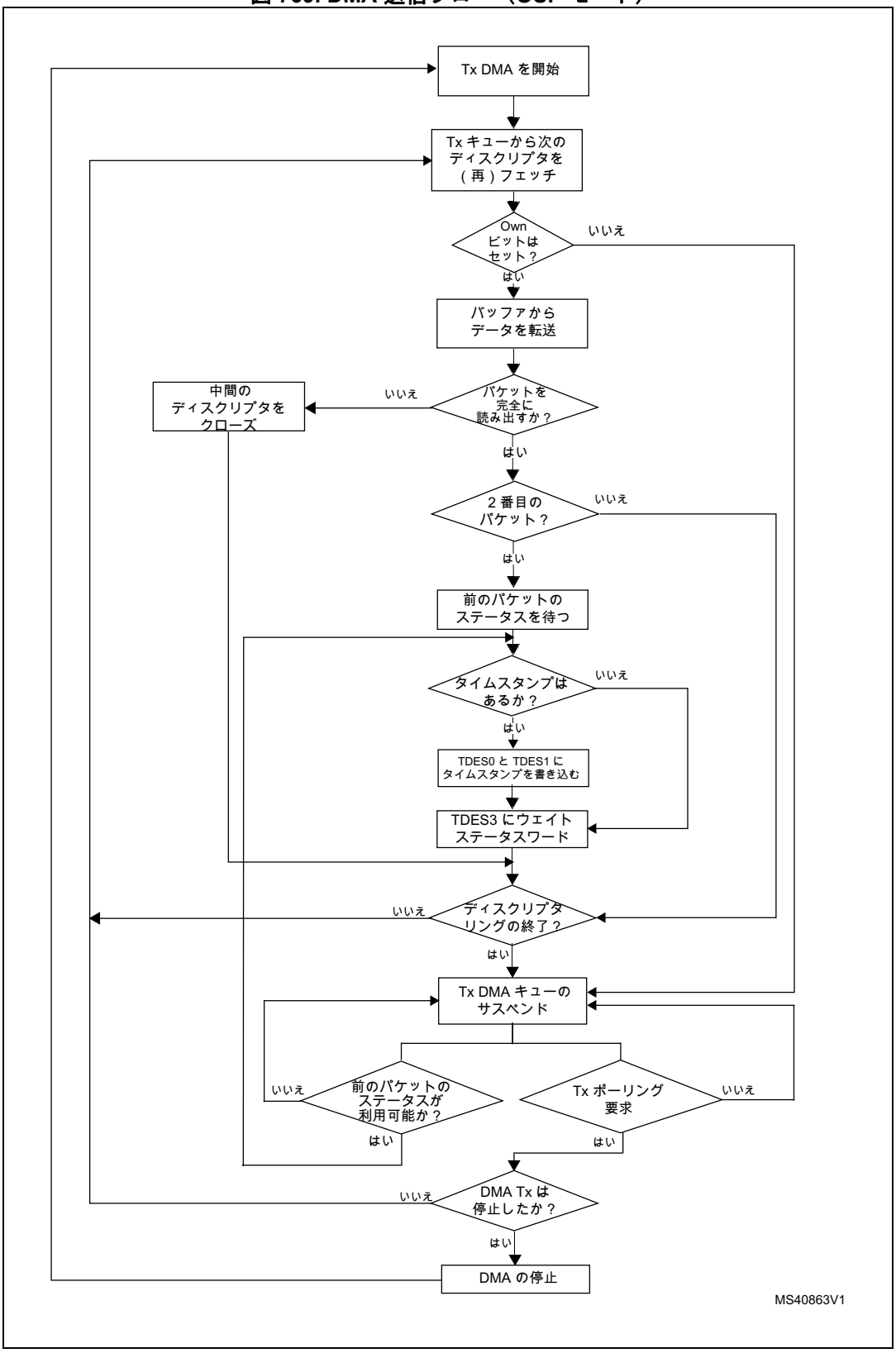
OSP モードでは、実行状態の DMA 送信は、以下のシーケンスに示すとおりに動作します。

1. DMA は、デフォルトモードで DMA 送信シーケンスの手順 1～7 を実行します（*セクション：デフォルトモードでの DMA 送信*を参照）。
2. DMA は、前のパケットの最終ディスクリプタをクローズせずに、次のディスクリプタをフェッチします。
3. DMA が取得したディスクリプタを所有している場合、DMA は、このディスクリプタ内の送信バッファアドレスをデコードします。DMA がディスクリプタを所有していない場合、DMA は、サスペンドモードに入り、手順 7 にジャンプします。
4. DMA は、システムメモリから送信パケットをフェッチして、EOP データが転送されるまで、MTL にパケットを転送し、このパケットが複数のディスクリプタに分割されている場合は、中間のディスクリプタをクローズします。
5. DMA は、前のパケットのパケット送信ステータスとタイムスタンプを待ちます。ステータスが入手可能なときには、DMA は、タイムスタンプがキャプチャされていた場合は（ステータスビットで示されます）、タイムスタンプを TDES0 と TDES1 に書き込みます。DMA は、Own ビットがクリアされたステータスを、対応する TDES3 に書き込み、ディスクリプタをクローズします。タイムスタンプ機能が前のパケットに対して有効でなかった場合、DMA は、TDES2 と TDES3 の内容を変更しません。
6. 送信割り込みがセットされます（有効時）。DMA は、次のディスクリプタをフェッチしてから、手順 3 へ進みます（ステータスが正常な場合）。前の送信ステータスがアンダーフローエラーを示している場合、DMA はサスペンドモードに移行します（手順 7）。
7. サスペンドモードでは、ペンディングステータスおよびタイムスタンプが MTL から受信される場合、DMA は次の操作を実行します。
   a) DMA は、TDES2 と TDES3 にタイムスタンプ（現在のパケットに対して有効な場合）を書き込みます。
   b) DMA は対応する TDES3 にステータスを書き込みます。
   c) DMA は、該当する割り込みをセットして、サスペンドモードに戻ります。

   ペンディング中のステータスがなく、アプリケーションが対応する DMA チャネルの送信制御レジスタのビット 0 をクリアすることで DMA を停止させた場合、DMA は停止状態に入ります。
8. DMA は、対応するチャネルの送信ディスクリプタのテイルポインタレジスタで送信ポーリング要求を受け取った後にのみ、サスペンドモードから抜けて、実行状態に入ることができます（ペンディングステータスに応じて、手順 1 または手順 2 へ移行します）。

OSP モードでの基本的な DMA 送信フローの説明については、*図 770：受信 DMA フロー*を参照してください。

図 769. DMA 送信フロー（OSP モード）



参考資料

DMA 受信

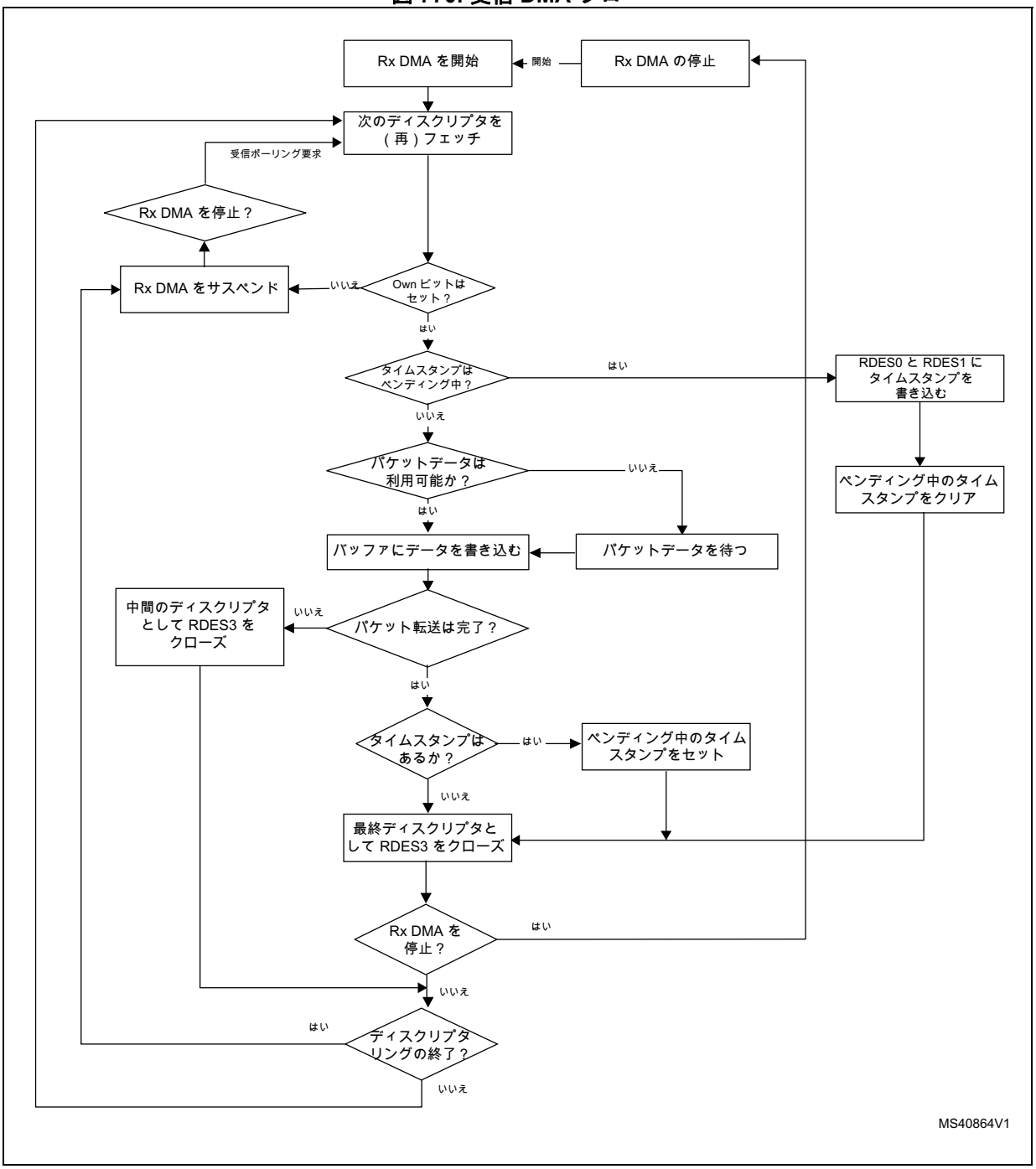
受信パスで、DMA は MTL 受信キューからパケットを読み出し、対応する DMA チャネルのパケットデータバッファに書き込みます。

Rx DMA エンジンの受信シーケンスは、次のとおりです（*図 770：受信 DMA フロー*も参照）。

1. アプリケーションは、Rx ディスクリプタ（RDES0-RDES3）と Own ビット（RDES3[31]）をセットアップします。アプリケーションは、対応する DMA チャネルの受信ディスクリプタのテイルポインタレジスタに正しい値をセットする必要があります。
2. *チャネル 受信制御レジスタ (ETH_DMACRxCR)* のビット 0 がセットされると、DMA は実行状態に入ります。DMA は Rx 現行ディスクリプタおよびディスクリプタのテイルポインタレジスタ値に基づいて、空いているディスクリプタを探します。空いているディスクリプタがない場合、DMA チャネルはサスペンド状態に入り、手順 11 に進みます。
3. DMA は、リングで次に使用できるディスクリプタをフェッチし、取得したディスクリプタから受信データバッファアドレスをデコードします。
4. IEEE 1588 タイムスタンプが有効で、タイムスタンプが前のパケットに使用できる場合、DMA はタイムスタンプ（使用できる場合）を現在のディスクリプタの RDES0 および RDES1 に書き込み、CTXT フィールド（RDES3[30]）をセットします。
5. DMA は着信パケットを処理して、取得したディスクリプタのデータバッファに格納します。
6. 現在のパケット転送をまだ完了していない場合、DMA は現在のディスクリプタを中間ディスクリプタとしてクローズして、手順 10 に進みます。
7. DMA は、MTL から受信フレームのステータスを取り出し、そのステータスワードを、Own ビットがクリアされ、最終ディスクリプタビットがセットされている現在のディスクリプタに書き込みます。
8. DMA はフレーム長を RDES3 に、VLAN タグを RDES0 に書き込みます。また、DMA は MAC 制御フレーム OP コード、OAM 制御フレームコード、拡張ステータス情報（使用できる場合）を、最終ディスクリプタの RDES1 に書き込みます。
9. IEEE 1588 タイムスタンプ機能が有効な場合、DMA はタイムスタンプ（使用できる場合）を保存します。DMA は、現在のパケットの最終ディスクリプタの後、コンテキストディスクリプタを（次に使用できるディスクリプタに）書き込みます。
10. 追加のディスクリプタが Rx DMA ディスクリプタリングで使用できる場合、手順 3 に進みます。そうしないとサスペンド状態（手順 11）に移行します。
11. 受信 DMA は、受信ポーリング要求が提示され、アプリケーションがチャネル受信テイルポインタレジスタをインクリメントすると、サスペンド状態から抜けます。エンジンは手順 2 に進み、次のディスクリプタを再びフェッチします。

参考資料

図 770. 受信 DMA フロー



58.4.2 MTL

MAC トランザクションレイヤ（MTL）は、アプリケーションシステムメモリと MAC 間のパケットを緩和して調整するための FIFO メモリインタフェースを備えています。また、アプリケーションクロックドメインと MAC クロックドメインの間でデータを転送できるようにします。MTL レイヤでは、送信パスと受信パスの 2 つの 32 ビット幅データパスを備えています。

- 送信パス

  アプリケーションまたは内部 DMA は、アプリケーションやシステムメモリから読み出された Ethernet パケットを Tx FIFO にプッシュします。キュー閾値に達した場合（閾値モード）または完了パケットがキューにある場合（ストアアンドフォワードモード）、パケットはポップアウトされ、MAC に転送されます。EOP が転送されると、送信ステータスが MAC から取り出されて、アプリケーションまたは内部 DMA に返送されます。Tx キューサイズは 2048 バイトです。

- 受信パス

  MTL の Rx モジュールは、MAC からパケットを受信して、Rx キューにプッシュします。キューのステータス（フィルレベル）が設定された受信閾値（*Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）* の RTC ビット[1:0]）を超えるか完了パケットを受信すると、そのことがアプリケーションまたは DMA に示されます。また、MTL はキューのフィルレベルを示して、DMA はマスタインタフェースに対して、事前設定されたバースト転送を開始できます。Rx キューサイズは 2048 バイトです。

58.4.3 MAC

MAC は Ethernet プロトコルを処理します。送信モードでは、MTL からデータを受信してから、PHY インタフェースに転送します。受信モードでは、MAC が PHY インタフェースからデータを受信してから、MTL モジュールの Rx FIFO に転送します。

このセクションでは、送受信シーケンスについて簡単に説明します。

MAC 送信

送信シーケンスは、次のとおりです。

1. MTL アプリケーションが SOP（パケットの開始）信号をアサートしてデータをプッシュすると送信が開始されます。
2. SOF 信号が検出されると、MAC はデータを受け入れて、MII への送信を開始します。
3. EOP（パケットの終了）が MAC に転送されると、MAC は次のいずれかを実行します。
   - MAC は通常送信を完了し、MTL に送信ステータスを提供します。
   - 送信中に通常の衝突（半二重モード）が発生した場合、MAC は、再試行ビットをセットして送信ステータスを MTL に提供します。MAC は、次のいずれかに当てはまるまで再試行リクエストを提供します。

     パケットが正常に送信されました。

     再試行リクエストの最大数に達しています。この場合、MAC は過剰衝突送信ステータスのパケット送信をアボートします。MAC は、次の SOP を受信するまですべての追加データを受け入れてドロップします。MAC から（ステータス内で）再試行リクエストが観測されたときには、MTL ブロックは同じパケットを SOF から再送信しなければなりません。
   - 次のいずれかが発生した場合、MAC はパケット送信をアボートします。

     キャリヤなし（半二重モード）

     キャリヤの喪失（半二重モード）

     過剰遅延（半二重モード）

     遅れ衝突（半二重モード）

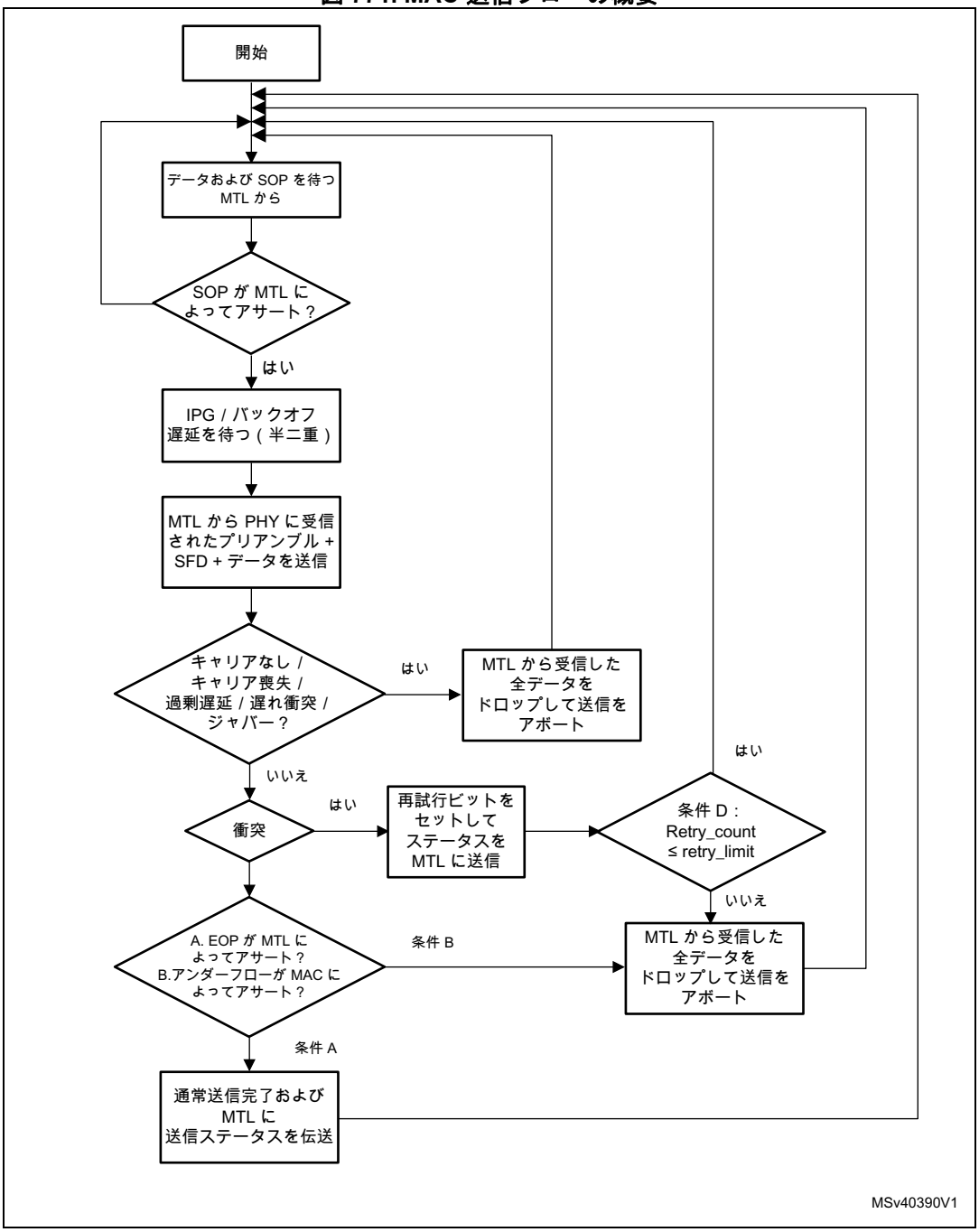
     ジャバー

     MAC は、次の SOP を受信するまですべての追加データを受け入れてドロップします。

4. MTL が送信中にデータを連続的に供給できなかった場合、MAC は、アンダーフローステータスを発行します。MAC は、次の SOP を受信するまですべての追加データを受け入れてドロップします。
5. MTL からのパケットの通常転送中に、MAC が前のパケットの EOP を受信していないのに SOP を受信した場合、SOP を無視して、新しいパケットを前のパケットの続きとみなします。

*図 771 : MAC 送信フローの概要*に、MAC 送信プロセスフローの図を示します。

図 771. MAC 送信フローの概要



MAC 受信

受信動作は、MAC が MII で SFD を検出したときに開始されます。MAC は、パケットの処理を進める前に、プリアンブルと SFD を削除します。ヘッダフィールドはフィルタリングのためにチェックされ、FCSフィールドはパケットのCRCを検証するために使用されます。受信したパケットは、アドレスフィルタが実行されるまで浅いバッファに格納されます。アドレスフィルタでエラーが発生した場合、そのパケットは MAC でドロップされます。

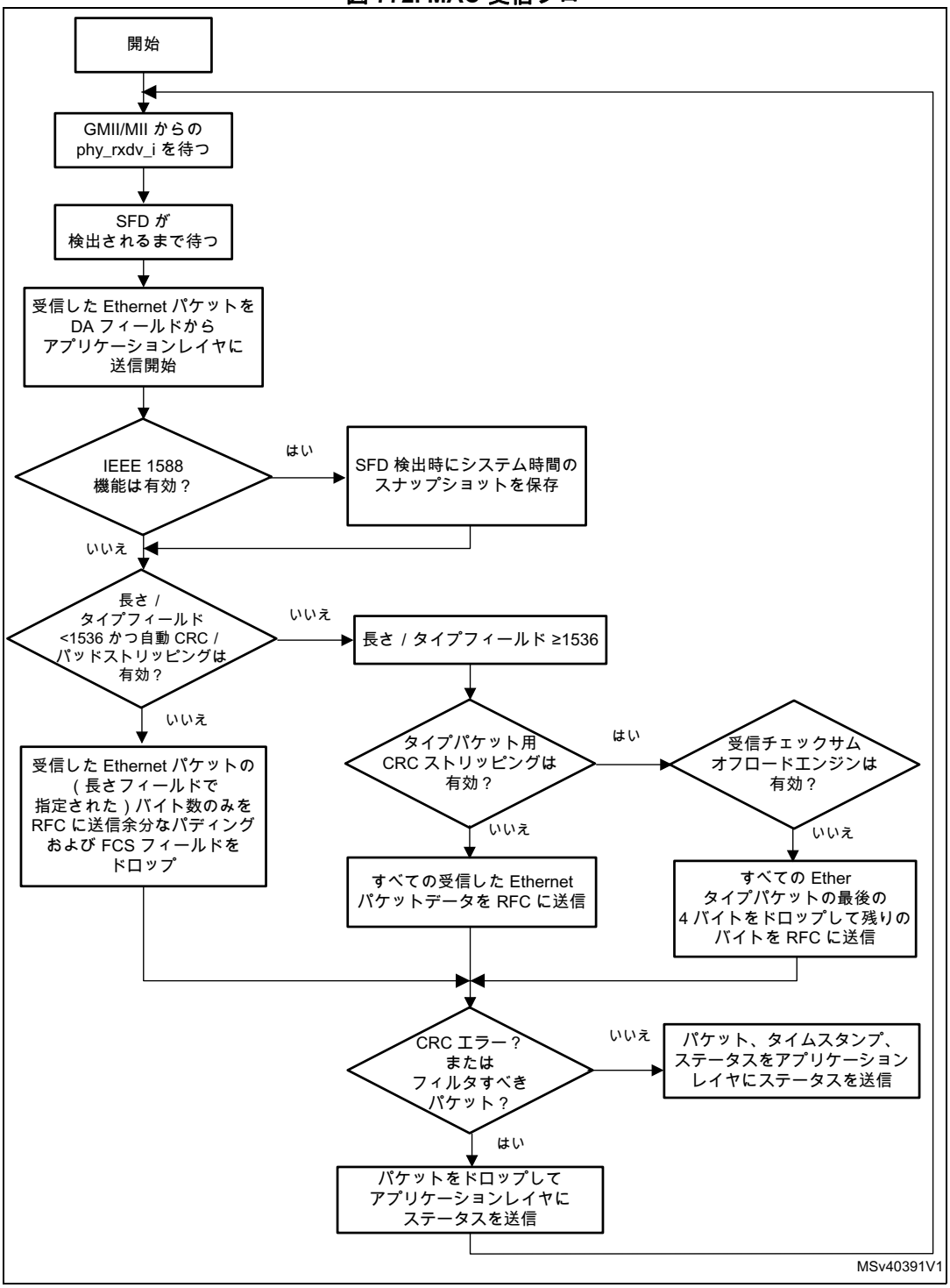
受信シーケンスは、次のとおりです。

1. MII の受信データ有効信号（RxDV）がアクティブになると、受信ステートマシン（RSM）は SFD フィールドの検索を開始します（0xD ニブル（訳注：4ビット）MII モード）。

   ステートマシンは SFD を検出するまで受信パケットをドロップします。
2. SFD が検出されると、ステートマシンは Ethernet パケットのデータの RPC モジュールへの送信を開始し、最初のバイトから始めて、次に SFD（デスティネーションアドレス）という順序で開始します。
3. IEEE 1588 タイムスタンプ機能が有効な場合、任意のパケットの SFD が MII で検出されたとき、MAC はシステム時間のスナップショットを取得します。このパケットが MAC フィルタ中にドロップされない場合、タイムスタンプはアプリケーションに渡されます。MAC は、受信したニブルデータをバイトに変換し、有効なパケットデータを RFC モジュールに転送します。
4. 受信ステートマシンは、受信した Ethernet パケットの長さ／タイプフィールドをデコードします。

   この長さ／タイプフィールドが 1,536 未満の場合と、MAC の自動 CRC／パッドストリッピング（*動作モード設定レジスタ（ETH_MACCR）*のビット 20）がプログラムされていた場合、ステートマシンは、パケットデータを長さ／タイプフィールドで指定されたカウントまで送信し、バイトのドロッピング（FCS フィールドを含む）を開始します。ステートマシンは、長さ／タイプフィールドをデコードして、長さ解釈をチェックします。
5. 長さ／タイプフィールドが 1,536 以上の場合、RPE モジュールは、*動作モード設定レジスタ（ETH_MACCR）*のビット 21 でタイプパケットの CRC ストリッピングを有効にしていなくても、受信したすべての Ethernet パケットデータを RFC モジュールに送信します。ただし、CRC ストリッピングがタイプパケットに対して有効で、受信チェックサムオフロードエンジンに有効ではない場合、MAC は Ether タイプのすべてのパケットの最後の 4 バイトを削除してドロップしてから、パケットをアプリケーションに転送します。
6. デフォルトでは、MAC はウォッチドッグタイマを有効にするようプログラムされており、2,048（ジャンボパケットが有効の場合は 10,240）バイト以上のパケット（DA + SA + LT + データ + パッド + FCS）は RPE モジュールで切り捨てられます。さらに、プログラム可能なウォッチドッグタイマ（*ウォッチドッグタイムアウトレジスタ（ETH_MACWTR）*のビット 16）を使用して、2,048 バイトまたは 10,240 バイトの固定タイムアウトを上書きできます。ウォッチドッグタイマは、*動作モード設定レジスタ（ETH_MACCR）*のビット 19 をプログラムすることで無効にできます。ただし、ウォッチドッグタイマが無効な場合でも、32 KB を超えるパケットは切り捨てられ、ウォッチドッグタイムアウトステータスが与えられます。

図 772. MAC 受信フロー



58.5 Ethernet機能の説明：MAC

58.5.1 ダブル VLAN 処理

Ethernet ペリフェラルは、MAC が最大 2 つの VLAN タグ（内部および外部）を処理できるダブル VLAN（仮想 LAN）タグ機能をサポートします。

MAC は、次の機能をサポートしています。

- 送信パスでの最大 2 つの VLAN タグの挿入、書き換え、または削除
- 受信パスでの 2 つの VLAN タグのうち 1 つに基づいたパケットフィルタおよびストリッピング。受信ステータスの一部としての受信パスでの最大 2 つの VLAN タグのストリッピングと提示

送信パス

*表 492：Tx パスでのダブル VLAN 処理機能*には、送信側で MAC によってサポートされる機能を記載しています。

表 492. Tx パスでのダブル VLAN 処理機能

機能	説明
C-VLAN および S-VLAN タグタイプのサポート	内部または外部 VLAN タグは、C-VLAN および S-VLAN タイプにできます。VLAN タイプは、それぞれ *VLAN 内包レジスタ（ETH_MACVIR）*および*内部 VLAN 内包レジスタ（ETH_MACIVIR）*の CSVL ビットを通じて指定されます。 Ethernet ペリフェラルは、外部および内部 VLAN タグのあらゆるシーケンスの処理をサポートします。ただし、C-VLAN S-VLAN シーケンスはサポートされません。 MACは、アプリケーションによって提供されたパケットが有効なVLANタグタイプのシーケンスを有するか、挿入または置換操作が無効なVLANタグタイプのシーケンスであるかをチェックしません。そのため、アプリケーションは VLAN タグタイプの正しいシーケンスを提供し、送信パケットで VLAN タグタイプのシーケンスが正しくなるように MAC をプログラムする必要があります。アプリケーションは次の内容を確認する必要があります。 – 外部 C-VLAN タグ挿入が有効である場合、内部タグは S-VLAN にしてはなりません。 – 内部 S-VLAN タグ挿入が有効である場合、外部タグは C-VLAN にしてはなりません。 – 外部タグを C-VLAN に書き換える必要がある場合、内部タグは S-VLAN にしてはなりません。 – 内部タグを S-VLAN に書き換える必要がある場合、外部タグは C-VLAN にしてはなりません。
VLAN タグの削除	VLAN タグの削除は、それぞれ *VLAN 内包レジスタ（ETH_MACVIR）*または*内部 VLAN 内包レジスタ（ETH_MACIVIR）*の VLC フィールドを通じて外部または内部タグに対して有効にできます。VLAN 削除が有効である場合、MAC は対応する位置にあるタグを削除します。パケットに 1 つだけタグがある場合は、外部タグとみなされます。内部タグ削除が有効で、パケットに 1 つだけタグがある場合は、MAC はタグを削除しません。
VLAN タグの挿入または書き換え	VLAN タグの挿入または書き換えは、それぞれ *VLAN 内包レジスタ（ETH_MACVIR）*または*内部 VLAN 内包レジスタ（ETH_MACIVIR）*の VLC フィールドを通じて外部または内部タグに対して有効にできます。VLAN タグの挿入または書き換えが有効である場合、前のレジスタの VLTI ビットを使用して、VLAN タグをそのレジスタまたは制御ワードから取得すべきかを決定します。

受信パス

*表 493：Rx パスでのダブル VLAN 処理*には、受信側で MAC によってサポートされる機能および *VLAN タグレジスタ（ETH_MACVTR）*で対応するビットを記載しています。

表 493. Rx パスでのダブル VLAN 処理

機能	説明
外部または内部 VLAN タグベースフィルタ	MAC は、ERIVLT ビットを通じて外部または内部 VLAN タグに基づいてパケットをフィルタできます。
C-VLAN または S-VLAN タグベースフィルタ	MAC は、ERSVLM ビットに基づく C-VLAN または S-VLAN タイプに基づいてパケットをフィルタできます。
外部または内部 VLAN タグストリッピング	MAC は、EVLS および EIVLS ビットに基づく受信フレームから外部および内部 VLAN タグを削除できます。
Rx ステータスでの 16 ビット外部または内部 VLAN タグおよびタイプ	MAC は、それぞれ EVLRXS および EIVLRXS ビットに基づく Rx ステータスで、16 ビット外部および内部 VLAN タグおよびタイプを提供できます。
外部 VLAN タグタイプのチェックの無効化またはスキップ	MAC は、DOVLTC ビットに基づく C-VLAN または S-VLAN に合わせるために、外部 VLAN タグタイプのチェックを無効化またはスキップできます。

58.5.2 ソースアドレスおよび VLAN の挿入、書き換え、または削除

ソースアドレスの挿入または書き換え

ソフトウェアは、SA（ソースアドレス）の挿入または書き換え機能を使用して、MAC に Tx パケットに対して次の操作を行うように命令できます。

- SA フィールドに MAC アドレスレジスタの内容を挿入
- SA フィールドの内容を MAC アドレスレジスタの内容に書き換え

SA の挿入が有効である場合、アプリケーションは MAC に送信されるパケットに SA フィールドがないことを確認する必要があります。MAC は、送信パケットに SA フィールドが存在するかチェックせず、SA フィールドに MAC アドレスレジスタの内容を挿入します。同様に、SA の書き換えが有効である場合、アプリケーションは MAC に送信されるパケットに SA フィールドがないことを確認する必要があります。MAC は、送信パケットのデスティネーションアドレスフィールドに続く 6 バイトを、MAC アドレスレジスタの内容に書き換えます。

SA の挿入または書き換え機能は、すべての送信パケットまたは選択パケットに対して有効にできます。

- すべてのパケットに対する SA の挿入または書き換えの有効化

  すべてのパケットに対してこの機能を有効にするには、*動作モード設定レジスタ (ETH_MACCR)* の SARC フィールドをプログラムします。
- 選択パケットに対する SA の挿入または書き換えの有効化

  選択パケットに対してこの機能を有効にするには、パケットの最初の送信ディスクリプタで次を使用して SA 挿入制御フィールド（送信ディスクリプタワード 3/TDES3 のビット [25:23]、*セクション 58.10.3：送信ディスクリプタ*を参照）をプログラムします。TDES3 のビット 25 がセットされた場合、SA 挿入制御フィールドは MAC アドレス 1 レジスタによる挿入または書き換えを示します。TDES3 のビット 25 がリセットされた場合、MAC アドレス 0 レジスタによる挿入または書き換えを示します。

MAC アドレス 1 レジスタが有効ではない場合、MAC アドレス 0 レジスタは SA 挿入制御フィールドの最上位ビットの値の挿入または書き換えに使用されます。

VLAN の挿入、書き換え、または削除

ソフトウェアは、VLAN の挿入、書き換え、または削除機能を使用して、MAC に Tx パケットに対して次の操作を行うように命令できます。

- VLAN タイプおよび VLAN タグフィールドの削除
- VLAN タイプおよび VLAN タグフィールドの挿入または書き換え

  挿入または書き換えは、*表 494：VLTI ビットに基づいた VLAN の挿入または書き換え*に示された *VLAN 内包レジスタ（ETH_MACVIR）* の VLTI ビットの設定に基づいて実行されます。

表 494. VLTI ビットに基づいた VLAN の挿入または書き換え

状態	説明
VLTI ビットがセットされている場合	MAC は、次の内容を挿入または書き換えします。 VLAN タイプフィールド（*VLAN 内包レジスタ（ETH_MACVIR）* の CSVL ビットによって示される C-VLAN または S-VLAN） パケットの送信コンテキストディスクリプタの VT フィールドがある VLAN タグフィールド
VLTI ビットがリセットされている場合	MAC は、次の内容を挿入または書き換えします。 VLAN タイプフィールド（*VLAN 内包レジスタ（ETH_MACVIR）* の CSVL ビットによって示される C-VLAN または S-VLAN） *VLAN 内包レジスタ（ETH_MACVIR）* の VLT フィールドがある VLAN タグフィールド

VLAN の書き換えまたは削除が有効である場合、MAC は送信パケットの DA および SA フィールドの後に VLAN タイプフィールド（0x8100 または 0x88A8）が存在しているかどうかチェックします。VLAN タイプフィールドが DA および SA フィールドに続く 2 バイトで検出されない場合、書き換えまたは削除の操作は発生しません。ただし、VLAN の挿入が有効である場合、MAC は送信パケットでの VLAN タイプフィールドの存在をチェックせず、ただ VLAN タイプと VLAN タグフィールドを挿入します。

すべての Tx パケットまたは選択パケットに対して VLAN の挿入、書き換えまたは削除を有効にできます。

- すべてのパケットに対してこの機能を有効にするには、*VLAN 内包レジスタ（ETH_MACVIR）* の VLP および VLP フィールドをプログラムします。
- 選択パケットに対してこの機能を有効にするには、TDES2 通常ディスクリプタの VTIR フィールドをプログラムします（*表 514：TDES2 通常ディスクリプタ（読み出しフォーマット）*を参照）。

  さらに、設定に応じて MAC がホストから制御入力を取得するために *VLAN 内包レジスタ（ETH_MACVIR）*（外部 VLAN の場合）および*内部 VLAN 内包レジスタ（ETH_MACIVIR）*（内部 VLAN の場合）で VLP（VLAN 優先度制御）ビットをリセットする必要があります。

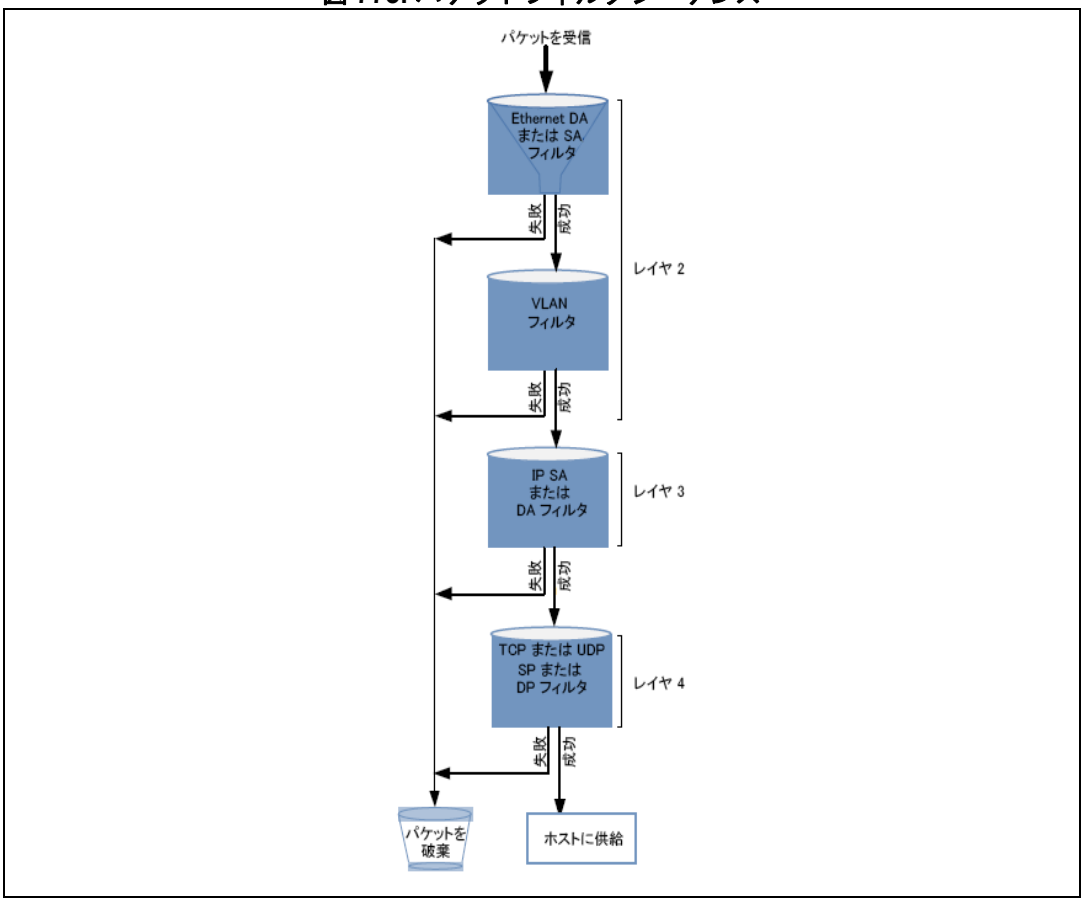
58.5.3 パケットフィルタ

MAC は Rx パケットの次のタイプのフィルタをサポートします。

- *MAC ソースアドレスまたはデスティネーションアドレスフィルタリング*：アドレスフィルタモジュール（AFM）が、各着信パケットのソースアドレスフィールドおよびデスティネーションアドレスフィールドをチェックします。
- *VLAN フィルタ*：MAC が VLAN タグベースおよび VLAN ハッシュフィルタをサポートします。
- *レイヤ 3 およびレイヤ 4 フィルタ*：レイヤ 3 フィルタは、IP ソースアドレスおよびデスティネーションアドレスのフィルタを参照します。レイヤ 4 フィルタは、ソースポートおよびデスティネーションポートのフィルタを参照します。

3 つのフィルタタイプは直列可能です。*図 773* に、Rx パケットのフィルタシーケンスを示します。

図 773. パケットフィルタシーケンス



MAC ソースアドレスまたはデスティネーションアドレスフィルタリング

MAC アドレスフィルタモジュールは、各着信パケットのソースアドレス（SA）フィールドおよびデスティネーションアドレス（DA）フィールドをチェックします。

ユニキャストデスティネーションアドレスフィルタ

MAC は、ユニキャストの完全なフィルタリングのために、4 つの MAC アドレスをサポートします。完全フィルタリングが選択された場合（*パケットフィルタ制御レジスタ（ETH_MACPFR）*の HUC ビットがリセット）、MAC は、受信したユニキャストアドレスの 48 ビットすべてを、プログラムされた MAC アドレスと比較して、一致しているかどうかを確認します。デフォルトの MacAddr0 は常に有効です。

MacAddr1～MacAddr3 のアドレスは、個別のイネーブルビットで選択されます。各バイトは、レジスタの対応するマスクバイト制御ビットをセットすることによって、対応する受信 DA バイトとの比較時にマスクできます。これは、DA のグループアドレスフィルタを可能にします。

ハッシュフィルタリングモード（HUC ビットがセットされているとき）では、MAC は、64 ビットのハッシュテーブルを使用して、ユニキャストアドレスの不完全フィルタリングを実行します。ハッシュフィルタリングの場合、MAC は、受信したデスティネーションアドレスの CRC の上位 6 ビットを使用して、ハッシュテーブルの内容をインデックスします。値 00000 は、選択されたレジスタのビット 0 を選択し、値 11111 は、ハッシュテーブルレジスタのビット 63 を選択します。対応するビット（6 ビット CRC によって示される）が 1 にセットされた場合、ユニキャストパケットはハッシュフィルタを通過したとみなされ、そうでない場合、パケットはハッシュフィルタ通過できなかったとみなされます。

マルチキャストデスティネーションアドレスフィルタ

すべてのマルチキャストパケットを通過する MAC をプログラムするには、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の PM ビットをセットします。PM ビットがリセットされた場合、MAC は、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の HMC ビットに基づいて、マルチキャストアドレスのフィルタリングを実行します。

完全フィルタリングモードでは、マルチキャストアドレスが、プログラムされた MAC デスティネーションアドレスレジスタ（1～31）と比較されます。グループアドレスフィルタリングもサポートされています。

ハッシュフィルタリングモードでは、MAC は、64 ビットのハッシュテーブルを使用して、不完全フィルタリングを実行します。MAC は、受信したマルチキャストアドレスの CRC の上位 6 ビットを使用して、ハッシュテーブルの内容をインデックスします。値 000000 は、選択されたレジスタのビット 0 を選択し、値 111111 は、ハッシュテーブルレジスタのビット 63 を選択します。対応するビットが '1' にセットされた場合は、マルチキャストパケットがハッシュフィルタを通過したとみなされます。そうでない場合は、パケットがハッシュフィルタを通過できなかったとみなされます。

ハッシュまたは完全アドレスフィルタ

DA がハッシュフィルタまたは完全フィルタに一致するときパケットを通過させるために DA フィルタを設定するには、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の HPF ビットおよび対応する HUC または HMC ビットをセットします。これは、ユニキャストパケットとマルチキャストパケットの両方に適用できます。HPF ビットがリセットされた場合は、1 つのフィルタ（ハッシュまたは完全）だけが受信パケットに適用されます。

ブロードキャストアドレスフィルタ

デフォルトでは、MAC は、ブロードキャストパケットをフィルタしません。すべてのブロードキャストパケットを拒否する MAC をプログラムするには、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の DBF ビットをセットします。

ユニキャストソースアドレスフィルタ

MAC は、受信パケットのソースアドレスフィールドに基づいて、完全フィルタリングを実行できます。デフォルトでは、MAC は、SA フィールドを、SA レジスタでプログラムされた値と比較します。対応するレジスタのビット 30 をセットすることによって、比較のために DA の代わりに SA を使用するように MAC アドレスレジスタを設定することができます。

MAC は SA でのグループフィルタリングもサポートします。アドレスの 1 つ以上のバイトをマスクすることで、アドレスのグループをフィルタできます。SAF ビットが*パケットフィルタ制御レジスタ（ETH_MACPFR）*でセットされている場合、MAC は SA フィルタに失敗したパケットをドロップします。そうでない場合、SA フィルタの結果は、受信ステータスワード内のステータスビットとして与えられます（*表 496* を参照）。SAF ビットがセットされた場合、SA フィルタと DA フィルタの結果が AND（論理積）されて、パケットを転送する必要があるかどうかが決まります。つまり、いずれかのフィルタに失敗した場合、パケットはドロップされます。パケットが両方のフィルタを順に通過した場合のみ、パケットはアプリケーションに転送されます。

逆フィルタ

DA および SA フィルタでは、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の DAIF および SAIF ビットをセットすることで、最終出力でフィルター致結果を反転させることができます。DAIF ビットは、ユニキャストとマルチキャストの両方の DA パケットに適用可能です。このモードでは、ユニキャストまたはマルチキャストデスティネーションアドレスフィルタの結果が反転されます。同様に、SAIF ビットがセットされると、ユニキャスト SA フィルタの結果が反転されます。

表 495 および*表 496* に、受信パケットのタイプに基づく DA と SA のフィルタリングを示します。

表 495. デスティネーションアドレスフィルタリング

パケットタイプ	PR	HPF	HUC	DAIF	HMC	PM	DBF	DA フィルタ動作
ブロードキャスト	1	X	X	X	X	X	X	成功
	0	X	X	X	X	X	0	成功
	0	X	X	X	X	X	1	失敗
ユニキャスト	1	X	X	X	X	X	X	すべてのパケットを通過。
	0	X	0	0	X	X	X	完全／グループフィルター致で通過。
	0	X	0	1	X	X	X	完全／グループフィルター致で失敗。
	0	0	1	0	X	X	X	ハッシュフィルター致で通過。
	0	0	1	1	X	X	X	ハッシュフィルター致で失敗。
	0	1	1	0	X	X	X	ハッシュまたは完全／グループフィルター致で通過。
	0	1	1	1	X	X	X	ハッシュまたは完全／グループフィルター致で失敗。
マルチキャスト	1	X	X	X	X	X	X	すべてのパケットを通過。
	X	X	X	X	X	1	X	すべてのパケットを通過。
	0	X	X	0	0	0	X	完全／グループフィルター致で通過、PCF = 0x の場合はポーズパケットをドロップ。
	0	0	X	0	1	0	X	ハッシュフィルター致で通過、PCF = 0x の場合はポーズパケットをドロップ。
	0	1	X	0	1	0	X	ハッシュまたは完全／グループフィルター致で通過、PCF = 0x の場合はポーズパケットをドロップ。
	0	X	X	1	0	0	X	完全／グループフィルター致で失敗、PCF = 0x の場合はポーズパケットをドロップ。
	0	0	X	1	1	0	X	ハッシュフィルター致で失敗、PCF = 0x の場合はポーズパケットをドロップ。
	0	1	X	1	1	0	X	ハッシュまたは完全／グループフィルター致で失敗、PCF = 0x の場合はポーズパケットをドロップ。

表 496. ソースアドレスフィルタリング

パケットタイプ	PR	SAIF	SAF	SA フィルタ動作
ユニキャスト	1	X	X	すべてのパケットを通過。
	0	0	0	完全またはグループフィルター致で通過ステータスだが、失敗したパケットをドロップしない。
	0	1	0	完全またはグループフィルター致で失敗ステータスだが、パケットをドロップしない。
	0	0	1	完全またはグループフィルター致で通過、失敗したパケットをドロップする。
	0	1	1	完全またはグループフィルター致で失敗、失敗したパケットをドロップする。

VLAN フィルタ

MAC は、完全およびハッシュ VLAN フィルタをサポートします。詳細なプログラミング手順については、*セクション 58.9.11：受信時の VLAN フィルタに関するプログラミングガイドライン*を参照してください。

VLAN タグ完全フィルタ

VLAN タグ完全フィルタでは、MAC は受信パケットの VLAN タグを比較して、VLAN パケットステータスをアプリケーションに提供します。プログラムされたモードに基づいて、MAC は受信した VLAN タグの下位 12 ビットまたは全 16 ビットを比較して完全一致を判断します。

VLAN タグ完全フィルタが有効である場合、MAC は VLAN タグ一致ステータスとともに VLAN タグ付きパケットを転送し、一致しない VLAN パケットをドロップします。*VLAN タグレジスタ（ETH_MACVTR）*の VTIM ビットをセットすることで、VLAN パケットの反転一致も有効にできます。さらに、ETH_MACVTR レジスタの ESVL ビットをセットすることで、デフォルトの C-VLAN タグ付きパケットと合わせて S-VLAN タグ付きパケットの一致を有効にできます。VLAN パケットステータスビット（RDES0 のビット 10）は、一致したパケットの VLAN タグ一致ステータスを示します。

注： ***ソースまたはデスティネーションアドレス（有効時）には、VLAN タグフィルタにわたって優先度があります。これは、ソースまたはデスティネーションアドレスフィルタに失敗したパケットが VLAN タグフィルタの結果に関係なくドロップされることを示します。デフォルトでは、VLAN タグベース完全フィルタはすべての設定で使用できます。***

VLAN タグハッシュフィルタ

MAC は、16 ビットのハッシュテーブルを持つ VLAN タグハッシュフィルタを提供します。MAC は、*VLAN タグレジスタ（ETH_MACVTR）*の VTHM に基づいて、VLAN ハッシュ一致を実行します。VTHM ビットがセットされている場合、VLAN タグの CRC-32 の最上位 4 ビットを使用して、VLAN ハッシュテーブルレジスタの内容をインデックスします。インデックスに対応する VLAN ハッシュテーブルレジスタでの値 1 は、パケットの VLAN タグが一致し、パケットを転送しなければならないことを示します。値 0 は、VLAN タグ付きパケットをドロップしなければならないことを示します。

注： ***VLAN タグの 16 ビットまたは 12 ビットは、ETH_MACVTR レジスタの ETV ビットに基づいた CRC-32 計算に考慮されます。***

ETV ビットがリセットされている場合、VLAN タグの CRC-32 の最上位 4 ビットが反転して使用され、VLAN ハッシュテーブルレジスタ（ETH_MACVHTR）の内容をインデックスします。

ETV ビットがセットされている場合、VLAN タグの CRC-32 の最上位 4 ビットは直接使用され、VLAN タグレジスタ（ETH_MACVTR）の内容をインデックスします。

MAC は、VLAN パケットの反転一致もサポートします。反転一致モードでは、パケットの VLAN タグが完全またはハッシュフィルタに一致するとき、パケットをドロップする必要があります。VLAN 完全および VLAN ハッシュ一致が有効である場合、VLAN ハッシュまたは VLAN 完全フィルタのいずれかに一致すれば、パケットは一致したも

のとみなされます。反転一致がセットされると、完全およびハッシュフィルタが不一致を示した場合のみパケットが転送されます。

*表 497*には、VLAN 一致について異なる可能な内容と最終的な LAN 一致ステータスを示しています。*パケットフィルタ制御レジスタ（ETH_MACPFR）*の RA ビットがセットされている場合、すべてのパケットが受信され、VLAN 一致ステータスが *RDES2 通常ディスクリプタ（書き戻しフォーマット）*の VF ビットに示されます。*パケットフィルタ制御レジスタ（ETH_MACPFR）*で RA ビットがセットされておらず、VTFE ビットがセットされているとき、VLAN 一致ステータスが失敗の場合、パケットはドロップされます。*表 497* で、値 *X* はこの列にどの値でも入ることを示します。

*VLAN タグレジスタ（ETH_MACVTR）*の VL フィールドで VLAN VID が 0 にプログラムされているとき、すべての VLAN タグ付きパケットは完全一致とみなされますが、VLAN ハッシュ一致のステータスは ETH_MACVTR レジスタの VTHM および VTIM ビットに依存します。

表 497. VLAN 一致ステータス(1)

VID	VLAN 完全フィルタ一致の結果	VTHM ビット	VLAN ハッシュフィルタ一致の結果	VTIM ビット	最終的な VLAN 一致ステータス
VID = 0	成功	0	X	X	成功
	成功	1	X	0	成功
	成功	1	失敗	1	成功
	成功	1	成功	1	失敗
VID!= 0	成功	X	X	0	成功
	失敗	0	X	0	失敗
	失敗	1	失敗	0	失敗
	失敗	1	成功	0	成功
	失敗	0	X	1	成功
	成功	X	X	1	失敗
	失敗	1	成功	1	失敗
	失敗	1	失敗	1	成功

1. この表では、X は任意の値を示します。

レイヤ 3 およびレイヤ 4 フィルタ

MAC は、レイヤ 3 およびレイヤ 4 ベースのパケットフィルタをサポートします。レイヤ 3 フィルタは IPv4 または IPv6 パケットでの IP ソースまたはデスティネーションアドレスフィルタを参照し、レイヤ 4 フィルタは TCP または UDP でのソースまたはデスティネーションのポート番号フィルタを参照します。

レイヤ 3 およびレイヤ 4 パケットフィルタ機能は、受信側で IPC フルチェックサムオフロードエンジンを自動的に有効にします。レイヤ 3 およびレイヤ 4 フィルタの操作には、*動作モード設定レジスタ（ETH_MACCR）*の IPC ビットをセットして、Rx チェックサムオフロードエンジンを有効にする必要があります。

レイヤ 3 およびレイヤ 4 フィルタが有効である場合、パケットは次のようにフィルタされます。

- **一致するパケット**

  MAC は、すべての有効化されたフィールドに一致するパケットをステータスとともにアプリケーションに転送します。*動作モード設定レジスタ（ETH_MACCR）* の IPC ビットがセットされ、次のいずれかの条件が当てはまる場合のみ、MAC は一致フィールドステータスを付与します。

  – すべての有効化されたレイヤ 3 およびレイヤ 4 フィールドが一致する。

  – 少なくとも 1 つの有効化されたフィールドが一致し、その他のフィールドはバイパスされるか無効化される。

  複数のレイヤ 3 およびレイヤ 4 フィルタが有効である場合、どのフィルター致でも一致とみなされます。複数のフィルタが一致する場合、MAC はフィルタ 0 が最小フィルタで、フィルタ 3 が最高フィルタであれば、最小フィルタのステータスを提供します。たとえば、フィルタ 0 とフィルタ 1 が一致する場合、MAC はフィルタ 0 に対応するステータスを付与します。

  ソースまたはデスティネーションアドレスおよび VLAN タグフィルタ（有効時）には、レイヤ 3 およびレイヤ 4 フィルタにわたって優先度があります。これは、ソースまたはデスティネーションアドレスまたは VLAN タグフィルタに失敗したパケットがレイヤ 3 およびレイヤ 4 フィルタの結果に関係なくドロップされることを示します。

- **一致しないパケット**

  MAC は、どの有効化されたフィールドにも一致しないパケットをドロップします。反転一致機能を使用して、IP フィールドにまたがる特定の TCP または UDP でパケットをブロックまたはドロップでき、他のすべてのパケットを転送できます。

- **非 TCP または非 UDP の IP パケット**

  デフォルトでは、すべての非 TCP または非 UDP の IP パケットは、レイヤ 3 およびレイヤ 4 フィルタからバイパスされます。オプションで MAC をプログラムして、IP パケットにまたがる非 TCP または非 UDP をすべてドロップすることができます。

レイヤ 3 およびレイヤ 4 フィルタのレジスタセット

MAC は、レイヤ 3 およびレイヤ 4 ベースのパケットフィルタ向けに 2 セットのレジスタを実装しています。レジスタセットには、ETH_MACL3L4C0R などパケットフィルタを制御する制御レジスタがあります。さらに、次のような一致するレイヤ 3 およびレイヤ 4 フィールドをプログラムする 5 つのアドレスレジスタがあります。

- *レイヤ 4 アドレスフィルタ 0 レジスタ（ETH_MACL4A0R）*
- *レイヤ 3 アドレス 0 フィルタ 0 レジスタ（ETH_MACL3A00R）*
- *レイヤ 3 アドレス 1 フィルタ 0 レジスタ（ETH_MACL3A10R）*
- *レイヤ 3 アドレス 2 フィルタ 0 レジスタ（ETH_MACL3A20）*
- *レイヤ 3 アドレス 3 フィルタ 0 レジスタ（ETH_MACL3A30）*

2 つ目および独立したレジスタのセットは次のとおりです。ETH_MACL3L4C1R、ETH_MACL4A01R、ETH_MACL4A11R、ETH_MACL4A21R および ETH_MACL4A31R

レイヤ 3 フィルタ

MAC は、IP ソースアドレスおよびデスティネーションアドレスの完全一致または反転一致をサポートします。さらに、完全な IP アドレスを一致させたり、下位ビット一致をマスク、つまり指定した下位マスクビットを除いたアドレスのすべてのビットを比較したりできます。

IPv6 パケットフィルタでは、レジスタセットの最後の 4 つのデータレジスタを有効にして、128 ビットの IP ソースアドレスや IP デスティネーションアドレスを含めることができます。IP ソースアドレスやデスティネーションアドレスは、IPv6 仕様で定義された順でプログラムしなければなりません。つまり、受信パケットの IP ソースアドレスやデスティネーションアドレスの最初のバイトはレジスタの上位バイトであり、それに続くレジスタは同じ順番に従います。

IPv4 パケットフィルタでは、レジスタセットの 2 番目と 3 番目のデータレジスタを有効にして、32 ビットの IP ソースアドレスや IP デスティネーションアドレスを含めることができます。残りの 2 つのデータレジスタは予約済

みとなります。IP ソースアドレスやデスティネーションアドレスは、IPv4 仕様で定義された順でプログラムしなければなりません。つまり、受信パケットの IP ソースアドレスやデスティネーションアドレスの最初のバイトは各レジスタの上位バイトとなります。

レイヤ 4 フィルタ

MAC は、TCP または UDP のソースおよびデスティネーションのポート番号の完全一致または反転一致をサポートします。ただし、一度に 1 つのタイプ（TCP または UDP）のみプログラムできます。最初のデータレジスタには、TCP または UDP の 16 ビットのソースおよびデスティネーションのポート番号が含まれます。つまり、ソースのポート番号の下位 16 ビットとデスティネーションのポート番号の上位 16 ビットです。

TCP または UDP のソースおよびデスティネーションのポート番号は、TCP や UDP の仕様で定義された順でプログラムしなければなりません。つまり、受信パケットの TCP または UDP のソースポート番号やデスティネーションポート番号の最初のバイトはレジスタの上位バイトとなります。

58.5.4 IEEE 1588 タイムスタンプ

IEEE 1588 標準は、ネットワーク通信、ローカルコンピューティング、分散オブジェクトなどのテクノロジーで実装される計測および制御システムでの高精度なクロック同期を可能にする高精度時間プロトコル（PTP）を定義しています。この PTP は、Ethernet をはじめとして（ただし、これに限られず）、マルチキャストメッセージングをサポートするローカルエリアネットワークによって通信するシステムに適用されます。このプロトコルは、固有精度、解像度、および安定性が異なるクロックを含む異種混合システムを同期することができます。このプロトコルは、最小限のネットワークおよびローカルクロック計算リソースを使用して、サブマイクロ秒範囲のシステム全体の正確な同期をサポートします。

Ethernet ペリフェラルは、IEEE 1588-2002（バージョン 1）および IEEE 1588-2008（バージョン 2）をサポートします。IEEE 1588-2002 は、UDP/IP 経由で転送される PTP をサポートします。IEEE 1588 2008 は、Ethernet 経由で転送される PTP をサポートします。ペリフェラルは、両方の標準に対してプログラム可能なサポートを提供します。以下の機能をサポートしています。

- 両方のタイムスタンプフォーマットのサポート
- すべてのパケットまたは PTP タイプパケットのみのオプションのスナップショット
- イベントメッセージのみのオプションのスナップショット
- クロックタイプに基づいたオプションのスナップショット：通常、境界、エンドツーエンドトランスペアレント、ピアツーピアトランスペアレント
- 通常および境界クロックにマスタまたはスレーブとして機能するノードのオプションの選択
- Ethernet 経由で直接送信されたパケットでの PTP、メッセージタイプ、バージョン、PTP ペイロードの識別およびステータスの送信
- デジタルまたはバイナリ形式でのオプションの測定のサブセカンドの時間

遅延リクエストレスポンスメカニズム

システムまたはネットワークは、タイミングおよびクロック情報を分配するために、マスタノードとスレーブノードに分類されます。*図 774* では、PTP メッセージを交換することによってスレーブノードをマスタノードに同期するために PTP が使用するプロセスを示します。

図 774. ネットワーク時間の同期

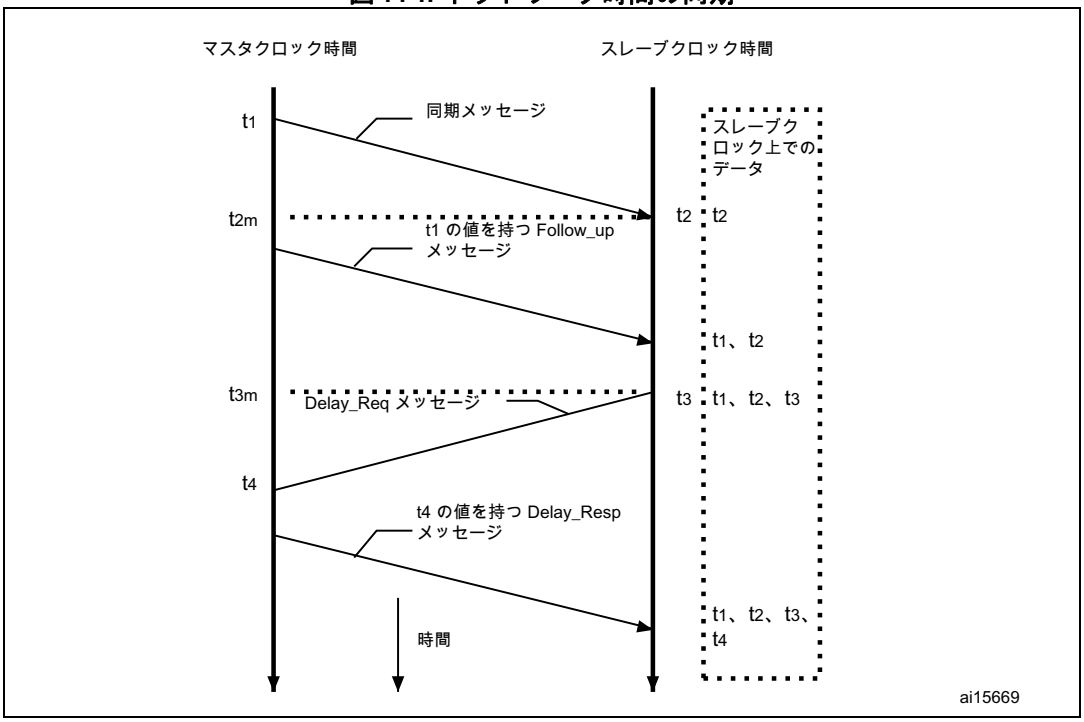


図 774 に示されるとおり、PTP は次のプロセスを使用します。

1. マスタは、PTP 同期メッセージをすべてのノードにブロードキャストします。同期メッセージは、マスタの基準時間情報を含んでいます。このメッセージは、t_1 でマスタのシステムから送信されます。この時間は、MII で Ethernet ポートにおいてキャプチャされます。
2. スレーブは、同期メッセージを受信し、その基準タイミングを使用して、正確な時間 t_2 もキャプチャします。
3. マスタは、後で使用する t_1 情報を含んだ Follow_up メッセージをスレーブに送信します。
4. スレーブは、マスタに Delay_Req メッセージを送信して、このパケットが MII インタフェースから送信された正確な時間 t_3 を知らせます。
5. マスタは、このメッセージを受信して、メッセージがシステムに着信した正確な時間 t_4 をキャプチャします。
6. マスタは、Delay_Resp メッセージで t_4 情報をスレーブに送信します。
7. スレーブは、t_1、t_2、t_3、および t_4 の 4 つの値を使用して、マスタのローカル基準タイミングをマスタの基準タイミングに同期します。

PTP 実装のほとんどは、UDP レイヤの上のソフトウェアで行われます。ただし、特定の PTP パケットが MII インタフェースの Ethernet ポートで送受信された正確な時間をキャプチャするには、ハードウェアサポートが必要です。PTP の正しく高精度な実装のためには、このタイミング情報がキャプチャされて、ソフトウェアに返されなければなりません。

参考資料

ピア・ツー・ピアトランスペアレントクロック（P2P TC）メッセージのサポート

IEEE 1588-2008 標準は、同期、遅延リクエスト、フォローアップ、遅延レスポンスのメッセージに加えて、ピアツーピア PTP（Pdelay）メッセージをサポートします。*図 775* では、ピアツーピアパス補正をサポートするクロック単位での伝搬遅延を計算する方法を示しています。

図 775. ピアツーピアパスパス補正をサポートする伝搬遅延計算

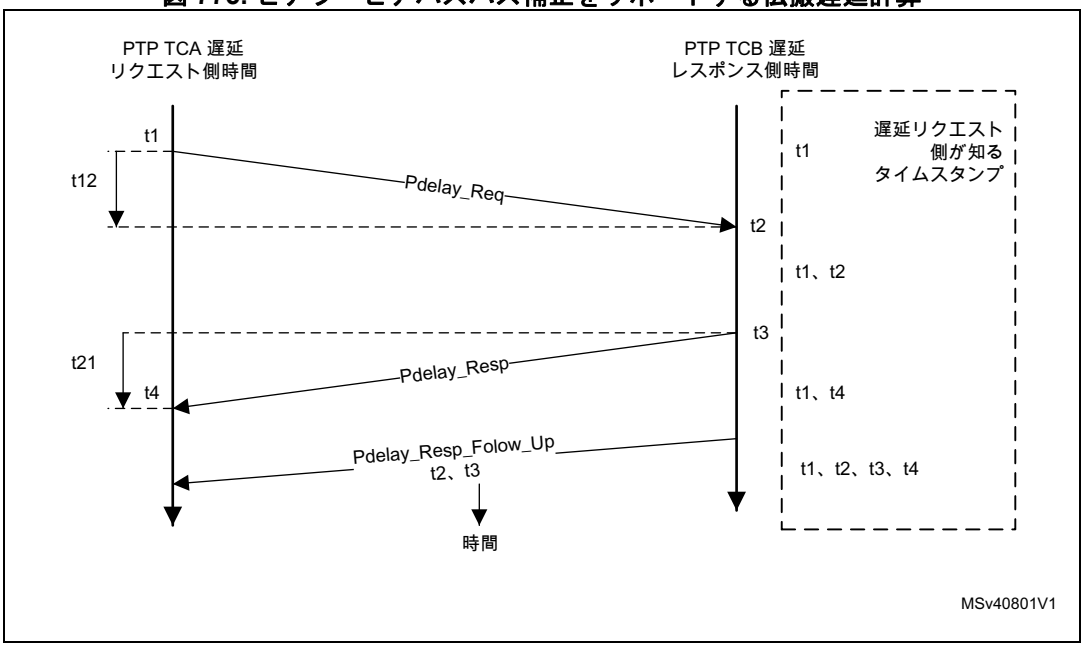


図 775 に示されるとおり、伝搬遅延は次のように計算されます。

1. ポート 1 は Pdelay_Req メッセージを発行し、Pdelay_Req メッセージにタイムスタンプ（t_1）を生成します。
2. ポート 2 は Pdelay_Req メッセージを受信し、このメッセージにタイムスタンプ（t_2）を生成します。
3. ポート 2 は Pdelay_Resp メッセージを返し、このメッセージにタイムスタンプ（t_3）を生成します。

   2 つのポート間の周波数オフセットによって生じる誤差を最低限にするために、ポート 2 は Pdelay_Req メッセージの受信後、可及的速やかに Pdelay_Resp メッセージを返します。ポート 2 は、次のいずれかを返します。

   – Pdelay_Resp メッセージでのタイムスタンプ t_2 と t_3 の差
   – Pdelay_Resp_Follow_Up メッセージでのタイムスタンプ t_2 と t_3 の差
   – Pdelay_Resp および Pdelay_Resp_Follow_Up メッセージでのそれぞれのタイムスタンプ t_2 と t_3

   ポート 1 は Pdelay_Resp メッセージを受信すると、タイムスタンプ（t_4）を生成します。

   ポート 1 は 4 つすべてのタイムスタンプを使用して、平均リンク遅延を計算します。

クロックタイプ

MAC は、IEEE 1588-2008 仕様で定義された次のクロックタイプをサポートします。

- **通常クロック**

  ドメインの通常クロックは、プロトコルのシングルコピーをサポートします。シングル PTP 状態とシングル物理ポートを備えています。標準的な工業自動化アプリケーションでは、通常クロックがセンサやアクチュ

エータなどのアプリケーションデバイスと関連付けられます。電気通信アプリケーションでは、通常クロックをタイミング境界設定デバイスと関連付けることができます。

通常クロックは、グランドマスタクロックまたはスレーブクロックとすることができます。以下の機能をサポートしています。

– PTP メッセージの送受信。タイムスタンプスナップショットは、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*に示すように制御できます。
– タイムスタンプの値などデータセットのメンテナンス。

下の表に、マスタノードおよびスレーブノードの受信側でタイムスタンプスナップショットを取得できるメッセージを示します。

表 498. 通常クロック：スナップショット用 PTP メッセージ

マスタ	スレーブ
Delay_Req	SYNC

通常クロックでは、バージョン 1 またはバージョン 2 のいずれかの PTP メッセージタイプのスナップショットを取得できます。両方の PTP メッセージタイプのスナップショットは取得できません。*タイムスタンプ制御レジスタ（ETH_MACTSCR）*で TSVER2ENA ビットをセットしてスナップショットモードを選択することで、スナップショットを取得できます。

- **境界クロック**

  通常、境界クロックにはネットワークと通信する複数の物理ポートがあります。同期、マスタスレーブ階層、シグナリングに関連するメッセージは、境界クロックのプロトコルエンジンで終了します。これらのメッセージは転送されません。MAC によって示された PTP メッセージタイプステータスは、メッセージのタイプを識別して、適切な操作を行うのに役立ちます。

  境界クロックは、次の特徴を除いて通常クロックに類似しています。

  – クロックのデータセットは、境界クロックではすべてのポートに共通です。
  – ローカルクロックは、境界クロックではすべてのポートに共通です。

- **エンドツーエンドトランスペアレントクロック**

  エンドツーエンドトランスペアレントクロックは、スレーブクロックとマスタクロックでのエンドツーエンド遅延測定メカニズムをサポートします。エンドツーエンドトランスペアレントクロックは、通常のブリッジ、ルータ、リピータのように、すべてのメッセージを転送します。PTP パケットの滞留時間は、入力ポートから出力ポートまで PTP パケットがかける時間です。

  エンドツーエンドトランスペアレントクロック内の SYNC パケットの滞留時間は、関連付けられた Follow_Up PTP パケットの補正フィールドで更新されてから送信されます。同様に、エンドツーエンドトランスペアレントクロック内の Delay_Req パケットの滞留時間は、関連付けられた Delay_Resp PTP パケットの補正フィールドで更新されてから送信されます。そのため、次の文章で言及されるメッセージに対してのみ、入力ポートと出力ポートの両方でスナップショットを取得する必要があります。そのため、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*で言及されるメッセージに対してのみ、入力ポートと出力ポートの両方でスナップショットを取得する必要があります。スナップショットは、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*で SNAPTYPSEL ビットを 10 にセットすると取得できます。*タイムスタンプ制御レジスタ（ETH_MACTSCR）*

表 499. エンドツーエンドトランスペアレントクロック：スナップショット用 PTP メッセージ

PTP メッセージ
SYNC
Delay_Req

- **ピア・ツー・ピアトランスペアレントクロック**

  ピアツーピアトランスペアレントクロックは、PTP タイミングメッセージを補正して処理する方法で、エンドツーエンドトランスペアレントクロックとは異なります。その他すべての面で、エンドツーエンドトランスペアレントクロックと同一です。

  ピアツーピアトランスペアレントクロックでは、リンク遅延の計算はリンクピアとの Pdelay_Req、Pdelay_Resp、および Pdelay_Resp_Follow_Up メッセージの交換に基づきます。Pdelay_Req および関連する Pdelay_Resp パケットの滞留時間が、関連付けられた Pdely_Resp_Followup パケットの補正フィールドに追加され挿入されます。そのため、Pdelay に関連するイベントメッセージに対するスナップショット取得のサポートが、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*に示されるとおり追加されます。

表 500. ピア・ツー・ピアトランスペアレントクロック：スナップショット用 PTP メッセージ

PTP メッセージ
SYNC
Pdelay_Req
Pdelay_Resp

スナップショットは、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*で SNAPTYPESEL ビットを 11 にセットすると取得できます。

基準タイミングソース

時間のスナップショットを取得するために、MAC は IEEE 1588-2008 仕様で定義される 80 ビット形式で内部基準時間を提供します。

MAC クロック入力は、基準時間（システム時間とも呼ばれます）を生成し、タイムスタンプをキャプチャするために使用されます。タイムスタンプには、次のフィールドがあります。

- UInteger48 secondsField
- secondsField は、秒単位でのタイムスタンプの整数部分です。これは 48 ビット幅になります。たとえば、2.000000001 秒は secondsField = 0x0000_0000_0002 で表されます。
- UInteger32 nanosecondsField
- nanosecondsField は、ナノ秒単位でのタイムスタンプの分数部分です。たとえば、2.000000001 ナノ秒は nanoSeconds = 0x0000_0001 で表されます。

  nanosecondsField は次の 2 つのモードをサポートします。

  – **デジタルロールオーバーモード**：このモードでは、ナノ秒フィールドの最大値が 0x3B9A_C9FF で、（10e9-1）ナノ秒となります。

  – **バイナリロールオーバーモード**：このモードでは、ナノ秒フィールドがロールオーバーし、値 0x7FFF_FFFF の後に秒フィールドをインクリメントします。精度はビット当たり～0.466 ns です。

  これらのモードは、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*の TSCTRLSSR ビットでセットできます。

基準タイミングは、pulse-per-second 信号（*Pulse-per-second 出力*を参照）の出力または専用 FIFO レジスタ（*外部イベントでの補助スナップショット*）に最大 4 つのスナップショットを保存することでアクセス可能です。

Pulse-per-second 出力

MAC は、固定の pulse-per-second 信号または柔軟性の高い pulse-per-second 信号をサポートします。

- 固定の pulse-per-second 出力

  このモードでは、*PPS 制御レジスタ（ETH_MACPPSCR）*の PPSCTRL0 ビットをセットすることによって、PPS 出力の周波数だけを変更することができます。

- 柔軟性の高い pulse-per-second 出力

  このモードでは、ptp_pps_o 出力で生成されるパルスの開始時間、停止時間、幅、間隔をプログラムできる柔軟性があります。

  – 開始時間と停止時間は、*PPS 目標時間秒レジスタ（ETH_MACPPSTTSR）*レジスタおよび *PPS 目標時間ナノ秒レジスタ（ETH_MACPPSTTNR）*レジスタでプログラムされます。

  – PPS の幅と間隔は、それぞれのレジスタ *PPS 幅レジスタ（ETH_MACPPSWR）*および *PPS 間隔レジスタ（ETH_MACPPSIR）*でシステム時間の粒度（サブセカンドのインクリメント値の単位の数）でプログラムされます。

柔軟性の高いパルス出力の設定方法の詳細については、*セクション 58.9.9：柔軟性の高い Pulse-Per-Second（PPS）出力に関するプログラミングガイドライン*を参照してください。

注： ***適切な PPS 信号出力であることを確認するために、開始時間や停止時間に高度なシステム時間をプログラムすることを推奨します。アプリケーションがすでに経過した開始時間や停止時間をプログラムした場合、MAC はプログラミングエラーを示すエラーステータスビットをセットします。有効になっている場合、MAC は目標時間到達の割り込みイベントもセットします。対応する開始時間または停止時間が経過していない場合のみ、アプリケーションは開始または停止リクエストをキャンセルできます。時間が経過した場合、キャンセルコマンドは有効ではありません。***

外部イベントでの補助スナップショット

補助スナップショット機能によって、外部イベントに基づいたシステム時間のスナップショットを保存できます。イベントは、ptp_aux_ts_trig_i サイドバンド信号の立ち上がりエッジとみなされます。

最大 4 つの補助スナップショット入力を設定して、最大 4 つのスナップショットを保存できます。FIFO は、次のレジスタでアクセス可能です。*補助タイムスタンプ秒レジスタ（ETH_MACATSSR）*および*補助タイムスタンプナノ秒レジスタ（ETH_MACATSNR）*。

入力に対して取得されたスナップショットは、共通の FIFO に保存されます。64 ビットのみ確保されます。アプリケーションは、*タイムスタンプステータスレジスタ（ETH_MACTSSR）*を読み出して、この FIFO の最上部での読み出しに入力が使用できるタイムスタンプを把握することができます。

スナップショットが保存されると、MAC は割り込みでアプリケーションに通知します。スナップショットの値は、FIFO レジスタアクセスから読み出されます。FIFO がフルになり、スナップショットを取得する外部トリガがアサートされた場合、*タイムスタンプステータスレジスタ（ETH_MACTSSR）*でスナップショットトリガミスステータス（ATSSTM）がセットされます。これは、タイムスタンプの最新の補助スナップショットが FIFO に保存されないことを示します。FIFO がフルである場合、最新のスナップショットは書き込まれません。

アプリケーションが FIFO から 64 ビットタイムスタンプを読み出すとき、次のスナップショットを保存するために空きが使用できるようになります。FIFO は、*補助制御レジスタ（ETH_MACACR）*で ATSFC ビットをセットするとクリアできます。複数のスナップショットが FIFO にある場合、カウントが ETH_MACTSSR レジスタのビット [27:25] に示されます。

参考資料

システム時間レジスタモジュール

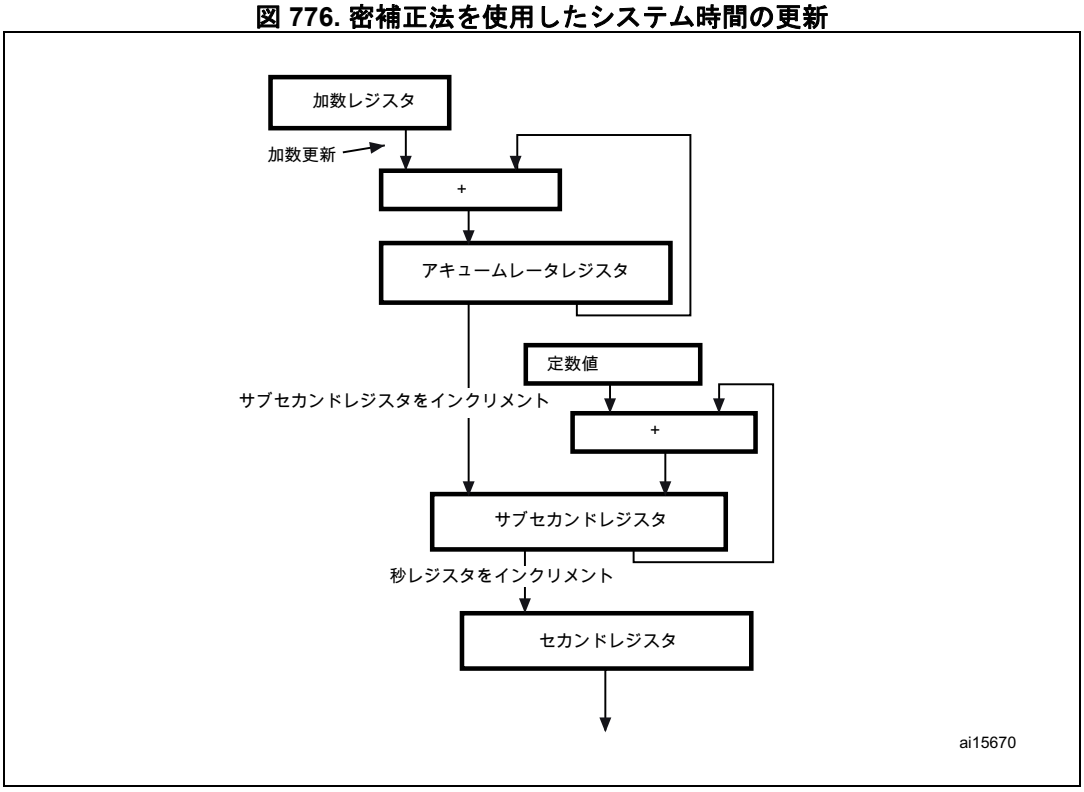
64 ビットの PTP 時間は、PTP 入力リファレンスクロック HCLK を使用して更新されます。この PTP 時間は、MII で送受信される Ethernet フレームのスナップショット（タイムスタンプ）をとるためのソースとして使用されます。システム時間カウンタは、粗密補正法を使用して初期化または補正できます。

粗補正法では、初期値またはオフセット値がタイムスタンプ更新レジスタに書き込まれます。初期化の場合は、システムタイムカウンタにタイムスタンプ更新レジスタの値が書き込まれますが、システム時間の補正の場合は、オフセット値（タイムスタンプ更新レジスタ）がシステム時間に加算または減算されます。

密補正法では、マスタクロックに対するスレーブクロック（基準クロック）の周波数ドリフト（IEEE 1588 仕様で定義）が、一定時間にわたって補正されますが、粗補正法では、補正は 1 クロックサイクルで行われます。補正時間が長いほど、リニア時間を維持しやすく、PTP 同期メッセージインターバル間の基準時間に大きな変化（または大きなジッタ）が生じません。この方法では、*図 776* に示されるように、加算器が加数レジスタの内容を累積します。加算器が生成する桁上げは、システム時間カウンタをインクリメントするためのパルスとして使用されます。加算器と加数レジスタは、32 ビットのレジスタです。加算器は、高精度な周波数逓倍器または分周器として動作します。

このシステム時間更新アルゴリズムは、*図 776* に示しています。

図 776. 密補正法を使用したシステム時間の更新



システム時間更新ロジックは、20 ns の精度を達成するために 50 MHz のクロック周波数を必要とします。周波数分周比は、必要なクロック周波数に対する基準周波数の比です。たとえば、リファレンスクロック（HCLK）が 66 MHz の場合、この分周比は 66 MHz/50 MHz = 1.32 になります。したがって、レジスタで設定するデフォルトの加数値は、2^{32}/1.32、0xC1F07C1F です。

リファレンスクロックが低周波数側にドリフトして、たとえば 65 MHz になった場合、分周比は 65/50、すなわち 1.3 となり、加数レジスタで設定する値は 2^{32} / 1.30、すなわち 0xC4EC4EC4 です。

基準クロックが高周波数側にドリフトして、たとえば 67 MHz になった場合は、加数レジスタを 0xBF0B7672 に設定する必要があります。クロックのドリフトがないときには、デフォルトの加数値として 0xC1F07C1F（2^{32} / 1.32）をプログラムする必要があります。

図 776 では、サブセカンドレジスタを累積するために使用される一定値は 0.43 で、20 ns のシステム時間精度を達成します（言い換えると、20 ns 単位でインクリメントされます）。外部時間更新が有効になっている場合、オプションのシステム時間モジュールは使用できません。設定に応じてシステム時間レジスタを更新するには、2 つの異なる方法を使用します。

ソフトウェアは、SYNC メッセージに基づいて周波数のドリフトを計算し、それに応じて加数レジスタを更新する必要があります。

最初に、スレーブクロックは加数レジスタで FreqCompensationValue0 で設定されます。この値は、次のとおりです。

$$FreqCompensationValue_0 = 2^{32} / FreqDivisionRatio$$

MasterToSlaveDelay が連続した同期メッセージで同じであると仮定した場合、このセクションで示されるアルゴリズムが適用されなければなりません。数サイクルの同期サイクル後、周波数ロックが発生します。スレーブクロックは、正確な MasterToSlaveDelay 値を決定して、新しい値を使用してマスタと再同期します。

アルゴリズムは、次のとおりです。

1. $MasterSyncTime_n$ の時点で、マスタはスレーブクロックに SYNC メッセージを送信します。スレーブは、ローカルクロックが $SlaveClockTime_n$ のときに、このメッセージを受信して、$MasterClockTime_n$ を次のように計算します。

$$MasterClockTime_n = MasterSyncTime_n + MasterToSlaveDelay_n$$

2. 現在の同期サイクルのマスタクロックカウント、$MasterClockCount_n$ は、次の式で求められます。

$$MasterClockCount_n = MasterClockTime_n - MasterClockTime_{n-1}$$

（MasterToSlaveDelay が同期サイクル n と n -1 で同じであると仮定）

3. 現在の同期サイクルのスレーブクロックカウント、$SlaveClockCount_n$ は、次の式で求められます。

$$SlaveClockCount_n = SlaveClockTime_n - SlaveClockTime_{n-1}$$

4. 現在の同期サイクルのマスタとスレーブのクロックカウントの差、$ClockDiffCount_n$ は、次の式で求められます。

$$ClockDiffCount_n = MasterClockTime_n - SlaveClockTime_n$$

5. スレーブクロックの周波数スケーリングファクタ、$FreqScaleFactor_n$ は、次の式で求められます。

$$FreqScaleFactor_n = (MasterClockCount_n + ClockDiffCount_n) / SlaveClockCount_n$$

6. 加数レジスタの周波数補正値、$FreqCompensationValue_n$ は、次の式で求められます。

$$FreqCompensationValue_n = FreqScaleFactor_n \times FreqCompensationValue_{n-1}$$

理論的には、このアルゴリズムは 1 同期サイクルでロックを達成します。ただし、ネットワークの伝搬遅延と動作条件の変動のために、数サイクルかかる場合もあります。このアルゴリズムは自己補正型です。最初にスレーブクロックがマスタからの誤った値に設定された場合でも、アルゴリズムは同期サイクルを数サイクル費やして、それを補正します。

詳細なプログラミング手順については、*セクション 58.9.7 : IEEE 1588 タイムスタンプに関するプログラミングガイドライン*を参照してください。

送信パス機能

MAC は、パケットの開始パケットデリミタ（SFD）が MII インタフェースで送信されたときにタイムスタンプをキャプチャします。タイムスタンプをキャプチャしたいパケットは、パケット単位の基準で制御できます。各送信パケットをマークして、タイムスタンプをキャプチャする必要があるかどうかを示すことができます。

MAC は、PTP パケットを識別するために送信されたパケットを処理しません。タイムスタンプをキャプチャしたいパケットを指定する必要があります。送信ディスクリプタ（*セクション 58.10.3 : 送信ディスクリプタ*を参照）で制御ビットを使用することでパケットを指定できます。MAC は、対応する送信ディスクリプタ内でソフトウェアにタイムスタンプを返して、特定の PTP パケットにタイムスタンプを自動的に結びつけます。

64 ビットタイムスタンプ情報は、TDES0 および TDES1 のフィールドに書き込まれます。TDES0 フィールドは、タイムスタンプの最下位 32 ビットを保持します。

受信パス機能

MAC は、MII インタフェースで受信したすべてのパケットのタイムスタンプをキャプチャしたり、パケットを処理して有効な PTP メッセージを特定したりするためにプログラムできます。*タイムスタンプ制御レジスタ（ETH_MACTSCR）*の次のオプションを使用して、アプリケーションに送信される時間のスナップショットを制御できます。

- すべてのパケットに対するスナップショットの有効化
- IEEE 1588 バージョン 1 またはバージョン 2 のタイムスタンプに対するスナップショットの有効化
- Ethernet または UDP-IP-Ethernet 経由で直接送信される PTP パケットに対するスナップショットの有効化
- IPv4 または IPv6 の受信パケットに対するタイムスタンプスナップショットの有効化
- イベントメッセージに対するタイムスタンプスナップショットの有効化（SYNC、DELAY_REQ、PDELAY_REQ、または PDELAY_RESP）
- マスタまたはスレーブになるノードの有効化とスナップショットタイプの選択

  この機能は、スナップショットを取得するメッセージのタイプを制御します。

DMA は、対応する受信ディスクリプタ内でソフトウェアアプリケーションにタイムスタンプを返します。タイムスタンプメッセージステータスおよび IPC ステータスを含む拡張ステータスは、RDES1 通常ディスクリプタに書き込まれ、タイムスタンプのスナップショットはコンテキストディスクリプタの RDES0 および RDES1 のフィールドに書き込みます。RDES0 フィールドは、タイムスタンプの最下位 32 ビットを保持します。

タイムスタンプ補正

IEEE 1588 仕様に従って、メッセージタイムスタンプポイント（開始フレームデリミタオクテットの直後のオクテットの最初のビットの先端）がノードとネットワークの境界を越えるときに、タイムスタンプをキャプチャしなければなりません。基準タイミングソース（PTP クロック ***HCLK***）が MAC Tx または Rx クロックと異なるため、キャプチャされたタイムスタンプは同期のためにレイテンシ問題について補正する必要があります。さらに、内部スナップショットポイントと推奨キャプチャポイントのレイテンシ問題（ノードとネットワークの境界）も補正しなければなりません。

入力補正

受信側では、内部スナップショットポイントでキャプチャされたタイムスタンプは、入力補正レジスタでプログラムされた補正値（入力補正値）を追加することで、レイテンシおよび同期に備えて補正されます。入力補正レジスタでプログラムする必要のある値は、次のように計算されます。

1. 同期のためのタイムスタンプ補正は、次のようにして同期されるタイムスタンプ値に INGRESS_SYNC_CORR を追加することで行われます。

   INGRESS_SYNC_CORR = -(2*PTP_CLK_PER)

2. メッセージタイムスタンプポイントと内部タイムスタンプスナップショットポイントのレイテンシ補正は、次のようにしてキャプチャされたタイムスタンプからレイテンシ値（INGRESS_LATENCY）を引くことで行われます。

   入力補正 = INGRESS_SYNC_CORR - INGRESS_LATENCY

3. 入力補正は、MAC のタイムスタンプ入力補正レジスタで TSIC フィールドをプログラミングすることによって実行されます。入力補正は常に負の数で、ナノ秒で表されます。この値は、次のように補数形式で表されます。
   - *タイムスタンプ制御レジスタ（ETH_MACTSCR）*の TSCTRLSSR ビットがセットされているとき、精度が 1 ns の場合、ビット 31 を ‘1’ にセットして、2 進法で表された 10^9 - <ingress_correction_value> を含むビット 30:0 で表されます。たとえば、必要な補正値が -5 ns の場合、プログラムされる値は、0xBB9A_C9FB でなければなりません。
   - ETH_MACTSCR レジスタの TSCTRLSSR ビットがリセットされているとき、精度が ～0.466 ns の場合、ビット 31 を ‘1’ にセットして、2 進法で表された 2^31 - <ingress_correction_value> を含むビット 30:0 で表されます。

出力補正

送信側では、内部スナップショットポイントでキャプチャされたタイムスタンプは、出力補正レジスタでプログラムされた補正値（出力補正値）を追加することで、レイテンシおよび同期に備えて補正されます。出力補正レジスタでプログラムする必要のある値は、次のように計算されます。

1. 同期のためのタイムスタンプ補正は、次のようにして同期されるタイムスタンプ値に EGRESS_SYNC_CORR を追加することで行われます。

   ワンステップタイムスタンプ機能の有効化が選択されている場合、

   EGRESS_SYNC_CORR = (1*PTP_CLK_PER + 4*TX_CLK_PER)

   それ以外の場合、

   EGRESS_SYNC_CORR = -(2*PTP_CLK_PER)

2. 推奨されるキャプチャポイントと内部タイムスタンプスナップショットポイントの出力レイテンシ補正は、次のようにしてキャプチャされたタイムスタンプにレイテンシ値（EGRESS_LATENCY）を追加することで取得されます。

   出力補正 = EGRESS_SYNC_CORR + EGRESS_LATENCY

3. 出力補正は、MAC のタイムスタンプ出力補正レジスタで TSEC フィールドをプログラミングすることによって実行されます。出力補正は正の数または負の数になりえます。これはナノ秒単位で表されます。負の値は、次のように補数形式で表されます。
   - *タイムスタンプ制御レジスタ（ETH_MACTSCR）*で TSCTRLSSR ビットがセットされた場合、精度が 1 ns となります。

     補正値が正の場合、ビット 31 を ‘0’ にセットして、2 進法で表された <egress_correction_value> を含むビット 30:0 で表されます。この値は 0x 3B9A_C9FF を超えてはなりません。

     補正値が負の場合、ビット 31 を ‘1’ にセットして、2 進法で表された 10^9 - <egress_correction_value> を含むビット 30:0 で表されます。

     たとえば、必要な補正値が -5 ns の場合、プログラムされる値は、0xBB9A_C9FB でなければなりません。
   - *タイムスタンプ制御レジスタ（ETH_MACTSCR）*で TSCTRLSSR ビットがリセットされた場合、精度が ～0.466 ns となります。

     補正値が正の場合、ビット 31 を ‘0’ にセットして、2 進法で表された <egress_correction_value> を含むビット 30:0 で表されます。最大値は 0x7FFF_FFFF です。

     補正値が負の場合、ビット 31 を ‘1’ にセットして、2 進法で表された 2^31 - <egress_correction_value> を含むビット 30:0 で表されます。

ワンステップタイムスタンプ

MAC は、ワンステップタイムスタンプ機能をサポートします。パケットでのオフセットを識別して、そのオフセットでアプリケーションから受信したタイムスタンプを挿入します。

ATI 制御ワードでビット 20（OSTC）をセットすることで、パケットに対してワンステップタイムスタンプ機能を有効化できます。挿入されたタイムスタンプは、アプリケーションから受信した 64 ビットの TSSL および TSSH で構成されます。

ワンステップタイムスタンプ機能は、Ethernet 経由 PTP パケットでのみサポートされます。IPv4/IPv6 経由 PTP パケットではサポートされません。

PTP オフロード機能

この機能は、MAC が PTP ネットワークで特定のノードとして動作しているときに、実行する特定の PTP パケットの自動生成を可能にします。これらのパケットは、周期的に生成したり、ホストソフトウェアでトリガしたりできます。その他のモードでは、この機能によってレシーバの着信 PTP パケットを解析でき、必要な PTP パケットを自動的に生成して応答できます。これは、特定の PTP ノードの機能をさらに高精度かつ低レスポンスレイテンシでオフロードするのに役立ちます。

プログラムされたモードに応じて、MAC は周期的、またはアプリケーションから、または特定の PTP メッセージの受信に基づいて、PTP Ethernet メッセージを生成します。*表 501* は、PTP メッセージ生成の基準を示します。

表 501. PTP メッセージ生成の基準

プログラミング			モード	PTP メッセージの生成の基準	生成される PTP メッセージタイプ
SNAPTYPSEL	TSMSTRENA	TSEVNTENA			
00	0	1	通常スレーブまたは境界スレーブ	SYNC メッセージの受信	Delay_Req
00	1	1	通常マスタまたは境界マスタ	周期的またはアプリケーションからのトリガ時	SYNC
				Delay_Req メッセージの受信	Delay_Resp
01	0	1	トランスペアレントスレーブ	周期的またはアプリケーションからのトリガ時	Pdelay_Req
				Pdelay_Req メッセージの受信	Pdelay_Resp
				SYNC メッセージの受信	Delay_Req

表 501. PTP メッセージ生成の基準（続き）

プログラミング			モード	PTP メッセージの生成の基準	生成される PTP メッセージタイプ
SNAPTYPSEL	TSMSTRENA	TSEVNTENA			
01	1	1	トランスペアレントマスタ	周期的またはアプリケーションからのトリガ時	Pdelay_Req
				Pdelay_Req メッセージの受信	Pdelay_Resp
				周期的またはアプリケーションからのトリガ時	SYNC
				Delay_Req メッセージの受信	Delay_Resp
11	X	X	ピアツーピアトランスペアレント	周期的またはアプリケーションからのトリガ時	Pdelay_Req
				Pdelay_Req メッセージの受信	Pdelay_Resp
他のすべてのプログラミングの組み合わせは、PTP オフロード機能では無効です。					

PTP オフロード機能は *PTP オフロード制御レジスタ（ETH_MACPOCR）* レジスタで設定します。80 ビット PTP ノードのアイデンティティは、次の 3 つのレジスタで付与されます。*PTP ソースポートアイデンティティ 0 レジスタ（ETH_MACSPI0R）*、*PTP ソースポートアイデンティティ 1 レジスタ（ETH_MACSPI1R）* および *PTP ソースポートアイデンティティ 2 レジスタ（ETH_MACSPI2R）* です。

58.5.5 IPv4 ARP オフロード

MAC は、IPv4 パケットのアドレス認識プロトコル（ARP）オフロードをサポートします。この機能は、受信パスで IPv4 ARP リクエストパケットを処理して、送信パスで対応する ARP レスポンスパケットを生成できます。

MAC は、適切な ARP リクエストパケットに対して ARP 応答パケットを生成します。IPv4 用 ARP パケットは、0x0806 の長さ／タイプを持つ L2 レイヤパケットです。

ARP オフロードシーケンスは、次のとおりです。

1. リクエストターゲットプロトコルアドレスが MAC L3 レジスタでプログラムされた IPv4 アドレスと一致する場合、MAC レシーバが ARP リクエストを取得します。
2. MAC は、ARP 応答パケットを生成します。
3. MAC は、ARP リクエストの送信側ハードウェアアドレスフィールドを次のフィールドにコピーします。
   - Ethernet パケットヘッダの DA フィールド
   - ARP 応答パケットのターゲットハードウェアアドレスフィールド

4. MAC は、ARP リクエストの送信側プロトコルアドレスフィールドを ARP 応答パケットのターゲットプロトコルアドレスフィールドにコピーします。
5. MAC は、次のフィールドに MAC アドレスを配置します。
   – Ethernet パケットヘッダの SA フィールド
   – ARP 応答パケットの送信側ハードウェアアドレスフィールド
6. MAC は、ARP リクエストのターゲットプロトコルアドレスフィールドを ARP 応答パケットの送信側プロトコルアドレスフィールドにコピーします。
7. MAC は、ARP 応答パケットの OP コードフィールドを ARP 応答を示す 2 にセットします。
8. MAC は、生成された ARP 応答パケットに対して CRC を再計算し、パディングを実行します。
9. MAC トランスミッタは ARP 応答を送信します。

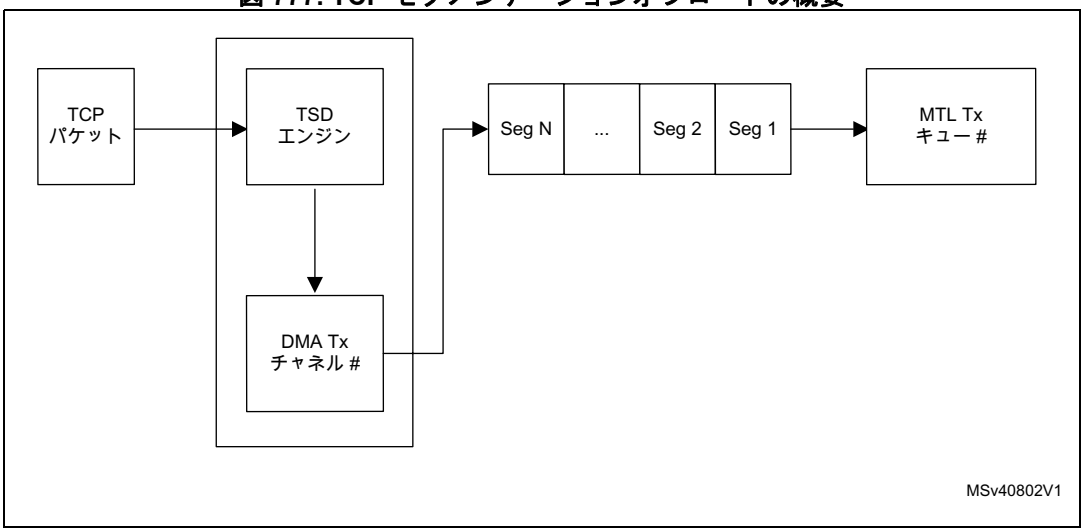
58.5.6 TCP セグメンテーションオフロード

MAC は、DMA が大きな TCP パケットを複数の小さいパケットに分割して、これらのパケットを*図 777*に示された MTL に渡す TCP セグメンテーションオフロード（TSO）機能をサポートします。

この機能は、対応する ETH_DMACCR レジスタの TSE ビットをプログラムすることで有効になります（*チャネル制御レジスタ (ETH_DMACCR)* を参照）。これは、MAC が全二重モードで動作している場合のみサポートされます。

詳細なプログラミング手順については、*セクション 58.9.10 : TSO に関するプログラミングガイドライン* を参照してください。

図 777. TCP セグメンテーションオフロードの概要



TSO 機能の有効化

パケットに対してセグメンテーションを有効にするために、アプリケーションは最初の通常ディスクリプタの TDES3 の TSE ビットをセットする必要があります（*セクション 58.10.3 : 送信ディスクリプタ*を参照）。

アプリケーションは、TDES3 [17 : 0]にTCPパケットペイロードの長さを、TDES3 [22:19]にTCPヘッダーをプログラムする必要があります。分割できる TCP パケットペイロードの最大長は 256 KB です。

Ethernet ヘッダ、L3 ヘッダ、L4 ヘッダを含むパケットのヘッダは、TSO パケットの最初の通常ディスクリプタのバッファ 1 で提供される必要があります。TSO の有効化されるパケットの最初の通常ディスクリプタのバッファ 1 のみ、ヘッダを含むバッファとして取得されます。

TCP ペイロードは、最初の通常ディスクリプタのバッファ 2 から開始し、2 番目の通常ディスクリプタのバッファ 1、バッファ 2、そして後続のディスクリプタへと続きます。

TCP ペイロードは、複数のバッファおよび複数のディスクリプタにまたがることができます。TCP ペイロードを含むバッファのサイズは、最初の通常ディスクリプタの TDES3[17:0] で提供される TCP ペイロード長と等しくなるまで追加する必要があります。

MAC は、DMA によって分割されたすべてのパケットに対して、必ず CRC を計算して付加し、パディング（必要な場合）を挿入します。TDES3 の TSE ビットが有効である場合、TDES3 の CRC PAD 制御（CPC）フィールドは予約済みになります。分割後の TCP パケットのサイズを決定するために、DMA はコンテキストディスクリプタを通じてアプリケーションが提供する最大セグメントサイズ（MSS）を使用します。DMA は、MSS を超えるペイロードサイズを持つパケットのみ分割します。アプリケーションは、ETH_DMACCR（*チャネル 制御レジスタ（ETH_DMACCR）*を参照）の MSS 値をプログラムするか、コンテキストディスクリプタを提供することによって MSS を提供する必要があります。DMA は、MSS の最後にプログラムされた値か、コンテキストを通じて提供される最後の MSS 値（後から提供されるどの値でも）を使用します。

ヘッダの長さ + MSS サイズ（各 TCP セグメントのサイズと同等）は 16383 バイトを超えてはなりません。そうしないと、MAC トランスミッタは CRC エラーを引き起こす 16383 バイト以降のパケットを切り詰めます。

ヘッダの長さ + MSS サイズ + ETH_DMACTxCR レジスタ（*チャネル 送信制御レジスタ（ETH_DMACTxCR）*を参照）でプログラムされた PBL 値は、ETH_MTLTxQOMR レジスタ（*Tx キュー動作モードレジスタ（ETH_MTLTxQOMR）*を参照）の TQS フィールドでプログラムされた Tx キューサイズ未満にする必要があります。MSS とプログラムされた Tx キューサイズの半分に相当するヘッダが推奨されます。

TCP パケットに VLAN タグがある場合、提供される VLAN タグタイプ（C-VLAN または S-VLAN）に関係なく、すべてのセグメントに対して同じタグが使用されます。VLAN タグ挿入／書き換え制御ビットは、すべてのセグメントに対して使用されます。

ダブル VLAN 機能を選択している場合、提供される VLAN タグタイプ（C-VLAN または S-VLAN）に関係なく、すべてのセグメントに対して DMA が両方のタグを通過します。両方のタグの VLAN タグ挿入／書き換え制御ビットは、すべてのセグメントに対して適用できます。ダブル VLAN 機能を選択していない場合、アプリケーションは 2 つのタグを持つ TCP/IP パケットに対して TDES3 で TSE ビットをセットしてはなりません。このシナリオでの DMA の動作は予想できません。

TCP/IP ヘッダフィールド

TCP パケットを分割しつつ、DMA は TCP/IP ヘッダフィールドを自動的に更新します。*表 502* では、TCP ヘッダおよび IP ヘッダがどのように更新されるかを示しています。

表 502. TSO：TCP ヘッダおよび IP ヘッダフィールド

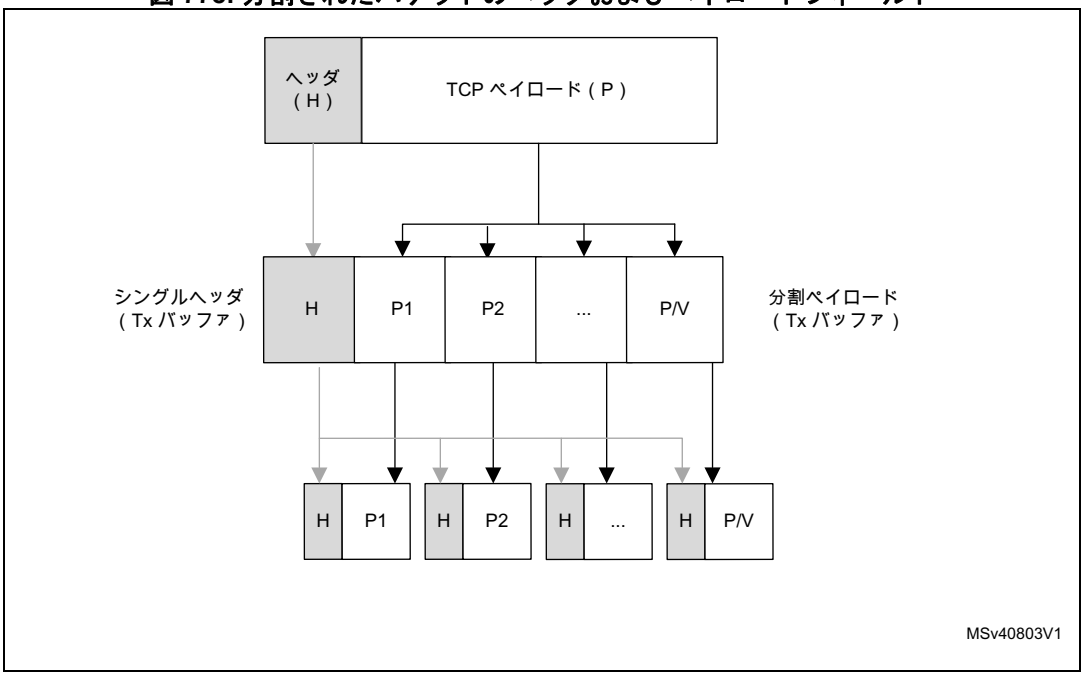
パケットシーケンス	TCP ヘッダ	IP ヘッダ
最初のパケット	1. シーケンス数は更新されません。ヘッダで提供された値が使用されます。 2. セットしている場合、FIN フラグおよび PSH フラグがクリアされます。 3. TCP チェックサムが再計算されます。	IPv4 ヘッダ – 合計の長さ = MSS + TCP ヘッダ長 + IP ヘッダ長です。 – 識別フィールドは変更されません。ソフトウェアによって提供されるヘッダに応じて送信されます。 – IPv4 ヘッダチェックサムが再計算されます。 IPv6 ヘッダ – ペイロード長 = MSS + TCP ヘッダ長 + IP 拡張ヘッダ長です。
後続のパケット	1. シーケンス数は更新されます。MSS 値が、前のセグメントのシーケンス数値に追加されます。 2. セットしている場合、FIN フラグおよび PSH フラグがクリアされます。 3. TCP チェックサムが再計算されます。	IPv4 ヘッダ – 合計の長さ = MSS + TCP ヘッダ長 + IP ヘッダ長です。 – 識別フィールド = 前の識別フィールド + 1 – IPv4 ヘッダチェックサムが再計算されます。 IPv6 ヘッダ – ペイロード長 = MSS + TCP ヘッダ長 + IP 拡張ヘッダ長です。
最後のパケット	1. シーケンス数は更新されます。MSS 値が、前のセグメントのシーケンス数値に追加されます。 2. FIN フラグおよび PSH フラグが元のヘッダでセットされた場合、これらのフラグがセットされます。 3. TCP チェックサムが再計算されます。	IPv4 ヘッダ – 合計の長さ = 残りのペイロード + TCP ヘッダ長 + IP ヘッダ長です。 – 識別フィールド = 前の識別フィールド + 1 – IPv4 ヘッダチェックサムが再計算されます。 IPv6 ヘッダ – ペイロード長 = 残りのペイロード長 + TCP ヘッダ長 + IP ヘッダ長です。

分割されたパケットのヘッダおよびペイロードフィールド

分割後、分割されたパケットは *表 502：TSO：TCP ヘッダおよび IP ヘッダフィールド*に示されたもの以外のヘッダフィールドに親 TCP パケットと同じヘッダを使用します。*図 778：分割されたパケットのヘッダおよびペイロードフィールド*では、分割されたパケットのヘッダフィールドに同じヘッダがどのように使用されるかを示しています。

アプリケーションは、分割されるパケットの先頭ディスクリプタのバッファ 1 でヘッダを作成し、先頭ディスクリプタ（FD = '1'）の TDES2 にヘッダの長さを提供します。FD ビットがセットされている場合、DMA は TDES0 が指定しているヘッダバッファからヘッダを読み出します。先頭ディスクリプタのバッファ 2 は、後続のディスクリプタのペイロードおよび TDES0 および TDES1 に使用できます。後続のディスクリプタ（FD = '0'）では、TDES0 および TDES1 が指定しているアドレスは同じパケットのペイロードバッファアドレスとして扱われます。

図 778. 分割されたパケットのヘッダおよびペイロードフィールド



コンテキストディスクリプタシーケンス

コンテキストディスクリプタは、セグメンテーションの最大セグメントサイズ（MSS）を提供できます。アプリケーションは、対応する TCP パケットに使用する通常ディスクリプタの前に、コンテキストディスクリプタを提供しなければなりません。アプリケーションが新しい MSS を提供する必要がある場合、新しい MSS 値で分割するパケットの最初の通常ディスクリプタの前に、ディスクリプタリストでコンテキストディスクリプタを作成する必要があります。コンテキストディスクリプタの MSS 値は、コンテキストディスクリプタの TDES3 の TCMSSV ビットがセットされ、OSTC ビットがリセットされている場合（*セクション 58.10.3：送信ディスクリプタ*を参照）のみ有効です。

アプリケーションが有効な MSS 値を持つコンテキストディスクリプタを提供すると、DMA は MSS 値を内部で保存し、TDES3 通常ディスクリプタの TSE ビットを通じて TSO が有効になっているすべての後続のパケットにこの値を使用します。

アプリケーションがパケットの中間（パケットの先頭と最終ディスクリプタの間）にコンテキストディスクリプタを配置する場合、DMA は次の内容を実行します。

1. DMA がコンテキストを無視して、ディスクリプタをクローズします。
2. DMA がディスクリプタステータスでエラーを示します。
3. CDEE ビットが DMA チャネルに対応する割り込み有効レジスタでセットされている場合（*チャネル割り込み有効レジスタ（ETH_DMACIER）*を参照）、DMA は割り込みを生成します。

アプリケーションは、DMA チャネルに対応するステータスレジスタの CDE ビットを通じて割り込みステータスを読み出せます（*チャネルステータスレジスタ（ETH_DMACSR）*を参照）。

58.5.7 ループバック

MAC は、送信したパケットのレシーバへのループバックをサポートします。デフォルトでは、MAC のループバック機能は無効ですが、*動作モード設定レジスタ（ETH_MACCR）*レジスタの LM ビットをプログラムすることによって有効にできます。

ループバック機能は、すべての PHY インタフェースに使用できます。どの PHY インタフェースが選択されているかにかかわらず、データは常に MII インタフェースでループバックされます。

58.5.8 フロー制御

ここでは、送信パスと受信パスのフロー制御について説明します。

送信フロー制御

送信フロー制御は、*Tx キューフロー制御レジスタ（ETH_MACQTxFCR）*で TFE ビットがセットされているとき有効になります。

フロー制御トリガ

送信フロー制御は、全二重モードでは送信しているポーズパケット、半二重モードではバックプレッシャを含め、リモートエンドからパケットのフローを制御します。アプリケーションは、MAC にポーズパケットを送信するように要求するか、対応する *Tx キューフロー制御レジスタ（ETH_MACQTxFCR）*で FCB ビットをセットすることでバックプレッシャを開始できます。

全二重モードでのフロー制御：ポーズパケット制御

全二重モードでは、MAC はフロー制御に IEEE 802.3x ポーズパケットを使用します。*表 503* に、ポーズパケットのフィールドを示します。

表 503. ポーズパケットのフィールド

フィールド	説明
DA	特別なマルチキャストアドレスを含みます。
SA	MAC アドレス 0 を含みます。
タイプ	8808 を含みます。
MAC 制御 OP コード	IEEE 802.3x ポーズ制御パケットでは 0001、PFC パケットでは 0101 を含みます。
PT	*Tx キューフロー制御レジスタ（ETH_MACQTxFCR）*の PT フィールドで指定されたポーズ時間を含みます。

FCB ビットがセットされている場合、MAC は 1 つのポーズパケットを生成して送信します。ポーズパケット送信が完了した後に FCB が再びセットされた場合、ポーズ時間が完了しているかどうかにかかわらず、MAC は別のポーズパケットを送信します。ポーズ時間を延長したり、前に送信されたポーズパケットで指定された時間の前にポーズを終了するには、アプリケーションは、ポーズ時間レジスタを適切な値でプログラムしてから、FCB ビットを再びセットする必要があります。

半二重モードでのフロー制御

半二重モードでは、MAC はフロー制御に遅延メカニズム（バックプレッシャ）を使用します。アプリケーションがパケットの受信を停止するように要求すると、MAC は、パケットの受信を検出したときに 32 バイトのジャムパターンを送信します。ただし、送信フロー制御が有効な場合に限ります。これにより衝突が発生し、リモートステーションがバックオフします。アプリケーションがパケットの送信を要求した場合、バックプレッシャが有効なときでも、送信がスケジュールされ、実行されます。バックプレッシャが長時間有効なままの場合（および 16 を超える衝突イベントが連続して発生した場合）、リモートステーションは過剰衝突のために送信を中止します。

表 504 では、次のビットの設定に基づいた Tx パスでのフロー制御について説明します。

- *Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）* の EHFC ビット
- *Tx キューフロー制御レジスタ（ETH_MACQTxFCR）* の TFE ビット
- *動作モード設定レジスタ（ETH_MACCR）* の DM ビット

フロー制御はすべてのキューで同様です。

表 504. Tx MAC フロー制御

EFC	TFE	DM	説明
x	0	x	MAC トランスミッタはフロー制御またはバックプレッシャの動作を実行しません。
0	1	0	*Tx キューフロー制御レジスタ (ETH_MACQTxFCR)* のビット 0 がセットされているとき、MAC トランスミッタはバックプレッシャを実行します。
1	1	0	*Tx キューフロー制御レジスタ (ETH_MACQTxFCR)* のビット 0 がセットされているとき、MAC トランスミッタはバックプレッシャを実行します。さらに、Rx キューレベルが *Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）* のビット [10:8] によってセットされた閾値を超えるとき、MAC Tx はバックプレッシャを実行します。
0	1	1	*Tx キューフロー制御レジスタ (ETH_MACQTxFCR)* のビット 0 がセットされているとき、MAC トランスミッタはポーズパケットを送信します。
1	1	1	*Tx キューフロー制御レジスタ (ETH_MACQTxFCR)* のビット 0 がセットされているとき、MAC トランスミッタはポーズパケットを送信します。さらに、Rx キューレベルが *Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）* のビット [10:8] によってセットされた閾値を超えるとき、MAC Tx はポーズパケットを送信します。

受信フロー制御

受信パスでは、フロー制御は全二重モードでのみ機能します。ポーズパケットが半二重モードで受信される場合、パケットは通常の制御パケットとみなされます。ポーズフロー制御は、*Rx フロー制御レジスタ（ETH_MACRxFCR）*の RFE ビットをセットすることによって有効にできます。*表 505* では、次のビットの設定に基づいた Rx パスでのフロー制御について説明します。

- *Rx フロー制御レジスタ（ETH_MACRxFCR）*の RFE ビット
- *動作モード設定レジスタ（ETH_MACCR）*の DM ビット

表 505. Rx MAC フロー制御

RFE	DM	説明
0	x	MAC レシーバは受信したポーズパケットを検出しません。
1	0	MAC レシーバは受信したポーズパケットを検出しませんが、これらのパケットを制御パケットとして認識します。
1	1	MAC レシーバはポーズパケットを検出したり処理したりして、MAC トランスミッタを停止することでこれらのパケットに応答します。

次のシーケンスでは、Rx フロー制御について説明します。

1. MAC は、マルチキャストまたはユニキャストのデスティネーションアドレスに対して、受信したポーズパケットのデスティネーションアドレスをチェックします。
2. MAC は、受信パケットのタイプ（0x8808）および OP コード（0x0001：ポーズパケット）フィールドをデコードします。（ポーズパケットの）ポーズ時間をキャプチャして、トランスミッタをブロックする必要がある時間を決定します。
3. ステータスのバイトカウントが 64 バイトを示している場合と、CRC エラーがない場合、MAC は、デコードしたポーズ時間の値にスロット時間（64 バイト時間）を掛けた時間の間、あらゆるデータパケットの送信を一時停止します。

58.5.9 チェックサムオフロードエンジン

TCP や UDP などの通信プロトコルは、ネットワーク経由で送信されたデータの完全性を確認できるチェックサムフィールドを実装しています。最も広く利用されている Ethernet の用途は、IP データグラムによる TCP や UDP のカプセル化です。MAC は、チェックサム計算と送信パスへの挿入、および受信パスでのエラー検出をサポートするチェックサムオフロードエンジン（COE）を備えています。

送信チェックサムオフロードエンジン

COE モジュールは、2 種類のチェックサム計算および挿入をサポートします。チェックサムエンジンは、CIC ビット（TDES3 ビット [17:16]）をセットすることによって、パケットごとに制御できます。

注： ***TCP、UDP、または ICMP のチェックサムは、完全なパケットに対して計算された後、対応するヘッダフィールドに挿入されます。この要件のため、MAC が閾値（カットスルー）モードに設定されている場合でも、Tx FIFO は自動的にストアアンドフォワードモードで動作します。***

IP ヘッダチェックサムエンジン

IPv4 データグラムでは、ヘッダフィールドの完全性は、16 ビットのヘッダチェックサムフィールド（IPv4 データグラムの 11 番目と 12 番目のバイト）によって示されます。COE は、Ethernet パケットのタイプフィールドの値が 0x0800 であり、IP データグラムのバージョンフィールドの値が 0x4 のときに、IPv4 データグラムを検出します。入力パケットのチェックサムフィールドは、計算時には無視され、計算された値に置き換えられます。

IPv6 ヘッダには、チェックサムフィールドがありません。そのため、COE は IPv6 ヘッダフィールドを変更しません。

この IP ヘッダチェックサムの計算結果は、送信ステータス（*表 519：TDES3 通常ディスクリプタ（書き戻しフォーマット）*のビット 0）の IP ヘッダエラーステータスビットによって示されます。

TCP/UDP/ICMP チェックサムエンジン

TCP/UDP/ICMP チェックサムエンジンは、IPv4 または IPv6 ヘッダ（拡張ヘッダを含む）を処理して、カプセル化されたペイロードが TCP、UDP、ICMP のいずれかを判断します。TCP、UDP、または ICMP ペイロードのチェックサムが計算され、ヘッダ内の対応するフィールドに挿入されます。Tx COE は次の 2 つのモードで動作します。

- TCP、UDP、または ICMPv6 擬似ヘッダは、チェックサムの計算に含まれず、入力パケットのチェックサムフィールドに存在するとみなされます。このエンジンは、チェックサムの計算にチェックサムフィールドを含め、このチェックサムフィールドを最終的に計算されたチェックサムに置き換えます。
- エンジンがチェックサムフィールドを無視し、TCP、UDP、または ICMPv6 擬似ヘッダデータをチェックサムの計算に含め、このチェックサムフィールドを最終的な計算値で上書きします。

注： ***ICMP-over-IPv4 パケットの場合、擬似ヘッダは定義されていないので、ICMP パケット内のチェックサムフィールドは、どちらのモードでも常に 0x0000 でなければなりません。0x0000 に等しくない場合、正しくないチェックサムがパケットに挿入される可能性があります。***

この操作の結果は、送信ステータスベクタのペイロードチェックサムエラーステータスビット（*表 519：TDES3 通常ディスクリプタ（書き戻しフォーマット）*のビット 12）によって示されます。パケットが FIFO に書き込まれているパケットの終了（EOP）のなしでストアアンドフォワードモードで MAC トランスミッタエンジンに転送されたことを検出した場合、または IP ヘッダのペイロード長フィールドで示されたバイト数を受信する前にパケットが終了した場合に、このエンジンはペイロードチェックサムエラーステータスビットをセットします。パケットが、示されているペイロード長よりも長いときには、COE はそれらをスタッフバイトとして無視し、エラーは報告されません。このエンジンが最初のタイプのエラーを検出したとき、TCP、UDP、または ICMP ヘッダを変更しません。2 番目のタイプのエラーの場合、計算されたチェックサムを対応するヘッダフィールドに挿入します。

表 506 では、パケットタイプに基づいて送信チェックサムオフロードエンジンによってサポートされる機能を説明します。MAC がチェックサムを挿入しないときは、表に「いいえ」と示されます。

参考資料

表 506. 異なるパケットタイプの送信チェックサムオフロードエンジンの機能

パケットタイプ	ハードウェア IP ヘッダチェックサム挿入	ハードウェア TCP/UDP チェックサム挿入
非 IPv4 または非 IPv6 パケット	いいえ	いいえ
TCP、UDP、または ICMP での IPv4	はい	はい
IP オプションがある IPv4 パケット（IP ヘッダが 20 バイトを超える）	はい	はい
IPv4 の断片のパケット	はい	いいえ
メインまたは拡張ヘッダで次のヘッダフィールドを持つ IPv6 パケット： – ホップバイホップオプション（IPv6 メインヘッダ） – ホップバイホップオプション（IPv6 拡張ヘッダ） – 転送先オプション – ルーティング（残りセグメント 0） – ルーティング（残りセグメント 1 以上） – TCP、UDP、または ICMP – 認証 – メインまたは拡張ヘッダでのその他の次のヘッダフィールド	 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外	 – はい – いいえ – はい – いいえ – いいえ – はい – はい – いいえ
オプションフィールドのある TCP ヘッダを持つ IPv4 パケット	はい	はい
IPv4 トンネル： – IPv4 トンネルでの IPv4 パケット – IPv4 トンネルでの IPv6 パケット	 – はい（IPv4 トンネルヘッダ） – はい（IPv4 トンネルヘッダ）	 – いいえ – いいえ
IPv6 トンネル： – IPv6 トンネルでの IPv4 パケット – IPv6 トンネルでの IPv6 パケット	 – 適用外 – 適用外	 – いいえ – いいえ
802.3ac タグのある IPv4 パケット（有効時 C-VLAN タグまたは S-VLAN タグ付き）	はい	はい
802.3ac タグのある IPv6 パケット（有効時 C-VLAN タグまたは S-VLAN タグ付き）	適用外	はい
セキュリティ機能のある IPv4 フレーム（カプセル化されたセキュリティペイロードなど）	はい	いいえ
セキュリティ機能のある IPv6 フレーム（カプセル化されたセキュリティペイロードなど）	適用外	いいえ

受信チェックサムオフロードエンジン

*受信 TCP/IP チェックサムチェックの有効化*オプションを選択して、*動作モード設定レジスタ（ETH_MACCR）*の IPC ビットをセットすることで、受信チェックサムオフロードエンジン（Rx COE）を有効化できます。このオプションを選択すると、データの完全性チェックのために、受信 Ethernet パケットの IPv4 または IPv6 パケットが検出され、処理されます。MAC レシーバは、受信した Ethernet パケットのタイプフィールドの値が 0x0800 か 0x86DD かをチェックすることによって、そのパケットが IPv4 か IPv6 かを識別します。この識別は、シングル VLAN タグ付きパケットに適用できます。*ダブル VLAN 処理の有効化*オプションが選択され、*VLAN タグレジスタ（ETH_MACVTR）*の EDVLP ビットがセットされている場合、ダブル VLAN タグ付きパケットにも適用できます。

Rx COE は、IPv4 ヘッダのチェックサムを計算して、受信した IPv4 ヘッダのチェックサムと一致するかどうかを確認します。この操作の結果（成功または失敗）は、受信ステータスワードへの挿入のための RFC モジュールに付与されます。示されているペイロードタイプ（Ethernet タイプフィールド）と IP ヘッダバージョン間に不一致があった場合、または、受信パケットのバイト数が、IPv4 ヘッダの長さフィールドで示されているバイト数より少ないとき（または、IPv4 または IPv6 ヘッダで使用可能なスペースが 20 バイト未満のとき）には、IP ヘッダエラービットがセットされます。

このエンジンは、受信した IP データグラム（IPv4 または IPv6）のペイロードが TCP、UDP、または ICMP のいずれであるかも識別して、TCP、UDP、または ICMP 仕様での定義に従って、それぞれのペイロードのチェックサムを計算します。このエンジンでは、チェックサムの計算に、TCP、UDP または ICMPv6 擬似ヘッダバイトも含まれ、受信したチェックサムフィールドの値が計算値に一致するかどうかをチェックします。この操作の結果は、受信ステータスワードのペイロードチェックサムエラービットとして与えられます。このステータスビットは、TCP、UDP、または ICMP ペイロードが IP ヘッダで示された期待されたペイロード長に一致しない場合にもセットされます。

*表 507：異なるパケットタイプの受信チェックサムオフロードエンジンの機能*では、パケットタイプに基づいて Rx COE によってサポートされる機能を説明します。IP パケットのペイロードが処理されない（表で「いいえ」と示されている）場合、その情報（チェックサムエンジンがバイパスされているかにかかわらず）は受信ステータスで付与されます。

注： ***MAC は、受信した Ethernet パケットにペイロードチェックサムバイトを付加しません。***

表 507. 異なるパケットタイプの受信チェックサムオフロードエンジンの機能

パケットタイプ	ハードウェア IP ヘッダチェックサムチェック	ハードウェア TCP/UDP/ICMP チェックサムチェック
非 IPv4 または非 IPv6	いいえ	いいえ
TCP、UDP、または ICMP での IPv4	はい	はい
TCP、UDP、または ICMP 以外のプロトコルを含む IPv4 ヘッダのプロトコルフィールド	はい	いいえ
IP オプションがある IPv4 パケット（IP ヘッダが 20 バイトを超える）	はい	はい
IPv4 の断片のパケット	はい	いいえ

表 507. 異なるパケットタイプの受信チェックサムオフロードエンジンの機能（続き）

パケットタイプ	ハードウェア IP ヘッダチェックサムチェック	ハードウェア TCP/UDP/ICMP チェックサムチェック
メインまたは拡張ヘッダで次のヘッダフィールドを持つ IPv6 パケット： – ホップバイホップオプション（IPv6 メインヘッダ） – ホップバイホップオプション（IPv6 拡張ヘッダ） – 転送先オプション – ルーティング（残りセグメント 0） – ルーティング（残りセグメント 1 以上） – TCP、UDP、または ICMP – メインまたは拡張ヘッダでのその他の次のヘッダフィールド	– 適用外 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外 – 適用外	– はい – いいえ – はい – はい – いいえ – はい – いいえ
オプションフィールドのある TCP ヘッダを持つ IPv4 パケット	はい	はい
IPv4 トンネル： – IPv4 トンネルでの IPv4 パケット – IPv4 トンネルでの IPv6 パケット	– はい（IPv4 トンネルヘッダ） – はい（IPv4 トンネルヘッダ）	– いいえ – いいえ
IPv6 トンネル： – IPv6 トンネルでの IPv4 パケット – IPv6 トンネルでの IPv6 パケット	– 適用外 – 適用外	– いいえ – いいえ
802.3ac タグのある IPv4 パケット （有効時 C-VLAN タグまたは S-VLAN タグ付き）	はい	はい
802.3ac タグのある IPv6 パケット （有効時 C-VLAN タグまたは S-VLAN タグ付き）	適用外	はい
セキュリティ機能のある IPv4 フレーム （カプセル化されたセキュリティペイロードなど）	はい	いいえ
セキュリティ機能のある IPv6 フレーム （カプセル化されたセキュリティペイロードなど）	適用外	いいえ

参考資料

58.5.10 MAC 管理カウンタ

MAC 管理カウンタ（MMC）モジュールのカウンタは、CSR モジュールのレジスタアドレス空間の拡張とみなすことができます。MMC モジュールは、受信および送信パケットの統計を集めるために、一連のレジスタを維持します。このレジスタセットには、レジスタの動作を制御する制御レジスタ、生成された割り込み（受信および送信）を含む 2 つの 32 ビットレジスタ、および割り込みレジスタ（受信および送信）のマスクを含む 2 つの 32 ビットレジスタがあります。これらのレジスタは、CSR レジスタのアクセスと同じ方法で AHB スレーブインタフェースを通じてアプリケーションからアクセス可能です。これらのレジスタの構造を*セクション 58.11.4 : Ethernet MAC および MMC レジスタ*に示します。

MMC カウンタはフリーランニングです。カウンタを開始するための個別の有効化はありません。対応するパケットが受信または送信されると、特定の MMC カウンタがカウントを開始します。

制御レジスタに加えて、2 セットのレジスタが実装されています。

- 衝突、エラーおよび良好なパケットカウンタに対応して使用される 6 つのレジスタ
- LPI モード遷移を記録する 4 つのレジスタ

58.5.11 MAC によって生成される割り込み

さまざまなイベントの結果として、MAC から割り込みが生成されることがあります。これらの割り込みイベントは、eth_sbd_intr_it 信号で DMA でのイベントと組み合わされます。MAC 割り込みはレベルタイプで、アプリケーションやソフトウェアによってクリアされるまで、割り込みはアサート（ハイ）されたままになります。

*割り込みステータスレジスタ（ETH_MACISR）*は、MAC からの割り込みの原因となるイベントを記述します。MAC 割り込みはデフォルトで有効にされます。*割り込み有効レジスタ（ETH_MACIER）*の対応するマスクビットをセットすることによって、各イベントが eth_sbd_intr_it 信号で割り込みをアサートしないようにすることができます。

割り込みレジスタビットは、イベントが報告されるブロックを示すだけです。割り込みをクリアするには、対応するステータスレジスタと他のレジスタを読み出す必要があります。

58.5.12 MAC および MMC レジスタの説明

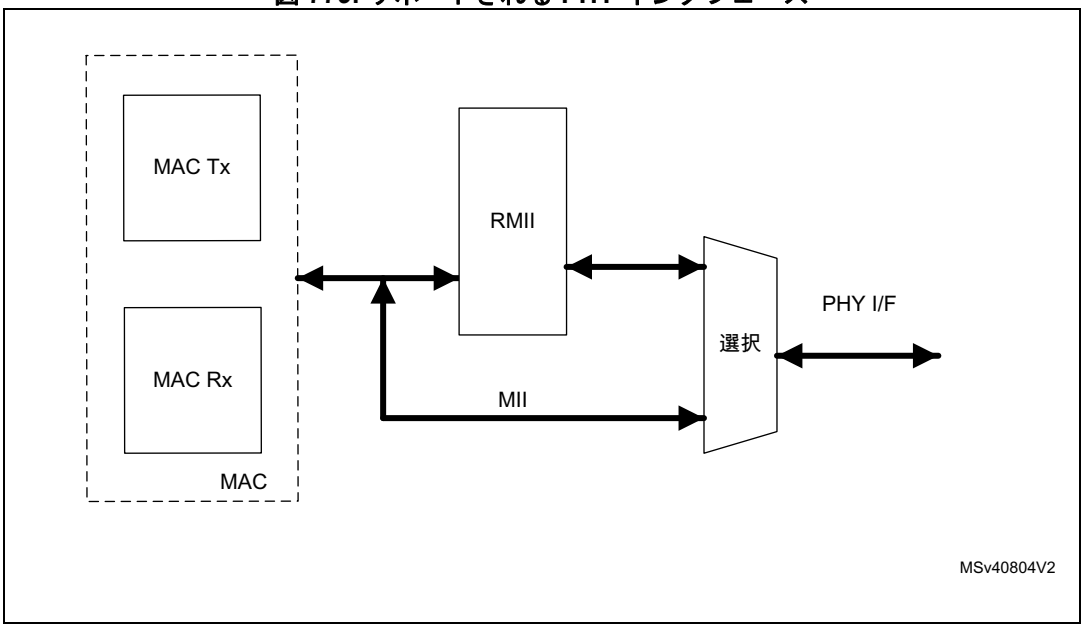
*セクション 58.11.4 : Ethernet MAC および MMC レジスタ*を参照してください。

参考資料

58.6 Ethernet機能の説明：PHY インタフェース

Ethernet ペリフェラルは、複数の PHY インタフェースをサポートします。ルートインタフェースは MII インタフェースです。その他すべてのインタフェースは、*図 779* に示すように、これから生成されます。

図 779. サポートされる PHY インタフェース



このセクションでは､PHY 制御インタフェースおよび異なる PHY インタフェースに使用される SMA モジュールについて説明します。以下のセクションが含まれています。

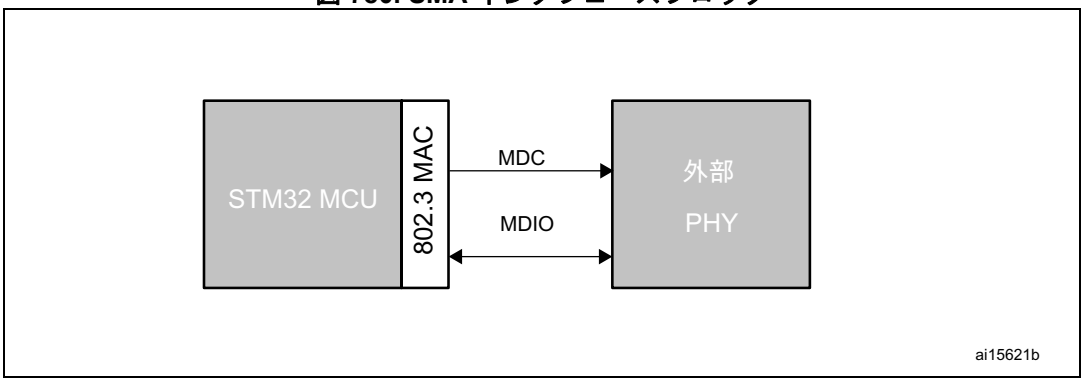
- *ステーション管理エージェント（SMA）*
- *メディア独立インタフェース（MII）*
- *減少メディア独立インタフェース信号（RMII）*

58.6.1 ステーション管理エージェント（SMA）

アプリケーションは、ステーション管理エージェント（SMA）モジュールを通じて PHY レジスタにアクセスできます。SMA には、2 線ステーション管理インタフェース（MIM）が搭載されています。

SMA モジュールは、最大 32 の PHY へのアクセスをサポートします。アプリケーションは 32 の PHY の 32 個のレジスタのうち 1 個に対応できます。一度にアドレス指定できるのは、1 つの PHY 内の 1 つのレジスタだけです。*図 780* に示すように、アプリケーションは PHY に制御データを送信して、SMA モジュールを通じて PHY からステータス情報を受信します。

図 780. SMA インタフェースブロック



SMA の機能概要

MAC は、MDC クロックに関する書き込みまたは読み出し動作の管理を開始します。MDC クロックは、CSR クロックから生成されます。分周比は、*MDIO アドレスレジスタ（ETH_MACMDIOAR）*レジスタのクロック範囲設定に依存します。MDC クロックは次のように選択されます。

表 508. MCD クロックの選択

選択	CSR クロック	MDC クロック
0000	60～100 MHz	CSR クロック/42
0001	100～150 MHz	CSR クロック/62
0010	20～35 MHz	CSR クロック/16
0011	35～60 MHz	CSR クロック/26
0100	150～250 MHz	CSR クロック/102
0101	250～300 MHz	CSR クロック/124
0110、0111	予約済み	-

MAC と PHY のデータ交換は、mdi_i, mdo_o および mdo_oe 信号を通じて実行されます。信号グループは、トライステートバッファを通じて伝送され、PHY に接続された MDIO ラインとして出力されます。

次の図は、MDIO パケットでのパケットの構造を示し、*表 509* ではパケットフィールドの詳細な説明を提示します。

図 781. MDIO パケットの構造

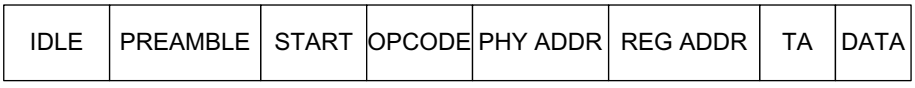


表 509. MDIO パケットフィールドの説明

フィールド	説明
IDLE	MDIO ラインはトライステートです。ETH_MDC にクロックはありません。

表 509. MDIO パケットフィールドの説明

PREAMBLE	値 1 の 32 個の連続ビット
START	パケットの開始は 2'b01 です。
OPCODE	読み出しで 2'b10、書き込みで 2'b01 です。
PHY ADDR	32 の PHY の 1 つに対して 5 ビットアドレス選択です。
REG ADDR	選択された PHY でのレジスタアドレスです。
TA	ターンアラウンドは、読み出しで 2'bZ0、書き込みで 2'b10 です。
DATA	任意の 16 ビット値です。書き込み動作では、MAC が MDIO を駆動します。 読み出し動作では、PHY が駆動します。

MII 管理書き込み動作

*MDIO アドレスレジスタ（ETH_MACMDIOAR）*で、ビット [3:2] が '01' に、ビット 0 が '1' にセットされているとき、MAC CSR モジュールは PHY アドレス、PHY でのレジスタアドレス、書き込みデータ（*MDIO データレジスタ（ETH_MACMDIODR）*）を SMA に転送して、PHY レジスタへの書き込み動作を開始します。この時点で、SMA モジュールは MII 仕様（IEEE 802.3-2002 仕様、*セクション 22.2.4.5* に準拠）で指定された管理パケットフォーマットを使用して、MII 管理インタフェースで書き込み動作を開始します。

SMA モジュールが書き込み動作を開始したとき、書き込みデータパケットが MDIO ラインに送信されます。MAC はパケットの全送信時間中、MDIO ラインを駆動します。ビジービットは、書き込み動作が完了するまでハイにセットされます。この時間（ビジービットがハイになっている時間）中、CSR は、*MDIO アドレスレジスタ（ETH_MACMDIOAR）*または *MDIO データレジスタ（ETH_MACMDIODR）*に実行される書き込み動作を無視します。書き込み動作が完了したとき、SMA モジュールは CSR に通知し、CSR はビジービットをリセットします。書き込み動作のパケットフォーマットは、以下のとおりです。

図 782. 書き込みデータパケット

IDLE	PREAMBLE	START	OPCODE	PHY ADDR	REG ADDR	TA	DATA	IDLE
Z	1111 ~ 11	01	01	AAAAA	RRRRR	10	DDD ~ DDD	Z

MSv40807V1

MII 管理読み出し動作

*MDIO アドレスレジスタ（ETH_MACMDIOAR）*で、ビット [3:2] が '11' に、ビット 0 が '1' にセットされているとき、MAC CSR モジュールは PHY アドレスおよび PHY でのレジスタアドレスを SMA に転送して、PHY レジスタでの読み出し動作を開始します。この時点で、SMA モジュールは MII 仕様（IEEE 802.3-2002 仕様、*セクション 22.2.4.5* に準拠）で指定された管理パケットフォーマットを使用して、MII 管理インタフェースで読み出し動作を開始します。

SMA モジュールが MDIO で読み出し動作を開始すると、CSR は、この時間（ビジービットがハイになっている時間）中、*MDIO アドレスレジスタ（ETH_MACMDIOAR）*または *MDIO データレジスタ（ETH_MACMDIODR）*レジスタへの書き込み動作を無視し、MCI インタフェースでエラーがない状態でトランザクションを完了します。読み出し動作が完了したとき、SMA は CSR に通知します。CSR は、ビジービットをリセットし、PHY から読み出したデータで *MDIO データレジスタ（ETH_MACMDIODR）*を更新します。MAC は、PHY が MDIO ラインを駆動し

ているデータフィールド中を除くフレームの全時間中、MDIO ラインを駆動します。アプリケーションから PHY への通信に関する詳細については、IEEE 802.3z、1000BASE Ethernet の調整サブレイヤおよびメディア独立インタフェースの仕様のセクションを参照してください。

読み出し動作のパケットフォーマットは、以下のとおりです。

図 783. 読み出しデータパケット

IDLE	PREAMBLE	START	OPCODE	PHY ADDR	REG ADDR	TA	DATA	IDLE
Z	1111 ~ 11	01	10	AAAAA	RRRRR	Z0	DDD ~ DDD	Z

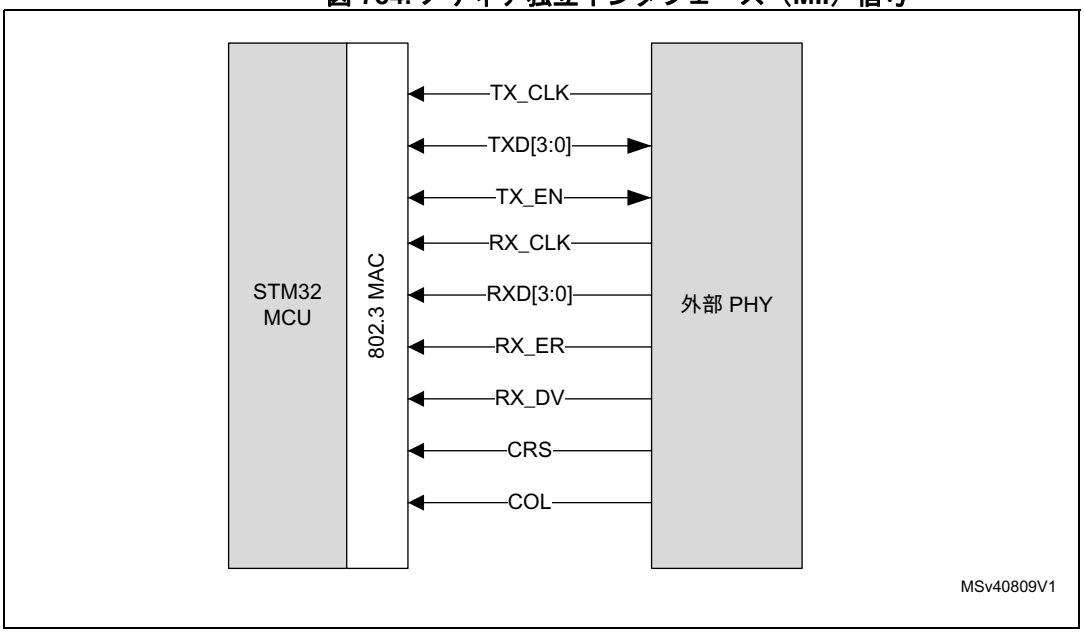
MSv40808V1

58.6.2 メディア独立インタフェース（MII）

メディア独立インタフェース（MII）は、10 Mbit/s および 100 Mbit/s でのデータ転送における MAC サブレイヤと PHY 間の相互接続を定義します。

MII 信号を*図 784：メディア独立インタフェース（MII）信号*に示します。

図 784. メディア独立インタフェース（MII）信号



- TX_CLK：Tx データ転送の基準タイミングを供給する連続クロックです。公称周波数は、10 Mbit/s で 2.5 MHz で、100 Mbit/s で 25 MHz です。
- RX_CLK：Rx データ転送の基準タイミングを供給する連続クロックです。公称周波数は、10 Mbit/s で 2.5 MHz で、100 Mbit/s で 25 MHz です。
- TX_EN：MAC が送信のために MII 上にニブルを提示していることを示す送信イネーブル信号です。プリアンブルの最初のニブル（TX_CLK）に同期してアサートされなければならず、送信されるすべてのニブルが MII に提示されるまで、アサート状態が維持されなければなりません。
- TXD[3:0]：送信データです。

  TXD は、MAC サブレイヤによって同期して駆動される 4 つのデータ信号の束であり、TX_EN 信号のアサーションによって有効なデータと認められます。TXD[0] は最下位ビットであり、TXD[3] が最上位ビットです。TX_EN がネゲートされている場合、送信データは PHY に何の影響も与えません。
- CRS：キャリアセンスです。

  この信号は、送信または受信メディアがアイドルでないとき、PHY によってアサートされます。送信メディアと受信メディアの両方ともアイドルのときには、PHY によってネゲートされなければなりません。PHY は、衝突条件の全期間を通じて、CS 信号がアサートされたままであることを保証する必要があります。この信号は、Tx および Rx クロックに同期して遷移する必要はありません。全二重モードでは、この信号の状態は MAC サブレイヤに影響を与えません。
- COL：衝突検出信号です。

  メディア上で衝突が検出された場合、PHY によってこの信号がアサートされなければならず、衝突条件が続いている間、アサート状態が保たれなければなりません。この信号は、Tx および Rx クロックに同期して遷移する必要はありません。全二重モードでは、この信号の状態は MAC サブレイヤに影響を与えません。
- RXD[3:0]：受信データです。

  RXD は、PHY によって同期して駆動される 4 つのデータ信号の束であり、RX_DV 信号のアサーションによって有効なデータと認められます。RXD[0] は最下位ビットであり、RXD[3] が最上位ビットです。RX_EN がネゲートされ、RX_ER がアサートされている間、PHY から特定の情報を転送するために特定の RXD[3:0] 値が使用されます（*表 510* を参照）。
- RX_DV：受信データ有効です。

  この信号は、PHY が MII 上に受信のために回収され、デコードされたニブルを提示していることを示します。フレームの最初の回収ニブルと同期してアサートされなければならず（RX_CLK）、最後の回収ニブルまでアサート状態を維持しなければなりません。最後のニブル後の最初のクロックの前にネゲートされなければなりません。フレームを正しく受信するためには、RX_DV 信号は、SFD フィールドよりも遅れることなく開始して、フレームを取り込む必要があります。
- RX_ER：受信エラーです。

  この信号は、フレーム内のどこかでエラーが検出されたことを MAC サブレイヤに示すために、1 つ以上のクロック周期（RX_CLK）の間、アサートされなければなりません。このエラー条件は、*表 510* での説明のように、RX_DV ビットのアサートにより有効とされなければなりません。

表 510. RX インタフェース信号のエンコーディング

RX_DV	RX_ERR	RXD[3:0]	説明
0	0	0000 から 1111 まで	通常のフレーム間
0	1	0000	通常のフレーム間
0	1	0001 から 1101 まで	予約済み
0	1	1110	不正キャリアの表示
0	1	1111	予約済み

表 510. RX インタフェース信号のエンコーディング（続き）

RX_DV	RX_ERR	RXD[3:0]	説明
1	0	0000 から 1111 まで	通常のデータ受信
1	1	0000 から 1111 まで	データ受信エラー

58.6.3 減少メディア独立インタフェース信号（RMII）

減少メディア独立インタフェース（RMII）仕様は、Ethernet PHY と STM32 マイクロコントローラのピン数を削減します。IEEE 802.3u に従って、MII はデータおよび制御のために 16 本のピンを備えています。RMII 仕様では、ピン数を 7 本に削減します。

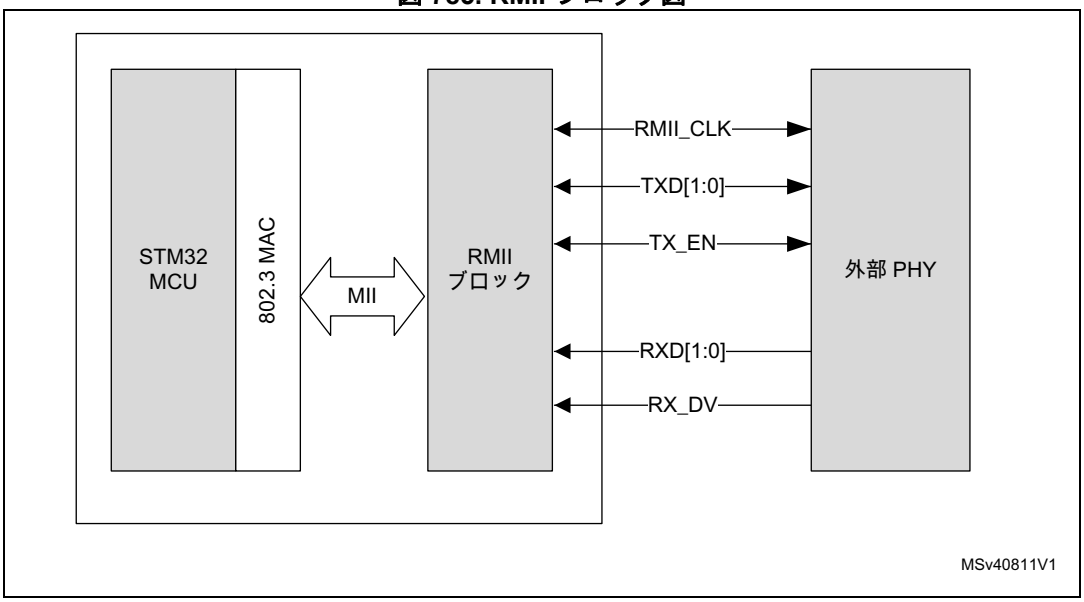
Ethernet ペリフェラルの一部、RMII モジュールは MAC 出力でインスタンス化されます。これは、MAC の MII を RMII に変換するのに役立ちます。RMII ブロックは、次のような特性を備えています。

- 10 Mbps および 100 Mbps の動作速度をサポートします。1000 Mbps 動作はサポートしていません。
- 2 つのクロック基準を外部から供給することで、独立した 2 ビット幅の送信および受信パスを提供します。

RMII ブロック図

*図 785：RMII ブロック図*では、MAC と RMII PHY に関連する RMII ブロックの位置を示しています。RMII ブロックは MAC の前に配置され、MII 信号を RMII 信号に変換します。

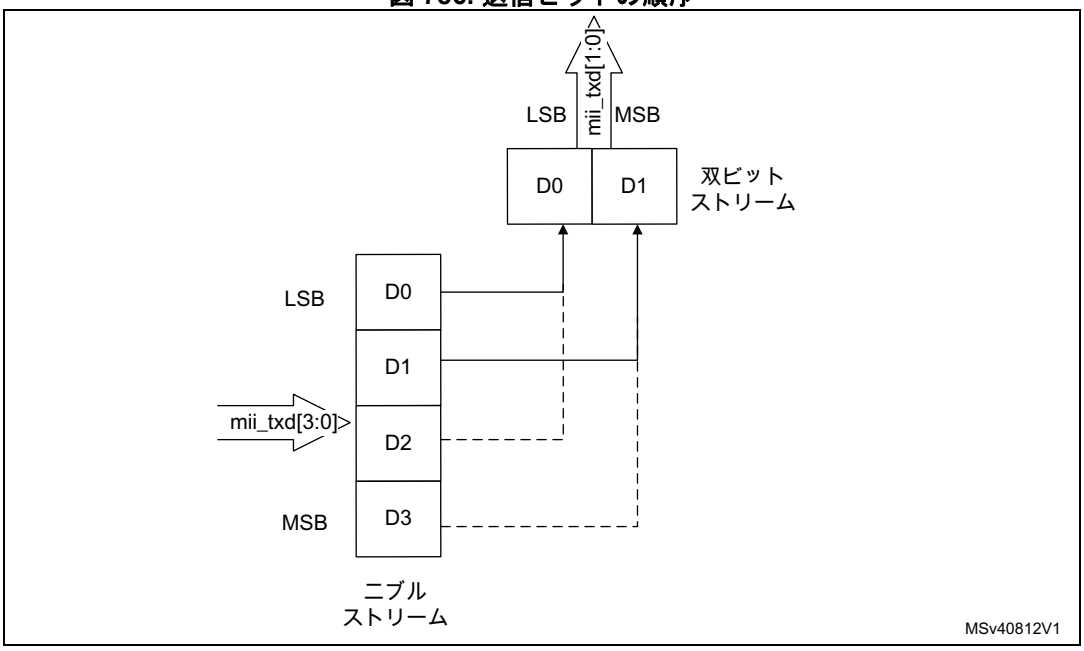
図 785. RMII ブロック図



送信ビットの順序

MII インタフェースからの各ニブルは、RMII インタフェースで一度に 2 ビット（双ビット）ずつ、*図 786：送信ビットの順序* に示されている双ビット送信の順序で送信する必要があります。下位ビット（D1 と D0）が最初に送信され、次に上位ビット（D2 と D3）が送信されます。

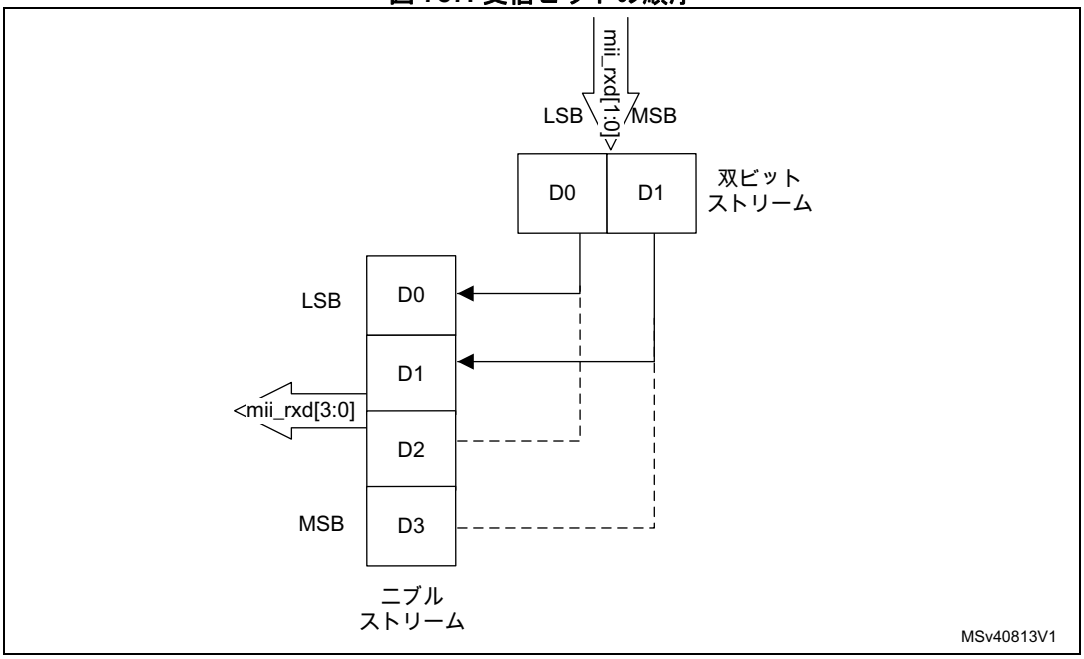
図 786. 送信ビットの順序



受信ビットの順序

各ニブルは、RMII インタフェースから受信した双ビットから MII インタフェースに、*図 787 : 受信ビットの順序* に示されているニブル送信順序で送信されます。下位ビット（D1 と D0）が最初に受信され、次に上位ビット（D2 と D3）が受信されます。

図 787. 受信ビットの順序



58.7 Ethernet 低電力モード

58.7.1 省電力型 Ethernet

省電力型 Ethernet（EEE）は、物理層のファミリに合わせた MAC サブレイヤの低電力アイドル（LPI）モードでの動作を可能にする動作モードです。EEE モードは、100 Mbps での動作をサポートします。

LPI モードでは、データが送受信されていないときに、通信デバイス機能の一部をオフに切り替えることで電力削減を実現できます。MAC は、システムの LPI モードへの移行または終了を制御し、これを PHY インタフェースに伝えます。

EEE は、リンクパートナーが EEE をサポートしているか判定してから、両方のデバイスに共通するパラメータのセットを選択するために使用できるネゴシエーション方式を指定します。

注： ***MAC が RMII シングル PHY インタフェースを使用するよう設定されている場合、EEE 機能はサポートされません。MAC が複数の PHY インタフェースをサポートしていても、MAC が MII インタフェースで動作する場合のみ EEE モードを有効にする必要があります。***

LPI モードは全二重モードでのみサポートされます。

送信パス機能

送信パスでは、ソフトウェアは *LPI 制御ステータスレジスタ（ETH_MACLCSR）* の LPIEN ビットをセットして、MAC に送信を停止して LPI プロトコルを開始するよう要求しなければなりません。

PHY の LPI 状態を終了させるためには、MAC は以下のタスクを実行します。

1. LPI パターンの送信を停止し、IDLE パターンの送信を開始します。
2. LPI TW TIMER を開始します。
   MAC は、PHY に指定されたウェイクアップ時間が終了するまで、送信を開始できません。自動ネゴシエーションウェイクアップの間隔は、*LPI 制御ステータスレジスタ（ETH_MACLCSR）* の TWT フィールドでプログラムされます。
3. LPI 終了ステータス（*LPI 制御ステータスレジスタ（ETH_MACLCSR）* の TLPIEX ビット）を更新して、割り込みを生成します。

LPI モードのプログラミングガイドラインについては、*セクション：Tx LPI モードの移行と終了*を参照してください。

Tx パスでの LPI モードの自動移行／終了

MAC トランスミッタは、一定時間アイドル状態であるか、または転送するパケットがあるかどうかに基づいて、自動的に LPI モードの移行／終了を行うようにプログラムできます。これらのモードは、*LPI 制御ステータスレジスタ（ETH_MACLCSR）* で有効化して制御されます。

LPI 制御ステータスレジスタ（ETH_MACLCSR） の LPITXA（ビット 19）および LPITXEN（ビット 16）がセットされているとき、MAC 送信パス（MTL レイヤおよび DMA レイヤを含む）がアイドル状態である場合、MAC トランスミッタは LPI IDLE 状態に移行します。Tx パスで何らかの機能（DMA、MTL または MAC）がパケット転送の開始によってアイドル状態ではなくなると、すぐに MAC トランスミッタは LPI IDLE 状態を終了し、LPITXEN ビットをクリアします。

上記に加えて、ビット [20]（LPIATE）もセットされているとき、*LPI 移行タイマレジスタ（ETH_MACLETR）* の値で示された時間、送信パスがアイドル状態（動作なし）のままである場合のみ、MAC トランスミッタは LPI IDLE 状態に移行します。このモードでも、何らかの機能がアイドル状態ではなくなると、すぐに MAC トランスミッタは LPI IDLE 状態を終了します。ただし、MAC が再びアイドルになったとき、ソフトウェアの介入がなくても LPI IDLE 状態に再移行できるように、LPITXEN ビットはクリアされずアクティブなままとなります。

LPIATE および LPITXA ビットの両方がクリアされた場合、LPITXEN ビットをプログラムすることで、LPI IDLE 状態の移行と終了を直接制御できます。

受信パス機能

受信パスでは、PHY が LPI 状態に移行するためにリンクパートナーから信号を受信すると、PHY は LPI モードに移行し、MAC が *LPI 制御ステータスレジスタ（ETH_MACLCSR）* の RLPIEN ビットを更新して、割り込みを即座に生成します。

PHY が LPI 状態を終了するためにリンクパートナーから信号を受信すると、PHY は LPI モードを終了し、MAC が *LPI 制御ステータスレジスタ（ETH_MACLCSR）* の RLPIEX ビットを更新します。割り込みが即座に生成されます。

LPI 割り込み

Tx 側または Rx 側が LPI 状態の移行時または終了時に、MAC は LPI 割り込みを生成します。LPI 割り込みは、*LPI 制御ステータスレジスタ（ETH_MACLCSR）* を読み出すことによってクリアできます。

サイドバンド信号 lpi_intr_o が割り込みとともに生成されます。この信号はウェイクアップメカニズムによって使用されます。これは、pmt_intr_o 信号（*セクション：PMT 割り込み* を参照）と論理和がとられ、EXTI ペリフェラル（ライン 86）に紐づけられます。

58.7.2 電源管理

電源管理（PMT）ブロックは、ネットワーク（リモート）ウェイクアップパケットとマジックパケットの受信をサポートします。PMT ブロックは、MAC が受信するリモートウェイクアップパケットおよびマジックパケットに対して割り込みを生成します。

PMT ブロックでパワーダウンモードが有効になると、MAC はすべての受信パケットをドロップし、Rx FIFO やアプリケーションにパケットを転送しません。MAC は、マジックパケットまたはリモートウェイクアップパケットが受信され、対応する検出機能が有効なときのみ、パワーダウンモードから抜けます。

PMT ブロックは、MAC の受信パスで使用できます。両方のタイプの電源管理パケット（リモートウェイクアップパケットおよびマジックパケット）を選択できます。*PMT 制御ステータスレジスタ（ETH_MACPCSR）* の RWKPKTEN および MGKPKTEN ビットをセットすると、電源管理イベントを生成できます。

以下に PMT ブロックレジスタを示します。

- *PMT 制御ステータスレジスタ（ETH_MACPCSR）*
- *リモートウェイクアップパケットフィルタレジスタ（ETH_MACRWKPFR）*

リモートウェイクアップパケットの検出

MAC が SLEEPモードであり、*PMT 制御ステータスレジスタ（ETH_MACPCSR）* のリモートウェイクアップビットが有効なときには、リモートウェイクアップパケットの受信後、通常の動作が再開されます。

リモートウェイクアップフレームフィルタレジスタ

PMT ブロックは、さまざまな受信パケットパターンをサポートできる 16 個のプログラム可能なフィルタをサポートします。着信パケットがフィルタコマンドのアドレスフィルタを通過し、フィルタ CRC-16 が着信パターンの CRC と一致した場合、MAC はそのパケットをウェイクアップパケットと識別します。

リモートウェイクアップパケットフィルタレジスタ（ETH_MACRWKPFR） は、フィルタ管理を定義します。

- フィルタバイトマスク：チェックするパケットのバイトを指定します。
- フィルタオフセット：チェックするパケットのオフセットを指定します。
- フィルタ CRC-16

リモートウェイクアップ CRC ブロックは、フィルタ CRC-16 と比較される CRC 値を決定します。リモートウェイクアップパケットは、長さエラー、FCS エラー、ドリブルビットエラー、受信エラー、衝突のみがチェックされます。さらに、リモートウェイクアップパケットは、ラントパケットではないことを保証するためにチェックされます。リモートウェイクアップパケットの長さが 512 バイトを超える場合でも、パケットの CRC 値が有効な場合は、有効とみなされます。リモートウェイクアップパケットの検出は、リモートウェイクアップパケットが受信される

たびに、PMT 制御およびステータスレジスタで更新されます。アプリケーションへの PMT 割り込みは、PMT 制御およびステータスレジスタに対する読み出しをトリガし、リモートウェイクアップパケットの受信を決定します。

マジックパケットの検出

マジックパケットは、Advanced Micro Devices のマジックパケットテクノロジを使用する方法に基づき、ネットワーク上のスリープ状態のデバイスを起動します。MAC は、ネットワーク上の特定のノードに宛てたマジックパケットと呼ばれる特定のパケット情報を受信します。

MAC は、MAC やマルチキャストアドレス（ブロードバンドアドレスを含む）に宛てたマジックパケットのみをチェックして、これらのパケットがウェイクアップ要件を満たすかどうかを判定します。アドレスフィルタ（ユニキャストまたはマルチキャスト（ブロードキャストを含む）アドレス）を通過したマジックパケットはチェックされ、すべて 1 の 6 バイトの後にユニキャスト MAC アドレス（MAC アドレス 0 の値に一致）が 16 回現われるというリモートウェイクアップパケットデータのフォーマットを満たしているかどうかが判定されます。

アプリケーションは、*PMT 制御ステータスレジスタ（ETH_MACPCSR）*の MGKPKTEN ビットに 1 を書き込むことによって、マジックパケットウェイクアップを有効にします。PMT ブロックは、ノードに宛てられた各パケットが特定のマジックパケットパターンを持っているかどうかを常時監視します。受信された各パケットは、デスティネーションおよびソースアドレスフィールドの後に 48’hFF_FF_FF_FF_FF_FF というパターンが続くかどうかがチェックされます。次に、PMT ブロックは、パケット内の MAC アドレスが、中断や途切れがなく、16 回繰り返されているかどうかをチェックします。アドレスの 16 回の繰り返しの途中で中断がある場合、PMT ブロックは、着信パケットに 48’hFF_FF_FF_FF_FF_FF パターンが含まれているかどうかが再度スキャンされます。16 回の繰り返しはパケット内のどこにあっても構いませんが、同期ストリーム（48’hFF_FF_FF_FF_FF_FF）の後でなければなりません。デバイスは、MAC アドレスの 16 回の繰り返しが検出される限り、マルチキャストパケットも受け入れます。8'hFF の繰り返し回数が 6 を超える場合、PMT ブロックは 8'hFF が最後に 6 回繰り返された後に、MAC アドレスが、中断や途切れがなく、16 回繰り返されているかどうかをチェックします。

ノードの MAC アドレスが 48'h00_11_22_33_44_55 である場合は、MAC は、次のデータシーケンスをスキャンします。

デスティネーションアドレスとソースアドレス .....FF FF FF FF FF FF

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

…CRC

MAC は、リモートウェイクアップパケットの長さエラー、FCS エラー、ドリブルビットエラー、受信エラー、または衝突エラーのみをチェックします。さらに、リモートウェイクアップパケットは、ラントパケットではないことを保証するためにチェックされます。リモートウェイクアップパケットの長さが 512 バイトを超える場合でも、パケットの CRC 値が有効な場合は、MAC が有効とみなします。

マジックパケットの検出は、受信されたマジックパケットの *PMT 制御ステータスレジスタ（ETH_MACPCSR）*で更新されます。アプリケーションへの PMT 割り込みは、*PMT 制御ステータスレジスタ（ETH_MACPCSR）*に対する読み出しをトリガし、マジックパケットが受信されたかどうかを判定します。

PMT 割り込み

PMT 割り込み信号は、有効なリモートウェイクアップパケットが受信されたときにアサートされます。

ソフトウェアが *リモートウェイクアップパケットフィルタレジスタ（ETH_MACRWKPFR）*で PWRDWN ビットをリセットしたとき、MAC はパワーダウンモードから抜けます。このイベントは PMT 割り込みを生成しません。

EEE モードの場合、サイドバンド信号 pmt_intr_o がウェイクアップ割り込みとともに生成されます。これは、lpi_intr_o 信号（*セクション：LPI 割り込み*を参照）と論理和がとられ、EXTI ペリフェラル（ライン 86）に紐づけられます。

参考資料

58.7.3 パワーダウンおよびウェイクアップシーケンス

推奨されるパワーダウンおよびウェイクアップシーケンスは、次のとおりです。

1. 送信 DMA を無効にして、前のパケット送信の完了を待ちます。これらの送信は、送信割り込み（ETH_DMACSR レジスタのビット 0）が受信されると検出できます。
2. MAC 設定レジスタ（ETH_MACCR ）の対応するビット（0 および 1）をクリアすることによって、MAC トランスミッタと MAC レシーバを無効にします。
3. 受信 DMA が Rx FIFO からシステムメモリのすべてのパケットを空にするのを待ちます。この操作は、DMA（ETH_DMADSR）および MTL（ETH_MTLTxQDR および ETH_MTLRxQDR）の対応するデバッグレジスタビットを読み出すことで実行できます。
4. PMT レジスタの ETH_MACPCSR および ETH_MACRWKPFR を適切に設定することで、パワーダウンモードを有効にします。
5. MAC レシーバを有効にして、パワーダウンモードに移行します。
6. 有効なリモートウェイクアップパケットを受信すると、Ethernet ペリフェラルは pmt_intr_o 信号をアサートし、パワーダウンモードを終了します。
7. PMT ステータスレジスタを読み出し、割り込みをクリアしてから、システムでその他のモジュールを有効にし、通常動作を再開します。

58.8 Ethernet 割り込み

Ethernet ペリフェラルは、1 つの割り込み信号（eth_sbd_intr_it）を生成します。この信号は、さまざまなイベントの結果として発生することがあります。これらのイベントはステータスレジスタでキャプチャされ、対応する割り込み有効がセットされたときのみ割り込み信号がイベントにアサートされるように、割り込み有効が割り込みの各ソースに対して提供されます。

割り込みステータスと対応する有効レジスタは階層形式で構成されているため、ソフトウェアが割り込みイベントのソースを素早く横断して識別しやすくなっています。割り込みがアサートされると、*割り込みステータスレジスタ（ETH_DMAISR）*レジスタは割り込みイベントソースの主なブロックを示す最初のレベルになります。このレジスタは読み出し専用で、各 DMA チャネル（TX & RX ペア）、MTL、MAC に対応するビットを含みます。次に、ソフトウェアアプリケーションは、セットされているビットに対応する次のレジスタのうち 1 つ（または複数）を読み出す必要があります。

- ETH_DMACSR：チャネルステータス（*チャネルステータスレジスタ（ETH_DMACSR）*を参照）
- ETH_MTLISR：割り込みステータス（*割り込みステータスレジスタ（ETH_MTLISR）*を参照）
- ETH_MACISR：割り込みステータス（*割り込みステータスレジスタ（ETH_MACISR）*を参照）

58.8.1 DMA 割り込み

割り込みレジスタの説明

ETH_DMACSR：チャネルステータスレジスタ（*チャネルステータスレジスタ（ETH_DMACSR）*を参照）は、その TxDMA および RxDMA チャネルのすべての割り込みイベントをキャプチャします。ETH_DMACIER：チャネル割り込み有効レジスタ（*チャネル割り込み有効レジスタ（ETH_DMACIER）*を参照）には、割り込みイベントごとに対応する有効ビットが含まれます。

DMA チャネルには、通常割り込みと異常割り込みという 2 つのグループの割り込みがあります。これらはそれぞれ ETH_DMACSR レジスタのビット [15:14] によって示されます。通常グループは、パケットの通常転送中に発生するイベント（TI：送信割り込み、RI：受信割り込み、TBU：送信バッファ使用不可）に対応し、異常割り込みイベントはエラーイベントに対応します。

割り込みはキューされません。ドライバが前のイベントに対応する前に、同じ割り込みイベントが再び発生した場合、追加の割り込みは生成されません。割り込みは、複数のイベントについて 1 つしか生成されません。ドライバは、*割り込みステータスレジスタ（ETH_DMAISR）*で割り込みの原因をスキャンして、それぞれのステータスレジスタのソースをクリアしなければなりません。*割り込みステータスレジスタ（ETH_DMAISR）*のすべてのビットがクリアされたときのみ割り込みがクリアされます。

送信および受信割り込みの周期的スケジュール

システムスループットのパフォーマンスを理由に、DMA によって転送される各パケットに対して割り込み（RI および TI）を生成することは推奨されません。Ethernet ペリフェラルは、2 つの方法を使用して一定間隔で割り込みをスケジュールする柔軟性を備えています。

1. 送信されるパケットのそれぞれの「必要」数につき 1 つ、送信ディスクリプタに完了時割り込みビット（*表 514：TDES2 通常ディスクリプタ（読み出しフォーマット）*の TDES2[31]）をセットします。
2. 同様に、受信ディスクリプタの指定した間隔でのみ IOC（*表 527：RDES3 通常ディスクリプタ（読み出しフォーマット）*の RDES3[30]）ビットをセットします。こうして、システムメモリへの受信パケットの転送が完了し、そのパケット転送に使用されるディスクリプタのいずれかに IOC ビットがセットされているときは、必ず RI イベントが生成されます。

上記のほかに、割り込みタイマ（ETH_DMACRxIWTR：チャネル Rx 割り込みウォッチドッグタイマ）が提供され、受信割り込みの柔軟性の高い制御および周期的なスケジューリングを可能にします。この割り込みタイマが 0 でない値でプログラムされると、対応する完了の割り込み IOC ビット（*表 527：RDES3 通常ディスクリプタ（読み出しフォーマット）*の RDES3[30]）がセットされていないため、Rx DMA が受信割り込みをアサートせずに、受信パケットのシステムメモリへの転送を完了すると、割り込みタイマが起動します。このタイマがプログラムされた値

に従ってタイムアウトになると、RI ビットがセットされ、ETH_DMACIER レジスタ（*チャネル割り込み有効レジスタ（ETH_DMACIER）*を参照）で対応する RIE が有効な場合、割り込みがアサートされます。ディスクリプタの IOC がセットされたパケット転送に対して RI がセットされている場合、タイマは終了する前に停止されてクリアされます。RI イベントが生成されずに次のパケット転送が完了した後、タイマは自動的に再び起動します。

チャネル転送完了割り込み

送信転送完了割り込み（TI）および受信転送完了割り込み（RI）は、チャネルステータスレジスタ（*チャネルステータスレジスタ（ETH_DMACSR）*）に反映されます。Tx DMA チャネルが IOC ビット（完了時割り込み - TDES2[31]）がセットされたディスクリプタをクローズすると必ず TI ビットがセットされます。同様に、Rx DMA チャネルが LD ビットがセットされたディスクリプタをクローズすると必ず RI ビットがセットされ、そのパケットの転送に使用されるディスクリプタのいずれかで、IOC ビット（完了時割り込み有効 - RDES3[30]）がセットされます。

割り込み信号は、チャネル割り込み有効レジスタ（*チャネル割り込み有効レジスタ（ETH_DMACIER）*）で対応する割り込みが有効なときのみ、転送完了割り込みに対してアサートされます。

RI/TI 割り込みの動作は、ETH_DMAMR レジスタ（*DMA モードレジスタ（ETH_DMAMR）*）での INTM フィールド（ビット [17:16]）の設定に応じて変化します。*表 511* で、転送完了割り込みの動作について説明しています。

表 511. 転送完了割り込みの動作

割り込みモード	TI/RI および割り込み信号の動作
INTM=0	TI/RI ステータス信号は、転送完了イベントが検出されると必ずセットされます。 これらのビットは、ソフトウェアドライバによって ‘1’ を書き込まれると必ずクリアされます。 割り込み信号は、対応する割り込みが ETH_DMACIER レジスタでも有効になっていると必ずアサートされます。
INTM=1	TI/RI は上記に説明されたようにセットされます。ただし、割り込みは RI/TI イベントに対してアサートされません。
INTM=2	RI/TI ステータスビットは、転送完了イベントが検出されると必ずセットされ、ソフトウェアドライバが ‘1’ を書き込むことによってクリアすると必ずリセットされます。ただし、ソフトウェアによってクリア（機能）される前に、別の転送完了イベントが検出された場合、これらのステータスビットは自動的に再びセットされます。ただし、割り込みは TI/RI に基づいて生成されません。

58.8.2 MTL 割り込み

MTL 割り込みイベントは、DMA でのイベントと組み合わせて、割り込み信号を生成します。

レジスタ *割り込みステータスレジスタ（ETH_MTLISR）*は、イベントを管理するキュー数をレポートします。ETH_MTLQICSR：キュー割り込み制御ステータスは、イベントの説明のために読み出されます。

MTL 割り込みはデフォルトで有効にされます。*割り込みステータスレジスタ（ETH_MTLISR）*レジスタの対応するマスクビットをセットすることによって、各イベントが割り込みをアサートしないようにすることができます。

MTL 割り込み信号は、次のいずれかのイベントによって駆動されます。

- 受信キューオーバーフロー割り込み
- 送信キューアンダーフロー

58.8.3 MAC 割り込み

MAC 割り込みイベントは、DMA でのイベントと組み合わせて、割り込み信号を生成します。

MAC 割り込みはレベルタイプで、アプリケーションやソフトウェアによってクリアされるまで、割り込みはアサート（ハイ）されたままになります。

*割り込みステータスレジスタ（ETH_MACISR）*は、MAC からの割り込みの原因となるイベントを記述します。MAC 割り込みはデフォルトで有効にされます。*割り込みステータスレジスタ（ETH_MACISR）*の対応するマスクビットをセットすることによって、各イベントが割り込みをアサートしないようにすることができます。

割り込みレジスタビットは、イベントが報告されるブロックを示すだけです。割り込みをクリアするには、対応するステータスレジスタと他のレジスタを読み出す必要があります。

MAC 割り込み信号は、次のいずれかのイベントによって駆動されます。

- 受信ステータス割り込み
- 送信ステータス割り込み
- タイムスタンプ割り込みステータス
- MMC 割り込みステータス
  - MMC 受信チェックサムオフロード割り込みステータス
  - MMC 送信割り込みステータス
  - MMC 受信割り込みステータス
- LPI 割り込みステータス
- PMT 割り込みステータス
- PHY 割り込み

注： ***2 つのサイドバンド信号が、LPI および PMT 割り込みとともに生成されます。lpi_intr_o および pmt_intr_o です。これらは EXTI レベルでのウェイクアップイベント検出に使用されます。***

参考資料

58.9 Ethernet のプログラミングモデル

この章では、適切なシーケンスでの DMA または MAC レジスタの初期化に関して説明します。以下のセクションが含まれています。

- DMA の初期化（*セクション 58.9.1* を参照）
- MTL の初期化（*セクション 58.9.2* を参照）
- MAC の初期化（*セクション 58.9.3* を参照）
- 通常受信および送信動作の実行（*セクション 58.9.4* を参照）
- 送信の停止および開始（*セクション 58.9.5* を参照）
- MII リンク状態遷移に関するプログラミングガイドライン（*セクション 58.9.6* を参照）
- IEEE 1588 タイムスタンプに関するプログラミングガイドライン（*セクション 58.9.7* を参照）
- 省電力型 Ethernet に関するプログラミングガイドライン（*セクション 58.9.8* を参照）
- 柔軟性の高い Pulse-Per-Second（PPS）出力に関するプログラミングガイドライン（*セクション 58.9.9* を参照）
- TSO に関するプログラミングガイドライン（*セクション 58.9.10* を参照）
- 受信時の VLAN フィルタに関するプログラミングガイドライン（*セクション 58.9.11* を参照）

58.9.1 DMA の初期化

DMA を初期化するには、次の手順を完了します。

1. すべての MAC 内部レジスタおよびロジック（*DMA モードレジスタ（ETH_DMAMR）* のビット 0）をリセットするためにソフトウェアリセットを提供します。
2. リセットプロセスの完了を待ちます（リセット動作が完了したときにクリアされる *DMA モードレジスタ（ETH_DMAMR）* のビット 0 にポーリングします）。
3. 次のフィールドをプログラムして、*システムバスモードレジスタ（ETH_DMASBMR）* を初期化します。
   a) AAL
   b) 固定バーストまたは未定義バースト
   c) AHB バスインタフェースの場合のバーストモード値
   d) 固定長の値が有効である場合、AXI バス（ビット [7:1]）で可能な最大バースト長の選択
4. 送信および受信のディスクリプタリストを作成します。さらに、受信ディスクリプタが DMA によって所有されていることを確認します（TDES3/RDES3 ディスクリプタのビット 31 をセットします）。ディスクリプタの詳細については、*セクション 58.10：ディスクリプタ*を参照してください。

注： ***リングの開始から終了までのディスクリプタアドレスは、4 GB の境界を越えないようにします。***

5. ETH_DMACTxRLR および ETH_DMACRxRLR レジスタをプログラムします（*チャネル Tx ディスクリプタリング長レジスタ（ETH_DMACTxRLR）* および *チャネル Rx ディスクリプタリング長レジスタ（ETH_DMACRxRLR）* を参照）。プログラムされるリングの長さは、4 以上にセットする必要があります。
6. 受信および送信のディスクリプタリストのアドレスを、送信および受信のディスクリプタのベースアドレスで初期化します（*チャネル Tx ディスクリプタリストアドレスレジスタ（ETH_DMACTxDLAR）*、*チャネル Rx ディスクリプタリストアドレスレジスタ（ETH_DMACRxDLAR）*）。さらに、DMA に使用できるディスクリプタを伝える送信および受信のテイルポインタレジスタをプログラムします（*チャネル Tx ディスクリプタテイルポインタレジスタ（ETH_DMACTxDTPR）* および *チャネル Rx ディスクリプタテイルポインタレジスタ（ETH_DMACRxDTPR）*）。
7. ETH_DMACCR、ETH_DMACTxCR および ETH_DMACRxCR レジスタ（*チャネル 制御レジスタ（ETH_DMACCR）* および *チャネル 送信制御レジスタ（ETH_DMACTxCR）* を参照）をプログラムして、DMA によって開始される最大バースト長（PBL）、ディスクリプタスキップ長、TxDMA の OSP、RxDMA の RBSZ などのパラメータを設定します。

8. ETH_DMACIER レジスタをプログラムすることで割り込みを有効にします（*チャネル割り込み有効レジスタ（ETH_DMACIER）*を参照）。
9. *チャネル 受信制御レジスタ（ETH_DMACRxCR）*の SR（ビット 0）および ETH_DMACTxCR の ST（ビット 0）をセットすることで受信および送信 DMA を開始します（*チャネル 送信制御レジスタ（ETH_DMACTxCR）*を参照）。

58.9.2 MTL の初期化

MTL レジスタを初期化するには、次の手順を完了します。

1. 次のフィールドをプログラムして、ETH_MTLTxQ0OMR の動作モードを初期化します（*Tx キュー動作モードレジスタ（ETH_MTLTxQOMR）*を参照）。
   a) 閾値モードが使用されている場合、送信ストアアンドフォワード（TSF）または送信閾値制御（TTC）
   b) 送信キュー有効（TXQEN）の値を 2'b10 にして送信キュー 0 を有効化
   c) 送信キューサイズ（TQS）
2. 次のフィールドをプログラムして、ETH_MTLRxQOMR レジスタの動作モードを初期化します（*Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）*を参照）。
   a) 閾値モードが使用されている場合、受信ストアアンドフォワード（RSF）または RTC
   b) MTL 受信 FIFO のフロー制御有効化および無効化閾値（RFA および RFD）
   c) 有効化に続くエラーパケットおよびアンダーサイズの良好パケット（FEP および FUP）
   d) 受信キューサイズ（TQS）

58.9.3 MAC の初期化

以下の MAC 初期化操作は、DMA 初期化後に実行できます。DMA が設定される前に MAC 初期化が完了した場合、DMA がアクティブになった後にのみ MAC レシーバを有効にします（以下のシーケンスの最後の手順）。そうしないと、受信フレームが Rx FIFO を満たしてオーバーフローします。

1. 次の MAC アドレスレジスタが提示されます。*アドレス x 下位レジスタ（ETH_MACAxLR）*および*アドレス 0 上位レジスタ（ETH_MACA0HR）*です。複数の MAC アドレスが設定で有効になっている場合（追加アドレス 3 つまで）、MAC アドレスを適切にプログラムします。
2. 次のフィールドをプログラムして、*パケットフィルタ制御レジスタ（ETH_MACPFR）*の着信フレームに適切なフィルタを設定します。
   a) すべてを受信
   b) プロミスキャスモード
   c) ハッシュまたは完全フィルタ
   d) ユニキャスト、マルチキャスト、ブロードキャストおよび制御フレームフィルタ設定
3. 次のフィールドをプログラムして、*Tx キューフロー制御レジスタ（ETH_MACQTxFCR）*で適切なフロー制御を行います。
   a) ポーズ時間およびその他のポーズフレーム制御ビット
   b) 送信フロー制御ビット
   c) フロー制御ビジー
4. 設定に適用できる場合、*割り込み有効レジスタ（ETH_MACIER）*を必須としてプログラムします。
5. *動作モード設定レジスタ（ETH_MACCR）*レジスタの適切なフィールドをプログラムします。
   例：送信およびジャバー無効化中のパケット間隔。
6. *動作モード設定レジスタ（ETH_MACCR）*レジスタのビット 0 および 1 をセットして、MAC トランスミッタおよびレシーバを開始します。

58.9.4 通常受信および送信動作の実行

通常動作では、次の手順を完了します。

1. 通常送信および受信割り込みでは、割り込みステータスを読み出します。次に、ホスト（送信または受信）によって所有されるディスクリプタのステータスを読み出し、ディスクリプタにポーリングします。
2. ディスクリプタに適切な値をセットします。データの送受信を再開するために、送信および受信ディスクリプタが DMA によって所有されていることを確認します。
3. ディスクリプタが DMA によって所有されていない場合（またはディスクリプタが使用できない場合）、DMA はサスペンド状態に入ります。ディスクリプタを開放して、ETH_DMACTxDTPR（*チャネル Tx ディスクリプタテイルポインタレジスタ（ETH_DMACTxDTPR）*および ETH_DMACRxDTPR（*チャネル Rx ディスクリプタテイルポインタレジスタ（ETH_DMACRxDTPR）*を参照）を参照）を書き込むことによって送受信を再開できます。
4. デバッグモードでは、現在のホストのトランスミッタおよびレシーバのディスクリプタアドレスポインタの値は、ETH_DMACCATxDR および ETH_DMACCARxDR レジスタで読み出せます（*チャネル現在のアプリケーション送信ディスクリプタレジスタ（ETH_DMACCATxDR）*および*チャネル現在のアプリケーション受信ディスクリプタレジスタ（ETH_DMACCARxDR）*）。
5. デバッグモードでは、現在のホストの送信バッファアドレスポインタおよび受信バッファアドレスポインタの値は、ETH_DMACCATxDR および ETH_DMACCARxDR レジスタで読み出せます（*チャネル現在のアプリケーション送信ディスクリプタレジスタ（ETH_DMACCATxDR）*および*チャネル現在のアプリケーション受信ディスクリプタレジスタ（ETH_DMACCARxDR）*）。

58.9.5 送信の停止および開始

次の手順を完了して、しばらく送信を一時停止します。

1. ETH_DMACTxCR レジスタのビット 0（ST）をクリアすることで送信 DMA（適用できる場合）を無効にします（*チャネル 送信制御レジスタ（ETH_DMACTxCR）*を参照）。
2. 前のフレーム送信の完了を待ちます。ETH_MTLTxQDR レジスタの適切なビットを読み出すことでこれをチェックできます（TRCSTS が 01 ではなく、TXQSTS=0）。
3. *動作モード設定レジスタ（ETH_MACCR）*レジスタのビット 0（RE）およびビット 1（TE）をクリアすることによって、MAC トランスミッタと MAC レシーバを無効にします。
4. Rx FIFO のデータがシステムメモリに転送されたことを（*Tx キューデバッグレジスタ（ETH_MTLTxQDR）*の適切なビット、PRXQ='0' および RXQSTS='00' を読み出すことによって）確認した後、受信 DMA（適用できる場合）を無効にします。
5. Tx キューおよび Rx キュー両方がエンプティであること（*Tx キューデバッグレジスタ（ETH_MTLTxQDR）*で TXQSTS が 0、RXQSTS が '0' にセットされていること）を確認します。
6. 動作を再開するには、まず DMA を開始してから、MAC トランスミッタおよびレシーバを有効にします。

58.9.6 MII リンク状態遷移に関するプログラミングガイドライン

リンクがダウンしているときに送信および受信クロックが動作している場合

送信および受信クロックが動作しているときにリンクがダウンした場合、次の手順を完了します。

1. *チャネル 制御レジスタ（ETH_DMACCR）*のビット 0（ST）をクリアすることで送信 DMA（適用できる場合）を無効にします。
2. *動作モード設定レジスタ(ETH_MACCR)*のビット 2(RE)をクリアすることで MAC レシーバを無効にします。
3. 前のフレーム送信の完了を待ちます。*Tx キューデバッグレジスタ（ETH_MTLTxQDR）*の適切なビットを読み出すことでこれをチェックできます（TRCSTS は 01 ではありません）。

   または

   より速いエンプティ操作を行うために Tx FIFO を一掃します。
4. *動作モード設定レジスタ（ETH_MACCR）*レジスタのビット 1（TE）をクリアすることで MAC トランスミッタを無効にします。
5. Tx および Rx キュー両方がエンプティであること（*Tx キューデバッグレジスタ（ETH_MTLTxQDR）*で TXQSTS が ‘0’、*Rx キューデバッグレジスタ(ETH_MTLRxQDR)*で RXQSTS が ‘0’ にセットされていること）を確認します。
6. リンクがアップになってから、PHY レジスタを読み出して最新の設定を特定し、それに応じて MAC レジスタをプログラムします。
7. Tx DMA を開始することで動作を再開します。続いて、MAC トランスミッタとレシーバを有効にします。

   Rx DMA は有効にする必要はありません。レシーバが無効になっているため、Rx FIFO にデータはありません。

リンクがダウンしているときに送信および受信クロックが停止している場合

リンクがダウンして、送信および受信クロックが停止している場合、次の手順を完了します。

1. *動作モード設定レジスタ（ETH_MACCR）*の RE および TE ビットをクリアすることによって、MAC トランスミッタとレシーバを無効にします。クロックがないため、これによる影響はただちにありません。
2. リンクがアップになり、クロックが復旧するまで待ちます。
3. 送信／受信クロックが停止したときに進行中だった操作がある場合、部分的なフレームの転送が完了するまで待ちます。これは、*デバッグレジスタ（ETH_MACDR）*を読み出すことでチェックできます（すべてのビットを “0” にセットする必要があります）。一部の古いパケットは MAC トランスミッタが停止した状態のまま TXFIFO に残る場合があります。
4. PHY レジスタを読み出して最新の動作モードを特定し、それに応じて MAC レジスタをプログラムします。
5. RE および TE ビットをセットすることによって、MAC トランスミッタとレシーバを再起動します。

58.9.7 IEEE 1588 タイムスタンプに関するプログラミングガイドライン

システム時間生成の初期化

タイムスタンプ機能は、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 0 をセットすることによって有効にできます。ただし、このビットをセットした後に、タイムスタンプカウンタを初期化する必要があります。ペリフェラルの初期化を実行するには、次の手順を完了します。

1. *割り込み有効レジスタ（ETH_MACIER）*のビット 16 をクリアすることでタイムスタンプトリガ割り込みをマスクします。
2. *タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 0 をセットしてタイムスタンプを有効にします。
3. PTP クロック周波数に基づいて、*サブセカンドインクリメントレジスタ（ETH_MACSSIR）*をプログラムします。
4. 密補正法を使用する場合、*タイムスタンプ加数レジスタ（ETH_MACTSAR）*をプログラムして*タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 5 をセットします。
5. ビット 5 がクリアされるまで、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*にポーリングします。
6. *タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 1 をプログラムして、密更新法（必要な場合）を選択します。
7. *システム時間秒更新レジスタ（ETH_MACSTSUR）*および*システム時間ナノ秒更新レジスタ（ETH_MACSTNUR）*を適切な時間値でプログラムします。
8. *タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 2 をセットします。

   タイムスタンプカウンタは、タイムスタンプ更新レジスタに書き込まれた値で初期化されると開始します。ワンステップタイムスタンプが必要な場合：

   a) TDES3 コンテキストディスクリプタのビット 27 をプログラムすることでワンステップタイムスタンプを有効にします。

   b) PDelay_Req PTP メッセージの補正フィールドを更新するために*タイムスタンプ入力非対称補正レジスタ（ETH_MACTSIACR）*をプログラムします。
9. 適切なタイムスタンピングのためには、MAC レシーバおよびトランスミッタを有効にします。

注： ***タイムスタンプ制御レジスタ（ETH_MACTSCR）のビット 0 をクリアすることによってタイムスタンプ動作が無効にされた場合、タイムスタンプ動作を再開するには、これらすべての手順を繰り返します。***

システム時間の補正

システム時間を一気に同期または更新するには（粗補正法）、次の手順を完了します。

1. タイムスタンプ更新レジスタ（*システム時間秒更新レジスタ（ETH_MACSTSUR）*および*システム時間ナノ秒更新レジスタ（ETH_MACSTNUR）*）でオフセットをセットします（正または負）。
2. *タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 3（TSUPDT）をセットします。

   TSUPDT ビットがクリアされると、タイムスタンプ更新レジスタの値がシステム時間に加えられるか、差し引かれます。

システム時間を同期または更新して、システム時間のジッタを減らすには（密補正法）、次の手順を完了します。

1. *セクション：システム時間レジスタモジュール*で説明されているアルゴリズムを使用して、システム時間のインクリメントまたはデクリメントの割合を計算します。
2. *タイムスタンプ加数レジスタ（ETH_MACTSAR）*を新しい値で更新して、*タイムスタンプ制御レジスタ（ETH_MACTSCR）*レジスタのビット 5 をセットします。
3. 加数レジスタの新しい値として有効にしたい時間を待ちます。この操作は、システム時間が目標値に達した後、タイムスタンプトリガ割り込みを有効にして実行できます。
4. *PPS 目標時間秒レジスタ（ETH_MACPPSTTSR）*および*PPS 目標時間ナノ秒レジスタ（ETH_MACPPSTTNR）*で、必要な目標時間をプログラムします。
5. *割り込み有効レジスタ（ETH_MACIER）*のビット 12 でタイムスタンプ割り込みを有効にします。
6. レジスタ *タイムスタンプ制御レジスタ（ETH_MACTSCR）*のビット 4 をセットします。
7. このトリガによって割り込みが生成されたときには、*割り込みステータスレジスタ（ETH_MACISR）*を読み出します。
8. *タイムスタンプ加数レジスタ(ETH_MACTSAR)*を古い値で再プログラムして、再びビット 5 をセットします。

58.9.8 省電力型 Ethernet に関するプログラミングガイドライン

Tx LPI モードの移行と終了

MAC 初期化中に次の手順を完了します。

1. MDIO インタフェースを通じて PHY レジスタを読み出し、リモートエンドに EEE 機能があるかどうかチェックします。次に、タイマの値をネゴシエイトします。
2. MDIO インタフェースを通じて PHY レジスタをプログラムします（LPI モードで Rx クロックを停止するかどうかを PHY に通知する RX_CLK_stoppable ビットを含む）。
3. *LPI タイマ制御レジスタ（ETH_MACLTCR）*でビット 25から16と、ビット 1 を 0 にプログラムします。
4. MDIO インタフェースを使用して PHY リンクステータスを読み出し、*LPI 制御ステータスレジスタ（ETH_MACLCSR）*のビット 17 を更新します。
5. それに応じて、*LPI 制御ステータスレジスタ（ETH_MACLCSR）*を更新します。この更新は、PHY チップのリンクステータスが変化するとき、必ず実行してください。
6. CSR スレーブポートにアクセスするために使用されるクロックの周波数に従って*1 マイクロ秒ティックカウンタレジスタ（ETH_MAC1USTCR）*をプログラムします。
7. MAC が LPI 状態に移行する前に待つ必要があるアイドル時間を *LPI 移行タイマレジスタ（ETH_MACLETR）*の LPIET ビットにプログラムします。
8. *LPI 制御ステータスレジスタ(ETH_MACLCSR)*の LPITE および LPITXA（ビット 20 から 19）をセットします。
9. *LPI 制御ステータスレジスタ（ETH_MACLCSR）*を更新して、LPI 状態から LPI 自動移行および MAC 自動終了を有効にします。
10. CSR スレーブポートにアクセスするために使用されるクロックの周波数に従って*1 マイクロ秒ティックカウンタレジスタ（ETH_MAC1USTCR）*をプログラムします。
11. MAC が LPI 状態に移行する前に待つ必要があるアイドル時間を *LPI 移行タイマレジスタ（ETH_MACLETR）*レジスタの LPIET ビットにプログラムします。
12. *LPI 制御ステータスレジスタ（ETH_MACLCSR）*の LPITE および LPITXA（ビット 20 および 19）をセットして、LPI 状態から LPI 自動移行および MAC 自動終了を有効にします。
13. *LPI 制御ステータスレジスタ（ETH_MACLCSR）*のビット 16 をセットして、MAC トランスミッタを LPI 状態にします。

    すべてのスケジュールされたパケットが完了すると、MAC は LPI 状態に移行します。LPIET ビットで示された時間はアイドル状態のままとなります。LPI 状態に移行した後、TLPIEN（ビット 0）をセットします。

14. パケット送信がスケジュールされているとき（TxDMA がアイドル状態を終了したとき、または ATI や MTI インタフェースにパケットがあるとき）、MAC トランスミッタは自動的に LPI 状態を終了します。TLPIEX 割り込みステータスビットがセットされるまで TWT の時間を待ってから、パケット送信を再開します。
15. LPIET の時間、MAC トランスミッタがアイドル状態のままだった場合、LPI 状態に再び移行します。そして、TLPIEN ビットがセットされ、移行と終了のサイクルが続きます。
16. アプリケーションが自動移行／終了モードを上書きして、MAC トランスミッタの LPI 状態を直接終了させる必要がある場合、LPITXEN ビットをリセットします。

58.9.9 柔軟性の高い Pulse-Per-Second（PPS）出力に関するプログラミングガイドライン

PPS でのシングルパルスの生成

PPS でのシングルパルスの生成方法：

1. *PPS 制御レジスタ（ETH_MACPPSCR）*で TRGTMODSEL ビットを ‘11’ または ‘10’（割り込み）にプログラムします。これで、MAC に目標時間レジスタ（レジスタ 736 および 737）を PPS 信号出力の開始時間として使用するよう命令します。
2. 目標時間レジスタ（レジスタ 736 および 737）で開始時間値をプログラムします。
3. *PPS 幅レジスタ（ETH_MACPPSWR）*レジスタで PPS 信号出力の幅をプログラムします。
4. *PPS 制御レジスタ（ETH_MACPPSCR）*の PPSCMD を ‘0001’ にプログラムします。これで、MAC に目標時間レジスタでプログラムされた時間に PPS 信号出力でシングルパルスを生成するよう命令します。

PPSCMD が実行されると（PPSCMD ビット = 0）、プログラムされた開始時間が経過する前に、開始キャンセルコマンド（PPSCMD=’0011’）を示すことでパルス生成をキャンセルできます。予め次のパルスの動作をプログラムすることもできます。次のパルスをプログラムする方法：

1. 目標時間レジスタで次のパルスの開始時間をプログラムします。この時間は、立ち下がりエッジが前のパルスに発生する時間より大きくする必要があります。
2. *PPS 幅レジスタ（ETH_MACPPSWR）*で次の PPS 信号出力の幅をプログラムします。
3. *PPS 制御レジスタ（ETH_MACPPSCR）*の PPSCMD ビットをプログラムして、前のパルスがネゲートされた後にシングルパルスを生成します。これで、MAC に目標時間レジスタでプログラムされた時間に PPS 信号出力でシングルパルスを生成するよう命令します。

   前のパルスがローになる前に、このコマンドが示された場合、新しいコマンドは前のコマンドを上書きして、QOS によって 1 つだけ拡張されたパルスが生成されます。

PPS でのパルス列の生成

PPS でのパルス列の生成方法：

1. *PPS 制御レジスタ（ETH_MACPPSCR）*で TRGTMODSEL ビットを ‘11’ または ‘10’（割り込み）にプログラムします。これで、MAC に目標時間レジスタ（レジスタ 736 および 737）を PPS 信号出力の開始時間として使用するよう命令します。
2. 目標時間レジスタ（レジスタ 736 および 737）で開始時間値をプログラムします。
3. *PPS 間隔レジスタ（ETH_MACPPSIR）*で PPS 信号出力でのパルス列の間隔をプログラムします。
4. *PPS 幅レジスタ（ETH_MACPPSWR）*で PPS 信号出力の幅をプログラムします。
5. *PPS 制御レジスタ（ETH_MACPPSCR）*の PPSCMD ビットを ‘0010’ にプログラムします。これで、MAC に目標時間レジスタでプログラムされた開始時間に PPS 信号出力でパルス列を生成するよう命令します。

   デフォルトでは、PPS パルス列は「パルス列の同時停止」または「パルス列の即時停止」コマンドを発行して停止されない限り、フリーランニングです。
6. 目標時間レジスタで停止値をプログラムします。目標時間レジスタを再びプログラムする前に、*PPS 目標時間ナノ秒レジスタ（ETH_MACPPSTTNR）*の TSTRBUSY ビットがリセットされていることを確認します。
7. *PPS 制御レジスタ（ETH_MACPPSCR）*の PPSCMD ビットを ‘0100’ にプログラムして、手順 6 で指定されプログラムされた停止時間が経過した後に PPS 信号出力のパルス列を停止します。

パルス列は、PPSCMD フィールドで ‘0101’ をプログラムすることで、いつでも停止できます。

同様に、（手順 5 に示された）パルス列停止コマンドは、（手順 6 でプログラムされた）時間が経過する前に PPSCMD ビットを ‘0110’ にプログラムすることでキャンセルできます。

パルス列の生成は、（手順 2 でプログラムされた）開始時間が経過する前に PPSCMD を ‘0011’ にプログラムすることで停止できます。

PPS に影響しない割り込みの生成

*PPS 制御レジスタ（ETH_MACPPSCR）*の TRGTMODSEL ビットで、目標時間レジスタ（レジスタ 736 および 737）をプログラムして、次のいずれかの操作が可能です。

- 割り込みのみの生成
- 割り込みおよび PPS の開始時間と停止時間の生成
- PPS の開始時間と停止時間のみの生成

目標時間レジスタをプログラムして割り込みイベントのみを生成する方法：

1. *PPS 制御レジスタ（ETH_MACPPSCR）*の TRGTMODSEL ビットを ‘00’（割り込み）にプログラムします。これで、MAC に目標時間レジスタを目標時間の割り込みに使用するよう命令します。
2. 目標時間レジスタで目標時間値をプログラムします。これで、MAC に目標時間が経過したときに割り込みを生成するよう命令します。

   （たとえば PPS を制御するために）TRGTMODSEL ビットが変更された場合、割り込み生成は新しいモードと新しくプログラムされた目標時間レジスタ値で上書きされます。

58.9.10 TSO に関するプログラミングガイドライン

TSO をプログラムするには、次の手順に従います。

1. 対応する ETH_DMACTxCR レジスタの TSE ビットをプログラムして、その DMA で TCP パケットのセグメンテーションを有効にします。
2. 通常の転送ディスクリプタ設定に加えて、次のディスクリプタフィールドをプログラムして現在のパケットに TSO を有効にしなければなりません。
   a) TDES3 の TSE（ビット 18）を有効化します。
   b) TDES3 のビット 17〜0 で分割されていない TCP/IP パケットペイロードの長さと、TDES3 のビット 22〜19 で TCP ヘッダをプログラムします。
   c) 以下でセグメントの最大サイズをプログラムします。
   – ETH_DMACCR の MSS
   – またはコンテキストディスクリプタの MSS

   MSS フィールドが ETH_DMACCR およびコンテキストディスクリプタでプログラムされている場合、最新のソフトウェアによってプログラムされたシーケンスが考慮されます。
3. 分割されていない TCP/IP パケットヘッダは、先頭ディスクリプタのバッファ 1 に格納する必要があります。このバッファはペイロードのバイトを保持してはなりません。ペイロードはバッファ 2 と後続のディスクリプタのバッファに割り当てられます。

注意：　非 TCP-IP パケットに対して TDES3 で TSE が有効になっている場合、結果は予想できません。

58.9.11 受信時の VLAN フィルタに関するプログラミングガイドライン

受信時に VLAN フィルタを実行するには、以下のシーケンスに従います。

1. 次のビットに *VLAN タグレジスタ（ETH_MACVTR）* をプログラムして、フィルタ方法を選択します。
   – ETV：12 ビット VLAN タグ比較または 16 ビット VLAN タグ比較を有効にします。
   – VTHM：VLAN タグハッシュテーブル一致を有効にします。
   – ERIVLT：内部 VLAN タグまたは外部 VLAN タグを有効にします（内部または外部 VLAN タグフィルタを有効にするには、EDVLP をセットしてダブル VLAN 処理を有効にする必要があります）。
   – ERSVLM：受信 S-VLAN 一致または C-VLAN 一致を有効にします（S-VLAN 処理を有効にするには、ESVL をセットします）。
   – DOVLTC：タグ一致の VLAN タイプを無視します。
   – VTIM：通常の VLAN タグ一致ではなく、VLAN タグ反転一致を有効にします。
2. 12 ビットまたは 16 ビット VLAN タグについては *VLAN タグレジスタ（ETH_MACVTR）* で VL ビットをプログラムします。
3. VLAN タグハッシュフィルタが有効になっている場合、*VLAN ハッシュテーブルレジスタ（ETH_MACVHTR）* をプログラムします。計算された CRC の上位 4 ビットを使用して、VLAN ハッシュテーブルの内容をインデックスします。たとえば、'1000' のハッシュ値は、VLAN ハッシュテーブルのビット 8 を選択します。

58.10 ディスクリプタ

58.10.1 ディスクリプタの概要

Ethernet ペリフェラルでは、DMA はリンクされたディスクリプタのリストに基づいてデータを転送します。アプリケーションは、システムメモリ（SRAM）内にディスクリプタを作成します。次の 2 種類のディスクリプタをサポートしています。

- **通常ディスクリプタ**

  通常ディスクリプタは、パケットデータに使用され、送信するパケットに適用できる制御情報を提供します。
- **コンテキストディスクリプタ**

  コンテキストディスクリプタは、送信するパケットに適用できる制御情報を提供するために使用されます。

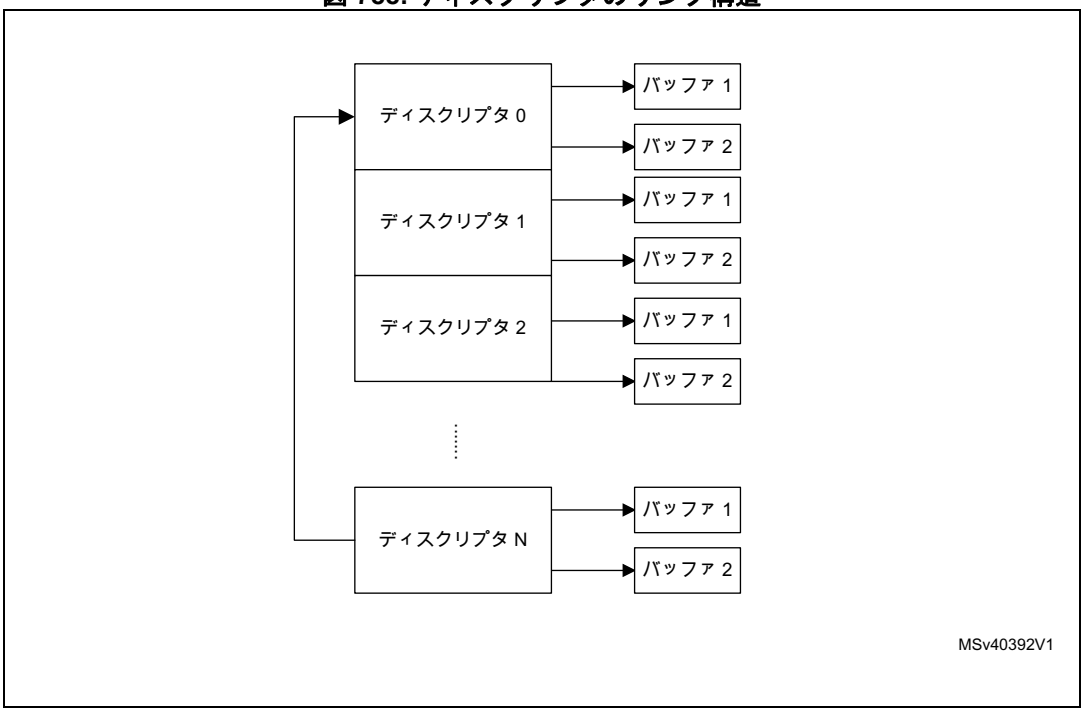
各通常ディスクリプタには、2 つのバッファと 2 つのアドレスポインタが含まれます。これらのバッファは、アダプタポートにさまざまなタイプのメモリ管理構成との互換性を持たせます。

1 パケットに使用できるディスクリプタの数に制限はありません。

58.10.2 ディスクリプタの構造

Ethernet ペリフェラルは、DMA ディスクリプタのリング構造をサポートします。

図 788. ディスクリプタのリング構造



リング構造では、ディスクリプタが*チャネル 制御レジスタ（ETH_DMACCR）*の DSL フィールドでプログラムされた 32 ビットワード数で分けられます。アプリケーションは、合計のリングの長さをプログラムする必要があり、これは DMA チャネルの次のレジスタでのリング範囲内のディスクリプタの合計数になります。

- *チャネル Tx ディスクリプタリング長レジスタ（ETH_DMACTxRLR）*
- *チャネル Rx ディスクリプタリング長レジスタ（ETH_DMACRxRLR）*

*チャネル Tx ディスクリプタテイルポインタレジスタ（ETH_DMACTxDTPR）*や*チャネル Rx ディスクリプタテイルポインタレジスタ（ETH_DMACRxDTPR）*には、ディスクリプタアドレスのポインタ（*N*）が含まれます。ベースアドレスと現在のディスクリプタポインタによって、DMA が処理できる現在のディスクリプタのアドレスが決定されます。ディスクリプタのテイルポインタによって示されたものより最大 1 か所少ないディスクリプタ（*N* – 1）が DMA によって所有されます。DMA は、次のいずれかの条件が発生するまでディスクリプタの処理を続けます。

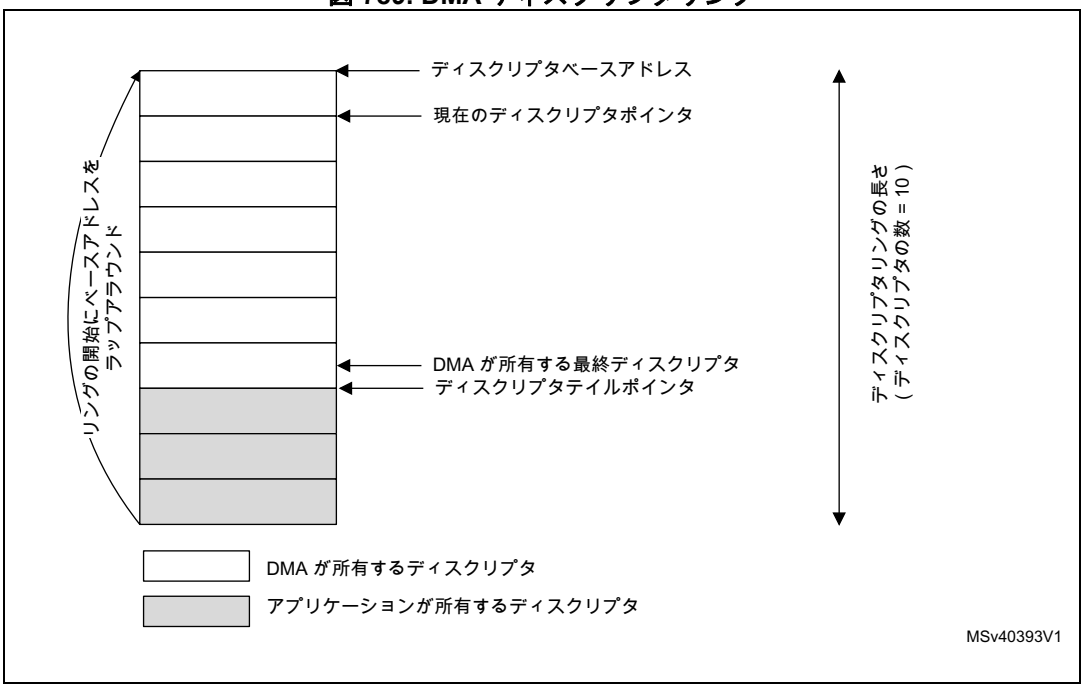
```
Current Descriptor Pointer == Descriptor Tail Pointer;
```

この条件が発生すると、DMA はサスペンド状態に移行します。アプリケーションは、ディスクリプタのテイルポインタレジスタに書き込み動作を実行し、次の条件を満たすようにテイルポインタを更新する必要があります。

```
Current Descriptor Pointer < Descriptor Tail Pointer;
```

*図 789 : DMA ディスクリプタリング*に示すように、リングの終わりに達すると、DMA は自動的にベースアドレスを巻き戻します。

図 789. DMA ディスクリプタリング



アプリケーションによって所有されるディスクリプタの場合、DES3 の OWN ビットが ‘0’ にリセットされます。

DMA によって所有されるディスクリプタの場合、OWN ビットが ‘1’ にセットされます。

最初に、アプリケーションがディスクリプタを 1 つだけ所有している場合、最終ディスクリプタアドレス（テイルポインタ）はディスクリプタベースアドレス + 1 にセットされます。次に、DMA は先頭ディスクリプタを処理して、アプリケーションがテイルポインタをインクリメントするのを待ちます。

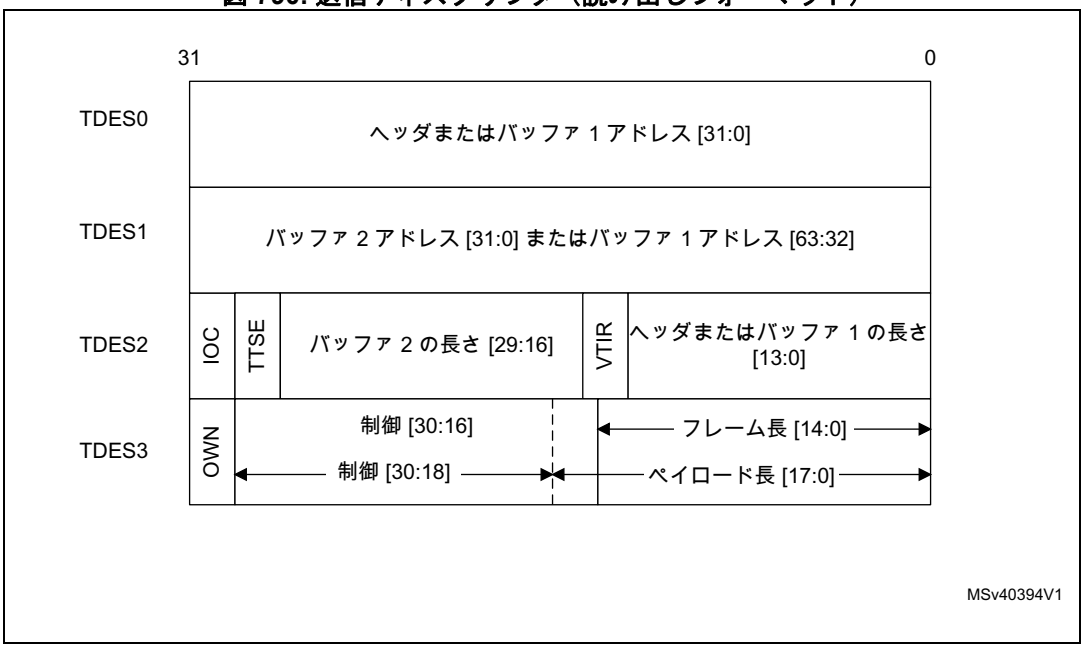
58.10.3 送信ディスクリプタ

Ethernet ペリフェラルの DMA は、送信パケットに最低 1 つのディスクリプタを必要とします。2 つのバッファ、2 つのバイトカウントバッファ、2 つのアドレスポインタに加えて、送信ディスクリプタ機能は、送信パケット単位で MAC 動作を管理するために使用できるフィールドを制御します。送信通常ディスクリプタには、次の 2 つのフォーマットがあります。読み出しフォーマットと書き戻しフォーマットです。

送信通常ディスクリプタ（読み出しフォーマット）

図 790 に、送信通常ディスクリプタの読み出しフォーマットを示します。*表 512*～*表 515* では、すべての送信通常ディスクリプタ（読み出しフォーマット）について詳細に説明しています。

図 790. 送信ディスクリプタ（読み出しフォーマット）



- **TDES0 通常ディスクリプタ（読み出しフォーマット）**

表 512. TDES0 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31:0	BUF1AP	**バッファ 1 アドレスポインタまたは TSO ヘッダアドレスポインタ** 次のビットがセットされているとき、これらのビットはバッファ 1 または TSO ヘッダのアドレスポインタを示します。 – TDES3 の TSE ビット – TDES3 の FD ビット

- **TDES1 通常ディスクリプタ（読み出しフォーマット）**

表 513. TDES1 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31:0	BUF2AP	**バッファ 2 またはバッファ 1 アドレスポインタ：** ディスクリプタのリング構造が使用されているとき、これらのビットはバッファ 2 の物理アドレスを示します。バッファアドレスの整列に制約はありません。

- **TDES2 通常ディスクリプタ（読み出しフォーマット）**

31	30	29:16	15:14	13:0
IOC	TTSE	B2L	VTIR	HL または B1L

表 514. TDES2 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31	IOC	**完了時割り込み：** このビットは、現在のパケット送信が完了したときに*チャネルステータスレジスタ（ETH_DMACSR）*で TI ビットをセットします。
30	TTSE	**送信タイムスタンプ有効化**
29:16	B2L	**バッファ 2 の長さ** ドライバがこのフィールドをセットします。セット時、このフィールドはバッファ 2 の長さを示します。
15:14	VTIR	**VLAN タグの挿入または書き換え：** これらのビットは、パケットを送信する前に VLAN タグ付けまたはタグ削除を実行するよう MAC に要求します。VLAN タグの挿入、書き換え、または削除がパケットに有効である場合、アプリケーションは CRC パッド制御ビットを適切にセットする必要があります。これらのビットの値は次のとおりです。 – 00：VLAN タグを追加しません。 – 01：送信前にパケットから VLAN タグを削除します。このオプションは、VLAN パケットでのみ使用する必要があります。 – 10：*VLAN 内包レジスタ（ETH_MACVIR）*でプログラムされたタグ値またはコンテキストディスクリプタで VLAN タグを挿入します。 – 11：*VLAN 内包レジスタ（ETH_MACVIR）*でプログラムされたタグ値またはコンテキストディスクリプタでパケットの VLAN タグを書き換えます。このオプションは、VLAN パケットでのみ使用する必要があります。
13:0	HL または B1L	**ヘッダの長さまたはバッファ 1 の長さ** ヘッダの長さの場合、ビット [9:0] のみ考慮されます。ビット 13～0 はバッファ 1 の長さにのみ適用できます。 TCP セグメンテーションオフロード機能が TDES3 の TSE ビットを通じて有効になっている場合、このフィールドはヘッダの長さと等しくなります。TSE ビットが TDES3 でセットされている場合、ヘッダの長さには Ethernet ソースアドレスから TCP ヘッダの終了までの 長さ（バイト単位）が含まれます。TSO 機能でサポートされる最大ヘッダ長は 1023 バイトです。TSO 機能でサポートされる最大ヘッダ長は 1023 バイトです。 TCP セグメンテーションオフロード機能が有効ではない場合、このフィールドはバッファ 1 の長さと等しくなります。

- **TDES3 通常ディスクリプタ（読み出しフォーマット）**

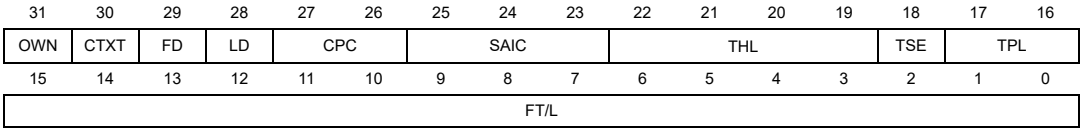


表 515. TDES3 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31	OWN	**オウンビット** – 1：DMA がディスクリプタを所有しています。 – 0：アプリケーションがディスクリプタを所有しています。 DMA は、関連付けられたバッファで示されたデータの転送を完了した後、このビットをクリアします。
30	CTXT	**コンテキストタイプ** このビットは通常ディスクリプタでは ‘0’ にセットする必要があります。
29	FD	**先頭ディスクリプタ** このビットがセットされた場合、バッファがパケットの最初のセグメントを含んでいることを示します。
28	LD	**最終ディスクリプタ** このビットがセットされた場合、バッファがパケットの最後のセグメントを含んでいることを示します。B1L または B2L フィールドは 0 以外の値でなければなりません。
27:26	CPC	**CRC パッド制御** このフィールドは、Tx パケットの CRC とパッド挿入を制御します。先頭ディスクリプタビット（TDES3[29]）がセットされた場合にのみ有効です。ビット [27:26] の値は次のとおりです。 – 00：CRC およびパッド挿入 MAC は、60 バイト以上の長さの送信パケットの最後に巡回冗長検査（CRC）を付加します。MAC は、60 バイト未満の長さのパケットには自動的にパディングと CRC を付加します。 – 01：CRC 挿入（パッド挿入無効化） MAC は、送信パケットの最後に CRC を付加しますが、パディングを付加しません。アプリケーションは、パディングバイトが送信バッファから送信されているパケットに存在している、つまり送信バッファから送信されているパケットが 60 バイト以上の長さであることを確認する必要があります。 – 10：CRC 挿入の無効化 MAC は、送信パケットの最後に CRC を付加しません。アプリケーションは、パディングおよび CRC バイトが送信バッファから送信されているパケットに存在していることを確認する必要があります。 – 11：CRC 書き換え MAC は、送信パケットの最後の 4 バイトを、再計算された CRC バイトで書き換えます。アプリケーションは、パディングおよび CRC バイトが送信バッファから送信されているパケットに存在していることを確認する必要があります。 ***注：　TSE ビットがセットされると、CRC およびパッド挿入がセグメンテーションに対して必ず行われるため、MAC はこのフィールドを無視します。***

表 515. TDES3 通常ディスクリプタ（読み出しフォーマット）（続き）

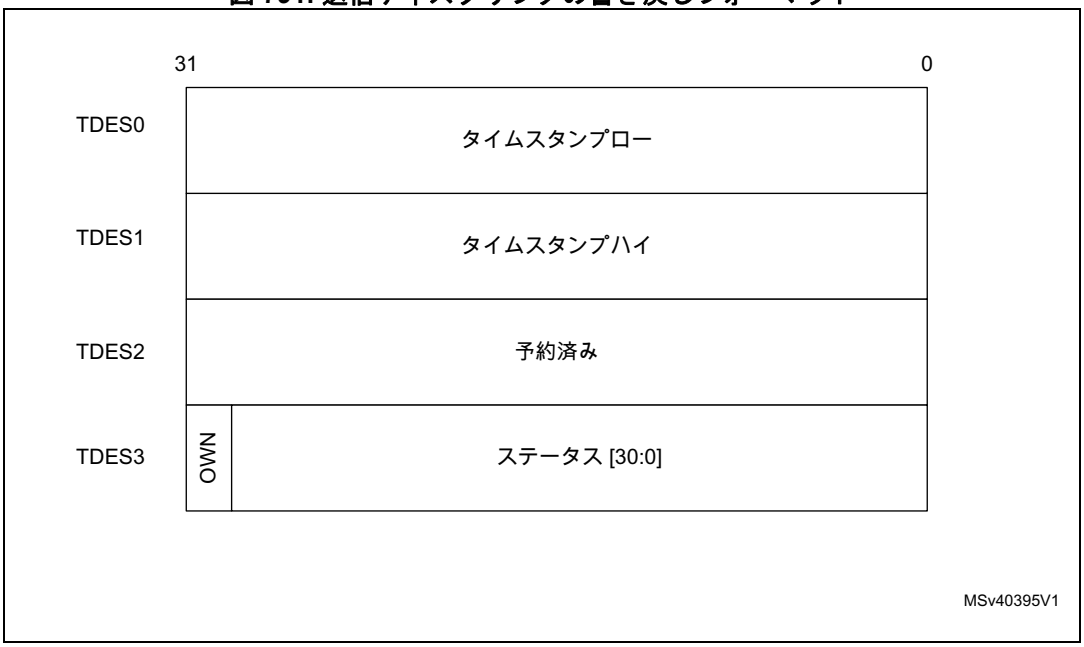
ビット	名前	説明
25:23	SAIC	**SA 挿入制御** これらのビットは、Ethernet パケットのソースアドレスフィールドに対して、MAC アドレス 0 レジスタに示された値の追加または書き換えを行うよう MAC に要求します。SA 挿入制御がパケットに対して有効である場合、アプリケーションは適切に CRC パッド制御ビットをセットする必要があります。 ビット 25 は、ソースアドレスの挿入または書き換えに使用される MAC アドレスレジスタ（1 または 0）の値を指定します。 次のリストではビット [24:23] の値を示しています。 – 00：ソースアドレスを含みません。 – 01：ソースアドレスを含む、または挿入します。信頼できる送信では、アプリケーションはソースアドレスなしでフレームを提供する必要があります。 – 10：ソースアドレスを書き換えます。信頼できる送信では、アプリケーションはソースアドレスありでフレームを提供する必要があります。 – 11：予約済み これらのフィールドは、最初のセグメント制御ビット（TDES3 [29]）がセットされているとき有効です。
22:19	THL	**THL**：TCP ヘッダの長さ TSE ビットがセットされている場合、このフィールドには TCP ヘッダの長さが入ります。このフィールドの最小値は 5 でなければなりません。
18	TSE	**TCP セグメンテーション有効化** このビットがセットされた場合、DMA がパケットに TCP セグメンテーションを実行します。このビットは、FD ビットがセットされたときにのみ有効です。
17:16	CIC/TPL	**チェックサム挿入制御または TCP ペイロード長** これらのビットは、チェックサムの計算と挿入を制御します。以下の値を取得できます。 – 00：チェックサム挿入が無効化されます。 – 01：IP ヘッダチェックサムの計算と挿入のみ有効化されます。 – 10：IP ヘッダチェックサムとペイロードチェックサムの計算と挿入が有効ですが、擬似ヘッダチェックサムはハードウェアでは計算されません。 – 11：IP ヘッダチェックサムとペイロードチェックサムの計算と挿入が有効であり、擬似ヘッダチェックサムはハードウェアで計算されます。 TSE ビットがリセットされているとき、このフィールドは有効です。TSE ビットがセットされているとき、TCP ペイロード長の上位ビット [17:16] を含みます。これにより、TCP パケット長フィールドが TDES3[17:0] にわたって拡大され、256 KB のパケット長をサポートします。
15	TPL	**予約済みまたは TCP ペイロード長** TSE ビットがリセットされると、このビットは予約済みとなります。TSE ビットがセットされると、TCP ペイロード長 [17:0] のビット 15 となります。
14:0	FL/TPL	**パケット長または TCP ペイロード長** このフィールドは、送信されるパケットの長さ（バイト単位）と等しくなります。TSE ビットがセットされていない場合、このフィールドは送信されるパケットの合計の長さと等しくなります。 Ethernet ヘッダ長 + TCP /IP ヘッダ長 – プリアンブル長 – SFD 長 + Ethernet ペイロード長 TSE ビットがセットされている場合、このフィールドは TCP ペイロード長の下位 15 ビットと等しくなります。この長さに、Ethernet ヘッダや TCP/IP の長さは含まれません。

送信通常ディスクリプタ（書き戻しフォーマット）

書き戻しフォーマットは、対応するパケットの最終ディスクリプタにのみ適用できます。LD ビット（TDES3[28]）は、DMA が対応する送信パケットのステータスおよびタイムスタンプの情報を書き戻すディスクリプタにセットされます。

図 791 に、送信通常ディスクリプタの書き戻しフォーマットを示します。*表 516*～*表 519* では、すべての送信通常ディスクリプタ（書き戻しフォーマット）について詳細に説明しています。

図 791. 送信ディスクリプタの書き戻しフォーマット



- **TDES0 通常ディスクリプタ（書き戻しフォーマット）**

表 516. DES0 通常ディスクリプタ（書き戻しフォーマット）(1)

ビット	名前	説明
31:0	TTSL	**送信パケットタイムスタンプロー** DMA は、このフィールドを、対応する送信パケットについてキャプチャされたタイムスタンプの下位 32 ビットで更新します。TDES2 の TTSE ビットがパケットの先頭ディスクリプタにセットされている場合のみ、DMA はタイムスタンプを書き込みます。このフィールドは、ディスクリプタの最後のセグメントビット（LS）がセットされ、タイムスタンプステータス（TTSS）ビットがセットされている場合にのみ、タイムスタンプを保持します。

1. このフォーマットは、パケットの最終ディスクリプタにのみ適用できます。

- **TDES1 通常ディスクリプタ（書き戻しフォーマット）**

表 517. TDES1 通常ディスクリプタ（書き戻しフォーマット）(1)

ビット	名前	説明
31:0	TTSH	**送信パケットタイムスタンプハイ** DMA は、このフィールドを、対応する受信パケットについてキャプチャされたタイムスタンプの上位 32 ビットで更新します。TDES2 の TTSE ビットがパケットの先頭ディスクリプタにセットされている場合のみ、DMA はタイムスタンプを書き込みます。このフィールドは、ディスクリプタの最後のセグメントビット（LS）がセットされ、タイムスタンプステータス（TTSS）ビットがセットされている場合にのみ、タイムスタンプを保持します。

1. このフォーマットは、パケットの最終ディスクリプタにのみ適用できます。

- **TDES2 通常ディスクリプタ（書き戻しフォーマット）**

表 518. TDES2 通常ディスクリプタ（書き戻しフォーマット）(1)

ビット	説明
31:0	予約済み

1. このフォーマットは、パケットの最終ディスクリプタにのみ適用できます。

- **TDES3 通常ディスクリプタ（書き戻しフォーマット）**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OWN	CTXT	FD	LD	予約済み										TTSS	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ES	JT	FF	PCE	LoC	NC	LC	EC	CC				ED	UF	DB	IHE

表 519. TDES3 通常ディスクリプタ（書き戻しフォーマット）(1)

ビット	名前	説明
31	OWN	**オウンビット** このビットがセットされた場合、DMA がディスクリプタを所有していることを示します。DMA はパケット送信を完了すると、このビットをクリアします。書き戻しが完了すると、このビットは ‘0’ にセットされます。
30	CTXT	**コンテキストタイプ** このビットは通常ディスクリプタでは ‘0’ にセットする必要があります。
29	FD	**先頭ディスクリプタ** このビット、バッファがパケットの最初のセグメントを含んでいることを示します。
28	LD	**最終ディスクリプタ** このビットは、パケットの最終ディスクリプタに対して 1 にセットされます。DMA はパケットの最終ディスクリプタでのみ、ステータスフィールドを書き込みます。
27:18	予約済み	

表 519. TDES3 通常ディスクリプタ（書き戻しフォーマット）(1)（続き）

ビット	名前	説明
17	TTSS	**Tx タイムスタンプステータス** このステータスビットは、対応する送信パケットに対してタイムスタンプがキャプチャされたことを示します。このビットがセットされると、TDES2 と TDES3 は、送信パケットに対してキャプチャされたタイムスタンプの値を持ちます。このフィールドは、ディスクリプタの最後のセグメント制御ビット（TDES3 [28]）がセットされたときのみ有効です。
16	予約済み	
15	ES	**エラー概要：** このビットは、以下のビットの論理 OR を示します。 – TDES3[0]：IP ヘッダエラー – TDES3[14]：ジャバータイムアウト – TDES3[13]：パケット一掃 – TDES3[12]：ペイロードチェックサムエラー – TDES3[11]：キャリア喪失 – TDES3[10]：キャリアなし – TDES3[9]：遅れ衝突 – TDES3[8]：過剰衝突 – TDES3[3]：過剰遅延 – TDES3[2]：アンダーフローエラー
14	JT	**ジャバータイムアウト** このビットは、MAC トランスミッタがジャバータイムアウトを検出しことを示します。*動作モード設定レジスタ（ETH_MACCR）*の JD ビットがセットされていないときにのみ、このビットがセットされます。
13	FF	**パケット一掃** このビットは、DMA/MTL が、CPU によって与えられたソフトウェア一掃コマンドにより、パケットを一掃したことを示します。
12	PCE	**ペイロードチェックサムエラー** このビットは、チェックサムオフロードエンジンに不具合があり、カプセル化された TCP、UDP、または ICMP ペイロードにチェックサムが挿入されなかったことを示します。この不具合は、IP ヘッダのペイロード長フィールドによって示されるバイト不足、または MTL がまだチェックサムが計算されていない状態でストアアンドフォワードモードでパケットを MAC トランスミッタに転送し始めることによって発生する可能性があります。この 2 つ目のエラー条件は、送信 FIFO の深さがデッドロックを回避するために送信される Ethernet パケットの長さより小さい場合にのみ発生します。ストアアンドフォワードモードでも FIFO がフルである場合、MTL はパケットの転送を開始します。
11	LoC	**キャリア喪失** このビットは、パケット送信中にキャリアの喪失が発生した（すなわち、パケット送信中に 1 つ以上の送信クロック周期の間、ETH_CRS 信号が無効であった）ことを示します。このビットは、MAC が半二重モードで動作しているとき、衝突なしで送信されたパケットについてのみ有効です。
10	NC	**キャリアなし** このビットは、送信中に PHY からのキャリアセンス信号がアサートされなかったことを示します。

表 519. TDES3 通常ディスクリプタ（書き戻しフォーマット）(1)（続き）

ビット	名前	説明
9	LC	**遅れ衝突** このビットは、衝突ウィンドウ（プリアンブルを含めて 64 バイト時間）の後で衝突が発生したためにパケット送信が中止されたことを示します。このビットは、アンダーフローエラーがセットされた場合は無効です。
8	EC	**過剰衝突** このビットは、現在のパケットの送信を試みている間に 16 の連続した衝突が発生した後、送信が中止されたことを示します。*動作モード設定レジスタ（ETH_MACCR）* で DR ビットがセットされている場合、このビットが最初の衝突後にセットされて、パケットの送信が中止されます。
7:4	CC	**衝突カウント** この 4 ビットのカウンタの値は、パケットが送信される前に発生した衝突の回数を示します。このカウントは、EC ビットがセットされているときには無効です。
3	ED	**過剰遅延** このビットは、*動作モード設定レジスタ（ETH_MACCR）* の DC ビットがセットされている場合、24,288 ビット時間を超える過剰遅延があったために送信が終了したことを示します。
2	UF	**アンダーフローエラー** このビットは、システムメモリからのデータが遅れて到着したために、MAC がパケット送信を中止したことを示します。アンダーフローエラーは、以下の条件のいずれかによって発生する可能性があります。 – DMA がパケット送信中に空の送信バッファを検出した場合 – アプリケーションが MAC の送信速度より遅く MTL Tx FIFO を満たす場合 送信プロセスはサスペンド状態に移行し、ETH_MTLISR レジスタでキューに対応するアンダーフロービットをセットします。
1	DB	**遅延ビット** このビットは、キャリアの存在のために MAC がパケット送信を延期したことを示します。このビットは、半二重モードでのみ有効です。
0	IHE	**IP ヘッダエラー** IP ヘッダエラーがセットされている場合、このビットはチェックサムオフロードエンジンが IP ヘッダエラーを検出したことを示します。このビットは、Tx チェックサムオフロードが有効化されている場合にのみ有効です。そうでない場合は、予約済みとなります。COE が IP ヘッダエラーを検出したときに、Ethernet タイプフィールドが IPv4 ペイロードを示している場合は、IPv4 ヘッダチェックサムを挿入します。

1. このフォーマットは、パケットの最終ディスクリプタにのみ適用できます。

送信コンテキストディスクリプタ

送信コンテキストディスクリプタは、パケットディスクリプタの前にいつでも提供できます。このコンテキストは、現在のパケットと後続のパケットに有効です。コンテキストディスクリプタは、ワンステップタイムスタンプ補正ではタイムスタンプ、VLAN 挿入機能では VLAN タグ ID、SA 挿入では SA 挿入ビットを提供するために使用されます。書き戻しは、OWN ビットをリセットするためにコンテキストディスクリプタでのみ実行されます。

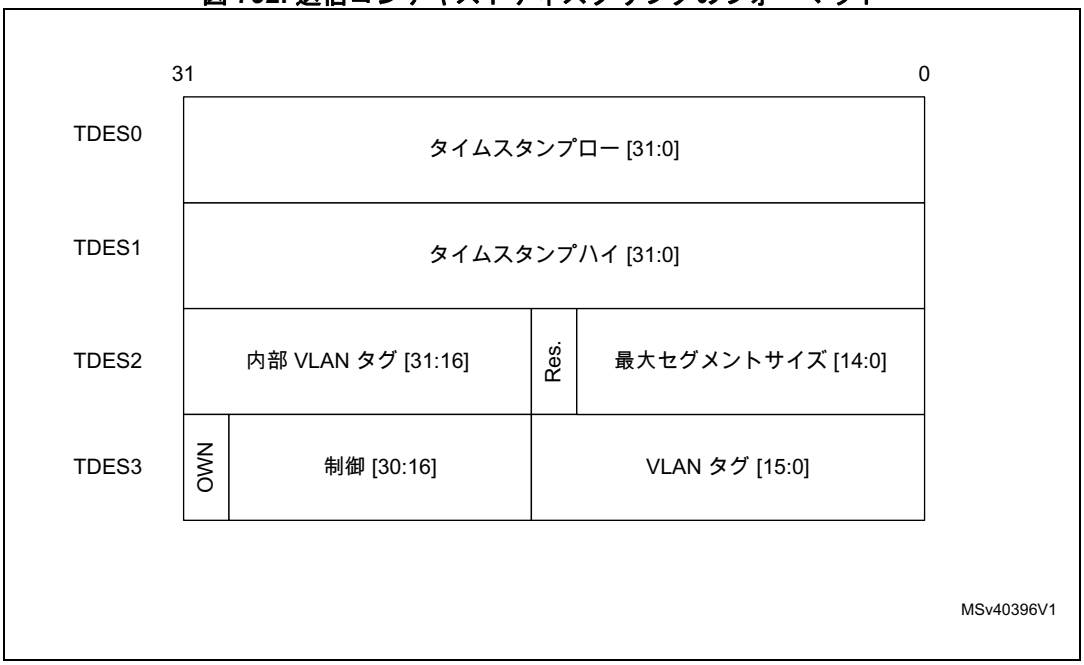
注： *VLAN タグ ID および MSS 値は、対応する有効ビットがセットされたコンテキストディスクリプタでアプリケーションによって提供され、DMA によって内部的に格納されます。*

外部または内部 VLAN タグが有効ビットがセットされた状態で提供される場合、DMA は常に最後の有効な VLAN タグを MTL に渡します。アプリケーションは、DMA によって格納された有効な VLAN タグを無効にできません。VLAN タグは、パケットに提供される制御入力に基づいて挿入または書き換えされます。

内部 VLAN タグ制御入力は、コンテキストディスクリプタの直後に続くパケットにのみ使用されます。アプリケーションは、DMA が内部 VLAN タグ制御入力を使用する各パケットの通常ディスクリプタの前に、コンテキストディスクリプタを提供しなければなりません。

図 792 に、送信コンテキストディスクリプタのフォーマットを示します。*表 520*～*表 523* では、すべての送信コンテキストディスクリプタについて詳細に説明しています。

図 792. 送信コンテキストディスクリプタのフォーマット



- **TDES0 コンテキストディスクリプタ（読み出しフォーマット）**

表 520. TDES0 コンテキストディスクリプタ

ビット	名前	説明
31:0	TTSL	**送信パケットタイムスタンプロー** ワンステップ補正では、ドライバはこのディスクリプタワードで下位 32 ビットのタイムスタンプを提供できます。DMA は、ワンステップタイムスタンプ補正を行うための下位ワードとしてこの値を使用します。このフィールドは、TDES3 コンテキストディスクリプタの OSTC および TCMSSV ビットがセットされた場合のみ有効です。

- **TDES1 コンテキストディスクリプタ（読み出しフォーマット）**

表 521. TDES1 コンテキストディスクリプタ

ビット	名前	説明
31:0	TTSH	**送信パケットタイムスタンプハイ** ワンステップ補正では、ドライバはこのディスクリプタで上位 32 ビットのタイムスタンプを提供できます。DMA は、ワンステップタイムスタンプ補正を行うための上位ワードとしてこの値を使用します。このフィールドは、TDES3 コンテキストディスクリプタの OSTC および TCMSSV ビットがセットされた場合のみ有効です。

- **TDES2 コンテキストディスクリプタ（読み出しフォーマット）**

表 522. TDES2 コンテキストディスクリプタ

ビット	名前	説明
31:16	IVT	**内部 VLAN タグ** TDES3 コンテキストディスクリプタの IVLTV ビットがセットされ、TDES3 コンテキストディスクリプタの TCMSSV および OSTC ビットがリセットされている場合、TDES2[31:16] には後続の送信パケットに挿入される内部 VLAN タグが含まれます。
15:14	予約済み	
13:0	MSS	**最大セグメントサイズ** このセグメントサイズは、TCP/IP ペイロードを分割する際に使用されます。このフィールドは、TDES3 コンテキストディスクリプタの TCMSSV ビットがセットされ、TDES3 コンテキストディスクリプタの OSTC ビットがリセットされている場合のみ有効です。

- **TDES3 コンテキストディスクリプタ（読み出しフォーマット）**



表 523. TDES3 コンテキストディスクリプタ

ビット	名前	説明
31	OWN	**オウンビット** – 1：DMA がディスクリプタを所有しています。 – 0：アプリケーションがディスクリプタを所有しています。 DMA は、以下の条件のいずれかが当てはまる場合に、このビットをクリアします。 – DMA がパケット受信を完了した場合 – ディスクリプタに関連付けられたバッファがフルである場合
30	CTXT	**コンテキストタイプ** このビットはコンテキストディスクリプタでは '1' にセットする必要があります。
29:28	予約済み	
27	OSTC	**ワンステップタイムスタンプ補正有効化** このビットがセットされた場合、DMA は TDES0 および TDES1 で提供されたタイムスタンプ値を参照して、ワンステップタイムスタンプ補正を実行します。

表 523. TDES3 コンテキストディスクリプタ（続き）

ビット	名前	説明
26	TCMSSV	**ワンステップタイムスタンプ補正入力または MSS 有効** このビットおよび OSTC ビットがセットされた場合、TDES0 および TDES1 で提供されたタイムスタンプ補正入力が有効であることを示します。 OSTC ビットがリセットされ、このビットおよび TDES3 の TSE ビットが後続の通常ディスクリプタでセットされた場合、TDES2 の MSS 入力が有効であることを示します。
25:24	予約済み	
23	CDE	**コンテキストディスクリプタエラー** このビットがセットされた場合、コンテキストディスクリプタが正しくないシーケンスで提供され、DMA がそれを無視したことを示します。DMA は、コンテキストディスクリプタをクローズする際の書き戻し中に、このビットをセットします。
22:20	予約済み	
19:18	IVTIR	**内部 VLAN タグの挿入または書き換え** これらのビットがセットされた場合、パケットを送信する前に VLAN タグ付けまたはタグ削除を実行するよう MAC に要求します。パケットが VLAN タグについて変更された場合、MAC は CRC バイトを自動的に再計算して書き換えます。 このビットフィールドには以下の値があります。 – 00：内部 VLAN タグを追加しません。 – 01：送信前にパケットから内部 VLAN タグを削除します。このオプションは、VLAN フレームでのみ使用する必要があります。 – 10：*内部 VLAN 内包レジスタ（ETH_MACIVIR）*でプログラムされたタグ値またはコンテキストディスクリプタで内部 VLAN タグを挿入します。 – 11：*内部 VLAN 内包レジスタ（ETH_MACIVIR）*でプログラムされたタグ値またはコンテキストディスクリプタでパケットの内部 VLAN タグを書き換えます。このオプションは、VLAN フレームでのみ使用する必要があります。
17	IVLTV	**内部 VLAN タグ有効** このビットがセットされた場合、TDES2 の IVT フィールドが有効であることを示します。
16	VLTV	**VLAN タグ有効** このビットがセットされた場合、TDES3 の VT フィールドが有効であることを示します。
15:0	VT	**VLAN タグ** このフィールドは、パケットで挿入または書き換えられる VLAN タグを含みます。このフィールドは、*VLAN 内包レジスタ（ETH_MACVIR）*の VLTI ビットがリセットされたときのみ、VLAN として使用されます。

58.10.4 受信ディスクリプタ

Ethernet ペリフェラルの DAM は、テイルポインタがベースポインタや現在のポインタと異なる場合のみディスクリプタの読み出しを試みます。MAC によって受信される完全なパケットを最低 2 つ収容できる長さを持ったディスクリプタリングを準備することを推奨します。そうしないと、ディスクリプタの使用不能のために DMA のパフォーマンスが大きな影響を受けます。このような状況では、MTL RxFIFO がフルになり、パケットのドロップが開始します。

次の受信ディスクリプタが存在します。

- 読み出しフォーマットと書き戻しフォーマットを持つ通常ディスクリプタ
- コンテキストディスクリプタ

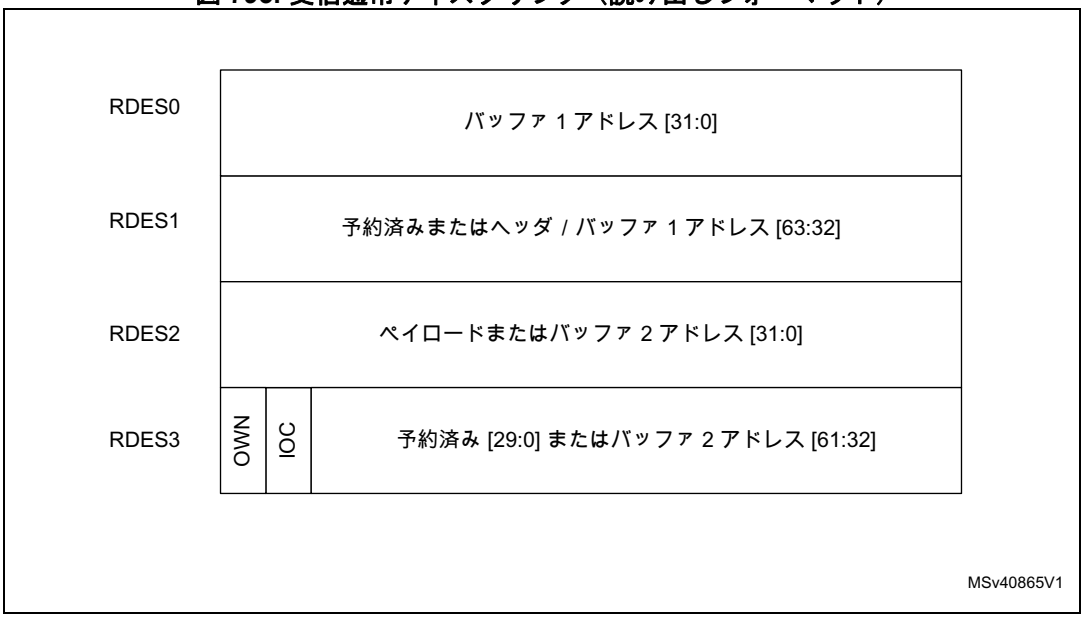
すべての受信したディスクリプタはソフトウェアによって準備され、DMA に「通常」ディスクリプタとして付与されます（それらの内容の詳細については、*図 793：受信通常ディスクリプタ（読み出しフォーマット）*を参照してください）。DMA は、このディスクリプタを読み出し、受信パケット（またはその一部）をディスクリプタによって示されたバッファに転送した後に、Rx DMA は対応するパケットステータスのディスクリプタをクローズします。ステータスフォーマットを*図 794：受信通常ディスクリプタ（書き戻しフォーマット）*に示します。

一部のパケットでは、通常ディスクリプタビットが足らず、完全なステータスを書き込めません。このようなパケットでは、Rx DMA は拡張ステータスを次のディスクリプタに（そのディスクリプタに埋め込まれたバッファポインタを処理したり使用したりせずに）書き込みます。この書き戻しディスクリプタのフォーマットおよび内容を*図 795：受信コンテキストディスクリプタ*に示します。

受信通常ディスクリプタ（読み出しフォーマット）

図 793 に、受信通常ディスクリプタの読み出しフォーマットを示します。*表 524*～*表 527* では、すべての受信通常ディスクリプタ（読み出しフォーマット）について詳細に説明しています。

図 793. 受信通常ディスクリプタ（読み出しフォーマット）



注： ***受信ディスクリプタ（読み出しフォーマット）で、バッファアドレスフィールドに 0 だけが含まれている場合、MAC はこのバッファにデータを転送せず、次のバッファまたは次のディスクリプタにスキップします。***

- **RDES0 通常ディスクリプタ（読み出しフォーマット）**

表 524. RDES0 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31:0	BUF1AP	**ヘッダまたはバッファ 1 アドレスポインタ** アプリケーションが、このバッファのバイト整列アドレスをプログラムでき、このフィールドの LS ビットを 0 以外にセットできることを示します。ただし、パケットの開始の転送中、DMA は RDES0[1:0]（64/128 ビット設定の場合は RDES0[2:0]/[3:0]）での書き込み動作を 0 で実行します。ただし、パケットデータは、バッファアドレスポインタで示される実際のオフセットによってシフトされます。 パケットの中央または最後の部分が格納されているバッファをアドレスポインタが指定する場合、DMA はオフセットアドレスを無視して、データ幅によって示された完全な位置に書き込みます。

- **RDES1 通常ディスクリプタ（読み出しフォーマット）**

表 525. RDES1 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31:0	予約済み	フィールドが予約済みです。

- **RDES2 通常ディスクリプタ（読み出しフォーマット）**

表 526. RDES2 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31:0	BUF2AP	**バッファ 2 アドレスポインタ** これらのビットはバッファ 2 物理アドレスを示します。 RxDMA は、パケットの開始バイトを転送する間だけ、ポインタアドレスの LS ビットを使用します。BUF2AP がパケットの中央または最後の部分が格納されているバッファのアドレスを提供している場合、DMA は RDES2[1:0]=0 を無視して完全な位置に書き込みます。

- **RDES3 通常ディスクリプタ（読み出しフォーマット）**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OWN	IOC	予約済み				BUF2V	BUF1V	予約済み							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み															

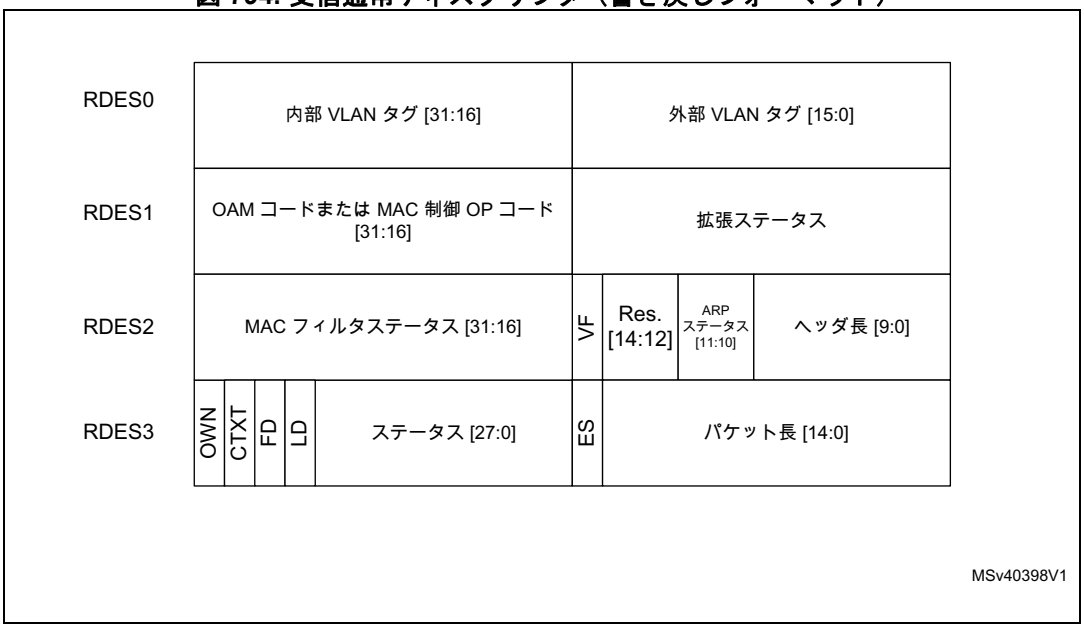
表 527. RDES3 通常ディスクリプタ（読み出しフォーマット）

ビット	名前	説明
31	OWN	**オウンビット** このビットがセットされた場合、DMA がディスクリプタを所有していることを示します。このビットがリセットされた場合、アプリケーションがディスクリプタを所有していることを示します。DMA は、以下の条件のいずれかが当てはまる場合に、このビットをクリアします。 – DMA がパケット受信を完了した場合 – ディスクリプタに関連付けられたバッファがフルである場合
30	IOC	**完了時割り込み有効化** このビットがセットされた場合、DMA がディスクリプタをクローズするときに割り込みがアプリケーションに対して発行されます。
29:26	予約済み	
25	BUF2V	**バッファ 2 アドレス有効** このビットがセットされた場合、RDES0 で指定されたバッファ 1 アドレスが有効であることを DMA に示します。アプリケーションは、RDES0 のバッファ 2 アドレスが指定しているアドレスを DMA が使用できるように、このビットをセットして受信パケットデータを書き込む必要があります。
24	BUF1V	**バッファ 1 アドレス有効** セット時、RDES1 で指定されたバッファ 1 アドレスが有効であることを DMA に示します。 アプリケーションは、RDES1 でバッファ 1 アドレスが指定するアドレスが DMA で使用できる場合に、この値をセットして受信パケットデータを書き込む必要があります。
23:0	予約済み	

受信通常ディスクリプタ（書き戻しフォーマット）

図 794 に、受信通常ディスクリプタの書き戻しフォーマットを示します。*表 528*～*表 531* では、すべての受信通常ディスクリプタ（書き戻しフォーマット）について詳細に説明しています。

図 794. 受信通常ディスクリプタ（書き戻しフォーマット）



- **RDES0 通常ディスクリプタ（書き戻しフォーマット）**

表 528. RDES0 通常ディスクリプタ（書き戻しフォーマット）

ビット	名前	説明
31:16	IVT	**内部 VLAN タグ** このフィールドには、RDES3 の RS0V ビットがセットされた場合に受信パケットの内部 VLAN タグが含まれます。
15:0	OVT	**外部 VLAN タグ** このフィールドには、RDES3 の RS0V ビットがセットされた場合に受信パケットの外部 VLAN タグが含まれます。

- **RDES1 通常ディスクリプタ（書き戻しフォーマット）**

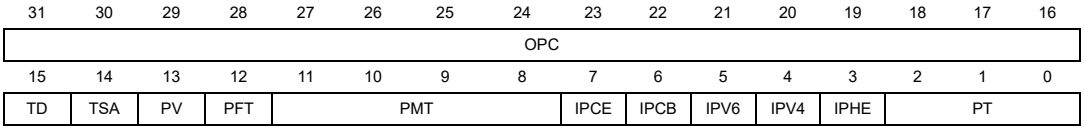


表 529. RDES1 通常ディスクリプタ（書き戻しフォーマット）(1)

ビット	名前	説明
31:16	OPC	**OAM サブタイプコード、または MAC 制御パケット OP コード** OAM サブタイプコード RDES3 のビット [18:16] が 111 にセットされている場合、このフィールドには OAM サブタイプおよびコードフィールドが入ります。 MAC 制御パケット OP コード RDES3 のビット [18:16] が 110 にセットされている場合、このフィールドには MAC 制御パケット OP コードフィールドが入ります。
15	TD	**タイムスタンプドロップ** このビットは、タイムスタンプがこのパケットに対してキャプチャされたが、オーバーフローのために MTL Rx FIFO でドロップしたことを示します。
14	TSA	**タイムスタンプ使用可能** タイムスタンプがある場合、このビットはタイムスタンプ値がコンテキストディスクリプタワード 2（RDES2）およびワード 1（RDES1）で使用可能なことを示します。これは、最終ディスクリプタビット（RDES3 [28]）がセットされた場合にのみ有効です。 コンテキストディスクリプタは、パケットの最終通常ディスクリプタの直後に、次のディスクリプタに書き込まれます。
13	PV	**PTP バージョン** 1：受信 PTP メッセージは IEEE 1588 バージョン 2 フォーマットです。 0：受信 PTP メッセージは IEEE 1588 バージョン 1 フォーマットです。
12	PFT	**PTP パケットタイプ** このビットは PTP メッセージが Ethernet を介して直接送信されることを示します。
11:8	PMT	**PTP メッセージタイプ** これらのビットは、受信メッセージのタイプを表すようにエンコードされます。 – 0000：受信 PTP メッセージなし – 0001：SYNC （全クロックタイプ） – 0010：Follow_Up （全クロックタイプ） – 0011：Delay_Req （全クロックタイプ） – 0100：Delay_Resp （全クロックタイプ） – 0101：Pdelay_Req （ピアツーピアトランスペアレントクロック） – 0110：Pdelay_Resp （ピアツーピアトランスペアレントクロック） – 0111：Pdelay_Resp_Follow_Up （ピアツーピアトランスペアレントクロック） – 1000：アナウンス – 1001：管理 – 1010：シグナリング – 1011～1110：予約済み – 1111：予約済みのある PTP パケットメッセージタイプ これらのビットは、タイムスタンプ機能を選択したときにのみ使用できます。

表 529. RDES1 通常ディスクリプタ（書き戻しフォーマット）(1)（続き）

ビット	名前	説明
7	IPCE	**IP ペイロードエラー** このビットがセットされた場合、以下のいずれかを示します。 – MAC によって計算された 16 ビット IP ペイロードチェックサム（すなわち、TCP、UDP、または ICMP チェックサム）が、受信セグメント内の対応するチェックサムフィールドに一致しません。 – TCP、UDP、または ICMP セグメント長が IP ヘッダフィールド内のペイロード長に一致しません。 – TCP、UDP、または ICMP セグメント長が TCP、UDP、または ICMP の最低許容セグメント長より短くなっています。 このビットがセットされると、RDES3 のビット 15（ES）はセットされません。
6	IPCB	**IP チェックサムのバイパス** このビットはチェックサムオフロードエンジンがバイパスされていることを示します。
5	IPV6	**IPv6 ヘッダあり** このビットは IPV6 ヘッダが検出されたを示します。
4	IPV4	**IPV4 ヘッダあり** このビットは IPV4 ヘッダが検出されたを示します。
3	IPHE	**IP ヘッダエラー** このビットがセットされた場合、以下のいずれかを示します。 – MAC によって計算された 16 ビット IPv4 ヘッダチェックサムが、受信チェックサムバイトに一致しません。 – IP データグラムバージョンが Ethernet タイプ値と一致していません。 – Ethernet パケットに、期待される IP ヘッダバイトの数がありません。 このビットは、ビット 5 またはビット 4 がセットされたときに有効です。
2:0	PT	**ペイロードタイプ** これらのビットは、受信チェックサムオフロードエンジン（COE）によって処理された IP データグラムでカプセル化されたペイロードのタイプを示します。 – 000：不明なタイプまたは処理されていない IP/AV ペイロード – 001：UDP – 010：TCP – 011：ICMP – その他：予約済み IP ヘッダエラーまたは断片化された IP があるため、COE が IP データグラムのペイロードを処理しない場合、これらのビットを 3'b000 にセットします。

1. 書き戻しフォーマットのステータスフィールドは、最終ディスクリプタ（RDES3[28] がセット）にのみ有効です。

- **RDES2 通常ディスクリプタ（書き戻しフォーマット）**

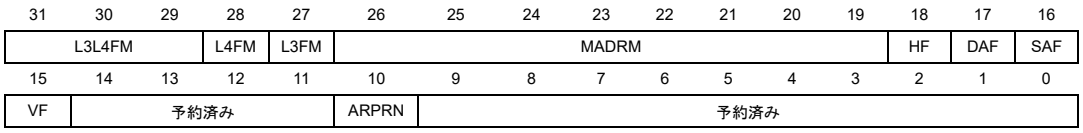


表 530. RDES2 通常ディスクリプタ（書き戻しフォーマット）

ビット	名前	説明
31:29	L3L4FM	**レイヤ 3 およびレイヤ 4 フィルタ数の一致** これらのビットは、受信パケットと一致したレイヤ 3 およびレイヤ 4 フィルタの数を示します。 – 000：フィルタ 0 – 001：フィルタ 1 – 010：フィルタ 2 – 011：フィルタ 3 – 100：フィルタ 4 – 101：フィルタ 5 – 110：フィルタ 6 – 111：フィルタ 7 このフィールドは、ビット 28 またはビット 27 がハイにセットされたときのみ有効です。複数のフィルタが一致する場合、これらのビットは最小フィルタの数を提供します。
28	L4FM	**レイヤ 4 フィルター致** このビットがセットされた場合、受信パケットが有効化されたレイヤ 4 ポート番号フィールドの 1 つと一致することを示します。以下の条件のいずれかが当てはまる場合のみ、このステータスが付与されます。 – レイヤ 3 フィールドが有効でなく、すべての有効化されたレイヤ 4 フィールドが一致する。 – すべての有効化されたレイヤ 3 およびレイヤ 4 フィルタフィールドが一致する。 複数のフィルタが一致する場合、このビットはビット [31:29] によって示されたフィルタのレイヤ 4 フィルタステータスを付与します。
27	L3FM	**レイヤ 3 フィルター致** このビットがセットされた場合、受信パケットが有効化されたレイヤ 3 IP アドレスフィールドの 1 つと一致することを示します。以下の条件のいずれかが当てはまる場合のみ、このステータスが付与されます。 – すべての有効化されたレイヤ 3が一致し、レイヤ 4 フィールドがバイパスされる。 – すべての有効化されたフィルタフィールドが一致する。 複数のフィルタが一致する場合、このビットはビット [31:29] によって示されたフィルタのレイヤ 3 フィルタステータスを付与します。
26:19	MADRM	**MAC アドレス一致またはハッシュ値** HF ビットがリセットされた場合、このフィールドに、受信パケットのデスティネーションアドレスと一致する MAC アドレスレジスタ番号が入ります。このフィールドは、DAF ビットがリセットされたときにのみ有効です。 HF ビットがセットされている場合、このフィールドには MAC によって計算されたハッシュ値が入ります。ハッシュ値に対応するビットがハッシュフィルタレジスタでセットされている場合、パケットがハッシュフィルタを通過します。
18	HF	**ハッシュフィルタステータス** このビットがセットされた場合、パケットが MAC アドレスハッシュフィルタを通過したことを示します。ビット [26:19] はハッシュ値を示します。
17	DAF	**デスティネーションアドレスフィルタ失敗** このビットがセットされた場合、パケットが MAC での DA フィルタに失敗したことを示します。

表 530. RDES2 通常ディスクリプタ（書き戻しフォーマット）（続き）

ビット	名前	説明
16	SAF	**SA アドレスフィルタ失敗** このビットがセットされた場合、パケットが MAC での SA フィルタに失敗したことを示します。
15	VF	**VLAN フィルタステータス** このビットがセットされた場合、受信パケットの VLAN タグが VLAN フィルタを通過したことを示します。
14:11	予約済み	
10	ARPNR	**ARP 応答の生成なし** このビットがセットされた場合、MAC が受信した ARP リクエストパケットに ARP 応答を生成しなかったことを示します。MAC が先の ARP リクエストへの ARP 応答の送信でビジー状態である場合、このビットがセットされます。
9:0	予約済み	

- **RDES3 通常ディスクリプタ（書き戻しフォーマット）**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OWN	CTXT	FD	LD	RS2V	RS1V	RS0V	CE	GP	RWT	OE	RE	DE	LT		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ES	PL														

表 531. RDES3 通常ディスクリプタ（書き戻しフォーマット）

ビット	名前	説明
31	OWN	**オウンビット** 1：DMA がディスクリプタを所有しています。 0：アプリケーションがディスクリプタを所有しています。 DMA は、以下の条件のいずれかが当てはまる場合に、このビットをクリアします。 – DMA がパケット受信を完了した場合 – ディスクリプタに関連付けられたバッファがフルである場合
30	CTXT	**受信コンテキストディスクリプタ** このビットがセットされた場合、現在のディスクリプタがコンテキストタイプのディスクリプタであることを示します。通常受信ディスクリプタでは、DMA がこのビットに ‘0’ を書き込みます。
29	FD	**先頭ディスクリプタ** このビットがセットされた場合、このディスクリプタがパケットの最初のバッファを含んでいることを示します。最初のバッファのサイズが 0 の場合、2 番目のバッファがパケットの開始を含みます。2番目のバッファのサイズも 0 の場合、次のディスクリプタがパケットの開始を含みます。
28	LD	**最終ディスクリプタ** このビットがセットされた場合、このディスクリプタが指定しているバッファが、パケットの最後のバッファであることを示します。
27	RS2V	**受信ステータス RDES2 有効** このビットがセットされた場合、RDES2 のステータスが有効で、DMA によって書き込まれることを示します。このビットは、RDES3 の LD ビットがセットされたときにのみ有効です。

表 531. RDES3 通常ディスクリプタ（書き戻しフォーマット）（続き）

ビット	名前	説明
26	RS1V	**受信ステータス RDES1 有効** このビットがセットされた場合、RDES1 のステータスが有効で、DMA によって書き込まれることを示します。このビットは、RDES3 の LD ビットがセットされたときにのみ有効です。
25	RS0V	**受信ステータス RDES0 有効** このビットがセットされた場合、RDES0 のステータスが有効で、DMA によって書き込まれることを示します。このビットは、RDES3 の LD ビットがセットされたときにのみ有効です。
24	CE	**CRC エラー** このビットがセットされた場合、受信パケットで巡回冗長検査（CRC）エラーが発生したことを示します。このフィールドは、RDES3 の LD ビットがセットされたときにのみ有効です。
23	GP	**ジャイアントパケット** このビットがセットされた場合、パケット長が 1518、1522、または 2000 バイト（ジャンボパケットが有効の場合は 9018 または 9022 バイト）の指定された最大 Ethernet サイズを超えることを示します。 ***注：　ジャイアントパケットは、パケット長のみ示します。パケットの切り詰めは発生しません。***
22	RWT	**受信ウォッチドッグタイムアウト** このビットがセットされた場合、現在のパケットの受信中に受信ウォッチドッグタイマが時間切れになったことを示します。現在のパケットは、ウォッチドッグタイムアウト後に切り詰められます。
21	OE	**オーバーフローエラー** このビットがセットされた場合、受信パケットが Rx FIFO でのバッファオーバーフローのため破損したことを示します。 ***注：　DMA が部分的なパケットをアプリケーションに転送した場合のみ、このビットがセットされます。これは、Rx FIFO が閾値モードで動作している場合のみ発生します。ストアアンドフォワードモードでは、すべての部分的なパケットは Rx FIFO で完全にドロップされます。***
20	RE	**受信エラー** このビットがセットされた場合、ETH_RX_DV 信号がパケット受信中にアサートされている間、ETH_RX_ER 信号がアサートされることを示します。
19	DE	**ドリブルビットエラー** このビットがセットされた場合、受信パケットに非整数倍（奇数ニブル）のバイトがあることを示します。このビットは、MII モードでのみ有効です。

表 531. RDES3 通常ディスクリプタ（書き戻しフォーマット）（続き）

ビット	名前	説明
18:16	LT	**長さ／タイプフィールド** このフィールドは、受信したパケットが長さパケットかタイプパケットかを示します。3 ビットのエンコーディングは次のとおりです。 – 000：パケットは、長さパケットです。 – 001：パケットは、タイプパケットです。 – 011：パケットは、ARP リクエストパケットタイプです。 – 100：パケットは、VLAN タグ付きタイプパケットです。 – 101：パケットは、ダブル VLAN タグ付きタイプパケットです。 – 110：パケットは、MAC 制御パケットタイプです。 – 111：パケットは、OAM パケットタイプです。 – 010：予約済み
15	ES	**エラー概要** – このビットがセットされた場合、以下のビットの論理 OR を示します。 – RDES3[24]：CRC エラー – RDES3[19]：ドリブルエラー – RDES3[20]：受信エラー – RDES3[22]：ウォッチドッグタイムアウト – RDES3[21]：オーバーフローエラー – RDES3[23]：ジャイアントパケット このフィールドは、RDES3 の LD ビットがセットされたときにのみ有効です。
14:0	PL	**パケット長** これらのビットは、システムメモリに転送された受信パケットのバイト長（CRC を含む）を示します。 このフィールドは、RDES3 の LD ビットがセットされ、ディスクリプタエラー（RDES3[13]）またはオーバーフローエラービットがリセットされたときに有効です。パケット長には、IP チェックサム計算が有効で、受信パケットが MAC 制御パケットではないときに Ethernet パケットに付加される 2 バイトも含まれます。 このフィールドは、RDES3 の LD ビットがセットされたときに有効です。最終ディスクリプタビットとエラー概要ビットがセットされていない場合、このフィールドは、現在のパケットで転送された累積バイト数を示します。

受信コンテキストディスクリプタ

このディスクリプタは、アプリケーションに対して読み出し専用です。このディスクリプタは、DMA によってのみ書き込みできます。

コンテキストディスクリプタは、最後に受信したパケットに関連する拡張ステータスについての情報を提供します。RDES3 のビット 30 は、コンテキストタイプのディスクリプタを示します。

図 795 に、受信コンテキストディスクリプタのフォーマットを示します。*表 532*～*表 535* では、すべての受信コンテキストディスクリプタについて詳細に説明しています。

図 795. 受信コンテキストディスクリプタ



- **RDES0 コンテキストディスクリプタ**

表 532. RDES0 コンテキストディスクリプタ

ビット	名前	説明
31:0	RTSL	**受信パケットタイムスタンプロー** DMA は、このフィールドを、対応する受信パケットについてキャプチャされたタイムスタンプの下位 32 ビットで更新します。このフィールドと RDES1 の RTSH フィールドがすべて 1 を示している場合、タイムスタンプは破損しているものとみなす必要があります。

- **RDES1 コンテキストディスクリプタ**

表 533. RDES1 コンテキストディスクリプタ

ビット	フィールド	説明
31:0	RTSH	**受信パケットタイムスタンプハイ** DMA は、このフィールドを、対応する受信パケットについてキャプチャされたタイムスタンプの上位 32 ビットで更新します。このフィールドと RDES0 の RTSL フィールドがすべて 1 の値を示している場合、タイムスタンプは破損しているものとみなす必要があります。

- **RDES2 コンテキストディスクリプタ**

表 534. RDES2 コンテキストディスクリプタ

ビット	説明
31:0	予約済み

- **RDES3 コンテキストディスクリプタ**

表 535. RDES3 コンテキストディスクリプタ

ビット	名前	説明
31	OWN	**オウンビット** 1：DMA がディスクリプタを所有しています。 0：アプリケーションがディスクリプタを所有しています。 DMA は、以下の条件のいずれかが当てはまる場合に、このビットをクリアします。 – DMA がパケット受信を完了した場合 – ディスクリプタに関連付けられたバッファがフルである場合
30	CTXT	**受信コンテキストディスクリプタ** このビットがセットされた場合、現在のディスクリプタがコンテキストディスクリプタであることを示します。コンテキストディスクリプタでは、DMA がこのビットに 1'b1 を書き込みます。
29:0	予約済み	

参考資料

58.11 Ethernet レジスタの説明

58.11.1 レジスタマップ

このセクションでは、以下のレジスタマップを提供します。

- DMA レジスタ（*セクション 58.11.2：Ethernet DMA レジスタ*を参照）
- MTL レジスタ（*セクション 58.11.3：Ethernet MTL レジスタ*を参照）
- MMC レジスタを含む MAC レジスタ（*セクション 58.11.4：Ethernet MAC および MMC レジスタ*を参照）

58.11.2 Ethernet DMA レジスタ

DMA モードレジスタ（ETH_DMAMR）

アドレスオフセット：0x1000

リセット値：0x0000 0000

DMA モードレジスタは、DMA のバス動作モードを確立します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTM[1:0]	
														rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PR[2:0]			TXPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.			DA	SWR
	r			r										r	rw

ビット 31:18	予約済みであり、リセット値に保持する必要があります。
ビット 17:16	**INTM[1:0]**：割り込みモード このフィールドは Ethernet ペリフェラルの割り込みモードを定義します。 割り込み信号および ETH_DMACSR レジスタの RI/TI ビットの動作は、INTM 値に依存して変化します（*表 511：転送完了割り込みの動作*を参照）。
ビット 15	予約済みであり、リセット値に保持する必要があります。
ビット 14:12	**PR[2:0]**：優先比率 これらのビットは、Rx DMA と Tx DMA の間の加重ラウンドロビン方式アービトレーションの優先比率を制御します。これらのビットは、DA ビットがリセットされたときのみ有効です。優先比率は、TXPR ビットがリセットされているかセットされているかに応じて、Rx:Tx または Tx:Rx となります。 000：優先比率は、1:1 です。 001：優先比率は、2:1 です。 010：優先比率は、3:1 です。 011：優先比率は、4:1 です。 100：優先比率は、5:1 です。 101：優先比率は、6:1 です。 110：優先比率は、7:1 です。 111：優先比率は、8:1 です。

ビット 11	**TXPR**：送信優先順位 セットされると、このビットは、システム側バスのアービトレーションの間、Tx DMA が Rx DMA より高い優先順位であることを示します。
ビット 10:5	予約済みであり、リセット値に保持する必要があります。
ビット 1	**DA**：DMA Tx または Rx アービトレーション構成 このビットは、すべてのチャネルの送信パスと受信パスのアービトレーション構成を指定します。 0：Rx:Tx または Tx:Rx での加重ラウンドロビン パス間の優先順位がビット [14:12] で指定された優先順位に従い、優先順位の重みが TXPR ビットで指定されます。 1：固定優先順位 TXPR ビットがセットされた場合、Tx パスは Rx パスより優先されます。それ以外は、Rx パスが Tx パスより優先されます。
ビット 0	**SWR**：ソフトウェアリセット このビットがセットされると、MAC および DMA コントローラが DMA、MTL、および MAC のロジックおよびすべての内部レジスタをリセットします。このビットは、リセット動作がすべてのクロックドメインで完了した後に自動でクリアされます。レジスタを再プログラムする前に、このビットで 0 の値を読み出す必要があります。 ***注：*** ***すべてのアクティブなクロックドメインのすべてのリセットがネゲートされたときのみ、リセット動作が完了します。そのため、（選択された PHY インタフェースに適用可能な）すべての PHY 入力クロックがソフトウェアリセット完了に対して存在する必要があります。ソフトウェアリセット動作を完了するための時間は、最も遅いアクティブクロックの周波数に依存します。***

システムバスモードレジスタ（ETH_DMASBMR）

アドレスオフセット：0x1004

リセット値：0x0101 0000

システムバスモードレジスタは、AHB マスタの動作を制御します。未処理リクエストのバーストスプリットおよび数を主に制御します。

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

ビット 31	予約済みであり、リセット値に保持する必要があります。
ビット 29:28	予約済みであり、リセット値に保持する必要があります。
ビット 23:16	予約済みであり、リセット値に保持する必要があります。
ビット 15	**RB**：INCRx バーストの再構築 このビットがハイにセットされ、AHB マスタが SPLIT、RETRY、または早期バースト終了（EBT）レスポンスを受信した場合、AHB マスタインタフェースは、開始されたバースト転送のペンディング中のビートを、INCRx および SINGLE 転送で再構築します。デフォルトでは、AHB マスタインタフェースは EBT のペンディング中のビートを、未指定（INCR）バーストで再構築します。
ビット 14	**MB**：混合バースト このビットがハイにセットされ、FB ビットがローの場合、AHB マスタは 16 以上のバースト長に対して未定義バースト転送（INCR）を実行します。16 未満のバースト長の場合、AHB マスタは固定バースト転送（INCRx および SINGLE）を実行します。
ビット 13	予約済みであり、リセット値に保持する必要があります。
ビット 12	**AAL**：アドレス整列ビート このビットが 1 にセットされた場合、マスタは読み出しおよび書き込みチャネルでアドレス整列バースト転送を実行します。
ビット 11:1	予約済みであり、リセット値に保持する必要があります。
ビット 0	**FB**：固定バースト長 このビットが 1 にセットされた場合、AHB マスタは指定された長さのバースト転送（INCRx または SINGLE）を開始します。 このビットが 0 にセットされた場合、AHB マスタは未指定の長さ（INCR）の転送または SINGLE 転送を開始します。

割り込みステータスレジスタ（ETH_DMAISR）

アドレスオフセット：0x1008

リセット値：0x0000 0000

アプリケーションは、割り込みサービスルーチン中またはポーリング中に、この割り込みステータスを読み出して、DMA チャネル、MTL キュー、MAC の割り込みステータスを決定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MACIS	MTLIS
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DC0IS
															r

ビット 31:18	予約済み
ビット 17	**MACIS**：MAC 割り込みステータス このビットは、MAC の割り込みイベントを示します。このビットを 1'b0 にリセットするために、ソフトウェアは MAC で対応するレジスタを読み出して、割り込みの正確な原因を取得し、そのソースをクリアしなければなりません。
ビット 16	**MTLIS**：MTL 割り込みステータス このビットは、MTL の割り込みイベントを示します。このビットを 1'b0 にリセットするために、ソフトウェアは MTL で対応するレジスタを読み出して、割り込みの正確な原因を取得し、そのソースをクリアしなければなりません。
ビット 15:1	予約済みであり、リセット値に保持する必要があります。
ビット 0	**DC0IS**：DMA チャネル割り込みステータス このビットは、DMA チャネルの割り込みイベントを示します。このビットを ‘0’ にリセットするために、ソフトウェアは DMA チャネルで対応するレジスタを読み出して、割り込みの正確な原因を取得し、そのソースをクリアしなければなりません。

デバッグステータスレジスタ（ETH_DMADSR）

アドレスオフセット：0x100C

リセット値：0x0000 0000

デバッグステータスレジスタは、デバッグの目的で DMA チャネルの受信および送信プロセスステータスを提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				Res.			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPS0[3:0]				RPS0[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	AXW HSTS
r				r											r

ビット 31:16	予約済みであり、リセット値に保持する必要があります。
ビット 15:12	**TPS0[3:0]**：DMA チャネル 送信プロセス状態 このフィールドは、チャネルの Tx DMA FSM 状態を示します。 000：停止（送信リセットまたは送信停止コマンドの発行） 001：実行中（Tx 転送ディスクリプタのフェッチ） 010：実行中（ステータス待ち） 011：実行中（システムメモリバッファからデータの読み出し、および Tx バッファ（TxFIFO）のキューへの挿入） 100：タイムスタンプ書き込み状態 101：将来の使用のために予約済み。 110：中断（Tx ディスクリプタ使用不可、または Tx バッファアンダーフロー） 111：実行中（送信ディスクリプタのクローズ） このフィールドの MSB は必ず 0 に戻ります。このフィールドは割り込みを生成しません。
ビット 11:8	**RPS0[3:0]**：DMA チャネル受信プロセス状態 このフィールドは、チャネルの Rx DMA FSM 状態を示します。 000：停止（受信リセットまたは受信停止コマンドの発行） 001：実行中（Rx 転送ディスクリプタのフェッチ） 010：将来の使用のために予約済み。 011：実行中（Rx パケット待ち） 100：中断（Rx ディスクリプタ使用不可） 101：実行中（Rx ディスクリプタのクローズ） 110：タイムスタンプ書き込み状態 111：実行中（Rx パケットデータの Rx バッファからシステムメモリへの転送） このフィールドの MSB は必ず 0 に戻ります。このフィールドは割り込みを生成しません。
ビット 7:1	予約済みであり、リセット値に保持する必要があります。
ビット 0	**AXWHSTS**：AHB マスタ書き込みチャネル ハイの場合、このビットは AHB マスタ FMS の書き込みチャネルがアイドル状態ではないことを示します。

チャネル 制御レジスタ（ETH_DMACCR）

アドレスオフセット：0x1100

リセット値：0x0000 0000

DMA チャネル 制御レジスタは、セグメンテーションの MSS 値、2 つのディスクリプタ間でスキップする長さを指定し、ヘッダスプリットおよび 8xPBL モードなども備えています。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSL[2:0]			Res.	PBL X8
											rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MSS[13:0]													
		rw													

ビット 31:24	予約済みであり、リセット値に保持する必要があります。
ビット 23:21	予約済みであり、リセット値に保持する必要があります。
ビット 20:18	**DSL[2:0]**：ディスクリプタスキップ長 このビットは、2 つの連結されていないディスクリプタ間でスキップする 32 ビットワードを指定します。アドレススキッピングは、現在のディスクリプタの終わりから始まり、次のディスクリプタの始めまでです。 DSL 値が 0 である場合、DMA は連続したディスクリプタテーブルを取得します。
ビット 17	予約済みであり、リセット値に保持する必要があります。
ビット 16	**PBLX8**：8xPBL モード このビットがセットされると、ETH_DMACTxCR のビット [21:16] にプログラムされた PBL 値は 8 倍になります。したがって、DMA は、PBL の値に応じて、8、16、32、64、128、および 256 ビートでデータを転送します。
ビット 15:14	予約済みであり、リセット値に保持する必要があります。
ビット 13:0	**MSS[13:0]**：最大セグメントサイズ このフィールドは、パケットの分割時に使用される最大セグメントサイズを指定します。このフィールドは、ETH_DMACTxCR レジスタの TSE ビットがセットされたときにのみ有効です。 このフィールドにプログラムされた値は、バイトで設定されたデータ幅より大きくなければなりません。64 バイト以上の MSS 値を使用することを推奨します。 このフィールドは、TCP/IP パケットの TCP セグメンテーションオフロード有効化オプションが選択されていないときは予約済みとなります。

チャネル 送信制御レジスタ（ETH_DMACTxCR）

アドレスオフセット：0x1104

リセット値：0x0000 0000

DMA チャネル 送信制御レジスタは、PBL、TCP セグメンテーション、Tx チャネルの重みなど Tx 機能を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.				Res.	Res.	TXPBL[5:0]					
										rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OSF	Res.			ST
			rw								rw				rw

ビット 31:22	予約済みであり、リセット値に保持する必要があります。
ビット 21:16	**TXPBL[5:0]**：送信のプログラム可能なバースト長 これらのビットは、1 回の DMA データ転送で転送される最大ビート数を示します。これは、1 回のブロック読み出しまたは書き込みで使用される最大値です。DMA は、常に、アプリケーションバス上でバースト転送を開始するたびに、PBL での指定に従ってバーストを試みます。PBL は、次の値でプログラムできます。1、2、4、8、16 または 32 です。その他の値は、未定義の動作をもたらします。 32 ビートを超える転送では、以下の手順を実行します。 1.ETH_DMACCR で PBLx8 モードをセットします。 2.PBL をセットします。
ビット 15:13	予約済みであり、リセット値に保持する必要があります。
ビット 12	**TSE**：TCP セグメンテーション有効 このビットがセットされた場合、DMA はチャネル i のパケットに TCP セグメンテーションを実行します。TCP セグメンテーションは、TSE ビット（TDES0[19]）が Tx 通常ディスクリプタでセットされているパケットにのみ実行されます。このビットがセットされた場合、TxPBL 値は 4 以上でなければなりません。 このフィールドは、TCP/IP パケットの TCP セグメンテーションオフロード有効化オプションが選択されていないときは予約済みとなります。
ビット 11:5	予約済みであり、リセット値に保持する必要があります。
ビット 4	**OSF**：2 番目のパケットに対する動作 このビットがセットされた場合、最初のパケットのステータスが取得される前でも、送信データの 2 番目のパケットを処理するように DMA に指示します。
ビット 3:1	予約済みであり、リセット値に保持する必要があります。

ビット 0	**ST**：送信開始または送信停止コマンド このビットがセットされた場合、送信が実行中状態に置かれます。DMA は、送信パケットの現在位置での送信リストをチェックします。 DMA は、以下のいずれかの位置からディスクリプタの取得を試みます。 – リスト内の現在位置：これは、ETH_DMACTxDLAR レジスタによってセットされた送信リストのベースアドレスです。 – 前に送信が停止された位置 DMA が現在のディスクリプタを所有していない場合、送信はサスペンド状態に入り、ETH_DMACSR の TBU ビットがセットされます。送信開始コマンドは、送信が停止されたときにのみ有効となります。ETH_DMACTxDLAR レジスタをセットする前にコマンドが発行された場合、DMA の動作は予想できません。 このビットがリセットされると、送信プロセスは現在のパケットの送信を終了した後、停止状態に置かれます。送信リスト内の次のディスクリプタの位置が保存され、送信がリスタートされるときに現在位置となります。リストアドレスを変更するには、このビットがリセットされたときに ETH_DMACTxDLAR レジスタを新しい値でプログラムする必要があります。新しい値は、このビットが再びセットされたときに考慮されます。送信停止コマンドは、現在のパケットの送信が完了したとき、または送信がサスペンド状態にあるときにのみ有効です。

チャネル 受信制御レジスタ（ETH_DMACRxCR）

アドレスオフセット：0x1108

リセット値：0x0000 0000

DMA チャネル 受信制御レジスタは、PBL、バッファサイズ、拡張ステータスなど Rx 機能を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RPF	Res.	Res.	Res.	Res.				Res.	Res.	RXPBL[5:0]					
rw										rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RBSZ														SR
	rw														rw

ビット 31	**RPF**：DMA Rx チャネルパケット一掃 このビットが 1 にセットされた場合、DMA は自動的に DMA Rx チャネルに指定された Rx キューからパケットを一掃します（DMA Rx チャネル がシステムバスエラー発生後に停止した場合）。 この一掃は、Rx キューの読み出し側で発生します。 このビットが 0 にセットされた場合、EQOS は DMA がシステムバスエラーによって停止された後、DMA Rx チャネルに指定された Rx キューのパケットを一掃しません。
ビット 30:22	予約済みであり、リセット値に保持する必要があります。
ビット 21:16	**RXPBL[5:0]**：受信のプログラム可能なバースト長 これらのビットは、1 回の DMA データ転送で転送される最大ビート数を示します。これは、1 回のブロック読み出しまたは書き込みで使用される最大値です。DMA は、常に、アプリケーションバス上でバースト転送を開始するたびに、PBL での指定に従ってバーストを試みます。PBL は、次の値でプログラムできます。1、2、4、8、16 または 32 です。その他の値は、未定義の動作をもたらします。 32 ビートを超える転送では、以下の手順を実行します。 1.ETH_DMACCR で PBLx8 モードをセットします。 2.PBL をセットします。
ビット 15	予約済みであり、リセット値に保持する必要があります。
ビット 14:4	**RBSZ**：受信バッファサイズ このフィールドは、Rx バッファのサイズをバイト単位で示します。最大バッファサイズは 16 KB に制限されます。スプリットヘッダが有効になっている場合、バッファサイズはペイロードバッファに適用されます。 ***注：*** ***バッファサイズは、バス幅（32、64、または 128 ビット）に応じて、それぞれ 4、8、または 16 の倍数でなければなりません。バッファアドレスポインタの値がバス幅に整列されていない場合でも、これは必須となります。バッファサイズが 4、8、または 16 の倍数でない場合、未定義の動作が発生する可能性があります。*** ***LSB ビット（32 ビット、64 ビット、または 128 ビットバス幅に対して 1:0、2:0、または 3:0）は無視され、DMA は LSB ビットを内部的にすべて 0 とみなします。したがって、これらの LSB ビットは読み出し専用（RO）です。***
ビット 3:1	予約済みであり、リセット値に保持する必要があります。

ビット 0	**SR**：受信開始または停止 このビットがセットされた場合、DMA は受信リストからのディスクリプタの取得を試みて、着信パケットを処理します。 DMA は、以下のいずれかの位置からディスクリプタの取得を試みます。 – リスト内の現在位置 : これは、ETH_DMACRxDLAR レジスタによってセットされたアドレスです。 – 前に Rx プロセスが停止された位置 DMA が現在のディスクリプタを所有していない場合、受信はサスペンドされ、ETH_DMACSR の RBU ビットがセットされます。受信開始コマンドは、受信が停止されたときにのみ有効となります。ETH_DMACRxDLAR レジスタをセットする前にコマンドが発行された場合、DMA の動作は予想できません。 このビットがリセットされると、Rx DMA の動作は、現在のパケットの転送後に停止されます。受信リスト内の次のディスクリプタの位置が保存され、Rx プロセスがリスタートされた後に現在位置となります。受信停止コマンドは、Rx プロセスが実行中（Rx パケットを待機中）またはサスペンド状態にあるときにのみ有効です。

チャネル Tx ディスクリプタリストアドレスレジスタ（ETH_DMACTxDLAR）

アドレスオフセット：0x1114

リセット値：0x0000 0000

チャネルディスクリプタリストアドレスレジスタは、DMA に送信ディスクリプタリストの開始を示します。ディスクリプタリストは、アプリケーションの物理メモリ空間にあり、ワード整列されていなければなりません。DMA は、対応する LSB をローにセットすることによって、内部でバス幅で整列されたアドレスに変換します。

Tx DMA が停止したとき（ETH_DMACiTxCR レジスタで ST ビットが 0 にセットされたとき）のみ、このレジスタに書き込むことができます。停止された場合、新しいディスクリプタリストアドレスをこのレジスタに書き込みできます。ST ビットが 1 にセットされた場合、DMA は新しくプログラムされたディスクリプタベースアドレスを取得します。ST ビットが 0 にセットされているときに、このレジスタが変更されない場合、DMA は前に停止した位置のディスクリプタアドレスを取得します。

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

ビット 31:2	**TDESLA**：送信リストの開始 このフィールドは、送信ディスクリプタリストの先頭ディスクリプタのベースアドレスを含みます。DMA は、32 ビットバス幅に対して LSB ビット（1:0、2:0、または 3:0）を無視し、これらのビットを内部的にすべて 0 とみなします。したがって、これらの LSB ビットは読み出し専用（RO）です。このフィールドの幅は設定に依存します。 32 ビット設定の場合は 31:2 です。
ビット 1:0	予約済みであり、リセット値に保持する必要があります。

チャネル Rx ディスクリプタリストアドレスレジスタ（ETH_DMACRxDLAR）

アドレスオフセット：0x111C

リセット値：0x0000 0000

チャネル Rx ディスクリプタリストアドレスレジスタは、DMA に受信ディスクリプタリストの開始を示します。

このレジスタは、受信ディスクリプタリストの開始を示します。ディスクリプタリストは、アプリケーションの物理メモリ空間にあり、ワード整列されていなければなりません。DMA は、対応する LS ビットをローにセットすることによって、内部でバス幅で整列されたアドレスに変換します。このレジスタへの書き込みは、受信が停止されたときにのみ許されます。受信が停止された場合、受信開始コマンドが与えられる前に、このレジスタに書き込まなければなりません。Rx DMA が停止したとき（ETH_DMACRxCR レジスタで SR ビットが 0 にセットされたとき）のみ、このレジスタに書き込むことができます。停止された場合、新しいディスクリプタリストアドレスをこのレジスタに書き込みできます。

SR ビットが 1 にセットされた場合､DMA は新しくプログラムされたディスクリプタベースアドレスを取得します。

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

ビット 31:2	**RDESLA[28:0]**：受信リストの開始 このフィールドは、Rx ディスクリプタリストの先頭ディスクリプタのベースアドレスを含みます。DMA は、32 ビットバス幅に対して LSB ビット（1:0、2:0、または 3:0）を無視し、これらのビットを内部的にすべて 0 とみなします。したがって、これらの LSB ビットは読み出し専用（RO）です。 このフィールドの幅は設定に依存します。 32 ビット設定の場合は 31:2 です。
ビット 1:0	予約済みであり、リセット値に保持する必要があります。

チャネル Tx ディスクリプタテイルポインタレジスタ（ETH_DMACTxDTPR）

アドレスオフセット：0x1120

リセット値：0x0000 0000

チャネルTx ディスクリプタテイルポインタレジスタは、ベースからのオフセットを示し、最後の有効なディスクリプタの位置を示します。

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

ビット 31:2	**TDT**：送信ディスクリプタテイルポインタ このフィールドは、Tx ディスクリプタリングのテイルポインタを含みます。このソフトウェアは、Tx チャネルにディスクリプタを追加するためにテイルポインタを書き込みます。ハードウェアは、ヘッドとテイルポインタレジスタの間のディスクリプタによって参照されるすべてのパケットの送信を試みます。 このフィールドの幅は設定に依存します。 32 ビット設定の場合は 31:2 です。
ビット 1:0	予約済みであり、リセット値に保持する必要があります。

チャネル Rx ディスクリプタテイルポインタレジスタ（ETH_DMACRxDTPR）

アドレスオフセット：0x1128

リセット値：0x0000 0000

チャネル Rx ディスクリプタテイルポインタは、ベースからのオフセットを示し、最後の有効なディスクリプタの位置を示します。

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

ビット 31:2	**RDT**：受信ディスクリプタテイルポインタ このフィールドは、Rx ディスクリプタリングのテイルポインタを含みます。このソフトウェアは、Rx チャネルにディスクリプタを追加するためにテイルポインタを書き込みます。ハードウェアは、ヘッドとテイルポインタレジスタの間で参照されるディスクリプタにすべての受信パケットの書き込みを試みます。 このフィールドの幅は設定に依存します。 32 ビット設定の場合は 31:2 です。
ビット 1:0	予約済みであり、リセット値に保持する必要があります。

チャネル Tx ディスクリプタリング長レジスタ（ETH_DMACTxRLR）

アドレスオフセット：0x112C

リセット値：0x0000 0000

Tx ディスクリプタリング長レジスタには、送信ディスクリプタリングの長さが含まれます。

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

ビット 31:10	予約済みであり、リセット値に保持する必要があります。

ビット 9:0	**TDRL[9:0]**：送信ディスクリプタリング長 このフィールドは、循環ディスクリプタリングでの Tx の最大数をセットします。ディスクリプタの最大数は 1000 個に制限されます。Synopsys は、最小リングディスクリプタ長を 4 とすることを推奨します。

チャネル Rx ディスクリプタリング長レジスタ（ETH_DMACRxRLR）

アドレスオフセット：0x1130

リセット値：0x0000 0000

チャネル Rx ディスクリプタリング長レジスタには、受信ディスクリプタ循環リングの長さが含まれます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	RDRL[9:0]									
						rw									

ビット 31:10	予約済みであり、リセット値に保持する必要があります。
ビット 9:0	**RDRL[9:0]**：受信ディスクリプタリング長 このレジスタは、循環ディスクリプタリングでの Rx の最大数をセットします。ディスクリプタの最大数は 1000 個に制限されます。

チャネル割り込み有効レジスタ（ETH_DMACIER）

アドレスオフセット：0x1134

リセット値：0x0000 0000

チャネル割り込み有効レジスタは、ステータスレジスタによって報告された割り込みを有効にします。

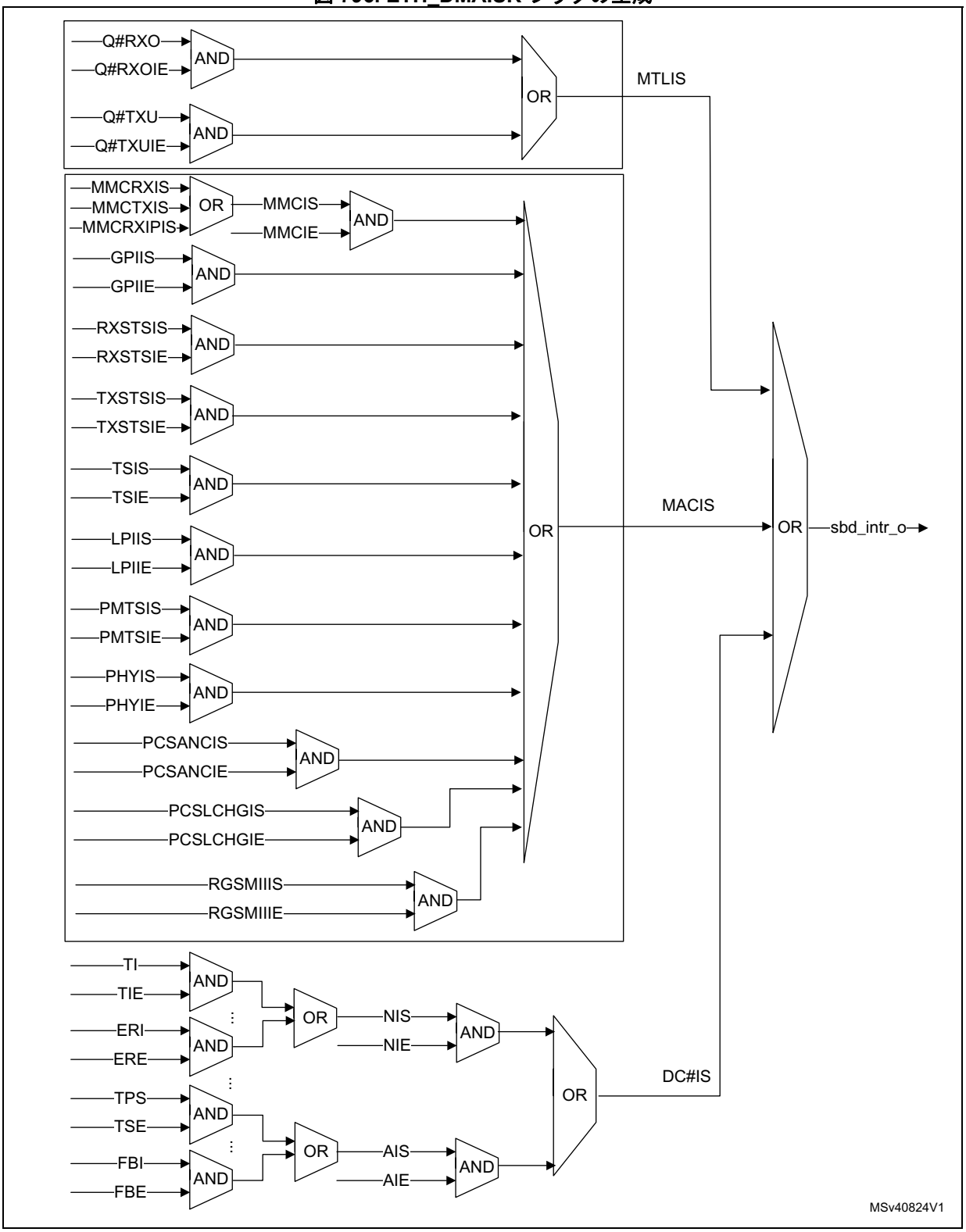
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NIE	AIE	CDEE	FBEE	ERIE	ETIE	RWTE	RSE	RBUE	RIE	Res.	Res.	Res.	TBUE	TXSE	TIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw

ビット 31:16	予約済みであり、リセット値に保持する必要があります。

ビット 15	**NIE**：通常割り込み要約有効化 このビットがセットされた場合、通常割り込み要約が有効になります。このビットは、ETH_DMACSR で次の割り込みを有効にします。 ビット 0：送信割り込み ビット 2：送信バッファ使用不可 ビット 6：受信割り込み ビット 11：アーリー受信割り込み このビットがリセットされた場合、通常割り込み要約が無効になります。
ビット 14	**AIE**：異常割り込み要約有効化 このビットがセットされた場合、異常割り込み要約が有効になります。このビットは、ETH_DMACSR で次の割り込みを有効にします。 ビット 1：送信プロセス停止 ビット 7：Rx バッファ使用不可 ビット 8：受信プロセス停止 ビット 9：受信ウォッチドッグタイムアウト ビット 10：アーリー送信割り込み ビット 12：致命的バスエラー このビットがリセットされた場合、異常割り込み要約が無効になります。
ビット 13	**CDEE**：コンテキストディスクリプタエラー有効化 このビットが AIE ビットとともにセットされた場合、コンテキストディスクリプタエラー割り込みが有効になります。このビットがリセットされると、コンテキストディスクリプタエラー割り込みが無効になります。
ビット 12	**FBEE**：致命的バスエラー有効化 このビットが AIE ビットとともにセットされた場合、致命的バスエラー割り込みが有効になります。このビットがリセットされると、致命的バスエラー割り込みが無効になります。
ビット 11	**ERIE**：アーリー受信割り込み有効化 このビットが NIE ビットとともにセットされた場合、アーリー受信割り込みが有効になります。このビットがリセットされると、アーリー受信割り込みが無効になります。
ビット 10	**ETIE**：アーリー送信割り込み有効化 このビットが AIE ビットとともにセットされた場合、アーリー送信割り込みが有効になります。このビットがリセットされると、アーリー送信割り込みが無効になります。
ビット 9	**RWTE**：受信ウォッチドッグタイムアウト有効化 このビットが AIE ビットとともにセットされた場合、受信ウォッチドッグタイムアウト割り込みが有効になります。このビットがリセットされると、受信ウォッチドッグタイムアウト割り込みが無効になります。
ビット 8	**RSE**：受信停止有効化 このビットが AIE ビットとともにセットされた場合、受信停止割り込みが有効になります。このビットがリセットされると、受信停止割り込みが無効になります。
ビット 7	**RBUE**：受信バッファ使用不可有効化 このビットが AIE ビットとともにセットされた場合、受信バッファ使用不可割り込みが有効になります。このビットがリセットされると、受信バッファ使用不可割り込みが無効になります。
ビット 6	**RIE**：受信割り込み有効化 このビットが NIE ビットとともにセットされた場合、受信割り込みが有効になります。このビットがリセットされると、受信割り込みが無効になります。
ビット 5:3	予約済みであり、リセット値に保持する必要があります。

ビット 2	**TBUE**：送信バッファ使用不可有効化 このビットが NIE ビットとともにセットされた場合、送信バッファ使用不可割り込みが有効になります。このビットがリセットされると、送信バッファ使用不可割り込みが無効になります。
ビット 1	**TXSE**：送信停止有効化 このビットが AIE ビットとともにセットされた場合、送信停止割り込みが有効になります。このビットがリセットされると、送信停止割り込みが無効になります。
ビット 0	**TIE**：送信割り込み有効化 このビットが NIE ビットとともにセットされた場合、送信割り込みが有効になります。このビットがリセットされると、送信割り込みが無効になります。

図 796. ETH_DMAISR フラグの生成



チャネル Rx 割り込みウォッチドッグタイマレジスタ（ETH_DMACRxIWTR）

アドレスオフセット：0x1138

リセット値：0x0000 0000

受信割り込みウォッチドッグタイマレジスタは、DMA からの受信割り込み（RI）のウォッチドッグタイムアウトを示します。このレジスタに 0 以外の値を書き込むと、ETH_DMACSR レジスタの RI ビットに対してウォッチドッグタイマを有効にします。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWT[7:0]							
								rw							

ビット 31:8	予約済みであり、リセット値に保持する必要があります。
ビット 7:0	**RWT[7:0]**：受信割り込みウォッチドッグタイマカウント このフィールドは、ウォッチドッグタイマがセットされる RWTU フィールドに示される係数によって掛けられた、システムクロックサイクル数を示します。 ウォッチドッグタイマは、対応するディスクリプタの割り込み有効ビット RDES3[30] の設定のため、Rx DMA が ETH_DMACSR で RI ビットがセットされていないパケットの転送を完了した後に、プログラムされた値でトリガされます。 ウォッチドッグタイマがタイムアウトすると、RI ビットがセットされ、タイマが停止します。受信パケットの割り込み有効ビット RDES3[30] に従う RI の自動設定のために RI ビットがハイに設定されたとき、ウォッチドッグタイマはリセットされます。

チャネル現在のアプリケーション送信ディスクリプタレジスタ（ETH_DMACCATxDR）

アドレスオフセット：0x1144

リセット値：0x0000 0000

チャネル現在のアプリケーション送信ディスクリプタレジスタは、DMA によって読み出された現在の送信ディスクリプタを示します。

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

ビット 31:0 **CURTDESAPTR[31:0]**：アプリケーション送信ディスクリプタアドレスポインタ
DMA は Tx 動作中にこのポインタを更新します。このポインタはリセット時にクリアされます。

チャネル現在のアプリケーション受信ディスクリプタレジスタ（ETH_DMACCARxDR）

アドレスオフセット：0x114C

リセット値：0x0000 0000

チャネル現在のアプリケーション受信ディスクリプタレジスタは、DMA によって読み出された現在の受信ディスクリプタを示します。

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

ビット 31:0	**CURRDESAPTR[31:0]**：アプリケーション受信ディスクリプタアドレスポインタ DMA は Rx 動作中にこのポインタを更新します。このポインタはリセット時にクリアされます。

チャネル現在のアプリケーション送信バッファレジスタ（ETH_DMACCATxBR）

アドレスオフセット：0x1154

リセット値：0x0000 0000

チャネル現在のアプリケーション送信バッファアドレスレジスタは、DMA によって読み出された現在の Tx バッファアドレスを示します。

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

ビット 31:0 **CURTBUFAPTR[31:0]**：アプリケーション送信バッファアドレスポインタ
DMA は Tx 動作中にこのポインタを更新します。このポインタはリセット時にクリアされます。

チャネル現在のアプリケーション受信バッファレジスタ（ETH_DMACCARxBR）

アドレスオフセット：0x115C

リセット値：0x0000 0000

チャネル現在のアプリケーション受信バッファアドレスレジスタは、DMA によって読み出された現在の Rx バッファアドレスを示します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CURRBUFAPTR															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **CURRBUFAPTR[31:0]**：アプリケーション受信バッファアドレスポインタ
DMA は Rx 動作中にこのポインタを更新します。このポインタはリセット時にクリアされます。

チャネルステータスレジスタ（ETH_DMACSR）

アドレスオフセット：0x1160

リセット値：0x0000 0000

ソフトウェアドライバ（アプリケーション）は、割り込みサービスルーチン中またはポーリング中に、ステータスレジスタを読み出して、DMA のステータスを決定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REB[2:0]			TEB[2:0]		
										r			r		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NIS	AIS	CDE	FBE	ERI	ETI	RWT	RPS	RBU	RI	Res.	Res.	Res.	TBU	TPS	TI
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw

ビット 31:22	予約済みであり、リセット値に保持する必要があります。
ビット 21:19	**REB[2:0]**：Rx DMA エラービット このフィールドは、バスエラーの原因となったエラーのタイプを示します。たとえば、AHB または AXI インタフェース上でのエラーレスポンスがあります。 ビット 21 1：Rx DMA によるデータ転送中のエラー 0：Rx DMA によるデータ転送中のエラーなし ビット 20 1：ディスクリプタアクセス中のエラー 0：データバッファアクセス中のエラー ビット 19 1：読み出し転送中のエラー 0：書き込み転送中のエラー このフィールドは、FBE ビットがセットされたときにのみ有効です。このフィールドは割り込みを生成しません。
ビット 18:16	**TEB[2:0]**：Tx DMA エラービット このフィールドは、バスエラーの原因となったエラーのタイプを示します。たとえば、AHB または AXI インタフェース上でのエラーレスポンスがあります。 ビット 18 1：Tx DMA によるデータ転送中のエラー 0：Tx DMA によるデータ転送中のエラーなし ビット 17 1：ディスクリプタアクセス中のエラー 0：データバッファアクセス中のエラー ビット 16 1：読み出し転送中のエラー 0：書き込み転送中のエラー このフィールドは、FBE ビットがセットされたときにのみ有効です。このフィールドは割り込みを生成しません。
ビット 15	**NIS**：通常割り込み要約 通常割り込み要約ビットの値は、ETH_DMACIER レジスタで対応する割り込みビットが有効なとき、以下のビットの論理 OR です。 ビット 0：送信割り込み ビット 2：送信バッファ使用不可 ビット 6：受信割り込み ビット 11：アーリー受信割り込み マスクされていないビット（割り込み有効化が ETH_DMACIER レジスタでセットされている割り込み）のみが、通常割り込み要約ビットに影響します。 これはスティッキービットです。NIS をセット状態にする対応ビットがクリアされるたびに、このビットを（このビットに 1 を書き込むことによって）クリアしなければなりません。

ビット 14	**AIS**：異常割り込み要約 異常割り込み要約ビットの値は、ETH_DMACIER レジスタで対応する割り込みビットが有効なとき、以下の論理 ORです。 ビット 1：送信プロセス停止 ビット 7：受信バッファ使用不可 ビット 8：受信プロセス停止 ビット 10：アーリー送信割り込み ビット 12：致命的バスエラー ビット 13：コンテキストディスクリプタエラー マスクされていないビットのみが異常割り込み要約ビットに影響します。 これはスティッキービットです。AIS をセット状態にする対応ビットがクリアされるたびに、このビットを（このビットに 1 を書き込むことによって）クリアしなければなりません。
ビット 13	**CDE**：コンテキストディスクリプタエラー このビットは、DMA Tx エンジンがパケットの中（中間ディスクリプタ）でコンテキストディスクリプタを受信し、DMA Tx エンジンはこれを無視したことを示します。
ビット 12	**FBE**：致命的バスエラー このビットは、（EB フィールドで示される）バスエラーが発生したことを示します。このビットがセットされると、対応する DMA チャネルエンジンは、すべてのバスアクセスを無効にします。
ビット 11	**ERI**：アーリー受信割り込み このビットは、DMA がパケットの最初のデータバッファを一杯にしたことを示します。このレジスタの RI ビットが、このビットを自動的にクリアします。
ビット 10	**ETI**：アーリー送信割り込み このビットは、パケットが完全に MTL Tx FIFO に転送されたことを示します。
ビット 9	**RWT**：受信ウォッチドッグタイムアウト このビットは、2,048 バイト（ジャンボパケットモードが有効の場合は 10,240 バイト）より長いパケットが受信されたときにアサートされます。
ビット 8	**RPS**：受信プロセス停止 このビットは、Rx プロセスが停止状態に入ったときにアサートされます。
ビット 7	**RBU**：受信バッファ使用不可 このビットは、アプリケーションが受信リストに次のディスクリプタを所有しており、DMA が取得できないことを示します。Rx プロセスはサスペンドされます。Rx ディスクリプタの処理を再開するには、アプリケーションは、ディスクリプタの所有権を変更して、受信ポーリング要求コマンドを発行しなければなりません。このコマンドが発行されなかった場合、Rx プロセスは、認識可能な次の着信パケットが受信されたときに再開されます。リングモードでは、アプリケーションはチャネルの受信ディスクリプタテイルポインタレジスタを上げる必要があります。このビットは、DMA が前の Rx ディスクリプタを所有している場合のみセットされます。
ビット 6	**RI**：受信割り込み このビットはパケット受信が完了したことを示します。パケット受信が完了すると、RDES1 のビット 31 が最終ディスクリプタでリセットされ、特定のパケットステータス情報がそのディスクリプタで更新されます。 受信は実行中状態のままです。
ビット 5:3	予約済みであり、リセット値に保持する必要があります。

ビット 2	**TBU**：送信バッファ使用不可 このビットは、アプリケーションが送信リストに次のディスクリプタを所有しており、DMA が取得できないことを示します。送信は中断されます。DMA_Debug_Status0 レジスタの TPS0 フィールドは、送信プロセス状態の遷移を表します。 送信ディスクリプタの処理を再開するには、アプリケーションは以下の手順を実行する必要があります。 1.TDES3 のビット 31 をセットすることによってディスクリプタの所有権を変更します。 2.送信ポーリング要求コマンドを発行します。 リングモードでは、アプリケーションはチャネルの送信ディスクリプタテイルポインタレジスタを上げる必要があります。
ビット 1	**TPS**：送信プロセス停止 このビットは、送信が停止されたときにセットされます。
ビット 0	**TI**：送信割り込み このビットはパケット送信が完了したことを示します。送信が完了すると、TDES3 のビット 31 が最終ディスクリプタでリセットされ、特定のパケットステータス情報がそのディスクリプタで更新されます。

チャネル欠落フレームカウントレジスタ（ETH_DMACMFCR）

アドレスオフセット：0x116C

リセット値：0x0000 0000

このレジスタには、バスエラーまたは ETH_DMACRxCR レジスタの RPF フィールドのプログラムにより、DMA がドロップしたパケット数のカウンタがあります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MFCO	Res.	Res.	Res.	Res.	MFC[10:0]										
r					r										

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15	**MFCO**：MFC カウンタのオーバーフローステータス このビットがセットされた場合、MFC カウンタはそれ以上インクリメントされません。このレジスタが読み出されると、ビットはクリアされます。
ビット 14:11	予約済みであり、リセット値に保持する必要があります。
ビット 10:0	**MFC[10:0]**：ドロップされたパケットのカウンタ このカウンタは、バスエラーまたは ETH_DMACRxCR レジスタの RPF フィールドのプログラムにより、DMA がドロップしたパケット数のカウンタを示します。このレジスタが読み出されると、カウンタはクリアされます。

Ethernet DMA レジスタマップとリセット値

表 536. ETH_DMA レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1000	ETH_DMAMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTM		Res.	PR[2:0]			TXPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.			DA	SWR
	リセット値															0	0		0	0	0	0							0	0	0	0	0
0x1004	ETH_DMASBMR	Res.	Res.	Res.	Res.	Res.				Res.	Res.	Res.	Res.	RD_OSR_LMT[1:0]				RB	MB	Res.	AAL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FB
	リセット値													0	0	0	1	0	0		0												0
0x1008	ETH_DMAISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MACIS	MTLIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DC0IS
	リセット値															0	0																0
0x100C	ETH_DMADSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				Res.				TPS0				RPS0				Res.	Res.	Res.	Res.	Res.	Res.	Res.	AXWHSTS
	リセット値																	0	0	0	0	0	0	0	0								0
0x1010 - 0x101C	予約済み																																

表 537. ETH_DMA_CH レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1100	ETH_DMACCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSL			Res.	PBLX8	Res.	Res.	MSS													
	リセット値												0	0	0		0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1104	ETH_DMACTxCR	Res.	Res.	Res.	Res.	Res.				Res.	Res.	TXPBL[5:0]						IPBL	Res.	Res.	TSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OSF	Res.			ST
	リセット値											0	0	0	0	0	0	0			0								0				0
0x1108	ETH_DMACRxCR	RPF	Res.	Res.	Res.	Res.				Res.	Res.	RXPBL[5:0]						Res.	RBSZ[10:0]										Res.	Res.	Res.	SR	
	リセット値	0										0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0				0
0x110C - 01110	予約済み																																
0x1114	ETH_DMACTxDLAR	TDESLA																														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x1118	予約済み																																
0x111C	ETH_DMACRxDLAR	RDESLA																														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x1120	ETH_DMACTxDTPR	TDT																														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x1124	予約済み																																

表 537. ETH_DMA_CH レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1128	ETH_ DMACRxDTPR	RDT																														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x112C	ETH_ DMACTxDRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDRL[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x1130	ETH_ DMACRxDRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDRL[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x1134	ETH_DMACIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NIE	AIE	CDEE	FBEE	ERIE	ETIE	RWTE	RSE	RBUE	RIE	Res.	Res.	Res.	TBUE	TXSE	TIE
	リセット値																	0	0	0	0	0	0	0	0	0	0				0	0	0
0x1138	ETH_ DMACRxIWTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWTU[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWT[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x1140	予約済み																																
0x1144	ETH_ DMACCATxDR	CURTDESAPTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1148	予約済み																																
0x0114C	ETH_ DMACCARxDR	CURRDESAPTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1150	予約済み																																
0x1154	ETH_ DMACCATxBR	CURTBUFAPTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1158	予約済み																																
0x115C	ETH_ DMACCARxBR	CURRBUFAPTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1160	ETH_DMACSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REB[2:0]			TEB[2:0]			NIS	AIS	CDE	FBE	ERI	ETI	RWT	RPS	RBU	RI	Res.	Res.	Res.	TBU	TPS	TI
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0
0x1164 - 0x1168	予約済み																																
0x116C	ETH_ DMACMFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MFCO	Res.	Res.	Res.	Res.	MFC[10:0]										
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

58.11.3 Ethernet MTL レジスタ

動作モードレジスタ（ETH_MTLOMR）

アドレスオフセット：0x0C00

リセット値：0x0000 0000

動作モードレジスタは、送信および受信動作モードとコマンドを確立します。

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

ビット 31:10	予約済みであり、リセット値に保持する必要があります。
ビット 9	**CNTCLR**：カウンタリセット このビットがセットされた場合、すべてのカウンタがリセットされます。このビットは、1 クロックサイクル後に自動的にクリアされます。 このビットが CNT_PRESET ビットとともにセットされた場合、CNT_PRESET が優先されます。
ビット 8	**CNTPRST**：カウンタプリセット このビットがセットされた場合： – ETH_MTLTxQUR レジスタは、0x7F0 に初期化／プリセットされます。 – ETH_MTLRxQMPOCR レジスタの欠落パケットおよびオーバーフローパケットカウンタは、0x7F0 に初期化／プリセットされます。
ビット 7	予約済みであり、リセット値に保持する必要があります。
ビット 6:2	予約済みであり、リセット値に保持する必要があります。
ビット 1	**DTXSTS**：ドロップ送信ステータス このビットがセットされた場合、MAC から受信した Tx パケットステータスが MTL でドロップされます。このビットがリセットされた場合、MAC から受信した Tx パケットステータスがアプリケーションに転送されます。 このビットは、コアを設定している間 MTL での送信ステータスの無効化機能が選択されているとき、予約済みかつ読み出し専用となります。
ビット 0	予約済みであり、リセット値に保持する必要があります。

割り込みステータスレジスタ（ETH_MTLISR）

アドレスオフセット：0x0C20

リセット値：0x0000 0000

ソフトウェアドライバ（アプリケーション）は、割り込みサービスルーチン中またはポーリング中に、このレジスタを読み出して、MTL キュー、MAC の割り込みステータスを決定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Q0IS
															r

ビット 31:1	予約済みであり、リセット値に保持する必要があります。
ビット 0	**Q0IS**：キュー 割り込みステータス このビットは、割り込みがキューによって生成されたことを示します。このビットをリセットするために、ETH_MTLQICSR レジスタを読み出して、割り込みの原因を特定し、そのソースをクリアします。

Tx キュー動作モードレジスタ（ETH_MTLTxQOMR）

アドレスオフセット：0x0D00

リセット値：0x0007 0008

キュー送信動作モードレジスタは、送信キュー動作モードとコマンドを確立します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TQS								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TTC[2:0]			TXQEN[1:0]		TSF	FTQ
									rw			r		rw	rw

ビット 31:25	予約済みであり、リセット値に保持する必要があります。
ビット 24:16	**TQS**：送信キューサイズ このフィールドは、割り当てられた送信キューのサイズを 256 バイトのブロック単位で示します。16 番目のビットがこのフィールドの開始ビットです。このフィールドの幅は設定で選択された Tx メモリサイズに依存します。たとえば、メモリサイズが 2048 の場合、このフィールドの幅は 3 ビットです。 LOG2(2048/256) = LOG2(8) = 3 ビット TQS = 0 は、256 バイトに対応します。
ビット 15:7	予約済みであり、リセット値に保持する必要があります。
ビット 6:4	**TTC[2:0]**：送信閾値制御 これらのビットは、MTL Tx キューの閾値レベルを制御します。送信は、MTL Tx キュー内のパケットサイズが閾値より大きいときに開始されます。さらに、閾値より短い完全なパケットも送信されます。これらのビットは、TSF ビットがリセットされたときのみ使用されます。 000：32 001：64 010：96 011：128 100：192 101：256 110：384 111：512
ビット 3:2	**TXQEN[1:0]**：送信キュー有効化 このフィールドは、送信キューの有効化／無効化に使用されます。 00：無効 01：予約済み 10：有効 11：予約済み ***注：　複数の Tx キュー設定では、すべてのキューがデフォルトで無効化されています。このフィールドをプログラムすることで Tx キューを有効にします。***
ビット 1	**TSF**：送信ストアアンドフォワード このビットがセットされると、完全なパケットが MTL Tx キューに存在しているときに送信が開始されます。このビットがセットされると、このレジスタのビット [6:4] で指定された TTC の値は無視されます。このビットは、送信が停止されたときにのみ変更されなければなりません。
ビット 0	**FTQ**：送信キューの一掃 このビットがセットされた場合、Tx キューコントローラロジックはデフォルト値にリセットされます。そのため、Tx キューのすべてのデータが失われるか一掃されます。このビットは、一掃動作が完了したときに内部的にリセットされます。このビットがリセットされるまで、ETH_MTLTxQ1OMR レジスタに書き込まないでください。MAC トランスミッタによってすでに許可されたデータは一掃されません。送信にスケジュールされ、アンダーフローおよびラントパケット送信となります。 ***注：　一掃動作は、Tx キューがエンプティで、アプリケーションがすべての送信パケットのペンディング中の Tx ステータスを許可したときのみ完了します。この一掃動作を完了するには、PHY Tx クロック（eth_mii_tx_clk）がアクティブでなければなりません。***

Tx キューアンダーフローレジスタ（ETH_MTLTxQUR）

アドレスオフセット：0x0D04

リセット値：0x0000 0000

キューアンダーフローカウンタレジスタには、送信キューアンダーフローのためにアボートされたパケットおよび受信キューパケット一掃のために欠落したパケットのカウンタがあります。

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

ビット 31:12	予約済みであり、リセット値に保持する必要があります。
ビット 11	**UFCNTOVF**：アンダーフローパケットカウンタのオーバーフロービット このビットは、Tx キューアンダーフローパケットカウンタフィールドがオーバーフローし、最大数を超えるたびにセットされます。このようなシナリオでは、オーバーフローパケットカウンタがすべて 0 にリセットされ、このビットはロールオーバーが発生したことを示します。
ビット 10:0	**UFFRMCNT[10:0]**：アンダーフローパケットカウンタ このフィールドは、Tx キューアンダーフローのためにコントローラによってアボートされたパケットの数を示します。このカウンタは、アンダーフローのために MAC が進行中のパケットをアボートするたびにインクリメントされます。このレジスタが 1'b1 の mci_be_i[0] で読み出されると、カウンタはクリアされます。

Tx キューデバッグレジスタ（ETH_MTLTxQDR）

アドレスオフセット：0x0D08

リセット値：0x0000 0000

キュー送信デバッグレジスタは、送信キューに関連するさまざまなブロックのデバッグステータスを提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STXSTSF[2:0]			Res.	PTXQ[2:0]		
									r				r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXSTS FSTS	TXQ STS	TWC STS	TRCSTS[1:0]		TXQPA USED
										r	r	r	r		r

ビット 31:23	予約済みであり、リセット値に保持する必要があります。
ビット 22:20	**STXSTSF[2:0]**：キューの Tx ステータス FIFO のステータスワード数 このフィールドは、このキューの Tx ステータス FIFO の現在のステータス数を示します。このフィールドは、コアを設定している間 MTL での送信ステータスの無効化機能が選択されているとき、予約済みとなります。 ETH_MTLOMR レジスタの DTXSTS ビットが 1 にセットされている場合、このフィールドは Tx ステータス FIFO のステータスワード数を反映しません。
ビット 19	予約済みであり、リセット値に保持する必要があります。
ビット 18:16	**PTXQ[2:0]**：送信キューのパケット数 このフィールドは、Tx キューの現在のパケット数を示します。このフィールドは、コアを設定している間 MTL での送信ステータスの無効化機能が選択されているとき、予約済みとなります。 ETH_MTLOMR レジスタの DTXSTS ビットが 1 にセットされている場合、このフィールドは送信キューのパケット数を反映しません。
ビット 15:6	予約済みであり、リセット値に保持する必要があります。
ビット 5	**TXSTSFSTS**：MTL Tx ステータス FIFO フルステータス ハイになっている場合、このビットは MTL Tx ステータス FIFO がフルになっていることを示します。そのため、MTL は送信のための追加のパケットを許可できません。 このフィールドは、コアを設定している間 MTL での送信ステータスの無効化機能が選択されているとき、予約済みとなります。
ビット 4	**TXQSTS**：MTL Tx キューノットエンプティステータス このビットがハイの場合、MTL Tx キューがエンプティではなく、送信するデータが残っていることを示します。
ビット 3	**TWCSTS**：MTL Tx キュー書き込みコントローラステータス ハイの場合、このビットは MTL Tx キュー書き込みコントローラがアクティブで、Tx キューにデータを転送していることを示します。
ビット 2:1	**TRCSTS[1:0]**：MTL Tx キュー読み出しコントローラステータス このフィールドは、Tx キュー読み出しコントローラの状態を示します。 00：アイドル状態 01：読み出し状態（データを MAC トランスミッタへ転送中） 10：MAC トランスミッタからのペンディング中の Tx ステータス待ち 11：MAC からのパケットアボートリクエストによる Tx キューの一掃
ビット 0	**TXQPAUSED**：ポーズ中の送信キュー このビットがハイで、Rx フロー制御が有効である場合、Tx キューは次の理由により、（全二重専用モードで）ポーズ状態となります。 – PFC 有効時に Tx キューに割り当てられた優先順位のための PFC パケットの受信 – PFC 有効時の 802.3x ポーズパケットの受信

キュー割り込み制御ステータスレジスタ（ETH_MTLQICSR）

アドレスオフセット：0x0D2C

リセット値：0x0000 0000

このレジスタは、キュー割り込みの割り込み有効化およびステータスビットを格納します。

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

ビット 31:25	予約済みであり、リセット値に保持する必要があります。
ビット 24	**RXOIE**：受信キューオーバーフロー割り込み有効化 このビットがセットされた場合、受信キューオーバーフロー割り込みが有効になります。このビットがリセットされた場合、受信キューオーバーフロー割り込みが無効になります。
ビット 23:17	予約済み
ビット 16	**RXOVFIS**：受信キューオーバーフロー割り込みステータス このビットは、パケット受信中に受信キューにオーバーフローがあったことを示します。部分的なパケットがアプリケーションに転送された場合、RDES3[21] でオーバーフローステータスがセットされます。このビットは、アプリケーションがこのビットに 1 を書き込むとクリアされます。
ビット 15:9	予約済みであり、リセット値に保持する必要があります。
ビット 8	**TXUIE**：送信キューアンダーフロー割り込み有効化 このビットがセットされた場合、送信キューアンダーフロー割り込みが有効になります。このビットがリセットされた場合、送信キューアンダーフロー割り込みが無効になります。
ビット 7:1	予約済みであり、リセット値に保持する必要があります。
ビット 0	**TXUNFIS**：送信キューアンダーフロー割り込みステータス このビットは、パケット送信中に送信キューにアンダーフローがあったことを示します。送信は中断され、アンダーフローエラー TDES3[2] がセットされます。このビットは、アプリケーションがこのビットに 1 を書き込むとクリアされます。

Rx キュー動作モードレジスタ（ETH_MTLRxQOMR）

アドレスオフセット：0x0D30

リセット値：0x0070 0000

キュー受信動作モードレジスタは、受信キュー動作モードとコマンドを確立します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RQS			Res.	Res.	Res.	RFD2
									r	r	r				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RFD1	RF0	Res.	Res.	Res.	RFA			EHFC	DIS_TCP_EF	RSF	FEP	FUP	Res.	RTC[1:0]	
rw	rw				rw			rw	rw	rw	rw	rw		rw	

ビット 31:24	予約済みであり、リセット値に保持する必要があります。
ビット 23:20	**RQS[3:0]**：受信キューサイズ このフィールドは、割り当てられた受信キューのサイズを 256 バイトのブロック単位で示します。Rx キューの数が複数で、リセット値が 0x0 の場合のみ、RQS フィールドは読み書きできます。
ビット 19:17	予約済みであり、リセット値に保持する必要があります。
ビット 16:14	**RFD[2:0]**：非アクティブフロー制御の閾値（半二重モードおよび全二重モード） これらのビットは、有効化した後にフロー制御がネゲートされる閾値（Rx キューのフィルレベル）を制御します。 0：フルマイナス 1 KB つまりフル 1 KB 1：フルマイナス 1.5 KB つまりフル 1.5 KB 2：フルマイナス 2 KB つまりフル 2 KB 3：フルマイナス 2.5 KB つまりフル 2.5 KB 4：フルマイナス 2.5 KB つまりフル 3 KB 5：フルマイナス 2.5 KB つまりフル 3.5 KB フロー制御がアサートされた後でのみ、ネゲートが有効となります。 ***注：　値は、閾値が正の数であることを確認するこのような方法でプログラムされる必要があります。EHFC がハイにセットされているとき、これらの値はこのレジスタの RQS フィールドで決められた Rx キューサイズが 4 KB 以上である場合のみ適用できます。指定されたキューサイズでは、値の範囲が 0 からフルマイナス（QSIZE - 0.5 KB）のエンコーディングとなり、それ以外すべての値は不正となります。ここで、フルと QSIZE の用語はこのレジスタの RQS フィールドで決められたキューサイズを参照します。***
ビット 13:11	予約済みであり、リセット値に保持する必要があります。
ビット 10:8	**RFA[2:0]**：アクティブフロー制御の閾値（半二重モードおよび全二重） これらのビットは、フロー制御が有効化される閾値（Rx キューのフィルレベル）を制御します。 このフィールドのエンコーディングの詳細については、RFD を参照してください。

ビット 7	**EHFC**：ハードウェアフロー制御の有効化 このビットがセットされた場合、フロー制御信号動作は、Rx キューのフィルレベルに基づいて有効になります。リセット時、フロー制御動作は無効になります。このビットは、Rx キューが 4 KB 未満のとき、使用されません（予約済みで常にリセットされます）。
ビット 6	**DIS_TCP_EF**：TCP/IP チェックサムエラーパケットのドロッピングディスエーブル このビットがセットされた場合、MAC は受信チェックサムオフロードエンジンによって検出されたエラーだけがあるパケットをドロップしません。このようなパケットでは、カプセル化されたペイロードでのみエラーがあります。MAC から受信した Ethernet パケットにエラー（FCS エラーを含む）はありません。 このビットがリセットされると、FEF ビットがリセットされていた場合、すべてのエラーパケットがドロップされます。
ビット 5	**RSF**：受信キューストアアンドフォワード このビットがセットされた場合、Ethernet ペリフェラルは、完全なパケットが書き込まれた後のみ、Rx キューからパケットを読み出し、このレジスタの RTC フィールドは無視されます。このビットがリセットされると、Rx キューは、このレジスタの RTC フィールドで指定された閾値に従って、閾値（カットスルー）モードで動作します。
ビット 4	**FEP**：エラーパケット転送 このビットがリセットされると、Rx キューは、エラーステータス（CRC エラー、受信エラー、ウォッチドッグタイムアウト、オーバーフロー）のパケットをドロップします。ただし、パケットの開始バイト（書き込み）ポインタが読み出しコントローラ側にすでに転送されている場合（閾値モードで）、パケットはドロップされません。 このビットがセットされた場合、ラントエラーパケットを除くすべてのパケットがアプリケーションまたは DMA に転送されます。RSF ビットがセットされ、部分的なパケットが書き込まれたときに Rx キューがオーバーフローした場合、このビットの設定にかかわらずパケットがドロップされます。ただし、RSF ビットがリセットされ、部分的なパケットが書き込まれたときに Rx キューがオーバーフローした場合、部分的なパケットはアプリケーションまたは DMA に転送できます。
ビット 3	**FUP**：アンダーサイズの良好パケットの転送 このビットがセットされた場合、Rx キューは、パッドバイトと CRC を含めて、アンダーサイズの良好パケット（エラーがなく、長さが 64 バイト未満のパケット）を転送します。このビットがリセットされると、RTC = ‘01’ など Rx 閾値の低い値であるために、パケットがすでに転送されていない限り、Rx キューは 64 バイト未満のすべてのパケットをドロップします。
ビット 2	予約済みであり、リセット値に保持する必要があります。
ビット 1:0	**RTC[1:0]**：受信キュー閾値制御 これらのビットは、MTL Rx キューの閾値レベルを（バイト単位で）制御します。 00：64 01：32 10：96 11：128 受信パケットは、MTL Rx キュー内のパケットサイズが閾値より大きいときに、アプリケーションまたは DMA に転送されます。さらに、閾値より短い完全なパケットは自動的に転送されます。 このフィールドは、RSF ビットが 0 である場合のみ有効です。このフィールドは、RSF ビットが ‘1’ にセットされた場合は無視されます。

Rx キュー欠落パケットおよびオーバーフローカウンタレジスタ（ETH_MTLRxQMPOCR）

アドレスオフセット：0x=0D34

リセット値：0x0000 0000

キュー欠落パケットおよびオーバーフローカウンタレジスタには、受信キューパケット一掃のために欠落したパケットおよび受信キューオーバーフローのために破棄されたパケットのカウンタがあります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MISCN TOVF	MISPKTCNT[10:0]										
				r	r										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	OVFCN TOVF	OVFPKTCNT[10:0]										
				r	r										

ビット 31:28	予約済みであり、リセット値に保持する必要があります。
ビット 27	**MISCNTOVF**：欠落パケットカウンタオーバーフロービット セットされた場合、このビットは Rx キュー欠落パケットカウンタが最大制限を超えたことを示します。
ビット 26:16	**MISPKTCNT[10:0]**：欠落パケットカウンタ このフィールドは、アプリケーションがこのキューに対して ari_pkt_flush_i[] をアサートしたため、Ethernet ペリフェラルによって見逃されたパケットの数を示します。このレジスタが 1b1 の mci_be_i[0] で読み出されると、カウンタはリセットされます。 このカウンタは、バッファ使用不可のために DMA がパケットを破棄すると 1 つインクリメントされます。
ビット 15:12	予約済みであり、リセット値に保持する必要があります。
ビット 11	**OVFCNTOVF**：オーバーフローカウンタオーバーフロービット セットされた場合、このビットは Rx キューオーバーフローパケットカウンタが最大制限を超えたことを示します。
ビット 10:0	**OVFPKTCNT[10:0]**：オーバーフローパケットカウンタ このフィールドは、受信キューオーバーフローのために Ethernet ペリフェラルによって破棄されたパケットの数を示します。このカウンタは、Ethernet ペリフェラルがオーバーフローのために着信パケットを破棄するたびにインクリメントされます。このレジスタが ‘1’ の mci_be_i[0] で読み出されると、カウンタはリセットされます。

Rx キューデバッグレジスタ（ETH_MTLRxQDR）

アドレスオフセット：0x0D38

リセット値：0x0000 0000

キュー受信デバッグレジスタは、受信キューに関連するさまざまなブロックのデバッグステータスを提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PRXQ[13:0]													
		r													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXQSTS[1:0]		Res.	RRCSTS[1:0]		RWCSTS
										r			r		r

ビット 31:30	予約済みであり、リセット値に保持する必要があります。
ビット 29:16	**PRXQ[13:0]**：受信キューのパケット数 このフィールドは、Rx キューの現在のパケット数を示します。このフィールドの理論上の最大値は 256KB/16B = 16K パケットで、Max_Queue_Size/Min_Packet_Size です。
ビット 15:6	予約済みであり、リセット値に保持する必要があります。
ビット 5:4	**RXQSTS[1:0]**：MTL Rx キューフィルレベルステータス このフィールドは、Rx キューのフィルレベルのステータスを提供します。 00：Rx キューは空です。 01：Rx キューフィルレベルがフロー制御停止閾値より下です。 10：Rx キューフィルレベルがフロー制御開始閾値より上です。 11：Rx キューはフルです。
ビット 3	予約済みであり、リセット値に保持する必要があります。
ビット 2:1	**RRCSTS[1:0]**：MTL Rx キュー読み出しコントローラ状態 このフィールドは、Rx キュー読み出しコントローラの状態を提供します。 00：アイドル状態 01：パケットデータの読み出し中 10：パケットステータス（またはタイムスタンプ）読み出し中 11：パケットデータとステータスの一掃中
ビット 0	**RWCSTS**：MTL Rx キュー書き込みコントローラアクティブステータス ハイの場合、このビットは MTL Rx キュー書き込みコントローラがアクティブで、Rx キューに受信パケットを転送していることを示します。

Ethernet MTL レジスタマップとリセット値

表 538. ETH_MTL レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C00	ETH_MTLOMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTCLR	CNTPRST	Res.	Res.		Res.	Res.	Res.	DTXSTS	Res.
	リセット値																							0	0							0	
0x0C04 - 0x0C1C	予約済み																																
0x0C20	ETH_MTLISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Q0TXOIE
	リセット値																																0
0x0C24 - 0x0CFC	予約済み																																
0x0D00	ETH_MTLTxQOMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TQS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TTC [2:0]			TXQEN[1:0]		TSF	FTQ
	リセット値													0	1	1	1										0	0	0	1	0	0	0
0x0D04	ETH_MTLTxQUR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UFCNTOVF	UFFRMCNT[10:0]										
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x0D08	ETH_MTLTxQDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STXSTSF[2:0]			Res.	PTXQ			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXSTSFSTS	TXQSTS	TWCSTS	TRCSTS[1:0]		TXQPAUSED
	リセット値										0	0	0		0	0	0											0	0	0	0	0	0
0x0D0C - 0x0D14	予約済み																																
0x0D18 - 0x0D28	予約済み																																
0x0D2C	ETH_MTLQICSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXOIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXOVFIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXUIE	Res.	Res.	Res.	Res.	Res.	Res.	TXUNFIS
	リセット値								0								0							0	0							0	0
0x0D30	ETH_MTLRxQOMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RQS			Res.	Res.	Res.	RFD				Res.	Res.	Res.	RFA			EHFC	DIS_TCP_EF	RSF	FEP	FUP	Res.	RTC	
	リセット値										1	1	1				0	0	0					0	0	0	0	0	0	0	0		0	0
0x0D34	ETH_MTLRxQMPOCR	Res.	Res.	Res.	Res.	MISCNTOVF	MISPKTCNT[10:0]											Res.	Res.	Res.	Res.	OVFCNTOVF	OVFPKTCNT[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0

表 538. ETH_MTL レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0D38	ETH_ MTLRxQDR	Res.	Res.	PRXQ[13:0]														Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXQSTS[1:0]		Res.	RRCSTS[1:0]		RWCSTS
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0											0	0		0	0	0
0xD3C から 0xD7C	予約済み																																
0x0D68	予約済み																																

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

58.11.4 Ethernet MAC および MMC レジスタ

動作モード設定レジスタ（ETH_MACCR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

MAC 設定レジスタは、MAC の動作モードを確立します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARPEN	SARC[2:0]			IPC	IPG[2:0]			GPSLCE	S2KP	CST	ACS	WD	Res.	JD	JE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FES	DM	LM	ECRSFD	DO	DCRS	DR	Res.	BL[1:0]		DC	PRELEN[1:0]		TE	RE
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31 **ARPEN**：ARP オフロード有効化

このビットがセットされた場合、MAC は着信 ARP リクエストパケットを認識でき、送信用の ARP パケットをスケジュールします。ARP パケットをアプリケーションに転送し、Rx ステータスでイベントも示します。

このビットがリセットされた場合、MAC は ARP パケットを認識できず、Rx ステータスでタイプフレームとして示します。

このビットは、IPv4 ARP オフロードの有効化が選択されている場合のみ使用できます。

ビット 30:28 **SARC[2:0]**：ソースアドレスの挿入または書き換え制御

このフィールドは、すべての送信パケットのソースアドレスの挿入または書き換えを制御します。ビット 30 は、ビット [29:28] の値に基づいたソースアドレスの挿入または書き換えに使用される MAC アドレスレジスタ（0 または 1）を指定します。

0x：mti_sa_ctrl_i および ati_sa_ctrl_i 入力信号が SA フィールドの生成を制御します。

10：

- ビット 30 が 0 にセットされた場合、MAC はすべての送信パケットの SA フィールドに、MAC アドレス 0 レジスタ（MAC レジスタ 192 および 193）の内容を挿入します。
- ビット 30 が 1 にセットされ、コアを設定している間 MAC アドレスレジスタ 1 の有効化オプションが選択されている場合、MAC はすべての送信パケットの SA フィールドに、MAC アドレス 1 レジスタ（MAC レジスタ 194 および 195）の内容を挿入します。

11：

- ビット 30 が 0 にセットされた場合、MAC はすべての送信パケットの SA フィールドの MAC アドレス 0 レジスタ（MAC レジスタ 192 および 193）の内容を書き換えます。
- ビット 30 が 1 にセットされ、MAC アドレスレジスタ 1 が有効化されている場合、MAC はすべての送信パケットの SA フィールドの MAC アドレス 1 レジスタ（MAC レジスタ 194 および 195）の内容を書き換えます。

注： ***このフィールドへの変更は、パケットの開始時にのみ有効になります。パケットが送信されているときに、このレジスタフィールドに書き込む場合、後続のパケットのみが更新された値を使用でき、現在のパケットは更新された値を使用しません。***

これらのビットは、コアを設定している間 Tx での SA および VLAN 挿入の有効化機能が選択されていないとき、予約済みかつ読み出し専用となります。

ビット 27 **IPC**：チェックサムオフロード

セットされた場合、このビットは、IPv4 ヘッダチェックサムのチェックおよび IPv4 または IPv6 の TCP、UDP、ICMP ペイロードチェックサムのチェックを有効にします。このビットがリセットされた場合、レシーバの COE 機能は無効になります。

IP チェックサムオフロード機能が、コアを設定している間、有効化されない場合、このビットは（デフォルト値で）予約済みかつ読み出し専用となります。

レイヤ 3 およびレイヤ 4 パケットフィルタおよびスプリットヘッダの有効化機能は、受信側で IPC フルチェックサムオフロードエンジンを自動的に選択します。これらの機能のいずれかが有効になっている場合、IPC ビットをセットしなければなりません。

ビット 26:24 **IPG[2:0]**：パケット間隔

これらのビットは、送信時のパケット間の最小 IPG を制御します。

000：96 ビット時間

001：88 ビット時間

010：80 ビット時間

...

111：40 ビット時間

最小 IPG の範囲は、全二重モードで有効です。

半二重モードでは、最小 IPG は、64 ビット時間（IPG = 100）についてのみ設定できます。それより小さな値は考慮されません。

JAM パターンがバックプレッシャ有効化のために送信されている場合、MAC は最小 IPG を考慮しません。

上記の機能（96 ビット時間未満の IPG）は、ETH_MACECR レジスタの EIPGEN ビットがリセットされているときのみ有効となります。EIPGEN がセットされた場合、最小 IPG（96 ビット時間を超える）は、ETH_MACECR レジスタの EIPG フィールドに示される説明に従って制御されます。

ビット 23 **GPSLCE**：ジャイアントパケットのサイズ制限制御の有効化

このビットがセットされた場合、MAC は ETH_MACECR レジスタの GPSL フィールドの値を考慮して、受信パケットをジャイアントパケットとして宣言します。このフィールドは、1,518 バイト以上にプログラムする必要があります。そうしないと、MAC は 1,518 バイトをジャイアントパケットの制限とみなします。

このビットがリセットされると、MAC は受信パケットのサイズが 1,518 バイト（タグ付きパケットの場合は 1,522 バイト）を超えた場合に、ジャイアントパケットとみなします。

ウォッチドッグタイムアウト制限、ジャンボパケットの有効化および 2K パケットの有効化は、このビットより高い優先順位であり、ジャンボパケットが有効化された状態では受信パケットのサイズが 9,018 バイト（タグ付きパケットの場合は 9,022 バイト）を超えた場合、および 2K パケットが有効化された状態では 2,000 バイトを超えた場合に、MAC はジャイアントパケットとみなします。有効化された場合、ウォッチドッグ制限に達すると、ウォッチドッグタイムアウトは受信パケットを終了します。そのため、プログラムされたジャイアントパケットの制限をウォッチドッグ制限未満にして、ジャイアントパケットステータスを取得する必要があります。

ビット 22 **S2KP**：2K パケットの IEEE 802.3as サポート

このビットがセットされた場合、MAC は 2,000 バイト長までのすべてのパケットを通常パケットとみなします。JE ビットがセットされていない場合、MAC は 2,000 バイトを超えるサイズのすべての受信パケットをジャイアントパケットとみなします。

このビットがリセットされ、JE ビットがセットされていない場合、MAC は 1,518 バイト（タグ付きの場合は 1,522 バイト）を超えるサイズのすべての受信パケットをジャイアントパケットとみなします。このビットと JE ビットの設定のジャイアントパケットステータスへの影響の詳細については、S2KP および JE ビットに基づいたジャイアントパケットステータスを参照してください。

注：　JE ビットがセットされた場合、このビットの設定によるジャイアントパケットステータスへの影響はありません。

ビット 21 **CST**：タイプパケット用 CRC ストリッピング

このビットがセットされた場合、パケットをアプリケーションに転送する前に Ether タイプ（1,536 より大きいタイプフィールド）の全パケットの最後の 4 バイト（FCS）が排除され、ドロップされます。この機能は、IP チェックサムエンジン（タイプ 1）が MAC レジスタで有効になっているとき有効ではありません。この機能は、タイプ 2 チェックサムオフロードエンジンが有効化されている場合に有効です。

注：　ACS ビットとこのビットの設定によるパケット長への影響の詳細については、CST および ACS ビットに基づいたパケット長を参照してください。

ビット 20 **ACS**：自動パッドまたは CRC ストリッピング

このビットがセットされた場合、MAC は、長さフィールドの値が 1,536 バイト未満の場合のみ、着信パケットのパッドおよび FCS フィールドを削除します。長さフィールドが 1,536 バイト以上の受信パケットは、パッドまたは FCS フィールドを削除せずに、そのままアプリケーションに渡されます。

このビットがリセットされると、MAC は、着信するすべてのパケットをそのままアプリケーションに渡します。

注： CST ビットとこのビットの設定によるパケット長への影響の詳細については、CST および ACS ビットに基づいたパケット長の表を参照してください。

ビット 19 **WD**：ウォッチドッグディスエーブル

このビットがセットされた場合、MAC は、レシーバのウォッチドッグタイマを無効にします。MAC は、最大 16,383 バイトのパケットを受信できます。

このビットがリセットされると、MAC は 2,048 バイト（JE がハイにセットされている場合は 10,240）を超えるパケットは受信できません。MAC は、2,048 バイト以降に受信したバイトを切り捨てます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **JD**：ジャバーディスエーブル

このビットがセットされた場合、MAC はトランスミッタのジャバータイマを無効にします。MAC は、最大 16,383 バイトのパケットを転送できます。

このビットがリセットされると、アプリケーションが送信中に 2,048 バイト（JE がハイにセットされている場合は 10,240）を超えるデータを送信する場合、MAC はそのパケット内の残りのバイトを送信しません。

ビット 16 **JE**：ジャンボパケット有効化

このビットがセットされた場合、MAC は Rx パケットステータスでジャイアントパケットエラーを報告せずに 9,018 バイト（VLAN タグ付きパケットの場合は 9,022 バイト）のジャンボパケットを許可します。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 14 **FES**：MAC 速度

このビットは、10/100 Mbps モードの速度を選択します。

0：10 Mbps

1：100 Mbps

1000 Mbps 専用設定では、このビットはリセット値で読み出し専用となります。10 または 100 Mbps 専用またはデフォルトの 10/100 Mbps の設定では、このビットは読み書きできます。mac_speed_o[0] 信号は、このビットの値を反映します。

ビット 13 **DM**：全二重モード

このビットがセットされた場合、MAC は全二重モードで動作し、送信と受信を同時に行うことができます。このビットは、全二重専用設定では 1'b1 のデフォルト値で読み出し専用となります。

ビット 12 **LM**：ループバックモード

このビットがセットされた場合、MAC は、MII でループバックモードで動作します。ループバックが正しく機能するためには、MII Rx クロック入力（eth_mii_rx_clk）が必要です。これは Tx クロックが内部的にループバックされないためです。

ビット 11 **ECRSFD**：全二重での送信前のキャリアセンスの有効化

このビットがセットされた場合、MAC トランスミッタは、全二重モードでのパケット送信の前に CRS 信号をチェックします。CRS 信号がローレベルのときのみ、MAC が送信を開始します。

このビットがリセットされた場合、MAC トランスミッタは CRS 信号のステータスを無視します。

ビット 10 **DO**：受信オウンの無効化

このビットがセットされた場合、半二重モードで ETH_TX_EN がアサートされたときに、MAC はパケットの受信を無効にします。このビットがリセットされた場合、MAC は PHY によって与えられるすべてのパケットを受信します。

このビットは、全二重モードでは適用できません。このビットは、全二重専用の設定ではデフォルト値で予約済みかつ読み出し専用（RO）となります。

ビット 9 **DCRS**：送信中のキャリアセンスの無効化

このビットがセットされた場合、MAC トランスミッタは、半二重モードでのパケット送信中に (G)MII CRS 信号を無視します。その結果、送信中のキャリアの喪失やキャリアなしによるエラーは生成されません。

このビットがリセットされた場合、MAC トランスミッタはキャリアセンスのためにエラーを生成します。MAC は送信もアボートする可能性があります。

このビットは、全二重専用の設定では予約済みかつ読み出し専用（RO）となります。

ビット 8 **DR**：再試行の無効化

このビットがセットされると、MAC は送信を 1 回だけ試みます。MII インタフェースで衝突が発生すると、MAC は現在のパケット送信を無視して、Tx パケットステータスで過剰衝突エラーによるパケットアボートを報告します。

このビットがリセットされた場合、MAC は、BL フィールドの設定に基づいて再試行を行います。このビットは、半二重モードでのみ適用されます。このビットは、全二重専用の設定では予約済みかつ読み出し専用（RO）となります。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:5 **BL[1:0]**：バックオフリミット

バックオフリミットは、衝突の後、再試行時に送信の試みを再スケジュールする前に、MAC が待機するスロット時間遅延（1000 Mbps の場合は 4,096 ビット時間、10/100 Mbps の場合は 512 ビット時間）のランダム整数値（r）を指定します。

00：k = min (n, 10)

01：k = min (n, 8)

10：k = min (n, 4)

11：k = min (n, 1)

ここで、- = 再送信試行回数

ランダム整数値 r は、0 <= r < 2^k の範囲の値を取ります。

このビットは、半二重モードでのみ適用されます。このビットは、全二重専用の設定では予約済みかつ読み出し専用（RO）となります。

ビット 4 **DC**：遅延チェック

このビットがセットされた場合、MAC の遅延チェック機能が有効になります。MAC は、Tx ステートマシンが 10 または 100 Mbps モードで 24,288 ビット時間以上遅延した場合、パケットアボートステータスを発行して、Tx パケットステータスの過剰遅延エラービットをセットします。

MAC が 1000 Mbps 動作に設定されている場合、遅延の閾値は 155,680 ビット時間です。遅延は、トランスミッタの送信準備ができたときに開始しますが、MII 上に有効なキャリアセンス信号（CRS）があるときには開始されません。

遅延時間は累積されません。たとえば、アクティブである CRS 信号が非アクティブになったために、トランスミッタが 10,000 ビット時間遅延した場合、トランスミッタが送信して衝突が発生します。衝突のため、トランスミッタはバックオフしてから、完了後に再度遅延する必要があります。このようなシナリオでは、遅延タイマが 0 にリセットされ、再開されます。

このビットがリセットされた場合、遅延チェック機能は無効であり、MAC は CRS 信号が無効になるまで遅延します。

このビットは、半二重モードでのみ適用されます。このビットは、全二重専用の設定では予約済みかつ読み出し専用（RO）となります。

ビット 3:2 **PRELEN[1:0]**：送信パケット用プリアンブル長

これらのビットは、各 Tx パケットの冒頭に追加されるプリアンブルバイトの数を制御します。MAC が全二重モードで動作している場合のみ、プリアンブルの削減が発生します。

00：7 バイトのプリアンブル

01：5 バイトのプリアンブル

10：3 バイトのプリアンブル

11：予約済み

ビット 1 **TE**：トランスミッタ有効化

このビットがセットされた場合、MAC の Tx ステートマシンは、MII インタフェース上で送信を行うために有効になります。このビットがリセットされると、MAC Tx ステートマシンは現在のパケットの送信完了後に無効になります。Tx ステートマシンはそれ以上パケットを送信しません。

ビット 0 **RE**：レシーバ有効化

このビットがセットされた場合、MAC の Rx ステートマシンは、MII インタフェースからパケットを受信するために有効になります。このビットがリセットされると、MAC Rx ステートマシンは現在のパケットの受信完了後に無効になります。Rx ステートマシンは、MII インタフェースからそれ以上パケットを受信しません。

表 539 に、ETH_MACCR レジスタの S2KP および JE ビットの設定によるジャイアントパケットステータスへの影響を示します。

表 539. S2KP および JE ビットに基づいたジャイアントパケットステータス

長さ／タイプフィールド	受信パケット長	S2KP	JE	ジャイアントパケットステータス
タグなしパケット	> 1,518	0	0	1
	> 2,000	1	0	1
	> 9,018	x	1	1
VLAN タグ付きパケット	> 1,522	0	0	1
	> 2,000	1	0	1
	> 9,022	x	1	1
注： 他のすべての組み合わせでは、ジャイアントパケットステータスは 0 です。				

表 540 に、ETH_MACCR レジスタの CST および ACS ビットの設定による CRC の長さがパケット長に含まれるかどうかへの影響を示します。

表 540. CST および ACS ビットに基づいたパケット長

受信チェックサムオフロードエンジン	受信パケット長	CST	ACS	FCS 削除終了
IPCHKSUM_EN = 0 および IPC_FULL_OFFLOAD = 0 または IPCHKSUM_EN = 1 および IPC_FULL_OFFLOAD = 1	< 1,536	x	0	いいえ
		x	1	はい（Ethernet パケット）
	>= 1,536	0	x	いいえ
		1	x	はい（タイプパケット）

表 540. CST および ACS ビットに基づいたパケット長

IPCHKSUM_EN = 1 および IPC_FULL_OFFLOAD = 0	< 1,536	x	0	いいえ
		x	1	はい（Ethernet パケット）
	>= 1,536	x	x	いいえ

拡張動作モード設定レジスタ（ETH_MACECR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

MAC 拡張設定レジスタは、MAC の動作モードを確立します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EIPG[4:0]					EIPGEN	Res.	RESERVED_HDSMS[2:0]			Res.	USP	SPEN	DCRCC
		rw					rw		r				rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	GPSL[13:0]													
		rw													

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:25 **EIPG[4:0]**：拡張パケット間隔

このフィールドの値は、EIPGEN ビットがセットされた場合に適用できます。このフィールド（最上位ビットとして）は、ETH_MACCR の IPG フィールドとともに、8 ビット時間単位で 96 ビット時間を超える最小 IPG を提供します。

{EIPG, IPG}

0x00：104 ビット時間

0x01：112 ビット時間

0x02：120 ビット時間

..

0xFF：2144 ビット時間

ビット 24 **EIPGEN**：拡張パケット間隔の有効化

このビットがセットされた場合、MAC は ETH_MACCR レジスタの EIPG フィールドおよび IPG フィールドを合わせて 8 ビット時間単位で 96 ビット時間を超える最小 IPG として解釈します。

このビットがリセットされた場合、MAC は ETH_MACCR の EIPG フィールドを無視して、IPG フィールドを 8 ビット時間単位で 96 ビット時間以下の最小 IPG として解釈します。

注：　拡張パケット間隔機能は、全二重モードだけで動作する場合に有効にする必要があります。半二重モードで有効にすると、バックプレッシャ機能およびフレーム送信で望ましくない効果をもたらす場合があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **RESERVED_HDSMS[2:0]**：予約済み。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **USP**：ユニキャストスロープロトコルパケット検出

このビットがセットされた場合、MAC は ETH_MACA0HR および ETH_MACA0LR レジスタで指定されたステーションのユニキャストアドレスを持つスロープロトコルパケットを検出します。MAC は、スロープロトコルマルチキャストアドレス（01-80-C2-00-00-02）を持つスロープロトコルパケットも検出します。

このビットがリセットされた場合、IEEE 802.3-2008、セクション 5 で指定されたスロープロトコルマルチキャストアドレスを持つスロープロトコルパケットを検出します。

ビット 17 **SPEN**：スロープロトコル検出有効化

このビットがセットされた場合、MAC はスロープロトコルパケット（Ether タイプ 0x8809）を処理し、Rx ステータスを提供します。MAC は、無効なサブタイプを持つスロープロトコルパケットを破棄します。

このビットがリセットされた場合、MAC は、エラーなしスロープロトコルパケットをすべてアプリケーションに転送します。MAC は、このパケットを通常タイプパケットとみなします。

ビット 16 **DCRCC**：受信パケット用 CRC チェックの無効化

このビットがセットされた場合、MAC レシーバは受信パケットの CRC フィールドをチェックしません。このビットがリセットされた場合、MAC レシーバは受信パケットの CRC フィールドを必ずチェックします。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **GPSL[13:0]**：ジャイアントパケットのサイズ制限

受信パケットサイズがバイト単位でこのフィールドにプログラムされた値を超える場合、MAC は受信パケットをジャイアントパケットとして宣言します。このフィールドにプログラムされた値は、1,518 バイト以上でなければなりません。それ以外のプログラムされた値は、1,518 バイトとみなされます。

VLAN タグ付きパケットでは、MAC はプログラムされた値に 4 バイトを追加します。ダブル VLAN 処理の有効化オプションが選択されている場合、MAC はダブル VLAN タグ付きパケットのプログラムされた値に 8 バイトを追加します。このフィールドの値は、ETH_MACCR レジスタで GPSLCE ビットがセットされた場合に適用できます。

パケットフィルタ制御レジスタ（ETH_MACPFR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

MAC のパケットフィルタレジスタは、パケットを受信するためのフィルタ制御ビットを含みます。このレジスタの制御の一部は、MAC のアドレスチェックブロックに渡されて、アドレスフィルタリングの最初のレベルを実行します。フィルタの第 2 のレベルは、不良パケットのパスや制御パケットのパスなど、その他の制御に基づいて、着信パケットに対して実行されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DNTU	IPFE	Res.	Res.	Res.	VTFE
rw										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HPF	SAF	SAIF	PCF[1:0]		DBF	PM	DAIF	HMC	HUC	PR
					rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

ビット 31 **RA**：すべてを受信

このビットがセットされた場合、MAC レシーバモジュールは、アドレスフィルタをパスしたかどうかに関係なく、すべての受信パケットをアプリケーションに渡します。SA または DA フィルタの結果は、Rx ステータスワードの対応するビットで更新（通過または失敗）されます。

このビットがリセットされた場合、レシーバモジュールは、SA または DA アドレスフィルタを通過するパケットだけをアプリケーションに渡します。

ビット 30:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **DNTU**：IP パケット経由の非 TCP/UDP のドロップ

このビットがセットされた場合、MAC が IP パケット経由の非 TCP または UDP をドロップします。MAC はレイヤ 4 フィルタによって処理されるこれらのパケットだけを転送します。このビットがリセットされた場合、MAC は IP パケット経由の非 TCP または UDP をすべて転送します。

レイヤ 3 およびレイヤ 4 パケットフィルタの有効化オプションが選択されている場合、このビットは予約済み（デフォルト値で読み出し専用）となります。

ビット 20 **IPFE**：レイヤ 3 およびレイヤ 4 フィルタの有効化

このビットがセットされた場合、MAC は有効化されたレイヤ 3 およびレイヤ 4 フィルタと一致しないパケットをドロップします。レイヤ 3 およびレイヤ 4 フィルタが一致に対して有効になっていない場合、このビットに効果はありません。

このビットがリセットされた場合、MAC はレイヤ 3 およびレイヤ 4 フィールドの一致ステータスに関係なく、すべてのパケットを転送します。

レイヤ 3 およびレイヤ 4 パケットフィルタの有効化オプションが選択されている場合、このビットは予約済み（デフォルト値で読み出し専用）となります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **VTFE**：VLAN タグフィルタの有効化

このビットがセットされた場合、MAC は VLAN タグに一致しない VLAN タグ付きパケットをドロップします。このビットがリセットされた場合、MAC は VLAN タグの一致ステータスに関係なく、すべてのパケットを転送します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HPF**：ハッシュまたは完全フィルタ

このビットがセットされると、アドレスフィルタが HMC または HUC ビットによってセットされた完全フィルタまたはハッシュフィルタに一致する場合、パケットを通過させます。

このビットがリセットされ、HUC または HMC ビットがセットされると、ハッシュフィルタに一致する場合のみ、パケットを通過させます。アドレスフィルタハッシュテーブルの有効化オプションが選択されていない場合、このビットは予約済み（および読み出し専用）となります。

ビット 9 **SAF**：ソースアドレスフィルタの有効化

このビットがセットされた場合、MAC は受信パケットの SA フィールドと、有効な SA レジスタでプログラムされた値とを比較します。比較に失敗した場合、MAC はパケットをドロップします。

このビットがリセットされた場合、MAC は、SA アドレス比較に応じて更新された Rx ステータスの SAF ビットで受信パケットをアプリケーションに転送します。

注： ***IEEE 仕様に従って、SA のビット 47 は予約済みとなります。ただし、MAC は全 48 ビットを比較します。ソフトウェアドライバは、SA のために MAC アドレスレジスタをプログラムしている間、そのことについて考慮する必要があります。***

ビット 8 **SAIF**：SA 逆フィルタ

このビットがセットされた場合、アドレスチェックブロックは、SA アドレス比較の際に逆フィルタモードで動作します。パケットの SA が SA レジスタでプログラムされた値と一致する場合、SA アドレスフィルタで失敗したものとしてマークされます。

このビットがリセットされると、パケットの SA が SA レジスタでプログラムされた値と一致しない場合、SA アドレスフィルタで失敗したものとしてマークされます。

ビット 7:6 **PCF[1:0]**：制御パケットの通過

これらのビットは、すべての制御パケット（ユニキャストおよびマルチキャストポーズパケットを含む）の転送を制御します。

00：MAC は、すべての制御パケットがアプリケーションに届くのをフィルタします。

01：MAC は、アドレスフィルタに失敗した場合もポーズパケットを除いたすべての制御パケットをアプリケーションへ転送します。

10：MAC は、アドレスフィルタに失敗した場合もすべての制御パケットをアプリケーションへ転送します。

11：MAC は、アドレスフィルタを通過する制御パケットを転送します。

ビット 5 **DBF**：ブロードキャストパケットの無効化

このビットがセットされた場合、AFM モジュールは、着信するすべてのブロードキャストパケットをブロックします。さらに、その他すべてのフィルタ設定を上書きします。

このビットがリセットされた場合、AFM モジュールは、受信したすべてのブロードキャストパケットを通過させます。

ビット 4 **PM**：すべてのマルチキャストを通過

このビットがセットされた場合、マルチキャストデスティネーションアドレス（デスティネーションアドレスフィールドの最初のビットが '1'）を持つすべての受信パケットが通過されることを示します。このビットがリセットされた場合、マルチキャストパケットのフィルタは、HMC ビットに依存します。

ビット 3 **DAIF**：DA 逆フィルタ

このビットがセットされた場合、アドレスチェックブロックは、ユニキャストおよびマルチキャストパケットの両方について、DA アドレス比較の際に逆フィルタモードで動作します。このビットがリセットされた場合、通常のパケットフィルタが行われます。

ビット 2 **HMC**：ハッシュマルチキャスト

このビットがセットされた場合、MAC はハッシュテーブルに従って、受信したマルチキャストパケットのデスティネーションアドレスフィルタを実行します。

このビットがリセットされた場合、MAC はマルチキャストパケットについて完全デスティネーションアドレスフィルタリングを実行します。すなわち、DA フィールドを DA レジスタでプログラムされた値と比較します。

アドレスフィルタハッシュテーブルの有効化オプションが選択されていない場合、このビットは予約済み（および読み出し専用）となります。

ビット 1 **HUC**：ハッシュユニキャスト

このビットがセットされた場合、MAC はハッシュテーブルに従って、ユニキャストパケットのデスティネーションアドレスフィルタを実行します。

このビットがリセットされた場合、MAC はユニキャストパケットについて完全デスティネーションアドレスフィルタリングを実行します。すなわち、DA フィールドを DA レジスタでプログラムされた値と比較します。

アドレスフィルタハッシュテーブルの有効化オプションが選択されていない場合、このビットは予約済み（および読み出し専用）となります。

ビット 0 **PR**：プロミスキャスモード

このビットがセットされると、アドレスフィルタモジュールは、デスティネーションまたはソースアドレスに関係なく、着信するすべてのパケットを通過させます。PR がセットされると、Rx ステータスワードの SA または DA フィルタ失敗ステータスビットが常にクリアされます。

ウォッチドッグタイムアウトレジスタ（ETH_MACWTR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

ウォッチドッグタイムアウトレジスタは、受信パケットのウォッチドッグタイムアウトを制御します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PWE	Res.	Res.	Res.	Res.	WTO[3:0]			
							rw					rw			

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PWE**：プログラム可能なウォッチドッグの有効化

このビットがセットされ、ETH_MACCR レジスタの WD ビットがリセットされた場合、WTO フィールドは受信パケットのウォッチドッグタイムアウトとして使用されます。このビットがクリアされた場合、受信パケットのウォッチドッグタイムアウトは、ETH_MACCR レジスタの WD および JE ビットの設定で制御されます。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **WTO[3:0]**：ウォッチドッグタイムアウト

PWE ビットがセットされ、ETH_MACCR レジスタの WD ビットがリセットされた場合、このフィールドは受信パケットのウォッチドッグタイムアウトとして使用されます。受信パケットの長さがこのフィールドの値を超える場合、このパケットは終了し、エラーパケットと宣言されます。

エンコーディングは、次のとおりです。

0x0：2 KB

0x1：3 KB

0x2：4 KB

0x3：5 KB

..

0xC：14 KB

0xD：15 KB

0xE：16383 バイト

0xF：予約済み

注： PWE ビットがセットされた場合、このフィールドの値は 1,522（0x05F2）以上の必要があります。そうしないと、IEEE 802.3 指定の有効なタグ付きパケットが、エラーパケットと宣言されてドロップされます。

ハッシュテーブル 0 レジスタ（ETH_MACHT0R）

アドレスオフセット：0x0010

リセット値：0x0000 0000

ハッシュテーブルレジスタ 0 は、ハッシュテーブルの最初の 32 ビット（64 ビット）を含みます。

ハッシュフィルタの場合、着信パケットのデスティネーションアドレスの内容は、CRC ロジックを通して渡され、CRC レジスタの上位 6 ビットがハッシュテーブルの内容のインデックスに使用されます。最上位ビットは、使用されるレジスタ（ハッシュテーブルレジスタ 0 または 1）を決めます。

デスティネーションアドレスのハッシュ値は、次の方法で計算されます。

1. DA の 32 ビット CRC を計算します（CRC32 の計算手順については IEEE 802.3、セクション 3.2.8 を参照）。
2. 手順 1 で得られた値のビットの反転を実行します。
3. 手順 2 で得られた値の上位 7 または 8 ビットを取得します。

レジスタの対応するビット値が 1 の場合、パケットは受け入れられます。そうでない場合は、拒否されます。ETH_MACFFR で PM ビットがセットされた場合、マルチキャストハッシュ値に関係なく、すべてのマルチキャストパケットが受け入れられます。

ハッシュテーブルレジスタが、MII クロックドメインと二重同期するよう設定された場合、ハッシュテーブルレジスタ X のビット [31:24]（リトルエンディアンモード）またはビット [7:0]（ビッグエンディアンモード）が書き込まれたときのみ、同期がトリガされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HT31T0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HT31T0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **HT31T0[31:0]**：MAC ハッシュテーブルの最初の 32 ビット
このフィールドは、ハッシュテーブルの最初の 32 ビット [31:0] を含みます。

ハッシュテーブル 1 レジスタ（ETH_MACHT1R）

アドレスオフセット：0x0014

リセット値：0x0000 0000

ハッシュテーブルレジスタ 1 は、ハッシュテーブル（64 ビット）の最後の 32ビットを含みます。

ハッシュフィルタの場合、着信パケットのデスティネーションアドレスの内容は、CRC ロジックを通して渡され、CRC レジスタの上位 6 ビットがハッシュテーブルの内容のインデックスに使用されます。最上位ビットは、使用されるレジスタ（ハッシュテーブルレジスタ 0 または 1）を決めます。

デスティネーションアドレスのハッシュ値は、次の方法で計算されます。

1. DA の 32 ビット CRC を計算します（CRC32 の計算手順については IEEE 802.3、セクション 3.2.8 を参照）。
2. 手順 1 で得られた値のビットの反転を実行します。
3. 手順 2 で得られた値の上位 7 または 8 ビットを取得します。

レジスタの対応するビット値が 1 の場合、パケットは受け入れられます。そうでない場合は、拒否されます。ETH_MACFFR で PM ビットがセットされた場合、マルチキャストハッシュ値に関係なく、すべてのマルチキャストパケットが受け入れられます。

ハッシュテーブルレジスタが、MII クロックドメインと二重同期するよう設定された場合、ハッシュテーブルレジスタ X のビット [31:24]（リトルエンディアンモード）またはビット [7:0]（ビッグエンディアンモード）が書き込まれたときのみ、同期がトリガされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HT63T32[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HT63T32[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **HT63T32[31:0]**：MAC ハッシュテーブルの 2 番目の 32 ビット

このフィールドは、ハッシュテーブルの 2 番目の 32 ビット [63:32] を含みます。

VLAN タグレジスタ（ETH_MACVTR）

アドレスオフセット：0x0050

リセット値：0x0000 0000

VLAN タグレジスタは、IEEE 802.1Q VLAN タイプのパケットを特定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIVLRXS	Res.	EIVLS[1:0]		ERIVLT	EDVLP	VTHM	EVLRXS	Res.	EVLS [1:0]		DOVLTC	ERSVLM	ESVL	VTIM	ETV
rw		rw		rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VL[15:0]															
rw															

ビット 31 **EIVLRXS**：Rx ステータスでの内部 VLAN タグの有効化

このビットがセットされた場合、MAC は Rx ステータスで内部 VLAN タグを提供します。このビットがリセットされた場合、MAC は Rx ステータスで内部 VLAN タグを提供しません。

ビット 30 予約済み

ビット 29:28 **EIVLS[1:0]**：受信時の内部 VLAN タグ削除の有効化

このフィールドは、受信したパケットの内部 VLAN タグでの削除操作を示します。

00：削除しない

01：VLAN フィルタを通過した場合に削除

10：VLAN フィルタに失敗した場合に削除

11：常に削除

ビット 27 **ERIVLT**：内部 VLAN タグの有効化

このビットおよび EDVLP フィールドがセットされた場合、MAC レシーバは内部 VLAN タグ（ある場合）の操作を有効にします。このビットリセットされた場合、MAC レシーバは外部 VLAN タグ（ある場合）の操作を有効にします。ERSVLM ビットは、フィルタや一致に対して有効にする VLAN タイプを決定します。

ビット 26 **EDVLP**：ダブル VLAN 処理の有効化

このビットがセットされた場合、MAC は Tx および Rx で最大 2 つの VLAN タグ（ある場合）の処理を有効にします。このビットがリセットされた場合、MAC は Tx および Rx で最大 1 つの VLAN タグ（ある場合）の処理を有効にします。

ビット 25 **VTHM**：VLAN タグハッシュテーブル一致の有効化

このビットがセットされた場合、VLAN タグの CRC の最上位 4 ビットを使用して、ETH_MACVLANHTR レジスタの内容をインデックスします。インデックスに対応する VLAN ハッシュテーブルレジスタでの値 1 は、パケットが VLAN ハッシュテーブルに一致したことを示します。

ETV ビットがセットされた場合、12 ビット VLAN 識別子（VID）の CRC が比較に使用されます。ETV ビットがリセットされた場合、16 ビット VLAN タグの CRC が比較に使用されます。

このビットがリセットされた場合、VLAN ハッシュー致操作は実行されません。VLAN ハッシュ機能が有効ではない場合、このビットは予約済み（デフォルト値で読み出し専用）となります。

ビット 24 **EVLRXS**：Rx ステータスでの VLAN タグの有効化

このビットがセットされた場合、MAC は Rx ステータスで外部 VLAN タグを提供します。このビットがリセットされた場合、MAC は Rx ステータスで外部 VLAN タグを提供しません。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **EVLS[1:0]**：受信時の VLAN タグ削除の有効化

このフィールドは、受信したパケットの外部 VLAN タグでの削除操作を示します。

00：削除しない

01：VLAN フィルタを通過した場合に削除

10：VLAN フィルタに失敗した場合に削除

11：常に削除

ビット 20 **DOVLTC**：VLAN タイプチェックの無効化

このビットがセットされた場合、MAC は ERIVLT ビットによって指定された VLAN タグが S-VLAN または C-VLAN のどのタイプであるかをチェックしません。

このビットがリセットされた場合、VLAN タグタイプが ERSVLM ビットで指定されたものと同様であるときのみ、MAC は ERIVLT ビットで指定された VLAN タグのフィルタまたは一致を実行します。

ビット 19 **ERSVLM**：受信 S-VLAN 一致の有効化

このビットがセットされた場合、MAC レシーバは S-VLAN（タイプ = 0x88A8）のフィルタまたは一致を有効にします。このビットがリセットされた場合、MAC レシーバは C-VLAN（タイプ = 0x8100）のフィルタまたは一致を有効にします。

ERIVLT ビットは、フィルタや一致に対して考慮される VLAN タグの位置を決定します。

ビット 18 **ESVL**：S-VLAN の有効化

このビットがセットされた場合、MAC トランスミッタおよびレシーバは S-VLAN パケット（タイプ = 0x88A8）を、有効な VLAN タグ付きパケットとみなします。

ビット 17 **VTIM**：VLAN タグ反転一致の有効化

このビットがセットされた場合、VLAN タグの反転一致が有効になります。一致する VLAN タグがないパケットが、一致とマークされます。リセットされた場合、このビットは VLAN タグの完全一致が有効になります。一致する VLAN タグがあるパケットが、一致とマークされます。

ビット 16 **ETV**：12 ビット VLAN タグ比較の有効化

このビットがセットされた場合、完全な 16 ビット VLAN タグの代わりに 12 ビット VLAN 識別子が比較およびフィリタに使用されます。VLAN タグのビット [11:0] が、受信した VLAN タグ付きパケット内の対応するフィールドと比較されます。同様に、有効化された場合、受信パケットの VLAN タグの 12 ビットだけがハッシュベース VLAN フィルタに使用されます。

このビットがリセットされた場合、受信した VLAN パケットの 15 番目と 16 番目のバイトの全 16 ビットが比較および VLAN ハッシュフィルタに使用されます。

ビット 15:0 **VL[15:0]**：受信パケット用 VLAN タグ識別子

このフィールドは、VLAN パケットを識別する 802.1Q VLAN タグを含んでいます。この VLAN タグ識別子は、VLAN パケットのために受信されるパケットの 15 番目および 16 番目のバイトと比較されます。次のリストでは、このフィールドのビットを示しています。

ビット [15:13]：ユーザ優先度

ビット 12：正規フォーマットインジケータ（CFI）またはドロップ適格インジケータ（DEI）

ビット [11:0]：VLAN タグの VLAN 識別子（VID）フィールド

ETV ビットがセットされると、VID のみが比較に使用されます。

このフィールド（ETV がセットされている場合、[11:0]）がすべて 0 の場合、MAC は、VLAN タグの比較のために 15 番目および 16 番目のバイトをチェックせず、タイプフィールドの値が 0x8100 または 0x88a8 のすべてのパケットを VLAN パケットとして宣言します。

VLAN ハッシュテーブルレジスタ（ETH_MACVHTR）

アドレスオフセット：0x0058

リセット値：0x0000 0000

ETH_MACHT1R レジスタの ERSVLM ビットがセットされた場合、16 ビット VLAN ハッシュテーブルレジスタが VLAN タグに基づいたグループアドレスフィルタに使用されます。ハッシュフィルタでは、着信パケットの 16 ビット VLAN タグまたは 12 ビット VLAN ID の内容（ETH_MACVTR レジスタの ETV ビットに基づく）は、CRC ロジックを通じて渡されます。計算された CRC の上位 4 ビットを使用して、VLAN ハッシュテーブルの内容をインデックスします。たとえば、1000 のハッシュ値は、VLAN ハッシュテーブルのビット 8 を選択します。

デスティネーションアドレスのハッシュ値は、次の方法で計算されます。

1. VLAN タグまたは ID の 32 ビット CRC を計算します（CRC32 を計算する手順については、IEEE 802.3 のセクション 3.2.8 を参照してください）。
2. 手順 1 で得られた値のビットの反転を実行します。
3. 手順 2 で得られた値の上位 4 ビットを取得します。

VLAN ハッシュテーブルレジスタが、MII クロックドメインと二重同期するよう設定された場合、このレジスタのビット [15:8]（リトルエンディアンモード）またはビット [7:0]（ビッグエンディアンモード）が書き込まれたときのみ、同期がトリガされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VLHT[15:0]															
rw															

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:0　**VLHT[15:0]**：VLAN ハッシュテーブル

このフィールドは、16 ビット VLAN ハッシュテーブルを含みます。

VLAN 内包レジスタ（ETH_MACVIR）

アドレスオフセット：0x0060

リセット値：0x0000 0000

VLAN タグ内包または書き換えレジスタには、送信パケットで挿入または書き換えを行うための VLAN タグが含まれます。これには、VLAN タグ挿入制御も含まれます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLTI	CSVL	VLP	VLC [1:0]	
											rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VLT[15:0]															
rw															

ビット 31:21　予約済みであり、リセット値に保持する必要があります。

ビット 20 **VLTI**：VLAN タグ入力

このビットがセットされた場合、Tx パケットで挿入または書き換える VLAN タグを、Tx ディスクリプタから取得する必要があることを示します。

ビット 19 **CSVL**：C-VLAN または S-VLAN

このビットがセットされた場合、S-VLAN タイプ（0x88A8）が送信パケットの 13 番目および 14 番目のバイトで挿入または書き換えられます。このビットがリセットされた場合、C-VLAN タイプ（0x8100）が送信パケットの 13 番目および 14 番目のバイトで挿入または書き換えられます。

ビット 18 **VLP**：VLAN 優先順位の制御

このビットがセットされた場合、制御ビット [17:16] が VLAN の削除、挿入、書き換えに使用されます。このビットがリセットされると、mti_vlan_ctrl_i 制御入力が使用され、ビット [17:16] は無視されます。

ビット 17:16 **VLC[1:0]**：送信パケットの VLAN タグ制御

00：VLAN タグの削除、挿入、書き換えはありません。

01：VLAN タグを削除します。MAC が VLAN タグ付きで送信されたパケットすべての VLAN タイプ（バイト 13 および 14）と VLAN タグ（バイト 15 および 16）を削除します。

10：VLAN タグを挿入します。MAC がタイプ値（0x8100 または 0x88a8）をバイト 13 および 14 に挿入した後に、VLT をパケットのバイト 15 および 16 に挿入します。この操作は、VLAN タグがすでにあるかどうかに関係なく、すべての送信パケットで実行されます。

11：VLAN タグを書き換えます。MAC がすべての VLAN タイプ送信パケットのバイト 15 および 16 の VLT を書き換えます（バイト 13 および 14 は 0x8100 または 0x88a8 です）。

注：　このフィールドへの変更は、パケットの開始時にのみ有効になります。パケットが送信されているときに、このレジスタフィールドに書き込む場合、後続のパケットのみが更新された値を使用でき、現在のパケットは更新された値を使用しません。

ビット 15:0 **VLT[15:0]**：送信パケット用 VLAN タグ

このフィールドは、挿入または書き換える VLAN タグの値を含んでいます。この値は、送信ラインが非アクティブである場合や初期化フェーズ中のみ変更する必要があります。

ビット [15:13] はユーザの優先度フィールドで、ビット 12 は CFI/DEI フィールドで、ビット [11:0] は VLAN タグの VID フィールドです。

次のリストでは、このフィールドのビットを示しています。

ビット [15:13]：ユーザー優先度

ビット 12：正規フォーマットインジケータ（CFI）またはドロップ適格インジケータ（DEI）

ビット [11:0]：VLAN タグの VLAN 識別子（VID）フィールド

内部 VLAN 内包レジスタ（ETH_MACIVIR）

アドレスオフセット：0x0064

リセット値：0x0000 0000

内部 VLAN タグ内包または書き換えレジスタには、送信パケットで挿入または書き換えを行うための内部 VLAN タグが含まれます。これには、内部 VLAN タグ挿入制御も含まれます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLTI	CSVL	VLP	VLC [1:0]	
											rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VLT[15:0]															
rw															

ビット 31:21　予約済みであり、リセット値に保持する必要があります。

ビット 20 **VLTI**：VLAN タグ入力

このビットがセットされた場合、Tx パケットで挿入または書き換える VLAN タグを、Tx ディスクリプタから取得する必要があることを示します。

ビット 19 **CSVL**：C-VLAN または S-VLAN

このビットがセットされた場合、S-VLAN タイプ（0x88A8）が送信パケットの 13 番目および 14 番目のバイトで挿入または書き換えられます。このビットがリセットされた場合、C-VLAN タイプ（0x8100）が送信パケットの 13 番目および 14 番目のバイトで挿入または書き換えられます。

ビット 18 **VLP**：VLAN 優先順位の制御

このビットがセットされた場合、VLC フィールドが VLAN の削除、挿入、書き換えに使用されます。このビットがリセットされると、mti_vlan_ctrl_i 制御入力が使用され、VLC フィールドは無視されます。

ビット 17:16 **VLC[1:0]**：送信パケットの VLAN タグ制御

00：VLAN タグの削除、挿入、書き換えはありません。

01：VLAN タグを削除します。

MAC が VLAN タグ付きで送信されたパケットすべての VLAN タイプ（バイト 17 および 18）と VLAN タグ（バイト 19 および 20）を削除します。

10：VLAN タグを挿入します。

MAC がタイプ値（0x8100 または 0x88a8）をバイト 17 および 18 に挿入した後に、VLT をパケットのバイト 19 および 20 に挿入します。この操作は、VLAN タグがすでにあるかどうかに関係なく、すべての送信パケットで実行されます。

11：VLAN タグを書き換えます。

MAC がすべての VLAN タイプ送信パケットのバイト 19 および 20 の VLT を書き換えます（バイト 17 および 18 は 0x8100 または 0x88a8 です）。

注：　このフィールドへの変更は、パケットの開始時にのみ有効になります。パケットが送信されているときに、このレジスタフィールドに書き込む場合、後続のパケットのみが更新された値を使用でき、現在のパケットは更新された値を使用しません。

ビット 15:0 **VLT[15:0]**：送信パケット用 VLAN タグ

このフィールドは、挿入または書き換える VLAN タグの値を含んでいます。この値は、送信ラインが非アクティブである場合や初期化フェーズ中のみ変更する必要があります。

ビット [15:13] はユーザの優先度フィールドで、ビット 12 は CFI/DEI フィールドで、ビット [11:0] は VLAN タグの VID フィールドです。

次のリストでは、このフィールドのビットを示しています。

ビット [15:13]：ユーザー優先度

ビット 12：正規フォーマットインジケータ（CFI）またはドロップ適格インジケータ（DEI）

ビット [11:0]：VLAN タグの VLAN 識別子（VID）フィールド

Tx キューフロー制御レジスタ（ETH_MACQTxFCR）

アドレスオフセット：0x0070

リセット値：0x0000 0000

フロー制御レジスタは、MAC のフロー制御モジュールによる制御パケット（ポーズコマンド）の生成と受信を制御します。ビジービットが 1 にセットされているレジスタに書き込むと、フロー制御ブロックがトリガされ、ポーズパケットを生成します。制御パケットのフィールドは、802.3x 仕様での規定に従って選択され、このレジスタのポーズ時間の値が制御パケットのポーズ時間フィールドで使用されます。ビジービットは、制御パケットがケーブルに転送されるまでセット状態を維持します。アプリケーションは、レジスタに書き込む前に、ビジービットがクリアされていることを確認する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PT[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DZPQ	PLT[2:0]			Res.	Res.	TFE	FCB_BPA
								rw	rw					rw	rw

ビット 31:16 **PT[15:0]**：ポーズ時間

このフィールドは、Tx 制御パケットのポーズ時間フィールドで使用される値を保持します。ポーズ時間ビットが (G)MII クロックドメインに二重同期されるように設定されている場合、このレジスタへの連続書き込みは、デスティネーションクロックドメインの少なくとも 4 クロックサイクル後にのみ実行されなければなりません。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **DZPQ**：ゼロクォンタムポーズの無効化

このビットがセットされた場合、FIFO レイヤからのフロー制御信号（MTL または外部サイドバンドフロー制御信号sbd_flowctrl_i か mti_flowctrl_i）のネゲート時の、ゼロクォンタムポーズパケットの自動生成を無効になります。

このビットがリセットされた場合、ゼロクォンタムポーズパケットの自動生成を伴う通常動作が有効になります。

ビット 6:4 **PLT[2:0]**：ポーズロー閾値

このフィールドは、入力フロー制御信号 mti_flowctrl_i（または sbd_flowctrl_i）がポーズパケットの自動再送信についてチェックされるポーズタイマの閾値を設定します。

この閾値は、常にビット [31:16] で設定されたポーズ時間未満でなければなりません。たとえば、PT = 100H（256 スロット時間）であり、PLT = 001 の場合、最初のポーズパケットの送信後、mti_flowctrl_i 信号が 228（256-28）スロット時間の時点でアサートされた場合に、2 番目のポーズパケットが自動的に送信されます。

次のリストでは異なる値での閾値を示しています。

000：ポーズ時間 - 4 スロット時間（PT - 4 スロット時間）

001：ポーズ時間 - 28 スロット時間（PT - 28 スロット時間）

010：ポーズ時間 - 36 スロット時間（PT - 36 スロット時間）

011：ポーズ時間 - 144 スロット時間（PT - 144 スロット時間）

100：ポーズ時間 - 256 スロット時間（PT - 256 スロット時間）

101：ポーズ時間 - 512 スロット時間（PT - 512 スロット時間）

110～111：予約済み

スロット時間は、MII インタフェース上で 512 ビット（64 バイト）を送信するための所要時間として定義されます。

この（概算の）計算は、パケットサイズ（64、1518、2000、9018、16384、または 32768）+ 2 ポーズパケットサイズ + スロット時間の IPG です。

ビット 3:2　予約済みであり、リセット値に保持する必要があります。

ビット 1 **TFE**：送信フロー制御の有効化

全二重モード：このビットがセットされた場合、MAC は、Tx ポーズパケットに対するフロー制御動作を有効にします。このビットがリセットされた場合、MAC のフロー制御動作は無効になり、MAC はポーズパケットを送信しません。

半二重モード：このビットがセットされた場合、MAC はバックプレッシャ動作を有効にします。このビットがリセットされた場合、バックプレッシャ機能は無効になります。

ビット 0 **FCB_BPA**：フロー制御ビジーまたはバックプレッシャ有効

このビットは、全二重モードで、ポーズパケットを開始し、半二重モードでは、TFE ビットがセットされている場合、バックプレッシャ機能を有効にします。

全二重モード：このレジスタに書き込む前に、このビットが 0 として読み出されなければなりません。ポーズパケットを開始するには、アプリケーションはこのビットを ‘1’ にセットする必要があります。制御パケットの転送時、このビットは、パケット送信が進行中であることを示すために、セットされたままです。ポーズパケットの送信が完了すると、MAC はこのビットを ‘0’ にリセットします。このビットがクリアされるまで、このレジスタへの書き込みが禁止されます。

半二重モード：このビットが半二重モードでセットされた場合（および TFE ビットがセットされた場合）、MAC はバックプレッシャをアサートします。バックプレッシャ時、MAC が新しいパケットを受信すると、トランスミッタは、JAM パターンの送出を開始し、結果として衝突が発生します。この制御レジスタビットは、バックプレッシャ機能では mti_flowctrl_i 入力信号と論理和が取られます。MAC が全二重モードに設定されると、BPA は自動的に無効になります。

Rx フロー制御レジスタ（ETH_MACRxFCR）

アドレスオフセット：0x0090

リセット値：0x0000 0000

受信フロー制御レジスタは、受信したポーズパケットに基づいて MAC 送信のポーズを制御します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESERVED_PFCE	Res.	Res.	Res.	Res.	Res.	Res.	UP	RFE
							r							rw	rw

ビット 31:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**UP**：ユニキャストポーズフレーム検出

ポーズパケットは、IEEE 802.3 で指定されている一意なマルチキャストアドレスを持つ場合に処理されます。このビットがセットされた場合、MAC はステーションのユニキャストアドレスを持つポーズパケットも検出できます。このユニキャストアドレスは ETH_MACA0HR および ETH_MACA0LR での指定に従う必要があります。

このビットがリセットされた場合、MAC は一意なマルチキャストアドレスを持つポーズフレームのみを検出します。

注：　マルチキャストアドレスが一意なマルチキャストアドレスではない場合、MAC はポーズパケットを処理しません。これは、優先フロー制御（PFC）が有効になっている場合、受信した PFC パケットにも適用できます。一意なマルチキャストアドレス（0x01_80_C2_00_00_01）は、IEEE 802.1 Qbb-2011 での指定に従います。

ビット 0　**RFE**：受信フロー制御有効化

このビットがセットされ、MAC が全二重モードで動作している場合、受信したポーズパケットをデコードして、指定された時間（ポーズ時間）の間、トランスミッタを無効にします。このビットがリセットされた場合や MAC が半二重モードで動作している場合、ポーズパケットのデコード機能は無効です。

PFC が有効になっている場合、フロー制御は PFC パケットに対して有効です。MAC は、受信した PFC パケットをデコードして、受信したポーズ時間の間、一致優先度で送信キューを無効にします。

割り込みステータスレジスタ（ETH_MACISR）

アドレスオフセット：0x00B0

リセット値：0x0000 0000

割り込みステータスレジスタには、割り込みのステータスが含まれます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESERVED_GPIIS	RXSTSIS	TXSTSIS	TSIS	RESERVED_MMCRXIPIS	MMCTXIS	MMCRXIS	MMCIS	Res.	Res.	LPIIS	PMTIS	PHYIS	RESERVED_PCSANCIS	RESERVED_PCSLCHGIS	Res.
r	r	r	r	r	r	r	r			r	r	r	r	r	

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15 **RESERVED_GPIIS**：予約済み。

ビット 14 **RXSTSIS**：受信ステータス割り込み

このビットは、受信パケットのステータスを示します。このビットは、ETH_MACISR レジスタで RWT ビットがセットされた場合にセットされます。このビットは、ETH_MACISR レジスタで対応する割り込みソースビットが読み出された場合にクリアされます。

ビット 13 **TXSTSIS**：送信ステータス割り込み

このビットは、送信パケットのステータスを示します。このビットは、ETH_MACISR レジスタで次のいずれかのビットがセットされた場合にセットされます。

- 過剰衝突（EXCOL）
- 遅れ衝突（LCOL）
- 過剰遅延（EXDEF）
- キャリア喪失（LCARR）
- キャリアなし（NCARR）
- ジャバータイムアウト（TJT）

このビットは、ETH_MACISR レジスタで対応する割り込みソースビットが読み出された場合にクリアされます。

ビット 12 **TSIS**：タイムスタンプ割り込みステータス

タイムスタンプ機能が有効になっている場合、以下の条件のいずれかが当てはまる場合に、このビットがセットされます。

– システム時間の値が、目標時間上位および下位レジスタで指定された値以上になっています。

– 秒レジスタにオーバーフローがあります。

– 目標時間エラーが発生しており、プログラムされた目標時間をすでに過ぎています。

補助スナップショット機能が有効になっている場合、補助スナップショットトリガがアサートされた場合に、このビットがセットされます。

ドロップ送信ステータスが MTL で有効になっている場合、キャプチャされた送信タイムスタンプが ETH_MACTxTSSNR および ETH_MACTxTSSSR レジスタで更新されると、このビットがセットされます。

PTP オフロード機能が有効になっている場合、キャプチャされた送信タイムスタンプが ETH_MACTxTSSNR および ETH_MACTxTSSSR レジスタで更新されたとき、PTO 生成遅延リクエストおよび Pdelay リクエストパケットに対して、このビットがセットされます。

このビットは、ETH_MACTSSR レジスタで対応する割り込みソースビットが読み出された場合にクリアされます。

ビット 11 **RESERVED_MMCRXIPIS**：予約済み。

ビット 10 **MMCTXIS**：MMC 送信割り込みステータス

このビットは、MMC 送信割り込みレジスタで割り込みが生成されると、ハイにセットされます。このビットは、この割り込みレジスタのすべてのビットがクリアされたときにクリアされます。

このビットは、MAC 管理カウンタ（MMC）の有効化オプションを選択したときのみ有効です。

ビット 9 **MMCRXIS**：MMC 受信割り込みステータス

このビットは、MMC 受信割り込みレジスタで割り込みが生成されると、ハイにセットされます。このビットは、この割り込みレジスタのすべてのビットがクリアされたときにクリアされます。

このビットは、MAC 管理カウンタ（MMC）の有効化オプションを選択したときのみ有効です。

ビット 8 **MMCIS**：MMC 割り込みステータス

このビットは、ビット 11、ビット 10、ビット 9 のいずれかがハイにセットされたときに、ハイにセットされます。このビットは、それらのビットすべてがローのときのみクリアされます。このビットは、MAC 管理カウンタ（MMC）の有効化オプションを選択したときのみ有効です。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **LPIIS**：LPI 割り込みステータス

省電力型 Ethernet 機能が有効になっている場合、このビットは、MAC トランスミッタまたはレシーバでの LPI 状態の移行または終了に対してセットされます。ETH_MACLCSR レジスタの TLPIEN ビットが読み出されると、このビットはクリアされます。その他のモードでは、このビットは予約済みです。

ビット 4 **PMTIS**：PMT 割り込みステータス

このビットは、パワーダウンモードで、マジックパケットまたは Wake-on-LAN パケットが受信されたときに、セットされます（ETH_MACPCSR レジスタの RWKPRCVD および MGKPRCVD ビット）。このビットは、ビット [6:5] が ETH_MACPCSR レジスタの読み出し動作のためにクリアされたときにクリアされます。

このビットは、電源管理の有効化オプションを選択したときのみ有効です。

ビット 3 **PHYIS**：PHY 割り込み

このビットは、立ち上がりエッジが phy_intr_i 入力で検出されたときにセットされます。このレジスタが読み出されると、このビットはクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

割り込み有効レジスタ（ETH_MACIER）

アドレスオフセット：0x00B4

リセット値：0x0000 0000

割り込み有効レジスタには、割り込みを生成するためのマスクが含まれます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RXSTSIE	TXSTSIE	TSIE	Res.	Res.	Res.	Res.	Res.	Res.	LPIIE	PMTIE	PHYIE	RESERVED_PCSANCIE	RESERVED_PCSLCHGIE	Res.
	rw	rw	rw							rw	rw	rw	r	r	

ビット 31:15　予約済みであり、リセット値に保持する必要があります。

ビット 14　**RXSTSIE**：受信ステータス割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの RXSTSIS ビットの設定のため、割り込み信号のアサートが有効になります。

ビット 13　**TXSTSIE**：送信ステータス割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの TXSTSIS ビットの設定のため、割り込み信号のアサートが有効になります。

ビット 12　**TSIE**：タイムスタンプ割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの TSIS ビットの設定のため、割り込み信号のアサートが有効になります。

このビットは、IEEE 1588 タイムスタンプサポートの有効化オプションが選択されたときにのみ有効です。それ以外、このビットは予約済みです。

ビット 11:6　予約済みであり、リセット値に保持する必要があります。

ビット 5　**LPIIE**：LPI 割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの LPIIS ビットの設定のため、割り込み信号のアサートが有効になります。

このビットは、省電力型 Ethernet（EEE）の有効化オプションが選択されたときにのみ有効です。それ以外、このビットは予約済みです。

ビット 4　**PMTIE**：PMT 割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの PMTIS ビットの設定のため、割り込み信号のアサートが有効になります。

ビット 3　**PHYIE**：PHY 割り込み有効化

このビットがセットされた場合、ETH_MACISR レジスタの PHYIS ビットの設定のため、割り込み信号のアサートが有効になります。

ビット 2:0　予約済みであり、リセット値に保持する必要があります。

参考資料

Rx Tx ステータスレジスタ（ETH_MACRxTxSR）

アドレスオフセット：0x00B8

リセット値：0x0000 0000

送受信ステータスレジスタには、送受信のエラーステータスが含まれます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWT	Res.	Res.	EXCOL	LCOL	EXDEF	LCARR	NCARR	TJT
							r			r	r	r	r	r	r

ビット 31:9　予約済みであり、リセット値に保持する必要があります。

ビット 8　**RWT**：受信ウォッチドッグタイムアウト

このビットは、2,048 バイト（ジャンボパケットモードが有効の場合は 10,240 バイト）より長いパケットが受信され、ETH_MACCR レジスタで WD ビットがリセットされたときにセットされます。このビットは、16,383 バイトより長いパケットが受信され、ETH_MACCR レジスタで WD ビットがセットされたときにセットされます。

ビット 7:6　予約済みであり、リセット値に保持する必要があります。

ビット 5　**EXCOL**：過剰衝突

MTL_Operation_Mode レジスタで DTXSTS ビットがセットされた場合、このビットは、現在のパケットの送信を試みている間に 16 の連続した衝突が発生した後、送信が中止されたことを示します。ETH_MACCR レジスタで DR ビットがセットされている場合、このビットが最初の衝突後にセットされて、パケットの送信が中止されます。

ビット 4　**LCOL**：遅れ衝突

MTL_Operation_Mode レジスタで DTXSTS ビットがセットされた場合、このビットは、MII モードで衝突ウィンドウ（プリアンブルを含めて 64 バイト）の後で衝突が発生したためにパケット送信が中止されたことを示します。

このビットは、アンダーフローエラーが発生した場合は無効です。

ビット 3　**EXDEF**：過剰遅延

MTL_Operation_Mode レジスタで DTXSTS ビットがセットされ、ETH_MACCR レジスタで DC ビットがセットされた場合、このビットは、24,288 ビット時間（1000 Mbps モードやジャンボパケットが有効になっている場合は 155,680）を超える過剰遅延があったために送信が終了したことを示します。

ビット 2　**LCARR**：キャリア喪失

MTL_Operation_Mode レジスタで DTXSTS ビットがセットされた場合、このビットは、パケット送信中にキャリアの喪失が発生した、すなわち、パケット送信中に 1 つ以上の送信クロック周期の間、phy_crs_i 信号が無効であったことを示します。このビットは、衝突なしで送信されたパケットにのみ有効です。

ビット 1　**NCARR**：キャリアなし

MTL_Operation_Mode レジスタで DTXSTS ビットがセットされた場合、このビットは、PHY からのキャリア信号がプリアンブル送信の最後に存在しないことを示します。

ビット 0　**TJT**：送信ジャバータイムアウト

このビットは、パケットサイズが 2,048 バイト（ジャンボパケットモードが有効の場合は 10,240 バイト）を超えるときに発生する送信ジャバータイマが終了して、ETH_MACCR レジスタでJD ビットがリセットされたことを示します。このビットは、パケットサイズが 16,383 バイトを超え、ETH_MACCR レジスタで JD ビットがセットされたときにセットされます。

PMT 制御ステータスレジスタ（ETH_MACPCSR）

アドレスオフセット：0x00C0

リセット値：0x0000 0000

PMT 制御およびステータスレジスタは、coreConsultant で PMT モジュールを選択した場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RWKFILTRST	Res.	Res.	RWKPTR[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw			r												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	RWKPFE	GLBLUCAST	Res.	Res.	RWKPRCVD	MGKPRCVD	Res.	Res.	RWKPKTEN	MGKPKTEN	PWRDWN
					rw	rw			r	r			rw	rw	rw

ビット 31 **RWKFILTRST**：リモートウェイクアップパケットフィルタレジスタのポインタのリセット

このビットがセットされた場合、リモートウェイクアップパケットフィルタレジスタのポインタを 3'b000 にリセットします。1 クロックサイクル後に自動的にクリアされます。

ビット 30:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **RWKPTR[4:0]**：リモートウェイクアップ FIFO のポインタ

このフィールドは、リモートパケットフィルタレジスタのポインタの現在の値（0～7）を提供します。このポインタが 7 である場合、リモートウェイクアップパケットフィルタレジスタへの書き込みが発生すると、その内容が eth_mii_rx_clk ドメインへ転送されます。

ビット 23:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **RWKPFE**：リモートウェイクアップパケット転送の有効化

このビットが RWKPKTEN とともにセットされた場合、MAC レシーバは期待されるウェイクアップフレームを受信するまで、すべての受信フレームをドロップします。受信したウェイクアップフレームを含むそのイベント以降のすべてのフレームはアプリケーションに転送されます。このビットは、ウェイクアップパケットの受信時にセルフクリアされます。

アプリケーションで、期待されるウェイクアップフレームが受信される前でも、このビットをクリアできます。このような場合、MAC は受信パケットがアプリケーションに転送されるデフォルトの動作に戻ります。このビットは、RWKPKTEN がハイに、PWRDWN がローにセットされているときのみ、セットされなければなりません。このビットの設定は、PWRDWN がハイにセットされているときには効果がありません。

注： ***マジックパケットの有効化およびウェイクアップフレームの有効化が、このビットの設定に合わせてセットされ、マジックパケットがウェイクアップフレームより先に受信される場合、このビットはマジックパケットの受信時にセルフクリアされ、受信したマジックパケットはドロップされ、受信したマジックパケット以降のすべてのフレームは、アプリケーションに転送されます。***

ビット 9 **GLBLUCAST**：グローバルユニキャスト

このビットがセットされた場合、MAC（DAF）アドレス認識によってフィルタされたすべてのユニキャストパケットは、リモートウェイクアップパケットとして検出されます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RWKPRCVD**：リモートウェイクアップパケットの受信

このビットがセットされた場合、リモートウェイクアップパケットを受信したため、電源管理イベントが生成されることを示します。このレジスタが読み出されると、このビットはクリアされます。

ビット 5 **MGKPRCVD**：マジックパケットの受信

このビットがセットされた場合、マジックパケットを受信したため、電源管理イベントが生成されることを示します。このレジスタが読み出されると、このビットはクリアされます。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **RWKPKTEN**：リモートウェイクアップパケットの有効化

このビットがセットされた場合、MAC がリモートウェイクアップパケットを受信したときに、電源管理イベントが生成されます。

ビット 1 **MGKPKTEN**：マジックパケットの有効化

このビットがセットされた場合、MAC がマジックパケットを受信したときに、電源管理イベントが生成されます。

ビット 0 **PWRDWN**：パワーダウン

このビットがセットされた場合、MAC レシーバは期待されるマジックパケットまたはリモートウェイクアップパケットを受信するまで、すべての受信パケットをドロップします。このビットはセルフクリアされ、パワーダウンモードは無効になります。ソフトウェアで、期待されるマジックパケットまたはリモートウェイクアップパケットが受信される前に、このビットをクリアできます。このビットがクリアされた後に MAC によって受信されたパケットは、アプリケーションに転送されます。このビットは、マジックパケット有効、グローバルユニキャスト、またはリモートウェイクアップパケット有効ビットがハイにセットされているときのみ、セットされなければなりません。

注： パワーダウンモード中に CSR クロックをゲートオフできます。ただし、CSR クロックがゲートオフされると、このレジスタで読み出しまたは書き込み操作を実行できません。そのため、ソフトウェアはこのビットをクリアできません。

リモートウェイクアップパケットフィルタレジスタ（ETH_MACRWKPFR）

アドレスオフセット：0x00C4

リセット値：0x0000 0000

アドレス 0C4H の wkuppktfilter_reg レジスタは、ウェイクアップパケットフィルタレジスタをロードします。

ウェイクアップパケットフィルタレジスタで値をロードするには、レジスタ全体（wkuppktfilter_reg）が書き込まれる必要があります。wkuppktfilter_reg0、wkuppktfilter_reg1、wkuppktfilter_reg31 それぞれに対して、アドレス（0C4H）で 8、16、または 32 のレジスタ値を連続でロードすることによって、wkuppktfilter_reg レジスタをロードします。wkuppktfilter_reg レジスタは、類似した方法で読み出されます。Ethernet ペリフェラルは、ETH_MACPCSR レジスタのビット [26:24] で wkuppktfilter_reg レジスタの現在のポインタ値を更新します。

表 541. リモートウェイクアップパケットフィルタレジスタ

ETH_MACRWKPFR + #	フィールド							
0	フィルタ 0 バイトマスク							
1	フィルタ 1 バイトマスク							
2	フィルタ 2 バイトマスク							
3	フィルタ 3 バイトマスク							
4	予約済み	フィルタ 3 コマンド	予約済み	フィルタ 2 コマンド	予約済み	フィルタ 1 コマンド	予約済み	フィルタ 0 コマンド
5	フィルタ 3 オフセット		フィルタ 2 オフセット		フィルタ 1 オフセット		フィルタ 0 オフセット	
6	フィルタ 1 CRC-16				フィルタ 0 CRC-16			
7	フィルタ 3 CRC-16				フィルタ 2 CRC-16			
8	フィルタ 4 バイトマスク							
9	フィルタ 5 バイトマスク							
10	フィルタ 6 バイトマスク							
11	フィルタ 7 バイトマスク							
12	予約済み	フィルタ 7 コマンド	予約済み	フィルタ 6 コマンド	予約済み	フィルタ 5 コマンド	予約済み	フィルタ 4 コマンド
13	フィルタ 7 オフセット		フィルタ 6 オフセット		フィルタ 5 オフセット		フィルタ 4 オフセット	
14	フィルタ 5 CRC-16				フィルタ 4 CRC-16			
15	フィルタ 7 CRC-16				フィルタ 6 CRC-16			
16	フィルタ 8 バイトマスク							
17	フィルタ 9 バイトマスク							
18	フィルタ 10 バイトマスク							
19	フィルタ 11 バイトマスク							
20	予約済み	フィルタ 11 コマンド	予約済み	フィルタ 10 コマンド	予約済み	フィルタ 9 コマンド	予約済み	フィルタ 8 コマンド
21	フィルタ 11 オフセット		フィルタ 10 オフセット		フィルタ 9 オフセット		フィルタ 8 オフセット	
22	フィルタ 9 CRC-16				フィルタ 8 CRC-16			
23	フィルタ 11 CRC-16				フィルタ 10 CRC-16			
24	フィルタ 12 バイトマスク							

表 541. リモートウェイクアップパケットフィルタレジスタ（続き）

ETH_MACRWKPFR + #	フィールド							
25	フィルタ 13 バイトマスク							
26	フィルタ 14 バイトマスク							
27	フィルタ 15 バイトマスク							
28	予約済み	フィルタ 15 コマンド	予約済み	フィルタ 14 コマンド	予約済み	フィルタ 13 コマンド	予約済み	フィルタ 12 コマンド
29	フィルタ 15 オフセット		フィルタ 14 オフセット		フィルタ 13 オフセット		フィルタ 12 オフセット	
30	フィルタ 13 CRC-16				フィルタ 12 CRC-16			
31	フィルタ 15 CRC-16				フィルタ 14 CRC-16			

表 542. ETH_MACRWKPFR

レジスタ	説明
フィルタ *i* バイトマスク	フィルタ *i* バイトマスクは、パケットがウェイクアップパケットであるかどうかを判定するために、フィルタ *i*（0、1、2、3、…、15）によって調べられるパケットのバイトを定義します。MSB（31 番目のビット）は 0 でなければなりません。 ビット j[30:0] はバイトマスクです。バイトマスクのビット j（バイト数）がセットされた場合、CRC ブロックが着信パケットのフィルタ *i* オフセット + j を処理します。そうでない場合、フィルタ *i* オフセット + j は無視されます。

表 542. ETH_MACRWKPFR（続き）

レジスタ	説明
フィルタ *i* コマンド	この 4 ビットのフィルタ *i* コマンドは、フィルタ *i* 動作を制御します。 – ビット 3 は、アドレスタイプを指定し、パターンのデスティネーションアドレスタイプを定義します。 このビットがセットされた場合、パターンがマルチキャストパケットにのみ適用され、このビットがリセットされた場合、パターンはユニキャストフレームにのみ適用されます。 – ビット 2（反転モード）がセットされると、CRC16 ハッシュ機能信号のロジックが反転し、一致する CRC_16 値を持つパケットを拒否します。 ビット 2 は、ビット 1 とともに、「パターン 1 AND NOT パターン 2」などのフィルタロジックを作成することで、MAC がリモートウェイクアップパケットのサブセットを拒否できます。 – ビット 1（And_Previous）は、ブール値ロジックを実装します。 セットされると、現在の入力の結果が、前のフィルタの結果と論理積されます。AND ロジックは、2、3、または 4 のフィルタの中のマスクを分割することで、32 バイトを超えるフィルタパターンを許可します。これは、And_Previous ビットをセットさせるフィルタ数に依存します。 And_Previous ビットは、フィルタの結果が前のフィルタの結果と論理積される場合、複数のフィルタ操作に適用できます。たとえば、And_Previous ビットがフィルタ 1 でセットされている場合、フィルタ 0 およびフィルタ 1 の両方が、*表 543*に示されたリモートウェイクアップパケット検出および割り込み生成基準を満たしている場合のみ、リモートウェイクアップパケットが検出され、PMT 割り込みが生成されます。 ***注：*** ***And_Previous ビットの設定は、4 フィルタのセット内で適用できます。*** ***注：*** ***有効になっていないフィルタの And_Previous ビットをセットしても効果はありません。つまり、4 フィルタのセットで最小数のフィルタの And_Previous ビットをセットしても効果はありません。たとえば、フィルタ 0 の And_Previous ビットをセットしても効果はありません。*** ***注：*** ***And_Previous ビットが論理積で結ばれたフィルタを形成するためにフィルタにセットされている場合、論理積のつながりはフィルタが有効でなくなった時点で破断されます。例：フィルタ 2 の And_Previous ビットがセット（フィルタ 2 コマンドのビット 1 がセット）されているが、フィルタ 1 が有効ではない（フィルタ 1 コマンドのビット 0 がリセットされている）場合、フィルタ 2 の結果だけが考慮されます。*** ***フィルタ 2 の And_Previous ビットがセット（フィルタ 2 コマンドのビット 1 がセット）され、フィルタ 3 の And_Previous ビットがセット（フィルタ 3 コマンドのビット 1 がセット）されているが、フィルタ 1 が有効ではない（フィルタ 1 コマンドのビット 0 がリセットされている）場合、フィルタ 3 の結果と論理積されたフィルタ 2 の結果だけが考慮されます。*** ***フィルタ 2 の And_Previous ビットがセット（フィルタ 2 コマンドのビット 1 がセット）され、フィルタ 3 の And_Previous ビットがセット（フィルタ 3 コマンドのビット 1 がセット）されているが、フィルタ 2 が有効ではない（フィルタ 2 コマンドのビット 0 がリセットされている）場合、フィルタ 2 の And_Previous ビットのセットによる効果がないため、フィルタ 3 の結果と論理和されたフィルタ 1 の結果だけが考慮されます。*** ***注：*** ***And_Previous ビットのセットによって結ばれたフィルタに、相補プログラミングが施されている場合、フレームが論理積で結ばれたフィルタを通過することはありません。たとえば、フィルタ 2 の And_Previous ビットがセット（フィルタ 2 コマンドのビット 1 がセット）され、マルチキャスト検出を示すフィルタ 1 の Address_Type ビットがセット（フィルタ 1 コマンドのビット 3 がセット）され、ユニキャスト検出を示すフィルタ 2 の Address_Type ビットがリセットされている（フィルタ 2 コマンドのビット 3 がリセットされている）場合、またはその逆の場合、リモートウェイクアップフレームがユニキャストおよびマルチキャストアドレスタイプの両方になりえないため、リモートウェイクアップフレームは論理積で結ばれたフィルタを通過しません。*** – ビット 0 はフィルタ *i* のイネーブルビットです。ビット 0 がセットされていない場合、フィルタ *i* は無効です。

表 542. ETH_MACRWKPFR（続き）

レジスタ	説明
フィルタ *i* オフセット	このフィルタ *i* オフセットレジスタは、フィルタ *i* がパケットを調べる（パケット内の）オフセットを定義します。 この 8 ビットパターンのオフセットは、フィルタ *i* によって調べられる最初のバイトのオフセットです。許される最小オフセット値は 12 であり、パケットの 13 番目のバイトを指します。オフセット値 0 は、パケットの最初のバイトを指します。
フィルタ *i* CRC-16	このフィルタ *i* CRC-16 レジスタは、パターンから計算された CRC_16 値と、ウェイクアップフィルタレジスタブロックに対してプログラムされたバイトマスクも含みます。 16 ビット CRC 計算には、次の多項式が使用されます。 G(x) = x^16 + x^15 + x^2 + 1 ハッシュ機能計算で使用される各マスクは、そのマスクに関連付けられた 16 ビット値で比較されます。各フィルタには以下の機能があります。 – 32 ビットマスク：このマスクの各ビットは、検出されたパケットの 1 バイトに対応します。ビットが ‘1’ である場合、対応するバイトが CRC16 の計算に取り入れられます。 – 8 ビットオフセットポインタ：CRC16 の計算を開始するバイトを指定します。 ポインタおよびマスクは、ともに CRC16 の計算で使用されるバイトの配置に使用されます。

次の表は、PMT 割り込みが生成されるリモートウェイクアップのシナリオを示しています。

表 543. リモートウェイクアップパケットおよび PMT 割り込みの生成(1)

フィルタ i コマンド			フレームタイプおよび CRC ステータス		割り込みの生成
CAST	INV	EN	受信フレームのキャストタイプ	CRC ステータス	RWK INTR
0	0	1	ユニキャスト	MATCH	リモートウェイクアップパケットが検出され、PMT 割り込みが生成されます。
0	1	1	ユニキャスト	MISMATCH	リモートウェイクアップパケットが検出され、PMT 割り込みが生成されます。
1	0	1	マルチキャスト	MATCH	リモートウェイクアップパケットが検出され、PMT 割り込みが生成されます。
1	1	1	マルチキャスト	MISMATCH	リモートウェイクアップパケットが検出され、PMT 割り込みが生成されます。

1. その他の組み合わせでは、リモートウェイクアップパケットは検出されず、PMT 割り込みは生成されません。

LPI 制御ステータスレジスタ（ETH_MACLCSR）

アドレスオフセット：0x00D0

リセット値：0x0000 0000

LPI 制御およびステータスレジスタは、LPI 機能を制御し、LPI 割り込みステータスを提供します。このレジスタが読み出されると、このステータスビットはクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPITCSE	LPITE	LPITXA	PLSEN	PLS	LPIEN
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	RLPIST	TLPIST	Res.	Res.	Res.	Res.	RLPIEX	RLPIEN	TLPIEX	TLPIEN
						r	r					r	r	r	r

ビット 31:22　予約済みであり、リセット値に保持する必要があります。

ビット 21 **LPITCSE**：LPI Tx クロック停止の有効化

このビットがセットされた場合、MAC は Tx LPI モードに移行した後で sbd_tx_clk_gating_ctrl_o 信号のハイをアサートして、MAC への Tx クロックを停止できることを示します。

このビットがリセットされた場合、MAC は Tx LPI モードに移行した後で sbd_tx_clk_gating_ctrl_o 信号のハイをアサートしません。

ビット 20 **LPITE**：LPI タイマ有効化

このビットは、MAC トランスミッタの LPI 状態への自動移行および自動終了を制御します。LPITE、LPITXA および LPITXEN ビットがセットされた場合、完全な MAC TX データパスが ETH_MACLETR レジスタで示された期間、アイドル状態であるときのみ、MAC トランスミッタが LPI 状態に移行します。

LPI 状態に移行後、データパスが（送信する新しいパケットを受け入れたために）アイドル状態ではなくなった場合、トランスミッタは LPI 状態を終了しますが、LPITXEN ビットをクリアしません。これにより、再びアイドル状態になったときに LPI 状態に再移行できます。

LPITE が 0 の場合、LPI 自動タイマは無効になり、MAC トランスミッタが LPITXA および LPITXEN ビットの説明にある設定に基づいて LPI 状態に移行します。

ビット 19 **LPITXA**：LPI Tx 自動化

このビットは、送信側で LPI モードの移行または終了を行う場合に MAC の動作を制御します。

LPITXA および LPIEN ビットが 1 にセットされた場合、MACは（コア内の）すべての未処理なパケットと（アプリケーションインタフェースでの）ペンディングパケットが送信された後にのみ、LPI モードに移行しますアプリケーションが送信用のパケットを送信する場合や Tx FIFO 一掃コマンドを発行する場合に、MAC は LPI モードを終了します。さらに、MAC は LPI 状態を終了するとき、LPIEN ビットを自動的にクリアします。Tx FIFO 一掃が ETH_MTLTxQOMR の FTQ ビットでセットされた場合、MAC が LPI モードであるときは LPI モードを終了します。

このビットが 0 である場合、LPIEN ビットが LPI モードの移行時または終了時の MAC の動作を直接制御します。

ビット 18 **PLSEN**：PHY リンクステータス有効化

このビットは、受信パスで受信されるリンクステータスを LPI LS TIMER の有効化に使用できるようにします。

このビットがセットされた場合、MAC は PLS ビットを LPI LS タイマのトリガに使用します。このビットがリセットされた場合、MAC は ETH_MACPHYCSR レジスタのリンクステータスビットを無視して、PLS ビットのみを取得します。

このビットは、PHY インタフェースを選択していない場合、読み出し専用かつ予約済みとなります。

ビット 17 **PLS**：PHY リンクステータス

このビットは、PHY のリンクステータスを示します。リンクステータスが少なくとも LPI LS TIMER で示された時間、アップ（OKAY）であるときのみ、MAC トランスミッタは LPI パターンをアサートします。このビットがセットされた場合、リンクは問題なし（UP）とみなされ、このビットがリセットされた場合、リンクがダウンしているとみなされます。

ビット 16 **LPIEN**：LPI 有効化

このビットがセットされた場合、MAC トランスミッタに LPI 状態に移行するよう命令します。このビットがリセットされた場合、MAC に LPI 状態を終了して通常の送信を再開するよう命令します。

LPITXA ビットがセットされ、MAC が送信用の新しいパケットを受け取ったために LPI 状態を終了するときに、このビットはクリアされます。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **RLPIST**：受信 LPI の状態

このビットがセットされた場合、MAC が MII インタフェースで LPI パターンを受信していることを示します。

ビット 8 **TLPIST**：送信 LPI の状態

このビットがセットされた場合、MAC が MII インタフェースで LPI パターンを送信していることを示します。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RLPIEX**：受信 LPI の終了

このビットがセットされた場合、MAC レシーバが MII インタフェースで LPI パターンの受信をやめ、LPI 状態を終了し、通常の受信を再開したことを示します。このビットは、このレジスタの読み出しによってクリアされます。

注： MAC が、CSR クロックの 3 クロックサイクル未満といったような非常に短い時間、LPI パターンの受信をやめる場合、このビットはセットできません。

ビット 2 **RLPIEN**：受信 LPI の移行

このビットがセットされた場合、MAC レシーバが LPI パターンを受信して LPI 状態に移行したことを示します。このビットは、このレジスタの読み出しによってクリアされます。

注： MAC が、CSR クロックの 3 クロックサイクル未満といったような非常に短い時間、LPI パターンの受信をやめる場合、このビットはセットできません。

ビット 1 **TLPIEX**：送信 LPI の終了

このビットがセットされた場合、アプリケーションが LPIEN ビットをクリアして LPI TW タイマが終了した後に、MAC トランスミッタが LPI 状態を終了したことを示します。このビットは、このレジスタの読み出しによってクリアされます。

ビット 0 **TLPIEN**：送信 LPI の移行

このビットがセットされた場合、LPIEN ビットの設定のため、MAC トランスミッタが LPI 状態に移行したことを示します。このビットは、このレジスタの読み出しによってクリアされます。

LPI タイマ制御レジスタ（ETH_MACLTCR）

アドレスオフセット：0x00D4

リセット値：0x03E8 0000

LPI タイマ制御レジスタは、LPI 状態でのタイムアウト値を制御します。MAC が LPI パターンを送信する時間と、通常の送信を再開するまで MAC が待機する時間も指定します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LST[9:0]									
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TWT[15:0]															
rw															

ビット 31:26　予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **LST[9:0]**：LPI LS タイマ

このフィールドは、LPI パターンを PHY に送信できるようになる前に、PHY からのリンクステータスをアップ（OKAY）にする必要がある最低時間（ミリ秒単位）を指定します。LPI LS タイマがプログラムされたターミナルカウントに達していない限り、MAC は LPIEN ビットがセットされている場合でも LPI パターンを送信しません。LPI LS タイマのデフォルト値は、IEEE 標準で定義された 1000（1 秒）です。

ビット 15:0 **TWT[15:0]**：LPI TW タイマ

このフィールドは、MAC が LPI パターンを PHY に送信した後から、通常の送信を再開するまで MAC が待機する最低時間（マイクロ秒単位）を指定します。TLPIEX ステータスビットは、このタイマが終了した後にセットされます。

LPI 移行タイマレジスタ（ETH_MACLETR）

アドレスオフセット：0x00D8

リセット値：0x0000 0000

LPI 移行タイマレジスタは、LPI アイドルタイマ値をマイクロ秒単位で格納するために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPIET[16:13]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPIET[12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット 31:20　予約済みであり、リセット値に保持する必要があります。

ビット 19:3 **LPIET[16:0]**：LPI 移行タイマ

このフィールドは、MAC がすべてのフレームを送信した後から、LPI モードに移行するまで MAC が待機する時間をマイクロ秒単位で指定します。このフィールドは、LPITE および LPITXA が 1 にセットされたときにのみ有効です。

ビット [2:0] は、このタイマのステップが 8 マイクロ秒単位となる読み出し専用です。

ビット 2:0　予約済みであり、リセット値に保持する必要があります。

1 マイクロ秒ティックカウンタレジスタ（ETH_MAC1USTCR）

アドレスオフセット：0x00DC

リセット値：0x0000 0000

このレジスタは、すべての LPI タイマの基準時間（1 マイクロ秒ティック）の生成を制御します。このタイマは最初にソフトウェアでプログラムする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TIC_1US_CNTR[11:0]											
				rw											

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:0　**TIC_1US_CNTR[11:0]**：1 µs ティックカウンタ

アプリケーションは、CSR クロックのクロックサイクル数が 1 µs となるように、このカウンタをプログラムする必要があります。
（プログラムする前の値から 1 を差し引きます）。
たとえば、CSR クロックが 100 MHz の場合、このフィールドを 100 - 1 = 99（この場合 0x63）にプログラムする必要があります。
これは、一部の EEE 関連カウンタを更新するために使用される 1 µs イベントを生成するために必要です。

バージョンレジスタ（ETH_MACVR）

アドレスオフセット：0x0110

リセット値：0x0000 3041

バージョンレジスタは Ethernet ペリフェラルのバージョンを特定します。このレジスタには、Synopsys がコアのリリース番号の特定に使用するバイトと、コアを設定している際にアプリケーションがセットできるバイトの 2 種類が含まれます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USERVER[7:0]								SNPSVER[7:0]							
r								r							

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:8　**USERVER[7:0]**：ユーザ定義バージョン（coreConsultant で設定）

ビット 7:0　**SNPSVER[7:0]**：Synopsys 定義バージョン（3.7）

デバッグレジスタ（ETH_MACDR）

アドレスオフセット：0x0114

リセット値：0x0000 0000

デバッグレジスタは、さまざまな MAC ブロックのデバッグステータスを提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFCSTS[1:0]		TPESTS
													r		r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFCFCSTS[1:0]		RPESTS
													r		r

ビット 31:19　予約済みであり、リセット値に保持する必要があります。

ビット 18:17　**TFCSTS[1:0]**：MAC 送信パケットコントローラステータス

このフィールドは、MAC 送信パケットコントローラモジュールの状態を示します。

00：アイドル状態

01：次のいずれかを待ちます。

– 前のパケットのステータス

– IPG またはバックオフ時間の経過

10：ポーズ制御パケット（全二重モード）の生成と送信

11：送信用に入力パケットを転送

ビット 16　**TPESTS**：MAC MII 送信プロトコルエンジンステータス

このビットがセットされた場合、MAC MII 送信プロトコルエンジンがアクティブにデータを送信しているため、アイドル状態にないことを示します。

ビット 15:3　予約済みであり、リセット値に保持する必要があります。

ビット 2:1　**RFCFCSTS[1:0]**：MAC 受信パケットコントローラ FIFO ステータス

このビットがセットされた場合、このフィールドは、MAC 受信パケットコントローラモジュールのスモール FIFO 読み出しおよび書き込みコントローラのアクティブ状態を示します。

ビット 0　**RPESTS**：MAC MII 受信プロトコルエンジンステータス

このビットがセットされた場合、MAC MII 受信プロトコルエンジンがアクティブにデータを受信しているため、アイドル状態にないことを示します。

HW 機能 1 レジスタ（ETH_MACHWF1R）

アドレスオフセット：0x0120

リセット値：0x1184 1904

このレジスタは、Ethernet ペリフェラルのオプションの特徴や機能の 2 番目のセットの存在を示します。ソフトウェアドライバは、このレジスタを使用して、オプションのブロックに関連するプログラムの有効化や無効化を動的に行えます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	L3L4FNUM[3:0]				Res.	HASHTBLSZ[1:0]		Res.	Res.	Res.	AVSEL	DBGMEMA	TSOEN	SPHEN	DCBEN
	r					r					r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR64[1:0]		ADVTHWORD	PTOEN	OSTEN	TXFIFOSIZE[4:0]					Res.	RXFIFOSIZE[4:0]				
r		r	r	r	r						r				

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:27 **L3L4FNUM[3:0]**：L3 または L4 フィルタの総数

このフィールドは、L3 または L4 フィルタの総数を示します。
0000：L3 または L4 フィルタなし
0001：L3 または L4 フィルタ 1 個
0010：L3 または L4 フィルタ 2 個
..
1000：L3 または L4 フィルタ 8 個

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **HASHTBLSZ[1:0]**：ハッシュテーブルサイズ

このフィールドは、ハッシュテーブルのサイズを示します。
00：ハッシュテーブルなし
01：64
10：128
11：256

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **AVSEL**：AV 機能有効化

オーディオビデオブリッジングの有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 19 **DBGMEMA**：DMA デバッグレジスタ有効化

デバッグモードの有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 18 **TSOEN**：TCP セグメンテーションオフロード有効化

TCP/IP パケットの TCP セグメンテーションオフロード有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 17 **SPHEN**：スプリットヘッダ機能有効化

スプリットヘッダ構造の有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 16 **DCBEN**：DCB 機能有効化
データセンターブリッジングの有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ADVTHWORD**：IEEE 1588 上位ワードレジスタ有効化
IEEE 1588 上位ワードレジスタの有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 12 **PTOEN**：PTP オフロード有効化
PTP タイムスタンプオフロード機能の有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 11 **OSTEN**：ワンステップタイムスタンプ有効化
ワンステップタイムスタンプ機能の有効化オプションが選択されている場合、このビットは 1 にセットされます。

ビット 10:6 **TXFIFOSIZE[4:0]**：MTL 送信 FIFO サイズ
このフィールドには、底を 2 とする対数から 7 を引いてバイト単位で表される MTL Tx FIFO の設定された値、つまり Log2(TXFIFO_SIZE) -7 が含まれます。
00000：128 バイト
00001：256 バイト
00010：512 バイト
00011：1,024 バイト
00100：2,048 バイト
00101：4,096 バイト
00110：8,192 バイト
00111：16,384 バイト
01000：32 KB
01001：64 KB
01010：128 KB
01011-11111：予約済み

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **RXFIFOSIZE[4:0]**：MTL 受信 FIFO サイズ
このフィールドには、底を 2 とする対数から 7 を引いてバイト単位で表される MTL Rx FIFO の設定された値、つまり Log2(RXFIFO_SIZE) -7 が含まれます。
00000：128 バイト
00001：256 バイト
00010：512 バイト
00011：1,024 バイト
00100：2,048 バイト
00101：4,096 バイト
00110：8,192 バイト
00111：16,384 バイト
01000：32,767 バイト
01000：32 KB
01001：64 KB
01010：128 KB
01011：256 KB
01100-11111：予約済み

HW 機能 2 レジスタ（ETH_MACHWF2R）

アドレスオフセット：0x0124

リセット値：0x4100 0000

このレジスタは、Ethernet ペリフェラルのオプションの特徴や機能の 3 番目のセットの存在を示します。ソフトウェアドライバは、このレジスタを使用して、オプションのブロックに関連するプログラムの有効化や無効化を動的に行えます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AUXSNAPNUM[2:0]			Res.	PPSOUTNUM[2:0]			Res.	Res.	TXCHCNT[3:0]				Res.	Res.
	r				r					r					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCHCNT[3:0]				Res.	Res.	TXQCNT[3:0]				Res.	Res.	RXQCNT[3:0]			
r						r						r			

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30:28　**AUXSNAPNUM[2:0]**：補助スナップショット入力数

このフィールドは、補助スナップショット入力数を示します。

000：補助入力なし

001：補助入力 1 個

010：補助入力 2 個

011：補助入力 3 個

100：補助入力 4 個

101-111：予約済み

ビット 27　予約済みであり、リセット値に保持する必要があります。

ビット 26:24　**PPSOUTNUM[2:0]**：PPS 出力数

このフィールドは、PPS 出力数を示します。

000：PPS 出力なし

001：PPS 出力 1 個

010：PPS 出力 2 個

011：PPS 出力 3 個

100：PPS 出力 4 個

101-111：予約済み

ビット 23:22　予約済みであり、リセット値に保持する必要があります。

ビット 21:18　**TXCHCNT[3:0]**：DMA 送信チャネル数

このフィールドは、DMA 送信チャネル数を示します。

0000：DMA Tx チャネル 1 個

0001：DMA Tx チャネル 2 個

..

0111：DMA Tx チャネル 8 個

ビット 17:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **RXCHCNT[3:0]**：DMA 受信チャネル数
このフィールドは、DMA 受信チャネル数を示します。
0000：DMA Rx チャネル 1 個
0001：DMA Rx チャネル 2 個
..
0111：DMA Rx チャネル 8 個

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:6 **TXQCNT[3:0]**：MTL 送信キュー数
このフィールドは、MTL 送信キュー数を示します。
0000：MTL Tx キュー 1 個
0001：MTL Tx キュー 2 個
..
0111：MTL Tx キュー 8 個

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **RXQCNT[3:0]**：MTL 受信キュー数
このフィールドは、MTL 受信キュー数を示します。
0000：MTL Rx キュー 1 個
0001：MTL Rx キュー 2 個
..
0111：MTL Rx キュー 8 個

MDIO アドレスレジスタ（ETH_MACMDIOAR）

アドレスオフセット：0x0200

リセット値：0x0000 0000

MDIO アドレスレジスタは、管理インタフェースを通じて外部 PHY の管理サイクルを制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	PSE	BTB	PA[4:0]					RDA[4:0]				
				rw	rw	rw					rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NTC[2:0]			CR[3:0]				Res.	Res.	Res.	SKAP	GOC_1	GOC_0	C45E	MB
	rw			rw							rw	rw	rw	rw	rw

ビット 31:28　予約済みであり、リセット値に保持する必要があります。

ビット 27 **PSE**：プリアンブル抑制有効化

このビットがセットされた場合、SMA は 32 ビットプリアンブルおよび 1 プリアンブルビットだけを持つ送信 MDIO フレームを抑制します。

このビットが 0 である場合、MDIO フレームには必ず IEEE 標準で定義された 32 ビットのプリアンブルがあります。

ビット 26 **BTB**：連続トランザクション

このビットがセットされ、NTC に 0 を超える値がある場合、MAC はフレーム転送の最後（立ち下がりクロックが送信される前）に読み出しまたは書き込みコマンドの完了を通知します。このように、ソフトウェアは、前のフレームに生成された立ち下がりクロックの数に関係なく、ただちに実行される次のコマンドを開始できます。

このビットがリセットされた場合、立ち下がりクロック後でのみ読み出し／書き込みコマンド完了（MII ビジーがクリア）が生成されます。このモードでは、各フレーム後に NTC が必ず生成されることを確認します。

NTC=0 の場合、このビットはセットしないでください。

ビット 25:21 **PA[4:0]**：物理層アドレス

このフィールドは、MAC がどの Clause 22 PHY デバイス（32 デバイスのうち）にアクセスしているかを示します。このフィールドは、MAC がどの Clause 45 対応 PHY（32 個の PHY のうち）にアクセスしているかを示します。

ビット 20:16 **RDA[4:0]**：レジスタ／デバイスアドレス

これらのビットは、選択された Clause 22 PHY デバイスで PHY レジスタを選択します。これらのビットは、選択された Clause 45 対応 PHY でデバイス（MMD）を選択します。

ビット 15　予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **NTC[2:0]**：立ち下がりクロックの数

このフィールドは、MDIO フレームの送信終了後に ETH_MDC で生成される立ち下がりクロックサイクル数を制御します。有効な値は 0～7 です。値を 3'h3 にプログラムすることで、MDIO フレームの転送終了後に MDC ラインでさらに 3 つのクロックサイクルがあることを示します。

ビット 11:8 **CR[3:0]**：CSR クロック範囲

CSR クロック範囲選択は、設計に使用される CSR クロック周波数に従って MDC クロックの周波数を決めます。

0000：CSR クロック = 60〜100 MHz、MDC クロック = CSR クロック/42

0001：CSR クロック = 100〜150 MHz、MDC クロック = CSR クロック/62

0010：CSR クロック = 20〜35 MHz、MDC クロック = CSR クロック/16

0011：CSR クロック = 35〜60 MHz、MDC クロック = CSR クロック/26

0100：CSR クロック = 150〜250 MHz、MDC クロック = CSR クロック/102

0101：CSR クロック = 250〜300 MHz、MDC クロック = CSR クロック/124

0110, 0111：予約済み

各値に適用可能な CSR クロック周波数の推奨範囲（ビット 11 = 0 のとき）は、MDC クロックが約 1.0〜2.5 MHz の周波数の範囲になることを保証します。

ビット 11 がセットされた場合、2.5 MHz の周波数制限（IEEE 802.3 で指定）を上回る MDC クロックの周波数を実現し、より低い値のクロック分周回路をプログラムできます。たとえば、CSR クロックの周波数が 100 MHz で、これらのビットを 1010 でプログラムする場合、その結果の MDC クロックは IEEE 802.3 で指定された範囲を超える 12.5 MHz となります。接続しているチップがより高速な MDC クロックをサポートしている場合のみ、次の値をプログラムします。

1000：CSR クロック/4

1001：CSR クロック/6

1010：CSR クロック/8

1011：CSR クロック/10

1100：CSR クロック/12

1101：CSR クロック/14

1110：CSR クロック/16

1111：CSR クロック/18

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SKAP**：スキップアドレスパケット

このビットがセットされると、SMA は読み出し、書き込み、または事後読み出しインクリメントアドレスパケットの前にアドレスパケットを送信しません。このビットは、C45E がセットされている場合のみ有効です。

ビット 3:2 **GOC**：MII 動作コマンド

このビットは、PHY の動作コマンドを示します。

00：予約済み

01：書き込み

10：Clause 45 PHY に対する事後読み出しインクリメントアドレス

11：読み出し

Clause 22 PHY が有効である場合、書き込みおよび読み出しコマンドは有効です。

ビット 1 **C45E**：Clause 45 PHY 有効化

このビットがセットされた場合、Clause 45 対応 PHY が MDIO に接続されます。このビットがリセットされた場合、Clause 22 対応 PHY が MDIO に接続されます。

ビット 0 **MB**：MII ビジー

アプリケーションはこのビットをセットして SMA に MDIO の読み出しまたは書き込み
アクセスを開始するよう命令します。MAC は、MDIO フレーム転送が完了した後に
このビットをクリアします。そのため、ソフトウェアは、このビットがセットされている限り、ETH_MACMDIOAR および ETH_MACMDIODR のフィールドへの書き込みや変更を行ってはなりません。

書き込み転送の場合、アプリケーションはまず ETH_MACMDIODR レジスタの MD フィールド（C45E がセットされている場合は RA フィールドも含む）に 16 ビットデータを書き込んでから
このビットをセットする必要があります。C45E がセットされている場合、ETH_MACMDIODR レジスタの RA フィールドにも書き込んでから
読み出し転送を開始する必要があります。読み出し
転送が完了が完了したとき（MII busy=0）、PHY レジスタからのデータ読み出しは
ETH_MACMDIODR レジスタの MD フィールドで有効です。

注： アドレス指定された PHY が存在しない場合でも、このビットの機能に変化はありません。

MDIO データレジスタ（ETH_MACMDIODR）

アドレスオフセット：0x0204

リセット値：0x0000 0000

MDIO データレジスタは、ETH_MACMDIOAR で指定されたアドレスにある PHY レジスタに書き込むデータを格納します。このレジスタは、MDIO アドレスレジスタによって指定されたアドレスにある PHY レジスタから読み出されたデータも格納します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RA[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MD[15:0]															
rw															

ビット 31:16 **RA[15:0]**：レジスタアドレス

このフィールドは、C45E がセットされている場合のみ有効です。これは、MDIO フレームが対象としている PHY のレジスタアドレスを含みます。

ビット 15:0 **MD[15:0]**：MII データ

このフィールドは、管理読み出し動作の後で PHY から読み出された 16 ビットのデータ、または管理書き込み動作の前に PHY に書き込まれる 16 ビットデータを含みます。

ARP アドレスレジスタ（ETH_MACARPAR）

アドレスオフセット：0x0AE0

リセット値：0x0000 0000

ARP アドレスレジスタは、MAC の IPv4 デスティネーションアドレスを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARPPA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARPPA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ARPPA[31:0]**：ARP プロトコルアドレス

このフィールドは、MAC の IPv4 デスティネーションアドレスを含みます。このアドレスは、受信した ARP パケットにあるターゲットフィールドのプロトコルアドレスとの完全一致に使用されます。

このフィールドは、IPv4 ARP オフロードの有効化オプションが選択されている場合のみ使用できます。

アドレス 0 上位レジスタ（ETH_MACA0HR）

アドレスオフセット：0x0300

リセット値：0x8000 FFFF

MAC アドレス 0 上位レジスタは、ステーションの最初の MAC アドレス 6 バイトのうち上位 16 ビットを保持します。MII インタフェース上で受信された最初の DA バイトは、MAC アドレス下位レジスタの LS バイト（ビット [7:0]）に対応します。たとえば、MII 上でデスティネーションアドレスとして 0x112233445566 が受信された場合（最初の列のレーン 0 で 0x11）、MacAddress0 レジスタ [47:0] は 0x665544332211 と比較されます。

MAC アドレスレジスタが、MII クロックドメインと二重同期するよう設定された場合、MAC Address0 下位レジスタのビット [31:24]（リトルエンディアンモード）またはビット [7:0]（ビッグエンディアンモード）が書き込まれたときのみ、同期がトリガされます。適切に同期更新を行うために、このアドレス下位レジスタへの連続書き込みは、デスティネーションクロックドメインの少なくとも 4 クロックサイクル後に実行されなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRHI[15:0]															
rw															

ビット 31 **AE**：アドレス有効化

このビットは常に 1 にセットされます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ADDRHI[15:0]**：MAC Address0[47:32]

このフィールドは、最初の MAC アドレス 6 バイトのうち上位 16 ビット [47:32] を含みます。MAC は、受信パケットのフィルタのためと、送信フロー制御（ポーズ）パケットに MAC アドレスを挿入するために、このフィールドを使用します。

アドレス x 下位レジスタ（ETH_MACAxLR）

アドレスオフセット：0x0304 + x*0x8（x = 0～3）

リセット値：0xFFFF FFFF

MAC アドレス x 下位レジスタは、ステーションの最初の MAC アドレス 6 バイトのうち下位 32 ビットを保持します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDRLO[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRLO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ADDRLO[31:0]**：MAC アドレス x [31:0]（x = 0～3）

このフィールドは、最初の MAC アドレス 6 バイトのうち下位 32 ビットを含みます。MAC は、受信パケットのフィルタのためと、送信フロー制御（ポーズ）パケットに MAC アドレスを挿入するために、このフィールドを使用します。

アドレス x 上位レジスタ（ETH_MACAxHR）

アドレスオフセット：0x0308 + (x-1)*0x8（x = 1〜3）

リセット値：0x0000 FFFF

MAC アドレス x 上位レジスタは、ステーションの 2 番目の MAC アドレス 6 バイトのうち上位 16 ビットを保持します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AE	SA	MBC[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRHI[15:0]															
rw															

ビット 31 **AE**：アドレス有効化

このビットがセットされると、アドレスフィルタモジュールは、完全フィルタに 2 番目の MAC アドレスを使用します。このビットがリセットされると、アドレスフィルタモジュールはフィルタについてアドレスを無視します。

ビット 30 **SA**：ソースアドレス

このビットがセットされると、受信パケットの SA フィールドとの比較に MAC Address1[47:0] が使用されます。このビットがリセットされると、受信パケットの DA フィールドとの比較に MAC Address x[47:0] が使用されます。

ビット 29:24 **MBC[5:0]**：マスクバイト制御

これらのビットは、MAC アドレスの各バイトの比較のためのマスク制御ビットです。これらのビットがハイにセットされると、MAC は、受信した DA または SA の対応するバイトと MAC Address1 レジスタの内容とを比較しません。各ビットは、以下のようにバイトのマスキングを制御します。

ビット 29：レジスタ 194[15:8]

ビット 28：レジスタ 194[7:0]

ビット 27：レジスタ 195[31:24]

..

ビット 24：レジスタ 195[7:0]

アドレスの 1 つ以上のバイトをマスクすることで、アドレスのグループをフィルタできます（グループアドレスフィルタとして知られる）。

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ADDRHI[15:0]**：MAC Address1 [47:32]

このフィールドは、2 番目の MAC アドレス 6 バイトのうち上位 16 ビット [47:32] を含みます。

MMC 制御レジスタ（MMC_CONTROL）

アドレスオフセット：0x0700

リセット値：0x0000 0000

このレジスタは、MMC 動作モードを設定します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCDBC	Res.	Res.	CNTPRSTLVL	CNTPRST	CNTFREEZ	RSTONRD	CNTSTOPRO	CNTRST
							rw			rw	rw	rw	rw	rw	rw

ビット 31:9　予約済みであり、リセット値に保持する必要があります。

ビット 8　**UCDBC**：ドロップされたブロードバンドパケットの MMC カウンタの更新

CNTRST ビットは、CNTPRST ビットより優先されます。そのため、ソフトウェアが同じ書き込みサイクルで両方のビットをセットしようとすると、すべてのカウンタがクリアされ、CNTPRST ビットはセットされません。

セットされた場合、MAC は ETH_MACPFR レジスタの DBF ビットの設定によってドロップされたブロードバンドパケットの関連する MMC カウンタすべてを更新します。

リセットされた場合、MMC カウンタはドロップされたブロードバンドパケットについて更新されません。

ビット 7:6　予約済みであり、リセット値に保持する必要があります。

ビット 5　**CNTPRSTLVL**：フルハーフプリセット

このビットがローで CNTPRST ビットがセットされた場合、すべての MMC カウンタがほぼハーフ値にプリセットされます。また、すべてのオクテットカウンタが 0x7FFF_F800（ハーフ 2 KB）にプリセットされ、すべてパケットカウンタが 0x7FFF_FFF0（ハーフ 16）にプリセットされます。

このビットがハイで CNTPRST ビットがセットされた場合、すべての MMC カウンタがほぼフル値にプリセットされます。また、すべてのオクテットカウンタが 0xFFFF_F800（フル 2 KB）にプリセットされ、すべてパケットカウンタが 0xFFFF_FFF0（フル 16）にプリセットされます。

16 ビットカウンタの場合、ほぼハーフのプリセット値は、オクテットカウンタおよびパケットカウンタそれぞれに対して 0x7800 および 0x7FF0 です。同様に、16 ビットカウンタのほぼフルのプリセット値は、0xF800 および 0xFFF0 です。

ビット 4　**CNTPRST**：カウンタプリセット

このビットがセットされた場合、すべてのカウンタが初期化されるか、CNTPRSTLVL ビットに従ってほぼフルまたはほぼハーフにプリセットされます。このビットは、1 クロックサイクル後に自動的にクリアされます。

CNTPRSTLVL ビットとともに、このビットは MMC カウンタがハーフフルまたはフルになっているために発生する割り込みのアサーションのデバッグやテストに有用です。

ビット 3　**CNTFREEZ**：MMC カウンタ固定

このビットがセットされた場合、すべての MMC カウンタを現在の値で固定します。

このビットが 0 にリセットされた場合、MMC カウンタはパケットの送受信のために更新されます。MMC カウンタのいずれかが読み出し時リセットビットがセットされた状態で読み出された場合、そのカウンタはこのモードでクリアされます。

ビット 2 **RSTONRD**：読み出し時リセット

このビットがセットされた場合、MMC カウンタは、読み出し後に 0 にリセットされます（リセット後、自動クリア）。カウンタは、最下位バイトレーン（ビット [7:0]）が読み出されると、クリアされます。

ビット 1 **CNTSTOPRO**：カウンタストップロールオーバー

このビットがセットされた場合、カウンタは、最大値に達した後も 0 にロールオーバーしません。

ビット 0 **CNTRST**：カウンタリセット

このビットがセットされた場合、すべてのカウンタがリセットされます。このビットは、1 クロックサイクル後に自動的にクリアされます。

MMC Rx 割り込みレジスタ（MMC_RX_INTERRUPT）

アドレスオフセット：0x0704

リセット値：0x0000 0000

このレジスタは、すべての受信統計カウンタから生成された割り込みを保持します。

MMC 受信割り込みレジスタは、次のことが発生した際に生成される割り込みを保持します。

- 受信統計カウンタが、最大値の半分に達しています（32 ビットカウンタでは 0x8000_0000、16 ビットカウンタでは 0x8000）。
- 受信統計カウンタが、最大値を超えています（32 ビットカウンタでは 0xFFFF_FFFF、16 ビットカウンタでは 0xFFFF）。

カウンタストップロールオーバーがセットされた場合、割り込みがセットされますが、カウンタはすべて 1 のままとなります。MMC 受信割り込みレジスタは 32 ビットレジスタです。割り込みビットは、割り込みの原因となったそれぞれの MMC カウンタが読み出されたときにクリアされます。割り込みビットをクリアするには、それぞれのカウンタの最下位バイトレーン（ビット [7:0]）が読み出されなければなりません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXALGNERPIS	RXCRCERPIS	Res.	Res.	Res.	Res.	Res.
									r	r					

ビット 31:28　予約済みであり、リセット値に保持する必要があります。

ビット 27 **RXLPITRCIS**：MMC 受信 LPI 遷移カウンタ割り込みステータス

このビットは、Rx_LPI_Tran_Cntr カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 26 **RXLPIUSCIS**：MMC 受信 LPI マイクロ秒カウンタ割り込みステータス

このビットは、Rx_LPI_USEC_Cntr カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 25:18　予約済みであり、リセット値に保持する必要があります。

ビット 17 **RXUCGPIS**：MMC 受信ユニキャスト良好パケットカウンタ割り込みステータス

このビットは、rxunicastpackets_g カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 16:7　予約済みであり、リセット値に保持する必要があります。

ビット 6 **RXALGNERPIS**：MMC 受信整列エラーパケットカウンタ割り込みステータス
このビットは、rxalignmenterror カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 5 **RXCRCERPIS**：MMC 受信 CRC エラーパケットカウンタ割り込みステータス
このビットは、rxcrcerror カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

MMC Tx 割り込みレジスタ（MMC_TX_INTERRUPT）

アドレスオフセット：0x0708

リセット値：0x0000 0000

このレジスタは、すべての送信統計カウンタから生成された割り込みを保持します。

MMC 送信割り込みレジスタは、送信統計カウンタが最大値の半分（32 ビットカウンタでは 0x8000_0000、16 ビットカウンタでは 0x8000）に達した場合、および最大値（32 ビットカウンタでは 0xFFFF_FFFF、16 ビットカウンタでは 0xFFFF）を超えた場合に生成される割り込みを保持します。

カウンタストップロールオーバーがセットされた場合、割り込みがセットされますが、カウンタはすべて 1 のままとなります。

MMC 送信割り込みレジスタは 32 ビットレジスタです。割り込みビットは、割り込みの原因となったそれぞれの MMC カウンタが読み出されたときにクリアされます。

割り込みビットをクリアするには、それぞれのカウンタの最下位バイトレーン（ビット [7:0]）が読み出されなければなりません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXMCOLGPIS	TXSCOLGPIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r														

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXLPITRCIS**：MMC 送信 LPI 遷移カウンタ割り込みステータス
このビットは、Tx_LPI_Tran_Cntr カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 26 **TXLPIUSCIS**：MMC 送信 LPI マイクロ秒カウンタ割り込みステータス
このビットは、Tx_LPI_USEC_Cntr カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TXGPKTIS**：MMC 送信良好パケットカウンタ割り込みステータス
このビットは、txpacketcount_g カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 20:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TXMCOLGPIS**：MMC 送信複数衝突良好パケットカウンタ割り込みステータス
このビットは、txmulticol_g カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 14 **TXSCOLGPIS**：MMC 送信単一衝突良好パケットカウンタ割り込みステータス
このビットは、txsinglecol_g カウンタが最大値の半分または最大値に達したときにセットされます。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

MMC Rx 割り込みマスクレジスタ（MMC_RX_INTERRUPT_MASK）

アドレスオフセット：0x070C

リセット値：0x0000 0000

MMC 受信割り込みマスクレジスタは、受信統計カウンタが最大値の半分または最大値に達したときに生成される割り込みのマスクを保持します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	RXLPITRCIM	RXLPIUSCIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXUCGPIM	Res.
				r	rw									rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXALGNERPIM	RXCRCERPIM	Res.	Res.	Res.	Res.	Res.
									rw	rw					

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **RXLPITRCIM**：MMC 受信 LPI 遷移カウンタ割り込みマスク
このビットがセットされると、Rx_LPI_Tran_Cntr カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 26 **RXLPIUSCIM**：MMC 受信 LPI マイクロ秒カウンタ割り込みマスク
このビットがセットされると、Rx_LPI_USEC_Cntr カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 25:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **RXUCGPIM**：MMC 受信ユニキャスト良好パケットカウンタ割り込みマスク
このビットがセットされると、rxunicastpackets_g カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 16:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RXALGNERPIM**：MMC 受信整列エラーパケットカウンタ割り込みマスク
このビットがセットされると、rxalignmenterror カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 5 **RXCRCERPIM**：MMC 受信 CRC エラーパケットカウンタ割り込みマスク
このビットがセットされると、rxcrcerror カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

MMC Tx 割り込みマスクレジスタ（MMC_TX_INTERRUPT_MASK）

アドレスオフセット：0x0710

リセット値：0x0000 0000

このレジスタは、すべての送信統計カウンタから生成された割り込みのマスクを保持します。

MMC 送信割り込みマスクレジスタは、送信統計カウンタが最大値の半分または最大値に達したときに生成される割り込みのマスクを保持します。このレジスタは 32 ビット幅です。このレジスタは、MMC 送信カウンタのいずれかがコア設定中に選択された場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXLPITRCIM	TXLPIUSCIM	Res.	Res.	Res.	Res.	TXGPKTIM	Res.	Res.	Res.	Res.	Res.
				r	rw					rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXMCOLGPIM	TXSCOLGPIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:28　予約済みであり、リセット値に保持する必要があります。

ビット 27　**TXLPITRCIM**：MMC 送信 LPI 遷移カウンタ割り込みマスク
このビットがセットされると、Tx_LPI_Tran_Cntr カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 26　**TXLPIUSCIM**：MMC 送信 LPI マイクロ秒カウンタ割り込みマスク
このビットがセットされると、Tx_LPI_USEC_Cntr カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 25:21　予約済みであり、リセット値に保持する必要があります。

ビット 21　**TXGPKTIM**：MMC 送信良好パケットカウンタ割り込みマスク
このビットがセットされると、txpacketcount_g カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 20:16　**RESERVED_TXGOCTIM**：予約済み。

ビット 15　**TXMCOLGPIM**：MMC 送信複数衝突良好パケットカウンタ割り込みマスク
このビットがセットされると、txmulticol_g カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 14　**TXSCOLGPIM**：MMC 送信単一衝突良好パケットカウンタ割り込みマスク
このビットがセットされると、txsinglecol_g カウンタが最大値の半分または最大値に達したときに割り込みをマスクします。

ビット 13:0　予約済みであり、リセット値に保持する必要があります。

Tx 単一衝突良好パケットレジスタ（TX_SINGLE_COLLISION_GOOD_PACKETS）

アドレスオフセット：0x074C

リセット値：0x0000 0000

このレジスタは、半二重モードで 1 回の衝突後に Ethernet ペリフェラルによって正常に送信されたパケットの数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXSNGLCOLG[31:0]																															
r																															

ビット 31:0 **TXSNGLCOLG[31:0]**：Tx 単一衝突良好パケット
このフィールドは、半二重モードで 1 回の衝突後に正常に送信されたパケットの数を示します。

Tx 複数衝突良好パケットレジスタ（TX_MULTIPLE_COLLISION_GOOD_PACKETS）

アドレスオフセット：0x0750

リセット値：0x0000 0000

このレジスタは、半二重モードで複数回の衝突後に Ethernet ペリフェラルによって正常に送信されたパケットの数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXMULTCOLG[31:0]																															
r																															

ビット 31:0 **TXMULTCOLG[31:0]**：Tx 複数衝突良好パケット
このフィールドは、半二重モードで複数回の衝突後に正常に送信されたパケットの数を示します。

Tx パケットカウント良好レジスタ（TX_PACKET_COUNT_GOOD）

アドレスオフセット：0x0768

リセット値：0x0000 0000

このレジスタは、Ethernet ペリフェラルによって送信された良好フレームの数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXPKTG[31:0]																															
r																															

ビット 31:0 **TXPKTG[31:0]**：Tx パケットカウント良好
このフィールドは、送信された良好パケットの数を示します。

Rx CRC エラーパケットレジスタ（RX_CRC_ERROR_PACKETS）

アドレスオフセット：0x0794

リセット値：0x0000 0000

このレジスタは、CRC エラーがある Ethernet ペリフェラルによって受信されたパケットの数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCRCERR[31:0]																															
r																															

ビット 31:0 **RXCRCERR[31:0]**：Rx CRC エラーパケット
このフィールドは、CRC エラーがある受信パケットの数を示します。

Rx 整列エラーパケットレジスタ（RX_ALIGNMENT_ERROR_PACKETS）

アドレスオフセット：0x0798

リセット値：0x0000 0000

このレジスタは、整列（ドリブル）エラーがある Ethernet ペリフェラルによって受信されたパケットの数を提供します。このレジスタは 10/100 モードでのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXALGNERR[31:0]																															
r																															

ビット 31:0 **RXALGNERR[31:0]**：Rx 整列エラーパケット
このフィールドは、整列（ドリブル）エラーがある受信パケットの数を示します。このレジスタは 10/100 モードでのみ有効です。

Rx ユニキャストパケットカウント良好レジスタ（RX_UNICAST_PACKETS_GOOD）

アドレスオフセット：0x07C4

リセット値：0x0000 0000

このレジスタは、Ethernet ペリフェラルによって受信された良好ユニキャストパケットの数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXUCASTG[31:0]																															
r																															

ビット 31:0 **RXUCASTG[31:0]**：Rx ユニキャストパケット良好
このフィールドは、受信された良好ユニキャストパケットの数を示します。

Tx LPI マイクロ秒タイマレジスタ（TX_LPI_USEC_CNTR）

アドレスオフセット：0x07EC

リセット値：0x0000 0000

このレジスタは、Tx LPI が Ethernet ペリフェラルによってアサートされたマイクロ秒数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXLPIUSC[31:0]																															
r																															

ビット 31:0 **TXLPIUSC[31:0]**：Tx LPI マイクロ秒カウンタ

このフィールドは、Tx LPI がアサートされたマイクロ秒数を示します。それぞれの Tx LPI の移行および終了に対して、タイマ値に +/- 1 マイクロ秒の誤差を設けることができます。

Tx LPI 遷移カウンタレジスタ（TX_LPI_TRAN_CNTR）

アドレスオフセット：0x07F0

リセット値：0x0000 0000

このレジスタは、Ethernet ペリフェラルが Tx LPI に移行した回数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXLPITRC[31:0]																															
r																															

ビット 31:0 **TXLPITRC[31:0]**：Tx LPI 遷移カウンタ

このフィールドは、Tx LPI 移行が発生した回数を示します。Tx LPI 移行が（LPI 制御およびステータスレジスタでセットされた LPITXA ビットによって）自動化モードで発生しても、カウンタはインクリメントします。

Rx LPI マイクロ秒カウンタレジスタ（RX_LPI_USEC_CNTR）

アドレスオフセット：0x07F4

リセット値：0x0000 0000

このレジスタは、Rx LPI が Ethernet ペリフェラルによってサンプリングされたマイクロ秒数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXLPIUSC[31:0]																															
r																															

ビット 31:0 **RXLPIUSC[31:0]**：Rx LPI マイクロ秒カウンタ

このフィールドは、Rx LPI がアサートされたマイクロ秒数を示します。それぞれの Rx LPI の移行および終了に対して、タイマ値に +/- 1 マイクロ秒の誤差を設けることができます。

Rx LPI 遷移カウンタ レジスタ（RX_LPI_TRAN_CNTR）

アドレスオフセット：0x07F8

リセット値：0x0000 0000

このレジスタは、Ethernet ペリフェラルが Rx LPI に移行した回数を提供します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXLPITRC[31:0]																															
r																															

ビット 31:0 **RXLPITRC[31:0]**：Rx LPI 遷移カウンタ

このフィールドは、Rx LPI 移行が発生した回数を示します。

L3 および L4 制御 0 レジスタ（ETH_MACL3L4C0R）

アドレスオフセット：0x0900

リセット値：0x0000 0000

レイヤ 3 およびレイヤ 4 制御レジスタは、レイヤ 3 およびレイヤ 4 のフィルタ 0 の動作を制御します。このレジスタは、レイヤ 3 およびレイヤ 4 フィルタ機能がコア設定中に選択されていない場合は予約済みとなります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	L4DPIM0	L4DPM0	L4SPIM0	L4SPM0	Res.	L4PEN0
										rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3HDBM0[4:0]					L3HSBM0[4:0]					L3DAIM0	L3DAM0	L3SAIM0	L3SAM0	Res.	L3PEN0
rw					rw					rw	rw	rw	rw		rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **L4DPIM0**：レイヤ 4 デスティネーションポート反転一致の有効化

このビットがセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの完全一致が有効になります。

このビットは、L4DPM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 20 **L4DPM0**：レイヤ 4 デスティネーションポート一致の有効化

このビットがセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 4 デスティネーションポート番号フィールドの一致を無視します。

ビット 19 **L4SPIM0**：レイヤ 4 ソースポート反転一致の有効化

このビットがセットされた場合、レイヤ 4 ソースポート番号フィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 4 ソースポート番号フィールドの完全一致が有効になります。

このビットは、L4SPM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 18 **L4SPM0**：レイヤ 4 ソースポート一致の有効化

このビットがセットされた場合、レイヤ 4 ソースポート番号フィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 4 ソースポート番号フィールドの一致を無視します。

ビット 17　予約済みであり、リセット値に保持する必要があります。

ビット 16 **L4PEN0**：レイヤ 4 プロトコルの有効化

このビットがセットされた場合、UDP パケットのソースおよびデスティネーションポート番号が一致に使用されます。このビットがリセットされた場合、TCP パケットのソースおよびデスティネーションポート番号が一致に使用されます。

レイヤ 4 一致は、L4SPM0 または L4DPM0 ビットがセットされたときにのみ実行されます。

ビット 15:11 **L3HDBM0[4:0]**：レイヤ 3 IP DA 上位ビット一致

IPv4 パケット：

このフィールドは、IPv4 パケットで一致する IP デスティネーションアドレスの上位ビットの数を含みます。次のリストでは、このフィールドの値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

31：MSb を除くすべてのビットがマスクされます。

IPv6 パケット：

このフィールドのビット [12:11] は、IPv6 パケットでマスクされる IP ソースまたはデスティネーションアドレスの下位ビットの数を示す L3HSBM0 のビット [6:5] に対応します。次のリストでは、L3HDBM0[1:0] および L3HSBM0 ビットの連結された値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

127：MSb を除くすべてのビットがマスクされます。

このフィールドは、L3DAM0 または L3SAM0 ビットがセットされたときにのみ有効であり、適用されます。

ビット 10:6 **L3HSBM0[4:0]**：レイヤ 3 IP SA 上位ビット一致

IPv4 パケット：

このフィールドは、IPv4 パケットでの一致に対してマスクされる IP ソースアドレスの下位ビットの数を含みます。次のリストでは、このフィールドの値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

31：MSb を除くすべてのビットがマスクされます。

IPv6 パケット：

このフィールドは、L3HSBM0 のビット [4:0] を含みます。これらのビットは、IPv6 パケットで一致する IP ソースまたはデスティネーションアドレスの上位ビットの数を示します。このフィールドは、L3DAM0 または L3SAM0 ビットがハイにセットされたときにのみ有効であり、適用されます。

ビット 5 **L3DAIM0**：レイヤ 3 IP DA 反転一致の有効化

このビットがセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの完全一致が有効になります。

このビットは、L3DAM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 4 **L3DAM0**：レイヤ 3 IP DA 一致の有効化

このビットがセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 3 IP デスティネーションアドレスフィールドの一致を無視します。

注：　L3PEN0 ビットがセットされた場合、IPv6 DA または SA がフィルタでチェックできるため、このビットか L3SAM0 ビットをセットする必要があります。

ビット 3 **L3SAIM0**：レイヤ 3 IP SA 反転一致の有効化

このビットがセットされた場合、レイヤ 3 IP ソースアドレスフィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 3 IP ソースアドレスフィールドの完全一致が有効になります。

このビットは、L3SAM0 ビットがセットされた場合にのみ有効であり、適用されます。

ビット 2 **L3SAM0**：レイヤ 3 IP SA 一致の有効化

このビットがセットされた場合、レイヤ 3 IP ソースアドレスフィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 3 IP ソースアドレスフィールドの一致を無視します。

注： L3PEN0 ビットがセットされた場合、IPv6 SA または DA がフィルタでチェックできるため、このビットか L3DAM0 ビットをセットする必要があります。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **L3PEN0**：レイヤ 3 プロトコルの有効化

このビットがセットされた場合、レイヤ 3 IP ソースまたはデスティネーションアドレスの一致が IPv6 パケットに対して有効になります。このビットがリセットされた場合、レイヤ 3 IP ソースまたはデスティネーションアドレスの一致が IPv4 パケットに対して有効になります。

レイヤ 3 一致は、L3SAM0 または L3DAM0 ビットがセットされたときにのみ実行されます。

レイヤ 4 アドレスフィルタ 0 レジスタ（ETH_MACL4A0R）

アドレスオフセット：0x0904

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L4DP0[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L4SP0[15:0]															
rw															

ビット 31:16 **L4DP0[15:0]**：レイヤ 4 デスティネーションポート番号フィールド

ETH_MACL3L4C0R レジスタで L4PEN0 ビットがリセットされ、L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの TCP デスティネーションポート番号フィールドと一致する値を含みます。

ETH_MACL3L4C0R レジスタで L4PEN0 および L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの UDP デスティネーションポート番号フィールドと一致する値を含みます。

ビット 15:0 **L4SP0[15:0]**：レイヤ 4 ソースポート番号フィールド

ETH_MACL3L4C0R レジスタで L4PEN0 ビットがリセットされ、L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの TCP ソースポート番号フィールドと一致する値を含みます。

ETH_MACL3L4C0R レジスタで L4PEN0 および L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの UDP ソースポート番号フィールドと一致する値を含みます。

レイヤ 3 アドレス 0 フィルタ 0 レジスタ（ETH_MACL3A00R）

アドレスオフセット：0x0910

リセット値：0x0000 0000

IPv4 パケットでは、レイヤ 3 アドレス 0 フィルタ 0 レジスタは、32 ビットの IP ソースアドレスフィールドを含みます。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [31:0] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A00[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A00[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A00[31:0]**：レイヤ 3 アドレス 0 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [31:0] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [31:0] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされ、L3SAM0 ビットがセットされた場合、このフィールドは IPv4 パケットの IP ソースアドレスフィールドと一致する値を含みます。

レイヤ 3 アドレス 1 フィルタ 0 レジスタ（ETH_MACL3A10R）

アドレスオフセット：0x0914

リセット値：0x0000 0000

IPv4 パケットでは、レイヤ 3 アドレス 1 フィルタ 0 レジスタは、32 ビットの IP デスティネーションアドレスフィールドを含みます。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [63:32] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A10[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A10[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A10[31:0]**：レイヤ 3 アドレス 1 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [63:32] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [63:32] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされ、L3SAM0 ビットがセットされた場合、このフィールドは IPv4 パケットの IP デスティネーションアドレスフィールドと一致する値を含みます。

レイヤ 3 アドレス 2 フィルタ 0 レジスタ（ETH_MACL3A20）

アドレスオフセット：0x0918

リセット値：0x0000 0000

レイヤ 3 アドレス 2 フィルタ 0 レジスタは、IPv4 パケットでは予約済みとなります。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [95:64] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A20[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A20[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A20[31:0]**：レイヤ 3 アドレス 2 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [95:64] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [95:64] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされた場合、このフィールドは使用されません。

レイヤ 3 アドレス 3 フィルタ 0 レジスタ（ETH_MACL3A30）

アドレスオフセット：0x091C

リセット値：0x0000 0000

レイヤ 3 アドレス 3 フィルタ 0 レジスタは、IPv4 パケットでは予約済みとなります。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [127:96] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A30[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A30[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A30[31:0]**：レイヤ 3 アドレス 3 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [127:96] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [127:96] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされた場合、このフィールドは使用されません。

L3 および L4 制御 1 レジスタ（ETH_MACL3L4C1R）

アドレスオフセット：0x0930

リセット値：0x0000 0000

レイヤ 3 およびレイヤ 4 制御レジスタは、レイヤ 3 およびレイヤ 4 のフィルタ 0 の動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	RESERVED_DMCHEN1	Res.	Res.	Res.	Res.	Res.	Res.	L4DPIM1	L4DPM1	L4SPIM1	L4SPM1	Res.	L4PEN1
			r							rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3HDBM1[4:0]					L3HSBM1[4:0]					L3DAIM1	L3DAM1	L3SAIM1	L3SAM1	Res.	L3PEN1
rw					rw					rw	rw	rw	rw		rw

ビット 31:22　予約済みであり、リセット値に保持する必要があります。

ビット 21　**L4DPIM1**：レイヤ 4 デスティネーションポート反転一致の有効化

このビットがセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの完全一致が有効になります。

このビットは、L4DPM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 20　**L4DPM1**：レイヤ 4 デスティネーションポート一致の有効化

このビットがセットされた場合、レイヤ 4 デスティネーションポート番号フィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 4 デスティネーションポート番号フィールドの一致を無視します。

ビット 19　**L4SPIM1**：レイヤ 4 ソースポート反転一致の有効化

このビットがセットされた場合、レイヤ 4 ソースポート番号フィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 4 ソースポート番号フィールドの完全一致が有効になります。

このビットは、L4SPM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 18　**L4SPM1**：レイヤ 4 ソースポート一致の有効化

このビットがセットされた場合、レイヤ 4 ソースポート番号フィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 4 ソースポート番号フィールドの一致を無視します。

ビット 17　予約済みであり、リセット値に保持する必要があります。

ビット 16　**L4PEN1**：レイヤ 4 プロトコルの有効化

このビットがセットされた場合、UDP パケットのソースおよびデスティネーションポート番号が一致に使用されます。このビットがリセットされた場合、TCP パケットのソースおよびデスティネーションポート番号が一致に使用されます。

レイヤ 4 一致は、L4SPM0 または L4DPM0 ビットがセットされたときにのみ実行されます。

ビット 15:11 **L3HDBM1[4:0]**：レイヤ 3 IP DA 上位ビット一致

IPv4 パケット：

このフィールドは、IPv4 パケットで一致する IP デスティネーションアドレスの上位ビットの数を含みます。次のリストでは、このフィールドの値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

31：MSb を除くすべてのビットがマスクされます。

IPv6 パケット：

このフィールドのビット [12:11] は、IPv6 パケットでマスクされる IP ソースまたはデスティネーションアドレスの下位ビットの数を示す L3HSBM0 のビット [6:5] に対応します。次のリストでは、L3HDBM0[1:0] および L3HSBM0 ビットの連結された値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

127：MSb を除くすべてのビットがマスクされます。

このフィールドは、L3DAM0 または L3SAM0 ビットがセットされたときにのみ有効であり、適用されます。

ビット 10:6 **L3HSBM1[4:0]**：レイヤ 3 IP SA 上位ビット一致

IPv4 パケット：

このフィールドは、IPv4 パケットでの一致に対してマスクされる IP ソースアドレスの下位ビットの数を含みます。次のリストでは、このフィールドの値を示しています。

0：ビットはマスクされません。

1：LSb[0] がマスクされます。

2：2 つの LSb [1:0] がマスクされます。

..

31：MSb を除くすべてのビットがマスクされます。

IPv6 パケット：

このフィールドは、L3HSBM0 のビット [4:0] を含みます。これらのビットは、IPv6 パケットで一致する IP ソースまたはデスティネーションアドレスの上位ビットの数を示します。このフィールドは、L3DAM0 または L3SAM0 ビットがハイにセットされたときにのみ有効であり、適用されます。

ビット 5 **L3DAIM1**：レイヤ 3 IP DA 反転一致の有効化

このビットがセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの完全一致が有効になります。

このビットは、L3DAM0 ビットがハイにセットされた場合にのみ有効であり、適用されます。

ビット 4 **L3DAM1**：レイヤ 3 IP DA 一致の有効化

このビットがセットされた場合、レイヤ 3 IP デスティネーションアドレスフィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 3 IP デスティネーションアドレスフィールドの一致を無視します。

注： L3PEN0 ビットがセットされた場合、IPv6 DA または SA がフィルタでチェックできるため、このビットか L3SAM0 ビットをセットする必要があります。

ビット 3 **L3SAIM1**：レイヤ 3 IP SA 反転一致の有効化

このビットがセットされた場合、レイヤ 3 IP ソースアドレスフィールドの反転一致が有効になります。このビットがリセットされた場合、レイヤ 3 IP ソースアドレスフィールドの完全一致が有効になります。

このビットは、L3SAM0 ビットがセットされた場合にのみ有効であり、適用されます。

ビット 2 **L3SAM1**：レイヤ 3 IP SA 一致の有効化

このビットがセットされた場合、レイヤ 3 IP ソースアドレスフィールドの一致が有効になります。このビットがリセットされた場合、MAC はレイヤ 3 IP ソースアドレスフィールドの一致を無視します。

注： L3PEN0 ビットがセットされた場合、IPv6 SA または DA がフィルタでチェックできるため、このビットか L3DAM0 ビットをセットする必要があります。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **L3PEN1**：レイヤ 3 プロトコルの有効化

このビットがセットされた場合、レイヤ 3 IP ソースまたはデスティネーションアドレスの一致が IPv6 パケットに対して有効になります。このビットがリセットされた場合、レイヤ 3 IP ソースまたはデスティネーションアドレスの一致が IPv4 パケットに対して有効になります。

レイヤ 3 一致は、L3SAM0 または L3DAM0 ビットがセットされたときにのみ実行されます。

レイヤ 4 アドレスフィルタ 1 レジスタ（ETH_MACL4A1R）

アドレスオフセット：0x0934

リセット値：0x0000 0000

レイヤ 4 アドレス 0 レジスタおよび 580～667 のレジスタは、レイヤ 3 およびレイヤ 4 パケットフィルタの有効化オプションがコア設定中に選択されていない場合、予約済み（デフォルト値で読み出し専用）となります。

レイヤ 3 およびレイヤ 4 アドレスレジスタは、コア設定中にレイヤ 3 およびレイヤ 4 アドレスレジスタの Rx クロックドメインへの同期オプションを選択することで、二重同期されるように設定できます。このオプションを選択した場合、レイヤ 3 およびレイヤ 4 アドレスレジスタのビット [31:24]（リトルエンディアンモード）またはビット [7:0]（ビッグエンディアンモード）が書き込まれたときのみ、同期がトリガされます。適切に同期更新を行うために、同じレイヤ 3 およびレイヤ 4 アドレスレジスタへの連続書き込みは、デスティネーションクロックの少なくとも 4 クロックサイクル遅延後に実行されなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L4DP1[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L4SP1[15:0]															
rw															

ビット 31:16 **L4DP1[15:0]**：レイヤ 4 デスティネーションポート番号フィールド

ETH_MACL3L4C0R レジスタで L4PEN0 ビットがリセットされ、L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの TCP デスティネーションポート番号フィールドと一致する値を含みます。

ETH_MACL3L4C0R レジスタで L4PEN0 および L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの UDP デスティネーションポート番号フィールドと一致する値を含みます。

ビット 15:0 **L4SP1[15:0]**：レイヤ 4 ソースポート番号フィールド

ETH_MACL3L4C0R レジスタで L4PEN0 ビットがリセットされ、L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの TCP ソースポート番号フィールドと一致する値を含みます。

ETH_MACL3L4C0R レジスタで L4PEN0 および L4DPM0 ビットがセットされた場合、このフィールドは IPv4 または IPv6 パケットの UDP ソースポート番号フィールドと一致する値を含みます。

レイヤ 3 アドレス 0 フィルタ 1 レジスタ（ETH_MACL3A01R）

アドレスオフセット：0x0940

リセット値：0x0000 0000

IPv4 パケットでは、レイヤ 3 アドレス 0 フィルタ 0 レジスタは、32 ビットの IP ソースアドレスフィールドを含みます。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [31:0] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A01[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A01[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A01[31:0]**：レイヤ 3 アドレス 0 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [31:0] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [31:0] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされ、L3SAM0 ビットがセットされた場合、このフィールドは IPv4 パケットの IP ソースアドレスフィールドと一致する値を含みます。

レイヤ 3 アドレス 1 フィルタ 1 レジスタ（ETH_MACL3A11R）

アドレスオフセット：0x0944

リセット値：0x0000 0000

IPv4 パケットでは、レイヤ 3 アドレス 1 フィルタ 0 レジスタは、32 ビットの IP デスティネーションアドレスフィールドを含みます。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [63:32] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A11[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A11[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A11[31:0]**：レイヤ 3 アドレス 1 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [63:32] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [63:32] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされ、L3SAM0 ビットがセットされた場合、このフィールドは IPv4 パケットの IP デスティネーションアドレスフィールドと一致する値を含みます。

レイヤ 3 アドレス 2 フィルタ 1 レジスタ（ETH_MACL3A21R）

アドレスオフセット：0x0948

リセット値：0x0000 0000

レイヤ 3 アドレス 2 フィルタ 0 レジスタは、IPv4 パケットでは予約済みとなります。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [95:64] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A21[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A21[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A21[31:0]**：レイヤ 3 アドレス 2 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [95:64] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [95:64] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされた場合、このフィールドは使用されません。

レイヤ 3 アドレス 3 フィルタ 1 レジスタ（ETH_MACL3A31R）

アドレスオフセット：0x94C

リセット値：0x0000 0000

レイヤ 3 アドレス 3 フィルタ 0 レジスタは、IPv4 パケットでは予約済みとなります。IPv6 パケットでは、128 ビットの IP ソースアドレスまたはデスティネーションアドレスフィールドのビット [127:96] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L3A31[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L3A31[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **L3A31[31:0]**：レイヤ 3 アドレス 3 フィールド

ETH_MACL3L4C0R レジスタで L3PEN0 および L3SAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP ソースアドレスフィールドのビット [127:96] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 および L3DAM0 ビットがセットされた場合、このフィールドは IPv6 パケットの IP デスティネーションアドレスフィールドのビット [127:96] と一致する値を含みます。

ETH_MACL3L4C0R レジスタで L3PEN0 ビットがリセットされた場合、このフィールドは使用されません。

タイムスタンプ制御レジスタ（ETH_MACTSCR）

アドレスオフセット：0x0B00

リセット値：0x0000 2000

このレジスタは、システム時間生成回路の動作およびレシーバのタイムスタンプの PTP パケットの処理を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXTSSTSM	Res.	Res.	Res.	Res.	CSC	TSENMACADDR	SNAPTYPSEL[1:0]	
							rw					r	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSMSTRENA	TSEVNTENA	TSIPV4ENA	TSIPV6ENA	TSIPENA	TSVER2ENA	TSCTRLSSR	TSENALL	Res.	Res.	TSADDREG	Res.	TSUPDT	TSINIT	TSCFUPDT	TSENA
rw	rw	rw	rw	rw	rw	rw	rw			rw		rw	rw	rw	rw

ビット 31:25　予約済みであり、リセット値に保持する必要があります。

ビット 24　**TXTSSTSM**：送信タイムスタンプステータスモード

このビットがセットされた場合、MAC はソフトウェアによって読み出されない場合でも、以前の送信タイムスタンプステータスを上書きします。MAC は、ETH_MACTxTSSNR レジスタの TXTSSTSMIS ビットをセットして示します。

このビットがリセットされると、MAC はソフトウェアによって前のパケットのタイムスタンプステータスが読み出されない場合、現在のタイムスタンプステータスを無視します。MAC は、ETH_MACTxTSSSR レジスタの TXTSSTSHI ビットをセットして示します。

ビット 23:20　予約済みであり、リセット値に保持する必要があります。

ビット 19　**CSC**：UDP/IPv4 経由 PTP パケットの OST 時のチェックサム補正の有効化

このビットがセットされた場合、UDP/IPv4 経由で送信された PTP メッセージの最後の 2 バイトは、ワンステップタイムスタンプ動作の一環として元のタイムスタンプおよび／または補正フィールドに施される変更に対して UDP チェックサムを正しい状態にするために更新されます。アプリケーションは、これらの 2 個のダミーバイトでパケットを形成します。

リセットされた場合、UDP チェックサムを正しい状態にする更新は行われません。アプリケーションは、UDP チェックサムを 0 にセットした状態でパケットを形成します。

ビット 18　**TSENMACADDR**：PTP パケットフィルタ用の MAC アドレスの有効化

このビットがセットされているとき、PTP が Ethernet 経由で直接送信される場合、（MAC アドレスレジスタと一致する）DA MAC アドレスは PTP パケットをフィルタするために使用されます。

このビットがセットされているとき、PTP が Ethernet 経由で直接送信される場合、MAC アドレスレジスタでプログラムされたものと一致する特別なマルチキャストまたはユニキャストアドレスを含む DA を持って受信される PTP パケットは、以下に示す処理を考慮します。

通常のタイムスタンプ動作の場合、MAC アドレスレジスタ 0～31 はユニキャストデスティネーションアドレスの一致で考慮されます。

PTP オフロードの場合、MAC アドレスレジスタ 0 だけがユニキャストデスティネーションアドレスの一致で考慮されます。

ビット 17:16　**SNAPTYPSEL[1:0]**：スナップショット取得のための PTP パケットの選択

これらのビットは、ビット 15 および 14 とともに、スナップショットを取得する必要がある PTP パケットタイプのセットを決定します。エンコーディングは、タイムスタンプスナップショットのレジスタビットに対する依存性の表に示されています。

ビット 15 **TSMSTRENA**：マスタに関係するメッセージのスナップショットの有効化

このビットがセットされている場合、マスタノードに関係するメッセージのみスナップショットが取られます。セットされていない場合は、スレーブノードに関係するメッセージのみスナップショットが取られます。

ビット 14 **TSEVNTENA**：イベントメッセージのタイムスタンプスナップショットの有効化

このビットがセットされた場合、イベントメッセージ （SYNC、Delay_Req、Pdelay_Req または Pdelay_Resp）に対してのみタイムスタンプスナップショットが取られます。このビットがリセットされた場合は、アナウンス、管理、シグナリング以外の全メッセージのスナップショットが取られます。タイムスタンプスナップショットの詳細については、タイムスタンプスナップショットのレジスタビットに対する依存性の表を参照してください。。

ビット 13 **TSIPV4ENA**：IPv4-UDP 経由で送信された PTP パケットの処理の有効化

このビットがセットされた場合、MAC レシーバは IPv4-UDP パケットにカプセル化された PTP パケットを処理します。このビットがリセットされた場合、MAC は IPv4-UDP 経由で転送された PTP パケットを無視します。このビットはデフォルトでセットされます。

ビット 12 **TSIPV6ENA**：IPv6-UDP 経由で送信された PTP パケットの処理の有効化

このビットがセットされた場合、MAC レシーバは IPv6-UDP パケットにカプセル化された PTP パケットを処理します。このビットがクリアされた場合、MAC は IPv6-UDP 経由で転送された PTP パケットを無視します。

ビット 11 **TSIPENA**：Ethernet 経由 PTP パケットの処理の有効化

このビットがセットされた場合、MAC レシーバは Ethernet パケットに直接カプセル化された PTP パケットを処理します。このビットがリセットされた場合、MAC は Ethernet 経由 PTP パケットを無視します。

ビット 10 **TSVER2ENA**：バージョン 2 フォーマット用 PTP パケット処理の有効化

このビットがセットされた場合、IEEE 1588 バージョン 2 フォーマットを使用して PTP パケットを処理します。このビットがリセットされた場合、IEEE 1588 バージョン 1 フォーマットを使用して PTP パケットを処理します。IEEE 1588 フォーマットは、「PTP の処理および制御」で説明しています。

ビット 9 **TSCTRLSSR**：タイムスタンプデジタルまたはバイナリロールオーバー制御

このビットがセットされた場合、0x3B9A_C9FF の値の後（1 ナノ秒精度）にタイムスタンプ下位レジスタがロールオーバーし、タイムスタンプ（ハイ）セカンドをインクリメントさせます。このビットがリセットされた場合、サブセカンドレジスタのロールオーバー値は 0x7FFF_FFFF です。サブセカンドのインクリメントは、PTP のリファレンスクロック周波数とこのビットの値に応じて正しくプログラムされる必要があります。

ビット 8 **TSENALL**：すべてのパケットに対するタイムスタンプの有効化

このビットがセットされた場合、MAC が受信する全パケットに対してタイムスタンプスナップショットが有効化されます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TSADDREG**：加数レジスタの更新

このビットがセットされた場合、タイムスタンプ加数レジスタの内容が、微調整のために PTP ブロックで更新されます。このビットは、更新が完了するとクリアされます。このビットはセット前に 0 にする必要があります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **TSUPDT**：タイムスタンプの更新

このビットがセットされた場合、システム時間は、ETH_MACSTSUR および ETH_MACSTNUR レジスタで指定された値で更新されます（加算または減算されます）。

このビットは更新前に 0 にする必要があります。このビットは、ハードウェアで更新が完了するとリセットされます。タイムスタンプ上位ワードレジスタ（コア設定中に有効にされた場合）は更新されません。

ビット 2 **TSINIT**：タイムスタンプの初期化

このビットがセットされた場合、システム時間は、MAC レジスタ 80（システム時間秒更新レジスタ）および MAC レジスタ 81（システム時間ナノ秒更新レジスタ）で指定された値で初期化されます（上書きされます）。

このビットは更新前に 0 にする必要があります。このビットは、初期化が完了するとリセットされます。タイムスタンプ上位ワードレジスタ（コア設定中に有効にされた場合）のみ初期化できます。

ビット 1 **TSCFUPDT**：粗密タイムスタンプ更新

このビットがセットされた場合、密更新法を使用してシステムタイムスタンプが更新されます。このビットがリセットされた場合、粗更新法を使用してシステムタイムスタンプが更新されます。

ビット 0 **TSENA**：タイムスタンプの有効化

このビットがセットされると、送信および受信パケットにタイムスタンプが追加されます。無効化された場合、送信および受信パケットにタイムスタンプは追加されず、タイムスタンプ生成回路もサスペンドされます。このモードを有効にした後、タイムスタンプ（システム時間）を初期化する必要があります。

受信側では、このビットがセットされた場合のみ、MAC は 1588 パケットを処理します。

表 544 では、ETH_MACTSCR レジスタの SNAPTYPSEL フィールドに応じてスナップショットが取られる PTP メッセージを示しています。

表 544. タイムスタンプスナップショットのレジスタビットに対する依存性

SNAPTYPSEL	TSMSTRENA	TSEVNTENA	PTP メッセージ
00	X	0	SYNC、Follow_Up、Delay_Req、Delay_Resp
00	0	1	SYNC
00	1	1	Delay_Req
01	X	0	SYNC、Follow_Up、Delay_Req、Delay_Resp、Pdelay_Req、Pdelay_Resp、Pdelay_Resp_Follow_Up
01	0	1	SYNC、Pdelay_Req、Pdelay_Resp
01	1	1	Delay_Req、Pdelay_Req、Pdelay_Resp
10	X	X	SYNC、Delay_Req
11	X	X	Pdelay_Req、Pdelay_Resp

サブセカンドインクリメントレジスタ（ETH_MACSSIR）

アドレスオフセット：0x0B04

リセット値：0x0000 0000

サブセカンドインクリメントレジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。粗更新モードでは（ETH_PTPTSCR レジスタのビット 1）、HCLK クロックサイクルごとに、このレジスタの値がシステム時間に加算されます。密更新モードでは、加算器がオーバーフローするたびに、このレジスタの値がシステム時間に加算されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSINC[7:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SNSINC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															

ビット 31:24　予約済みであり、リセット値に保持する必要があります。

ビット 23:16　**SSINC[7:0]**：サブセカンドインクリメント値

このフィールドにプログラムされる値は、サブセカンドレジスタの内容で（clk_ptp_i の）各クロックサイクルごとに累積されます。たとえば、PTP クロックが 50 MHz（周期は 20 ns）であるとき、システム時間ナノ秒レジスタに 1 ns の精度がある場合（ETH_MACTSCR でビット 9（TSCTRLSSR）がセットされている場合）、20（0x14）をプログラムする必要があります。TSCTRLSSR がクリアされた場合、ナノ秒レジスタの分解能は～0.465 ns です。この場合、20 ns/0.465 を基にした 43（0x2B）の値をプログラムする必要があります。

SNSINC　**SSINC[7:0]**：サブナノセカンドインクリメント値

このフィールドは、2^8 で掛けられたナノ秒単位で表されるサブナノセカンドインクリメント値を含みます。

この値は、サブセカンドレジスタのサブナノセカンドフィールドで累積されます。

たとえば、ETH_MACTSCR レジスタの TSCTRLSSR フィールドがセットされた場合、かつ必要なインクリメントが 5.3ns である場合、SSINC は 0x05、SNSINC は 0x4C に設定する必要があります。

ビット 15:0　予約済みであり、リセット値に保持する必要があります。

システム時間秒レジスタ（ETH_MACSTSR）

アドレスオフセット：0x0B08

リセット値：0x0000 0000

システム時間秒レジスタは、システム時間ナノ秒レジスタとともに、MAC によって維持されているシステム時間の現在値を示します。これは連続的に更新されますが、クロックドメイン転送レイテンシ（HCLK から CSR へ）によって実際の時間から少し遅延します。

このレジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSS[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **TSS[31:0]**：タイムスタンプの秒

このフィールドの値は、MAC によって維持されているシステム時間の現在値を秒で示します。

システム時間ナノ秒レジスタ（ETH_MACSTNR）

アドレスオフセット：0x0B0C

リセット値：0x0000 0000

システム時間ナノ秒レジスタは、システム時間秒レジスタとともに、MAC によって維持されているシステム時間の現在値を示します。

このレジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:0 **TSSS[30:0]**：タイムスタンプのサブセカンド

このフィールドの値は、0.46 ns 精度のサブセカンド表示です。ETH_MACTSCR でビット 9 がセットされた場合、各ビットは 1 ns を表します。最大値は 0x3B9A_C9FF で、これ以降は 0 にロールオーバーします。

システム時間秒更新レジスタ（ETH_MACSTSUR）

アドレスオフセット：0x0B10

リセット値：0x0000 0000

システム時間秒更新レジスタは、システム時間ナノ秒更新レジスタとともに、MAC によって維持されているシステム時間を初期化または更新します。ETH_MACTSCR レジスタで TSINIT または TSUPDT ビットをセットする前に、両方のレジスタを書き込む必要があります。

このレジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSS[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSS[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TSS[31:0]**：タイムスタンプの秒

このフィールドの値は、更新のサブセカンド部分です。ADDSUB がリセットされた場合、このフィールドには ETH_MACTSCR レジスタの TSCTRLSSR ビットに基づいた精度で、更新値のサブセカンド部分をプログラムする必要があります。ADDSUB がセットされた場合、このフィールドには以下に示す更新値のサブセカンド部分の補数をプログラムする必要があります。

TSCTRLSSR がセットされた場合、プログラムされる値は 10^9 - <サブセカンド値> にする必要があります。TSCTRLSSR がリセットされた場合、プログラムされる値は 2^31 - <サブセカンド値> にする必要があります。

たとえば、TSCTRLSSR ビットがセットされ、2.000000001 秒をシステム時間から減算する必要がある場合、MAC_Timestamp 秒更新レジスタの TSS フィールドを 0xFFFF_FFFE（つまり、2^32 - 2）にし、このレジスタの ADDSUB ビットをセットし、TSSS フィールドを 0x3B9A_C9FF（つまり、10^9 - 1）にする必要があります。

システム時間ナノ秒更新レジスタ（ETH_MACSTNUR）

アドレスオフセット：0x0B14

リセット値：0x0000 0000

このレジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDSUB	TSSS[30:16]														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSSS[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ADDSUB**：時間の加算または減算

このビットがセットされた場合、時間値は更新レジスタの内容で減算されます。このビットがリセットされた場合、時間値は更新レジスタの内容で加算されます。

ビット 30:0 **TSSS[30:0]**：タイムスタンプのサブセカンド

このフィールドの値は、0.46 ns 精度のサブセカンド表示です。TSCTRLSSR ビットが ETH_MACTSCR レジスタでセットされている場合、各ビットは 1 ns を表し、プログラムされる値は 0x3B9A_C9FF を超えてはなりません。

タイムスタンプ加数レジスタ（ETH_MACTSAR）

アドレスオフセット：0x0B18

リセット値：0x0000 0000

タイムスタンプ加数レジスタは、IEEE 1588 タイムスタンプ機能が外部タイムスタンプ入力がない状態で選択されている場合のみ存在します。このレジスタ値は、システム時間が密更新モード（ETH_MACTSCR レジスタの TSCFUPDT ビット）で設定されているときにのみ使用されます。このレジスタの内容は、（HCLK の）クロックサイクルごとに 32 ビットの加算器に加算され、加算器がオーバーフローするたびにシステム時間が更新されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TSAR[31:0]**：タイムスタンプ加数レジスタ

このフィールドは、時間同期を達成するために、加算器レジスタに加算される 32 ビットの時間値を示します。

タイムスタンプステータスレジスタ（ETH_MACTSSR）

アドレスオフセット：0x0B20

リセット値：0x0000 0000

タイムスタンプステータスレジスタは、IEEE 1588 タイムスタンプ機能が選択されている場合のみ存在します。アプリケーションがこのレジスタを読み出す場合、ビット [27:25] を除くすべてのビットがクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ATSNS[4:0]					ATSSTM	Res.	Res.	Res.	Res.	ATSSTN[3:0]			
		r					r					r			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXTSSIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSTRGTERR0	AUXTSTRIG	TSTARGT0	TSSOVF
r												r	r	r	r

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29:25 **ATSNS[4:0]**：補助タイムスタンプスナップショットの数

このフィールドは、FIFO で使用できるスナップショットの数を示します。選択された FIFO の深さに等しい値（4、8、または 16）は、補助スナップショット FIFO がフルであることを示します。これらのビットは、補助スナップショット FIFO クリアビットがセットされたときに（00000 に）クリアされます。このビットは、IEEE 1588 補助スナップショットの追加オプションが選択されたときにのみ有効です。

ビット 24 **ATSSTM**：補助タイムスタンプスナップショットトリガの見逃し

補助タイムスタンプスナップショット FIFO がフルで、外部トリガがセットされたときに、このビットがセットされます。これは、最新のスナップショットが FIFO に保存されないことを示します。このビットは、IEEE 1588 補助スナップショットの追加オプションが選択されたときにのみ有効です。

ビット 23:20　予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **ATSSTN[3:0]**：補助タイムスタンプスナップショットトリガの識別子

これらのビットは、補助スナップショットレジスタで使用できるタイムスタンプが適用できる補助トリガ入力を識別します。複数のビットが同時にセットされた場合、対応する補助トリガが同じクロックでサンプリングされたことを示します。これらのビットは、補助スナップショットの数が複数である場合のみ適用できます。1 ビットは次のリストに示すとおり各トリガに割り当てられます。

ビット 16：補助トリガ 0

ビット 17：補助トリガ 1

ビット 18：補助トリガ 2

ビット 19：補助トリガ 3

ソフトウェアはこのレジスタを読み出して、タイムスタンプが取得されたときにセットされるトリガを検索できます。

ビット 15 **TXTSSIS**：Tx タイムスタンプステータス割り込みステータス

ドロップ送信ステータスが MTL で有効になっている場合、キャプチャされた送信タイムスタンプが ETH_MACTxTSSNR および ETH_MACTxTSSSR レジスタで更新されると、このビットがセットされます。

PTP オフロード機能が有効になっている場合、キャプチャされた送信タイムスタンプが ETH_MACTxTSSNR および ETH_MACTxTSSSR レジスタで更新されたとき、PTO 生成遅延リクエストおよび Pdelay リクエストパケットに対して、このビットがセットされます。

ETH_MACTxTSSSR レジスタが読み出されると、このビットはクリアされます。

このビットはその他すべての設定で予約済みです。

ビット 14:4　予約済みであり、リセット値に保持する必要があります。

ビット 3 **TSTRGTERR0**：タイムスタンプ目標時間エラー

このビットは、ETH_MACPPS_Target_Time_seconds および ETH_MACPPS_Target_Time_Nanoseconds レジスタでプログラムされた最新の目標時間が経過したときにセットされます。このビットは、アプリケーションがこのビットを読み出すとクリアされます。

ビット 2 **AUXTSTRIG**：補助タイムスタンプトリガスナップショット

このビットは、補助スナップショットが FIFO に書き込まれたときにハイにセットされます。このビットは、IEEE 1588 補助スナップショットの追加オプションが選択されたときにのみ有効です。

ビット 1 **TSTARGT0**：タイムスタンプ目標時間到達

セットされている場合、このビットは、システム時間の値が ETH_MACPPS_Target_Time_seconds および ETH_MACPPS_Target_Time_Nanoseconds レジスタで指定される値以上であることを示します。

ビット 0 **TSSOVF**：タイムスタンプの秒オーバーフロー

このビットがセットされた場合、タイムスタンプの秒の値（バージョン 2 フォーマットをサポートしている場合）が 32'hFFFF_FFFF を超えてオーバーフローしたことを示します。

Tx タイムスタンプステータスナノ秒レジスタ（ETH_MACTxTSSNR）

アドレスオフセット：0x0B30

リセット値：0x0000 0000

このレジスタは、Tx ステータスが無効化されたときに送信パケットにキャプチャされたタイムスタンプのナノ秒部分を含みます。



ビット 31 **TXTSSMIS**：送信タイムスタンプステータスの見逃し

このビットがセットされた場合、以下のいずれかを示します。

– ETH_MACTSCR レジスタの TXTSSTSM ビットがリセットされた場合、現在のパケットのタイムスタンプは無視されます。

– ETH_MACTSCR レジスタの TXTSSTSM がセットされた場合、前のパケットのタイムスタンプが現在のパケットのタイムスタンプで上書きされます。

ビット 30:0 **TXTSSLO[30:0]**：送信タイムスタンプステータスロー

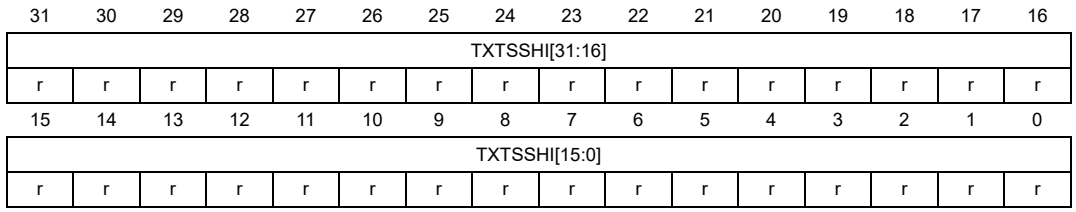
このフィールドは、送信パケットのキャプチャされたタイムスタンプのナノ秒フィールドの 31 ビットを含みます。

Tx タイムスタンプステータス秒レジスタ（ETH_MACTxTSSSR）

アドレスオフセット：0x0B34

リセット値：0x0000 0000

このレジスタは PTP パケットが送信されるときにキャプチャされるタイムスタンプ（秒単位）の上位 32 ビットを含みます。



ビット 31:0 **TXTSSHI[31:0]**：送信タイムスタンプステータスハイ

このフィールドは、送信パケットのキャプチャされたタイムスタンプの秒フィールドの下位 32 ビットを含みます。

補助制御レジスタ（ETH_MACACR）

アドレスオフセット：0x0B40

リセット値：0x0000 0000

補助タイムスタンプ制御レジスタは、補助タイムスタンプスナップショットを制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ATSEN3	ATSEN2	ATSEN1	ATSEN0	Res.	Res.	Res.	ATSFC
								rw	rw	rw	rw				rw

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7　**ATSEN3**：補助スナップショット 3 有効化

このビットは、補助スナップショットトリガ 3 のキャプチャを制御します。このビットがセットされた場合、ptp_aux_trig_i[3] 入力でのイベントの補助スナップショットが有効化されます。このビットがリセットされた場合、この入力でのイベントは無視されます。

以下のいずれかが当てはまる場合、このビットは予約済みとなります。

– IEEE 1588 補助スナップショットの追加オプションがコア設定中に選択されていません。

– IEEE 1588 補助スナップショット入力数オプションで選択された数が 4 未満です。

ビット 6　**ATSEN2**：補助スナップショット 2 有効化

このビットは、補助スナップショットトリガ 2 のキャプチャを制御します。このビットがセットされた場合、ptp_aux_trig_i[2] 入力でのイベントの補助スナップショットが有効化されます。このビットがリセットされた場合、この入力でのイベントは無視されます。

以下のいずれかが当てはまる場合、このビットは予約済みとなります。

– IEEE 1588 補助スナップショットの追加オプションがコア設定中に選択されていません。

– IEEE 1588 補助スナップショット入力数オプションで選択された数が 3 未満です。

ビット 5　**ATSEN1**：補助スナップショット 1 有効化

このビットは、補助スナップショットトリガ 1 のキャプチャを制御します。このビットがセットされた場合、ptp_aux_trig_i[1] 入力でのイベントの補助スナップショットが有効化されます。このビットがリセットされた場合、この入力でのイベントは無視されます。

以下のいずれかが当てはまる場合、このビットは予約済みとなります。

– IEEE 1588 補助スナップショットの追加オプションがコア設定中に選択されていません。

– IEEE 1588 補助スナップショット入力数オプションで選択された数が 2 未満です。

ビット 4 **ATSEN0**：補助スナップショット 0 有効化

このビットは、補助スナップショットトリガ 0 のキャプチャを制御します。このビットがセットされた場合、ptp_aux_trig_i[0] 入力でのイベントの補助スナップショットが有効化されます。このビットがリセットされた場合、この入力でのイベントは無視されます。

IEEE 1588 補助スナップショットの追加オプションがコア設定中に選択されていない場合、このビットは予約済みとなります。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **ATSFC**：補助スナップショット FIFO クリア

セットされると、このビットは補助スナップショット FIFO のポインタをリセットします。このビットは、ポインタがリセットされ FIFO がエンプティになったときにクリアされます。このビットがハイになった場合、補助スナップショットが FIFO に保存されます。

IEEE 1588 補助スナップショットの追加オプションがコア設定中に選択されていない場合、このビットは予約済みとなります。

補助タイムスタンプナノ秒レジスタ（ETH_MACATSNR）

アドレスオフセット：0x0B48

リセット値：0x0000 0000

補助タイムスタンプナノ秒レジスタは、ETH_MACATSSR とともに補助スナップショットとして保存される 64 ビットタイムスタンプを提供します。これら 2 つのレジスタは、4 ワードの深さを持つ 64 ビット幅 FIFO の読み出しポートを形成します。

この FIFO には複数のスナップショットを保存できます。ETH_MACTSSR のビット [29:25] は、FIFO のフィルレベルを示します。MAC レジスタ 91 の最後のバイト（補助タイムスタンプ - 秒レジスタ）が読み出されたときのみ、FIFO の最上部が削除されます。リトルエンディアンモードでは、ビット [31:24] が読み出されるとき、ビッグエンディアンモードでは、ビット [7:0] が読み出されるときを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AUXTSLO[30:16]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUXTSLO[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:0 **AUXTSLO[30:0]**：補助タイムスタンプ

補助タイムスタンプの下位 31 ビット（ナノ秒フィールド）を含みます。

補助タイムスタンプ秒レジスタ（ETH_MACATSSR）

アドレスオフセット：0x0B4C

リセット値：0x0000 0000

補助タイムスタンプ - 秒レジスタは、補助タイムスタンプレジスタの秒フィールドの下位 32 ビットを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AUXTSHI[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUXTSHI[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **AUXTSHI[31:0]**：補助タイムスタンプ

補助タイムスタンプの秒フィールドの下位 32 ビットを含みます。

タイムスタンプ入力非対称補正レジスタ（ETH_MACTSIACR）

アドレスオフセット：0x0B50

リセット値：0x0000 0000

MAC タイムスタンプ入力非対称補正レジスタは、PDelay_Resp PTP メッセージの補正フィールドを更新する際に使用する入力非対称補正値を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSTIAC[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSTIAC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **OSTIAC[31:0]**：ワンステップタイムスタンプ入力非対称補正

このフィールドは、Pdelay_Resp PTP パケットの correctionField に追加する入力パス非対称値を含みます。プログラムされる値はナノ秒単位とし、2^16 で掛ける必要があります。たとえば、2.5 ns は 0x00028000 で表されます。

この値は負にもでき、符号ビットを表すビット 31 で 2 の補数形式で表わすことができます。

タイムスタンプ出力非対称補正レジスタ（ETH_MACTSEACR）

アドレスオフセット：0x0B54

リセット値：0x0000 0000

MAC タイムスタンプ出力非対称補正レジスタは、PDelay_Req PTP メッセージの補正フィールドを更新する際に使用する出力非対称補正値を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSTEAC[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSTEAC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **OSTEAC[31:0]**：ワンステップタイムスタンプ出力非対称補正

このフィールドは、Pdelay_Resp PTP パケットの correctionField から差し引かれる出力パス非対称値を含みます。プログラムされる値は 2^16 で掛けられたナノ秒単位のネゲートされた値にする必要があります。

たとえば、必要な補正値が +2.5 ns の場合、プログラムされる値は 0xFFFD_8000 でなければならず、0x0002_8000（2.5 * 216）という 2 の補数となります。同様に、必要な補正値が -3.3 ns の場合、プログラムされる値は 0x0003_4CCC（3.3 * 216）となります。

タイムスタンプ入力補正ナノ秒レジスタ（ETH_MACTSICNR）

アドレスオフセット：0x0B58

リセット値：0x0000 0000

このレジスタは、入力パスでキャプチャされたタイムスタンプ値とともに使用されるナノ秒単位の補正値を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSIC[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TSIC[31:0]**：タイムスタンプ入力補正

このフィールドは、入力補正の式によって定義される入力パス補正値を含みます。

タイムスタンプ出力補正ナノ秒レジスタ（ETH_MACTSECNR）

アドレスオフセット：0x0B5C

リセット値：0x0000 0000

このレジスタは、出力パスでキャプチャされたタイムスタンプ値とともに使用されるナノ秒単位の補正値を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSEC[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSEC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TSEC[31:0]**：タイムスタンプ出力補正

このフィールドは、出力補正の式によって定義される出力パス補正値のナノ秒部分を含みます。

PPS 制御レジスタ（ETH_MACPPSCR）

アドレスオフセット：0x0B70

リセット値：0x0000 0000

PPS 制御レジスタは、タイムスタンプ機能が選択され、外部タイムスタンプが有効化されていない場合のみ存在します。

このレジスタのビット [30:24] は、4 つのフレキシブル PPS 出力が選択されている場合のみ有効です。ビット [22:16] は、3 つ以上のフレキシブル PPS 出力が選択されている場合のみ有効です。ビット [14:8] は、2 つ以上のフレキシブル PPS 出力が選択されている場合のみ有効です。ビット [6:4] は、フレキシブル PPS 機能が選択されている場合のみ有効です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGTMODSEL0[1:0]		PPSEN0	PPSCTRL_PPSCMD[3:0]			
									rw		rw	rw			

ビット 31:7　予約済みであり、リセット値に保持する必要があります。

ビット 6:5　**TRGTMODSEL0[1:0]**：PPS 出力の目標時間レジスタモード

このフィールドは、PPS 出力信号の目標時間レジスタ（MAC レジスタ 96 および 97）モードを示します。

00：目標時間レジスタは、割り込みイベントを生成するためだけにプログラムされます。

01：予約済みです。

10：目標時間レジスタは、割り込みイベントを生成して、PPS 出力信号生成を開始または停止するためにプログラムされます。

11：目標時間レジスタは、PPS 出力信号生成を開始または停止するためだけにプログラムされます。割り込みはアサートされません。

ビット 4 **PPSEN0**：フレキシブル PPS 出力モード有効化

このビットがセットされた場合、ビット [3:0] が PPSCMD として機能します。このビットがリセットされた場合、ビット [3:0] が PPSCTRL（固定 PPS モード）として機能します。

ビット 3:0 **PPSCTRL[3:0]**：PPS 出力周波数制御

このレジスタは、PPS 出力（ptp_pps_o）信号の周波数を制御します。PPSCTRL のデフォルト値は 0000 で、PPS 出力は毎秒 1 パルス（width clk_ptp_i）です。PPSCTRL のその他の値では、PPS 出力は次の周波数のクロック生成となります。

0001：バイナリロールオーバーは 2 Hz で、デジタルロールオーバーは 1 Hz です。

0010：バイナリロールオーバーは 4 Hz で、デジタルロールオーバーは 2 Hz です。

0011：バイナリロールオーバーは 8 Hz で、デジタルロールオーバーは 4 Hz です。

0100：バイナリロールオーバーは 16 Hz で、デジタルロールオーバーは 8 Hz です。

..

1111：バイナリロールオーバーは 32.768 KHz で、デジタルロールオーバーは 16.384 KHz です。

注： ***バイナリロールオーバーモードでは、PPS 出力（ptp_pps_o）は、これらの周波数で 50% のデューティサイクルとなります。バイナリロールオーバーモードでは、PPS 出力周波数は平均数です。実際のクロックは、毎秒同期される異なった周波数となります。例：***

– *PPSCTRL = 0001 の場合、PPS（1 Hz）には、537 ms のロー周期と、463 ms のハイ周期があります。*

– *PPSCTRL = 0010 の場合、PPS（2 Hz）は、*
50% のデューティサイクルの一連の 1 クロックと、537 ms の周期となります。
463 ms 周期の 2 番目のクロック（268 ms ローおよび 195 ms ハイ）

– *PPSCTRL = 0011 の場合、PPS（4 Hz）は、*
50% のデューティサイクルの一連の 3 クロックと、268 ms の周期となります。
195 ms 周期の 4 番目のクロック（134 ms ローおよび 61 ms ハイ）

この動作は、ETH_MACSTNR レジスタのデジタルロールオーバーモードでのビットのノンリニア切り替えによるためです。

ビット 3:0 または

PPSCMD[3:0]：フレキシブル PPS 出力（ptp_pps_o[0]）制御

これらのビットを 0 以外の値にプログラムして、MAC にイベントを開始するよう命令します。コマンドが PTP クロックドメインに転送または同期されるときに、これらのビットは自動的にクリアされます。ソフトウェアは、これらのビットが「すべて 0」である場合のみプログラムされることを確認する必要があります。次のリストでは、PPSCMD0 の値を示しています。

0000：コマンドなし

0001：シングルパルス開始

このコマンドは、目標時間レジスタ（レジスタ 455 および 456）で定義された開始位置および PPS 幅レジスタで定義された時間の開始位置で立ち上がるシングルパルスを生成します。

0010：パルス列開始

このコマンドは、目標時間レジスタで定義された開始位置および PPS 幅レジスタで定義された時間の開始位置で立ち上がり、PPS 間隔レジスタで定義された間隔で繰り返されるパルスの列を生成します。デフォルトでは、PPS パルス列は「パルス列の同時停止」または「パルス列の即時停止」コマンドによって停止されない限り、フリーランニングです。

0011：開始のキャンセル

このコマンドは、システム時間がプログラムされた開始時間を超えなかった場合に、シングルパルス開始およびパルス列開始コマンドをキャンセルします。

0100：パルス列の同時停止

このコマンドは、目標時間レジスタでプログラムされた時間が経過した後にパルス列開始コマンド（PPSCMD = 0010）で開始されたパルス列を停止します。

0101：パルス列の即時停止

このコマンドは、パルス列開始コマンド（PPSCMD = 0010）で開始されたパルス列をただちに停止します。

0110：パルス列停止のキャンセル

このコマンドは、プログラムされた停止時間が経過していない場合に、パルス列の同時停止のコマンドをキャンセルします。PPS パルス列は、このコマンドが正常に実行されるとフリーランニングになります。

0111-1111：予約済み

PPS 目標時間秒レジスタ（ETH_MACPPSTTSR）

アドレスオフセット：0x0B80

リセット値：0x0000 0000

PPS 目標時間秒レジスタは、PPS 目標時間ナノ秒レジスタとともに、システム時間がこれらのレジスタでプログラムされた値を超えたときに割り込みイベント [ETH_MACTSSR のビット 1] をスケジュールするために使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSTRH0[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSTRH0[31:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **TSTRH0[31:0]**：PPS 目標時間秒レジスタ

このフィールドは、時間を秒単位で格納します。タイムスタンプ値が両方の目標タイムスタンプレジスタと一致するか、それを超えた場合に、MAC は PPS 信号出力を開始または停止して、ETH_MACPPSCR レジスタの対応する PPS 出力に選択された目標時間モードに基づいて割り込み（有効である場合）を生成します。

PPS 目標時間ナノ秒レジスタ（ETH_MACPPSTTNR）

アドレスオフセット：0x0B84

リセット値：0x0000 0000

PPS 目標時間ナノ秒レジスタは、複数のフレキシブル PPS 出力が選択されている場合のみ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TRGTBUSY0	TTSL0[30:16]														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TTSL0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TRGTBUSY0**：PPS 目標時間レジスタビジー

MAC は、ETH_MACPPSCR レジスタの PPSCMD0 フィールドが 010 か 011 にプログラムされている場合に、このビットをセットします。PPSCMD0 フィールドを 010 か 011 にプログラムして、MAC に目標時間レジスタを PTP クロックドメインと同期するよう命令します。

目標時間レジスタを PTP クロックドメインと同期した後に、MAC はこのビットをクリアします。このビットが 1 で読み出される場合、アプリケーションは目標時間レジスタを更新してはなりません。そうしないと、以前にプログラムされた時間の同期が破損します。

ビット 30:0 **TTSL0[30:0]**：PPS 出力の目標時間ロー

このレジスタは、時間値を（符号付き）ナノ秒単位で格納します。タイムスタンプの値が両方の目標タイムスタンプレジスタの値と一致する場合に、MAC は PPS 信号出力を開始または停止して、ETH_MACPPSCR の TRGTMODSEL0 フィールド（ビット [6:5]）に基づいて割り込み（有効である場合）を生成します。

TSCTRLSSR ビットが ETH_MACTSCR レジスタでセットされている場合、この値は 0x3B9A_C9FF を超えてはなりません。PPS 信号出力の実際の開始または停止時間に、サブセカンドインクリメント値の 1 単位までのエラーマージンが発生する可能性があります。

PPS 間隔レジスタ（ETH_MACPPSIR）

アドレスオフセット：0x0B88

リセット値：0x0000 0000

PPS 間隔レジスタは、PPS 信号出力（ptp_pps_o[0]）の立ち上がりエッジ間のサブセカンドインクリメント値の単位数を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PPSINT0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PPSINT0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PPSINT0[31:0]**：PPS 出力信号間隔

これらのビットは、PPS 信号出力の立ち上がりエッジ間の間隔を格納します。この間隔は、サブセカンドインクリメント値の単位数で格納されます。

必要な間隔未満の 1 つの値をプログラムする必要があります。たとえば、PTP リファレンスクロックが 50 MHz（20 ns の周期）であり、PPS 信号出力の立ち上がりエッジの期待する間隔が 100 ns（サブセカンドインクリメント値の 5 単位）である場合、このレジスタで値 4（5-1）をプログラムする必要があります。

PPS 幅レジスタ（ETH_MACPPSWR）

アドレスオフセット：0x0B8C

リセット値：0x0000 0000

PPS 幅レジスタは、PPS 信号出力（ptp_pps_o）の立ち上がりエッジと、対応する立ち下がりエッジの間のサブセカンドインクリメント値の単位数を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PPSWIDTH0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PPSWIDTH0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PPSWIDTH0[31:0]**：PPS 出力信号幅

これらのビットは、PPS 信号出力の立ち上がりエッジと、対応する立ち下がりエッジの間の幅を格納します。この幅は、サブセカンドインクリメント値の単位数で格納されます。

必要な間隔未満の 1 つの値をプログラムする必要があります。たとえば、PTP リファレンスクロックが 50 MHz（20 ns の周期）であり、PPS 信号出力の立ち上がりエッジと立ち上がりエッジの間の幅が 80 ns（サブセカンドインクリメント値の 4 単位）である場合、このレジスタで値 3（4-1）をプログラムする必要があります。

注：　このレジスタにプログラムされる値は、ETH_MACPP0IR　レジスタにプログラムされる値未満にする必要があります。

PTP オフロード制御レジスタ（ETH_MACPOCR）

アドレスオフセット：0x0BC0

リセット値：0x0000 0000

このレジスタは、PTP オフロードエンジン動作を制御します。PTP タイムスタンプオフロード機能の有効化オプションが選択されている場合のみ、このレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DN[7:0]								PDRDIS	DRRDIS	APDREQTRIG	ASYNCTRIG	Res.	APDREQEN	ASYNCEN	PTOEN
rw								rw	rw	rw	rw		rw	rw	rw

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DN[7:0]**：ドメイン数
このフィールドは、PTP ノードが動作しているドメイン数を示します。

ビット 7　予約済みであり、リセット値に保持する必要があります。

ビット 6 **DRRDIS**：PTO 遅延リクエスト／レスポンス生成の無効化
このビットがセットされた場合、プログラムされたモードの必要に応じて、受信 SYNC および遅延リクエストパケットそれぞれに対して遅延リクエストおよび遅延レスポンスが生成されません。

ビット 5 **APDREQTRIG**：自動 PTP Pdelay_Req メッセージトリガ
このビットがセットされた場合、1 つの PTP Pdelay_Req メッセージが送信されます。このビットは PTP Pdelay_Req メッセージが送信された後に自動的にクリアされます。アプリケーションは、この動作に APDREQEN ビットをセットする必要があります。

ビット 4 **ASYNCTRIG**：自動 PTP SYNC メッセージトリガ
このビットがセットされた場合、1 つの PTP SYNC メッセージが送信されます。このビットは PTP SYNC メッセージが送信された後に自動的にクリアされます。アプリケーションは、この動作に ASYNCEN ビットをセットする必要があります。

ビット 3　予約済みであり、リセット値に保持する必要があります。

ビット 2 **APDREQEN**：自動 PTP Pdelay_Req メッセージ有効化
このビットがセットされた場合、MAC がピアツーピアトランスペアレントモードになるようプログラムされているときに、プログラムされた間隔またはアプリケーションからのトリガに基づいて、PTP Pdelay_Req メッセージが周期的に生成されます。

ビット 1 **ASYNCEN**：自動 PTP SYNC メッセージ有効化
このビットがセットされた場合、MAC がクロックマスタモードになるようプログラムされているときに、プログラムされた間隔またはアプリケーションからのトリガに基づいて、PTP SYNC メッセージが周期的に生成されます。

ビット 0 **PTOEN**：PTP オフロード有効化
このビットがセットされた場合、PTP オフロード機能が有効になります。

PTP ソースポートアイデンティティ 0 レジスタ（ETH_MACSPI0R）

アドレスオフセット：0x0BC4

リセット値：0x0000 0000

このレジスタは、PTP ノードの 80 ビットソースポートアイデンティティのビット [31:0] を含みます。PTP タイムスタンプオフロード機能の有効化オプションが選択されている場合のみ、このレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **SPI0[31:0]**：ソースポートアイデンティティ 0
このフィールドは、PTP ノードの sourcePortIdentity のビット [31:0] を示します。

PTP ソースポートアイデンティティ 1 レジスタ（ETH_MACSPI1R）

アドレスオフセット：0x0BC8

リセット値：0x0000 0000

このレジスタは、PTP ノードの 80 ビットソースポートアイデンティティのビット [63:32] を含みます。PTP タイムスタンプオフロード機能の有効化オプションが選択されている場合のみ、このレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI1[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **SPI1[31:0]**：ソースポートアイデンティティ 1

このフィールドは、PTP ノードの sourcePortIdentity のビット [63:32] を示します。

PTP ソースポートアイデンティティ 2 レジスタ（ETH_MACSPI2R）

アドレスオフセット：0x0BCC

リセット値：0x0000 0000

このレジスタは、PTP ノードの 80 ビットソースポートアイデンティティのビット [79:64] を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI2[15:0]															
rw															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SPI2[15:0]**：ソースポートアイデンティティ 2

このフィールドは、PTP ノードの sourcePortIdentity のビット [79:64] を示します。

ログメッセージ間隔レジスタ（ETH_MACLMIR）

アドレスオフセット：0x0BD0

リセット値：0x0000 0000

このレジスタは自動 PTP パケット生成の周期的間隔を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LMPDRI[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DRSYNCR[2:0]			LSI[7:0]							
					rw			rw							

ビット 31:24 **LMPDRI[7:0]**：ログ最小 Pdelay_Req 間隔

このフィールドは、PTP ノードの logMinPdelayReqInterval を示します。このビットは、周期的 Pdelay リクエストパケット送信のスケジュールに使用されます。許可される値は -15～15 です。負の値は 2 の補数形式で表す必要があります。たとえば、必要な値が -1 の場合、プログラムされる値は、0xFF でなければなりません。

ビット 23:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DRSYNCR[2:0]**：

SYNC に対する Delay_Req の比率

スレーブモードでは、送信される Delay_Req メッセージの周波数制御に使用されます。

0：DelayReq が受信 SYNC ごとに生成されます。

1：DelayReq が SYNC のオルタネート受信ごとに生成されます。

2：4 個の SYNC メッセージごとに生成されます。

3：8 個の SYNC メッセージごとに生成されます。

4：16 個の SYNC メッセージごとに生成されます。

5：32 個の SYNC メッセージごとに生成されます。

6～7：予約済み

マスタは、DelayResp PTP メッセージのこの情報（logMinDelayReqInterval）をスレーブに送信します。受信時に、受信した DelayResp メッセージのこの値を処理し、それに応じてこのフィールドを更新します。スレーブモードでは、ホストは受信した値を上書きする必要がない限り、このレジスタの書き込み／更新を行ってはなりません。マスタモードでは、このフィールドと logSyncInterval （LSI）フィールドの合計が、生成されたマルチキャスト Delay_Resp PTP メッセージの logMinDelayReqInterval フィールドで提供されます。

ビット 7:0 **LSI[7:0]**：

ログ同期間隔

このフィールドは、PTP ノードがマスタであるときに自動生成された SYNC メッセージの周期性をを示します。許可される値は -15～15 です。負の値は 2 の補数形式で表す必要があります。たとえば、必要な値が -1 の場合、プログラムされる値は、0xFF でなければなりません。

Ethernet MAC レジスタマップとリセット値

表 545. Ethernet MAC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	ETH_MACCR	ARPEN	SARC[2:0]			IPC	IPG[2:0]			GPSLCE	S2KP	CST	ACS	WD	Res.	JD	JE	Res.	FES	DM	LM	ECRSFD	DO	DCRS	DR	Res.	BL[1:0]		DC	PRELEN[1:0]		TE	RE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x0004	ETH_MACECR	Res.	Res.	EIPG[4:0]					EIPGEN	Res.	Res.	Res.	Res.	Res.	USP	SPEN	DCRCC	Res.	Res.	GPSL[13:0]														
	リセット値			0	0	0	0	0	0						0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0008	ETH_MACPFR	RA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DNTU	IPFE	Res.	Res.	Res.	VTFE	Res.	Res.	Res.	Res.	Res.	HPF	SAF	SAIF	PCF[1:0]		DBF	PM	DAIF	HMC	HUC	PR
	リセット値	0										0	0				0						0	0	0	0	0	0	0	0	0	0	0
0x000C	ETH_MACWTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PWE	Res.	Res.	Res.	Res.	WTO[3:0]			
	リセット値																								0					0	0	0	0
0x0010	ETH_MACHT0R	HT31T0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	ETH_MACHT1R	HT63T32[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018 - 0x004C	予約済み																																
0x0050	ETH_MACVTR	EIVLRXS	Res.	EIVLS[1:0]		ERIVLT	EDVLP	VTHM	EVLRXS	Res.	EVLS[1:0]		DOVLTC	ERSVLM	ESVL	VTIM	ETV	VL[15:0]																
	リセット値	0		0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	予約済み																																
0x0058	ETH_MACVHTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLHT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	予約済み																																
0x0060	ETH_MACVIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLTI	CSVL	VLP	VLC[1:0]		VLT[15:0]															
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	ETH_MACIVIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VLTI	CSVL	VLP	VLC[1:0]		VLT[15:0]															
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0068 - 0x006C	予約済み																																
0x0070	ETH_MACQTxFCR	PT[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DZPQ	PLT[2:0]			Res.	Res.	TFE	FCB_BPA
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0			0	0
0x0074 - 0x008C	予約済み																																
0x0090	ETH_MACRxFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UP	RFE
	リセット値																															0	0
0x0094 - 0x00A8	予約済み																																
0x00AC	予約済み																																

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00B0	ETH_MACISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSTSIS	TXSTSIS	TSIS	Res.	MMCTXIS	MMCRXIS	MMCIS	Res.	Res.	LPIIS	PMTIS	PHYIS	Res.	Res.	Res. S
	リセット値																		0	0	0		0	0	0			0	0	0			
0x00B4	ETH_MACIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSTSIE	TXSTSIE	TSIE	Res.	Res.	Res.	Res.	Res.	Res.	LPIIE	PMTIE	PHYIE	Res.	Res.	Res.
	リセット値																		0	0	0							0	0	0			0
0x00B8	ETH_ MACRXTXSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWT	Res.	Res.	EXCOL	LCOL	EXDEF	LCARR	NCARR	TJT
	リセット値																								0			0	0	0	0	0	
0x00BC	予約済み																																
0x00C0	ETH_MACPCSR	RWKFILTRST	Res.	Res.	RWKPTR[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWKPFE	GLBLUCAST	Res.	Res.	RWKPRCVD	MGKPRCVD	Res.	Res.	RWKPKTEN	MGKPKTEN	PWRDWN
	リセット値	0			0	0	0	0	0														0	0			0	0			0	0	0
0x00C4	ETH_ MACRWKPFR	WKUPFRMFTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8 - 0x00CC	予約済み																																
0x00D0	ETH_MACLCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPITCSE	LPIATE	LPITXA	PLSEN	PLS	LPIEN	Res.	Res.	Res.	Res.	Res.	Res.	RLPIST	TLPIST	Res.	Res.	Res.	Res.	RLPIEX	RLPIEN	TLPIEX	TLPIEN
	リセット値											0	0	0	0	0	0							0	0					0	0	0	0
0x00D4	ETH_MACLTCR	Res.	Res.	Res.	Res.	Res.	Res.	LST[9:0]										TWT[15:0]															
	リセット値							1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
								0x03E8										0															
0x00D8	ETH_MACLETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPIET[16:0]																	Res.	Res.	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x00DC	ETH_ MAC1USTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIC_1US_CNTR[11:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x00E0 - 0x00F8	予約済み																																
0x00FC - 0x010C	予約済み																																
0x0110	ETH_MACVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USERVER[7:0]								SNPSVER[7:0]							
	リセット値																		0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	1
0x0114	ETH_MACDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFCSTS[1:0]		TPESTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFCFCSTS[1:0]		RPESTS
	リセット値														0	0	0														0	0	0
0x0118	予約済み																																

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0120	ETH_MACHWF1R	Res.	L3L4FNUM[3:0]				Res.	HASHTBLSZ[1:0]		POUOST	Res.	RAVSEL	AVSEL	DBGMEMA	TSOEN	SPHEN	DCBEN	ADDR64[1:0]		ADVTHWORD	PTOEN	OSTEN	TXFIFOSIZE[4:0]					SPRAM	RXFIFOSIZE[4:0]				
	リセット値		0	0	1	0		0	1	1		0	0	0	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	1	0	0
0x0124	ETH_MACHWF2R	Res.	AUXSNAPNUM[2:0]			Res.	PPSOUTNUM[2:0]			Res.	Res.	TXCHCNT[3:0]				Res.	Res.	RXCHCNT[3:0]				Res.	Res.	TXQCNT[3:0]				Res.	Res.	RXQCNT[3:0]			
	リセット値		1	0	0		0	0	1			0	0	0	0			0	0	0	0			0	0	0	0			0	0	0	0
0x012C - 0x01FC	予約済み																																
0x0200	ETH_MACMDIOAR	Res.	Res.	Res.	Res.	PSE	BTB	PA[4:0]				RDA[4:0]						Res.	NTC[2:0]			CR[3:0]				Res.	Res.	Res.	SKAP	GOC_1	GOC_0	C45E	MB
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0				0	0	0	0	0
0x0204	ETH_ MACMDIODR	RA[15:0]																MD[15:0]															
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0208 - 0x020C	予約済み																																
0x0AE0	ETH_MACARPAR	ARPPA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0214 - 0x02FC	予約済み																																
0x0300	ETH_MACA0HR	AE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDRHI[15:0]															
	リセット値	1																1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0304	ETH_MACA0LR	ADDRLO[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0308	ETH_MACA1HR	AE	SA	MBC[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDRHI[15:0]															
	リセット値	0	0	0	0	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30C	ETH_MACA1LR	ADDRLO[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0310	ETH_MACA2HR	AE	SA	MBC[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDRHI[15:0]															
	リセット値	0	0	0	0	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0314	ETH_MACA2LR	ADDRLO[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0318	ETH_MACA3HR	AE	SA	MBC[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDRHI[15:0]															
	リセット値	0	0	0	0	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x031C	ETH_MACA3LR	ADDRLO[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0320 - 0x06FC	予約済み																																

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0700	MMC_CONTROL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCDBC	Res.	Res.	CNTPRSTLVL	CNTPRST	CNTFREEZ	RSTONRD	CNTSTOPRO	CNTRST
	リセット値																								0			0	0	0	0	0	0
0x0704	MMC_RX_INTERRUPT	Res.	Res.	Res.	Res.	RXLPITRCIS	RXLPIUSCIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXUCGPIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXALGNERPIS	RXCRCERPIS	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0									0											0	0					
0x0708	MMC_TX_INTERRUPT	Res.	Res.	Res.	Res.	TXLPITRCIS	TXLPIUSCIS	Res.	Res.	Res.	Res.	TXGPKTIS	Res.	Res.	Res.	Res.	Res.	TXMCOLGPIS	TXSCOLGPIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0					0						0	0														
0x070C	MMC_RX_INTERRUPT_MASK	Res.	Res.	Res.	Res.	RXLPITRCIM	RXLPIUSCIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXUCGPIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXALGNERPIM	RXCRCERPIM	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0									0											0	0					
0x0710	MMC_TX_INTERRUPT_MASK	Res.	Res.	Res.	Res.	TXLPITRCIM	TXLPIUSCIM	Res.	Res.	Res.	Res.	TXGPKTIM	Res.	Res.	Res.	Res.	Res.	TXMCOLGPIM	TXSCOLGPIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値					0	0					0						0	0														
0x0714 - 0x0748	予約済み																																
0x074C	TX_SINGLE_COLLISION_GOOD_PACKETS	TXSNGLCOLG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0750	TX_MULTIPLE_COLLISION_GOOD_PACKETS	TXMULTCOLG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0754 - 0x0764	予約済み																																
0x0768	TX_PACKET_COUNT_GOOD	TXPKTG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x076C - 0x0790	予約済み																																
0x0794	RX_CRC_ERROR_PACKETS	RXCRCERR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0798	RX_ALIGNMENT_ERROR_PACKETS	RXALGNERR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x079C - 0x07C0	予約済み																																
0x07C4	RX_UNICAST_PACKETS_GOOD	RXUCASTG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07C8 - 0x07E8	予約済み																																
0x07EC	TX_LPI_USEC_CNTR	TXLPIUSC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07F0	TX_LPI_TRAN_CNTR	TXLPITRC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07F4	RX_LPI_USEC_CNTR	RXLPIUSC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07F8	RX_LPI_TRAN_CNTR	RXLPITRC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07FC - 0x08FC	予約済み																																
0x0900	ETH_MACL3L4C0R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	L4DPIM0	L4DPM0	L4SPIM0	L4SPM0	Res.	L4PEN0	L3HDBM0[4:0]					L3HSBM0[4:0]					L3DAIM0	L3DAM0	L3SAIM0	L3SAM0	Res.	L3PEN0
	リセット値											0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x0904	ETH_MACL4A0R	L4DP0[15:0]																L4SP0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0908 - 0x090C	予約済み																																
0x0910	ETH_MACL3A00R	L3A00[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0914	ETH_MACL3A10R	L3A10[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0918	ETH_MACL3A20R	L3A20[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x091C	ETH_MACL3A30R	L3A30[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0920 - 0x092C	予約済み																																
0x0930	ETH_MACL3L4C1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	L4DPIM1	L4DPM1	L4SPIM1	L4SPM1	Res.	L4PEN1	L3HDBM1[4:0]					L3HSBM1[4:0]					L3DAIM1	L3DAM1	L3SAIM1	L3SAM1	Res.	L3PEN1
	リセット値											0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x0934	ETH_MACL4A1R	L4DP1[15:0]																L4SP1[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0938 - 0x093C	予約済み																																

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0940	ETH_MACL3A01R	L3A01[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0944	ETH_MACL3A11R	L3A11[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0948	ETH_MACL3A21R	L3A21[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094C	ETH_MACL3A31R	L3A31[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0950 - 0x0AFC	予約済み																																
0x0B00	ETH_MACTSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXTSSTSM	Res.	Res.	Res.	Res.	CSC	TSENMACADDR	SNAPTYPSEL[1:0]		TSMSTRENA	TSEVNTENA	TSIPV4ENA	TSIPV6ENA	TSIPENA	TSVER2ENA	TSCTRLSSR	TSENALL	Res.	Res.	TSADDREG	Res.	TSUPDT	TSINIT	TSCFUPDT	TSENA
	リセット値				0				0					0	0	0	0	0	0	1	0	0	0	0	0			0		0	0	0	0
0x0B04	ETH_MACSSIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SSINC[7:0]								RESERVED_SNSINC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x0B08	ETH_MACSTSR	TSS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B0C	ETH_MACSTNR	Res.	TSSS[30:0]																														
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B10	ETH_MACSTSUR	TSS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B14	ETH_MACSTNUR	ADDSUB	TSSS[30:0]																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B18	ETH_MACTSAR	TSAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B1C	予約済み																																
0x0B20	ETH_MACTSSR	Res.	Res.	ATSNS[4:0]					ATSSTM	Res.	Res.	Res.	Res.	ATSSTN[3:0]				TXTSSIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSTRGTERR0	AUXTSTRIG	TSTARGT0	TSSOVF
	リセット値			0	0	0	0	0	0					0	0	0	0	0												0	0	0	0
0x0B24 - 0x0B2C	予約済み																																

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0B30	ETH_MACTxTSSNR	TXTSSMIS	TXTSSLO[30:0]																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B34	ETH_MACTxTSSSR	TXTSSHI[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B38 - 0x0B3C	予約済み																																
0x0B40	ETH_MACACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ATSEN3	ATSEN2	ATSEN1	ATSEN0	Res.	Res.	Res.	ATSFC
	リセット値																									0	0	0	0				0
0x0B44	予約済み																																
0x0B48	ETH_MACATSNR	Res.	AUXTSLO[30:0]																														
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B4C	ETH_MACATSSR	AUXTSHI[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B50	ETH_MACTSIACR	OSTIAC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B54	ETH_MACTSEACR	OSTEAC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B58	ETH_ MACTSICNR	TSIC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B5C	ETH_ MACTSECNR	TSEC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B60 - 0x0B6C	予約済み																																
0x0B70	ETH_MACPPSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGTMODSEL0[1:0]		PPSEN0		PPSCTRL_PPSCMD[3:0]		
	リセット値																										0	0	0	0	0	0	0
0x0B74 - 0x0B7C	予約済み																																
0x0B80	ETH_ MACPPSTTSR	TSTRH0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B84	ETH_ MACPPSTTNR	TRGTBUSY0	TTSL0[30:0]																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 545. Ethernet MAC レジスタマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0B88	ETH_MACPPSIR	PPSINT0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B8C	ETH_MACPPSWR	PPSWIDTH0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B90 - 0x0BBC	予約済み																																
0x0BC0	ETH_MACPOCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DN[7:0]								PDRDIS	DRRDIS	APDREQTRIG	ASYNCTRIG	Res.	APDREQEN	ASYNCEN	PTOEN
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x0BC4	ETH_MACSPI0R	SPI0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0BC8	ETH_MACSPI1R	SPI1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0BCC	ETH_MACSPI2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPI2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0BD0	ETH_MACLMIR	LMPDRI[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DRSYNCR[2:0]			LSI[7:0]							
	リセット値	0	0	0	0	0	0	0	0														0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*104 ページのセクション 2.2.2*を参照してください。

参考資料

59 HDMI-CEC コントローラ（HDMI-CEC）

59.1 概要

Consumer Electronics Control（CEC）は HDMI（高解像度マルチメディアインタフェース）規格の一部です（*付録 1*）。多様なオーディオビジュアル製品に高品位な制御機能を提供するプロトコルで構成されています。CEC はロースピード、かつ最小限の処理とメモリのオーバヘッドで動作します。

HDMI-CEC コントローラでは、このプロトコルに対するハードウェアサポートを提供しています。

59.2 HDMI-CEC コントローラの主な機能

- HDMI-CEC v1.4 仕様に準拠
- 独立した 32 kHz CEC カーネル（*RCC カーネルクロック分配*のセクションを参照）
- STOP モードの超低電力アプリケーションで動作
- 送信開始前に設定可能な信号フリータイム
  - CEC 状態と送信履歴に従って、ハードウェアによって自動的に実行される
  - ソフトウェアによって固定化（7 つのタイミングオプション）
- 設定可能なペリフェラルアドレス（OAR）
- リッスンモードをサポート
  - CEC ラインに干渉することなく、OAR とは異なる転送先アドレスに送信された CEC メッセージの受信を有効化
- 設定可能な Rx 許容誤差マージン
  - 標準の許容誤差
  - 拡張された許容誤差
- 受信 – エラー検出
  - ビット立ち上がりエラー（BRE）、オプションによる受信の停止（BRESTP）
  - ショートビット周期エラー（SBPE）
  - ロングビット周期エラー（LBPE）
- 設定可能なエラービット生成
  - BRE 検出（BREGEN）
  - LBPE 検出（LBPEGEN）
  - 常に SBPE 検出で生成される
- 送信エラー検出（TXERR）
- アービトレーション喪失検出（ARBLST）
  - 自動送信を再試行
- 送信アンダーラン検出（TXUDR）
- 受信オーバーラン検出（RXOVR）

参考資料

59.3 HDMI-CEC の機能詳細

59.3.1 HDMI-CEC ピン および内部信号

CEC バスは、デバイスからデータを送受信するために使用される 1 本の双方向ラインで構成されています。このバスは、27 kΩ プルアップレジスタを介して、+3.3 V 電源電圧に接続されています。ワイヤード AND 接続を使用するには、デバイスの出力段はオープンドレインまたはオープンコレクタである必要があります。

HDMI-CEC コントローラでは、CEC 双方向ラインをオルタネート機能オープンドレインとして設定されたものとみなし、標準 GPIO のオルタネート機能として管理します。27 kΩ プルアップを外部的に STM32 に追加する必要があります。

アプリケーションの電源をオフにした際に CEC バスと干渉しないよう、このような条件ではバスから CEC ピンを離す必要があります。これは、*図 797* に示すような MOS トランジスタを使用して実行できます。

表 547 は HDMI-CEC と他の機能ブロック（RCC、EXTI など）の間で交換される内部信号のリストです。

表 546. HDMI ピン

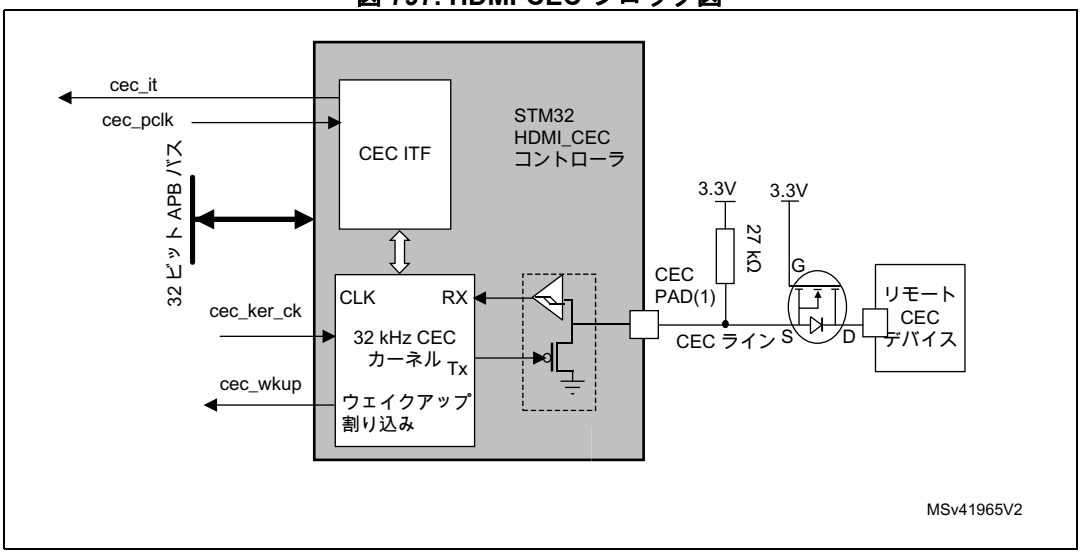
名前	信号タイプ	説明
CEC	双方向	2 つの状態： 1 = ハイインピーダンス 0 = ローインピーダンス 27 kΩ を外部的に追加する必要があります。

表 547. HDMI-CEC 内部入力／出力信号

信号名	信号タイプ	説明
cec_wkup	デジタル出力	HDMI-CEC ウェイクアップ信号
cec_it	デジタル出力	HDMI-CEC 割り込み信号
cec_pclk	デジタル入力	APB クロック
cec_ker_ck	デジタル入力	HDMI-CEC カーネルクロック

59.3.2 HDMI-CEC ブロック図

図 797. HDMI-CEC ブロック図



1. 出力オープンドレインのオルタネート機能として設定された GPIO。
2. 出力オープンドレインのオルタネート機能として設定された場合、シュミットトリガは有効なままとなります。

59.3.3 メッセージの説明

CEC ライン上のすべてのトランザクションは、イニシエータと 1 つまたは複数のフォロワで構成されています。イニシエータは、メッセージの構造とデータを送信します。フォロワは、すべてのデータを受信し、すべての確認応答ビットを設定します。

メッセージは単一フレームで送信されます。このフレームは、開始ビットとそれに続くヘッダブロックによって構成され、これにオプションとして OP コードとオペランドブロックの変数を含めることもできます。

これらのすべてのブロックは 8 ビットペイロードによって作成され（最上位ビットが最初に送信される）、続けてメッセージの終了（EOM）ビット、および確認応答（ACK）ビットによって作成されます。

EOM ビットはメッセージの最後のブロックにセットされ、その他のブロックではリセットされます。EOM が示された後でメッセージに追加のブロックが含まれている場合、このブロックは無視されます。EOM ビットは、他のデバイスがアクティブであることを確認するために、ヘッダブロックにセットして他のデバイスを「ピング」することができます。

確認応答ビットは、イニシエータによって常にハイインピーダンスにセットされています。これは、ヘッダの自己アドレスを読み出したフォロワ、またはブロードキャストメッセージを拒否する必要があるフォロワが、ローに駆動できるようにするためです。

ヘッダは、ソース論理アドレスフィールドおよび転送先論理アドレスフィールドで構成されています。ブロードキャストメッセージには、特殊アドレス 0xF が使用されている点に注意してください。

図 798. メッセージの構造

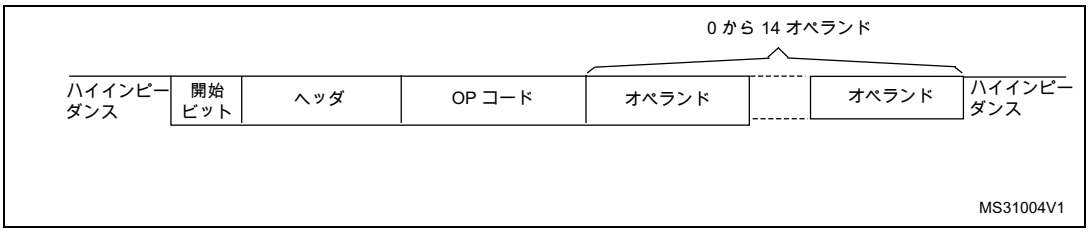
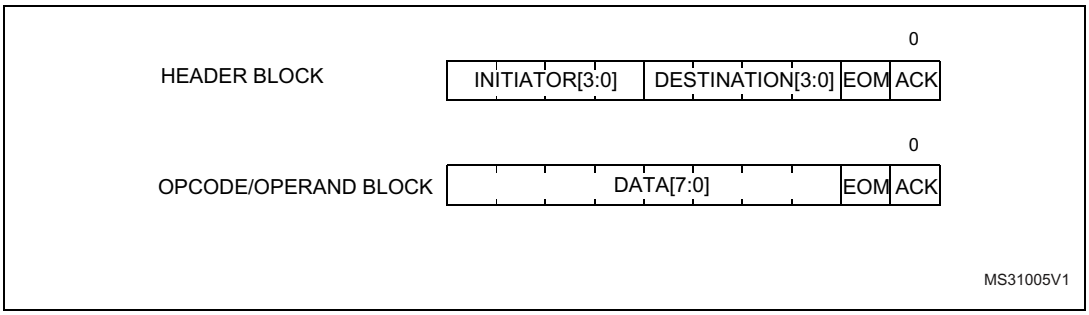


図 799. ブロック

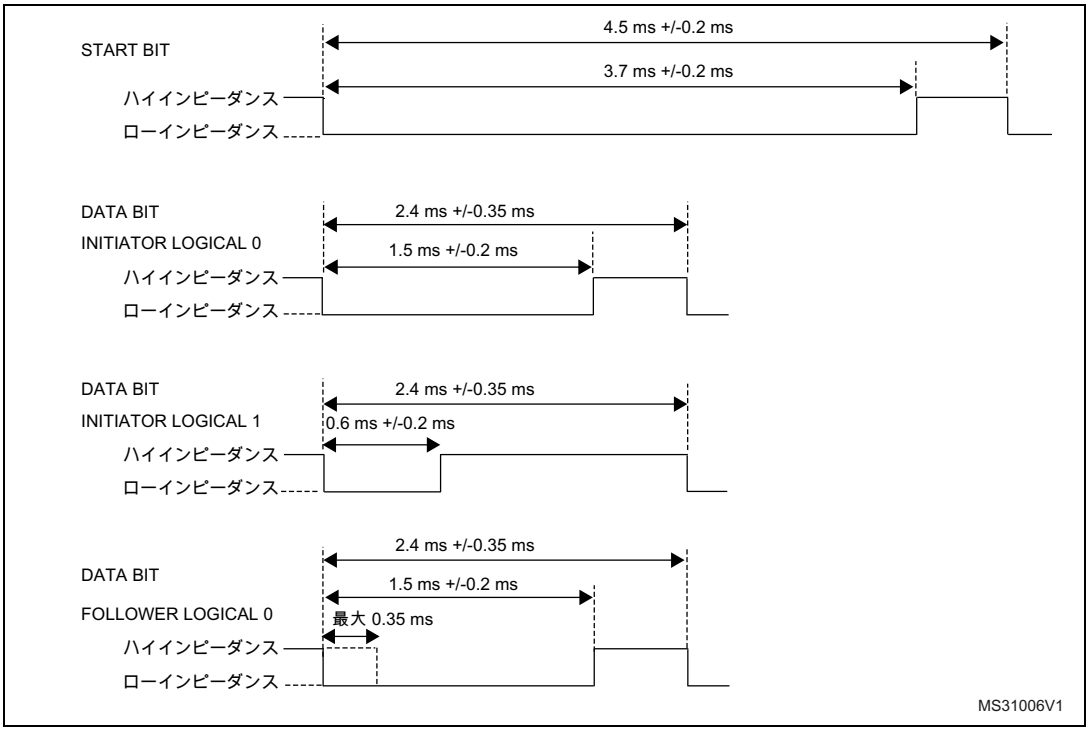


59.3.4 ビットタイミング

開始ビットの形式は一意で、メッセージの始まりを識別します。ロー時間および合計時間で検証されます。

メッセージの開始ビット後の残りのすべてのデータビットのタイミングは一定です。データビットの終わりのハイからローへの遷移は、CEC ラインがハイのままとなる最終ビットを除き、次のデータビットの始まりです。

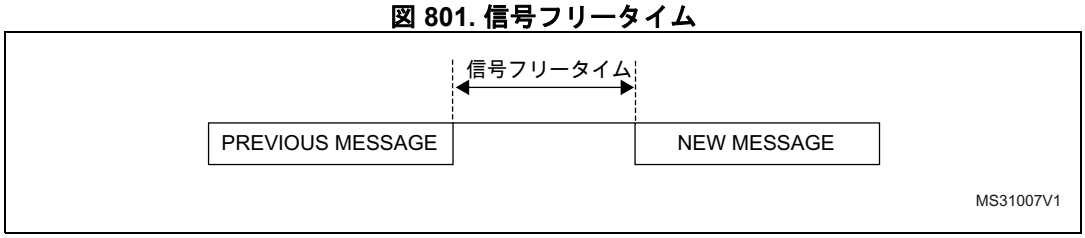
図 800. ビットタイミング



59.4 アービトレーション

CEC ラインにメッセージを送信または再送信する必要のあるすべてのデバイスは、いくつかのビット周期において無効であったことを確認する必要があります。信号フリータイムは、前のフレームの最終ビットからの開始時間として定義され、開始するデバイスと次の図に示す現在のステータスに依存します。

図 801. 信号フリータイム



1 度に使用できるイニシエータは 1 つのみであるため、同時に複数のイニシエータが送信を開始する場合に競合を避けるために、アービトレーション機構が提供されています。

CEC ラインアービトレーションは、開始ビットの先端から開始し、ヘッダブロック内のイニシエータアドレスビットの最後まで続けられます。この周期中、イニシエータは CEC ラインを監視し、このラインをハイインピーダンス状態に駆動する際に 0 に読み戻すと、アービトレーションを喪失したとみなして送信を停止し、フォロワになります。

図 802. アービトレーションフェーズ

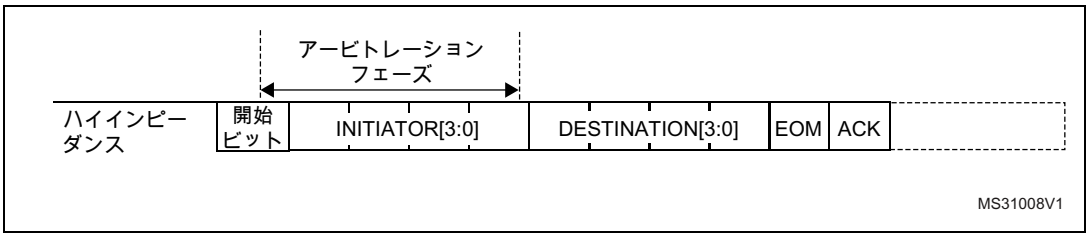
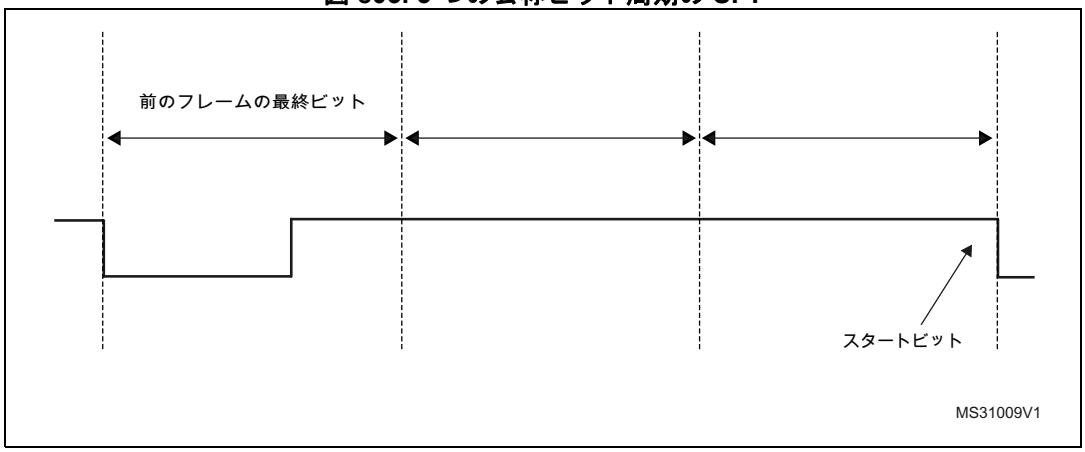


図 803 は、3 つの公称ビット周期の SFT の例を表します。

図 803. 3 つの公称ビット周期の SFT



送信を開始する前に、設定可能な時間枠がカウントされます。

SFT=0x0 設定では、HDMI-CEC デバイスが自動 SFT 計算を実行し、HDMI-CEC 規格に準拠していることを保証します。

- CEC が、送信が失敗した最後のバスイニシエータである場合、2.5 データビット周期
- CEC が新しいバスイニシエータである場合、4 データビット周期
- CEC が、送信が成功した最後のバスイニシエータである場合、6 データビット周期

これは、失敗した送信に最上位の優先順位、成功した送信の最後のイニシエータに最下位の優先順位を保証するために実行されます。

それ以外の場合は、固定されたタイミング値をカウントするために、SFT ビットを設定できる場合があります。可能な値は、0.5、1.5、2.5、3.5、4.5、5.5、6.5 データビット周期です。

59.4.1 SFT オプションビット

SFTOPT=0 の設定の場合、ソフトウェアによって送信開始コマンドがセットされると SFT のカウントが開始されます（TXSOM=1）。

SFTOPT=1 の場合、バスアイドル条件またはラインエラー条件が検出されると HDMI-CEC デバイスによって自動的に SFT のカウントが開始されます。TXSOM コマンドのセット時に SFT タイマが完了すると、送信は遅延なくただちに開始されます。SFT タイマが実行されたままである場合、システムは送信が開始される前にタイマが経過するまで待ちます。

SFTOPT=1 の場合、SFT タイマを開始させるバスイベント条件が、以下の場合に検出されます。

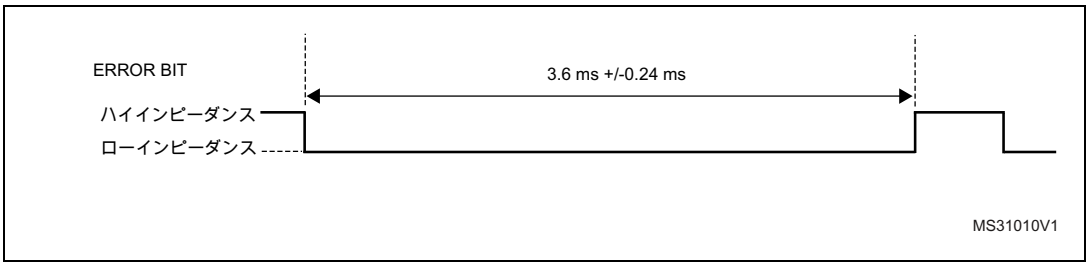
- 通常の送信／受信終了時に、TXEND/RXEND ビットはメッセージの最後のビット（ACK ビット）の最小公称データビット時間にセットされます。
- 送信エラーが検出された場合、TXERR 送信エラー検出時に SFT タイマが開始されます（TXERR=1）。
- CEC フォロワからの確認応答を喪失した場合、TXACKE ビットがセットされたとき、つまり ACK ビットの公称サンプリング時間に、SFT タイマが開始されます。
- 送信アンダーランエラーが発生した場合、ACK ビットの終了時に TXUDR ビットをセットすると SFT タイマが開始されます。
- 受信のアボートを意味する受信エラーが検証された場合、SFT タイマはエラー検出と同じ時間に開始されます。エラービットが生成された場合、エラービットの終了時に SFT のカウントが開始されます。
- 開始ビットに誤りがある場合や、アイドル状態からコード化されていないローインピーダンスバス状態のいずれかである場合、SFT タイマはバスがハイインピーダンスアイドル状態から戻るとすぐに再開されます。

59.5 エラー処理

59.5.1 ビットエラー

開始ビットを除くデータビットが無効であるとみなされる場合は、フォロワによってそのようなエラーが通知されることが期待されます。これには、公称データビット周期の 1.4 から 1.6 倍の CEC ライン上でロービット周期（つまり、公称 3.6 ms）を生成して実行されます。

図 804. エラービットタイミング



59.5.2 メッセージエラー

メッセージは喪失したものとみなされ、以下の条件により再送信することができます。

- メッセージは、直接アドレス指定されたメッセージでは確認応答されません。
- メッセージは、ブロードキャストメッセージでは否定応答されます。
- 予期しないローインピーダンスが、CEC ライン上で検出されました（ラインエラー）。

CEC インタフェースがデータビットを受信する際に、3 種類のエラーフラグを検出できます。

59.5.3 ビット立ち上がりエラー（BRE）

ビット立ち上がりエラー（BRE）：想定したウィンドウの外にビット立ち上がりエッジが検出されるとセットされます（*図 805*を参照）。BRE フラグは BREIE=1 の場合にも CEC 割り込みを生成します。

BRE 検出の場合、BRESTP ビット値によってメッセージの受信を停止することができます。BREGEN ビットをセットすると、エラービットが生成されます。

BRESTP=1 のブロードキャストメッセージで BRE が検出されると、BREGEN=0 の場合でもエラービットが生成され、イニシエータによって失敗した送信の再試行が強制されます。エラービット生成は、BREGEN=0、BRDNOGEN=1 を設定することで無効化できます。

59.5.4 ショートビット周期エラー（SBPE）

SBPE は、ビット立ち下がりエッジが予想よりも早く検出された場合にセットされます（*図 805* を参照）。SBPE フラグは SBPEIE=1 の場合にも CEC 割り込みを生成します。

エラービットは、SBPE エラー検出時に常にライン上に生成されます。エラービットは、リッスンモードのみがセットされ（LSTN=1）、以下の条件を満たしている場合には SBPE 検出時に生成されません。

- 直接アドレス指定されたメッセージは、SBPE とともに受信されます。
- ブロードキャストメッセージは、SBPE AND BRDNOGEN=1 とともに受信されます。

59.5.5 ロングビット周期エラー（LBPE）

LBPE は、有効なウィンドウでビット立ち下がりエッジが検出されない場合にセットされます（*図 805* を参照）。LBPE フラグは LBPEIE=1 の場合にも CEC 割り込みを生成します。

LBPE は常に受信を停止し、LBPEGEN ビットがセットされるとライン上にエラービットが生成されます。

ブロードキャストメッセージで LBPE が検出されると、LBPEGEN=0 の場合でもエラービットが生成され、イニシエータによって失敗した送信の再試行が強制されます。エラービット生成は、LBPEGEN=0、BRDNOGEN=1 を設定することで無効化できます。

注： ***BREGEN=1、BRESTP=0 設定を避ける必要があります。***

図 805. エラー処理

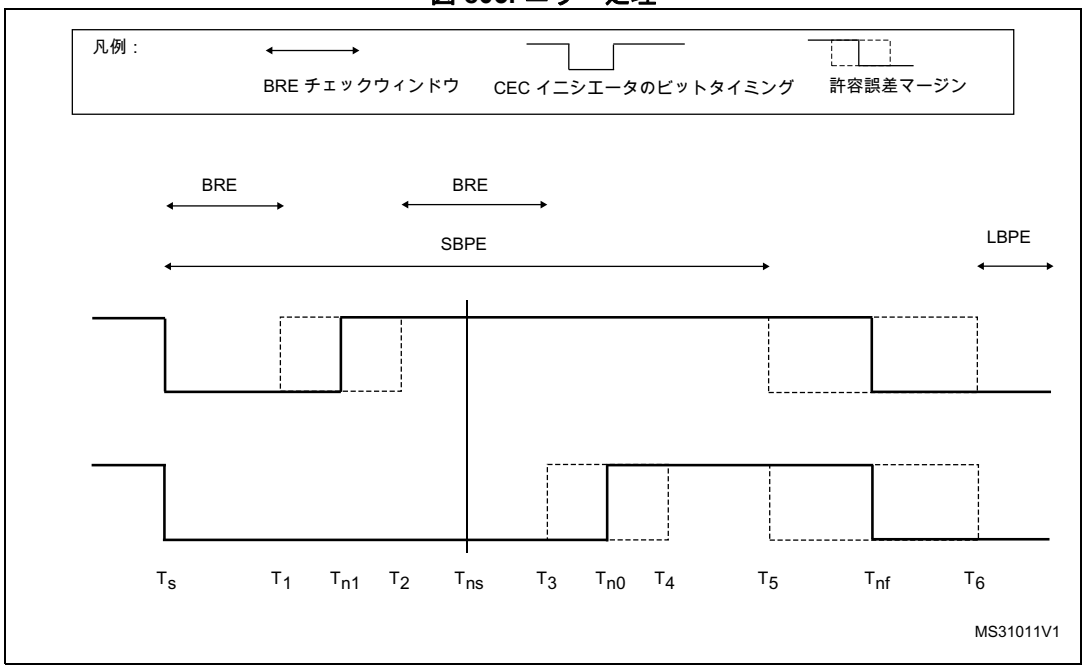


表 548. エラー処理タイミングパラメータ

時間	RXTOL	ms	説明
T_s	x	0	ビット開始イベント。
T_1	1	0.3	論理 1 を示すためにローからハイに遷移する一番早い時間。
	0	0.4	
T_{n1}	x	0.6	論理 1 を示すためにローからハイに遷移する公称時間。
T_2	0	0.8	論理 1 を示すためにローからハイに遷移する一番遅い時間。
	1	0.9	
T_{ns}	x	1.05	公称サンプリング時間。
T_3	1	1.2	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番早い時間。
	0	1.3	
T_{n0}	x	1.5	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される公称時間。
T_4	0	1.7	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番遅い時間。
	1	1.8	
T_5	1	1.85	後続のビットが開始される一番早い時間。
	0	2.05	
T_{nf}	x	2.4	公称データビット周期。
T_6	0	2.75	後続のビットが開始される一番遅い時間。
	1	2.95	

59.5.6 送信エラー検出（TXERR）

CEC イニシエータでは、ハイインピーダンスの送信中でフォロワがビットをアサートしないと想定される場合に、CEC ラインでローインピーダンスを検出すると、TXERR フラグをセットします。TXERR フラグは TXERRIE=1 の場合にも CEC 割り込みを生成します。

TXERR のアサートによってメッセージの送信が停止されます。アプリケーションでは、失敗した送信を最大 5 回まで再試行することができます。

TXERR のチェックは、さまざまな CEC ラインの状態または RX 許容誤差設定によって、異なる方法で実行されます。

図 806. TXERR 検出

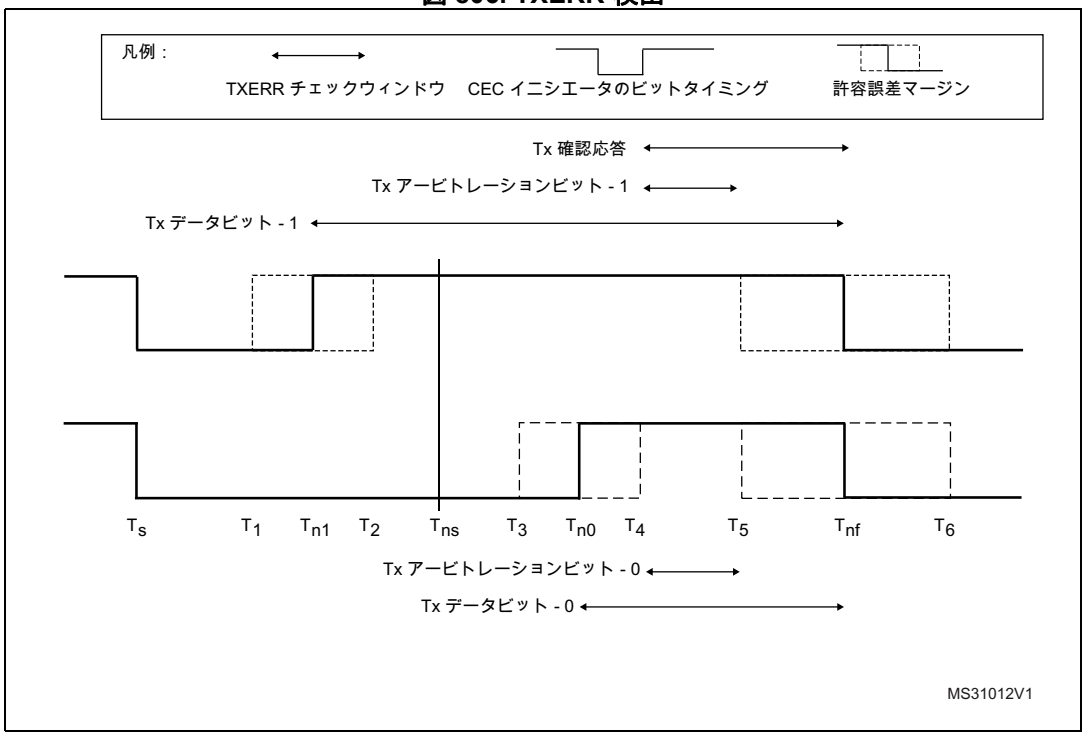


表 549. TXERR タイミングパラメータ

時間	RXTOL	ms	説明
T_s	x	0	ビット開始イベント。
T_1	1	0.3	論理 1 を示すためにローからハイに遷移する一番早い時間。
	0	0.4	
T_{n1}	x	0.6	論理 1 を示すためにローからハイに遷移する公称時間。
T_2	0	0.8	論理 1 を示すためにローからハイに遷移する一番遅い時間。
	1	0.9	
T_{ns}	x	1.05	公称サンプリング時間。
T_3	1	1.2	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番早い時間。
	0	1.3	

表 549. TXERR タイミングパラメータ（続き）

時間	RXTOL	ms	説明
T_{n0}	x	1.5	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される公称時間。
T_4	0	1.7	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番遅い時間。
	1	1.8	
T_5	1	1.85	後続のビットが開始される一番早い時間。
	0	2.05	
T_{nf}	x	2.4	公称データビット周期。
T_6	0	2.75	後続のビットが開始される一番遅い時間。
	1	2.95	

59.6 HDMI-CEC 割り込み

次の場合に割り込みを生成できます。

- 受信ブロック転送が完了または受信エラーが発生した場合の受信時。
- 送信ブロック転送が完了または送信エラーが発生した場合の送信時。

表 550. HDMI-CEC 割り込み

割り込みイベント	イベントフラグ	イネーブル制御ビット
Rx バイト受信	RXBR	RXBRIE
受信終了	RXEND	RXENDIE
Rx オーバーラン	RXOVR	RXOVRIE
Rx ビット立ち上がりエラー	BRE	BREIE
Rx ショートビット周期エラー	SBPE	SBPEIE
Rx ロングビット周期エラー	LBPE	LBPEIE
Rx 喪失確認応答エラー	RXACKE	RXACKEIE
アービトレーション喪失	ARBLST	ARBLSTIE
Tx バイトリクエスト	TXBR	TXBRIE
送信終了	TXEND	TXENDIE
Tx バッファアンダーラン	TXUDR	TXUDRIE
Tx エラー	TXERR	TXERRIE
Tx 喪失確認応答エラー	TXACKE	TXACKEIE

59.7 HDMI-CEC レジスタ

レジスタの説明で使用されている略語のリストについては、*96 ページのセクション 1.1* を参照してください。

59.7.1 CEC 制御レジスタ（CEC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TXEOM**：Tx メッセージの終了

TXEOM ビットは、CEC メッセージの最後のバイトの送信を指示するためにソフトウェアによってセットされます。

TXEOM は、TXSOM の場合と同時に、同じ条件下でハードウェアによってクリアされます。

0：TXDR データバイトは EOM=0 で送信されます。

1：TXDR データバイトは EOM=1 で送信されます。

注： ***TXEOM は、CECEN=1 のときにセットする必要があります。***

TXEOM は、TXDR に送信データを書き込む前にセットする必要があります。

TXSOM=0 のときに TXEOM がセットされると、送信メッセージは 1 バイト（HEADER）のみで構成されます（PING メッセージ）。

ビット 1 **TXSOM**：Tx メッセージの開始

TXSOM は、CEC メッセージの最初のバイトの送信を指示するためにソフトウェアによってセットされます。CEC メッセージが 1 バイトのみで構成される場合、TXEOM は TXSOM の前にセットする必要があります。

スタートビットは、SFT のカウント後に CEC ライン上で効果的に開始されます。メッセージの受信中に TXSOM がセットされると、送信は受信終了後に開始されます。

TXSOM は、送信アンダーラン（TXUDR=1）、否定応答（TXACKE=1）、および送信エラー（TXERR=1）が発生したときに、メッセージの最後のバイトが肯定応答（TXEND=1）とともに送信された後で、ハードウェアによってクリアされます。また、CECEN=0 でもクリアされます。アービトレーション喪失（ARBLST=1）の場合はクリアされず、送信は自動的に再試行されます。

TXSOM は、保留中や実行中の送信リクエストがあるかどうかをアプリケーションに知らせる、ステータスビットとしても使用できます。アプリケーションは、CECEN ビットをクリアすることで送信リクエストをいつでもアボートできます。

0：実行中の CEC 送信はありません。

1：CEC 送信コマンド。

注： TXSOM は、CECEN=1 のときにセットする必要があります。

TXSOM は、送信データを TXDR で使用できる場合にセットする必要があります。

自己のペリフェラルアドレスを含む HEADER の最初の 4 つのビットは、受信用としてのみ使用される CEC_CFGR.OAR ではなく、TXDR[7:4] から取得されます。

ビット 0 **CECEN**：CEC 有効

CECEN ビットは、ソフトウェアによってセット／クリアされます。CECEN=1 の場合にメッセージの受信が開始され、TXSOM 制御が有効化されます。CECEN=0 の場合に CEC ペリフェラルが無効化され、CEC_CR レジスタのすべてのビットがクリアされ、実行中のすべての送受信がアボートされます。

0：CEC ペリフェラルはオフです。

1：CEC ペリフェラルはオンです。

59.7.2 CEC 設定レジスタ（CEC_CFGR）

このレジスタは HDMI-CEC コントローラの設定に使用されます。

アドレスオフセット：0x04

リセット値：0x0000 0000

注意： CECEN=0 の場合のみ、CEC_CFGR に書き込む必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LSTN	OAR[14:0]														
rw	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SFT OPT	BRDN OGEN	LBPE GEN	BRE GEN	BRE STP	RX TOL	SFT[2:0]		
							rw	rw	rw	rw	rw	rw	rw		

ビット 31 **LSTN**：リッスンモード

LSTN ビットは、ソフトウェアによってセット／クリアされます。

0：CEC ペリフェラルでは、自己アドレス（OAR）に宛てられたメッセージのみを受信します。他の転送先に宛てられたメッセージは無視されます。ブロードキャストメッセージは常に受信されます。

1：CEC ペリフェラルでは、肯定応答とともに自己アドレス（OAR）に宛てられたメッセージを受信します。他の転送先に宛てられたメッセージは受信されますが、CEC バスに干渉することはありません。確認応答は送信されません。

ビット 30:16 **OAR**：自己アドレス設定

OAR ビットは、受信モードで考慮する必要がある転送先論理アドレスを選択するために、ソフトウェアによってセットされます。各ビットをセットすると、特定のビット位置によって指定された CEC 論理アドレスが有効になります。

HEADER 受信終了時に、受信した転送先アドレスが有効化されたアドレスと比較されます。アドレスが一致する場合は、受信メッセージが確認応答され、受信されます。アドレスが一致しない場合は、受信メッセージはリッスンモード（LSTN=1）でのみ受信され、確認応答は送信されません。ブロードキャストメッセージは常に受信されます。

例：

OAR = 0b000 0000 0010 0001 は、アドレス 0x0 および 0x5 が CEC で確認応答されることを意味します。結果として、これらのアドレスのいずれかにそれぞれ宛てられたメッセージが受信されます。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SFTOP**：*SFT オプションビット*

SFTOPT ビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェアによって TXSOM がセットされると、SFT タイマが開始されます。

1：メッセージの送受信の終了時に SFT タイマが自動的に開始されます。

ビット 7 **BRDNOGEN**：*ブロードキャストでの生成*エラービットの回避

BRDNOGEN ビットは、ソフトウェアによってセット／クリアされます。

0：BRESTP=1 および BREGEN=0 のブロードキャストメッセージで BRE が検出されると、CEC ライン上にエラービットが生成されます。LBPEGEN=0 のブロードキャストメッセージで LBPE が検出されると、CEC ライン上にエラービットが生成されます。

1：エラービットは、上記と同じ条件では生成されません。リッスンモードがセットされている場合、ブロードキャストメッセージで SBPE が検出されてもエラービットは生成されません。

ビット 6 **LBPEGEN**：ロング*ビット周期エラー*でのエラービットの生成

LBPEGEN ビットは、ソフトウェアによってセット／クリアされます。

0：LBPE 検出で、CEC ライン上にエラービットは生成されません。

1：LBPE 検出で、CEC ライン上にエラービットが生成されます。

注： ***BRDNOGEN=0 の場合、LBPEGEN=0 の場合でもブロードキャストでは LBPE 検出時にエラービットが生成されます。***

ビット 5 **BREGEN**：*ビット立ち上がりエラー*でのエラービットの生成

BREGEN ビットは、ソフトウェアによってセット／クリアされます。

0：BRE 検出で、CEC ライン上にエラービットは生成されません。

1：BRE 検出で、CEC ライン上にエラービットが生成されます（BRESTP がセットされている場合）。

注： ***BRDNOGEN=0 の場合、BREGEN=0 の場合でもブロードキャストで BRESTP=1 の BRE 検出時にエラービットが生成されます。***

ビット 4 **BRESTP**：*ビット立ち上がりエラーでの Rx 停止*

BRESTP ビットは、ソフトウェアによってセット／クリアされます。

0：BRE 検出で CEC メッセージの受信が停止されることはありません。データビットは 1.05 ms でサンプリングされます。

1：BRE 検出でメッセージの受信が停止されます。

ビット 3 **RXTOL**：Rx 許容誤差

RXTOL ビットは、ソフトウェアによってセット／クリアされます。

0：標準許容誤差マージン

– スタートビット：+/- 200 µs 立ち上がり、+/- 200 µs 立ち下がり

– データビット：+/- 200 µs 立ち上がり、+/- 350 µs 立ち下がり

1：拡張された許容誤差

– スタートビット：+/- 400 µs 立ち上がり、+/- 400 µs 立ち下がり

– データビット：+/- 300 µs 立ち上がり、+/- 500 µs 立ち下がり

ビット 2:0 **SFT**：*信号フリータイム*

SFT ビットは、ソフトウェアによってセットされます。SFT=0x0 設定では、送信前に待った公称データビット周期数は、送信履歴に従って、ハードウェアによって支配されます。その他の設定では、SFT 番号はソフトウェアによって決定されます。

- 0x0
  - CEC が、送信が失敗した最後のバスイニシエータである場合、2.5 データビット周期（ARBLST=1、TXERR=1、TXUDR=1、または TXACKE= 1）
  - CEC が新しいバスイニシエータである場合、4 データビット周期
  - CEC が、送信が成功した最後のバスイニシエータである場合、6 データビット周期（TXEOM=1）
- 0x1：0.5 公称データビット周期
- 0x2：1.5 公称データビット周期
- 0x3：2.5 公称データビット周期
- 0x4：3.5 公称データビット周期
- 0x5：4.5 公称データビット周期
- 0x6：5.5 公称データビット周期
- 0x7：6.5 公称データビット周期

59.7.3 CEC Tx データレジスタ（CEC_TXDR）

アドレスオフセット：0x8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXD[7:0]							
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXD[7:0]**：Tx データレジスタ

TXD は、送信されるデータバイトを含む書き込み専用のレジスタです。

注： TXD は、TXSTART=1 のときに書き込む必要があります。

59.7.4 CEC Rx データレジスタ（CEC_RXDR）

アドレスオフセット：0xC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXD[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXD[7:0]**：Rx データレジスタ

RXD は読み出し専用で、CEC ラインから受信した最後のデータバイトを含みます。

59.7.5 CEC 割り込みおよびステータスレジスタ（CEC_ISR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Tx ACKE	Tx ERR	Tx UDR	Tx END	TXBR	ARB LST	RX ACKE	LBPE	SBPE	BRE	RX OVR	RX END	RXBR
			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TXACKE**：Tx 喪失確認応答エラー

送信モードでは、TXACKE はハードウェアによってセットされ、アプリケーションに確認応答を受信していないことを知らせます。ブロードキャスト送信の場合、TXACKE はアプリケーションに否定応答を受信したことを知らせます。TXACKE はメッセージの送信をアボートし、TXSOM および TXEOM 制御をクリアします。

TXACKE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 11 **TXERR**：Tx エラー

送信モードでは、CEC イニシエータがリリース中に CEC ラインでローインピーダンスを検出した場合に、ハードウェアによって TXERR がセットされます。TXERR はメッセージの送信をアボートし、TXSOM および TXEOM 制御をクリアします。

TXERR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 10 **TXUDR**：Tx バッファアンダーラン

送信モードでは、次のバイト送信前の TXDR のロードにアプリケーションが間に合わなかった場合に、ハードウェアによって TXUDR がセットされます。TXUDR はメッセージの送信をアボートし、TXSOM および TXEOM 制御ビットをクリアします。

TXUDR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 9 **TXEND**：送信終了

TXEND は、CEC メッセージの最後のバイトが正常に送信されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。TXEND は、TXSOM および TXEOM 制御ビットをクリアします。

TXEND はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 8 **TXBR**：Tx バイトリクエスト

TXBR は、次の送信データを TXDR に書き込む必要があることをアプリケーションに知らせるために、ハードウェアによってセットされます。TXBR は、現在送信されたバイトの 4 番目のビットが送信されたときにセットされます。アプリケーションでは、送信アンダーランエラー（TXUDR）が発生する前に、6 公称データビット周期内の TXDR に次のバイトを書き込む必要があります。

TXBR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 7 **ARBLST**：アービトレーション喪失

ARBLST は、TXSOM コマンドに続くアービトレーション喪失イベントによって CEC デバイスが受信に切り換わることをアプリケーションに知らせるために、ハードウェアによってセットされます。ARBLST の原因は、先に開始される、またはより高い HEADER 優先順位で同時に開始される、競合するいずれかの CEC デバイスです。ARBLST のアサート後、TXSOM ビットでは次の送信試行を保留のままとします。

ARBLST はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 6 **RXACKE**：Rx 喪失確認応答

受信モードでは、RXACKE はハードウェアによってセットされ、CEC ラインで確認応答を確認できなかったことをアプリケーションに知らせます。RXACKE はブロードキャストメッセージにのみ適用でき、またリッスンモードでは直接アドレス指定されていないメッセージにも適用できます（転送先アドレスは OAR では無効）。RXACKE はメッセージの受信をアボートします。

RXACKE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 5 **LBPE**：Rx ロングビット周期エラー

ロングビット周期エラーのデータビット波形が検出された場合、LBPE はハードウェアによってセットされます。立ち下がりエッジが存在している場合、LBPE は、RXTOL が許容する最大ビット拡張許容誤差の最後にセットされます。LBPE は、常に CEC メッセージの受信を停止します。LBPE で、LBPEGEN=1 のときに CEC ライン上にエラービットが生成されます。ブロードキャストの場合は、LBPEGEN=0 の場合でもエラービットが生成されます。

LBPE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 4 **SBPE**：Rx ショートビット周期エラー

ショートビット周期エラーのデータビット波形が検出された場合、SBPE はハードウェアによってセットされます。SBPE は、想定された立ち下がりエッジの発生時にセットされます。SBPE で、CEC ライン上にエラービットが生成されます。

SBPE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 3 **BRE**：Rx ビット立ち上がりエラー

ビット立ち上がりエラーのデータビット波形が検出された場合、BRE はハードウェアによってセットされます。立ち上がりエッジが存在している場合、BRE は、立ち上がりエッジの誤配置が発生したとき、または RXTOL が許容する最大 BRE 許容誤差の最後にセットされます。BRESTP=1 のとき、BRE ではメッセージの受信を停止します。BRE で、BREGEN=1 のときに CEC ライン上にエラービットが生成されます。

BRE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 2 **RXOVR**：Rx オーバーラン

新しいバイトの受信時に RXBR がクリアされていない場合や RXD に保存されていない場合、RXOVR はハードウェアによってセットされます。RXOVR アサートではメッセージの受信を停止するため、確認応答は送信されません。ブロードキャストの場合、否定応答が送信されます。

RXOVR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 1 **RXEND**：受信終了

RXEND は、CEC ラインから CEC メッセージの最後のバイトを受信し、RXD バッファに格納したことをアプリケーションに知らせるために、ハードウェアによってセットされます。RXEND は、RXBR と同じ時間でセットされます。

RXEND はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 0 **RXBR**：Rx バイト受信

RXBR ビットは、CEC ラインから新しいバイトを受信し、RXD バッファに格納したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

RXBR はソフトウェアで 1 を書き込むことによってクリアされます。

59.7.6 CEC 割り込み有効レジスタ（CEC_IER）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TXACKIE	TXERRIE	TxUDRIE	TXENDIE	TXBRIE	ARBLSTIE	RXACKIE	LBPEIE	SBPEIE	BREIE	RXOVRIE	RXENDIE	RXBRIE
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TXACKIE**：Tx 喪失確認応答エラー割り込み有効
TXACKEIE ビットは、ソフトウェアによってセット／クリアされます。
0：TXACKE 割り込み無効
1：TXACKE 割り込み有効

ビット 11 **TXERRIE**：Tx エラー割り込み有効
TXERRIE ビットは、ソフトウェアによってセット／クリアされます。
0：TXERR 割り込み無効
1：TXERR 割り込み有効

ビット 10 **TXUDRIE**：Tx アンダーラン割り込み有効
TXUDRIE ビットは、ソフトウェアによってセット／クリアされます。
0：TXUDR 割り込み無効
1：TXUDR 割り込み有効

ビット 9 **TXENDIE**：Tx メッセージの終了割り込み有効
TXENDIE ビットは、ソフトウェアによってセット／クリアされます。
0：TXEND 割り込み無効
1：TXEND 割り込み有効

ビット 8 **TXBRIE**：Tx バイトリクエスト割り込み有効
TXBRIE ビットは、ソフトウェアによってセット／クリアされます。
0：TXBR 割り込み無効
1：TXBR 割り込み有効

ビット 7 **ARBLSTIE**：アービトレーション喪失割り込み有効
ARBLSTIE ビットは、ソフトウェアによってセット／クリアされます。
0：ARBLST 割り込み無効
1：ARBLST 割り込み有効

ビット 6 **RXACKIE**：Rx 喪失確認応答エラー割り込み有効
RXACKIE ビットは、ソフトウェアによってセット／クリアされます。
0：RXACKE 割り込み無効
1：RXACKE 割り込み有効

ビット 5 **LBPEIE**：ロングビット周期エラー割り込み有効
LBPEIE ビットは、ソフトウェアによってセット／クリアされます。
0：LBPE 割り込み無効
1：LBPE 割り込み有効

ビット 4 **SBPEIE**：ショートビット周期エラー割り込み有効
SBPEIE ビットは、ソフトウェアによってセット／クリアされます。
0：SBPE 割り込み無効
1：SBPE 割り込み有効

ビット 3 **BREIE**：ビット立ち上がりエラー割り込み有効
BREIE ビットは、ソフトウェアによってセット／クリアされます。
0：BRE 割り込み無効
1：BRE 割り込み有効

ビット 2 **RXOVRIE**：Rx バッファオーバーラン割り込み有効
RXOVRIE ビットは、ソフトウェアによってセット／クリアされます。
0：RXOVR 割り込み無効
1：RXOVR 割り込み有効

ビット 1 **RXENDIE**：受信終了割り込み有効
RXENDIE ビットは、ソフトウェアによってセット／クリアされます。
0：RXEND 割り込み無効
1：RXEND 割り込み有効

ビット 0 **RXBRIE**：Rx バイト受信割り込み有効
RXBRIE ビットは、ソフトウェアによってセット／クリアされます。
0：RXBR 割り込み無効
1：RXBR 割り込み有効

注意： *(*) CECEN=0 の場合は、CEC_IER に書き込む必要があります。*

59.7.7 HDMI-CEC レジスタマップ

次の表に HDMI-CEC レジスタの一覧を示します。

表 551. HDMI-CEC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CEC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXEOM	TXSOM	CECEN
	リセット値																														0	0	0
0x04	CEC_CFGR	LSTN	OAR[14:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	SFTOPT	BRDNOGEN	LBPEGEN	BREGEN	BRESTP	RXTOL	SFT[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0	0	0	0
0x08	CEC_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXD[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0C	CEC_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXD[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x10	CEC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXACKE	TXERR	TXUDR	TXEND	TXBR	ARBLST	RXACKE	LBPE	SBPE	BRE	RXOVR	RXEND	RXBR
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	CEC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXACKIE	TXERRIE	TXUDRIE	TXENDIE	TXBRIE	ARBLSTIE	RXACKIE	LBPEIE	SBPEIE	BREIE	RXOVRIE	RXENDIE	RXBRIE
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては *104 ページのセクション 2.2.2* を参照してください。

参考資料

60 デバッグインフラストラクチャ

60.1 概要

ソフトウェア設計者は デバッグインフラストラクチャ を使用して組み込みソフトをデバッグおよびトレースすることができます。

デバッグ機能は JTAG／シリアルワイヤデバッグ用のアクセスポートを介し、業界標準のデバッグツールを使用して制御することができます。トレース用のポートを使用すると、ログおよび分析のためにデータをキャプチャすることができます。

トレース・デバッグシステムは、さまざまな典型的使用例に対応するよう設計されています。

- **低価格トレース**

  シングルワイヤデバッグの出力では、トレース機能の利用に制限があります。この機能は、「printf」を使用したコード実装、データやアドレスのウォッチポイントのトレース、割り込み検出、およびプログラムカウンタのサンプリングをサポートします。シングルワイヤトレースは、プロセッサが 電源オフまたはクロック停止している場合でも継続することができます。

- **ブレークポイントのデバッグ**

  プロセッサ は、JTAG/SWD デバッグポートに接続された機器を使用してデバッグすることができます。これにより、ブレークポイントとウォッチポイントの設定、コードのステップ実行、メモリアクセスなどが可能になります。

- **トレースポート経由での トレースコードの実行**

  トレース情報は、1つのトレースストリームに結合され、リアルタイムでトレースポートアナライザーに出力されます。トレースに埋め込まれた ID により、アナライザーは各情報パケットの転送元を特定できます。

- **サーキュラバッファ内でのトレースの連続キャプチャ**

  結合されたトレース情報は、チップ外にストリーミングする代わりに、チップ上のサーキュラバッファに保存することができます。トレースストレージの開始や停止は、デバッガのコマンド、ソフトウェアのコマンド、外部トリガ信号、内部イベントなどにより起動されます。

- **トレースポートへのバッファドレイン**

  保存したトレースはチップからトレースポートアナライザーへダンプすることができます。バッファドレインは、デバッガ、ソフトウェア、外部トリガ、内部イベントなどにより起動されます。

- **デバッガを使用したバッファ読み出し**

  デバッガは、デバッグポート経由でトレースバッファの内容を読み出すことができます。この読み出しはトレースポートより低速ですが、トレースポートアナライザーの費用をかけずにデバッガの基本的なトレース機能を提供できます。

- **保存されたトレースのソフトウェアによる分析**

  トレースバッファは プロセッサコア によって読み出すか、システムメモリに DMA 転送することができます。このようなパワフルな機能では、組み込みテストソフトウェアによって、コード実行のリアルタイム監視、故障の解析と特定、自動例外処理などが行われます。

- **保存されたトレースのアップロード**

  保存されたトレースは、MCU の数多くの通信インタフェース（USB、USART、SPI、I2C、Ethernet、CAN など）の中から 1 つを使用して、ホストマシンにアップロードすることもできます。特に、配備された製品のリモート監視や故障解析といったトレースポートにアクセスできない場合に有用です。

60.2 デバッグインフラストラクチャ 機能

以下に示すトレースとデバッグの機能を統合させたもので、ソフトウェア開発およびシステムインテグレーションをサポートするために提供されます。

- ブレークポイント を使用したデバッグ
- コード実行トレース
- ソフトウェア計測
- クロストリガ
- JTAG デバッグポート
- シリアルワイヤデバッグポート
- トリガ入出力
- シリアルワイヤトレースポート
- トレースポート
- ARM® CoreSight™ デバッグおよびトレースのコンポーネント

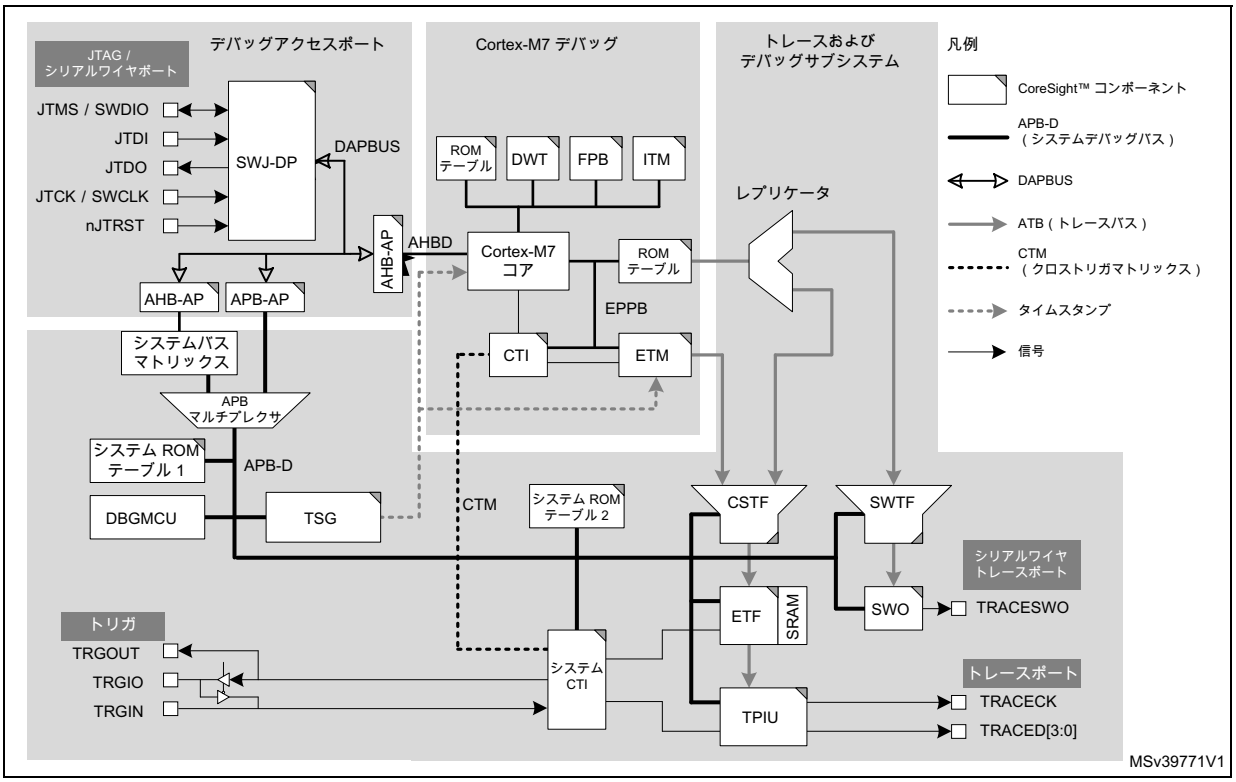
CoreSight コンポーネントについて、本書では概略の説明を提供しています。詳細情報は *セクション 60.7*に掲載されいている ARM® の資料をご覧ください。

60.3 デバッグインフラストラクチャ機能詳細

60.3.1 デバッグインフラストラクチャ ブロック図

以下のブロック図はデバッグインフラストラクチャの論理パーティションを示します。

図 807. デバッグインフラストラクチャのブロック図



60.3.2 デバッグインフラストラクチャ ピンおよび内部信号

表 552. JTAG／シリアルワイヤデバッグポートのピン

ピン名	JTAG デバッグポート		SW デバッグポート		ピン割り当て
	タイプ	説明	タイプ	説明	
JTMS／SWDIO	I	JTAG テストモード選択	IO	シリアルワイヤデータ入出力	PA13
JTCK／SWCLK	I	JTAG テストクロック	I	シリアルワイヤクロック	PA14
JTDI	I	JTAG テストデータ入力	-	-	PA15
JTDO	O	JTAG テストデータ出力	-	-	PB3
nJTRST	I	JTAG テストリセット	-	-	PB4

表 553. トレースポートピン

ピン名	タイプ	説明	ピン割り当て
TRACED0	O	トレース同期データ出力 0	データシート参照
TRACED1	O	トレース同期データ出力 1	
TRACED2	O	トレース同期データ出力 2	
TRACED3	O	トレース同期データ出力 3	
TRACECK	O	トレースクロック	

表 554. シリアルワイヤトレースポートのピン

ピン名	タイプ	説明	ピン割り当て
TRACESWO	O	シングルワイヤトレース非同期データ出力	PB3(1)

1. TRACESWO が JTDOとマルチプレクスされます。つまり、シングルワイヤトレースは JTAG ではなく、シリアルワイヤデバッグ用のインタフェースを使用する場合にしか使用できません。

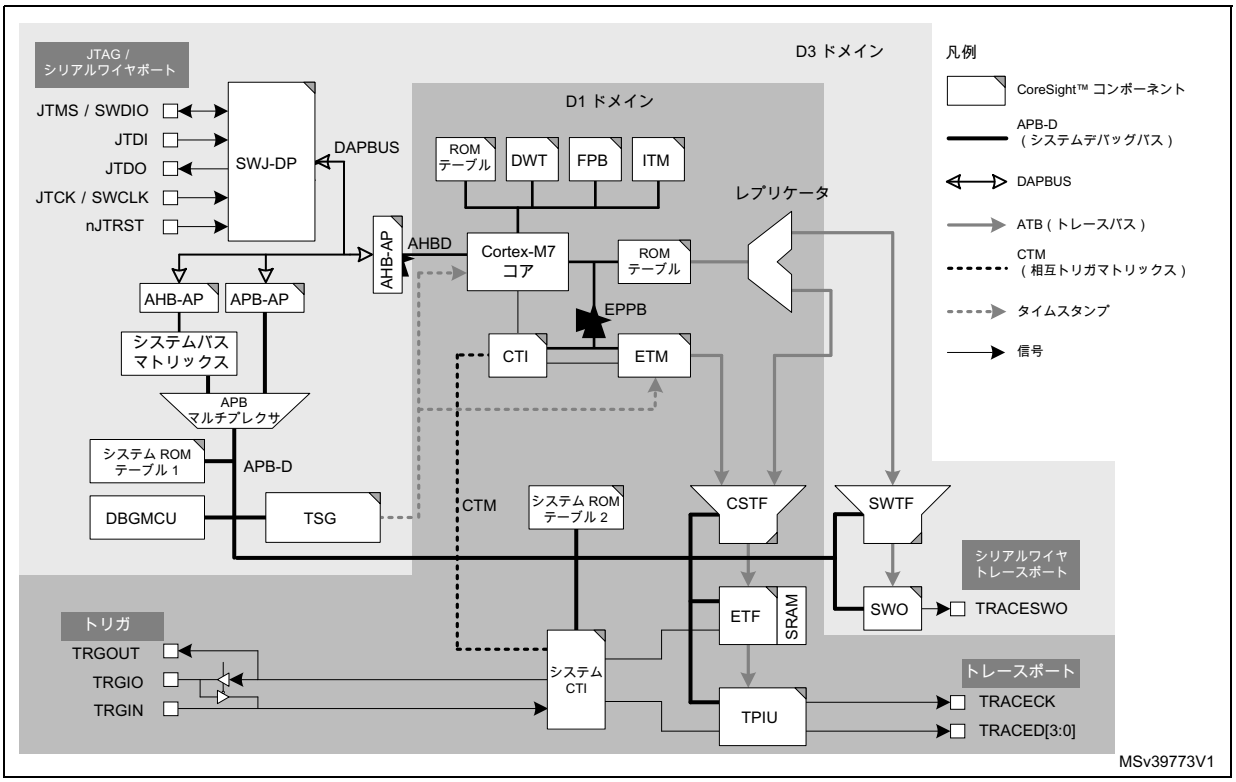
表 555. トリガピン

ピン名	タイプ	説明	ピン割り当て
TRGIN	I	外部トリガ入力	データシート参照
TRGOUT	O	外部トリガ出力	
TRGIO	IO	外部トリガ双方向(1)	

1. TRGIO は、DBGMCU の TRGOEN ビットによって、入力または出力として設定できます。入力として設定された場合は TRGIN に接続されます。出力として設定された場合は TRGOUT に接続されます。特定のパッケージでは TRGIN と TRGOUT を使用できないためです。

60.3.3 デバッグインフラストラクチャ 電力供給、クロック制御、およびリセット電力ドメイン

図 808. デバッグインフラストラクチャの電力ドメイン

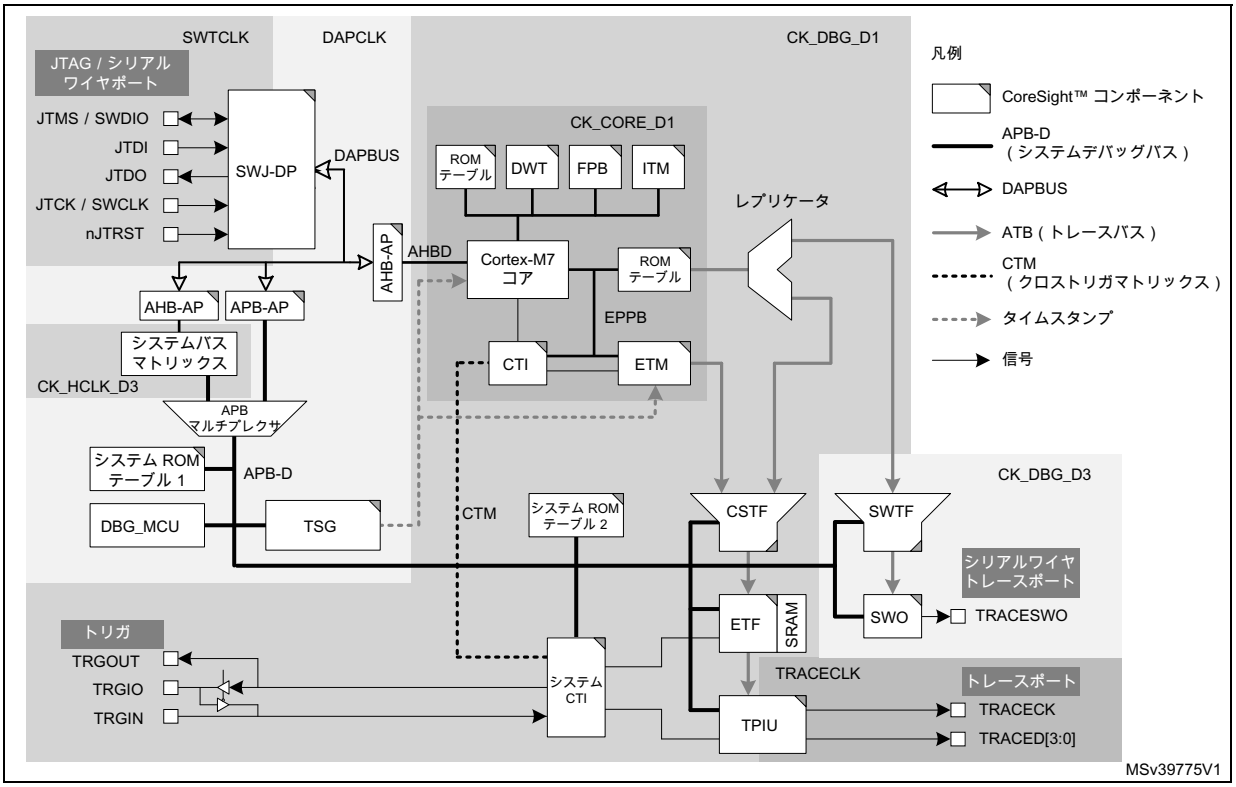


デバッグコンポーネントは電力ドメイン D1 から D3 までに分散しています。デバッガが接続されている場合、電力ドメイン D3 は常時オンとみなされます。したがって、デバッガは 他の電力ドメインが 電源オフの場合に SoC との接続が切れることがないように、SWJ-DP が内蔵しています。さらに、タイムスタンプジェネレータ、DBGMCU、およびシリアルワイヤトレース機能も内蔵しています。

電力ドメイン D1 には Cortex-M7 コアと関連のデバッグ・トレースコンポーネントが含まれます。また、APB-D 上にはシステムトレースコンポーネントも配置されています。したがって、Cortex-M7 へのデバッグアクセスが要求されるとき、あるいは プロセッサ のトレース機能がアクティブのときには、必ずこの電力ドメインはオンでなければなりません。

クロックドメイン

図 809. デバッグインフラストラクチャのクロックドメイン



デバッガは、デバッグインタフェースのピン、JTCK/SWCLK を介して、デバッガポート用のクロック SWTCLK を提供します。このクロックを使用して、シリアルワイヤおよびJTAGモード でシリアル入力データを登録し、同時にデバッグポートのステートマシンや内部ロジックを運用します。したがって、デバッグポートを確実にアイドルステートに戻すためには、クロックはアクセス終了後に数回のサイクルの間トグルし続ける必要があります。

SWJ-DP は DAPCLK ドメインとの非同期インタフェースを内蔵しており、このインタフェースは残りの SWJ-DP およびアクセスポートを範囲に含みます。DBGMCU、タイムスタンプジェネレータ、およびシステム ROM テーブル 1 も DAPCLK ドメインに含まれています。

CK_DBG_D3 は SWO とシリアルワイヤトレースのファネルをクロック供給します。

DAPCLK と CK_DBG_D3 はどちらも D3 ドメインシステムクロック（CK_HCLK_D3）のゲート付きバージョンです。

CK_DBG_D1 は、次に挙げる D1 電力ドメイン内のトレースコンポーネントをクロック供給します。システム ROM テーブル 2、CoreSight トレースファネル、ETF、システム CTI および TPIU。これは D1 ドメインシステムクロック（CK_HCLK_D1）のゲート付きバージョンです。

TRACECLK はトレースポート出力クロックで、PLL1 がシステムクロックソースである場合を除き、システムクロック（CK_SYS）のゲート付きバージョンです。この場合、TRACECLK を PLL1 VCO 出力から直接取り出し、3 で割ります。これは、プロセッサが 最高周波数 で 動作する 場合に、トレースポートの高いデータスループットをサポートするために必要です。

すべてのデバッグクロック（DAPCLK を除く）は、DBGMCU のレジスタビットで有効または無効にすることができます。DAPCLK ドメインは、デバッグポート CTRL/STAT レジスタの CDBGPWRUPREQE ビットを使用して、デバッガによって有効にできます。デバッガがデバイスのデバッグ機能のいずれか

にアクセスするには、先にクロックを有効にする必要があります。パワーアップ時やデバッガの切断時には、エネルギー節約のために無効にしてください。

プロセッサに内蔵されているデバッグおよびトレースコンポーネント（ETM ITM、DWG、FPB など）は、対応するコアクロック（CK_CORE_D1）が供給されます。

低電力モードでのデバッグ

デバイスには省電力機能が内蔵されており、必要ないときは電力ドメインごとに電源オフや停止ができるようになっています。電力ドメインが電源オフであるか、またはクロック供給されていない場合、デバッガへはドメイン内のどのデバッグコンポーネントからもアクセスできません。このような事態を防ぐために、省電力モードのエミュレーションが実装されています。あるドメインでエミュレーションが有効な場合、そのドメインは省電力モードに入るが、、そのクロックと電力が維持されます。言い換えれば、ドメインはあたかも省電力モードにいるかのように振る舞いながら、デバッガとの接続は切れません。

エミュレーションモードは、MCU デバッグ（DBGMCU）ユニットでプログラムされます。詳細については、*セクション 60.5.8*を参照してください。

デバッグインフラストラクチャのリセット

デバッグポートとアクセスポート以外のデバッグコンポーネントは、それぞれの電力ドメインリセットでリセットされます。デバッグポート（SWJ-DP）は、D3 ドメインのパワーオンリセットによってのみリセットされます。

60.4 デバッグアクセスポートの機能詳細

デバッグアクセスポート（DAP）は、シリアルワイヤ／JTAG デバッグポート（SWJ-DP）および 3 つのアクセスポートで構成されるデバッグサブシステムです。

60.4.1 シリアルワイヤ／JTAG デバッグポート（SWJ-DP）

SWJ-DP は CoreSight コンポーネントの 1 つで、デバッグ装置との接続用の外部アクセスポートを実装しています。

そのポートは次のように設定することができます。

- 5 ピンの標準 JTAG デバッグポート（JTAG-DP）
- 2 ピン（クロック + データ）のシリアルワイヤデバッグポート（SW-DP）

2 つのモードは同じ IO ピンを共有しているため、相互に排他的です。

デフォルトでは、JTAG-DP はシステムリセットまたはパワーオンリセット時に選択されます。5つの IO はデバッグ代替機能モードでハードウェアによって設定されます。SWJ-DP は、JTDI、JTMS/SWDIO、および nJTRST ラインにプルアップレジスタ、JTCK/SWCLK ラインにプルダウンレジスタを搭載しています。

デバッガはJTMS/SWDIO に次のようなシリアルデータシーケンスを送信することによって、SW-DP を選択することができます。

..., （1 が 50 個以上）,...,0,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1,..., （1 が 50 個以上）,...

JTCK/SWCLK はデータビットごとに循環させる必要があります。

SW-DP モードでは、使用していない JTAG ライン、JTDI、JTDO、および nJTRST を他の機能に使用することができます。

SWJ ポートのすべての IO はソフトウェアで他の機能に合わせて再設定できますが、その場合デバッグはできなくなります。

シリアルワイヤデバッグポート

シリアルワイヤデバッグプロトコルでは、次の 2 種類のピンを使用します。

- SWCLK：ホストからターゲットへのクロック
- SWDIO：双方向シリアルデータ（100 kΩ プルアップが必要）

シリアルデータはクロックと同期して LSBが先に転送されます。転送は以下の 3 フェーズで構成されます。

1. ホストによって送信されるパケットリクエスト（8 ビット）
2. ターゲットによって送信される確認応答（3 ビット）
3. ホスト（書き込みの場合）またはターゲット（読み出しの場合）によって送信されるデータ転送（33 ビット）

確認応答が OK である場合にのみ、データ転送が実行されます。

データ方向が逆転した場合、各フェーズの間に1 クロックサイクルのターンアラウンド時間が挿入されます。

表 556. パケットリクエスト

フィールドのビット	名前	説明
0	Start	”1”である必要があります。
1	APnDP	0：DP レジスタアクセス - DP レジスタのリストは *表 560* を参照のこと。 1：AP レジスタアクセス - *セクション 60.4.2*を参照のこと。
2	RnW	0：書き込みリクエスト 1：読み出しリクエスト
4:3	A[3:2]	DP/AP レジスタのアドレスフィールド（*表 560* および *表 561*を参照）
5	Parity	先行するビットの 1 ビットパリティ
6	STOP	0
7	Park	ホストによって駆動されません。ターゲットによって「1」と読み出される必要があります。

表 557. ACK 応答

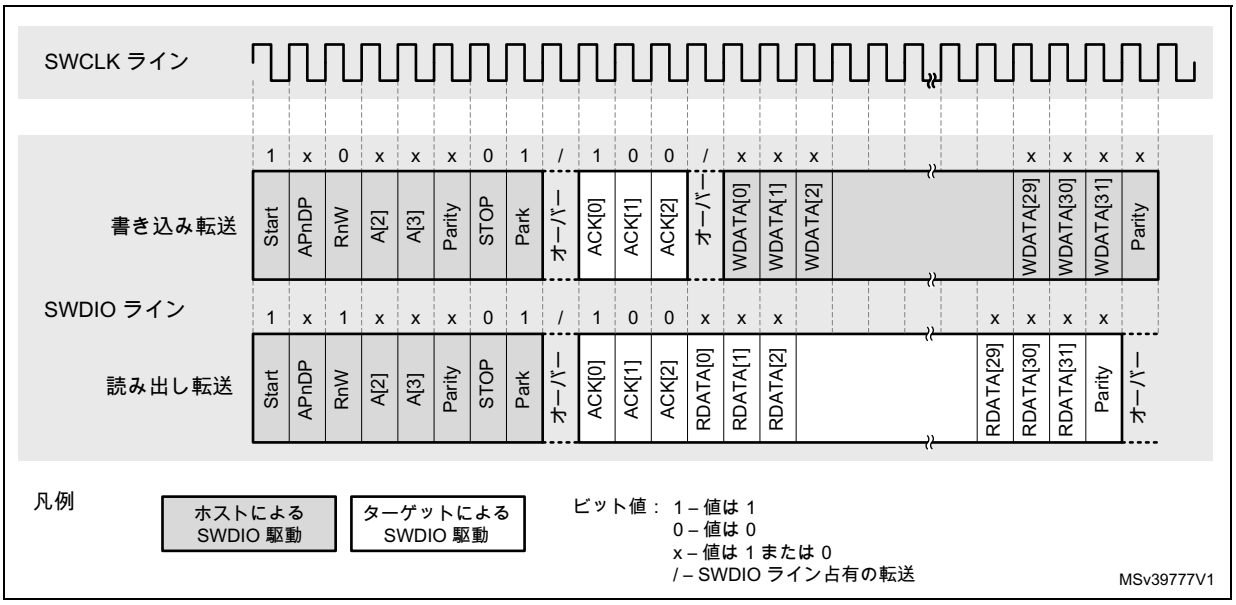
フィールドのビット	名前	説明
2:0	ACK	000b：フォールト 010b：ウェイト 100b：OK

表 558. データ転送

ビットフィールド	名前	説明
31:0	WDATA または RDATA	書き込み／読み出しデータ
32	Parity	32 データビットのうちの 1 ビットパリティ

図 810 は正常な書き込み・読み出し転送を示します。

図 810. SWD正常データ転送



ターゲットからフォールトまたはウェイトの ACK 応答があった場合、オーバーラン検出が有効でなければ、データ転送フェーズは取り消されます。この場合、データはターゲットから無視されるか（書き込み時）または駆動されません（読み出し時）。

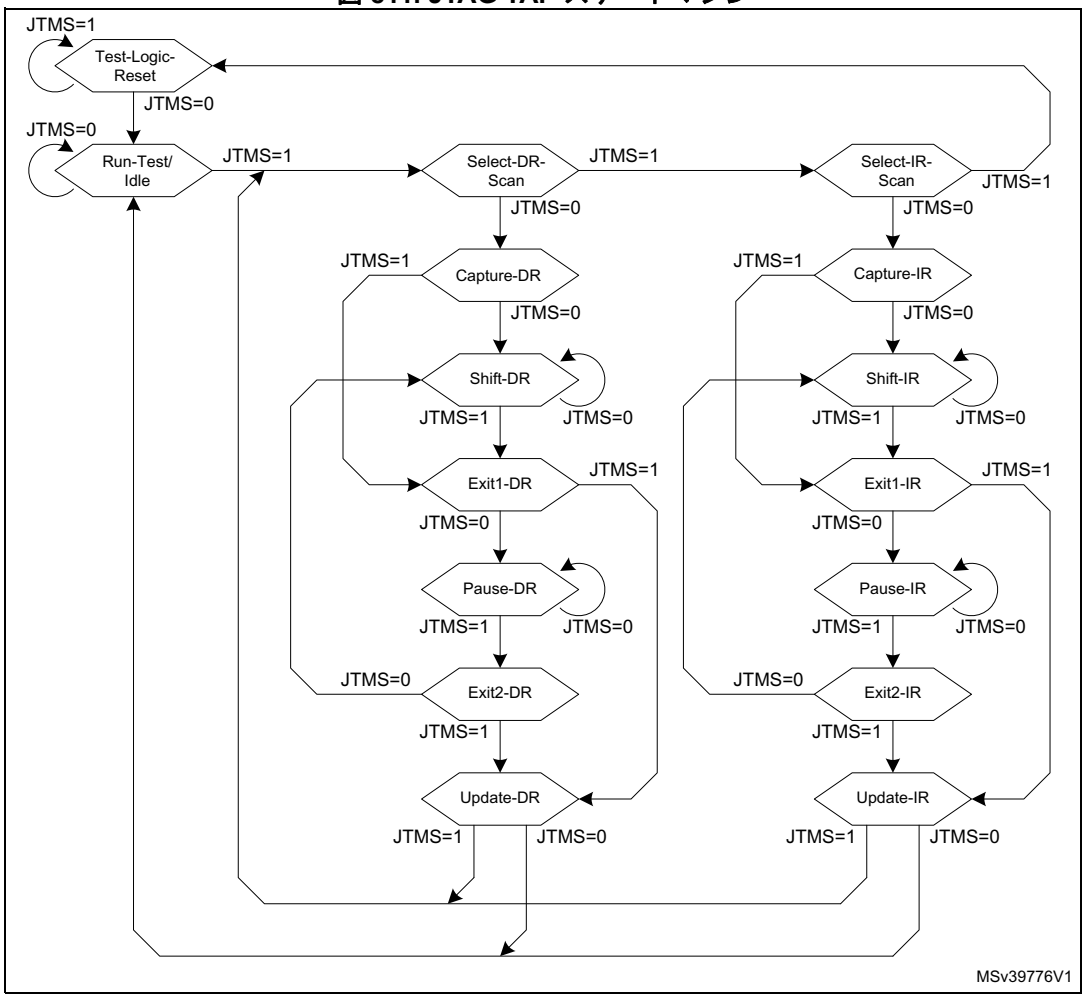
初回接続時またはプロトコルエラー発生時には、ホストはラインリセットを生成する必要があります。ラインリセットは、SWDIOが ハイの状態で SWCLK サイクルが 50 回以上、続いて SWDIOがローの状態でSWCLK サイクルが 2回で構成されています。

シリアルワイヤデバッグプロトコルの詳細については、ARM® Debug Interface Architecture Specification を参照してください *[1]*。

注： ***SWJ-DP は SWD プロトコルのバージョン 2 を実装しています。***

JTAG デバッグポート

図 811. JTAG TAP ステートマシン



JTAG-DP は IEEE 1149.1-1990 に基づく、TAP ステートマシン（TAPSM）を実装しています。ステートマシンを *図 811*に示します。ステートマシンは 2つのスキャンチェインを制御します。一方は 1 つの命令レジスタ（IR）に、もう一方は複数のデータレジスタ（DR）に関連付けられています。

TAPSM が Capture-IR ステートに入ると、0b0001 が命令レジスタ（IR）のスキャンチェインに転送されます。IR スキャンチェインは JTDI と JTDO の間に接続されます。

TAPSM が Shift-IR ステートにあるとき、IR スキャンチェインは JTCK の各立ち上がりエッジで 1 ビットだけシフトします。つまり、最初の単位時間で以下のアクションが起こります。

- IR スキャンチェインの LSB が JTDO に出力されます。
- IR スキャンチェインのビット [n] がビット [n-1] に転送されます。
- JTDI の値が IR スキャンチェインの MSB に転送されます。

TAPSM が Update-IR ステートに入ると、IR スキャンチェインにスキャンされた値が命令レジスタに転送されます。

TAPSM が Capture-DR ステートに入ると、ある値がデータレジスタの 1 つから、JTDI と JTDO の間に接続された DR スキャンチェインの 1 つに転送されます。

命令レジスタに保持された値によって、どのデータレジスタ、および該当する DR スキャンチェインを選択するかが決まります。

次に、このデータは、TAPSM が Shift-DR ステートにある間に、Shift-IR ステートで IR がシフトしたのと同じようにシフトします。

TAPSM が Update-DR ステートに入ると、DR スキャンチェインにスキャンされた値が選択されたデータレジスタに転送されます。

TAPSM が Run-Test/Idle ステートにあるときは、特別なアクションは起こりません。IDCODE 命令は IR にロードされます。

アクティブな場合、nJTRST 信号は Test-Logic-Reset ステートとは非同期にステートマシンをリセットします。

4 ビットの IR 命令に対応するデータレジスタを *表 559*に記載します。

表 559. JTAG-DP データレジスタ

命令レジスタ	データレジスタ	スキャンチェイン長	説明
0000～0111	（BYPASS）	1	実装されません。BYPASS が選択されます。
1000	ABORT	35	**アボートレジスタ** – ビット 31:1 = 予約済み – ビット 0 = APABORT：AP アボートを生成するには 1 を書き込みます。
1001	（BYPASS）	1	予約済み。BYPASS が選択されます。
1010	DPACC	35	**デバッグポートアクセスレジスタ** デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット 34:3 = DATA[31:0] = 書き込みリクエスト用に転送する 32 ビットデータ ビット 2:1 = A[3:2] = デバッグポートレジスタの 2 ビットアドレス ビット 0 = RnW = 読み出しリクエスト（1）または書き込みリクエスト（0） – 出力データ転送時： ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答： 010b = OK／フォールト 001b = ウェイト その他 = 予約済み

表 559. JTAG-DP データレジスタ（続き）

命令レジスタ	データレジスタ	スキャンチェイン長	説明
1011	APACC	35	**アクセスポートアクセスレジスタ** アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 – 入力データ転送時 ビット 34:3 = DATA[31:0] = 書き込みリクエスト用にシフトインする 32 ビットデータ ビット 2:1 = A[3:2] = アクセスポートレジスタの 2 ビットサブアドレス ビット 0 = RnW = 読み出しリクエスト（1）または書き込みリクエスト（0） – 出力データ転送時 ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答： 010b = OK／フォールト 001b = ウェイト その他 = 予約済み
1100	（BYPASS）	1	予約済み。BYPASS が選択されます。
1101	（BYPASS）	1	予約済み。BYPASS が選択されます。
1110	IDCODE	32	**ID コード** 0x05BA 0477：ARM® JTAG デバッグポートの ID コード
1111	BYPASS	1	**バイパス** 1 JTCK サイクル遅延が JTDI と JTDO の間に挿入されます。

DR レジスタの詳細は、ARM® Debug Interface Architecture Specification を参照してください *[1]*。

デバッグポートレジスタ

SW-DP と JTAG-DP はいずれもデバッグポート（DP）レジスタにアクセスします。これらを *表 560* に記載します。

デバッガは次のように DP レジスタへアクセスすることができます。

1. アクセスするレジスタバンクを選択するために、DP 内で SELECT レジスタの DPBANKSEL フィールドをプログラムします（*表 560*を参照）。
2. JTAG を使用する場合は、DPACC レジスタの A(3:2) フィールドにバンク内のレジスタアドレスを書き込みます。読み出しか書き込みかを選択するために、R/W ビットをプログラムします。書き込みの場合は、DATA フィールドに書き込みデータをプログラムします。SWD を使用する場合、A(3:2) および R/W フィールドは、APnDP ビットをリセットした状態で SW-DP に送信されるパケットリクエストワードの一部となります（*表 556*を参照）。書き込みデータはデータフェーズで送信されます。

表 560. デバッグポートレジスタ

アドレス	A[3:2] フィールドの値	読み出し／書き込み	説明
0x0	00	R	DP_DPIDR レジスタ。このレジスタは、デバッグポート用の IDCODE を格納します。
		W	DP_ABORT レジスタ(1)。現在の AP トランザクションをアボートします。また、このレジスタを使用して、DP_CTRL/STAT レジスタのエラーフラグをクリアします。
0x4	01	読み出し／書き込み	DPBANKSEL[3:0] = 0x0 （DP_SELECT レジスタ）の場合： CTRL/STAT レジスタ。DP を制御し、ステータス情報を提供します。
			DPBANKSEL[3:0] = 0x1 （DP_SELECT レジスタ）の場合： DP_DLCR レジスタ(2)。SWD データリンクの動作モードを制御します。
			DPBANKSEL[3:0] = 0x2 （DP_SELECT レジスタ）の場合： DP_TARGETID レジスタ。ターゲット識別情報を提供します。
			DPBANKSEL[3:0] = 0x3 （DP_SELECT レジスタ）の場合： DLPIDR レジスタ (2)。SWD プロトコルのバージョンを提供します。
0x8	10	R	RESEND レジスタ (2)。最後の AP 読み出しまたは DP_RDBUFF 読み出しによる戻り値を返すレジスタで、破損した読み出し転送が発生した場合に使用します。
		W	DP_SELECT レジスタ。アクセスポート、アクセスポートのレジスタバンク、および DP レジスタのアドレス 0x4 を選択します。
0xC	11	R	DP_RDBUFF レジスタ JTAG-DP 経由でこのレジスタを使用して、デバッガが一連の動作のあとで最終結果を得られるようにします（新しい JTAG-DP 動作のリクエストなし）。 SW-DP経由で使用する場合は、直前の AP 読み出しアクセスの結果を格納し、新たな AP アクセスを防ぎます。
		W	DP_TARGETSEL レジスタ (2) 。ラインリセットシーケンスの直後に DP_TARGETSEL に書き込む際に、次の 2 つの条件が満たされれば、ターゲットが選択されます。 – ビット [31:28] は、DP_DLPIDR レジスタのビット [31:28] と一致します。 – ビット [27:0] は、DP_TARGETID レジスタのビット [27:0] と一致します。 その他の値を書き込むと、ターゲットの選択が解除されます。選択をすべて解除するには、デバッグツールで 0xFFFFFFFF を書き込む必要があります。これは無効な DP_TARGETID の値です。その他の無効な DP_TARGETID の値はすべて保持されます。

1. JTAG-DP から AP ABORT レジスタへアクセスするには ABORT 命令を使用します。
2. SW-DP 経由でのみアクセス可能。レジスタは JTAG-DP 経由で「予約済み」です。

デバッグポートの識別レジスタ（DP_DPIDR）

アドレスオフセット：0x00

リセット値：0x6BA0 2477

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				PARTNO[7:0]								Res.	Res.	Res.	MIN
r	r	r	r	r	r	r	r	r	r	r	r				r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VERSION[3:0]				DESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット 31:28 **REVISION[3:0]**：リビジョンコード
0x6

ビット 27:20 **PARTNO[7:0]**：デバッグポートの部品番号
0xBA

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **MIN**：最小デバッグポート（MINDP)の実装
0：MINDP は実装されません（トランザクションカウンタとプッシュ動作をサポート）

ビット 15:12 **VERSION[3:0]**：DP アーキテクチャバージョン
0x2：DPv2

ビット 11:1 **DESIGNER[10:0]**：JEDEC 設計者 ID コード
0x23B：ARM®

ビット 0 予約済みであり、リセット値に保持する必要があります。

デバッグポートのアボートレジスタ（DP_ABORT）

アドレスオフセット：0x0

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ORUNERRCLR**：オーバーランエラーのクリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYORUN ビットをクリアします。

ビット 3 **WDERRCLR**：書き込みデータエラーのクリアビット

0：影響なし。

1：CTRL/STAT レジスタの WDATAERR ビットをクリアします。

ビット 2 **STKERRCLR**：スティッキーエラーのクリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYERR ビットをクリアします。

ビット 1 **STKCMPCLR**：スティッキー比較のクリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYCMP ビットをクリアします。

ビット 0 **DAPABORT**：現在の AP トランザクションのアボート

トランザクションがストールしていることを示す過剰な数のウェイト応答が返されると、トランザクションはアボートします。

0：影響なし。

1：トランザクションをアボートします。

デバッグポートの制御／ステータスレジスタ（DP_CTRL/STAT）

アドレスオフセット：0x4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CSYSPWRUPACK	CSYSPWRUPREQ	CDBGPWRUPACK	CDBGPWRUPREQ	CDBGRSTACK	CDBGRSTREQ	Res.	Res.	TRNCNT[11:4]							
r	rw	r	rw	r	rw			rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRNCNT[3:0]				MASKLANE[3:0]				WDATAERR	READOK	STICKYERR	ORUNERRCLR	WDERRCLR	STKERRCLR	STKCMPCLR	DAPABORT
rw	rw	rw	rw	rw	rw	rw	rw	r	r	rw	w	w	w	w	w

ビット 31 **CSYSPWRUPACK**：システムドメインのパワーアップステータスビット - 本デバイスでは使用されません。

ビット 30 **CSYSPWRUPREQ**：システムドメインのパワーアップ制御ビット - 本デバイスでは使用されません。

ビット 29 **CDBGPWRUPACK**：デバッグドメインのパワーアップステータスビット

このビットは読み出し専用です。電源コントローラからのデバッグドメインのパワーアップ ACK 信号のステータスを返します。

0：ドメインは電源オフ状態です。
1：ドメインは電源オン状態です。

ビット 28 **CDBGPWRUPREQ**：デバッグドメインのパワーアップ／ダウン制御ビット

このビットは、電源コントローラへのデバッグドメインのパワーアップ／ダウンリクエスト信号を制御します。

0：パワーダウンがリクエストされます。
1：パワーアップがリクエストされます。

ビット 27 **CDBGRSTACK**：デバッグドメインのリセットステータスビット - 本デバイスでは使用されません。

ビット 26 **CDBGRSTREQ**：デバッグドメインのリセット制御ビット - 本デバイスでは使用されません。

ビット 25:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:12 **TRNCNT[11:0]**：トランザクションカウンタ

APを介してインクリメントアドレスへの一連のトランザクションをプログラムするために、実行するトランザクション数が TRNCNT ビットにロードされます。このビットは、トランザクションが正常終了するたびにデクリメントされます。

ビット 11:8 **MASKLANE[3:0]**：プッシュ比較およびプッシュ検証のマスキングビット

このフィールドは、プッシュ比較／プッシュ検証操作でマスクされるバイトを示します（DP_CTRL/STAT レジスタのフィールド TRNMODE = 1 または 2）。プッシュ操作では、AP 書き込みトランザクションで提供されるワードを現在のターゲット AP アドレスの値と比較します。

0b1XXX：比較にバイトレーン 3 を含める。
0bX1XX：比較にバイトレーン 2 を含める。
0bXX1X：比較にバイトレーン 1 を含める。
0bXXX1：比較にバイトレーン 0 を含める。

ビット 7 **WDATAERR**：SW-DP の書き込みデータエラー

このビットは以下のことを示します。

- 書き込み操作のデータフェーズでパリティエラーまたはフレーミングエラーが発生しています。または、
- DP によって受け付けられた書き込み操作が AP に提供されることなく破棄されました。

このビットは読み出し専用です。DP_ABORT レジスタの WDERRCLR ビットに 1 を書き込むことによってリセットされます。

0：エラーはありません。
1：エラーが発生しました。

このビットは JTAG-DP に予約済みです。

ビット 6 **READOK**：SW-DP の AP 読み出し応答

このビットは、最後の AP 読み出しアクセスへの応答を示します。読み出し専用です。

0：読み出しは NG です。
1：読み出しは OK です。

このビットは JTAG-DP に予約済みです。

ビット 5 **STICKYERR**：トランザクションエラー（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、AP トランザクション中にエラーが発生したことを示します。

0：エラーはありません。
1：エラーが発生しました。

SW-DP では、このビットは、DP_ABORT レジスタの STKERRCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 をプログラムすることによってリセットされます。

ビット 4 **STICKYCMP**：比較一致（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、プッシュ操作中に一致が発生したことを示します。

0：TRNMODE = 0x1 であれば一致、TRNMODE = 0x2 であれば不一致です。
1：TRNMODE = 0x1 であれば不一致、TRNMODE = 0x2 であれば一致です。

SW-DP では、このビットは、DP_ABORT レジスタの STKCMPCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 をプログラムすることによってリセットされます。

ビット 3:2 **TRNMODE**：AP 書き込み操作の転送モード

読み出し操作 では、このフィールドを 0x0 に設定する必要があります。

0x0：通常動作 - AP トランザクションは直接 AP に渡されます。
0x1：プッシュ検証操作。DP は書き込みデータを保存し、ターゲット AP アドレスで読み出しトランザクションを実行します。読み出し操作の結果を保存データと比較します。不一致の場合、STICKYCMP ビットがセットされます。
0x2：プッシュ比較操作。DP は書き込みデータを保存し、ターゲット AP アドレスで読み出しトランザクションを実行します。読み出し結果を保存データと比較します。一致した場合、STICKYCMP ビットがセットされます。
0x3：予約済み

プッシュ操作では、MASKLANE フィールドで示されたデータバイトのみが比較に含まれます。

ビット 1 **STICKYORUN**：オーバーラン（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、オーバーラン（前のトランザクションの完了前に新たなトランザクションを受信）が発生したことを示します。このビットは、ORUNDETECT ビットがセットされている場合のみセットされます。

0： オーバーランはありません。
1：オーバーランが発生しました。

SW-DP では、このビットは、ABORT レジスタの ORUNERRCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 を書き込むことによってリセットされます。

ビット 0 **ORUNDETECT**：オーバーラン検出モード有効

0：オーバーラン検出は無効です。
1：オーバーラン検出は有効です。オーバーランが発生した場合、STICKYORUN ビットがセットされ、このビットがクリアされるまで次のトランザクションはブロックされます。

デバッグポートのデータリンク制御レジスタ（DP_DLCR）

アドレスオフセット：0x4

リセット値：0x0000 0040

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

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **TURNROUND[1:0]**：SWDIO のトライステート周期

0x0：1 データビット周期
0x1：2 データビット周期
0x2：3 データビット周期
0x3：4 データビット周期

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

デバッグポートのターゲット識別レジスタ（DP_TARGETID）

アドレスオフセット：0x4

リセット値：0x1045 0401

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TREVISION[3:0]				TPARTNO[15:4]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPARTNO[3:0]				TDESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット 31:28 **TREVISION[3:0]**：ターゲットのリビジョン

0x1：リビジョン 1

ビット 27:12 **TPARTNO[15:0]**：ターゲットの部品番号

0x0450：STM32H7

ビット 11:1 **TDESIGNER[10:0]**：ターゲットの設計者 JEDEC コード

0x020：STMicroelectronics

ビット 0 予約済みであり、リセット値に保持する必要があります。

デバッグポートのデータリンクプロトコル識別レジスタ（DP_DLPIDR）

アドレスオフセット：0x4

リセット値：0x0000 0001

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

ビット 31:28 **TINSTANCE[3:0]**：ターゲットのインスタンス番号

これらのビットは、マルチドロップシステムで本デバイスを使用する場合のインスタンス番号を定義します。

0x0

ビット 27:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PROTSVN[3:0]**：シリアルワイヤデバッグプロトコルのバージョン

0x1：バージョン 2

デバッグポートの再送信レジスタ（DP_RESEND）

アドレスオフセット：0x8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESEND[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESEND[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RESEND**：最後の AP 読み出し／DP RDBUFF 読み出しの値

これらのビットには、最後の AP 読み出しまたは DP_RDBUFF 読み出しによって返された値が格納されます。読み出し転送の破損時に使用されます。

デバッグポートのアクセスポート選択レジスタ（DP_SELECT）

アドレスオフセット：0x8

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
APSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w	w	w	w												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBANKSEL[3:0]				DPBANKSEL[3:0]			
								w	w	w	w	w	w	w	w

ビット 31:28 **APSEL[3:0]**：アクセスポート選択ビット

これらのビットによって次のトランザクションのアクセスポートを選択します。

0x0：AP0 - Cortex-M7 デバッグアクセスポート（AHB-AP）
0x1：AP1 - D3 アクセスポート（AHB-AP）
0x2：AP2 - システムデバッグアクセスポート（APB-AP）
0x3 ～0xF：予約済み

ビット 27:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **APBANKSEL[3:0]**：AP レジスタバンク選択ビット

これらのビットによって、次のトランザクションのアクティブな AP で 4 ワードレジスタバンクを選択します。

ビット 3:0 **DPBANKSEL[3:0]**：DP レジスタバンク選択ビット

これらのビットによって、デバッグポートのアドレス 0x4 のレジスタを選択します。

0x0：CTRL/STAT レジスタ
0x1：DLCR レジスタ
0x2：TARGETID レジスタ
0x3：DLPIDR レジスタ
0x4 から 0xF：予約済み

デバッグポートの読み出しバッファレジスタ（DP_RDBUFF）

アドレスオフセット：0xC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDBUFF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDBUFF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RDBUFF[31:0]**：最後の AP 読み出し値

このフィールドには、最後の AP 読み出しアクセスによって返された値が格納されます。AP 読み出しアクセスによる返り値を取り出すには、以下の 2 通りの方法があります。

– 同じアドレスへの 2 回目の読出しアクセスを行います。これにより対応するバスで新たなトランザクションが開始されます。
– 最後の AP 読み出しアクセスの返り値を DP_RDBUFF レジスタから読み出します。この場合、新たな AP トランザクションは開始されません。

デバッグポートのターゲット識別レジスタ（DP_TARGETSEL）

アドレスオフセット：0xC

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TINSTANCE[3:0]				TPARTNO[15:4]											
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPARTNO[3:0]				TDESIGNER[11:0]											Res.
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	

ビット 31:28 **TINSTANCE[3:0]**：ターゲットのインスタンス番号

このフィールドは、マルチドロップシステムにおけるターゲットデバイスのインスタンス番号を定義します。本デバイスを選択するには、このフィールドに DP_DLPIDR レジスタの TINSTANCE フィールドと同じ値を書き込んでください。

ビット 27:12 **TPARTNO[15:0]**：ターゲットの部品番号

このフィールドは、ターゲットデバイスの部品番号を定義します。本デバイスを選択するには、このフィールドに DP_TARGETID レジスタの TPARTNO フィールドと同じ値を 書き込んでください。

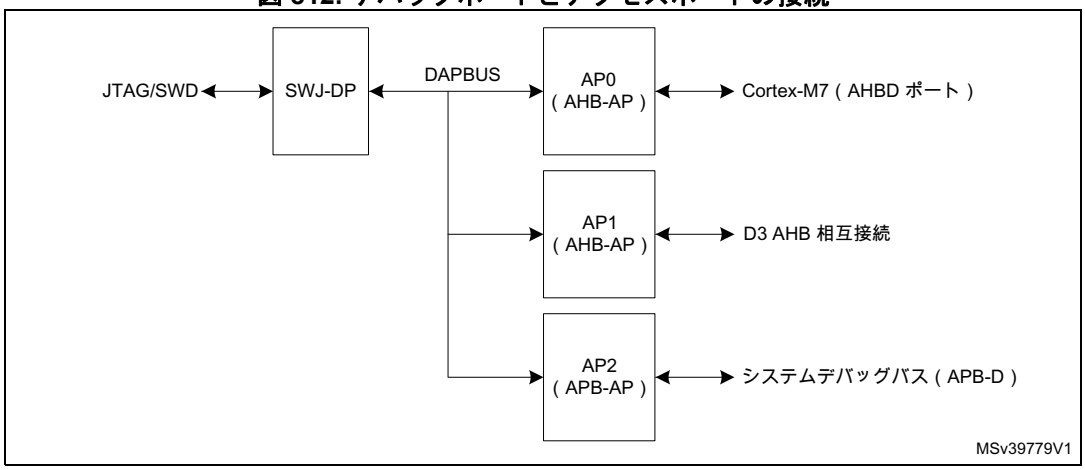
ビット 11:1 **TDESIGNER[10:0]**：ターゲットの設計者 JEDEC コード

このフィールドは、ターゲットデバイスの JEDEC コードを定義します。本デバイスを選択するには、このフィールドに DP_TARGETID レジスタの TDESIGNER フィールドと同じ値を書き込んでください。

ビット 0 予約済みであり、リセット値に保持する必要があります。

60.4.2 アクセスポート

図 812. デバッグポートとアクセスポートの接続



以下に挙げるアクセスポート（AP）が DP に接続されます。

1. AP0：Cortex-M7 アクセスポート（AHB-AP）。プロセッサの AHBD ポートに接続されている AHB-Lite バス を介して、Cortex-M7 プロセッサコアに組み込まれているデバッグおよびトレース機能へのアクセスを可能にします。
2. AP1：D3 アクセスポート（AHB-AP）。D3 ドメイン内のバスマトリックスへのアクセスを可能にします。これにより、D1 および D2 ドメインの電源がオフのときに、D3 ドメインメモリおよびペリフェラルが可視化されます。CoreSight コンポーネントはこのポートからアクセスできません。
3. AP2：システムアクセスポート（APB-AP）。システム APB デバッグバスに搭載されているデバッグおよびトレース機能、すなわち、プロセッサコアに内蔵されていないすべてのコンポーネントへのアクセスを可能にします。

アクセスポートはすべて MEM-AP タイプです。つまり、デバッグおよびトレースコンポーネントのレジスタは関連のデバッグバスのアドレス空間にマップされます。APは、デバッガより、それぞれ4つのレジスタからなるバンクとして構成された一連の32 ビットレジスタと見なされます。これらのレジスタの一部を使用して AP 自体を設定または監視し、その他のレジスタはバスで転送を行うために使用します。AP レジスタを *表 561*に記載します。

AP レジスタのアドレス構成は以下のとおりです。

- ビット [7:4] : DP_SELECT レジスタの APBANKSEL フィールドの内容
- ビット [3:2] : 使用するデバッグインタフェースに応じて、JTAG-DP の APACC データレジスタの A(3:2) フィールドの内容（*表 559*を参照）、または SW-DP パケットリクエストの内容（*表 556* を参照）のいずれか。
- ビット [1:0] : 常に 0 に設定されます。

DP の SELECT レジスタの APSEL フィールドには、アクセスされる MEM-AP を定義します。

デバッガは次のように AP レジスタへアクセスすることができます。

1. AP のうちの 1 つを選択するには DP_SELECT レジスタの APSEL フィールドを、アクセスするレジスタバンクを選択するには APBANKSEL フィールドをプログラムします。
2. JTAG を使用する場合は、バンク内にあるレジスタアドレスを APACC レジスタの A(3:2) フィールドに書き込みます。読み出しか書き込みかを選択するために、RnW ビットをプログラムします。書き込みの場合は、DATA フィールドに書き込みデータをプログラムします。SWD を使用する場合、A(3:2) および RnW フィールドは、APnDP ビットをセットした状態で SW-DP に送信されるパケットリクエストワードの一部となります（*表 556*を参照）。書き込みデータはデータフェーズで送信されます。

デバッガは、MEM-AP レジスタを使用して、以下の手順でメモリマップドデバッグコンポーネントレジスタにアクセス可能です（上記の AP レジスタアクセス手順を使用）。

1. TAR レジスタのトランザクションターゲットアドレスをプログラムします。
2. 必要であれば、転送パラメータ（AddrInc など）を CSW レジスタに書き込みます。
3. TAR レジスタに保持されているアドレスのバストランザクションを開始するには、DRW レジスタに対する書き込みまたは読み出しを行います。あるいは、バンクデータレジスタ BDn に対する読み出し／書き込みを行うことで、アドレス TAR[31:4] + n へのアクセスを開始します（TAR レジスタのアドレスを変更せずに、最大4つの連続したアドレスにアクセス可能）。

MEM-AP の詳細については、ARM® Debug Interface Architecture Specification を参照してください *[1]*。デバッグポートをデバッグコンポーネント（プロセッサ、ETM、ROM テーブルなど）に接続するための MEM-AP の使用については、セクション 7.1.2. に記載されています。

MEM-AP レジスタ

表 561. MEM-AP レジスタ

アドレス	APBANKSEL	A[3:2]	名前	説明
0x00	0x0	0	AP_CSW	制御／ステータスワードレジスタ
0x04	0x0	1	AP_TAR	転送アドレスレジスタ バストランザクションのターゲットアドレス
0x08	-	-	-	予約済み
0x0C	0x0	3	AP_DRW	データ読み出し／書き込みレジスタ このレジスタへアクセスすると、TAR[31:0] に格納されているアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x10	0x1	0	AP_BD0	バンクデータ 0 レジスタ このレジスタへアクセスすると、TAR[31:4] << (4 + 0x0) に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x14	0x1	1	AP_BD1	バンクデータ 1 レジスタ このレジスタへアクセスすると、TAR[31:4] << (4 + 0x4) に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x18	0x1	2	AP_BD2	バンクデータ 2 レジスタ このレジスタへアクセスすると、TAR[31:4] << (4 + 0x8) に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x1C	0x1	3	AP_BD3	バンクデータ 3 レジスタ このレジスタへアクセスすると、TAR[31:4] << (4 + 0xC) に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x20～0xEC	-	-	-	予約済み
0xF0	-	-	-	予約済み
0xF4	-	-	-	予約済み
0xF8	0xF	2	AP_BASE	デバッグ用ベースアドレスレジスタ（RO） ROM テーブルのベースアドレス
0xFC	0xF	3	AP_IDR	識別レジスタ（RO）

アクセスポートの制御／ステータスワードレジスタ（AP_CSW）

アドレスオフセット：0x0

リセット値：0x0000 0002 （APB-AP）、0x4000 0002 （AHB-AP）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SPROT	Res.	PROT[4:0]					SPI STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw	r							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MODE[3:0]				TRIN PROG	DEVIC EEN	ADDRINC[1:0]		Res.	SIZE[2:0]		
				rw	rw	rw	rw	r	r	rw	rw		rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **SPROT**：セキュア転送リクエストビット

APB-AP では、このフィールドは予約済みです。AHB-AP では、このフィールドにはバス転送の保護属性 HPROT[6] をセットします。

0：SPIDEN がハイであればセキュア転送です。SPIDEN がローであれば非セキュア転送です。

1：非セキュア転送です。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **PROT[4:0]** バス転送保護ビット

APB-AP では、このフィールドは予約済みです。AHB-AP では、このフィールドにはバス転送の保護属性 HPROT[4:0] をセットします。

0bXXXX0：命令フェッチ

0bXXXX1：データアクセス

0bXXX0X：ユーザモード

0bXXX1X：特権モード

0bXX0XX：バッファ格納不可

0bXX1XX：バッファ格納可

0bX0XXX：キャッシュ格納不可

0bX1XXX：キャッシュ格納可

0b0XXXX：非排他的

0b1XXXX：排他的

ビット 23 **SPISTATUS**：SPIDEN オプションビットのステータス

このビットは、デバッガがセキュアメモリにアクセスできるかどうかを決定します。このフィールドは APB-AP に予約済みです。

0：セキュア AHB 転送はブロックされます。

1：セキュア AHB 転送は許可されています。

ビット 22:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MODE[3:0]**：バリアサポート有効ビット

これらのビットは、メモリバリア操作がサポートされるかどうかを定義します。

0x0：サポートされません

ビット 7 **TRINPROG**：実行中の転送

このビットは AP でバス転送を実行中であることを示します。

0：実行中の転送はありません。

1：バス転送を実行中です。

ビット 6 **DEVICEEN**：デバイス有効ビット

このビットは AP にアクセスできるかどうかを定義します。

1：AP アクセスは有効です。

ビット 5:4 **ADDRINC[1:0]**：オートインクリメントモードビット

これらのビットは、TAR アドレスがトランザクション後に自動的にインクリメントされるかどうかを定義します。

0x0：自動インクリメントされません。
0x1：アドレスはトランザクションのバイトサイズ（SIZE フィールド）でインクリメントされます。
0x2：パックされた転送が有効です（AHB-AP の場合のみ - APB-AP では予約済み）。32 ビットの AP アクセスにより、プログラムされたトランザクションサイズに応じて、1 x 32 ビット、2 x 16 ビット、または 4 x 8 ビットのバストランザクションが開始されます。データはそれに応じてパック／アンパックされます。
0x3：予約済み

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SIZE[2:0]**：次のメモリアクセストランザクションのサイズ（AHB-AP の場合のみ）

0x0：バイト（8 ビット）
0x1：ハーフワード（16 ビット）
0x2：ワード（32 ビット）
0x3～0x7：予約済み

APB-AP の場合、このフィールドは読出し専用で、0x2（32 ビット）に固定されています。

アクセスポートのベースアドレスレジスタ（AP_BASE）

アドレスオフセット：0xF8

リセット値：0xE00F E003 (AP0)、0x0000 0002 (AP1)、0xE00E 0003 (AP2)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BASEADDR[19:4]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASEADDR[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FORMAT	ENTRYPRESENT
r	r	r	r											r	r

ビット 31:12 **BASEADDR[19:0]**：AP の ROM テーブルの ベースアドレス（ビット 31～12）

ROM テーブルは 4 KB の境界上に整列させる必要があるため、12 LSB はゼロです。
AP0 （Cortex-M7 AHB-AP）：0xE00FE
AP1 （D3 AHB-AP）：0x00000 （ROM テーブルなし）
AP2 （システム APB-AP）：0xE00E0

ビット 11:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **FORMAT**：ベースアドレスレジスタのフォーマット

1：ARM® Debug Interface V5

ビット 0 **ENTRYPRESENT**：デバッグコンポーネント有無ステータスビット

このビットは、アクセスポートバス上のデバッグコンポーネントの有無を示します。

0：デバッグコンポーネントはありません（AP1）。

1：デバッグコンポーネントがあります（AP0、AP2）。

アクセスポートの識別レジスタ（AP_IDR）

アドレスオフセット：0xFC

リセット値：0x6477 0001 （AP0 および AP1）、0x4477 0002 （AP2）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				JEDECBANK[3:0]				JEDECBANK[6:0]							MEM AP
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDENTITY[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:28 **REVISION[3:0]：**ARM コアリビジョン

0x2：r0p3

0x4：r0p5

0x6：r0p7

ビット 27:24 **JEDECBANK[3:0]：**JEDEC バンク

0x4：ARM®

ビット 23:17 **JEDECCODE[6:0]：**JEDEC コード

0x3B：ARM®

ビット 16 **MEMAP**：メモリアクセスポート

1：標準レジスタマップ

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **IDENTITY[7:0]**：AP タイプ識別

0x01：AHB-AP （AP0 および AP1）

0x02：APB-AP （AP2）

0x11：予約済み

参考資料

60.5 トレースおよびデバッグサブシステムの機能説明

トレースおよびデバッグサブシステムは、以下のCoreSight コンポーネントの機能を備えています。

- システム ROM テーブル
- グローバルタイムスタンプジェネレータ（TSG）
- システムクロストリガインタフェース（CTI）
- クロストリガマトリックス（CTM）
- トレースポートインタフェースユニット（TPIU）
- トレースバスファネル（CSTF）
- 組み込みトレース FIFO （ETF）
- シリアルワイヤ出力（SWO）
- シリアルワイヤ出力用トレースファネル（SWTF）

上記のコンポーネントへは、システム APB-AP およびその関連の APB-D デバッグバスを介してデバッガからアクセス可能です。また、Cortex-M7 プロセッサによってもアクセスできます。

MCU デバッグユニット（DBGMCU）へも APB-D経由でアクセス可能です。このCoreSightではないコンポーネントには、デバッグモードにおけるデバイスの動作を設定するレジスタが内蔵されています。

トレースバスのレプリケータは、トレースバスのファネルを使用して、CPUの ITM CoreSight コンポーネントから ETF と SWO までのトレースバスを分岐 させます。

60.5.1 システム ROM テーブル

APB-D バスには 2 つの ROM テーブルがあります。ROM テーブルは CoreSight コンポーネントの一つで、APB-D バス上にあるすべての CoreSight コンポーネントのベースアドレスを格納します。このようなテーブルを使用することにより、デバッガは CoreSight コンポーネントのトポロジを自動検出することができます。

1 番目のテーブルが 2 番目のテーブルを、さらに D3 電力ドメイン内の CoreSight コンポーネント、SWO、SWTF、TSGを指し示します。DBGMCU は CoreSight の標準コンポーネントではないので、テーブルは見に行きません。テーブルは 4 KB、すなわち APB-D アドレス空間の 32 ビット幅の領域（デバッガによるアクセス時は 0xE00E0000～0xE00E0FFC、システムバスによるアクセス時は 0x59000000～0x59000FFC）を占有します。

表 562. システム ROM テーブル 1

ROM テーブルのアドレスオフセット	コンポーネント名	コンポーネントベースアドレス（デバッガ）	コンポーネントベースアドレス（システムバス）	コンポーネントアドレスオフセット	サイズ	エントリ
0x000	システム ROM テーブル 2	0xE00F0000	0x59010000	0x10000	4 KB	0x00010003
0x004	SWO	0xE00E2000	0x59002000	0x02000	4 KB	0x00002003
0x008	タイムスタンプジェネレータ	0xE00E3000	0x59003000	0x03000	4 KB	0x00003003
0x00C	SWO ファネル	0xE00E4000	0x59004000	0x04000	4 KB	0x00004003
0x010	テーブルの先頭	-	-	-	-	0x00000000

表 562. システム ROM テーブル 1（続き）

ROM テーブルのアドレスオフセット	コンポーネント名	コンポーネントベースアドレス（デバッガ）	コンポーネントベースアドレス（システムバス）	コンポーネントアドレスオフセット	サイズ	エントリ
0x014 から 0xFC8	予約済み	-	-	-	-	0x00000000
0xFCC から 0xFFC	ROM テーブルレジスタ	-	-	-	-	*システム ROM レジスタ*を参照してください。

2番目のテーブルは 4 KB、すなわち APB-D アドレス空間の 32 ビット幅の領域（デバッガによるアクセス時は 0xE00F0000～0xE00F0FFC、システムバスによるアクセス時は 0x59010000～0x59010FFC）を占有します。

表 563. システム ROM テーブル 2

ROM テーブルのアドレスオフセット	コンポーネント名	コンポーネントベースアドレス（デバッガ）	コンポーネントベースアドレス（システムバス）	コンポーネントアドレスオフセット	サイズ	エントリ
0x000	システム CTI	0xE00F1000	0x59011000	0x1000	4 KB	0x00001003
0x004	トレースファネル	0xE00F3000	0x59013000	0x3000	4 KB	0x00003003
0x008	ETF	0xE00F4000	0x59014000	0x4000	4 KB	0x00004003
0x00C	TPIU	0xE00F5000	0x59015000	0x5000	4 KB	0x00005003
0x010	テーブルの先頭	-	-	-	-	0x00000000
0x014 から 0xFC8	予約済み	-	-	-	-	0x00000000
0xFCC から 0xFFC	ROM テーブルレジスタ	-	-	-	-	*システム ROM レジスタ*を参照してください。

各 ROM テーブルの先頭には、標準 CoreSight コンポーネントおよびペリフェラルの ID レジスタを含む、複数の読出し専用レジスタが格納されます。セクション *システム ROM レジスタ*を参照してください。

各デバッグコンポーネントは、アドレス空間の 4 KB のブロックを 1 ブロック以上占有します。このアドレス空間のブロックを、コンポーネントのデバッグレジスタファイルと呼びます。

ROM テーブルエントリのコンポーネントアドレスオフセットフィールドは、コンポーネントのアドレス空間の最後の 4 KB ブロックの先頭を指し示します。このブロックには常にコンポーネントおよびコンポ—ネントのペリフェラル ID レジスタが格納され、開始位置はブロックの先頭からオフセット 0xFD0 の位置です。4 KB カウントフィールド PIDR4 [7:4] には、コンポーネントの 4 KB ブロックの数を指定します。ここで、あるコンポーネントのアドレス空間の開始位置を見つける手順を以下に示します。

1. そのコンポーネントの ROM テーブルエントリを読み出し、ROM テーブルエントリのビット [31:12] から Address_Offset[18:0] を抜き出します。

2. アドレスオフセットと ROM テーブルのベースアドレス ROM_Base_Address を使用して、そのコンポーネントのベースアドレスを計算します。

   Component_Base_Address = ROM_Base_Address + Address_Offset

   Component_Base_Address はそのコンポーネントのアドレス空間の 4 KB ブロックの開始アドレスです。

3. そのコンポーネントのペリフェラル ID4 レジスタを読み出します。このレジスタのアドレスは次のように求められます。

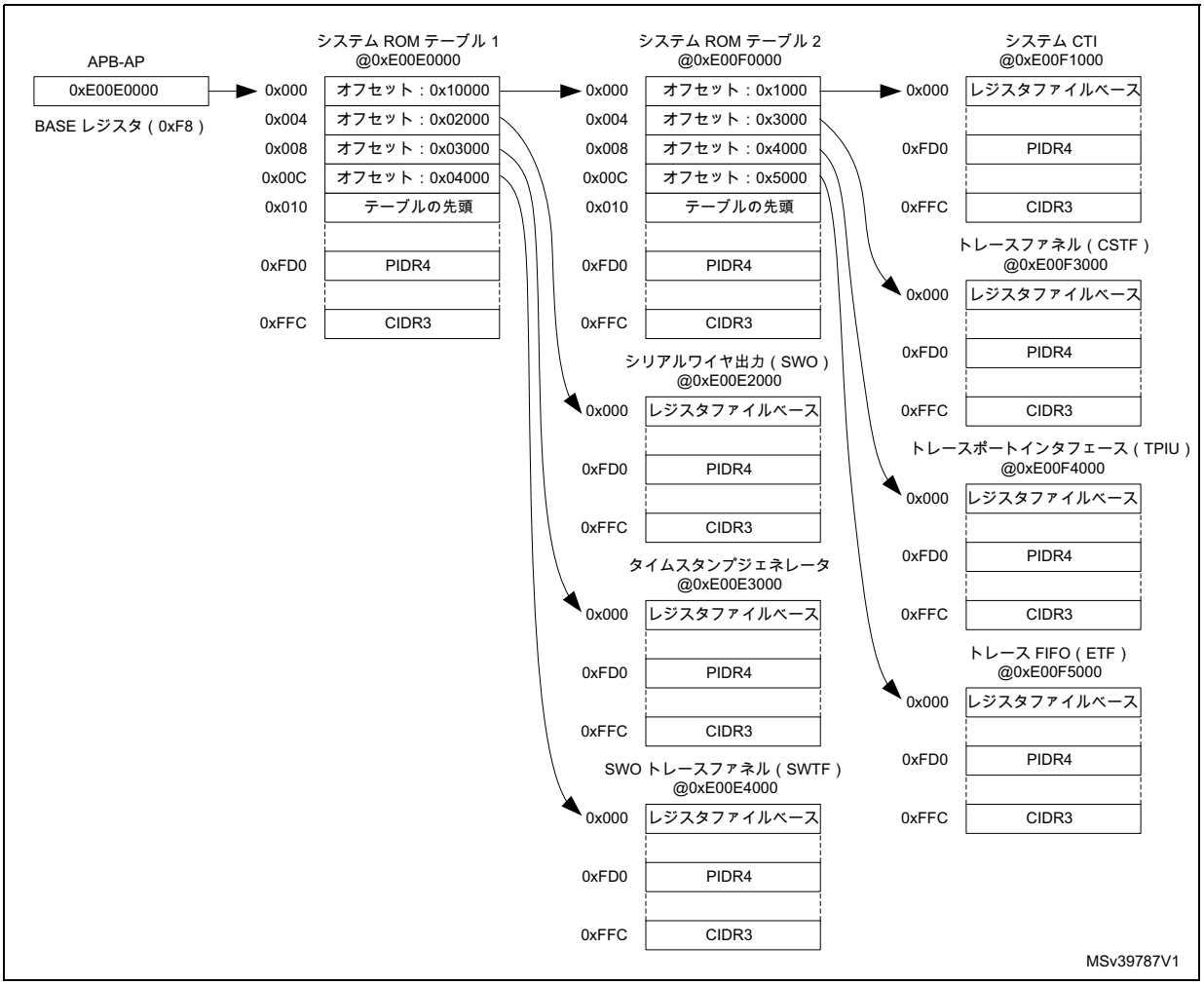
   Peripheral_ID4_address = Component_Base_Address + 0xFD0

4. ペリフェラル ID4 レジスタの値から 4 KB カウントフィールド [7:4] を抜き出します。

5. 4 KB カウントフィールドの値を使用して、そのコンポーネントのアドレス空間の開始アドレスを計算します。フィールドの値が 0b0000、すなわちカウント値 1 に相当する場合、そのコンポーネントのアドレス空間は手順 2 で求めた Component_Base_Address から始まります。

APB-D にあるCoreSight コンポーネントのトポロジを *図 813*に示します。

図 813. APB-D CoreSight コンポーネントのトポロジ



ROM テーブルの使用に関する詳細は、ARM® Debug Interface Architecture Specification を参照してください *[1]*。

システム ROM レジスタ

SYSROM メモリタイプレジスタ（SYSROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSMEM**：システムメモリ

0：このバス上にはシステムメモリはありません。

SYSROM CoreSight ペリフェラル ID レジスタ 4（SYSROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x0：STMicroelectronics JEDEC 継続コード

SYSROM CoreSight ペリフェラル ID レジスタ 0（SYSROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0050

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：デバイスの部品番号フィールド、ビット [7:0]

0x50：STM32H7 デバイス

SYSROM CoreSight ペリフェラル ID レジスタ 1（SYSROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0x0：STMicroelectronics JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：デバイスの部品番号フィールド、ビット [11:8]

0x4：STM32H7 デバイス

SYSROM CoreSight ペリフェラル ID レジスタ 2（SYSROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：デバイスのリビジョン番号
0x1：リビジョン 1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x2：STMicroelectronics JEDEC コード

SYSROM CoreSight ペリフェラル ID レジスタ 3（SYSROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

SYSROM CoreSight コンポーネント ID レジスタ 0（SYSROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

SYSROM CoreSight コンポーネント ID レジスタ 1（SYSROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0x1：ROM テーブルコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

SYSROM CoreSight コンポーネント ID レジスタ 2（SYSROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]

0x05：コモン ID 値

SYSROM CoreSight コンポーネント ID レジスタ 3（SYSROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE**：コンポーネント ID フィールド、ビット [31:24]

0xB1：コモン ID 値

システム ROM レジスタのマップとリセット値

表 564. システム ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	SYSROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																0
0xFD0	SYSROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD4	SYSROM_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0xFD8	SYSROM_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0xFDC	SYSROM_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み							
	リセット値																									0	0	0	0	0	0	0	0
0xFE0	SYSROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	1	0	1	0	0	0	0
0xFE4	SYSROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									0	0	0	0	0	1	0	0
0xFE8	SYSROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	0
0xFEC	SYSROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND [3:0]				CMOD [3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	SYSROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	SYSROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS [3:0]				PREAMBLE [11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	SYSROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	SYSROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.2 グローバルタイムスタンプジェネレータ（TSG）

グローバルタイムスタンプジェネレータには 64 ビットのカウンタが内蔵されており、システム内のすべてのトレースソース、すなわち プロセッサコア内の ETM および ITM に共通のタイミング基準を提供します。上記のコンポーネントは、トレースアナライザーによるトレースパケットの時系列のリカバリを可能にするタイムスタンプをトレースストリームに挿入します。タイムスタンプは、複数のトレースソースがファネル部で 1 つのストリームにマルチプレクスされた場合は失われます。

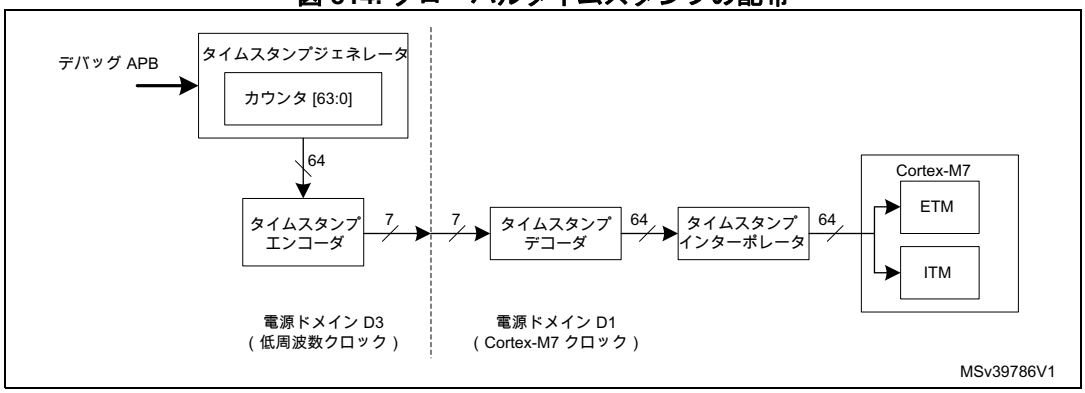
TSG レジスタは APB-D を通じてアクセス可能です。これによりデバッガまたはデバッグソフトウェアは以下の動作が可能になります。

- タイムスタンプのインクリメントを開始／停止します。
- 現在のタイムスタンプの値を読み取ります。
- 現在のタイムスタンプの値を変更します。
  - 値の変更中はタイムスタンプカウンタを停止しなければなりません。タイムスタンプの値が変更されると、タイムスタンプジェネレータはすべてのトレースソースと同期をとり直します。
- 報告されたタイムスタンプのインクリメントを変更します。

グローバルタイムスタンプジェネレータ CoreSight コンポーネントの詳細情報は、ARM® CoreSight™ SoC-400 Technical Reference Manual を参照してください *[2]*。

タイムスタンプジェネレータは D3 電力ドメイン内にあり、タイムスタンプは Cortex-M7に配布されます。電力ドメインの境界での配布を簡単にするために、64 ビットのタイムスタンプは 7 ビットにエンコードされ、その後転送先の電力ドメインでデコードされます。また、プロセッサのクロックがジェネレータのクロックよりかなり速い場合は、その分解能を上げるためにタイムスタンプを補間します。タイムスタンプの配布を *図 814*に示します。

図 814. グローバルタイムスタンプの配布



TSG レジスタ

TSG カウンタ制御レジスタ（TSG_CNTCR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDBG	EN
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HDBG**：デバッグ時の停止

0：通常動作

1：システム全体のデバッグステートが検出されるとカウンタを停止します - 実装されていません。

ビット 0 **EN**：有効化

0：カウンタは無効です。

1：カウンタは有効で、インクリメントしています。

TSG カウンタステータスレジスタ（TSG_CNTSR）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGH	Res.
														r	

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DBGH**：デバッグ停止

0：通常動作

1：システム全体がデバッグステートであるためカウンタが停止されます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

TSG カウンタ現在値下位レジスタ（TSG_CNTCVL）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNTCVL[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNTCVL[15:0]															
rw															

ビット 31:0 **CNTCVL[31:0]**：TSG カウンタ現在値フィールド、ビット [31:0]

現在のタイムスタンプの値を変更するには、最初に新しい値の下位 32 ビットをこのレジスタに書き込んでから、上位 32 ビットを CNTCVU に書き込みます。タイムスタンプの値は、CNTVCVU レジスタに書き込まれるまでは変わりません。注：このレジスタに書き込む前に、TSG_CNTCR レジスタの EN ビットをクリアする必要があります。

TSG カウンタ現在値上位レジスタ（TSG_CNTCVU）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNTCVU[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNTCVU[15:0]															
rw															

ビット 31:0 **CNTCVU[31:0]**：TSG カウンタ現在値フィールド、ビット [63:32]

現在のタイムスタンプの値を変更するには、最初に新しい値の下位 32 ビットを CNTCVL に書き込んでから、上位 32 ビットをこのレジスタに書き込みます。64 ビットのタイムスタンプの値は、このレジスタに書き込まれた時点で、両方の書き込み値に更新されます。注：このレジスタに書き込む前に、TSG_CNTCR レジスタの EN ビットをクリアしてはなりません。

TSG 基本周波数 ID レジスタ（TSG_CNTFID0）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FREQ[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FREQ[15:0]															
rw															

ビット 31:0 **FREQ**：TSG カウンタ周波数（Hz）のインクリメント

このフィールドが変わるときは必ず、トレースジェネレータのクロック周波数を書き込む必要があります。

TSG CoreSight ペリフェラル ID レジスタ 4（TSG_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

TSG CoreSight ペリフェラル ID レジスタ 0（TSG_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x01：TSG 部品番号

TSG CoreSight ペリフェラル ID レジスタ 1（TSG_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]

0x1：TSG 部品番号

TSG CoreSight ペリフェラル ID レジスタ 2（TSG_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

TSG CoreSight ペリフェラル ID レジスタ 3（TSG_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

TSG CoreSight コンポーネント ID レジスタ 0（TSG_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

TSG CoreSight コンポーネント ID レジスタ 1（TSG_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00F0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0xF：CoreSight Soc-400 コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

TSG CoreSight コンポーネント ID レジスタ 2（TSG_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]

0x05：コモン ID 値

TSG CoreSight コンポーネント ID レジスタ 3（TSG_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]

0xB1：コモン ID 値

TSG レジスタのマップとリセット値

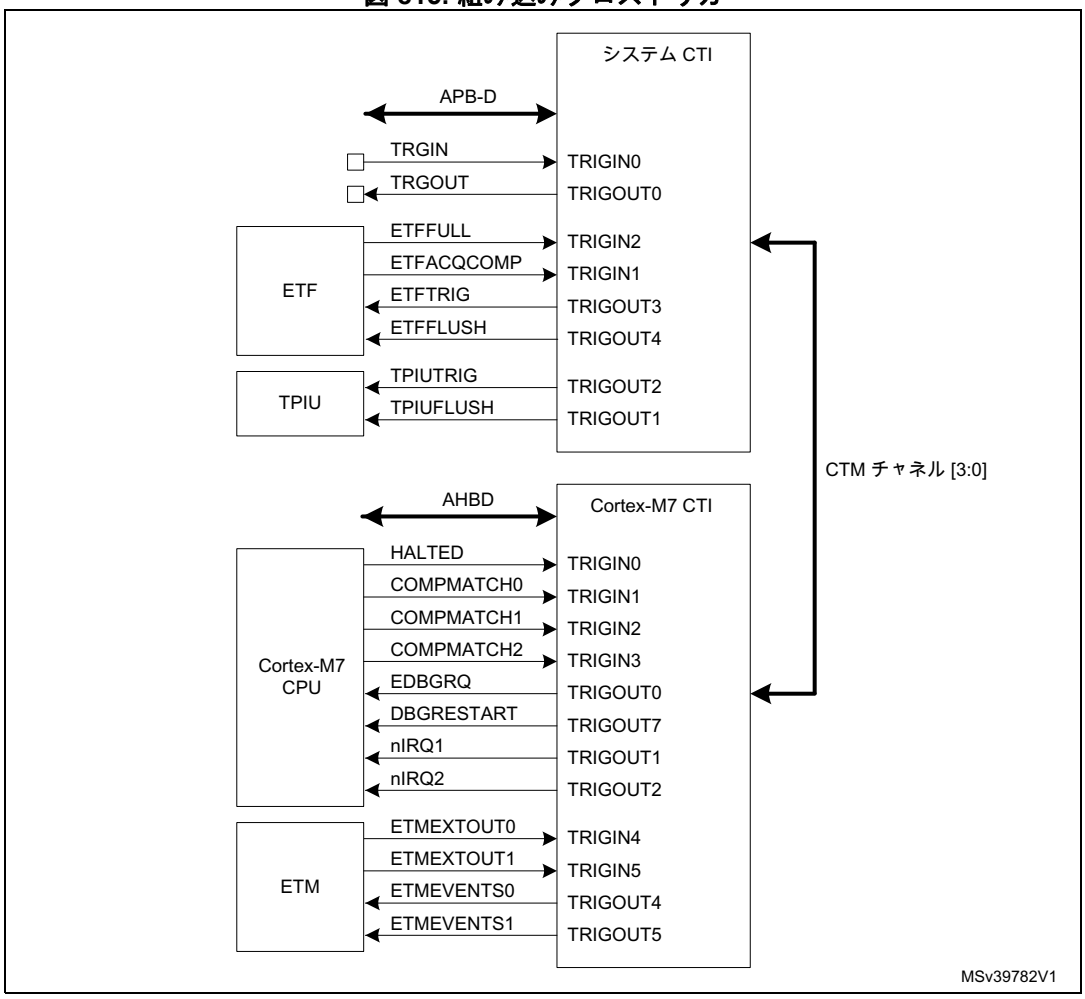
表 565. TSG レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	TSG_CNTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDBG	EN
	リセット値																															0	0
0x004	TSG_CNTSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGH	Res.
	リセット値																															0	
0x008	TSG_CNTCVL	CNTCVL_L_32																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	TSG_CNTCVU	CNTCVU_U_32																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	TSG_CNTFID0	FREQ																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xFD0	TSG_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT				JEP106CON			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	TSG_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	TSG_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	TSG_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	TSG_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	1	1	0	0	0	0	1
0xFE4	TSG_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	TSG_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	TSG_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND [3:0]				CMOD [3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	TSG_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	TSG_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS [3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	TSG_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	TSG_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.3 クロストリガインタフェース（CTI）／マトリックス（CTM）

クロストリガインタフェース（CTI）とクロストリガマトリックス（CTM）はともに CoreSight に埋め込まれているクロストリガ機能です。CTI には 2 種類の コンポーネントがあり、1 つはシステムレベル 、もう 1 つは Cortex-M7専用のものです。その 2 つの CTI は CTM を介して相互接続されています。システムレベルの CTI は システムのアクセスポートおよび関連の APB-D 経由でデバッガにアクセスできます。Cortex-M7 CTI は Cortex-M7 コアに物理的に統合されており、Cortex-M7 アクセスポートおよび関連の AHBD 経由でアクセス可能です。

図 815. 組み込みクロストリガ



CTI 機能を使用することで、さまざまな転送元からのイベントによりデバッグ／トレース動作、またはその両方をトリガすることができます。たとえば、外部トリガ入力で検出された遷移によりコードトレースを開始できます。

各 CTI には、トリガ入力とトリガ出力がそれぞれ最大 8つあります。同じ CTI 、または CTM を介して別の CTI で、任意の入力と出力を接続することができます。

各 CTI のトリガ入力および出力信号は *表 566* ～ *表 569*に記載されています。

表 566. システム CTI 入力

#	転送元信号	転送元コンポーネント	コメント
0	DBTRIGI	GPIO	外部トリガ入力 - 外部信号によるデバッグイベントの生成を可能にします。
1	ETFACQCOMP	ETF	ETF キャプチャ終了 - トレース FIFO が空のときにデバッグイベントの生成を可能にします。
2	ETFFULL	ETF	ETF フルフラグ - トレース FIFO がフルのときにデバッグイベントの生成を可能にします。
3	-	-	未使用
4	-	-	未使用
5	-	-	未使用
6	-	-	未使用
7	-	-	未使用

表 567. システム CTI 出力

#	出力信号	転送先コンポーネント	コメント
0	DBTRIGO	GPIO	外部 IO トリガ出力 - 外部 DBTRIGO ピンでのイベント監視を可能にします。
1	TPIUFLUSH	TPIU	トレースポート一掃トリガ - TPIU FIFO を一掃させます。
2	TPIUTRIG	TPIU	トレースポート有効トリガ - 外部トレースポートでのトレース出力を開始させます。
3	ETFTRIG	ETF	ETF 有効トリガ - トレース FIFO への入力を開始させます。
4	ETFFLUSH	ETF	ETF 一掃トリガ - トレース FIFO を一掃させます。
5	-	-	未使用
6	-	-	未使用
7	-	-	未使用

表 568. Cortex-M7 CTI 入力

#	転送元信号	転送元コンポーネント	コメント
0	HALTED	Cortex-M7 CPU	CPU 停止 - CPU がデバッグモードであることを示します。
1	COMPMATCH0	Cortex-M7 DWT	DWT コンパレータ 0 一致
2	COMPMATCH1	Cortex-M7 DWT	DWT コンパレータ 1 一致
3	COMPMATCH2	Cortex-M7 DWT	DWT コンパレータ 2 一致
4	ETMEXTOUT0	Cortex-M7 ETM	ETM 外部トリガ出力
5	ETMEXTOUT1	Cortex-M7 ETM	ETM 外部トリガ出力
6	-	-	未使用
7	-	-	未使用

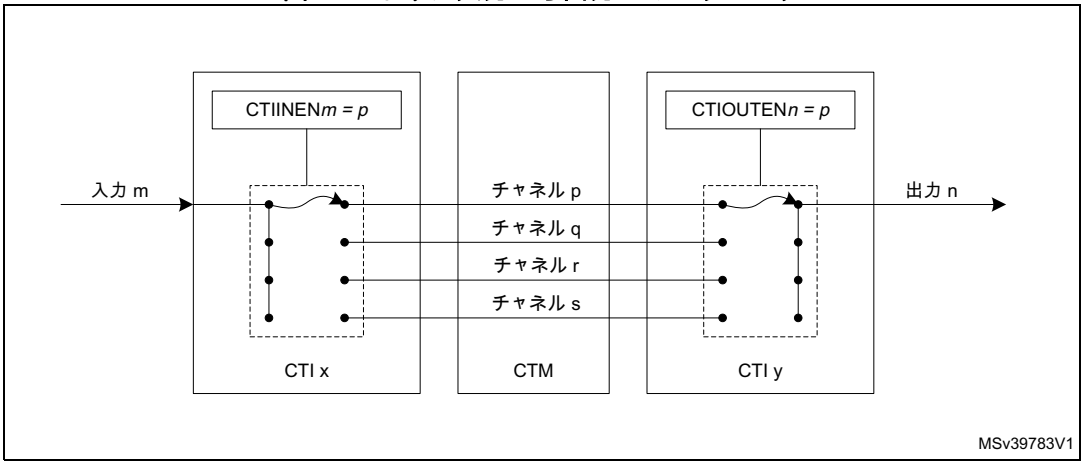
表 569. Cortex-M7 CTI 出力

#	出力信号	転送先コンポーネント	コメント
0	EDBGRQ	Cortex-M7 CPU	CPU 停止リクエスト - CPU をデバッグモードにします。
1	nIRQ1	Cortex-M7 NVIC	割り込みリクエスト
2	nIRQ2	Cortex-M7 NVIC	割り込みリクエスト
3	-	-	未使用
4	ETMEVENTS0	Cortex-M7 ETM	ETM トリガリクエスト - CPU 実行トレースを有効にします。
5	ETMEVENTS1	Cortex-M7 ETM	ETM トリガリクエスト - CPU 実行トレースを有効にします。
6	-	-	未使用
7	DBGRESTART	Cortex-M7 CPU	CPU 再開リクエスト - CPU はデバッグモードを終了します。

クロストリガマトリックスには 4 つのイベントチャネルがあります。それにより、異なる CTI のトリガ入力と出力の間に最大 4 つの双方向パラレル接続を設けることができます。CTI *x* の入力番号 *m* を CTI *y*, の出力番号 *n* に接続するには、CTI *x* の CTIINEN*m* レジスタを使用して、入力をイベントチャネル *p* に接続する必要があります。CTI *y* の CTIOUTEN*n* レジスタを使用して、同じチャネル *p* を出力に接続してください。注：上記は、入力と出力が同じ CTI に属する場合でも適用されます。

1つの入力は 1つ以上のチャネル（最大４つ）に接続できるため、複数の出力に振り分けることができます。同様に、出力も複数の入力に接続できます。複数の入力／出力を同じチャネルに接続することも可能です。

図 816. トリガ入力から出力へのマッピング



クロストリガインタフェース CoreSight コンポーネントの詳細情報は、ARM® CoreSight™ SoC-400 Technical Reference Manual を参照してください *[2]*。

CTI レジスタ

各 CTI のレジスタファイルのベースアドレスは、接続先のバスの ROM テーブルで定義されます。レジスタはどの CTI でも同じです。

CTI 制御レジスタ（CTI_CONTROL）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**GLBEN**：グローバル有効

0：クロストリガは無効です。

1：クロストリガは有効です。

CTI トリガ確認応答レジスタ（CTI_INTACK）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**INTACK[7:0]**：トリガ確認応答

CTITRIGOUT 出力ごとに レジスタの 1 ビットが割り当てられます。このレジスタの 1 ビットに 1 が書き込まれると、対応する CTITRIGOUT 出力が確認応答され、クリアされます。

CTI アプリケーショントリガ設定レジスタ（CTI_APPSET）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **APPSET[3:0]**：チャネルイベントの設定

読み出し：
0bXXX0：チャネル 0 イベントはインアクティブです。
0bXXX0：チャネル 0 イベントはアクティブです。
0bXX0X：チャネル 1 イベントはインアクティブです。
0bXX1X：チャネル 1 イベントはアクティブです。
0bX0XX：チャネル 2 イベントはインアクティブです。
0bX1XX：チャネル 2 イベントはアクティブです。
0b0XXX：チャネル 3 イベントはインアクティブです。
0b1XXX：チャネル 3 イベントはアクティブです。

書き込み：
0bXXX0：影響なし。
0bXXX0：チャネル 0 にイベントを設定します。
0bXX0X：影響なし。
0bXX1X：チャネル 1 にイベントを設定します。
0bX0XX：影響なし。
0bX1XX：チャネル 2 にイベントを設定します。
0b0XXX：影響なし。
0b1XXX：チャネル 3 にイベントを設定します。

CTI アプリケーショントリガクリアレジスタ（CTI_APPCLEAR）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR[3:0]			
												w			

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**APPCLEAR[3:0]**：チャネルイベントのクリア

0bXXX0：影響なし。
0bXXX0：チャネル 0 のイベントをクリアします。
0bXX0X：影響なし。
0bXX1X：チャネル 1 のイベントをクリアします。
0bX0XX：影響なし。
0bX1XX：チャネル 2 のイベントをクリアします。
0b0XXX：影響なし。
0b1XXX：チャネル 3 のイベントをクリアします。

CTI アプリケーションパルスレジスタ（CTI_APPPULSE）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE[3:0]			
												w			

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**APPULSE**：チャネルイベントのパルス出力

このレジスタ自体がただちにクリアされます。

0bXXX0：影響なし。
0bXXX0：チャネル 0 のパルスを生成します。
0bXX0X：影響なし。
0bXX1X：チャネル 1 のパルスを生成します。
0bX0XX：影響なし。
0bX1XX：チャネル 2 のパルスを生成します。
0b0XXX：影響なし。
0b1XXX：チャネル 3 のパルスを生成します。

CTI トリガ入力 x 有効レジスタ（CTI_INENx）

アドレスオフセット：0x020 + 4 * x （x = 0～7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TRIGINEN[3:0]**：クロストリガイベント有効化

CTITRIGINx がアクティブ化された時点で（x = 0～7）、4つのチャネルそれぞれのクロストリガイベントを有効または無効にします。

0bXXX0：トリガ n はチャネル 0 のイベントを生成しません。
0bXXX0：トリガ n はチャネル 0 のイベントを生成します。
0bXX0X：トリガ n はチャネル 1 のイベントを生成しません。
0bXX1X：トリガ n はチャネル 1 のイベントを生成します。
0bX0XX：トリガ n はチャネル 2 のイベントを生成しません。
0bX1XX：トリガ n はチャネル 2 のイベントを生成します。
0b0XXX：トリガ n はチャネル 3 のイベントを生成しません。
0b1XXX：トリガ n はチャネル 3 のイベントを生成します。

CTI トリガ出力 x 有効レジスタ（CTI_OUTENx）

アドレスオフセット：0x0A0 + 4 * x （x = 0～7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.					TRIGOUTEN[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TRIGOUTEN[3:0]**：イベント発生時のトリガ有効化

各チャネルのこのフィールドは、そのチャネルのイベントが CTITRIGOUTx （x = 0～7）のトリガを生成するかどうかを定義します。

0bXXX0：チャネル 0 イベントはトリガ出力 n のトリガを生成しません。
0bXXX0：チャネル 0 イベントはトリガ出力 n のトリガを生成します。
0bXX0X：チャネル 1 イベントはトリガ出力 n のトリガを生成しません。
0bXX1X：チャネル 1 イベントはトリガ出力 n のトリガを生成します。
0bX0XX：チャネル 2 イベントはトリガ出力 n のトリガを生成しません。
0bX1XX：チャネル 2 イベントはトリガ出力 n のトリガを生成します。
0b0XXX：チャネル 3 イベントはトリガ出力 n のトリガを生成しません。
0b1XXX：チャネル 3 イベントはトリガ出力 n のトリガを生成します。

CTI トリガ入力ステータスレジスタ（CTI_TRGISTS）

アドレスオフセット：0x130

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TRIGINSTATUS[7:0]**：トリガ入力ステータス

CTITRIGIN 入力ごとに レジスタの 1 ビットが割り当てられます。このビットが 1 にセットされている場合は、対応するトリガ入力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するトリガ入力はインアクティブです。

CTI トリガ出力ステータスレジスタ（CTI_TRGOSTS）

アドレスオフセット：0x134

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TRIGOUTSTATUS[7:0]**：トリガ出力ステータス

CTITRIGOUT 出力ごとに レジスタの 1 ビットが割り当てられます。このビットが 1 にセットされている場合は、対応するトリガ出力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するトリガ出力はインアクティブです。

CTI チャネル入力ステータスレジスタ（CTI_CHINSTS）

アドレスオフセット：0x138

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHINSTATUS[3:0]			
												r			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CHINSTATUS[3:0]**：チャネル入力ステータス

チャネル入力ごとに レジスタの 1 ビットが割り当てられます。このビットが 1 にセットされている場合は、対応するチャネル入力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するチャネル入力はインアクティブです。

CTI チャネル出力ステータスレジスタ（CTI_CHOUTSTS）

アドレスオフセット：0x13C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHOUTSTATUS[3:0]			
												r			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CHOUTSTATUS[3:0]**：チャネル出力ステータス

チャネル出力ごとに レジスタの 1 ビットが割り当てられます。このビットが 1 にセットされている場合は、対応するチャネル出力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するチャネル出力はインアクティブです。

CTI チャネルゲートレジスタ（CTI_GATE）

アドレスオフセット：0x140

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GATEEN[3:0]			
												rw			

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**GATEEN[3:0]**：チャネル出力有効化

チャネルごとに、そのチャネルのイベントが CTM を通って 他のCTI へ伝わり得るかどうかを定義します。

0bXXX0：チャネル 0 イベントは伝わりません。
0bXXX0：チャネル 0 イベントは伝わります。
0bXX0X：チャネル 1 イベントは伝わりません。
0bXX1X：チャネル 1 イベントは伝わります。
0bX0XX：チャネル 2 イベントは伝わりません。
0bX1XX：チャネル 2 イベントは伝わります。
0b0XXX：チャネル 3 イベントは伝わりません。
0b1XXX：チャネル 3 イベントは伝わります。

CTI 要求タグ設定レジスタ（CTI_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

CTI 要求タグクリアレジスタ（CTI_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

CTI ロックアクセスレジスタ（CTI_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：CTI レジスタ書き込みアクセス有効化

プロセッサコア による一部の CTI レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

CTI ロックステータスレジスタ（CTI_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：CTI_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT：**現在のロックステータス

このビットは、外部デバッガによる読み出し時に常に 0 を返します。

0：書き込みアクセスが許可されます。
1：書き込みアクセスはブロックされます。読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：**ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に常に 0 を返します。

0：ロック制御メカニズムは存在しません。
1：ロック制御メカニズムが実装されています。

CTI 認証ステータスレジスタ（CTI_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されていません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されていません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

CTI デバイス設定レジスタ（CTI_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0004 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMCH[3:0]			
												r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMTRIG[7:0]								Res.	Res.	Res.	EXTMUXNUM[4:0]				
r											r				

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **NUMCH[3:0]**：使用できる ECT チャネルの数
0x4：4 チャネル

ビット 15:8 **NUMTRIG[7:0]**：使用できる ECT トリガの数
0x8：8 つのトリガ入力と 8 つのトリガ出力

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **EXTMUXNUM[4:0]**：トリガ入力／出力マルチプレクサの数
0x0：なし

CTI デバイスタイプ識別子レジスタ（CTI_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0014

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類

0x1：このコンポーネントがクロストリガコンポーネントであることを示します。

ビット 3:0 **MAJORTYPE[3:0]**：大分類

0x4：このコンポーネントによりデバッガが CoreSight SoC-400 システムの他のコンポーネントを制御できることを示します。

CTI CoreSight ペリフェラル ID レジスタ 4（CTI_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

CTI CoreSight ペリフェラル ID レジスタ 0（CTI_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0006

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x06：CTI 部品番号

CTI CoreSight ペリフェラル ID レジスタ 1（CTI_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x9：CTI 部品番号

CTI CoreSight ペリフェラル ID レジスタ 2（CTI_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 004B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x4：r0p5

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

CTI CoreSight ペリフェラル ID レジスタ 3（CTI_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]：**メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CTI CoreSight コンポーネント ID レジスタ 0（CTI_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]
0x0D：コモン ID 値

CTI CoreSight コンポーネント ID レジスタ 1（CTI_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

CTI CoreSight コンポーネント ID レジスタ 2（CTI_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

CTI CoreSight コンポーネント ID レジスタ 3（CTI_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

CTI レジスタのマップとリセット値

表 570. CTI レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	CTI_CONTROL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLBEN
	リセット値																																0
0x010	CTI_INTACK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTACK[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x014	CTI_APPSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPSET[3:0]			
	リセット値																													0	0	0	0
0x018	CTI_APPCLEAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR [3:0]			
	リセット値																													0	0	0	0
0x01C	CTI_APPPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE [3:0]			
	リセット値																													0	0	0	0
0x020	CTI_INEN0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x024	CTI_INEN1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x028	CTI_INEN2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x02C	CTI_INEN3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x030	CTI_INEN4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x034	CTI_INEN5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x038	CTI_INEN6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x03C	CTI_INEN7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN [3:0]			
	リセット値																													0	0	0	0
0x0A0	CTI_OUTEN0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0S4	CTI_OUTEN1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0S8	CTI_OUTEN2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0

表 570. CTI レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0SC	CTI_OUTEN3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0B0	CTI_OUTEN4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0B4	CTI_OUTEN5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0B8	CTI_OUTEN6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x0BC	CTI_OUTEN7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN [3:0]			
	リセット値																													0	0	0	0
0x130	CTI_TRIGISTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x134	CTI_TRIGOSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x138	CTI_CHINSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHISTATUS [3:0]			
	リセット値																													0	0	0	0
0x13C	CTI_CHOUTSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHOSTATUS [3:0]			
	リセット値																													0	0	0	0
0x140	CTI_GATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GATEEN [3:0]			
	リセット値																													1	1	1	1
0xFA0	CTI_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1
0xFA4	CTI_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFB0	CTI_LAR	KEY																															
	リセット値																																
0xFB4	CTI_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTT	SLK	SLI
	リセット値																														0	1	1
0xFB8	CTI_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									0	0	0	0	0	1	0	1
0xFC8	CTI_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMCH[3:0]				NUMTRIG[7:0]								Res.	Res.	Res.	EXMUXNUM [4:0]				
	リセット値													0	1	0	0	0	0	0	0	1	0	0	0				0	0	0	0	0
0xFCC	CTI_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUB[3:0]				MAJOR[3:0]			
	リセット値																									0	0	0	1	0	1	0	0

表 570. CTI レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	CTI_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	CTI_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	CTI_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	CTI_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	CTI_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	1	1	0
0xFE4	CTI_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	CTI_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	1	0	0	1	0	1	1
0xFEC	CTI_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CTI_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	CTI_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	CTI_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CTI_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.4 トレースファネル（CSTF）

トレースファネルは、CoreSight コンポーネントの一つで、ATB バスを 2 つの トレースソースから 1 つの ATB へ結合させます。CSTF には 2つの ATB スレーブポートと 1 つの ATB マスタポートがあります。アービタはプログラム可能な優先度に従ってスレーブポートを選択します。

スレーブポートは、以下のように接続されます。

- S0：Cortex-M7 ETM
- S1：Cortex-M7 ITM

CSTF レジスタにより、スレーブポートを個別に有効化でき、それぞれの優先度を設定することができます。優先度はトレースが無効なときにのみ変更できます。アービトレーションは以下のように機能します。

- アービタでは、データが有効な最も高い優先度のスレーブポートが選択されます。
- *min_hold_time* 転送までが選択されたスレーブポートからマスタポートに渡され、そこで CONTROL レジスタ内に *min_hold_time* をプログラム可能です。
- その時点で新たなアービトレーションが行われます。

高い優先度は、バッファの容量が小さい転送元に接続されているスレーブポートか、またはデータ損失が許されない場所に割り当てるべきです。低い優先度は重要でない転送元または大容量のバッファを持つ転送元に割り当てるべきです。

ATB ファネル CoreSight コンポーネントの詳細情報は、ARM® CoreSight™ SoC-400 Technical Reference Manual を参照してください *[2]*。

トレースファネルレジスタ

CSTF 制御レジスタ（CSTF_CTRL）

アドレスオフセット：0x000

リセット値：0x0000 0300

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MIN_HOLD_TIME[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	ENS1	ENS0
				rw										rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MIN_HOLD_TIME[3:0]**：アービトレーション間に実行するトランザクションの数
0x0：1 トランザクション
:
0xE：15 トランザクション
0xF：予約済み

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENS1**：スレーブポート S1 の有効化
0：ポートを無効にします。
1：ポートを有効にします。

ビット 0 **ENS0**：スレーブポート S0 有効化
0：ポートを無効にします。
1：ポートを有効にします。

CSTF 優先度レジスタ（CSTF_PRIORITY）

アドレスオフセット：0x004

リセット値：0x0000 0688

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT1[2:0]			PRIPORT0[2:0]		
										rw			rw		

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:3 **PRIPORT1[2:0]**：スレーブポート S1 の優先度

0：最高優先度

:

7：最低優先度

ビット 2:0 **PRIPORT0[2:0]**：スレーブポート S0 の優先度

0：最高優先度

:

7：最低優先度

CSTF 要求タグ設定レジスタ（CSTF_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書き込み：

0000：影響なし。

xxx1：ビット 0 をセットします。

xx1x：ビット 1 をセットします。

x1xx：ビット 2 をセットします。

1xxx：ビット 3 をセットします。

読み出し：

0xF：要求タグには 4 つのビットがあることを示します。

CSTF 要求タグクリアレジスタ（CSTF_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

CSTF ロックアクセスレジスタ（CSTF_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：CSTF レジスタ書き込みアクセス有効化

このフィールドは、プロセッサコア による一部の CSTF レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

CSTF ロックステータスレジスタ（CSTF_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK TYPE	LOCK GRANT	LOCK EXIST
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：CSTF_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT：**現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。

1：書き込みアクセスはブロックされます。読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：**ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。

1：ロック制御メカニズムが実装されています。

CSTF 認証ステータスレジスタ（CSTF_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

CSTF CoreSight デバイス ID レジスタ（CSTF_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0024

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME[3:0]				PORTCNT[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SCHEME[3:0]**：優先度スキーム
0x2：静的優先度

ビット 3:0 **PORTCNT[3:0]**：接続されている入力ポートの数
0x4：4つの入力ポート

CSTF CoreSight デバイスタイプ ID レジスタ（CSTF_TYPEID）

アドレスオフセット：0xFCC

リセット値：0x0000 0012

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DEVTYPEID[7:0]**：デバイスタイプID
0x12：トレースファネル

CSTF CoreSight ペリフェラル ID レジスタ 0（CSTF_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x08：CSTF 部品番号

CSTF CoreSight ペリフェラル ID レジスタ 1（CSTF_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]

0x9：CSTF 部品番号

CSTF CoreSight ペリフェラル ID レジスタ 2（CSTF_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

CSTF CoreSight ペリフェラル ID レジスタ 3（CSTF_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CSTF CoreSight ペリフェラル ID レジスタ 4（CSTF_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ
0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード
0x4：ARM® JEDEC コード

CSTF CoreSight コンポーネント ID レジスタ 0（CSTF_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

CSTF CoreSight コンポーネント ID レジスタ 1（CSTF_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0] :** コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

CSTF CoreSight コンポーネント ID レジスタ 2（CSTF_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

CSTF CoreSight コンポーネント ID レジスタ 3（CSTF_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

トレースファネルレジスタのマップとリセット値

表 571. CSTF レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	CSTF_CTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIN_HOLD_TIME [3:0]				Res.	Res.	Res.	Res.	ENS3	ENS2	ENS1	ENS0
	リセット値																					0	0	1	1					0	0	0	0
0x004	CSTF_PRIORITY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT3 [2:0]			PRIPORT2 [2:0]			PRIPORT1 [2:0]			PRIPORT0 [2:0]		
	リセット値																					0	1	1	0	1	0	0	0	1	0	0	0
0xFA0	CSTF_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1
0xFA4	CSTF_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFB0	CSTF_LAR	KEY[31:0]																															
	リセット値																																

表 571. CSTF レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFB4	CSTF_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTT	SLK	SLI
	リセット値																														0	1	1
0xFB8	CSTF_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFC8	CSTF_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME [3:0]				PORTCNT [3:0]			
	リセット値																									0	0	1	0	0	1	0	0
0xFCC	CSTF_TYPEID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
	リセット値																									0	0	0	1	0	0	1	0
0xFD0	CSTF_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	CSTF_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	CSTF_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	CSTF_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	CSTF_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	1	0	0	0
0xFE4	CSTF_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	CSTF_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	CSTF_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CSTF_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	CSTF_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	CSTF_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CSTF_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.5 組み込みトレース FIFO （ETF）

ETF は 2 つの トレースソース、すなわち CPU コアの ETM と ITM からトレースデータをキャプチャする 8 KB のメモリです。ETFはCoreSight™ トレースメモリコントローラコンポーネントの設計設定です。

ＥＴＦ は以下に記載する 3 種類のモードで使用可能です（モードレジスタで選択）。

1. ハードウェア FIFO モード

   トレースメモリは ATB マスタインタフェースを介してドレインされる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合は受信トレースストリームはストールします。トレースバッファが空でない場合、トレースデータは ATB マスタインタフェースを介して TPIU にドレインされます。

   このモードでの FIFO の役割は、トレースポートに届いたトレース情報の流れを円滑にすることです。トレースデータは本質的に非常にバースト性が強いので、ピークデータ速度がポートの能力を簡単に上回り、結果的にオーバーフローが発生する可能性があります。ETF はトレースポートのデータ速度を安定させます。そのため、ピーク速度ではなく平均速度に応じてサイズ変更が可能です。トレースはトレースポートアナライザーツールによりリアルタイムでチップ外に保存されます。そのため、トレースログが大量になることがあります。

2. ソフトウェア FIFO モード

   トレースメモリは、トレースのキャプチャ中に RRD レジスタによって読み出すことができる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合は受信トレースストリームはストールします。

   このモードでは、トレースを DMA によってシステムメモリやハイスピードインタフェース（SPI、USB など）に転送したり、ソフトウェアによって監視することもできます。なお、ハードウェア FIFO モードと異なり、このモードはプロセッサによって共有されるシステムリソースを使用しているため、侵入型です。

3. サーキュラバッファモード

   トレースメモリはサーキュラバッファとして使用されます。トレースデータはトレースメモリにキャプチャされ、書き込みポインタレジスタが指し示す位置から書き込まれます。トレースメモリがフルの場合でも、STOP コンディションが発生するまで、受信トレースデータはトレースメモリへの上書きを続けます。

   このモードでは、ETF はトレースデータをチップ上に保存するため、トレースログのサイズは ETF SRAM のサイズ（この場合 8 KB）に制限されます。サーキュラバッファとして使用される場合、FIFO がフルになると、受信トレースデータは一番古い保存データを上書きし、一番古いデータは失われます。したがって、トレースバッファの内容は、トレース開始以降のプロセッサの全動作ではなく、バッファが停止した時点までで最新のプロセッサの動作となります。

   トレースが停止した時点でバッファの内容を読み出すには、以下の 3 通りの方法が考えられます。

   – トレースポート経由 - TPIU を有効にした状態で、バッファの内容がトレースポート経由で出力されます。この方法は、ETF_FFCR レジスタの DRAINBUF ビットをセットすることで実施可能です。

   – デバッグポート経由 - デバッガが、システム APB-D を通じてアクセス可能な RRD レジスタ経由でバッファを読み出すことができます。

   – ソフトウェアによる - プロセッサ が RRD レジスタ経由でバッファを読み出すことができます。これはシステムバスから APB-D にアクセスできるためです。

ETF は以下のステート間で遷移します。

- **無効**

  リセット後またはトレースキャプチャが無効なときに、このステートに遷移します。ETF のプログラミングは必ずこのステートで行ってください。
- **実行中**

  トレースキャプチャはこのステートで行われます。無効ステート中にトレースキャプチャを有効にすることで、このステートに遷移します。
- **停止**

  トレースキャプチャはこのステートで停止しますが、バッファの内容は読み出しまたはドレインすることができます。停止イベント（トリガまたは一掃）後に、このステートに遷移します。
- **無効化中**

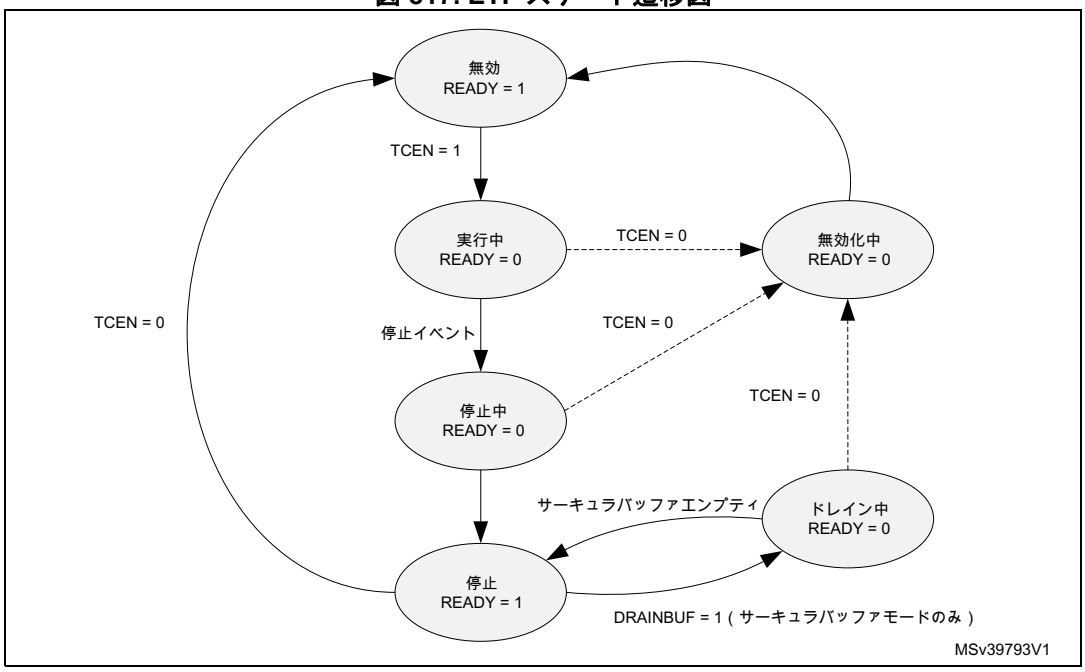
  トレースキャプチャを無効化している間の過渡的なステートです。
- **停止中**

  トレースキャプチャを停止している間の過渡的なステートです。
- **ドレイン中**

  停止ステートでのバッファのドレイン中に、このステートに遷移します。

ステート遷移図を *図 817*に示します。

図 817. ETF ステート遷移図



トレースメモリコントローラ CoreSight™ コンポーネントの詳細情報は、ARM® CoreSight™ Trace Memory Controller Technical Reference Manual を参照してください *[3]*。

ETF レジスタ

ETF RAM サイズレジスタ（ETF_RSZ）

アドレスオフセット：0x004

リセット値：0x0000 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RSZ[30:16]														
	r														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSZ[15:0]															
r															

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:0 **RSZ[30:0]**：RAM サイズ

このフィールドの値は 32 ビットワードの数を示します。

0x800：2048 ワード = 8 KB

ETF ステータスレジスタ（ETF_STS）

アドレスオフセット：0x00C

リセット値：0x0000 001C

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY	FTEMPTY	READY	TRIGD	FULL
											r	r	r	r	r

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **EMPTY**：トレース FIFO エンプティ

このビットは ETF_CTL レジスタの TCEN ビットがハイの場合にのみ有効で、TCEN がローのときはゼロとして読み出されます。

0：トレース FIFO にはデータが格納されています。

1：トレース FIFO は空です。

注：　空のトレースFIFO とは、ETF のパイプラインが空という意味ではありません。後者を示すビットは FTEMPTY です。

ビット 3 **FTEMPTY**：フォーマッタエンプティ

このビットは、トレースキャプチャが停止し、内部パイプラインおよびバッファがすべてドレインされた時点でセットされます。READY とは異なり、バッファのドレインによる影響を受けません。ACQCOMP 出力がこのビット値を反映します。

ビット 2 **READY**：ETF レディ

このビットは、トレースキャプチャが停止し、内部パイプラインおよびバッファがすべてドレインされた時点でセットされます（停止または無効ステート）。

ビット 1 **TRIGD**：トリガ

トリガビットは、トレースキャプチャの実行中で、TMC がトリガイベントを検出した時点でセットされます。このビットは無効ステートを終了するときにクリアされます。

このビットはサーキュラバッファモードでのみ機能し、他のすべてのモードでは常にローです。

このビットは、TMC からのフォーマットされた出力トレースデータにトリガが埋め込まれていることを示すものではありません。出力トレースストリームでのトリガ表示は、フォーマッタおよび一掃の制御レジスタ ETF_FFCR をプログラムすることにより決定されます。

ビット 0 **FULL**：トレースバッファフル

サーキュラバッファモードでは、このフラグは RAM 書き込みポインタがバッファの先頭に折り返すときにセットされ、ETF_CTL レジスタの TCEN ビットがクリアおよびセットされるまでは、その値のままです。

ソフトウェアおよびハードウェア FIFO モードでは、このフラグはトレースメモリ内の現在の空きが ETF_BUFWM レジスタにプログラムされた値以下、すなわちフィルレベルが MEM_SIZE - BUFWM で求められる値以上であることを示します。

このビットは無効ステートを終了するときにクリアされます。FULL 出力がこのレジスタのビット値を反映します。

ETF RAM 読み出しデータレジスタ（ETF_RRD）

アドレスオフセット：0x010

リセット値：不明

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RRD[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RRD[15:0]															
r															

ビット 31:0 **RRD[31:0]**：RAM 読み出しデータ

サーキュラバッファモードの場合：

停止ステートで、バッファが空でない場合にこのレジスタを読み出すと、トレースバッファから次のデータのワードが返されます。トレースバッファのすべてを読み出すと、ETF_STS レジスタの EMPTY ビットがセットされ、次の読み出しでは 0xFFFFFFFF が返されます。停止ステートでないときにこのレジスタを読み出すと、0xFFFFFFFF が返されます。

ソフトウェア FIFO モードの場合：

このレジスタを読み出すと、FIFO からデータが返されます。FIFO が空のときにこのレジスタを読み出すと、返されるデータは 0xFFFFFFFF です。

ハードウェア FIFO モードの場合：

このレジスタを読み出すと 0xFFFFFFFF が返されます。

ETF RAM 読み出しポインタレジスタ（ETF_RRP）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RRP[12:0]												
			rw												

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **RRP[12:0]**：RAM 読み出しポインタ

RAM 読み出しポインタのレジスタには、読み出しポインタの値が格納されます。この値を使用して、ETF_RRD レジスタを介して APB インタフェース上のトレースメモリからエントリを読み出します。ポインタはバイトアドレス、すなわち64ビットのアラインメント（ビット 0～3 はゼロとする）を使用してプログラムすることができます。ポインタはフルの 64 ビット FIFO エントリが書き込まれるたびに 8 ずつインクリメントされます。ポインタはその最大値に達すると折り返します。

このレジスタの書き込みができるのは無効ステートのみで、読み出しができるのは、無効ステート、サーキュラバッファモードと SW FIFO モードにおける停止ステート、およびSW FIFO モードにおける実行中と停止中ステートにあるときです。

ETF RAM 書き込みポインタレジスタ（ETF_RWP）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RWP[12:0]												
			rw												

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **RWP[12:0]**：RAM 書き込みポインタ

RAM 書き込みポインタのレジスタには、書き込みポインタの値が格納されます。この値を使用して、ETF_RWD レジスタを介して APB インタフェース上のトレースメモリにエントリを書き込みます。ポインタはバイトアドレス、すなわちアラインメントされた 64 ビット（ビット 0～3 はゼロとする）を使用してプログラムすることができます。ポインタはフルの 64 ビット FIFO エントリが読み出されるたびに 8 ずつインクリメントされます。ポインタはその最大値に達すると折り返します。

このレジスタの書き込みができるのは無効ステート、読み出しができるのは、無効ステート、サーキュラバッファモードと SW FIFO モードにおける停止ステート、およびSW FIFO モードにおける実行中と停止中ステートにあるときです。

ETF トリガカウンタレジスタ（ETF_TRG）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TRG[10:0]										
					rw										

ビット 31:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:0　**TRG[10:0]**：トリガカウンタ

サーキュラバッファモードで、このビットには、TRIGIN 入力の立ち上がりエッジまたは受信トレースストリームのトリガパケット（ATID =7'h7D）のいずれかを検出した後にトレース RAM にキャプチャする 32 ビットワードの数を指定します。指定したデータワード数をキャプチャすると同時に、トリガイベントが発生します。トリガイベントの ETF 動作への影響は FFCR レジスタによって制御されます。

トリガ後にトレース RAM に書き込まれる 32 ビットワードの数は、このレジスタの保存されている値に 1 を足した値です。ETF がソフトウェア FIFO モードまたはハードウェア FIFO モードのとき、このレジスタは無視されます。トリガカウンタがカウントし始めると、TRIGIN または受信トレースストリームのいずれかで追加のトリガが発生した場合、そのトリガはカウンタがゼロに達するまで無視されます。ゼロに達したトリガカウンタは、このレジスタに書き込みをすることで再プログラムされるまでは、ゼロのままです。

トレースキャプチャ停止時のカウンタの状態が次のトレースキャプチャセッションに影響を及ぼさないように、READY がハイになった時点でこのレジスタはクリアされます。無効ステートでないときにこのレジスタに書き込むと、予期せぬ動作を引き起こします。

このレジスタへの読み出しアクセスは、無効ステートまたはサーキュラバッファモードでは常に許可されます。読み出しアクセスにより、トリガカウンタの現在値が返されます。

ETF 制御レジスタ（ETF_CTL）

アドレスオフセット：0x020

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCEN
															rw

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **TCEN**：トレースキャプチャの有効化

書き込み時：

0：トレースキャプチャを無効にします（実行中、停止中、または停止ステートから無効化中または無効ステートに移行）。

1：トレースキャプチャを有効にします（無効ステートから実行中ステートに移行）。

読み出し時、このビットは、無効化中または無効ステートであればロー、それ以外のステートではハイです。

ETF RAM 書き込みデータレジスタ（ETF_RWD）

アドレスオフセット：0x024

リセット値：不明

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RWD[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RWD[15:0]															
w															

ビット 31:0 **RWD[31:0]**：RAM 書き込みデータ

無効ステートのとき、このレジスタへ書き込むと、そのデータは RWP が指し示す位置へ保存されます。無効ステートでないときのこのレジスタへの書き込みは無視されます。データのメモリ全幅（64 ビット）が書き込まれた場合、そのデータはメモリに書き込まれ、RAM 書き込みポインタは次のメモリワードまでインクリメントされます。

このレジスタは、テストのために使用されます。

ETF モードレジスタ（ETF_MODE）

アドレスオフセット：0x028

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MODE[1:0]**：動作モード

00b：サーキュラバッファモード

このモードでは、トレースメモリはサーキュラバッファとして使用されます。トレースデータはトレースメモリにキャプチャされ、書き込みポインタレジスタが指し示す位置から書き込まれます。トレースメモリがフルの場合でも、STOP コンディションが発生するまで、受信トレースデータはトレースメモリへの上書きを続けます。

01b：ソフトウェア FIFO モード

このモードでは、トレースメモリは、トレースのキャプチャ中に RRD レジスタによって読み出すことができる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合は受信トレースストリームはストールします。

10b：ハードウェア FIFO モード

このモードでは、トレースメモリは ATB マスタインタフェースを介してドレインされる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合は受信トレースストリームはストールします。トレースバッファが空でない場合、トレースデータは ATB マスタインタフェースを通して TPIU にドレインされます。

11b：予約済み

ETF ラッチバッファのフィルレベルレジスタ（ETF_LBUFLVL）

アドレスオフセット：0x02C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LBUFLEVEL[11:0]											
				r											

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **LBUFLEVEL[11:0]**：ラッチバッファのフィルレベル

このレジスタは最後に読み出されたものであるため、読み出すと、トレースメモリの最大フィルレベルが 32 ビットワードで返されます。また、このレジスタを読み出すと、その内容が現在のフィルレベルに更新されます。

無効ステートに入った時点で、このレジスタはその最後の値を保持します。無効ステートの間は、このレジスタから読み出してもその値は変わりません。無効ステートを終了すると、LBUFLEVEL レジスタがクリアされます。

このレジスタはトレースシステムのパフォーマンス分析に使用されます。

ETFの現在のバッファのフィルレベルレジスタ（ETF_CBUFLVL）

アドレスオフセット：0x030

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CBUFLEVEL[11:0]											
				r											

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **CBUFLEVEL[11:0]**：現在のバッファのフィルレベル

このレジスタを読み出すと、トレースメモリの現在のフィルレベルが 32 ビットワードで返されます。
このレジスタは TCEN がローのときにクリアされます。

ETF バッファレベルウォーターマークレジスタ（ETF_BUFWM）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BUFWM[10:0]										
					rw										

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **BUFWM[10:0]**：バッファレベルウォーターマーク

このレジスタにプログラムされた値は、トレースメモリの要求空きレベルの閾値を 32 ビットワードで示します。FIFO 内の空きがこの値以下の場合、すなわち、フィルレベルが MEM_SIZE - BUFWM で求められた値以上のとき、FULL 出力はハイに引き上げられ、STS レジスタの FULL ビットがセットされます。

このレジスタはソフトウェア FIFO モードとハードウェア FIFO モードでのみ使用されます。サーキュラバッファモードでこの機能を用いるには、RWP を要求される空きトリガレベルにプログラミングします。そうすることで、ポインタが折り返した時点で FULL ビットがセットされ、空きレベルが要求レベルを下回ったことを示します。

このレジスタに書き込むことができる最大値は MEM_SIZE から 1 を引いた値です。この場合、FULL ビット出力がアサートされるのは最初の 32 ビットワードがトレースメモリに書き込まれた後です。

無効以外のステートでこのレジスタに書き込むと、予期せぬ動作を引き起こします。

ETF フォーマッタおよび一掃のステータスレジスタ（ETF_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTSTOPPED	FLINPROG
														r	r

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **FTSTOPPED**：フォーマッタ停止

このビットは、ETF_STS レジスタの FTEMPTY ビットと同じように動作します。

ビット 0 **FLINPROG**：一掃の実行中

ATB スレーブポートで一掃を実行中かどうかを示します。このビットは、AFVALIDS 出力の状態を反映します。一掃は ETF_FFCR レジスタの一掃制御ビットによる開始、または ATB マスタポートからのリクエストが可能です。

0：実行中の一掃はありません。
1：一掃の実行中です。

ETF フォーマッタおよび一掃の制御スレジスタ（ETF_FFCR）

アドレスオフセット：0x304

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DRAINBUF	STOPONTRGEV	STOPONFL	Res.	TRIGONFL	TRGONTRGEV	TRGONTRGIN	Res.	FLUSHMAN	FONTRGEV	FONFLIN	Res.	Res.	ENTI	ENFT
	rw	rw	rw		rw	rw	rw		rw	rw	rw			rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DRAINBUF**：ドレインバッファ

このビットを使用して、フォーマッタ停止後に ATB マスタインタフェースを介してトレースデータをドレインする機能を有効にします。これは、サーキュラバッファモードで、トレースデータをトレースメモリにキャプチャし、その後キャプチャされたトレースを ATB マスタインタフェースを介してドレインするのに役立つ機能です。

停止ステートでこのビットに 1 を書き込むと、トレースバッファの内容がATB マスタインタフェースを介してドレインされ始めます。このビットの読み出し値は常に 0 です。ドレイン実行中は ETF_STS レジスタの READY ビットはローになります。

このビットは、ETF がサーキュラバッファモードで、かつフォーマットが有効、すなわち ETF_FFCR レジスタの ENFT ビットがセットされている場合にのみ機能します。ETF が上記以外のモード、すなわち停止ステートでないときにこのビットをセットすると、予期せぬ動作を引き起こします。

サーキュラバッファモードでトレースキャプチャが完了した時点で、キャプチャされたトレースすべてを同じ方法によってトレースメモリから取得しなければなりません。この方法とは、RRD 読み出しによりすべてのトレースデータを読み出すか、または DRAINBUF ビットをセットすることによってすべてのトレースデータをドレインする方法です。キャプチャされたデータの一部を RRD から読み出した後で DRAINBUF ビットをセットすると、予期せぬ動作を引き起こします。

ビット 13 **STPONTRGEV**：トリガイベント時の停止

0：影響なし。

1：トリガイベント発生時にトレースキャプチャを停止します。

ソフトウェア FIFO モードまたはハードウェア FIFO モードでこのビットをセットして ETF を有効にすると、予期せぬ動作を引き起こします。

ビット 12 **STOPONFL**：一掃時の停止

0：影響なし。

1：一掃完了時にトレースキャプチャを停止します。

一掃が ATB マスタインタフェースによって開始される場合、このビットにプログラムされた値に関係なく、その一掃が完了してもフォーマッタは停止されません。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIGONFL**：一掃時のトリガ

0：影響なし。

1：一掃完了時にトレースストリームにあるトリガを示します。

NFT と ENTI のどちらもクリアされている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

一掃が ATB マスタインタフェースによって開始される場合、このビットにプログラムされた値に関係なく、その一掃が完了してもトリガは表示はされません。

ビット 9 **TRGONTRGEV**：トリガイベント時のトリガ

0：影響なし。

1：トリガイベント発生時にトレースストリームにあるトリガを示します。

ENFT と ENTI のどちらの値もはっきり分かっている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

このビットはソフトウェア FIFO モードまたはハードウェア FIFO モードではサポートされません。

ビット 8 **TRGONTRGIN**：トリガ入力時のトリガ

0：影響なし。

1：TRIGIN 入力で立ち上がりエッジが検出されたときにトレースストリームにトリガがあることを示します。

ENFT と ENTI のどちらの値もはっきり分かっている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FLUSHMAN**：マニュアル一掃

0：影響なし。

1：トレース FIFO とパイプラインを一掃します。

このビットは、一掃完了時に自動的にクリアされます。ETF_CTL レジスタの TCEN ビットが 0 のとき、このビットへの書き込みは無視されます。

ビット 5 **FONTRGEV**：トリガイベント時の一掃

0：影響なし。

1：トリガイベントが発生したら、トレース FIFO とパイプラインを一掃します。

このビットはソフトウェア FIFO モードまたはハードウェア FIFO モードではサポートされません。STPONTRGEV がセットされると、このビットは無視されます。

ビット 4 **FONFLIN**：一掃入力時の一掃

0：影響なし。

1：FLUSHIN 入力で立ち上がりエッジが検出されたら、トレース FIFO とパイプラインを一掃します。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENTI**：トリガ挿入の有効化

このビットをセットすると、フォーマットされたトレースストリームへのトリガの挿入が有効になります。トリガはトレースストリーム内の ATID 7'h7D を用いて、データ 8'h00 の 1 バイトを挿入することで表示されます。さらに、トレースストリームでのトリガ表示は、FFCR レジスタのレジスタビット TRIGONFL、TRGONTRGEV、および TRGONTRGIN によっても制御されます。このビットは、READY がハイのとき、および TCEN がローのときにのみ変更可能です。このビットは、このレジスタの ENFT レジスタビットがセットされているときにのみ有効です。ENFT がローのときに ENTI ビットがハイにセットされると、フォーマット機能が有効化されます。

ビット 0 ENFT：フォーマットの有効化

0：フォーマットは無効です。受信トレースデータのトレース転送元は 1 つと考えられます。

1：フォーマットは有効です。

トレースキャプチャが有効で、かつフォーマッタが無効なときに ETF が複数の ATID を受信すると、トレースデータのインターリーブが起こります。フォーマットの無効化はサーキュラバッファモードでのみサポートされます。サーキュラバッファモード以外のモードで ENFT をローにして ETF を有効にすると、フォーマット機能が有効化されます。ENFT がローのときに ENTI ビットがハイにセットされると、フォーマット機能が有効化されます。

このビットは無効ステートでは無視されます。

ETF 周期的同期カウンタレジスタ（ETF_PSCR）

アドレスオフセット：0x308

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSCOUNT[4:0]				
											rw				

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **PSCOUNT[4:0]**：同期カウンタの再ロード値

同期カウンタの再ロード値を決定します。再ロード値は、次にカウンタがゼロに達した時点で有効になります。このレジスタを読み出すと、このレジスタにプログラムされた再ロード値が返されます。リセット時、このレジスタは 0xA にセットされており、1024 バイトの同期周期に相当します。

0x0：同期は無効です。
0x1～0x6：予約済み
0x7～0x1B：同期周期は $2^{PSCOUNT}$ バイトです。
0x1C～0x1F：予約済み

ETF 要求タグ設定レジスタ（ETF_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

ETF 要求タグクリアレジスタ（ETF_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw			

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

ETF ロックアクセスレジスタ（ETF_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0　**ACCESS_W[31:0]**：ETF レジスタアクセス有効化

プロセッサコアによる一部の ETF レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

ETF ロックステータスレジスタ（ETF_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：ETF_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。
1：書き込みアクセスはブロックされます。読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。
1：ロック制御メカニズムが実装されています。

ETF 認証ステータスレジスタ（ETF_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x0：実装されません。

ETF デバイス設定レジスタ（ETF_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 01C0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	MEMWIDTH[2:0]			CONFIGTYP[1:0]		CLK SCHEM	ATBINPORTCNT[4:0]				
					r			r		r	r				

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **MEMWIDTH[2:0]**：メモリインタフェースのデータバス幅
0x3：64 ビット（32 ビット ATB データに相当）

ビット 7:6 **CONFIGTYP[1:0]**：コンポーネントの設定タイプ（ETB、ETR、または ETF）
0x2：ETF

ビット 5 **CLKSCHEM**：RAM クロック制御スキーム（同期または非同期）
0：同期

ビット 4:0 **ATBINPORTCNT[4:0]**：ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

ETF デバイスタイプ識別子レジスタ（ETF_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0032

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類

0x3：トレースデータを ATB スレーブインタフェースから、ATB マスタインタフェースを介してドレイン可能な RAM にキャプチャします。

ビット 3:0 **MAJORTYPE[3:0]**：大分類

0x2：コンポーネントは ATB マスタインタフェースを有しており、それを介してハードウェア FIFO モードでトレースデータをドレインできるため、トレースリンクに分類されます。

ETF CoreSight ペリフェラル ID レジスタ 4（ETF_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

ETF CoreSight ペリフェラル ID レジスタ 0（ETF_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0061

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x61：ETF 部品番号

ETF CoreSight ペリフェラル ID レジスタ 1（ETF_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x9：ETF 部品番号

ETF CoreSight ペリフェラル ID レジスタ 2（ETF_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

ETF CoreSight ペリフェラル ID レジスタ 3（ETF_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]：**メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

ETF CoreSight コンポーネント ID レジスタ 0（ETF_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

ETF CoreSight コンポーネント ID レジスタ 1（ETF_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

ETF CoreSight コンポーネント ID レジスタ 2（ETF_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

ETF CoreSight コンポーネント ID レジスタ 3（ETF_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

ETF レジスタのマップとリセット値

表 572. ETF レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004	ETF_RSZ	Res.	RSZ[30:0]																														
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0x00C	ETF_STS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY	FTEMPTY	READY	TRIGD	FULL
	リセット値																												1	1	1	0	0
0x010	ETF_RRD	RRD[31:0]																															
	リセット値																																
0x014	ETF_RRP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRP[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	ETF_RWP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWP[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	ETF_TRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRG[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x020	ETF_CTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCEN
	リセット値																																0
0x024	ETF_RWD	RWD[31:0]																															
	リセット値																																
0x028	ETF_MODE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[1:0]	
	リセット値																															0	0
0x02C	ETF_LBUFLVL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LBUFLEVEL[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x030	ETF_CBUFLVL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBUFLEVEL[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x034	ETF_BUFWM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUFWM[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x300	ETF_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTSTOPPED	FLINPROG
	リセット値																															1	0
0x304	ETF_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DRAINBUF	STPONTRGEV	STOPONFL	Res.	TRGONFL	TRGONTRGEV	TRGONTRGIN	Res.	FLUSHMAN	FONTRGEV	FONFLIN	Res.	Res.	ENTI	ENFT
	リセット値																		0	0	0		0	0	0		0	0	0			0	0
0x308	ETF_PSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSCOUNT[4:0]				
	リセット値																												0	1	0	1	0

表 572. ETF レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFA0	ETF_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1
0xFA4	ETF_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFB0	ETF_LAR	ACCESS_W[31:0]																															
	リセット値																																
0xFB4	ETF_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	ETF_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFC8	ETF_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEMWDTH [2:0]			CONFIGTYP [1:0]		CLKSCHEM	ATBINPORTCNT [4:0]				
	リセット値																						0	1	1	1	0	0	0	0	0	0	0
0xFD0	ETF_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYPE [3:0]			
	リセット値																									0	0	1	1	0	0	1	0
0xFD0	ETF_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	ETF_PIDR5	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
	リセット値																																
0xFD8	ETF_PIDR6	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
	リセット値																																
0xFDC	ETF_PIDR7	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
	リセット値																																
0xFE0	ETF_PIDR0	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PARTNUM[7:0]							
	リセット値																									0	1	1	0	0	0	0	1
0xFE4	ETF_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1

表 572. ETF レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE8	ETF_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	ETF_PIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	ETF_CIDR0	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	ETF_CIDR1	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	ETF_CIDR2	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	ETF_CIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.6 トレースポートインタフェースユニット（TPIU）

TPIU は CoreSight™ コンポーネントの一つで、トレースストリームをフォーマットし、それを外部トレースポート信号に出力します。TPIU には、受信トレースデータ用の ATB スレーブポートが 1つ搭載されています。トレースポートは同期パラレルポートで、クロック出力 TRACECK と 4つのデータ出力 TRACED(7:0) から成ります。トレースポートの幅はレンジ 1～8 でプログラム可能です。より幅の狭いポート設定を使用すると、必要なテスト点／コネクタピンの数が減り、他の目的のために入出力ポートを解放することができます。しかし、それによりトレースポートの帯域幅が制限され、リアルタイムで出力できるトレース情報量の減少につながります。トレースが TPIU に送信される前に DBGMCU 制御レジスタの TRACECLKEN ビットをセットすることで、TRACECK 出力を有効にする必要があります。さらに、TRACECK 周波数は RCC でプログラムすることができます。

トレースポートインタフェース CoreSight コンポーネントの詳細情報は、ARM® CoreSight™ SoC-400 Technical Reference Manual を参照してください *[2]*。

TPIU レジスタ

TPIU サポートされるポートサイズのレジスタ（TPIU_SUPPSIZE）

アドレスオフセット：0x000

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PORTSIZE[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PORTSIZE[15:0]															
r															

ビット 31:0 **PORTSIZE[31:0]**：サポートされる 1～32　ピンのトレースポートサイズビット n-1 がセットされている場合、ポートサイズnがサポートされることを示します。

0x0000 000F：ポートサイズ 1～4 がサポートされます。

TPIU 現在のポートサイズのレジスタ（TPIU_CURPSIZE）

アドレスオフセット：0x004

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PORTSIZE[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PORTSIZE[15:0]															
rw															

ビット 31:0 **PORTSIZE[31:0]**：現在のトレースポートサイズ

ビット n-1 がセットされている場合、現在のポートサイズが n ピンであることを示します。n の値はサポートされるポートサイズレンジ内（1～4）でなければなりません。1 ビットのみをセットしてください。そうしないと、予期せぬ動作を引き起こす場合があります。このレジスタの変更は必ずフォーマッタの停止時に行ってください。

TPIU サポートされるトリガモードのレジスタ（TPIU_SUPTRGM）

アドレスオフセット：0x100

リセット値：0x0000 011F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRG RUN	TRGD
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCOUNT8	Res.	Res.	Res.	MULT 64K	MULT 256	MULT 16	MULT4	MULT2
							r				r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **TRGRUN**：トリガ実行中

0：トリガは発生していません、またはカウンタが 0 です。

1：トリガが発生し、カウンタが 0 ではありません。

ビット 16 **TRIGD**：トリガ

0：トリガは発生していません。

1：トリガが発生し、カウンタが 0 に達しました。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **TCOUNT8**：8 ビットカウンタレジスタ

1：実装されます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MULT64K**：トリガカウンタに 65536 を乗ずる操作のサポート
1：サポートされます。

ビット 3 **MULT256**：トリガカウンタに 256 を乗ずる操作のサポート
1：サポートされます。

ビット 2 **MULT16**：トリガカウンタに 16 を乗ずる操作操作のサポート
1：サポートされます。

ビット 1 **MULT4**：トリガカウンタに 4 を乗ずる操作操作のサポート
1：サポートされます。

ビット 0 **MULT2**：トリガカウンタに 2 を乗ずる操作操作のサポート
1：サポートされます。

TPIU トリガカウンタ値レジスタ（TPIU_TRGCNT）

アドレスオフセット：0x104

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGCOUNT[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TRIGCOUNT[7:0]**：トリガ遅延表示の有効化
外部接続されているトレースキャプチャまたはストレージデバイスへのトリガ表示の遅延を有効にします。このカウンタは 8 ビット幅しかなく、トリガ乗算器レジスタ 0x108 のカウンタ乗算器と組み合わせて使用されるように作られています。トリガ起動時、乗算器と併用されるこの値は、トリガが表示される前のワード数となります。トリガカウンタが 0 に達すると、ここに書き込まれた値が再ロードされます。このレジスタに書き込むと、トリガカウンタ値がリセットされますが、乗算器の値がリセットされることはありません。このレジスタを読み出すと、現在のカウントではなく、プリセットされた値が返されます。

TPIU トリガ乗算器レジスタ（TPIU_TRGMULT）

アドレスオフセット：0x108

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MULT 64K	MULT 256	MULT 16	MULT4	MULT2
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MULT64K**：トリガカウンタに 65536 を乗ずる操作

0：無効

1：有効

ビット 3 **MULT256**：トリガカウンタに 256 を乗ずる操作

0：無効

1：有効

ビット 2 **MULT16**：トリガカウンタに 16 を乗ずる操作

0：無効

1：有効

ビット 1 **MULT4**：トリガカウンタに 4 を乗ずる操作

0：無効

1：有効

ビット 0 **MULT2**：トリガカウンタに 2 を乗ずる操作

0：無効

1：有効

TPIU サポートされるテストパターン／モードのレジスタ（TPIU_SUPTPM）

アドレスオフセット：0x200

リセット値：0x0003 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
												r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PCONTEN**：連続モード のサポート

1：サポートされます。

ビット 16 **PTIMEEN**：タイムモードのサポート

1：サポートされます。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PATF0**：FF/00 パターン のサポート

FF/00 パターンがトレースポート経由の出力としてサポートされるかを示します。

1：サポートされます。

ビット 2 **PATA5**：AA/55 パターン のサポート
AA/55 パターンがトレースポート経由の出力としてサポートされるかを示します。
1：サポートされます。

ビット 1 **PATW0**：ウォーキング 0 のパターン のサポート
ウォーキング 0 のパターンがトレースポート経由の出力としてサポートされるかを示します。
1：サポートされます。

ビット 0 **PATW1**：ウォーキング 1 のパターン のサポート
ウォーキング 1 のパターンがトレースポート経由の出力としてサポートされるかを示します。
1：サポートされます。

TPIU 現在のテストパターン／モードのレジスタ（TPIU_CURTPM）

アドレスオフセット：0x204

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
												rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PCONTEN**：連続モードの有効化
0：無効
1：有効

ビット 16 **PTIMEEN**：タイムモードの有効化
0：無効
1：有効

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PATF0**：FF/00 パターンの有効化
FF/00 パターンがトレースポート経由の出力として有効かを示します。
0：無効
1：有効

ビット 2 **PATA5**：AA/55 パターンの有効化
AA/55 パターンがトレースポート経由の出力として有効かを示します。
0：無効
1：有効

ビット 1 **PATW0**：ウォーキング 0 のパターン の有効化
ウォーキング 0 のパターンがトレースポート経由の出力として有効かを示します。
0：無効
1：有効

ビット 0 **PATW1**：ウォーキング 1 のパターン の有効化
ウォーキング 1 のパターンがトレースポート経由の出力として有効かを示します。
0：無効
1：有効

TPIU テストパターン繰り返しカウンタのレジスタ（TPIU_TPRCR）

アドレスオフセット：0x208

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PATTCOUNT[7:0]**：TRACECLKIN サイクルの数

このフィールドは 8 ビットのカウンタ値で表され、1 つのパターンが次のパターンに切り替わるまでに実行するのに必要な TRACECLKIN サイクルの数を示します。

TPIU フォーマッタおよび一掃のステータスレジスタ（TPIU_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0002

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TCPRESENT**：TRACECTL 出力ピンの使用可能性

オプションの TRACECTL 出力ピンを使用可能かどうかを示します。

0：TRACECTL ピンはこのデバイスには存在しません。

ビット 1 **FTSTOPPED**：フォーマッタ停止

フォーマッタは停止リクエスト信号を受信しており、すべてのトレースデータおよびポストアンブルが送信されます。ATB インタフェースでの追加のトレースデータは無視されます。

0：フォーマッタは停止していません。
1：フォーマッタは停止しています。

ビット 0 **FLINPROG**：一掃の実行中

ATB スレーブポートで一掃を実行中かどうかを示します。このビットは、AFVALIDS 出力の状態を反映します。一掃は TPIU_FFCR レジスタの一掃制御ビットによって開始することができます。

0：実行中の一掃はありません。
1：一掃の実行中です。

TPIU フォーマッタおよび一掃の制御レジスタ（TPIU_FFCR）

アドレスオフセット：0x304

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	STOP TRIG	STOPF L	Res.	TRIG FL	TRIG EVT	TRIGIN	Res.	FON MAN	FON TRIG	FON FLIN	Res.	Res.	ENF CONT	EN FTC
		rw	rw		rw	rw	rw		rw	rw	rw			rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **STOPTRIG**：トリガイベント時の停止

0：影響なし。

1：トリガイベント発生時にフォーマッタを停止します。

ビット 12 **STOPFL**：一掃時の停止

0：影響なし。

1：一掃完了時にフォーマッタを停止します。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIGFL**：一掃時のトリガ

0：影響なし。

1：一掃完了時にトレースストリームにあるトリガを示します。

ビット 9 **TRIGEVT**：トリガイベント時のトリガ

0：影響なし。

1：トリガイベント発生時にトレースストリームにあるトリガを示します。

ビット 8 **TRIGIN**：トリガ入力時のトリガ

0：影響なし。

1：システム CTI からの TRIGIN 入力がアサートされたときにトレースストリームにあるトリガを示します。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FONMAN**：マニュアル一掃の生成

0：影響なし。

1：トレースを一掃します。

このビットは、一掃完了時に自動的にクリアされます。

ビット 5 **FONTRIG**：トリガイベント時の一掃

トリガカウンタが 0 に達したとき、またはシステム CTI からの TRIGIN 入力がハイのときにトリガカウンタが 0 であれば、トリガイベントが発生します。

0：影響なし。

1：トリガイベントが発生したら、トレースを一掃します。

ビット 4 **FONFLIN**：一掃入力時の一掃

0：影響なし。

1：システム CTI からの FLUSHIN 入力がアサートされたら、トレースを一掃します。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENFCONT**：連続フォーマットの有効化

0：連続フォーマットは無効です。

1：連続フォーマットは有効です。

ビット 0 **ENFTC**：フォーマットされたトレースへのトリガ組み込みの有効化

0：フォーマットは無効です。

1：フォーマットは有効です。

TPIU フォーマッタ同期カウンタのレジスタ（TPIU_FSCR）

アドレスオフセット：0x308

リセット値：0x0000 0040

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CYCCOUNT[4:0]				
											rw				

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **CYCCOUNT[4:0]**：TPA の効果的使用の有効化

キャプチャデバイスの大容量のストレージを無駄にすることなく、サイズの異なる複数の TPA を効果的に使用できるようにします。このカウンタには、最後の 128 ビット同期パケット以降のフォーマッタフレームの数が格納されます。12 ビットのカウンタで、最大カウント値は 4096 です。この値は 65536 バイトごとの同期、すなわち 4096 パケット x パケット当たり 16 バイトに等しい数です。デフォルトは 1024 バイトごと、すなわち 64 フォーマッタフレームごとの同期パケットに対して設定されます。フォーマッタが連続モード用に設定されている場合は、フルワードとハーフワードの同期フレームが通常動作中に挿入されます。このような状況では、カウント値はフル同期パケット間の完了フレームの最大数となります。

TPIU 要求タグ設定レジスタ（TPIU_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

TPIU 要求タグクリアレジスタ（TPIU_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

TPIU ロックアクセスレジスタ（TPIU_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:15]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：TPIU レジスタアクセスの有効化

プロセッサコアによる一部の TPIU レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

TPIU ロックステータスレジスタ（TPIU_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK TYPE	LOCK GRANT	LOCK EXIST
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：TPIU_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。
1：書き込みアクセスはブロックされます。読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。
1：ロック制御メカニズムが実装されています。

TPIU 認証ステータスレジスタ（TPIU_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 5:4 **SID[1:0]：**保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 1:0 **NSID[1:0]：**無保護の侵入型デバッグのセキュリティレベル
0x0：実装されません。

TPIU デバイス設定レジスタ（TPIU_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 00A0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWO UART NRZ	SWO MAN	TCLK DATA	FIFO SIZE[2:0]			CLK RELAT	MAXNUM[3:0]				
				r	r	r	r			r	r				

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWOUARTNRZ：**SWO UART または NRZ のサポート
シリアルワイヤ出力の UART または NRZ がサポートされるかを示します。

0：サポートされません

ビット 10 **SWOMAN：**SWO マンチェスタフォーマットのサポート
シリアルワイヤ出力のマンチェスタ符号化フォーマットがサポートされるかを示します。

0：サポートされません

ビット 9 **TCLKDATA：**トレースクロックおよびデータのサポート
0：サポートされません

ビット 8:6 **FIFOSIZE[2:0]**：FIFO のサイズ（2 の累乗値)
0x2：FIFO のサイズ = 4 （16 バイト）

ビット 5 **CLKRELAT：**ATB クロックと TRACECLKIN の関係
ATB クロックと TRACECLKIN の関係（同期または非同期）を示します。

1：非同期

ビット 4:0 **MAXNUM[4:0]：**ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

TPIU デバイスタイプ識別子レジスタ（TPIU_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0011

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類

0x1：トレースポートコンポーネント

ビット 3:0 **MAJORTYPE[3:0]**：大分類

0x1：トレースシンクコンポーネント

TPIU CoreSight ペリフェラル ID レジスタ 4（TPIU_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

TPIU CoreSight ペリフェラル ID レジスタ 0（TPIU_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0012

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x12：TPIU 部品番号

TPIU CoreSight ペリフェラル ID レジスタ 1（TPIU_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]

0x9：TPIU 部品番号

TPIU CoreSight ペリフェラル ID レジスタ 2（TPIU_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 004B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x4：r0p5

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

TPIU CoreSight ペリフェラル ID レジスタ 3（TPIU_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

TPIU CoreSight コンポーネント ID レジスタ 0（TPIU_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]
0x0D：コモン ID 値

TPIU CoreSight コンポーネント ID レジスタ 1（TPIU_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

TPIU CoreSight コンポーネント ID レジスタ 2（TPIU_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

TPIU CoreSight コンポーネント ID レジスタ 3（TPIU_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

TPIU レジスタのマップとリセット値

表 573. TPIU レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	TPIU_SUPPSIZE	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x004	TPIU_CURPSIZE	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x100	TPIU_SUPTRGM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGRUN	TRIGD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCOUNT8	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
	リセット値															0	0								1				1	1	1	1	1
0x104	TPIU_TRGCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGCOUNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x108	TPIU_TRGMULT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
	リセット値																												0	0	0	0	0
0x200	TPIU_SUPTPM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCONTEN	PTIMEEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
	リセット値															1	1													1	1	1	1
0x204	TPIU_CURTPM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCONTEN	PTIMEEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
	リセット値															0	0													0	0	0	0
0x104	TPIU_TPRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATTCOUNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x300	TPIU_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCPRESENT	FTSTOPPED	FLINPROG
	リセット値																														0	1	0
0x304	TPIU_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOPTRIG	STOPFL	Res.	TRiGFL	TRIGEVt	TRIGIN	Res.	FONMAN	FONTRIG	FONFLIN	Res.	Res.	ENFCONT	ENFTC
	リセット値																			0	0		0	0	0		0	0	0			0	0
0x308	TPIU_FSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CYCCOUNT[11:0]											
	リセット値																					0	0	0	0	0	1	0	0	0	0	0	0
0xFA0	TPIU_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1

表 573. TPIU レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFA4	TPIU_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFB0	TPIU_LAR	ACCESS_W[31:0]																															
	リセット値																																
0xFB4	TPIU_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	TPIU_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFC8	TPIU_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
	リセット値																					0	0	0	0	1	0	1	0	0	0	0	0
0xFD0	TPIU_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYPE [3:0]			
	リセット値																									0	0	0	1	0	0	0	1
0xFD0	TPIU_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	TPIU_PIDR5	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
	リセット値																																
0xFD8	TPIU_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	TPIU_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	TPIU_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	0	0	1	0	0	1	0
0xFE4	TPIU_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	TPIU_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	1	0	0	1	0	1	1
0xFEC	TPIU_PIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	TPIU_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1

表 573. TPIU レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF4	TPIU_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	TPIU_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	TPIU_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.5.7 シリアルワイヤ出力（SWO）と SWO トレースファネル（SWTF）

SWO は CoreSight™ コンポーネントの一つで、プロセッサ ITM からのトレースストリームをフォーマットし、それをシングルワイヤ TRACESWO 出力に出力します。SWO トレースファネル（SWTF）は、トレースが有効になる前に Cortex-M7 ITM からのトレースバスを有効にするよう プログラムする必要があります。SWTF レジスタを *表 575*に記載します。

TPIU に比べ、SWO には以下の制限があります。

- フォーマッタはありません.。
- パターンジェネレータはありません。
- 8 ビット ATB 入力
- 同期トレース出力（= TRACEDATA/TRACECLKピン）はありません。
- 一掃のサポート（不要なため）はありません。
- トリガのサポートはありません。

SWO 出力はマンチェスタ符号化フォーマットおよび UART NRZ フォーマットをサポートしています。

シリアルワイヤ出力 CoreSight™ コンポーネントの詳細情報は、ARM® CoreSight™ Components Technical Reference Manual を参照してください *[4]*。

SWO レジスタ

SWO 現在の出力の除数レジスタ（SWO_CODR）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PRESCALER[12:0]												
			rw												

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **PRESCALER[12:0]**：SWO ボーレートスケーリング

ボーレートは PRESCALER - 1 の解で割ったトレースクロック周波数です。ボーレートはすぐに変わるので、トレースソースを停止させ、ポートがアイドルになるまで待ってからこのレジスタに書き込むことをお勧めします。

SWO 選択ピンプロトコルレジスタ（SWO_SPPR）

アドレスオフセット：0x0F0

リセット値：0x0000 0001

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PPROT[1:0]**：ピンプロトコル

0x0：予約済み
0x1：マンチェスタ
0x2：NRZ
0x3：予約済み

SWO フォーマッタおよび一掃のステータスレジスタ（SWO_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTNON STOP	TCPRE SENT	FTSTO PPED	FLIN PROG
												r	r	r	r

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **FTNONSTOP**：フォーマッタ停止なしの設定変更

1：フォーマッタを実行させた状態での設定変更が可能です。

ビット 2 **TCPRESENT**：TRACECTL ピンは SWO には存在しません。
0：TRACECTL ピンが存在しません。

ビット 1 **FTSTOPPED**：フォーマッタ停止
0：フォーマッタは実行中です。
このデバイスでは SWO フォーマッタは停止不能なため、このビットは常に 0 を返します。

ビット 0 **FLINPROG**：一掃の実行中
0：一掃は実行中ではありません。
このデバイスでは SWO 一掃はサポートされないため、このビットは常に 0 を返します。

SWO 要求タグ設定レジスタ（SWO_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット
書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

SWO 要求タグクリアレジスタ（SWO_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

SWO ロックアクセスレジスタ（SWO_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:15]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：SWO レジスタ書き込みアクセスの有効化

プロセッサコアによる一部の SWO レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

SWO ロックステータスレジスタ（SWO_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：SWO_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。

1：書き込みアクセスはブロックされ、読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。

1：ロック制御メカニズムが実装されています。

SWO 認証ステータスレジスタ（SWO_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル

0x0：実装されません。

SWO デバイス設定レジスタ（SWO_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0EA0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWO UART NRZ	SWO MAN	TCLK DATA	FIFO SIZE[2:0]			CLK RELAT	MAXNUM[4:0]				
				r	r	r	r			r	r				

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWOUARTNRZ**：SWO UART または NRZ のサポート
シリアルワイヤ出力の UART または NRZ がサポートされるかを示します。

1：サポートされます。

ビット 10 **SWOMAN**：SWO マンチェスタフォーマットのサポート
シリアルワイヤ出力のマンチェスタ符号化フォーマットがサポートされるかを示します。
1：サポートされます。

ビット 9 **TCLKDATA**：トレースクロックおよびデータのサポート
トレースクロックおよびデータがサポートされるかを示します。

1：サポートされます。

ビット 8:6 **FIFOSIZE[2:0]**：FIFO のサイズ（2 の累乗値）
0x2：FIFO のサイズ = 4 （16 バイト）

ビット 5 **CLKRELAT**：ATB クロックと TRACECLKIN の関係
ATB クロックと TRACECLKIN の関係（同期または非同期）を示します。

1：非同期

ビット 4:0 **MAXNUM[4:0]**：ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

SWO デバイスタイプ識別子レジスタ（SWO_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0011

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類

0x1：トレースポートコンポーネント

ビット 3:0 **MAJORTYPE[3:0]**：大分類

0x1：トレースシンクコンポーネント

SWO CoreSight ペリフェラル ID レジスタ 4（SWO_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

SWO CoreSight ペリフェラル ID レジスタ 0（SWO_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0014

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x14：SWO 部品番号

SWO CoreSight ペリフェラル ID レジスタ 1（SWO_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x9：SWO 部品番号

SWO CoreSight ペリフェラル ID レジスタ 2（SWO_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p0

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

SWO CoreSight ペリフェラル ID レジスタ 3（SWO_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]：**メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

SWO CoreSight コンポーネント ID レジスタ 0（SWO_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

SWO CoreSight コンポーネント ID レジスタ 1（SWO_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

SWO CoreSight コンポーネント ID レジスタ 2（SWO_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

SWO CoreSight コンポーネント ID レジスタ 3（SWO_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

SWO レジスタのマップとリセット値

表 574. SWO レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	SWO_SPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x004	SWO_CPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x010	SWO_CODR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x0F0	SWO_SPPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPROT [1:0]	
	リセット値																															0	1
0x100	SWO_STMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x200	SWO_STPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x300	SWO_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTNONSTOP	TCPRESENT	FTSTOPPED	FLINPROG
	リセット値																													1	0	0	0
0x304	SWO_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFA0	SWO_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET [3:0]			
	リセット値																													1	1	1	1
0xFA4	SWO_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR [3:0]			
	リセット値																													0	0	0	0
0xFB0	SWO_LAR	ACCESS_W[31:0]																															
	リセット値																																
0xFB4	SWO_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	SWO_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																								0	0	0	0		0	0	0	0
0xFC8	SWO_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	1	0	0	0	0	0

表 574. SWO レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	SWO_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYPE [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0xFD0	SWO_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0xFD4	SWO_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	SWO_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	SWO_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	SWO_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	0	0	1	0	1	0	0
0xFE4	SWO_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	SWO_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	SWO_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	SWO_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	SWO_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	SWO_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	SWO_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

SWTF レジスタ

SWTF 制御レジスタ（SWTF_CTRL）

アドレスオフセット：0x000

リセット値：0x0000 0300

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MIN_HOLD_TIME[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENS0
				rw											rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MIN_HOLD_TIME[3:0]**：アービトレーション間に実行するトランザクションの数
0x0：1 トランザクション
:
0xE：15 トランザクション
0xF：予約済み

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **ENS0**：スレーブポート S0 有効化
0：ポートを無効にします。
1：ポートを有効にします。

SWTF 優先度レジスタ（SWTF_PRIORITY）

アドレスオフセット：0x004

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT0[2:0]		
													rw		

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PRIPORT0[2:0]**：スレーブポート S0 の優先度
0：最高優先度
:
7：最低優先度

SWTF 要求タグ設定レジスタ（SWTF_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

SWTF 要求タグクリアレジスタ（SWTF_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

SWTF ロックアクセスレジスタ（SWTF_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：SWTF レジスタ書き込みアクセスの有効化

プロセッサコアによる一部の SWTF レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

SWTF ロックステータスレジスタ（SWTF_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：SWTF_LAR レジスタのサイズ

0：32 ビット

ビット 1 **LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。
1：書き込みアクセスはブロックされ、読み出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。
1：ロック制御メカニズムが実装されています。

SWTF 認証ステータスレジスタ（SWTF_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]：**保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル

0x2：無効

0x3：有効

ビット 1:0 **NSID[1:0]：**無保護の侵入型デバッグのセキュリティレベル

0x2：無効

0x3：有効

SWTF CoreSight デバイス ID レジスタ（SWTF_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0022

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME[3:0]				PORTCNT[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SCHEME[3:0]**：優先度スキーム

0x2：静的優先度

ビット 3:0 **PORTCNT[3:0]**：接続されている入力ポートの数

0x2：２つの入力ポート

SWTF CoreSight デバイスタイプ ID レジスタ（SWTF_TYPEID）

アドレスオフセット：0xFCC

リセット値：0x0000 0012

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DEVTYPEID[7:0]**：デバイスタイプ識別子

0x12：トレースファネル

SWTF CoreSight ペリフェラル ID レジスタ 0（SWTF_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x08：SWTF 部品番号

SWTF CoreSight ペリフェラル ID レジスタ 1（SWTF_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x9：SWTF 部品番号

SWTF CoreSight ペリフェラル ID レジスタ 2（SWTF_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

SWTF CoreSight ペリフェラル ID レジスタ 3（SWTF_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

SWTF CoreSight ペリフェラル ID レジスタ 4（SWTF_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

SWTF CoreSight コンポーネント ID レジスタ 0（SWTF_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

SWTF CoreSight コンポーネント ID レジスタ 1（SWTF_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

SWTF CoreSight コンポーネント ID レジスタ 2（SWTF_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

SWTF CoreSight コンポーネント ID レジスタ 3（SWTF_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

SWTF レジスタのマップとリセット値

表 575. SWTF レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	SWTF_CTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIN_HOLD_TIME[3:0]										ENS1	ENS0
	リセット値																					0	0	1	1							0	0
0x004	SWTF_PRIORITY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT1[2:0]			PRIPORT0[2:0]		
	リセット値																											0	0	1	0	0	0
0xFA0	SWTF_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	SWTF_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFB0	SWTF_LAR	ACCESS_W[31:0]																															
	リセット値																																
0xFB4	SWTF_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	SWTF_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFC8	SWTF_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME[3:0]				PORTCNT[3:0]			
	リセット値																									0	0	1	0	0	0	1	0
0xFCC	SWTF_TYPEID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
	リセット値																									0	0	0	1	0	0	1	0
0xFD0	SWTF_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	SWTF_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	SWTF_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	SWTF_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	SWTF_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	1	0	0	0

表 575. SWTF レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE4	SWTF_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	SWTF_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	SWTF_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND				CMOD			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	SWTF_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	SWTF_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS				PREAMBLE			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	SWTF_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	SWTF_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE							
	リセット値																									1	0	1	1	0	0	0	1

60.5.8 マイクロコントローラデバッグユニット（DBGMCU）

DBGMCU コンポーネントには、デバッグモードで電源やクロックの動作を制御するレジスタが複数内蔵されています。具体的には、デバッガ、またはデバッグソフトウェアの以下の動作を可能にします。

- 低電力モード（SLEEP、STOP、STANDBY ）時に、プロセッサコアへのクロックや電源の供給を維持します。
- 低電力モード（SLEEP、STOP、STANDBY ）時に、システムデバッグおよびトレース用のコンポーネントへのクロックや電源の供給を維持します。
- 一方のプロセッサコアがデバッグモードで停止した場合、特定のペリフェラルへのクロック（CAN、SMBUS タイムアウト、ウォッチドッグ、タイマ、RTC）の供給を停止します。相補出力を持つタイマの場合、カウンタ停止時には (DBGMCU_APB2FZ1 では、TIM1/8/15/16/17 = 1)、その出力は安全のために無効化されます（MOE ビットがリセットされたかのように動作する）。

DBGMCU レジスタは、システムリセットではリセットされず、パワーオンリセットによってのみリセットされます。それらは APB-D バス経由でデバッガのベースアドレス 0xE00E1000 にアクセス可能です。また、両方のプロセッサコアによってベースアドレス 0x5C001000 にアクセスすることもできます。

注：DBGMCU は標準 CoreSight コンポーネントではないので、システム ROM テーブルには含まれません。

参考資料

DBGMCU レジスタ

DBGMCU ID コードレジスタ（DBGMCU_IDC）

アドレスオフセット：0x000

リセット値：0x100X 6450

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

ビット 31:16 **REV_ID[15:0]**：版
0x1001 = リビジョン Z
0x1003 = リビジョン Y

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DEV_ID[11:0]**：デバイス ID
0x450：STM32H7

DBGMCU 設定レジスタ（DBGMCU_CR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	TRGOEN	Res.	Res.	Res.	Res.	Res.	D3DBGCKEN	D1DBGCKEN	TRACECLKEN	Res.	Res.	Res.	Res.
			rw						rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTBY_D3	DBGSTOP_D3	Res.	Res.	Res.	Res.	DBGSTBY_D1	DBGSTOP_D1	DBGSLEEP_D1
							rw	rw					rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **TRGOEN**：外部トリガ出力の有効化
このビットは、双方向トリガピン TRGIO の方向を制御します。

0：入力 - TRGIO は TRGIN に接続されます。
1：出力 - TRGIO は TRGOUT に接続されます。

ビット 27:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **D3DBGCKEN**：D3 デバッグクロックの有効化

このビットは、D3 ドメイン （DAPCLK ドメインを除く）のデバッグコンポーネントが必要でないときに、それらの電源をオフにします。

0：無効 - D3 ドメインのデバッグコンポーネントは無効化され、それらのクロックはゲートされます。
1：有効 - D3 ドメインのデバッグコンポーネントは、対応するドメインクロック（CK_HCLK_D3）がアクティブなときは常にクロック供給されます。

ビット 21 **D1DBGCKEN**：D1 デバッグクロックの有効化

このビットは、D1 ドメインのデバッグコンポーネント （プロセッサコア内のものを除く）が必要でないときに、それらの電源をオフにします。

0：無効 - D1 ドメインのデバッグコンポーネントは無効化され、それらのクロックはゲートされます。
1：有効 - D1 ドメインのデバッグコンポーネントは、対応するドメインクロック（CK_HCLK_D1）がアクティブなときは常にクロック制御されます。

ビット 20 **TRACECLKEN**：トレースポートクロックの有効化

このビットは、トレースポートクロック TRACECLK を有効にします。

0：無効 - TRACECLK は無効です。
1：有効 - TRACECLK はアクティブです。

ビット 19:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DBGSTBY_D3**：D3 の STANDBY モードでデバッグ許可

0：通常動作 - STANDBY モードでは、すべてのクロックが無効化され、ドメインの電源は自動的に切断されます(1)。
1：自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STANDBY モード 中も動作し続け、ドメイン供給は維持され、すべてのデバッグ機能を許可します。STANDBY モード を終了すると、システムリセットが行われます。

ビット 7 **DBGSTOP_D3**：D3 の STOP モードでデバッグ許可

0：通常動作 - STOP モードでは、ドメインクロックが自動的に無効化されます。(2)
1: 自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STOP モード中も動作し続けます。STOP モードから抜けると、クロック設定はSTOP モード終了ステートに設定されます。

ビット 6:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DBGSTBY_D1**：STANDBY モードで D1 ドメインデバッグ許可

0：通常動作 - STANDBY モードでは、すべてのクロックが無効化され、ドメインの電源は自動的に切断されます。
1：自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STANDBY モード 中も動作し続け、ドメイン供給は維持され、すべてのデバッグ機能を許可します。STANDBY モード を終了すると、ドメインリセットが行われます。

ビット 1 **DBGSTOP_D1**：STOP モードで D1 ドメインデバッグ許可

0：通常動作 - STOP モードでは、すべてのクロックが自動的に無効化されます。
1：自動クロック停止は無効 - すべてのアクティブなクロックおよびオシレータは STOP モード中も動作し続け、すべてのデバッグ機能を許可します。STOP モードから抜けると、クロック設定はSTOP モード終了ステートに設定されます。

ビット 0 **DBGSLEEP_D1**：SLEEP モードで D1 ドメインデバッグ許可

0：通常動作 - SLEEP モードでは、プロセッサクロックが自動停止されます。
1：自動クロック停止は無効 - プロセッサクロックは動作し続け、すべてのデバッグ機能を許可します。

1. デバッグポート制御／ステータスレジスタの CDBGPWRUPREQ がアサートされると、STANDBY モードでは、たとえ DBGSTBY_D3 がリセットされても、D3 は電源はオンのままで、DAPCLK はアクティブです。ただし、残りの D3 ドメインクロックは電源オフされます。

2. デバッグポート制御／ステータスレジスタの CDBGPWRUPREQ がアサートされると、STOP モードでは、たとえ DBGSTOP_D3 がリセットされても、D3 は電源はオンのままで、DAPCLK はアクティブです。ただし、残りの D3 ドメインクロックは電源オフされます。

DBGMCU APB3 ペリフェラルフリーズレジスタ CPU（DBGMCU_APB3FZ1）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1	Res.	Res.	Res.	Res.	Res.	Res.
									rw						

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDG1**：デバッグ時の WWDG1 停止

0：通常動作 - WWDG1 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - WWDG1 は、コアがデバッグモードの間は 停止します。

ビット 5:0 予約済みであり、リセット値に保持する必要があります。

DBGMCU APB1L ペリフェラルフリーズレジスタ CPU（DBGMCU_APB1LFZ1）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C3	I2C2	I2C1	Res.	Res.	Res.	Res.	Res.
								rw	rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LPTIM1	TIM14	TIM13	TIM12	TIM7	TIM6	TIM5	TIM4	TIM3	TIM2
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3**：デバッグ時の I2C3 SMBUS タイムアウトの停止

0：通常動作 - I2C3 SMBUS タイムアウトは、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - I2C3 SMBUS タイムアウトは、Cortex-M7 がデバッグモードの間は 停止します。

ビット 22 **I2C2**：デバッグ時の I2C2 SMBUS タイムアウトの停止

0：通常動作 - I2C2 SMBUS タイムアウトは、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - I2C2 SMBUS タイムアウトは、Cortex-M7 がデバッグモードの間は 停止します。

ビット 21 **I2C1**：デバッグ時の I2C1 SMBUS タイムアウトの停止

0：通常動作 - I2C1 SMBUS タイムアウトは、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - I2C1 SMBUS タイムアウト は、コアがデバッグモードの間は 停止します。

ビット 20:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1**：デバッグ時の LPTIM1 停止

0：通常動作 - LPTIM1 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - LPTIM1 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 8 **TIM14**：デバッグ時の TIM14 停止

0：通常動作 - TIM14 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM14 は、Cortex-M7 がデバッグモードの間は 停止します。 f

ビット 7 **TIM13**：デバッグ時の TIM13 停止

0：通常動作 - TIM13 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM13 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 6 **TIM12**：デバッグ時の TIM12 停止

0：通常動作 - TIM12 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM12 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 5 **TIM7**：デバッグ時の TIM7 停止

0：通常動作 - TIM7 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM7 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 4 **TIM6**：デバッグ時の TIM6 停止

0：通常動作 - TIM6 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM6 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 3 **TIM5**：デバッグ時の TIM5 停止

0：通常動作 - TIM5 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM5 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 2 **TIM4**：デバッグ時の TIM4 停止

0：通常動作 - TIM4 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM4 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 1 **TIM3**：デバッグ時の TIM3 停止

0：通常動作 - TIM3 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM3 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 0 **TIM2**：デバッグ時の TIM2 停止

0：通常動作 - TIM2 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - TIM2 は、Cortex-M7 がデバッグモードの間は 停止します。

DBGMCU APB1H ペリフェラルフリーズレジスタ CPU（DBGMCU_APB1HFZ1）

アドレスオフセット：0x044

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCAN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

ビット 31:9　予約済みであり、リセット値に保持する必要があります。

ビット 8　FDCAN：デバッグ時の FDCAN 停止

0：通常動作 - FDCAN は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - FDCAN は、コアがデバッグモードの間は 停止します。

ビット 7:0　予約済みであり、リセット値に保持する必要があります。

DBGMCU APB2 ペリフェラルフリーズレジスタ CPU（DBGMCU_APB2FZ1）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	HRTIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM17	TIM16	TIM15
		rw											rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM8	TIM1
														rw	rw

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29　**HRTIM**：デバッグ時の HRTIM 停止

0：通常動作 - HRTIM は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - HRTIM は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 27:18　予約済みであり、リセット値に保持する必要があります。

ビット 18　**TIM17**：デバッグ時の TIM17 停止

0：通常動作 - TIM17 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM17 は 停止し、TIM17 出力は 無効化されます。

ビット 17　**TIM16**：デバッグ時の TIM16 停止

0：通常動作 - TIM16 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM16 は 停止し、TIM16 出力は 無効化されます。

ビット 16　**TIM15**：デバッグ時の TIM15 停止

0：通常動作 - TIM15 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM15 は 停止し、TIM15 出力は 無効化されます。

ビット 15:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**TIM8**：デバッグ時の TIM8 停止

0：通常動作 - TIM8 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM8 は 停止し、TIM8 出力は 無効化されます。

ビット 0　**TIM1**：デバッグ時の TIM1 停止

0：通常動作 - TIM1 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM1 は 停止し、TIM1 出力は 無効化されます。

DBGMCU APB4 ペリフェラルフリーズレジスタ CPU（DBGMCU_APB4FZ1）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGLSD1	Res.	RTC
													rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LPTIM5	LPTIM4	LPTIM3	LPTIM2	Res.	I2C4	Res.	Res.	Res.	Res.	Res.	Res.	Res.
			rw	rw	rw	rw		rw							

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **WDGLSD1**：デバッグ時の D1 用ロースピードウォッチドッグ停止

0：通常動作 - ウォッチドッグは、コアがデバッグモードの間は カウントし続けます。
1：デバッグ時の停止 - ウォッチドッグは、Cortex-M7 がデバッグモードの間は 停止します。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTC**：デバッグ時の RTC 停止

0：通常動作 - RTC は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - RTC は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPTIM5**：デバッグ時の LPTIM5 停止

0：通常動作 - LPTIM5 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - LPTIM5 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 11 **LPTIM4**：デバッグ時の LPTIM4 停止

0：通常動作 - LPTIM4 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - LPTIM4 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 10 **LPTIM3**：デバッグ時の LPTIM3 停止

0：通常動作 - LPTIM3 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - LPTIM3 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 9 **LPTIM2**：デバッグ時の LPTIM2 停止

0：通常動作 - LPTIM2 は、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - LPTIM2 は、Cortex-M7 がデバッグモードの間は 停止します。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4**：デバッグ時の I2C4 SMBUS タイムアウトの停止

0：通常動作 - I2C4 SMBUS タイムアウトは、コアがデバッグモードの間は 動作し続けます。
1：デバッグ時の停止 - I2C4 SMBUS タイムアウト は、コアがデバッグモードの間は 停止します。

ビット 6:0 予約済みであり、リセット値に保持する必要があります。

参考資料

DBGMCU レジスタのマップとリセット値

表 576. DBGMCU レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DBGMCU_IDC	REV_ID[15:0]																Res.	Res.	Res.	Res.	DEV_ID[11:0]											
	リセット値	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0	1	0	0	0	0
0x004	DBGMCU_CR	Res.	Res.	Res.	TRGOEN	Res.	Res.	Res.	Res.	Res.	D3DBGCKEN	D1DBGCKEN	TRACECLKEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTBD3	DBGSTPD3	Res.	DBGSTBD2	DBGSTPD2	DBGSLPD2	DBGSTBD1	DBGSTPD1	DBGSLPD1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C から 0x030	予約済み																																
0x034	DBGMCU_APB3FZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG1	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																										0						
0x038	予約済み																																
0x03C	DBGMCU_APB1LFZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C3	I2C2	I2C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM1	TIM14	TIM13	TIM12	TIM7	TIM6	TIM5	TIM4	TIM3	TIM2
	リセット値									0	0	0												0	0	0	0	0	0	0	0	0	0
0x040	予約済み																																
0x044	DBGMCU_APB1HFZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCAN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								0								
0x048	予約済み																																
0x04C	DBGMCU_APB2FZ1	Res.	Res.	HRTIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM17	TIM16	TIM15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM8	TIM1
	リセット値			0											0	0	0															0	0
0x050	予約済み																																
0x054	DBGMCU_APB4FZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGLSD1	Res.	Res.	Res.	Res.	Res.	LPTIM5	LPTIM4	LPTIM3	LPTIM2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0						0	0	0	0									
0x058	予約済み																																

60.6 Cortex-M7 デバッグ機能詳細

Cortex-M7 サブシステムは、以下のCoreSight コンポーネントの機能を備えています。

- ROM テーブル
- システム制御空間（SCS）
- ブレークポイントユニット（FPB）
- データウォッチポイントおよびトレースユニット（DWT）
- 計装トレースマクロセル（ITM）
- 組み込みトレースマクロセル（ETM）
- クロストリガインタフェース（CTI）

上記のコンポーネントへは、Cortex-M7 AHB-AP およびその関連の AHBD バスを介してデバッガからアクセス可能です。

60.6.1 Cortex-M7 の ROM テーブル

ROM テーブルは CoreSight™ コンポーネントの一つで、AHBD を介してアクセス可能なすべての CoreSight デバッグコンポーネントのベースアドレスを格納します。このようなテーブルを使用することにより、デバッガは CoreSight システムのトポロジを自動検出することができます。

Cortex-M7 サブシステムには 2 つの ROM テーブルがあります。

- Cortex-M7 CPU の ROM テーブル

  このテーブルは Cortex-M7 AHB-AP の BASE レジスタによって指し示されます。テーブルには、ETM と CTI のベースアドレスポインタ、ならびに Cortex-M7 CPU の ROM テーブルのベースアドレスポインタが格納されます。
- Cortex-M7 PPB （プライベートペリフェラルバス）の ROM テーブル

  このテーブルには、デバッガが CPU コアを識別できるようにする Cortex-M7 システム制御空間レジスタ、および Cortex-M7 サブシステムの他の CoreSight コンポーネント（FPB、DWT、および ITM）を指し示すポインタが格納されます。

CPU の ROM テーブルは 4 KB、すなわち AHBD アドレス空間の 32 ビット幅の領域（0xE00FE000～0xE00FEFFC）を占有します。

表 577. Cortex-M7 CPU の ROM テーブル

ROM テーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ	エントリ
0xE00FE000	Cortex-M7 PPB の ROM テーブル	0xE00FF000	0x00001000	4 KB	0x00001003
0xE00FE004	Cortex-M7 ETM	0xE0041000	0xFFF43000	4 KB	0xFFF43003
0xE00FE008	Cortex-M7 CTI	0xE0043000	0xFFF44000	4 KB	0xFFF44003
0xE00FE00C	予約済み	-	-	-	0x1FF02002
0xE00FE010	テーブルの先頭	-	-	-	0x00000000
0xE00FE010 から 0xE00FEFC8	予約済み	-	-	-	0x00000000
0xE00FEFCC から 0xE00FEFFC	ROM テーブルレジスタ	-	-	-	*表 579*を参照してください。

Cortex-M7 PPB の ROM テーブルは 4 KB、すなわち APB-D アドレス空間の 32 ビット幅の領域（0xE00FF000～0xE00FFFFC）を占有します。

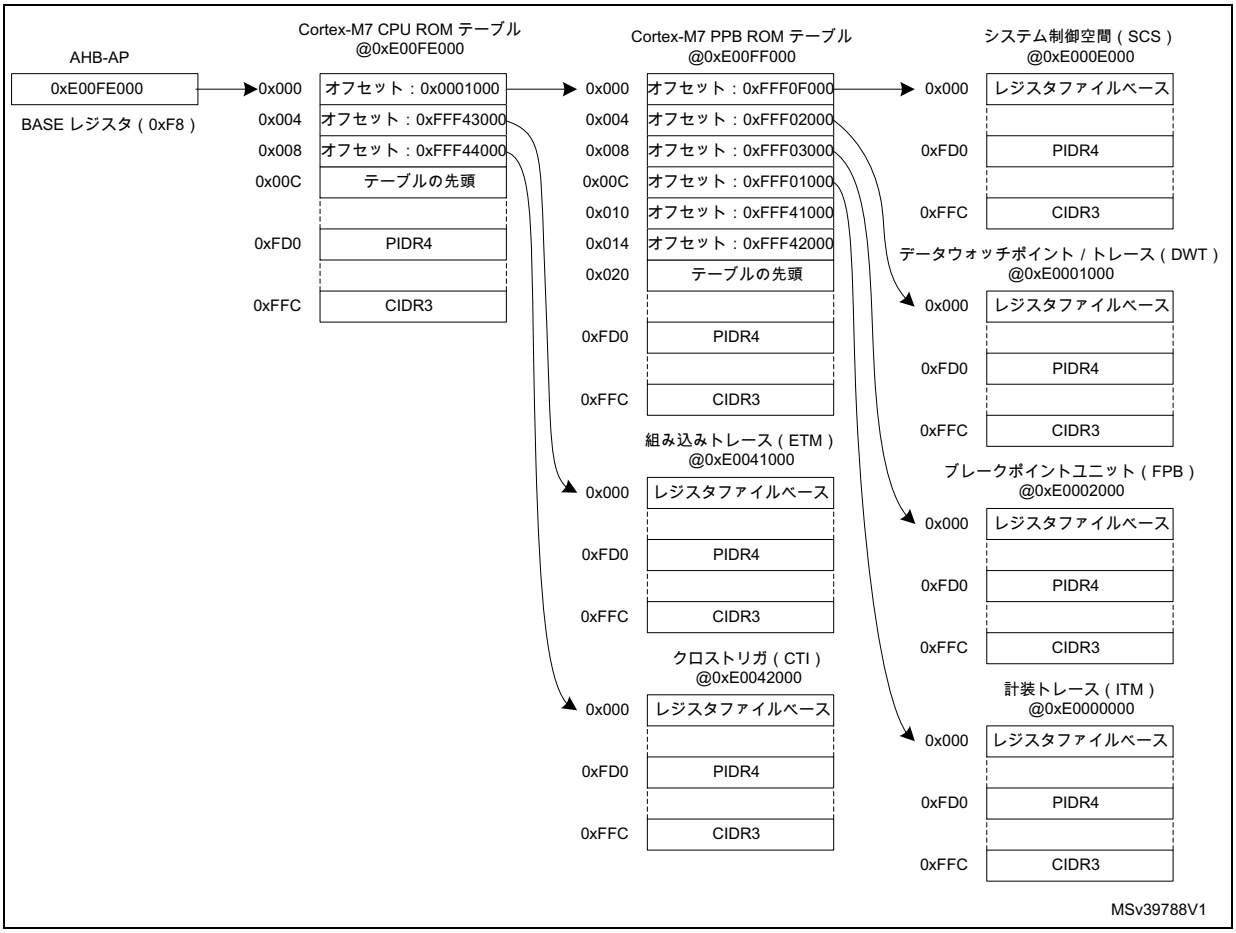
表 578. Cortex-M7 PPB の ROM テーブル

ROM テーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ	エントリ
0xE00FF000	SCS	0xE000E000	0xFFF0F000	4 KB	0xFFF0F003
0xE00FF004	DWT	0xE0001000	0xFFF02000	4 KB	0xFFF02003
0xE00FF008	FPB	0xE0002000	0xFFF03000	4 KB	0xFFF03003
0xE00FF00C	ITM	0xE0000000	0xFFF01000	4 KB	0xFFF01003
0xE00FF010	TPIU(1)	0xE0040000	0xFFF41000	4 KB	0xFFF41002
0xE00FF014	ETM (1)	0xE0041000	0xFFF42000	4 KB	0xFFF42002
0xE00FF018	テーブルの先頭	-	-	-	0x00000000
0xE00FF01C から 0xE00FFFC8	予約済み	-	-	-	0x00000000
0xE00FFFCC から 0xE00FFFFC	ROM テーブルレジスタ	-	-	-	*表 580*を参照してください。

1. TPIU と ETM はデフォルトでこのテーブルに含まれますが、それらが存在しないことを示すためにビット 0 がりセットされます。

Cortex-M7 サブシステム内の CoreSight™ コンポーネントのトポロジを *図 818*に示します。

図 818. Cortex-M7 の CoreSight トポロジ



Cortex-M7 CPU の ROM レジスタ

CPU ROM メモリタイプレジスタ（M7_CPUROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
															r

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSMEM**：システムメモリの有無

1：このバス上にシステムメモリがあります。

CPU ROM CoreSight ペリフェラル ID レジスタ 4（M7_CPUROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC 継続コード

CPU ROM CoreSight ペリフェラル ID レジスタ 0（M7_CPUROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0c8

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0xC8：Cortex-M7 プロセッサの ROM テーブル

CPU ROM CoreSight ペリフェラル ID レジスタ 1（M7_CPUROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B4

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x4：Cortex-M7 プロセッサの ROM テーブル

CPU ROM CoreSight ペリフェラル ID レジスタ 2（M7_CPUROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョン r0p0

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

CPU ROM CoreSight ペリフェラル ID レジスタ 3（M7_CPUROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CPU ROM CoreSight コンポーネント ID レジスタ 0（M7_CPUROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

CPU ROM CoreSight コンポーネント ID レジスタ 1（M7_CPUROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0x1：ROM テーブルコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

CPU ROM CoreSight コンポーネント ID レジスタ 2（M7_CPUROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

CPU ROM CoreSight コンポーネント ID レジスタ 3（M7_CPUROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値

表 579. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	M7_CPUROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD0	M7_CPUROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD4	M7_CPUROM_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	M7_CPUROM_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_CPUROM_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_CPUROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									1	1	0	0	1	0	0	0
0xFE4	M7_CPUROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	1	0	0

表 579. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE8	M7_CPUROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	M7_CPUROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_CPUROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_CPUROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	M7_CPUROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_CPUROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

Cortex-M7 PPB の ROM レジスタ

PPB ROM メモリタイプレジスタ（M7_PPBROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
															r

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSMEM**：システムメモリの有無

1：このバス上にシステムメモリがあります。

PPB ROM CoreSight ペリフェラル ID レジスタ 4（M7_PPBROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC 継続コード

PPB ROM CoreSight ペリフェラル ID レジスタ 0（M7_PPBROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0c8

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0xC7：Cortex-M7 PPB の ROM テーブル

PPB ROM CoreSight ペリフェラル ID レジスタ 1（M7_PPBROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B4

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x4：Cortex-M7 PPB の ROM テーブル

PPB ROM CoreSight ペリフェラル ID レジスタ 2（M7_PPBROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョン r0p0

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

PPB ROM CoreSight ペリフェラル ID レジスタ 3（M7_PPBROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

PPB ROM CoreSight コンポーネント ID レジスタ 0（M7_PPBROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

PPB ROM CoreSight コンポーネント ID レジスタ 1（M7_PPBROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0x1：ROM テーブルコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

PPB ROM CoreSight コンポーネント ID レジスタ 2（M7_PPBROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

PPB ROM CoreSight コンポーネント ID レジスタ 3（M7_PPBROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値

表 580. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	M7_PPBROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD0	M7_PPBROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	M7_PPBROM_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	M7_PPBROM_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_PPBROM_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_PPBROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									1	1	0	0	0	1	1	1
0xFE4	M7_PPBROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	1	0	0

表 580. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE8	M7_PPBROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1
0xFEC	M7_PPBROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_PPBROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_PPBROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	M7_PPBROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_PPBROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.6.2 Cortex-M7 データウォッチポイントおよびトレースユニット（DWT）

DWT には 4つのコンパレータがあり、それぞれ以下の役割に使用できます。

- ウォッチポイント
- ETM トリガ
- PC サンプリングトリガ
- データアドレスサンプリングトリガ
- データコンパレータ（コンパレータ 1 のみ）
- クロックサイクルカウンタコンパレータ（コンパレータ 0 のみ）

DWT は以下をカウントするためのカウンタも内蔵しています。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット（LSU）の動作数
- スリープサイクル数
- 命令当たりのサイクル数
- 割り込みオーバーヘッドの回数

DWT コンパレータは、以下の項目のうちの 1 つを、DWT_COMP レジスタに保持されている値と比較します。

- データアドレス
- 命令アドレス
- データ値
- サイクルカウント値（コンパレータ 0 のみ）

アドレス照合の場合、コンパレータはマスクを使用することができるため、あるレンジのアドレスと一致します。

照合が成立すると、コンパレータは以下のうちの 1 つを生成します。

- 1つ以上の DWT データトレースパケットで、以下を 1 つ以上含むもの。
  - データアクセスを伴う命令のアドレス
  - アドレスオフセット（データアクセスアドレスのビット [15:0] ）
  - 一致したデータ値
- PC 値またはアクセスしたデータアドレスのいずれかで発生するウォッチポイントデバッグイベント
- DWT ユニット外での一致を信号で伝える CMPMATCH[N] イベント

ウォッチポイントデバッグイベントが発生すると、DebugMonitor 例外が生成されるか、またはプロセッサが停止してデバッグステートに入ります。

DWT の使用方法の詳細については、ARM® v7-M Architecture Reference Manual を参照してください *[5]*。

Cortex-M7 DWT レジスタ

DWT 制御レジスタ（M7_DWT_CTRL）

アドレスオフセット：0x000

リセット値：0x4000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCOMP[3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	Res.	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA
r				r	r	r	r		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PCSAMPLENA	SYNCTAP[1:0]		CYCTAP	POSTINIT[3:0]				POSTRESET[3:0]				CYCCNTENA
			rw	rw		rw	rw				rw				rw

ビット 31:28 **NUMCOMP[3:0]**：実装されるコンパレータの数（読み出し専用）
0x4：4つのコンパレータ

ビット 27 **NOTRCPKT**：トレースサンプリングおよび例外トレースのサポート（読み出し専用）
0：サポートされます。

ビット 26 **NOEXTTRIG**：外部一致信号、CMPMATCH のサポート（読み出し専用）
0：サポートされます。

ビット 25 **NOCYCCNT**：サイクルカウンタのサポート（読み出し専用）
0：サポートされます。

ビット 24 **NOPRFCNT**：プロファイリングカウンタのサポート（読み出し専用）
0：サポートされます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **CYCEVTENA**：POSTCNT アンダーフローイベントカウンタのパケット生成の有効化
0：無効
1：有効

ビット 21 **FOLDEVTENA**：フォールドされた命令数カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 20 **LSUEVTENA**：LSU カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 19 **SLEEPEVTENA**：スリープカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 18 **EXCEVTENA**：例外オーバーヘッドカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 17 **CPIEVTENA**：CPI カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 16 **EXCTRCENA**：例外トレース生成の有効化
0：無効
1：有効

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **PCSAMPLENA**：POSTCNT カウンタ使用の有効化
POSTCNT カウンタを周期的 PC サンプルパケット生成用のタイマとして使用することを有効にします。

0：無効
1：有効

ビット 11:10 **SYNCTAP[1:0]**：CYCCNT カウンタ上の同期パケットカウンタのタップ位置
ここでの選択が、同期パケットの速度を決定します。

0x0：無効 - 同期パケットはありません。
0x1：CYCCNT[24] でタップします。
0x2：CYCCNT[26] でタップします。
0x3：CYCCNT[28] でタップします。

ビット 9 **CYCTAP**：CYCCNT カウンタ上の POSTCNT のタップ位置
0：CYCCNT[6] でタップします。
1：CYCCNT[10] でタップします。

ビット 8:5 **POSTINIT[3:0]**：POSTCNT カウンタの初期値
このフィールドへの書き込みは POSTCNT カウンタが有効であれば、無視されます（すなわち、POSTINIT を書き込む前に CYCEVTENA または PCSAMPLENA をリセットする必要があります）。

ビット 4:1 **POSTPRESET[3:0]**：POSTCNT カウンタの再ロード値

ビット 0 **CYCCNTENA**：CYCCNT カウンタ有効化
0：無効
1：有効

DWT サイクルカウントレジスタ（M7_DWT_CYCCNT）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CYCCNT[31:15]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CYCCNT[15:0]															
rw															

ビット 31:0 **CYCCNT[31:0]**：プロセッサのクロックサイクルカウンタ

DWT CPI カウントレジスタ（M7_DWT_CPICNT）

アドレスオフセット：0x008

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CPICNT[7:0]**：CPI カウンタ

複数サイクルの命令を実行するために必要な追加サイクル数（DWT_LSUCNT によって記録されるものを除く）をカウントし、命令フェッチストールがあればカウントします。

DWT 例外カウントレジスタ（M7_DWT_EXCCNT）

アドレスオフセット：0x00C

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **EXCCNT[7:0]**：例外オーバーヘッドサイクルカウンタ
例外処理に費やされるサイクル数をカウントします。

DWT スリープカウントレジスタ（M7_DWT_SLPCNT）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEPCNT[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SLEEPCNT[7:0]**：スリープサイクルカウンタ
SLEEPモードに費やされるサイクル数をカウントします（WFI、WFE、Sleep-on-exit）。

DWT LSU カウントレジスタ（M7_DWT_LSUCNT）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **LSUCNT[7:0]**：ロード・ストアカウンタ
ロードおよびストア命令を実行するために必要な追加サイクル数をカウントします。

DWT フォールドカウントレジスタ（M7_DWT_FOLDCNT）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **FOLDCNT[7:0]**：フォールドされた命令数のカウンタ
サイクル数 0 の命令のたびにインクリメントします。

DWT プログラムカウンタサンプルレジスタ（M7_DWT_PCSR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIASAMPLE[31:15]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIASAMPLE[15:0]															
rw															

ビット 31:0 **EIASAMPLE[31:0]**：実行された命令アドレスのサンプル値
プログラムカウンタの現在値をサンプリングします。

DWT コンパレータレジスタ x （M7_DWT_COMPx）

アドレスオフセット：0x020 + x * 0x10、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
COMP[31:15]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[15:0]															
rw															

ビット 31:0 **COMP[31:0]**：比較のための基準値

DWT マスクレジスタ x （M7_DWT_MASKx）

アドレスオフセット：0x024 + x * 0x10 （x = 0～3）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
											rw				

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **MASK[4:0]**：コンパレータマスクのサイズ

コンパレータ n によるアドレスレンジ照合のためのアクセスアドレスに適用される無視マスクのサイズを提供します。デバッガはこのフィールドに 0b11111 を書き込み、その後そのレジスタを再び読み出して、サポートされる最大マスクサイズを決定することができます。

DWT 機能レジスタ x （M7_DWT_FUNCTx）

アドレスオフセット：0x028 + x * 0x10、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1[3:0]			
							r					rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAVADDR0[3:0]				DATAVSIZE[1:0]		LINK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
rw				rw		rw	rw	rw		rw		rw			

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **MATCHED**：コンパレーター一致（読み出し専用）

レジスタが最後に読み出されたあと、コンパレータの一致が発生したかどうかを示します。

0：一致していません。
1：一致が発生しました。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **DATAVADDR1[3:0]**：2 番目のコンパレータのコンパレータ番号

DATAVMATCH ビットと LNK1ENA ビットがどちらも 1 の場合、このフィールドには、リンクされたアドレスの比較に使用する 2 番目のコンパレータのコンパレータ番号が保持されます。

ビット 15:12 **DATAVADDR0[3:0]**：コンパレータのコンパレータ番号

DATAVMATCH ビットと LNK1ENA ビットがどちらも 1 の場合、このフィールドには、リンクされたアドレスの比較に使用するコンパレータのコンパレータ番号が保持されます。

ビット 11:10 **DATAVSIZE[1:0]**：必要なデータ比較のサイズ
データ値照合のために必要なデータ比較のサイズを指定します。

0x0：バイト
0x1：ハーフワード
0x2：ワード
0x3：予約済み

ビット 9 **LNK1ENA**：リンクされた 2 番目のコンパレータのサポート（読み出し専用）
リンクされた 2 番目のコンパレータの使用がサポートされるかを示します（読み出し専用）。

1：サポートされます。

ビット 8 **DATAVMATCH**：サイクル比較の有効化
0：アドレス比較を実行します。
1：データ値比較を実行します。

ビット 7 **CYCMATCH**：コンパレータ 0 でのサイクルカウント比較の有効化
このフィールドは 他のコンパレータのために予約済みです。

0：サイクルカウント比較は行われません。
1：DWT_COMP0 をサイクルカウンタ DWT_CYCCNT と比較します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **EMITRANGE**：データトレースのアドレスオフセットパケットの有効化
データトレースのアドレスオフセットパケット（データアドレスのビット 0～15 を含む）の生成を有効にします。
0：無効
1：有効

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **FUNCTION[3:0]**：コンパレーター致時のアクション
このビットフィールドの意味は、DATAVMATCH および CYCMATCH フィールドの設定に依存します。*[5]*を参照してください。

DWT CoreSight ペリフェラル ID レジスタ 4（M7_DWT_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

DWT CoreSight ペリフェラル ID レジスタ 0（M7_DWT_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x02：DWT 部品番号

DWT CoreSight ペリフェラル ID レジスタ 1（M7_DWT_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]

0x0：DWT 部品番号

DWT CoreSight ペリフェラル ID レジスタ 2（M7_DWT_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 003B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x3：r0p4

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

DWT CoreSight ペリフェラル ID レジスタ 3（M7_DWT_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

DWT CoreSight コンポーネント ID レジスタ 0（M7_DWT_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

DWT CoreSight コンポーネント ID レジスタ 1（M7_DWT_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0xE：トレースジェネレータのコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

DWT CoreSight コンポーネント ID レジスタ 2（M7_DWT_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

DWT CoreSight コンポーネント ID レジスタ 3（M7_DWT_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

Cortex-M7 DWT レジスタのマップとリセット値

Cortex-M7 DWT レジスタは AHBD のアドレスレンジ 0xE0001000～0xE0001FFC に配置されています。

表 581. Cortex-M7 DWT レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	M7_DWT_CTRL	NUMCOMP [3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	Res.	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA	Res.	Res.	Res.	PCSAMPLENA	SYNCTAP [1:0]		CYCTAP	POSINIT [3:0]				POSTPRESET [3:0]				CYCCNTENA
	リセット値	0	1	0	0	0	0	0	0		0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	M7_DWT_CYCCNT	CYCCNT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	M7_DWT_CPICNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPICNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x00C	M7_DWT_EXCCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXCCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x010	M7_DWT_SLPCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEPCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x014	M7_DWT_LSUCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x018	M7_DWT_FOLDCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 581. Cortex-M7 DWT レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	M7_DWT_PCSR	EIASAMPLE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	M7_DWT_COMP0	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	M7_DWT_MASK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x028	M7_DWT_FUNCT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE [1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION [3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x030	M7_DWT_COMP1	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	M7_DWT_MASK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x038	M7_DWT_FUNCT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE [1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION [3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x040	M7_DWT_COMP2	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	M7_DWT_MASK2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x048	M7_DWT_FUNCT2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE [1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION [3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x050	M7_DWT_COMP3	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	M7_DWT_MASK3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x058	M7_DWT_FUNCT3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE [1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION [3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0xFD0	M7_DWT_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]			JEP106CON [3:0]				
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	M7_DWT_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

表 581. Cortex-M7 DWT レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD8	M7_DWT_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_DWT_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_DWT_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	0	0	0	0	0	1	0
0xFE4	M7_DWT_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	0	0	0
0xFE8	M7_DWT_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	1	1	0	1	1
0xFEC	M7_DWT_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_DWT_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_DWT_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									0	1	1	1	0	0	0	0
0xFF8	M7_DWT_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_DWT_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.6.3 Cortex-M7 計装トレースマクロセル（ITM）

ITM はトレース情報をパケットとして生成します。パケットは 4 つのソースから生成することができます。パケットが複数のソースから同時に生成される場合、ITM はパケットの出力順番についてアービトレーションを行います。4 つのソースを優先度の高いものから順に挙げます。

1. ソフトウェアトレース

   ソフトウェアは、32 x 32 ビットの ITM スティムラスレジスタのいずれかに直接書き込みを行って、パケットを生成します。各ポートの許可レベルはプログラムで設定できます。ソフトウェアが有効なスティムラスポートに書き込むと、ITM は FIFO に書き込むパケットの中に、ポートの ID、書き込みアクセスのサイズ、および書き込まれたデータを統合します。ITM は FIFO からトレースバスにパケットを出力します。スティムラスポートレジスタを読み出すと、ビット 0 のスティムラスレジスタのステータス（エンプティまたはペンディング）を返します。

2. ハードウェアトレース

   DWT はデータトレースイベント、すなわち PC サンプルやパフォーマンスプロファイリングカウンタの折り返しに応じて、トレースパケットを生成します。ITM がこれらのパケットをトレースバスに出力します。

3. ローカルタイムスタンプ

   ITM は（分周済みの）プロセッサクロックが供給される 21 ビットカウンタを内蔵しています。カウンタの値はトレースバスのタイムスタンプパケットに出力されます。タイムスタンプパケットが生成されるたびに、カウンタはゼロにリセットされます。よって、タイムスタンプは前のタイムスタンプパケットからの経過時間となります。

4. グローバルシステムタイムスタンプ

   タイムスタンプは、タイムスタンプジェネレータのコンポーネントから受信したシステム全体の 64 ビットのカウント値を使用して生成することもできます。

Cortex-M7 ITM レジスタ

ITM スティムラスレジスタ x （M7_ITM_STIMx）

アドレスオフセット：0x000 + x * 0x4 （x = 0〜31）

リセット値：未定義

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STIMULUS[31:15]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STIMULUS[15:0]															
rw															

ビット 31:0 **STIMULUS[31:0]**：ソフトウェアイベントパケット／FIFOREADY

書き込みデータは、ソフトウェアイベントパケットとしてトレースバスに出力されます。読み出し時、ビット 0 は FIFOREADY インジケータとなります。

0：スティムラスポートのバッファはフルです（またはポートが無効）。
1：スティムラスポートは新しい書き込みデータを受け付けることができます。

ITM トレース有効レジスタ（M7_ITM_TER）

アドレスオフセット：0x080

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STIMENA[31:15]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STIMENA[15:0]															
rw															

ビット 31:0 **STIMENA[31:0]**：スティムラスポートの有効化

各ビット n （0:31）は、M7_ITM_STIMn レジスタと関連のあるスティムラスポートを有効にします。

0：ポートを無効化
1：ポートを有効化

ITM トレース特権レジスタ（M7_ITM_TPR）

アドレスオフセット：0xE00

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRIVMASK[31:15]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIVMASK[15:0]															
rw															

ビット 31:0 **PRIVMASK[31:0]**：ITM スティムラスポートへの非特権のアクセスを有効にします。
各ビットは 8つのスティムラスポートを制御します。

0bXXX0：ポート 0～7 では非特権のアクセスが許可されます。
0bXXX1：ポート 0～7 では特権アクセスのみが許可されます。
0bXX0X：ポート 8～15 では非特権のアクセスが許可されます。
0bXX1X：ポート 8～15 では特権アクセスのみが許可されます。
0bX0XX：ポート 16～23 では非特権のアクセスが許可されます。
0bX1XX：ポート 16～23 では特権アクセスのみが許可されます。
0b0XXX：ポート 24～31 では非特権のアクセスが許可されます。
0b1XXX：ポート 24～31 では特権アクセスのみが許可されます。

ITM トレース制御レジスタ（M7_ITM_TCR）

アドレスオフセット：0xE80

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]						
								rw	rw						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	GTSFREQ[1:0]		TSPRESCALE[1:0]		Res.	Res.	Res.	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
				rw		rw					r	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **BUSY**：ITM ビジー
ITM が現在イベント処理中かどうかを示します（読み出し専用）。

0：非ビジー
1：ビジー

ビット 22:16 **TRACEBUSID[6:0]**：マルチソーストレースストリームのフォーマット識別子
マルチソーストレースを使用している場合、デバッガはこのフィールドにゼロ以外の値を書き込まなければなりません。注：システム内のトレースソースにはそれぞれ異なる ID を使用してください。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **GTSFREQ[1:0]**：グローバルタイムスタンプの頻度

グローバルタイムスタンプのクロック周波数に基づいて ITM がグローバルタイムスタンプを生成する頻度を定義したり、グローバルタイムスタンプの生成を無効にしたりします。設定可能な値は以下のとおり。

0x0：グローバルタイムスタンプの生成を無効にします。
0x1：ITM がグローバルタイムスタンプカウンタのビット [63:7] の変化を検出するたびに、タイムスタンプリクエストを生成します。頻度は 128 サイクルに 1 回程度です。
0x2：ITM がグローバルタイムスタンプカウンタのビット [63:13] の変化を検出するたびに、タイムスタンプリクエストを生成します。頻度は 8192 サイクルに 1 回程度です。
0x3：出力 FIFO が空の場合、毎パケット後にタイムスタンプを生成します。

ビット 9:8 TSPRESCALE [1:0]：ローカルタイムスタンプのプリスケール
トレースパケットのリファレンスクロックに用いるプリスケール。設定可能な値は以下のとおり。

0x0：プリスケールなし。
0x1：分周比 4
0x2：分周比 16
0x3：分周比 64

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SWOENA:** タイムスタンプカウンタの非同期クロックの有効化（読み出し専用）
0：タイムスタンプカウンタはプロセッサクロックを使用します。

ビット 3 **TXENA：**ハードウェアイベントパケット転送の有効化
DWT ユニットからトレースポートへのハードウェアイベントパケットの転送を有効にします。

0：無効
1：有効

ビット 2 **SYNCENA：**同期パケット送信の有効化
デバッガがこのビットをセットした場合、正しい同期速度にするために、DWT の DWT_CTRL レジスタの SYNCTAP フィールドも設定する必要があります。

0：無効
1：有効

ビット 1 **TSENA：**ローカルタイムスタンプ生成の有効化
0：無効
1：有効

ビット 0 **ITMENA：**ITM 有効化
0：無効
1：有効

ITM CoreSight ペリフェラル ID レジスタ 4（M7_ITM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

ITM CoreSight ペリフェラル ID レジスタ 0（M7_ITM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x01：ITM 部品番号

ITM CoreSight ペリフェラル ID レジスタ 1（M7_ITM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x1：ITM 部品番号

ITM CoreSight ペリフェラル ID レジスタ 2（M7_ITM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 003B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x3：r0p4

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

ITM CoreSight ペリフェラル ID レジスタ 3（M7_ITM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

ITM CoreSight コンポーネント ID レジスタ 0（M7_ITM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

ITM CoreSight コンポーネント ID レジスタ 1（M7_ITM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス

0xE：トレースジェネレータのコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]

0x0：コモン ID 値

ITM CoreSight コンポーネント ID レジスタ 2（M7_ITM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

ITM CoreSight コンポーネント ID レジスタ 3（M7_ITM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

Cortex-M7 ITM レジスタのマップとリセット値

ITM レジスタは AHBD のアドレスレンジ 0xE0000000～0xE0000FFC に配置されています。

表 582. Cortex-M7 ITM レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000 から 0x07C	M7_ITM_STIM0-31	STIMULUS[31:0]																															
	リセット値																																
0x0E00	M7_ITM_TER	STIMENA[31:0]																															
	リセット値																																
0x0E40	M7_ITM_TPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIVMASK[3:0]			
	リセット値																													0	0	0	0
0xE80	M7_ITM_TCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]							Res.	Res.	Res.	Res.	GTSFREQ[1:0]		TSPRESCALE[1:0]		Res.	Res.	Res.	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
	リセット値									0	0	0	0	0	0	0	0					0	0	0	0				0	0	0	0	0
0xFD0	M7_ITM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	M7_ITM_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

表 582. Cortex-M7 ITM レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD8	M7_ITM_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_ITM_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_ITM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0xFE4	M7_ITM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	0	0	0
0xFE8	M7_ITM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	1	1	0	1	1
0xFEC	M7_ITM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_ITM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_ITM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									0	1	1	1	0	0	0	0
0xFF8	M7_ITM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_ITM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.6.4 Cortex-M7 ブレークポイントユニット（FPB）

FPB を使用することでハードウェアのブレークポイントを設定することができます。このユニットには、一致が検出されたときに、命令フェッチアドレスを監視し、ブレークポイント命令を返すコンパレータが 8つ内蔵されています。Cortex-M7 FPB はフラッシュパッチ機能をサポートしません。

Cortex-M7 FPB レジスタ

FPB 制御レジスタ（M7_FPB_CTRL）

アドレスオフセット：0x000

リセット値：0x0000 0080

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NUM_CODE[6:4]			NUM_LIT[3:0]				NUM_CODE[3:0]				Res.	Res.	KEY	ENABLE
	r			r				r						rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **NUM_CODE[6:4]**：命令アドレスコンパレータの数のフィールド、MSB x 3
この読み出し専用フィールドには、サポートされる命令アドレスコンパレータの数の 3 MSB が保持されます。

0x0：番号の MSB はすべて 0 です。

ビット 11:8 **NUM_LIT[3:0]**：サポートされるリテラルアドレスコンパレータの数（読み出し専用）
0x0：リテラルコンパレータはサポートされません。

ビット 7:4 **NUM_CODE[3:0]**：命令アドレスコンパレータの数のフィールド、LSB x 4
この読み出し専用フィールドには、サポートされる命令アドレスコンパレータの数の 4 LSB が保持されます。

0x8：8つの命令コンパレータがサポートされます。

ビット 1 **KEY**：書き込み保護キー
M7_FPB_CTRL レジスタへの書き込みは、このビットに「1」がセットされていない場合は無視されます。

ビット 0 **ENABLE**：FPB 有効化
0：無効化
1：有効化

参考資料

FPB リマップレジスタ（M7_FPB_REMAP）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	RMPSPT	REMAP[23:11]												
		r	rw												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REMAP[10:0]											Res.	Res.	Res.	Res.	Res.
rw															

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 29 **RMPSPT**：フラッシュパッチリマップのサポート（読み出し専用）

0：リマップはサポートされません。

ビット 28:5 **REMAP[23:0]**：予約済み - サポートされません。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

FPB コンパレータレジスタ（M7_FPB_COMPx）

アドレスオフセット：0x008 + x * 0x4 （x = 0～7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REPLACE[1:0]		Res.	COMP[26:14]												
rw			rw												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[13:0]														Res.	ENABLE
rw															rw

ビット 31:30 **REPLACE[1:0]**：COMP と命令フェッチアドレスが一致したときの動作

COMP フィールドと命令フェッチアドレスの一致が発生したときの動作を定義します。

0x0：予約済み

0x1：下位ハーフワードにブレークポイントが定義され、上位ハーフワードは定義されません。

0x2：上位ハーフワードにブレークポイントが定義され、下位ハーフワードは定義されません。

0x3：上位および下位のハーフワードにそれぞれブレークポイントが定義されます。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:2 **COMP[26:0]**：コードメモリアクセスアドレスとの比較値
命令コードメモリへのアクセスアドレスのビット 28:2 と比較するための値です（0x00000000～0x1FFFFFFF）。一致の発生時に取るべき動作は、REPLACE フィールドに定義されます。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **ENABLE**：コンパレータの有効化
コンパレータは、このビットと M7_FPB_CTRL レジスタの FPB ENABLE ビットの両方がセットされている場合にのみ有効です。

0：無効
1：有効

FPB CoreSight ペリフェラル ID レジスタ 4（M7_FPB_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ
0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード
0x4：ARM® JEDEC コード

FPB CoreSight ペリフェラル ID レジスタ 0（M7_FPB_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 000C

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]
0x0C：FPB 部品番号

FPB CoreSight ペリフェラル ID レジスタ 1（M7_FPB_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]
0x0：FPB 部品番号

FPB CoreSight ペリフェラル ID レジスタ 2（M7_FPB_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 002B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x2：r0p3

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

FPB CoreSight ペリフェラル ID レジスタ 3（M7_FPB_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]：**メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

FPB CoreSight コンポーネント ID レジスタ 0（M7_FPB_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]

0x0D：コモン ID 値

FPB CoreSight コンポーネント ID レジスタ 1（M7_FPB_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0xE：トレースジェネレータのコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

FPB CoreSight コンポーネント ID レジスタ 2（M7_FPB_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

FPB CoreSight コンポーネント ID レジスタ 3（M7_FPB_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]
0xB1：コモン ID 値

Cortex-M7 FPB レジスタのマップとリセット値

Cortex-M7 FPB レジスタはアドレスレンジ 0xE0002000～0xE0002FFC に配置されています。

表 583. Cortex-M7 FPB レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	M7_FPB_CTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUM_CODE [6:4]			NUM_LIT [3:0]				NUM_CODE [3:0]				Res.	Res.	KEY	ENABLE
	リセット値			0															0	0	0	0	0	0	0	1	0	0	0			0	0
0x004	M7_FPB_REMAP	Res.	Res.	RMPSPT	REMAP[23:0]																									Res.	Res.	Res.	Res.	Res.
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x008 から 0x024	M7_FPB_COMP0-7	REPLACE [1:0]		Res.	COMP[26:0]																											Res.	ENABLE
	リセット値																																0
0xFD0	M7_FPB_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	M7_FPB_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	M7_FPB_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_FPB_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_FPB_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	0	0	0	1	1	0	0
0xFE4	M7_FPB_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	0	0	0	0
0xFE8	M7_FPB_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	0	1	0	1	1
0xFEC	M7_FPB_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_FPB_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_FPB_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	1	1	0	0	0	0	0
0xFF8	M7_FPB_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_FPB_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.6.5 Cortex-M7 組み込みトレースマクロセル（ETM）

Cortex-M7 ETM は CPU と密接に連結された CoreSight™ コンポーネントです。ETM は Cortex-M7 コアの実行をトレースすることができるトレースパケットを生成します。STM32H7 では、ETM は命令トレース専用に設定されているため、トレース情報にデータアクセスは含まれません。

ETM はプロセッサのトレースインタフェース経由で CPU から以下の情報を受信します。

- 同一サイクルで実行される命令の数
- プログラムフローの変更
- プロセッサの現在の命令ステート
- ロードおよびストア命令によってアクセスされるメモリ位置のアドレス
- 転送のタイプ、方向、およびサイズ
- 条件コードに関する情報
- 例外情報
- 割り込み待ちステートに関する情報

詳細については、ARM® CoreSight™ ETM™-M7 Technical Reference Manual を参照してください *[6]*。

Cortex-M7 ETM レジスタ

ETM プログラミング制御レジスタ（M7_ETM_PRGCTL）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN**：トレースプログラムの有効化

0：トレースユニットは無効です。

1：トレースユニットは有効です。

ETM プロセッサ選択制御レジスタ（M7_ETM_PROCSEL）

アドレスオフセット：0x008

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **PROCSEL**：プロセッサの選択

Cortex-M7 のみ、この ETM を使用するので、このフィールドの影響はありません。

ETM ステータスレジスタ（M7_ETM_STAT）

アドレスオフセット：0x00C

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **PMSTABLE**：プログラマモデルの安定性

ETM レジスタが安定しており、読み出し可能かを示します。

0：レジスタは安定していません。

1：レジスタは安定しています。

ビット 0 **IDLE**：トレースユニットインアクティブ

0：ETM はアイドル状態ではありません。

1：ETM はアイドル状態です。

ETM トレース設定レジスタ（M7_ETM_CONFIG）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RS	TS	COND[2:0]			Res.	Res.	Res.	CCI	BB	INSTP0[1:0]		Res.
			rw	rw	rw						rw	rw	r		

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DV**：データ値のトレース（読み出し専用）
0：無効

ビット 16 **DA**：データアドレスのトレース（読み出し専用）
0：無効

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **RS**：リターンスタックの有効化
0：無効
1：有効

ビット 11 **TS**：グローバルタイムスタンプのトレース
0：無効
1：有効

ビット 10:8 **COND[2:0]**：条件付き命令のトレース
0x0：条件付き命令のトレースは無効です。
0x1：条件付きロード命令をトレースします。
0x2：条件付きストア命令をトレースします。
0x3：条件付きロードおよびストア命令をトレースします。
0x7：すべての条件付き命令をトレースします。
その他：予約済み

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CCI**：命令トレース時のサイクルのカウント
0：無効
1：有効

ビット 3 **BB**：分岐ブロードキャストモード
0：無効
1：有効

ビット 2:1 **INSTP0[1:0]**：P0 命令の決定（読み出し専用）
0x0：分岐のみが P0 命令です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

ETM イベント制御 0 レジスタ（M7_ETM_EVENTCTL0）

トレースユニットが無効のときにのみ書き込みを受け付けます。

アドレスオフセット：0x020

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPE1	Res.	Res.	Res.	SEL1[3:0]				TYPE0	Res.	Res.	Res.	SEL0[3:0]			
rw				rw				rw				rw			

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TYPE1**：イベント 1 のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **SEL1[3:0]**：イベント 1 のリソース／ブール組み合わせリソースペア
TYPE1 が 0 のとき、ビット [3:0] によって定義される 0~15の中から単一選択リソースを選択します。
TYPE1 が 1 のとき、ビット [2:0] によって定義される 0~7の中からブール組み合わせリソースペアを選択します。

ビット 7 **TYPE0**：イベント 0 のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL0[3:0]**：イベント 0 のリソース／ブール組み合わせリソースペア
TYPE0 が 0 のとき、ビット [3:0] によって定義される 0~15の中から単一選択リソースを選択します。
TYPE0 が 1 のとき、ビット [2:0] によって定義される 0~7の中からブール組み合わせリソースペアを選択します。

ETM イベント制御 1 レジスタ（M7_ETM_EVENTCTL1）

トレースユニットが無効のときにのみ書き込みを受け付けます。

アドレスオフセット：0x024

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[3:0]			
			rw	rw								rw			

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LPOVERRIDE**：低電力ステートでの動作のオーバーライド

0：低電力ステートで通常の動作を行います。

1：低電力ステートに入っても、リソースやイベントトレース生成には影響ありません。

ビット 11 **ATB**：ATB トリガ有効

0：無効

1：有効

ビット 10:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **INSTEN[3:0]**：命令トレースイベント要素有効

1 ビットは 1 イベントに相当します。

0bXXX0：イベント 0 はイベント要素の原因とはなりません。
0bXXX1：イベント 0 はイベント要素の原因となります。
0bXX0X：イベント 1 はイベント要素の原因とはなりません。
0bXX1X：イベント 1 はイベント要素の原因となります。
0bX0XX：イベント 2 はイベント要素の原因とはなりません。
0bX1XX：イベント 2 はイベント要素の原因となります。
0b0XXX：イベント 3 はイベント要素の原因とはなりません。
0b1XXX：イベント 3 はイベント要素の原因となります。

ETM ストール制御レジスタ（M7_ETM_STALLCTL）

トレースユニットが無効のときにのみ書き込みを受け付けます。

アドレスオフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	DSTALL	ISTALL	Res.	Res.	Res.	Res.	LEVEL[1:0]		Res.	Res.
						rw	rw					rw			

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **DSTALL**：データトレースのバッファスペースに基づくプロセッサのストール

0：プロセッサをストールしません。

1：プロセッサをストールします。

ビット 8 **ISTALL**：命令トレースのバッファスペースに基づくプロセッサのストール

0：プロセッサをストールしません。

1：プロセッサをストールします。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **LEVEL[1:0]**：ストール閾値レベル

低レベルの場合、プロセッサのストール量は最小となり、FIFO オーバーフローのリスクが高まります。
高レベルの場合、FIFO オーバーフローのリスクは最小となりますが、プロセッサのストール量が増えます。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

ETM グローバルタイムスタンプ制御レジスタ（M7_ETM_TSCTL）

トレースユニットが無効のときにのみ書き込みを受け付けます。

アドレスオフセット：0x030

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TYPE	Res.	Res.	Res.	SEL[1:0]			
								rw				rw			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TYPE**：タイムスタンプ挿入のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL[3:0]**：リソース／ブール組み合わせリソースペア
TYPE が 0 のとき、ビット [3:0] によって定義される 0~15の中から単一選択リソースを選択します。
TYPE が 1 のとき、ビット [2:0] によって定義される 0~7の中からブール組み合わせリソースペアを選択します。

ETM 同期周期レジスタ（M7_ETM_SYNCP）

アドレスオフセット：0x034

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
											r				

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **PERIOD[4:0]**：同期リクエスト間のトレースのバイト数
トレース同期リクエスト間のトレース情報のバイト数を定義します。
0xA：1024 バイト

ETM サイクルカウント制御レジスタ（M7_ETM_CCCTL）

アドレスオフセット：0x038

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PERIOD[11:0]											
				rw											

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:0　**THRESHOLD[11:0]**：命令トレースサイクルカウントの閾値

この閾値は、サイクルカウントトレースパケットの最小間隔を表します。

0x0：予約済み

その他：閾値

ETM トレース ID レジスタ（M7_ETM_TRACEID）

アドレスオフセット：0x040

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.						TRACEID[6:0]						
									rw						

ビット 31:7　予約済みであり、リセット値に保持する必要があります。

ビット 6:0　**TRACEID[6:0]**：トレース ID

0x00：予約済み

0x01 から 0x6F：有効 ID

0x70 から 0x7F：予約済み

ETM ViewInst メイン制御レジスタ（M7_ETM_VICTL）

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S3	Res.	Res.	EXLEVEL_S0
												rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSSTATUS	Res.	TYPE				SEL[3:0]			
				rw	rw	rw		rw				rw			

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **EXLEVEL_S3**：例外レベル 3 のトレースの無効化

例外レベル 3 のセキュアステートにおいて、指定された例外レベルでトレースを無効にします。

0：この例外レベルで ViewInst を有効にします。

1：この例外レベルで ViewInst を無効にします。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **EXLEVEL_S0**：例外レベル 0 のトレースの無効化

例外レベル 0 のセキュアステートにおいて、指定された例外レベルでトレースを無効にします。

0：この例外レベルで ViewInst を有効にします。

1：この例外レベルで ViewInst を無効にします。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TRCERR**：システムエラー例外のトレース

システムエラー例外を常にトレースする必要があるかどうかを選択します。

0：システムエラー例外の直前の命令または例外がトレースされる場合のみ、システムエラー例外はトレースされます。

1：ViewInst の値に関係なく、システムエラー例外は常にトレースされます。

ビット 10 **TRCRESET**：リセット例外のトレース

リセット例外を常にトレースする必要があるかどうかを選択します。

0：リセット例外の直前の命令または例外がトレースされる場合のみ、リセット例外はトレースされます。

1：ViewInst の値に関係なく、リセット例外は常にトレースされます。

ビット 9 **SSSTATUS**：開始／停止ロジックの現在のステータス

0：停止ステート

1：開始ステート

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TYPE**：リソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 6:14 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL[3:0]**：リソース／ブール組み合わせリソースペア
TYPE が 0 のとき、ビット [3:0] によって定義される 0~15の中から単一選択リソースを選択します。
TYPE が 1 のとき、ビット [2:0] によって定義される 0~7の中からブール組み合わせリソースペアを選択します。

ETM ViewInst 開始／停止制御レジスタ（M7_ETM_VISSCTL）

アドレスオフセット：0x088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[7:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[7:0]							
									rw						

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **STOP[7:0]**：トレースを停止させる単一アドレスコンパレータのセレクタ
ViewInst 開始／停止制御でトレースを停止させる単一アドレスコンパレータを定義します。実装されている各単一アドレスコンパレータに対して 1 ビットが提供されます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **START[7:0]**：トレースを開始させる単一アドレスコンパレータのセレクタ
ViewInst 開始／停止制御でトレースを開始させる単一アドレスコンパレータを定義します。実装されている各単一アドレスコンパレータに対して 1 ビットが提供されます。

ETM ViewInst 開始／停止プロセッサコンパレータの制御レジスタ（M7_ETM_VIPCSSCTL）

アドレスオフセット：0x08C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[3:0]			
												rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[3:0]			
												rw			

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **STOP[3:0]**：トレースを停止させるプロセッサコンパレータ入力のセレクタ
トレースを停止させる目的で、ViewInst 開始／停止制御とともに使用するプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 ビットが提供されます。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **START[3:0]**：トレースを開始させるプロセッサコンパレータ入力のセレクタ
トレースを開始させる目的で、ViewInst 開始／停止制御とともに使用するプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 ビットが提供されます。

ETM カウンタ再ロード値レジスタ（M7_ETM_CNTRLDV）

アドレスオフセット：0x140

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VALUE[15:0]															
rw															

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **VALUE[15:0]**：カウンタ再ロード値
この値は、再ロードイベント発生のたびにカウンタにロードされます。

ETM ID レジスタ 8（M7_ETM_IDR8）

アドレスオフセット：0x180

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MAXSPEC[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAXSPEC[15:0]															
r															

ビット 31:0 **MAXSPEC[31:0]**：最大スペキュレーション深さ
命令トレースストリームの最大スペキュレーション深さを示します。これは、トレースストリームで同時にコミットされることがない P0 要素の最大数です。
0x2：トレーススペキュレーションの最大深さは 2 です。

ETM ID レジスタ 9（M7_ETM_IDR9）

アドレスオフセット：0x184

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP0KEY[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP0KEY[15:0]															
r															

ビット 31:0 **NUMP0KEY[31:0]**：使用する P0 右側キーの数

0x0：命令トレースのみの設定では P0 キーは使用しません。

ETM ID レジスタ 10（M7_ETM_IDR10）

アドレスオフセット：0x188

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1KEY[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1KEY[15:0]															
r															

ビット 31:0 **NUMP1KEY[31:0]**：P1 右側キーの総数

通常キーと特殊キーを含む P1 右側キーの総数を示します。

0x0：命令トレースのみの設定では P1 キーは使用しません。

ETM ID レジスタ 11（M7_ETM_IDR11）

アドレスオフセット：0x18C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1SPC[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1SPC[15:0]															
r															

ビット 31:0 **NUMP1SPC[31:0]**：使用する特殊な P1 右側キーの総数

0x0：特殊な P1 キーは使用しません。

ETM ID レジスタ 12（M7_ETM_IDR12）

アドレスオフセット：0x190

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDKEY[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDKEY[15:0]															
r															

ビット 31:0 **NUMCONDKEY[31:0]**：
通常キーと特殊キーを含む条件付き命令右側キーの総数を示します。
0x1：条件付き命令右側キーが 1つ実装されます。

ETM ID レジスタ 13（M7_ETM_IDR13）

アドレスオフセット：0x194

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDSPC[31:16]															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDSPC[15:0]															
r															

ビット 31:0 **NUMCONDSPC[31:0]**：特殊な条件付き命令右側キーの数
0x0：条件付き命令右側キーは実装されません。

ETM 実装固有レジスタ 0（M7_ETM_IMSPEC0）

アドレスオフセット：0x1C0

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SUPPORT[3:0]**：実装固有の拡張機能のサポート
0x0：実装固有の拡張機能はサポートされません。

ETM ID レジスタ 0（M7_ETM_IDR0）

アドレスオフセット：0x1E0

リセット値：0x0C00 1EE1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	COMMOPT	TSSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSUPP[1]
		r	r												r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
QSUPP[0]	Res.	CONDTYPE[1:0]		NUMEVENT[1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	Res.	Res.	Res.	Res.	Res.
r		r		r		r		r	r	r					

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **COMMOPT**：一部のパケットにおけるコミットフィールドの意味
0：コミットモード 0

ビット 28:24 **TSSIZE[4:0]**：グローバルタイムスタンプのサイズ
0x08：最大 64 ビットのグローバルタイムスタンプが実装されます。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16:15 **QSUPP[1:0]**：Q 要素のサポート
0x0：Q 要素はサポートされません。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **CONDTYPE[1:0]**：条件付き結果のトレース方法
0x1：APSR 条件フラグの値をトレースします。

ビット 11:10 **NUMEVENT[1:0]**：トレースでサポートされるイベント数
0x1：命令のみの設定では 2つのイベントがサポートされます。

ビット 9 **RETSTACK**：リターンスタックのサポート
1：2つのエントリリターンスタックがサポートされます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TRCCCI**：命令トレース時のサイクルカウントのサポート
1：命令トレース時のサイクルカウントが実装されます。

ビット 6 **TRCCOND**：条件付き命令のトレースのサポート
1：条件付き命令のトレースが実装されます。

ビット 5 **TRCBB**：分岐ブロードキャストのトレースのサポート
1：分岐ブロードキャスト のトレースが実装されます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

ETM ID レジスタ 1（M7_ETM_IDR1）

アドレスオフセット：0x1E4

リセット値：0x4100 F401

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCARCHMAJ[3:0]				TRCARCHMIN[3:0]				REVISION[3:0]			
				r				r				r			

ビット 31:24 **DESIGNER[7:0]**：トレースユニット設計者エンティティ
0x41：ARM®

ビット 23:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TRCARCHMAJ[3:0]**：メジャーなトレースユニットアーキテクチャのバージョン番号
0x4：ETM v4

ビット 7:4 **TRCARCHMIN[3:0]**：マイナーなトレースユニットアーキテクチャのバージョン番号
0x0：マイナーバージョン 0

ビット 3:0 **REVISION[3:0]**：実装のリビジョン番号
0x1：リビジョン 1

ETM ID レジスタ 2（M7_ETM_IDR2）

アドレスオフセット：0x1E8

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:1]			
			r				r					r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DASIZE[0]	VMIDSIZE[4:0]					CIDSIZE[4:0]					IASIZE[4:0]				
r	r					r					r				

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **CCSIZE[3:0]**：サイクルカウンタのサイズ
サイクルカウンタのサイズ（ビット単位）から 12 を引いた値を示します。
0x0：サイクルカウンタは 12 ビットです。

ビット 24:20 **DVSIZE[4:0]**：データ値のサイズ（バイト単位）
0x0：命令のみの設定では、データ値のサイズはサポートされません。

ビット 19:15 **DASIZE[4:0]**：データアドレスのサイズ（バイト単位）
0x0：命令のみの設定では、データアドレスのサイズはサポートされません。

ビット 14:10 **VMIDSIZE[4:0]**：仮想マシン ID サイズ
0x0：仮想マシン ID のトレースは実装されません。

ビット 9:5 **CIDSIZE[4:0]**：コンテキスト ID サイズ
0x0：コンテキスト ID のトレースは実装されません。

ビット 4:0 **IASIZE[4:0]**：命令アドレスのサイズ
0x4：最大 32 ビットアドレスサイズ

ETM ID レジスタ 3（M7_ETM_IDR3）

アドレスオフセット：0x1EC

リセット値：0x0509 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NOOVERFLOW	NUMPROC[2:0]			SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]			
r	r			r	r	r	r					r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CCITMIN[11:0]											
				r											

ビット 31 **NOOVERFLOW**：NOOVERFLOW のサポート
トレーススストール制御の NOOVERFLOW が実装されるかどうかを示します。
0：実装されません。

ビット 30:28 **NUMPROC[2:0]**：トレースに利用できるプロセッサの数
0x0：1つのプロセッサのみをトレースすることができます。

ビット 27 **SYSSTALL**：プロセッサのストール制御のシステムサポート
0：サポートされません

ビット 26 **STALLCTL**：ストール制御のサポート
1：トレーススストール制御（TRCSTALLCTLR）が実装されます。

ビット 25 **SYNCPR**：トレース同期周期のサポート
0：命令トレースのみの設定では TRCSYNCPR は読み出し専用です。したがって、トレース同期周期は固定です。

ビット 24 **TRCERR**：TRCVICTLR.TRCERR のサポート
TRCVICTLR.TRCERR が実装されるかどうかを示します。
0x4：最大 32 ビットアドレスサイズ

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **EXLEVEL_S[3:0]**：特権レベルのサポート
特権レベルが実装されます（各レベルに 1 ビット）。
0x9：特権レベル、スレッドとハンドラーが実装されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **CCITMIN[11:0]**：命令トレースサイクルカウントの最小閾値
0x4：最小閾値は 4 命令トレースサイクルです。

ETM ID レジスタ 4（M7_ETM_IDR4）

アドレスオフセット：0x1F0

リセット値：0x0001 4000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMVMIDC[3:0]				NUMCIDC[3:0]				NUMSSCC[3:0]				NUMRSPAIR[3:0]			
r				r				r				r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMPC[3:0]				Res.	Res.	Res.	SUPPADC	NUMDVC[3:0]				NUMACPAIRS[3:0]			
r							r	r				r			

ビット 31:28 **NUMVMIDC[3:0]**：実装される仮想マシン ID コンパレータの数
0x0：なし

ビット 27:24 **NUMCIDC[3:0]**：実装されるコンテキスト ID コンパレータの数
0x0：なし

ビット 23:20 **NUMSSCC[3:0]**：実装されるシングルショットコンパレータ制御の数
0x0：なし

ビット 19:16 **NUMRSPAIR[3:0]**：実装されるリソース選択ペアの数
0x1：なし

ビット 15:12 **NUMPC[3:0]**：実装されるプロセッサコンパレータ入力の数
0x4：4

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SUPPADC**：データアドレス比較のサポート
0：実装されません。

ビット 7:4 **NUMDVC[3:0]**：実装されるデータ値コンパレータの数
0x0：なし

ビット 3:0 **NUMACPAIRS[3:0]**：実装されるアドレスコンパレータペアの数
0x0：なし

ETM ID レジスタ 5（M7_ETM_IDR5）

アドレスオフセット：0x1F4

リセット値：0x90C7 0402

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE[2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE[5:0]					
r	r			r				r	r	r					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NUMEXTINSEL[2:0]			NUMEXTIN[8:0]								
				r			r								

ビット 31 **REDFUNCNTR**：低減機能カウンタのサポート
1：実装されます。

ビット 30:28 **NUMCNTR[2:0]**：実装されるカウンタの数
0x1：1つのカウンタが実装されます。

ビット 27:25 **NUMSEQSTATE[2:0]**：実装されるシーケンサのステート数
0x0：なし

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **LPOVERRIDE**：低電力ステートのオーバーライドのサポート
1：実装されます。

ビット 22 **ATBTRIG**：ATB トリガのサポート
1：実装されます。

ビット 21:16 **TRACEIDSIZE[5:0]**：トレース ID のビット数
0x07：7 ビットトレース ID が実装されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **NUMEXTINSEL[2:0]**：実装される外部入力セレクタの数
0x2：2 つの外部入力セレクタが実装されます。

ビット 8:0 **NUMEXTIN[8:0]**：実装されている外部入力の数
0x2：2つの外部入力が実装されます。

ETM リソース選択レジスタ 2（M7_ETM_RSCTL2）

アドレスオフセット：0x208

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP[2:0]		
										rw	rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw							

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **PAIRINV**：組み合わせリソースペアの結果の反転
0：反転されません。
1：反転されます。

ビット 20 **INV**：選択リソースの反転
0：反転されません。
1：反転されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **GROUP[2:0]**：リソースグループの選択

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SELECT[7:0]**：希望するグループからの リソースの選択

希望するグループから 1 つ以上のリソースを選択します。グループからの各リソースに対して 1 ビットが提供されます。

ETM リソース選択レジスタ 3（M7_ETM_RSCTL3）

アドレスオフセット：0x20C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INV	Res.	GROUP[2:0]		
											rw		rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw							

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INV**：選択リソースの反転

0：反転されません。

1：反転されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **GROUP[2:0]**：リソースグループの選択

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SELECT[7:0]**：希望するグループからの リソースの選択

希望するグループから 1 つ以上のリソースを選択します。グループからの各リソースに対して 1 ビットが提供されます。

ETM シングルショットコンパレータ制御レジスタ 0（M7_ETM_SSCC0）

アドレスオフセット：0x280

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **RST**：シングルショットコンパレータリソースのリセットの有効化

シングルショットコンパレータリソースの発生時にそのリセットを有効にし、別のコンパレータの一致を検出できるようにします。

0：無効

1：リセットは有効です。複数の一致が発生する場合があります。

ビット 23:0 予約済みであり、リセット値に保持する必要があります。

ETM シングルショットコンパレータのステータスレジスタ 0（M7_ETM_SSCS0）

アドレスオフセット：0x2A0

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	INST
													r	r	r

ビット 31 **STATUS**：シングルショットステータス

選択されたコンパレータのいずれかで一致が発生したかどうかを示します。SSCC0.RST が 0 に設定されている場合、シングルショットコンパレータの制御を有効にするために、STATUS ビットを 0で書き込む必要があります。

0：一致は発生しませんでした。

1：最低でも 1 回の一致が発生しました。

ビット 30:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DV**：データ値コンパレータのサポート

0：シングルショットデータ値の比較はサポートされません。

ビット 1 **DA**：データアドレスコンパレータのサポート

0：シングルショットデータアドレスの比較はサポートされます。

ビット 0 **INST**：命令アドレスコンパレータのサポート

1：シングルショット命令アドレスの比較はサポートされます。

ETM シングルショットプロセッサコンパレータの入力制御レジスタ（M7_ETM_SSPCIC0）

アドレスオフセット：0x2C0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[7:0]							
								rw							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PC[7:0]**：シングルショット制御のコンパレータ入力セレクタ

シングルショット制御のために 1 つ以上のプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 ビットが提供されます。

ETM パワーダウン制御レジスタ（M7_ETM_PDC）

アドレスオフセット：0x2C0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU	Res.	Res.	Res.
												r			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PU**：パワーアップリクエスト

ETM への電源を維持し、トレースレジスタにアクセスするようリクエストします。

0：電源はリクエストされません。
1：電源がリクエストされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

ETM パワーダウンステータスレジスタ（M7_ETM_PDS）

アドレスオフセット：0x2C0

リセット値：0x0000 0003

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
														r	r

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **STICKYPD**：スティッキーなパワーダウンステート

ETM レジスタへの電源が取り外された場合に、このビットは 1 にセットされプログラミングステートが失われたことを知らせます。このビットは、TRCPDSR の読み出し後にクリアされます。

0：トレースレジスタの電源は PDS レジスタを最後に読み出してから遮断されていません。
1：トレースレジスタの電源は PDS レジスタを最後に読み出したあと遮断されました。

ビット 0 **POWER**：ETM のパワーアップ

1：ETM がパワーアップされ、すべてのレジスタにアクセス可能です。

ETM 要求タグセットレジスタ（M7_ETM_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMSET[3:0]**：要求タグビットのセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。

読み出し：
0xF：要求タグには 4 つのビットがあることを示します。

ETM 要求タグクリアレジスタ（M7_ETM_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書き込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読み出し：要求タグの現在値を返します。

ETM ロックアクセスレジスタ（M7_ETM_LAR）

アドレスオフセット：0xFB0

リセット値：N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w															

ビット 31:0 **ACCESS_W[31:0]**：ETM レジスタ書き込みアクセス

プロセッサコアによる一部の ETM レジスタへの書き込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。

0xC5ACCE55：書き込みアクセスを有効にします。
その他の値：書き込みアクセスを無効にします。

ETM ロックステータスレジスタ（M7_ETM_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCK TYPE	LOCK GRANT	LOCK EXIST
													r	r	r

ビット 31:3　予約済みであり、リセット値に保持する必要があります。

ビット 2　**LOCKTYPE**：M7_ETM_LAR レジスタのサイズ

0：32 ビット

ビット 1　**LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読み出し時に毎回 0 を返します。

0：書き込みアクセスが許可されます。

1：書き込みアクセスはブロックされます。読み出しアクセスのみが許可されます。

ビット 0　**LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読み出し時に毎回 0 を返します。

0：ロック制御メカニズムは存在しません。

1：ロック制御メカニズムが実装されています。

ETM 認証ステータスレジスタ（M7_ETM_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r		r		r		r	

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:6　**SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]：**保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ビット 1:0 **NSID[1:0]：**無保護の侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ETM CoreSight デバイスアーキテクチャレジスタ（M7_ETM_DEVARCH）

アドレスオフセット：0xFBC

リセット値：0x4770 4A13

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARCHITECT[10:0]											PRESENT	REVISION[3:0]			
r											r	r			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARCHID[15:0]															
r															

ビット 31:21 **ARCHITECT[10:0]**：コンポーネントアーキテクチャ
0x23B：ARM®

ビット 20 **PRESENT**：このレジスタの有無を示します。
1：有ります。

ビット 19:16 **REVISION[3:0]**：アーキテクチャのリビジョン
0x0：リビジョン 0

ビット 15:0 **ARCHID[15:0]**：アーキテクチャ ID
0x4A13：ETMv4 コンポーネント

ETM CoreSight デバイス ID レジスタ（M7_ETM_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:0 予約済みであり、リセット値に保持する必要があります。

ETM CoreSight デバイスタイプ ID レジスタ（M7_ETM_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0013

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE**：デバイスのサブタイプ識別子

0x1：プロセッサのトレース

ビット 3:0 **MAJORTYPE**：デバイスのメインタイプ識別子

0x3：トレースソース

ETM CoreSight ペリフェラル ID レジスタ 4（M7_ETM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：ARM® JEDEC コード

ETM CoreSight ペリフェラル ID レジスタ 0（M7_ETM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0075

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールド、ビット [7:0]

0x75：ETM 部品番号

ETM CoreSight ペリフェラル ID レジスタ 1（M7_ETM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：ARM® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールド、ビット [11:8]

0x9：ETM 部品番号

ETM CoreSight ペリフェラル ID レジスタ 2（M7_ETM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 002B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
								r				r	r		

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p2

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定される設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：ARM® JEDEC コード

ETM CoreSight ペリフェラル ID レジスタ 3（M7_ETM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

ETM CoreSight コンポーネント ID レジスタ 0（M7_ETM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールド、ビット [7:0]
0x0D：コモン ID 値

ETM CoreSight コンポーネント ID レジスタ 1（M7_ETM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールド、ビット [15:12] - コンポーネントクラス
0x9：CoreSight 互換レジスタ付きデバッグコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールド、ビット [11:8]
0x0：コモン ID 値

ETM CoreSight コンポーネント ID レジスタ 2（M7_ETM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールド、ビット [23:16]
0x05：コモン ID 値

ETM CoreSight コンポーネント ID レジスタ 3（M7_ETM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：コンポーネント ID フィールド、ビット [31:24]

0xB1：コモン ID 値

Cortex-M7 ETM レジスタのマップとリセット値

ETM レジスタへは、Cortex-M7 PPB を介してデバッガからアクセスでき、アクセス可能なアドレスレンジは 0xE0041000~0xE0041FFC です。

表 584. Cortex-M7 ETM レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004	M7_ETM_PRGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																																0
0x008	M7_ETM_PROCSEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PROCSEL[2:0]		
	リセット値																														0	0	0
0x00C	M7_ETM_STAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PMSTABLE	IDLE
	リセット値																															0	0
0x010	M7_ETM_CONFIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	Res.	Res.	Res.	RS	TS	COND[2:0]			VMID	CID	Res.	CCI	BB	INSTP0[1:0]		Res.
	リセット値															0	0				0	0	0	0	0	0	0		0	0	0	0	1
0x020	M7_ETM_EVENTCTL0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TYPE1	Res.	Res.	Res.	SEL1[3:0]				TYPE0	Res.	Res.	Res.	SEL0[3:0]			
	リセット値																	0				0	0	0	0	0				0	0	0	0
0x024	M7_ETM_EVENTCTL1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[3:0]			
	リセット値																				0	0								0	0	0	0
0x02C	M7_ETM_STALLCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSTALL	ISATLL	Res.	Res.	Res.	Res.	LEVEL[1:0]		Res.	Res.
	リセット値																							0	0					0	0		
0x030	M7_ETM_TSCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TYPE	Res.	Res.	Res.	SEL[3:0]			
	リセット値																									0				0	0	0	0
0x034	M7_ETM_SYNCP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
	リセット値																												0	0	0	0	0
0x038	M7_ETM_CCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	THRESHOLD[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x040	M7_ETM_TRACEID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACEID[6:0]						
	リセット値																										0	0	0	0	0	0	0

表 584. Cortex-M7 ETM レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080	M7_ETM_VICTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S3	Res.	Res.	EXLEVEL_S0	Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSSTATUS	Res.	TYPE	Res.	Res.	Res.	SEL[3:0]			
	リセット値													0			0					0	0	0		0				0	0	0	0
0x088	M7_ETM_VISSCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[7:0]							
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x08C	M7_ETM_VIPCSSCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[3:0]			
	リセット値													0	0	0	0													0	0	0	0
0x140	M7_ETM_CNTRLDV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALUE[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x180	M7_ETM_IDR8	MAXSPEC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0x184	M7_ETM_IDR9	NUMP0KEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x188	M7_ETM_IDR10	NUMP1KEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18C	M7_ETM_IDR11	NUMP1SPC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x190	M7_ETM_IDR12	NUMCONDKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x194	M7_ETM_IDR13	NUMCONDSPC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C0	M7_ETM_IMSPEC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUPPORT[3:0]			
	リセット値																													0	0	0	0
0x1E0	M7_ETM_IDR0	Res.	Res.	COMMOPT	TSSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSUPP[1:0]		Res.	CONDTYPE[1:0]		NUMEVENT[1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	Res.	Res.	Res.	Res.	Res.
	リセット値			0	0	1	0	0	0								0	0		0	1	1	1	1		1	1	1				1
0x1E4	M7_ETM_IDR1	DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRCARCHMAJ[3:0]				TRCARCHMIN[3:0]				REVISION[3:0]			
	リセット値	0	0	1	0	0	0	0	1									1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	1
0x1E8	M7_ETM_IDR2	Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:0]				VMIDSIZE[4:0]				CIDSIZE[4:0]				IASIZE[4:0]							
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0x1EC	M7_ETM_IDR3	NOOVERFLOW	NUMPROC[2:0]			SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]				Res.	Res.	Res.	Res.	CCITMIN[11:0]											
	リセット値	0	0	0	0	0	1	0	1					1	0	0	1					0	0	0	0	0	0	0	0	0	0	1	0	0

表 584. Cortex-M7 ETM レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1F0	M7_ETM_IDR4	NUMVMIDC [3:0]				NUMCIDC [3:0]				NUMSSCC [3:0]				NUMRSPAIR [3:0]				NUMPC [3:0]				Res.	Res.	Res.	SUPPDAC	NUMDVC [3:0]				NUMACPAIRS [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0				0	0	0	0	0	0	0	0	0
0x1F4	M7_ETM_IDR5	REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE [2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE[5:0]						Res.	Res.	Res.	Res.	NUMEXTINSEL [2:0]			NUMEXTIN[8:0]								
	リセット値	1	0	0	1	0	0	0		1	1	0	0	0	1	1	1					0	1	0	0	0	0	0	0	0	0	1	0
0x208	M7_ETM_RSCTL2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値											0	0		0	0	0									0	0	0	0	0	0	0	0
0x20C	M7_ETM_RSCTL3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値											0	0		0	0	0									0	0	0	0	0	0	0	0
0x280	M7_ETM_SSCC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0																								
0x2A0	M7_ETM_SSCS0	STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	INST
	リセット値	0																												0	0	1
0x2C0	M7_ETM_SSPCIC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x310	M7_ETM_PDC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																													0			
0x314	M7_ETM_PDS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
	リセット値																															1	1
0xFA0	M7_ETM _CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SET[3:0]			
	リセット値																													0	1	0	0
0xFA4	M7_ETM _CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR[3:0]			
	リセット値																													0	0	0	0
0xFB0	M7_ETM_LAR	KEY[31:0]																															
	リセット値																																
0xFB4	M7_ETM_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTT	SLK	SLI
	リセット値																														0	1	1
0xFB8	M7_ETM _AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID [1:0]		SID [1:0]		NSNID [1:0]		NSID [1:0]	
	リセット値																									0	0	0	0	0	0	0	0

表 584. Cortex-M7 ETM レジスタのマップとリセット値（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFBC	M7_ETM _DEVARCH	ARCHITECT[10:0]											PRESENT	REVISION [3:0]				ARCHID[15:0]															
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	1	0	0	1	1
0xFC8	M7_ETM_DEVID	DEVICEID[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xFD0	M7_ETM _DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE [3:0]				MAJORTYPE [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
0xFD0	M7_ETM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4	M7_ETM_PIDR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFD8	M7_ETM_PIDR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFDC	M7_ETM_PIDR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xFE0	M7_ETM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM [7:0]							
	リセット値																									0	1	1	1	0	1	0	1
0xFE4	M7_ETM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM [11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	M7_ETM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION [3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	M7_ETM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_ETM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_ETM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE [11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	M7_ETM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_ETM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

60.6.6 Cortex-M7 クロストリガインタフェース（CTI）

*セクション 60.5.3*を参照してください。

参考資料

60.7 デバッグインフラストラクチャのリファレンスマニュアル

1. IHI 0031C （ID080813）- ARM® Debug Interface Architecture Specification ADIv5.0~ADIv5.2、C 版
2. DDI 0480F （ID100313）- ARM® CoreSight™ SoC-400 r3p2 Technical Reference Manual、G 版
3. DDI 0461B （ID010111）- ARM® CoreSight™ Trace Memory Controller r0p1 Technical Reference Manual、B 版
4. DDI 0314H - ARM® CoreSight™ Components Technical Reference Manual、H 版
5. DDI 0403D （ID100710）- ARM®v7-M Architecture Reference Manual、E.b 版
6. DDI 0494-2a （ID062813）- ARM® CoreSight™ ETM™-M7 r0p1 Technical Reference Manual、D 版

61 デバイス電子署名

電子署名は、フラッシュメモリ領域に格納され、JTAG／SWD または CPU を使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データを使用すれば、ユーザファームウェアやその他の外部デバイスは、そのインタフェースを STM32H7x3 マイクロコントローラの特性に自動的に整合させることができます。

61.1 ユニークデバイス ID レジスタ（96 ビット）

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号（例：USB 文字列シリアル番号やその他のエンドアプリケーション）として使用
- 内部フラッシュメモリをプログラムする前に、このユニーク ID をソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、フラッシュメモリ内のコードのセキュリティを高めるためのセキュリティキーとして使用
- セキュアなブートプロセスなどの起動に使用

96 ビットのユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

96 ビットのユニークデバイス識別子は、さまざまな方法で 1 バイト／ハーフワード／ワード単位で読み出し、カスタムアルゴリズムを使用して連結することもできます。

ベースアドレス：0x1FF0 F420

アドレスオフセット：0x00

読み出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(31:0)																															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(31:0)：**31:0 ユニーク ID ビット

アドレスオフセット：0x04

読み出し専用　= 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(63:48)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(47:32)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(63:32)：**63:32 ユニーク ID ビット

アドレスオフセット：0x08

読み出し専用　= 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(95:80)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(79:64)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(95:64) :** 95:64 ユニーク ID ビット

61.2 フラッシュサイズ

ベースアドレス：0x1FF0 F442

アドレスオフセット：0x00

読み出し専用　= 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F_SIZE															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15:0 F_ID(15:0) : フラッシュメモリサイズ

このビットフィールドは、デバイスのフラッシュメモリサイズを KB 単位で示します。
たとえば、0x0400 は 1024 KB に対応します。

61.3 パッケージデータレジスタ

ベースアドレス：0x1FFF 7BF0

アドレスオフセット：0x00

読み出し専用　= 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PKG[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw	rw	rw								

ビット 15:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:8　PKG[2:0]：パッケージタイプ

TBD：LQFP100 レガシーライン
TBD：TQFP144 レガシーライン／UFBGA169 グラフィックライン／TQFP176 グラフィックライン
TBD：TQFP144 インダストリアルライン
TBD：TQFP176 レガシーライン／UFBGA176 レガシーライン
TBD：TQFP176 インダストリアルライン
TBD：UFBGA176 インダストリアルライン
TBD：LQFP208 レガシーライン／TFBGA 240
TBD：TQFP208 グラフィックライン
TBD：TQFP208 インダストリアルライン

ビット 7:0　予約済みであり、リセット値に保持する必要があります。

改版履歴

表 585. 文書改版履歴

日付	版	変更内容
2016 年 12 月 12 日	1	初版発行
2017 年 5 月 26 日	2	*セクション 2* メモリとバスのアーキテクチャ名を変更。 *セクション ：内蔵ブートローダ*を更新。 ***セクション 3：内蔵フラッシュメモリ（フラッシュ）*** *セクション ：フラッシュセクタの消去*、*セクション ：フラッシュバンク消去*、*セクション ：自動保護解除を伴うフラッシュバンク消去* および *セクション ：フラッシュ全体消去*を更新。 *セクション ：自動保護解除を伴うフラッシュ全体消去*を追加。 *表 16：レジスタマップとリセット値の表*を更新。 – FLASH_BOOT7_CURR/PRG （アドレスオフセット = 0x140/0x144）は RM0399 でのみ使用可能。 – アドレスオフセット = 0x158 に FLASH_CRCEADD2R を追加。 – アドレスオフセット = 0x15C に FLASH_CRCEADD2R を追加。 ***セクション 6：電源制御（PWR）*** *セクション 6.3.1：PWR ピンおよび内部信号*を更新。 *セクション 6.4：電源*の V_{BAT} を更新。 *図 13：電源の概要* および *図 15：電圧レギュレータから供給される VCORE によるデバイスの起動*を更新。 *セクション 6.5：電源供給の監視*の VCORE を削除。 *図 24：VCORE 電圧スケーリングとシステム電力モード* および *図 25：電源制御モードの詳細な状態図*を更新。 ***セクション 8：リセットおよびクロック制御（RCC）*** すべてのクロックソース選択ビットを XXSCR から XXSEL に名称変更。 RC48 から HSI48 へ名称変更。 RCC_BDCR の VSWRST ビットを BDRST に名称変更し、ビットの説明を変更。 RCC_AHB2ENR の CAMITFEN ビットを DCMIEN に名称変更、 RCC_AHB2LPENR の CAMITFLPEN ビットを DCMILPEN に名称変更。 RCC_AHB3LPENR の FLITFLPEN ビットを FLASHLPEN に名称変更。 RCC_APB1LLPENR の HDMICECLPEN ビットを CECLPEN に名称変更。 文書全体で、ペリフェラルのカーネルクロックの名称を更新。 V_{DDA} を削除して、*図 34：システムリセット回路* を更新。 *表 51：カーネルクロック分配の概要*の ADC1、2、3 の最大周波数を更新。 *セクション ：制御およびデータ転送専用のペリフェラル*に USBxOTG クロックとして記載されている lsi_ck を削除。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 10：ハードウェアセマフォ（HSEM）*** 文書全体で、pclk を hsem_hclk に名称変更。 *セクション 10.4.1：HSEM レジスタ（HSEM_R0 - HSEM_R31）*の COREID ビットの説明を更新。 ***セクション 11：汎用 I/O（GPIO）*** *表 84：GPIO レジスタマップとリセット値*： – GPIOx_MODER のリセット値を更新。 – GPIOx_AFRH のインデックスを A～K に変更。 – GPIOC～K_PUPDR を追加。 ***セクション 12：システム設定コントローラ（SYSCFG）*** SYSCFG_UR3 レジスタを更新。 ***セクション 13：ブロック相互接続*** *表 92：DMAMUX1、DMA1、および DMA2 接続*の dac1_dma と dac2_dma をそれぞれ dac_ch1_dma と dac_ch2_dma に名称変更。 *表 88：ペリフェラル相互接続マトリックスの詳細* および *表 89：EXTI ウェイクアップ入力*のいくつかの転送元信号と転送先信号を更新。 ***セクション 17：DMA リクエストマルチプレクサ（DMAMUX）*** *表 110：DMAMUX1：リソースへのマルチプレクサ入力の割り当て* ～ *表 112：DMAMUX1：リソースへの同期入力の割り当て*のリソースを更新。 ***セクション 19：ネスト化されたベクタ割り込みコントローラ*** *表 130：NVIC*の LCD-TFT 割りこみ（ltdc_it および ltdc_err_it）を追加。 ***拡張割り込み／イベントコントローラ（EXTI）*** *表 133：EXTI イベント入力の配置*のイベント 66～73 について、DMA1 を BDMA に置換。 ***セクション 22：フレキシブルメモリコントローラ（FMC）*** – *図 86：FMC ブロック図*の内部信号を更新。 – HCLK から fmc_hclk に名称変更。 – KCK_FMC から fmc_ker_ck に名称変更。 32 ビットアクセスを追加して、*セクション 22.5：AXI インタフェース* を更新。 読み出し FIFO の深さを 6x64 ビットに変更。AXI バスの幅を修正（32 ビットではなく 64 ビットに）。 データバスおよび NBL ビットに関しては、すべての波形を一般波形に変更。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 22：フレキシブルメモリコントローラ（FMC）*（続き）** *セクション ：SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）*において、DATAST の例を変更し、BURSTURN の説明を更新。*セクション ：SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1～4（FMC_BWTR1～4）*において、BURSTURN の説明を更新。ビットフィールド幅を追加して、*セクション ：SDRAM 制御レジスタ 1、2（FMC_SDCR1、2）*を更新。 *表 181：FMC レジスタマップ*に不足していた FMC_SDCR2 ビットを追加。 ***セクション 23：Quad SPI インタフェース（QUADSPI）*** *図 115：QUADSPI ブロック図（デュアルフラッシュモードが無効のとき）*および *図 116：QUADSPI ブロック図（デュアルフラッシュモードが有効のとき）*の内部信号を更新。*セクション 23.3.2：QUADSPI ピンおよび内部信号*を追加。*セクション 23.3.7：QUADSPI メモリマップドモード*において、Cortex CPU によるアクセスのエラータイプを変更。 *セクション 23.3.8：QUADSPI フリーランニングクロックモード*および *セクション 23.5.6：QUADSPI 通信設定レジスタ（QUADSPI_CCR）*の FRCM ビットを追加。*セクション 23.5.1：QUADSPI 制御レジスタ（QUADSPI_CR）*を更新。*表 185：QUADSPI レジスタマップとリセット値*において、QUADSPI_CR レジスタの DMAEN ビットを予約済みに変更。 ***セクション 24：遅延ブロック（DLYB）*** *図 123：DLYB ブロック図*に内部信号を追加し、*セクション 24.3.2：DLYB ピンおよび内部信号*を追加。 ***セクション 25：アナログデジタルコンバータ（ADC）*** *セクション 25.1：概要*の ADC の数を 3 に変更。 *セクション 25.3.1：ADC ブロック図*に内部信号を追加。ADCx_IN[19:0] を *図 124：ADC ブロック図*では ADCx_INP[19:0] に、*表 190：ADC ピン*では ADCx_INN[19:0] に変更。ADC_CLK から adc_ker_ck に変更。 *図 126：ADC1 接続性*、*図 127：ADC2 接続性*、および *図 128：ADC3 接続性* を更新し、下図に注を追加。*図 175：デュアル ADC ブロック図（1）*を更新。 *セクション ：デュアルクロックドメインアーキテクチャ*に注を更新。 *セクション 25.3.11：チャネルの選択（SQRx、JSQRx）*および *セクション 25.3.12：プリチャネル選択レジスタ（ADCx_PCSEL）*を更新。 *セクション 25.3.13：チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）*および *セクション 25.3.17：タイミング*に、72 MHz ではなく 24 MHz の例を表示。 6 ビット解像度を削除し、*表 195：TSAR タイミングは分解能に依存*の *セクション 25.3.23：プログラム可能な分解能（RES）- 高速変換モード*を更新。 *セクション ：レギュラ／インジェクト同時モードの組み合わせ*および *セクション ：レギュラ同時 + オルタネートトリガモードの組み合わせ*の注を更新。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 25：アナログデジタルコンバータ（ADC）*（続き）** *セクション ：デュアル ADC 同時モードにおける DFSDM モード*を追加。 *セクション 25.3.33：温度センサ*の ADC3_IN18 を ADC3 VINP[18] に変更。 *セクション 25.3.34：VBAT 電源監視*の ADC3_IN17 を ADC3 VINP[17] に変更。 *セクション 25.3.35：内部電圧基準の監視*の ADC3_IN19 を ADC3 VINP[19] に変更。 *セクション 25.5.15：ADC レギュラデータレジスタ（ADCx_DR、x=1..3）*のすべてのビットのアクセスタイプを「r」に変更。 *表 206：ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300）*を更新。 ***セクション 26：D/A コンバータ（DAC）*** *図 194：DAC チャネルブロック図*の内部信号を更新または追加。*セクション 26.3.2：DAC ピンおよび内部信号*を追加。 文書全体で、APB1 クロックおよび LSI クロックをそれぞれ dac_pclk と lsi_ck に置換。 ***セクション 28：コンパレータ（COMP）*** *セクション 28.7.1：コンパレータステータスレジスタ（COMP_SR）*の COMP_IFCR を COMP_ICFR に置換。 ***セクション 29：オペアンプ（OPAMP）*** DACx_int をすべて dac_outx に置換（x = 1、2）。*セクション 29.6.1：OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR）*レジスタの VP_SEL ビットの説明を更新。 ***セクション 31：デジタルカメラインタフェース（DCMI）*** セクション*DCMI ピン*を *セクション 31.4.4：DCMI 物理インタフェース*と結合。 文書全体で、外部信号 HSYNC、VSYNC、および PIXCLK をそれぞれ DCMI_HSYNC、DCMI_VSYNC、および DCIM_PIXCLK に標準化。 *図 224：DCMI のブロック図* および *図 225：最上位ブロック図*を更新。 ***セクション 32：LCD-TFT ディスプレイコントローラ（LTDC）*** *LCD-TFT ピンと信号インタフェース* を *LCD-TFT ピンと外部信号インタフェース*に名称変更。 *図 233：LTDC のブロック図*の内部信号名を更新。 文書全体で、ck_axi_d1 を ltdc_aclk に名称変更。 *セクション 32.3.4：LTDC のリセットおよびクロック*を更新。 *セクション 32.7.23：LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）（x=1..2）*の CFBP ビットフィールドのサイズを更新。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 33：JPEG コーデック（JPEG）*** *図 238：JPEG コーデックブロック図*に内部信号を追加し、*セクション 33.3.2：JPEG 内部信号*を追加。 DMA の機能を削除。 *表 258：JPEG コーデックレジスタマップとリセット値*の JPEG_CR、JPEG_SR、および JPRG_CFR レジスタを更新。 ***セクション 37：高分解能タイマ（HRTIM）*** セクション全体で、SCOUT を HRTIM_SCOUT に名称変更。 参照されるすべての図において、Tx を HRTIM_CHxy に名称変更。 *図 274：高分解能タイマのブロック図*の内部信号を更新。*表 282：HRTIM の入出力の概要*に hrtim_in_sync1 および hrtim_out_sync1 を追加。 *セクション ：用語の定義*を更新。 参照されるすべての図および HRTIM 機能の説明に内部信号名を更新。 hrtim_ker_ck に関する注を *表 282：HRTIM の入出力の概要*に追加。 *セクション 37.5.56：HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R）*の ADC3TAPER ビットの説明を変更。 *セクション 37.5.57：HRTIM ADC トリガ 4 レジスタ（HRTIM_ADC4R）*の ADC4TCRST、ADC4TAPER、ADC4TAC2、ADC4EEV6、ADC4MPER、および DC4MC1 ビットの説明を変更。 ***セクション 43：低電力タイマ（LPTIM）*** *図 514：低電力タイマのブロック図（LPTIM1 と LPTIM2）*、*図 515：低電力タイマのブロック図（LPTIM3）*、および *図 516：低電力タイマのブロック図（LPTIM4 と LPTIM5）*に内部信号を追加。*セクション 43.4.2：LPTIM ピンおよび内部信号*を追加。 *表 334：LPTIM1 外部トリガ接続* を *表 338：LPTIM5 外部トリガ接続*に更新し、*セクション 43.4.3：LPTIM 入力およびトリガマッピング*に移動。 *表 339：LPTIM1 入力 1 接続* を *表 343：LPTIM3 入力 1 接続*に更新し、*セクション 43.4.3：LPTIM 入力およびトリガマッピング*に移動。 *セクション 43.6.4：LPTIM 設定レジスタ（LPTIM_CFGR）*の TRIGSEL ビットフィールドの説明を更新。 *セクション 43.6.5：LPTIM 制御レジスタ（LPTIM_CR）*に COUNTRST ビットに関する注意書きを追加。 *セクション 43.6.9：LPTIM 設定レジスタ 2 （LPTIMx_CFGR2、x=1、2）*において、IN1SEL および IN2SEL ビットフィールドの説明を更新し、注意書きを追加。 *セクション 43.6.10：LPTIM3 設定レジスタ 2（LPTIM3_CFGR2）*に注意書きを追加。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 46：リアルタイムクロック（RTC）*** *図 526：RTC ブロックの概要*を追加。 *図 527：詳細な RTC ブロック図*のタイトルを更新。 *セクション 46.3.2：RTC ピンおよび内部信号*を追加。*表 352：RTC ピンおよび内部信号*を更新。 *セクション 46.3.3：RTC によって制御される GPIO*を更新。 *セクション 46.6.16：RTC タンパ設定レジスタ（RTC_TAMPCR）*を更新。 ***セクション 47：I2C（Inter-integrated circuit）インタフェース*** *セクション 47.7.3：Own Address 1 レジスタ（I2C_OAR1）*および *セクション 47.7.4：Own Address 2 レジスタ（I2C_OAR2）*の OA1[7:1] および OA2[7:1] ビットの説明を更新。 *セクション 47.4.14：アドレス一致時に STOP モードからウェイクアップ*の HSI16 を HSI、CSI、または内部オシレータに置換。 ***セクション 48：USART（Universal synchronous asynchronous receiver transmitter）*** *セクション 48.5.14：USART 同期モード*において、図 *RX データセットアップ／ホールド時間*を削除、*図 573：同期マスタモードでの USART データクロックタイミング図（M ビット="00"）*および *図 574：同期マスタモードでの USART データクロックタイミング図（M ビット="01"）*に同期マスタモードについての記載を追加し、1 個ではなく 2 個の M ビットについて示すよう図の内容を更新。 *セクション 48.5.21：USART 低電力管理*の*図 585：確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）*を更新。 ***セクション 51：シリアルオーディオインタフェース（SAI）*** *セクション 51.5.1：グローバル設定レジスタ（SAI_GCR）*の SYNCIN ビットフィールドの説明を更新。 ***セクション 50：シリアルペリフェラルインタフェース（SPI）*** *セクション ：単方向通信*の注を更新。 *セクション ：PCM 規格*に PCM のロングおよびショートフレームの定義を追加。*セクション 50.9.3：I2S/PCM モードで使用できるビットおよびフィールド*の記載および *表 390：PCM/I2S モードで使用できるビットフィールド*を追加。 *セクション 50.11.14：SPI/I2S 設定レジスタ（SPI_I2SCGFR）*および *セクション 50.12：SPI レジスタマップとリセット値*の WSINV を更新。 ***セクション 52：SPDIF レシーバインタフェース（SPDIFRX）*** *セクション 図 657.：SPDIFRX ブロック図*に内部信号を追加し、*セクション 52.3.1：SPDIFRX ピンおよび内部信号*を追加。

表 585. 文書改版履歴（続き）

日付	版	変更内容
2017 年 5 月 26 日	2（続き）	***セクション 55：セキュアデジタル入力／出力 MultiMediaCard インタフェース（SDMMC）*** – セクション全体の内部信号を名称変更（*図 693：SDMMC ブロック図*の更新、および *セクション 55.4.2：SDMMC ピンおよび内部信号*の追加）。 – *セクション 55.4.7：MDMA リクエストの生成*を追加 – SDMMC_VER、SDMMC_ID、および SDMMC_SID を削除。 – *表 448：SDMMC レジスタマップ*の SDMMC_STAR ビットの 12 と 13 を更新。 ***セクション 56：FD コントローラエリアネットワーク（FDCAN）*** セクション全体から ASC（非同期シリアル通信）を削除。 *FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）*および *FDCAN Rx FIFO 1 設定レジスタ（FDCAN_RXF1C）*のぞれぞれに、F0OM ビット（ビット 31）と F1OM ビット（ビット 31）を追加。***セクション 57：USB On-The-Go ハイスピード（OTG_HS）*** *セクション 57.4.2：USB OTG ピンおよび内部信号*を追加。 ***セクション 60：デバッグインフラストラクチャ*** *セクション 60.5.8：マイクロコントローラデバッグユニット（DBGMCU）*を更新。 DBGMCU のレジスタ名をすべて次のように置換。 – D1APB1 を APB3 に置換 – D2APB1 を APB1 に置換 – D2APB2 を APB2 に置換 – D3APB4byAPB4 *セクション ：DBGMCU APB4 ペリフェラルフリーズレジスタ CPU（DBGMCU_APB4FZ1）*DBGMCU_APB4FZ1 の WDGLSD2 ビットを予約済みに変更。 *表 576：DBGMCU レジスタのマップとリセット値*の予約済みレジスタを削除。

表 586. 日本語版文書改版履歴

日付	版	変更内容
2018 年 3 月 30 日	1	日本語版　初版発行

索引

A

ADCx_AWD2CR 956
ADCx_AWD3CR 957
ADCx_CALFACT 960
ADCx_CALFACT2、x=1..3 961
ADCx_CCR 964
ADCx_CDR 967
ADCx_CDR2、x=12 または 3 967
ADCx_CFGR 938
ADCx_CFGR2 942
ADCx_CR 933
ADCx_CSR 962
ADCx_DIFSEL 959
ADCx_DR 952
ADCx_HTR2、x=1..3 958
ADCx_HTR3、x=1..3 959
ADCx_IER 931
ADCx_ISR 929
ADCx_JDRy 956
ADCx_JSQR 953
ADCx_LTR2、x=1..3 957
ADCx_LTR3 958
ADCx_OFRy 955
ADCx_OR、x=1..3 961
ADCx_PCSEL、x=1..3 946
ADCx_SMPR1 944
ADCx_SMPR2 945
ADCx_SQR1 948
ADCx_SQR2 949
ADCx_SQR3 950
ADCx_SQR4 951
ADCx_TR1 946-947
AXI_COMP_ID_0 202
AXI_COMP_ID_1 203
AXI_COMP_ID_2 203
AXI_COMP_ID_3 204
AXI_INIx_FN_MOD 208
AXI_INIx_FN_MOD_AHB 207
AXI_INIx_FN_MOD2 206
AXI_INIx_READ_QOS 207
AXI_INIx_WRITE_QOS 208
AXI_PERIPH_ID_0 200
AXI_PERIPH_ID_1 201
AXI_PERIPH_ID_2 201
AXI_PERIPH_ID_3 202
AXI_PERIPH_ID_4 200
AXI_TARGx_FN_MOD 206
AXI_TARGx_FN_MOD_ISS_BM 204
AXI_TARGx_FN_MOD_LB 205
AXI_TARGx_FN_MOD2 205

C

CAN_TTGTP 2463
CCU_CCFG 2482
CCU_CREL 2482
CCU_CSTAT 2484
CCU_CWD 2485
CCU_IE 2486
CCU_IR 2486
CEC_CFGR 2902
CEC_CR 2901
CEC_IER 2907
CEC_ISR 2905
CEC_RXDR 2904
CEC_TXDR 2904
COMP_CFGR1 1021
COMP_CFGR2 1023
COMP_ICFR 1019
COMP_OR 1020
COMP_SR 1019
CRC_CR 727
CRC_DR 726
CRC_IDR 726
CRC_INIT 728
CRC_POL 728
CRS_CFGR 459
CRS_CR 458
CRS_ICR 462
CRS_ISR 460
CRYP_CR 1245
CRYP_DIN 1247
CRYP_DMACR 1249
CRYP_DOUT 1248
CRYP_IMSCR 1249
CRYP_IV0RR 1256
CRYP_IV1LR 1256
CRYP_IV1RR 1257
CRYP_K0LR 1252
CRYP_K1LR 1252
CRYP_K1RR 1253
CRYP_K2LR 1253
CRYP_K2RR 1254
CRYP_K3LR 1254
CRYP_K3RR 1255
CRYP_MISR 1251
CRYP_RISR 1250

CRYP_SR 1247
CSTF_AUTHSTAT 2977
CSTF_CIDR0 2981
CSTF_CIDR1 2981
CSTF_CIDR2 2981
CSTF_CIDR3 2982
CSTF_CLAIMCLR 2976
CSTF_CLAIMSET 2975
CSTF_CTRL 2974
CSTF_DEVID 2978
CSTF_LAR 2976
CSTF_LSR 2977
CSTF_PIDR0 2979
CSTF_PIDR1 2979
CSTF_PIDR2 2979
CSTF_PIDR3 2980
CSTF_PIDR4 2980
CSTF_PRIORITY 2975
CSTF_TYPEID 2978
CTI_APPCLEAR 2959
CTI_APPPULSE 2959
CTI_APPSET 2958
CTI_AUTHSTAT 2965
CTI_CHINSTS 2962
CTI_CHOUTSTS 2962
CTI_CIDR0 2969
CTI_CIDR1 2969
CTI_CIDR2 2970
CTI_CIDR3 2970
CTI_CLAIMCLR 2964
CTI_CLAIMSET 2963
CTI_CONTROL 2957
CTI_DEVID 2966
CTI_DEVTYPE 2966
CTI_GATE 2963
CTI_INENx 2960
CTI_INTACK 2957
CTI_LAR 2964
CTI_LSR 2965
CTI_OUTENx 2960
CTI_PIDR0 2967
CTI_PIDR1 2968
CTI_PIDR2 2968
CTI_PIDR3 2969
CTI_PIDR4 2967
CTI_TRGISTS 2961
CTI_TRGOSTS 2961

D

DAC_CCR 1000
DAC_CR 989
DAC_DHR12L1 994
DAC_DHR12L2 995
DAC_DHR12LD 997
DAC_DHR12R1 993
DAC_DHR12R2 995
DAC_DHR12RD 996
DAC_DHR8R1 994
DAC_DHR8R2 996
DAC_DHR8RD 997
DAC_DOR1 998
DAC_DOR2 998
DAC_MCR 1000
DAC_OR 1004
DAC_SHHR 1002
DAC_SHRR 1003
DAC_SHSR1 1001
DAC_SHSR2 1002
DAC_SR 999
DAC_SWTRIGR 993
DBGMCU_CR 3045
DBGMCU_D1APB1FZ1 3047
DBGMCU_D2APB1HFZ1 3048
DBGMCU_D2APB1LFZ1 3047
DBGMCU_D2APB2FZ1 3049
DBGMCU_D3APB4FZ1 3050
DBGMCU_IDC 3045
DCMI_CR 1116
DCMI_CWSIZE 1126
DCMI_CWSTRT 1126
DCMI_DR 1127
DCMI_ESCR 1124
DCMI_ESUR 1125
DCMI_ICR 1123
DCMI_IER 1121
DCMI_MIS 1122
DCMI_RIS 1120
DCMI_SR 1119
DFSDM_CHyAWSCDR 1077
DFSDM_CHyCFGR1 1074
DFSDM_CHyCFGR2 1076
DFSDM_CHyDATINR 1078
DFSDM_CHyWDATR 1078
DFSDM_FLTxAWCFR 1091
DFSDM_FLTxAWHTR 1089
DFSDM_FLTxAWLTR 1090
DFSDM_FLTxAWSR 1090
DFSDM_FLTxCNVTIMR 1092
DFSDM_FLTxCR1 1079
DFSDM_FLTxCR2 1082
DFSDM_FLTxEXMAX 1091
DFSDM_FLTxEXMIN 1092
DFSDM_FLTxFCR 1087

DFSDM_FLTxICR 1085
DFSDM_FLTxISR 1083
DFSDM_FLTxJCHGR 1086
DFSDM_FLTxJDATAR 1088
DFSDM_FLTxRDATAR 1088
DLYB_CFGR 845
DLYB_CR 845
DMA_CCRx 622
DMA_CMARx 625
DMA_CNDTRx 624
DMA_CPARx 624
DMA_HIFCR 603
DMA_HISR 602
DMA_IFCR 621
DMA_ISR 620
DMA_LIFCR 563, 603
DMA_LISR 561-562, 564, 601
DMA_SxCR 566, 568, 604
DMA_SxFCR 609
DMA_SxM0AR 608
DMA_SxM1AR 609
DMA_SxNDTR 571, 607
DMA_SxPAR 573, 575-578, 608
DMA2D_AMTCR 680
DMA2D_BGCMAR 676
DMA2D_BGCOLR 675
DMA2D_BGMAR 669
DMA2D_BGOR 669
DMA2D_BGPFCCR 673
DMA2D_CR 664
DMA2D_FGCMAR 675
DMA2D_FGCOLR 672
DMA2D_FGMAR 668
DMA2D_FGOR 668
DMA2D_FGPFCCR 670
DMA2D_IFCR 667
DMA2D_ISR 666
DMA2D_LWR 680
DMA2D_NLR 679
DMA2D_OCOLR 677
DMA2D_OMAR 678
DMA2D_OOR 679
DMA2D_OPFCCR 676
DMAMUX1_CSR 642
DP_DPIDR 2922

E

ETF_AUTHSTAT 2998
ETF_BUFWM 2992
ETF_CBUFLVL 2992
ETF_CIDR0 3002
ETF_CIDR1 3002
ETF_CIDR2 3003
ETF_CIDR3 3003
ETF_CLAIMCLR 2997
ETF_CLAIMSET 2996
ETF_CTL 2989
ETF_DEVID 2999
ETF_DEVTYPE 2999
ETF_FFCR 2993
ETF_FFSR 2993
ETF_LAR 2997
ETF_LBUFLVL 2991
ETF_LSR 2998
ETF_MODE 2990
ETF_PIDR0 3000
ETF_PIDR1 3001
ETF_PIDR2 3001
ETF_PIDR3 3002
ETF_PIDR4 3000
ETF_PSCR 2996
ETF_RRD 2987
ETF_RRP 2988
ETF_RSZ 2986
ETF_RWD 2990
ETF_RWP 2988
ETF_STS 2986
ETF_TRG 2989
ETH_DMAC0CARxBR 2777
ETH_DMAC0CARxDR 2776
ETH_DMAC0RxCR 2765
ETH_DMAC0RxDLAR 2768
ETH_DMAC0RxDTPR 2770
ETH_DMAC0RxIWTR 2775
ETH_DMAC0RxRLR 2771
ETH_DMACiCATxBR 2776
ETH_DMACiCATxDR 2775
ETH_DMACiCR 2762
ETH_DMACiIER 2771
ETH_DMACiMFCR 2780
ETH_DMACiSR 2777
ETH_DMACiTxCR 2762
ETH_DMACiTxDLAR 2767
ETH_DMACiTxDTPR 2769
ETH_DMACiTxRLR 2770
ETH_DMADSR 2761
ETH_DMAISR 2760
ETH_DMAMR 2757
ETH_DMASBMR 2759
ETH_MAC1USTCR 2830
ETH_MACA0HR 2839
ETH_MACACR 2869
ETH_MACARPAR 2839

ETH_MACATSNR 2870
ETH_MACATSSR 2871
ETH_MACAxHR 2841
ETH_MACAxLR 2840
ETH_MACCR 2795
ETH_MACDR 2831
ETH_MACECR 2801
ETH_MACHT0R 2807
ETH_MACHT1R 2808
ETH_MACHWF1R 2832
ETH_MACHWF2R 2834
ETH_MACIER 2819
ETH_MACISR 2817
ETH_MACIVIR 2813
ETH_MACL3A00R 2853
ETH_MACL3A01R 2858
ETH_MACL3A10R 2853
ETH_MACL3A11R 2858
ETH_MACL3A20 2854
ETH_MACL3A21R 2859
ETH_MACL3A30 2854
ETH_MACL3A31R 2859
ETH_MACL3L4C0R 2850
ETH_MACL3L4C1R 2855
ETH_MACL4A0R 2852
ETH_MACL4A1R 2857
ETH_MACLCSR 2827
ETH_MACLETR 2829
ETH_MACLMIR 2881
ETH_MACLTCR 2829
ETH_MACMDIOAR 2835
ETH_MACMDIODR 2838
ETH_MACPCSR 2821
ETH_MACPFR 2803
ETH_MACPOCR 2878
ETH_MACPPSCR 2874
ETH_MACPPSIR 2877
ETH_MACPPSTTNR 2877
ETH_MACPPSTTSR 2876
ETH_MACPPSWR 2878
ETH_MACQ0TxFCR 2814
ETH_MACRWKPFR 2823
ETH_MACRxFCR 2816
ETH_MACRxTxSR 2820
ETH_MACSPI0R 2879
ETH_MACSPI1R 2880
ETH_MACSPI2R 2880
ETH_MACSSIR 2863
ETH_MACSTNR 2864
ETH_MACSTNUR 2865
ETH_MACSTSR 2864
ETH_MACSTSUR 2865
ETH_MACTSAR 2866
ETH_MACTSCR 2860
ETH_MACTSEACR 2872
ETH_MACTSECNR 2873
ETH_MACTSIACR 2871
ETH_MACTSICNR 2872
ETH_MACTSSR 2866
ETH_MACTxTSSNR 2868
ETH_MACTxTSSSR 2868
ETH_MACVHTR 2811
ETH_MACVIR 2812
ETH_MACVR 2830
ETH_MACVTR 2809
ETH_MACWTR 2806
ETH_MTLISR 2784
ETH_MTLOMR 2783
ETH_MTLQiICSR 2788
ETH_MTLRxQ0OMR 2789
ETH_MTLRxQiDR 2792
ETH_MTLRxQiMPOCR 2791
ETH_MTLTxQiDR 2786
ETH_MTLTxQiOMR 2784
ETH_MTLTxQiUR 2786
EXTI_EMR 715, 717
EXTI_FTSR 705, 709, 712
EXTI_IMR 706-707, 710-711, 713-716, 718
EXTI_PR 716-717, 719
EXTI_RTSR 705, 708, 712
EXTI_SWIER 706, 709, 713

F

FDCAN_ TXBCIE 2454
FDCAN_CCCR 2422
FDCAN_CREL 2419
FDCAN_DBTP 2420
FDCAN_ECR 2428
FDCAN_ENDN 2419
FDCAN_GFC 2438
FDCAN_HPMS 2441
FDCAN_IE 2434
FDCAN_ILE 2438
FDCAN_ILS 2437
FDCAN_IR 2431
FDCAN_NBTP 2424
FDCAN_NDAT1 2442
FDCAN_NDAT2 2442
FDCAN_PSR 2429
FDCAN_RWD 2421
FDCAN_RXBC 2445
FDCAN_RXESC 2447
FDCAN_RXF0A 2444

FDCAN_RXF0C 2443
FDCAN_RXF0S 2443
FDCAN_RXF1A 2447
FDCAN_RXF1C 2445
FDCAN_RXF1S 2446
FDCAN_SIDFC 2439
FDCAN_TDCR 2431
FDCAN_TEST 2421
FDCAN_TOCC 2427
FDCAN_TOCV 2427
FDCAN_TSCC 2425
FDCAN_TSCV 2426
FDCAN_TTCPT 2474
FDCAN_TTCSM 2474
FDCAN_TTCTC 2473
FDCAN_TTIE 2467
FDCAN_TTILS 2468
FDCAN_TTIR 2465
FDCAN_TTLGT 2473
FDCAN_TTMLM 2459
FDCAN_TTOCF 2458
FDCAN_TTOCN 2461
FDCAN_TTOST 2470
FDCAN_TTRMC 2457
FDCAN_TTTMC 2456
FDCAN_TTTMK 2464
FDCAN_TTTS 2475
FDCAN_TURCF 2460
FDCAN_TURNA 2472
FDCAN_TXBAR 2451
FDCAN_TXBC 2448
FDCAN_TXBCF 2453
FDCAN_TXBCR 2452
FDCAN_TXBRP 2451
FDCAN_TXBTIE 2453
FDCAN_TXBTO 2452
FDCAN_TXEFA 2456
FDCAN_TXEFC 2454
FDCAN_TXEFS 2455
FDCAN_TXESC 2450
FDCAN_TXFQS 2449
FDCAN_XIDAM 2441
FDCAN_XIDFC 2440
FLASH_ACR 139
FLASH_BOOT7_CUR 162
FLASH_BOOT7_PRG 162
FLASH_CCR_A 148
FLASH_CCR_B 173
FLASH_CR_A 141
FLASH_CRCCR_A 163
FLASH_CRCCR_B 179
FLASH_CRCDATA_A 165
FLASH_CRCEADD_A 165
FLASH_CRCEADD_B 181
FLASH_CRCSADD_A 164
FLASH_CRCSADD_B 180
FLASH_ECC_FA_A 166, 174
FLASH_ECC_FA_B 181
FLASH_FKEYR_B 166
FLASH_FLASH_CR_B 167
FLASH_KEYR_A 140
FLASH_OPTCCR 156
FLASH_OPTCR 149
FLASH_OPTKEYR 141
FLASH_OPTSR_CUR 151
FLASH_OPTSR_PRG 153
FLASH_PRAR_CUR_A 157
FLASH_PRAR_CUR_B 175
FLASH_PRAR_PRG_A 158
FLASH_PRAR_PRG_B 176
FLASH_SCAR_CUR_A 159
FLASH_SCAR_CUR_B 176
FLASH_SCAR_PRG_A 160
FLASH_SCAR_PRG_B 177
FLASH_SR_A 145
FLASH_SR_B 170
FLASH_WPSn_CUR_A 161
FLASH_WPSn_CUR_B 178
FLASH_WPSn_PRG_A 161
FLASH_WPSn_PRG_B 178
FMC_BCR1..4 771
FMC_BTR1..4 775
FMC_BWTR1..4 778
FMC_ECCR 791
FMC_PATT 789
FMC_PCR 786
FMC_PMEM 788
FMC_SDCMR 806
FMC_SDCR1、2 803
FMC_SDRTR 807
FMC_SDSR 809
FMC_SDTR1、2 804
FMC_SR 787
FMPI2C_ISR 1939

G

GPIOx_AFRH 493
GPIOx_AFRL 492
GPIOx_BSRR 491
GPIOx_IDR 490
GPIOx_LCKR 491
GPIOx_MODER 488
GPIOx_ODR 490

GPIOx_OSPEEDR 489
GPIOx_OTYPER 488
GPIOx_PUPDR 489

H

HASH_CR 1274
HASH_CSRx 1283
HASH_DIN 1277
HASH_HR0 1279
HASH_HR1 . . . . . .1279, 1281
HASH_HR2 1280-1281
HASH_HR3 1280
HASH_HR4 1280
HASH_IMR 1281
HASH_SR 1282
HASH_STR 1278
HRTIM_ADC1R 1434
HRTIM_ADC2R 1435
HRTIM_ADC3R 1436
HRTIM_ADC4R 1438
HRTIM_BDMADR 1446
HRTIM_BDMUPR 1444
HRTIM_BDTxUPR 1445
HRTIM_BMCMPR 1429
HRTIM_BMCR 1425
HRTIM_BMPER 1429
HRTIM_BMTRGR 1427
HRTIM_CHPxR 1407
HRTIM_CMP1CxR 1392
HRTIM_CMP1xR 1391
HRTIM_CMP2xR 1392
HRTIM_CMP3xR 1393
HRTIM_CMP4xR 1393
HRTIM_CNTxR 1390
HRTIM_CPT1xCR 1408
HRTIM_CPT1xR 1394
HRTIM_CPT2xCR 1409
HRTIM_CPT2xR 1394
HRTIM_CR1 1416
HRTIM_CR2 1418
HRTIM_DTxR 1395
HRTIM_EECR1 1430
HRTIM_EECR2 1432
HRTIM_EECR3 1433
HRTIM_EEFxR1 1401
HRTIM_EEFxR2 1403
HRTIM_FLTINR1 1440
HRTIM_FLTINR2 1442
HRTIM_FLTxR 1415
HRTIM_ICR 1420
HRTIM_IER 1421
HRTIM_ISR 1419
HRTIM_MCMP1R 1377
HRTIM_MCMP2R 1378
HRTIM_MCMP3R 1378
HRTIM_MCMP4R 1379
HRTIM_MCNTR 1376
HRTIM_MCR 1369
HRTIM_MDIER 1374
HRTIM_MICR 1373
HRTIM_MISR 1372
HRTIM_MPER 1376
HRTIM_MREP 1377
HRTIM_ODISR 1423
HRTIM_ODSR 1424
HRTIM_OENR 1422
HRTIM_OUTxR 1412
HRTIM_PERxR 1390
HRTIM_REPxR 1391
HRTIM_RSTAR 1404
HRTIM_RSTBR 1406
HRTIM_RSTCR 1406
HRTIM_RSTDR 1406
HRTIM_RSTER 1407
HRTIM_RSTx1R 1399
HRTIM_RSTx2R 1400
HRTIM_SETx1R 1397
HRTIM_SETx2R 1399
HRTIM_TIMxCR 1380
HRTIM_TIMxDIER 1387
HRTIM_TIMxICR 1386
HRTIM_TIMxISR 1384
HSEM_CnICR 473
HSEM_CnIER 473
HSEM_CnISR 474
HSEM_CnMISR 474
HSEM_CR 475
HSEM_KEYR 475
HSEM_R0 - HSEM_R31 471
HSEM_RLR0 - HSEM_RLR31 472

I

I2C_CR1 1929
I2C_CR2 1932
I2C_ICR 1941
I2C_ISR 1939
I2C_OAR1 1935
I2C_OAR2 1936
I2C_PECR 1942
I2C_RXDR 1943
I2C_TIMEOUTR 1938
I2C_TIMINGR 1937

参考資料

I2C_TXDR 1943
I2Cx_CR2 1932
IWDG_KR 1824
IWDG_PR 1825
IWDG_RLR 1826
IWDG_SR 1827
IWDG_WINR 1828

J

JPEG_CFR 1174
JPEG_CONFR0 1168
JPEG_CONFR1 1169
JPEG_CONFR2 1170
JPEG_CONFR3 1170
JPEG_CONFR4-7 1171
JPEG_CR 1172
JPEG_DIR 1175
JPEG_DOR 1175
JPEG_SR 1173

L

LPTIM_ARR 1810
LPTIM_CFGR 1805
LPTIM_CMP 1810
LPTIM_CNT 1811
LPTIM_CR 1808
LPTIM_ICR 1803
LPTIM_IER 1804
LPTIM_ISR 1802
LPUART_PRESC 2065
LTDC_AWCR 1143
LTDC_BCCR 1146
LTDC_BPCR 1142
LTDC_CDSR 1150
LTDC_CPSR 1149
LTDC_GCR 1144
LTDC_ICR 1148
LTDC_IER 1147
LTDC_ISR 1148
LTDC_LIPCR 1149
LTDC_LxBFCR 1156
LTDC_LxCACR 1155
LTDC_LxCFBAR 1157
LTDC_LxCFBLNR 1158
LTDC_LxCFBLR 1157
LTDC_LxCKCR 1153
LTDC_LxCLUTWR 1159
LTDC_LxCR 1151
LTDC_LxDCCR 1155
LTDC_LxPFCR 1154
LTDC_LxWHPCR 1151
LTDC_LxWVPCR 1152
LTDC_SRCR 1146
LTDC_SSCR 1142
LTDC_TWCR 1144

M

M7_DWT_CIDR0 3074
M7_DWT_CIDR1 3074
M7_DWT_CIDR2 3074
M7_DWT_CIDR3 3075
M7_DWT_COMPx 3069
M7_DWT_CPICNT 3067
M7_DWT_CTRL 3065
M7_DWT_CYCCNT 3067
M7_DWT_EXCCNT 3067
M7_DWT_FOLDCNT 3069
M7_DWT_FUNCTx 3070
M7_DWT_LSUCNT 3068
M7_DWT_MASKx 3070
M7_DWT_PCSR 3069
M7_DWT_PIDR0 3072
M7_DWT_PIDR1 3072
M7_DWT_PIDR2 3073
M7_DWT_PIDR3 3073
M7_DWT_PIDR4 3071
M7_DWT_SLPCNT 3068
M7_ETM_AUTHSTAT 3115
M7_ETM_CCCTL 3099
M7_ETM_CIDR0 3119
M7_ETM_CIDR1 3120
M7_ETM_CIDR2 3120
M7_ETM_CIDR3 3120
M7_ETM_CLAIMCLR 3114
M7_ETM_CLAIMSET 3113
M7_ETM_CNTRLDV 3102
M7_ETM_CONFIG 3095
M7_ETM_DEVARCH 3116
M7_ETM_DEVID 3116
M7_ETM_DEVTYPE 3117
M7_ETM_EVENTCTL0 3096
M7_ETM_EVENTCTL1 3096
M7_ETM_IDR0 3105
M7_ETM_IDR1 3106
M7_ETM_IDR10 3103
M7_ETM_IDR11 3103
M7_ETM_IDR12 3104
M7_ETM_IDR13 3104
M7_ETM_IDR2 3106
M7_ETM_IDR3 3107
M7_ETM_IDR4 3108
M7_ETM_IDR5 3108

参考資料

M7_ETM_IDR8 3102
M7_ETM_IDR9 3103
M7_ETM_IMSPEC0 3104
M7_ETM_LAR 3114
M7_ETM_LSR 3115
M7_ETM_PDC 3112
M7_ETM_PDS 3113
M7_ETM_PIDR0 3118
M7_ETM_PIDR1 3118
M7_ETM_PIDR2 3118
M7_ETM_PIDR3 3119
M7_ETM_PIDR4 3117
M7_ETM_PRGCTL 3093
M7_ETM_PROCSEL 3094
M7_ETM_RSCTL2 3109
M7_ETM_RSCTL3 3110
M7_ETM_SSCC0 3110
M7_ETM_SSCS0 3111
M7_ETM_SSPCIC0 3112
M7_ETM_STALLCTL 3097
M7_ETM_STAT 3094
M7_ETM_SYNCP 3098
M7_ETM_TRACEID 3099
M7_ETM_TSCTL 3098
M7_ETM_VICTL 3100
M7_ETM_VIPCSSCTL 3101
M7_ETM_VISSCTL 3101
M7_FPB_CIDR0 3090
M7_FPB_CIDR1 3090
M7_FPB_CIDR2 3091
M7_FPB_CIDR3 3091
M7_FPB_COMPx 3087
M7_FPB_CTRL 3086
M7_FPB_PIDR0 3088
M7_FPB_PIDR1 3089
M7_FPB_PIDR2 3089
M7_FPB_PIDR3 3090
M7_FPB_PIDR4 3088
M7_FPB_REMAP 3087
M7_ITM_CIDR0 3083
M7_ITM_CIDR1 3083
M7_ITM_CIDR2 3083
M7_ITM_CIDR3 3084
M7_ITM_PIDR0 3081
M7_ITM_PIDR1 3081
M7_ITM_PIDR2 3082
M7_ITM_PIDR3 3082
M7_ITM_PIDR4 3081
M7_ITM_STIMx 3078
M7_ITM_TCR 3079
M7_ITM_TER 3078
M7_ITM_TPR 3079
M7_PILROM_CIDR0 3057
M7_PILROM_CIDR1 3057
M7_PILROM_CIDR2 3057
M7_PILROM_CIDR3 3058
M7_PILROM_MEMTYPE 3054
M7_PILROM_PIDR0 3055
M7_PILROM_PIDR1 3055
M7_PILROM_PIDR2 3056
M7_PILROM_PIDR3 3056
M7_PILROM_PIDR4 3055
M7_PPBROM_CIDR0 3062
M7_PPBROM_CIDR1 3062
M7_PPBROM_CIDR2 3062
M7_PPBROM_CIDR3 3063
M7_PPBROM_MEMTYPE 3059
M7_PPBROM_PIDR0 3060
M7_PPBROM_PIDR1 3060
M7_PPBROM_PIDR2 3061
M7_PPBROM_PIDR3 3061
M7_PPBROM_PIDR4 3060
MDIOS_CLRFR 2277
MDIOS_CRDFR 2275
MDIOS_CWRFR 2274
MDIOS_DINR0-MDIOS_DINR31 2278
MDIOS_DOUTR0-MDIOS_DOUTR31 2278
MDIOS_RDFR 2275
MDIOS_SR 2276
MMC_CONTROL 2842
MMC_RX_INTERRUPT 2843
MMC_RX_INTERRUPT_MASK 2845
MMC_TX_INTERRUPT 2844
MMC_TX_INTERRUPT_MASK 2846

N

NBTP 2424

O

OPAMP_OR 1040
OPAMP1_CSR 1037
OPAMP1_HSOTR 1040
OPAMP1_OTR 1039
OPAMP2_CSR 1040
OPAMP2_HSOTR 1043
OPAMP2_OTR 1042
OTG_CID 2543
OTG_DAINT 2569
OTG_DAINTMSK 2570
OTG_DCFG 2561
OTG_DCTL 2563
OTG_DEACHINT 2572
OTG_DEACHINTMSK 2573

OTG_DIEPCTLx 2574
OTG_DIEPEMPMSK 2573
OTG_DIEPINTx 2580
OTG_DIEPMSK 2567
OTG_DIEPTSIZ0 2583
OTG_DIEPTSIZx 2583-2585
OTG_DIEPTXF0 2539
OTG_DIEPTXFx 2548
OTG_DOEPCTL0 2576
OTG_DOEPCTLx 2578
OTG_DOEPINTx 2582
OTG_DOEPMSK 2568
OTG_DOEPTSIZ0 2584
OTG_DOEPTSIZx 2587
OTG_DSTS 2566
OTG_DTHRCTL 2571
OTG_DTXFSTSx 2586
OTG_DVBUSDIS 2570
OTG_DVBUSPULSE 2571
OTG_GAHBCFG 2522
OTG_GCCFG 2542
OTG_GI2CCTL 2541
OTG_GINTMSK 2534
OTG_GINTSTS 2529
OTG_GLPMCFG 2544
OTG_GOTGCTL 2518
OTG_GOTGINT 2521
OTG_GRSTCTL 2527
OTG_GRXFSIZ 2538
OTG_GRXSTSP 2537
OTG_GRXSTSR 2537
OTG_GUSBCFG 2523
OTG_HAINT 2552
OTG_HAINTMSK 2552
OTG_HCCHARx 2555
OTG_HCDMAx 2561
OTG_HCFG 2549
OTG_HCINTMSKx 2558
OTG_HCINTx 2557
OTG_HCSPLTx 2557
OTG_HCTSIZx 2560
OTG_HFIR 2549
OTG_HFNUM 2550
OTG_HNPTXFSIZ 2539
OTG_HNPTXSTS 2540
OTG_HPRT 2553
OTG_HPTXFSIZ 2548
OTG_HPTXSTS 2551
OTG_PCGCCTL 2588

P

PWR_CSR 264, 268-269

Q

QUADSPI _PIR 840
QUADSPI _PSMAR 839
QUADSPI _PSMKR 839
QUADSPI_ABR 838
QUADSPI_AR 837
QUADSPI_CCR 835
QUADSPI_CR 829
QUADSPI_DCR 832
QUADSPI_DLR 834
QUADSPI_DR 838
QUADSPI_FCR 834
QUADSPI_LPTR 840
QUADSPI_SR 833

R

RCC_BDCR 377
RCC_CFGR 343
RCC_CICR 375
RCC_CIER 371
RCC_CIFR 373
RCC_CR 337
RCC_CRRCR 342
RCC_CSR 379
RCC_D1AHB1ENR 402
RCC_D1AHB1LPENR 422
RCC_D1AHB1RSTR 380
RCC_D1APB1ENR 410
RCC_D1APB1LPENR 431-432
RCC_D1APB1RSTR 387
RCC_D1CCIPR 364
RCC_D1CFGR 346
RCC_D2AHB1ENR 404
RCC_D2AHB1LPENR 424
RCC_D2AHB1RSTR 382
RCC_D2AHB2ENR 406
RCC_D2AHB2LPENR 426
RCC_D2AHB2RSTR 384
RCC_D2APB1HENR 415
RCC_D2APB1HLPENR 436
RCC_D2APB1HRSTR 391
RCC_D2APB1LENR 411
RCC_D2APB1LRSTR 388
RCC_D2APB2ENR 416
RCC_D2APB2LPENR 437
RCC_D2APB2RSTR 392
RCC_D2CCIP1R 365

参考資料

RCC_D2CCIP2R 367
RCC_D2CFGR 348
RCC_D3AHB1ENR 408
RCC_D3AHB1LPENR 428
RCC_D3AHB1RSTR 385
RCC_D3AMR 397
RCC_D3APB1ENR 419
RCC_D3APB1LPENR 440
RCC_D3APB1RSTR 394
RCC_D3CCIPR 369
RCC_D3CFGR 349
RCC_GCR 396
RCC_ICSCR 341
RCC_PLL1DIVR 355
RCC_PLL1FRACR 357
RCC_PLL2DIVR 358
RCC_PLL2FRACR 360
RCC_PLL3DIVR 361
RCC_PLL3FRACR 363
RCC_PLLCFGR 352
RCC_PLLCKSELR 350
RCC_RSR 400
RNG_CR 1189
RNG_DR 1191
RNG_SR 1190
RTC_ALRMAR 1860
RTC_ALRMBR 1861
RTC_ALRMBSSR 1872
RTC_BKPxR 1873
RTC_CALR 1867
RTC_CR 1852
RTC_DR 1851
RTC_ISR 1855
RTC_OR 1873
RTC_PRER 1858
RTC_SHIFTR 1863
RTC_SSR 1862
RTC_TR . . 261-263, 266-267, 1850, 2273-2274
RTC_TSDR 1865
RTC_TSSSR 1866
RTC_TSTR 1864
RTC_WPR 1862
RTC_WUTR 1859
RX_ALIGNMENT_ERROR_PACKETS 2848
RX_CRC_ERROR_PACKETS 2848
RX_LPI_TRAN_CNTR 2850
RX_LPI_USEC_CNTR 2849
RX_UNICAST_PACKETS_GOOD 2848

S

SAI_ACLRFR 2190
SAI_ACR1 2179
SAI_ACR2 2182
SAI_ADR 2192
SAI_AFRCR 2184
SAI_AIM 2186
SAI_ASLOTR 2185
SAI_ASR 2188
SAI_BCLRFR 2190
SAI_BCR1 2179
SAI_BCR2 2182
SAI_BDR 2192
SAI_BFRCR 2184
SAI_BIM 2186
SAI_BSLOTR 2185
SAI_BSR 2188
SAI_GCR 2178
SAI_PDMCR 2192
SAI_PDMDLY 2193
SDMMC_ACKTIMER 2351
SDMMC_ARGR 2335
SDMMC_CLKCR 2332
SDMMC_CMDR 2336
SDMMC_DCNTR 2342
SDMMC_DCTRL 2340
SDMMC_DLENR 2339
SDMMC_DTIMER 2338
SDMMC_FIFOR 2352
SDMMC_ICR 2345
SDMMC_IDMABASE0R 2354
SDMMC_IDMABASE1R 2354
SDMMC_IDMABSIZER 2353
SDMMC_IDMACTRLR 2352
SDMMC_MASKR 2348
SDMMC_POWER 2331
SDMMC_RESPCMDR 2337
SDMMC_RESPxR 2338
SDMMC_STAR 2343
SMPMI_IER 2261
SPI_DR 2135-2137
SPI_I2SCFGR 2138
SPIx_SR 2130-2131, 2134
SWO_AUTHSTAT 3027
SWO_CIDR0 3031
SWO_CIDR1 3031
SWO_CIDR2 3032
SWO_CIDR3 3032
SWO_CLAIMCLR 3025
SWO_CLAIMSET 3025
SWO_CODR 3023
SWO_DEVID 3028
SWO_DEVTYPE 3028
SWO_FFSR 3024

SWO_LAR 3026
SWO_LSR 3026
SWO_PIDR0 3029
SWO_PIDR1 3030
SWO_PIDR2 3030
SWO_PIDR3 3031
SWO_PIDR4 3029
SWO_SPPR 3024
SWPMI_BRR 2257
SWPMI_CR 2256
SWPMI_ICR 2260
SWPMI_ISR 2258
SWPMI_OR 2263
SWPMI_RDR 2263
SWPMI_RFL 2262
SWPMI_TDR 2262
SWTF_AUTHSTAT 3038
SWTF_CIDR0 3041
SWTF_CIDR1 3041
SWTF_CIDR2 3042
SWTF_CIDR3 3042
SWTF_CLAIMCLR 3036
SWTF_CLAIMSET 3036
SWTF_CTRL 3035
SWTF_DEVID 3038
SWTF_LAR 3037
SWTF_LSR 3037
SWTF_PIDR0 3039
SWTF_PIDR1 3039
SWTF_PIDR2 3040
SWTF_PIDR3 3040
SWTF_PIDR4 3041
SWTF_PRIORITY 3035
SWTF_TYPEID 3039
SYSCFG_CCCR 503
SYSCFG_CCCSR 502
SYSCFG_CCVR 503
SYSCFG_EXTICR1 498
SYSCFG_EXTICR2 499
SYSCFG_EXTICR3 500
SYSCFG_EXTICR4 501
SYSCFG_PKGR 504
SYSCFG_PMCR 496
SYSCFG_UR0 505
SYSCFG_UR10 509
SYSCFG_UR11 510
SYSCFG_UR12 510
SYSCFG_UR13 511
SYSCFG_UR14 512
SYSCFG_UR15 513
SYSCFG_UR16 514
SYSCFG_UR17 514
SYSCFG_UR2 505
SYSCFG_UR3 506
SYSCFG_UR4 506
SYSCFG_UR5 507
SYSCFG_UR6 507
SYSCFG_UR7 508
SYSCFG_UR8 508
SYSCFG_UR9 509
SYSROM_CIDR0 2942
SYSROM_CIDR1 2942
SYSROM_CIDR2 2943
SYSROM_CIDR3 2943
SYSROM_MEMTYPE 2939
SYSROM_PIDR0 2940
SYSROM_PIDR1 2940
SYSROM_PIDR2 2941
SYSROM_PIDR3 2941
SYSROM_PIDR4 2939

T

TIM1_AF1 1548
TIM1_AF2 1550
TIM1_TISEL 1555
TIM12_TISEL 1675
TIM13_TISEL 1686
TIM14_TISEL 1686
TIM15_AF1 1746
TIM15_ARR 1740
TIM15_BDTR 1742
TIM15_CCER 1737
TIM15_CCMR1 1734
TIM15_CCR1 1741
TIM15_CCR2 1742
TIM15_CNT 1740
TIM15_CR1 1726
TIM15_CR2 1727
TIM15_DCR 1745
TIM15_DIER 1730
TIM15_DMAR 1745
TIM15_EGR 1733
TIM15_PSC 1740
TIM15_RCR 1741
TIM15_SMCR 1729
TIM15_SR 1731
TIM15_TISEL 1747
TIM16_AF1 1767
TIM16_TISEL 1768
TIM17_AF1 1769
TIM17_TISEL 1770
TIM2_AF1 1630
TIM2_TISEL 1631

参考資料

TIM3_AF1 1630
TIM3_TISEL 1632
TIM5_AF1 1631
TIM5_TISEL 1633
TIM8_AF1 1551
TIM8_AF2 1553
TIM8_TISEL 1556
TIMx_ARR 1538, 1626, 1673, 1685, 1761, 1784
TIMx_BDTR 1541, 1763
TIMx_CCER 1533, 1623, 1671, 1684, 1758
TIMx_CCMR1 . . 1527, 1618, 1667, 1681, 1755
TIMx_CCMR2 . . . . . .1531, 1621
TIMx_CCMR3 1546
TIMx_CCR1 . . . . 1539, 1626, 1674, 1686, 1762
TIMx_CCR2 . . . . . 1539, 1627, 1674
TIMx_CCR3 . . . . . .1540, 1627
TIMx_CCR4 . . . . . .1540, 1628
TIMx_CCR5 1547
TIMx_CCR6 1548
TIMx_CNT 1537, 1625, 1673, 1685, 1761, 1784
TIMx_CR1 1516, 1608, 1662, 1678, 1750, 1781
TIMx_CR2 . . . . . 1518, 1610, 1751, 1782
TIMx_DCR 1545, 1629, 1766
TIMx_DIER 1523, 1614, 1665, 1679, 1752, 1782
TIMx_DMAR 1546, 1629, 1766
TIMx_EGR 1526, 1617, 1666, 1681, 1754, 1783
TIMx_PSC 1537, 1625, 1673, 1685, 1761, 1784
TIMx_RCR 1538, 1762
TIMx_SMCR 1520, 1611, 1663
TIMx_SR . 1524, 1615, 1665, 1680, 1753, 1783
TPIU_AUTHSTAT 3015
TPIU_CIDR0 3019
TPIU_CIDR1 3020
TPIU_CIDR2 3020
TPIU_CIDR3 3020
TPIU_CLAIMCLR 3014
TPIU_CLAIMSET 3013
TPIU_CURPSIZE 3007
TPIU_CURTPM 3010
TPIU_DEVID 3016
TPIU_DEVTYPE 3017
TPIU_FFCR 3012
TPIU_FFSR 3011
TPIU_FSCR 3013
TPIU_LAR 3014
TPIU_LSR 3015
TPIU_PIDR0 3018
TPIU_PIDR1 3018
TPIU_PIDR2 3018
TPIU_PIDR3 3019
TPIU_PIDR4 3017
TPIU_SUPPSIZE 3006
TPIU_SUPTPM 3009
TPIU_SUPTRGM 3007
TPIU_TPRCR 3011
TPIU_TRGCNT 3008
TPIU_TRGMULT 3008
TSG_CIDR0 2951
TSG_CIDR1 2951
TSG_CIDR2 2952
TSG_CIDR3 2952
TSG_CNTCR 2946
TSG_CNTCVL 2947
TSG_CNTCVU 2947
TSG_CNTFID0 2948
TSG_CNTSR 2946
TSG_PIDR1 2949
TSG_PIDR2 2950
TSG_PIDR3 2950
TSG_PIDR4 2948
TX_LPI_TRAN_CNTR 2849
TX_LPI_USEC_CNTR 2849
TX_MULTIPLE_COLLISION_GOOD_PACKETS 2847
TX_PACKET_COUNT_GOOD 2847
TX_SINGLE_COLLISION_GOOD_PACKETS . 2847

U

USART_BRR 2006, 2057
USART_CR1 . . . . .380, 1994, 2049, 2222, 2225
USART_CR2 . . . . . .1998, 2052
USART_CR3 . . . . . .2002, 2054
USART_DR 2017-2018, 2064, 2229-2233
USART_GTPR 2007-2008
USART_ISR 2010, 2058
USART_PRESC 2018
USART_SR 2009, 2016, 2058, 2063, 2226, 2228

V

VREFBUF_CCR 1009
VREFBUF_CSR 1008

W

WWDG_CFR 1819
WWDG_CR 1818
WWDG_SR 1820

重要なお知らせ（よくお読み下さい）

STMicroelectronics NV およびその子会社（以下、ST）は、ST製品及び本書の内容をいつでも予告なく変更、修正、改善、改定及び改良する権利を留保します。購入される方は、発注前にST製品に関する最新の関連情報を必ず入手してください。ST 製品は、注文請書発行時点で有効なST の販売条件に従って販売されます。

ST製品の選択並びに使用については購入される方が全ての責任を負うものとします。購入される方の製品上の操作や設計に関してST は一切の責任を負いません。

明示又は黙示を問わず、STは本書においていかなる知的財産権の実施権も許諾致しません。

本書で説明されている情報とは異なる条件でST 製品が再販された場合、その製品についてSTが与えたいかなる保証も無効となります。

ST およびST ロゴはSTMicroelectronics の商標です。その他の製品またはサービスの名称は、それぞれの所有者に帰属します。

本書の情報は本書の以前のバージョンで提供された全ての情報に優先し、これに代わるものです。

この資料は、STMicroelectronics NV 並びにその子会社(以下ST)が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くためにSTマイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST及びSTマイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

© 2018 STMicroelectronics - All rights reserved